Computer Vision

Marc Pollefeys

Luc Van Gool

Vittorio Ferrari

The team

Marc Pollefeys

Luc Van Gool

Vittorio Ferrari

Pablo Speciale

Administrivia

- Classes: Wed, 13-16, CHN C 14
- Exercises: Thu, 15-16, CHN C 14

- Instructors: Marc Pollefeys
  Luc Van Gool
  Vittorio Ferrari

- Main assistant: Pablo Speciale

- Prerequisite: Visual Computing (or equivalent)

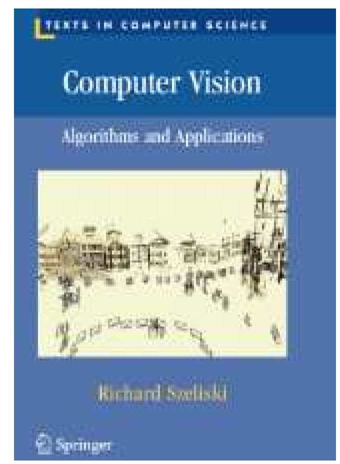
(if missing background, probably better to choose:
Image Analysis and Computer Vision, Székely, Van Gool)

Material

- Slides

- Webpage (with slides, assignments.)
  http://www.cvg.ethz.ch/teaching/compvis/

- Reference book:
  Computer Vision:
  Algorithms and Applications
  by Rick Szeliski
  http://szeliski.org/Book/



Also Simon Prince's book, available free online
http://computervisionmodels.blogspot.com/

What is computer vision?

Done?

What is computer vision?

- Automatic understanding of images and video
  - Computing properties of the 3D world from visual data *(measurement)*
  - Algorithms and representations to allow a machine to recognize objects, people, scenes, and activities. *(perception and interpretation)*

Lots of slides borrowed from Svetlana Lazebnik, Trevor Darrell

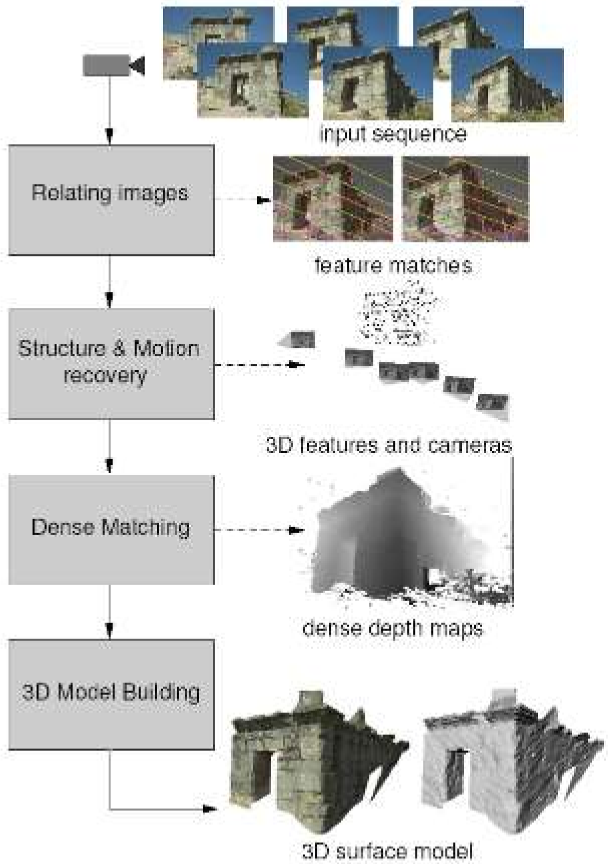
Vision for measurement

Real-time stereo

NASA Mars Rover

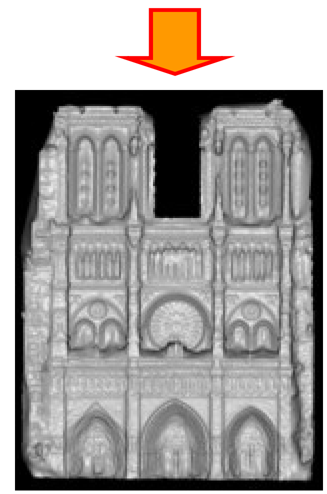
Yang and Pollefeys

Structure from motion



Pollefeys et al

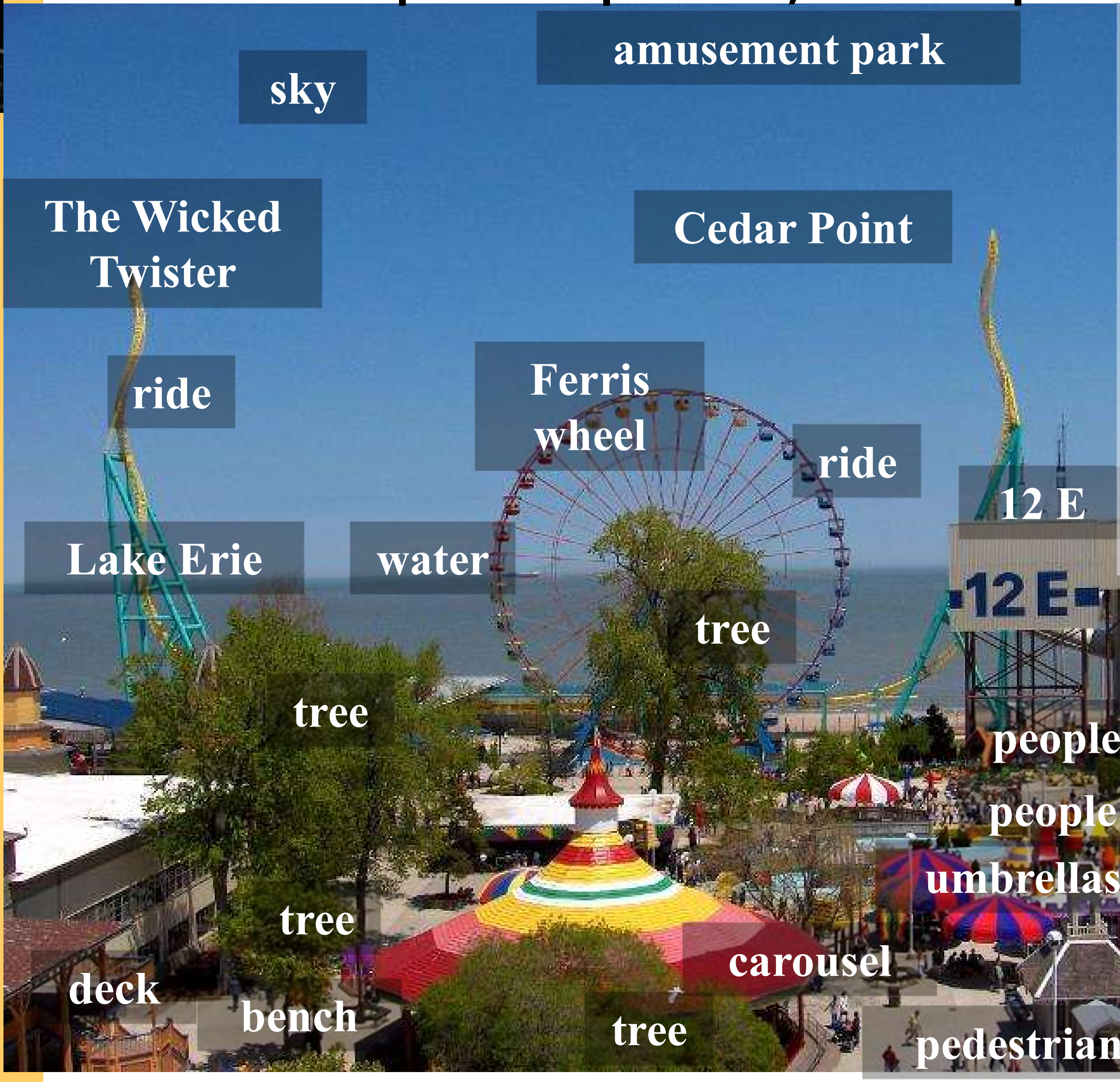
Multi-view stereo for community photo collections



Goesele et al.

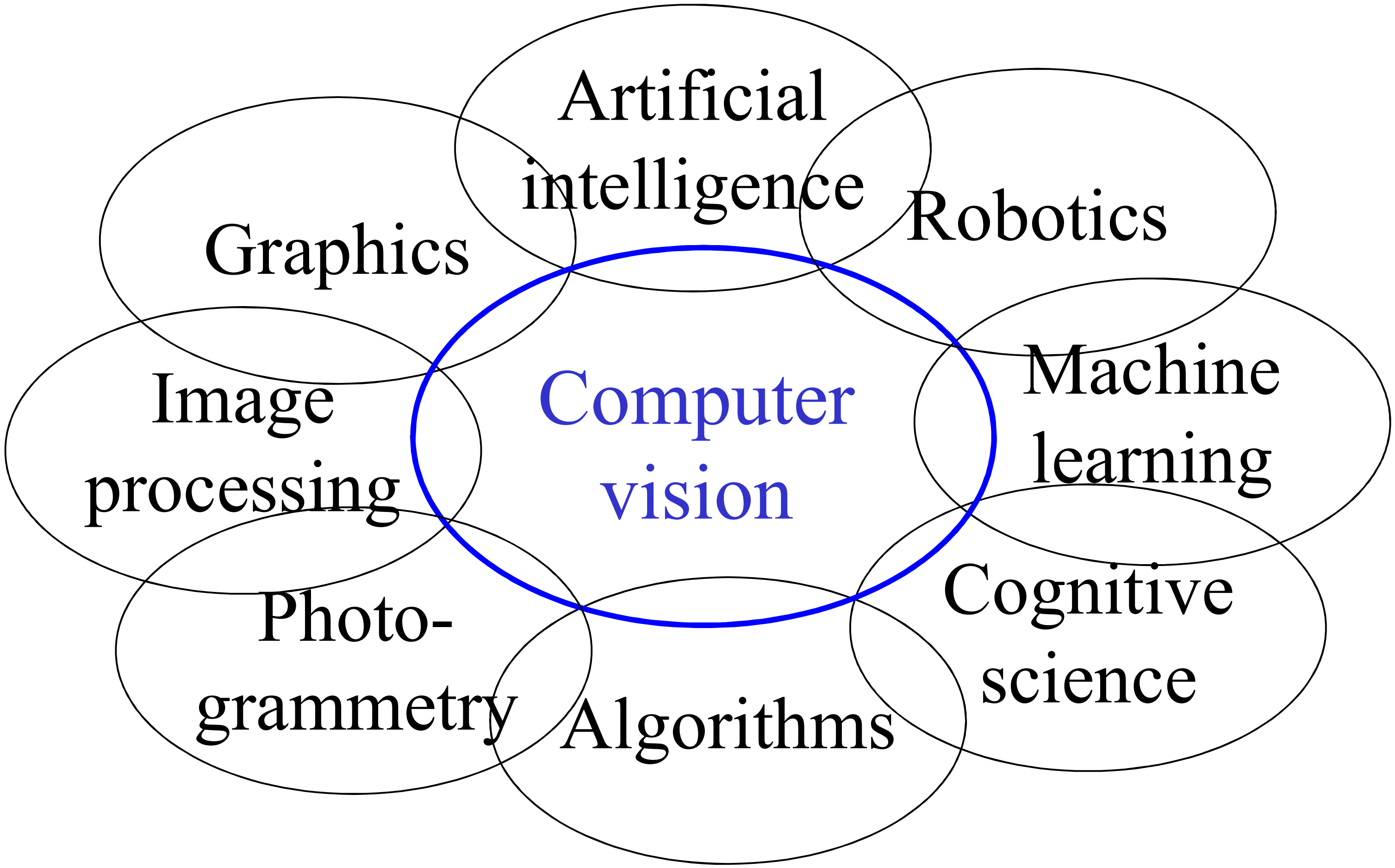
Slide credit: L. Lazebnik

Vision for perception, interpretation

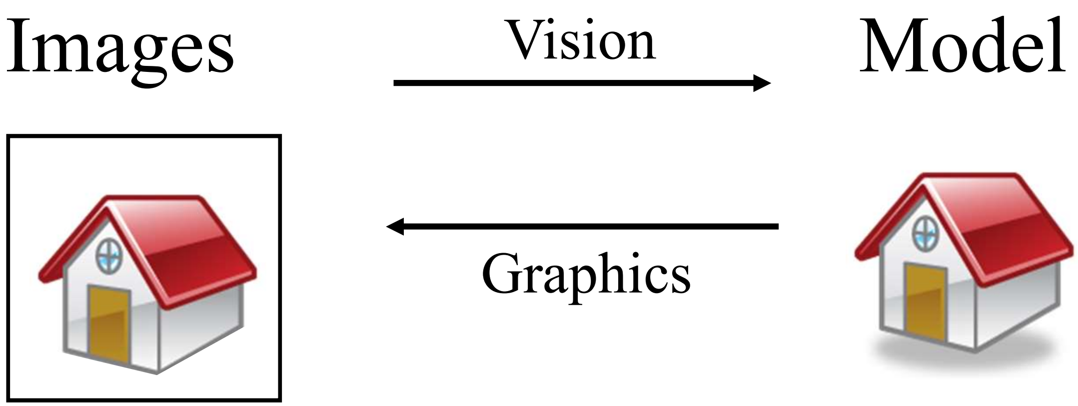


Objects
Activities
Scenes
Locations
Text / writing
Faces
Gestures
Motions
Emotions…

Related disciplines



Vision and graphics



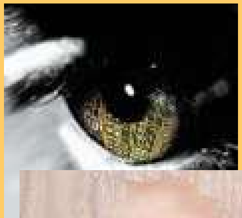
Inverse problems: analysis and synthesis.

Why vision?

- As image sources multiply, so do applications
  - Relieve humans of boring, easy tasks
  - Enhance human abilities: human-computer interaction, visualization
  - Perception for robotics / autonomous agents
  - Organize and give access to visual content

Why vision?

Images and video are everywhere!

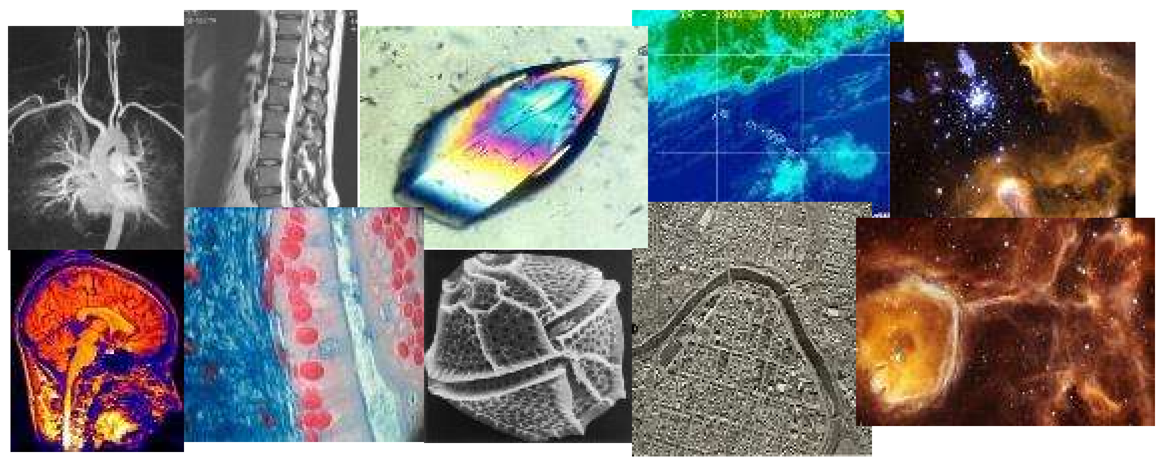


Personal photo albums

Movies, news, sports

Google Image Search · Picasa · flickr · webshots · picsearch · YouTube Broadcast Yourself

Surveillance and security



Medical and scientific images

Slide credit; L. Lazebnik

Why is vision hard?

