CPSC 425: Computer Vision

Image Credit: Devi Parikh

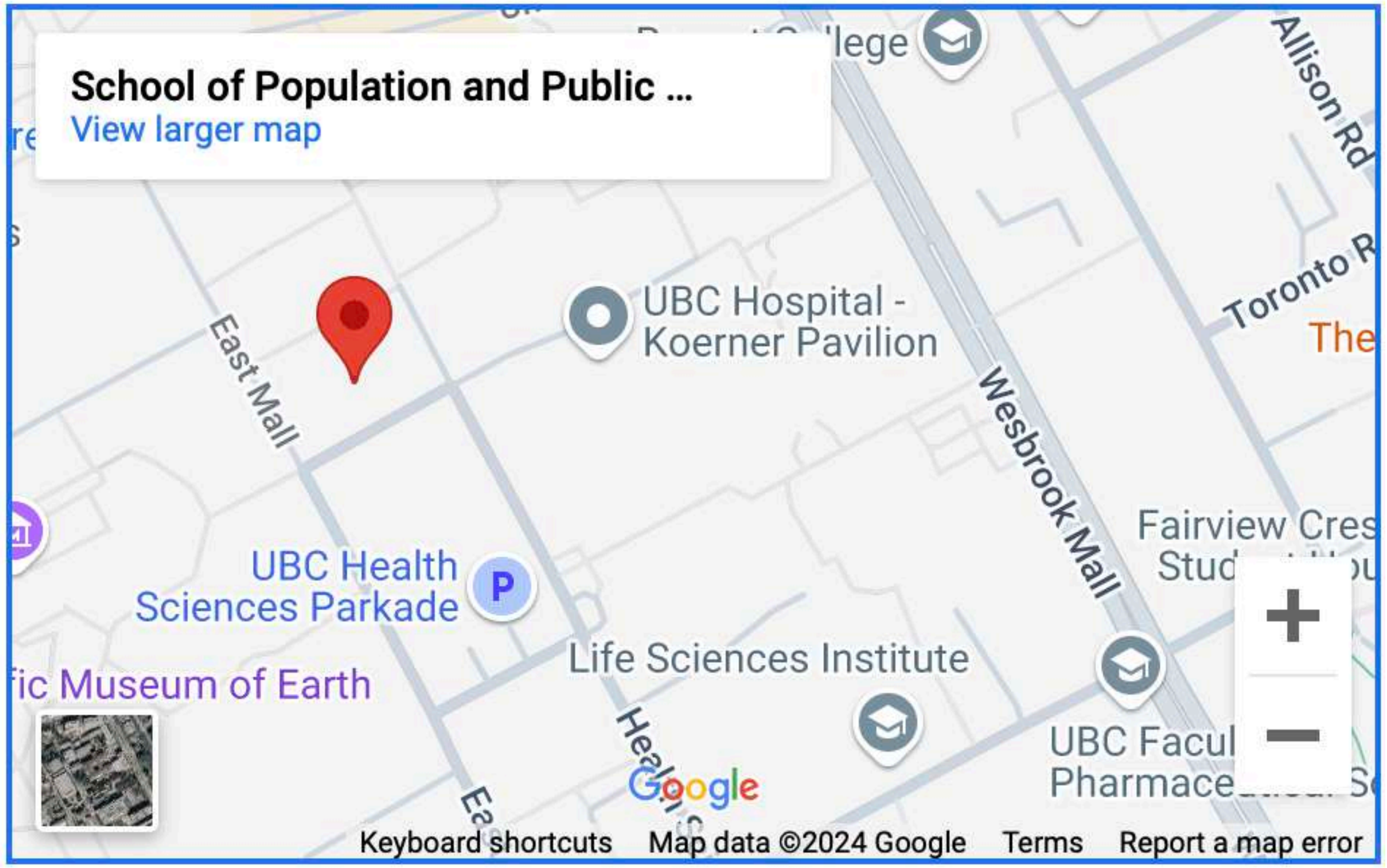
Lecture 1: Introduction and Course Logistics

Course **logistics**

Times: Mon, Wed 12:30-2:00pm

Locations: SPPH, B151

Instructor: Kwang Moo **YI**



E-mail: kmyi@cs.ubc.ca

Office: ICICS 115

About **me** …

I have been working in **Computer Vision** for the last 15+ years

Assistant Professor
2020–present
THE UNIVERSITY OF BRITISH COLUMBIA

Assistant Professor
2017–2020
UVIC

Postdoctoral Researcher
2014–2017
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE
EPFL

PhD
2007–2014
Seoul National University

About **me …**

Image is in the public domain

About **me ...**

Course **logistics**

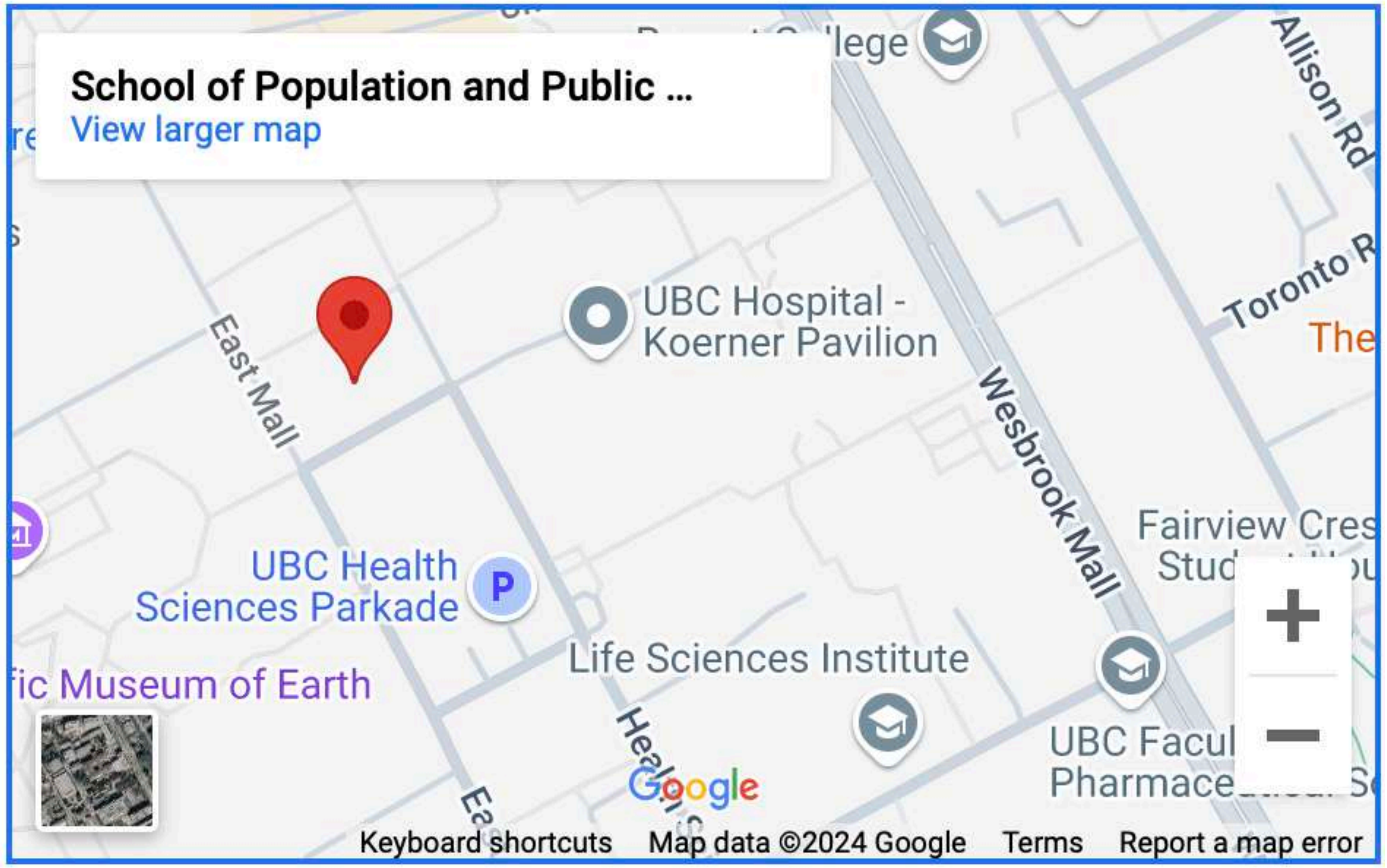
Times: Mon, Wed 12:30-2:00pm

Locations: SPPH, B151

Instructor: Kwang Moo **YI**

E-mail: kmyi@cs.ubc.ca

Office: ICICS 115



Course **logistics**

Times: Mon, Wed 12:30-2:00pm

Locations: SPPH, B151

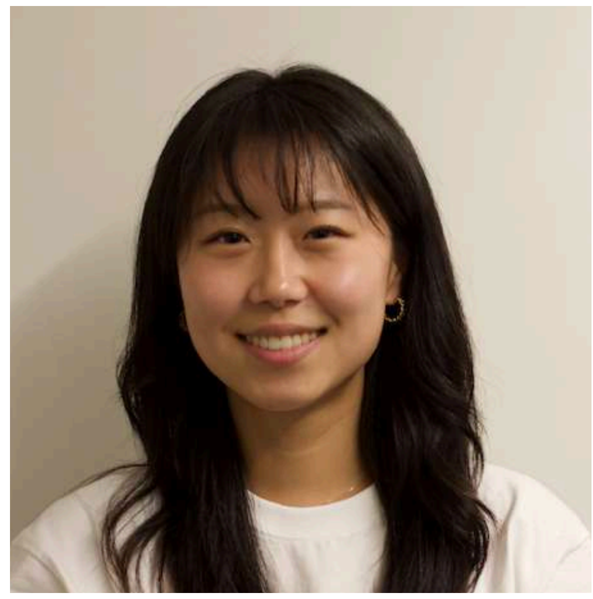
Instructor: Kwang Moo **YI**

E-mail: kmyi@cs.ubc.ca

Office: ICICS 115

Teaching Assistants

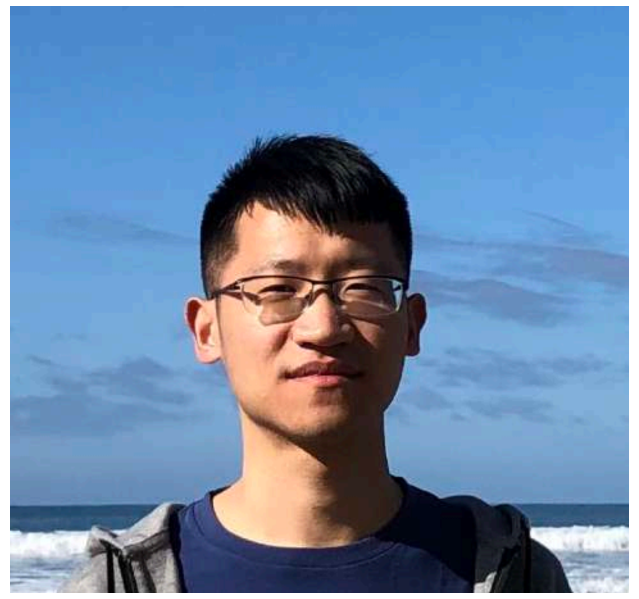
Yeojun Han



Jeff Tseng

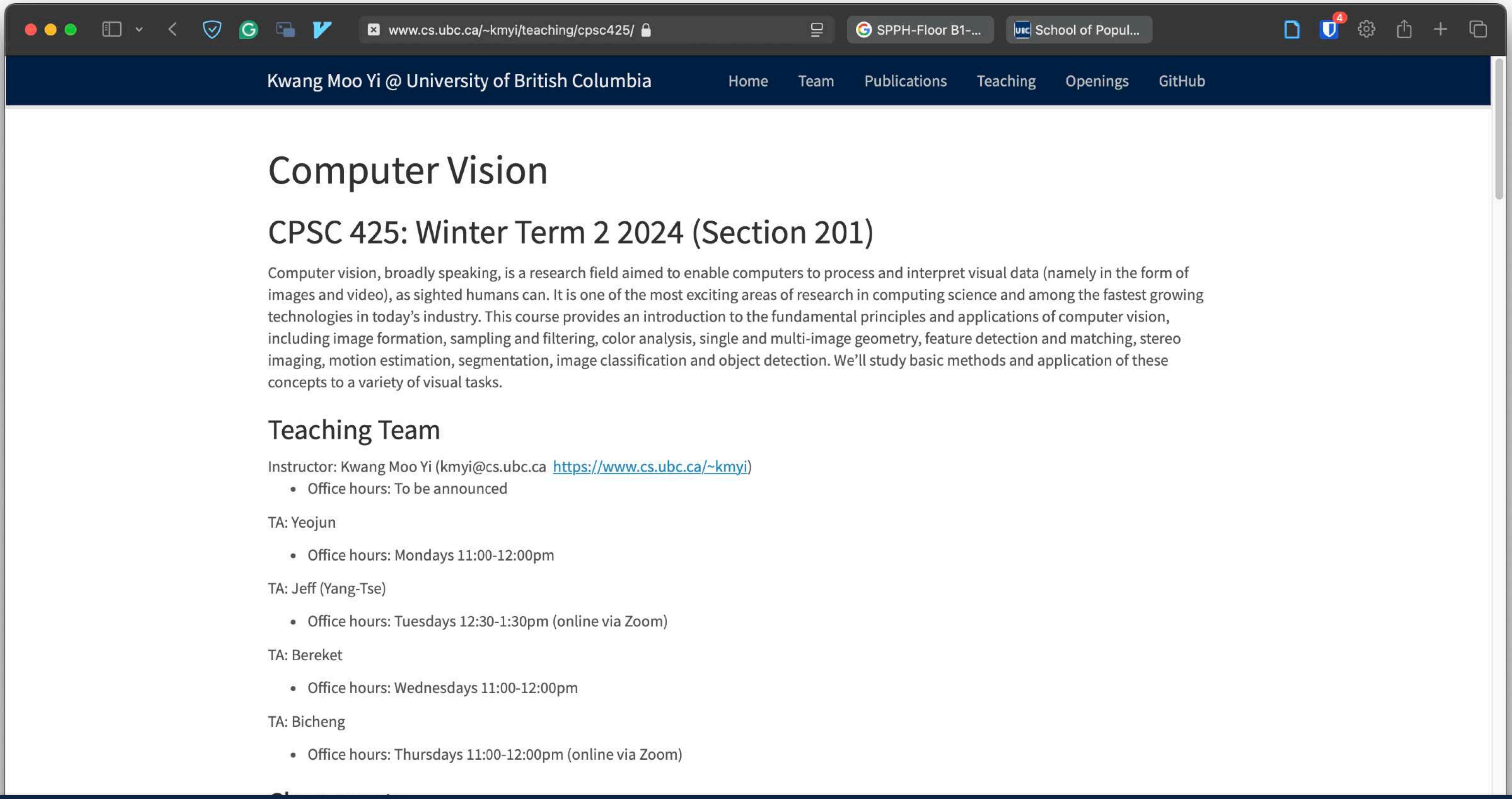
Bereket Guta

Bicheng Xu



Post questions on Piazza!

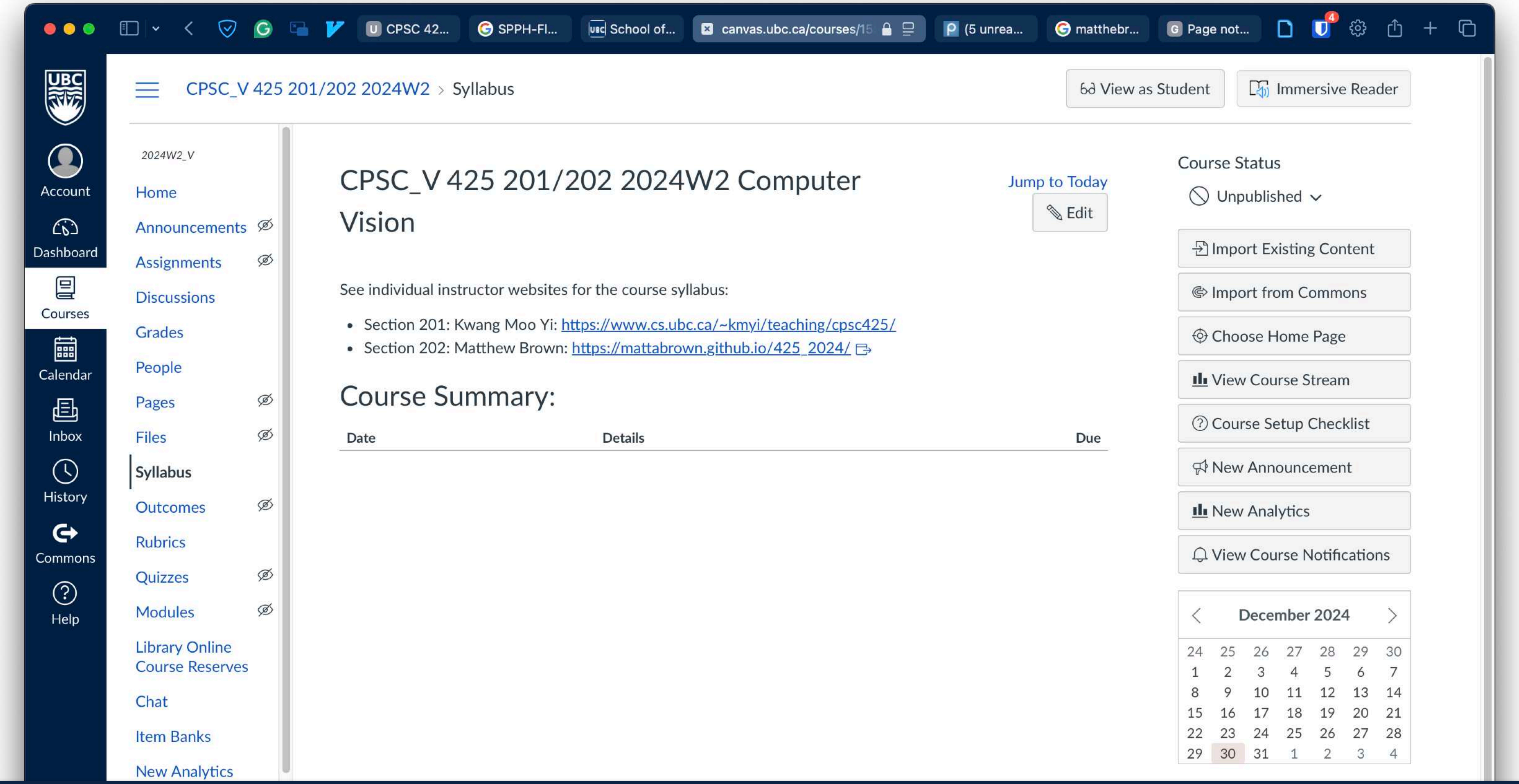
Course Webpage



- Schedule, Assignments
- Lecture Slides and Notes
- Course Information (public)

https://www.cs.ubc.ca/~kmyi/teaching/cpsc425

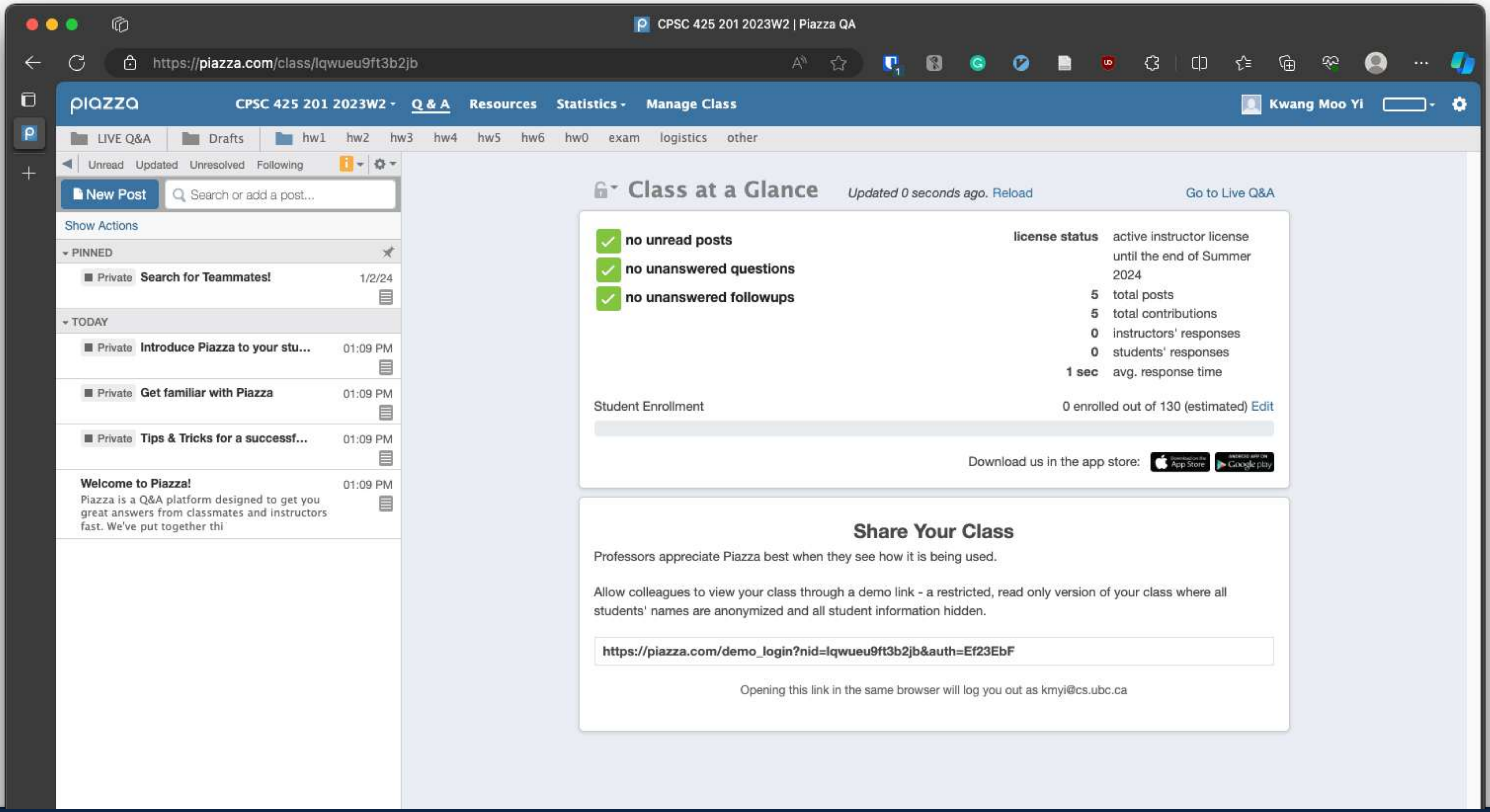
Canvas



- Assignment hand-in
- Course Information (private)
- Gradescope linke (assignments)
- Piazza link

https://canvas.ubc.ca/courses/153996

Piazza



- Discussions and Q+A
- Confused? Likely someone else has the same question as you!
- Lectures, Technical Issues, Assignments …
- Instructor/TA will answer only between **9am6pm—, weekdays**

