MAEG 5720: Computer Vision in Practice

Lecture 1:

Course Introduction

Dr. Terry Chang

2021-2022

Semester 1

香港中文大學
The Chinese University of Hong Kong

Department of Mechanical and Automation Engineering
機械與自動化工程學系

Welcome to Computer Vision class!

Today’s Agenda

- Introduction to Computer Vision
- Some Applications
- Course structure
- Questions

A picture is worth a thousand words

Image Courtesy: https://brightside.me

A picture is worth a thousand words

Image Courtesy: blog.livedoor.jp

What is Computer Vision?

Game

- I shall show you a picture for 0.1 sec. Tell me what you see

- Ready?

Game

One more time?

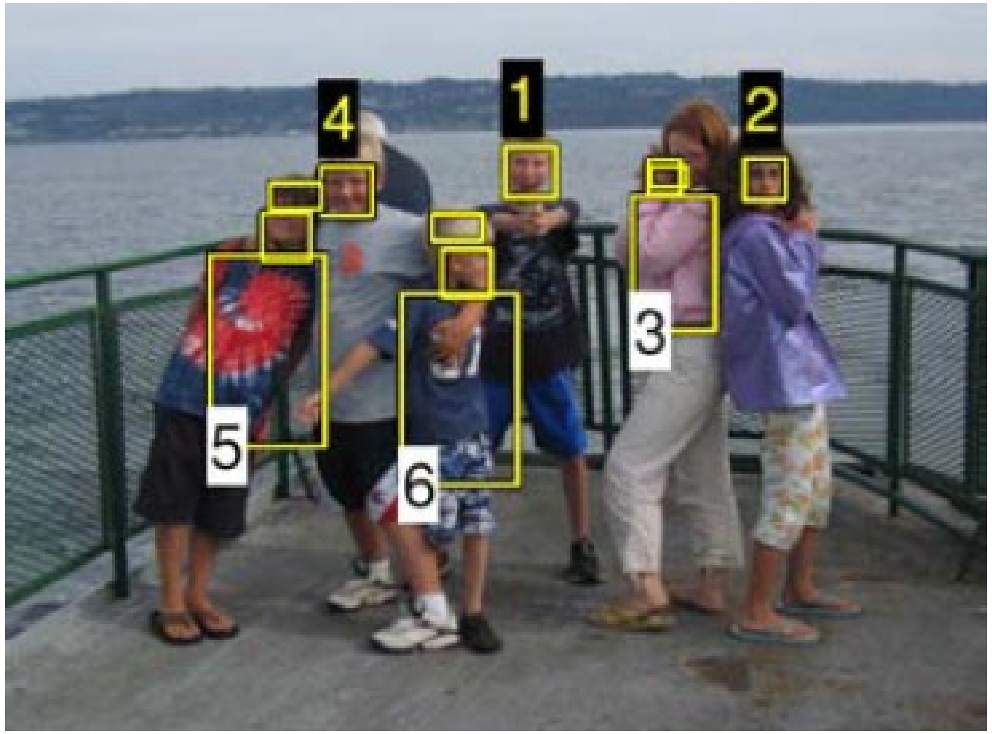
What Did you See?

What did you See?

Blue Sky, One Plane,
4 people

Other Examples: What can you tell about the pictures?

- ***Shape*** and ***Translucency*** of each petal through the subtle patterns of light and shading.
- Effortlessly ***segment*** each flower from the background of the scene



- ***Easily count (and name)*** the people in the picture
- ***Guess*** their ***emotions*** from their facial appearances.

Image Courtesy: R. Szeliski

What is Computer Vision?

https://en.wikipedia.org/wiki/Tower_Bridge

To Make Computer See and Understand!

Image

Sensing Device

Interpretation Device

Image Interpretation

Bridge (Tower Bridge)

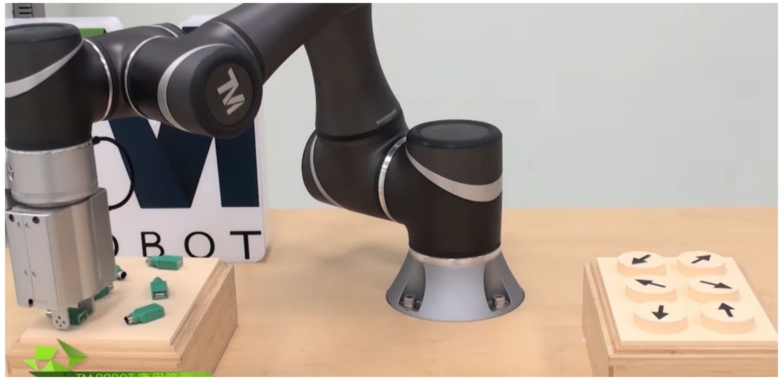
What is Computer Vision?

Sources: https://www.fiverr.com/rupesh_w/object-detection-and-tracking

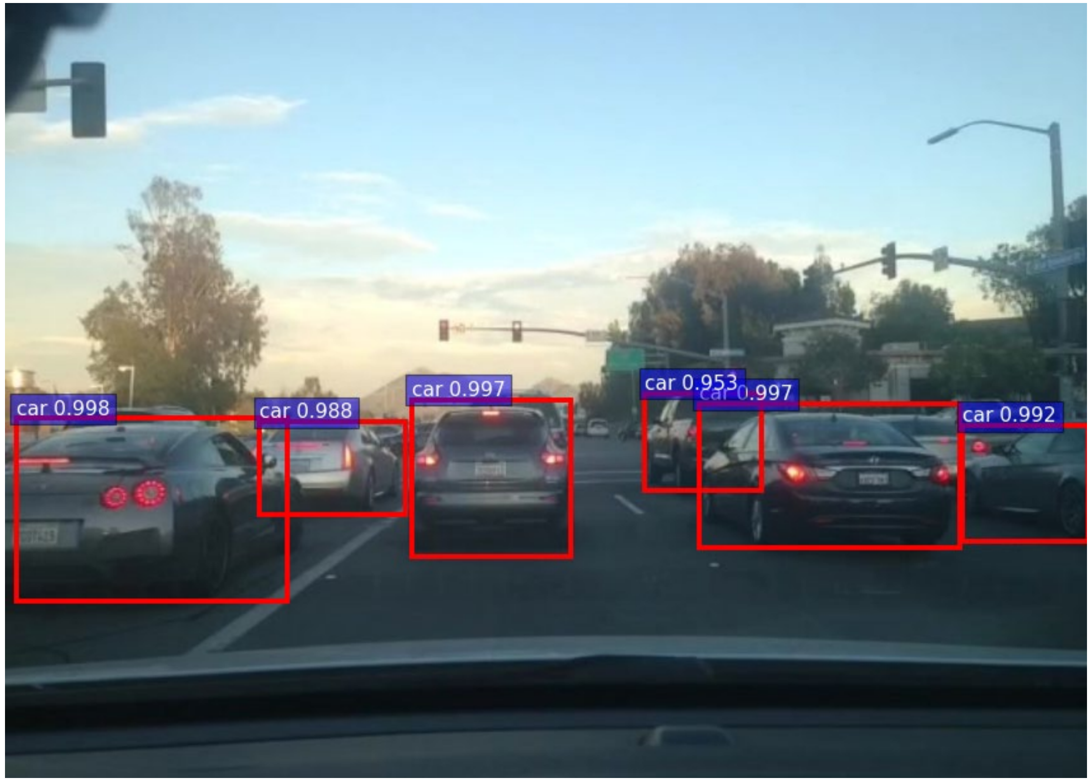
- Allowing the computer to ***see*** and ***understand***!

What is Computer Vision?

- Once the computer can see, the robot can react.



Robot can perform human tasks



https://www.xsens.com/customer-cases/vector-ai-shaping-the-future-for-autonomous-driving/

Origins of computer vision: an MIT undergraduate summer project

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

Artificial Intelligence Group July 7, 1966

Vision Memo. No. 100.

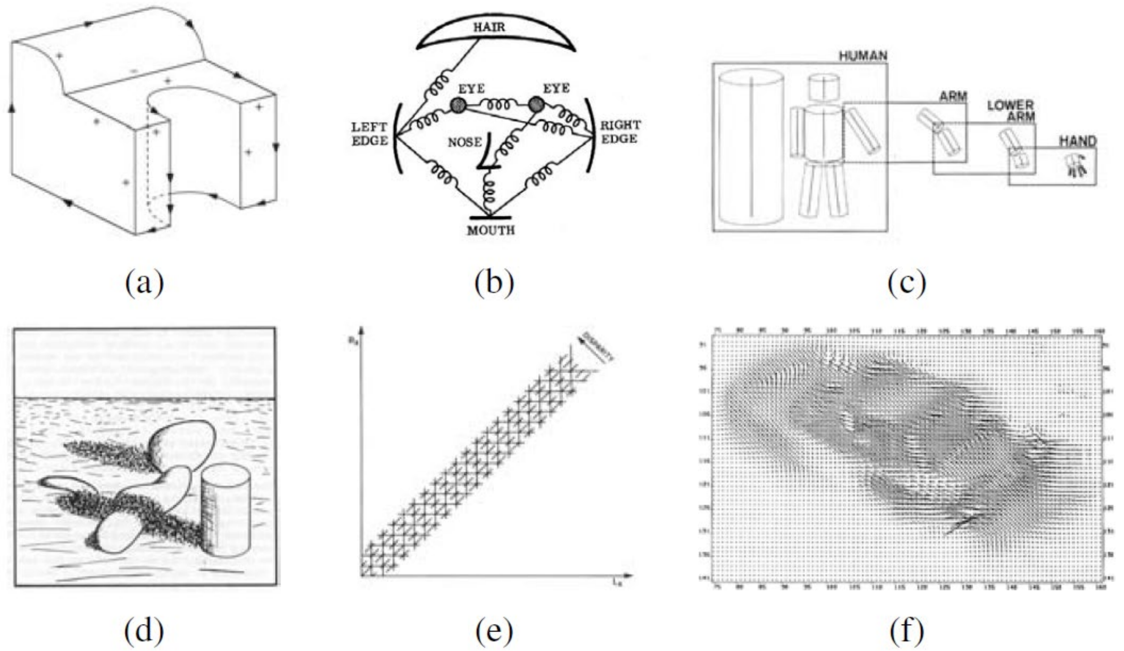
THE SUMMER VISION PROJECT

Seymour Papert

The summer vision project is an attempt to use our summer workers effectively in the construction of a significant part of a visual system. The particular task was chosen partly because it can be segmented into sub-problems which will allow individuals to work independently and yet participate in the construction of a system complex enough to be a real landmark in the development of "pattern recognition".

Papert, Seymour (1966-07-01). "The Summer Vision Project". MIT AI Memos (1959 - 2004). hdl:1721.1/6125

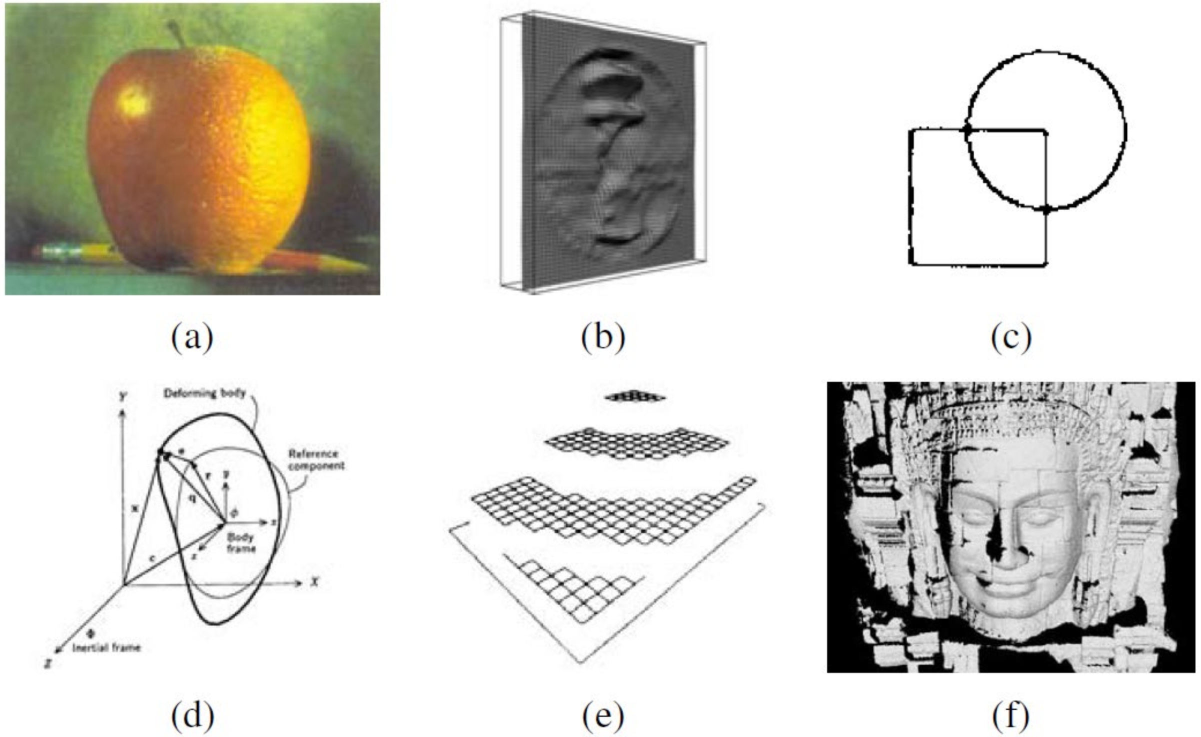
Example of works in 70's



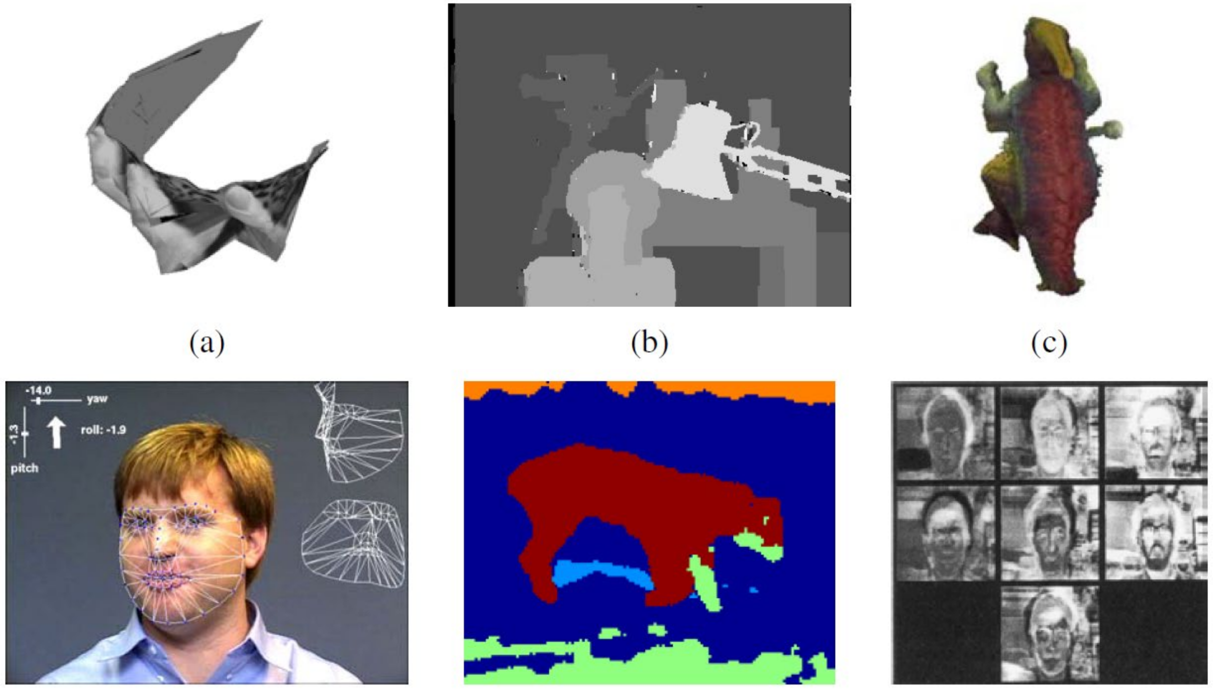
a. Line labelling
b. Pictorial structures
c. Articulated body model
d. Intrinsic images
e. Stereo Correspondence
f. Optical flow

Example of works in 80’s

a. Image pyramid & blending
b. Shape from shading
c. Edge detection
d. Physical based model
e. Regularization-base surface reconstruction
f. Range data acquisition and merging



Example of works in 90's

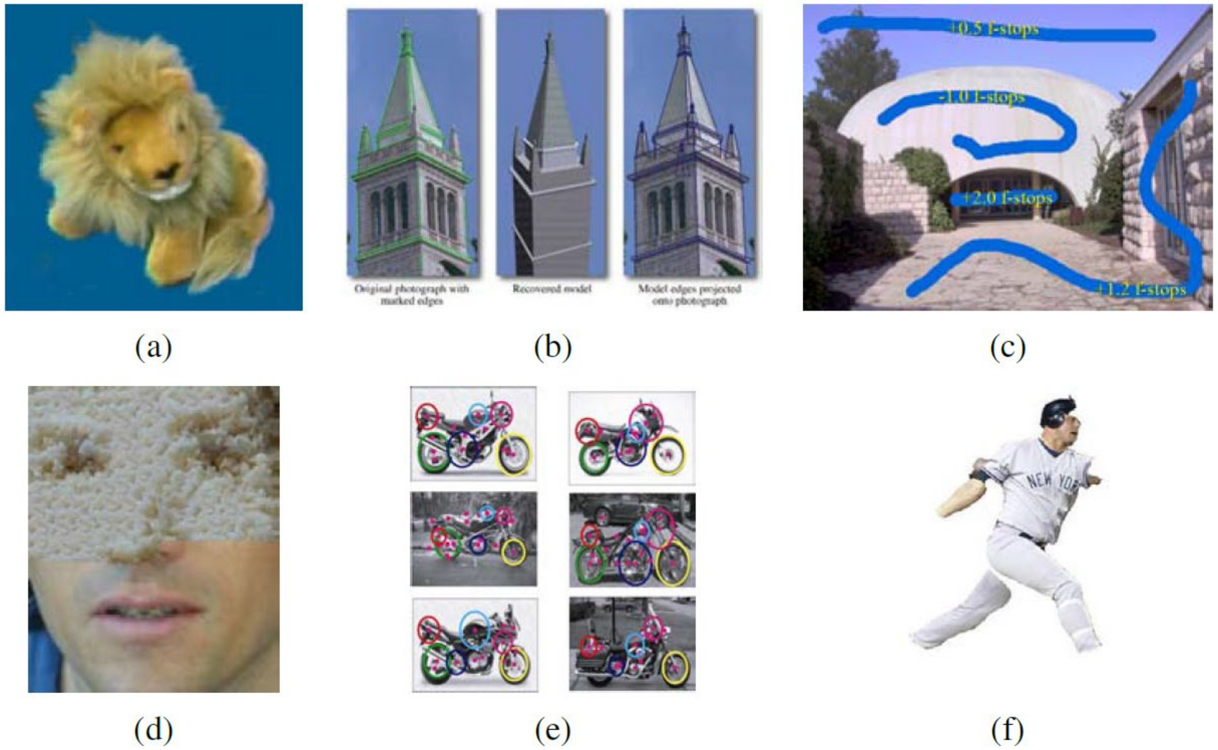


(d) (e) (f)

a. Structure from motion
b. Dense stereo matching
c. Multi-view reconstruction
d. Face tracking
e. Image segmentation
f. Face recognition

Example of works in 2000s

a) Image-based rendering
b) Image-based modelling
c) Interactive tone mapping
d) Texture synthesis
e) Feature-based Recognition
f) Region-based Recognition



Brief History of Computer Vision - timeline

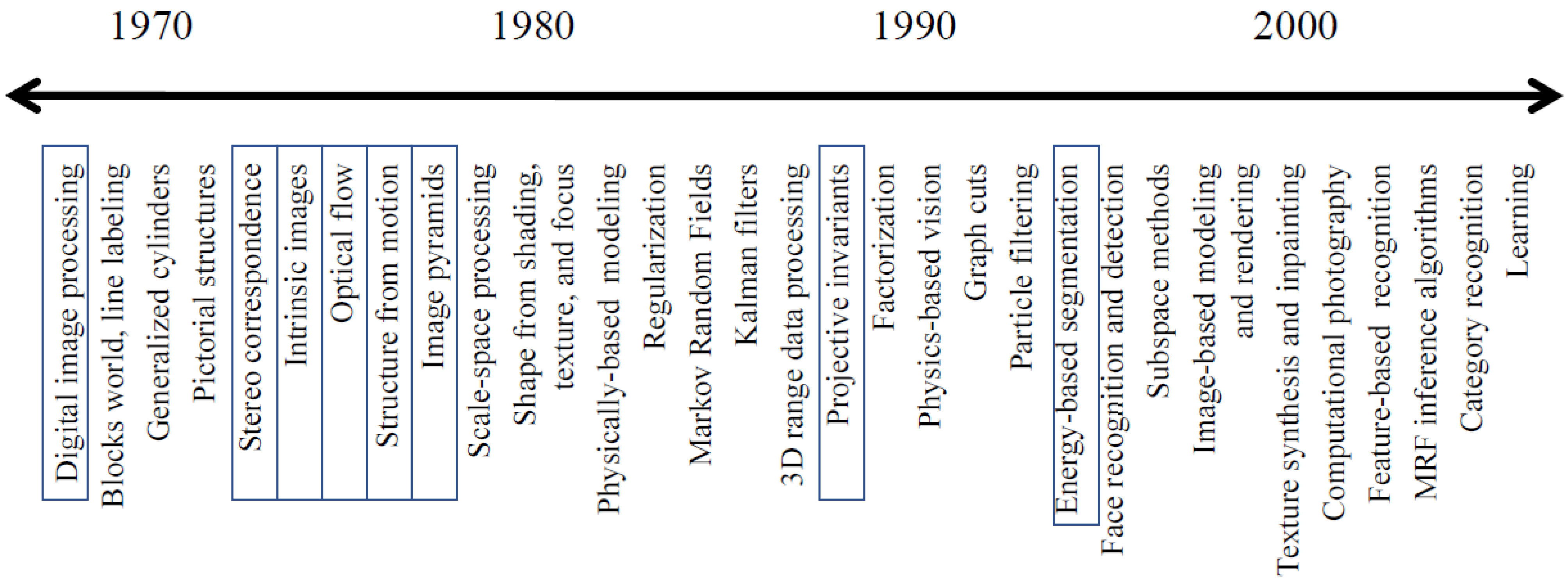
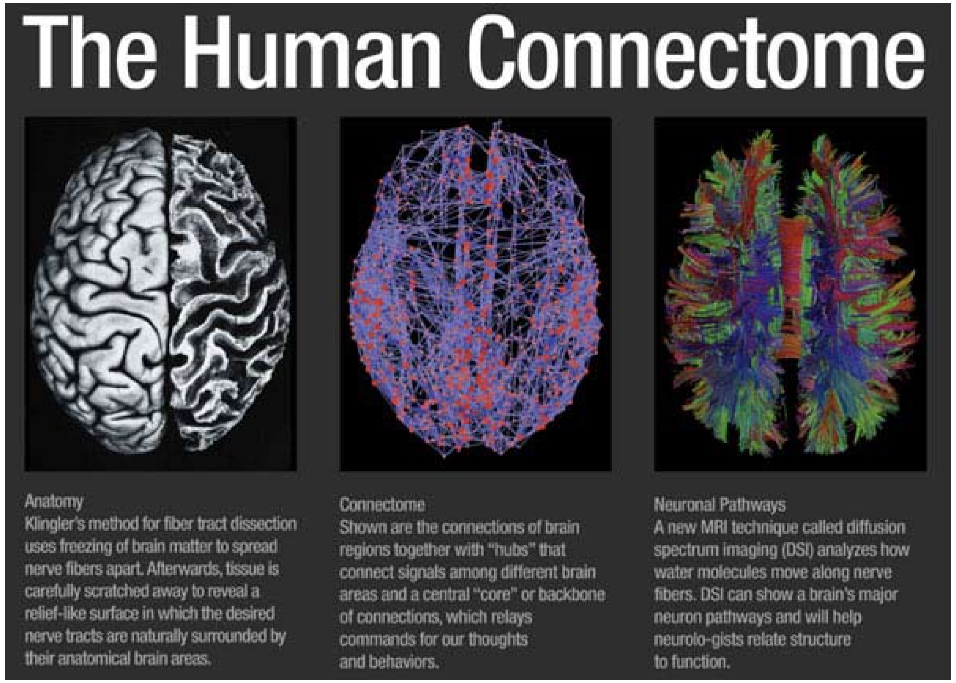


Image Credit: Richard Szeliki, “Computer Vision”

Why Computer Vision is really hard?

- Vision is an amazing feature of natural intelligence
  - Visual cortex occupies about 50% of Macaque brain
  - More human brain devoted to vision than anything else



Source: S. Seung

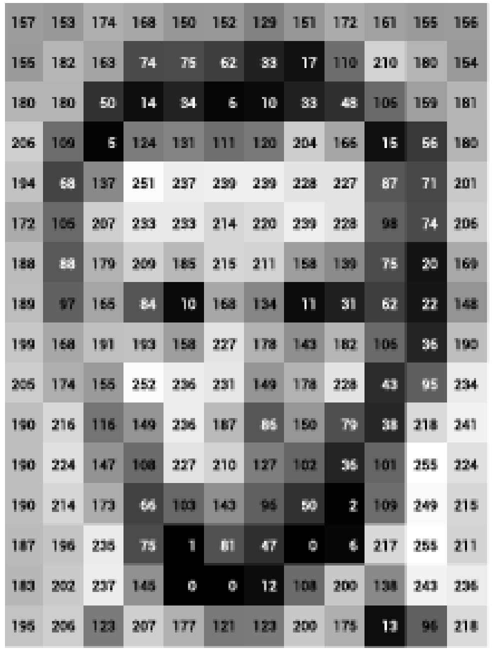
Why Computer vision is really hard?

157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

Source: http://techundred.com/how-snapchat-filter-work/

- Grey Image is represented as matrix of integers from 0 to 255
  - 0 is black color
  - 255 is white color

Vision is really hard



157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

Source: http://techundred.com/how-snapchat-filter-work/

- Grey Image is represented as matrix of integers from 0 to 255
  - 0 is black color
  - 255 is white color

The goal of Computer Vision

163	162	161	160	163	157	163	162	165	161	163	161	154	165	159	154	159	153	163	161	153	157
162	162	162	161	163	156	163	162	165	161	162	160	154	165	159	154	158	153	163	161	153	156
162	162	163	161	163	157	163	162	166	161	161	160	155	164	160	155	157	155	163	161	154	155
162	163	163	161	162	157	163	162	166	162	160	161	155	163	161	155	157	157	162	161	155	155
164	164	162	159	161	157	164	162	164	162	158	161	154	160	160	154	156	158	160	159	155	153
163	163	161	157	161	157	163	159	161	161	156	161	153	156	158	152	154	157	156	157	155	152
161	161	159	157	161	158	161	155	159	161	157	162	152	153	156	151	154	158	153	155	156	151
159	159	158	156	162	158	160	152	158	162	158	163	153	153	156	151	155	159	152	155	157	152
155	156	157	157	158	159	159	158	168	157	159	159	162	154	157	156	155	158	160	158	155	154
155	156	157	158	158	157	155	154	161	155	157	156	158	154	157	154	154	154	154	155	157	158
156	156	157	159	158	154	151	151	158	156	159	156	156	155	158	155	156	156	155	154	156	158
157	156	157	159	158	154	151	151	160	158	159	158	156	155	157	157	156	159	159	157	155	155
157	156	157	158	158	155	154	155	160	156	156	159	155	153	155	159	155	158	159	157	156	156
158	157	157	157	157	157	157	157	159	154	153	160	155	154	155	160	159	158	156	155	157	158
157	159	159	156	156	158	158	156	158	155	154	161	155	157	156	158	159	158	156	155	156	157
157	160	160	156	155	158	157	153	156	157	156	161	154	158	155	153	153	156	158	157	155	155
158	156	156	158	158	156	155	154	156	156	154	160	156	158	154	158	154	155	154	160	160	155
158	157	157	158	158	156	154	153	158	158	156	156	154	156	155	155	154	155	153	157	157	153
157	156	157	158	158	156	154	153	157	158	158	153	155	157	160	158	159	159	156	158	156	152
155	155	156	158	158	157	156	155	157	156	158	154	158	157	162	161	161	162	159	159	157	153
155	156	157	157	157	157	157	156	160	155	158	157	161	155	158	159	156	159	156	158	157	154
157	158	158	156	155	156	156	156	158	153	155	159	161	155	156	159	158	160	157	158	156	153
157	158	157	155	153	155	156	156	156	154	154	159	159	158	157	161	161	162	158	158	157	155
155	157	156	153	152	154	156	157	158	159	157	160	156	160	157	159	158	159	155	156	158	157
155	157	156	152	152	156	159	159	160	160	156	159	159	156	159	159	160	160	156	157	157	155
153	154	154	151	152	156	158	158	158	160	157	160	160	156	159	158	158	159	156	159	159	155
153	154	154	152	153	157	157	156	157	159	159	162	161	156	158	156	159	158	156	161	160	155
157	157	156	155	156	158	157	155	156	159	159	162	160	156	157	156	162	160	156	160	160	155
159	159	158	158	159	159	158	156	157	159	158	161	160	156	158	157	161	160	156	160	160	156

What computer sees

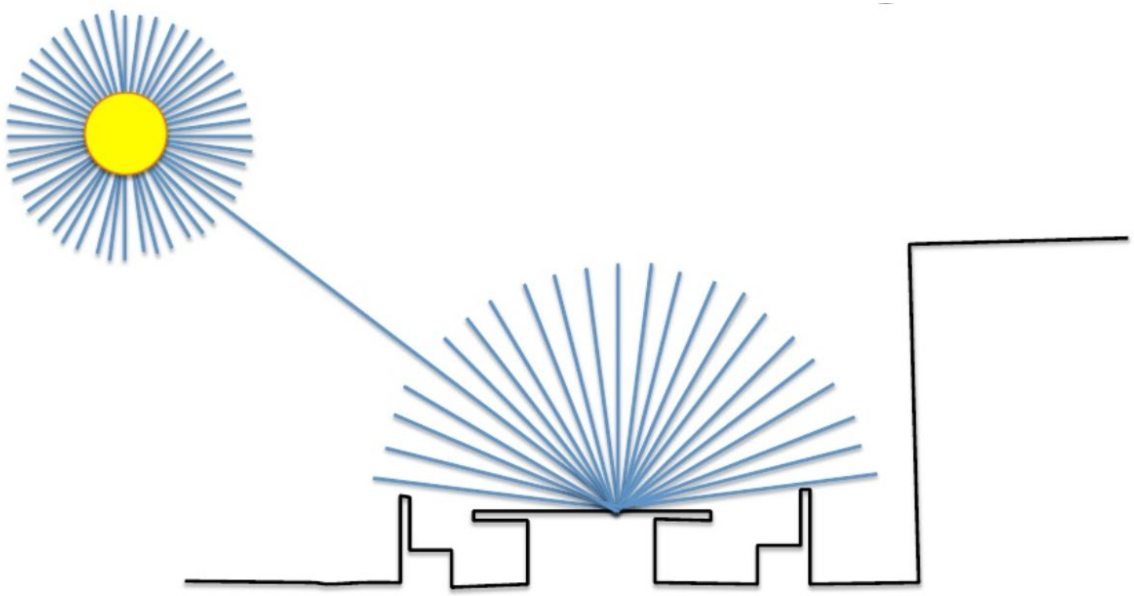
What we see

Why Computer Vision is really hard?

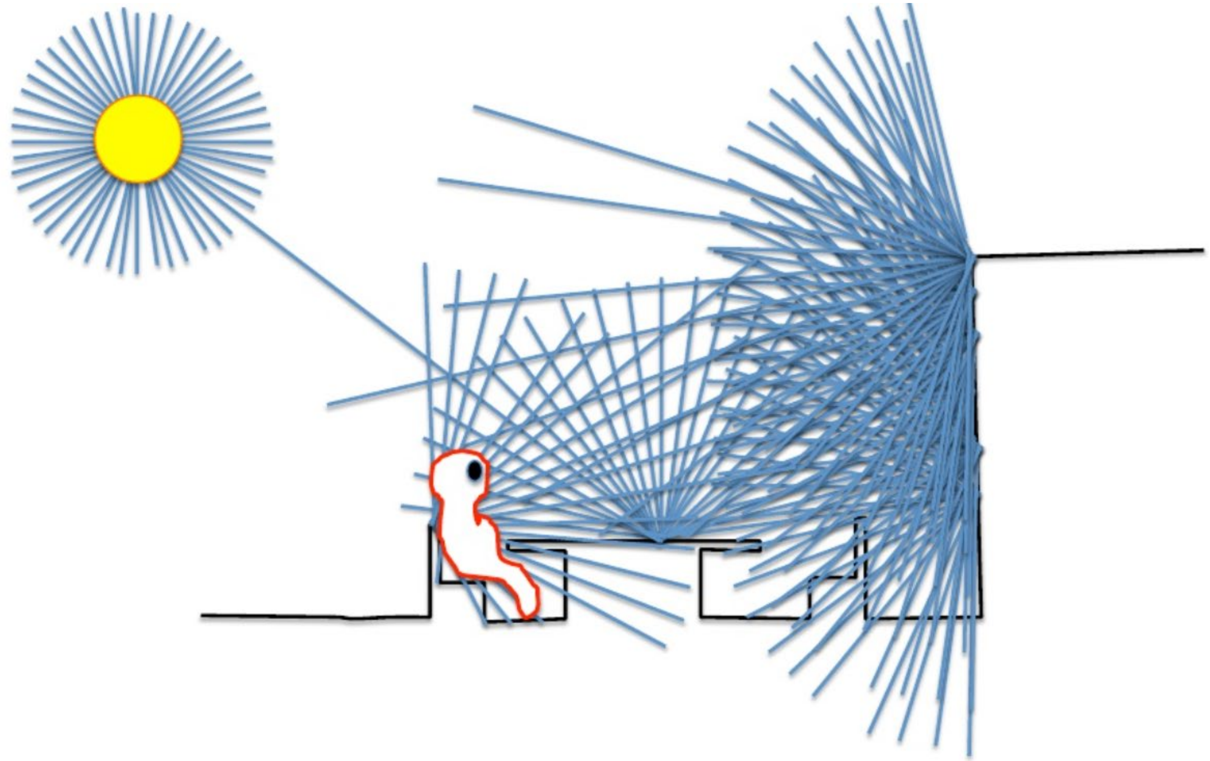
- Images are affected by different factors
  - Illumination (lighting)
  - Shadow
  - Perspective
  - Shape of Object
  - Color of Object
  - Texture of Object
- What is the Gold Standard?
  - Human Vision!

