清華大學 Tsinghua University
交叉信息研究院 Institute for Interdisciplinary Information Sciences

Lecture 1: Introduction

Li Yi

2025.02.20

About me

Li Yi (弋力)

- 2009 - 2013, B.E. @ Tsinghua University
- 2013 - 2019, Ph.D. @ Stanford University
- 2019 - 2021, Research Scientist @ Google Research
- 2021 - now, Assistant Professor @ Tsinghua University
- Research: 3D Visual Computing and Embodied AI
- Homepage: https://ericyi.github.io/
- Email: ericyi0124@gmail.com

Course Staff

TA: Xueyi Liu

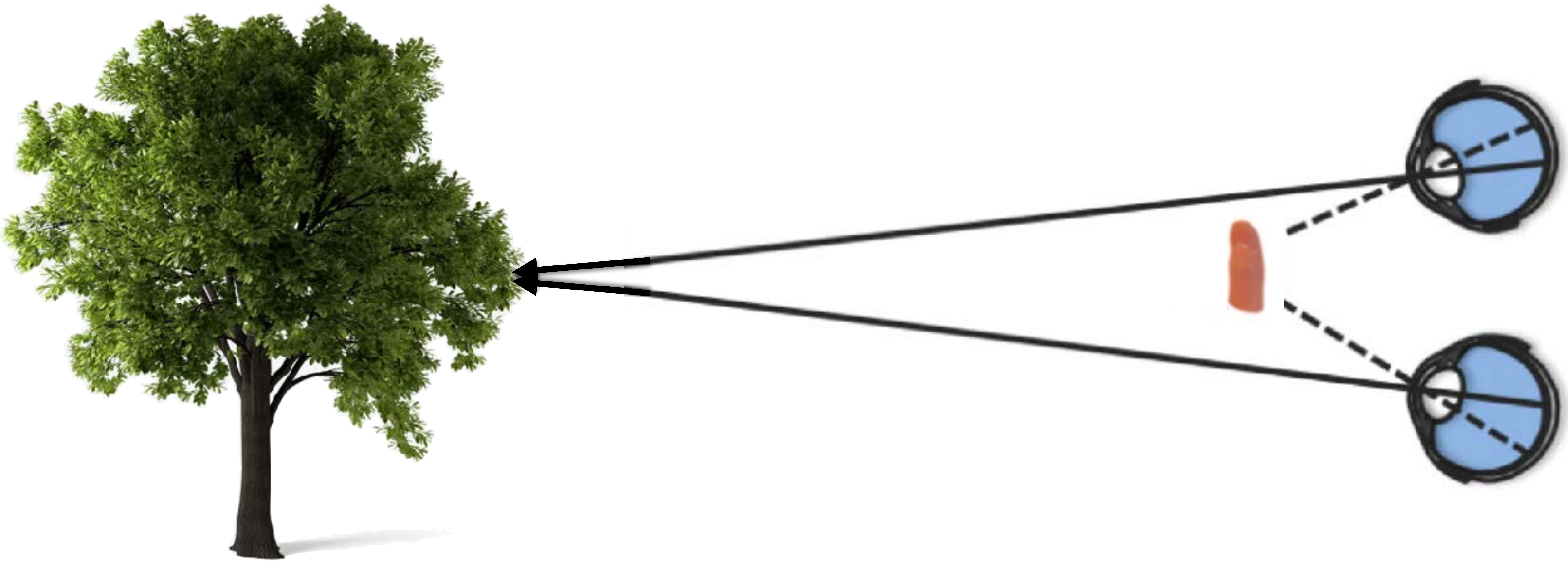
xymeow7@gmail.com

Agenda today

- *A brief overview of 3D visual computing*
- Course Overview
  - Topics covered and syllabus
  - Logistics
- Curve and Surface

We live in a 3D world

Humans have a 3D visual perception system



Many CS disciplines deal with 3D visual data

Models of 3D World

Pictures

Computer Graphics (Rendering)

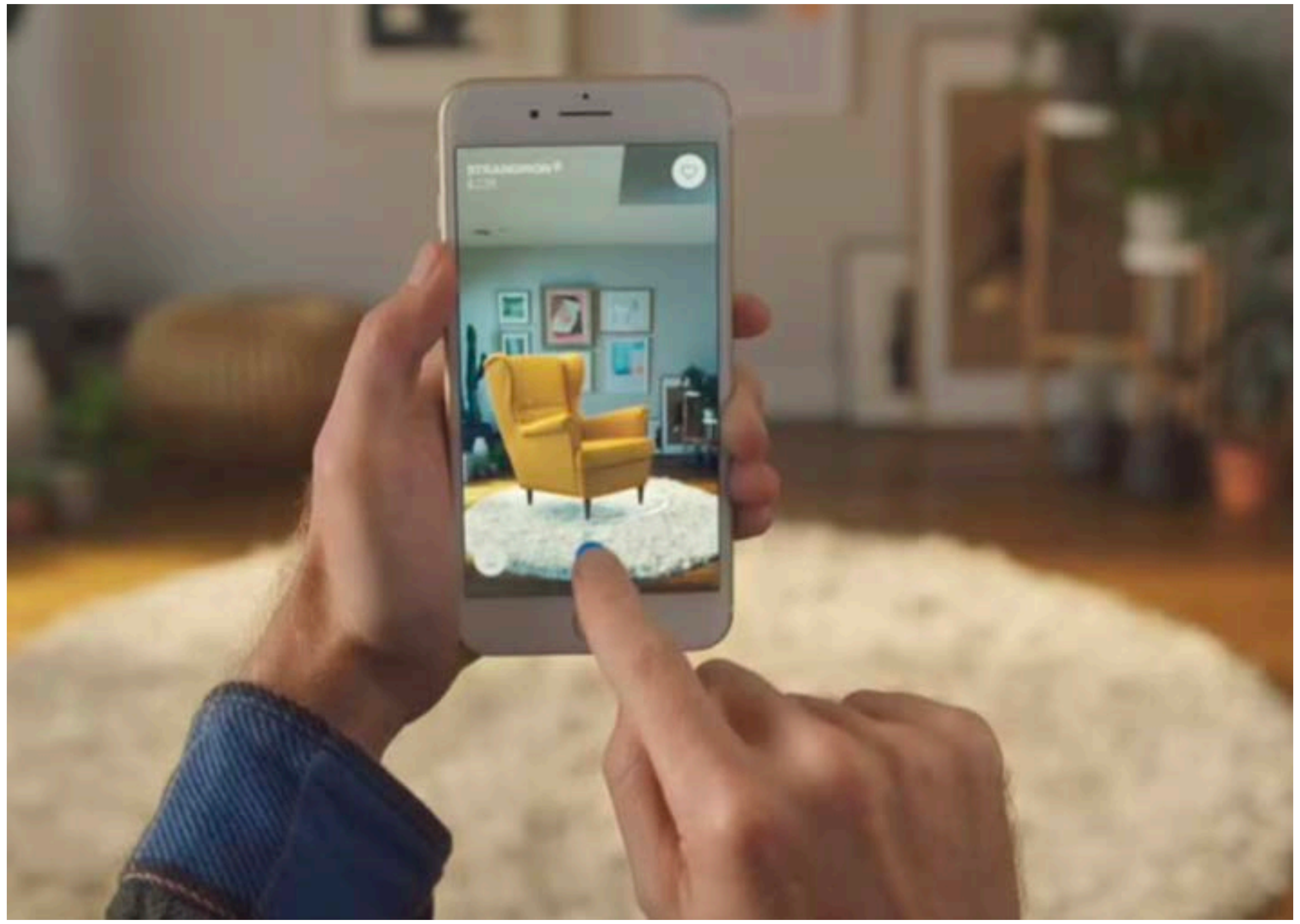
Computer Vision

Shape Processing

- Methods and applications have overlapped increasingly

3D visual computing

- 3D visual computing is a generic term for all computer science disciplines dealing with 3D visual data (3D models, 3D point clouds, etc.)
- Applications call for techniques from more than one of these fields concurrently



3D is Everywhere

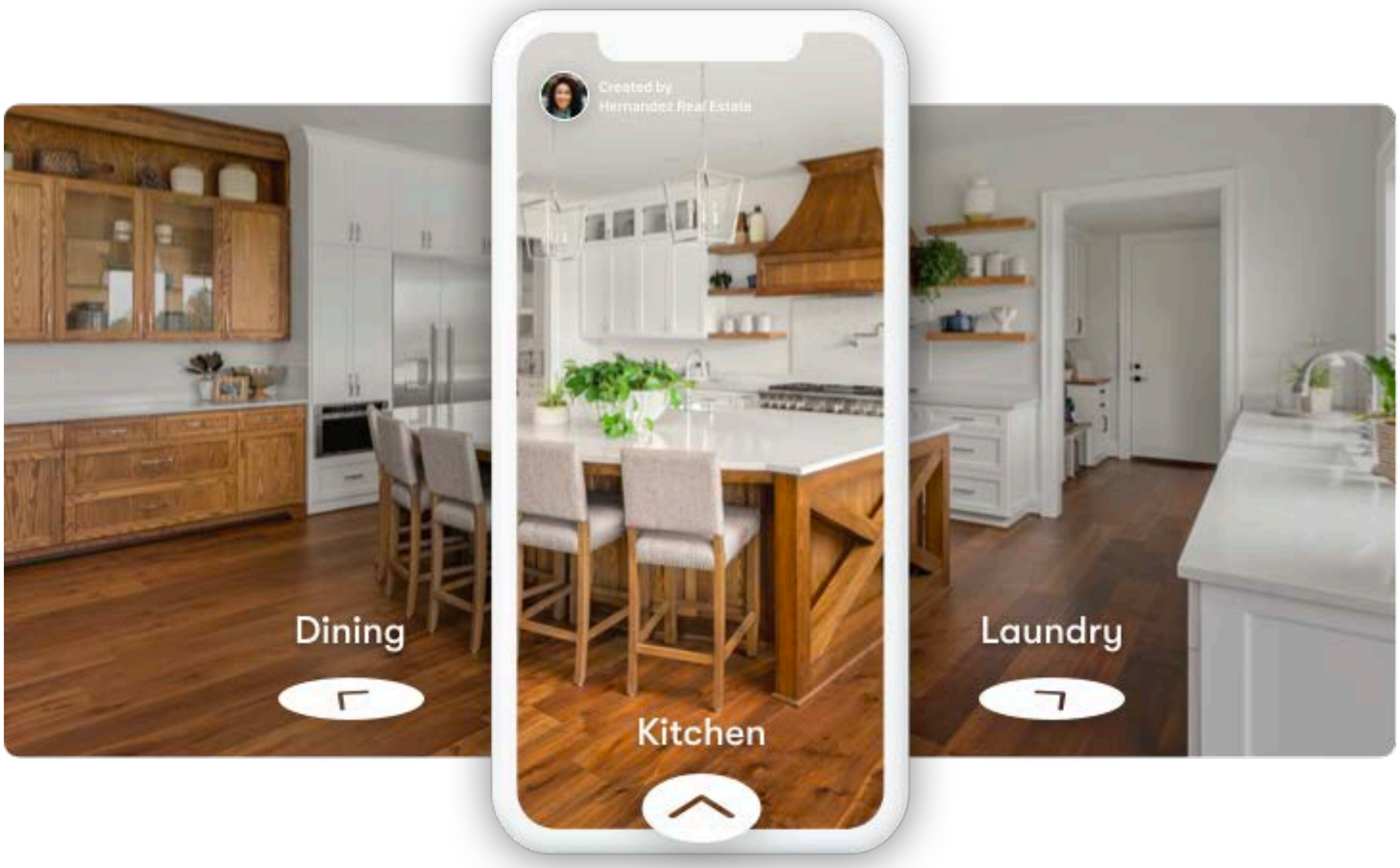


Image credits: *Zillow*

3D is Everywhere

Image credits: *Techcrunch*

3D is Everywhere

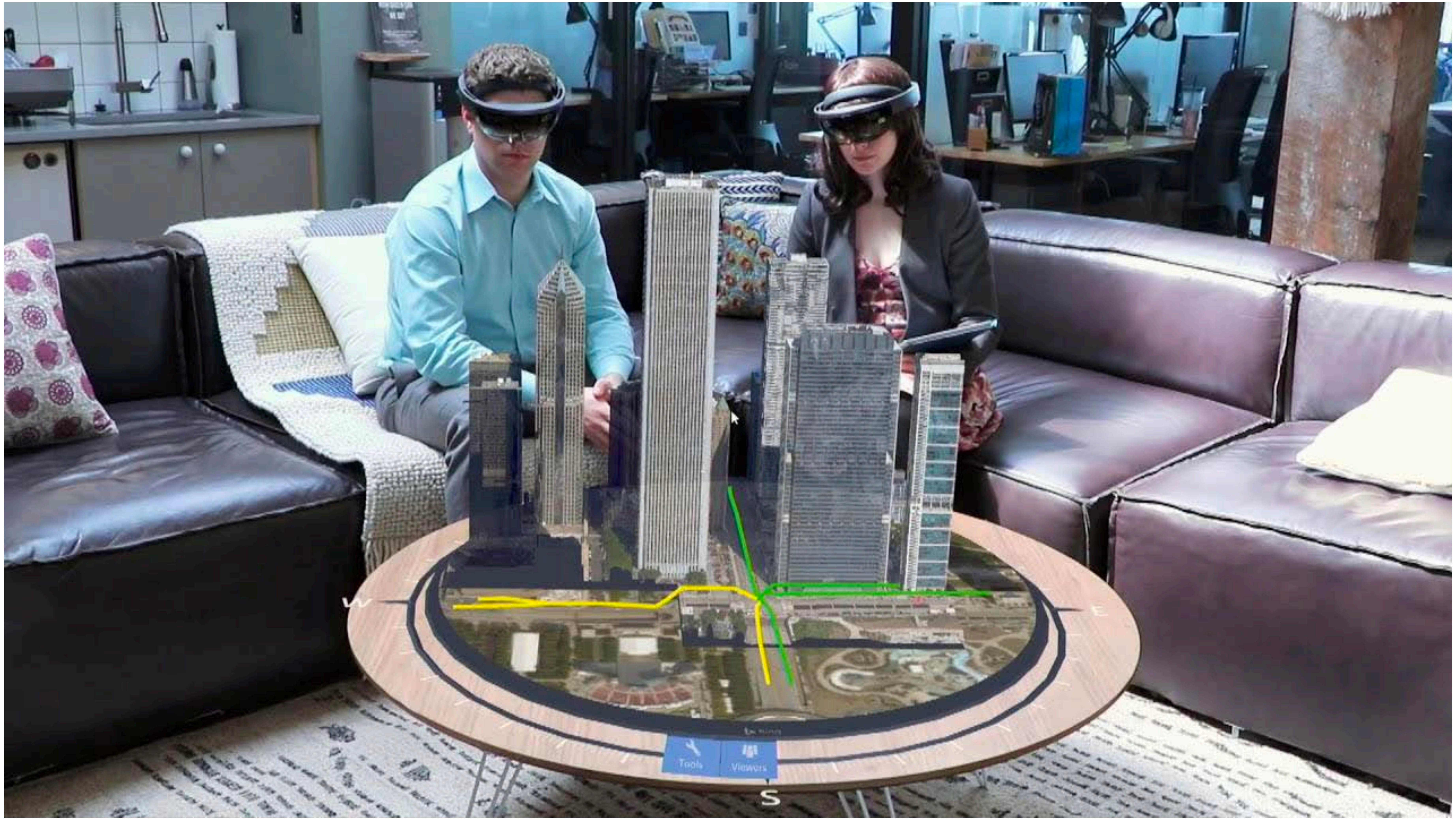


Image credits: *Taqtile*

3D is Everywhere

Image credits: *Oculus*

3D is Everywhere

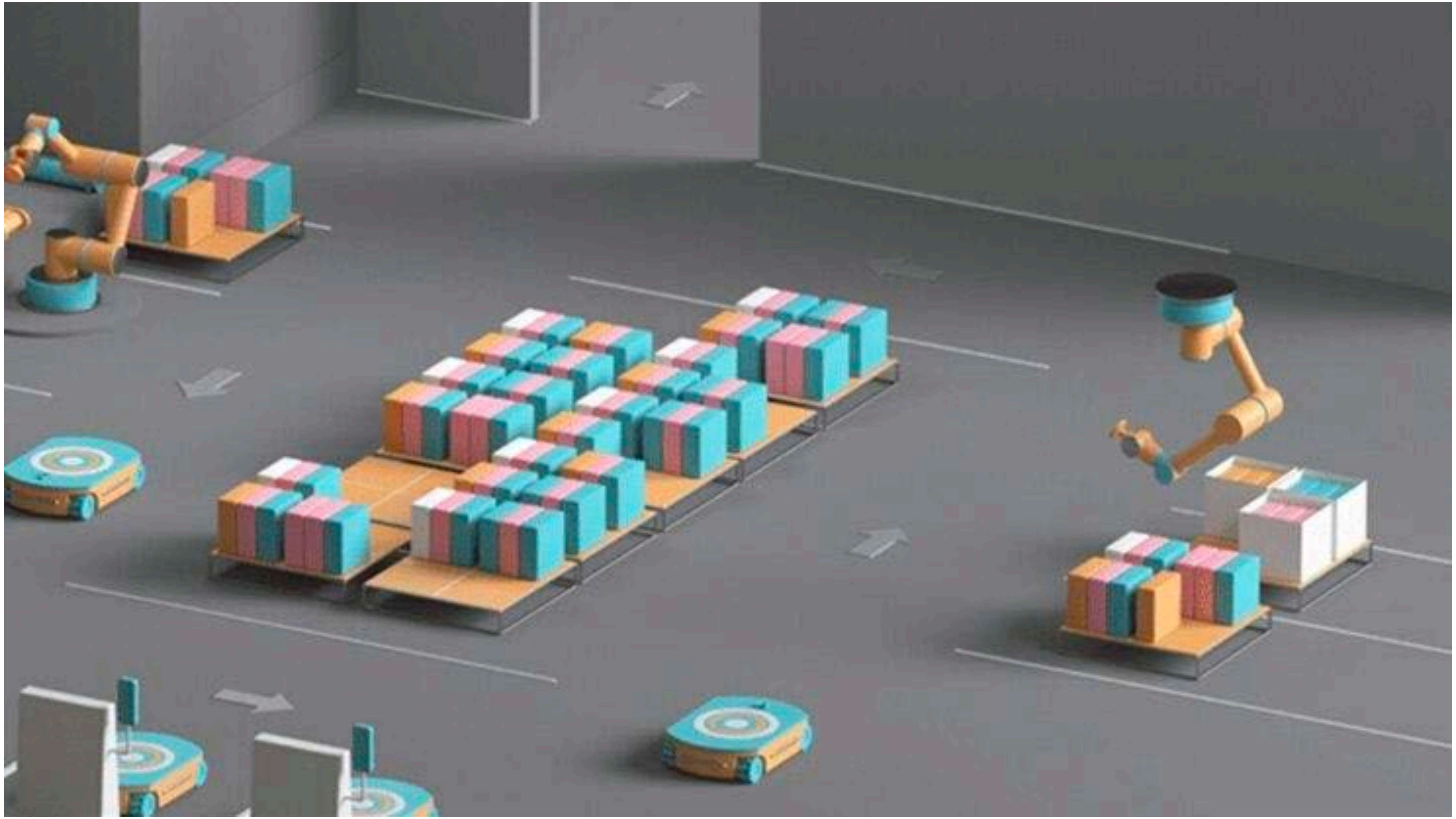
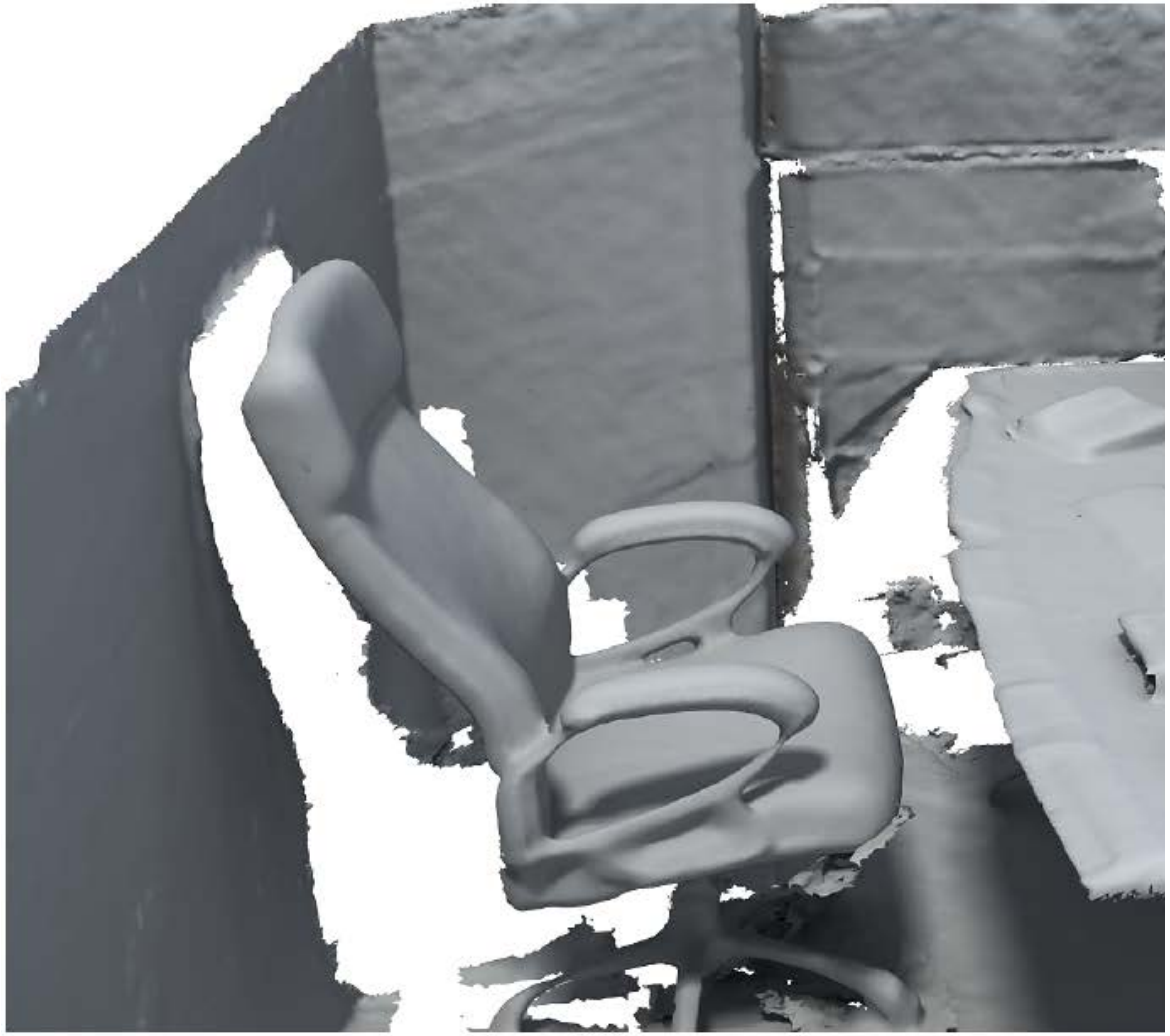
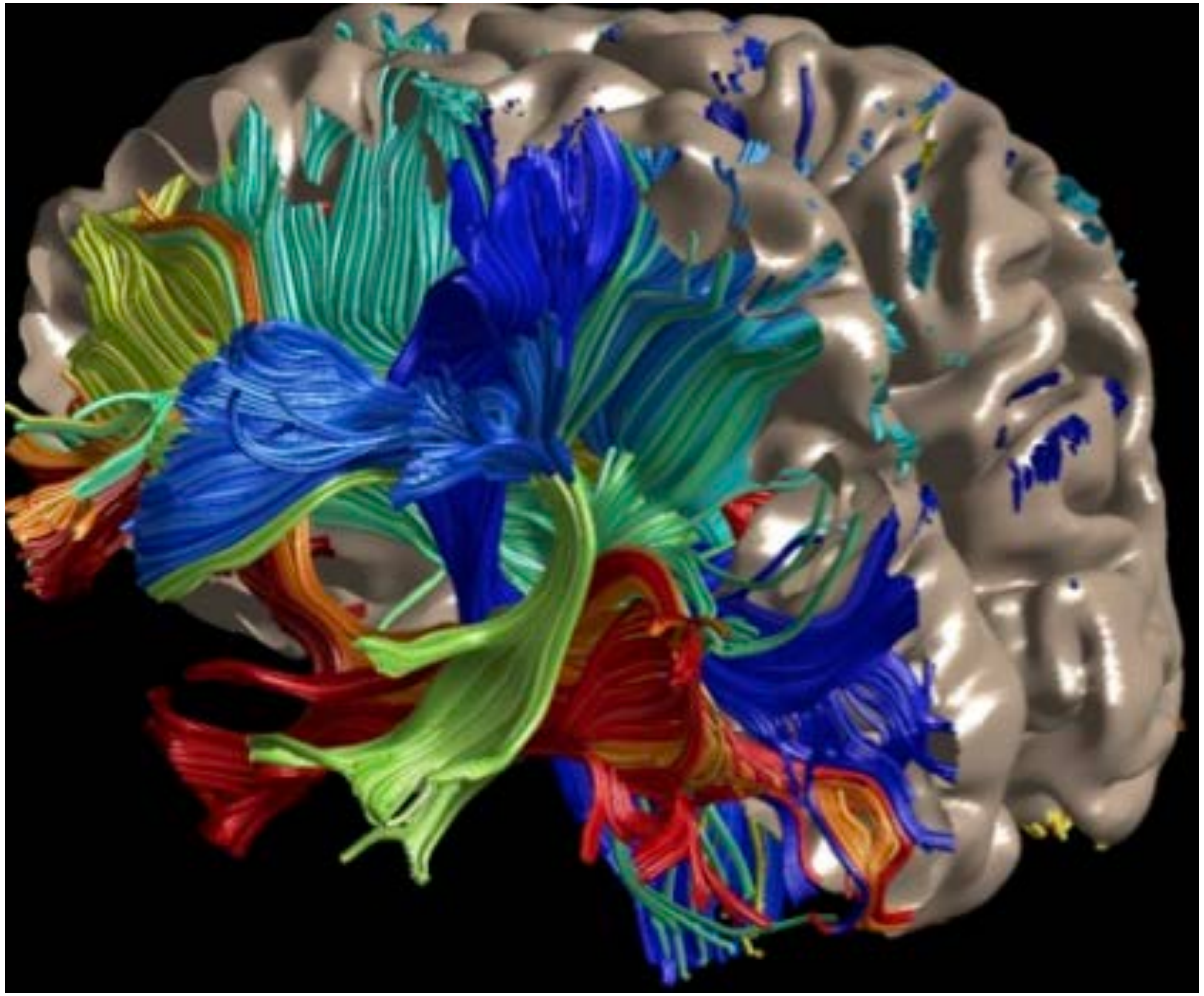


Image credits: *Unity*

3D is Everywhere



3D is Everywhere



3D is Everywhere

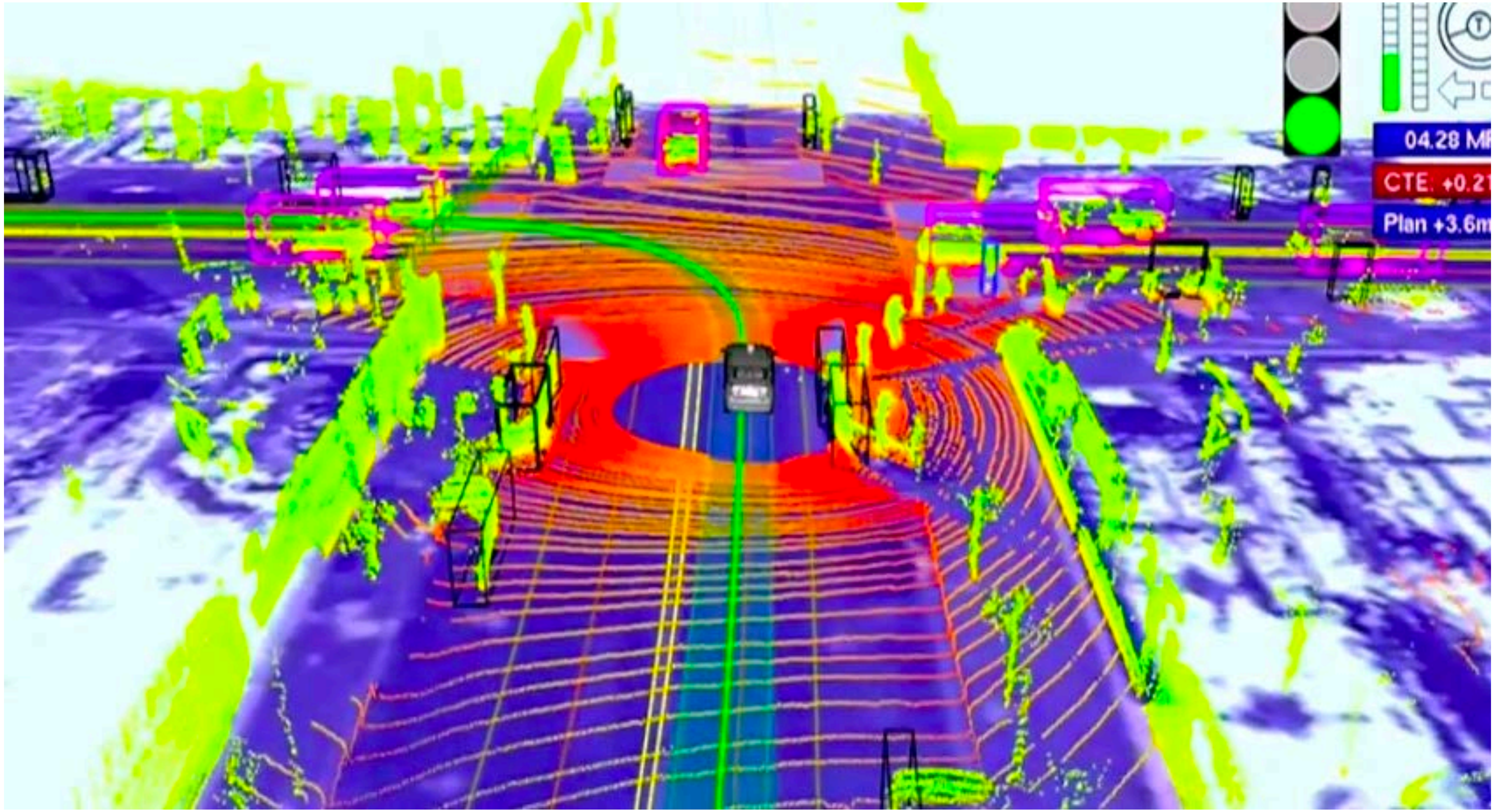
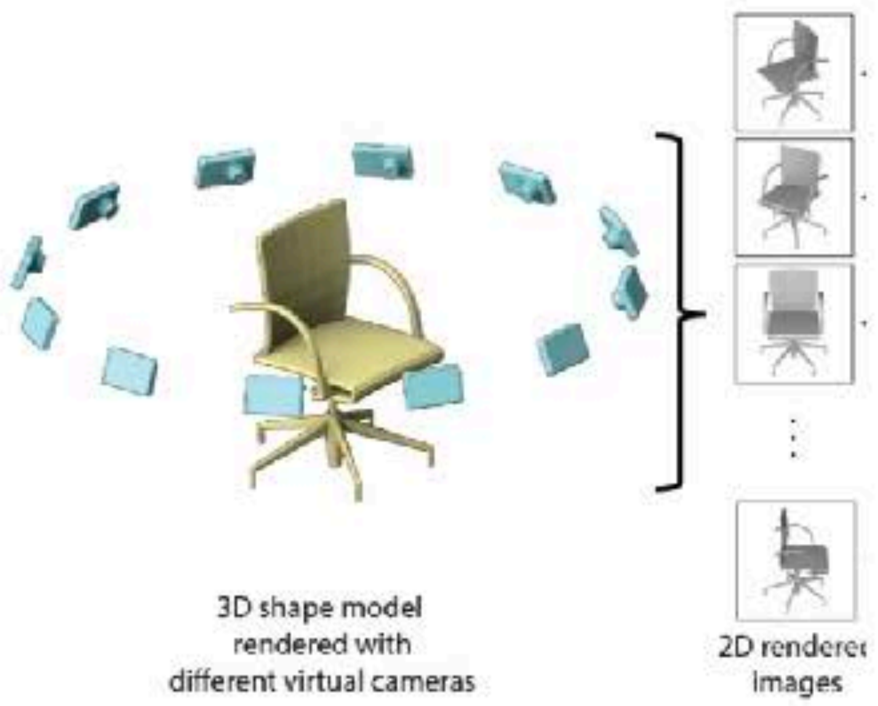


