Analysis of Smoothly Varying Textures

with Applications to Segmentation

Jason Chang

CSAIL, MIT
jchang7@csail.mit.edu

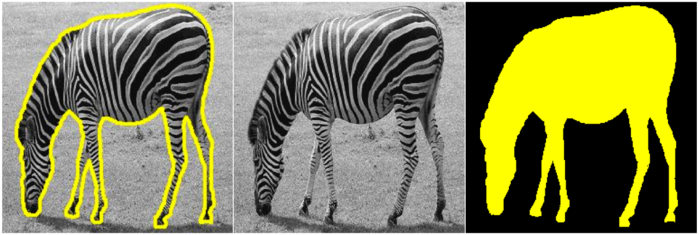
June 2, 2010

Outline

Outline

Image Segmentation

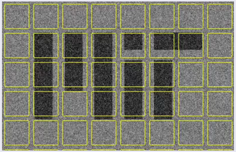
- Separate an image into disjoint regions
- Assume regions have common statistical properties



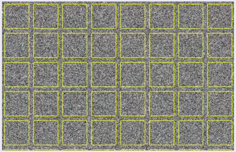
Simple Images with i.i.d. Pixel Intensities [6]

When are results good?

Minimal spatial dependencies conditioned on the label



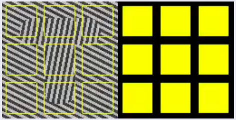
movies/mit-unimodal.mp4



movies/mit-bimodal.mp4

When does this approximation fail?

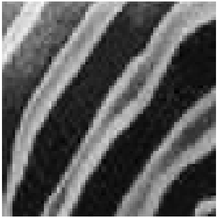
Strong spatial structures within regions