Image credit: https://www.subpng.com/png-o6e8mj/

The structure of Ambient Light

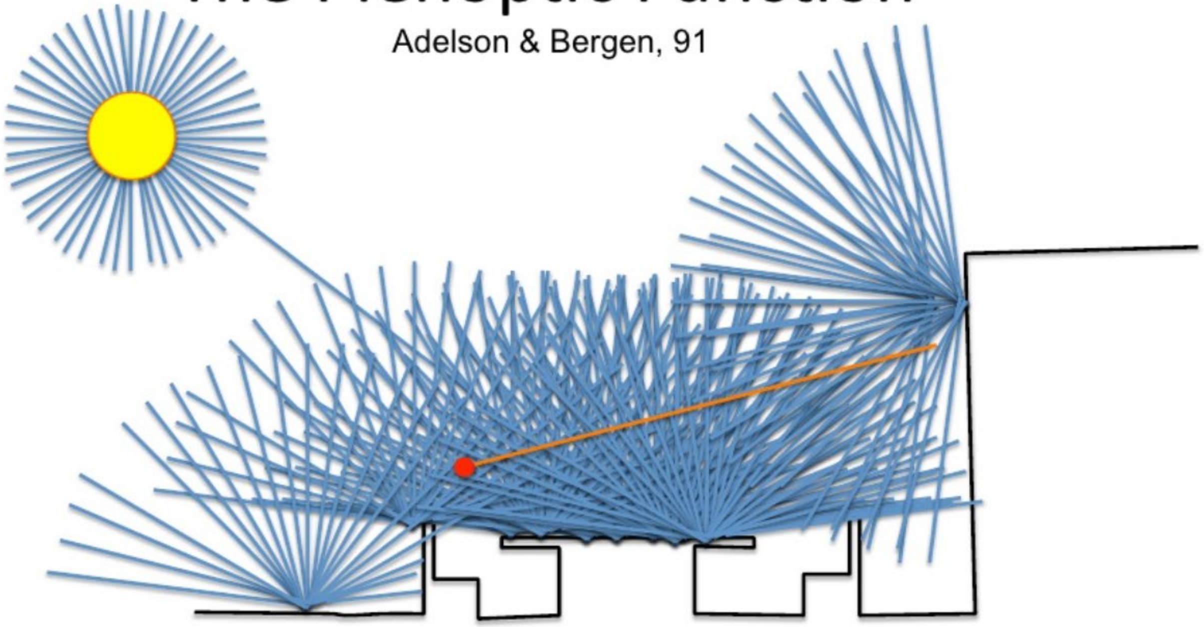


Structure of Ambient Light



The Plenoptic Function

Adelson & Bergen, 91



The intensity P can be parameterized as:

$$P\,(\theta, \phi, \lambda, t, X, Y, Z)$$

“The complete set of all convergence points constitutes the permanent possibilities of vision.” Gibson

Let’s play another game

#thedress

- What is the color of the dress?

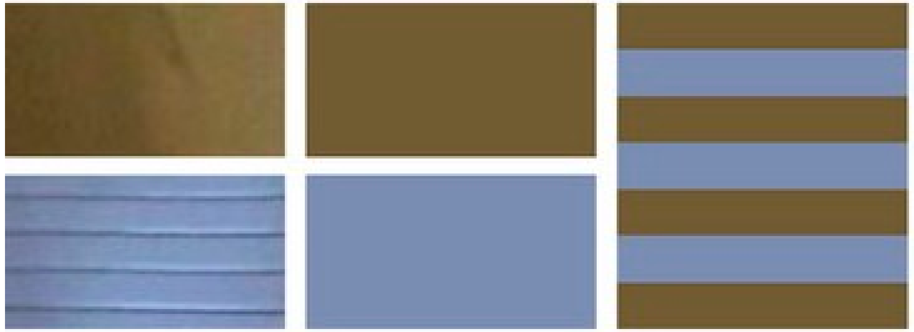
- blue and black
- white and gold
- blue and brown

- What #thedress tell about our color perception?

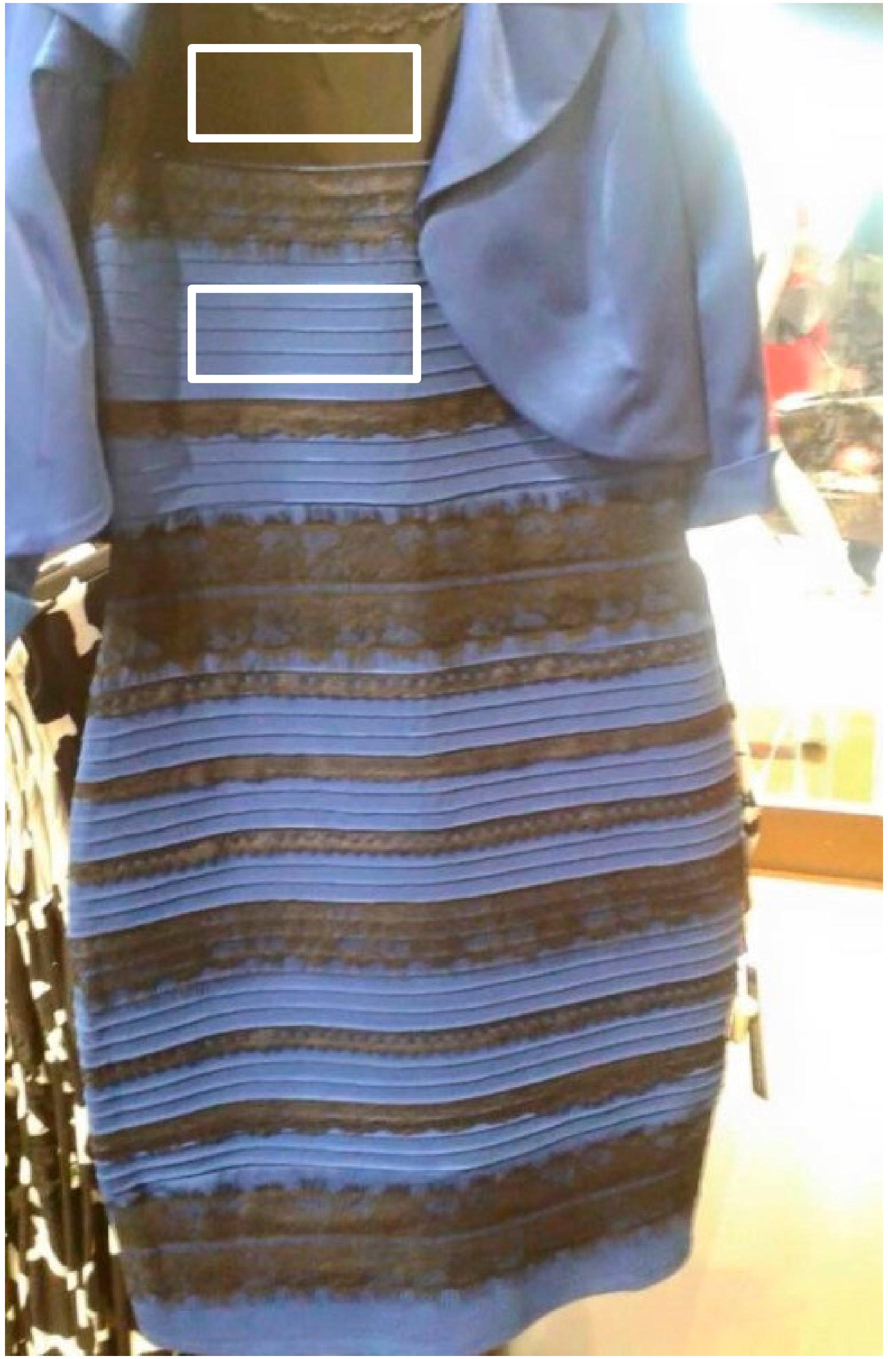
http://nyti.ms/186m3wE

#thedress

- Let’s take averages



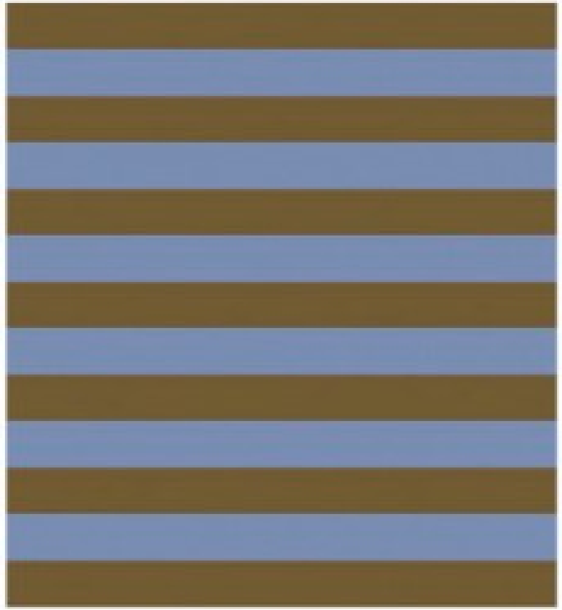
two pieces of the dress | averages | basic pattern

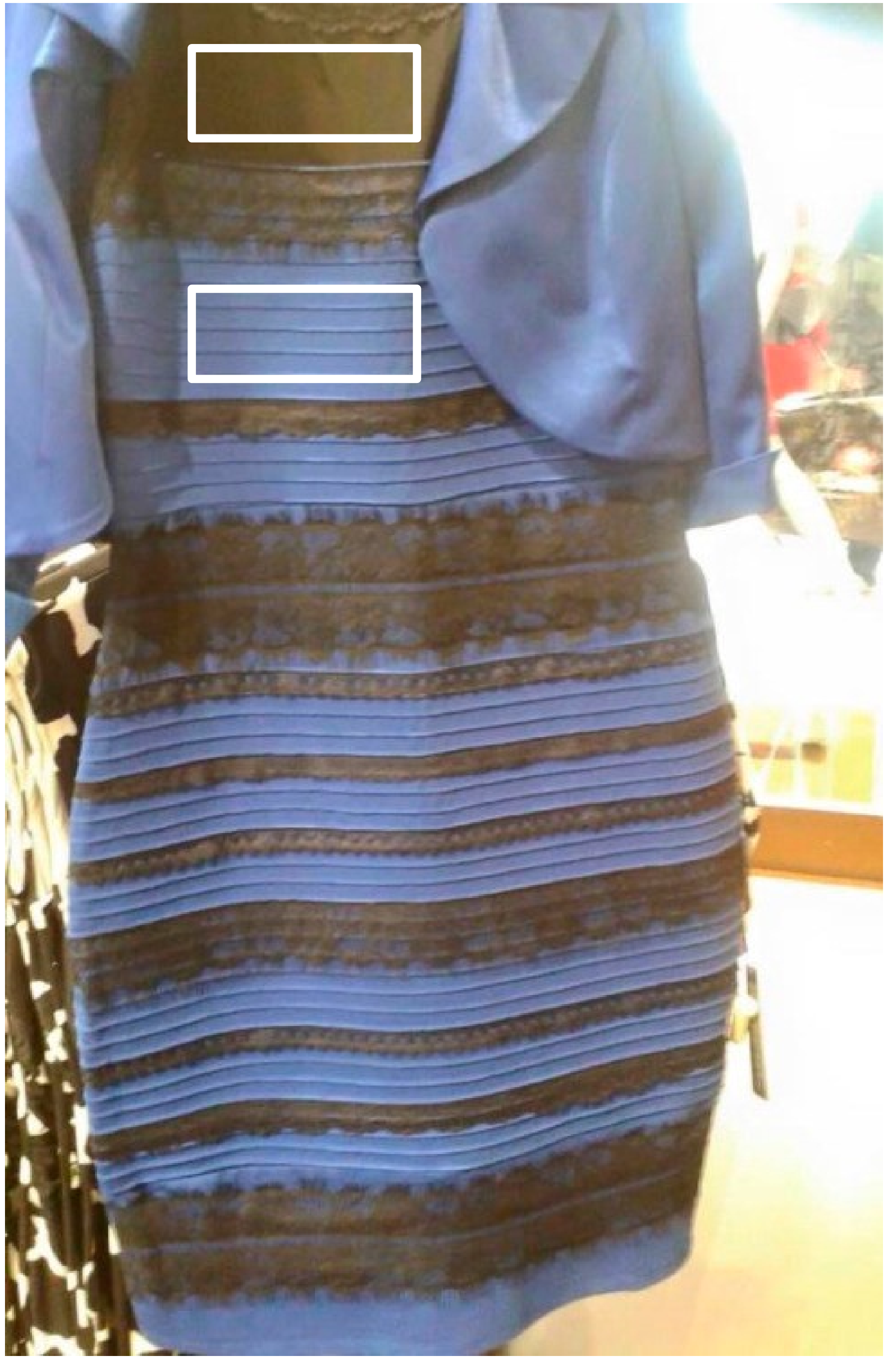


http://nyti.ms/186m3wE

#thedress

- The dress in the photograph

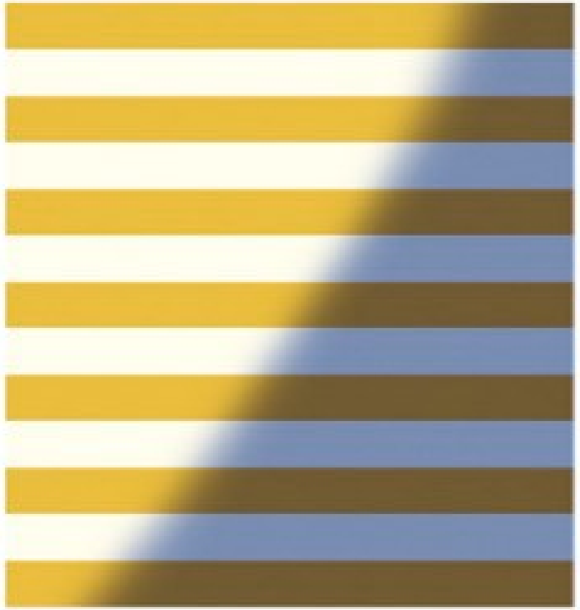




http://nyti.ms/186m3wE

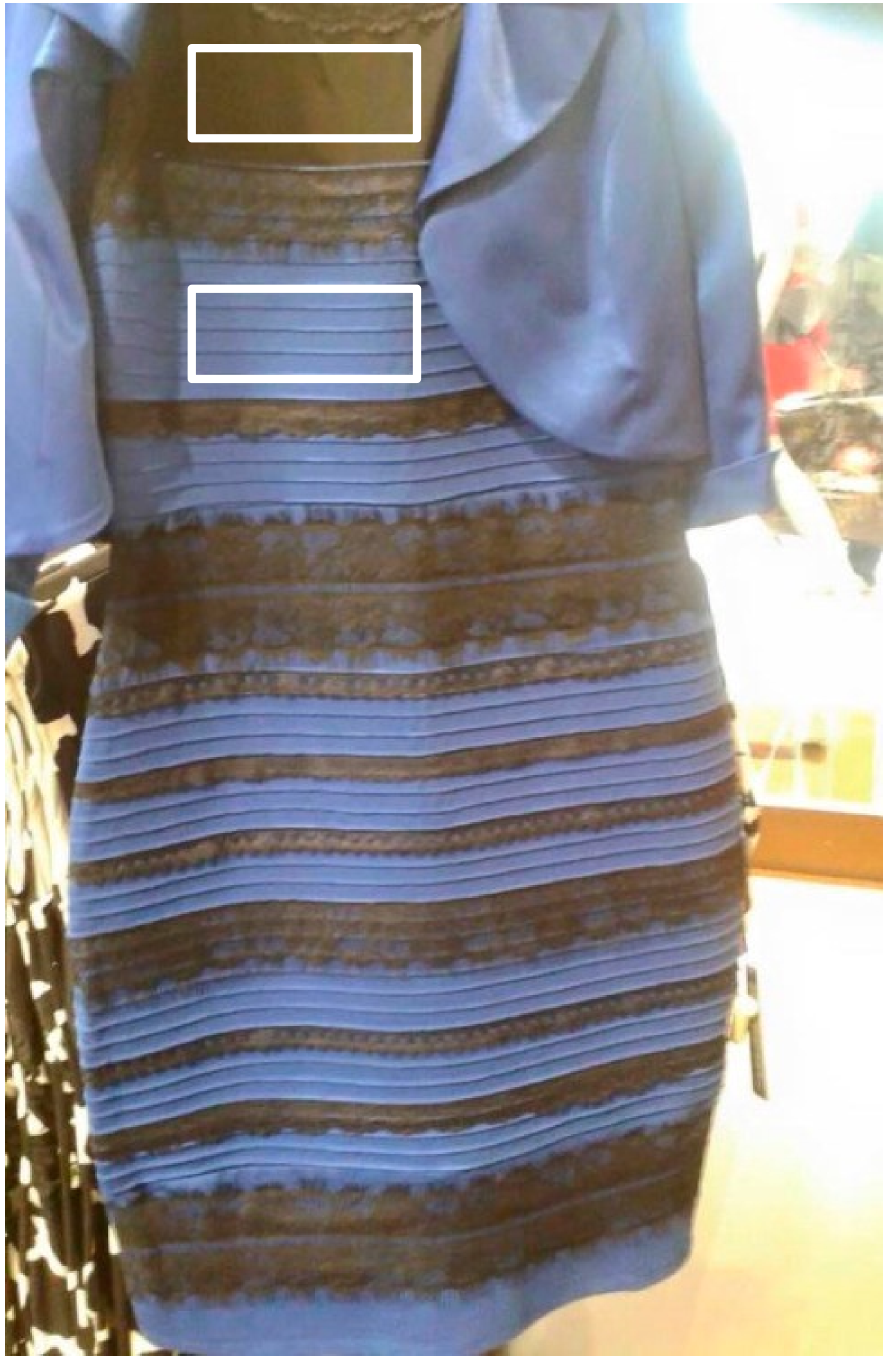
#thedress

- Consider the dress is in shadow.



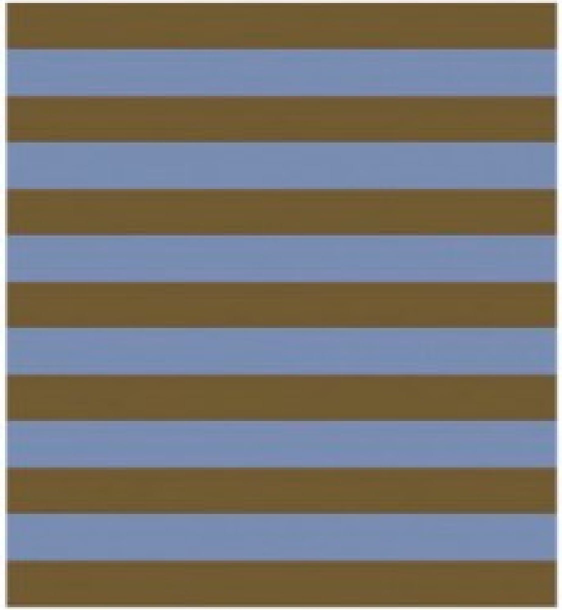
- Your brain remove the blue cast, and perceive it as white and gold.

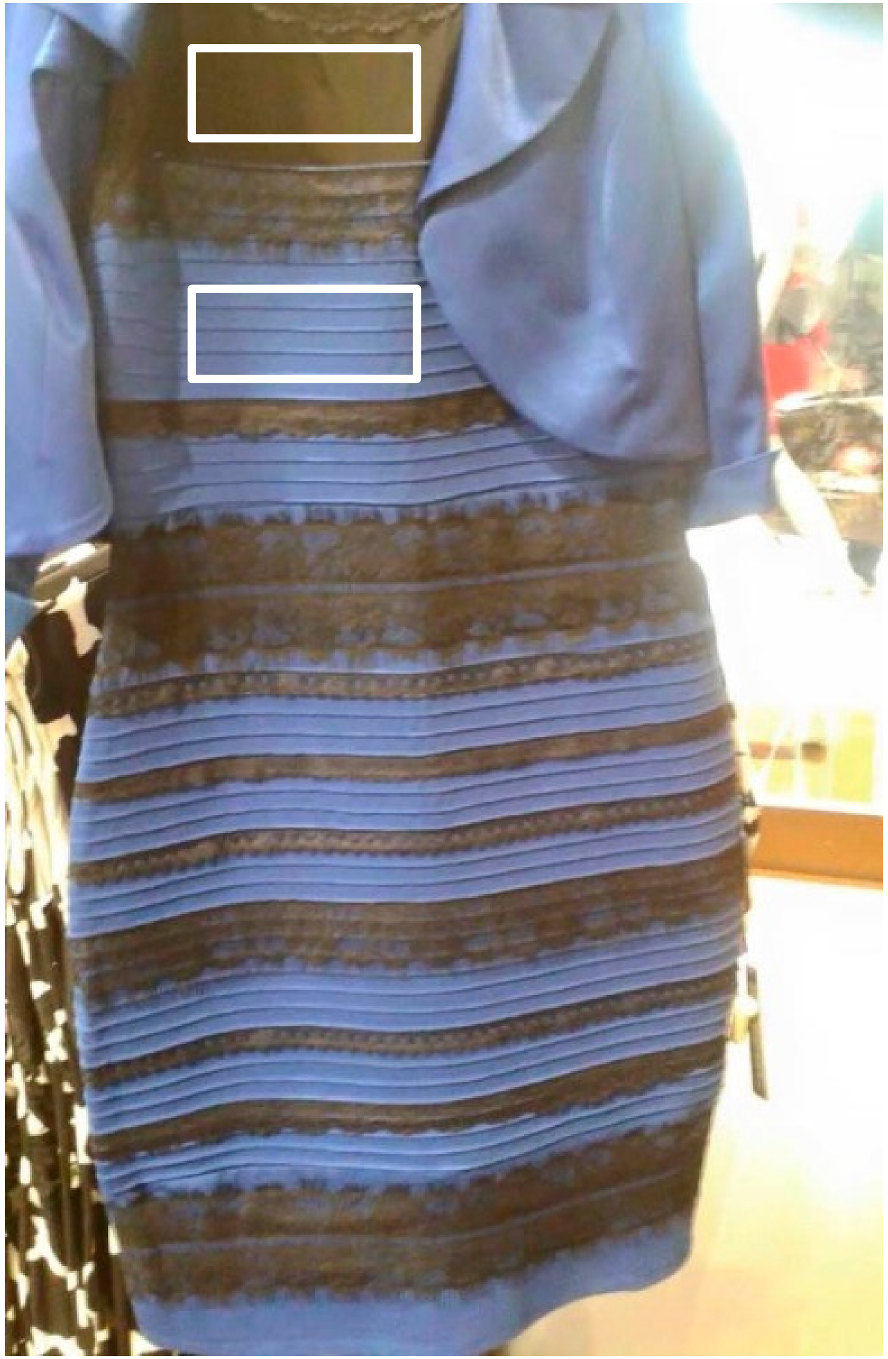
http://nyti.ms/186m3wE



#thedress

- The dress in the photograph

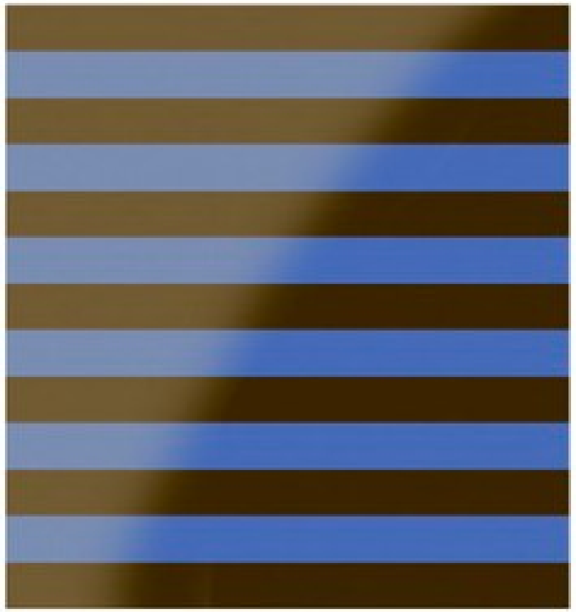




http://nyti.ms/186m3wE

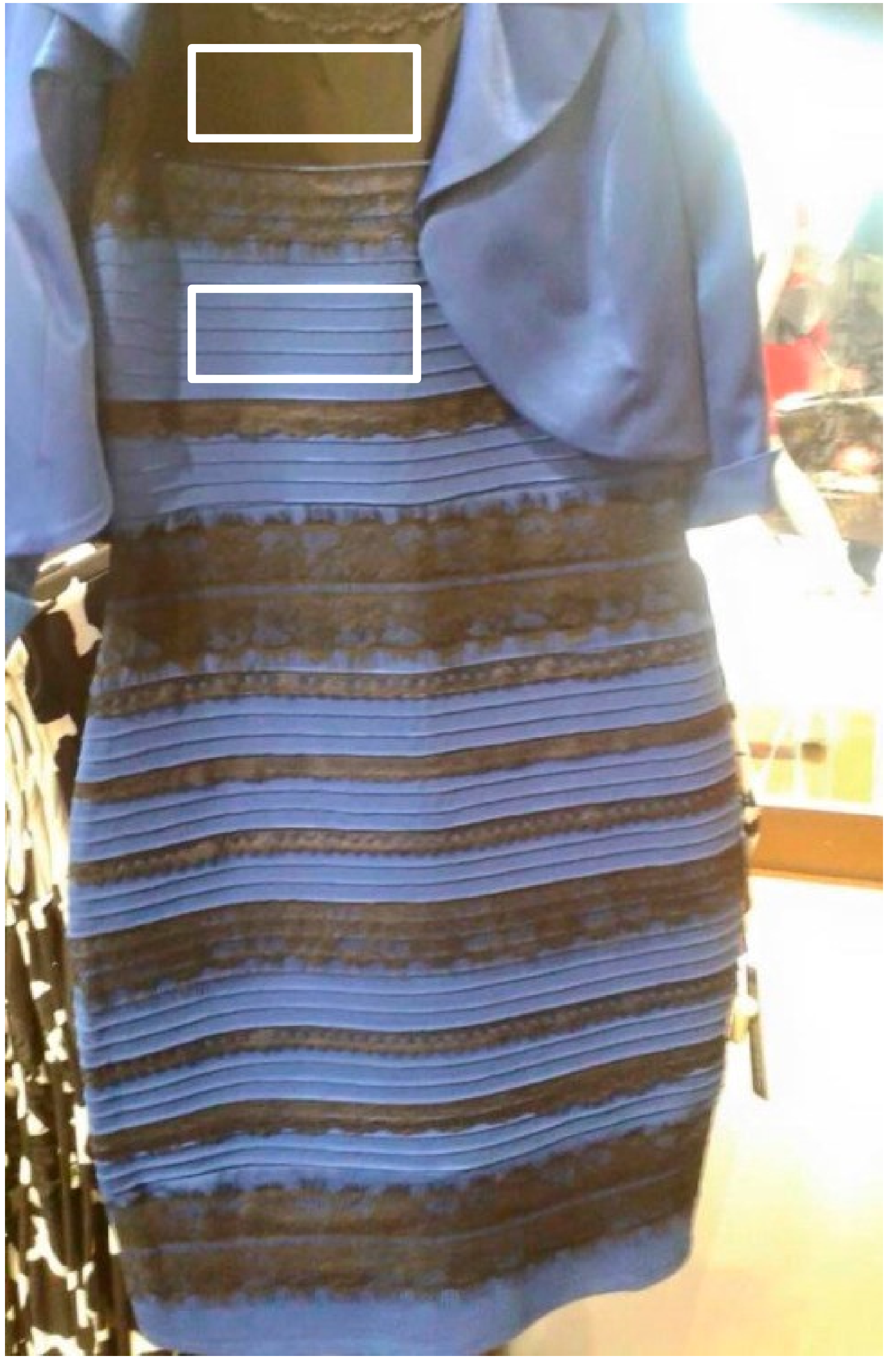
#thedress

- Consider the dress is in bright light.



- Your brain perceive the dress as a darker blue and black

http://nyti.ms/186m3wE

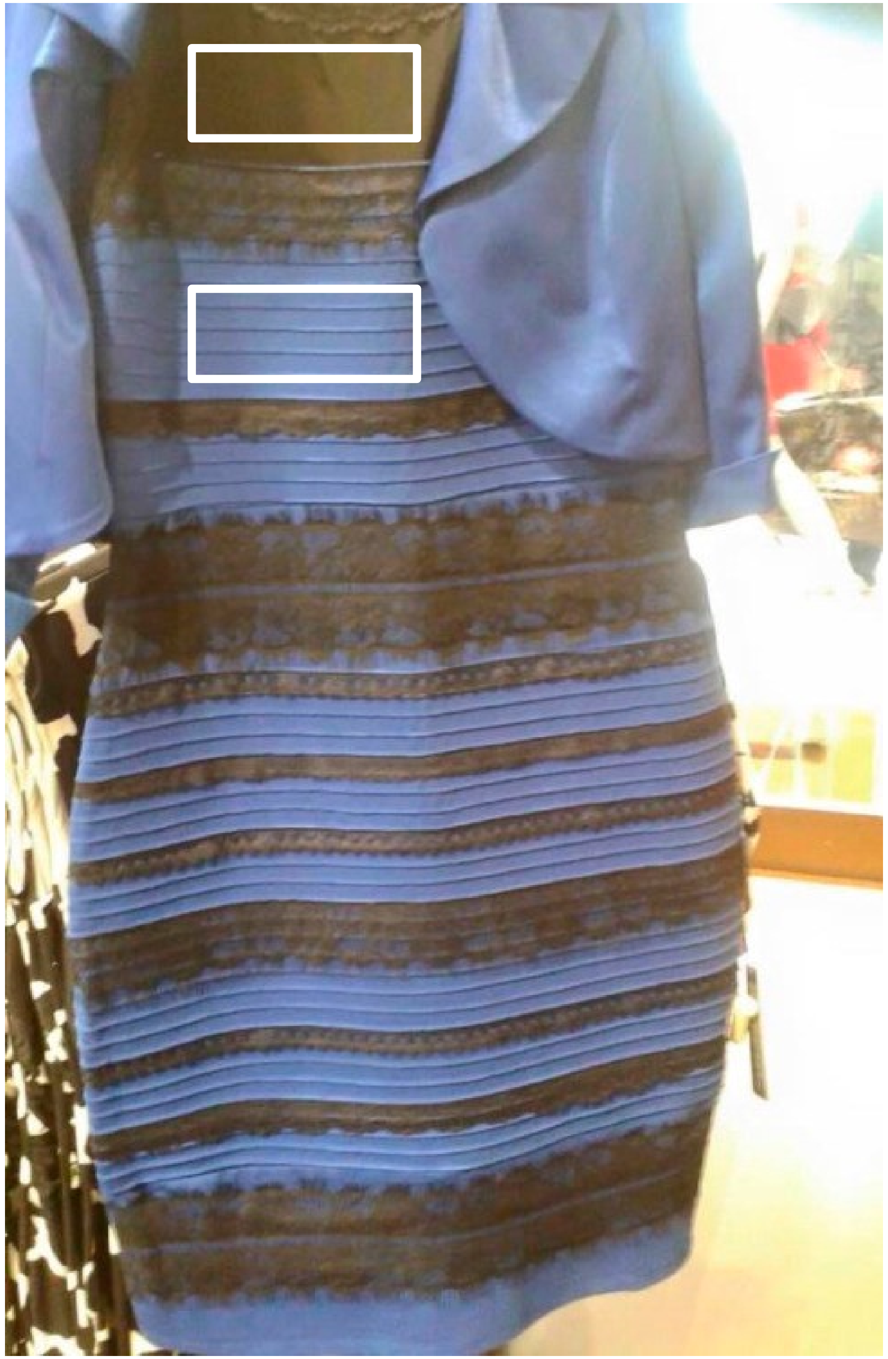


#thedress

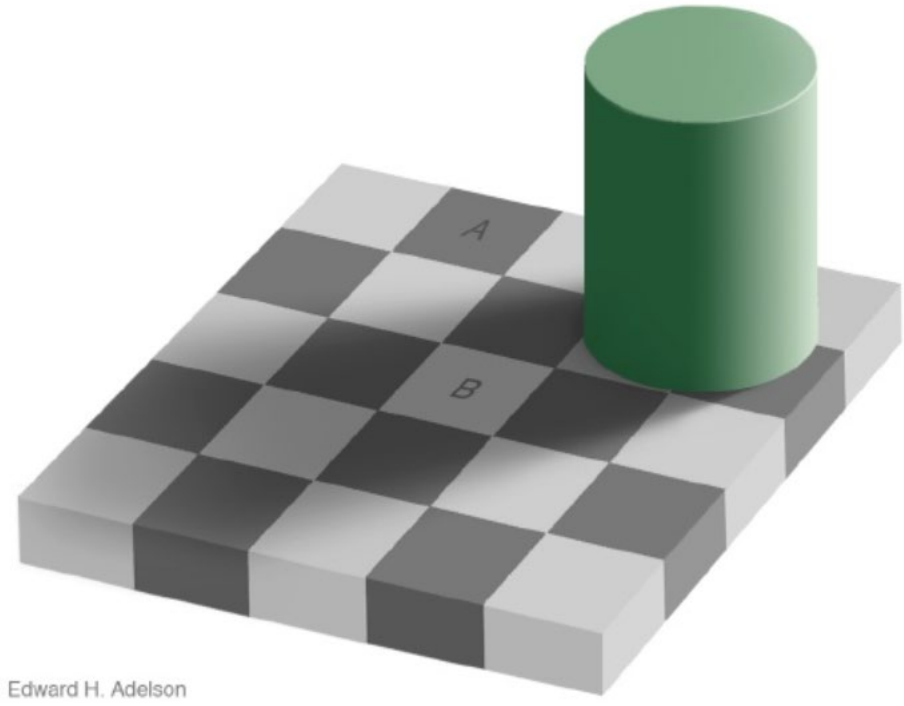
- Answer:

- The dress is actually blue and black.

