Perceptual and Artistic Principles for Effective Computer Depiction

Course Notes for SIGGRAPH 2002
San Antonio, Texas
July 2002

Organizer

Frédo Durand	Massachusetts Institute of Technology

Speakers

Maneesh Agrawala	Stanford University
Frédo Durand	Massachusetts Institute of Technology
Bruce Gooch	University of Utah
Victoria Interrante	University of Minnesota
Victor Ostromoukhov	University of Montreal
Denis Zorin	New York University

Course description

This course presents connections between human visual perception and the art of picture production. Findings from perceptual and cognitive sciences are used to explore pictorial techniques used by artists as they address the challenges raised by the depiction of three-dimensional scenes on a flat media. The course introduces perceptual explanations for a variety of artistic styles. Finally, we present mechanisms that can make an image more effective, and demonstrate the adaptation of these mechanisms to computer graphics. The course is intended for both artists and scientists. Although it offers some practical insights, it is intended more as an in-breadth overview.

Prerequisites

This course assumes that the audience has a familiarity with pictures, which should include most of the SIGGRAPH audience! An open mind and curiosity for connections between art and science will enhance your experience.

Course modules

Overcoming the limitations of the medium
This module introduces the major limitations of pictures compared to the depicted scene, such as flatness or limited contrast. It provides an overview of the pictorial techniques developed by artists to cope with these issues, and then goes into more details about effective shape visualization.

Color and perspective
This module studies the two classical themes associated with art and science: color and perspective. It introduces elements of human color vision, their relation with pictures, and cultural and artistic aspects of colors. It then explores perspective from a perceptual point of view, as well as techniques to overcome perspective distortions.

Picture composition and organization
Gestalt theory has certainly been the most successful transfer from psychology to design. We review classical elements of visual organization in relation with picture com-position, and introduce recent neurological theories of aesthetic.

Beyond projection
Effective pictures involve more than a simple projection and recording of the optical flow, and successful visualizations are often far from realistic. Higher-level cognitive processes must be taken into account. This module demonstrates how computational theories of vision can explain the variety of pictorial styles, and how mid-level cognitive psychology can drive the development of effective automatic visualization software.

Table of Content

Schedule

Overcoming the limitations of the medium

8:30 Introduction (Durand)
8:40 Limitations of the medium (Durand)
9:25 Perception and representation of shape and depth (Interrante)

10:15 Break

Color and Perspective

10:30 Color (Ostromoukhov)
11:25 Perspective and perception (Zorin)

12:15 Lunch break

Picture composition and organization

1:30 Focus and gaze (Durand)
1:55 Gestalt and composition (Ostromoukhov)
2:35 Neurological theories of aesthetic (Gooch)

3:15 Break

Beyond projection

3:30 Computational vision and pictures (Durand)
4:25 Effective visualization and illustration using cognitive science (Agrawala)

Syllabus

Overcoming the limitations of the medium

Introduction (Durand, 10 minutes)

Limitations of the medium (Durand, 45 min)
The medium has limitations that can be compensated, eliminated, or accentuated.
• Flatness (depth enhancement, spatial cues)
• Limited contrast (tone mapping, photo printing, depiction of night scenes, flare)
• Static (e.g. motion blur, motion lines, multiple snapshots)

Perception and representation of shape and depth (Interrante, 45 minutes)
• Motivation
• An overview of the perception research in this area
• Techniques for enhancing the representation of shape and depth in computer graphics images

Color and Perspective

Color (Ostromoukhov, 50min)
• Objective color (physics, measurement and mechanisms of capturing light; objective color spaces)
• Subjective color (perception of color in human; stages of color processing – LMS, opponents, HSV, categories; color appearance and constancy; perceptually-based color spaces)
• Color and Meaning (color as essential visual cue and media in art; theories of artistic interpretation of color from Antiquity to our days)

Perspective and perception (Zorin, 50 min)
• Perception of 3D scenes represented by 2D images, representation of 3D scenes in different art systems.
• Perceptual distortion in images.
• Properties of perspective images; perceptual distortion in wide-angle images.
• Alternatives to perspective images: distortion-reducing nonlinear mapping of 3D to 2D; multiperspective images.

Picture composition and organization

Focus and gaze (Durand, 25 min)

- Eye movements, saccades, top-down and bottom-up mechanisms
- Studies of gaze movement and paintings (task dependent, relation with balance, fixation time and style)
- How to attract the gaze (e.g. contrast, perspective, top-down)
- Number of focal points, composition

Gestalt and composition (Ostromoukhov, 40 minutes)

- Grouping
- Closure
- Completion

Neurological theories of aesthetic (Gooch, 40 min)

Based on the polemical article by Ramachandran et al. on the "8 laws of art" and on the work by Zeki, *Inner Vision.*

- The peak shift principle (art, exaggeration and caricature)
- Perceptual grouping and binding is directly reinforcing (quick review)
- Isolation of a single module (e.g. line drawing, black and white photo, kinetic art)
- Problem solving (pictures as puzzles)
- Contrast extraction
- Symmetry (quick review)
- Generic viewpoint
- Use of metaphor

Beyond projection

Computational vision and pictures (Durand, 50 min)

- Vision as an unconscious inference of the scene
- Marr's stages of vision (from viewer-centered to object-centered)
- Object-centered vs. viewer-centered picture styles (what I know vs. what I see)
- Non-trivial mapping of 3D properties to 2D features

Effective visualization and illustration using cognitive science (Agrawala, 50 min)

- Methodology: mid-level cognitive studies
- Fundamental principles: congruence and apprehension
- A paradigm: route maps

Presenter information

Maneesh Agrawala

Stanford University
Gates 3B, Room 381 Stanford, CA 94305
Office Tel: (650) 723-0618
Home Tel: (415) 401-9987
Fax: (650) 723-0033
maneesh@graphics.stanford.edu
http://graphics.stanford.edu/ maneesh

Maneesh Agrawala is a Ph.D. student in the Computer Science Department at Stan-ford University. His primary areas of interest are computer graphics, human computer interaction, and visualization. Maneesh's current work is focused on understanding how the visual abstractions and distortions commonly found in hand-designed illustrations can improve the communicative intent of an image. He has published articles on diverse range of topics including interaction techniques for collaborative virtual environments, compression algorithms for synthetic images and animations, efficient methods for rendering soft shadows and techniques for rendering with non-traditional perspective projections. Maneesh has previously worked as a software consultant at NASA Ames and in the rendering software group at Pixar Animation Studios. He received film credits for the 1998 Disney/Pixar feature *A Bugs Lif*e. For the last several years he has been teaching digital animation through the Art Department at Stanford University. He holds a B.S. in Mathematics from Stanford University.

Frédo Durand

MIT Lab for Computer Science
NE43-242, 200 technology square, Cambridge, MA, 02139
Office Tel: (617) 253 7223
Home Tel: (617) 536 9453
Fax (617) 253 4640
fredo@graphics.lcs.mit.edu, http://gfx.lcs.mit.edu/˜fredo

Frédo Durand is a post-doctoral associate in the Laboratory for Computer Science at the Massachusetts Institute of Technology. He received a Ph.D. in Computer Science from Grenoble University for his work on visibility and applications to realistic and real-time rendering. He taught these subjects in the SIGGRAPH'01 course on visibility. His research interests now span perception and non-photorealistic rendering, and aims at the development of algorithms for the production of images that are not necessarily photorealistic, but also "effective," "expressive," or even "beautiful." He has published work on tone mapping and computer-assisted traditional drawing, and has recently written a paper providing artistic and perceptual perspective on computer depiction. At

MIT, he taught a new multidisciplinary graduate course "The Art and Science of Depiction" that explores perceptual and technical principles behind picture making.

Bruce Gooch

University of Utah School of Computing
50 South Campus Drive, Room 3190 Salt Lake City, UT 84112
bgooch@cs.utah.edu

Bruce Gooch is a graduate student in Computer Science at the University of Utah. He has worked for Ford and for Bacon and Davis Inc. conducting research in the areas of time-dependent three-dimensional magnetic fields, and corrosion defect prediction in gas and oil pipelines. Bruce earned a B.S. in Mathematics and a Masters degree in Computer Science from the University of Utah. Bruce is a National Science Foundation fellowship awardee. Bruce is the author of over a dozen research papers and technical reports in the area of non-photorealistic rendering (NPR). He is also a coauthor of the book *Non Photorealistic Rendering* which was published by A.K. Peters in 2001. Bruce has taught courses on NPR at SIGGRAPH 99 and for Disney feature films. His current research focuses on using non-photorealistic rendering to communicate shape, structure, and material properties in automatically drawn illustrations.

Victoria Interrante

Department of Computer Science and Engineering
University of Minnesota
4-192 EE/CS Bldg., 200 Union St. SE, Minneapolis, MN 55455
interran@cs.umn.edu
www.cs.umn.edu/ interran

Victoria Interrante is a McKnight Land-Grant Professor in the Department of Computer Science and Engineering at the University of Minnesota and an affiliate of the Center for Cognitive Sciences. She received her Ph.D. in 1996 from the University of North Carolina at Chapel Hill, where she studied under the joint direction of Dr. Henry Fuchs and Dr. Stephen Pizer. From 1996-1998, she worked as a staff scientist at ICASE, a non-profit research center operated by the Universities Space Research Association at NASA Langley. Her research focuses on the application of insights from visual perception, art, and illustration to the design of more effective techniques for visualizing data. Some of her current projects include: the study of texture's effect on shape perception and texture synthesis for shape representation, the study of texture perception and classification for texture synthesis in multivariate data visualization and uncertainty representation, the segmentation and tracking of vortical structures in turbulent boundary layer data, and the representation and interpretation of ambiguity in 3D models. Her work involves collaborations with researchers from a variety of departments at the University of Minnesota, including the College of Architecture and Landscape Architecture, Electrical

Engineering, Aerospace Engineering, Mechanical Engineering, and the Institute for Child Development. In 1999, she received a Presidential Early Career Award for Scientists and Engineers. She has contributed to a number of courses at SIGGRAPH on perception, visualization, and non-photorealistic rendering, and she co-organized a panel in 1999 on ”what computer graphics can learn from perceptual psychology and vice versa”.

Victor Ostromoukhov

University of Montreal, Quebec, Canada
University of Montreal, Dept.Comp.Sc.& Op.Res. C.P. 6128, Succ. Centre-Ville
Montreal
(Quebec) Canada, H3C 3J7
ostrom@iro.umontreal.ca

Victor Ostromoukhov studied mathematics, physics and computer science at Moscow Institute of Physics and Technology (Psys-Tech, MFTI). After graduating in 1980, he spent several years with prominent European and American industrial companies (SG2; Olivetti; Canon) as a research scientist. He completed his Ph.D. in C.S. at Swiss Federal Institute of Technology (EPFL, Lausanne, 1995), where he continued to work as a lecturer and senior researcher. He was invited professor at University of Washington in 1997, and research scientist at MIT in 1999-2000. He has been Associate Professor at University of Montreal, since August 2000. His research interests are mainly in computer graphics, and more specifically in non-photorealistic rendering, texture synthesis, color science, halftoning, and digital art. He has published many scientific articles on non-photorealistic rendering and color science, including six SIGGRAPH papers. He gives a graduate course at University of Montreal entitled ”Art and Science of Image Making”.

Denis Zorin

Media Research Laboratory Computer Science Department Courant Institute of
Mathematical
Sciences New York University
719 Broadway, 12th floor New York, NY 10003
dzorin@mrl.nyu.edu

Denis Zorin has received a B.S. degree from the Moscow Institute of Physics and Technology, a M.S. degree in Mathematics from Ohio State University and a Ph.D. in Computer Science from the California Institute of Technology. In 1997-98, he was a research associate at the Computer Science Department of Stanford University. His research interests include subdivision surfaces, multiresolution modeling and perception of images. The results of his work has been published in venues ranging from SIGGRAPH to SIAM Journal of Numerical Analysis. He is Sloan Foundation Fellow and a recipient of NSF CAREER and IBM Partnership Awards. In the past he has co-organized three SIGGRAPH courses.

An Invitation to Discuss Computer Depiction

Frédo Durand

Laboratory for Computer Science, MIT*

Abstract

This paper draws from art history and perception to place computer depiction in the broader context of picture production. It highlights the often underestimated complexity of the interactions between features in the picture and features of the represented scene. Depiction is not always a unidirectional projection from a 3D scene to a 2D picture, but involves much feedback and influence from the picture space to the object space. Depiction can be seen as a pre-existing 3D reality projected onto 2D, but also as a 2D pictorial representation that is superficially compatible with an hypothetic 3D scene. We show that depiction is essentially an optimization problem, producing the best picture given goals and constraints.

We introduce a classification of basic depiction techniques based on four kinds of issue. The *spatial* system deals with the mapping of spatial properties between 3D and 2D (including, but not restricted to, perspective projection). The *primitive* system deals with the dimensionality and mappings between picture primitives and scene primitives. *Attributes* deal with the assignment of visual properties such as colors, texture, or thickness. Finally, *marks* are the physical implementations of the picture (e.g. brush strokes, mosaic cells). A distinction is introduced between interaction and picture-generation methods, and techniques are then organized depending on the dimensionality of the inputs and outputs.

Keywords: Non-photorealistic rendering, computer depiction, perception, visual arts, interaction

1 Introduction

This paper discusses the general problem of *depiction*, that is, the creation of a picture that represents a scene, real or imaginary. It is an attempt to step back and initiate a discussion about the goals and context of computer depiction. There is a variety of picture production purposes, resulting in very different contexts and specificities. We show the complexity and richness of depiction, and the discussion is independent of any implementation. Our main goal is to introduce a vocabulary that will make a principled discussion possible, and to raise questions rather than providing answers. We review and build upon visual arts and perception literature. We outline important issues of depiction that we use to discuss the field of non-photorealistic rendering, and more generally, computer depiction.

Computer graphics has long been defined as a quest to achieve *photorealism*. As it gets closer to this grail, the field realizes that there is more to images than realism alone. Non-photorealistic pictures can be more effective at conveying information, more expressive or more beautiful. The recent field of *Non-Photorealistic Rendering* has developed a wealth of original and effective techniques [Gooch and Gooch 2001; Green et al. 1999; Lansdown and Schofield 1995; Reynolds 2000; Green 1999]. The flip side of this creative explosion is the difficulty of determining the structure of this area and its fundamental goals. These issues were discussed at the recent Symposium on Non-Photorealistic Animation and Rendering [NPAR 2000].

Most authors also agree that the term "non-photorealistic" is not satisfying [NPAR 2000]. The border between photorealism and non-photorealism is fuzzy, and the notion of realism itself is complex [Ferwerda 1999]. Thomas and Ollie tell an enlightening anecdote about Walt Disney [Thomas and Johnston 1981], p. 66. Disney would keep asking his animators for more *realism*, which was a cause of misunderstanding, since no one would qualify Disney's animation as realistic. Their interpretation is that he meant *convincing* rather than realistic.

The production of good realistic pictures cannot be reduced to a mechanical recording or, for that matter, to physical simulation. Realistic and non-realistic pictures need to cope with the same issues, and *pictorial* techniques, such as photographic lighting, processing, or dodging and burning, allow the image maker to control expressivity, clarity, and aesthetic, e.g. [Adams 1995; Apodaca 1999].

Moreover, many pictures represent scenes that do not actually exist. The extreme example of impossible figures shows that a picture can superficially look like the representation of a 3D reality, while there is no reasonable objective scene that

*fredo@graphics.lcs.mit.edu http://gfx.lcs.mit.edu˜fredo

can be projected to such a picture. This challenges the view where depiction proceeds unidirectionally from an object-space description to a 2D pictorial space.

Artists and other picture makers have developed a rich set of techniques to produce effective pictures. We believe that computer graphics has much to learn from this large body of knowledge, as well as from the analysis performed in the perception community. The task is not easy because the craft is often elusive or expressed in terms that are not easily translatable to algorithms.

This paper proposes a discussion of computer depiction that encompasses both photorealism and non-photorealism. Non-photorealistic rendering techniques can be different from traditional computer graphics with two respects: They introduce a broader variety of styles and they often offer original computer-human interactions. These differences will be at the heart of the discussion. We discuss the complex interplay between 3D and 2D aspects of depiction, which explains the variety of possible interaction strategies. We also introduce a classification of depiction issues into four systems that provide the basis for a coarse-grain definition of style.

As the title implies, this paper is only a first steps towards a principled discussion of computer depiction. We are working on an extension to this paper, and we hope that articles from other authors will join the discussion. We are looking forward to the reactions and comments of the readers, which will certainly strengthen and broaden the extended version of this article.

1.1 Paper overview

We first discuss vocabulary issues, and place computer depiction in the scope of computer graphics. In section 3, we discuss the complex interplay between the depicted scene and the picture. In particular, we show that depiction involves more than the unidirectional optical projection of a 3D model onto a 2D plane. This explains the variety of both picture styles and interaction strategies. In section 4, we argue that depiction is essentially an optimization problem that aims at producing the most relevant picture for a given purpose. We acknowledge that this optimization problem should most of the time be solved by the user, but the optimization nature of the process requires the design of specific tools for efficient user interaction. In Section 5, we describe a classification of basic depiction issues based on work in perception and art history. Finally, in section 6, we propose a brief review of computer depiction in the light of the previous discussion.

2 Computer depiction

We first discuss the various levels in visual representation. We describe the difference between *image*, *picture*, and *visualization*. We base this discussion on the definition of the Webster dictionary [Webster 1983]. We then place computer depiction in the context of computer graphics.

Image: An image is a "reproduction or imitation", or "the optical counterpart of an object" [Webster 1983]. An image is characterized by optical accuracy to a visual scene or object. The discussion of the various levels of accuracy is beyond the scope of this paper, see e.g. [Hunt 1995; Ferwerda 1999].

Picture: A picture is "a design or representation," or "a description so vivid or graphic as to suggest a mental image or give an accurate idea of something" [Webster 1983]. The picture is more loosely defined than the image, and it corresponds both to the graphical object and to a representation. In what follows, we use the term "picture" to describe a visual representation of a visual scene, but this representation is not necessarily optically accurate. For example, a line drawing is a picture but not an image. Moreover, as we will discuss, a picture is not necessarily the representation of an existing real scene or object. We can draw pictures of dragons or one-eyed monsters, although none of us has ever seen such animals. Depiction is the production of a picture that represents a scene (real or imaginary).

Visualization: Visualization is "the act or process of interpreting in visual terms or of putting in visual form" [Webster 1983]. The main difference between visualization and depiction is that a visualization can represent visually data or subjects that are not themselves visual. Visualization therefore mainly relies on *metaphors*. Depiction is a special instance of visualization, and realistic image production is a special instance of depiction.

Non-photorealistic: "Non-photorealistic" is a loosely-defined term. It should be used only to qualify a pictorial style. The only meaning of non-photorealistic is that the picture does not attempt to imitate photography and to reach optical accuracy.

Rendering: The field of rendering is concerned with the development of algorithms and numerical methods for the production of pictures given a scene description. Rendering deals with purely automatic techniques and is traditionally not concerned with user interaction.

Non-photorealistic rendering: The field of non-photorealistic rendering has suffered from a loose definition. In particular, it mixes rendering aspects (generation of pictures) together with interaction issues. This is why we advocate the use of a more general term, *computer depiction*.

Computer depiction: Computer depiction deals with all aspects of picture production, and in particular it is concerned with both rendering and interaction. It encompasses both photorealistic and non-photorealistic styles. We will advocate in this paper that most depiction issues are common to realistic and non-photorealistic styles, and that photorealistic rendering is only a special instance of depiction.

3 From 2D to 3D and back

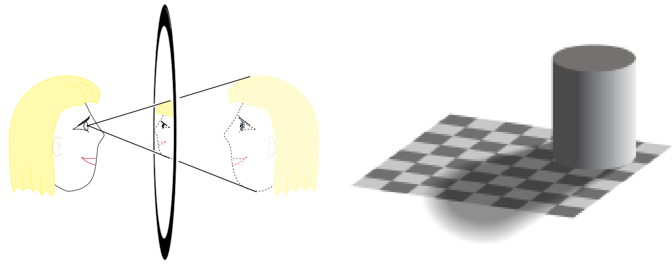
Traditional computer graphics is a unidirectional projection from a 3D objective scene to a 2D image. The typical object-space inputs are a 3D geometric description of the objects, their material properties and light sources. Perspective matrices, hidden-surface removal, and lighting simulation are then used to project this model onto the 2D image. In this section, we challenge this view, and show that the relation between the object-space scene and the 2D picture can be quite complex, and that picture generation is not unidirectional, but involves many back-and-forth exchanges, feedback, constraints, and goals linking the scene and the picture. This is related to the complexity of the human visual system, and to the dual nature of pictures, both flat objects and representation of an objective scene.

3.1 Intrinsic vs. extrinsic

The notions of *invariant* and *constancy* are crucial in studying vision and the complex dualism of pictures. Invariants are *intrinsic* properties of scenes or objects, such as reflectance, as opposed to accidental *extrinsic* properties such as outgoing light that vary with, e.g., the lighting condition or the viewpoint. Constancy is the ability to discount the accidental conditions and to extract invariants. For example, color constancy consists in discounting the color of the illuminant: We see a red apple as red under illuminant with very different color temperatures, although the physical stimuli have very different objective chromaticities. Size constancy allows us to infer the true size of objects instead of their accidental visual angle: An object does not seem to become smaller when it goes away, because our visual system is somehow able to compensate for foreshortening due to distance.

Constancy is not perfect, but it works surprisingly well. In fact, constancy is usually so efficient that we hardly have conscious access to the extrinsic information present in the retinal image. We do not experience visual angles, we experience objects with their true size and shape. A classical example is when we look at our face in a mirror: We do not realize that the surface of the image on the mirror is half our real size [Gombrich 1956] (Fig. 1(a)). Similarly, we can estimate the intrinsic color of an object, but it is very hard to assess the color of the light leaving it (Fig. 1(b)). This is, for example, explained by Land's Retinex theory [Land 1977].

When looking at a picture, constancy might not operate the same way as when looking at the scene. For example, chromatic adaptation does not function equally. This is why white balance is needed for video cameras, or why different films are required for outdoor photography and for indoor photography without flash. Indeed, when we look at a picture, our visual system adapts to the color of the illuminant of the room in which we look at the picture. In contrast, we are able to discount the intensity of the illuminant in a picture, as demonstrated by Fig. 1(b).



(a) (b)

Figure 1: (a) Mirror illusion. The size of our reflection on the surface of a mirror is half our size. (b) In this picture, the white cells in the shadow of the cylinder have the same grey level as the black cells in full light. After an illusion by Ted Adelson.

Constancy has caused fundamental difficulties in Western depiction. Constancy is what makes perspective or realistic shading challenging: Because we do not experience visual angles, foreshortening is hard to depict, and because we do not experience absolute extrinsic light intensity but subjective intrinsic lightness, the naive eye is not good at evaluating shading effects. The goal of impressionist painting was to get closer to the transient extrinsic qualities of scenes, and we know how hard an endeavor it was. As noted by Gombrich [Gombrich 1956], many realistic painters find it hard to depict a scene without the help of a photograph to visualize the accidental appearance. David Hockney also hypothesizes that painters as early as the 15th century have used optical devices in order to reach realism [Hockney 2001].

In contrast, other styles produce pictures that are closer to the intrinsic invariants than to the extrinsic appearance. Hogarth [Hogarth 1981] tells the anecdote of a Chinese emperor looking at the portrait of a Western king, painted with strong Baroque chiaroscuro (use of light and shade). Commenting on the shadowed half of the face, the emperor asked about the king's disability. For him, a painting represents intrinsic or essential characteristics, and this black half of the portrait had to mean that the king had lost an eye and half of his face. Invariants are often represented directly, not only because invariants are easier for us to consciously access, but also because invariants are by nature a "better," or at least more immutable representation. Some authors strongly believe that the goal of art is the same as the goal of the brain: to extract the *essential* [Zeki 2000; Ramachandran and Hirstein 1999].

The difference can also be stated in terms of depicting "what I see" (extrinsic) as opposed to depicting "what I know" (intrinsic). It suffices to read the opposite statements made by the 19th century painter Turner who claimed, "My business is to paint not what I know, but what I see," and by the 20th century Picasso who declared, "I do not paint what I see, I paint what I know."

In fact, most pictures are hybrid, and managing the balance between extrinsic and intrinsic properties is one of the keys to good depiction. For example, one-point perspective provides an extrinsic view, but preserves the intrinsic orientation of line parallel to the picture plane (horizontals and verticals of the picture). Renaissance chiaroscuro shading renders shapes using light and dark, but emphasizes the intrinsic color, rather than some accidental lighting, as opposed to Baroque tenebrism.

A common way to solve the dilemma between extrinsic and intrinsic characteristic is to choose the depiction such that the extrinsic characteristics match the intrinsic ones. For example, the confusion faced by the Chinese emperor is often avoided by first using a frontal view which preserves the symmetry of the face, and in cinema and photography, by using a *fill light* that illuminates the shadowed areas [Millerson 1991]. Note that this means choosing the depiction situation (constrained viewpoint, additional light source) in order to improve the picture: The 2D picture influences the depicted scene. This is reminiscent of quantum theory and the influence of the observer on the observation. We will come back to these issues.

3.2 Complex mapping

Before discussing further the complex interaction between the picture and the represented scene, and the preservation of intrinsic properties, consider the following striking counter-example to the view of pictures as geometric projections (Fig. 2). When shown a 6-color die, 7-year old children tend to draw it as a single rectangle with 6 vertical or horizontal stripes [Willats 1997]. The presence of all the colors inside the rectangle rules out the possibility that it may correspond to the projective view of one face. A similar demonstration involves a numbered die: All the numbers are drawn in the rectangle. This demonstrates that the children have mapped the notion of a 3D object with corners, a cube, onto a 2D object with corners, the rectangle.

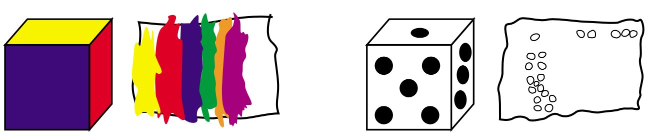


Figure 2: Depiction of a die by children at age 6-7. Redrawn after [Willats 1997].

This might seem like a very odd example due to the lack of skill. In fact, this is a caricatural but paradigmatic demonstration of a very fundamental principle of depiction: Depiction is not about projecting a scene onto a picture, it is about *mapping properties* in the scene to *properties* in the picture. Projection happens to be a very powerful means to obtain relevant mappings, but it is not the only one, and it is not necessarily the best one.

Consider the drawing of a sphere. Linear perspective projects a sphere onto an ellipse (unless it is in the center of the image). However, most pictures represent off-center spheres as disks, and the projectively correct ellipse is experienced as distorted [Pirenne 1970; Zorin and Barr 1995]. This is because a perfectly symmetric 3D object should be depicted as a perfectly symmetric 2D object.

We do not advocate abandoning projection matrices. Instead, we suggest that they are only a means, to obtain efficiently a reasonable solution to a much more intricate problem than it seems. And from an epistemological point of view, we should not confuse the means and the end, especially since linear perspective can produce artifacts that cannot be understood from the point of view of projective geometry.

An important issue is the preservation of invariants [Hagen 1986], and whether a given 3D property is preserved by the mapping to the 2D picture. Some systems preserve alignment (e.g. the projective systems commonly used in graphics), some also preserve parallelism (orthographic projection), but for example, perspective does not preserve relative size or the symmetry of spheres.

An interesting aspect of the 2D/3D mapping arises for the line drawing of smooth surfaces. The occluding contour of a surface depends on its differential properties [Koenderink 1990]. As illustrated in Fig. 3, convex regions of the surface project as convex outlines, saddle regions project to concave contours, and concave parts can never be represented because they are occluded. However, Willats shows that some artists map the concavity property of the 3D surface to a concave contour in the picture, in order to denote the property of "concavity" [Willats 1997].

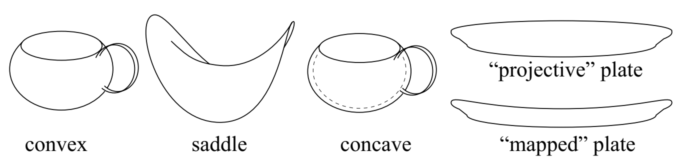


Figure 3: Line primitives and differential geometry mapping. A concave shape, e.g. interior of the cup, is never visible. Nonetheless, some artists choose to depict the concave interior of a plate as a concave 2D contour.

3.3 Primary and secondary space

We now come to the distinction between the expression of representation in the primary vs. secondary space. This point is very counter-intuitive from a computer graphics point of view. The primary space is the 3D objective space, while the secondary space is the picture. This was introduced for the discussion of projection (or drawing) systems [Booker 1963; Dubery and Willats 1983; Willats 1997], and we will extend it to other issues. As we will see, computer graphics has been developed in terms of primary space, while secondary space can provide more flexibility and fits better the mental process of picture production. It can, for instance, explain the difference between one-point, two-point, and three-point perspective, although these three projections fundamentally correspond to the same geometrical operation in the primary space.

The geometry of projection is usually expressed in terms of the intersection between 3D light rays and a picture plane. This is called the *primary geometry* of the projection system [Willats 1997; Booker 1963]. It can also be expressed directly in the pictorial space, in terms of *secondary geometry*. Secondary geometry can be seen as a set of rules that teach how to draw various features of the scene, in particular straight lines of the three main axes. For example, perspective projection can be described by stating that distant objects are foreshortened and that orthogonals to the picture plane converge to a vanishing point. Essentially, these are two different descriptions of the same geometrical operation.

Descriptions in secondary geometry are usually less compact than in primary geometry, and harder to adapt to computer graphics. In particular, it is challenging to devise a sufficient and coherent set of generative rules in terms of secondary geometry. However, secondary geometry provides a better account of the mental processes, it permits the expression of a larger variety of drawing and projection systems, and it is more amenable to the description of the evolution in art history and children drawing [Willats 1997]. Moreover, the complex mapping between 3D and 2D described above are more naturally described in secondary space.

There are two distinct but related difference that make secondary expression more powerful: The expression in picture space makes it easier to express the relation between scene and picture, and the decomposition into a variety of rules for the mapping of various features permits more flexibility. Complex systems often can be described only in terms of secondary geometry. This is the case if only the topology of the scene is preserved, e.g. for subway plans or route maps [Agrawala and Stolte 2001]. In this case, drawing is mostly a purely 2D layout problem.

Introducing concepts from secondary geometry is important to provide a larger variety of options, and to design better user interfaces. There is a continuum from pure linear perspective to topological drawing that fit to different purposes, and depending on the context and goals, an expression in primary or secondary geometry will be more useful. And a single technique can mix primary and secondary aspects, such as through-the-lens camera control [Gleicher and Witkin 1992], where user interaction in secondary space specifies a camera that is stored internally in primary space.

The distinction between primary and secondary spaces was initially developed to discuss projection systems, but it can be extended to all aspects of depiction. Line drawing is an interesting example. Its primary-space expression is the projection of edges and occluding contours onto the picture plane. However, as shown by e.g. Huffman [Huffman 1971], Clowes [Clowes 1971] and Guzman [Guzman 1971], there is a set of sufficient rules in the picture plane that characterize the line drawing of a 3D objects. These rules in the secondary space describe vertices, edges, T-vertices and end-junctions, and ensure that the direction of occlusion is coherent within the picture. Any picture that respects these rules corresponds to the image of a 3D object. Willats showed that artists intuitively use these rules, and that breaking them results in less realistic pictures [Willats 1997]. There are a variety of impossible figures based on this: They respect the rules locally, but the global coherence of occlusion is not respected (Fig. 4). It would be interesting to assist the user of a line-drawing system to obtain locally consistent or globally consistent line drawings.

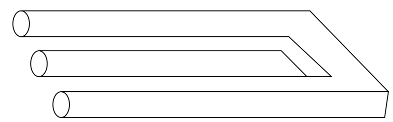


Figure 4: Illusion that respects some secondary space rules of line drawing, but not global occlusion consistency.

Colors are usually assigned using a primary space specification, through realistic shading and lighting: Incoming light and BRDFs result in the visible color at a given point. In contrast, an example of shading purely in the secondary space occurs in the depiction of a sphere using an illustration software (Fig. 5). A disc is drawn, and a concentric gradient is specified in picture space, resulting in a convincing sphere. Note that the projection is specified in secondary geometry too: The disc is drawn in picture space, regardless of any 3D to 2D projection.

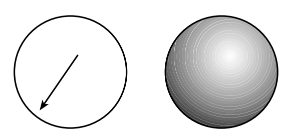


Figure 5: Shading a sphere in picture space.

Similar to the projection, it is fruitful to express realistic shading and lighting in the secondary space and to separate them into various phenomenological rules rather than relying on the more compact rendering equation [Kajiya 1986]. It may seem anti-scientific to break a set of phenomena that can be described by a single compact expression down to a set of phenomenological entities. However, as discussed above, this can provide a better account of the mental process and lead to better user control, and also to a larger variety of styles.

3.4 1kg of 2D, 1kg of 3D, which is heavier?

Our discussion challenges the importance of the primary space. It hints that in many cases, depiction happens mainly in the picture plane. We can go further and wonder about the chicken-and-egg problem between 2D and 3D. Depiction can be seen as a pre-existing 3D reality projected onto the 2D plane, or as a 2D pictorial feature that is superficially compatible with an hypothetic 3D scene. While this may look at first sight like splitting hairs, it reflects very different depiction purposes and contexts. In many cases, the 3D aspects are incidental, and the only significant characteristic lie in the final picture.

Classical computer graphics starts from the 3D model and simulates a view. The typical applications are driving or flying simulation, or architecture rendering, where the fidelity to a given objective scene is paramount. On the other end of the spectrum, illustrations such as the figures in this paper are drawn purely in secondary space, and only imply an hypothetical 3D scene. The depiction of a sphere in Fig. 5 is a good example where depiction is specified only in secondary space. It is a common assumption in graphics that the latter case is an exception, and that most images are projections of 3D scenes.

Consider however the case of movies. For close-ups of dialogues, if the two actors have a different size, the technique of *trenching* is used. A hole is dug in the ground to lower the tallest actor, or the more vertically challenged is put on a box to make their faces level. This means that the 3D scene is altered in order to obtain a good composition in 2D picture space. We are very far from a simulation going unidirectionally from 3D to 2D. A simpler example is group photography: People are asked to take a 3D position that is motivated by visibility issues in the final picture.

Most depiction situations present a mix of 3D and 2D specifications. Acknowledging this richness can result in original techniques that are more relevant to specific contexts. Examples include view-dependent models, where a 3D model is deformed with the only goal of obtaining the desired 2D picture [Rademacher 1999; Cohen et al. 2000; Martin et al. 2000], or projective drawing that combines the power of 3D notions with the ease of use and flexibility of 2D drawing [Tolba et al. 2001].

4 Depiction as optimization

We have argued that depiction involves complex interactions between the scene and the picture, and that different contexts result in very different depiction strategies. Because pictures always have a purpose, producing a picture is essentially an *optimization process*. Depiction consists in producing the picture that best satisfies the goals. The specification of these goals and the assessment of the quality of the result are obviously intricate issues that go well beyond the scope of computer graphics. Nonetheless, understanding the optimization nature of picture generation has important consequences. This ties up with the previous discussion, in that it invalidates the simple unidirectional projective view of computer graphics.

Vision is an ill-posed inverse problem. It is usually assumed that computer graphics is the corresponding direct image generation, and that it is therefore simple. However, to fully account for the diversity of picture styles and to understand the mental processes involved, one has to think of depiction as the *inverse of the inverse* problem. Indeed, representing a given scene consists in producing a picture that induces a similar impression to beholders as they would have in front of the real scene (Fig. 6). Informally, if we note $V(S)$ the vision operator for a stimulus S, we want $V(S_{picture}) \approx V(S_{scene})$ which means $S_{picture} \approx V^{-1}V(S_{scene})$. If a strict definition is taken for "similar," and if imaging and vision were invertible operations, depiction would be easy and would be reduced to optical simulation.

Unfortunately, vision is a very complicated operator, it is non-invertible since the problem is ill-posed. Moreover, very different stimuli can depict the same scene. For example, a line drawing is a very different optical stimulus from a photograph, but they can as efficiently represent the same scene [Ryan and Schwartz 1956].

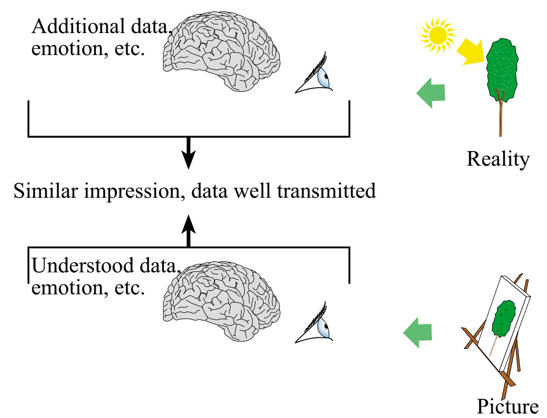


Figure 6: Depiction as the inverse of an inverse problem.

Moreover, pictures have limitations compared to the real optical flow [von Helmoholtz 1881; Barbour and Meyer 1992]: They are flat, of finite extent, often static, and they have a limited gamut and contrast. These additional constraints are most challenging for realistic images. A very important consequence is that the direct recording of the optical flow (i.e. photography) might not result in the most realistic image. This can be due to, e.g., the absence of depth cues, or to the limited contrast. An image where the contrast at the occluding contour is reinforced might provide a more faithful depth impression, because this compensates for the lack of stereovision or accommodation cues. This is an example of *pictorial techniques* to compensate for the limitation of the medium. A missing cue is rendered through a different perceptual channel (here, stereovision is compensated through occlusion).

Most pictures do not only represent visual properties of the scene. The purpose of the picture can be a message, collaborative work, education, aesthetic, emotions, etc. These additional goals set new constraints on depiction, in terms of clarity, representation of intrinsic vs. extrinsic qualities, 2D layout, etc. Added to the aforementioned limitations of the medium and to the complexity and ambiguities of vision, this results in a very complex optimization problem, where the function to minimize and the degrees of freedom depends heavily on the context and goal. The art and craft of picture creation aims at optimizing the final picture according to a goal, under given constraints set by e.g. the medium, the social context, the artistic fashion. Artists usually do not produce a picture ex nihilo, they work on studies and sketches, and the final picture is retouched until it looks right.

One way to look at realistic graphics is that it is one of the rare cases where the optimization goal (physical accuracy in the primary space) yields a direct analytical formulation of the optimization process. This does not mean that the problem is easy to solve, but at least that it is reasonably simple to state [Kajiya 1986]. We have seen that this might not result in the most realistic picture. Nonetheless, this provides a very close initial estimate, and additional techniques such as model touch-up, photographic lighting, make up, photo processing, can be seen as refinement steps, similar to gradient descent.

In traditional 3D graphics, optimization is dealt with by the user in a feedback loop: The user generates an image, views it, assesses the 2D qualities, reverse-engineers the image, and then performs the hopefully necessary 3D modifications. A new image is rendered, and the process is iterated. We do not propose to replace the feedback loop performed by the user by optimization software. In most situation, this would prove impossible because of the difficulty to translate and solve for artistic goals. Moreover, many users want to keep control of the process. However, there are cases where software optimization proves useful, e.g. [Agrawala and Stolte 2001; Gooch et al. 2001; Shacked and Lischinski 2001; Hausner 2001; Kaplan and Salesin 2000; Glassner et al. 1995; Hertzmann 2001].

Our argument is at a more "philosophical" level: We need to recognize the complexity of the depiction problem and its optimization dimension in order to develop relevant solutions. There are essentially three strategies to solve this optimization problem: The user can solve it, the computer can solve it, or the solution might involve both user and computer decisions. All approaches are of course not contradictory and can be blended. Each strategy raises a number of issues, which we only briefly outline.

If the user solves the optimization and basically explores the parameter space:

- Provide relevant degrees of freedom in the rendering algorithm, e.g. [Barzel 1997].
- Linearize parameter space. In particular, the controls should be predictable and uniform, that is, a small change in a parameter should result in a predictable change, and the perceptual magnitude of the change should be uniform. Good examples are the CIE-LAB color space [Fairchild 1998] or the perceptually-uniform gloss model by Pellacini et al. [Pellacini et al. 2000].
- Provide controls in image space to control the primary space (inverse kinematics [Parent 2001], painting with light [Schoeneman et al. 1993], through-the lens camera control [Gleicher and Witkin 1992]).
- Provide high-level controls directly related to the goals and constraints of the user, e.g. [Durand et al. 2001].
- Develop purely secondary-space pictorial techniques. Since the standard "projection" is often close to the desired solution, a small perturbation is often enough to obtain the desired picture. Examples include digital dodging and burning tools or tone mapping, e.g. [Tumblin 1999].
- Speed up the internal loop to provide faster feedback to the user, e.g. [Guenter et al. 1995; Gershbein and Hanrahan 2000].

Design galleries is a typical tool to help users explore a complex parameter space. The computer performs all the computations based on the primary parameter space, and presents a choice to the user based on the secondary characteristics of the output [Marks et al. 1997].

If the computer solves the optimization:

- Define the energy function. This involves cognitive psychology and understanding of traditional techniques, e.g. [Seligmann and Feiner July 1991; He et al. 1996; Agrawala and Stolte 2001; Shacked and Lischinski 2001].
- The traditional optimization issue: exploration of a highly non-linear parameter space. This ties up with the need for predictable and uniform parameter spaces.

The general case is mixed. The computer has to take decisions automatically, but the user wants to keep some control and influence the decisions. This is for example one of the exciting issues raised by the convergence of games and movies: The computer has to respond automatically to the user interaction, but the equivalent of the movie director want to keep control of the style of pictures. The technique by Hertzmann is an example where the user has some high-level control on stylistic parameters [Hertzmann 1998]. The amount of user vs. computer control is an exciting issue in designing computer depiction systems.

5 Organizing computer depiction

The difficulty in classifying and comparing non-photorealistic rendering techniques is parallel to the difficulty faced in picture studies to discuss very different styles of pictures. This is why we introduce and adapt the classification developed by Willats [Willats 1990; Willats 1997]. He builds upon various fields to propose a structural study of representation that encompasses not only fine art from all eras and civilizations, but also any kind of picture, be it a child's drawing, a traffic sign, a repair manual or a logo. His initial goal was to provide a structured language to describe "how" these pictures are different. While introducing new vocabulary, he notes that "Physics as a science simply did not exist before the introduction of a precise terminology (...) New words had to be introduced for new concepts and old words (...) had to be given a new and precise meaning" [Willats 1997], page 5.

5.1 Representation systems

The central thesis of Willats is that depiction (or representation in his terms) can be described in terms of two systems: the *drawing* systems and the *denotation system*. In Willats's words, "the drawing systems are systems such as perspective, oblique projection and orthogonal projection that map spatial relations in the scene into corresponding relations in the picture" [Willats 1997], page 2. "The denotation systems map (...) *scene primitives* (...), into corresponding *picture primitives*, such as regions, lines, or points," [Willats 1997], page 4. To summarize Willats's theory, depiction involves two kinds of decisions: which primitives to use (denotation), and where to put them (drawing).

The term "drawing" in Willats's classification introduces potential ambiguities, because it is used to describe both the spatial system and the line-drawing denotation system (use of 1D primitives). This can be explained by the historical role of line drawing in Western art. It was used for studies of paintings, in order to find the right composition and the right spatial mapping for the various features. It is only recently that line drawing has acquired the status of art form. We use the term "spatial" instead of "drawing," and "primitive" instead of "denotation," because these terms carry less ambiguity.

We extend Willats's framework, and we decompose depiction into four kinds of systems: spatial, primitives, attribute, and marks. An information processing point of view would state that direct picture production goes through a pipeline of 4 stages: spatial mapping, choice of primitives, attributes of these primitives, and mark implementation. However, we have shown that picture production is not always direct and that the mapping involved can be intricate. The pipeline metaphor is only meaningful in the very particular case of the mechanical rendering from 3D to 2D.

Spatial system: The spatial system deals with the spatial properties of the picture. In the case of direct image generation, it maps 3D spatial properties to 2D spatial properties. Note that the mapping can be implicit, in particular when the picture does not represent a real 3D scene.

In traditional computer graphics, the spatial system is handled by projective matrices that project 3D coordinates onto 2D picture coordinates. However, more elaborate spatial systems have been used, e.g. non-linear perspective [Bourgoin et al. 1995; Löffelmann and Gröller 1996], multiple perspectives in a single image [Agrawala et al. 2000], or purely topological spatial layout [Agrawala and Stolte 2001].

Primitive system: The primitive system maps primitives in the object space (points, lines, surfaces, volumes) to primitives in the picture space (points, lines, regions). In contrast to Willats's classification, we introduce the distinction between continuous and discrete point primitives. A discrete point primitive is for example the symbol representing a station in a subway map, while the pixels in a ray-traced image are continuous point primitives.

The primitive system has long been neglected because the traditional systems are trivial. For example, in classical computer graphics, the primitive system maps visible points in the scene to point primitives in the image. Willats calls this *optical* denotation. In the *line drawing* primitive system, 1D lines in the picture denote silhouettes of the scene. Silhouette extraction is the main primitive issue in NPR, e.g. [Elber and Cohen 1990; Gooch 1998; Markosian et al. 1997; Zorin and Hertzmann 2000; Saito and Takahashi 1990; Raskar and Cohen 1999; Curtis 1998; Buchanan and Sousa 2000]. There are also non-trivial primitive systems, for example ball-and-stick drawing, where an elongated volumetric cylindrical shape such as an arm is mapped to a line primitive (Fig. 7) [Willats 1997; Hodgins et al. 1998].

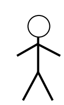


Figure 7: Ball-and-stick drawing of a man.

Attribute system: The attribute system assigns visual properties such as color, texture, thickness, transparency, wiggleness, or orientation to picture primitives. The list of relevant visual attributes depends on the primitive, on the mark system and on the context (see below).

Willats discusses attributes only for the optical denotation system (continuous point primitive), but attribute issues occur for all primitive systems. It is, for example, common in line drawing to assign the color and thickness of strokes to depict shading. In realistic graphics, the attribute system is physically-based lighting and shading. Recent work on attribute systems include [Williams 1991; Corrêa et al. 1998; Gooch et al. 1998; Sloan et al. 2001; Gooch et al. 1999].

Mark system: The mark system is the implementation of the *primitives* placed at their *spatial* location with the corresponding *attributes*. The mark system describes the physical strokes in traditional depiction, and in rendering, it is responsible for medium simulation (e.g. oil painting, pencil brush, watercolor, engraving).

Traditional computer graphics simply uses pixels as marks, and the correspondence between primitives and marks is direct. Nevertheless, there is a fundamental difference between picture primitives, and marks that are only the physical implementation of primitives. For example, a line primitive can be implemented as a series of dot marks, e.g. in a mosaic, and a paintbrush can be used to implement either 1D long brush strokes or 0D pointillism. Many mark techniques have been presented in NPR, e.g. [Curtis et al. 1997; Sousa and Buchanan 1999; Ostromoukhov 1999].

5.2 Depiction cannot be reduced to systems

The systems presented above permit a principled coarse-grain decoupling of depiction issues. They are crucial to understand the various aspects of depiction. Nevertheless, it is equally important to discuss the complex interaction between these systems, and the inherent limitations of the decomposition of depiction into sub-tasks. This framework does not provide a strict and complete classification, due to the richness and complexity of the endeavor.

We first insist that these systems can be more complex than a simple projection from 3D to 2D. They assign mappings between the object space and the picture space. The mappings can be non-trivial (as in the die example Fig. 2 or as for the plate in Fig. 3). They can also be implicit, from 2D to 3D when depiction is specified purely in picture space as in Fig. 5.

There can be very rich interactions between the various systems. The simplest interactions between systems is the constraints set by one system on another. In particular, the mark system imposes constraints on the range of possible primitives and attributes (color gamut, physically-possible thickness). Each primitive also comes with a different set of attributes. Thickness, for example, is not relevant for regions or continuous points.

But there can be less mechanical interactions as well. For example, the decision to include a primitive (such as a line in line drawing) might depend on its spatial proximity with other primitives to avoid cluttering. And the balance and composition of an image relies on the spatial layout, on the arrangement of colors and intensities, but also on the saliency of various primitives. The art and craft of effective picture making relies on the rich and complex interaction between all aspects.

Consider for example the variation of color inside a single mark such as oil painting or watercolor. Depending on the point of view, this can be viewed at three different levels. It can be directly specified by the attribute system. It can also be a simple stroke texture purely at the mark level. It can also be partially controlled by the attribute system via a variance of color attribute that controls the amount of color variation inside a stroke Hatching is another example where the continuous point and line drawing primitive systems interact in a very intimate and rich way with the stroke mark system. The same mark primitive is used to implement both hatching and silhouettes [Salisbury et al. July 1994; Winkenbach and Salesin 1994; Salisbury et al. 1997; Durand et al. 2001], and in master's drawing, it is hard to tell one from the other.

Decomposing a given picture into these four sub-systems can be ill-posed. However, they provide a vocabulary to discuss basic techniques and to relate computer depiction to traditional picture production.

5.3 Classification

Now that we have introduced important issues and vocabulary, we are about to present a brief survey of low-level computer depiction techniques, focusing on technique categories rather than on depiction style. This survey is partial because the domain is vast, but we hope that it outlines major issues. NPR research is usually organized according to the kind of systems (interactive, automatic, 2D or 3D) or depending on the simulated media. These classifications are useful and correspond to some of the issues discussed above. However, we believe that it is also important to decompose computer depiction software into lower-level modules performing precise tasks. This is crucial to permit the cross-integration of different techniques, and to provide a better account of the potential of each method.

We use our classification of representational issues: spatial, primitive, attributes, and marks. Techniques can then be classified according to their representation *style* and to their inputs/outputs. In this paper, we focus on the inputs and outputs. For each technique, the main inputs can be in 3D primary space, or in 2D secondary picture space, or hybrid: e.g. z- or G-Buffer, which we will denote loosely as $2.5D$. This classification is related to the difference between object precision and image precision [Sutherland et al. 1974], and to the difference between discrete and continuous representations.

We will use a simple notation $nD \rightarrow mD$ to describe a technique with inputs in the n-dimensional space and output in the m-dimensional space. A method can have two distinct goals: actual picture generation or interaction. Note that by interaction, we not only mean user interaction, but also computer-aided techniques such as optimization that take depiction decisions. Straight picture generation globally goes from 3D to 2D (but can also use some $3D \rightarrow 3D$ or $2D \rightarrow 2D$ techniques). In contrast, interaction can include some direct 3D or 2D manipulation, but may also include feedback from $2D$ to $3D$. For example, the through-the-lens camera control allows a user to control the 3D camera using 2D interaction [Gleicher and Witkin 1992].

This classification according to the dimensionality of the inputs and output accounts for the recent diversity in interaction strategies, and permits the discussion of recent subfields such as image-based modeling ($2D \rightarrow 3D$) and rendering ($2.5D \rightarrow 2D$), or sketch-based modeling ($2D \rightarrow 3D$) .

Additional important criteria are whether the method sets absolute or relative properties, and if it is global or local. For example, lighting and shading set absolutes color values, while atmospheric perspective is more a relative modification of the color. And methods can be used globally on the whole picture, or vary spatially, or be limited to a subset of objects in the scene.

Some complex techniques might prove hard to fit strictly in our classification, in that they involve intimate coupling between different systems. As we have seen, depiction is quite an intricate endeavor, and it is unlikely that a single framework will rigidly account for the variety of solutions. However, our classification provides a vocabulary and a reference to discuss such complex or original systems. We believe that a principled discussion of basic techniques is a necessary step to be able to discuss more complex solutions, and we encourage readers to devise new depiction styles and new interaction solutions, by building upon our classification or by building upon the limitations of our classification. Moreover, computer depiction should not be limited to the imitation of traditional techniques and media, but has the potential to produce novel forms of depiction.

6 A tentative overview

In what follows, we simply illustrate the descriptive potential of our framework and discuss examples of work in the various categories. The discussions are unfortunately brief, and are not meant as a comprehensive survey. Instead, they are provided as additional illustrations of the concepts discussed so far. This section is less conversational, and is more intended as the skeleton of a larger discussion. We invite the reader to pursue the reflection along those lines, and we are working on an extended version of this paper.

6.1 Spatial

Traditional $3D \rightarrow 2D$ spatial techniques include linear perspective and orthographic projection expressed in primary geometry, e.g. [Carlbom and Paciorek 1978]. Non-linear spatial systems have also been used [Max 1983; Gröller 1994; Löffelmann and Gröller 1996; Glaeser and Gröller 1999; Bourgoin et al. 1995; Levene 1998; Agrawala et al. 2000; Rademacher and Bishop 1998; Glassner 2000; Chu and Tai 2001].

On the other hand, $2D \rightarrow 2D$ techniques consist in warping within the picture plane [Gomes et al. 1998; Litwinowicz 1991]. A good illustration of $2D \rightarrow 2D$ spatial system is the reprojection of panoramas (curvilinear perspective) to obtain linear perspective views, e.g. [Chen 1995; Tolba et al. 2001]. The method by Zorin and Barr corrects for perspective distortion using a $2D \rightarrow 2D$ technique to preserve either alignment or sphere symmetry [Zorin and Barr 1995]. Seitz and Dyer present another $2D \rightarrow 2D$ spatial technique that is particularly interesting because it occurs purely in the secondary space, but respects an hypothetical 3D geometry [Seitz and Dyer 1996].

Perspective has been a subject of intense debates in the visual arts [Hagen 1986; Pirenne 1970; Panofsky 1927; Kubovy 1986; Kemp 1990; Elkins 1994]. The difference between $3D \rightarrow 2D$ issues – basically visibility – and $2D \rightarrow 2D$ issues is often overlooked and results in misunderstandings between parties. Some authors argue that linear perspective is "natural" and respects human vision because it faithfully respects visibility, that is, it is the projection from a given point [Gombrich 1982]. However, they miss the fact that any 2D warping of a linear-perspective image also respects visibility. Expressed in primary geometry, it means that the projection with respect to a point can be performed on any manifold, e.g. plane, sphere or cylinder. We have seen that it is more fruitful to state the debate in terms of invariant or property preservation. In this case, there is no perfect "natural" solution since we cannot preserve both linearity and the symmetry of spheres.

Interaction techniques going from 2D inputs to 3D can be used to control the camera [Gleicher and Witkin 1992; Eades et al. 1997]. Injecting more secondary geometry controls in Agrawala et al.'s multiperspective technique [Agrawala et al. 2000], as well as in other non-linear perspective work [Löffelmann and Gröller 1996; Glaeser and Gröller 1999; Bourgoin et al. 1995; Levene 1998; Rademacher and Bishop 1998; Glassner 2000] would greatly improve their usability.

Another class of $2D \rightarrow 3D$ interactions facilitates the modeling phase. Approaches have been proposed to sketch 3D objects using 2D strokes [Zeleznik et al. 1996; Igarashi et al. 1999; Cohen et al. 2000], or to build 3D models from photographs, e.g. [Faugeras et al. 1995; Debevec et al. 1996]. Similar $2D \rightarrow 2.5D$ techniques also exist [Horry et al. 1997; Oh et al. 2001; Chu and Tai 2001; Zhang et al. 2001]. Other hybrid interactive spatial systems allow a user to draw in 2D but modify the view or move objects in pseudo-3D [Tolba et al. 2001; Bourguignon et al. 2001]. And as discussed above, view-dependent models allow 2D spatial objectives to control 3D models [Rademacher 1999; Martin et al. 2000]. Gooch et al. [Gooch et al. 2001] use optimization to choose the 3D camera parameters with a 2D goal: good composition.

Optimization has also been used to solve 2D spatial aspects. Agrawala et al. compute route maps using 2D optimization loosely respecting the 3D geometry according to cognitive findings [Agrawala and Stolte 2001]. It is particularly interesting to note that their approach is based on shape properties (length, angle) and not directly on spatial coordinates. Graph drawing is also a pure 2D optimization problem [Battista et al. 1999], and recently, Escherization optimizes a shape to tile the plane [Kaplan and Salesin 2000]. An extension to optimizing over the 3D domain would be quite exciting.

Finally, we discuss the multiperspective cell panorama technique by Wood et al. [Wood et al. 1997]. This technique is very interesting because from the point of view of the beholder, it looks like linear perspective, while for the artist, it requires a highly non-linear spatial system. The authors noted that Disney's artists were able to produce more convincing multiperspective panorama than the computer-assisted method. We hypothesize that this is because their spatial system operates on the primary geometry. In contrast artists reason only in terms of secondary geometry, which alleviates them from the constraints of primary geometry, and allows them to think directly in terms of goals and property mapping. An exciting subject of future work would start from the automatic primary-geometry solution, and use relaxation to optimize the multiperspective, in order to minimize distortions in terms of secondary geometry.

6.2 Primitive

Recall that there are four different kinds of picture primitives: continuous points, discrete points, lines and regions. 1D primitives probably yield the richest variety of denotation systems. Lines can denote a large class of scene primitives. They can be classified into view-independent and view-dependent primitives. View independent primitives include very thin objects (such as strings), elongated objects (such as legs), edges of objects, reflectance discontinuity (such as the limit of a patch on a cow), shadow boundaries, or transparency edges.

View-dependent 1D primitives consist in occluding contour, and a special case of occluding contour, the external silhouette of objects, and the limits of specular highlights. The latter is a case where the denotation system (line drawing of the highlight) interacts with the attribute system (shininess of the material). Shadows raise similar issues, since they can depend on the primitive, attribute and spatial systems.

We distinguish three approaches to silhouette extraction $3D \rightarrow 2D$ [Elber and Cohen 1990; Gooch 1998; Markosian et al. 1997; Sander et al. 2000; Zorin and Hertzmann 2000], $2.5D \rightarrow 2D$ [Saito and Takahashi 1990; Raskar and Cohen 1999; Curtis 1998; Buchanan and Sousa 2000] or $2D \rightarrow 2D$ [Canny 1986; Pearson et al. 1990].

An important issue of future work is the design of edge selection algorithms. Artists have the ability to draw only the relevant edges to depict an object. This can be addressed by devising selection rules, or interactive selection tools.

Elder et al. propose to adapt image editing to work in what they call the *contour domain* [Elder and Goldberg 2001]. An image is represented as a set of edges and continuous smooth regions. Their technique basically transforms the image from a continuous-point representation, to a 1D-line representation, which facilitates some editing operations for the user.

6.3 Attribute

The set of possible picture attributes depend on the primitives, marks, and on the context. Attributes include color, tone, transparency, texture, thickness, wiggling (for lines primitives, e.g. [Finkelstein and Salesin 1994]), or orientation. Color can be expressed in different color spaces, such as RGB, HSV, or other dimensions such as cool-to-warm can be used, e.g. [Gooch et al. 1998].

The classical $3D \rightarrow 2D$ attribute system is lighting and shading, where illuminance and BRDF are combined to compute a visible color. The non-photorealistic technique by Gooch et al. is particularly interesting because it uses the intrinsic color of the objects with a relative extrinsic lighting mapped on the cool-to-warm dimension [Gooch et al. 1998]. Shading methods have been introduced for line primitives as well, e.g. [Saito et al. 1989; Tanaka et al. 1991; Gooch et al. 1999]. Atmospheric perspective is a very interesting pictorial technique where the distance can be mapped to different attributes. It can affect saturation, make distant objects more bluish, decrease sharpness, etc. In fact, the most important aspect of aerial perspective is to *group* parts of the scene at a similar distance by assigning them a common property.

$2D \rightarrow 2D$ attribute techniques include standard image controls such as contrast/brightness or color modification, e.g. [Reinhard et al. 2001], or dodging and burning [Adams 1995]. Tone mapping is also a $2D \rightarrow 2D$ attribute technique, that specifically copes with the limitations of the medium, e.g. [Tumblin and Turk 1999]. Note that the same pictorial effect – decreasing the contrast – can also be obtained in a $3D \rightarrow 3D$ way, using appropriate lighting [Millerson 1991].

Hybrid approaches include works such as shading in two dimensions [Williams 1991], the commercial product ZBrush [ZBrush n. d.], and the comprehensive rendering of shape [Saito and Takahashi 1990].

The case of graftals is a rather complex attribute system, since it heavily interacts with denotation and marks [Kowalski et al. 1999; Markosian et al. 2000]. Complex materials such as fur or plants are rendered using procedurally-generated strokes. Their work moreover permits high-level as well as spatially-varying graftal style specification, and can be used in conjunction with lighting and shading.

As mentioned before, attributes can be modified by altering the 3D scene. Examples of $3D \rightarrow 3D$ attribute techniques include photography lighting [Millerson 1991] or make up [Aucoin 1999].

$2D \rightarrow 3D$ interaction techniques permit the deduction of lighting from desired color [Schoeneman et al. 1993], or from sketching highlights and shadow [Poulin and Fournier 1992; Poulin et al. 1997]. In the recent lit-sphere method, an artist paints an example sphere, which is remapped to an environment map for 3D rendering [Sloan et al. 2001]. 3D painting [Hanrahan and Haeberli 1990; Agrawala et al. 1995; 3D n. d.] allows a user to edit the texture maps of the object-space model using a 2D interface. Image-based editing systems offer similar possibilities [Seitz and Kutulakos 1998;

Oh et al. 2001] and can be classified as $2D \rightarrow 2.5D$.

$2D \rightarrow 3D$ interaction techniques have been developed to inject some 3D attribute notions into an otherwise purely two-dimensional depiction context. This includes texture mapping for cell animation[Corrêa et al. 1998], and shadows for cell animation [Petrovic et al. 2000] or for architectural sketching [Tolba et al. 2001].

We finish this overview of attribute systems with a discussion of orientation, e.g. [Haeberli 1990; Salisbury et al. 1997; Zorin and Hertzmann 2000; Hausner 2001]. While orientation is used to drive the mark system, it is important to consider it as an attribute, since orientation is a general issue, common to a variety of mark styles. Separating the orientation issue from the particular marks is crucial to build generic modules. Most mark system using orientation attributes can be used to display any 2D vector field, e.g. pen and ink strokes [Salisbury et al. 1997], streamlines [Turk and Banks 1996], or LIC [Cabral and Leedom 1993]. $3D \rightarrow 3D$ orientation computation have been proposed using iso-parametric curves [Elber 1995] or principal curvatures [Zorin and Hertzmann 2000]. $3D \rightarrow 2D$ [Saito and Takahashi 1990]. Similarly, $2.5D \rightarrow 2D$ exist, e.g. [Rössl and Kobbelt 2000]. The $2D \rightarrow 2D$ category offers both automatic [Haeberli 1990; Hausner 2001] and user-controlled [Haeberli 1990; Ostromoukhov 1999; Salisbury et al. 1997] approaches.

6.4 Mark

The mark system is the last representational aspect. It deals with the physical medium of the picture. The mark system can be trivial, in particular for realistic graphics where the mark is simply a pixel. On the other end of the spectrum, photomosaics use pictures as marks, and some very advanced physically based simulation have been developed for various media, e.g. [Curtis et al. 1997; Sousa and Buchanan 1999; Takagi et al. 1999; Baxter et al. 2001; Geigel and Musgrave 1997].

The mark system is a special case, because it does not really involve a 3D to 2D mapping, which has been dealt with by the previous systems. The mark system is thus mainly a 2D problem. However, we will see that in some cases, especially for animation, 3D aspects are important.

The example of halftoning [Ulichney 1987] is paradigmatic of mark system because the input, output and specification are clearly defined. Halftoning takes as input a grey-scale (or color) image and translates it into a binary image that provides the viewer with a faithful tonal impression. Central to halftoning is the linearity of the reproduction curve, which more generally means that the output should be predictable from the input. Halftoning has also been extended to richer patterns [Ostromoukhov and Hersch 1995; Veryovka and Buchanan 1999].

Optimization has been used for mark systems, either to optimize their location [Haeberli 1990; Hausner 2001; Deussen et al. 2000], or tonal fidelity [Ostromoukhov and Hersch 1995; Ostromoukhov 1999].

One of the challenges for mark systems is raised by NPR animation, where mark coherence is paramount. Approaches based on the 3D geometry [Meier 1996; Curtis 1998; Praun et al. 2001] or motion flow have been proposed [Litwinowicz 1997; Hertzmann and Perlin 2000]. So far, the most successful approaches have used a combination of 3D-based coherence with picture-based criteria [Meier 1996; Curtis 1998; Praun et al. 2001]. Again, the interplay between 3D and 2D is at the heart of the richness and complexity of depiction.

7 Invitations

We hope that this article will stimulate discussion and future work in computer depiction. We insist that the framework proposed in this paper is not intended as a rigid set of boxes for the sake of classification. We hope that it provides a vocabulary and raises issues. We are also well aware that, due to the complexity of depiction, different classification can be proposed along dimensions similar or orthogonal to the ones discussed in this paper.

The extension to animation is far from straightforward. We have seen that 2D pictures are not a simple section of the optical flow. Similarly, they are not a simple cross section of space-time.

We want to study existing NPR software in this framework, and describe their *image generation* and *interaction* work flows. This should highlight similarities, potential cross-integrability, as well as original designs that can be applied to different problems. The design of a versatile NPR system implementing the diversity of depiction styles and interactions is a challenging task, raising both software design and depiction issues.

Higher-level issues need further discussion. This include notions of abstraction, precision, selection, as well as aesthetic issues such as composition, balance or color harmony.

The classification into four depiction systems provides a structure for a coarse-grain definition of style. The refinement of this definition raises exciting issues in stylistics, and could allow us to parameterize, capture and reuse style.

The availability of this new variety of styles raises the important question of the choice of an appropriate style, especially when clarity is paramount. These are cognitive psychology questions, but we hope that computer depiction can provide both an experimental testbed, and theoretical hints.

Acknowledgment

The careful comments of the reviewers greatly helped to improve the paper. Many thanks to Julie Dorsey, Victor Ostromoukhov, Pat Hanrahan, Maneesh Agrawala, Fabrice Neyret, Joëlle Thollot, Byong Mok Oh, and the students of the 4.209 course *The Art and Science of Depiction*. This research was partially funded by an NSF CISE Research Infrastructure Award (EIA-9802220).

References

3D, D. P. http://www.deeppaint3d.com.

ADAMS, A. 1995. *The Camera+The Negative+The Print*. Little Brown and Co.

AGRAWALA, M., AND STOLTE, C. 2001. Rendering effective route maps: Improving usability through generalization. In *Computer Graphics (Proc. SIGGRAPH)*.

AGRAWALA, M., BEERS, A., AND LEVOY, M. 1995. 3d painting on scanned surfaces. *ACM Symp. on Interactive 3D Graphics*.

AGRAWALA, M., ZORIN, D., AND MUNZNER, T. 2000. Artistic multiprojection rendering. In *Eurographics Workshop on Rendering*.

APODACA, A. 1999. Photosurrealism. In *Advanced Renderman : Creating CGI for Motion Pictures*, A. Apodaca and L. Gritz, Eds. Morgan Kaufmann.

AUCOIN, K. 1999. *Making Faces*. Little Brown and Co.

BARBOUR, AND MEYER. 1992. Visual cues and pictorial limitations for computer generated photorealistic images. *the Visual Computer 9*.

BARZEL, R. 1997. Lighting controls for computer cinematography. *J. Graphics Tools 2*, 1.

BATTISTA, G. D., EADES, P., TAMASSIA, R., AND TOLLIS, I. 1999. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall.

BAXTER, B., SCHEIB, V., LIN, M., AND MANOCHA, D. 2001. Dab: Interactive haptic painting with 3d virtual brushes. *Proc. SIGGRAPH*.

BOOKER, P. J. 1963. *A History of Engineering Drawing*. Chatto and Windus.

BOURGOIN, V., FARENC, N., AND ROELENS, M. 1995. Creating special effects by ray-tracing with non classical perspectives. Tech. Rep. 1995-13, École Nationale des Mines de St Etienne.

BOURGUIGNON, D., CANI, M., AND DRETTAKIS, G. 2001. Drawing for illustration and annotation in 3d. *Computer Graphics Forum 20*, 3.

BUCHANAN, J., AND SOUSA, M. 2000. The edge buffer: A data structure for easy silhouette rendering. In *Non-Photorealistic Animation and Rendering*.

CABRAL, B., AND LEEDOM, L. C. 1993. Imaging vector fields using line integral convolution. In *Proc. SIGGRAPH*.

CANNY, J. 1986. A computational approach to edge detection. *IEEE Trans PAMI 8*, 6, 679–698.

CARLBOM, I., AND PACIOREK, J. 1978. Planar geometric projections and viewing transformations. *Comp. Surveys 10*, 465—502.

CHEN, E. 1995. Quicktime VR - an image-based approach to virtual environment navigation. *Proc. of SIGGRAPH*.

CHU, S.-H., AND TAI, C.-L. 2001. Animating chinese landscape paintings and panorama using multi-perspective modeling. In *Computer Graphics International*.

CLOWES, M. 1971. On seeing things. *Artificial Intelligence 2*, 1, 79–116.

COHEN, J., HUGHES, J., AND ZELEZNIK, R. 2000. Harold: A world made of drawings. In *Non-Photorealistic Animation and Rendering*.

CORRÊA, W. T., JENSEN, R., THAYER, C., AND FINKELSTEIN, A. 1998. Texture mapping for cel animation. *Proc. SIGGRAPH*.

CURTIS, C., ANDERSON, S., SEIMS, J., FLEISCHER, K., AND SALESIN, D. 1997. Computer-generated watercolor. *Proc. SIGGRAPH*.

CURTIS, C. 1998. Loose and Sketchy Animation. In *SIGGRAPH: Conference Abstracts and Applications*.

DEBEVEC, P., TAYLOR, C., AND MALIK, J. 1996. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In *Proc. of SIGGRAPH 96*.

DEUSSEN, O., HILLER, S., VAN OVERVELD, C., AND STROTHOTTE, T. 2000. Floating points: A method for computing stipple drawings. *Computer Graphics Forum 19*, 3.

DUBERY, F., AND WILLATS, J. 1983. *Perspective and Other Drawing Systems*. Van Nostrand.

DURAND, F., OSTROMOUKHOV, V., MILLER, M., DURANLEAU, F., AND DORSEY, J. 2001. Decoupling strokes and high-level attributes for interactive traditional drawing. In *Eurographics Workshop on Rendering*.

EADES, P., HOULE, M. E., AND WEBBER, R. 1997. Finding the best viewpoints for three-dimensional graph drawings. *Lecture Notes in Computer Science 1353*, 87–??

ELBER, G., AND COHEN, E. 1990. Hidden Curve Removal for Free Form Surfaces. In *Computer Graphics (Proc. SIGGRAPH)*.

ELBER, G. 1995. Line art rendering via a coverage of isoparametric curves. *IEEE Trans. on Visualization and Computer Graphics 1*, 3 (Sept.), 231–239.

ELDER, J., AND GOLDBERG, R. 2001. Image editing in the contour domain. *IEEE Trans. on Pattern Analysis and Machine Intelligence 23*, 3.

ELKINS, J. 1994. *The Poetics of Perspective*. Cornell U. Pr.

FAIRCHILD. 1998. *Color Appearance Models*. Addison-Wesley.

FAUGERAS, O., LAVEAU, S., ROBERT, L., CSURKA, G., AND ZELLER, C. 1995. 3-d reconstruction of urban scenes from sequences of images. In *Automatic Extraction of Man-Made Objects from Aerial and Space Images*, A. Gruen, O. Kuebler, and P. Agouris, Eds. Birkhauser.

FERWERDA, J., 1999. Varieties of realism. talk, Cornell Workshop on Rendering, Perception, and Measurement, http://www.graphics.cornell.edu/workshop/.

FINKELSTEIN, A., AND SALESIN, D. 1994. Multiresolution curves. In *Proc. SIGGRAPH*.

GEIGEL, J., AND MUSGRAVE, K. 1997. A model for simulating the photographic development process on digital images. *Proc. SIGGRAPH*.

GERSHBEIN, R., AND HANRAHAN, P. 2000. A fast relighting engine for interactive cinematic lighting design. In *Proc. SIGGRAPH*, Computer Graphics.

GLAESER, G., AND GRÖLLER, E. 1999. Fast generation of curved perspective for ultra-wide-angle lenses in vr applications. *Visual Computer 15*, 365–376.

GLASSNER, A., FISHKIN, K., MARIMONT, D., AND STONE, M. 1995. Device-directed rendering. *ACM Trans. on Graphics 14*, 1 (January), 58–76.

GLASSNER, A. 2000. Cubism and cameras: Free-form optics for computer graphics. Tech. Rep. MSR-TR-2000-05, Microsoft Research.

GLEICHER, M., AND WITKIN, A. 1992. Through-the-lens camera control. *Computer Graphics (Proc. SIGGRAPH)*.

GOMBRICH. 1956. *Art and Illusion*. Princeton Press.

GOMBRICH, E. 1982. *The image and the eye : further studies in the psychology of pictorial representation*. Cornell U. Pr.

GOMES, J., DARSA, L., COSTA, B., AND VELHO, L. 1998. *Warping And Morphing Of Graphical Objects*. Morgan Kaufman.

GOOCH, AND GOOCH. 2001. *Non-Photorealistic Rendering*. AK-Peters.

GOOCH, A., GOOCH, B., SHIRLEY, P., AND COHEN, E. 1998. A non-photorealistic lighting model for automatic technical illustration. *Proc. SIGGRAPH*.

GOOCH, B., SLOAN, P., GOOCH, A., SHIRLEY, P., AND RIESENFELD, R. 1999. Interactive Technical Illustration. *ACM Symposium on Interactive 3D Graphics*.

GOOCH, B., REINHARD, E., MOULDING, C., AND SHIRLEY, P. 2001. Artistic composition for image creation. In *Eurographics Workshop on Rendering*.

GOOCH, A. 1998. *Interactive Non-photorealistic Technical Illustration*. Master's thesis, U. of Utah.

GREEN, S., SALESIN, D., SCHOFIELD, S., HERTZMANN, A., LITWINOWICZ, P., GOOCH, A., CURTIS, C., AND GOOCH., B. 1999. Non-Photorealistic Rendering. *SIGGRAPH Non-Photorealistic Rendering Course Notes*.

GREEN, S. 1999. Beyond photorealism. In *Eurographics Workshop on Rendering*.

GRÖLLER, E. 1994. Nonlinear ray tracing: visualizing strange worlds. *Visual Computer 11*, 5.

GUENTER, B., KNOBLOCK, T., AND RUF, E. 1995. Specializing shaders. *Proc. SIGGRAPH*.

GUZMAN, A. 1971. Analysis of curved line drawings using context and global information. In *Machine Intelligence V6*, B. M. D. Mitchie, Ed. Edinburgh U. Pr., 325–375.

HAEBERLI, P. 1990. Paint by numbers: Abstract image representations. *Proce. SIGGRAPH*.

HAGEN, M. 1986. *Varieties of Realism: Geometries of Representational Art*. Cambridge U. Pr.

HANRAHAN, P., AND HAEBERLI, P. 1990. Direct WYSIWYG painting and texturing on 3D shapes. *Computer Graphics (Proc. SIGGRAPH)*.

HAUSNER, A. 2001. Simulating decorative mosaics. *Proc. SIGGRAPH*.

HE, L., COHEN, M., AND SALESIN, D. 1996. The virtual cinematographer: A paradigm for automatic real-time camera control and directing. *Computer Graphics (Proc. SIGGRAPH)*.

HERTZMANN, A., AND PERLIN, K. 2000. Painterly rendering for video and interaction. In *Non-Photorealistic Animation and Rendering*.

HERTZMANN, A. 1998. Painterly rendering with curved brush strokes of multiple sizes. *Proc. SIGGRAPH*.

HERTZMANN, A. 2001. Paint by relaxation. In *CGI*.

HOCKNEY, D. 2001. *Secret Knowledge: Rediscovering the Lost Techniques of the Old Masters*. Viking Press.

HODGINS, J., O'BRIEN, J., AND TUMBLIN, J. 1998. Perception of human motion with different geometric models. *IEEE TVCG 4*, 4 (Oct.).

HOGARTH, B. 1981. *Dynamic Light and Shade*. Watson Guptill.

HORRY, Y., ANJYO, K., AND ARAI, K. 1997. Tour into the picture: Using a spidery mesh interface to make animation from a single image. In *Proc. of SIGGRAPH 97*.

HUFFMAN, D. 1971. Impossible objects as nonsense sentences. In *Machine Intelligence 6*, B. Meltzer and D. Michie, Eds. Edinburgh U. Pr., 295–324.

HUNT. 1995. *The reproduction of Color (5th ed.)*. Kings Langley:Fountain Press.

IGARASHI, T., MATSUOKA, S., AND TANAKA, H. 1999. Teddy: A sketching interface for 3d freeform design. *Proc. SIGGRAPH*.

KAJIYA, J. 1986. The rendering equation. In *Computer Graphics (Proc. SIGGRAPH)*.

KAPLAN, C., AND SALESIN, D. 2000. Escherization. *Proc. SIGGRAPH*.

KEMP, M. 1990. *The Science of Art*. Yale University Press.

KOENDERINK, J. 1990. *Solid Shape*. MIT Press.

KOWALSKI, M., MARKOSIAN, L., NORTHRUP, J. D., BOURDEV, L., BARZEL, R., HOLDEN, L., AND HUGHES, J. 1999. Art-based rendering of fur, grass, and trees. *Proc. SIGGRAPH*.

KUBOVY, M. 1986. *The Psychology of Perspective and Renissance Art*. Cambridge University Press.

LAND, E. H. 1977. The retinex theory of color vision. *Scientific American 237*, 6 (Dec.), 108–128.

LANSDOWN, J., AND SCHOFIELD, S. 1995. Expressive Rendering: A Review of Nonphotorealistic Techniques. *IEEE CG & A 15*, 3, 29–37.

LEVENE, J. 1998. *A Framework for Non-Realistic Projections*. Master's thesis, Massachusetts Institute of Technology.

LITWINOWICZ, P. 1991. Inkwell: A 2 -d animation system. *Computer Graphics (Proc. SIGGRAPH)*.

LITWINOWICZ, P. 1997. Processing images and video for an impressionist effect. *Proc. SIGGRAPH*.

LÖFFELMANN, H., AND GRÖLLER, E. 1996. Ray tracing with extended cameras. *J. of Visualization and Comp. Animation 7*, 4, 211–227.

MARKOSIAN, L., KOWALSKI, M., TRYCHIN, S., BOURDEV, L., GOLDSTEIN, D., AND HUGHES, J. 1997. Real-time nonphotorealistic rendering. *Proc. SIGGRAPH*.

MARKOSIAN, L., MEIER, B., KOWALSKI, M., HOLDEN, L., NORTHRUP, J., AND HUGHES, J. 2000. Art-based rendering with continuous levels of detail. In *Non-Photorealistic Animation and Rendering*.

MARKS, J., ANDALMAN, B., BEARDSLEY, P., FREEMAN, W., GIBSON, S., HODGINS, J., KANG, T., MIRTICH, B., PFISTER, H., RUML, W., RYALL, K., SEIMS, J., AND SHIEBER, S. 1997. Design galleries: A general approach to setting parameters for computer graphics and animation. In *Proc. SIGGRAPH 97*.

MARTIN, D., GARCIA, S., AND TORRES, J. C. 2000. Observer dependent deformations in illustration. In *Non-Photorealistic Animation and Rendering*.

MAX, N. L. 1983. Computer graphics distortion for IMAX and OMNIMAX projection. In *Nicograph '83 Proceedings*, 137–159.

MEIER, B. 1996. Painterly rendering for animation. *Proc. SIGGRAPH*.

MILLERSON. 1991. *Lighting for Television and Film, 3rd ed.* Focal Press.

NPAR, 2000. Discussion. First International Symposium on Non Photorealistic Animation and Rendering.

OH, B. M., CHEN, M., DORSEY, J., AND DURAND, F. 2001. Image-based modeling and photo editing. *Proc. SIGGRAPH*.

OSTROMOUKHOV, V., AND HERSCH, R. 1995. Artistic screening. *Proc. SIGGRAPH*.

OSTROMOUKHOV, V. 1999. Digital facial engraving. *Proc. SIGGRAPH*.

PANOFSKY, E. 1927. *Perspective As Symbolic Form (La perspective comme forme symbolique)*. Zone Books.

PARENT, R. 2001. *Computer Animation: Algorithms and Techniques,*. Morgan-Kaufmann.

PEARSON, D., HANNA, E., AND MARTINEZ, K. 1990. Computer generated cartoons. In *Images and Understanding*, Barlow, Blakemore, and Weston-Smith, Eds. Cambrdge U. P.

PELLACINI, F., FERWERDA, J., AND GREENBERG, D. 2000. Toward a psychophysically-based light reflection model for image synthesis. *Proc. SIGGRAPH*.

PETROVIC, L., FUJITO, B., FINKELSTEIN, A., AND L.WILLIAMS. 2000. Shadows for cel animation. *Proc. SIGGRAPH*.

PIRENNE, M. 1970. *Optics, Painting and Photography*. Cambridge U. Pr.

POULIN, P., AND FOURNIER, A. 1992. Lights from highlights and shadows. In *ACM Symposium on Interactive 3D Graphics*.

POULIN, P., RATIB, K., AND JACQUES, M. 1997. Sketching shadows and highlights to position lights. In *Proc. Computer Graphics International*.

PRAUN, E., HOPPE, H., M., AND FINKELSTEIN, A. 2001. Real-time hatching. *Proc. SIGGRAPH*.

RADEMACHER, P., AND BISHOP, G. 1998. Multiple-center-of-projection images. In *Proc. SIGGRAPH*.

RADEMACHER, P. 1999. View-dependent geometry. *Proc. SIGGRAPH*.

RAMACHANDRAN, V., AND HIRSTEIN, W. 1999. The science of art: A neurological theory of aesthetic experience. *Journal of Consciousness Studies 6*, 6/7.

RASKAR, R., AND COHEN, M. 1999. Image Precision Silhouette Edges. In *Proc. 1999 ACM Symp. on Interactive 3D Graphics*.

REINHARD, E., ADHIKHMIN, M., GOOCH, B., AND SHIRLEY, P. 2001. Color transfer between images. *IEEE Computer Graphics and Applications 21*, 4, 34–41.

REYNOLDS, C., 2000. Stylized depiction in computer graphics non-photorealistic, painterly and 'toon rendering. http://www.red3d.com/cwr/npr/.

RÖSSL, C., AND KOBBELT, L., 2000. Line-art rendering of 3d models.

RYAN, T., AND SCHWARTZ, C. 1956. Speed of perception as a function of mode of representation. *Am. J. of Psychology 69*, 60–69.

SAITO, T., AND TAKAHASHI, T. 1990. Comprehensible Rendering of 3-D Shapes. In *Computer Graphics (Proc. SIGGRAPH)*.

SAITO, SHINYA, AND TAKAHASHI. 1989. Highlighting rounded edges. In *Computer Graphics International*.

SALISBURY, M., WONG, M., HUGHES, J., AND SALESIN, D. 1997. Orientable textures for image-based pen-and-ink illustration. *Proc. SIGGRAPH*.

SALISBURY, M. P., ANDERSON, S. E., BARZEL, R., AND SALESIN, D. H. July 1994. Interactive pen-and-ink illustration. *Proceedings of SIGGRAPH 94*, 101–108. ISBN 0-89791-667-0. Held in Orlando, Florida.

SANDER, P., GU, X., GORTLER, S., HOPPE, H., AND SNYDER, J. 2000. Silhouette clipping. *Proc. SIGGRAPH*.

SCHOENEMAN, C., DORSEY, J., SMITS, B., ARVO, J., AND GREENBERG, D. 1993. Painting with light. *Computer Graphics*.

SEITZ, S., AND DYER, C. 1996. View morphing: Synthesizing 3d metamorphoses using image transforms. *Proc. SIGGRAPH*.

SEITZ, S., AND KUTULAKOS, K. 1998. Plenoptic image editing. In *Proc. 5th Int. Conf.on Computer Vision*.

SELIGMANN, D. D., AND FEINER, S. July 1991. Automated generation of intent-based 3d illustrations. *Computer Graphics (Proceedings of SIGGRAPH 91) 25*, 4, 123–132. ISBN 0-201-56291-X. Held in Las Vegas, Nevada.

SHACKED, R., AND LISCHINSKI, D. 2001. Automatic lighting design using a perceptual quality metric. *Computer Graphics Forum 20*, 3, 215–226.

SLOAN, P., MARTIN, W., GOOCH, A., AND GOOCH, B. 2001. The lit sphere: A model for capturing npr shading from art. In *Proc. Graphics Interface*.

SOUSA, M., AND BUCHANAN, J. 1999. Observational model of blenders and erasers in computer-generated pencil rendering. In *Graphics Interface*.

SUTHERLAND, I. E., SPROULL, R. F., AND SCHUMACKER, R. A. 1974. A characterization of ten hidden-surface algorithms. *ACM Computing Surveys 6*, 1 (Mar.), 1–55.

TAKAGI, S., FUJISHIRO, I., AND NAKAJIMA, M. 1999. Volumetric modeling of artistic techniques in colored pencil drawing. In *SIGGRAPH: Conf. abstracts and applications*.

TANAKA, TOKIICHIRO, AND TAKAHASHI. 1991. Precise rendering method for edge highlighting. In *Computer Graphics International*.

THOMAS, F., AND JOHNSTON, O. 1981. *The Illusion of Life : Disney Animation*. Abbeville Pr.

TOLBA, O., DORSEY, J., AND MCMILLAN, L. 2001. A projective drawing system. In *ACM Symp. on Interactive 3D Graphics*.

TUMBLIN, AND TURK. 1999. LCIS: a boundary hierarchy for detail-preserving contrast reduction. In *Proc. Siggraph*.

TUMBLIN, J. 1999. *Three methods of detail-preserving contrast reduction for displayed images*. PhD thesis, College of Computing Georgia Inst. of Technology.

TURK, G., AND BANKS, D. 1996. Image-guided streamline placement. In *Proc. SIGGRAPH*.

ULICHNEY, R. 1987. *Digital Halftoning*. MIT Press.

VERYOVKA, O., AND BUCHANAN, J. 1999. Halftoning with image-based dither screens. *Graphics Interface*.

VON HELMOHOLTZ, H. 1881. On the relation of optics to painting. In *Popular scientific lectures*. Appleton.

WEBSTER. 1983. *Ninth New Collegiate Dictionary*. Merriam Webster.

WILLATS, J. 1990. The draughtsman's contract. In *Images and Understanding*, H. Barlow, C. Blakemore, and M. Weston-Smith, Eds. Cambrdge U. Pr.

WILLATS, J. 1997. *Art and Representation*. Princeton U. Pr.

WILLIAMS, L. 1991. Shading in two dimensions. In *Proc. of Graphics Interface '91*, 143–151.

WINKENBACH, G., AND SALESIN, D. 1994. Computer-generated pen-and-ink illustration. *Proc. SIGGRAPH*.

WOOD, D., FINKELSTEIN, A., HUGHES, J., THAYER, C., AND SALESIN, D. 1997. Multiperspective panoramas for cel animation. *Proc. SIGGRAPH*.

ZBRUSH, P. http://pixologic.com/.

ZEKI, S. 2000. *Inner Vision : An Exploration of Art and the Brain*. Oxford U. Pr.

ZELEZNIK, R., HERNDON, K., AND HUGHES, J. 1996. SKETCH: An interface for sketching 3D scenes. *Proc. SIGGRAPH*.

ZHANG, L., DUGAS-PHOCION, G., SAMSON, J., AND SEITZ, S. 2001. Single view modeling of free-form scenes. In *Proc. of CVPR*.

ZORIN, D., AND BARR, A. 1995. Correction of geometric perceptual distortion in pictures. In *Proc. SIGGRAPH*.

ZORIN, D., AND HERTZMANN, A. 2000. Illustrating smooth surfaces. *Proc. SIGGRAPH*.

Limitations of the Medium and Pictorial Techniques

Frédo Durand
Laboratory for Computer Science, Massachusetts Institute of Technology

Abstract

Pictures have limitations compared to the real optical world. They are flat, they have a limited field of view, they represent the scene from a single viewpoint, they have a limited color gamut and contrast, and they are static. This text discusses these limitations and how image maker have addressed them. Limitations can be eliminated (usually through technological solutions), they can be compensated for (using pictorial techniques), but they also have advantages, they can bring important richness to pictures, and can therefore be accentuated.

Introduction

In this essay, we discuss the fundamental limitations that make pictures different from the real optical flow of a scene. Consider the example of a photograph. Although it is a recording of the light outgoing from the real world, the medium has limitations compared to the real stimulus. First, the photo is flat, and many depth cues are absent, which limits our three-dimensional impression. Second, the field of view is limited to the rectangular frame of the photograph. The contrast is also limited, usually to two or three orders of magnitude: Very bright parts of the image saturate the chemical recording, resulting in uniform white areas in the photo. The darkest black in a print is typically only 100 times darker than the brightest white, while real scene can exhibit a contrast of 1 to 100,000. The color gamut can also be limited, some colors can be hard to reproduce [Hunt 1995]. And of course, the photo is static while the real world is dynamic.

These limitations are a fundamental aspect of the art and craft of depiction. They set some of the differences between the picture and the depicted scene, and they set the limits of the possibilities available to the artist. Their study has been pioneered by the great 19th century scientist Von Helmholtz [von Helmholtz 1925], and have been recently discussed in a computer graphics context by Barbour and Meyer [1992].

In many cases, an art form could not exist in the absence of these limitations. Black-and-white photography is an obvious example, but most styles also rely on the limited field of view of pictures to organize composition. The *frame* of the picture can be taken with its two meanings: the outer border and the essential structure. It is important to realize that despite the negative term “limitation”, these characteristics of the medium are one of the great richness of pictures.

In particular, limitations usually restrict the range of the features of the picture in a limited space, a limited gamut, or a limited time. This in turn permits the relation of various elements of the picture. For example, once projected onto an image, a character and its background are closer than in the three-dimensional space, because the 3D distance has been obviated by the flatness of the medium. Similarly, the limited contrast of pictures reduces the difference between very bright parts and darker parts of the scene. This can install, for example, a relation between the sun and other features, which would not be possible in the real scene because the sun is too bright.

Different medium have different limitations. For example, oil painting and watercolor permit different color gamuts, and in black-and-white photography, the color gamut is extremely limited! In this text, we are not interested in the different medium and their precise limitations, but in general categories of limitations and in general strategies to deal with them.

There are three classes of strategies to deal with a limitation:

Elimination: One can extend the pictorial medium and reintroduce the missing dimension of the visual experience. This solution is usually technological rather than really pictorial. For example, the use of stereo-pairs eliminates some of the limitations related to flatness, and the invention of color photography eliminated the limitation to black-and-white optical reproduction.

Compensation: One can compensate for a limitation by conveying the missing cue or dimension using a different mode. For example, flatness can be compensated for by accentuating the contrast at the occluding silhouette. In this case, the lack of depth cues such as stereo is compensated for by an increased occlusion cue.

Accentuation: Finally, a limitation can be desirable for aesthetic or clarity reasons. In this case, the image maker can deliberately choose to accentuate the limitation. For example, black-and-white photography is often considered more artistic despite its missing color dimension.

We will discuss the three strategies, but we are mostly interested in compensation techniques. Elimination usually involves purely technological solutions, and accentuation can in general be obtained by taking the opposite of the compensation techniques.

Pictorial techniques that compensate for a limitation can still be useful if the dimension is present. They can be used to enhance an effect and make the picture more compelling. As discussed elsewhere in these course notes [Gooch 2002; Ramachandran and Hirstein 1999; Zeki 2000], some authors believe that the impact of the picture comes from a "peak-shift" effect, where exaggerated pictures are more effective than the original stimulus.

We do not argue that the pictorial features or techniques that we discuss below have the sole goal of compensating for a limitation. For example, central perspective (linear perspective with a vanishing point at the center of the image) is used to compensate for the flatness of the picture and to create a depth impression. But the converging lines also focus the gaze of the viewer towards the center of the image (Fig. 1). This is the richness and complexity of depiction: Any pictorial choice derives from multiple considerations and has multiple effects.

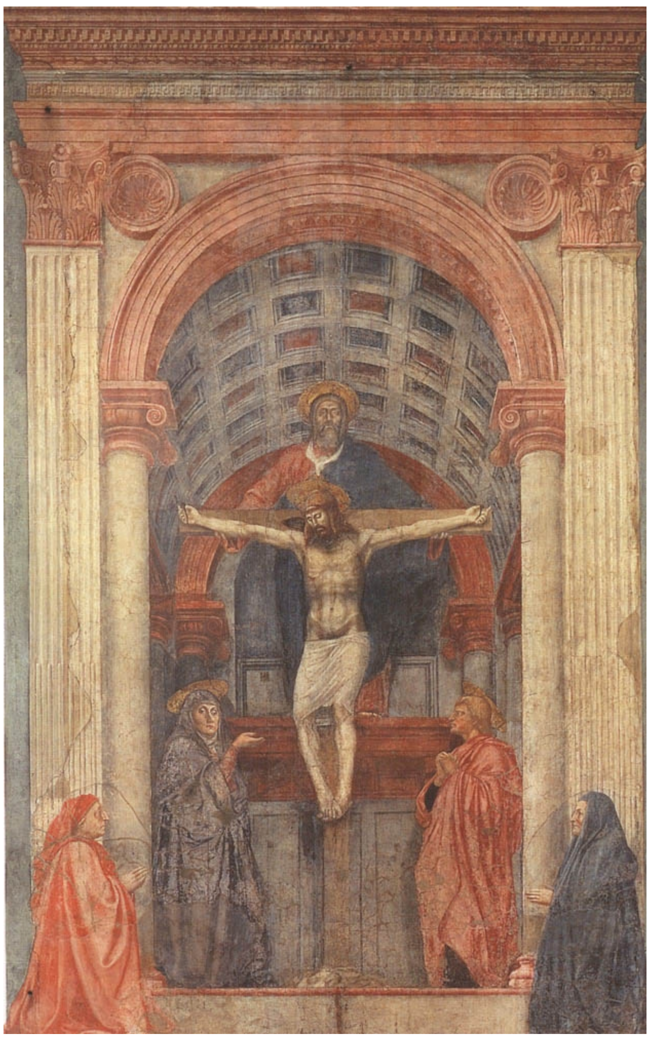


Figure 1: Masaccio *Trinity*, 1427-28

In what follows, we discuss a variety of limitations that restrict the range of the medium in space, time, and light.

- The picture is flat
- The picture has a single viewpoint
- The picture has a limited field of view
- The picture is static
- The picture has limited color gamut and contrast

As discussed above, these limitations do not apply equally to all media; for example, cinema is less limited in the temporal dimension.

1 The picture is flat

The most obvious limitation of pictures is that they are flat, that the third dimension is removed. Before discussing how this issue has been addressed by image maker, we need to outline the perceptual mechanisms that allow us to perceive our three-dimensional environment.

1.1 Depth cues

The human visual system is somehow able to extract depth information from the flat stimulus received by our two eyes. Vision scientists have extensively studied this amazing capacity, know as depth perception. The classic depth cue example is due to stereovision, but as we will see, its importance is often overrated, and it is only one among several types of depth cues. We follow the discussion by Palmer [1999]. Depth cues can be classified according to four characteristics:

Ocular vs. optical Some depth cues come from the optical stimulus (*optical*), while others depend on the physical status of the visual system (*occular*). The two depth cues belonging to the occular category are accommodation and convergence. Accomodation corresponds to the distance of focus of the ocular system. Convergence is related to stereovision. When our two eyes focus on the same point in 3D space, the angle between the optical axis of the two eyes varies: larger when we squint because the point is close, and smaller when the point is distant, as shown in Fig. 3. Our visual system has access to this ocular information from the state of the small muscles governing the focus and direction of each eye. Occular information is usually not available with pictures. On the other end, optical information is extracted from the visual stimulus itself.

Binocular vs. monocular Monocular depth cues can be obtained from a single eye, while binocular depth cues require two eyes. There are two sorts of binocular cues: convergence as discussed above, and binocular disparity. Binocular disparity is extracted from the fusion of the information from our two eyes. It relies on the parallax between the two different views.

Depth cue	ocular/ optical	binocular/ monocular	static/ dynamic	relative/ absolute	quantitative/ qualitative
Accommodation	ocular	monocular	static	absolute	quantitative
Convergence	ocular	binocular	static	absolute	quantitative
Binocular disparity	optical	binocular	static	relative	quantitative
Motion parallax	optical	monocular	dynamic	relative	quantitative
Texture accretion/deletion	optical	monocular	dynamic	relative	qualitative
Convergence of parallels	optical	monocular	static	relative	quantitative
Position relative to the horizon	optical	monocular	static	relative	quantitative
Relative size	optical	monocular	static	relative	quantitative
Familiar size	optical	monocular	static	absolute	quantitative
Texture gradient	optical	monocular	static	relative	quantitative
Edge interpretation (occlusion)	optical	monocular	static	relative	qualitative
Shading and shadows	optical	monocular	static	relative	qualitative
Aerial perspective	optical	monocular	static	relative	qualitative

Figure 2: Classification of depth cues. Adapted from [Palmer 1999].

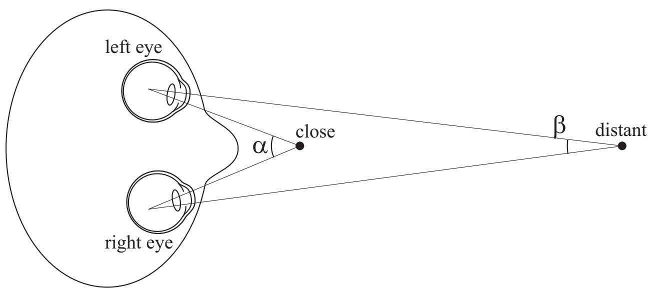


Figure 3: Convergence depth cue. The angle between the direction of the two eyes is larger when the object is close. After [Palmer 1999].

Static vs. dynamic The majority of depth cues are available to the static observer, but additional information can be extracted by moving around the scene. Namely, parallax information is a powerful cue, and the accretion or decretion (appearance/disappearance) of visible texture details as we move closer or farther from an object also provides distance information.

Relative vs. absolute Depth information can be absolute (direct estimation of the distance to the eye) or relative (estimation of the relative distance between parts of the scene). For example, if a series of objects such as electric poles are foreshortened, we can deduce their relative depth. Moreover, if we approximately know the size of a pole, we can deduce their absolute distance. Since familiar size is a powerful cue, it is often advised to add a familiar object or a person in a photograph to provide an absolute scale.

Quantitative vs. qualitative Some cues provide quantitative information (i.e. that could be expressed in meters), while other cues only provide qualitative information, such as this object is closer than this object, or this object is in in front of this object.

1.1.1 Pictorial depth cues

The various depth cues are summarized in Fig. 1. From this classification, we can extract *pictorial cues*, that is, depth cues that can be conveyed by simple pictures. This excludes ocular cues (status of the ocular system), binocular cues (use of two eyes), and dynamic cues (obtained through the motion of the observer). The cues that can be rendered in simple pictures are:

Occlusion When an object partially hides another object, it shows which one is closer than the other. Occlusion information is provided at occluding edges, that is, at the limit of occlusion.

Size The relative size in the picture of similar objects such as a series of telephone poles indicates their relative depth. If the object is familiar, we can deduce a more absolute estimate of their distance.

Position relative to the horizon Because objects usually rest on a flat ground, their position relative to the horizon (i.e. vertical dimension in the picture) is a good cue of their distance. Objects higher relative to the horizon are usually farther.

Convergence of parallel lines Depth can be deduced from the convergence of parallel lines such as railroad tracks. More distant parts of the lines are closer, while closer parts are more wide apart.

Shading and shadows When an object has a uniform color, the variation of intensity are due to its shading, that is, on how the light is reflected depending on the surface orientation. Shading can be used to determine the shape of the object, and therefore the relative distance of its parts (shape from shading). Cast shadows provide additional information, and can, for example, indicate whether an object touches the ground or not.

Texture gradient If an object has a uniform texture, its features will be foreshortened.

Aerial perspective Aerial, or atmospheric perspective describes the effect of the atmosphere on the appearance of distant objects [Minnaert 1993]. Distant object appear more blue, less saturated, less contrasted, and more blurry.

1.2 Elimination of flatness

Many techniques have been developed to extend pictures and provide a full three-dimensional impression. We describe three approaches: actually adding the third dimension, stereo pairs, and motion picture.

3D medium Obviously, the best way to reproduce the third dimension is to use a three-dimensional medium such as sculpture. Holography is also a purely three-dimensional medium. Intermediate solutions such as pop-up books or dioramas make use of a mix of three-dimensional elements and a flat backdrop. This solution is also used for theater stages. Indeed, most non-pictorial depth cues such as stereo and accommodation are not efficient for distant ranges (beyond a few meters), which can make hybrid solutions very convincing.

Stereo Stereo pairs have been popular for over a century. Stereo cameras allow striking recordings. Stereograms and random-dot stereograms are pairs of image that only represent a texture, but when we cross our eyes and fuse the two pictures, a compelling three-dimensional shape appears. Movies and computer displays have also been enhanced through the use of stereo glasses. The principle is to present a different image to each eye, either using color separation (blue/red glasses), polarization, or shutter glasses (which use liquid crystal to occlude one eye while the image destined to the other eye is displayed). The main difference between stereo images and real 3D media is that the accommodation cue (distance of focus) is not present in stereo pairs.

Motion picture Finally, animated media can re-introduce dynamic depth cues such as motion parallax and texture accretion/decretion. The choice of the viewpoint and the motion of the camera are then crucial to provide a compelling impression of depth. For example, when the camera moves close to the ground, the impression of speed and thus the three-dimensional layout of the scene are more dramatically rendered.

1.3 Compensation for flatness

Flatness can be compensated for by increasing the saliency of the remaining pictorial depth cues, or more simply, by ensuring that the scene contains some of these cues. This is an important concept: the scene can be chosen or altered such that cues are present.

Occluding contours Occlusion is a powerful qualitative depth cue. An occluding edge proves that an object is in front of another. Making sure that the view of the scene contains objects occluding parts of the scene can reinforce the three-dimensional impression. For example, it is often advised to include foreground objects in landscape photos. This not only adds more interest, it also provides additional occlusion cues.

Moreover, by increasing the strength of the occluding edge, one puts more emphasis on the occlusion and increases the three-dimensional impression. This can be obtained through specific lighting [Millerson 1991], and in particular through *rim lighting*, that is, the use of a back and grazing light that illuminates the silhouette of the foreground characters (see Fig. 32). This can also be obtained in painting by simply altering the tones around edges in order to increase their contrast. See for example the discussion by Arnheim [1954] of a painting by Titian (Fig. 4).

Figure 4: Titian, *Noli Me Tangere*, 1511. Note how the tone used to depict the houses is modified artificially to increase the contrast at occultation.

Position relative to the horizon Many art styles use the vertical dimension of the picture to convey distance. More distant objects are placed higher in the picture. This is the case of medieval Western art, and some forms of Oriental art.

Size As discussed above, the size of similar or familiar objects provides strong depth information. It is therefore a classical pictorial technique to place a series of similar objects at various distances and use foreshortening to convey distance. Linear perspective is the most classical tool to obtain the required foreshortening of objects.

The effect of size as a depth cue can be increased by facilitating perceptual grouping (see the part about Gestalt in these course notes [Durand 2002b; Gooch 2002]). For example, if a group of objects have the same shape, the same color, and if they are located next to each other in the picture, the foreshortening will pop-out due to perceptual grouping, providing a strong depth impression.

Similarly, if familiar objects are placed in the scene to provide absolute scale, their saliency can increase the perception of space. Conversely, if a familiar object is less obviously visible, it will take longer for the beholder to determine scale, and the size of the scene will remain a riddle, which can eventually increase the importance of the scale impression once the cue is found in a "aha" revelation.

