UC San Diego

Lecture 7:

Deep Learning on Extrinsic Geometry

Instructor: Hao Su

Jan 30, 2018

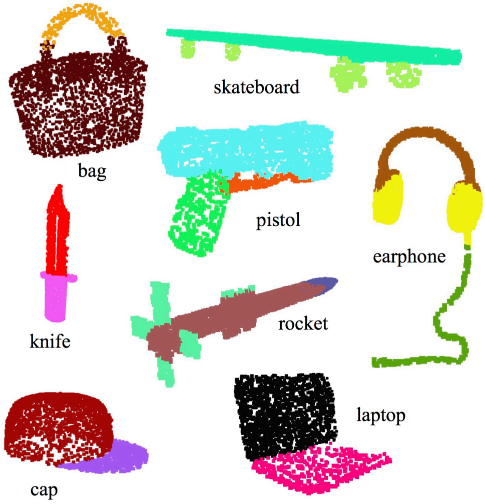
slides credits: Justin Solomon, Chengcheng Tang

3D deep learning tasks

3D geometry analysis

It is a chair!
:e

Classification

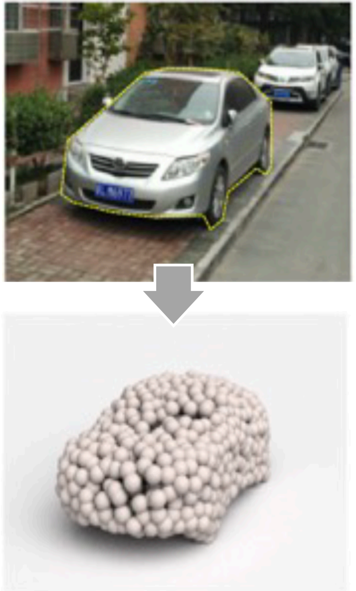


Parsing (object/scene)

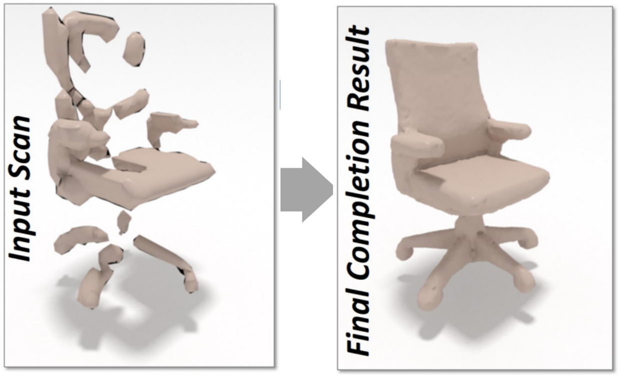
Correspondence

3D deep learning tasks

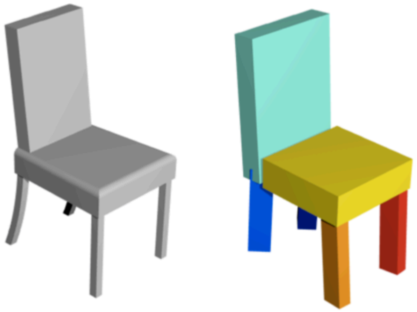
3D synthesis



Monocular
3D reconstruction

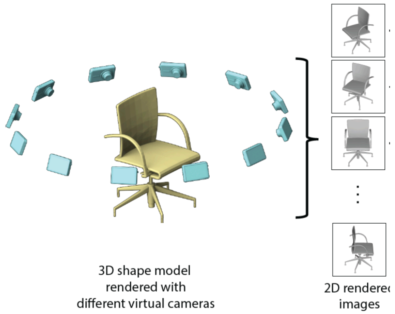


Shape completion



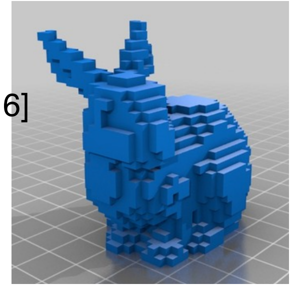
Shape modeling

3D deep learning algorithms (by representations)



[Su et al. 2015]
[Kalogerakis et al. 2016]
…

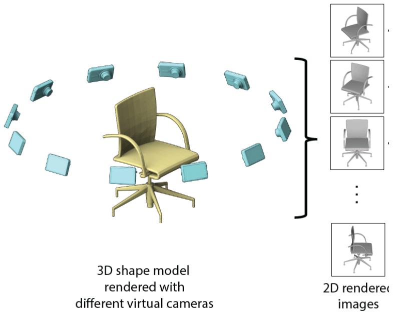
Multi-view



[Maturana et al. 2015]
[Wu et al. 2015] (GAN)
[Qi et al. 2016]
[Liu et al. 2016]
[Wang et al. 2017] (O-Net)
[Tatarchenko et al. 2017] (OGN)
…

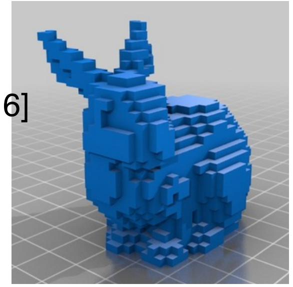
Volumetric

3D deep learning algorithms (by representations)



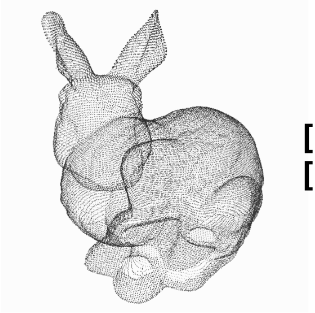
[Su et al. 2015]
[Kalogerakis et al. 2016]
…

Multi-view



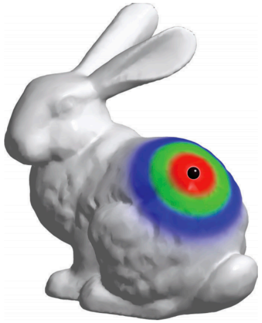
[Maturana et al. 2015]
[Wu et al. 2015] (GAN)
[Qi et al. 2016]
[Liu et al. 2016]
[Wang et al. 2017] (O-Net)
[Tatarchenko et al. 2017] (OGN)
…

Volumetric



[Qi et al. 2017] (PointN
[Fan et al. 2017] (Poin

Point cloud



[Defferard et al. 2016]
[Henaff et al. 2015]
[Yi et al. 2017] (SyncSpecC
…

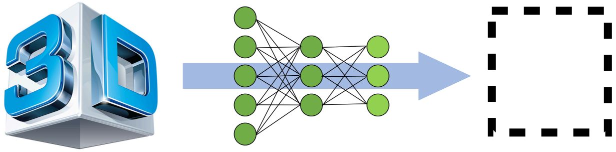
Mesh (Graph CNN)

[Tulsiani et al. 2017]
[Li et al. 2017] (GRASS)

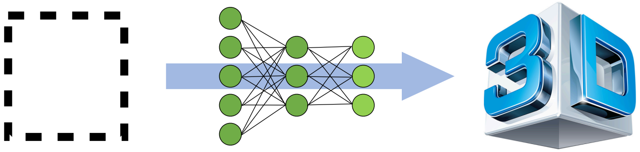
Part assembly

Cartesian product space of “task” and “representation”

3D geometry analysis



3D synthesis



Deep Learning on Point Cloud Data

Agenda

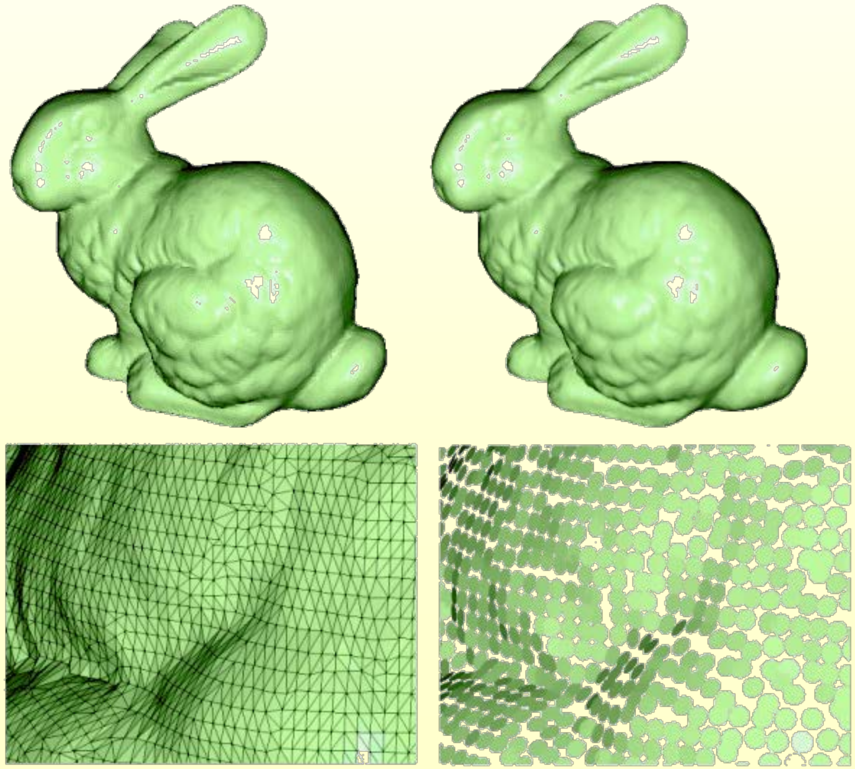
- Why point cloud?
- Comparison of point cloud
- Point cloud generation by deep learning

Agenda

- **Why point cloud?**
- Comparison of point cloud
- Point cloud generation by deep learning

Point Clouds

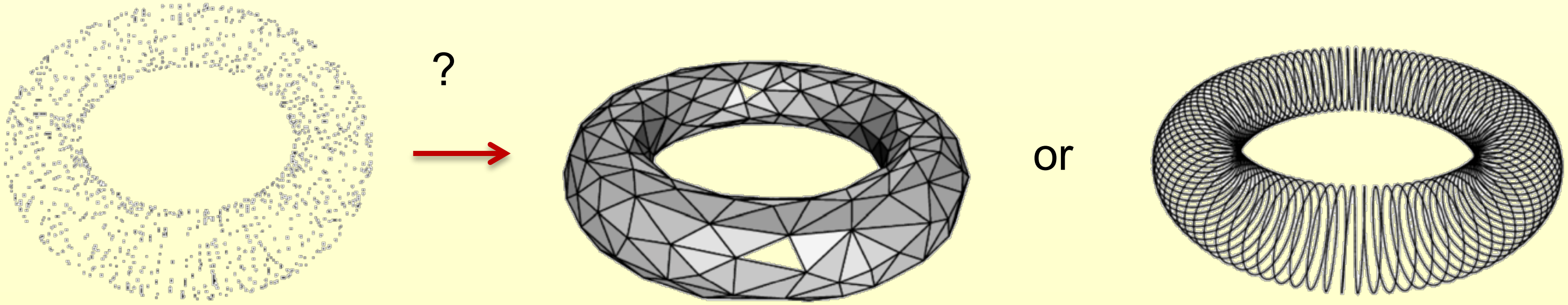
- Simplest representation: **only points,** no connectivity
- Collection of (x,y,z) coordinates, possibly with normals
- Points with orientation are called **surfels**



Filip van Bouwel

Point Clouds

- Simplest representation: **only points,** no connectivity
- Collection of (x,y,z) coordinates, possibly with normals
- Points with orientation are called **surfels**
- Severe limitations:
  - **no** simplification or subdivision
  - **no** direct smooth rendering
  - **no** topological information

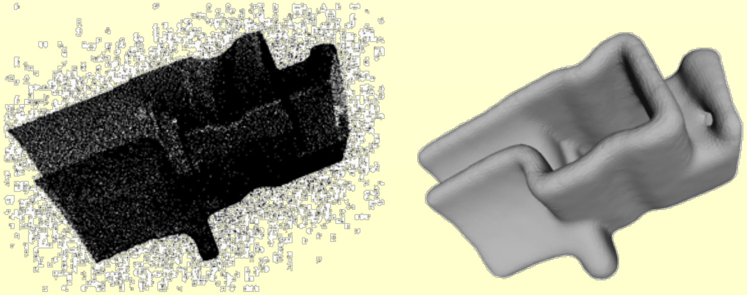


Point Clouds

- Simplest representation: **only points,** no connectivity
- Collection of (x,y,z) coordinates, possibly with normals
- Points with orientation are called **surfels**
- Severe limitations:
  - **no** simplification or subdivision
  - **no** direct smooth rendering
  - **no** topological information
  - weak approximation power: $O(h)$ for point clouds
    - need *square* number of points for the same approximation power as meshes

Point Clouds

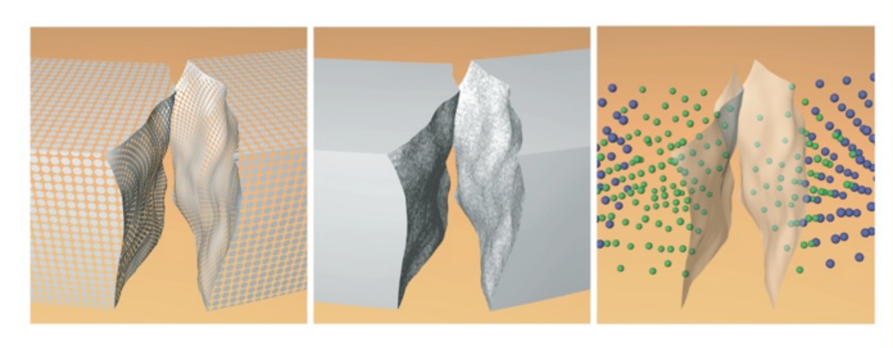
- Simplest representation: **only points,** no connectivity
- Collection of (x,y,z) coordinates, possibly with normals
- Points with orientation are called **surfels**
- Severe limitations:
  - **no** Simplification or subdivision
  - **no** direct smooth rendering
  - **no** topological information
  - weak approximation power
  - noise and outliers



Why Point Clouds?

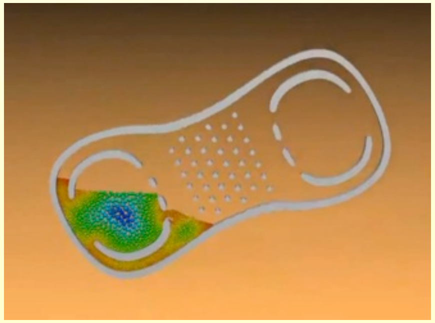
1) Typically, that's the only thing that's available
2) Isolation: sometimes, easier to handle (esp. in hardware).

Fracturing Solids



Meshless Animation of Fracturing Solids
Pauly et al., SIGGRAPH '05

Fluids



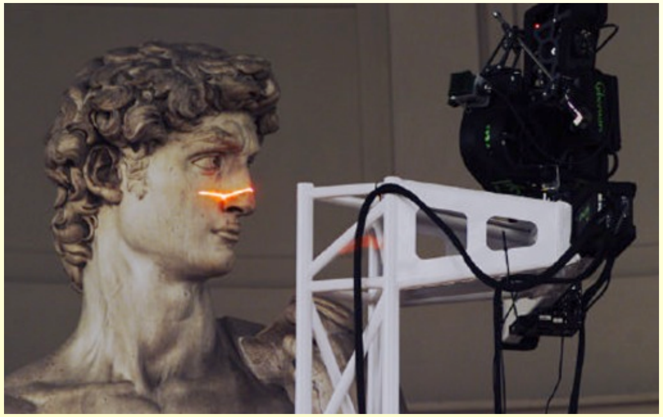
Adaptively sampled particle fluids,
Adams et al. SIGGRAPH '07

Why Point Clouds?

- Typically, that’s the only thing that’s available

  Nearly all 3D scanning devices produce point clouds

XBOX 360
KINECT
for XBOX 360



Agenda

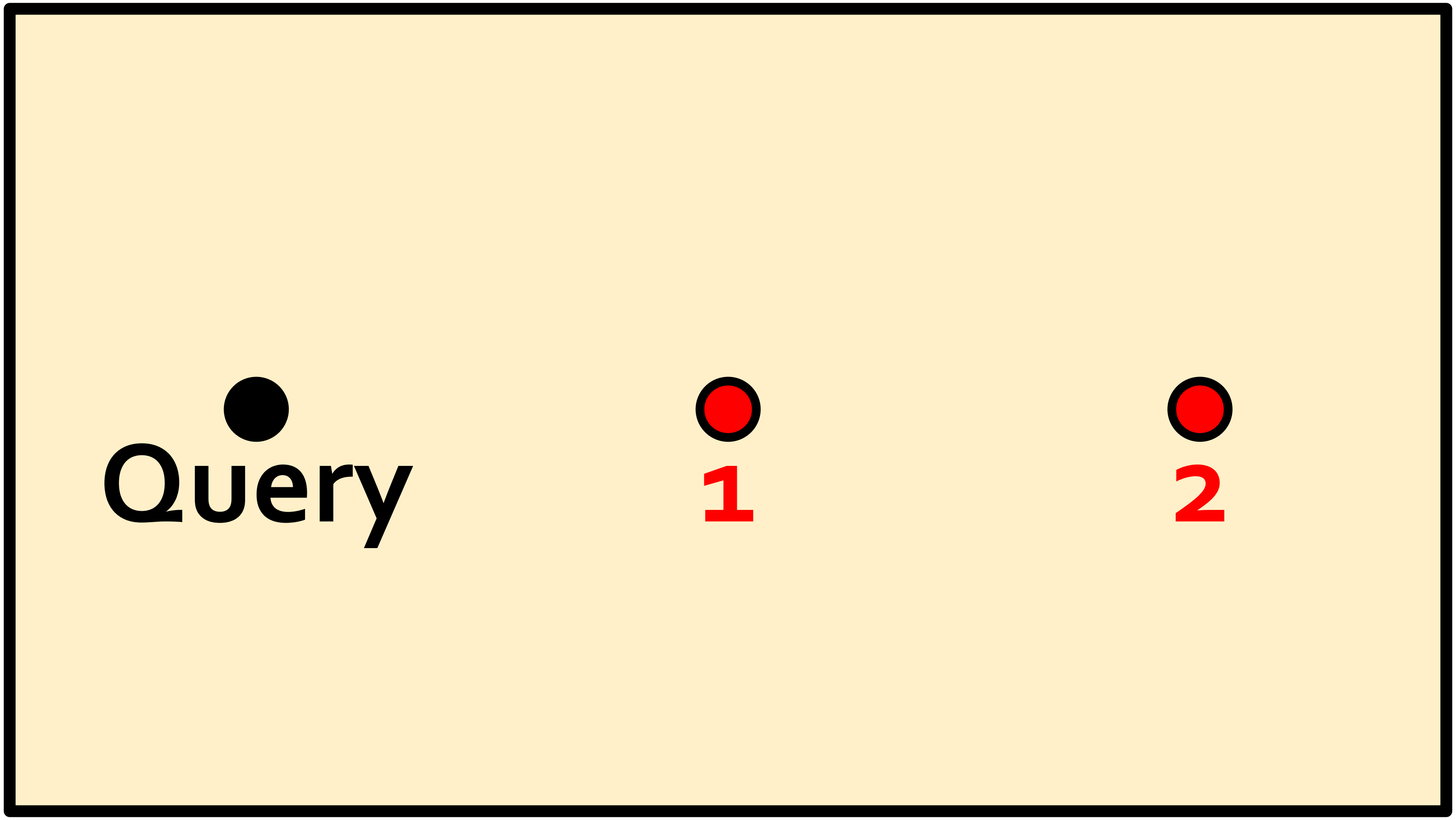
- Why point cloud?
- **Comparison of point cloud**
- Point cloud generation by deep learning