Grayscale Image

y = \ x =	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72
41	210	209	204	202	197	247	143	71	64	80	84	54	54	57	58
42	206	196	203	197	195	210	207	56	63	58	53	53	61	62	51
43	201	207	192	201	198	213	156	69	65	57	55	52	53	60	53
44	216	206	211	193	202	207	208	57	69	60	55	77	49	62	51
45	221	206	211	194	196	197	220	56	63	60	55	46	97	58	106
46	209	214	224	199	194	193	204	173	64	60	59	51	62	56	48
47	204	212	213	208	191	190	191	214	60	62	66	76	51	49	55
48	214	215	215	207	208	180	172	188	69	72	55	49	56	52	56
49	209	205	214	205	204	196	187	196	86	62	66	87	57	60	48
50	208	209	205	203	202	186	174	185	149	71	63	55	55	45	56
51	207	210	211	199	217	194	183	177	209	90	62	64	52	93	52
52	208	205	209	209	197	194	183	187	187	239	58	68	61	51	56
53	204	206	203	209	195	203	188	185	183	221	75	61	58	60	60
54	200	203	199	236	188	197	183	190	183	196	122	63	58	64	66
55	205	210	202	203	199	197	196	181	173	186	105	62	57	64	63

How do we go from an array of numbers recognizing fruit?

slide credit: Michael Black

Challenges: viewpoint variation

Michelangelo 1475-1564

slide credit: Fei-Fei, Fergus & Torralba

Challenges: illumination

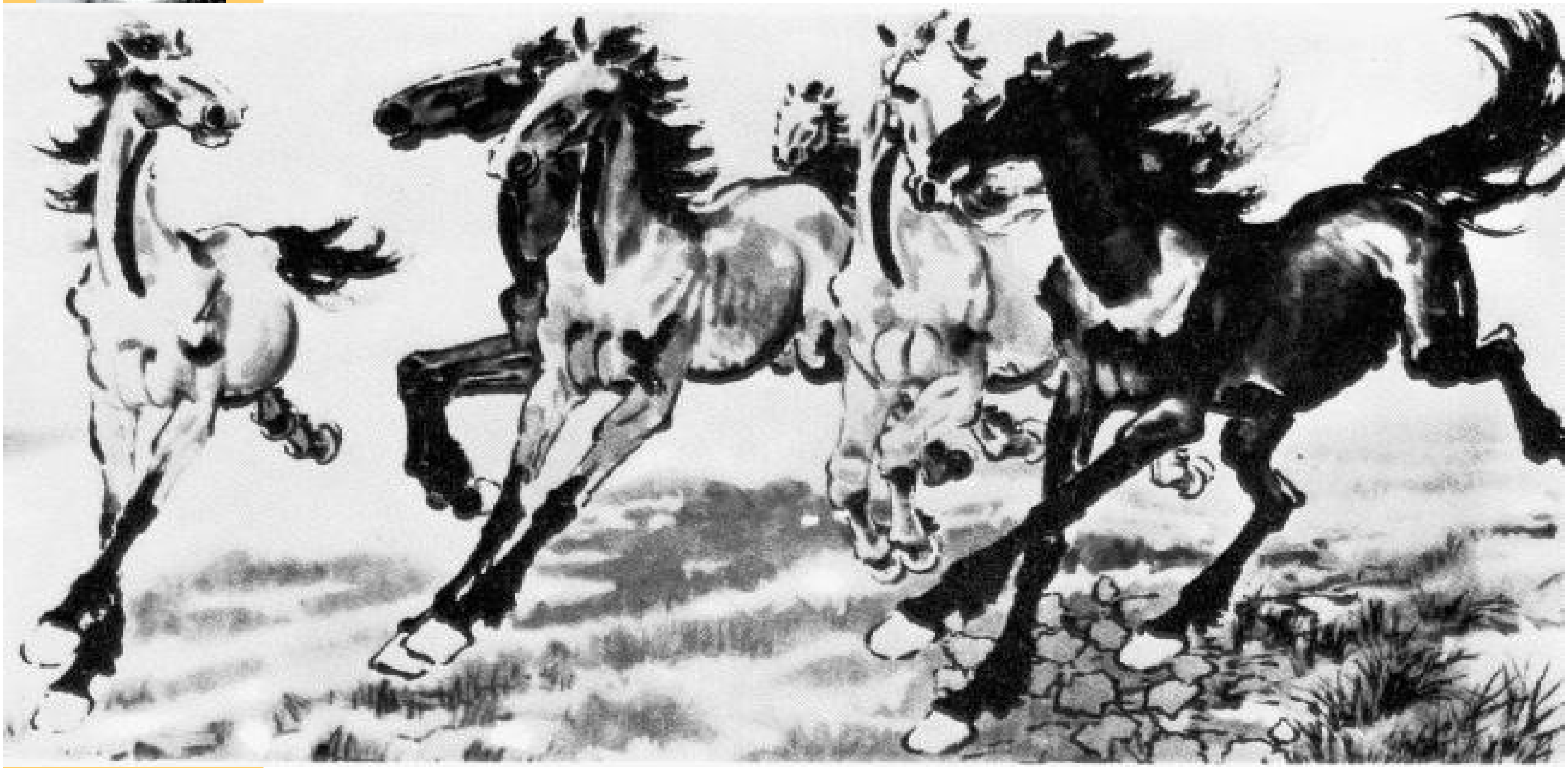
The picture can't be displayed.

image credit: J. Koenderink

Challenges: scale

slide credit: Fei-Fei, Fergus & Torralba

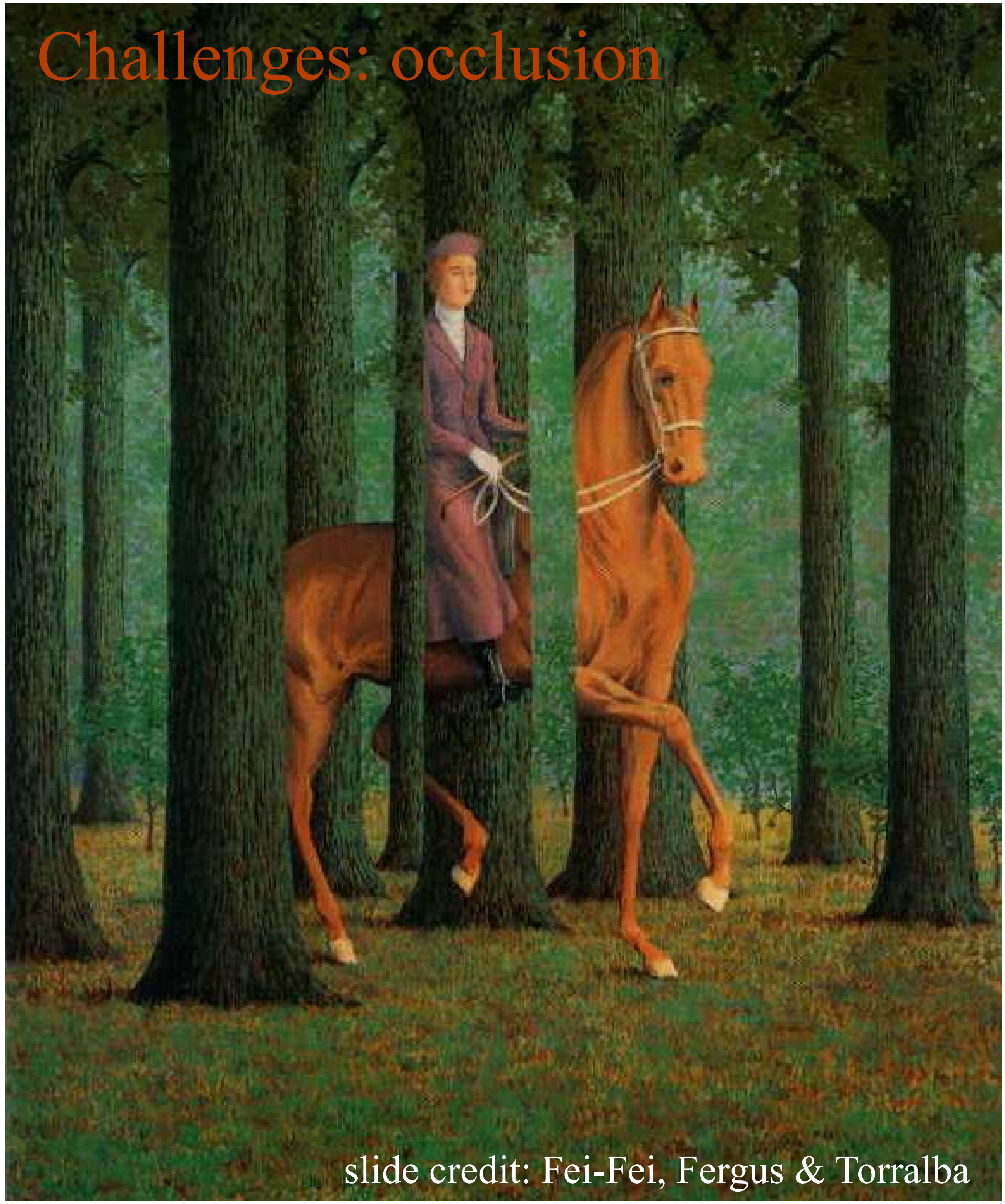
Challenges: deformation



Xu, Beihong 1943

slide credit: Fei-Fei, Fergus & Torralba

Challenges: occlusion



Magritte, 1957

slide credit: Fei-Fei, Fergus & Torralba

Challenges: background clutter

slide credit: Svetlana Lazebnik

Challenges: Motion

Challenges: object intra-class variation

slide credit: Fei-Fei, Fergus & Torralba

Challenges: local ambiguity

slide credit: Fei-Fei, Fergus & Torralba

Challenges or opportunities?

- Images are confusing, but they also reveal the structure of the world through numerous cues
- Our job is to interpret the cues!

Image source: J. Koenderink

Depth cues: Linear perspective

slide credit: Svetlana Lazebnik

Depth cues: Aerial perspective

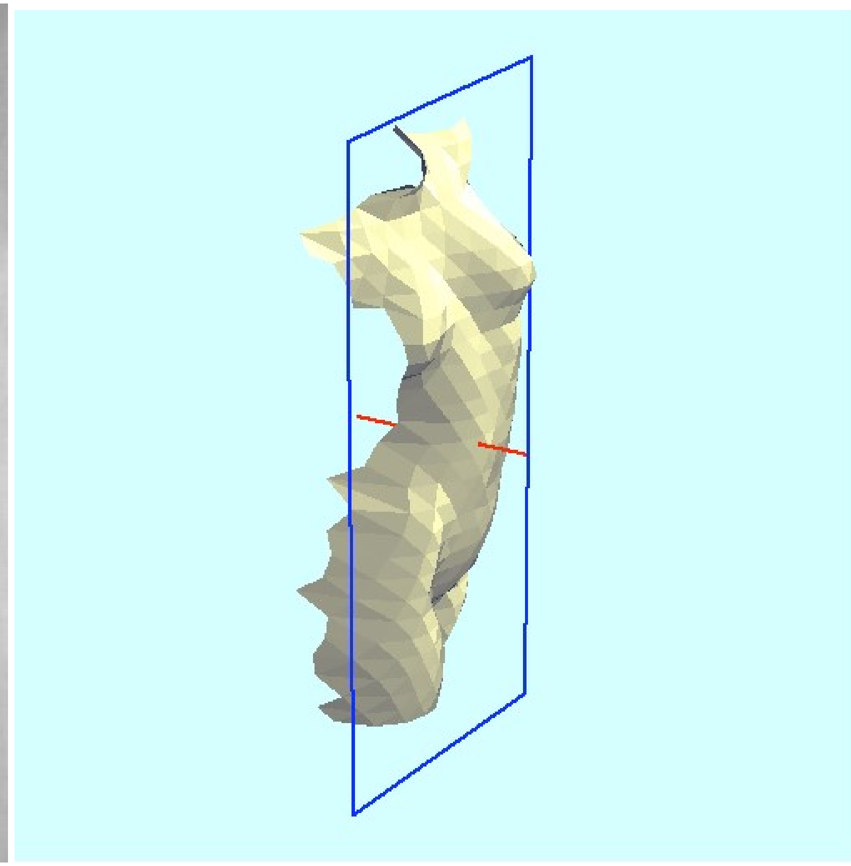
slide credit: Svetlana Lazebnik

Depth ordering cues: Occlusion

Shape cues: Texture gradient

slide credit: Svetlana Lazebnik

Shape and lighting cues: Shading



Source: J. Koenderink

Position and lighting cues: Cast shadows

Source: J. Koenderink

Grouping cues: Similarity (color, texture, proximity)

slide credit: Svetlana Lazebnik

Grouping cues: “Common fate”

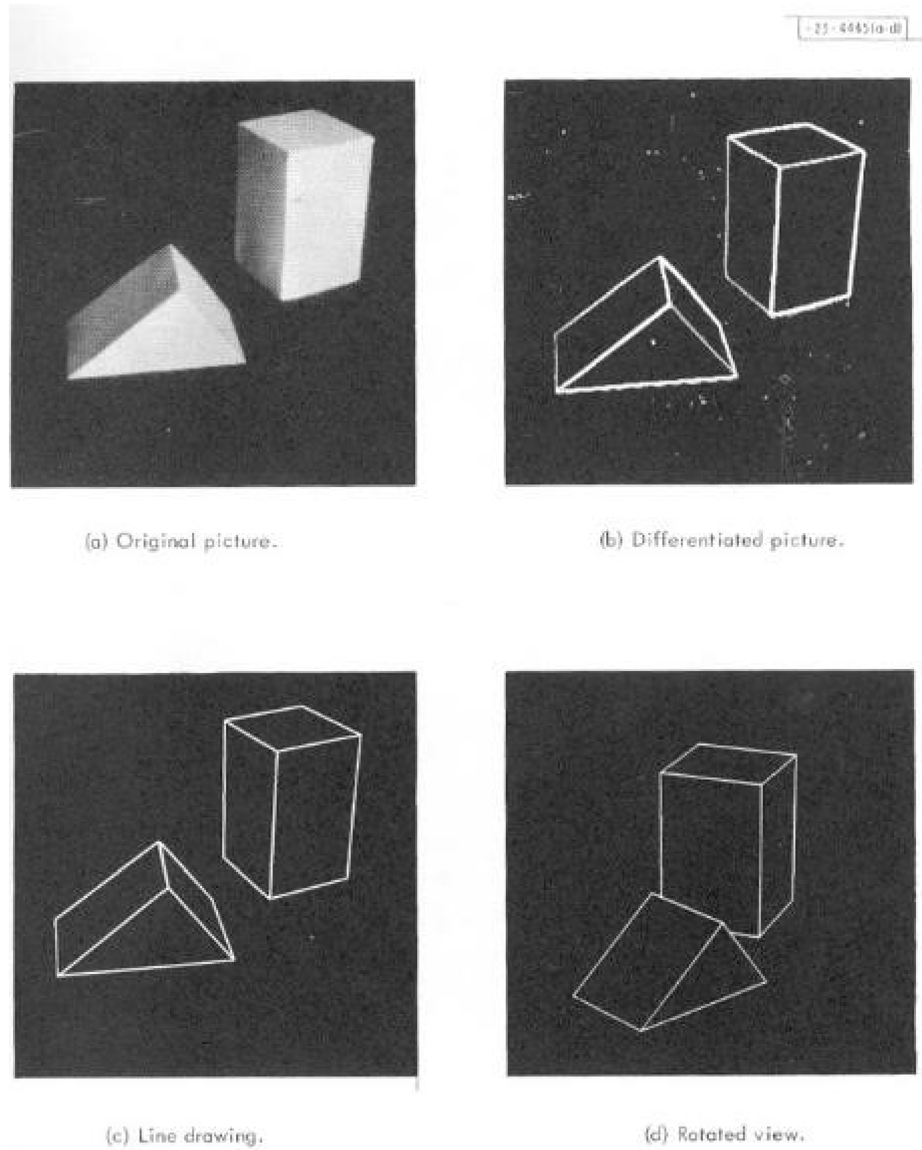
Image credit: Arthus-Bertrand (via F. Durand)

Bottom line

- Perception is an inherently ambiguous problem
  - Many different 3D scenes could have given rise to a particular 2D picture

- Possible solutions
  - Bring in more constraints (more images)
  - Use prior knowledge about the structure of the world
- Need a combination of different methods

Origins of computer vision



L. G. Roberts, *Machine Perception of Three Dimensional Solids,* Ph.D. thesis, MIT Department of Electrical Engineering, 1963.

Again, what is computer vision?

- Mathematics of geometry of image formation?
- Statistics of the natural world?
- Models for neuroscience?
- Engineering methods for matching images?
- Science Fiction?

Vision Demo?

Terminator 2

we're not quite there yet….

Every picture tells a story

Goal of computer vision is to write computer programs that can interpret images

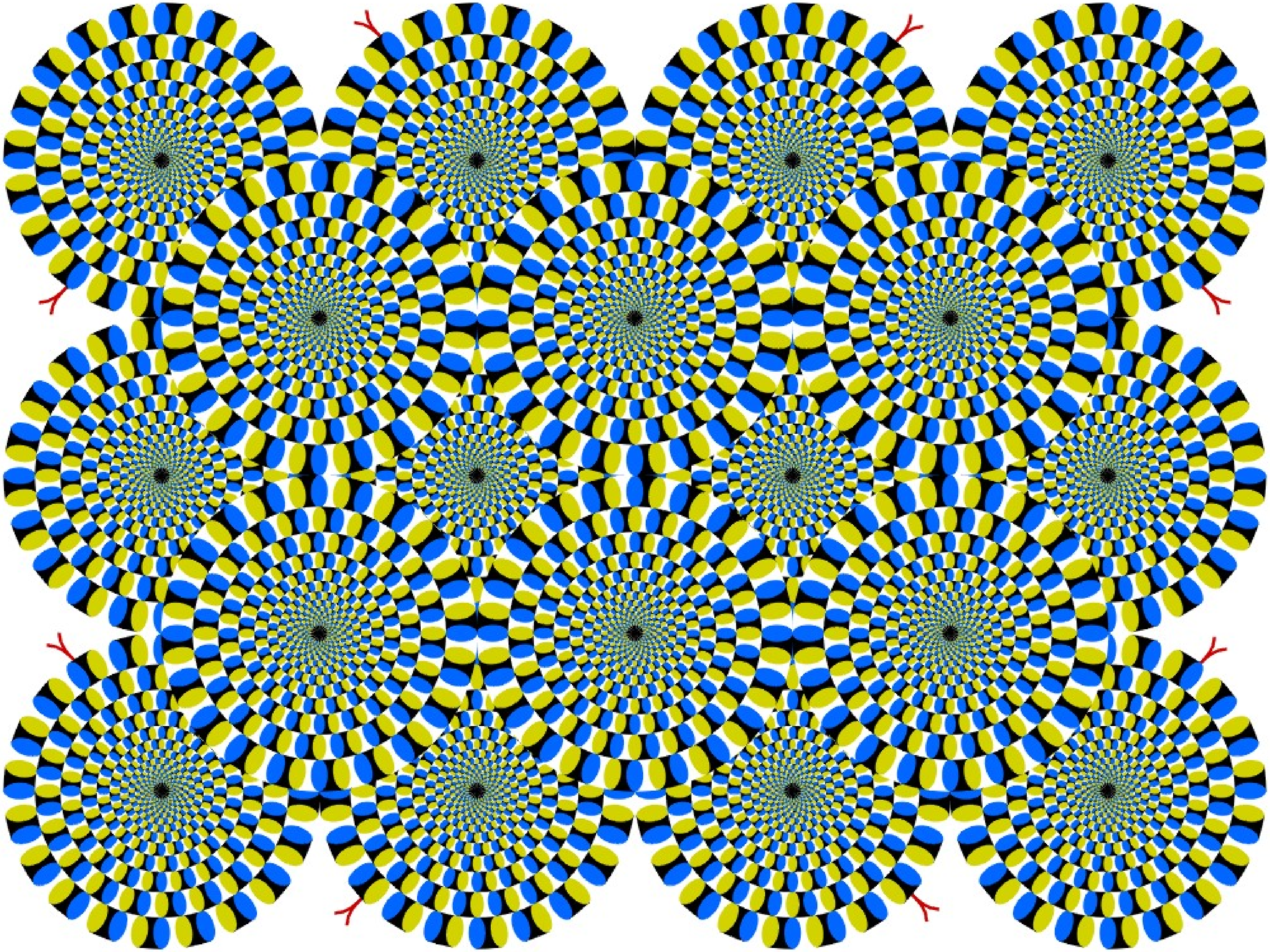
Can computers match (or beat) human vision?