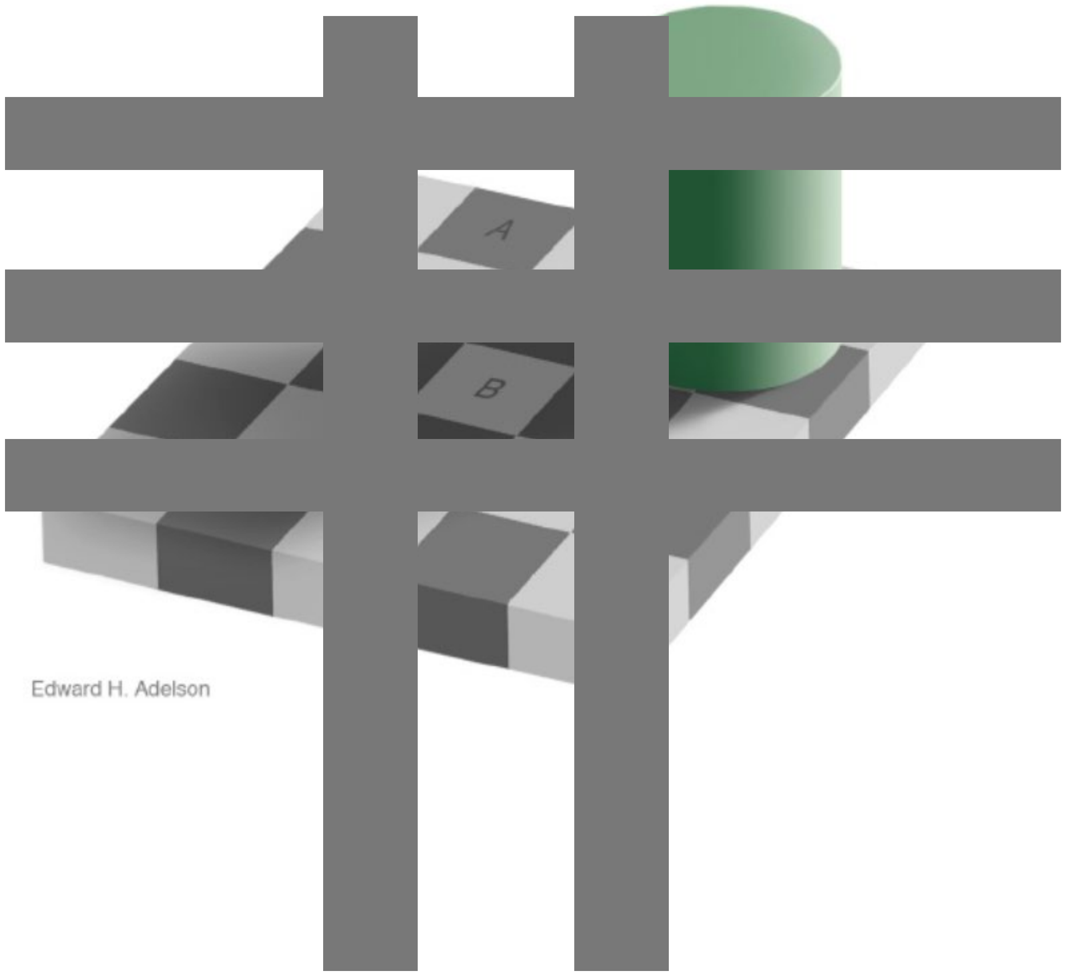
http://nyti.ms/186m3wE



Shadow

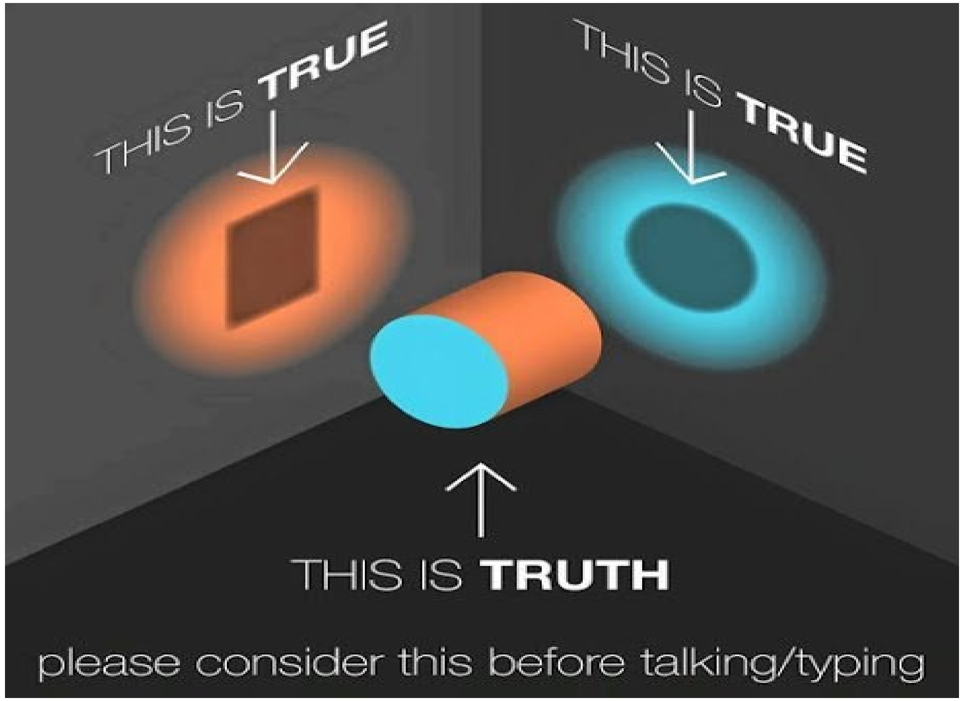


Shadow



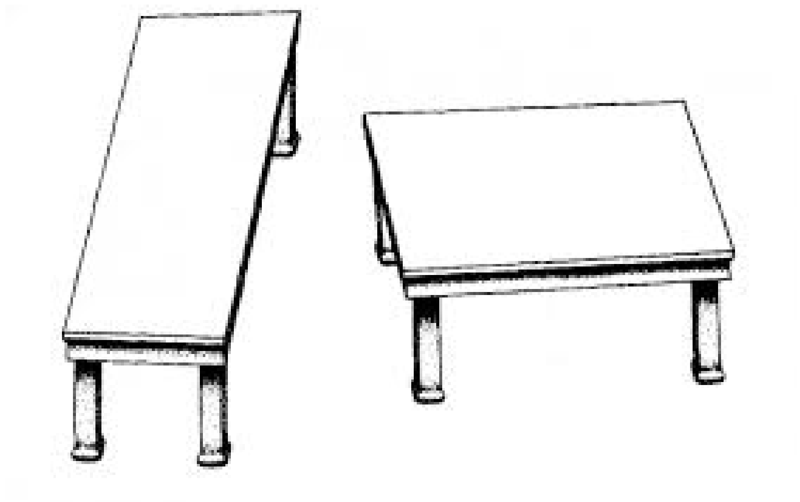
Different Perspectives

- Same object viewing from different perspectives results in different images
- Information is lost during transformation from 3D object to 2D image.

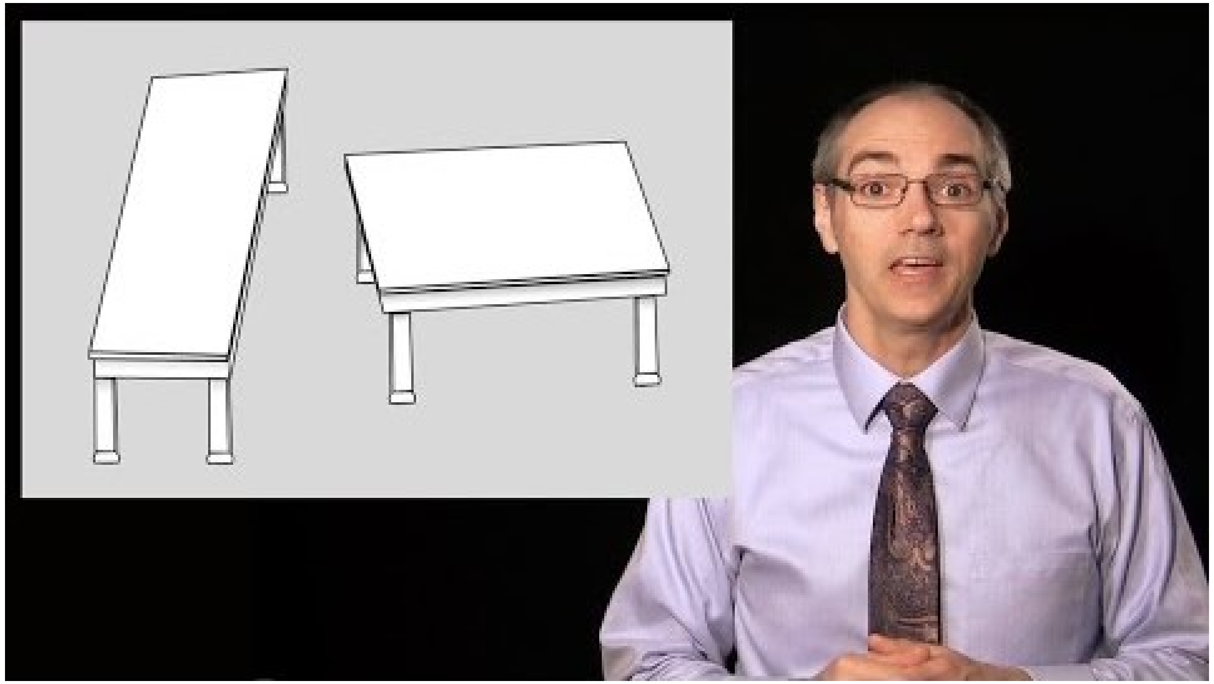


Source: Patrick @TrickFreee

Depth Perception



Turning the Tables by Roger Shepard



Why Computer Vision is Hard?

- High Shape Variation

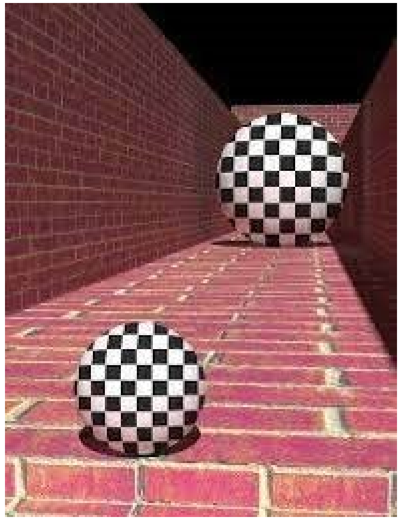


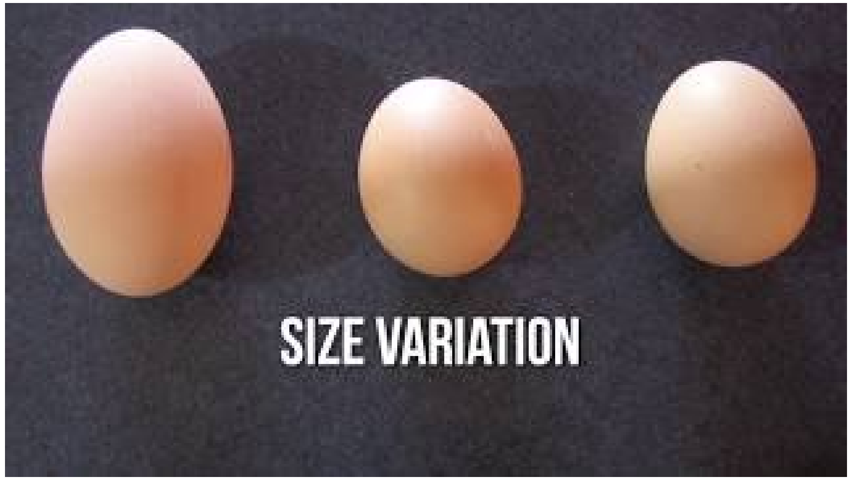





What Computer Vision is hard?

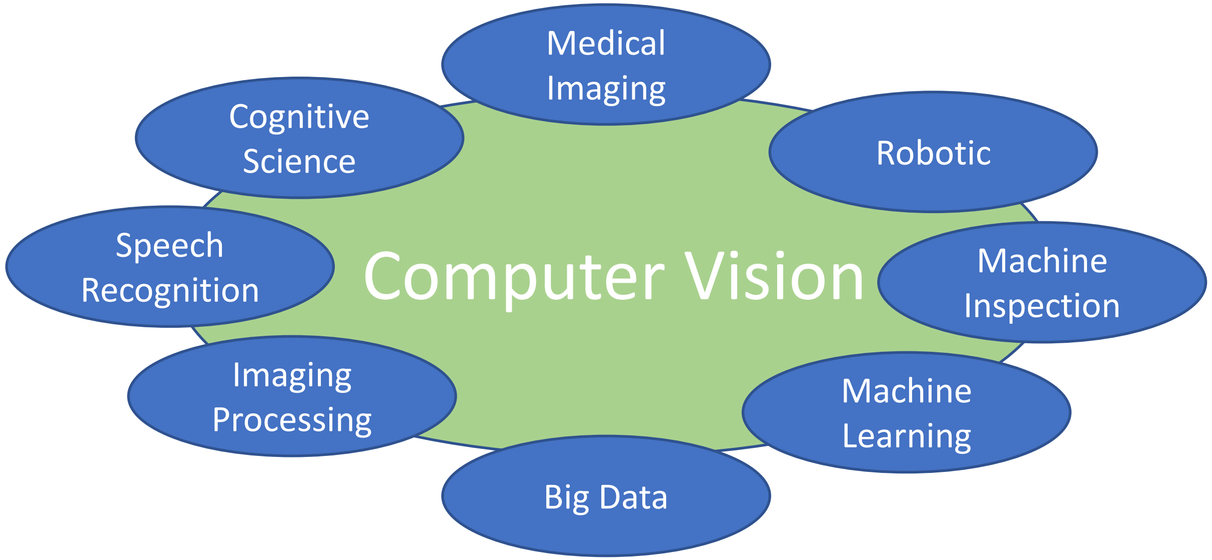
Size Variation





Why Computer Vision Does Matter?

- With its varied useful applications across industries, computer vision has shown its potential to revolutionize almost every sector.



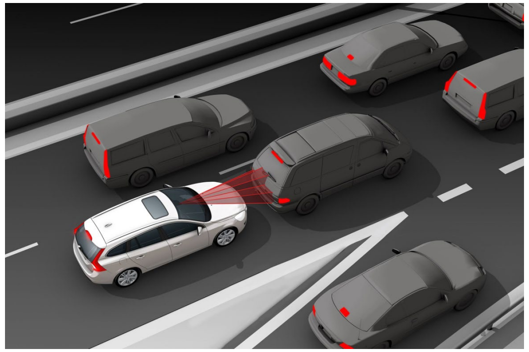
- Forecasts show that computer vision software will show a tremendous revenue growth to USD 26 billion by 2025

Source: https://smebook.eu/knowledge-base/computer-vision/why-computer-vision-is-important/

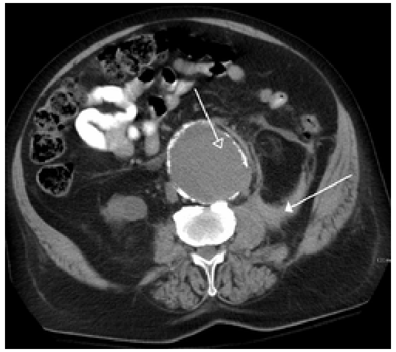
Why computer vision does matter?

Measurement

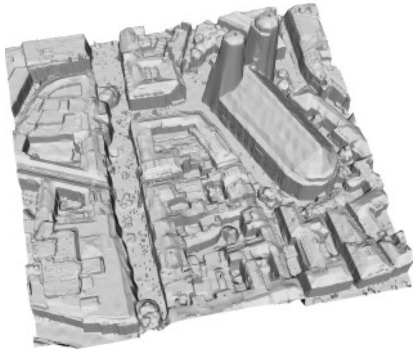
Security



Auto Driving



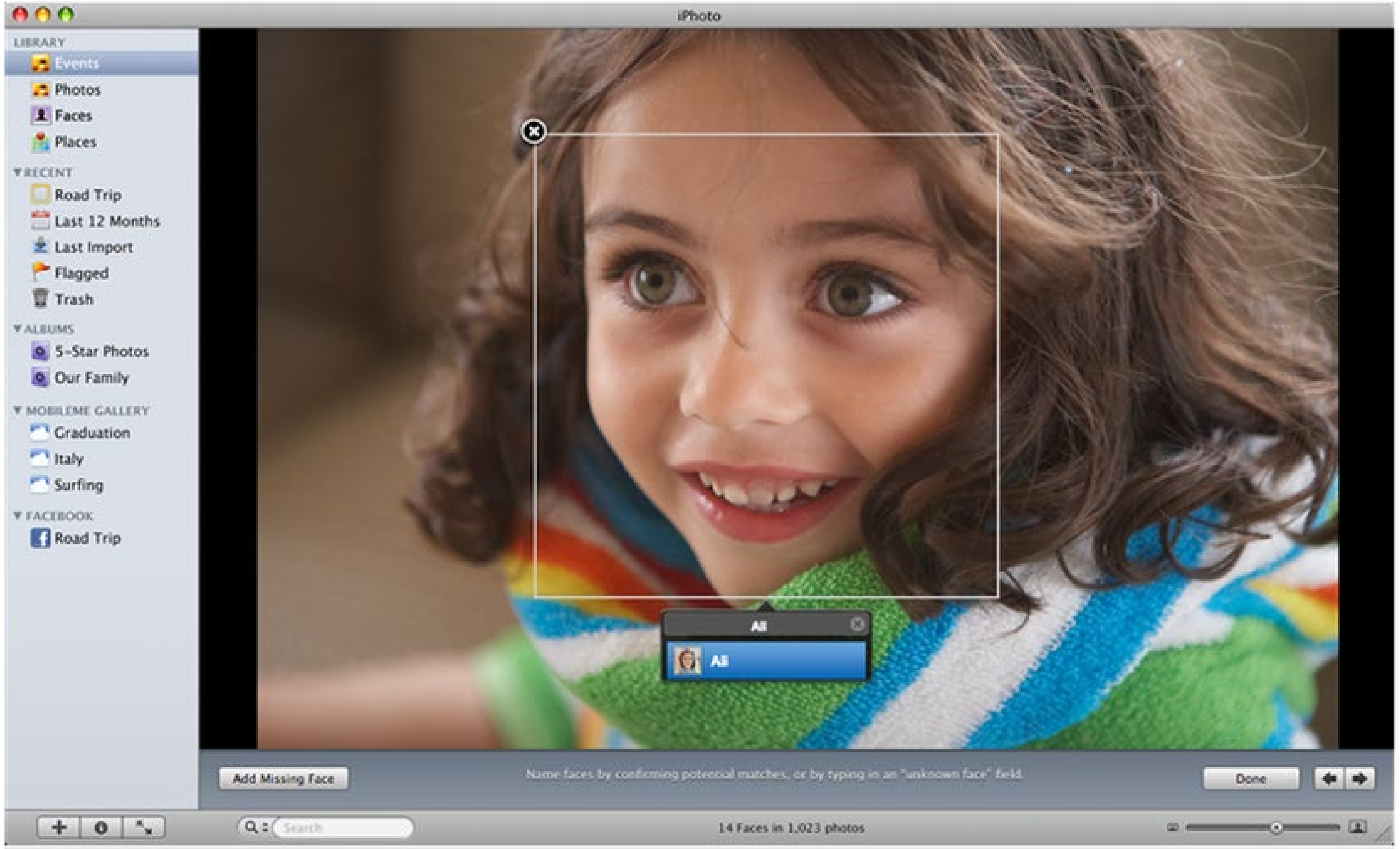
Health



3D Mapping

Entertainment

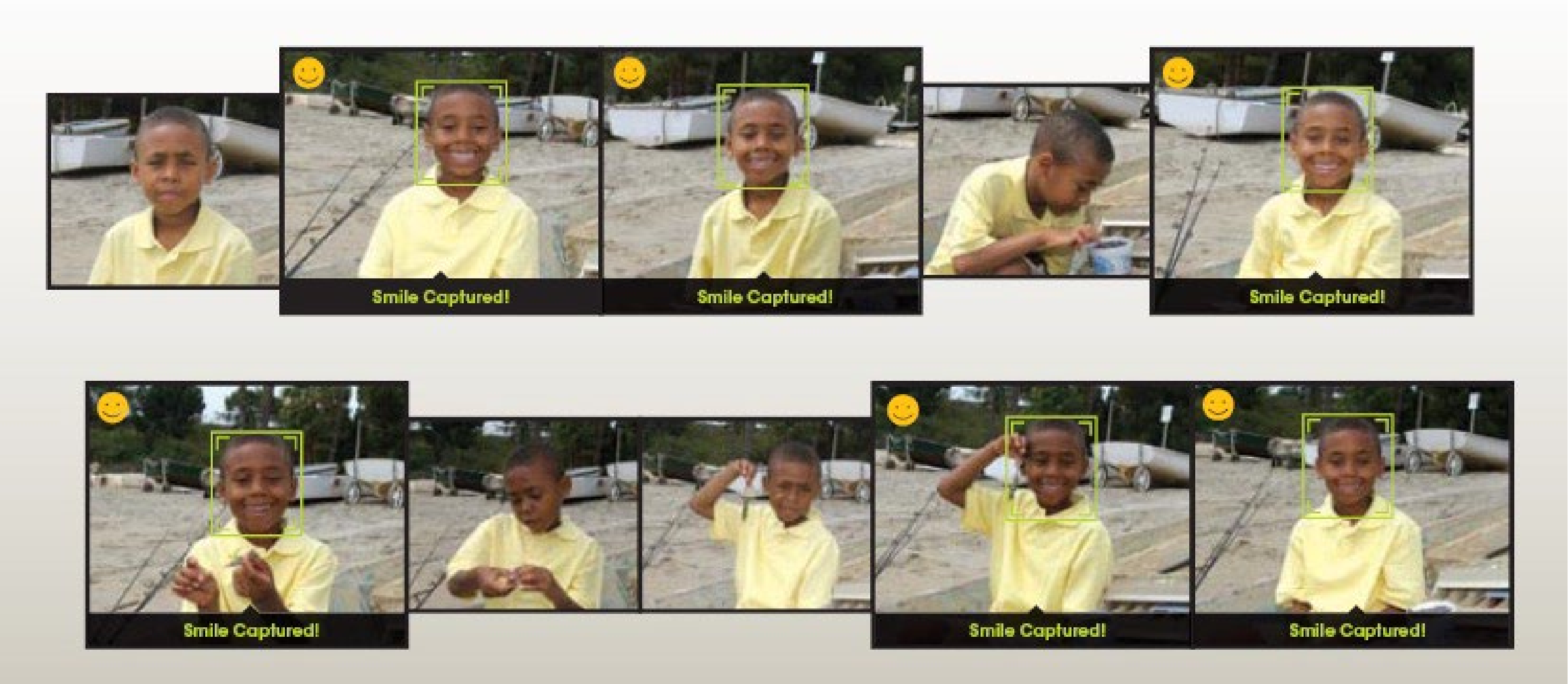
Face Recognition



http://www.apple.com/ilife/iphoto

Simile detection

- The Smile shutter flow
  - The camera can automatically capture every smile



Source: S.Seitz

Biometrics

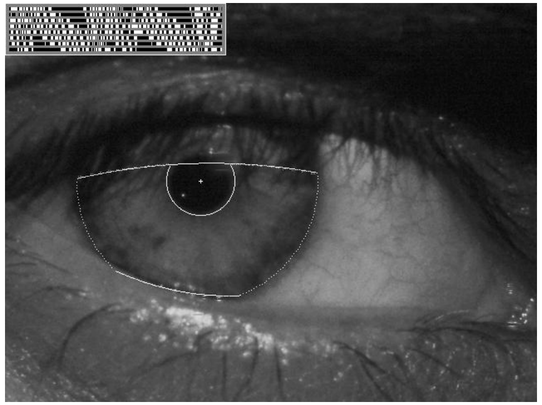
Fingerprint for logging security

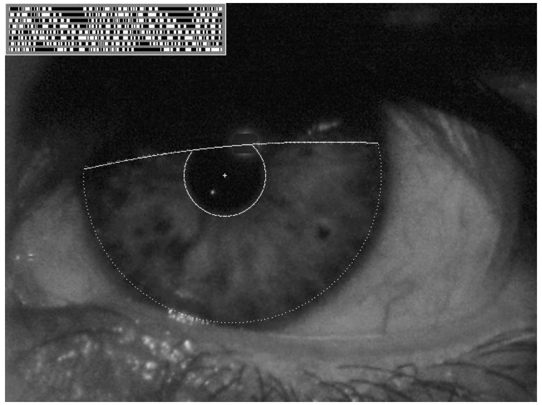
https://www.apple.com/hk/iphone-xs/face-id/

Biometrics

How the Afghan Girl was Identified by Her Iris Patterns

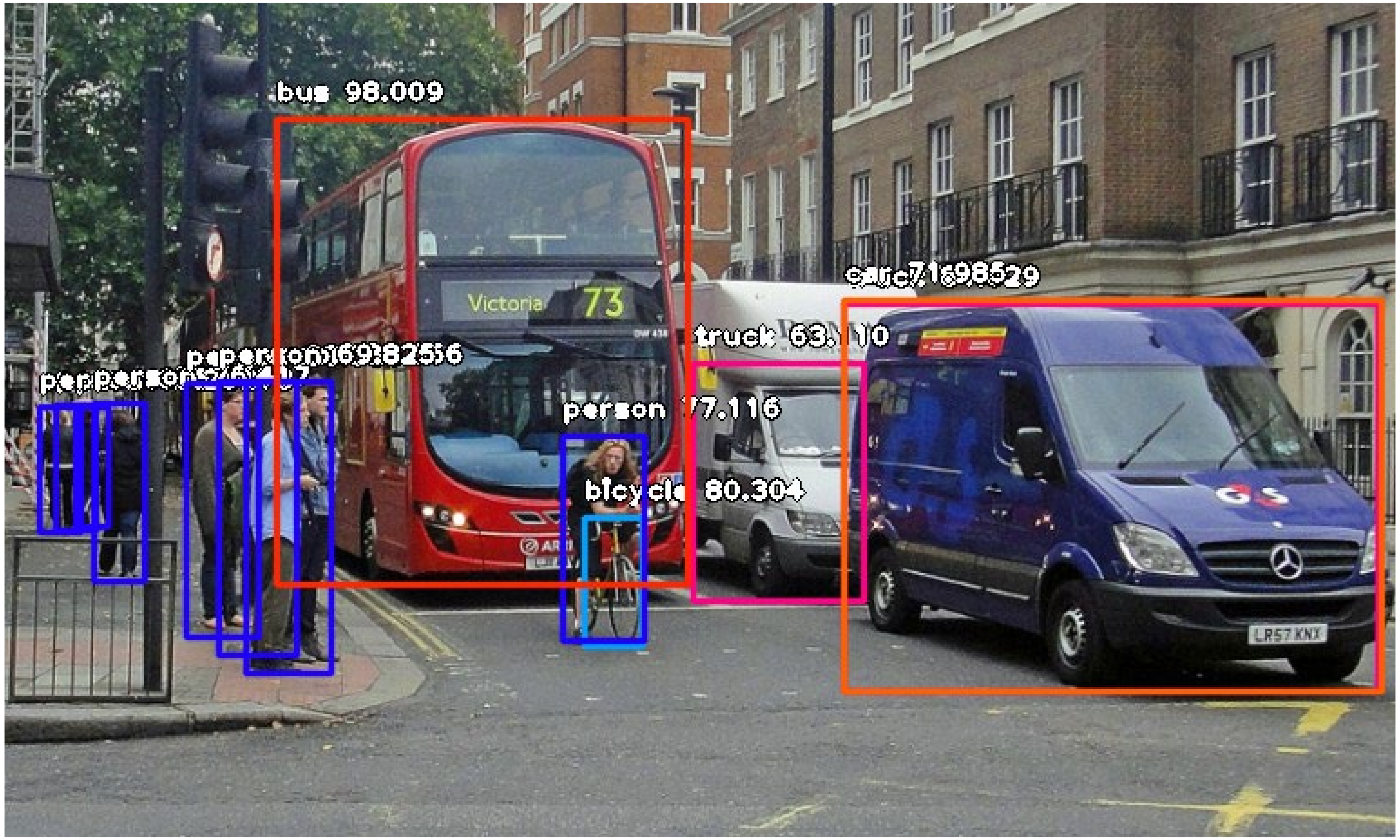
https://www.cl.cam.ac.uk/~jgd1000/afghan.html





https://www.youtube.com/watch?v=cK6EnFu3NHc

Computer Vision for Semantic Information Extraction

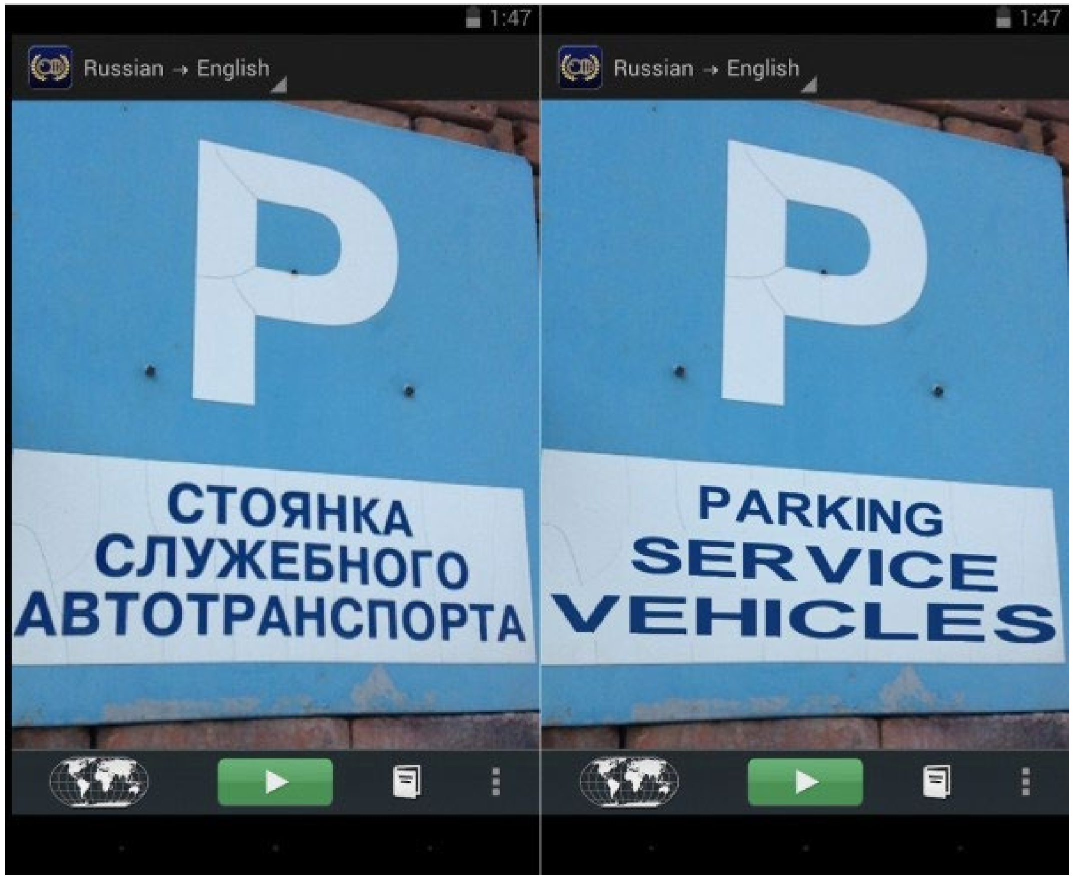


https://towardsdatascience.com/object-detection-with-10-lines-of-code-d6cb4d86f606

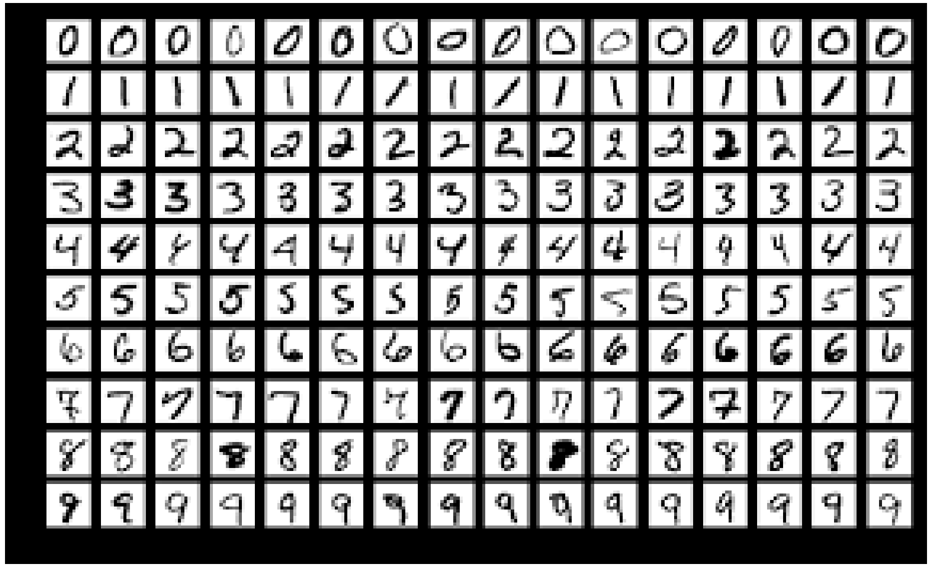
Computer Vision for Motion Capture

Source: https://imerit.net/computer-vision-revolutionizes-baseball-game-motion-capture-technology/