Link in Canvas and the course website

Office Hours

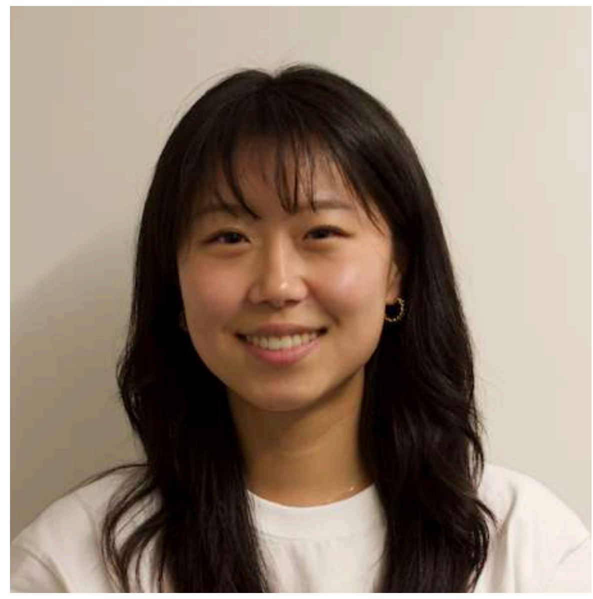
Starts next week (Week of Jan 15th)

Instructor: Kwang Moo **YI**

TBD

Teaching Assistants

Yeojun Han



Mon 11am

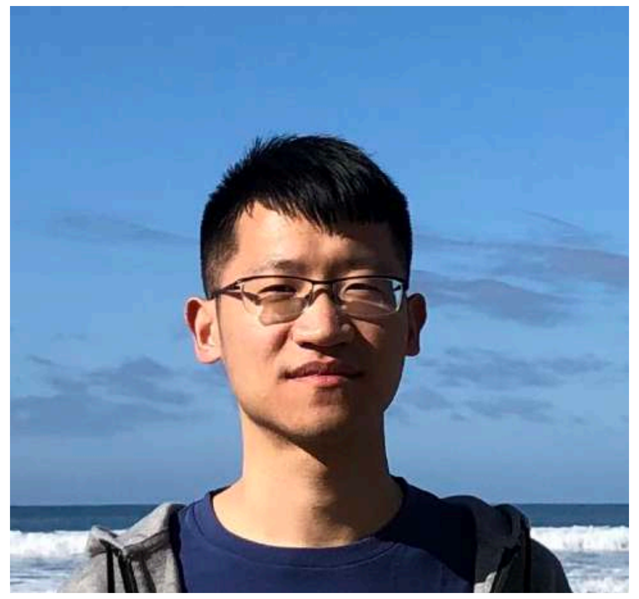
Jeff Tseng

Tue 12:30 pm
(online)

Bereket Guta

Wed 11am

Bicheng Xu



Thur 11am
(online)

See Course website for Links and Locations
(announced next week)

What is **Computer Vision**?

Image Credit: https://www.deviantart.com/infinitecreations/art/BioMech-Eye-168367549

What is **Computer Vision**?

Compute vision, broadly speaking, is a research field aimed to enable computers to **process and interpret visual data**, as sighted humans can.

Image (or video)

Sensing Device

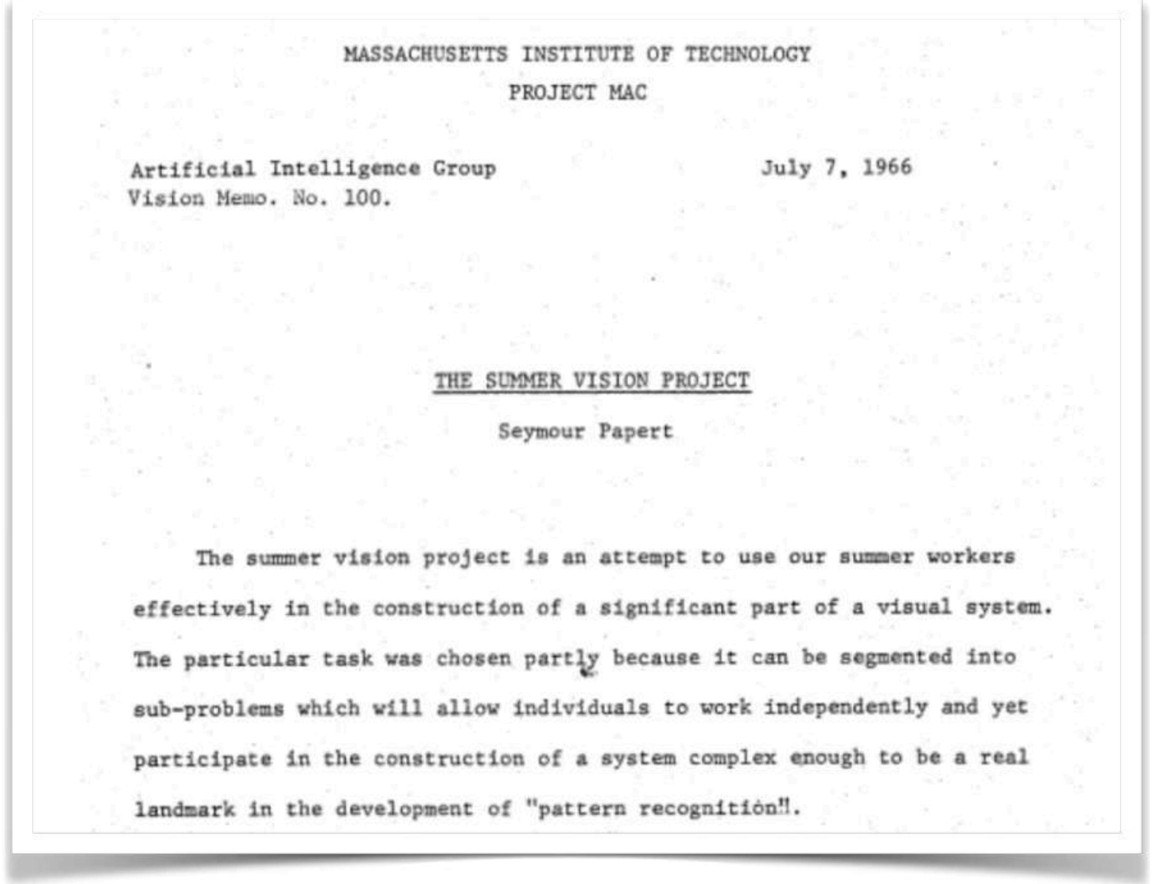
Interpreting Device

Image Credit: https://www.flickr.com/photos/flamephoenix1991/8376271918

Interpretation

blue sky,
trees,
fountains,
UBC, …

Computer vision … the beginning …



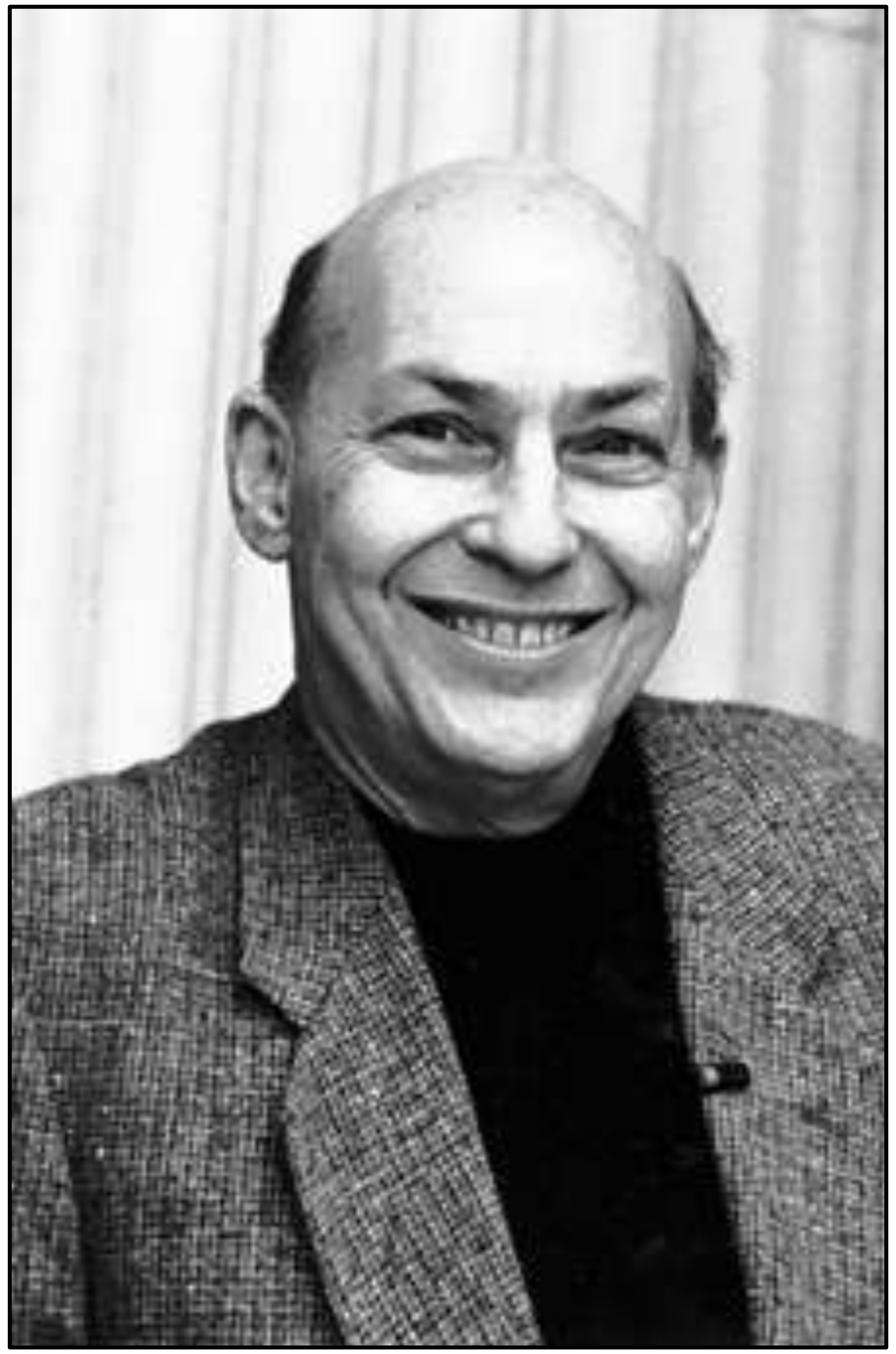
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
PROJECT MAC

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".



The Summer Vision Project

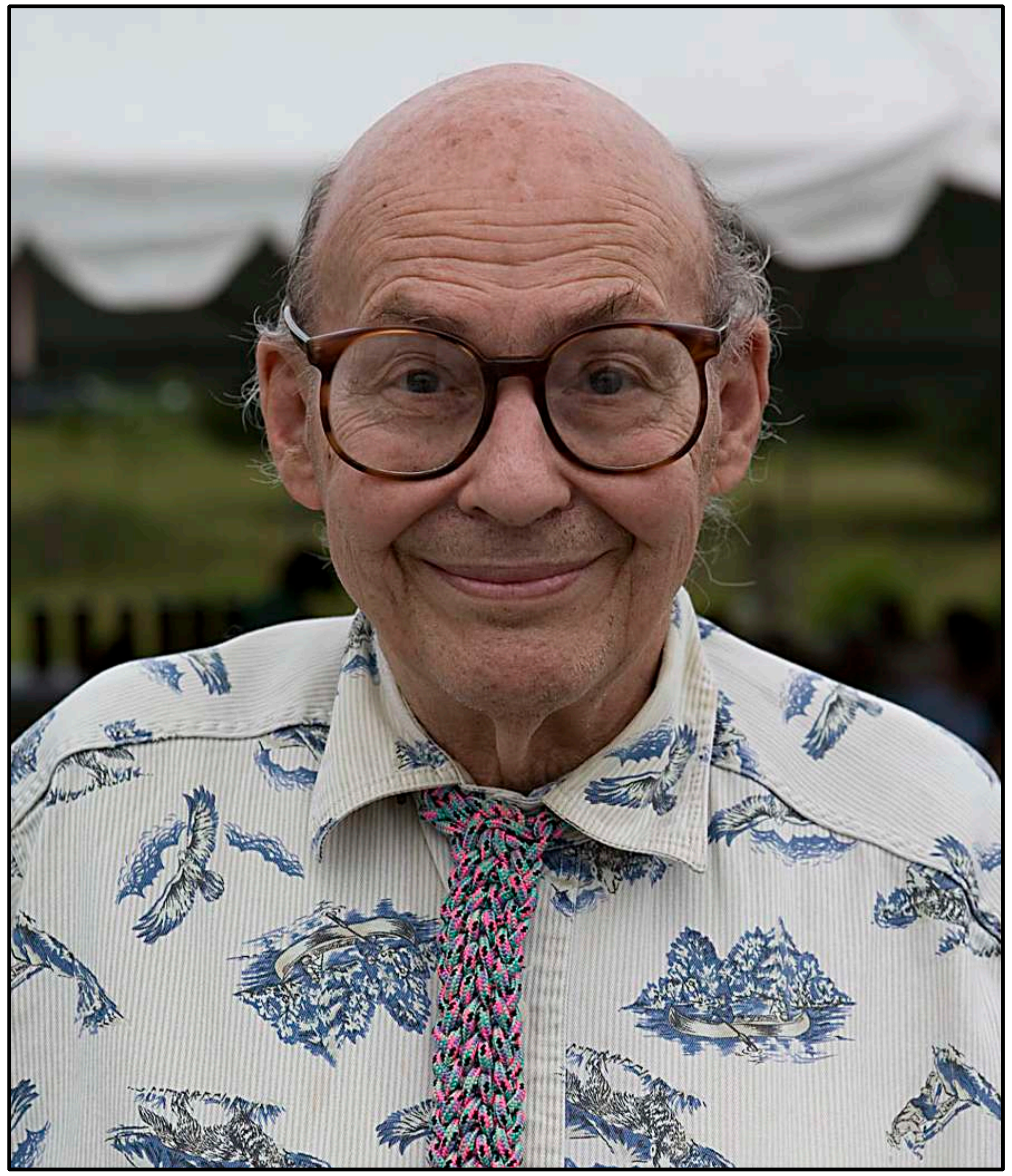
"spend the summer linking a camera to a computer and getting the computer to describe what it saw"

- Marvin Minsky (1966), MIT
  Turing Award (1969)

… >50 years later

Slide Credit: Devi Parikh (GA Tech)

Computer vision … the beginning …



Gerald Sussman, MIT

"You'll notice that **Sussman** never worked in vision again!" – Berthold Horn

Slide Credit: Devi Parikh (GA Tech)

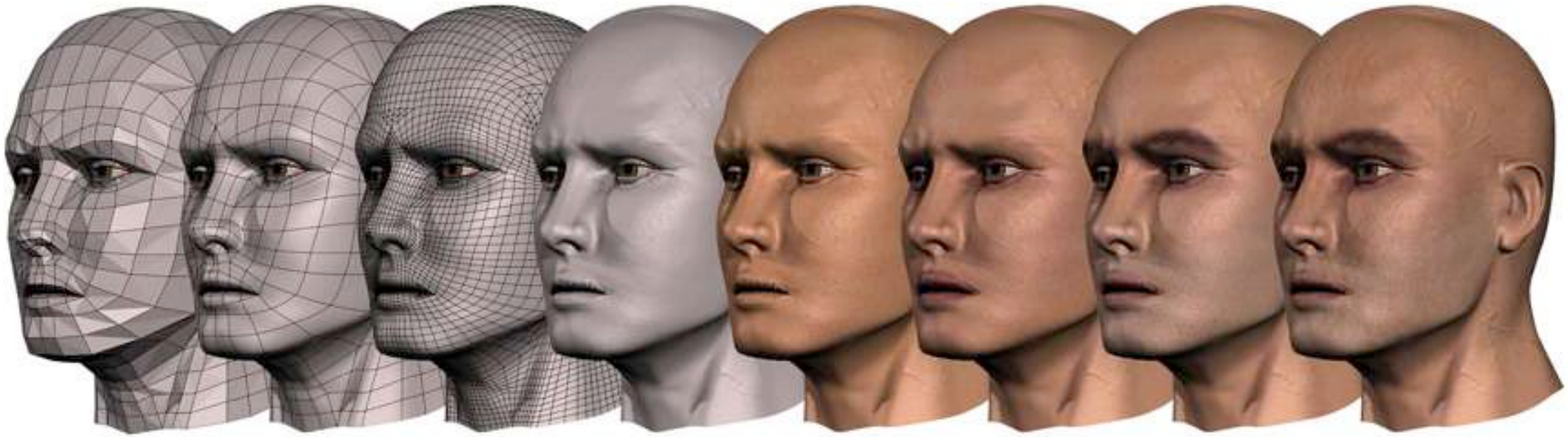
Definitions of Computer Vision #1

"Inverse Computer Graphics"

[J. Buckner]

Definitions of Computer Vision #1

"Inverse Computer Graphics"