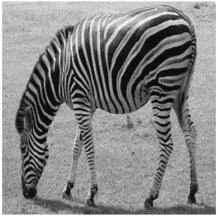


movies/t-orientation-scalar.mp4

Textured Image Segmentation

- Ill-posed nature - Segment the stripes or the zebra

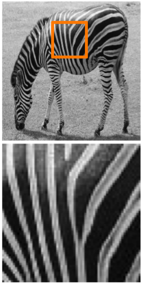


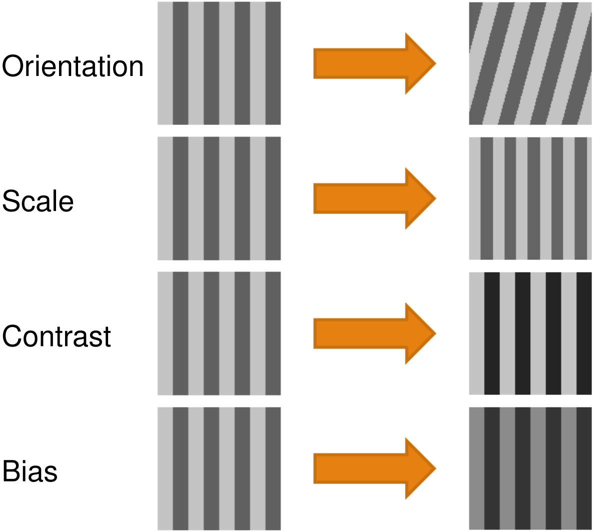


- Must consider local neighborhoods instead of pixels

Outline

A Motivating Example





Modelling Goals

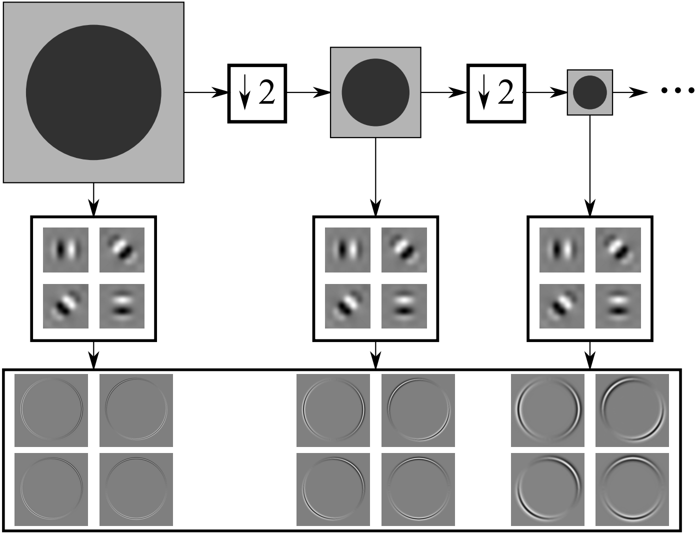
Approach

Find the best scale to represent a texture at each pixel

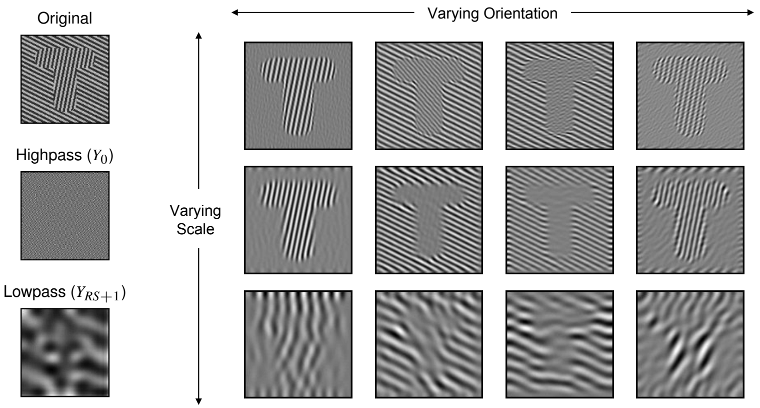
- Measure a notion of contrast, bias, and orientation at that scale
- Features should be not vary much within a constant texture
- Estimate spatial dependencies in features

Outline

Steerable Pyramids [8]



Steerable Pyramids [8]



Steering the Filters [8]

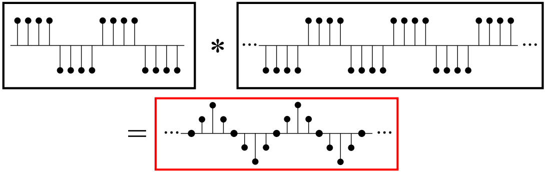
Theorem

With bounded error, the output at any orientation can be computed from a linear combination of the basis

$$y_i^s(\theta) = \sum_{\phi \in \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}} b_\phi(\theta)\, y_i^s(\phi)$$

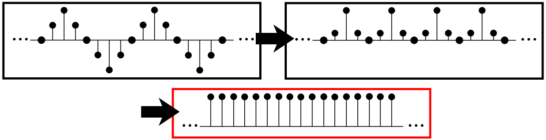
$$b_\phi(\theta) = \frac{\cos(\theta - \phi) + \cos(3(\theta - \phi))}{2}$$

Feature Extraction - Local Energy



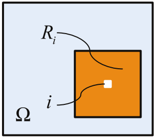
Observation

Filter outputs go positive and negative, and vary in between modes. To get a location independent measure, consider the local energy of the filter output.



Feature Extraction

For each pixel, i, we consider a neighborhood around it, R_i

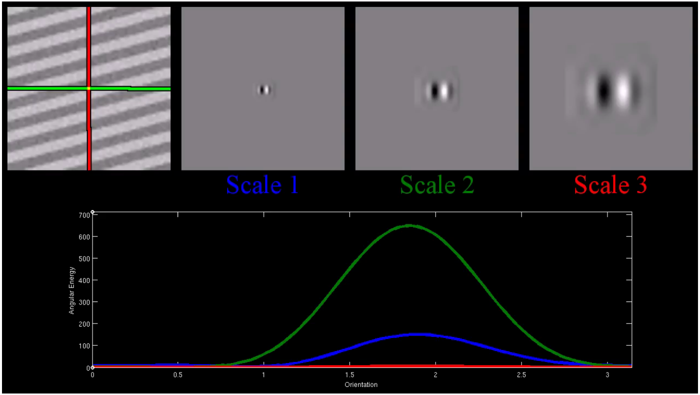


Analyzing Angular Energy

In scale, s, we consider the following local angular energy:

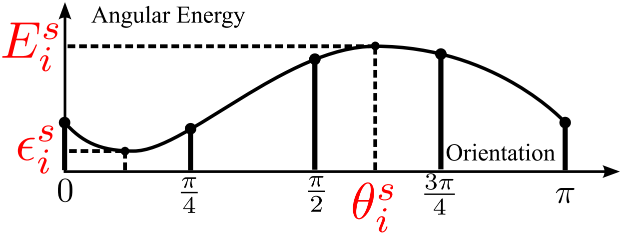
$$E_i^s(\theta) = \frac{1}{|R_i^s|} \sum_{j \in R_i^s} y_j^s(\theta)^2$$

Local Angular Energy



movies/energy-sweep.mp4

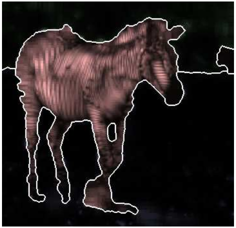
Feature Extraction

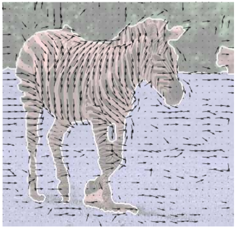


Feature Set

Scale	$\eta_i = \arg\max_s \max_\theta E_i^s(\theta)$	Orientation	$\theta_i = \arg\max_\theta E_i^{\eta_i}(\theta)$
Contrast Energy	$E_i = E_i^{\eta_i}(\theta_i^{\eta_i})$	Residual Energy	$\epsilon_i = E_i^{\eta_i}(\theta_i + \pi/2)$
Bias	$\mu_i = \frac{1}{\lvert R_i^{\eta_i}\rvert} \sum_{j \in R_i^{\eta_i}} x_j$		

Visualizing Features





Outline

Boundary Effects

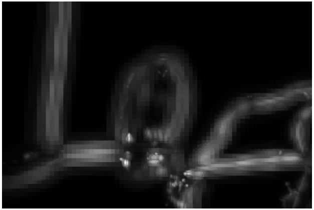
Object boundaries affect features in two ways:

1. Filtered outputs are corrupted near boundaries
2. Local neighborhood, R_i, in angular energy can span boundaries

Boundary Effects

Object boundaries affect features in two ways:

1. Filtered outputs are corrupted near boundaries
2. Local neighborhood, R_i, in angular energy can span boundaries



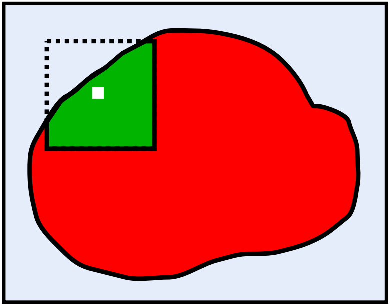
Corrupted Filter Outputs

- Similar to image boundary effects, zero padding regions creates artifacts
- Conditioned on a segmentation, we reflect each region across the object boundary and re-filter the image

Local Region in Angular Energy

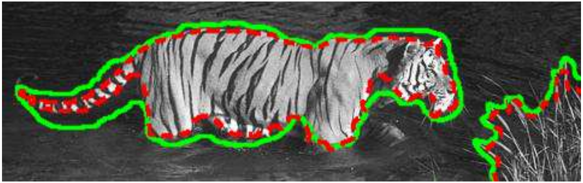
$$E_i^s(\theta) = \frac{1}{|R_i^s|} \sum_{j \in R_i^s} y_j^s(\theta)^2$$

Intstead of using R_i, we use $R_i' = R_i \cap R^{\pm}$



Boundary Effects

- We call these steps the border refinement step
- The refinement step is computationally expensive and creates many more local extrema
- We first segment an image without refinement (until convergence) and then perform refinement.



Outline

Intrinsic Feature Model

- We want to capture smooth changes in the features
- Model feature as output of intrinsic feature ($*$) subject to smooth, additive Markov random field
- Intrinsic feature distributions are estimated non-parametrically (using a kernel density estimate[7])