Optical Character Recognition (OCR)

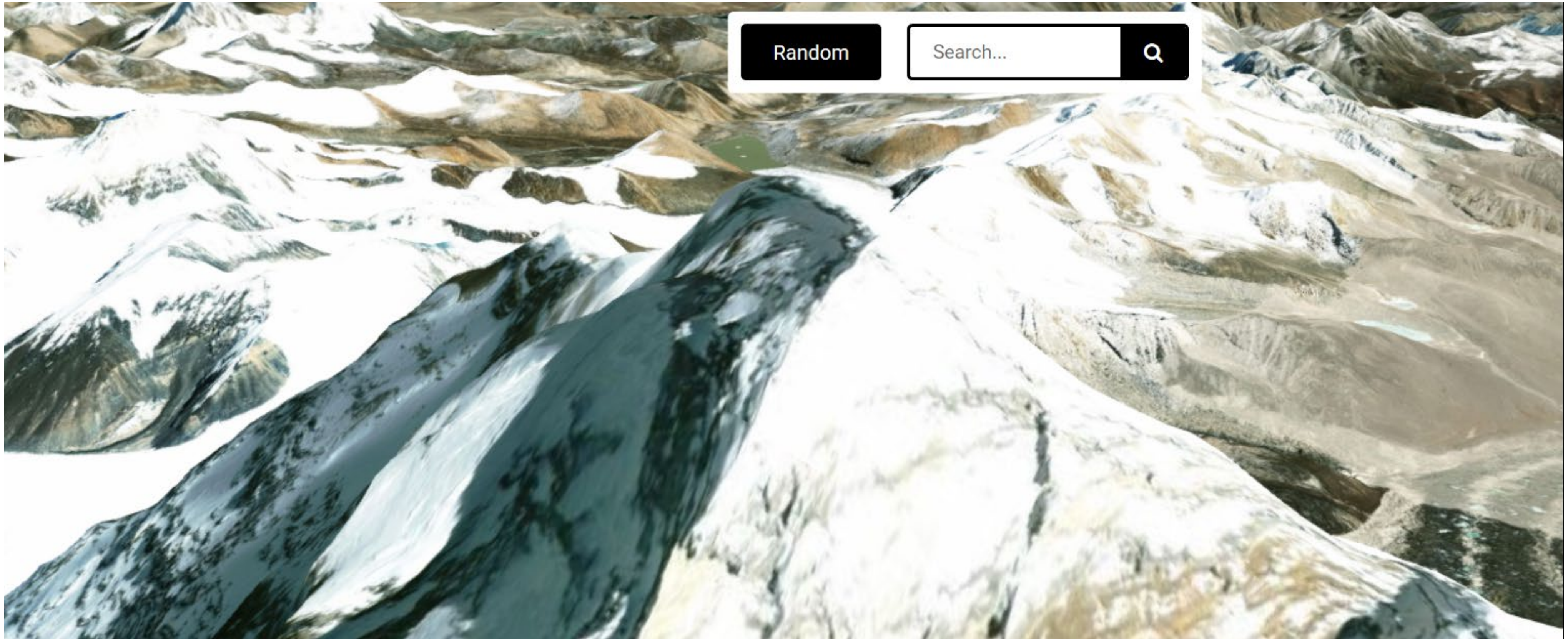


https://betanews.com/2014/05/20/google-buys-translator-app-word-lens-and-now-its-free/



MNIST Data Set for OCR

3D Maps



http://earth3dmap.com

Toys and Robots

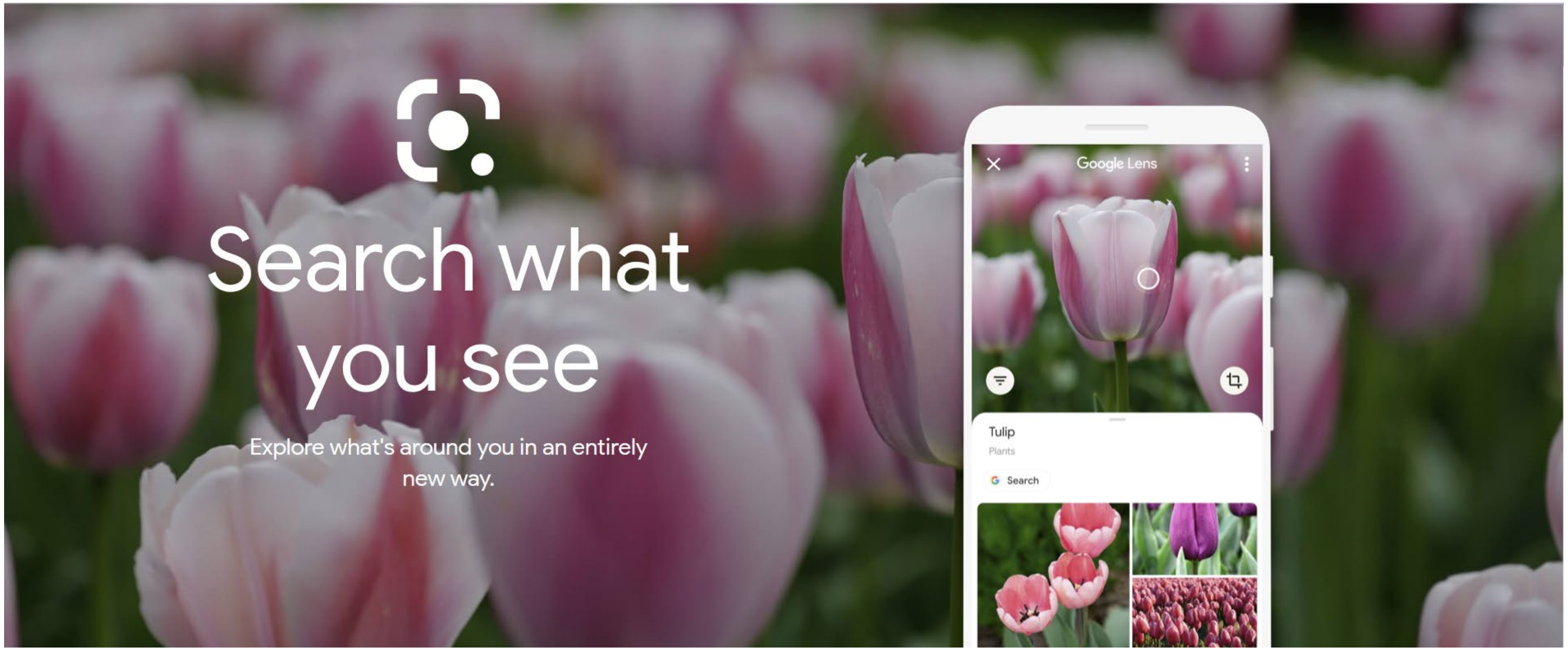
Sony Aibo

Industrial robot

AGV

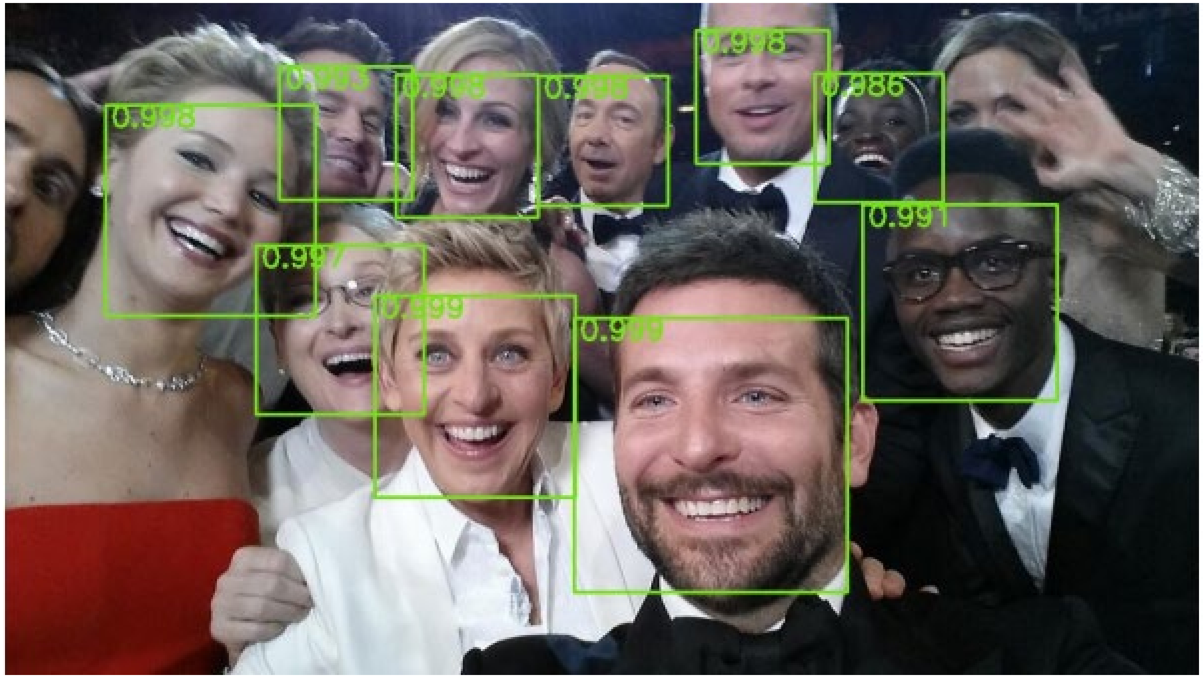
Household robot

Object Recognition



https://lens.google.com/

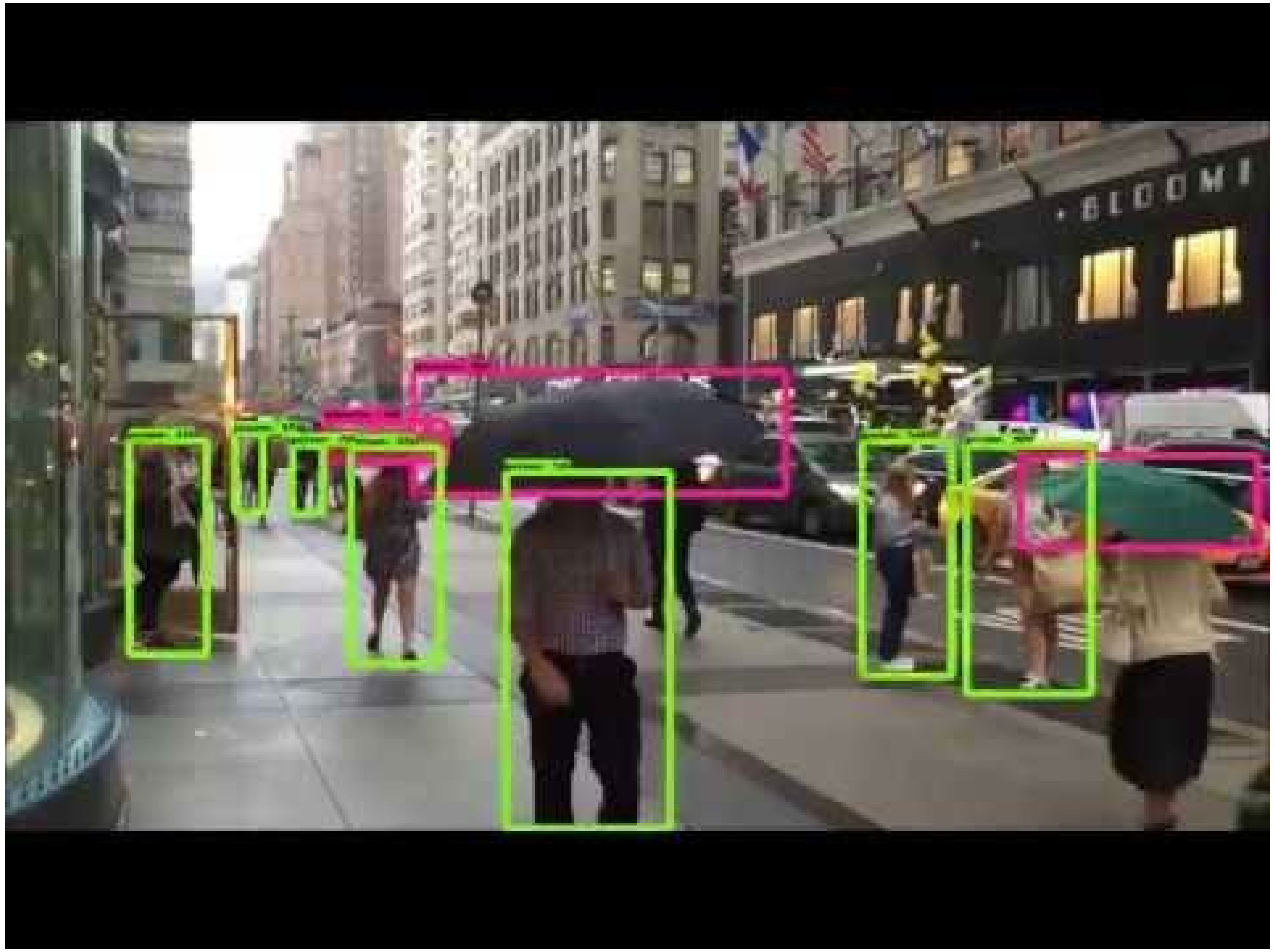
Face Detection and Smile Detection



https://www.technologyreview.com/s/535201/the-face-detection-algorithm-set-to-revolutionize-image-search/

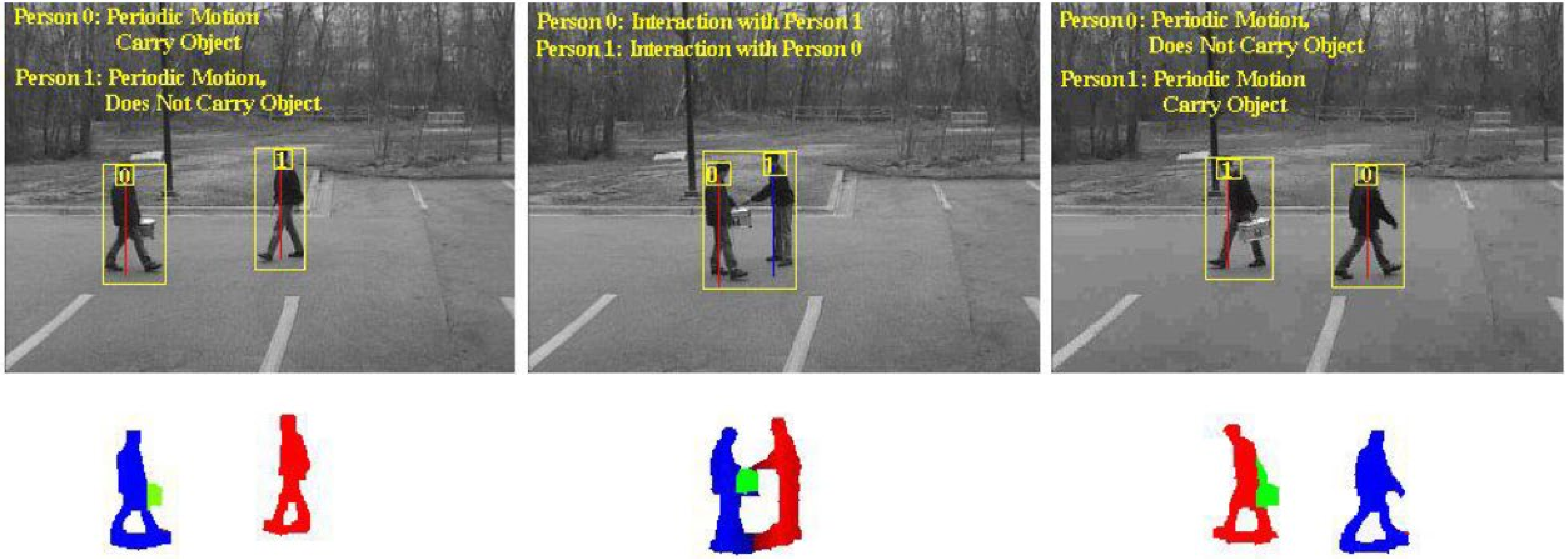
Simile Shutter from Sony

Video Surveillance and Tracking



Slide source: R. Dockter

Human Activity Recognition



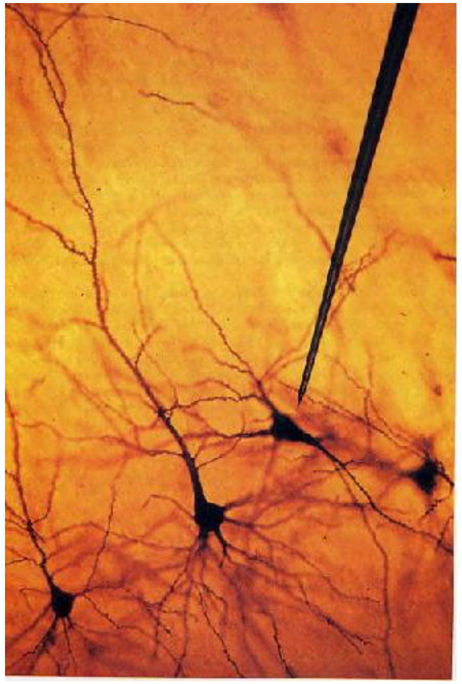
Slide source: R. Dockter

Autonomous Driving

Telsa Autonomous Driving System

Medical Imaging

- 1981: Nobel Prize in medicine



Hubel &Wiesel

Vision for robotics, space exploration

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 tasks
  - Panorama stitching
  - 3D terrain modeling
  - Obstacle detection, position tacking

Source:+S.Seitz

Can computers match human perception?

- Not yet!
- Computer vision is still no match for human perception
- But computer are catching up in certain areas.
- Classifying the ImageNet Database:
  - 1.2 million images, 1000 categories
  - Human: ~5.1% Error Rate
  - Inception-V3 Convolution Neural Network: 3.46% Error Rate

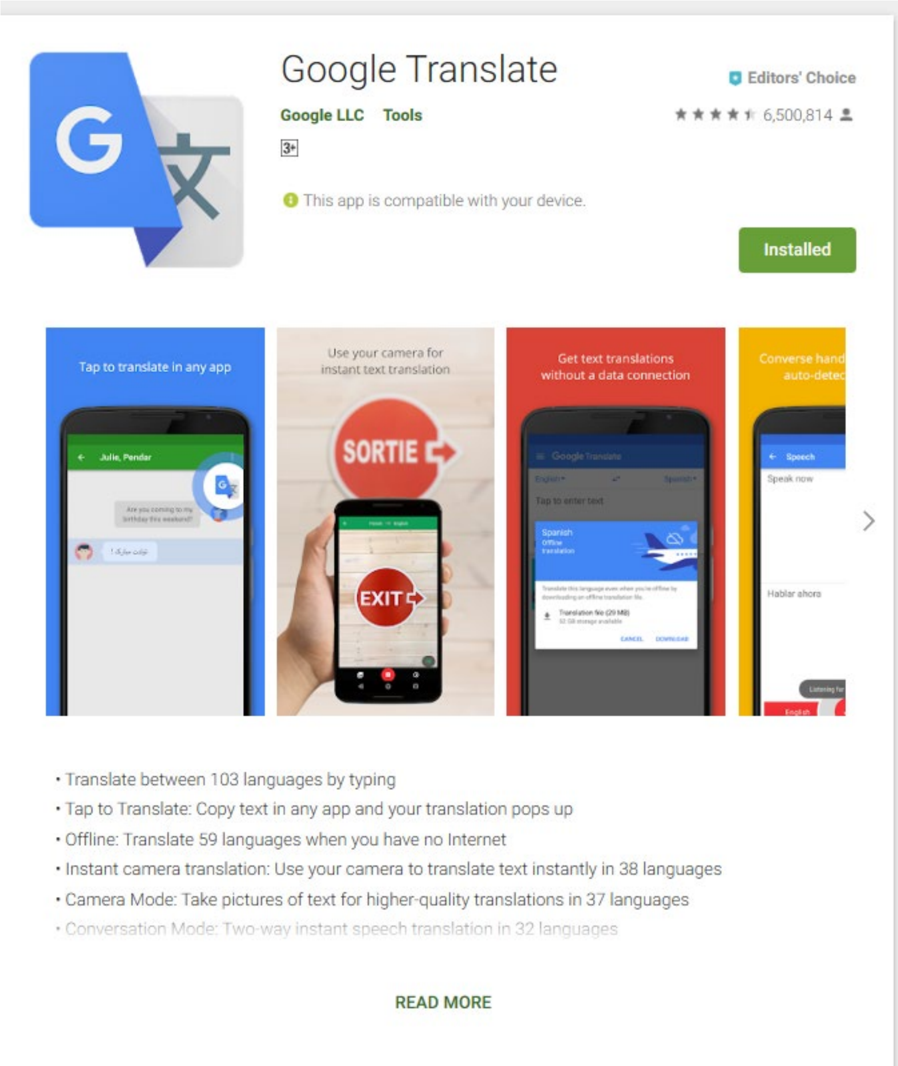
https://karpathy.github.io/2014/09/02/what-i-learned-from-competing-against-a-convnet-on-imagenet/ Slide: R. Doctker

In Summary

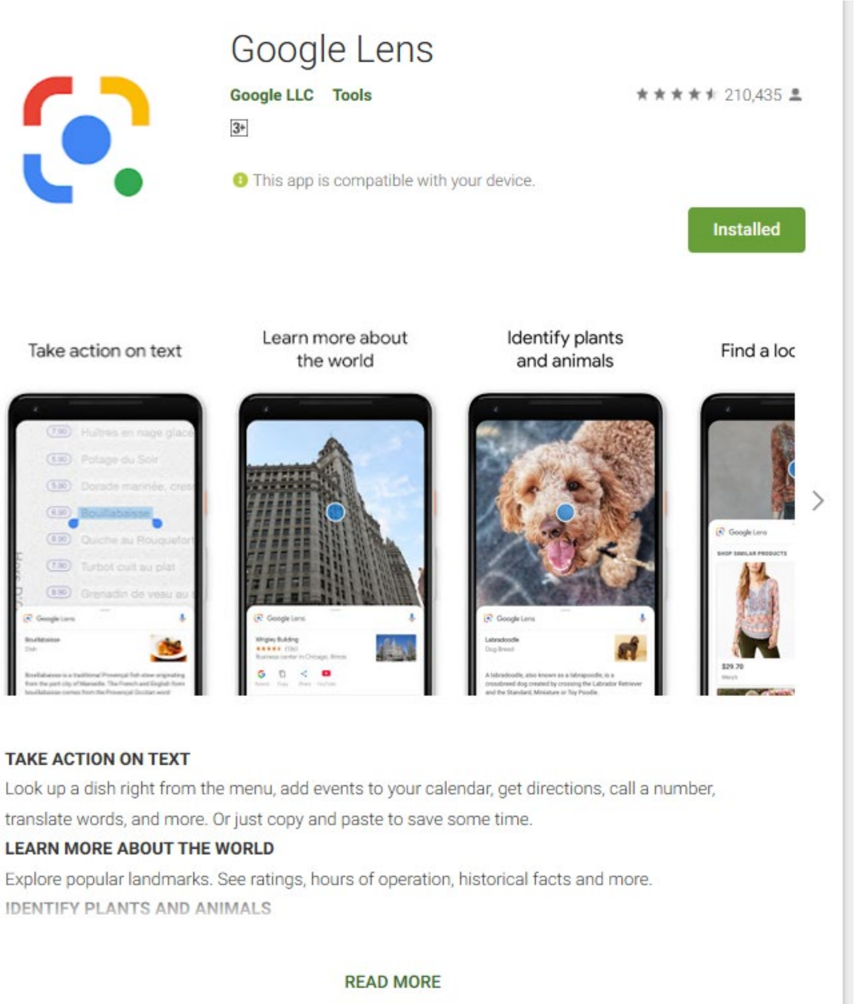
- Computer Vision is a challenging topic
- It has many real-life applications
- It has tremendous progress in last decades
- Much work to be done!

Vision Apps

Google Translate



Google Lens


Google Lens
Google LLC
Tools
210,435
3+
This app is compatible with your device.
Installed
Take action on text
Learn more about the world
Identify plants and animals
Find a loc
TAKE ACTION ON TEXT
Look up a dish right from the menu, add events to your calendar, get directions, call a number, translate words, and more. Or just copy and paste to save some time.
LEARN MORE ABOUT THE WORLD
Explore popular landmarks. See ratings, hours of operation, historical facts and more.
IDENTIFY PLANTS AND ANIMALS
READ MORE

Cam Scanner


CamScanner for high work and learning efficiency
Scan, save, archive, upload, search, and easily acquire and manage information anytime, anywhere
Mobile scanner
Say goodbye to the cumbersome operation of the scanner, a mobile phone can do it all anytime, anywhere
Auto trimming, image beauty
Snap Docs with mobile phone, auto remove messy background, and generate HD JPEG images or PDF files
Wireless printing, global fax
Allow wireless printing, and fax to more than 30 countries in the world
CS
CAMSCANNER
Securely Sync across Platforms
OCR recognition, Image to text
Long text in image can be converted to text instantaneously, and 40 languages (including Chinese, English, Japanese and Korean) can be recognized
Multiple devices, sync any time
File synchronization and backup for reading work anytime and anywhere
Edit files in PDF freely
Online PDF toolkit supports free merging, splitting, and compression of files, as well as quick addition of electronic signatures and watermarks

Any more?

Course Outline

- Introduction
- Pin-hole Camera & Image filtering
- Edge and Interest point detection
- Segmentation
- Optical flow and tracking
- Projective Geometry in 2D/3D
- Camera Calibration
- Multi-View Geometry
- Shape from Stereo
- Structure from Motion

Class Schedule

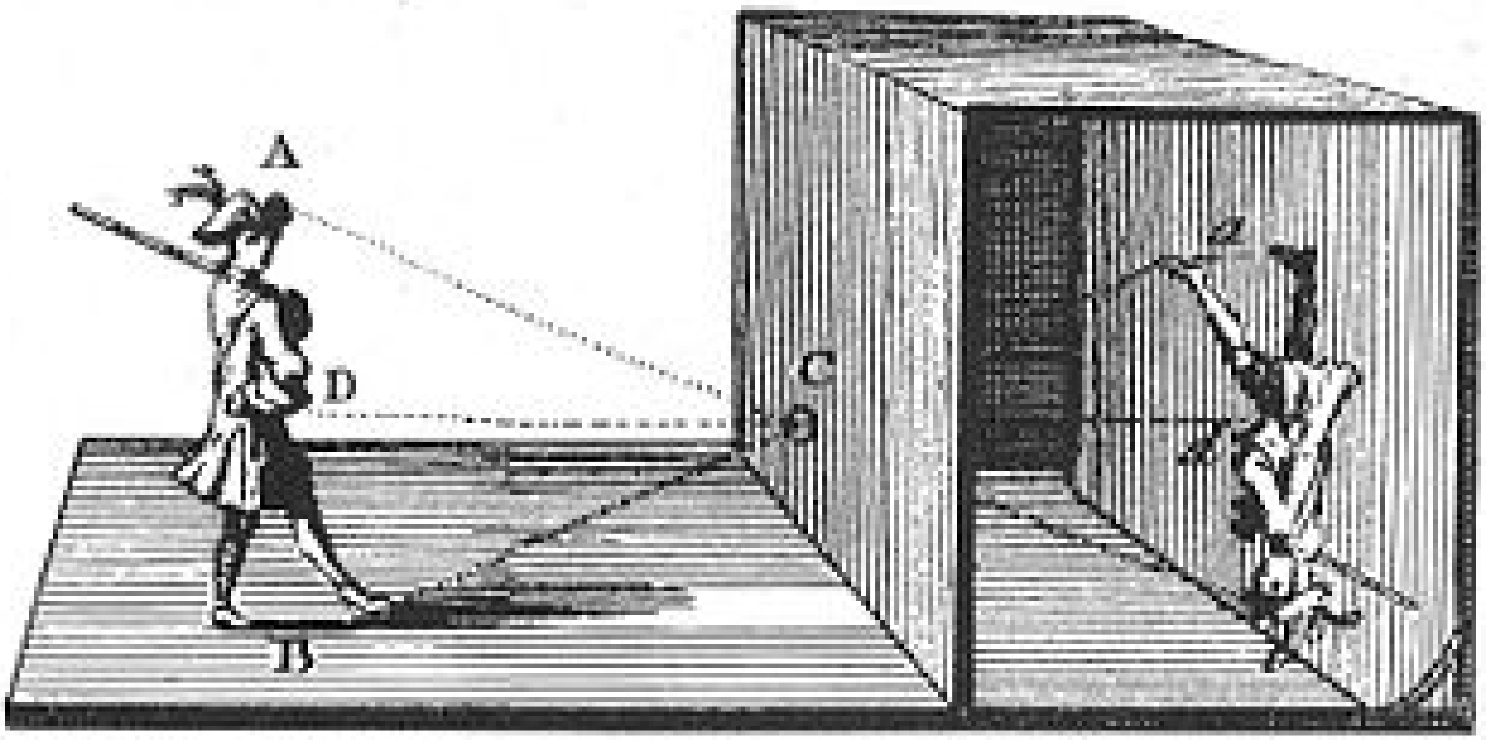
Class	Date	Topic
1	Friday, 10 September 2021	Course Introduction
2	Friday, 17 September 2021	Camera Model and image formation
3	Friday, 24 September 2021	Image filtering and template matching
	Friday, 1 October 2021	National Date Holiday
4	Friday, 8 October 2021	Edge and Interest point detection
5	Friday, 15 October 2021	Feature Descriptor(SIFT/SURF,etc)
6	Friday, 22 October 2021	Image Segementation
7	Friday, 29 October 2021	Optical Flow and Tracking
8	Friday, 5 November 2021	Projective Geometry in 2D
9	Friday, 12 November 2021	Transformation & Projective Geometry in 3D
10	Friday, 19 November 2021	Homography and Image Stitching
11	Friday, 26 November 2021	Camera Parameters and Calibration
12	Friday, 3 December 2021	More on Single view geometry
13	Friday, 10 December 2021	Epipolar Geometry and Stereo Vision
14	Friday, 17 December 2021	More on 3D Reconstruction

Subject to Changes

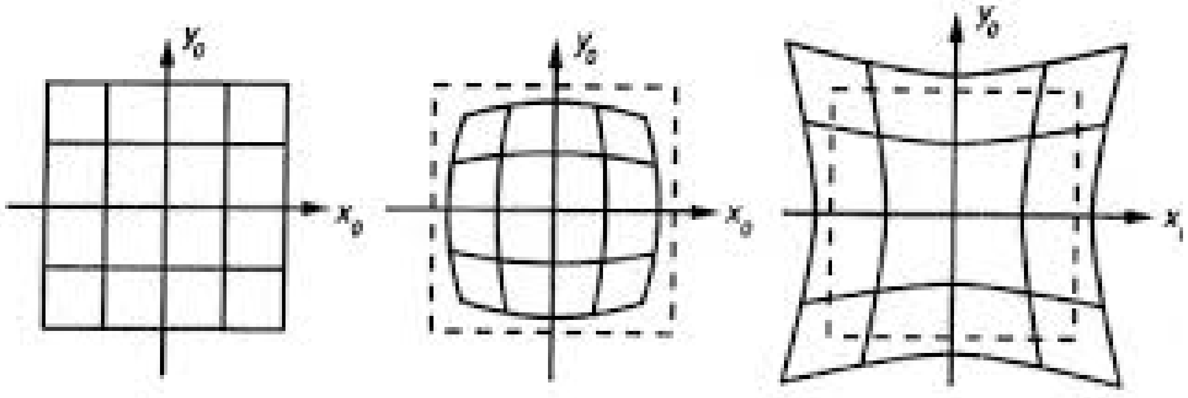
Course Content

Image Formation

- **Camera obscura** means dark room *camera* chamber also refer to a Pinhole Camera



Effect of camera



No Distortion

Barrel Distortion

Pincushion Distortion

Homogenous coordinates

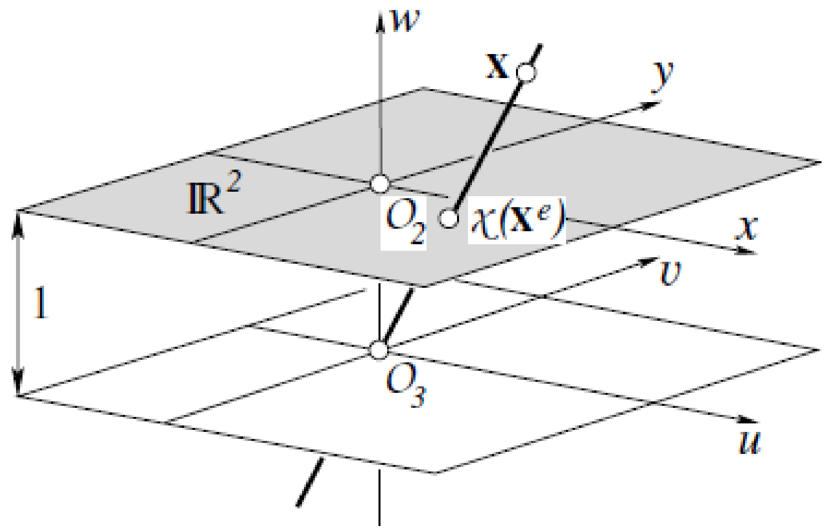
- Homogeneous Coordinates of a point χ in the plane in $\mathbb{R}^2$ is a 3-dimensional vector

$$\chi: \qquad \mathbf{x} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \text{ where}$$