Point cloud as samples

- Point cloud can be thought as a representation of prob. distribution

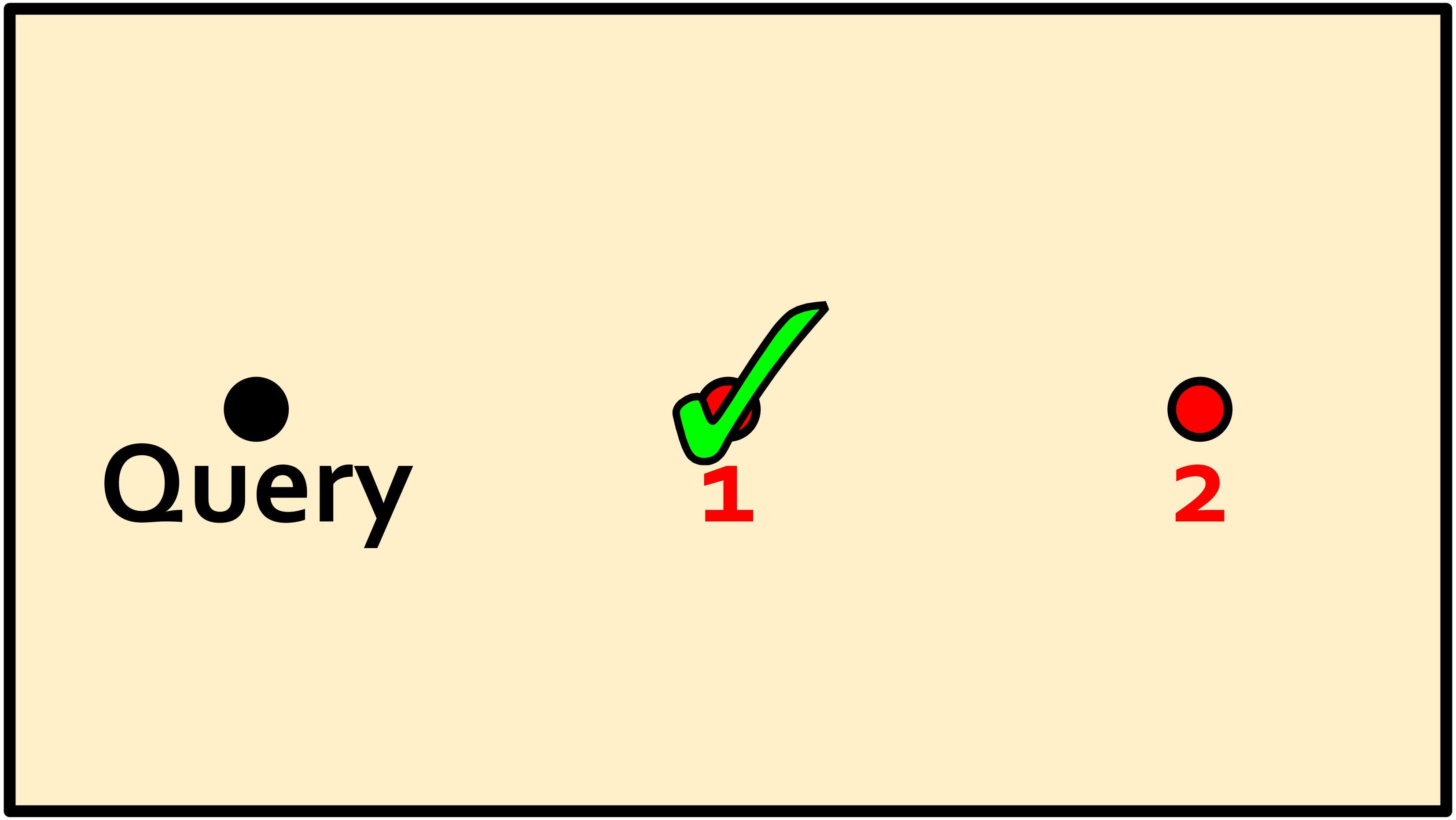
- Compare point cloud is to compare underlying distributions

Motivating Question



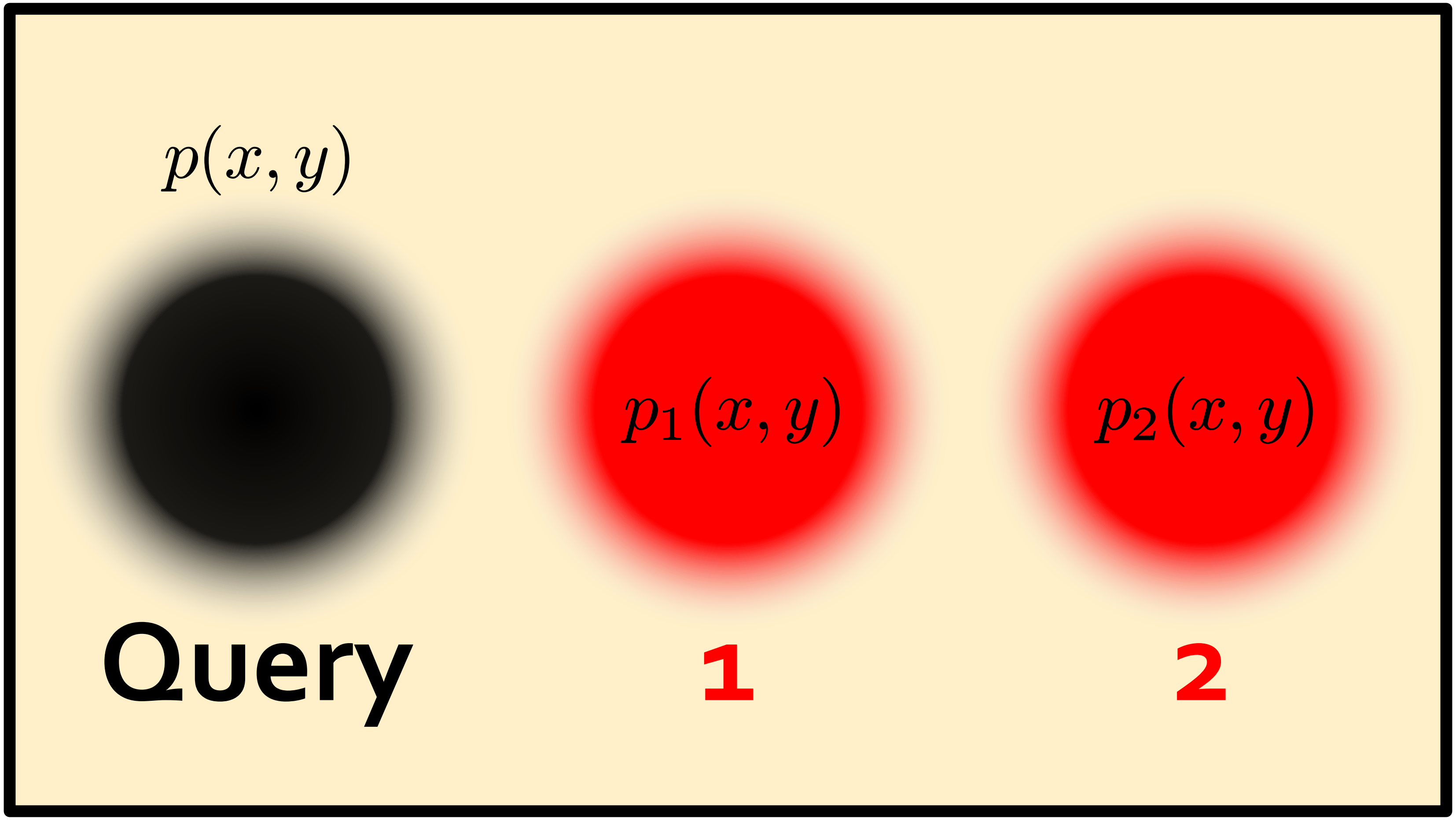
Which is closer, 1 or 2?

Motivating Question



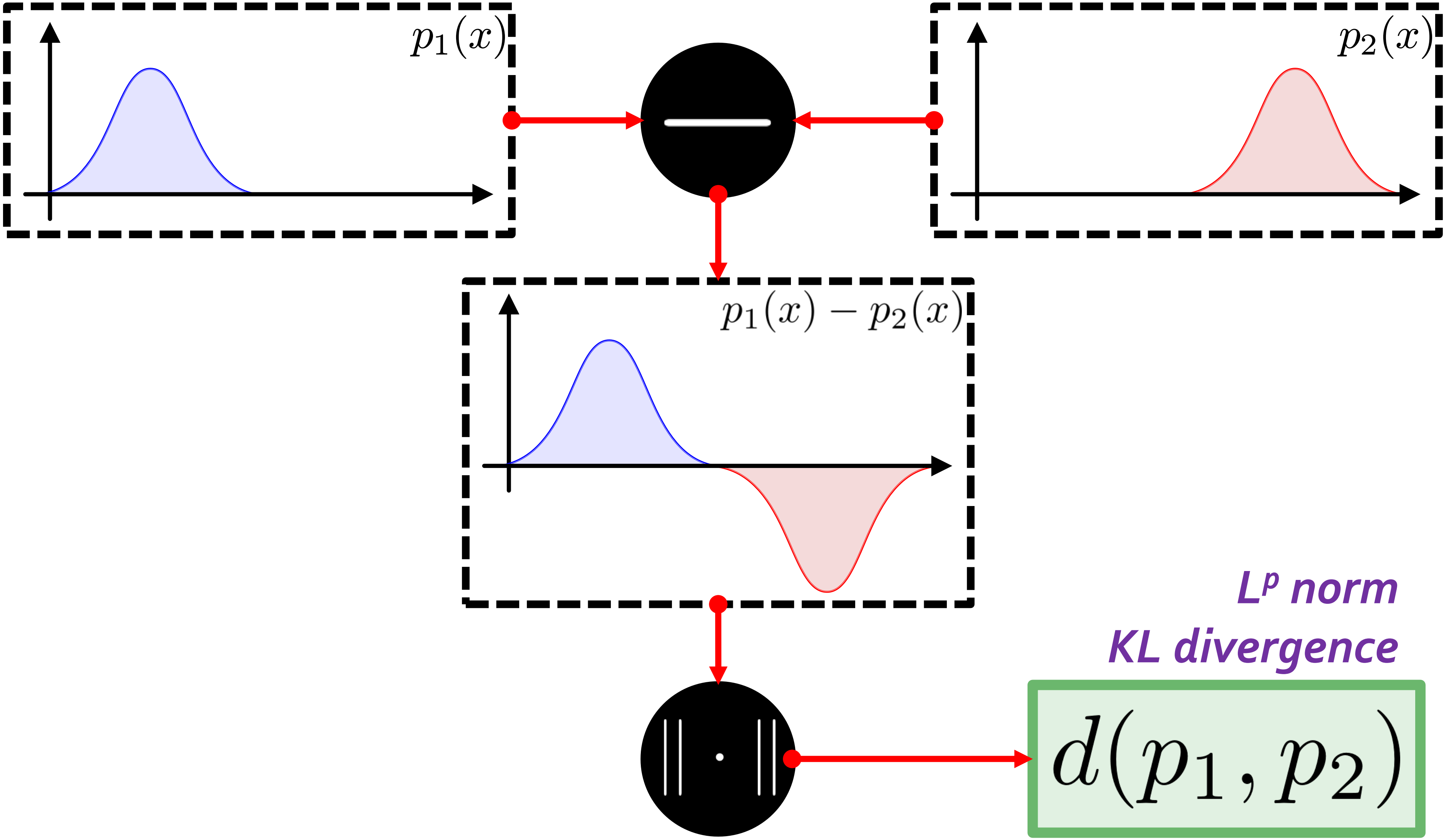
Which is closer, 1 or 2?

Fuzzy Version



Which is closer, 1 or 2?

Typical Measurement

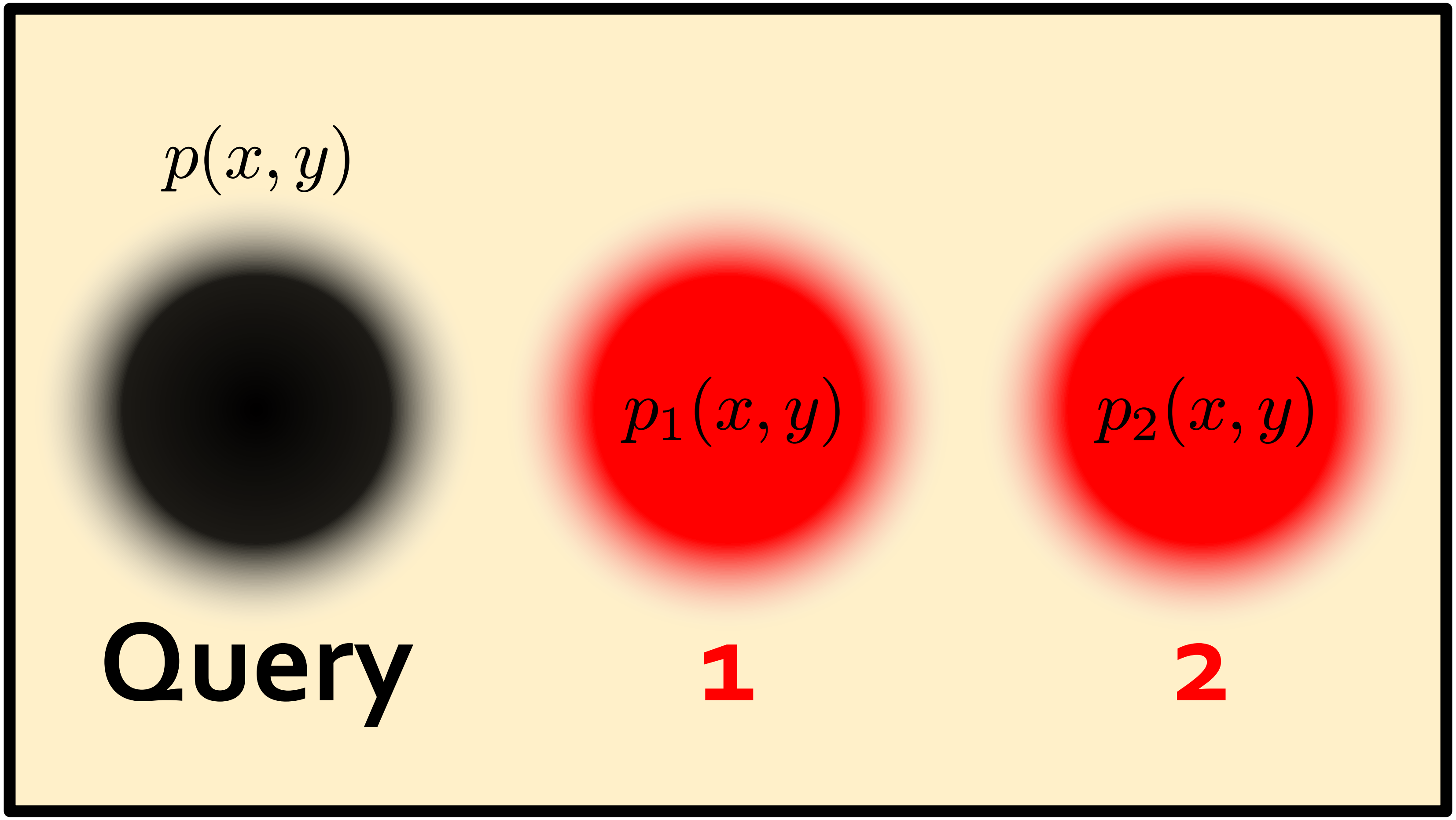


L^p norm

KL divergence

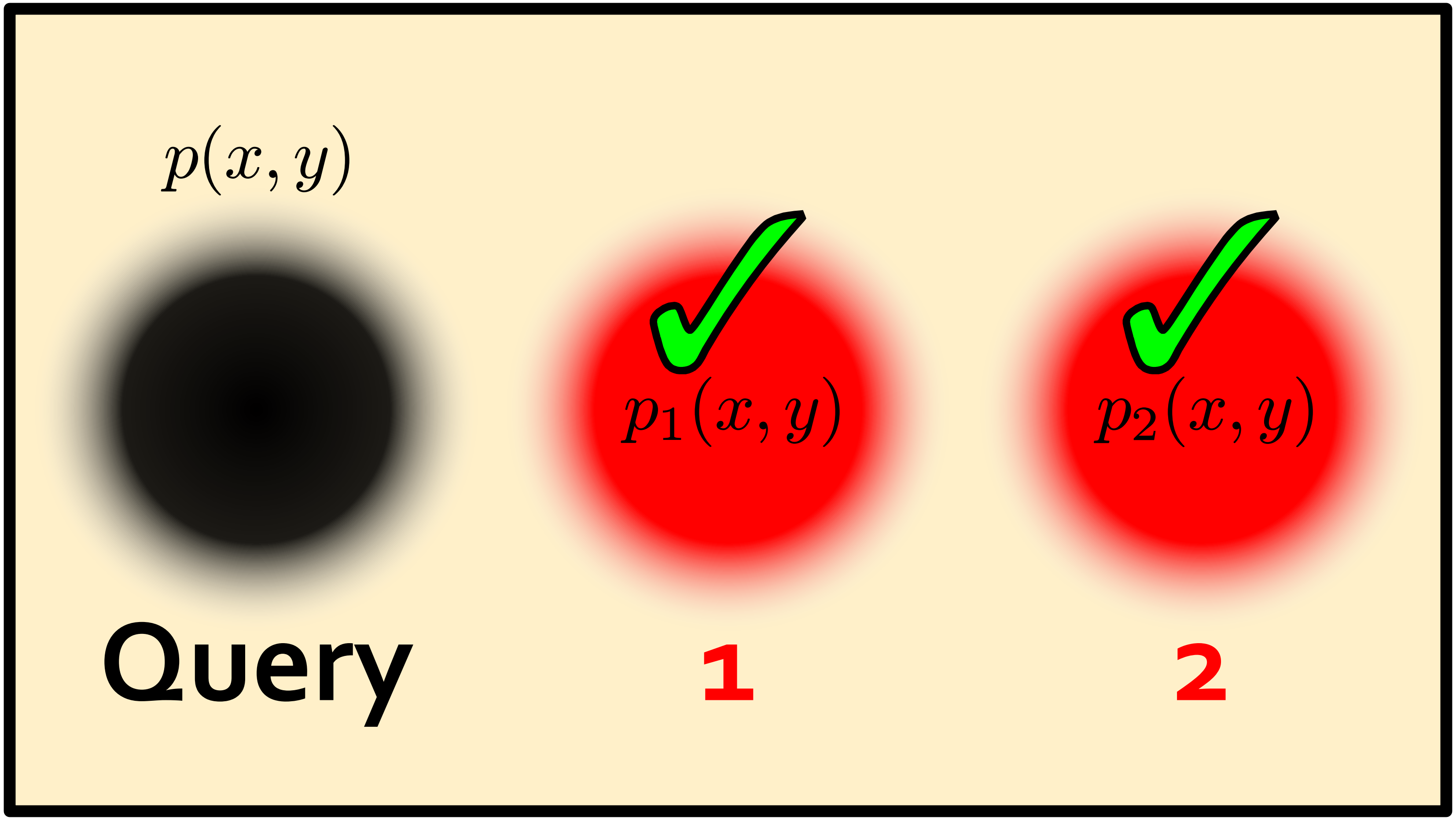
$$d(p_1, p_2)$$

Returning to the Question



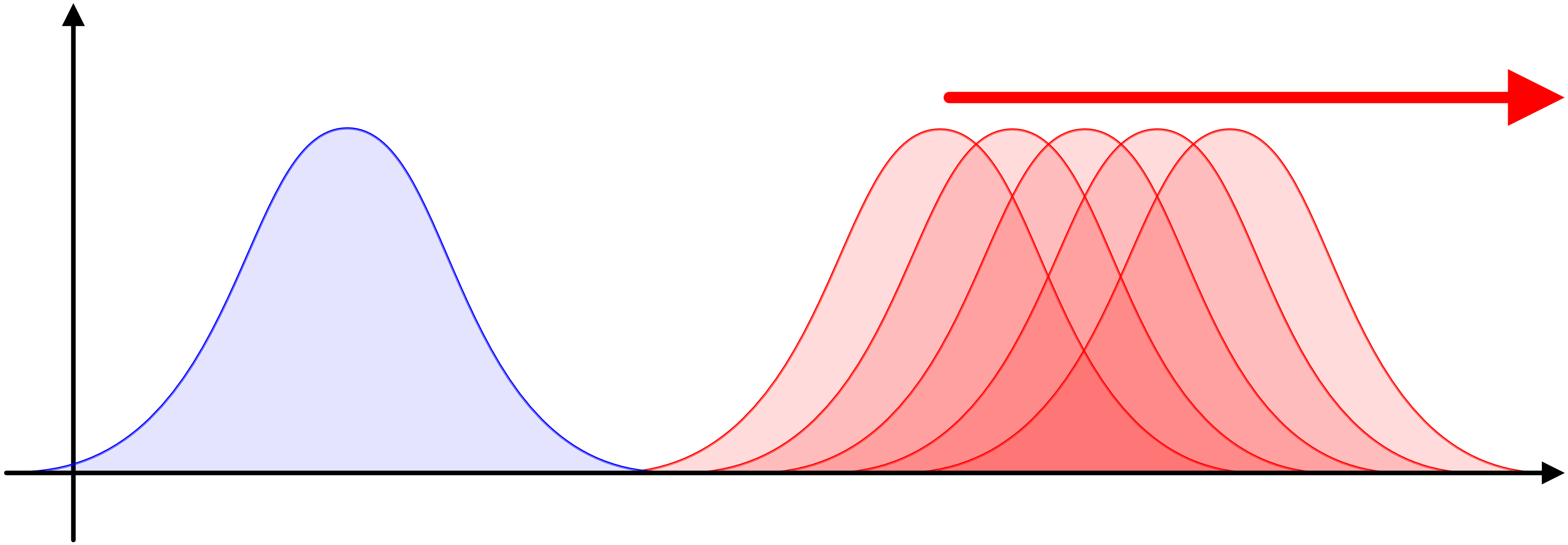
Which is closer, 1 or 2?