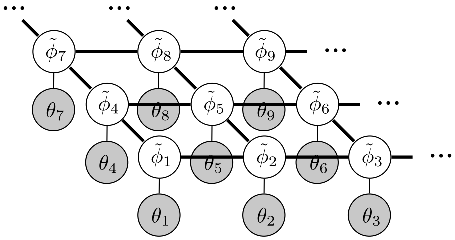
$$\underline{\theta} = \underline{\tilde{\phi}} + \underline{\theta}^*$$

$$\underline{\eta} = \underline{\tilde{\nu}} + \underline{\eta}^*$$

$$\underline{x} = \underline{g} \circ (\underline{b} + \underline{\mathcal{R}})$$

$$\Rightarrow \log \underline{E} = \log \underline{g} + \log \underline{E}^*$$

$$\Rightarrow \underline{\mu} = \underline{g} \circ (\underline{b} + \underline{\mu}^*)$$



MRF Estimation

- Perform MAP estimation of the smooth fields:

$$\underline{\tilde{\phi}} = \arg\max_{\underline{\phi}} p(\underline{\phi}|\underline{\theta})$$

- Using Bayes rule and differentiating:

$$\underline{\tilde{\phi}}^{(k+1)} = F^{-1}\left(\underline{\theta} - w_p^{\theta}\left(\underline{\theta} - \underline{\tilde{\phi}}^{(k)}\right)\right)$$

$$w_p^{\theta}(\cdot) = \frac{\sum_s \left(\theta_s - \tilde{\phi}_s\right) K\left((\cdot) - \theta_s + \tilde{\phi}_s\right)}{\sum_s K\left((\cdot) - \theta_s + \tilde{\phi}_s\right)}$$

$$F = \left(\frac{2}{h^2}\Lambda_{\phi}\right)^{-1} + I$$

MRF Estimation

- We show here that F performs highpass filtering with unity DC gain

$$F = \left(\frac{2}{h^2}\Lambda_\phi\right)^{-1} + I$$

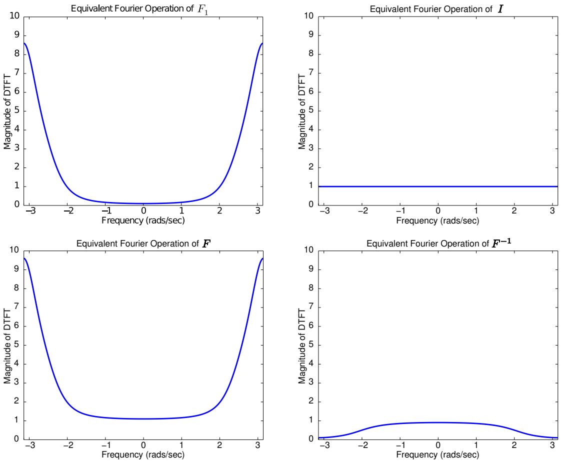
- Treat Λ_ϕ as lowpass filtered i.i.d. noise

$$F = \left(\frac{2}{h^2}L\sigma_\phi^2 IL^T\right)^{-1} + I = \underbrace{\frac{h^2}{2\sigma_\phi^2}H^T H}_{F_1} + I$$

- The lowpass filter operator, L, has unity DC gain. H much be a highpass filter operator with unity DC gain.
- Assuming $h \ll \sigma_\phi$, F_1 has a DC gain close to zero

MRF Estimation

$$F = F_1 + I$$



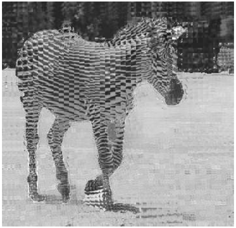
Feature-Specific Considerations

- Orientation is periodic
- Gain field ($\underline{g}$) assumed to be smooth in log domain
- Gain and bias fields coupled in bias feature
  - Hard to estimate jointly
  - Estimate $\underline{g}$ via E and treat as point estimate in $\underline{\mu}$

$$\underline{\mu}/\underline{g} = \underline{b} + \underline{\mu}^*$$

Intrinsic Feature Visualization

Once the smooth fields are estimated, we remove their effects to obtain the intrinsic features.