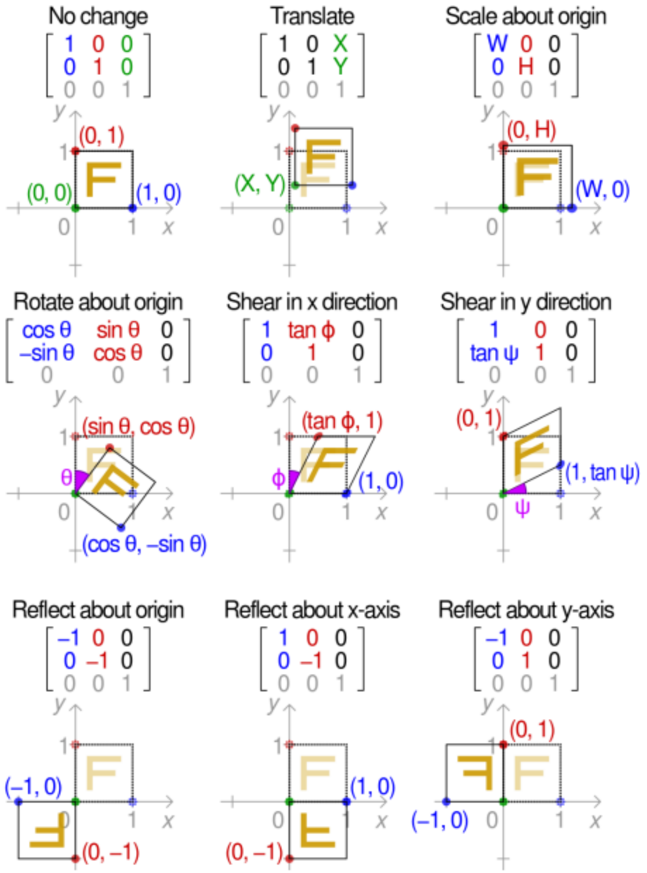
$$u^2 + v^2 + w^2 \neq 0$$

- In Euclidian coordinates

$$\chi: \qquad \boldsymbol{x} = \begin{bmatrix} u/w \\ v/w \end{bmatrix} \text{ where } w \neq 0$$



Transformation with homogenous coordinates



$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Translate

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Scale

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

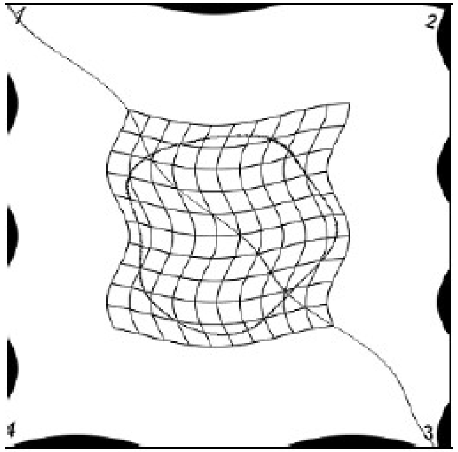
2D *in-plane* rotation

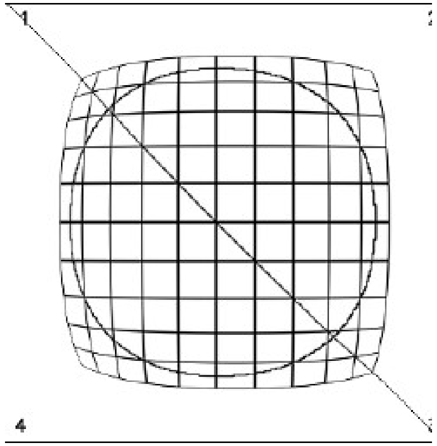
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & sh_x & 0 \\ sh_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

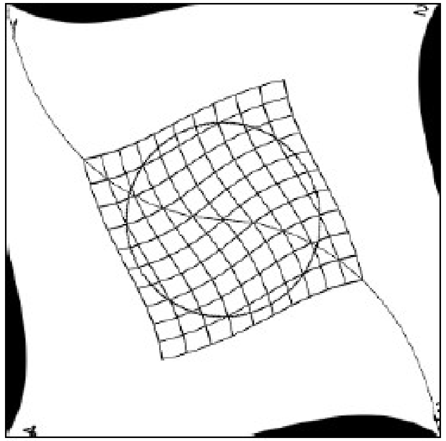
Shear

Basic Image Transformation

Other transformation


(b)


(c)


(a)


(e)


(f)


(d)

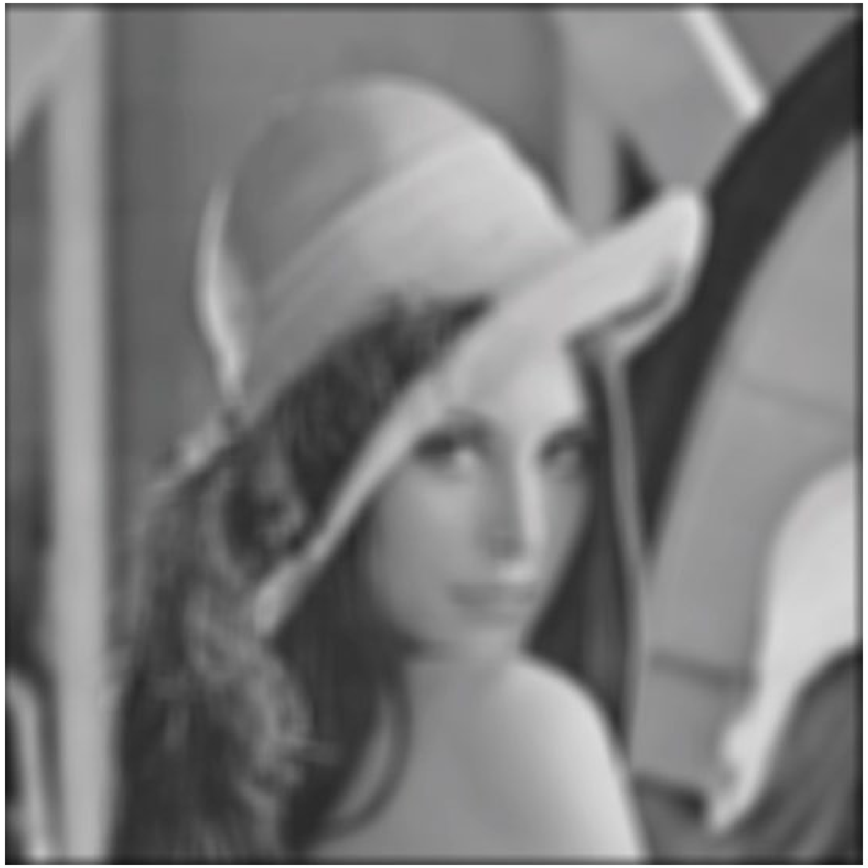
Image Credit: E. Agu. WPI

Image Warping

Source: http://medium .com

Image Filtering (Smoothing)

Original



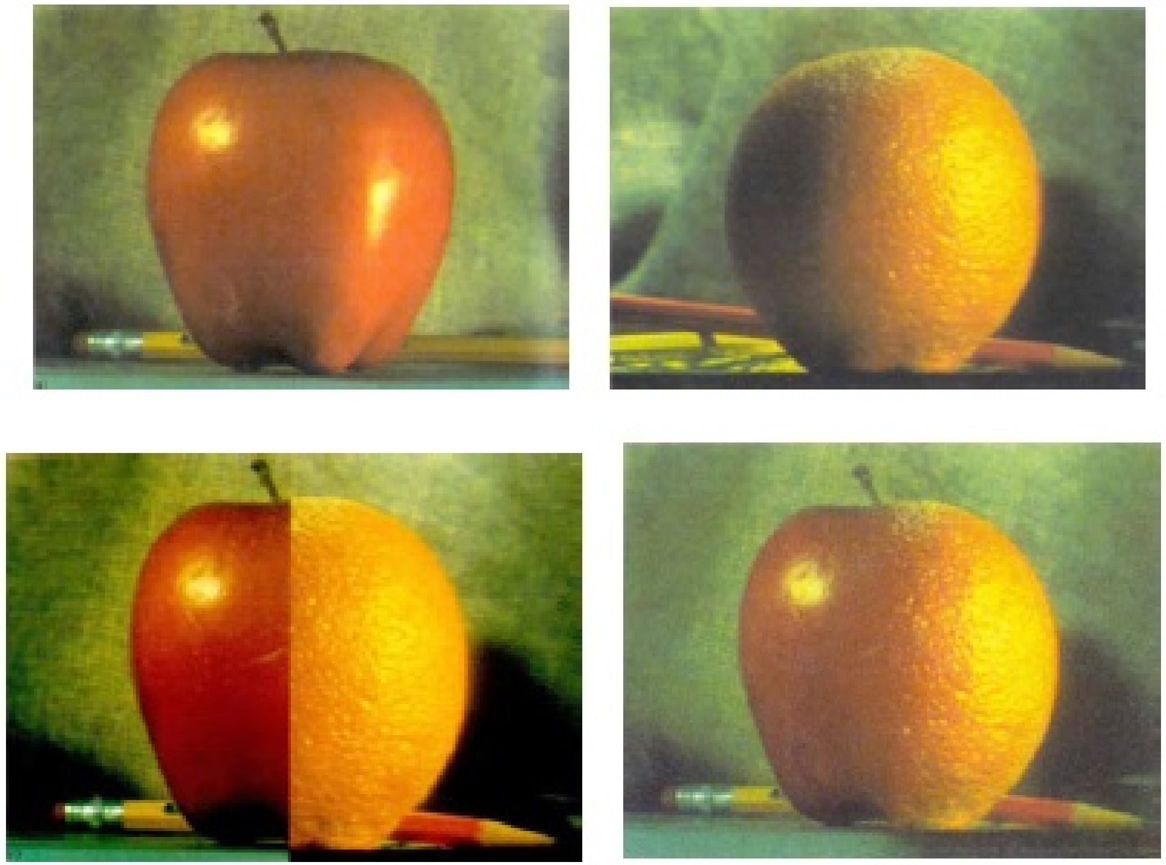
Sharpened

Image Filtering (Sharpening)

Original

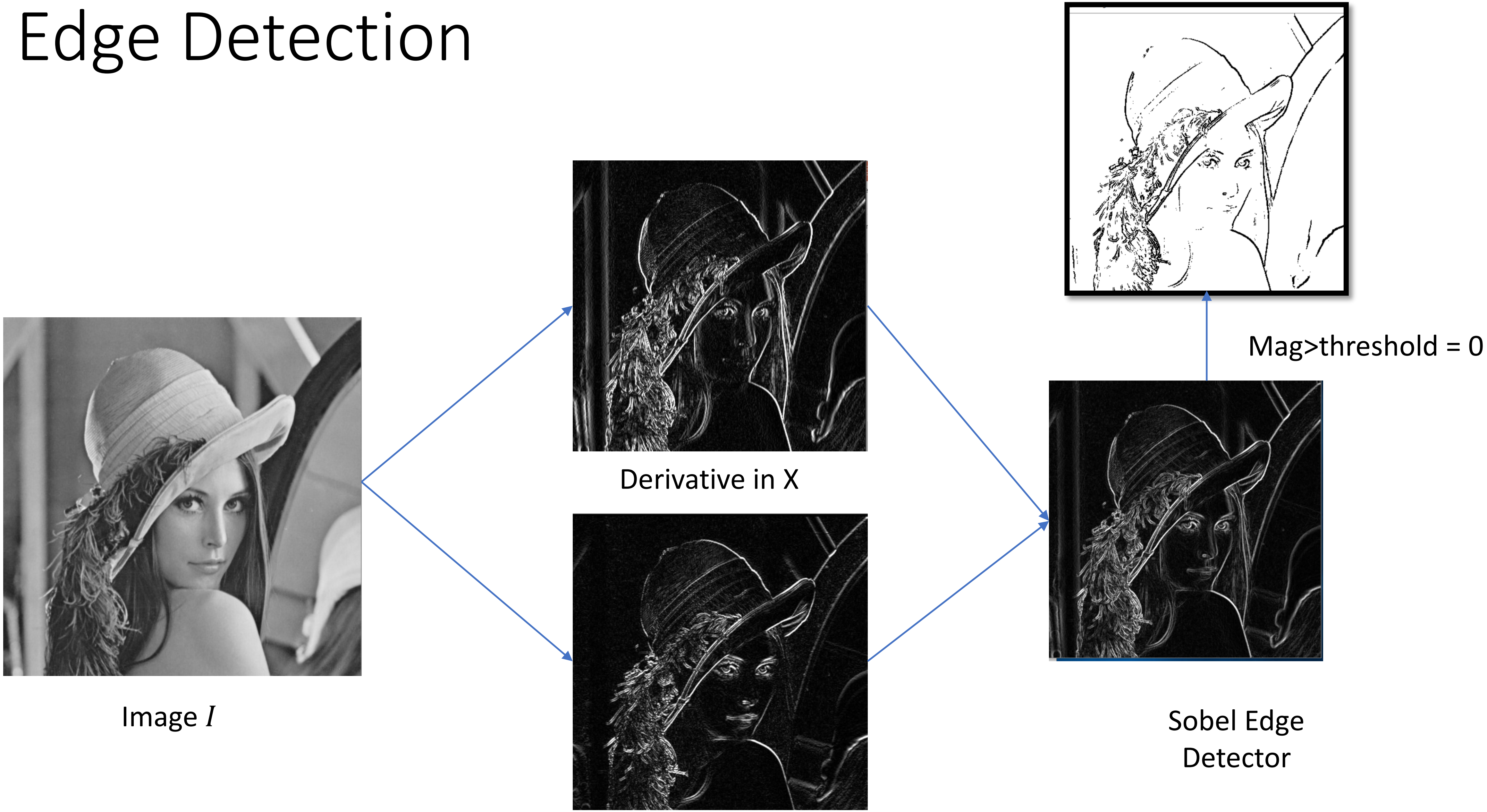
Sharpened

Image Bending



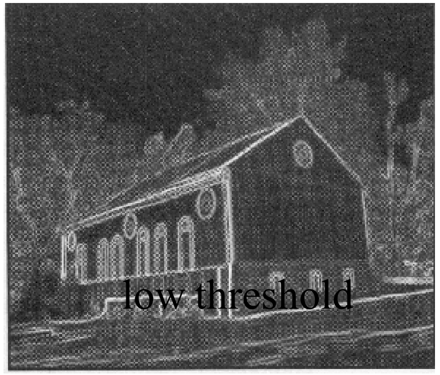
(c)

Edge Detection



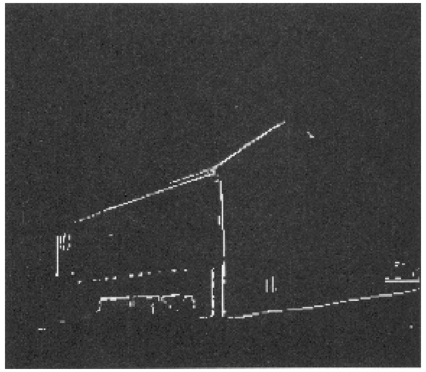
Effect the threshold?

Image



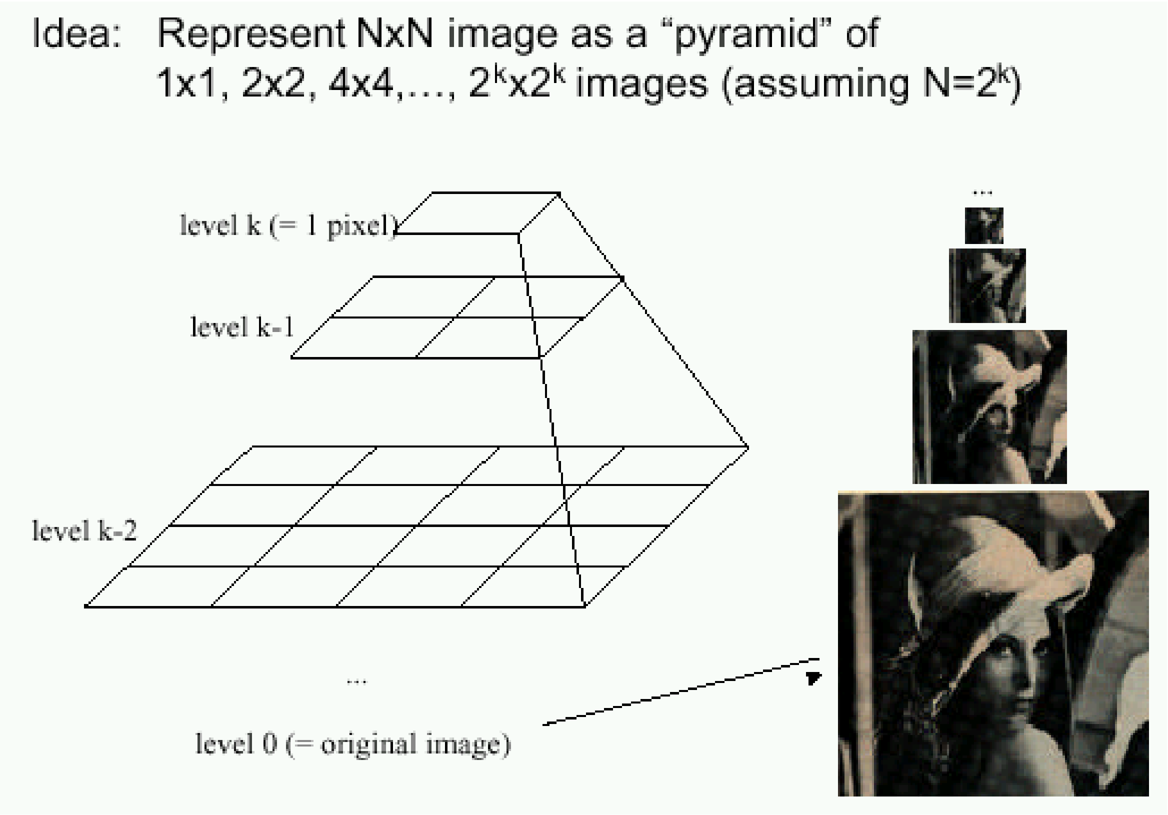
Gradient Mag

Low Threshold



High Threshold

Image Pyramids



- In computer graphics, a *mip map* [Williams, 1983]

Source: S. Seitz

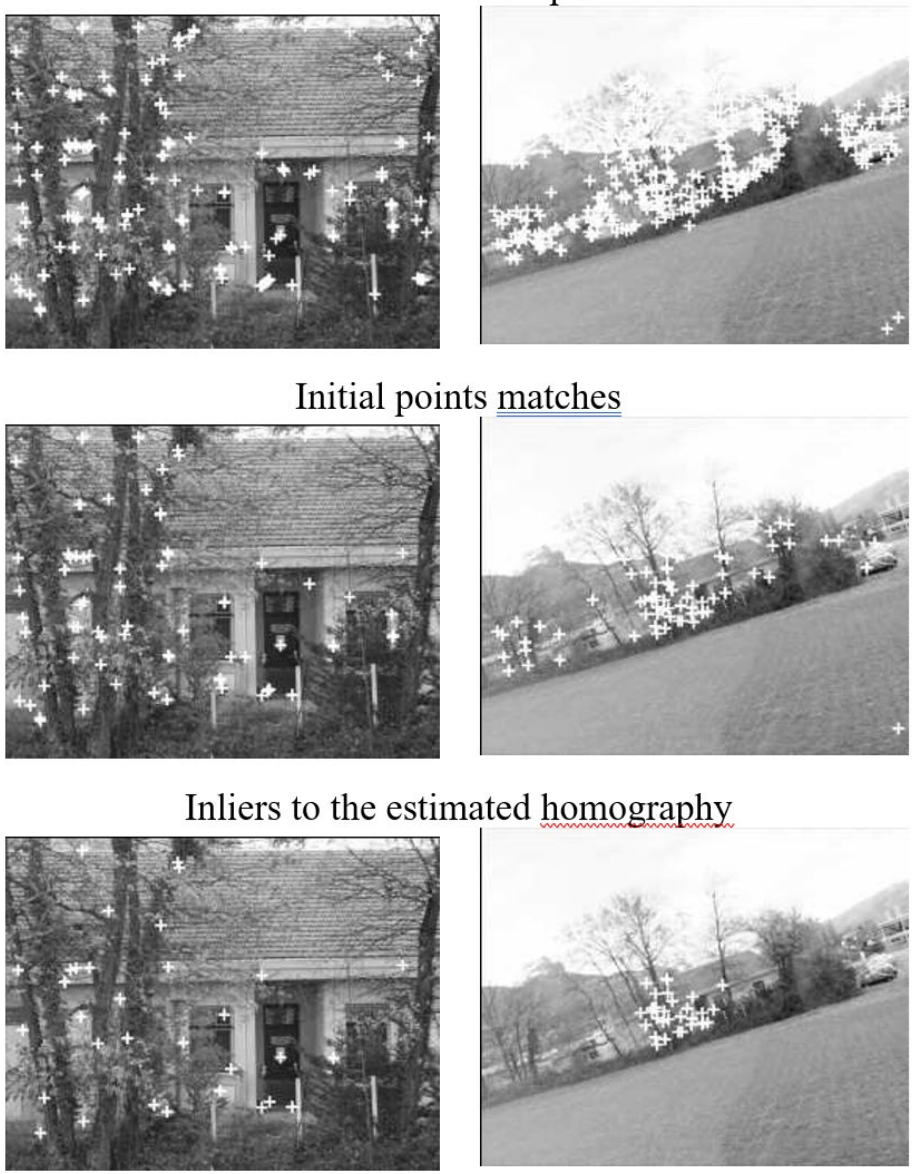
Interest point detection

- What is interest point?
- The use of interest point
    - Autonomous vehicle navigation
    - Image matching and retrieval

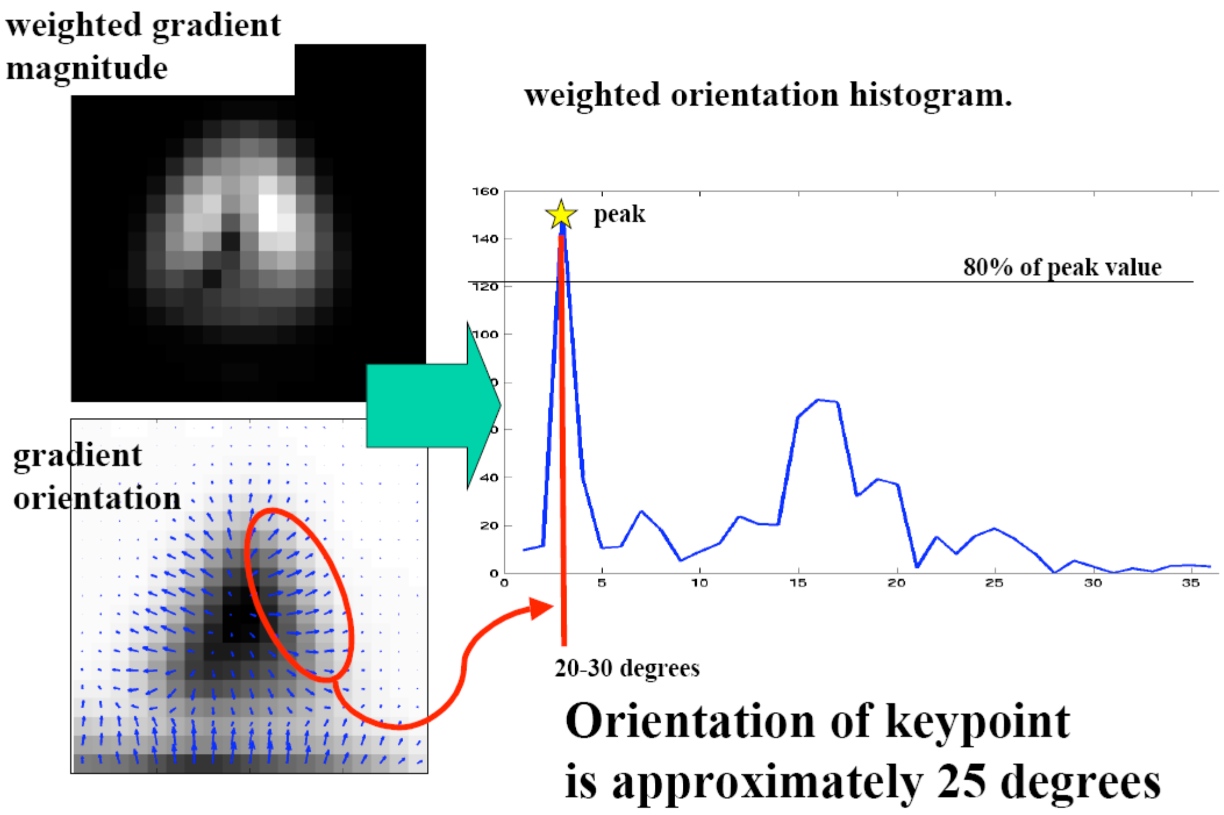
Extracted interest points

Initial points matches

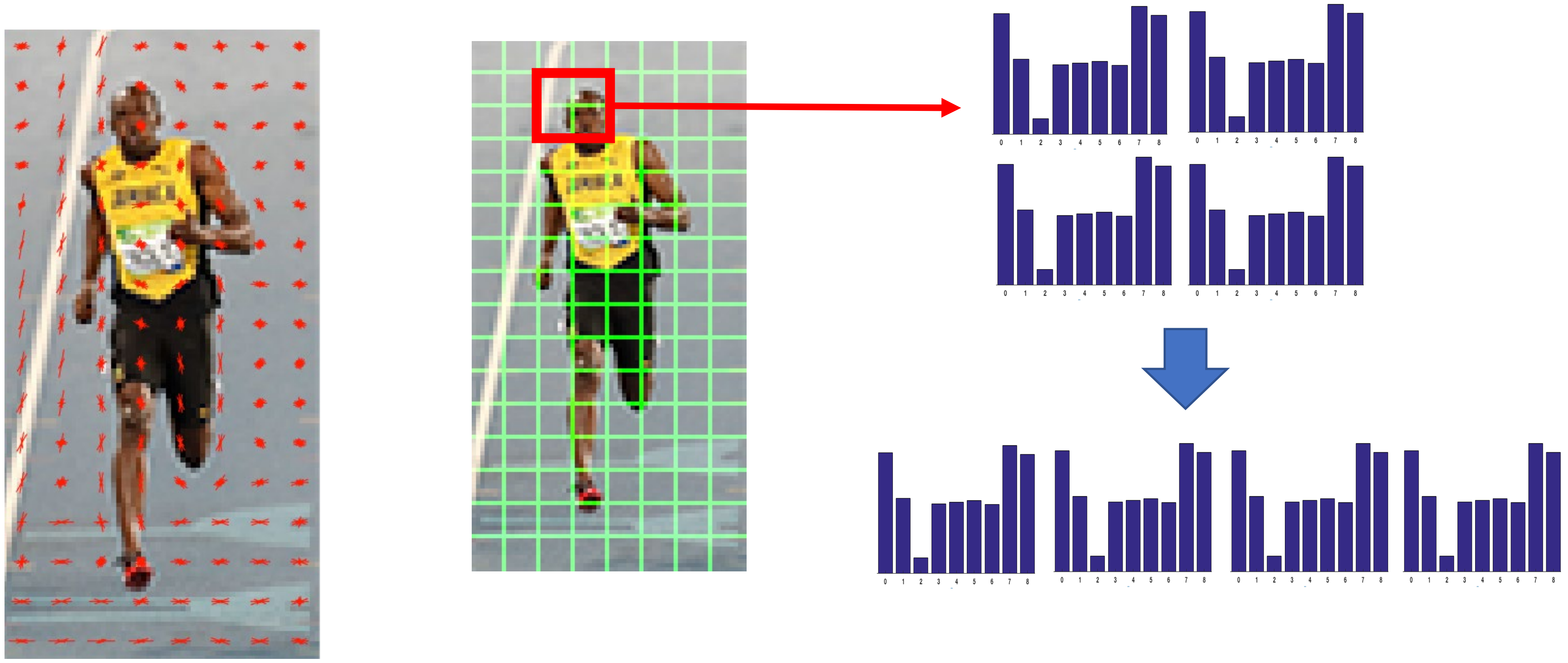
Inliers to the estimated homography



Feature Descriptors (SIFT/SURF/ORB/HOG)



Histogram of Oriented Gradient



Use of Shape Descriptor

e.g. scale,
translation,
rotation

Image matching

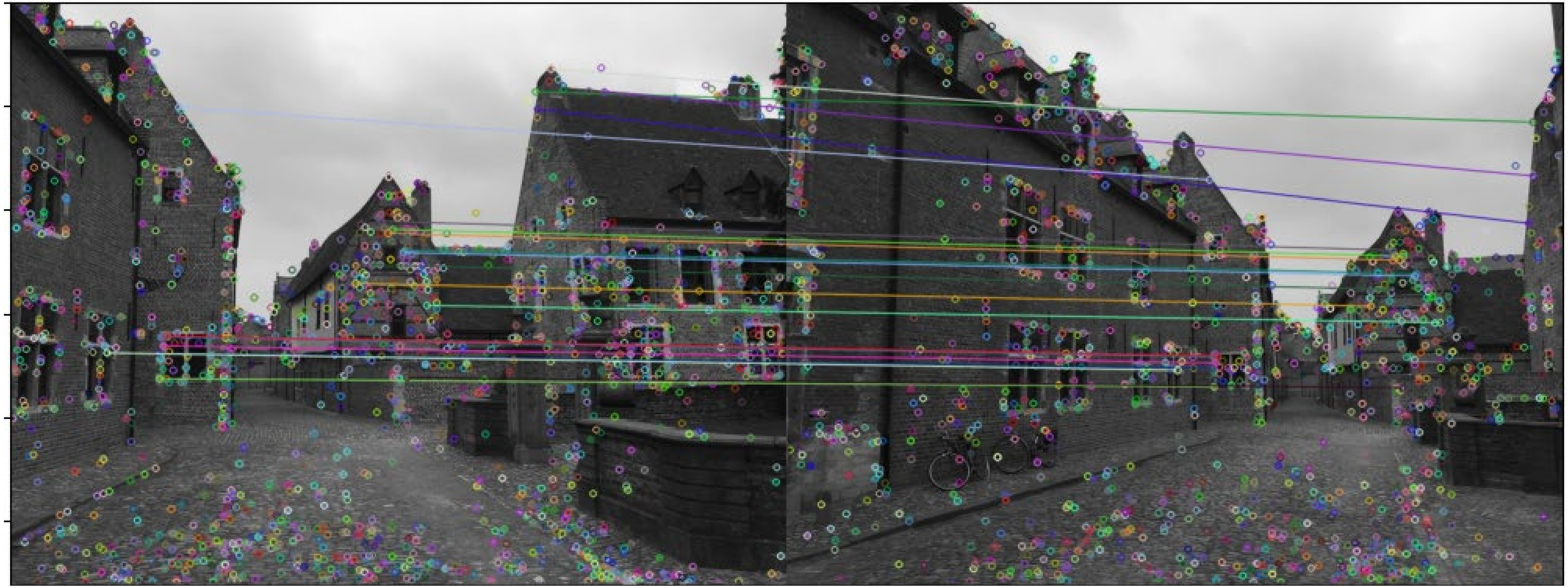
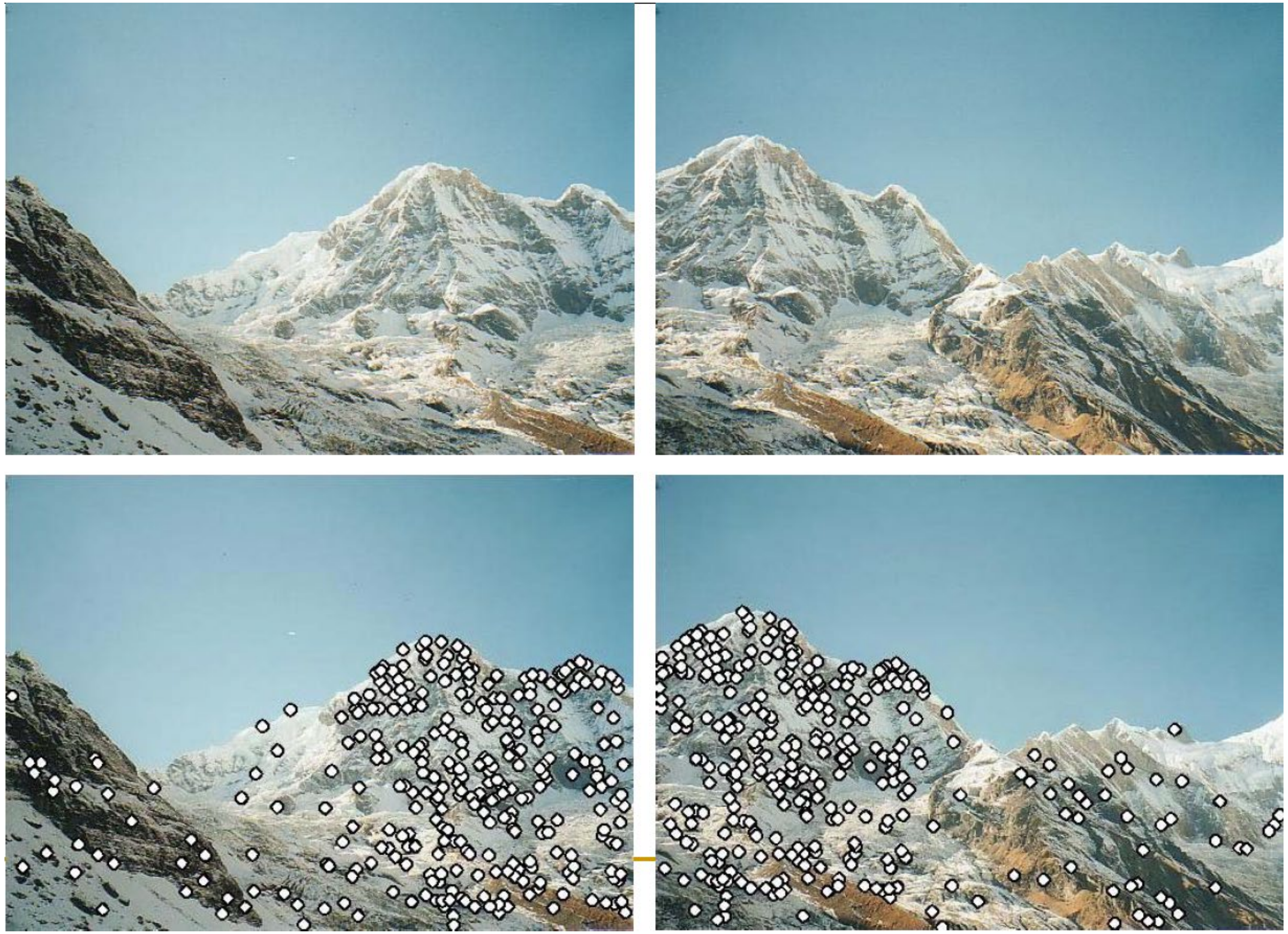