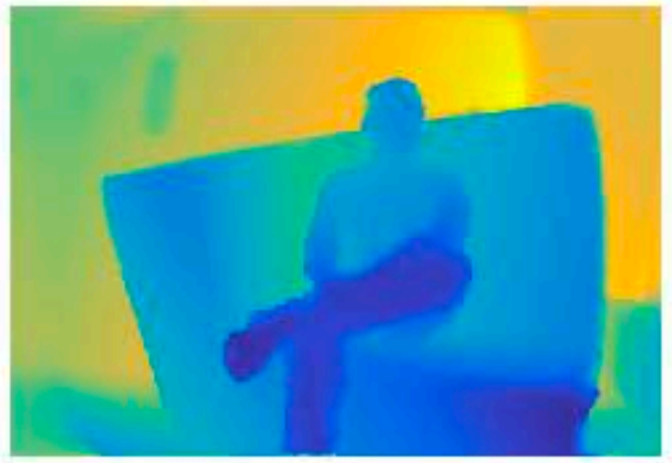
Image credits: *Waymo*

Important topics: 3D representation

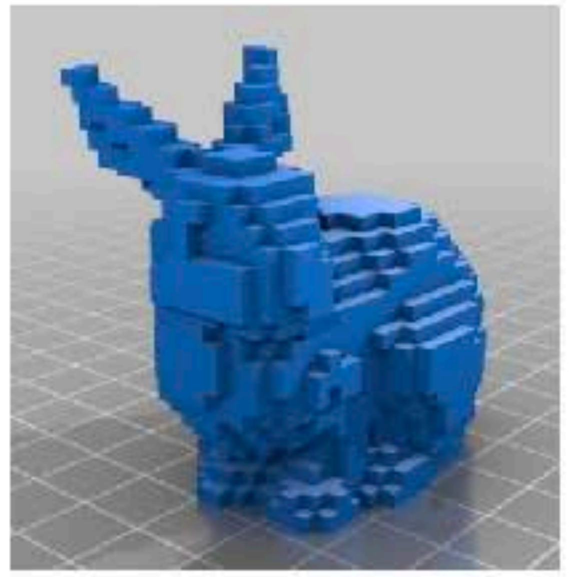
Rasterized form (regular grids)



Multi-view



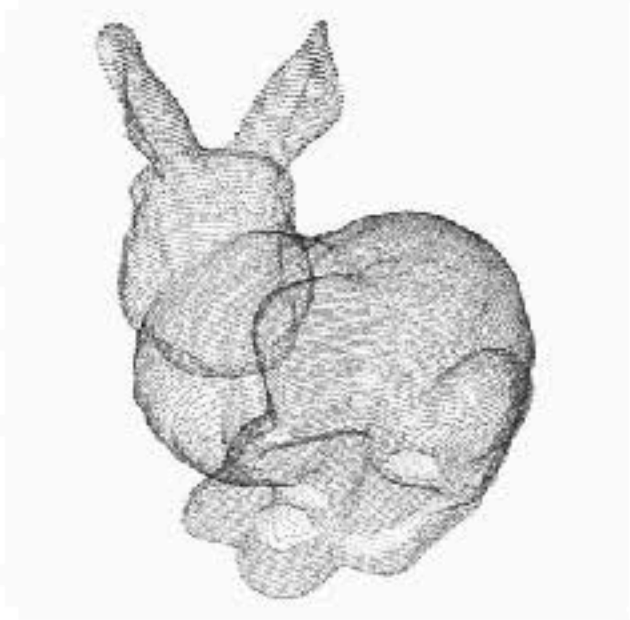
Depth Map



Volumetric

Geometric form (irregular)

Mesh



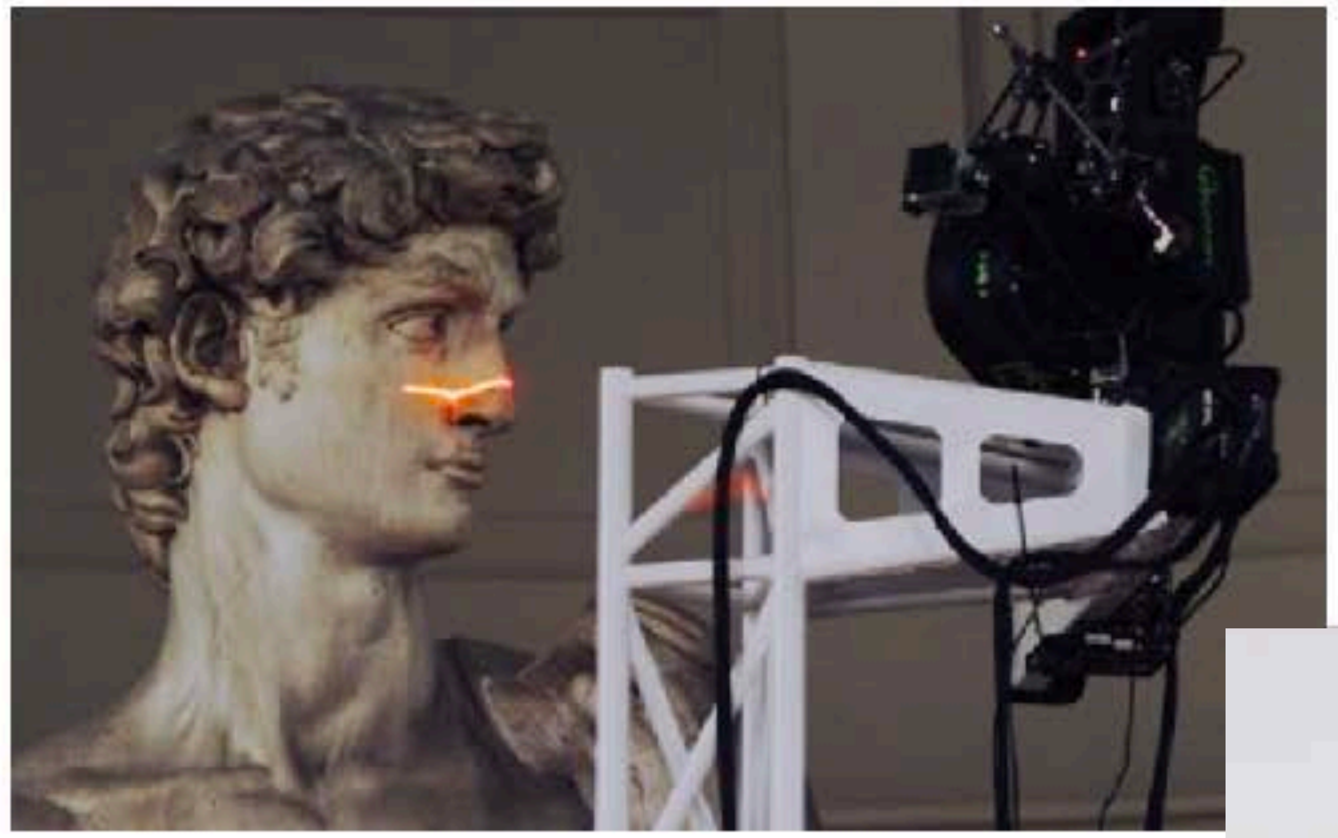
Point Cloud

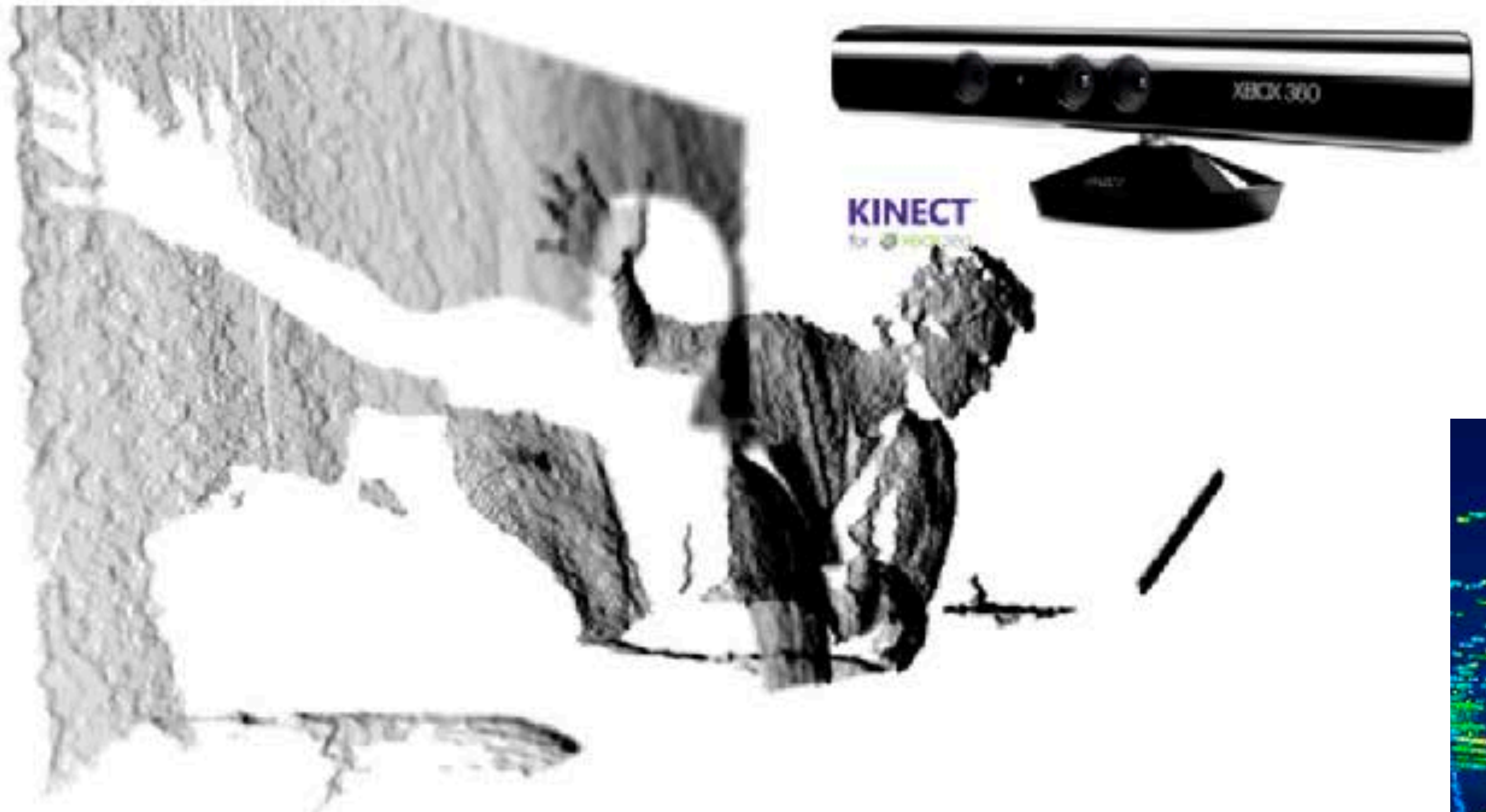
$$F(x) = 0$$

Implicit Shape

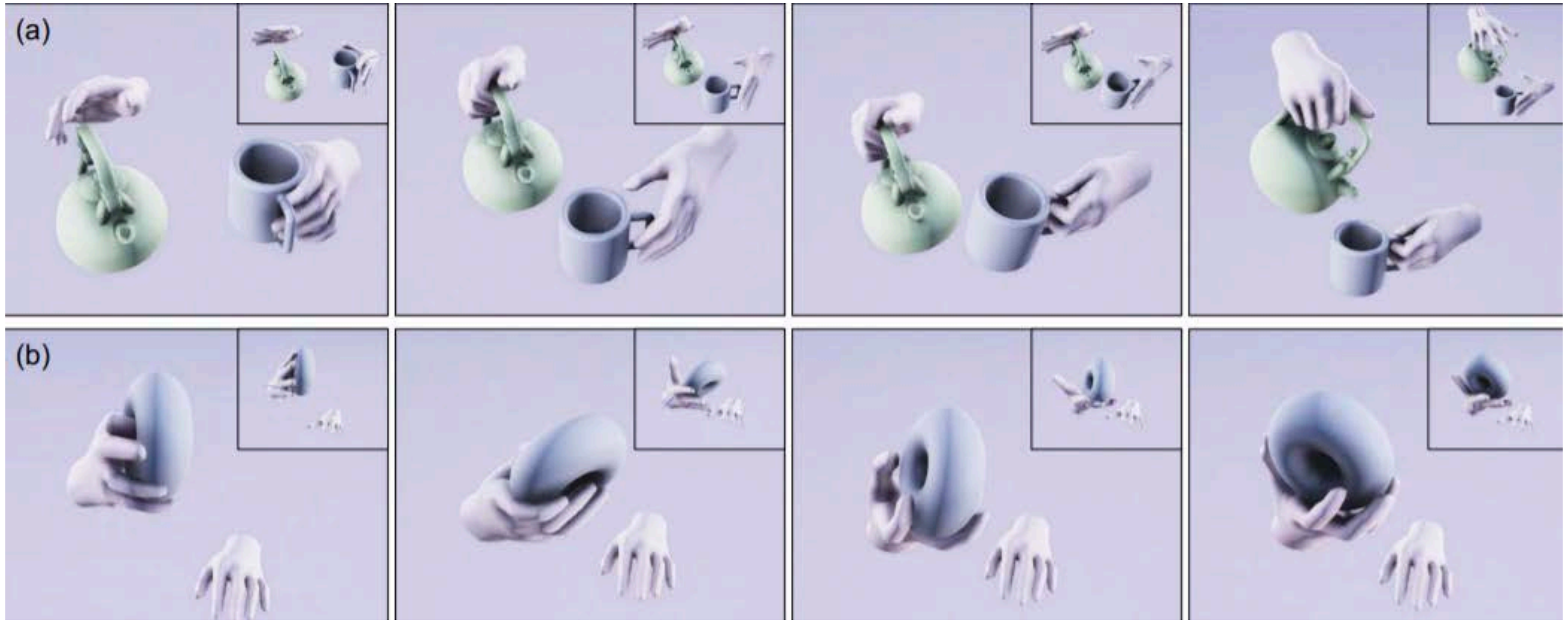
Slides credit: Hao Su

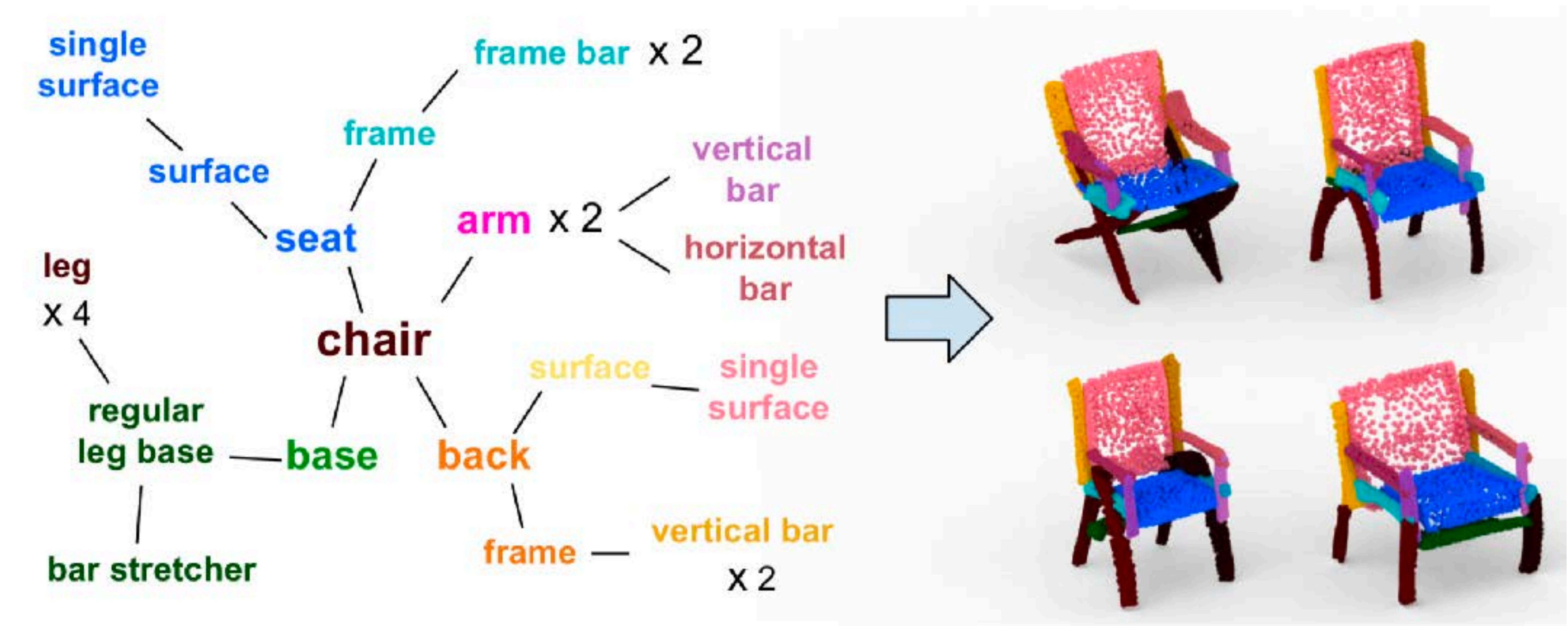
Important topics: 3D acquisition





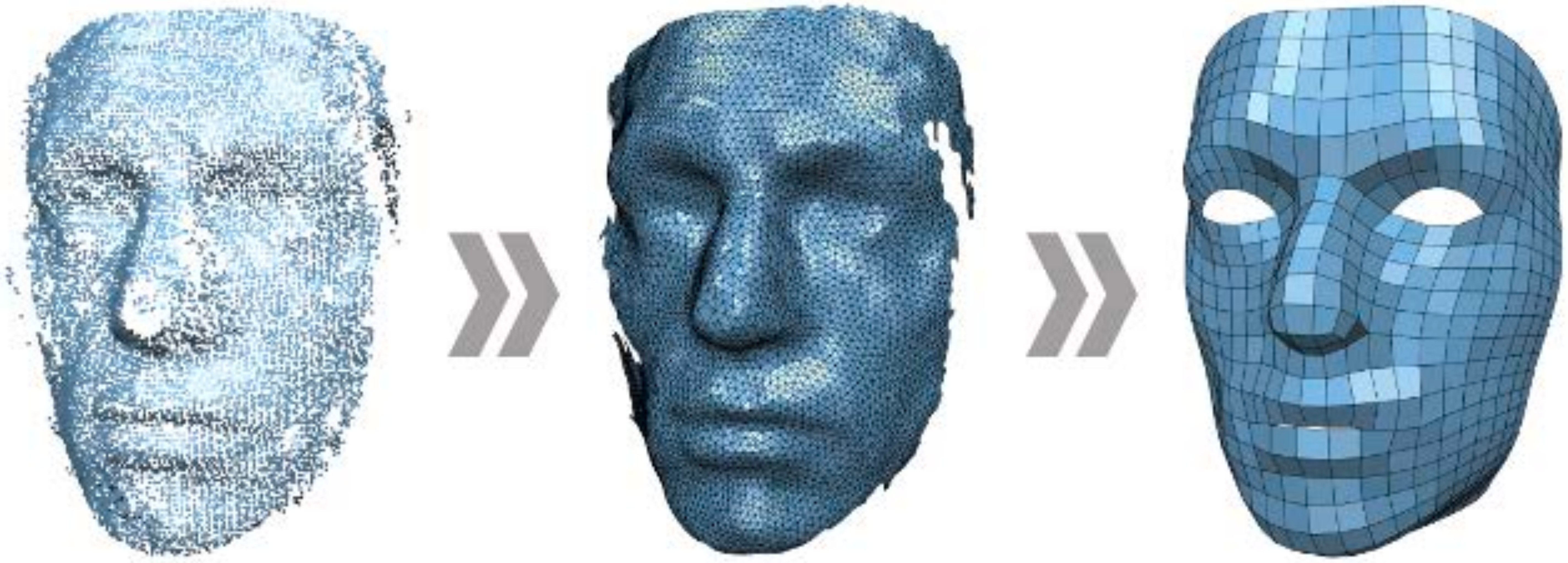
Important topics: 3D synthesis

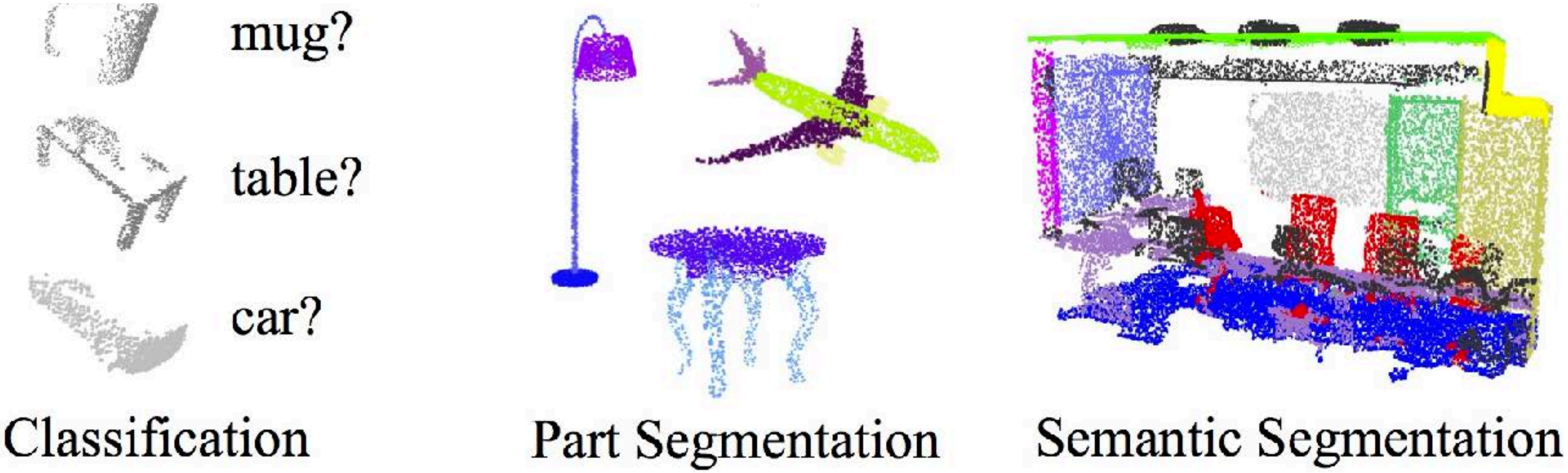




"There is a room with a table and a cake. There is a red chair to the right of the table."

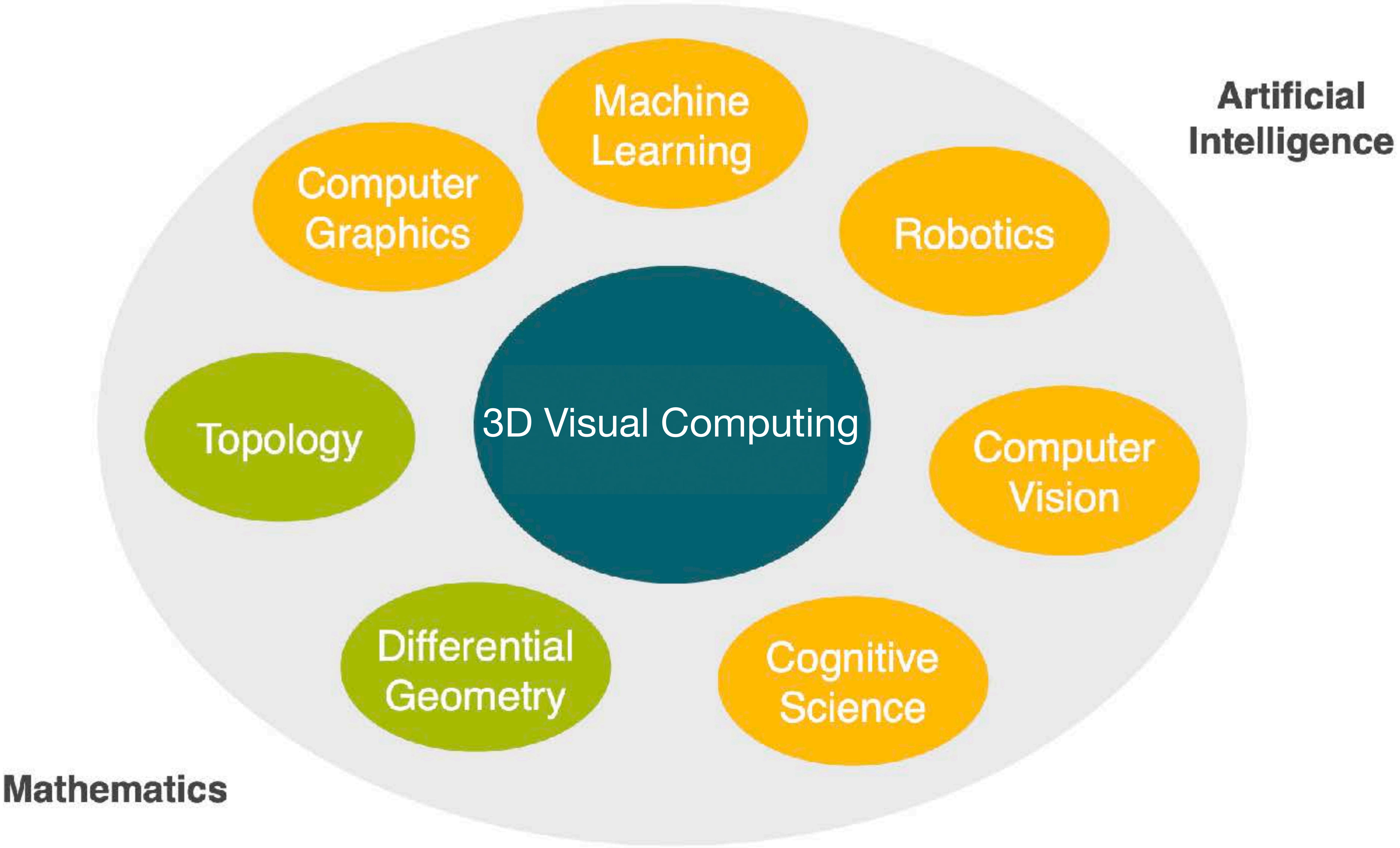
Important topics: 3D processing and analysis





Important topics: Rendering, simulation and animation

Highly interdisciplinary field



A big trend in the community

1	3D from multi-view and sensors	1,090	246
2	Image and video synthesis and generation	889	185
3	Humans: Face, body, pose, gesture, movement	813	166
4	Transfer, meta, low-shot, continual, or long-tail learning	688	153
5	Recognition: Categorization, detection, retrieval	673	139
6	Vision, language, and reasoning	631	118
7	Low-level vision	553	126
8	Segmentation, grouping and shape analysis	524	113
9	Deep learning architectures and techniques	485	92
10	Multi-modal learning	450	89
11	3D from single images	431	91
12	Medical and biological vision, cell microscopy	420	53
13	Video: Action and event understanding	373	83
14	Autonomous driving	359	69
15	Self-supervised or unsupervised representation learning	349	71
16	Datasets and evaluation	344	54
17	Scene analysis and understanding	276	54
18	Adversarial attack and defense	274	61
19	Efficient and scalable vision	252	48
20	Computational imaging	226	53
21	Video: Low-level analysis, motion, and tracking	215	46
22	Vision applications and systems	171	35
23	Vision + graphics	155	32
24	Robotics	141	23
25	Transparency, fairness, accountability, privacy, ethics in vision	129	30
26	Explainable computer vision	107	24
27	Embodied vision: Active agents, simulation	80	14
28	Document analysis and understanding	72	12
29	Machine learning (other than deep learning)	65	14
30	Physics-based vision and shape-from-X	55	12
31	Biometrics	51	11
32	Others	47	12
33	Optimization methods (other than deep learning)	46	12
34	Photogrammetry and remote sensing	38	8
35	Computer vision theory	33	5
36	Computer vision for social good	25	5