Outline

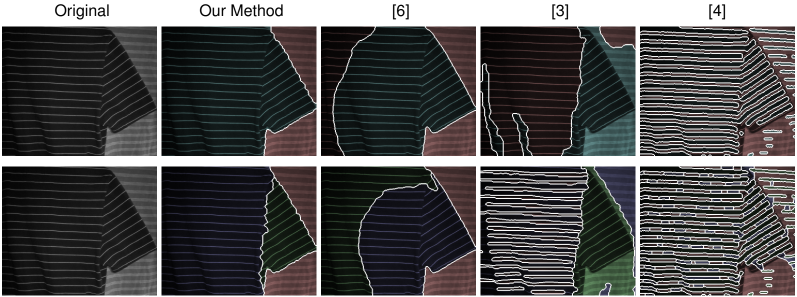
Image Segmentation

- Maximize $I(E, \mu, \theta, \eta; L)$ [6]
- Treat features as indepedent
- Estimate distributions using kernel density estimate

Algorithm Overview

1. Segment an image without regard to object boundaries
2. Refine the segmentation considering boundary effects
3. Estimate smooth fields conditioned on segmentation
4. Repeat Steps 1-3 until convergence

An Illustrative Example



Our features are invariant to smooth changes, but able to distinguish abrupt changes.

Image Segmentation Examples

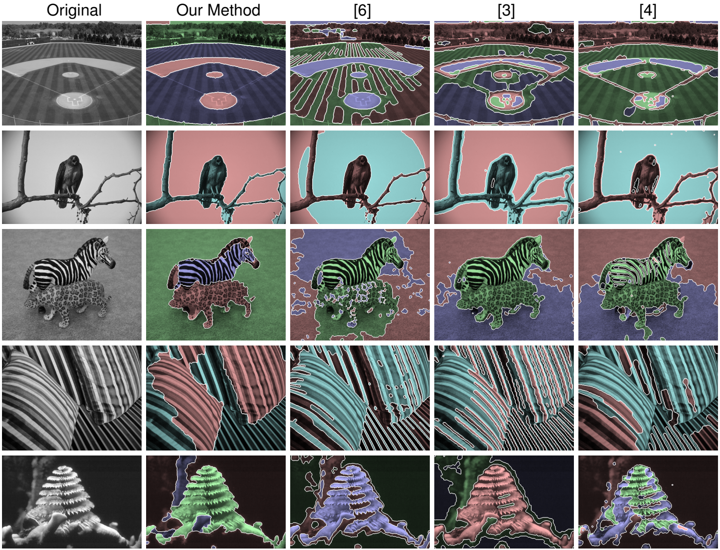
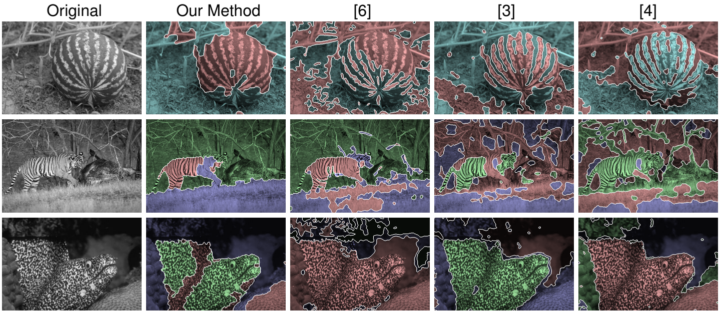


Image Segmentation Examples

Model Errors

Our appearance model does not always hold in natural images. Consider the following examples.



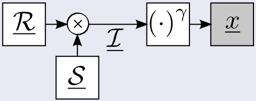
Outline

Estimating the Nonlinear Camera Response

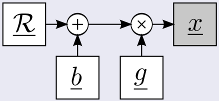
- Cameras typically have a nonlinear intensity response
- Linear imaging is desired for some common computer vision tasks

Model Similarities

A simple camera model



Our smooth MRF model



- If there was no γ correction (i.e. $\gamma = 1$), the models are equivalent when $\underline{b} = \underline{0}$ and $\underline{\mathcal{S}} = \underline{g}$.

Gamma Estimation

Algorithm Overview

- Given a γ, estimate $\underline{g}(\gamma)$, $\underline{b}(\gamma)$, $\underline{\mathcal{R}}(\gamma)$, image:

$$\underline{\mathcal{R}}(\gamma) = \frac{\underline{x}^{1/\gamma}}{\underline{g}(\gamma)} - \underline{b}(\gamma)$$

- Reconstruct the image without a bias field

$$\hat{\underline{x}} = \left(\underline{\mathcal{R}}(\gamma) \cdot \underline{g}(\gamma)\right)^{\gamma}$$

- Find the γ that minimizes reconstruction error using golden section search [5]

$$\gamma^* = \arg\min_{\gamma} \|\hat{\underline{x}} - \underline{x}\|_1$$

Gamma Estimation Results

Various Scenes

Generating the Data

Each scene was photographed using linear imaging ($\gamma = 1$) and post-processed with nine different γ values. One set of γ values is shown below.