Yes and no (but mostly no!)

- humans are much better at “hard” things
- computers can be better at “easy” things

Human perception has its shortcomings…

[Sinha and Poggio, *Nature*, 1996]

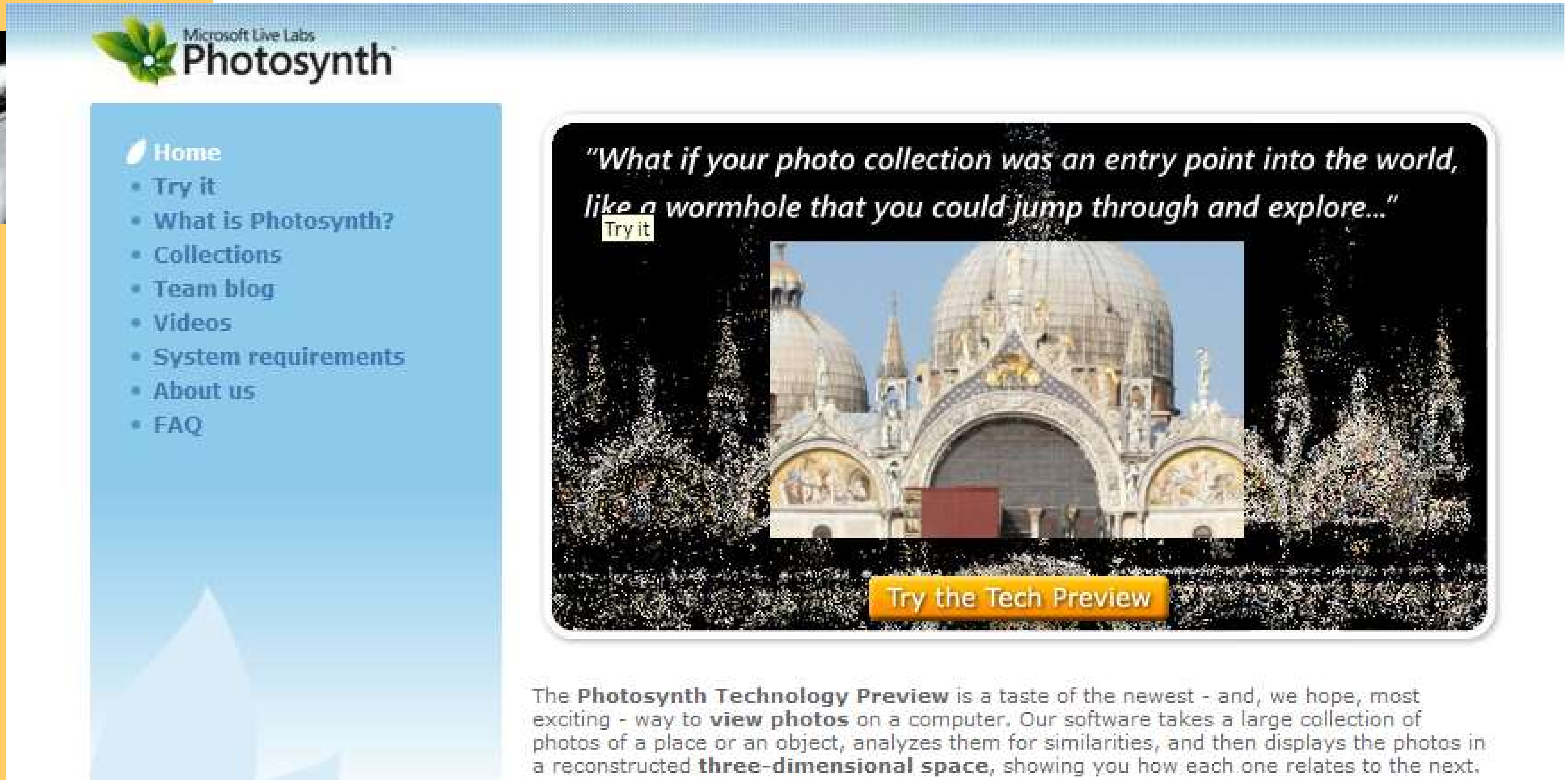


Computer vision applications

- The next slides show some examples of what current vision systems can do

Earth viewers (3D modeling)

Example from: Google Earth



http://photosynth.net/

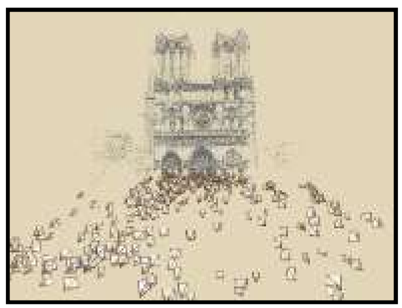
Based on Photo Tourism technology developed by Noah Snavely, Steve Seitz, and Rick Szeliski

PhotoSynth example

Photo Tourism overview

Input photographs

Scene reconstruction



Relative camera positions and orientations

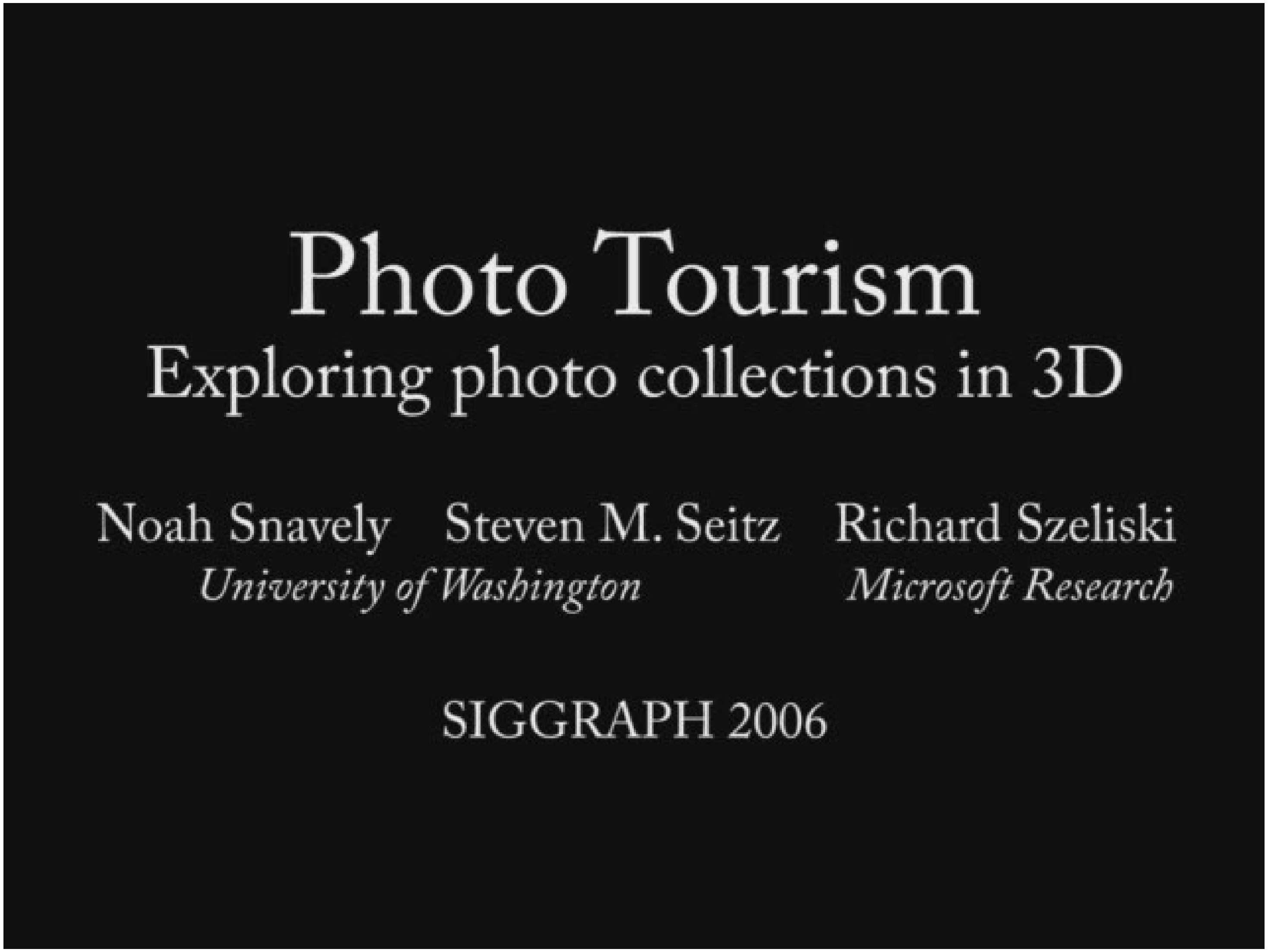
Point cloud

Sparse correspondence

Photo Explorer

System for interactive browsing and exploring large collections of photos of a scene.
Computes viewpoint of each photo as well as a sparse 3d model of the scene.

Photo Tourism overview

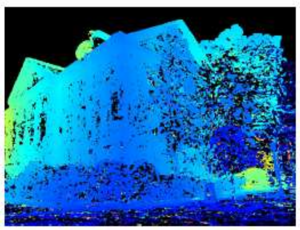


Urban 3D Reconstruction

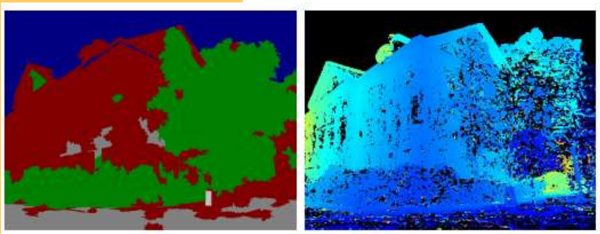
(Gallup, Frahm, Pollefeys DAGM2010)

Joint 3D reconstruction and class segmentation

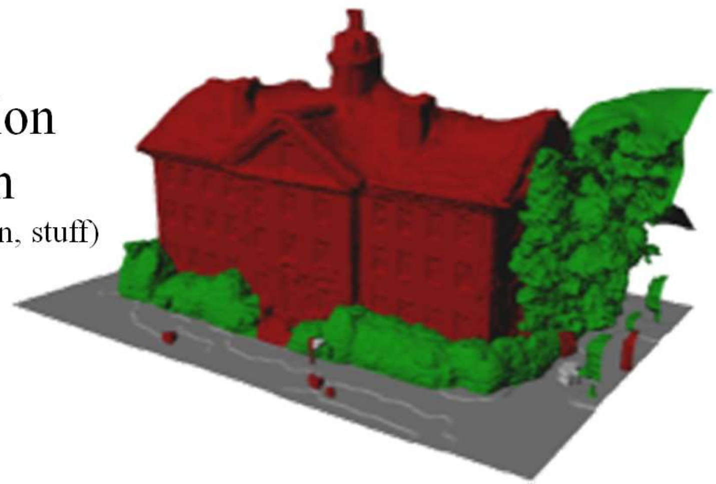
(Haene et al CVPR13)



reconstruction only
(uniform smoothness prior)



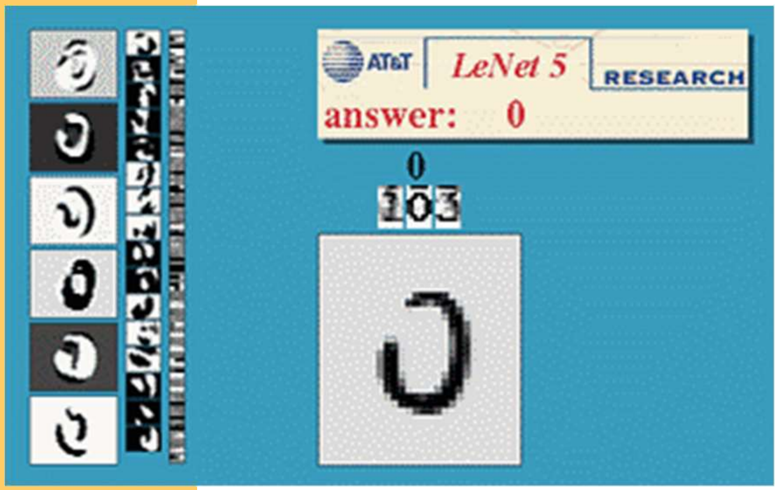
joint reconstruction
and segmentation
(ground, building, vegetation, stuff)



Optical character recognition (OCR)

Technology to convert scanned docs to text

- If you have a scanner, it probably came with OCR software



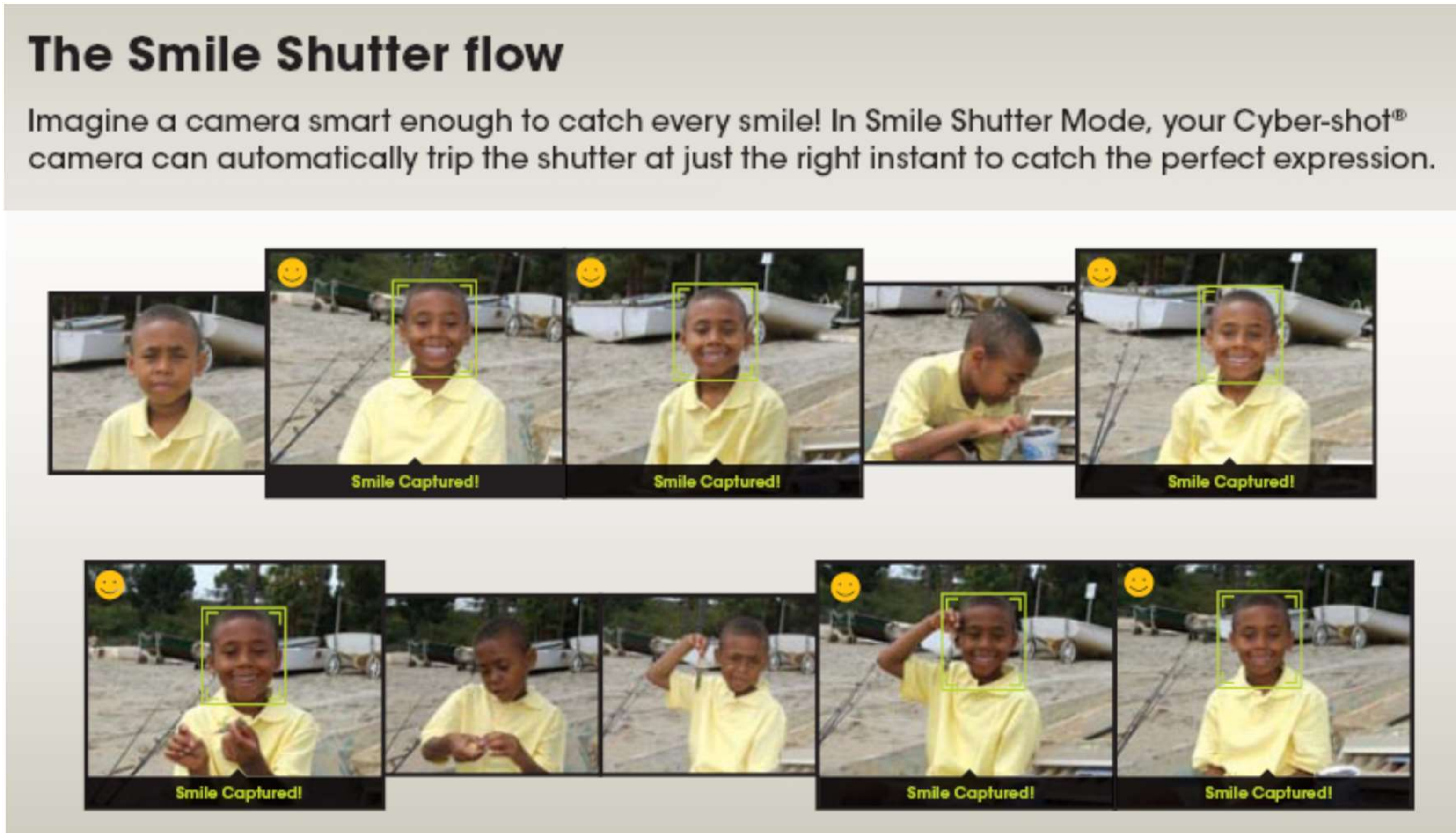
Digit recognition, AT&T labs
http://www.research.att.com/~yann/

License plate readers
http://en.wikipedia.org/wiki/Automatic_number_plate_recognition

Face detection

Many new digital cameras now detect faces
- Canon, Sony, Fuji, …

Smile detection?



