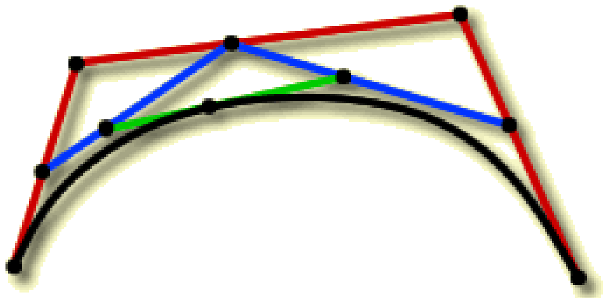
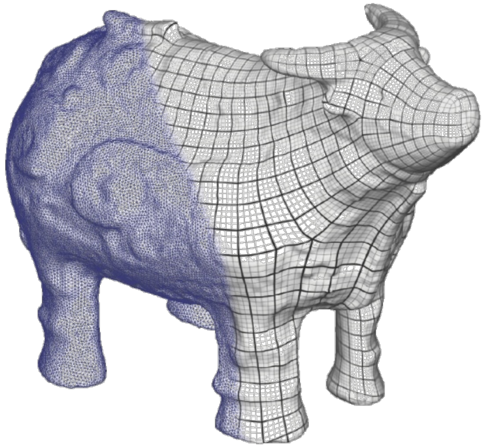
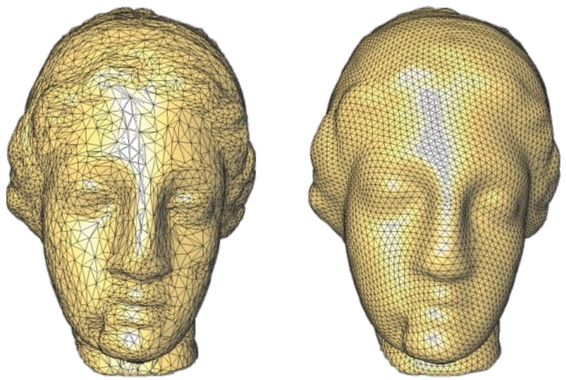
CS348a: Geometric Modeling and Processing







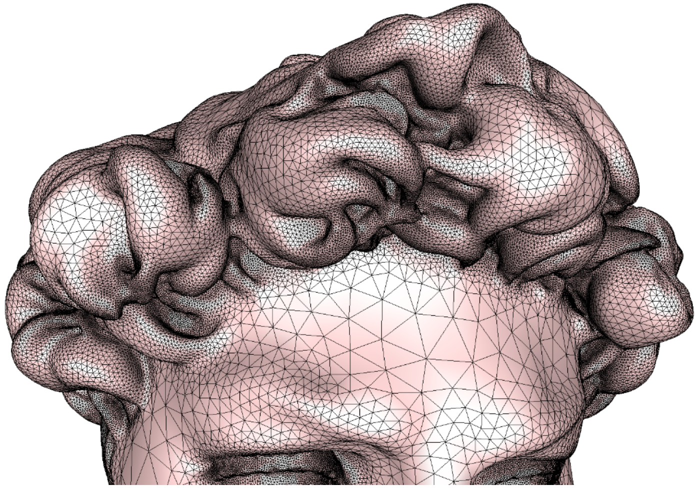
Leonidas Guibas
Computer Science Department
Stanford University

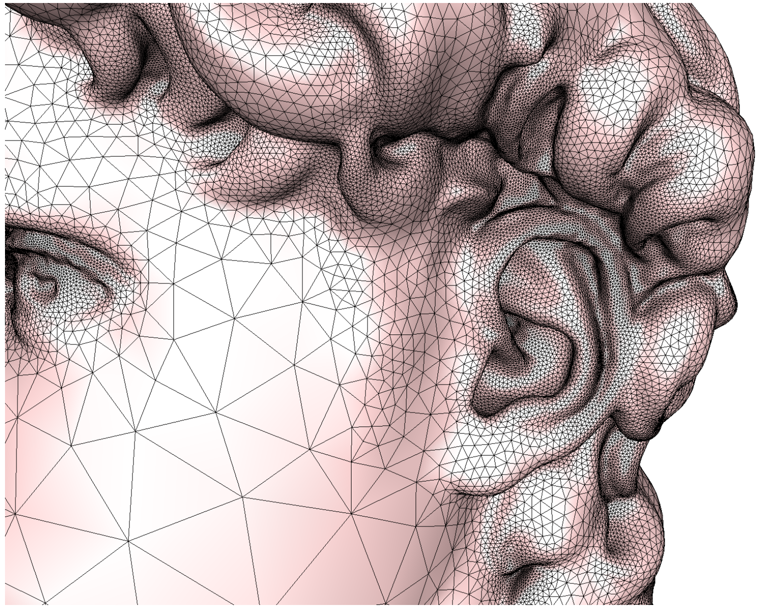
Leonidas Guibas Laboratory

Geometric Computing

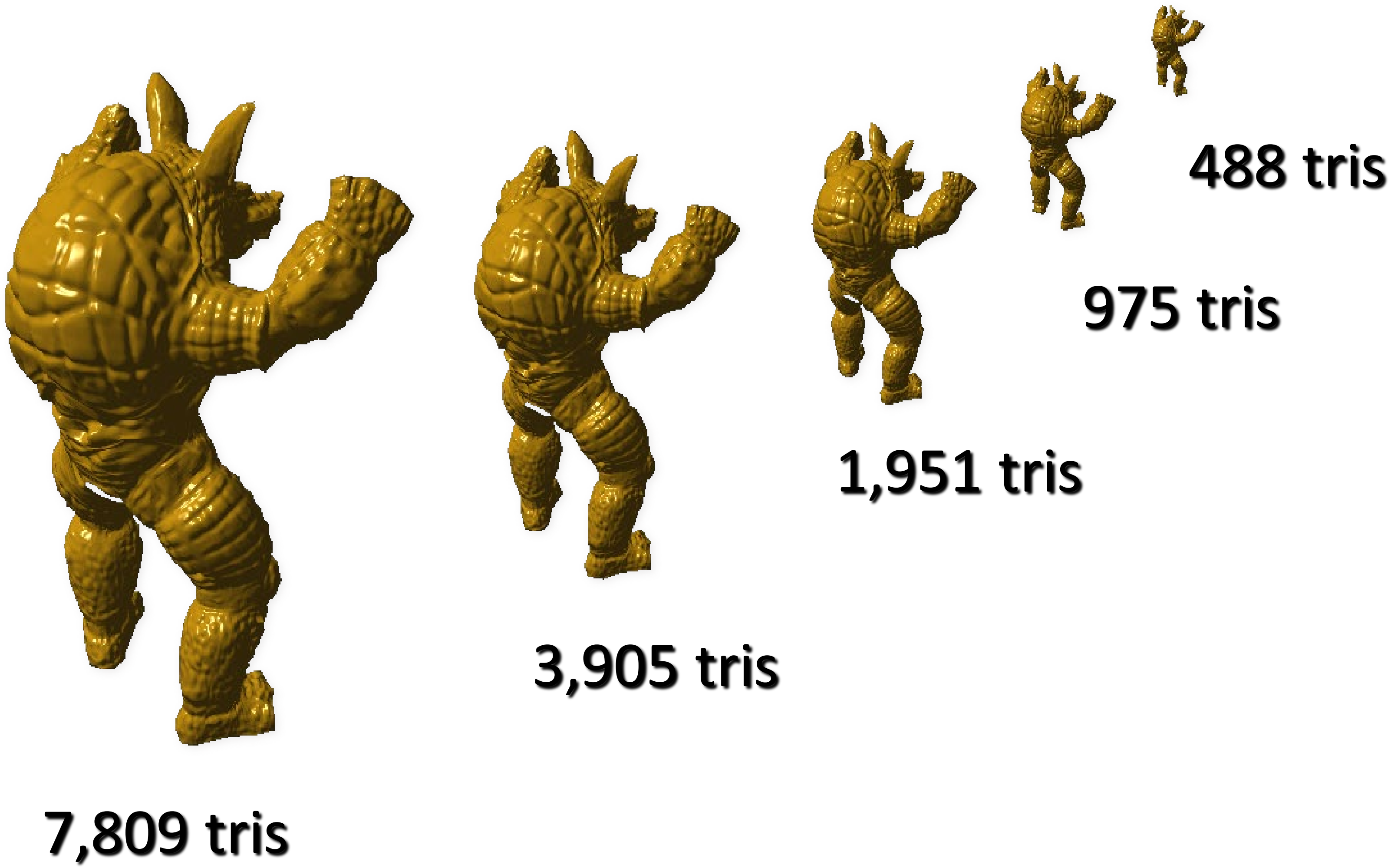
Last Time:
Mesh Simplification w.
Quadratic Error Metrics

Adaptive Level of Detail Simplification





Appearance Preserving Simplification

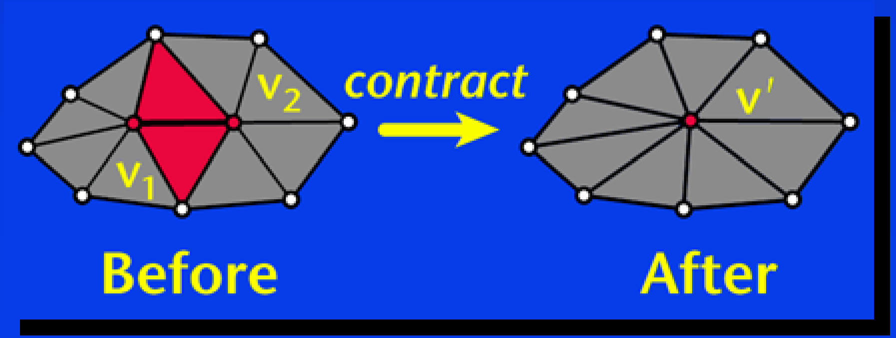


Caltech & Stanford Graphics Labs and Jonathan Cohen

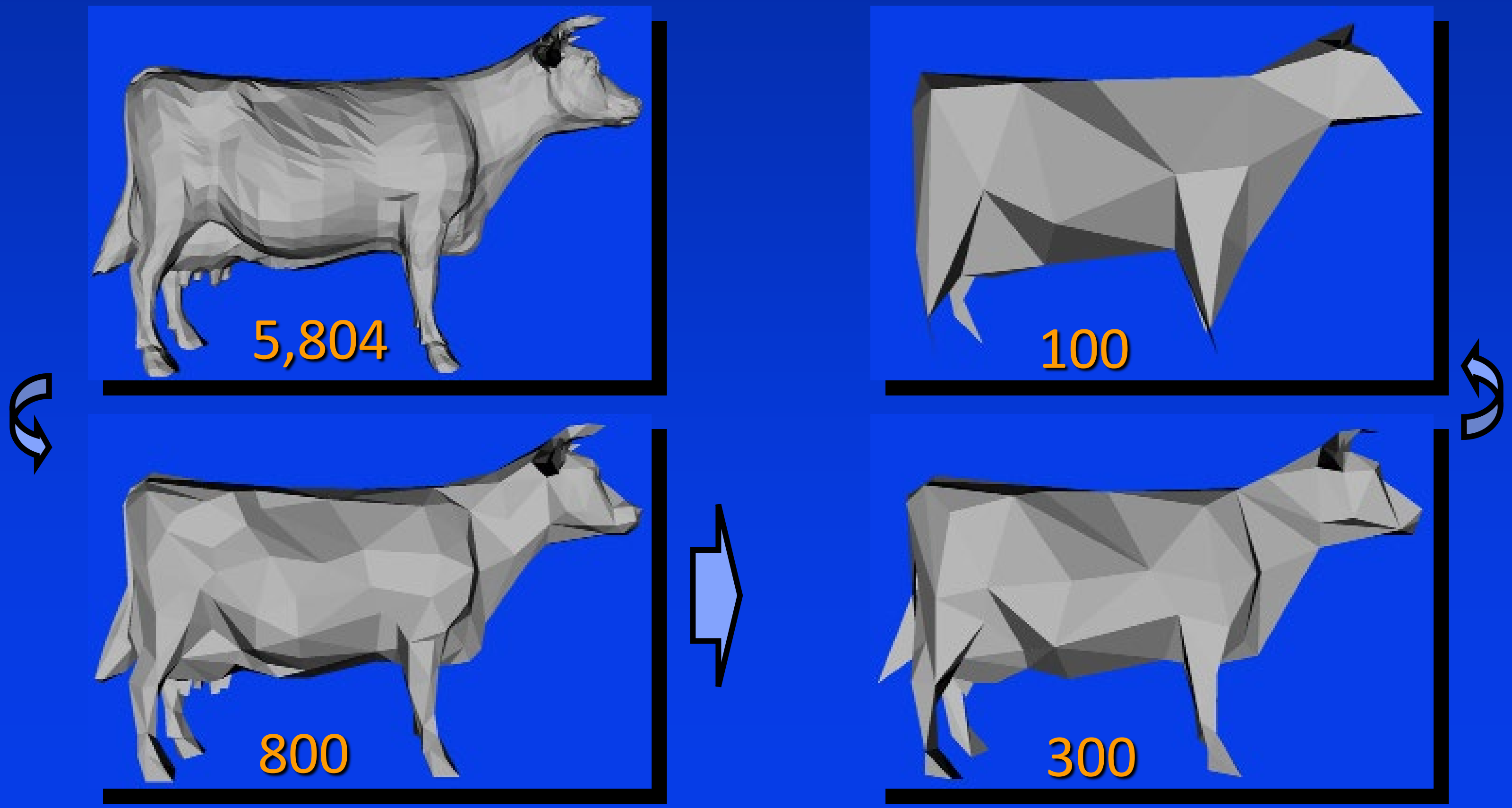
Our Basic Operation: Vertex Pair (Edge) Contraction

Contract vertex pair $(v_1,v_2) \rightarrow v'$

- Move v_1 and v_2 to position v'
- Replace all occurrences of v_2 with v_1
- Remove v_2 and degenerate triangles
- Typically, we contract edges, as others have done



Simplifying a Cow in Under One Second



How We Measure Error

Measure error at current vertices

For a given point v, measure sum of squared distances to associated set of planes

- Each vertex v has an associated set of planes
  - *Initialize with planes of incident faces in original*
  - *Merge sets when contracting pairs*
  - *Initial error of each vertex is 0*

Measuring Error With Quadrics

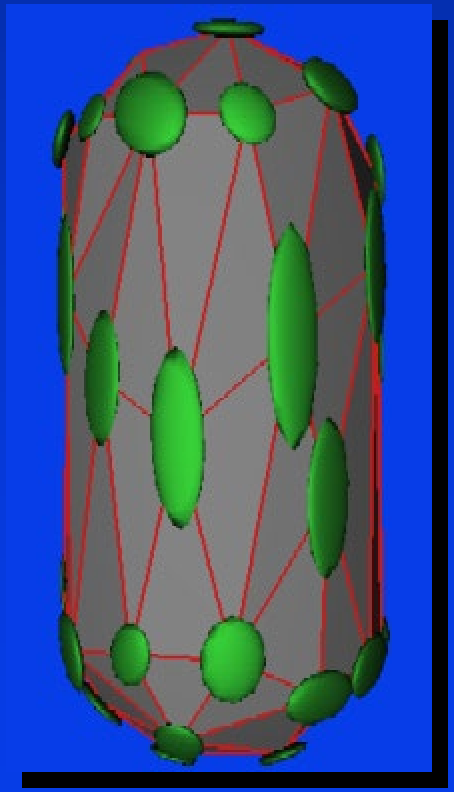
Sum of squared distance to a set of planes

- Vertex v has associated set of planes
- Planes defined by $ax+by+cz+d = 0$, $a^2+b^2+c^2=1$

$$\boxed{\mathrm{Error}(v) = v^T \left(\sum K_p\right) v}$$

- Each plane p defines a quadric matrix K_p
- Set of planes represented by sum of quadrics

But What Are These Quadrics Really Doing?