Gamma Estimation Results

Image	Our RMSE	RMSE of [2]	Our $\|\|\hat{\gamma}-\gamma\|\|_1$	$\|\|\hat{\gamma}-\gamma\|\|_1$ of [2]
Bookshelf	3.673	46.640	0.025	0.431
Glass Ceiling	4.771	37.468	0.040	0.302
Bricks & Wood	5.381	39.411	0.053	0.288
Wood Cabinet	7.753	45.579	0.076	0.448
Keyboard	8.938	40.228	0.075	0.327
Floor Squares	13.028	41.607	0.176	0.343
Chair	17.516	29.170	0.296	0.236
Railing	18.262	20.401	0.231	0.156
Mean	9.915	37.563	0.122	0.316

Gamma Estimation Results

Analysis of Smoothly Varying Textures

Jason Chang

Introduction
Problem Statement

Texture Modelling
Motivation
Feature Extraction
Boundary Effects
Spatial Smoothness

Applications
Segmentation
Gamma Estimation
Shading Reflectance

Contributions

Image	Our RMSE	RMSE of [2]	Our $\|\hat{\gamma} - \gamma\|_1$	$\|\hat{\gamma} - \gamma\|_1$ of [2]
Bookshelf	3.673	46.640	0.025	0.431
Glass Ceiling	4.771	37.468	0.040	0.302
Bricks & Wood	5.381	39.411	0.053	0.288
Wood Cabinet	7.753	45.579	0.076	0.448
Keyboard	8.938	40.228	0.075	0.327
Floor Squares	13.028	41.607	0.176	0.343
Chair	17.516	29.170	0.296	0.236
Railing	18.262	20.401	0.231	0.156
Mean	9.915	37.563	0.122	0.316

Outline

Shading and Reflectance Decomposition

- $\mathcal{I} = \mathcal{R} \times \mathcal{S}$
- Smooth MRF estimation of shading
  - No bias field $\Rightarrow$ shading affects contrast and bias
  - MRF estimation for a set of parameters

$$\underline{f}^{(k+1)} = \sum_{\theta \in \Theta} F_\theta^{-1} \left(\underline{\theta} - w_p^\theta \left(\underline{\theta} - \underline{f}^{(k)} \right) \right)$$

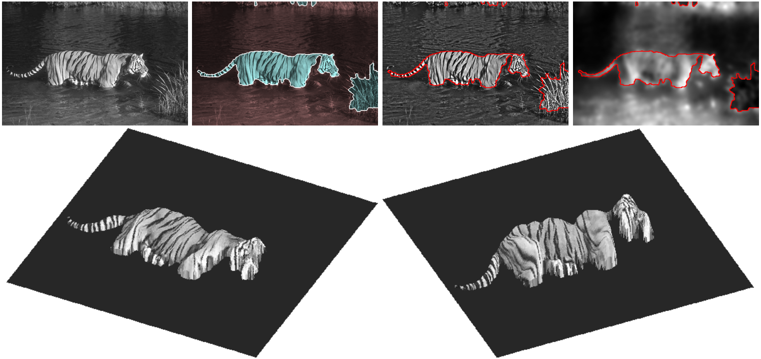
$$\text{DC Gain}\left(F_\theta^{-1}\right) = \frac{\prod_{\theta_1 \neq \theta} h_{\theta_1}^2}{\sum_{\theta_1 \in \Theta} \prod_{\theta_2 \neq \theta_1} h_{\theta_2}^2}$$