Returning to the Question



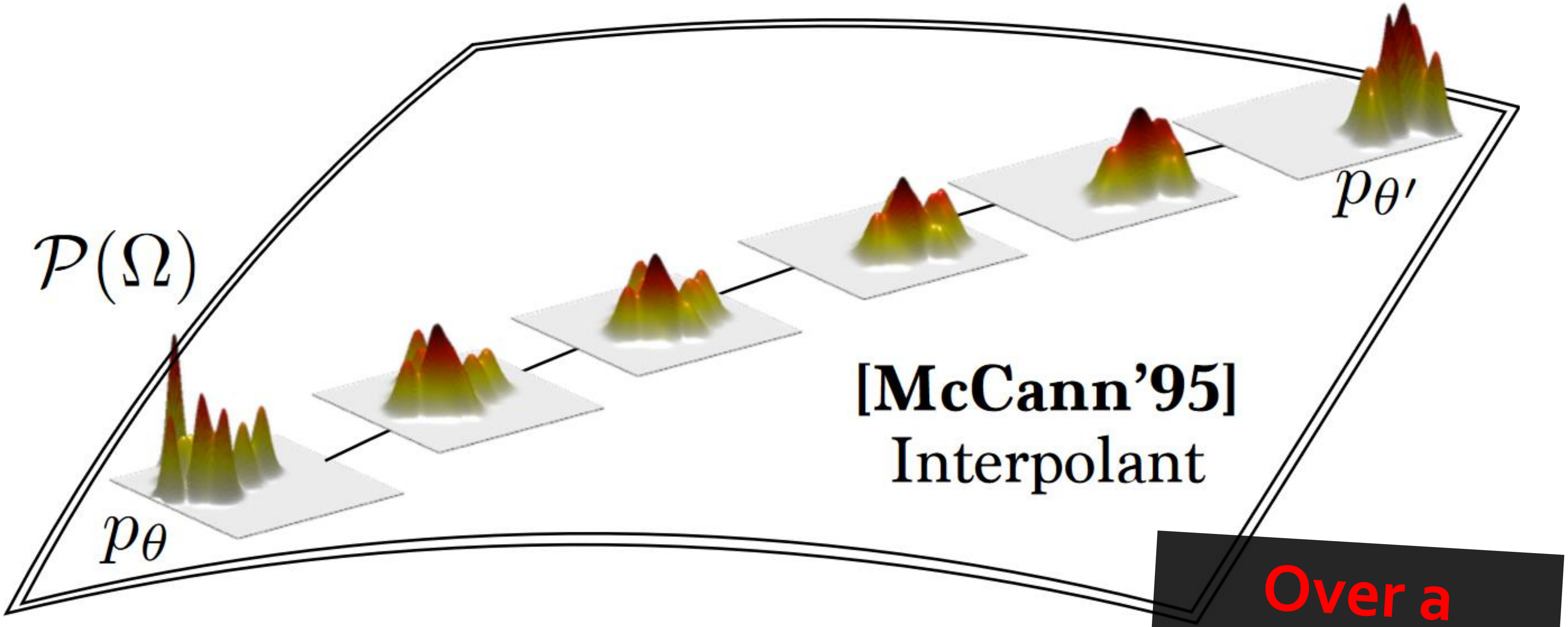
Neither! Equidistant.

What's Wrong?



Measured overlap, not displacement.

Optimal Transport

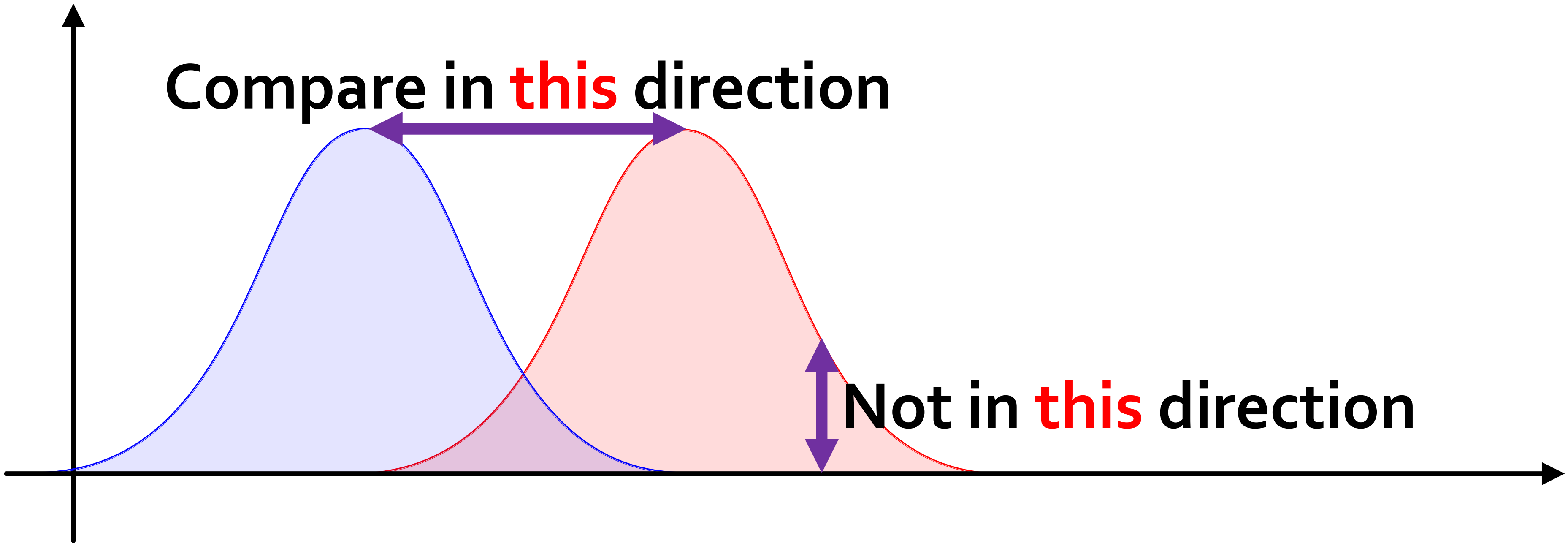


Over a metric space

Image courtesy M. Cuturi

Geometric theory of probability

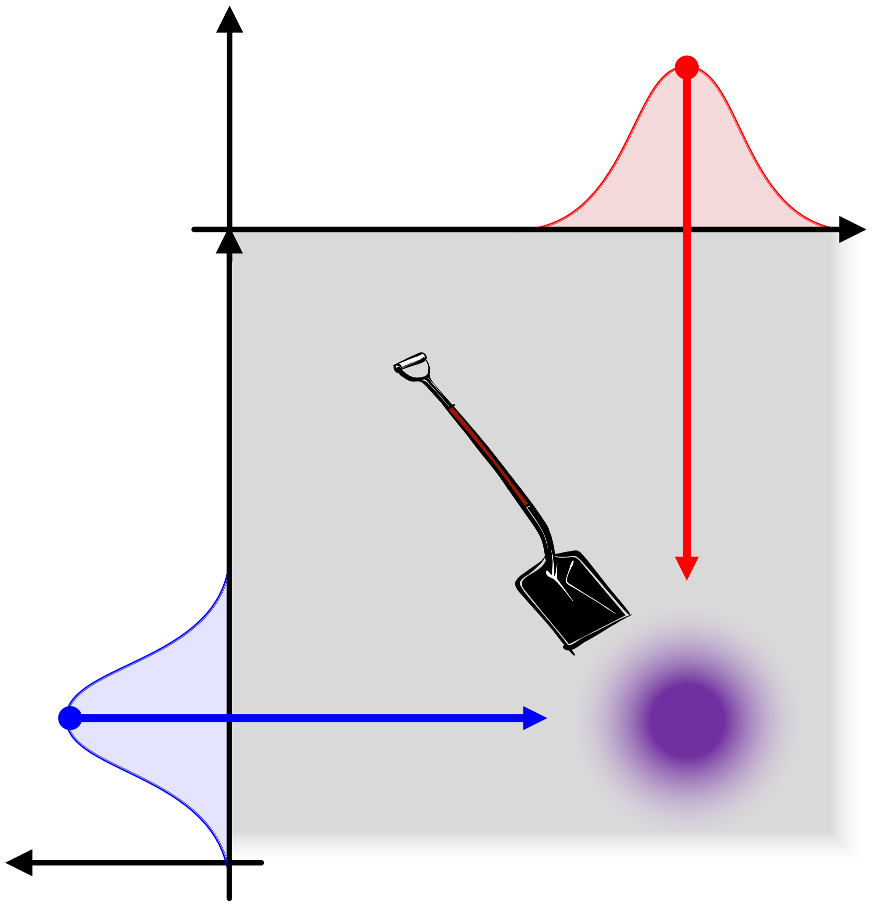
Alternative Idea



Compare in **this** direction

Not in **this** direction

Alternative Idea



Match mass from the distributions

Transportation Matrix

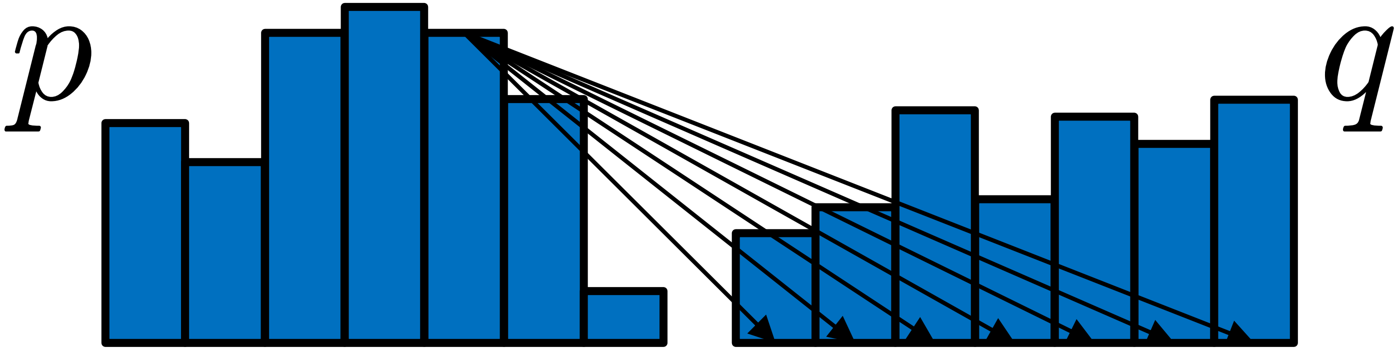
- Supply distribution $\boldsymbol{p_0}$
- Demand distribution $\boldsymbol{p_1}$

$$T \geq 0$$

$$T\mathbf{1} = p_0$$

$$T^\top \mathbf{1} = p_1$$

Earth Mover's Distance



$$\begin{aligned} \min_T \textstyle\sum_{ij} T_{ij} d(x_i, x_j) \\ \text{s.t.} \textstyle\sum_j T_{ij} = p_i \\ \textstyle\sum_i T_{ij} = q_j \\ T \geq 0 \end{aligned}$$

$\boldsymbol{m \cdot d(x, y)}$

Starts at $\boldsymbol{p}$

Ends at $\boldsymbol{q}$

Positive mass

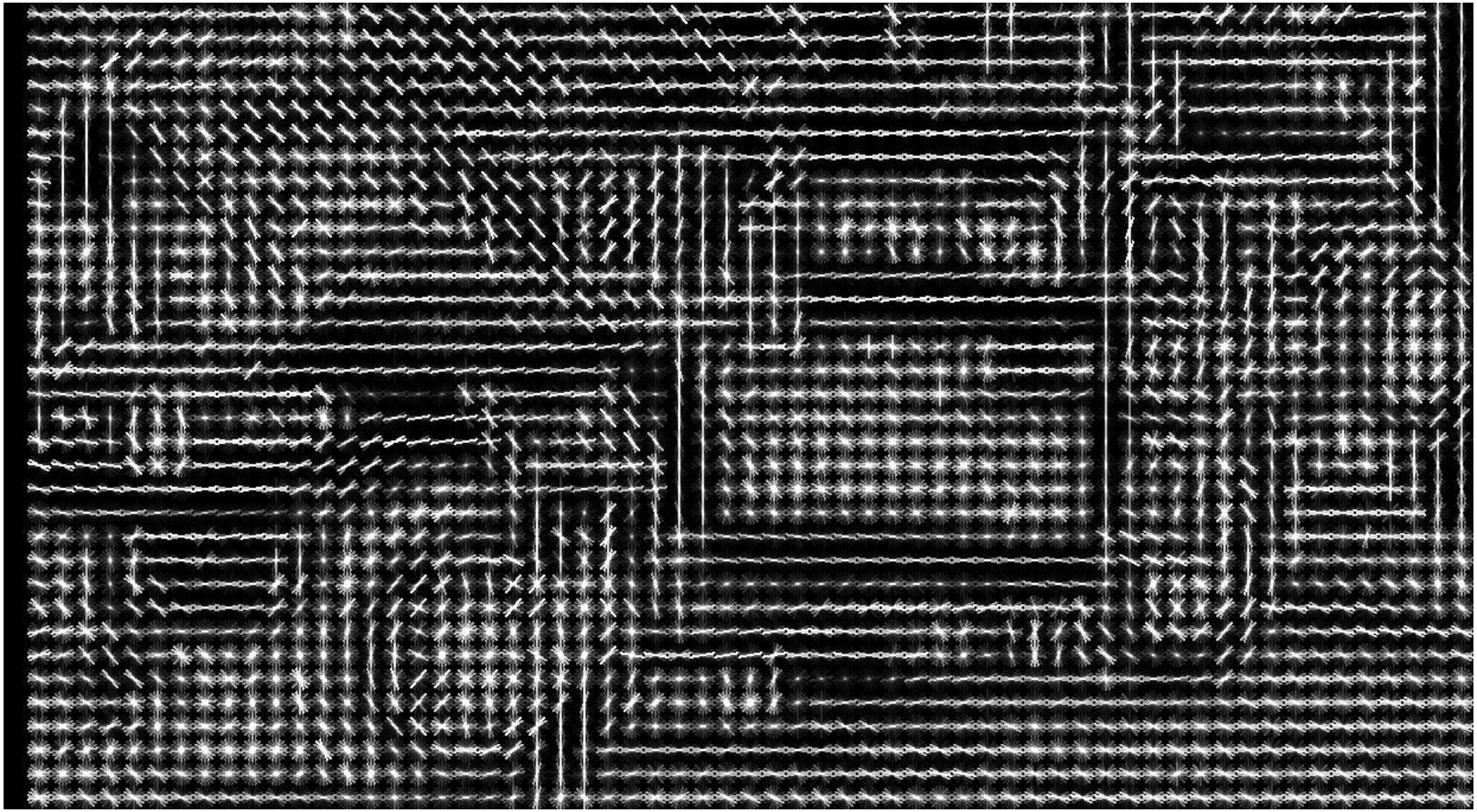
Important Theorem

EMD is a metric when d(x,y) satisfies the triangle inequality.

"The Earth Mover's Distance as a Metric for Image Retrieval"
Rubner, Tomasi, and Guibas; IJCV 40.2 (2000): 99—121.

Revised in:
"Ground Metric Learning"
Cuturi and Avis; JMLR 15 (2014)

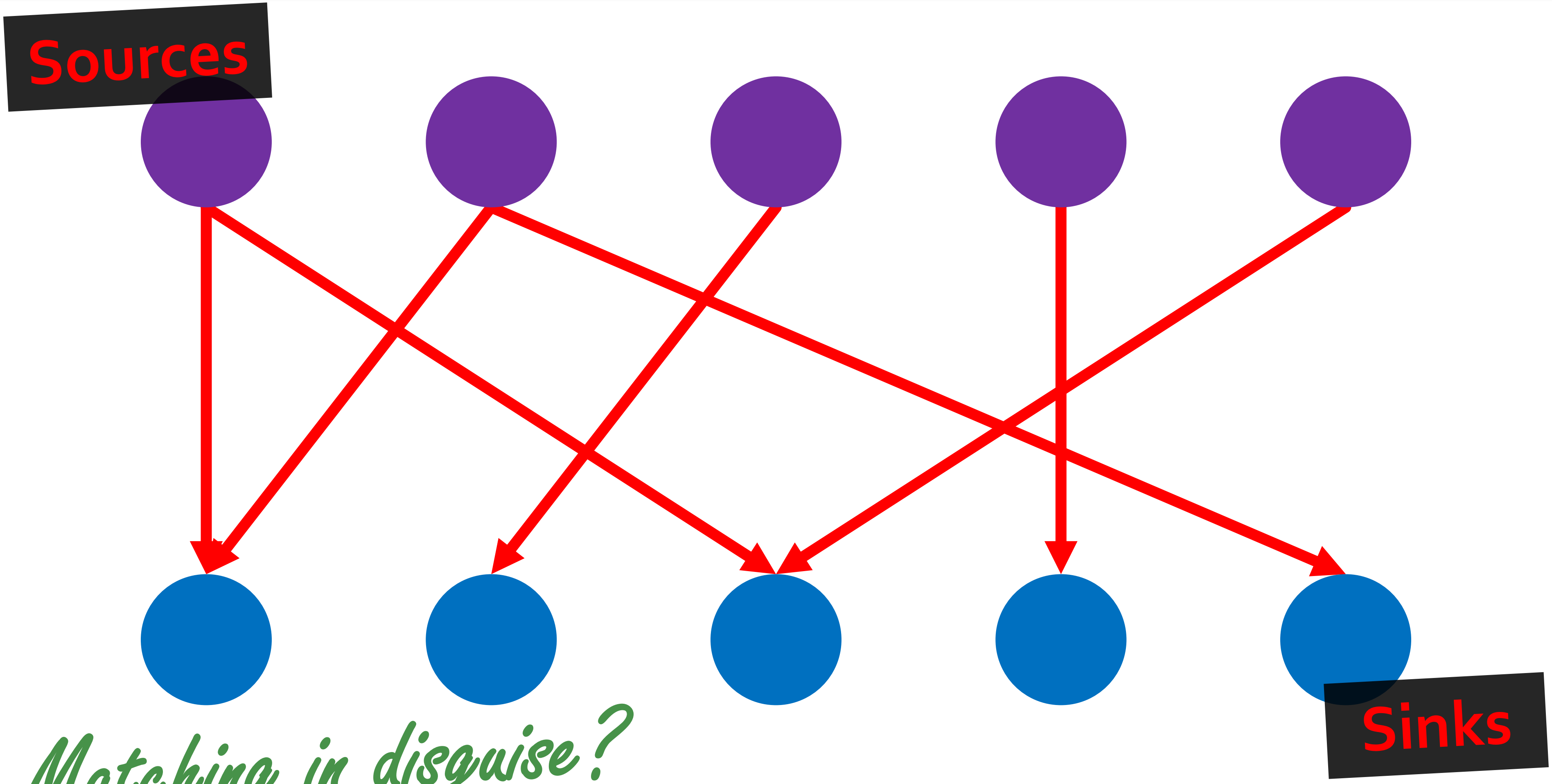
Basic Application



http://web.mit.edu/vondrick/ihog/

Comparing histogram descriptors

Discrete Perspective



Matching in disguise?