Converging lines Converging lines are the most classical feature of linear perspective and the depiction of depth. Their effect can be strengthened with a wider field of view. For example, in photography, wide-angle lenses often produce strong perspective, as opposed to telephotos that are known to "flatten" the image. However, wide-angle lenses tend to introduce distortions [Zorin 2002], and a compromise has to be found.

Not surprisingly, converging lines and relative sizes of objects are related, as in the example of telephone poles. The lines joining the poles, whether they are drawn or not, converge because the poles are foreshortened towards the vanishing point. If the lines are not drawn, the strength of perceptual grouping as described by the Gestalt theory, can introduce an additional depth cue (converging illusory lines).

Shading and shadows The term *chiaroscuro* denotes the art of using dark and light tones to convey the three-dimensional shape of objects [Da Vinci 1989]. It depicts the interaction between light and objects, and renders surfaces oriented towards the light in light tones, while surfaces oriented further from the light are rendered with darker tones. Note that often, chiaroscuro is used to model shape more than to render the interplay of light and objects. In particular, many pictures do not use a completely coherent lighting scheme, and chiaroscuro is used independently on each object. This is for example the case in most Renaissance paintings, as opposed to the Baroque tenebrism style, where a very coherent and dramatic lighting is used. In a tenebrism painting, light is the main feature, while in a Renaissance painting, shape is preponderant. Similarly, one of the goals of photography lighting is to reveal shape [Millerson 1991]. The light sources are placed such that the shading of the objects emphasizes their shape through the interplay of light and dark tones. In addition, makeup can change the perceived shape of a face through shading and highlights [Aucoin 1999]. Darker and lighter shades of makeup are used to imply shape variation.

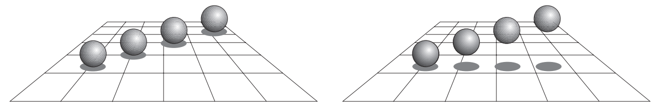


Figure 5: Shadows are important spatial cues.

The depiction of cast shadows can give additional spatial layout information (Fig. 5). Shadows can attach an object to the ground. The absence of good shadows in computer graphics images is known to raise problems of "floating characters" [Blinn 1988]. However, the importance of shadows as a proximity or depth cue is still a subject of active discussions [Wanger et al. 1992; Hu et al. 2000]

Figure 6: Camille Pissaro, *Avenue de l'Opéra, Place du Théâtre Français, Mysty Weather*, 1898.

Atmospheric perspective Reproducing the effect of the atmosphere on the appearance of distant objects is a classical way to convey depth [Da Vinci 1989; Kemp 1990; Solso 1994]. This includes making objects more bluish, less saturated, less contrasted or more blurry (Fig. 6). These effects can be accentuated to increase the depth impression conveyed by the picture. Moreover, these effects are not limited to the accurate simulation of atmospheric conditions, they can be used in a purely pictorial way at ranges where the effect of the atmosphere is negligible. For example, some artists make objects in the back of a room more blue than close objects, although no such optical effect occurs at such a short distance. The effectiveness of this blue shift might be related to our familiarity with atmospheric perspective at larger distances, but also to chromatic aberration in the ocular system: The blue and red wavelengths do not focus at the exact same distance of the retina, and blue objects require a slightly more distant focus. The artist may also compose the scene such that reddish objects lie in the foreground. Constable often used this technique (Fig. 7).

Figure 7: John Constable *The Opening of Waterloo Bridge*, 1829. Note the red boats in the foreground.

Figure 8: Goya *Self-Portrait*, c. 1890-95.

A classical artistic principle states that dark objects tend to perceptually recede while bright objects seem to move forward. This can be used to convey depth. However, many artists also use bright backgrounds with dark foreground characters. This installs a very dynamic balance, since the occlusion cue (foreground partially hides the background) contradicts this black-recedes/white-moves-forward cue (Fig. 8).

Grouping by distance More generally, a depth organization can be obtained by perceptually grouping objects at similar distances. This is notably the case in the cinema lighting technique known as *planes of light*. The scene is organized into layers by alternate intensities of lighting. This moreover tends to increase the occlusion cues, by causing strong contrasts at occluding edges between layers. In general, grouping that binds objects at similar distance (by similar color, proximity in the image, similar lighting, or similar blurriness) makes the spatial organization of the scene pop-out more effectively.

1.4 Advantages of flatness and accentuation

As discussed in the introduction, flatness is not necessarily a disadvantage. The spatial extent of a real scene can impede its unity. But once projected onto the flat medium, distant parts become much closer and can easily be related. Objects of different sizes can also be put on a more similar basis using foreshortening. Our eyes do not have to constantly refocus for objects at different distances. The flatness of the picture, by bringing everything into a more limited range, can provide a unity that is missing from the depicted scene.

The inverse of compensation techniques can be used to accentuate the flatness of the medium. Occlusion edges can be avoided or softened (Fig. 9), shading can be absent, linear perspective eliminated.

In addition, the picture maker can make the physical material of the marks more apparent. For example, the coarse and textured brush strokes of the Impressionists and Van Gogh, or the use of gold as in the paintings by Klimt and Medieval artists puts more emphasis on the picture as a two-dimensional object than on the three-dimensional depicted scene. The presence of written text on the picture also accentuates its 2D nature.

A pictorial style does not necessarily make a clear cut choice between the three strategies of elimination, compensation or accentuation. Each limitation has many facets and rich consequences. The artist may be interested in some of these aspects and want to avoid others. They may also want to produce a contrast by accentuating both the depth impression and the flatness impression, by using different and opposite pictorial techniques. For example, Impressionists usually depict depth using a free linear perspective or texture gradient, but draw attention to the 2D nature of the painting using salient brush strokes.

Figure 9: Claude Monet *Le bassin aux Nymphas*, 1923. The occluding contours of the bridge are blurred, which reduces the depth impression and insists on the 2D qualities of the picture.

2 The viewpoint is unique

When we explore our environment, we are usually free to move and look at objects from a variety of viewpoints, which provides more comprehensive information. In contrast, an image such as a photograph represents the world from a single viewpoint, preventing further exploration. This issue is related to the perception of depth, since observing the scene from multiple viewpoints provides additional information about the three-dimensional layout.

However, the limitation to a single viewpoint has a crucial advantage: the artist is in complete control of this viewpoint. They can choose the viewpoint that results in the "best" picture, the viewpoint that best convey the message. As we consider developing interactive TV or movies, it is important to remember this point: The movie director and the director of photography are usually much more talented than the audience to choose a viewpoint. The limitation to a single viewpoint might restrict our ability to explore the scene, but it also allows us to look through the eyes of the artists and benefit from a carefully crafted choice.

This raises two exciting questions: Which viewpoint should be chosen?, and how can more information be conveyed despite the single viewpoint limit?

2.1 Elimination

Techniques to eliminate the limitation of a single viewpoint fall into two categories: allowing the beholder to freely guide secene exploration, or presentation of different viewpoints imposed by the picture maker. Moreover, the additional viewpoints can be discrete (afinite set of possible viewpoints) or continuous.

The use of a three-dimensional medium such as sculpture allows the beholder to freely and continuously explore the objects from different points of view. This is basically the natural situation in which we explore our environment. Virtual reality provides similar flexibility. This is one of the big differences between 3D media andstereo photography, where the viewpoint is fixed and imposed.

In some cases, 3D media rely on the limitation to a buset of possible viewpoints. For example, most statues should be seen from a particular range of viewpoints, and artists optimize their statue depending on its location with respect to the viewer. The David by Michelangelo does not look right when seen frontally, but looks terrific from all the viewpoints allowed by its location.

On the other hand, in motion pictures, the director can use multiple viewpoints to display a scene, either continuously using camera motion, or discretely using different cameras and cuts through careful editing [Arijon 1991]. This solution does not give any degrees of freedom to the viewer. But this can guarantee high image quality. Future interactive TV systems might provide degrees of freedom, to the viewer, who would be able to use their remote control to change the viewpoint, either continuously or discretely.

The virtualized reality system demonstrated by the CMU team of Kanade, the bullet-time system used in the movie the Matrix or other image-based rendering techniques can provide a continuous and more flexible range of viewpoints to the director, by using a large number of cameras distributed along a smooth path.

2.2 Viewpoint choice

The main issue when choosing a viewpoint is whether the view of the object is characteristic or unusual. As discussed elsewhere in these notes [Gooch 2002], some viewpoint results in a more characteristic view and simpler recognition of the object. They are called *canonical viewpoint*, and are typically off-centered viewpoints that show most features of the object [Palmer 1999; Ramachandran and Hirstein 1999]. See the work by Gooch et al. [2001] for an automatic computer graphics application

The stability of the view and the absence of accidental features are important issues. A view is called stable if it does not change qualitatively when the viewpoint is changed slightly. This rules out accidental alignments, such as the visual superimposition of two vertices of a wireframe cube (Fig. 10), because such alignments disappear when the viewpoint is moved. Accidental alignments make the understanding of the scene harder, because some contacts in the image do not correspond to real 3D attachments.

However, accidental viewpoints and false attachments can be used to relate, in the picture, features that are unrelated

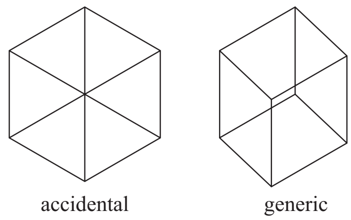


Figure 10: Unstable accidental vs. generic view of a wire-frame cube.

in 3D. This pictorial effect is used to createthe classic false attachment “giant” in Fig. 11.

Figure 11: Classical special effect based on the flatness of the picture, foreshortening, and use of an unstable viewpoint containing accidental attachments. Photo courtesy of Fred Vernier.

2.3 Compensation

The restriction to a single viewpoint can be compensated for by two related strategies: the use of multiple views, and the use of non-linear projection systems, where the image projection is distorted to include information from a set of viewpoints. This distinction is essentially the same as the discrete vs. continuous exploration possibilities discussed above.

Multiple views The most typical use of a discrete set of multiple views is found in engineering drawing or in CAD systems, where three orthographic views and a perspective view of the object are used.

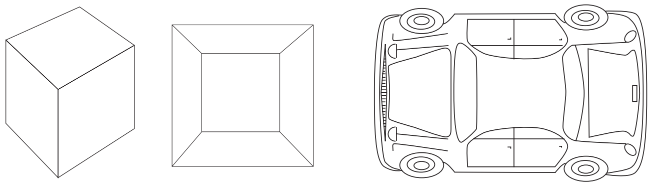
The inclusion of mirrors in an image is a classical way to represent the subject from multiple viewpoints in a single perspectively-correct view. The *Triple Self-Portrait* by Norman Rockwell is a classical example of multiple views, in which he represents himself painting a self-portrait, and we can see him from different viewpoints in the mirror, in the portrait and directly in the scene (Fig. 12).

Figure 12: Norman Rockwell *Triple Self-Portrait*, 1960

Cast shadows can be used to provide a side view of the object. The cast shadow only provides information about the silhouette. It can provide information about objects that are not in the field of view (see the next section about the limitation of the field of view). A jail setting can be suggested with cast shadows from the bars.

“Distorted” projection Projections such as *inverted* or *diverging* perspective permit the depiction of more facets of an object. Parallel lines do not converge towards a vanishing point, instead they diverge, as if the object was viewed from inside (Fig. 13). Other non-linear projections allow similar effects, as discussed by Glassner [2000]. Although they exhibit strong distortions, fish-eye views represent the scene from a single viewpoint and present the same restriction as linear perspective (yet with a wider field of view).

Some art styles such as Egyptian art use non-linear projections where each feature is represented from a canonical viewpoint. For humans, the eyes are frontal, the head is seen from the side, the shoulders are shown frontally, the



(a) (b) (c)

Figure 13: (a) A perspective view of a cube can show at most three faces. (b) An inverted (or diverging) perspective can show five faces. (c) diverging drawing of a car used by a rental company to indicate scratches. All parts of the car hood are visible, which would not be possible with classical perspective (redrawn after a form by Hertz inc.).

legs from the side. This can be described as the use of best local views (Fig. 14).

Figure 14: *Garden of Nebamon*, around 1400 B.C., painting on a wall of the tomb in Thebes. The projection respects individual constraints where the 2D aspect is paramount: the 2D trees are perpendicular to the pond, the 2D pond is a rectangle, the fishes are represented in a profile view where they are more recognizable.

Agrawala et al. have introduced the use of multiple projections in computer graphics [Agrawala et al. 2000] (see the reprint in these course notes). They use different cameras for individual objects to avoid perspective distortions [Zorin 2002] and to optimize the viewpoint locally for each object.

The photomontages by David Hockney lie at the boundary between discrete multiple views and continuous distorted perspective. Hockney combines many Polaroid shots of the scene that represent local gaze fixations. He creates a spatial layout related to their location in the scene, but in a distorted way that depends on their importance or on an exploratory motion from the virtual observer [Hockney 1993]. He also combines the issue of exploration by the motion of the viewer and by gaze movements.

Cubism is the best-known use of multiple perspectives in a single picture. Depending on the period of cubism, these multiple viewpoints can be continuously blended, or multiple gazes can be depicted, such as in analytical cubism.

3 The picture is finite and has a frame

Just as the picture is limited in the third dimension by its flatness, it is also limited in the two other dimensions: It has a limited field of view, traditionally delimited by a rectangular frame. The aspect ratio between height and width has been a subject of intense discussions, including the mathematical mysticism surrounding the golden ratio.

3.1 Consequences

The rectangular border or frame present in most pictures has important consequences. First, it provides a reference for vertical and horizontal lines in the picture. In particular, this means that lines that are slightly off-vertical are very apparent.

Consider the example of the correction of converging verticals in architecture photography or illustration (Fig. 15). When taking a photo of a building from the ground, one usually has to direct the camera upwards in order to see the top of the building. As a result, the camera is not orthogonal to the façade, resulting in a perspective effect: the vertical lines converge towards a vanishing point. This is usually considered distracting, and special architecture lenses can tilt and shift the optical axis to correct this effect and keep verticals in the scene vertical in the picture [Harris 2002]. The problem is largely due to the presence of the vertical frame: By providing a reference, the frame emphasizes the convergence of the verticals in the picture.

Experiments have also shown that the rectangular frame is an important cue to correct the distortion when looking at a picture from an off-center position [Hagen 1980].

The frame can be used to explicitly depict a window frame, which reinforces the concept of the picture as a window to a 3D scene [Kubovy 1986].

The major artistic consequence of the frame is that it permits composition. The balance of the picture is anchored and defined with respect to its boundaries. Features are located carefully to balance them within the frame of the picture. The complement of the main figures with respect to the picture area define the negative spaces, as shown in Fig. 16 [Durand 2002b; Arnheim 1954; Zakia 1997; **?**]. The balance between positive and negative space is crucial for picture layout, and it directly depends on the field of view of picture. By

Figure 15: Photography of an architectural scene, without and with compensation for the convergence of verticals.

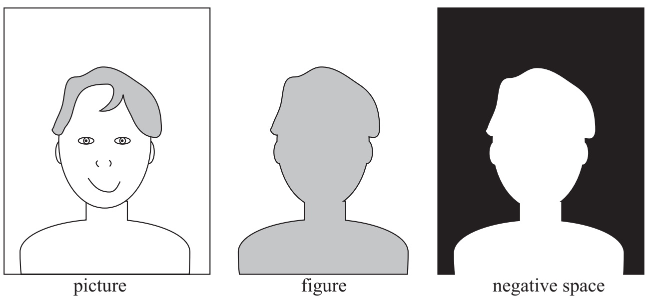


Figure 16: The negative space is delimited by the figure and the frame.

cropping the image, the picture maker selects the part of the scene that is of interest. Just like for the choice of the viewpoint, the framing allows the artist to select the most relevant and aesthetic view of the scene. The picture maker can also choose to hide part of the scene beyond the frame in order to stimulate the imagination of the viewer [Ramachandran and Hirstein 1999; Gooch 2002].

Restricting the picture to a limited spatial extent can also greatly improve the display of information. For example, a route map in which distances have been distorted to fit a tiny sheet of paper is much easier to follow than a more comprehensive and larger map drawn to scale. See the paper by Agrawala and Stolte reproduced in this volume [2001].

When linear perspective is used, a limited field of view also prevents the distortions that appear in wide-angle views [Zorin 2002].

3.2 Elimination

Different technical solutions have been developed to increase the field of view of pictures. Interestingly, the width dimension has received more attention than the height dimension. The screen of movie theaters has been dramatically enlarged, and the 16/9 ratio is now a commercial argument to sell television sets. However, adaptation to the new format has not necessarily been a direct gain for directors of photography [American Society of Cinematographers 2000]. The stretched format obeys very different rules of composition. Some have argued that wide screen formats impede good composition and balance [Arnheim 1957].

Even wider fields of view are provided by IMAX theaters. Panoramas can be seen as their static equivalent. They were very popular in the 19th century, and invited visitors inside spherical or cylindrical buildings whose walls presented a 360 degree view of an outside environment. Panoramas could consist of a real three-dimensional foreground and a flat backdrop in order to increase the immersion. The painter David even advised his students to avoid going outside in order to study landscape, because they could receive the same experience in panoramas.

CAVEs [Cruz-Neira et al. 1993] and head-mounted display have been developed to offer a large field of view. Quicktime VR and other digital panorama encode the same information, but display only a limited portion at a time and allow the user to freely rotate the view [Chen 1995]. Chinese handrolls offer a virtually unlimited width.

3.3 Strategies

Following the frame Some pictorial styles such as Byzantine paintings and mosaics view the picture as a finite world, and no object may lie outside of its boundary. The drawing of objects might have to be distorted to fit within the frame (Fig. 17).

Figure 17: *Scene of the life of Abraham*, mosaic of the San Vitale basilica, in Ravenna, around 532. In this image, the 2D aspects are often prevail over 3D aspects: for example, the leftmost tree is bent to follow the limit of the image.

Framing In contrast, other art forms such as Japanese woodcuts exhibit foreground objects that can be dramatically cropped by the frame. The resulting composition is very dynamic and it reinforces the vision of the picture as a window to the world. 19th century painters such as Degas were strongly influenced by this technique (Fig. 18).

Figure 18: Edgar Degas, Edgar *The Dance Examination*, c. 1880.

Hybrid Not surprisingly, many styles involve a mixed strategy. The picture stands as a window to the depicted scene, but the artists carefully manages framing to make sure that all the important features lie within the frame and that cropping does not occur. This for example happens when we take a group photo: we ask people to move in order to fit everyone in the frame.

4 The picture is static

As our lives are more and more dominated by a flow of moving images, and since many in the computer graphics world think that motion pictures constitute the only noble goal, it is good to discuss the qualities and advantages of static pictures.

Static pictures allow more time and freedom for exploration, interpretation and analysis. Like the limitation in space allows the gathering and the relation of features in a restricted area, the limitation in time provides a compact and unified depiction. Moreover, aesthetically, the timeless nature of static images can bring a very balanced and immutable impression.

This once again motivates the use of compensation techniques, since they permit the depiction of motion while preserving the qualities of static images.

Since motion can be defined as the interaction between space and time, it is not surprising that some of the pictorial techniques dealing with these two domains are related. Conveying an impression of space is necessary to provide room for motion.

4.1 Partial elimination: motion pictures

The advent of the motion picture eliminated the limitation to static images. However, this elimination is only partial, and a film still has to fit in a limited amount of time. In most cases, the story takes a much longer time than the duration of the movie that tells it. This motivates the subtle art and craft of editing.

Note that compensation techniques might still be necessary, because the flatness of the picture also impedes the depiction of motion.

The elimination of the temporal limitation also has drawbacks. The pace of the artwork is entirely set by the director, while the beholder of a static image has more freedom. Recently, video textures have introduced as a fascinating way to create infinite yet non-repetitive motion sequences [**?**].

4.2 Compensation: motion depiction

Content Some objects imply motion by their mere presence in the picture. Flames and rivers are dynamic by essence. The purpose of vehicles is transportation and thus motion.

Pose Depicting an object in a pose that is not at rest suggests motion. This is one of the major differences between Egyptian and Greek art. Egyptian art attempts to depict the essential and immutable characteristics, while the Greek revolution introduced the rendering of motion, notably through sculptures and paintings of the human body in athletic poses [Gombrich 1995; Gombrich 1956].

An object that is not at equilibrium implies motion. Runners cannot remain still with both legs in the air. A plane falls if it is not moving. A curtain that is not straight and vertical must be subject to some wind or motion [Hagen 1980].

The dynamism of the pose can be accentuated to increase the impression of motion (Fig. 20). This is used for animation movies, although these pictures are not static [Thomas and Johnston 1981].

Motion lines and motion blur Motion blur corresponds to two related phenomena. In photography, due to the finite shutter speed, objects that are moving relative to the camera appear blurry. Nonetheless, motion blur has been used in paintings well before the advent of film and cameras, because it corresponds to the difficulty of our visual system

Figure 19: Toulouse-Lautrec *La Loie Fuller aux Folies Bergeres*, 1893. Motion is depicted through posture, blur, and vertical motion lines.

Figure 20: Jacques Henri Lartigue, *Grand Prix de l'Automobile Club de France*, 1912. The use of a vertical shutter plane and the motion of the car and camera result in a distortion of the car and spectators. The bottom top of the image is a more recent moment than the bottom. This gives the car a very dynamic car similar to a cartoon effect.

Figure 21: Diego Velasquez, *Las Hilanderas (the spinners)* 1657. Note the motion blur of the spinning wheel.

to stabilize moving objects (see also [Durand 2002a]). Velasquez was among the first to use this technique, as shown in Fig. 21 [Gombrich 1956].

The background or the moving target, or both can appear blurred.

Motion blur can be depicted using a simple blurry halo in the direction of motion, or using more distinctive lines. Lines can appear in photography due to more contrasted features of the object.

Arrows add a metaphorical effect to reinforce the depiction of movement.

Multiple snapshots Just like the limitation in space can be compensated for by the use of multiple perspectives in a single picture, motion can be depicted using multiple snapshot, either superimposed in a single image or in a contiguous series of images. Like for viewpoint exploration, we distinguish between the quasi continuous decomposition of motion (as in the chrono-photography of Muybridge or Marey, Fig. 23) and the use of discrete and clearly separated time steps, often used for story telling (as in graphics novels).

The most famous decomposition of motion is the *Nude Descending a Staircase* by Marcel Duchamp. In general, a decomposition of motion is more powerful than the moving sequence in that it permits the comparison of different phases and allow a different comprehension of the motion. This is the typical case where the limitation brings distant features in time in a common range. In the limit, a continuous decomposition of motion is similar to motion blur, as in the painting by Balla shown in Fig. 24.

The use of discrete time moments can take place in different frames (as in graphic novels, picture books, or in manuals), or can be blended in a single frame. This is often the case in Medieval and Renaissance biblical paintings, which include in a single picture the various episode of a story. This

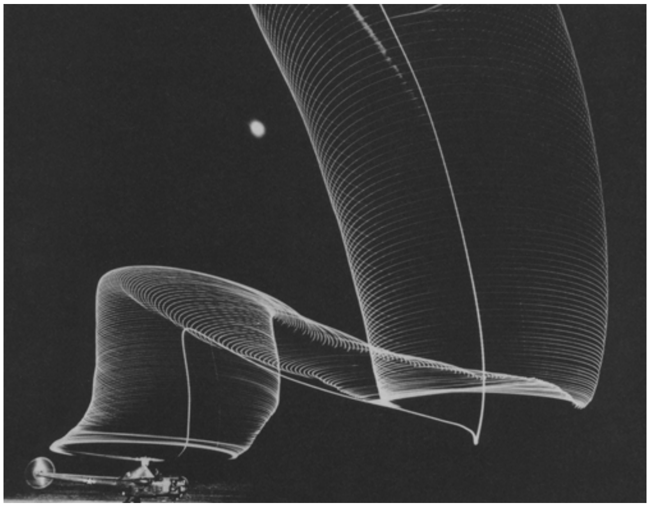


Figure 22: Feininger, *US Navy Helicopter Taking Off at Night*, 1949

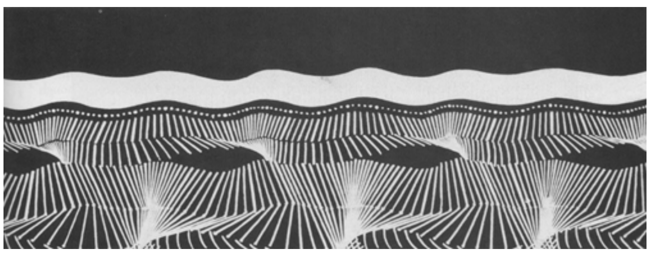


Figure 23: Etienne-Jules Marey, Chronophotograph.

is easy to undertand when the same character is represented multiple times (Fig. 25). But it can lead to important questions when, although each character is represented once, they are not represented at the same time, like in the *Last Supper* by Leonardo.

In his photomontages, Hockney combines gaze and viewpoint exploration as discussed above, but also multiple temporal snapshots. Parts of the scene that are moving are sampled by multiple gazes at different time steps.

At the frontier of motion decomposition, repetitive patterns can be used to symbolize motion and introduce dynamism in the picture. The famous painting *Broadway Boogie Woogie* by Mondrian translates the rhythm of music into a colorful pattern of squares. In Fig. 26, the leaves of the artichoke are both similar and evolving, which results in a very dynamic picture, the artichoke seem to be opening up.

Trading space for time Spatial dimensions can be used to represent the evolution of time. This is the case for timelines, but also usually the case for motion decomposition. For example, in Fig. 23, the horizontal dimension also represents time.

Photos taken at the arrival of races (photofinish) explicitly use the horizontal dimension to encode time, which can result in surprising distortions.

Figure 24: Giacomo Balla, *Dynamism Of A Dog On A Leash*, 1912.

Figure 25: Sassetta, *The Meeting of St Anthony and Saint Paul*, 1440.

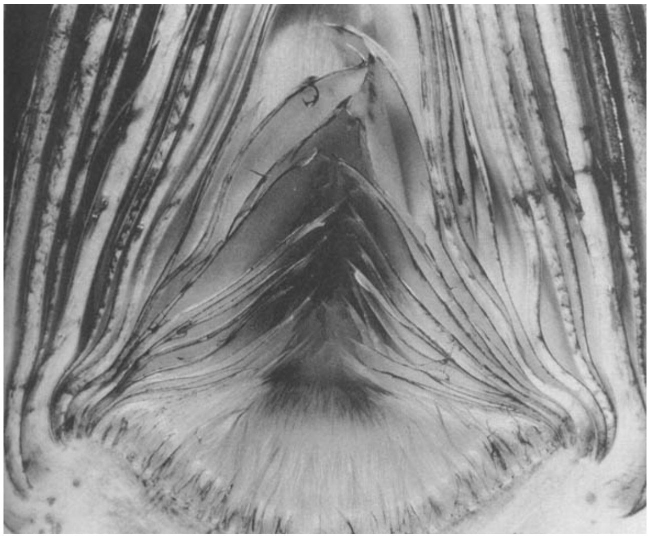


Figure 26: Weston *Artichoke Halved*, 1930.

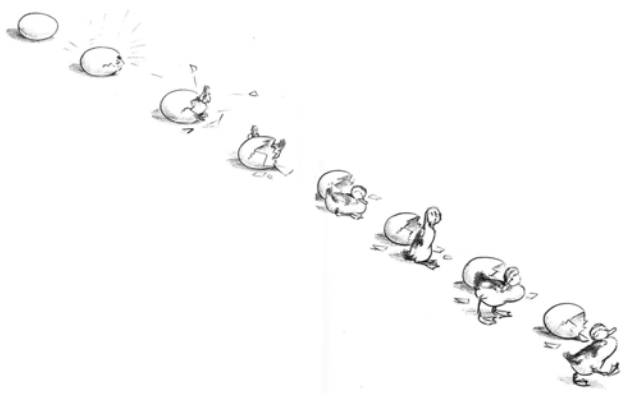


Figure 27: Drawing from *Making way for ducklings* by Robert McCloskey. The picture represents 8 different ducklings, but they can also be seen as a temporal decomposition of the eclosion. Composition moreover uses the diagonal dimension to depict time.

Viewpoint and composition A dramatic viewpoint can increase the impression of dynamism. As discussed above, increasing the impression of space provides more room for motion. A viewpoint that emphasizes perspective and the impression of space in the direction of motion is a great way to accentuate the dynamism, be it for a static or motion picture.

The composition of the image can also reinforce motion. Space can be left void in front of or behind the object in motion, which provides space for the movement. Framing that cuts the moving subject can imply that the object is leaving or entering the field of view because of the motion.

More generally, the composition can direct the gaze of the beholder along the path of motion.

Perceptual effect Some art forms such as the Op'Art play with the low-level mechanisms of our visual perception to obtain pictures that seem to be shimmering. They use fine-scale repetitive patterns that are at the limit of our acuity, which creates Moiré effects. And because our acuity varies between the center of our visual field and the periphery, our gaze movements cause the perceived animation of the picture.

Recently, Livingstone hypothesized that the ever-changing smile of Mona Lisa might be due to low-level perceptual effects. She argues that due to different frequency cutoffs, the smile seen by our peripheral vision is not the same as the smile seen when she is in the middle of our visual field. While it is clear that Leonardo was not aware of the frequency response of our visual system, the ambiguity of the smile was an effect he desired. He mastered the art of *sfumato*, which means blurring some key parts of the portrait. This leaves more freedom to the interpretation by the beholder, and thus make the portrait more lively [Da Vinci 1989].

In the computer graphics community, Freeman and Adelson have developed a technique to produce motion without movement [Freeman et al. 1991]. Their pictures are not strictly static but they manage to imply an impression of motion although the objects do not move in the picture. Their technique can be useful to display motion information without altering the spatial organization of pictures, e.g. for forecast maps or for airport control. This is a typical example of compensation that permits a seemingly impossible separation: motion that is normally a relation between space and time is represented without affecting space.

5 The gamut and contrast are limited

Pictures are also limited in their ability to reproduce the light color and intensity. The issue of color reproduction raises exciting psychological and technological issues. We invite the interested reader to consult the related slides in the present volume [Ostromoukhov 2002] and the seminal book by Hunt [Hunt 1995]. We focus on intensity , since it is the limitation of picture that has received the most explicit attention in computer graphics, since its introduction by Tumblin and Rushmeier [1993]. We focus on photography, because this is the art form where contrast management is explicitly paramount.

The human visual system performs effectively over a vast range of luminous intensities, ranging from below starlight, at $10^{-6}cd/m^2$, to sunlight, at $10^6cd/m^2$. The visual system copes with the high dynamic range present in real scenes by varying its sensitivity through a process known as *visual adaptation*. Unfortunately, pictures are usually limited to a much lower contrast. The darkest available black is usually about a hundred times darker than pure white. Since it is not rare to see scenes with a range of intensity as high as 1 to 100,000, pictorial techniques are required to manage the contrast and compress the range of intensity.

The elimination of this limitation has received a lot of technological attention. The improvement of film chemistry and paper contrast is a major endeavor in photography, and the contrast of videoprojectors or display is a large concern for manufacturers. Other techniques exists, such as the inclusion of active light source directly in the picture, or the use of phosphorescent paint. In his famous experiment where he demonstrated linear perspective, Bruneleschi seems to have used a mirror to depict the sky. This trick greatly increases the realism of the picture, because the reflected sky can then be much brighter than the brightest intensity available on the painting.

However, similar to other limitations, the limited contrast also has crucial advantages. In particular, by reducing their intensity difference, it permits relations that are not possible in the real scene (Fig. 28).

Figure 28: W. Eugene Smith, portrait of Dr. Albert Schweitzer – "A Man Of Mercy" Series, 1954. Note how the balance between the lamp and the shirt result in a carefully balanced composition. In the real scene, the difference of intensity between these two features would prevent their relation. The limitation of the field of view moreover permit the balanced placement of these two features within the frame.

The limited range of intensity in pictures raises two issues. First, the absolute intensity of the real scene cannot be reproduced on the picture, and since our vision does not function equally at all light levels, some adjustments must be performed if faithful reproduction is required. Second, the range of intensity must be compressed to fit in the limited contrast of the medium.

5.1 Coping with different absolute intensities

We are able to cope with large absolute intensities because our visual system is mainly sensitive to contrast, that is, to intensity ratios rather than to absolute values. In a very luminous scene, a black object can be as luminous as $1000cd/m^2$ (the precise definition of the unit is not important), while in a dark scene, an object can appear white although it is much darker, say around $0.01cd/m^2$. This is because our notion of black or white is based on the comparison between intensities in the scene. The object at $1000cd/m^2$ can appear dark if the rest of the scene is around $100,000cd/m^2$. See also the part about color in this volume [Ostromoukhov 2002] or books on visual perception [Palmer 1999; Bruce et al. 1996; Wandell 1995].

Gamma However, our vision does not function strictly equally at all light level. In particular, our perception of brightness varies with the intensity in the scene. Similar intensity ratios will look more contrasted in more luminous scenes, and colors look more vivid [Hunt 1995; Fairchild 1998; Stevens 1961; Stevens and Stevens 1963; Hunt 1952]. This is why sunny landscapes look more vivid. When we get back from vacation with photos, we typically look at them in a darker surrounding, which results in a duller appearance. To cope with this, the response of films is tuned to increase their contrast. Technically, this is described by a gamma correction, which is a simple deviation from linear reproduction using a γ exponent [Hunt 1995; Stroebel et al. 1986] (Fig. 30). Gamma correction is probably the least well understood issue of tone reproduction, because the same term is used to describe different phenomena. Gamma correction is related to the compensation for darker surrounding as described above, but also to the non-linearity of the response of CRT monitors, and finally to color quantization issues. See the excellent FAQ by Charles Poynton [Poynton 2002].

A related phenomenon is the Bartleson-Breneman effect [Hunt 1995; Fairchild 1998]. It describes counter-intuitive results on the effect of the surround of an image. When the surround is darker, the perceived contrast is lower, which is contrary to the common practice of turning off the light to enjoy a more vivid television image. In fact, a darker surround decreases the perceived contrast because when we compare the black on the image with the surrounding, we realize that it is grayish. In contrast, a bright surround does not really affect our perception of white. However, turning off the light does also have positive effects on contrast because usually the light is reflected or produces flares on the screen. The best solution is thus to illuminate the surrounding without illuminating the screen, which is usually not easy. However, the immersion is stronger when we do not see the surrounding.

Night scenes The impression of night is elusive and challenging. In the real world, night is characterized by lower light intensity, but in a picture, the impression of night must be conveyed using the same absolute luminous range as for pictures of sunny scenes. Artists usually rely on the use of darker tone, low saturation, but also on a blue shift (Fig. 29). Indeed, night vision is handled by rods, which does not permit the distinction of colors. Moreover, psychophysics experiments have shown that as luminosity decreases, we tend to judge colors as slightly more bluish [Hunt 1952]. The existence and the explanation of this blue shift are still a subject of discussion. For cinematographic pictures, many night scenes are actually filmed during the day (day-for-night) and then post-processed, lowering the contrast and using blue filters [Millerson 1991].

Figure 29: Camille Pissaro Top: *Boulevard de Montmartre: Afternoon Sunshine*, 1897. Bottom: *Boulevard de Montmartre: Night*, 1897

5.2 Contrast management

We now discuss techniques, mostly in photography, to compress the dynamic range of real scene into the limited contrast of the medium. Some techniques are global and act equally on the whole image, while local methods permit a finer control by managing tone and contrast in a local spatial portion of the photo.

Compressive response The response of films is not linear, but has an S-shape (Fig. 30) [Adams 1995; Stroebel et al. 1986; Geigel and Musgrave 1997]. This means that intensities are not reproduced uniformly. If linear reproduction was used, too much clamping would occur for dark and bright tone because of the limitations of the print. Instead, the slope of the reproduction curve is lower for dark and bright tones (toe and shoulder of the curve), which means that contrast is reduced in these ranges, but that the range of intensities that can be reproduced is higher.

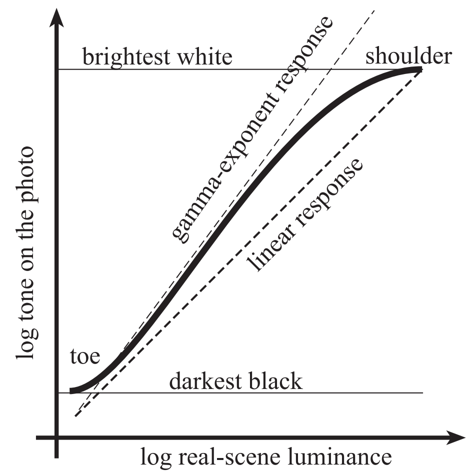
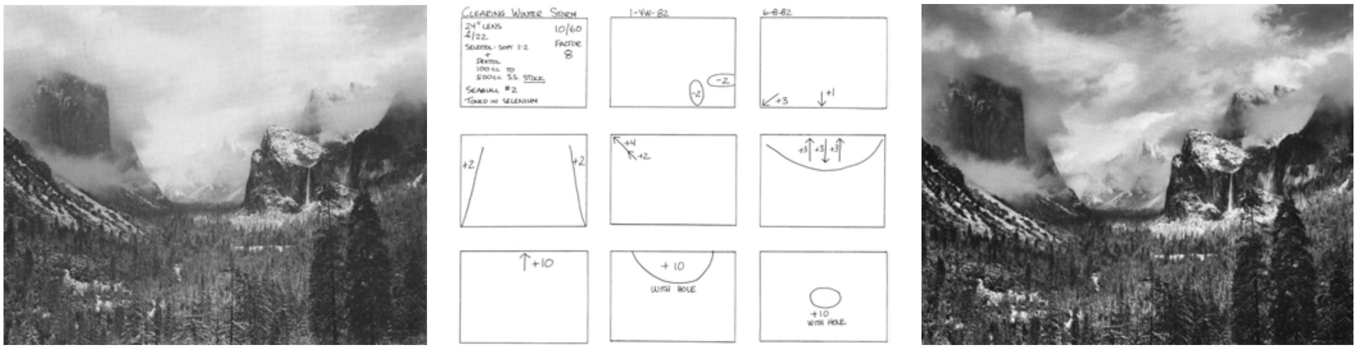


Figure 30: Tone reproduction curve for a typical film.

This curve can then be managed by the photographer by choosing various chemical and reaction time during the processing of the negative, and using different papers and methods for the print [Adams 1995].

dodging and burning Negatives provide a higher dynamic range than the final print. This means that more information is encoded in the negative. This leaves some latitude during the printing process, and allows photographers to alter tone locally. They do this through *dodging* and *burning* [Adams 1995; Rudman 2001; Schaub 1999]. This technique is used when the print is exposed. It consists in masking for a certain time part of the print by placing pieces of cardboard or the hand between the projector and the print. This causes the masked areas to be whiter, because they are less exposed. Slightly moving the mask is essential to obtain a soft boundary and avoid artifacts. Dodging and burning has to be performed for each print (Fig. 31).



straight print   dodging and burning notes   final print

Figure 31: Ansel Adams, *Clearing Winter Storm*.

Chemicals can also be used locally to alter the tones [Rudman 2001].

Lighting In many cases, the only way to manage the contrast is to actually modify the lighting in the real scene to reduce the range of intensities. In particular, one of the most classical lighting scheme for objects and characters is called *three-point lighting* [Millerson 1991; Lowell 1999] (Fig. 32). A fill light is used to diffusively illuminate the shadow and bring them within the reproducible range. Often, a passive reflector is used instead of a lamp.

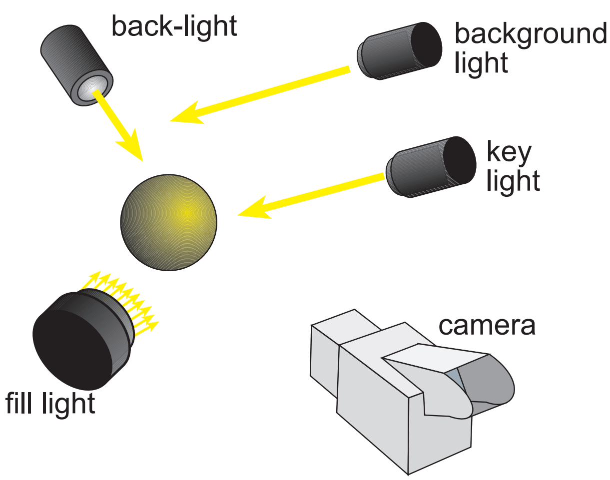


Figure 32: Three point lighting (adapted from [Millerson 1991]). The *fill* lamp permits to fill-in shadows and to reduce the overall contrast. The back light emphasizes the silhouettes of the subject. The background often has a different lighting.

Interior photography is a particularly challenging field, because of the high contrast between the often dark interior and the bright exterior seen through windows. The only solution is then to add more artificial lights inside. The balance between the tones of the interior and exterior parts is then paramount to set the ambiance and depth of the space [Harris 1998].

Lighting can also be controlled locally through the technique of *painting with light*. In a dark scene, a very long exposure is used (in the range of tens of minutes) while the photographers moves within the scene with a flash and illuminates various parts.

Filters The use of filters during when the image is taken is another way to manage contrast, and in particular to handle the sky. Indeed, the sky is usually too bright, which tends to saturate the film and result in a flat white sky in the picture. Many filters can cope with this [Kodak 1981]. First, a polarizing filter can take advantage of the *polarization* of light [Falk et al. 1986; University of Colorado at Boulder 2002]. It is well known that light can be seen as a wave that travels in straight light. Usually, light is not polarized and is composed of waves in all orientation relative to the direction of transport (Fig. 33). But some phenomena can polarize light, such as reflection or scattering in the sky: The light reflected by most materials or the light coming from the sky have a preferred orientation. A polarizing filter can block light in a certain orientation, which the user controls by rotating the filter. Because skylight and reflection are polarized, they are more affected by the polarizer, which can be used to darken the sky.

A gradient filter can also be used to make the upper part of the image (the sky usually) darker. Finally, in black and white photography, various colored filter can block specific colors. For example, a red filter makes the sky much darker [Kodak 1981].

Flare Flares or halos can be used to increase the subjective brightness of parts of the picture. Flare is actually experienced by the human visual system because of scattering in the optical system of the eye. In photography, flares can be obtained using special filters [Kodak 1981]. In computer graphics, Spencer et al. have proposed a digital flare filter based on physiological data [Spencer et al. 1995].

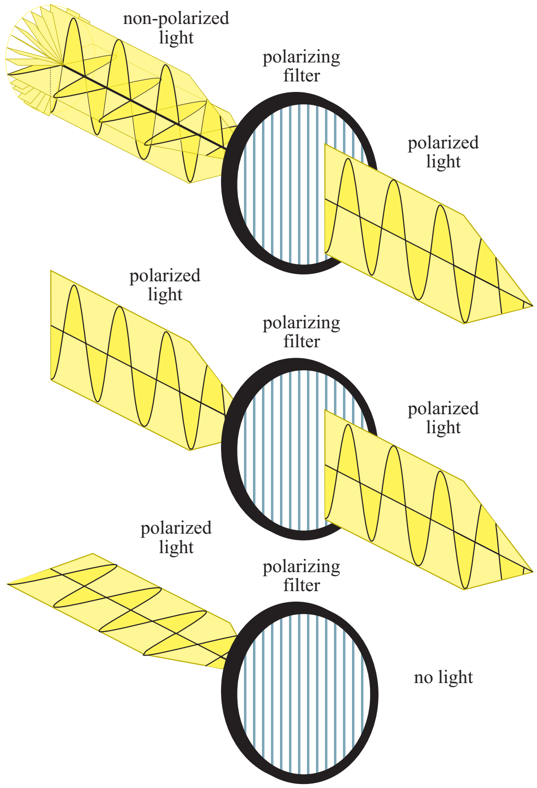


Figure 33: Polarizing filter.

Acknowledgment

Many thanks to Barb Cutler for intense proofreading. She is however not reponsible for the errors that I have not corrected.
Yes, I meant Barb, not Bob.

References

ADAMS, A. 1995. *The Camera+The Negative+The Print*. Little Brown and Co.

AGRAWALA, M., AND STOLTE, C. 2001. Rendering effective route maps: Improving usability through generalization. In *Computer Graphics (Proc. SIGGRAPH)*.

AGRAWALA, M., ZORIN, D., AND MUNZNER, T. 2000. Artistic multiprojection rendering. In *Eurographics Workshop on Rendering*.

AMERICAN SOCIETY OF CINEMATOGRAPHERS, 2000. Visions of light. DVD.

ARIJON, D. 1991. *Grammar of the Film Language*. Silman-James Press.

ARNHEIM, R. 1954. *Art and Visual Perception: a psychology of the creative eye*. U. of California Pr.

ARNHEIM, R. 1957. *Film as Art*. U. of California Pr.

AUCOIN, K. 1999. *Making Faces*. Little Brown and Co.

BARBOUR, AND MEYER. 1992. Visual cues and pictorial limitations for computer generated photorealistic images. *the Visual Computer 9*.

BLINN, J. F. 1988. Jim blinn's corner: Me and my (fake) shadow. *IEEE Computer Graphics & Applications 8*, 1 (January), 82–86.

BRUCE, V., GREEN, P., AND GEORGESON, M. 1996. *Visual Perception : Physiology, Psychology and Ecology*, 3rd ed. Psychology Pr.

CHEN, E. 1995. Quicktime VR - an image-based approach to virtual environment navigation. *Proc. of SIGGRAPH*.

CRUZ-NEIRA, C., SANDIN, D. J., AND DEFANTI, T. A. 1993. Surround-screen projection-based virtual reality: The design and implementation of the CAVE. In *Computer Graphics (SIGGRAPH '93 Proceedings)*, J. T. Kajiya, Ed., vol. 27, 135–142.

DA VINCI, L. 1989. *Leonardo on Painting : An Anthology of Writings by Leonardo Da Vinci With a Selection of Documents*. Yale University Pr.

DURAND, F. 2002. Gaze exploration and focal points. In *Perceptual and artistic principles for effective computer depiction*, F. Durand, Ed. ACM SIGGRAPH course notes.

DURAND, F. 2002. Gestalt and picture composition. In *Perceptual and artistic principles for effective computer depiction*, F. Durand, Ed. ACM SIGGRAPH course notes.

FAIRCHILD. 1998. *Color Appearance Models*. Addison-Wesley.

FALK, D., BRILL, D., AND STORK, D. G. 1986. *Seeing the Light: Optics in Nature, Photography, Color, Vision and Holography*. John Wiley & Sons, New York.

FREEMAN, W. T., ADELSON, E. H., AND HEEGER, D. J. 1991. Motion without movement. *Computer Graphics 25*, 4, 27–30.

GEIGEL, J., AND MUSGRAVE, K. 1997. A model for simulating the photographic development process on digital images. *Proc. SIGGRAPH*.

GLASSNER, A. 2000. Cubism and cameras: Free-form optics for computer graphics. Tech. Rep. MSR-TR-2000-05, Microsoft Research.

GOMBRICH. 1956. *Art and Illusion*. Princeton Press.

GOMBRICH, E. H. 1995. *The Story of Art (L'histoire de l'art)*. Phaidon Press. 16th edition.

GOOCH, B., REINHARD, E., MOULDING, C., AND SHIRLEY, P. 2001. Artistic composition for image creation. In *Eurographics Workshop on Rendering*.

GOOCH, B. 2002. Ramachandran and hirstein's neurological theories of aesthetic for computer graphics. In *Perceptual and artistic principles for effective computer depiction*, F. Durand, Ed. ACM SIGGRAPH course notes.

HAGEN, M. 1980. *The perception of pictures*. New York: Academic Press.

HARRIS, M. G. 1998. *Professional Interior Photography*, 2nd edition ed. Focal Press.

HARRIS, M. G. 2002. *Professional ArchitecturalPhoto Photography*, 3rd edition ed. Focal Press.

HOCKNEY, D. 1993. *That's the Way I See It*. Chronicle Books.

HU, H. H., GOOCH, A. A., THOMPSON, W. B., SMITS, B. E., RISER, J. J., AND SHIRLEY, P. 2000. Visual cues for imminent object contact in realistic virtual environments. In *Visualization 2000*.

HUNT. 1952. Light and dark adaptation and the perception of color. *JOSA A 42*, 3, 190.

HUNT. 1995. *The reproduction of Color (5th ed.)*. Kings Langley:Fountain Press.

KEMP, M. 1990. *The Science of Art*. Yale University Press.

KODAK. 1981. *the Kodak Workshop Series: Using filters*. Kodak Editor.

KUBOVY, M. 1986. *The Psychology of Perspective and Renissance Art*. Cambridge University Press.

LOWELL, R. 1999. *Matters of Light and Depth*. Lowel-Light Manufacturing.

MILLERSON. 1991. *Lighting for Television and Film, 3rd ed.* Focal Press.

MINNAERT, M. G. J. 1993. *Light and Color in the Outdoors*. Springer Verlag.

OSTROMOUKHOV, V. 2002. Color in art and science. In *Perceptual and artistic principles for effective computer depiction*, F. Durand, Ed. ACM SIGGRAPH course notes.

PALMER, S. 1999. *Vision Science, Photons to Phenomenology*. MIT Press.

POYNTON, C., 2002. Gamma FAQ. http://www.inforamp.net/ poynton/GammaFAQ.html.

RAMACHANDRAN, V., AND HIRSTEIN, W. 1999. The science of art: A neurological theory of aesthetic experience. *Journal of Consciousness Studies 6*, 6/7.

RUDMAN, T. 2001. *The Photographer's Master Printing Course*. Boston: Focal Press.

SCHAUB, G. 1999. *The Digital Darkroom : Black-And-White Techniques Using Photoshop*. Silver Pixel Press.

SOLSO, R. L. 1994. *Cognition and the visual arts*. MIT Press.

SPENCER, SHIRLEY, ZIMMERMAN, AND GREENBERG. 1995. Physically-based glare effects for digital images. In *Computer Graphics (Proc. Siggraph)*.

STEVENS, AND STEVENS. 1963. Brightness functions: Effects of adaptation. *JOSA A 53*, 375.

STEVENS. 1961. To honor Fechner and repeal his law. *Science 133*, 80.

STROEBEL, L., COMPTON, J., CURRENT, I., AND ZAKIA, R. 1986. *Basic Photographic Materials and Processes*, second edition ed. Boston: Focal Press.

THOMAS, F., AND JOHNSTON, O. 1981. *The Illusion of Life: Disney Animation*. Abbeville Pr.

TUMBLIN, J., AND RUSHMEIER, H. 1993. Tone reproduction for realistic images. *IEEE Comp. Graphics & Applications 13*, 6.

UNIVERSITY OF COLORADO AT BOULDER, 2002. Polarization, in Physics 2000. http://www.colorado.edu/physics/2000/polarization/.

VON HELMHOLTZ, H. 1925. *Treatise on Physiological Optics*. Dover.

WANDELL, B. A. 1995. *Foundations of Vision*. Sinauer.

WANGER, L., FERWERDA, J., AND GREENBERG, D. 1992. Perceiving spatial relationships in computer-generated images. *IEEE Computer Graphics and Applications 12*, 3, 44–58.

ZAKIA, R. D. 1997. *Perception and Imaging*. Butterworth-Heinemann.

ZEKI, S. 2000. *Inner Vision : An Exploration of Art and the Brain*. Oxford U. Pr.

ZORIN, D. 2002. Perception of pictures. In *Perceptual and artistic principles for effective computer depiction*, F. Durand, Ed. ACM SIGGRAPH course notes.

Investigating the Effect of Texture Orientation on the Perception of 3D Shape

Victoria Interrante and Sunghee Kim
Department of Computer Science and Engineering
University of Minnesota

ABSTRACT

Perception of the 3D shape of a smoothly curving surface can be facilitated or impeded by the use of different surface texture patterns. In this paper we report the results of a series of experiments intended to provide insight into how to select or design an appropriate texture for shape representation in computer graphics. In these experiments, we examine the effect of the presence and direction of luminance texture pattern anisotropy on the accuracy of observers' judgments of 3D surface shape. Our stimuli consist of complicated, smoothly curving level surfaces from a typical volumetric dataset, across which we have generated four different texture patterns via 3D line integral convolution: one isotropic and three anisotropic, with the anisotropic patterns oriented across the surface either in a single uniform direction, in a coherently varying direction, or in the first principal direction at every surface point. Observers indicated shape judgements via manipulating an array of local probes so that their circular bases appeared to lie in the tangent plane to the surface at the probe's center, and the perpendicular extensions appeared to point in the direction of the local surface normal. Stimuli were displayed as binocularly viewed flat images in the first trials, and in stereo during the second trials. Under flat viewing, performance was found to be better in the cases of the isotropic pattern and the anisotropic pattern that followed the first principal direction than in the cases of the other two anisotropic pattems. Under stereo viewing, accuracy increased for all texture types, but was still greater for the isotropic and principal direction patterns than for the other two. Our results are consistent with a hypothesis that texture pattern anisotropy impedes surface shape perception in the case that the direction of the anisotropy does not locally follow the direction of greatest normal curvature.

Keywords: texture, shape representation, principal directions, shape perception.

1 INTRODUCTION

A key objective in the field of visualization is to design and implement algorithms for effectively communicating information through images. Given a set of data, we must design a visual representation for that data which facilitates its understanding. The investigations reported in this paper were motivated by applications in which one needs to be able to accurately and intuitively convey the three-dimensional shape of large, smooth, arbitrarily curving surfaces. Previous studies [cf. Interrante *et al.* 97] have indicated that shape perception can facilitated by the addition of surface texture markings, but the question of how to characterize the kind of surface texture that will show shape best remains open. Texture can also be used to *mask* surface shape features, as was shown by Ferwerda *et al.* [1997]. It has been suggested [Cumming *et al.* 1993] that the perception of shape from texture can be impeded when the texture pattern is highly anisotropic, consisting of elements that are systematically elongated in a specific direction. However a wide variety of textures consisting of line-like elements have been shown to indicate surface curvature [Todd and Reichel 1990]. Knill [1999] shows that across developable surfaces[1] any homogeneous texture pattern will appear to flow along parallel geodesics[2], and suggests that our visual system uses shape-from-contour[3] to infer shape from the systematic projective distortion or flow of the pattern. Inspired by the considerable amount of research [Stevens 1983, Mamassian and Landy 1998, Li and Zaidi 2000] that seems to imply that surface shape may be perceived most accurately from line-like markings when they follow the lines of curvature, we sought in the series of experiments described in this paper to further experimentally investigate the effect of the direction of surface texture pattern anisotropy on the accuracy of observers' shape judgments. Specifically we wanted to know: can an anisotropic pattern that follows the principal directions show shape more effectively than a pattern in which the direction of anisotropy follows some other path? Than an isotropic pattern? Are the effects the same in the case of shaded displacement texture? To what extent are these effects mitigated by stereo viewing?

Correspondence: Interrante; Other author info: VI; email: interran@cs.umn.edu; web: www.cs.umn.edu/~interran; SK; email: skim@cs.umn.edu; web: www.cs.umn.edu/~skim.
[1] surfaces that can be unrolled to lay out flat in the plane;
[2] curves that do not turn in the surface
[3] the set of all points at which the surface normal is orthogonal to the line of sight

2 METHODS

We conducted a series of two experiments intended to investigate the effect of the presence and direction of texture pattern anisotropy on the ability of observers to accurately perceive the 3D shape of a smoothly curving surface. The goal of these experiments was to gain insight that might facilitate our efforts to use texture most effectively to facilitate the accurate perception of surface shape in renderings of scientific data. In the following sections we provide the details of the experimental set up and design.

2.1 Stimuli

The stimuli that we used in our experiments were cropped images of the front-facing portions of textured level surfaces rendered in perspective projection using a hybrid renderer [Interrante *et al.* 97] that uses raycasting [Levoy 88] together with a Marching Cubes algorithm [Lorensen and Cline 87] for surface localization. The volumetric test data from which we extract these surfaces is a three dimensional dose distribution calculated for a radiation therapy treatment plan. We chose to use the radiation data as our testbed, rather than a more restricted type of analytically-defined surface, because this data is typical of the kind of data whose shape features we seek to be able to more effectively portray through the use of surface texture.

The first step in image generation was to define the solid texture patterns that would appear on the level surfaces. We used a high-quality three-dimensional line integral convolution algorithm [Stalling and Hege 95] to synthesis the textures in the vicinity of the selected level surface. Beginning with a three-dimensional array of binary noise, line integral convolution produces an output texture in which the input values are correlated along the directions indicated by an accompanying vector field. We defined four different vector fields to produce four different types of texture patterns.

The procedure that we used to obtain the principal direction vector field is fully described in [Interrante 97], but is briefly restated here for completeness. We begin by computing an orthogonal frame at each sample point in the 433x357x325 voxel 3D volumetric dataset. We define the third frame vector to be in the direction of the grey-level gradient, which is the normal to the level surface that passes through the sample. We compute the gradient using Gaussian-weighted central differences in the axial directions over the 3x3x3 area surrounding the sample point. We next choose an arbitrary point in the tangent plane to define the direction of the first frame vector and then take the cross product of these two vectors to obtain the remaining orthogonal direction. Finally, we estimate the 2nd Fundamental Form [Koenderink 90] from the Gaussian-weighted central differences of the gradients trilinearly interpolated at sample positions over a 3x3x3 grid aligned with the local frame, diagonalize to obtain the 2D principal directions (eigenvectors) and principal curvatures (eigenvectors) in the tangent plane, and convert to 3D object space coordinates. The direction corresponding to the eigenvalue with the greatest unsigned magnitude is saved in the 3D principal direction vector array and used to create the first anisotropic texture ('pdir').

The remaining 3D vector fields are obtained by simpler means. First, we obtain the vector field of uniform directions by taking at each point the direction given by the intersection of the tangent plane with the plane orthogonal to the z axis that passes through the sample point: $udir_x = -n_y$, $udir_y = n_x$, $udir_z = 0$, where (n_x, n_y, n_z) is the surface normal or gradient. Then, we obtain the vector field of random directions that is used to create the isotropic texture pattern by rotating the uniform direction previously obtained at each point by a random angle θ_1 about the surface normal, $-\pi/2 \le \theta_1 \le \pi/2$. Finally, we obtain the vector field of coherently varying directions that is used to create the anisotropic texture pattern that contains lines with non-zero geodesic curvature by rotating the original uniform direction about the surface normal by an angle $\theta_2 = 10\pi(x+y+z)/n$ where (x,y,z) is the index of the sample point in the volume and n is the total number of sample points in the 3D array.

Figure 1 illustrates the process of texture synthesis, showing a single slice (z=263) from the 3D input texture volume and from each of the different 3D output texture volumes. Not all of the values are filled in, because we have elected to initiate the streamlines that are used to compute the output texture values only at the voxels that are in the vicinity of the level surface that is being used to create the test image.

During rendering, the intensity value interpolated from the 3D texture at the ray/surface intersection point is taken as the base color of the surface at the ray surface intersection point, and Phong shading is then applied to obtain the final surface color. We rendered 48 test images for the experiment, 24 for the left eye views and 24 for the right eye views, using the four different textures applied to views from six different vantage points around a single level surface. Figure 2 shows three of these images, all computed for the same viewing position. In order to avoid the potentially confounding influence of shape-from-contour information, as a last step we cropped each image to a 400x400 pixel region that did not contain any points on the silhouette edges of the object.

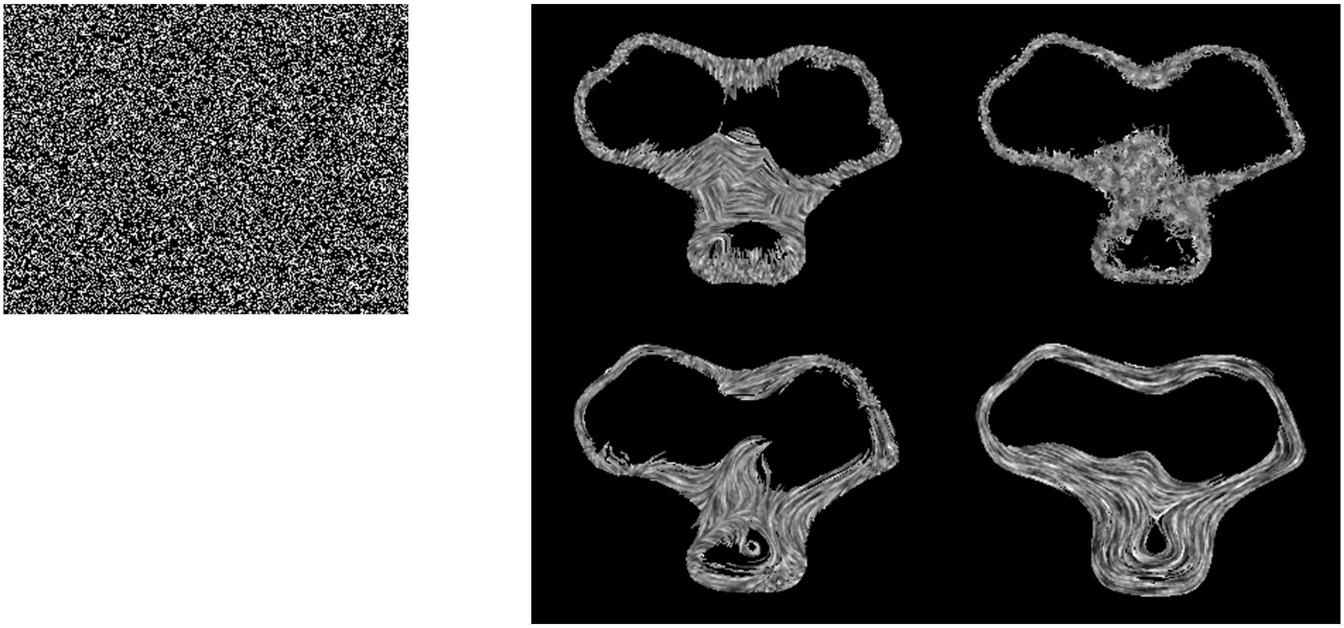


Figure 1: Slices from the 3D solid textures. Left: The slice z=263 before line integral convolution; Right: The same slice after line integral convolution along (in clockwise order) first principal directions (pdir), random directions (rdir), uniform directions (udir) and coherently varying or swirling directions (sdir) computed at each sample point.

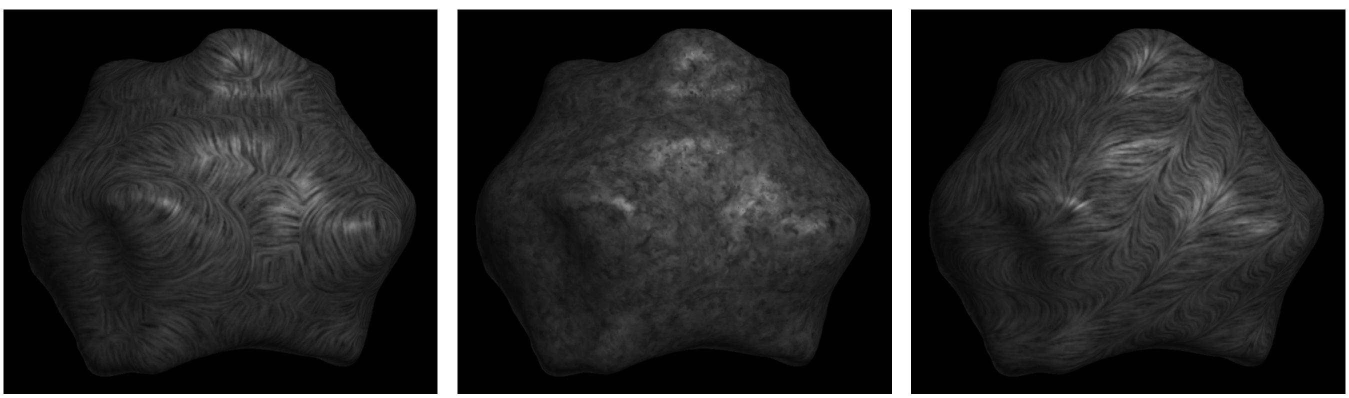


Figure 2: Examples of the 3D textured surfaces. From left to right: pdir, rdir and sdir. Note that informal assessment of the potential impact of texture type on shape judgments is complicated in these images by the prominence of shape-from-contour cues, which tend to dominate when other information about shape is less readily accessible. Because we are most interested in studying how the presence of texture might facilitate shape judgments across non-trivially structured interior regions where shape-from-contour information is not available, we cropped all of the images to eliminate the edge cues before testing.

2.2 Task

In originally planning these investigations, we had hoped to be able to design an experimental task that could reveal the effect of different texture types on the accuracy and efficiency of an observer's perception of the global 3D shape of a displayed object (shape from a glance). However we had great difficulty coming up with a means to evaluate observers' immediate global impressions of surface shape in a way that avoided confounding influences such as isolated 2D feature recognition or partial picture matching. Hence we decided to proceed with estimates of surface shape perception accumulated from individual judgments of the orientation of the surface at local points. Because it is well known that our visual system does not build up an estimate of shape from the accumulation of isolated individual local estimates of surface heading, but rather obtains shape understanding from the comparative relationships between nearby points, we decided to present an array of probes [Koenderink *et al.* 1992] that completely covered the central area of the presented surface and to ask observers not only to adjust each probe by pulling on its handle until the circular base appeared to lie in the tangent plane to the surface at its central point and the perpendicular extension appeared to point in the surface normal direction, but also before proceeding to the next trial to verify that the shape of the surface they had implicitly indicated through the collective orientations of all of the probes appeared to faithfully match the shape of the underlying textured surface at all points.

Unfortunately, we neglected to recognize, before beginning the experiments, that our decision to place the probes at exactly evenly spaced intervals over a rectangular grid would interfere with observers' ability perceive all of the probes as lying in the surface at the same time, due to violation of the generic viewpoint assumption. (If the probes did all lie in a smooth surface that varied in depth, and still appeared to be evenly spaced in a single view, then any tiny translation of the viewing position would have to break the symmetry of the spacing. Our visual system hence preferentially adopts the more likely interpretation that the probes are arrayed on a transparent flat plane in front of the underlying curved surface.) Our subjects did not report an inability to see the probes as lying in the surface on an individual basis, but, as will be discussed later, certain of the individual responses appeared to indicate that the probes were not always consistently visualized as a coherent unit across each image. Figure 3 shows the user interface at the beginning of the 5th trial.

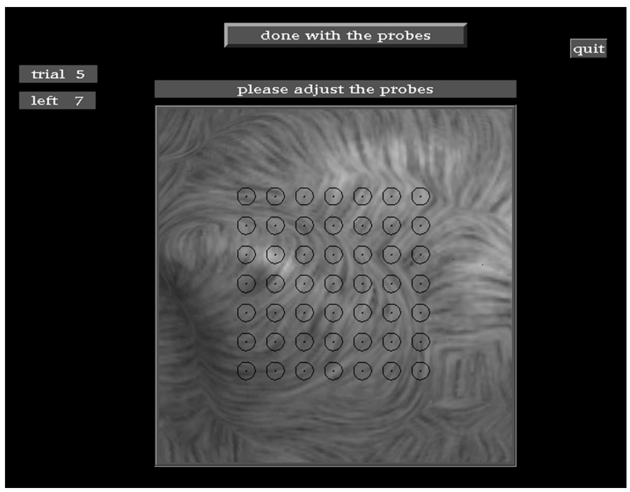


Figure 3: The graphical user interface with all probes displayed in their starting positions.

In designing the experiment we were particularly concerned about avoiding a situation in which differences dues to texture type might be confounded with differences due to other unanticipated or uncontrolled factors such as individual differences, or particular surface shape configurations. Ideally, we would have liked to present identical views of each surface under all four texture conditions, and to have all subjects make judgements on all of the images. However we were also concerned about the possibility of subjects' current shape judgments being biased by information they obtained from previous trials in which the same surface had been shown under a different texture condition. With only 24 binocular images (6 views x 4 texture types), and an small anticipated subject pool size, we had to make some difficult tradeoffs. What we did was to divide the subjects and the stimuli into two different groups, so that each subject made shape judgments at the 49 probe locations on only half of the data (12 images). Each set contained each view and each texture type, in equal proportions, but did not contain all of the possible combinations. Within each set, the stimuli were further grouped into two lots, in which each lot had no surface repeated. Figures 4 and 5 show the complete set of stimuli presented to each group of observers.

The six images within each lot were presented in random order, and subjects were required to take a 10 minute rest break after finishing the 6th trial, thereby avoiding the possibility that any two differently textured but identically shaped surfaces might be presented immediately in sequence, and minimizing the likelihood of surface recognition and any consequent possible learning effects. After adjusting the probes on the 12 images in the flat viewing condition, subjects repeated the entire process under conditions of stereo viewing. To facilitate estimation of the effects of viewing condition, subjects were presented with the same stimuli, differently ordered, in the two viewing conditions. The entire process took about two hours for most of the subjects.

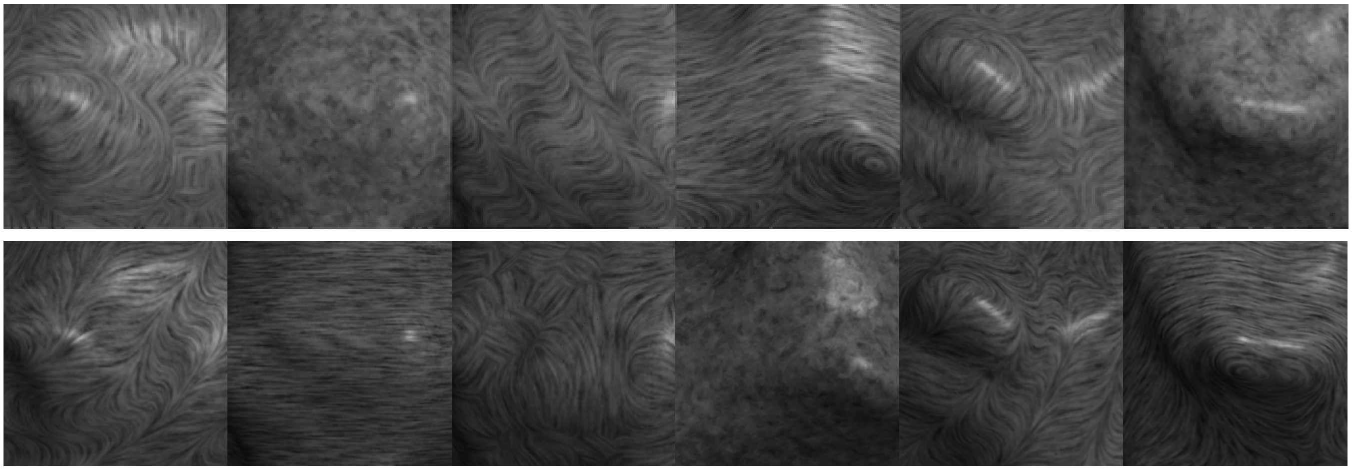


Figure 4: The set of stimuli seen by group A. First row: lot 1; second row: lot 2. The presentation order was randomly determined and was different for each subject.

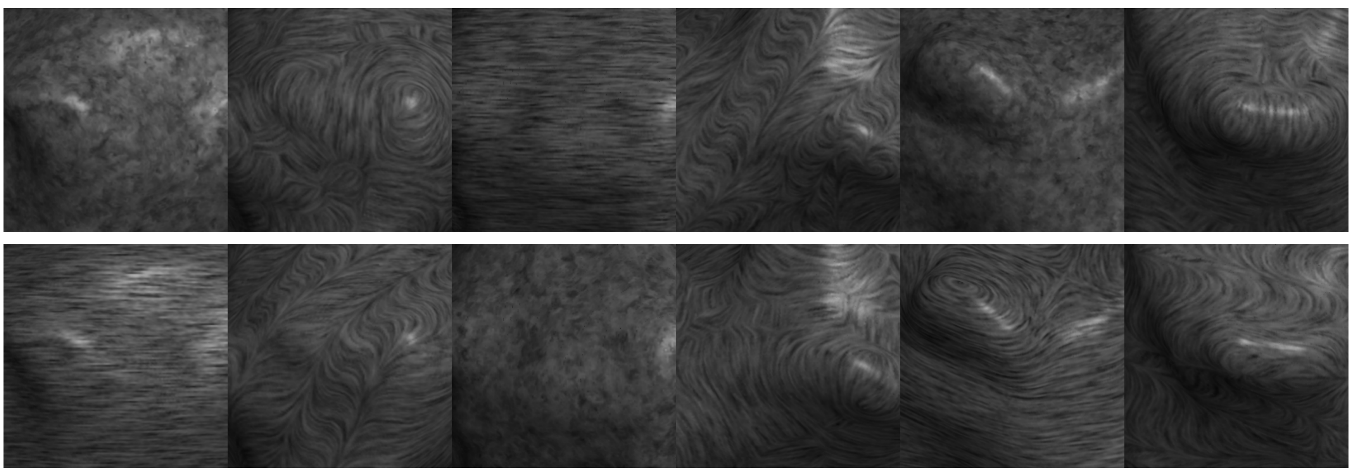


Figure 5: The set of stimuli seen by group B. First row: lot 1; second row: lot 2. The presentation order was randomly determined and was different for each subject.

2.3 Observers

We had five subjects participate in the experiments. All of the subjects were male EE and CS graduate students from the University of Minnesota, who agreed to participate as a favor to the second author and for compensation in the form of gift certificates to local coffee shops and/or eateries. All subjects were kept fairly naïve to the purposes of this experiment, though some of the subjects were certainly aware of the authors' previous work with principal direction texture. We informed the subjects that we were conducting experiments to evaluate peoples' ability to accurately perceive 3D shape in images but we specifically did not mention anything about texture. Our goal in doing this was to keep the subjects as free as possible of any potential biases and to avoid leading them into certain behaviors (such as lining up the direction of probe base elongation with the direction of the texture pattern) that they might not otherwise have considered. Before beginning the experiment, the subjects were asked to read a set of written instructions which described the probe positioning task. We used written rather than verbal instructions in an effort to maintain consistency. Subjects were also shown a single "training" image (figure 6) that portrayed ground truth answers in the form of correctly positioned probes for a seventh surface not included in the test data and rendered without texture. Note that several of these probes appear to point straight out of the screen. We showed them this image in order to give them an idea of what a set of exactly correctly positioned probes might look like. We were fairly selective in attempting to obtain participants that we hoped would be diligent in their efforts, and in the written instructions we stressed the importance of trying hard to do a consistently good job on all of the images, even if the shape was difficult to perceive. As an extra incentive, however, we told the subjects that after all results had been tallied, we would give a $20 bonus certificate to the student who gave the most accurate answers, overall.

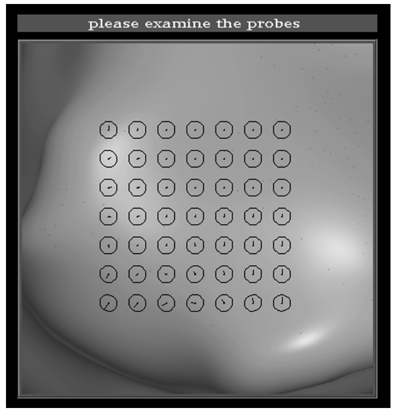
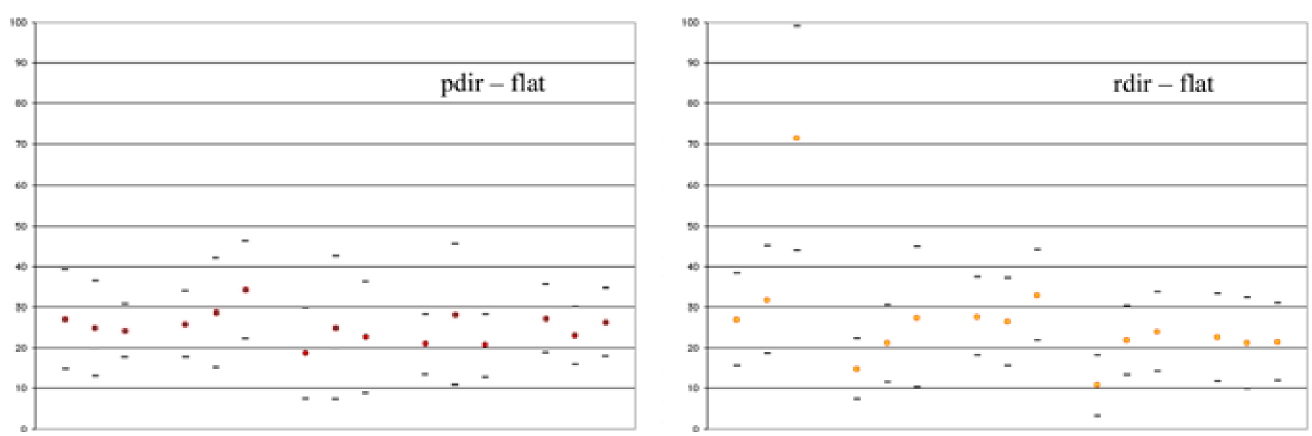


Figure 6: Training image, showing ground truth answers (correct probe orientations) at points across an untextured surface.

3 RESULTS

Having observed that other investigators studying shape perception using local probes analyze the perceived surface orientation in terms of slant and tilt, where slant is the angle of rotation out of the fronto-parallel plane, and the tilt is the angle of rotation about the viewing direction, we had initially hoped to be able to do the same in our studies – measuring the accuracy of observers' estimates of local heading in terms of the deviation in slant and tilt from the ground truth answers. While fairly satisfied with the indication of error provided by deviations in slant, we had several serious problems interpreting the magnitude of the errors due to incorrect estimates of tilt. The root of our difficulties was that too many of the points on our surfaces were too near to being parallel with the image plane. In numerous incidences the angular deviation in tilt was degenerate, because the estimated normal projected to a single point, and it was not clear how to appropriately handle these cases. We could not simply exclude these samples from our error calculations, because their occurrence was not uniform but tended to predominate in "bad texture" conditions, where the cues to shape were inadequate and subjects reverted to the default assumption that the surface lay in the plane of the image, or subjects simply gave up in frustration and left the probes untouched at their default original positions. Furthermore, even in the cases where the tilt angle was not degenerate, the lengths of the projected normal vectors could be exceedingly small, on the order of one or two pixels, and it was therefore possible to register huge estimated errors in the tilt component in places where the observer had merely misplaced the endpoint of the vector by two or three pixels (less than 1mm on the screen) in a particularly unfortunate direction. We therefore reluctantly decided to break with tradition and simply use as an error metric the angle in $\mathfrak{R}^3$ between the estimated normal direction specified by the probe and the true surface normal direction at the probe center. Figure 7 shows the mean angular error and standard deviations computed over the 49 probe positions at which estimates were made by each subject for each image, with each texture type, under conditions of binocular flat viewing. The results are grouped into different images by texture type, and then grouped within each image by test subject. Figure 8 shows the results under conditions of stereo viewing.



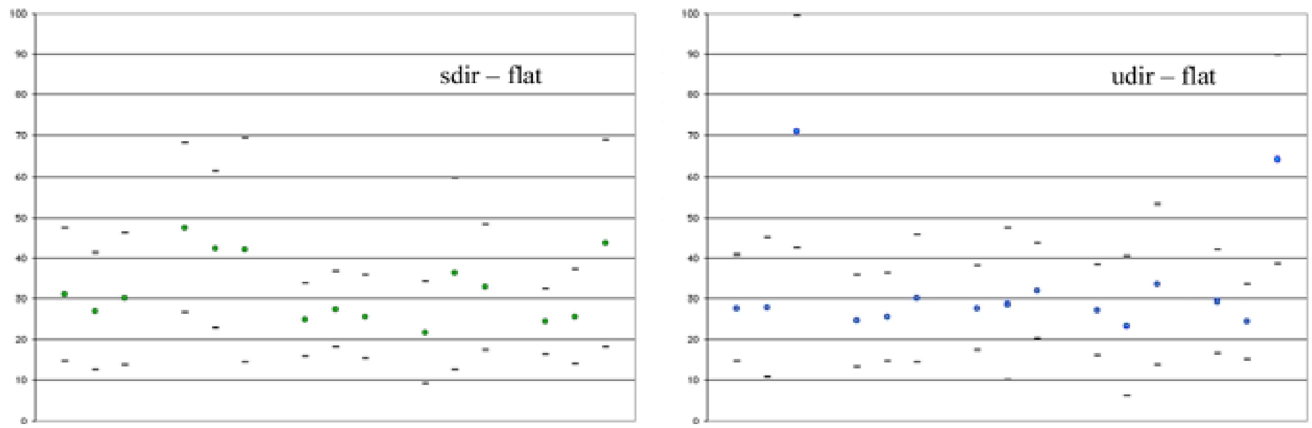


Figure 7: Individual results for the flat viewing condition. The height of each point represents mean angular error over the 49 probe locations per image. Subject number is the unspecified independent variable along the horizontal axis. Judgements from a single subject for different surfaces rendered with the same texture type are grouped by proximity along this direction. The textures are (clockwise from the top left): principal direction (pdir), isotropic (rdir), uniform (udir), and swirling (sdir).

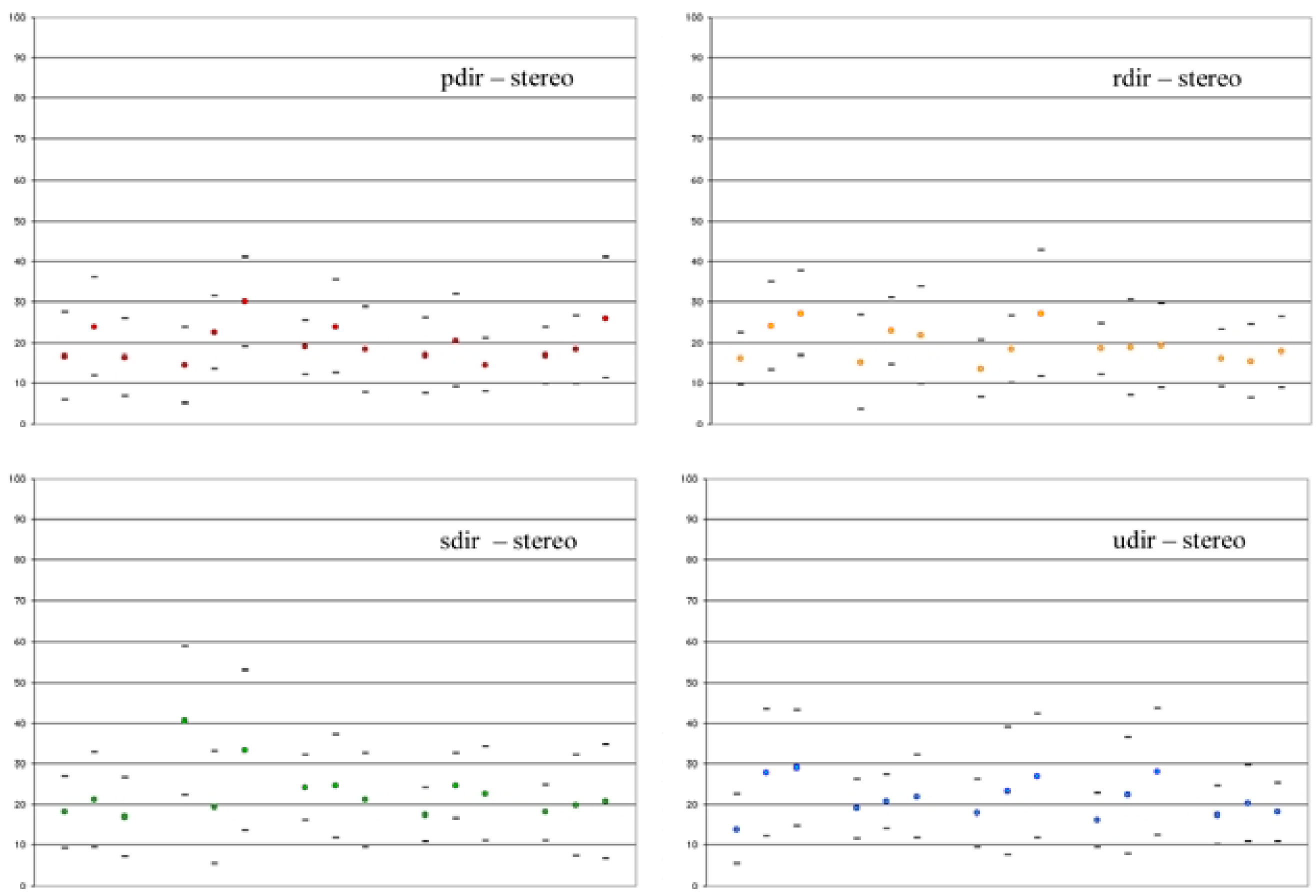


Figure 8: Individual results for the stereo viewing condition. Each point represents mean angular error over the 49 probe locations per image. Clockwise from top left: principal direction (pdir), isotropic (rdir), uniform (udir), swirling (sdir).

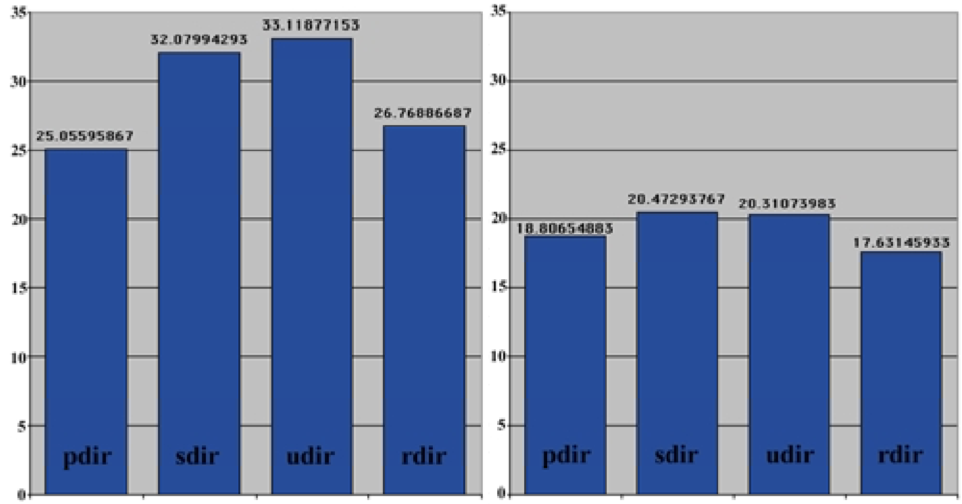


Figure 9: Pooled results (mean angle error) for all subjects, all surfaces, by texture type. Left: flat presentation; Right stereo.

4 DISCUSSION

A definitive statement of the results is hampered by the fact that we have not yet succeeded in doing a thorough statistical analysis of the data and hence cannot make any claims about the statistical significance of the differences in the mean angular errors observed under the different texture conditions. Overall, subjects seem to do somewhat better in the principal direction oriented and isotropic texture conditions than in either of the other two. It appears, from inspection of the individual results, that subjects may be less prone to making catastrophic errors when stimuli are viewed as flat images if the surfaces are rendered with the pdir texture. However, closer inspection of the pattern of errors is needed. A preliminary inspection suggests the presence of two different types of errors: coherent errors due to perceived depth inversion, and incoherent errors, as shown in figure 10. Errors appear to accumulate in the principal direction texture around discontinuities in the pattern where the first and second principal directions switch places. We had anticipated the possibility of an advantage in using an anisotropic texture in which the direction of the anisotropy followed lines of curvature over the surface, but this interpretation is not strongly supported by the experimental results. Most subjects appeared to perform equally well or better with the purely isotropic pattern. However some subjects were clearly misled in some places by the anisotropic patterns that followed directions different from the principal direction, suggesting that if one must use an anisotropic pattern, one must be careful about how it is applied over the object.

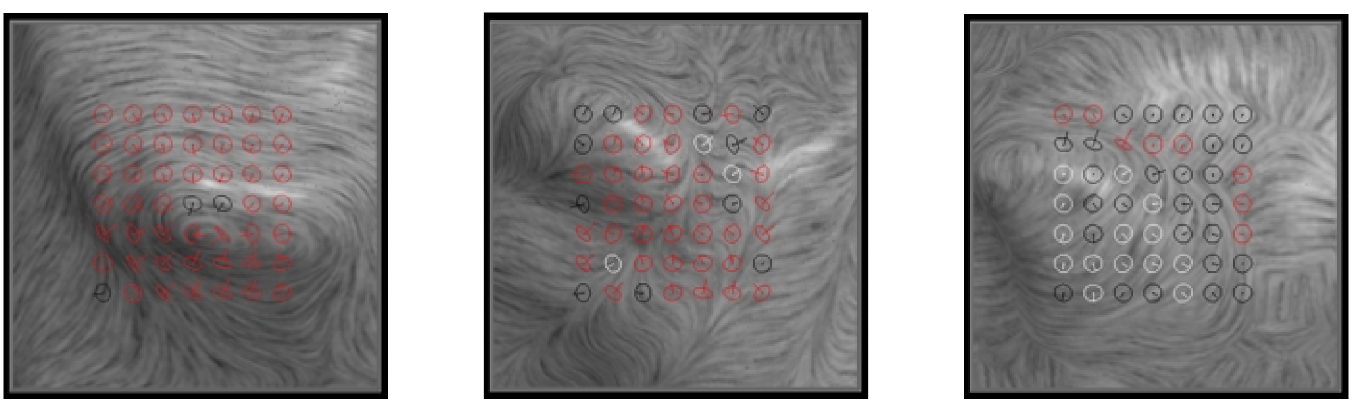


Figure 10: Some detailed individual results: Left: coherent errors due to depth inversion; Middle: incoherent errors apparently due to shape misperception; Right: errors tend to pile up at texture flow discontinuities, where the first and second principal directions switch places.

5 FUTURE WORK

There is considerable room for future work. One of the primary factors motivating this research was the desire to gain insight into how to select or define a texture pattern that could be used to facilitate the accurate and intuitive appreciation of 3D shape of a rendered surface. It appears clear that the principal direction textures defined above leave something to be desired in this respect. Shape representation from line orientation seems to be good in places where one of the two principal curvature values is high, but errors accumulate in the flatter areas where the directional information is less useful and less reliable. One direction for future work is to develop a more effective texture model that combines the strengths of several different texture definition approaches. A perhaps more immediate direction for future work is the investigation of the effect of texture orientation on surface shape judgments when the texture pattern is defined by surface relief rather than surface luminance. Does texture orientation affect shape perception in the same way in the two cases? Examples of some preliminary stimuli for subsequent experiments on this subject are shown in figure 11.

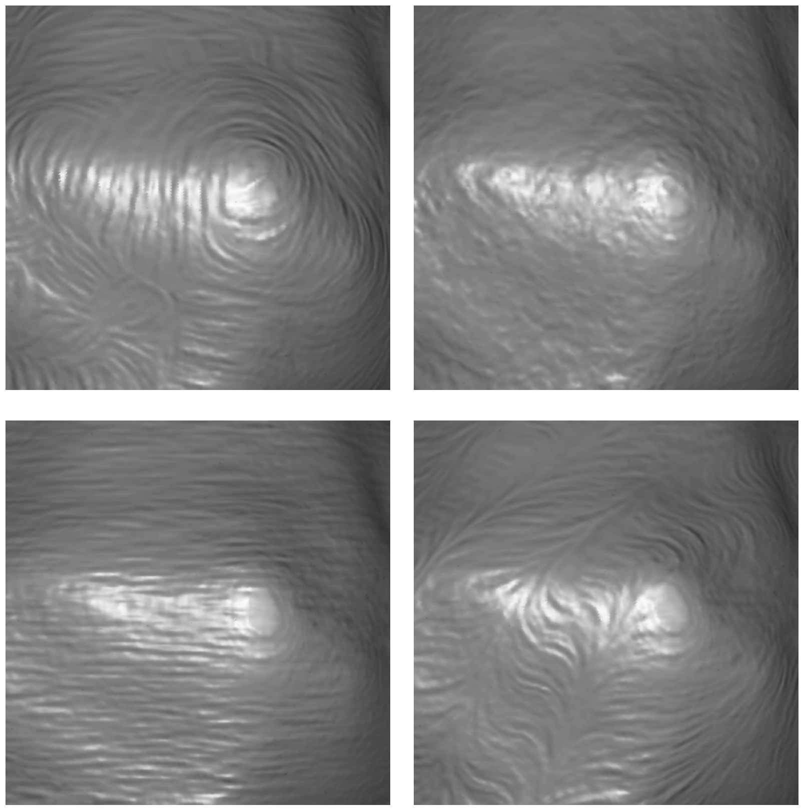


Figure 11: The same stimuli with the same textures, this time rendered as shaded relief rather than as luminance patterns.

6 ACKNOWLEDGMENTS

This work was partially supported by an NSF Presidential Early Career Award for Scientists and Engineers (PECASE). The images were rendered using parts of volume rendering software originally written by Marc Levoy. The dose data was provided by Dr. Julian Rosenman. Jeremy Leboy assisted with the implementation of the probe. We are grateful to our subjects TU, GG, MH, MM and HN for their dedicated efforts.

7 REFERENCES

Cumming, Bruce G., Elizabeth B. Johnston and Andrew J. Parker. (1993) Effects of Different Texture Cues on Curved Surfaces Viewed Stereoscopically, *Vision Research*, **33**(5/6): 827–838.

Ferwerda, James A., Sumanta N. Pattanaik, Peter Shirley and Donald P. Greenberg. (1997) A Model of Visual Masking for Computer Graphics, *ACM SIGGRAPH Proceedings*, pp.143–152.

Interrante, Victoria, Henry Fuchs and Stephen Pizer. (1997) Conveying the 3D Shape of Smoothly Curving Transparent Surfaces via Texture, *IEEE Computer Graphics and Applications*, **3**(2): 98–117.

Interrante, Victoria. (1997) Illustrating Smooth Surfaces in Volume Data via Principal Direction-Driven 3D Line Integral Convolution, *ACM SIGGRAPH Proceedings*, pp. 109–116.

Knill, David C. (1999) Contour into Texture: The Information Content of Surface Contours and Texture Flow, *www.cvs.rochester.edu/~knill/papers/postscript/texture_flow.pdf.*

Koenderink, Jan J. (1990) Solid Shape, MIT Press.

Koenderink, Jan J., Andrea van Doorn and Astrid M. L. Kappers. (1992) Surface Perception in Pictures, *Perception*, **52**, pp. 487–496.

Levoy, Marc. (1988) Display of Surfaces in Volume Data. *IEEE Computer Graphics and Applications*, **8**(3): 29–37.

Li, Andrea and Qasim Zaidi. (2000) Perception of Three-Dimensional Shape from Texture is Based on Patterns of Oriented Energy, *Vision Research*, **40**, pp. 217–242.

Lorensen, William E. and Harvey E. Cline (1987) Marching Cubes: A High Resolution 3D Surface Construction Algorithm, *ACM SIGGRAPH Proceedings*, pp.163–169.

Mamassian, Pascal and Michael P. Landy. (1998) Observer Biases in the 3D Interpretation of Line Drawings, *Vision Research*, **38**, pp. 2817–2832.

Stalling, Detlev and Hans-Christian Hege. (1995) Fast and Resolution Independent Line Integral Convolution. *ACM SIGGRAPH Proceedings*, pp. 249–256.

Stevens, Kent (1983) The Line of Curvature Constraint and the Interpretation of 3-D Shape from Parallel Surface Contours, *Proceedings of the International Joint Conference on Artificial Intelligence*, pp.1057–1061.

Todd, James T. and Francene D. Reichel. (1990) The Visual Perception of Smoothly Curved Surfaces from Double Projected Contour Patterns, *Journal of Experimental Psychology: Human Perception and Performance*, **16**(3): 665-674.

Conveying 3D Shape with Texture: Recent Advances and Experimental Findings

Victoria Interrante , Sunghee Kim and Haleh Hagh-Shenas
Department of Computer Science and Engineering, University of Minnesota

ABSTRACT

If we could design the perfect texture pattern to apply to any smooth surface in order to enable observers to more accurately perceive the surface's shape in a static monocular image taken from an arbitrary generic viewpoint under standard lighting conditions, what would the characteristics of that texture pattern be? In order to gain insight into this question, our group has developed an efficient algorithm for synthesizing a high resolution texture pattern, derived from a provided 2D sample, over an arbitrary doubly curved surface in such a way that the orientation of the texture is constrained to follow a specified underlying vector field over the surface, at a per-pixel level, without evidence of seams or projective distortion artifacts. In this paper, we report the findings of a recent experiment in which we attempt to use this new texture synthesis method to assess the shape information carrying capacity of two different types of directional texture patterns (unidirectional and bi-directional) under three different orientation conditions (following the first principal direction, following a constant uniform direction, or swirling sinusoidally in the surface). In a four alternative forced choice task, we asked participants to identify the quadrant in which two B-spline surfaces, illuminated from different random directions and simultaneously and persistently displayed, differed in their shapes. We found, after all subjects had gained sufficient training in the task, that accuracy increased fairly consistently with increasing magnitude of surface shape disparity, but that the characteristics of this increase differed under the different texture orientation conditions. Subjects were able to more reliably perceive smaller shape differences when the surfaces were textured with a pattern whose orientation followed one of the principal directions than when the surfaces were textured with a pattern that either gradually swirled in the surface or followed a constant uniform direction in the tangent plane regardless of the surface shape characteristics. These findings appear to support our hypothesis that anisotropic textures aligned with the first principal direction may facilitate shape perception, for a generic view, by making more, reliable information about the extent of the surface curvature explicitly available to the observer than would be available if the texture pattern were oriented in any other way.

Keywords: texture synthesis, shape representation, principal directions, shape perception.

1. INTRODUCTION

Numerous studies have shown that shape perception can be facilitated by the presence of surface texture[13,4], but the mechanisms of texture's effect on shape perception are not yet completely understood and the question of how best to design and apply a texture pattern to a surface in order to most effectively facilitate the accurate perception of its shape remains open. Recent findings support the idea that the facility with which we can accurately perceive surface shape in the presence of texture depends not only upon the intrinsic characteristics of the texture pattern itself but also upon how the pattern is laid down over the surface[12,8,11,9,6,10]. The results of studies that we conducted last year[6] using a restricted class of uni-directional texture patterns[5] appeared to support the hypothesis that accurate shape perception is most severely impeded by texture anisotropy when the flow of the texture pattern turns in the surface, and that shape perception accuracy is not significantly different in the case of a unidirectional texture pattern that is locally aligned with the first principal direction than in the case of an isotropic texture pattern of similar spatial frequency. However two important questions were raised by this earlier work.

First: why, if there is little ecological justification for a texture pattern being oriented in the principal directions across a doubly curved surface, does shape perception seem to be most accurate in the principal direction orientation condition? Is it because observers are biased to interpret surface markings as being aligned with the principal directions[12,11], or is it because principal direction oriented textures intrinsically carry more shape information by virtue of their tracing out

Email: {interran, skim, haleh}@cs.umn.edu; Web: http://www.cs.umn.edu/{~interran, ~skim}.

lines of maximum curvature over the surface – a 3D analogy of the effect found in 2D by Biederman[1]? (From a generic viewpoint, the contours traced by a principal direction texture have the greatest potential to reveal the surface curvature to a maximum extent; the contour traced out by the texture flow along any other direction at that point and for the same view will be intrinsically more flat, and this may represent a loss of shape information that is not recoverable.)

Second: with arbitrary curved surfaces there are two orthogonal directions in which the normal curvature generically assumes a non-zero extrema. Although these directions can be reliably classified into two types, the first principal direction and the second principal direction, it is not always clear which of these two directions a singly-oriented directional texture should follow. The first and second principal directions may "switch places" at many points, and it is not easy to reliably choose which direction to follow in order to minimize the apparent turning of the texture pattern in the surface. Might we be able to more effectively facilitate shape perception using an orthogonally bi-directional principal direction oriented pattern — one that has 90-degree rotational symmetry — rather than a uni-directional pattern that seems inevitably to exhibit artifacts at the umbilic points and parabolic lines where the first and second principal directions 'switch places'?

In order to answer these questions we undertook a new experiment, using a shape difference discriminability task and a new, more flexible shape-following texture synthesis method for the rendering of the surface stimuli.

1.1 Background and Previous Work

Because of the historical limitations of the capabilities of classical texture mapping software and algorithms, with few exceptions nearly all studies investigating the effect of surface texture on shape perception that have been conducted to date have been restricted either to the use of developable surfaces, which can be rolled out to lie flat on a plane, or to the use of procedurally defined solid texture patterns, whose characteristics are in general independent of the geometry of the surfaces to which they are applied. For several years we have believed that important new insights into texture's effect on shape perception might be gained through studies conducted under less restrictive surface and texture pattern conditions. Hence we have been working on the development of several different types of texture synthesis and rendering algorithms to enable the creation of the stimuli essential to such investigations.

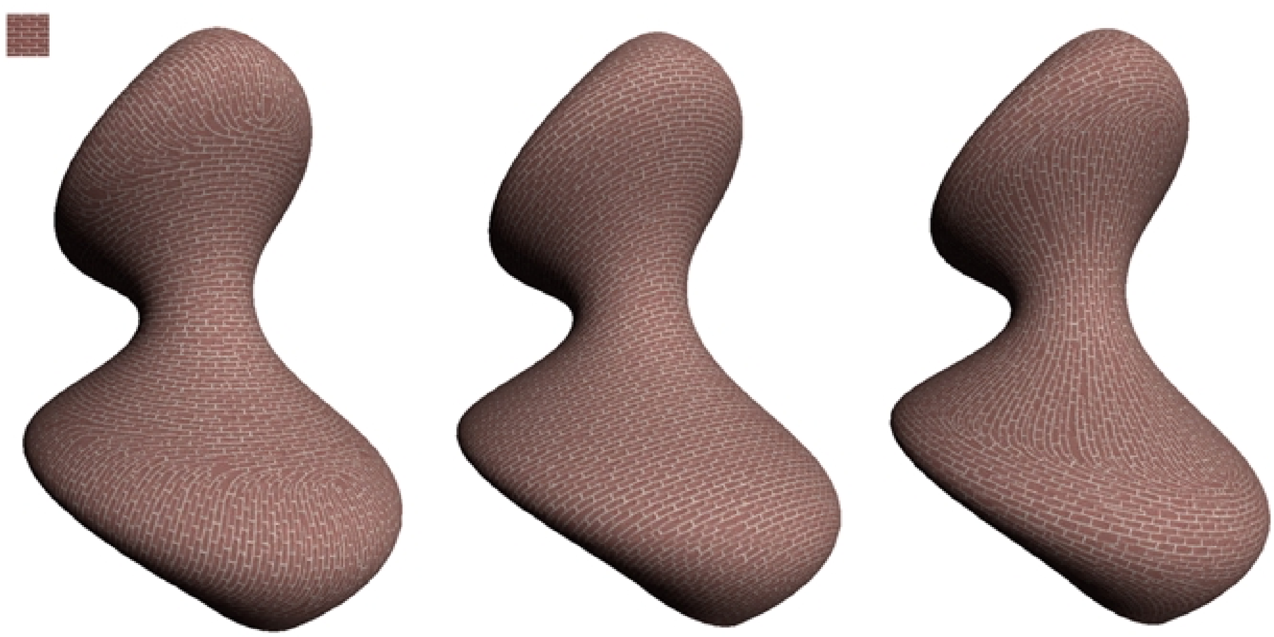


Figure 1: A brick texture pattern synthesized over a doubly curved surface according to three different texture orientation conditions. Left: rows of bricks follow the first principal direction; Center: rows of bricks are aligned in a globally uniform direction; Right: rows of bricks follow the second principal direction.

Recently, researchers in our group have developed a new algorithm capable of mapping the wide class of 2D texture patterns that can be described by Markov random processes onto arbitrary manifold surfaces, without visible seams or projective distortion, and in such a way that the dominant direction in the texture pattern is constrained to follow a specified vector field over the surface at a per-pixel level. With this system we now have a means to study, in an unprecedentedly well-controlled fashion, the effects on shape perception of multiple specific texture pattern characteristics, including but not limited to orientation. Figure 1 shows some illustrative results of this new texture synthesis method, in which it is possible to informally compare the shape representation efficacy of the first and second principal direction texture orientation schemes with the standard uniform direction approach.