[Sony Cyber-shot® T70 Digital Still Camera](#)

Face recognition

- E.g. Google photo

Category recognition

… still very hard problem for computers

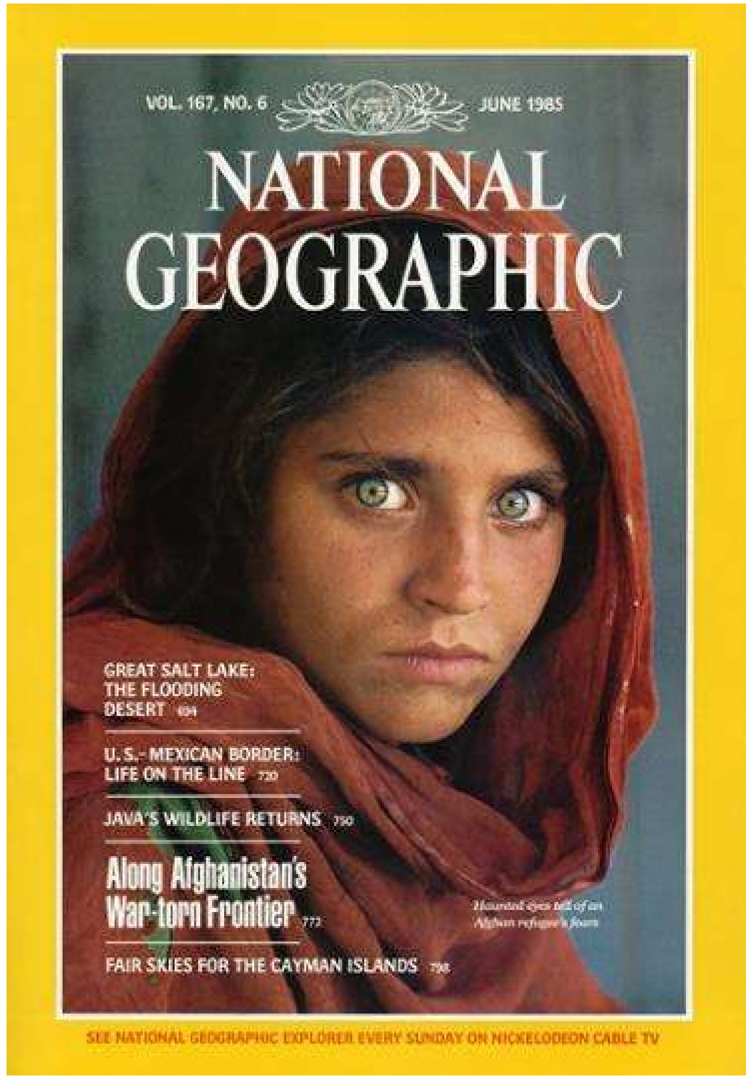
Object recognition (in supermarkets)

LaneHawk by EvolutionRobotics

"A smart camera is flush-mounted in the checkout lane, continuously watching for items. When an item is detected and recognized, the cashier verifies the quantity of items that were found under the basket, and continues to close the transaction. The item can remain under the basket, and with LaneHawk,you are assured to get paid for it… "

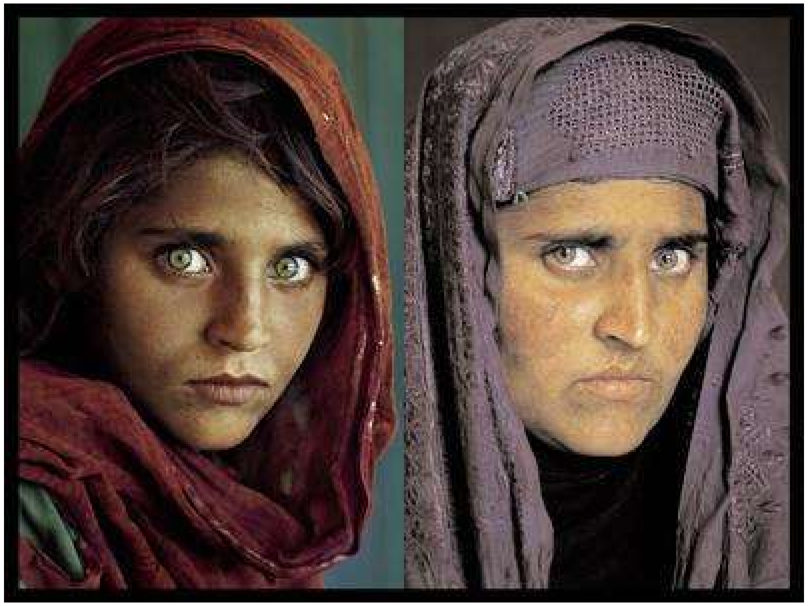
Amazon Go: https://www.youtube.com/watch?v=NrmMk1Myrxc

Face recognition

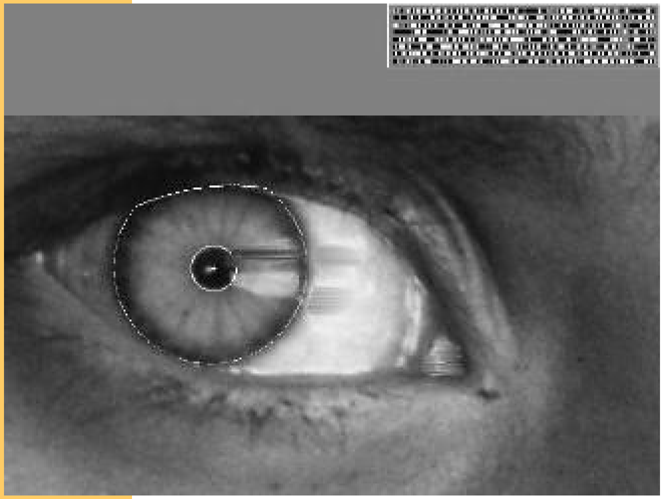


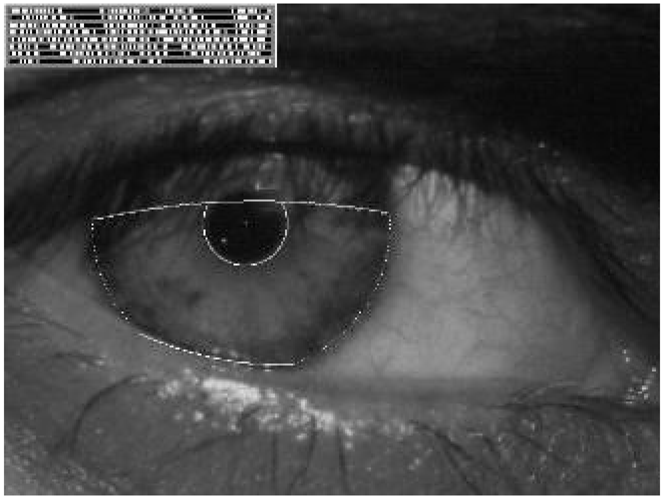
Who is she?

Vision-based biometrics



"*How the Afghan Girl was Identified by Her Iris Patterns*" Read the story





Login without a password…

Windows Hello

https://www.groovypost.com/unplugged/can-you-trick-windows-hello-with-a-photo/

OneVisage

start-up

Facial animation capture

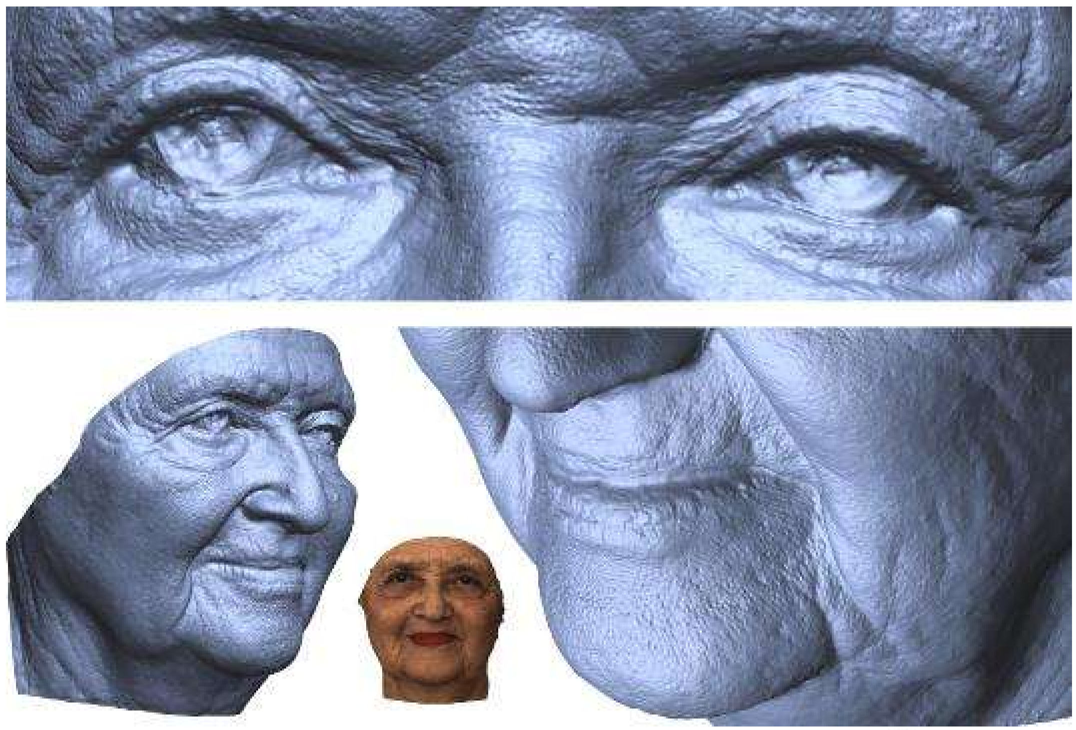
Object recognition (in mobile phones)

- This is becoming real:
  - Google Image search
  - [Point & Find](), Nokia
  - [Kooaba]()(ETH start-up, now Qualcomm
  - snaptell (now Amazon)

Special effects: shape capture

The Matrix movies, ESC Entertainment, XYZRGB, NRC

Special effects: shape capture



(Beeler et al. Siggraph 2010)

Special effects: motion capture

Pirates of the Carribean, Industrial Light and Magic

Sports

Sportvision first down line
Nice explanation on www.howstuffworks.com

LiberoVision / VizRT

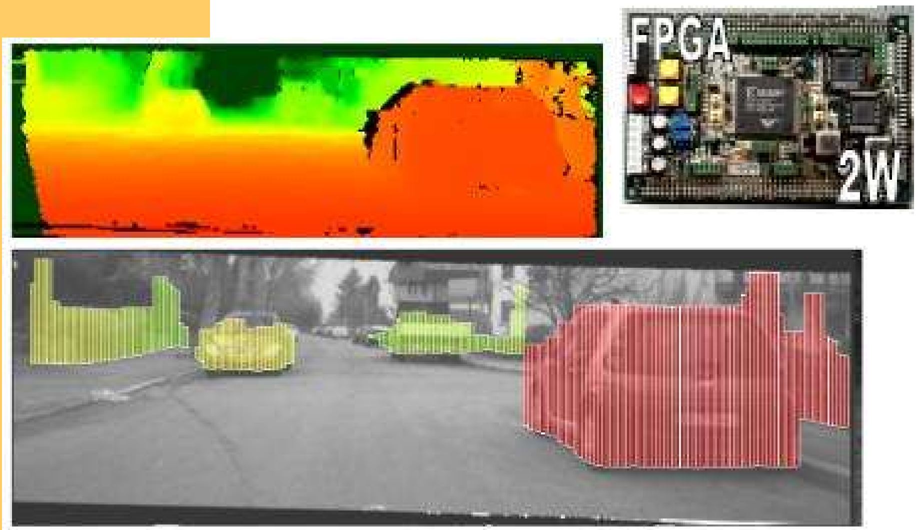
Smart cars

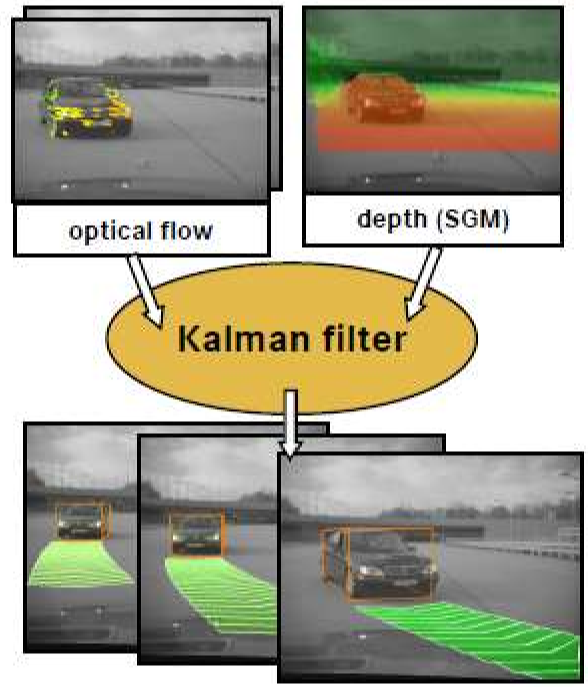
- [Mobileye]
  - Vision systems currently in high-end BMW, GM, Volvo models
  - By 2010: 70% of car manufacturers.
  - [Video demo]

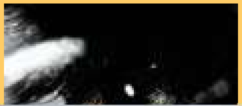
Slide content courtesy of Amnon Shashua

Driver Assistance

Daimler stereo system



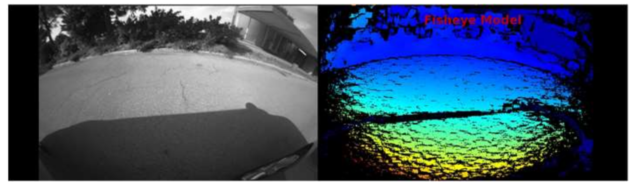




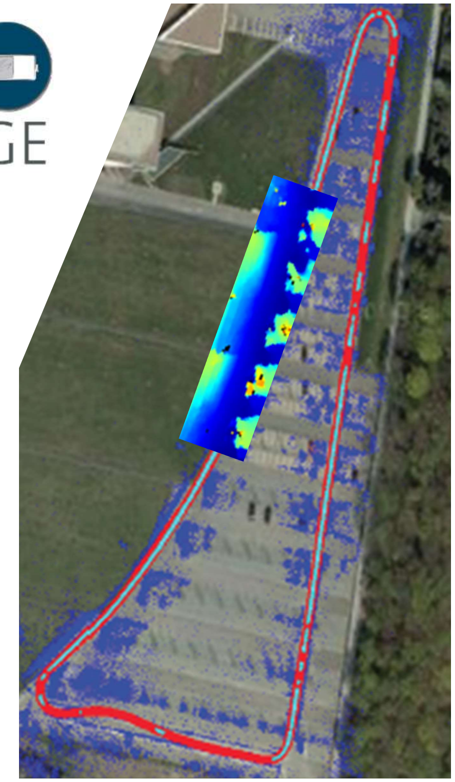

BOSCH

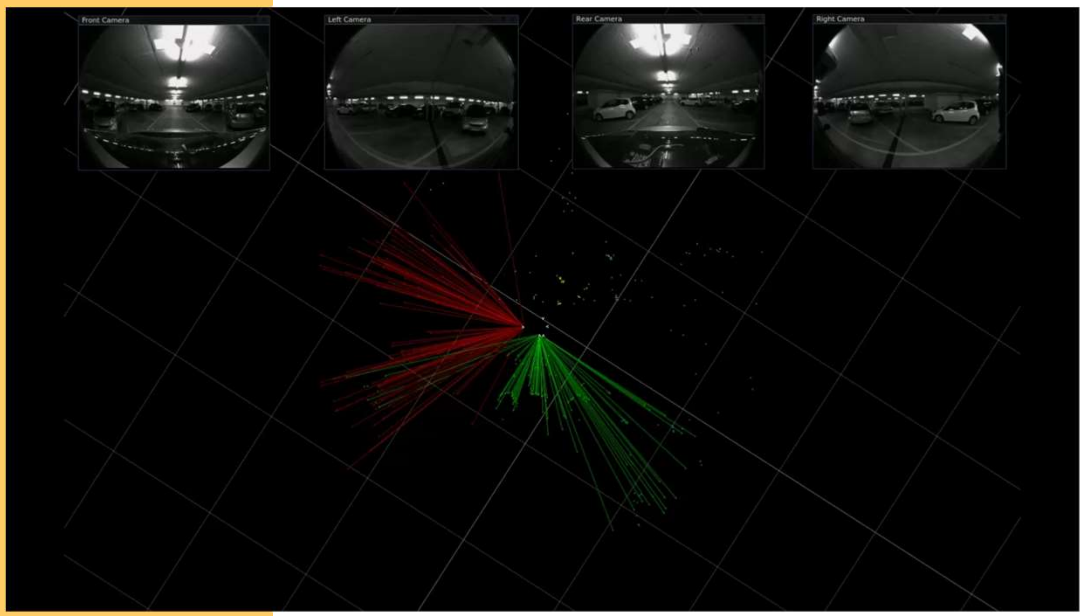
3D mapping for autonomous driving

(Lee et al CVPR13;
Heng et al. ICRA14;
Haene et al…)

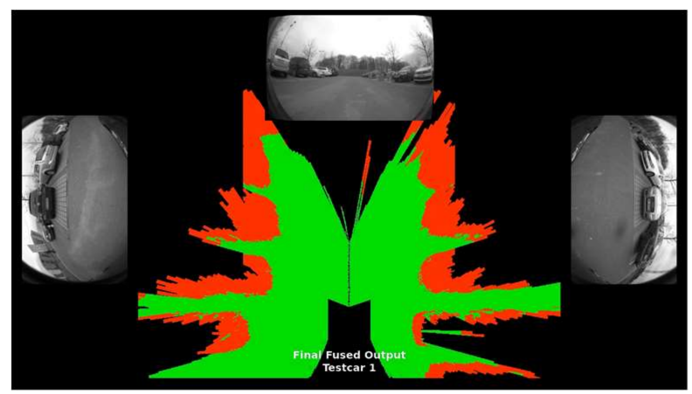


Dense real-time temporal fisheye stereo (GPU)