Min-cost flow

Algorithm for Small-Scale Problems

- **Step 1:** Compute D_{ij}
- **Step 2:** Solve linear program
  - Simplex
  - Interior point
  - Hungarian algorithm
  - …

Transportation Matrix Structure

	🟥			
		🟥		
🟥				
				🟥
			🟥	

Matches bins

Underlying map!

p-Wasserstein Distance

$$\mathcal{W}_p(\mu,\nu) \equiv \min_{\pi\in\Pi(\mu,\nu)} \left(\iint_{X\times X} d(x,y)^p \, d\pi(x,y)\right)^{1/p}$$

Shortest path distance

Expectation

General cost:
"Monge-Kantorovich problem"

Geodesic distance d(x,y)

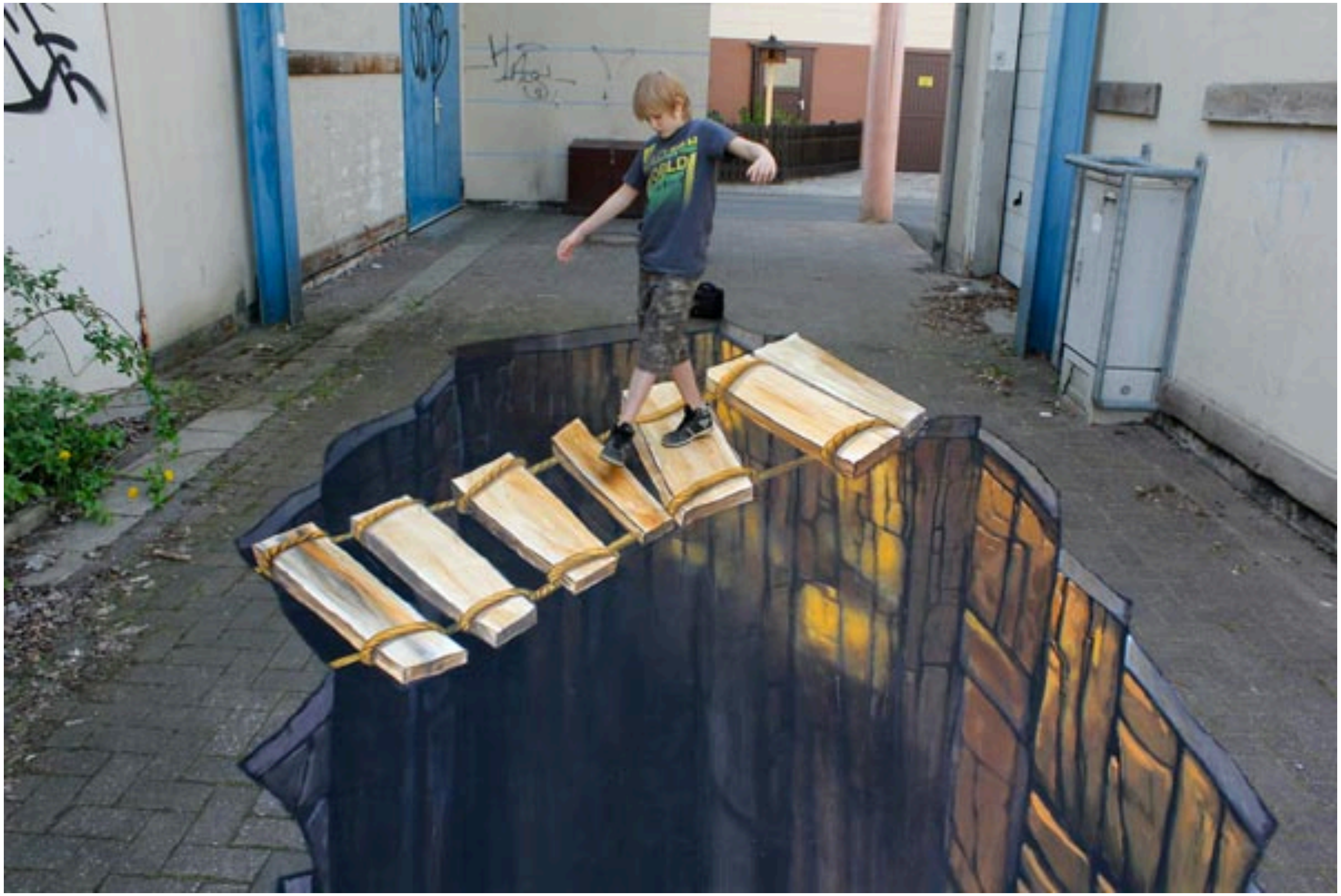
http://www.sciencedirect.com/science/article/pii/S152407031200029X#

Continuous analog of EMD

Agenda

- Why point cloud?
- Comparison of point cloud
- **Point cloud generation by deep learning**

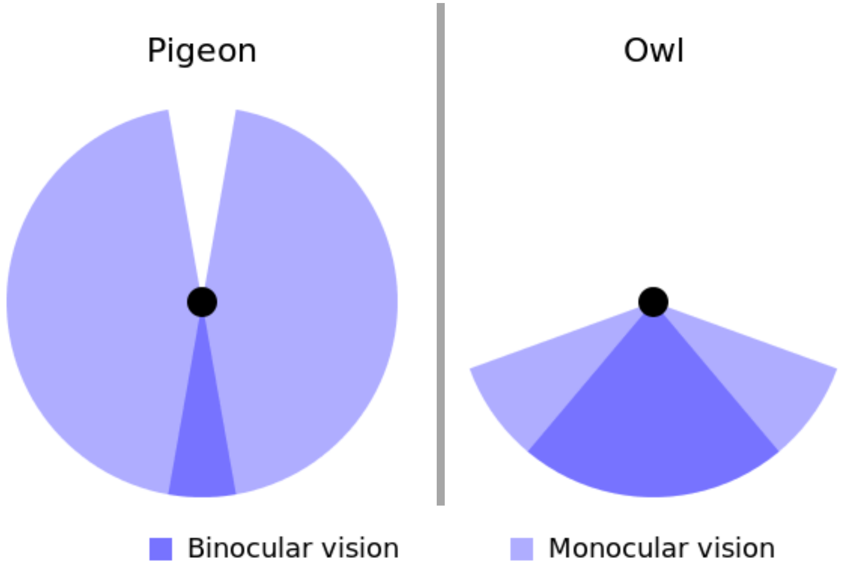
3D perception from a single image



Monocular vision

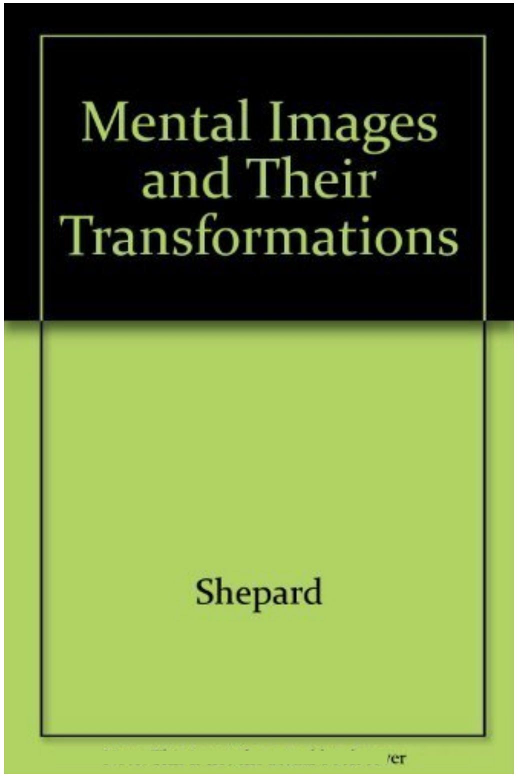
a typical prey

a typical predator



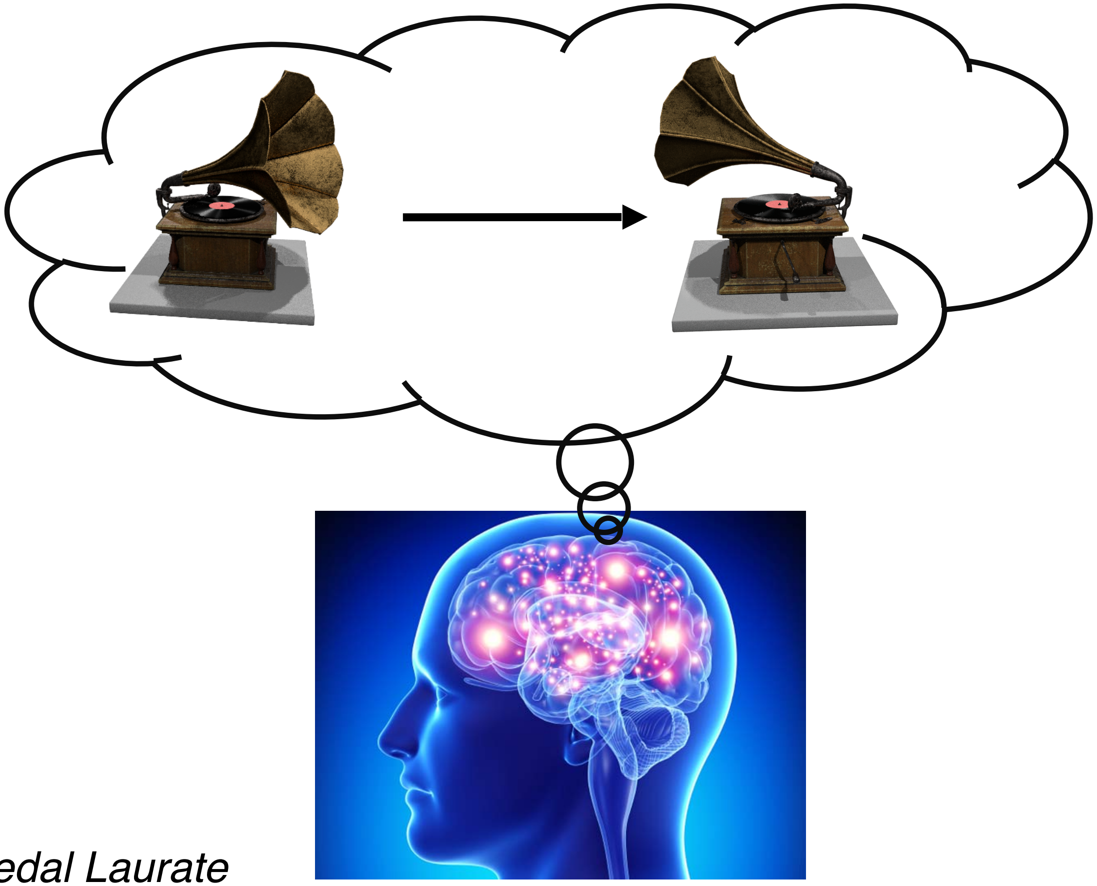
Cited from https://en.wikipedia.org/wiki/Binocular_vision

A psychological evidence – mental rotation



Mental Images and Their Transformations

Shepard



by Roger N. Shepard, National Science Medal Laurate and Lynn Cooper, Professor at Columbia University

Visual cues are complicated

contrast

color

texture

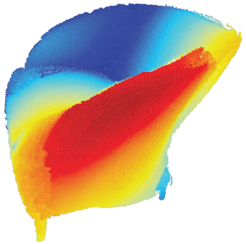
motion

symmetry

part

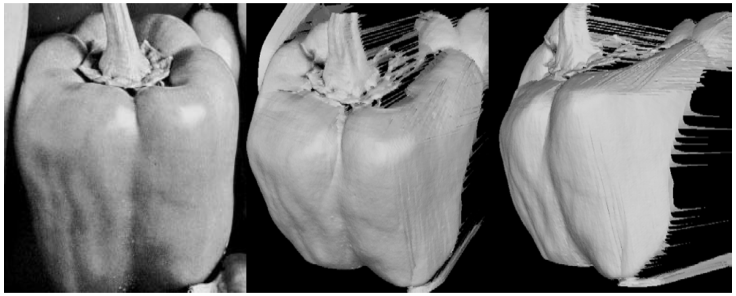
category-specific 3D knowledge

……

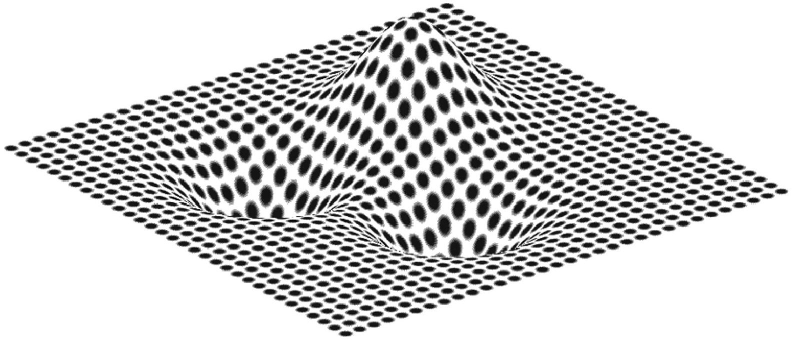


Status review of monocular vision algorithms

- Shape from X (texture, shading, …)



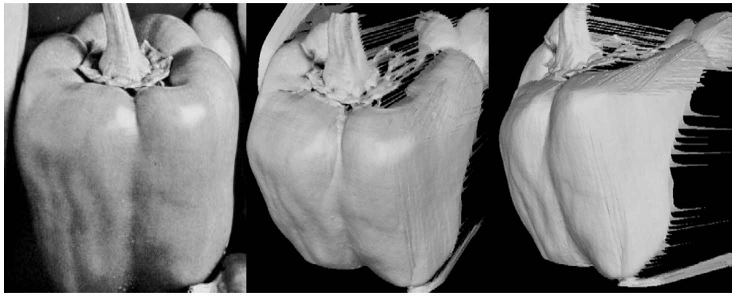
[Horn, 1989]



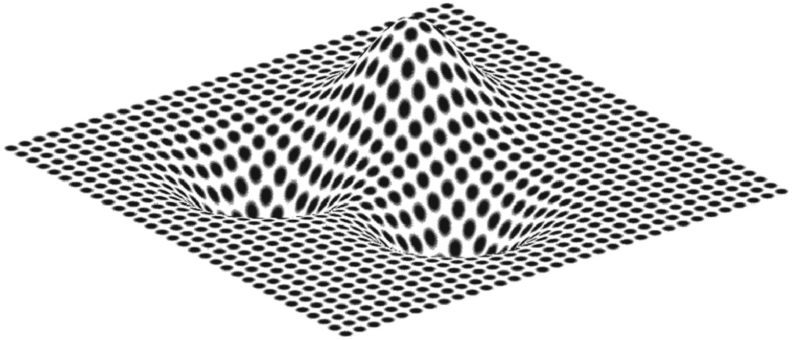
[Kender, 1979]

Status review of monocular vision algorithms

- Shape from X (texture, shading, …)

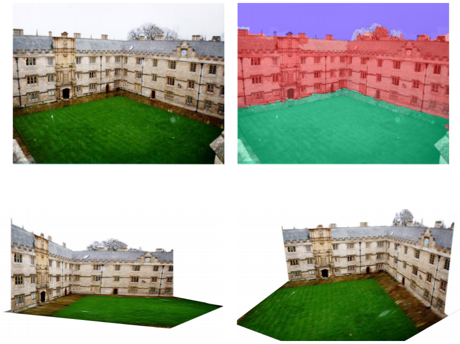


[Horn, 1989]



[Kender, 1979]

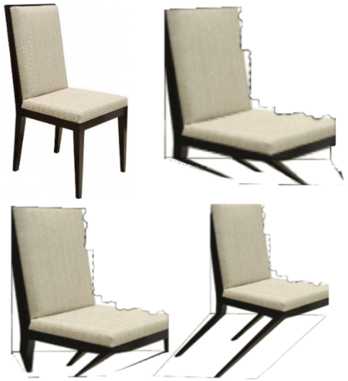
- Learning-based (from small data)



Hoiem et al, ICCV'05
Saxena et al, NIPS'05
…

🙂

- large planes

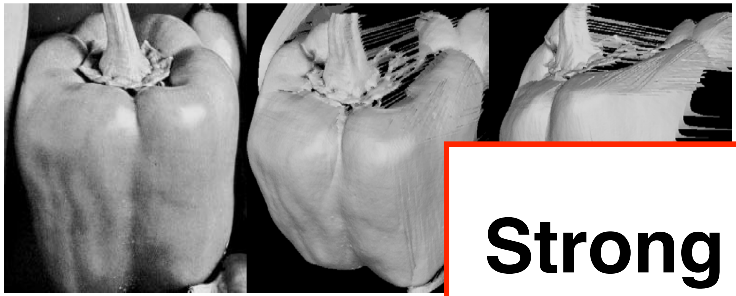


😢

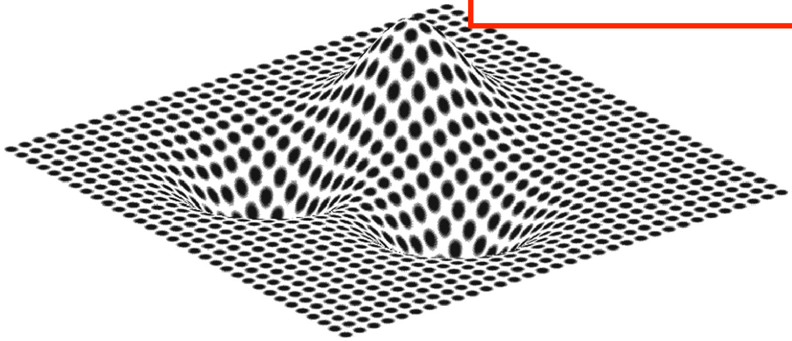
- fine structure
- topological variatio
- …

Status review of monocular vision algorithms

- Shape from X (texture, shading, …)

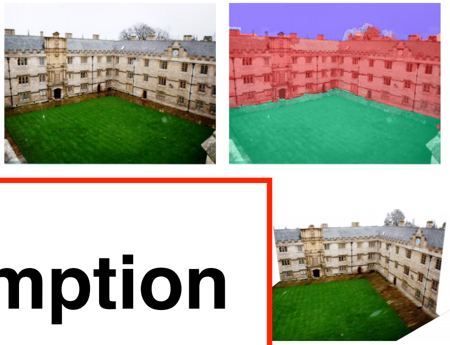


[Horn, 19



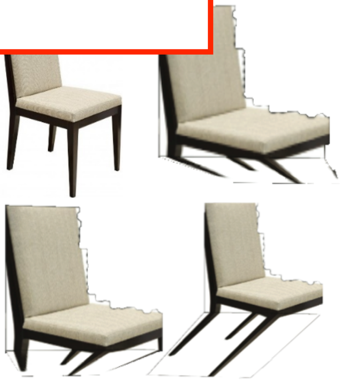
[Kender, 1979]

- Learning-based (from small data)



Hoiem et al, ICCV'05
Saxena et al, NIPS'05
…

- large planes



- fine structure
- topological variatio
- …

Strong assumption
Not robust

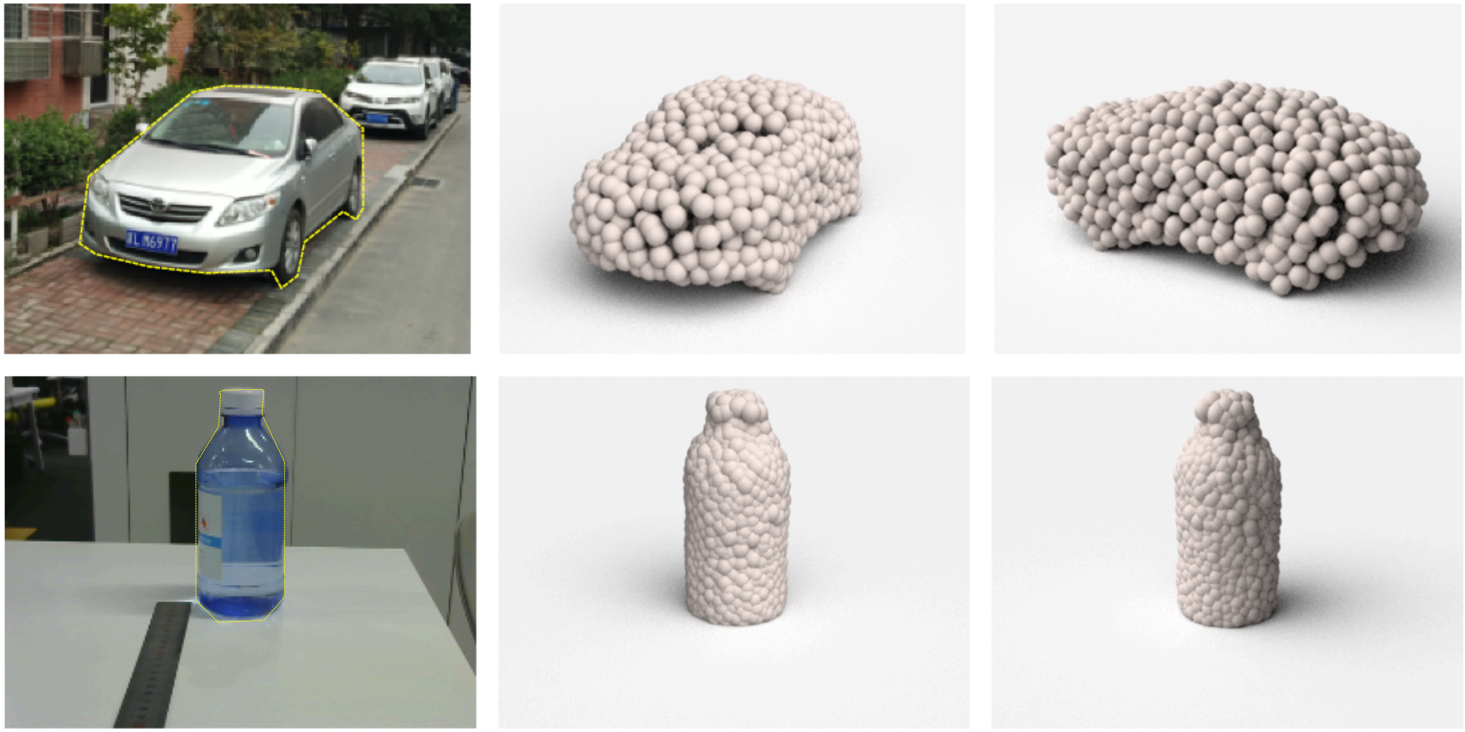
Data-driven 2D-3D lifting

Many 3D objects

A priori knowledge of the 3D world

Our result: 3D reconstruction from real Images

CVPR 2017, A Point Set Generation Network for 3D Object Reconstruction from a Single Image



