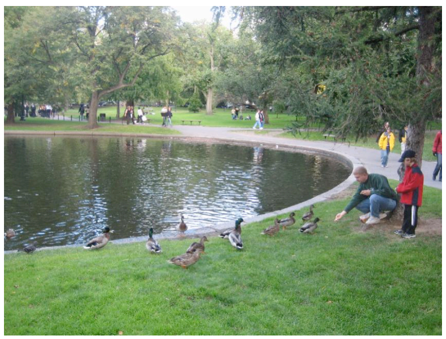
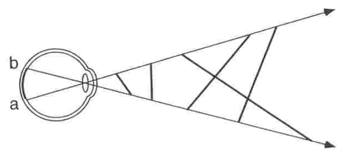
6.869 Advances in Computer Vision Spring 2010

http://people.csail.mit.edu/torralba/courses/6.869/6.869. computervision.htm

Lecture 16
3D

projections





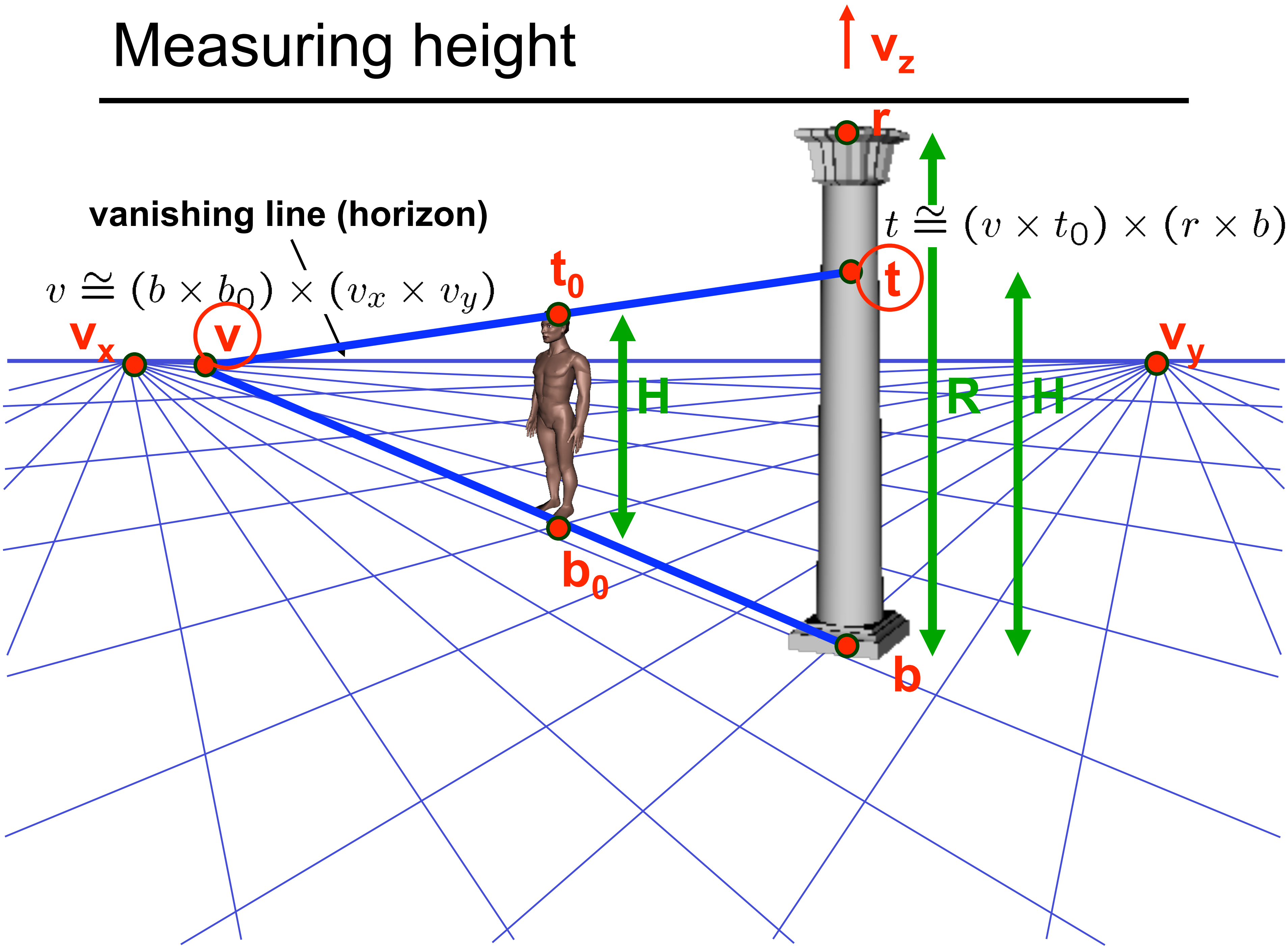
?

3D from pixel values

D. Hoiem, A.A. Efros, and M. Hebert, "Automatic Photo Pop-up”. SIGGRAPH 2005.

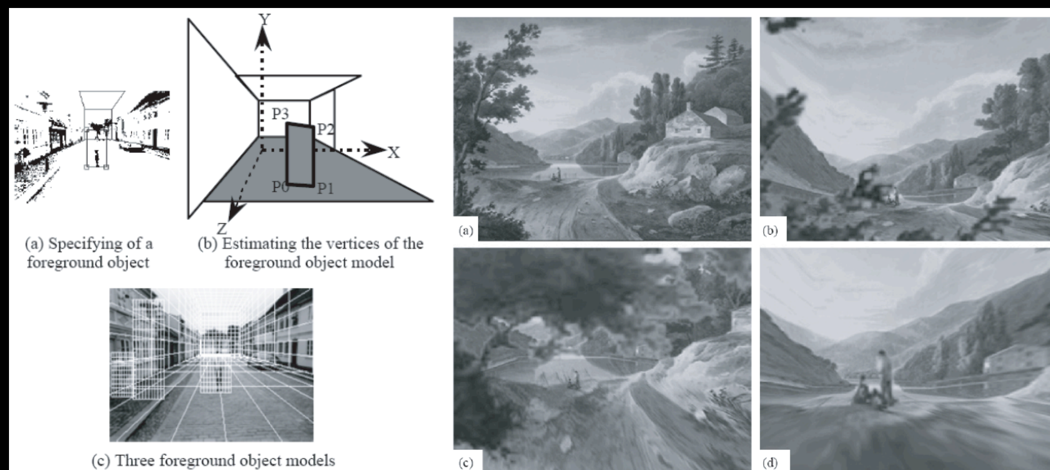
A. Saxena, M. Sun, A. Y. Ng. "Learning 3-D Scene Structure from a Single Still Imag
In ICCV workshop on 3D Representation for Recognition (3dRR-07), 2007.

Measuring height

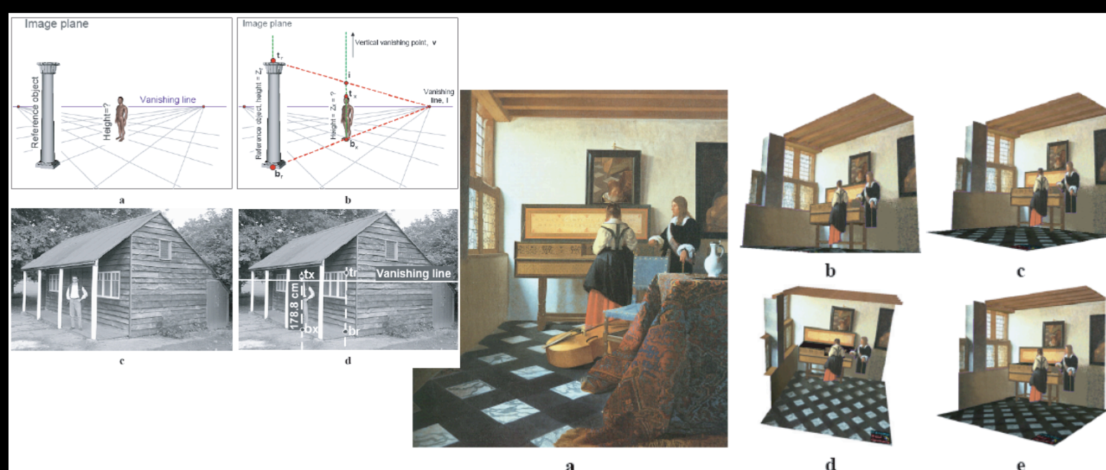


Humans label cues for 3D

Y. Horry, K.I. Anjyo and K. Arai. "Tour Into the Picture: Using a spidery mesh user interface to make animation from a single image". ACM SIGGRAPH 1997



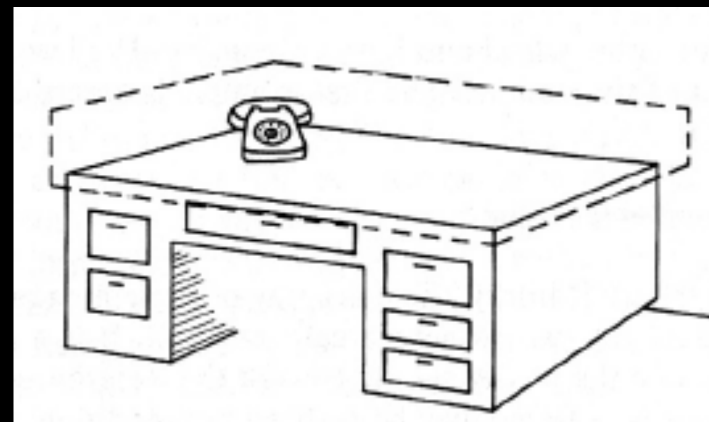
A. Criminisi, I. Reid, and A. Zisserman. "Single View Metrology". ICCV, Kerkyra, Greece, 1999.



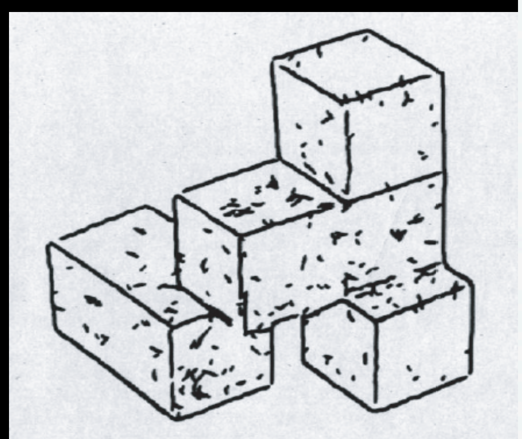
Reasoning about spatial relationships between objects

1. LEFT OF
2. RIGHT OF
3. BESIDE (alongside, next to)
4. ABOVE (over, higher than, on top of)
5. BELOW (under, underneath, lower than)
6. BEHIND (in back of)
7. IN FRONT OF
8. NEAR (close to, next to?)
9. FAR
10. TOUCHING
11. BETWEEN
12. INSIDE (within)
13. OUTSIDE

Freeman, 1974



Ballard & Brown, 1982



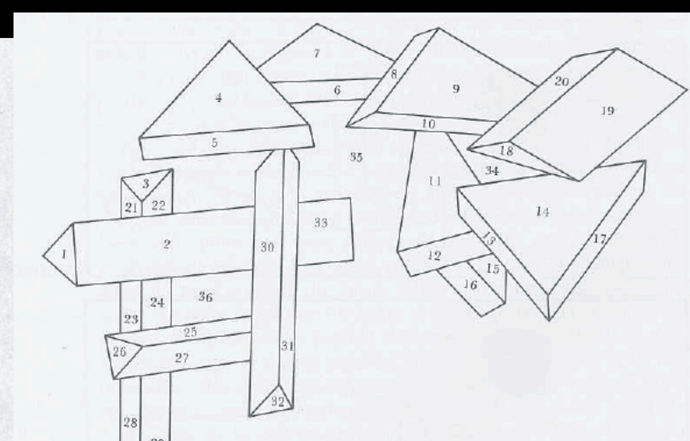


FIGURE 1-20

From Guzmán (1969).

Guzman, 1969

Tool went online July 1st, 2005
250,000 object annotations
Labelme.csail.mit.edu

B. Russell, A. Torralba, W.T. Freeman. IJCV 2008

Polygon quality

Testing

Most common labels:

test

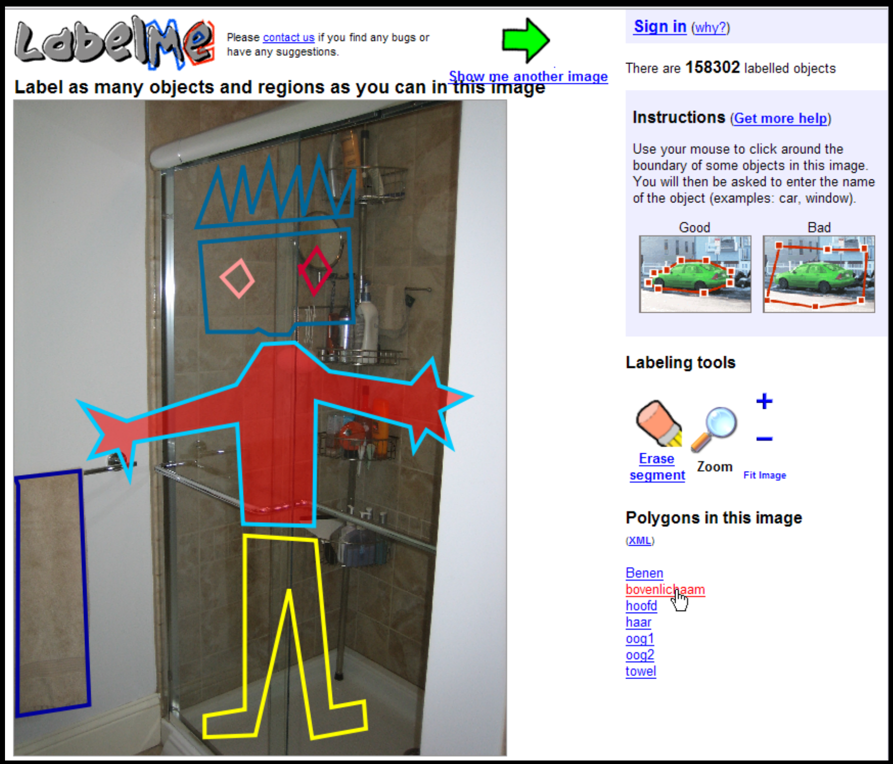
adksdsa

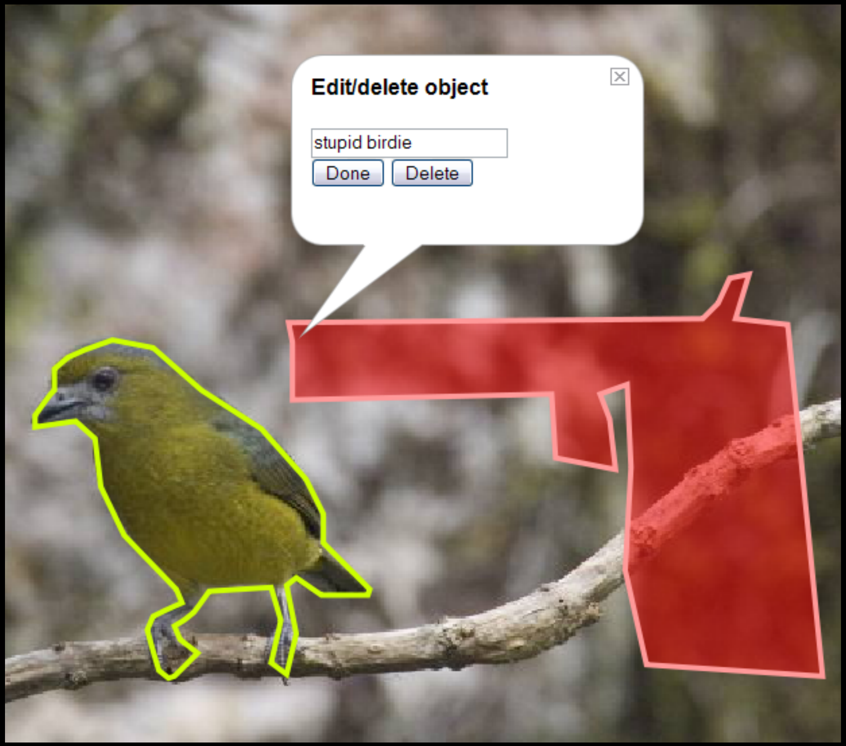
woiieiie

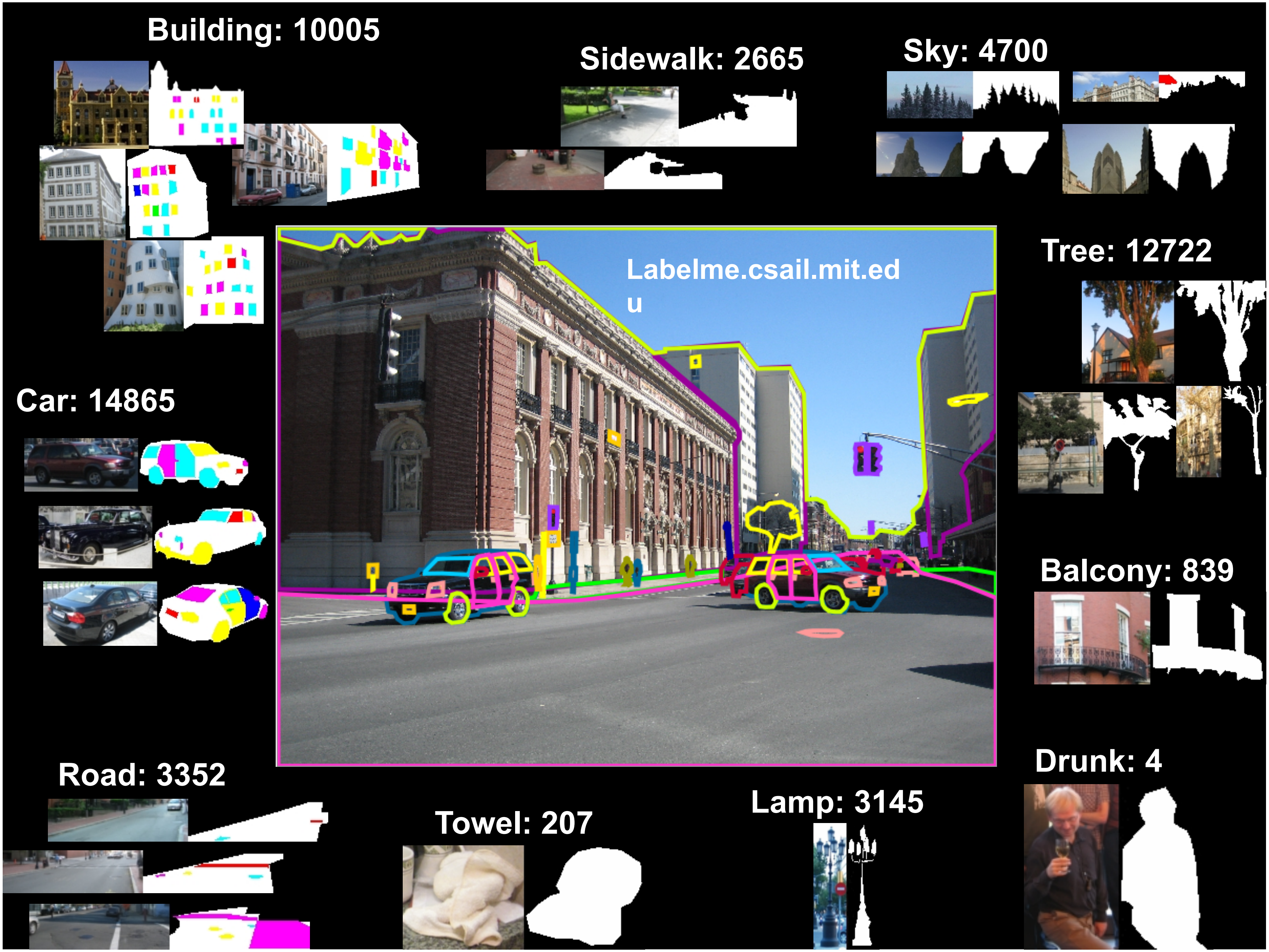
…

Online Hooligans

Do not try this at home






Building: 10005
Sidewalk: 2665
Sky: 4700
Labelme.csail.mit.ed
u
Tree: 12722
Car: 14865
Balcony: 839
Drunk: 4
Road: 3352
Lamp: 3145
Towel: 207

Overlapping segments

(tree – building)

Transparent and wiry objects

Key idea: analyze overlap statistics of labeled objects

(Car – door)

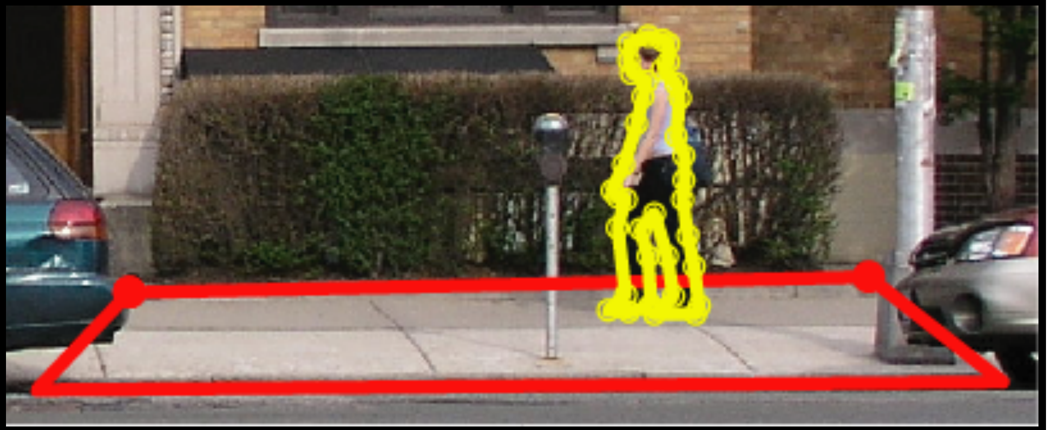
Object – parts relations

(Car – road)

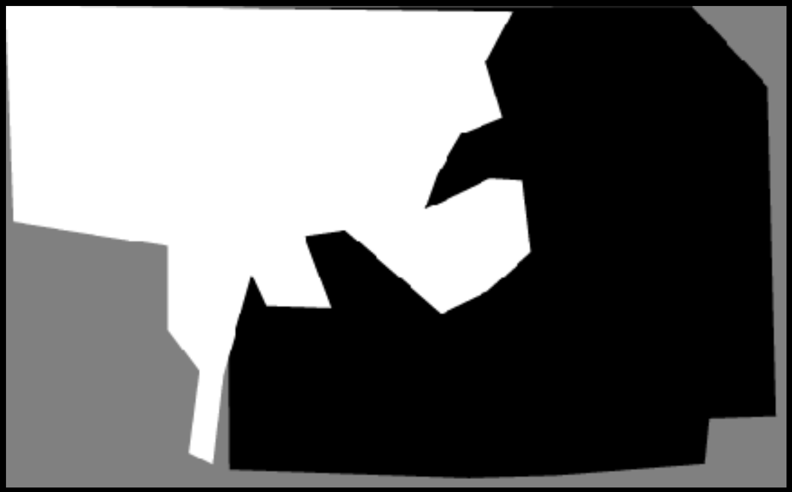
Completed objects behind occlusions

- **Occlusion relations**
- **Support – object relations**

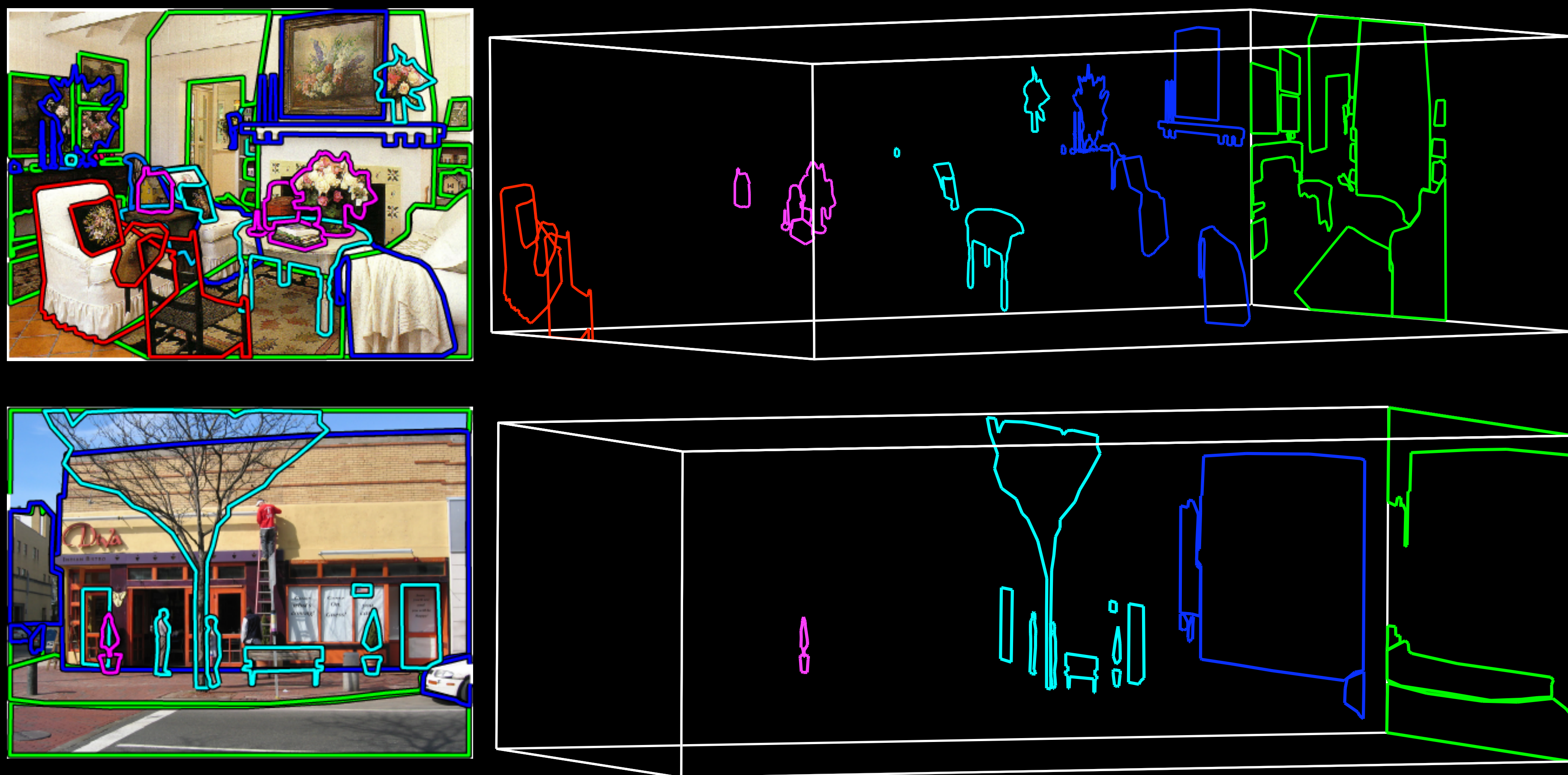
Depth ordering



The object on the foreground has more control points in the shared segment (95%)