Graphics

Vision

Definitions of Computer Vision #2

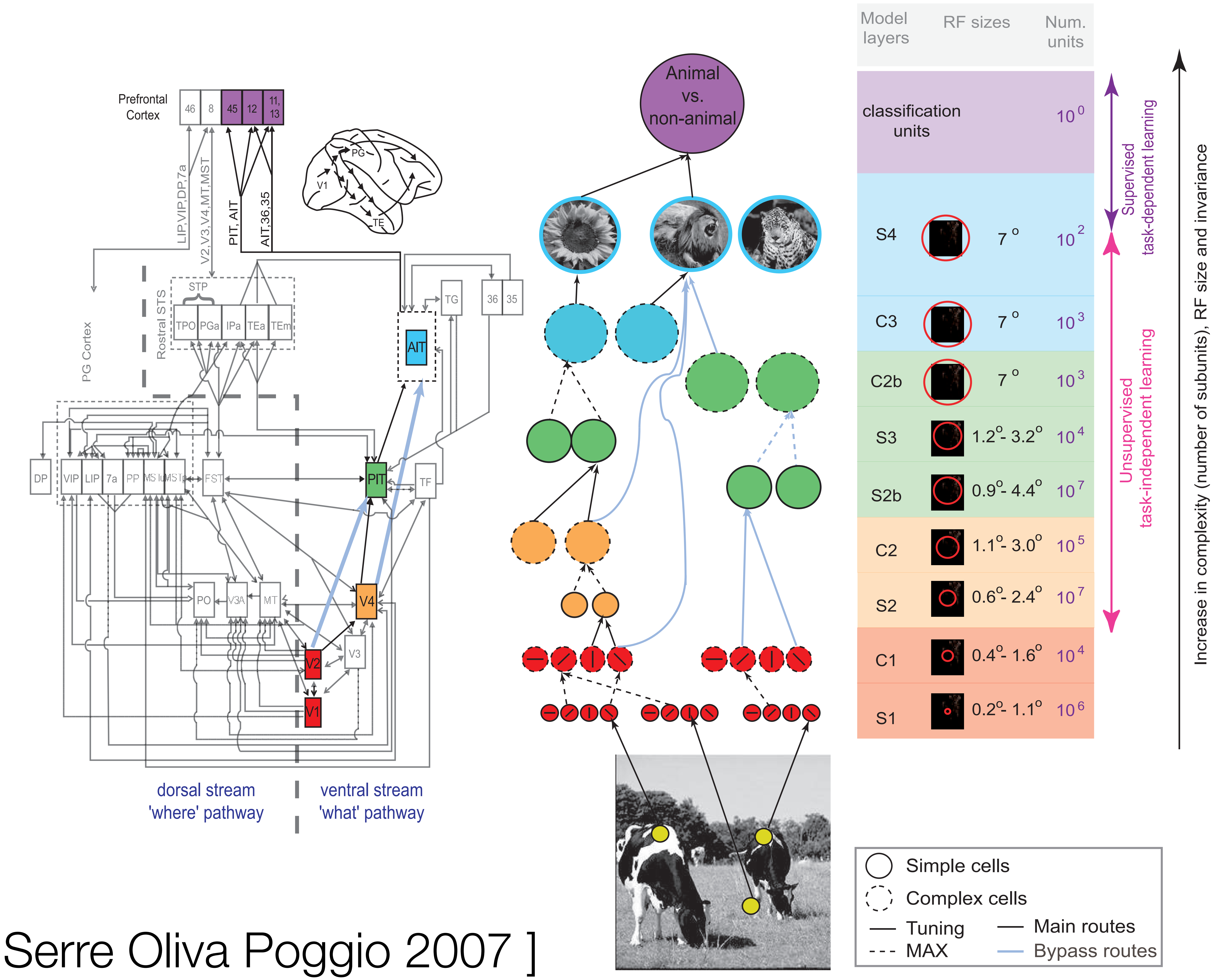
“Replicate Human Vision”

=

=

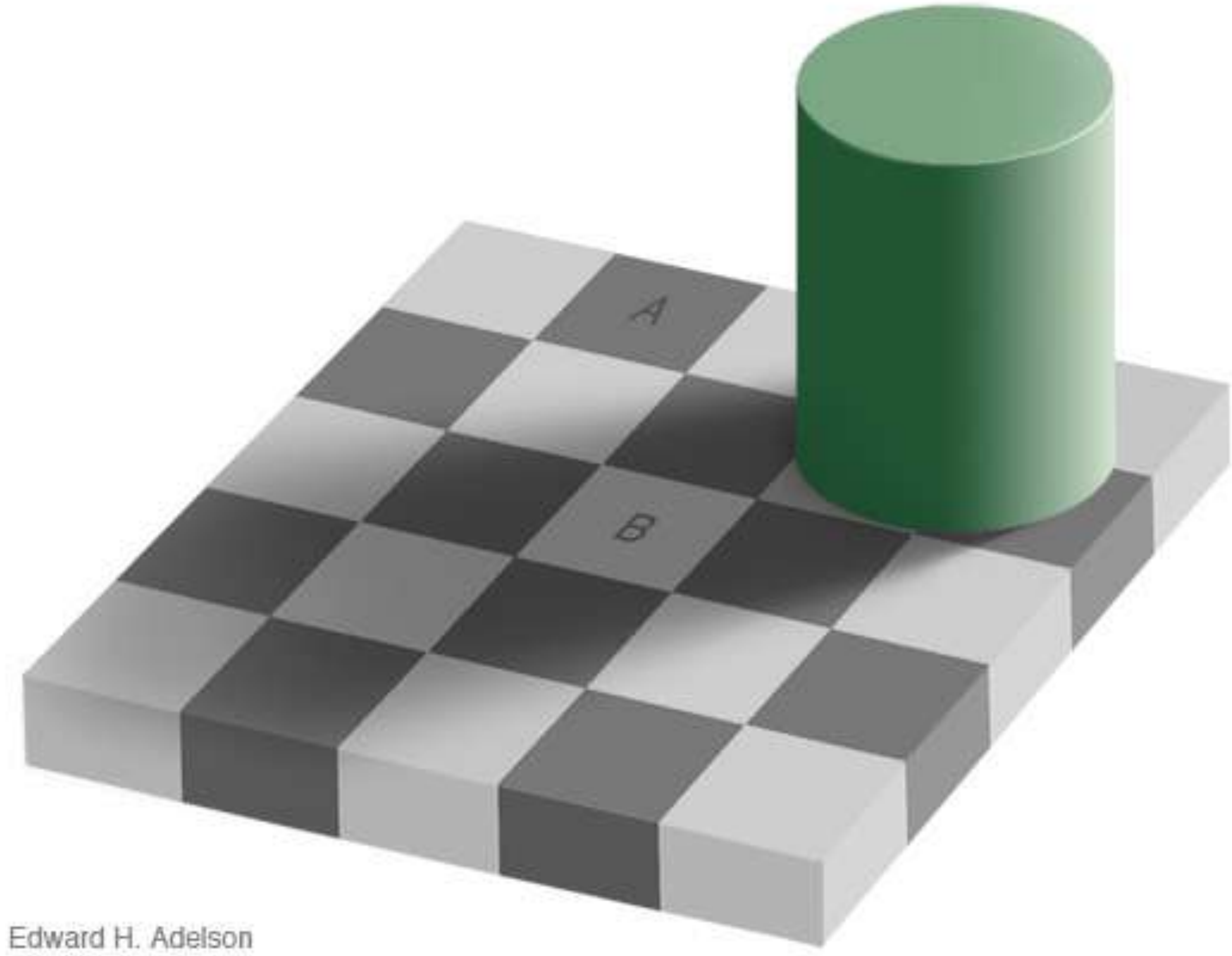
Definitions of Computer Vision #2

“Replicate Human Vision”

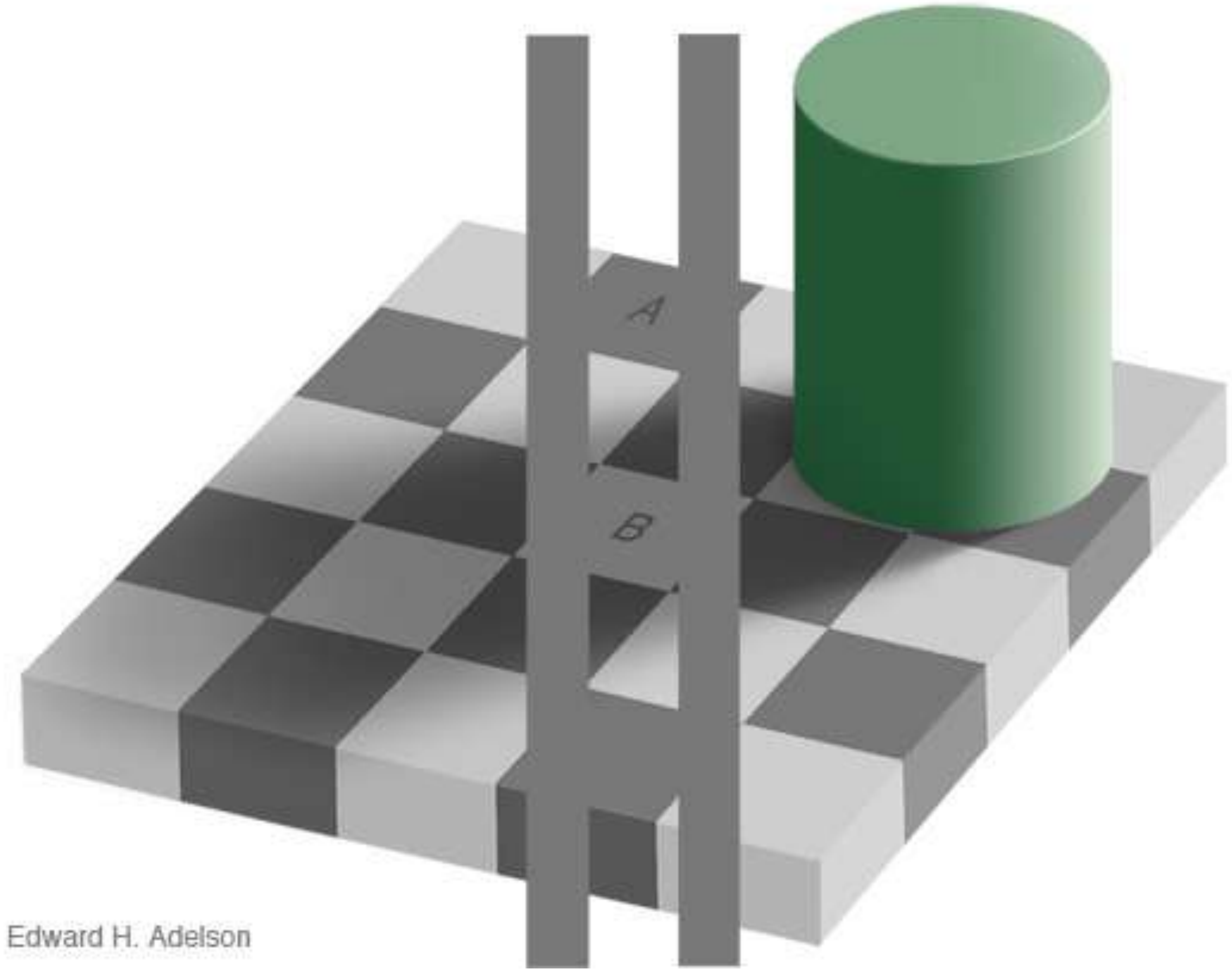


[Serre Oliva Poggio 2007]

Can computers **match (or beat)** human vision?

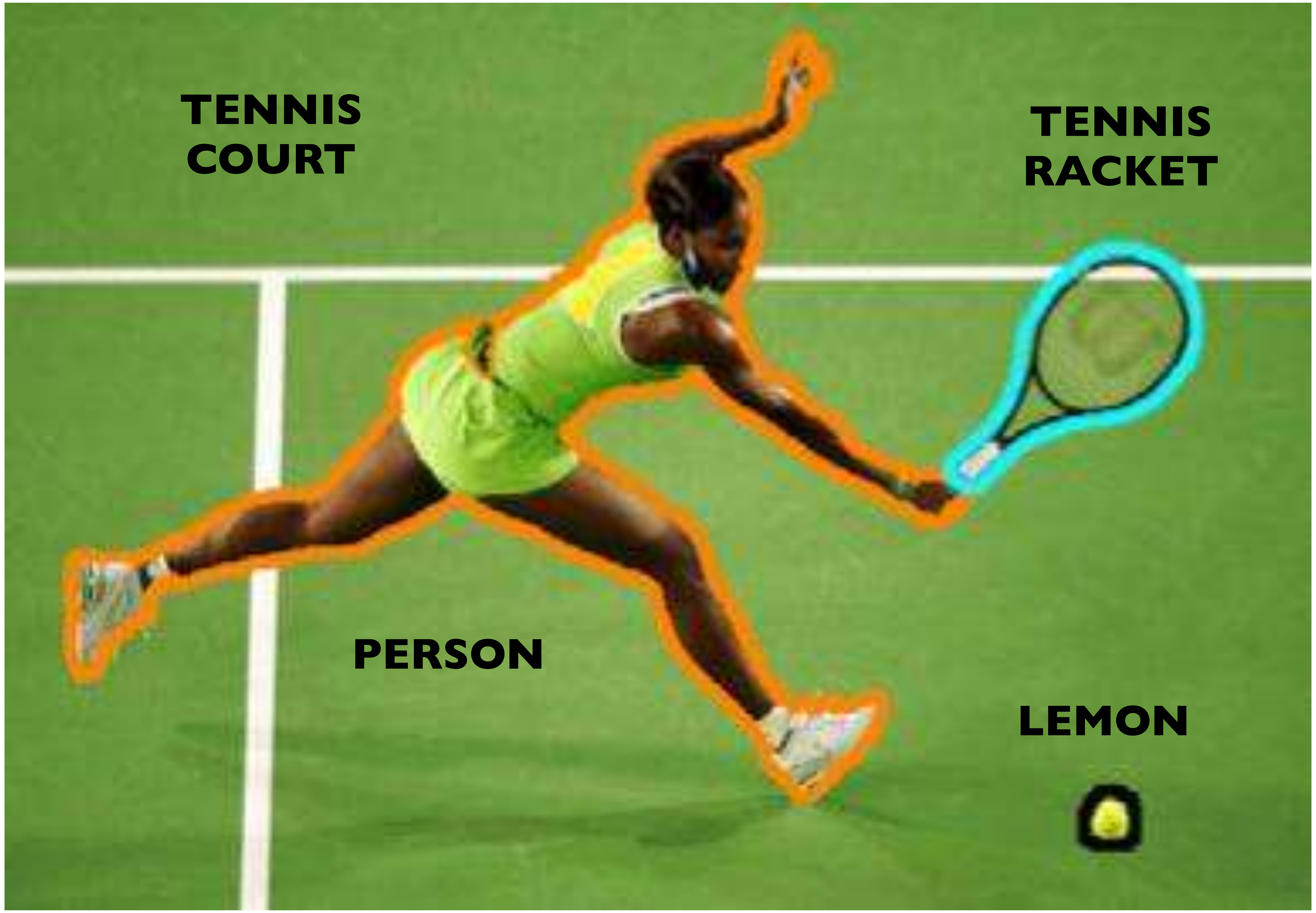


Can computers **match (or beat)** human vision?



Definitions of Computer Vision #3

“Image/Video Understanding”

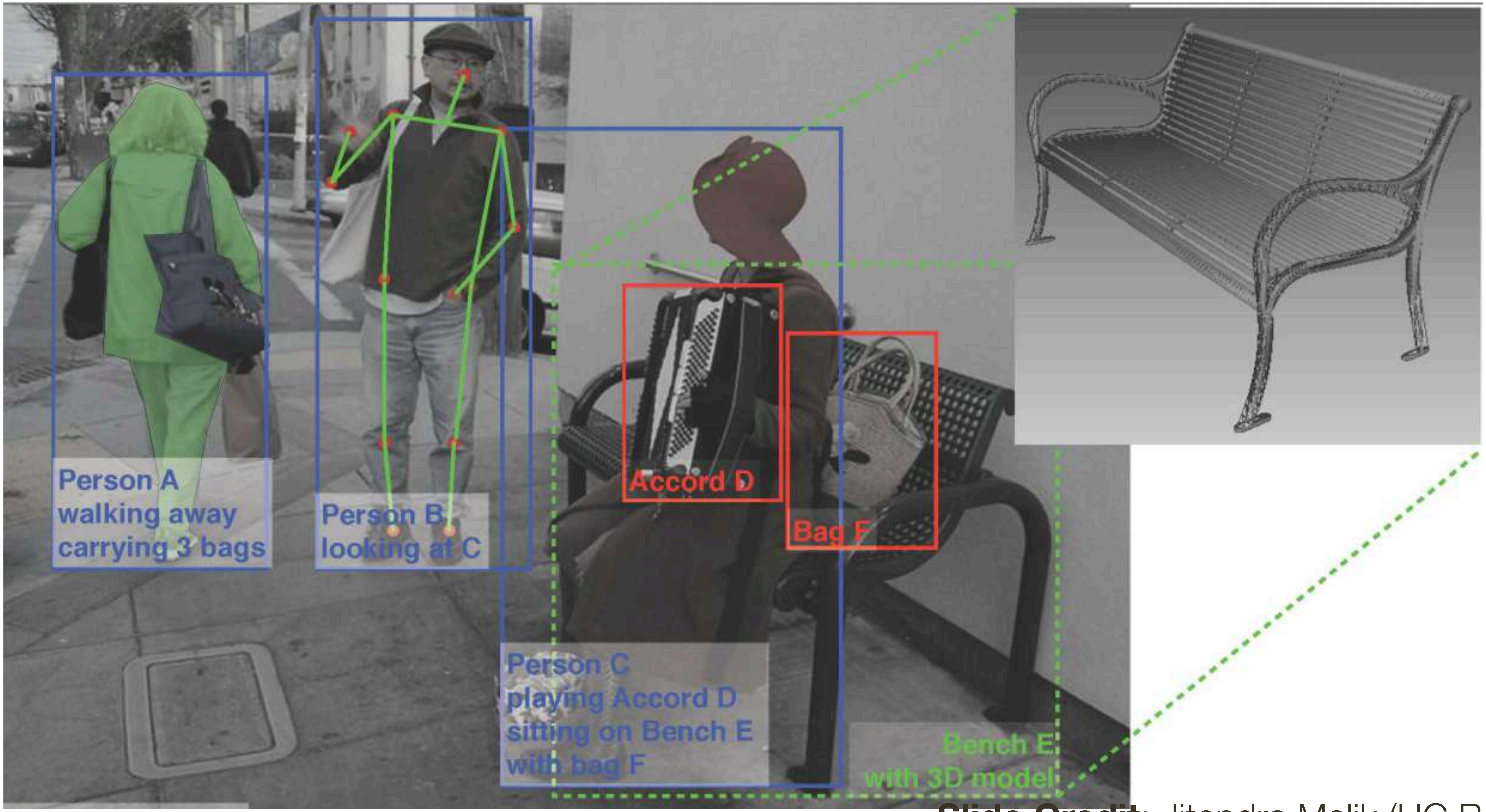


[Rabinovich, Galleguillos, Wiewiora, Belongie 2007]

What do **you** see?

Slide Credit: Jitendra Malik (UC Berkeley)

What we would like **computer to infer**?



Slide Credit: Jitendra Malik (UC Berkeley)

What we would like **computer to infer**?

Will person B put some money into person C's cup?