Image Stitching with SIFT and RANSAC



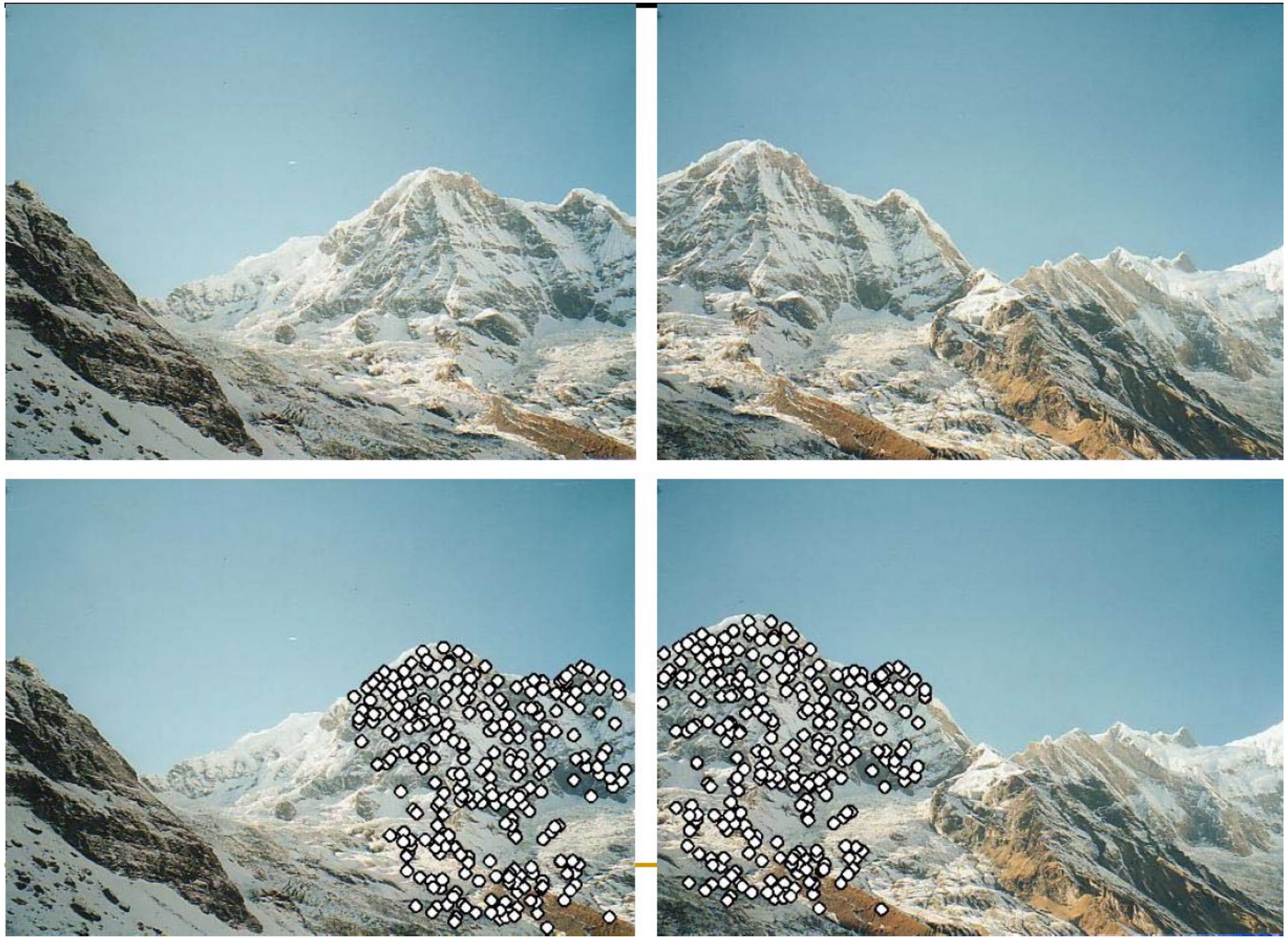
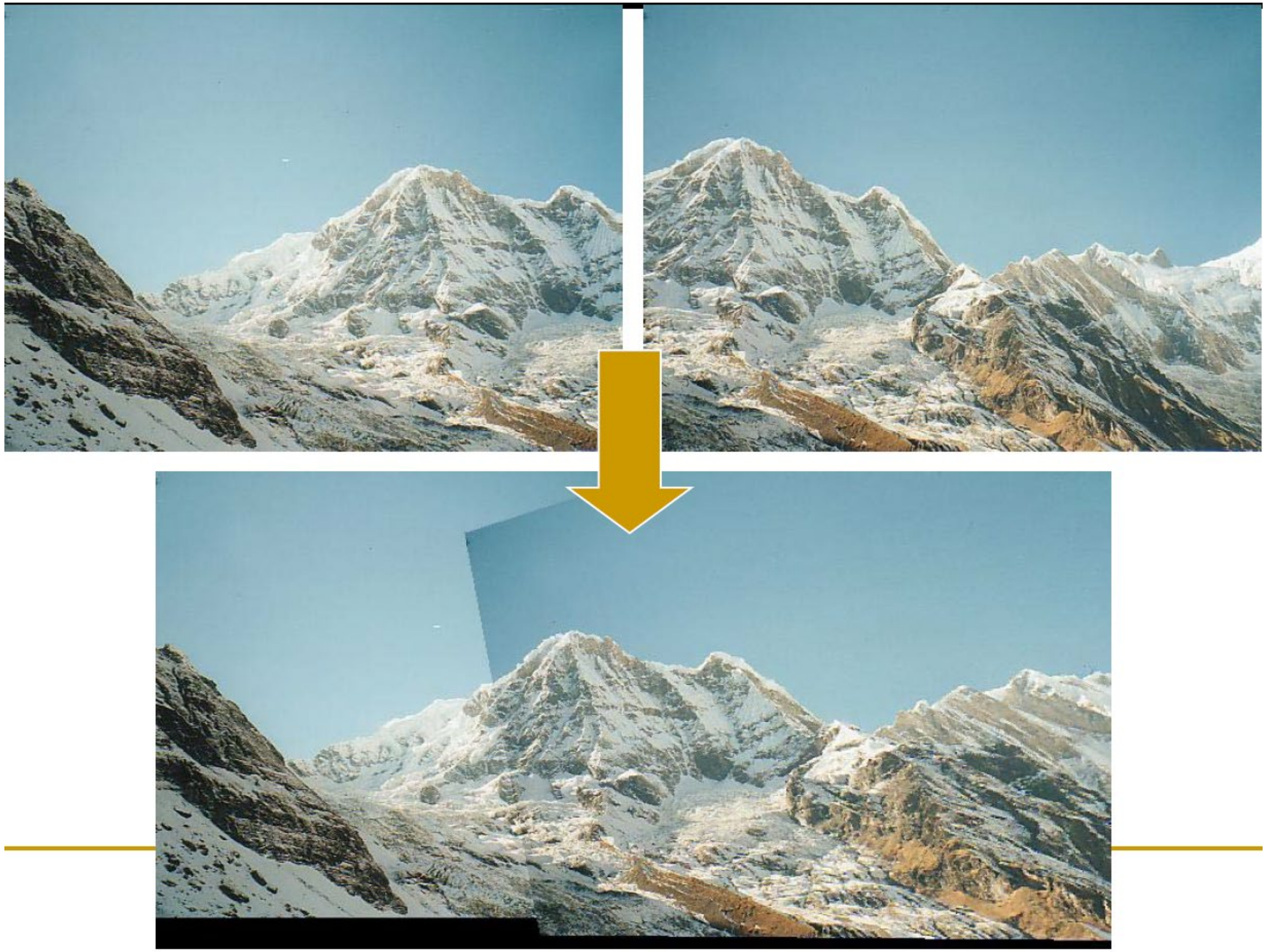


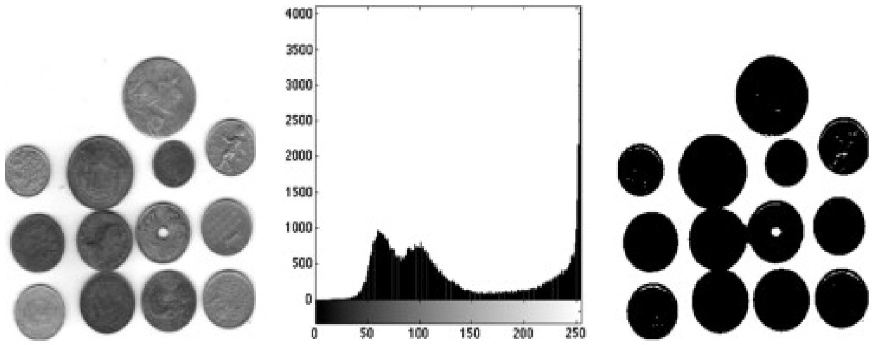
Image Stitching



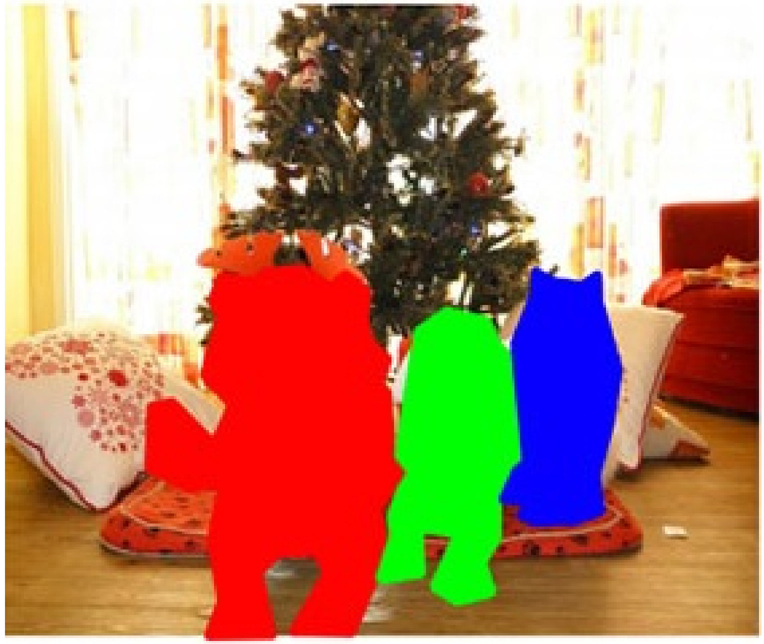
Pedestrian Detection

HOG

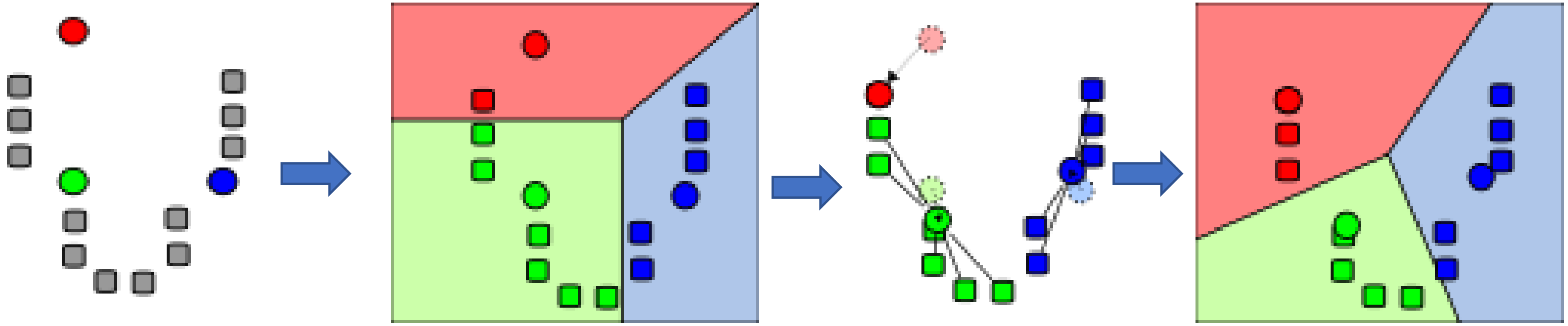
Image Segmentation



(a) (b) (c)



K-means Clustering



1. Initialize Cluster Centers

2. Assign Points to Clusters

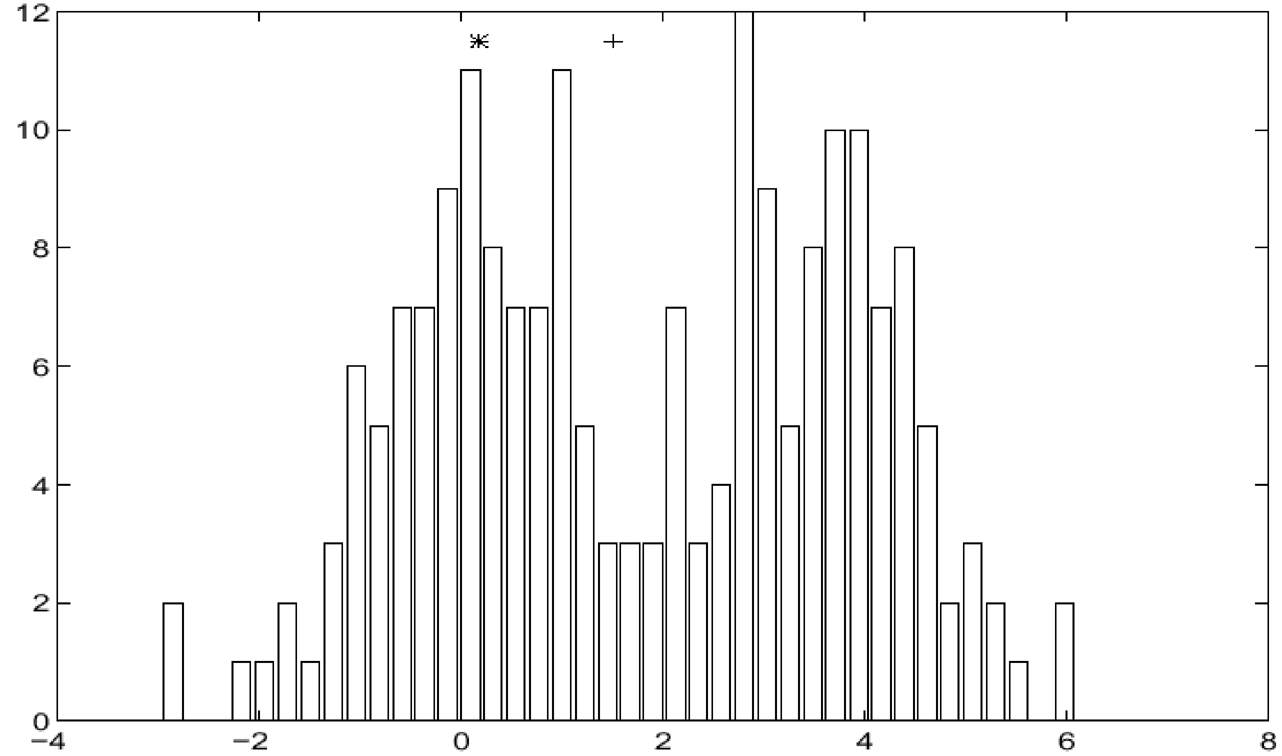
3. Re-compute Means

Repeat (2) and (3)

[(118) K-Means Clustering Example - YouTube]

Picture Credit: J. Carlos

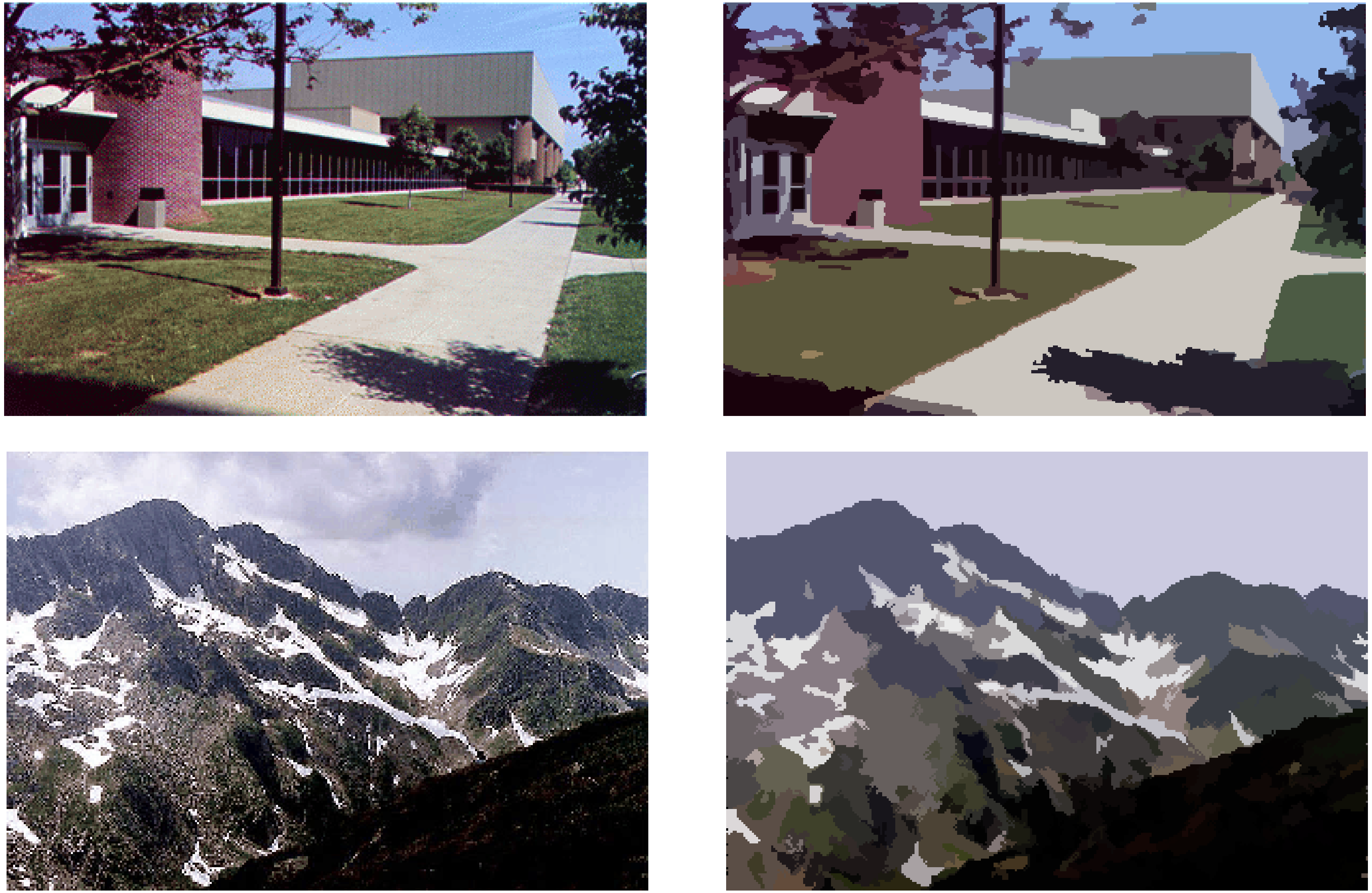
Mean-Shift Algorithm



- Iterative Mode Search
  1. Initialize random seed, and window W
  2. Calculate center of gravity (the "mean") of W: $\sum_{x \in W} xH(x)$
  3. Shift the search window to the mean
  4. Repeat Step 2 until convergence

Slide credit: Steve Seitz

Mean-Shift Segmentation Results



Optical Flow (Horn & Schunck)

- Compute Derivatives

```
fx = conv2(im1,[-1 1; -1 1], 'valid'); % partial on x
fy = conv2(im1, [-1 -1; 1 1], 'valid'); % partial on y
ft = conv2(im1, ones(2), 'valid') + conv2(im2, -ones(2), 'valid'); % partial on t
```

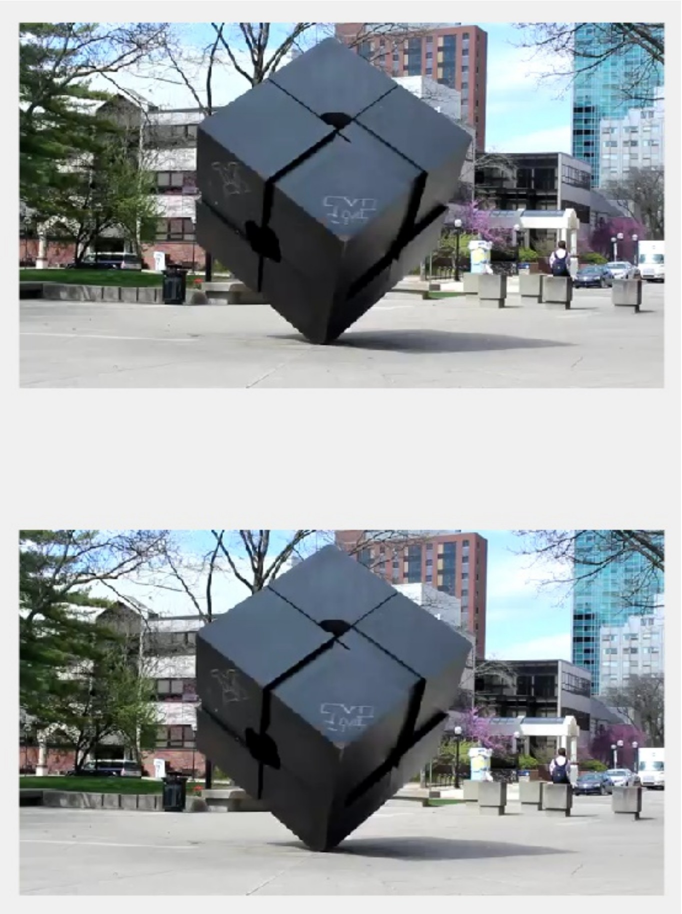
- Laplacian Mask and averaging

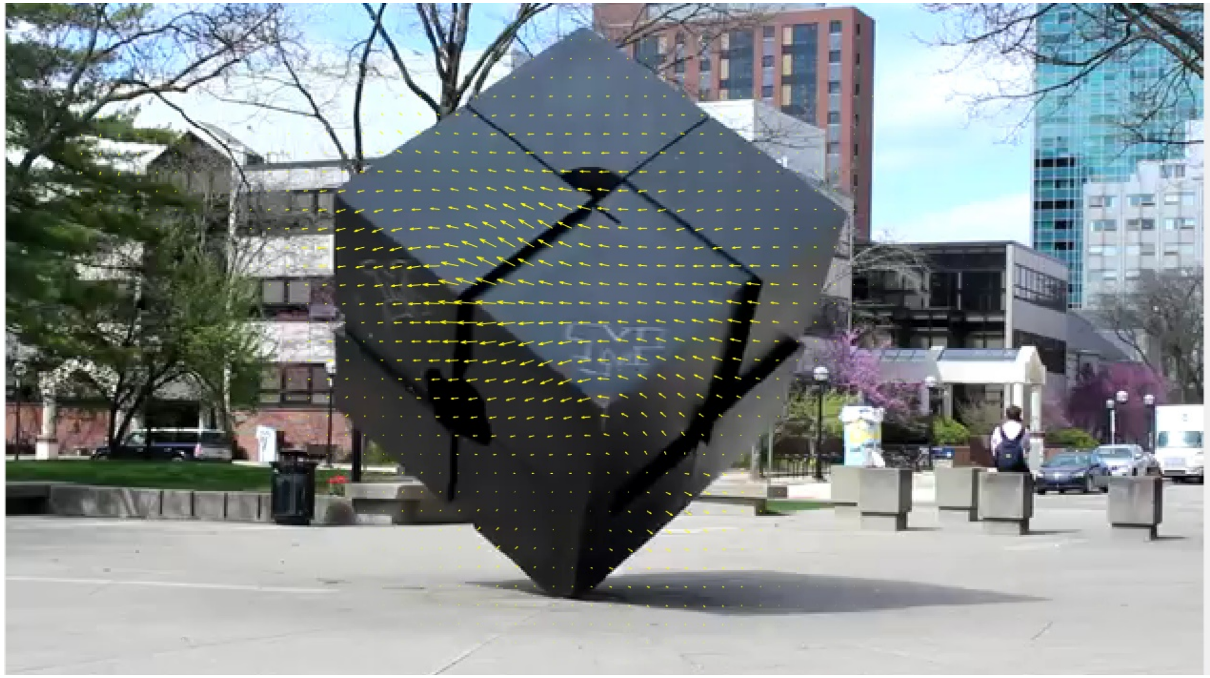
```
kernel_1=[0 1/4 0;1/4 0 1/4;0 1/4 0];

% or

kernel_1=[1/12 1/6 1/12;1/6 0 1/6;1/12 1/6 1/12];
```

```
uAvg=conv2(u,kernel_1,'same');
vAvg=conv2(v,kernel_1,'same');
```

- Iteration

```
alpha = (fx.*uAvg+fy.*vAvg+ft)./(1+lamda*(fx.^2+fy.^2));
u = uAvg-alpha.*fx;
v = vAvg-alpha.*fy;
```





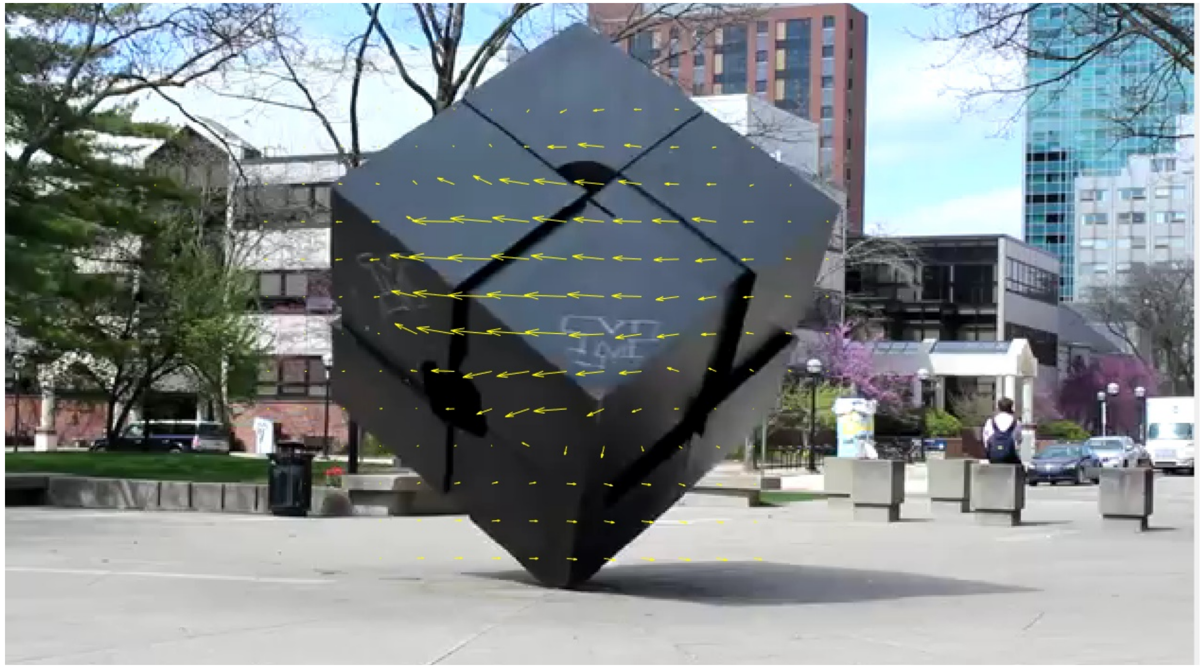
Lucas Kanade Matlab Example

- Compute Derivatives

```
fx = conv2(im1,[-1 1; -1 1], 'valid'); % partial on x
fy = conv2(im1, [-1 -1; 1 1], 'valid'); % partial on y
ft = conv2(im1, ones(2), 'valid') + conv2(im2, -ones(2), 'valid'); % partial on t
```

- Define the local windows

```
for i = w+1:size(Ix_m,1)-w
   for j = w+1:size(Ix_m,2)-w
      % Copy Ix, Iy, It to the windows
      Ix = Ix_m(i-w:i+w, j-w:j+w);
      Iy = Iy_m(i-w:i+w, j-w:j+w);
      It = It_m(i-w:i+w, j-w:j+w);

      Ix = Ix(:);
      Iy = Iy(:);
      b = -It(:); % This defines b

      A = [Ix Iy]; % This defines a
      vel = pinv(A)*b; % vel = pesudo inverse(a)*b

      u(i,j)=Vel(1);
      v(i,j)=Vel(2);
   end;
end;
```



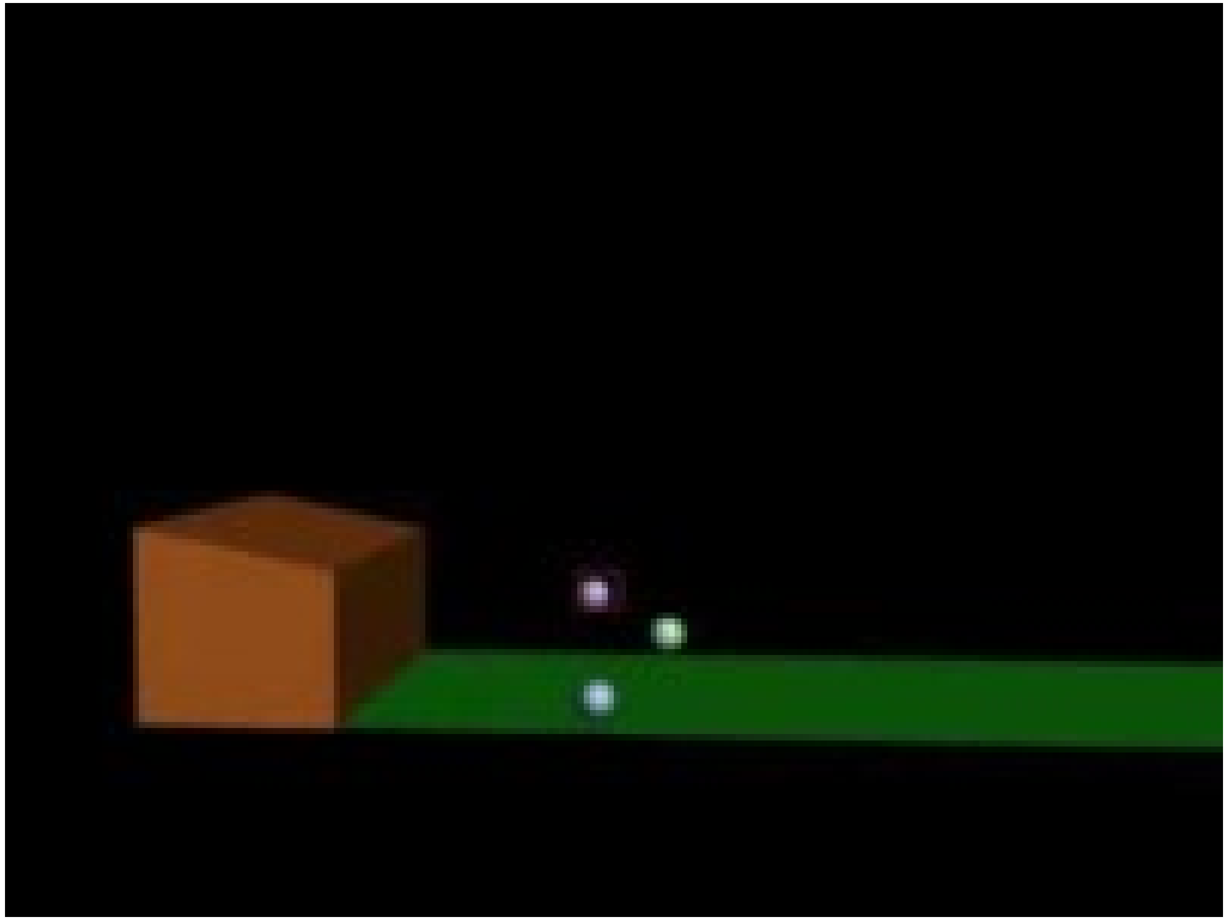
Tracking Objects (KLT, Mean Shift)

- Key assumptions:
  - **Brightness constancy:** projection of the same point looks the same in every frame (uses SSD as metric)
  - **Small motion:** points do not move very far (from guessed location)
  - **Spatial coherence:** points move in some coherent way (according to some parametric motion model)
- For this example, assume whole object just translates in (u,v)

Template T

Image Frame I

Tracking (Mean-Shift)



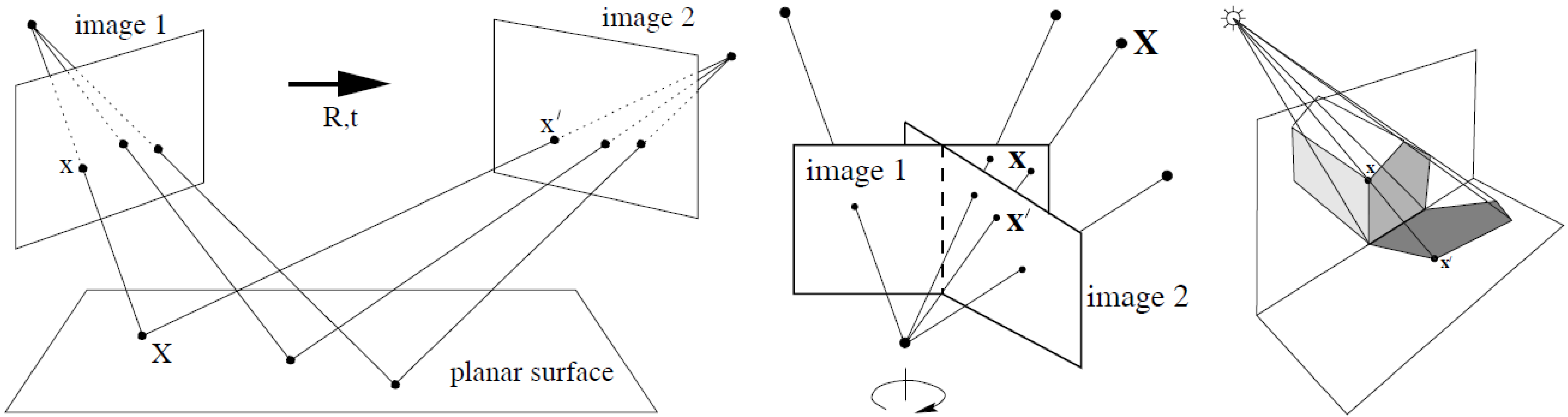
Projective Geometry

Size preserving?

lines remains lines?