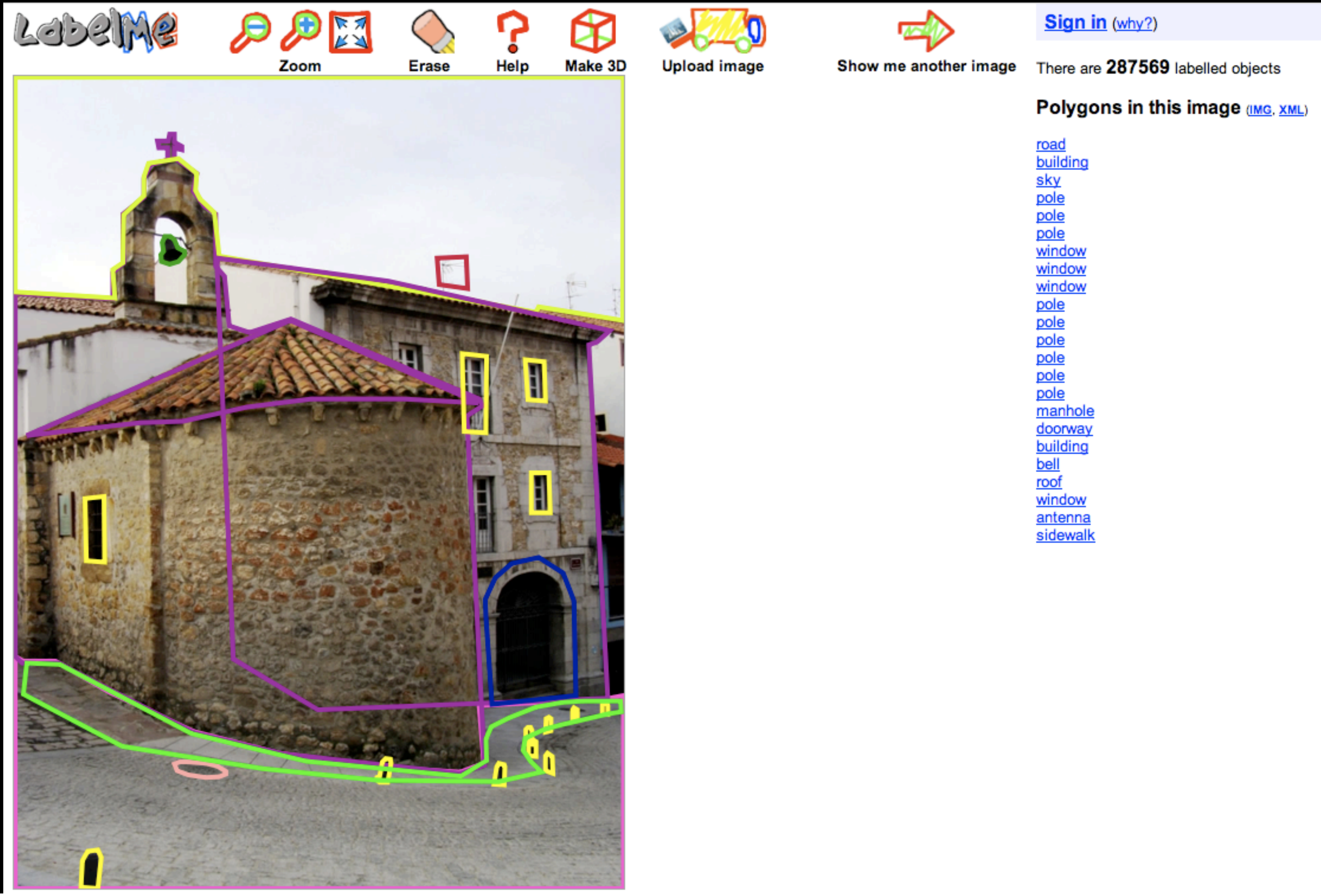


Depth ordering

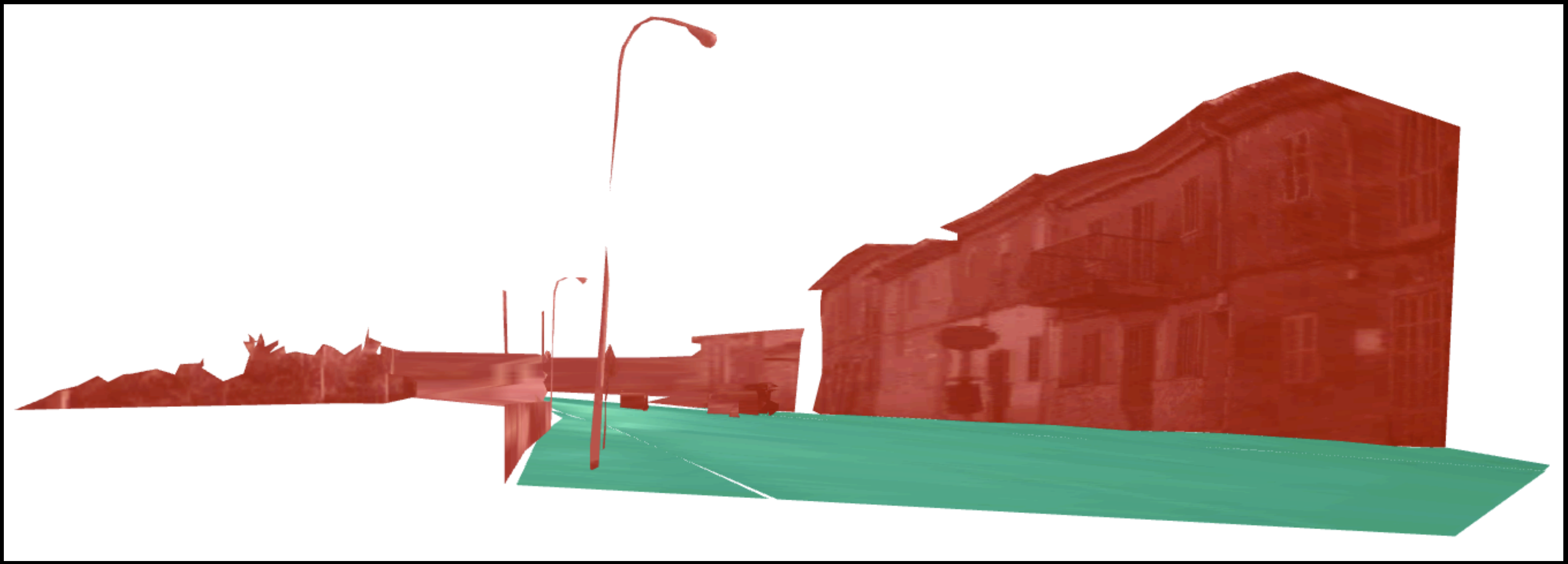


http://labelme.csail.mit.edu/LabelMeToolbox/index.html

How to infer the geometry of a scene?

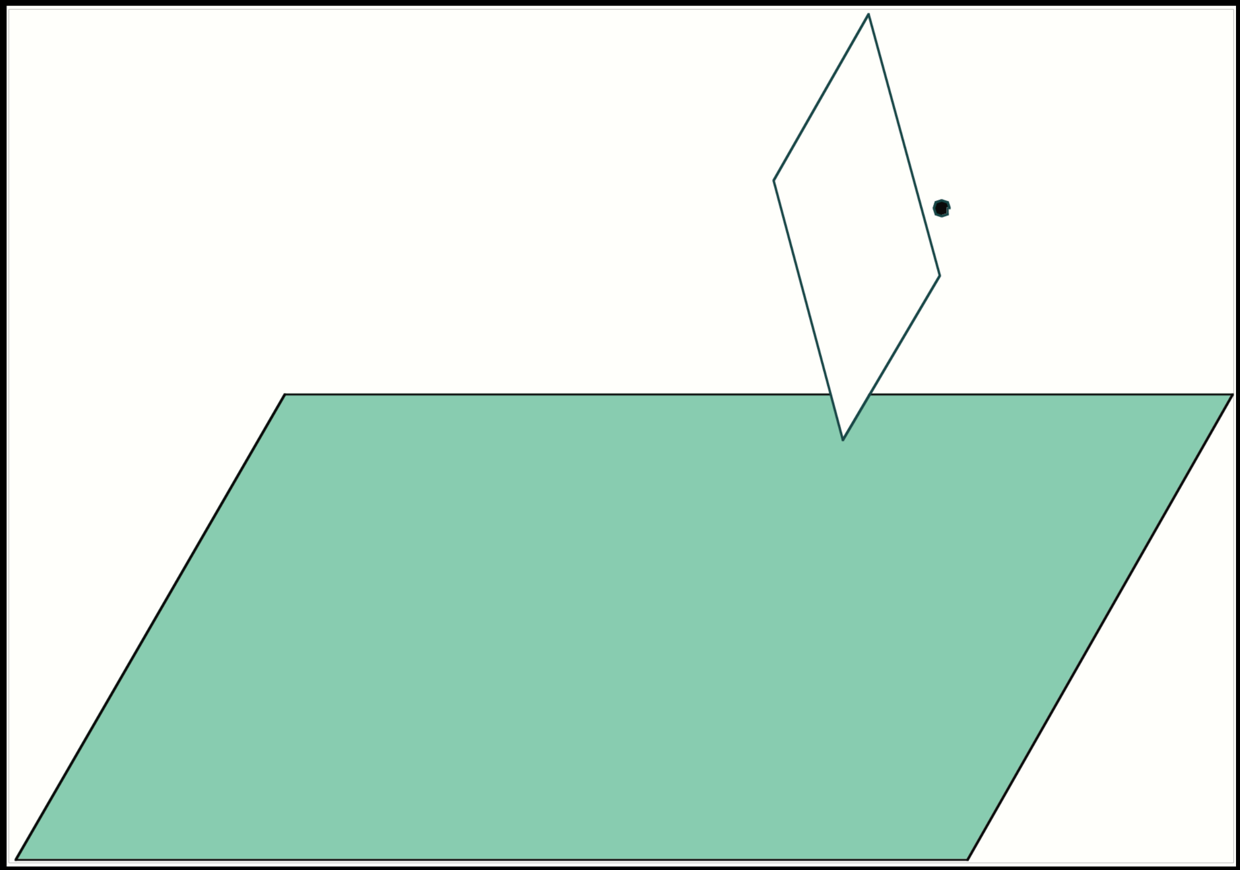


Scene layout assumptions



Assumption: objects stand on ground plane

Camera and ground



Camera and ground

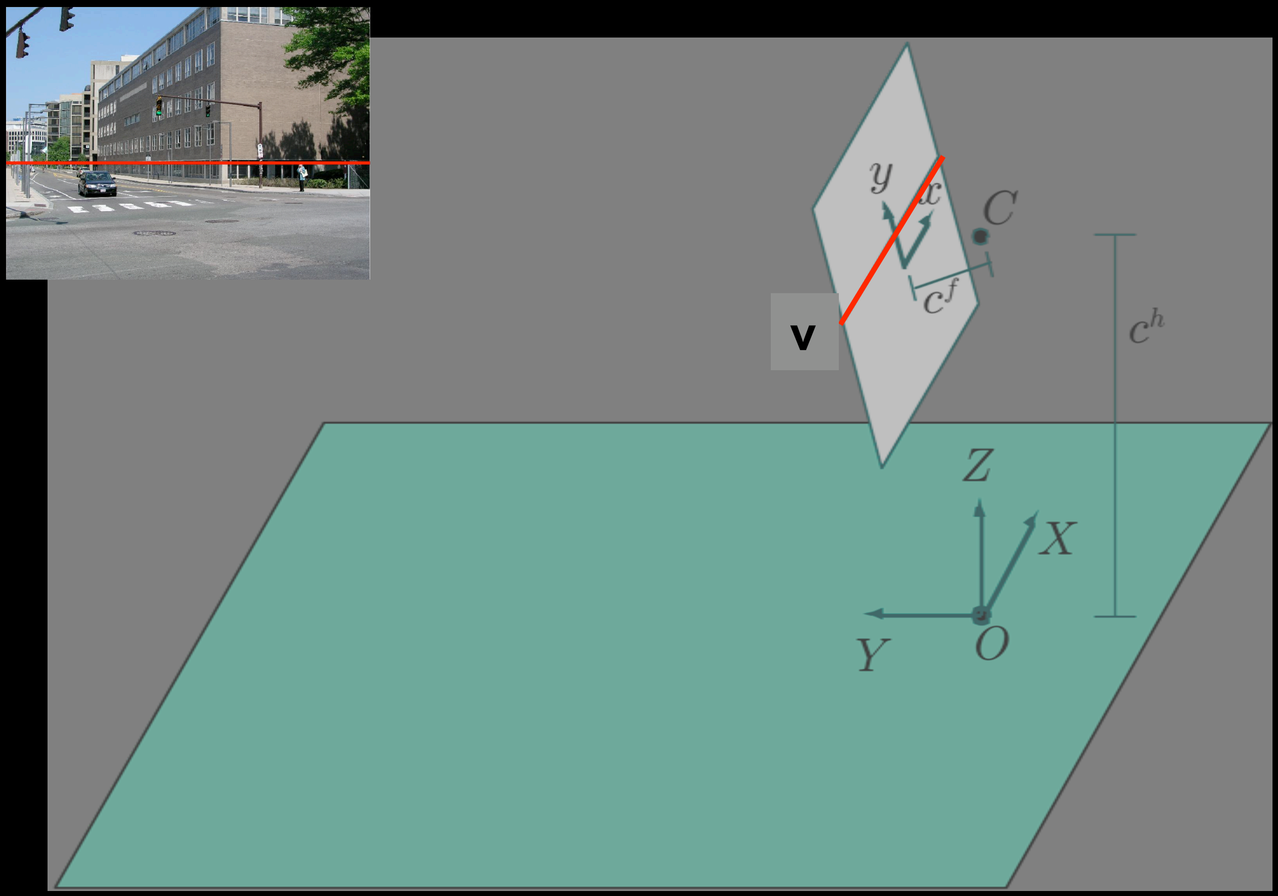
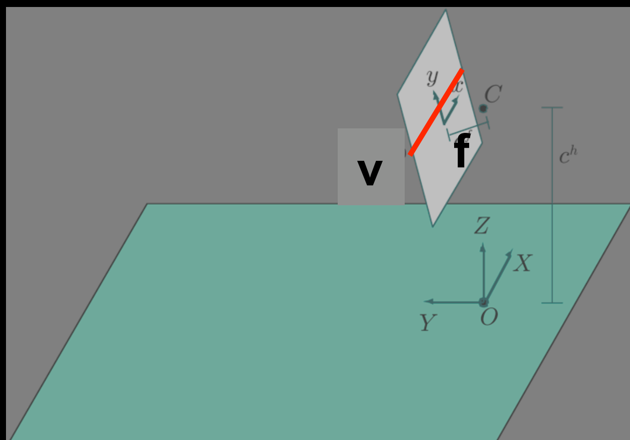


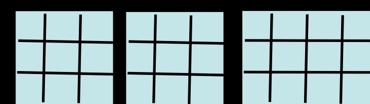
Image formation model



3D -> 2D

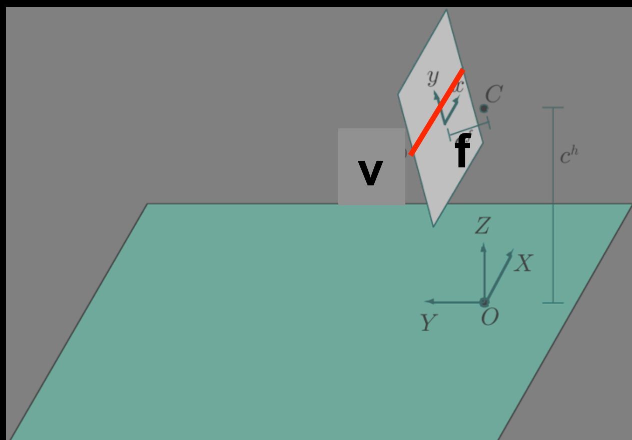
$\mathbf{X} = (X, Y, Z, 1)^T$ $\mathbf{x} = (x, y, 1)^T$

$$\mathbf{x} = \mathbf{P}\mathbf{X}$$

P= 

K R [I | -C]

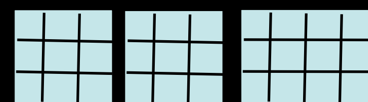
Image formation model



3D -> 2D

$\mathbf{X} = (X, Y, Z, 1)^T$ $\quad$ $\mathbf{x} = (x, y, 1)^T$

$$\mathbf{x} = \mathbf{P}\mathbf{X}$$

P= 

K **R** **[I | -C]**

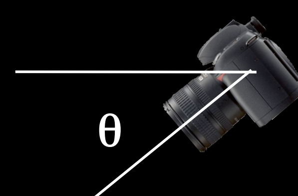
$\mathbf{C} = (0, 0, C_z)^T$

$$\mathbf{K} = \begin{pmatrix} \alpha_x f & s & p_x \\ 0 & \alpha_y f & p_y \\ 0 & 0 & 1 \end{pmatrix}$$

$$\mathbf{R} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix}$$

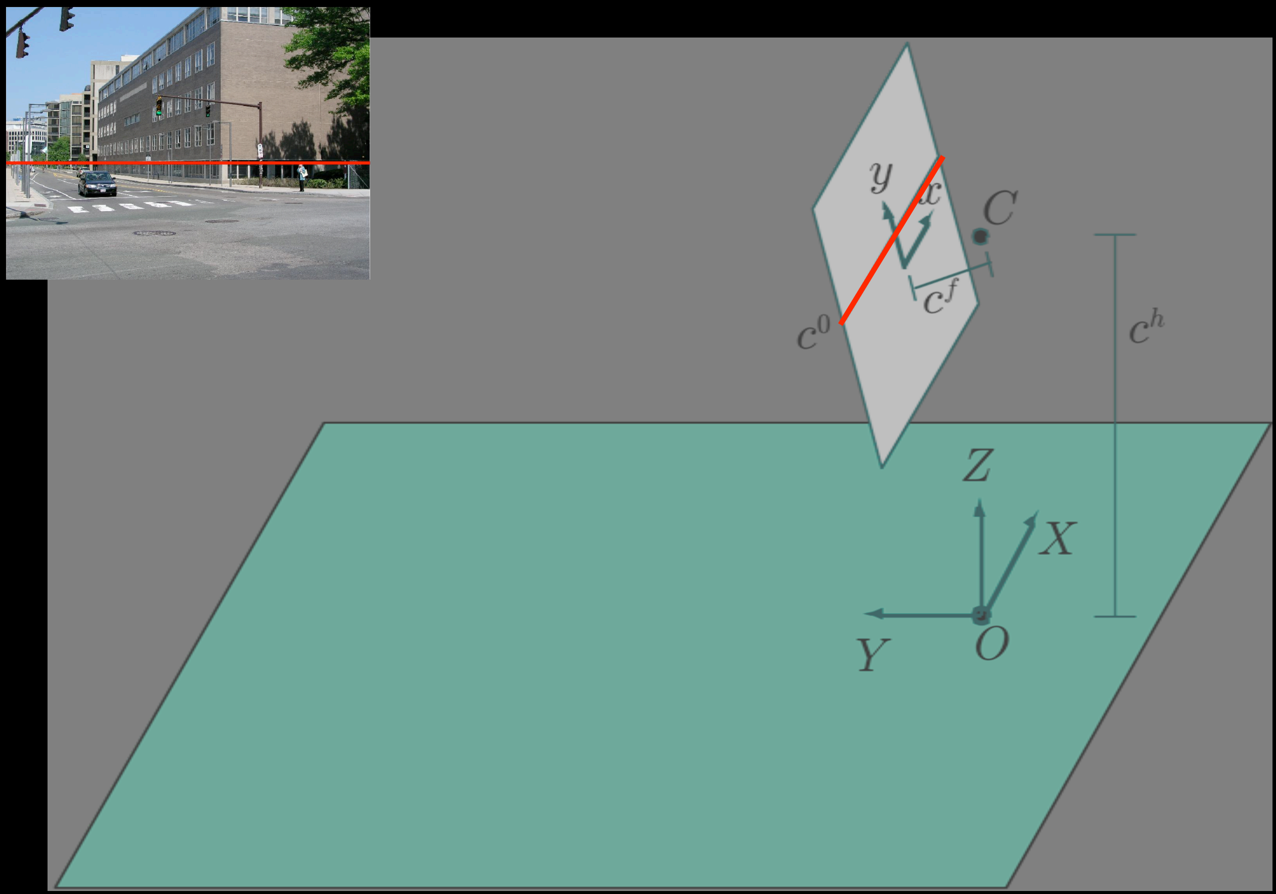
$$\tan\theta = \frac{v}{f}$$

f=focal length = ?
(ax,ay) = pixels size = (1,1)
s = skew = 0
(px,py) = principal point = (0,0) image center



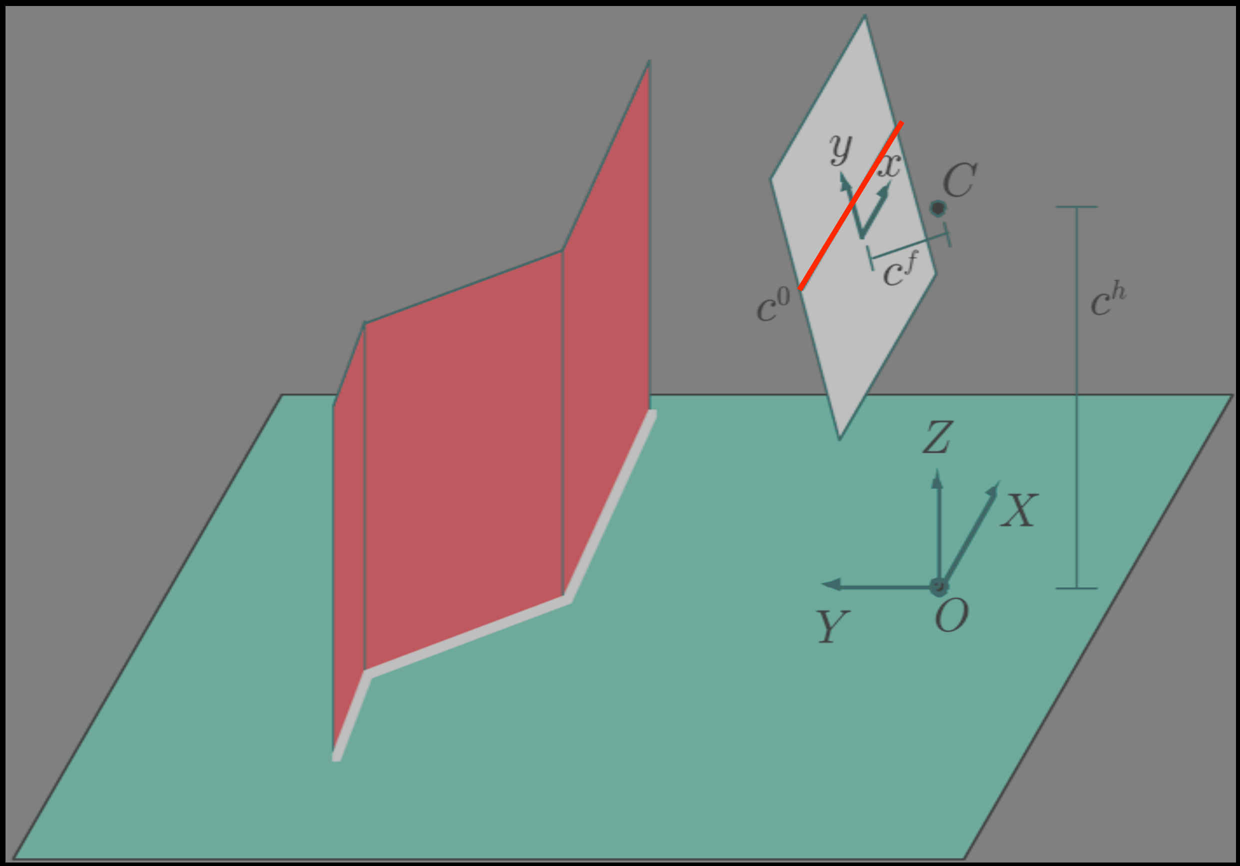
Unknowns: f (focal length), v (horizon line), Cz (camera height)

Camera and ground



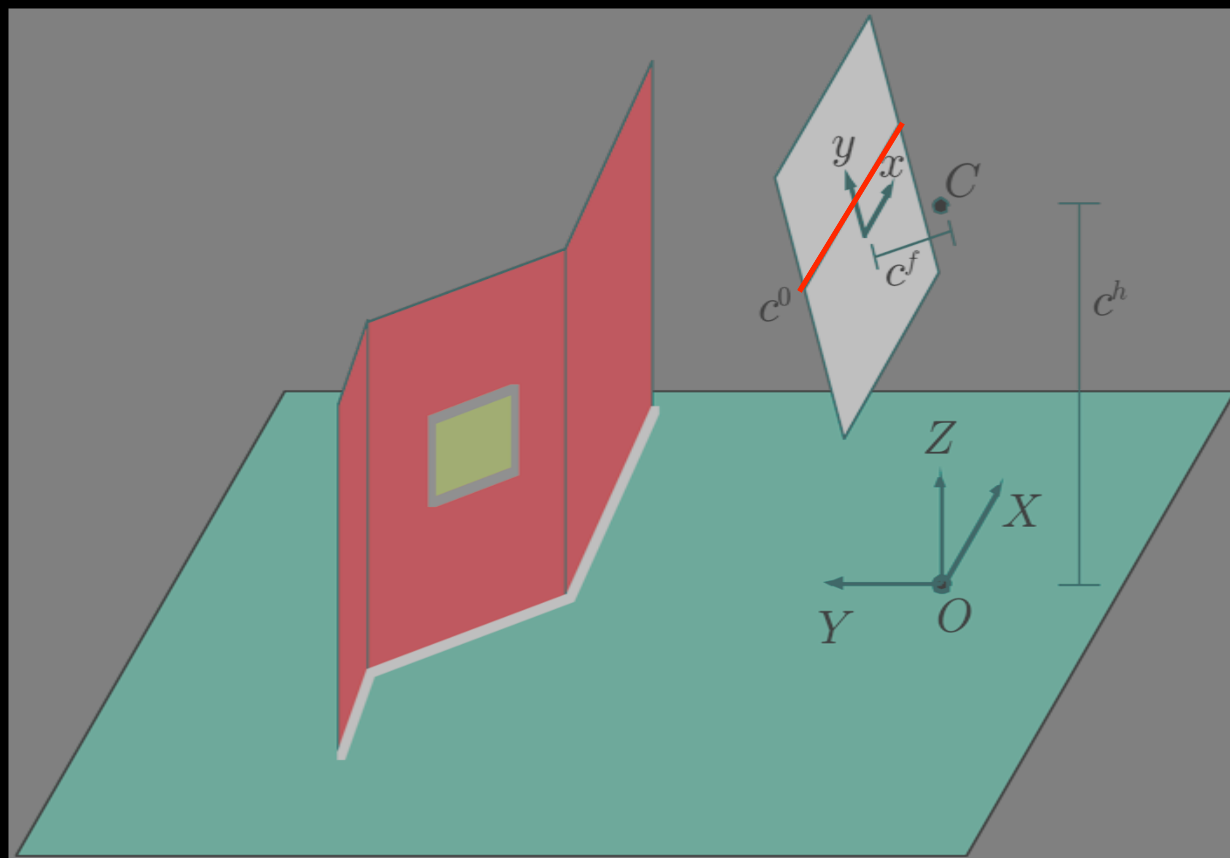
- Assume camera is held level with ground
- Camera parameters: camera height, horizon line, focal length
- Can relate ground and image planes via homography

Standing objects



- Standing objects represented by vertical piecewise-connected planes
- 3D coordinates on standing planes related to ground plane via the contact line

Attached objects



- 3D coordinates of attached objects determined by object it is attached to

Recovering scene geometry

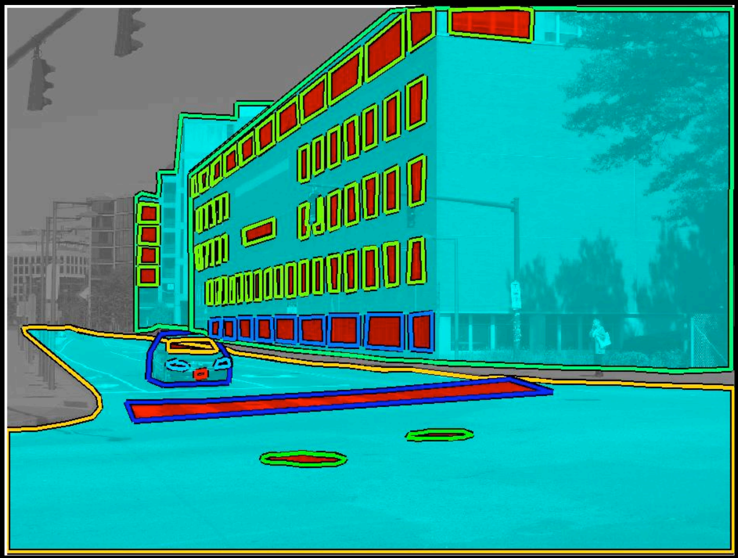
- Polygon types
  - Ground
  - Standing
  - Attached
- Edge types
  - Contact
  - Attached
  - Occluded
- Camera parameters