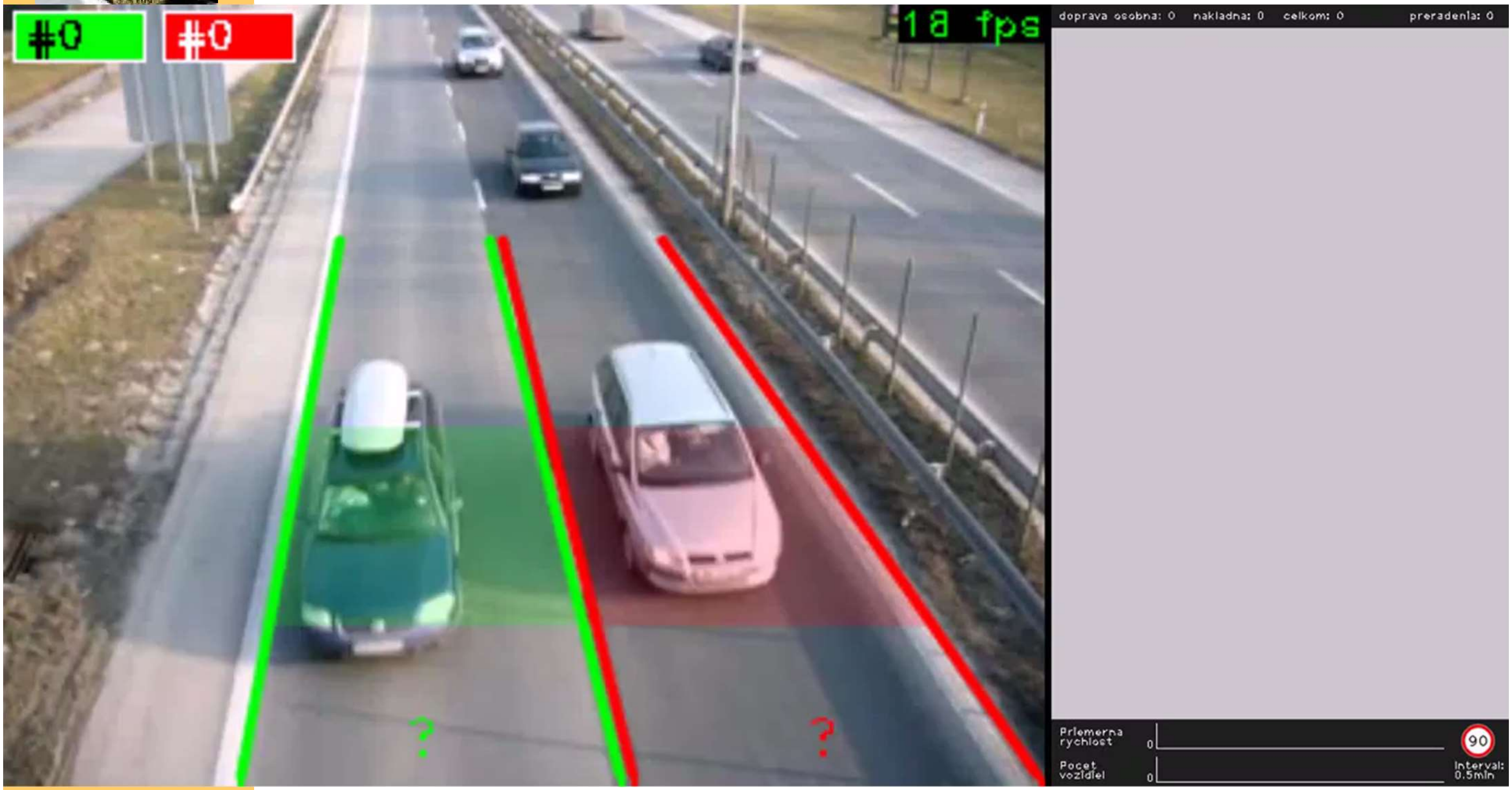


omnidirectional visual simultaneous localization and mapping



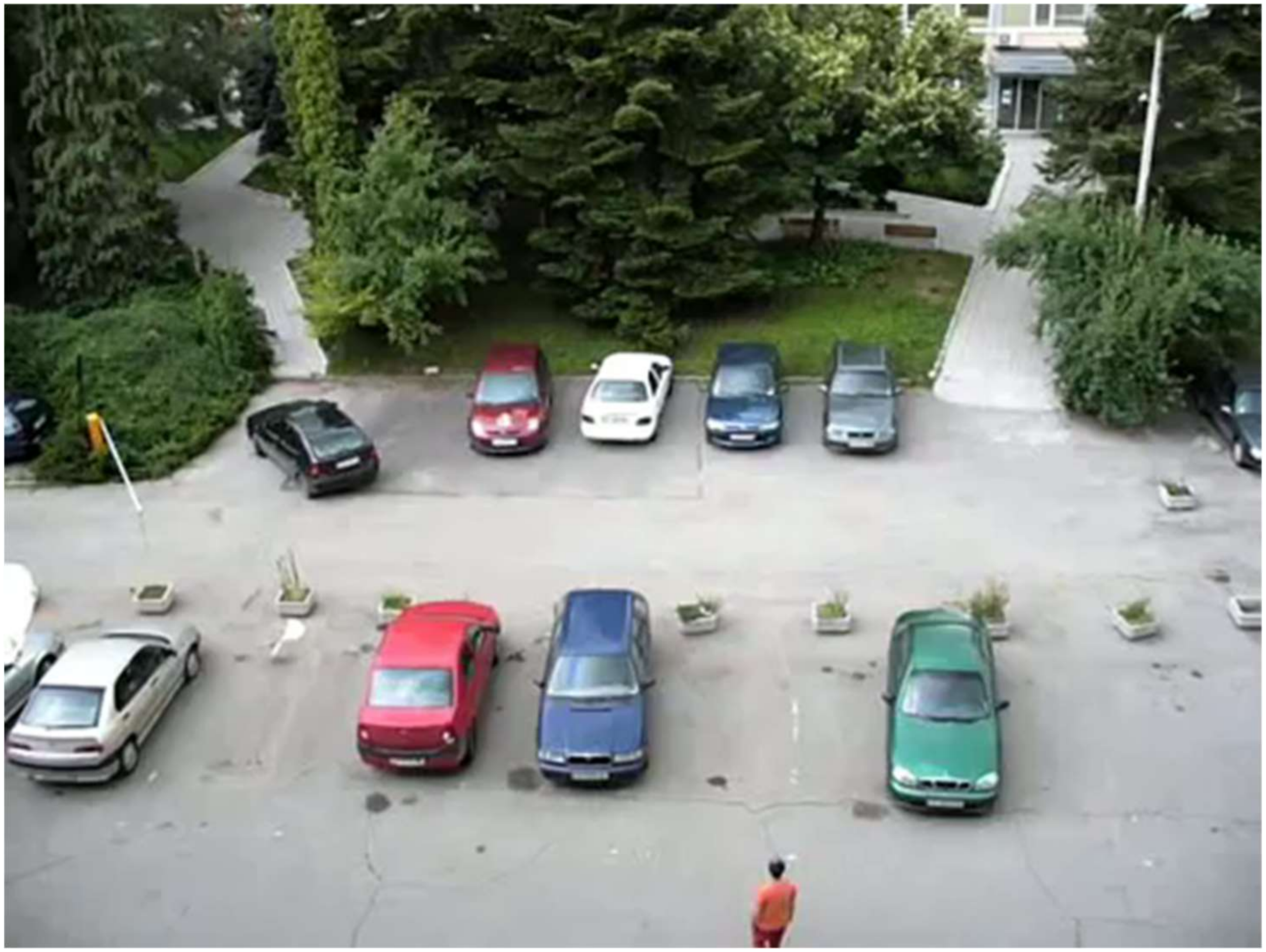
cameras as main sensors: *no lasers, no*

Traffic surveillance



https://www.youtube.com/watch?v=aV7sq81qFS4

Video surveillance
Tracking people



https://sites.google.com/site/surveillancesystembas/

Vision-based interaction (and games)

Digimask: put your face on a 3D avatar.

Nintendo Wii has camera-based IR tracking built in. See Lee's work at CMU on clever tricks on using it to create a multi-touch display!

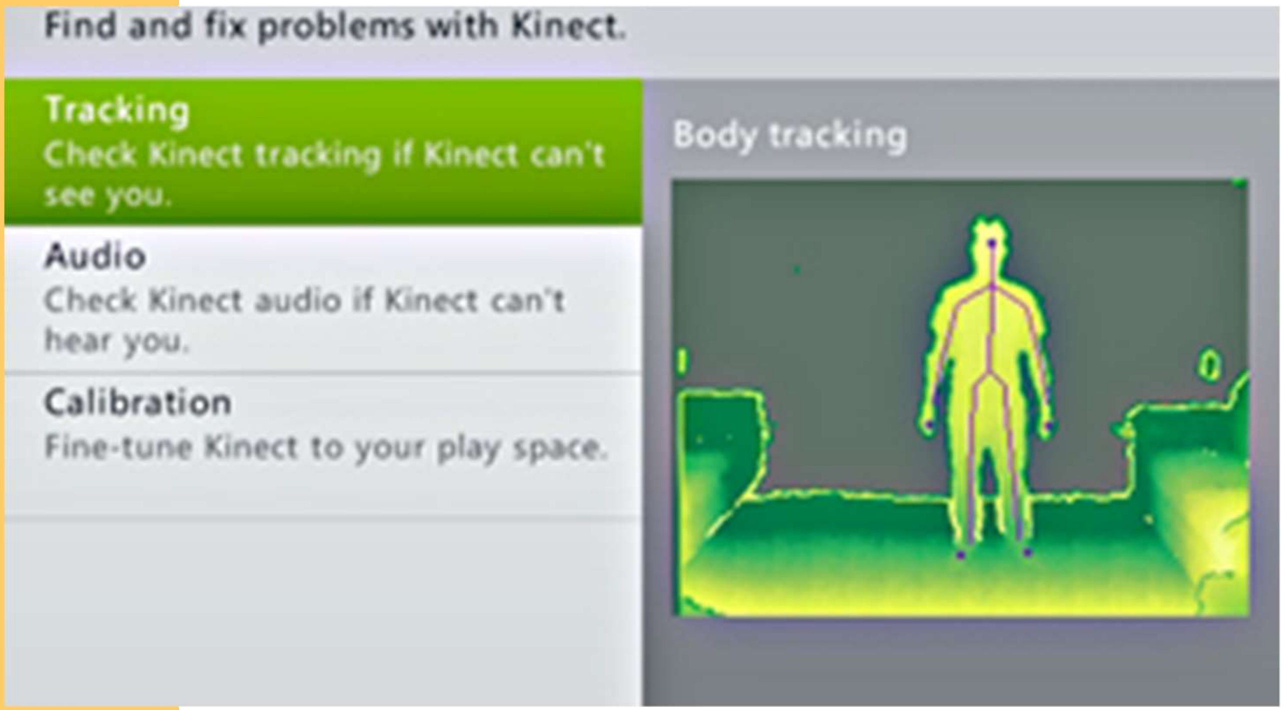
"Game turns moviegoers into Human Joysticks", CNET
Camera tracking a crowd, based on this work.

Vision-based interaction (and games)

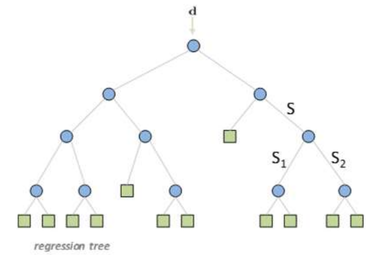
MS project natal/Kinect

http://www.xbox.com/en-US/kinect

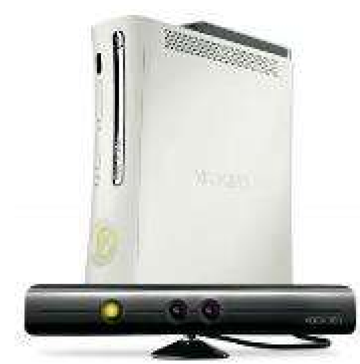
Bodytracking



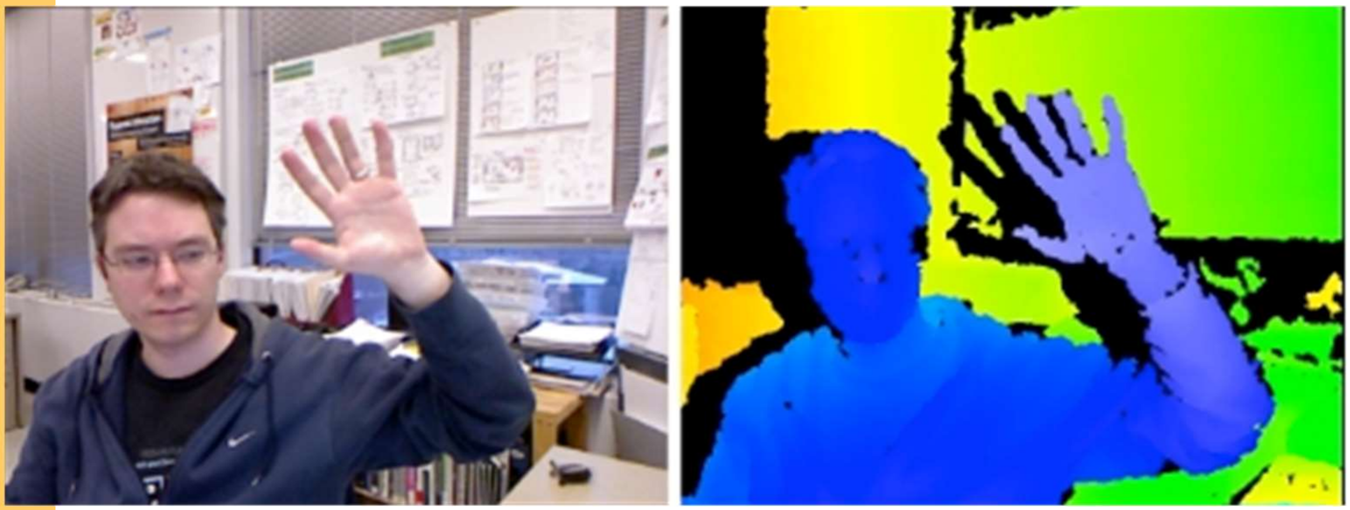
Randomized forest (Shotton et al.)



based on simple depth difference tests at each node

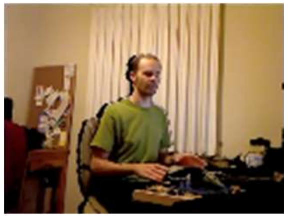


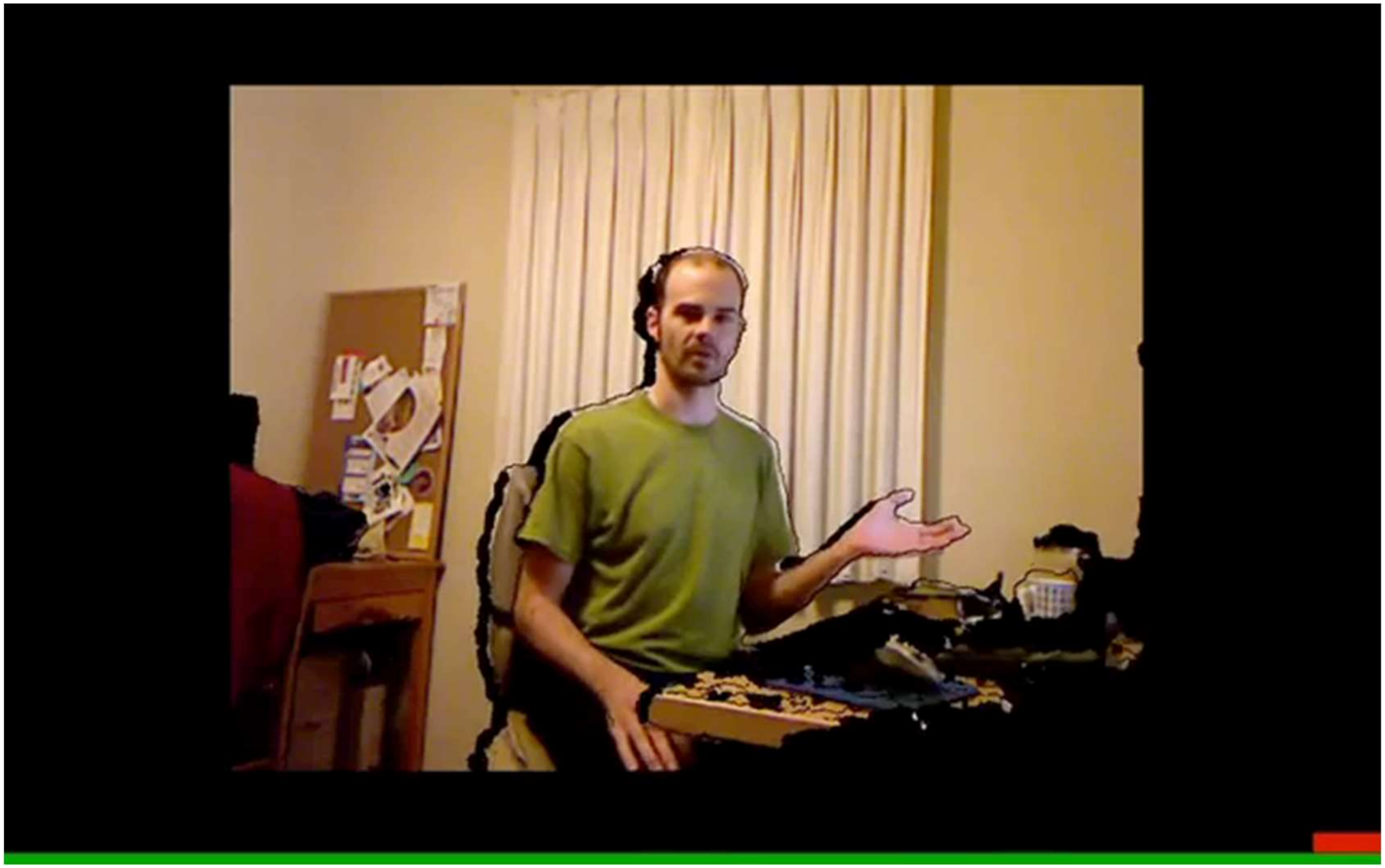
Raw Kinect output: Color + Depth = RGBD



http://grouplab.cpsc.ucalgary.ca/cookbook/index.php/Technologies/Kinect

3D Video with Kinect





SIGGRAPH Talks 2011

KinectFusion:

Real-Time Dynamic 3D Surface Reconstruction and Interaction

Shahram Izadi 1, Richard Newcombe 2, David Kim 1,3, Otmar Hilliges 1, David Molyneaux 1,4, Pushmeet Kohli 1, Jamie Shotton 1, Steve Hodges 1, Dustin Freeman 5, Andrew Davison 2, Andrew Fitzgibbon 1

1 Microsoft Research Cambridge 2 Imperial College London
3 Newcastle University 4 Lancaster University
5 University of Toronto