Angle preserving?

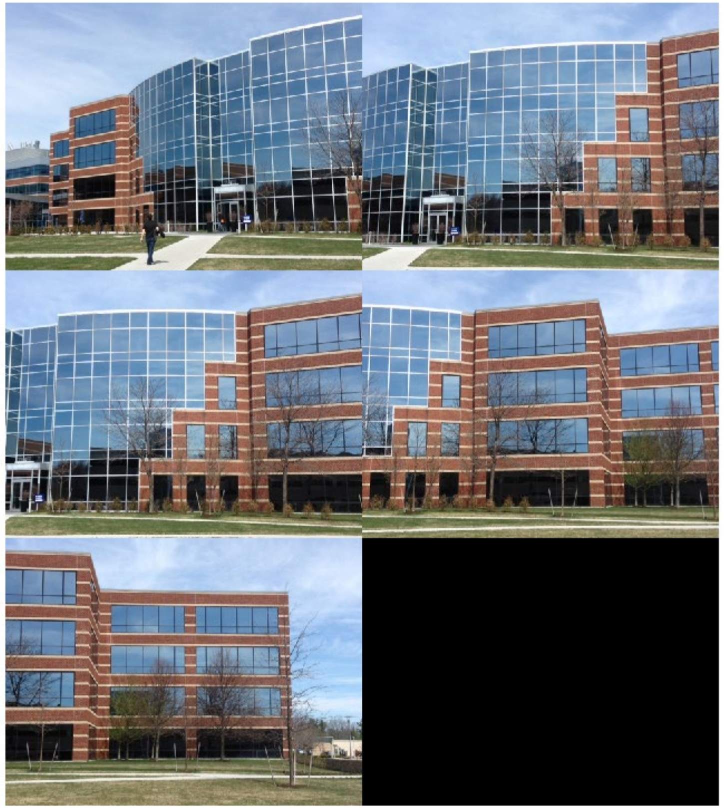
Examples of Projective transformation

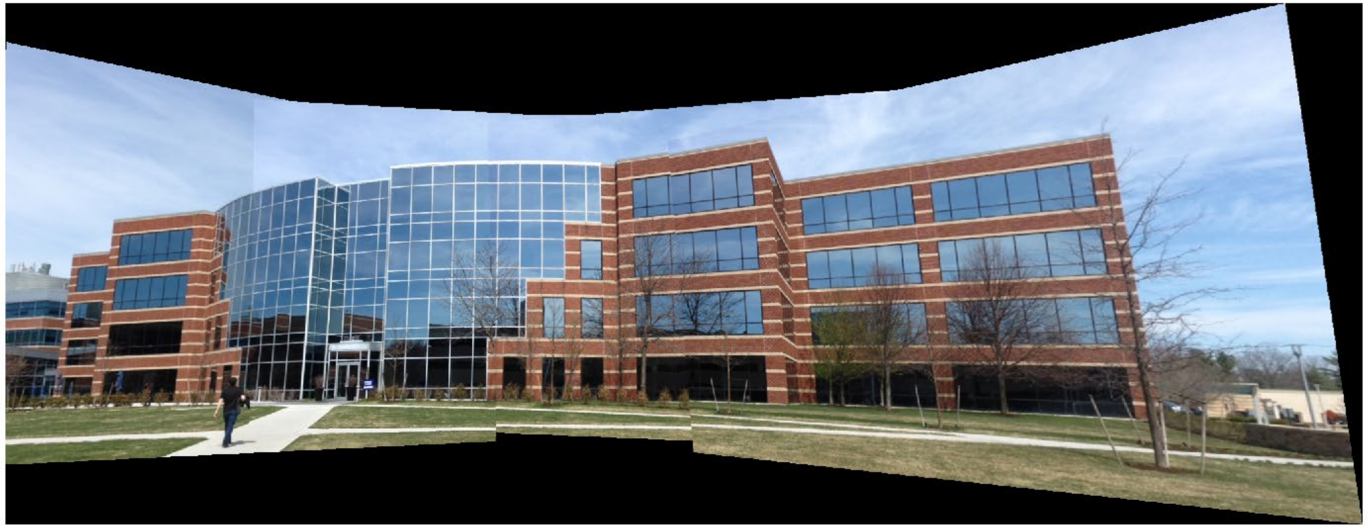


- The transformation $\mathrm{x}' = \mathrm{Hx}$ examples
  - Two images induced by a world plane(concatenation of two projective transformations is a projective transformation)
  - Two images with same camera centres
  - Image of a plane and image of its shadow onto another plane.

Image Credit: R. Hartley and A. Zisserman, “Multiple View Geometry in Computer Vision”

Homography and Image Stitching





Removing projective distortion of images

- Distortions arising under different transformations

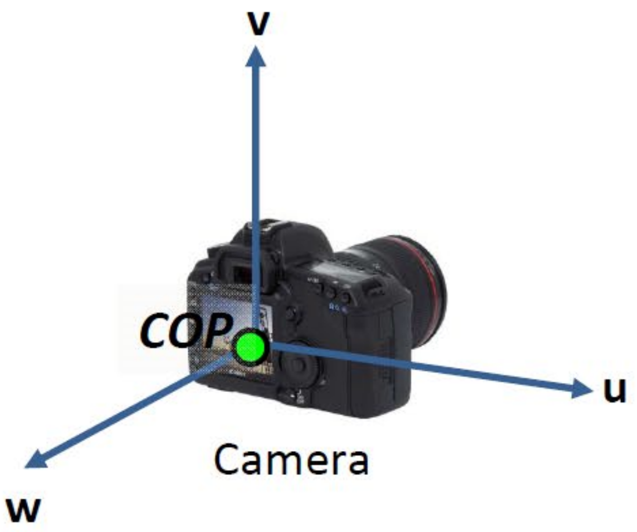
Similarity

Affine

Projective

Image Credit: R. Hartley and A. Zisserman, "Multiple View Geometry in Computer Vision"

Camera and World Co-ordinate Systems



Camera

Two important coordinate systems:

1. *World* coordinate system
2. *Camera* coordinate system

"The World"

Camera Parameters and Calibration

$$\mathbf{x} = \mathbf{K}\left[\mathbf{R} \middle| \mathbf{t}\right]\mathbf{X}$$

Camera Calibration

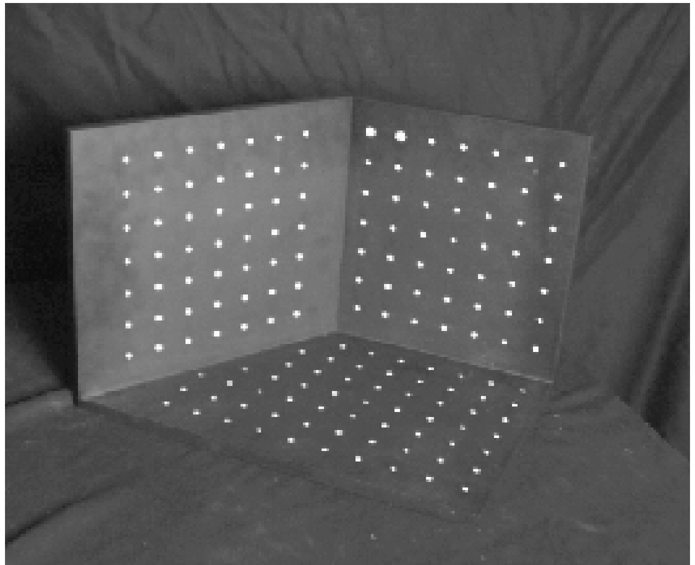
- Mapping from World Coord to image

$$\mathbf{x}=\mathrm{K}R[I_3|-\boldsymbol{X}_o]\mathbf{X}$$

- Goal: To estimate the intrinsic and extrinsic parameters of the camera

- Given: Known 3D points

- Observation: corresponding 2d points



Method 1: Direct Linear Transform (DLT)

$$\mathbf{x} = \mathrm{K}R[I_3| - \boldsymbol{X}_o]\mathbf{X}$$

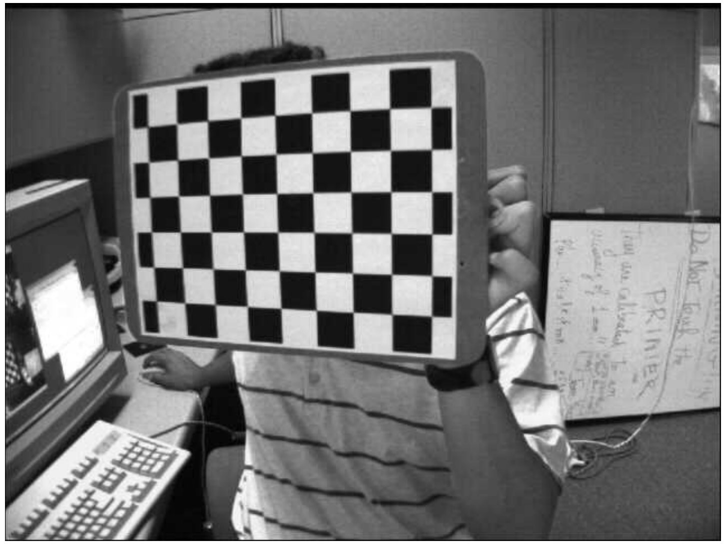
$$\mathbf{x} = \begin{bmatrix} u \\ \boldsymbol{v} \\ \boldsymbol{w} \end{bmatrix} = \underbrace{\begin{bmatrix} c & s & x_H \\ 0 & c(1+m) & y_H \\ 0 & 0 & 1 \end{bmatrix}}_{\boxed{3\text{x}3}} \underbrace{\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}}_{\boxed{3\text{x}3}} \underbrace{\begin{bmatrix} 1 & 0 & 0 & t_X \\ 0 & 1 & 0 & t_Y \\ 1 & 0 & 1 & t_Z \end{bmatrix}}_{\boxed{3\text{x}4}} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

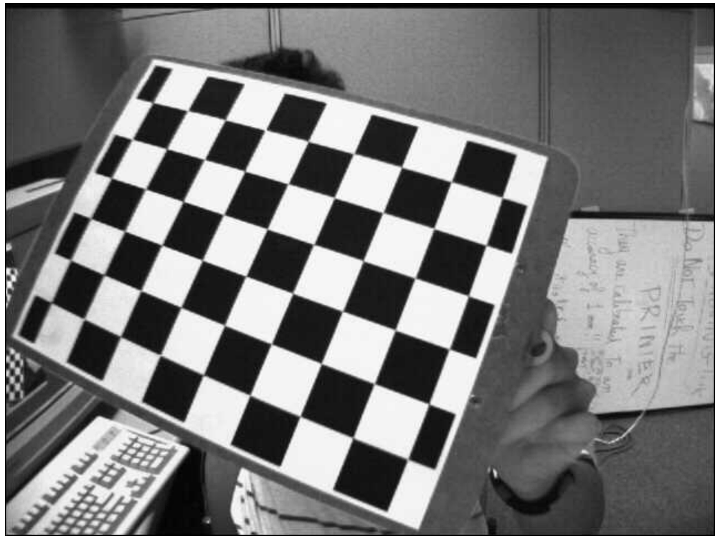
- can be combined as

$$\mathbf{x} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Method 2: Multi-Plane Camera Calibration(Zhang 2000)

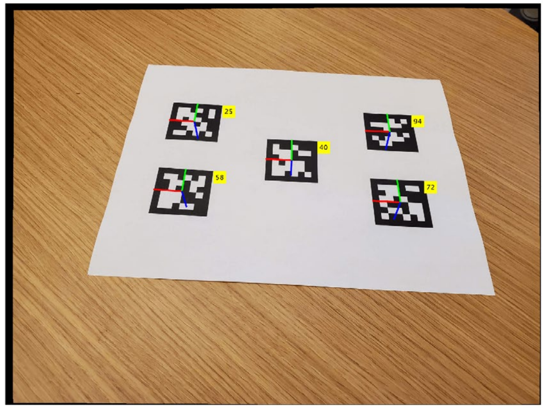
- Use a 2D known pattern (checkerboard) for calibration





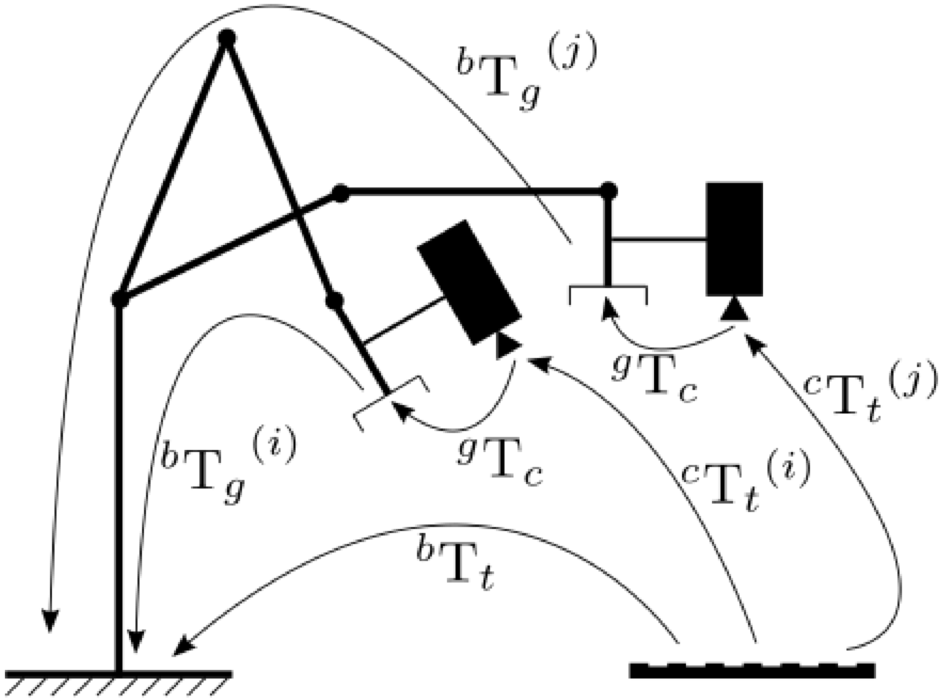
Applications

Distortion Removal



Camera Pose estimation

Image Credit: www.mathworks.com



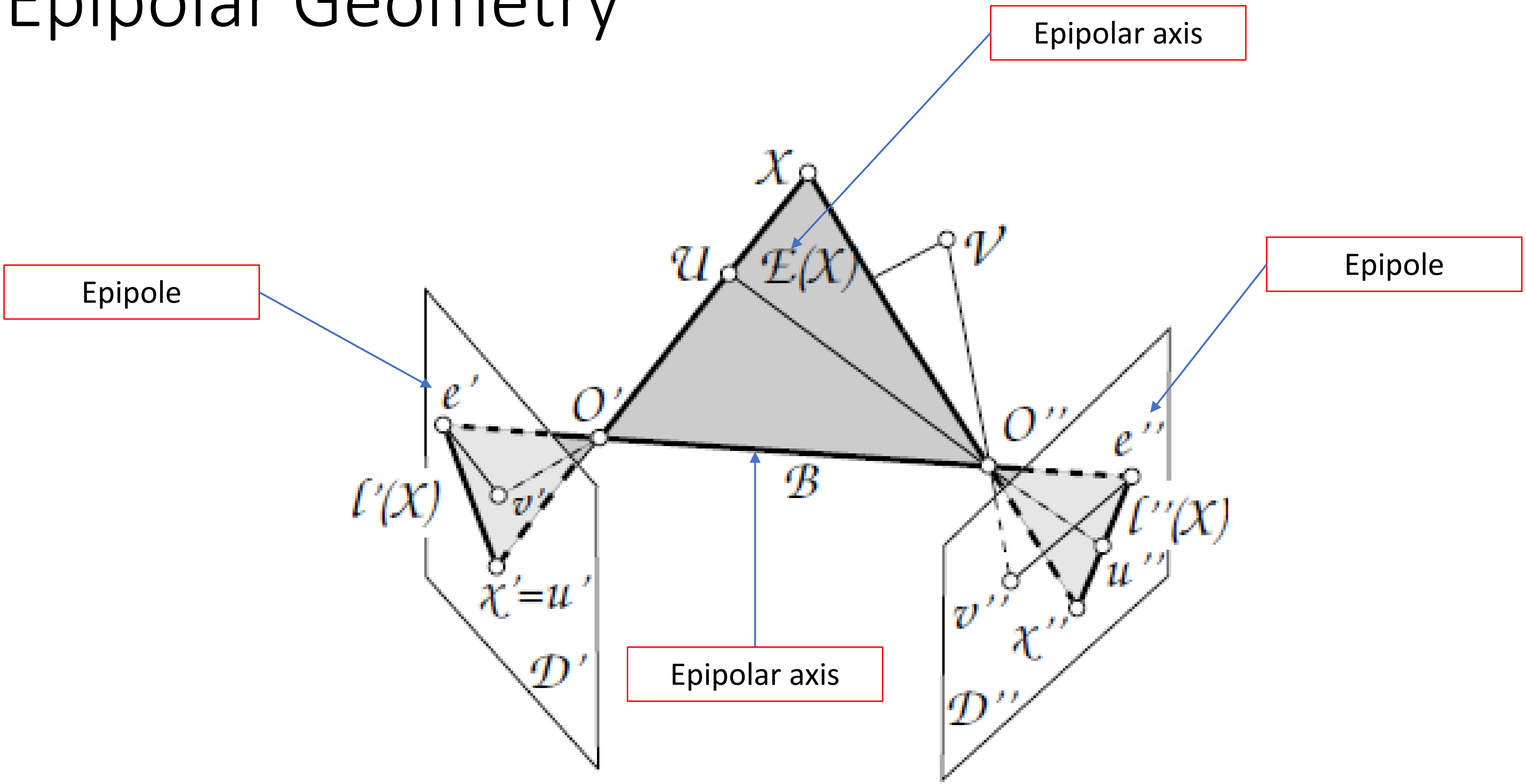
Eye-in-hand calibration

Image Credit: www.opencv.org

Epipolar Geometry

- Given a single image, we are not able to infer its three-dimensional structure as the depth information is lost.
- Therefore image pairs are considered!

Epipolar Geometry



Example – Epipolar Line

- Given: Two stereo image with known correspondence.
- Aim: to find the epipolar lines and epipoles.

I1 (left); I2 (right)

Correspondence Points

- Given: Correspondence Points in Image 1 and Image 2
- $\mathrm{l}'' = \mathsf{F}^{\boldsymbol{T}}\mathbf{x}'$

Correspondence Points in Image 1

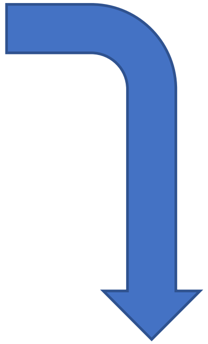
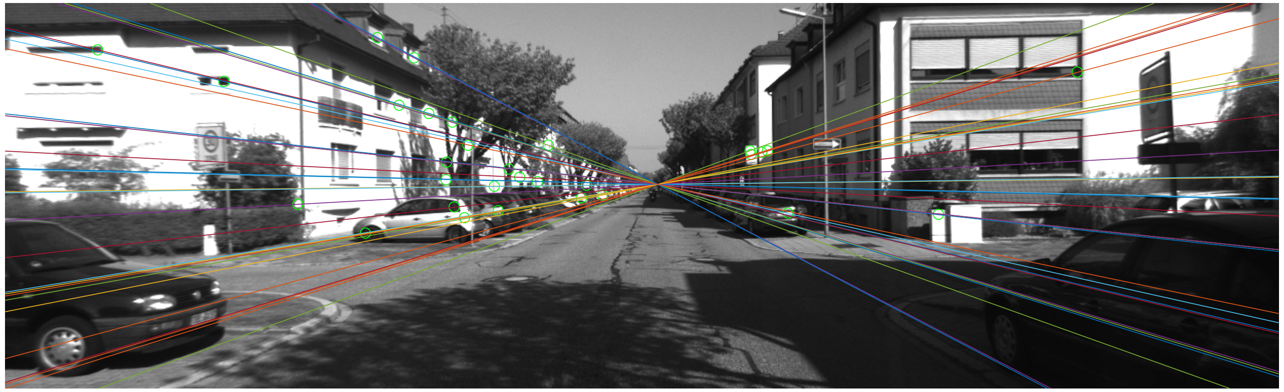


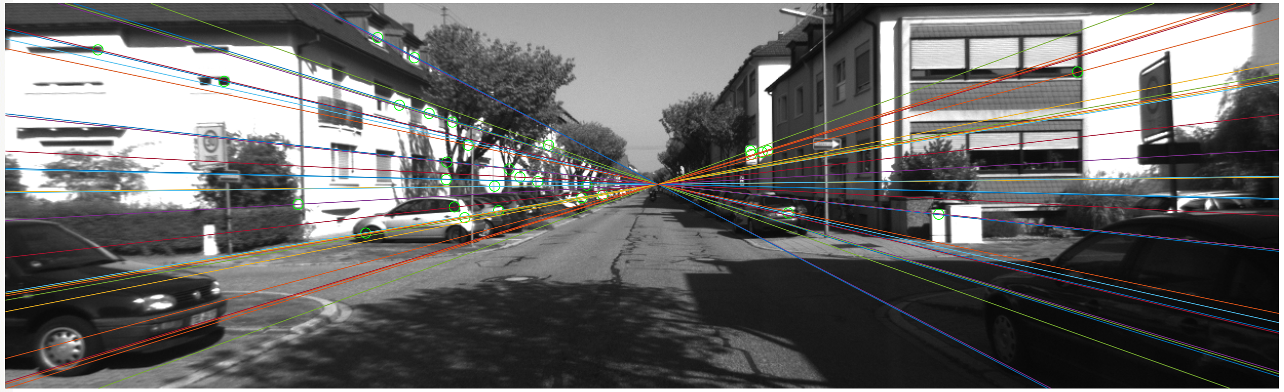
Image 2 Epipolar Lines correspond to $\mathbf{x}'$



MATLAB Codes

```
figure; ax=axes;
imshow(I2);

%showMatchedFeatures(I1, I2, inlierPoints1a, inlierPoints2a);
%legend('Inlier points in I1', 'Inlier points in I2');
hold on;
%plot(inlierPoints2a(:,1),inlierPoints2a(:,2),'go')

%drawing the epipole
epipoleHC1=null(fMatrix2);
plot(epipoleHC1(1)/epipoleHC1(3),epipoleHC1(2)/epipoleHC1(3),'go');


%Calculate by formula l''=F'X'
inlierPoints1aHC=cart2hom(inlierPoints1a);
lss = zeros(30,3);
for i=1:30
    point = inlierPoints1aHC(i,:);
    lss(i,:)=fMatrix2'*point';  %the epipolar lines in image2
end

points = lineToBorderPoints(lss, size(I2));
line(points(:, [1,3])', points(:, [2,4])');
truesize;
hold off;
```

Epipolar Lines correspond to $\mathbf{x}'$



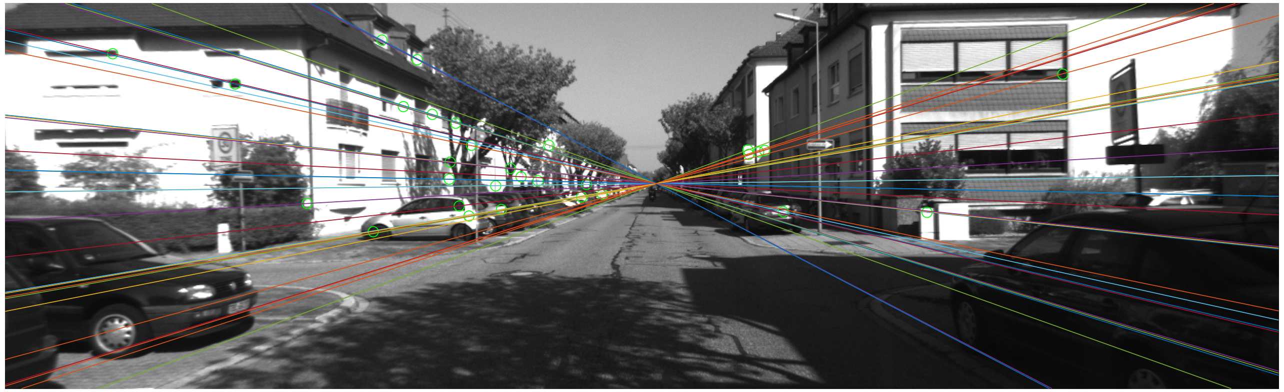
Similarly for Image 2

- Similarly for Image 1
- Epipolar lines in Image 1 are

$$\mathbf{l}' = \mathbf{F}\mathbf{x}''$$

Correspondence Points in Image 2

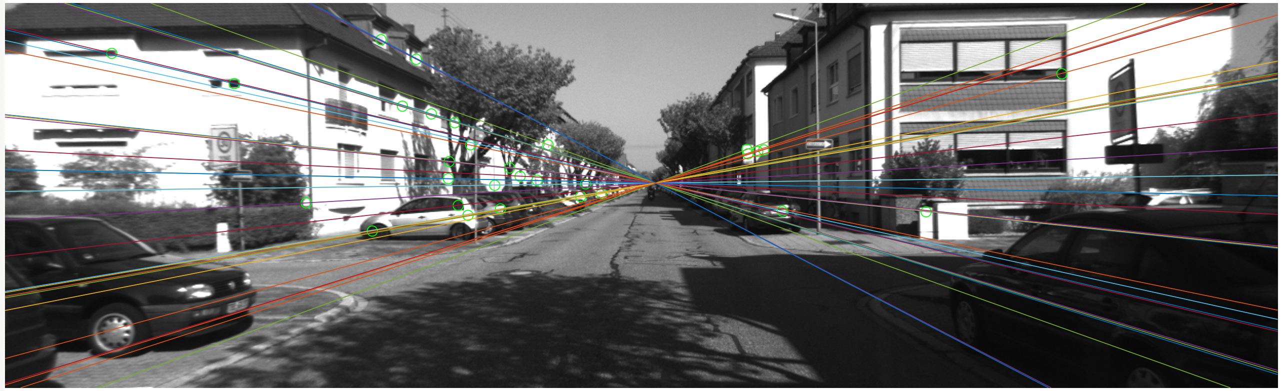
Image 1: Epipolar Lines correspond to $\mathbf{x}''$



MATLAB Codes

```
figure; ax=axes;
imshow(I1);
hold on;
%drawing the corresponding points
%plot(inlierPoints1a(:,1),inlierPoints1a(:,2),'go');

%drawing the epipole
epipoleHC1=null(fMatrix2');
plot(epipoleHC1(1)/epipoleHC1(3),epipoleHC1(2)/epipoleHC1(3),'go');


% Compute the Epipolar Lines from corresondence point on second image
inlierPoints2aHC=cart2hom(inlierPoints2a);
ls = zeros(30,3);
for i=1:30
    point = inlierPoints2aHC(i,:);
    ls(i,:)=fMatrix2*point'; % The epipolar line in image 1
end

%drawing the lines
points = lineToBorderPoints(ls, size(I2));
line(points(:, [1,3])', points(:, [2,4])');
truesize;
hold off;

figure; ax=axes;
imshow(I2);
```

Epipolar Lines correspond to $\mathbf{x}''$



Fundamental Matrix

F is the *fundamental matrix* of the *relative orientation* of a pair of images of *uncalibrated cameras*

$$\mathrm{F}=(\mathrm{K}')^{-T} R' \, S_b \, (R'')^T (\mathrm{K}'')^{-1}$$

Which fulfills the equation

$$\mathbf{x}'^{\,T} \mathrm{F} \, \mathbf{x}'' = 0$$

- Remarks:
  - Fundamental matrix F 3x3 matrix with 7 DoF.
  - F is homogenous (AF=0)
  - F is singular det(F)=0

Essential Matrix

- The essential matrix for calibrated camera

$$\mathrm{E} = R' \, S_b \, (R'')^T$$

- E express the R.O. and can be parametrized through

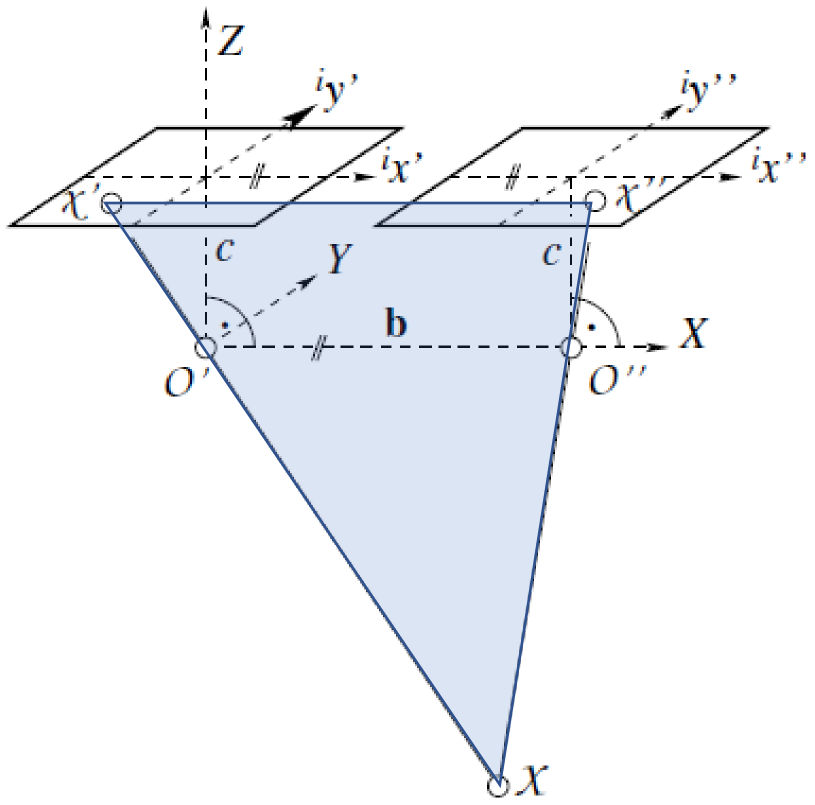
$$\mathrm{E} = \; S_b \, R^T$$

The coplanarity constraint for calibrated cameras

$${}^{c}\mathrm{x}'^{T} \mathrm{E} \; {}^{c}\mathrm{x}'' \; = 0$$

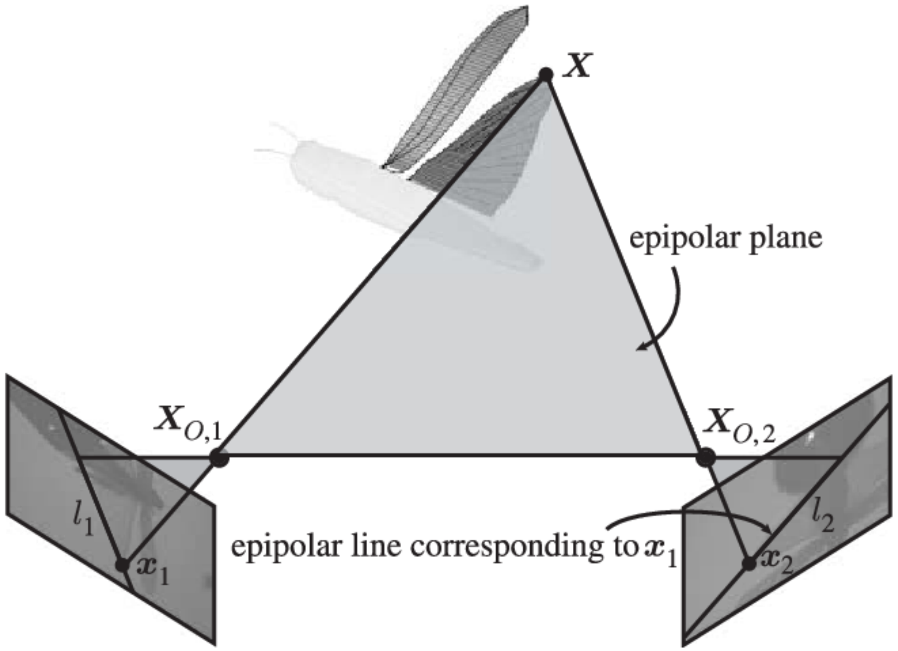
Reconstruction - Stereo Normal Case

- Images in the same plane
- Identically oriented (R=I)
- Basis Vector B in X-direction