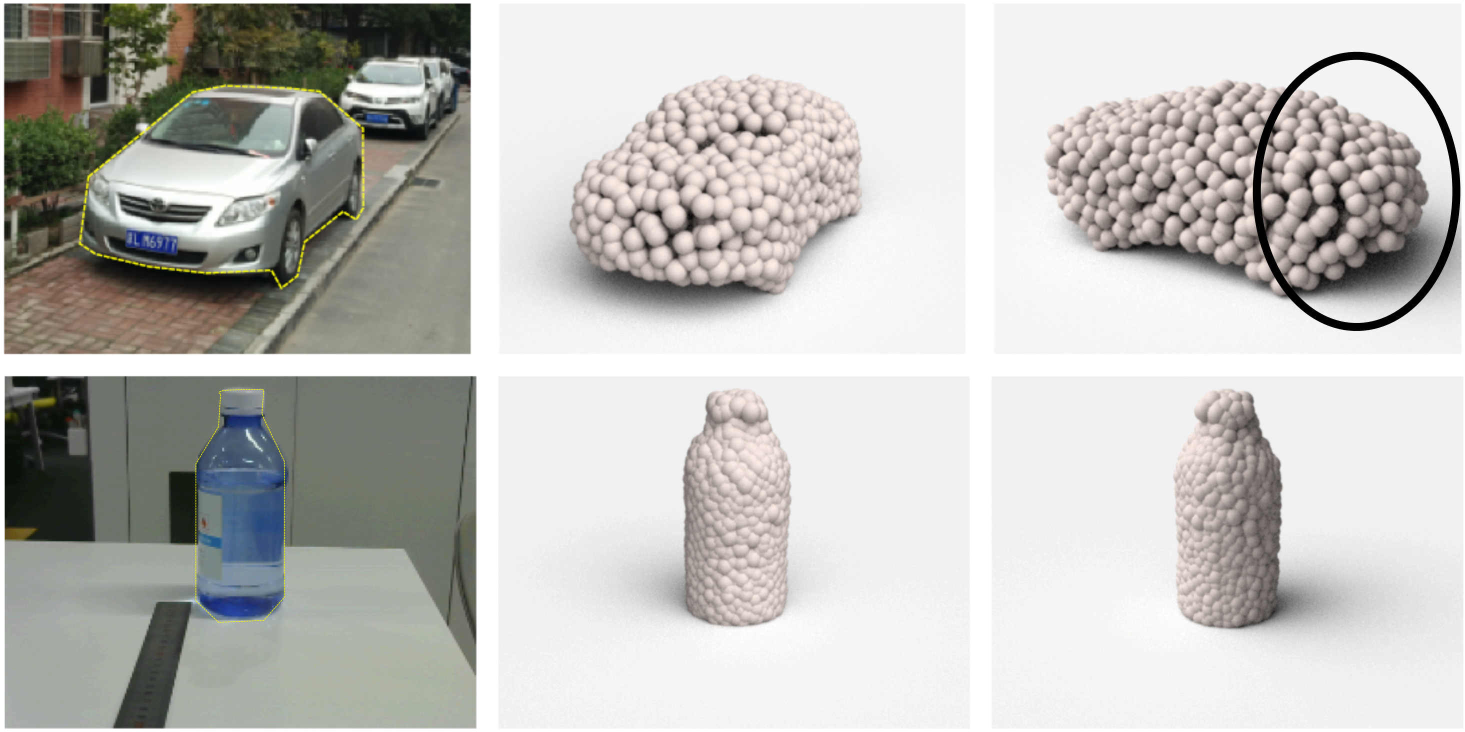
Input

Reconstructed 3D point cloud

CVPR ’17, Point Set Generation

Our result: 3D reconstruction from real Images

CVPR 2017, A Point Set Generation Network for 3D Object Reconstruction from a Single Image



Input

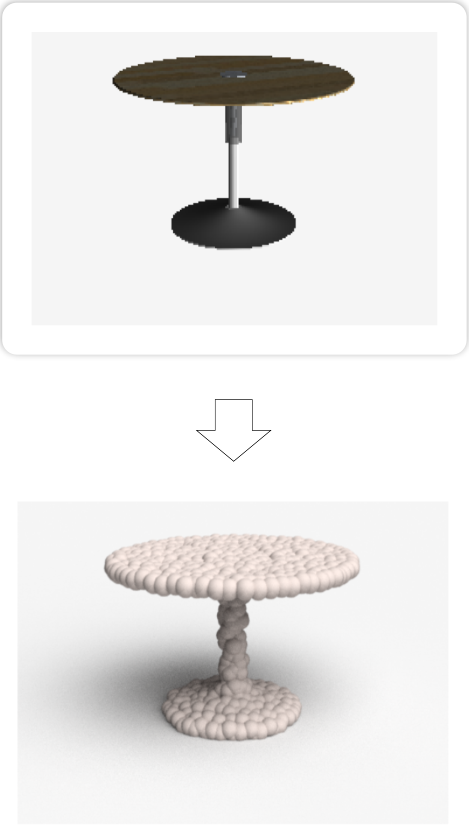
Reconstructed 3D point cloud

CVPR '17, Point Set Generation

3D point clouds

✓ Flexible

- a few thousands of points can precisely model a great variety of shapes



CVPR ’17, Point Set Generation

3D point clouds

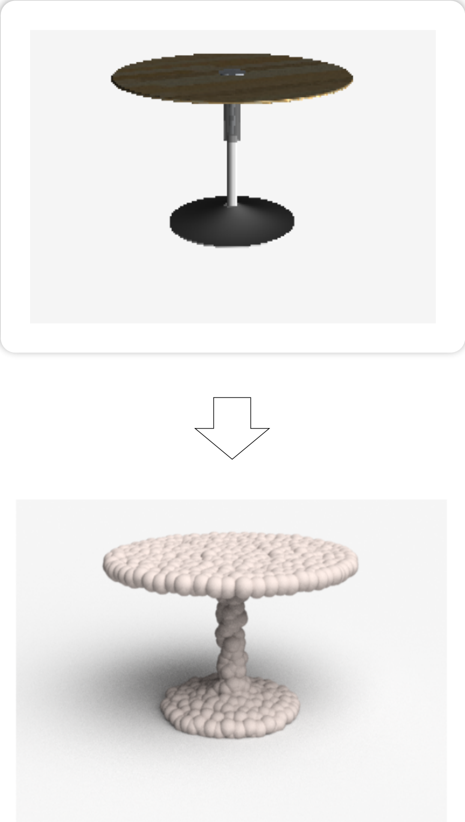
✓ Flexible

- a few thousands of points can precisely model a great variety of shapes

✓

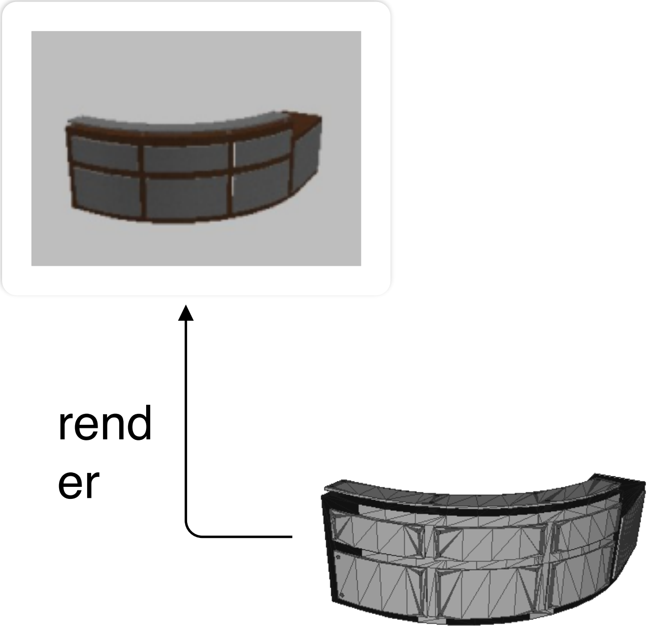
Geometrically manipulable

- deformable
- interpolable, extrapolable
- convenient to impose structural constraints



CVPR '17, Point Set Generation

Pipeline

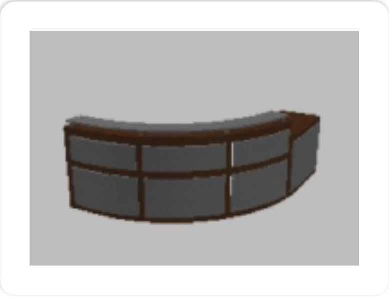


rend
er

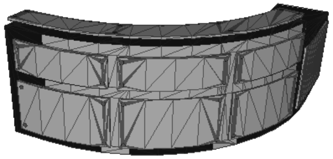
CVPR ’17, Point Set Generation

Pipeline

2K object categories
200K shapes
~10M image/point set pairs



render



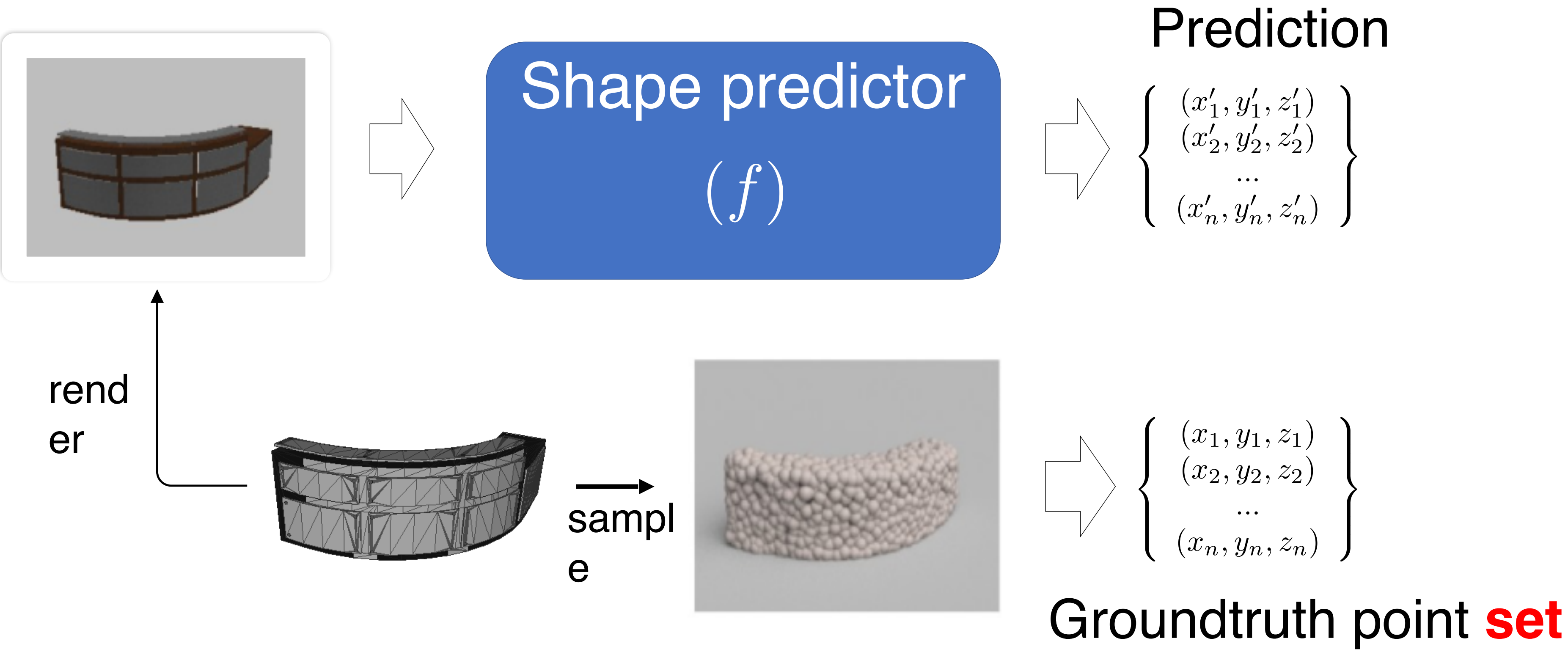
sample

$$\left\{ \begin{array}{c} (x_1, y_1, z_1) \\ (x_2, y_2, z_2) \\ \dots \\ (x_n, y_n, z_n) \end{array} \right\}$$

Groundtruth point **set**

CVPR ’17, Point Set Generation

Pipeline



Shape predictor

(f)

Prediction

$$\left\{ \begin{array}{c} (x'_1, y'_1, z'_1) \\ (x'_2, y'_2, z'_2) \\ \cdots \\ (x'_n, y'_n, z'_n) \end{array} \right\}$$

render

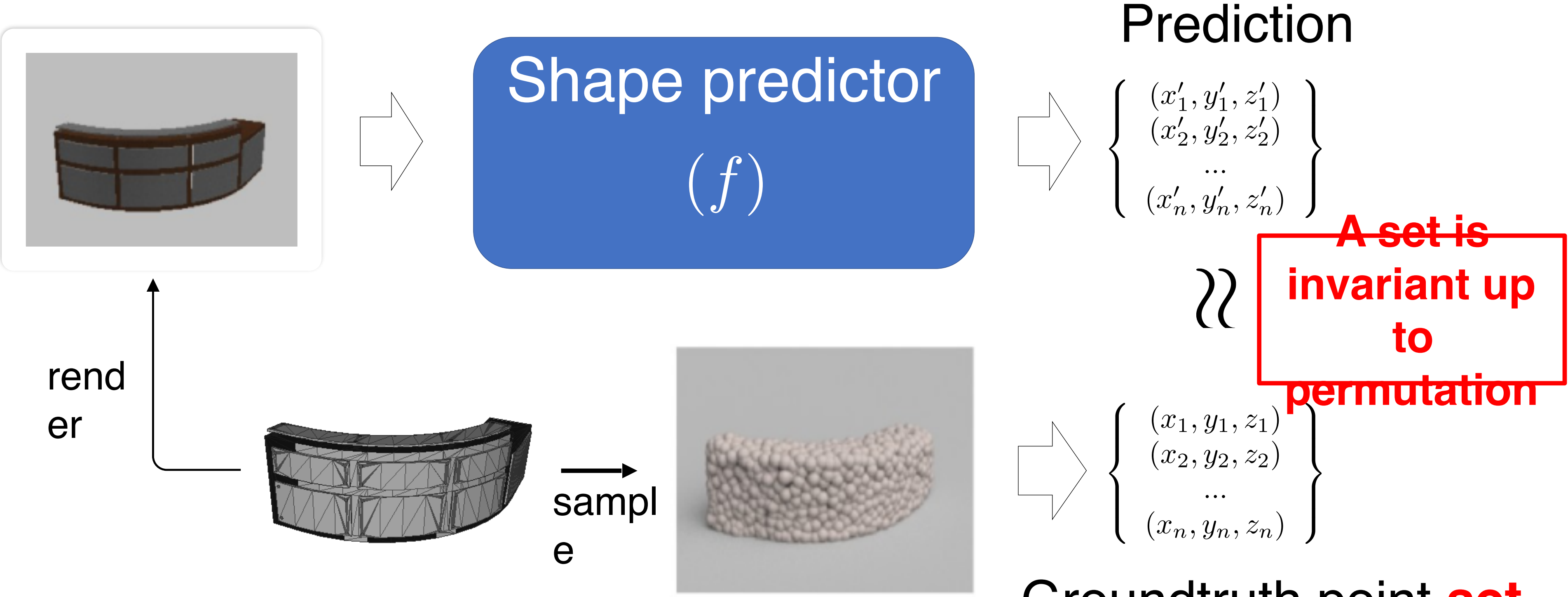
sample

$$\left\{ \begin{array}{c} (x_1, y_1, z_1) \\ (x_2, y_2, z_2) \\ \cdots \\ (x_n, y_n, z_n) \end{array} \right\}$$

Groundtruth point **set**

CVPR ’17, Point Set Generation

Pipeline



Shape predictor

(f)

Prediction

$$\left\{\begin{array}{c}(x'_1, y'_1, z'_1)\\(x'_2, y'_2, z'_2)\\\cdots\\(x'_n, y'_n, z'_n)\end{array}\right\}$$

$\wr\wr$

A set is invariant up to permutation

$$\left\{\begin{array}{c}(x_1, y_1, z_1)\\(x_2, y_2, z_2)\\\cdots\\(x_n, y_n, z_n)\end{array}\right\}$$

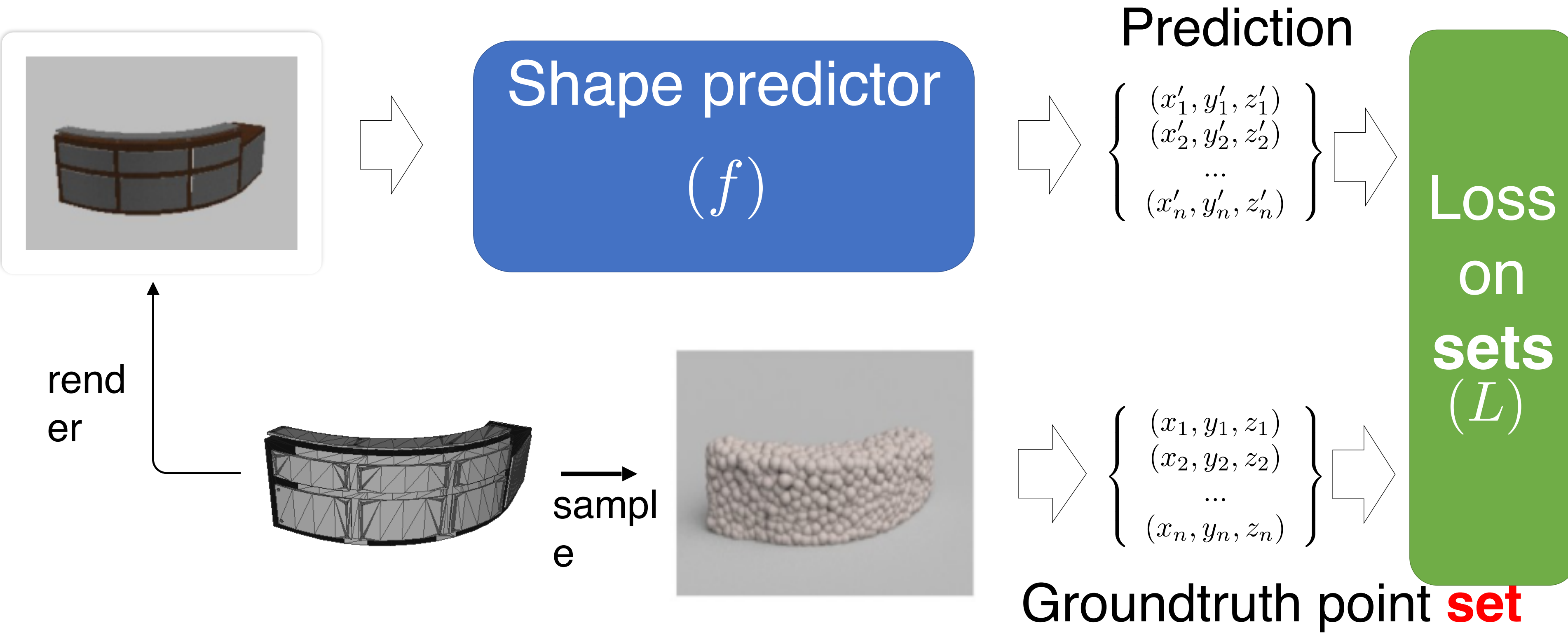
Groundtruth point **set**

render

sample

CVPR '17, Point Set Generation

Pipeline



Shape predictor

(f)

Prediction

$$\begin{Bmatrix} (x'_1, y'_1, z'_1) \\ (x'_2, y'_2, z'_2) \\ \cdots \\ (x'_n, y'_n, z'_n) \end{Bmatrix}$$