The details of our texture synthesis method are described elsewhere[3] and will be only briefly summarized here. Basically the method is a two-step process in which the surface is first split into a collection of nearly planar patches, and then the texture pattern is synthesized over each patch using the boundary conditions supplied by neighboring patches to maintain the pattern continuity between the adjacent patches. Figure 2 shows several intermediate results of the texture synthesis process for one of the sample surfaces used in the experiment described in this paper.

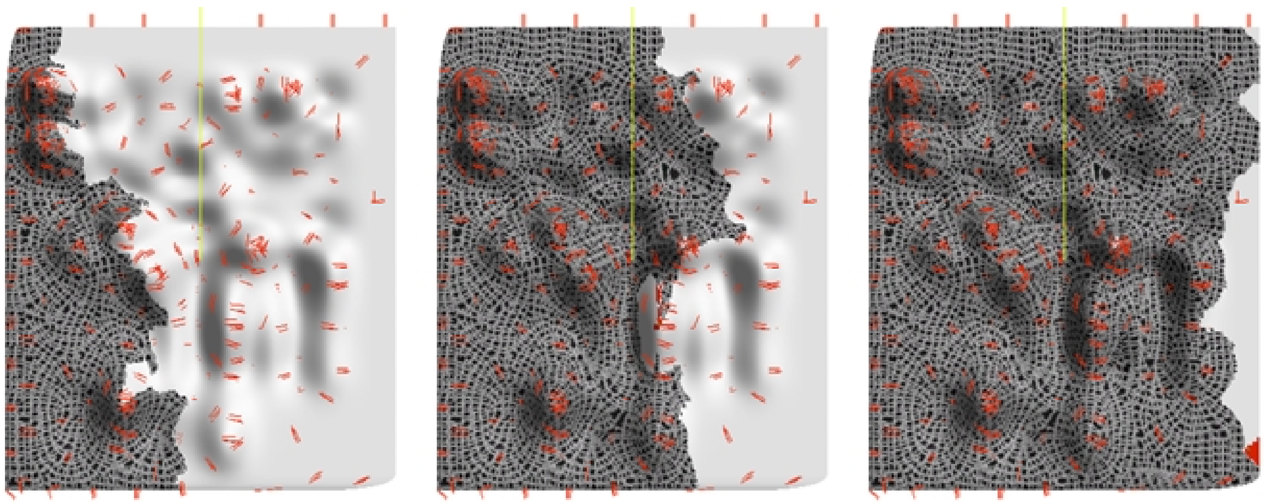


Figure 2: The process of texture synthesis illustrated on a sample surface (the red lines indicate the direction of the vector field at selected vertices).

2. METHODS

The two objectives of the experiment that we describe in this paper were: 1) to evaluate the validity of the hypothesis that shape perception accuracy declines in non-principal direction texture orientation conditions because less shape information is available to observers in those situations, and 2) to assess the advantages of using a bi-directional texture pattern rather than a unidirectional pattern in order to finesse the problem of choosing which of the two principal directions to align the texture pattern with at each point. We decided to use a four alternative forced choice surface shape discrimination task primarily for the first purpose, hypothesizing that if it is true that in the case of non-principal direction oriented patterns there is less information available to be used to make shape judgments, then we should find a higher threshold for the detection of changes in surface shape under these conditions due to the greater resulting ambiguity between multiple surfaces of different shapes which might all appear to be similar. In the following sections we provide the details of the experimental set up and design.

2.1 Stimuli

Beginning with a flat B-spline surface defined by a 16 x 16 grid of control points initialized to lie at uniform intervals in x and y across the z=0 plane, we defined an initial reference surface containing randomly dispersed hills and valleys using 100 repetitions of an iterative process in which we selected a random interior control point and displaced it by a constant amount, equivalent to $1/16^{th}$ of the width of the image, in either the +z or –z direction, alternately. Then, we partitioned the reference surface into 4 quadrants, noted the control points in each quadrant that defined either a hill or a valley, and then randomly selected one of these special control points in each quadrant to control the feature that would be changing over the course of the trials. For each selected feature we defined 8 different displacements of the shape-defining control point, in the +z direction for the bumps, and in the –z direction for the valleys, and then randomly selected from among these, pairs of displacements to define 7 distinct shape difference intervals, increasing in range from 1 unit of difference to 7 units of difference. Note that the perceptibility of a *k* unit 'shape difference' was thus equally likely to be tested with any pair of images from this set that were *k* units apart. Figure 3 shows the 8 different displacements used in quadrant 1. We were careful to compute the shading of each surface using a different random direction of illumination, selected from a solid angle of pre-determined valid illumination directions, in order to encourage our participants to attend to the 3D shape information separately conveyed in each image, and to prevent the shape difference discrimination task from degenerating into a simple 2D picture-difference discrimination task.

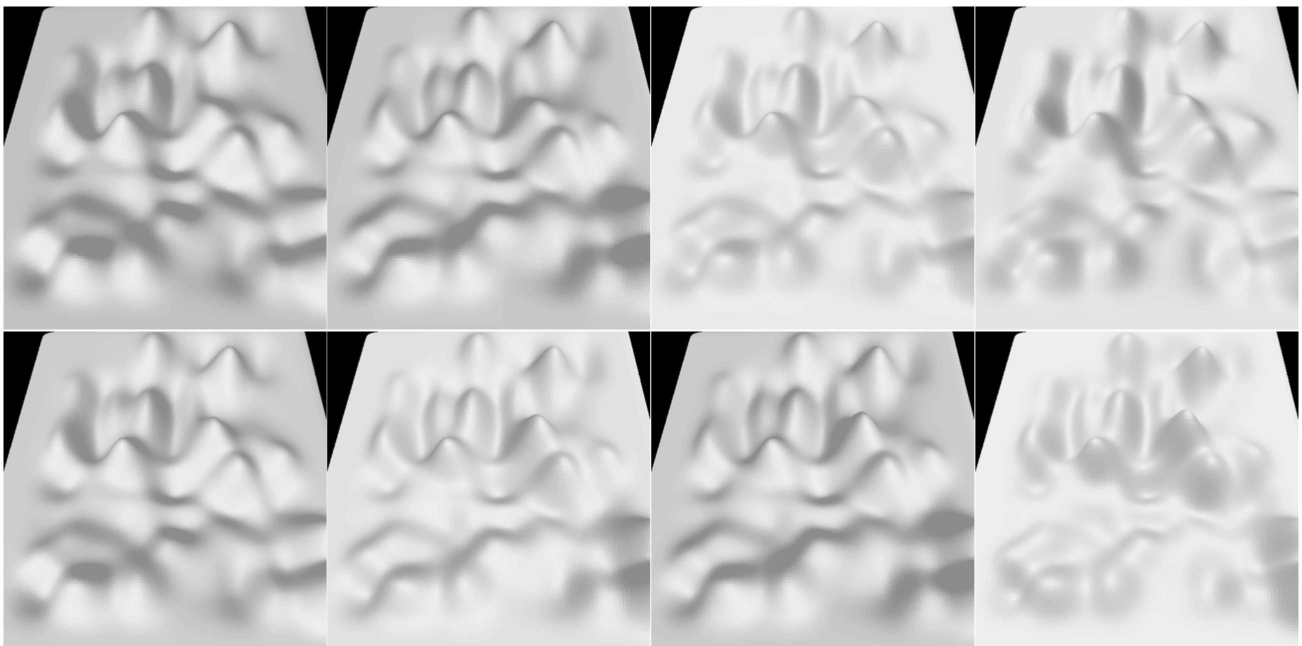


Figure 3: Shown from upper left to lower right are the eight different displacement surfaces used to represent shape changes in quadrant 1 (upper right portion of the surface). We used a different randomly selected illumination direction in each case in order to encourage our participants to focus on the 3D shape information and to minimize the usefulness of simple 2D pixel difference cues.

Once we had defined the 32 sample surfaces (4 quadrants x 8 levels of displacement on the selected feature in each), the next task was to define the three different vector fields that would control the orientation of the texture patterns over each surface. The three different texture orientation conditions that we wished to compare were: the principal direction condition, in which the texture was constrained to follow one or both of the principal directions at every point over the surface; the uniform direction condition, in which the texture was constrained to remain oriented in a constant uniform direction across the surface, and the swirly direction condition, in which the orientation of the texture pattern was allowed to twist and turn in a sinusoidal manner over the surface.

Having the parametric definition of the Bspline surfaces, we were able to compute the first and second principal directions analytically at every vertex in the surface mesh. We obtained the uniform direction vector field by calculating the line of intersection of the tangent plane at every vertex with the plane that passed through that vertex in an orientation parallel to the x=y direction. Finally, we obtained the swirly direction vector field by rotating each of the uniform direction vectors by a coherently varying amount that was defined as a continuous function of the 3D position of the vertex in world coordinate space.

The final step in the preparation of the stimuli was the surface texturing, for which we would use the new 'fitted texture' synthesis method developed in our lab. This method is capable of efficiently synthesizing unlimited quantities of a texture pattern that is perceptually equivalent to the pattern in a provided 2D sample, and doing so in such a way that the resulting texture can be applied nearly seamlessly over the surface without incurring projective distortion artifacts. We began with two base texture patterns selected from the Brodatz texture album[2]. As the first texture we chose a simple pattern, D49: Straw Screening, that was as plainly unidirectional as possible, and as the second texture we selected the pattern that seemed to use to be as similar looking to the first as possible while also being bi-directional, D20: French Canvas. We divided each of these patterns into eight subimages, in order to have the surface texture created from a different but similar subsample pattern for each of the different surface displacement intervals. It was necessary to take this precaution in order to avoid unwanted pixel-by-pixel texture similarities in areas of the surface that did not undergo a shape change. Figure 4 shows the same series of surfaces presented in figure 3, with the bi-directional texture pattern applied following the orthogonal principal directions. Figure 5 shows the results of using the three different texture orientation conditions (uniform, principal and swirly) over the same underlying surface.

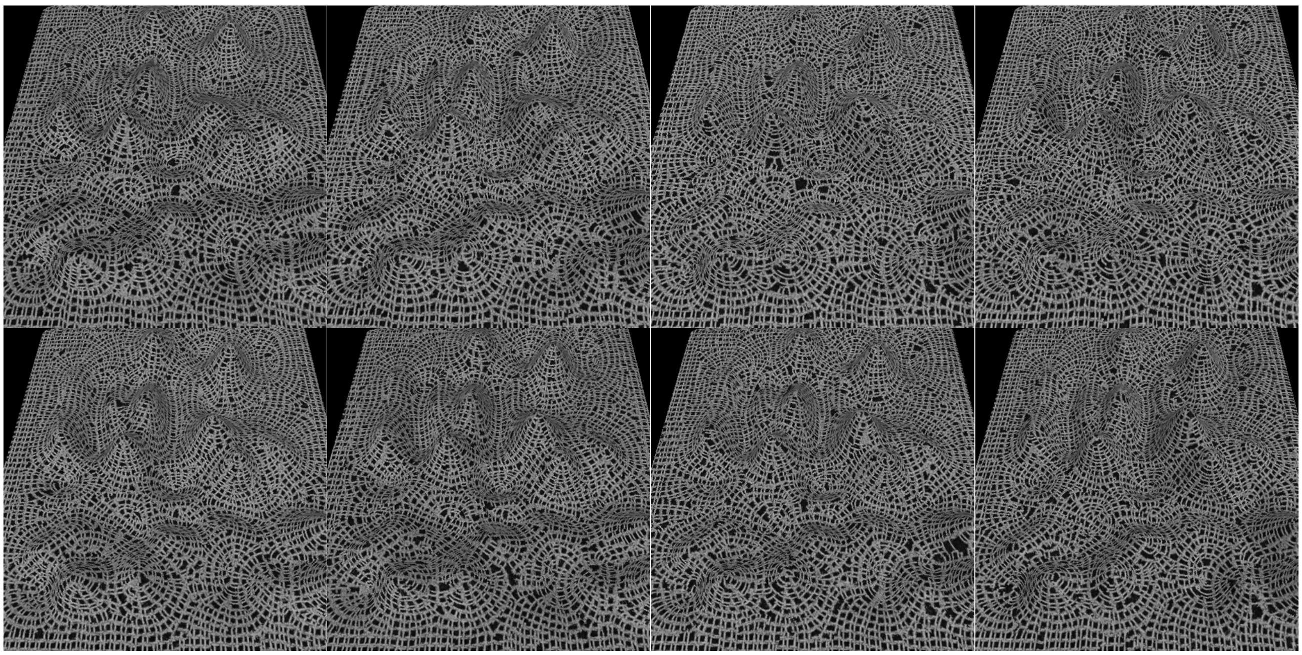


Figure 4: From upper left to lower right, the same eight different displacement surfaces, textured with principal direction oriented patterns derived from eight different subsample swatches from the same larger original texture pattern image.

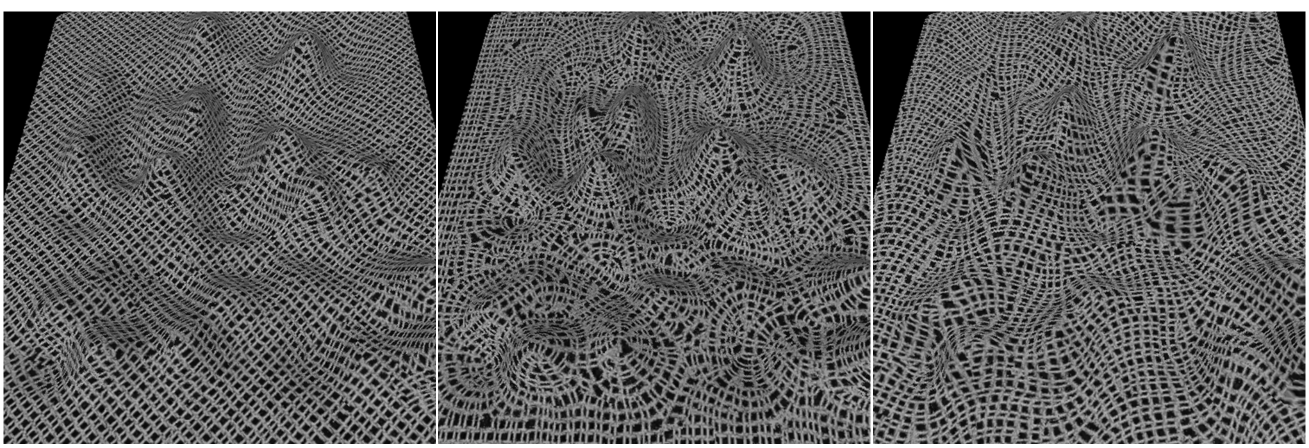


Figure 5: The three different texture orientation conditions. Left: uniform direction; Center: principal direction, Right: swirly direction.

Because we thought it was important to try to avoid the possibility of any unfortunate biases due to the choice of viewing angle, we chose to use two different viewing conditions for all surfaces: frontal and tilted. Examples of the frontal view and the unidirectional texture can be seen in the figures in the following section.

2.2 Task

Over the course of 672 trials (3 orientation conditions x 2 texture pattern conditions x 4 shape types/quadrants x 7 shape displacement amounts x 2 viewing conditions x 2 repeated measures), observers were shown pairs of images, side-by-side, and asked to specify in which quadrant the surface shapes appeared to be different. Viewing time was unrestricted. In analyzing the data, we found that subjects took, on average, between 14-25 seconds to make their decisions, spending a total of between 2.7 – 4.7 hours overall, including breaks. Figures 6 and 7 show the user interface for the experiment with representative examples of the remaining conditions. When the user clicks on the "toggle lines" button, the boundaries between the quadrants would be explicitly drawn over the image (the boundaries were defined in 3D). At the beginning of the experiment, the lines were turned on by default. Once the user elected to turn the lines off they would remaine off for subsequent trials unless the toggle lines button was pressed a second time to turn them on again. We have elected to show the interface in the lines-off condition here in order to simplify the presentation.

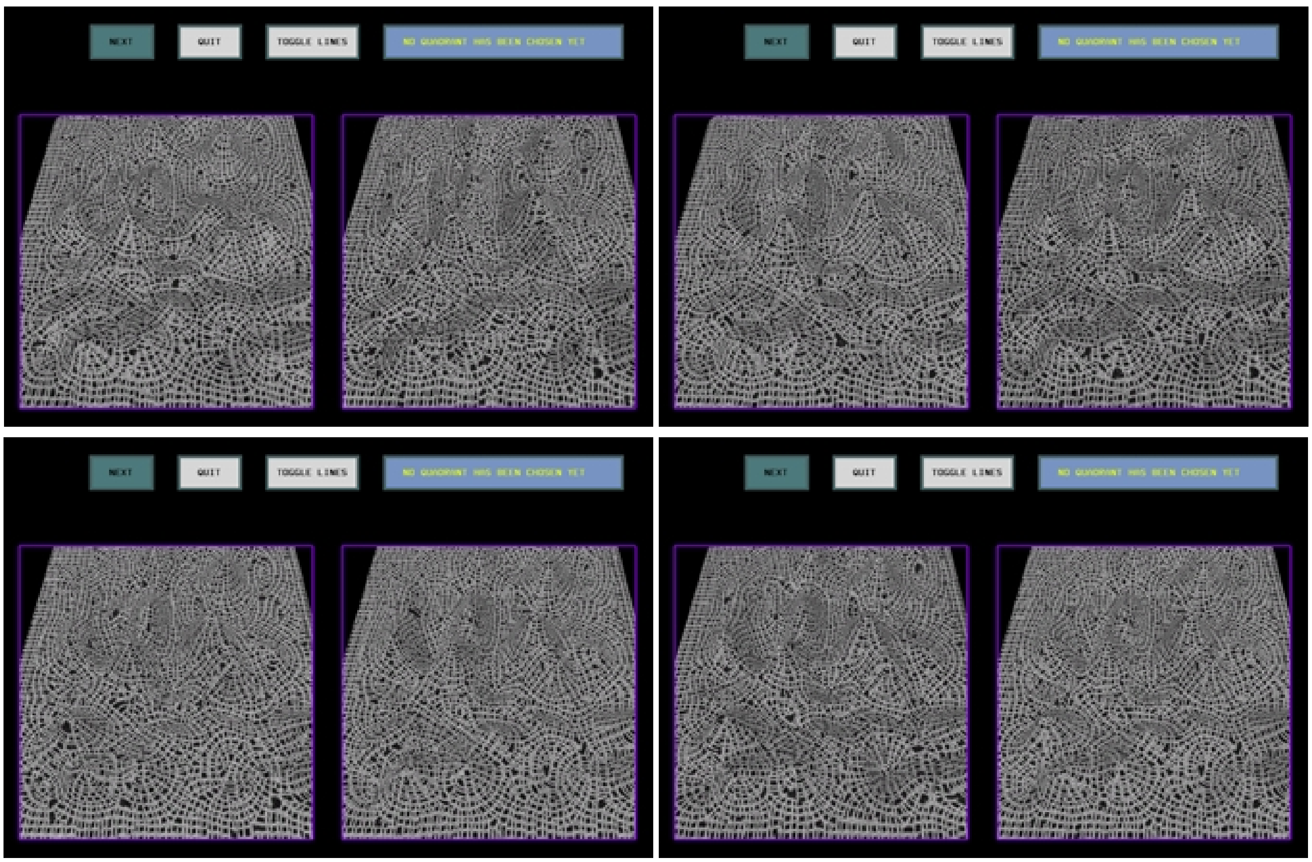


Figure 6: Four screen shots from the experiment, showing the four different shape difference types. The bi-directional pattern and the principal direction orientation condition were used in all of the cases shown here. Clockwise from the upper left, the surface shape differences appear in the following quadrants of each image respectively: upper right, upper left, lower right and lower left quadrant.

Because of the apparent simplicity of the task, we did not anticipate the need to have our subjects go through a training phase before beginning the experiment. In retrospect we discovered that this was a mistake, as we will describe in the following section.

We collected data from three subjects, who were chosen because of their availability and known reliability. One of the subjects was an author of this paper, who was intimately familiar with both the task and the experimental objectives but who was for obvious reasons not involved in the preparation of the sample surfaces used. The other two subjects were professionals from outside of our lab and outside of computer science, who were new to the task but informally aware of our basic research objectives.

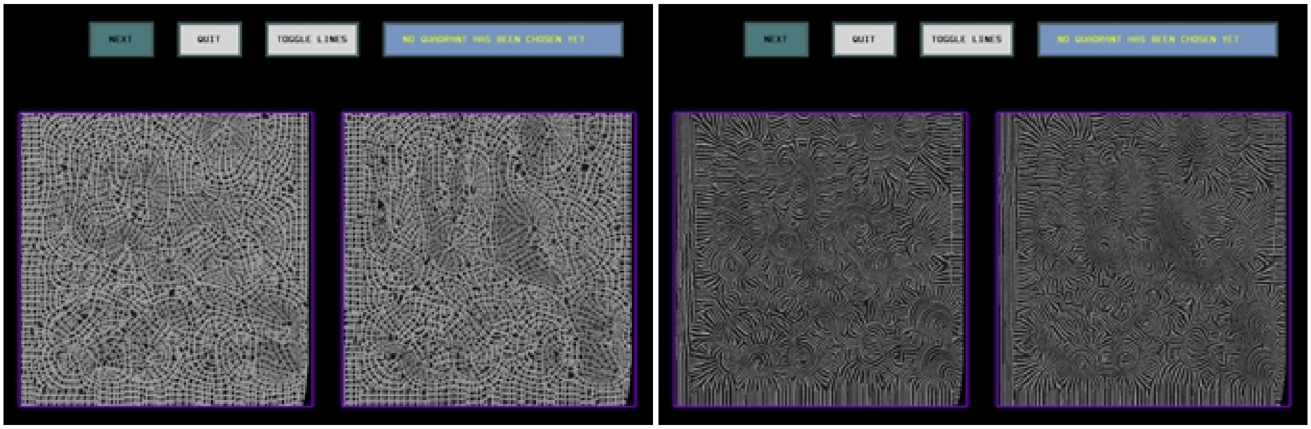


Figure 7: Examples of the two remaining conditions: the frontal view, and, on the right, the unidirectional texture pattern in the first principal direction orientation condition.

3. RESULTS

Figure 8 summarizes the overall results of our pilot experiment. Each data point in these images represents the percentage of correct quadrant selections that each user made over the 32 different trials corresponding to each shape difference level, for each texture orientation condition. (Two repeated measures were taken for each pair, and to make these graphs we combined the data across the two viewing conditions, the two texture pattern conditions, and the four quadrant/shape types.) We did not find it illuminating to consider each of the texture pattern conditions separately because the results turned out to be so similar in both cases. The pattern of responses seems similar across the three subjects, though the level of performance seems higher, overall, for the more experienced subject. Accuracy rates seem to rise as the disparity of the compared surface shapes increases, with a faster, earlier increase in the case of the principal direction textures, and a more linear but slower increase in the case of the swirly and the uniform direction textures.

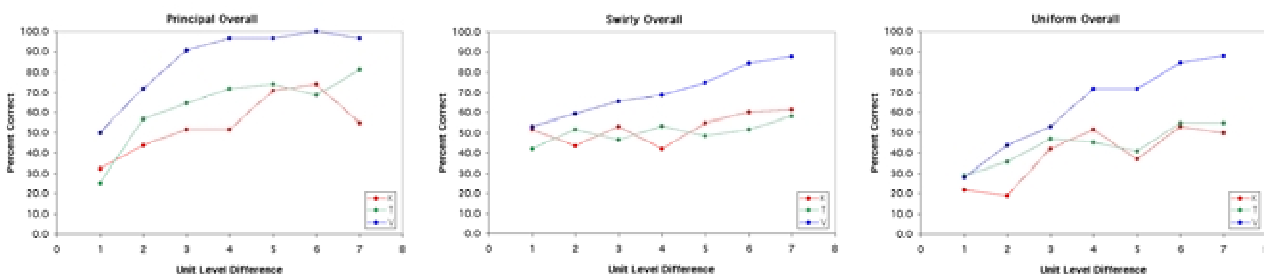


Figure 8: Overall results from our pilot experiment, per texture orientation condition. Each line represents a different subject.

Based on consistent reports from all three subjects that “the task got a lot easier once you figured out what to look for”, we decided to look separately at the first and second half of the data collected. This is of course not an ideal way to break down the analysis, but it proved to be an illuminating exercise. Figures 9-10 show the results tabulated separately for the first 336 responses and the second 336 responses per subject.

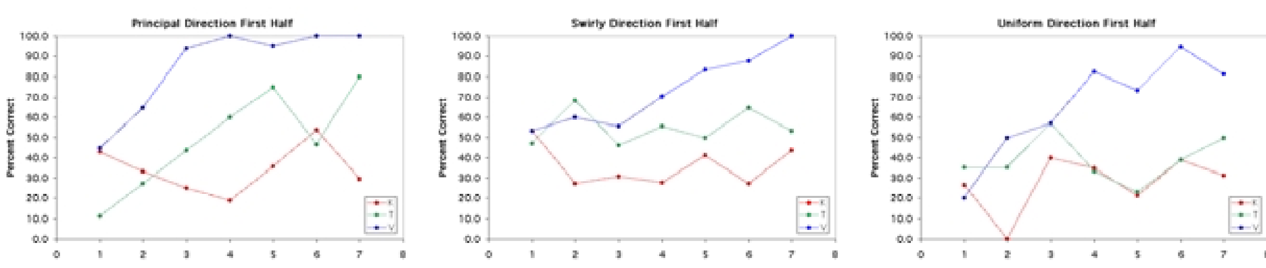


Figure 9: Results from the ‘first half’ only, per texture orientation condition.

Figure 10: Results from the 'second half' only, per texture orientation condition.

It is clear from the data in figures 9 and 10 that something very different is happening in the first and second halves of the experiment, at least in the case of the two observers less familiar with the task and the subject. Noting that the three observers' performance converges well in the second half, but not well at all in the first, we suspect that some considerable unreliability is being introduced into the results at the beginning of the experiment, most likely due to confusion on the part of the observers about how to interpret all of the different kinds of differences that appear between the two images whose shapes are supposed to be compared. Because only four different distinct shape features are actually changing, it's likely that all of the observers eventually figured out how to differentiate between the apparent image differences that were due to the vagaries of the texture synthesis process, which was working from a different sample image in each case, and the differences that were caused by subtle changes in the shapes of the underlying surfaces. After this point, the observers' task performance may have been more closely tied to the actual availability of shape information and less sensitive to distraction by irrelevant texture variations.

In figure 11, we combine the performance results across the displacement intervals to get a rough comparative overview of the overall task performance under the three different texture orientation conditions. We note that the experienced participant, V, seems to display fairly consistent performance across the two halves of the experiment, while naïve participant K seems to exhibit a tremendous improvement in performance from levels just above chance in the first half to levels equivalent to those achieved by V in the second half, across all three texture orientation conditions, and participant T seems to show a dramatic improvement in performance only for the principal direction textures, simultaneously accompanied by an apparent *decrease* in performance in the case of the swirly textures. Unfortunately, because so much varied information has been combined to obtain these numbers, it is difficult for us to be confident of the significance of these apparent differences.

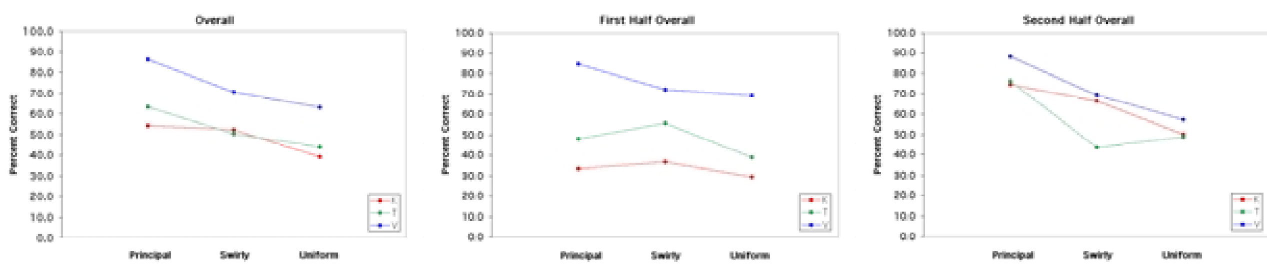


Figure 11: Combined results (averaged over all displacement ranges), per texture orientation condition.

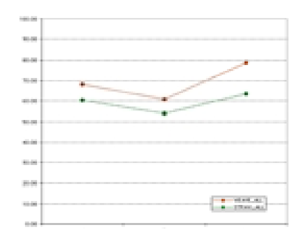


Figure 12: A comparison of performance under the two viewing conditions (left) and the two texture type conditions (right).

Figure 12 summarizes the overall differences, or lack thereof, in the cases of the two viewing conditions and the two texture type conditions. Performance appears to be slightly better, on average, for the tilted surface position compared to the forward-facing position, probably because it is easier to gauge the heights of the peaks under the tilted condition. Performance also appears slightly better for the bi-directional texture than the unidirectional texture, but not by much.

4. DISCUSSION

The results the we obtained in this experiment seem to lend support to the hypothesis that principal direction oriented texture patterns have a potential advantage over non principal direction oriented pattern in facilitating shape perception because of the fact that they provide more explicit evidence of the potential amount of surface curvature present than do directional patterns oriented over the surface in any other way. Disappointingly, but in hindsight perhaps predictably, the experiment did not provide us with much insight into the second of our objective queries – to assess the relative potential advantage for shape representation offered by a texture pattern than explicitly followed both the first and second principal directions simultaneously rather than following only one of these directions. We speak more about this in the future work section.

One question that is raised by this experiment is, why did we seem to find, in general, that task performance was worse in the uniform direction condition than in the swirly condition? Theoretically we would have expected that the in-surface undulations of the swirly texture would have made things worse for shape understanding. The answer, we believe lies in recognizing that task performance ultimately depended not on the accuracy of the shape understanding but only upon the discriminabilty of the presence of shape changes. All of the shape displacements were due to the translation of selected control points in a direction orthogonal to the original base plane. Because the elongated elements in the uniformly oriented textures were defined to lie in a constant direction parallel to the x=y plane, stretching the surface in the z direction had remarkably little effect on the appearance of the texture pattern – surfaces that were quite different in shape had uniform direction vector fields that were nearly indistinguishable when viewed from above. As we have earlier noted, this is an intrinsic problem with uniform direction textures. They do not depend much on the surface geometry, and hence there will inevitably be some aspects of a surface's shape that from some viewpoint such textures are not well-equipped to represent.

4.1 Limitations

In an attempt to minimize the possibility of inadvertently biasing the experiment through unconscious interference, we were careful to ensure that the entire surface definition process was done in a fully automated way without any manual intervention. Unfortunately we neglected to foresee the possibility that some shape differences might be in some extreme cases be partially hidden from view of the observer due to occlusion by other parts of the surface. As it turns out, in the many of the tilted views of our surfaces the bottom-most tip of the valley feature in quadrant 0 is not fully visible to the observer. We do not believe that our overall findings were biased by this occurrence, however we feel that for completeness this observation should be noted.

It is possible that our decision to randomly vary the illumination direction may have had some unintended and undesirable consequences. It is well-known that the accuracy of shape perception judgments vary under different illumination conditions. In particular, shape perception has been shown to be facilitated to a greater extent when the incident light strikes the surface at a grazing angle than when the light hits the surface head-on[7]. By varying the direction of illumination randomly for all surfaces, it is possible that we inadvertently made the shape discrimination task more or less difficult in some cases than in others, as a consequence of the illumination direction. Because of the high number of trials (225 judgments per texture condition), and the subtlety of the lighting effects on top of the texture, we believe that it is unlikely that the overall final findings have been significantly biased because of this issue. However it is worth considering whether a different strategy might be used, to avoid this problem and potentially obtain more reliable results.

Finally, a principal rule in good experimental design is that only one independent variable should change at a time. This means that if we are going to compare the discriminability of one unit of shape difference between two surfaces under three different texture conditions, we would like to be sure to have the only difference in these three cases be the mode of orientation of texture over the surface, with the random selection of lighting direction and random selection of difference interval endpoints being the same for all three texture conditions in that particular one unit shape difference case. Unfortunately that is not how it turned out that things were managed in this experiment. Too late, we discovered that the random selection feature for both lighting and interval endpoint choice was left on throughout all trials, resulting in the situation that the shape difference discrimination task was *not* performed under strictly identical conditions across all texture types. We have every reason to believe that the conditions were in all respects equivalent, and we do not believe that any systematic bias was introduced through these random variations, especially because of the large number of trials. However it will be important for us to re-validate our findings in the immediate future with a followup experiment in which these extraneous random variations are more carefully controlled for.

5. FUTURE WORK

While it is logical to assume that observers' ability to accurately perceive surface shape is poor when they are unable to accurately identify the quadrant of an image in which two presented surfaces differ in shape, the inverse assumption is not well founded – one cannot infer that in instances where observers are able to accurately identify the quadrant in which two presented surfaces differ in shape, that this is because they are in fact able to accurately perceive the two shapes. It is quite easy to imagine a scenario in which a subject has an invalid interpretation of the shapes of each of two surfaces, but can still discern that the two surfaces are not shaped the same. Hence, the conclusions that can be drawn from this study can be used only to inform our understanding of the *potential* of textures of various orientations to carry shape information. It is not surprising that the principal direction textures, because they are defined according to surface shape properties, would exhibit more prominent variation in response to shape changes than the other two types of textures, whose relationship to the surface shape is more indirect. Further investigations are needed to probe the extent to which shape understanding is facilitated under different texture type conditions.

6. ACKNOWLEDGMENTS

This work was partially supported by the National Science Foundation (NSF ACI-9875368). We are grateful to Paul Schrater and Cindee Madison for many helpful suggestions on the experimental design. The texture synthesis and image rendering was done using software originally written by Gabriele Gorla. Prototype surface modeling code was written by Krista Janssen, with support from the CRA-W Distributed Mentor Project. We are grateful to our observers TF and KB for their dedicated efforts.

7. REFERENCES

1. I. Biederman. "Human Image Understanding: Recent Research and a Theory", in Human and Machine Vision, Azriel Rosenfeld, ed., Academic Press, pp. 13-57, 1985.
2. P. Brodatz. *Textures: A Photographic Album for Artists and Designers*, Dover Publications, 1966.
3. G. Gorla, V. Interrante and G. Sapiro. "Texture Synthesis for 3D Shape Representation", *IEEE Transactions on Visualization and Computer Graphics, to appear.*
4. V. Interrante, H. Fuchs and S. Pizer. "Conveying the 3D Shape of Smoothly Curving Transparent Surfaces via Texture", *IEEE Computer Graphics and Applications*, **3**(2): 98–117, 1997.
5. V. Interrante. "Illustrating Smooth Surfaces in Volume Data via Principal Direction-Driven 3D Line Integral Convolution", *ACM SIGGRAPH Proceedings*, pp. 109–116, 1997
6. V. Interrante and S. Kim. "Investigating the Effect of Texture Orientation on Shape Perception", *Human Vision and Electronic Imaging VI*, SPIE **4299**, pp.. 330-339, 2001.
7. A. Johnston and P. J. Passmore. "Shape from Shading I: Surface Curvature and Orientation", *Perception*, **23**, pp. 169-189, 1994.
8. D. C. Knill. "Contour into Texture: The Information Content of Surface Contours and Texture Flow", *Journal of the Optical Society of America, A*, **18**(1), pp. 12-35, 2001.
9. A. Li and Q. Zaidi. "Perception of Three-Dimensional Shape from Texture is Based on Patterns of Oriented Energy", *Vision Research*, **40**, pp. 217–242, 2000.
10. A. Li and Q. Zaidi. "Information Limitations in Perception of Shape from Texture", *Vision Research*, **41**, pp. 1519-1534, 2001.
11. P. Mamassian and M. P. Landy. "Observer Biases in the 3D Interpretation of Line Drawings", *Vision Research*, **38**, pp. 2817–2832, 1998.
12. K. Stevens. "The Line of Curvature Constraint and the Interpretation of 3-D Shape from Parallel Surface Contours", *Proceedings of the International Joint Conference on Artificial Intelligence*, pp.1057–1061, 1983.
13. J. T. Todd, J. F. Norman, J. J. Koenderink and A. M. L. Kappers. "Effects of Texture, Illumination and Surface Reflectance on Stereoscopic Shape Perception", *Perception*, **26**(7), pp. 807-822, 1997.

Showing Shape with Texture – Two Directions are Better than One

Sunghee Kim, Haleh Hagh-Shenas and Victoria Interrante
University of Minnesota

ABSTRACT

If we could design the perfect texture pattern to apply to any smooth surface in order to enable observers to more accurately perceive the surface's shape, what would the characteristics of that texture pattern be? The answers to this question have important potential impact across a wide range of visualization applications, from molecular modeling to radiation therapy treatment planning, in which scientists need to attain an accurate, intuitive understanding of the shapes of complicated, smoothly curving surfaces in their data. Over the past several years, researchers in our lab have carried out a series of experiments intended to investigate the impact on shape perception of various characteristics of surface texture patterns. In this paper we report the results of our most recent study, in which we compare performance on a surface attitude probe adjustment task under three distinct conditions of principal direction pattern orientation and a control condition in which no texture was present. The three texture conditions were: a doubly-oriented texture in which approximately evenly-spaced lines follow both of the principal directions, a singly-oriented line texture which follows only the first principal direction, and a singly-oriented line integral convolution texture, from which information about texture compression in the direction of the texture flow may be indirectly accessible. Over a series of 200 trials (4 texture conditions x 10 surface/probe locations x 5 repeated measures), a total of five naïve participants were asked to adjust a circular probe, randomly located on an arbitrary doubly curved surface, so that its base appeared to lie in the displayed surface and its perpendicular extension appeared to be oriented in the direction of the surface normal. An analysis of the results showed that performance was best in the two-directional texture condition, closely followed by the line integral convolution condition. Performance was further decreased in the one-directional and no texture conditions (in that order).

The paper is organized as follows. In section 1 we describe the motivation for our work. In section 2 we very briefly discuss previous and related work in the field of vision research, and briefly recap the findings of our first two experiments [9, 10] which focused on the effects of pattern anisotropy and the impact of texture orientation. In section 3 we describe our experimental methods, including a brief summary of the process of the stimuli preparation, and we present a detailed statistical analysis of our experimental results. In section 4 we discuss the implications of our findings, and in section 5 we outline plans for future work.

CR categories: I.3.7, I.2.10, J.4, H.5.m
Keywords: Shape representation, shape perception, texture synthesis, texture mapping.

1 INTRODUCTION

As visualization designers, our goal is to determine how to most effectively portray a set of data in order that its essential features can be easily and accurately understood. When we choose to render a surface or object, we have tremendous latitude in choosing how we want to model its material properties. The most common practice is to use a simple Phong shading model without any surface texture, because it is easy to implement and is the default on most systems. However it is becoming increasingly clear that this model is not optimal for all purposes, and in particular is not optimal for shape representation. Unfortunately, the existing theories on shape perception do not provide sufficient guidance to definitively answer the question of how best to define the surface material properties of an object in order to best facilitate the accurate understanding of its shape. The overall objective of the study reported in this paper is to pursue investigations into the effects of texture pattern characteristics on surface shape perception that have the potential to yield fundamental theoretical insights into what works, what doesn't, and *why*.

2 PREVIOUS WORK

Although the mechanisms of texture's effect on shape perception are not yet completely understood, numerous studies over the years have found evidence that the accuracy of observers' judgments of surface orientation and curvature can be significantly affected by the presence of a surface texture pattern [c.f. 7, 20]. However, attempting to use texture to facilitate veridical shape perception can be a tricky business. We must avoid employing texture in a way that masks shape information [3] or that leads to an exaggerated or inaccurate perception of surface curvature or orientation, for example as occurs in op art.

Early research in the perception of shape, or surface orientation, from texture focused on the impact of pattern characteristics such as regularity or anisotropy, concluding that regularity can help [5, 4], but see [19] and that anisotropy can hurt [2, 21]. Recent findings support the idea that the facility with which we can accurately perceive surface shape in the presence of texture depends not only upon the intrinsic characteristics of the texture pattern itself but also upon how the pattern is laid down over the surface [18, 12, 16, 14, 9, 15].

Because of historical limitations of the capabilities of classical texture mapping software and algorithms, with few exceptions nearly all studies investigating the effect of surface texture on shape perception that have been conducted to date have been restricted either to the use of developable surfaces, which can be rolled out to lie flat on a plane, or to the use of procedurally defined solid texture patterns, whose characteristics are in general independent of the geometry of the surfaces to which they are applied. For several years we have believed that important new insights into texture's effect on shape perception might be gained through studies conducted under less restrictive surface and texture pattern conditions.

In the first of several recent of experiments designed to yield insights that might guide us in using texture effectively for shape representation [9], we examined the effect of the presence and direction of luminance texture pattern anisotropy on the accuracy of observers' judgments of 3D surface shape. Our stimuli consisted of complicated, smoothly curving level surfaces extracted from a volumetric dataset, across which we generated four different texture patterns via 3D line integral convolution. One of these patterns (rdir) was isotropic and the other three were anisotropic. In the 'udir' condition the texture flow followed a single uniform direction in the tangent plane to the surface, with the result that the pattern exhibited zero geodesic curvature. In the 'sdir' condition the texture flow followed a vector field that swirled over the surface (so that the pattern exhibited nonzero geodesic curvature)., and in the 'pdir' condition the texture flow followed the direction of greatest normal curvature (first principal direction) at every surface point. Figure 1 shows some representative stimuli from this experiment. We measured the accuracy of observers' shape perception judgements by having them manipulate an array of surface attitude probes [13] so that their circular bases appeared to lie in the tangent plane to the surface at the probe's center, and the perpendicular extensions appeared to point in the direction of the local surface normal. Stimuli were displayed as flat images in the first series of trials, and then the process was repeated with the same surfaces displayed in stereo. In the flat viewing condition, performance was significantly better in the cases of the pdir and rdir patterns than in the cases of the sdir and udir patterns. In the stereo viewing condition, accuracy increased for all texture types, but was still marginally greater in the cases of the isotropic and principal direction patterns than under the other anisotropic conditions. These results, shown in figure 2, are consistent with a hypothesis that texture pattern anisotropy can impede surface shape perception when the elongated markings are oriented in a way that is different from the principal direction.

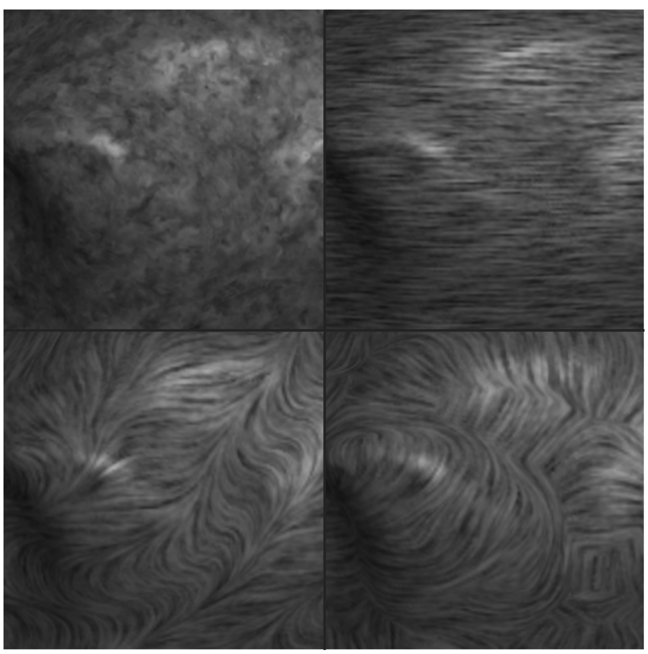


Figure 1: Representative examples of the sample stimuli used in one of our earlier experiments investigating the effect of texture orientation on the accuracy of observers' surface shape judgments. Clockwise from the upper left: Isotropic (rdir), Uniform (udir), Swirly (sdir), and Principal Direction (pdir).

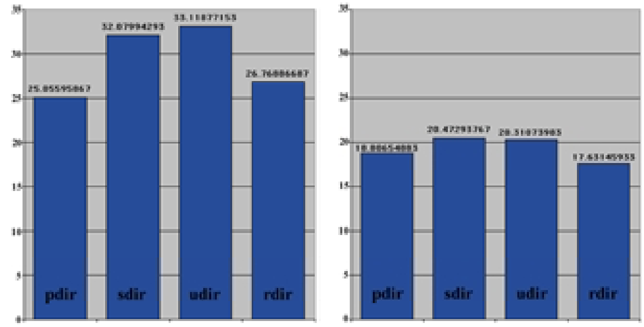


Figure 2: Pooled results (mean angle error) for all subjects, all surfaces, by texture type. Left: flat presentation; Right stereo. Differences between pdir and sdir, pdir and udir were statistically significant; differences between pdir and rdir, were not significant at the 0.05 level.

In an unpublished followup study, we repeated the experiment using displacement textures instead of luminance textures, and found the same pattern of results. However, two important questions were raised by this work.

First: why does shape perception seem to be most accurate in the principal direction orientation condition, when there is little ecological justification for a texture pattern being oriented in the principal directions across a doubly curved surface? Is it because observers are biased to interpret surface markings as being aligned with the principal directions [18, 16], or is it because principal direction oriented textures intrinsically carry more shape information by virtue of their tracing out lines of maximum curvature over the surface – a 3D analogy of the effect found in 2D by Biederman [1]? From a generic viewpoint, the contours traced by a principal direction texture have the greatest potential to reveal the surface curvature to a maximum extent; the contour traced out by the texture flow along any other direction at that point and for the same view will be intrinsically more flat, and this may represent a loss of shape information that is not recoverable.

Second: on arbitrary doubly curved surfaces there are two orthogonal directions in which the normal curvature generically assumes a non-zero extrema. Although these directions can be reliably classified into two types, the first principal direction and the second principal direction, there is not a clear algorithm for determining which of these two directions a singly-oriented directional texture should follow at any point, in order to minimize artifacts due to the apparent turning of the texture pattern in the surface. Is it possible that the effectiveness of the pdir textures used in this first experiment was compromised by these 'corner' artifacts, and that we might be able to more effectively facilitate shape perception using an orthogonally bi-directional principal direction oriented pattern — one that has 90-degree rotational symmetry?

Recently, researchers in our group developed a new algorithm capable of mapping the wide class of 2D texture patterns that can be described by Markov random processes onto arbitrary manifold surfaces, without visible seams or projective distortion, and in such a way that the dominant direction in the texture pattern is constrained to follow a specified vector field over the surface at a per-pixel level. With this system we now have a means to study, in an unprecedentedly well-controlled fashion, the effects on shape perception of multiple specific texture pattern characteristics, including but not limited to orientation.
In order to answer these questions we undertook a second experiment [10] that used the new texture synthesis method for

the rendering of the surface stimuli, and employed a shape difference discrimination task to evaluate the information carrying capacity of one single and one double oriented texture pattern under three different orientation conditions and two different viewing conditions each. In a four alternative forced choice task, we asked participants to identify the quadrant in which two simultaneously displayed B-spline surfaces, illuminated from different random directions, differed in their shapes. After all participants had gained sufficient training in the task, we found that accuracy increased fairly consistently with increasing magnitude of surface shape disparity, but the characteristics of this increase differed under the different texture orientation conditions. Participants were nearly consistently able to more reliably perceive smaller shape differences when the surfaces were textured with a pattern whose orientation followed one of the principal directions than when the surfaces were textured with a pattern that either gradually swirled in the surface or followed a constant uniform direction in the tangent plane. There were no apparent significant differences in the pattern of observer responses in the cases of the one-directional vs two-directional textures (performance was only marginally better overall in the 2-dir case), and no evidence of an interaction between texture and base surface orientation (tilted vs front-facing). These findings appeared to support our hypothesis that anisotropic textures aligned with the first principal direction may facilitate shape perception, for a generic view, by making more, reliable information about the extent of the surface curvature explicitly available to the observer than would be available if the texture pattern were oriented in any other way. However they did not yield much insight into the potential effects on shape perception of principal direction texture type.

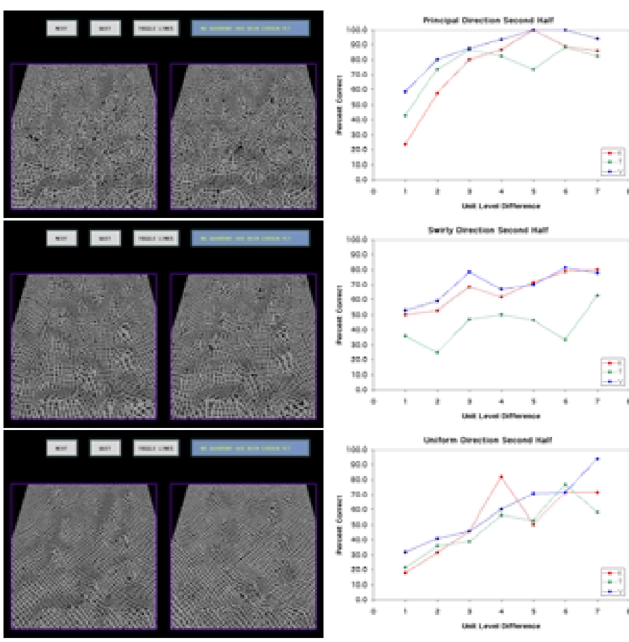


Figure 3: Representative stimuli used in an earlier experiment aimed at evaluating the relative extents to which differently oriented patterns have the potential to mask the perceptibility of subtle shape differences (left), and summary results (right). The texture conditions are, from top to bottom: principal direction, swirly direction, uniform direction.

3 CURRENT EXPERIMENT

Having put to rest the question of whether the orientation characteristics of an anisotropic texture pattern matter (they do), we were now freer to address the remaining important questions about how best to determine the characteristics of a subtle and aesthetic texture pattern that can be used to facilitate veridical shape perception without introducing unwanted visual noise. If we want to use a principal direction oriented pattern, what *kind* of principal direction texture is best? Does it matter? Now that we can apply any pattern we choose, how can we characterize what helps? In the current experiment we used a surface attitude probe adjustment task to evaluate the relative effectiveness of three different principal direction oriented texture patterns for facilitating accurate shape perception. For added insurance we decided to also include a control condition of no texture, because several of the participants in our most recent previous experiment had expressed a sentiment that “the texture seemed to be just getting in the way and making the task harder”.

3.1 Objectives

The immediate goal of the current experiment was to determine whether observers are able to make more accurate surface shape judgments under some principal direction texture conditions than under others. In order to make the problem tractable, we attempted to choose patterns that varied in only one or two important respects. Specifically we were interested in determining whether shape perception might be better facilitated in the condition of a texture that contains elongated elements that can be interpreted to follow both of the principal directions simultaneously than with a texture in which the elongated elements are oriented solely in one of the two principal directions. Additionally, we were interested in probing the potential effects of other texture pattern characteristics, besides orientation. For this reason we decided to also evaluate shape perception in the case of a singly-oriented line integral convolution-like pattern. For completeness it would have been nice to include a doubly-oriented lic-like texture as well, but we had some difficulties determining how to obtain such a pattern without destroying key aspects of the perceptual equivalence of the pattern to the singly oriented sample, so we ultimately decided to leave this matter for future consideration.

3.2 Method

3.2.1 Stimulus Preparation

The first step in the preparation of the experimental stimuli was to define the texture samples that would characterize the patterns that would be synthesized over the surface stimuli. Using Inklination’s Pen-and-Ink Crosshatching Filter plug-in for Adobe Photoshop, we created the two-directional and one-directional

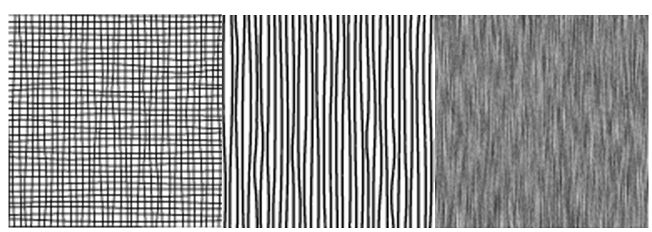


Figure 4: The sample texture patterns used in the study. From left to right: two-directional (2dir), one-directional (1dir) and lic-like (lic).

patterns shown in the left and center of figure 4 from the same uniform light grey base pattern. We obtained the lic-like pattern shown in the right of figure 4 by applying Photoshop’s built-in motion blur filter to an input image of high frequency random noise. The three patterns have nearly equal luminance means

(μ_{2D}=156.26, μ_{1D}=159.69, μ_{LIC}=127.04), but the line patterns have significantly different luminance histograms and standard deviations from the lic pattern (σ_{2D}=84.37, σ_{1D}=97.93, σ_{LIC}=21.10). All three patterns span an equivalent range of spatial frequencies, but the histograms of the amplitude spectra differ in significant respects.

The second step in the preparation of the experimental stimuli was to define the arbitrary smoothly curving surfaces that the participants would use in making their surface shape judgments. Figure 5 shows the textures on one of these test surface. Following the same procedures that we used in our previous study, we began with a flat Bspline surface defined by a 16 by 16 grid of control points distributed at uniform intervals in the x and y directions across the z=0 plane and over a series of 100 iterations we randomly chose a single interior control point to perturb by one unit (equivalent to $1/16^{th}$ of the width of the surface) in either the +z or -z direction. Having the parametric

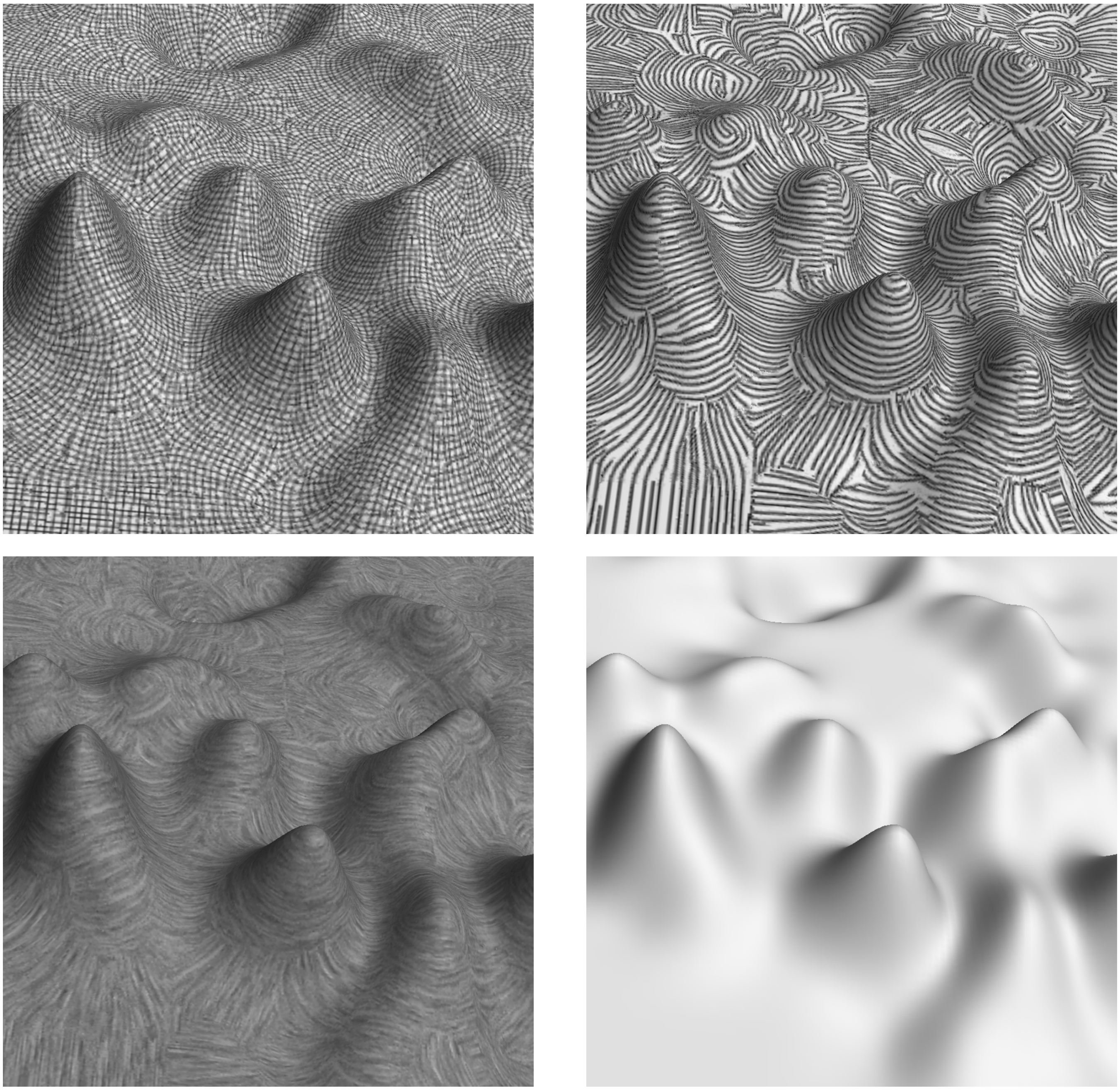


Figure 5: The four texture types used in our experiments. Upper left: Two-directional line pattern, following the first and second principal directions. Upper right: One-directional line pattern, following the first principal direction. Lower left: One-directional LIC pattern, following the first principal direction. Lower right: No texture (control condition).

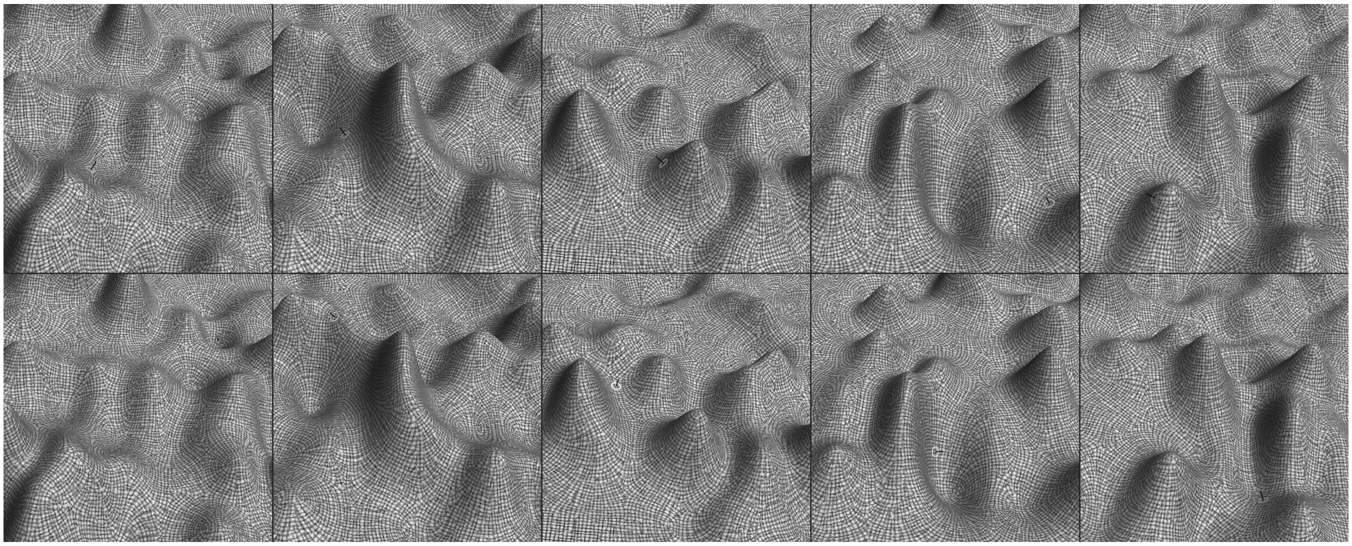


Figure 6: The five surfaces and ten probe positions used in our study. Een numbered probes 0-8 appear in the top row, with the odd numbered probes below them.

definition of the Bspline surfaces, we were able to compute the first and second principal directions analytically at every vertex in the surface mesh to use in the texture synthesis.

The final step in the preparation of the stimuli was the definition of the actual surface texture, for which we used the new "fitted texture" synthesis method developed in our lab. As the details of our texture synthesis method are described elsewhere [6] they will be only briefly summarized here. Basically the method is a two-step process in which the surface is first split into a collection of nearly planar patches, and then the texture pattern is synthesized over each patch using the boundary conditions supplied by neighboring patches to maintain the pattern continuity across the surface. As previously mentioned, this method is capable of efficiently synthesizing unlimited quantities of a texture pattern that is perceptually equivalent to the pattern in a provided 2D sample, and does so in such a way that the resulting texture can be applied nearly seamlessly over the surface without incurring projective distortion artifacts. We synthesized each of the three test patterns over each of the five test surfaces shown in figure 6.

3.2.2 Experimental Setup and Task Description

After defining the surfaces, the next step was to select the locations of the surface attitude probes. For consistency between observers we pre-determined a fixed set of ten locations, two on each surface, at which the users would make surface orientation judgments. Although we were tempted to choose the locations by hand, to ensure that the probes were located in "good" areas, we realized that it was in fact essential to the integrity of the experiment that the probe locations be determined completely randomly, in order to avoid inadvertently biasing the results through an unconscious preferential selection of positions at which the shape appeared "well-behaved" or comprehensible. However we did have to go in manually to reject probe positions that were not visible from the predetermined viewpoint, and probe positions at which the default initial probe orientation was within 10 degrees of the true surface normal direction, in which case participants would be able to get "good" results without performing any task. We limited the study to ten probe positions because we would be asking observers to make 5 repeated measures at each probe, under each of 4 texture condition, and to control for fatigue-related factors we did not want the sessions to exceed two hours in length.

Stimuli were displayed in a 900x900 pixel window on a 21" Sony Trinitron Multiscan E500 monitor located in our shared computer graphics lab and freely viewed from an approximate distance of 24". Both the surface and the probe were modeled in 3D and displayed in perspective projection using an OpenGL based renderer, and shaded using a standard Phong illumination model. The viewing angle and lighting parameters were held constant over all trials. Observers could freely rotate the probe in 3D by clicking and dragging the mouse in a way that simulated the effect of pulling on the probe handle. The interaction was structured so that the probe did not snap to the mouse location, so that observers were not required to position the cursor over the probe in order to manipulate it. Providing the opportunity for unobstructed viewing of the probe during the entire manipulation procedure was useful for ease of task performance. As the probe was repositioned, the image was continuously refreshed, with depth buffering turned off after the drawing of the surface and before the drawing of the probe in order to eliminate occlusion cues to the proper positioning of the probe base.

The procedure that we used to determine the presentation order of the 200 trials was refined through a small pilot study involving two of the authors of this paper. We determined that because of the greater ambiguity of the local surface orientation in the untextured condition, and the strong incentive to carry over inferred surface orientation information gleaned from textured trials, it would be necessary to have participants fully complete the portion of the experiment involving the untextured trials before proceeding to any trials in which the surface was textured. In addition, to avoid the potential bunching up of presentations involving repeated measures at any individual probe location, we used the following method to determine the trial order. First, randomly permute the array of 10 probe ids. For each probe id, randomly choose one of the three texture conditions. Do this two more times, being careful not to choose any particular probe/texture combination twice, and to avoid the immediate repetition of any probe location between adjacent sets. Then repeat the entire process four more times.

Before starting the experiment, participants were given an instruction sheet explaining what the experiment was about and what their task was. We provided written instructions in order to minimize the chances of inadvertently our coaching different participants in different ways. The five participants were students known to the authors who agreed to come to campus over spring break to assist with the study. They were each compensated with a $20 gift certificate for their efforts. In order to encourage extra diligence in the task performance, we told participants that an addition $20 gift certificate would be awarded to the student with the best performance overall.

3.2.3 Training

In order to ensure that participants had an adequate understanding of the task, and that all participants could successfully complete the task with a minimum level of competence, we required each participant to complete a training session immediately before the experiment. We generated a sixth surface for the training, which we textured using an isotropic random noise texture that was close to the test patterns in mean luminance and spatial frequency. As in the actual study, a probe was superimposed on top of the textured surface in each trial. However only one surface was used, with 15 different probe locations individually presented. Subjects were asked to manipulate the probe until they were satisfied that the probe's perpendicular extension has the same direction as the normal to the surface at that point. After pushing the "NEXT" button, if the probe orientation selected by the user was within 10 degree of the true surface normal orientation, they would automatically proceed to the next trial. Otherwise, the probe would be color-coded based on the magnitude of the error, measured as the three dimensional angle between the true normal and the user selected probe normal. At this point, the user would be able to continue with the probe manipulation until they had determined an adequately accurate position for the probe, assisted by the ability to use the color as a cue to correct their estimates dynamically. In order to prevent users from relying 100% on the color-coding without actually trying to understand the shape of the surface, we required that each subject pass three trials out of the 15 without using the color-coding cue. Otherwise after the end of 15 trials, the training restarts from the first image until the user can meet the condition. Four out of our five subjects (plus the authors of the papers) passed the pilot study in the first run. However, one of our subjects, who ended up achieving the best results among the group in the actual experiment, went through three sets of 15 trials before she could pass and move on to the actual study. Each set of fifteen trials took on average about 12 minutes. Figure 7 shows an example of the training data.

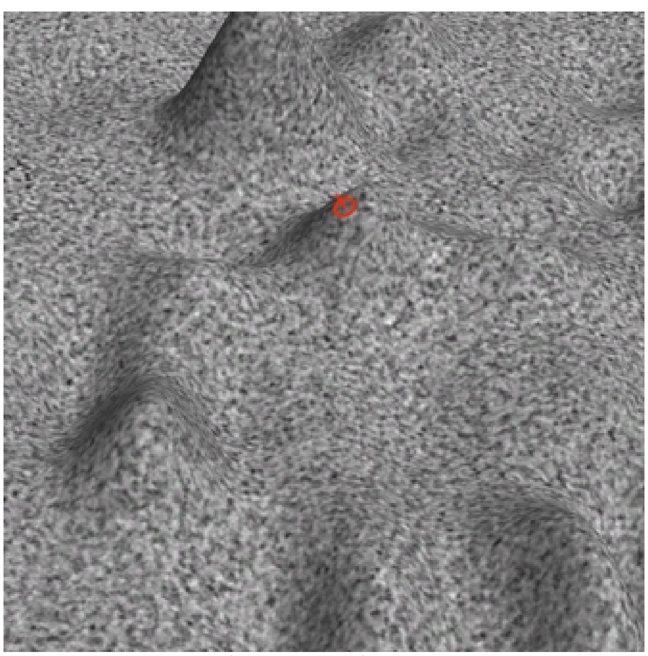


Figure 7: The training surface with one of the 15 test probes, shown in an orientation that is within 10-15 degrees of the true position.

3.3 Results

Figure 8 shows an overall summary of the results that we found in this experiment. Both the mean and median angle errors, across all observers and all probe locations, followed this pattern. Performance was best in the case of the two-directional pattern, closely followed by the lic-like pattern, and then the one-directional pattern. As expected, performance was worst in the no texture condition. We used the statistical software package 'MacAnova', developed by Prof. Gary Oehlert from the Department of Statistics at the University of Minnesota, to perform a three-way, within subjects mixed analysis of variance (ANOVA) to evaluate the statistical significance of the results. We found significant main effects of probe location ($p=0.0000264$) and texture type ($p=0.0002843$), and a significant two-way interaction between texture type and probe location ($p<0.00000001$). We did not find a significant main effect of subject id ($p = 0.18$) nor of a significant interaction between subject and texture type ($p = 0.62$). We used Tukey's HSD ("Honestly Significant Difference") method to perform post-hoc pairwise comparisons of the means of the angle errors under the different texture conditions. We found that the following differences were statistically significant at the 0.01 level: 2-dir < 1-dir, 2-dir < None, 1-dir < None, and LIC < None. The difference between performance in the 2-dir and LIC conditions was not statistically significant at the 0.01 level, nor was the performance difference between the LIC and 1-dir conditions, at this level.

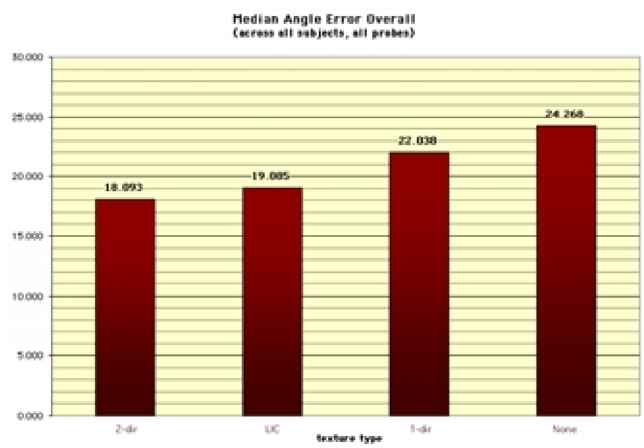


Figure 8: Median angle errors in the different texture conditions, over all subjects and all probe locations.

From the charts in figure 9 is it possible to gain some deeper insight into the nature of the interaction between probe location and texture type. The first graph shows the median angle errors across all subjects, broken down by probe location. The next 5 graphs show the mean angle errors and standard deviations across the 5 repeated measures for each subject individually, again broken down by probe location.

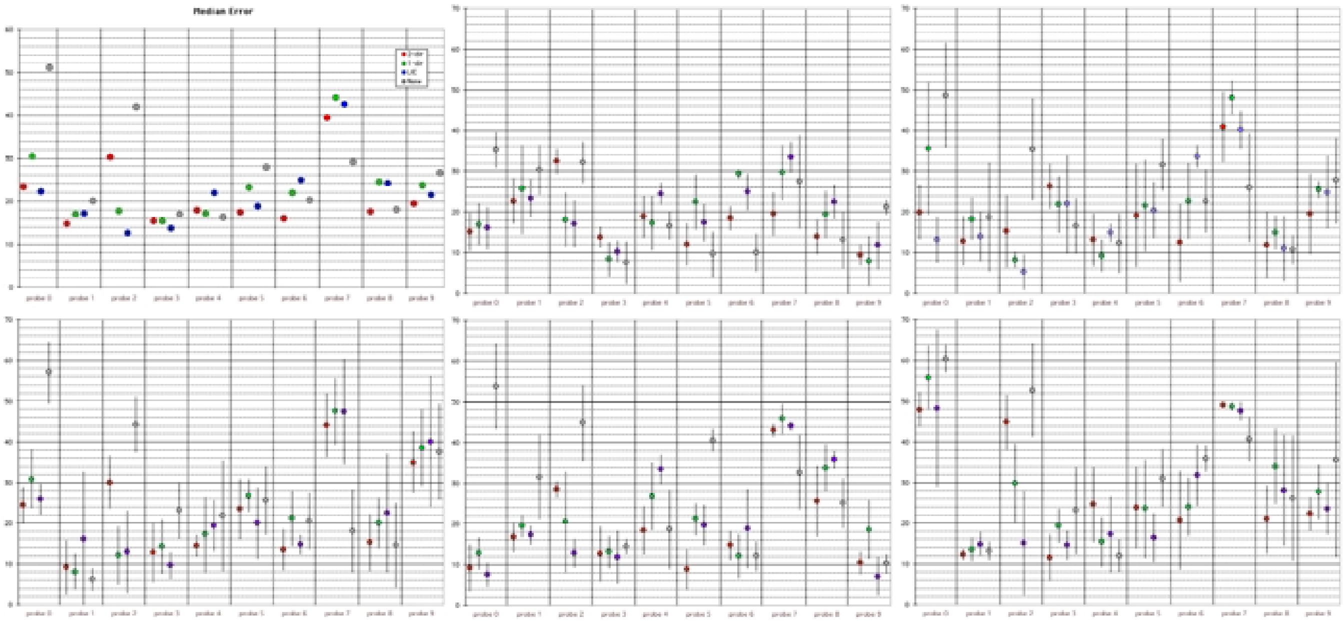


Figure 9: Charts illustrating the details of the experimental results. In the upper left, a graph of the median angle error across all subjects under each texture condition, broken down by probe location. Following that are five graphs showing the mean angle errors and standard deviations for each subject individually, again broken down by probe location. Remarkable consistency can be seen in the pattern of performance by texture type, across subjects, at the same probe locations.

4 DISCUSSION

It appears clear from the results of this experiment that there are small but significant differences in the extent to which various different principal direction-oriented patterns can facilitate accurate shape perception.

There are several possible explanations for these results. It could be that performance is better in the case of the patterns that carry information about distance along the principal direction than in the case of patterns that only indicate the direction itself. Or it could be that performance is worse in the case of the high contrast, high regularity pattern whose statistics least resemble the statistics of patterns found in nature [17]. Despite our concerted efforts to maintain a basic equivalence among the three texture patterns used in this study, it is clear that many differences among the three patterns persist, including differences in spatial frequency and contrast, and differences in higher order statistical characteristics of the patterns. Hence it is possible that the observed performance differences might be alternatively explained by one of these uncontrolled-for differences, rather than or in addition to the effects due to representing surface orientation information in two rather than only one direction. For example, there could be an interaction between the spatial frequency of the texture pattern and the resolution accuracy with which surface attitude adjustments can be made, or there could be some interaction between contrast and the availability of shape-from-shading information.

5 FUTURE WORK

In going through the study ourselves during the pilot phase of the current experiment, we realized that the amount of global vs local information that we were using to make our probe adjustment decisions was varying between different texture type conditions and different probe locations. As an addendum to our current study we plan over the next several months to conduct followup experiments in which we explicitly control the size of the visible area surrounding a probe, and investigate potential interactions between texture type and task performance as a function of the this neighborhood size[1].

Now that we have a tool for applying any arbitrary pattern to any arbitrary surface at a high resolution while controlling the pattern orientation at a per-pixel level, we have the potential to pursue investigations of the effects of a wide variety of texture pattern characteristics on shape perception in the more complicated case of doubly curved surfaces.

In particular, we have plans to explicitly investigate the impact on shape perception of variations in the contrast and spatial frequency characteristics of a single base pattern (probably LIC, because it is easiest to control at a fine-tuned level). Graphic designers have always been sensitive to the fact that certain patterns are "hard on the eyes" or "annoying to look at", but we are not aware of any formal definition of the characteristics of these patterns, apart from 'extreme regularity'. However in recent years, vision researchers have been discovering increasing evidence that our perceptual system is optimized for the kinds of input found in our natural environment. It is possible that by carefully selecting texture patterns whose statistics match the statistics of natural scenes we can avoid the pitfalls of "annoying" textures and simultaneously improve both the aesthetics and the usefulness of our surface representations and we would like to look into this further.

Also, we would like to revisit the question of determining the relative effectiveness of isotropic vs anisotropic textures for shape representation. In earlier tests we had found no significant differences between these two conditions overall, but we suspect that these findings might have the result of a confluence of several competing factors. In particular, we suspect that certain pattern characteristics, such as the prominent texture flow discontinuities that can arise in 1-directional patterns, are detrimental to shape perception while other aspects, such as the explicit emphasis of

[1] In the event that this paper is accepted for publication, we would plan to report the results of these new findings in our conference presentation.

the maximal extent of the surface normal curvature in the principal directions, probably facilitate shape perception. By addressing the weaknesses in the principal direction texture model and incorporating some of the strengths of the isotropic texture model, it is possible that we will be able to achieve a pattern that more optimally facilitates shape perception. Investigations of the kind pursued in this study help us to determine where the most fertile ground lies for such pursuits.

6 ACKNOWLEGMENTS

This work was supported by a National Science Foundation Presidential Early Career Award for Scientists and Engineers (ACI-9875368). Brenda DeBlois provided us with assistance in the experimental analysis through the Statistical Consulting Clinic, which is supported by funding from the Minnesota Agricultural Experiment Station. The texture synthesis and image rendering was done using software originally written by Gabriele Gorla, with support from a University of Minnesota Grant-in-Aid of Research, Scholarship and Artistry, and with co-advice from Guillermo Sapiro. Thanks go to David Banks for suggesting early on that we look into using surface markings that follow both of the principal directions. We are grateful to Jeremy Leboy for implementing an early version of the surface attitude probe, and to Krista Janssen for developing prototype surface modeling code, with support from the CRA-W Distributed Mentor Project. We are also indebted to our most recent observers CP, SL, SK, II and TU for their dedicated and conscientious efforts.

7 REFERENCES

[1] Irving Biederman (1985). "Human Image Understanding: Recent Research and a Theory", in Human and Machine Vision, Azriel Rosenfeld, ed., Academic Press, pp. 13-57.

[2] Bruce G. Cumming, Elizabeth B. Johnston and Andrew J. Parker (1993). Effects of Different Texture Cues on Curved Surfaces Viewed Stereoscopically, *Vision Research*, **33**(5/6), pp. 827-838.

[3] James A. Ferwerda, Sumanta N. Pattanaik, Peter Shirley and Donald P. Greenberg (1997). A Model of Visual Masking for Computer Graphics, *Proceedings of SIGGRAPH 97*, pp. 143–152.

[4] Howard R. Flock and Anthony Moscatelli (1964). Variables of Surface Texture and Accuracy of Space Perceptions, *Perceptual and Motor Skills*, **19**, pp. 327-334.

[5] James J. Gibson (1950). The Perception of Visual Surfaces, *American Journal of Psychology*, **63**, pp. 367-384.

[6] Gabriele Gorla, Victoria Interrante and Guillermo Sapiro (2002). Texture Synthesis for 3D Shape Representation, *IEEE Transactions on Visualization and Computer Graphics*, to appear.

[7] Victoria Interrante, Henry Fuchs and Stephen Pizer (1997). Conveying the 3D Shape of Smoothly Curving Transparent Surfaces via Texture, I*EEE Computer Graphics and Applications*, **3**(2): 98–117.

[8] Victoria Interrante (1997). Illustrating Smooth Surfaces in Volume Data via Principal Direction-Driven 3D Line Integral Convolution, *Proceedings of ACM SIGGRAP*H, pp. 109–116.

[9] Victoria Interrante and Sunghee Kim (2001). Investigating the Effect of Texture Orientation on Shape Perception, *Human Vision and Electronic Imaging VI*, SPIE **4299**, pp.. 330-339.

[10] Victoria Interrante, Sunghee Kim and Haleh-Hagh-Shenas (2002). Conveying 3D Shape with Texture: Recent Advances and Experimental Findings, *Human Vision and Electronic Imaging VII*, SPIE **4662**, *to appear*.

[11] Alan Johnston and Peter J. Passmore (1994). Shape from Shading. I: Surface curvature and orientation, *Perception*, **23**, pp. 169-189.

[12] David C. Knill (2001). Contour into Texture: The Information Content of Surface Contours and Texture Flow, *Journal of the Optical Society of America, A*, **18**(1), pp. 12-35.

[13] Jan J Koenderink,. Andrea van Doorn and Astrid M. L. Kappers (1992) Surface Perception in Pictures, *Perception*, **52**, pp. 487–496.

[14] Andrea Li and Qasim Zaidi (2000). Perception of Three-Dimensional Shape from Texture is Based on Patterns of Oriented Energy, *Vision Research*, **40**, pp. 217–242.

[15] Andrea Li and Qasim Zaidi (2001). Information Limitations in Perception of Shape from Texture, *Vision Research*, **41**, pp. 1519-1534.

[16] Pascal Mamassian and Michael P. Landy (1998). Observer Biases in the 3D Interpretation of Line Drawings, *Vision Research*, **38**, pp. 2817–2832.

[17] C. Alej Párraga, Tom Troscianko and David J. Tolhurst (2000). The Human Visual System is Optimized for Processing the Spatial Information in Natural Visual Images, *Current Biology*, **10**, pp. 35-38.

[18] Kent Stevens (1983). The Line of Curvature Constraint and the Interpretation of 3-D Shape from Parallel Surface Contours, *Proceedings of the International Joint Conference on Artificial Intelligence*, pp.1057–1061.

[19] James T. Todd and Francene D. Reichel (1990) Visual Perception of Smoothly Curved Surfaces from Double-Projected Contour Patterns, J*ournal of Experimental Psychology: Human Perception and Performance*, **16**(3), pp. 665-674.

[20] James T. Todd, J. Farley Norman, Jan J. Koenderink and Astrid M. L. Kappers (1997). Effects of Texture, Illumination and Surface Reflectance on Stereoscopic Shape Perception, *Perception*, **26**(7), pp. 807-822.

[21] Andrew P. Witkin (1981). Recovering Surface Shape and Orientation from Texture, *Artificial Intelligence*, **17**, pp. 17-45.

Texture Synthesis for 3D Shape Representation

Gabriele Gorla[†], Victoria Interrante[‡] and Guillermo Sapiro[†]
[†]*Electrical and Computer Engineering,* [‡]*Computer Science and Engineering*
University of Minnesota

Abstract

Considerable evidence suggests that a viewer's perception of the 3D shape of a polygonally-defined object can be significantly affected (either masked or enhanced) by the presence of a surface texture pattern. However investigations into the specific mechanisms of texture's effect on shape perception are still ongoing and the question of how to design and apply a texture pattern to a surface in order to best facilitate shape perception remains open. Recently, we have suggested that for anisotropic texture patterns, the accuracy of shape judgments may be significantly affected by the orientation of the surface texture pattern anisotropy with respect to the principal directions of curvature over the surface. However it has been difficult, until this time, to conduct controlled studies specifically investigating the effect of texture orientation on shape perception because there has been no simple and reliable method for texturing an arbitrary doubly curved surface with a specified input pattern such that the dominant orientation of the pattern everywhere follows a pre-defined directional vector field over the surface, while seams and projective distortion of the pattern are avoided. In this paper, we present a straightforward and highly efficient method for achieving such a texture and describe how it can potentially be used to enhance shape representation. Specifically, we describe a novel, efficient, automatic algorithm for seamlessly synthesizing, from a sample 2D pattern, a high resolution fitted surface texture in which the dominant orientation of the pattern locally follows a specified vector field over the surface *at a per-pixel level*, and in which seams, projective distortion, and repetition artifacts in the texture pattern are nearly completely avoided. We demonstrate the robustness of our method with a variety of texture swatches applied to standard graphics datasets, and we explain how our method can be used to facilitate research in the perception of shape from texture.

CR categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism — Texture.
Keywords: Texture synthesis, texture mapping, shape perception, shape representation.

1. Introduction

Adding texture to the surface of a polygonal model can not only profoundly enhance its visual richness, but can also significantly affect our perception of the object's geometry. Certain textures have been shown to impede accurate shape perception, for example by masking faceting artifacts [11]. Others may have the potential to enhance shape perception by emphasizing the lines of curvature of the form [19]. In computer graphics and visualization, where we have the ability to model a textured object in any way that we desire – to both define the texture pattern and define how it is applied over the surface – we have the potential to use texture in highly controlled ways to influence shape perception. Unfortunately, the existing theories on shape perception from texture do not provide sufficient guidance to answer the question of how to best design and apply a texture to a surface in order to facilitate the accurate understanding of its shape.

Researchers in perceptual psychology have been investigating the effects of various texture pattern characteristics on surface shape perception for many years through controlled observer experiments [6, 22, 24, 26, 36]. Unfortunately the scope of these studies has been limited by the capabilities of available rendering and texture-mapping utilities and algorithms. In particular it has not been possible, in general, to map an arbitrary pattern onto an arbitrary doubly curved surface so that the orientation of the pattern everywhere follows a specific predefined vector field at a per-pixel level, while minimizing any distortion of the underlying pattern and avoiding the introduction of pattern discontinuities. Hence most of the studies conducted to date have either not considered the effect of the orientation of the texture pattern with respect to the surface curvature [6, 36], or have been limited to highly restricted synthetic texture patterns, such as isolated pairs of line segments [26], or have been limited to highly restricted surface geometries, such as the case of singly-curved surfaces [22, 24]. As a result, our ability to gain deeper insight into the specific impact of texture pattern orientation on surface shape perception, and to use this insight to inform theories of shape perception from texture, has been limited. In particular, it is not yet clear to what extent the introduction of texture pattern anisotropy *per se* interferes with surface shape perception [6], or to what extent it is sufficient for unimpeded shape perception to ensure only that the texture pattern does not turn in the surface (i.e. that it does not contain significant geodesic curvature) [22].

In computer graphics and visualization, many feel that the importance to shape perception of the particular alignment over the surface of an anisotropic texture pattern remains open to debate. However we have come to believe, based on informal observations of numerous surfaces under numerous texture conditions, that one's ability to make accurate judgments about the shape of an underlying surface can be significantly influenced both by the characteristics of the texture pattern itself *and* the way in which the pattern is laid down over the surface. Clearly, in order to objectively assess the impact of texture orientation on surface shape perception, it is necessary to conduct further controlled, quantitative experiments. In a recent study using line integral convolution based texture [20], we found indications that observers' shape judgments of a doubly-curved surface are more accurate in the presence of a purely anisotropic texture that follows the first principal directions of curvature over the surface (a special instance of a pattern with zero geodesic curvature) than in the presence of either a purely anisotropic texture that exhibits zero geodesic curvature but does not follow one of the principal directions, or a pattern that contains significant non-zero geodesic curvature (figure 1) [20]. However we found no indications that shape perception is significantly better in the presence of a purely anisotropic principal direction oriented texture pattern than it is in the control case of the purely isotropic texture pattern.

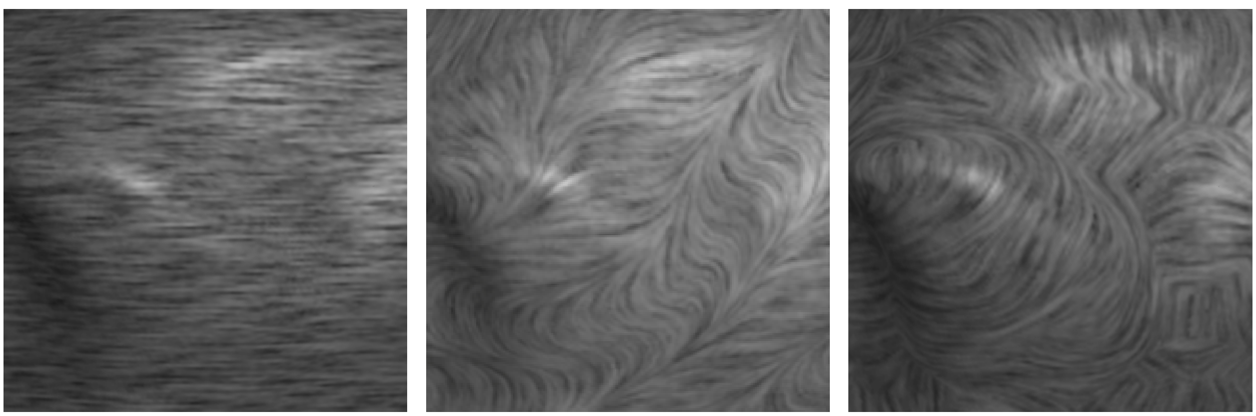


Figure 1: Sample close-up images in an experiment examining the effect on shape perception of differently oriented anisotropic texture patterns synthesized via line integral convolution. Left: a uniformly oriented texture with zero geodesic curvature, Center: a texture with non-zero geodesic curvature; Right: a texture with nearly zero geodesic curvature that follows the first principal direction over the surface. The underlying surface shape is identical in all three cases.

In order to further investigate the key question of how we might best both define and apply a texture pattern to facilitate surface shape perception, we will have to conduct additional studies using a wider variety of surface texture patterns. This requires an algorithm for texturing an arbitrary doubly curved surface with an arbitrary 2D pattern such that the dominant direction of the pattern follows a specified vector field over the surface at a per-pixel level. In this paper we describe such an algorithm that we developed for this purpose. Our algorithm is very straightforward, easy to implement, and highly efficient, and has the potential to be useful for a wide variety of graphics applications that require the aesthetic mapping of a given texture pattern to a given surface, independent of the desire to effectively portray the surface shape.

Given a 2D texture pattern and a polygonal surface model, the historical challenge has been to determine how to apply the pattern to the surface in a manner that minimizies the visual impact of seams and projective distortion while orienting the pattern so that it flows over the shape in a desirable way.

Many different approaches to this basic problem are possible and, concurrently with our work, many similar approaches have been developed. The key distinguishing features of the method that we describe in this paper stem from the fact that it was specifically developed for the purposes of shape representation, in which curvature information is carried by a high resolution texture pattern. Our method has the advantages of being highly efficient for large quantities of texture, very straightforward to implement, and producing high quality results across a wide variety of texture types and models. For the purposes of shape-from-texture investigations, our fitted texture approach is superior to our previous 3D line integral convolution approach [20] because the resulting textured objects can be easily displayed at interactive frame rates using a conventional renderer on a standard PC with texture mapping hardware.

Our technique consists of the following main steps:

- Partition the polygons of the model into contiguous patches, as nearly planar (to prevent distortion) and as nearly similarly sized (to simplify texture map handling) as reasonably possible.
- Compute a vector field over the object, or read a pre-defined field from a file.
- Synthesize the texture pattern over each patch, maintaining pattern continuity across the boundaries with neighboring patches, using an efficient, orientation-adaptive variation of the non-parametric sampling method proposed by Efros and Leung [9].

An example of the results of our algorithm is presented in figure 2.

2. Previous Work

A variety of methods have been previously proposed for texturing polygonal models with patterns that are as free as possible of seams and distortion artifacts.

One method is solid texturing [29, 30, 42], in which the texture pattern is defined over a 3D volume. Particularly good results have been achieved with this method for water, as well as for objects made of wood and stone. However, there are significant challenges in synthesizing 3D textures modeled after sampled materials [17, 8], and current methods for creating custom-fitted 3D textures whose features follow a surface's shape [19] are severely limited in scope and applicability.

Methods for applying 2D image-based texture to arbitrary polygonal models for the most part must balance the inherent trade-off between seams and distortion (one cannot in general apply a 2D image to a non-developable surface without incurring one or the other), employing piecewise flattening in the case of arbitrary parametric [3] or polygonally-defined [25] models, or using careful surface parameterization [23], or pre-distortion of the texture [1, 41] to achieve desired results in other particular cases. Conformal mapping [15] offers a global solution that preserves angles, but not lengths or areas.

Closer to our objectives, Neyret and Cani [28] proposed an excellent technique for achieving seamless and virtually distortion-free mapping of 2D isotropic texture patterns on arbitrary objects via custom-defined triangular texture tiles that are continuous with one another across various of their boundaries. Unfortunately an extension of this method to anisotropic texture patterns is not obvious. Praun et al. [33] subsequently proposed "lapped textures", which provides capabilities that are the most similar to those towards which our method aspires, although the approach that we take is very different. The lapped texture method repeatedly pastes copies of a sample texture swatch onto overlapping patches across a surface after some subtle warping and reorientation to align the pattern with a user-defined vector field. This method produces very good results when used with texture patterns that contain enough high frequency detail and natural irregularity in feature element sizes and relative positions. This is needed to perceptually mask artifacts due to the partial overlap or misalignment of feature elements across patch boundaries. As currently formulated, the lapped texture approach is not particularly well-suited for rigidly structured patterns, such as a checkerboard, or textures such as netting, which are characterized by the global continuity of specific elongated elements. It is also less well-suited for use with vector fields that contain significant high frequency variation. In addition, the lapped texturing process as described in [33] involves considerable amounts of user interaction.

Figure 2: An example of a synthesized surface texture produced by our method. No manual intervention of any kind was employed. This texture was grown from an original 92x92 swatch [5], pre-rotated to 63 orientations each cropped to 64x64 pixels, to cover 291 surface patches at 128x128 resolution following a vector field locally defined by the projection of (0,1,0) onto the tangent plane at each point. The entire process required approximately 12 minutes.

The method that we describe in this paper provides capabilities beyond those offered by these previous methods. It achieves nearly seamless and distortion-free texturing of arbitrary polygonally-defined models with a texture pattern derived from a provided sample. Most importantly, the method is suitable for use with anisotropic patterns. It generally preserves larger scale texture pattern continuity across patch boundaries, does not require manual user intervention, and allows the orientation of the applied pattern to locally follow a specified vector field on a per-pixel basis.

Our method falls into the category of methods that achieve texture pattern continuity without distortion by in effect synthesizing the texture "in place" over the surface of the object. Previous methods in this category include direct painting [16], and reaction-diffusion texture synthesis [37, 41], which yield excellent results for hand-crafted textures and textures modeled after organic processes. Our goal was to achieve similarly good results with automatically synthesized textures that are perceptually equivalent to a given sample swatch.

Our work is perhaps most fundamentally motivated by the impressive advances in texture synthesis methods [17, 7, 32, 44, 9, 39] which have made it possible to create, for an increasingly wide range of patterns, unlimited quantities of a texture that is perceptually equivalent to a small provided sample.

Leveraging research in human texture perception, Heeger and Bergen [17] developed a highly successful method for synthesizing textures that capture the essential perceptual properties of a variety of homogeneous stochastic sample patterns. Their method works by iteratively modifying a random noise image so that its intensity histogram matches the histogram of the sample texture across each of the subbands in a steerable pyramid representation of each image. De Bonet [7] developed a related method based on interchanging elements in the Laplacian pyramid representation of a self-tiling pattern where possible, while preserving the joint-occurrence relationships of features across multiple resolutions. This method yields impressive results for an even wider variety of patterns, though some difficulties remain in preserving larger scale globally significant structure. Several other highly sophisticated texture analysis/synthesis approaches have been subsequently developed [9, 32, 44]. Of these, we chose to follow the texture synthesis approach proposed by Efros and Leung [9] because of its combination of simplicity and quality of output. In this method, a new texture pattern is grown, pixel-by-pixel, by sampling into a provided template pattern and choosing randomly from among the pixels whose neighborhoods are close matches to the yet partially-defined neighborhood of the pixel to be filled in, in the pattern being synthesized.

Subsequent to the appearance of the original Efros and Leung paper, and concurrently with the development of our method, a number of new advancements in 2D and 3D texture synthesis have been achieved. Wei and Levoy [39] proposed a method that addressed one of the most serious concerns with the method of [9] which was speed. Their method used tree-structured vector quantization to improve, by several orders of magnitude, the speed of the search for the pixel with the best matching neighborhood, at the cost of some loss of quality in the resulting synthesized patterns. Ashikmin [2] proposed a method, based on the cutting and pasting of larger areas than a single pixel from the sample texture, that achieved improved results for highly structured textures such as flowers and leaves. Efros and Freeman [10] proposed a new method, called 'texture quilting', that produces even more consistently excellent results across a broad spectrum of texture patterns by specifically maintaining both continuity and coherence across broader local regions of the pattern. Most similar to the objectives of our work, Wei and Levoy [40], Turk [38], and Ying *et al.* [43] all proposed methods for synthesizing a sample 2D texture pattern over an arbitrary mesh in 3D, with the objective of resolving the classical texture mapping problem: to avoid seams and to minimize pattern distortion. The results produced by our method are very similar in many respects to the results produced by these concurrently developed methods, with some subtle differences that will be discussed in a later section. The distinguishing characteristic of our method is that it is optimized for the synthesis of large quantities of *high resolution* texture in which the direction of the texture pattern follows a specified vector field, such as the direction of greatest normal curvature, over the surface at a per *pixel* level, in order to be useful for applications in which one is specifically concerned with the use of texture for facilitating shape representation.

In the remainder of this paper we describe the method that we have developed and the details of its implementation, talk about some of the issues that arise in automatically determining a good way to orient a texture pattern over a surface, show representative results, and conclude with a discussion of the current limitations of our implementation and directions for future work.

3. Proposed Method

Our proposed method is basically a two step process. First the surface is partitioned into small, almost flat patches. Then, the texture is grown over the planar projection of each individual patch taking into consideration the proper boundary conditions to maintain the continuity of the texture pattern across seams at the patch boundaries. During the synthesis of an anisotropic pattern, the texture is locally constrained into alignment with a specified vector field over the surface. For simplicity, we store the resulting synthesized texture as a collection of separate small images for each patch, although other approaches are certainly possible.

3.1 Partitioning

The goal of the first stage is to partition the mesh into a minimum number of approximately planar patches (collections of triangles). Obviously these are conflicting goals for any closed surface and a suitable tradeoff must be found. To keep our implementation as simple as possible, we restrict patches to be of approximately the same size. Maintaining relatively consistent patch sizes simplifies texture memory management by allowing us to allocate and synthesize texture maps of a consistent fixed resolution for each patch. Please note that mesh partitioning is a task that is common to many computer graphics algorithms and many approaches have been previously described [cf. 34, 27]. In this section we describe for the sake of completeness the details of the particular approach that we used.

Two input parameters define the maximum patch size (which influences the scale at which the texture appears over the surface) and the maximum projective distortion that the user is willing to tolerate. The initial partitioning is done with a greedy algorithm, after which an optimization step is performed to reduce the average projection error.

The process for the initial partitioning can be summarized by the following pseudo code:

```
while (unassigned_triangles > 0) {
    pick an arbitrary unassigned triangle T;
    assign T to a new group G;
    add to group G all connected triangles C that satisfy:
        - Normal(C) • Normal(T) > min_cosine_displacement;
        - distance from the center of C to the farthest vertex of T is less than max_dist ;}
```

The image on the left side of figure 3 shows a representative result after the first stage in the splitting. The green triangles are the reference triangles that define the plane onto which the patch will ultimately be flattened. It is easy to notice that some of the patches obtained at this point are very small and/or contain triangles that are relatively far from being aligned with the reference plane. A refinement pass is used to reduce both the number of patches and the number of triangles that are oriented at a sharp angle to the plane into which they will ultimately be projected. Two simple experimental rules are iteratively applied until no significant improvement is observed:

- remove a patch if it is very small, and its triangles can be added to a neighboring patch without violating the distance constraint;
- reassign a triangle T from patch P1 to a neighboring patch P2 if T borders P2 and is more closely aligned with the reference plane of that patch than with its own.

The image on the right side of figure 3 shows the results after iterative refinement. The simple refinement procedure that we use is not guaranteed to converge to the theoretically optimal result but we have found that the results are consistently good and quite sufficient for our purposes. Since both the splitting and optimization stages are of linear complexity and account, on average, for between 1 and 4% of the total computational time we did not feel the need to improve their speed. The greedy splitting step took about 2 seconds, and the refinement about 10 seconds for the 70,000 triangle data set shown. In the rare event that acceptable results are not achieved in this phase, the splitting process can be repeated using a tighter limit on the acceptable normal error (which will result in more, smaller patches). Increasing the

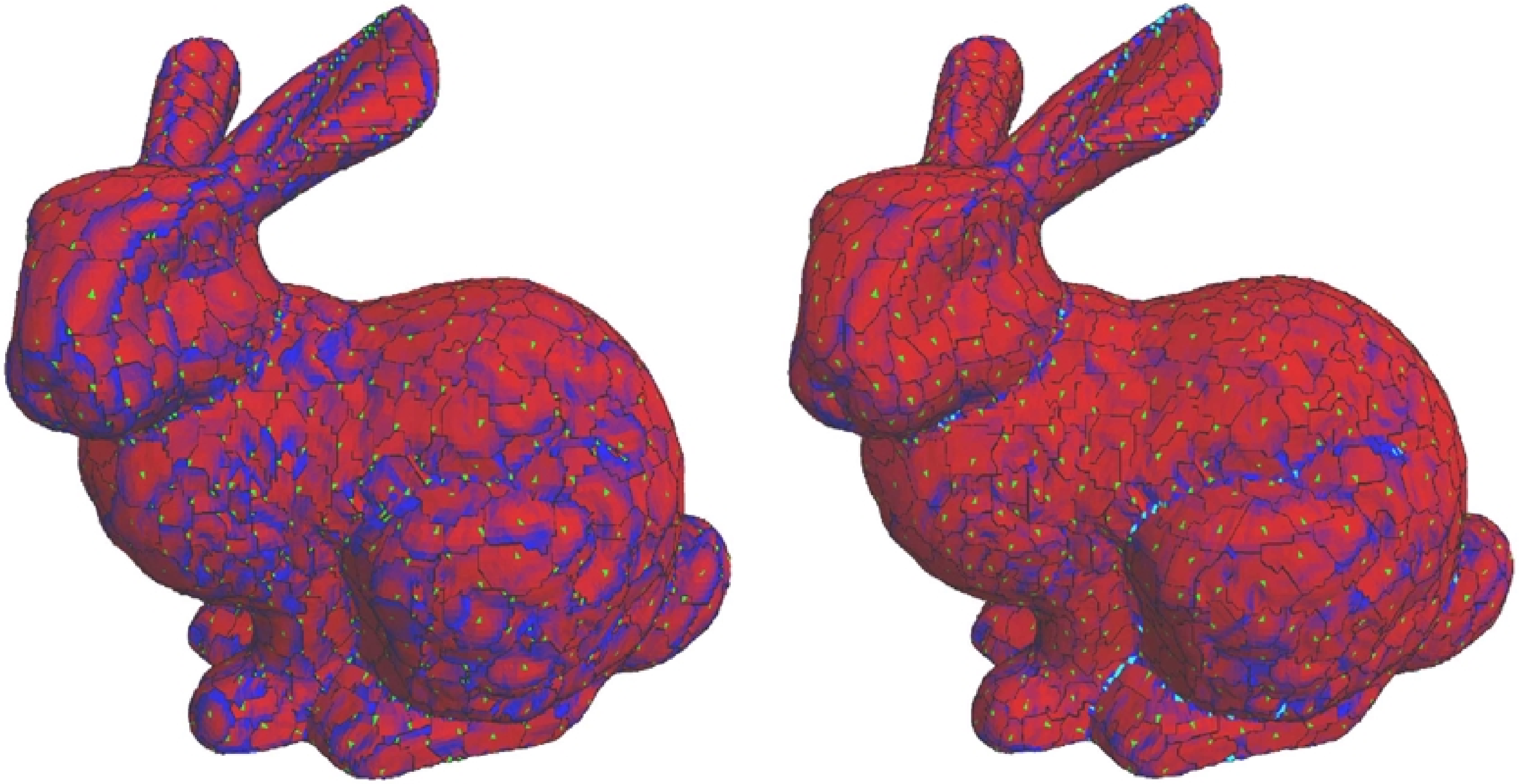


Figure 3: The partitioned bunny, after the first stage of our "greedy" splitting algorithm (left) and after iterative refinement (right), which decreased the number of patches from 988 to 699. Patch boundaries are outlined in black on each model. The green triangles are the seed triangles, which also define the reference plane for each patch. Triangles whose normal directions differ by less than 18 degrees from the direction of the normal to the reference plane, corresponding to a 5% error in the linear projection, are colored red. Triangles up to 25 degrees out of alignment with the reference plane for their patch, corresponding to a 10% error in the projection, are colored blue. Triangles rotated more than 25 degrees from the reference plane (>10% error) are colored cyan.

number of patches does not cause an increase in the computational expense in the subsequent texture synthesis step since the synthesis cost is only dependent on the total number of pixel synthesized. More significant is the issue that with very small patches comes an increased risk that the texture synthesis process will run into difficulties and “grow garbage”, either due to the paucity of available contextual information along the shortened boundary, or to the near proximity of mutually incompatible pre-defined boundary conditions.

3.2 Parameterization

After the model has been partitioned into contiguous patches, the triangles comprising each patch are projected onto their common reference plane, and the texture coordinates are defined at each vertex according the coordinates of the projected vertices in the reference plane coordinate system. One major advantage of such a simple parameterization is that there is no need to store the texture coordinates with the output model as they can be easily recomputed at runtime. Adjacent triangles from the neighboring patches, which provide the boundary conditions for maintaining the continuity of the texture pattern during synthesis, are then rotated about their shared edges into the reference plane. We use rotation for these triangles rather than projection to minimize the projective distortion of the texture that we will need to refer to for reference purposes. However, it is necessary to check for the very infrequently encountered cases where it is not possible to rotate each of the adjacent triangles of a particular patch into the projection plane without causing some of these triangles to overlap. This entire process is illustrated in figure 4.