Almost always ellipsoids

- When Q is positive definite

Characterize error at vertex

- Vertex at center of each ellipsoid
- Move it anywhere on ellipsoid with constant error

Capture local shape of surface

- Stretch in least curved direction

Algorithm Outline

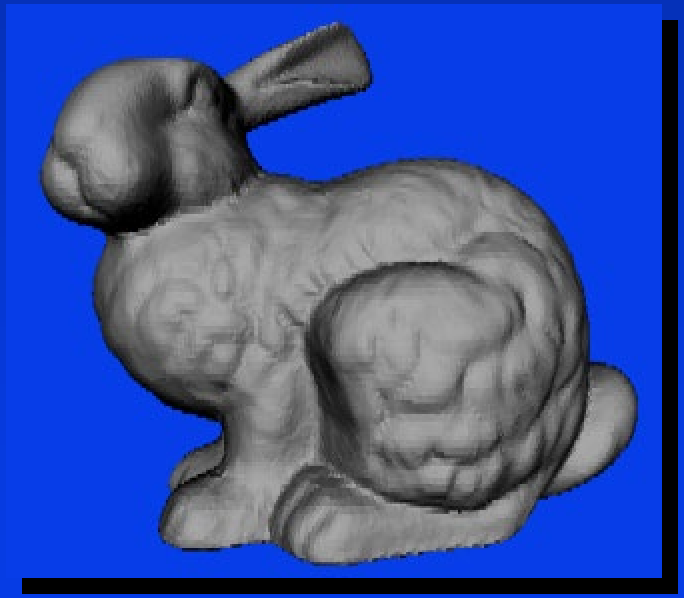
Initialization

- Compute quadric Q for each vertex
- Select set of valid vertex pairs (edges + non-edges)
- Compute minimal cost candidate for each pair

Iteration

- Select lowest cost pair (v_1,v_2)
- Contract (v_1,v_2) — Q for new vertex is Q_1+Q_2
- Update all pairs involving v_1 & v_2

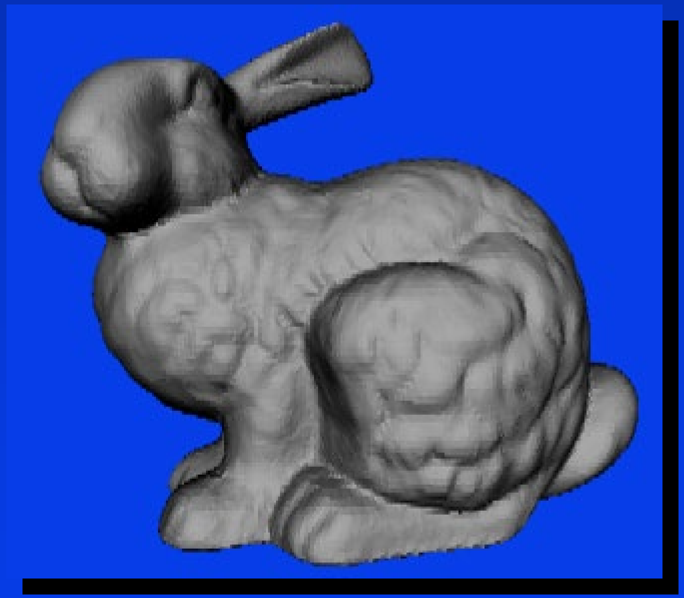
Sample Model: Stanford Bunny



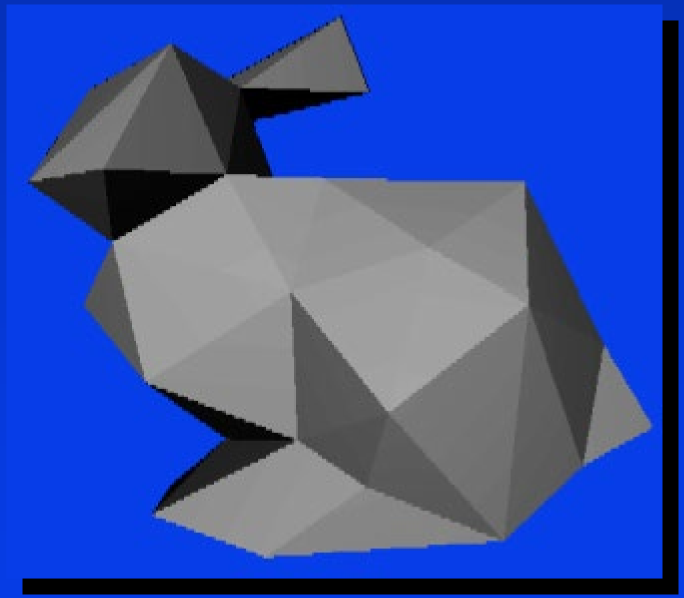
69,451 faces

1,000 faces (30 sec)

Sample Model: Stanford Bunny



69,451 faces



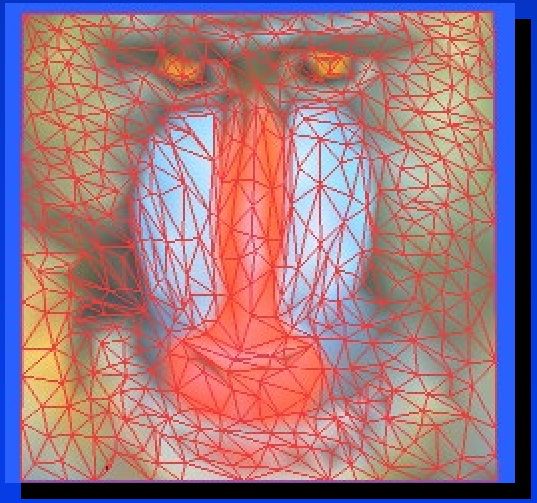
100 faces (30 sec)