Slide Credit: Jitendra Malik (UC Berkeley)

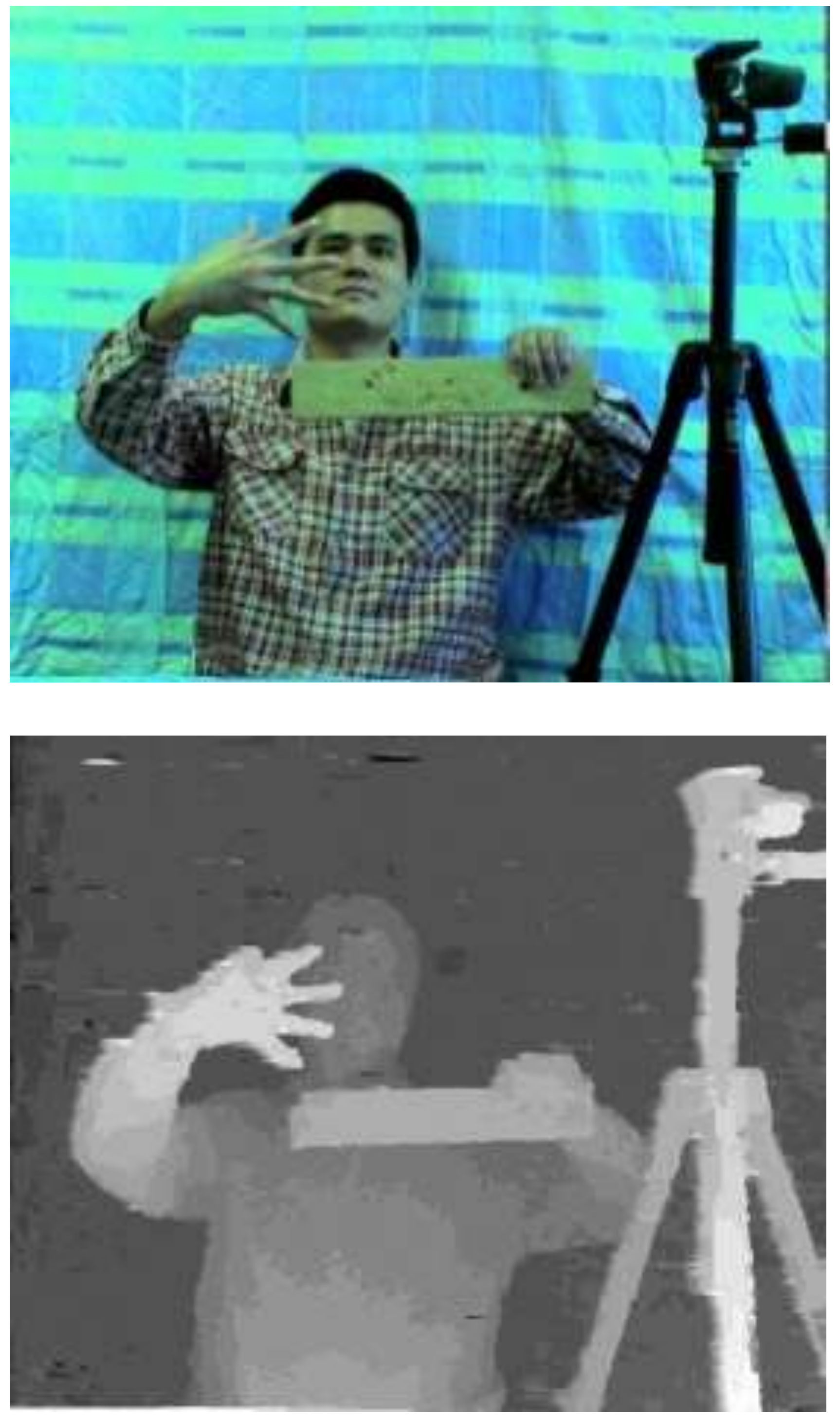
Computer **Vision Problems**

1. Computing properties of the 3D world from visual data (***measurement***)

Slide Credit: Kristen Grauman (UT Austin)

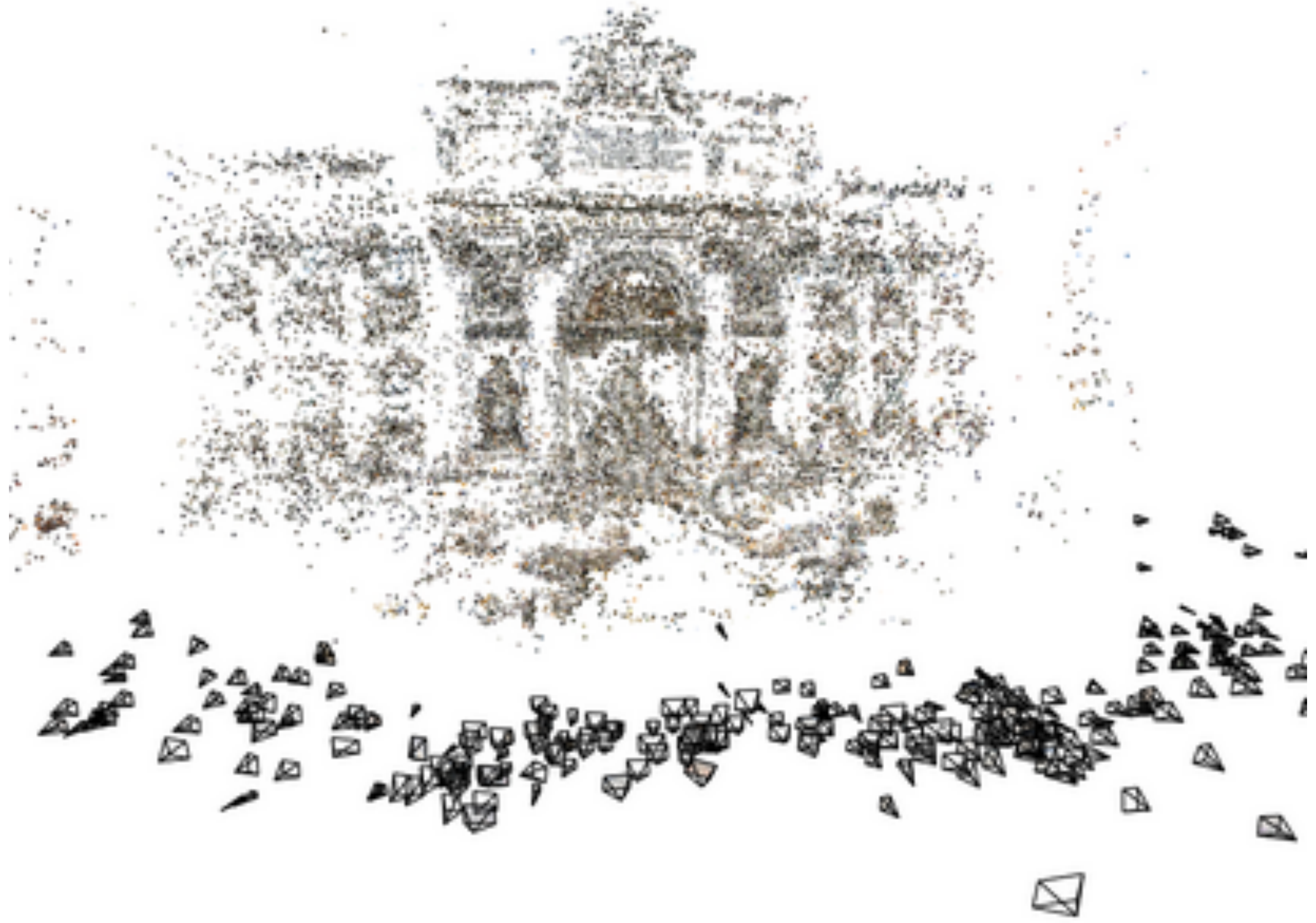
1. Vision for **Measurement**

Real-time stereo



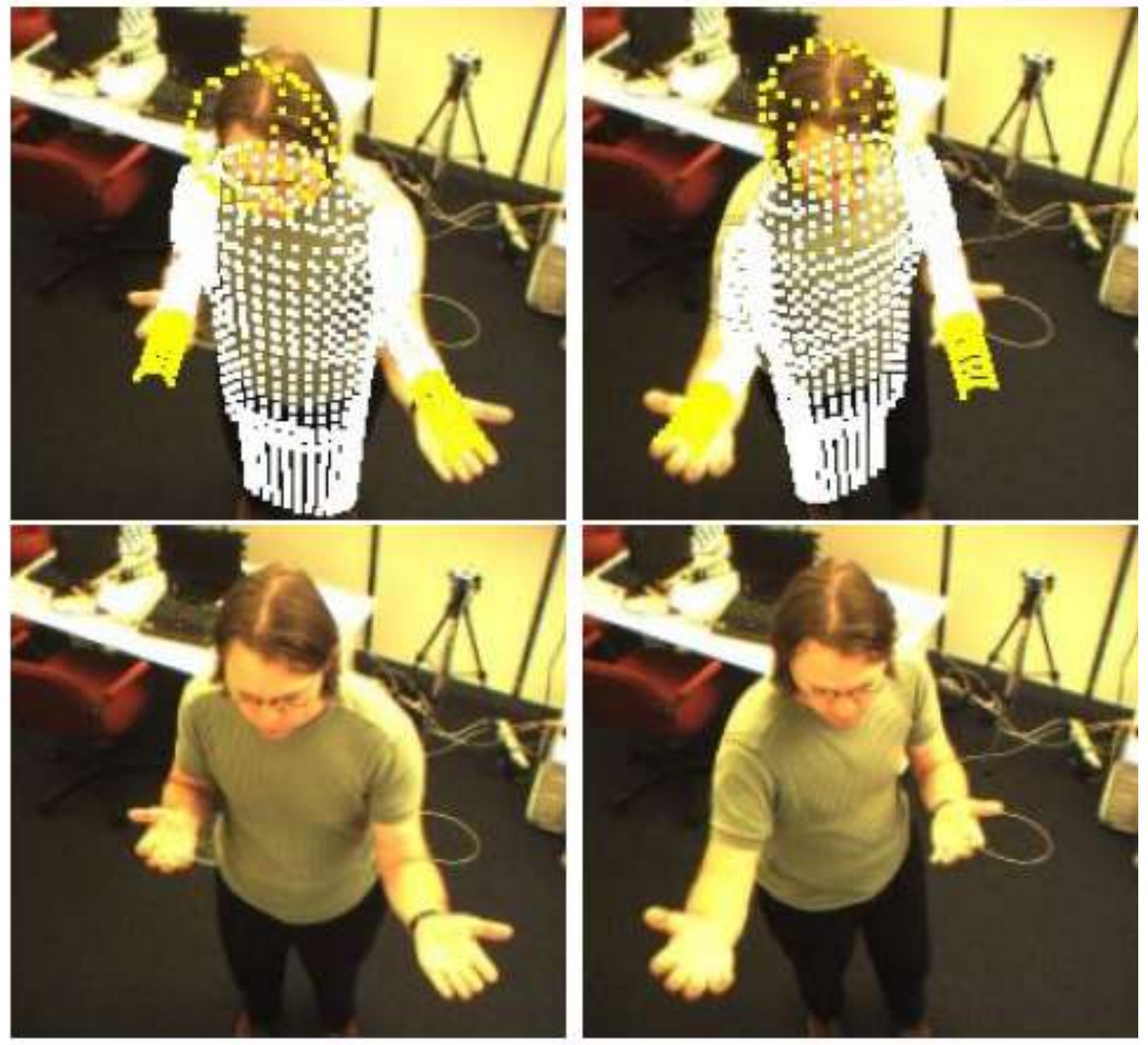
Wang et al.

Structure from motion



Snavely et al.

Tracking



Demirdjian et al.

Slide Credit: Kristen Grauman (UT Austin)

Computer **Vision Problems**

1. Computing properties of the 3D world from visual data (***measurement***)

Ill-posed problem: real world is much more complex than what we can measure in images: 3D -> 2D

It is (literally) impossible to invert the image formation process

Slide Credit: Kristen Grauman (UT Austin)

Computer **Vision Problems**

1. Computing properties of the 3D world from visual data (***measurement***)

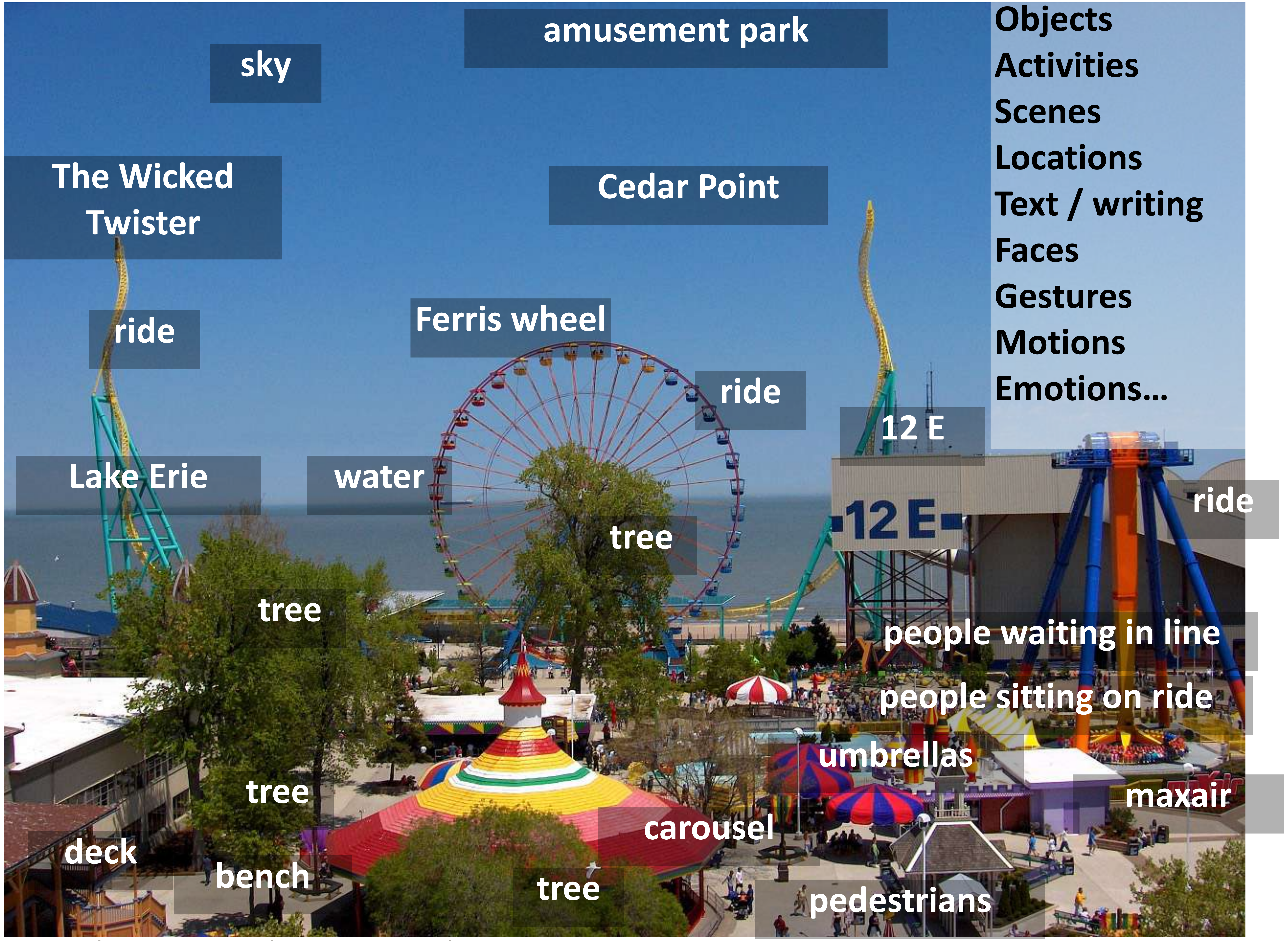
2. Algorithms and representations to allow a machine to recognize objects, people, scenes, and activities (***perception and interpretation***)

Slide Credit: Kristen Grauman (UT Austin)

2. Vision for **Perception and Interpretation**

Slide Credit: Kristen Grauman (UT Austin)

2. Vision for **Perception and Interpretation**



Slide Credit: Kristen Grauman (UT Austin)

Computer **Vision Problems**

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***)

It is computationally intensive / expensive

Slide Credit: Kristen Grauman (UT Austin)

2. Vision for **Perception and Interpretation**

~ 55% of **cerebral cortex** in humans (13 billion neurons) are devoted to vision

more human brain devoted to vision than anything else

Computer **Vision Problems**

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***)

It is computationally intensive / expensive

We do not (fully) understand the processing mechanisms involved

Slide Credit: Kristen Grauman (UT Austin)

Computer **Vision Problems**

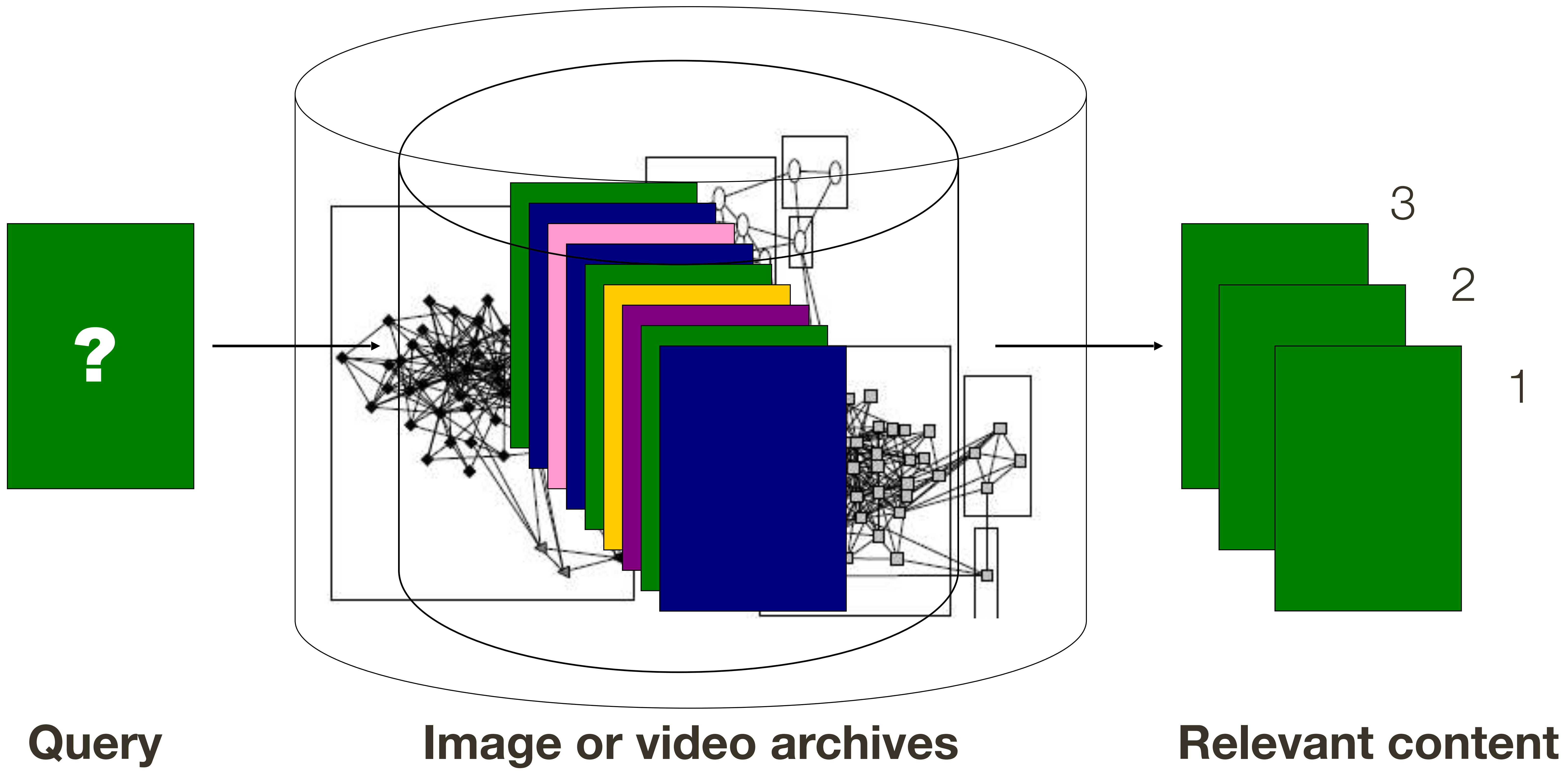
1. Computing properties of the 3D world from visual data (***measurement***)

2. Algorithms and representations to allow a machine to recognize objects, people, scenes, and activities (***perception and interpretation***)

3. Algorithms to mine, search, and interact with visual data (***search and organization***)

Slide Credit: Kristen Grauman (UT Austin)

3. **Search and Organization**



Slide Credit: Kristen Grauman (UT Austin)

Computer **Vision Problems**

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***)

3. Algorithms to mine, search, and interact with visual data (***search and organization***)

Scale is enormous, explosion of visual content

Slide Credit: Kristen Grauman (UT Austin)

3. **Search and Organization**

*from iStock by GettyImages

Snapchat	WhatsApp	Facebook
31.7 Million / hour	**29.2 Million** / hour	**14.6 Million** / hour

Instagram	Flickr	YouTube
2.9 Million / hour	**0.2 Million** / hour	**18K hours** / hour

*based on article by Kimberlee Morrison in Social Times (2015)