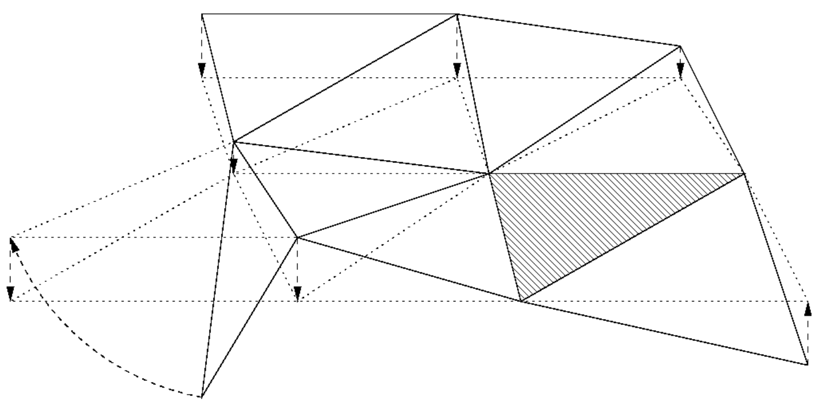


Figure 4: A diagrammatic illustration of the flattening process. The normal of the grey-shaded triangle defines the plane into which the remaining triangles in the patch are projected. The union of these triangles defines the area across which the texture will be synthesized. Neighboring triangles from the adjacent patches are rotated (to minimize projective distortion) into the plane so that any texture already present in these triangles can provide boundary conditions for the texture synthesis, in order to enable the achievement of a seamless final result.

3.3 Synthesis

At this point, we have created a 2D image for each patch containing:

- an area, defined by the projection of the triangles of the patch onto the reference plane, which contains the pixels to be filled by the synthesized texture; and
- an area, defined by the rotation into the reference plane of the neighboring triangles from the adjacent patches, that will hold any previously synthesized texture and provide the boundary conditions necessary to avoid seams due to discontinuities between texture element features in adjacent patches.

It is important to note that the partitioning process described in the immediately previous sections is completely independent from the synthesis algorithm. Any constrained synthesis method that can fill arbitrary regions with arbitrary boundary conditions could potentially be used, although none of the existing algorithms we reviewed appeared to provide both the image quality and the speed required for this project. The work of Efros and Leung [9] came closest to meeting our needs and thus we elected to follow their general approach, which has been shown to produce good results for a wide range of texture patterns that can be modeled by Markov random fields (i.e. textures whose characteristics are fairly consistent under translation over the image and in which the value at any given point can be fully characterized by the values at its closely neighboring points over a limited range).

In order to make tractable the problem of efficiently synthesizing enough texture to cover a standard model of arbitrary topology at a reasonable resolution, the first objective of our proposed method is to achieve results that are of the same caliber as those demonstrated by Efros and Leung but that require significantly less time, while also preserving the flexible applicability of their approach. To do this, we use a new two-pass search strategy. The first pass, which is exhaustive, is done using a very small unweighted neighborhood (usually between 1/3 to 1/2 of the diameter of the full size neighborhood). The *n* best matches, where *n* is a user-definable parameter, are saved in a list to be processed by the second pass. This two pass approach presumes strong locality in the input textures (which holds true for many natural texture patterns) and has the effect of rapidly eliminating most of the uninteresting part of the search space. The size of this preselect list ultimately determines the overall speed of the synthesis algorithm. We found that some textures produced excellent results with preselect lists of as few elements as the number contained in one scanline of the original image; these we considered easy to synthesize. Others required 4 or even 8 times more elements in the preselect list, and these we considered hard to synthesize. Instances of easy and hard textures are shown in figure 2 and figure 9 respectively. In the second pass, each of the pixels in the preselect list is tested against the full size weighted neighborhood and the error metrics are updated. Among the best 10 or 10% of matches (whichever is greater) a random pixel is chosen and used in the synthesized image. Figure 5 shows a sample result of this synthesis algorithm in the 2D case. The speed of our method does not match the speed of Wei and Levoy's tree-structured vector quantization [39] for the synthesis of rectangular swatches of texture, but unlike their method it does not require the use of a fixed size causal neighborhood, and the results it produces are of consistently high quality. The proposed method is still fast enough to make feasible our goal of growing a fitted surface texture via the Markov random field sampling approach. Figure 6 illustrates the complete texture synthesis process. The image on the upper left represents the state of the model after the synthesis of the first two patches. The image in the upper center identifies the triangles comprising the third patch, with the triangle that defines the plane of the patch rendered in green, and the rest of the triangles rendered in red. The image on the upper right shows the results after the patch is filled with the synthesized texture. The image on the lower left shows the projection of the patch onto the plane. The image in the left center of the lower row shows the texture boundary conditions supplied by the neighboring triangles to this patch. The next two images in the lower row show what the patch looks like midway through and at the end of the texture synthesis step.

3.3.1 Isotropic Textures

As basically formulated, the approach we have just described can be used to cover the surface of a model with an isotropic texture pattern in an orientation-insensitive way. In other words, if we assume that our sample texture pattern is perfectly rotationally symmetric, we can directly use this approach, in its most basic form, to seamlessly synthesize the texture pattern across all of the patches in the model without any special considerations apart from the boundary conditions. However, as we quickly found, there are very few acquired textures that can be used with good results without regard to orientation. Even patterns which we initially believed to be isotropic based on inspection of the 2D sample image revealed unexpected orientation dependencies due to the subtle structuring that stems from the illumination process. Figure 7 illustrates this problem. Notice how the wool texture appears flat when applied without regard to orientation. This is due to the disruption of the pattern of shading.

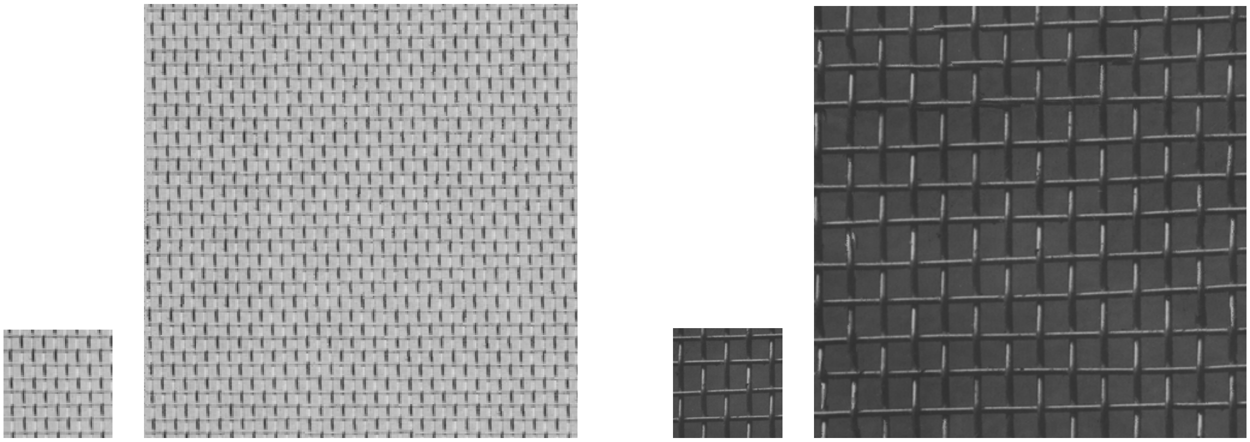


Figure 5: Left: An example of the results of our two-pass texture synthesis method using pattern D06 from the Brodatz album [5]. The speed of the synthesis approach varies from pattern to pattern depending on the sizes of the match-defining neighborhoods and the length of the preselect list. In this case we used a first pass neighborhood of 5x5, a maximum preselect list length of 64, and a second pass neighborhood of 12x12 to synthesize the 256x256 patch on the right from the 64x64 pixel sample on the left in about 73s. Right: an example using D01.

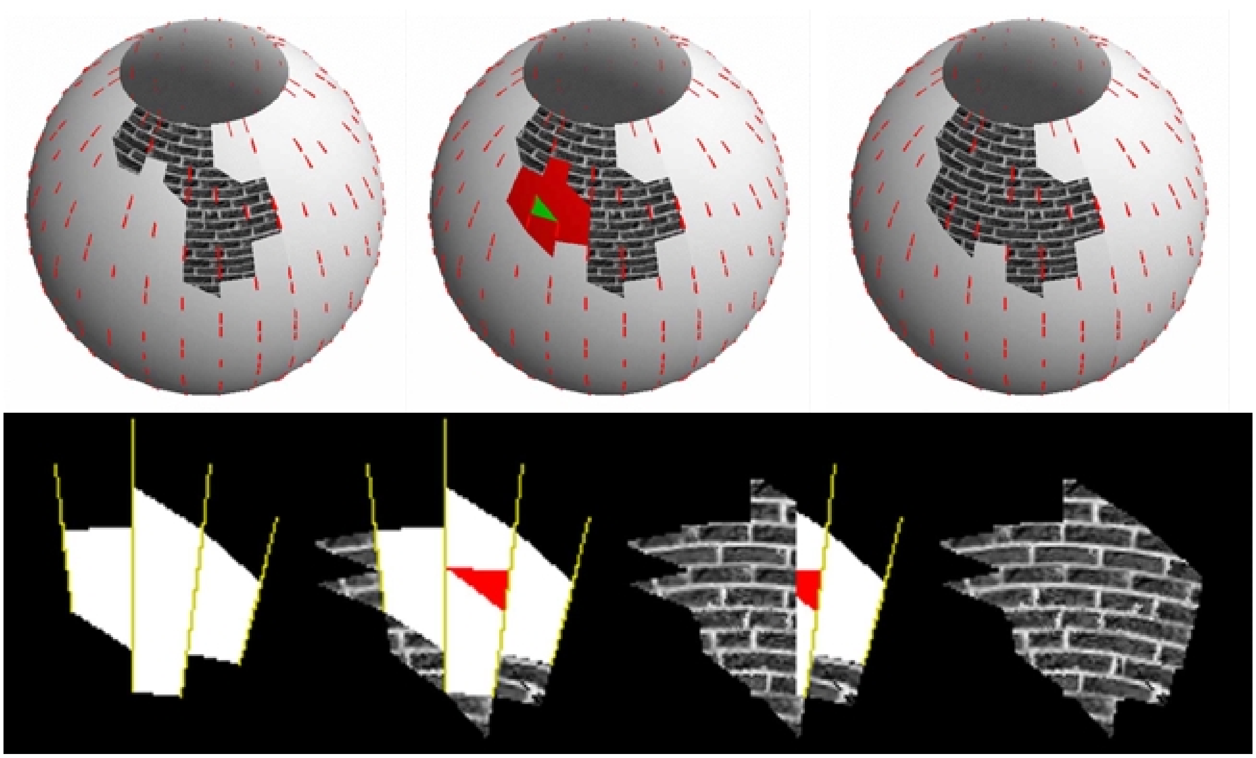


Figure 6: A step-by-step illustration of the basic process of our method. Upper row: the identification of the patch to be synthesized, and the synthesis result. Lower row (from left to right): the planar projection of the patch; the boundary conditions provided by the neighboring textured triangles rotated into the plane of the patch; midway through the texture synthesis process (synthesis is proceeding from left to right); the complete synthesized patch.

Figure 7: A texture ('wool.bw', from SGI) that originally appeared to be isotropic, reveals its anisotropic nature (due to the effects of shading) when synthesized over the Stanford bunny dataset via an approach in which the texture orientation is allowed to vary arbitrarily between each patch.

3.3.2 Directional Textures

In the vast majority of cases, it is necessary to control the orientation of the texture over the surface. For greatest flexibility, we allow a directional texture to follow any specified direction field. In the next section some examples will be discussed. We note that in the lapped textures method, Praun et al. [33] also align textures on a per patch basis, slightly distorting the parameterization to achieve good local continuity within the patch with the underlying directional specification. In the case of sparse triangulation, they reduce undersampling of the vector field by locally subdividing the mesh.

In our presented method the synthesis algorithm has been enhanced to allow per-pixel texture re-orientation. We pre-rotate the original texture into a quantized number of orientations, and during synthesis perform the search for best-matching neighborhoods in the pre-rotated image that is most closely aligned to the direction locally specified by the vector field. Figure 8 shows a sample of one

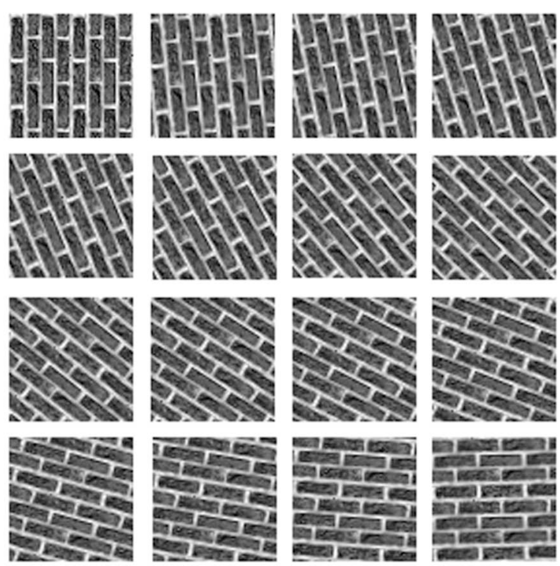


Figure 8: A quadrant of pre-rotated brick texture samples.

quadrant of pre-rotated brick texture. If the number of pre-rotated images is sufficiently high, the synthesized texture will follow the vector field smoothly. For the examples in this paper, we used between 64 to 128 different rotations of the input texture. Although it is of course possible to specify the use of any arbitrary number of pre-rotated images, we did not notice an appreciable increase in the quality of the results when finer quantizations were used.

Searching for matches in pre-rotated texture images allows considerably faster synthesis than would be possible if we had to perform the rotation on-the-fly for each pixel of the texture during synthesis. While it is relatively fast, with modern 3D hardware, to compute any arbitrary rotation of the original image, in most implementations there is a very high cost associated with reading the results from the frame buffer.

3.3.2.1 Constant Direction Fields

We originally began this work with the intent to explore the possibility of applying textures along the principal directions of curvature. Despite the latent potential in that approach, it is not without its difficulties, which we will discuss in greater detail in the following section. We quickly discovered that aesthetic results could be also achieved for a wide range of models using other, much simpler, vector field definitions. Notably, the field of "up" directions, locally projected onto the tangent plane at each point, appears to yield particularly nice results for many textures and datasets, as shown in figures 9 and 10. It is worth mentioning in the context of these two images that we worked hard to challenge our texture synthesis method, testing its performance on difficult texture patterns such as the crocodile skin, which contains potentially problematic sets of features spanning a wide range of spatial frequencies (from 3–21 pixels in diameter), and the square glass blocks pattern, which is a highly structured checkerboard-style design in which irregularities in the size, shape and/or positioning of any of the elements have the potential to stand out especially prominently.

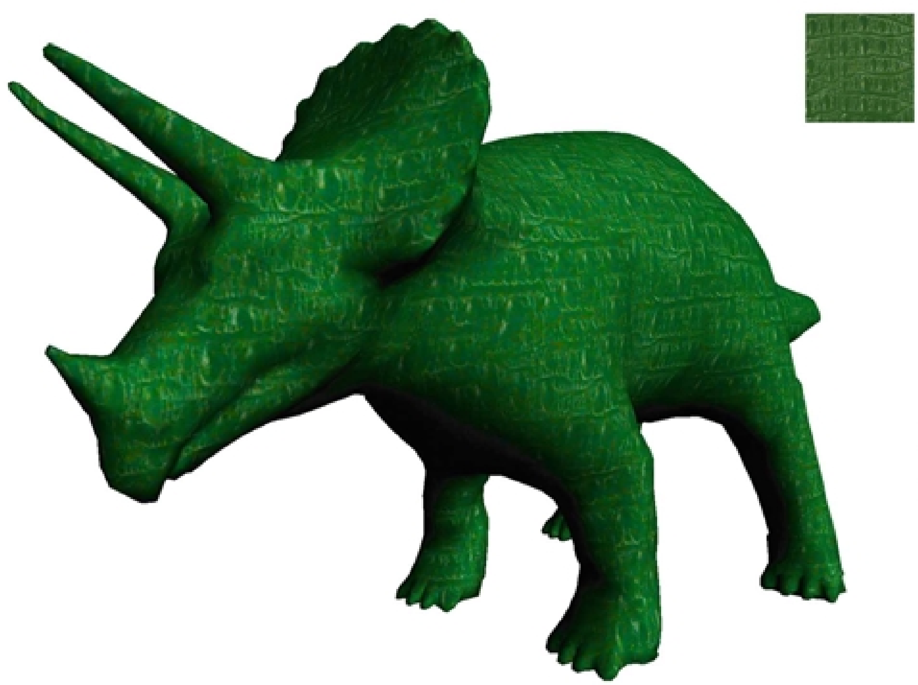


Figure 9: The crocodile skin texture (D10) synthesized over the triceratops model following the direction field (0,1,0) locally projected onto the tangent plane.

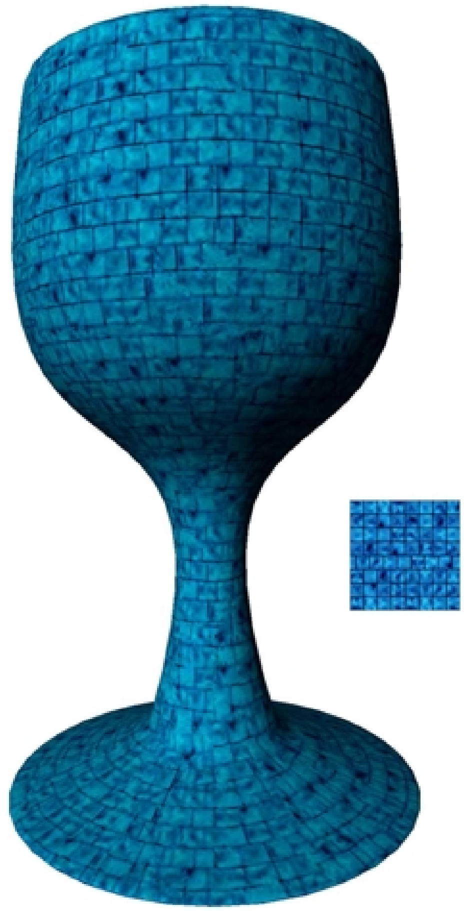


Figure 10: Glass block texture applied to a simple model with a constant directional field. Note the general preservation of continuity in the texture pattern and the relative consistency of the bricks' shapes and sizes, despite scattered artifacts. The direction field is locally given by the projection of the central axis of the object onto the tangent plane of each patch. Some of the patches in this model were resynthesized in a postprocess.

3.3.2.2 Principal Direction Fields

Although the constant direction field produces aesthetic results in many instances, there are also many cases for which it is not well suited. Specifically, it tends to fail for models that do not have a single well-defined intrinsic orientation, and it can not successfully emphasize local shape features. Of greatest intrinsic interest to our ongoing research is the possibility of applying an oriented texture pattern to the surface of an object such that it will be everywhere aligned with the principal directions of curvature.

Recent results in biological vision research support the idea that the principal directions play an important role in surface shape understanding, and we are interested in probing these ideas further through controlled studies of the effects of texture pattern orientation on observers' perception of the 3D shapes of complicated underlying models. Mammassian and Landy [26] have shown that observers' interpretations of line drawings of simple patches are consistent with an inherent bias, among other things, towards interpreting lines on objects as being oriented in the principal directions, supporting an observation made by Stevens [35] nearly 20 years ago. Li and Zaidi [24] examined observers' ability to estimate the relative curvatures of developable surfaces textured with various implicitly or explicitly plaid-like patterns, and concluded that shape perception depends critically upon the observation of changes in oriented energy along lines corresponding to the principal directions. However these ideas remain to be examined in the context of more complicated, arbitrary surfaces, where the first and second

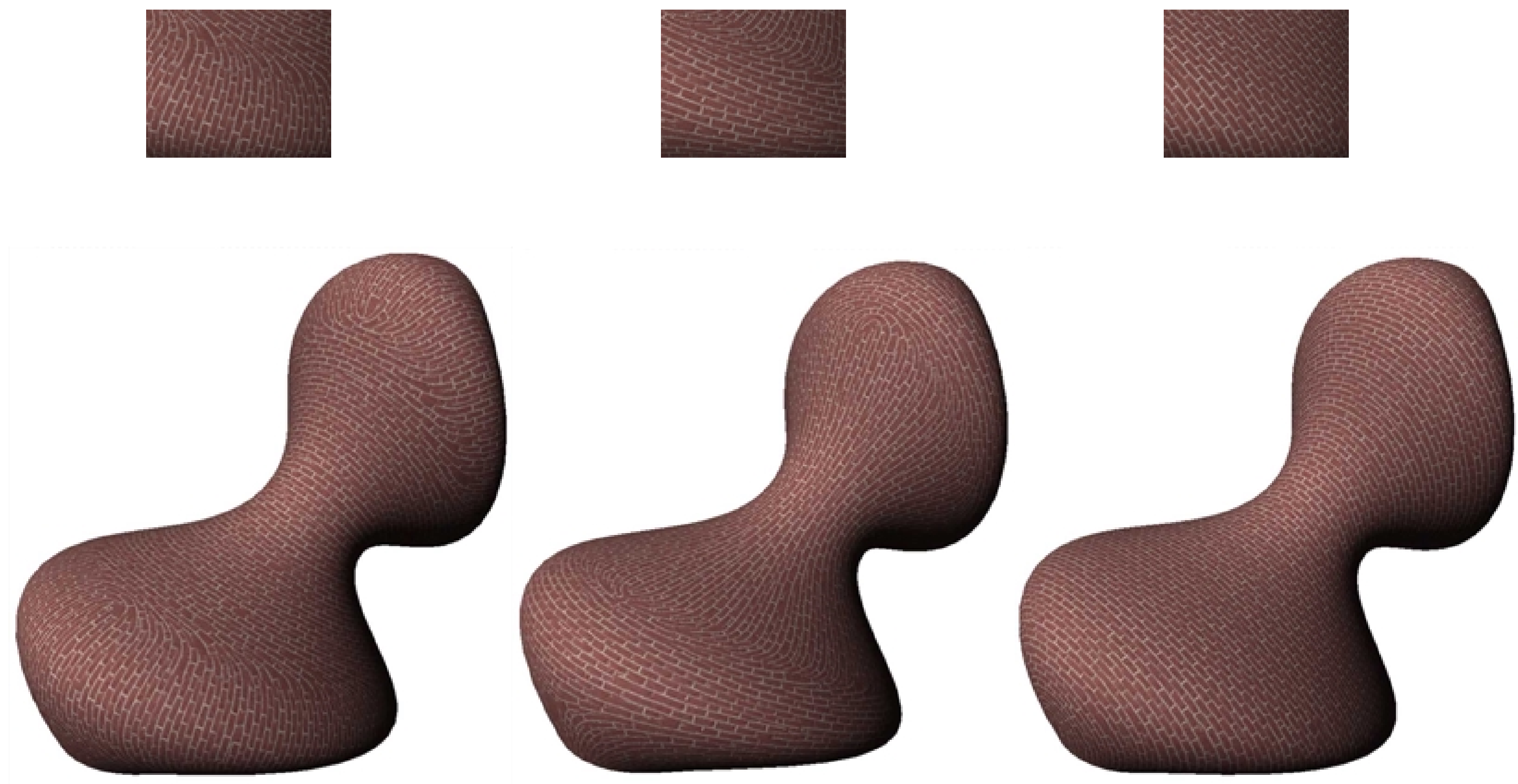


Figure 11: An illustration of the effect that the orientation of a directed pattern over a curved surface can have on our perception of the surface's 3D shape. On the left, the bricks are oriented in the direction of greatest signed normal curvature; in the middle they are oriented in the direction of least signed normal curvature, and on the right they are oriented in the same constant "up" direction used for the models in figures 9-10. Below the entire surface is shown, with the silhouette cues to shape available for reference.

principal directions can switch places numerous times. A significant challenge in this effort is to obtain accurate computations of the principal direction vector fields.

We recently worked with Dr. Jack Goldfeather to develop robust methods for computing smooth, accurate principal direction vector fields across arbitrary polygonally-defined objects [13]. A complementary approach developed by Bertalmio *et al.* [4] has the potential to facilitate the anisotropic smoothing of these fields. Our present results in applying an anisotropic texture over the surface of an object such that its dominant orientation is everywhere aligned with the first and second principal directions are shown in figure 11, and contrasted there with the results obtained using a constant "up" direction. Although it is clearly not possible to prove the benefit of a principal direction oriented texture for shape representation through one or even a handful of representative examples, with the existence of a method for synthesizing a variety of principal direction textures over arbitrary curving forms it becomes feasible to rigorously investigate the impact of various texture orientation strategies on the accuracy of shape perception judgments through controlled observer experiments. We are currently in the midst of carrying out a set of such studies and expect to report the results shortly [21].

4. Implementation

Our system is fully automatic, and does not require user interaction during either the splitting or texture synthesis process. The system has several parameters which can be adjusted by the user to increase the likelihood of obtaining optimal results with different kinds of textures or models. Within the splitting stage, these parameters include: an upper bound on the size of any single patch during splitting (which ultimately affects the scale of the texture on the model) and an upper bound on the angle that the normal of any member triangle can make with the reference direction for a patch (which affects the size and total number of patches). Within the texture synthesis stage, the user may first choose among several possible texture orientation options: to have the texture follow the direction of greatest or least signed or unsigned normal curvature, to follow the projection onto the local tangent plane of a constant specified direction, or to follow no specific direction (in which case the pattern is assumed to be invariant under rotation). With regard to texture synthesis, the user may also control: the sizes of the neighborhoods used in the each pass of the texture synthesis (larger neighborhoods generally increase the computational expense of the synthesis but are sometimes necessary in to preserve features across a range of different scales); the number of first-round preselected locations to be tested for a match on the second pass; and the weighting scheme used over the neighborhood during the matching process.

In most cases, it is sufficient to define the direction of texture synthesis across a patch according to the distribution pattern of previously textured pixels in the boundary region, under the assumption that starting from the side containing the greatest number of previously filled pixels will provide the most stable seed for the synthesis. Unfortunately, this is not always true (see the section on errors below). Certain strongly directional patterns seem to yield better results when the synthesis is performed following the direction of the vector field controlling the texture orientation. This approach was used also by Turk [38]. For quickly varying direction fields, starting the synthesis from an area of the patch in which the direction field is most calm seems to improve the quality of the result.

We use one of two different methods to determine the direction in which the synthesis proceeds from patch to patch. The simple method, which seems to works well on fairly uniform vector fields, is to begin with a randomly chosen patch and proceed to any blank connected patch, filling in any holes at the end. For principal direction vector fields better results can be achieved choosing the blank connected patch in which the difference between the two principal curvatures is greatest. This favors working first in areas over which the principal directions are clearly defined, providing a stable seed for the synthesis of the other patches.

The most computationally expensive part of the algorithm is the texture synthesis, which accounts for 96-99% of the running time. Partitioning and optimizing the patches takes only about 1-3 seconds for simple meshes such as the Venus and triceratops, up to 10-12 seconds for larger meshes such as the 70,000 triangle bunny.

The complexity of the synthesis algorithm is linear with respect to the number of pixels and practically independent from the number of polygons in the input mesh. Growing the weave texture on the Venus model in figure 2 took slightly less than 12 min. The goblet/glass block combination required about 20 minutes. These timings refer to a C++ implementation compiled with gcc[1] running on a standard linux (2.2.16) 933MHz Pentium III PC with a 32Mb GeForce 2 GTS. Table 1 provides a detailed comparison of the speeds of our method and other similar concurrently developed methods. Notice that although some of the other methods report faster execution times, they all involve the synthesis of significantly smaller amounts of texture. The relative efficiency of our method becomes clear when one normalizes for texture quantity.

	Our method	Wei & Levoy [40]	Turk [38]	Ying et al. [43]
Published time for texture synthesis, from a 64x64 sample	10m	82s (TSVQ) 695s (exaustive)	23m	10m (multires.) 3m (coherent)
Texels per vertices synthesized	1,000,000	25,000	256,000	400,000
Machine	P3-933MHz	P2-450MHz	R12K-360MHz	unknown
Estimated machine speed normalization factor	1	2	2.1	unknown
Estimated normalized texture synthesis rate	1600 pixels/s	600 vertices/s (TSVQ) 70 vertices/s (exaustive)	380 vertices/s	unknown

Table 1: A comparison of the speed of our method vs. similar concurrently developed methods.

4.1 Limitations

A major limitation of our method is its inability to capture low frequency texture features across several patches. Since our objective in developing this algorithm was to achieve a method for facilitating shape perception, we assumed that the scale of our desired texture pattern would be substantially finer than the scale of the shape features in our model. Thus our method is not as generic as the methods described in [40], [38] and [43]. However for high resolution textures our method provides substantial speedups over these algorithms.

As described in [8] and [10] all synthesis algorithms based on the "one-pixel-at-a-time" approach are susceptible to "catastrophic failure", in which the algorithm falls into the wrong part of the search space and "grows garbage". Our algorithm is no exception and will occasionally fail and produce undesirable results across all or part of a patch. Depending on the texture, we find that 0-3% of the patches typically contain some synthesis errors. Unfortunately, areas as small as 5x5 pixels (in a 128x128 pixel patch) are easily noticed. The difficulties in automatically detecting such small areas complicate efforts to implement a mechanism for automatic correction. Our current implementation allows the user to interactively select and re-synthesize individual unaesthetic patches after the main automatic synthesis process has finished. Figs 9-11 in this paper show models in which parts of the texture were resynthesized across one or more patches. Figs 2, 5 and 7 show results that were obtained without any such postprocessing

Another limitation that bears mentioning is that we found that the large amounts of texture generated by the synthesis caused problems for certain machine architectures, specifically those that had hard built-in limitations on the amount of texture memory that could be used. When a high level of detail is desired, without any possibility for pattern repetition, the total amount of texture required to cover the mesh can easily exceed the total size of the texture memory. All of the machines that did not allow the storage of textures in main memory failed to display the more complex models (e.g. the crocodile skinned triceratops). Using a texture atlas similar to the one described in [33] might help reduce the texture memory usage, but would incur the cost of incorrect mipmapping and having to store texture coordinates. In our current implementation each patch is stored as a single texture, producing a texture memory waste

[1] optimization flags -O2 -funroll-loops -fstrict-aliasing -march=i686

of up to 25% to 50% depending on the model. However thanks to the OpenGL texture compression extensions, all of the models presented in this paper can maintain interactive frame rates on our standard PC, even in cases where the amount of uncompressed texture exceeds 100Mb.

The method that we have proposed is currently designed to be applied to static models, and we have not thought about how to extend it to the case of deforming animated objects. Modifications to make the texture synthesis process deterministic are an obvious first step toward satisfying this requirement, but it is not immediately clear how one could guarantee that independently synthesized patterns will not differ profoundly from frame to frame as the mesh defining the object is globally deformed.

To achieve good results with the shape-following textures, the direction field must be band-limited: for any given texture there is a maximum spatial frequency that can be followed in the synthesis process and still produce correct results. Nevertheless we found that the method did a good job at singular points on the brick-textured lava lamp object, as can be seen in figure 11. Figures 12a and 12b provide further examples of the behaviour of our method near singularities.

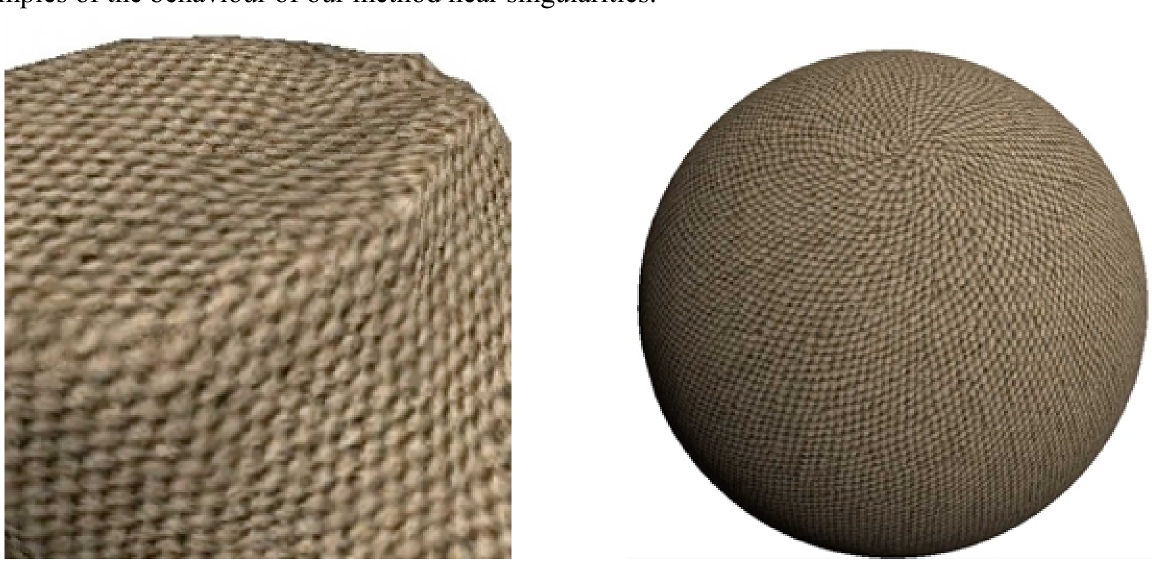


Figure 12: Left: A zoomed-in view of the singular point in the “up” directional field on the Venus model shown in figure 2. Right: A singular point on a sphere.

5. Applications And Future Work

There are many promising applications for this system and many directions for future work. One of the most interesting of these is multi-texturing. On a per-pixel basis it is possible to change not only the direction of the synthesized texture but even the texture itself according to any arbitrary function. Figures 13 and 14 are made using the illumination equation and two and four different textures respectively, each one with 63 rotations. These models, like all of the others in this paper, can be displayed at interactive frames rates on our standard PC.

The multi-texturing methods described in this paper have the potential to be useful for important applications in scientific visualization, for example in encoding a scalar distribution using texture type variations across an arbitrary domain in 2D or 3D. Other direction fields, such as gradient descent, hold promise for different applications, such as non-photorealistic rendering of terrain models (esp. in the case when it is desired to see through the surface). The methods that we have proposed can also be used for the visualization of scientifically computed vector fields over surfaces. An intriguing possible use for an extension of this work is in defining texture mixtures.

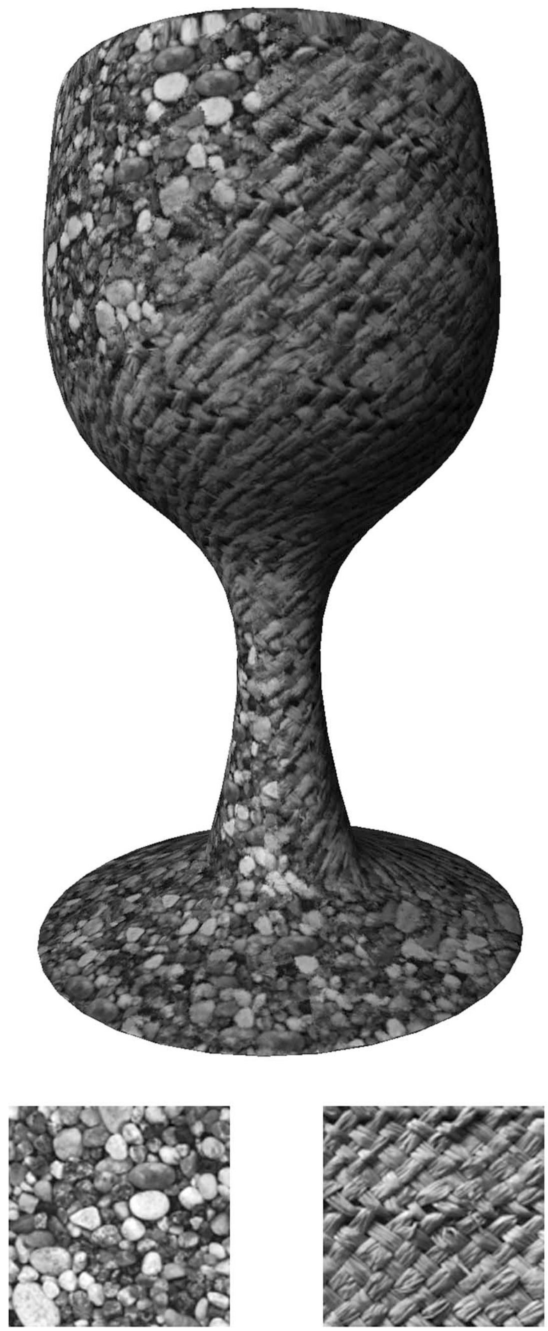


Figure 13: A demonstration of multi-texturing, in which the search for matches is performed within an array of different texture types. The texture type index can be defined by any function. In this example, we used the illumination function. In a real application one would want to use something more meaningful, such as soil type over a topographical terrain model.

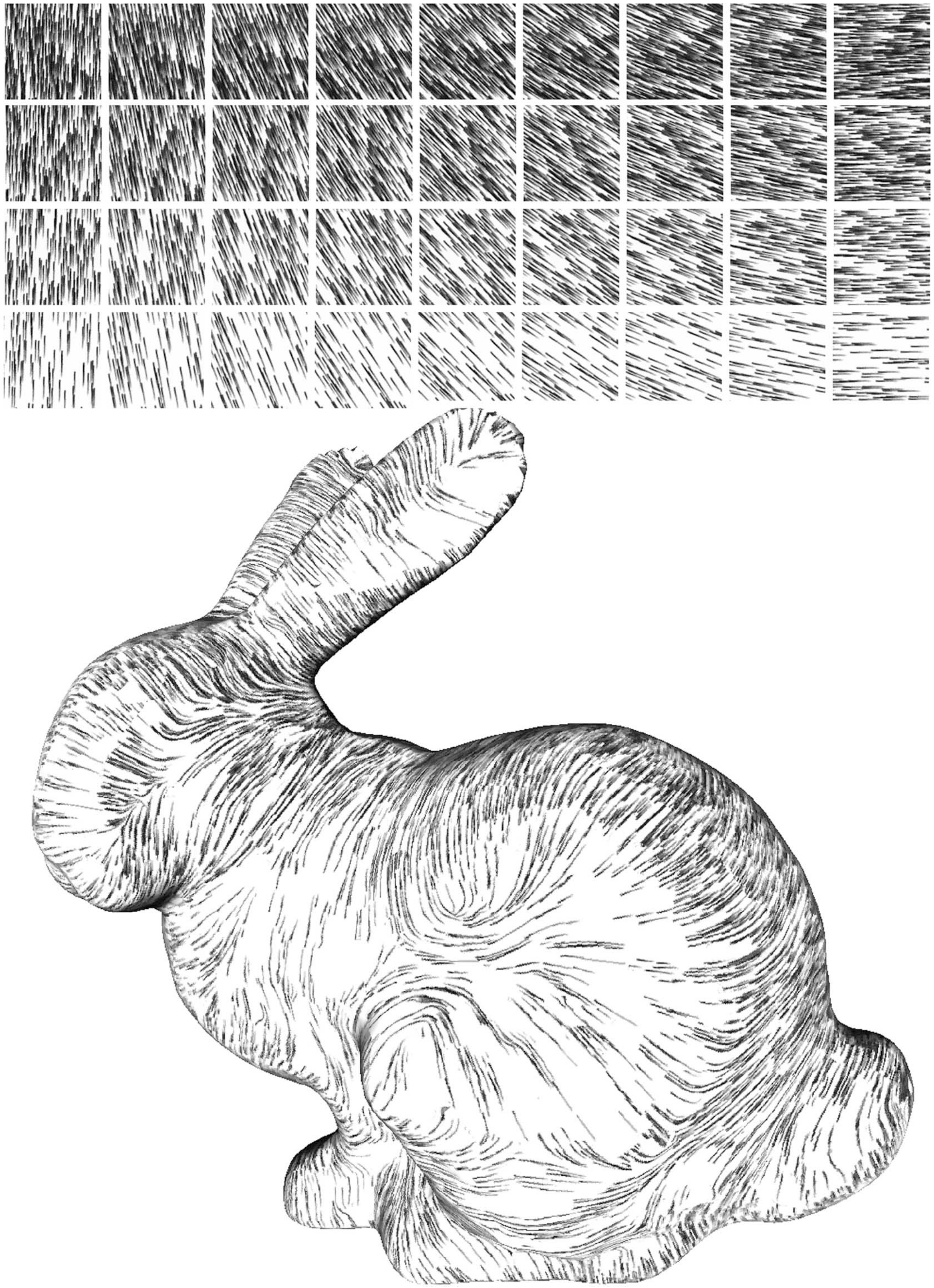


Figure 14: Multiple textures containing lines of different widths applied to an automatically-defined smooth vector field approximating the first principal direction over the Stanford bunny. Indexing along the dimension of varying stroke density was done as a function of the illumination.

6. Conclusions

In this paper we describe a novel surface texturing algorithm with a number of useful applications. Our described method for automatically synthesizing a desired texture pattern in a controlled orientation over the surface of an arbitrary object not only has potential applications in computer graphics, where it provides a simple and efficient solution to the classic problem of fitting a planar pattern to a non-developable surface in a way that minimizes both discontinuity and distortion. More importantly, it also has important potential applications in visualization, where it provides a means for texturing an arbitrary doubly curved surface with an arbitrary anisotropic pattern such that the orientation of the pattern follows a specified directional field over the surface at a per pixel level. With this capability it becomes possible to more thoroughly and directly investigate the effects on shape perception of a broader range of texture characteristics including orientation, and to come closer to answering the question of how best to design and apply a texture that can facilitate the accurate and intuitive perception of a surface's 3D shape.

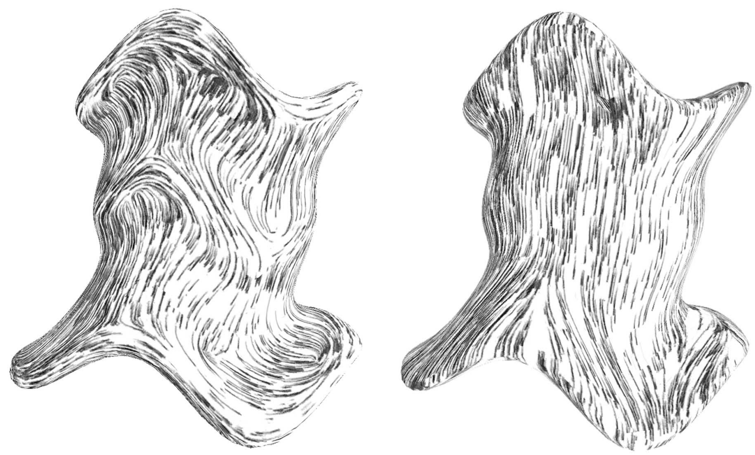


Figure 15: A final demonstration of the effect of texture orientation on shape perception — pen-and-ink style texture synthesized on an unfamiliar surface following the second principal direction (left) vs. a constant 'up' direction (right).

7. Acknowledgments

This work was supported by a University of Minnesota Grant-in-Aid of Research, Artistry and Scholarship. Additional support was provided by an NSF Presidential Early Career Award for Scientists and Engineers (PECASE) 9875368, NSF CAREER award CCR-9873670, ONR-N00014-97-2-0509, an Office of Naval Research Young Investigator Award, Presidential Early Career Award for Scientists and Engineers (PECASE) Navy/N00014-98-1-0631, and the NSF Learning and Intelligent Systems Program.

8. References

[1] Nur Arad and Gershon Elber. Isomeric Texture Mapping for Free-form Surfaces, *Computer Graphics Forum*, *1997*, **16**(5): 247– 256.

[2] Michael Ashikmin. Synthesizing Natural Textures. *Proceedings of the 2001 Symposium on Interactive 3D Graphics*, March 2001, pp. 217 – 226.

[3] Chakib Bennis, Jean-Marc Vézien and Gérard Iglésias. Piecewise Surface Flattening For Non-Distorted Texture Mapping, *Proceedings of SIGGRAPH 91*, pp. 237 – 246.

[4] Marcelo Bertalmio, Guillermo Sapiro, Li-Tien Cheng and Stanley Osher. A Framework for Solving Surface Differential Equations for Computer Graphics Applications, UCLA-CAM report 00-43, December 2000 (www.math.ucla.edu).

[5] Phil Brodatz. Textures: A Photographic Album for Artists and Designers, Dover Publications, 1966.

[6] Jeremy S. De Bonet. Multiresolution Sampling Procedure for Analysis and Synthesis of Texture Images, *Proceedings of SIGGRAPH 97*, pp. 361 – 368

[7] Bruce G. Cumming, Elizabeth B. Johnston and Andrew J. Parker. Effects of Different Texture Cues on Curved Surfaces Viewed Stereoscopically, *Vision Research*, 1993, **33**(5/6): 827 – 838.

[8] J.-M. Dischler, D. Ghazanfarpour and R. Freydier. Anisotropic Solid Texture Synthesis Using Orthogonal 2D Views, Computer Graphics Forum, **17**(3), September 1998.

[9] Alexei A. Efros and Thomas K. Leung. Texture Synthesis by Non-Parametric Sampling, *Proceedings of the International Conference on Computer Vision*, vol. 2, 1999, pp. 1033 – 1038.

[10] Alexei A. Efros and William T. Freeman. Image Quilting for Texture Synthesis and Transfer, *Proceedings of ACM SIGGRAPH 2001*, pp. 341-346.

[11] James A. Ferwerda, Sumanta N. Pattanaik, Peter Shirley and Donald P. Greenberg. A Model of Visual Masking for Computer Graphics, *Proceedings of SIGGRAPH 97*, pp. 143 – 152.

[12] Ahna Girshick, Victoria Interrante, Steve Haker and Todd LeMoine. "Line Direction Matters: an Argument for the use of Principal Directions in Line Drawings", *Proceedings of the First International Symposium on Non-Photorealistic Animation and Rendering*, pp. 43 – 52, June 2000.

[13] Jack Goldfeather. "Understanding Errors in Approximating Principal Direction Vectors", TR01-006, Dept. of Computer Science and Engineering, Univ. of Minnesota, Jan. 2001.

[14] Gabriele Gorla, Victoria Interrante and Guillermo Sapiro. Growing Fitted Textures, *IMA preprint 1748*, Feb. 2001.

[15] Steven Haker, Sigurd Angenent, Allen Tannenbaum, Ron Kikinis, Guillermo Sapiro and Michael Halle. Conformal Surface Parameterization for Texture Mapping, *IEEE Transactions on Visualization and Computer Graphics*, **6**(2), April/June 2000, pp. 181 – 189.

[16] Pat Hanrahan and Paul Haeberli. Direct WYSIWYG Painting and Texturing on 3D Shapes, *Proceedings of SIGGRAPH 90*, pp. 215 – 223.

[17] David J. Heeger and James R. Bergen. Pyramid-Based Texture Analysis/Synthesis, *Proceedings of SIGGRAPH 95*, pp. 229 – 238.

[18] Victoria Interrante, Henry Fuchs and Stephen Pizer. Conveying the 3D Shape of Smoothly Curving Transparent Surfaces via Texture, *IEEE Computer Graphics and Applications*, 1997, **3**(2): 98 – 117.

[19] Victoria Interrante. Illustrating Surface Shape in Volume Data via Principal Direction-Driven 3D Line Integral Convolution, *Proceedings of SIGGRAPH 97,* pp. 109 – 116.

[20] Victoria Interrante and Sunghee Kim. Investigating the Effect of Texture Orientation on the Perception of 3D Shape, *Human Vision and Electronic Imaging VI*, SPIE **4299**, January 2001, pp. 330 – 339.

[21] Victoria Interrante, Sunghee Kim and Haleh Hagh-Shenas. Conveying 3D Shape with Texture: Recent Advances and Experimental Findings, *Human Vision and Electronic Imaging VII*, SPIE **4662**, January 2002, to appear.

[22] David C. Knill. Contour into Texture: The Information Content of Surface Contours and Texture Flow, *Journal of the Optical Society of America, A*, **18**(1), January 2001, pp. 12-35.

[23] Bruno Lévy and Jean-Laurent Mallet. Non-distorted Texture Mapping for Sheared Triangulated Meshes, *Proceedings of SIGGRAPH 98*, pp. 343 – 352.

[24] Andrea Li and Qasim Zaidi. Perception of Three-Dimensional Shape From Texture is Based on Patterns of Oriented Energy, *Vision Research*, **40**(2), January 2000, pp. 217 – 242.

[25] Jérôme Maillot, Hussein Yahia and Anne Verroust. Interactive texture mapping, *Proceedings of SIGGRAPH 93*, pp. 27 – 34

[26] Pascal Mamassian and Michael S. Landy. Observer Biases in the 3D Interpretation of Line Drawings, *Vision Research*, **38**(18), September 1998, pp. 2817 – 2832.

[27]Alan P. Mangan and Ross T. Whitaker. Partitioning of 3D Surface Meshes using Watershed Segmentation, *IEEE Transactions on Visualization and Computer Graphics,* **5**(4), pp, 308-321.

[28]Fabrice Neyret and Marie-Paul Cani. Pattern-Based Texturing Revisited, *Proceedings of SIGGRAPH 99*, pp. 235 – 242.

[29]Darwyn R. Peachey. Solid texturing of Complex Surfaces, *Proceedings of SIGGRAPH 85*, pp. 279–286.

[30]Ken Perlin. An Image Synthesizer, *Proceedings of SIGGRAPH 85*, pp. 287 – 296.

[31]Dan Piponi and George Borshukov. Seamless Texture Mapping of Subdivision Surfaces by Model Pelting and Texture Blending, *Proceedings of SIGGRAPH 2000*, pp. 471 – 478.

[32]Javier Portilla and Eero P. Simoncelli. A Parametric Texture Model Based on Joint Statistics of Complex Wavelet Coefficients, *International Journal of Computer Vision*, **40**(1), October 2000, pp. 49 – 70.

[33]Emil Praun, Adam Finkelstein and Hughes Hoppe. Lapped Textures, *Proceedings of SIGGRAPH 2000*, pp. 465 – 470.

[34]Pedro V. Sander, John Snyder, Steven J. Gortler and Hughes Hoppe. Texture Mapping Progressive Meshes, *Proceedings of ACM SIGGRAPH 2001*, pp. 409-416.

[35]Kent A. Stevens. The Visual Interpretation of Surface Contours, *Artificial Intelligence*, **17**(1-3), August 1981, pp. 47 – 73.

[36]James T. Todd and Robin A. Akerstrom. Perception of Three-Dimensional Form from Patterns of Optical Texture, *Journal of Experimental Psychology; Human Perception and Performance*, **13**(2), 1987, pp. 242 – 255.

[37]Greg Turk. Generating Textures for Arbitrary Surfaces Using Reaction-Diffusion, *Proceedings of SIGGRAPH 91*, pp. 289 – 298.

[38]Greg Turk. Texture Synthesis on Surfaces, *Proceedings of ACM SIGGRAPH 2001*, pp. 347 – 354.

[39]Li-Yi Wei and Marc Levoy. Fast Texture Synthesis Using Tree-structured Vector Quantization, *Proceedings of SIGGRAPH 2000*, pp. 479 – 488.

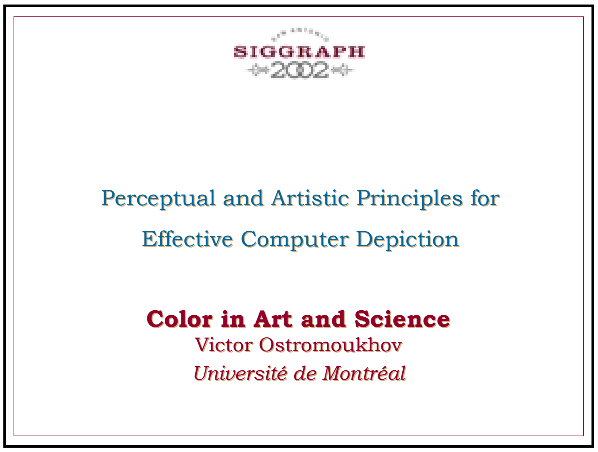
[40]Li-Yi Wei and Marc Levoy. Texture Synthesis Over Arbitrary Manifold Surfaces, *Proceedings of ACM SIGGRAPH 2001*, pp. 355 – 360.

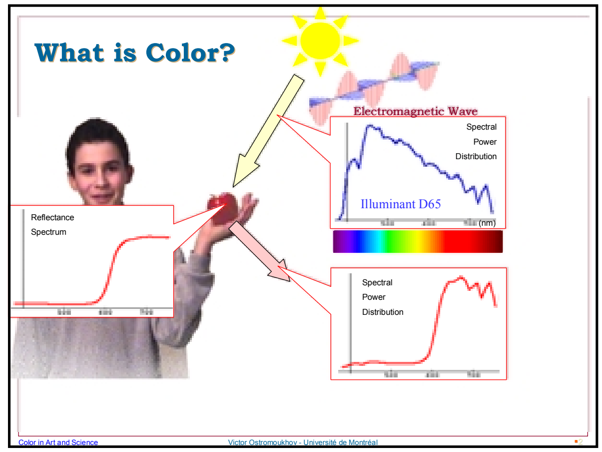
[41]Andrew Witkin and Michael Kass. Reaction-Diffusion Textures, *Proceedings of SIGGRAPH 91*, pp. 299– 308.

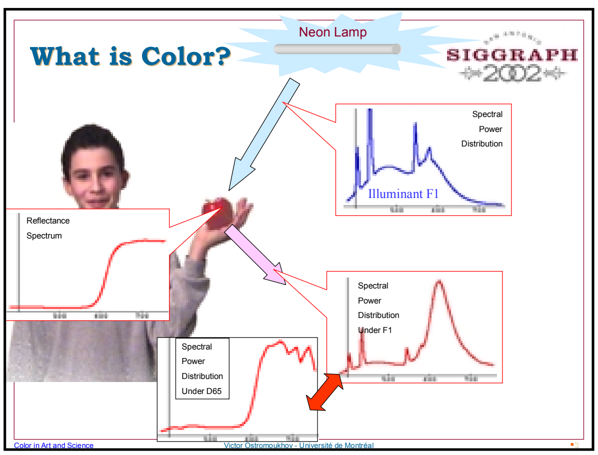
[42]Steven Worley. A Cellular Texture Basis Function, *Proceedings of SIGGRAPH 96*, pp. 291– 294.

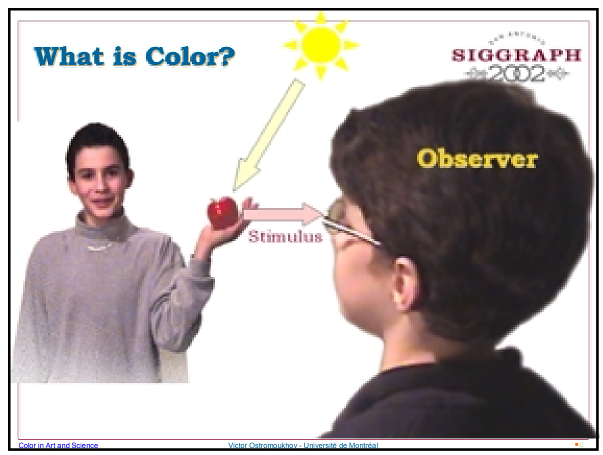
[43]Lexing Ying, Aaron Hertzmann, Henning Biermann, and Denis Zorin. Texture and Shape Synthesis on Surfaces, *Proceeedings of the 12th Eurographics Workshop on Rendering*, June 2001.

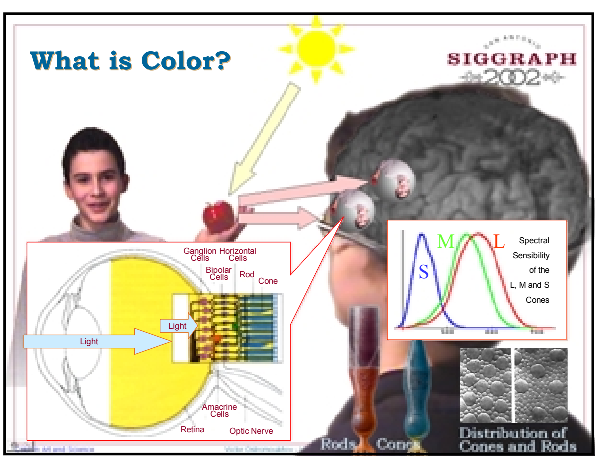
[44]Song Chun Zhu, Ying Nian Wu and David Mumford. Filters, Random Fields and Maximum Entropy (FRAME): Towards a Unified Theory for Texture Modeling, *International Journal of Computer Vision*, **27**(2), March 1998, pp. 107 – 126.

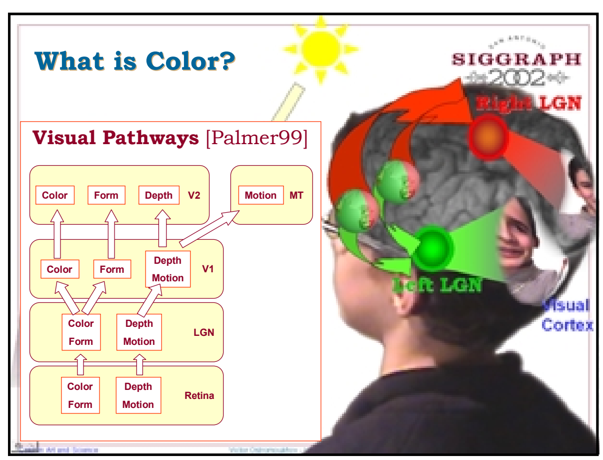

SIGGRAPH 2002
Perceptual and Artistic Principles for
Effective Computer Depiction
Color in Art and Science
Victor Ostromoukhov
Université de Montréal


What is Color?
Electromagnetic Wave
Spectral Power Distribution
Illuminant D65
(nm)
Reflectance Spectrum
Spectral Power Distribution
Color in Art and Science
Victor Ostromoukhov - Université de Montréal


What is Color?
Neon Lamp
SIGGRAPH 2002
Spectral Power Distribution
Illuminant F1
Reflectance Spectrum
Spectral Power Distribution Under F1
Spectral Power Distribution Under D65
Color in Art and Science
Victor Ostromoukhov - Université de Montréal


What is Color?
SIGGRAPH 2002
Observer
Stimulus
Color in Art and Science
Victor Ostromoukhov - Université de Montréal


What is Color?
SIGGRAPH 2002
Ganglion Cells
Horizontal Cells
Bipolar Cells
Rod
Cone
Light
Light
Amacrine Cells
Retina
Optic Nerve
M
L
S
Spectral Sensibility of the L, M and S Cones
Rods
Cones
Distribution of Cones and Rods


What is Color?
SIGGRAPH 2002
Visual Pathways [Palmer99]
Color
Form
Depth
V2
Motion
MT
Color
Form
Depth
Motion
V1
Color
Form
Depth
Motion
LGN
Color
Form
Depth
Motion
Retina
Right LGN
Left LGN
Visual Cortex

What is Color?

- Color is a Subjective Feeling Produced in Our Brain
- Physical Stimuli are Defined by Interactions Between Light and Objects
- Physical Stimuli are Processed in
  - Eye's Optics and Retina
  - Low Vision: LGN, Primary Cortex
  - Higher Vision Brain Areas
- Color Vision is a Complex, Active Process

Outline

SIGGRAPH 2002

- Milestones
- Scientific Approach:
  Experimental Psychology & Neuroscience
- Technological Approach:
  Colorimetry
- Artistic Approach
- Conclusions

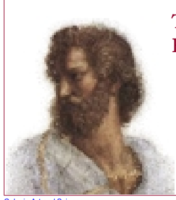
Two Great Western Traditions

SIGGRAPH 2002

Time

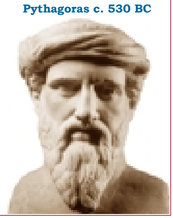
Antiquity — Newton's Time (XVII-XVIII cent) — XIX century — XX century

Aristotle c. 340 BC



Two Great Traditions: Rational and Mystical

Pythagoras c. 530 BC



Rational Tradition

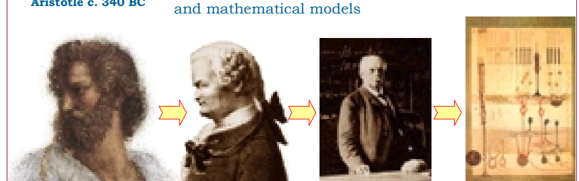
SIGGRAPH 2002

Time

Antiquity — Newton's Time (XVII-XVIII cent) — XIX century — XX century

Aristotle c. 340 BC

Rational approach based on measurements, analysis and mathematical models



Lambert — Von Helmholtz — Modern Neuroscience

Mystic (Artistic, Emotional) Tradition

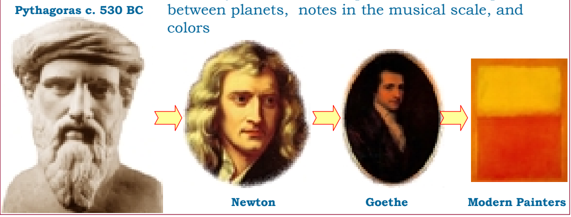
SIGGRAPH 2002

Time

Antiquity — Newton's Time (XVII-XVIII cent) — XIX century — XX century

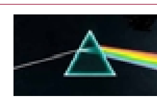
Pythagoras c. 530 BC

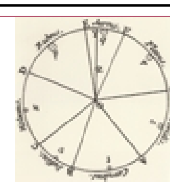
Harmony of the celestial spheres: relationship between planets, notes in the musical scale, and colors



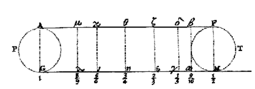
Newton — Goethe — Modern Painters

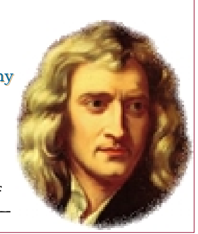
Newton 1672





- Belongs to Both Great Traditions
- Rational Tradition:
  - Discovery of Truly Primary Colors
  - Theory if Light Color Mixture
- Pythagorean Tradition:
  - Color is a Manifestation of the Harmony of the Celestial Spheres
  - Studied Relations between Colors and Music



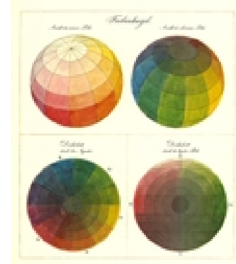


Lambert 1760

- Color pyramid
  - Base color triangle
  - Tertiary colors
- Basis for mathematical approach to colorimetry
- Intended for printers, dyers and craftsmen


Runge 1803

- Theorist and Romantic Painter

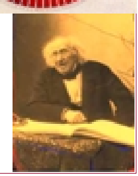




Goethe 1810

- Circular diagram:
  - primary colors (red, blue and yellow) alternate with secondary colors (orange, violet and green)
- Related aesthetic and emotional values of colors:
  - *"powerful", "gentle" and "radiant"*
- Exerted huge influence on generations of artists, scientists and philosophers


Chevreul 1839

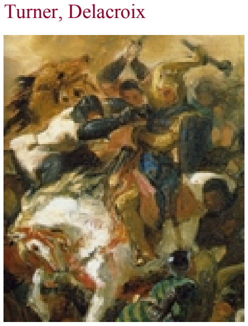
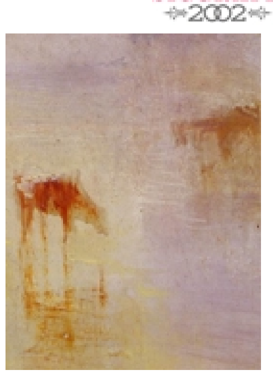
- Great Influence on XIX century painters: Delacroix, Impressionists, Post-Impressionists, Pointillists


Chevreul's chromatic circle

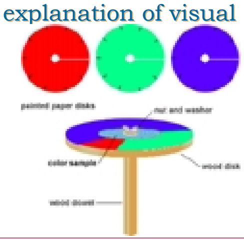
Romantic Painters of XIX century

Turner, Delacroix

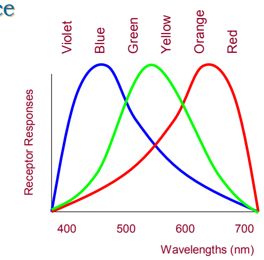
Maxwell 1867

- Origins of modern quantitative color measurement (Colorimetry)
- Experiences with Color Mixtures
- Psychological explanation of visual phenomena

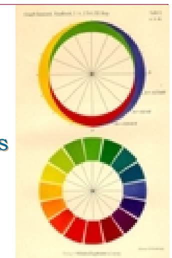


von Helmholtz 1859

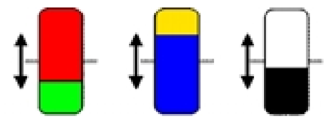
- Trichromatic color theory
- Improved color diagram: a curved line of spectral colors: ancestor of the CIE-XYZ color space

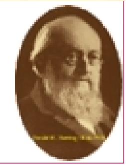


Hering 1874



- Father of the opponent process color theory
- Focus on psychological interpretation of colors



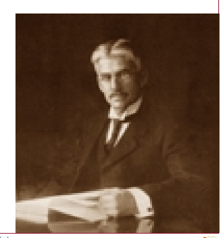


Impressionism, Post-Impressionism, Pointillism

Claude Monet

Munsell 1905

- System of visually uniform distribution of color patches
- Ancestor of CIE-LAB and CIE-LUV color spaces



Retinal Neural Interconnection

- 1900s

Ramón y Cajal
1906 Nobel Prize

Neural Codes

1930s

Hartline
1967 Nobel Prize

The horseshoe crab (Limulus polyphemus) *The Hero*

- Discovery of the mechanism of communication between the eye and the brain
- Lateral inhibition explained

Oscillogram of the electrical activity of a receptor unit in the lateral eye of *Limulus*
From Hartline's Nobel Prize Lecture, December 12, 1967

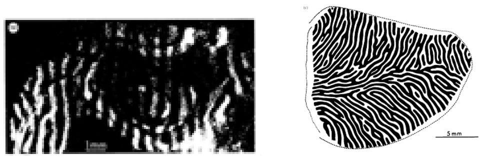
Milestones

SIGGRAPH 2002

1950-70s


Hubel Wiesel 1981 Nobel Prize

- Discovery of complex receptive fields in the visual cortex
- Neural mechanisms of visual information processing



Dark-field autoradiograph of striate cortex in an adult macaque, and reconstruction of layer Ivc.
From Hubel's Nobel Prize Lecture, 8 December 1981

Outline

SIGGRAPH 2002

- Milestones
- **Scientific Approach:**
  Experimental Psychology & Neuroscience
- Technological Approach:
  Colorimetry
- Artistic Approach
- Conclusions

Experimental Psychology & Neuroscience

SIGGRAPH 2002

Goals:

- Understand and explain basic phenomena in color vision
  - Basic sensory process
  - Low-level vision
  - Higher-level (category-based) vision
- Provide a comprehensive framework, compatible with other visual perception tasks

Experimental Psychology & Neuroscience

SIGGRAPH 2002

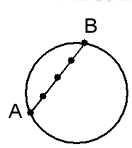
Basic Phenomena

- Light Mixture
- Color Blindness
- Color Afterimage
- Simultaneous Color Contrast
- Chromatic Adaptation
- Metamerism

Light Mixture

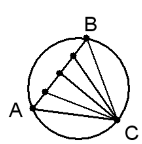
SIGGRAPH 2002

- Mixing two lights in varying proportions produces colors that lie along the straight line between them in color space



Mixing Two Lights

- Mixing three lights in varying proportions produces colors that lie within the triangle they define in color space

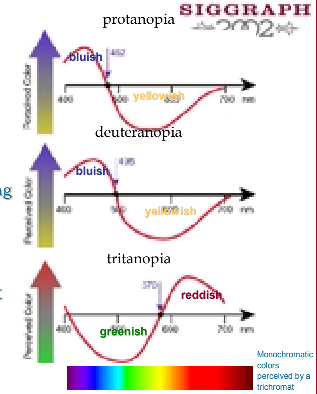


Mixing Three Lights

Color blindness

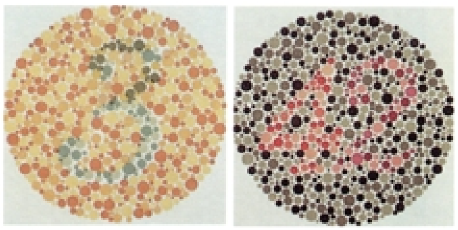
SIGGRAPH 2002

- 8% male, 0.6% female
- Genetic
- Dichromate (2% male)
  - One type of cone missing
    - L (protanope)
    - M (deuteranope)
    - S (tritanope)
- Anomalous trichromat
  - Shifted sensitivity

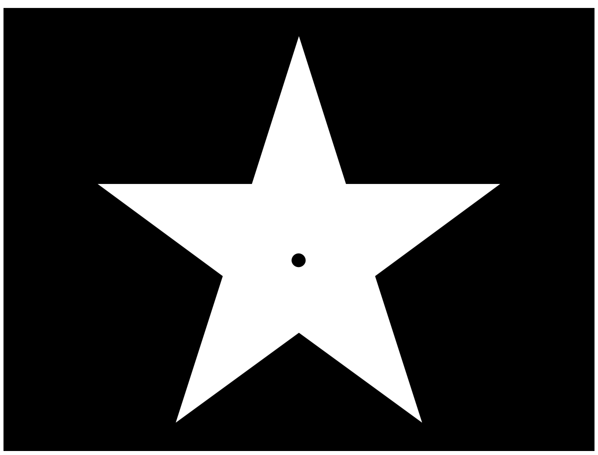


Color blindness test

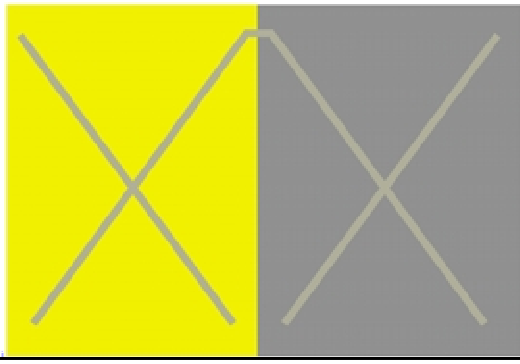
- A simple test permits to whether you are a trichromate or a dichromate
- People with anomalous cone pigmentation will fail to distinguish background from foreground



Color blind impressions

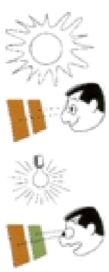


Simultaneous Color Contrast



Chromatic Adaptation

Metamerism

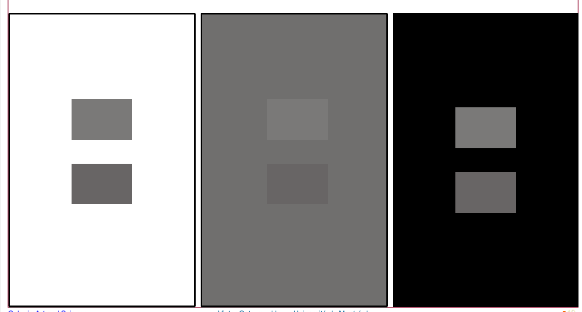


- Illuminant metamerism: two patches that appear identical under one illuminant (sunlight) appear different when lighting conditions change (incandescent light)
- Observer metamerism: two patches that appear identical for one observer may appear different for another observer, due to slightly different absorption curves of three types of cones (nobody's perfect!)

Illuminant Metamerism   Observer Metamerism

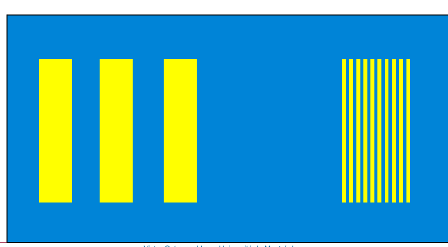
Crispening

- Increased sensitivity



Spreading

- Optical mix when spatial frequency increases
- But before fusion frequency
- Additive mix! (opposed to pigment mix)

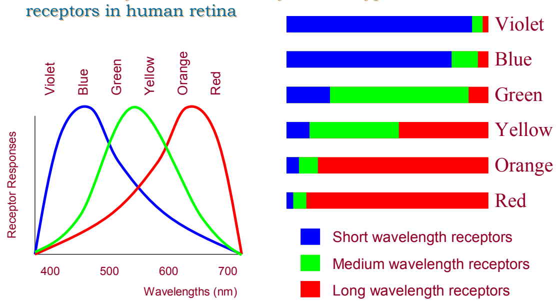


Color science: psychological approach

- Trichromatic Theory
- Opponent Process Theory
- Dual Process Theory

Trichromatic Theory

- Originally proposed by von Helmholtz on pure theoretical basis
- Confirmed by the later discovery of three types of color receptors in human retina

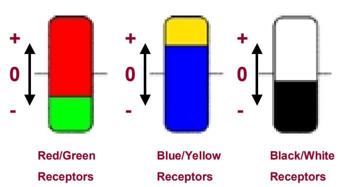


Trichromatic Theory

- Supposes 3 types of receptors
- Assumes 3 images are formed and transmitted to the brain
- Fails to explain several visual phenomena
  - Certain hues never appear together
  - Simultaneous contrast
  - Afterimage
  - Color vision deficiencies appear in pairs: blue-yellow and green-red

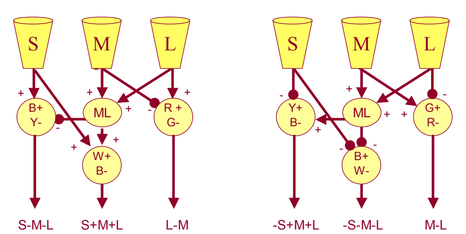
Opponent Process Theory

- Hypothesis of 3 types of receptors: Red/Green, Blue/Yellow, Black/White
- Explains well several visual phenomena



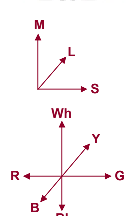
Color opponents wiring

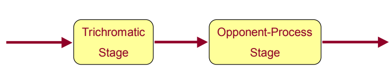
- Sums for brightness
- Differences for color opponents



Dual Process Theory

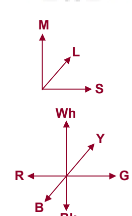
- The input is LMS
- The output has a different parameterization:
  - Light-dark
  - Blue-yellow
  - Red-green

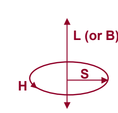




Color reparameterization

- The input is LMS
- The output has a different parameterization:
  - Light-dark
  - Blue-yellow
  - Red-green
- A later stage may reparameterize:
  - Brightness or Luminance or Value
  - Hue
  - Saturation





Challenges, Open Issues

- Understand High-Level Color Perception
- Build a Unified Color Vision Model
  that would include Retinal, Low- and High-Level Color Perception

Outline

- Milestones
- Scientific Approach:
  Experimental Psychology & Neuroscience
- **Technological Approach:**
  Colorimetry
- Artistic Approach
- Conclusions

Technological Approach: Colorimetry

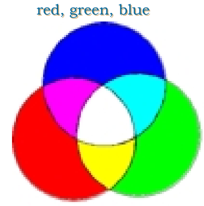
Goals:

- Provide scientific framework for color: objectivity, repetitiveness, measurability
- Provide an efficient support for color management tasks:
  - Color reproduction (e.g. printing) and displaying (e.g. TV, computer displays, video cameras)
  - Color management for inter-device imaging (e.g. network-based imaging, display-printer compatibility etc.)

Color synthesis

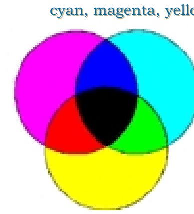
Additive

red, green, blue

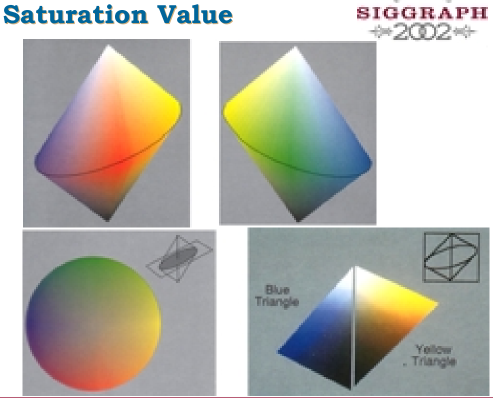


Subtractive

cyan, magenta, yellow

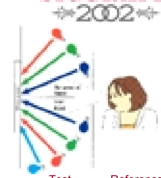


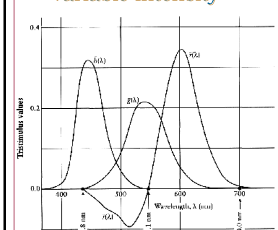
Hue Saturation Value

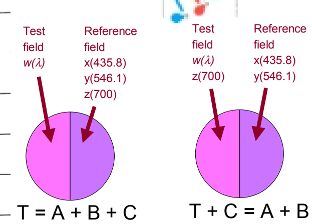


CIE: Color Matching

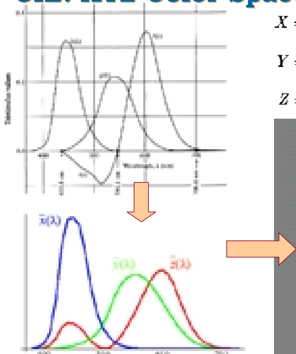
- Match a pure spectral test field *w(λ)* with a reference field: a mixture of red *z(700)*, green *y(546.1)* and blue *x(435.8)* of variable intensity



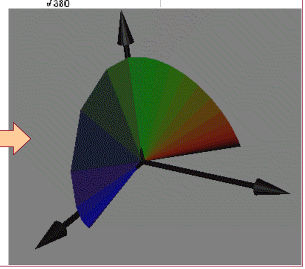




CIE: XYZ Color Space



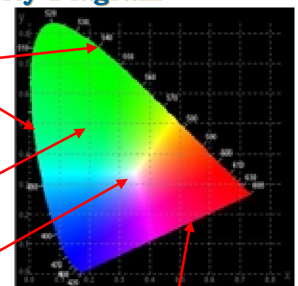
$$X = \int_{380}^{780} C(\lambda)\bar{x}(\lambda)d\lambda$$
$$Y = \int_{380}^{780} C(\lambda)\bar{y}(\lambda)d\lambda$$
$$Z = \int_{380}^{780} C(\lambda)\bar{z}(\lambda)d\lambda$$



SIGGRAPH 2002

The Colors in the Chromaticity Diagram

SIGGRAPH 2002



Spectrally pure colors (monochromatic) on the contour

Visible spectrum

Neutral illuminant white

Non-spectral colors (purples and magentas) no dominant wavelength

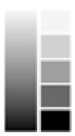
$$x = \frac{X}{X+Y+Z} \quad y = \frac{Y}{X+Y+Z} \quad z = \frac{Z}{X+Y+Z} = 1 - x - y$$

Perceptually Uniform Space Munsell

SIGGRAPH 2002

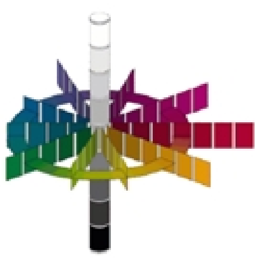
Hue

Value



Chroma

Munsell Color Space

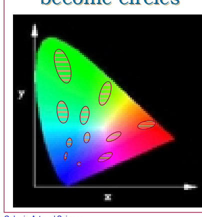


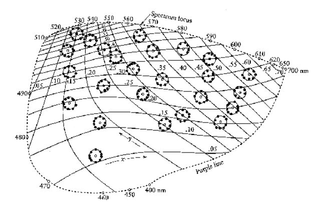
munsell.com

Perceptually Uniform Space: MacAdam

SIGGRAPH 2002

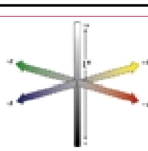
- In color space CIE-XYZ, the perceived distance between colors is not equal everywhere
- In perceptually uniform color space, Euclidean distances reflect perceived differences between colors
- MacAdam ellipses (areas of unperceivable differences) become circles



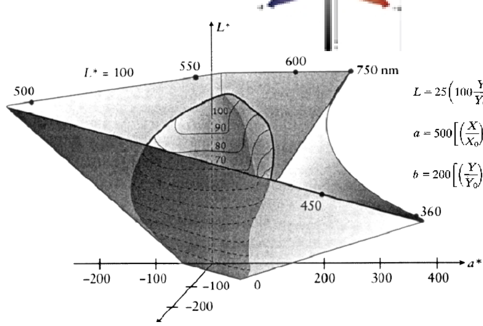


Source: [Wyszecki and Stiles '82]

CIE-LAB



SIGGRAPH 2002

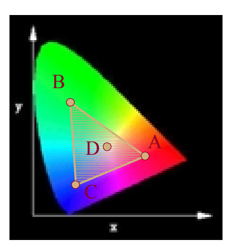


$$L = 25\left(100\frac{Y}{Y_0}\right)^{1/3} - 16$$
$$a = 500\left[\left(\frac{X}{X_0}\right)^{1/3} - \left(\frac{Y}{Y_0}\right)^{1/3}\right]$$
$$b = 200\left[\left(\frac{Y}{Y_0}\right)^{1/3} - \left(\frac{Z}{Z_0}\right)^{1/3}\right]$$

Source: [Wyszecki and Stiles '82]

Color Gamut

SIGGRAPH 2002

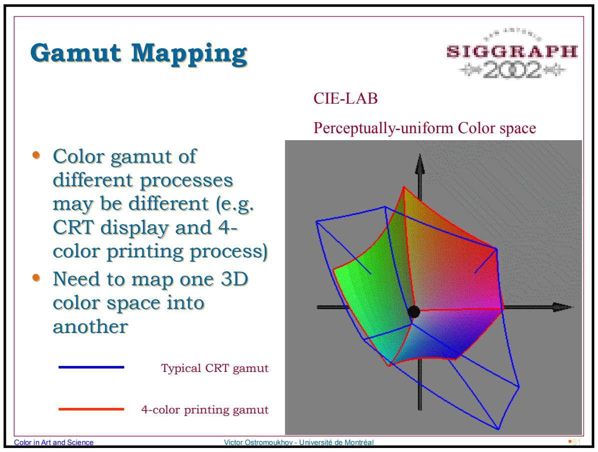


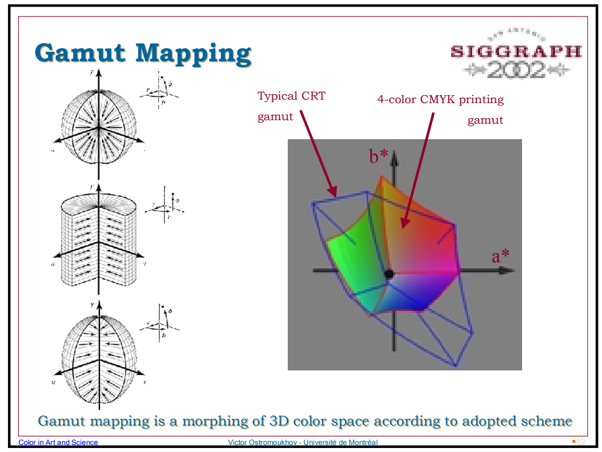
AB: all colors defined by a mix of colors A and B

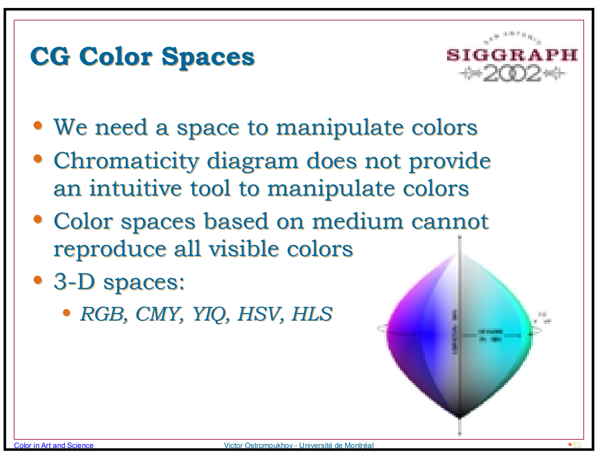
ABC: all colors defined by a linear combination of A, B, and C

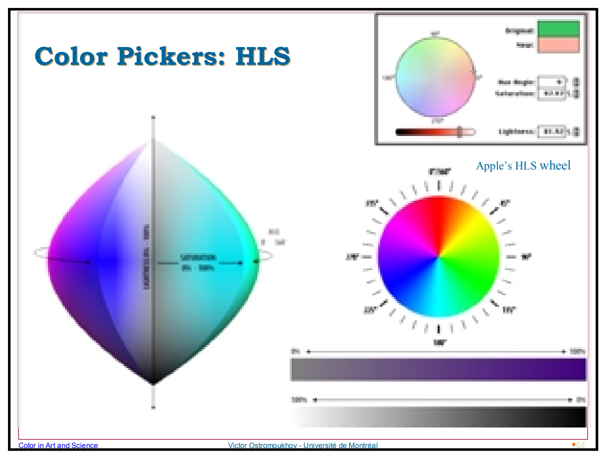
Three visible colors cannot produce all visible colors

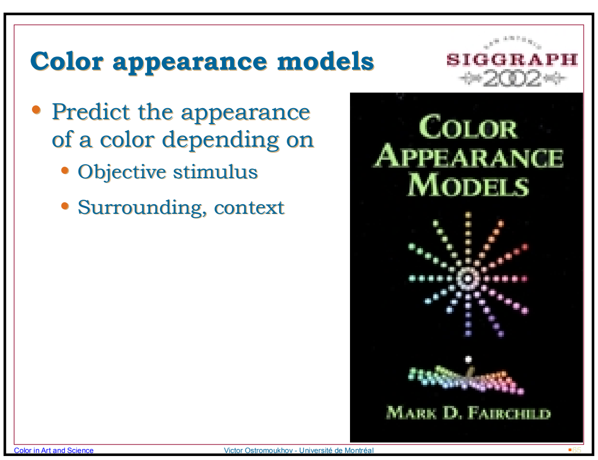
Gamut is formed by a convex polygon with primaries at the vertices

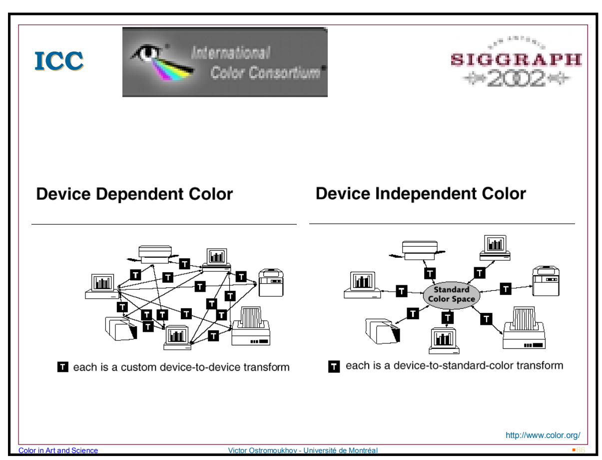

Gamut Mapping
SIGGRAPH 2002
CIE-LAB
Perceptually-uniform Color space
Color gamut of different processes may be different (e.g. CRT display and 4-color printing process)
Need to map one 3D color space into another
Typical CRT gamut
4-color printing gamut
Color in Art and Science
Victor Ostromoukhov - Université de Montréal


Gamut Mapping
SIGGRAPH 2002
Typical CRT gamut
4-color CMYK printing gamut
b*
a*
Gamut mapping is a morphing of 3D color space according to adopted scheme
Color in Art and Science
Victor Ostromoukhov - Université de Montréal


CG Color Spaces
SIGGRAPH 2002
We need a space to manipulate colors
Chromaticity diagram does not provide an intuitive tool to manipulate colors
Color spaces based on medium cannot reproduce all visible colors
3-D spaces:
RGB, CMY, YIQ, HSV, HLS
Color in Art and Science
Victor Ostromoukhov - Université de Montréal


Color Pickers: HLS
Apple's HLS wheel
Color in Art and Science
Victor Ostromoukhov - Université de Montréal


Color appearance models
SIGGRAPH 2002
Predict the appearance of a color depending on
Objective stimulus
Surrounding, context
COLOR APPEARANCE MODELS
MARK D. FAIRCHILD
Color in Art and Science
Victor Ostromoukhov - Université de Montréal


ICC
International Color Consortium
SIGGRAPH 2002
Device Dependent Color
Device Independent Color
Standard Color Space
each is a custom device-to-device transform
each is a device-to-standard-color transform
http://www.color.org/
Color in Art and Science
Victor Ostromoukhov - Université de Montréal

Challenges, Open Issues

SIGGRAPH 2002

- Multi-Platform, Multi-Device, Efficient and User-Friendly Color Management System
  e.g.: Digital-Camera → Computer → www → Printer
- Spectral Color Management Systems
- Operational Color Appearance Models

Outline

SIGGRAPH 2002

- Milestones
- Scientific Approach:
  Experimental Psychology & Neuroscience
- Technological Approach:
  Colorimetry
- **Artistic Approach**
- Conclusions

Artistic Approach

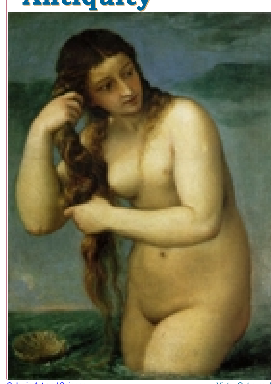
SIGGRAPH 2002

"Goals"

- Provide support for esthetical goals:
  - Support for forms, shapes, and texture of the depicted scene
  - Provide desired degree of realism/abstraction
  - To be visually appropriate, pleasant
  - Convey color harmony/expressiveness
  - Convey a feeling, an emotion, an impression
- Esthetical goals changed through ages

Antiquity

SIGGRAPH 2002



- None of Antique Greek Masterpieces Survived
- Exerted Huge Influence on Western Plastic Arts

Titian
Venus Anadyomene, 1520
Reconstructs a famous painting by Apelles

Middle Age

SIGGRAPH 2002

- Fine Control of the Process
- Notions of Optical Color Mixture
- Mastery of Color Contrast

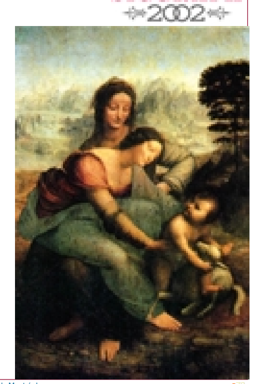
Byzantine Mosaic From Ravenna, VI century

The Scourging of Christ, Esslingen 1320

Leonardo 1510

SIGGRAPH 2002

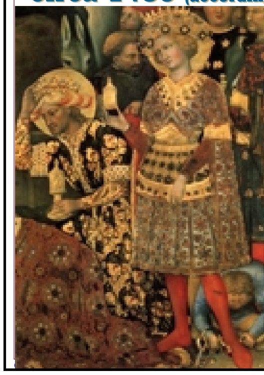
- Distinction between primary and secondary colors
- Effects of aerial attenuation of colors
- Simultaneous contrast



Renaissance Masters

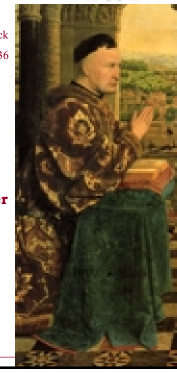
Fra Angelico, XV century

Breakthrough in Realism circa 1430 (according to Hockney)



Gentile da Fabriano
1423

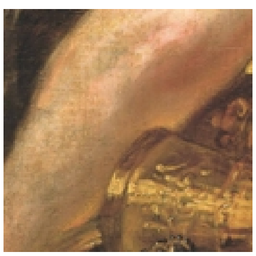
Before → After

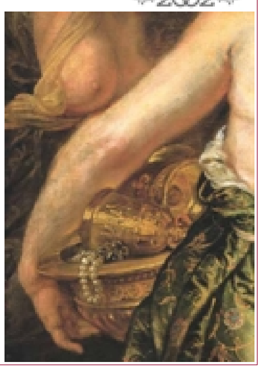


Jan Van Eyck
1436

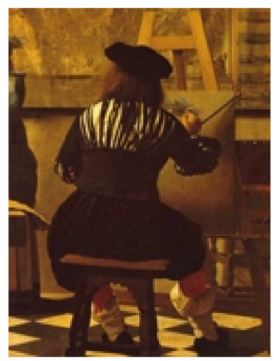
Rubens beg. of XVII century

- Mastery of Flesh Colors
- Glazes, Transparent and Opaque Colors

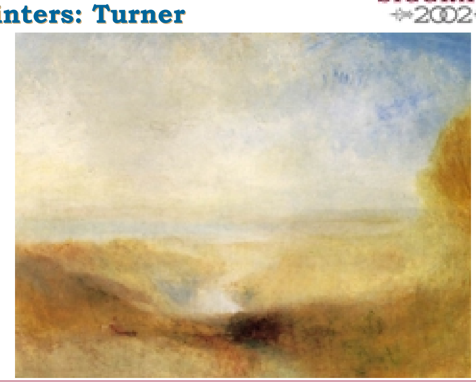




Vermeer mid-XVII century



XIX century Romantic Painters: Turner



XIX century Romantic Painters: Eugène Delacroix

- Explicit Usage of Opponent Colors
- Simultaneous Contrast

Impressionism

SIGGRAPH 2002

- Retinal Image: Painted What They Saw

Claude Monet

Post-Impressionism

SIGGRAPH 2002

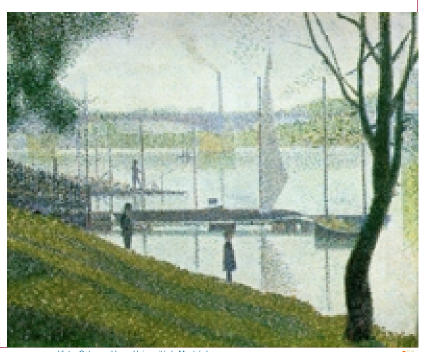
- Expressive Power of Complementary Colors

Vincent van Gogh

Pointillism

SIGGRAPH 2002

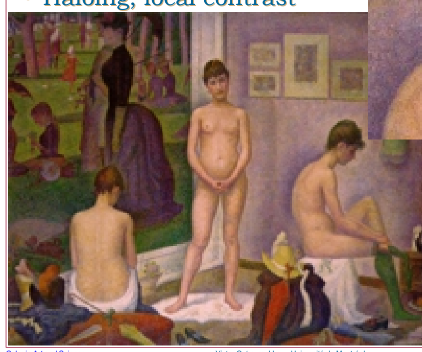
- Use of pure colors
- Reduced palette
- Additive rather than subtractive mixture

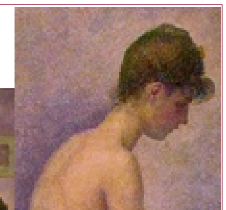


Georges Seurat

Pointillism

- Haloing, local contrast

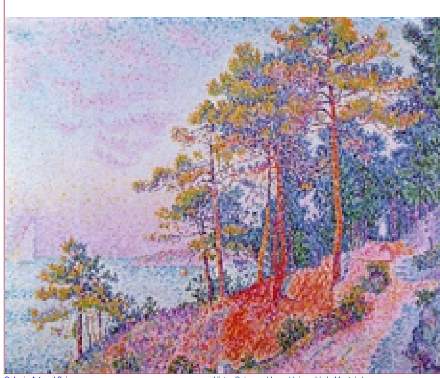




Georges Seurat

Divisionism

SIGGRAPH 2002



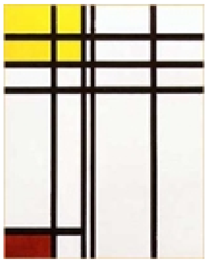
Paul Signac

Expressionism

SIGGRAPH 2002

Edvard Munch

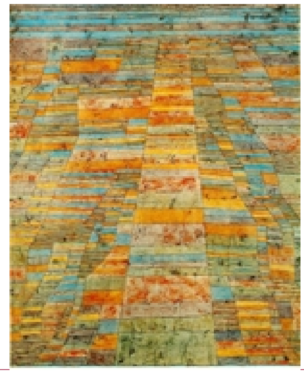
De Stijl



Mondrian
1921 Composition

Rietveld
1918 Red and Blue Chair

Bauhaus

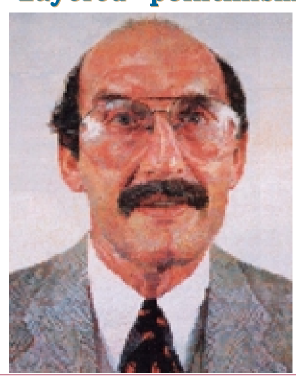


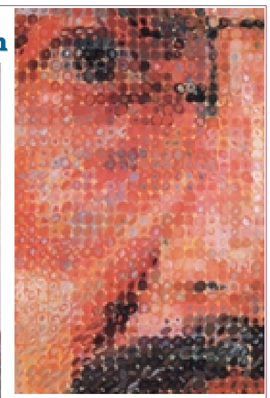
Paul Klee

Post-Modernism

Mark Rothko

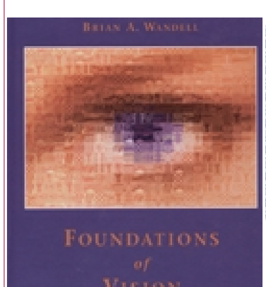
"Layered" pointillism

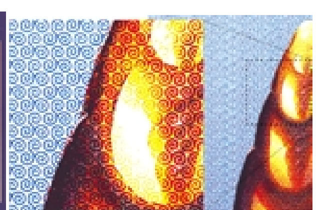




Chuck Close, *Stanley*, 1980-81

Photo-Mosaics, Artistic Halftoning





[Ostromoukhov 1999]

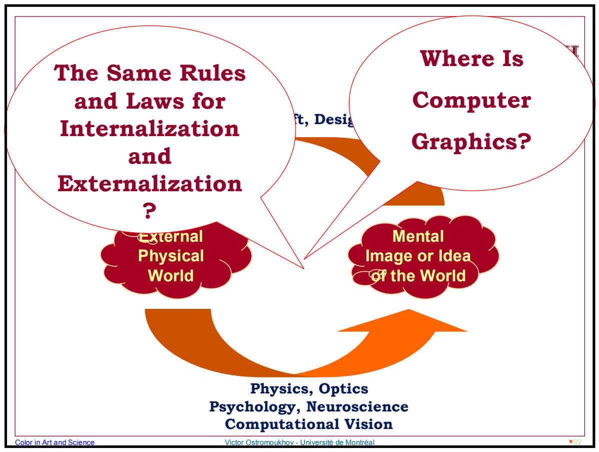
Outline

- Milestones
- Scientific Approach:
  Experimental Psychology & Neuroscience
- Technological Approach:
  Colorimetry
- Artistic Approach
- **Conclusions**

Conclusions

SIGGRAPH 2002

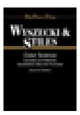
- Scientists and artists have developed their own techniques and interpretations of color
- Their influences were mutually beneficial
- Further development in neuroscience, experimental psychology and technology of color may be very beneficial for computer graphics
- Inversely, incoming computer graphics applications may stimulate new fundamental research in neuroscience and experimental psychology



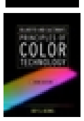
Selected Bibliography

SIGGRAPH 2002

- Colorimetry, Color Science



Color Science : Concepts and Methods, Quantitative Data and Formulae (Wiley Series in Pure and Applied Optics)
by Gunter Wyszecki, W. S. Stiles
John Wiley & Sons; ISBN: 0471399183
968 pages 2nd edition (July 2000)



Billmeyer and Saltzman's Principles of Color Technology, 3rd Edition
by Roy S. Berns, Fred W. Billmeyer, Max Saltzman
Wiley-Interscience; ISBN: 047119459X
304 pages 3 edition (March 31, 2000)

Color Appearance Models
by Mark D. Fairchild
Prentice Hall PTR; ISBN: 0201634643
448 pages (November 1997)

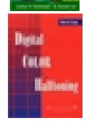
Selected Bibliography

SIGGRAPH 2002

- Colorimetry, Color Science

The Reproduction of Colour: Sixth Edition
by Dr. R.W.G. Hunt
Fountain Pr Ltd; ISBN: 0863433685
850 pages 6th edition (March 31, 2002)
Measuring Colour
by Dr. R.W.G. Hunt
Fountain Pr Ltd; ISBN: 0863433871
3rd edition (January 2001)

Colour Image Science : Exploiting Digital Media
by Lindsay W. MacDonald, Ronnier Luo
John Wiley & Sons; ISBN: 0471499277
300 pages (March 2002)

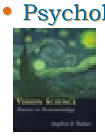


Color Technology for Electronic Imaging Devices (Vol PM28)
by Henry R. Kang
Society of Photo-optical Instrumentation Engineers; ISBN: 0819421081
384 pages (March 1997)
Digital Color Halftoning (Spie/IEEE Series)
by Henry R. Kang
Society of Photo-optical Instrumentation Engineers; ISBN: 0819433187
528 pages (November 1, 1999)

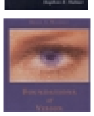
Selected Bibliography

SIGGRAPH 2002

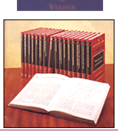
- Psychology, Neuroscience



Vision Science
by Stephen E. Palmer
MIT Press; ISBN: 0262161834
760 pages (May 7, 1999)



Foundations of Vision
by Brian A. Wandell
Sinauer Assoc; ISBN: 0878938532
(May 1995)

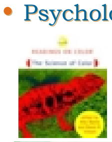


Perception of Colour (Vision and Visual Dysfunction)
by Peter Gouras (Editor)
CRC Press; ISBN: 0849375061
Vol 006 (April 1991)

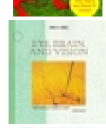
Selected Bibliography

SIGGRAPH 2002

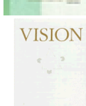
- Psychology, Neuroscience



Readings on Color - Vol. 1 & 2
by Alexander Byrne (Editor), David R. Hilbert (Editor), Alex Byrne
MIT Press; ISBN: 0262024241, 0262522314
317 pages Vol 1 (May 2, 1997)
465 pages Vol 2 (May 9, 1997)



Eye, Brain, and Vision (Scientific American Library, No 22)
by David H. Hubel
W H Freeman & Co; ISBN: 0716760096
Reprint edition (April 1995)

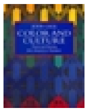


Vision : A Computational Investigation into the Human Representation and Processing of Visual Information
by David Marr
W H Freeman & Co.; ISBN: 0716715678
(September 1983)

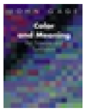
Selected Bibliography

SIGGRAPH 2002

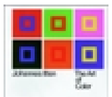
- Color and Art



Color and Culture : Practice and Meaning from Antiquity to Abstraction
by John Gage
University of California Press; ISBN: 0520222253
336 pages Reprint edition (August 1999)



Color and Meaning: Art, Science, and Symbolism
by John Gage
University of California Press; ISBN: 0520226119
320 pages (August 1, 2000)

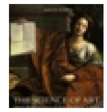


The Art of Color: The Subjective Experience and Objective Rationale of Color
by Johannes Itten
John Wiley & Sons; ISBN: 0471289280
160 pages Revised edition (December 1997)

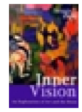
Selected Bibliography

SIGGRAPH 2002

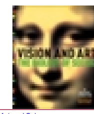
- Color and Art



The Science of Art : Optical Themes in Western Art from Brunelleschi to Seurat
by Martin Kemp
Yale Univ Pr; ISBN: 0300052413
Reprint edition (April 1992)



Inner Vision : An Exploration of Art and the Brain
by Semir Zeki
Oxford University Press; ISBN: 0198505191
240 pages illustrate edition (January 2000)

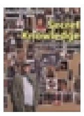


Vision and Art : The Biology of Seeing
by Margaret Livingstone, David H. Hubel
Harry N Abrams; ISBN: 0810904063
208 pages (May 2002)

Selected Bibliography

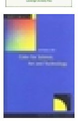
SIGGRAPH 2002

- Color and Art



Secret Knowledge: Rediscovering the Lost Techniques of the Old Masters
by David Hockney
Viking Press; ISBN: 0670030260
296 pages (October 25, 2001)

Goethe Contra Newton : Polemics and the Project for a New Science of Color
by Dennis L. Sepper
Cambridge Univ Pr (Short); ISBN: 0521342546
(March 1988)



Color for Science, Art and Technology
by Kurt Nassau (Editor)
North-Holland; ISBN: 0444898468
510 pages (December 1, 1997)

Acknowledgements

SIGGRAPH 2002

Thanks to Roy Berns, Frédo Durand, Eric Stollnitz, Pierre Poulin, and many web resources for providing access to their images

Perception of Pictures

Denis Zorin, New York University

One of the goals of image synthesis is to achieve realism in pictures. although most people have an intuitive notion of realism, it is hard to define explicitly. We call this intuitive notion *perceptual realism*. In computer graphics the most common solution is to define realism as photorealism. Photographs are often considered to be a standard of truth in images and the modeling of an idealized photographic process becomes the main task of the image synthesis. This solution has a number of advantages, but inevitably suffers from a fundamental flaw: the image synthesis algorithms are constructed and evaluated as models of a photographic process, while the output of these algorithms (pictures) is evaluated primarily by a perceptual process. If we use this approach, the perceptual quality of computer-generated images is inherently limited by that of photographs.