Recovering scene geometry

- Polygon types
  - Ground
  - Standing
  - Attached
- Edge types
  - Contact
  - Attached
  - Occluded
- Camera parameters

Relationships between polygons

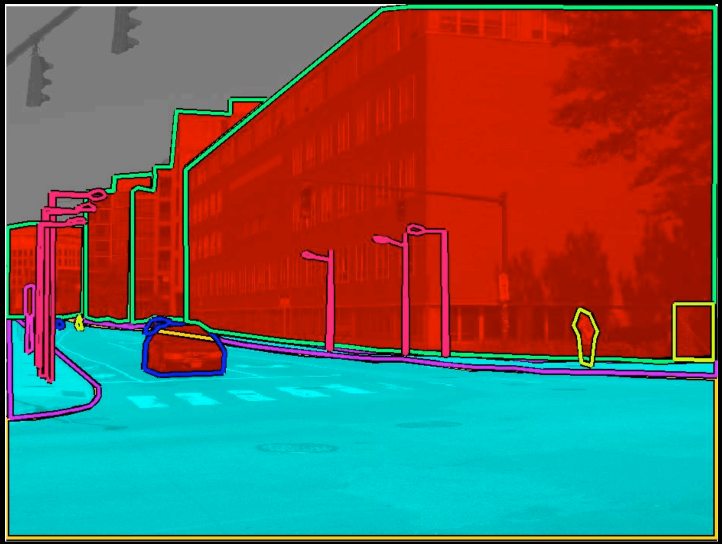
Part-of



Attached

Standing / Ground / Attached

Supported-by

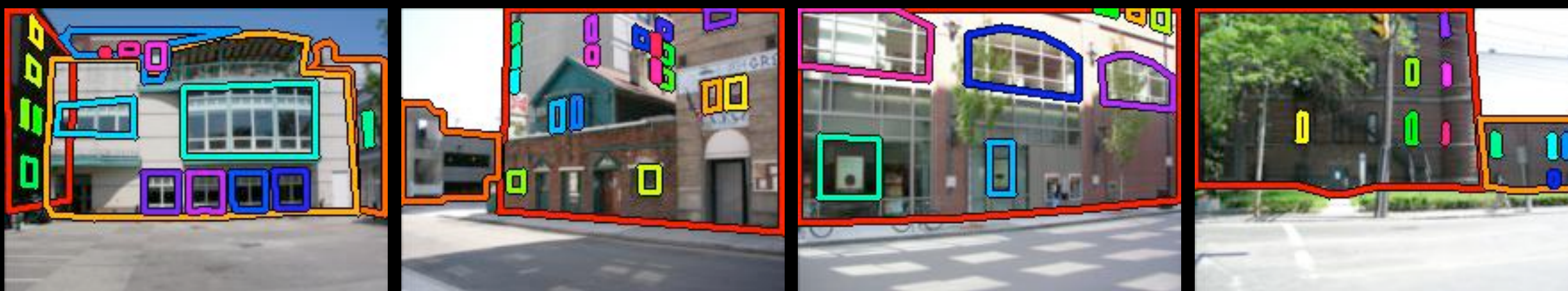


Standing

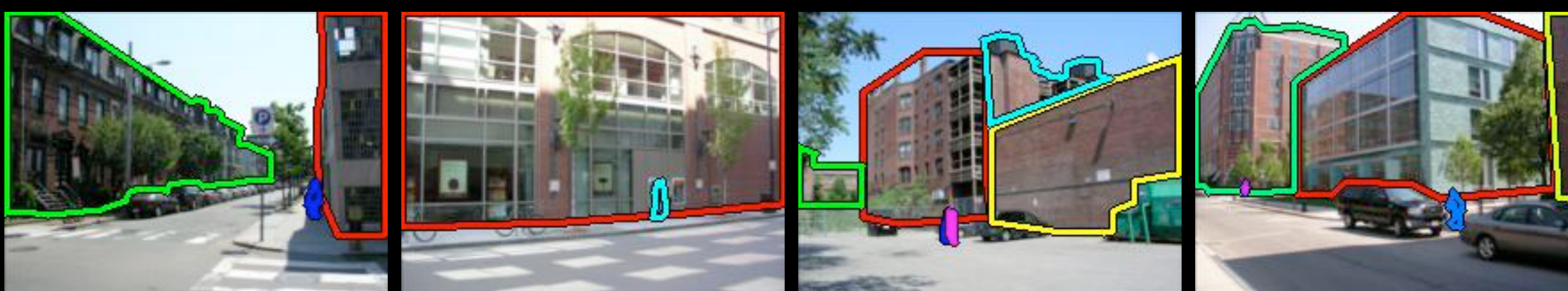
Ground

Cues for attachment relationships

1. Consistency of relationship across database



building, windows



building, person

Cues for attachment relationships

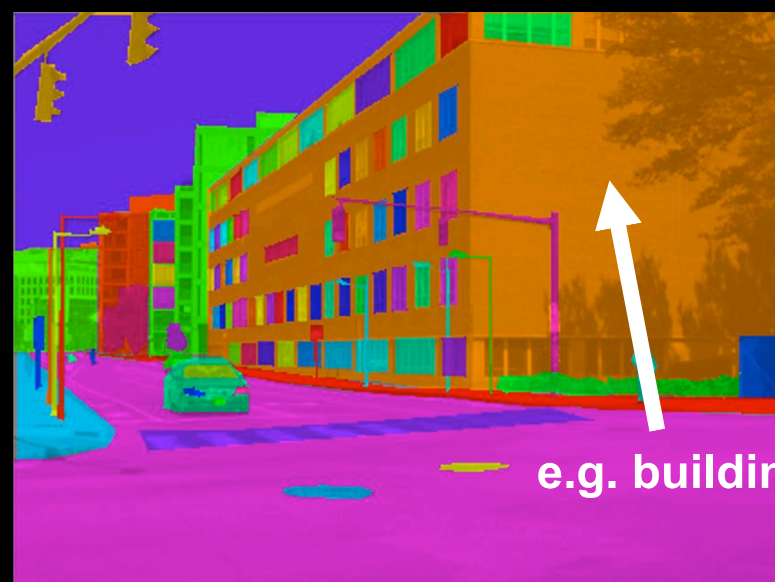
2. High relative overlap between part and object

$$\frac{\text{area}(\text{part} \cap \text{object})}{\text{area}(\text{part})}$$

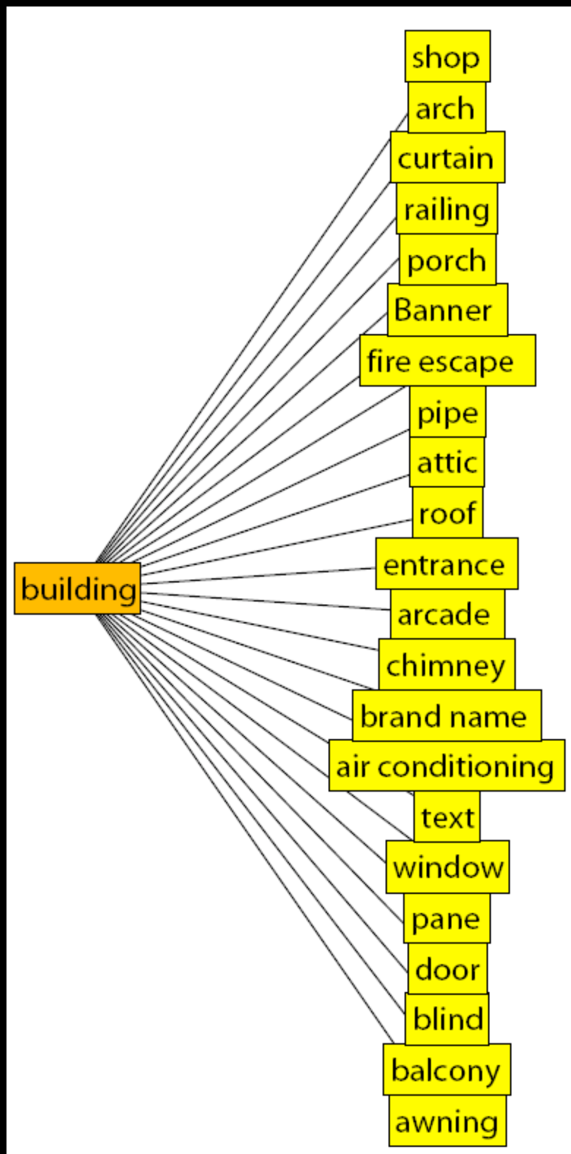


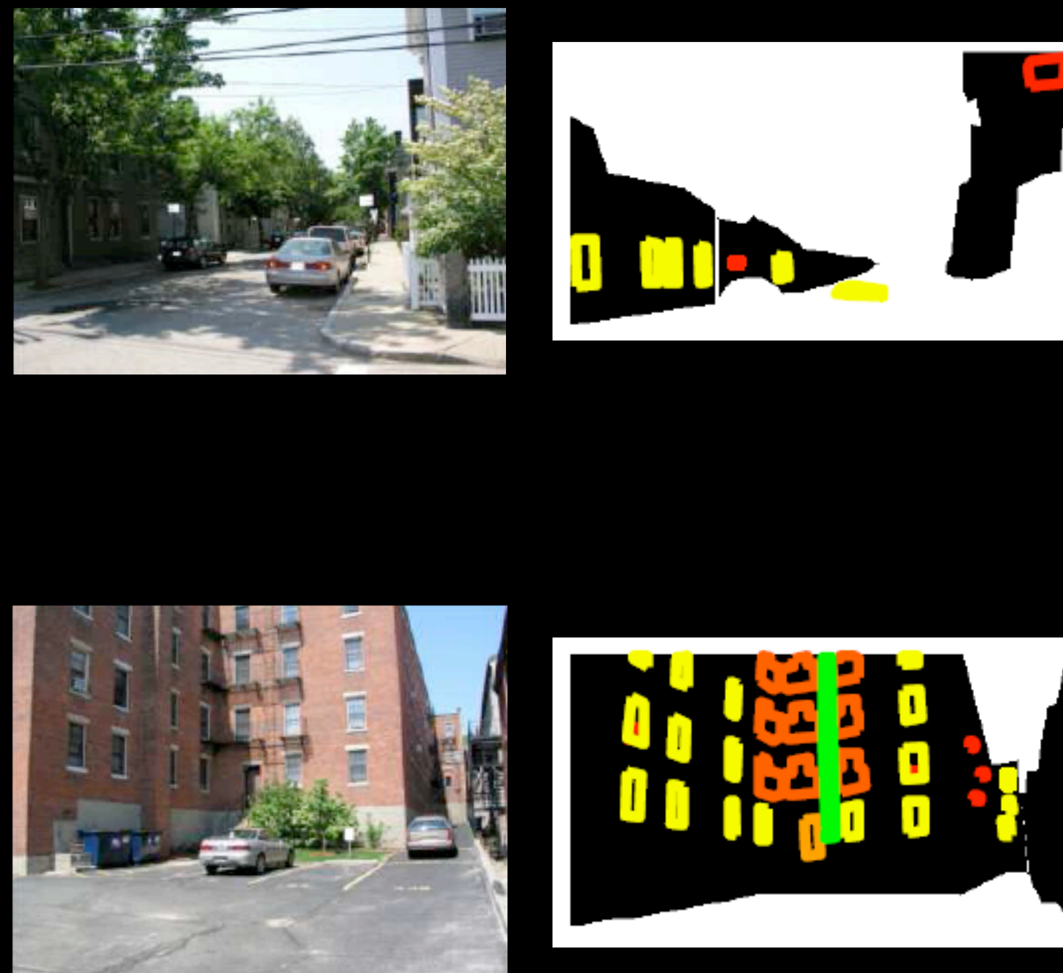
3. Probability of coincidental overlap

$$\frac{\text{area}(\text{object})}{\text{area}(\text{image})}$$

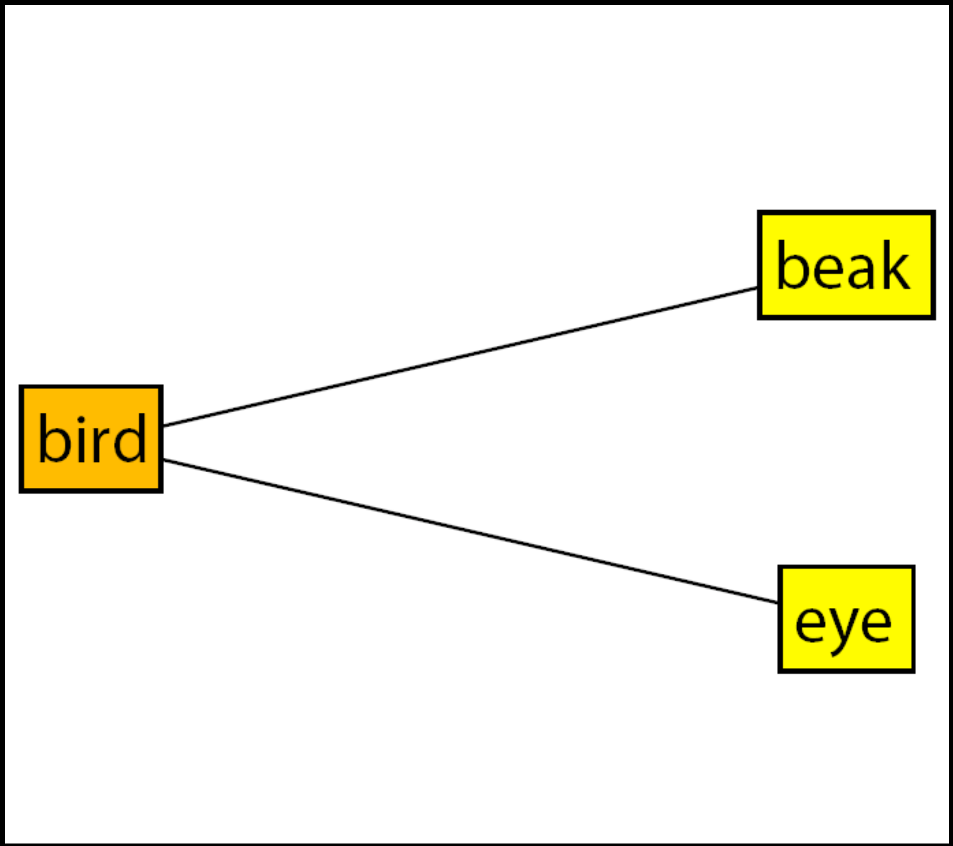


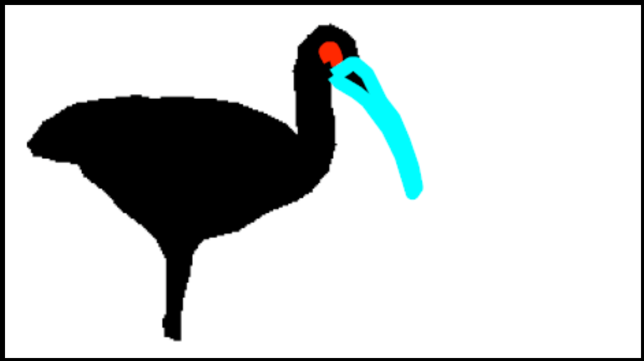
Learned/inferred attachment relationships





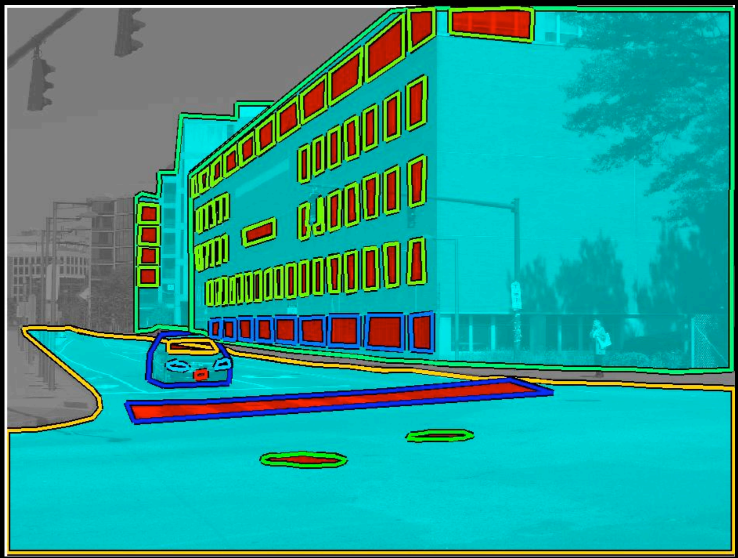
Learned/inferred attachment relationships





Relationships between polygons

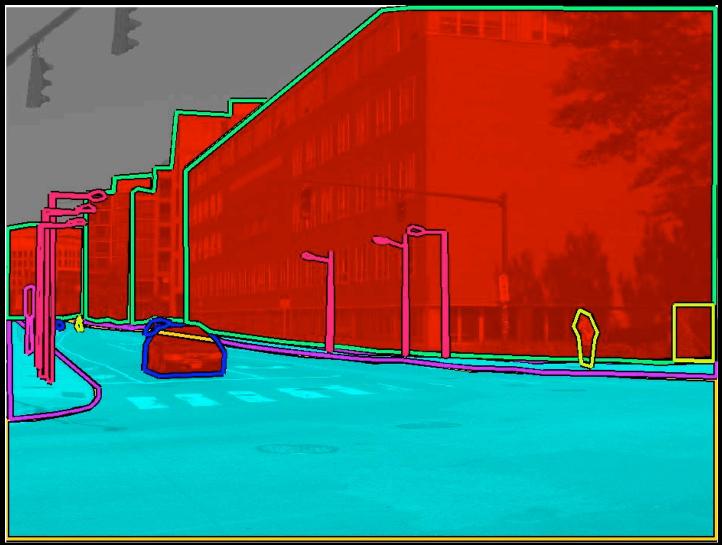
Part-of



Attached

Standing / Ground / Attached

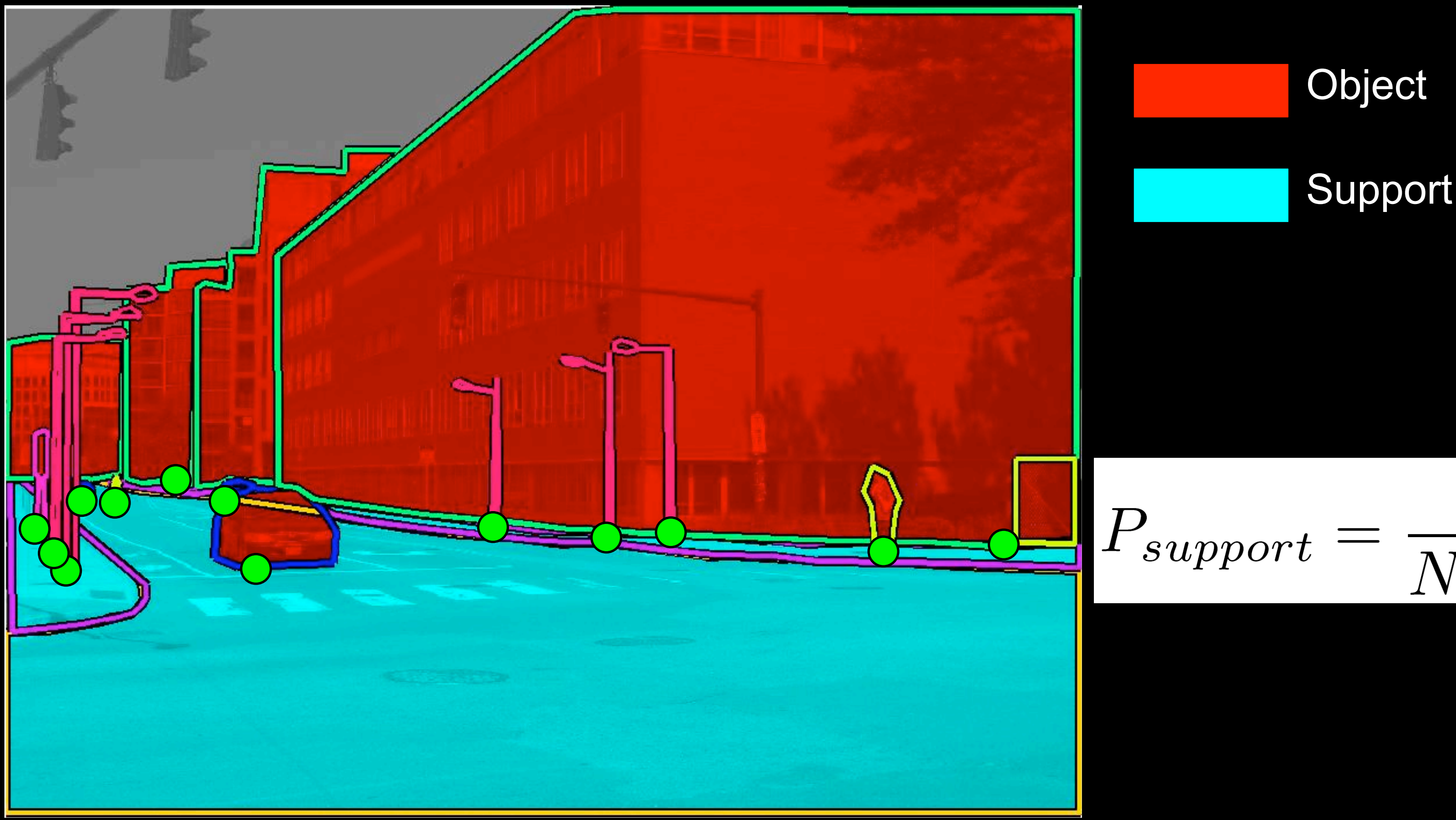
Supported-by



Standing

Ground

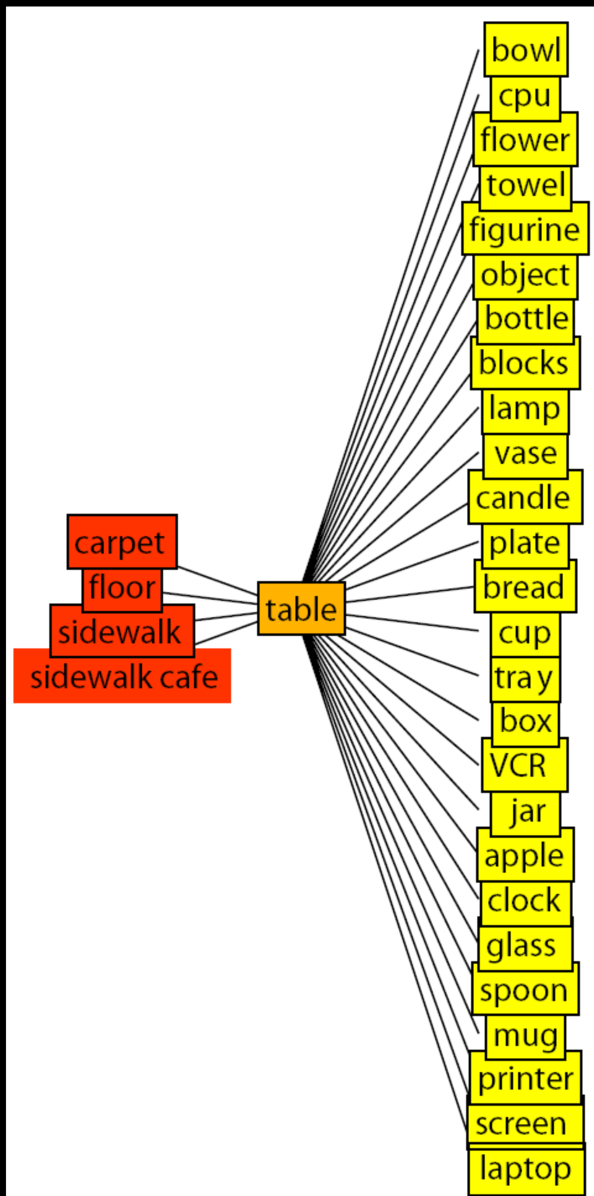
Recover support relations

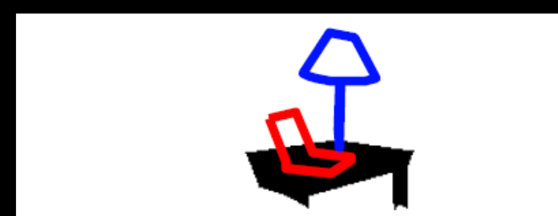


$$P_{support} = \frac{N_s}{N + \alpha}$$

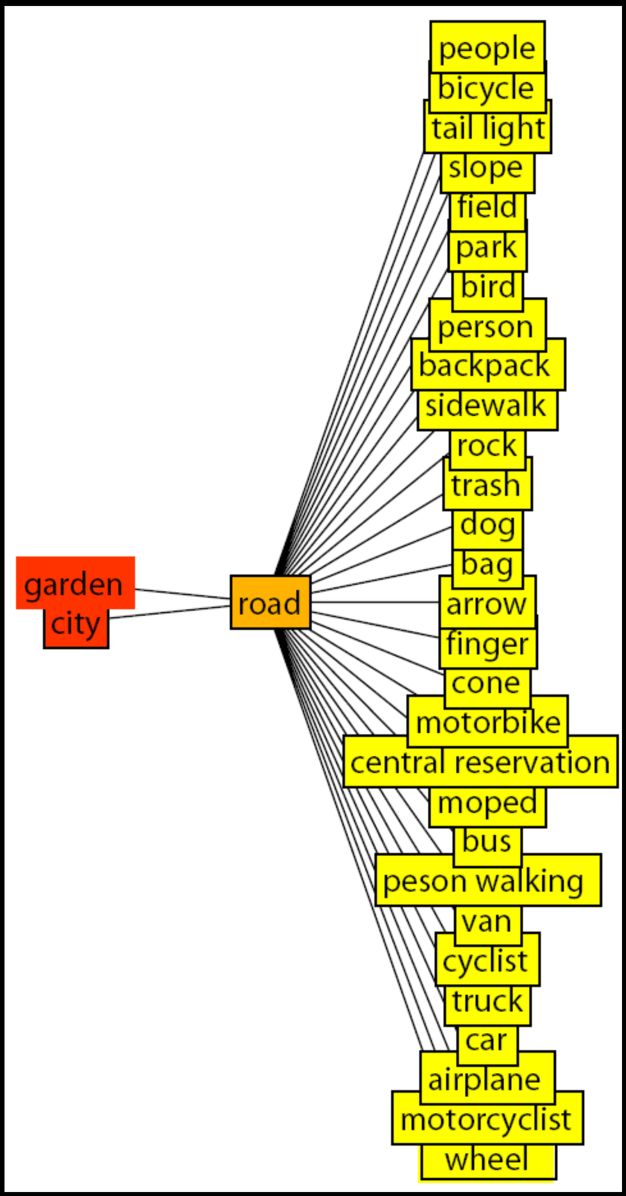
Over entire dataset, count number of images where bottom of object is inside support object

