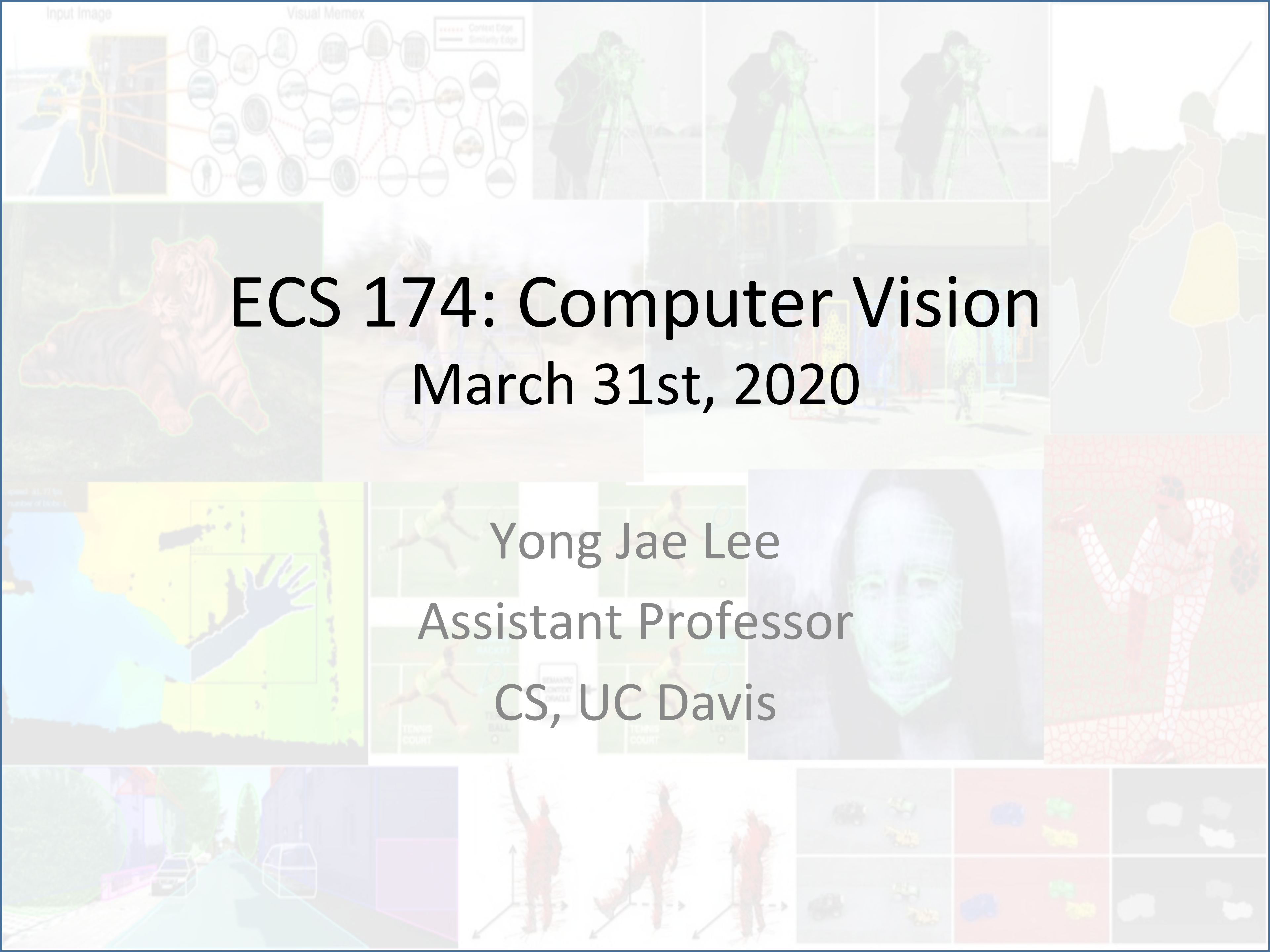


ECS 174: Computer Vision

March 31st, 2020

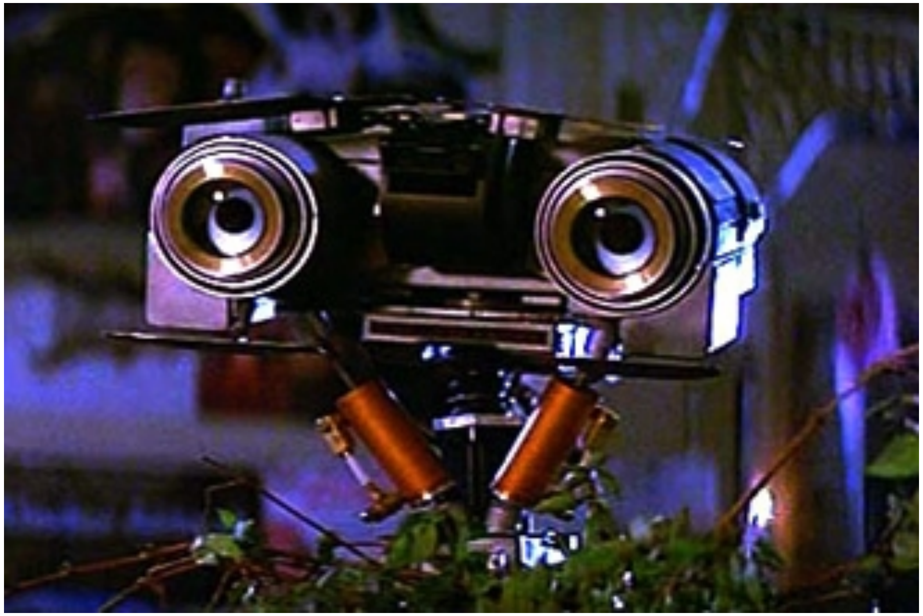
Yong Jae Lee
Assistant Professor
CS, UC Davis

Plan for today

- Topic overview
- Introductions
- Course overview
  - Logistics and requirements

What is Computer Vision?

Computer Vision



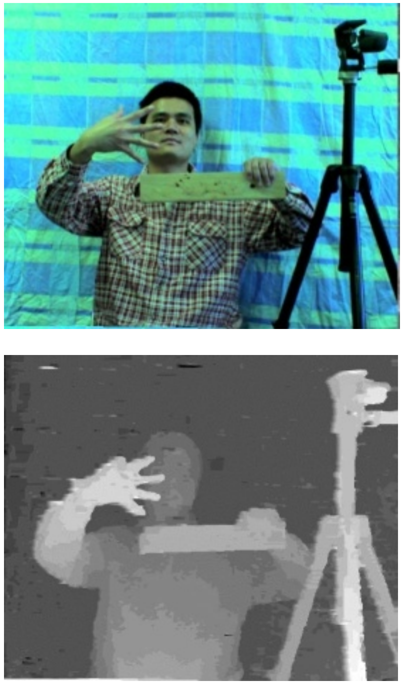
Enable machines to "see" the visual world as we do

Computer Vision

- Automatic understanding of images and video
    1. Computing properties of the 3D world from visual data *(measurement)*

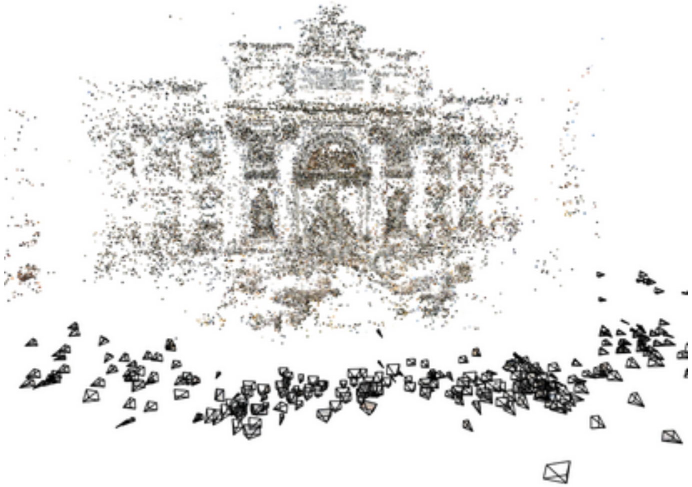
1. Vision for measurement

Real-time stereo



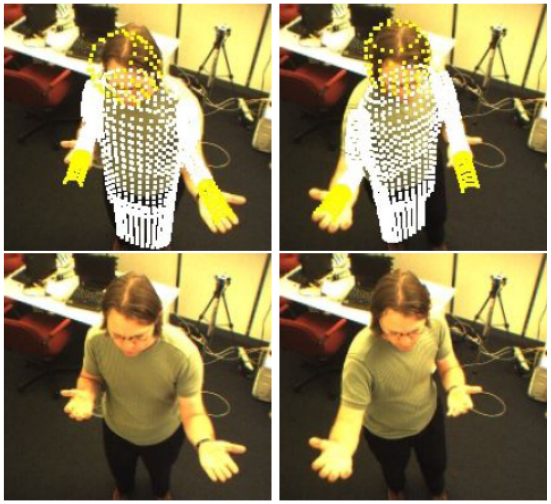
Wang et al.

Structure from motion



Snavely et al.

Tracking



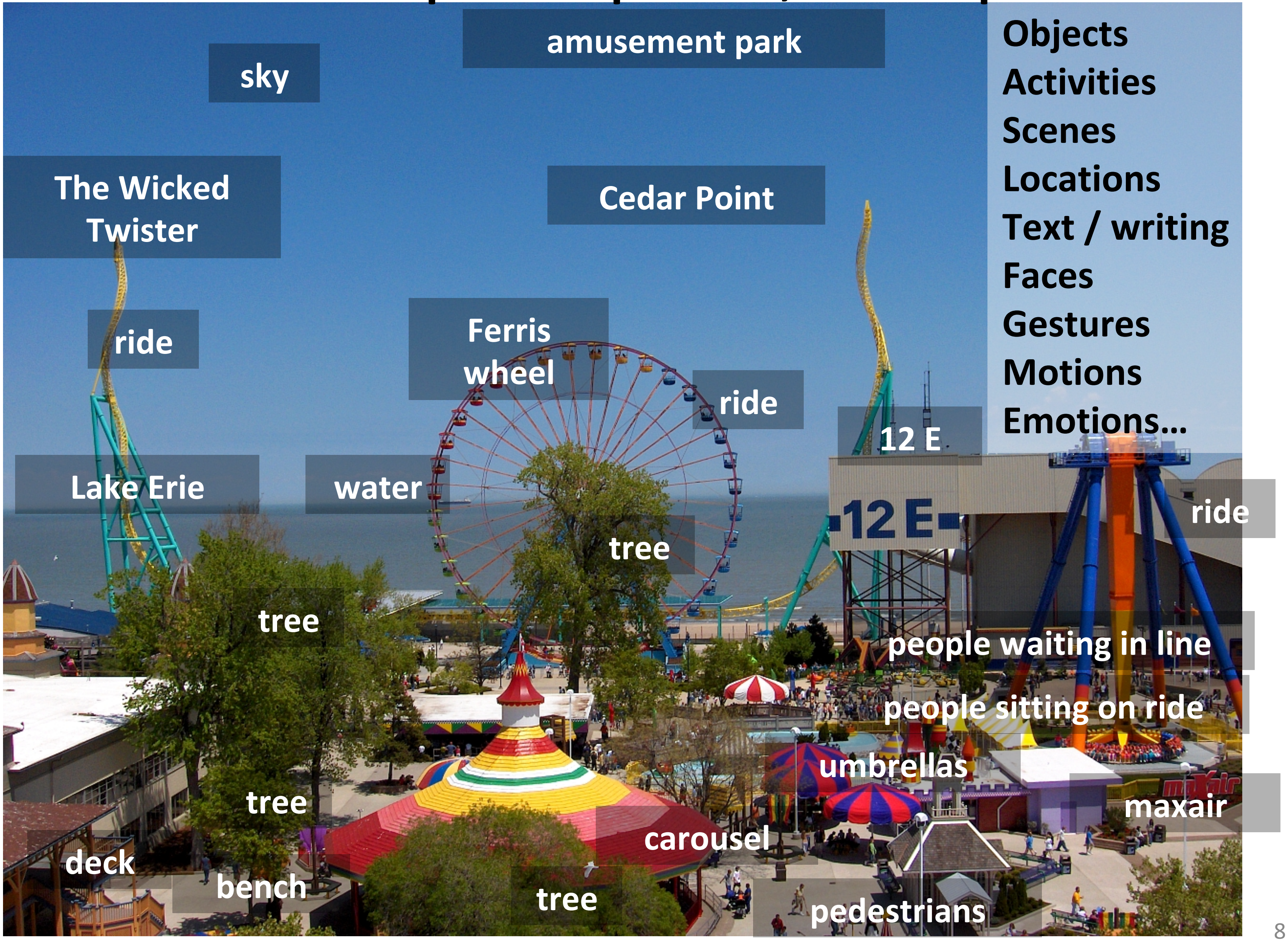
Demirdjian et al.

Slide credit: Kristen Grauman

Computer Vision

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

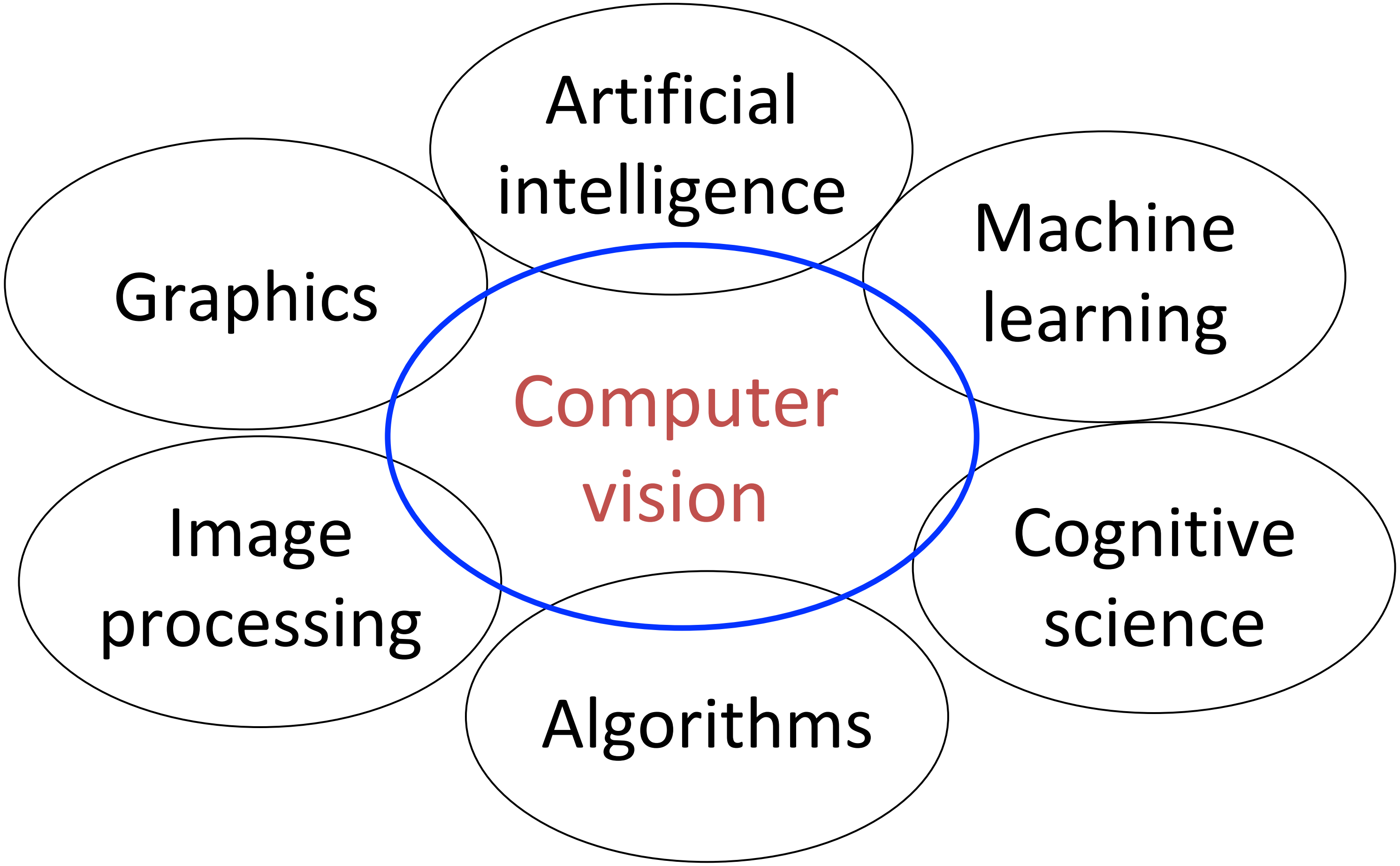
2. Vision for perception, interpretation