It is well known that an arbitrary photograph does not necessarily look good. Although photographers use some types of perceptual distortions occurring in photographs to achieve various artistic effects, in many cases distortions are undesirable and must be avoided. Avoiding these distortions by purely photographic means (choice of the viewing point or of the field of view) results in restrictions on the scenes that can be realistically reproduced in photographs. Computer generation of images is more flexible, and instead of faithfully imitating the photographic process, we can try to use perceptual principles directly: instead of modeling a camera we can redefine our working concept of realism formalizing some facts about picture perception. In this survey we discuss various facts about perception of pictures that can be used for this purpose.

There are three main sources of information about picture perception available to us: psychophysical research, art history and theory, and our everyday experience. Quite a few facts about perception appear to be so obvious that nobody ever states them explicitly or bothers to test them experimentally. We will see that some of these obvious facts are quite important (Section 1.3).

There is a considerable difference between our approach to the problems of perception and the approach that is typically used in psychophysical research. Our main goal is not to prove or disprove some particular theory of perception but to try to collect facts about perception that can be used for making better pictures. We cannot avoid making some theoretical assumptions altogether, but we try not to make too many; we describe our assumptions in Section 1.

Most experimental studies consider linear perspective images, but their results apply to any projection producing similar images. Whenever possible, we will emphasize two-dimensional characteristics

This is a revised version of Chapter 2 of [Zor95].

of images, rather than the characteristics of the central projection used for their construction. Linear perspective nevertheless is extremely important: it does not produce objectionable distortion in many cases. In Section 5 we discuss linear perspective in greater detail.

1 Assumptions and Nomenclature

Before we proceed with description of various observations about perception of pictures we need to define a variety of terms that we use and what type of pictures we are interested in.

1.1 Perceptual Nomenclature

We start with defining more or less precisely what we mean by a number of common terms. We use the word *quantity* to refer to numerical characteristics of three-dimensional objects and scenes represented in a picture, such as slant, size, distance or orientation. It is important to distinguish between true, apparent and judged quantities.

true A true quantity (size, slant, distance etc.) is the actual value for the object or scene represented in the picture.

apparent An apparent quantity is the value as perceived or deduced from the picture. Apparent quantity cannot be measured directly: it can be estimated from the judged quantity or from some other response (such as ball-throwing in [Smi61]).

judged A judged quantity is the value as reported by a subject; in a good experiment it is determined by the apparent measure, but can be biased by the mechanism of reporting or by the specified task. This bias, however, should be the same for identical apparent quantities.

The *perceptual space* for an observer is the model of the true space space that the observer infers from a picture. The perceptual space need not possess a consistent geometry in mathematical sense: the value of an apparent quantity can depend on the task, attention focus and many other factors. In most cases, the perceptual space of a picture is not connected to the real space around the observer. A possible reason for this is the scalability of the pictures: in most cases familiar objects in the pictures are smaller or larger than their real size, and they cannot be directly placed in the real space. There are several notable exceptions: the perceptual space of *trompe-l'oeil* pictures [Mil86] is merged with the real space; the *orientation* of objects in the perceptual space can be judged with respect to the viewer.

Most names of quantities (e.g. size, slant, distance) do not require explanation. The definition of *orientation* requires some comments.

orientation We shall distinguish between two type of orientation: relative to the viewer and relative to the other objects in the picture. We can estimate the angle between the viewing direction and some direction specified by an elongated object in a picture. This angle measures orientation with respect to the viewer. It is also possible to estimate the relative orientation, which is the angle between two directions in a picture.

The geometry of three-dimensional objects can be partially inferred from two-dimensional geometry of their images. Most terms that we use for two-dimensional geometry of images are just general geometric terms (e.g. angles, lines, curves, areas etc). A specific characteristic of images which is inherently two-dimensional, but cannot be described without referring to the three-dimensional object is *foreshortening.* For perspective projections we shall distinguish two types of foreshortening: *parallel* and *perpendicular*.

foreshortening If the size of the image of an object depends on its position in the scene, we shall call this effect *foreshortening*.

parallel foreshortening The decrease in the size of the image of an object when it is placed at increasing distances from the picture plane. It is also called *perspective convergence*.

perpendicular foreshortening Suppose it is possible to distinguish a feature of the object perpendicular to the projection plane and another feature which is parallel to the projection plane, such as faces in a cube or the lateral and frontal side of the head. The difference between the ratio of the sizes of these features and the ratio of the sizes of their images is the perpendicular foreshortening.

It is possible to extend these definitions to the case of non-perspective projections, if we can somehow define the distance to the projection plane (which need not coincide with the physical distance).

Finally, it is important to make a clear distinction between several related perceptual phenomena:

distortion Some images of familiar objects can be identified as *distorted*, if perceptual information in the picture is sufficient to identify the object with a high degree of confidence, yet some part of this information results in conclusions about the object that contradict experience. Looking at the example in Figure 1, one is reasonably confident that the depicted object is a sphere, because a sphere is more likely to be a part of architectural decoration than a tilted ellipsoid. At the same time, the shape of the image suggests that the object is not spherical, which results in a contradiction.

misperception Some images of objects can convey incorrect information about the objects, without causing distortion. If objects of certain type vary in size, some images result in an apparent size which is close to the true size, while others will result in a different apparent size. We will call the later case *misperception.* For example, the right monitor in Figure 2 appears to have an aspect ratio significantly greater than the actual one (no more than 1:1.4) and than that of the left monitor (the monitors are known to be identical).

illusion Some pictures under special conditions can be mistaken for the real scene that they represent. We shall call this condition *illusion.*

Figure 1: Wide-angle pinhole photograph taken on the roof of the Church of St. Ignatzio in Rome, classical example of perspective distortions from [Pir70].

1.2 Pictures

Next we describe more precisely what type of pictures we consider. It is also important to describe assumptions about viewing conditions –the perception of a picture depends on the position of the viewer with respect to the picture, presence or absence of apertures, frames, etc.

We are interested in pictures that are flat or nearly flat pictures on paper, a projection screen, or any other flat surface, observed binocularly, without any restrictions on the position of the head and without any special devices. Pictures of this type include book illustrations, photos, posters, screen projections of slides, motion pictures and pictures on computer displays. Excluded are stereograms of all types, pictures that are designed for observation through a fixed small aperture, pictures in head-mounted displays and anamorphic pictures.

Further, we are interested in pictures that are representations of three-dimensional objects. There are many different types of representation, from purely symbolic, like a verbal description or a *kanji* character, to a highly realistic photographic image which, when viewed from a correct position monocularly, can be confused for a real object. Linear perspective is used to some extent in most pictures of the latter type.

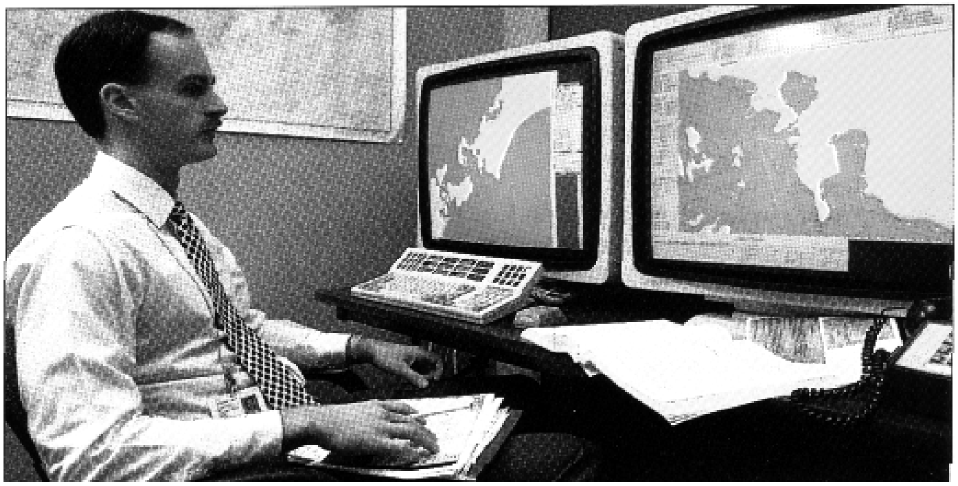


Figure 2: Photo from the article "Navigating Close to Shore" by Dave Dooling ("IEEE Spectrum", Dec. 1994), © 1994 IEEE, photo by Intergraph Corp. 92° viewing angle. The two monitors have the same size. But the left one appears to be wider (misperception).

Some authors have argued that essentially all forms of pictorial representation are based on convention [Arn54, Goo76]. For example, it has sometimes been claimed that linear perspective is a matter of convention and perspective images can be understood only within the context of a particular culture. It appears that such radical approach contradicts experimental evidence. Cross-cultural [Der91] and developmental [Hag76a, HJ78] studies indicate that no culture-dependent learning is required for adequate perception of perspective images. The existence of various phenomena in perception of pictures also demonstrate that there is a fundamental difference between the perception of most pictures and the perception of purely conventional representations, such as text.

The presence or absence of a conventional component in a picture is difficult to determine. In some cases (road signs, maps) it is easy to classify a drawing as almost purely conventional. We will be most interested in drawings that do not use a symbolic method of representation or use it in a very limited way. This includes a wide range of pictures (Figure 3).

1.3 Assumptions about Perception

Our assumptions can be most clearly formulated in terms of retinal images and internal representations. Our ability to recognize objects and textures that we have seen before is crucial both to perception of the three-dimensional world and pictures. We assume *existence of some internal representation of objects, and a process that compares information extracted from the retinal image with this representation.*

All visual information about our environment is contained in the images formed on the retina of

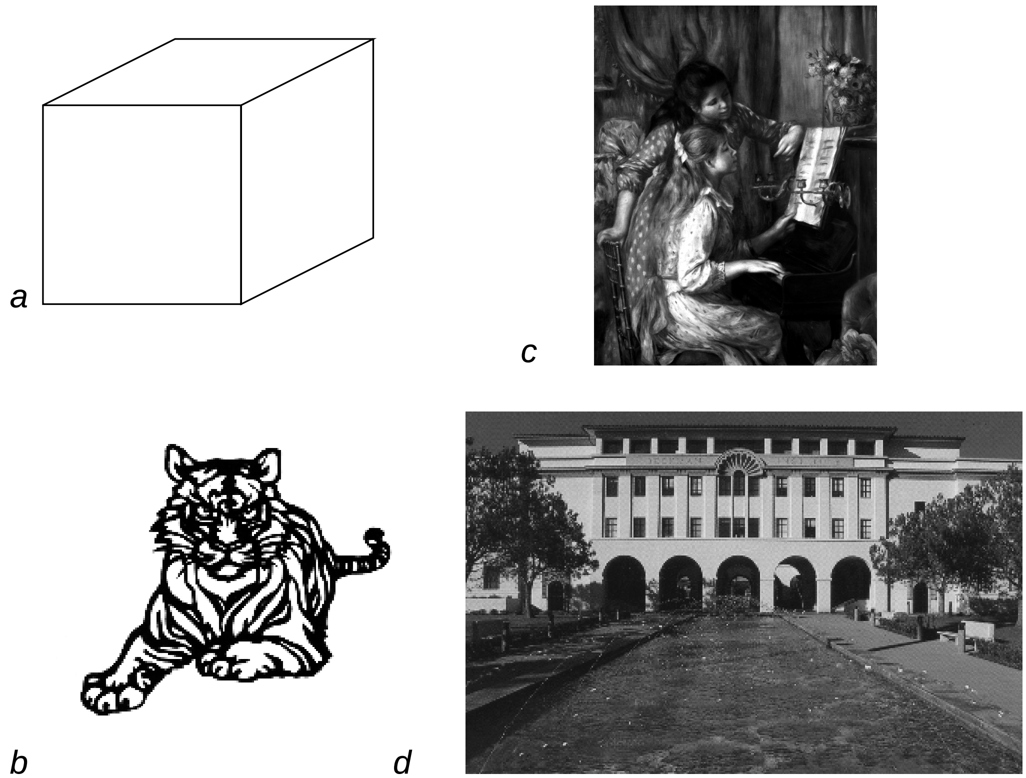


Figure 3: Examples of pictures that use symbols in a limited way, or do not use them: line drawings, a painting (“Piano lesson” by A. Renoir), a photograph (Beckman Institute at Caltech).

the eye. Therefore, *our internal representation of three-dimensional objects should be based on the information contained in these images.* We do not know what part of the information contained in retinal images is utilized in this internal representation.

As retinal images are two-dimensional central projections, there is one-to-one correspondence between a retinal image of an object and a projection of the object onto a plane (a linear perspective picture) and we can consider geometry of perspective projections instead of geometry of retinal images.

We distinguish at least two types of geometric information in retinal images: structural and nonstructural.

Structural features have more qualitative nature and are the easiest to detect. Examples of structural information are the number of holes in an image of an object, dimension of the image or its parts, the number of edges meeting at a vertex (Figure 4). Mathematically this information corresponds to the topological properties of the image. We assume that *structural information about projections of an object is present in the internal representation of the object. If an image of an object has structural* *features that do not match those of any perspective projection, it is perceived as contradictory, distorted or the object is not recognized at all.*

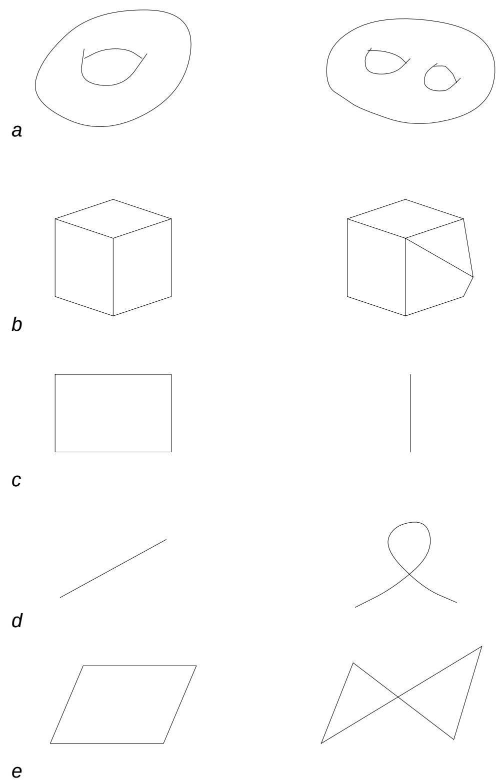


Figure 4: Examples of structural differences in images: a) the number of holes b) number of edges meeting at a vertex c) dimension d) number of self-intersections, local structure e) number of intersections of edges, local structure

Our assumptions about the importance of structural features are mostly motivated by common sense and intuitive ideas about perception due to the lack of experimental data. Also the concept of structural features in an idealization: the precision of the visual system is finite [FK81] and if a structural feature in an image is too small, it is not detected by the visual system.

Nonstructural features typically can vary without causing any significant change in perception. Examples of nonstructural features are the degree of convergence, foreshortening, gradients of texture, angles and curvature of edges (Figure 5). The important difference from structural features is that variation in nonstructural features can be registered by the visual system without producing considerable change in perception.

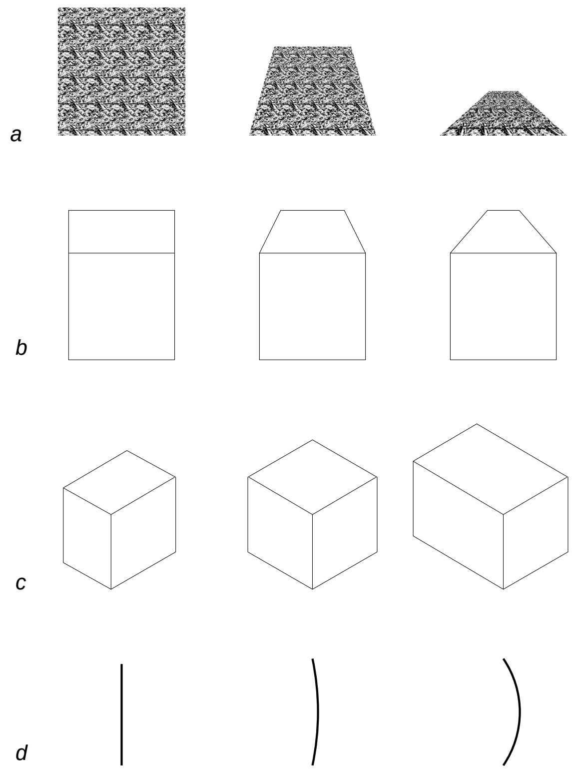


Figure 5: Examples of nonstructural differences in images: a) texture gradients b) parallel foreshortening c) perpendicular foreshortening d) curvature

2 Robustness of Pictures

In this section we start considering perceptual properties of images. One of the most important properties is *robustness*. The term *robustness of pictures* was introduced by [Kub86]. The perception of a picture typically does not depend on the viewing point: we can walk past a painting, tilt a book, move our head away, closer, or to the side of a computer display, and the objects in the picture are unlikely to change their apparent shape or position. This is one of the most important properties of pictures; were it not true, a viewing apparatus or correct choice of viewing position would be required for correct perception. Robustness is not total: some perceptual variables are less robust then others.

Robust quantities. A number of perceptual variables were confirmed to be independent of the viewing point. Several studies [RF80, RMDF80] has shown that for normal viewing conditions perception of slant does not depend on viewing position. This studies were extended by [Hal89] who showed that for extremely oblique viewing directions with the angle between the viewing direction and the picture surface less then 20°, robustness of the perception of slant breaks down.

There is some evidence [Hag76b, Smi58b, RF80, Experiment 1], that relative size judgements are robust as well.

Robustness is very important for moving pictures: for example, inverse perspective constructions, such as described in [Der89, LG59], applied to the images of rotating bodies might result in non-rigid objects if the projection point used for reconstruction is different from the actual projection point. However, this does not happen in most cases. As it was shown by [Cut87] for images of rotating near-rectangular solids, there is a general tendency to ignore small deformations, especially if they preserve angles between edges, and that rigidity is preserved at least up to a 45° viewing direction.

Relative orientations and layout are quite robust, ([Hal89, Experiment 5], [Gol79, Experiment 4], [Gol87, Experiment 1]) although relative distances between objects in the direction perpendicular to the picture plane depend somewhat on the viewing distance (see discussion below).

Breakdown of robustness. Not all perceptual variables are robust, and the range of viewing points within which robustness is preserved might depend on the depicted object.

We have already mentioned that relative distance between objects in the direction perpendicular to the picture plane is not very robust. Experiments described in [Smi58b, Smi58a] show that the viewing position affects judgements of relative distance although the effect is less than predicted by geometric reconstruction. It should be noted however, that reduced viewing conditions were used in these studies (monocular viewing through a peephole) and the absence of robustness can be attributed to the lack of information about the surface of the picture.

The orientation of the objects with respect to the observer is not robust at all in some cases. It is the most apparent nonrobust perceptual variable: it was noted by many authors that objects pointing perpendicular to the plane of the picture such as gun barrels or fingers (Figure 6) appear to be following the observer as he moves past the picture. Portraits often appear to be following the observer with their eyes. It turns out [Gol79, Gol87, Hal89] that this effect is more pronounced for orientations close to the perpendicular to the picture and decreases with the angle of orientation. This results in the following perceptual paradox: while spatial layout is perceived as more or less invariant, orientations of different objects change in different ways does not rotate much, while the direction of the road rotates considerably. Thus the difference between apparent orientations changes, while the relative orientation of the rut and the road does not change. This is an example of a more general phenomenon: the structure of perceived space need not be consistent; different perceptual mechanisms might produce contradictory information without causing any perceptual problems.

Robustness of perception may depend on the contents of the picture. For example, parallel projections of rotating rigid parallelepipeds are perceived as rigid in a wider range of viewing directions than central projections [Cut87, Experiments 1 and 2].

Robustness of pictures is related to the dual nature of pictures: we perceive both the picture surface

Figure 6: A World War I poster. (from [Tay63]). The finger appears to be following the viewer when he walks past the picture. Similar posters were created in the US ("Uncle Sam wants you!") and in Russia ("Did YOU volunteer?") during times when the government felt it necessary to point fingers at each citizen.

and the three-dimensional scene represented in this surface. Surface texture, flatness, and visible frame are important factors in perception of pictures. Numerous experiments provide evidence that the absence of these factors results in a decrease in the robustness.

Perception of slant becomes completely nonrobust when all the information about the picture surface is removed [RF80, RMDF80]. In unpublished experiments by W. Purdy [Lum80] only the picture frame was absent and the information about the picture surface was not completely removed. Still, for the simple pictures used in the experiments (slanted striped surfaces) picture frame removal was sufficient for almost total suppression of robustness.

3 Perception of Objects in Pictures

While perception of elementary quantities such as size or slant has some general trends described in the previous section, it cannot be separated from perception of specific shapes. Most distortions are associated with images of particular types of objects. Two groups of objects which frequently appear to be distorted in photographs [Kub86]. The first group included objects with rectangular three-dimensional corners (i.e. corners formed by three edges with all angles between them equal to 90°.) The second group included spherical and cylindrical objects and humans.

3.1 Rectangular Objects

Perception of pictures of rectangular corners is well described by simple rules [Per72, Per73, She81]. These basic facts about perception of rectangular corners are usually called *Perkins' laws*. We call three line segments meeting at a point a *three-star* [Per68]. We call an image of a rectangular corner *two-faced* if only two out of three boundary surfaces are visible, and *three-faced* if all three are visible (Figure 7).

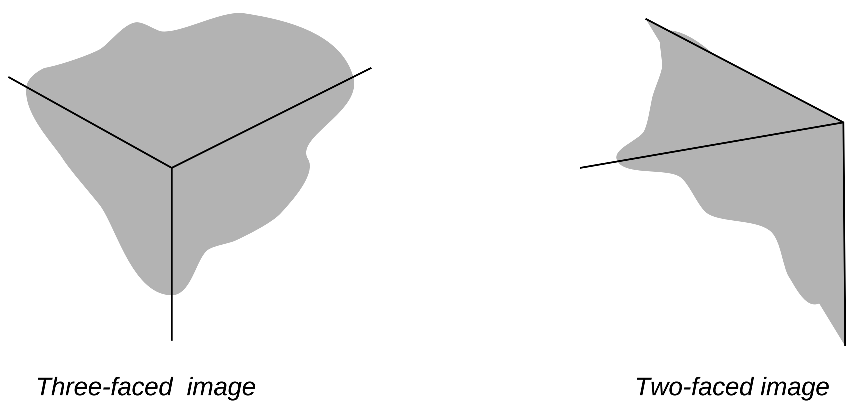


Figure 7: Two types of images of rectangular corners

> **Perkins' first law**. A three-star is acceptable as a two-faced image of a rectangular corner if and only if it contains two angles less than or equal to 90°, whose sum is greater then 90°.
>
> **Perkins' second law**. A three-star is acceptable as a three-faced image of a rectangular corner if and only if all three angles are greater than 90°.

Small deviations from Perkins' laws can occur, but in general there is a good agreement between the experiment and theory [Per72, She81].

It is important to note that Perkins' laws have a very simple geometric interpretation: acceptable projections coincide with orthogonal perspective projections. There is no difference between central and parallel projections in this case, because only angles between lines are important. For the central orthogonal projection we have to assume that the principal ray goes through the center of the three-star.

From Perkins' data we can make an estimate of the deviations from Perkins' laws that do not result in considerable distortion: the three-stars deviating from Perkins' laws by less then 5° still can be perceived as rectangular corners. Of course, this is a rough estimate. We discuss Perkins' laws in greater detail in Section 5.1.

3.2 Spheres, Cylinders, Humans and Animals

Perceptual requirements on the images of spheres are remarkably restrictive: only disks are generally accepted as good images of spheres [Pir70, Kub86]. No data on detection of "non-circularity" were available to us. It appears to be safe to assume that 1.1 aspect ratio is detectable (Figure 8) and accept it as an approximate upper boundary. Spheres are not that common in real environments, although quite popular in computer graphics images. They are also convenient test objects, because the distortion of shape in the image of a sphere is easy to detect and describe.

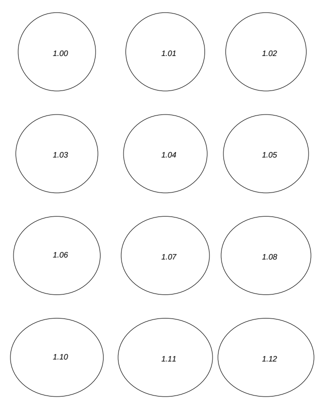


Figure 8: Ellipses with different aspect ratios (the aspect ratio is shown in the center of each ellipse.)

There are two distinct problems associated with images of cylinders, such as columns: the one that is most often mentioned arises when a row of cylinders is depicted. It is not a problem associated with the image of any particular cylinder, but rather a problem of unacceptable relative sizes of images. We discuss it in the Section 5.2.

The second problem can be considered a separate case of a more general class of problems that is associated with axially-symmetric objects.

The image of the foundation of the column or the upper part of a cup or a bowl is generally acceptable if it is an ellipse with major axis oriented perpendicular to the axis of symmetry (cf. Section 5.2, Figure 11).

Our perception of humans is likely to be very specialized. Humans come in many shapes, colors and sizes, so the result of perception of an inadequate image is more often a misperception (a false conclusion about the person in the picture) rather than direct perception of distortion as it is the case with spheres. If we stretch or shrink the image vertically or horizontally within a wide range, these changes are likely to produce acceptable although sometimes misleading pictures. For example, the men in Figure 9 are all identical but those close to the edges of the picture appear to be quite different from those in the middle. The part of the body which is most affected by deformation is the head, due to less variation in the shape of the head between individuals compared to other body parts.

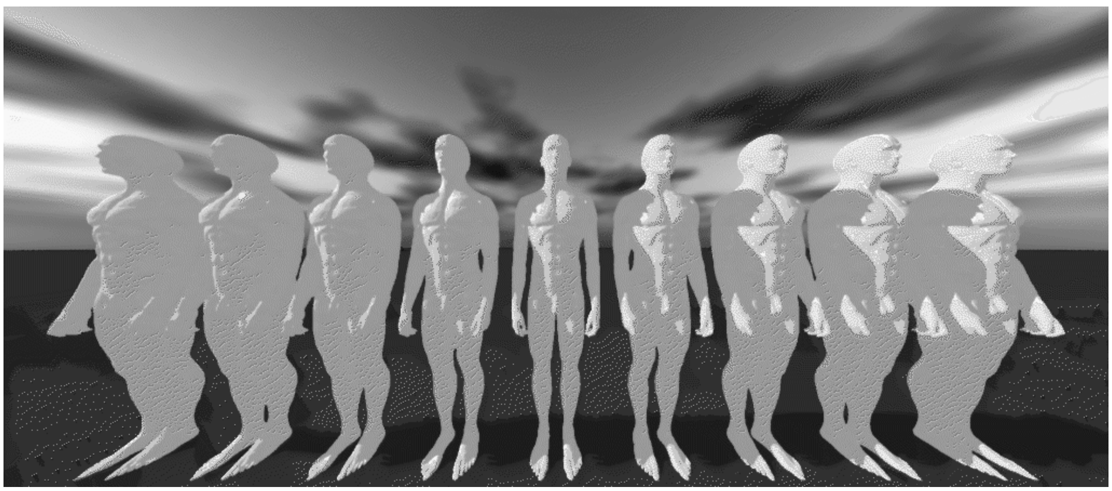


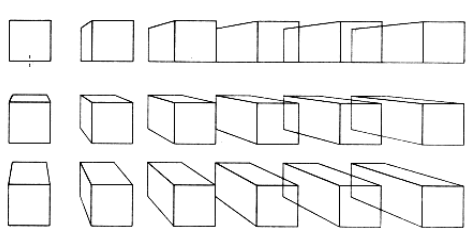
Figure 9: Deformations of human figures in a wide-angle perspective picture: 140° hhorizontally.

There are types of distortion that are more likely to produce deformed, not just misleading images. An important feature of acceptable frontal pictures of humans is their axial symmetry. When this symmetry is broken, the image becomes unacceptable. Similar conclusions can be reached about images of animals, although larger deformations are tolerated.

3.3 Foreshortening

Two types of foreshortening can be identified, parallel and perpendicular foreshortening.

For rectangular solids the preferred amount of perpendicular foreshortening depends on the aspect ratio [NK93b]. It was established that this amount is almost constant for cubes: the preferred amount was close to 1:0.6–1:0.7 for all viewing conditions. It should be noted that the use of line drawings in the study [NK93b] could considerably affect the results: perception of the whole the absence of any distinctive features in the drawing, specifying absolute size(e.g. label on a match box or windows on a building) could affect the preferred amount of foreshortening in an unpredictable way.



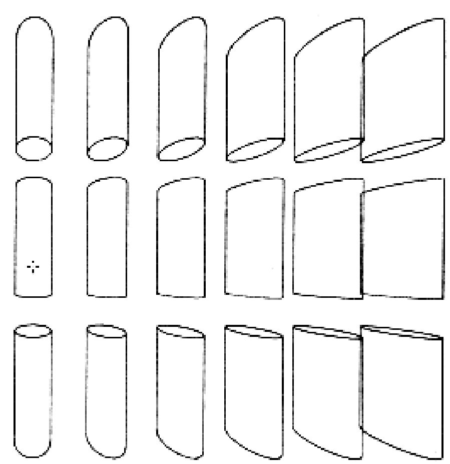


Figure 10: (Left) Wide-angle projection of cubes (based on [Pir70])

Figure 11: (Right) wide-angle projection of cylinders from (based on [Pir70])

The amount of perpendicular foreshortening deviating significantly from the preferred one results in distortion if it is known that the depicted objects are supposed to be cubes, and in misperception if no such information is available. (Figure 10).

Perpendicular foreshortening is easy to define only for objects with distinct edges parallel and perpendicular to the picture plane; it can be also defined for familiar objects with distinctive features parallel or perpendicular to the picture plane. Some distortions of the human figures can be described in terms of the perpendicular foreshortening ratio.

Parallel foreshortening across objects can vary in a wide range without causing perceptual problems. In perspective images the ratio of parallel foreshortening is directly related to the viewing angle: the greater the viewing angle, the greater the ratio for an object of a given size. [NK93a] studied the effects of the change in the viewing angle on the perception of cubes; it was found that a moderate degree of parallel foreshortening (approximately 1:0.75) was consistently preferred independent of the viewing conditions.

4 Straightness and Verticality

Intuitively it is clear that the images of straight edges and lines should be straight. It is important to know the accuracy of the perception of straightness. Experimental studies that we know about considered the perception of straightness only for very short lines with angular size close to 1°. It was found that for small perturbations perceptual threshold is determined by the solid angle subtended by the maximal bump with respect to the least-squares straight line (Figure 12) [WWC87]. The threshold value was determined to be close to 0.3 sq arc min.

In [OD67] a similar measure was used although a chord rather than a least-squares line was considered and there was only one bump. Resulting thresholds had the same order of magnitude (0.4–2.2 sq. arc. min.) These values are remarkably small: [WWC87] points out that the receptor density is not sufficient for the detection of perturbations of that size and some type of higher-level processing should be involved to account for these thresholds (hyperacuity). It appears that this threshold is likely to increase with the angular size of the line. However, it is bound to stay quite low and any considerable deviation from straightness causes perceptual distortion.

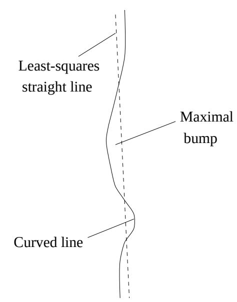


Figure 12: Watt's measure of perceptual curvature

The perception of verticality is influenced by two main factors: vestibular perception of the force of gravity and visual perception. Objects that we assume to be vertical (walls, trees, etc.) or horizontal (the surface of the ground) determine the visual vertical. A number of experiments ([OK71, DR82]) show that when there is a conflict between vestibular and visual information, there is no clear preference for the objective vertical determined by the vestibular system. The "rod-and-frame" effect ([DR82]) indicates that a visible frame affects perception of actual vertical: when a vertical rod is viewed inside a tilted frame it appears to be tilted in the direction opposite to the direction of the frame.

In pictures, lines that are not parallel to the edges of a rectangular frame are perceived as non-vertical. If these lines represent vertical or horizontal edges of objects (buildings or furniture), the picture creates a feeling of instability which is sometimes undesirable. The "rod-and-frame" effect explains to some extent the origin of this feeling.

5 Linear Perspective

Linear perspective was introduced as a systematic tool during the Renaissance. To some extent linear perspective was known to the Romans (frescoes of Pompeii) and Chinese (see discussion of other perspective systems in Section 6) but during the Renaissance it was introduced as a rigorous system which was considered to be the foundation of painting and drawing till the second half of the nineteenth century. The first artist to use perspective during the Renaissance was Filippo Bruneleschi, but the first written description and analysis was done by Leon Alberti in [Alb76] and Leonardo da Vinci in [dV70]. Leonardo da Vinci was also the first to observe the limitations of the method. As it is discussed in [Kub86], Leonardo was not convinced that perspective pictures are robust, hence his overly stringent requirements to the application of perspective ("Leonardo's rule"):

> ...do not trouble yourself about representing anything, unless you take your view-point at a distance of at least twenty times the maximum width and height of the thing that you represent; and this will satisfy every beholder who places himself in front of the work at any angle whatsoever.

This means that angular size of any object in a picture should be less than 3° clearly an overkill. Leonardo offers the following example of perspective distortion (Figure 13): in the image of a row of columns depicted from a close projection point, the width of the images of columns closer to the edges of a picture would increase, which is quite counterintuitive.

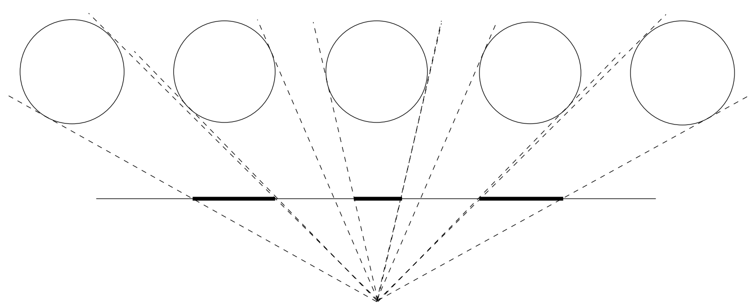


Figure 13: Leonardo's example of a perspective distortion

Statements similar to Leonardo's rule are quite common. For example, [Gla94] does not recommend using viewing angles above 40–50°. [Olm49] recommends viewing angles no more than 37° horizontally and 28° vertically.

We see that the limitations of linear perspective were recognized simultaneously with its discovery. As it is pointed out by [Kub86] it is not correct to identify artistic linear perspective with central projection of a halfspace onto a plane: only a limited range of viewing angles is used, and, as the images of many objects are often painted with deviations from linear perspective.

Visual field. Field of view is one of the most important parameters of linear perspective images. It is important to relate it to the field of view of the human eye. The total size of the human field of view is quite large; it extends more than 160° horizontally and 150° vertically for each eye [CF75]. However, the density of the receptors is very nonuniform, and maximal resolution is achieved only in the fovea of the eye which has an angular size of only about 2°. Lateral vision is very limited in resolution; its main function is presumably motion detection. As the resolution decreases, less and less detail and precision is available to the rest of the visual processing system. [FK81] found that the size of the field where gratings 1° apart can be resolved is approximately 30°. The difference in resolution is most apparent if we consider the following example. The angular size of the moon is 0.5°. Outside 30° field of view we cannot detect any details of the lunar surface and cannot even tell if the moon is round or not.

In photographs and computer generated images linear perspective is practically identical to central projection (exact linear perspective): wide-angle images are not uncommon, and, due to the physics of the process or nature of the algorithms, resulting images follow the rules of the central projection exactly. Intuitive compensation used by the artists is not available in this case.

Two methods are helpful in the analysis of exact linear perspective: first, we can check how well linear perspective images satisfy various perceptually desirable requirements described in the previous sections; second, we can examine deliberate deviations from linear perspective that are common in painting.

Structural features. Linear perspective images have no undesirable structural features (see Section 1.3).

Robustness. The studies of robustness were done using perspective images. Summarizing their results, we can say that under normal viewing conditions perspective images are robust when the viewing angle is sufficiently small or the viewing direction is not too oblique. In general robustness is a concern only for moving images.

5.1 Rectangular Objects in Linear Perspective

In perspective images Perkins' laws are always violated to some extent, with the only exception of orthogonal parallel projection. The extent of this violation depends for the parallel projection on the direction of projection and for the central projection on the direction from the center to the apex of the rectangular corner. We call both directions directions of projection.

We have selected two parameters to characterize the violation of Perkins' laws: fraction of rectangular corners with projections violating Perkins laws for a given direction of projection and maximal deviation of the *offending angle* from 90°.

There are infinitely many possible positions of rectangular corners in space, and each of them can be represented by three angles: two angles θ and ψ determining the orientation of one of the edges, and the angle of rotation ϕ of the remaining pair of edges around the first one.

Define a triple (θ, ψ, ϕ) to be "good" if the projection of the corner in the given direction satisfies Perkins laws.

Then we can define the fraction of "good" projections to be

$$\frac{\int_{\text{all "good" } (\theta,\psi,\phi)} d\theta d\psi d\phi}{\int_{\text{all } (\theta,\psi,\phi)} d\theta d\psi d\phi} = \frac{1}{4\pi^3} \int_{\text{all "good" } (\theta,\psi,\phi)} d\theta d\psi d\phi$$

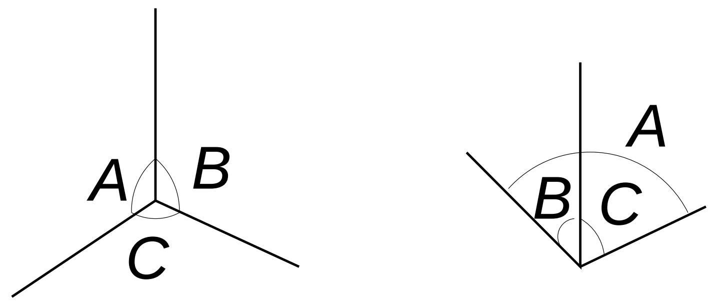


Figure 14: Three-sided and two-sided images of a rectangular corner

We define the offending angle in the following way. Let A, B, C be the angles between the edges of the three-star (always less then 180°). Let $A \geq B \geq C$. Then there are two possible violations of Perkins' laws: either all three angles are less than 90°, or only one of them is less than 90° (C). In the first case $A = B + C$, and A should be greater than 90°, while B and C can be less than 90°. In this case we define A (the largest angle) to be offending angle. In the second case, there are two possibilities: $A = B + C$ or $A = 360^\circ - B - C$. In the first case B should be less than 90° (second Perkins law applies) and we define it to be the offending angle. In the second case all angles should be greater than 90° and we define C to be the offending angle.

It can be easily shown that our definition is equivalent to the following simpler one:

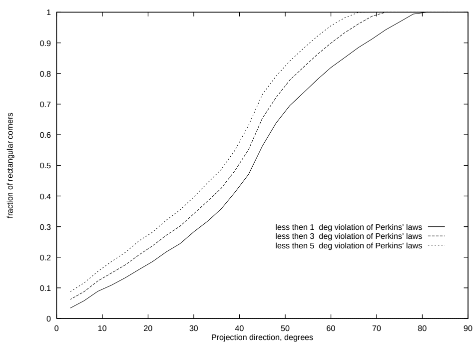
the **offending angle** is the angle between edges of the three-star which is the closest to 90°.

Figure 15 shows the fraction of "good" corners as a function of the angle between the projection direction and the picture plane (Monte-Carlo evaluation). It also shows the fractions of "almost good" corners - those with deviations less than 1, 3, 5 degrees.

Figure 16 shows the maximal offending angle as a function of the angle between the projection direction and the picture plane.

We can see that all factors are negligibly small for small angles between the projection direction and the projection plane. From [Per72] we can see that deviations of approximately 5 degrees are still tolerated in almost 50% of the cases. Therefore, in most cases pictures with projection angles that are small enough not to create distortions of more than 5° can be considered acceptable. These plots can be used to quantify robustness of pictures at least as far as depiction of rectangular objects is concerned.

Foreshortening is another important aspect of appearance of rectangular objects. Haagen and Nicholls [HJ78, NK93a] examined preferred parallel foreshortening for parallelepipeds and other types of prisms. Again, the general trend of their results is in the direction of lesser projection angles, although [NK93a] argues that a moderate degree of parallel foreshortening is preferred to parallel projection, while [HJ78] claims that parallel projection has the highest rating. Detailed examination of the data in two papers suggests that the former paper is more reliable.



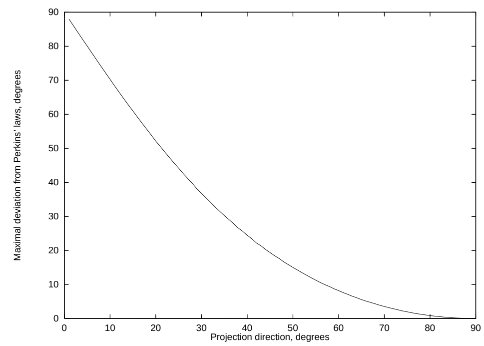


Figure 15: (Left) Fraction of "good" rectangular corners *vs* projection angle

Figure 16: (Right) Maximal offending angle *vs* projection angle

The perpendicular foreshortening in perspective images of rectangles might also create problems (Figure 10). Comparing to the results of [NK93b], we can see that the ratio of foreshortening can be quite far from the perceptually acceptable range. It should be noted that cubes are not that common and much wider range of foreshortening ratios can be tolerated for arbitrary parallelepipeds, unless this is something familiar like a computer terminal or a TV.

5.2 Spheres, Cylinders, Humans

All nonorthogonal projections of spheres are ellipses. The aspect ratio of these ellipses is a useful measure of deviation of the image from perceptually acceptable. Figure 17 shows the aspect ratio of the image of a small sphere as a function of the projection direction. Assuming 1.1 upper bound for the acceptable aspect ratio, we can see that the projection direction should not be less than 75-77° (projection angle 45-50°). Images of spheres are not that common in painting. When they are present, they are depicted as circles regardless of their position. The most famous example of this type is "The School of Athens" by Rafael (Figure 18) [Pir70][LG59].

In the beginning of this chapter we pointed out the problem with rows of cylinders that is specific to the linear perspective. Another, more general problem is the tilt of the axes of the image of a cross-section (Figure 11): major axis of the horizontal cross-section is always oriented towards the center of the image. The most apparent distortions occur in the images close to the lines through the center of the picture tilted at 45°.

Images of humans in exact linear perspective may look grossly distorted because of extreme changes of the aspect ratios and violations of symmetry (Figure 19a). No such asymmetry is observed in the art: human bodies are typically drawn as if the projection point was located directly in front of them.

Another problem, which is well-known to photographers, is the distortion of the features of a face resulting from high degrees of perpendicular foreshortening.

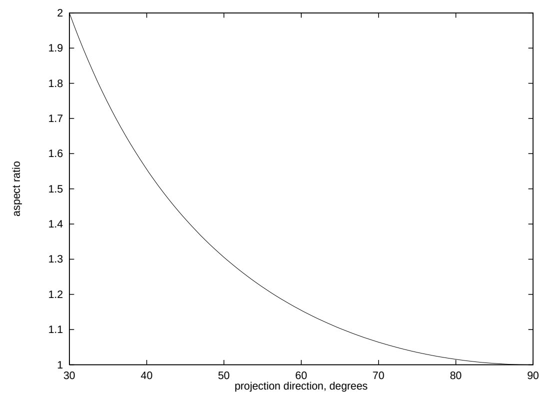


Figure 17: Aspect ratio of the image of a small sphere *vs* projection direction angle.

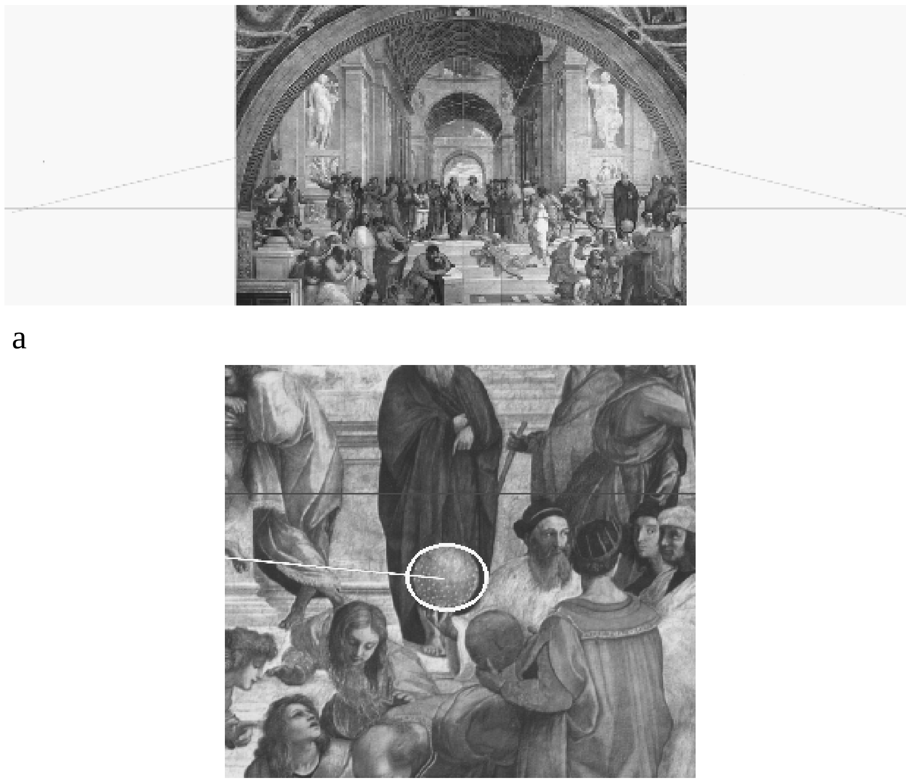
5.3 Straightness, Verticality, Texture Density, Movement

Straightness. Linear perspective images of straight lines are straight. Linear perspective is the only projection that has this property [Kle39]; in this sense it is optimal. However, this comes at the cost of shape distortion as discussed in the previous section.

Verticality. A number of problems are associated with the perception of verticality in linear perspective images. Unless the picture plane is perpendicular to the ground, vertical lines converge or diverge. In some cases desirable artistic effect, in other cases it is undesirable. Even small tilts of the picture plane create considerable distortion in wide-angle images partly due to the violation of the Perkins laws and partly to the divergence of the vertical line and the conflict with the frame of the picture. Feininger, a photographer famous for his images of New York mentions that the extreme effects of the tilt of the camera plane may be used to create "rather nonorthodox picture far more interesting than the conventional view" [Fei53]. However, in many cases it is highly desirable to have vertical lines parallel to the frame. In particular, special cameras with film plane that can be tilted with respect to the axis of the lens are often used in architectural photography.

Texture density. Another problem is related to the change in the texture density: texture elements of identical size have more extended projections when they are close to the edges of the image. This effect can be observed in Figure 20: the tree appears to be stretched to the left partly because of the texture density gradient.

Motion. For moving linear perspective images (computer animations, cinema) some of the problems associated with oblique and wide-angle projections are amplified.



a

b

Figure 18: ’The School of Athens” by Rafael. a. General view; b. Detail; compare the image of the sphere as painted by Rafael with reconstruction of the central projection of the sphere (position of the center of projection were determined from the images of the columns and walls).

For example, as it is described by Cutting [Cut87], rotating rectangular solids are more often perceived as non-rigid for oblique projections and wide-angle perspective. Cutting attributes this fact to the greater percentage of frames that do not satisfy Perkins’ laws for a given viewing point. Because of the changes in perpendicular foreshortening, many objects appear to stretch or shrink when they move across the field of view.

The height of remote objects, such as mountains or clouds, above the horizon changes when they move from the periphery of the picture to the center. These changes are quite apparent in wide-angle pictures and in many cases are perceptually undesirable (Figure 21,) although correct geometrically.

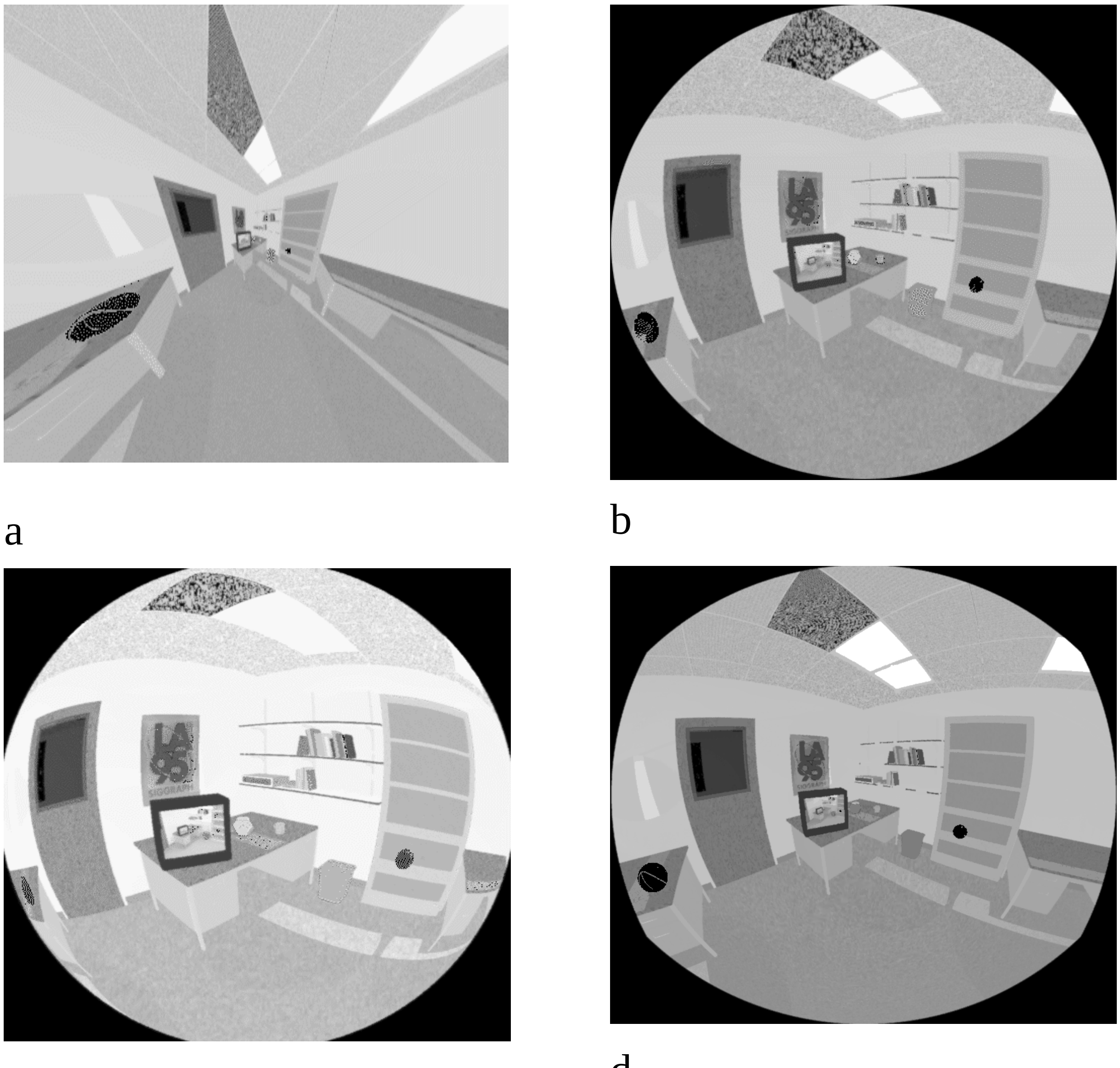


Figure 19: Extremely wide angle images (160°): a. Linear perspective. b. Angular fisheye (equidistant projection). c. Hemispherical fisheye (orthographic projection). d. Our direct projection. Images (b) and (d) are quite similar, but in (c) we can observe excessive compression of the objects in the margins: the tables appear to be narrower and the ball does not look spherical.

Figure 20: Beckman Institute courtyard at Caltech, photo by the author

6 Other Traditional Perspective Systems

From the Renaissance until the second half of nineteenth century Western painting and drawing was based on linear perspective. As we have mentioned before, some deviations from its laws were common, and analysis of perception provides some explanations for them. Artistic perspective is mostly a tool for depiction of relative positions and sizes of objects in space; its role in depiction of separate objects is much less important.

In the Middle Ages European art was quite advanced, but did not make use of any projection system in a consistent manner. The same can be said about Byzantine and ancient Russian art. Compositions created by the artists were often quite complicated, and some facts about their organization can be useful to understand. Other cultures produced evolved forms of pictorial art independently.

Perhaps Chinese and Japanese art is the most advanced system different from Western art. Japanese and Chinese approach to the depiction of objects might differ in some details but similarities are considerable: the use of color and shading is quite different but geometrically there is less difference. Most im-

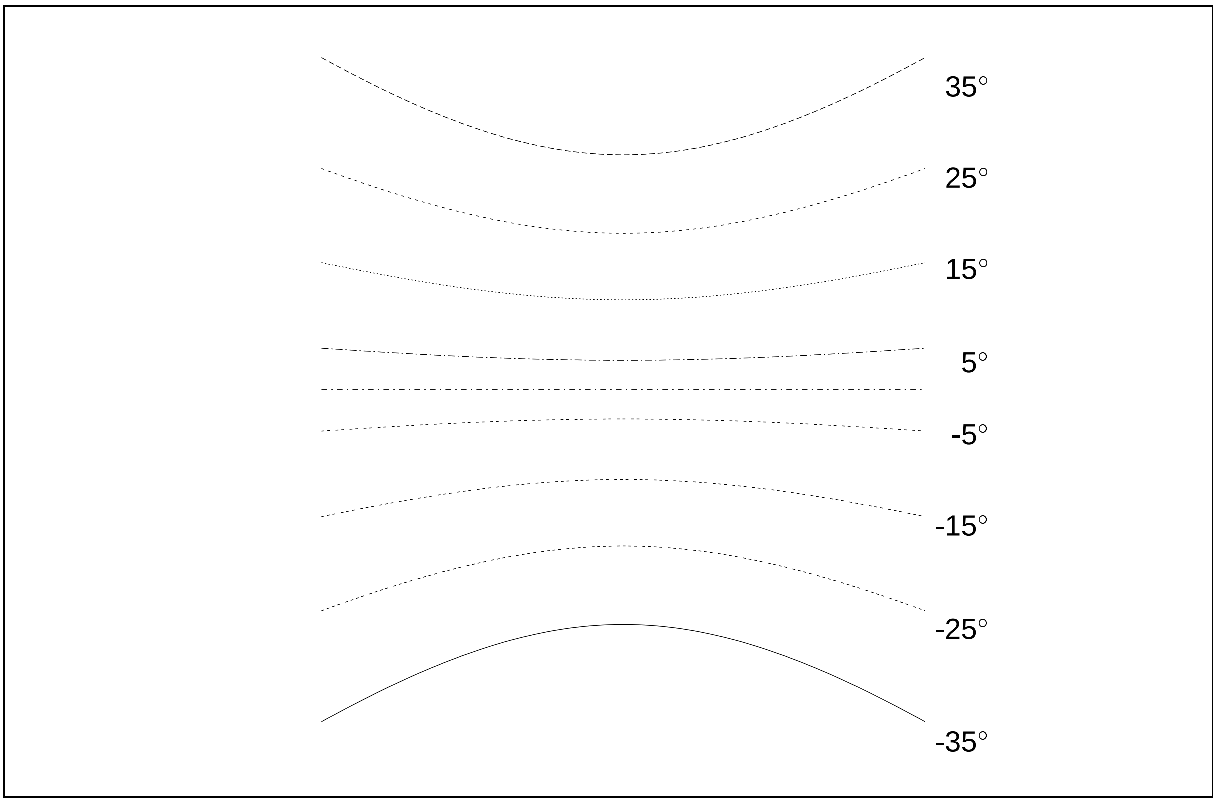


Figure 21: Trajectories of points in a wide-angle motion picture when the camera turns. The projection angle is 90° in vertical and horizontal directions.

portantly, the approach to the depiction of space adopted in Oriental art is quite consistent and somewhat different from the concept of linear perspective. As it was pointed out by many authors (for example, [Hag86]) Japanese and Chinese artists typically used oblique parallel projection in their drawings. As in Western art, projection was used mostly for depiction of relative positions of objects rather than for separate objects.

Parallel projection is just a special case of perspective projection with the projection point removed to infinity. The main problem of this projection is that there is no way to combine images of very large objects (mountains) and small objects (people) in the same picture. This problem was often solved in the following way: parallel projection was used for each part of the picture, but some amount of perspective diminution with distance was introduced for objects located further away. As a smooth transition between these separate areas was impossible, they were separated by symbolic clouds.

In a sense, inside the continuous areas Oriental practice is more consistent than Western. The shape of the linear perspective image of an object depends upon its position within the image. For parallel projection this is no longer true; therefore, a rigorous construction similar to an aerial photograph is possible. For oblique projections, however, it means not the absence of the distortion of shape but a constant distortion. If the angle between the direction of projection and the projection plane is greater than 70-75°, all types of distortion are quite small and can be ignored. Of course, this is mostly theoretical observation and no precise geometric constructions were ever utilized.

It appears that oriental artists did not use linear perspective not because they were completely unfamiliar with it. A painting by Zhang Zheduan dating as early as XII century (Figure 22), clearly exhibits perspective diminution. [Sul89] quotes a Sung dynasty critic Shen Kua who criticized a tenth-century landscapist Li Ch'eng for his use of linear perspective. "Why look at a building, said Shen Kua, from only one point of view? Li Ch'eng's 'angles and corners of buildings' and his 'eaves seen from below' are all very well, but only continually shifting perspective enables us to grasp the whole."

Another interesting aspect of Oriental art is occasional use of divergent perspective for rectangular objects. The same trend can be even more consistently observed in the Byzantine and ancient Russian art (Figure 23). It cannot be attributed to the lack of skill; other aspects of images demonstrate very high skill and artistic talent. In Byzantine art in particular it became to large extent a matter of convention. But no religious or symbolic basis is firmly established for this particular convention, and instances of such images in Oriental and Persian art demonstrate that it might have some perceptual basis.

Deregowski et al. [DP92, DPM94] argue that laterally displaced objects (i.e. those located in the periphery of the visual field) are perceived in this way. A number of other explanations were attempted [Rau86]. This phenomenon is mostly of historical interest, because it is likely that most contemporary people would not find these images to be good representations of rectangular objects. It demonstrates that potentially the conventional element in representation is very significant: it could be that some of these images were perceived as quite acceptable by contemporaries.

The organization of space in Byzantine and ancient Russian paintings is of greater interest: in some sense, the general principle of the representation of space is similar to the approach some of the Chinese and Japanese artists were taking. The scene is subdivided into several parts without connecting elements.

Figure 22: Chang Tse-tuan (c. 1100–1130) *Going Up River at Ch'ing-ming Festival Time*, Detail of a handscroll. Palace Museum, Beijing.

Each part of the scene is depicted separately. The difference from the Oriental art is in considerable occlusion and much less consistent size relationships.

Summarizing our observations on different artistic cultures that have attempted to depict space rather than separate objects, we can observe that they have at least one point in common: all straight lines are depicted as straight lines. This immediately restricts the class of possible projections to piecewise central and parallel linear projections.

Divergent perspective can be considered as central projection applied "backwards". A common trend can be observed in the depiction of space: the whole composition is divided into several parts, and each of this parts is drawn independently. The projection used for each part is close to parallel. There are quite a few exceptions to the above, but if any system is ever used consistently, it follows this pattern.

It should be noted that other cultures developed even more different systems of pictorial representation. For example, [Hag86] describes the system that is used in the art of Northwest Coast Indians. These images appear even less realistic than Byzantine art to a modern Western observer and are not particularly relevant to our discussion.

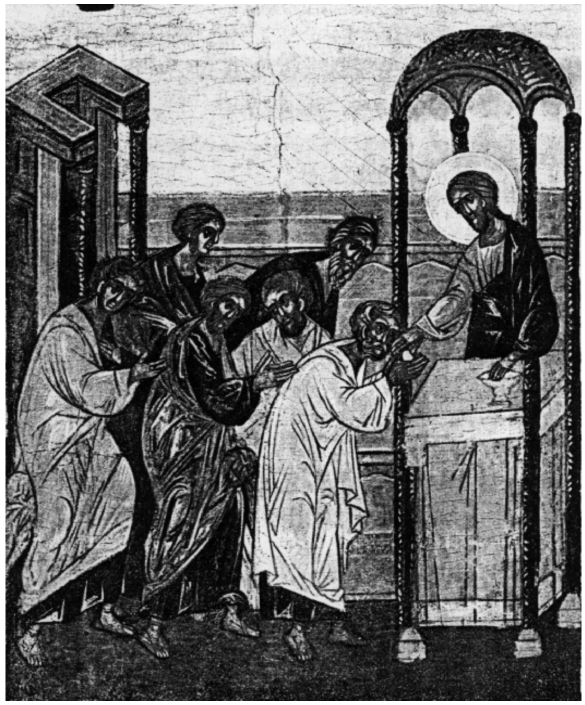


Figure 23: *Eucharist* (XV century), State Russian Museum (from [Rau80])

7 Summary

From practical point of view, the following facts about perception discussed in this survey appear to be of greatest relevance:

- Images of objects should be topologically similar to a linear perspective image of the object.
- To be perceptually acceptable, images of objects should satisfy certain criteria, such as Perkins' laws for rectangular corners, have aspect ratio close to 1 for the images of spheres and limited perpendicular foreshortening ratio for parallelepipeds, cylinders and humans.
- Linear perspective works best for smaller projection angles. Estimates of the critical angle vary, but for projection angles less than 35° we are guaranteed that there will be practically no distortion. Within these bounds most of the perceptual criteria mentioned above are satisfied. This fact is confirmed by the artistic practice in different cultures.
- For wider projection angles distortions may occur. They become quite objectionable when the projection angle exceeds 80°. Some familiar geometric shapes (rectangular corners, spheres,

cylinders, human figures) are especially sensitive to such distortions. Distortions are amplified in moving pictures.

References

[Alb76] Leon Battista Alberti. *On painting*. Greenwood Press, Westport, Connecticut, 1976.

[Arn54] Rudolf Arnheim. *Art and visual perception; a psychology of the creative eye*. University of California Press, Berkeley, 1954.

[CF75] Edward C. Carterette and Morton P. Friedman, editors. *Handbook of Perception: Seeing*, volume 5. Academic Press, New York, 1975.

[Cut87] James E. Cutting. Rigidity in cinema seen from the front row, side aisle. *Journal of Experimental Psychology: Human Perception and Performance*, 13(3):323–334, 1987.

[Der89] Jan B. Deregowski. Geometric restitution of perspective: Bartel's method. *Perception*, 18(5):595–600, 1989.

[Der91] Jan B. Deregowski. Intercultural search for the origins of perspective. In Nico Bleichrodt and Pieter J. D. Drenth, editors, *Contemporary issues in cross-cultural psychology*, pages 334–346. Swets & Zeitlinger, Amsterdam, Netherlands, 1991.

[DP92] Jan B. Deregowski and Denis M. Parker. Convergent and divergent perspective. *Perception*, 21(4):441–447, 1992.

[DPM94] Jan B. Deregowski, Denis M. Parker, and Manfredo Massironi. The perception of spatial structure with oblique viewing: An explanation for byzantine perspective? *Perception*, 23(1):5–13, 1 1994.

[DR82] Joseph R. DiLorenzo and Irvin Rock. The rod-and-frame effect as a function of the righting of the frame. *Journal of Experimental Psychology: Human Perception & Performance*, 8(4):536–546, Aug 1982.

[dV70] Leonardo da Vinci. *Notebooks I, II*. Dover, New York, 1970.

[Fei53] Andreas Feininger. *Feininger on photography*. Crown Publishers, Inc., New York, revised edition, 1953.

[FK81] Ronald A. Finke and Howard S. Kurtzman. Mapping the visual field in mental imagery. *Journal of Experimental Psychology: General*, 110(4):501–517, Dec 1981.

[Gla94] G. Glaeser. *Fast Algorithms for 3-D Graphics*. Springer-Verlag, New York, 1994.

[Gol79] E. Bruce Goldstein. Rotation of objects in pictures viewed at an angle: Evidence for different properties of two types of pictorial space. *Journal of Experimental Psychology: Human Perception & Performance*, 5(1):78–87, Feb 1979.

[Gol87] E. Bruce Goldstein. Spatial layout, orientation relative to the observer, and perceived projection in pictures viewed at an angle. *Journal of Experimental Psychology: Human Perception & Performance*, 13(2):256–266, May 1987.

[Goo76] Nelson Goodman. *Languages of art*. Hackett, Indianapolis, IN, 2nd edition edition, 1976. constructivist approach to picture perception.

[Hag76a] Margaret A. Hagen. Development of ability to perceive and produce pictorial depth cue of overlapping. *Perceptual and Motor Skills*, 42(3, part 1):1007–1014, June 1976.

[Hag76b] Margaret A. Hagen. Influence of picture surface and station point on the ability to compensate for oblique view in pictorial perception. *Developmental Psychology*, 12(1):57–63, January 1976.

[Hag80] Margaret A. Hagen, editor. Academic Press series in cognition and perception. Academic Press, New York, 1980.

[Hag86] Margaret A. Hagen. *Varieties of realism : geometries of representational art*. Cambridge University Press, Cambridge ; New York, 1986.

[Hal89] Thomas O. Halloran. Picture perception is array-specific: Viewing angle versus apparent orientation. *Perception and Psychophysics*, 45(5):467–482, May 1989.

[HJ78] Margaret A. Hagen and Rebecca K. Jones. Differential patterns of preference for modified linear perspective in children and adults. *Journal of Experimental Child Psychology*, 26(2):205–215, October 1978.

[Kle39] Felix Klein. *Elementary mathematics from an advanced standpoint:geometry*. Macmillan company, New York, 1939.

[Kub86] Michael. Kubovy. *The psychology of perspective and Renaissance art*. Cambridge University Press, Cambridge ¡Cambridgeshire¿ ; New York, 1986.

[LG59] J. de La Gournerie. *Traité de perspective linéare contenant les tracés pour les tableaux plans et courbes, les bas-reliefs et les décorations théatrales, avec une théorie des effets de perspective*. Dalmont et Dunod, Mallet-Bachelier, Paris, 1859. 1 volume and 1 atlas of plates.

[Lum80] Ernest A. Lumsden. Problems of magnification and minification: An explanation of the distortions of distance, slant, shape, and velocity. In Hagen [Hag80], pages 91–135.

[Mas93] Penelope Mason. *History of Japanese Art*. Harry N. Abrams, Inc., New York, 1993.

[Mil86] Miriam Milman. *Trompe-l'oeil, painted architecture*. Skira/Rizzoli, New York, 1986.

[NK93a] Andrea L. Nicholls and John M. Kennedy. Angular subtense effects on perception of polar and parallel projections of cubes. *Perception & Psychophysics*, 54(6):763–772, Dec 1993.

[NK93b] Andrea L. Nicholls and John M. Kennedy. Foreshortening and the perception of parallel projections. *Perception & Psychophysics*, 54(5):665–674, Nov 1993.

[OD67] John Ogilvie and Eva Daicar. The perception of curvature. *Canadian Journal of Psychology*, 21(6):521–525, 6 1967.

[OK71] Jin Ong and Dorothy J. Kessinger. Perception of verticality with a rod and frame apparatus. *American Journal of Optometry & Archives of American Academy of Optometry*, 48(8):662–666, Aug 1971.

[Olm49] P. Olmer. *Perspective artistique*. Plon, Paris, 1943,1949.

[Per68] David N. Perkins. Cubic corners. Quarterly Progress Report 89, MIT, Research Laboratory of Electronics, 1968.

[Per72] David N. Perkins. Visual discrimination between rectangular and nonrectangular parallelopipeds. *Perception & Psychophysics*, 12(5):396–400, 1972.

[Per73] David N. Perkins. Compensating for distortion in viewing pictures obliquely. *Perception and Psychophysics*, 14(1):13–18, 1973.

[Pir70] M. H. Pirenne. *Optics, painting and photography*. Cambridge University Press, New York, 1970.

[Rau80] B. V. Raushenbakh. *Prostranstvennye postroenya v zhivopisi*. Nauka, Moskva, 1980. in Russian.

[Rau86] B. V. Raushenbakh. *Sistemy perspektivy v izobrazitel'nom iskusstve : obshchaia teoriia perspektivy*. Nauka, Moskva, 1986. in Russian.

[RF80] Richard R. Rosinski and James Faber. Compensation for viewing point in the perception of pictured space. In Hagen [Hag80], pages 137–176.

[RMDF80] Richard R. Rosinski, Timothy Mulholland, Douglas Degelman, and James Farber. Picture perception: An analysis of visual compensation. *Perception & Psychophysics*, 28(6):521–526, Dec 1980.

[She81] Roger N. Shepard. Psychophysical complimentarity. In Michael Kubovy and James R. Pomerantz, editors, *Perceptual Organization*, pages 279–343. Lawrence Erlbaum Associates, Publishers, Hillsdale, New Jersey, 1981.

[Smi58a] Olin W. Smith. Comparison of apparent depth in a photograph viewed from two distances. *Perceptual and Motor Skills*, 8:79–81, 1958.

[Smi58b] Olin W. Smith. Judgements of size and distance in photographs. *The American Journal of Psychology*, 71:529–538, 1958.

[Smi61] Smith Olin W. Smith, Patricia C. Ball throwing responses to photographically portrayed targets. *Journal of Experimental Psychology*, 62(3):223–233, 1961.

[Sul89] Michael Sullivan. *The Meeting of Eastern and Western Art*. University of California Press, Berkeley and Los Angeles, 1989.

[Tay63] A. J. P. Taylor. *The First World War: an illustrated history*. Hamish Hamilton, London, 1963.

[WWC87] R. J. Watt, R. M. Ward, and C. Casco. The detection of deviation from straightness in lines. *Vision Research*, 27(9):1659–1678, 1987.

[Zor95] Denis Zorin. Correction of geometric perceptual distortions in pictures. Master's thesis, California Institute of Technology, 1995.

Correction of Geometric Perceptual Distortions in Pictures.

Denis Zorin, Alan H. Barr
California Institute of Technology, Pasadena, CA 91125

Abstract

We suggest an approach for correcting several types of perceived geometric distortions in computer-generated and photographic images. The approach is based on a mathematical formalization of desirable properties of pictures.

From a small set of simple assumptions we obtain perceptually preferable viewing transformations and show that these transformations can be decomposed into a perspective or parallel projection followed by a planar transformation. The decomposition is easily implemented and provides a convenient framework for further analysis of the image mapping.

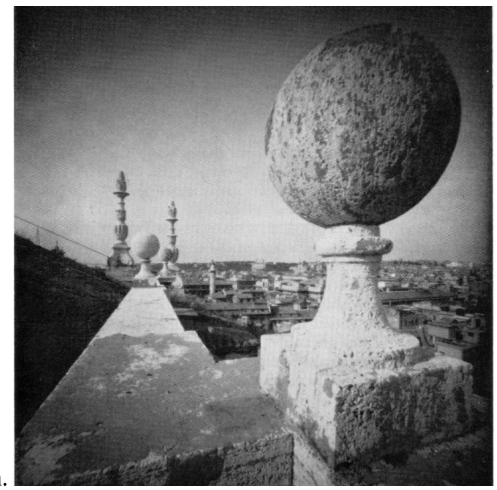
We prove that two perceptually important properties are incompatible and cannot be satisfied simultaneously. It is impossible to construct a viewing transformation such that the images of all lines are straight and the images of all spheres are exact circles. Perceptually preferable tradeoffs between these two types of distortions can depend on the content of the picture. We construct parametric families of transformations with parameters representing the relative importance of the perceptual characteristics. By adjusting the settings of the parameters we can minimize the overall distortion of the picture.

It turns out that a simple family of transformations produces results that are sufficiently close to optimal. We implement the proposed transformations and apply them to computer-generated and photographic perspective projection images. Our transformations can considerably reduce distortion in wide-angle motion pictures and computer-generated animations.

Keywords: Perception, distortion, viewing transformations, perspective.

1 Introduction

The process of realistic image synthesis can be subdivided into two stages: modeling the physics of light propagation in three-dimensional environments and projecting the geometry of three-dimensional space into the picture plane (the "viewing transformation.") While the first stage is relatively independent of our understanding of visual perception, the viewing transformations are based on the fact that we are able to perceive two-dimensional patterns - pictures - as reasonably accurate representations of three-dimensional objects. We can evaluate the quality of modeling the propagation of light objectively, by comparing calculated photometric values with experimental measurements. For viewing transformations the quality is much more subjective.


a.

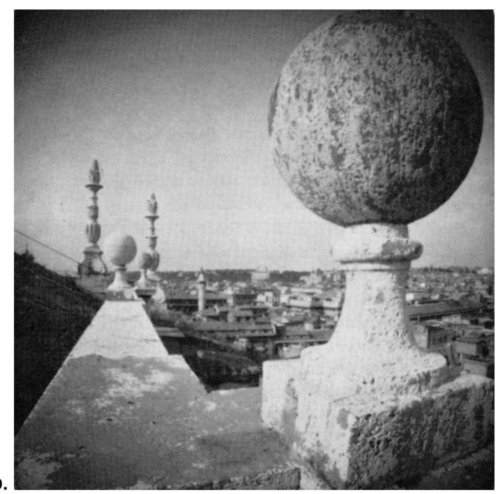

b.

Figure 1. a. Wide-angle pinhole photograph taken on the roof of the Church of St. Ignatzio in Rome, classical example of perspective distortions from [Pir70]; reprinted with the permission of Cambridge University Press. b. Corrected version of the picture with transformation applied.

The perspective projection [1] is the viewing transformation that has been primarily used for producing realistic images, in art, photography and in computer graphics.

One motivation for using perspective projection in computer

[1] By perspective projection or *linear perspective* we mean either central projection or parallel projection into a plane.

graphics is the idea of *photorealism*: since photographic images have one of the highest degrees of realism that we can achieve, perhaps realistic rendering should model the photographic process.

But the intuitive concept of realism in many cases differs from photorealism: photographic images, are often perceived as distorted (note the shape of the sphere in Fig. 1a.) On the other hand, some paintings, while using perspective projection, contain considerable deviations from it (Fig. 2). These paintings, however, are perceptually correct and realistic.

In this paper we derive viewing transformations from some basic principles of perception of pictures rather than by modeling a particular physical process of picture generation. Our approach is based on formalizations of desirable perceptual properties of pictures as mathematical restrictions on viewing transformations.

The main result (Section 5.1) allows us to construct usable families of transformations; it is a decomposition theorem which states that under some assumptions, any perceptually acceptable picture-independent viewing transformation can be decomposed into a perspective or parallel projection and a two-dimensional transformation.

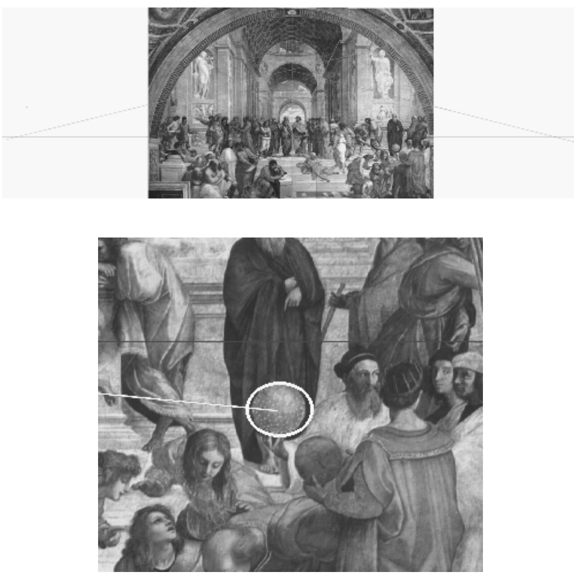


Figure 2. "School of Athens" by Rafael. (©1994-95 Christus Rex, reproduced by permission) It is possible to reconstruct the center of projection from the architectural details. The calculated image of the sphere in the right part of the picture is an ellipse with aspect ratio 1:1.2, while the painting is a perfect circle.

Our approach allows us to achieve several goals:

- We construct new viewing transformations that reduce distortions that appear in perspective projection images. It turns out that some of these transformations can be implemented as a postprocessing stage (Equation 1 , pseudocode in Section 7) after perspective projection and can be applied to existing images and photos (Figs. 1,7,8,9,10.)
- We provide a basis for understanding limitations of two-dimensional images of three-dimensional space; certain perceptual distortions can be eliminated only at the expense of increasing other distortions.
- Our transformation works well in animations and movies.
- Our families of transformations can be modified or extended by adding or removing auxiliary perceptual requirements; this provides a general basis for constructing pictures with desirable perceptual properties.

The transformations that we propose may have a number of applications: creation of computer-generated wide-angle pictures and wide-angle animations with reduced distortion, and correction of photographic images and movies.

Related Work Considerable data on picture perception have been accumulated by experimental psychologists; overviews can be found in [Kub86], [Hag80]. Computer graphics was influenced by the study of human vision in many ways: for example, RGB color representation is based on the trichromatic theory of color perception and anti-aliasing is based on various observations in visual perception.

Principles of the perception of color have been applied to computer graphics [TR93]. A curvilinear perspective system based on experimental data is described in [Rau86].

Limitations of perspective projection are well known in art and photography. [Gla94] mentions the limitations of linear perspective.

As far as we know, this paper is the first attempt to apply perceptual principles to the analysis and construction of viewing transformations in computer graphics.

Outline of the Paper. The paper is organized as follows:[2] In Section 2 we discuss the properties of linear perspective, in Section 3 we formulate our assumptions about perception of pictures and formulate some desirable properties. Section 4 describes some restrictions that we have to impose on the viewing transformation to make it practical. In Section 5.1 we discuss the decomposition theorem for viewing transformations. In Section 5.3 we discuss construction of the 2D component of decomposition,

Section 6 describes a perceptual basis for the choice of the projection component of the decomposition of viewing transformation.

In Section 7 we discuss the implementation issues and we propose some applications of our methods.

Sketches of mathematical proofs can be found in appendices in the CD-ROM version of the paper and in [Zor95].

2 Analysis of linear perspective.

The theory of linear perspective is based on the following construction (Fig. 3). Suppose the eye of an observer is located at the point *O*. Then, the image on the retina of his eye is created by the rays reflected from the objects in the direction of point O. If we put a plane between the observer and the scene, and paint each point on the plane with the color of the ray of light going into O and crossing the plane at this point, the image on the retina will be indistinguishable from the real scene.

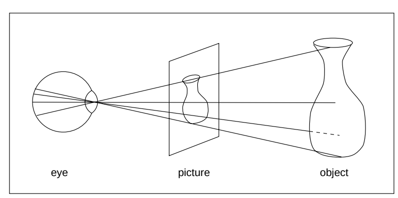


Figure 3. Pictures can produce the same retinal projection as a real object

The above argument contains some important assumptions:

- the observer looks at the scene with one eye, (or is located far enough from the image plane to consider the images in both eyes identical);
- when we look at the picture, the position of the eye coincides with the position of the eye or camera when the picture was made.

In fact, both assumptions for linear perspective are almost never true. We can look at a picture from various distances and directions

[2]The reader who is interested primarily in the implementation can go directly to Equation 1 in Section 5.3 and pseudocode in Section 7

with both eyes, but our perception of the picture doesn't change much in most cases [Hag76]. This property of pictures makes them different from illusions: while stereograms of all kinds should be observed from a particular point, traditional pictures are relatively insensitive to the changes in the viewing point. As the assumptions are not always true, it is not clear why perspective projection should be the preferred method of mapping the three-dimensional space into the plane.

In many cases we observe that perspective projection produces pictures with apparent distortions of shape and size of objects, such as distortions of shape in the margins (Figs. 1,7,8,9,10). These distortions are significantly amplified in animations and movies, resulting in shape changes of rigid bodies.

Leonardo's rule. The fact that linear perspective doesn't always produce pictures that look right was known to painters a long time ago. Leonardo da Vinci [dV70] formulated a rule which said that the best distance for depicting an object is about twenty times the size of the object. It is well-known in photography that in order to make a good portrait the camera should be placed far enough away from the object. In many paintings we can observe considerable deviations from linear perspective which in fact improve their appearance (Fig. 2.)

We conclude that there are a number of reasons to believe that linear perspective is not the only way to create realistic pictures.

3 Properties of pictures.

In this section we will describe our main assumptions about the nature of picture perception and specify the requirements that we will use in our constructions. A more detailed exposition can be found in [Zor95].

Structural features. We believe the that the features of images that are most essential for good representation are the *structural features* such as dimension (whether the image of an object is a point, a curve or an area) and presence or absence of holes or self-intersections in the image. The presence or absence of these features can be determined unambigously.

Most of the visual information that is available to the brain is contained in the images formed on the retina. We will postulate the following general requirement, which will define our concept of realistic pictures: *The retinal projections of a two-dimensional image of an object should not contain any structural features that are not present in any retinal projection of the object itself.* Structural properties of retinal images are identical to the properties of the projections into an arbitrarily chosen plane [Zor95]; our requirement can be restated in more intuitive form:

> A two-dimensional image of an object should have only structural features that are present in some planar projection of the object.

We can identify many examples of structural requirements: the image of a convex object without holes should not have holes in it, the image of a connected object cannot be disconnected, images of two intersecting objects should intersect etc. We choose a set of three structural requirements that we will use to prove the decomposition theorem in Section 5.1.

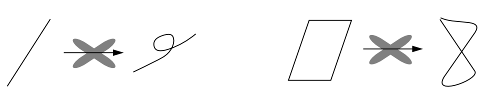


Figure 4. Mappings forbidden by structural conditions 2 and 3

1. The image of a surface should not be a point.
2. The image of a part of a straight line either shouldn't have self-intersections ("loops") or else should be a point (Fig. 4).
3. The image of a plane shouldn't have "twists" in it. This means that either each point of the plane is projected to a different point in the image, or the whole plane is projected onto a curve (Fig. 4).

We will call these conditions *structural conditions 1, 2 and 3*. Note that these requirements are quite weak: we don't require that features of some particular planar projection are represented; we just don't want to see the features that are not present in *any* projection.

Desirable properties. Next, we formulate some requirements that are not as essential as the structural ones; the corresponding features of the images can be varied continuously and can be changed within some intervals of tolerance. Examples of such features include relative sizes of objects, angles between lines, verticality. We will refer to these properties as "desirable properties." We will use two of them which we consider to be the most important. One of the most restrictive desirable properties is the following one:

Zero-curvature condition. *Images of straight lines should be straight.*

Note that this is different from the structural requirement 2 above, which is weaker. However, as we can judge straightness of lines only with some finite precision, some deviations from this property can be tolerated.

Another requirement is based on the following observation: the images of objects in the center of the picture never look distorted, given that the distance to the center of projection is large compared to the size of the object (Leonardo's rule). We will call perspective projections into the plane perpendicular to the line connecting the center of projection with the object *direct view projections*. Then the requirement eliminating distortions of shape can be stated as follows:

Direct view condition *Objects in the image should look as if they are viewed directly – as they appear in the middle of a photograph.*

Unfortunately, as we will see later, the two properties formulated above cannot be satisfied exactly at the same time.

We found several other requirements (foreshortening of objects, relative size of objects, verticality) to be of importance, but having much larger tolerance intervals. We will discuss their significance in Section 7.

4 Technical requirements

To narrow down the area of search for perceptually acceptable viewing transformations we are going to specify several additional technical requirements. They don't have any perceptual basis and are quite restrictive; however, they make the task of constructing viewing transformations manageable and the resulting transformation can be applied to a wide class of images.

1. We need a parametric family of viewing transformations so that an appropriate one can be chosen for each image.
2. The number of parameters should be small, and they must have a clear intuitive meaning.
3. The mapping must be sufficiently universal and should not depend on the details of the scene too much.

5 Derivation of viewing transformations

In the following sections we formalize the perceptual and technical conditions that were stated above and use them to prove that any viewing transformation that satisfies the structural conditions and technical conditions for any image can be implemented as a perspective projection and subsequent transformation of the picture plane. We show that direct view and zero curvature properties cannot be exactly satisfied simultaneously. We introduce quantitative measures of corresponding distortions and describe a simple parametrized family of transformations (Equation 1) where values of parameters correspond to the tradeoff between the two types of distortion. This family of transformation is close to optimal in a sense described in Section 5.3 and is easy to implement (Section 7).

5.1 General structure of viewing transformations

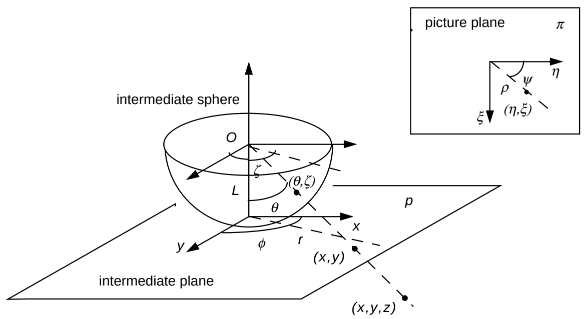


Figure 5. Coordinate systems

In this section we present a decomposition theorem derived from structural conditions 1 -3 (Section 3) and technical requirements of the previous section. The seemingly weak structural conditions 1-3 turn out to be quite restrictive if we want to construct image mappings that don't depend on the details of any particular picture, specifically, on the presence of lines or planes in any particular point of the depicted scene.

Applicaton of these requirements to *all* possible lines and planes allows us to prove that the viewing transformation with no "twists" in the images of planes and no "loops" or "folds" in the images of lines should be a composition of perspective or parallel projection and a one-to-one transformation of the plane.

In order to formulate the precise result let's introduce some definitions and notation:

We will use $x, y, ..$ for the points in the domain of a viewing transformation (a volume in 3D space), and $\xi, \psi...$ for the points in the range (a point in the picture plane).

By a *line segment* we mean any connected subset of a line.

A viewing transformation maps many points in space to the same point in the picture plane:

Definition 1 *The set of all points of the domain of a mapping that map to a fixed point ξ is called the* **fiber** *of the mapping at the point ξ.*

In our case, fibers typically are curves in space that are mapped into single points in the picture plane.

Consider a viewing transformation which is a continuous mapping P of a region of space to the picture plane, more particularly, of an open path-connected domain $V \subset \mathbf{R}^3$ to an open subset of $\mathbf{R}^2$, satisfying the following formalizations of structural conditions:

1. The mapping of any line l to its image $P(l)$ is one-to-one everywhere or nowhere on l.
   This condition prevents "loops" in the images of lines. It is more restrictive: it doesn't allow not only "loops" but also "folds", that is, situations when each point in $P(l)$ is the image of at least two points of l.
2. The dimension of the fiber dim $P^{-1}(\xi) = 1$ for all ξ in the range of P, and all the fibers are path-connected.
   This condition prevents mapping of regions of space to single points and continuous regions to discontinuous images.
3. the mapping of a subset of a plane m to the image $P(m)$ is one-to-one everywhere or nowhere.
   This condition prevents "twists" in the images of planes.

Then our theorem can be stated as follows:

Theorem 1 *For any viewing transformation P, satisfying the conditions above, there is a perspective projection Π such that the fibers of P are subsets of fibers of Π.*

An outline of the proof is given in Appendix A.

Our practical applications are based on the following corollary:

Corollary 1 *Let a viewing transformation P satisfy the assumptions of Theorem 1 and Π be the corresponding perspective projection. If Π is a central projection, assume, in addition, that the region V lies entirely in one half-space with respect to a plane going through the center of Π. Then P can be decomposed in two ways:*

– as a composition of a perspective projection Π_{plane} into a plane followed by a transformation T_{plane} of the plane ,

– as a projection into a sphere Π_{sphere} followed by one-to-one mapping T_{sphere} of the sphere into the picture plane .

It is not true that any picture satisfying only structural conditions (without additional technical requirements) should be generated with an image mapping which has this particular decomposition, because for a particular scene the structural conditions have to be satisfied only by the objects that are actually present in it. It also should be noted that our theorem is an example of a large number of statements that can be proven given some specific choice of structural conditions. We believe that our choice is reasonable for many situations, but it is quite possible that there are cases when least restrictive requirements are sufficient and larger families of transformations can be considered.

While the choice of possible viewing transformations is considerably restricted by this theorem, there are still several degrees of freedom left:

- 2D mappings T_{plane} and T_{sphere} can be any continuous mappings.
- We can choose the center of projection for the first part of the decomposition; it is important to note that the theorem places no restrictions on the position of this center. For example, in an office scene it can be located outside the room, which is impossible for physical cameras.
- If we can split our scene into several disconnected domains (for example, foreground and background), the viewing transformation can be chosen independently for each connected part of the scene. However, separation of the space into several path connected domains introduces dependence of the transformation on the scene.

In the next sections we will consider how we can use these degrees of freedom to minimize the perceptual distortions.

5.2 Formalization of desirable conditions

In this section we formalize the conditions listed in Section 5.1, to apply them to the construction of viewing transformations. We will find error functions for both conditions that can be used as a local measure of distortion and error functionals that measure the global distortion for the whole picture.

Let's establish some notation for the viewing transformations that satisfy the conditions of the theorem.

$$\begin{array}{ccc} V \subset \mathbf{R}^3 & \xrightarrow{\Pi_{plane}} & W_{plane} \subset \mathbf{R}^2 = p \\ \downarrow \Pi_{sphere} & \searrow P & \downarrow T_{plane} \\ W_{sphere} \subset \mathbf{S}^2 & \xrightarrow{T_{sphere}} & \mathbf{R}^2 = \pi \end{array}$$

We will consider viewing transformations $P: V \subset \mathbf{R}^3 \rightarrow \mathbf{R}^2$, from an open connected domain V into the *picture plane* π, which are compositions of the projection Π_{plane} from the center O into the *intermediate plane* p and a mapping $T_{plane} : W_{plane} \subset \mathbf{R}^2 \rightarrow \mathbf{R}^2$, $W_{plane} = \Pi_{plane}(V)$, We can choose the plane p so that that the distance from O to the plane is L.

P can be also represented as a composition of central projection from O into the sphere of radius L with center at O (*intermediate sphere*) $\Pi_{sphere} : V \subset \mathbf{R}^3 \rightarrow \mathbf{S}^2$ and some mapping $T_{sphere} : W_{sphere} \subset \mathbf{S}^2 \rightarrow \mathbf{R}^2$, $W_{sphere} = \Pi_{sphere}(V)$, We will assume that the image of V in the sphere belongs to a hemisphere.

Let's introduce rectangular coordinates (x,y) and polar coordinates (r, ϕ) on the plane p; rectangular coordinates (η, ξ) and polar coordinates (ρ, ψ) in the plane π. On the sphere we will choose angular coordinate system (θ, ζ), and local coordinates in the neighborhood of a fixed point (θ_0, ζ_0): $u = L(\theta - \theta_0), v = L(\zeta - \zeta_0)\sin\theta_0$ (Fig. 5).

The correspondence between T_{plane} and T_{sphere} is given by the mapping $\mathbf{S}^2 \rightarrow \mathbf{R}^2$: $\phi = \zeta$, $r = \tan\theta$.

Curvature error function Formally the restriction on images of line segments from section 3 can be expressed as follows:

Images of line segments should have zero curvature at each point.

We will call this requirement the *zero-curvature condition*. Curvature of the image of a line at a fixed point gives us a measure of how well the viewing transformation satisfies the zero-curvature condition.

If we consider the decomposition of $P = T_{plane} \circ \Pi_{plane}$, we can observe that Π_{plane} satisfies the zero-curvature condition. Therefore, we have to consider only the mapping T_{plane}. We will denote components of $T_{plane}(x, y)$, which is a point in the picture plane, by $(\eta(x, y), \xi(x, y)$

The curvature depends not only on the point but also on the direction of the line, whose image we are considering. As an error function for the zero-curvature condition at a point x we will use an estimate of the maximum curvature of the image of a line going through x.

It can be shown (see Appendix B on CD-ROM, [Zor95]) that the curvature

$$|\kappa| \leq \frac{\sqrt{|\eta_{xx}|^2 + |\eta_{yy}|^2 + 2\,|\eta_{xy}|^2 + |\xi_{xx}|^2 + |\xi_{yy}|^2 + 2\,|\xi_{xy}|^2}}{\frac{1}{2}((A + C) - \sqrt{(A - C)^2 + 4B^2})}$$

where $A = (\eta_x)^2 + (\xi_x)^2$, $B = \eta_x\eta_y + \xi_x\xi_y$, $C = (\eta_y)^2 + (\xi_y)^2$. We use a characteristic size of $T_{plane}(W)$ (corresponding to the size of the picture for perspective projection) R_0 as a scaling coefficient to obtain a dimensionless error-function. We also use the square of the curvature estimate to make the expression simpler:

$$K(T_{plane}, x, y) = R_0^2 \frac{|\eta_{xx}|^2 + |\eta_{yy}|^2 + 2\,|\eta_{xy}|^2 + |\xi_{xx}|^2 + |\xi_{yy}|^2 + 2\,|\xi_{xy}|^2}{\frac{1}{4}\left((A + C) - \sqrt{(A - C)^2 + 4B^2}\right)^2}$$

If we set $K(T_{plane}, x, y) = 0$, we can see that the all the second derivatives of η and ξ should be equal to zero, therefore, T_{plane} should be a linear transformation. This coincides with the fundamental theorem of affine geometry which says that the only transformations of the plane which map lines into lines are linear transformations.

Direct view error function. In order to formalize the direct view condition we consider mappings which are locally equivalent to direct projection as defined in Section 3. We can observe that the projection onto the sphere is locally a direct projection. Therefore, if we use the decomposition $P = T_{sphere} \circ \Pi_{sphere}$ we have to construct the mapping T_{sphere} which is locally is as close to a similarity mapping as possible. Formally, it means that the differential of the mapping T_{sphere}, which maps the tangent plane of the sphere at each point x to the plane $T_{f(x)}\mathbf{R}^2 = \mathbf{R}^2$ coinciding with the picture plane π at the point $f(x)$, should be close to a similarity mapping. The differential $DT_{f(x)}$ can be represented by the Jacobian matrix J of the mapping T_{sphere} at the point x. A nondegenerate linear transformation J is a similarity transformation if and only if $|Jw|/|w|$ doesn't depend on w.

If this ratio depends on w, then we define the direct view error function to be

$$D(T_{sphere}, \theta, \zeta) = \left| \max\left(\frac{|Jw|^2}{|w|^2}\right) / \min\left(\frac{|Jw|^2}{|w|^2}\right) - 1 \right|$$

can be used as the measure of "non-directness" of the transformation at the point (for more detailed discussion see [Zor95].).

It can be shown (see Appendix B on CD-ROM, [Zor95]) that

$$D(T_{sphere}, \theta, \zeta) = \frac{(E + G) - \sqrt{(E - G)^2 + 4F^2}}{(E + G) + \sqrt{(E - G)^2 + 4F^2}} - 1$$

where $E = (\eta_u)^2 + (\xi_u)^2$, $F = \eta_u\eta_v + \xi_u\xi_v$, $G = (\eta_v)^2 + (\xi_v)^2$.

Using the correspondence between intermediate sphere and plane we can write D as a function of T_{plane}, x and y.

Global Error functionals We want to be able to characterize the global error for each of the two perceptual requirements. We can use a norm of the error functions D and K as a measure of the global error. The choice of the norm can be different: the L^1-norm corresponds to the average error, the *sup*-norm corresponds to the maximal local error. In the first case, the error functionals are defined as

$$\mathcal{K}[T_{plane}] = \int\int_W K(T_{plane}, x, y)dxdy,$$
$$\mathcal{D}[T_{plane}] = \int\int_W D(T_{plane}, x, y)dxdy.$$

In the second case,

$$\mathcal{K}[T_{plane}] = \max_{(x,y)\in W} K(T_{plane}, x, y) \qquad \mathcal{D}[T_{plane}] = \max_{(x,y)\in W} D(T_{plane}, x, y)$$

.

5.3 2D Transformation

We are going to use the error functionals defined in Section 5.2 to construct families of transformations by solving an optimization problem. We consider only the 2D part of the decomposition of viewing transformation assuming that the perspective projection is fixed. Structural conditions suggest only that it be continuous and one-to-one.

Optimization problem. The error functionals defined above depend on the domain W_{plane} and the planar transformation T_{plane}. The first parameter is defined by the projection part of the viewing transformation, so we will assume it to be fixed now. In this case, functions satisfying $\mathcal{K}[T_{plane}] = 0$ are linear functions and the only functions satisfying $\mathcal{D}[T_{plane}] = 0$ are those derived from conformal mappings of the sphere onto the plane. Unfortunately, these two classes of functions don't intersect.

In this case for a given level μ for either error functional $\mathcal{K}$ or $\mathcal{D}$ we minimize the level of the other. The corresponding optimization problems are

$$\mathcal{K}[T_{plane}] = \min,\ \mathcal{D}[T_{plane}] = \mu \quad \text{or} \quad \mathcal{K}[T_{plane}] = \mu,\ \mathcal{D}[T_{plane}] = \min$$

These optimization problems are equivalent and can be reduced [Zor95]. to an unconstrained optimization problem for the functional $\mathcal{F}[T_{plane}] = \mu\mathcal{K}[T_{plane}] + (1 - \mu)\mathcal{D}[T_{plane}]$), where μ represents the desired tradeoff between two functionals: for $\mu = 0$ we completely ignore the zero-curvature condition, for $\mu = 1$ the direct view condition.

We also have to specify the boundary conditions in order to make the problem well-defined. This can be done by fixing the frame of the picture, that is, the values of T_{plane} on the boundary of W_{plane}.

We will consider solutions of this optimization problem in the next section.

Error functions for transformations with central symmetry From now on we will restrict our attention to transformations that also have central symmetry. This assumption allows distribution of the error evenly in all directions in the picture. The advantage of this additional restriction is a considerable simplification of the problem. The disadvantage is that real pictures seldom have this type of symmetry and, therefore, non-symmetric transformation might result in better images. We will discuss a way to create nonsymmetric transformations in Section 5.4.

In polar coordinates transformation T_{plane} can be written as $\rho = \rho(r) \quad \psi = \phi$. In this case we get the following simplified expressions for the error functions

$$K(\rho, r) = R_0^2 \frac{\frac{3}{r^2}\left(\frac{\rho}{r} - \rho'\right)^2 + \rho''^2}{\min\left(\frac{\rho^2}{r^2}, \rho'^2\right)^2}$$

$$D(\rho, r) = \frac{\max(\rho'^2(1+r^2)^2, \rho^2(1+\frac{1}{r^2}))}{\min(\rho'^2(1+r^2)^2, \rho^2(1+\frac{1}{r^2}))} - 1$$

We note that in both cases the dependence on the angular coordinate completely disappeared, so now the problem is one-dimensional. We did not use symmetry in our derivation for the general expression for $\mathcal{K}$; absence of dependence on the angle in the formulae above suggests that our bounds are quite tight.

In order for the problem to have a solution, the boundary conditions should have the same type of symmetry. We can take V to be the cone with angle at the apex θ_0. In this case $W = \Pi(V)$ will be a circle of radius $R = L \tan\theta_0$. The corresponding boundary conditions are $\rho(R) = 1$, $\rho(0) = 0$ (from continuity). Here we assume that the radius of the picture $P(V)$, corresponding to R_0 in Section 5.2, is 1.

Now there are unique functions ρ satisfying $K = 0$ or $D = 0$. For K it is obvious: $\rho_K = r/R$. For D it is $\rho_D(r) = \rho_{1D}(r)/\rho_{1D}(R)$, where $\rho_{1D}(r) = \sqrt{r^2+1} - 1/r$

The solutions of the optimization problem will form a parametric family $\rho(\mu, r)$ and $\rho(0, r) = \rho_D(r)$, $\rho(1, r) = \rho_K(r)$.

We consider solutions for the *sup* norm, which is more appropriate from perceptual point of view: we are guaranteed that the distortion doesn't exceed a specified amount.

Now we can state the optimization problem that we have to solve:

Minimize the functional

$$\mathcal{F}[\rho] = \max_{[0\ R]} F(\rho, r)$$

subject to boundary conditions $\rho(0) = 0$, $\rho(R) = 1$, $\rho''(0) = 0$, *where*

$$F(\rho, r) = \mu K(\rho, r) + (1-\mu)D(\rho, r)$$

Solving a minimization problem of this type (Chebyshev minimax functional) is in general quite difficult. We found the lower estimate for the values of $\mathcal{F}$, and numerically approximated the optimal solution. It turns out that the values of $\mathcal{F}$ for linear interpolation between solutions for $\mathcal{K} = 0$ and $\mathcal{D} = 0$ are close to the optimal values.

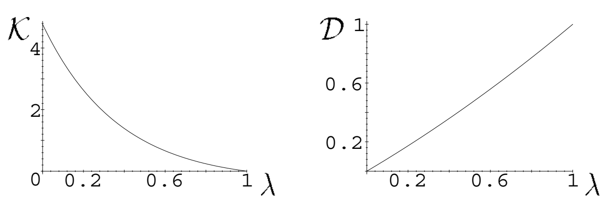


Figure 6. Functionals $\mathcal{K}$ and $\mathcal{D}$ for ρ given by Equation 1 as functions of λ for the field of view 90°. Note that when $\lambda = 1$, there is no curvature distortion. When $\lambda = 0$, then there is no distortion of shape. For a given μ, we can find λ that will approximately minimize the functional $\mathcal{F}$.

It appears that for practical purposes linear interpolation can be used. The resulting transformations have the following form:

$$\rho(r) = \lambda r/R + (1-\lambda)\frac{R(\sqrt{r^2+1}-1)}{r(\sqrt{R^2+1}-1)}; \quad \psi = \phi \quad (1)$$

where the original image is represented in polar coordinate system (r, ϕ), the transformed image in polar coordinate system (ρ, ψ).

5.4 Generalization to non-symmetric cases

We can use Equation 1 to construct more general transformations by replacing a constant λ with a λ depending on the angle. In this case we can choose the balance between direct view and zero curvature conditions to be different for different directions. First, an initial constant value of λ is chosen for the whole image. Then λ is specified for a set of important directions and then interpolated for the rest of the directions. (Figure 7c). Making λ dependent on the radius and angle is more difficult, but possible; we leave this as future work.

6 Choice of viewing transformation

In the previous section we obtained an analytical expression for a family of viewing transformations parameterized by L and λ. The distance L from the center of projection to the intermediate plane p determines Π_{plane}, and λ determines the tradeoff between the zero-curvature and direct-view conditions.