CVPR 2024

Computer Vision and Pattern Recognition Conference
Seattle | June 17-21, 2024

Select
(全部)

Hover or
research
in top lef

Paper Topics and Totals

	Paper Topics and Totals		Oral (top 5% papers)	Highlight (top 10% papers)
1	Image and video synthesis and generation	329	12	50
2	3D from multi-view and sensors	276	10	45
3	Humans: Face, body, pose, gesture, movement	202	7	28
4	Vision, language, and reasoning	152	7	23
5	Low-level vision	131	6	18
6	Recognition: Categorization, detection, retrieval	127	6	15
7	Transfer, meta, low-shot, continual, or long-tail learning	123	5	14
8	Multi-modal learning	110	5	11
9	Segmentation, grouping and shape analysis	107	5	10
10	3D from single images	106	3	9
11	Datasets and evaluation	95	3	9
12	Autonomous driving	87	3	9
13	Video: Action and event understanding	78	2	8
14	Deep learning architectures and techniques	69	2	8
15	Medical and biological vision, cell microscopy	66	2	7
16	Adversarial attack and defense	59	2	6
17	Scene analysis and understanding	56	2	5
18	Vision + graphics	56	1	5
19	Computational imaging	53	1	4
20	Efficient and scalable vision	51	1	4
21	Self-supervised or unsupervised representation learning	49	1	4
22	Transparency, fairness, accountability, privacy, ethics	49	1	4
23	Vision applications and systems	44	1	4
24	Video: Low-level analysis, motion, and tracking	38	1	3
25	Robotics	29	1	3
26	Embodied vision: Active agents, simulation	27		3
27	Explainable computer vision	23		3
28	Photogrammetry and remote sensing	19		3
29	Physics-based vision and shape-from-X	17		2
30	Machine learning (other than deep learning)	16		2
31	Biometrics	15		2
32	Document analysis and understanding	14		1
33	Others	14		1
34	Computer vision for social good	12		1
35	Computer vision theory	11		
36	Optimization methods (other than deep learning)	6		

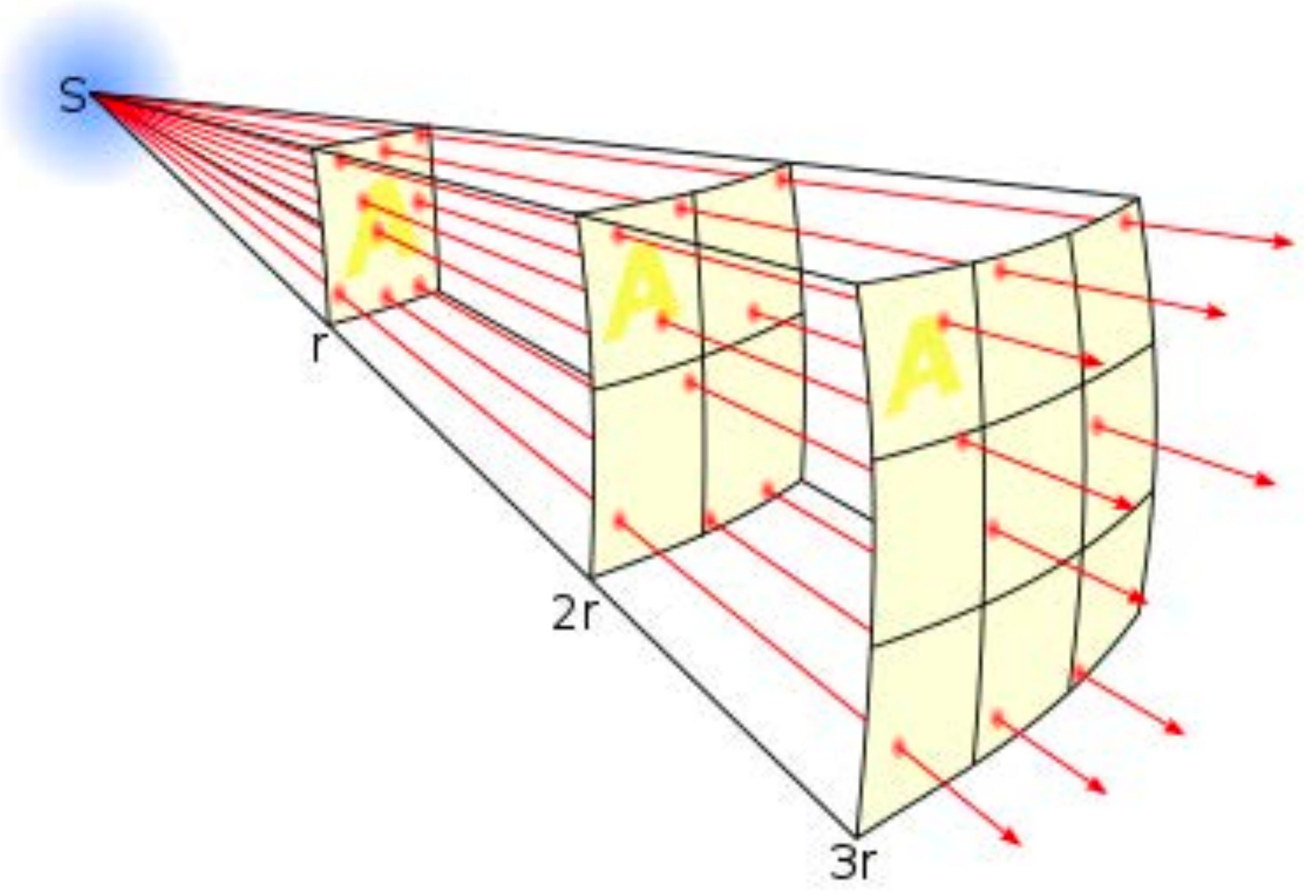
This Course Focuses on 3D Visual Computing in the Deep Learning Era

Why the focus on **3D?**

String Theorist to a Mathematician: *How do you imagine a 17-d space?*

Mathematician: *It's easy. I imagine a n-dimensional space, and set n=17.*

Why not (just) develop general techniques instead?



Inverse square law is specific to 3D

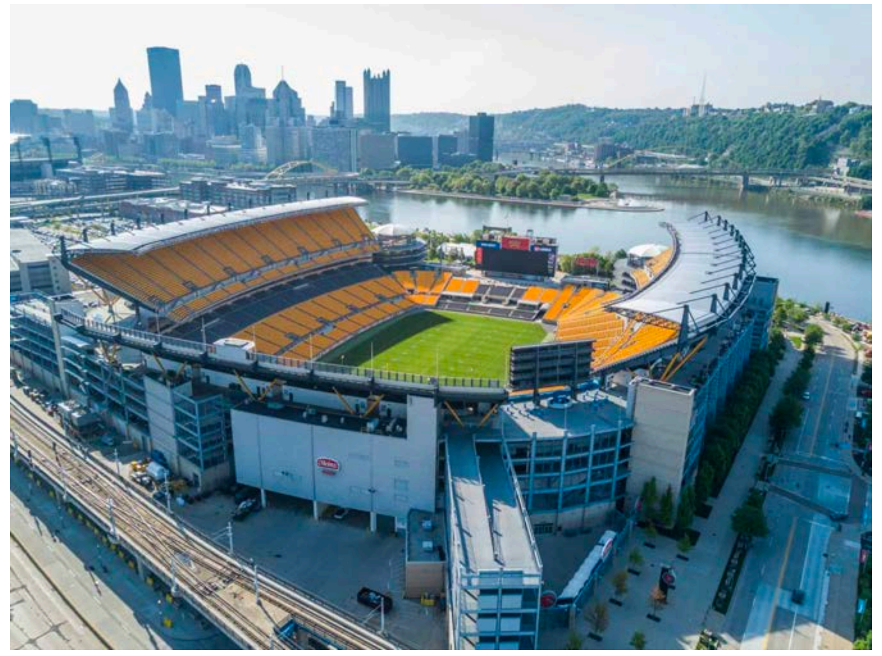
(no stable planetary orbits in high-dimensional worlds!)

But our world ***is**** 3D

Specialized techniques/ representations may work better

This Course Focuses on 3D Visual Computing in the Deep Learning Era

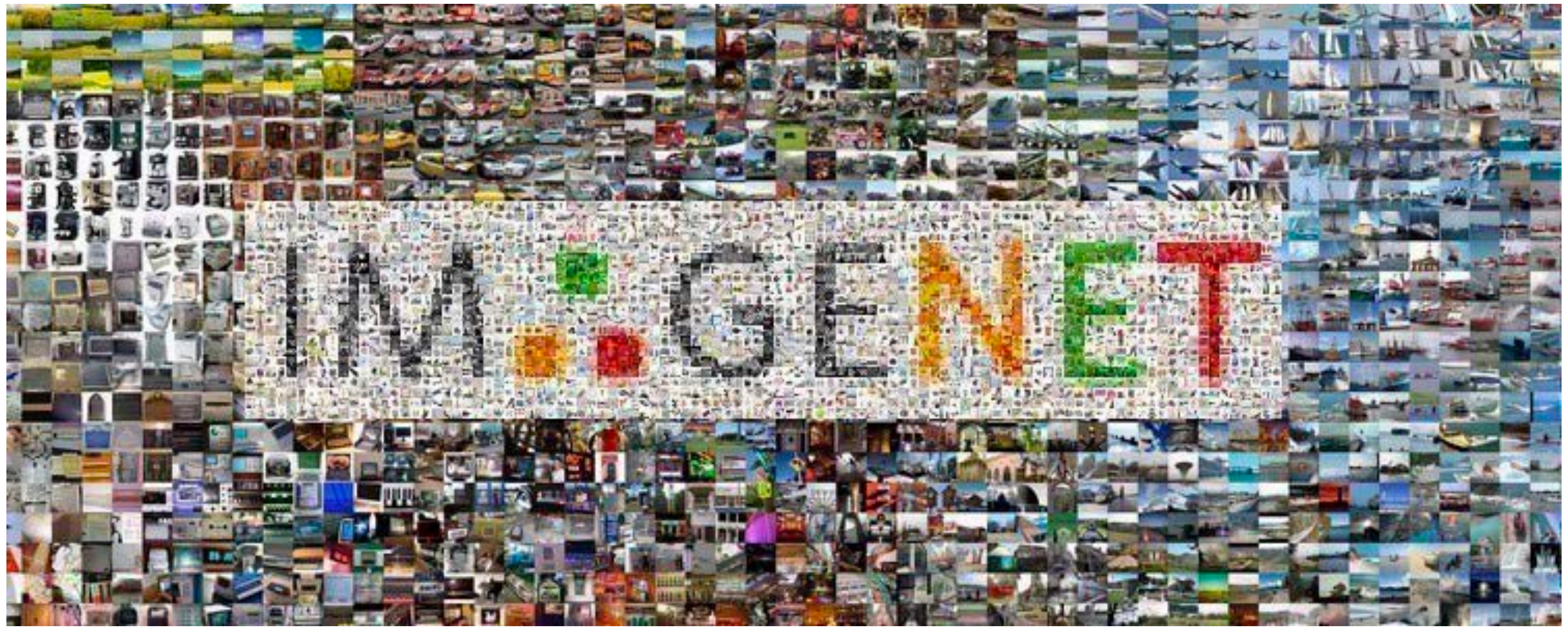
Why the focus on **Learning?**



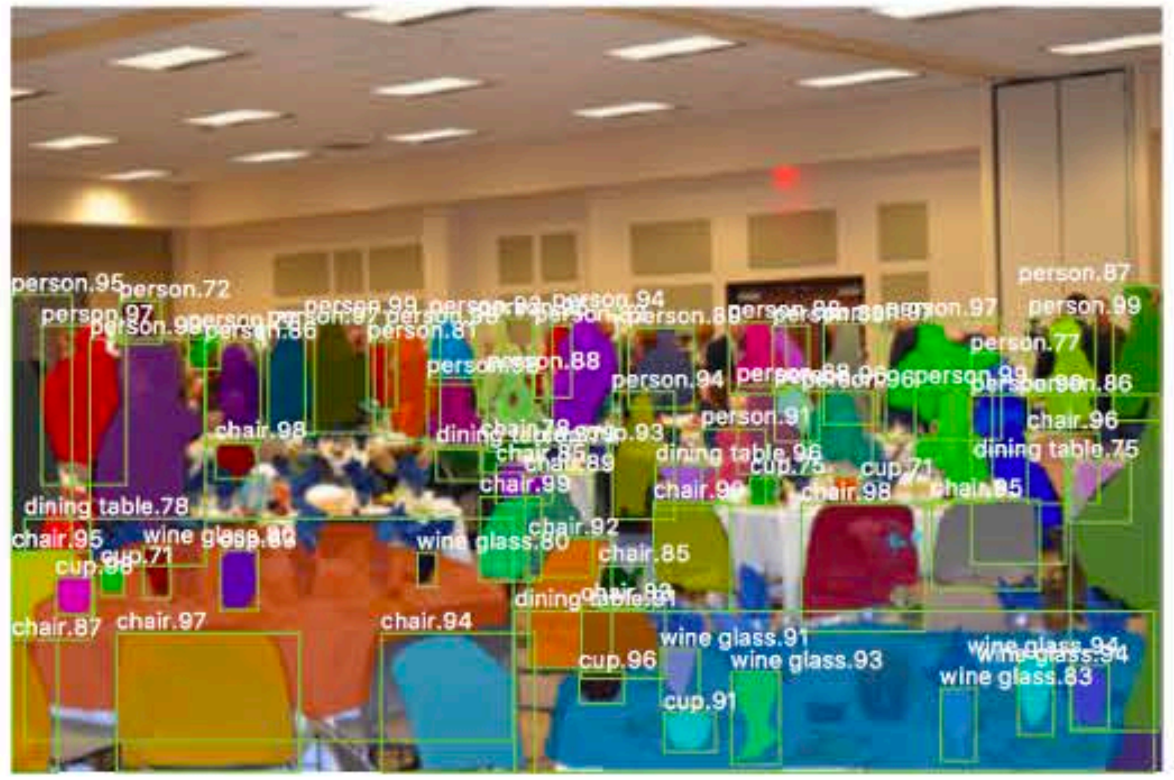
?

Purely geometric methods cannot
leverage past experiences

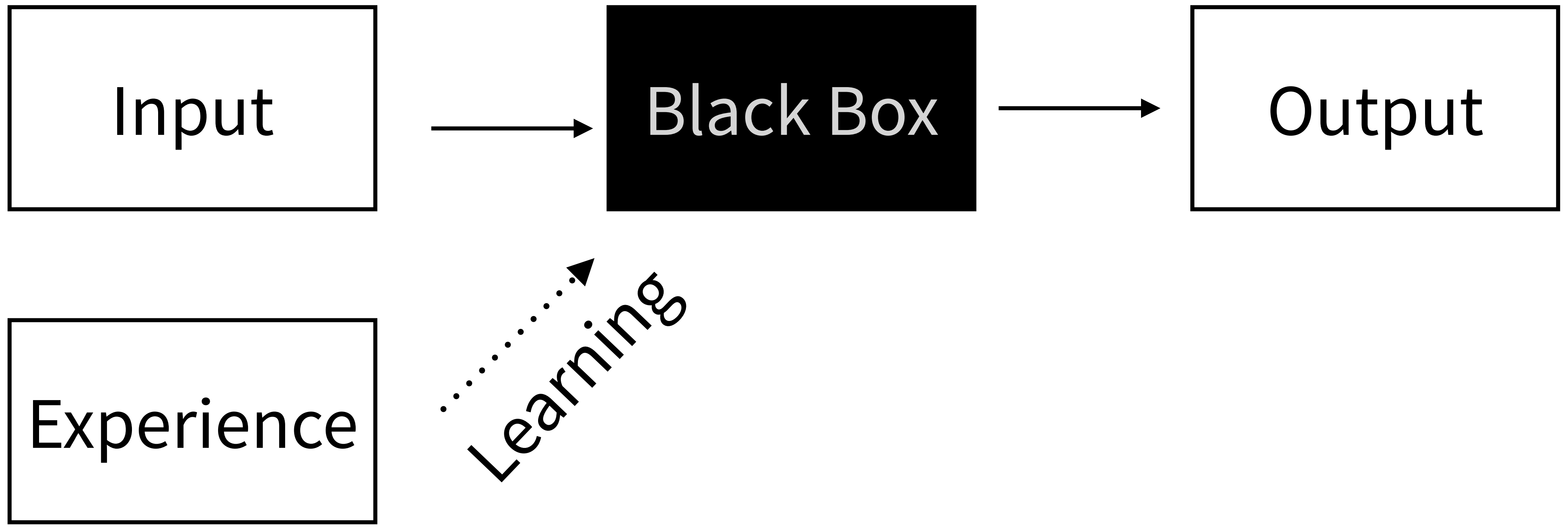
This Course Focuses on 3D Visual Computing in the Deep Learning Era



COCO
Common Objects in Context

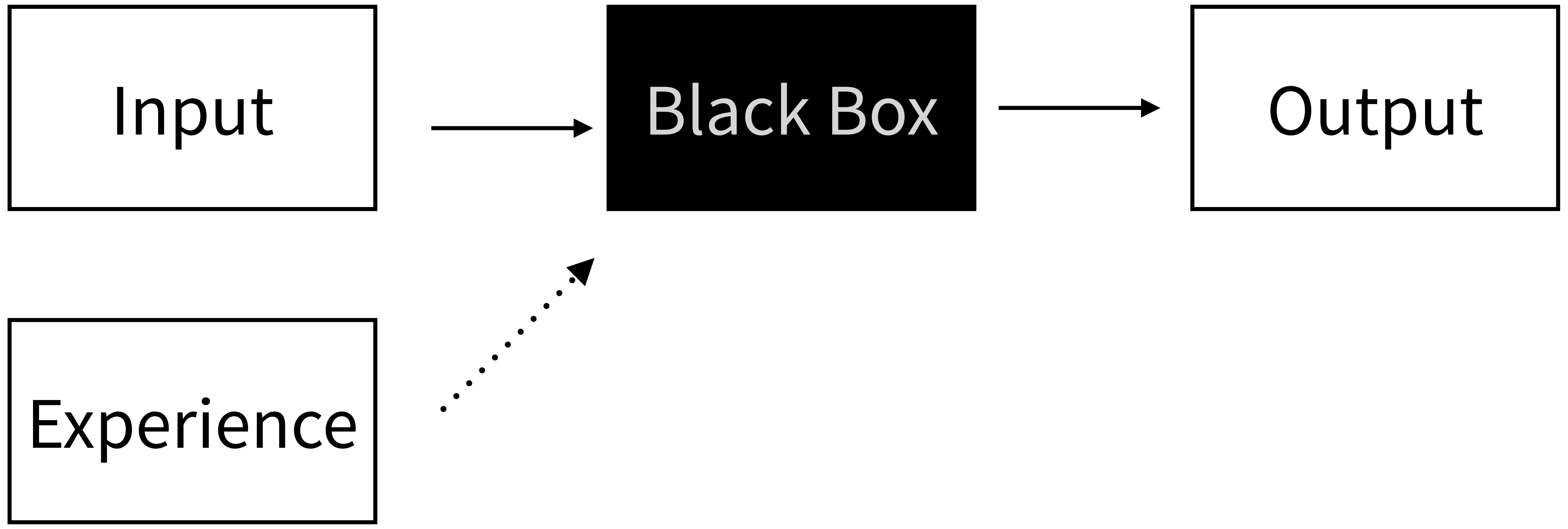


Learning-Based 3D Visual Computing



Allows data-driven inference

Learning-Based 3D Visual Computing



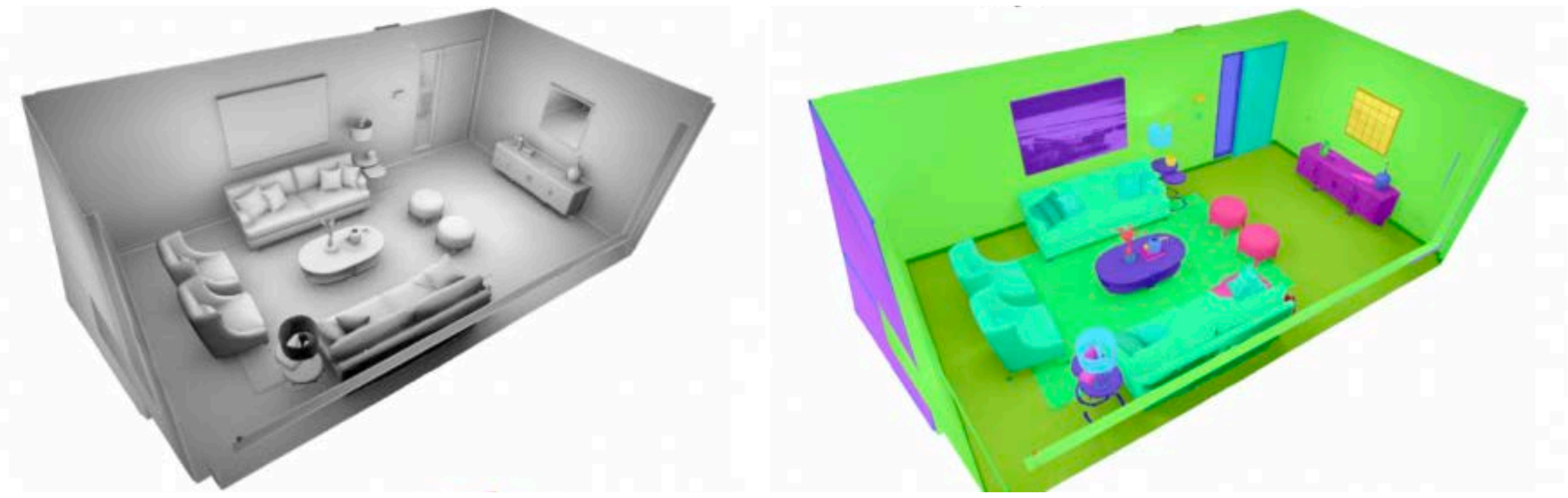
Allows data-driven inference

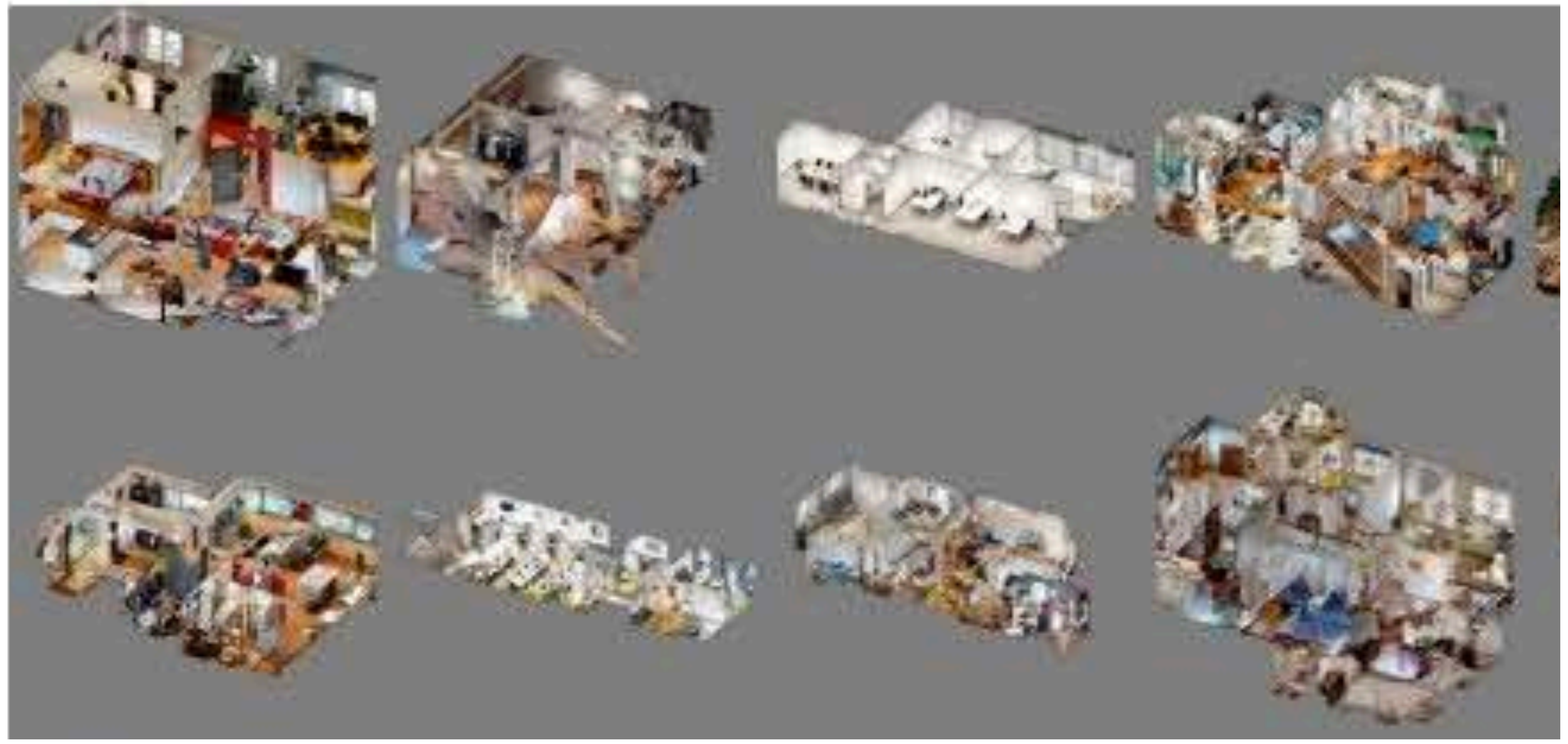
Image credits: Birds of a Feather Capturing Avian Shape Models from Images. *Wang et. al*