Computer **Vision Problems**

1. Computing properties of the 3D world from visual data (***measurement***)

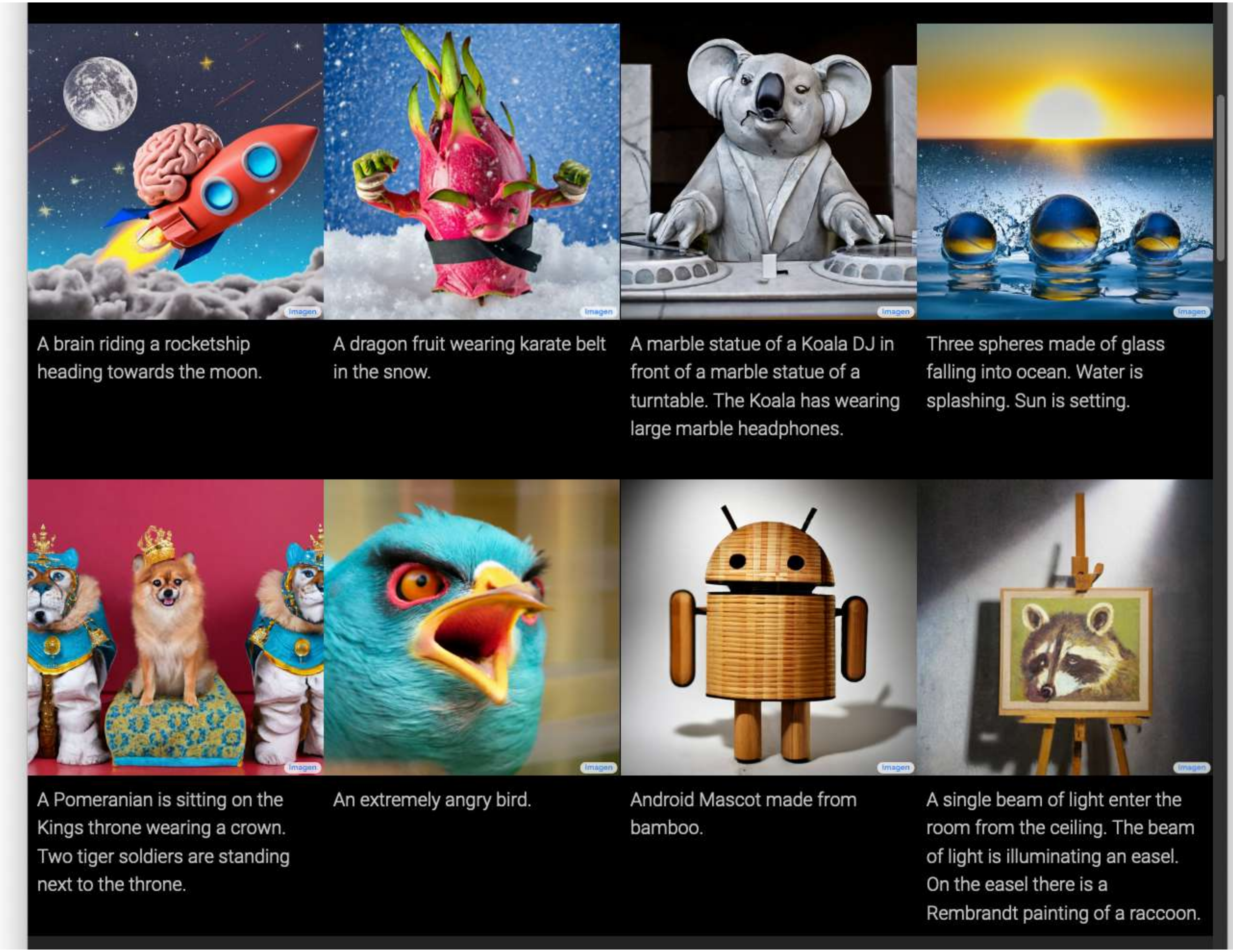
2. Algorithms and representations to allow a machine to recognize objects, people, scenes, and activities (***perception and interpretation***)

3. Algorithms to mine, search, and interact with visual data (***search and organization***)

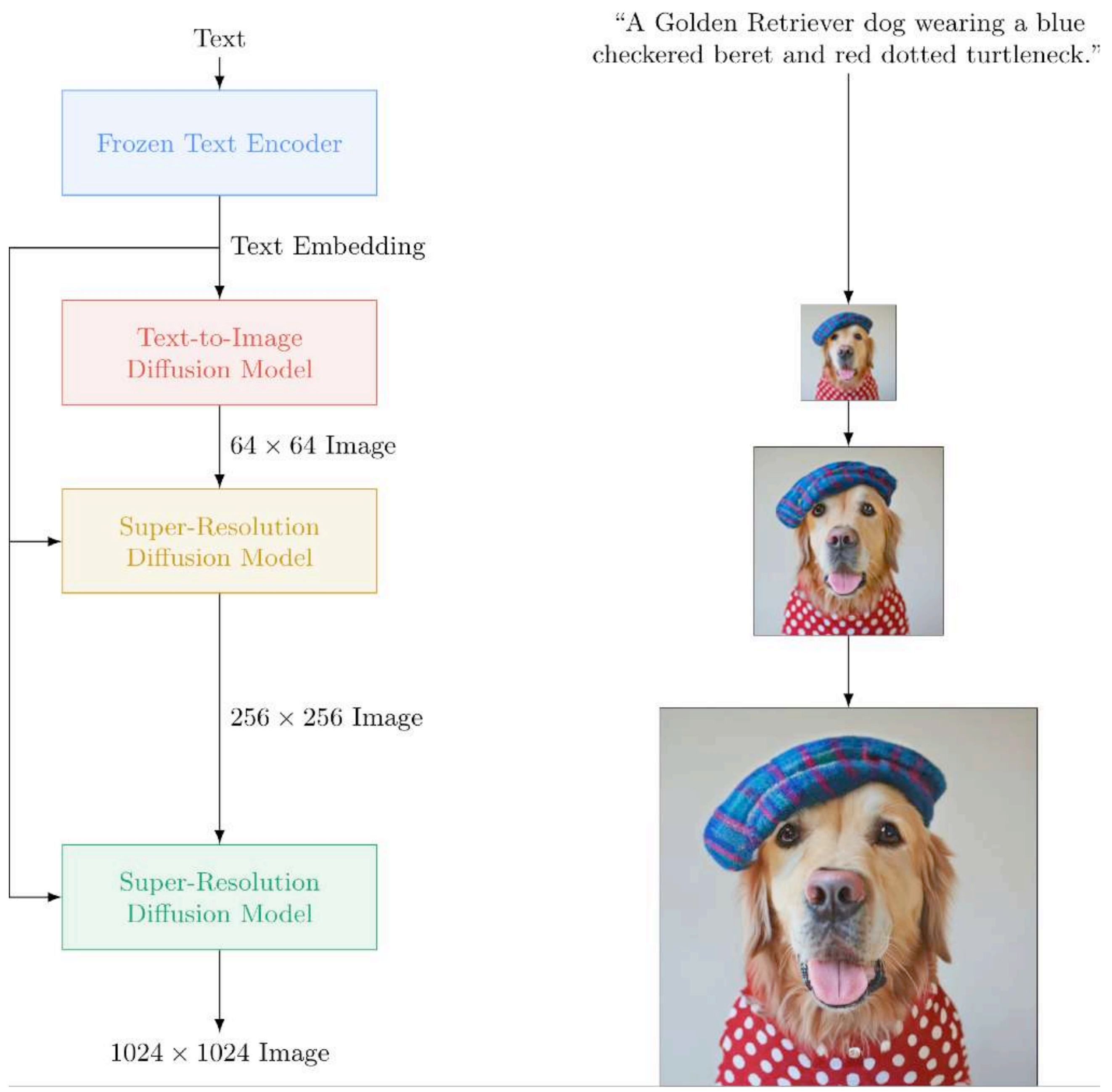
4. Algorithms for manipulation or creation of image or video content (***visual imagination***)

Slide Credit: Kristen Grauman (UT Austin)

4. Visual **Imagination**



- imagen.research.google
- Text to image generation
- Uses diffusion process, training using large dataset of text (web scale) and image-text (400M) pairs



Computer **Vision Problems**

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***)

3. Algorithms to mine, search, and interact with visual data (***search and organization***)

4. Algorithms for manipulation or creation of image or video content (***visual imagination***)

Slide Credit: Kristen Grauman (UT Austin)

Challenges: Viewpoint invariance

Optional subtitle

Michelangelo 1475-1564

*slide credit Fei-Fei, Fergus & Torralba

Challenges: Lighting

Optional subtitle

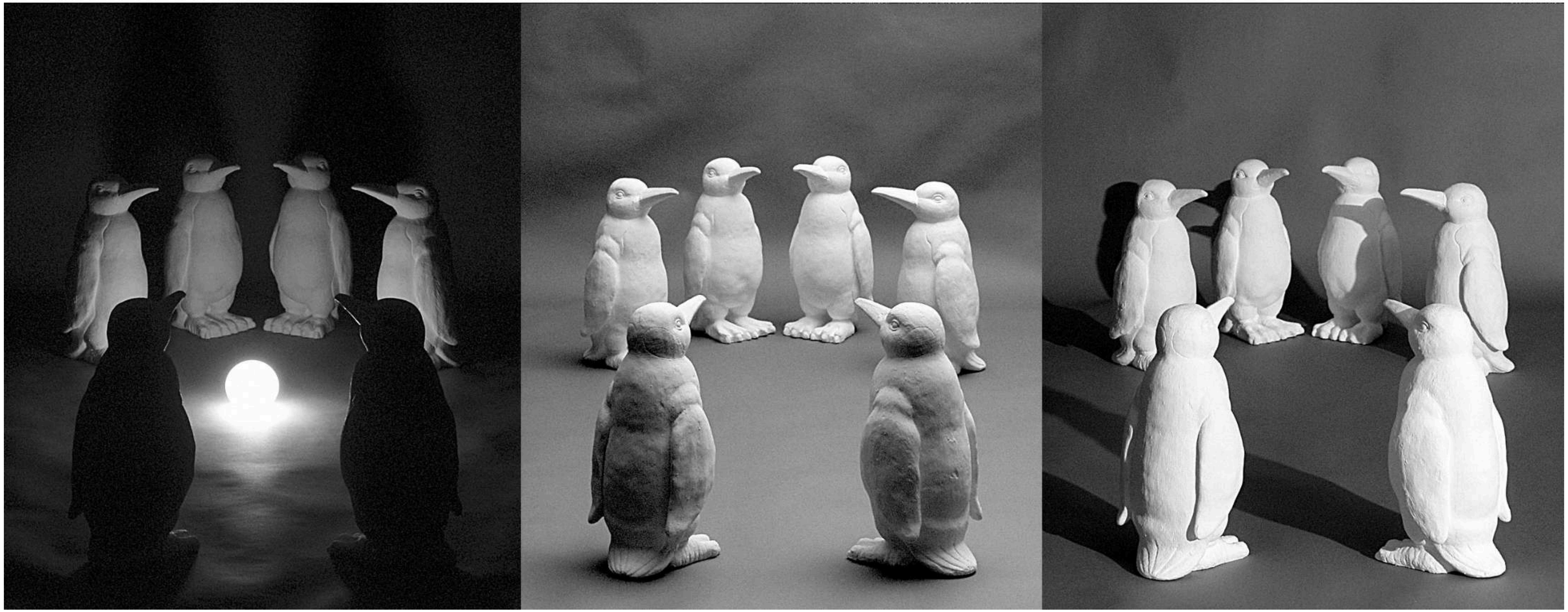


*image credit J. Koenderink

Challenges: Scale

Optional subtitle

*slide credit Fei-Fei, Fergus & Torralba

Challenges: Deformation

Optional subtitle

*image credit Peter Meer

Challenges: Occlusions

Optional subtitle

Rene Magritte 1965

Challenges: Background clutter

Optional subtitle

Kilmeny Niland 1995

Challenges: Local ambiguity and context

Optional subtitle

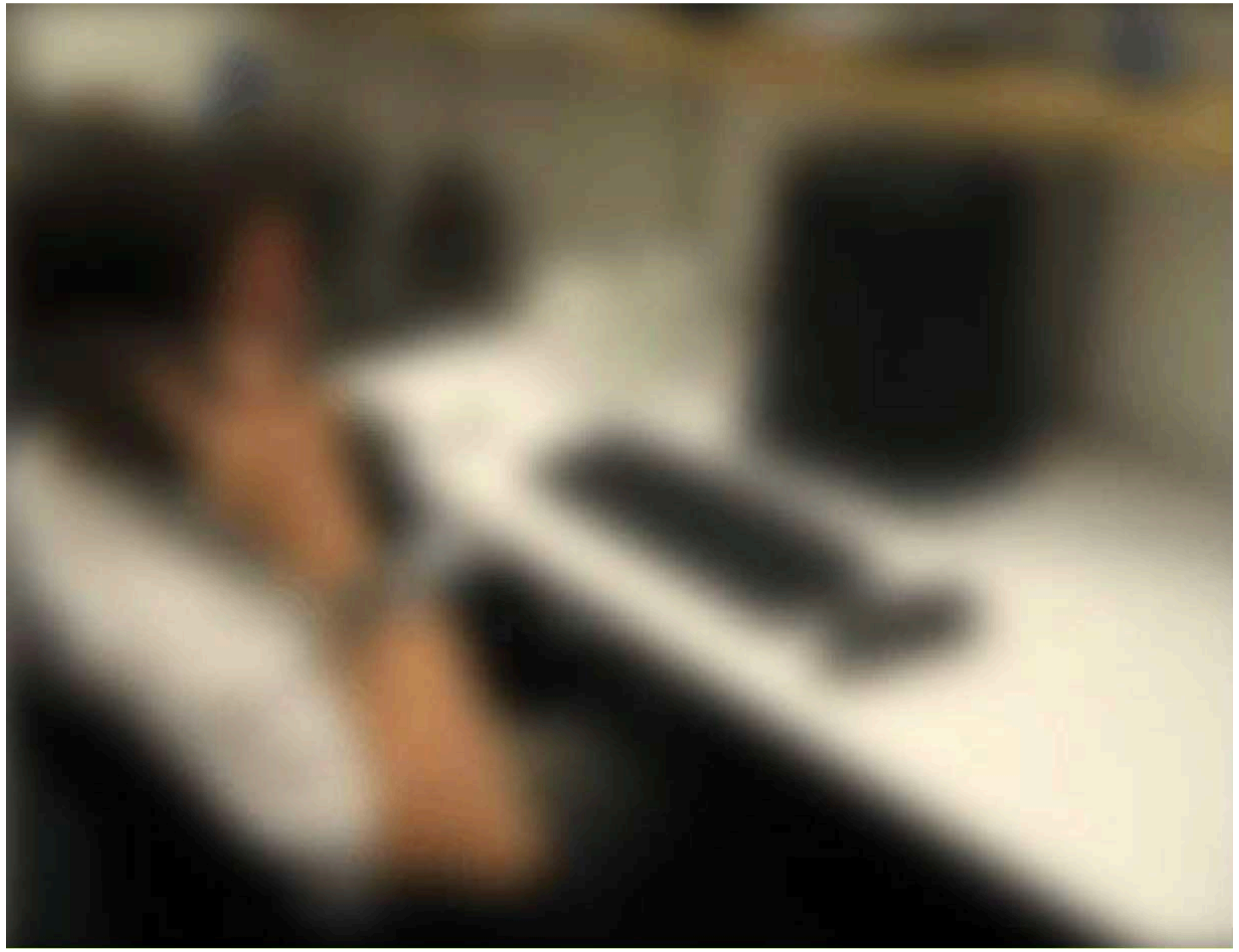


*image credit Fergus & Torralba

Challenges: Local ambiguity and context

Optional subtitle

*image credit Fergus & Torralba

Challenges: Motion

Optional subtitle

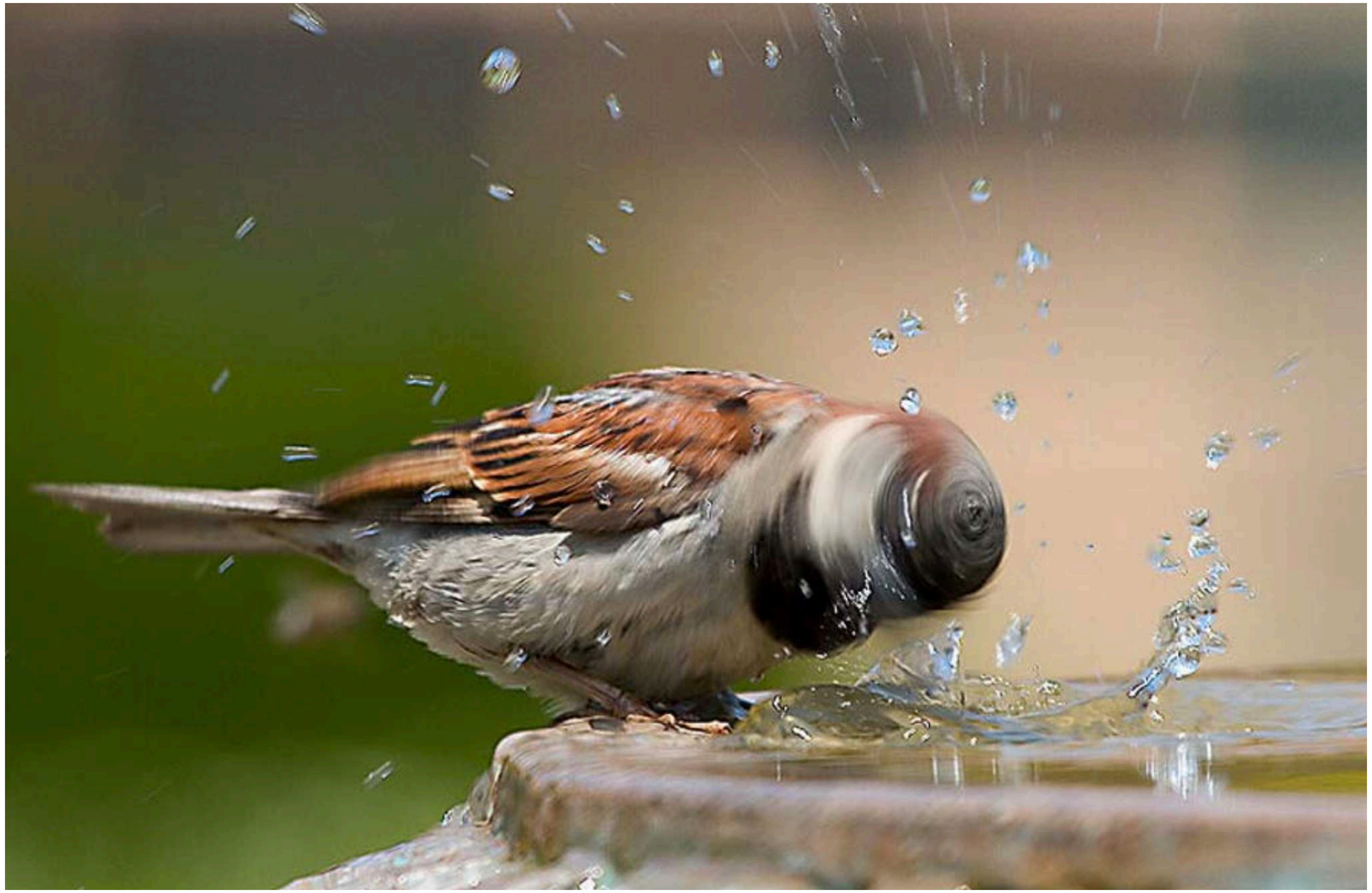


*image credit Peter Meer

Challenges: Object inter-class variation

Optional subtitle






*slide credit Fei-Fei, Fergus & Torralba

Computer Vision **Applications**

- Let’s see some examples of state-of-the-art and where it is used

Face Detection

[Motorola]

Face **Recognition**

Facebook

Apple's iPhoto

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

Slide Credit: Devi Parikh (GA Tech) and Fei-Fei Li (Stanford)

Vision for **Biometrics**

Fingerprint scanners on many new laptops, other devices

iPhone X Face ID

Face recognition systems are not part of widely used technologies

How it works and how to fool it:

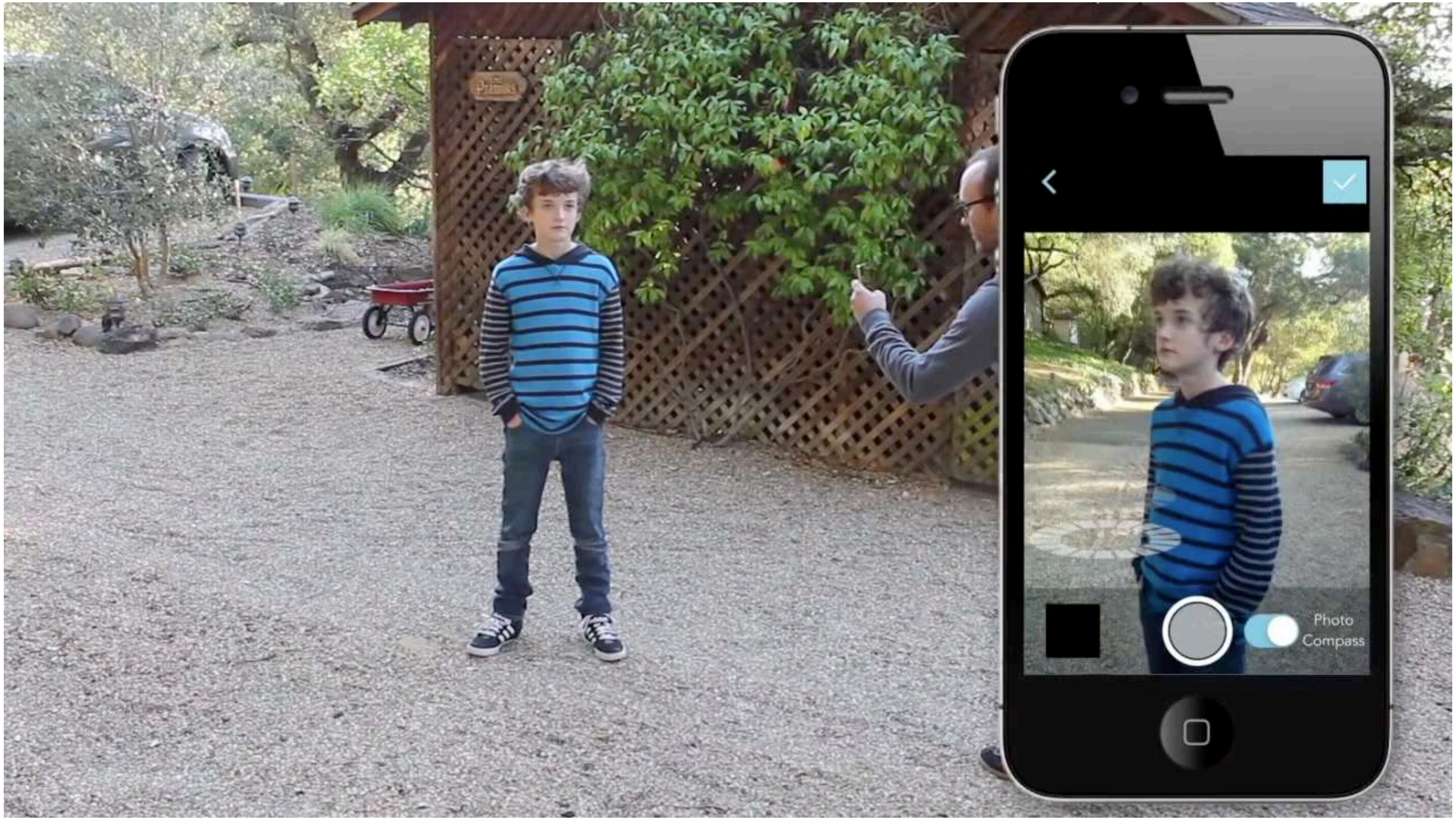
https://www.youtube.com/watch?v=FhbMLmsCax0

Image Credit: James Hays (GA Tech)

Camera Tracking

[Boujou — Vicon / OMG]

3D Reconstruction



[Autodesk 123D Catch]

3D Reconstruction

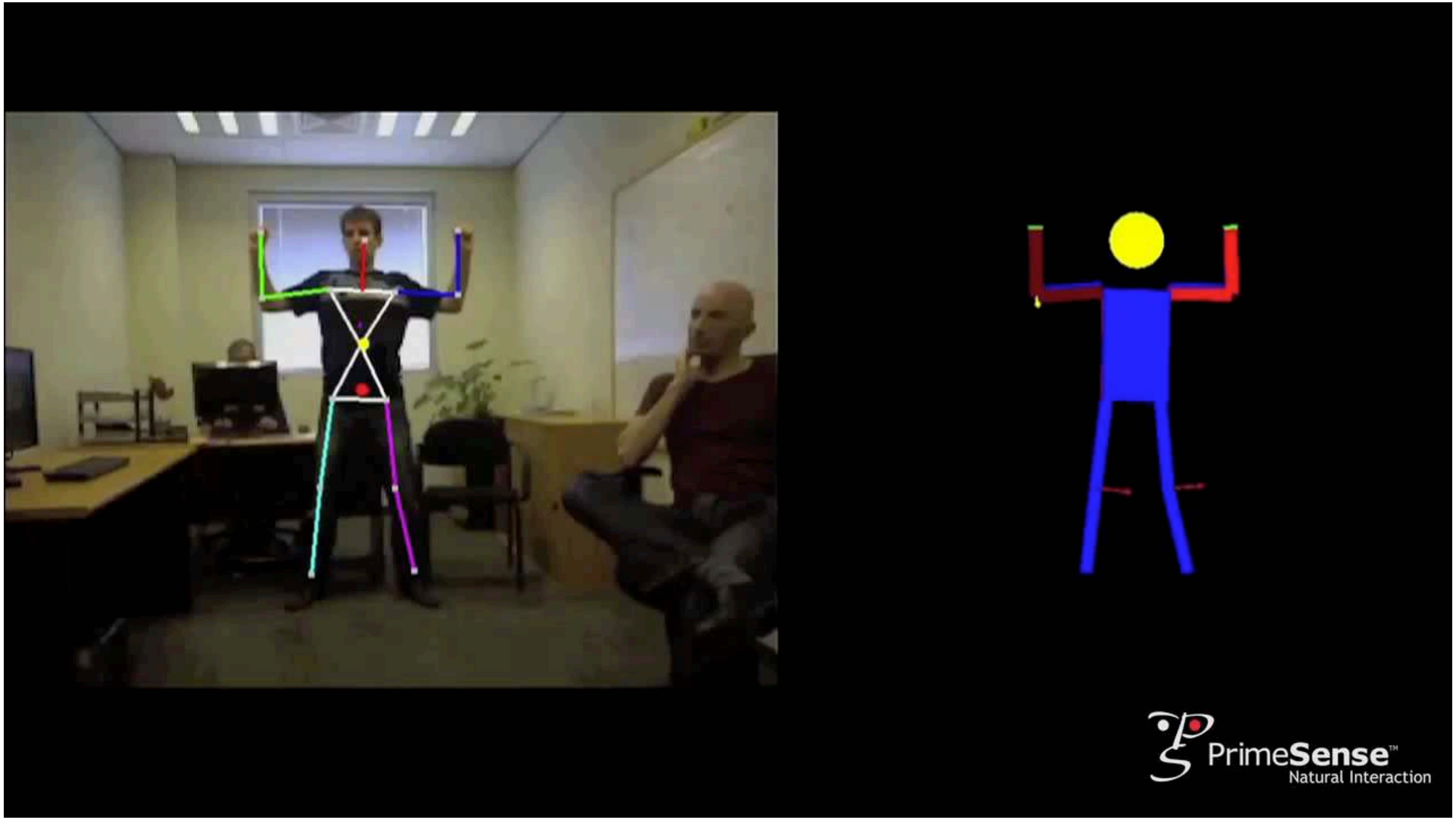
[Apple 3D Photo]

https://support.apple.com/en-ca/guide/apple-vision-pro/tan1be9a3a0b/visionos

Body Pose Tracking

[Microsoft Xbox Kinect]

Body Pose Tracking



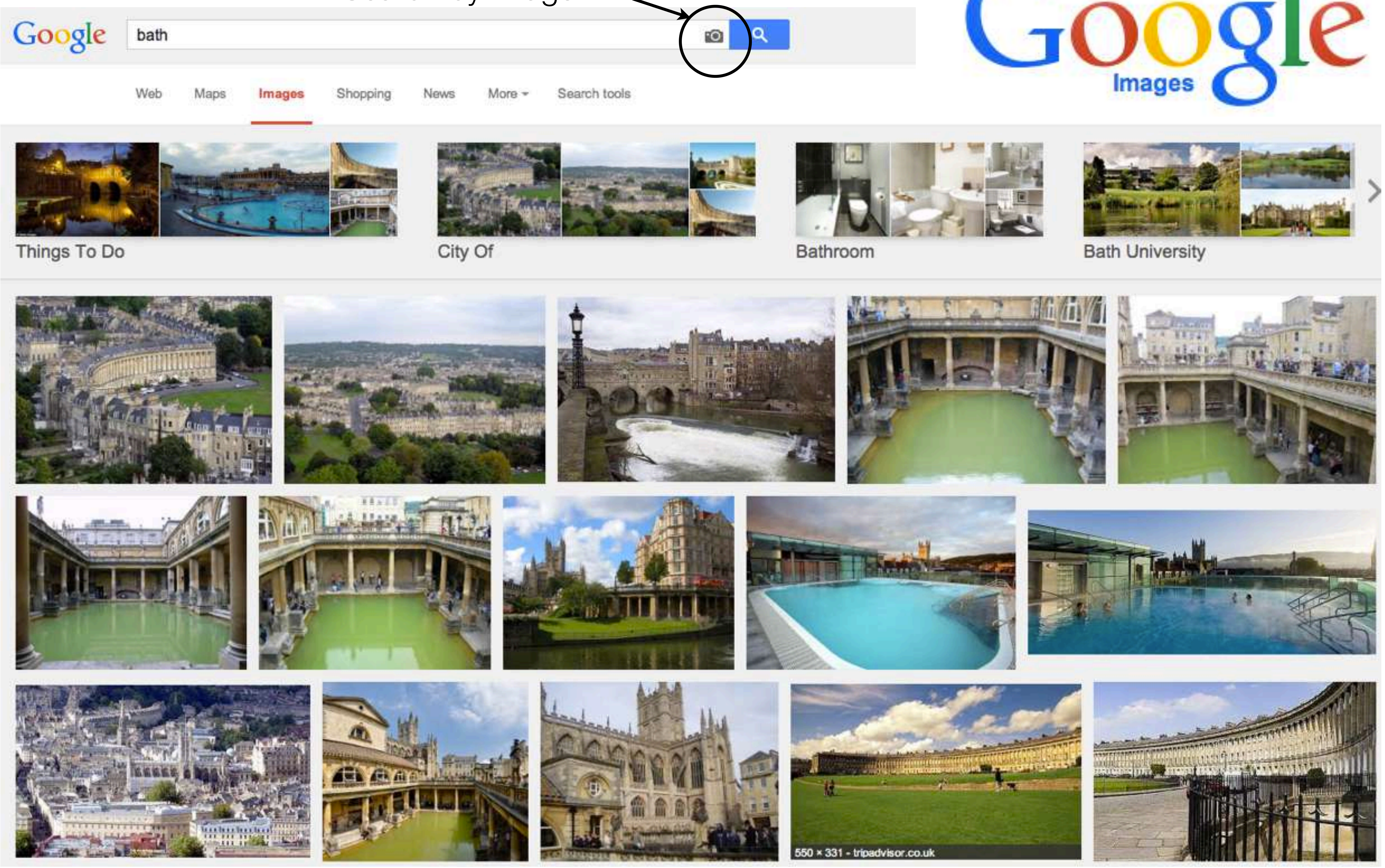
[PrimeSense]

Body Pose Tracking

https://shubham-goel.github.io/4dhumans/

Image Recognition and Search

Search by image



Self-Driving Cars

[Google]

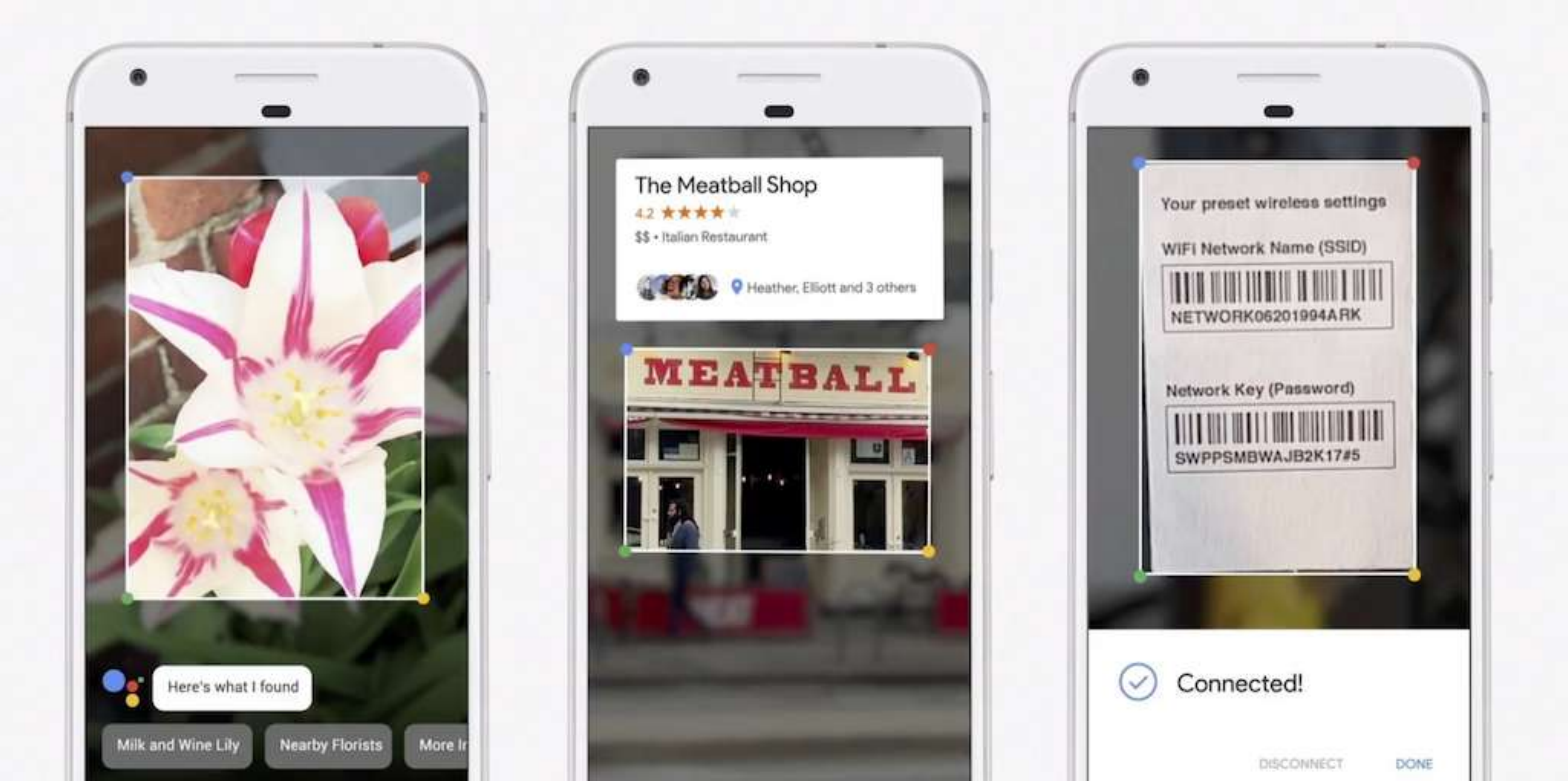
Flying Vehicles

www.skydio.com

AR / VR

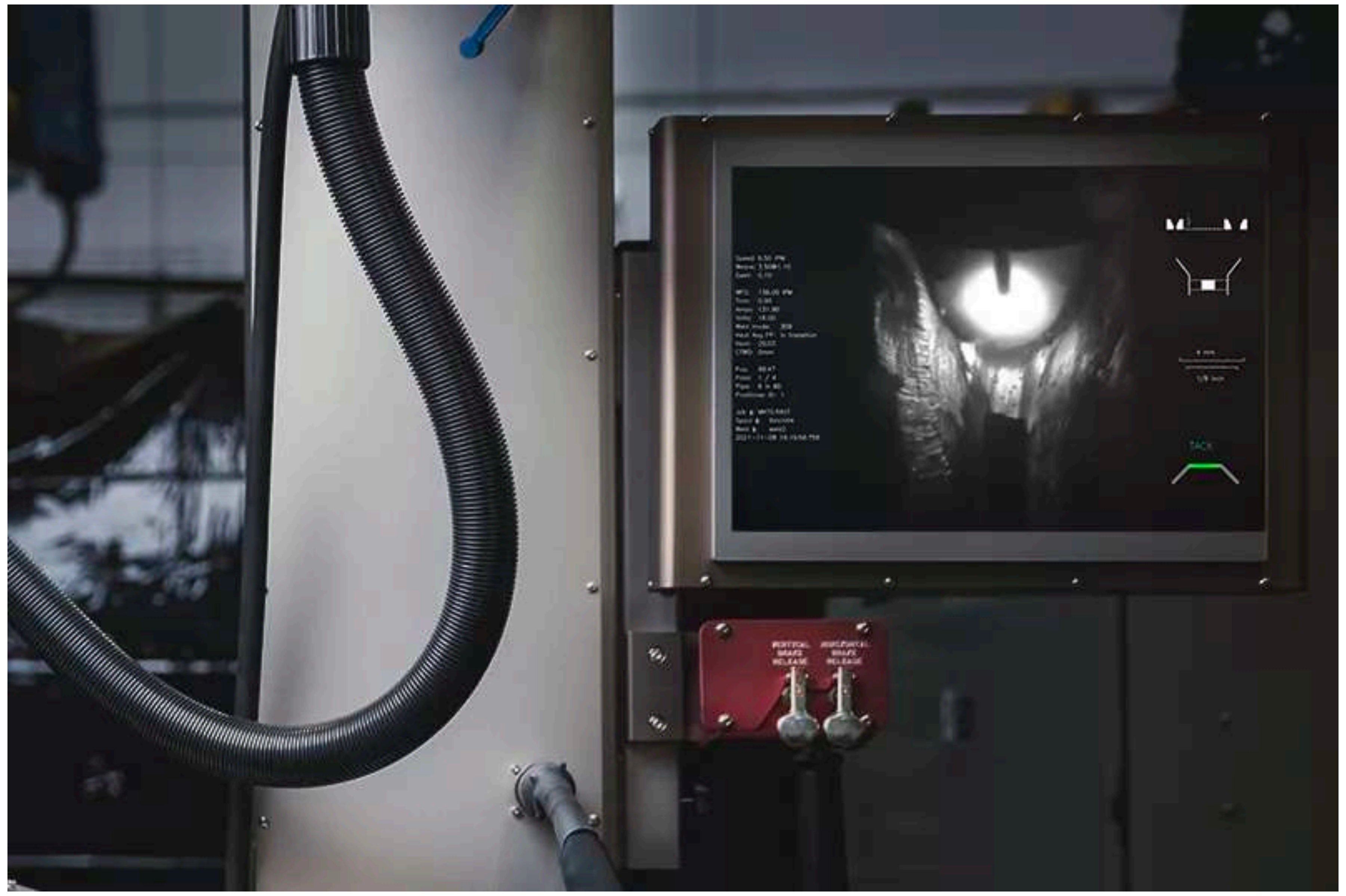
[Microsoft HoloLens]

Mobile Apps



[Google Lens]