We need to choose both parameters for a particular scene or image. As we have mentioned before, in our approach the center of projection need not be the position of a hypothetical camera or observer; we are free to choose it using perceptual considerations. However, we are restricted in our choice by the content of the picture that we want to obtain. In many cases, the most important constraint is the amount of *foreshortening* that we want to have across the scene. By the amount of foreshortening we mean the desired ratio of sizes of identical objects placed in the closest and most remote part of the scene (for example, human figures in the foreground and background of Fig. 9a). This ratio can be small for scenes which contain only objects of comparable size placed close to each other, such as the office scene (Fig. 9), and should be large for scenes with landscape background (Fig. 10).

According to [HEJ78] people typically prefer pictures with a small amount of foreshortening in individual objects. The behavior of the error functionals is in agreement with this fact: as we move the center of projection away from the intermediate plane ($L \to \infty$), the size of the intermediate image W_{plane} goes to 0 and it is possible to show that both direct view and zero-curvature error functionals decrease. However, a total absence of foreshortening produces distortion (Fig. 9b). The best choice of the center of projection typically corresponds to the field of view in the range 10 to 50 degrees. When such a choice is possible, we can achieve reasonably good results simply by choosing a small field of view and taking λ to be equal to 1 (Fig. 9c).

There are some types of scenes, however, that don't allow us to choose small fields of view. If we try to decrease the field of view in some scenes, either parts of the scene are lost, or the amount foreshortening becomes too close to 1 and objects in the foreground become too small. (Fig. 10c,d).

In this case, we can choose the 2D transformation by varying λ to achieve the appropriate balance between two types of distortion that we described. We choose a "global" λ for the whole image; if parts of the image still look distorted, we can make additional corrections in various parts of the image by varying λ as described in Section 5.4. (Fig 10b, Fig. 7c).

7 Implementation and Applications

Implementation. The implementation of our viewing transformations is straightforward. The Π_{plane} projection practically coincides with the standard perspective/parallel projection. There is, however, an important implementation detail that is absent in some systems. As we mentioned before, our center of projection need not coincide with the position of the camera or the eye. It is chosen according to perceptual requirements. For instance, it can happen that the most appropriate center of projection for an office scene is outside the room. In these cases it is necessary to have a mechanism for making parts of the model invisible (these parts of the model should participate in lighting calculations but should be ignored by the viewing transformation). This can be done using clipping planes.

The 2D part of the viewing transformation T_{plane} can be implemented as a separate postprocessing stage. The advantage of such an implementation is that it allows us to apply it to any perspective image, computer-generated or photographic. The only additional information required is the position of the center of projection relative to the image. The basic structure of the implementation is very simple:

```
for all output pixels (i, j) do
  r := sqrt(i^2 + j^2)
  setpixel( i, j,
      interpolated_color( ρ^{-1}(r)i/r, ρ^{-1}(r)j/r))
end
```

The inverse function ρ^{-1} can be computed numerically with any of the standard root-finding methods, such as those found in *Numerical Recipes* [PFTV88]. The interpolated_color(x, y) function computes the color for any point (x, y) with real coordinates in the original image by interpolating the colors of the integer pixels.

The position of the center of projection is usually known for computer-generated images, but is more difficult to obtain for photos. For photos it can be calculated if we know the size of the film and focal distance of the lens used in the camera. Alternatively, it can be computed directly from the image if there is a rectangular object of known aspect ratio present in the picture [Zor95].

Applications. Examples of applications of our viewing transformations were mentioned throughout the paper. We can identify the following most important applications:

- Creation of wide-angle pictures with minimal distortions. (Figs. 9,10).
- Reduction of distortions in photographic images. (Figs 7,8).
- Creation of wide-angle animation with reduced distortion of shape.
- A better alternative to fisheye views. Fisheye views are used for making images with extremely wide angle (up to 180 for hemispherical fisheye), when the distortions in linear projection make it impossible to produce any reasonable picture. However, fisheye pictures have considerable distortions of their own. The pictures that we obtain using our transformations look significantly less distorted than fisheye views.
- Zooming of parts of a wide-angle picture: For example, we can cut out a portrait of one of the authors from the transformed image in Fig. 7b, while it would look quite distorted if we had used the original photo (Fig. 7a)

8 Conclusion and Future Work

We developed an approach for constructing viewing transformations on perceptual basis. We demonstrate that two important perceptually desirable requirements are incompatible and there is no unique viewing transformation producing perceptually correct images for any scene. We described a simple family of viewing transformations suitable for reducing distortions in wide-angle images. These transformations are straightforward to implement as a postprocessing stage in a rendering system or for photographs and motion pictures.

As we have mentioned in Section 5.1, Theorem 1 applies only in cases when we consider all possible lines and planes in the scene, not only the ones present in it. Better results can be achieved by introducing direct dependence of the viewing transformation on the objects of the scene.

Possible extensions of this work include considering the dependence of λ in the Equation 1 on r, and conformal transformations of the plane preserving the direct-view property. We also can introduce new perceptually desirable properties and finding new families of transformations that produce optimal images with respect to these properties.

9 Acknowledgements

We wish to thank Bena Currin for her help with writing the software and Allen Concorran for the help with preparing the images. Many thanks to Greg Ward for his RADIANCE system that was used to render the image in the paper. We also thank the members of Caltech Computer Graphics Group for many useful discussions and suggestions.

This work was supported in part by grants from Apple, DEC, Hewlett Packard, and IBM. Additional support was provided by NSF (ASC-89-20219), as part of the NSF/DARPA STC for Computer Graphics and Scientific Visualization. All opinions, findings, conclusions, or recommendations expressed in this document are those of the author and do not necessarily reflect the views of the sponsoring agencies.

References

[dV70] Leonardo da Vinci. *Notebooks I, II*. Dover, New York, 1970.

[Gla94] G. Glaeser. *Fast Algorithms for 3-D Graphics*. Springer-Verlag, New York, 1994.

[Hag76] Margaret A. Hagen. Influence of picture surface and station point on the ability to compensate for oblique view in pictorial perception. *Developmental Psychology*, 12(1):57–63, January 1976.

[Hag80] Margaret A. Hagen, editor. *The Perception of pictures*. Academic Press series in cognition and perception. Academic Press, New York, 1980.

[HEJ78] Margaret A. Hagen, Harry B. Elliott, and Rebecca K. Jones. A distinctive characteristic of pictorial perception: The zoom effect. *Perception*, 7(6):625–633, 1978.

[Kub86] Michael. Kubovy. *The psychology of perspective and Renaissance art*. Cambridge University Press, Cambridge <Cambridgeshire> ; New York, 1986.

[PFTV88] William H. Press, Brian P. Flannery, Saul A. Teukolsky, and William T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, Cambridge, UK, 1988.

[Pir70] M. H. Pirenne. *Optics, painting and photography*. Cambridge University Press, New York, 1970.

[Pon62] L. S. Pontriagin. *The mathematical theory of optimal processes*. Interscience Publishers, New York, 1962.

[Rau86] B. V. Raushenbakh. *Sistemy perspektivy v izobrazitel'nom iskusstve : obshchaia teoriia perspektivy*. Nauka, Moskva, 1986. in Russian.

[TR93] Jack Tumblin and Holly E. Rushmeier. Tone reproduction for realistic images. *IEEE Computer Graphics and Applications*, 13(6):42–48, November 1993. also appeared as Tech. Report GIT-GVU-91-13, Graphics, Visualization & Usability Center, Coll. of Computing, Georgia Institute of Tech.

[Zor95] Denis Zorin. Perceptual distortions in images. Master's thesis, Caltech, 1995.

Figure 7. A wide-angle photograph of a room. a. Original image (approximately 100 ° angle. b.Transformation 1 applied with $\lambda = 0$. c. Generalization of transformation 1, (Section 5.4) applied, λ varies from 0 to 1.0 across the image. Note the correct shape of the head and straightness of the walls.

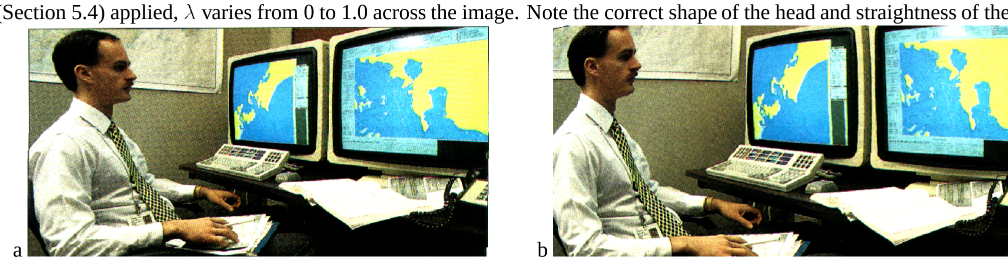


Figure 8. Photo from the article "Navigating Close to Shore" by Dave Dooling ("IEEE Spectrum", Dec. 1994), © 1994 IEEE, photo by Intergraph Corp. 92° viewing angle. a. Original image. b. Transformation 1 applied with $\lambda = 0$.

Figure 9. Shallow scene: model of an office (frames from the video), standard projections. a. 92° viewing angle. b. 3° viewing angle. c. 36° viewing angle, close to perceptually optimal for most people.

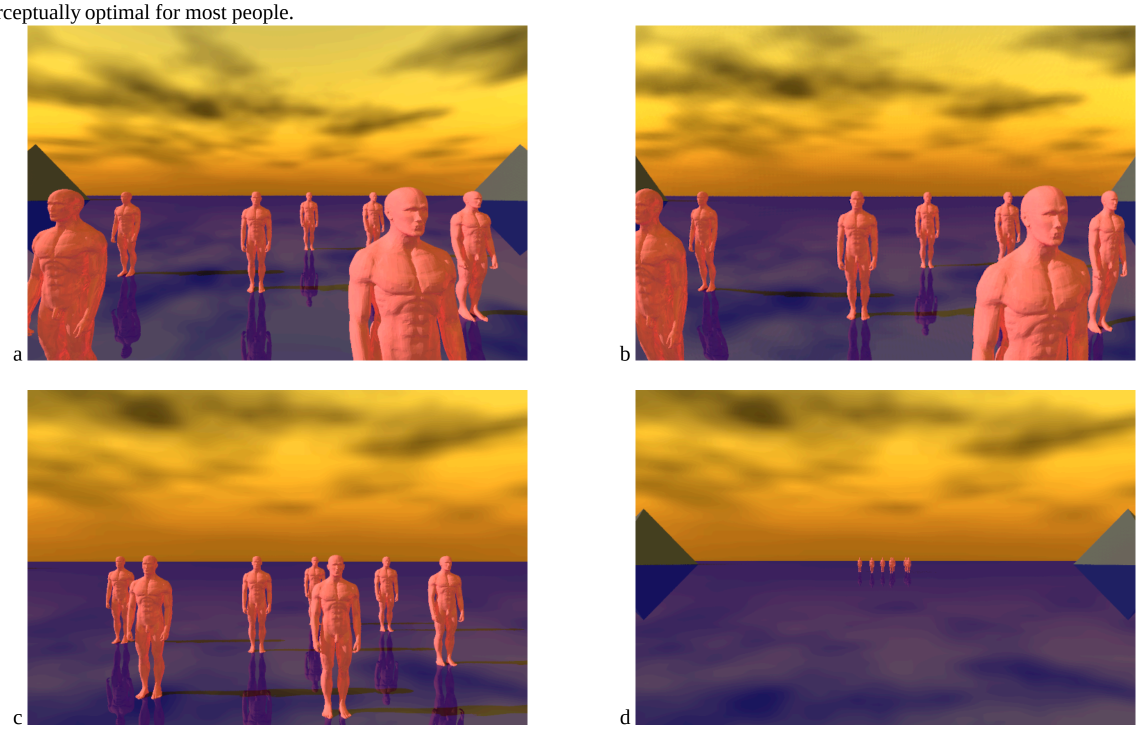


Figure 10. Deep scene, (frames from the video). If we want to have large images of men in the foreground while keeping the images of pyramids in the background, we have to make the angle of the picture wide enough. a. 90° viewing angle. Note the distorted form of the head of the men near the margins of the picture and differences in the shape of the bodies of the men in the middle and close to the margins. b. 90° viewing angle, transformation 1 applied, $\lambda = 0$. c. 60° viewing angle, keeping pyramids in the same position in the picture. d. 60° viewing angle, keeping people in the center in the same position.

Artistic Multiprojection Rendering

Maneesh Agrawala* Denis Zorin† Tamara Munzner*
*Stanford University †New York University

Abstract

In composing hand-drawn images of 3D scenes, artists often alter the projection for each object in the scene independently, thereby generating multiprojection images. We present a tool for creating such multiprojection images and animations, consisting of two parts: a multiprojection rendering algorithm and an interactive interface for attaching local cameras to the scene geometry. We describe a new set of techniques for resolving visibility between geometry rendered with different local cameras. We also develop several camera constraints that are useful when initially setting local camera parameters and when animating the scene. We demonstrate applications of our methods for generating a variety of artistic effects in still images and in animations.

1 Introduction

In computer graphics we typically use a single linear projection – often a perspective projection – to generate a realistic view of a scene. Linear projections achieve this realism at the cost of imposing restrictions on the 2D shape of each object in the image and on the overall composition of the picture. Artists have developed a variety of techniques for composing images of 3D scenes that deviate from the standard perspective projection. One of the most common techniques is to combine multiple projections in a single image.

Artists create such multiprojection images for several reasons, including: expressing a mood, feeling or idea; improving the representation or comprehensibility of the scene; and visualizing information about the spatial relationships and structure of the scene. Multiple projections could similarly enhance computer-generated images and animations, but simple and efficient methods for multiprojection rendering have not been available.

Today, the most common method for creating a multiprojection image requires a combination of 3D rendering and 2D image compositing. The process is a labor intensive cycle that involves rendering multiple views of the scene, transferring the images into a compositing application and then manually merging them into a single multiprojection image. Although some research systems [12, 11] shortcut this cycle by combining rendering and compositing into a single application, resolving visibility between the images remains a manual process. In this paper we present interactive methods for creating multiprojection images and animations. The main technical contributions of our work are new algorithms designed for:

Resolving Visibility: In the multiprojection setting there is no uniquely defined solution to the visibility problem. However, in many cases the user wishes to maintain the visibility ordering of a *master camera* while using different *local cameras* to introduce shape distortions to individual objects. Based on this insight, we propose an algorithm that automatically resolves visibility for most practical cases and allows user adjustments when the automatically computed visibility is not satisfactory.

Constraining Cameras: We suggest a simple and intuitive set of camera constraints allowing the user to choose appropriate projections for a variety of artistic effects. These constraints are particularly effective when initially placing cameras and when animating the scene.

Interactive Rendering: We leverage multipass hardware rendering to achieve interactive rendering rates. The user can immediately see how changing the parameters of any camera or moving any object will affect the final image.

The remainder of this paper is organized as follows. In section 2, we describe artistic uses of multiple projections. After presenting related work in section 3, we describe our multiprojection rendering algorithm with an emphasis on resolving occlusion in section 4. In section 5, we present a set of camera constraints that provide intuitive controls over the camera parameters. Examples of images generated with our system appear in section 6. Finally, section 7 outlines future directions and conclusions.

2 Artistic Uses of Multiple Projections

Using a single projection for an entire scene is restrictive. The ideal projection for one object may not be the best projection for all objects in the scene. Artists have a long history of solving this problem by creating multiprojection images, which generally serve some combination of three functions:

- **Artistic Expression** – Multiple projections help the artist express a mood, feeling or idea.
- **Representation** – Multiple projections improve the representation or comprehensibility of the scene.
- **Visualization** – Multiple projections communicate information about the structure of the objects and spatial relationships in the scene.

In this section, we consider several specific examples of each function. In section 6, we present several images and animations that are based on these examples and were created using our multiprojection rendering tools.

2.1 Artistic Expression

Viewing Anomalies: In Giorgio de Chirico's *The Mystery and Melancholy of a Street*, figure 1(a), the buildings, the van and the ground plane all have different viewpoints. Willats [19] suggests that the melancholy aura is created by the unusual arrangement of the objects which results in an incongruous spatial system. Despite the large disparities between projections, the overall impression of a three-dimensional space remains intact.

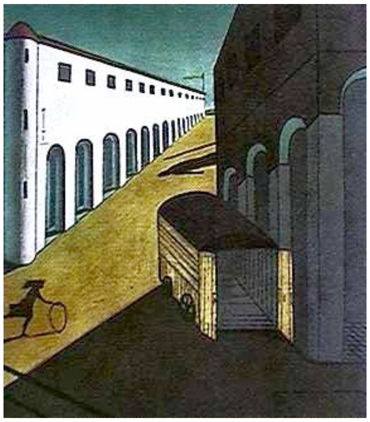
Cézanne similarly incorporates multiple viewpoints in *Still Life with Fruit Basket*, figure 1(b). Loran [13] describes how these viewing distortions generate tension between different planes in the image; the distortions flatten some regions of the picture, while enhancing depth in other regions. He explains that the inconsistencies in projection generate an "emotional nonrealistic illusion of space."

Foreground Elements in Animation: In cel-based animations, moving foreground elements sometimes translate across the image with little to no change in parallax (like cutouts or sprites). Equivalently, as the element translates in the scene, its local camera moves with it, thereby maintaining an identical view of the element from frame to frame. Meanwhile, the background is rendered using its own separate projection. Although fixing the view of foreground elements is done primarily to reuse the same drawing from frame to frame, it has become a stylistic convention in hand-drawn animation since the foreground elements appear flatter and more "cartoony."

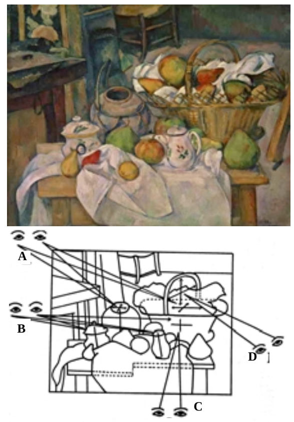
In contrast to the flattened foreground elements, animators often exaggerate the perspective projection of the background to create a deeper, more dynamic environment. However, such a strong perspective could introduce distortions in the foreground of the image. A fixed view of the foreground objects alleviates this problem.

2.2 Representation

Best Views: Certain viewpoints are better than others for comprehending the overall geometric shape of an object. By using multiple viewpoints the artists can present the best views of all objects in the scene. Graham [5] points out that a single viewpoint is often inadequate for large format pictures like murals, frescoes, or billboards. Such images are often placed above standard eye-level and are seen from a much wider range of viewpoints than smaller format pictures. For these reasons, many large format images are created with multiple projections. As

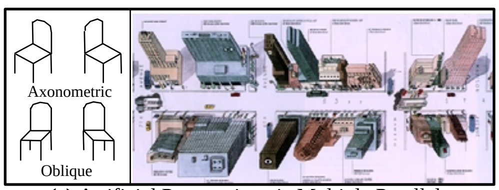


(a) Giorgio de Chirico's *The Mystery and Melancholy of a Street*



(b) Cezanne's *Still Life with Fruit Basket* (Diagram from [13])

(c) Paolo Uccelo's *Sir John Hawkwood*



(e) Artificial Perspective via Multiple Parallel Projections (Chairs from [2], 53rd Street from [6])

(d) Raphael's *School of Athens*

Fig. 1. Multiprojection examples (clockwise labeling)

(a) A different projection is used for each major structure. Note the difference in vanishing points for the buildings, the van, and the ground plane. An exaggerated perspective projection elongates the white building and the ground plane is placed so that the horizon is high up in the image to create the long receding path.

(b) Cezanne incorporates many viewpoints into this still life, of which four are shown in the diagram. Notice how viewpoint **A** is much higher than the other viewpoints, thus the ginger jar and basket appear to be tipped forward. Differences in viewpoint also cause the table to appear to split under the table cloth.

(c) The projection for the base differs from the projection for the horse and rider. In its original setting viewers stood below and to the left of the fresco, so the entire picture was first created for this viewpoint. While the low−set viewpoint was effective for the base, Uccelo found that it exposed too much of the horse's belly and distorted the rider. He repainted the top of the picture and raised the viewpoint of the horse and rider to eliminate the distortion.

(d) The foreground humans would appear distorted if rendered using the projection of the background architecture. Thus, Raphael altered the projections for the humans to give each one a more central projection. Without the correction, the sphere (inset) would appear elliptical rather than circular[10, 21].

(e) Multiple parallel projections can produce an artificial sense of perspective by pointing the receding parallels towards a central vanishing point. Dubery and Willats[2] argue that this technique works better with oblique projections than with axonometric projections. Each building in the illustration of 53rd Street is projected using a different oblique projection. Note how the receding parallels differ for the buildings at the left, right, top, and bottom of the image.

large wall-sized displays become more prevalent [8, 16], we expect that multiprojection rendering techniques will be required to reduce perceptual distortions in images generated for such displays. Paolo Uccelo's fresco of *Sir John Hawkwood* is a well-known large format multiprojection example. He used two projections, one for the base and one for the horse and rider, to produce the best view of both elements, as described in figure 1(c).

Reducing Wide-Angle Distortion: A well known problem with wide-angle, perspective projections is that curved objects located near the edges of the viewing frustum, close to the image plane, can appear unnaturally stretched and distorted. The most common technique for decreasing this distortion is to alter the projection for every object to provide a perceptually "correct" view of each one. The deviations in projection are often subtle and in many cases even the artist, focused on producing a comprehensible image of the scene, may not realize he altered the projections. Kubovy [10] shows that the foreground human figures in Raphael's *School of Athens*, figure 1(d), are inconsistent with the strong central perspective projection of the background architecture. The inconsistencies improve the comprehensibility of the figures and make them easier to recognize.

2.3 Visualization

Artificial Perspective via Multiple Parallel Projections: It is possible to create an artificial sense of perspective using multiple axonometric or multiple oblique projections. The "trick" is to orient the receding parallels of each object towards some pre-chosen vanishing point. In the illustration of 53rd Street [6] (figure 1(e)) the buildings are drawn from above in oblique projection. The receding parallels for each building point towards a central vanishing point, thereby creating the illusion of perspective.

An advantage of such artificial perspective over true perspective projection is that objects do not diminish in size as they recede from the viewer. With oblique projections, one face of each object retains its true shape, allowing the viewer to perform some size and area comparisons. Applying this principle to the 53rd Street example, it is possible to visually compare the 2D area covered by each building. The convergence of true perspective would cause displayed rooftop areas to depend on building height, making such comparisons more difficult.

3 Related Work

In computer graphics, alternatives to standard linear projections have been developed in a variety of contexts. Max [14] shows how to project images onto a curved Omnimax screen, while Dorsey et al. [1] present methods for projecting images onto planar screens for off-axis viewing. Zorin and Barr [21] show how to correct perceptual distortions in photographs by reprojecting them. Glaeser and Gröller [3] use cartographic projection techniques to reduce wide-angle distortions in synthetic images. Both Wood et al. [20] and Rademacher and Bishop [15] describe techniques for smoothly integrating all the views for a given camera path into a single image.

Savranksy et al. [17] describe methods for rendering Escher-like "impossible" scenes. They treat geometric transformations between pairs of objects as constraints on the viewing projection and then solve for a single projection that best meets all the constraints. They compute orthographic viewing projections for several "impossible" scenes. It is unclear how well such an approach would extend to perspective projection. They show that the only way to obtain a single projection for Escher-like scenes is by relaxing the constraints, so they remain valid in the 2D image but not necessarily in 3D object space. In fact, many of the "impossible" scenes only appear impossible from a single viewpoint and animated camera paths would destroy the illusion.

Several systems have been developed to allow more general alternatives to planar-geometric projection. Inakage [9] derives mathematical formulations for three alternatives: curvilinear projection, inverted projection and arbitrary 3D warps. However, he does not consider combining multiple projections within a single image. Löffelmann and Gröller [12] explore the use of curvilinear and non-linear projections to create Escher-like images. Levene [11] investigates artistic uses of curvilinear, inverted, and oblique projections as alternatives to planar perspective. He derives a mathematical framework that unifies several classes of projections. The last two systems are notable because they contain mechanisms for specifying multiple projections within a single image. However, to our knowledge, no system adequately addresses the problem of resolving visibility in the multiprojection setting.

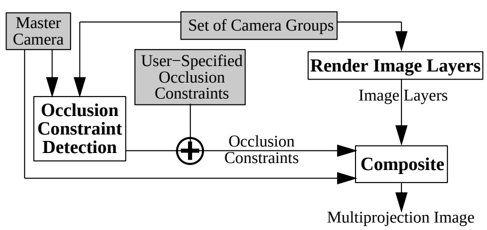
We will show in section 4.1 that resolving visibility is perhaps the most difficult problem in generating multiprojection images because visibility ordering between objects is different under each projection. Our solution is to use a master camera to specify visibility ordering while maintaining shape distortions due to local cameras attached to each object. Although Levene uses the concept of a master camera to compose his multiprojection images, he points out that visibility is not properly resolved with his approach and leaves this as an open problem for future work.

The previous multiprojection rendering systems require the user to directly manipulate the parameters of their generalized projections. Such controls are not always natural and their effect on the final image may be unintuitive. In contrast, we provide several novel camera constraints that allow the user to obtain commonly desired effects with relative ease. Users can also directly specify projection parameters when necessary.

Creating a multiprojection image is far easier with interactive rendering so that the user can immediately see how changing a projection effects the final image. Earlier systems [12, 11] use software ray tracing renderers; therefore, image updates are not interactive for typical image sizes. Our system maintains interactive rendering rates by leveraging graphics hardware. Although this restricts our current implementation to linear planar-projections, our visibility ordering algorithms would work with any invertible projection, including the generalized projection formulations proposed by Inakage, Löffelmann and Gröller, or Levene.

4 Multiprojection Rendering

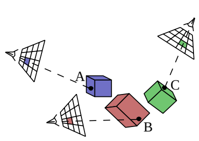
Our multiprojection rendering algorithm includes three computational stages. The input to the algorithm is a set of *camera groups*, each associating a collection of geometric objects with one *local camera*. The user must provide a *master camera* and can optionally specify *object-level occlusion constraints*, both of which are used to resolve visibility. In the block diagram, white boxes represent computational stages of the algorithm, while gray boxes represent user-specified data. The first stage of the algorithm renders each camera group into a separate *image layer*. We then merge the image layers together to form the multiprojection image.

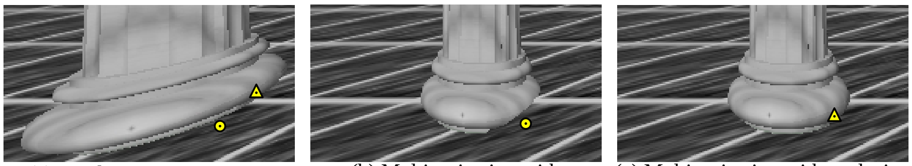


The main difficulty in the compositing stage is the absence of a natural visibility ordering. When visibility ordering differs from camera to camera, there is no unique way to resolve occlusion. Our key observation is that in most multiprojection images, all the local cameras are relatively similar to one another and therefore generate similar visibility orderings. Instead of specifying the occlusion relationship between every pair of objects in the scene, the user simply specifies a master camera (often a local camera doubles as the master). We then use the master camera to resolve visibility through a combination of two automatic techniques: 3D depth-based compositing and standard 2D compositing based on object-level occlusion constraints. If necessary, the user can directly modify the visibility ordering by specifying additional pairwise occlusion relationships between image layers.

4.1 Visibility Ordering

With a single linear projection, visibility is defined unambiguously; the *fibres*, that is, the set of points in 3D space that map to a point on the image surface, are straight lines. For any two points that lie on the same fibre, occlusion is resolved by displaying the point closest to the center of projection. This approach can resolve occlusion whenever the fibres are continuous curves. With multiprojection images, the fibres generally have a more complicated structure since the mapping from 3D space to the image surface can be discontinuous. Suppose, as in the diagram, that points A, B and C project to the same pixel in their local images. The fibre of the multiprojection image at this pixel consists of the union of the three dotted lines. It is difficult to automatically compute a visibility ordering with complicated fibres because no natural ordering exists for the points on different lines.





(a) Single projection master camera view

(b) Multiprojection with depth compositing only

(c) Multiprojection with occlusion constraints and depth compositing

Fig. 2. To reduce the distortion of the column in the single projection image (a) we alter its projection as shown in figure 5(c). In the multiprojection image, the point on the column (triangle) and the point on the floor (circle) coincide. With depth-based compositing alone (b) the floor point "incorrectly" occludes the column point since it is closer to the master camera. However, the column occludes the floor in the master view. Applying this object-level occlusion constraint during compositing yields the desired image (c).

It may be tempting to resolve visibility for the multiprojection image by directly comparing local depth values stored with each image layer. The rendering algorithm would then be quite simple: we could add the local camera projection to the modeling transformation matrix for each object and render the scene using a standard z-buffer pipeline without resorting to layer based compositing. However, this approach would lead to objectionable occlusion artifacts. Suppose our scene consists of a vase sitting on a table. If we simply add the vase's local camera projection into its modeling transform, in most cases the vase will intersect the table. Our algorithm handles these situations more gracefully by using the master camera to impose visibility ordering while employing local cameras to provide shape distortion. In our example, the master camera would be the original projection, in which the vase and table do not intersect, and the local projection would affect only the shape of the vase without affecting visibility.

Given the master camera, we can define an ordering for any set of points in 3D space based on the distance from each point to the master camera viewpoint. To merge the image layers rendered for each camera group, we transform all the pixels in each image-layer into world space using their pixel coordinates and z-values. We then apply the master camera projection to these world space points and use a standard z-buffer test to determine the frontmost point at each pixel of the master camera image. However, figure 2 shows that the results produced by this depth-based approach are not always satisfactory. The problem occurs because visibility is resolved for each pixel independently, yet we want to preserve object-level occlusion constraints, such as "column occludes floor", that occur in the single projection master camera view. In the next section, we describe a simple algorithm for automatically computing these constraints with respect to the master camera. However, we will also show that visibility can not always be resolved using object-level occlusion constraints alone. Thus, the compositing stage of our algorithm combines two approaches. We use object-level occlusion constraints wherever possible and fall back onto depth-based compositing only when the occlusion constraints are ambiguous.

While additional user intervention is not required, the user may explicitly specify occlusion constraints between pairs of objects. These user-defined occlusion constraints are added to the list of constraints computed via the occlusion detection algorithm. Conflicts between user-specified constraints and computed constraints are resolved in favor of the user and the conflicting computed constraints are removed from the list.

4.2 Object-Level Occlusion Constraints

Object-level occlusion constraints are defined for whole objects rather than individual points of the objects. If every point of object A is in front of object B, we say that A occludes B. To compute the occlusion constraints with respect to the master camera, we must determine for each pair of objects A and B whether A occludes B, B occludes A or neither. Our occlusion constraint detection algorithm is based on an algorithm described by Snyder and Lengyel[18].

Occlusion Constraint Detection Algorithm

```
Input: Set of Objects,
       Master Camera
Output: Set of Occlusion Constraints

foreach camera group
  Render object using Master Camera into a separate
  buffer storing (Camera Group ID, depth) per-pixel
foreach pixel
  Sort objects in pixel by depth
  foreach pair of objects in sorted list
    if conflicting occlusion constraint appears in list
      Mark object pair as non-ordered
    else
      Add occlusion constraint to list
```

Object-level occlusion constraints may not provide enough information to merge all the image layers. There are two forms of ambiguity. When the convex hulls of A and B intersect we may find that A occludes B in some regions of the master view, while B occludes A in other regions. Our constraint detection algorithm checks for such binary occlusion cycles and marks the objects as non-ordered. Another type of ambiguity arises when A and B do not occlude one another in the master view at all, but their local image layers do overlap. Our occlusion constraint detection algorithm cannot provide any constraint in this case. We handle these ambiguities during the compositing phase of the algorithm.

4.3 Compositing

The last stage of the algorithm, compositing, combines the multiple image layers produced in the first step into a single image. If two objects map to the same pixel and there is an occlusion constraint between them, we simply use the constraint to determine which object is visible. When no such occlusion constraint is available (as in the cases described in the previous section), we use depth compositing to resolve visibility.

In certain cases, occlusion cycles can pose a problem for our algorithm. Suppose there is a three object cycle[1] in which A occludes B, B occludes C, and C occludes A. Our pixel-by-pixel compositing algorithm will produce the desired multiprojection image regardless of the order in which it considers the objects, unless a single pixel contains all three. While it is possible to detect such loops in the occlusion constraint graph, resolving occlusion in such cases requires an arbitrary choice as to which object in the cycle is visible. In practice, we have found that this problem rarely occurs, and when it does, we provide controls so that the user can choose which object is visible.

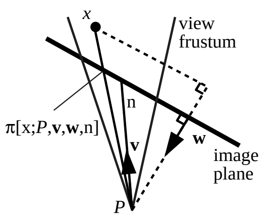
5 Camera Constraints

When creating multiprojection images and animations, it is useful to be able to constrain the motions of an object and its local camera in relation to one another. For example, the user may want to dolly the camera closer to its object without changing the object's size in the image plane. This would produce a close-up, wide angle view of the object, thus increasing its perceived depth without changing its relative size in the image. Such camera constraints are indispensable for multiprojection animations, since direct specification of several dependent cameras is quite difficult. The user might keyframe an object motion and with such constraints in place, the cameras would automatically move in relation to the object to enforce the constraint.

We consider three camera constraints we have found particularly useful. Each constraint is a system of equations, relating the camera parameters and object position before and after modification. The system is typically underconstrained and we can choose some of the camera parameters arbitrarily. Once these parameters are chosen, we use the constaints to determine the other parameters.

Notation: Each equation will be preceded by a label (i.e. pos, dir, size, dist) describing the constraint it imposes. We use $'$ to denote the quantities corresponding to the camera and the objects after modification. We denote the vector length as $\|\mathbf{a}\|$ and the vector dot product as $\mathbf{a}\cdot\mathbf{b}$. We use $\pi[x; P, \mathbf{v}, \mathbf{w}, n]$ to denote the world-space coordinates of the projection of a point x into the image plane with a camera defined by parameters $P, \mathbf{v}, \mathbf{w}, n$. The image plane coordinates of the projection of x are denoted $\pi_1[x; P, \mathbf{v}, \mathbf{w}, n]$ and $\pi_2[x; P, \mathbf{v}, \mathbf{w}, n]$, as shown in figure 3.

[1] We remove binary cycles as described in section 4.2.



Camera parameters:

P:	center of projection
v:	camera view unit vector
w:	image plane unit normal
u:	world space up unit vector
up:	image plane up unit vector
n:	distance to image plane along ***v***
height:	image plane height
width:	image plane width
f:	distance to far plane

Projection Equations

World Space Coordinates:

$$\pi[x;P,\mathbf{v},\mathbf{w},n] = (n\mathbf{v}\cdot\mathbf{w})\frac{x-P}{(x-P)\cdot\mathbf{w}} + P$$

Image Plane Coordinates:

$$\pi_1[x;P,\mathbf{v},\mathbf{w},n] = (\pi[x;P,\mathbf{v},\mathbf{w},n] - n\mathbf{v})\cdot\mathbf{up}$$

$$\pi_2[x;P,\mathbf{v},\mathbf{w},n] = (\pi[x;P,\mathbf{v},\mathbf{w},n] - n\mathbf{v})\cdot(\mathbf{up} \times \mathbf{w})$$

Fig. 3. Camera model and parameters.

5.1 Object-Size Constraint

Normally, as a camera moves towards an object, the projected image size of the object grows and the perspective convergence of the object changes. The goal of the object-size constraint is to keep the object's size and position approximately constant while changing its perspective convergence by dollying the camera towards (or away from) it. We characterize the object's size by the distances between two points on the object, x_1 and x_2. The constraint maintains a constant distance between the projections of the two points and is expressed in the following equation:

size : $\quad \|\pi[x_1;P',\mathbf{v}',\mathbf{w}',n'] - \pi[x_2;P',\mathbf{v}',\mathbf{w}',n']\| = \|\pi[x_1;P,\mathbf{v},\mathbf{w},n] - \pi[x_2;P,\mathbf{v},\mathbf{w},n]\|$

The simplest way to adjust perspective convergence for an object is to change the near parameter n of the camera model while leaving the $height$ and $width$ at the near plane unchanged. Suppose, as in figure 4(a), we move the camera towards the center of the object to a new position $P' = P + (x - P)\Delta t$. We assume that the view vector and image plane normal remain fixed (i.e. $\mathbf{v}' = \mathbf{v}$ and $\mathbf{w}' = \mathbf{w}$), and compute the new near distance n':

$$n' = n\frac{\|\pi[x_1;P',\mathbf{v},\mathbf{w},n] - \pi[x_2;P',\mathbf{v},\mathbf{w},n]\|}{\|\pi[x_1;P,\mathbf{v},\mathbf{w},n] - \pi[x_2;P,\mathbf{v},\mathbf{w},n]\|}$$

In the film *Vertigo*, Hitchcock uses this technique to give the audience a sense of the vertigo as experienced by the main character. Hitchcock simultaneously dollies the camera and adjusts its focal length (i.e. near distance) to maintain the object-size constraint. The change in perspective, with little or no change in object size and position conveys the impression of dizziness and falling.

Another interesting use of the object-size constraint is constructing artificial perspective by combining multiple oblique projections as described in section 2.3. If we enforce the object-size constraint, move the camera out to infinity, and also force the view vector **v** to be parallel to the line between the camera center of projection P and the object center x, we obtain an oblique projection. Although there is no vanishing point for the oblique projections, the receding parallels will point towards the vanishing point of the original perspective projection. By separately applying the constraint to each object in the scene, we combine multiple oblique projections to produce an artificial sense of perspective. We used this approach to create plate 3.

5.2 Fixed-View Constraint

Under a perspective projection, as an object translates parallel to the image plane in world space, different sides of the object are exposed. As the object moves from left to right with respect to the camera center of projection, the intra-object occlusions change.

At times we wish to both maintain a fixed view of an object and move its position in the image plane. The result is an object that translates in the image much like a 2D sprite. However, unlike a normal sprite, an object under the fixed-view constraint continues to move in 3D and can therefore rotate and deform as a 3D body.

As shown in figure 4(b), we must maintain the direction vector between the object center and the camera center of projection. In addition, we would like the size and position of the object in the image plane to be approximately the same as if the camera were not changed. That is, we

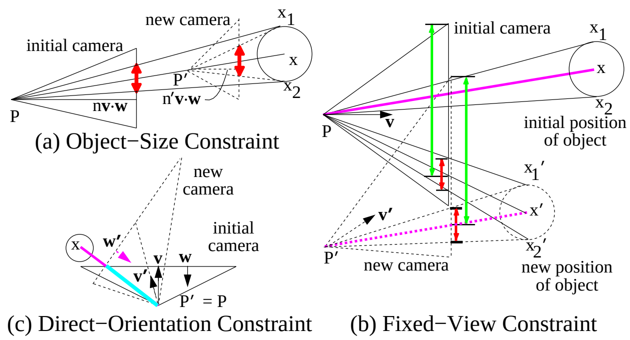


Fig. 4. Camera constraint diagrams. In (a) the image size (red) of the object remains fixed as the camera moves towards it. In (b) as the object translates in world space, its image size (red), image position (green) and the direction between it and the center of projection (magenta) remain fixed. For clarity we introduce a slight separation between the image planes. In (c) the camera is reoriented so the image plane normal $\mathbf{w}'$ matches the direction of the line joining x with P (magenta). The distance (cyan) between the camera and the object's image also remains fixed.

wish to maintain:

$$\textbf{dir}: \quad \frac{x' - P'}{\|x' - P'\|} = \frac{x - P}{\|x - P\|}$$

$$\textbf{pos}: \quad \pi_i[x'; P', \mathbf{v}', \mathbf{w}', n'] = \pi_i[x'; P, \mathbf{v}, \mathbf{w}, n] \quad i = 1, 2$$

$$\textbf{size}: \quad \|\pi[x_1'; P', \mathbf{v}', \mathbf{w}', n'] - \pi[x_2'; P', \mathbf{v}', \mathbf{w}', n']\| = \|\pi[x_1'; P, \mathbf{v}, \mathbf{w}, n] - \pi[x_2'; P, \mathbf{v}, \mathbf{w}, n]\|$$

where, as before, x_1, and x_2 are two points that we use to define the size of the object.

We have found that it is convenient to choose x to be the center of the bounding sphere of an object, and x_1 and x_2 to be the two extremal points of the bounding sphere such that the plane formed by x, x_1 and x_2 is parallel to the image plane. Thus, x_1 and x_2 are the endpoints of an interval centered at x, and their positions with respect to x do not change as the object moves. We assume that both $\mathbf{w}$ and the distance from the center of projection to the image plane remain fixed (i.e. $\mathbf{w} = \mathbf{w}'$ and $n\mathbf{v} \cdot \mathbf{w} = n'\mathbf{v}' \cdot \mathbf{w}$). Then the first constraint equation can be solved for P', and the other two can be solved for $\mathbf{v}'$ in order to determine the new position of the image plane.

5.3 Fixed-Position Constraint

The fixed-position constraint is similar to the fixed-view constraint. Instead of maintaining a particular view of the object, the fixed-position constraint maintains the position of the object in the image plane. As the object translates, different sides of it are exposed, but it remains in the same place in the image plane. While the effect is similar to rotating the camera about the object, this constraint actually produces an off-axis projection. The following constraint ensures that the center of the object remains fixed in the image plane:

$$\textbf{pos}: \quad \pi_i[x; P, \mathbf{v}, \mathbf{w}, n] = \pi_i[x'; P', \mathbf{v}', \mathbf{w}', n'] \quad i = 1, 2$$

The desired effect is that as the object moves in world space, the window of the image plane through which we see the object shifts with it. However, the image plane does not move. Thus, P and $\mathbf{w}$ remain fixed (i.e. $P' = P$ and $\mathbf{w}' = \mathbf{w}$), while $\mathbf{v}$ and n change.

As with the fixed-view constraint, the distance from the center of projection to the image plane remains fixed, so $n\mathbf{v} \cdot \mathbf{w} = n'\mathbf{v}' \cdot \mathbf{w}$. Assuming no change in camera parameters, we first compute the displacement d of the object's image and then use this displacement to determine the center of the image plane for the new camera $n'\mathbf{v}'$.

$$d_i = \pi_i[x'; P, \mathbf{v}, \mathbf{w}, n] - \pi_i[x; P, \mathbf{v}, \mathbf{w}, n] \quad i = 1, 2$$
$$n'\mathbf{v}' = n\mathbf{v} + d_1\,\mathbf{up} + d_2\,(\mathbf{up} \times \mathbf{w})$$

5.4 Direct-Orientation Constraint

In section 2.2, we observed that curved objects can appear extremely distorted under a wide-angle projection. There are several ways to reduce this distortion.

The simplest approach is to use a parallel projection for each distorted object. For example, with a scene consisting of a row of columns, figure 5, we would use the same orthographic camera for all the columns to ensure that they all appear to be the same size. The drawbacks of this approach are that the columns appear flat and that we see each column from exactly the same side.

A better approach is to use the direct-orientation constraint. For each object we create a camera pointed directly at this object. Equivalently, the image plane normal for each local camera is parallel to the direction vector to the object. At the same time, we preserve the position of the object in the image plane. Finally, we try to keep the size of the object approximately constant in the image plane (see figure 4(c)). Instead of using a general equation of the type derived for the object-size constraint, we use a simpler condition requiring that the distance from the camera to the image of the object remains fixed. We obtain the following constraint equations for the camera corresponding to the object located at x:

$$\begin{aligned}
\textbf{dir}: &\quad \mathbf{w}' = \frac{x - P}{\|x - P\|} \\
\textbf{pos}: &\quad \pi_i[x, P, \mathbf{v}', \mathbf{w}', n'] = \pi_i[x, P, \mathbf{v}, \mathbf{w}, n] \quad i = 1, 2 \\
\textbf{dist}: &\quad \|\pi[x, P, \mathbf{v}', \mathbf{w}', n'] - P\| = \|\pi[x, P, \mathbf{v}, \mathbf{w}, n] - P\|
\end{aligned}$$

Assuming that the up vector **u** does not change, these equations uniquely determine $\mathbf{v}'$ and n' when **v** and n are known.

6 Results

We have created several images and animations using our multiprojection rendering system as shown in the color plates, figure 5, and the accompanying video[2]. In all our examples, all occlusion relationships were computed by our algorithm. Plate 1 is our reconstruction of de Chirico's *Mystery and Melancholy of a Street*. The scene is modeled as shown in the plan view thumbnail image. The drop shadows from the building are explicitly modeled as polygons. Interactively adjusting the five local cameras took about an hour and a half. Matching the painting took about 20 minutes and the remainder of the time was spent animating the van to move through the scene. We made extensive use of our camera constraints throughout the process. To place the local camera of the brown building, for example, we initially set the image plane of the camera parallel to the front face of the building. We then used the object-size and fixed-position constraints to interactively adjust the vanishing point for its receding faces until they matched the painting. The van was animated to go around the brown building using a combination of object motion and camera motion. Note that as the van moves from being in front of the brown building to going behind it, the occlusion relationship between the two objects is updated automatically.

The multiprojection still life in plate 2 was inspired by Cézanne's *Still Life with Fruit Basket* and adjusting the 10 local cameras took about an hour for this scene. Plate 3 is shows a comparison of an overhead view of a city rendered using multiple oblique projection to create an artificial sense of perspective in (a), and a true perspective projection in (b). There are two advantages to using multiple oblique projections. First, more of the scene is visible; in the true perspective many building extend beyond the field of view the image and there is more occlusion between

[2] http://graphics.stanford.edu/papers/mpr/video

buildings than in the multiple oblique view. Second, it is easier to judge the relative rooftop sizes, and the area of the block that each building covers in the multiple oblique view. Plate 4 shows some frames from an animation we created using the fixed-view constraint. In parts (a) and (b) we show how the fixed-view constraint improves the composition of the scene. The fixed-view constraint only affects translational motions. Unlike traditional sprites, objects can rotate and deform as 3D bodies as shown in the frames on left side of plate 4.

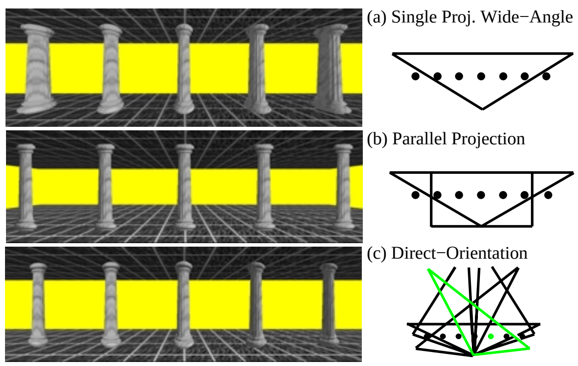


Fig. 5. Reducing the wide-angle distortion of curved objects

The columns in figure 5(a) appear distorted due to a wide-angle camera. If we use a parallel projection for the columns as in figure 5(b) (see camera schematic), the distortion is corrected, but each column appears flat and the lighting is wrong: every column is lit identically. A better approach is to use the direct-orientation constraint to aim each local camera at its column as in figure 5(c). Since the local cameras all have the same center of projection, we see exactly the same parts of each column as in the original view and the lighting is correct because the visibility is unchanged. We have colored one of the columns and its local camera green in the camera schematic to show how the local camera is reoriented.

Our implementation of the multiprojection rendering algorithm uses hardware rendering to maintain an interactive interface. The main bottleneck in the rendering algorithm is this multipass read/write of the framebuffer required before rendering each camera group and resolving visibility. Therefore rendering performance is based on the number of camera groups in the scene and the size of the image. Despite the read/write bottleneck, we are able to achieve interactive framerates for most of our test scenes running on a SGI RealityEngine graphics system. For example, the multiprojection still life contains 10 camera groups and renders at a rate of 3 to 5 frames per second at a resolution of 333 x 256. Other scenes containing fewer cameras render more quickly. The de Chirico scene containing five camera groups renders at 10 - 12 frames per second at a resolution of 402 x 491.

In addition to the interactive interface, we built a script-based animation system for creating keyframed multiprojection animations. Obtaining visually pleasing results with a multiprojection animation is somewhat difficult because animation adds motion parallax as a cue about the spatial relationship between objects. The motion parallax may conflict with the perspective and occlusion cues, disconcerting the viewer. When the entire scene is moved by applying the same transformation to every object in the scene (or equivalently to every camera in the scene), each object will move and scale at a different rate since the local cameras may be very different. We have found that in most cases animating objects in the scene, while holding the background environment fixed, works well. Moving the entire scene tends to only work for relatively small translations and rotations. One notable exception is when all the local cameras have the same center of projection as in the row of columns example in figure 5. In this case, moving the entire scene does not cause any unexpected effects.

7 Future Work and Conclusions

We have presented a multiprojection rendering algorithm as well as a constraint-based camera specification interface. Using this system, it is possible to create a large variety of still and animated visual effects that can not be created with a standard, single projection rendering system.

There are several directions in which to expand this work. As we described in the previous section, not all types of animation produce pleasing results. Moreover, animation requires keyframing the motion over every object by hand. While our camera constraints can make it easier to produce certain types of camera motions, the constraints are designed to produce very specific effects. It would be useful to incorporate a more flexible constraint-based interface for cameras, similar to the design of Gleicher and Witkin [4], in both our camera specification interface and our animation system.

It is unclear how to handle lighting effects like shadows and reflection in the multiprojection setting. We currently assume that all lights are specified in the global scene and each camera group is lit based on its local camera viewpoint. The difficulty arises when we try to cast a shadow from one camera group onto another. Despite these issues, multiprojection images are an effective device for creating a variety of artistic effects.

8 Acknowledgements

Thanks to Pat Hanrahan for giving us insightful feedback throughout the course of this work. Gregory Lam Niemeyer's artistic guidance was invaluable during the early stages of the project.

References

[1] J. O. Dorsey, F. X. Sillion, and D. P. Greenberg. Design and simulation of opera lighting and projection effects. In *Computer Graphics (SIGGRAPH '91 Proceedings)*, volume 25, pages 41–50, July 1991.

[2] F. Dubery and J. Willats. *Perspective and Other Drawing Systems*. Van Nostrand, 1983.

[3] G. Glaeser and E. Gröller. Fast generation of curved perspectives for ultra-wide-angle lenses in VR applications. *The Visual Computer*, 15:365–376, 1999.

[4] M. Gleicher and A. Witkin. Through-the-lens camera control. In E. E. Catmull, editor, *Computer Graphics (SIGGRAPH '92 Proceedings)*, volume 26, pages 331–340, July 1992.

[5] D. W. Graham. *Composing Pictures*. Van Nostrand Reinhold Company, 1970.

[6] S. Guarnaccia. 53rd Street Map. In N. Holmes, editor, *The Best in Diagrammatic Graphics*, pages 174–175. Quarto Publishing, 1993.

[7] A. Hertzmann. Painterly rendering with curved brush strokes of multiple sizes. In *SIGGRAPH 98 Conference Proceedings*, pages 453–460. ACM SIGGRAPH, Addison Wesley, July 1998.

[8] G. Humphreys and P. Hanrahan. A distributed graphics system for large tiled displays. *IEEE Visualization '99*, pages 215–224, October 1999.

[9] M. Inakage. Non-linear perspective projections. In *Modeling in Computer Graphics (Proceedings of the IFIG WG 5.10 Working Conference)*, pages 203–215, Apr. 1991.

[10] M. Kubovy. *The Psychology of Perspective and Renaissance Art*. Cambridge University Press, 1986.

[11] J. Levene. A framework for non-realistic projections. M.eng. thesis, MIT, 1998.

[12] H. Löffelmann and E. Gröller. Ray tracing with extended cameras. *The Journal of Visualization and Computer Animation*, 7(4):211–227, Oct.-Dec. 1996.

[13] E. Loran. *Cezanne's Composition*. University of California Press, 1943.

[14] N. L. Max. Computer graphics distortion for IMAX and OMNIMAX projection. In *Nicograph '83 Proceedings*, pages 137–159, Dec. 1983.

[15] P. Rademacher and G. Bishop. Multiple-center-of-projection images. In *SIGGRAPH 98 Conference Proceedings*, pages 199–206, July 1998.

[16] R. Raskar, G. Welch, M. Cutts, A. Lake, L. Stesin, and H. Fuchs. The office of the future: A unified approach to image-based modeling and spatially immersive displays. *Proceedings of SIGGRAPH 98*, pages 179–188, July 1998.

[17] G. Savransky, D. Dimerman, and C. Gotsman. Modeling and rendering Escher-like impossible scenes. *Computer Graphics Forum*, 18(2):173–179, June 1999.

[18] J. Snyder and J. Lengyel. Visibility sorting and compositing without splitting for image layer decomposition. In *SIGGRAPH 98 Conference Proceedings*, pages 219–230, July 1998.

[19] J. Willats. *Art and Representation: New Principles in the Analysis of Pictures*. Princeton University Press, 1997.

[20] D. N. Wood, A. Finkelstein, J. F. Hughes, C. E. Thayer, and D. H. Salesin. Multiperspective panoramas for cel animation. In *SIGGRAPH 97 Conference Proceedings*, pages 243–250, Aug. 1997.

[21] D. Zorin and A. H. Barr. Correction of geometric perceptual distortion in pictures. In R. Cook, editor, *SIGGRAPH 95 Conference Proceedings*, Annual Conference Series, pages 257–264. ACM SIGGRAPH, Addison Wesley, Aug. 1995.

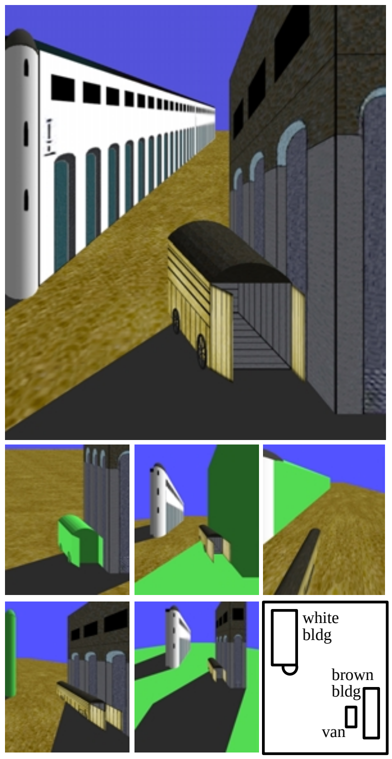
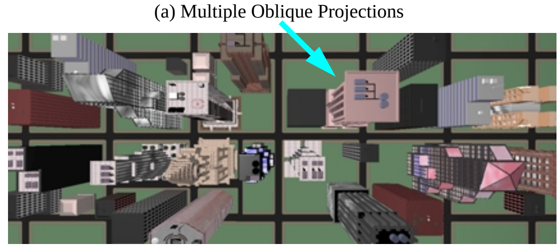


Plate 1. Our reconstruction of Giorgio de Chirico’s *Mystery and Melancholy of a Street*. The thumbnails show the 5 local camera views, with attached geometry highlighted in green.

Plate 2. A multiprojection still life containing 10 camera groups took about an hour to create with our system. The impressionist style painting was created in a post−process using Hertzmann’s [7] image processing algorithm.

(a) Multiple Oblique Projections



(b) True Perspective Projection

Plate 3.Multiple oblique projections create an artificial sense of perspective, but still allow some area comparisons. In (b) the pink building’s rooftop area (arrow) is exaggerated. In (a) it correctly appears to be about the same size as the gray rooftop next to it.

(a) Fixed−View Constraint on Car

(b) Single Projection

Plate 4.Our fixed−view constraint can improve composition and give scenes a "cartoony" feel. In (a) it is possible to see the faces of both characters. In (b) the sitting character’s face is not visible. The constraint only affects translational motion, so objects can rotate and deform as 3D bodies. In the animation frames to the left, the fixed−view constraint is enforced on both cars. In the first and last frame the views of the cars are the same, but when the blue car turns to pass the red car in the middle two frames, we can see the tires rotate and the blue character’s uniform becomes more visible than that of the red character.

The Art and Science of Depiction

Gaze Movement and Focal Points

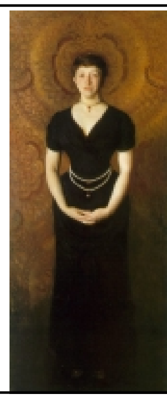
Fredo Durand
MIT- Lab for Computer Science

Focus, gaze

Kathe Kollwitz
Self Portrait
1891-92

Focus, gaze

Sargent



Delacroix

Need for exploration

- Our acuity is not uniform over the visual field
- Resolution is concentrated in the fovea (the ~2 degree region at the center of our retina)
- We need to align the fovea with relevant features
- We explore our visual environment with gaze movements
- Attention is restricted to a small area
- How we then stitch all these observations together is still a mystery

Plan

- Different eye movements
- Visual exploration
- Saliency
- Focal points, composition

Eye movements

- Physiological nystagmus (involuntary)
- Saccade (scan visual field)
- Smooth pursuit (track moving objects)
- Vergence (depth adjustment)
- Vestibular (compensate head movement)
- Optokinetic (in moving environment)

Physiological nystagmus

- Involuntary movement
- All the time
- We need to avoid stabilized images
  - If an image is stabilized on the retina (through a mechanical device like a lens fixed on the eye)
  - It progressively disappears

Saccade

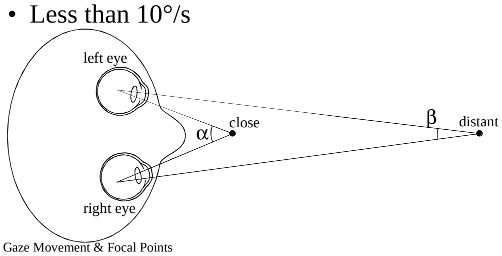
- Used to scan the visual field
- Can be controlled
- The most important for us
- Two phases
  - Ballistic movement: 30 ms and up to 900°/s
  - Fixation ~300ms
- Saccadic suppression
  - No blur is experienced during the ballistic movement
  - We "suppress" our vision while the gaze moves

Smooth Pursuit

- Track moving objects
- Smooth movements
  - Constant feedback and readjustment
- Slower than saccades (max 100°/s)
- Acuity
  - The image of the tracked object remains sharp

Vergence

- For stereo vision
- The eyes align their direction to the object
- Depends on object distance (depth cue)
- Less than 10°/s



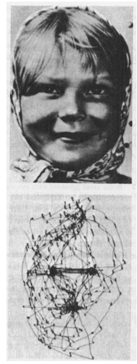
Other movements

- Vestibular
  - compensate head movement
- Optokinetic
  - in moving environment
  - we follow the moving background

Saccadic exploration

- Reading: Javal, 1878
- Images: Yarbus, 1965

- Two important issues:
- Path
- Fixation time



David Hockney's collages

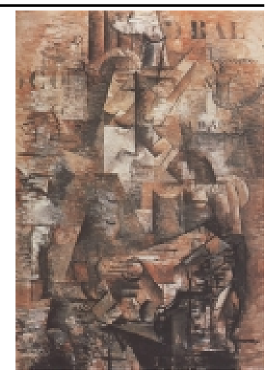
- 1 photo= 1 gaze
- Distorted perspective because saliency

David Hockney's collages

- Temporal too

Gaze movement & cubism

- George Bracque
  Le Portugais
  1911-1912



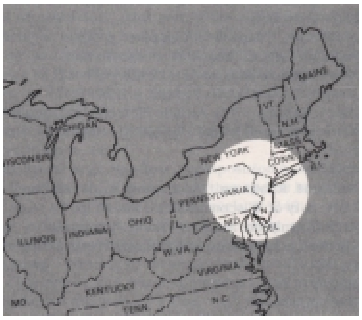
Gaze attraction

- Bottom-up (stimulus-driven)
  - Contrast
  - Color
  - Patterns
- Top-bottom (High-level driven, potentially conscious)
  - Semantic information, familiarity
  - Human beings, eyes
  - Task
  - Personal context

Focal point

- Contrast
- Amount of details
- Image dynamics (lines)
- Semantics

Focus via "spotlight"



Focus via contrast

Focus via contrast

- Tofoli

Foveal zone

- Eugene Delacroix Study for a portrait of Chopin

Focus on human

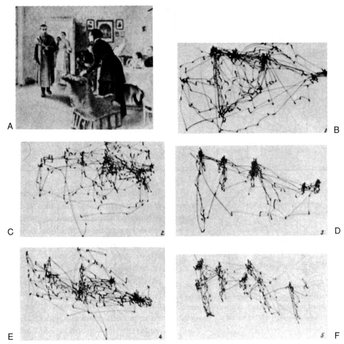
- Chardin 1735
- Compensate high-level with low-level
- [Baxandall *Patterns of Intention*]

Gaze and image cognition

- Similar to scientific method
  - Make hypothesis (mental model of the scene)
  - Perform experiments (gaze)

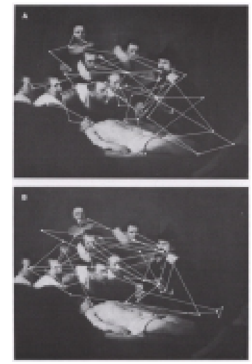
Depends on task

- painting by Repin
- B: free
- C: economic level
- D: ages
- E: what were they doing
- F: remember cloth



Depends on task?

- Rembrandt, *The Anatomy Lesson*
- Different tasks:
  - A: Aesthetic
  - B: Semantic
- Very similar paths



Diversive vs. specific

- Different strategies (Berlyne 1971)
- Diversive exploration
  - Hunt for new stimulation
  - Dispersed
  - Shorter fixation (<300ms)
- Specific exploration
  - Seeks specific information
  - Longer fixation (>400ms)

Effect of training

- Compare naïve beholders with specialists
  - Radiologists
  - Art students, art historians
- Specialists more specific
- Naïve more diversive

Fixation time & style

- Depends on style “complexity”
- Shorter fixation for more complex style
- E.g. fixation time is longer for classical style than for Baroque
- Because in more complex styles, there is so much stimulation that the eye seem to uses a diversive strategy

Number of focal point

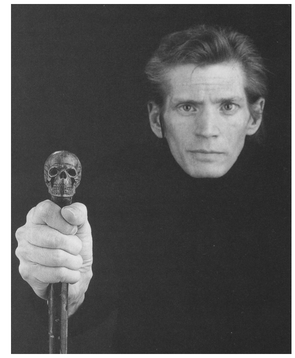
- The number of focal points is a crucial aspect of composition
- Dynamics of the image
- 1 region: imitates 1 foveation, striking
- Many regions: the gaze is transported, dynamism
- Path

Focus through contrast

- Rembrandt

Two focal zones

- Robert Mapplethorpe
  Self-portrait, 1988



Focus through perspective

- Raphael, The School of Athens

Focus: saliency + semantics

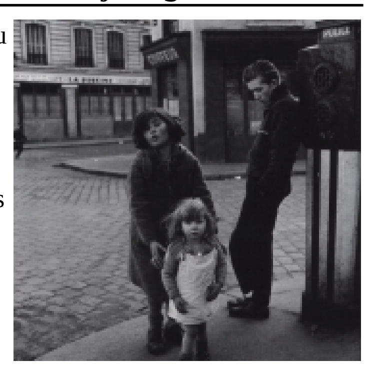
- Wyett *Christina's World*
- *3 focal points+direction of her gaze+her orientation*

Turner's Loire journey

- The gaze follows the journey
- [See part on motion depiction page 27]

Triple focus and subject gaze

- Robert Doisneau
  Les Gosses
  de la place
  Hebert
- The path of our gaze follows their gaze direction



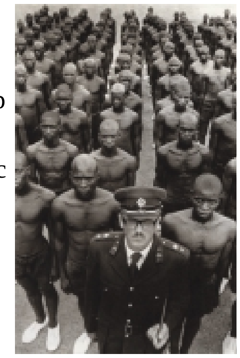
Figure/ground and comics

- The background is more detailed
- The low-level gaze attraction (details) conflicts/compensate for the high level (interest for the character)
- [See the part on Gestalt page 175]

From

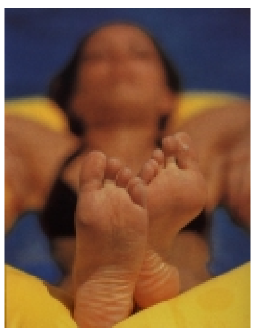
Focal point and dynamics

- Abbas, 1978
- Pop-out leads to uniform
- Perspective then leads to top of the image
- This induces a very dynamic composition



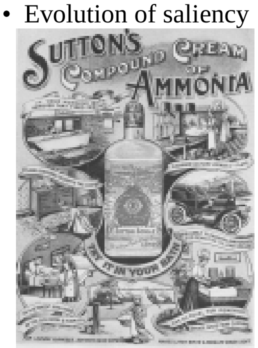
Focal point conflict

- Bottom-up (more detail on the foot) is different from top down (attraction to faces)
- Makes image dynamic



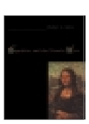
Advertisement and focal points

- Evolution of saliency



Further reading

Vision Science, from photons to phenomenology
Stephen E. Palmer, MIT Press, 1999
- Excellent reference on all aspects of vision



Cognition and the Visual Arts
Robert Solso, MIT Press, 1996
- Introduction to visual perception and relation with the visual arts

Perceptual and Artistic Principles for Effective Computer Depiction

Gestalt and Picture Organization

Fredo Durand
MIT- Lab for Computer Science

Grouping by color

Georgia O'Keeffe

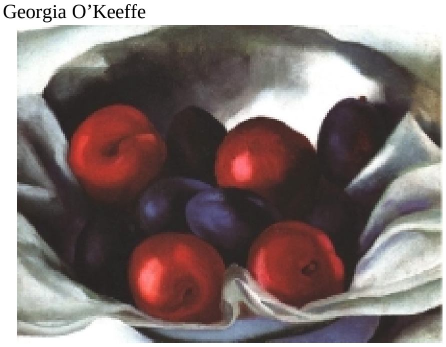
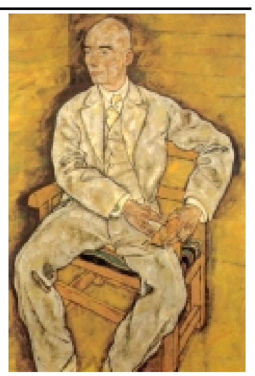


Figure-ground logos

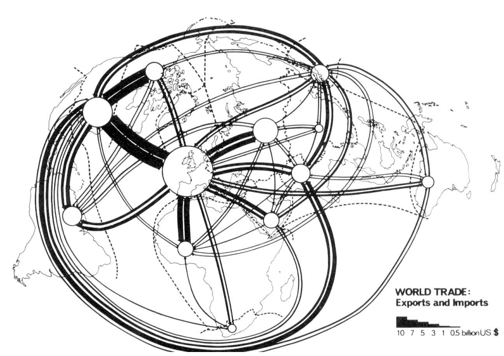
Figure-ground logos

Figure-ground enhancement

- Egon Schiele



Continuation and Map-Making



Grouping, illusory contour & fig/gnd

Absolut

Illusory contour

- From [Ramachandran 1995]

Illusory contour

Different philosophies of vision

- [Palmer 99]
- Why do things look as they do?

- Environment vs. organism
- Empiricism vs. nativism (vs. maturation)
- Atomism vs. holism
- Introspection vs. behavior

Different philosophies of vision

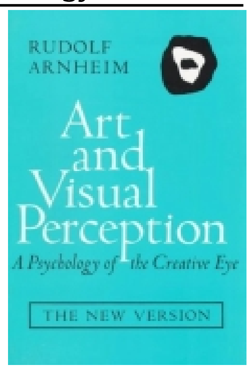
- [Palmer 99]
- Structuralism
  – (empiricism, atomism, organism, introspection)
- Holistic and Gestalt
  – (nativism, holism, organism, introspection)
- Ecological
  – (nativism, holism, environment, stimulus analysis)
- Constructivism
  – Mix of everything, unconscious inference

Context: Gestalt psychology

- [Palmer 99]
- Early 20th century
- Inspired by field theory in physics
- Holistic philosophy of vision
  – “Spontaneous” organization
  – Opposed to unconscious inference
- Has been integrated recently into modern framework

Context: Gestalt psychology

- Early 20th century
- Arnheim had a Gestalt psychology background
- Very popular in design
- Advertisement vs. art
- Interesting epistemology



Overview

- After low-level vision, we only know where edges are
- Need to organize the image
  - Segment by region, find structure

Prägnanz

- Cornerstone of Gestalt
- "Goodness"
- "Simplest" possible figure or organization
- Things are organized spontaneously and assumed to be in the simplest configuration
- Has recently been related to information theory (simple in terms of amount of information required to encode it)

Plan

- Grouping
- Figure-ground
- Completion and illusory contours

Grouping

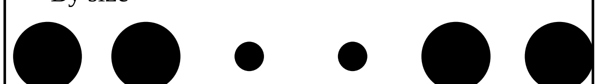
- "Similar" or "close" objects are perceived to belong to groups
- Spontaneous and powerful perceptual effect

Grouping

- By Proximity

- By Color

- By size

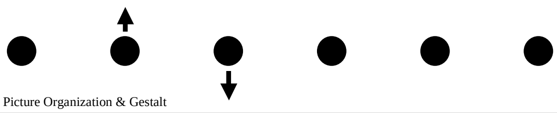


Grouping

- By Shape

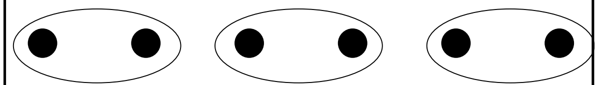
- By Orientation

- By Synchronicity (simultaneous motion)



Grouping

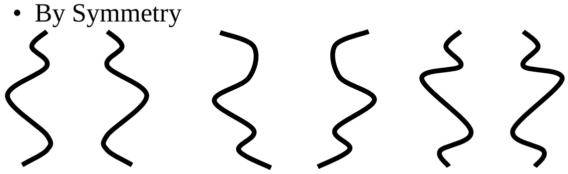
- By Region



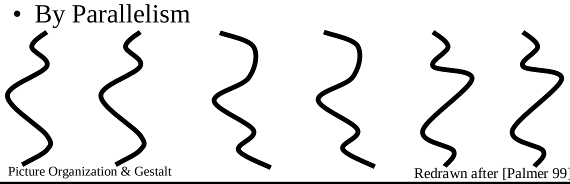
- By connectedness

Grouping

- By Symmetry



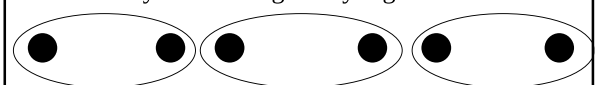
- By Parallelism



Redrawn after [Palmer 99]

Grouping conflict

- Proximity is overweighed by region



- Proximity is overweighed by connectedness

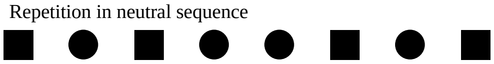
Grouping effect

- Task: Detect repetition of a shape in a sequence
- The repetition can be inside or across a group
- Slower when between groups (~0.7 vs. ~1.1s)

Repetition within group

Repetition across group

Repetition in neutral sequence



Redrawn after [Palmer 99]

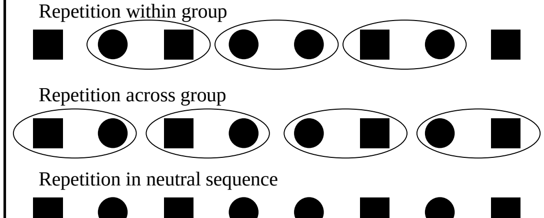
Grouping effect

- Task: Detect repetition of a shape in a sequence
- The repetition can be inside or across a group
- Slower when between groups (~0.7 vs. ~1.1s)

Repetition within group

Repetition across group

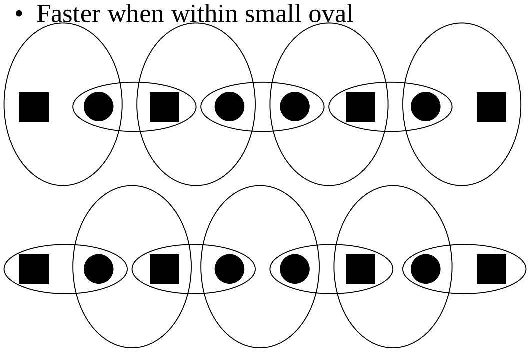
Repetition in neutral sequence



Redrawn after [Palmer 99]

Grouping conflict

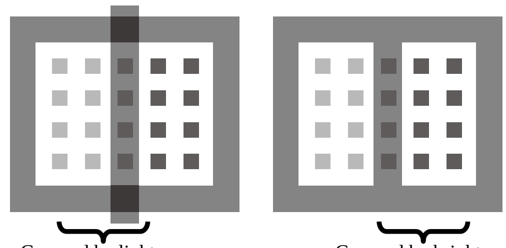
- Faster when within small oval



Redrawn after [Palmer 99]

Grouping after lightness constancy

- Does grouping occur before scene layout and illumination inference?



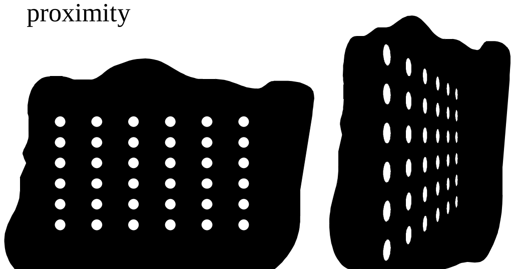
Grouped by lightness (intrinsic)

Grouped by brightness (extrinsic)

Redrawn after [Palmer 99]

Grouping after size constancy

- Grouped by 3D –proximity, not by retinal image proximity



Grouped by columns (proximity)

Still grouped by columns (3D proximity before perspective)

Redrawn after [Palmer 99]

Grouping in complex situations

- No quantitative rule yet!
- Very complex problem
- Too many parameters

Grouping and photo

Edward

Weston

Grouping and photo

Jean-Pierre Sudre

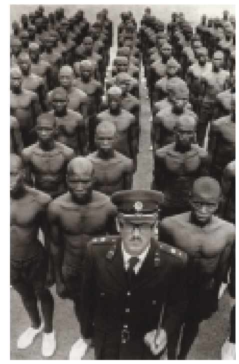
Grouping

- Grouping by proximity tells story

Picture Organization & Gestalt

Grouping

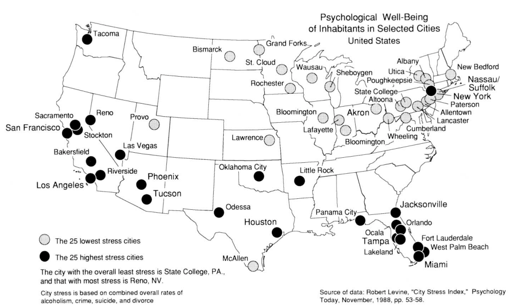
- Abbas
  South African Police in Training, 1978
- Grouping by proximity and similarity tells story



Picture Organization & Gestalt

Grouping & Map Making

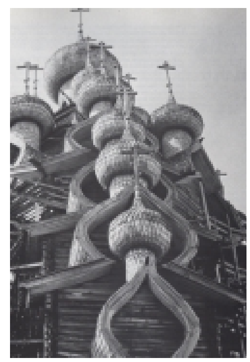
- Grouping provides efficient analysis



Picture Organization & Gestalt

Grouping & Architecture

- Grouping by similarity



Picture Organization & Gestalt

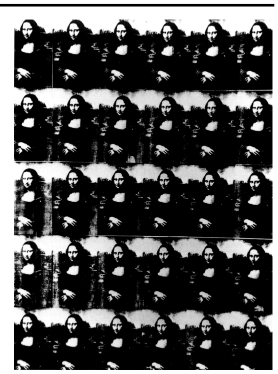
Grouping & Architecture

- Grouping and symmetry
- Cesar Pelli
  Petronas Towers
  Kuala Lumpur, Malaysia
  1991-97

Picture Organization & Gestalt

Grouping and repetition

- Andy Warhol
  30 Are Better than One
  1963



Picture Organization & Gestalt

Grouping and ornament

- Repetition, rhythm

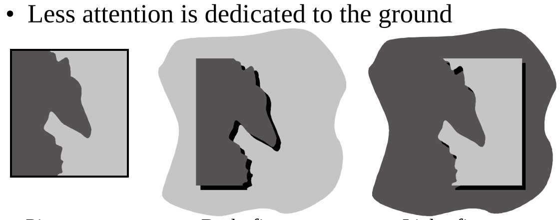
Closure & grouping

Plan

- Grouping
- Figure-ground
- Completion and illusory contours

Figure-ground

- What is in front (figure), and behind (ground)?
- There has to be one figure and one ground
- Related to occlusion and thus to depth
- Less attention is dedicated to the ground



Picture Dark=figure Light=figure

Redrawn after [Palmer 99]

Figure-ground

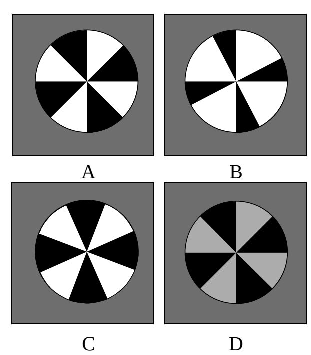
- The shape with the best “Prägnanz” is the figure
- Can be bimodal: we switch from one interpretation to the other
  - Visible on brain imagery
- But only one at a time

Picture

Redrawn after [Palmer 99]

Figure-ground

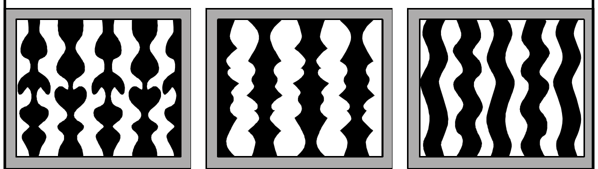
- A: ambiguous
- B: relative size
- C: symmetry & main axis
- D: contrast



Redrawn after [Palmer 99]

Figure-ground

- Effect of geometric properties on the “Prägnanz”

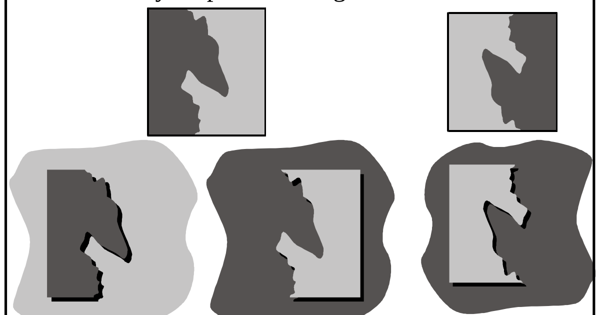


symmetry | Convexity vs. parallelism | parallelism

Redrawn after [Palmer 99]

Figure-ground & familiarity

- Familiarity helps: We recognize a horse



Redrawn after [Palmer 99]

Figure-ground painting

- Vassarely

Figure-ground pun

- Rubin vase

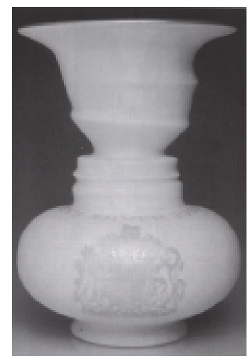


Figure-ground transition

- +grouping

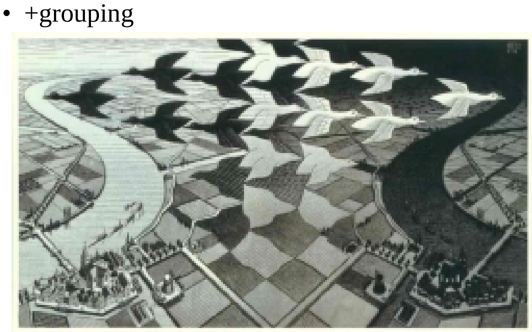
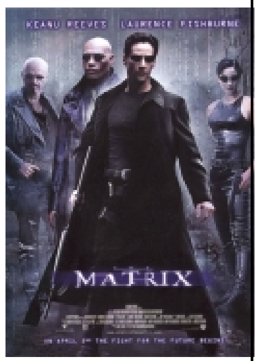


Figure Ground in design

- Sharon Gresh,
  Michael Mc Ginn

Figure-ground simplification

- For depth enhancement
- The contrast at the occlusion edge is enhanced
- The figure is easier to extract
- See notes on limitations p.27



Enhancing depth through contrast

Figure ground simplification

- Using rim-lighting (a.k.a. back-lighting)
- See notes on limitations p.27

Figure-ground and map-making

- To separate the land from the sea

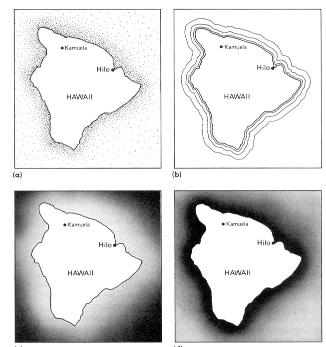


Figure ground – not so easy

- Monet

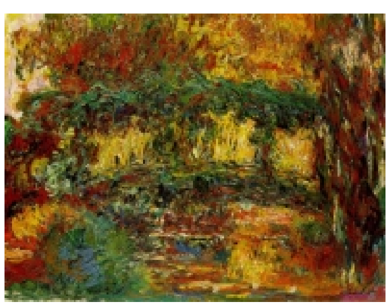
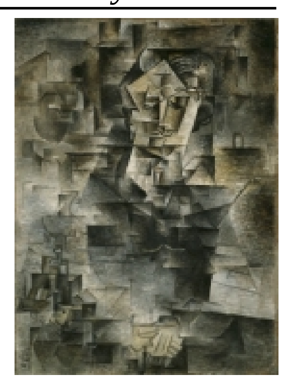


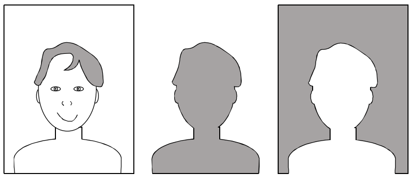
Figure ground – not so easy

- Picasso



Negative space

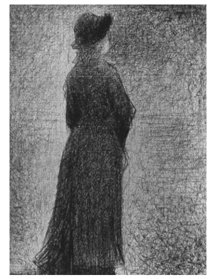
- The ground defines the negative space
- Usually overlooked
- Fundamental for balance
  - Also for typography

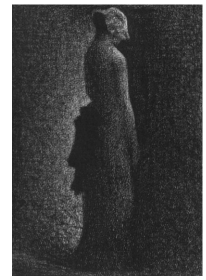


picture figure negative space

Closure & Negative space

- George Seurat
- Negative space are enclosed in the picture frame





Figure/ground and BD

From

Figure/ground and animation

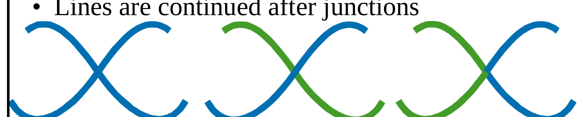
- The style is usually very different between figure and ground

Plan

- Grouping
- Figure-ground
- Completion and illusory contours

Continuation

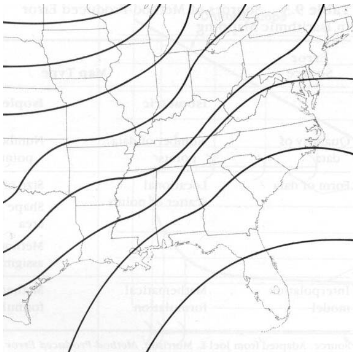
- Lines are continued after junctions



Picture good “Prägnanz” not so good “Prägnanz”

- And after gaps

Continuation and Map-Making



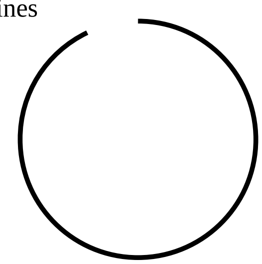
Continuation and design

- El Lissitzky, *Self Portrait: The Constructor* 1924

Continuation

Closure

- Closed shapes have better “Prägnanz”
- + continuation
- + illusory lines



Closure

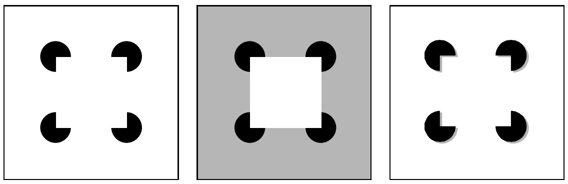
- Matisse

Closure

- La Linea, by Oswalso Cavandolini

Illusory contour

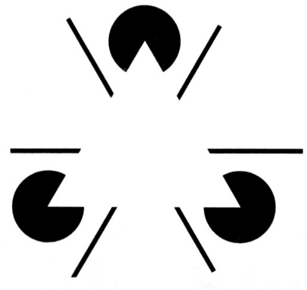
- An illusory contour is implied by continuation of the lines
- Related to figure ground



Picture Organization & Gestalt

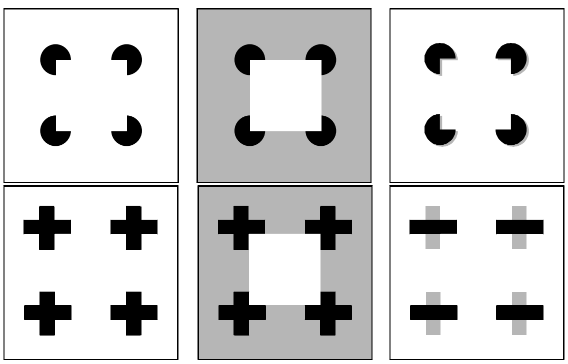
Illusory contours

- Kanisza



Picture Organization & Gestalt

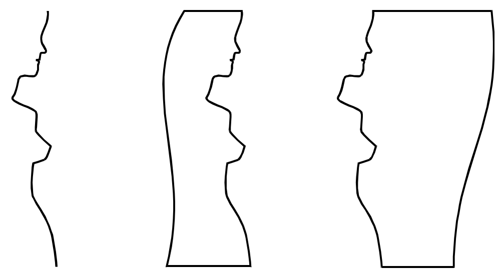
Illusory contour



Picture Organization & Gestalt

Closure & figure-ground

- A figure-ground effect can be transformed in a closure effect

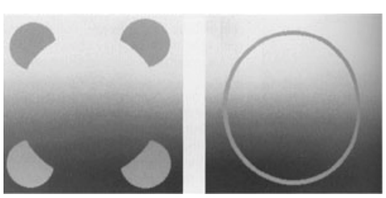


Picture Organization & Gestalt

Redrawn after [Palmer 99]

Illusory contour

- Can be more effective



Picture Organization & Gestalt

Illusory contour

Picture Organization & Gestalt

Illusory contour

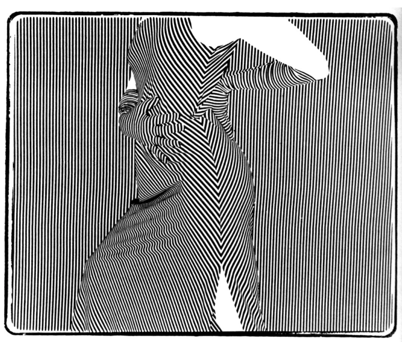
- Matisse

Picture Organization & Gestalt

Figure-ground and illusory contour

Picture Organization & Gestalt

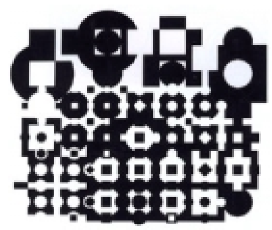
Figure-ground puzzle



Picture Organization & Gestalt

Illusory contours

Vasarely, *Helios*



Picture Organization & Gestalt

Illusory contour

- William Anders
  Earthshine
  1969
- Prägnanz:
  a circle is
  “simpler”

Picture Organization & Gestalt

Illusory contour

- Familiarity helps

Picture Organization & Gestalt

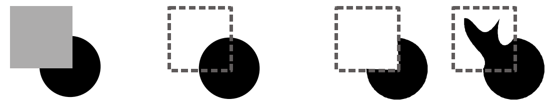
Illusory contour

- Familiarity helps

Picture Organization & Gestalt

Visual completion

- We complete the occluded part with the simplest shape (best “Prägnanz”)
- Related to continuation and closure

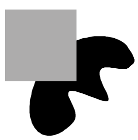


Picture · Simplest interpretations · Other possible interpretations

Picture Organization & Gestalt · Redrawn after [Palmer 99]

Visual completion

- With no context

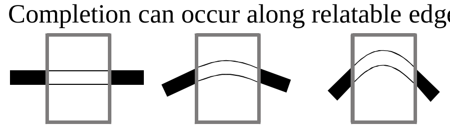


- With context

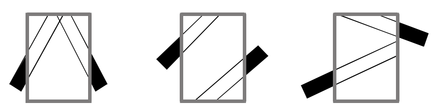
Picture Organization & Gestalt · Redrawn after [Palmer 99]

Relatable edges

- Completion can occur along relatable edges



Relatable Edges

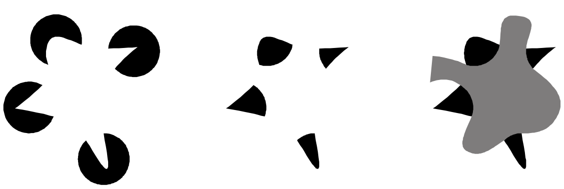


Unrelatable Edges

Picture Organization & Gestalt · Redrawn after [Palmer 99]

Illusory contour & completion

- Both “see” a figure from incomplete information
- An illusory edge effect can be turned into a completion edge

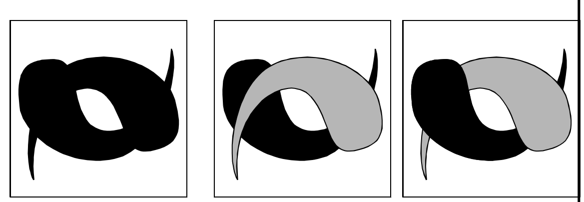


Illusory contour · Completion

Picture Organization & Gestalt · Redrawn after [Palmer 99]

Illusory contour & figure ground

- Amodal completion:
  2 interpretation are possible
  but no clear figure-ground



Picture Organization & Gestalt

Visual completion

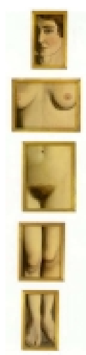
- Clarence Lee, 1977

Visual completion

- Greg Brown (mural)

Completion

- Magritte



Completion

- Degas
- Framing

Completion

- Marc Riboud
- Completion is challenged

Summary

- Prägnanz (goodness, simple in terms of information)

- Grouping
- Figure-ground
- Completion

- As usual pictures can
  - Simplify
  - Challenge

History of science

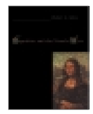
- Initially, strong opposition between Gestalt and other theories
- Lack of experimental data
- Has been applied beyond its scope
- Has been taken too literally

- Now, has been integrated with other theories
- Experiments
- Computational models

Further reading

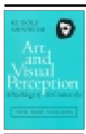
Vision Science, from photons to phenomenology
Stephen E. Palmer, MIT Press, 1999
- Excellent reference on all aspects of vision
- Introduces modern theories of Gestalt and visual organization

Visual Perception: Physiology, Psychology And Ecology
Vicki Bruce, Patrick R. Green, Mark Georgeson, M.A. Georgeson, Psychology Press, 1996
- Very good introduction to vision following three approaches, include Gestalt

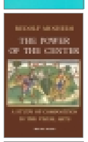


Cognition and the Visual Arts
Robert Solso, MIT Press, 1996
- Introduction to visual perception and relation with the visual arts

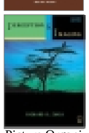
Further reading



Art and Visual Perception: A Psycholohy of the creative eye
Rudolph Arnheim, U. of California Press, 1954
- Seminal work on the relation between psychology and art history



The Power of the Center : A Study of Composition in the Visual Arts
Rudolph Arnheim, U. of California Press, 1988
- Art history, on composition and picture layout



Perception and Imaging
Richard D. Zakia, Focal Press, 1997
- More a design manual based on Gestalt principles

Ramachandran and Hirstein's Neurological Theories of Aesthetic for Computer Graphics

Bruce Gooch
University of Utah

One way of looking at the history of computer science is to examine the type of communication the computers of the day allowed. In the 70's and 80's human computer interaction was the basis for research. In the 90's the focus of research shifted to networking or computer to computer interaction. In the new millennium we are starting to look into human to human interaction via a computer. Humans communicate using three basic methods: via the spoken word, in writing, or using images. This text deals with communication between human minds via computer generated images. Because our goal is communication, we not only want a viewer to see a picture, we want them to understand an image. As the saying goes, "A picture is worth a thousand words, but an image is worth a thousand pictures."

Images are a logical choice for interpersonal communication because they utilize the highest bandwidth input to the brain, the eyes. While written language also uses the eyes for input, it adds further levels of abstraction which tend to slow and confuse the communication process. The purpose of this text is to educate the reader in some methods for enhancing the communication content of the images they create using computer graphics.

The article "Neurological Theories of Aesthetic" [28] by Vilayanur S. Ramachandran and William Hirstein lists what the authors call the eight laws of art. The authors argue that artists either consciously or unconsciously deploy these eight laws of art to optimally excite the visual areas of the brain. They also suggest that the first three rules are primary were as the last five serve to support the first three. These theories of Ramachandran and Hirstein explain many familiar experiences, such as why a cartoon squiggle can evoke a well-known face more quickly than a full color photograph. Their theories address three questions:

1. What are the rules of art that make something visually pleasing?
2. What form do these rules have and why did the rules evolve?
3. What brain mechanisms are involved?

Previous theories of art have looked at one or two of these questions, but never all three together. One problem with Ramachandran and Hirstein's analysis, which authors make clear, is that art is a diverse enterprise that may not be amenable to a simple treatment. In addition, these "laws" may not form a complete set of artistic principles. Ramachandran and Hirstein make no mention of the power of the center emphasized by Rudolph Arnheim [4], the widely recognized principle of balance in composition [7], or of the dynamic interplay of visual forces emphasized by Wassily Kandinsky.

Searching for a universal rule underlying the artistic experience is not a new quest [19]. During the 1920's and 1930's the mathematician G.D. Birkhoff attempted to reduce aesthetics to mathematics by defining the aesthetic beauty of an object to be the ratio of its symmetry to its complexity [8, 9]. Although Birkhoff's work is regarded as a failure, it can be said that his attempt advanced our knowledge of the difficulty of quantify beauty. In the field of computer graphics a number of works on visual communication and creating images based on the human visual system exist. Feiner and Seligmann [17, 31] borrowed principles from technical illustration. Kawai et al. [23] automated the creation of pleasing lighting. Both He et al. [21] and Karp and Feiner [22] examined how animation sequences are developed. Kowalski et al. [24] have explored user guided composition.

Ramachandran and Hirstein cite studies by other researchers in perceptual psychology as evidence for their eight laws. However, the true strength of Ramachandran and Hirstein's work is the experimental program that they propose for directly validating their observations [19]. They propose an experimental methodology, involving physiological measurements such as galvanic skin response, to explore the human experience of creating and viewing works of art. Such projects are currently being carried out. In one such study Robert L. Solso presented the results of a preliminary study of an accomplished artist as he drew a portrait while in an functional magnetic resonance imaging (fMRI) machine [32]. In addition countless studies involving visual stimulation, including works of art, and measured response have been reported in the medical and psychophysical literature [11, 16, 18, 13].

In this text Ramachandran and Hirstein's eight laws of art are refined to focus on images created using computer graphics. In addition this text expands the work of Ramachandran and Hirstein by including information from "Cognition and the Visual Arts", by Robert L. Solso [33], "Perception and Imaging" by Richard D. Zakia [42], and "Inner Vision" by Samir Zeki [43]. The eight laws are listed below, and each is expanded upon in a separate section of the text.

1. The Peak Shift Principle – Exaggerated elements are attractive.
2. Grouping and Binding – Perceptual grouping and binding makes objects stand out from the background.
3. Isolation of a Single Visual Module – Isolating a single visual cue helps to focus attention.
4. Problem Solving – Perceptual "problem solving" is reinforcing.
5. Contrast Extraction – Contrast is reinforcing.
6. Symmetry – Symmetry is attractive.
7. Generic Viewpoint – Unique vantage points are suspect.
8. Use of Metaphor – Visual puns and metaphors enhance art.

1 The Peak Shift Principle

Ramachandran and Hirstein define the peak shift effect as the use of supernormal stimuli to excite areas in the brain more strongly than natural stimuli. The peak shift effect is a well-known principle in animal learning [20]. For example, if a rat is taught to discriminate a square from a rectangle and rewarded for the rectangle, it will soon learn to respond more frequently to the rectangle. Moreover, if the rat is trained with a prototype rectangle of aspect ratio 3:2, it will respond even more positively to a longer thinner rectangle of aspect ratio 4:1. This result implies that what the rat is learning to value is not a particular rectangle but a rule: rectangles are better than squares. So the greater the ratio between the long and the short sides, i.e. the less square-like it is, the better the rectangle is in the rats eyes. This is the peak shift effect. Ramachandran argues that this principle holds the key for understanding the evocativeness of much of visual art.

Caricatures of human faces are a well studied example of the peak shift effect in human visual perception. Caricatures constitute a powerful medium to express and exaggerate distinctive features of human faces. Caricatures are usually created by skilled artists who use lines to represent facial features. The skill of the artist lies in knowing which facial features are essential and which are incidental. For facial caricatures the usual assumption is that the feature shift for a particular face should be to enhance its difference from an average face [12, 29].

It could be argued that line drawings as well as caricatures derived from such drawings form impoverished environments when compared with their photographic counterparts. Within such impoverished environments, many studies have shown that caricatures can be recognized faster than line drawings that accurately portray someone's face (these are called veridical line drawings) [6, 12, 29, 34]. Similarly, line-drawn caricatures tend to be learned faster in learning tasks than veridical line drawings [34]. This is known as the super portrait effect. Examples of faces shown as photographs, line art, and caricatures are shown in Figure 1.

Ramachandran and Hirstein explain that the peak shift effect can occur in any visual modality. The human responses to color, motion, form, highlight, outline, and depth are all susceptible to peak shift effects. An example of a peak shift in color space is shown in the paintings of Van Gough as seen in Figure 2. Rodin's "Burghers of Calais" is an example of peak shift in form. Rodin exaggerates the details of each of the figures in order to create emotional impact.

The key fact concerning the peak shift phenomena is that the reward stimulus and the non-reward stimulus must be close in order for the peak shift stimulus to exist. For example, if a rat is trained to respond to a tone at 1000Hz and not to respond to a tone at 500Hz no peak shift will be observed. However, if the rat is trained to respond to a tone at 1000Hz and not to respond to a tone at 950Hz a peak shift stimulus will be observed at approximately 1010Hz. Some researchers believe that due to this fact, peak shift may not be responsible for an individuals response to art, but may explain quite a bit about art history and the fashion industry. For example in fashion the shorting of skirts and widening of mens ties during the 60's may be an example of peak shift phenomena over time.

2 Perceptual Grouping and Binding

The second principle suggested by Ramachandran and Hirstein is grouping and binding. When we look at a collection of discrete entities we often perceive the collection as organized into subsets. When our mind recognizes differing subsets as a unit, that unit becomes bound in our mind and is perceived as different from the surrounding collection from then on. Ramachandran and Hirstein explain that visual areas of the brain may have evolved specifically to extract correlations in different visual modalities and that this process is facilitated and reinforced by direct connections from these brain areas to limbic structures (pleasure centers).

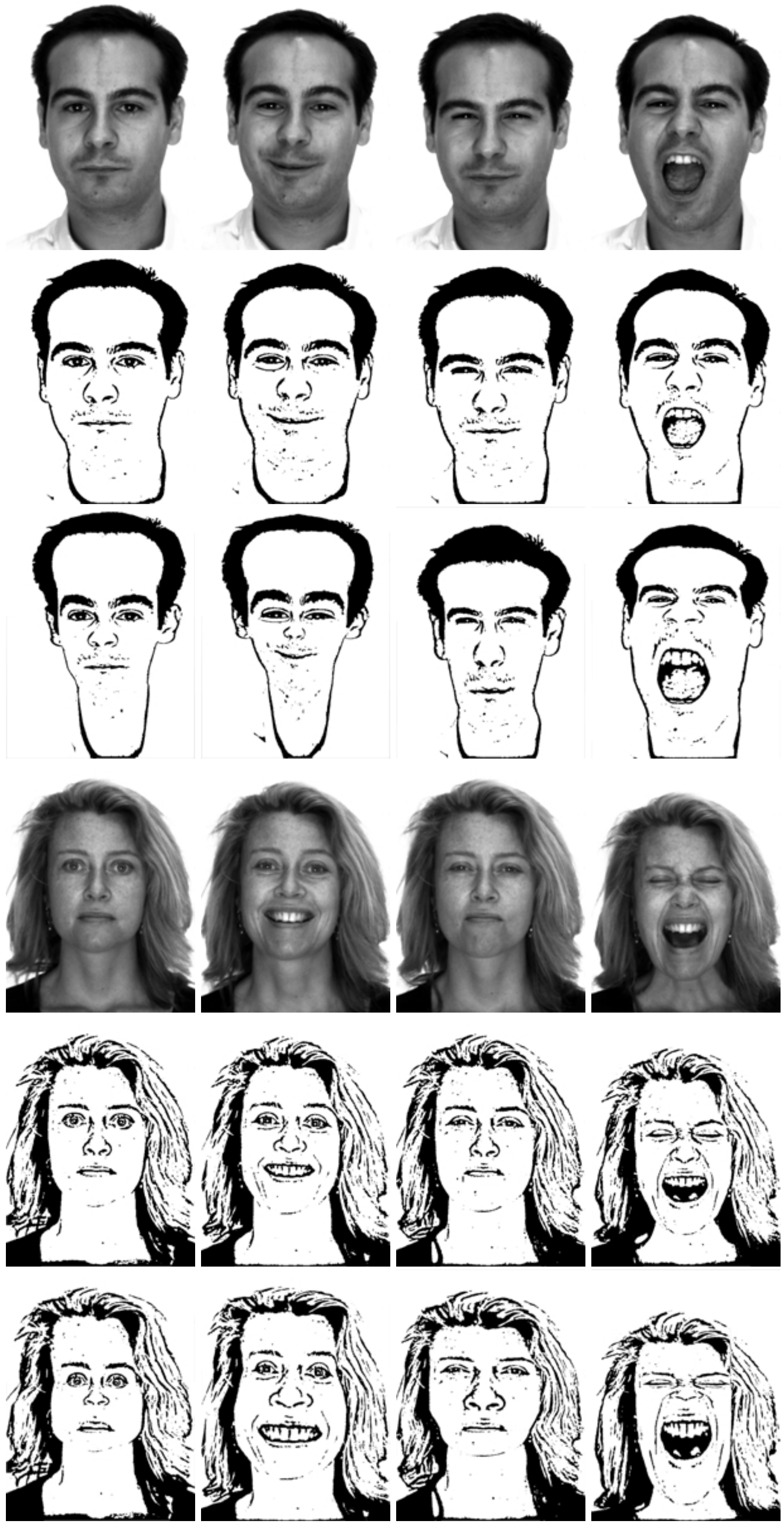


Figure 1: *Examples of photographs, black-and-white line art and black-and-white line art caricatures of human faces. The line art is an example of a single mode, “edge lines”, of human faces. The caricatures are an example of a peak shift in the “edge lines” mode. The images with differing facial expressions are courtesy of Aleix Martinez of Purdue University. The entire face database is online [25].*

Figure 2: *Left and Center: Vangogh's "Irises, Saint-Remy" and "The Cafe Terrace on the Place du Forum, Arles, at Night" are examples of peak shift in color space. Color images of these works are available online [1]. Right: Rodin's "Burghers of Calais" is an example of peak shift in form (Image courtesy of The Marion Koogler McNay Art Museum [36]).*

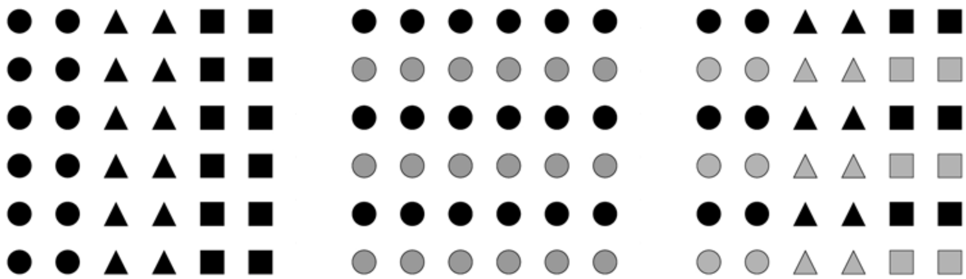


Figure 3: *Left: An example of grouping by similarity of shape. Center: An example of grouping by similarity of color. Right: An example of grouping by similarity overwhelming the perception of grouping by shape.*

2.1 Perceptual Grouping

Perceptual grouping was first considered by the gestalt psychologist Max Wertheimer [40] who investigated what our mind does versus what our mind might have done. Wertheimer explored the way elements in a visual scene are typically perceptually grouped into units. He constructed simple examples consisting of sets of dots. The purpose of these examples is to illustrate factors that influence the grouping of elements into units. Wertheimer suggested grouping factors that influence how elements are organized. The fact that perceptual grouping tendencies are genetic, not learned, is suggested by the cross-cultural effectiveness of sleight-of-hand magic and camouflage both of which work by subverting Wertheimer's grouping factors. Visual examples of Wertheimer's grouping factors are shown in Figures [3-8] and listed below:

1. Similarity – Items that are the same are grouped together.
2. Proximity – Items that are physically close are grouped.
3. Common Fate – Items that move together are grouped.
4. Continuity – Items that form or are joined by a line are grouped.
5. Closure – Items that form closed regions of space are grouped.
6. Past Experience – Items are interpreted based on surrounding items.