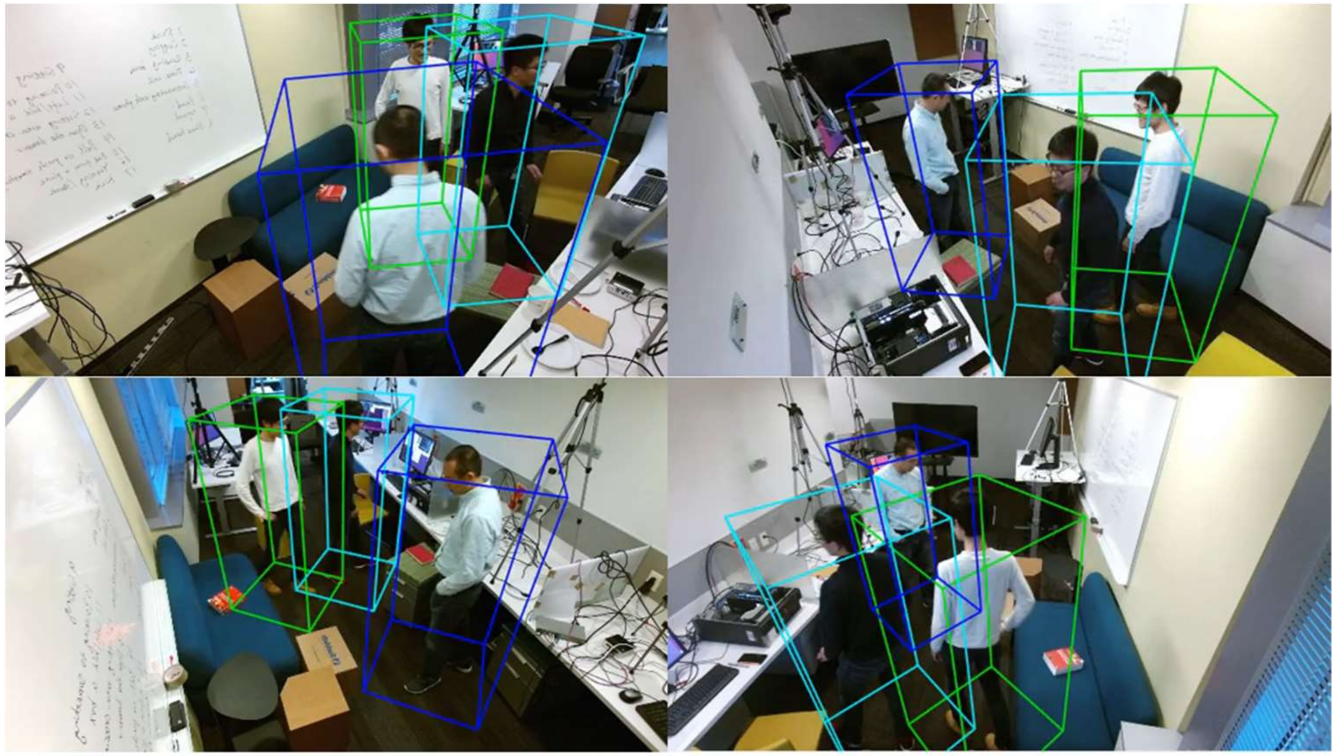
Next Generation Depth Camera

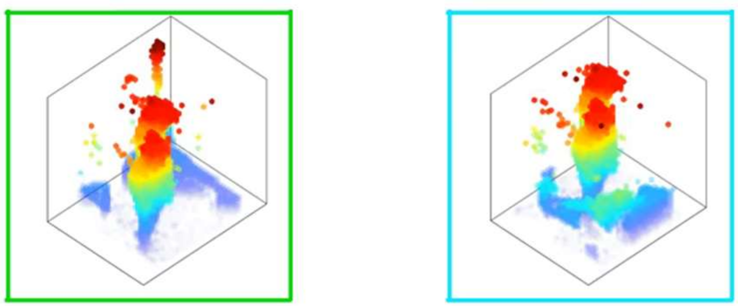
Microsoft depth-sensing

- HoloLens
- Project Kinect for Azure

https://aka.ms/iwantkinect/

1Mpix depth, consumes less than 1 Watt





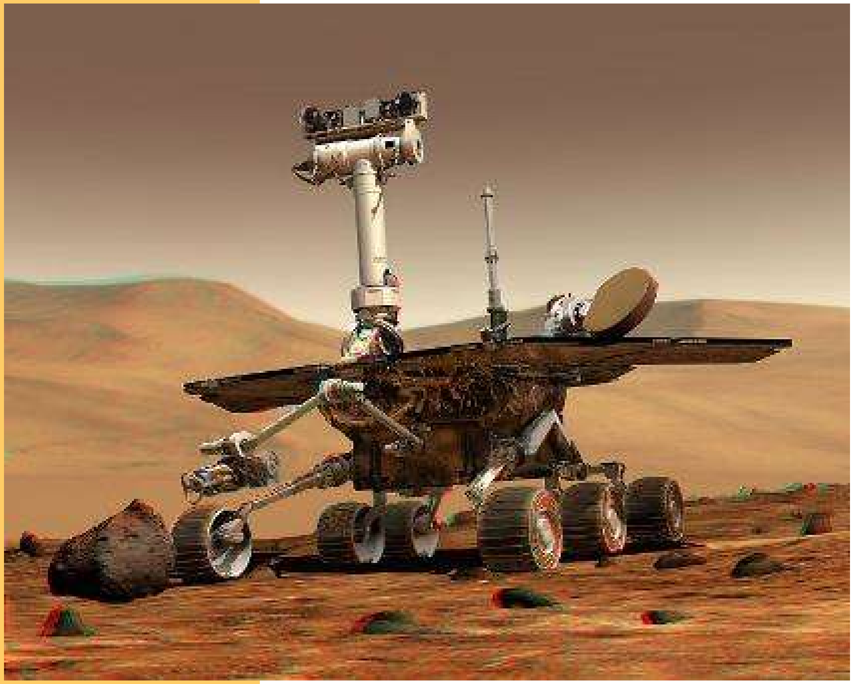
Vision in space

NASA'S Mars Exploration Rover Spirit captured this westward view from atop a low plateau where Spirit spent the closing months of 2007.

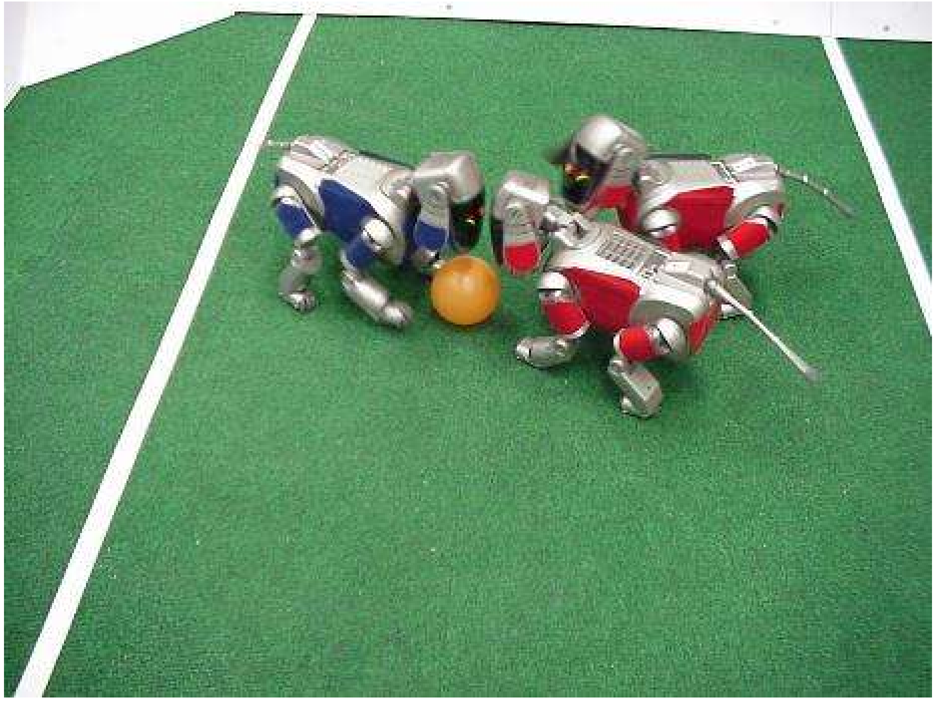
Vision systems (JPL) used for several tasks

- Panorama stitching
- 3D terrain modeling
- Obstacle detection, position tracking
- For more, read "Computer Vision on Mars" by Matthies et al.

Robotics



NASA's Mars Spirit Rover
http://en.wikipedia.org/wiki/Spirit_rover



http://www.robocup.org/

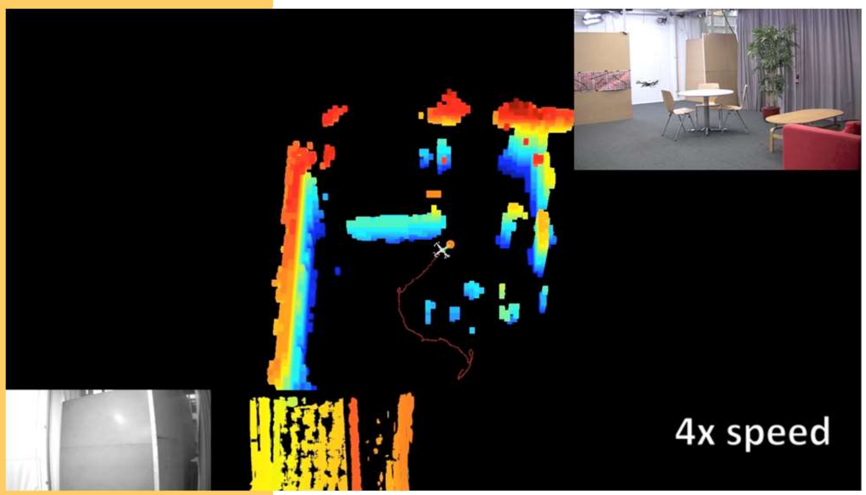
Full autonomous vision-based navigation and mapping

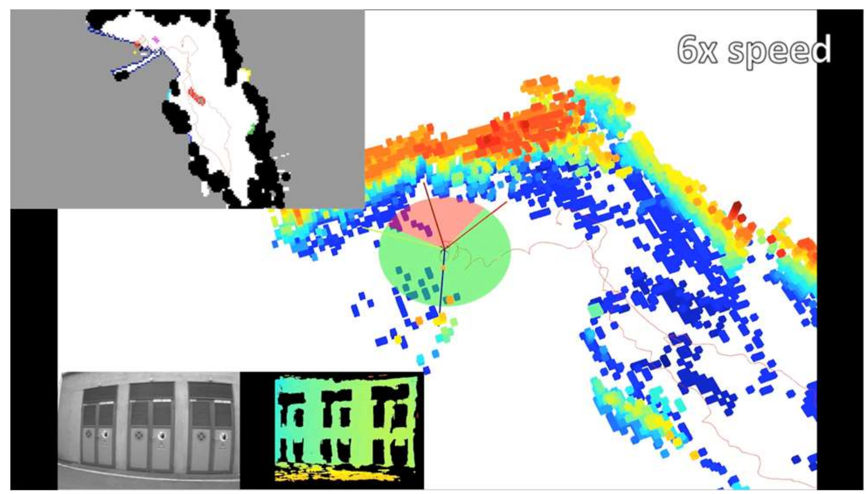
(Fraundorfer et al. IROS12 best paper finalist)

full on-board processing
2+1 cameras + IMU

indoor and outdoor operation
obstacle avoidance, mapping
and exploration

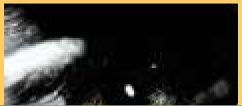
no laser, no GPS, no network





 More on PixHawk: http://pixhawk.ethz.ch

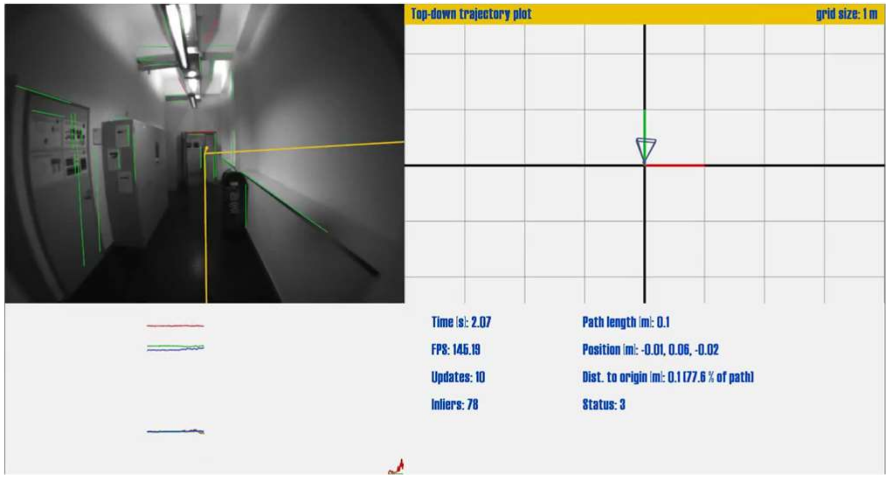
sFly



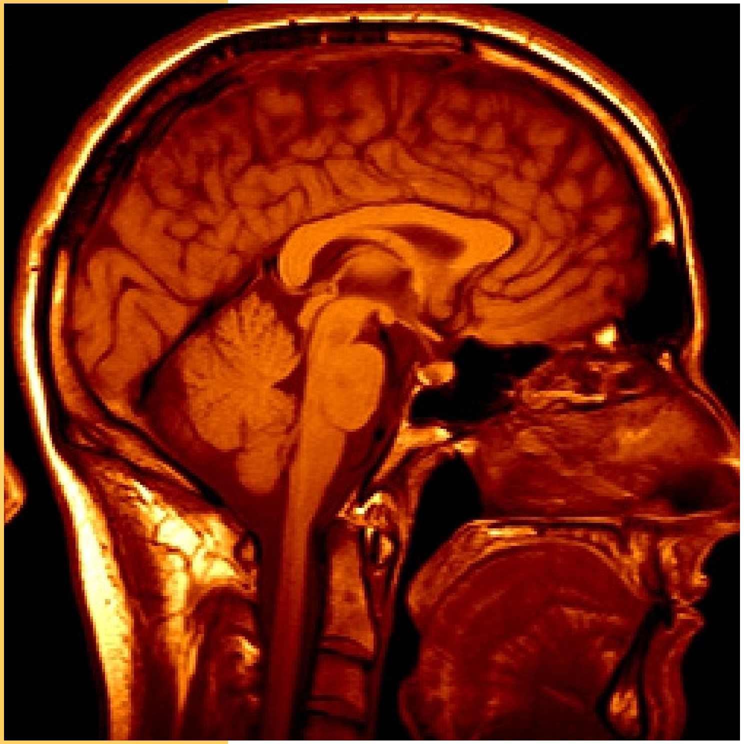

Project Tango

Research partner of Google's Project Tango

Project Tango



Medical imaging



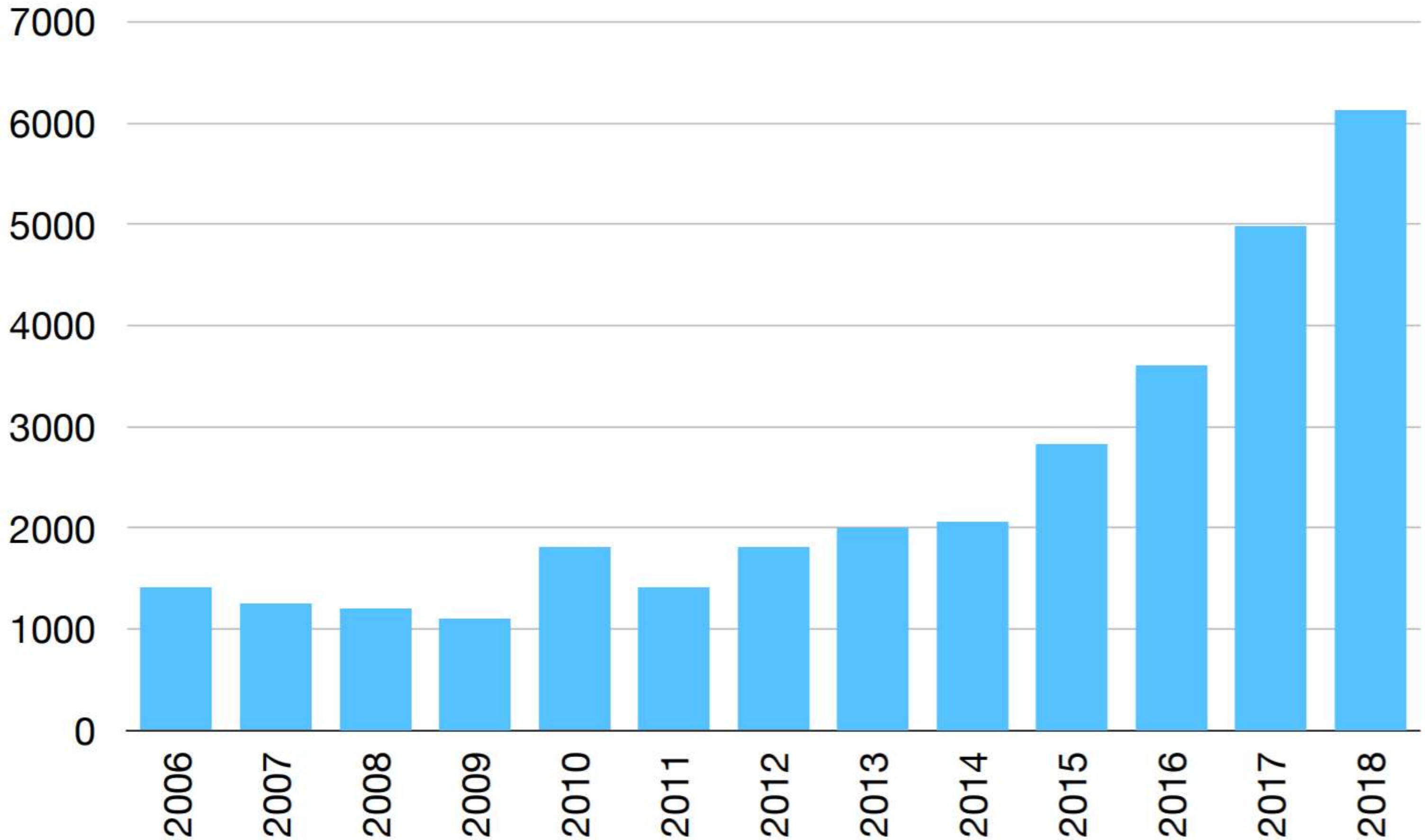
3D imaging
MRI, CT

Image guided surgery
Grimson et al., MIT

Current state of the art

- You just saw examples of current systems.
  - Many of these are less than 5 years old
- This is a very active research area, and rapidly changing
  - Many new apps in the next 5 years
- To learn more about vision applications and companies
  - David Lowe maintains an excellent overview of vision companies
    - http://www.cs.ubc.ca/spider/lowe/vision.html