Handling Surfaces With Colored Vertices

Before: v = (x,y,z,1) and Q is a 4x4 matrix

Now: v = (x,y,z,r,g,b,1) and Q is a 7x7 matrix

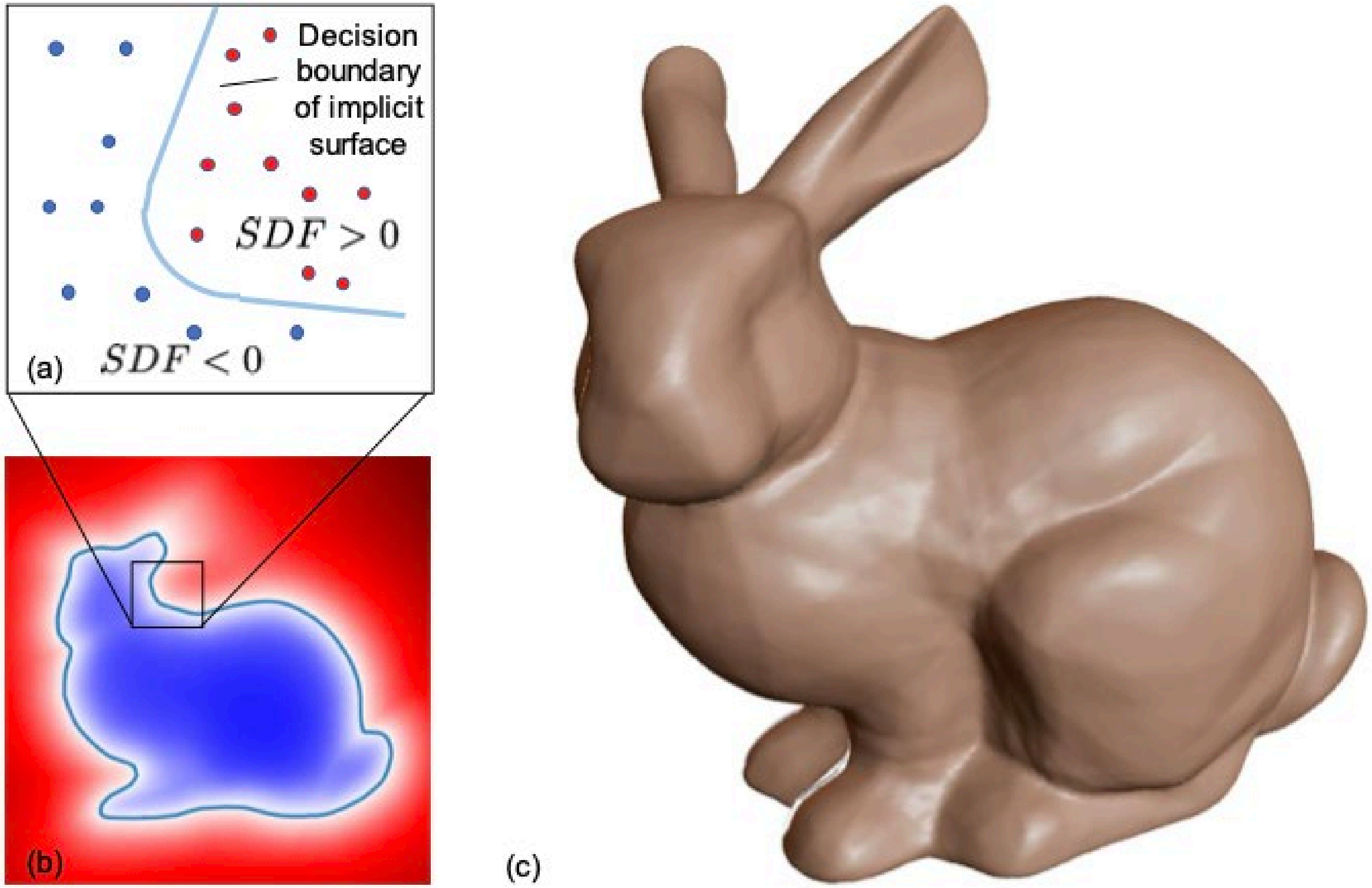
19,404 faces



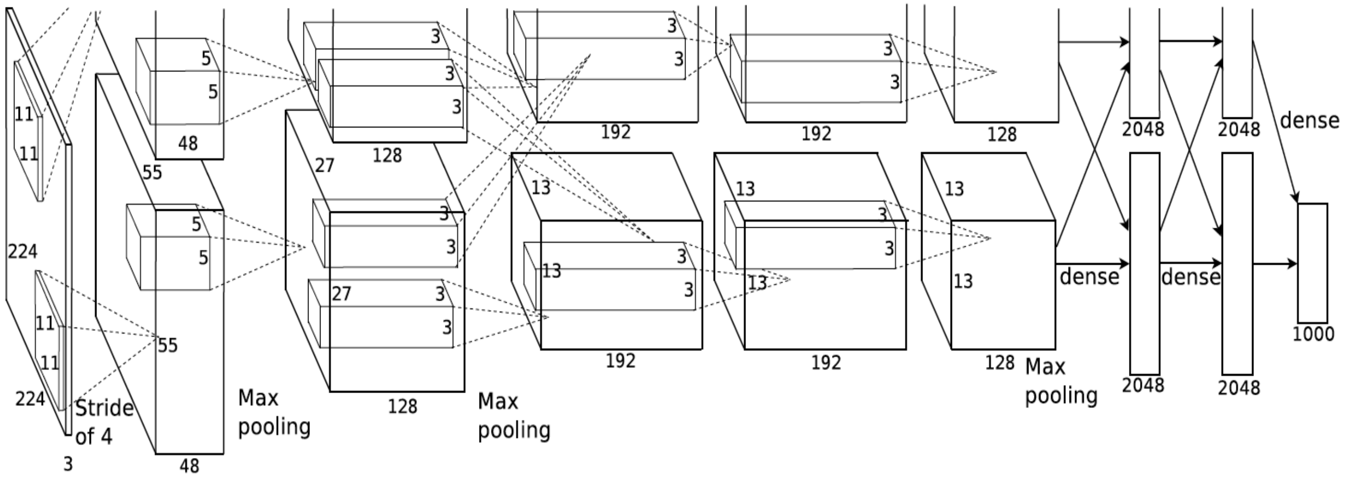
1,000 face approximation

Today:
Neural Implicit Representations,
Class Wrap Up

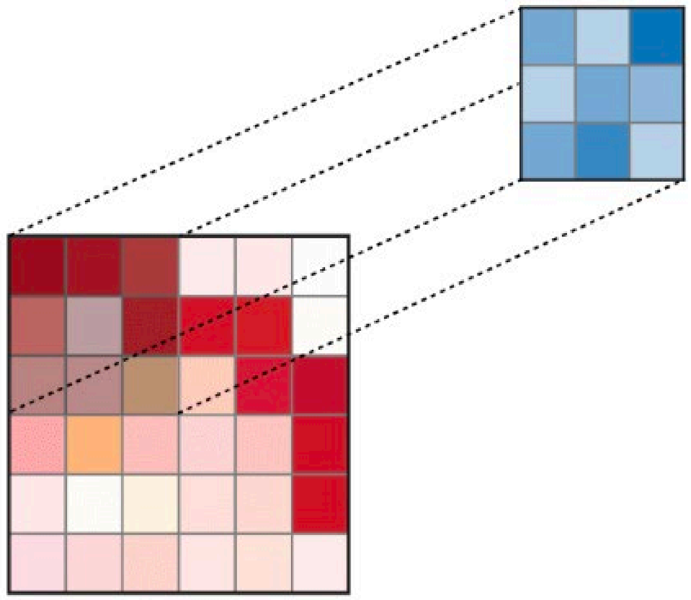
DeepSDF: CVPR 2019



Representations for 2D Deep Learning



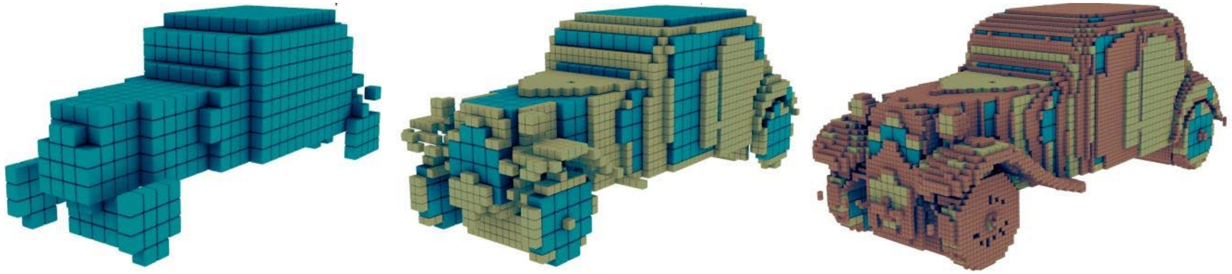
ImageNet. 2012



Convolution Layer

Convolutional Image Encoders

Extended to 3D Voxel Grids

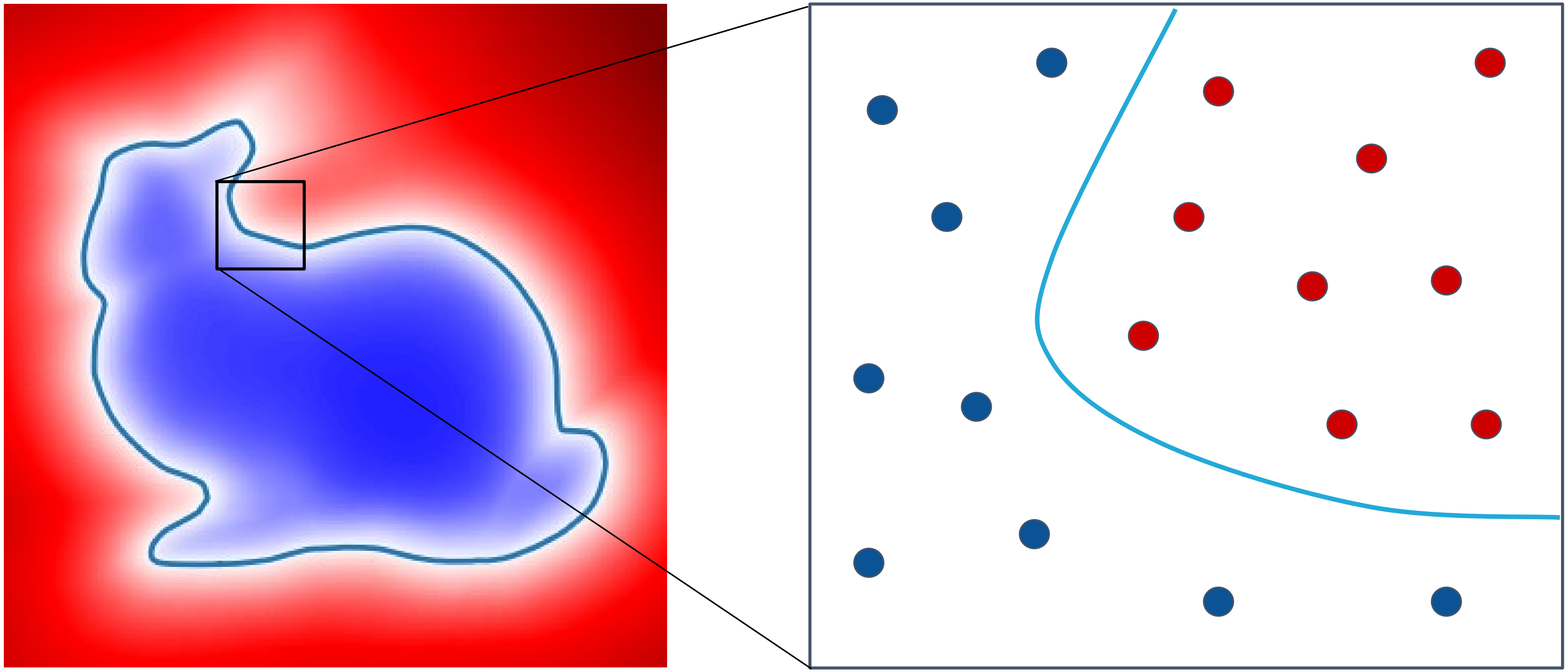


Dai et al. 2017

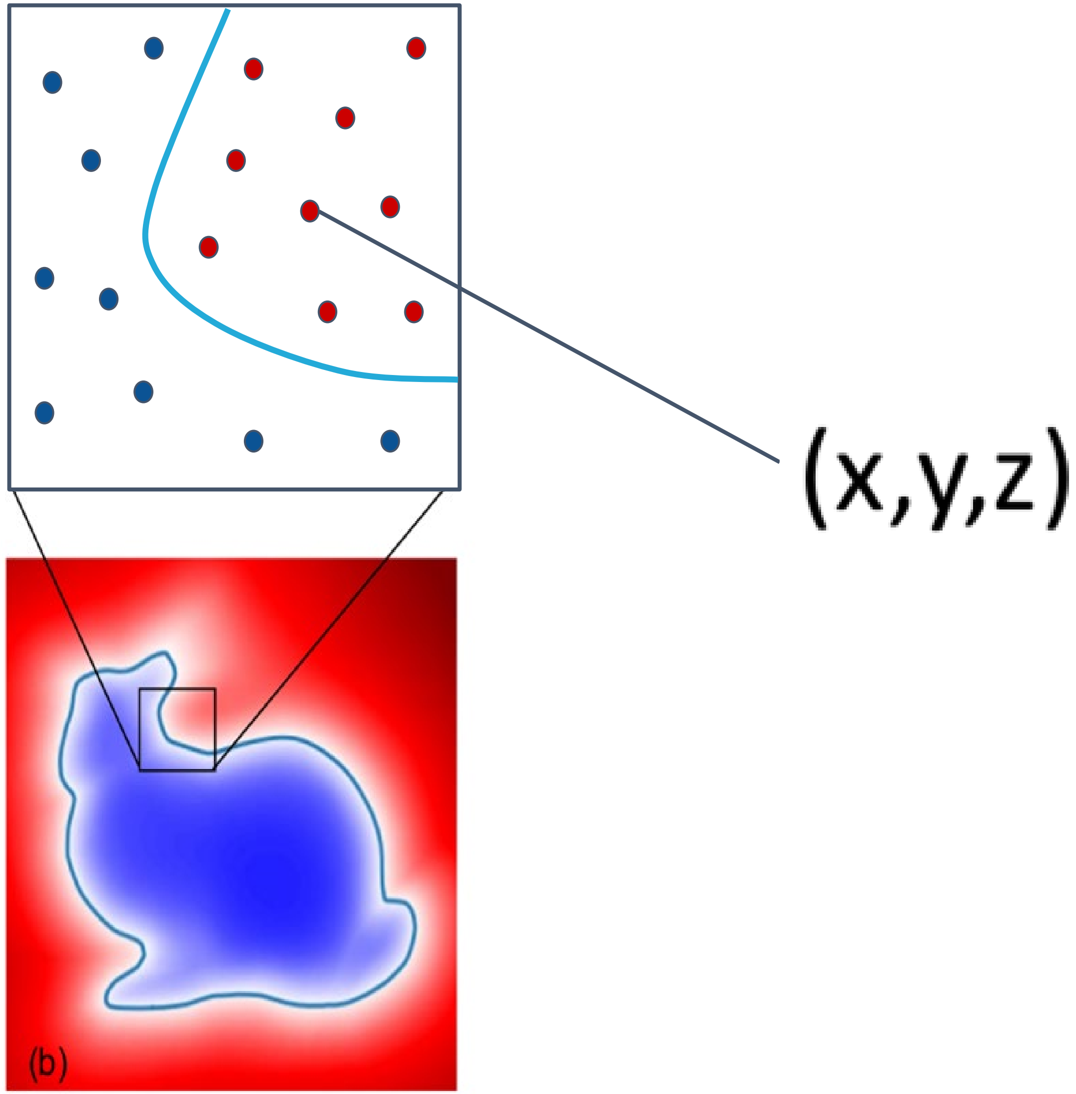
Wu et al. 2016

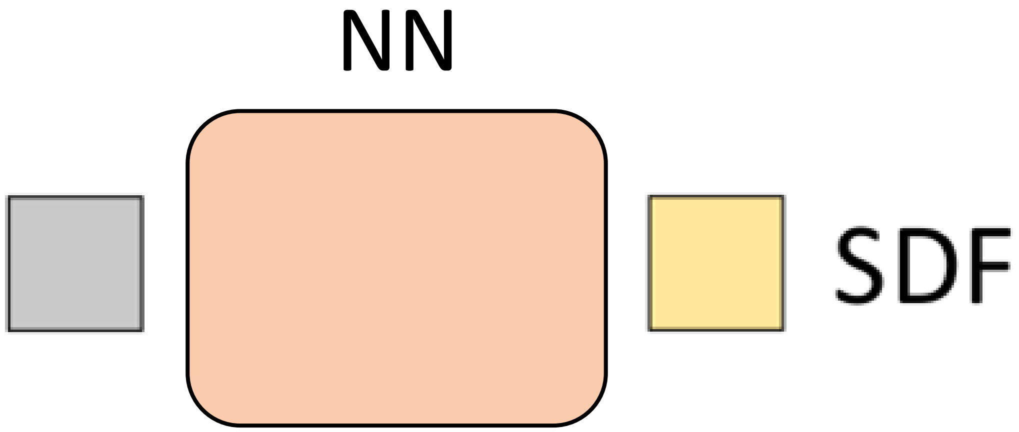
Tatarchenko et al. 2017