Interaction between perceptual grouping factors is far from simple for three reasons: First, the appearance of parts is determined by wholes. Second, judgments about similarity and proximity are always comparative. Third, one grouping factor can override another. For example grouping by color can overcome grouping by shape as shown in Figure 3.

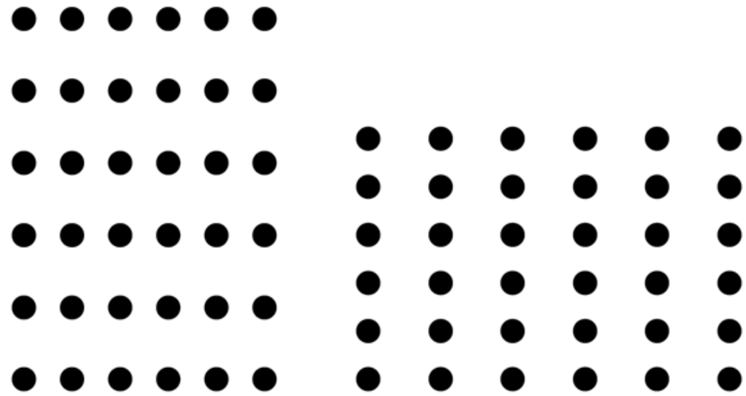


Figure 4: *Examples of grouping by proximity. The matrix of dots on the left is perceived as being composed of rows while the matrix of dots on the right is perceived as being composed of columns.*

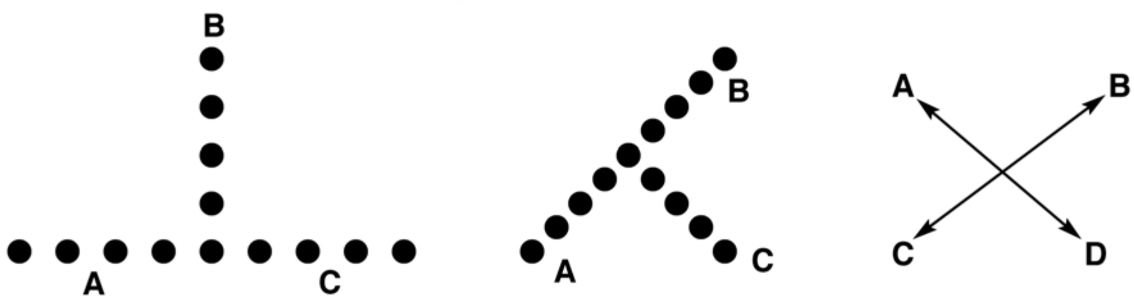


Figure 5: *Left and Center: Examples of grouping by continuity. Most observers perceive these figures as composed of two line segments instead of three separate segments. Right: An example of another aspect of grouping by continuity. Observers group this scene into two sets A, D and C, B because of the lines joining the letters.*

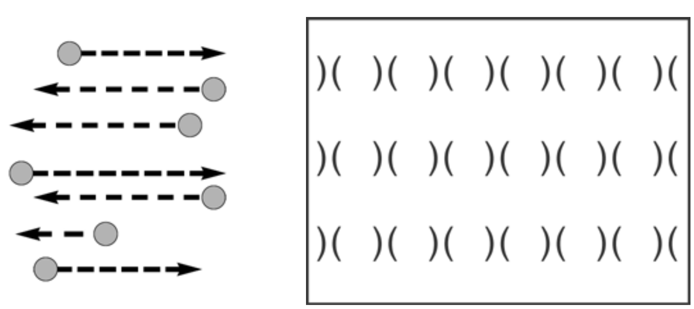


Figure 6: *Left: An example of grouping by common fate. Items moving on similar paths are grouped into units. Right: These parentheses could be grouped by proximity to produce hourglass shaped figures, however most observers find that grouping the parentheses as closed regions is more pleasing.*

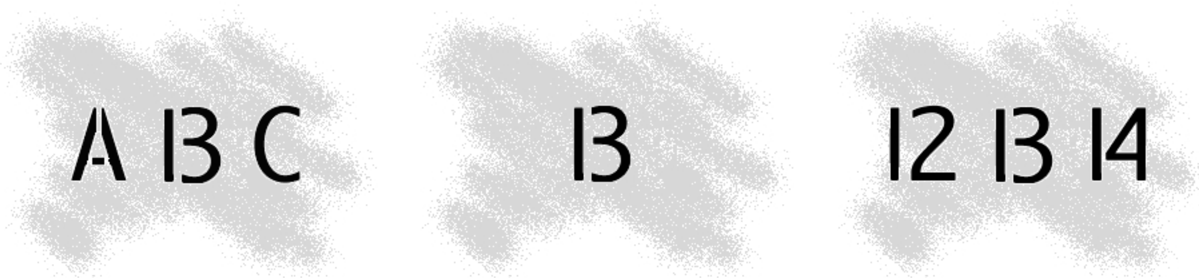


Figure 7: *An example of grouping by experience. On the left the center figure is observed to be the letter B. On the right the center figure is viewed as the number 13. In the center the figure is shown on its own. Images courtesy of Professor Charles Schmidt, Rutgers University [30].*

Figure 8: *Left: This image is initially observed to be a random field of dots. However, once the dog is perceived the "dog spots" are grouped together which is experienced as a pleasing effect. Ramachandran and Hirstein believe that this pleasing effect may be due to stimulation of the limbic system by the temporal lobe cortex of the brain. Center: The "old-young woman" image can only be bound in a single phase (old or young) by most viewers. Right: The object of the third example is initially ambiguous, by turning the page the grouping of figure and ground becomes apparent. (After Solso [33])*

Artists have been explicitly using these rules in their work for some time. For example Paul Klee used gestalt grouping diagrams in his paintings in the 1930s [35]. Later generations of artists learned of Wertheimer's laws of visual organization from two books on art and design education: "Language of Vision" by Gyorgy Kepes, a graphic designer who taught at the New Bauhaus in Chicago; and "Art and Visual Perception: A Psychology of the Creative Eye" by Rudolf Arnheim at Harvard University. Anthony A. Apodaca and Larry Gritz include a chapter on how gestalt grouping and binding can be facilitated using computer graphics in their book "Advanced Renderman Creating CGI for Motion Pictures" [2].

2.2 Perceptual Binding

Perceptual binding is illustrated in the examples shown in Figure 8. In the first example, the "dog image" is initially observed as a field of random spots. However once the dog grouping is interpreted by your visual system the subset of "dog spots" is linked in your mind and it becomes nearly impossible to perceive the image as a random field of spots again. The discovery of the dog and the linking of the dog spot group generates a pleasant sensation. Artists understand the pleasure given by such effects and are masters of producing the "aha" sensation in a viewer. In the second example, the "old-young woman image" the lines can either be perceptually grouped and interpreted as the face of a young woman or an old woman. However, the visual system will not allow both interpretations to be held at the same time. In the third example the image is initially ambiguous. When the page is rotated the grouping of figure and ground becomes apparent. Binding most likely serves to aid in the detection of predators and prey and is therefore an evolutionary advantage.

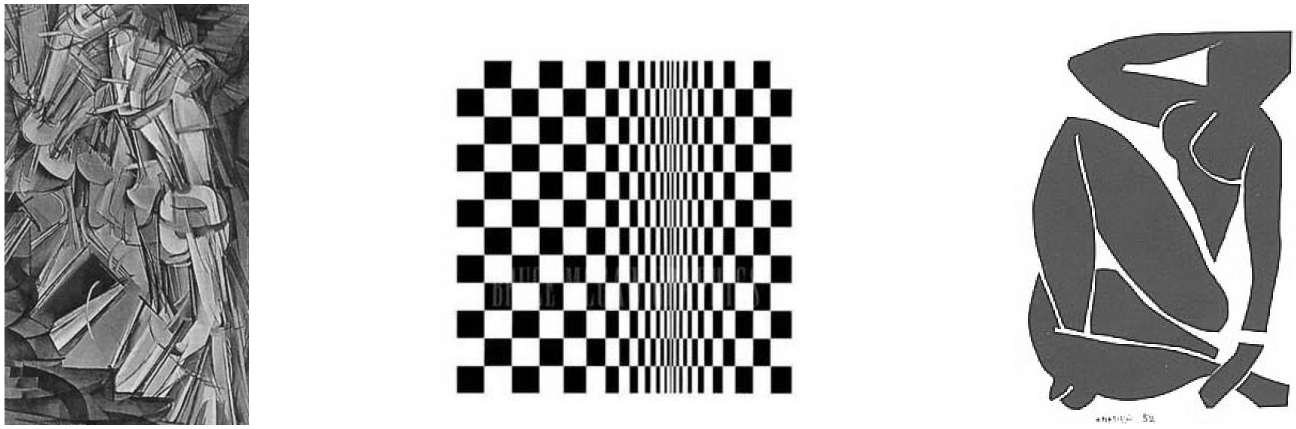


Figure 9: *Left: Marcel Duchamp's "Nude Descending a Staircase, No. 2" is a painted example of kinetic art. Center: Bridget Riley's "Movement in Squares" is an example of an artist creating the illusion of motion were none exits. Right: The "Blue Nude 3" of Henri Matisse is an example of fauvism. Color images of these works are available online [1].*

3 Isolation of a Single Module

The third principle of Ramachandran and Hirstein is the need to isolate a visual modality before applying the peak shift stimulus. They believe that by providing only a single visual module the attention of the observer is more easily focused onto the peak shifted stimulus. The ability of the visual to isolate a visual modality can explain the effectiveness of outline drawings or sketches.

The neurophysiologist Semir Zeki [43] has provided evidence that the brain does indeed process visual information into separate modalities in his quest for a "theory of aesthetics based on an understanding of the workings of the brain." Zeki has shown that movement, color and form are processed using different methods by different areas of the brain. In relating his work to the understanding of works of art, Zeki states,"artists are neurologists, studying the brain with techniques that are unique to them and reaching interesting but unspecified conclusions about the organization of the brain." Zeki's most compelling argument is: Artists who are especially interested in property X have found ingenious ways to partially isolate property X from property Y, using methods which have a clear basis in known neuroanatomy.

In order to understand this idea consider kinetic art and fauvism. Kinetic art refers to painted or sculptured works that include motion as a significant dimension. Fauvism is a style of painting that flourished in France at the beginning of the 20th century. The Fauves used pure brilliant colors and applied their paint straight form the tubes in an aggressive, direct manner. The Fauves painted directly from nature, but their works contain a strong expressive reaction to the subjects they painted. In the case of kinetic art, Zeki's property X is movement and property Y is color, while in the case of fauvism, property X is color and property Y is form. Examples of kinetic art and fauvism are shown in Figure 9.

An important fact about vision is the massive feedback from higher to lower centers, including the retina. This suggests that vision, far from being a passive reception of "what's out there", is an active search for "what's important". With the search is based on the viewers expectations and prior experience. Work by Zeki has shown that there is significant feedback among the areas of the brain associated with visual processing. The most basic insight to be gained from his work is that a great deal of parallel distributed processing is needed in order to create perceptual constancy and that most visual processing is unconscious. It seems likely that we will see further insights into the neural bases of visual phenomena as the result of continuing experiments in brain imaging. A complete description of neural workings of the vision centers of the brain is beyond the scope of this text. Interested readers are directed to Zeki's book "Inner Vision".

4 Problem Solving

Perceptual problem solving refers to the pleasure the brain takes in deciphering ambiguous scenes. Ramachandran and Hirstein argue that under the right conditions ambiguity itself can be a source of pleasure. For example the Mona Lisa's smile.

Perceptual problem solving also explains how the symbolic representation of an images subject may be given added significance. Perceptual problem solving is a constructive process based on the interplay between features of which the pattern is composed (bottom-up processes) and knowledge-based perceptual hypotheses (top-down processes). At a low level, patterns and forms are visually bound. Then at a higher level of visual analysis, recognized patterns and forms may summon a chain of

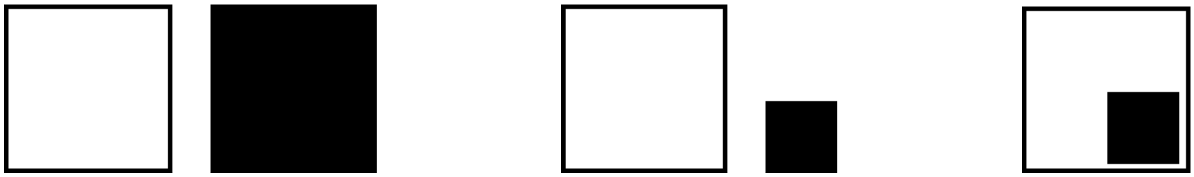


Figure 10: *Left: An example of intensity contrast between the two squares. Center: An example of size contrast between the two squares. Right: An example of position contrast between the two squares (inside versus outside).*

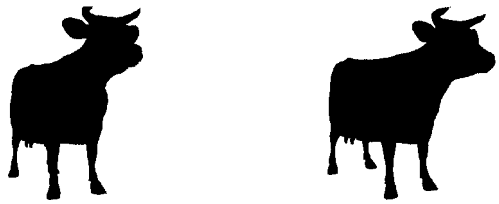


Figure 11: *Left: an "accidental" view where one of the cows hind legs ends up directly behind a front leg. Right: the same cow from a slightly perturbed viewing direction.*

associations. The mental effort that a viewer puts forth in extracting meaning from an image increases the emotional response that the viewer will have toward that image. Abstraction in the image requires the viewer to make a perceptual effort to extract the theme of the image. This effort on the part of the viewer forms an essential component of the viewing experience. This effect can be taken advantage of by an artist to produce more compelling images through the use of abstraction and symbolism.

5 Contrast Extraction

Contrast occurs between dissimilar features that are physically close together. Nearly any object in an image can be contrasted to any other object in the scene based on one of its aspects: color, size, shape, font, texture, etc. Ramachandran and Hirstein suggest that the visual system allocates attention to contrasting regions due to the fact that information generally resides in regions of change. This makes regions of an image which contain higher contrast more interesting and therefore more pleasing. Three simple examples of visual contrast are shown in Figure 10.

6 Symmetry

Symmetry establishes a ridiculous and wonderful cousinship between objects, phenomena and theories outwardly unrelated: terrestrial magnetism, woman's veils, polarized light, natural selection, the theory of groups, structure of space, vase designs, quantum physics, scarabs, flower petals, X-ray interference patterns, cell division in sea urchins, equilibrium positions of crystals, Romanesque cathedrals, snowflakes, music, the theory of relativity...

–Herman Weil

That symmetry is an important visual cue can be seen in the recurrence of symmetric patterns and designs throughout human history. Virtually all elements of the constructed environment from architecture and art to furniture and transportation contain at least one axis of symmetry. Symmetry is a special case of the gestalt grouping principle of similarity and it can been argued that symmetry is a useful cue for discriminating living organisms from inanimate objects. It has been shown that both humans and animals prefer bilateral symmetry when choosing a mate [37].

7 Generic Viewpoint

Psychologists have studied viewers' preferences for one viewpoint over another for particular objects. A viewpoint that is preferred by most viewers is called a *canonical viewpoint*. Palmer et al. [27] found that canonical viewpoints are off-axis, while Verfaillie [39] discovered that a three-quarter view of a familiar object is preferred.

A thorough investigation of canonical views was recently carried out by Blantz et al. [10]. They found three predictors of whether a view is canonical: the significance of visible features for a given observer, the stability of the view with respect to small transformations, and the extent to which features are occluded.

Significant features for an observer may include the facial portion of a head, the handle of a tool, or the seat of a chair. In viewing objects, Blantz et al. found that people preferred views which expressed the manner in which an object was seen in its environment, i.e. chairs are viewed from above, while airplanes may be viewed from above or below. They also found a distinct lack of "handedness" when humans choose preferred views. For example, when viewing a teapot a right handed viewer did not mind if the handle was placed on the left side of the image.

Image stability means that the viewpoint can be moved with little or no change in the resulting image. Many psychology researchers have shown that objects in a scene which share an edge will confuse a viewer [5, 7, 28]. For example the viewpoint that produces the "three legged cow" in Figure 11 is never picked as a canonical view.

When subjects in the Blantz et al. study were given the ability to choose the viewpoint for an object, it was discovered that the subjects performed an internal optimization to find a viewpoint that showed the smallest number of occlusions. This occurred for both familiar objects and artificial geometric constructs. For instance, when choosing a viewpoint for a teapot the subjects always choose a viewpoint that shows both the handle and the spout. This result agrees with Edelman et al. [15] who showed that canonical views for "nonsense" objects may also exist.

Artists have their own heuristics for choosing view directions that are consistent with the psychology results: pick an off-axis view from a natural eye height. Direct 45° angles are avoided. Another rule is to have the projections of front/side/top of the object to have relative areas of 4/2/1 on the canvas [3, 33] (often expressed as 55%/30%/15%). The front and side dimensions can be exchanged depending on the object.

8 Use of Metaphor

A metaphor expresses one thing in terms of another to suggest a likeness or analogy between them. An example of an illustrative metaphor is "the atom is like a solar system." The atom has a nucleus just as the sun is the solar system's nucleus. The atom has electrons whirling around that nucleus just as the sun has planets circling around it [14]. This metaphor draws a visual analogy between something we have a metal image of (the solar system) and something we may not (the structure of an atom).

Visual metaphors surround us and provide the most prevalent mode of sharing knowledge. Nearly every television and magazine advertisement is composed of a modern visual metaphor. A favorite visual metaphor used to signify "speed" is time-lapse photography of traffic at night. Anti-depressant drug companies run adds with images of "the sun coming out". Cultural metaphors form a common visual lexicon which can be used to emphasize the subject of an image or enhance the emotional response to an image.

9 Conclusion

Ramachandran and Hirstein's article, "Neurological Theories of Aesthetic" provides a useful framework for enhancing the communication content of computer generated images. Ramachandran and Hirstein present a series of rules which use evolutionary developed mechanisms of the human visual system for the perception of images. These rules allow the creator of an image to guide the attention of a viewer into a more in-depth reaction to the subject of the image.

Ramachandran and Hirstein's work has opened the door to a new frontier research into how the human visual system processes the information contained in works of art. The knowledge gained in this research will give credence to, or debunk the artistic rules of thumb currently used to judge the communication content of images. This work may also allow scientists to answer fundamental philosophical questions about the nature of art. Questions such as; "What is the difference between viewing a landscape and viewing a painting of a landscape?", can at least be answered in terms of how the brain responds to these very different stimuli.

Ramachandran and Hirstein have also drawn a fire-storm of commentary from critics in both the scientific and artistic communities. It seems that everyone either loves "Neurological Theories of Aesthetic" or hates it, but no one is indifferent. Examples of some comments on their work are:

"Perception may seem to some to be a phenomenological experience inaccessible to scientific rigor, but the efforts of generations of perceptual psychologists have shown that many aspects of perception are governed by a body of lawful relationships no less tractable than those of quantum physics, for example. The extension of such relationships to the subtleties of aesthetics is another kettle of slippery fish taken up by Ramachandran and Hirstein in a thought-provoking article in the Journal of Consciousness Studies."

–Christopher W. Tyler, "Is Art Lawful?" [38].

"I will demonstrate that Ramachandran and Hirstein confuse arousal (in a certain technical sense) with beauty, with the disastrous result of excluding most of what is usually taken to distinguish high art from its lower forms, such as advertising, industrial design, and pornography." *–Donnya Wheelwell, "Against the Reduction of Art to Galvanic Skin Response" [41],*

Donnya Wheelwell is the nom de guerre of a science professional who wishes to remain anonymous to avoid the scorn of her colleagues.

"Unfortunately, the flaw which undermines Ramachandran and Hirstein's attempts is a confusion regarding what constitutes an experience of beauty. They conflate pleasurable responses of a sexually titillating nature and other agreeably sensuous pleasures with the pleasurable response evoked by beauty."

–Jennifer Anne McMahon "Perceptual Principles as the Basis for Genuine Judgments of Beauty" [26].

References

[1] ALLPOSTERS.COM. http://allposters.com.

[2] APODACA, A. A., AND GRITZ, L. *Advanced Renderman Creating CGI for Motion Pictures*. Morgan Kaufmann, 2000.

[3] ARNHEIM, R. *Art and Visual Perception: A Psychology of the Creative Eye*. University of California Press, 1974.

[4] ARNHEIM, R. *The Power of the Center*. University of California Press, 1988.

[5] BARBOUR, C. G., AND MEYER, G. W. Visual cues and pictorial limitations in photorealistic images. *The Visual Computer 9*, 4 (1992), 151–165.

[6] BENSON, P. J., AND PERRET, D. I. Perception and recognition of photographic quality facial caricatures: implications for the recognition of natural images. *European Journal of Cognitive Psychology 3*, 1 (1991), 105–135.

[7] BETHERS, R. *Composition in Pictures*. Pitman Publishing Corporation, 1964.

[8] BIRKHOFF, G. Atti congressi bologna. 315–33.

[9] BIRKHOFF, G. Aesthetic measure. *Cambridge University Press* (1933).

[10] BLANZ, V., TARR, M. J., AND BULTHOFF, H. H. What object attributes determine canonical views. *Perception 28*, 5 (1999), 575–600.

[11] BOYNTON GM, DEMB JB, G. G. H. D. Attentional modulation of neural processing of shape, color, and velocity in humans. *Science 248* (1990).

[12] BRENNAN, S. E. Caricature generator: The dynamic exaggeration of faces by computer. *Leonardo 18*, 3 (1985), 170–178.

[13] CHRIS MIALL, J. T. The painter's eye. *http://www.physiol.ox.ac.uk/%7Ercm/pem/*.

[14] CLAIR, R. N. S. Visual metaphor, cultural knowledge, and the new rhetoric. *Learn in Beauty: Indigenous Education for a New Century* (2000), 85–101.

[15] EDELMAN, S., AND BULTHOFF, H. Orientation dependence in the recognition of familiar and novel views of three-dimensional objects. *Vision Research 32*, 12 (1992), 2385–2400.

[16] ENGEL S, ZHANG X, W. B. Colour tuning in human visual cortex measured with functional magnetic resonance imaging. *Nature 388* (1997).

[17] FEINER, S. Apex: an experiment in the automated creation of pictorial explanations. *IEEE Computer Graphics & Applications 5*, 11 (November 1985), 29–37.

[18] GAUTHIER I, ANDERSON AW, T. M. S. P. G. J. Activation of the middle fusiform face area increases with expertise in recognizing novel objects. *Nature Neuroscience* (1999).

[19] GOGUEN, J. A. Art and the brain editorial introduction. *Journal of Consciousness Studies 6*, 6-7 (1999), 15–51.

[20] HANSEN, M. H. Effects of discrimination training on stimulus generalization. 3221–334.

[21] HE, L., COHEN, M. F., AND SALESIN, D. H. The virtual cinematographer: A paradigm for automatic real-time camera control and directing. In *Proceedings of SIGGRAPH* (1996), pp. 217–224.

[22] KARP, P., AND FEINER, S. Issues in the automated generation of animated presentations. In *Graphics Interface* (1990), pp. 39–48.

[23] KAWAI, J. K., PAINTER, J. S., AND COHEN, M. F. Radioptimization - goal based rendering. In *Proceedings of SIGGRAPH* (1993), pp. 147–154.

[24] KOWALSKI, M. A., HUGHES, J. F., RUBIN, C. B., AND OHYA, J. User-guided composition effects for art-based rendering. *2001 ACM Symposium on Interactive 3D Graphics* (March 2001), 99–102. ISBN 1-58113-292-1.

[25] MARTINEZ, A. http://rvl1.ecn.purdue.edu/ aleix/aleix_face_db.html.

[26] MCMAHON, J. A. Perceptual principles as the basis for genuine judgments of beauty. *Journal of Consciousness Studies 7*, 8-9 (2000).

[27] PALMER, S., ROSCH, E., AND CHASE, P. Canonical perspective and the perception of objects. *Attention and Performance 9* (1981), 135–151.

[28] RAMACHANDRAN, V., AND HIRSTEIN, W. The science of art a neurological theory of esthetic experience. *Journal of Consciousness Studies 6*, 6-7 (1999), 15–51.

[29] RHODES, G., BRENNAN, S., AND CAREY, S. Identification and ratings of caricatures: implications for mental representations of faces. *Cognitive Psychology 19* (1987), 473–497.

[30] SCHMIDT, C. http://www.rci.rutgers.edu/ cfs/305_html/gestalt/wertheimer2.html.

[31] SELIGMANN, D. D., AND FEINER, S. Automated generation of intent-based 3d illustrations. In *Proceedings of SIGGRAPH* (1991), pp. 123–132.

[32] SOLSO, R. The cognitive neuroscience of art: A preliminary fmri observation. *Journal of Consciousness Studies 7*, 8-9 (2000).

[33] SOLSO, R. L. *Cognition and the Visual Arts*. MIT Press/Bradford Books Series in Cognitive Psychology, 1999.

[34] STEVENAGE, S. V. Can caricatures really produce distinctiveness effects? *British Journal of Psychology 86* (1995), 127–146.

[35] TEUBER, M. Blue night by paul klee. *Vision and Artifact* (1976), 131–151.

[36] THE_MARION_KOOGLER_MCNAY_ART_MUSEUM. http://www.mcnayart.org/.

[37] THORNHILL, R. Darwinian aesthetics. *handbook of Evolutionary Psychology: Ideas, Issues and Applications* (1998).

[38] TYLER, C. W. Is art lawful? *Perspectives: Neuroscience. Science. 285* (1999).

[39] VERFAILLIE, K., AND BOUTSEN, L. A corpus of 714 full-color images of depth-rotated objects. *Perception and Psychophysics 57*, 7 (1995), 925–961.

[40] WERTHEIMER, M. Untersuchen zur lehre von der gestalt. *Psychologishe Forshung 4* (1923), 301–305.

[41] WHEELWELL, D. Against the reduction of art to galvanic skin response. *Journal of Consciousness Studies 7*, 8-9 (2000).

[42] ZAKIA, R. D. *Perception and Imaging*. Focal Press Publications, 1997.

[43] ZEKI, S. *Inner Vision*. Oxford University Press, 1999.

The Art and Science of Depiction

Computational Vision and Picture

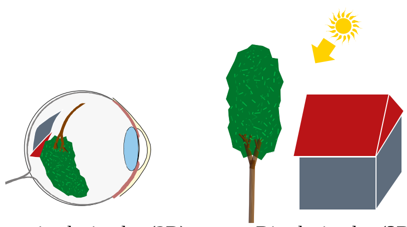
Fredo Durand
MIT- Lab for Computer Science

Plan

- Vision as an cognitive process
- Computational theory of vision
- Complex mapping

Distal vs. proximal stimulus

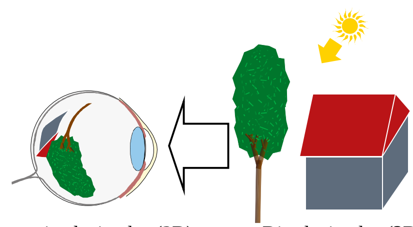
- Distal stimulus: reality
- Proximal stimulus: retinal image



proximal stimulus (2D) Distal stimulus (3D)

Vision as an inverse problem

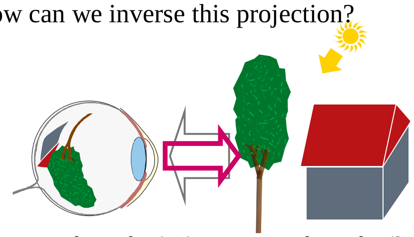
- The distal stimulus is projected into a proximal stimulus



proximal stimulus (2D) Distal stimulus (3D)

Vision as an inverse problem

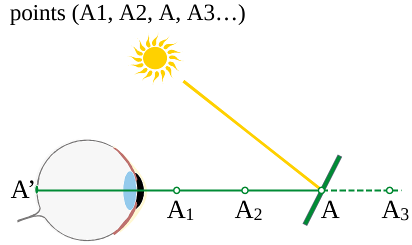
- The distal stimulus is projected into a proximal stimulus
- How can we inverse this projection?



proximal stimulus (2D) Distal stimulus (3D)

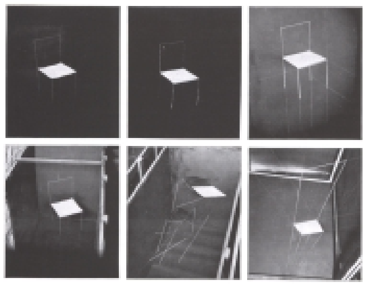
Unconscious inference (Helmholtz)

- Our vision system solves a problem
- Under-constrained problem
  - A visible point A' can correspond to an infinity of 3D points (A1, A2, A, A3…)



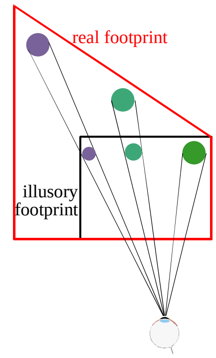
How assumptions help

- Ames chair
  - 3 different scenes
  - Same projection
  - We assume it is a chair
  - Resolves ambiguity
  - Can be wrong



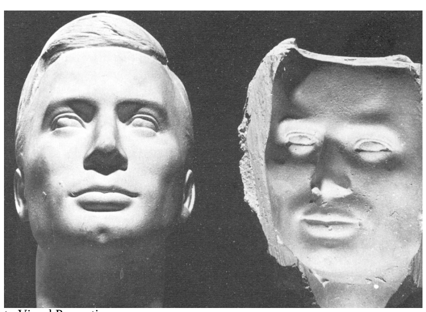
The Ames room

- Invalid assumption
  - Walls perpendicular
- Wrong conclusions
  - Men have different sizes



Positive and hollow face

- Both seen convex because hollow faces are rare!

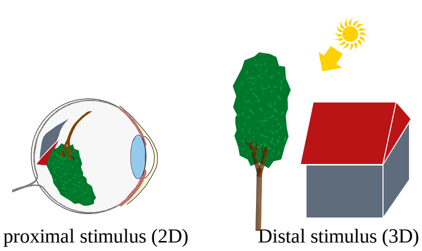


Constancy & architecture

- Palazzo Spada in Rome (by Boromini)
- Short corridor
- Column size decreases
- Appears longer

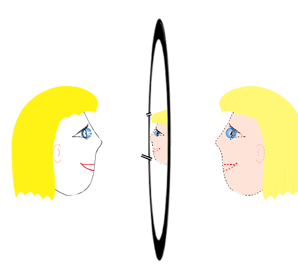
The paradox of vision

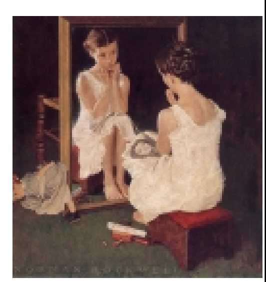
- Available information: proximal stimulus
- Conscious information: distal stimulus



Face in mirror

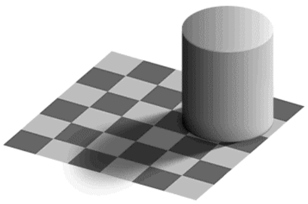
- When you look at yourself in a mirror, the size of your image is half your real size





Brightness vs. lightness

- Brightness: subjective amount of light
- Lightness: how "white"

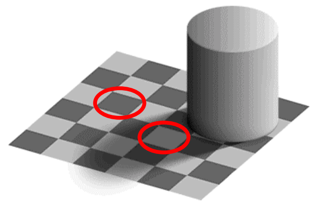


Illusion by
Ted Adelson

The white cells in shadow are as dark as the black illuminated cells

Brightness vs. lightness

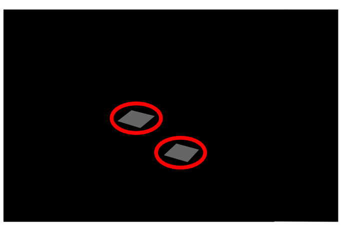
- Brightness: subjective amount of light
- Lightness: how "white"



The white cells in shadow are as dark as the black illuminated cells

Brightness vs. lightness

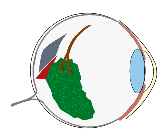
- Brightness: subjective amount of light
- Lightness: how "white"



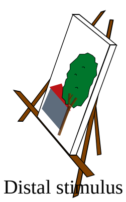
The white cells in shadow are as dark as the black illuminated cells

The paradox of Pictures

- Distal vs. proximal
- Available information: proximal stimulus
- Conscious information: distal stimulus



proximal stimulus (2D)



Distal stimulus (2D/3D)

Pictures and the inverse problem

- Pictures can
  - Simplify the analysis
  - Be a puzzle, a riddle
- Picture making

Plan

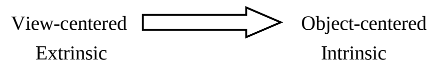
- Vision as an cognitive process
- Computational theory of vision
- Complex mapping

Vision as information processing

- Input: retinal image
- Output: 3D layout, object recognition, etc.

Computational theory of vision

- Marr's stages (extended by Palmer et al.)
- Human and Computer Vision
- Classification of different kinds of processes
- Has proved fruitful in art studies



Retinal image

- Intensity: hard to comprehend

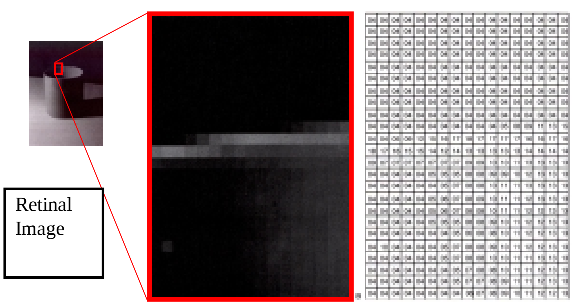
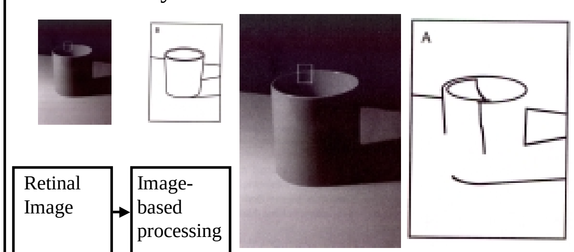


Image-based (primary sketch)

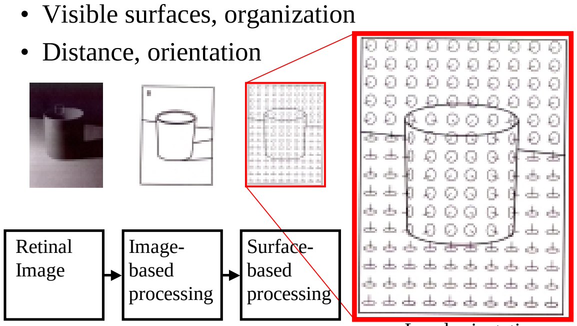
- Contrast, edge detection
- Not so easy



Raw edge detection

Surface-based

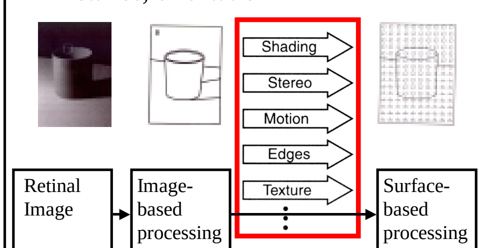
- Visible surfaces, organization
- Distance, orientation



Local orientation

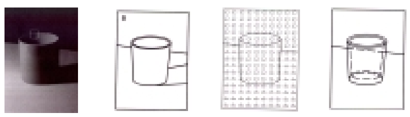
Surface-based

- Visible surfaces, organization
- Distance, orientation



Object-based

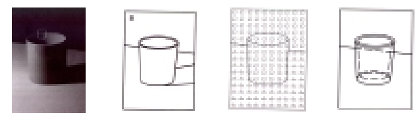
- 3D properties, structure
- Nature of the description highly discussed



Retinal Image → Image-based processing → Surface-based processing → Object-based processing

Category-based

- Recognition, category, function

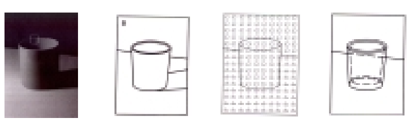


Cup

Retinal Image → Image-based processing → Surface-based processing → Object-based processing → Category-based processing

Feedback

- Bottom-up and top-bottom



Cup

Retinal Image → Image-based processing → Surface-based processing → Object-based processing → Category-based processing

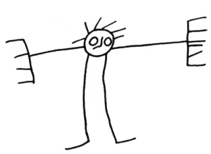
Scope of the theory

- Computer Vision
- Human Vision
- No direct correspondence in the brain
- Has proved fruitful conceptual tool

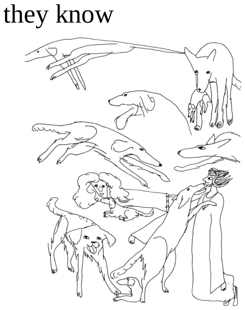
Retinal Image → Image-based processing → Surface-based processing → Object-based processing → Category-based processing

Relation to children drawing

- First children draw what they know
  - Object-centered
- Then, what they see
  - View-centered



Age 5

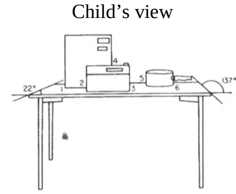


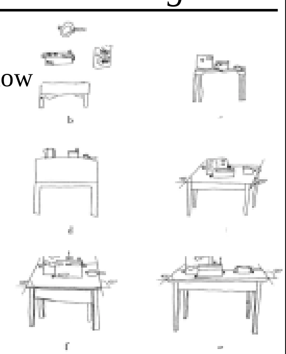
Age 9 (gifted!)

Evolution of children's drawings

- Asked to draw a table
- First, draw what they know
- Later, what they see

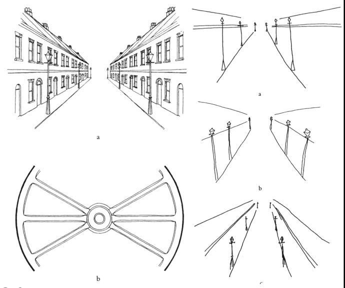
Child's view





What about adults?

- Reproduce two drawing with similar angles
- Wheel:
  - Accuracy ~5°
- Street:
  - Error: 32 °
- Because in the first case, they focus on the 3D (distal) interpretation



Drawing reproduction

- From *Drawing on the right side of the brain*
- Betty Edwards advises to reproduce drawings upside down
- This way, the distal interpretation does not impede the precise reproduction
- Forgers often reproduce paintings upside-down

Relation to pictures

- Different classes of pictures for different stages
- Not a strict classification

View-centered ⟹ Object-centered

Extrinsic Intrinsic

Relation to pictures

- Chinese painting refuse extrinsic, only essential
- No shadow

View-centered ⟹ Object-centered

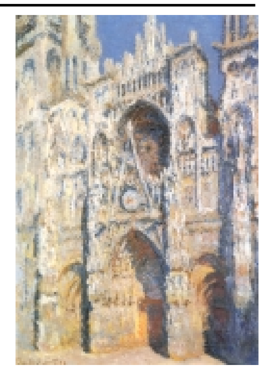
Extrinsic Intrinsic

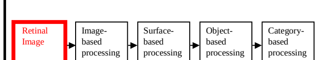
Retinal image

- Turner
- "My business is to paint not what I know, but what I see"

Retinal image

- Impressionism





Retinal image

- Impressionism
- Photography

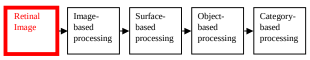


Image-based

- Line Drawing
- Rivera

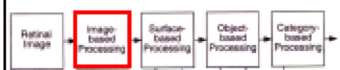
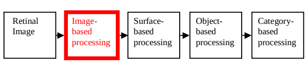


Image-based

- Line Drawing
- Ingres

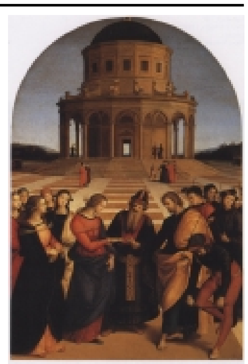


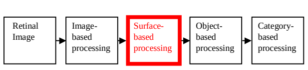
Intrinsic vs. Extrinsic

- Visual angle vs. true size
- Vermeer:
  too accurate to be true!

Intermediate

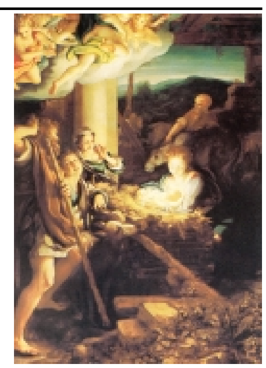
- View-based
- Cues for surface-based feature extraction are enhanced
  - Depth cues
  - Orientation cues
- No subjective feature (e.g. lighting)

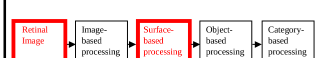




Intermediate

- View-based
- Cues for surface-based feature extraction are enhanced
  - Depth cues
  - Orientation cues
- More subjective feature (lighting)





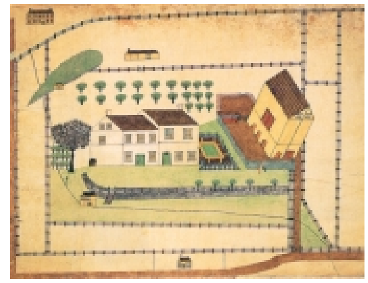
Higher level

- Primitive art
- Cubism
- Schema
- "What I know"

Retinal Image → Image-based processing → Surface-based processing → **Object-based processing** → Category-based processing

Higher level

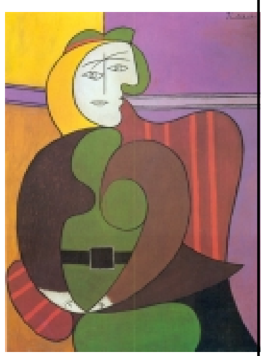
- Primitive art
- Cubism
- Schema
- "What I know"



Retinal Image → Image-based processing → Surface-based processing → **Object-based processing** → Category-based processing

Higher level

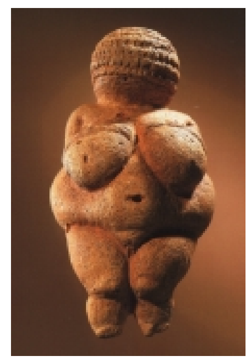
- Primitive art
- Cubism
- Schema
- "What I know"



Retinal Image → Image-based processing → Surface-based processing → **Object-based processing** → Category-based processing

Higher level

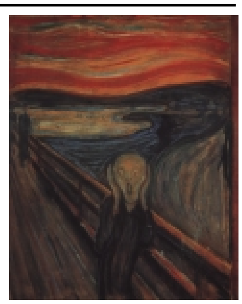
- Primitive art
- Cubism
- Schema
- "What I know"
- Not limited to picture



Retinal Image → Image-based processing → Surface-based processing → **Object-based processing** → Category-based processing

Expressionism

- "What I feel"



Retinal Image → Image-based processing → Surface-based processing → Object-based processing → **Category-based processing | Other mode**

Relation with 2D/3D emphasis

- Almost the opposite!
- 3D impression corresponds to retinal image
- 2D quality arises from higher-level pictures
- Because of vision paradox
  - Distal is seen when proximal is shown

Relation with 2D/3D qualities

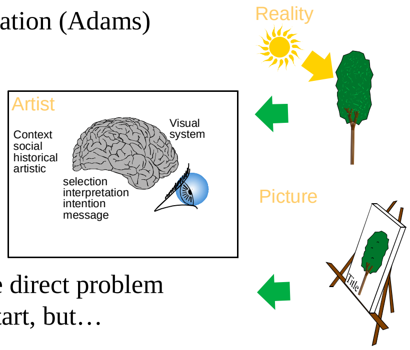
- 3D impression but Retinal image

Relation with 2D/3D qualities

- 2D emphasis but
  Higher level

Making pictures: inverse of inverse

- Previsualization (Adams)



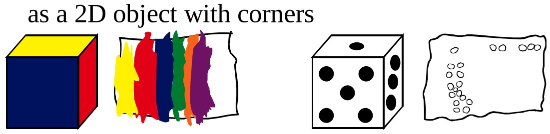
- Solving the direct problem
  is a good start, but…

Plan

- Vision as an cognitive process
- Computational theory of vision
- Complex mapping

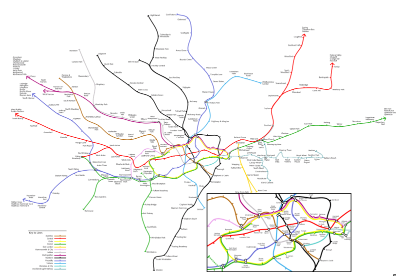
3D and 2D attributes

- [Willats 97]
- Show coloured or numbered die to children (6-7)
- The still draw a rectangle
- But different colours or many points
- The rectangle stands for the whole dice
- The notion of 3D object with corners is translated as a 2D object with corners



Projection: Topographical

- London underground
- Metric properties are used



Projection: Topological

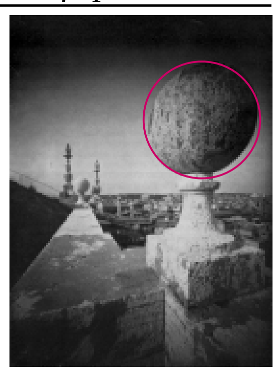
- Beck's map of London underground, 1931
- Only the connectedness and organization are preserved
- [Agrawala, in this volume]

Spheres and symmetry

- Linear perspective states that a sphere projects as an ellipse
- This maps a perfectly symmetric 3D object to a not so symmetric 2D object
- It is thus preferable to depict a sphere as a disk, breaking the rule of linear perspective
- Mapping properties can be more important than projective geometry
- See also [Zorin, page 115] [Durand page 11]

Perspective distortion of spheres

- The sphere is projected as an ellipse
- It is perceived as distorted

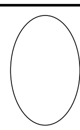


Spheres and symmetry

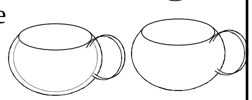
- Raphael, The School of Athens
- The sphere is depicted as a disk

Mapping of curvature

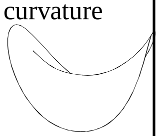
- Convex: positive curvature
  - 3D example: Egg
  - 2D: Convex contour



- Concave: negative curvature
  - 3D example: Interior of cup
  - 2D: Nothing, hidden contour



- Saddle: mix of positive and negative curvature
  - 3D example:Saddle (surprising!)
  - 2D: Concave contour

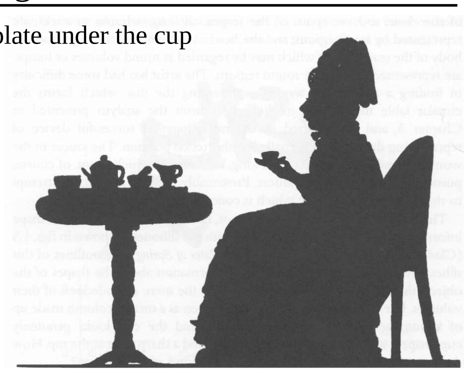


Mapping of curvature

- But some artists map 3D concave objects to 2D concave outlines
- This maps the property of concavity
- The left view of the plate is more “correct” but does not convey the notion of concavity

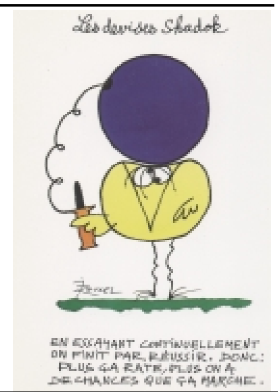
Mapping of curvature

- Small plate under the cup



Mapping of curvature

- Complex denotation
- See [Durand, page 15]



Further reading

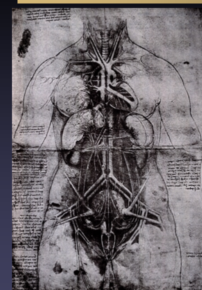
Automating the Design of Visualizations

Maneesh Agrawala

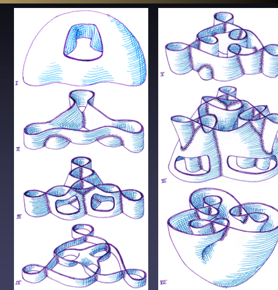
April 8, 2002

Stanford University

Visualization: Explore & Present Data

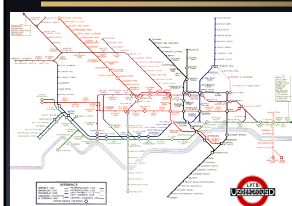


Principal Organs & Vascular System [Leonardo da Vinci ca. 1490]

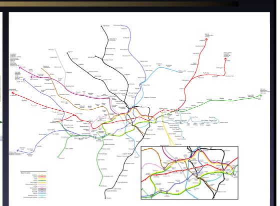


Strange Immersion of Torus in 3-Space [Curtis 92]

Good Design Improves Usability



London Underground [Beck 33]



Geographic Version of Map

- Visualizations are common
  - Newspapers, textbooks, training manuals, scientific papers, …
- Creating effective designs is time-consuming

Challenge

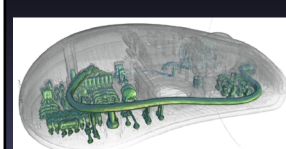
- Best visualizations are designed by humans
- Computing becoming ubiquitous
  - Data collection / dissemination getting faster
  - Most displays computer generated
- Therefore: Visualizations are regressing
- **Can we build automated systems capable of designing effective visualizations?**

Automation Allows Customization

- **Purpose:** Present data relevant to specific goals
- **Device:** Adapt to capabilities of display
- **Situation:** Update as data / goals change
- **Person:** Adapt to knowledge of user
- Customization increases effectiveness

Emulating Artistic Rendering Styles

- Artistic rendering can improve perception



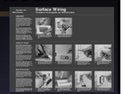
Mouse [Lum & Ma 02]

House [Winkenbach & Salesin 94]

- **High-level design still specified manually**

Automated Design as Optimization

- Page design
  - TeX [Knuth 81], GRIDS [Feiner 88], LayLab [Graf 92], [Weitzman & Wittenburg 94], [Borning et al. 97, 00]



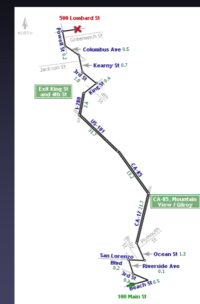
- 3D object visualization
  - APEX [Feiner 85], IBIS [Seligmann & Feiner 91], WIP [Rist et al. 94]

- Data graphics presentations
  - APT [Mackinlay 86], SAGE [Roth et al. 94, 96], SYSTAT [Wilkinson 99]

- UI layout, Label layout, VLSI design, Camera planning, 2D/3D packing, Graph drawing, …
- **Need domain specific constraints**

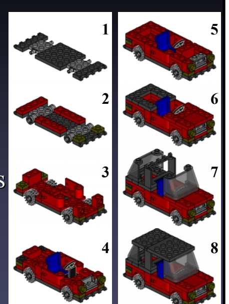
Contributions

- **Analysis**
  - Identify design principles
    - Route maps
    - Assembly instructions
- **Synthesis**
  - Automated design systems



Contributions

- **Analysis**
  - Identify design principles
    - Route maps
    - Assembly instructions
- **Synthesis**
  - Automated design systems

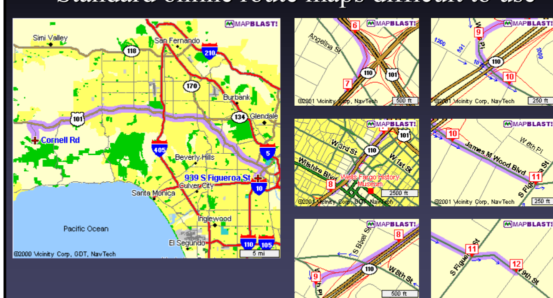


Outline

- Motivation
- Automated Route Map Design
- Framework for Automated Design
- Automated Assembly Instruction Design
- Future Directions

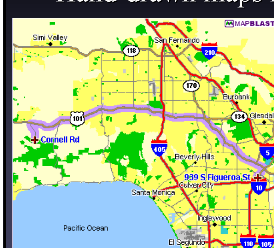
Visualizing Routes

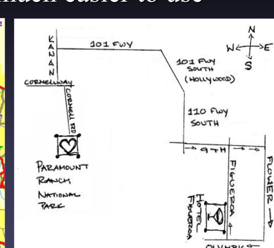
- Standard online route maps difficult to use



A Better Visualization

- Hand-drawn maps much easier to use



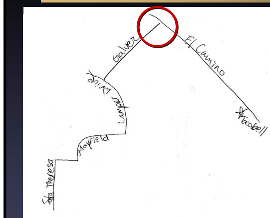


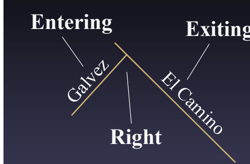
Communicative Intent of Route Maps

- Route is a sequence of turns [Tversky 92] [MacEachren 95]
  1. Start at 100 Serra
  2. Turn Right on University
  3. Turn Left on El Camino
  4. Turn Right on San Antonio

  …
  - Verbal directions emphasize turns [Denis 97]
  - Hand-drawn maps highlight turns [Tversky & Lee 99]
- **Maps must communicate turning points**

Depicting Turns

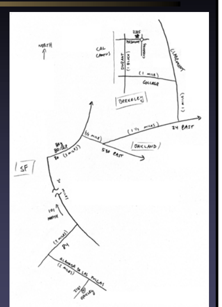




- Pair of roads (entering / exiting the turn)
- Turn direction (left / right)
- **These graphic elements must be visible**

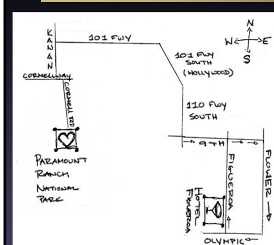
Context can Facilitate Navigation

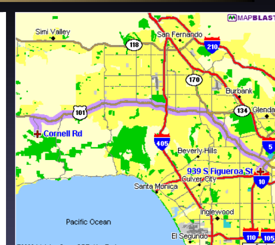
- Local context
  - Consistency checks
    - Cross-streets
    - Landmarks along route
    - Distance along each road
- Overview context
  - Orient route to geography
    - Large area landmarks
    - Overall shape & heading



- **Context is secondary to turning points**

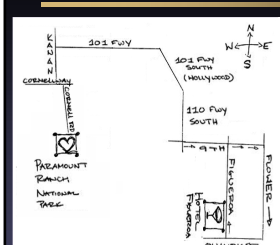
Geometric Properties Distorted



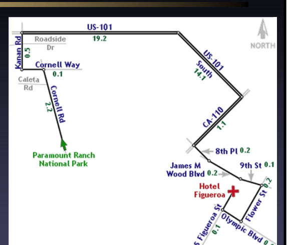


- Geometry *not* apprehended accurately [Tversky 81]
- Geometry *not* drawn accurately [Tversky & Lee 99]
  - Topology is accurate

LineDrive: Route Map Design System



Hand-drawn Route Map



LineDrive Route Map

Automating Route Map Design

- Layout problem
  - Set of graphic elements
    - Roads
    - Labels
    - Cross-streets
  - Choose visual attributes
    - Position
    - Orientation
    - Size
- Distortions increase choices
- Large space of possible layouts

Layout as Search-Based Optimization

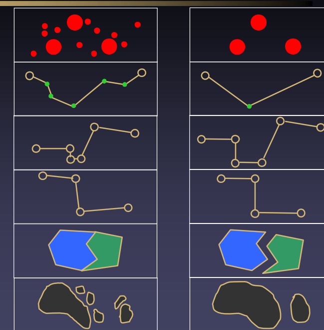
- Hard constraints
  - Required characteristics
- Soft constraints
  - Desired characteristics

- **Challenge: Develop relevant constraints**

- Simulated annealing
  - *Perturb:* Form a layout
  - *Score:* Evaluate quality
  - Minimize score

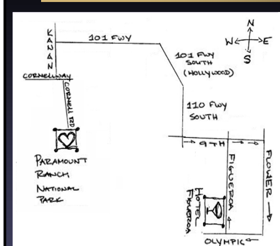
Cartographic Generalization

- Selection
- Simplification
- Exaggeration
- Regularization
- Displacement
- Aggregation



[Monmonier 96], [MacEachren 94], [DiBiase 91]

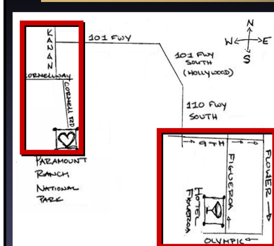
Three Generalizations for Route Maps

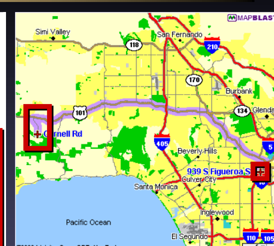


- Our observations from hand-drawn examples:
  - Exaggeration
    - Road length
  - Regularization
    - Turning angle
  - Simplification
    - Road shape

- **Generalizations emphasize turning points!**

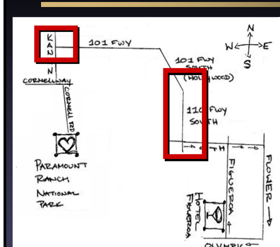
Exaggeration: Length Generalization

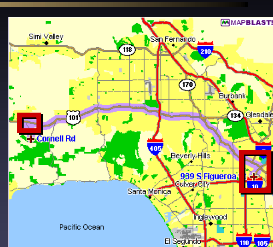




- Grow short roads, shrink long roads
  - Ensures all roads visible
  - Maintain relative ordering by length

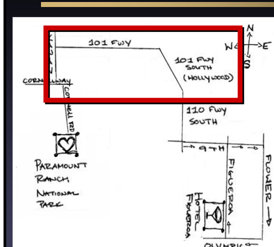
Regularization: Angle Generalization

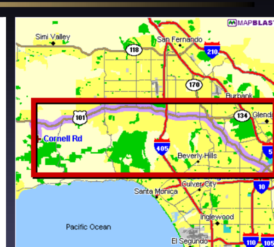




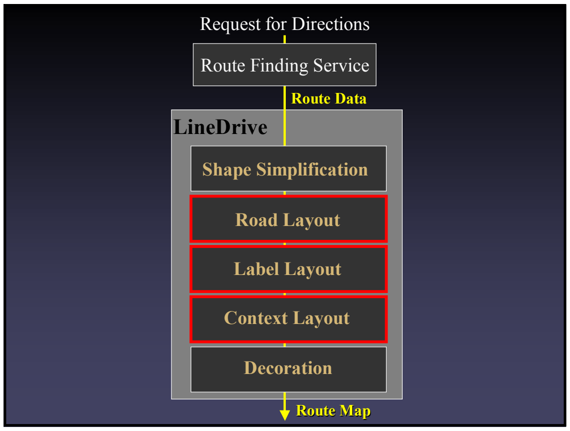
- Regularize turning angles
  - Reduces visual complexity
  - Maintain consistent turn direction

Simplification: Shape Generalization



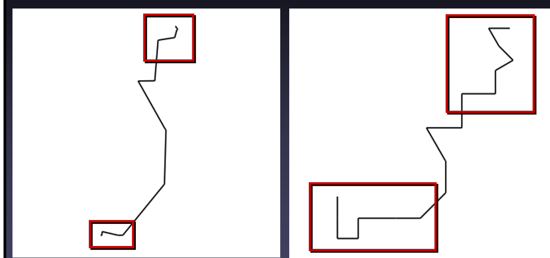


- Simplify roads to straight lines
  - Differentiates roads and turning points
  - Maintain overall shape of route



Stage 2: Road Layout

- Goal: Choose road lengths & orientations



without road layout — with road layout

Road Layout Search

- Initialize
  - Uniformly scale route to fit given viewport
- Perturb
  - Pick random road
  - Either
    - Rescale by random factor
    - Reorient by random angle
  - Rescale entire route to fit viewport
- Hard Constraints
  - Must fit in viewport
  - Must maintain consistent turn direction

Designing Soft Constraints

- Challenges
  - Choose desirable characteristics
  - Express as numerical score function
  - Balance constraints, deal with conflicts
- Desired characteristics for road layout
  - All roads visible
  - Prevent excessive distortion

Constraints

- **Length**
  - Ensure all roads visible — $((L_{min} - l(r_i))/L_{min})^2 * W_{small}$
  - Maintain ordering by length — $W_{shuffle}$
- **Orientation**
  - Maintain original orientation — $|a_{curr}(r_i) - a_{orig}(r_i)| * W_{orient}$
- **Topological errors**
  - Prevent false — $\min(d_{origin}, d_{dest}) * W_{false}$
  - Prevent missing — $d * W_{missing}$
  - Ensure separation — $\min(d_{ext}, E) * W_{ext}$
- **Overall route shape**
  - Maintain endpoint direction — $|a_{curr}(v) - a_{orig}(v)| * W_{enddir}$
  - Maintain endpoint distance — $|d_{curr}(v) - d_{orig}(v)| * W_{enddist}$

Ensure All Roads Visible

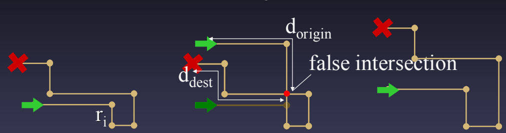
- Penalize roads shorter than min length L_{min}

Penalty = $((L_{min} - l(r_i))/L_{min})^2 * W_{small}$

bad layout — desired layout

Prevent False Intersections

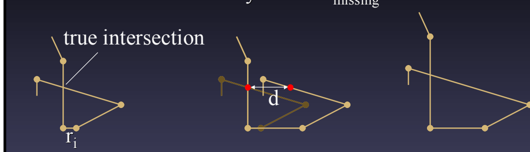
$$\text{Penalty} = \min(d_{origin}, d_{dest}) * W_{false}$$



original layout | bad layout | desired layout

Prevent Missing Intersections

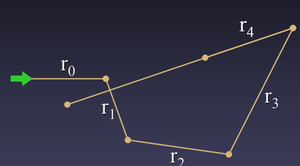
$$\text{Penalty} = d * W_{missing}$$



original layout | bad layout | desired layout

Efficiency: Road Intersections

- Computing road-road intersection is expensive
- But most roads cannot intersect

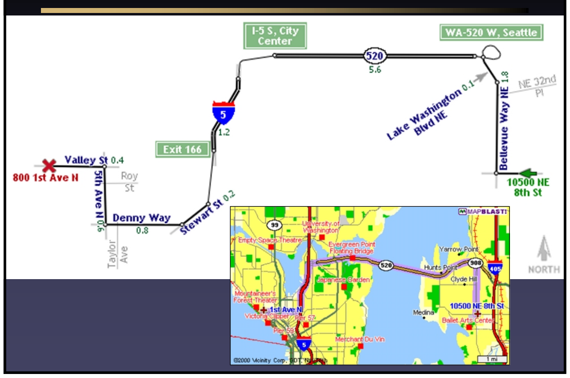


- Intersection requires
  - Turning angle interval > 180.0
  - Route has to loop back towards itself

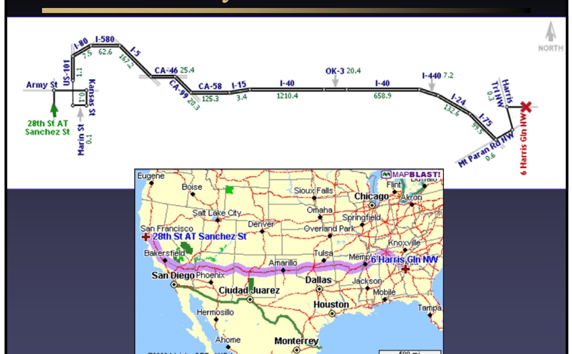
Balancing Soft Constraints

- Prioritize scores by importance
  1. Prevent topological errors
  2. Ensure all roads visible
  3. Maintain original orientation
  4. Maintain ordering by length
  5. Maintain overall route shape
- Informal usability engineering
  - Consider maps containing errors
  - Rate which errors most confusing

Bellevue to Seattle



Cross-Country Route



Limited Resolution: Palm

User Response

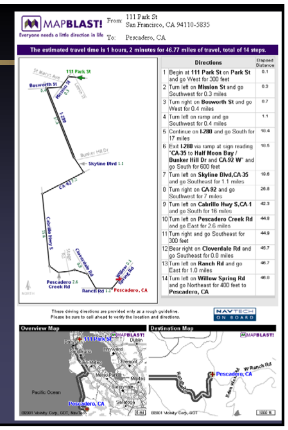
- Beta publicly accessible   Oct 00 – Mar 01
- 150,000 maps served
- 2242 voluntary responses
  - Should replace standard maps   55.6 %
  - Use along with standard maps   43.5 %
  - Standard maps preferable   0.9 %
- Most common suggestion
  - Choose better routes (not a LineDrive issue)

System Performance

- 7727 routes   (sampled over 1 day at MapBlast!)
  - Median distance   52.5 miles
  - Median number turning points   13
  - Median computation time   0.7 sec
  - Short roads   5.4 %
  - False intersections   0.3 %
  - Missing intersections   0.2 %
  - Label-label overlap   0.5 %
  - Label-road overlap   11.7 %

Current Status

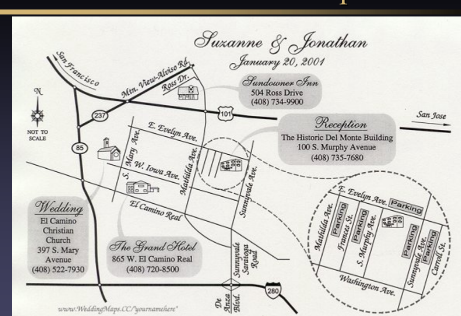
- Default rendering style
  www.mapblast.com
- 250,000 maps/day



Next Steps

- Map enhancements
  - Cross-street after turning point
  - Large area landmarks
- In-depth user study
  - Watch users following LineDrive maps

Future: Point Location Maps



Hand-designed Wedding Map [www.WeddingMaps.CC 01]

Outline

- Motivation
- Automated Route Map Design
- Framework for Automated Design
- Automated Assembly Instruction Design
- Future Directions

Two-Step Approach

Analyze cognitive science research and examples of most effective hand-designed visualizations

↓

Step 1: Identify visualization design principles

↓

Low-level visualization design principles

↓

Step 2: Encode principles as constraints and algorithmically find design satisfying constraints

↓

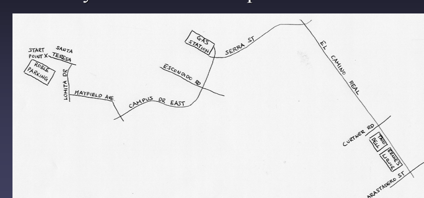
Automated design system

Step 1: Identify Design Principles

- Cognitive science
  - How people *conceive* information
  - How people *apprehend* visual representations
- **Conception**
  - Routes conceived as sequence of turns
- **Apprehension**
  - Route geometry not apprehended accurately
- **High-level cognitive model**

Step 1: Identify Design Principles

- Analyze hand-designed visualizations
  - Identify essential graphic elements
  - Identify distortion techniques



- **Low-level visualization design principles**

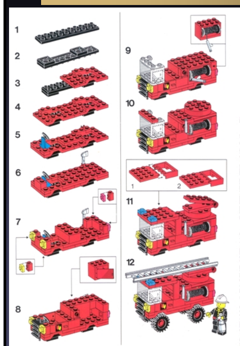
Step 2: Build Automated Algorithm

- Space of possible visualization designs
  - Graphic elements
  - Visual attributes
- Design principles → Constraints
  - Generative rules: How to vary visual attributes
  - Evaluation criteria: Measure effectiveness
  - **Main algorithmic challenge**
- Find most effective visualization design
  - Search-based optimization
  - Balance constraints
  - Efficiency

Outline

- Motivation
- Automated Route Map Design
- Framework for Automated Design
- Automated Assembly Instruction Design
- Future Directions

Assembly Instructions



Goal: Create step-by-step instructions from 3D model

Geometric model in assembled configuration

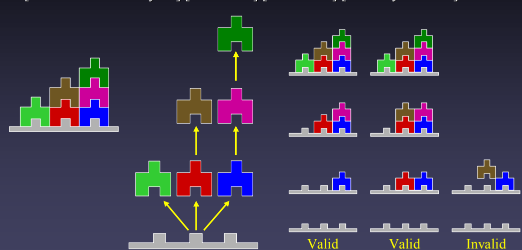
Compute geometrically valid assembly sequences

Evaluate effectiveness of sequences and choose most *effective* assembly sequence

Assembly instructions

Geometrically Valid Sequences

- Robotics / Mechanical Engineering
  [DeFazio & Whitney 87] [Wolter 89] [Wilson 95] [Romney et al. 95]



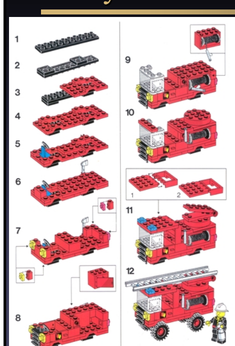
Many Geometrically Valid Sequences

- **How do we choose most effective sequence?**

Cognitive Science

- Experiments to learn how people understand assembly instructions [Heiser in progress]
- **Assemblies conceived as groupings of parts**
  - Coarse level - functional units
  - Finer levels - symmetry, similarity, proximity
- **People prefer certain assembly sequences**
  - Add *all* supporting parts then supported parts
  - Add *all* internal parts then external parts
  - Add grouped parts in same step, or in sequence
  - Add new parts onto existing parts

Analysis of Hand-Designed Examples



- **Essential graphic elements**
  - Parts added in step (visibility)
  - Previous parts (context)
- **Graphic design techniques**
  - Small multiples
  - Technical illustration style
  - Insets improve part visibility
  - Arrows show attachments

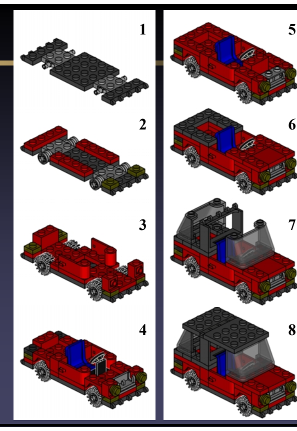
Constraints

- **Support**: All supporting parts added before supported
- **Adjacency**: All parts in step touch previous parts
- **Symmetry**: All symmetric parts added in same step
- **Linearity**: New parts added onto existing parts

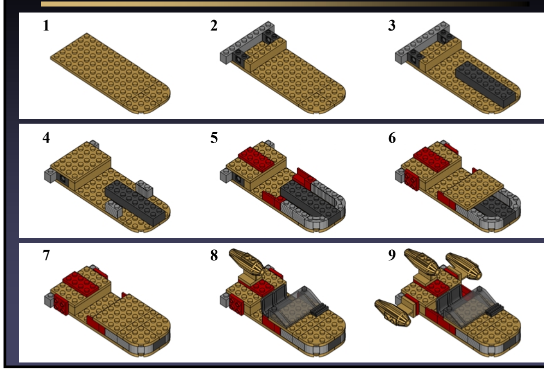
- **Visibility**: If part *A* occludes *B*
  Penalty = Occlusion (A, B) * $W_{visibility}$

- **Context**: If < 25% of step *N-1* parts visible
  Penalty = Occlusion (Step N, Step $N-1$) * $W_{context}$

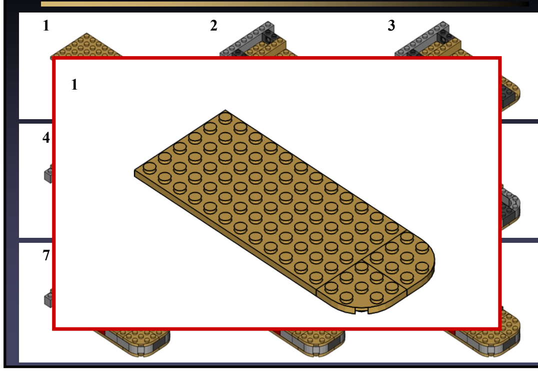
Lego Car



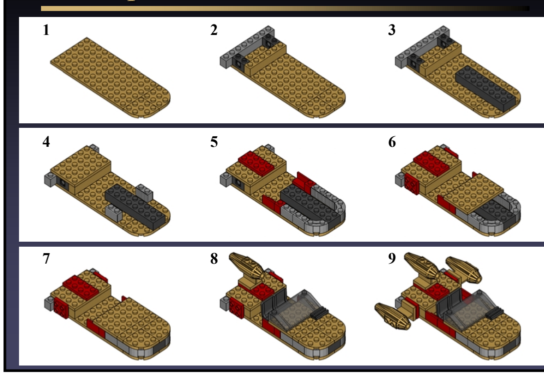
Landspeeder



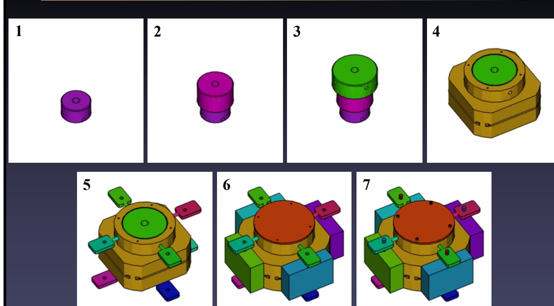
Landspeeder



Landspeeder



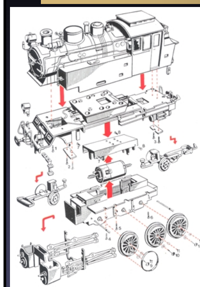
Mechanical Assembly



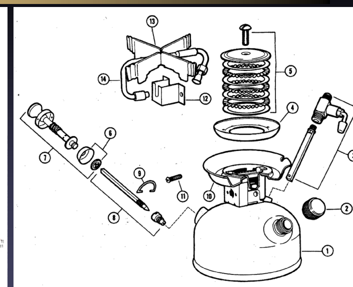
Current Agenda

- Identify more design principles
- Incorporate other graphic design techniques
  - **Insets**
  - **Arrows**
  - Scale exaggeration
  - Cutaways
  - Sections
  - Text labels
- User studies

Future: Exploded Views



Train [from Mijksenaar 99]



Camping Stove [from Mijksenaar 99]

Summary

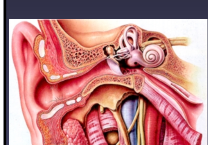
- General two-step approach
  - Step 1: Identify cognitive design principles
  - Step 2: Encode principles as constraints and find most effective visualization
- Automated design systems
  - Route maps
  - Assembly instructions
- Benefits
  - Novices can leverage skills of experts
  - Deal with data overload

Outline

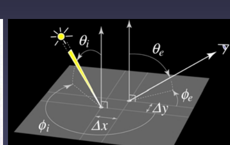
- Motivation
- Automated Route Map Design
- Framework for Automated Design
- Automated Assembly Instruction Design
- Future Directions

Many Other Domains To Consider

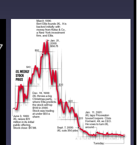
- Medical illustration: Complex biological organisms
- Scientific diagrams: Depict scientific concept
- Graphs and charts: Scatter plots, bar charts, etc.
- Architectural plans: Room and furniture layout
- Proof visualization: Depict complex logical statements



Medical Illustration



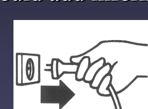
Scientific Diagram



Graphs and Charts

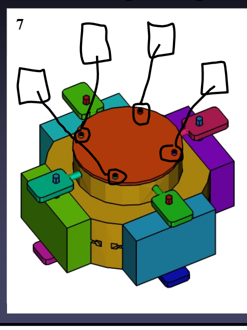
Interaction and Animation

- Interaction
  - Hide clutter, let user request details
  - Direct, intuitive, navigation controls
- Animation
  - Should add information [Hegarty 00] [Morrison 01]



Human Guidance

- Sketch edits on computer-generated designs



Long-Term Challenge

- Current focus on *how*
  - Simulate realistic lighting, shading
  - Emulate artistic media (paint, pen & ink, …)
  - Display data using std. metaphors (bar graph, binary tree, …)
  - …
- Need principles guiding *where, what, why*
  - Where to place lights to communicate a mood?
  - What information does an artistic rendering style convey?
  - Why is a particular metaphor effective?
  - …
- **Must understand and appreciate what makes an effective visualization**

Acknowledgements

- Pat Hanrahan
- Chris Stolte
- Barbara Tversky
- Boris Yamrom
- Vicinity Corporation

Rendering Effective Route Maps: Improving Usability Through Generalization

Maneesh Agrawala Chris Stolte

Stanford University*

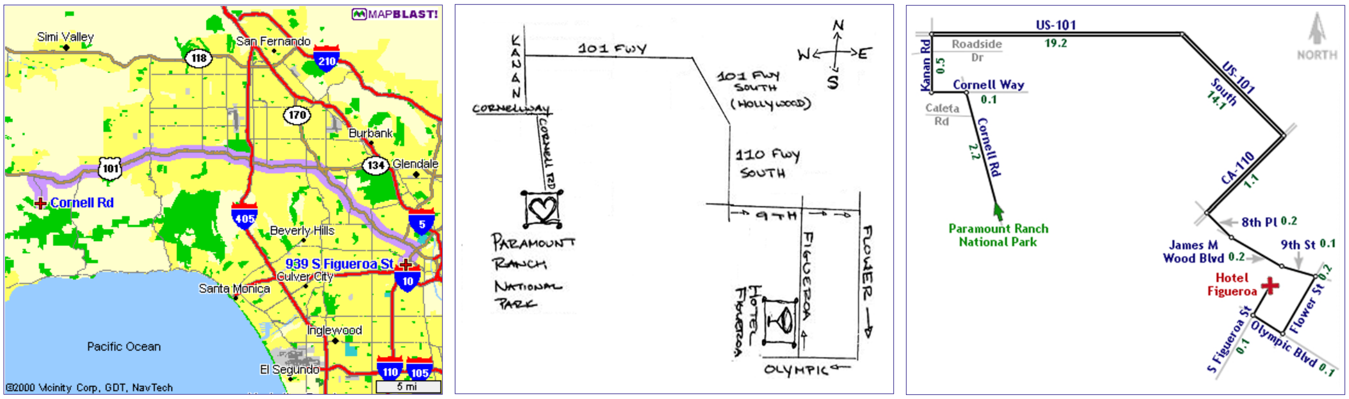


Figure 1: Three route maps for the same route rendered by (left) a standard computer-mapping system, (middle) a person, and (right) LineDrive, our route map rendering system. The standard computer-generated map is difficult to use because its large, constant scale factor causes the short roads to vanish and because it is cluttered with extraneous details such as city names, parks, and roads that are far away from the route. Both the handdrawn map and the LineDrive map exaggerate the lengths of the short roads to ensure their visibility while maintainaing a simple, clean design that emphasizes the most essential information for following the route. Note that the handdrawn map was created without seeing either the standard computer-generated map or the LineDrive map. *(Handdrawn map courtesy of Mia Trachinger.)*

Abstract

Route maps, which depict a path from one location to another, have emerged as one of the most popular applications on the Web. Current computer-generated route maps, however, are often very difficult to use. In this paper we present a set of cartographic generalization techniques specifically designed to improve the usability of route maps. Our generalization techniques are based both on cognitive psychology research studying how route maps are used and on an analysis of the generalizations commonly found in handdrawn route maps. We describe algorithmic implementations of these generalization techniques within LineDrive, a real-time system for automatically designing and rendering route maps. Feedback from over 2200 users indicates that almost all believe LineDrive maps are preferable to using standard computer-generated route maps alone.

Keywords: Information Visualization, Non-Realistic Rendering, WWW Applications, Human Factors

1 Introduction

Route maps, which depict a path from one location to another, are one of the most common forms of graphic communication. Although creating a route map may seem to be a straightforward task, the underlying design of most route maps is quite complex. Mapmakers use a variety of cartographic generalization techniques including distortion, simplification, and abstraction to improve the clarity of the map and to emphasize the most important information [16, 21]. This type of generalization, performed either consciously or sub-consciously, is prevalent both in quickly sketched maps and in professionally designed route maps that appear in print advertisements, invitations, and subway schedules [25, 13].

Recently, route maps in the form of driving directions have become widely available through the Web. In contrast to hand-designed route maps, these computer-generated route maps are often more precise and contain more information. Yet these maps are more difficult to use. The main shortcoming of current systems for automatically generating route maps is that they do not distinguish between essential and extraneous information, and as a result, cannot apply the generalizations used in hand-designed maps to emphasize the information needed to follow the route.

Figure 1 shows several problems arising from the lack of differentiation between necessary and unnecessary information. The primary problem is that current computer-mapping systems maintain a constant scale factor for the entire map. For many routes, the lengths of roads can vary over several orders of magnitude, from tens of feet within a neighborhood to hundreds of miles along a highway. When a constant scale factor is used for these routes, it forces the shorter roads to shrink to a point and essentially vanish. This can be particularly problematic near the origin and destination of the route where many quick turns are often required to enter or exit a neighborhood. Even though precisely scaled roads might help navigators judge how far they must travel along a road, it is far more important that all roads and turning points are visible. Handdrawn maps make this distinction and exaggerate the lengths of shorter roads to ensure they are visible.

Another problem with computer-generated maps is that they are often cluttered with information irrelevant to navigation. This extraneous information, such as the names and locations of cities, parks, and roads far away from the route, often hides or masks information that is essential for following the route. The clutter makes the maps very difficult to read, especially while driving. Handdrawn maps usually include only the most essential information and are very simple and clean. This can be seen in figure 1(middle) where even the shape of the roads has been distorted and simplified to improve the readability of the map. Furthermore, distorting

*(maneesh,cstolte)@graphics.stanford.edu

the lengths of shorter roads and removing unnecessary information makes it possible to include and emphasize helpful navigational aids such as major cross-streets or landmarks before the turns.

Despite the fact that the distortion techniques used in hand-designed maps improve usability, there has been surprisingly little work on developing automatic cartographic generalization techniques based on these distortions. Existing research on automatic generalization has focused on developing simplification and abstraction techniques for standard road, geographical, and political maps [4, 16, 14]. Unlike route maps, these general purpose maps are designed to convey information about an entire region without any particular focus area. Therefore, these maps cannot include the specific types of distortion that are used in route maps.

This paper presents two main contributions:

Route Map Generalization Techniques: We have developed a set of generalization techniques specifically designed to improve route map usability. Our techniques are based on cognitive psychology research showing that an effective route map must clearly communicate all the turning points on the route [6], and that precisely depicting the exact length, angle, and shape of each road is much less important [28]. We consider how these techniques are applied in handdrawn maps and show that by carefully distorting road lengths and angles and simplifying road shape, it is possible to clearly and concisely present all the turning points along the route.

Automatic Route Map Design System: We describe LineDrive, an automatic system for designing and rendering route maps. LineDrive takes advantage of our route map generalization techniques to produce maps that are much more usable than those produced by standard computer-based map rendering systems. LineDrive performs a focused randomized search over the large space of possible map designs to quickly find a near-optimal layout for the roads, labels, and context information. An example of a LineDrive map is shown in figure 1(right). Feedback from over 2200 users indicates that almost all believe LineDrive maps are preferable to using standard computer-generated route maps alone.

In computer graphics we usually consider distortion and abstraction techniques within the area of non-photorealistic rendering. To apply these techniques to visualization requires understanding how the techniques can improve the perception, cognition, and communicative intent of an image. Earlier examples of this approach to visualization include Mackinlay's [17] investigation of methods for automating chart and graph design, Seligmann and Feiner's [23] research on the automatic design of intent-based illustrations, and Interrante's [15] work on using illustration techniques to improve the perception of 3D surface shape in volume data. In this paper we extend this same approach to the automatic design of route maps.

The remainder of this paper is organized as follows. In section 2, we examine the specific generalization techniques applied in handdrawn route maps and how these techniques improve map usability. Section 3 describes algorithmic implementations of these techniques in LineDrive. Results are presented in section 4, and section 5 discusses conclusions and future work.

2 Route Map Design

In order to design a better route map, we begin by analyzing the tasks involved in following a route. From this analysis, we identify the essential information a route map must communicate to support these tasks. We then describe how we use specific generalization techniques, including distortion and abstraction, to present this information in a clear, concise, and convenient form.

2.1 Information Conveyed by Route Maps

Understanding how people think about and communicate routes can provide great insight into what information should be emphasized in a computer-generated route map. A common theory in the field of cognitive psychology is that people think of routes as a sequence of turns [27, 16]. It has been shown that verbal route directions are generally structured as a series of turns from one road to the next and that emphasis is placed on communicating turn directions and the names of the roads [7]. Tversky and Lee [28] have shown that handdrawn maps maintain a similar structure with emphasis on communicating the roads and turn direction at each turning point.

A turning point can be defined by a pair of roads (the road entering and the road exiting the turning point) and the turn direction (left or right) between those two roads. Route maps depict this information visually, so navigators can quickly scan the map to find the road they are currently following and look ahead to determine the name of the next road they will turn onto. Once the name of the next road is known, the navigator can search for the corresponding road in the physical world. The turn direction specifies the action navigators must take at the turning point.

Although it is possible to follow a route map that only indicates the road names and turn direction at each turning point, additional information can greatly facilitate navigation and is often included in hand-designed maps. For example, if the map labels each road with the distance to be travelled along that road, navigators can use their odometer to determine how close they are to the next turn. Cross-streets and local landmarks along the route, such as buildings, bridges, rivers, and railroad tracks, can also be used for gauging progress. Navigators can also use this information to verify that they are still following the route and did not miss a turn. However, cross-streets and local landmarks are not essential for following the route and are usually included in the map only when they do not interfere with the primary turning point information.

2.2 Generalizing Route Maps

Although route maps may be used before a trip for planning purposes, they are most commonly used while actually traversing the route. In many cases, navigators are also drivers and their attention is divided between many tasks. As a result, they can only take quick glances at the map. Therefore, maps must convey the turning point information in a clear, easy-to-read manner and must have a form-factor that is convenient to carry and manipulate.

Most current styles of route maps fail these requirements. A common approach to route mapping is to highlight a route on a standard road map that uses a constant scale factor and depicts all the roads within a region. This style is used by current computer-based route map rendering systems and, as shown in figure 1(left), the constant scale factor makes it impossible to see short roads and their associated turning points. Strip maps, or triptiks, address the issue of varying scale by breaking the route up onto several maps, each with its own orientation and scale. However, the changing orientation and scale make it difficult to understand the overall layout of the route and how the different maps correspond to one another.

One existing route mapping style, the handdrawn map, manages to display each turning point along the route clearly and simultaneously maintain simplicity and a convenient form factor. This is accomplished by performing three types of generalization on the route: (1) the lengths of roads are distorted, (2) the angles at turning points are altered, and (3) the shapes of the individual roads are simplified. We consider each of these in turn:

Length Generalization: Handdrawn maps often exaggerate the lengths of shorter roads on the route while shortening longer roads to ensure that all the roads and the turning points between them are visible, as shown in figure 1. This distortion allows routes containing roads that vary over several orders of magnitude to fit within a conveniently sized image (i.e. a single small sheet of paper). The distortion is usually performed in a controlled manner so that shorter roads remain perceptually shorter than longer roads, while maintaining the overall shape of the route as much as possible.

Angle Generalization: Mapmakers often alter the angles of

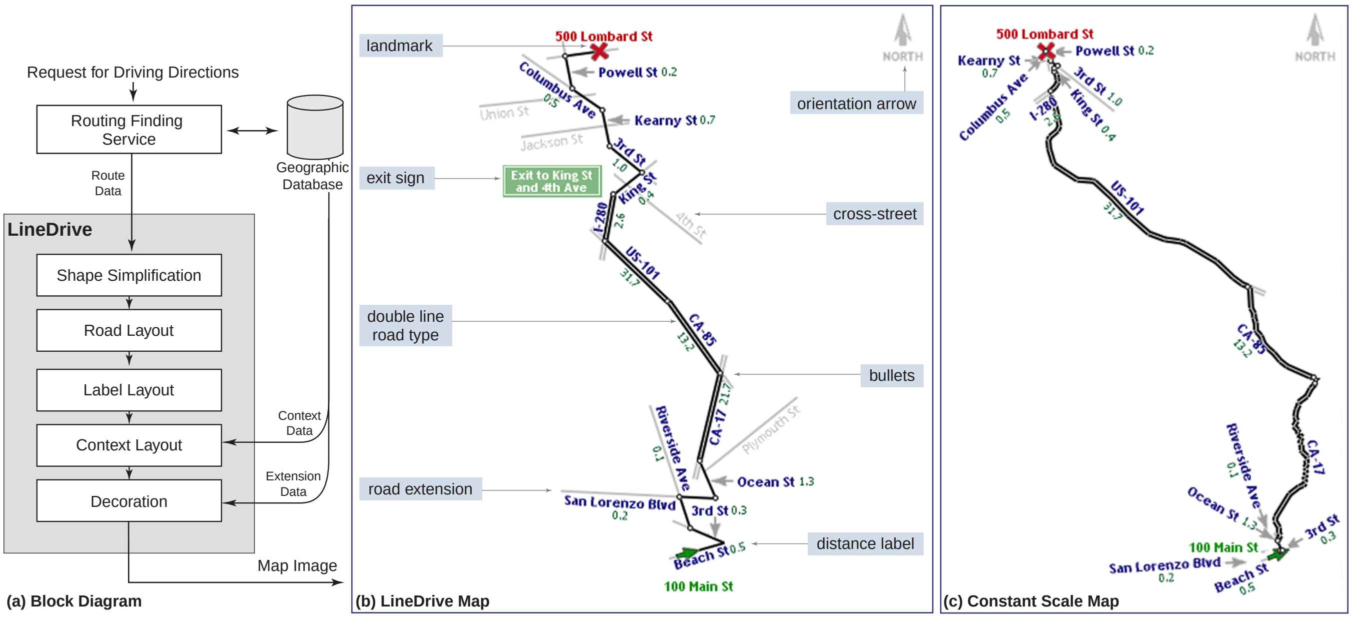


Figure 2: The LineDrive system. (a) Given a route as a sequence of roads, LineDrive designs a route map by processing the route through five consecutive stages. (b) The resulting LineDrive map. (c) The same map rendered without applying the generalization techniques performed by LineDrive. The constant scale factor and retention of detailed road shape make it difficult to identify many of the roads.

turns to improve the clarity of the turning points. Very tight angles are opened up to provide more space for growing shorter roads and labeling roads clearly. Roads are often aligned with the horizontal or vertical axes of the image viewport, to form a cleaner looking, regularized map [26]. Such angular distortions are acceptable because reorienting correctly requires knowing only the turn direction (left or right), not the exact turning angle.

Shape Generalization: Since a navigator does not need to make active decisions when following individual roads, knowing the exact shape of a road is usually not important. Simplifying the road shape removes extraneous information and places more emphasis on the turning points, where decisions need to be made. Roads with simplified shape are perceptually easier to differentiate as separate entities and are also easier to label clearly.

While these generalization techniques can increase the usability of the route map, they can also cause confusion and mislead the navigator if carried to an extreme. By simplifying road shape and distorting road lengths and angles, it is possible to drastically change the topology and overall shape of the route. When these generalizations are performed carefully, however, they can dramatically improve the usability of the map.

3 System

The LineDrive system automatically designs route maps in real-time using the generalization techniques commonly found in hand-drawn maps. The space of all possible route map designs and layouts is extremely large and contains many dimensions. We reduce the dimensionality of this space by performing the map design in five independent stages as shown in figure 2.

All of the geographic data is stored in a database in the standard latitude/longitude geographical coordinate system. The route finding service computes the sequence of roads required to get from a given origin to a given destination and passes this sequence into LineDrive. Each road is represented as a piecewise linear curve described by a sequence of latitude/longitude shape points.

The first stage of LineDrive is shape simplification, which removes extraneous shape detail from the roads, as described in section 3.1. The next three stages, road layout, label layout, and context layout, each deal with automating a layout problem. We use a similar search-based approach in all three stages, described in section 3.2. The details of each layout stage are then presented in sections 3.3 through 3.5. The decoration stage, described in section 3.6, adds elements such as road extensions and an orientation arrow to the map to enhance its overall usability. We conclude our system description in section 3.7 by presenting methods for computing image size based on the aspect-ratio of the route and the size of the output display device. Our system description provides an overview of how we automate the route map design process. Further system implementation details can be found in [1].

3.1 Shape Simplification

LineDrive's shape simplification stage reduces the number of segments in each road while leaving the overall shape of the route intact. Shape simplification not only yields a cleaner looking map but also increases the speed and memory efficiency of the subsequent layout stages of the system.

Techniques for curve smoothing, interpolation, and simplification have been well-studied in a variety of contexts including cartography [22, 8, 2] and computer graphics [10, 12, 5]. We take a standard approach to simplification that ranks the relevance of all the shape points of the curve and then removes all interior shape points that fall below a given threshold. However, our simplification algorithm must not introduce the three undesirable effects shown in the rightmost column of figure 3: false intersections, missing intersections and inconsistent turn directions.

We include three tests during simplification to prevent these problems. To ensure that the simplification process does not introduce false or missing intersections, we initially compute all the *true* intersection points between each pair of roads. Suppose roads r_1 and r_2 initially intersect at point p_1. We add the intersection point p_1 to the set of shape points for both r_1 and r_2 and mark p_1 as *unremovable*. Since the simplification algorithm cannot remove these unremovable intersection points, a missing intersection cannot be generated. Moreover, we only accept the removal of a shape point as long as its removal does not create a *new* intersection point that is not in our original list of true intersection points. This test ensures that the simplification will not introduce any false intersections. Finally, we check for inconsistent turn direction at the turning

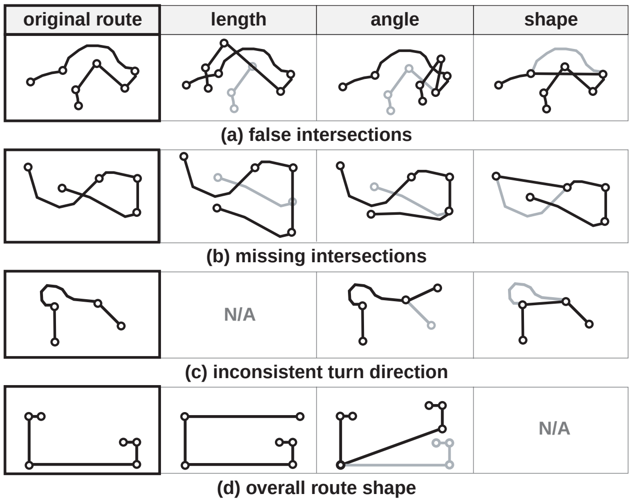


Figure 3: Generalization can cause four types of undesirable effects. Each column shows the route after generalizing the length, angle, or shape of a single road. For comparison, the undistorted route is shown in gray. (a) The original route does not contain an intersection but generalization causes false intersections. (b) The original route contains an intersection (this usually occurs when one road passes over another road) but after generalization the intersection is missing. (c) Generalization causes a right turn to appear as left turn or vice versa. Note that distorting road length cannot generate an inconsistent turn direction. (d) Generalization causes drastic changes in overall route shape. This is reflected in substantial changes in the length and direction of the vector between the route endpoints. Our shape simplification algorithm cannot cause drastic changes to the overall route shape because it only removes shape points from each road and never removes the first or last shape point of a road.

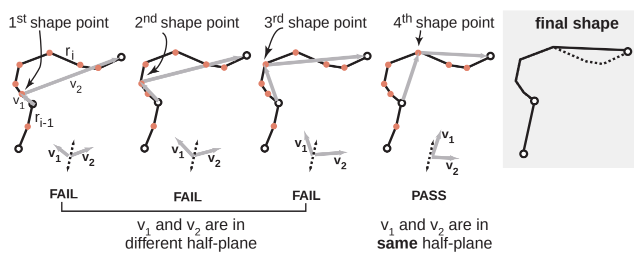


Figure 4: Turn direction consistency check between roads r_i and r_{i-1}. We step through the shape points of r_i, forming two vectors: v_1, between the endpoint of r_{i-1} and the current shape point, and v_2, between the current shape point and the endpoint of r_i. If v_1 and v_2 are not in the same half-plane with respect to the coordinate system oriented along the last segment of the r_{i-1}, we mark the current shape point as unremovable. The test continues until a shape point is not marked as unremovable.

points between each road r_i and the roads r_{i-1} and r_{i+1} adjacent to it. We describe the test between r_i and r_{i-1} in figure 4. The test between r_i and r_{i+1} is similar.

For most roads we are very aggressive about simplification. We remove all shape points that are not marked as unremovable by the previous tests, so most roads are simplified to a single line segment. For some roads, such as highway on- and off-ramps, depicting more realistic shape can be useful. Knowing whether a ramp curves around tightly to form a cloverleaf or only bends slightly can make it easier to enter or exit the highway. Thus, when simplifying ramps we use a more conservative simplification relevance metric to retain more shape [2].

Some long routes between distant cities require traversing many highways. Depicting all the short ramps between the highways can clutter the map with unnecessary detail. Therefore, if the route is longer than a given threshold we remove all ramps from the map that can be removed without creating a false or missing intersection or inconsistent turn direction. Note that all the ramps have been removed from the map in figure 2(b).

3.2 Formulating Layout As Search

In almost any layout problem there are constraints on how the information can be laid out, and there are a set of criteria that can be used to evaluate the quality of the layout. Many such layout problems can be posed as a search for an optimal layout over a space of possible layouts. To frame the layout problem as a search we need to define an initial layout and two functions: a *score* function that assesses the quality of a layout based on the evaluation criteria, and a *perturb* function that manipulates a given layout to produce a new layout within the search space. We can then perform simulated annealing [20] to search for a layout that minimizes the score, as shown in the following pseudo-code:

procedure SimAnneal()
InitializeLayout()
$E \leftarrow$ ScoreLayout()
while(! termination condition)
  PerturbLayout()
  $newE \leftarrow$ ScoreLayout()
  if (($newE > E$) and (Random() $< (1.0 - e^{-\Delta E/T})$))
    RevertLayout()
  else
    $E \leftarrow newE$
  Decrease(T)

The simulated annealing algorithm accepts all good moves within the search space and, with a probability that is an exponential function of a temperature T, accepts some bad moves as well. As the algorithm progresses, T is annealed (or decreased), resulting in a decreasing probability of accepting bad moves. Accepting bad moves in this manner allows the algorithm to escape local minima in the score function.

The difficult aspects of characterizing the layout problem as a search are designing an efficient score function that captures all of the desirable features of the optimal layout and defining a perturb function that covers a significant portion of the search space. As we discuss the different layout stages of LineDrive, we will focus on explaining these aspects of our algorithm design.

3.3 Road Layout

The goal of road layout is to determine a length and an orientation for each road such that all roads are visible and the entire map image fits within a pre-specified image size. Moreover, the layout must avoid the problems shown in the second and third columns of figure 3 and preserve the topology and overall shape of the route.

To generate an initial layout for the search, we first build an axis-aligned bounding box for the original route and compute a single factor to scale the entire route to fit within the given image viewport. Next, we grow all roads that are shorter than a predefined minimum pixel length, L_{min}, to be L_{min} pixels long. Since we initially scaled all the roads to fit exactly within the bounds of the image, growing the short roads may extend the map outside the viewport. We finish the initial layout phase by again scaling the entire route to fit within the image viewport.

To perturb a road layout during the search, we randomly choose a road r_i and either scale its length $l(r_i)$ by a random factor between $0.8x$ and $1.2x$, or change its orientation by a random reorientation angle between ± 5 degrees. The ± 5 degree bound on road reorientation is decreased as necessary to ensure that an inconsistent turn direction is not introduced. After modifying a road, we rescale the route to fit within the image viewport. By disallowing perturbations that cause inconsistent turn directions and forcing the route to always fit the viewport, we limit our search space to maps that meet our turn direction and image size constraints.

All other constraints on road layout are enforced through the scoring function which examines three aspects of the road layout: road length and orientation, intersections between roads, and the shape of the overall route. We discuss the computation of these component scores in the next three subsections. After the road layout search is complete, we fine-tune the road orientations as described in section 3.3.4.

3.3.1 Road Length and Orientation

Each road r_i is scored on two length-based criteria. First, we penalize any road that is shorter than L_{min} using the following formula:

$$score(r_i) = ((l(r_i) - L_{min})/L_{min})^2 * W_{small} \quad (1)$$

where W_{small} is a predefined constant used to control the weight of the score in relation to the other scoring criteria[1]. The function is quadratic rather than linear, so roads that are much shorter than L_{min} are given a higher penalty than roads that are just a little shorter than L_{min}. Recall that simulated annealing decides whether to accept the current layout based on the difference between the current score and the previous score. By using a quadratic function, we increase the probability of accepting perturbations which grow the shortest roads because such perturbations yield the largest change in score per pixel length. If we used a linear function, growing any road by an amount x would yield the same change in score with no preference for growing the shortest roads.

The second length-based scoring criterion considers the relative ordering of the roads by length. We add a constant penalty for each pair of roads whose length ordering has shuffled between the original map and the current layout. The purpose of this score is to encourage layouts in which the longer roads *appear* longer than shorter roads in the final map. Therefore, we only consider roads as being shuffled when the difference in their lengths is greater than a predefined perceptual threshold (usually 5-10 pixels).

We also penalize each road by a score proportional to the difference between its current orientation α_{curr} and its original orientation α_{orig} using the following formula:

$$score(r_i) = |\alpha_{curr} - \alpha_{orig}| * W_{orient} \quad (2)$$

Since this score is minimized when the current orientation is closest to its original orientation, we only introduce substantial changes to road orientation if the change helps minimize some other component of the road layout score. For example, a substantial change in orientation may be introduced to resolve a false intersection.

3.3.2 Intersections

Both missing and false intersections can be extremely misleading, so we severely penalize any proposed layout containing these problems. We first describe how simple missing and false intersections are resolved independently and then describe how scoring must change when a layout contains both missing and false intersections.

Simple Missing Intersections: There are two forms of missing intersections. A true *missing* intersection occurs when two roads should intersect, such as when a highway ramp loops over or under the highway, but don't. A *misplaced* intersection occurs when two roads should intersect and do, but at the wrong point. As shown in figure 5, in both cases we compute a score that is proportional to the Euclidean distance between the proper intersection point on each road. However, since it is more important for the proper pair of roads to intersect than it is for the point of intersection to be placed exactly, we set the scoring weight for a misplaced intersection to be much lower than for a missing intersection.