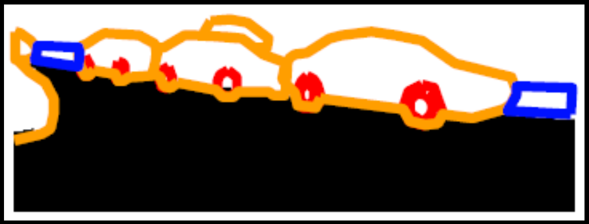
Learned/inferred support relations

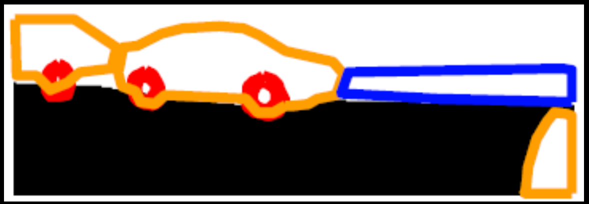




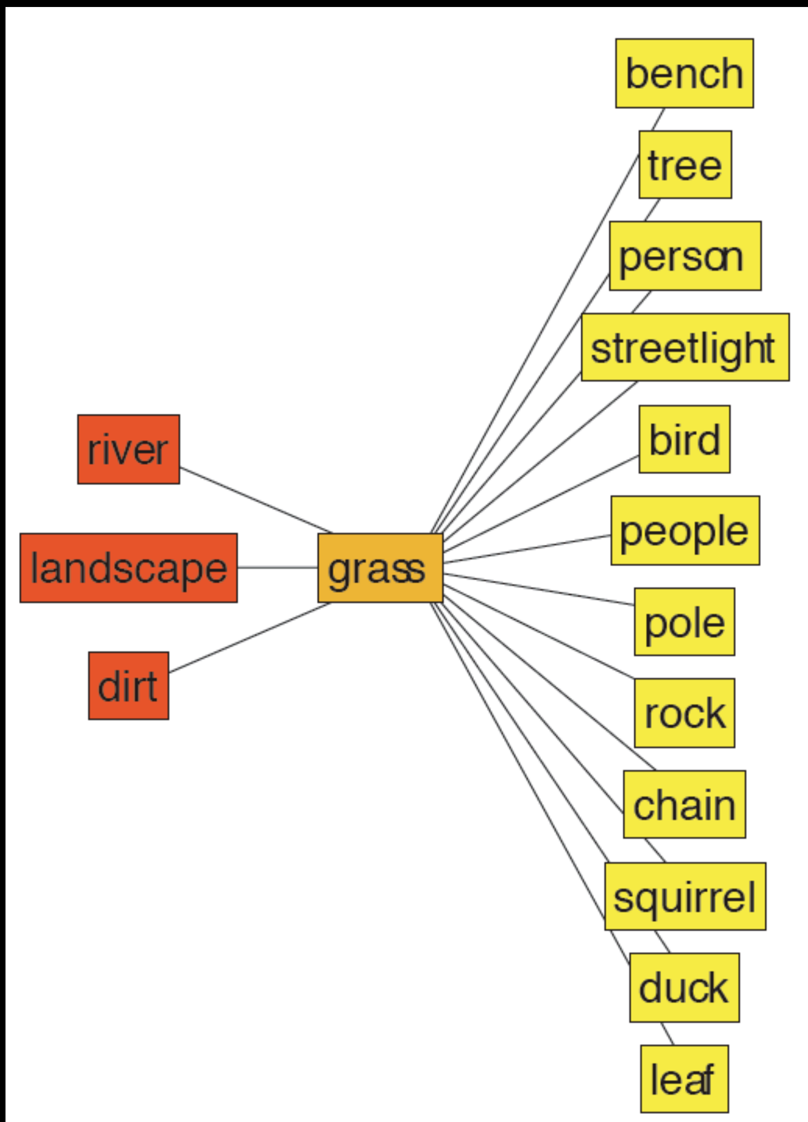
Learned/inferred support relations

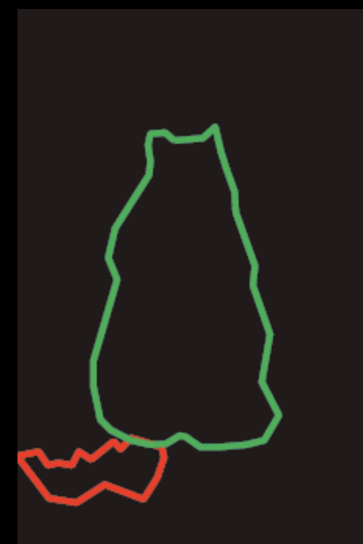






Learned/inferred support relations





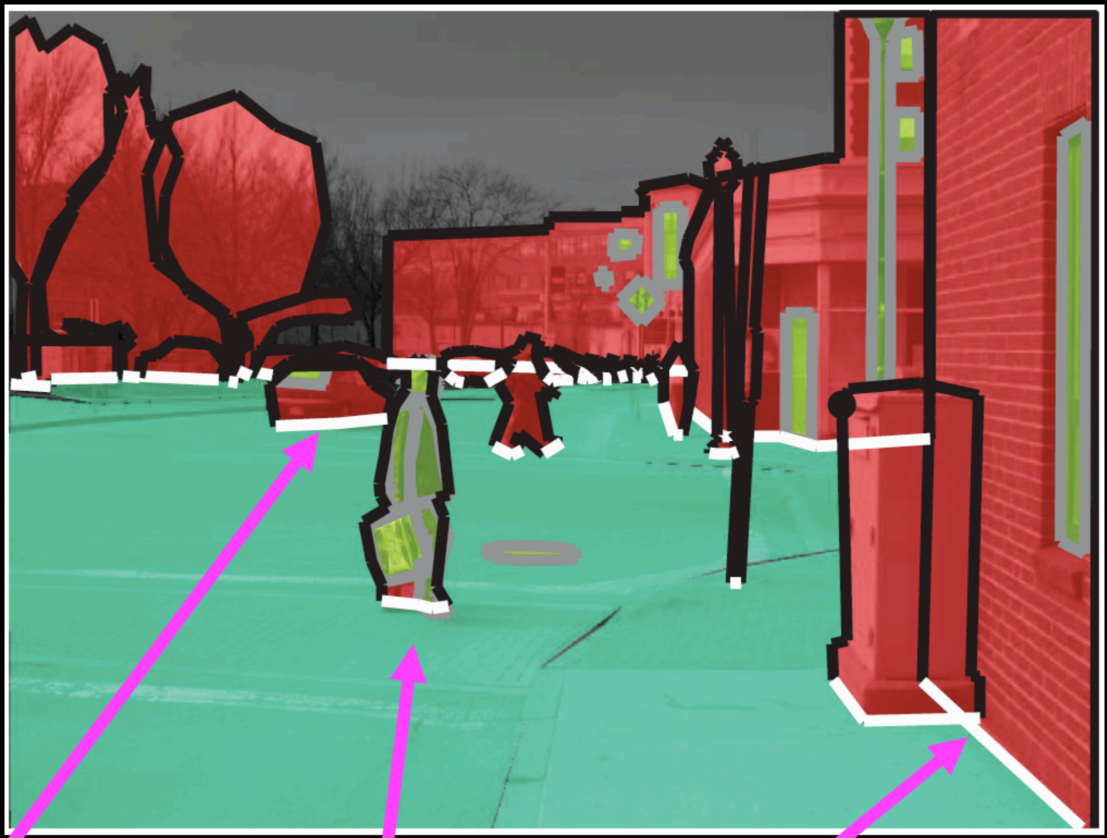
Recovering scene geometry

- Polygon types
  - Ground
  - Standing
  - Attached
- Edge types
  - Contact
  - Attached
  - Occluded
- Camera parameters

Edge types

Ground and attached objects have attached edges

Standing objects can have contact or occluding edges



Cues for contact edges:

Orientation

Proximity to ground

Length

Recovering scene geometry

- Polygon types
  - Ground
  - Standing
  - Attached
- Edge types
  - Contact
  - Attached
  - Occluded
- Camera parameters

Absolute (monocular) 3D cues

Are there any monocular cues that give us absolute 3D information from a single image?

Camera parameters

- Assume
  - flat ground plane
  - camera roll is negligible (consider pitch only)
- Camera parameters: height and orientation

Slide from J-F Lalonde

Camera parameters

v

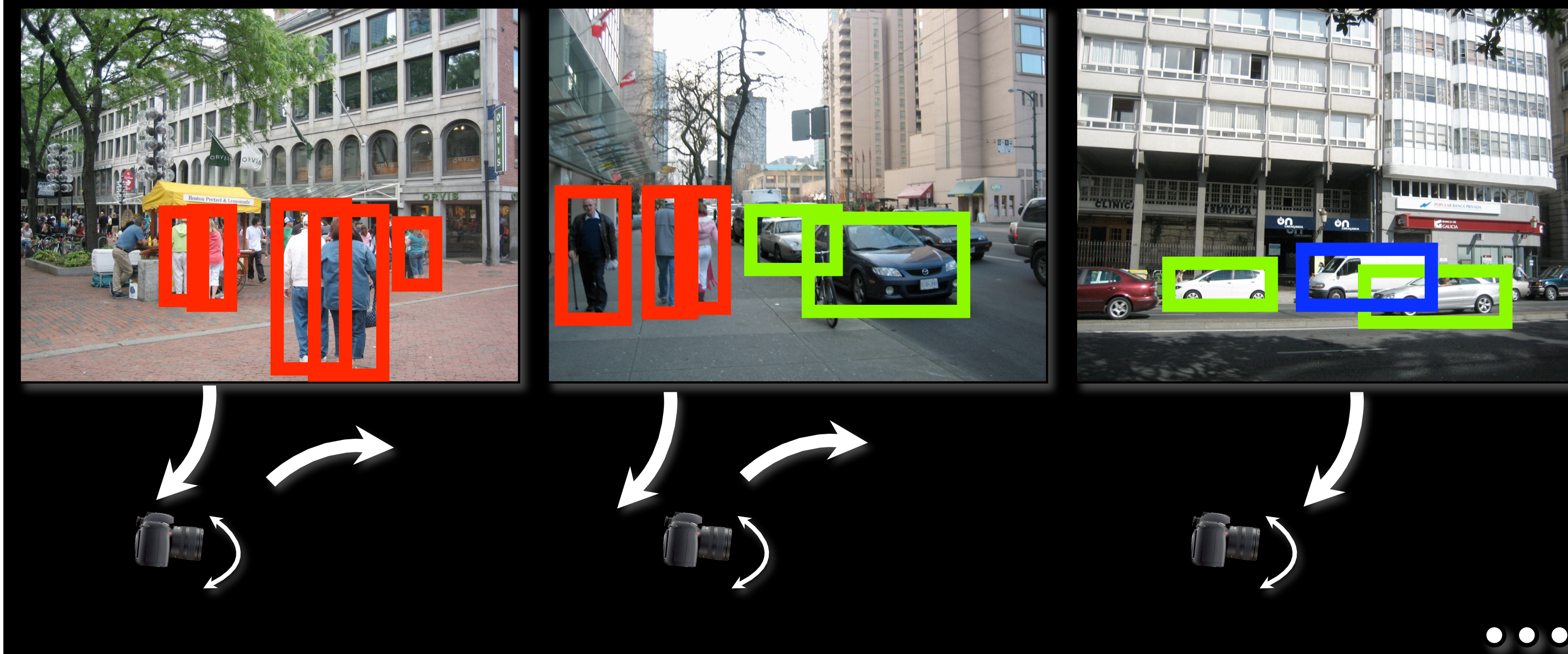
$$\frac{t-b}{X} = \frac{v-b}{C}$$

X – World object height (in meters)
C – World camera height (in meters)

Camera parameters

Human height distribution
1.7 +/- 0.085 m
(National Center for Health Statistics)

Car height distribution
1.5 +/- 0.19 m
(automatically learned)

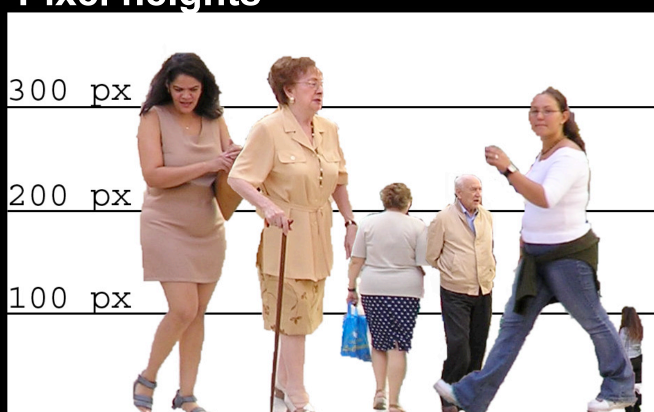


Slide from J-F Lalonde

Object heights

Database image

Pixel heights



Real heights

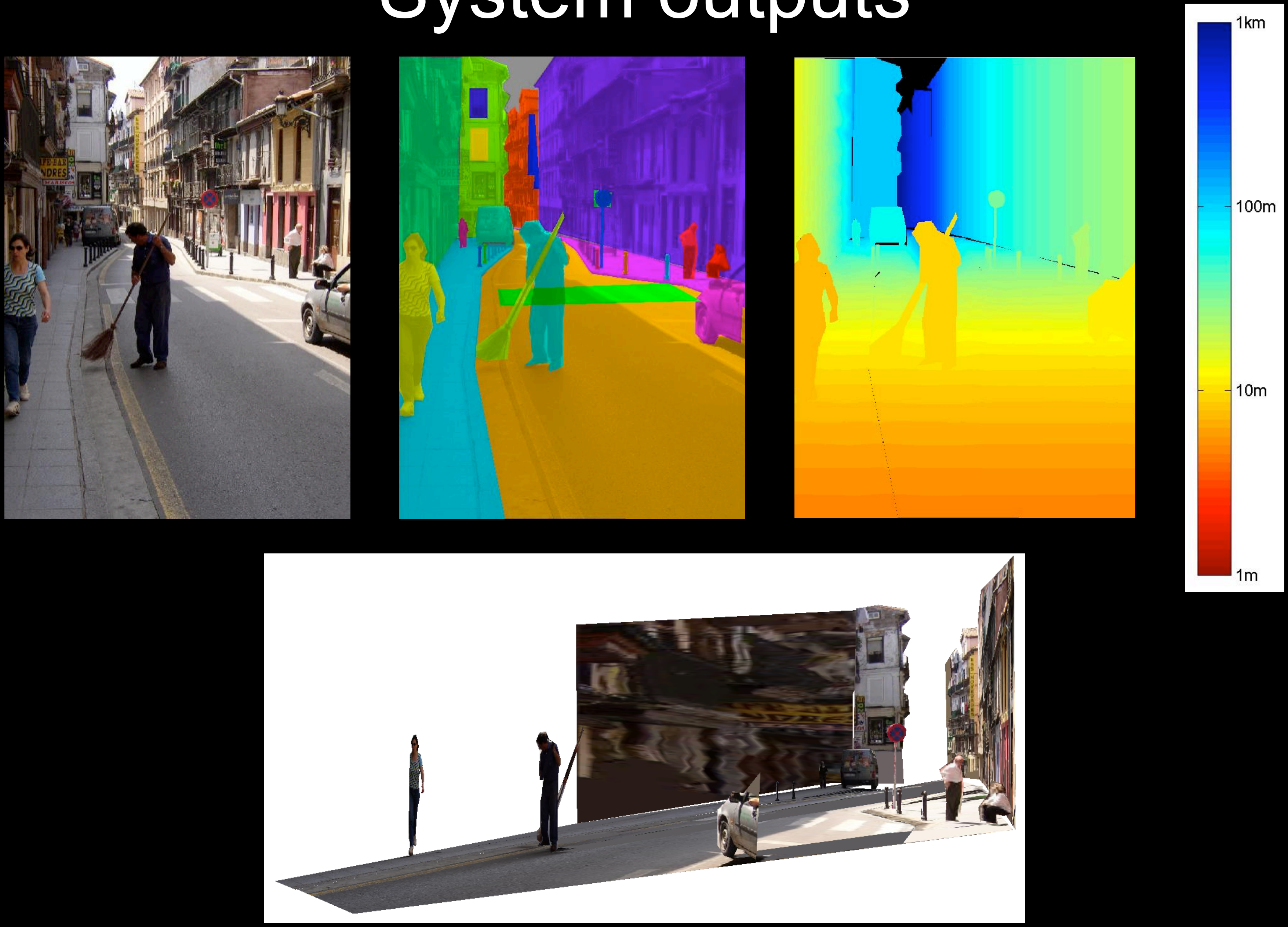
Slide from J-F Lalonde

Recovered object heights

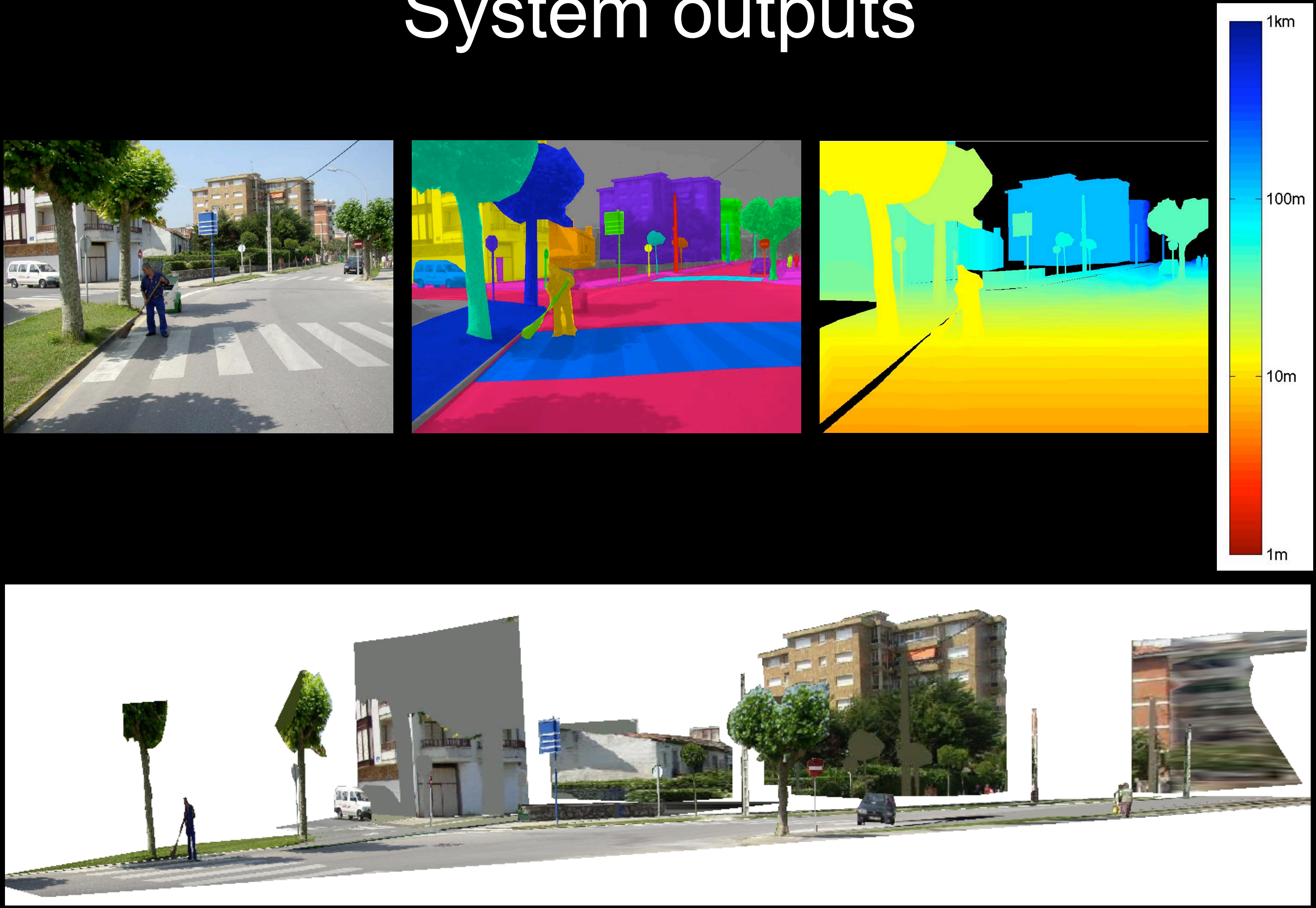
(Average, in meters)

Standing objects		Attached objects	
Person	1.65	Wheel	0.62
Car	1.46	Window	2.16
Bicycle	1.05	Arm	0.72
Trash	1.24	Windshield	0.47
Parking meter	1.58	Head	0.41
Fence	1.89	Tail light	0.34
Van	1.89	Headlight	0.26
Firehydrant	0.87	License plate	0.23
Cone	0.74	Mirror	0.22