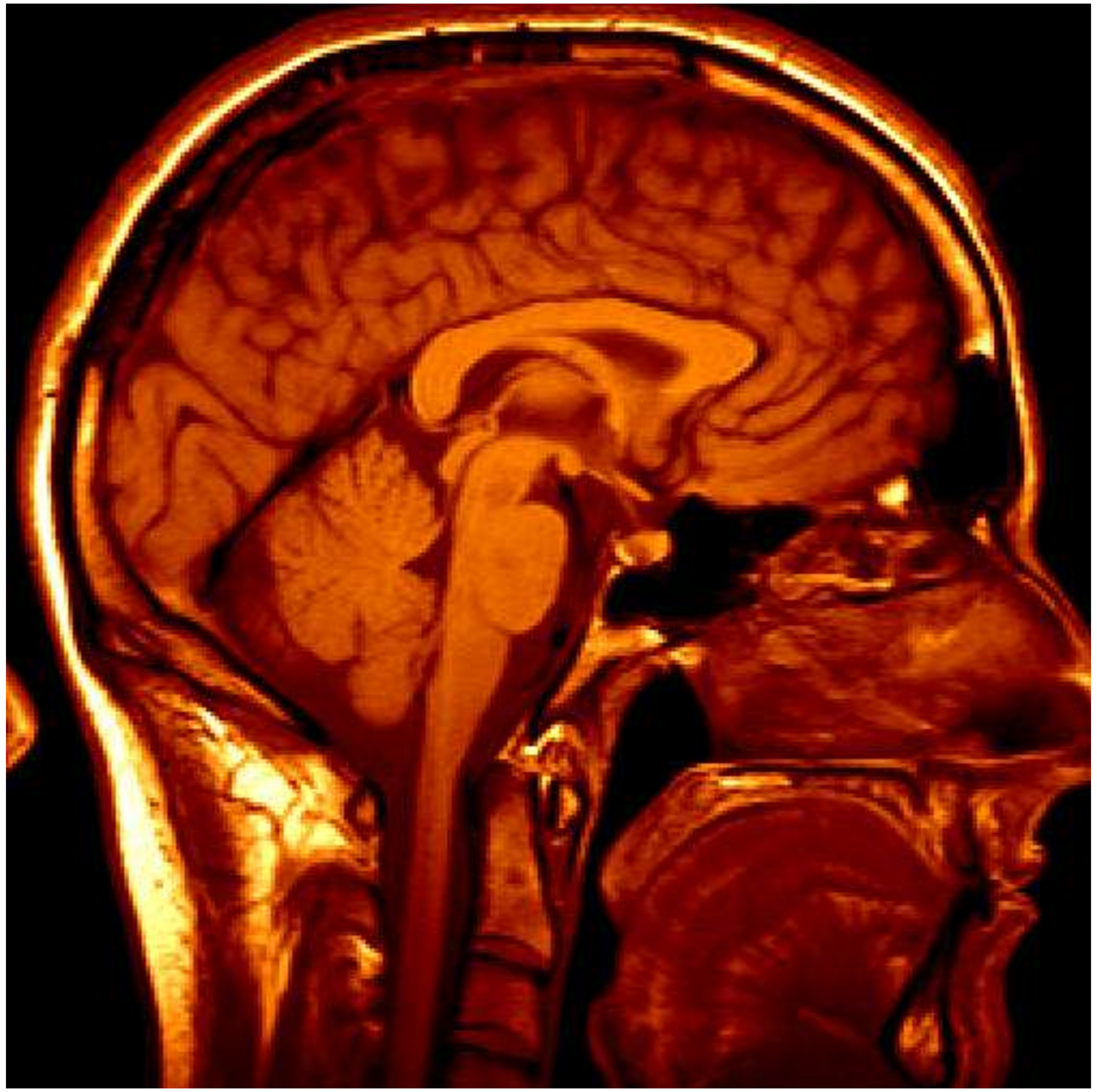
Industrial



Machine Vision controlled welding robotics

NOVARC TECHNOLOGIES

Medicine



3D imaging
MRI, CT

Image guided surgery
[Grimson et al., MIT]

Slide Credit: James Hays (GA Tech)

Art

[Gatys, Ecker, Bethge 2015]

Art

TEXT DESCRIPTION

An astronaut Teddy bears A bowl of soup

riding a horse lounging in a tropical resort in space playing basketball with cats in space

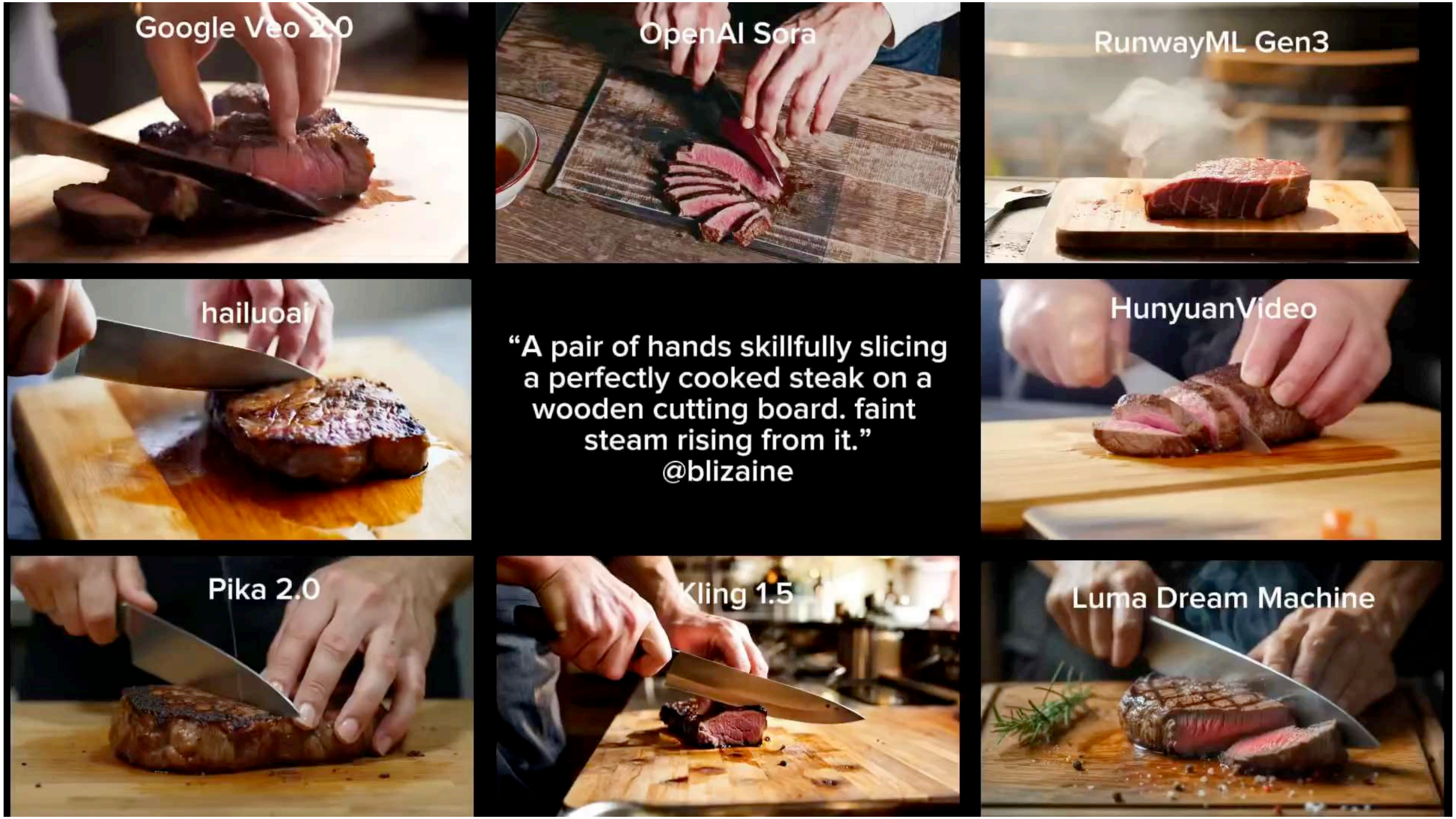
in a photorealistic style in the style of Andy Warhol as a pencil drawing

→

DALL·E 2

[Dall-E v2]

Videos



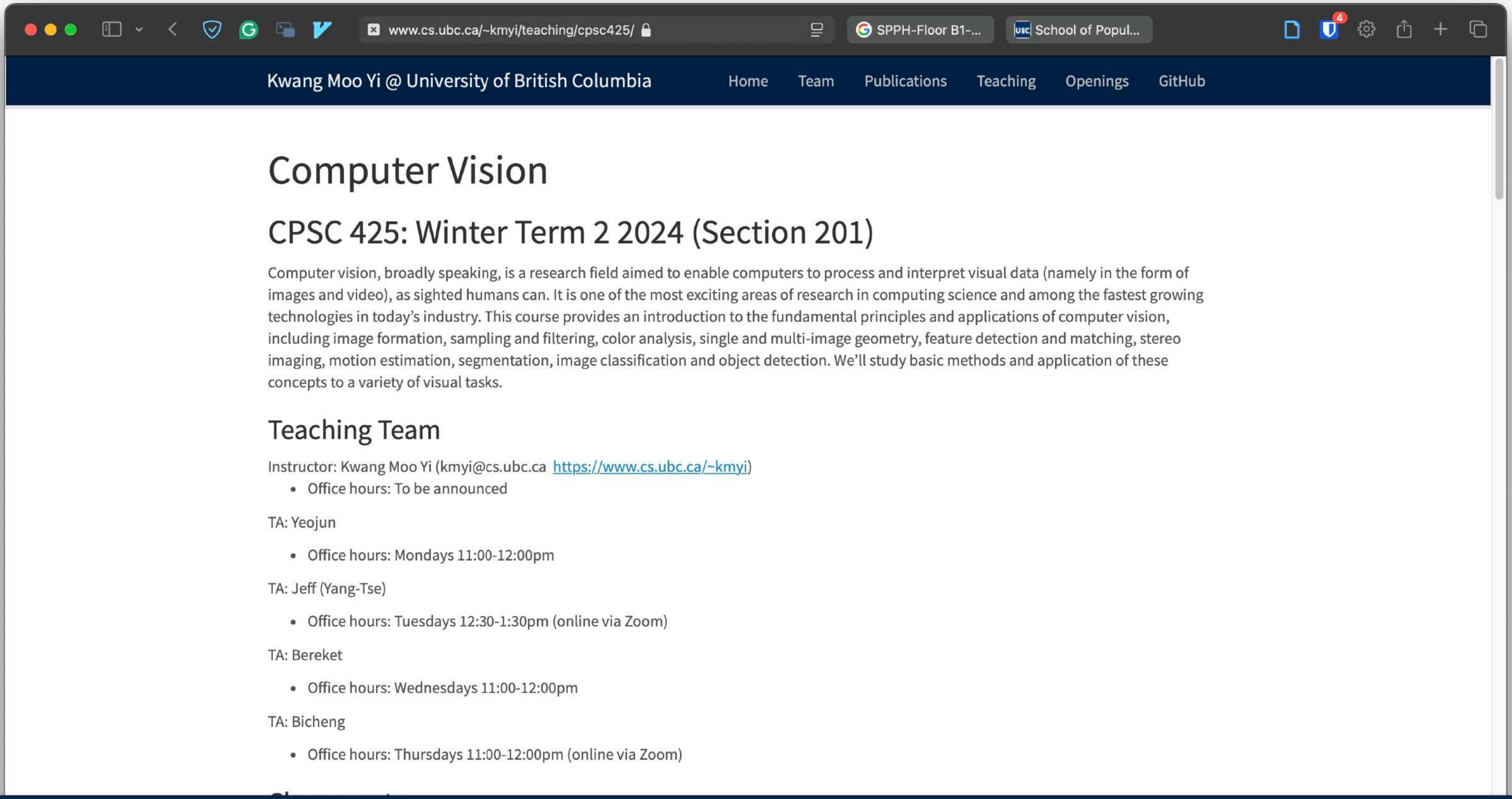
https://x.com/blizaine/status/1868850653759783033

Why Study Computer Vision?

It is one of the **most exciting areas of research** in computer science

Among the **fastest growing technologies** in the industry today

Course Webpage



www.cs.ubc.ca/~kmyi/teaching/cpsc425/

Kwang Moo Yi @ University of British Columbia   Home   Team   Publications   Teaching   Openings   GitHub

Computer Vision

CPSC 425: Winter Term 2 2024 (Section 201)

Computer vision, broadly speaking, is a research field aimed to enable computers to process and interpret visual data (namely in the form of images and video), as sighted humans can. It is one of the most exciting areas of research in computing science and among the fastest growing technologies in today's industry. This course provides an introduction to the fundamental principles and applications of computer vision, including image formation, sampling and filtering, color analysis, single and multi-image geometry, feature detection and matching, stereo imaging, motion estimation, segmentation, image classification and object detection. We'll study basic methods and application of these concepts to a variety of visual tasks.

Teaching Team

Instructor: Kwang Moo Yi (kmyi@cs.ubc.ca https://www.cs.ubc.ca/~kmyi)
- Office hours: To be announced

TA: Yeojun
- Office hours: Mondays 11:00-12:00pm

TA: Jeff (Yang-Tse)
- Office hours: Tuesdays 12:30-1:30pm (online via Zoom)

TA: Bereket
- Office hours: Wednesdays 11:00-12:00pm

TA: Bicheng
- Office hours: Thursdays 11:00-12:00pm (online via Zoom)

- Schedule, Assignments
- Lecture Slides and Notes
- Course Information (public)

https://www.cs.ubc.ca/~kmyi/teaching/cpsc425

Topics Covered

– Image Processing (Linear Filtering, Convolution)
– Filters as Templates
– Image Feature Detection (Edges & Corners)
– Texture & Colour
– Image Feature Description (SIFT)
– Model Fitting (RANSAC, The Hough Transform)
– Camera Models, Stereo Geometry
– Motion and Optical Flow
– Clustering and Image Segmentation
– Learning and Image Classification
– Deep Learning Introduction

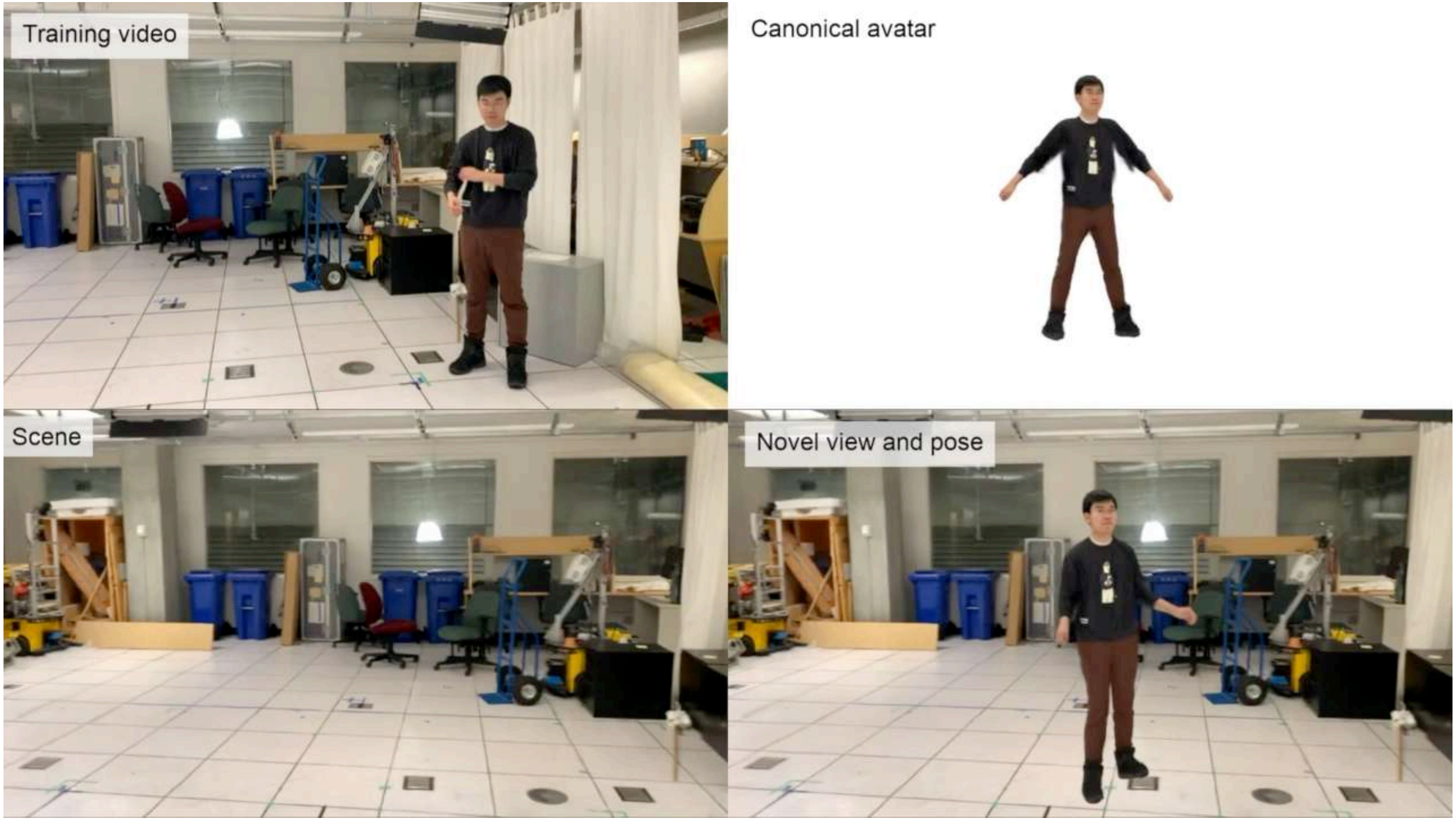
Topics Covered

– Image Processing (Linear Filtering, Convolution)
– Filters as Templates
– Image Feature Detection (Edges & Corners)
– Texture & Colour
– Image Feature Description (SIFT)
– Model Fitting (RANSAC, The Hough Transform)
– Camera Models, Stereo Geometry
– Motion and Optical Flow
– Clustering and Image Segmentation
– Learning and Image Classification
– Deep Learning Introduction

A “NeRF” commercial (2023)

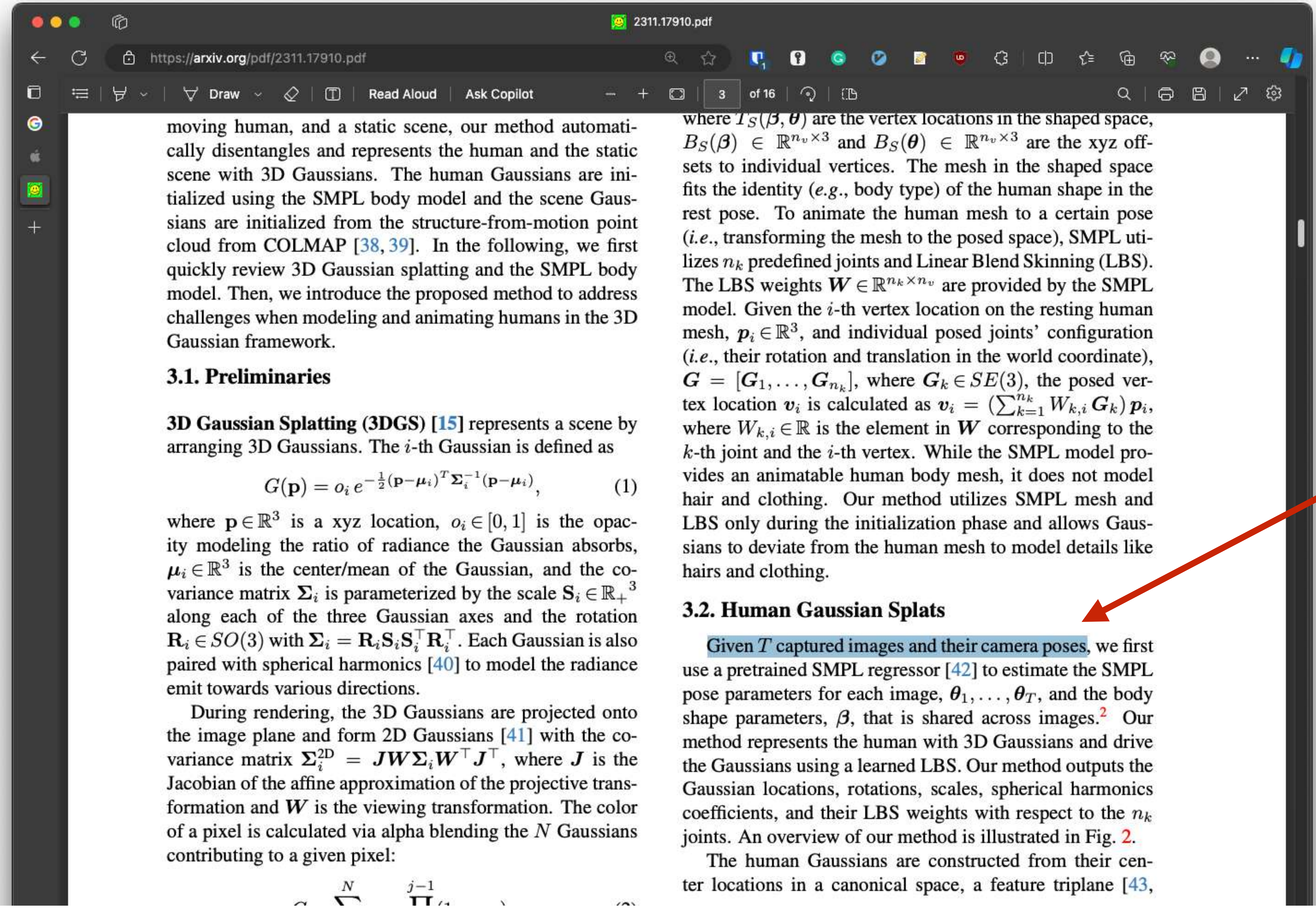
[Video from https://twitter.com/karenxcheng/status/1615404573367361542] reproduced for educational purposes]

Animatable “avatars” from a video



[Video from https://machinelearning.apple.com/research/hugs reproduced for educational purposes]

Behind the scenes, they still rely on traditional stuff



moving human, and a static scene, our method automatically disentangles and represents the human and the static scene with 3D Gaussians. The human Gaussians are initialized using the SMPL body model and the scene Gaussians are initialized from the structure-from-motion point cloud from COLMAP [38, 39]. In the following, we first quickly review 3D Gaussian splatting and the SMPL body model. Then, we introduce the proposed method to address challenges when modeling and animating humans in the 3D Gaussian framework.