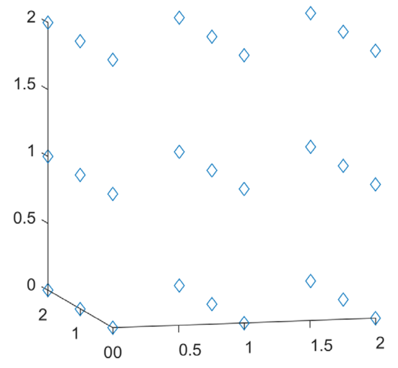


Reconstruction - Triangulation

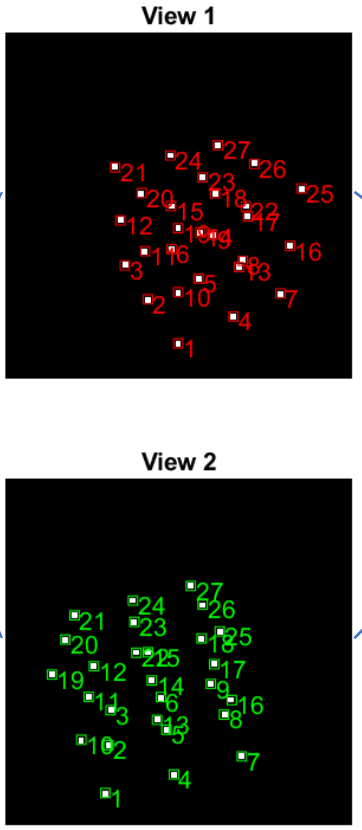
- **Given**: A pair of cameras with known relative orientation
- **Aim**: To compute the points in 3D



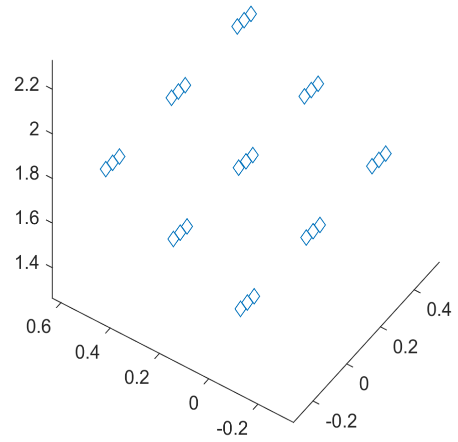
Example



Original Points



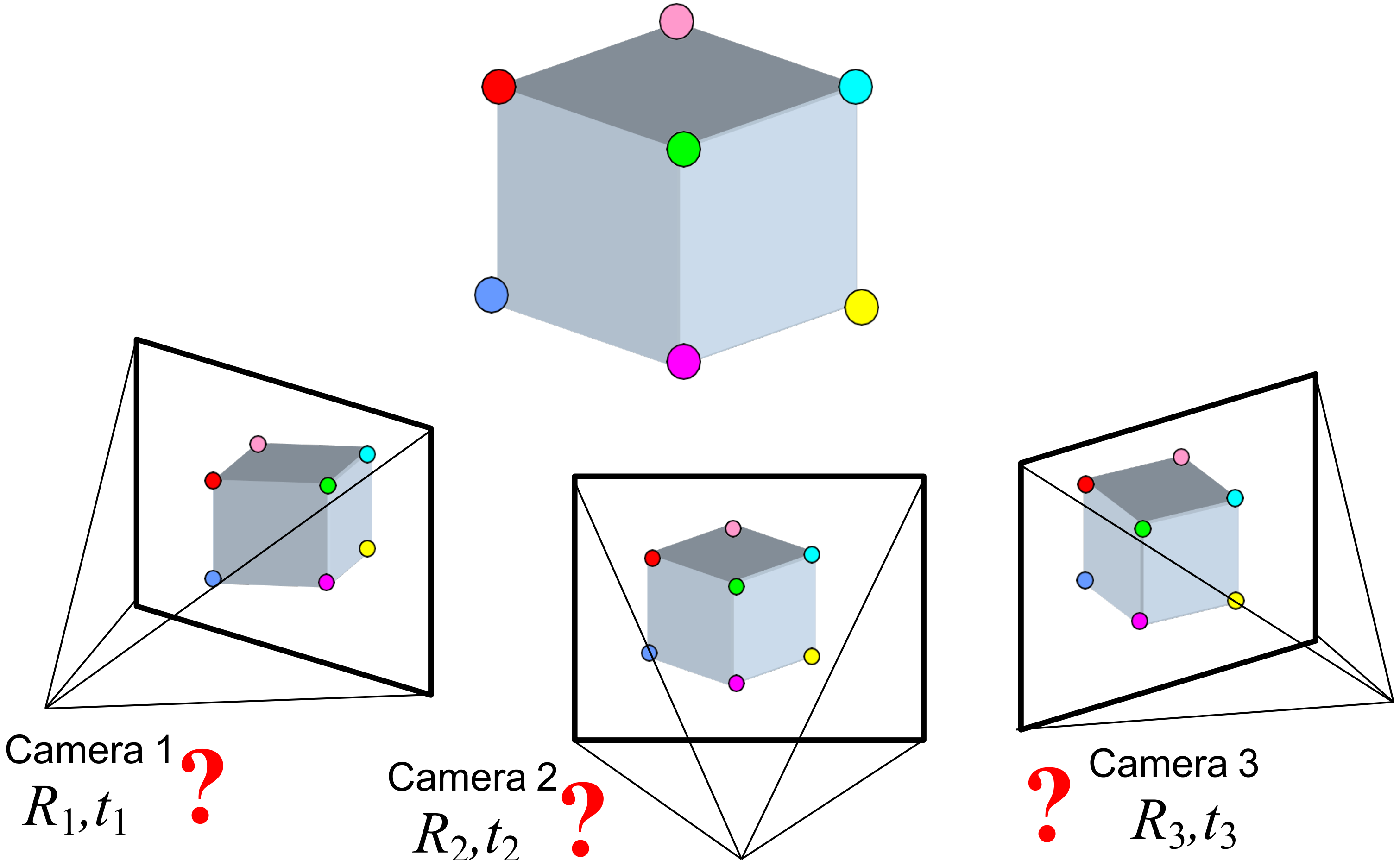
Projected Images



Reconstructed Points

Reconstruction: Structure from multiple views

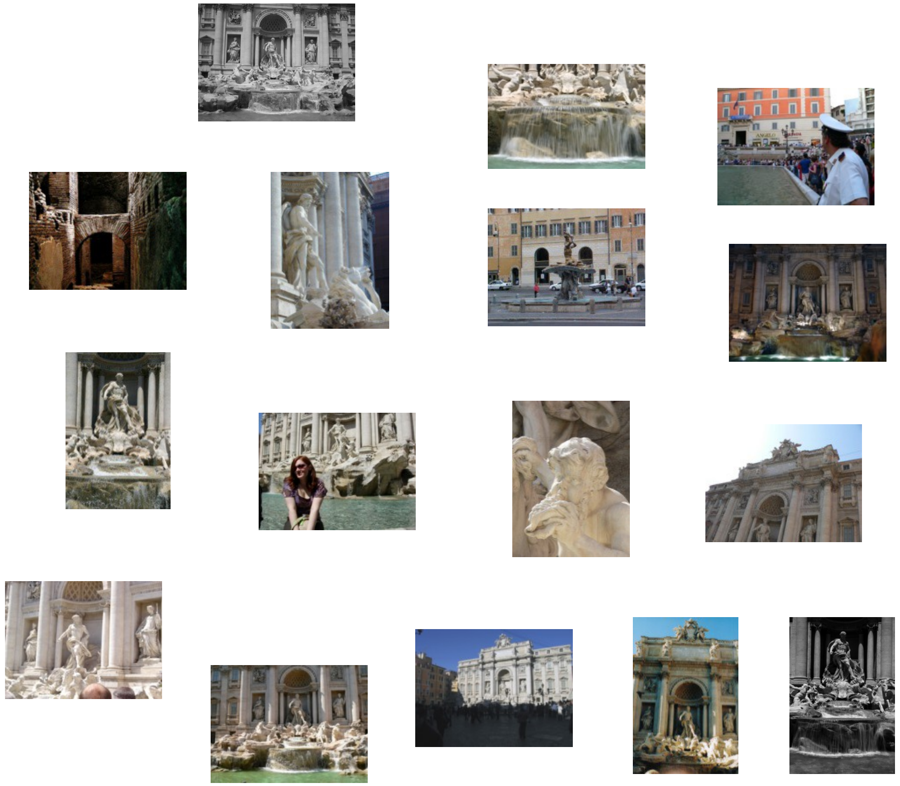
- Given a set of corresponding points in two or more images, compute the camera parameters and the 3D point coordinates



Slide credit: Noah Snavely

Feature Detection

- Detect SIFT features



Feature matching

- Use RANSAC to estimate fundamental matrix between each pair

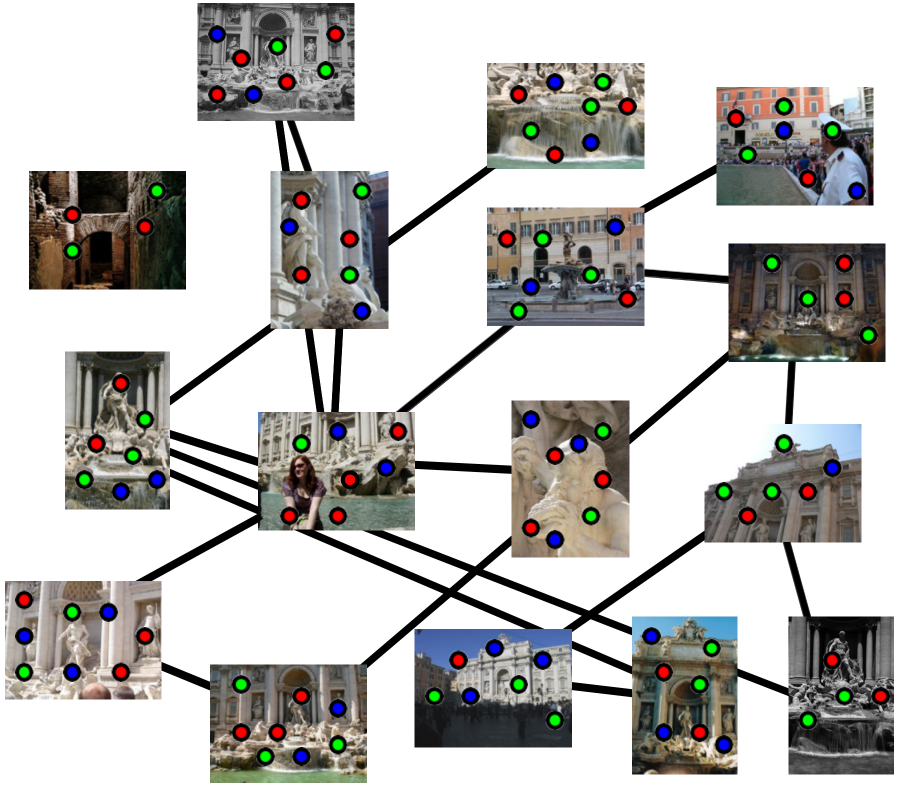
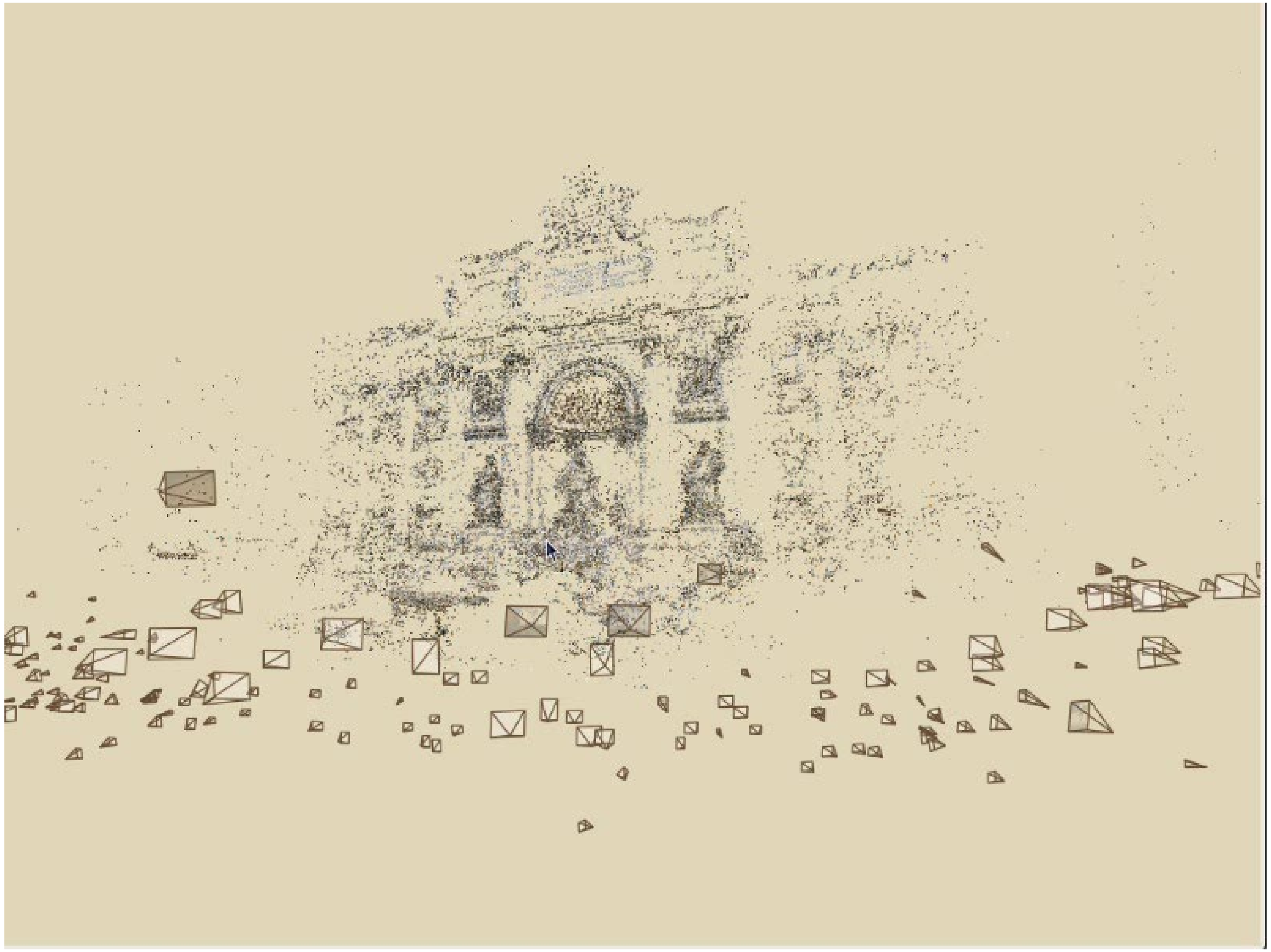
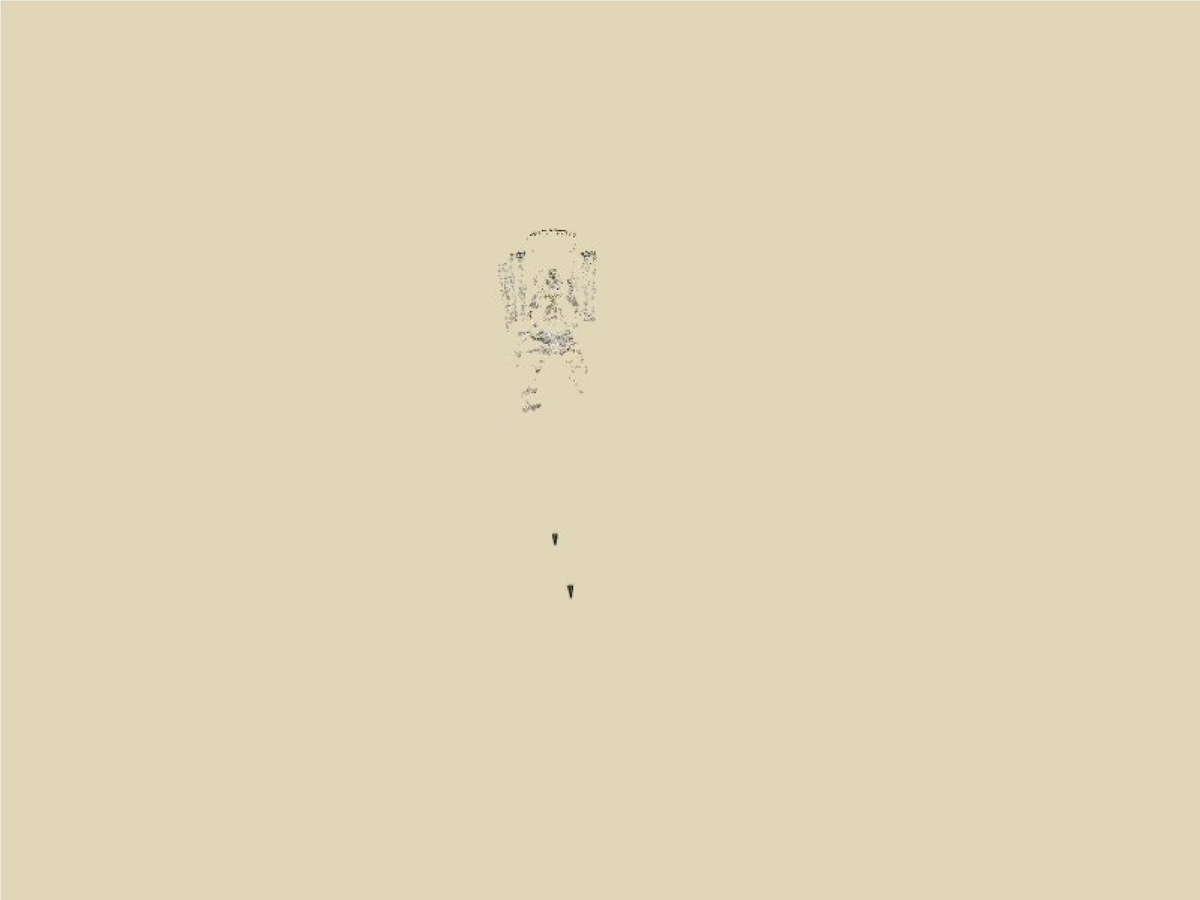


Photo Tourism



Slide Source: Ioannis (Yannis) Gkioulekas

Incremental structure from motion

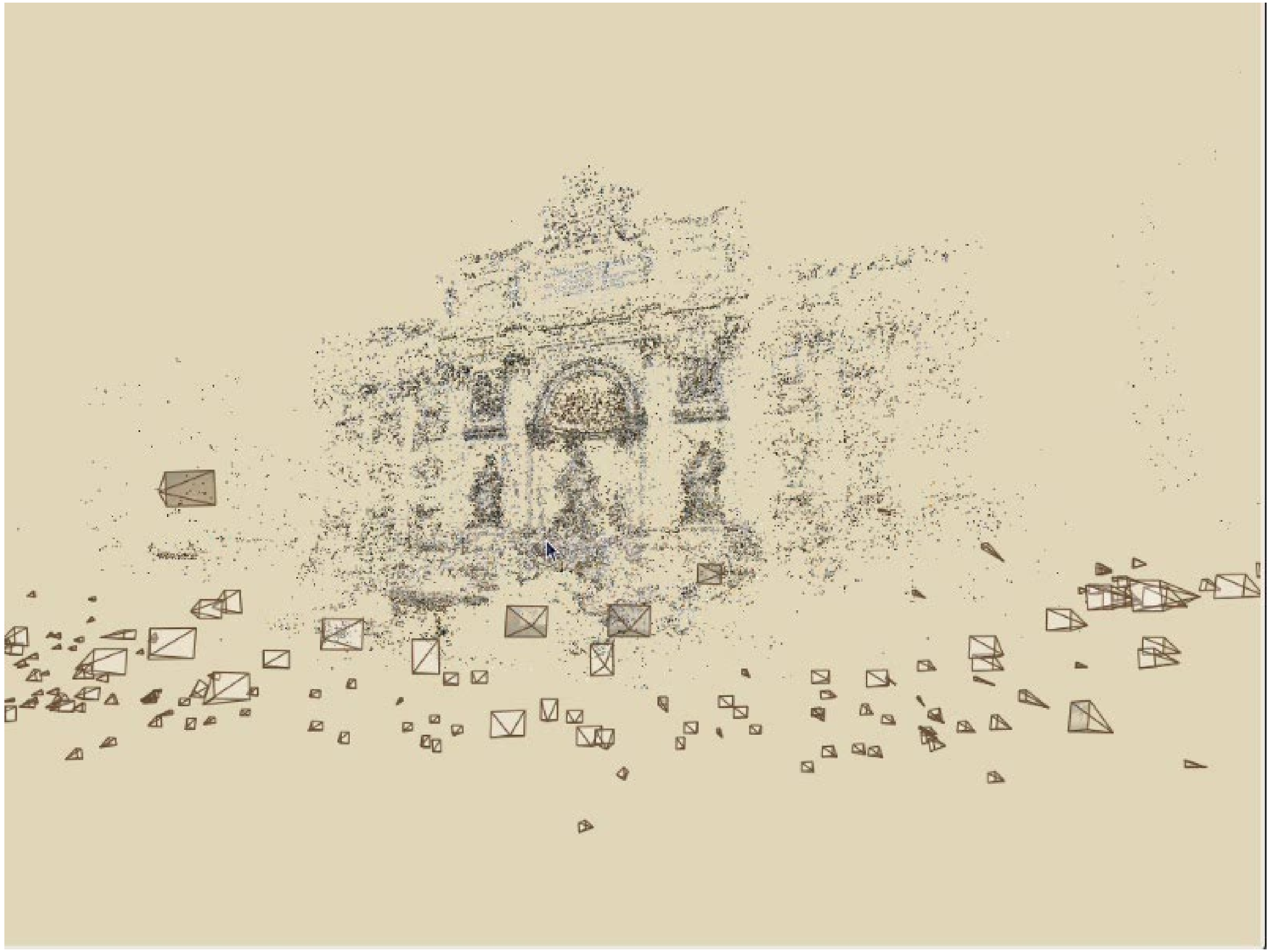


Slide Source: Ioannis (Yannis) Gkioulekas

Final reconstruction

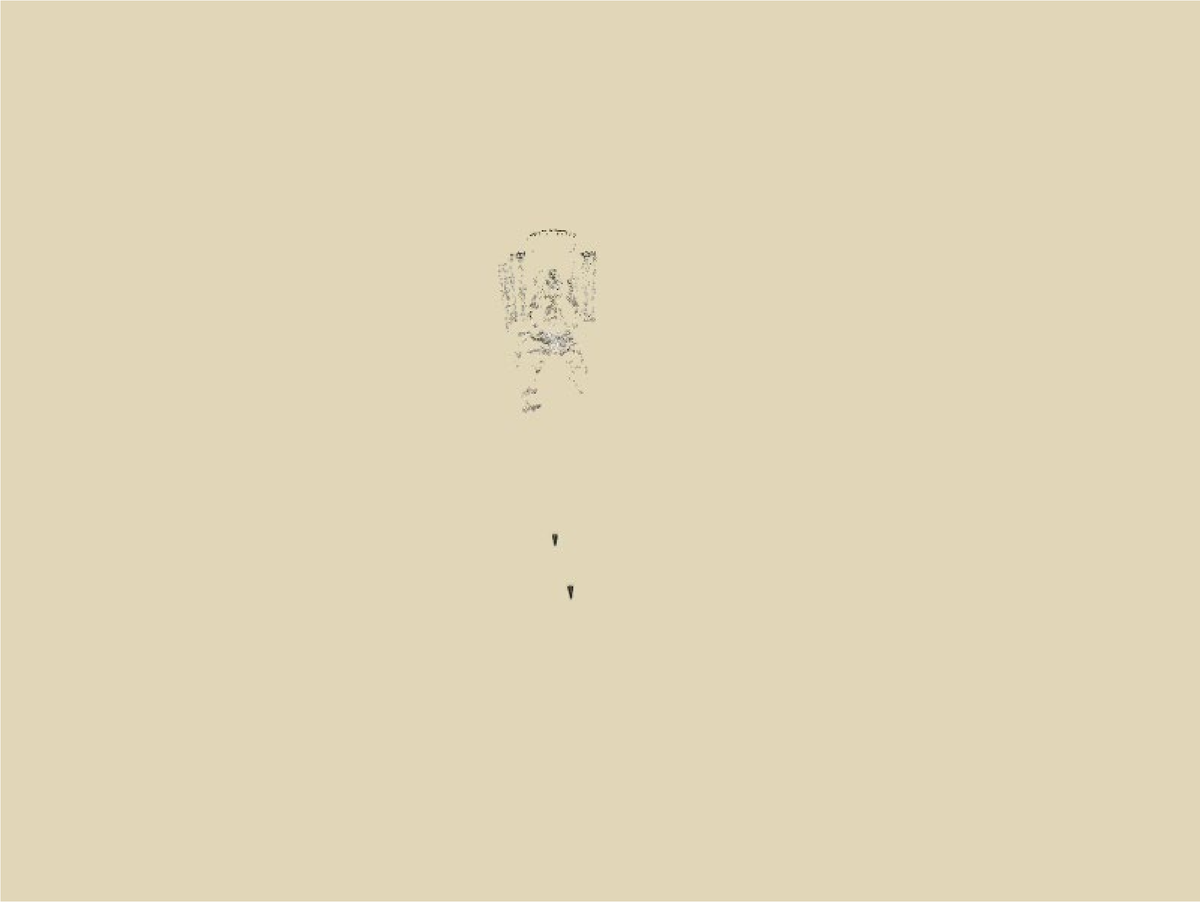
Slide Source: Ioannis (Yannis) Gkioulekas

Photo Tourism



Slide Source: Ioannis (Yannis) Gkioulekas

Incremental structure from motion



Slide Source: Ioannis (Yannis) Gkioulekas

Final reconstruction

Slide Source: Ioannis (Yannis) Gkioulekas

Some Useful Mathematics

- Interpolation (Nearest Neighbour, Bilinear, Bicubic)
- Eigen Value Decomposition
- Singular Value Decomposition (SVD)
- QR Factorization
- Cholesky Factorization
- Linear/Total least squares
- RANSAC

Assessment policy

- Homeworks 50%
  - 4 MATLAB programming homework
- Mini Projects 50%
  - 2 mini-projects

- Late Policy
  - Late assignment and project will be penalized by 5% for each day it is late, and no extra credit will be awarded.

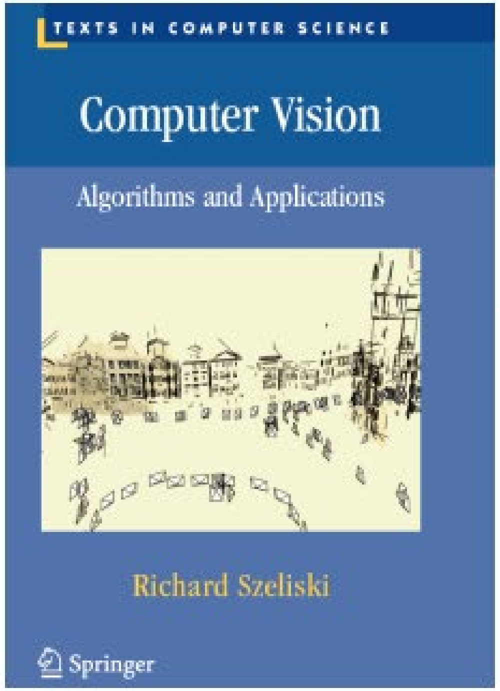
Prerequisites

- Interest in Computer Vision
- How basic programming technique (MATLAB)
  - Need Tutorial?

Reference Book

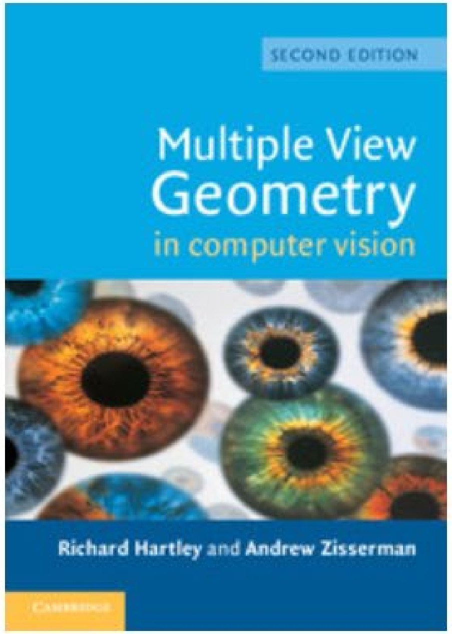
Computer Vision: Algorithms and Applications

© 2010 [Richard Szeliski](), Microsoft Research



http://szeliski.org/Book/

Reference Book



Get access Cited by **4602**

2nd edition

Richard Hartley, *Australian National University, Canberra*, Andrew Zisserman, *University of Oxford*

Publisher:	Cambridge University Press
Online publication date:	January 2011
Print publication year:	2004
Online ISBN:	9780511811685
DOI:	https://doi.org/10.1017/CBO9780511811685

Subjects: Computer Science, Computer Graphics, Image Processing and Robotics

Reference Book

igg

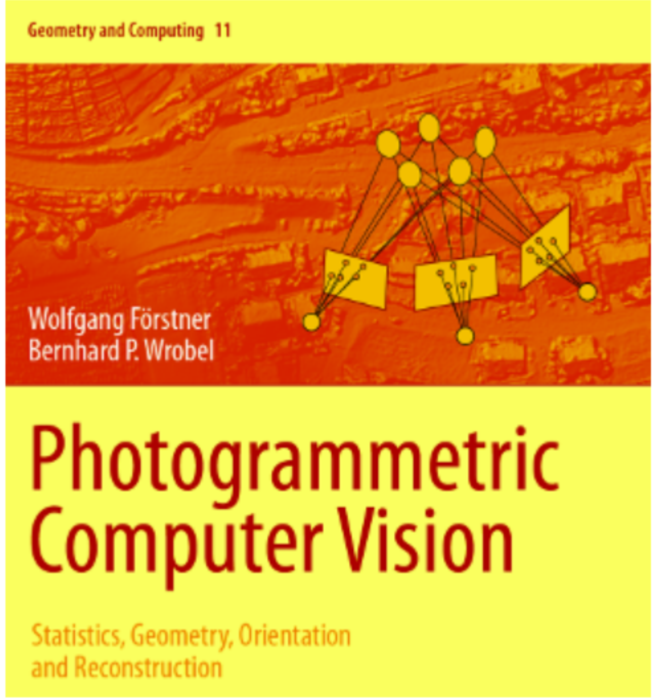
Springer

Photogrammetric Computer Vision

Statistics, Geometry, and Reconstruction

Wolfgang Förstner, Bernhard P. Wrobel

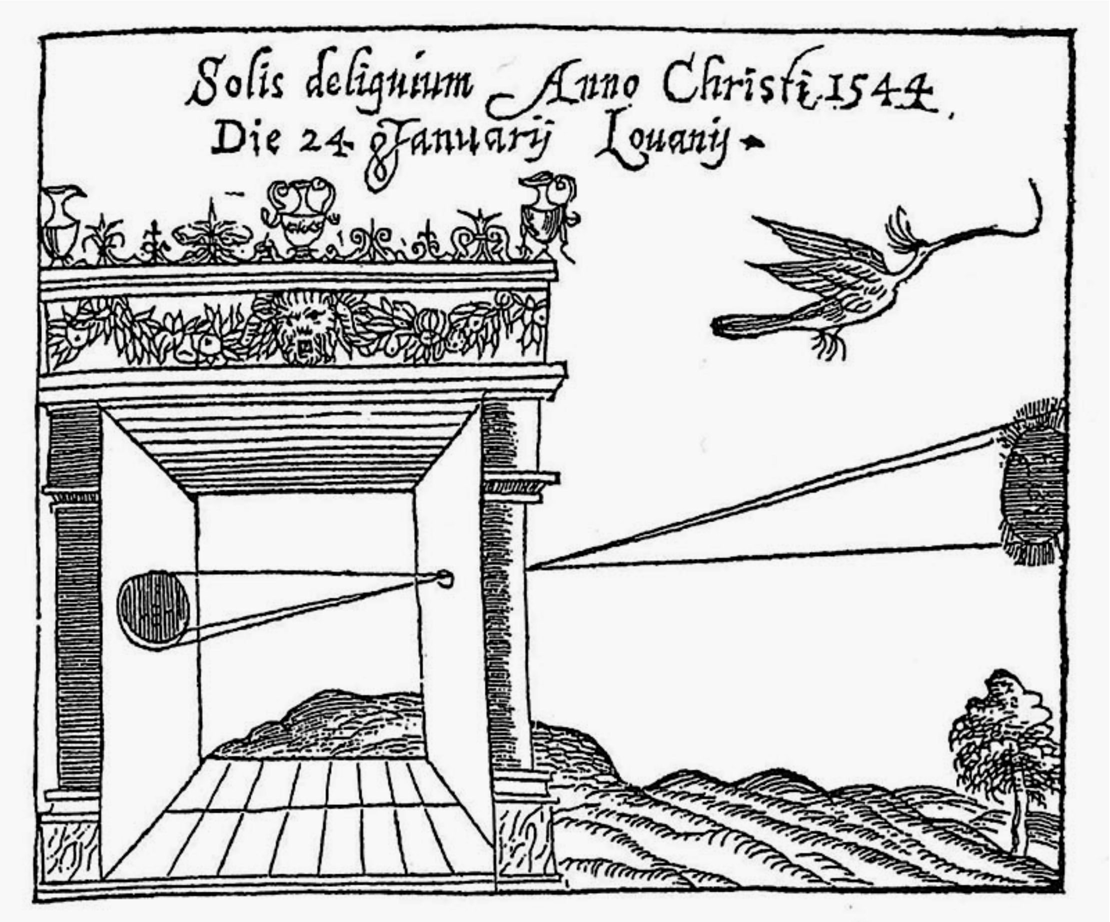
A Springer book



Photogrammetric Computer Vision -- Statistics, Geometry, and Reconstruction (uni-bonn.de)

https://www.ipb.uni-bonn.de/book-pcv/#start

Next Class



Reference

Richard Szeliki, “Computer Vision: Algorithm and Applications, Springer 2010 (http://szeliski.org/Book/.) Chapter 1