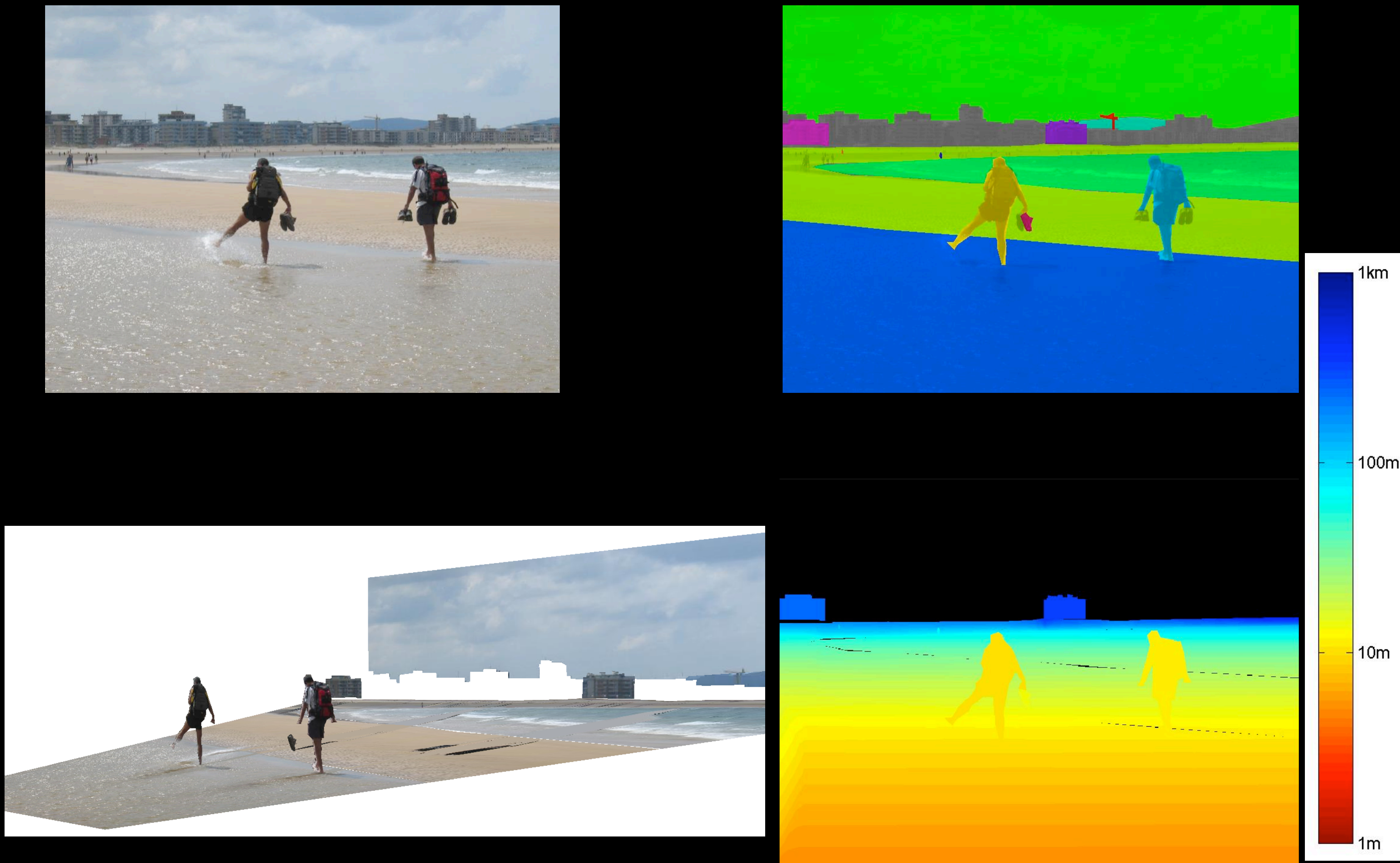
System outputs



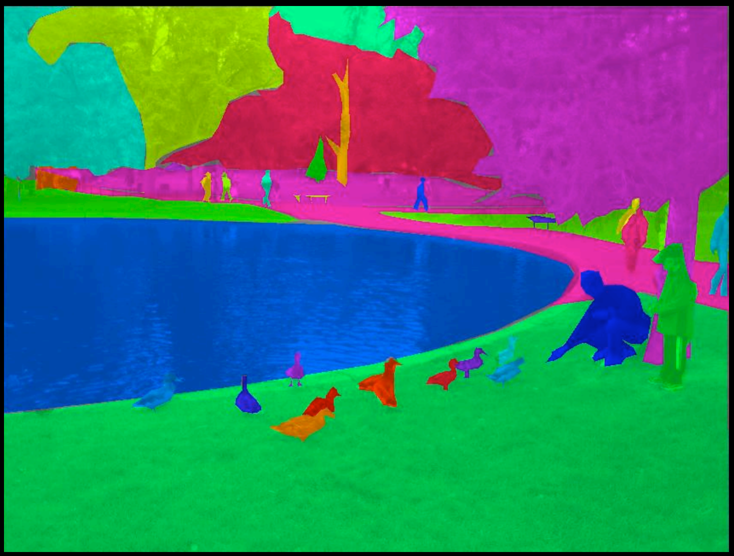
System outputs

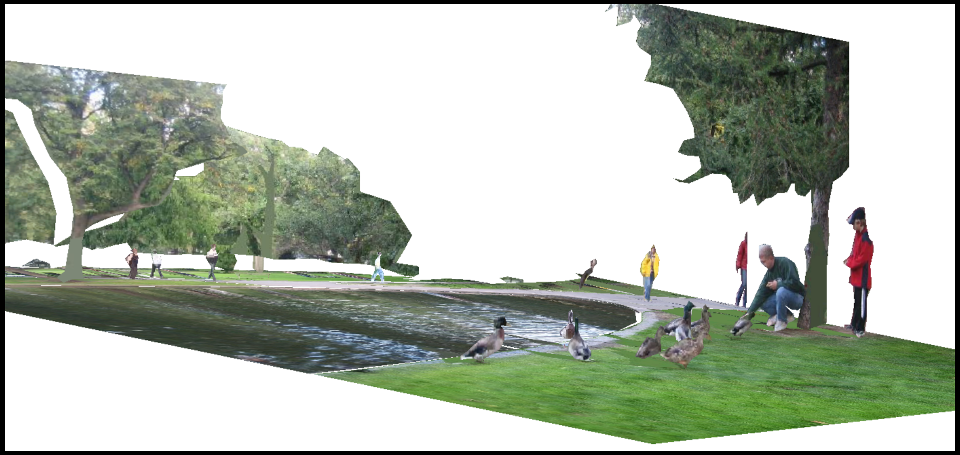


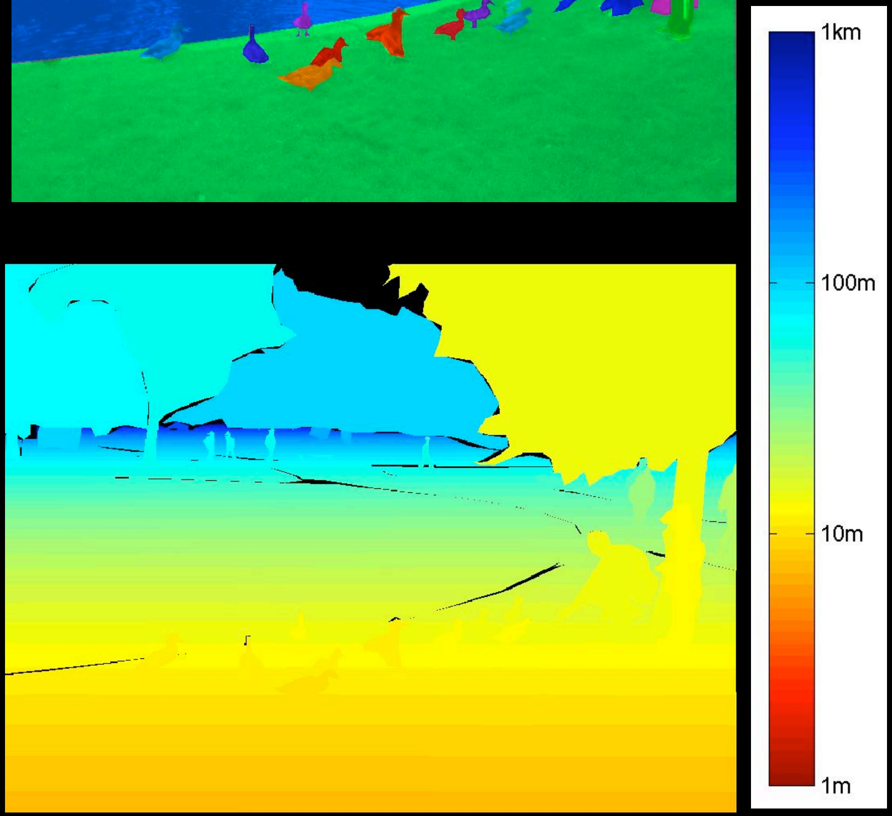
System outputs



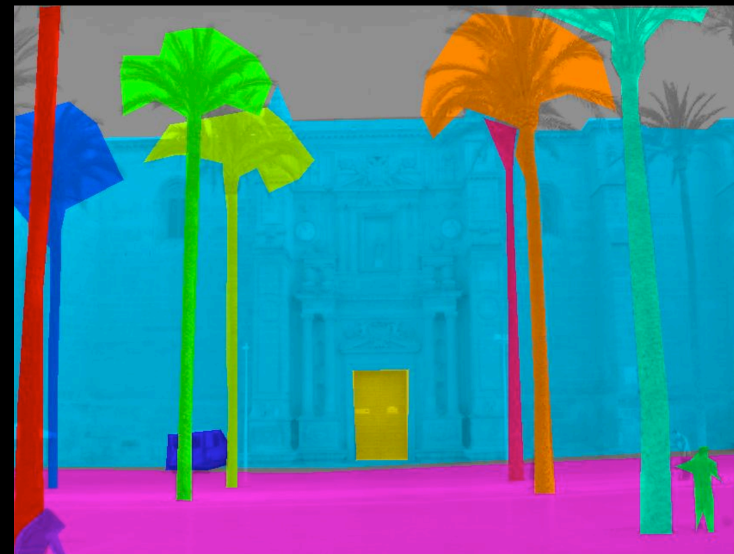
System outputs

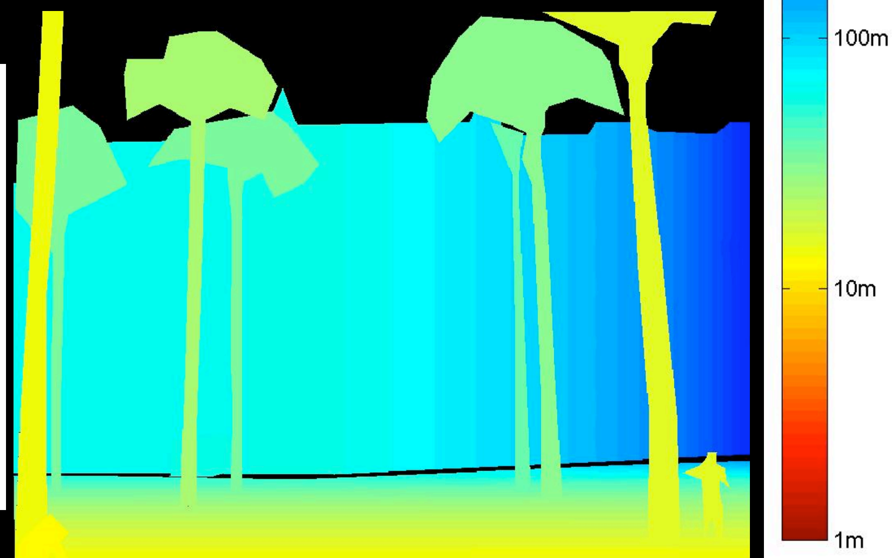




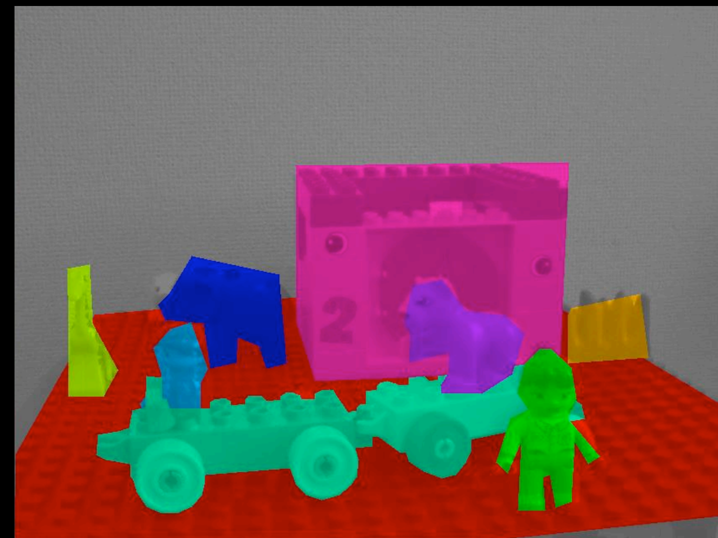


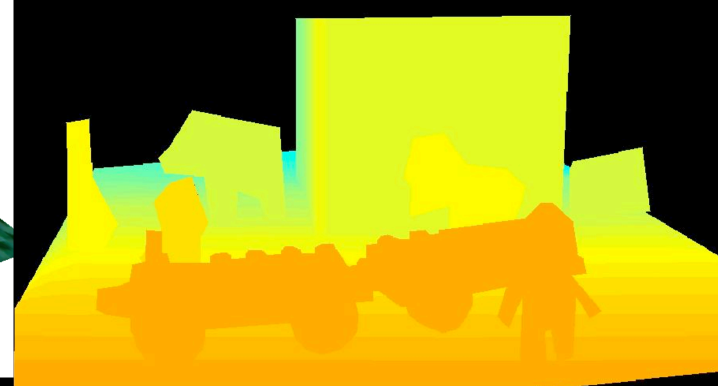
System outputs

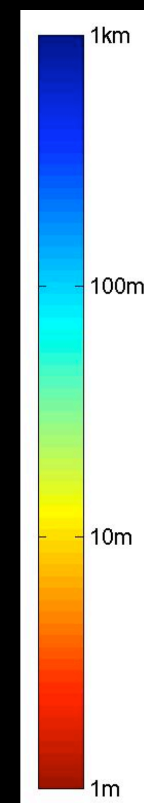




Toy example…





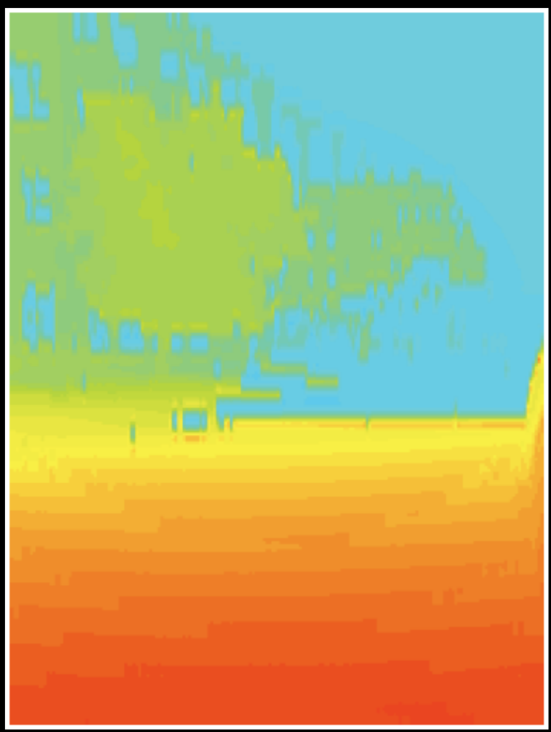


Submitted images

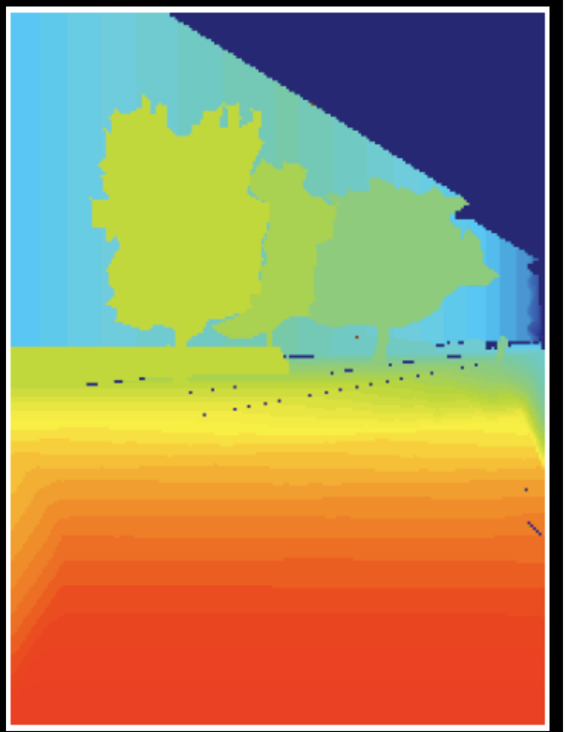
Accuracy of 3D outputs

Evaluation with range data [Saxena et al. 2007]
Relative error: 0.29
Computed over 5-70 meter range (46% of pixels)

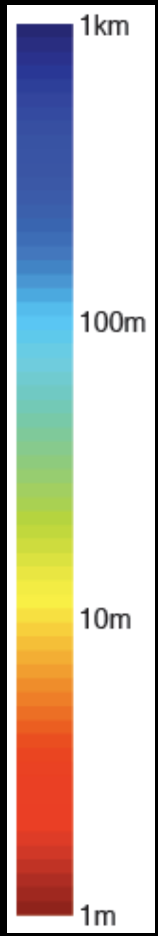
Input image



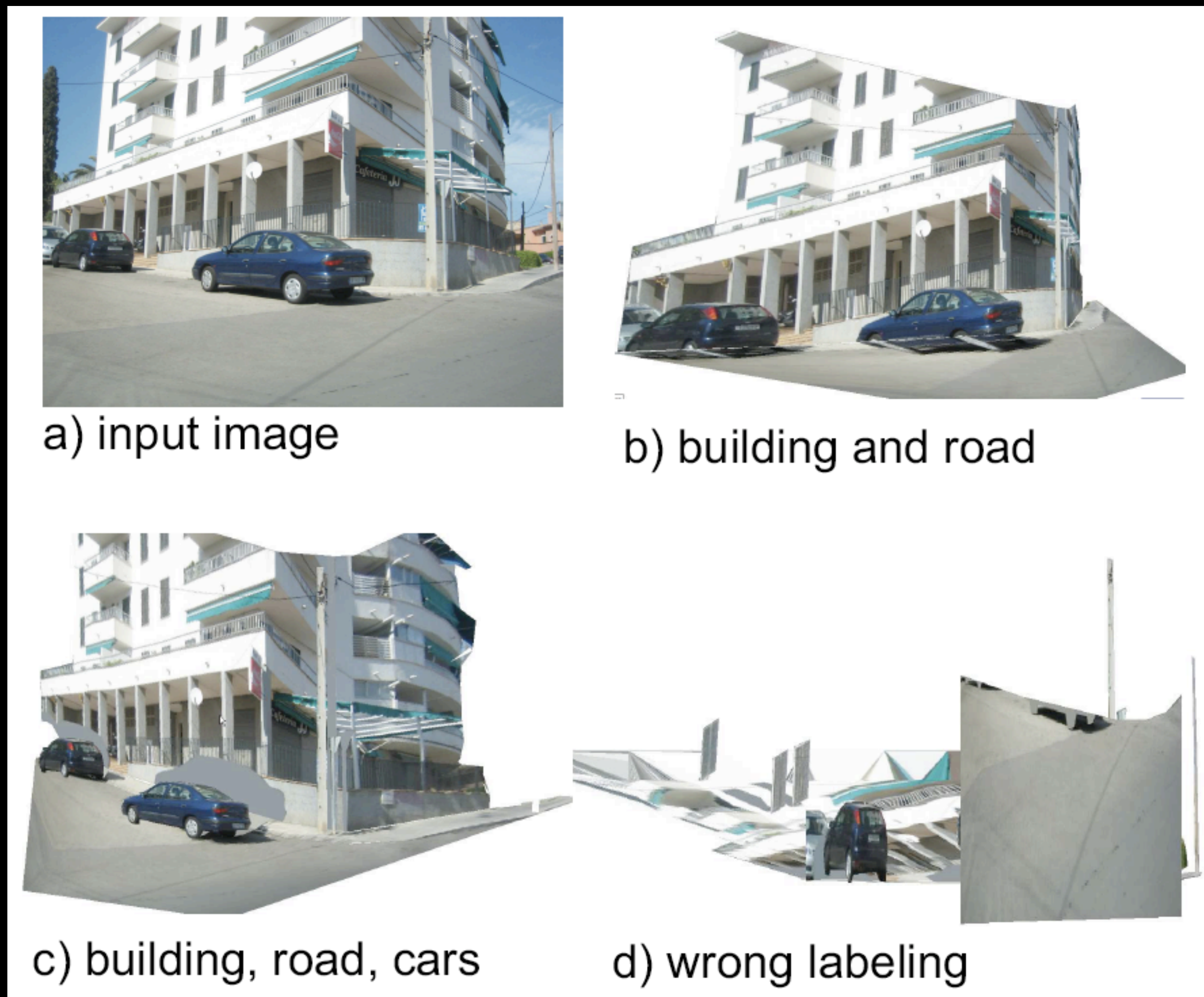
Range scan



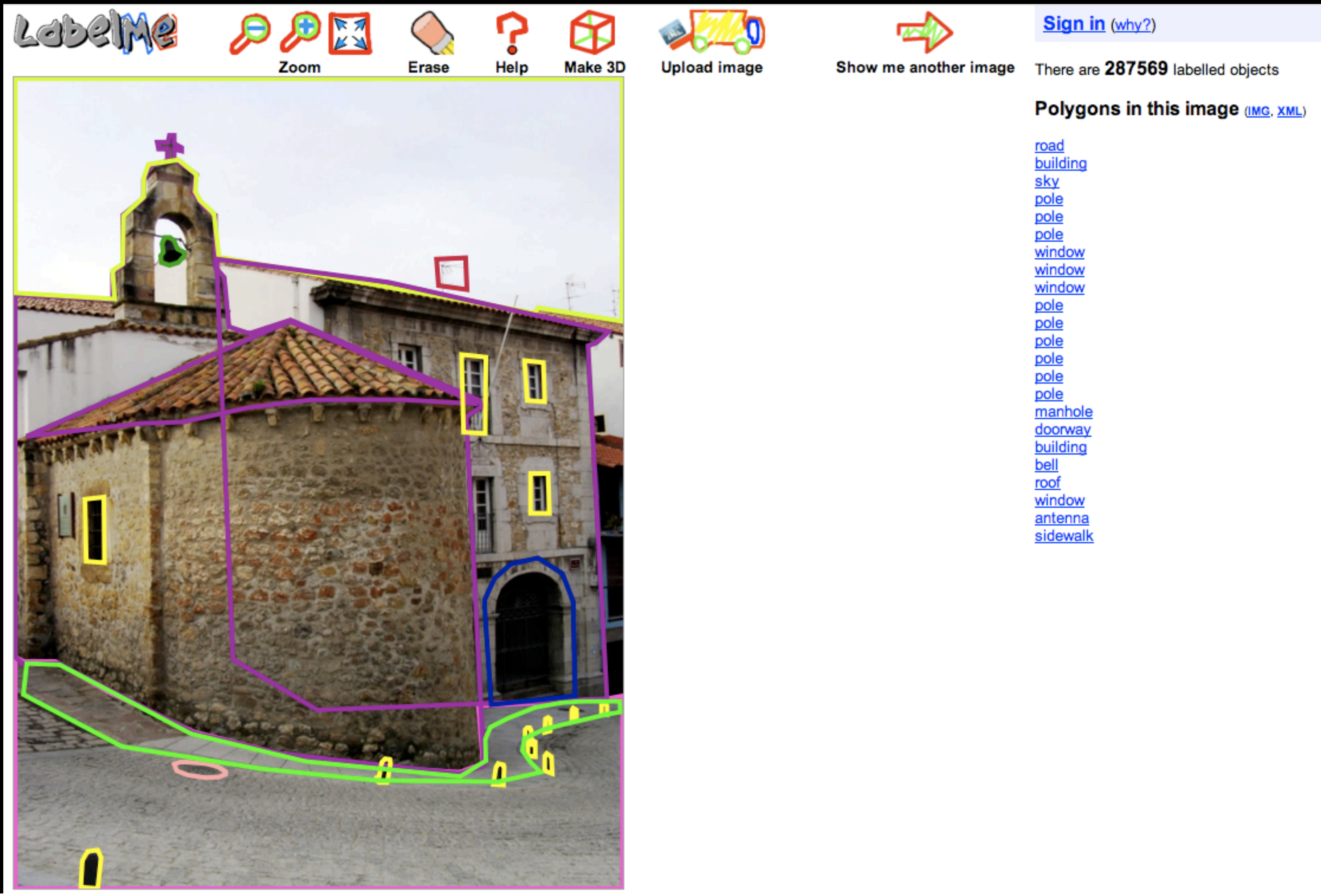
System output



How does labeling accuracy affect outputs?



Labeling 3D

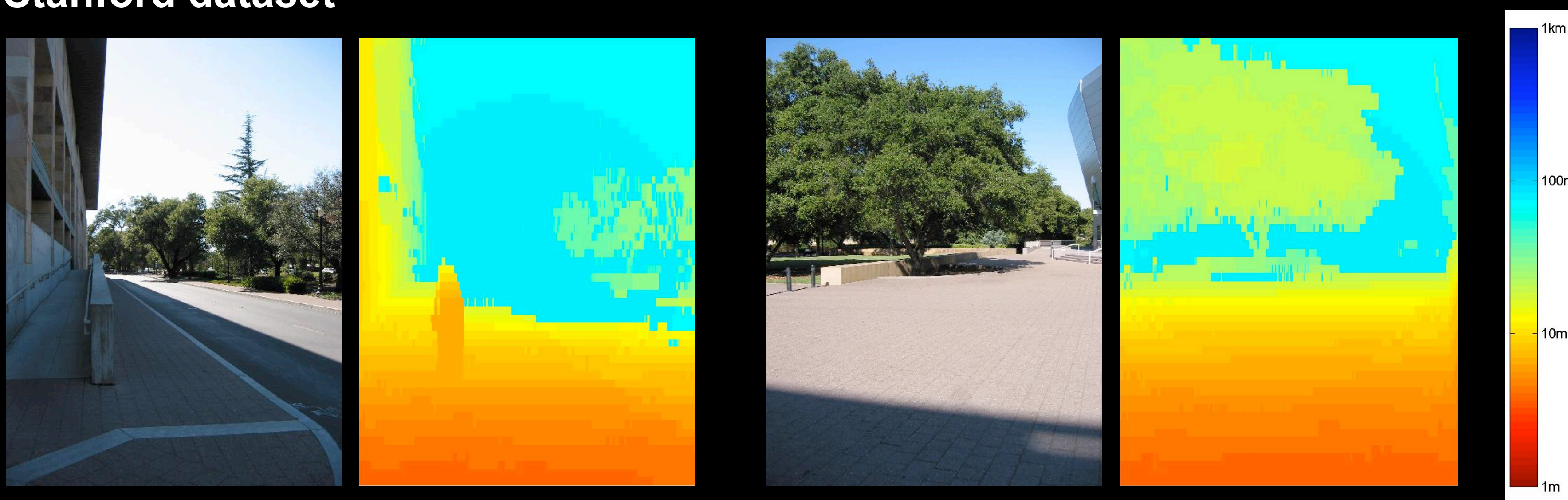


LabelMe
Zoom
Erase
Help
Make 3D
Upload image
Show me another image
Sign in (why?)
There are 287569 labelled objects
Polygons in this image (IMG, XML)
road
building
sky
pole
pole
pole
window
window
window
pole
pole
pole
pole
pole
pole
manhole
doorway
building
bell
roof
window
antenna
sidewalk

Cut and glue!

Range scanners, stereo cameras

Stanford dataset

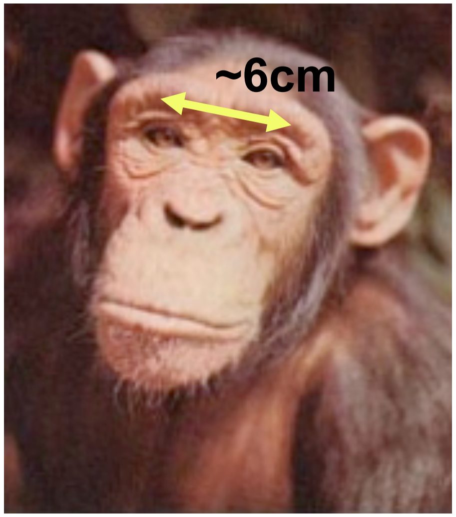


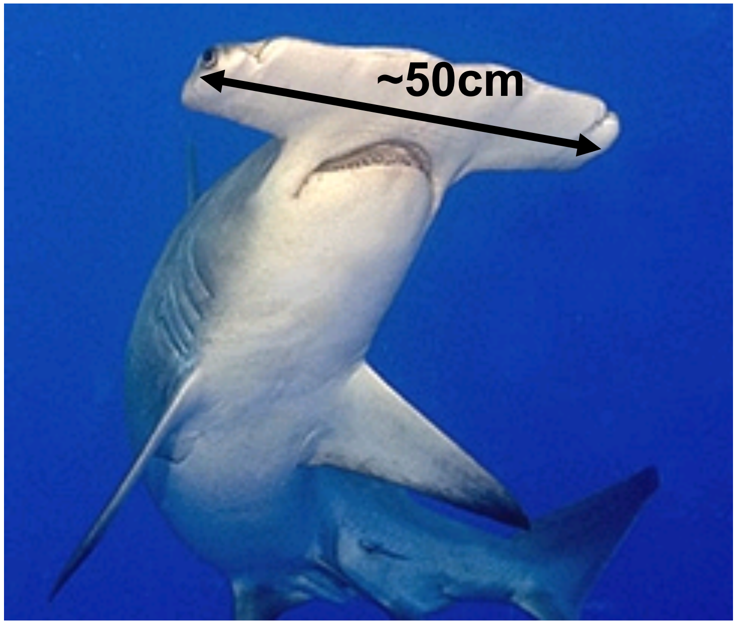
Depth map **Depth map**

Stereo

- Two eyes
- Depth without recognition: random dot stereogram, Julesz. The world is structured but with two eyes we can see even in random worlds.
- Hollow face illusion
- Illusion street inversed
- Simple stereo

Stereo vision





Depth for familiar objects

(Gregory 1970; Hill and Bruce 1993, 1994; Papathomas and DeCarlo 1999)

Depth without objects

Random dot stereograms (Bela Julesz)

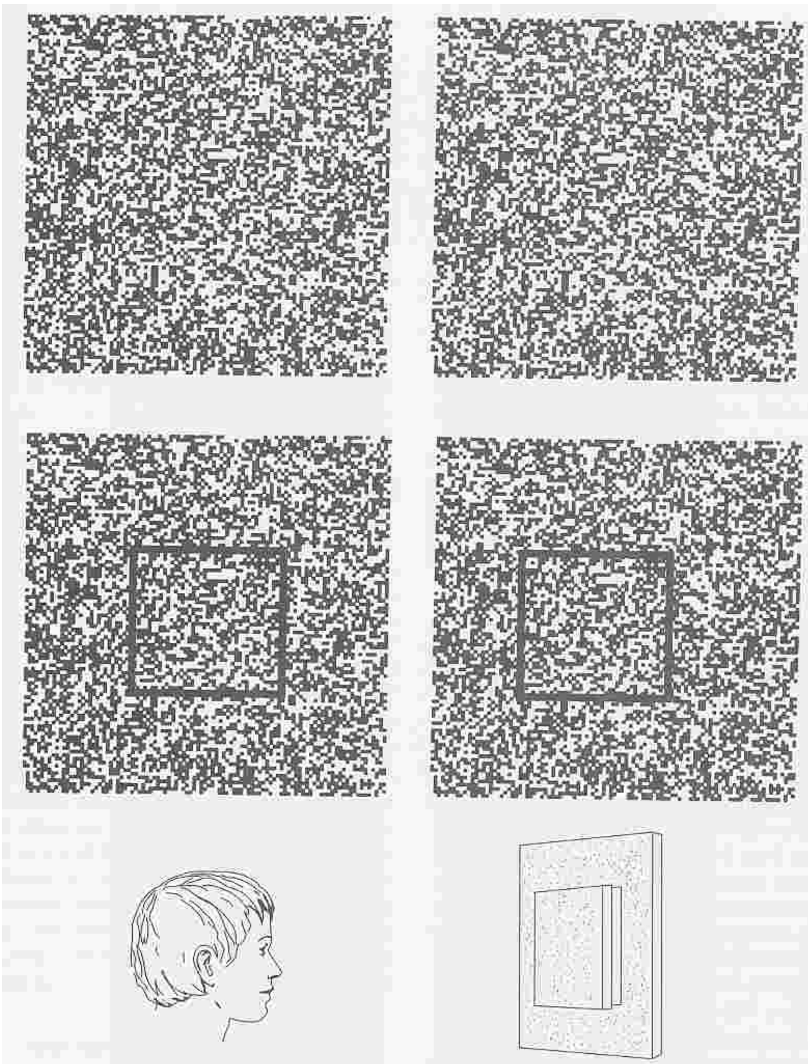


FIGURE 8.13

1	0	1	0	1	0	0	1	0	1
1	0	0	1	0	1	0	1	0	0
0	0	1	1	0	1	1	0	1	0
0	1	0	Y	A	A	B	B	0	1
1	1	1	X	B	A	B	A	0	1
0	0	1	X	A	A	B	A	1	0
1	1	1	Y	B	B	A	B	0	1
1	0	0	1	1	0	1	1	0	1
1	1	0	0	1	1	0	1	1	1
0	1	0	0	0	1	1	1	1	0

1	0	1	0	1	0	0	1	0	1
1	0	0	1	0	1	0	1	0	0
0	0	1	1	0	1	1	0	1	0
0	1	0	A	A	B	B	X	0	1
1	1	1	B	A	B	A	Y	0	1
0	0	1	A	A	B	A	Y	1	0
1	1	1	B	B	A	B	X	0	1
1	0	0	1	1	0	1	1	0	1
1	1	0	0	1	1	0	1	1	1
0	1	0	0	0	1	1	1	1	0

Julesz, 1971

Stereo photography and stereo viewers

Take two pictures of the same subject from two slightly different viewpoints and display so that each eye sees only one of the images.

Invented by Sir Charles Wheatstone, 1838

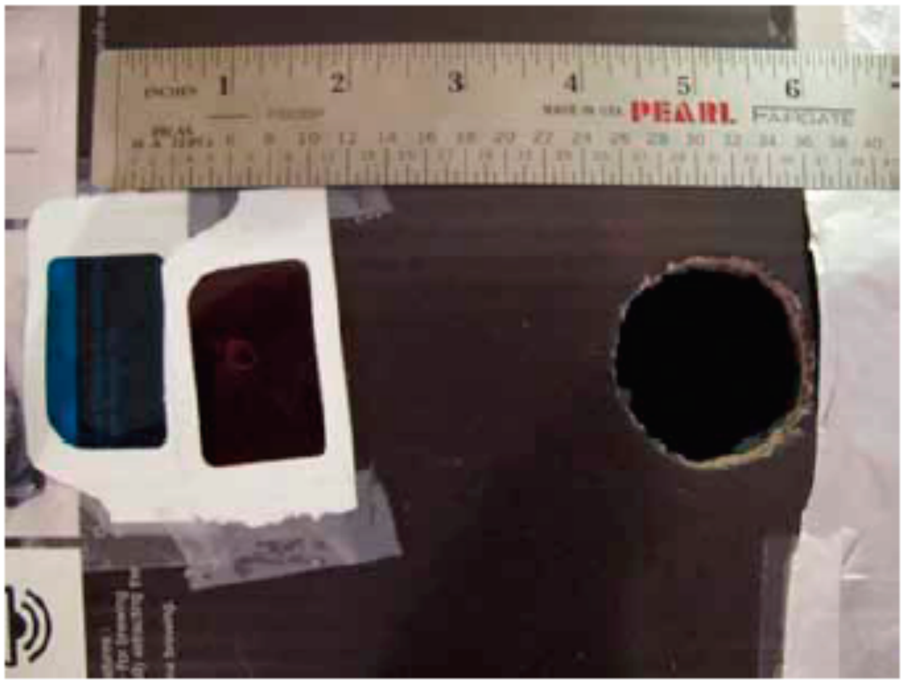
Image courtesy of fisher-price.com

Slide credit: Kristen Grauman

Public Library, Stereoscopic Looking Room, Chicago, by Phillips, 1923

de credit: Kristen Grauman

Anaglyph pinhole camera



Autostereograms

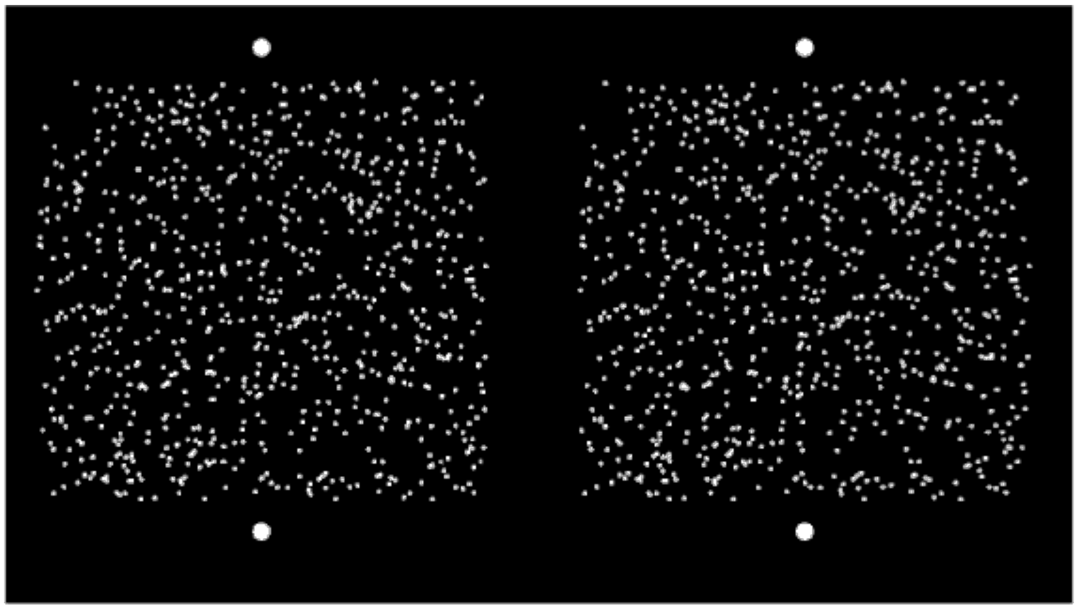
Exploit disparity as depth cue using single image.