Learning-Based 3D Visual Computing

Input → Black Box → Output

Experience ⇢ Black Box

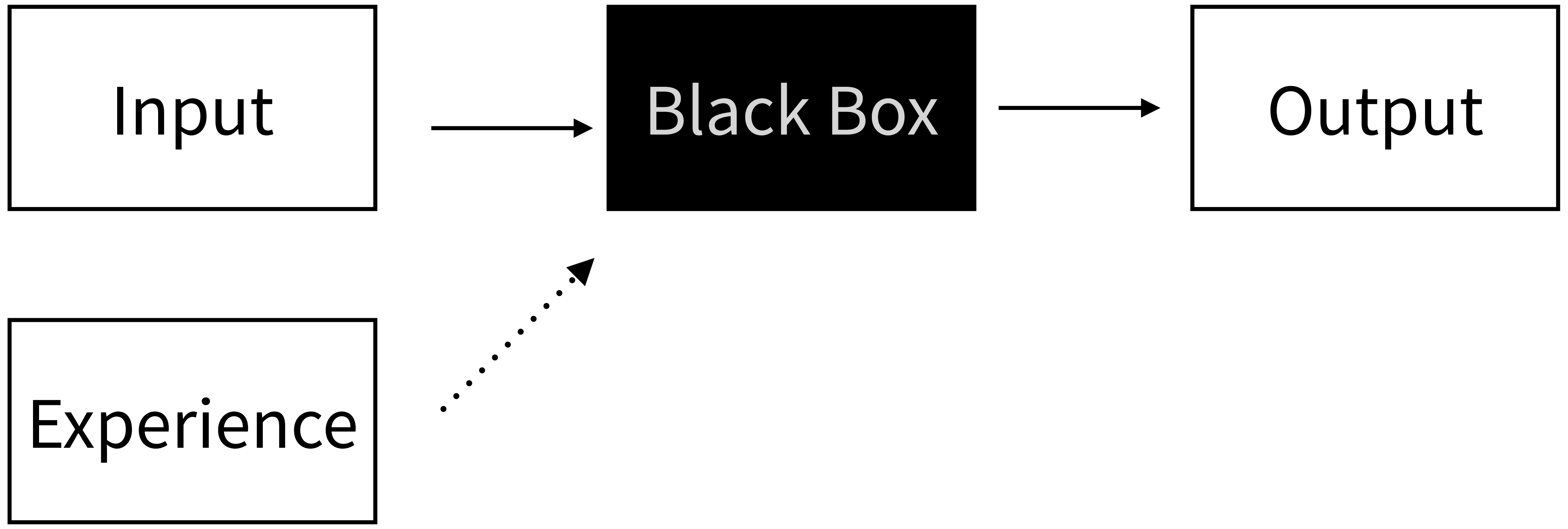




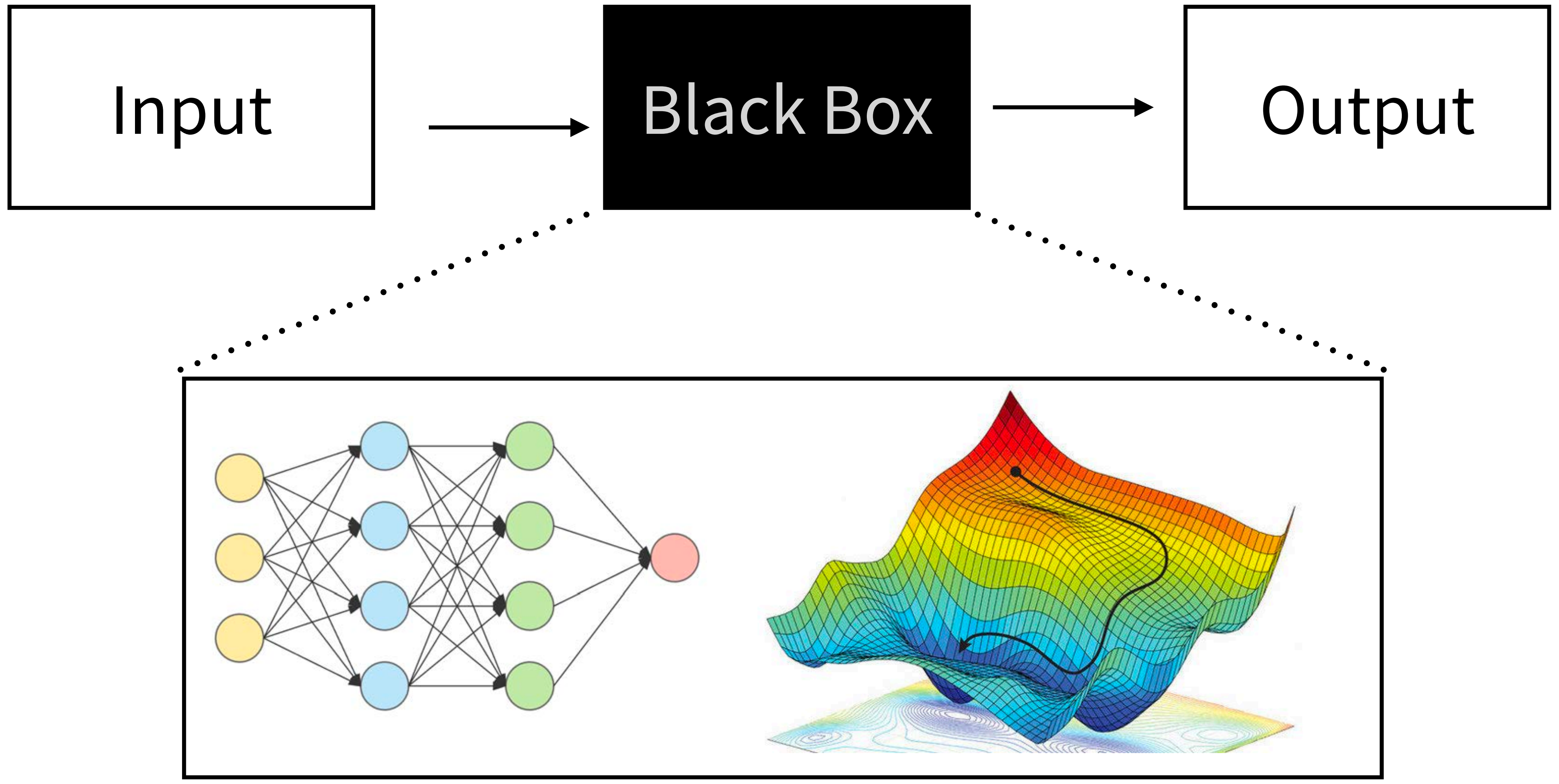
Allows data-driven inference

Image credits: Habitat-Matterport 3D Dataset. *Ramakrishnan et. al*

Learning-Based 3D Visual Computing

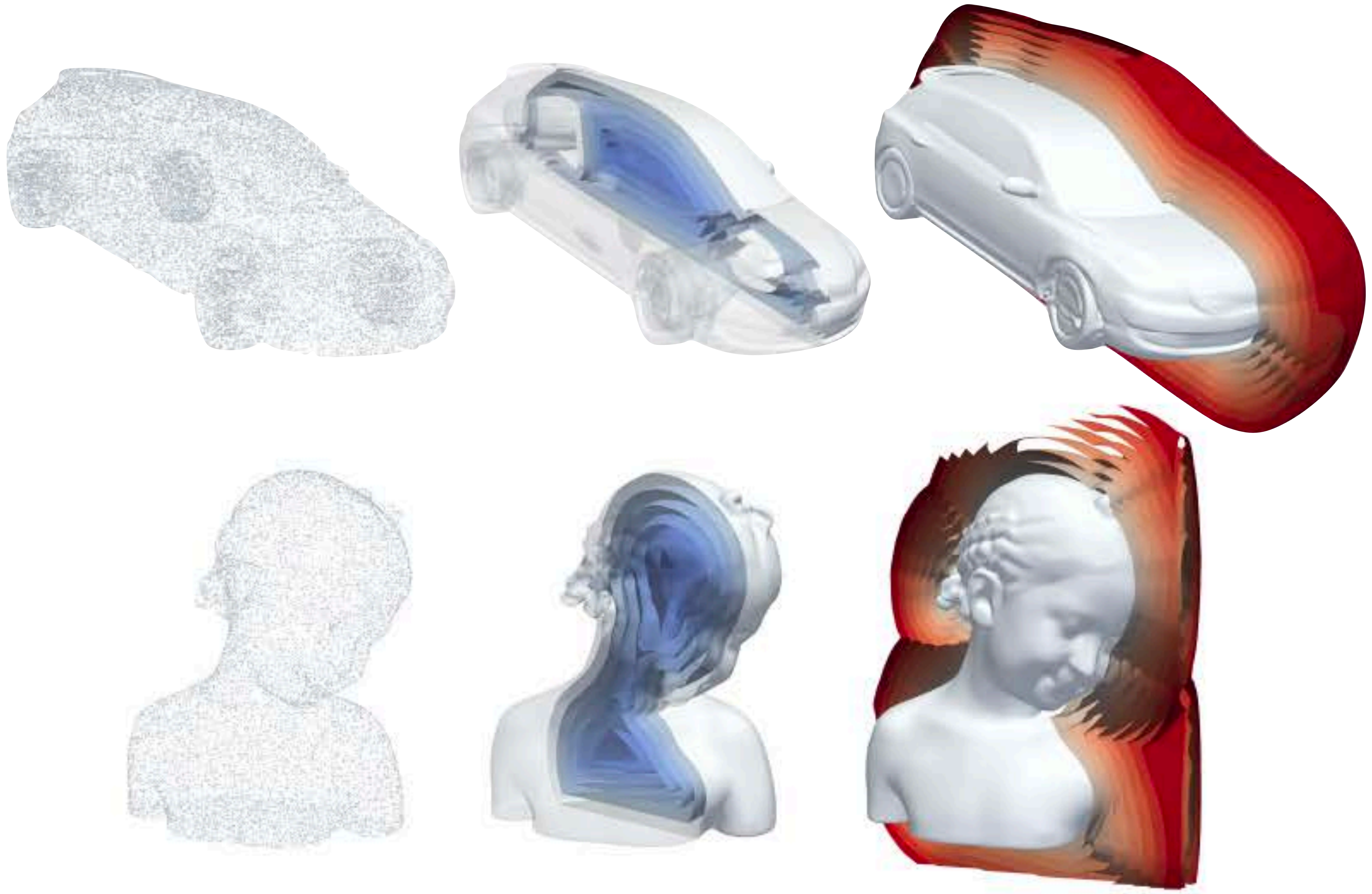


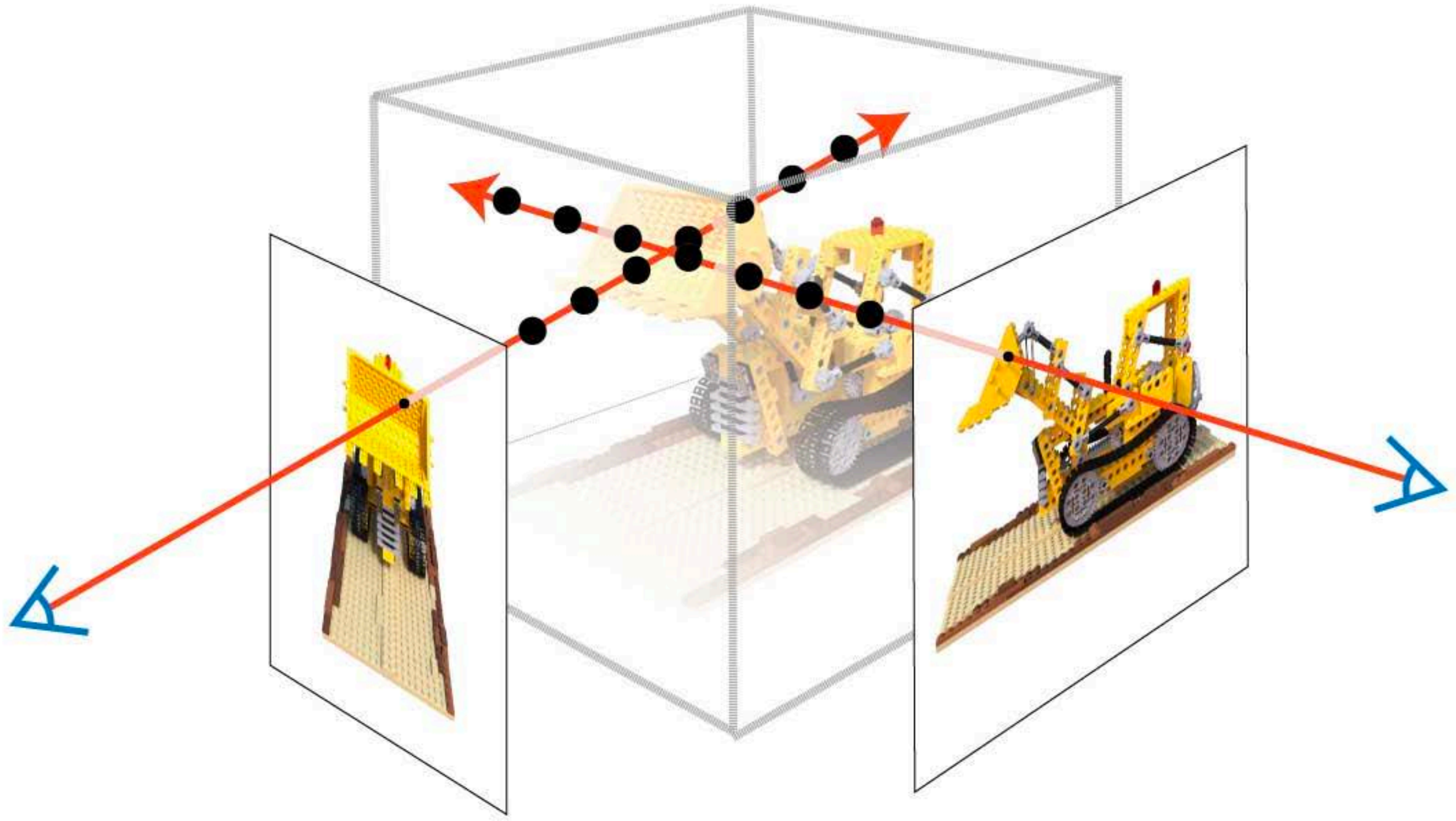
Allows data-driven inference



Interesting representation properties and easier optimization landscape (sometimes!)

Learning-Based 3D Visual Computing

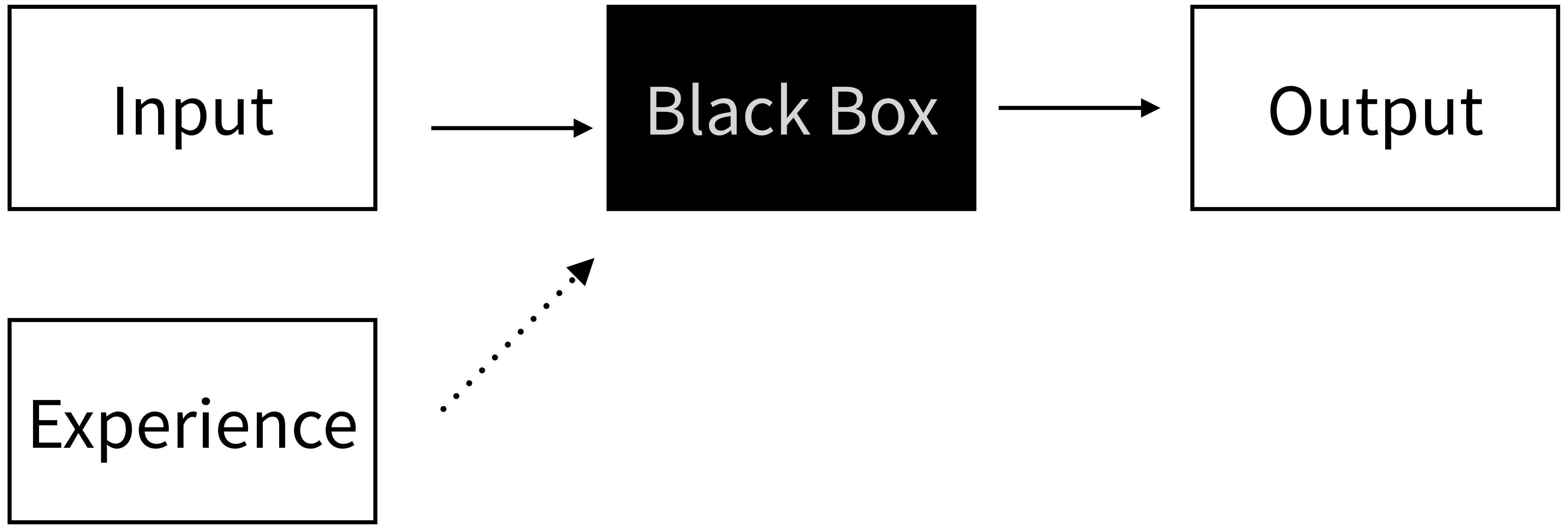




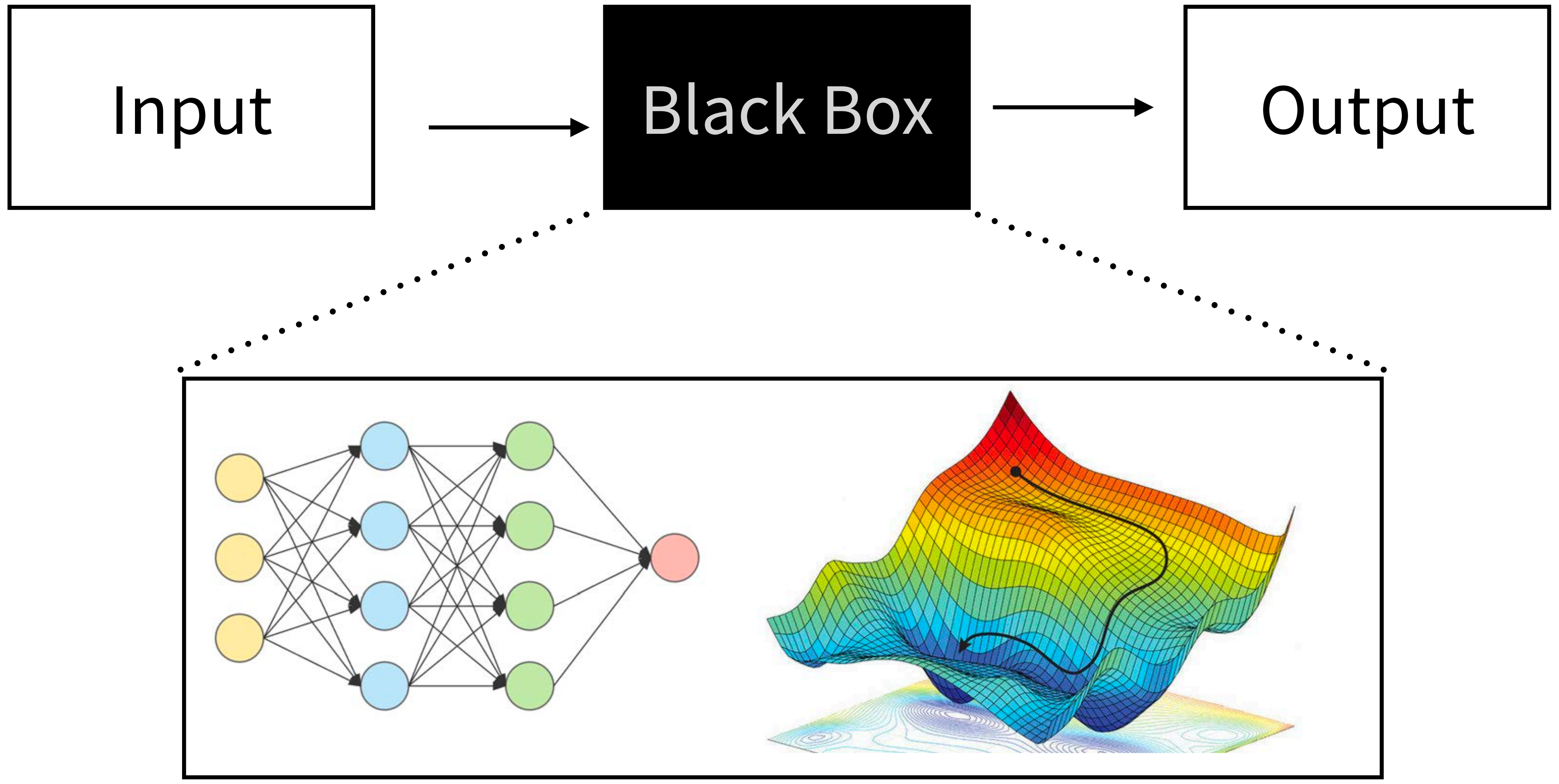
Neural Representations for recovering
3D from Point Clouds or Images

Image credits: IGR: Implicit Geometric Regularization for Learning Shapes. *Gropp et. al,* Neural Reflectance Fields. *Mildenhall et. al*

Learning-Based 3D Visual Computing



Data-driven inference of/on
3D Representations

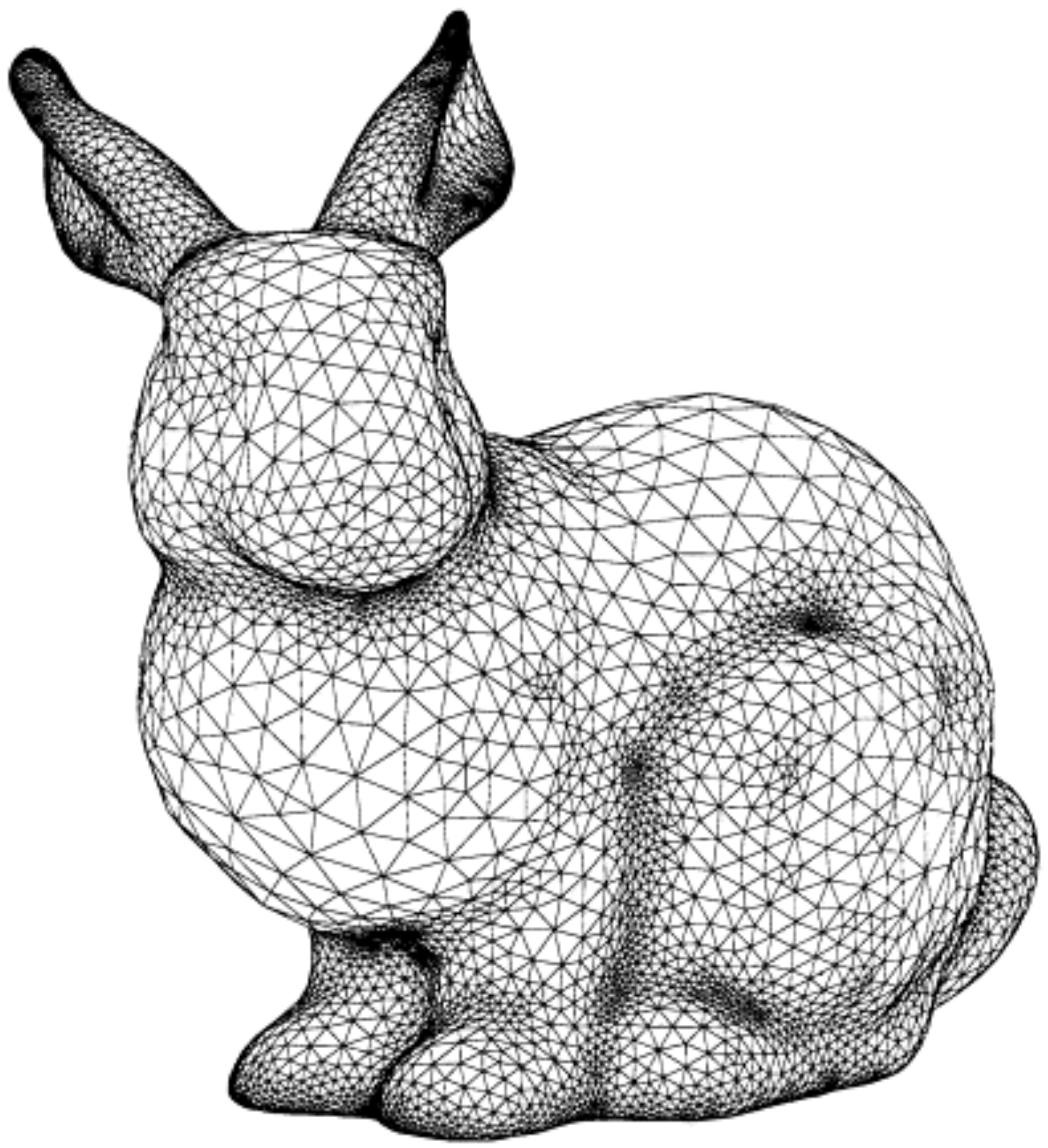


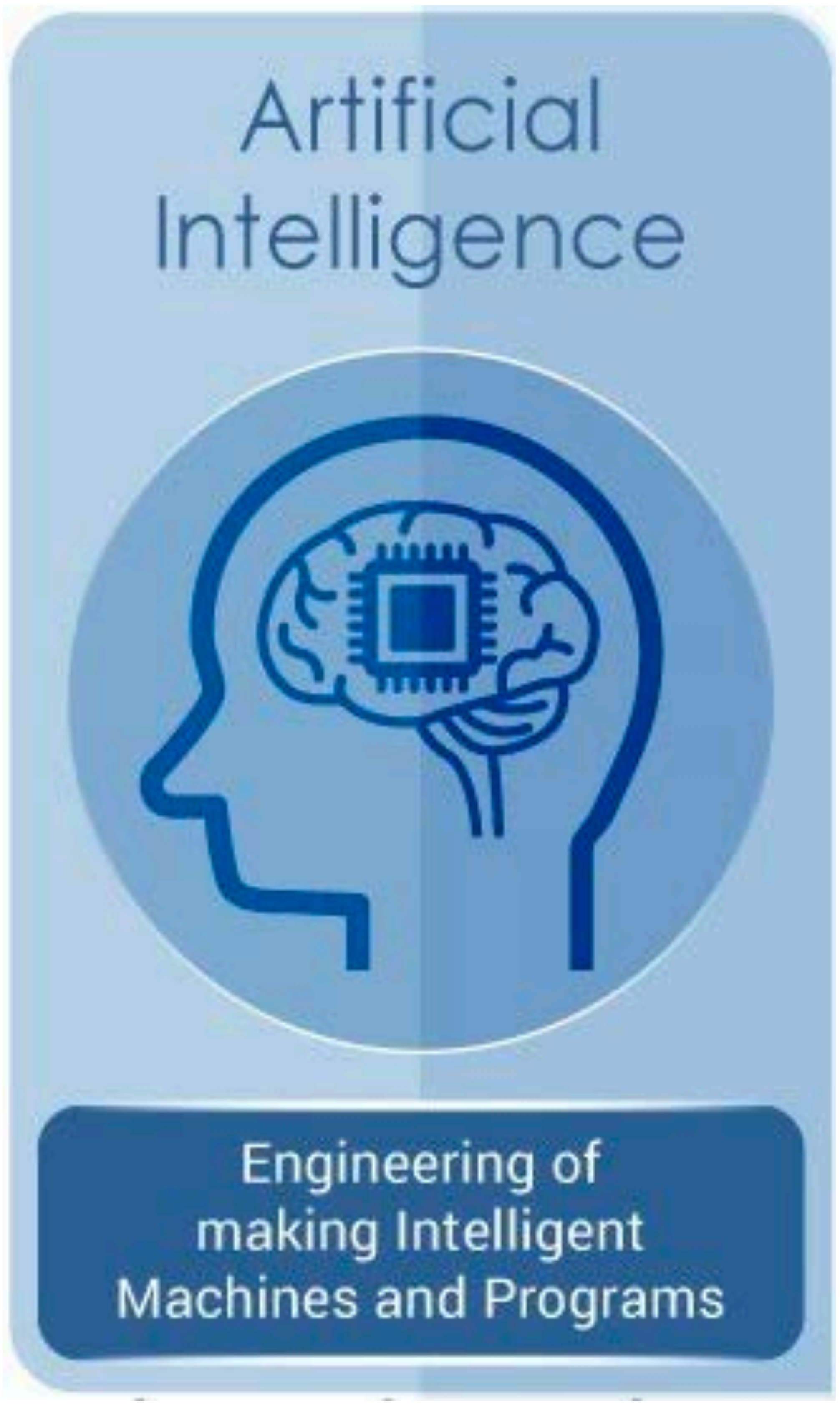
3D-specific Learning Mechanisms
and (Neural) Representations

Agenda today

- A brief overview of 3D visual computing
- Course Overview
  - ***Topics covered and syllabus***
  - Logistics
- Curve and Surface