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

Slide credit: Kristen Grauman

Related disciplines



Artificial intelligence

Graphics

Machine learning

Computer vision

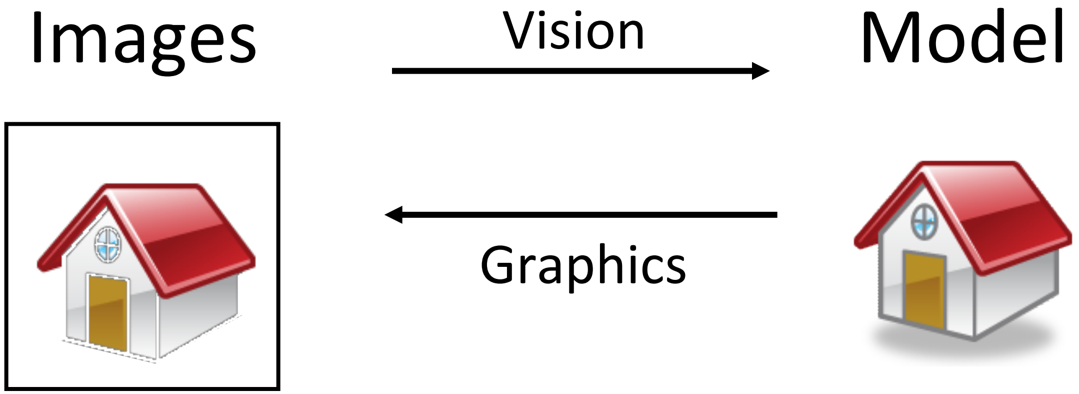
Image processing

Cognitive science

Algorithms

Slide credit: Kristen Grauman

Vision and graphics



Images

Vision

Model

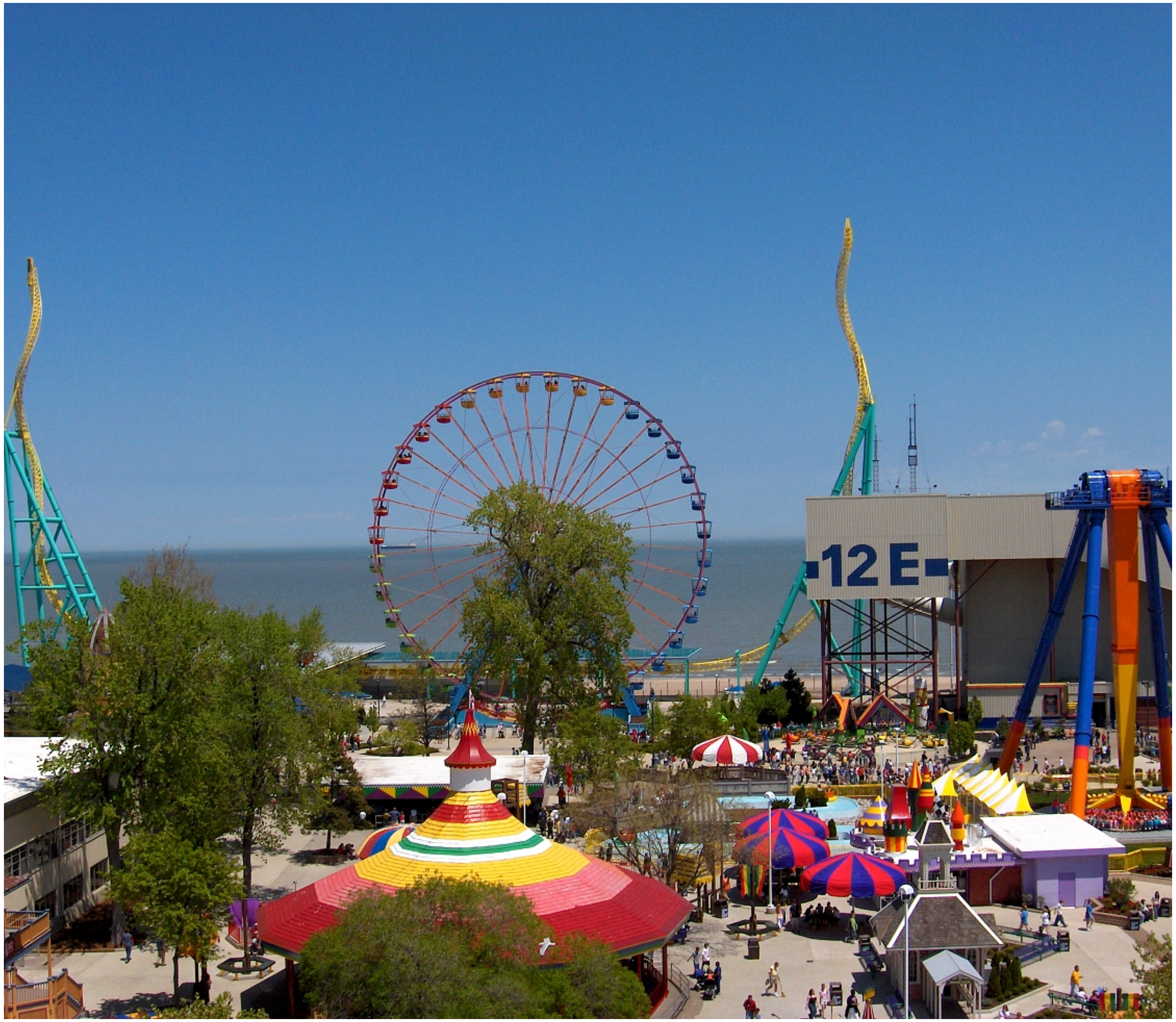
Graphics

Inverse problems: analysis and synthesis

Slide credit: Kristen Grauman

Why is vision difficult?

What humans see



What computers see

Slide credit: Larry Zitnick

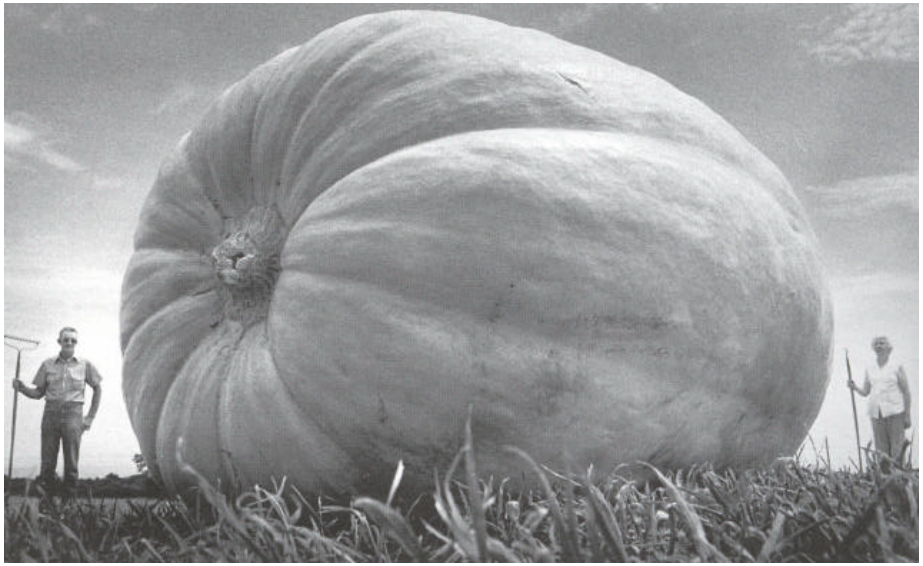
Why is vision difficult?

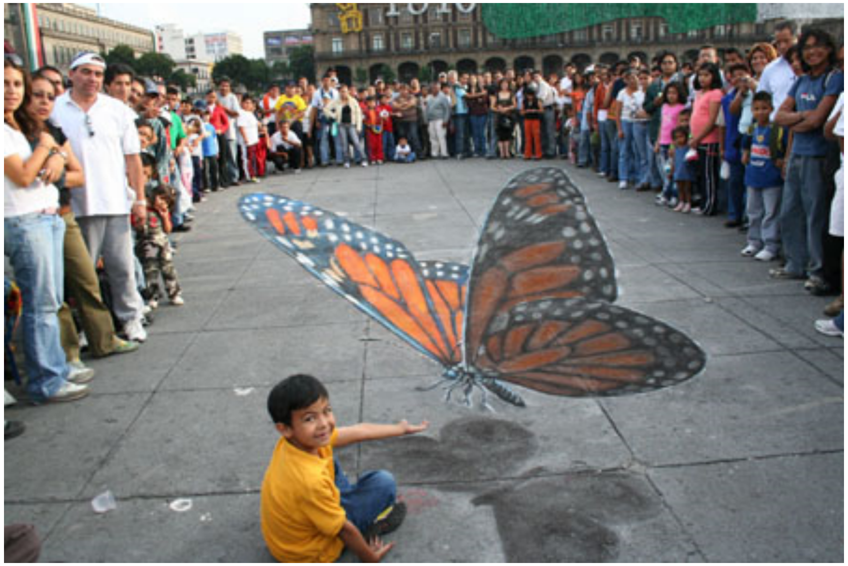
- Ill-posed problem: real world much more complex than what we can measure in images
  - 3D → 2D
- Impossible to literally "invert" image formation process