Surfaces as Decision Boundary



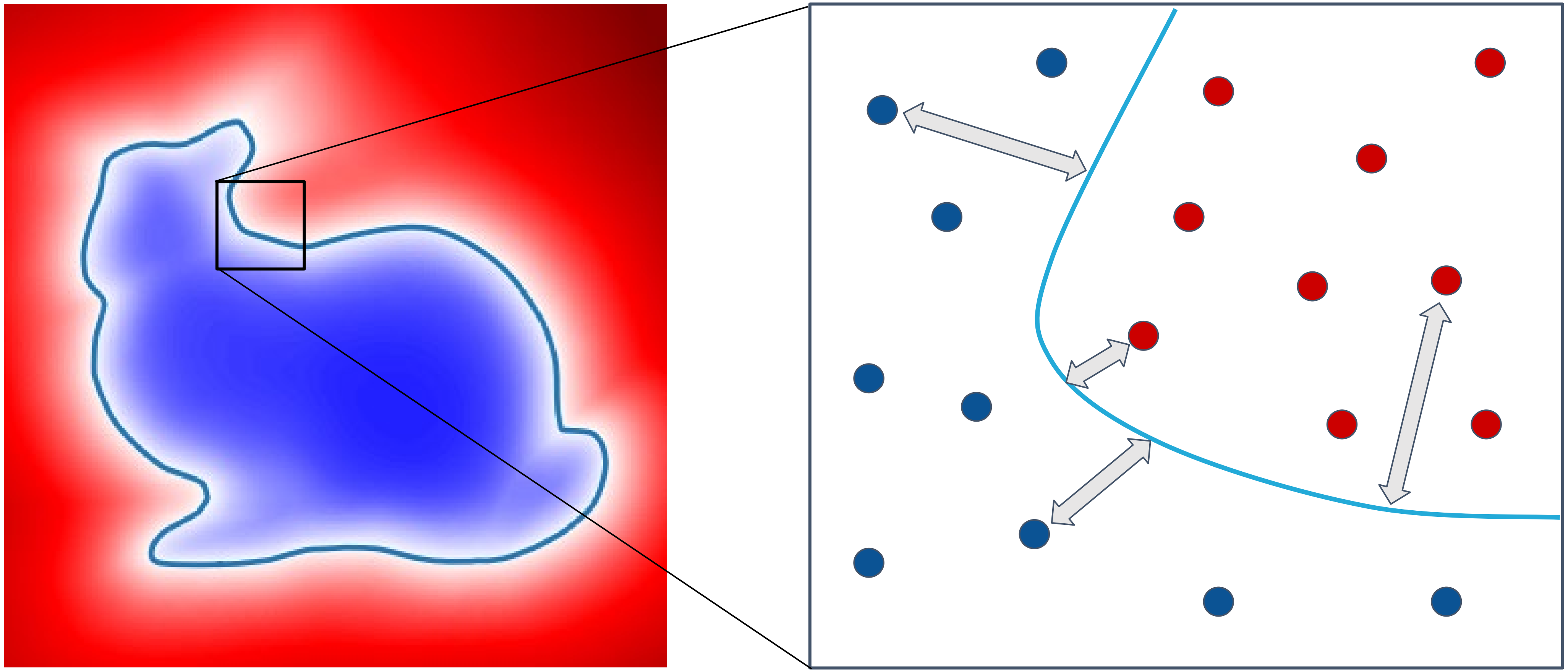
Regression of Continuous SDF by a NN



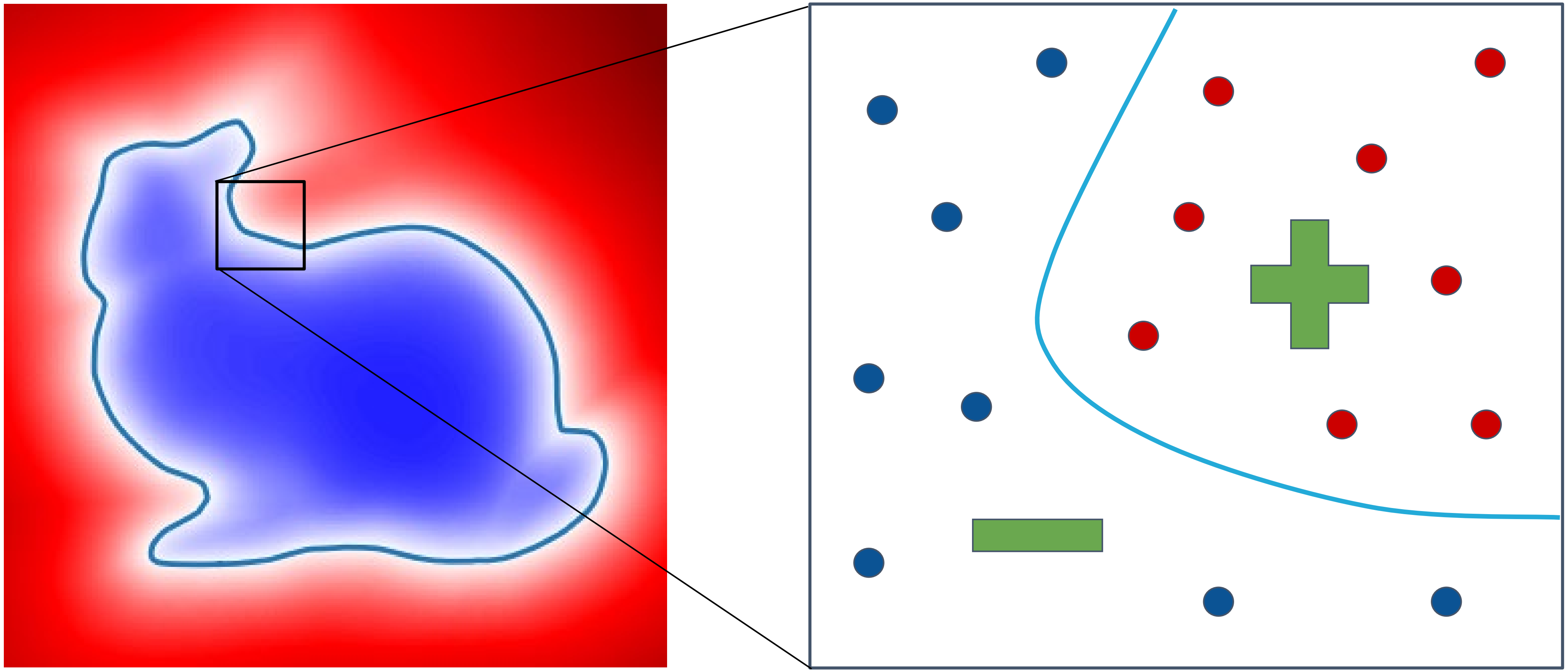


Signed Distance Functions

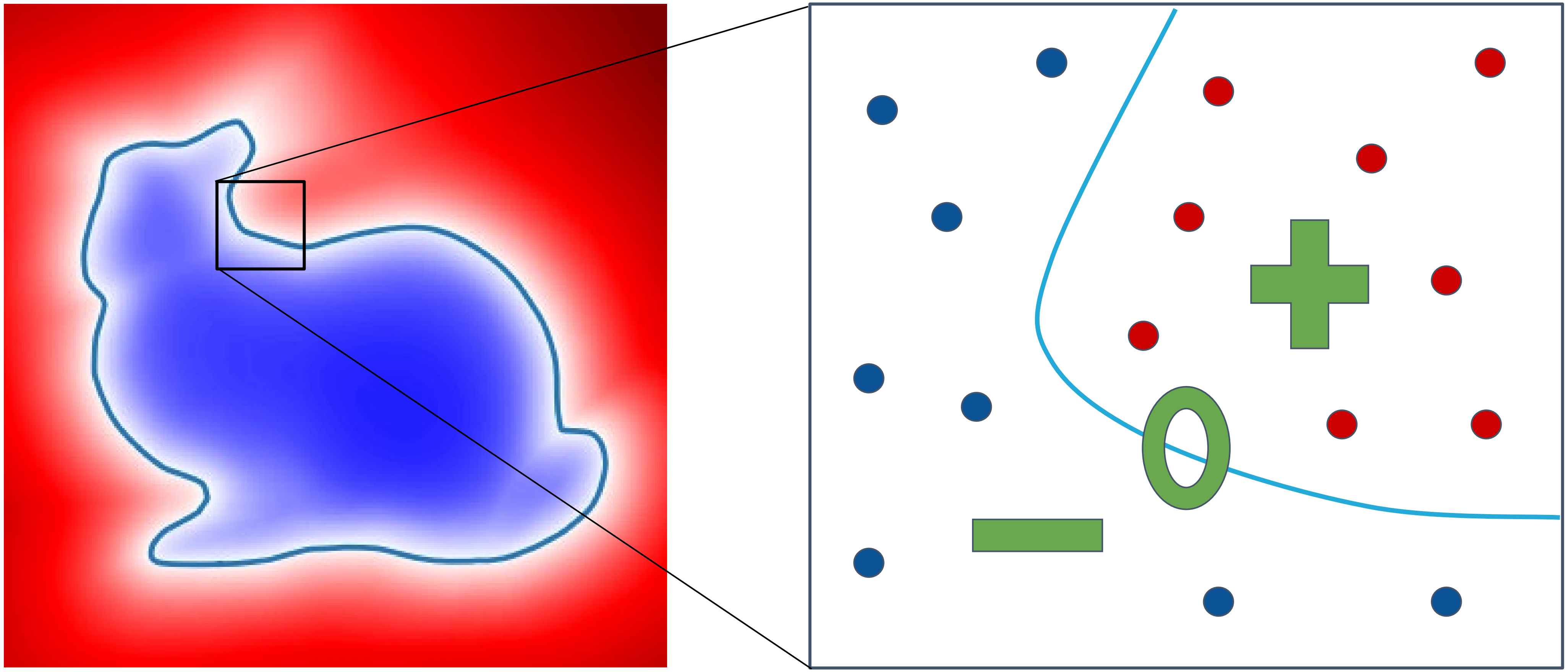
Signed Distance Function



Signed Distance Function



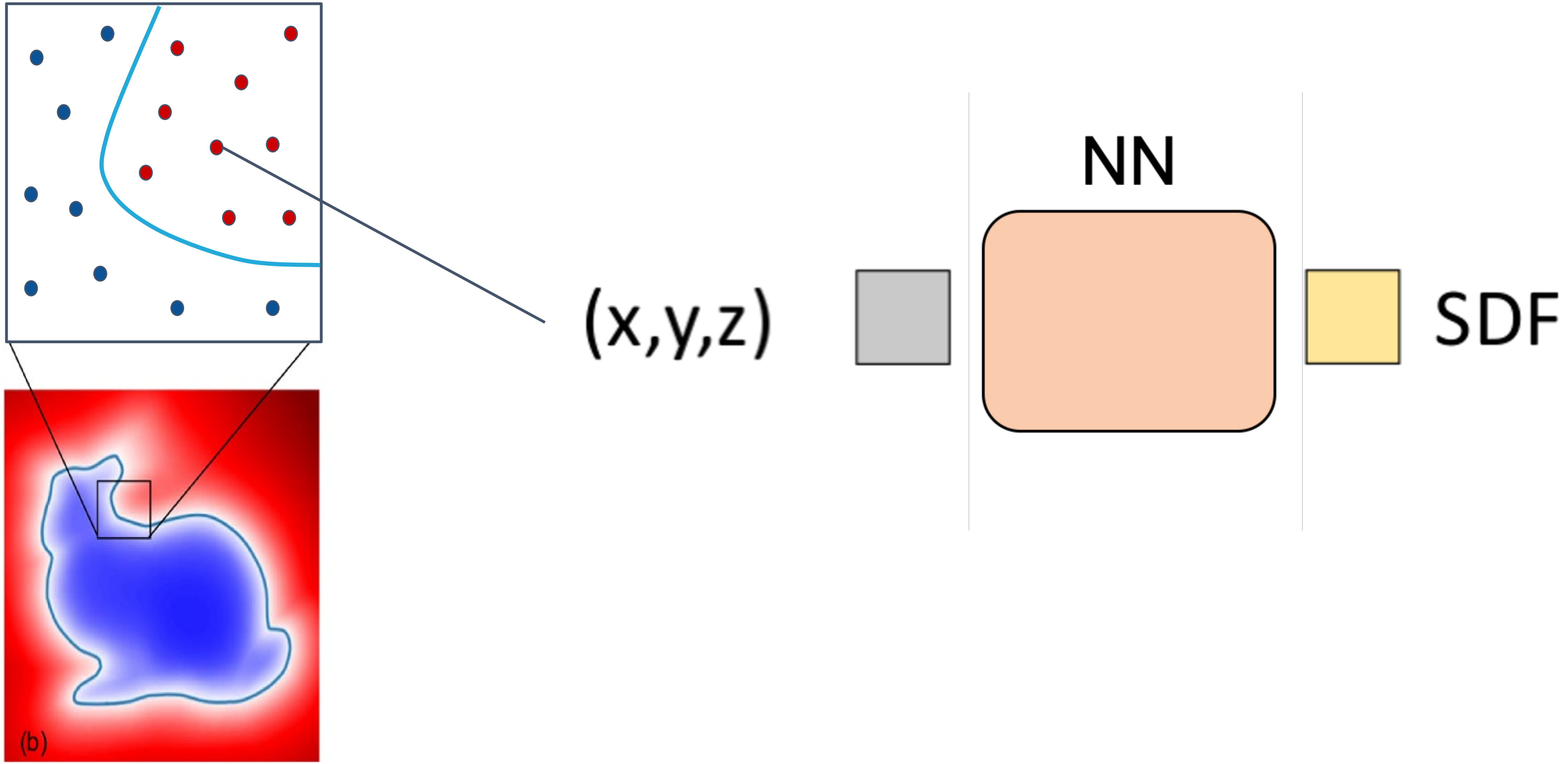
Signed Distance Function



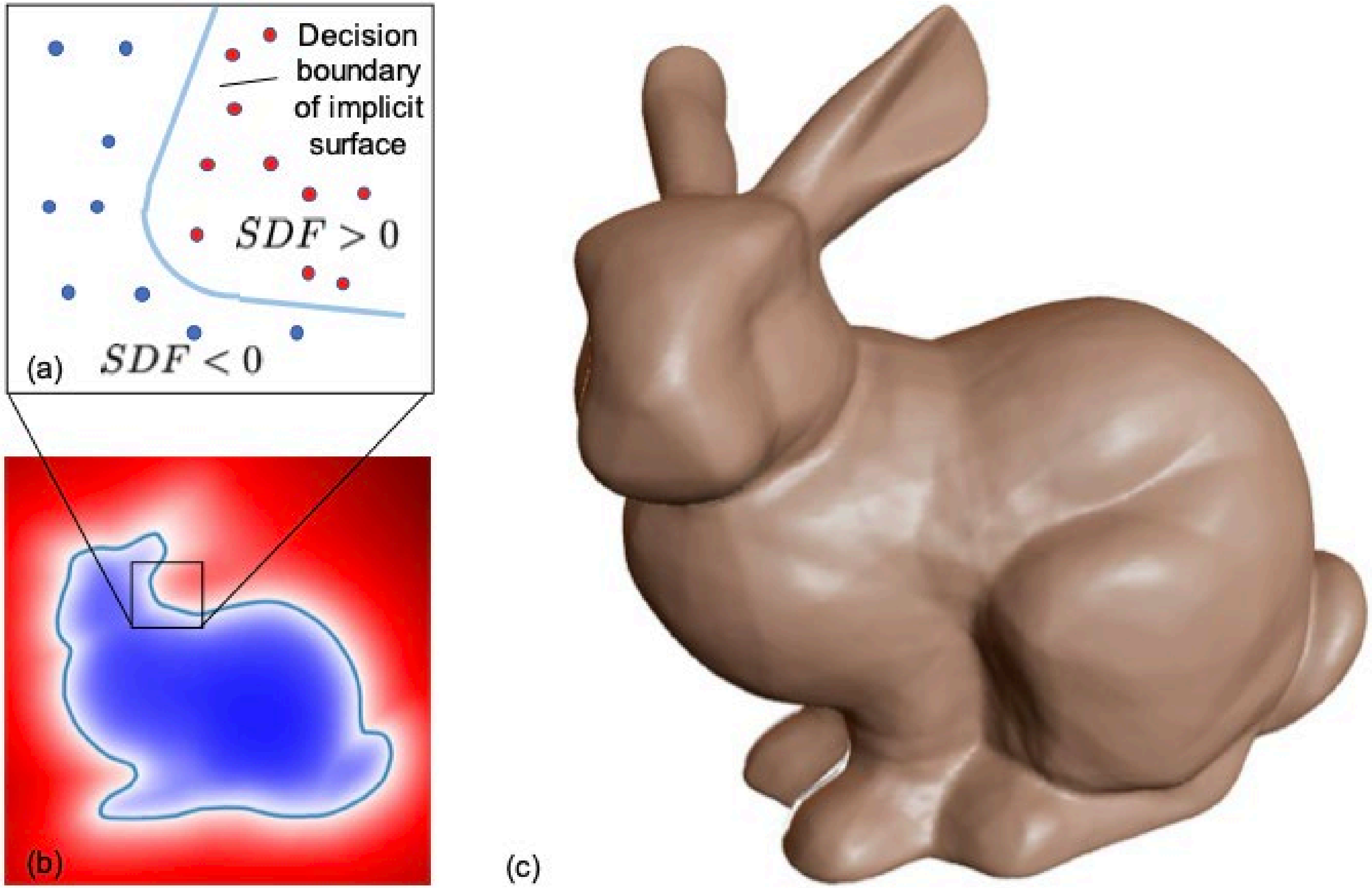
Discretized SDFs

-0.9	-0.3	0.0	0.2	1	1	1	1	1
-1	-0.9	-0.2	0.0	0.2	1	1	1	1
-1	-0.9	-0.3	0.0	0.1	0.9	1	1	1
-1	-0.8	-0.3	0.0	0.2	0.8	1	1	1
-1	-0.9	-0.4	-0.1	0.1	0.8	0.9	1	1
-1	-0.7	-0.3	0.0	0.3	0.6	1	1	1
-1	-0.7	-0.4	0.0	0.2	0.7	0.8	1	1
-0.9	-0.7	-0.2	0.0	0.2	0.8	0.9	1	1
-0.1	0.0	0.0	0.1	0.3	1	1	1	1
0.5	0.3	0.2	0.4	0.8	1	1	1	1