Topics covered in this course



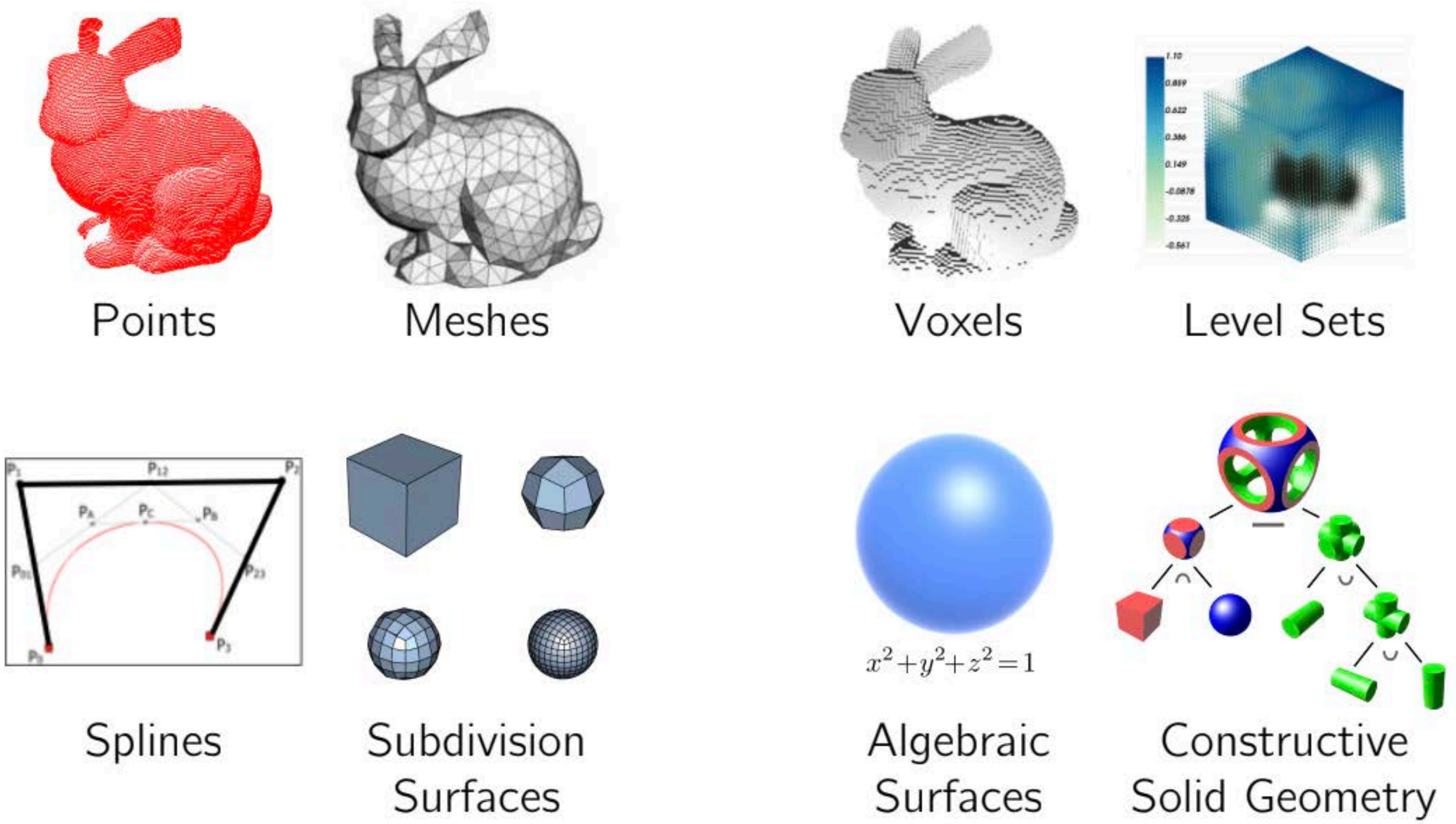


Topics covered in this course

- Basic geometric entities

Topics covered in this course

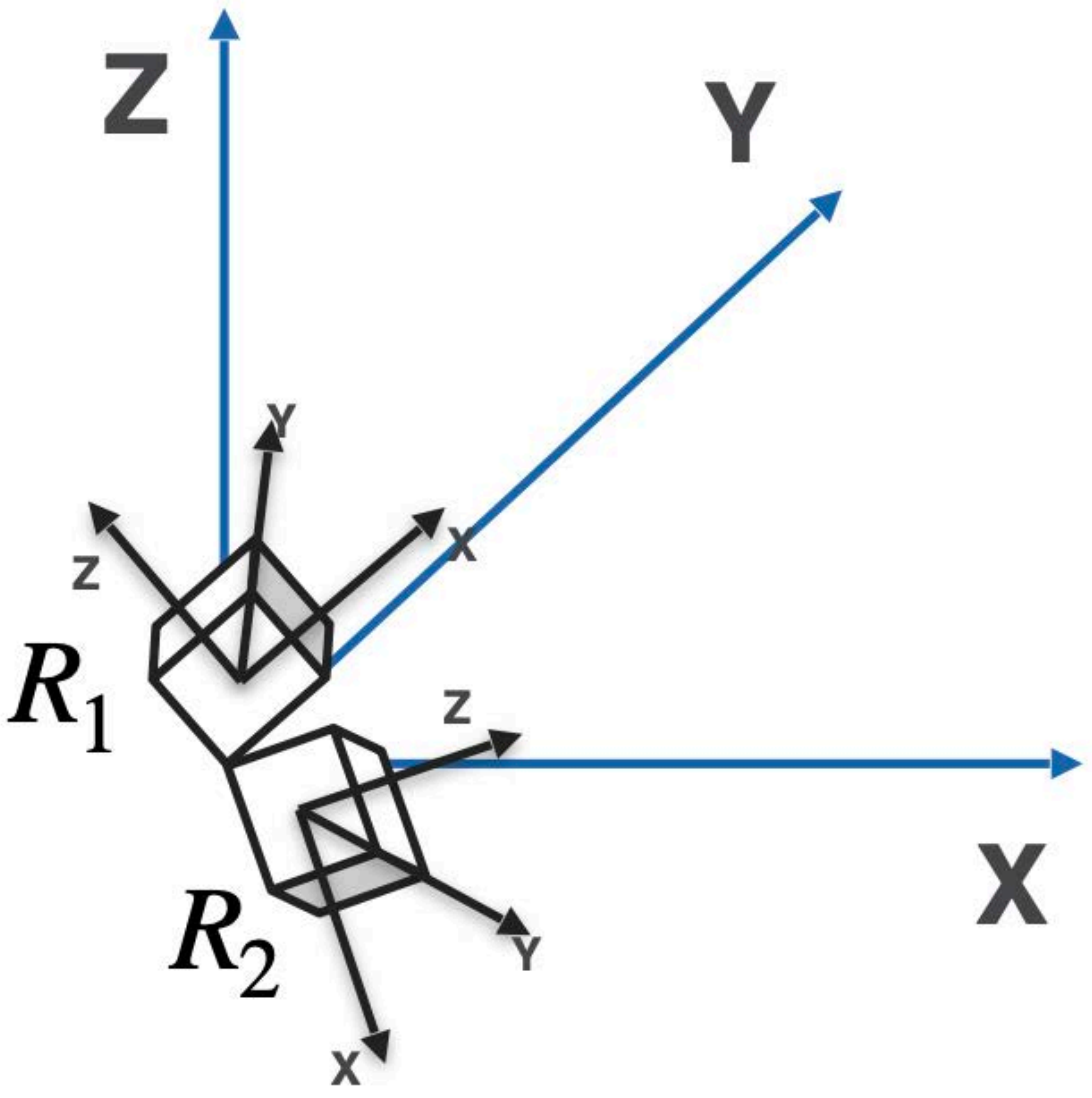
- 3D representations



Slides credit: Jiajun Wu

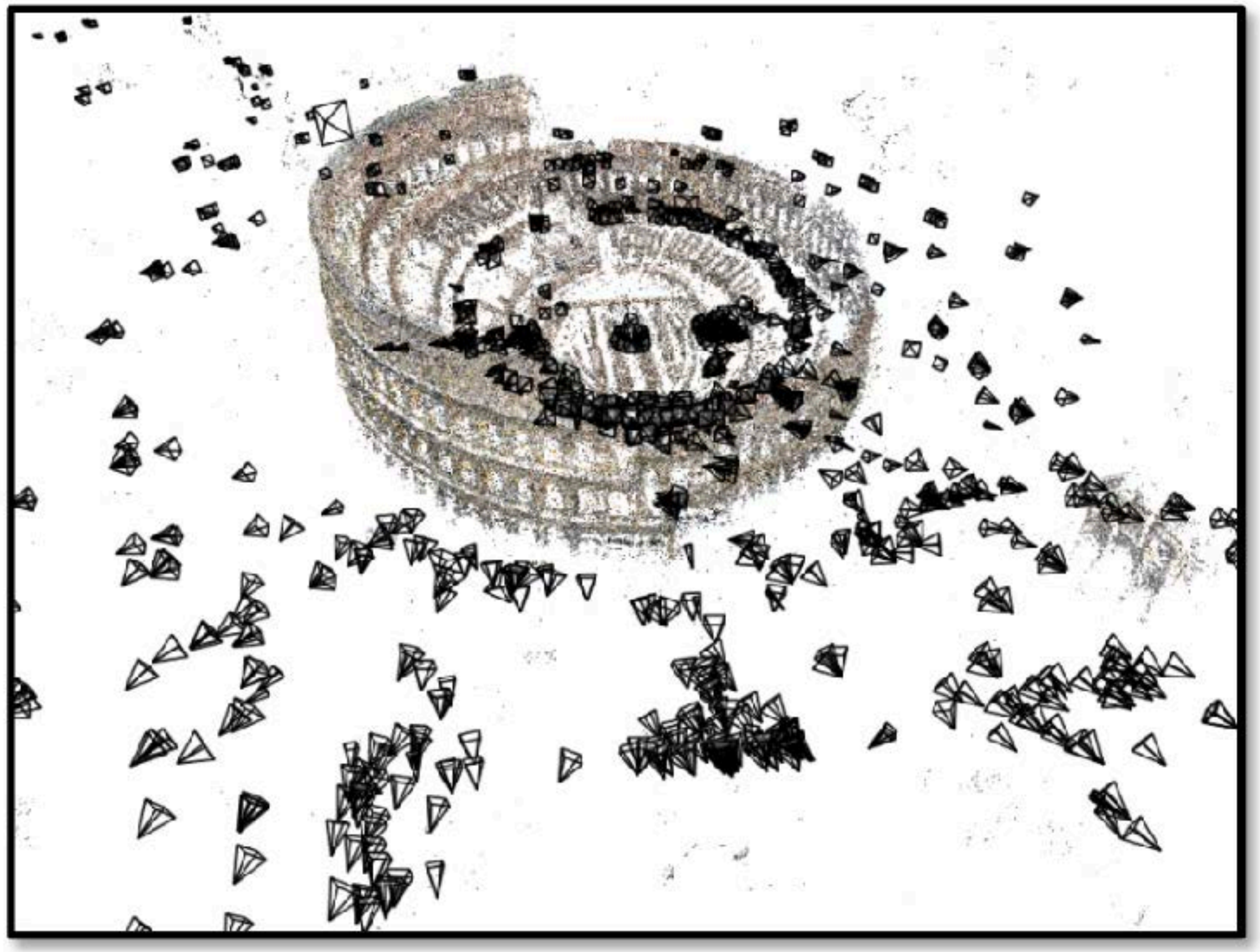
Topics covered in this course

- 3D transformations



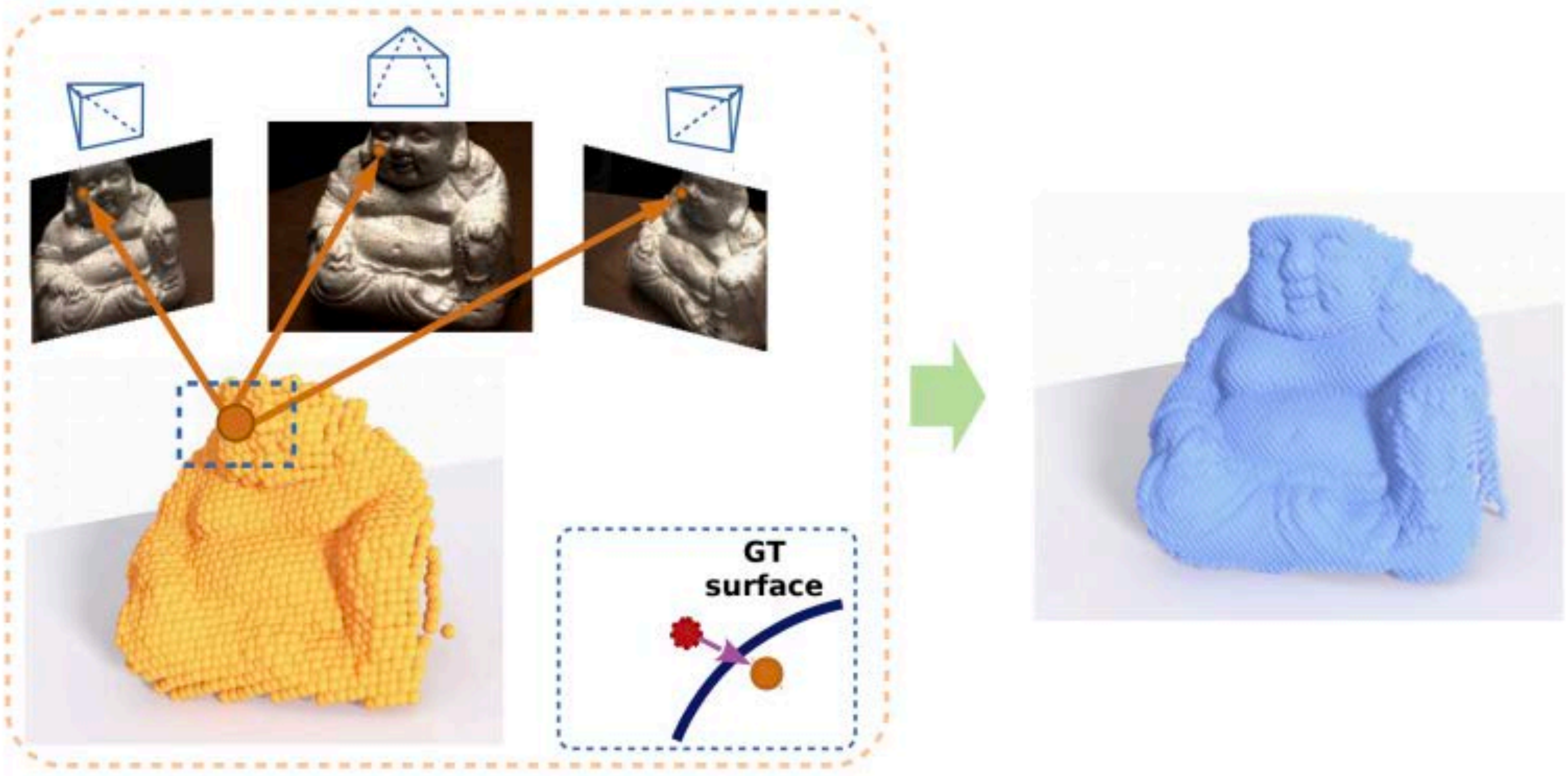
Topics covered in this course

- 3D reconstruction and synthesis - SFM and SLAM



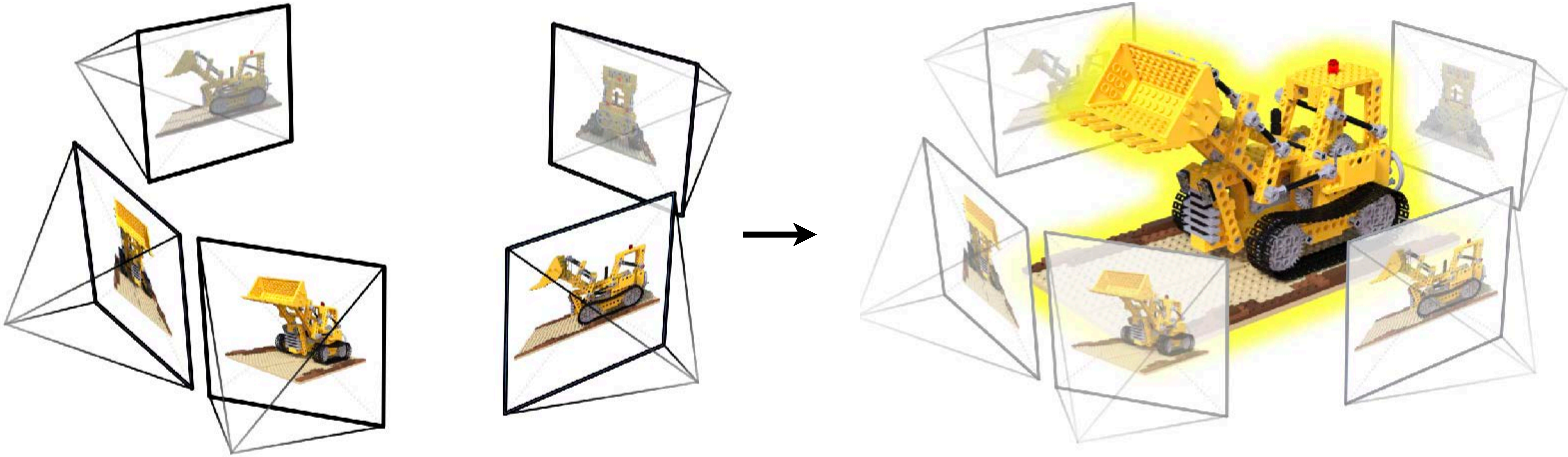
Topics covered in this course

- 3D reconstruction and synthesis - learning-based multiview stereo



Topics covered in this course

- 3D reconstruction and synthesis - NeRF

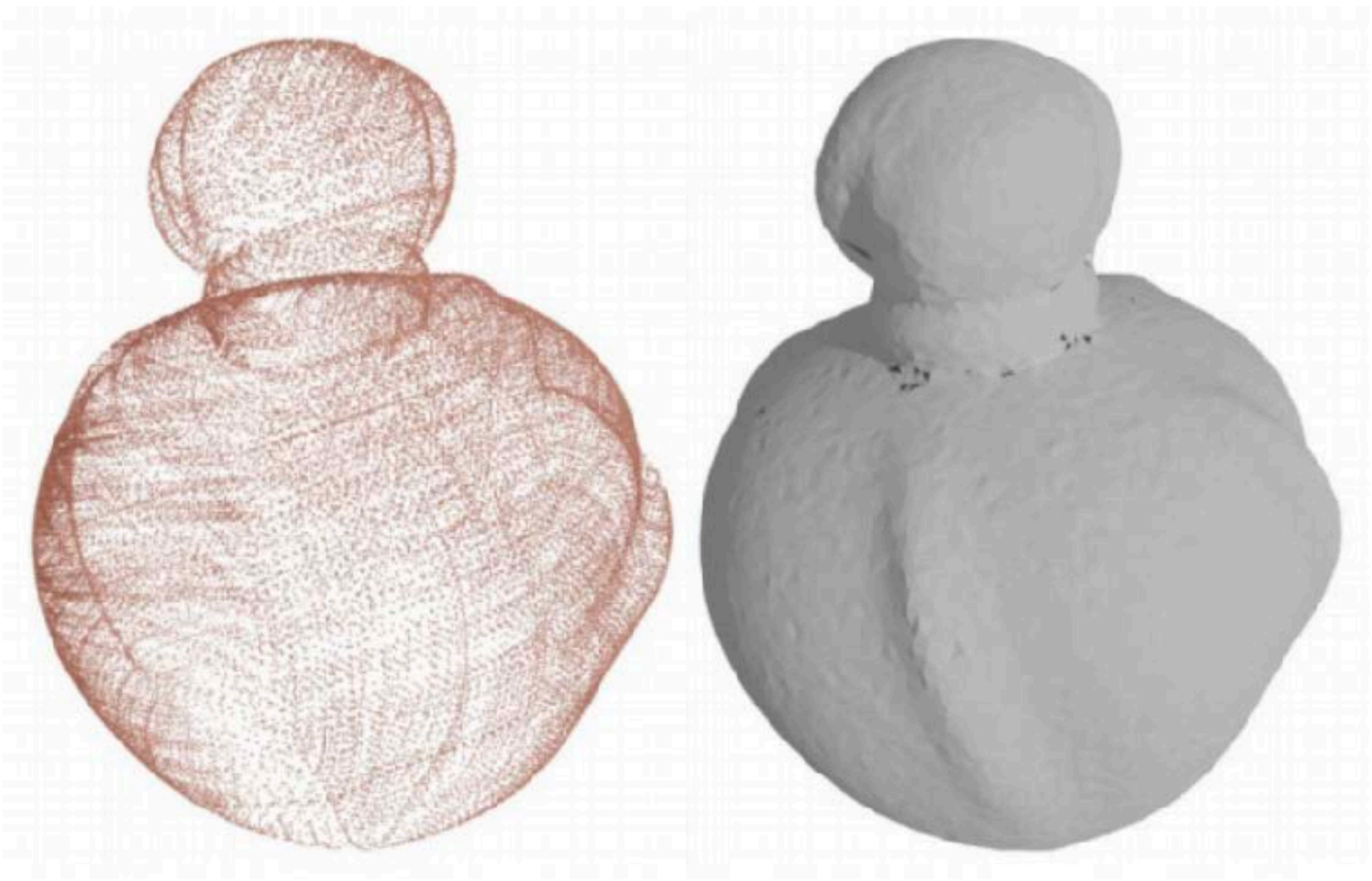


input images + camera poses → 3D scene representation

Topics covered in this course

- 3D reconstruction and synthesis

Single Image to 3D



Surface Reconstruction

Topics covered in this course

- 3D reconstruction and synthesis - AIGC in 3D

a sculpture of a rooster.

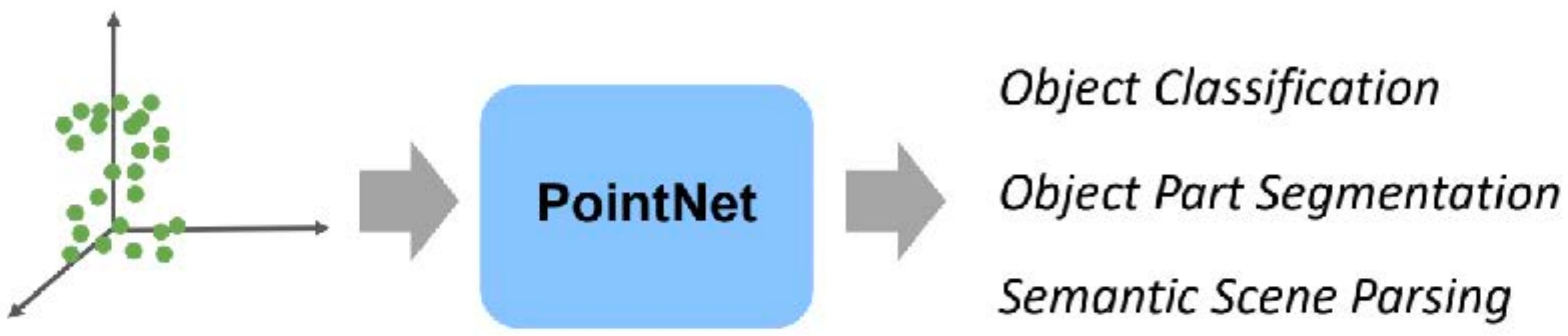
a robotic dog. a robot in the shape of a dog.

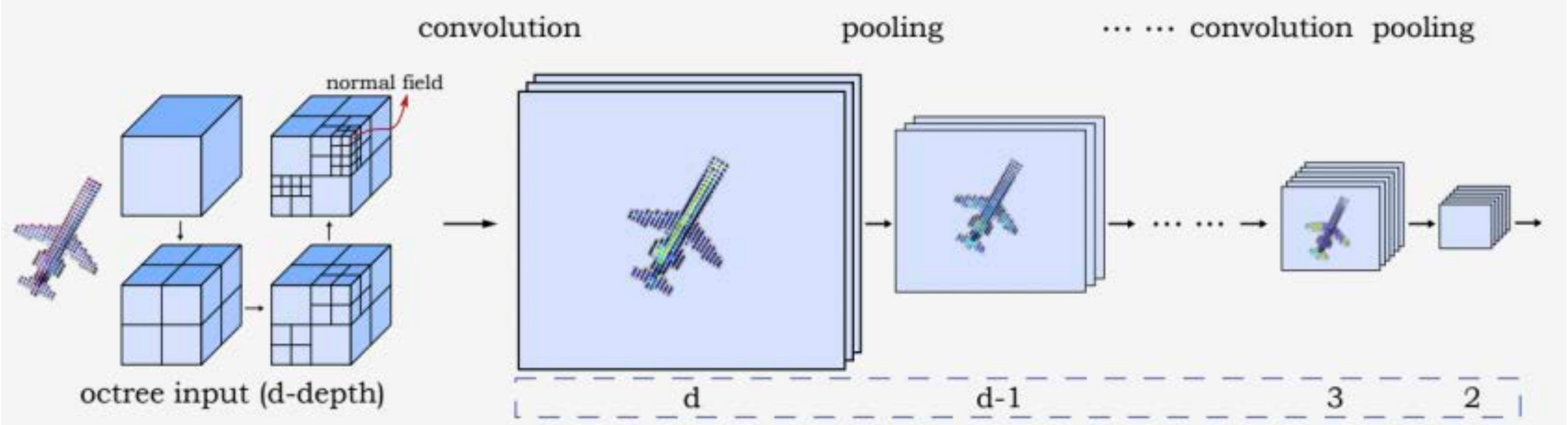
a pile of crab is seasoned and well cooked.