(Single image random dot stereogram, Single image stereogram)

Images from magiceye.com

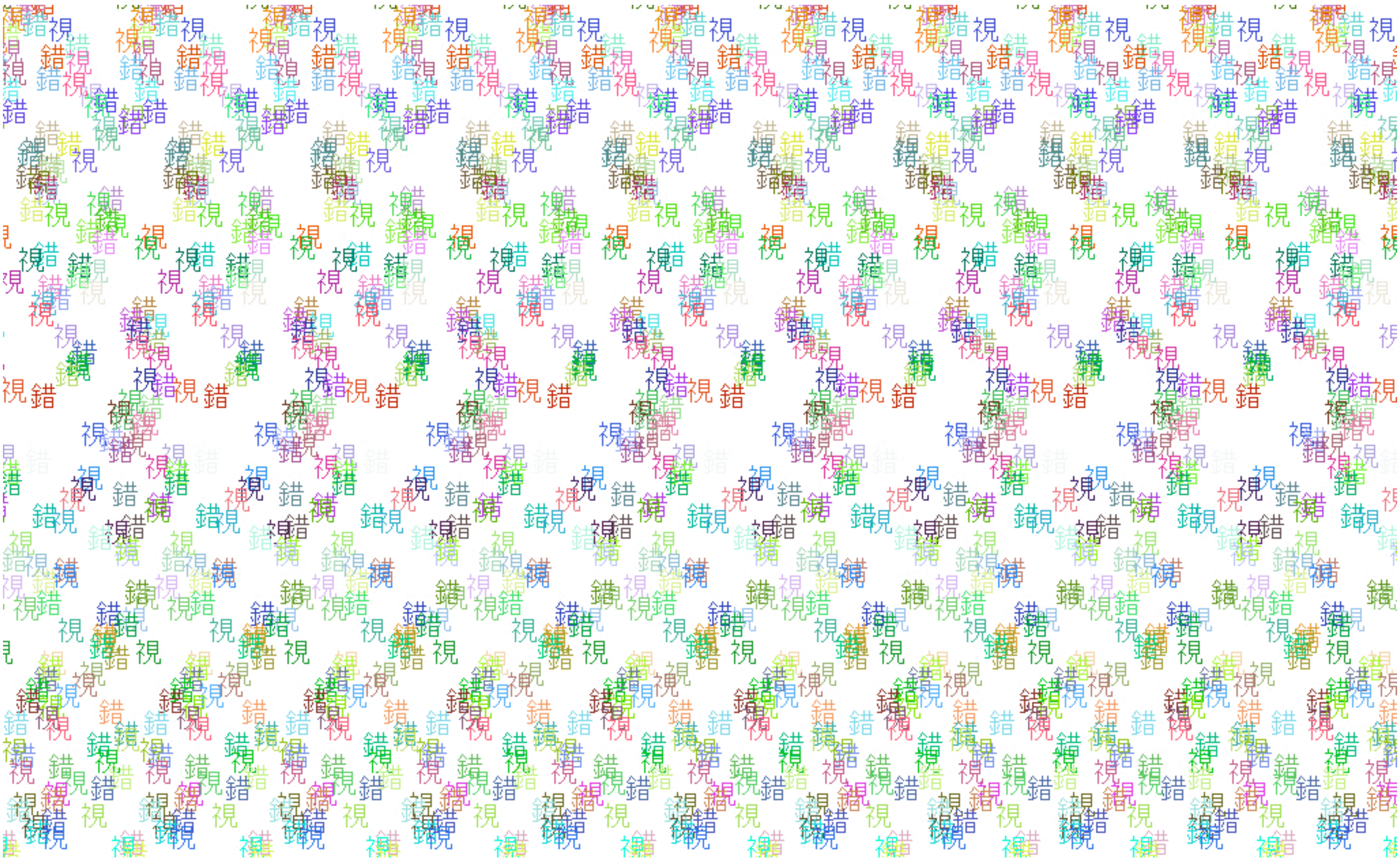
Slide credit: Kristen Grauman

Cross-fusion

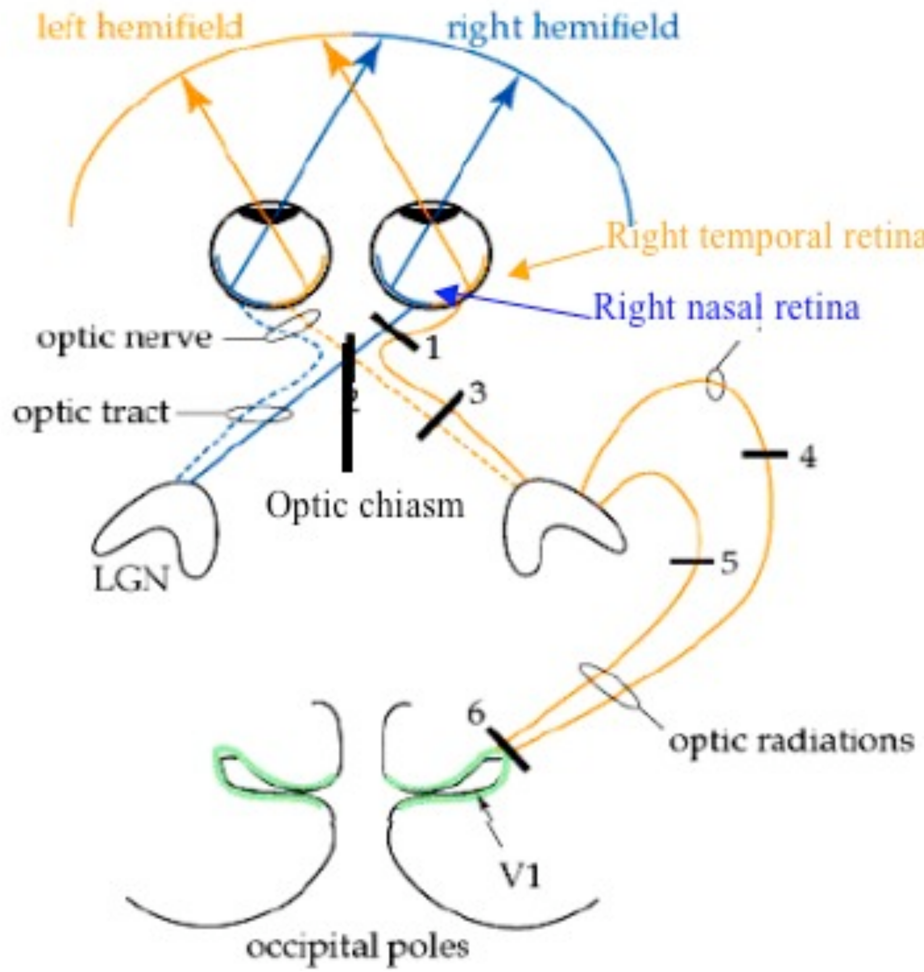


http://www.journalofvision.org/8/8/5/article.aspx

A typical disparity-defined stimulus from the experiment, showing a horizontally oriented half-cylinder. This figure is designed for cross-fusion, but in the experiment the stimuli were viewed through LCD-shuttered glasses and the large dots were not present.



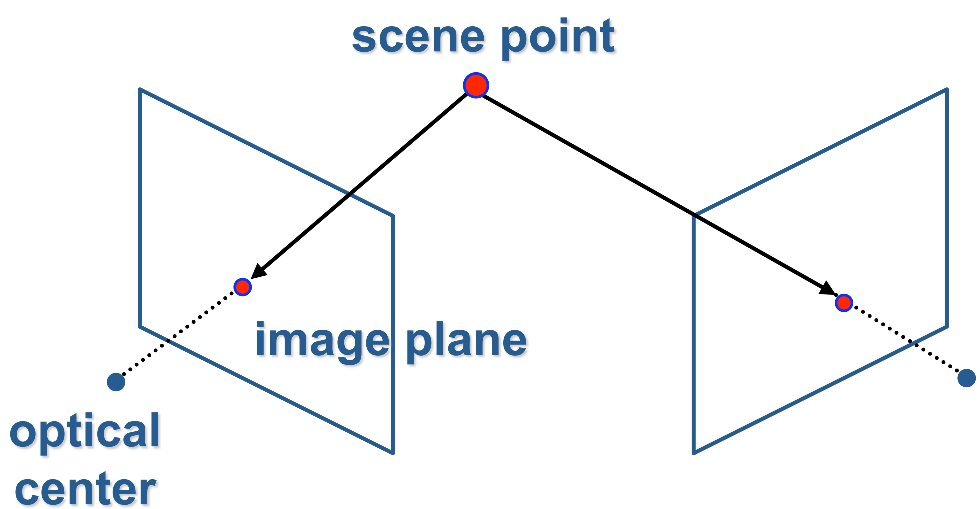
http://www.psy.ritsumei.ac.jp/~akitaoka/stereo3e.html

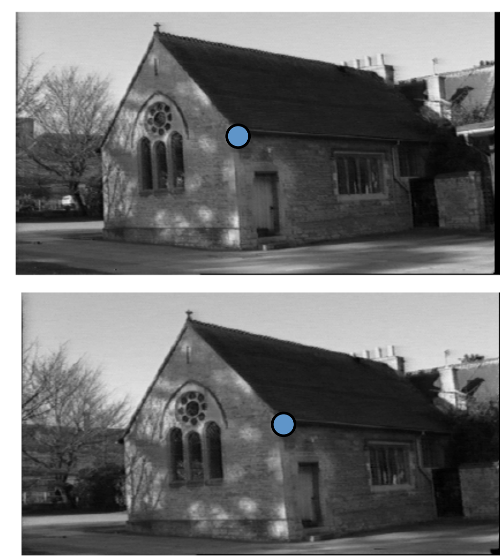


http://www.ssc.education.ed.ac.uk/courses/vi&multi/cgilmour.html

Estimating depth with stereo

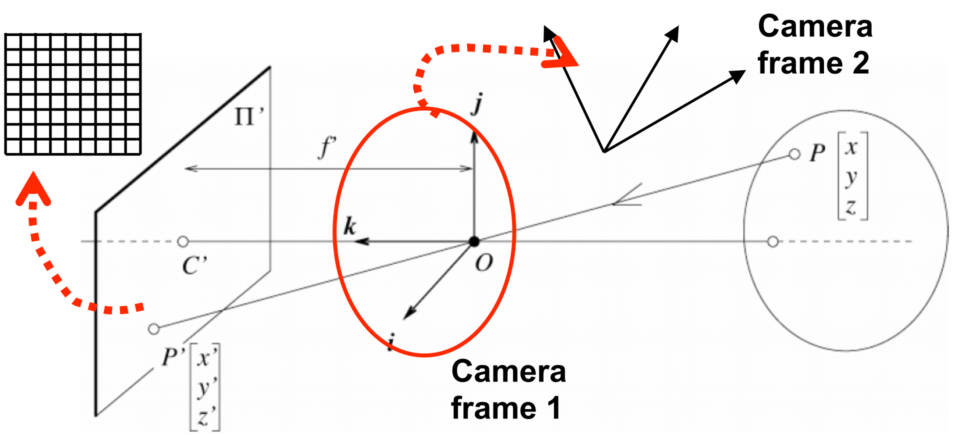
- **Stereo**: shape from "motion" between two views
- We'll need to consider:
  - Info on camera pose ("calibration")
  - Image point correspondences





Slide credit: Kristen Grauman

Camera parameters



Extrinsic parameters:
Camera frame 1 ←→ Camera frame 2

Intrinsic parameters:
Image coordinates relative to camera ←→ Pixel coordinates

- ***Extrinsic* params: rotation matrix and translation vector**
- ***Intrinsic* params: focal length, pixel sizes (mm), image center point, radial distortion parameters**

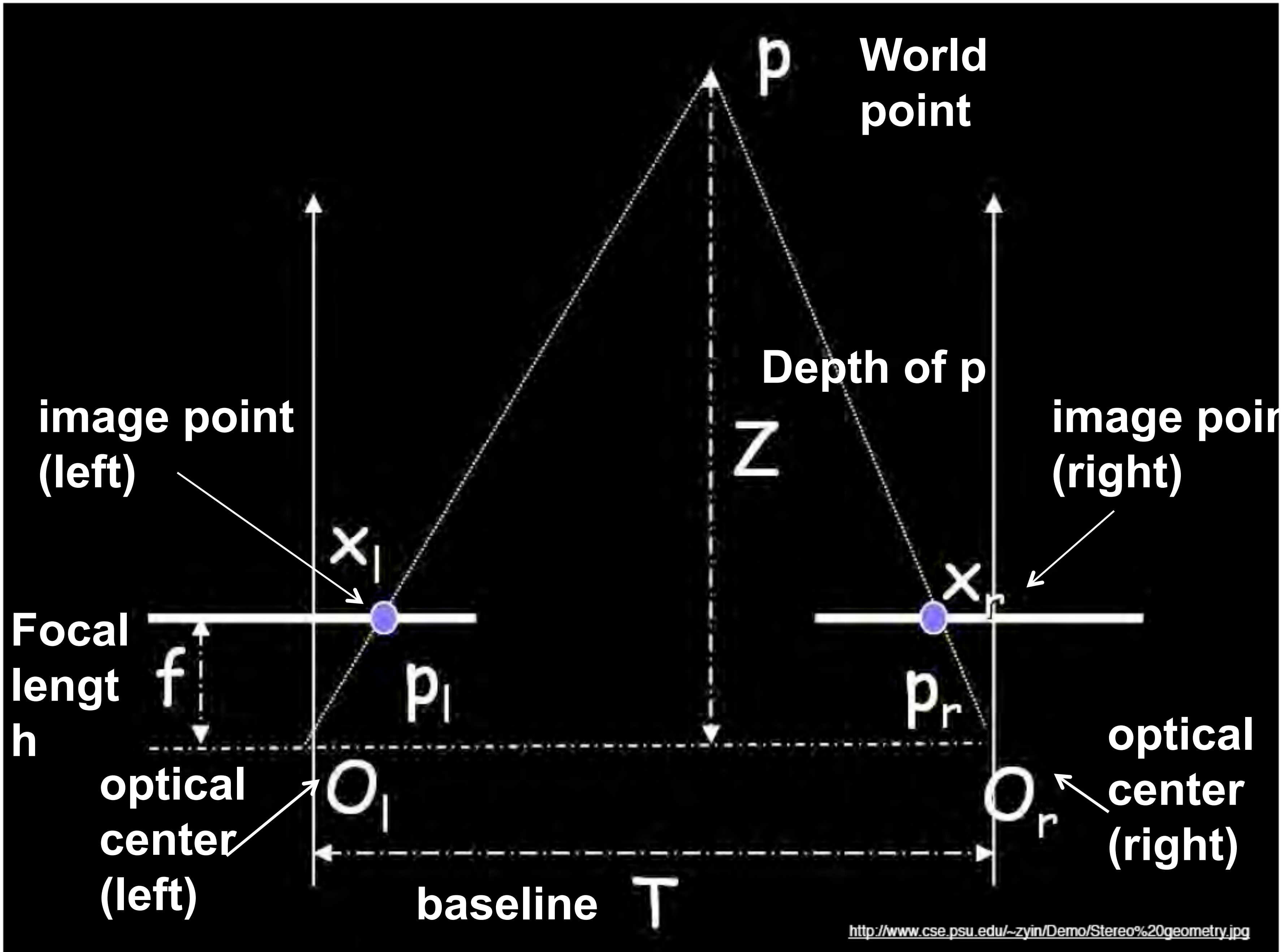
We'll assume for now that these parameters are given and fixed.

Slide credit: Kristen Grauman

Geometry for a simple stereo system

- First, assuming parallel optical axes, known camera parameters (i.e., calibrated cameras):

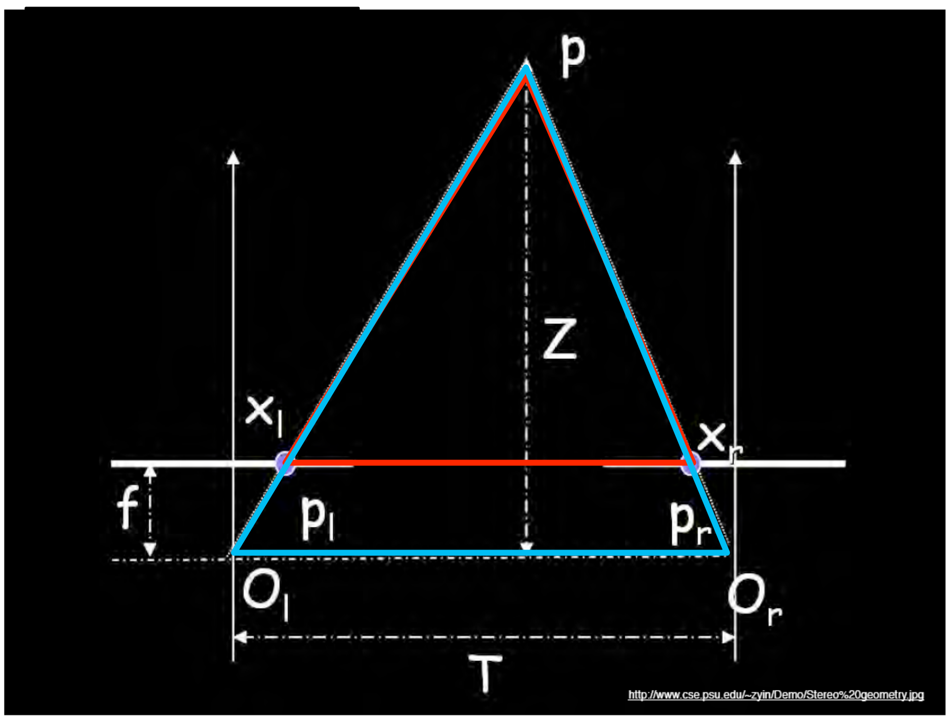
Slide credit: Kristen Grauman



Slide credit: Kristen Grauman

Geometry for a simple stereo system

- Assume parallel optical axes, known camera parameters (i.e., calibrated cameras). We can triangulate via:



Similar triangles (p_l, P, p_r) and (O_l, P, O_r):

$$\frac{T + x_l - x_r}{Z - f} = \frac{T}{Z}$$

$$Z = f \frac{T}{x_r - x_l}$$

disparity → $x_r - x_l$

Slide credit: Kristen Grauman

Depth from disparity

image I(x,y)

Disparity map D(x,y)

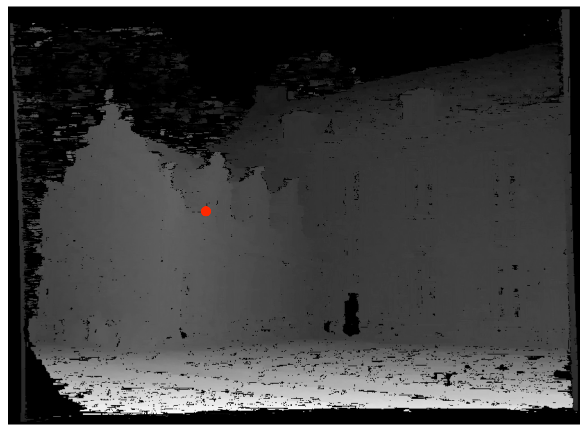
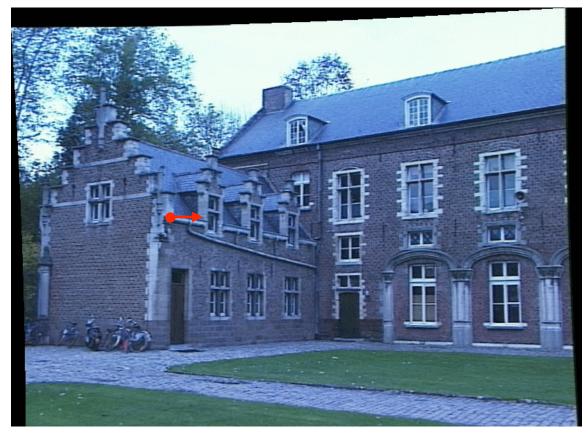


image I´(x´,y´)

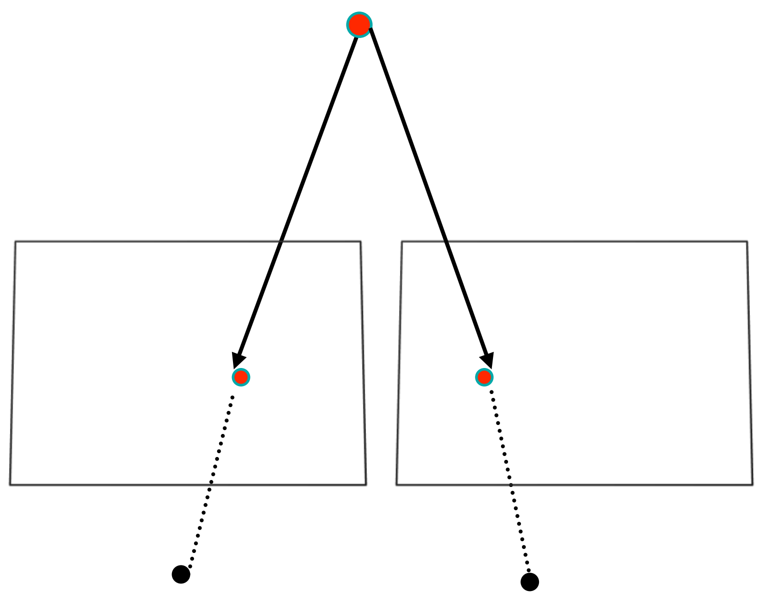


$$(x',y')=(x+D(x,y), y)$$

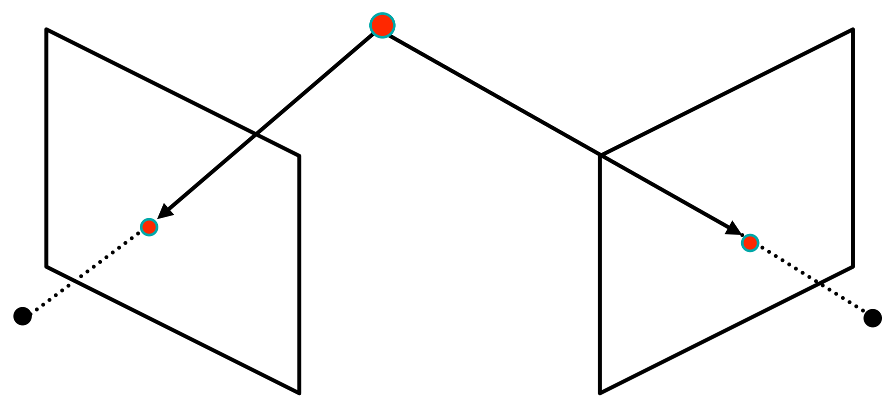
Slide credit: Kristen Grauman

General case, with calibrated cameras

- The two cameras need not have parallel optical axes.

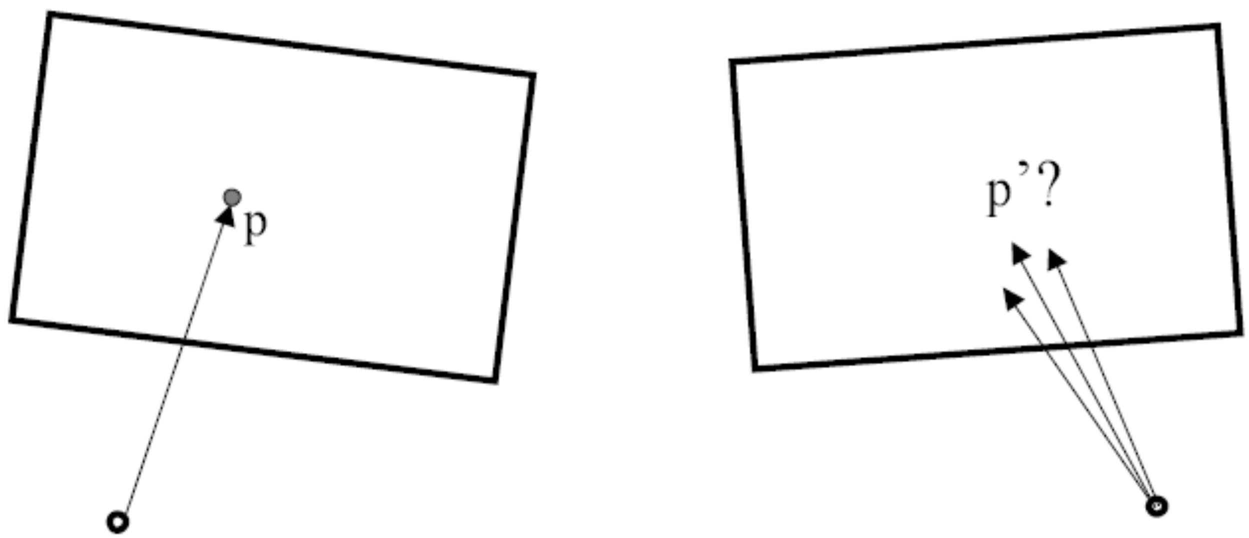


Vs.