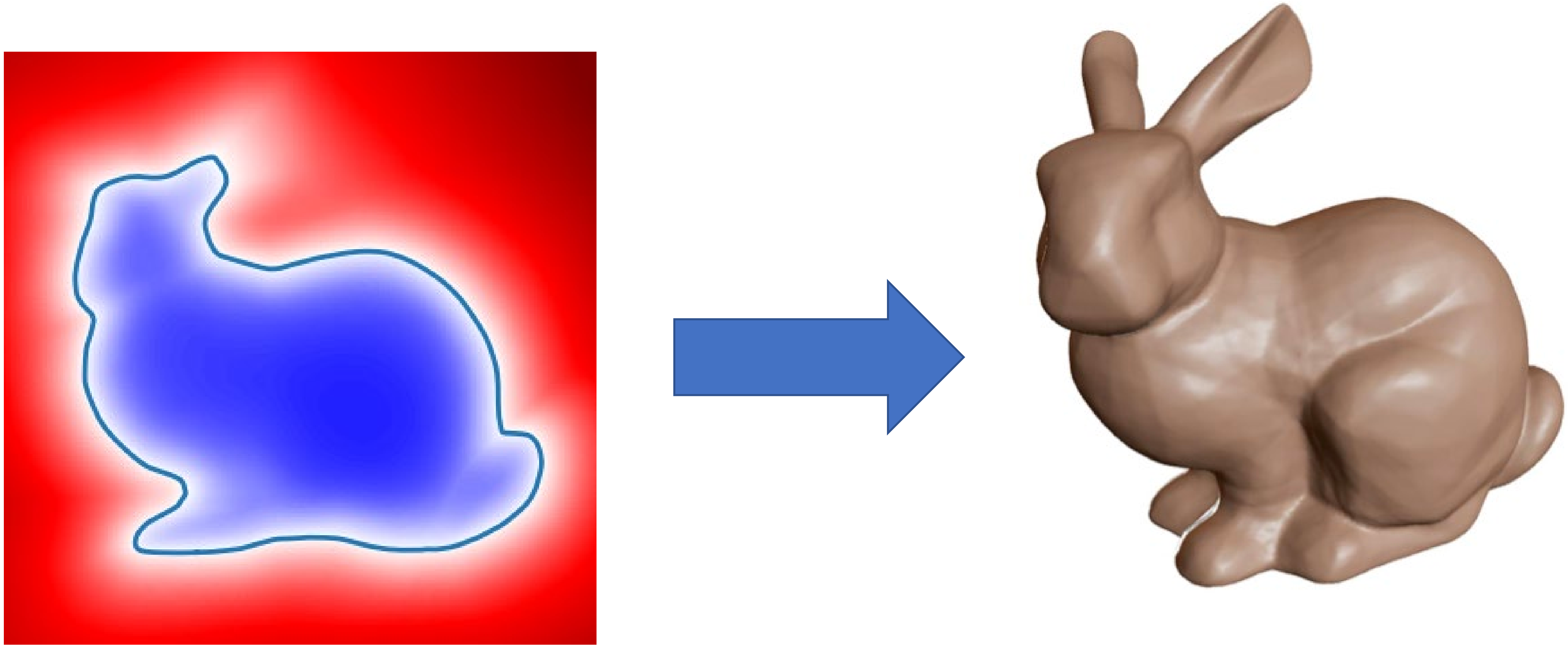
Regression of Continuous SDF



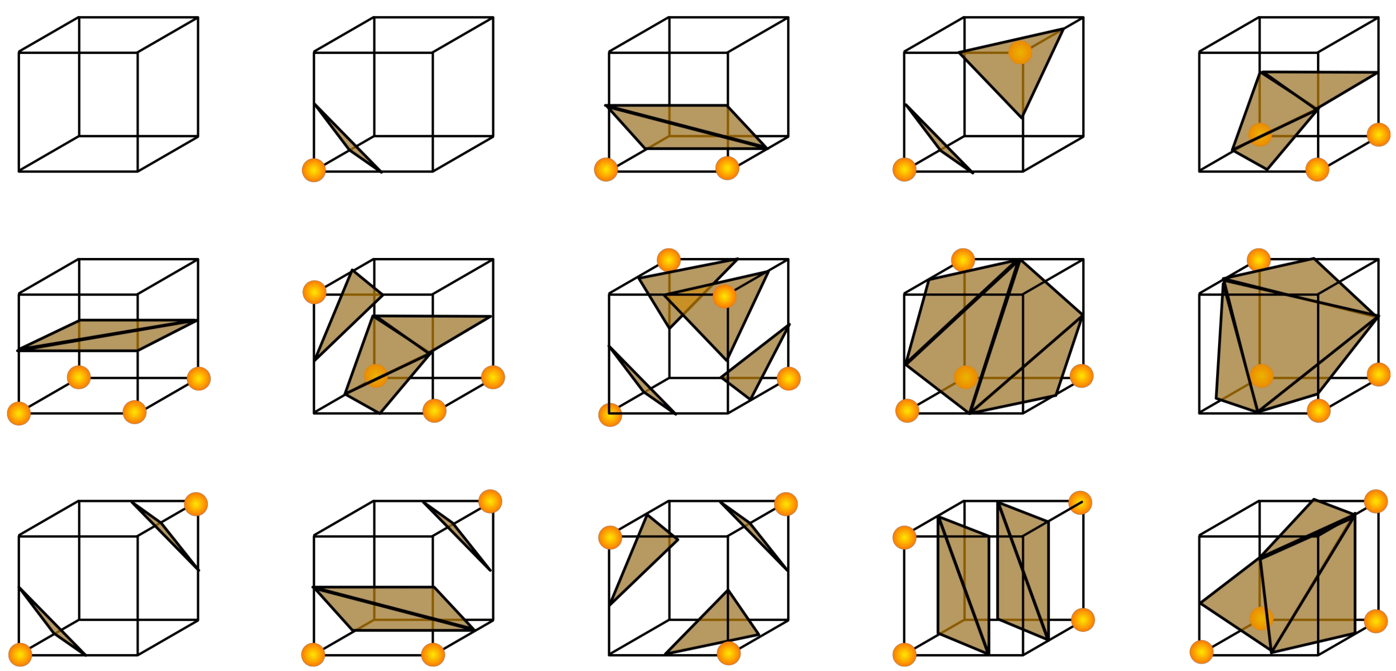
Universal Approximation of Functions by NNs



Implicit to Explicit

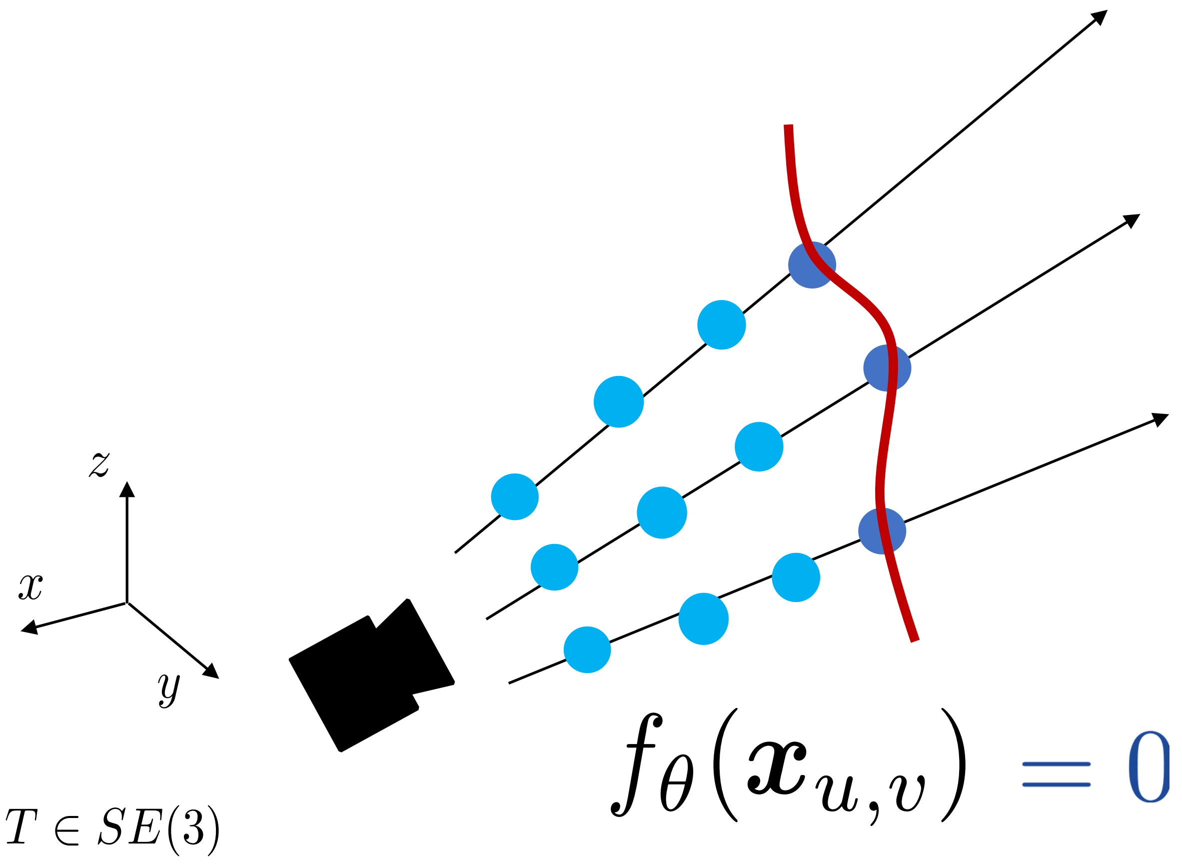


Marching Cubes – Extract Mesh

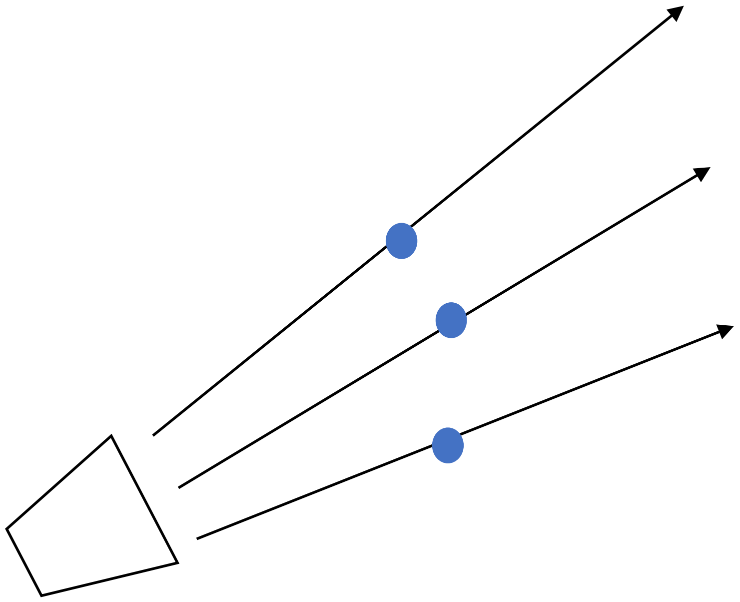


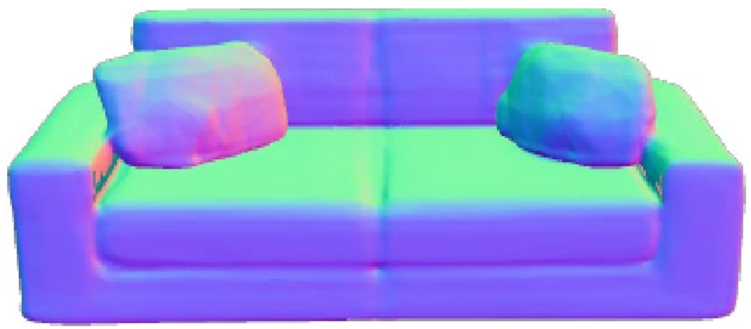
Lorensen et al., 1987

Ray Casting for Rendering

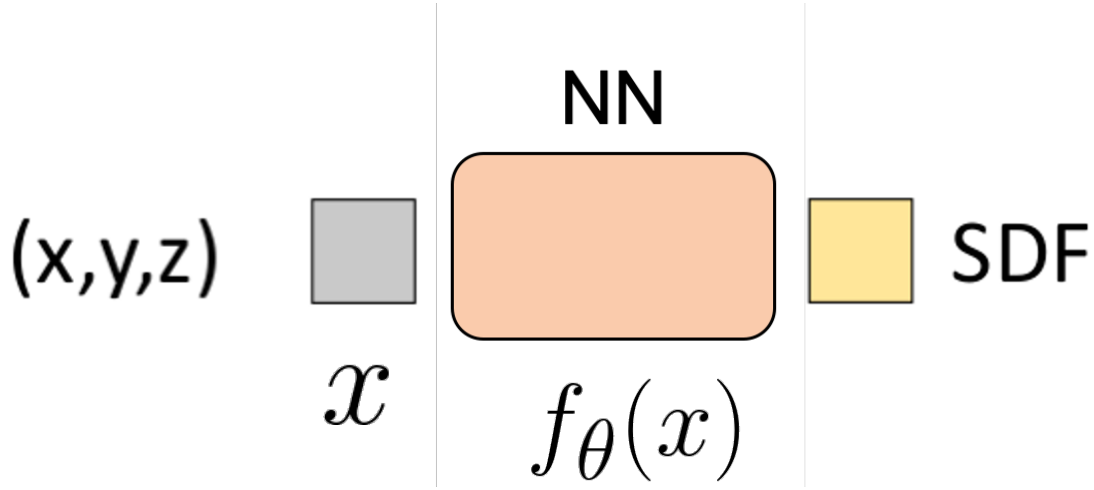


Normals at the Surface via SDF Gradients

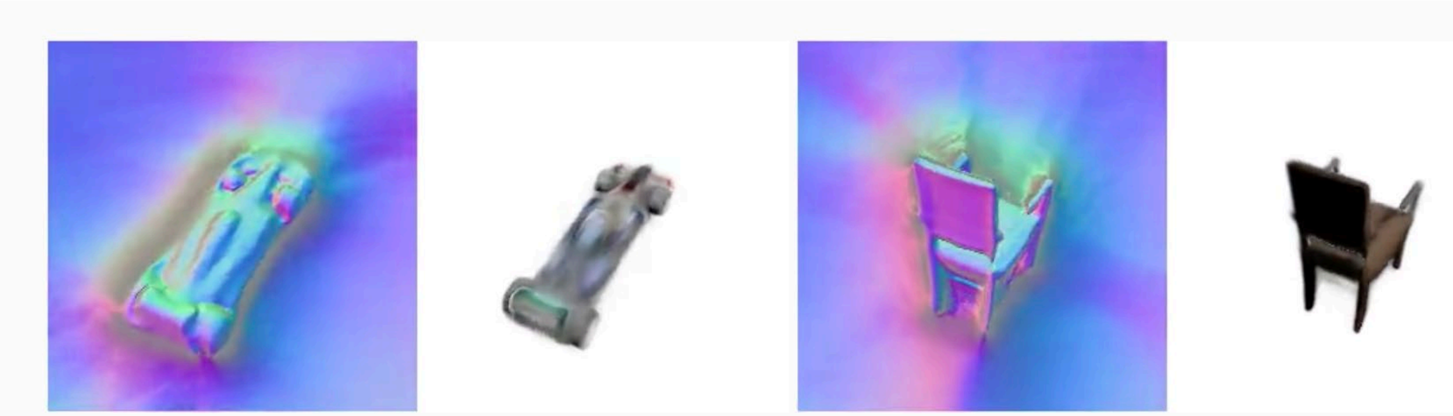




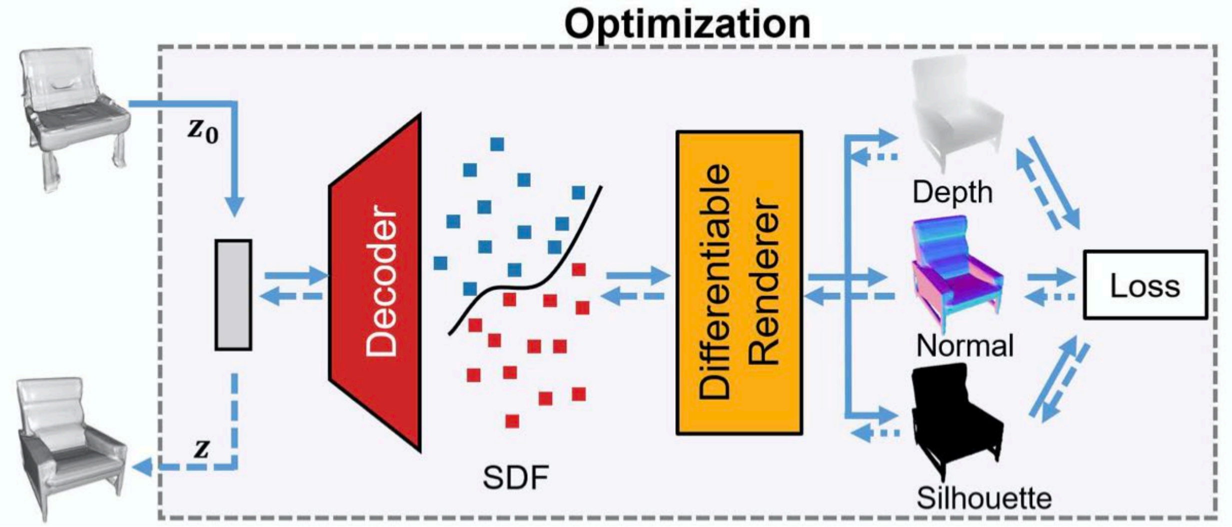
$$\frac{\partial f_\theta(x)}{\partial x}$$