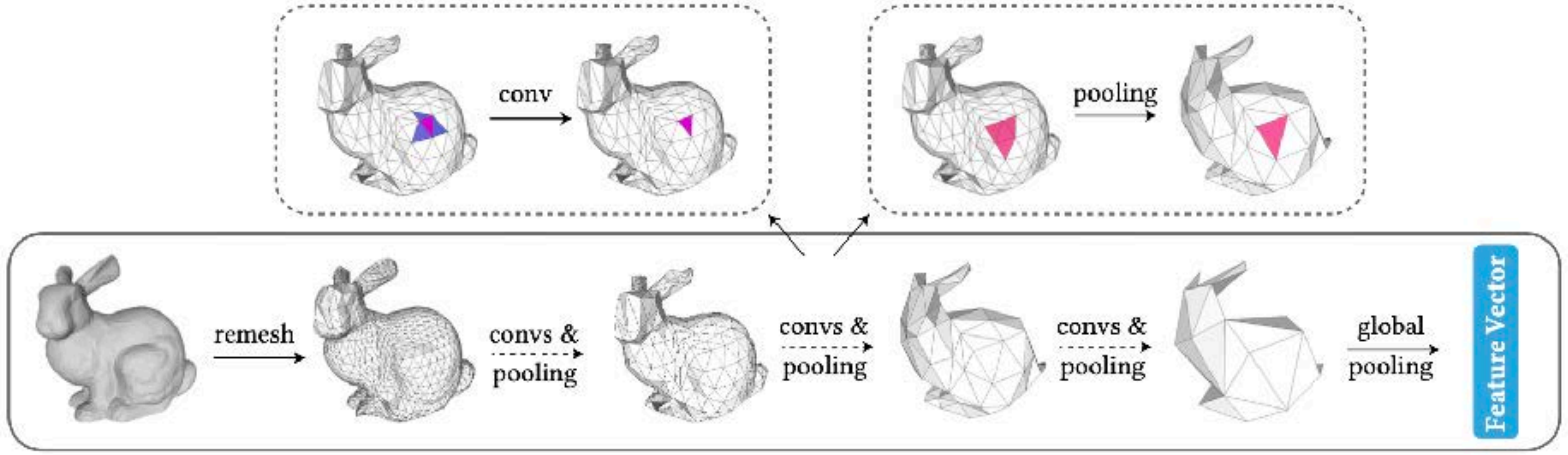
Point cloud diffusion model

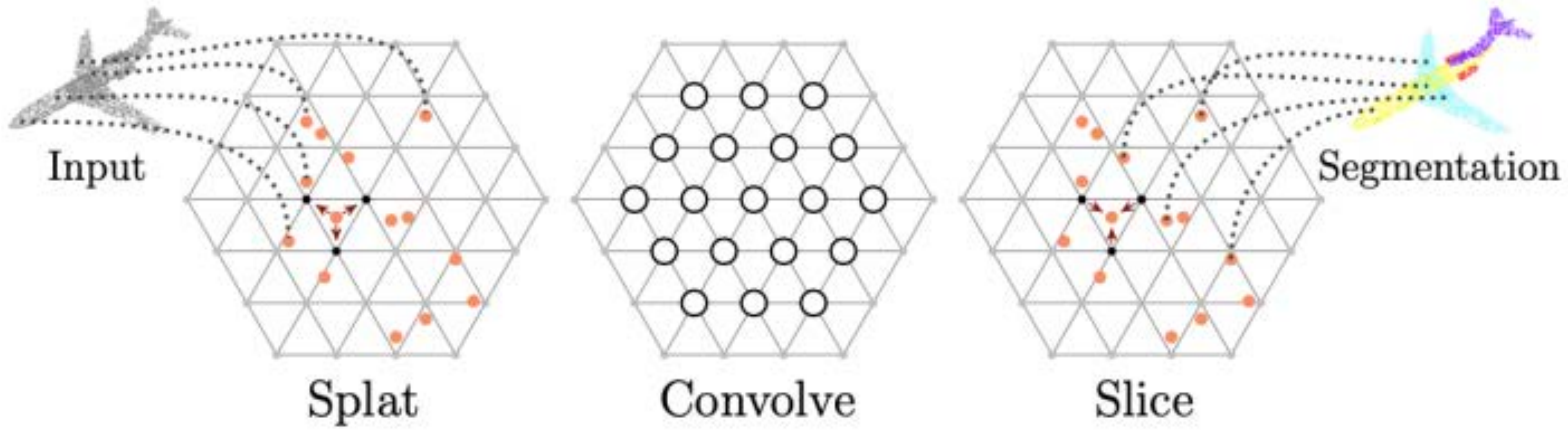
Topics covered in this course

- 3D analysis - 3D deep learning backbones



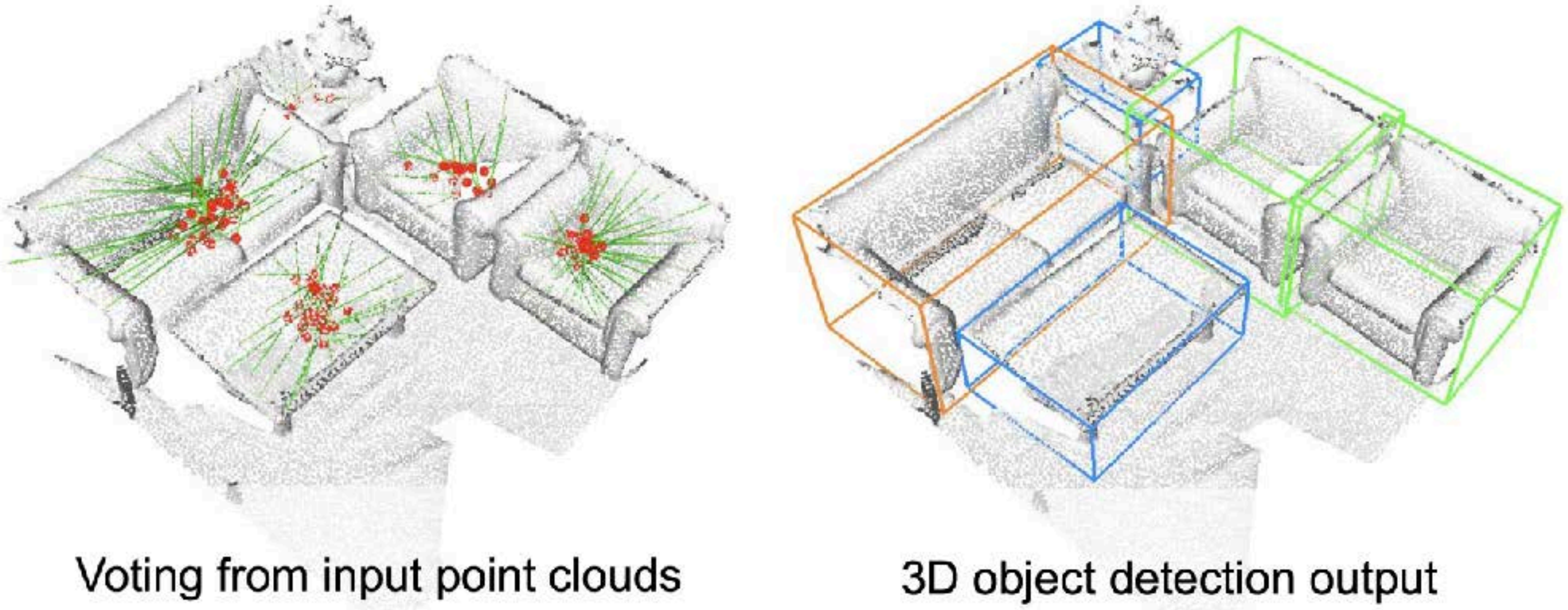




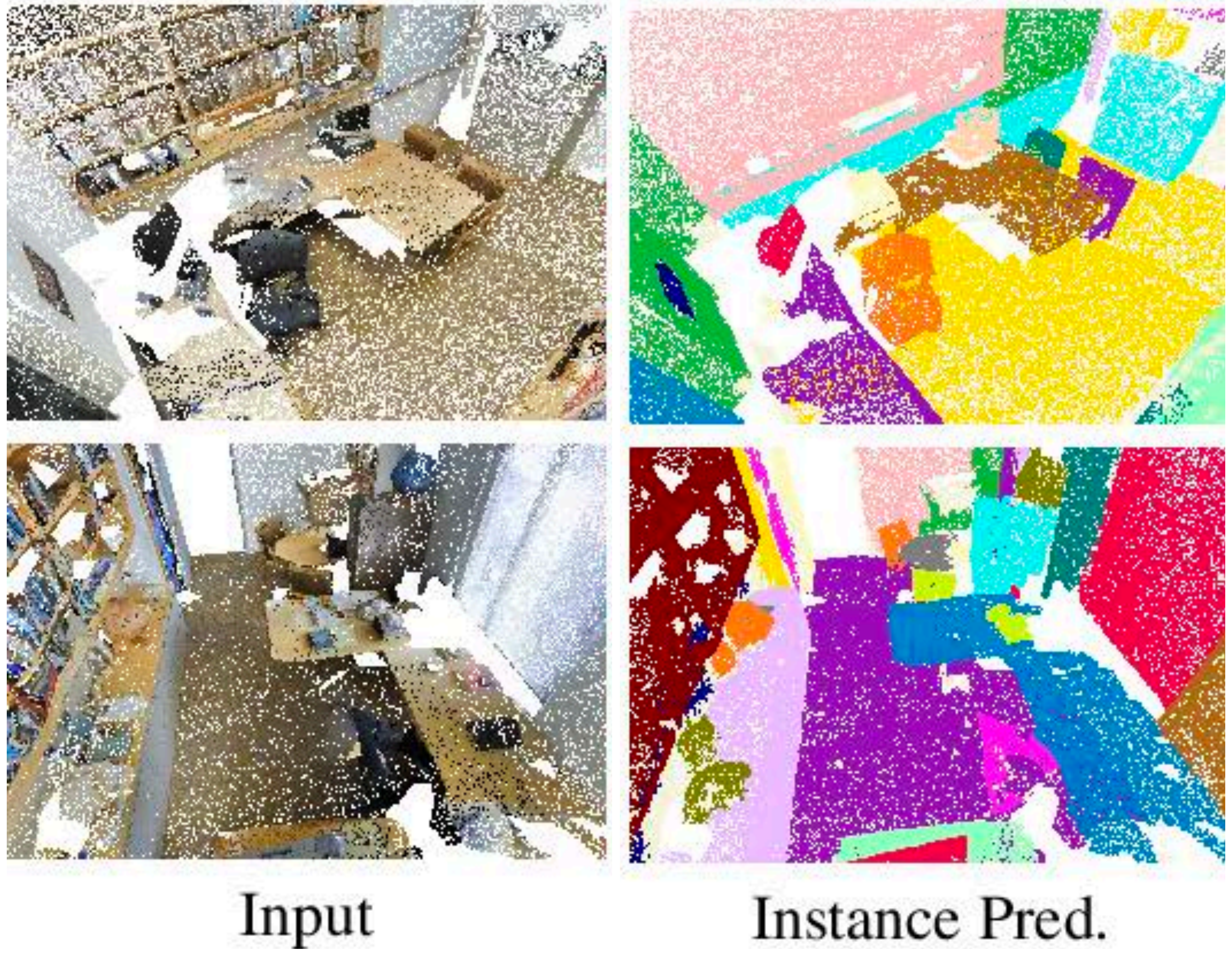


Topics covered in this course

- 3D analysis - recognition



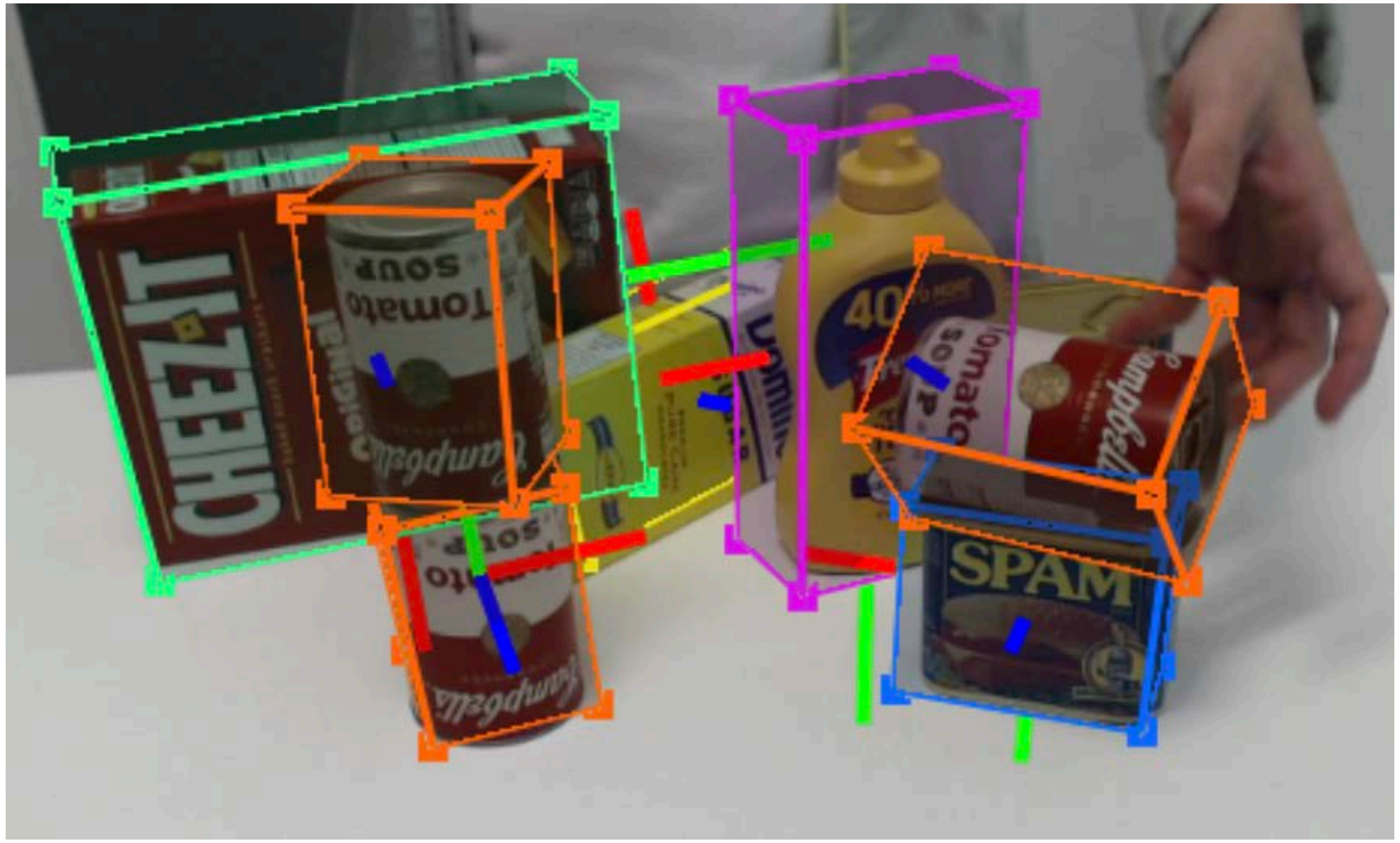
3D Detection



3D Segmentation

Topics covered in this course

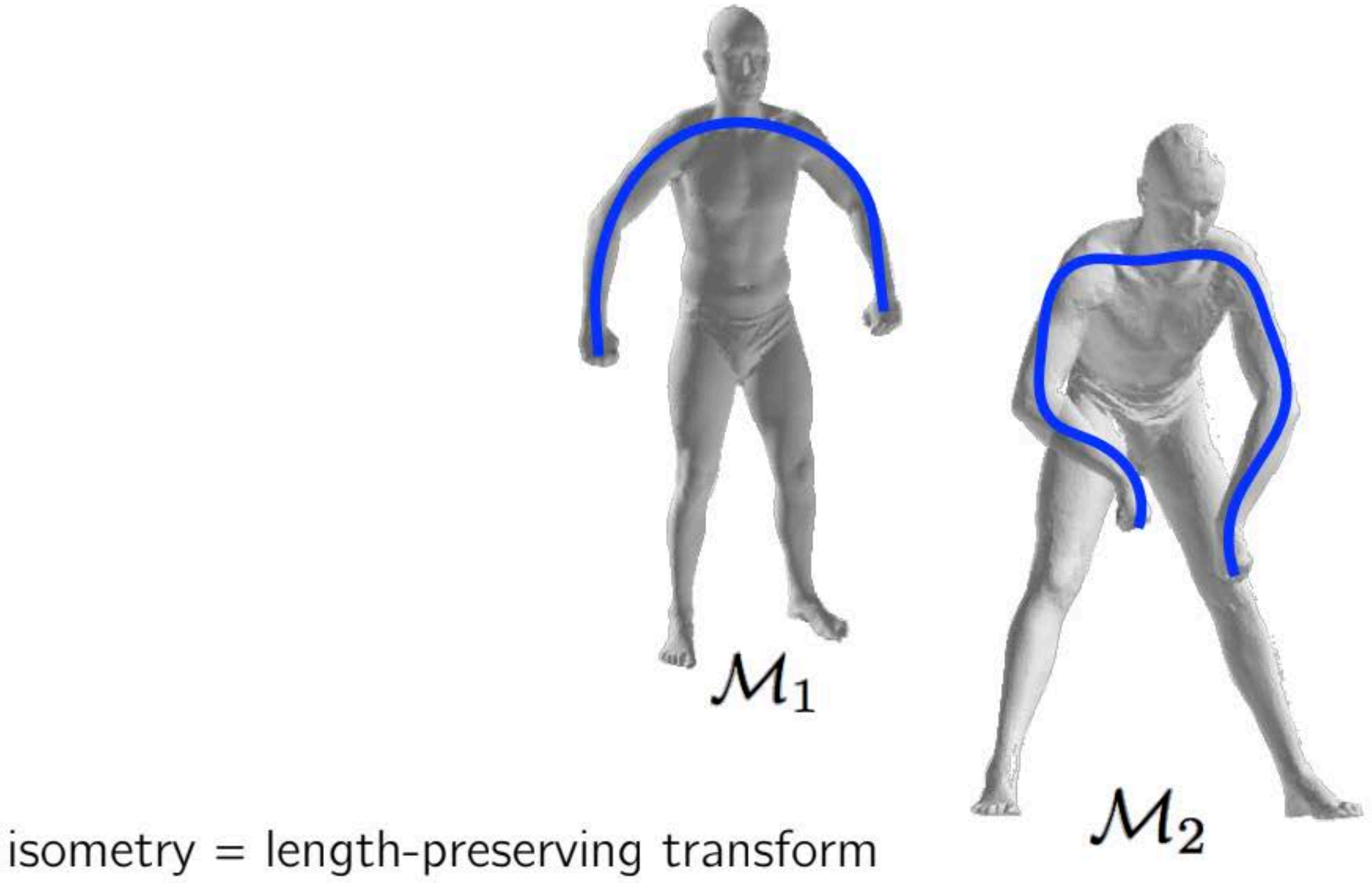
- 3D analysis - recognition



Pose Estimation

Topics covered in this course

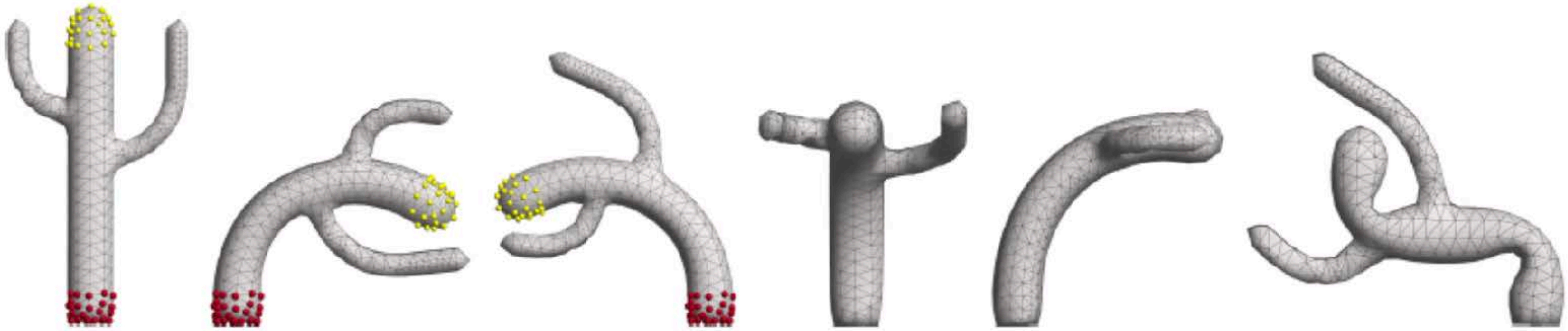
- 3D analysis - intrinsic shape understanding

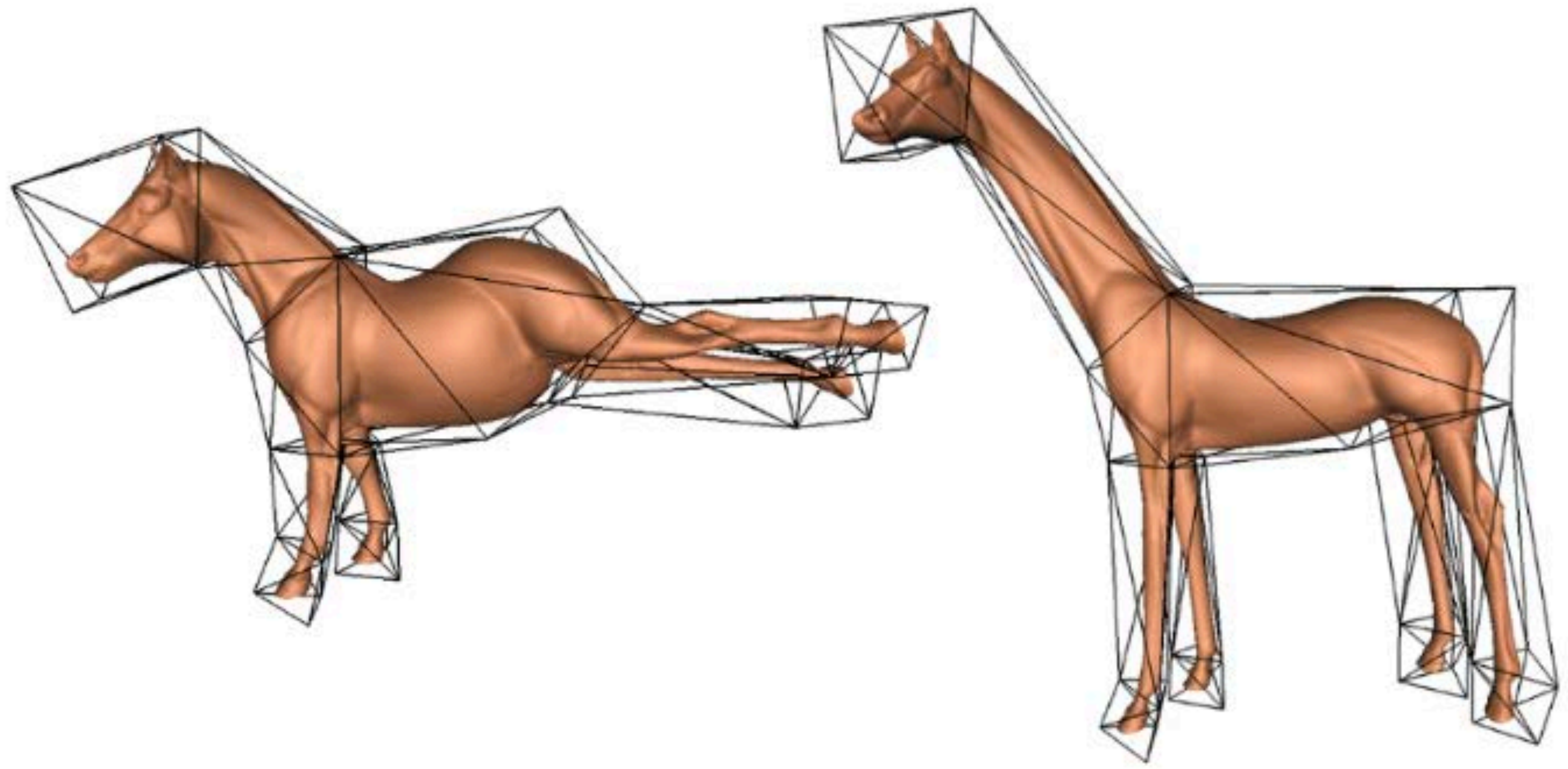


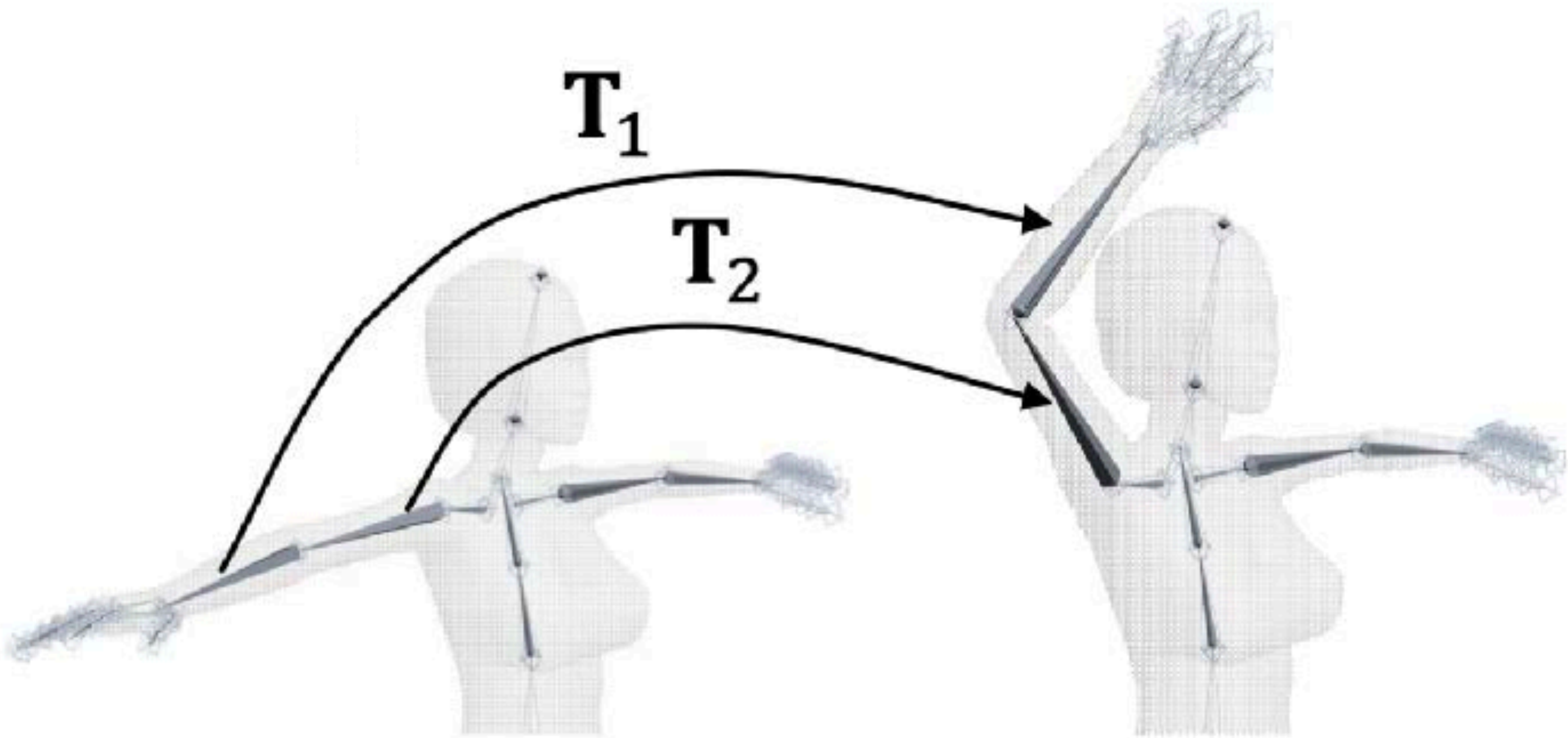
Slides credit: Hao Su

Topics covered in this course

- 3D analysis - shape deformation







Syllabus

Syllabus for 3D Visual Computing Spring 2025

	Geometric Foundation		**Apr 17 (Week 9)**	Sufrace Reconstruction	Project Announced
Feb 20 (Week 1)	Curve and Surface			***3D Analysis***	
Feb 27 (Week 2)	3D Representations	Hw1 Out	**Apr 24 (Week 10)**	3D Backbone Networks	Hw2 Due;
Mar 6 (Week 3)	3D Transformations		**May 1 (Week 11)**	Holiday	
	3D Reconstruction and Synthesis		**May 8 (Week 12)**	3D Backbone Networks II	In-Class Midterm
Mar 13 (Week 4)	Learning-Based SFM and SLAM		**May 15 (Week 13)**	3D Detection and Segmentation	
Mar 20 (Week 5)	Learning-Based Multi-view Stereo	Hw1 Due; Hw 2 Out	**May 22 (Week 14)**	Pose Estimation	
Mar 27 (Week 6)	NeRF		**May 29 (Week 15)**	Intrinsic Shape Understanding	
Apr 3 (Week 7)	Single Image to 3D		**Jun 5 (Week 16)**	Shape Deformation	
Apr 10 (Week 8)	3D Generative Models		**Jun 12 (Week 17)**		Project Due

Agenda today

- A brief overview of 3D visual computing
- Course Overview
  - Topics covered and syllabus
  - ***Logistics***
- Curve and Surface

Web Learning (网络学堂)

- We use Web Learning to
  - Make announcement
  - Distribute course slides
  - Handle assignments and projects

Wechat group

- Contribute
  - Build a community on Wechat
  - Casual discussion
  - Help one another
  - Discuss topics you enjoy
- TA will answer your questions in the Wechat group

Office hours

- Every Friday 4-5pm @ Qidi Building C19
- Email me beforehand if you plan to come
- Ask me anything but not homework

What we will not learn in this class

- Computer Graphics
  - We will use basics of rendering, but will not cover the whole pipeline of rendering, simulation or animation
  - Consider taking the Advanced Computer Graphics (from me in autumn semester)
- Computer Vision
  - We will talk about 3D computer vision a lot without delving deep into basic 2D computer vision problems and CNNs
  - Consider taking the Computer Vision course (provided by Prof. Yang Gao)
- Physical Simulation
  - We talk about reconstructing and synthesizing static 3D content but how to faithfully simulate the dynamics in the real world is out of our scope
  - Consider taking the Physics-Based Simulation (provided by Prof. Tao Du)

Learning objectives

- Study approaches from **3D capturing**, **3D content creation** to **3D processing** and **3D prediction**
- Understand learning-based techniques common across 3D applications
- Gain an understanding where the field is and where it is heading for
- Hands on experience through programming assignments and projects
- ***Important:*** gain interest and have fun!

Pre-requisite

- Skilled in linear algebra
- Familiar with Multi-variable Calculus
- Familiar with Probability and Numerical Methods
- Good programming skills - familiar with python, numpy, and pytorch
- Basic understanding of computer vision or deep learning

Grading

- 2 assignments + midterm quiz + 1 final project
- HW1: due Mar 20 (20 points)
- HW2: due Apr 24 (20 points)
- Midterm quiz (10 points)
- Final project: due Jun 12 (50 points)
- No final exams
- Bonus (Q&A in WeChat group, each Q or A counts for 0.5 point, 5 points max)
- Late policy: 10% grade reduction for each 12 hours late

Collaboration policy

- Rule 1: Don’t look at solutions or code that are not your own; everything you submit should be your own work
- Rule 2: Don’t share your solution code with others; however discussing ideas or general strategies is fine and encouraged
- Rule 3: Indicate in your submissions anyone you worked with
- Turning in something late / incomplete is better than violating the rules

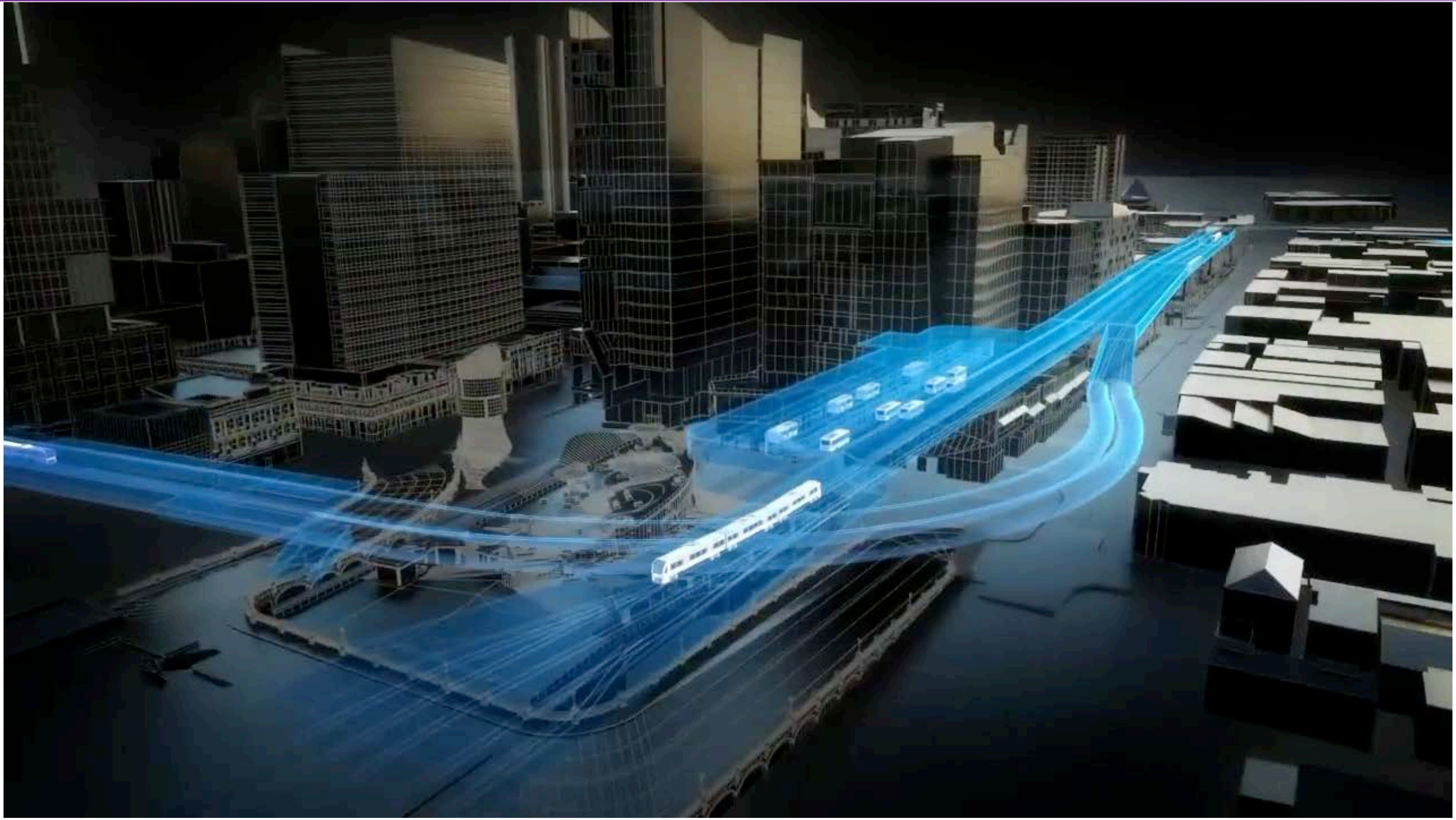
Optional textbooks and resources

- Yonghuai Liu, Nick Pears, Paul L. Rosin, Patrik Huber. 3D Imaging, Analysis and Applications, 2nd edition, Springer, 2020
- Yi Ma, Stefano Soatto, Jana Kosecka, Shankar Sastry. An Invitation to 3-D Vision From Images to Models, Springer, 2003.
- Frank Nielsen. Visual Computing: Geometry, Graphics And Vision, Charles River Media, Inc. 2005
- 3D machine learning material collection: https://github.com/timzhang642/3D-Machine-Learning

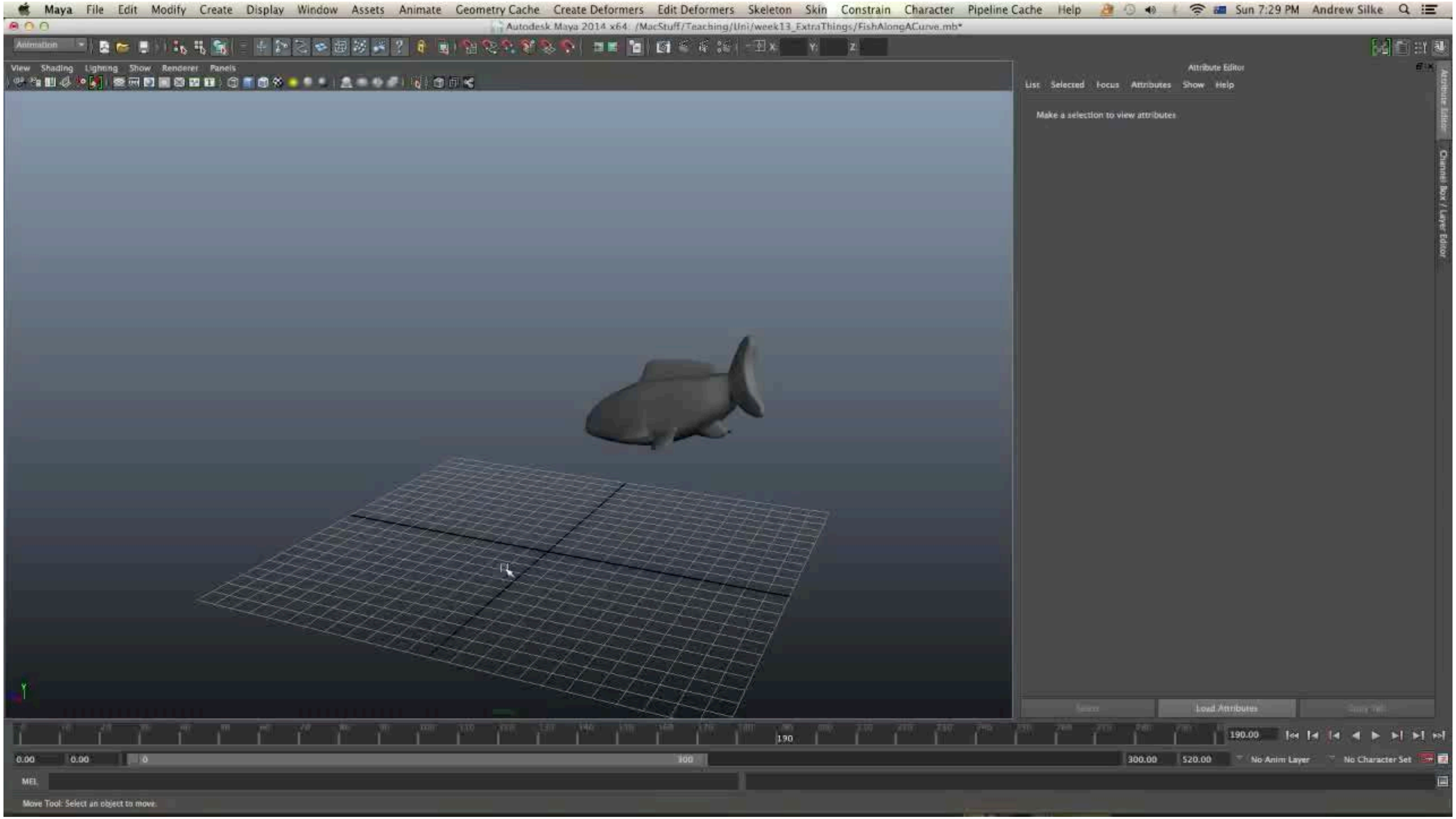
Agenda today

- A brief overview of 3D visual computing
- Course Overview
  - Topics covered and syllabus
  - Logistics
- ***Curve and Surface***