Loss on **sets** (L)

rend
er

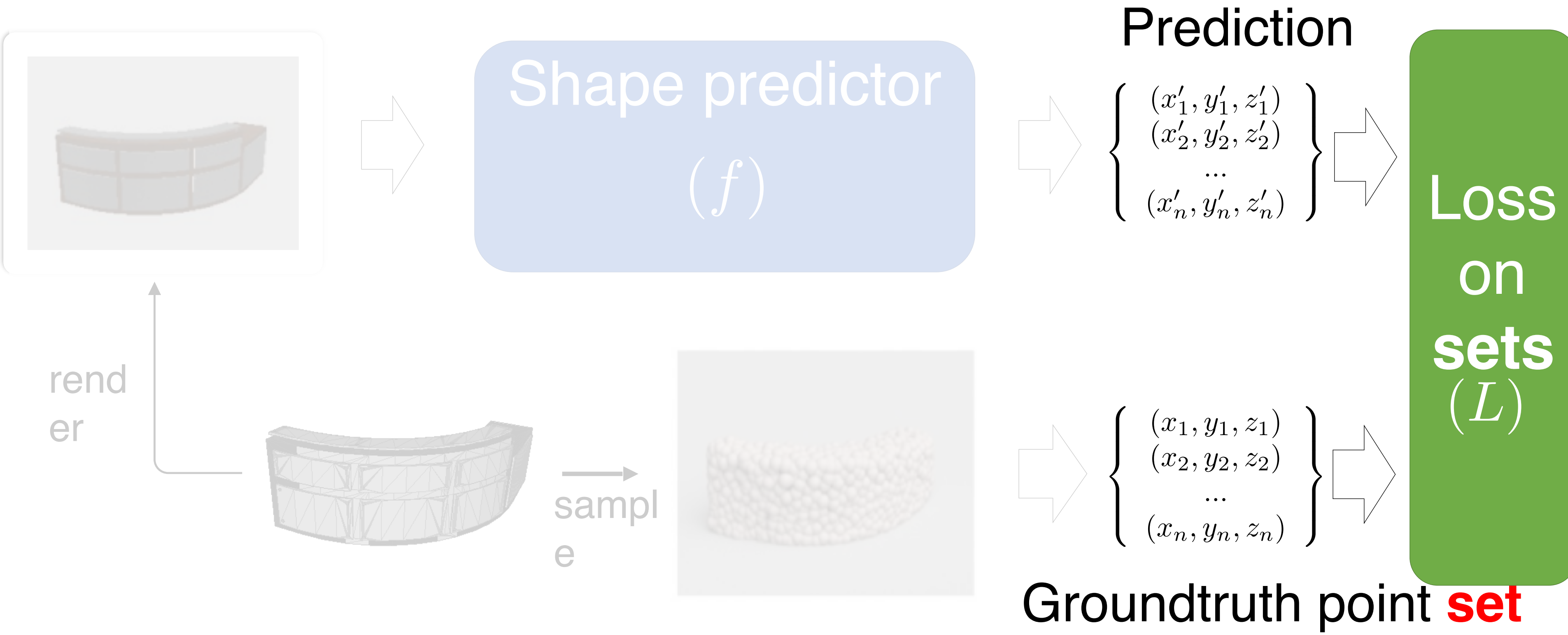
sampl
e

$$\begin{Bmatrix} (x_1, y_1, z_1) \\ (x_2, y_2, z_2) \\ \cdots \\ (x_n, y_n, z_n) \end{Bmatrix}$$

Groundtruth point **set**

CVPR '17, Point Set Generation

Pipeline



Prediction

Shape predictor (f)

$$\begin{Bmatrix} (x'_1, y'_1, z'_1) \\ (x'_2, y'_2, z'_2) \\ \cdots \\ (x'_n, y'_n, z'_n) \end{Bmatrix}$$

Loss on **sets** (L)

render

sample

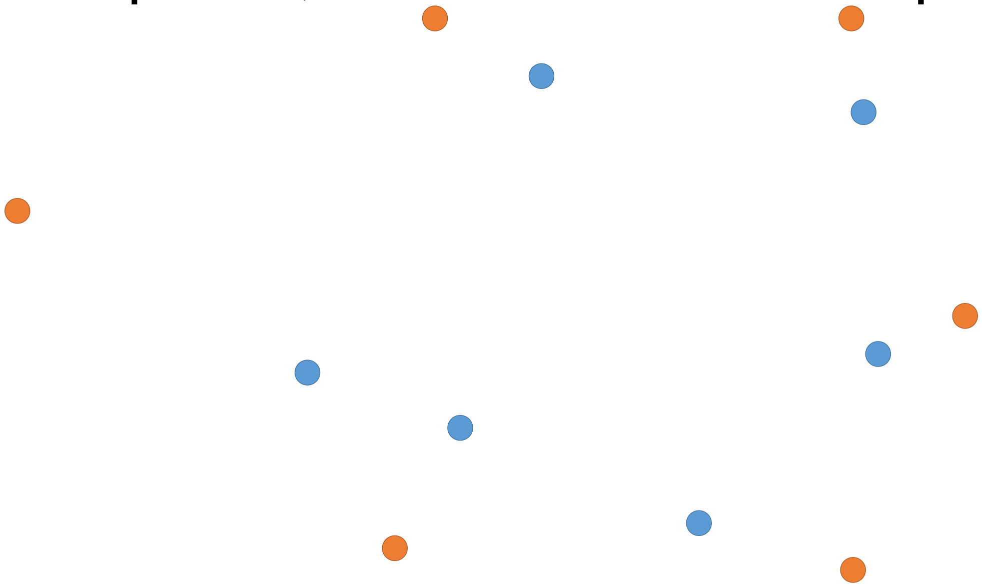
$$\begin{Bmatrix} (x_1, y_1, z_1) \\ (x_2, y_2, z_2) \\ \cdots \\ (x_n, y_n, z_n) \end{Bmatrix}$$

Groundtruth point **set**

CVPR '17, Point Set Generation

Set comparison

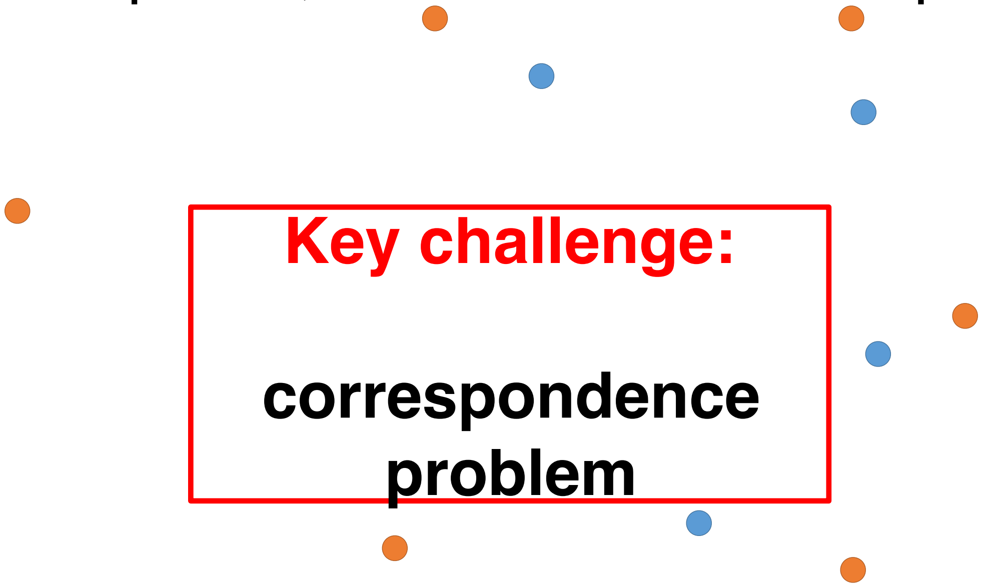
Given two sets of points, measure their discrepancy



CVPR '17, Point Set Generation

Set comparison

Given two sets of points, measure their discrepancy



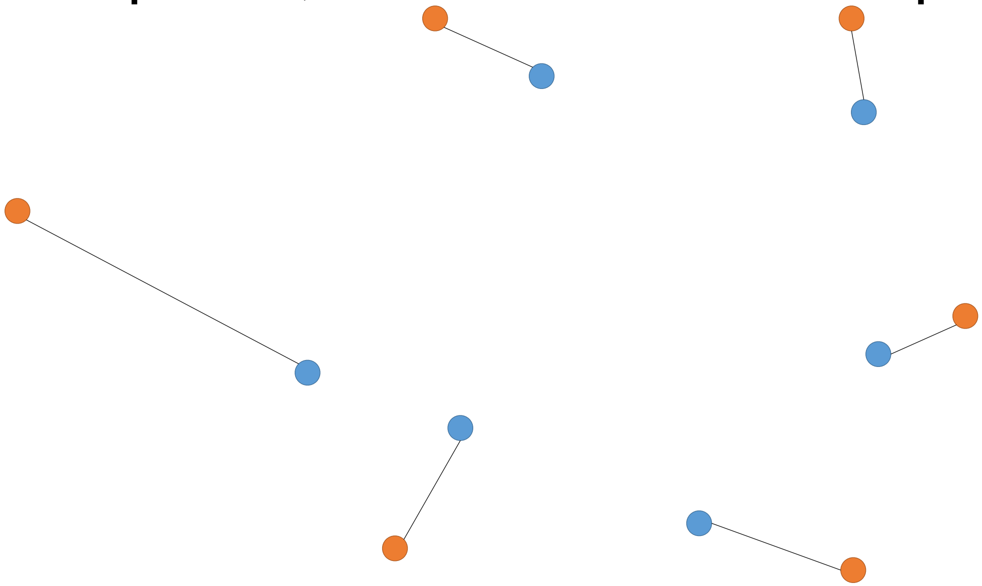
Key challenge:

correspondence problem

CVPR ’17, Point Set Generation

Correspondence (I): optimal assignment

Given two sets of points, measure their discrepancy



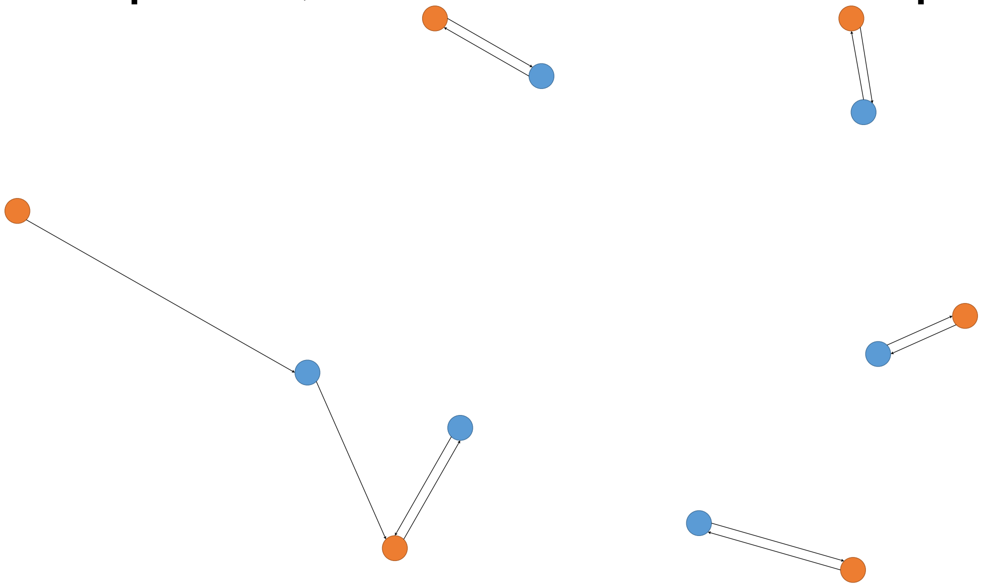
a.k.a Earth Mover's distance (EMD)

$$d_{EMD}(S_1, S_2) = \min_{\phi: S_1 \rightarrow S_2} \sum_{x \in S_1} \|x - \phi(x)\|_2 \quad \text{where } \phi : S_1 \rightarrow S_2 \text{ is a bijection.}$$

CVPR '17, Point Set Generation

Correspondence (II): closest point

Given two sets of points, measure their discrepancy



a.k.a Chamfer distance (CD)

$$d_{CD}(S_1, S_2) = \sum_{x \in S_1} \min_{y \in S_2} \|x - y\|_2^2 + \sum_{y \in S_2} \min_{x \in S_1} \|x - y\|_2^2$$

CVPR '17, Point Set Generation

Required properties of distance metrics

Geometric requirement

Computational requirement

CVPR ’17, Point Set Generation

Required properties of distance metrics

Geometric requirement

- Reflects natural shape differences
- Induce a nice space for ***shape interpolations***

Computational requirement

CVPR '17, Point Set Generation

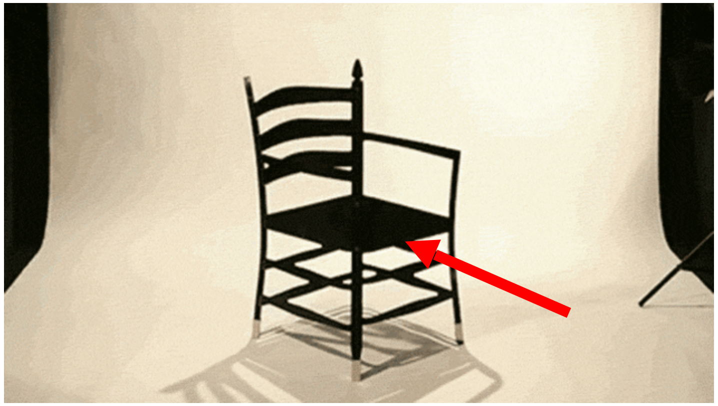
How distance metric affects learning?

A fundamental issue: inherent ambiguity in 2D-3D dimension lifting

CVPR '17, Point Set Generation

How distance metric affects learning?

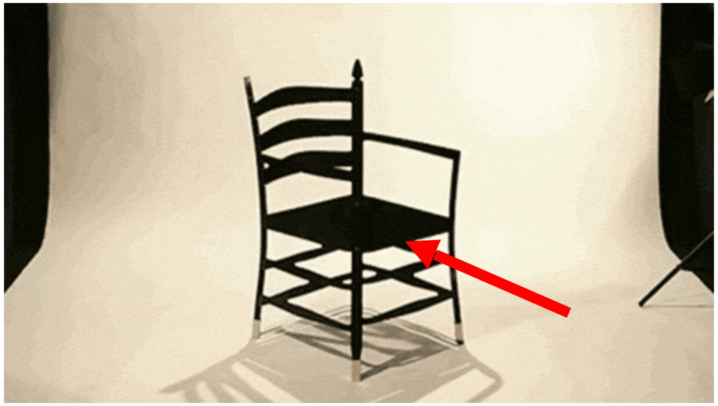
A fundamental issue: inherent ambiguity in 2D-3D dimension lifting



CVPR ’17, Point Set Generation

How distance metric affects learning?

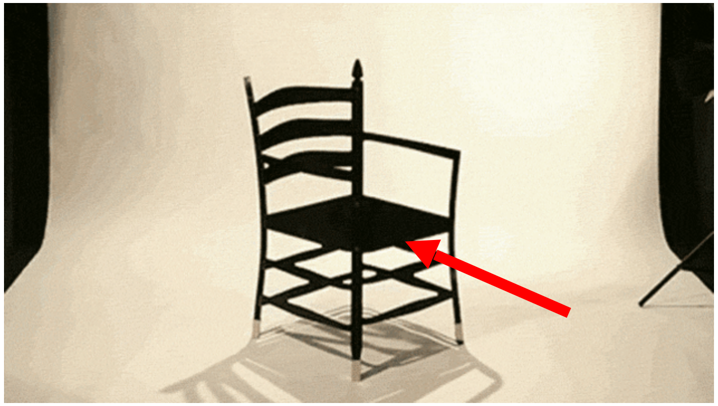
A fundamental issue: inherent ambiguity in 2D-3D dimension lifting

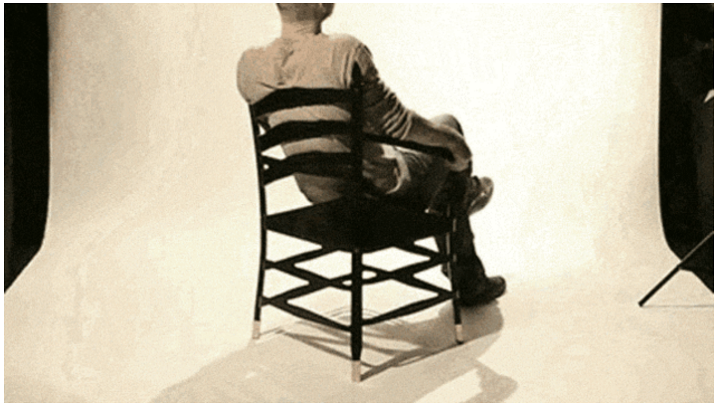


CVPR ’17, Point Set Generation

How distance metric affects learning?

A fundamental issue: inherent ambiguity in 2D-3D dimension lifting





- By loss minimization, the network tends to predict a “**mean shape**” that **averages out** uncertainty

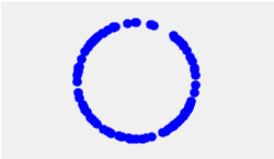
CVPR ’17, Point Set Generation

Distance metrics affect mean shapes

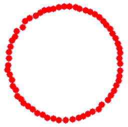
The mean shape carries characteristics of the distance metric

$$\bar{x} = \operatorname*{argmin}_{x} \mathbb{E}_{s\sim\mathbb{S}}[d(x,s)]$$

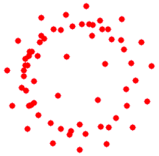
continuous hidden variable (radius)



Input



EMD mean



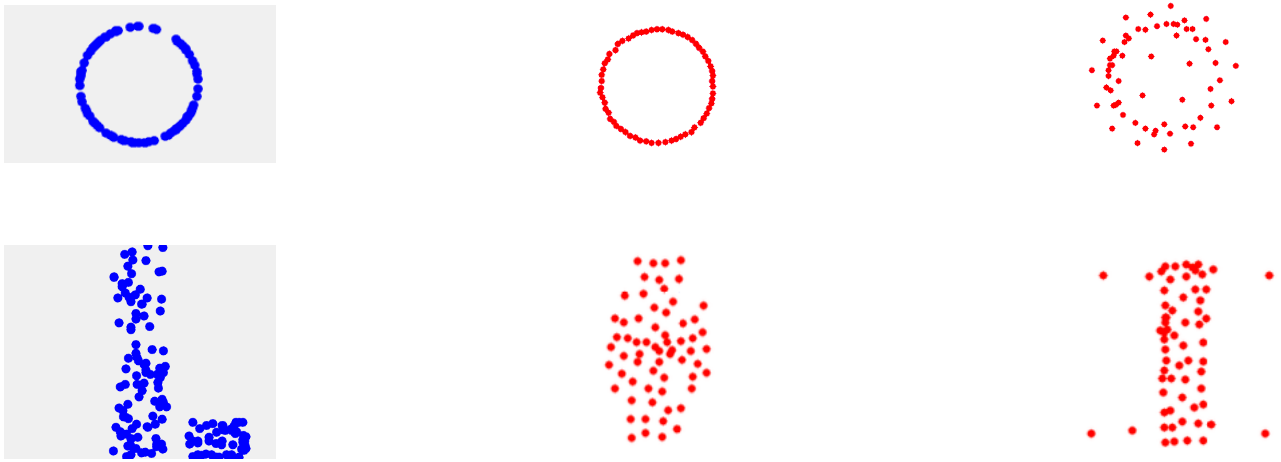
Chamfer mean

CVPR ’17, Point Set Generation

Mean shapes from distance metrics

The mean shape carries characteristics of the distance metric

$$\bar{x} = \underset{x}{\operatorname{argmin}}\ \mathbb{E}_{s \sim \mathbb{S}}[d(x, s)]$$



continuous hidden variable (radius)

discrete hidden variable add-on location)