CVPR Attendance



Schedule (tentative)

#	date	topic
1	Sep.19	Introduction and geometry
2	Sep.26	Camera models and calibration
3	Oct.3	Invariant features
4	Oct.10	Optical flow & Particle Filters
5	Oct.17	Model fitting (RANSAC, EM, …)
6	Oct.24	Multiple-view geometry
7	Oct.31	Image segmentation
8	Nov.7	Stereo Matching & MVS
9	Nov.14	Structure-from-Motion & SLAM
10	Nov.21	Specific object recognition
11	Nov.28	Shape from X
12	Dec.5	Object category recognition
13	Dec.12	Tracking
14	Dec.19	Research Overview & Lab tours

Projective Geometry

points, lines, planes
conics and quadrics
transformations
camera model

Read tutorial chapter 2 and 3.1
http://www.cs.unc.edu/~marc/tutorial/

Szeliski's book pp.29-41

Homogeneous coordinates

Homogeneous representation of 2D points and lines

$$ax + by + c = 0 \qquad\qquad (a,b,c)^{\mathsf{T}}(x,y,1) = 0$$

The point $\mathbf{x}$ lies on the line $\mathbf{l}$ if and only if

$$\mathbf{l}^{\mathsf{T}}\mathbf{x} = 0$$

Note that scale is unimportant for incidence relation

$$(a,b,c)^{\mathsf{T}} \sim k(a,b,c)^{\mathsf{T}}, \forall k \neq 0 \qquad (x,y,1)^{\mathsf{T}} \sim k(x,y,1)^{\mathsf{T}}, \forall k \neq 0$$

equivalence class of vectors, any vector is representative

Set of all equivalence classes in $\mathbf{R}^3$–$(0,0,0)^{\mathsf{T}}$ forms $\mathbf{P}^2$

Homogeneous coordinates $(x_1, x_2, x_3)^{\mathsf{T}}$ but only 2DOF

Inhomogeneous coordinates $(x, y)^{\mathsf{T}}$

Points from lines and vice-versa

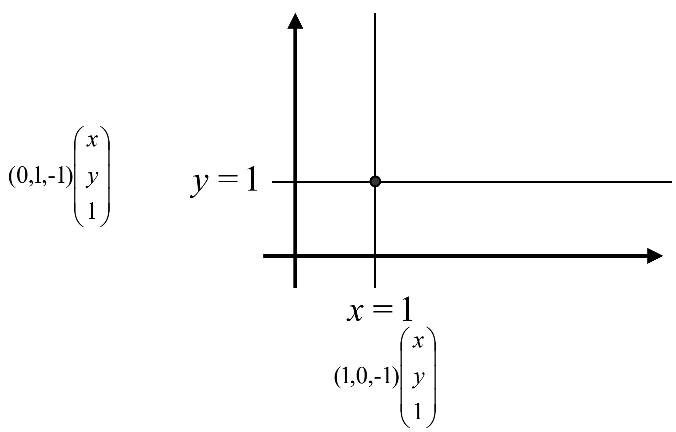
Intersections of lines

The intersection of two lines l and l' is $\mathrm{x} = \mathrm{l} \times \mathrm{l}'$

Line joining two points

The line through two points x and x' is $\mathrm{l} = \mathrm{x} \times \mathrm{x}'$

Example



Note:

$$\mathrm{x} \times \mathrm{x}' = [\mathrm{x}]_\times \mathrm{x}'$$

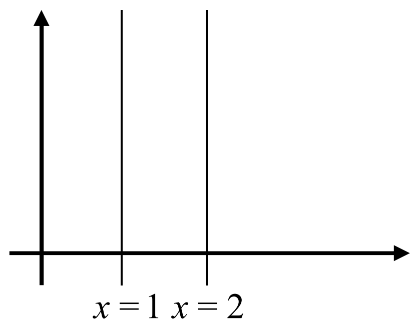
with $[\mathrm{x}]_\times = \begin{bmatrix} 0 & z & -y \\ -z & 0 & x \\ y & -x & 0 \end{bmatrix}$

Ideal points and the line at infinity

Intersections of parallel lines

$$\mathbf{l} = (a, b, c)^{\mathsf{T}} \text{ and } \mathbf{l}' = (a, b, c')^{\mathsf{T}} \qquad \mathbf{l} \times \mathbf{l}' = (b, -a, 0)^{\mathsf{T}}$$

Example



$(b, -a)$ tangent vector

(a, b) normal direction

Ideal points $(x_1, x_2, 0)^{\mathsf{T}}$

Line at infinity $\mathbf{l}_\infty = (0, 0, 1)^{\mathsf{T}}$

$$\mathbf{P}^2 = \mathbf{R}^2 \cup \mathbf{l}_\infty$$

Note that in **P**^{2} there is no distinction between ideal points and others

3D points and planes

Homogeneous representation of 3D points and planes

$$\pi_1 X_1 + \pi_2 X_2 + \pi_3 X_3 + \pi_4 X_4 = 0$$

The point X lies on the plane π if and only if

$$\pi^{\mathsf{T}}\mathrm{X} = 0$$

The plane π goes through the point X if and only if

$$\pi^{\mathsf{T}}\mathrm{X} = 0$$

Planes from points

Solve π from $\mathrm{X}_1^{\mathsf{T}}\pi = 0$, $\mathrm{X}_2^{\mathsf{T}}\pi = 0$ and $\mathrm{X}_3^{\mathsf{T}}\pi = 0$

$$\begin{bmatrix} \mathrm{X}_1^{\mathsf{T}} \\ \mathrm{X}_2^{\mathsf{T}} \\ \mathrm{X}_3^{\mathsf{T}} \end{bmatrix} \pi = 0$$

(solve π as right nullspace of $\begin{bmatrix} \mathrm{X}_1^{\mathsf{T}} \\ \mathrm{X}_2^{\mathsf{T}} \\ \mathrm{X}_3^{\mathsf{T}} \end{bmatrix}$)

Points from planes

Solve X from $\pi_1^{\mathsf{T}}\mathrm{X}=0$, $\pi_2^{\mathsf{T}}\mathrm{X}=0$ and $\pi_3^{\mathsf{T}}\mathrm{X}=0$

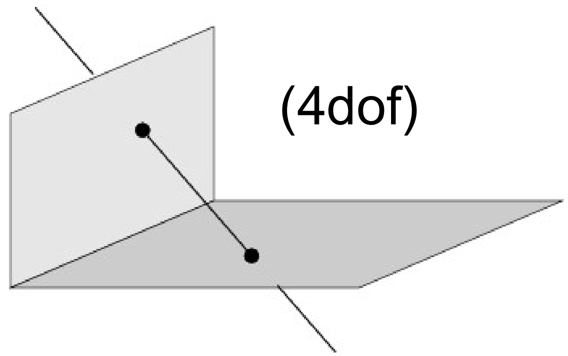
$$\begin{bmatrix}\pi_1^{\mathsf{T}}\\ \pi_2^{\mathsf{T}}\\ \pi_3^{\mathsf{T}}\end{bmatrix}\mathrm{X}=0 \quad \text{(solve } \mathrm{X} \text{ as right nullspace of } \begin{bmatrix}\pi_1^{\mathsf{T}}\\ \pi_2^{\mathsf{T}}\\ \pi_3^{\mathsf{T}}\end{bmatrix}\text{)}$$

Representing a plane by its span

$$\mathrm{X}=\mathbf{M}\,\mathrm{x} \qquad \mathbf{M}=\left[\mathrm{X}_1\mathrm{X}_2\mathrm{X}_3\right]$$

$$\pi^{\mathsf{T}}\mathbf{M}=0$$

Lines



(4dof)

Representing a line by its span

$$W = \begin{bmatrix} A^\top \\ B^\top \end{bmatrix} \qquad \lambda A + \mu B$$

Dual representation

$$W^* = \begin{bmatrix} P^\top \\ Q^\top \end{bmatrix} \qquad \lambda P + \mu Q$$

$$W^* W^\top = W W^{*\top} = 0_{2\times 2}$$

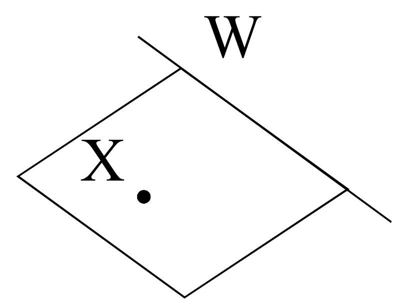
Example: X-axis

$$W = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \qquad W^* = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

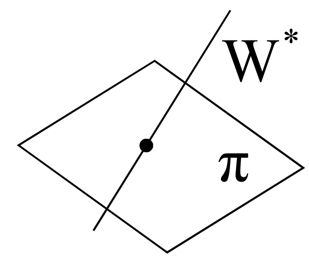
(Alternative: Plücker representation, details see e.g. H&Z)

Points, lines and planes

$$\mathbf{M} = \begin{bmatrix} W \\ X^{\mathsf{T}} \end{bmatrix} \qquad \mathbf{M}\pi = 0$$

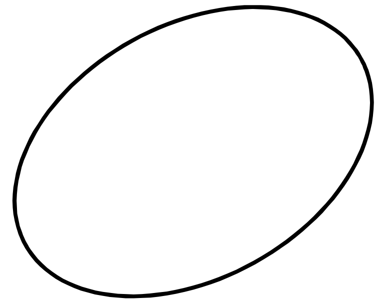


$$\mathbf{M} = \begin{bmatrix} W^{*} \\ \pi^{\mathsf{T}} \end{bmatrix} \qquad \mathbf{M}X = 0$$



Conics

Curve described by 2nd-degree equation in the plane



$$ax^2 + bxy + cy^2 + dx + ey + f = 0$$

or *homogenized* $x \mapsto x_1/x_3,\ y \mapsto x_2/x_3$

$$a{x_1}^2 + bx_1x_2 + c{x_2}^2 + dx_1x_3 + ex_2x_3 + fx_3^2 = 0$$

or in matrix form

$$\mathrm{x}^\mathsf{T}\mathbf{C}\,\mathrm{x} = 0 \quad \text{with} \quad \mathbf{C} = \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix}$$

5DOF: $\{a:b:c:d:e:f\}$

Five points define a conic

For each point the conic passes through

$$ax_i^2 + bx_iy_i + cy_i^2 + dx_i + ey_i + f = 0$$

or

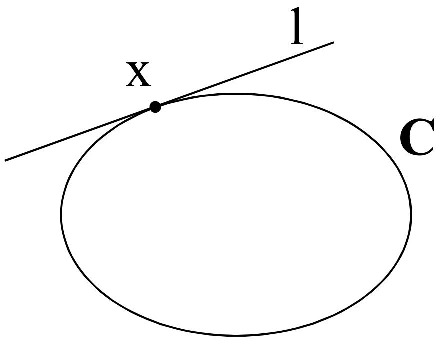
$$(x_i^2, x_iy_i, y_i^2, x_i, y_i, 1)\mathbf{c} = 0 \qquad \mathbf{c} = (a, b, c, d, e, f)^{\mathsf{T}}$$

stacking constraints yields

$$\begin{bmatrix} x_1^2 & x_1y_1 & y_1^2 & x_1 & y_1 & 1 \\ x_2^2 & x_2y_2 & y_2^2 & x_2 & y_2 & 1 \\ x_3^2 & x_3y_3 & y_3^2 & x_3 & y_3 & 1 \\ x_4^2 & x_4y_4 & y_4^2 & x_4 & y_4 & 1 \\ x_5^2 & x_5y_5 & y_5^2 & x_5 & y_5 & 1 \end{bmatrix}\mathbf{c} = 0$$

Tangent lines to conics

The line l tangent to $\mathbf{C}$ at point x on $\mathbf{C}$ is given by $\mathrm{l}=\mathbf{C}\mathrm{x}$

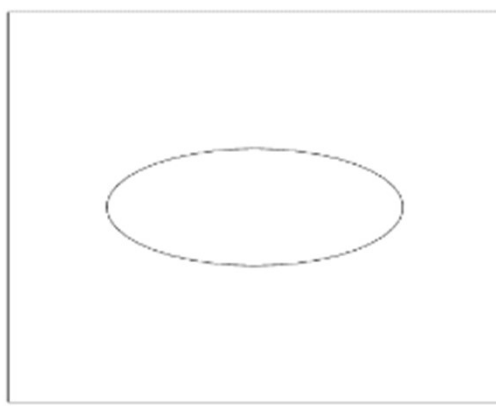


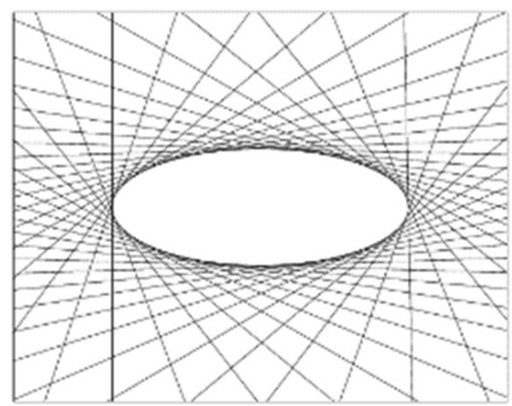
Dual conics

A line tangent to the conic $\mathbf{C}$ satisfies $\mathbf{l}^{\mathsf{T}}\,\mathbf{C}^{*}\,\mathbf{l}=0$

In general ($\mathbf{C}$ full rank): $\mathbf{C}^{*}=\mathbf{C}^{-1}$

Dual conics = line conics = conic envelopes

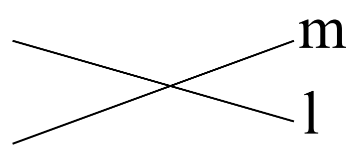




Degenerate conics

A conic is degenerate if matrix $\mathbf{C}$ is not of full rank

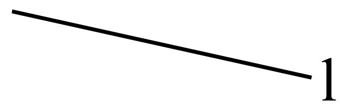
e.g. two lines (rank 2)



$$\mathbf{C} = \mathrm{l}\mathrm{m}^{\mathsf{T}} + \mathrm{m}\mathrm{l}^{\mathsf{T}}$$

e.g. repeated line (rank 1)

$$\mathbf{C} = \mathrm{l}\mathrm{l}^{\mathsf{T}}$$



Degenerate line conics: 2 points (rank 2), double point (rank1)

Note that for degenerate conics $\left(\mathbf{C}^{*}\right)^{*} \neq \mathbf{C}$

Quadrics and dual quadrics

$X^{T}QX = 0$ (Q : 4x4 symmetric matrix)

$$Q = \begin{bmatrix} \bullet & \bullet & \bullet & \bullet \\ \circ & \bullet & \bullet & \bullet \\ \circ & \circ & \bullet & \bullet \\ \circ & \circ & \circ & \bullet \end{bmatrix}$$

- 9 d.o.f.
- in general 9 points define quadric
- det Q=0 $\leftrightarrow$ degenerate quadric
- tangent plane $\pi = QX$

$\pi^{T}Q^{*}\pi = 0$

- relation to quadric $Q^{*} = Q^{-1}$ (non-degenerate)

2D projective transformations

Definition:

A *projectivity* is an invertible mapping h from P^2 to itself such that three points x_1,x_2,x_3 lie on the same line if and only if $h(x_1)$,$h(x_2)$,$h(x_3)$ do.

Theorem:

A mapping $h:P^2 \rightarrow P^2$ is a projectivity if and only if there exist a non-singular 3x3 matrix $\mathbf{H}$ such that for any point in P^2 reprented by a vector x it is true that $h(x)=\mathbf{H}x$

Definition: Projective transformation

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

or $x' = \mathbf{H}\, x$

8DOF

projectivity=collineation=projective transformation=homography

Transformation of 2D points, lines and conics

For a point transformation

$$\mathrm{x}' = \mathbf{H}\,\mathrm{x}$$

Transformation for lines

$$\mathrm{l}' = \mathbf{H}^{-\mathsf{T}}\,\mathrm{l}$$

Transformation for conics

$$\mathbf{C}' = \mathbf{H}^{-\mathsf{T}}\mathbf{C}\mathbf{H}^{-1}$$

Transformation for dual conics

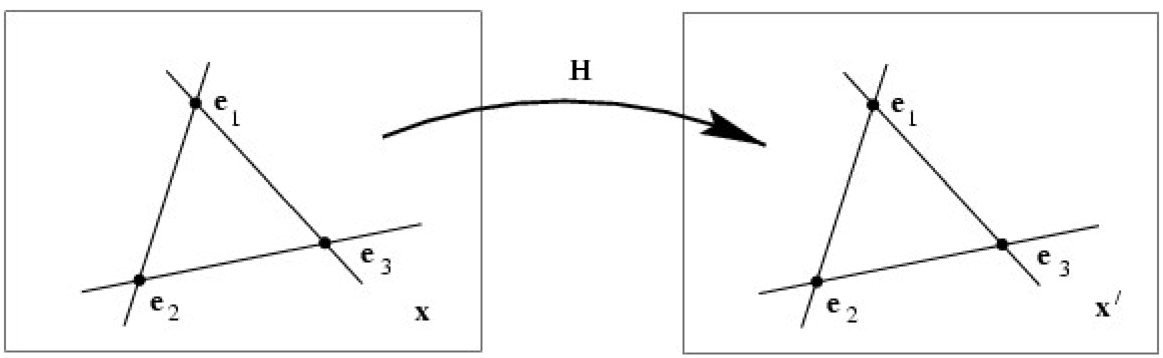
$$\mathbf{C}'^{*} = \mathbf{H}\mathbf{C}^{*}\mathbf{H}^{\mathsf{T}}$$

Fixed points and lines

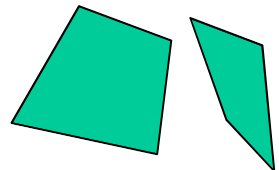
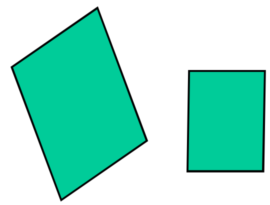
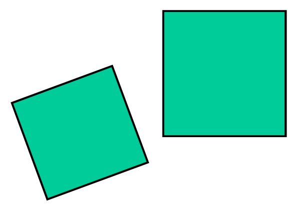
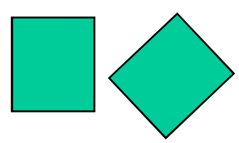
$$\mathbf{H}\,\mathrm{e} = \lambda\,\mathrm{e}$$ (eigenvectors $\mathbf{H}$ =fixed points)

(λ_1=λ_2 $\Rightarrow$ pointwise fixed line)

$$\mathbf{H}^{-\mathrm{T}}\,\mathrm{l} = \lambda\,\mathrm{l}$$ (eigenvectors $\mathbf{H}^{-\mathrm{T}}$ =fixed lines)



Hierarchy of 2D transformations

		transformed squares	invariants
Projective 8dof	$\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$		Concurrency, collinearity, order of contact (intersection, tangency, inflection, etc.), cross ratio
Affine 6dof	$\begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$		Parallellism, ratio of areas, ratio of lengths on parallel lines (e.g midpoints), linear combinations of vectors (centroids). **The line at infinity l_∞**
Similarity 4dof	$\begin{bmatrix} sr_{11} & sr_{12} & t_x \\ sr_{21} & sr_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$		Ratios of lengths, angles. **The circular points I,J**
Euclidean 3dof	$\begin{bmatrix} r_{11} & r_{12} & t_x \\ r_{21} & r_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$		lengths, areas.