Slide credit: Kristen Grauman

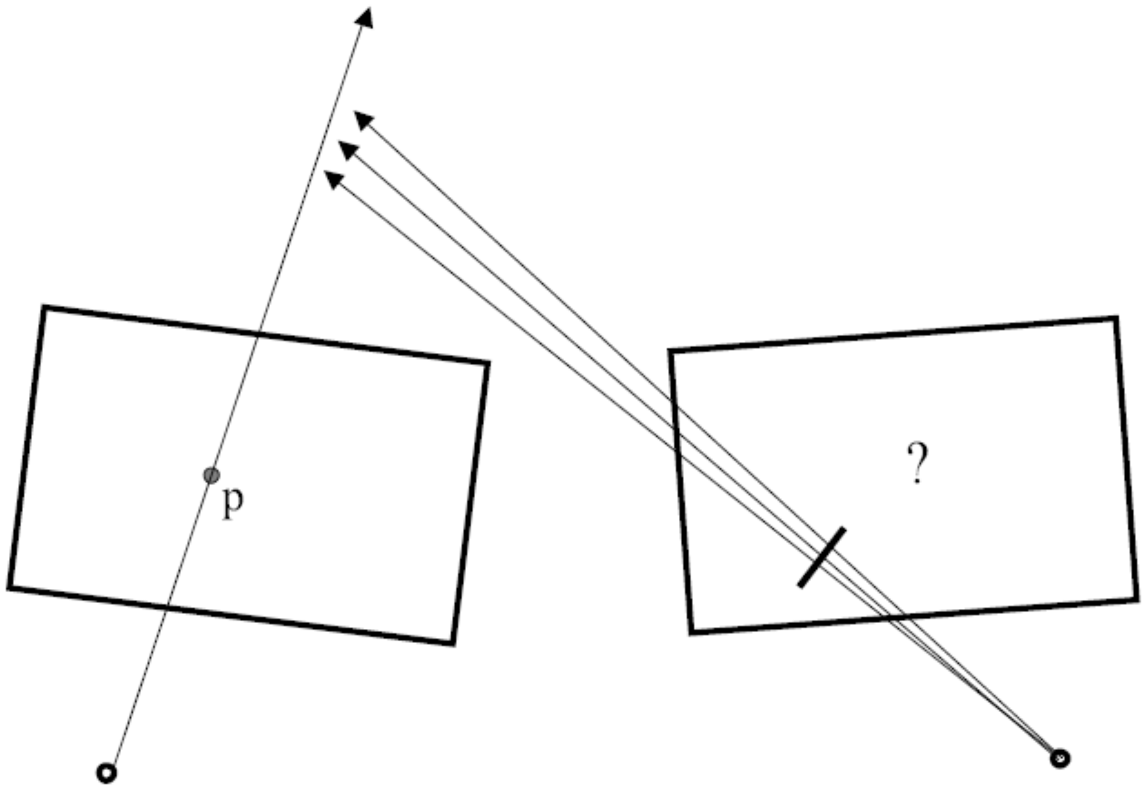
Stereo correspondence constraints



- **Given p in left image, where can corresponding point p' be?**

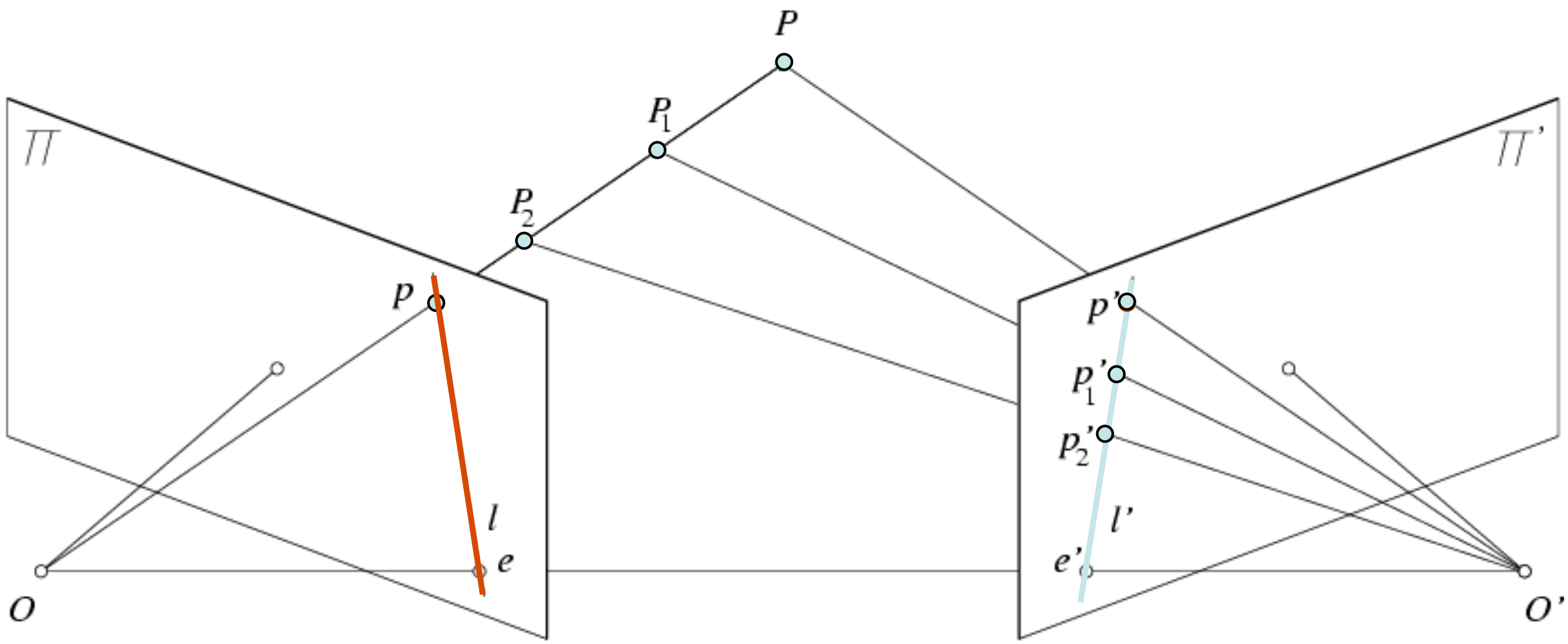
Slide credit: Kristen Grauman

Stereo correspondence constraints



Slide credit: Kristen Grauman

Epipolar constraint



Geometry of two views constrains where the corresponding pixel for some image point in the first view must occur in the second view:

- **It must be on the line carved out by a plane connecting the world point and optical centers.**

Why is this useful?

Slide credit: Kristen Grauman

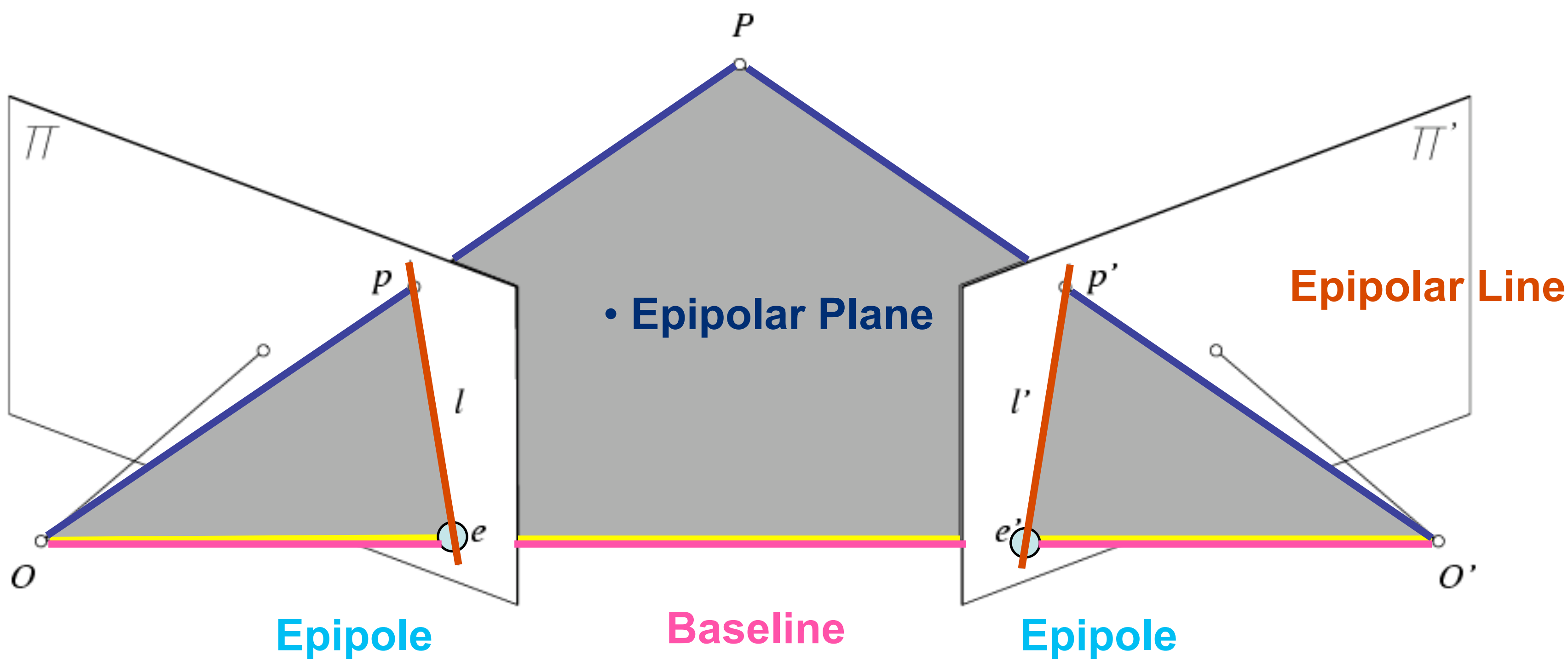
Epipolar constraint

This is useful because it reduces the correspondence problem to a 1D search along an epipolar line.

Image from Andrew Zisserman

Slide credit: Kristen Grauman

Epipolar geometry



http://www.ai.sri.com/~luong/research/Meta3DViewer/EpipolarGeo.html

Slide credit: Kristen Grauman

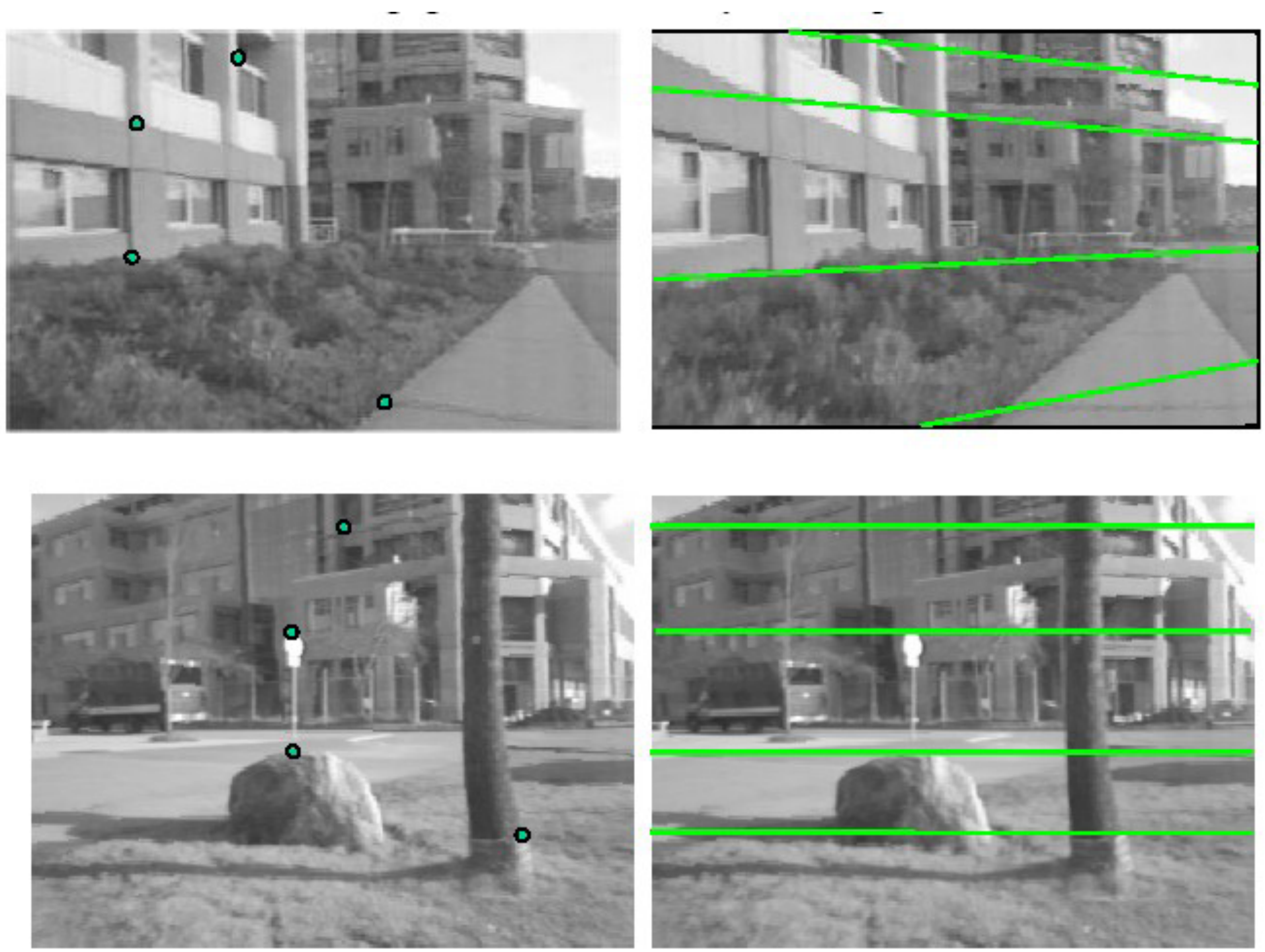
Epipolar geometry: terms

- **Baseline**: line joining the camera centers
- **Epipole**: point of intersection of baseline with the image plane
- **Epipolar plane**: plane containing baseline and world point
- **Epipolar line**: intersection of epipolar plane with the image plane

- All epipolar lines intersect at the epipole
- An epipolar plane intersects the left and right image planes in epipolar lines

Slide credit: Kristen Grauman

Example



Slide credit: Kristen Grauman

Example: converging cameras

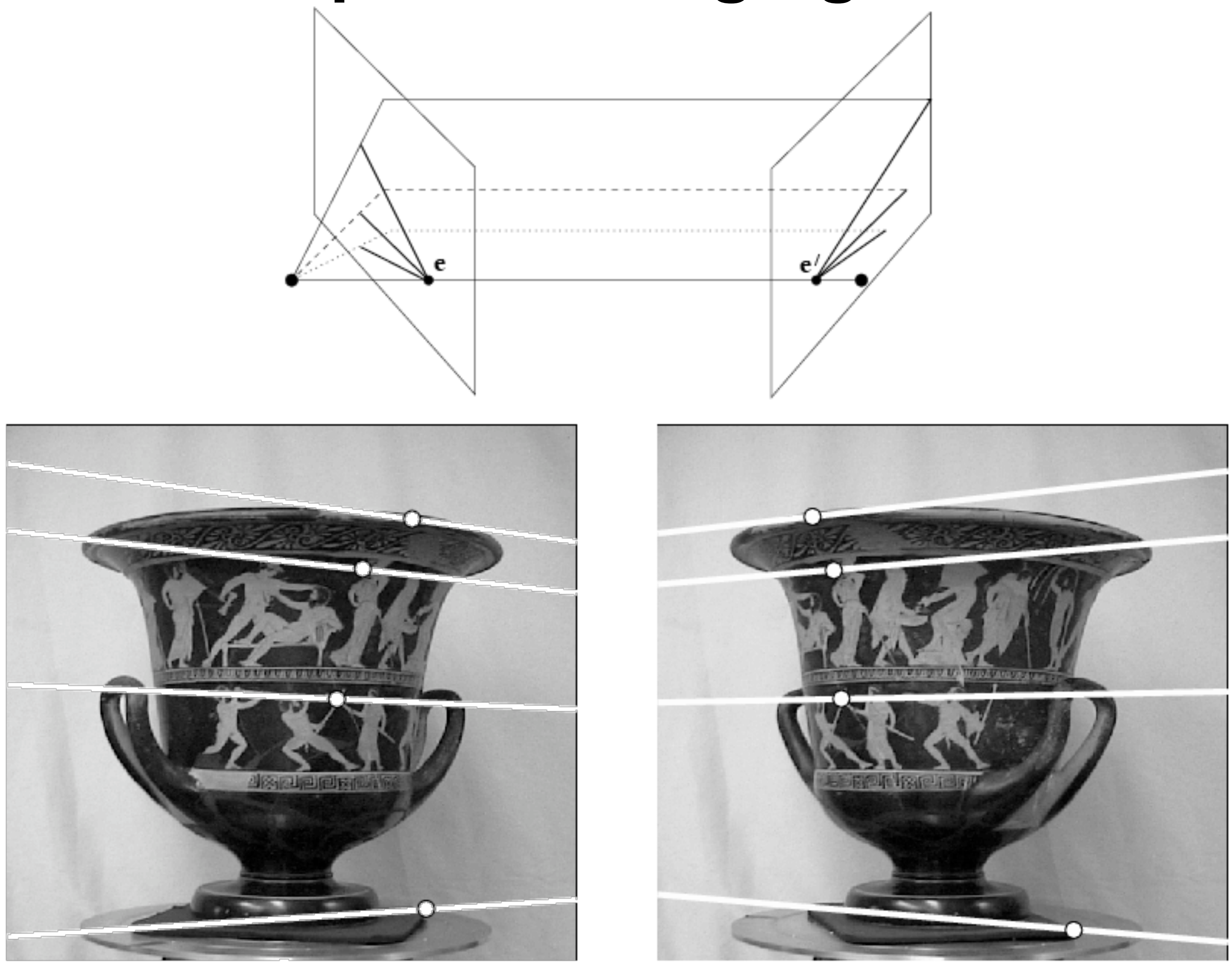
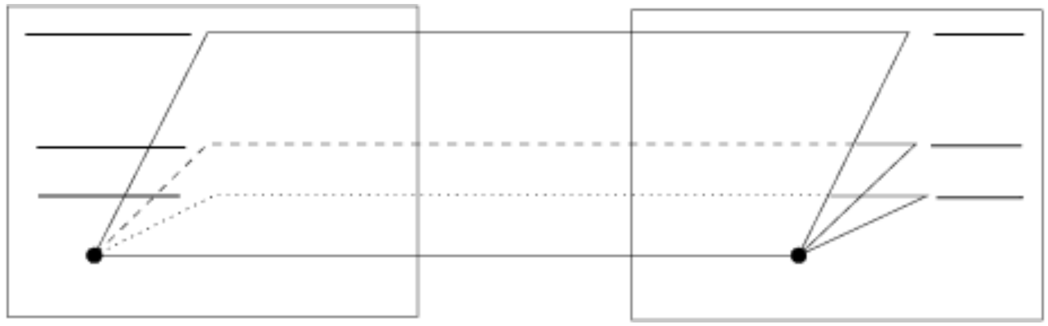


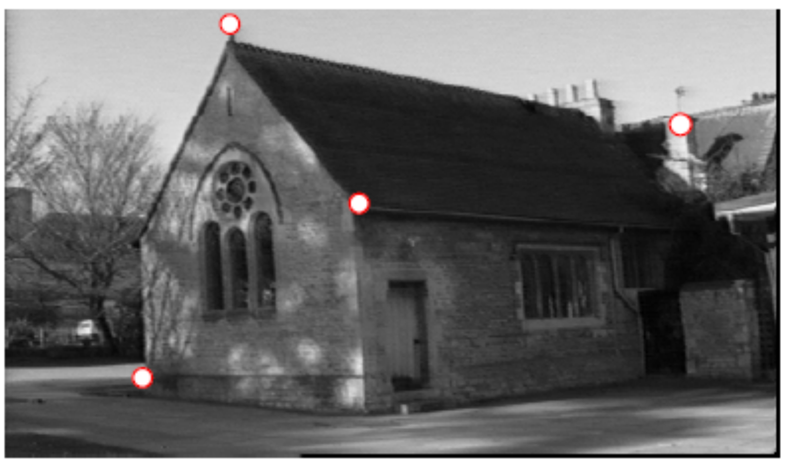
Figure from Hartley & Zisserman

Slide credit: Kristen Grauman

Example: parallel cameras



Where are the epipoles?



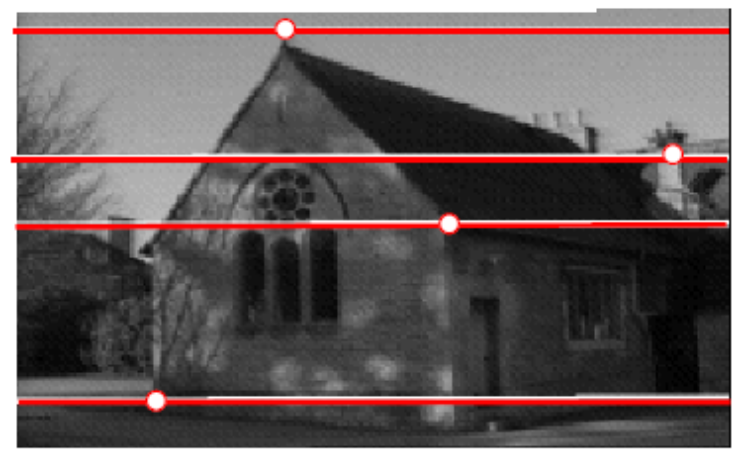


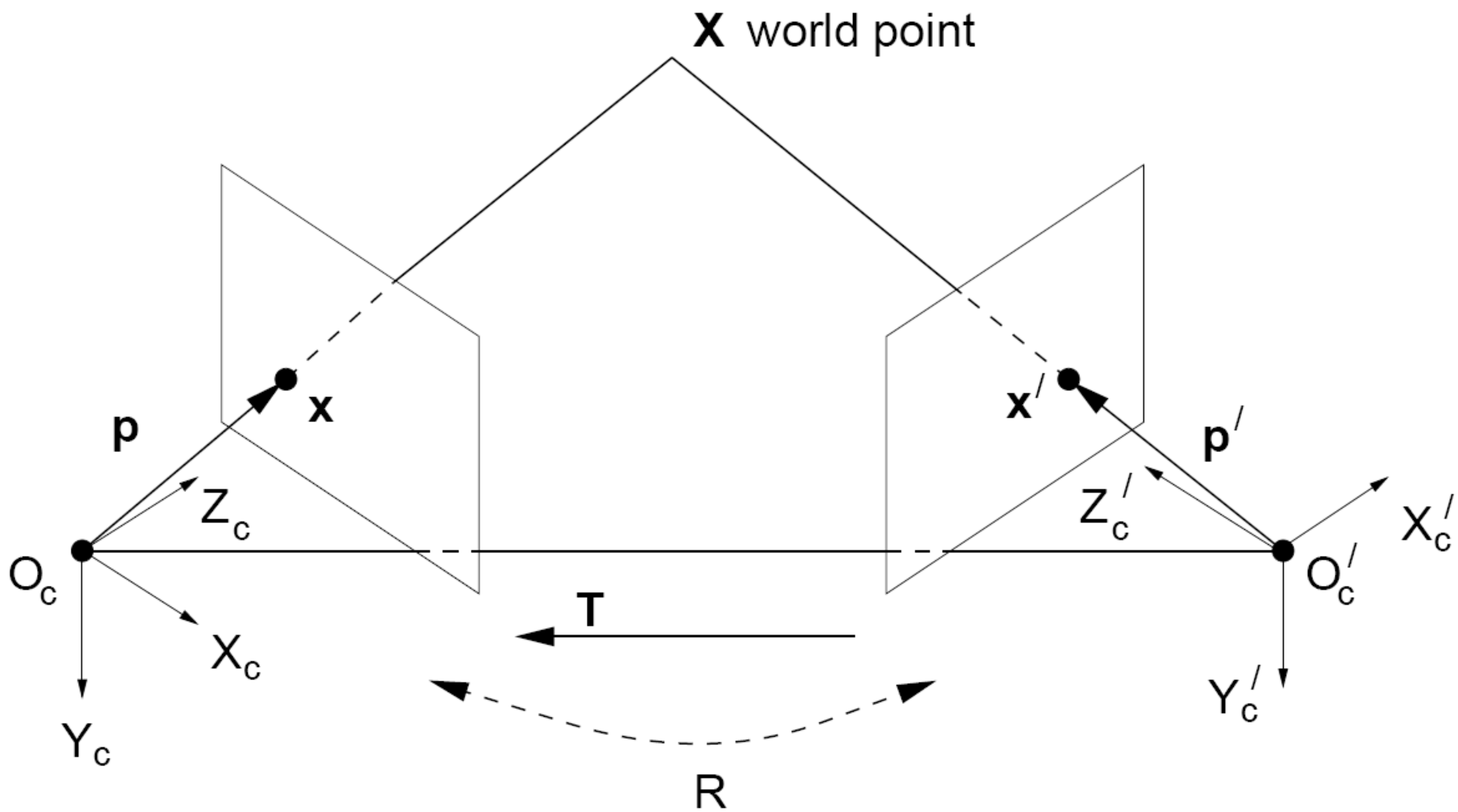
Figure from Hartley & Zisserman

Slide credit: Kristen Grauman

- So far, we have the explanation in terms of geometry.
- Now, how to express the epipolar constraints algebraically?

Slide credit: Kristen Grauman

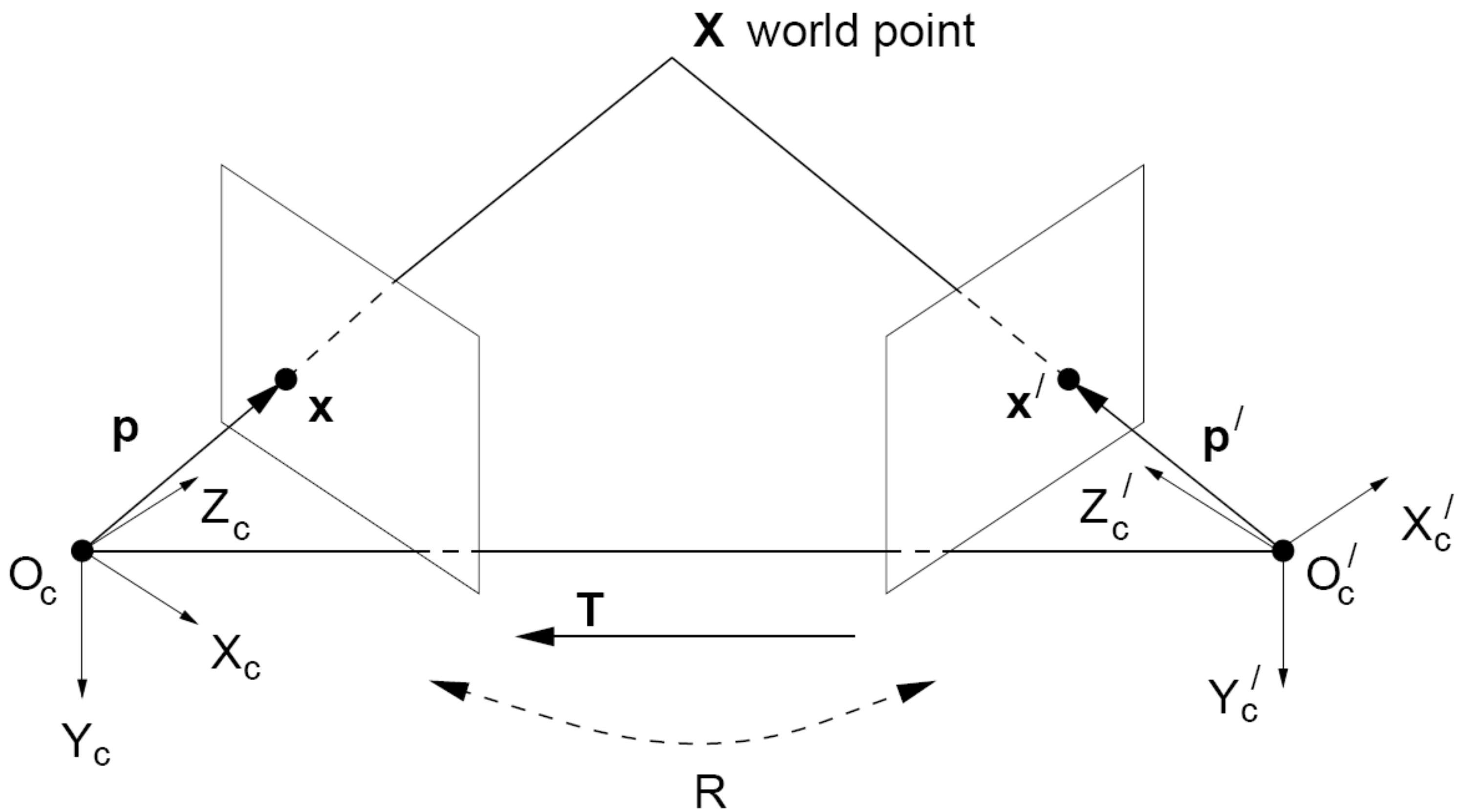
Stereo geometry, with calibrated cameras



Main idea

Slide credit: Kristen Grauman

Stereo geometry, with calibrated cameras



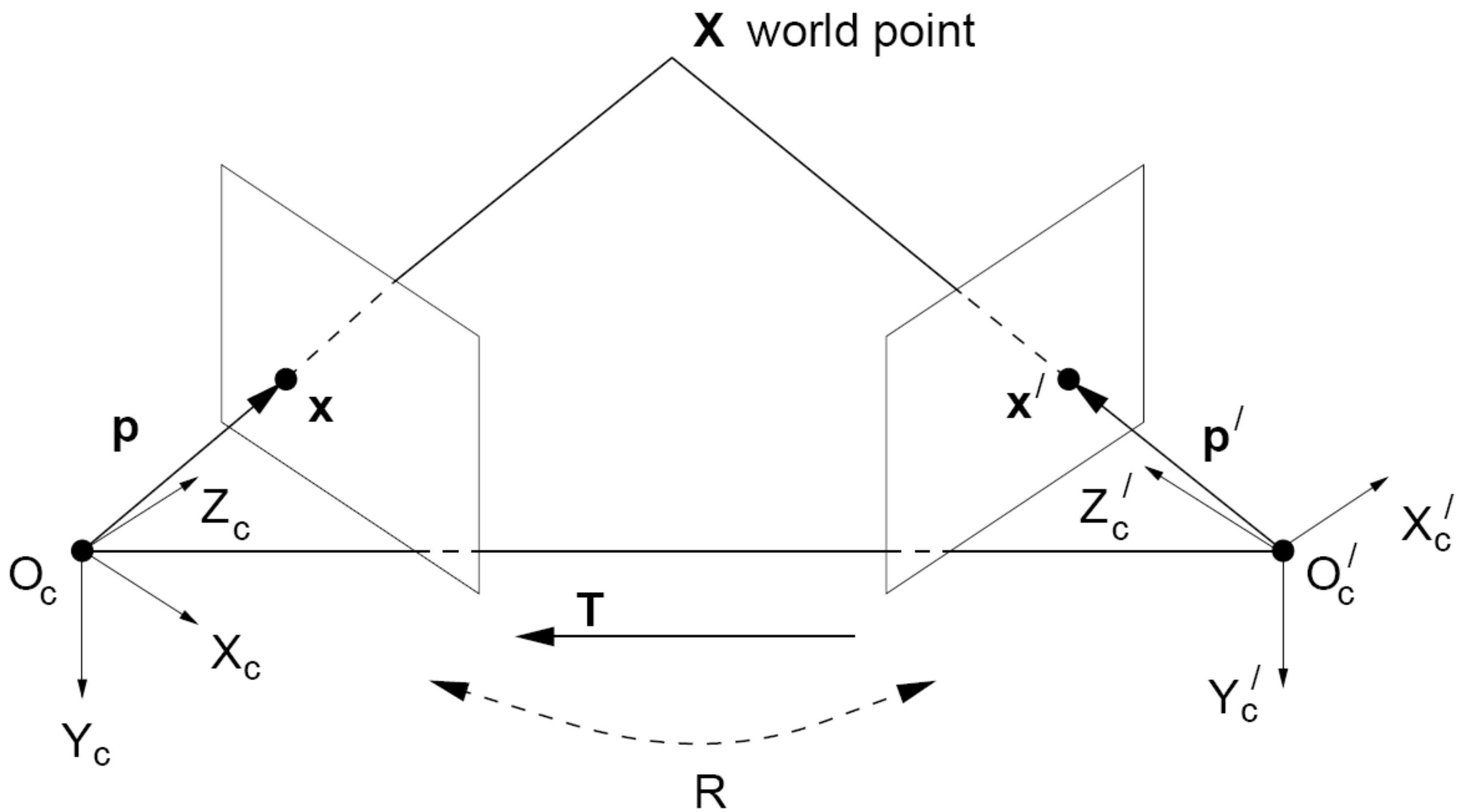
If the stereo rig is calibrated, we know :

how to rotate and translate camera reference frame 1 to get to camera reference frame 2.