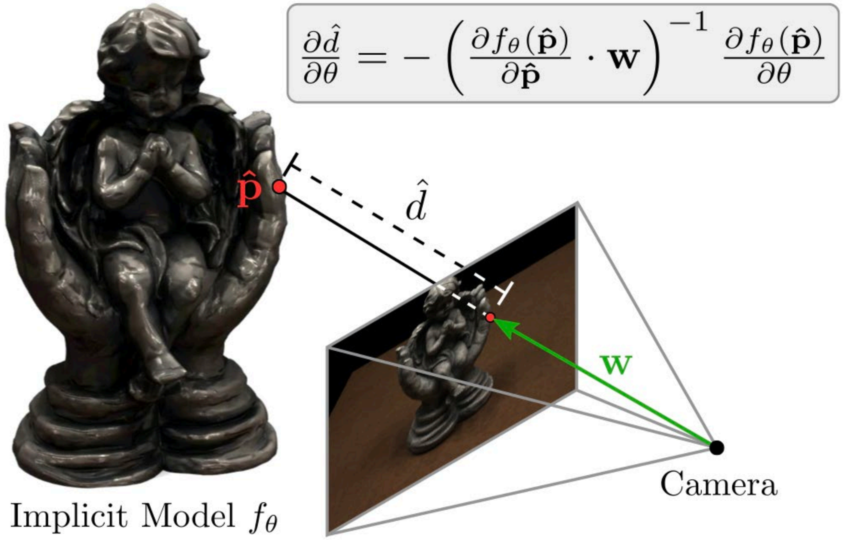
Differentiable Rendering Pipelines



Sitzmann et al. 2019

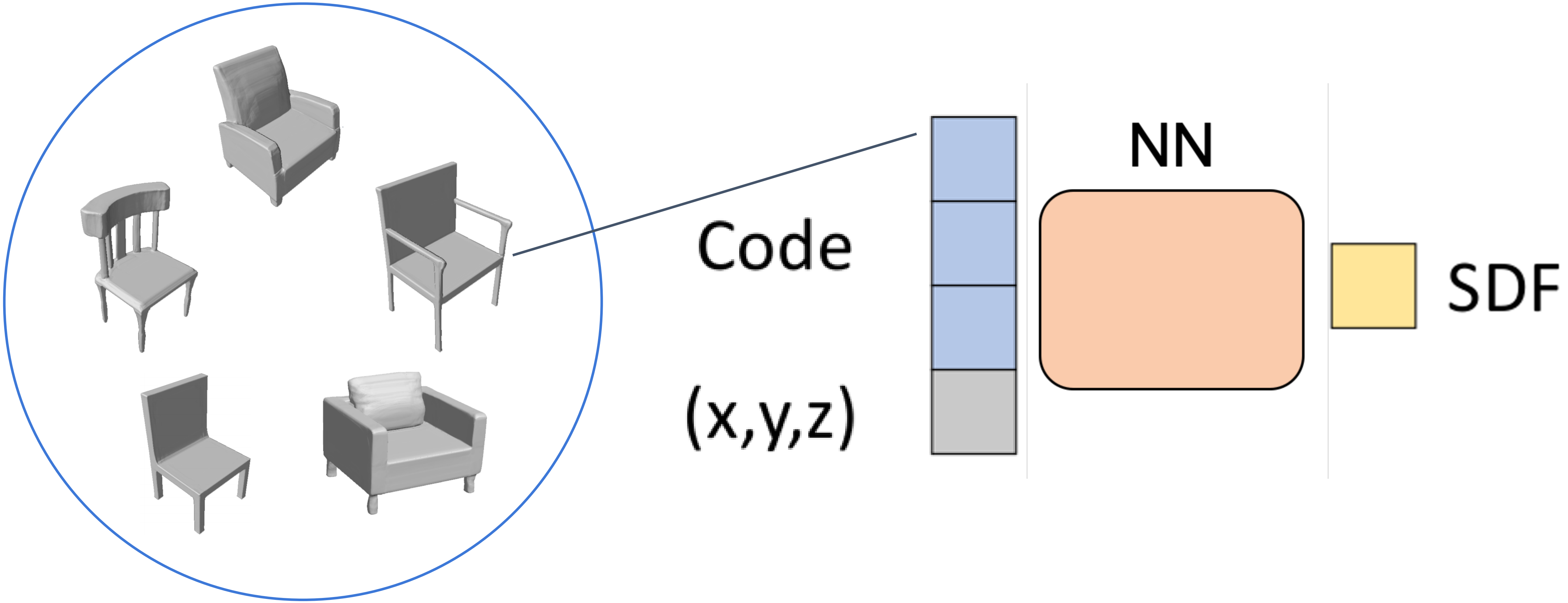


Liu et al. 2019

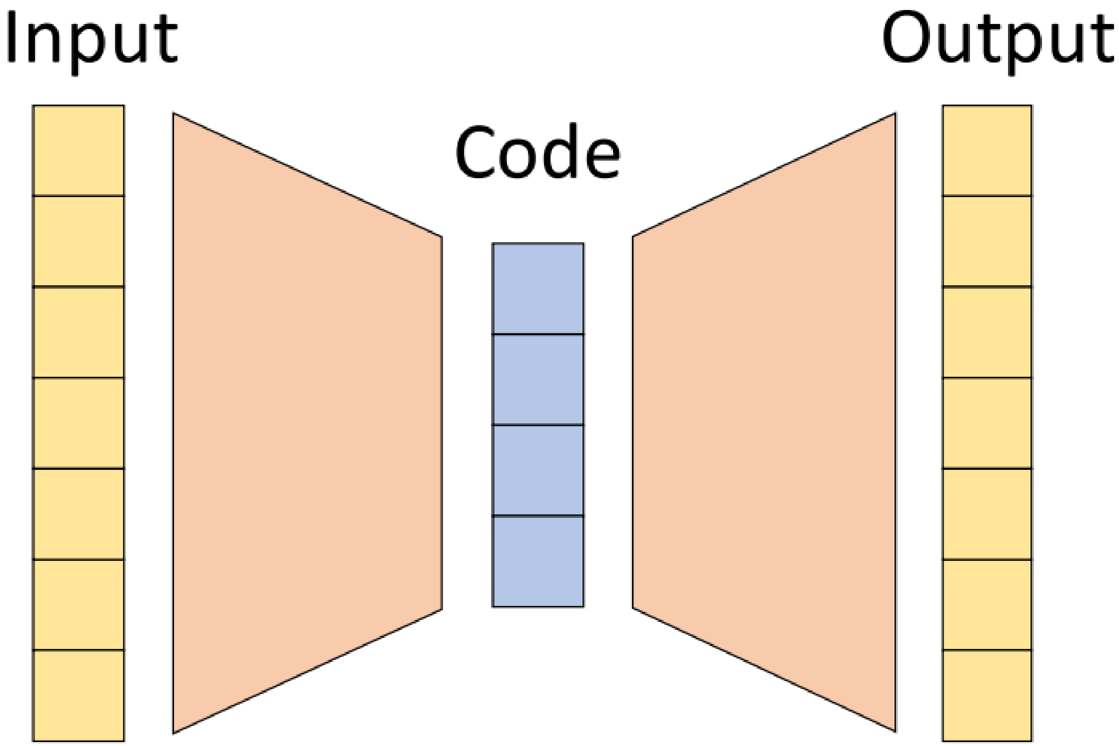


Niemeyer et al. 2020

Coding Multiple Shapes

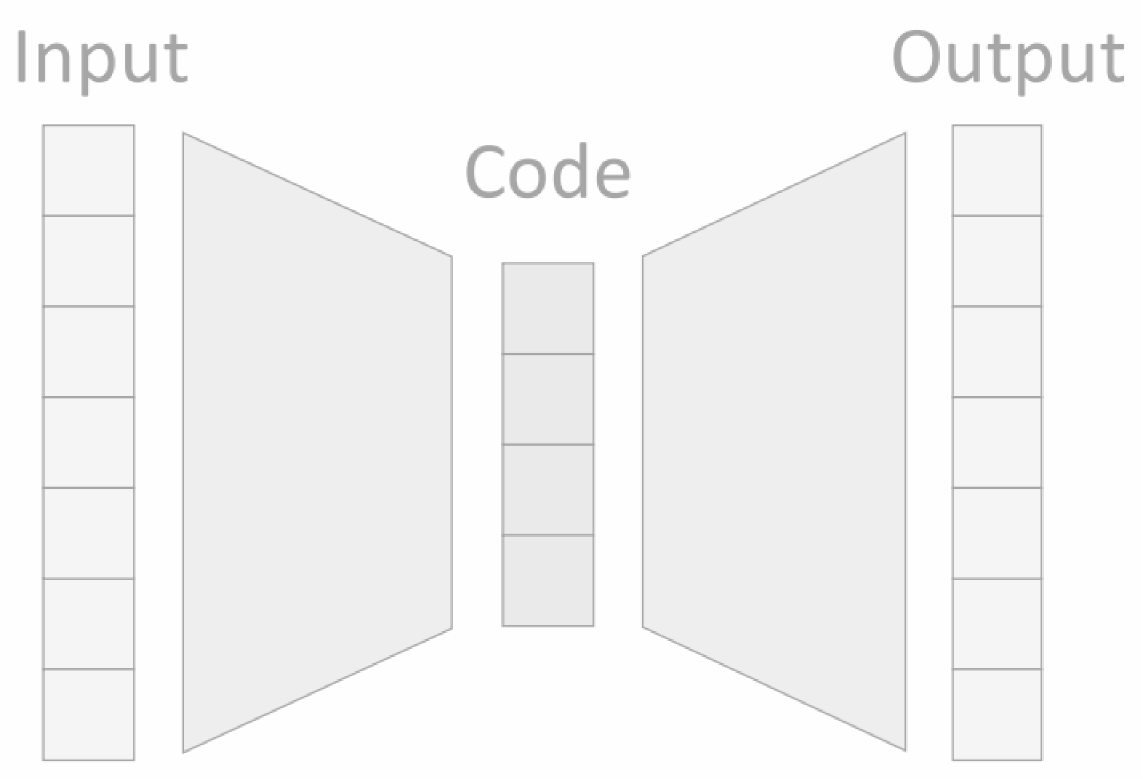


Auto-Encoders

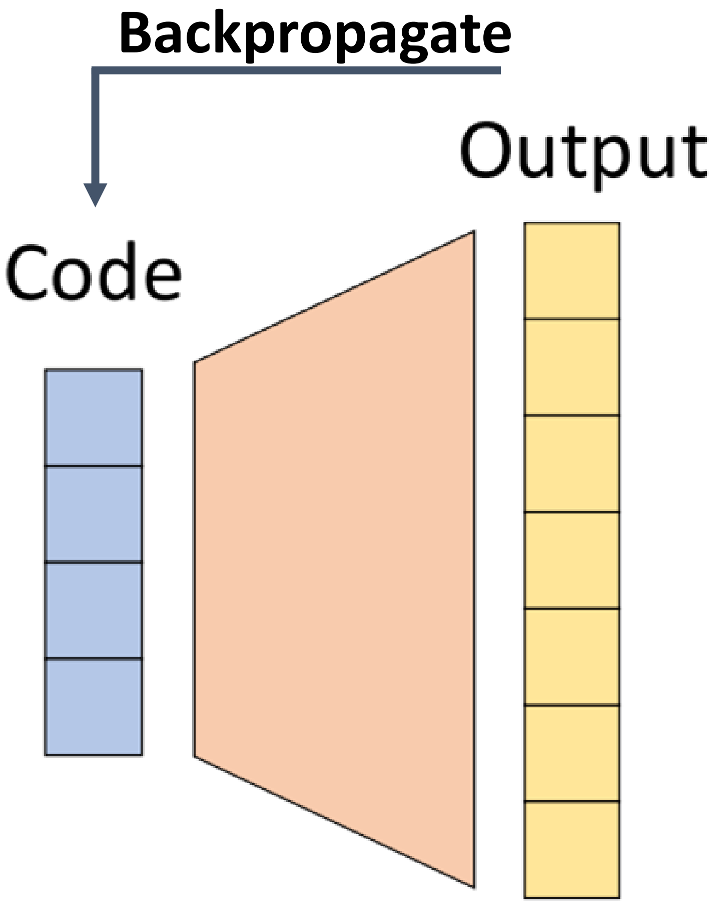


Auto-Encoder

Auto-Decoders

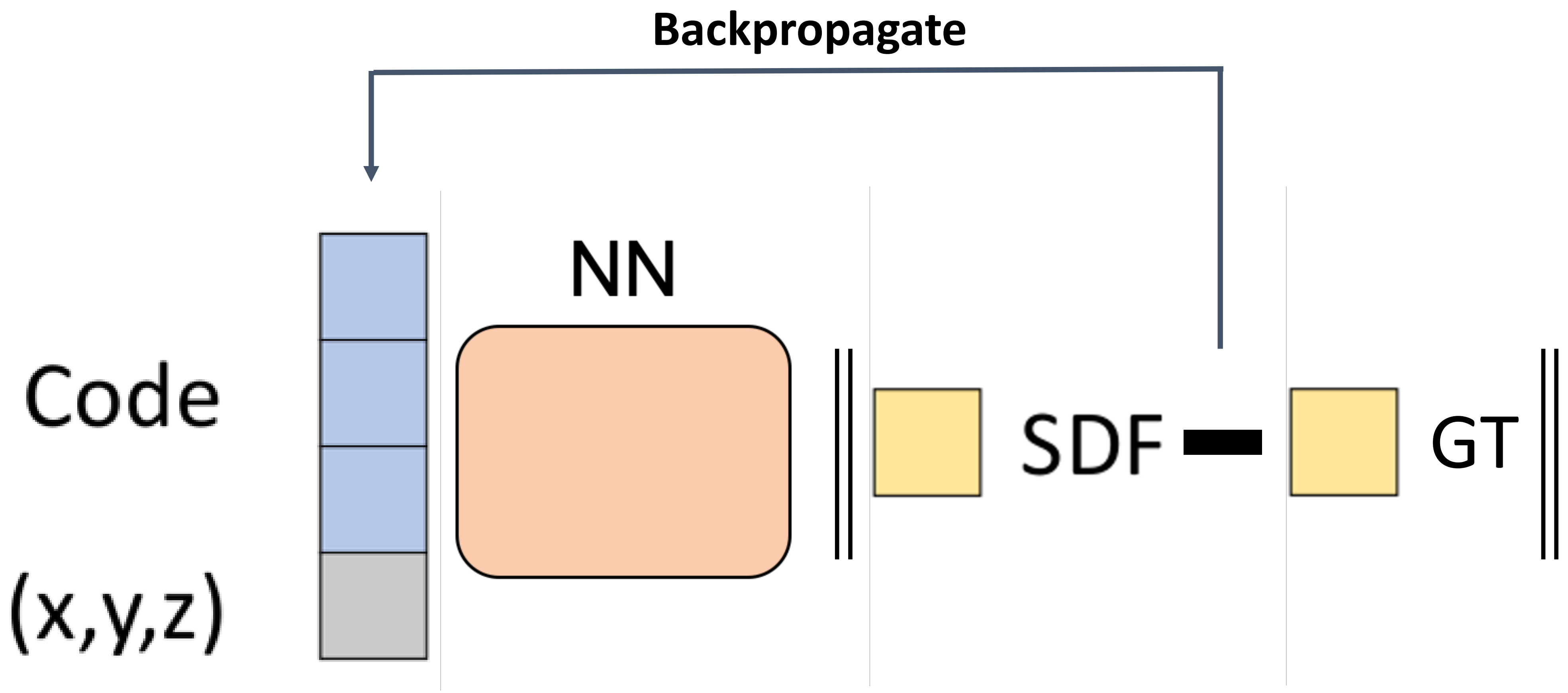


Auto-Encoder

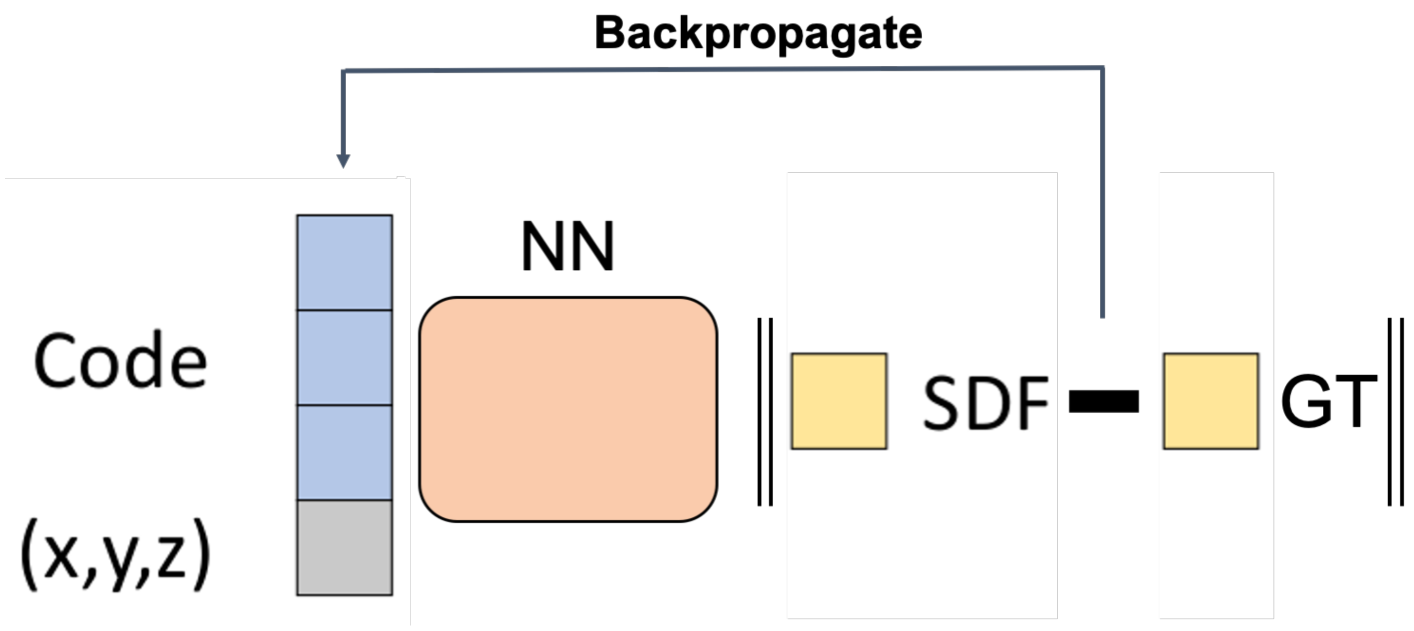


Auto-Decoder

DeepSDF: Auto-Decoder



Advantages of Optimization During Inference

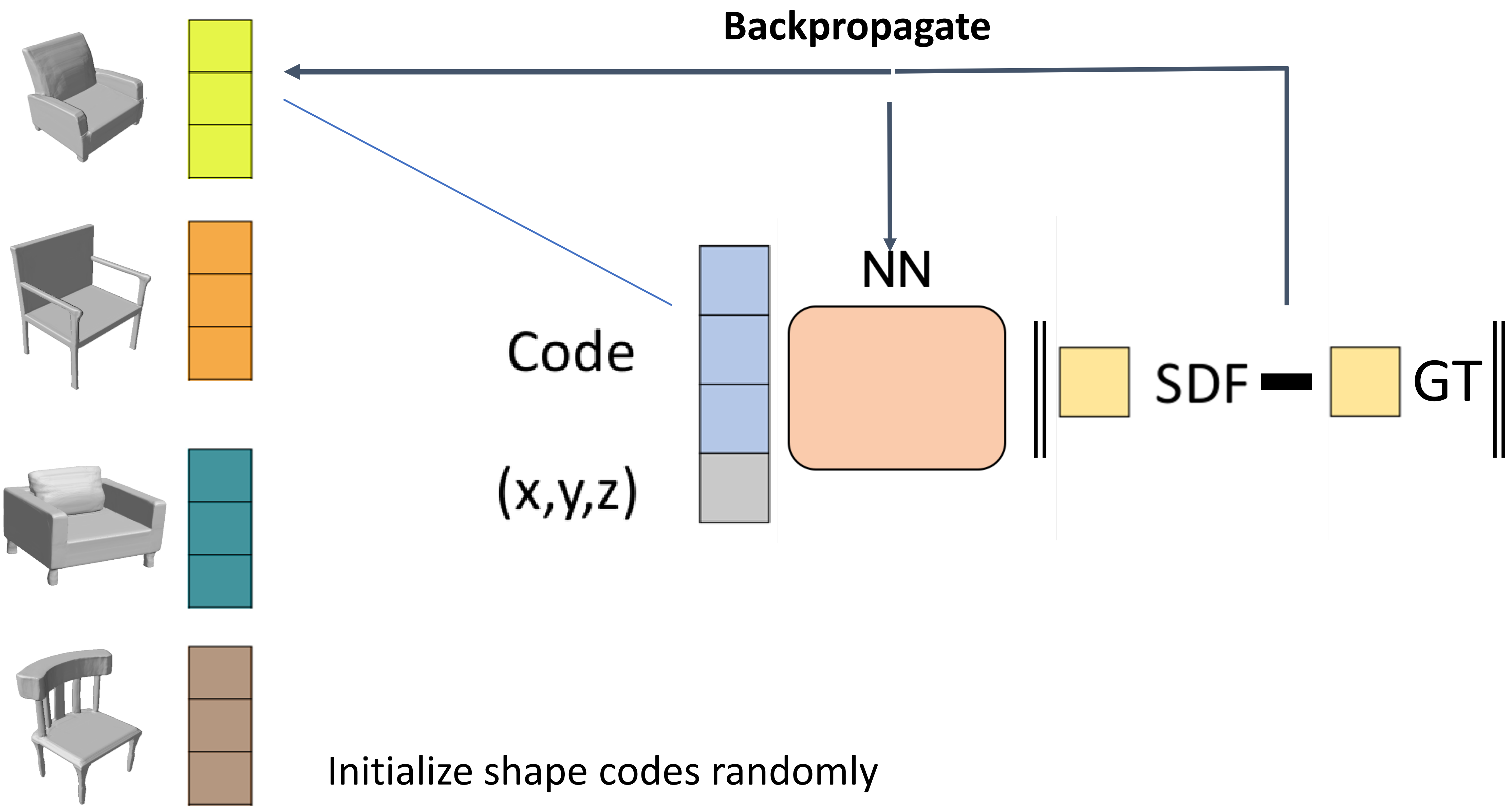


Benefits during Inference

1. Any Number of Observations – Partial

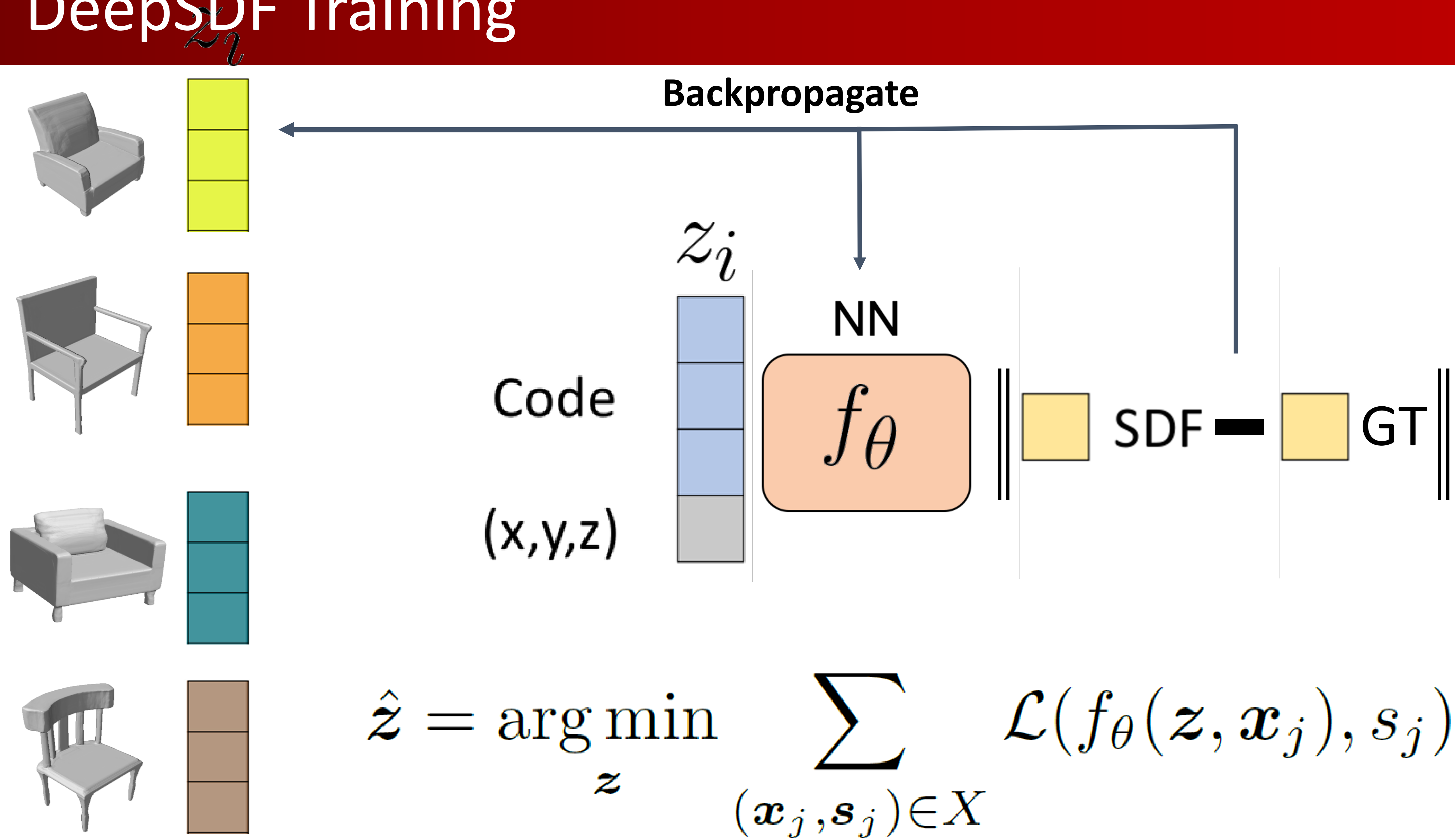
2. More Controlled Inference – e.g. Accuracy, Priors

DeepSDF Training



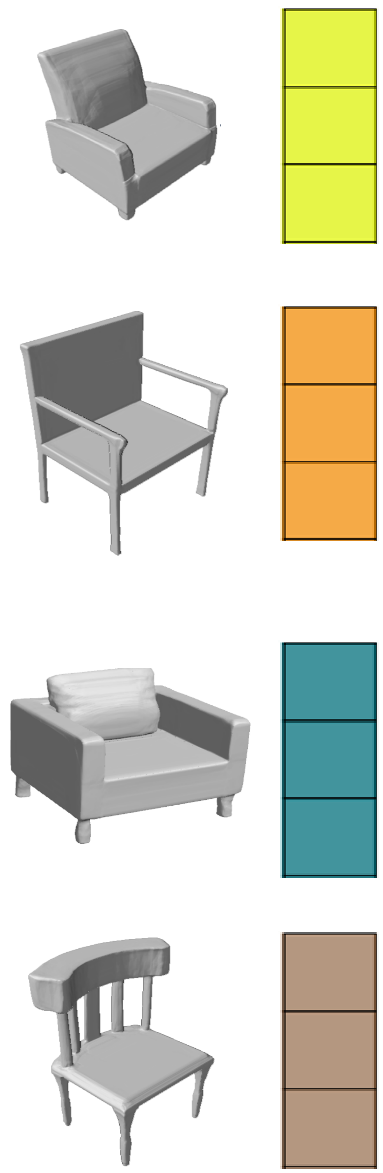
Initialize shape codes randomly

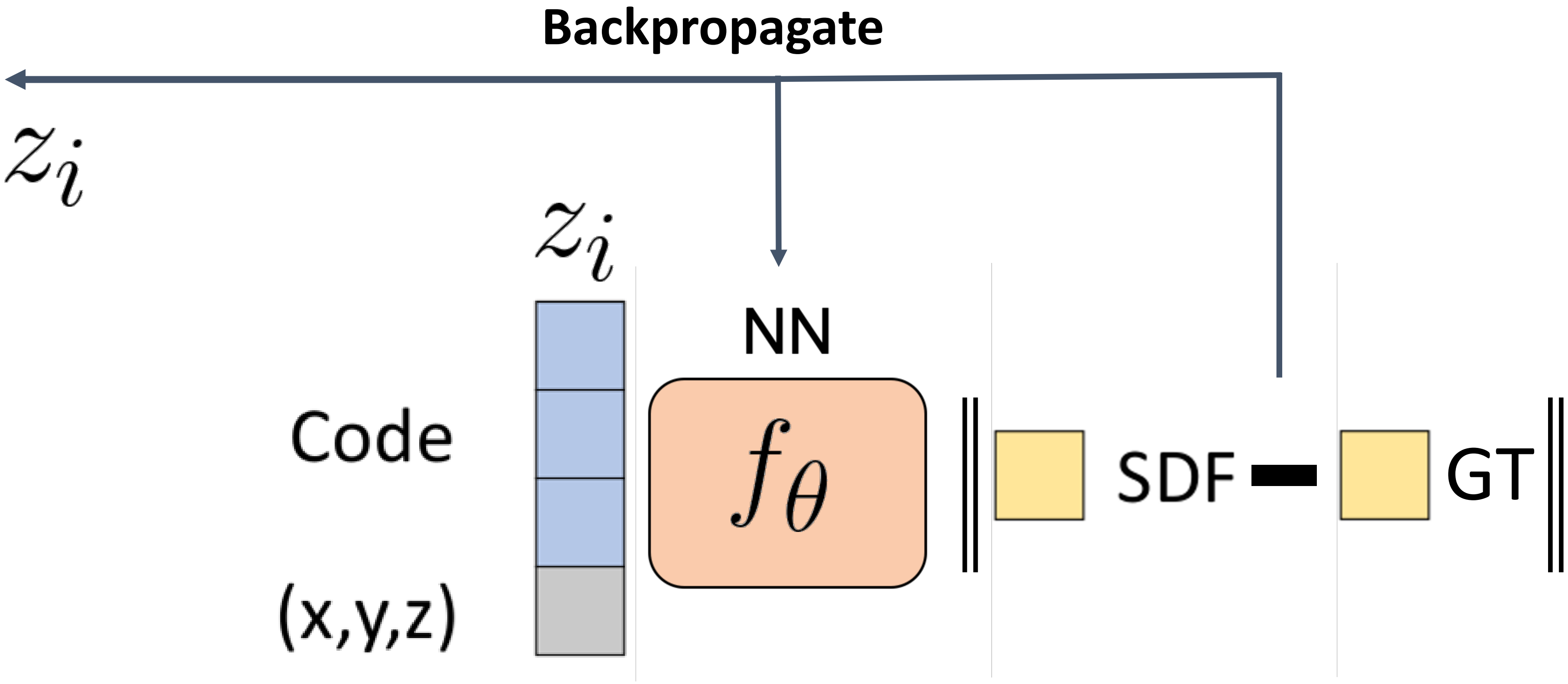
DeepSDF Training



$$\hat{\boldsymbol{z}} = \arg\min_{\boldsymbol{z}} \sum_{(\boldsymbol{x}_j, \boldsymbol{s}_j) \in X} \mathcal{L}(f_\theta(\boldsymbol{z}, \boldsymbol{x}_j), s_j)$$