Slide credit: Kristen Grauman

Challenges: ambiguity

- Many different 3D scenes could have given rise to a particular 2D picture





Slide credit: Svetlana Lazebnik

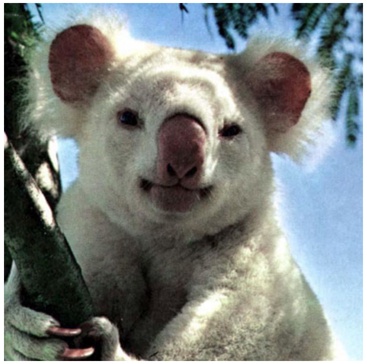
Challenges: variations

Illumination

Object pose

Clutter

Occlusions

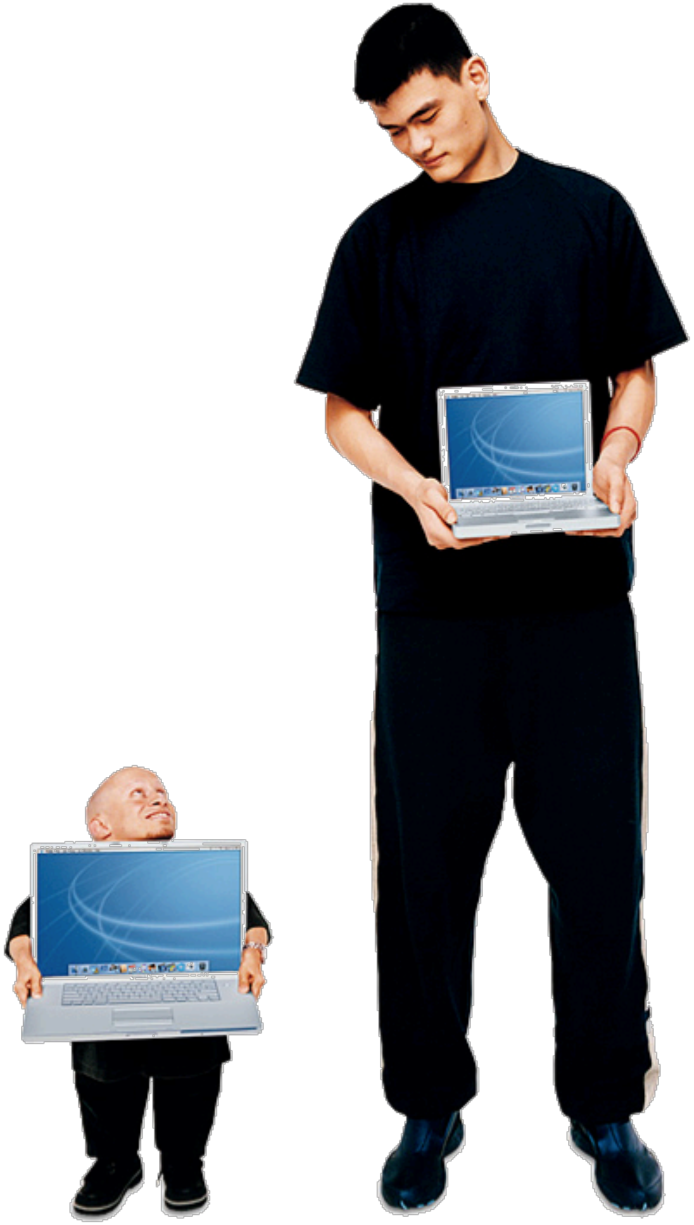


Intra-class appearance

Viewpoint

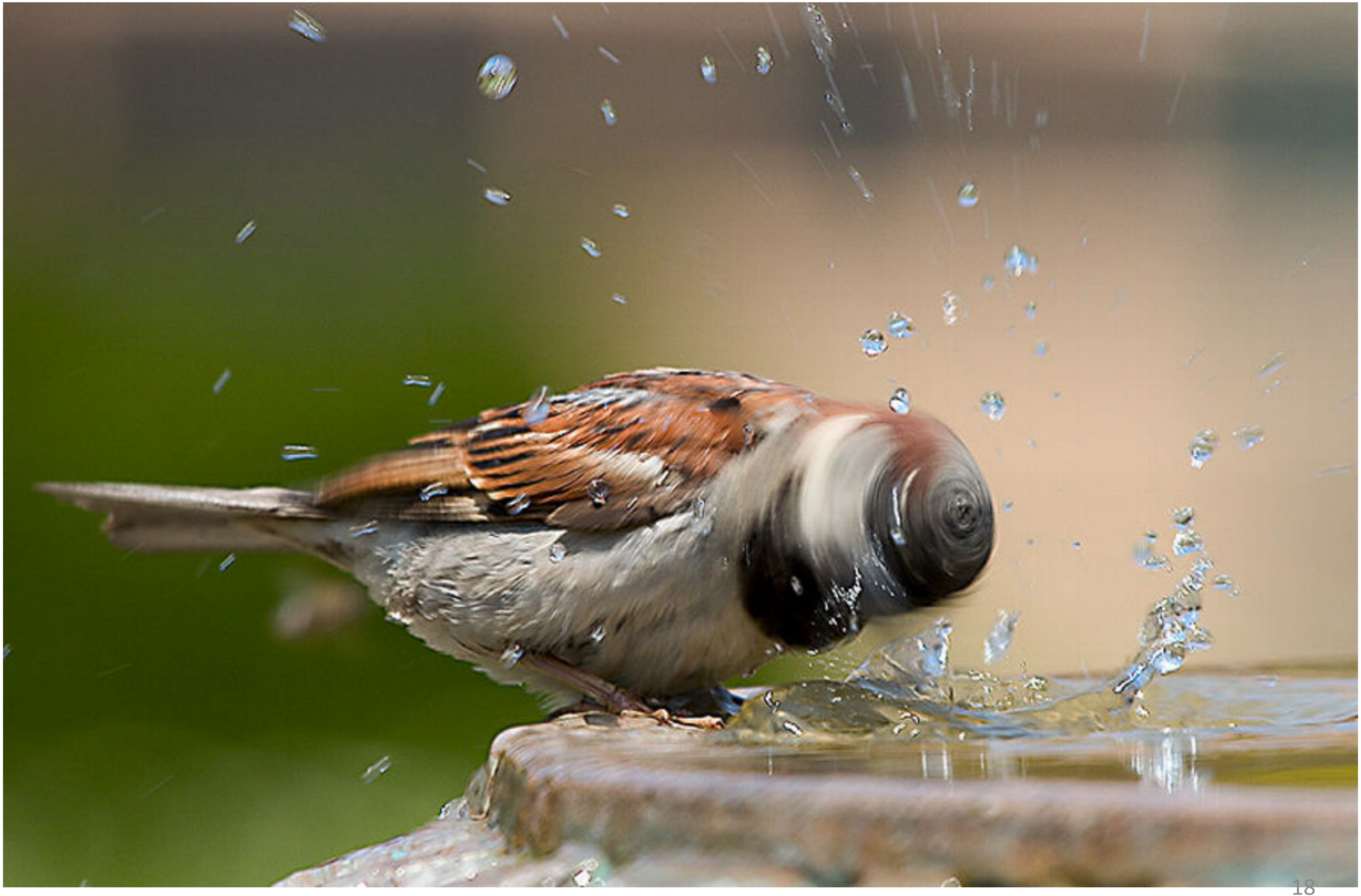
Slide credit: Kristen Grauman

Challenges: scale



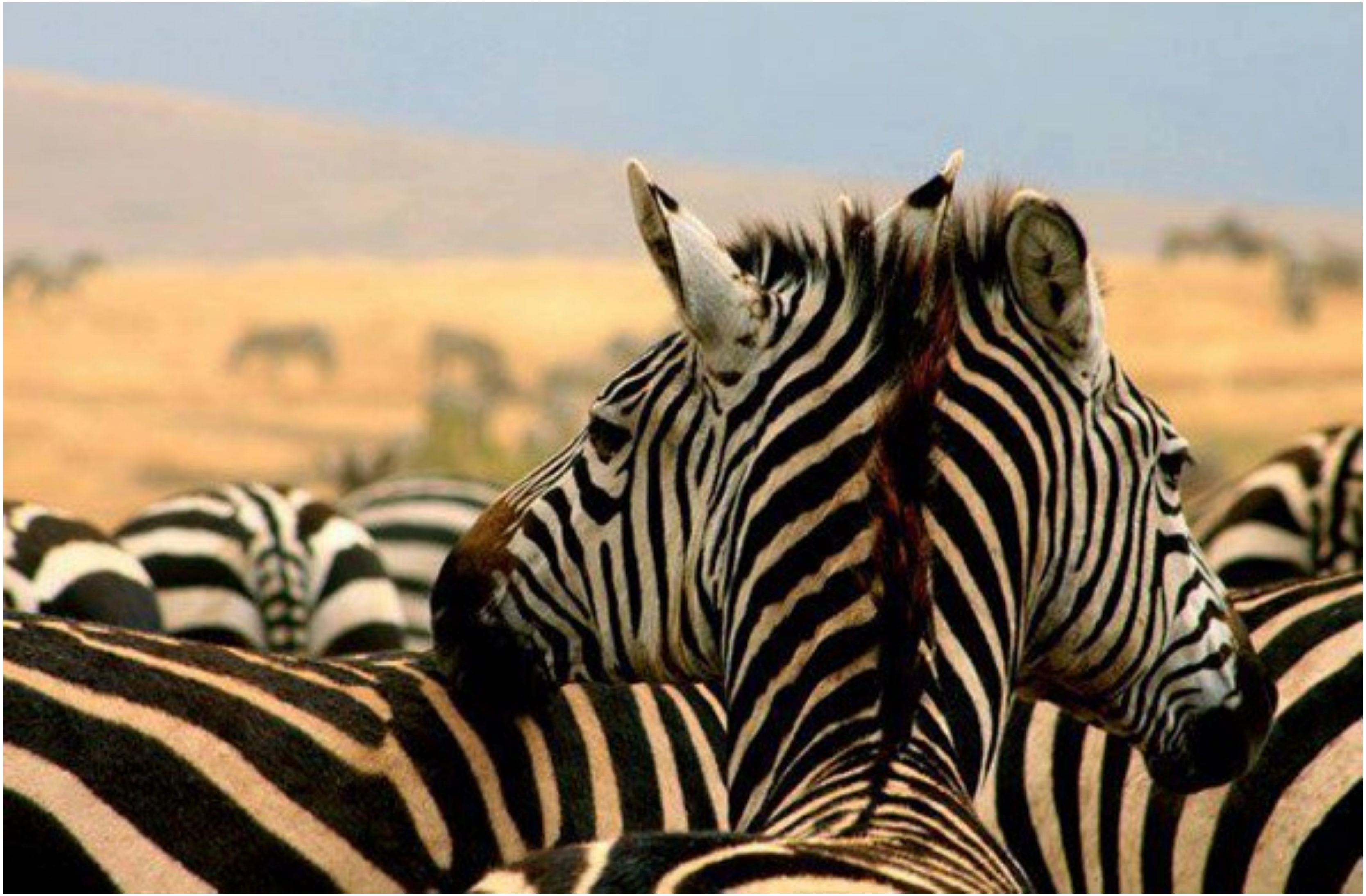
slide credit: Fei-Fei, Fergus, Torralba

Challenges: Motion



slide credit: Svetlana Lazebnik

Challenges: occlusion, clutter

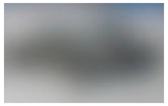


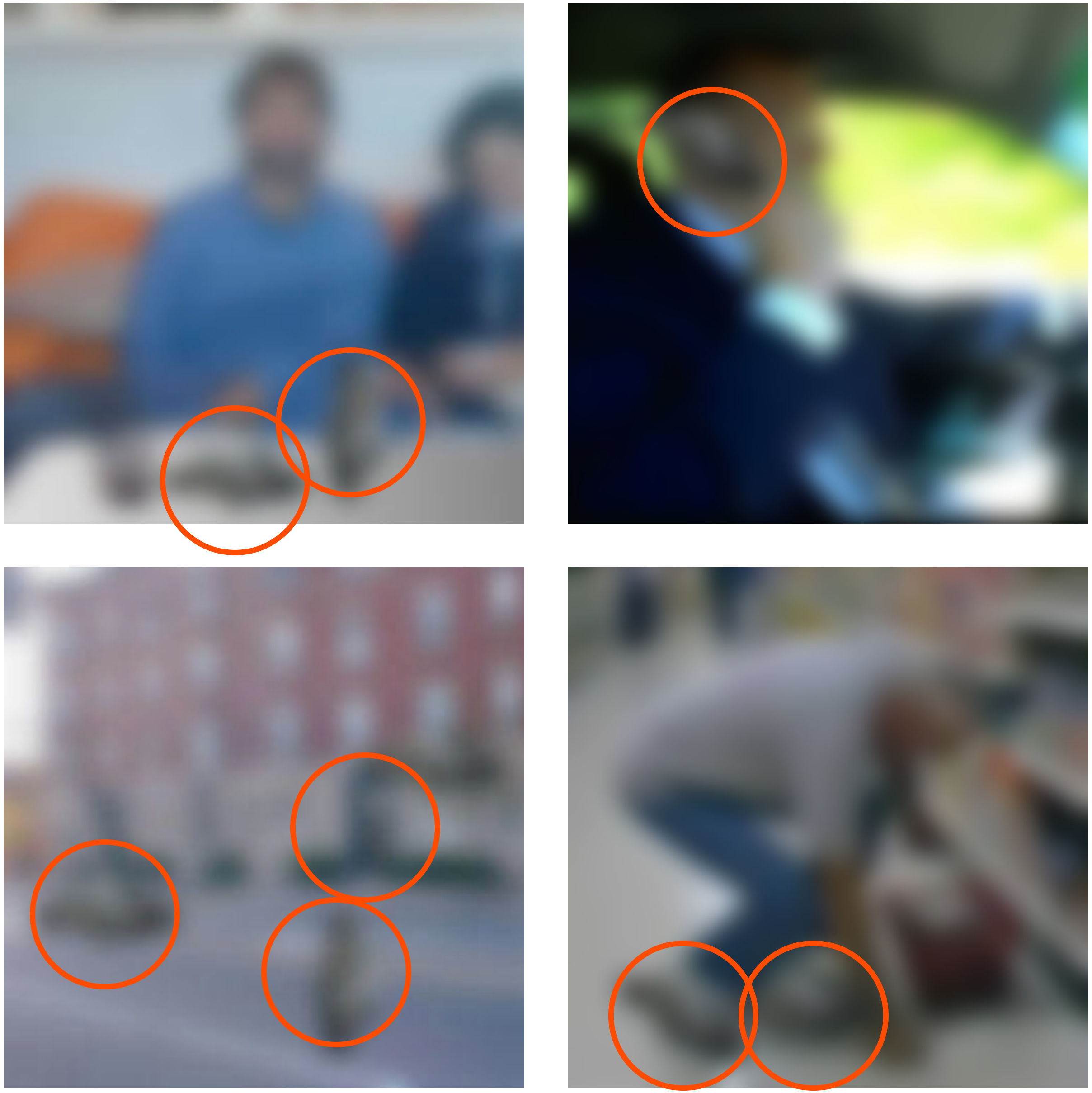
slide credit: Svetlana Lazebnik

Challenges: object intra-class variation

slide credit: Fei-Fei, Fergus, Torralba

Challenges: context and human experience



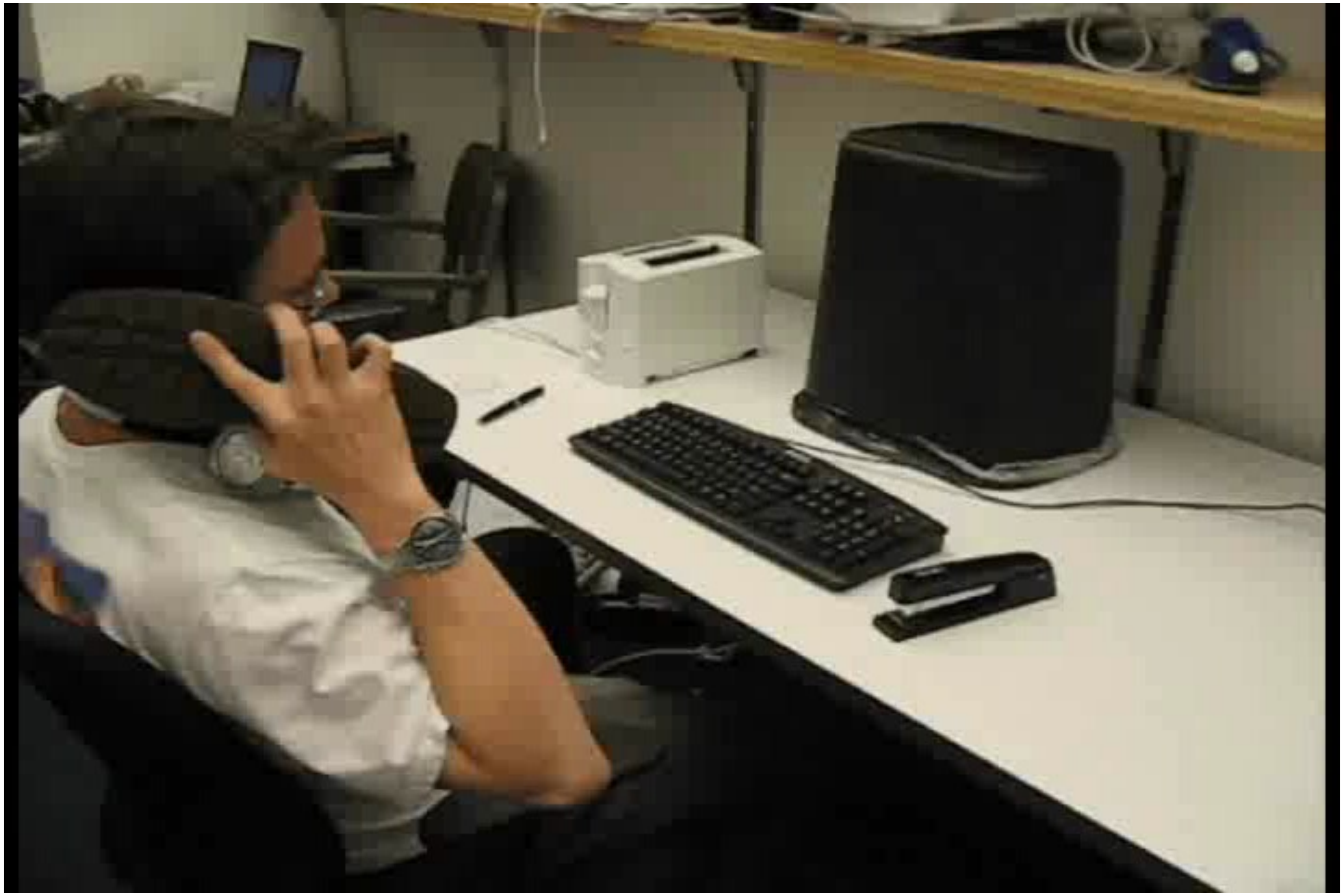


Slide credit: Fei-Fei, Fergus, Torralba

Challenges: context and human experience

Fei Fei Li, Rob Fergus, Antonio Torralba

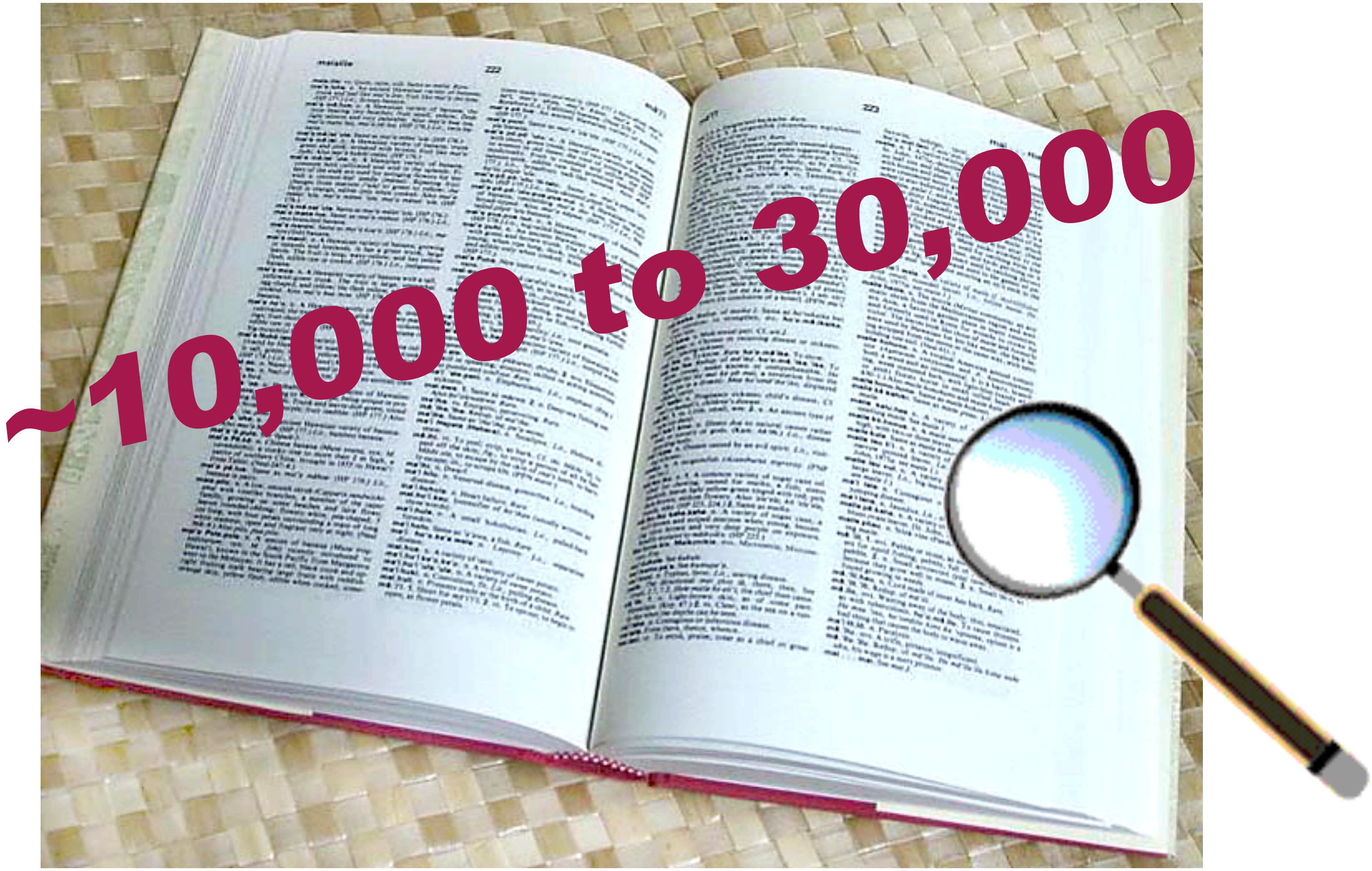
Challenges: context and human experience



Fei Fei Li, Rob Fergus, Antonio Torralba

Challenges: complexity

How many object categories are there?



~10,000 to 30,000

Biederman 1987

Slide credit: Fei-Fei, Fergus, Torralba

Challenges: complexity

flickr

10 billion images

facebook

250 billion images

the simple image sharer
imgur

1 billion images served daily

YouTube

300 hours uploaded per minute

photobucket

10 billion images

From cisco:

Almost 90% of web traffic is visual!

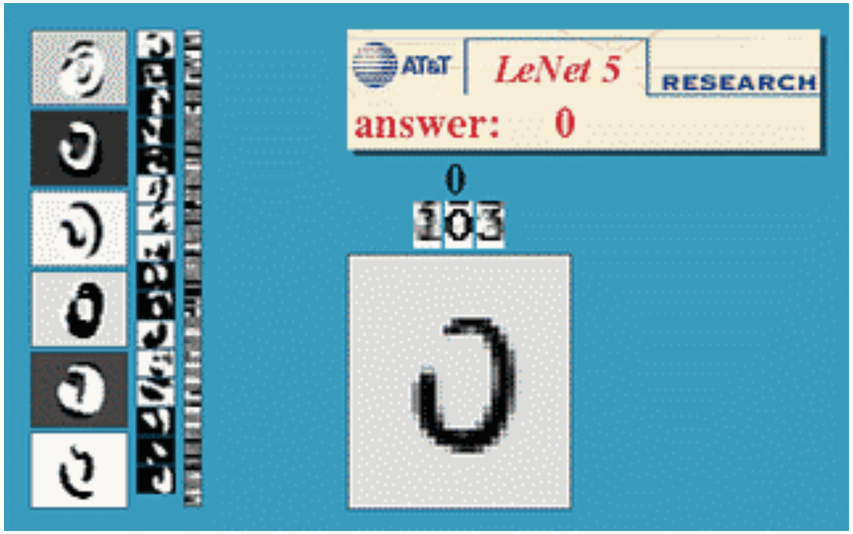
Challenges: complexity

- Thousands to millions of pixels in an image
- About half of the cerebral cortex in primates is devoted to processing visual information [Felleman and van Essen 1991]

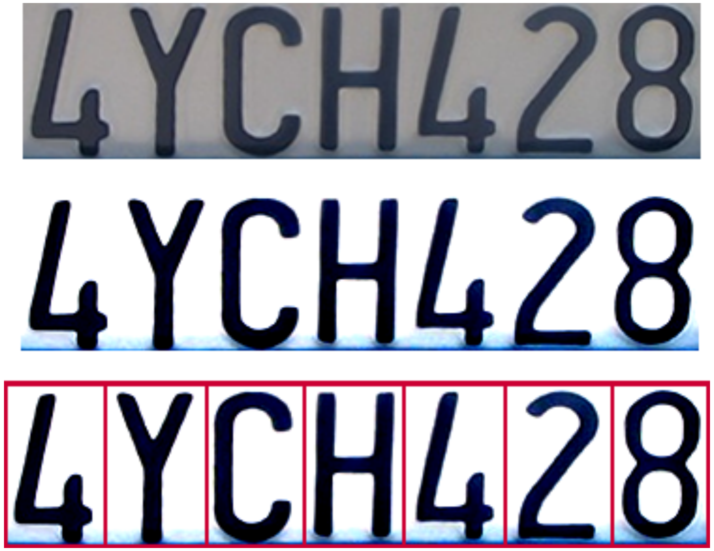
Slide credit: Kristen Grauman

What works well today?

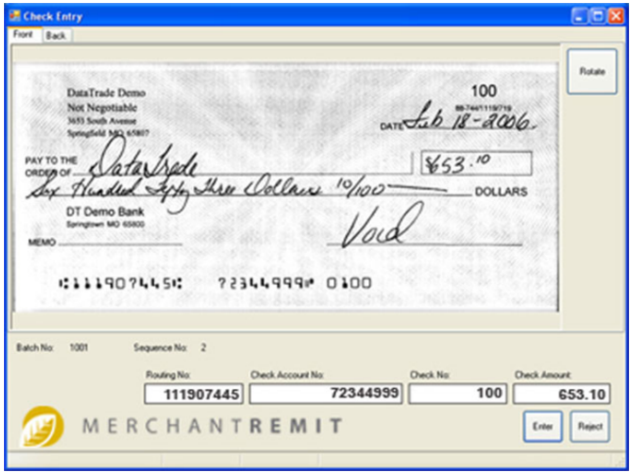
Optical character recognition (OCR)



Digit recognition
yann.lecun.com



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



DataTrade Demo
Not Negotiable
3655 South Avenue
Springfield MO 65807

100

DATE Feb 18-2006

PAY TO THE ORDER OF DataTrade $653.10

Six Hundred Fifty Three Dollars 10/100 DOLLARS

DT Demo Bank
Springtown MO 65800

MEMO

Void

⑆111907445⑆ 72344999⑈ 0100

Batch No: 1001 Sequence No: 2

Routing No:	Check Account No:	Check No:	Check Amount:
111907445	72344999	100	653.10