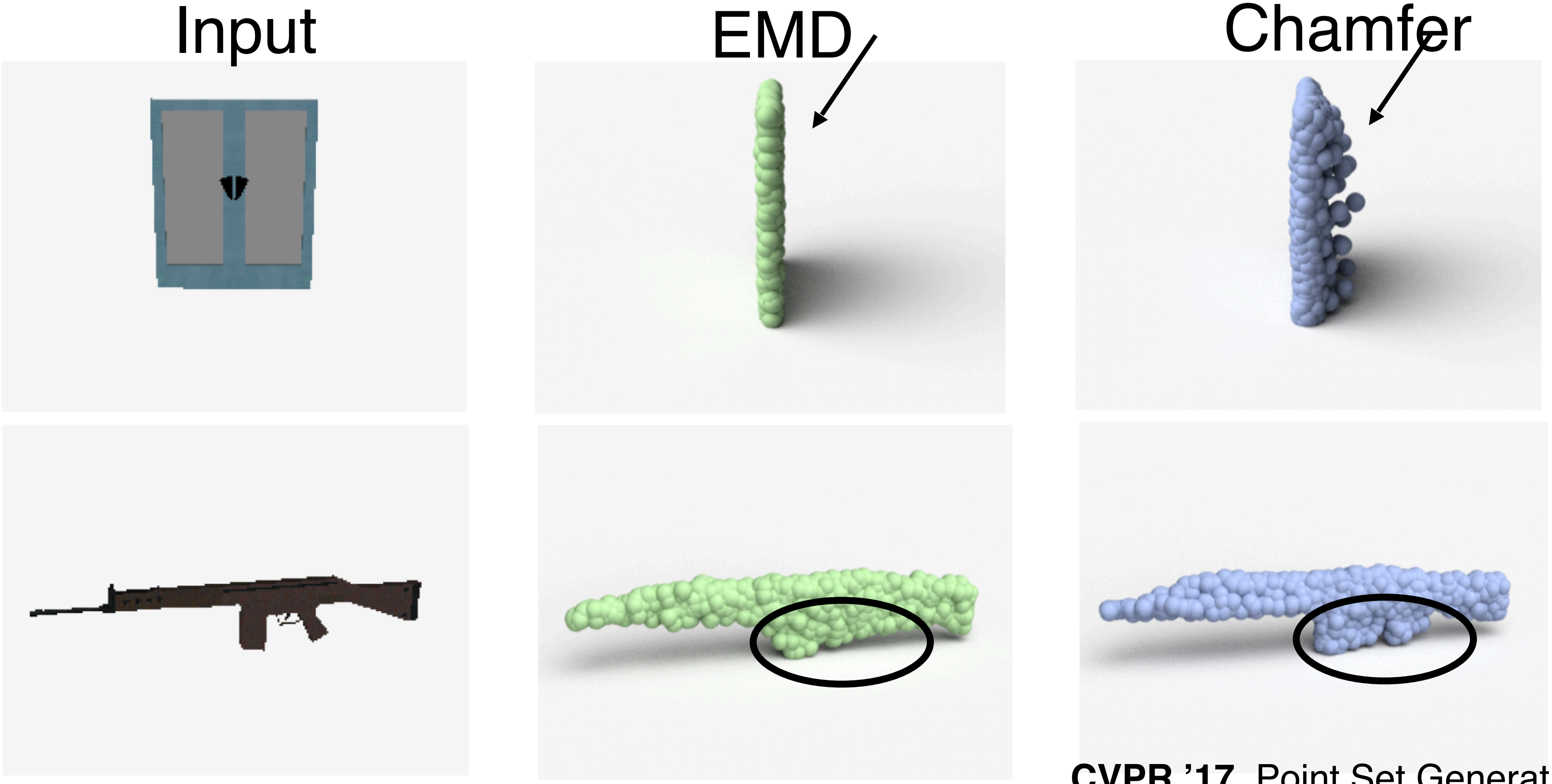
Input

EMD mean

Chamfer mean

CVPR ’17, Point Set Generation

Comparison of predictions by EMD versus CD



Input

EMD

Chamfer

CVPR ’17, Point Set Generation

Required properties of distance metrics

Geometric requirement

- Reflects natural shape differences
- Induce a nice space for shape interpolations

Computational requirement

- Defines a loss function that is numerically easy to optimize

CVPR ’17, Point Set Generation

Computational requirement of metrics

To be used as a loss function, the metric has to be

- **Differentiable** with respect to point locations
- **Efficient** to compute

CVPR '17, Point Set Generation

Computational requirement of metrics

- **Differentiable** with respect to point location

Chamfer distance

$$d_{CD}(S_1, S_2) = \sum_{x \in S_1} \min_{y \in S_2} \|x - y\|_2^2 + \sum_{y \in S_2} \min_{x \in S_1} \|x - y\|_2^2$$ ✓

Earth Mover's distance

$$d_{EMD}(S_1, S_2) = \min_{\phi: S_1 \to S_2} \sum_{x \in S_1} \|x - \phi(x)\|_2 \quad \text{where } \phi : S_1 \to S_2 \text{ is a bijection.}$$ ✓

- Simple function of coordinates
- In general positions, the correspondence is unique
- **With infinitesimal movement, the correspondence does not change**

Conclusion: differentiable almost everywhere

Computational requirement of metrics

- **Differentiable** with respect to point location

- For many **algorithms** (sorting, shortest path, network flow, …),
- an infinitesimal change to model parameters

Co **ere**

CVPR '17, Point Set Generation

Computational requirement of metrics

- **Efficient** to compute

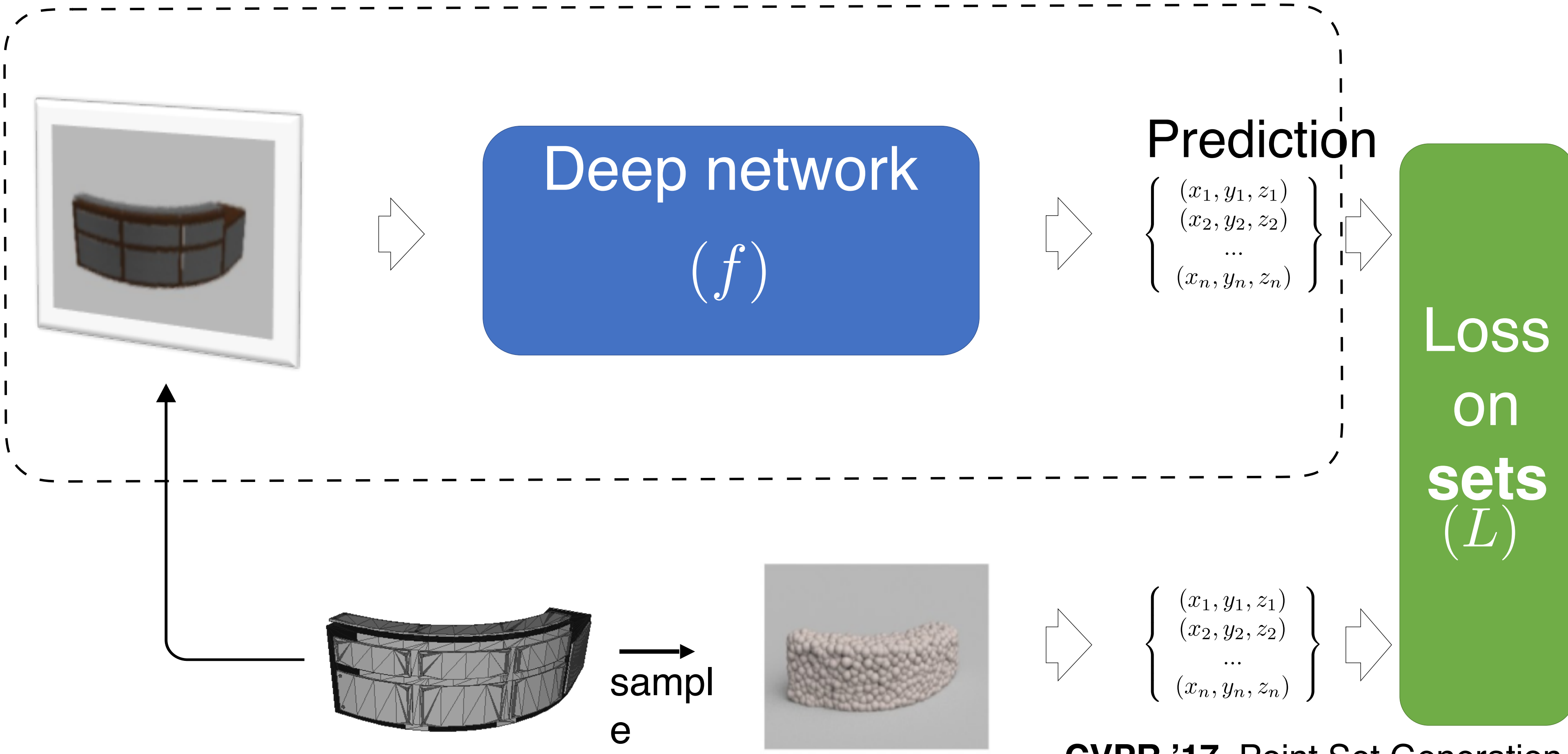
Chamfer distance: trivially parallelizable on CUDA

Earth Mover's distance (optimal assignment):

- We implement a **distributed** approximation algorithm on CUDA
- Based upon [Bertsekas, 1985], $(1+\epsilon)$-approximation

CVPR '17, Point Set Generation

Pipeline



Deep network

(f)

Prediction

$$\left\{\begin{array}{c}(x_1,y_1,z_1)\\(x_2,y_2,z_2)\\\dots\\(x_n,y_n,z_n)\end{array}\right\}$$

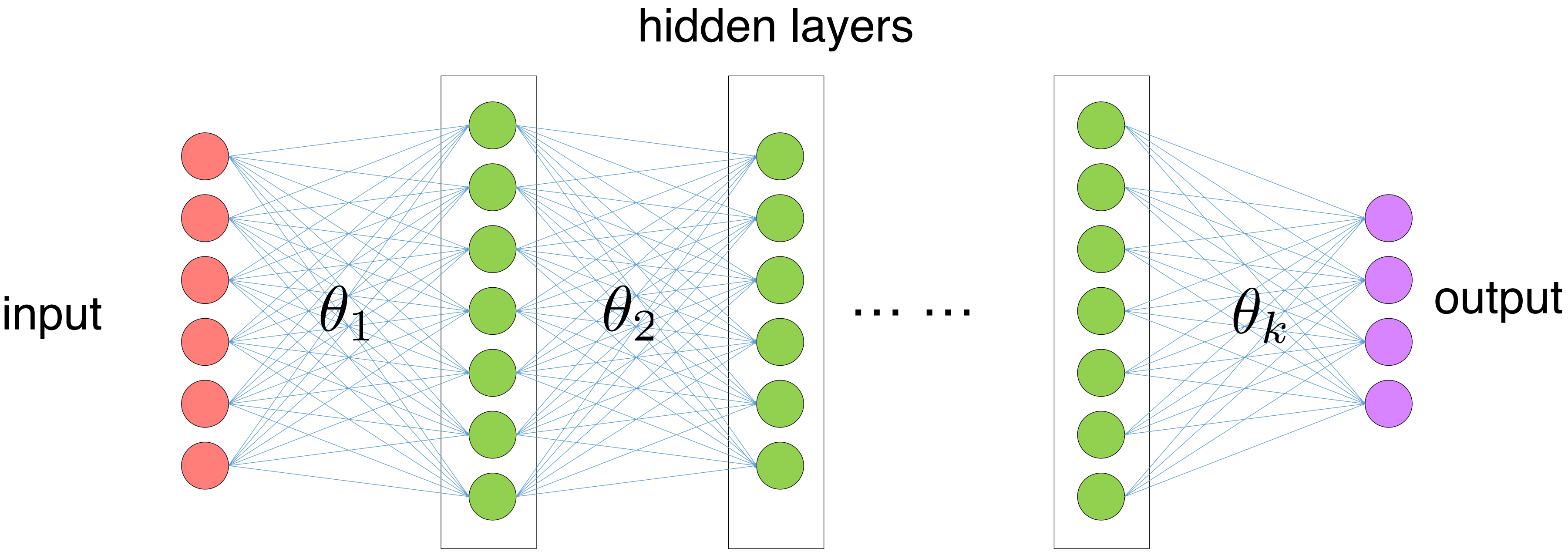
Loss on **sets** (L)

sample

$$\left\{\begin{array}{c}(x_1,y_1,z_1)\\(x_2,y_2,z_2)\\\dots\\(x_n,y_n,z_n)\end{array}\right\}$$

CVPR ’17, Point Set Generation

Deep neural network

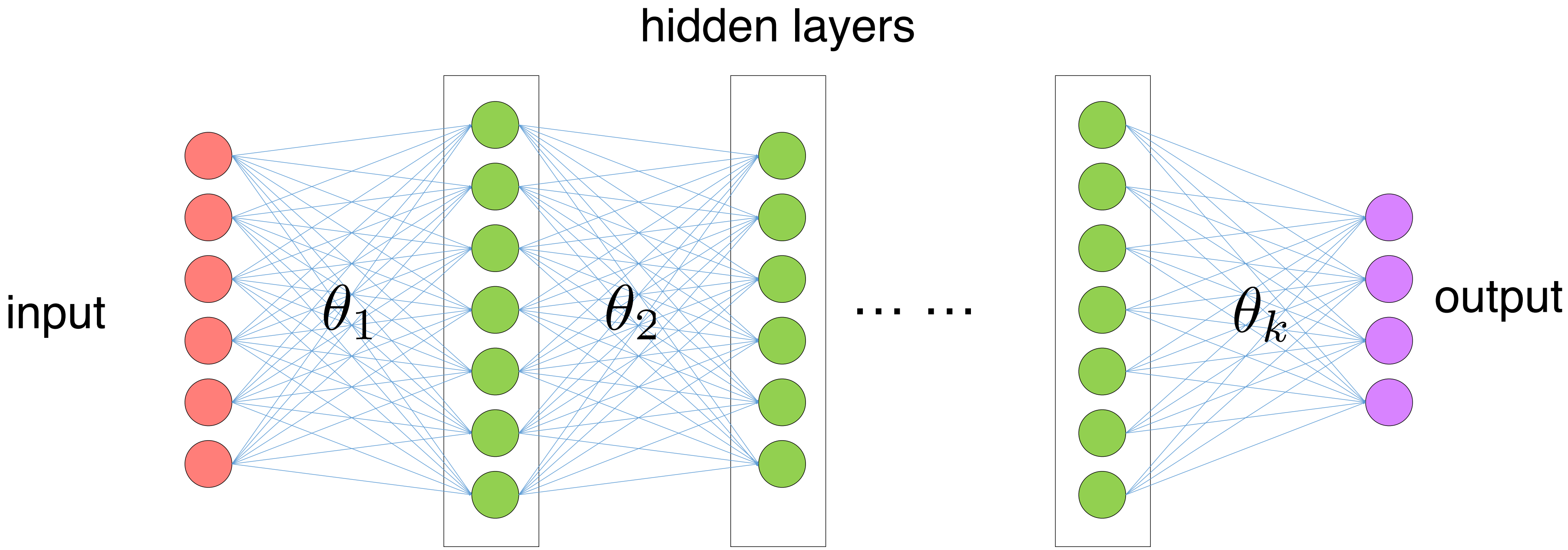


Universal function approximator

- A cascade of layers

CVPR ’17, Point Set Generation

Deep neural network

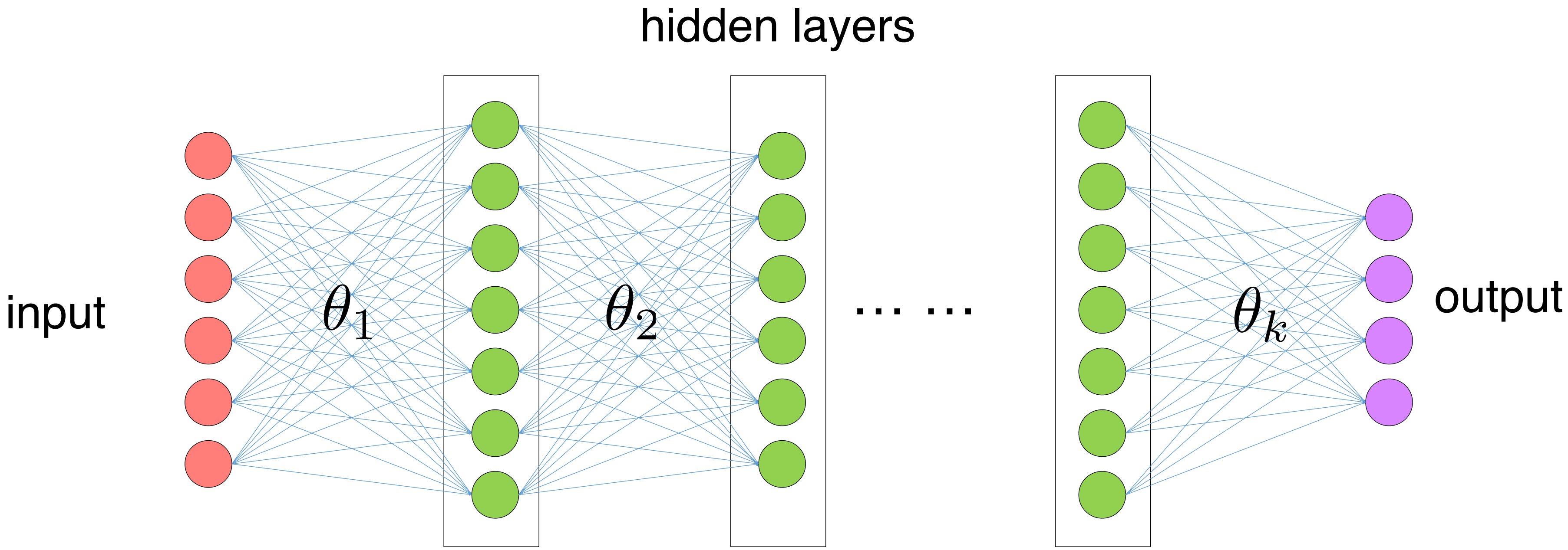


Universal function approximator

- A cascade of layers
- Each layer conducts a simple transformation (parameterized)

CVPR '17, Point Set Generation

Deep neural network

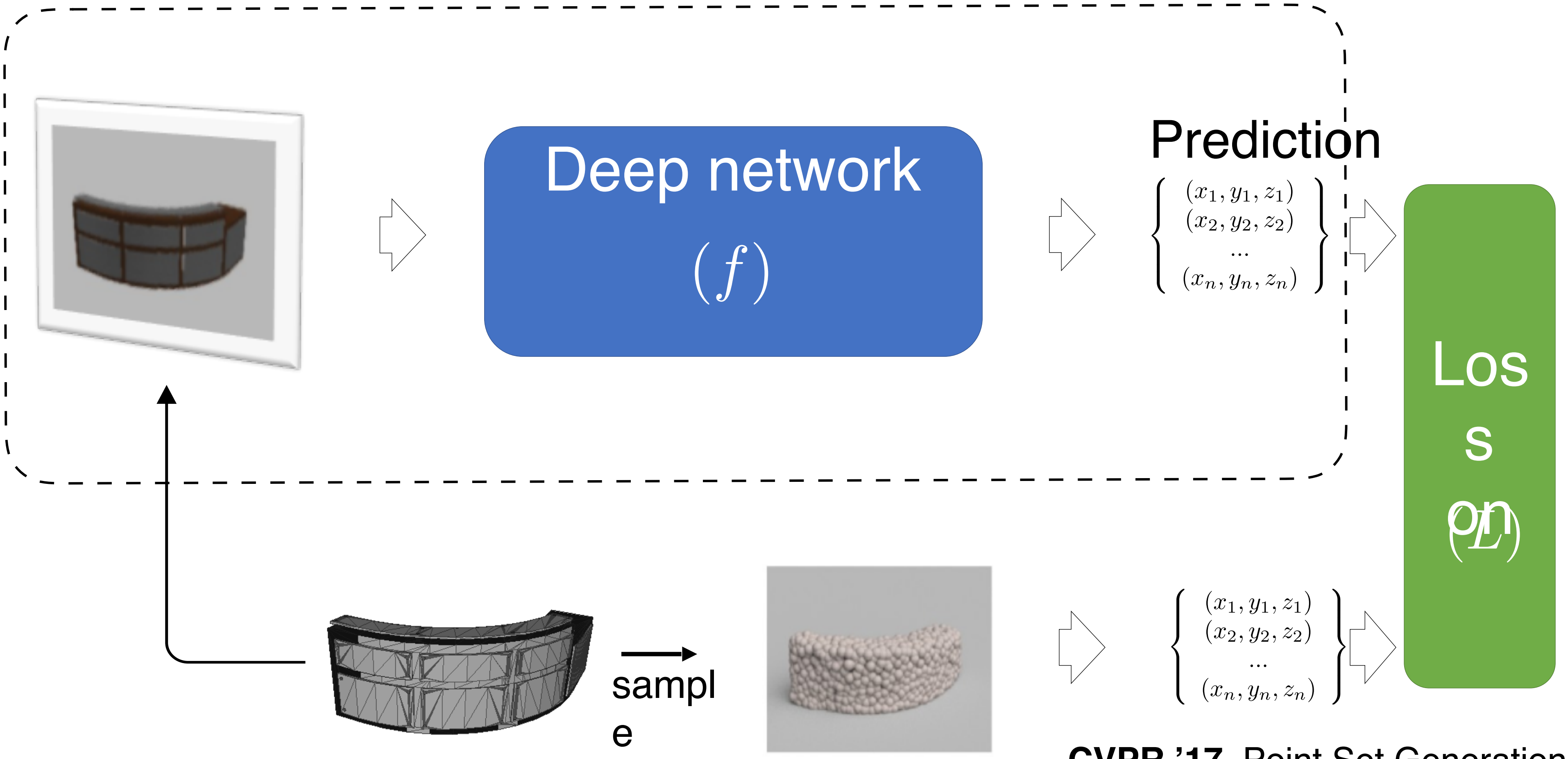


Universal function approximator

- A cascade of layers
- Each layer conducts a simple transformation (parameterized)
- Millions of parameters, has to be fitted by many data

CVPR '17, Point Set Generation

Pipeline



Deep network

(f)

Prediction

$$\begin{Bmatrix} (x_1, y_1, z_1) \\ (x_2, y_2, z_2) \\ \ldots \\ (x_n, y_n, z_n) \end{Bmatrix}$$