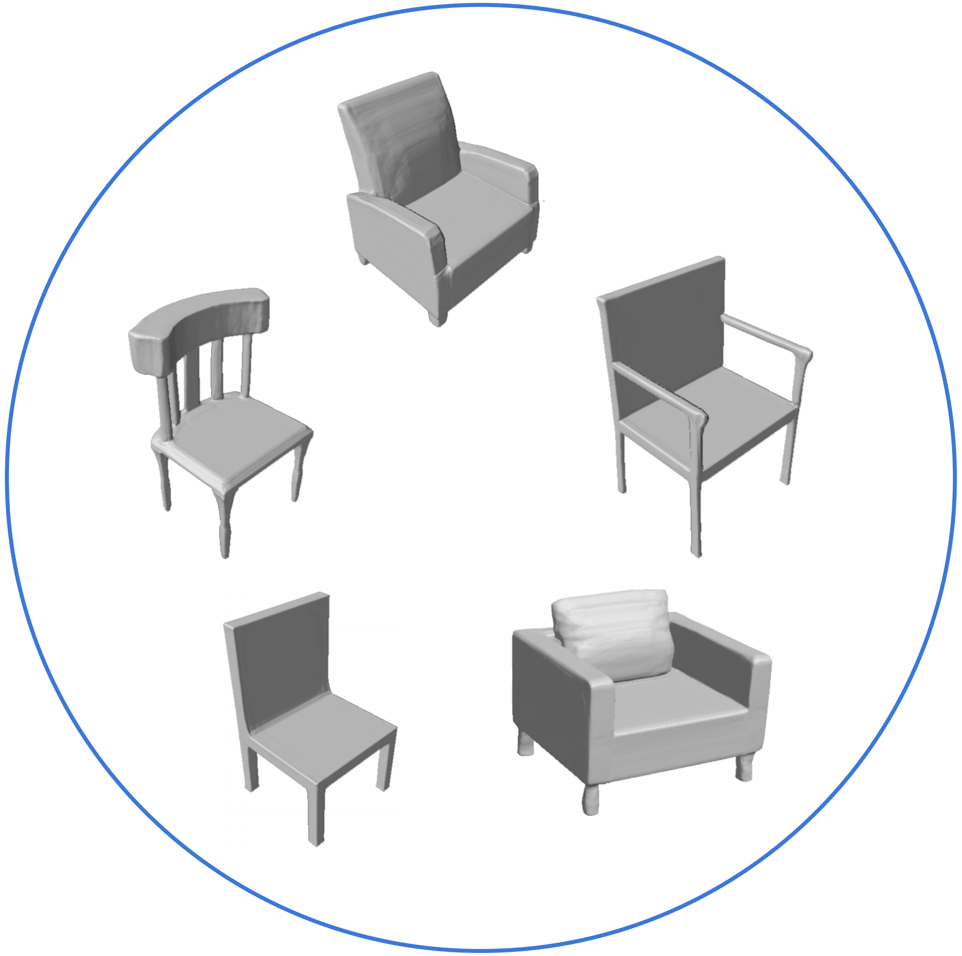
DeepSDF Training



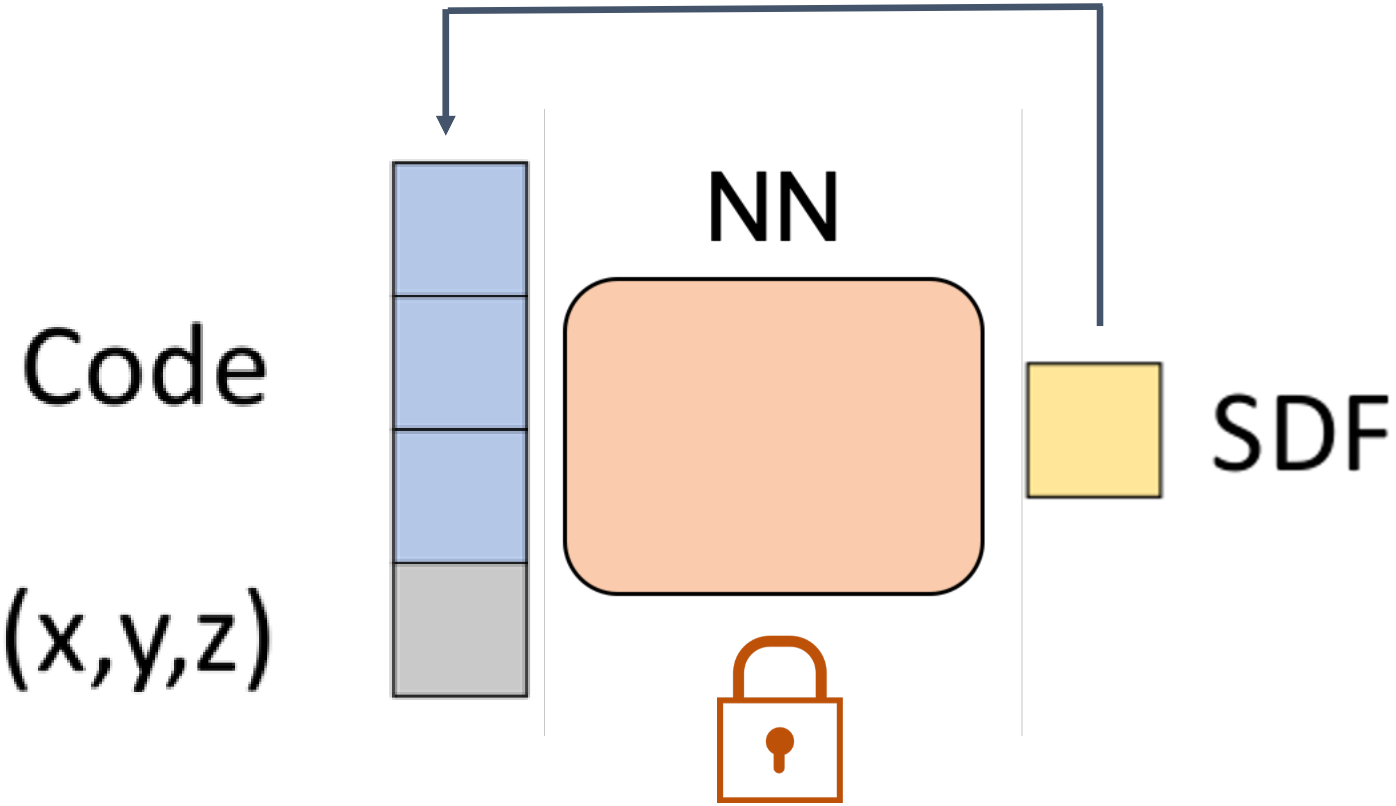


$$\hat{\boldsymbol{z}} = \arg\min_{\boldsymbol{z}} \sum_{(\boldsymbol{x}_j, \boldsymbol{s}_j) \in X} \mathcal{L}(f_\theta(\boldsymbol{z}, \boldsymbol{x}_j), s_j) + \frac{1}{\sigma^2} ||\boldsymbol{z}||_2^2$$

Latent Space of Shapes

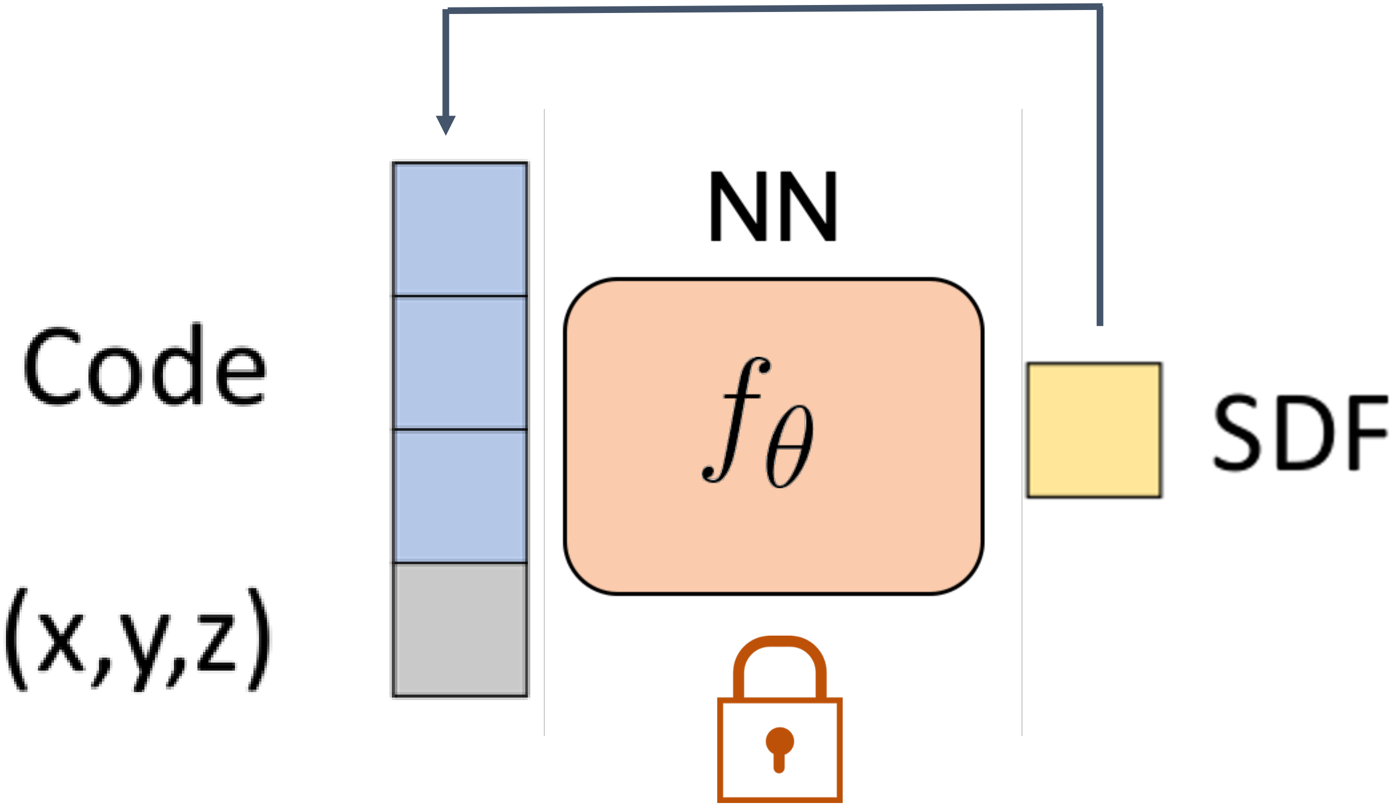


Auto-Decoder - Inference



Test Shape

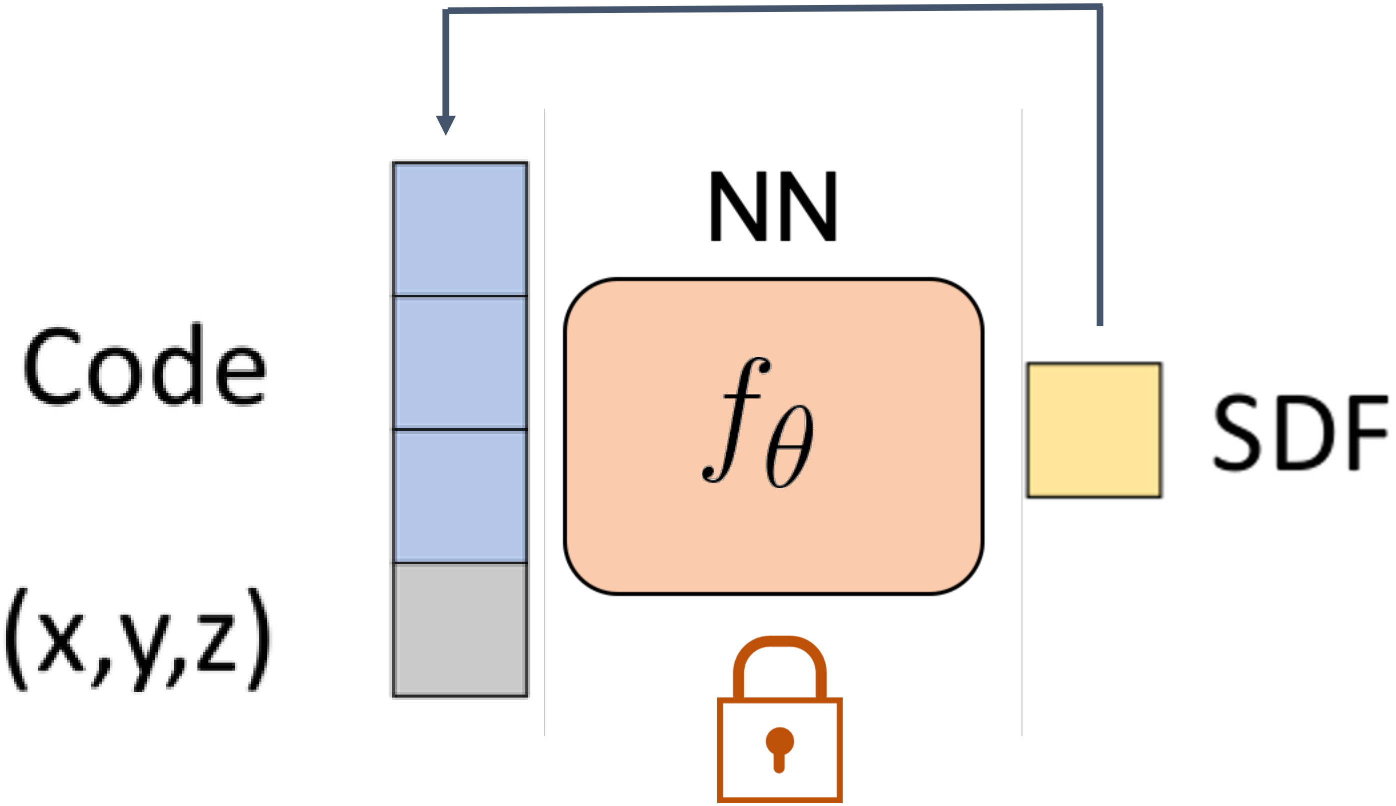
Auto-Decoder - Inference



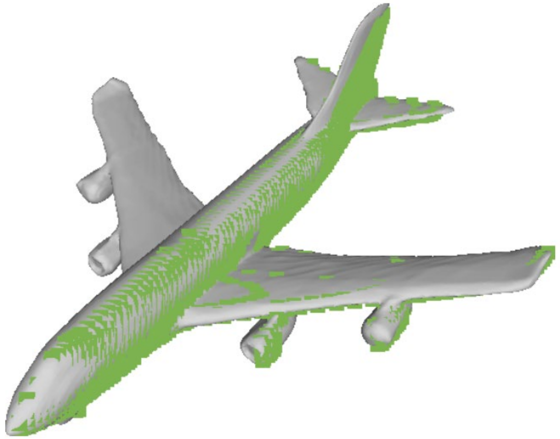
Test Shape

Reconstruction

Auto-Decoder - Inference

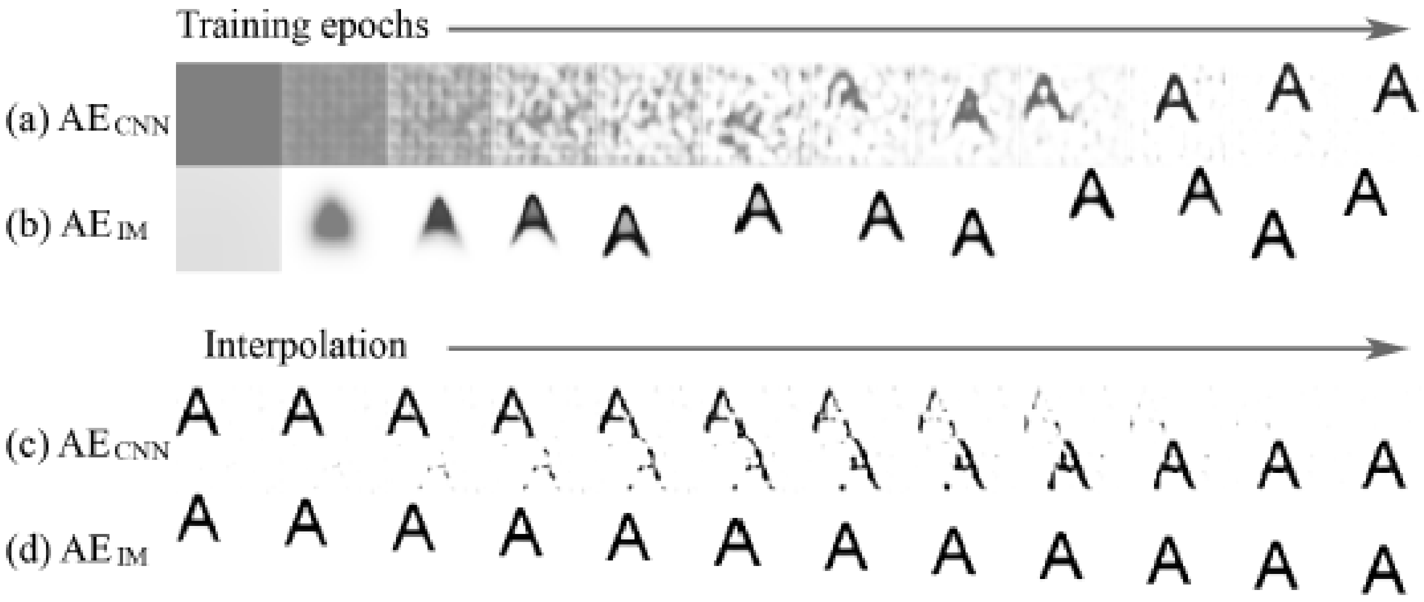


Input



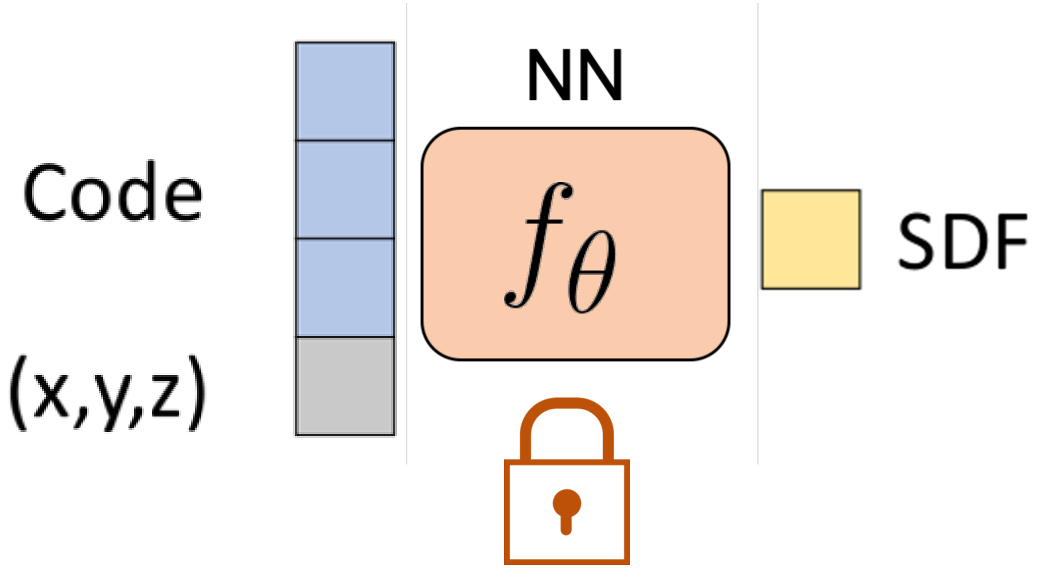
Reconstruction

IM-GAN



Inference Time

$$\hat{\boldsymbol{z}} = \arg\min_{\boldsymbol{z}} \sum_{(\boldsymbol{x}_j, \boldsymbol{s}_j) \in X} \mathcal{L}(f_\theta(\boldsymbol{z}, \boldsymbol{x}_j), s_j)$$