Automatic check processing

Sudoku grabber
http://sudokugrab.blogspot.com/

Source: S. Seitz, N. Snavely

Image classification and object detection

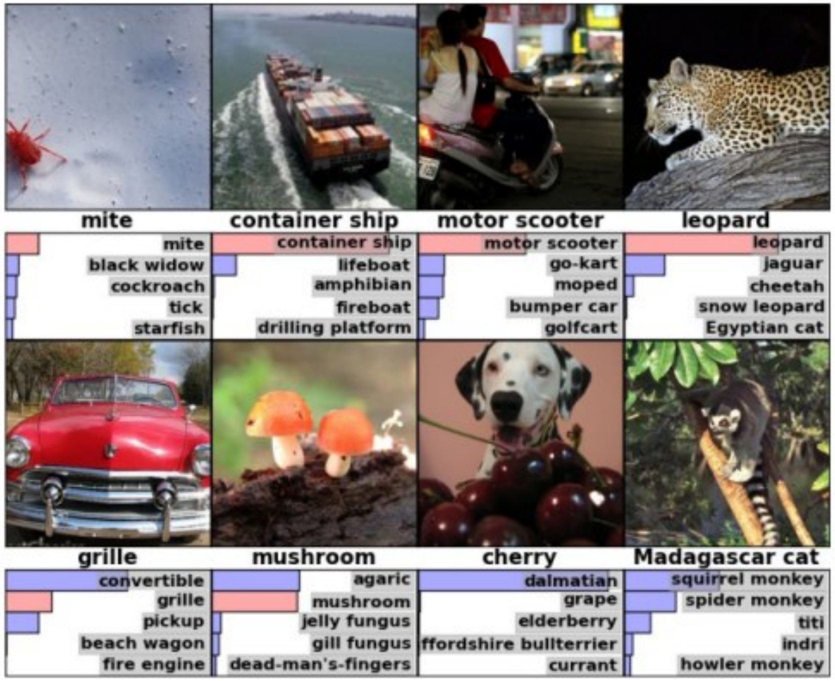
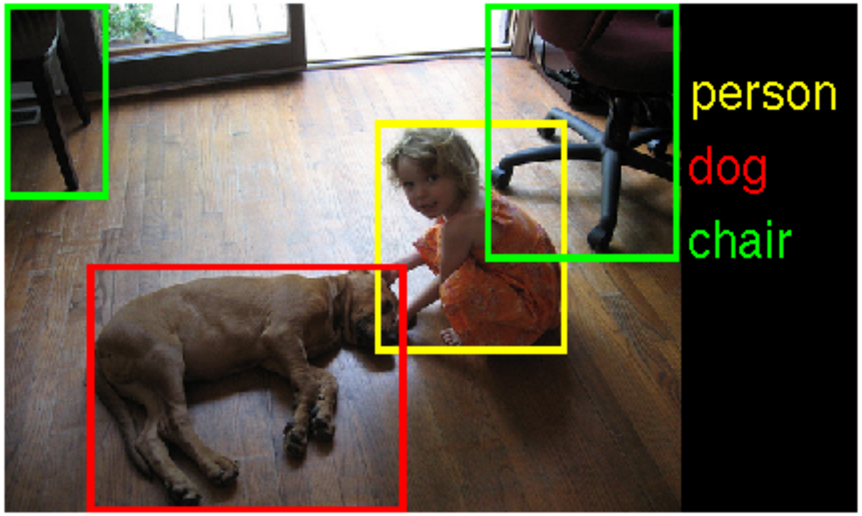
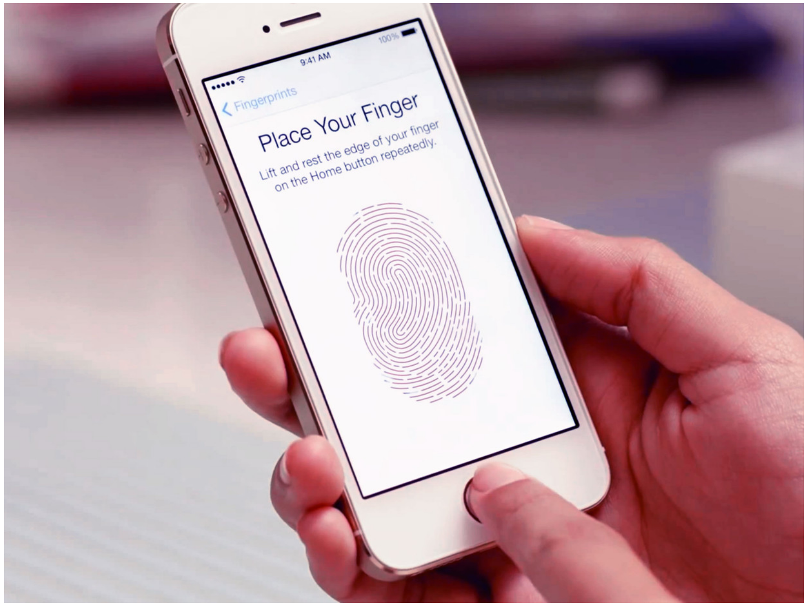


Image classification



Object detection

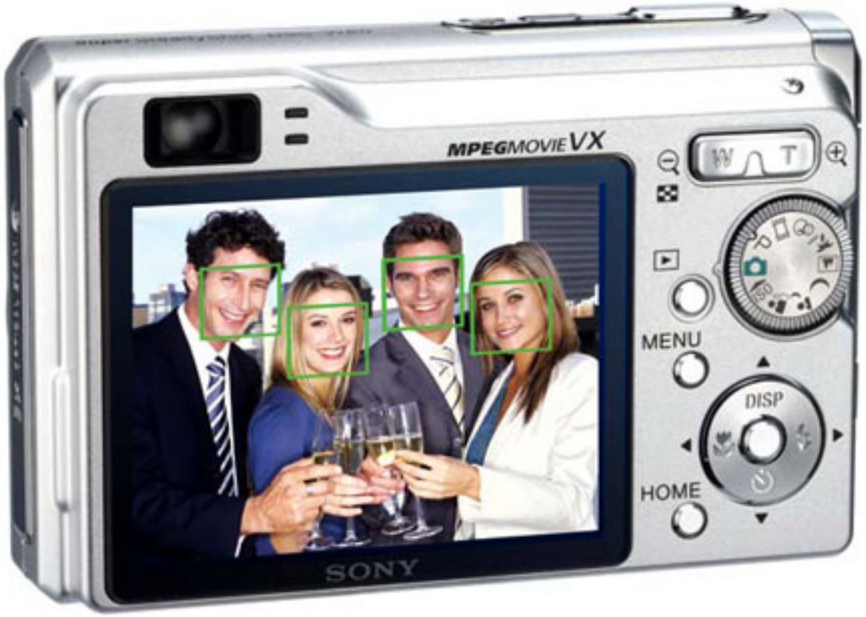
Biometrics


9:41 AM
100%
Fingerprints
Place Your Finger
Lift and rest the edge of your finger on the Home button repeatedly.

Fingerprint scanners

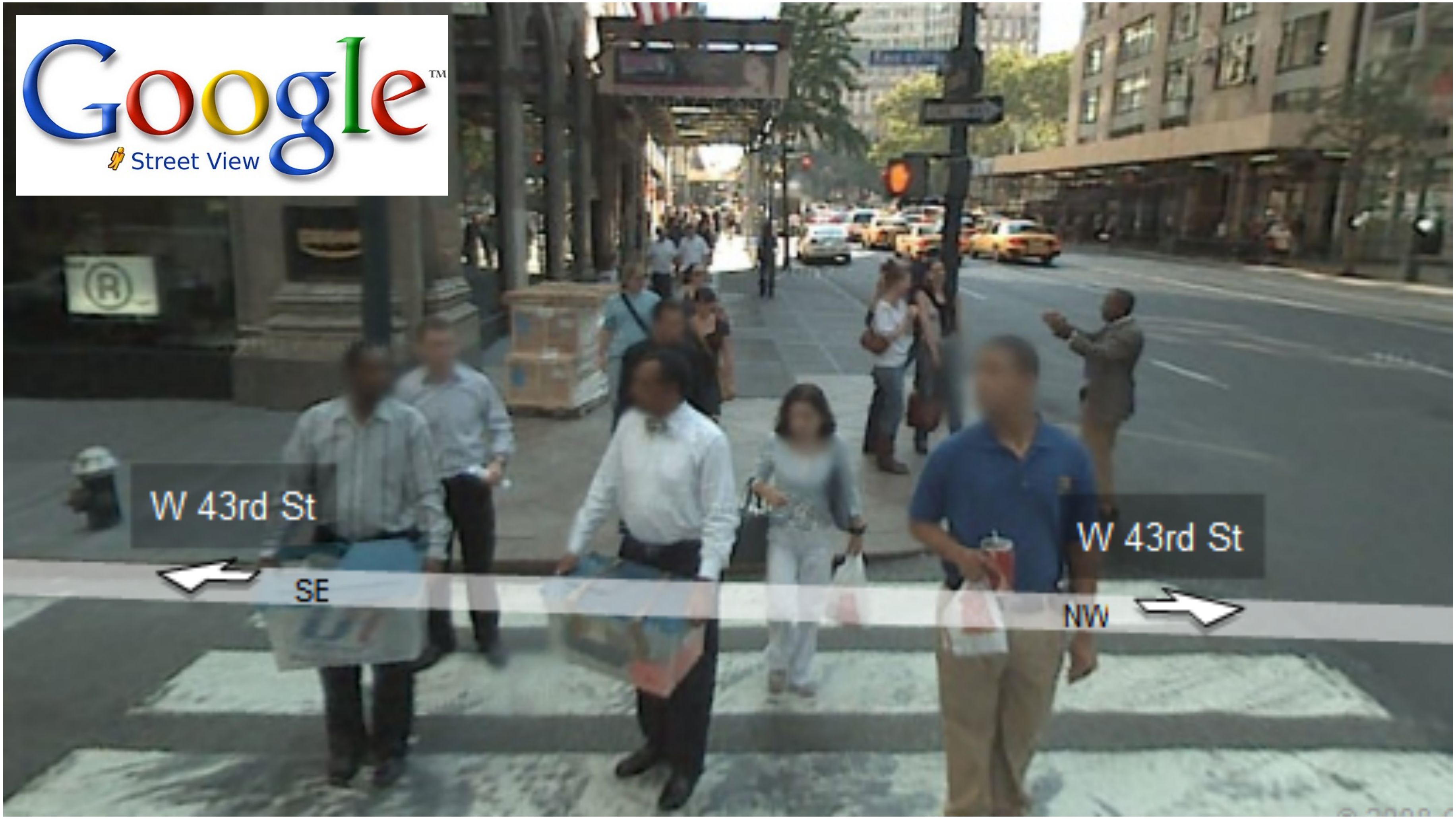
Face recognition systems

Face detection



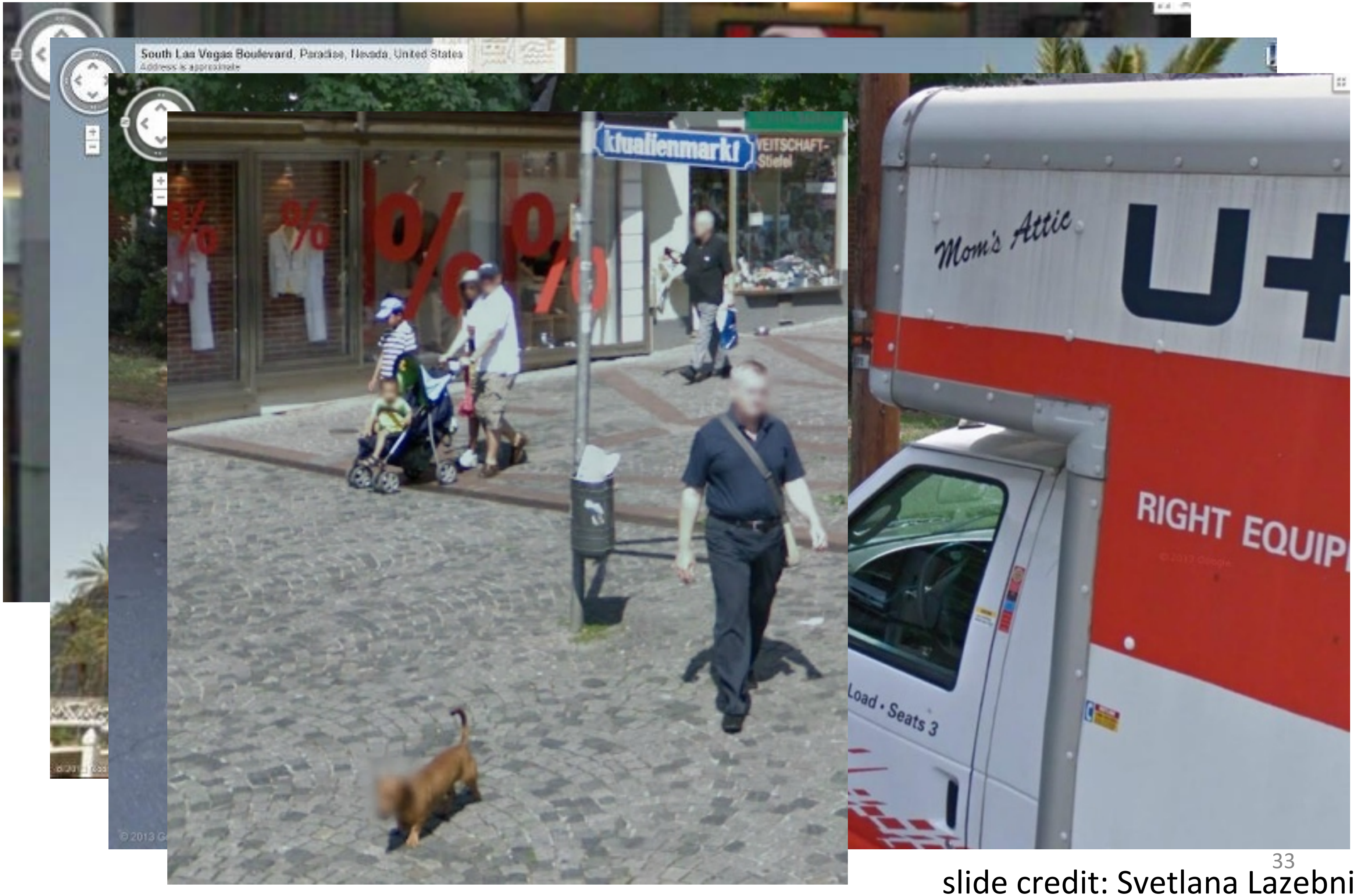
Source: S. Seitz

Face detection for privacy protection



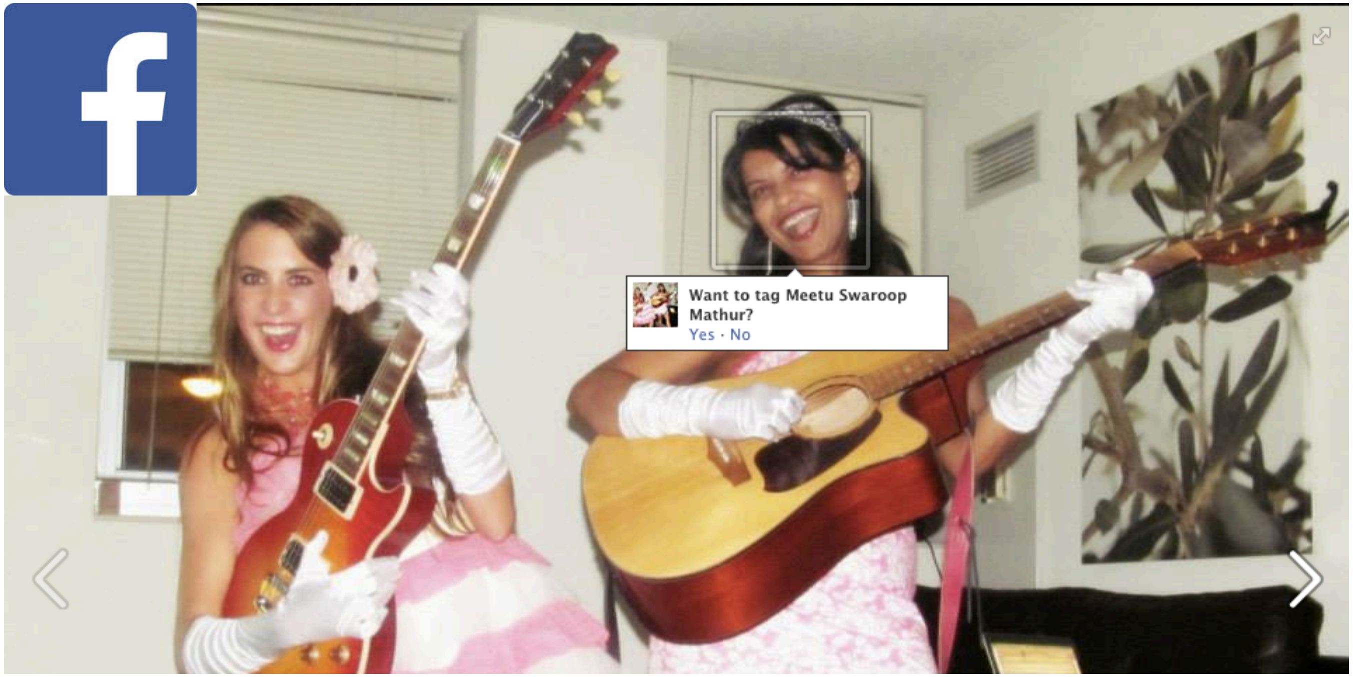
slide credit: Svetlana Lazebnik

Technology gone wild…



slide credit: Svetlana Lazebnik

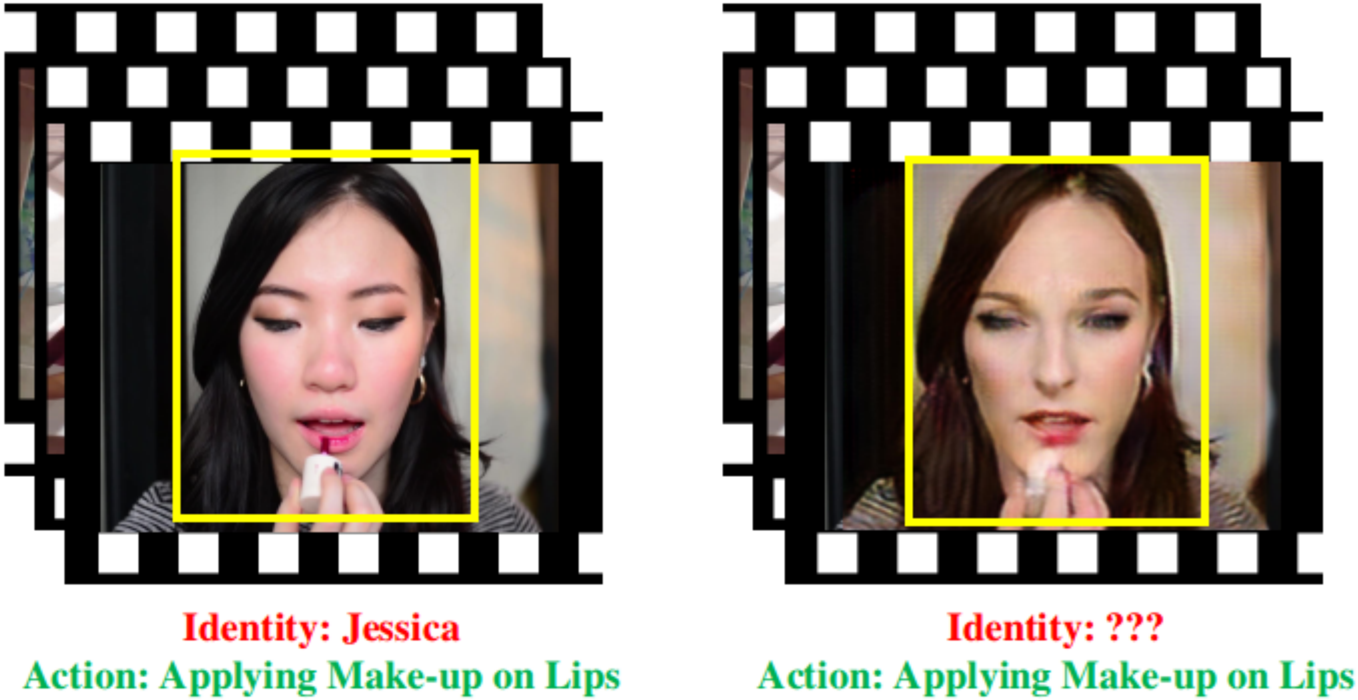
Face recognition



Want to tag Meetu Swaroop Mathur?
Yes · No

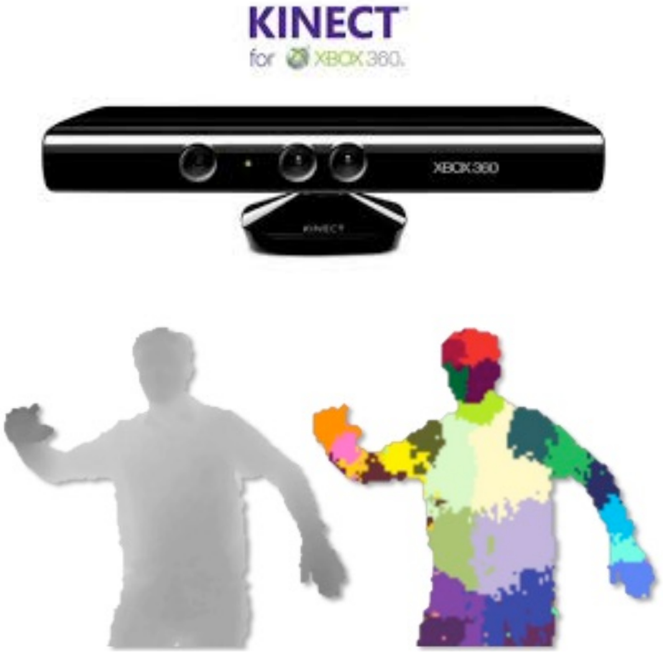
Slide credit: Devi Parikh

Face anonymization


Identity: Jessica
Action: Applying Make-up on Lips
Identity: ???
Action: Applying Make-up on Lips

Zhongzheng Ren, Yong Jae Lee, and Michael Ryoo.
Learning to Anonymize Faces for Privacy Preserving Action Detection. ECCV 2018.