3.1. Preliminaries

3D Gaussian Splatting (3DGS) [15] represents a scene by arranging 3D Gaussians. The i-th Gaussian is defined as

$$G(\mathbf{p}) = o_i \, e^{-\frac{1}{2}(\mathbf{p}-\boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1}(\mathbf{p}-\boldsymbol{\mu}_i)}, \quad (1)$$

where $\mathbf{p} \in \mathbb{R}^3$ is a xyz location, $o_i \in [0,1]$ is the opacity modeling the ratio of radiance the Gaussian absorbs, $\boldsymbol{\mu}_i \in \mathbb{R}^3$ is the center/mean of the Gaussian, and the covariance matrix $\boldsymbol{\Sigma}_i$ is parameterized by the scale $\mathbf{S}_i \in \mathbb{R}_+{}^3$ along each of the three Gaussian axes and the rotation $\mathbf{R}_i \in SO(3)$ with $\boldsymbol{\Sigma}_i = \mathbf{R}_i \mathbf{S}_i \mathbf{S}_i^\top \mathbf{R}_i^\top$. Each Gaussian is also paired with spherical harmonics [40] to model the radiance emit towards various directions.

During rendering, the 3D Gaussians are projected onto the image plane and form 2D Gaussians [41] with the covariance matrix $\boldsymbol{\Sigma}_i^{2D} = \boldsymbol{J}\boldsymbol{W}\boldsymbol{\Sigma}_i\boldsymbol{W}^\top\boldsymbol{J}^\top$, where $\boldsymbol{J}$ is the Jacobian of the affine approximation of the projective transformation and $\boldsymbol{W}$ is the viewing transformation. The color of a pixel is calculated via alpha blending the N Gaussians contributing to a given pixel:

where $T_S(\boldsymbol{\beta}, \boldsymbol{\theta})$ are the vertex locations in the shaped space, $B_S(\boldsymbol{\beta}) \in \mathbb{R}^{n_v \times 3}$ and $B_S(\boldsymbol{\theta}) \in \mathbb{R}^{n_v \times 3}$ are the xyz offsets to individual vertices. The mesh in the shaped space fits the identity (*e.g.*, body type) of the human shape in the rest pose. To animate the human mesh to a certain pose (*i.e.*, transforming the mesh to the posed space), SMPL utilizes n_k predefined joints and Linear Blend Skinning (LBS). The LBS weights $\boldsymbol{W} \in \mathbb{R}^{n_k \times n_v}$ are provided by the SMPL model. Given the i-th vertex location on the resting human mesh, $\boldsymbol{p}_i \in \mathbb{R}^3$, and individual posed joints' configuration (*i.e.*, their rotation and translation in the world coordinate), $\boldsymbol{G} = [\boldsymbol{G}_1, \ldots, \boldsymbol{G}_{n_k}]$, where $\boldsymbol{G}_k \in SE(3)$, the posed vertex location $\boldsymbol{v}_i$ is calculated as $\boldsymbol{v}_i = (\sum_{k=1}^{n_k} W_{k,i} \boldsymbol{G}_k)\boldsymbol{p}_i$, where $W_{k,i} \in \mathbb{R}$ is the element in $\boldsymbol{W}$ corresponding to the k-th joint and the i-th vertex. While the SMPL model provides an animatable human body mesh, it does not model hair and clothing. Our method utilizes SMPL mesh and LBS only during the initialization phase and allows Gaussians to deviate from the human mesh to model details like hairs and clothing.

3.2. Human Gaussian Splats

Given T captured images and their camera poses, we first use a pretrained SMPL regressor [42] to estimate the SMPL pose parameters for each image, $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_T$, and the body shape parameters, $\boldsymbol{\beta}$, that is shared across images.[2] Our method represents the human with 3D Gaussians and drive the Gaussians using a learned LBS. Our method outputs the Gaussian locations, rotations, scales, spherical harmonics coefficients, and their LBS weights with respect to the n_k joints. An overview of our method is illustrated in Fig. 2.

The human Gaussians are constructed from their center locations in a canonical space, a feature triplane [43,

- Image Processing (Linear Filtering, Convolution)
- Filters as Templates
- Image Feature Detection (Edges & Corners)
- Texture & Colour
- Image Feature Description (SIFT)
- Model Fitting (RANSAC, The Hough Transform)
- Camera Models, Stereo Geometry
- Motion and Optical Flow
- Clustering and Image Segmentation
- Learning and Image Classification
- Deep Learning Introduction

Course Origins

CPSC 425 was originally developed by **Bob Woodham** and has evolved over the years. Much of the material this year is adapted from material prepared by Bob, as well as extensions developed by others who taught this course

Previously taught by:

— 2024-2025 Term 1 by **Leonid Sigal**
— 2023-2024 Term 2 by **Kwang Moo Yi**
— 2023-2024 Term 1 by **Matthew Brown**
— 2022-2023 Term 2 by **Leonid Sigal**
— 2022-2023 Term 1 by **Matthew Brown**
— 2021-2022 Term 1 & 2 by **Jim Little**
— 2020-2021 Term 1 by **Leonid Sigal**
— 2019-2020 Term 2 by **Leonid Sigal**
— 2019-2020 Term 1 by **Jim Little**
— 2018-2019 Term 1 & 2 by **Leonid Sigal**
— 2016-2017 Term 2 by **Jim Little**
— 2015-2016 Term 2 by **Fred Tung**
— 2015-2015 Term 2 by **Jim Little**

How to **Learn** from the **Course**?

— The course is very **broad**, but relatively **shallow** introduction to a very diverse and complex field that draws material from geometry, statistics, AI, machine learning, computer graphics, psychology and many others.

— It is easy to think that material is easy and course requires no studying

— Part of your job should be going over the slides and carefully analyzing not just what is on them, but the underlying assumptions, algorithmic steps and so on

— Don’t strive for “**template matching**” strive for true “**understanding**”

Grading Criteria

In-class **quiz**: 10%

Programming Assignments: 40%

6 graded and 1 ungraded (optional) assignment

Midterm Exam (February 24th): 15%

Final Exam (April 16 — 27): 35%

iClicker Setup

Quizzes will be run via **iClicker**

Please make sure you have an iClicker account with your student ID:

https://lthub.ubc.ca/guides/iclicker-cloud-student-guide/

You should set **UBC** as the **institution**, use the **same email** as for your **canvas** account, and enter your **student number** in the student ID field.

You should be automatically added and the course:

CPSC 425 101 2024W2 Computer Vision Section 201

iClicker Quizzes

Setup before class! We'll do a test next week

Join the class at student.iclicker.com

There will be around 6 multiple choice questions per quiz
— 1/2 point for participation
— 1/2 point for correct answer
*not all clicker quizzes are worth the same # of points, depends on # of questions.

The clicker questions contribute <u>10%</u> to your total grade

Missing Quiz Policy: If you miss a quiz for a legitimate and documented reason, that quiz will be dropped (legitimate reasons: illness, conference travel, etc.) You are required to contact instructor and provide proof within 1 week of missed quiz.

Assignments

There will be **6+1 assignments** in total (6 marked)

— Approximately 1 every 2 weeks

— You will hand these in by 11:59pm on the due date (read hand in instructions and late policy on course webpage

— To be done individually by each student

You will use the **Python**, with the following libraries: Python Imaging Library (PIL), NumPy, Matplotlib, SciPy, Scikit-Learn

— Assignment 0 (which is ungraded) will introduce you to this.

Assignments contribute 40% to your final score

Midterm Exam

Scheduled for **February 24th**

— Here in class during the lecture period

— Closed book, no notes allowed

Multiple choice, true / false and short answer questions

— Aimed to test your “understanding” of the content of the course

The Midterm exam will contribute 15% to your final score

Final Exam

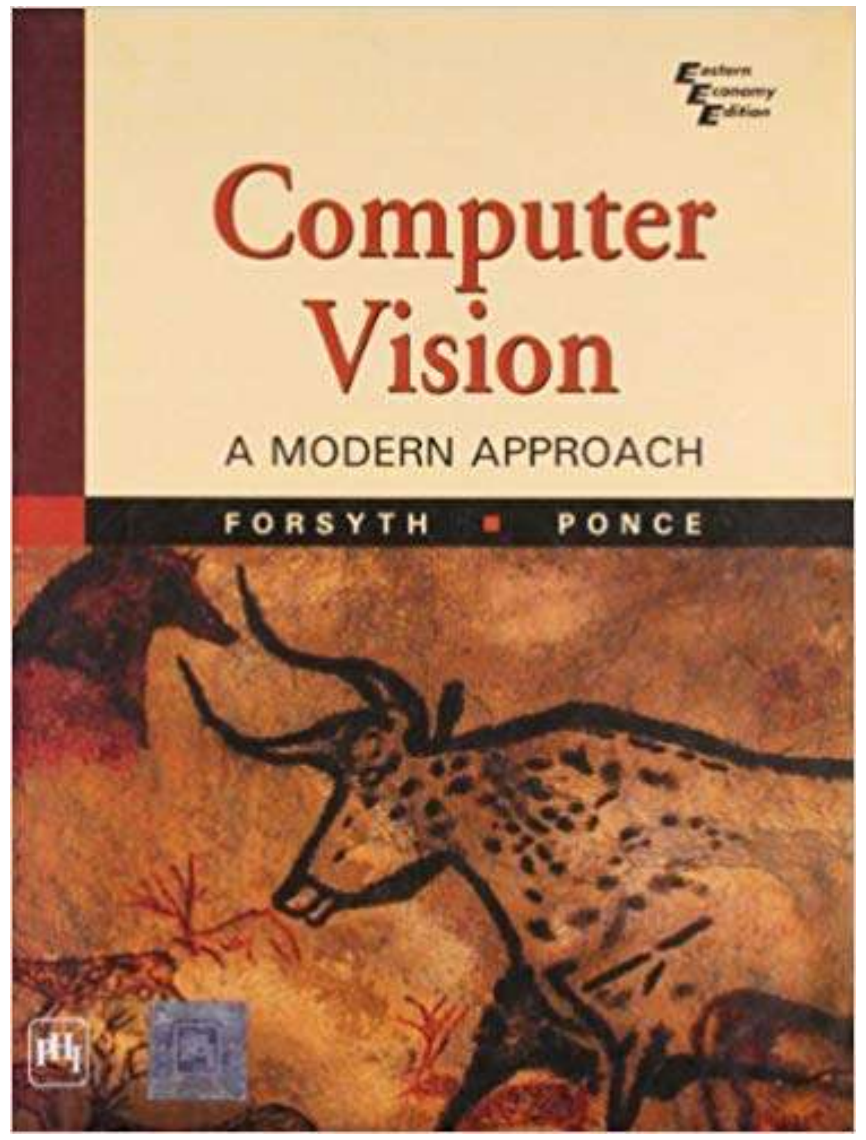
The Final exam is held during the regular examination period, **April 16-27**, and is scheduled by the Registrar's Office

Similar to the midterm but longer and with more extensive short/medium answer questions

The Final exam will contribute 35% to your final score

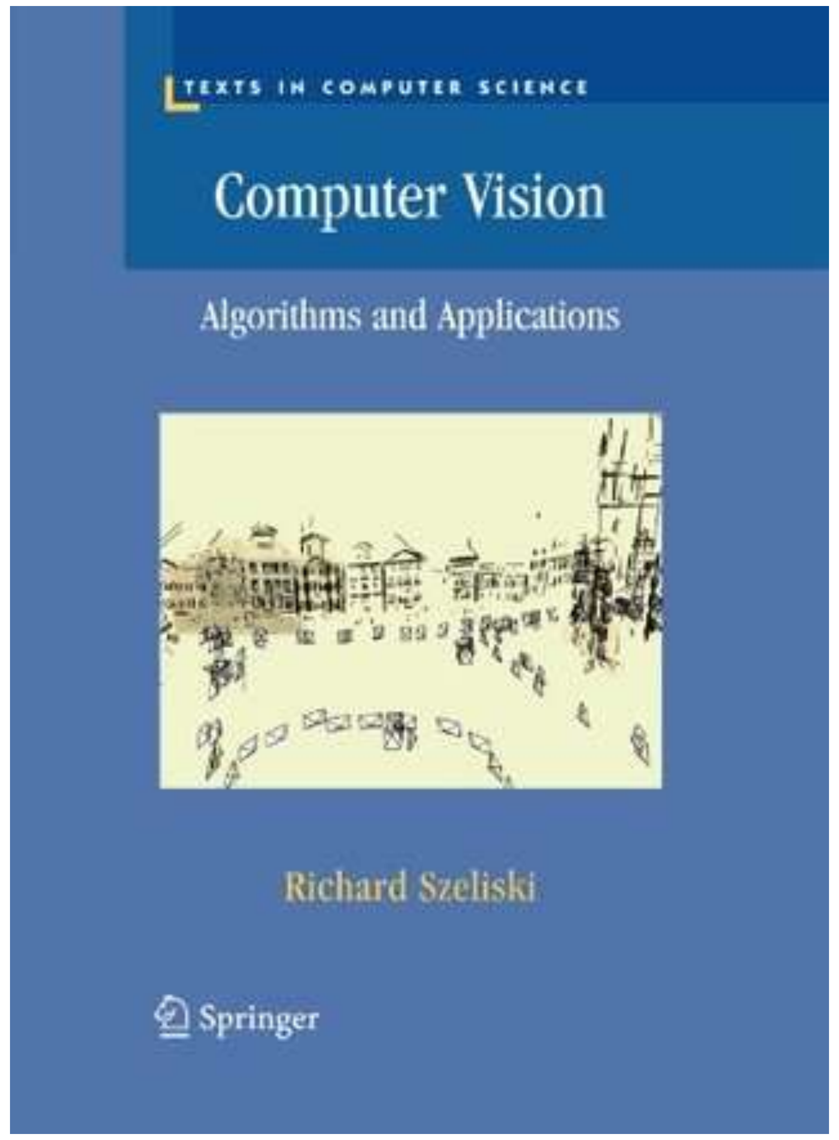
Textbooks

The course uses the following textbooks, which are recommended (but **not required**):



Computer Vision: A Modern Approach (2nd ed)

By: D. Forsyth & J. Ponce
Publisher: Pearson 2012



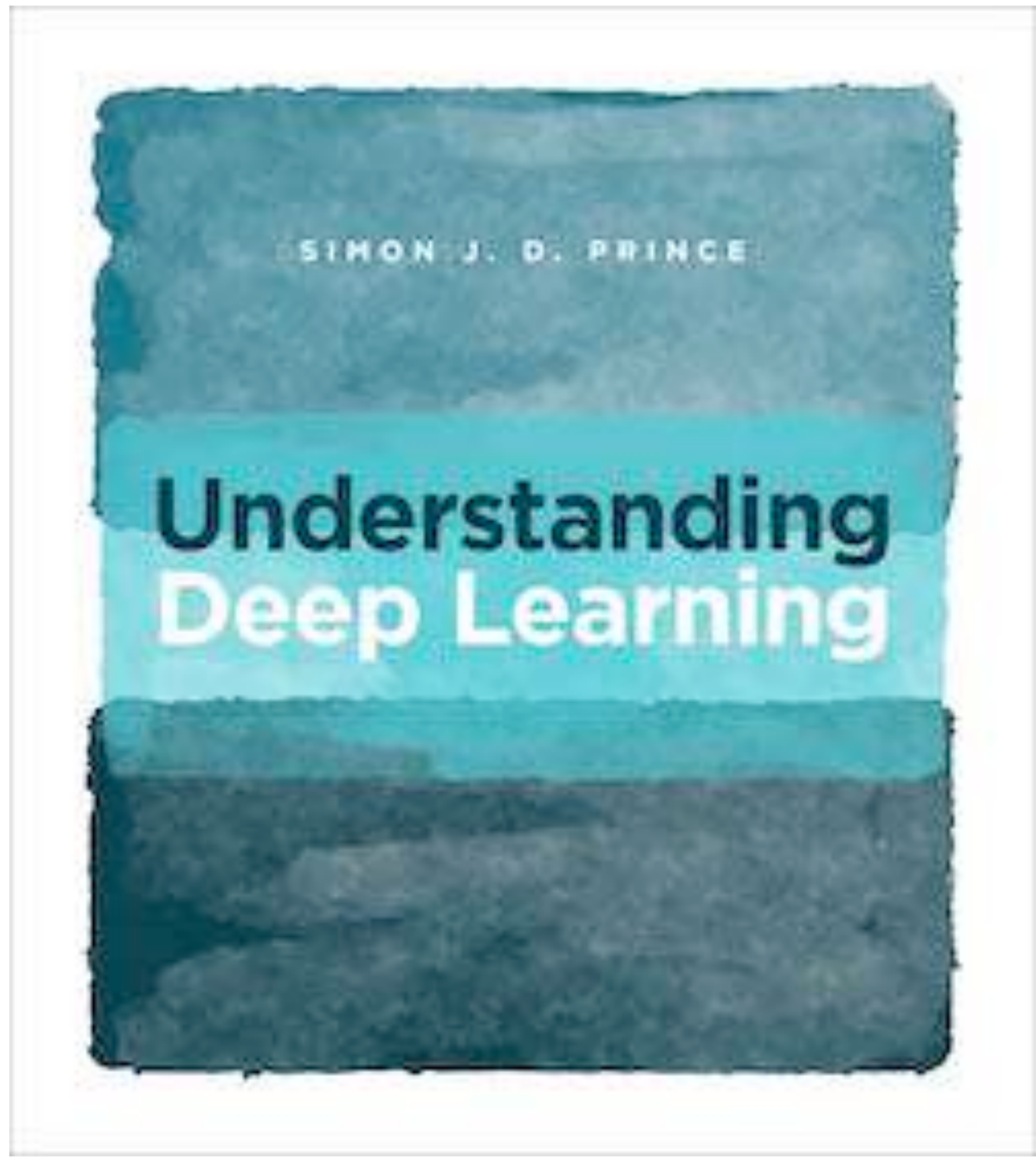
Computer Vision: Algorithms and Applications (2nd ed)

By: R. Szeliski
Publisher: Springer 2022

https://szeliski.org/Book/

Text**books**

The course uses the following textbooks, which are recommended (but **not required**):



Understanding Deep Learning

By: Simon J.D. Prince
Publisher: MIT Press 2023

https://udlbook.github.io/udlbook/

Readings

You will be assigned **readings**.

— Sometimes you will be assigned readings from other sources

Do the reading **after coming** to the lecture

— Reading assignments will be posted on course webpage

— They will also be mentioned in class