Rotation: 3 x 3 matrix R; translation: 3 vector T.

Slide credit: Kristen Grauman

Stereo geometry, with calibrated cameras



If the stereo rig is calibrated, we know :

how to rotate and translate camera reference frame 1 to get to camera reference frame 2.

$$\mathbf{X}'_c = \mathbf{R}\mathbf{X}_c + \mathbf{T}$$

Slide credit: Kristen Grauman

An aside: cross product

$$\vec{a} \times \vec{b} = \vec{c}$$

$$\vec{a} \cdot \vec{c} = 0$$

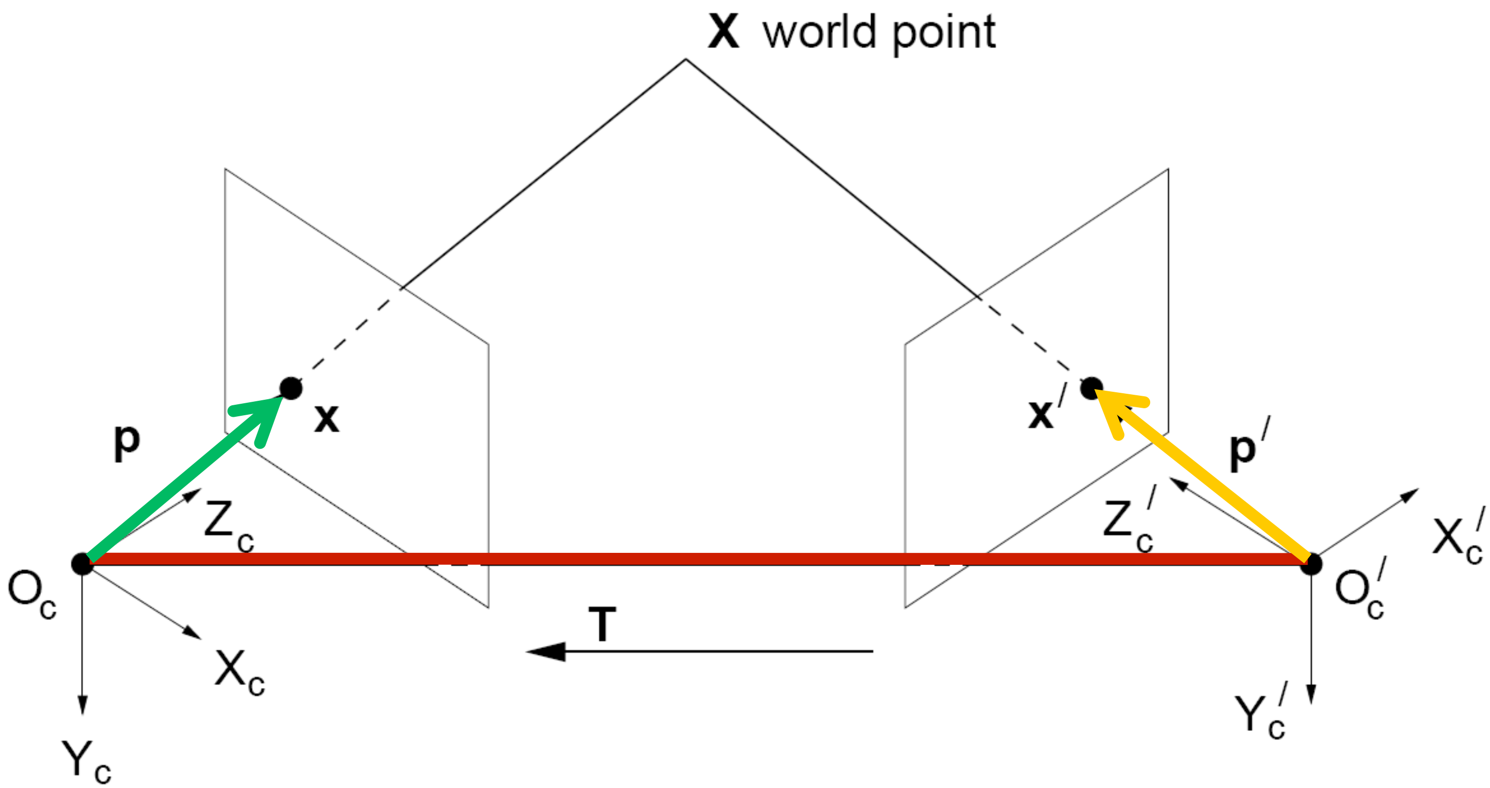
$$\vec{b} \cdot \vec{c} = 0$$

Vector cross product takes two vectors and returns a third vector that's perpendicular to both inputs.

So here, c is perpendicular to both a and b, which means the dot product = 0.

Slide credit: Kristen Grauman

From geometry to algebra



$\mathbf{X'} = \mathbf{R}\mathbf{X} + \mathbf{T}$

$$\underbrace{\mathbf{T} \times \mathbf{X'}}_{\text{Normal to the plane}} = \mathbf{T} \times \mathbf{R}\mathbf{X}$$

$$\mathbf{X'} \cdot (\mathbf{T} \times \mathbf{X'}) = \mathbf{X'} \cdot (\mathbf{T} \times \mathbf{R}\mathbf{X})$$

$$= 0$$

Slide credit: Kristen Grauman

Another aside: Matrix form of cross product

$$\vec{a} \times \vec{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \vec{c}$$

$$\vec{a} \cdot \vec{c} = 0$$

$$\vec{b} \cdot \vec{c} = 0$$

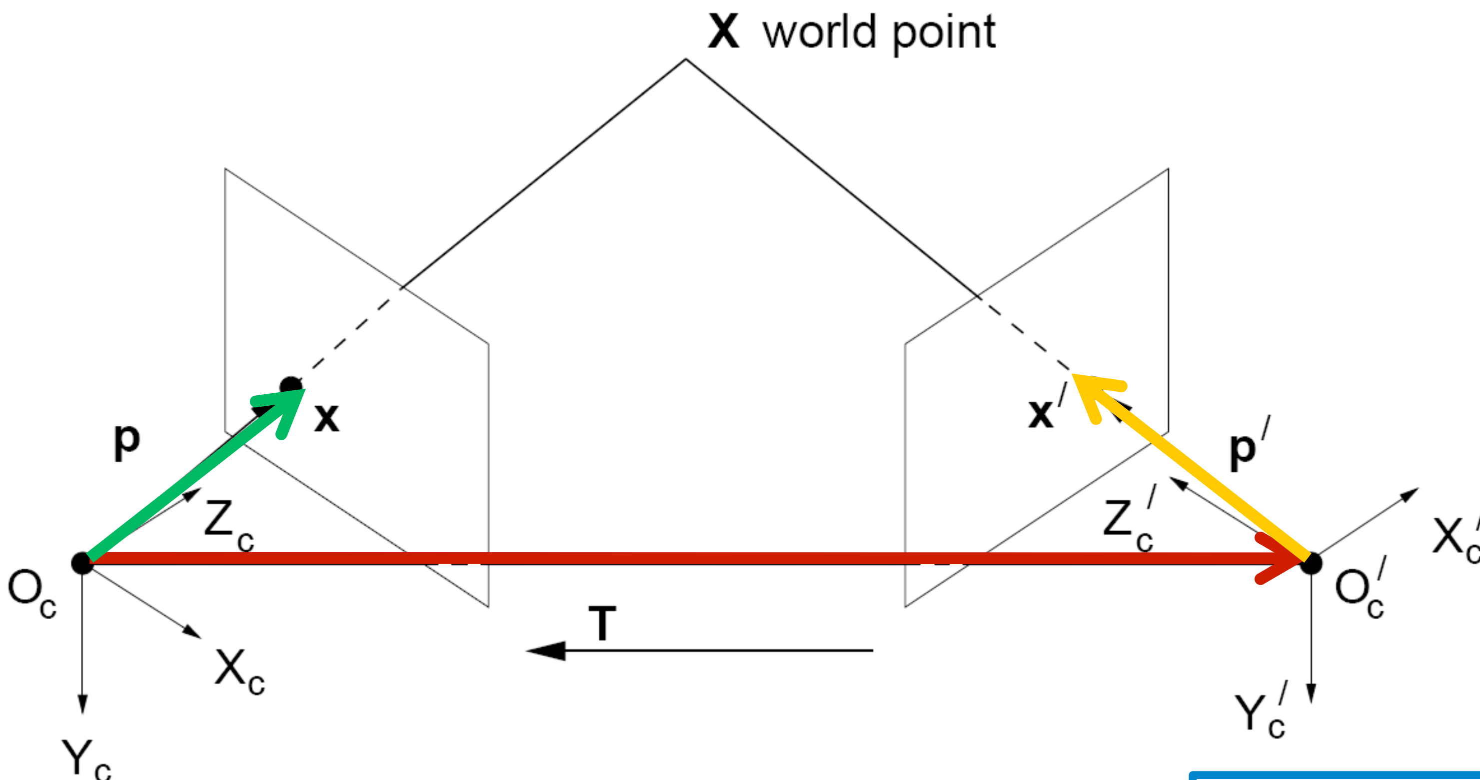
Can be expressed as a matrix multiplication.

$$[a_x] = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$$

$$\boxed{\vec{a} \times \vec{b} = [a_x]\vec{b}}$$

Slide credit: Kristen Grauman

From geometry to algebra



$$\mathbf{X'} = \mathbf{RX} + \mathbf{T}$$

$$\underbrace{\mathbf{T} \times \mathbf{X}'}_{\text{Normal to the plane}} = \mathbf{T} \times \mathbf{RX} + \mathbf{T} \times \mathbf{T}$$

$$= \mathbf{T} \times \mathbf{RX}$$

$$\mathbf{X}' \cdot (\mathbf{T} \times \mathbf{X}') = \mathbf{X}' \cdot (\mathbf{T} \times \mathbf{RX})$$

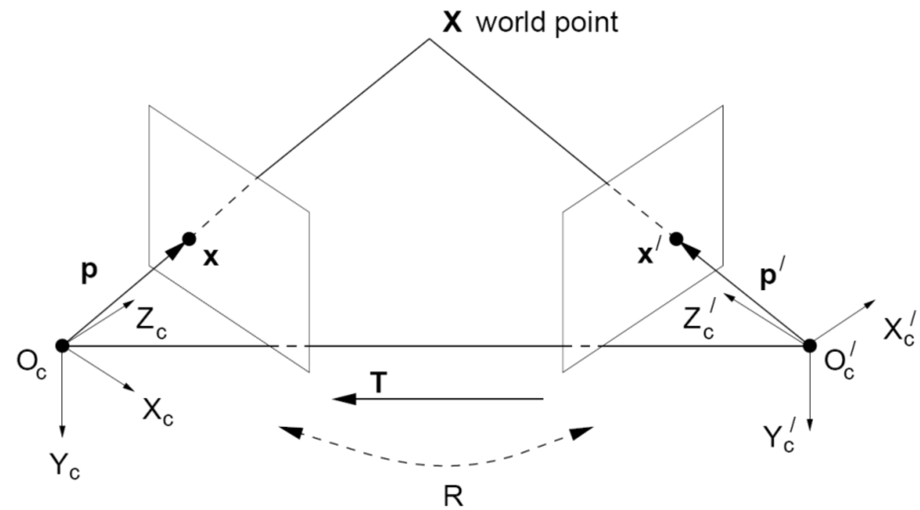
$$= 0$$

Slide credit: Kristen Grauman

Essential matrix

$$\mathbf{X}' \cdot (\mathbf{T} \times \mathbf{R}\mathbf{X}) = 0$$

$$\mathbf{X}' \cdot (\mathbf{T}_x \ \mathbf{R}\mathbf{X}) = 0$$



Let $\mathbf{E} = \mathbf{T}_x\mathbf{R}$

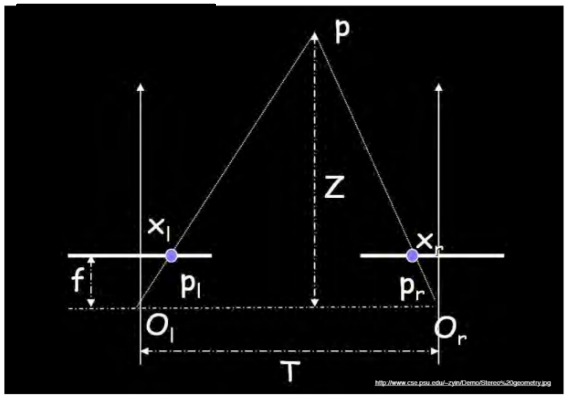
$$\mathbf{X}'^{T}\mathbf{E}\mathbf{X} = 0$$

E is called the essential matrix, and it relates corresponding image points between both cameras, given the rotation and translation.

If we observe a point in one image, its position in other image is constrained to lie on line defined by above.

Note: these points are in camera coordinate systems.

Essential matrix example: parallel cameras



$\mathbf{R} =$

$\mathbf{T} =$

$\mathbf{E} = [\mathbf{T_x}]\mathbf{R} =$

$$\mathbf{p} = [x, y, f]$$

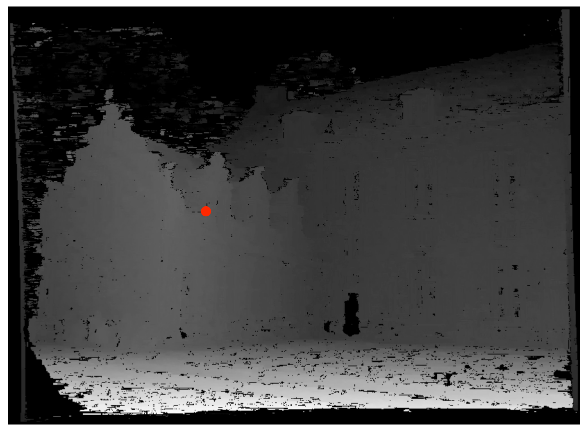
$$\mathbf{p'} = [x', y', f]$$

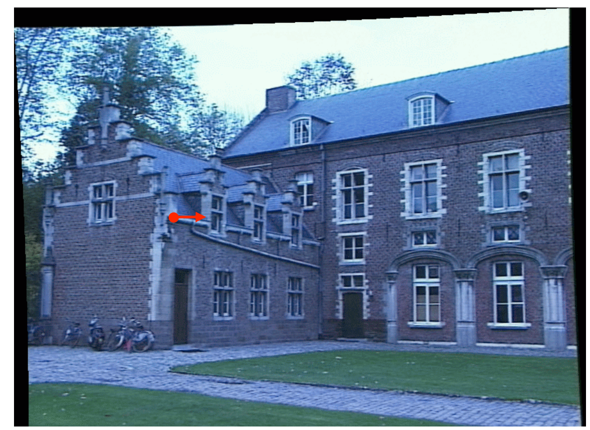
$$\mathbf{p}'^{\mathrm{T}}\mathbf{E}\mathbf{p} = 0$$

For the parallel cameras, image of any point must lie on same horizontal line in each image plane.

Slide credit: Kristen Grauman

image I(x,y) **Disparity map D(x,y)** **image I´(x´,y´)**





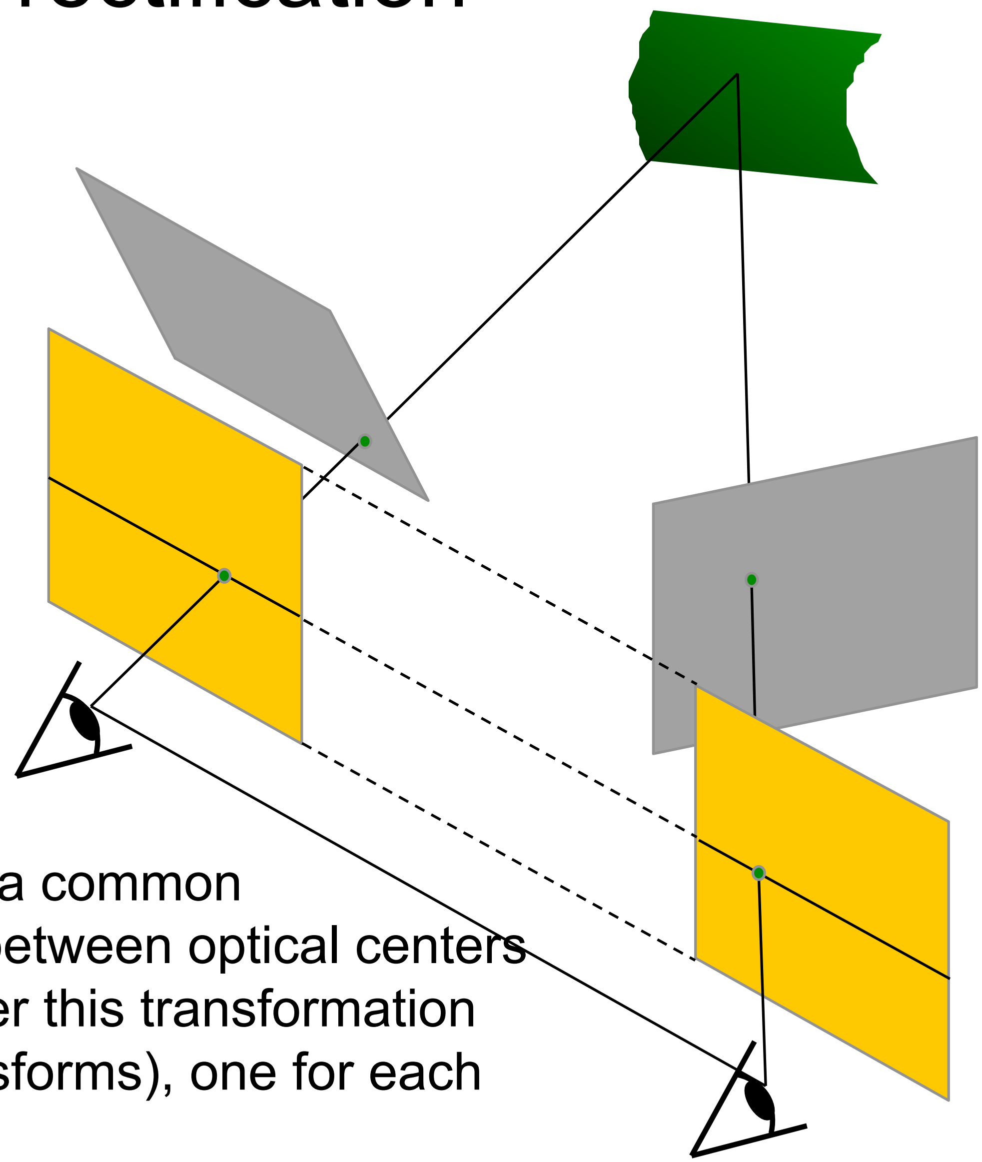
$$(x',y')=(x+D(x,y),y)$$

What about when cameras' optical axes are not parallel?

Slide credit: Kristen Grauman

Stereo image rectification

In practice, it is convenient if image scanlines (rows) are the epipolar lines.

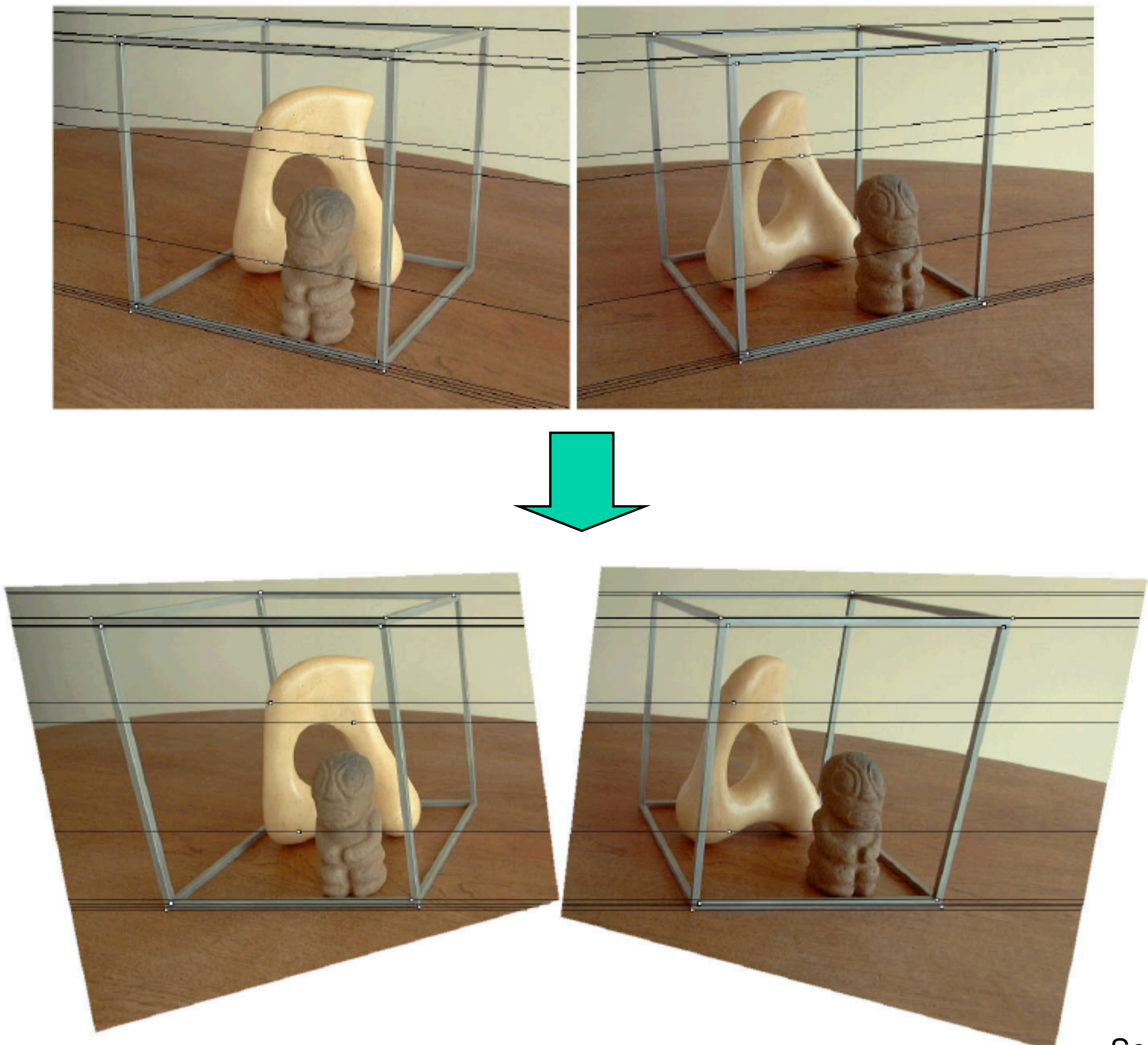


- reproject image planes onto a common plane parallel to the line between optical centers
- pixel motion is horizontal after this transformation
- two homographies (3x3 transforms), one for each input image reprojection

Adapted from Li Zhang

Slide credit: Kristen Grauman

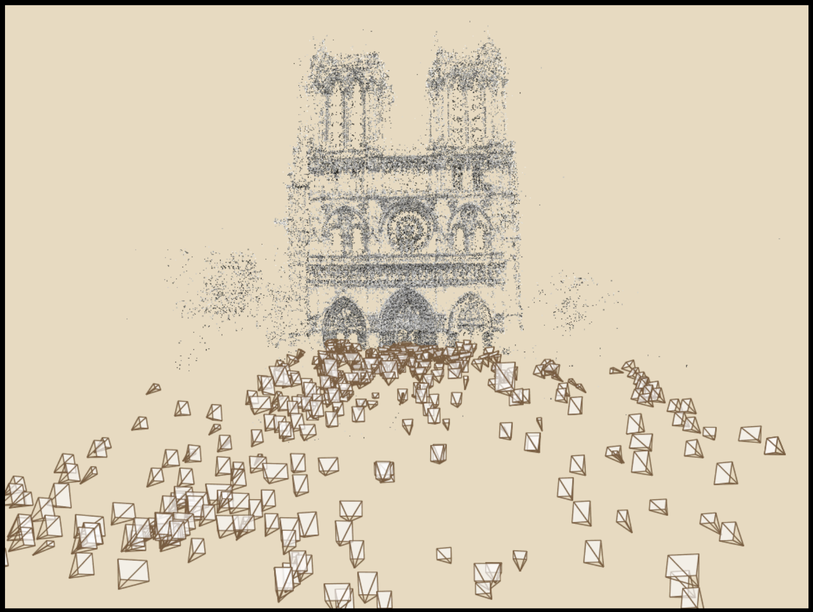
Stereo image rectification: example



Source: Alyosha Efros

Multiview geometry

Structure from motion (SfM)



- N. Snavely, S. M. Seitz, R. Szeliski, 2007
- M. Vergauwen, L. Van Gool, 2006
- M. Brown, D. Lowe, 2005
- F. Schaffalitzky, A. Zisserman, 2002

Dense multiview stereo

- Y. Furukawa, J. Ponce, 2009
- P. Labatut, J.-P. Pons, R. Keriven, 2009
- M. Goesele, N. Snavely, B. Curless, H. Hoppe, S. M. Seitz., 2007