Interactive systems



Shotton et al.

Instance recognition

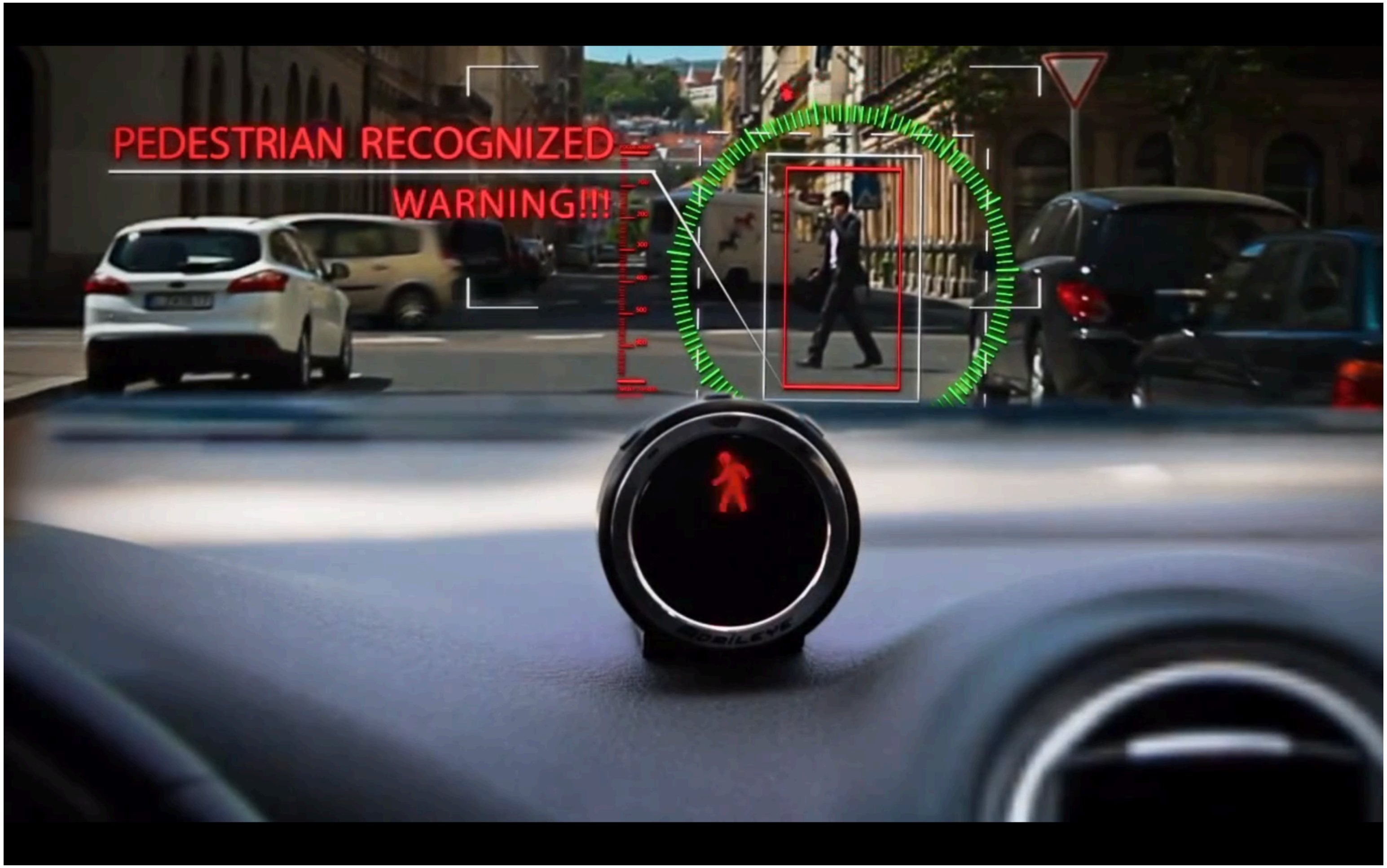
Google Goggles

Use pictures to search the web.

MOTOROLA
18:03
ON THE ROAD THE ORIGINAL SCROLL
On the Road: The Original Scroll
Book · Bargain Price
Abc
ON ROAD THE ORIGINAL SCROL
Text
kerouac on the road
Similar Image
verizon