Los
s
on

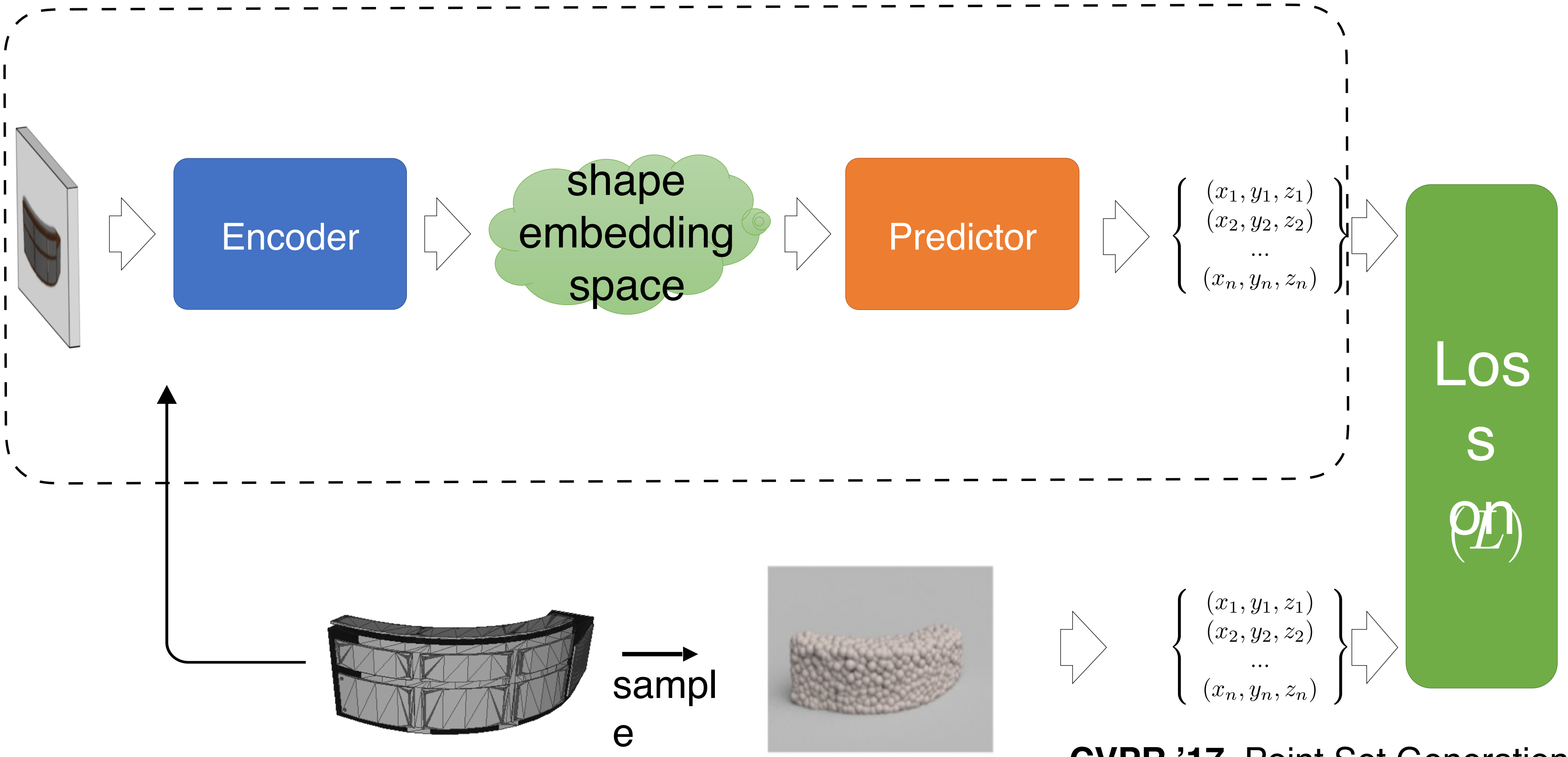
$(\mathcal{L})$

sampl
e

$$\begin{Bmatrix} (x_1, y_1, z_1) \\ (x_2, y_2, z_2) \\ \ldots \\ (x_n, y_n, z_n) \end{Bmatrix}$$

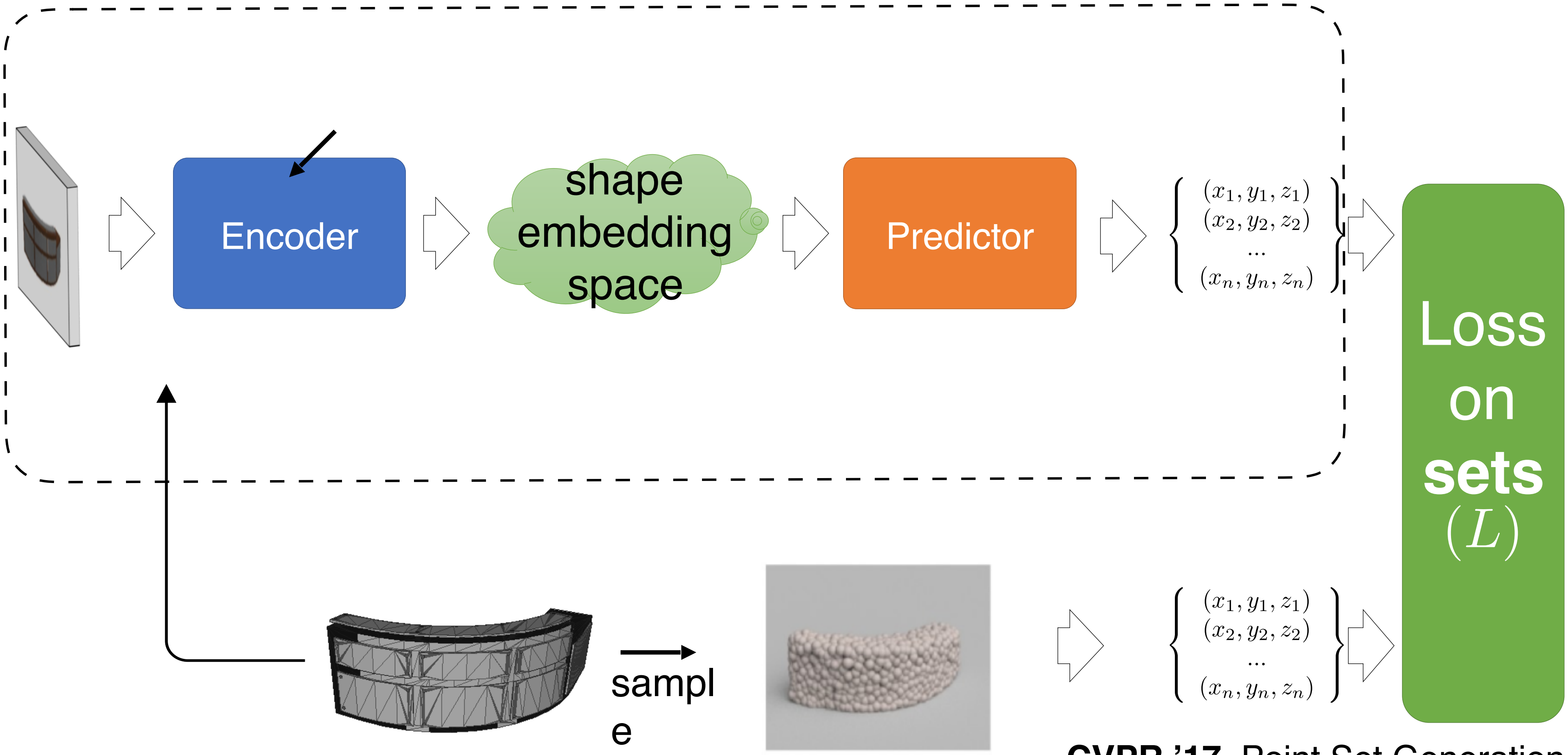
CVPR '17, Point Set Generation

Pipeline


Encoder
shape embedding space
Predictor
$\left\{\begin{matrix}(x_1,y_1,z_1)\\(x_2,y_2,z_2)\\\cdots\\(x_n,y_n,z_n)\end{matrix}\right\}$
Los s on $(\mathcal{L})$
sampl e
$\left\{\begin{matrix}(x_1,y_1,z_1)\\(x_2,y_2,z_2)\\\cdots\\(x_n,y_n,z_n)\end{matrix}\right\}$

CVPR '17, Point Set Generation

Pipeline



Encoder

shape embedding space

Predictor

$$\left\{\begin{matrix}(x_1, y_1, z_1)\\(x_2, y_2, z_2)\\\dots\\(x_n, y_n, z_n)\end{matrix}\right\}$$

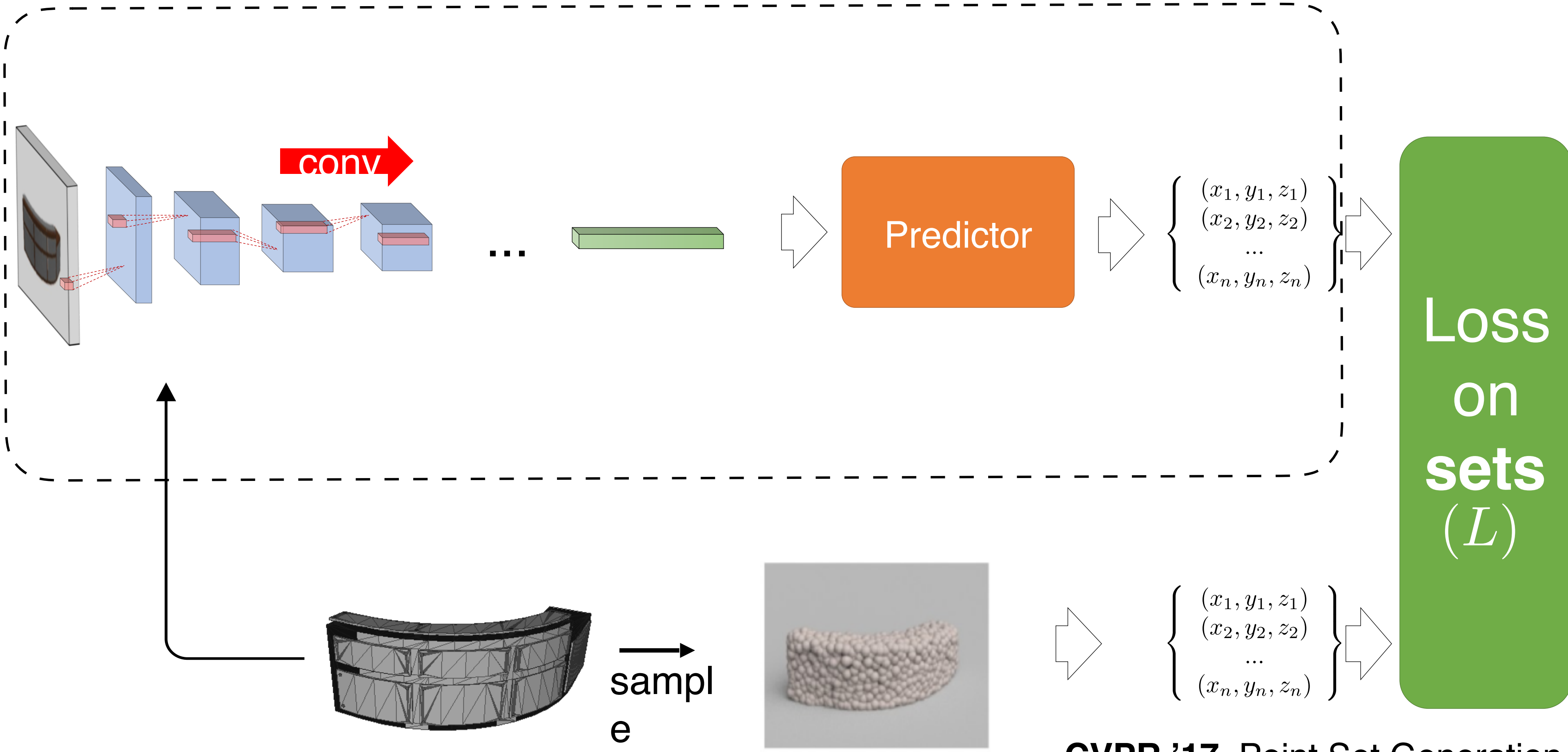
sample

$$\left\{\begin{matrix}(x_1, y_1, z_1)\\(x_2, y_2, z_2)\\\dots\\(x_n, y_n, z_n)\end{matrix}\right\}$$

Loss on **sets** (L)

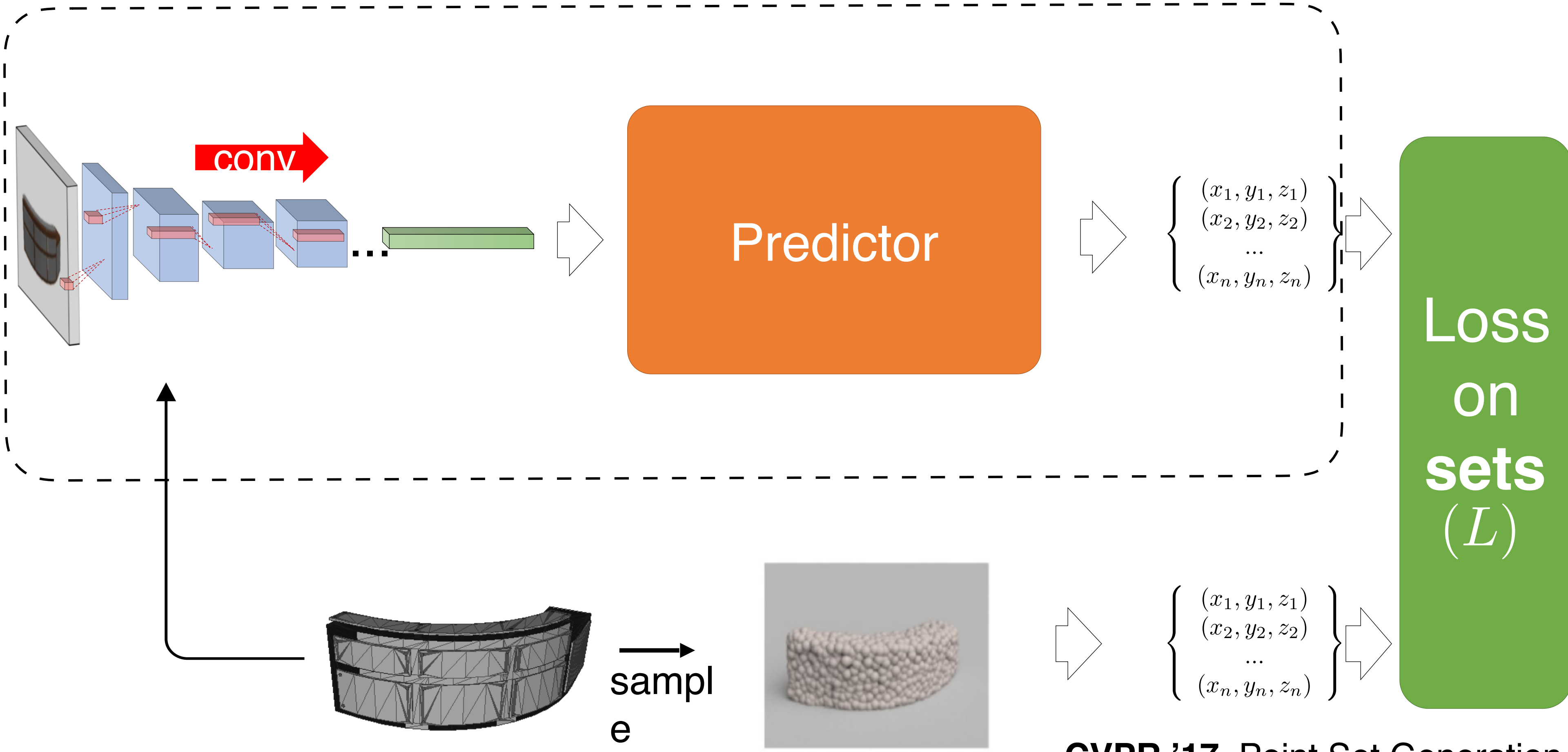
CVPR ’17, Point Set Generation

Pipeline


conv
...
Predictor
$\left\{\begin{matrix}(x_1, y_1, z_1)\\(x_2, y_2, z_2)\\\cdots\\(x_n, y_n, z_n)\end{matrix}\right\}$
Loss
on
sets
(L)
sampl
e
$\left\{\begin{matrix}(x_1, y_1, z_1)\\(x_2, y_2, z_2)\\\cdots\\(x_n, y_n, z_n)\end{matrix}\right\}$

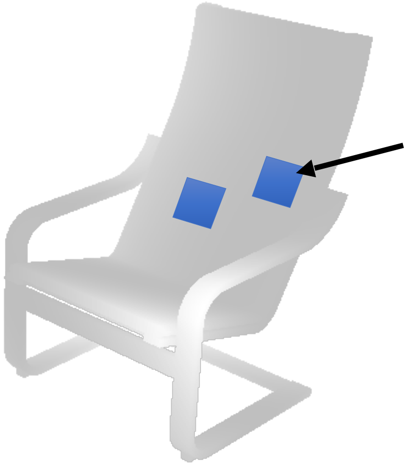
CVPR ’17, Point Set Generation

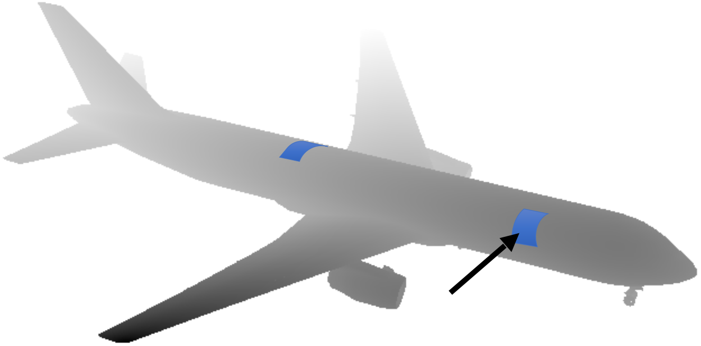
Pipeline


conv
...
Predictor
$\left\{\begin{matrix}(x_1, y_1, z_1)\\(x_2, y_2, z_2)\\\cdots\\(x_n, y_n, z_n)\end{matrix}\right\}$
Loss on **sets** (L)
sampl
e
$\left\{\begin{matrix}(x_1, y_1, z_1)\\(x_2, y_2, z_2)\\\cdots\\(x_n, y_n, z_n)\end{matrix}\right\}$

CVPR ’17, Point Set Generation

Natural statistics of geometry

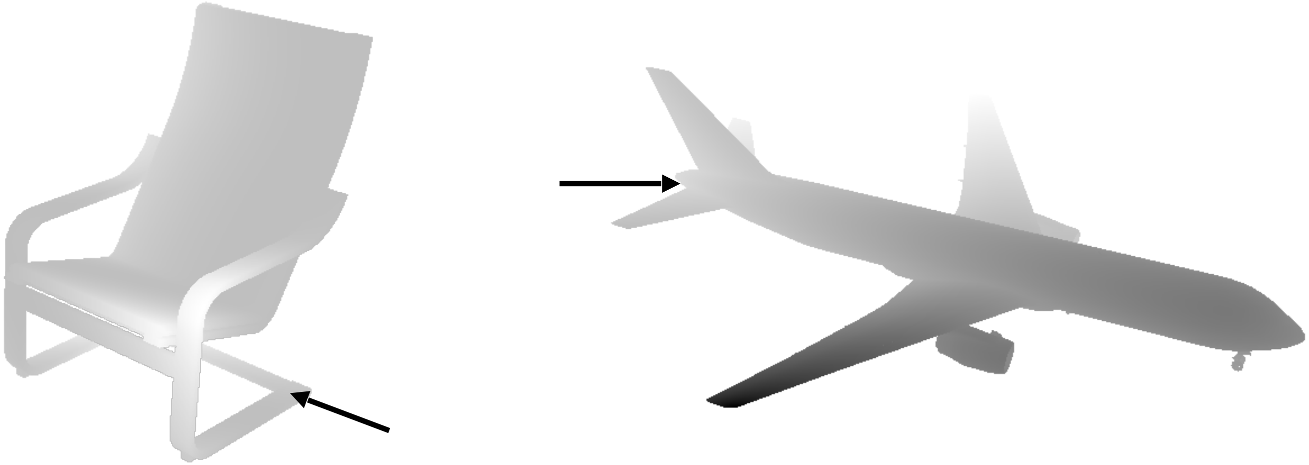




- Many local structures are common
  - e.g., planar patches, cylindrical patches
  - **strong local correlation** among point coordinates

CVPR '17, Point Set Generation

Natural statistics of geometry



- Many local structures are common
  - e.g., planar patches, cylindrical patches
  - **strong local correlation** among point coordinates
- Also some intricate structures
  - points have **high local variation**

CVPR '17, Point Set Generation