Camera Paths



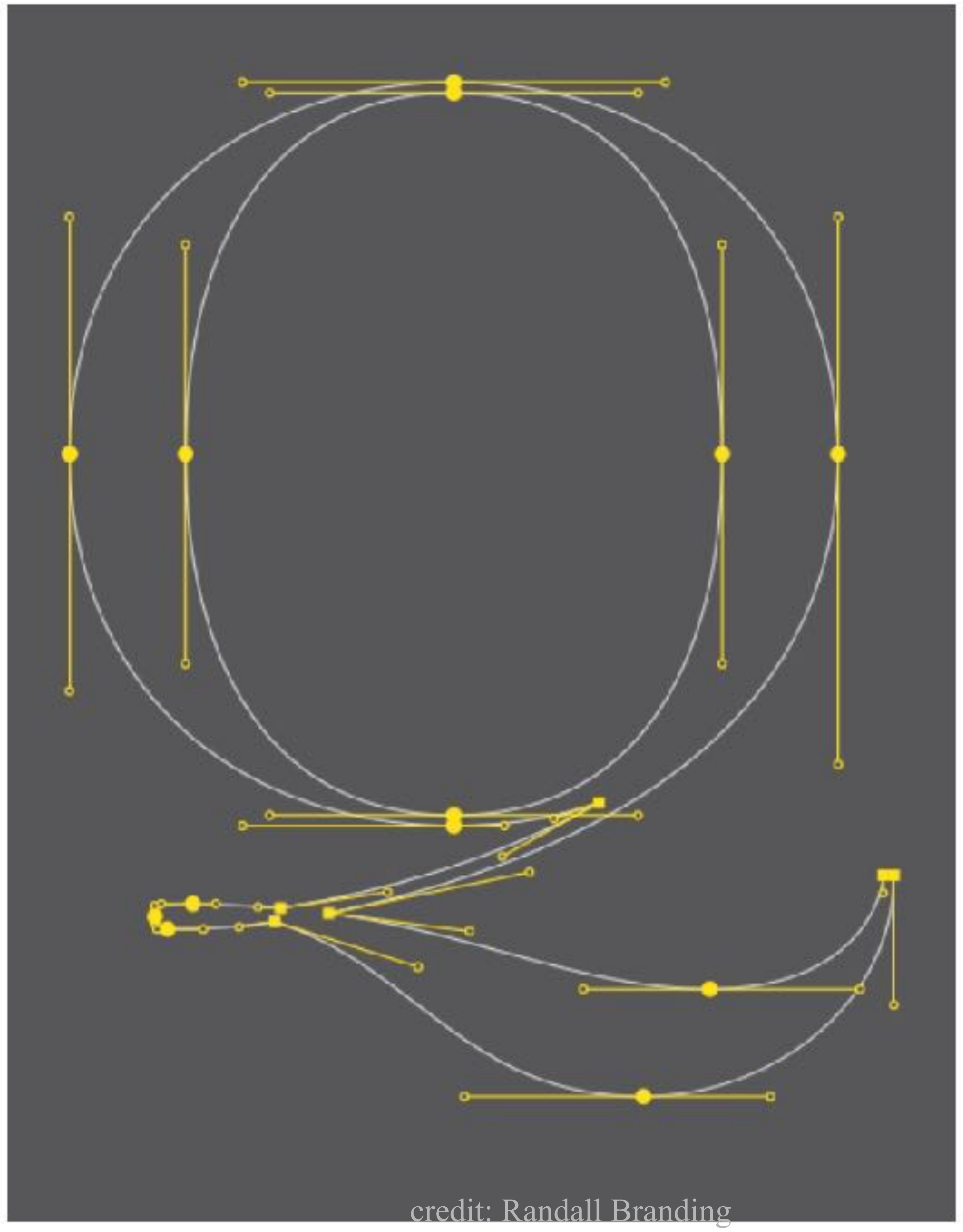
Animation Curves



Vector Fonts

The Quick Brown Fox Jumps Over The Lazy Dog

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz 0123456789



Baskerville font - represented as piecewise cubic Bézier curves

Curve

- Explicit curve $y = f(x)$
- Implicit curve $x^2 + y^2 = 1$
- Parametrized curves $\gamma(t) = (x(t), y(t))$
  - Intuition: a particle is moving in space and its position at time t

Example

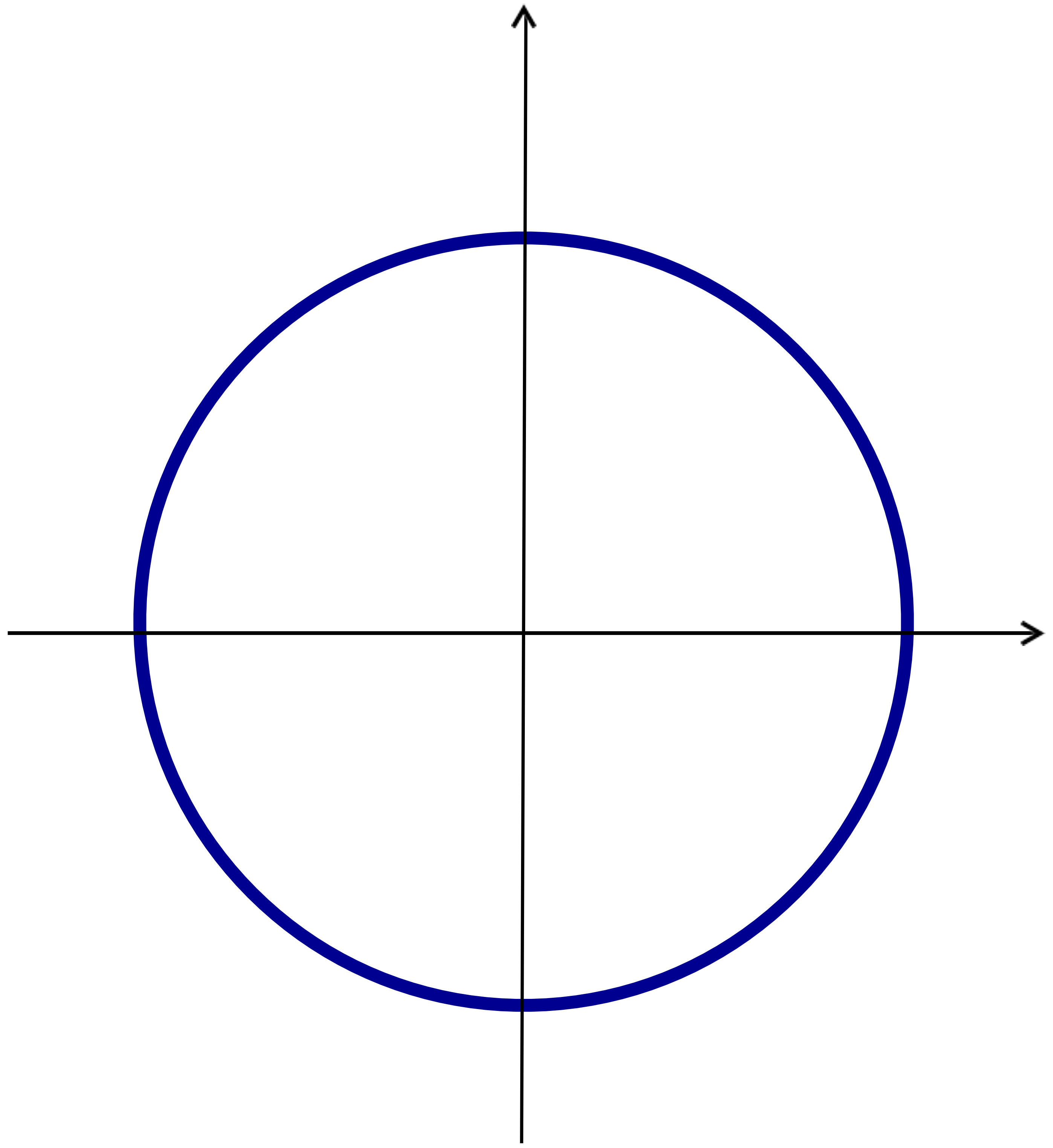
- Circle in 2D

$$\mathbf{p} : \mathbb{R} \to \mathbb{R}^2$$

$$t \mapsto \mathbf{p}(t) = (x(t), y(t))$$

$$\mathbf{p}(t) = r\left(\cos(t), \sin(t)\right)$$

$$t \in [0, 2\pi)$$

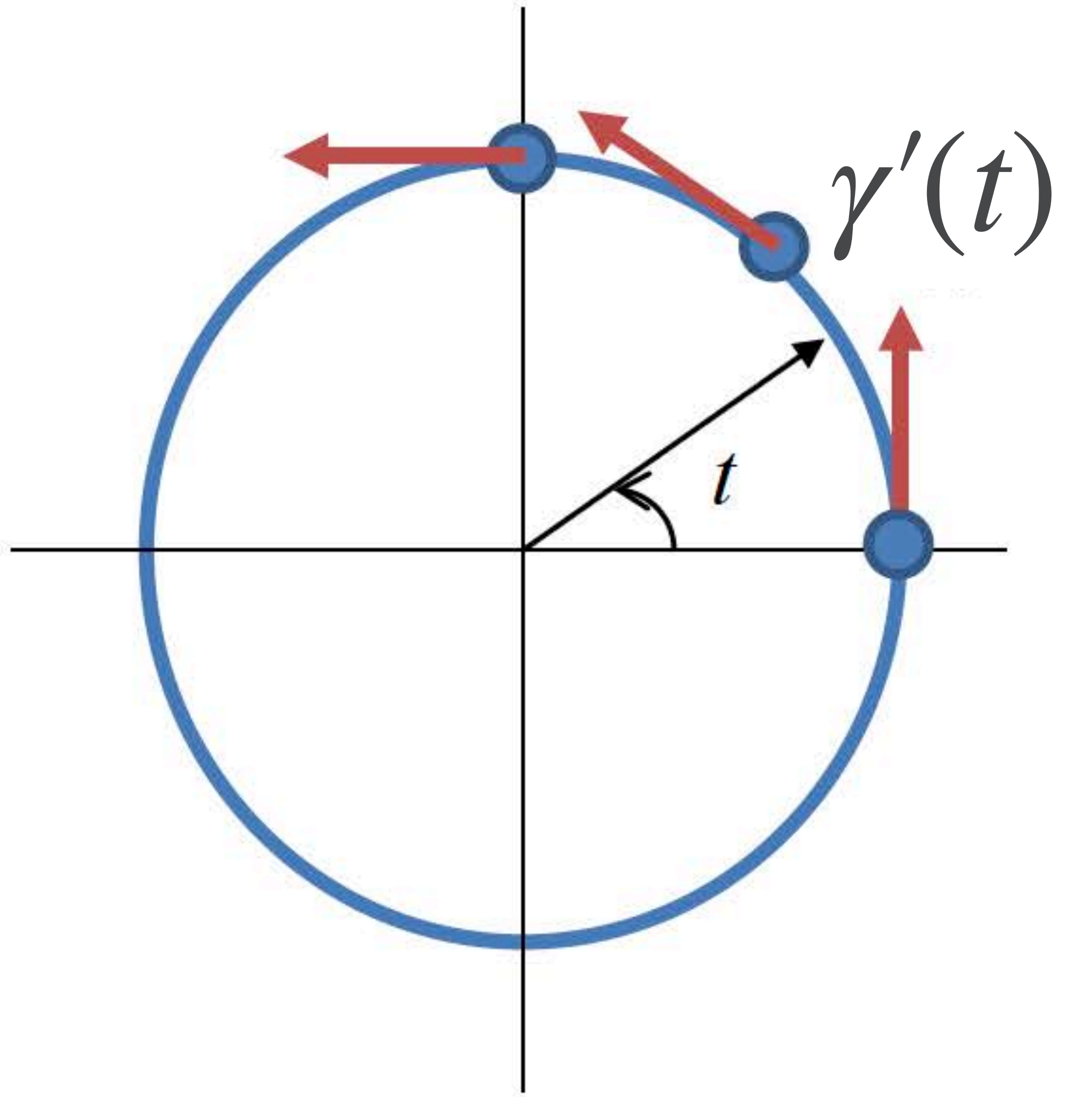


Tangent

- $\gamma'(t) = (x'(t), y'(t)) \in \mathbb{R}^2$ is the tangent vector of the curve $\gamma(t)$ at t

Quiz: Tangent of a circle

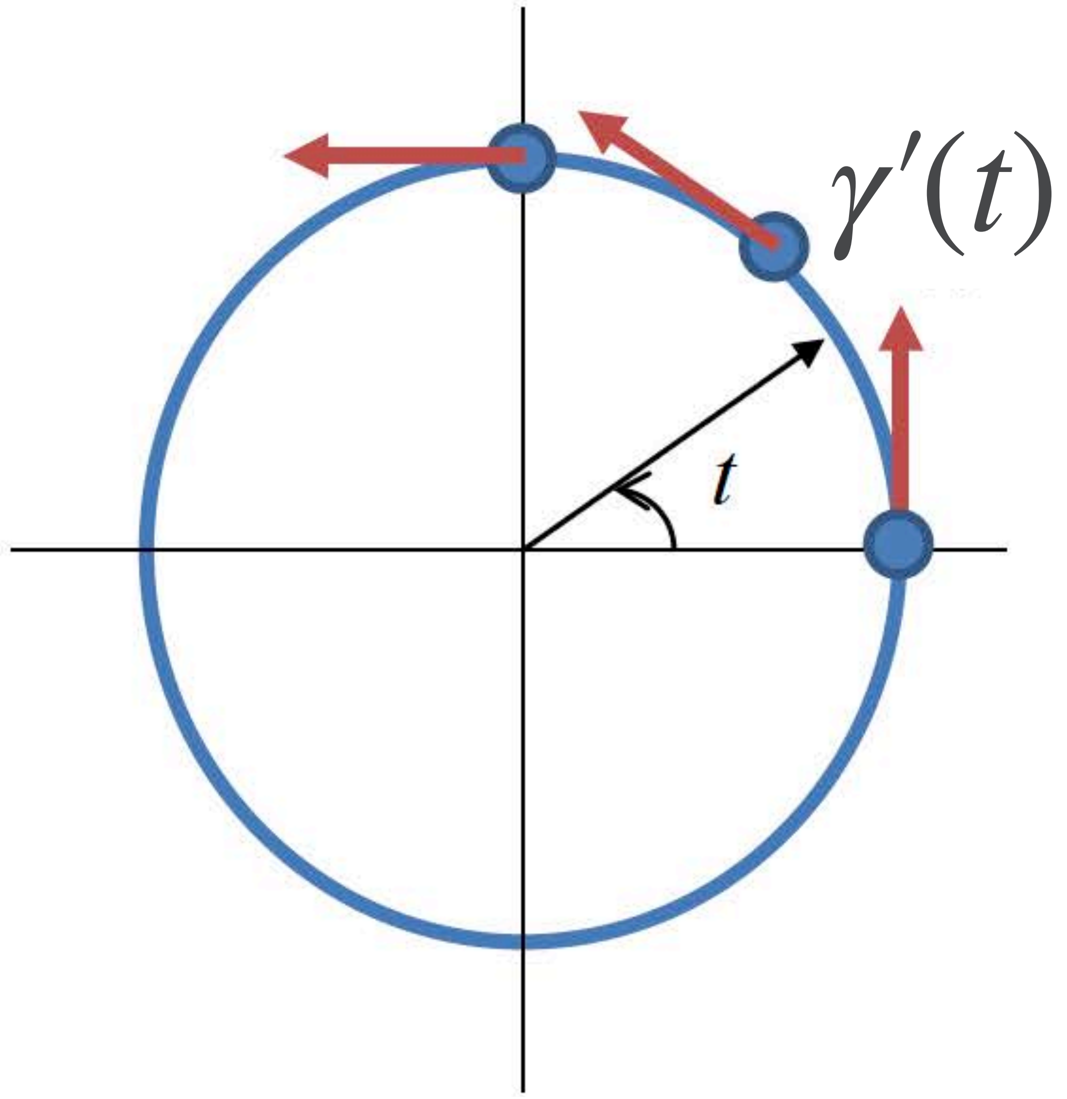
- $\gamma'(t) = (x'(t), y'(t)) \in \mathbb{R}^2$ is the tangent vector of the curve $\gamma(t)$ at t



$$\gamma(t) = (\cos(t), \sin(t))$$

Quiz: Tangent of a circle

- $\gamma'(t) = (x'(t), y'(t)) \in \mathbb{R}^2$ is the tangent vector of the curve $\gamma(t)$ at t



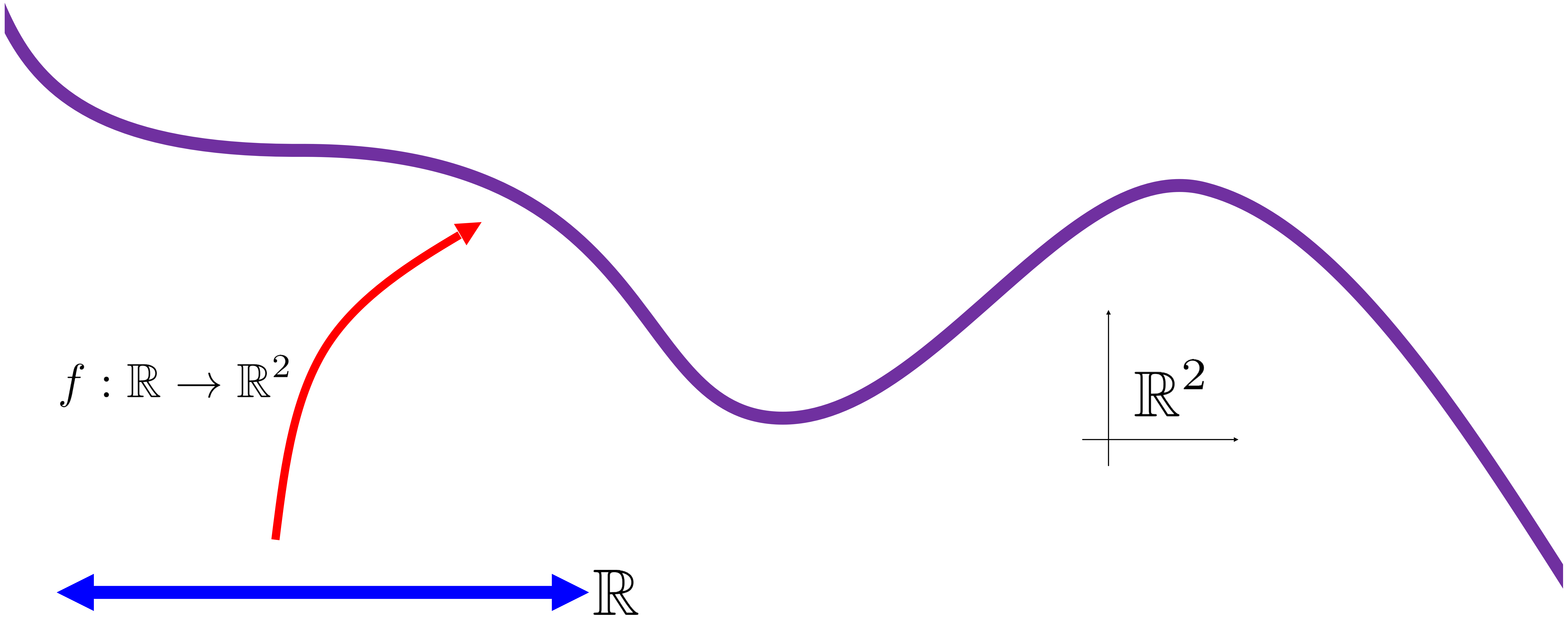
$$\gamma(t) = (\cos(t), \sin(t))$$

$$\gamma'(t) = (-\sin(t), \cos(t))$$

$\gamma'(t)$ - direction of movement

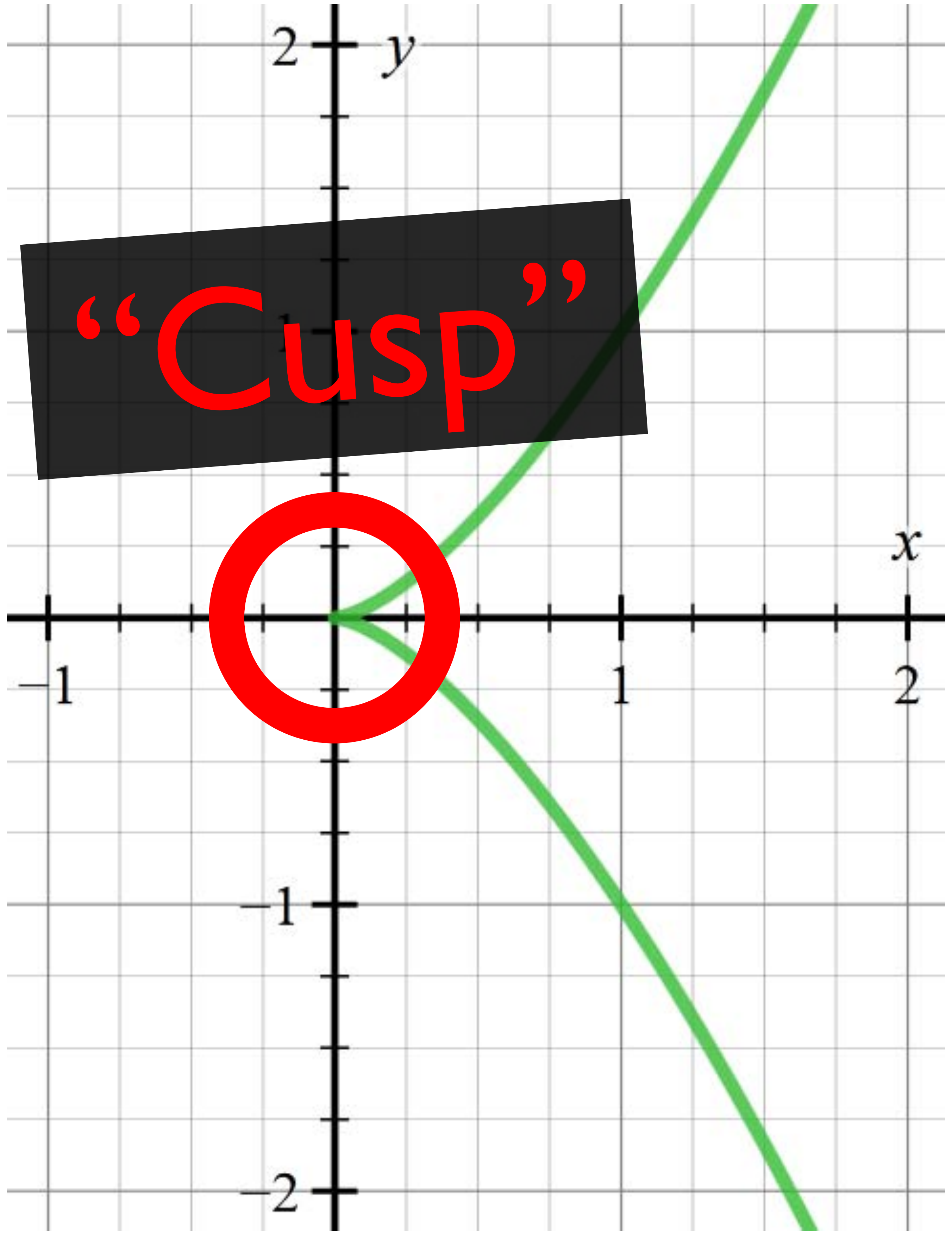
$||\gamma'(t)||$ - speed of movement

Is curve a function?



$$\gamma_3(t) := (0, 0)$$

Graphs of smooth functions



$$f(t) = (t^2, t^3)$$

How to ensure the smoothness of a curve?

Geometric definition

- A smooth curve is a set of points locally resembling one dimensional smooth "manifold"

Set of points that locally looks like a line.

Change of parameters

$$\gamma(t) = (x(t), y(t)), t \in (a, b)$$

$$\bar{t} = 2t$$

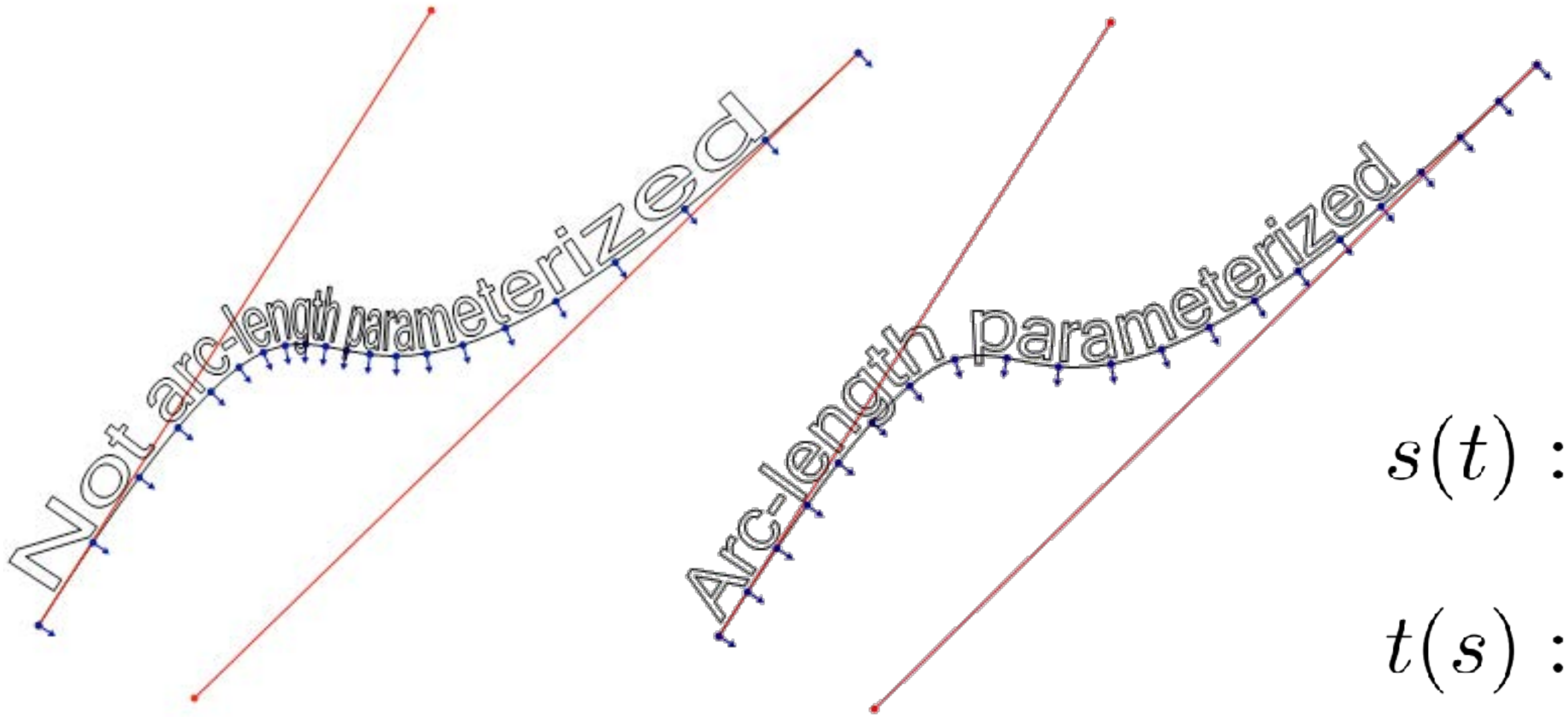
$$\gamma(\bar{t}) = (x(\frac{\bar{t}}{2}), y(\frac{\bar{t}}{2})), \bar{t} \in (2a, 2b)$$

- These coordinates are frame dependent and parametrization dependent
- Geometric measurements (how a curve bends, curve length, etc.) should be invariant to changes of frames and parametrizations

Arc length

$$\int_a^b \|\gamma'(t)\| \, dt$$

Parameterization by arc length



$$s(t) := \int_{t_0}^{t} \|\gamma'(t)\| \, dt$$

$$t(s) := \text{inverse of } s(t)$$

$$\gamma(s) := \gamma(s(t))$$

- Constant speed parametrization

Normal and curvature

Tangent: $T(s) = \gamma'(s)$ $\longrightarrow$ velocity

Normal: $N(s) = \dfrac{T'(s)}{||T'(s)||}$ $\longrightarrow$ acceleration direction

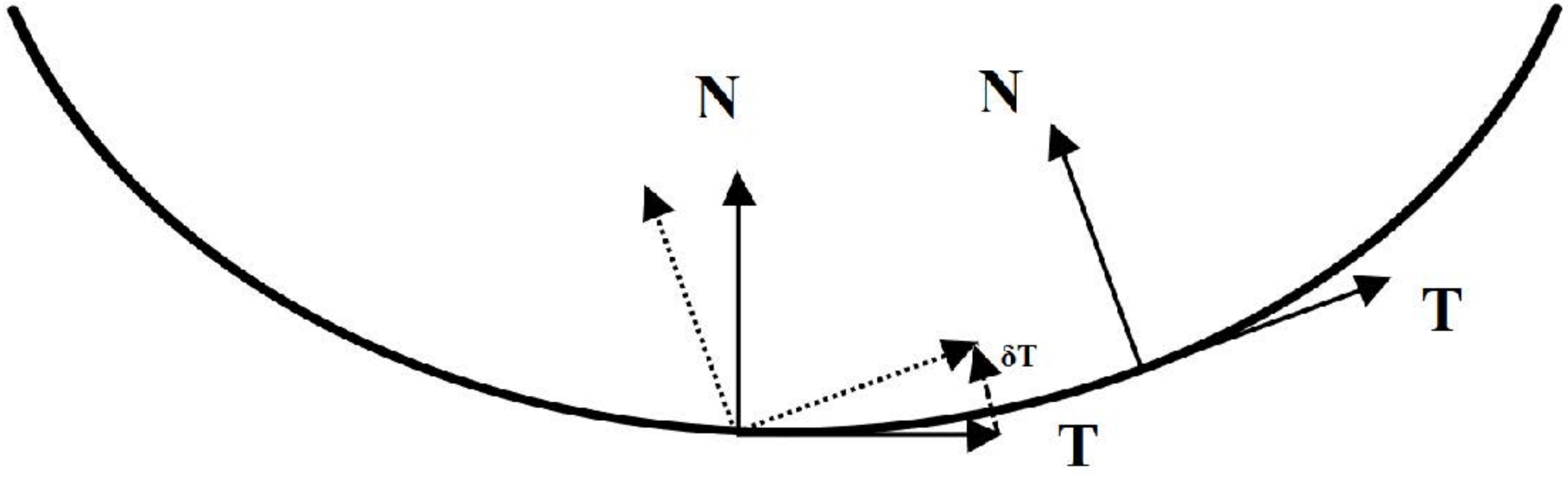
Curvature: $\kappa(s) = ||T'(s)||$ $\longrightarrow$ acceleration magnitude