Slide credit: Devi Parikh

Pedestrian detection



Slide credit: Devi Parikh

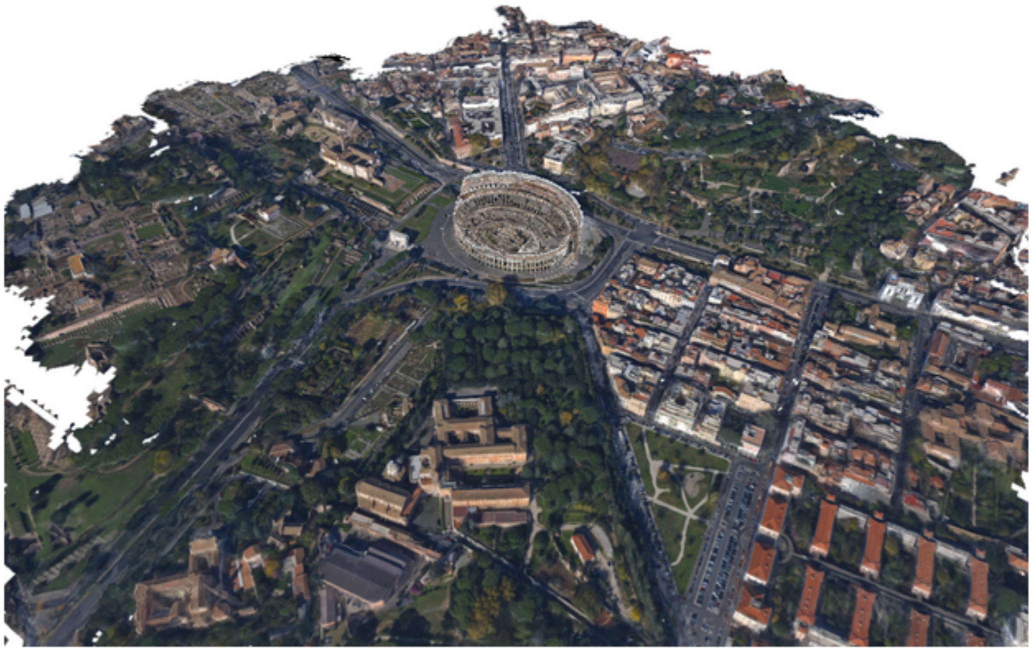
Autonomous agents

Mars rover

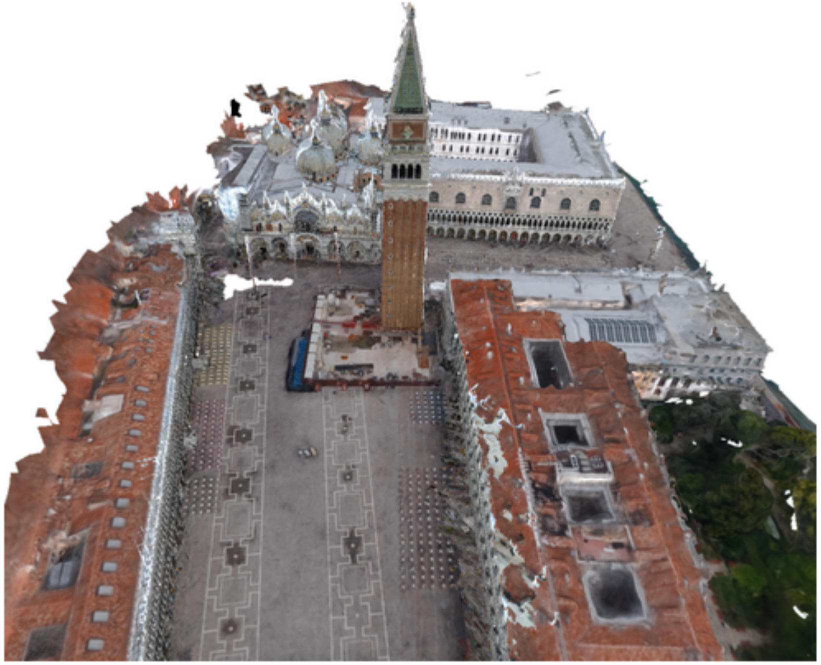
Self-driving car

3D reconstruction from photo collections

Colosseum, Rome, Italy



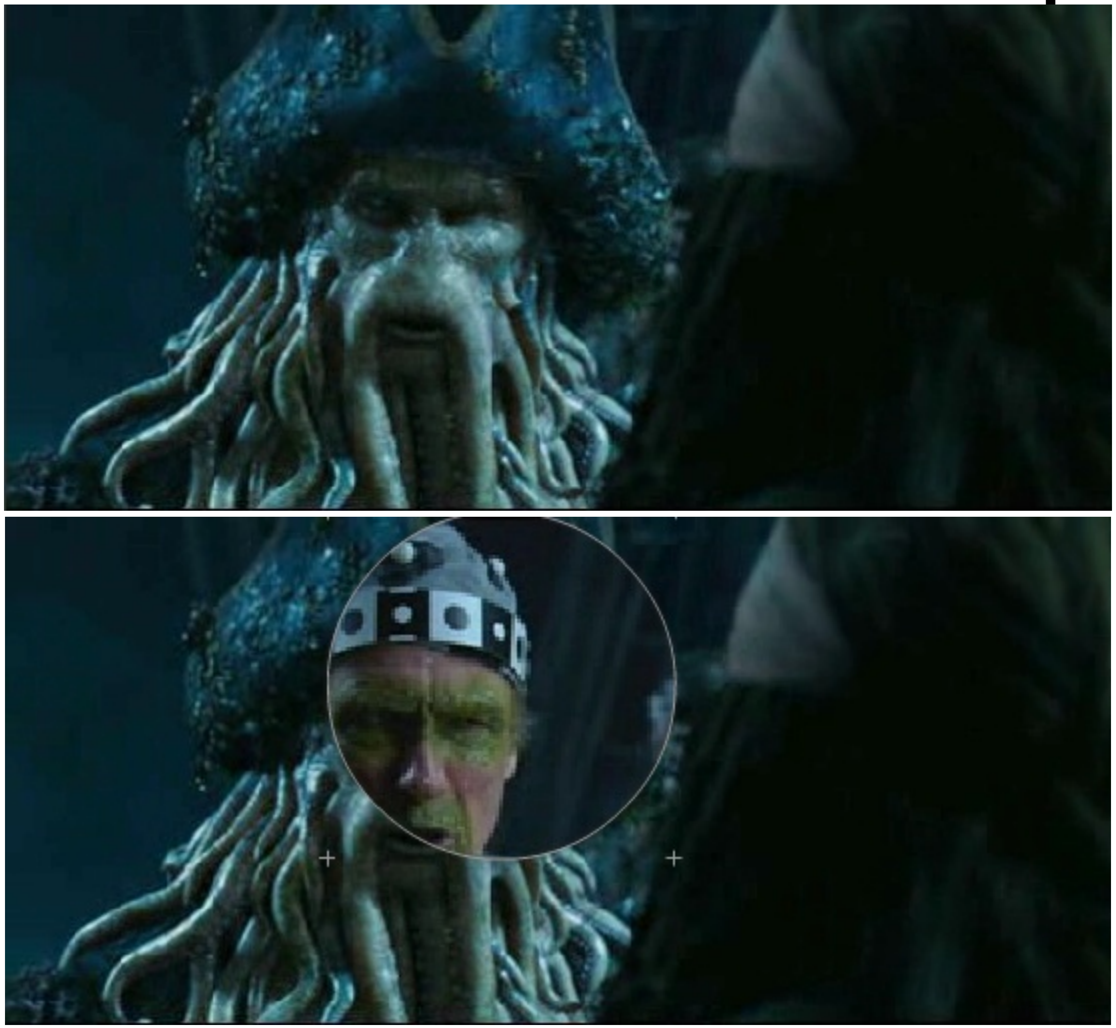
San Marco Square, Venice, Italy



Q. Shan, R. Adams, B. Curless, Y. Furukawa, and S. Seitz,
The Visual Turing Test for Scene Reconstruction, 3DV 2013

slide credit: Svetlana Lazebnik

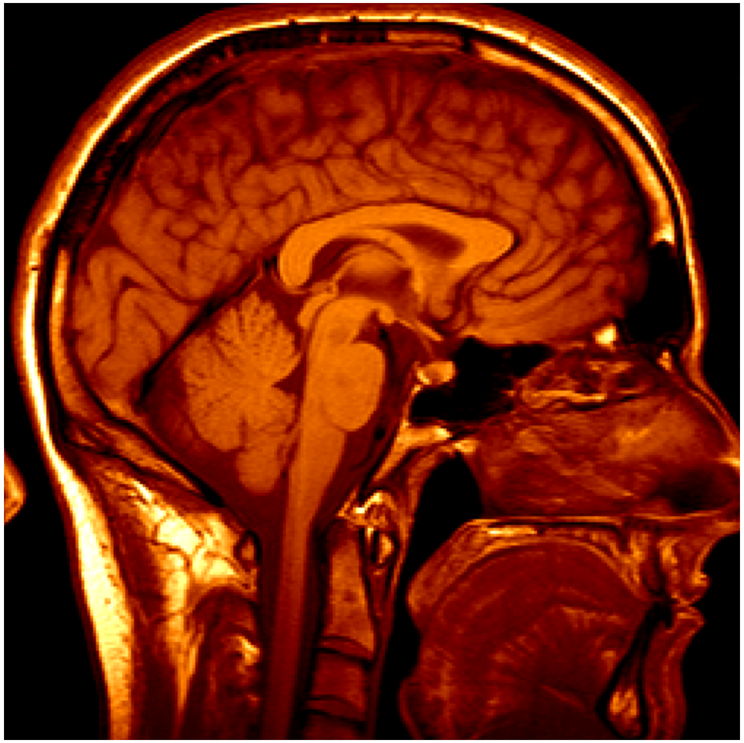
Special effects: motion capture



Pirates of the Carribean, Industrial Light and Magic

Source: S. Seitz

Medical imaging



3D imaging
MRI, CT

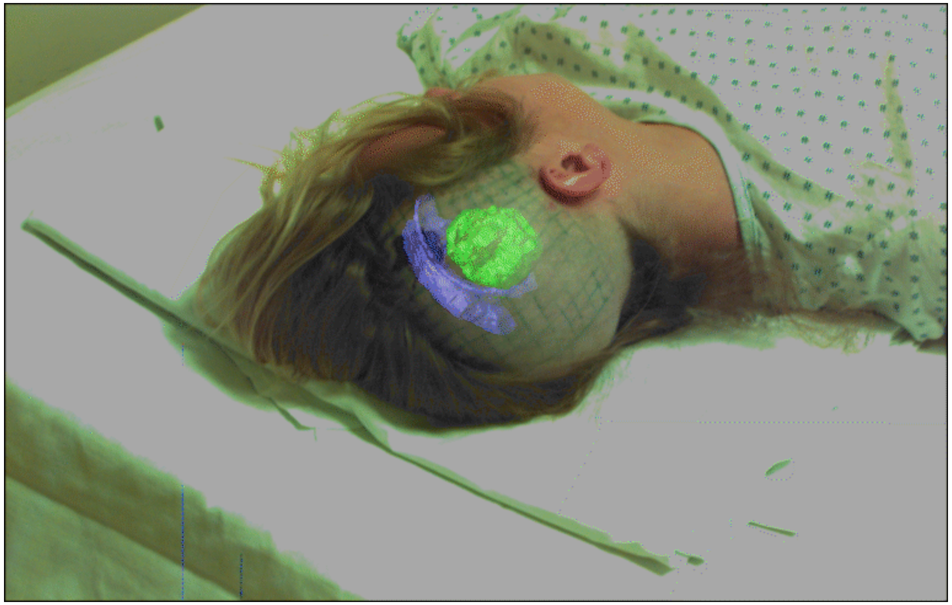
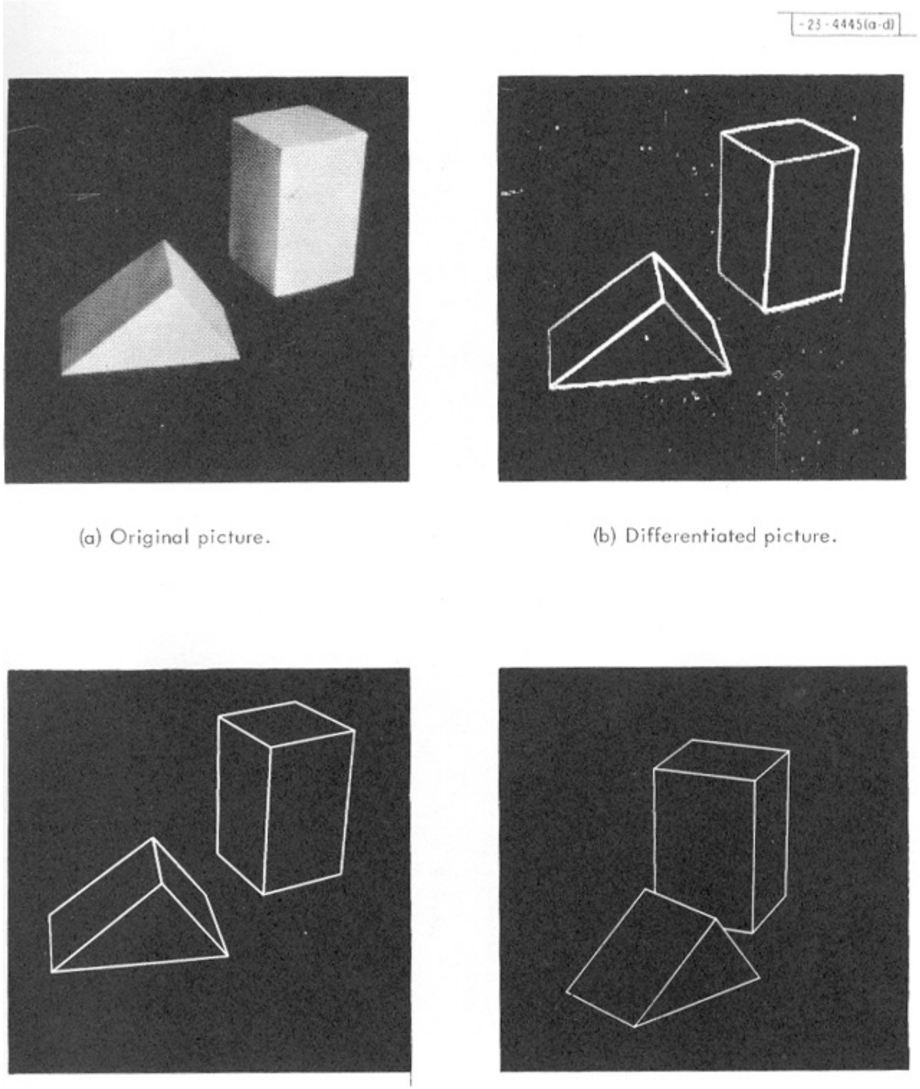


Image guided surgery
Grimson et al., MIT

Source: S. Seitz

Visual data in 1963



(a) Original picture.

(b) Differentiated picture.

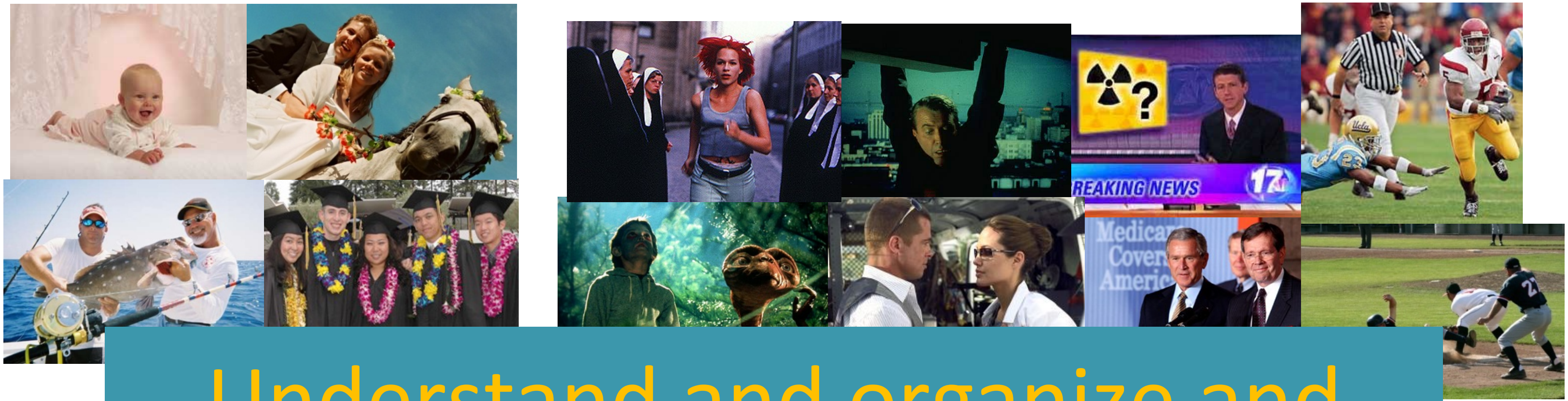
(c) Line drawing.

(d) Rotated view.

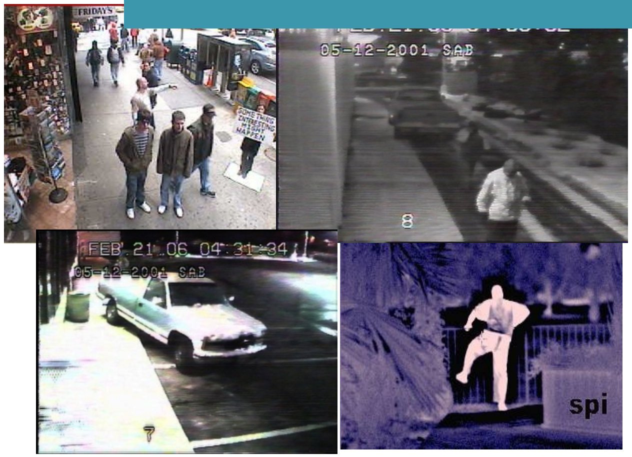
L. G. Roberts, *Machine Perception of Three Dimensional Solids*, Ph.D. thesis, MIT Department of Electrical Engineering, 1963.

Slide credit: Kristen Grauman

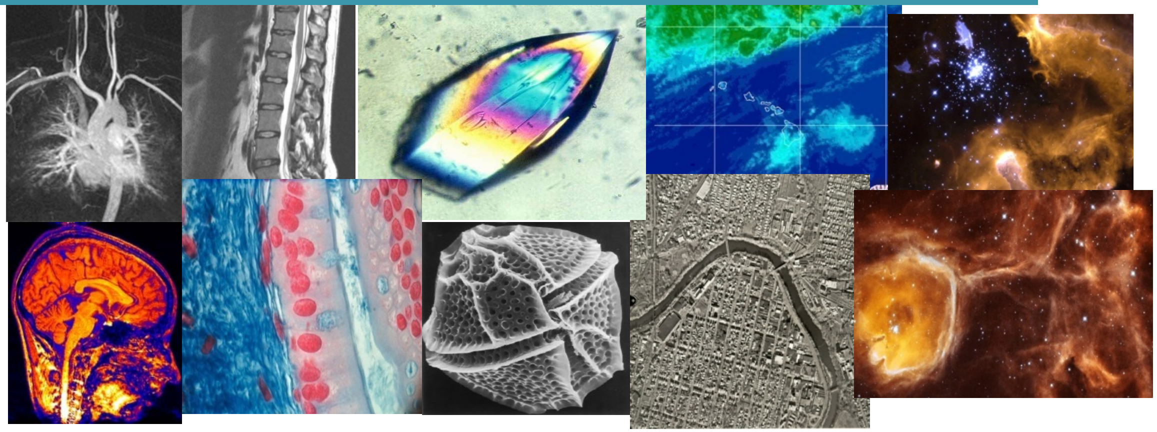
Visual data today



Understand and organize and index all this data!



Surveillance and security



Medical and scientific images

Svetlana Lazebnik

Why vision?

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

Slide credit: Kristen Grauman

Applications

- Law enforcement / Surveillance
- Robotics
- Autonomous driving
- Medical imaging
- Photo organization
- Image search
- E-commerce
- … cell phone cameras, social media, Google Glass, etc.

Slide adapted from Devi Parikh

Summary

- Computer Vision is useful, interesting, and difficult
- A growing and exciting field
- Lots of cool and important applications
- New teams in existing companies, startups, etc.

Plan for today

- Topic overview
- **Introductions**
- Course overview
  - Logistics and requirements

Introductions

- Instructors
  - Yong Jae Lee (first half of the quarter)
  - yongjaelee@ucdavis.edu
  - Assistant Professor in CS, UC Davis since July 2014

  - Krishna Kumar Singh (second half of the quarter)
  - krsingh@ucdavis.edu
  - PhD student in CS, will be research scientist at Adobe Research starting July 2020

  - Research areas: computer vision and machine learning

Introductions

- TAs:
  - Utkarsh Ojha
  - ojha@ucdavis.edu
  - PhD student in CS

  - Haotian Liu
  - lhtliu@ucdavis.edu
  - PhD student in CS

  - Yuheng Li
  - yhnli@ucdavis.edu
  - MS student in CS

This course

- ECS 174 (4-units)

- Lecture: Tues & Thurs 4:40-6:00 pm

- Discussion section: Fri 1:10-2 pm

- Office hours:
  - Yong Jae/Krishna: Mon 10 am - 12 pm
  - Yuheng: Tues 10 am - 12 pm
  - Haotian: Wed 10 am - 12 pm
  - Utkarsh: Thurs 10 am - 12 pm

This course

- Course webpage
  https://sites.google.com/a/ucdavis.edu/ecs-174-computer-vision---spring-2020/

- Canvas (assignment submission, grades)
  https://canvas.ucdavis.edu/courses/439841

- Piazza
- http://piazza.com/uc_davis/spring2020/ecs174

Zoom

- Zoom link for lectures: https://ucdaviscoe.zoom.us/j/206207886?pwd=NVNaNkRYNTlnYXRnakhlNWt5VW1nZz09

- Zoom link for discussion: https://ucdaviscoe.zoom.us/j/875025777?pwd=NCtvQ1BOWjFnWXlMdmthaWlZQ2lkZz09

- Zoom link for office hours: https://ucdaviscoe.zoom.us/j/840739812?pwd=d1Z2dEdabjFzL2N2c0VlVWU5aG9lUT09

- Password sent via email, and available on Canvas

Zoom

- All lectures will be recorded, which means your audio and video will also be recorded if you interact during lectures

- We need to record the lecture videos for later viewing since some students may have connectivity issues during the designated lecture times -- thank you for your understanding

- Recordings will be available 1-2 days after the lecture on Canvas

Goals of this course

- Introduction to primary topics in Computer Vision
- Basics and fundamentals
- Practical experience through assignments
- Views of computer vision as a research area

Prerequisites

- Upper-division undergrad course

- Basic knowledge of probability and linear algebra
- Data structures, algorithms
- Programming experience

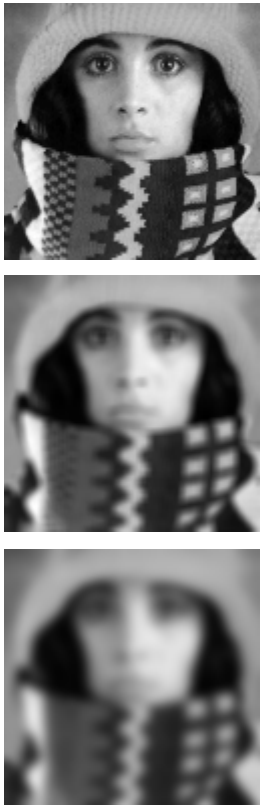
- Experience with image processing or Matlab/ Python will help but is not necessary

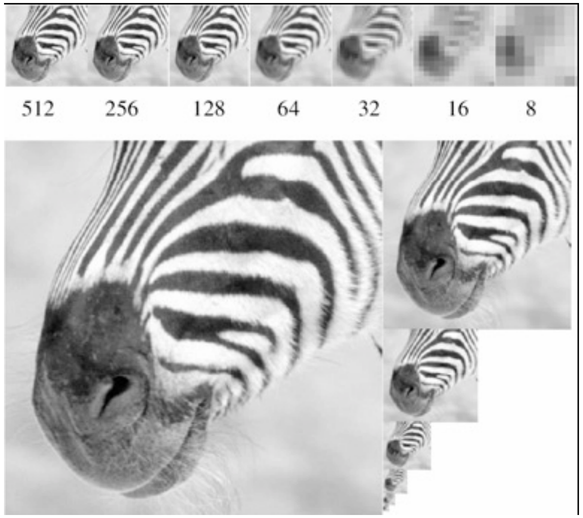
Topics overview

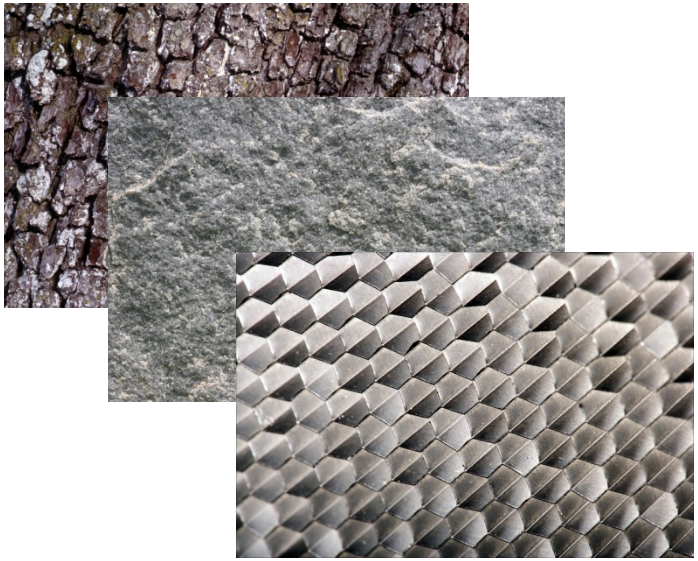
- Features and filters
- Grouping and fitting
- Recognition and learning

Focus is on algorithms, rather than specific systems

Features and filters



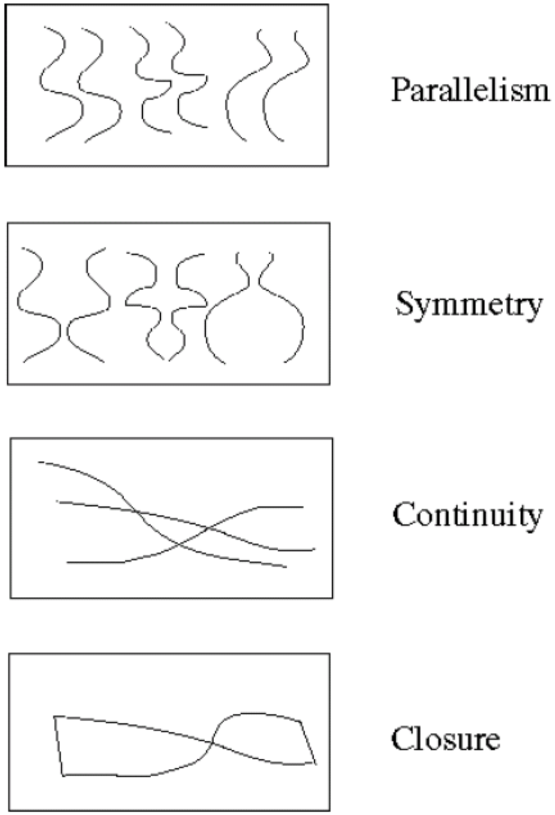


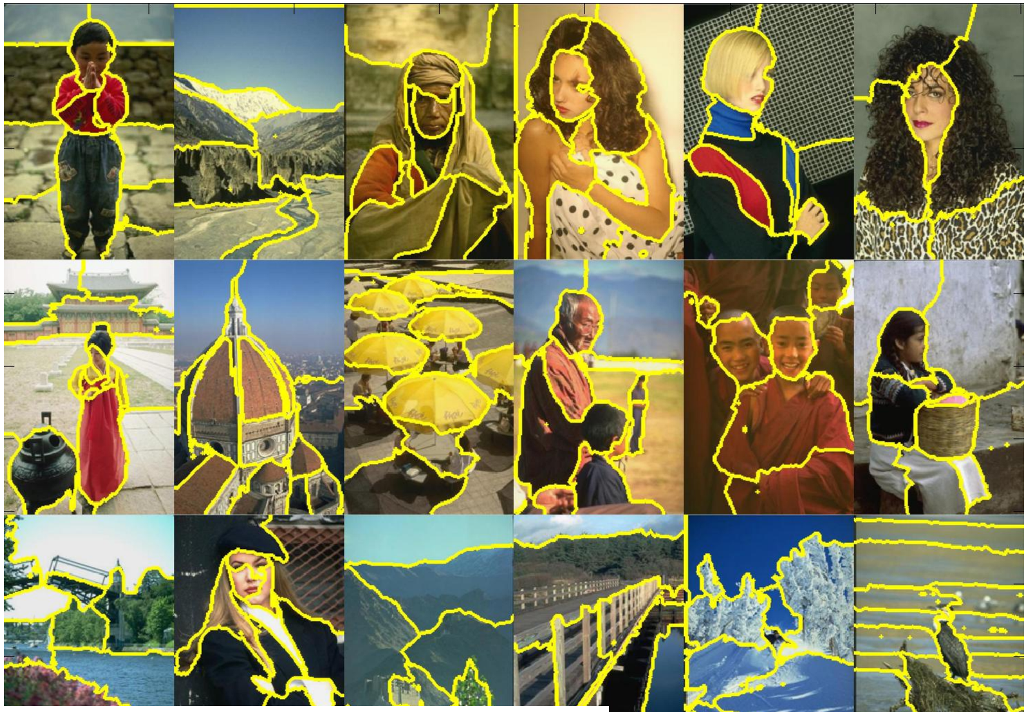


Transforming and describing images; textures, edges

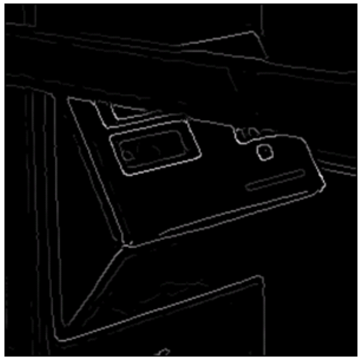
Slide credit: Kristen Grauman

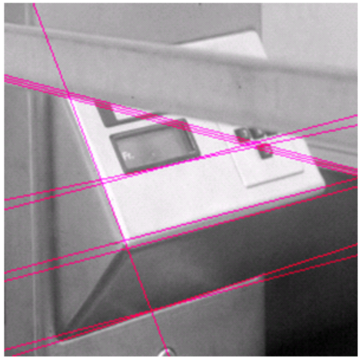
Grouping and fitting

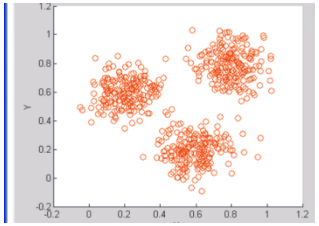




[fig from Shi et al]