Shading and Reflectance Decomposition

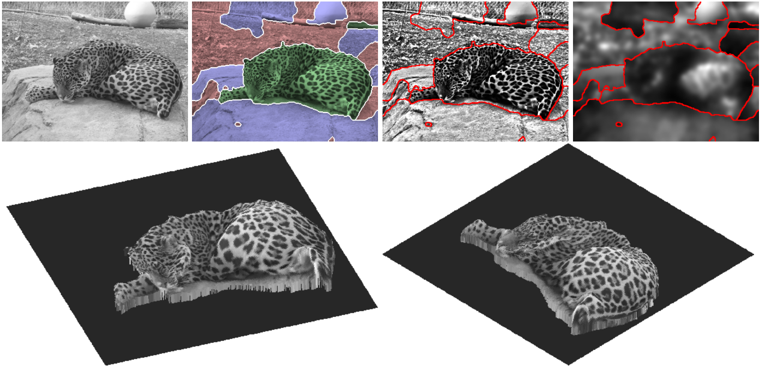
Algorithm Overview

1. Segment the image
2. Estimate the γ factor of the camera and obtain the irradiance image
3. Estimate the smooth shading image from the irradiance image
4. Estimate the shape from shading using [9]

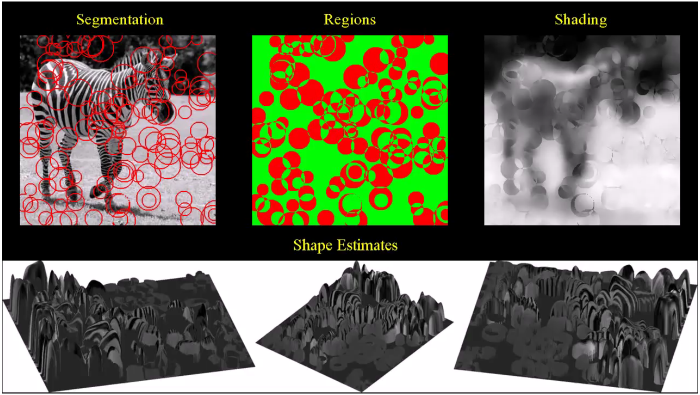
Shading and Reflectance Results



Shading and Reflectance Results



Shading and Reflectance Results



movies/segmentation-shading.mp4

Contributions

- Developed a texture model that captures scale, orientation, contrast, and bias
- Modelled smooth spatial changes in features
- Achieved robust texture segmentation, estimation of an unknown camera response, and shading/reflectance decomposition

Possible Future Directions

- Better boundary effect handling
- Probabilistic feature measurements
- Using the shading / shape estimation to improve segmentaiton
- Speed improvements

References I

P. Brodatz.
Textures: A Photographic Album for Artists and Designers.
Dover Publications Inc., 1966.

H. Farid.
Blind inverse gamma correction.
Image Processing, IEEE Transactions on, 10(10):1428–1433, Oct 2001.

M. Heiler and C. Schnorr.
Natural image statistics for natural image segmentation.
Computer Vision, 2003. Proceedings Ninth IEEE International Conference on, pages 1259–1266 vol.2, Oct. 2003.

N. Houhou, J.-P. Thiran, and X. Bresson.
Fast texture segmentation model based on the shape operator and active contour.
In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, June 2008.

J. Kiefer.
Sequential minimax search for a maximum.
Proceedings of the American Mathematical Society 4, pages 502–506, 1953.

J. Kim, I. Fisher, J.W., A. Yezzi, M. Cetin, and A. Willsky.
A nonparametric statistical method for image segmentation using information theory and curve evolution.
Image Processing, IEEE Transactions on, 14(10):1486–1502, Oct. 2005.

References II

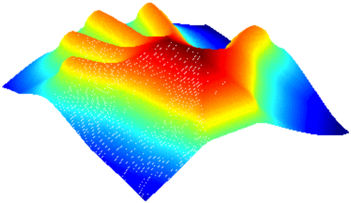
E. Parzen.
On estimation of a probability density function and mode.
The Annals of Mathematical Statistics, 33(3):1065–1076, 1962.

E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger.
Shiftable multi-scale transforms.
Information Theory, IEEE transactions on, 38(2), 1992.

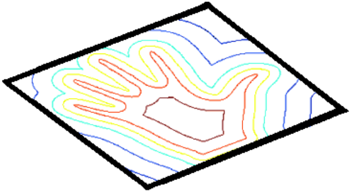
P. sing Tsai and M. Shah.
Shape from shading using linear approximation.
Image and Vision Computing, 12:487–498, 1994.