Simple False Intersections: False intersections occur when the path incorrectly folds back on itself, forming a loop or knot. One way to remove an individual knot is to move the route endpoint closest to the intersection (measured in pixels along the route) towards the intersection point. Figure 6(a)-(c) illustrates several false intersection scenarios, showing for each intersection point which direction the closest endpoint must move to remove the knot.

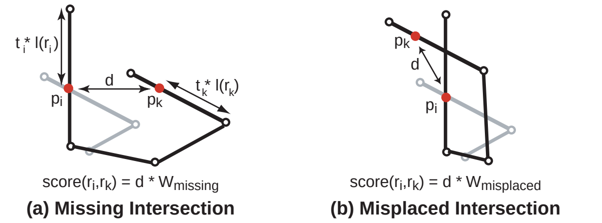


Figure 5: Scoring missing and misplaced intersections. In both cases the score is proportional to d, the Euclidean distance between the two points p_i and p_k that should intersect (marked in red). Initially for each pair of intersecting roads r_i and r_k we compute the parametric values t_i and t_k of the intersection point. Multiplying these parameters by the current lengths of the roads $l(r_i)$ and $l(r_k)$ gives us the current position of p_i and p_k. For comparison, the original route is shown in gray.

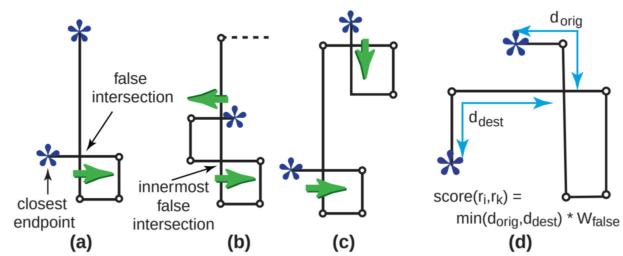


Figure 6: Handling false intersections. (a),(b),(c) The direction the route endpoints should move to independently resolve each false intersection is indicated by the large green arrows. (b) The two false intersections pull the endpoint in opposite directions. This is addressed by counting only the innermost false intersection score. (c) The innermost false intersection is scored for each endpoint independently, so in this case both false intersections are included in the final score. (d) The score for a simple false intersection is proportional to the distance to the closest endpoint of the route as measured in pixels *along the route.*

For each false intersection we compute a score proportional to the distance in pixels along the route to the nearest endpoint, as shown in figure 6(d). This approach is conceptually equivalent to building a scoring hill along the route that guides the closest endpoint towards the intersection point, thereby unravelling the knot.

When a route contains multiple false intersections, the false intersection scores may conflict and push the endpoint in opposite directions, as shown in figure 6(b). We address this problem by counting only the score for the innermost false intersection (working inwards from the endpoint to the center of the route). By penalizing the layout for only the innermost false intersection, we guide the endpoint towards the desired direction and eventually resolve both false intersections.

False Intersections and Missing Intersections: In most cases when false and missing intersections occur in the same map, the scores interact properly to resolve both problems. There is one exceptional situation that occurs when the loop formed by a false intersection contains a missing intersection. As shown in figure 7, one score may push in one direction and the other score in the other direction, resulting in a stalemate in which neither problem can be resolved. In both of these cases there is supposed to be an intersection; it is just occurring between the wrong roads. We resolve the situation with an additional rule: if either point of a missing intersection is inside the loop formed by a false intersection, we add a constant penalty for the false intersection, rather than a hill-based score. Using a constant false intersection score allows the missing intersection score to guide the intersection to the desired location.

Extended Intersections: While the false and missing intersec-

[1] Each of our component scores uses a similar weighting constant.

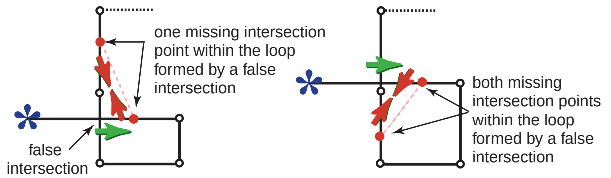


Figure 7: Interactions between false and missing intersections. In both cases, the false and missing intersection scores push points on the route in conflicting directions, as indicated by the arrows. To resolve the conflict, we add a constant penalty for the false intersection and allow the missing intersection score to pull the intersection to the desired location.

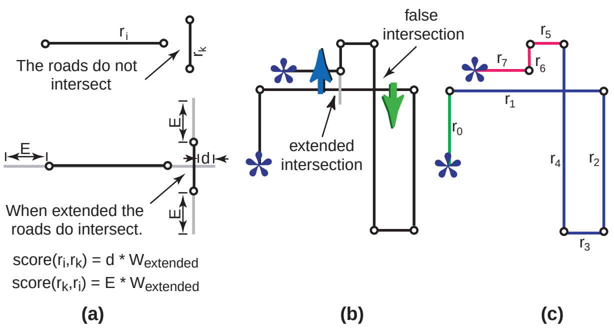


Figure 8: (a) Scoring extended intersections. (b) The extended intersection and false intersection scores conflict and push the layout in opposite directions. (c) All roads between a route endpoint and a false intersection or between a pair of false intersections are considered to be in the same *false intersection interval*. In this case, there are three intervals $[r_0]$, $[r_1, r_2, r_3, r_4]$, and $[r_5, r_6, r_7]$. We resolve the conflict by only counting extended scores between roads in the same false intersection interval. Since r_1 and r_6 are in different intervals, their extended intersection score is not counted.

tions scores are essential for maintaining the overall topology of the route, they do not consider the spacing between roads. It is possible for the perturb function to generate road layouts in which non-intersecting roads pass so close to one another that they incorrectly appear to touch. We identify such layouts by checking for *extended intersections* between each pair of roads. We extend the endpoints of each road by a fixed pixel length E and then check if the resulting roads intersect.

Extended intersections are scored as shown in figure 8(a). If the intersection occurs on the extended portion of the road as for r_i, the score is proportional to the distance between the intersection point and the extended endpoint of the road. If the intersection occurs within the main extent of the road as for r_k, the score is set to the largest possible penalty for intersection with the extended portion of the road. As shown in figure 8(b), it is possible for an extended intersection score to conflict with a false intersection score. To reduce such conflicts, we include extended intersection scores only when the extended intersection occurs between two roads in the same *false intersection interval*, as shown in figure 8(c).

3.3.3 Route Shape

The final road layout score considers the overall shape of the route. As shown in figure 3, perturbing the lengths and angles of each road can drastically alter the overall shape. It is possible for a destination that should appear to the west of the origin to end up appearing to the east of the origin, and the origin can sometimes appear much closer to the destination than it actually is.

To reduce such problems, we compute two road layout scores based on the vector between the origin and destination of the route. The endpoint direction score penalizes layouts that alter the direction of this vector and is proportional to the difference in angle between this vector in the original map and in the current map. The endpoint distance score penalizes layouts in which distance between the origin and destination is smaller than a minimum length based on the original distance between them.

3.3.4 Fine-Tuning Road Orientation

Once the search phase of road layout is complete, we snap each shallow angle road in the final layout to the nearest horizontal or vertical axis. Roads that form shallow angles (i.e. < 15 degrees) with the image plane horizontal or vertical axes tend to increase the visual complexity of the map. Furthermore, such roads can be difficult to antialias, especially on personal digital assistant (PDA) displays with limited color support. Note that we only reorient a road if doing so does not introduce an inconsistent turn direction or a false, missing, or extended intersection.

3.4 Label Layout

For the route map to be usable, each road on the map must be labeled with its name. Similarly, the origin and destination of the route should be labeled with their addresses. Each label is added to the map to communicate a piece of information (e.g. a road's name) through a combination of text and images. The label's placement and style further communicate which map object (e.g. road, landmark, etc.) it is labeling. We refer to this object as the label target.

There are many different ways to label a given target object. A typical method for labeling roads is to simply write the name directly above or below the road. This approach uses proximity to associate the label with its target road. Another style is to put the text near the road and then add an arrow pointing to the road to form the association between the name and its target. Figure 9 shows several styles that might be used to label different objects. As shown in figure 10, a *labeling style* is comprised of three components:

- **Graphic Elements:** A set of text and image elements. The primary graphic element is usually a name, and secondary graphic elements can include distance to travel, arrows, highway shields, etc.
- **Arrangement:** The arrangement of the secondary graphic elements relative to the primary element. For example, the arrow-left-of-name labeling style puts the arrow graphic to the left of the primary name graphic.
- **Placement Constraints:** Each constraint is a region in the map image defining a set of valid positions and orientations for the center of the primary graphic.

To place a given label in the map, we must choose both a labeling style and a label location from within one of the placement constraint regions for that style. Therefore, our label layout search space is defined by the set of possible labeling styles and the placement constraints for each style, for every label in the map.

In the first phase of label layout, we create a list of possible labeling styles for each target object by considering factors such as the size, shape, and type of the target (e.g. highway, residential road, or landmark) and the length of the label name (e.g. if the name is long we might create a word-wrapping style). Each style is also given a rank based on its desirability. For example, for roads, the *along-road* style is preferable to to the *arrow-left-of-name* style.

We create an initial label layout by placing each label at the most central position within its highest ranked labeling style. We then deterministically fix as many labels as possible. We check if each label in its initial position could ever conflict with the placement of any other label by intersecting each label in its initial position with all potential positions for every other label. The potential positions are determined from the placement constraints defined for each labeling style. If no conflict is possible, then the label is fixed in its

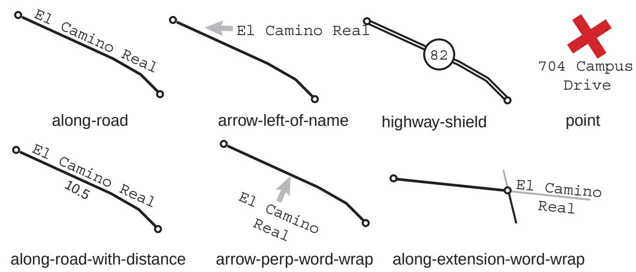


Figure 9: Several different labeling styles that might be used to label roads or landmarks along the route.

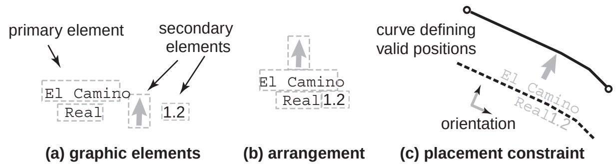


Figure 10: Components of a *labeling style*. (a) a set of graphic elements, (b) an arrangement of those graphic elements relative to a primary graphic, and (c) a constraint specifying the valid positions and orientations for the center of the primary graphic.

initial position and only those labels that are not fixed in this phase are placed during the label layout search.

The perturb function for the label layout search randomly picks a label to alter, randomly selects a labeling style for that label, and then randomly chooses a new location for the label from within one of the style's placement constraints. The label layout scoring function evaluates each label on the following criteria: (1) whether the label intersects or overlaps any other object in the map, (2) the proximity of the label to the center of its target, and (3) the rank of the chosen label style. The score for the complete map labeling is computed as the sum of the scores for each label.

Our general approach to the label layout problem is based on previous work on labeling point and line features in traditional geographic maps. Marks and Shieber [19] have shown that finding optimal label placements is NP-complete and several previous systems have used randomized search to find near-optimal label placements [29, 9]. These systems usually consider only a discrete set of possible locations and a single style for each label. LineDrive extends the search-based approach to handle a continuous range of label locations and a wider variety of potential labeling styles.

3.5 Context Layout

Although context features are secondary information not necessary for communicating the basic structure of route, they can improve the usability of a route map. LineDrive handles two forms of context: (1) linear features that intersect the main route, such as cross-streets, and (2) point landmarks along the route such as buildings and highway exit signs. We use the same basic approach for placing both cross-street and local landmarks. For brevity, we will describe the approach in terms of placing cross-streets[2].

Each cross-street is specified to the layout system by a piecewise linear curve of latitude/longitude points, the name of the cross street and an importance value for the cross-street. If the importance value is not pre-specified, we place highest importance on the last major cross-street just before each turning point. We have found that these streets are helpful as a warning that the turn is approaching. We initially compute the intersection point between every cross-street and the main route and then place the cross-streets at these intersection points. We also create a constraint region around the intersection

[2] Interested readers should consult [1] for the details of landmark layout.

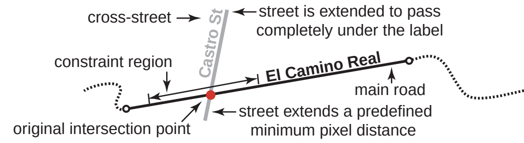


Figure 11: Placing a cross-street. The search considers placing Castro street within the constraint region as close to the original intersection point as possible. Once the cross-street and its label are placed, the cross street is extended to a minimum pixel length on either side of its base road and then further extended to pass under its label.

point which specifies the acceptable range of positions for the intersection point, as shown in figure 11. Cross-street labels are created just like main road labels and initially placed using the same rules.

The perturb function for cross-street layout randomly selects a cross-street and then randomly changes either the position of the intersection point between the cross-street and the main road, the position of the cross-street label, or whether the cross-street is hidden. Once the street is perturbed, we set the length of the cross-street to a predefined minimum extension length. Then, if the label has been placed directly above or below the street, we extend the street to pass completely over or under its label.

We score each cross-street based on four criteria: (1) the distance between the current position of the cross-street intersection point and the true intersection position, (2) the number of other objects the cross-street intersects, (3) the layout score of the cross-street label, which is computed using the same scoring function as for regular road labels, and (4) if the cross-street is hidden, we penalize the layout by an amount proportional to the cross-street importance.

Once the search phase of cross-street layout is complete, we clean up the layout. If the label of a cross-street overlaps any other object on the map, we remove the cross-street from the map. Label-object overlap can make the label difficult to read and obscure important route information. Since cross-streets are secondary features, removing them from the map is preferable to allowing such overlap. We do, however, allow the cross-streets to intersect other map objects. This is acceptable because cross-streets are thin, 1D objects, and are drawn underneath the other map objects in a light gray color so that they do not interfere with the legibility of the other objects. Finally, we clip each cross-street to every other road and cross-street in the route. This ensures that we do not introduce any false cross-street intersections in the maps.

3.6 Decoration

The decoration stage is responsible for adding four types of graphic decorations to the map to enhance its usability. Extensions on the ends of each road accentuate the turning points and help associate the road's label with the road. An orientation arrow shows the overall route orientation with respect to global north and can make it easier for navigators to geographically place the route. Bullets at each turning point show exactly where each turn decision must be made and help differentiate between roads that are headed in the same general direction. Finally, the rendering style for each road is set according to the type of the road.

Before adding extensions, we look up the pair of roads at each turning point in the database to check if they continue beyond the turning point. If a road does extend, we set the length of the extension to a predefined minimum extension length. If during label layout, the center of the road's label was placed directly above or below an extension, we grow the extension so that it passes completely over or under the label. Growing the extension in this manner helps form the proper association between the label and its target road. Finally, we clip the extension to all other roads and cross-streets.

To place the orientation arrow, we search the map image for an empty region large enough to hold the arrow. We accelerate the search by building a fixed resolution occupancy grid over the map image and only searching in empty cells of this grid. The search is

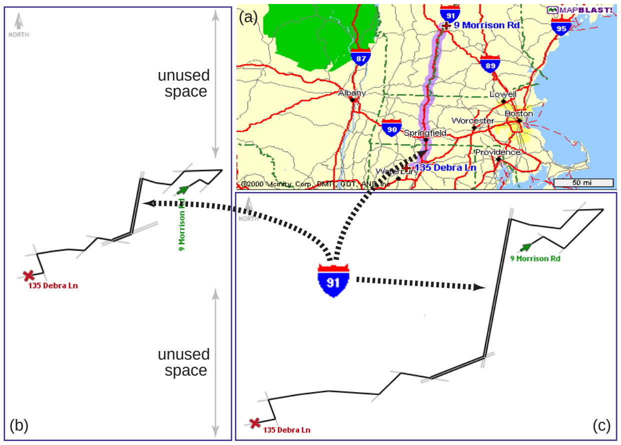


Figure 12: Selecting image size. (a) The route contains one long north-south road (I-91) and many short east-west roads near its origin and destination. (b) If the image size is selected based on the original north-south aspect ratio, the image is given more vertical space than horizontal space. The top and bottom of the image go unused because after growing all the short roads the aspect ratio of the map becomes much wider. (c) Computing the aspect ratio for selecting image size after growing the roads yields a horizontal image size and a more effective use of the space.

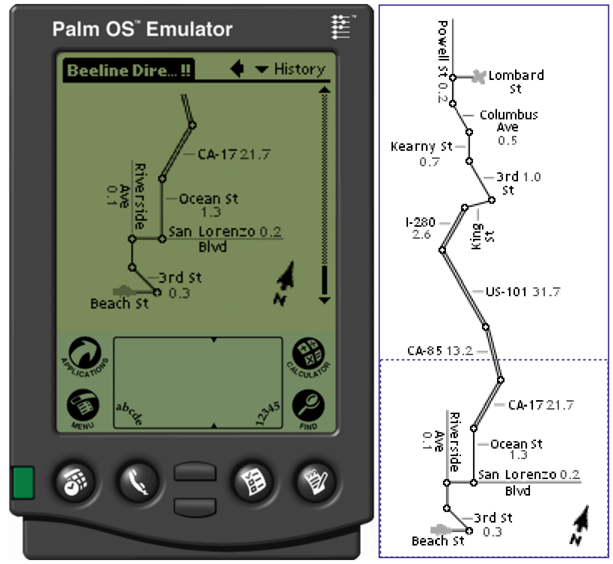


Figure 13: LineDrive map on a PDA. The route is rotated so that it fits the horizontally constrained image size of the PDA. The vertical dimension is unconstrained and users can scroll up to see the remainder of the route. This is the same route shown in figure 2.

ordered to first look for space in the four corners of the image and then search through the remaining image.

Our road database differentiates between three types of roads: limited access highways, highway ramps, and standard residential roads. In the decoration stage we set the rendering style for each road based on its type. Limited access highways are drawn as double lines, while ramps are drawn at half the thickness of the standard roads.

3.7 Image Size Selection

Since LineDrive designs route maps to fit within a given image size, the image size can have a large effect on the layout of the map. Consider a route map created for a predominantly north-south route that is designed to fit a wide aspect ratio viewport. All of the north-south roads would end up squashed while large regions of the image to the left and right of route would remain unused.

A better approach is to choose the viewport size based on the aspect ratio of the route. However, simply using the aspect ratio of the original uniformly scaled route does not always produce the desired result. Suppose, as in figure 12, the original route contains many east-west roads near its origin and destination, with one extremely long north-south road in between. Although the original aspect ratio for the route is north-south, after growing the short roads in our road layout, the aspect ratio of the route changes substantially. To estimate the aspect ratio of our final map before performing road layout, we initially fit all the roads at their original lengths to a large square viewport. We then grow all the short roads to their minimum pixel length and finally compute the aspect ratio of this new map, thus generating a more realistic estimate.

The image size of our maps may be limited by the resolution of the output device. Personal digital assistants (PDAs) usually have small screens, and long routes containing more than a few steps usually will not fit on these screens, even using our layout techniques. One solution is to split such routes into multiple segments, each containing a fixed number of turning points. The main drawback is that this approach requires flipping through multiple maps.

Another solution is to create a larger map image that can be scrolled. However, most PDAs provide good controls for scrolling vertically but not horizontally. In such situations, our image size is constrained only in the horizontal direction. Luckily, most routes have some predominant orientation. We find the predominant orientation by fitting a tight, oriented bounding box [11] to the route after growing all the short roads just as we did for the aspect ratio computation. We then fit the map to our horizontally constrained image by rotating the entire route so that the largest extent of the map is aligned with the vertical axis of the page. This approach provides extra space in the direction the route needs it most. As shown in figure 13, the orientation arrow helps indicate that the map has been rotated.

A common cartographic convention is that the north orientation arrow should align as closely as possible with the vertical axis of the page. Thus, we choose the rotation angle, either clockwise or counter-clockwise, which ensures that north arrow points in the upward semi-circle of directions. The rotation angle is bounded between ± 90.0 degrees and although the north arrow may not be aligned with the vertical axis of the page after the rotation it usually has a strong component in the vertical direction. Once the map has been verticalized, we can compute a vertical resolution for the image based on the number of steps in the route. We have empirically found that providing a vertical resolution of 200 pixels for maps with less than 10 steps, and adding 10 pixels for each step thereafter, works well.

4 Results

Examples of several route maps generated using LineDrive are presented in figure 14. We have tested the performance of the LineDrive system in two ways: (1) by collecting detailed statistics on a test suite of 7727 routes and (2) by providing web access to a beta version of LineDrive in order to receive user feedback.

Our test suite is comprised of 7727 routes queried over one day at `www.mapblast.com`. The median route distance for the test suite is 52.5 miles and the median number of turning points is 13. We ran each route through the system twice, first generating a webpage size image at a fixed resolution of 600 x 400 and then generating a PDA size image with a fixed horizontal resolution of 160 and a variable vertical resolution. The running time is largely dependent on the number of objects (i.e. roads, labels, etc.) that must be placed in the map. The median run time for a single map on an 800 MHz Pentium III was 0.7 seconds for the first run and 0.8 seconds for the second run. Although the vast majority of maps are clustered around these median times, a few outliers containing over 100 roads took about 13 seconds to generate for the webpage size.

A small percentage of the LineDrive maps generated from the

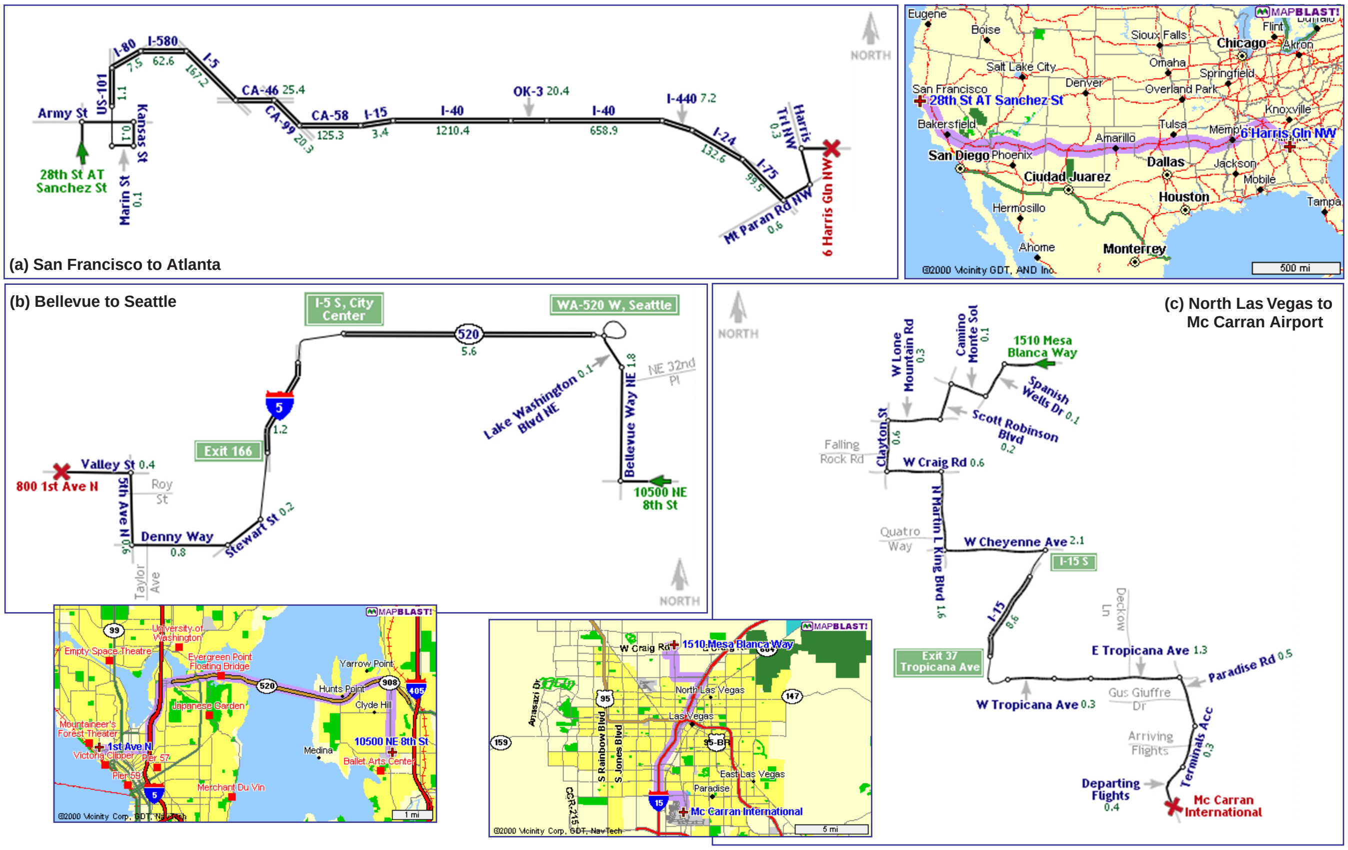


Figure 14: Examples of LineDrive maps with thumbnails of standard computer-generated maps for the same routes. (a) Non-uniform scaling allows all roads to be visible in this cross-country route. Since the ramp between Marin St. and US-101 intersects Army Street (actually passes above Army) it is not dropped from the map and proper intersection topology is maintained. (b) All ramps are maintained in this relatively short route from Bellevue to Seattle. Road shape is retained at both ends of I-5 in order to maintain a consistent turn angle with the adjacent ramps. The exit signs provide important context information for entering and exiting the highways. The highways are labeled using the *highway-shield* labeling style which helps differentiate the interstate, state and local highways from residential roads. (c) Cross-streets provide context and aid navigation in this route from North Las Vegas to McCarran Airport. The sketchy rendering style in this map is a subtle cue that the map is not drawn to scale.

test suite of routes contained layout problems such as topological errors or label-label overlap. In many cases, these problems were unavoidable because it is not always possible to make all roads large enough to be visible and simultaneously maintain the topology of the route. In a few cases, the problems could have been avoided but the randomized search did not converge to a near-optimal layout. The frequency of various layout problems for the 7727 route test suite are summarized in table 1.

The most significant problems that can arise in road layout are (1) that some roads may not be made large enough to be visible and clearly labeled or (2) that false or missing intersections may be introduced during the layout. Short roads, defined as less than 10 pixels in length, occurred in 5.3% of the webpage maps and 5.6% of the PDA maps. In most cases, the short roads could not be made longer either because there were a large number of roads all heading in the same direction or because lengthening the roads would have introduced a false or missing intersection. Although the PDA is horizontally constrained, the increase in the number of maps containing short roads is small because verticalization of these maps provides space for the short roads to grow. False and missing intersections occurred much less frequently than short roads and in all cases, avoiding the false or missing intersection would have required shrinking one or more roads to be extremely small.

The main problems that can occur in label layout are (1) that a label will be placed overlapping another label, or (2) that a label may be placed overlapping a road or landmark. Less than 0.5% of webpage sized maps contained overlapping labels, while 3.7% of PDA sized maps contained label-label overlap. This increase is due to the fact that long labels are especially difficult to place without overlap on the horizontally constrained PDA. Although label-road overlap occurs in a significantly larger number of maps, such intersections are much less detrimental to the overall usability of the map than label-label overlap.

The beta version of LineDrive was available to the public from October, 2000 until March, 2001 and served over 150,000 maps. Over 2200 users voluntarily filled out a feedback form describing their impressions of the LineDrive maps. While the group of respondents was self-selected, it is unclear whether any resulting bias would be positive or negative. Despite the potential bias, we believe that the feedback provides valuable insight into users' reactions to the maps. As shown in table 2, the general response to the LineDrive maps was overwhelmingly positive. Less than one percent of respondents said they would rather use the standard computer-generated maps than the LineDrive maps.

Nearly half of the respondents said they would like to use LineDrive maps in conjunction with standard maps. One difficulty with using LineDrive maps alone is that they provide little detail outside of the main route. If the navigator accidently strays from the route, it can be difficult to find a way back onto it. This can be especially problematic near the destination of the route where the navigator is less likely to be familiar with the area and may need to stray from the route in order to find parking. We address these problems on the website by providing a standard computer-generated map of the region near the destination of the route along with the LineDrive map.

Long distance trips often require more context than LineDrive

Performance Statistics		(7727 routes)		
	Web		PDA	
Median Time		0.7s		0.8s
Short Roads ($<$ 10 pixels)	415	5.4%	430	5.6%
False Intersections	25	0.3%	23	0.3%
Missing Intersections	15	0.2%	14	0.2%
Label-Label Overlaps	37	0.5%	289	3.7%
Label-Road Intersections	901	11.7%	2096	27.1%

Table 1: Performance statistics for a test suite of 7727 routes with a median of 13 turning points per route and a median distance of 52.5 miles. Every row except for median time indicates the number of maps containing at least one instance of the problem. For example, the short roads row presents the number of maps containing at least one road less than 10 pixels long.

User Feedback		(2242 responses)
Would you use LineDrive maps in the future?		
1246	55.6%	Yes, I would use them instead of standard driving directions.
976	43.5%	Yes, I would use them along with standard driving directions.
20	0.9%	No thanks, I'll stick with standard driving directions.
How would you rate this feature?		
1787	79.7%	It's a blast.
253	11.3%	Just fine.
202	9.0%	Needs some work ...

Table 2: User feedback. The beta version of LineDrive has been accessed over 150,000 times and we have received 2242 responses to the system.

maps provide. While the cross-country map in figure 14(a) is a good stress-test showing that LineDrive can produce readable maps for routes containing many steps at vastly different scales, it is probably not the ideal map during such a long trip. Most navigators taking this trip would require a road atlas showing detailed local context along the way. LineDrive maps are designed for relatively short trips (i.e. under 100 miles) within a familiar region. Our experience is that most car-based trips fall within this range and the majority of people who use web-based mapping services generate directions to locations within their own greater metropolitan area.

About 9% of the respondents said the LineDrive system needs some work. However, most concerns were not with the LineDrive map, but instead with the particular route chosen by the route finding service. The beta version of LineDrive did not support cross-streets and local landmarks and the most common feature requests applicable to the maps were for the addition of cross-streets and exit signs. Based on the results of the beta test, LineDrive became the default map style for driving directions at `www.mapblast.com` in March 2001. This version supports cross-streets.

We have experimented with rendering LineDrive maps using a stroke-based, pen-and-ink style [18]. As shown in figure 14(c), the variations in the lines makes the map look more like a sketch than a precise computer-generated image. Strothotte et al. [24] have shown that rendering style can influence how people interpret architectural drawings, and we believe a similar principle applies to route maps. The sketchy rendering style is a subtle cue that the map is not drawn to scale.

5 Conclusions and Future Work

In this paper we have described a set of generalization techniques based on detailed study of the distortions made in handdrawn maps and designed to improve route map usability. We have also presented LineDrive, an automatic system for designing and rendering route maps that uses these techniques to ensure that all information required to follow a route is communicated clearly and concisely.

There are several directions for future research. We are currently exploring the use of insets as an approach for depicting route detail at turning points. The algorithm must automatically select the set of roads that should appear in each inset and the placement of the inset in the overall map.

Area landmarks, such as cities, and bodies of water, can make it easier for navigators to orient the route with respect to local geography. However, placing such landmarks in our maps can be difficult. In order to appear in their correct location with respect to the roads on the route, the size and shape of the area landmarks may need to be distorted. We are considering an approach that uses feature-based morphing [3] to incorporate such landmarks.

Acknowledgments: We are indebted to Christian Chabot for his invaluable help in designing, and championing LineDrive. Thanks to Vicinity corporation for allowing us to integrate LineDrive with their products and their help in extending LineDrive functionality. We are extremely grateful to Pat Hanrahan for giving us the freedom to pursue this line of research.

References

[1] M. Agrawala. *Visualizing Route Maps*. PhD thesis, Stanford University, 2001.

[2] T. Barkowsky, L. J. Latecki, and K. Richter. Schematizing maps: Simplification of geographic shape by discrete curve evolution. In C. Habel C. Freska, W. Brauer and K.F. Wender, editors, *Spatial Cognition II*, pages 41–53. Springer-Verlag, 2000.

[3] T. Beier and S. Neely. Feature-based image metamorphosis. *Computer Graphics (Proceedings of SIGGRAPH 92)*, 26(2):35–42, July 1992.

[4] B. P. Buttenfield and R. B. McMaster, editors. *Map Generalization: Making rules for knowledge representation*. Longman Scientific, 1991.

[5] M. de Berg, M. van Krevald, M. Overmars, and O. Schwarzkopf, editors. *Computational Geometry: Algorithms and Applications*. Springer, 1997.

[6] M. Denis. The description of routes: A cognitive approach to the production of spatial discourse. *Cahiers de Psychologie Cognitive*, 16(4):409–458, 1997.

[7] M. Denis, F. Pazzaglia, C. Cornoldi, and L. Bertolo. Spatial discourse and navigation: An analysis of route directions in the city of Venice. *Applied Cognitive Psychology*, 13(2):145–174, 1999.

[8] D. H. Douglas and T. K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *The Canadian Cartographer*, 10(2):112–122, 1973.

[9] S. Edmondson, J. Christensen, J. Marks, and S. Shieber. A general cartographic labeling algorithm. *Cartographica*, 33(4):12–23, 1997.

[10] G. Farin. *Curves and Surfaces for Computer-Aided Geometric Design*. Academic Press Ltd., 1988.

[11] S. Gottschalk, M. Lin, and D. Manocha. OBB-tree: A hierarchical structure for rapid interference detection. *Proceedings of SIGGRAPH 96*, pages 171–180, August 1996.

[12] J. Hershberger and J. Snoeyink. Speeding up the Douglas-Peucker line-simplification algorithm. In *5th Intl. Symp. on Spatial Data Handling*, pages 134–143, 1992.

[13] N. Holmes. *The Best in Diagrammatic Graphics*. Quarto Publishing, 1993.

[14] E. Imhof. *Cartographic Relief Presentation*. Berlin: de Gruyter, 1982.

[15] V. L. Interrante. Illustrating surface shape in volume data via principal direction-driven 3d line integral convolution. *Proceedings of SIGGRAPH 97*, pages 109–116, August 1997.

[16] A. M. MacEachren. *How Maps Work*. The Guilford Press, 1995.

[17] J. Mackinlay. Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics*, 5(2):110–141, 1986.

[18] L. Markosian, M. A. Kowalski, S. J. Trychin, L. D. Bourdev, D. Goldstein, and J. F. Hughes. Real-time nonphotorealistic rendering. *Proceedings of SIGGRAPH 97*, pages 415–420, August 1997.

[19] J. Marks and S. Shieber. The computational complexity of cartographic label placement. Technical Report ITR-05-91, Center for Research in Computing Techniology, Harvard University, March 1991.

[20] Z. Michalewicz and D. B. Fogel, editors. *How to Solve It: Modern Hueristics*. Springer, 2000.

[21] M. Monmonier. *Mapping It Out*. The University of Chicago Press, 1995.

[22] U. Ramer. An iterative apprach for polygonal approximation of planar closed curves. *Computer Graphics and Image Processing*, 1:244–256, 1972.

[23] D. D. Seligmann and S. Feiner. Automated generation of intent-based 3D illustrations. *Computer Graphics (Proceedings of SIGGRAPH 91)*, 25(4):123–132, July 1991.

[24] T. Strothotte, B. Preim, A. Raab, J. Schumann, and D. R. Forsey. How to render frames and influence people. *Computer Graphics Forum*, 13(3):455–466, 1994.

[25] E. Tufte. *Envisioning Information*. Conneticut: Graphics Press, 1990.

[26] B. Tversky. Distortions in memory for maps. *Cognitive Psychology*, 13(3):407–433, 1981.

[27] B. Tversky. Distortions in cognitive maps. *Geoforum*, 23(2):131–138, 1992.

[28] B. Tversky and P. Lee. Pictorial and verbal tools for conveying routes. In C. Freska and D. M. Mark, editors, *COSIT*, pages 51–64, 1999.

[29] S. Zoraster. Practical results using simulated annealing for point feature label placement. *Cartography and GIS*, 24(4):228–238, 1997.

Automating the Design of Route Maps

Maneesh Agrawala
Stanford University

April 12, 2002

The Design of Route Maps

Understanding how people think about and communicate routes can provide great insight into what information should be emphasized in a computer-generated route map. In order to design a better route map, we begin with a brief summary of the history of route maps. We then analyze the tasks involved in building a mental representation of an environment and following a route – commonly referred to as *wayfinding*. From this analysis we identify the essential information a route map must communicate to support wayfinding. Next we analyze how several of the most common styles of route mapping often fail to emphasize the essential route information. Finally we describe how we use specific generalization techniques within LineDrive, including distortion and abstraction, to present the essential route information in a clear, concise, and convenient form.

1.1 A Brief History of Route Maps

The history of route maps begins with the earliest forms of human generated graphics. In this section we present a brief overview of the variety of guises in which route maps have appeared over their long history. Our overview is not a comprehensive review but rather presents examples that have provided inspiration for our work. Bell's [Bel95] study of strip maps is an excellent reference that provides a more complete history.

Archaeologists and cartographers studying the artifacts of ancient cultures have posited that one of the purposes of cave painting, clay markings, stone carvings and stick assemblies was to communicate routes [MB83, Thr96]. Early examples of strip maps, dating from about 2000 B.C., have been found in ancient Egyptian tombs. The Egyptians painted a basic form of strip maps on the bottoms of coffins depicting routes for the dead to follow to the afterlife [Bel95, Thr72]. The Romans created route itineraries that consisted of lists of stops along the route, such as villages, towns and cities, and the distances between them. These later evolved into graphical stripmaps [MJ87]. Today, subway routes are often depicted in the same fashion as the Roman itineraries.

During the 17th and 18th centuries strip maps were used throughout Britain. John Ogilby's famous road atlas, *Britannia* [Ogi89], published in 1675, took this form. Bell [Bel95] argues that strip maps were popular at the time because there were very few routes between destinations. Tufte [Tuf90] presents examples of strip maps created in London at the end of 18th century that are very similar to their modern day equivalents, the American Automobile Association triptik. Tufte also points out that while the Japanese strip maps from this era were similar to their European counterparts, the Japanese maps use a single, long, thin, sheet of paper to show the entire route, rather than splitting the map across multiple sheets.

In the early 20th century, with the widespread use of the automobile, AAA's triptik gained mass popularity in the United States. The most important property of these maps is that they are customized to emphasize the particular route chosen by the navigator. The complexity of the overall road network is hidden, thereby making the map easier to follow. An example of a triptik is shown in figure 1.2.

In 1931 Harry Beck designed an extremely influential map for the London's subway, the London Underground. His map used a variety generalization techniques to emphasize the routes. Beck locally distorted the scale and shape of the subway routes while preserving the overall topology of the route network. He color-coded the routes and used intuitive icons to represent stations and crossover points. The resultant map is simple, clean and easy-to-read. It has heavily influenced map design in general and subway map design in particular. The design of almost all current subway maps can be traced back to Beck's London Underground map.

Route maps are commonly found in large-scale tourist destinations such as museums, amusement parks and monuments. Tufte [Tuf90] presents a map from Japan's Ise Shrine that extends the generalization techniques used by Beck in the London Underground map to include multiple perspectives. The Ise Shrine map contains scale distortions and employs two distinct perspective systems, an oblique bird's eye view for the main part of the image showing the pedestrian route and an orthographic view on the left side of the image showing train routes and how they connect with the pedestrian routes.

Today, Web-based route mapping services are commonly used to obtain driving directions. These services typically provide directions as an overview map complemented with text directions. In some cases they also provide turn-by-turn focus maps. An example of a Web-based overview/focus map collection appears in figure 1.3. While the text directions generally work well, it is our experience that the accompanying maps are often difficult and frustrating to use. Although such computer-generated route maps are largely negative examples, they provided an initial impetus for the work presented in this dissertation. Our frustration with these maps led to primary goal of this dissertation, which is to develop an automated system for creating more usable route maps.

1.2 Mental Representations of Routes

Cognitive psychologists commonly refer to the mental representation of spatial information as a *cognitive map* [Tol48], while the act of following a route is called *wayfinding* [Gol99]. Building a cognitive map of an environment and wayfinding within it are tightly coupled processes that often occur concurrently. When navigators encounter a new environment they either travel through it (e.g. wayfinding) or view it from a survey perspective (e.g. through a map or some high point in the environment) and then integrate the information into a cognitive map. Once a cognitive map is built navigators rely on it to find their way through the environment. In this section we consider the relationship between cognitive map and wayfinding.

1.2.1 Cognitive Maps

Cognitive maps encode three basic categories of features: points, lines and areas[1]. For navigators, the most important point features in an environment are *landmarks*, the most important linear features are *routes* and the most important area features are *networks*. A commonly accepted theory [Lyn60, Gol99, Cho99] is that cognitive maps are continuously built up from these elementary features in three stages. Navigators first learn individual landmarks as disconnected points in the environment. Navigators then learn routes as a linked sequence of landmarks and finally they learn the network of routes covering a region.

Many studies have shown that cognitive maps are quantitatively inexact and contain many distortions [Tve81, Tve92, But86, Gol99]. Even an expert, such as a cab driver, who is extremely familiar with an environment will generally have trouble estimating Euclidean distances between locations [Pai69, Cha83]. Yet, even though cognitive maps lack quantitative accuracy, they maintain topological consistency and can provide reliable information on topological relations such as inclusion, exclusion, and connectedness. Chown [Cho99] argues that the topological information is enough

[1] These categories can be refined further. For example Golledge [Gol99] adds surfaces to this list, while we group surfaces into the area category. Lynch [Lyn60] splits the points category into landmarks and nodes, and the lines category into paths and edges. He also refers to areas as districts.

Wayfinding techniques	Wayfinding tasks Travel to familiar destinations	Exploratory travel	Travel to novel destinations
Oriented search	X	X	X
Following a marked trail	X	X	X
Habitual locomotion	X		
Path integration	X	X	
Piloting between landmarks	X	X	X

Table 1.1: Wayfinding tasks and techniques. The set of wayfinding tasks that can be achieved with Allen's [All99] set of wayfinding techniques. *(After table 2.1 in Allen [All99])*

to properly navigate the environment and the distortions allow for a more compact representation of the cognitive map.

1.2.2 Wayfinding

Navigators travel through an environment for three reasons: to reach a familiar destination, to explore the environment or to find a novel destination. Allen [All99] describes several of the most common wayfinding techniques that are used to accomplish these wayfinding tasks. Table 1.1 shows each wayfinding technique and the wayfinding tasks it supports. We summarize Allen's descriptions of the techniques.

Oriented search is the simplest wayfinding technique, but is also the least efficient. Starting at an origin the navigator visually (or aurally in the case of the visually impaired) locates a destination and then searches for a path to the destination. Oriented search is most useful for exploratory travel, especially when the navigator has little prior knowledge of an environment.

Following a continuously marked trail is the most reliable wayfinding technique and can be applied to all three wayfinding tasks. While this technique reduces cognitive demands on the navigator, continuously marked trails can be very expensive to create and are therefore extremely rare.

Habitual locomotion occurs when the navigator becomes extremely familiar with a route. The repetitive pattern of the route is imprinted on the navigator and the movements required to complete the trip become somewhat automated. For most commuters the daily route between home and work is habitual and they pay minimum attention to the details of the trip.

Path integration requires navigators to self-monitor their velocity and acceleration to compute the distance and direction to their destinations via dead reckoning. Although humans are not well-equipped to precisely estimate velocity and acceleration path integration can be used to travel to familiar destinations and for some exploratory travel.

Piloting between landmarks requires the the navigator to follow a sequence of landmarks to reach the destination. Each landmark is associated with a path leading to the next landmark and therefore the route is completely defined by the sequence of landmarks along it. Given the sequence of landmarks this wayfinding technique can be used for all three wayfinding tasks[2].

Of these wayfinding techniques only *oriented search* and *piloting between landmarks* apply to all three wayfinding tasks. While oriented search is useful in some situations, piloting between landmarks is far more efficient and it is the most commonly used wayfinding technique. Therefore we focus our work on this approach.

Piloting between landmarks requires the navigator to know the sequence of landmarks from origin to destination. Each landmark is essentially a turning point on the route and wayfinding consists of two alternating activities: following a road until reaching a turning point and then changing orientation to follow another road [Den97]. As described in the previous section the sequence of turning points (or landmarks) is precisely the sequence that is built in the second stage of cognitive mapping. When the sequence of turning points is not stored in the navigator's cognitive map some form of a route description such as text directions or a map, is required in order to determine the turning point sequence.

1.3 Information Conveyed by Route Maps

The research on cognitive mapping and wayfinding has shown that routes are often thought of as a sequence of turns [Tve92, Mac95, All99]. Furthermore, verbal route directions are typically structured as a series of turns from one road to the next with emphasis on communicating turn directions and the names of the roads [DPCB99]. Tversky and Lee [TL99] have shown that hand-drawn maps maintain a similar structure with emphasis on communicating the roads and turn direction at each turning

[2] Our definition of landmarks refers both to physical structures such as building as well as other uniquely marked points along the route. For example intersections between roads are uniquely marked by the names of the roads at the intersection.

point.

It follows then that computer-generated route maps should also emphasize the turning point information. Yet, even though it is possible to follow a route map that only indicates the road names and turn direction at each turning point, additional information can greatly facilitate navigation. The information that can be depicted in route maps falls into three broad classes: turning point information, local context, and overview context. We consider each of these in order of importance for the navigator.

1.3.1 Turning Point Information

A turning point can be defined by a pair of roads (the road entering and the road exiting the turning point) and the turn direction (left or right) between those two roads. Route maps depict this information visually, so navigators can quickly scan the map to find the road they are currently following and look ahead to determine the name of the next road they will turn onto. Once the name of the next road is known, the navigator can search for the corresponding road in the physical world. The turn direction specifies the action navigators must take at the turning point. The turning point information is essential for a route map to be useful. It would be nearly impossible to follow a route without it.

1.3.2 Local Context

Local context consists of information about the route itself as well as the environment immediately surrounding the route. For example, if the map labels each road with the distance to be traveled along that road, navigators can use their odometer to determine how close they are to the next turn. Cross-streets and local landmarks along the route, such as buildings, bridges, rivers, and railroad tracks, can also be used for gauging progress. Navigators can also use this information to verify that they are still following the route and did not miss a turn. The cross-street immediately before a turn is especially useful because it can warn the navigator that the next turn is approaching. Similarly, providing the first cross-street after each turning point can cue navigators that they missed a turn. However, local context is not essential for following the route and is usually included in the map only when it does not interfere with the primary turning point information.

1.3.3 Overview Context

Overview context consists of large scale area landmarks as well as global properties of the route. Depicting large scale landmarks such as cities and bodies of water near the route can make it easier to correlate the map with the physical world. Navigators can use these landmarks to orient the route with respect to the local geography. Moreover, such landmarks can help navigators quickly locate their position in the map. For example, navigators who know they are in San Francisco can quickly narrow in on the streets shown near the city. Similarly, maintaining the overall shape and heading of the route (e.g. north-south vs. east-west) can also make it easier to place the route in the larger geographic context. However, overview context is the least important of the three classes of route map information. Like local context, it is not required for traversing the route.

1.4 Ranking the Information Classes

Although route maps may be used before a trip for planning purposes, they are most commonly used while actually traversing the route. Our ranking of the three classes of route map information reflects this fact. For a navigator, en route to a destination, finding the next turn on the route is often the most important task. In many cases, navigators are also drivers and their attention is divided between many tasks. As a result, they can only take quick glances at the map. Therefore, maps must convey the turning point information in a clear, easy-to-read manner and must have a form-factor that is convenient to carry and manipulate. Hand-drawn maps achieve all of these requirements and emphasize the turning point information by omitting some of the local and most of the overview context information. For these reasons we treat turning point information as essential, while local context is less important and overview context is least important.

Route maps created based on this ranking are not ideal in all situations. In particular, if the navigator strays from the route, context information such as nearby roads is far more important than the original turning point information for the route. Similarly, once navigators reach their destinations they often need context information to find local parking areas. Hand-drawn route maps that emphasize turning point information by omitting context may not be very useful in such situations. A general purpose road map of the area, which emphasizes all regions of the map equally and therefore contains the appropriate context information, would be much more helpful.

However, we are most interested in improving the usability of route maps designed

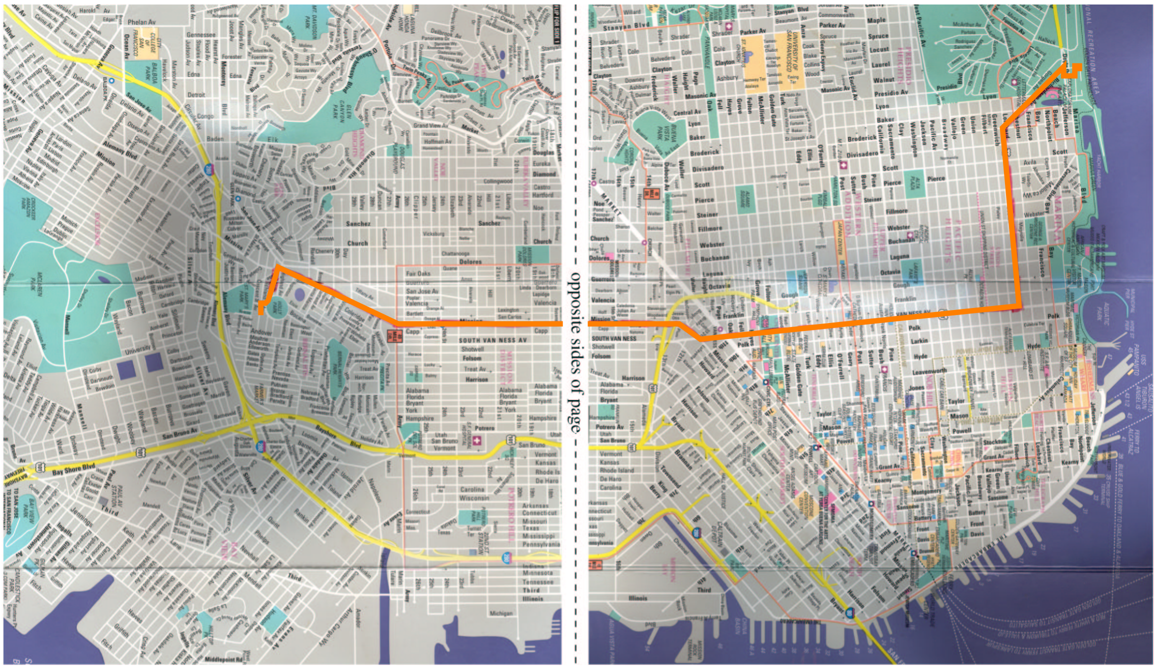


Figure 1.1: A route highlight map. Although the route is highlighted in orange, the map shows the entire city of San Francisco and therefore turns at the beginning and end of the route are barely visible. The extraneous context information and the variety of colors can make it difficult to find the route with a quick glance. The form-factor of the map is inconvenient because the route is on both sides of the map and therefore the navigator must search both sides when looking for the route. *(map courtesy of StreetWise Maps)*

for navigators who are on the main route. The additional context information in a standard road map map is distracting and makes it difficult to find the route being traversed. Although it is can be useful to include some context information in a route map, it should only be included when it does not reduce the clarity of the turning point information.

1.5 Analysis of Current Route Mapping Styles

Most current styles of route maps fail to present the essential turning point information of the route in a clear, easy-to-read manner within a convenient form-factor.

We analyze five of the most common route mapping styles and consider the design choices made in each.

1.5.1 Route Highlight Maps

Route highlight maps simply highlight the route on a general road map of the region. Since the purpose of a road map is to provide an understanding of the entire road system in a region, they typically employ constant scale factors and display extraneous context throughout the map.

An example of this route mapping style appears in figure 1.1. The primary problem with this style is the constant scale factor which makes it impossible to see short roads and their associated turning points. Since general road maps are not optimized to show any particular route, a route highlight map will often suffer from both a large scale factor that hides important information and an inconvenient form-factor that is too large to easily manipulate while driving.

The only property differentiating the route from the other roads is the highlighting. Therefore the clarity of the route depends on the highlighting technique. Usually the route is distinctively colored, but because general road maps provide context information over the entire map and often make liberal use of color to encode properties of this extraneous information, it can sometimes be difficult to distinguish the route from the other elements of the map.

1.5.2 Strip maps

Strip maps, or triptiks, are similar to route highlight maps, but are designed to communicate a particular route. As shown in figure 1.2, strip maps address the issue of varying scale by breaking the route up onto multiple pages. Each page is oriented so that the route runs roughly down its center. Although the scale factor is fixed for each page, the scale factor can differ across pages. The differing scale factors allow strip maps to depict more detailed turning point information where needed. However, because the map stretches over many pages and the orientation and scale factor varies from page to page, forming a general understanding of the overall route can be difficult.

1.5.3 Overview/Focus Map Collections

Overview/Focus map collections consist of a set of maps rendered at different scales that present a single route. This the most common route mapping style avail-

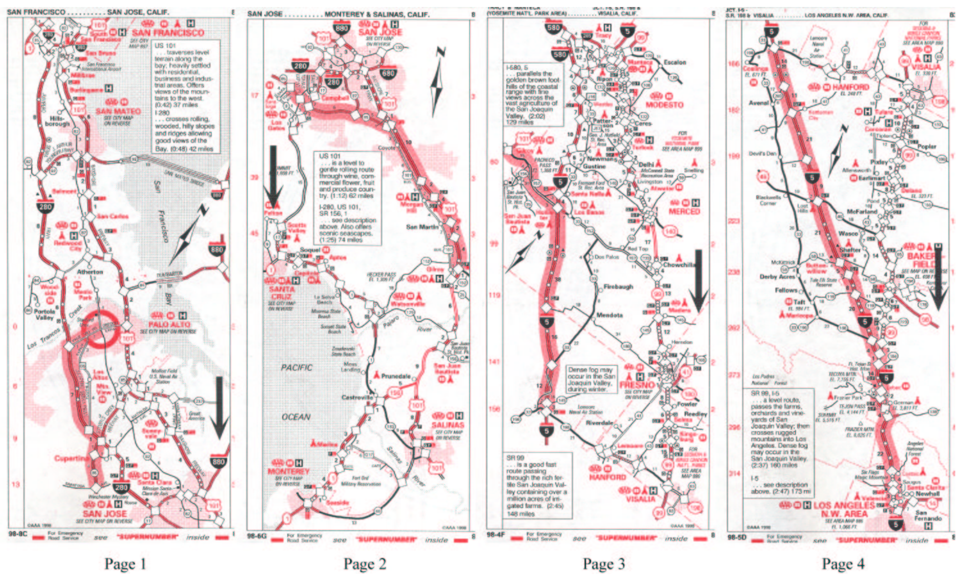


Figure 1.2: A strip map. A strip map depicting the first part of a route, highlighted in red, from Palo Alto to Los Angeles. The scale factor and orientation vary from page to page in order to center the route making it difficult to form a general understanding of the overall route. *(map courtesy of California State Automobile Association)*

able through the Web-based route mapping services and an example of such an overview/focus map collection is shown in figure 1.3. A constant scale factor is used within each map, but the scale factor differs across the maps. One of the maps is scaled so that it provides an overview of the entire route. This overview map is essentially a route highlight map and suffers many of the problems associated with that route mapping style. Since the scale of the overview map often reduces the readability of local turn information, focus maps showing turn-by-turn steps are also provided.

While this may seem like an effective combination, in practice the two sets of maps can be extremely difficult to use. The overview map rarely presents more than the overall direction, and placement of the route within the larger geographic context. Although turn-by-turn maps provide detailed turning point information, the use of distinct maps for each turn, often with differing orientation and scale, makes it difficult to understand how the maps correspond to one another. No single map provides a complete description of the route at the appropriate level of detail. Thus, just as with strip maps, the navigator may have difficulty forming an overall understanding of the route, leading to frustration and confusion.

Figure 1.3: An overview/focus map collection. A route from Stanford to Berkeley depicted with an overview/focus map collection. The overview show the entire route, but turning point information is not visible. The turn-by-turn focus maps depict each turn individually, but the varying scale factors and orientations make it difficult to understand how the maps correspond to one another. *(map courtesy of MapBlast!)*

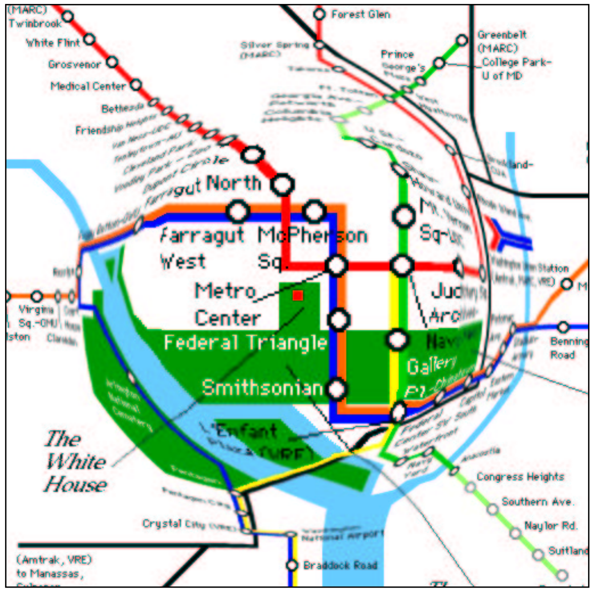


Figure 1.4: A 2D nonlinear distortion map. Spherical magnification is applied to a subway map to emphasize the important stations on the route, but extreme distortion at the edges of the spherical region makes it difficult to understand the surrounding context. Moreover, the severe distortion of the text labels makes the map very difficult to read. *(image courtesy of Keahey [Kea00])*

1.5.4 2D Nonlinear Distortion Maps

To ensure clear communication of the turning point information, different regions of the route often need to be depicted at different levels of detail. Recently several researchers [CCF95, Kea98] have attempted to use image warping techniques on general road maps and subway route maps in order to emphasize turning point information while showing all of the surrounding context. These methods allow users to choose regions of the map they wish to focus on and then apply nonlinear distortions, such as spherical magnification to enlarge these focus regions.

Such 2D distortion allows detailed information to be displayed only where relevant and often produces route maps that can be conveniently displayed on a single page. However, as shown in figure 1.4, a major problem with 2D image distortion techniques is that there are areas of extreme distortion at the edges of the focal points which make the overall route difficult to understand and follow. The severe distortion of the text labels makes the map particularly difficult to read. One way to improve this image warping approach would be to apply the warp only to the underlying lines representing the network of routes. The unwarped text labels could then be placed appropriately in the warped image so that the text would remain readable. This approach is common in graph layout systems such as Lamping and Rau's [LR94] hyperbolic tree browser. They apply warps to the node-edge structure of the graph and then overlay unwarped text labels. However, to our knowledge approach has not been applied to visualizing routes.

1.5.5 Hand-drawn Route Maps

One existing route mapping style, the hand-drawn map, manages to display each turning point along the route clearly and simultaneously maintain simplicity and a convenient form factor, as exemplified in figure 1.5. Instead of using a constant scale factor hand-drawn maps only maintain the relative ordering of the roads by length. Although each road is scaled by a different factor, longer roads appear longer than shorter roads.

Hand-drawn maps also omit most contextual information that does not lie directly along the route. This strategy reduces overall clutter and improves clarity. As shown in figure 1.5, at most a sparse set of local context information such as distances along each road and nearby city names are depicted because they greatly facilitate navigation with little impact on the overall readability of the map. The angles formed by the roads are regularized and road shape is generalized. Roads are often depicted as generically straight lines or simple curves. These distortions make the map simpler and thereby help to emphasize the essential turning point information.

1.6 Generalizing Route Maps

The generalization techniques used in LineDrive are based on those found in hand-drawn route maps. As we saw in the previous section, hand-drawn route maps use four basic types of generalization techniques: (1) the lengths of roads are distorted, (2) the angles at turning points are altered, (3) the shapes of the individual roads are simplified, and (4) extraneous context is reduced and the graphic representation of the roads and turning points are carefully chosen to emphasize the turning points.

Length Generalization: Hand-drawn maps often exaggerate the lengths of shorter roads on the route while shortening longer roads to ensure that all the roads and the turning points between them are visible. Often the map designer does not

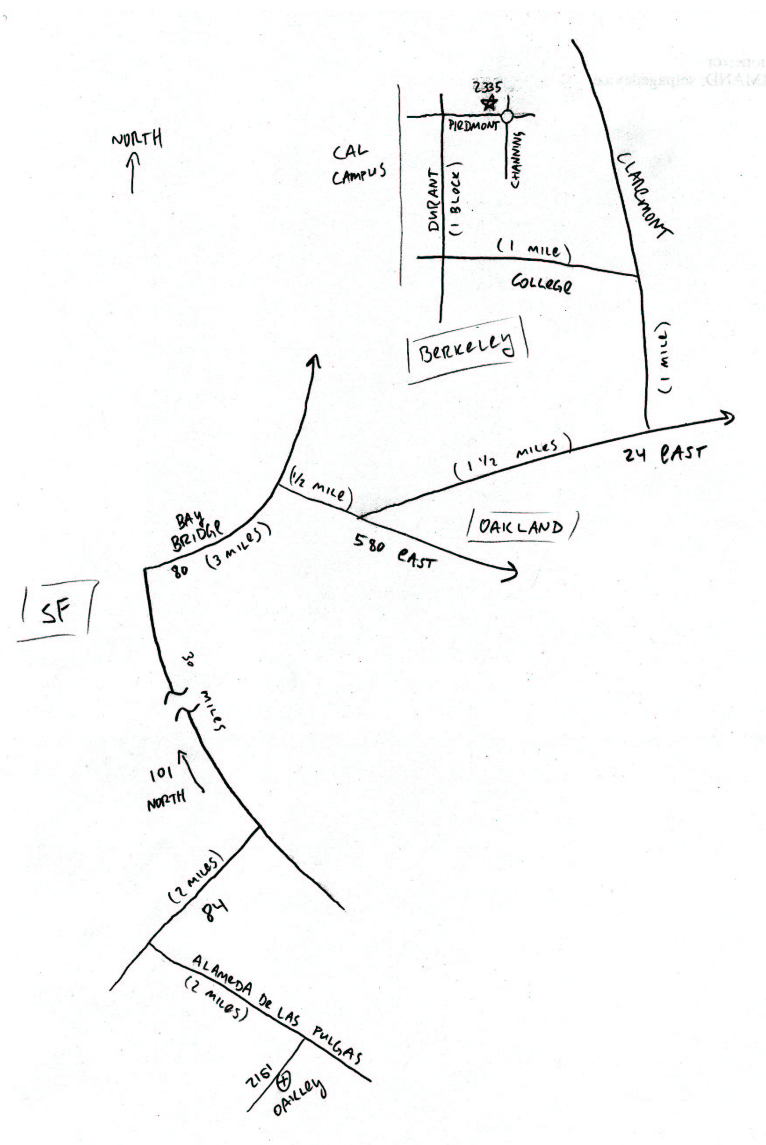


Figure 1.5: A hand-drawn route map. Roads lengths, angles and shape are distorted in order to emphasize the turning point information. Some context information such as distances and city names near the main route facilitate navigation without reducing the clarity of the overall map. Note that the mapmaker created this map from memory, without the aid of any reference maps. *(map courtesy of John Owens)*

know the exact length of the roads [Tve92] and only knows their lengths relative to one another. The flexibility of relative scaling allows hand-drawn route maps containing roads that vary over several orders of magnitude to fit within a conveniently sized image (i.e. a single small sheet of paper)and remain readable. The distortion is usually performed in a controlled manner so that shorter roads remain perceptually shorter than longer roads, and the overall shape of the route is maintained as much as possible.

Angle Generalization: Mapmakers often alter the angles of turns to improve the clarity of the turning points. Very tight angles are opened up to provide more space for growing shorter roads and labeling roads clearly. Roads are often aligned with the horizontal or vertical axes of the image viewport, to form a cleaner looking, regularized map [Tve81, Byr82]. Such angular distortions are acceptable because reorienting correctly requires knowing only the turn direction (left or right), not the exact turning angle.

Shape Generalization: Since a navigator does not need to make active decisions when following individual roads, knowing the exact shape of a road is usually not important. Simplifying the road shape removes extraneous information and places more emphasis on the turning points, where decisions need to be made. Roads with simplified shape are perceptually easier to differentiate as separate entities and are also easier to label clearly.

Graphic Generalization: The main technique for emphasizing turning point information in hand-drawn maps is to reduce extraneous context information by simply omitting it from the map. The route, depicted as a one-dimensional curve, is then the primary graphical element in the map. The negative space surrounding the route creates a simple, clean design that is clear and easy-to-read.

Some hand-drawn route maps use standard graphic design techniques to enhance the turning point information. For example, roads may be color-coded or their drawing style might change depending on the importance of the road. Highways might be drawn as double lines while residential roads are drawn as a single thin line. Professionally designed route maps sometimes place bullets at each turning point in order to emphasize that a decision is required at each of those points. Finally, the context information that does appear in hand-drawn maps is often de-emphasized. For example, cross-streets may be sketched in a lighter color to reduce interference with the main route.

1.7 Errors Due to Excessive Generalization

While these generalization techniques can increase the usability of the route map, they can also cause confusion and mislead the navigator if carried to an extreme. By simplifying road shape and distorting road lengths and angles, it is possible to drastically change the topology and overall shape of the route. We consider four types of errors due to excessive generalization, as shown in figure 1.6: false intersections, missing intersections, inconsistent turn directions and incorrect overall route shape.

False Intersections: A false intersection occurs in a route map when two roads are drawn as intersecting, even though they do not in fact intersect. All three forms of road generalization can generate false intersections. False intersections are topological errors that can deceive navigators into thinking that the route contains a loop or a shortcut when no such shortcut really exists.

Missing Intersections: A missing intersection occurs when the original route contains an intersection, but the intersection is missing in the generalized route map. This usually occurs when one road passes over another road as is often the case at highway interchanges. Although missing intersections, like false intersections, are topological errors that can deceive navigators they are less misleading than false intersections.

Inconsistent Turn Direction: Generalization can cause a right turn to appear as a left turn or vice versa. Inconsistent turn direction is always generated by either angle or shape generalization. Length generalization only affects the length of each road and therefore cannot create an inconsistent turn direction. Because turn direction is a fundamental component of turning point information such errors can cause severe problems for navigators.

Overall Route Shape: Generalization can also create drastic changes in the overall shape of the route. For example, growing one road while maintaining all the others at their original size via length generalization can alter the overall direction between the origin and destination of the route. Similarly angle generalization could cause an east-west route could become a north-south route. Shape generalization essentially requires simplifying road shape. The most extreme form of road simplification replaces each road with a straight line connecting its endpoints. Since each road is simplified individually and the endpoints of each road are never removed,

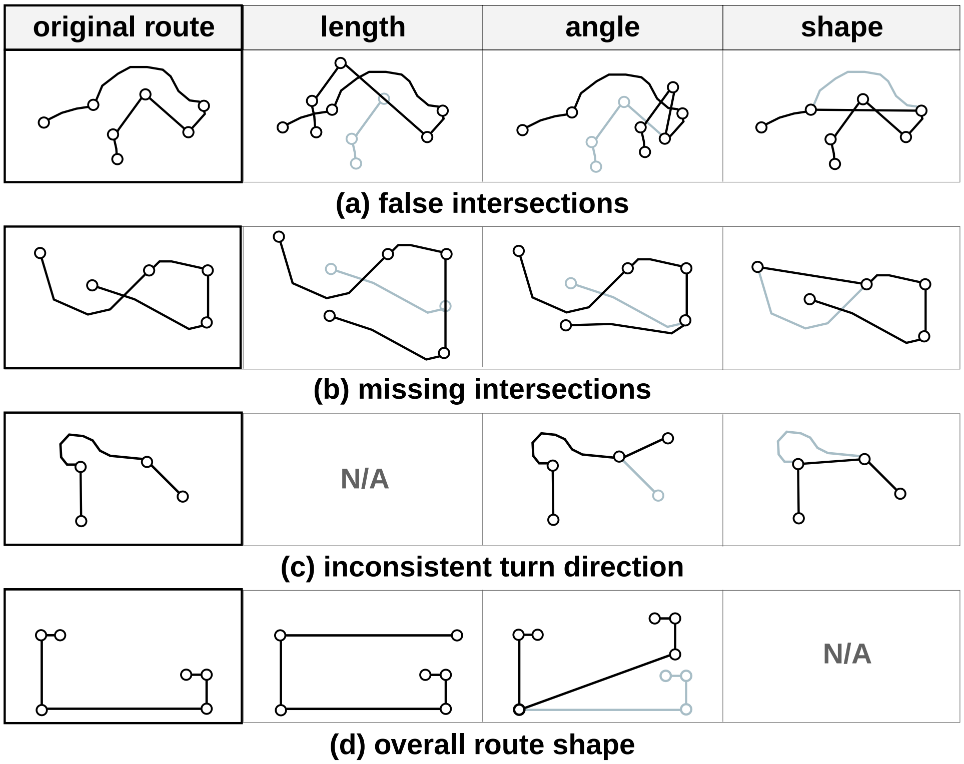


Figure 1.6: Errors due to excessive generalization. Excessive generalization can cause four types of errors in the topology and shape of the route. Each column shows the route after generalizing road length, angle or shape. For comparison, the undistorted route is shown in gray. While humans mapmakers are careful not to over-generalize a map to the point of introducing such errors in the map, automated mapmaking systems must explicitly check for these errors. (a) The original route does not contain an intersection but generalization causes a false intersection. (b) The original route contains an intersection but after generalization it is missing. (c) Generalization changes the turn direction so that a right turn appears to be a left turn or vice versa. Note that distorting road length cannot cause this error. (d) Generalization causes drastic changes in the overall shape of the route. Note that shape generalization cannot generate this error because each road is simplified individually and road endpoints are never removed.

extreme changes in overall route shape are not possible due to shape generalization. Although errors in overall route shape can lead to a distorted cognitive map of the route and make it difficult to estimate distances or orientation, such errors are the least detrimental of the four errors we consider.

Human mapmakers rarely generalize routes to the extreme required to induce these errors. Generalization is performed almost subconsciously, with constant error-checking to ensure that such misleading effects are not generated. Automated map-making systems, however, must be explicitly perform such error-checking. The bulk of LineDrive's road layout algorithm is designed to perform these checks. Like human mapmakers, LineDrive carefully generalizes road length, angle and shape, to dramatically improve the usability of the route maps.

Overview of Automated Design Techniques

Over the last two decades cartographers, graphic designers and computer scientists have started to work together to create automated map generalization algorithms. However, despite the fact that the distortion techniques used in hand-drawn route maps improve usability, there has been surprisingly little work on developing automatic generalization techniques based on these distortions. Most of the existing research has focused on developing simplification and abstraction techniques for standard road, geographical and political maps [BM91, Mac95, Imh82, Mon91]. Unlike route maps, these general purpose maps are designed to convey information about an entire region without any particular focus area. Thus, these maps cannot include the specific types of distortion that are used in route maps.

However, route maps can exploit some of the generalization techniques originally designed for standard regional maps. In this chapter we present two threads of related work on shape simplification and automated layout techniques for 2D displays. We consider how simplification and automated layout techniques can be applied to the problem of designing and rendering route maps.

1.8 Shape Simplification

Techniques for curve smoothing, interpolation, and simplification have been well-studied in a variety of contexts including cartography [DP73, VW93, WM98, BLR00] and computer graphics [Ram72, Far88, HS92, dBvKOS97]. Today, the most common simplification technique used in geographic information systems (GIS) and automated mapmaking systems, is the Douglas-Peucker algorithm which was independently developed in the early 1970's, first by Ramer [Ram72] and then by Douglas and Peucker [DP73].

Given a piecewise linear curve, specified as a set of *shape points* connected with

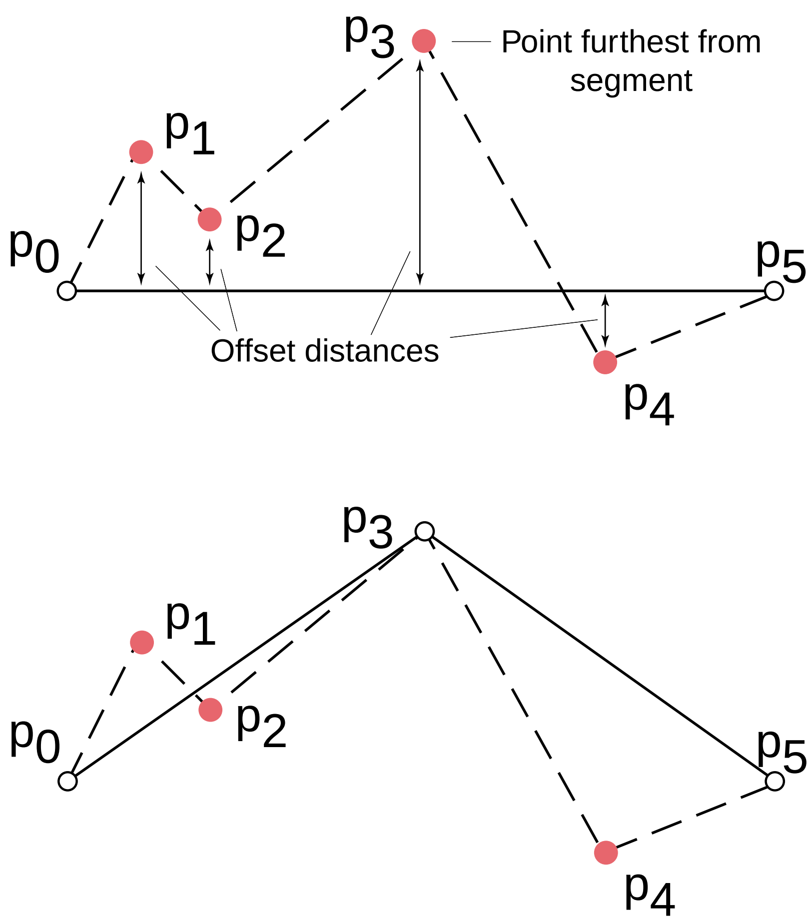


Figure 1.7: Douglas-Peucker algorithm. Initially only the extreme shape points p_0 and p_5 of the original piecewise linear curve (dotted) are retained forming a single segment $\overline{p_0 p_5}$. The algorithm computes the offset distance between every interior shape point and the segment and then retains the point furthest from the segment, in this case p_3. The algorithm proceeds recursively on both subsegments (in this case $\overline{p_0 p_3}$ and $\overline{p_3 p_5}$) until a given error threshold is achieved.

line segments, the Douglas-Peucker algorithm begins by throwing away all but the two extreme shape points of the curve and then iteratively attempts to add back only the most important interior shape points. On each iteration the algorithm forms a segment between the extreme shape points, computes the offset distance from each interior shape point to the segment and then adds back the shape point furthest from the segment. The curve is now broken into two segments and the algorithm performs the same test on each sub-segment recursively until an error threshold (e.g. the maximum offset distance is less than some pre-defined tolerance) is achieved. The procedure is illustrated in figure 1.7.

The offset distance is essentially a relevance metric for each shape point, and on each iteration the Douglas-Peucker algorithm retains on the most relevant shape point based on this metric. This approach can be thought of as an *additive* algorithm because it adds shape points back to the curve on each iteration. In contrast, a *subtractive* simplification algorithm begins with all the shape points and on each iteration throws away the least relevant shape points. Both approaches are similar and in fact the Douglas-Peucker algorithm can be easily framed as a subtractive algorithm; start with all the shape points and on each iteration throw away the interior shape point with the smallest offset distance, as long as the offset distance is smaller than the error threshold.

This basic approach to simplification has been applied with a variety of relevance metrics to simplify all types of one-dimensional curves found in maps, including roads, rivers and boundaries between regions such as state lines. One of the main directions of current research in curve simplification is designing relevance metrics that maintain important perceptual characteristics of the curve. Visvalingam and Whyatt [VW93] and Barkowsky et al. [BLR00] have shown that the effective area of a shape point, defined as the area subtended by the point and its two neighbors, is a better metric than offset distance for capturing the overall shape of the curve. This metric tends to eliminate the irrelevant small-scale bends in the curve, while preserving significant large-scale kinks. Wang and Müller [WM98] have designed a metric that finds and evaluates the relevance of sets of adjacent shape points that form a significant bend in the curve. Their elimination operator them removes all points within bends that fall below a given threshold.

One drawback with this class of simplification techniques is that they may introduce false intersection in the simplified curves. Several groups have proposed simplification algorithms that produce topologically consistent simplifications. However, these techniques tend to be complex and computationally expensive[dBvKS98, JBW95]. Recently, Saalfeld [Saa99] described a simple modification to the Douglas-Peucker algorithm for maintaining topological consistency using efficient local tests. These tests compute which shape points to add to the simplified curve in order to guarantee that the simplified curve will not contain false intersections. However this approach relies on properties of the Douglas-Peucker algorithm's offset distance relevance metric and therefore cannot be applied to simplification algorithms that use other types of relevance metrics.

One similarity among all the algorithms we have considered so far is that they only remove unnecessary shape points from the original curve. These algorithms never change the position of a shape point or introduce new shape points. Although a variety of smoothing and simplification algorithms that remove these restrictions have been developed [PBR98, LPL97, PAF96, ZB96, GHMS93, Far88], Guibas et al. [GHMS93] have shown that maintaining topological consistency in this more general case is NP-hard.

We use a relevance metric approach to simplifying roads in LineDrive. Like the previous approaches we only remove shape points from the original curve. However, we introduce several new constraints on the simplification process to avoid the topological and shape errors shown in figure 1.6.

1.9 Automated Layout

We define *layout* as the process of creating an arrangement for a set of visual elements. From a geometric standpoint, we must choose a position, orientation and scale for each element. Planning a layout is a common problem in a variety of domains that involve visual communication, including the design of user interfaces, architecture, web pages, documents, newspapers, magazines, posters, billboards and maps.

Today, the majority of layouts are created by hand, often by an expert graphic designer, and usually require many hours of experimentation to fully develop. At these rates, it is impossible to hand-craft layouts for time-critical applications requiring the communication of visual information. Moreover, novice users without a formal background in graphic design may not know the necessary heuristics to create the most effective layouts. Therefore, developing efficient tools for automatically creating effective, high-quality layouts for graphical presentations has become a major area of research.

In this section we consider previous approaches to automated layout that have influenced the automated layout algorithms used in LineDrive. Our review focuses on systems that design layouts meant to communicate information to human viewers. In particular, we do not cover automated VSLI design techniques [Len90] because the layouts created for VSLI chips are constrained by the peculiarities of a fabrication process and are therefore usually not meant to communicate information to humans. Similarly we do not present packing algorithms that are designed to generate minimum area or volume layouts [MMD97, Hof95, MDL92]. While many of the systems we describe are designed to solve layout problems in domains outside of cartography, we only present systems that are most relevant to automated route map layout. Lok and Feiner [LF01] as well as Hower and Graf [HG96] have compiled comprehensive surveys of research on automated layout techniques for graphical presentations.

1.9.1 Formulating Layout as Constraint-Based Optimization

In almost any layout problem there are restrictions on how the information can be laid out, and there are a set of criteria that can be used to evaluate the quality of the layout. A standard restriction for example is that the final layout can not exceed the dimensions of given display device. Evaluation criteria are often composed of domain specific rules that assess the usability and aesthetics of the layout. In our case, they include issues such as whether roads are large enough to be visible and the desirability of a label's placement relative to the object it is labeling.

Layout constraints can be used to encode both the restrictions on the layout and the evaluation criteria for the layout. In general constraints provide a natural abstraction for specifying the spatial relationships between visual elements. It is usually much simpler to specify a local constraint among a small subset of elements, such as element A should appear above element B, than it is to specify a complete procedure for creating a final layout for all the elements. Constraints allow a designer to describe the layout locally at a high level of abstraction without having to specify exactly how to achieve the final layout. Given a set of constraints, the goal of constraint-based optimization is to find a layout that best meets all of the constraints.

Many general-purpose automated layout systems require the user to specify the spatial constraints on the visual elements. In some cases the constraints are specified as text directives such as `(element A ABOVE element B)` [WW94]. Another approach is to provide visual interfaces for specifying constraints. A number of systems provide graphical user interfaces that allow users to graphically specify the relationships between elements [Gra92]. This approach is particularly common in user interface design toolkits [Mic97, Ous94, MASC85]. The most prevalent example of this form of constraint specification is in page layout systems such as Microsoft Word [RR99] that allow users to anchor images within documents.

While these approaches allow the users to specify a layout at a high level of abstraction, creating an effective layout requires some design knowledge. The user must decide that element A should appear above element B. Recently some researchers have applied machine learning techniques to automatically extract constraints for creating effective layouts. Most of these systems learn by watching expert users interactively specify constraints [Mas94, MMK93, BD86]. Zhou and Ma [ZM00, ZM01] have explored machine learning techniques for extracting these constraints directly from a set of example layouts.

When the layout problem is restricted to a particular domain, expert graphic designers can use domain knowledge to specify the spatial constraints as a pre-defined set of rules. For example, expert cartographers such as Imhof [Imh75], have developed a set of placement constraints for map labels. A common constraint is that labels should appear above the features they are labeling rather than below. Almost all automated map label placement systems [Zor97, ECMS97] define this set of constraints a priori so that novice end-users do not have to design the constraints themselves. Such systems eliminate the layout design work an end user must perform. If the computed layout is not optimal, interactive tools may be provided so the user can guide the system or modify the final result [AAL$^+$00, RMS97] For domain specific applications the fully automated method with predefined constraints produces excellent results and this is the approach we take in LineDrive.

1.9.2 Resolving Constraints

The constraint satisfaction problem appears in a wide variety of forms and the general problem has been well-studied in computer science [Mac92]. Although there are a wide variety of techniques for resolving a system of constraints, automated layout systems usually take one of four approaches; (1) grid-based geometry management, (2) linear programming with constraint propagation, (3) dynamic simulation and (4) search. We consider examples of each of these in turn.

1.9.2.1 Grid-Based Geometry Management

Axis-aligned grids are a common tool for organizing the layout of two-dimensional presentations such as newspapers. Several automated layout systems simplify constraint resolution by only allowing the specification of constraints that conform to an underlying grid. Each visual element is treated as content for a grid cell and constraints are specified relative to other grid cells. Positional constraints are used to specify the position of one cell with respect to another, while hierarchical constraints are used to specify that one cell should appear within another cell. Initially the entire page forms a single cell at the base of the hierarchy and all the other cells are placed within it.

Given a set of constraints specified in this grid-based manner, a geometry manager then expands the constraints from the top down to determine the absolute position of each grid cell. Such systems often rely on the user to solve any inconsistencies among the constraints. Feiner's GRIDS [Fei88] system was one of the first to use this approach to automating layout. Today this grid-based approach is especially popular among user interface toolkits including Tk [Ous94], the X toolkit [MASC85], and Microsoft's Foundation Classes [Mic97].

1.9.2.2 Linear Programming With Constraint Propagation

The widespread interest in the constraint satisfaction problem has led to the development of general-purpose systems for resolving systems of linear constraints [BB98, SMFBB93]. Each constraint must be specified as either a linear equation or an inequality, and each constraint equation is given a priority so that higher priority constraints are resolved in favor of lower priority constraints if there is an inconsistency in the constraint system. The solvers then apply linear programming in combination with constraint propagation in order to find the best solution to the system.

In order to use a general-purpose constraint solver to design a graphical presentation, higher-level constraints must be translated into a set of linear equations and inequalities. Weitzman and Wittenburg [WW94] have developed a full relational grammar for specifying the high-level constraints necessary to generate multimedia presentations. Users enter a set of text rules using the grammar and their system converts these into a set of low-level constraint equations which are fed into a back-end constraint solver that generates the final layout. Graf's LayLab [Gra92] system performs a similar transformation of high-level constraints into low-level constraint equations. However, unlike Weitzman and Wittenburg's system, LayLab allows users to visually specify the high-level constraints though a graphical user interface.

1.9.2.3 Dynamic Simulation

Another technique for resolving layout constraints is to treat the visual elements as masses and specify the constraints as forces acting on the masses. Often the forces are described using the standard physics model of spring in which the force is proportional to the distance between the element center of mass and its desired position. Once a mass-spring system has been defined in this manner, standard physically-based constrained dynamics simulators [WGW90] can be used to find the rest positions, and hence the layout for the visual elements

This approach has been widely explored in the domain of graph drawing [BETT99, FR91]. Its popularity may lie in the fact that it is very intuitive to think of each node in the graph as a mass and each edge between nodes as a spring that links them. The nodes are placed in an initial position and the system is relaxed until it converges to an energy-minimizing state. One drawback of this approach is that minor tweaking of layout parameters can result in major changes in the final layout.

In computer graphics, Gleicher and Witkin [GW94] have applied constrained dynamic simulation to two-dimensional drawing programs. Their system can simultaneously enforce layout constraints while allowing the user to drag, thereby allowing the user to interactively explore the configurations of the model consistent with the layout constraints. More recently, Harada et al. [HWB95] have used constrained dynamics to provide an interactive user interface for an architectural layout system.

The text layout engine used in TeX [Knu99] builds a representation similar to a mass-spring system. Each word within a paragraph represents a mass and the spaces between them represent springs. The goal of the layout engine is to determine the spacing between the words so that the line-breaks in the formatted text appear aesthetically pleasing. Although the problem is set up to resemble a mass-spring system, the TeX layout engine uses dynamic programming techniques rather than physically-based dynamic simulation.

1.9.2.4 Search

Many layout problems can be posed as a search for an optimal layout over a space of possible layouts. To frame the layout problem as a search we need to define an initial layout and two functions: a *score* function that assesses the quality of a layout based on the evaluation criteria, and a *perturb* function that manipulates a given layout to produce a new layout within the search space. Both the score and the perturb functions are defined by the set of constraints on the layout. Given these two functions the search can be performed using any search technique including A*, tabu search, gradient descent and simulated annealing [MF00].

In the graph drawing domain a variety of search techniques have been explored. Kamada and Kawai [KK89] have used gradient descent while Davidson and Harel [DH96] exploit simulated annealing. Brandenburg et al. [BHR95] and Tunkelang [Tun93] describe more complicated algorithms that combine several search techniques. In the domain of cartography, search techniques are widely used to automatically label point and line features in traditional geographic maps. Marks and Shieber [MS91] have shown that finding optimal label placements is NP-complete and several previous systems have used randomized search to find near-optimal label placements [Zor97, ECMS97]. Simulated annealing is the most commonly used randomized search algorithm because it efficiently covers the search space and is simple to implement. For these reasons, we also use simulated annealing, not only for placing labels, but also for laying out the roads and context information in LineDrive route maps. The simulated annealing algorithm is described by the following pseudo-code:

procedure SimAnneal()

```
InitializeLayout()
E ← ScoreLayout()
while(! termination condition)
  PerturbLayout()
  newE ← ScoreLayout()
  if ((newE > E) and (Random() < (1.0 − e^(−ΔE/T))))
    RevertLayout()
  else
    E ← newE
  Decrease(T)
```

Implementing the algorithm is straightforward and requires the specification of four functions. The InitializeLayout() function defines the initial placement for each of the visual elements and thereby provides a starting point for the search. The PerturbLayout() function provides a method for changing a given layout into a new layout, while the RevertLayout() functions inverts the actions of PerturbLayout() to go from the new layout back to the previous layout. Finally the ScoreLayout() function computes how close to optimal the current layout is. By convention, scores are defined to always be positive and the lower the score the better the layout. Therefore, the goal of simulated annealing is to minimize the score.

As shown in the pseudo-code, the simulated annealing algorithm accepts all good moves within the search space and, with a probability that is an exponential function of a temperature T, accepts some bad moves as well. As the algorithm progresses, T is annealed (or decreased), resulting in a decreasing probability of accepting bad moves. Accepting bad moves in this manner allows the algorithm to escape local minima in the score function.

We can divide our constraints into two sets; 1) *hard constraints* consist of characteristics required of any acceptable layout and therefore bound the space of possible layouts, while 2) *soft constraints* consist of characteristics desired in the final layout but not required. In designing the constraints, it is important not to impose too many hard constraints or the layout problem will be overconstrained making it impossible to find any acceptable layout. The hard constraints are typically imposed through the perturb function which is designed to only generate layouts that meet the hard constraints. The score function checks how well a given layout achieves the soft constraints.

The difficult aspects of characterizing the layout problem as a search are designing a numerical score function that efficiently captures all of the desirable features of the optimal layout and defining a perturb function that covers a significant portion of the search space. In the following chapters as we discuss the different layout stages of LineDrive, we will focus on explaining these aspects of our algorithm design.

Bibliography

[AAL$^+$00] David Anderson, Emily Anderson, Neal Lesh, Joe Marks, Brian Mirtich, David Ratajczak, and Kathy Ryall. Human-guided simple search. *Proceedings of AAAI 2000*, pages 209–216, August 2000.

[All99] Gary L. Allen. Spatial abilities, cognitive maps, and wayfinding: Bases for individual differences in spatial cognition and behavior. In Reginald G. Golledge, editor, *Wayfinding Behavior*, pages 46–80. Johns Hopkins University Press, 1999.

[BB98] Greg J. Badros and Alan Borning. The cassowary linear arithmetic cosntraint solving algorithm: Interface and implementation. Technical Report UW-98-06-04, University of Washington, 1998.

[BD86] Alan Borning and Robert Duisberg. Constraint-based tools for building user interfaces. *ACM Transactions on Graphics*, 5(4):345–374, 1986.

[Bel95] Scott M. Bell. Cartographic presentation as an aid to spatial knowledge acquisition in unknown environments. Master's thesis, Department of Geography, University of Santa Barbara, 1995.
`http://www.geog.ucsb.edu/ bell/thesis/contents.html`.

[BETT99] Giuseppe Di Battista, Peter Eades, Roberto Tamassia, and Ioannis G. Tolls. *Graph Drawing - Algorithms for the visualization of graphs*. Prentice Hall, 1999.

[BHR95] Franz J. Brandenburg, Michael Himsolt, and Christoph Rohrer. An Experimental Comparison of Force-Directed and Randomized Graph Dr awing Algorithms. In *Proceedings of Graph Drawing '95, Lecture Notes in Computer Science 1027*, pages 76–87. Springer-Verlag, 1995.

[BLR00] Thomas Barkowsky, Login J. Latecki, and Kai-Florian Richter. Schematizing maps: Simplification of geographic shape by discrete curve evolution. In C. Habel C. Freska, W. Brauer and K.F. Wender, editors, *Spatial Cognition II*, pages 41–53. Springer-Verlag, 2000.

[BM91] Barbara P. Buttenfield and Robert B. McMaster, editors. *Map Generalization: Making rules for knowledge representation*. Longman Scientific, 1991.

[But86] Barbara P. Buttenfield. Comparing distortion on sketch maps and mds configurations. *Professional Geographer*, 38(3):238–246, 1986.

[Byr82] R. W. Byrne. Geographical knowledge and orientation. In A. Ellis, editor, *Normality and Pathology in Cognitive Functions*, pages 239–264. London: Academic Press, 1982.

[CCF95] M. Sheelagh T. Carpendale, David J. Cowperthwaite, and F. David Fracchia. Three-dimensional pliable surfaces: For effective presentation of visual information. In *ACM Symposium on User Interface Software and Technology (UIST 95)*, pages 217–226. ACM Press, 1995.

[Cha83] William G. Chase. Spatial representations of taxi drivers. In D. R. Rogers and J. A. Sloboda, editors, *Acquisition of Symbolic Skills*, pages 391–405. New York: Plenum Press, 1983.

[Cho99] Eric Chown. Error tolerance and generalization in cognitive maps: Performance without precision. In Reginald G. Golledge, editor, *Wayfinding Behavior*, pages 349–370. Johns Hopkins University Press, 1999.

[dBvKOS97] Mark de Berg, Marc van Krevald, Mark Overmars, and Otfried Schwarzkopf, editors. *Computational Geometry: Algorithms and Applications*. Springer, 1997.

[dBvKS98] Mark de Berg, Marc van Krevald, and Stefan Schirra. Topologically correct subdivision simplification using the bandwidth criterion. *Cartography and Geographic Information Systems*, 25(4):243–257, 1998.

[Den97] Michel Denis. The description of routes: A cognitive approach to the production of spatial discourse. *Cahiers de Psychologie Cognitive*, 16(4):409–458, 1997.

[DH96] Ron Davidson and David Harel. Drawing Graphics Nicely Using Simulated Annealing. *ACM Transactions on Graphics*, 15(4):301–331, 1996.

[DP73] David H. Douglas and Thomas K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *The Canadian Cartographer*, 10(2):112–122, 1973.

[DPCB99] Michel Denis, Francesca Pazzaglia, Cesare Cornoldi, and Laura Bertolo. Spatial discourse and navigation: An analysis of route directions in the city of Venice. *Applied Cognitive Psychology*, 13(2):145–174, 1999.

[ECMS97] Shawn Edmondson, Jon Christensen, Joe Marks, and Stuart Shieber. A general cartographic labeling algorithm. *Cartographica*, 33(4):12–23, 1997.

[Far88] Gerald Farin. *Curves and Surfaces for Computer-Aided Geometric Design*. Academic Press Ltd., 1988.

[Fei88] Steven Feiner. A grid-based approach to automating display layout. In *Proceedings of Graphics Interface '88*, pages 192–197, 1988.

[FR91] Thomas M. J. Fruchterman and Edward M. Reingold. Graph drawing by force-directed placement. *Software–Practice and Experience*, 21(11):1129–1164, 1991.

[GHMS93] Leonidas J. Guibas, John E. Hershberger, Joseph B. Mitchell, and Jack S. Snoeyink. Approximating polygons and subdivisions with minimum-link paths. *International Journal of Computational Geometry and Applications*, 3(4):383–415, 1993.

[Gol99] Reginald G. Golledge. Human wayfinding and cognitive maps. In Reginald G. Golledge, editor, *Wayfinding Behavior*, pages 5–45. Johns Hopkins University Press, 1999.

[Gra92] Winfried H. Graf. Constraint-based graphical layout of multimodal presentations. In T. Catarci, M. F. Costabile, and S. Levialdi, editors, *Proceedings Advanced Visual Interfaces AVI 92*, pages 365–385. World Scientific, May 1992.

[GW94] Michael Gleicher and Andrew Witkin. Drawing with constraints. *The Visual Computer*, 11:39–51, 1994.

[HG96] Walter Hower and Winfried H. Graf. A bibliographic survey of constraint-based approaches to CAD, graphics, layout, visualization and related topics. *Knowledge-Based Systems*, 9(7):449–464, 1996.

[Hof95] Micha Hofri, editor. *Analysis of Algorithms.* New York: Oxford University Press, 1995.

[HS92] John Hershberger and Jack Snoeyink. Speeding up the Douglas-Peucker line-simplification algorithm. In *5th Intl. Symp. on Spatial Data Handling*, pages 134–143, 1992.

[HWB95] Mikako Harada, Andrew Witkin, and David Baraff. Interactive physically-based manipulation of discrete/continuous models. *Computer Graphics*, 29(Annual Conference Series):199–208, 1995.

[Imh75] Eduard Imhof. Positioning names on maps. *The American Cartographer*, 2(2):128–144, 1975.

[Imh82] Eduard Imhof. *Cartographic Relief Presentation.* Berlin: de Gruyter, 1982.

[JBW95] Christopher B. Jones, Geraint L. Bundy, and J. Mark Ware. Map genrealization with a triangulated data structure. *Cartography and Geographic Information Systems*, 22(4):317–331, 1995.

[Kea98] T. Alan Keahey. The generalized detail-in-context problem. *Proceedings of the IEEE Symposium on Information Visualization*, pages 171–180, October 1998.

[Kea00] T. Alan Keahey. A brief tour of nonlinear magnification, 2000. `http://www.cs.indiana.edu/hyplan/tkeahey/research/nlm/nlmTour.html`.

[KK89] Tomihisa Kamada and Satoru Kawai. An Algorithm for Drawing General Undirected Graphs. *Information Processing Letters*, 31(1):7–15, April 1989.

[Knu99] Donald E. Knuth. *Digital Typography (CSLI Lecture Notes 78).* Center for the Study of Language and Information, Stanford, California, 1999.

[Len90] Thomas Lengauer, editor. *Combinatorial Algorithms for Integrated Circuit Layout.* John Wiley & Sons Ltd., 1990.

[LF01] Simon Lok and Steven Feiner. A survey of automated layout techniques for information presentations. In *1st International Symposium on Smart Graphics*, March 2001.
`http://www.smartgraphics.org/schedule.html`.

[LPL97] François Lecordix, Corinne Plazanet, and Jean-Philippe Lagrange. PlaGe: A platform for research in generlization. application to caricature. *GeoInformatica International Journal*, 1(2):161–182, 1997.

[LR94] John Lamping and Ramana Rao. Laying out and visualizing large trees using hyperbolic space. In *ACM Symposium on User Interface Software and Technology (UIST 94)*, pages 13–14. ACM Press, 1994.

[Lyn60] Kevin Lynch. *The Image of the City*. Cambridge, Massachusetts: The MIT Press, 1960.

[Mac92] Alan K. Mackworth. The logic of constraint satisfaction. *Artificial Intelligence*, 58(1–3):3–20, 1992.

[Mac95] Alan M. MacEachren. *How Maps Work*. The Guilford Press, 1995.

[Mas94] Toshiyuki Masui. Evolutionary learning of graph layout constraints from examples. In *ACM Symposium on User Interface Software and Technology*, pages 103–108, 1994.

[MASC85] Joel McCormack, Paul Assente, Ralph Swick, and D. Converse. *X Toolkit Intrinsics – C Language Interface*. Digital Equipment Corporation, Maynard, Massachusetts, 1985.

[MB83] Carl Moreland and David Bannister, editors. *Antique Maps*. Phaidon Press, 1983.

[MDL92] Victor J. Milenkovic, Karen Daniels, and Zhenyu Li. Placement and compaction of nonconvex polygons for clothing manufacture. In *Proceedings 4th Canadian Conference on Computational Geometry*, pages 236–243, 1992.

[MF00] Zbigniew Michalewicz and David B. Fogel, editors. *How to Solve It: Modern Heuristics*. Springer, 2000.

[Mic97] Microsoft. *Microsoft Visual C++ MFC Library Reference*. Microsoft Press, Redmond, Washington, 1997.

[MJ87] Alan M. MacEachren and Gregory B. Johnson. The evolution, application and implications of strip format travel maps. *The Cartographic Journal*, 24(2):147–158, 1987.

[MMD97] Francesco Maffioli, Silvano Martello, and Mauro Dell'Amico, editors. *Annotated Bibliographies in Combinatorial Optimization*. John Wiley & Sons, 1997.

[MMK93] Brad A. Myers, Richard G. McDaniel, and David S. Kosbie. Marquise: Creating complete user interfaces by demonstration. In *INTERCHI '93, Human Factors in Computing Systems*, pages 293–300, Amsterdam, Netherlands, April 1993.

[Mon91] Mark Monmonier. *How to Lie With Maps*. The University of Chicago Press, 1991.

[MS91] Joe Marks and Stuart Shieber. The computational complexity of cartographic label placement. Technical Report ITR-05-91, Center for Research in Computing Techniology, Harvard University, March 1991.

[Ogi89] John Ogilby. *Britannia*. 1675: Old Hall Press, 1989. Facsimile of 1675 edition.

[Ous94] John K. Ousterhout. *Tcl and the Tk Toolkit*. Addison-Wesley, 1994.

[PAF96] Corinne Plazanet, Jean-George Affholder, and Emmanuel Fritsch. The importance of geometric modelling in linear feature generalisation. *Cartography and Geographic Information Systems*, 22(4):291–305, 1996.

[Pai69] J. Pailhous. Representation de l'espace urbain et cheminements (representation of urban space and traveling). *Le Travail Humain*, 32:239–250, 1969.

[PBR98] Crinne Plazanet, Nara M. Bigolin, and Anne Ruas. Experiments with learning techniques for spatial model enrichment and line generalization. *GeoInformatica International Journal*, 2(4):315–333, 1998.

[Ram72] Urs Ramer. An iterative apprach for polygonal approximation of planar closed curves. *Computer Graphics and Image Processing*, 1:244–256, 1972.

[RMS97] Kathy Ryall, Joe Marks, and Stuart Shieber. An interactive constraint-based system for drawing graphs. In *ACM Symposium on User Interface Software and Technology (UIST 97)*, pages 97–104. ACM Press, 1997.

[RR99] Charles Rubin and B. Russel. *Running Microsoft Word 2000*. Microsoft Press, Redmond, Washington, 1999.

[Saa99] Alan Saalfeld. Topologically consistent line simplification with the Douglas-Peucker algorithm. *Cartography and Geographic Information Systems*, 26(1):7–18, 1999.

[SMFBB93] Michael Sannella, John Maloney, Bjorn Freeman-Benson, and Alan Borning. Multi-way versus one-way constraints in user interfaces: Experience with the deltablue algorithm. *Software–Practice and Experience*, 23(5):529–566, 1993.

[Thr72] Norman J. Thrower. *Maps and Man: An Examination of Cartography in Relation to Culture and Civilization*. Englewood Cliffs, New Jersey: Prentice Hall, 1972.

[Thr96] Norman J. Thrower. *Maps in Civilization: Cartography in Culture and Society*. Chicago: University of Chicago Press, 1996.

[TL99] Barbara Tversky and Paul Lee. Pictorial and verbal tools for conveying routes. In C. Freska and D. M. Mark, editors, *COSIT*, pages 51–64, 1999.

[Tol48] Edward C. Tolman. Cogntive maps in rats and men. *Psychological Review*, 55(4):189–208, 1948.

[Tuf90] Edward Tufte. *Envisioning Information*. Conneticut: Graphics Press, 1990.

[Tun93] Daniel Tunkelang. A Layout Algorithm for Undirected Graphs. In *Proceedings of Graph Drawing '93, ALCOM International Workshop PA RIS 1993 on Graph Drawing and Topological Graph Algorithms*, 1993.
`http://reports-archive.adm.cs.cmu.edu/anon/1994/ CMU-CS-94-161.ps.`

[Tve81] Barbara Tversky. Distortions in memory for maps. *Cognitive Psychology*, 13(3):407–433, 1981.

[Tve92] Barbara Tversky. Distortions in cognitive maps. *Geoforum*, 23(2):131–138, 1992.

[VW93] Maheswari Visvalingam and J.D. Whyatt. Line generalisation by the repeated elimination of points. *Cartographic Journal*, 30(1):46–51, 1993.

[WGW90] Andrew Witkin, Michael Gleicher, and William Welch. Interactive dynamics. *Computer Graphics (1990 Symposium on Interactive 3D Graphics)*, 24(2):11–21, 1990.

[WM98] Zeshen Wang and Jean-Claude Müller. Line generalization based on analysis of shape characteristics. *Cartography and Geographic Information Systems*, 25(1):3–15, 1998.

[WW94] Louise Weitzman and Kent Wittenburg. Automatic presentation of multimedia documents using relational grammars. In *In Proceedings of ACM International Conference on Multimedia MULTIMEDIA 94*, pages 443–452. New York: ACM Press, October 1994.

[ZB96] F. Benjamin Zhan and Barbara P. Buttenfield. Multi-scale representation of a digital line. *Cartography and Geographic Information Systems*, 23(4):206–228, 1996.

[ZM00] Michelle Zhou and Sheng Ma. Towards applying machine learning to design rule acquisition for automated graphics generation. In *In Proceedings 2000 AAAI Spring Symposium Series on Smart Graphics*, March 2000.

[ZM01] Michelle Zhou and Sheng Ma. Representing and retrieving visual presentations for example based graphics generation. In *1st International Symposium on Smart Graphics*, March 2001. `http://www.smartgraphics.org/schedule.html`.

[Zor97] Steven Zoraster. Practical results using simulated annealing for point feature label placement. *Cartography and GIS*, 24(4):228–238, 1997.