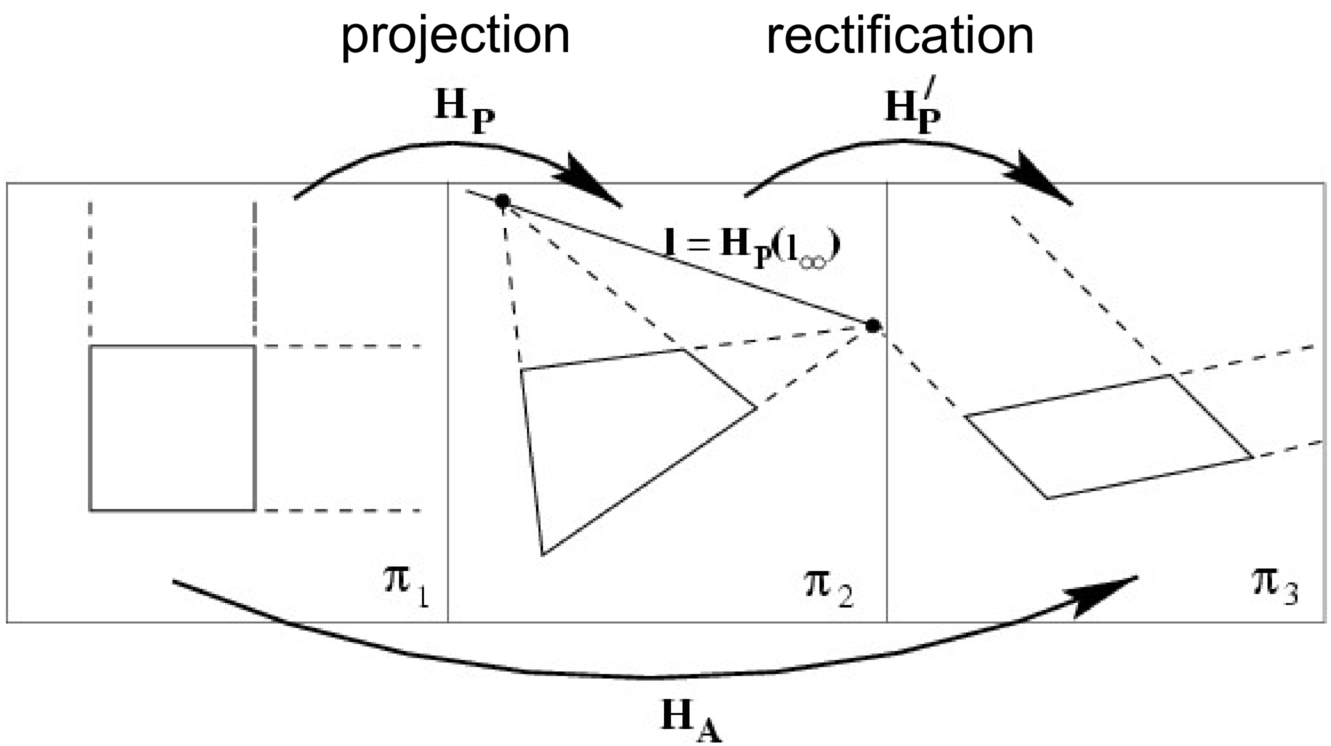
The line at infinity

$$\mathbf{l}'_\infty = \mathbf{H}_A^{-\mathsf{T}}\,\mathbf{l}_\infty = \begin{bmatrix} \mathbf{A}^{-\mathsf{T}} & 0 \\ -\mathbf{t}^{\mathsf{T}}\mathbf{A}^{-\mathsf{T}} & 1 \end{bmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \mathbf{l}_\infty$$

The line at infinity l_∞ is a fixed line under a projective transformation H if and only if H is an affinity

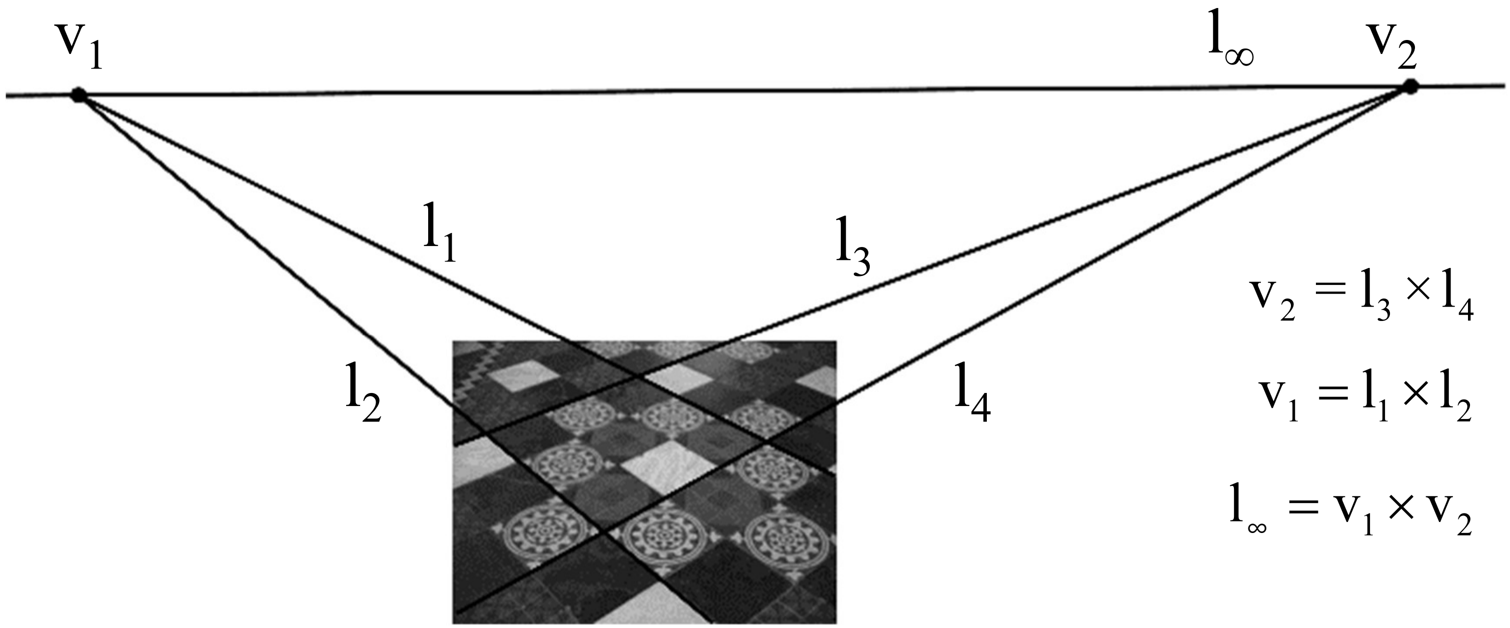
Note: not fixed pointwise

Affine properties from images



$$\mathbf{H}_{PA} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_1 & l_2 & l_3 \end{bmatrix} \mathbf{H}_A \qquad l_\infty = [l_1 \quad l_2 \quad l_3]^T, l_3 \neq 0$$

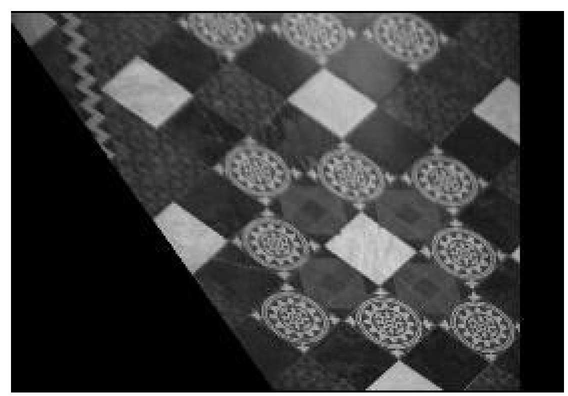
Affine rectification



$$v_2 = l_3 \times l_4$$

$$v_1 = l_1 \times l_2$$

$$l_\infty = v_1 \times v_2$$



The circular points

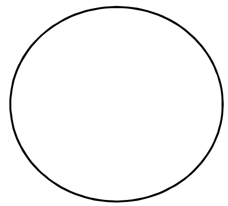
$$\mathrm{I} = \begin{pmatrix} 1 \\ i \\ 0 \end{pmatrix} \qquad \mathrm{J} = \begin{pmatrix} 1 \\ -i \\ 0 \end{pmatrix}$$

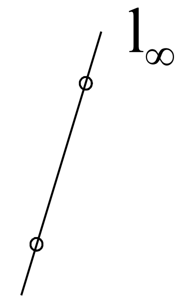
$$\mathrm{I}' = \mathbf{H}_S\,\mathrm{I} = \begin{bmatrix} s\cos\theta & s\sin\theta & t_x \\ -s\sin\theta & s\cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} 1 \\ i \\ 0 \end{pmatrix} = se^{i\theta} \begin{pmatrix} 1 \\ i \\ 0 \end{pmatrix} = \mathrm{I}$$

The circular points I, J are fixed points under the projective transformation $\mathbf{H}$ iff $\mathbf{H}$ is a similarity

The circular points

"circular points"





$$x_1^2 + x_2^2 + dx_1x_3 + ex_2x_3 + fx_3^2 = 0$$
$$x_3 = 0$$

$$x_1^2 + x_2^2 = 0$$

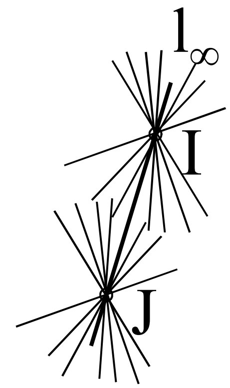
$$\mathrm{I} = (1, i, 0)^\mathsf{T}$$
$$\mathrm{J} = (1, \text{-}i, 0)^\mathsf{T}$$

Algebraically, encodes orthogonal directions

$$\mathrm{I} = (1,0,0)^\mathsf{T} + i(0,1,0)^\mathsf{T}$$

Conic dual to the circular points

$$\mathbf{C}^*_\infty = \mathbf{I}\mathbf{J}^\mathsf{T} + \mathbf{J}\mathbf{I}^\mathsf{T} \qquad \mathbf{C}^*_\infty = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$



$$\mathbf{C}^*_\infty = \mathbf{H}_S \mathbf{C}^*_\infty \mathbf{H}_S^\mathsf{T}$$

The dual conic $\mathbf{C}^*_\infty$ is fixed conic under the projective transformation $\mathbf{H}$ iff $\mathbf{H}$ is a similarity

Note: $\mathbf{C}^*_\infty$ has 4DOF

l_∞ is the nullvector

Angles

Euclidean: $\mathrm{l} = (l_1, l_2, l_3)^\mathsf{T} \qquad \mathrm{m} = (m_1, m_2, m_3)^\mathsf{T}$

$$\cos\theta = \frac{l_1 m_1 + l_2 m_2}{\sqrt{(l_1^2 + l_2^2)(m_1^2 + m_2^2)}}$$

Projective:

(orthogonal)

Transformation of 3D points, planes and quadrics

For a point transformation (cfr. 2D equivalent)

$$X' = \mathbf{H}\,X \qquad (x' = \mathbf{H}\,x)$$

Transformation for lines

$$\pi' = \mathbf{H}^{-T}\,\pi \qquad (l' = \mathbf{H}^{-T}\,l)$$

Transformation for quadrics

$$Q' = H^{-T} Q H^{-1} \qquad (\mathbf{C}' = \mathbf{H}^{-T}\mathbf{C}\mathbf{H}^{-1})$$

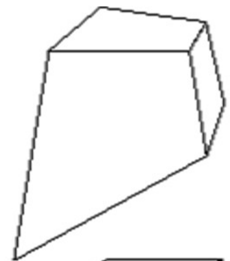
Transformation for dual quadrics

$$Q'^{*} = H Q^{*} H^{T} \qquad (\mathbf{C}'^{*} = \mathbf{H}\mathbf{C}^{*}\mathbf{H}^{T})$$

Hierarchy of 3D transformations

Projective
15dof

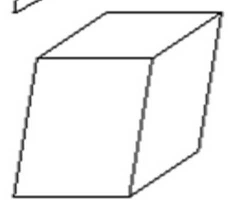
$$\begin{bmatrix} \mathrm{A} & \mathrm{t} \\ \mathrm{v}^{\mathsf{T}} & v \end{bmatrix}$$



Intersection and tangency

Affine
12dof

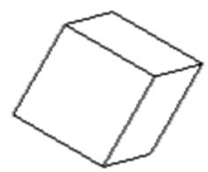
$$\begin{bmatrix} \mathrm{A} & \mathrm{t} \\ 0^{\mathsf{T}} & 1 \end{bmatrix}$$



Parallellism of planes,
Volume ratios, centroids,
The plane at infinity π_∞

Similarity
7dof

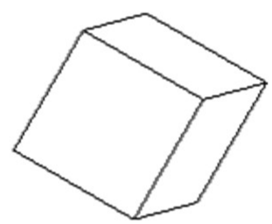
$$\begin{bmatrix} s\mathrm{R} & \mathrm{t} \\ 0^{\mathsf{T}} & 1 \end{bmatrix}$$



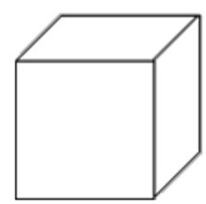
Angles, ratios of length
The absolute conic Ω_∞

Euclidean
6dof

$$\begin{bmatrix} \mathrm{R} & \mathrm{t} \\ 0^{\mathsf{T}} & 1 \end{bmatrix}$$



Volume



The plane at infinity

$$\pi'_{\infty} = \mathbf{H}_A^{-\mathsf{T}} \pi_{\infty} = \begin{bmatrix} \mathbf{A}^{-\mathsf{T}} & 0 \\ -\mathbf{t}^{\mathsf{T}}\mathbf{A}^{-\mathsf{T}} & 1 \end{bmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \pi_{\infty}$$

The plane at infinity π_{∞} is a fixed plane under a projective transformation H iff H is an affinity

1. canonical position $\pi_{\infty} = (0,0,0,1)^{\mathsf{T}}$
2. contains directions $\mathrm{D} = (X_1, X_2, X_3, 0)^{\mathsf{T}}$
3. two planes are parallel $\Leftrightarrow$ line of intersection in π_{∞}
4. line // line (or plane) $\Leftrightarrow$ point of intersection in π_{∞}

The absolute conic

The absolute conic Ω_∞ is a (point) conic on π_∞.

In a metric frame:

$$\left.\begin{matrix} X_1^2 + X_2^2 + X_3^2 \\ X_4 \end{matrix}\right\} = 0$$

or conic for directions: $(X_1, X_2, X_3)\mathrm{I}(X_1, X_2, X_3)^\mathsf{T}$

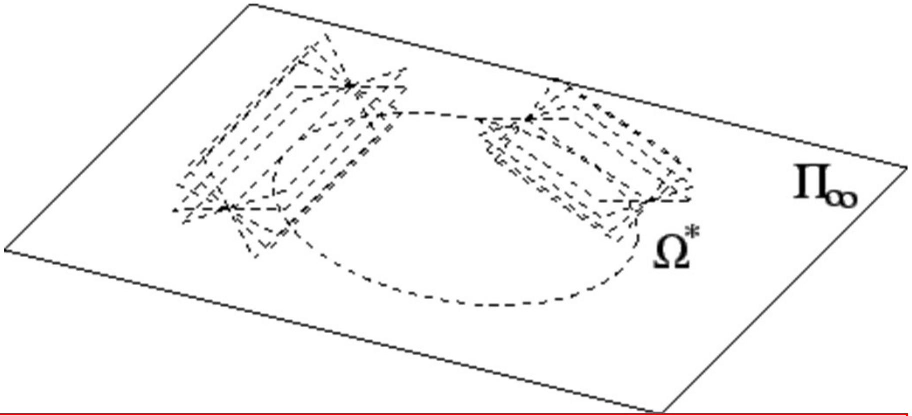
(with no real points)

> The absolute conic Ω_∞ is a fixed conic under the projective transformation $\mathbf{H}$ iff $\mathbf{H}$ is a similarity

1. Ω_∞ is only fixed as a set
2. Circle intersect Ω_∞ in two circular points
3. Spheres intersect π_∞ in Ω_∞

The absolute dual quadric

$$\Omega^*_\infty = \begin{bmatrix} I & 0 \\ 0^T & 0 \end{bmatrix}$$



The absolute dual quadric Ω^*_∞ is a fixed conic under the projective transformation $\mathbf{H}$ iff $\mathbf{H}$ is a similarity

1. 8 dof
2. plane at infinity π_∞ is the nullvector of Ω_∞
3. Angles:

$$\cos\theta = \frac{\pi_1^T \Omega^*_\infty \pi_2}{\sqrt{\left(\pi_1^T \Omega^*_\infty \pi_1\right)\left(\pi_2^T \Omega^*_\infty \pi_2\right)}}$$

Next week: camera models