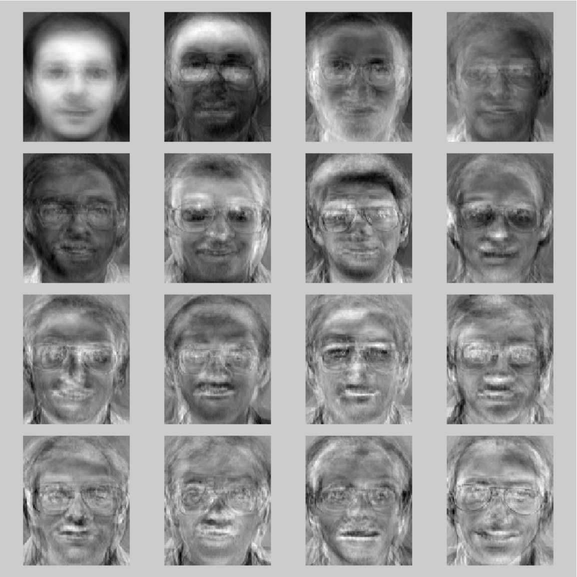


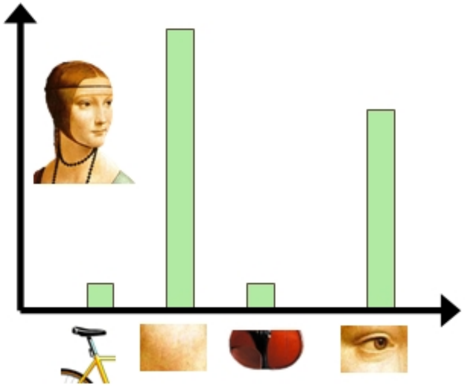


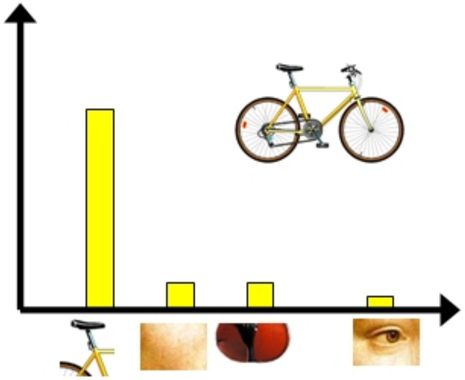
Clustering, segmentation, fitting; what parts belong together?

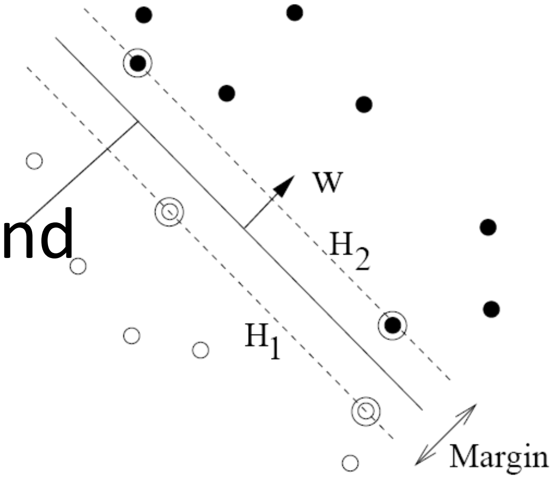
Slide credit: Kristen Grauman

Recognition and learning





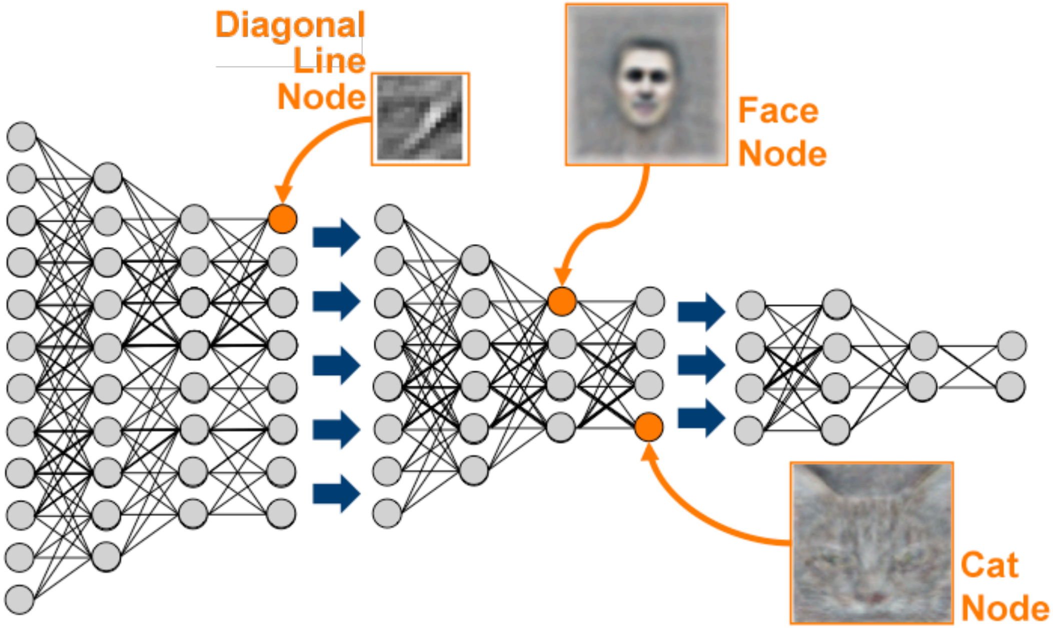


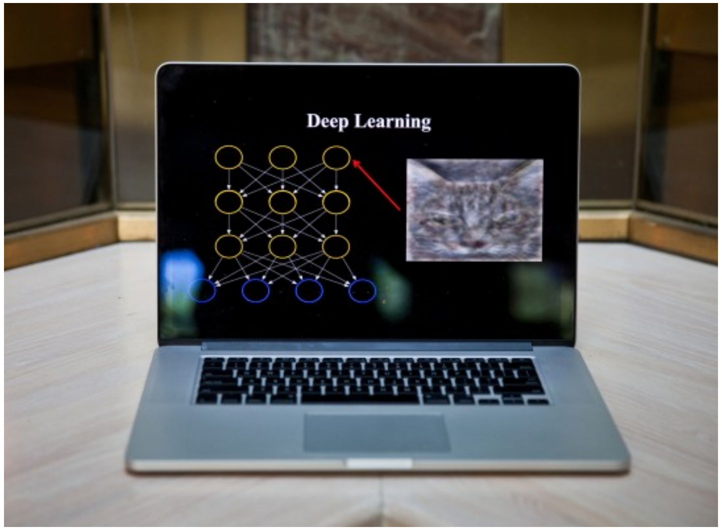


Recognizing objects and categories, learning techniques

Slide credit: Kristen Grauman

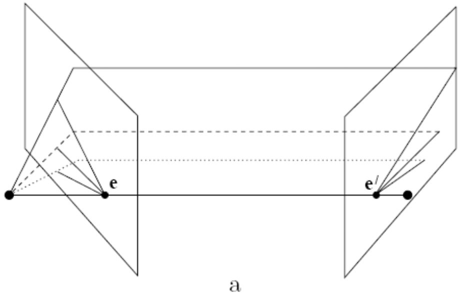
Recognition and learning

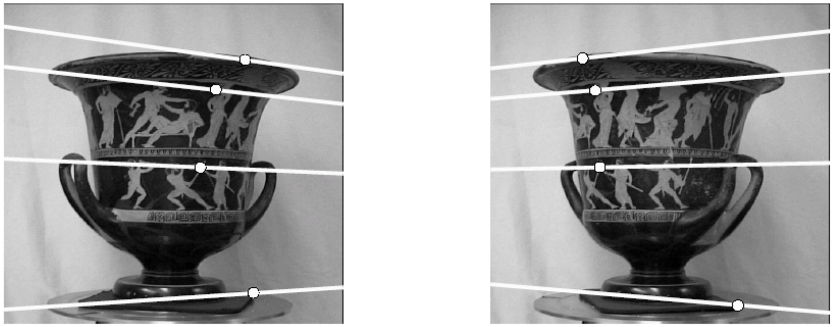




Deep learning

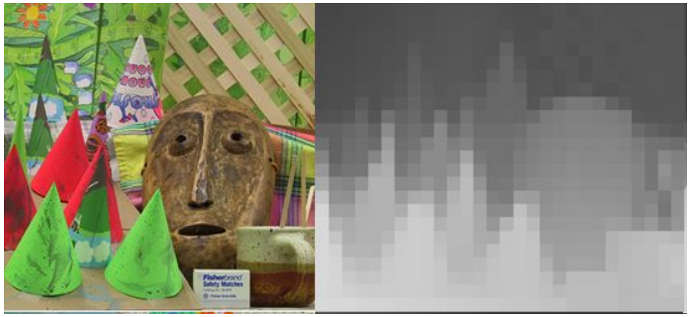
Not covered: Multiple views and motion





Hartley and Zisserman

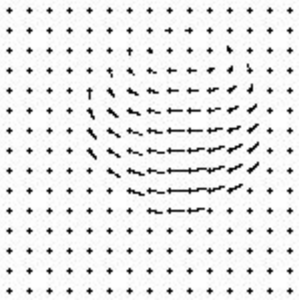
Multi-view geometry, stereo vision

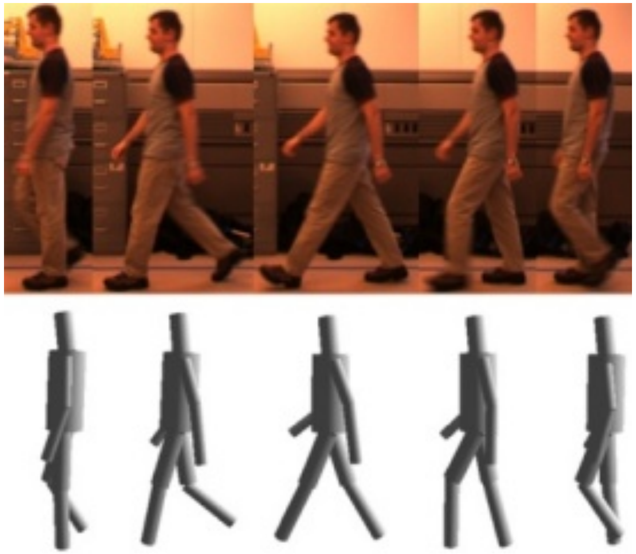


Slide credit: Kristen Grauman

Not covered: Video processing

Tracking objects, video analysis, low level motion, optical flow





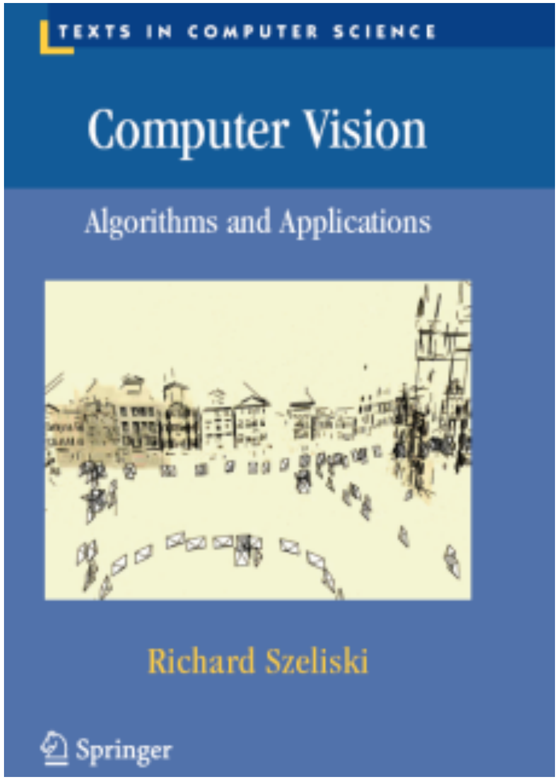
Tomas Izo

Slide credit: Kristen Grauman

Plan for today

- Topic overview
- Introductions
- **Course overview**
  - **Logistics and requirements**

Textbooks



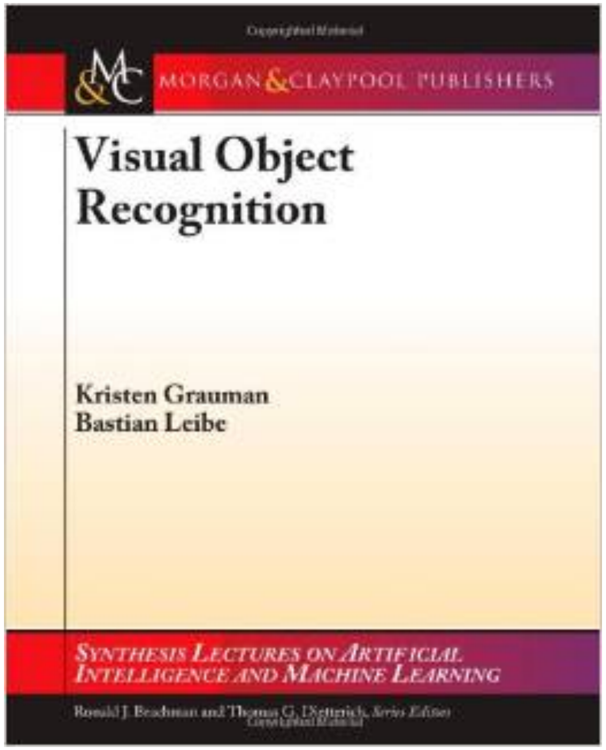
TEXTS IN COMPUTER SCIENCE

Computer Vision

Algorithms and Applications

Richard Szeliski

Springer



Copyrighted Material

MORGAN & CLAYPOOL PUBLISHERS

Visual Object Recognition

Kristen Grauman
Bastian Leibe

SYNTHESIS LECTURES ON ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Ronald J. Brachman and Thomas G. Dietterich, *Series Editors*

Copyrighted Material

By Rick Szeliski

http://szeliski.org/Book/

By Kristen Grauman, Bastian Leibe

Visual Object Recognition

Requirements / Grading

- Problem sets (97%)

- No final exam (0%)

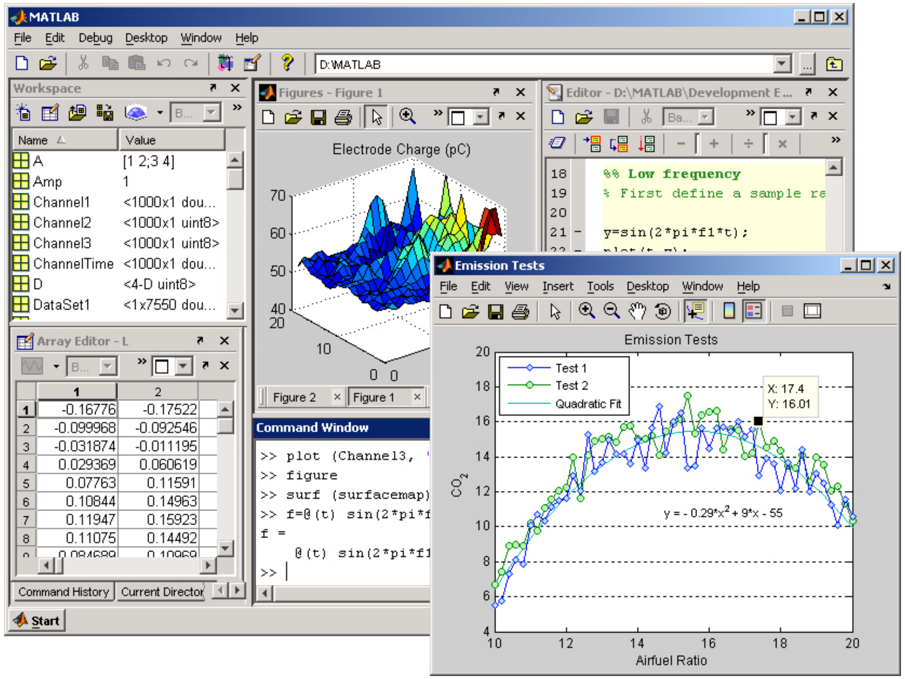
- Class and Piazza participation (3%)
  - Piazza: participation points for posting (sensible) questions and answers

Problem sets

- 5 problem sets (first and last will be short)
- Some short answer concept questions
- Matlab programming problems
  - Implementation
  - Explanation, results
- Follow instructions; points will be deducted if we can’t run your code out of the box
- Ask questions on Piazza first
- Submit to Canvas
- The assignments will take significant time to do -- **start early!**

- TA will go over problem set during discussion sections after release (others will be used as extra office hours)

Matlab



Name	Value
A	[1 2;3 4]
Amp	1
Channel1	<1000x1 dou...
Channel2	<1000x1 uint8>
Channel3	<1000x1 uint8>
ChannelTime	<1000x1 dou...
D	<4-D uint8>
DataSet1	<1x7550 dou...