Moving frame in 2D

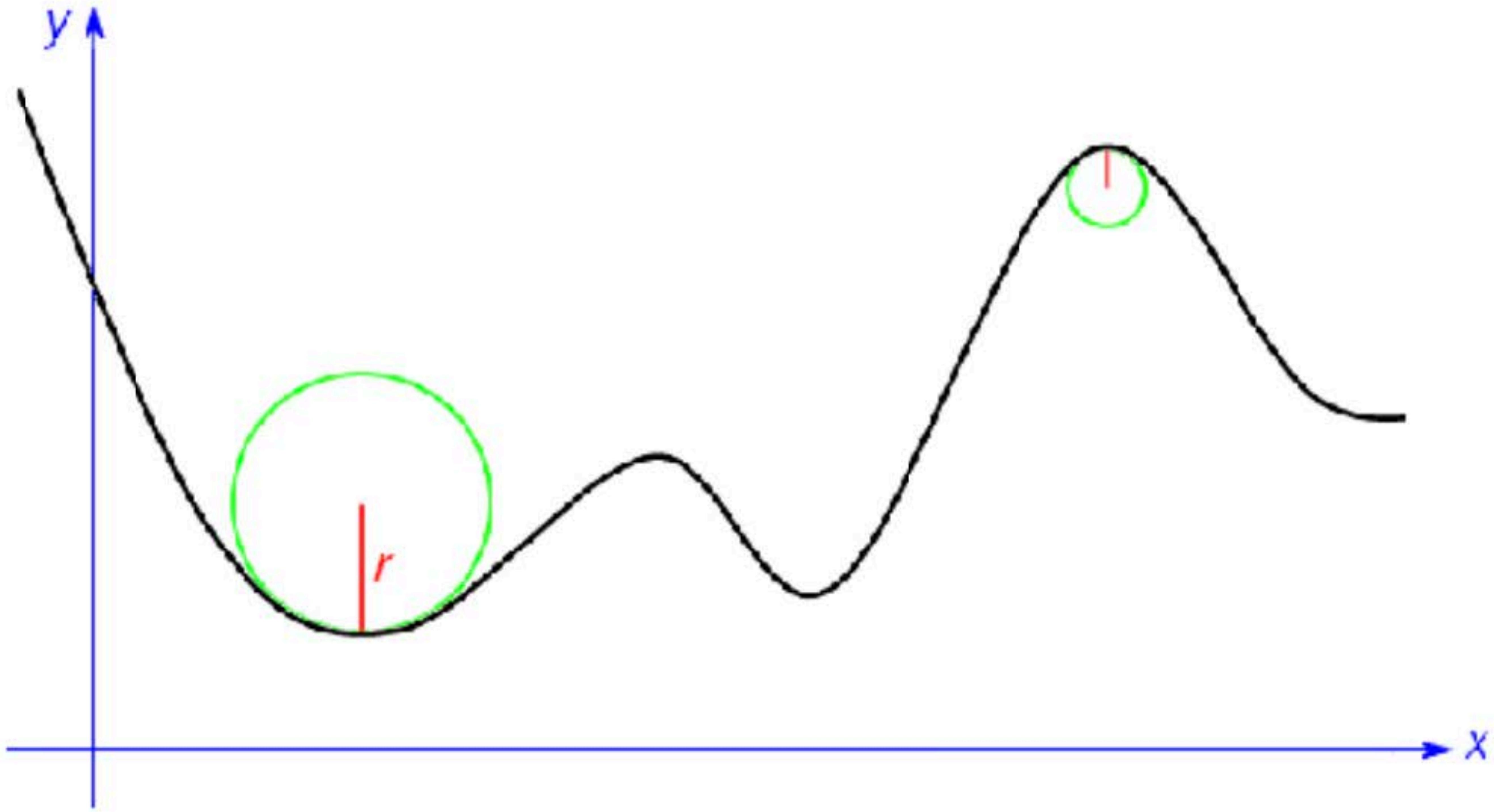
Tangent: $T(s) = \gamma'(s)$ $\longrightarrow$ velocity

Normal: $N(s) = \dfrac{T'(s)}{||T'(s)||}$ $\longrightarrow$ acceleration direction

Curvature: $\kappa(s) = ||T'(s)||$ $\longrightarrow$ acceleration magnitude



Radius of Curvature

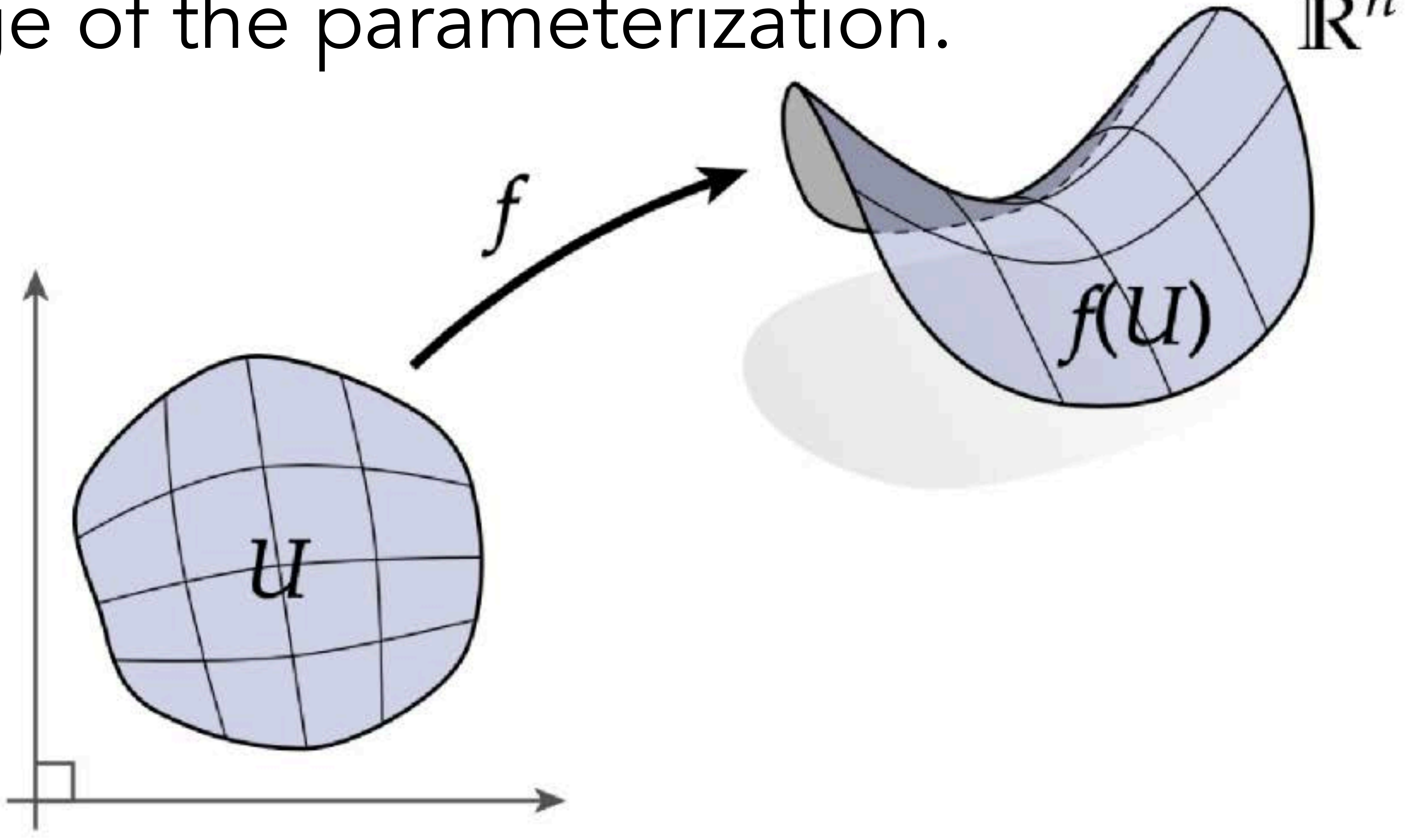


$$r(s) := \frac{1}{k(s)}$$

Parametrized Surface

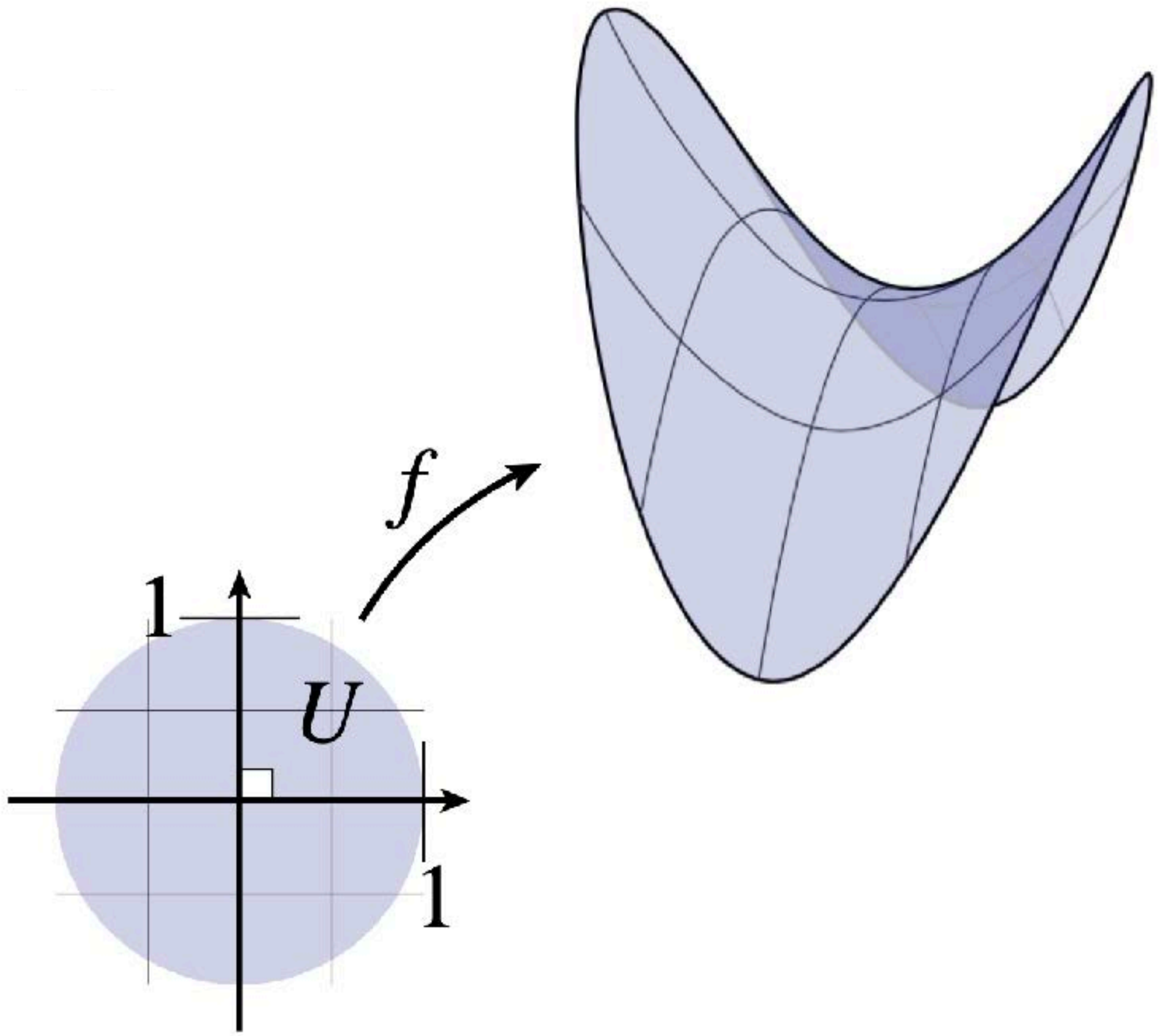
- A parameterized surface is a map f from a two-dimensional region U into $\mathbb{R}^n$
- $U \subset \mathbb{R}^2$ is the parameter domain
- The set of points $f(U)$ is called the image of the parameterization.

$$f : U \to \mathbb{R}^n$$

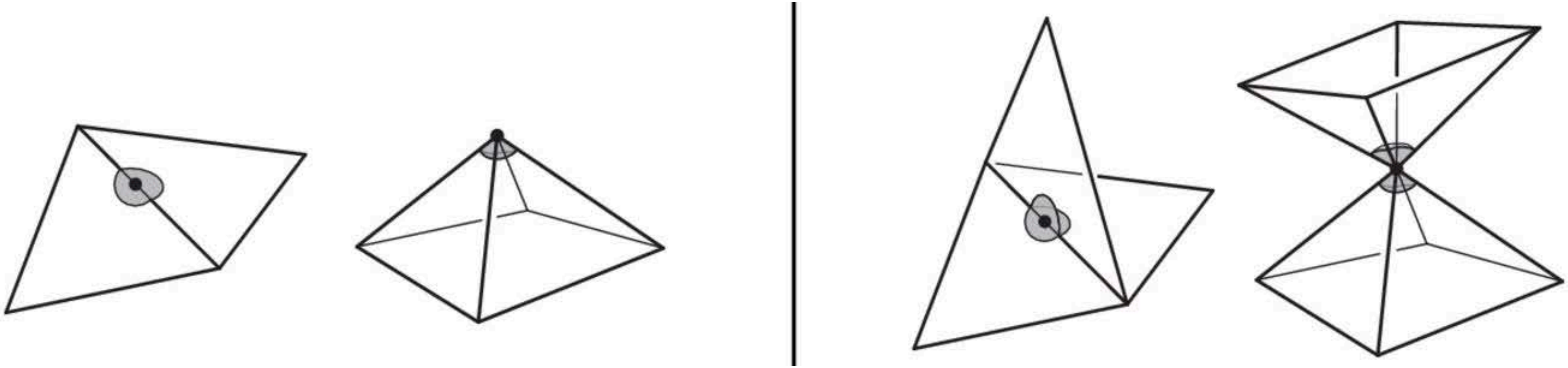


Example

- We can express a saddle as a parameterized surface:
  - $U := \{(u, v) \in \mathbb{R}^2 : u^2 + v^2 \leq 1\}$
  - $f(u, v) = [u, v, u^2 - v^2]^T$



Smoothness of surfaces

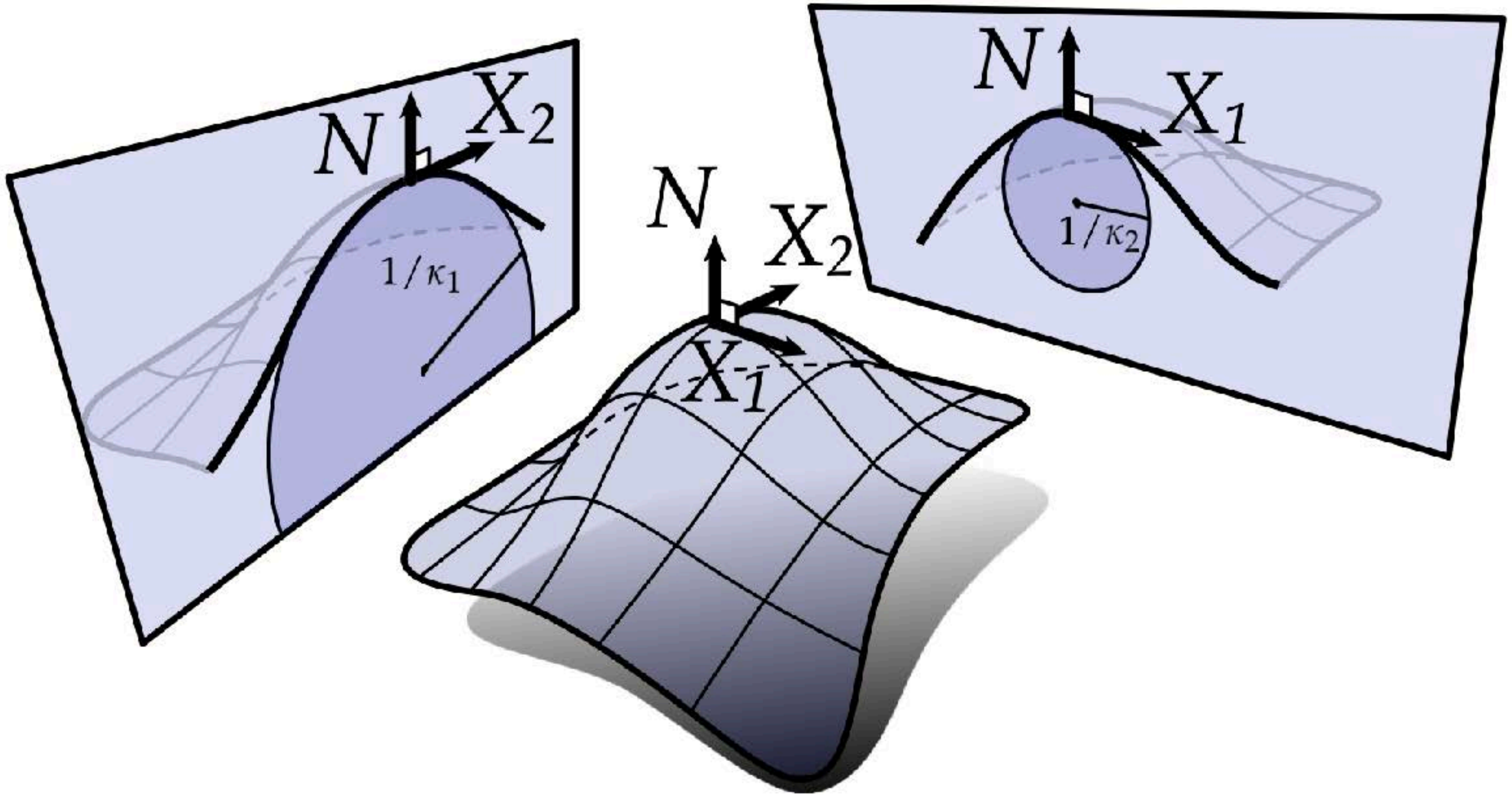


Slides credit: Ren Ng

Principal curvatures

Maximal curvature: $\kappa_1 = \kappa_{\max} = \max_{\varphi} \ \kappa_n(\varphi)$

Minimal curvature: $\kappa_2 = \kappa_{\min} = \min_{\varphi} \ \kappa_n(\varphi)$

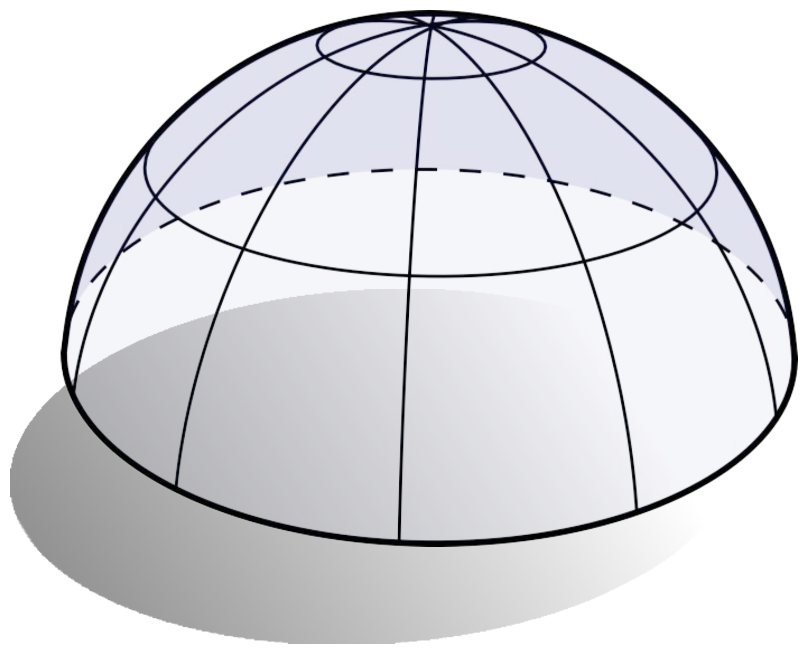


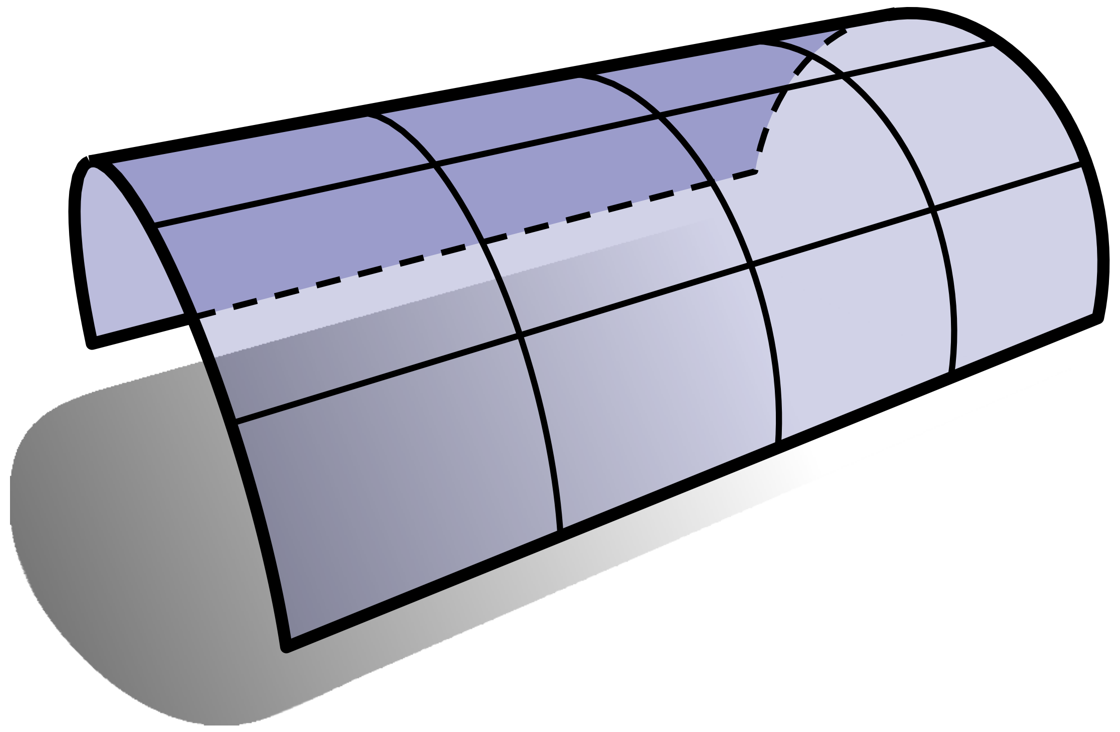
Gaussian and mean curvature

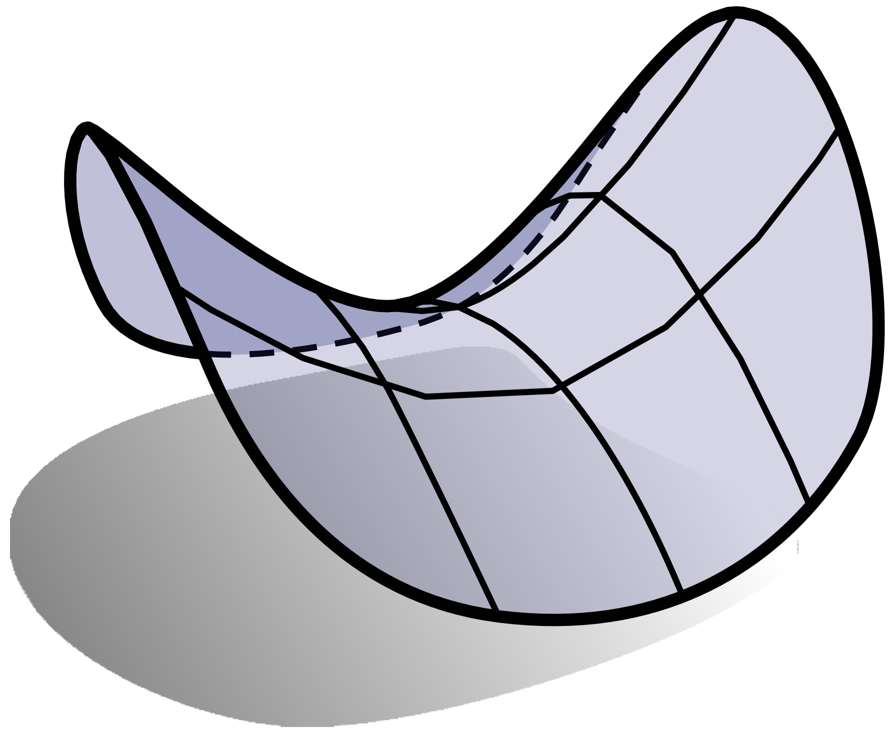
- Gaussian and mean curvature also fully describe local bending:

$$\text{Gaussian:} \quad K := \kappa_1 \kappa_2$$

$$\text{mean:} \quad H := \frac{1}{2}(\kappa_1 + \kappa_2)$$







$K > 0$ | *"developable"* $\boxed{K = 0}$ | $K < 0$

$H \neq 0$ | $H \neq 0$ | *"minimal"* $\boxed{H = 0}$

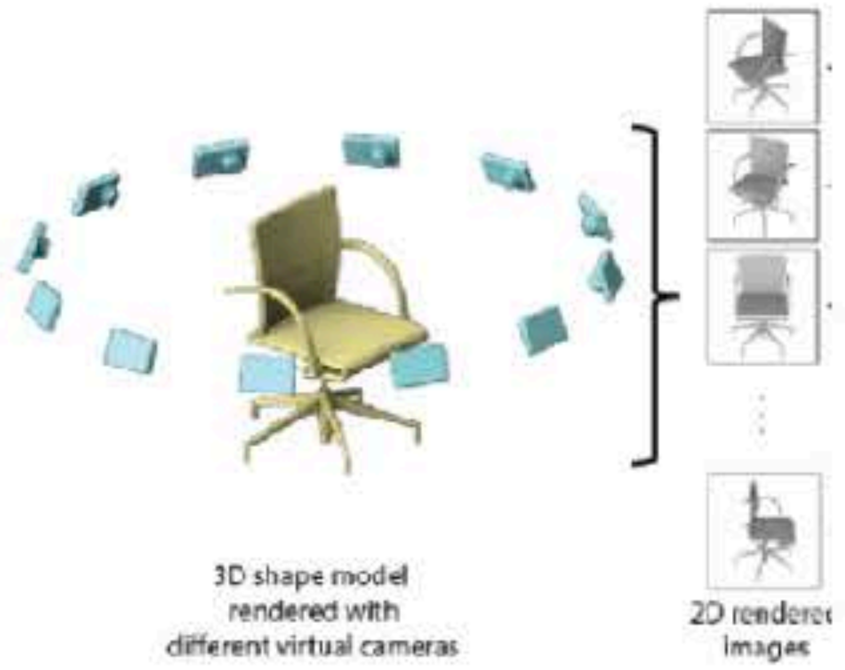
Slides credit: Hao Su

Summary

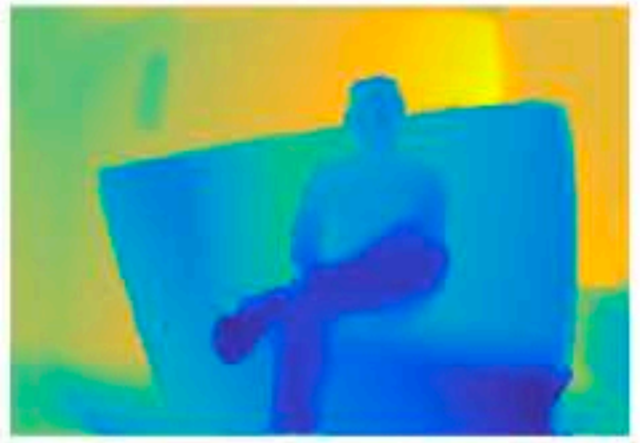
- We can use parametrized form to represent smooth curves and surfaces
- Such parametrized form allows us to define local properties such as tangent, normal and curvature
- We usually require "smoothness" for easier analysis.

Next time: 3D representations

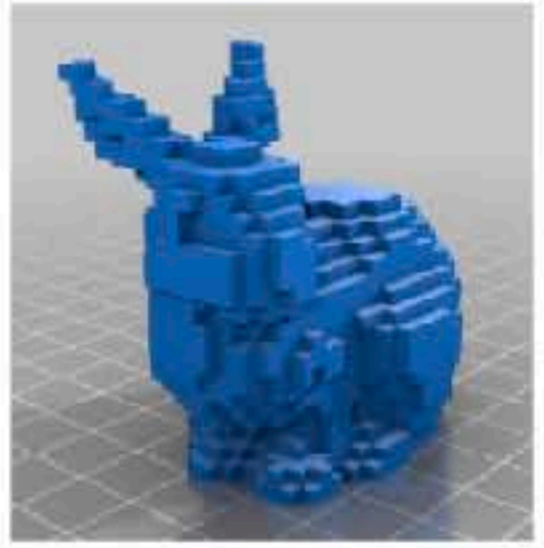
Rasterized form (regular grids)



Multi-view



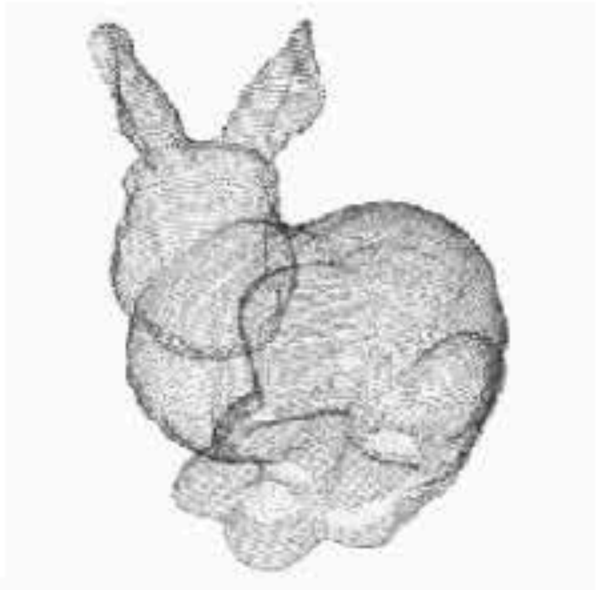
Depth Map



Volumetric

Geometric form (irregular)

Mesh



Point Cloud

$$F(x) = 0$$

Implicit Shape