Level Set Methods

- Implicitly define the curve within a 3D surface
- define a height at every pixel in the image



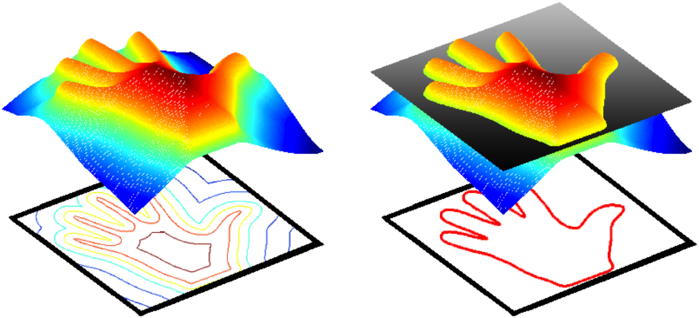
The Surface φ



The Level Sets of φ

Level Set Methods

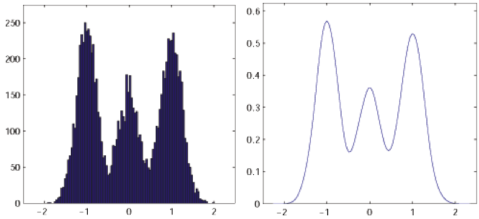
- The zero level set of φ implicitly represents the 2D curve
- Variational calculus is used to perform gradient descent on some energy functional



Kernel Density Estimate

$$p_X(x) \approx \frac{1}{Nh} \sum_{s=1}^{N} K\left(\frac{x - x_s}{h}\right)$$

$$K(x) = -\frac{1}{\sqrt{2\pi}} \exp\left[-x^2\right]$$



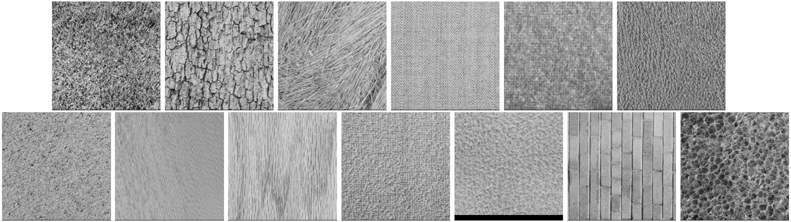
Optimizing Mutual Information

$$
\begin{aligned}
&\arg\max_L |\Omega|\, I\left(E,\mu,\theta,\eta;L\right) - \alpha \oint_{\mathcal{C}} ds \\
&= \arg\max_L |\Omega| \left[H\left(E,\mu,\theta,\eta\right) - H\left(E,\mu,\theta,\eta|L\right)\right] - \alpha \oint_{\mathcal{C}} ds \\
&= \arg\max_L -|\Omega|\, H\left(E,\mu,\theta,\eta|L\right) - \alpha \oint_{\mathcal{C}} ds \\
&= \arg\max_L -|\Omega| \sum_{\ell\in L} p_L\left(\ell\right) H\left(E,\mu,\theta,\eta|L=\ell\right) - \alpha \oint_{\mathcal{C}} ds \\
&\approx \arg\max_L -|\Omega| \sum_{\ell\in L} \frac{|R^\ell|}{|\Omega|} \frac{1}{|R^\ell|} \int_{R^\ell} \log p\left(E_i,\mu_i,\theta_i,\eta_i|\ell\right) di - \alpha \oint_{\mathcal{C}} ds \\
&= \arg\max_L -\sum_{\ell\in L} \int_{R^\ell} \log p_E^\ell\left(E_i\right) p_\mu^\ell\left(\mu_i\right) p_\theta^\ell\left(\theta_i\right) p_\eta^\ell\left(\eta_i\right) di - \alpha \oint_{\mathcal{C}} ds
\end{aligned}
$$

Level-set Gradient Descent of $I\left(X;L\right)$

$$\frac{\partial \varphi_i}{\partial t} = \log \frac{p_X^+\left(x_i\right)}{p_X^-\left(x_i\right)} - \alpha \kappa_i, \ \forall i \in \mathcal{C}$$

Brodatz Classification [1]



- 100% correct classification on Brodatz textures
- Able to segment Brodatz mosaics