	1	2
1	-0.16776	-0.17522
2	-0.099968	-0.092546
3	-0.031874	-0.011195
4	0.029369	0.060619
5	0.07763	0.11591
6	0.10844	0.14963
7	0.11947	0.15923
8	0.11075	0.14492
9	0.084689	0.10969

```
%% Low frequency
% First define a sample ra

y=sin(2*pi*f1*t);
```

```
>> plot (Channel3, '
>> figure
>> surf (surfacemap)
>> f=@(t) sin(2*pi*f
f =
    @(t) sin(2*pi*f1
>> |
```

- Built-in toolboxes for low-level image processing, visualization

- Compact programs

- Intuitive interactive debugging

- Widely used in engineering

Slide credit: Kristen Grauman

Matlab

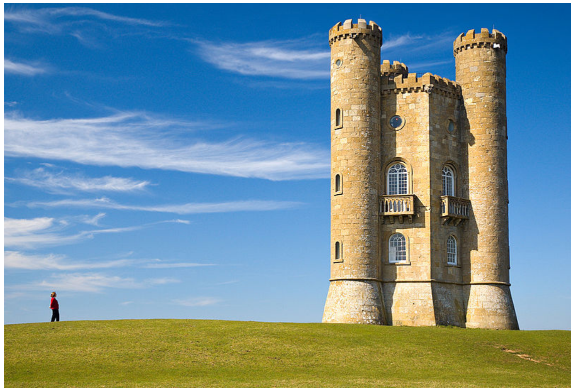
- Matlab available for free from campus software site

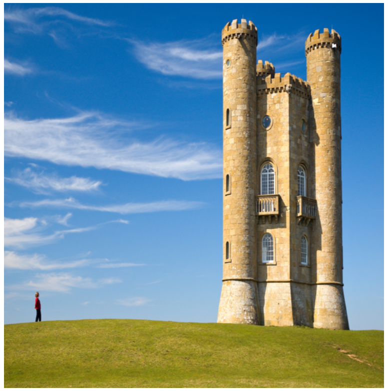
- In case we return to campus later on:
  - CSIF labs 67, 71, 75 (pc1-pc60)
  - Academic Surge 1044 and 1116
  - Remote access info found on class website

Problem Set 0

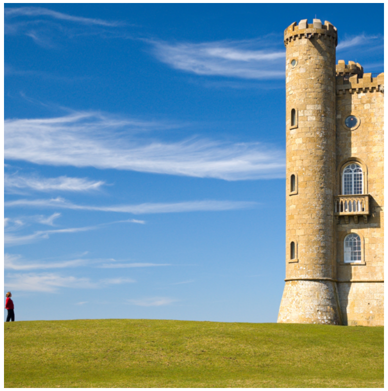
- Matlab warmup
- Basic image manipulation
- Out Thursday, due 4/9

Preview of some problem sets

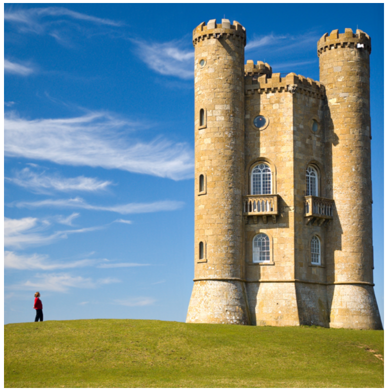




resize: castle squished



crop: castle cropped



content aware resizing:
seam carving

Preview of some problem sets

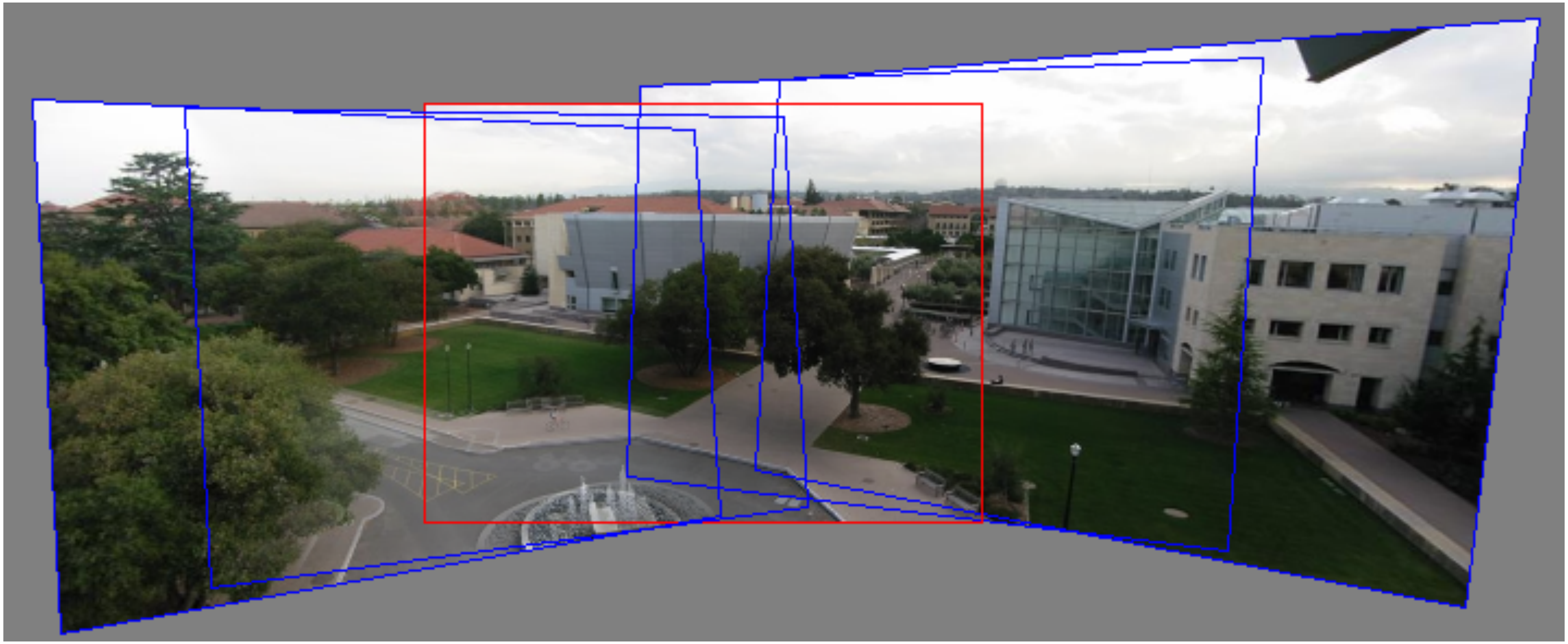
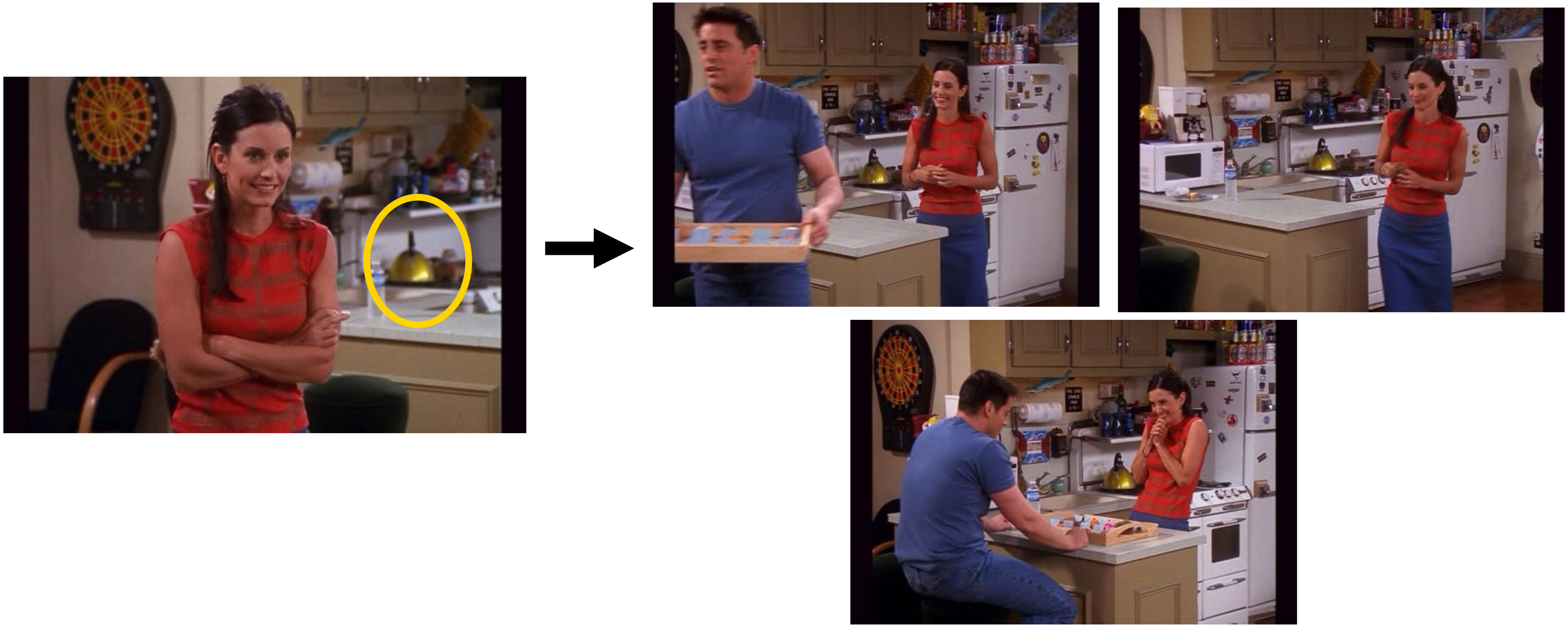


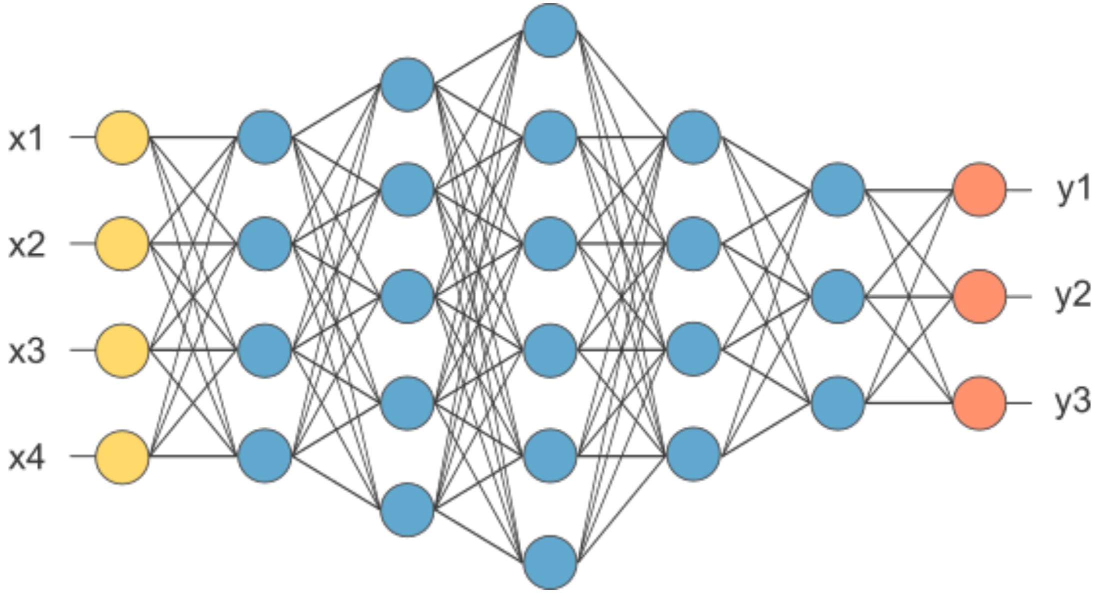
Image panorama

Preview of some problem sets



Object search and recognition

Preview of some problem sets



- Training and evaluating deep neural networks
- Will require python (we'll give a tutorial)

Problem set deadlines

- Problem sets due 11:59 PM PDT
  - Follow submission instructions given in assignment
  - Submit to Canvas; no hard copy submissions
  - Deadlines are firm. We'll use Canvas timestamp. Even 1 minute late is late.
- Late submissions: 1 point deduction for every hour after the deadline up to 72 hours; after 72 hours, you will receive a 0
- If your program doesn't work, clean up the code, comment it well, explain what you have, and still submit. Draw our attention to this in your answer sheet.

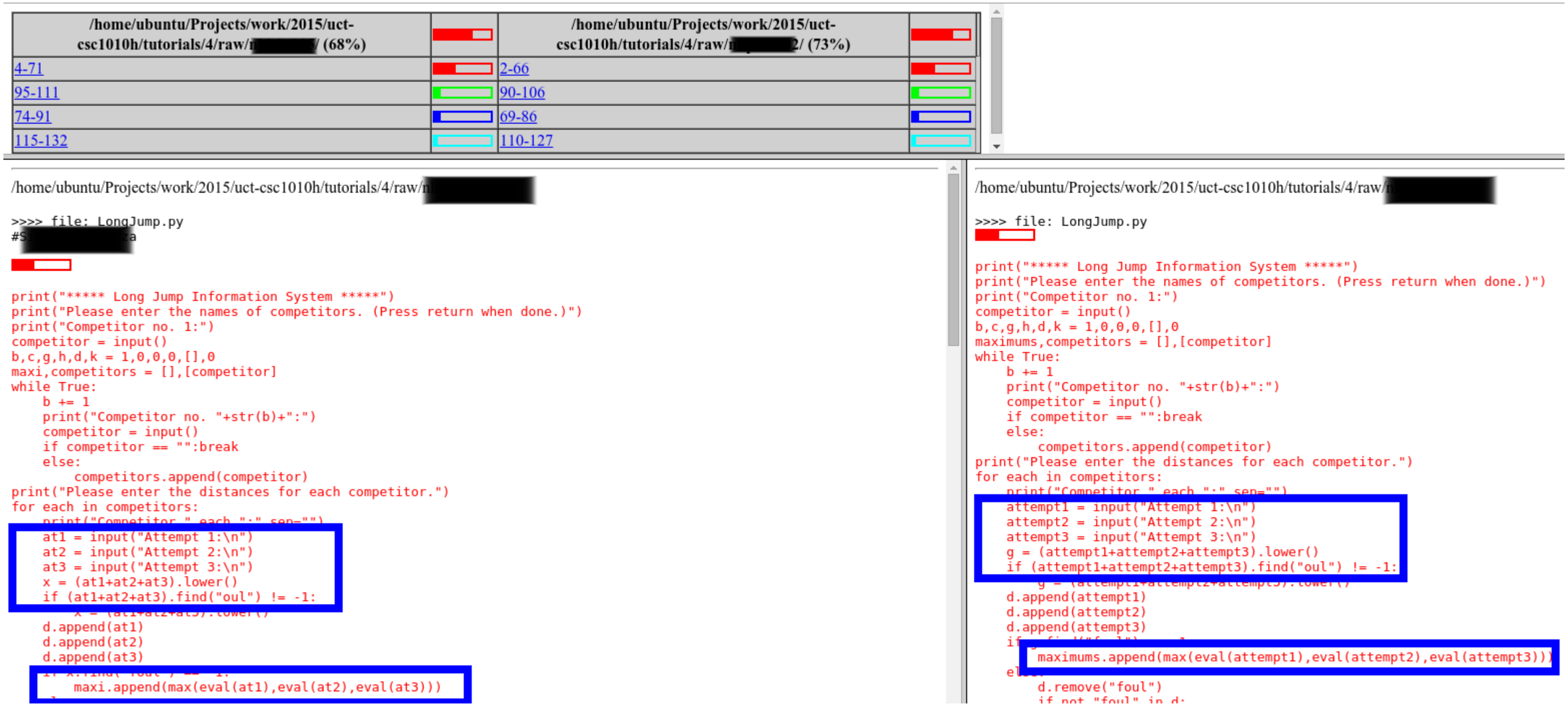
Collaboration policy

- **All responses and code must be written individually or in pairs (a group of 2)**

- Students submitting answers or code found to be identical or substantially similar (due to inappropriate collaboration) risk failing the course
  - We will be using MOSS to check for cheating!
  - Copying online solutions also counts as cheating!
  - Please don't cheat… you are going to get caught!

- Read and follow UC Davis code of conduct

MOSS



/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/4/raw/[redacted]/ (68%)	🟥	/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/4/raw/[redacted]/ (73%)	🟥
4-71	🟥	2-66	🟥
95-111	🟩	90-106	🟩
74-91	🟦	69-86	🟦
115-132	▭	110-127	▭

/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/4/raw/[redacted]

```
>>>> file: LongJump.py
#[redacted]

print("***** Long Jump Information System *****")
print("Please enter the names of competitors. (Press return when done.)")
print("Competitor no. 1:")
competitor = input()
b,c,g,h,d,k = 1,0,0,0,[],0
maxi,competitors = [],[competitor]
while True:
    b += 1
    print("Competitor no. "+str(b)+":")
    competitor = input()
    if competitor == "":break
    else:
        competitors.append(competitor)
print("Please enter the distances for each competitor.")
for each in competitors:
    print("Competitor ",each,":",sep="")
    at1 = input("Attempt 1:\n")
    at2 = input("Attempt 2:\n")
    at3 = input("Attempt 3:\n")
    x = (at1+at2+at3).lower()
    if (at1+at2+at3).find("oul") != -1:
        x = (at1+at2+at3).lower()
    d.append(at1)
    d.append(at2)
    d.append(at3)
    if x.find("foul") == -1:
        maxi.append(max(eval(at1),eval(at2),eval(at3)))
```

/home/ubuntu/Projects/work/2015/uct-csc1010h/tutorials/4/raw/[redacted]

```
>>>> file: LongJump.py

print("***** Long Jump Information System *****")
print("Please enter the names of competitors. (Press return when done.)")
print("Competitor no. 1:")
competitor = input()
b,c,g,h,d,k = 1,0,0,0,[],0
maximums,competitors = [],[competitor]
while True:
    b += 1
    print("Competitor no. "+str(b)+":")
    competitor = input()
    if competitor == "":break
    else:
        competitors.append(competitor)
print("Please enter the distances for each competitor.")
for each in competitors:
    print("Competitor ",each,":",sep="")
    attempt1 = input("Attempt 1:\n")
    attempt2 = input("Attempt 2:\n")
    attempt3 = input("Attempt 3:\n")
    g = (attempt1+attempt2+attempt3).lower()
    if (attempt1+attempt2+attempt3).find("oul") != -1:
        g = (attempt1+attempt2+attempt3).lower()
    d.append(attempt1)
    d.append(attempt2)
    d.append(attempt3)
    if g.find("foul") == -1:
        maximums.append(max(eval(attempt1),eval(attempt2),eval(attempt3)))
    else:
        d.remove("foul")
        if not "foul" in d:
```

Miscellaneous

- Check class website regularly for assignment files, notes, announcements, etc.
- Come to virtual lecture on time
- Please interrupt with questions at any time by virtually raising hand
- (If we return to campus) No laptops, phones, tablets, etc. in class please

Coming up

- Read the class webpage carefully
- Next class (Thurs): Matlab tutorial
- PS0 out Thursday, due 4/9