Adding Priors to Inference

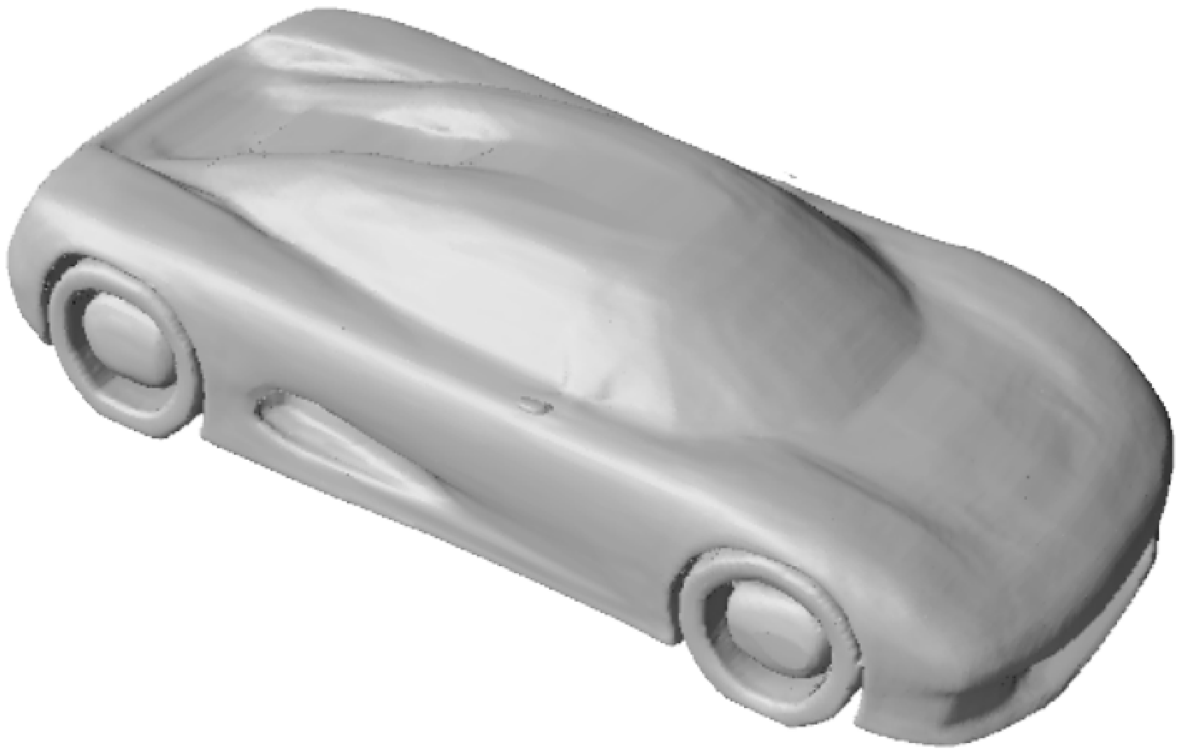
$$\hat{\boldsymbol{z}} = \arg\min_{\boldsymbol{z}} \sum_{(\boldsymbol{x}_j, \boldsymbol{s}_j) \in X} \mathcal{L}(f_\theta(\boldsymbol{z}, \boldsymbol{x}_j), s_j)$$

Distribution Prior: $\frac{1}{\sigma^2} ||\boldsymbol{z}||_2^2$

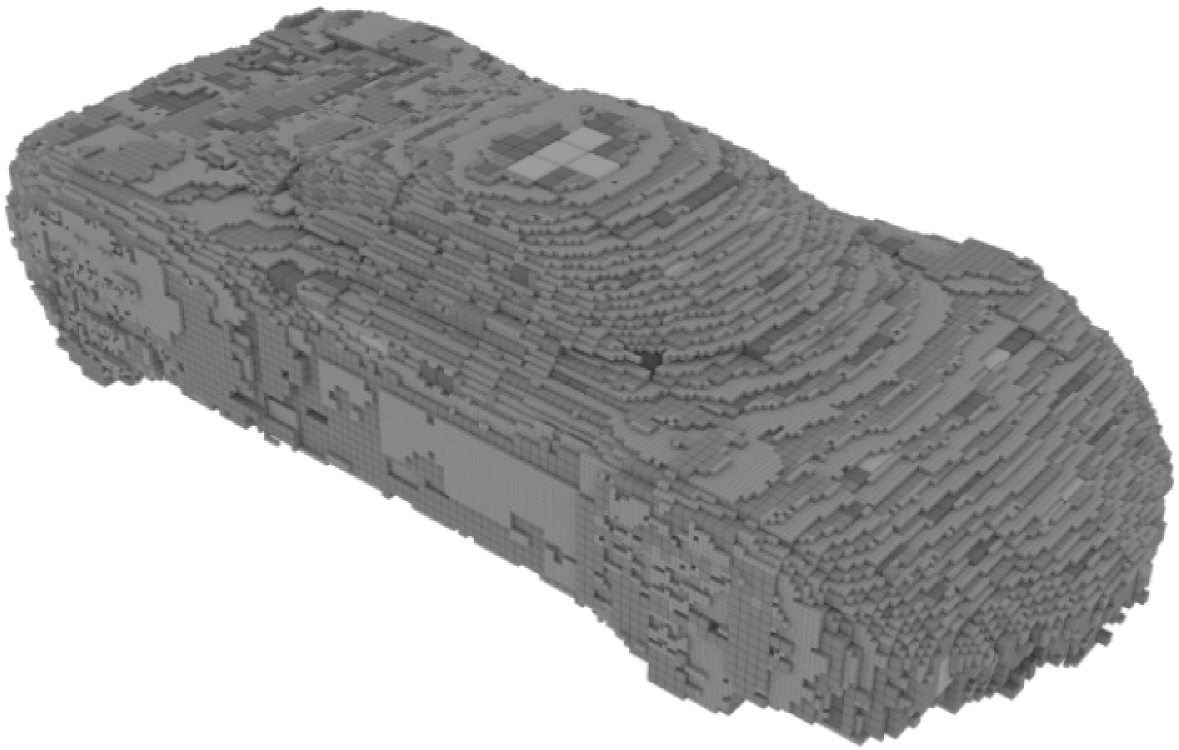
SDF Regularization: $\left(\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x};\theta)\| - 1 \right)^2$ (Matan et al. 2020)

Normal Regularization: $\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_i;\theta) - \boldsymbol{n}_i\|$

Results: Comparison with Octree-Based



Our
Reconstruction



Octree Based

Results: Auto-Encoding Unknown Shapes

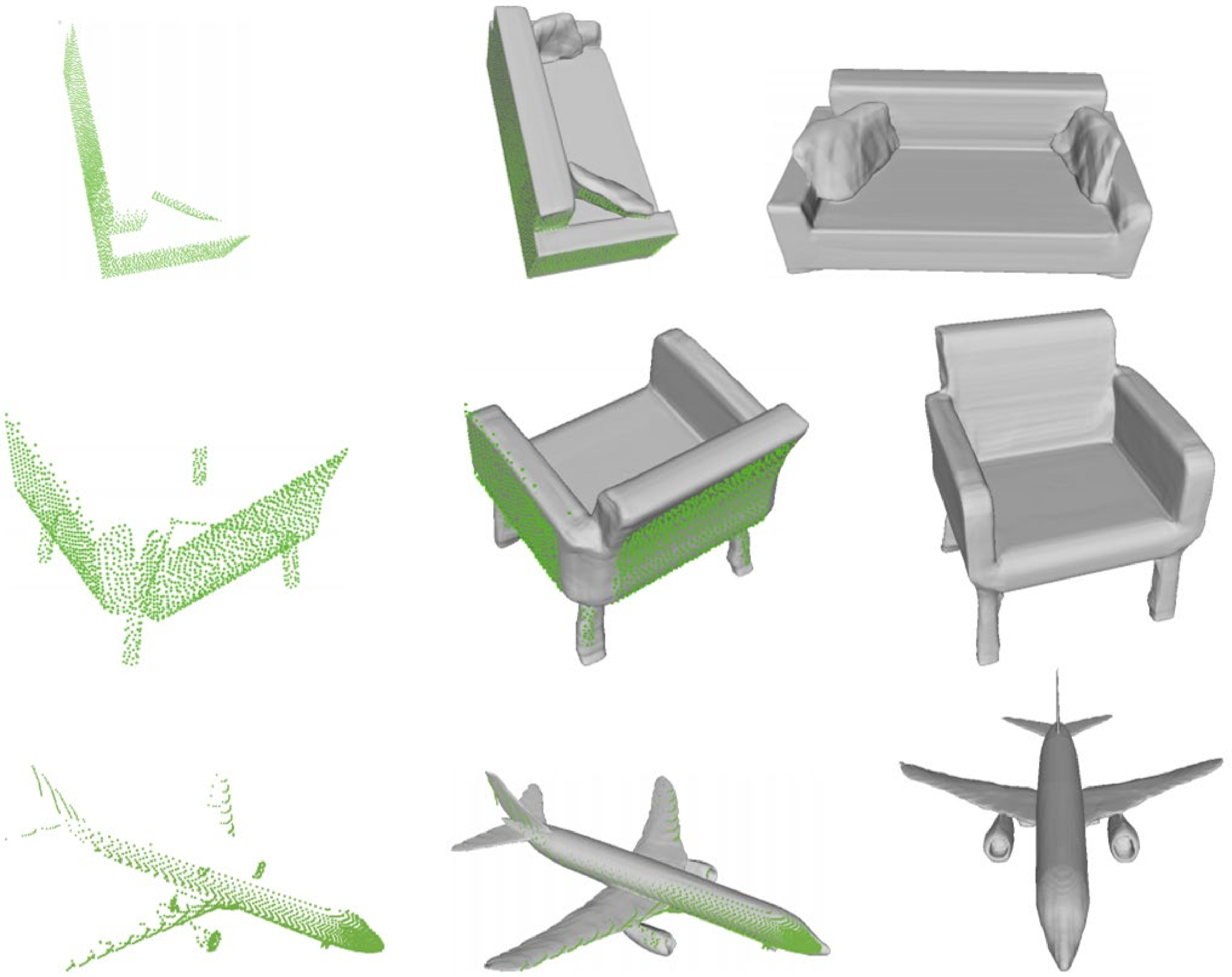

Ground Truth


Our Reconstruction

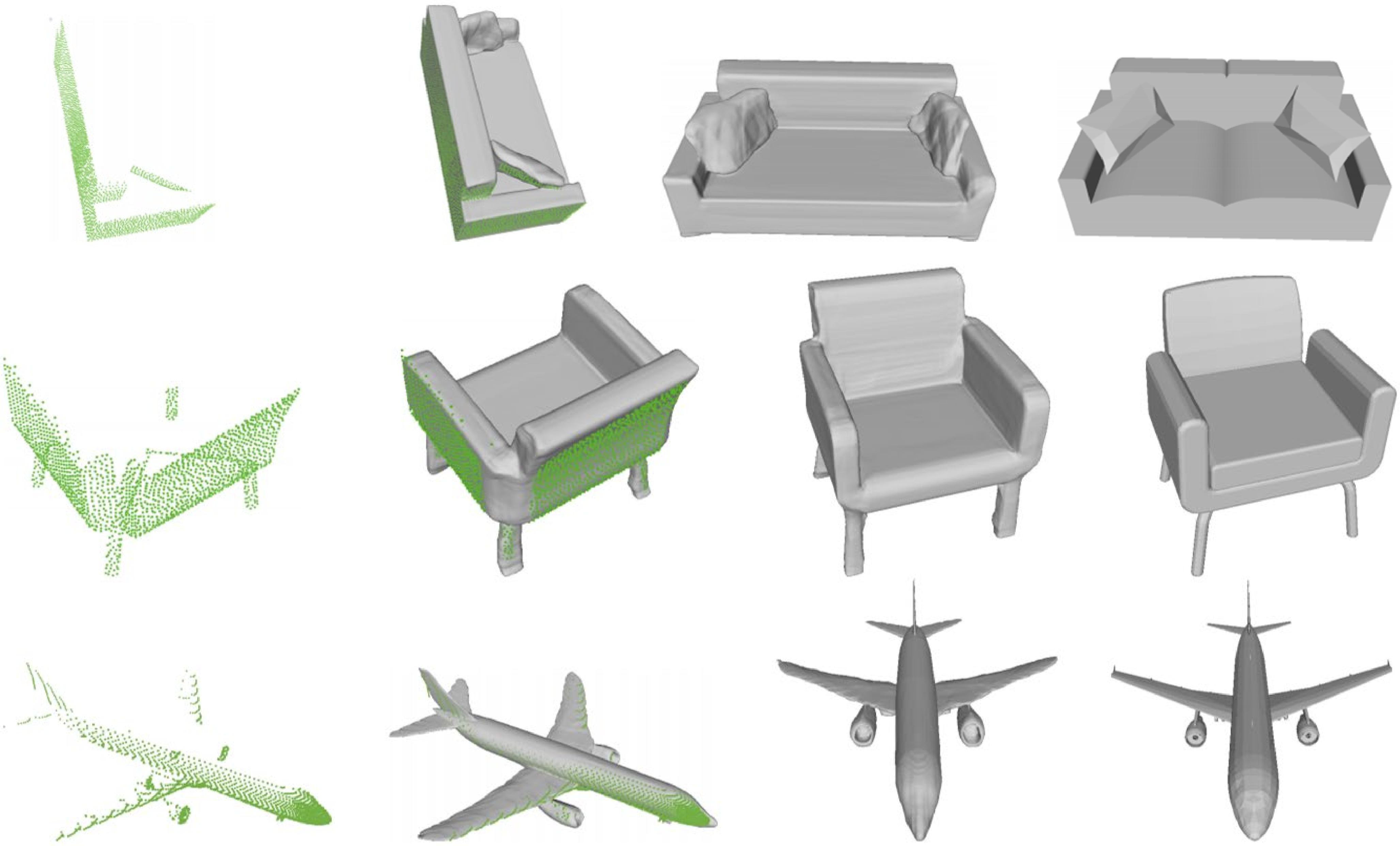

Atlasnet (25 Patches)


Atlasnet (1 Patch)

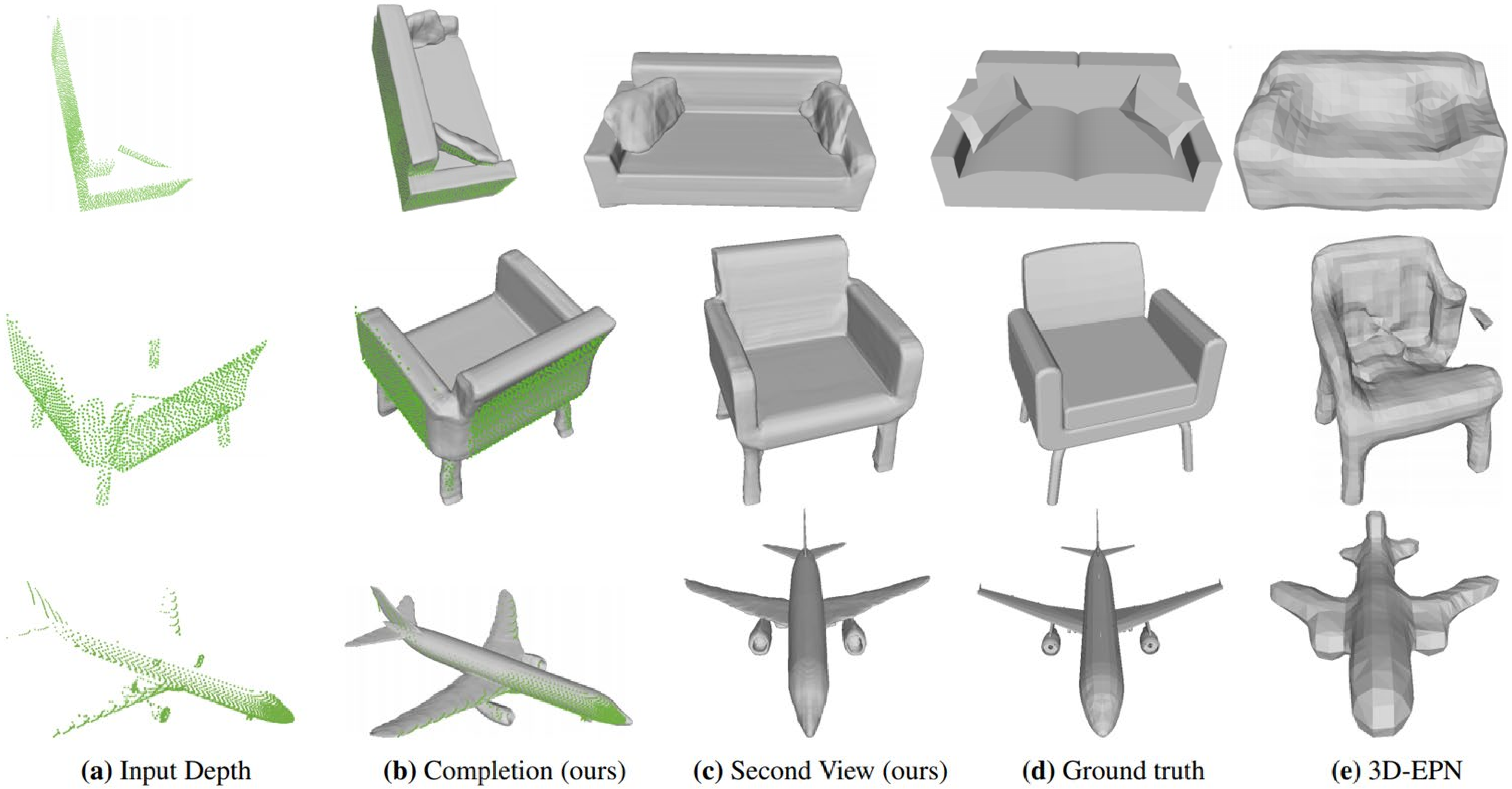
(a) Input Depth



(a) Input Depth **(b)** Completion (ours) **(c)** Second View (ours)

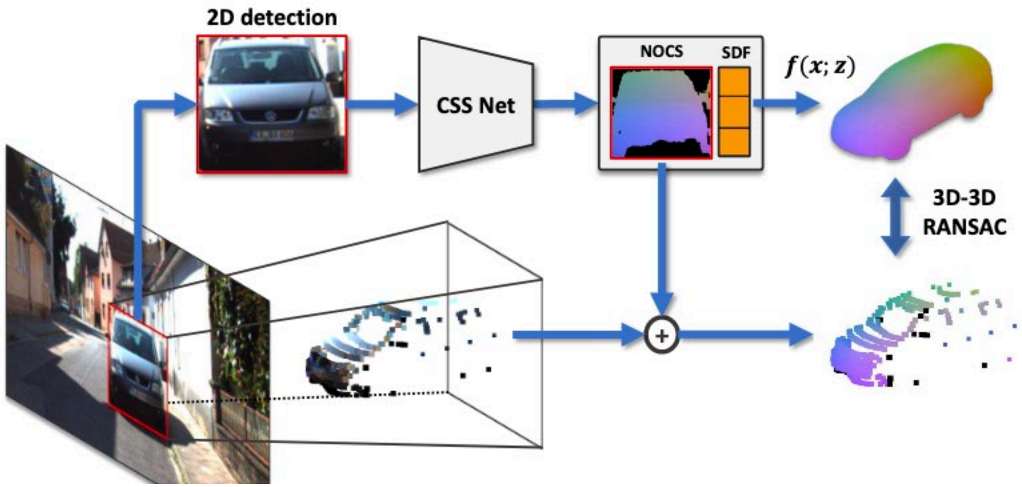


(a) Input Depth **(b)** Completion (ours) **(c)** Second View (ours) **(d)** Ground truth

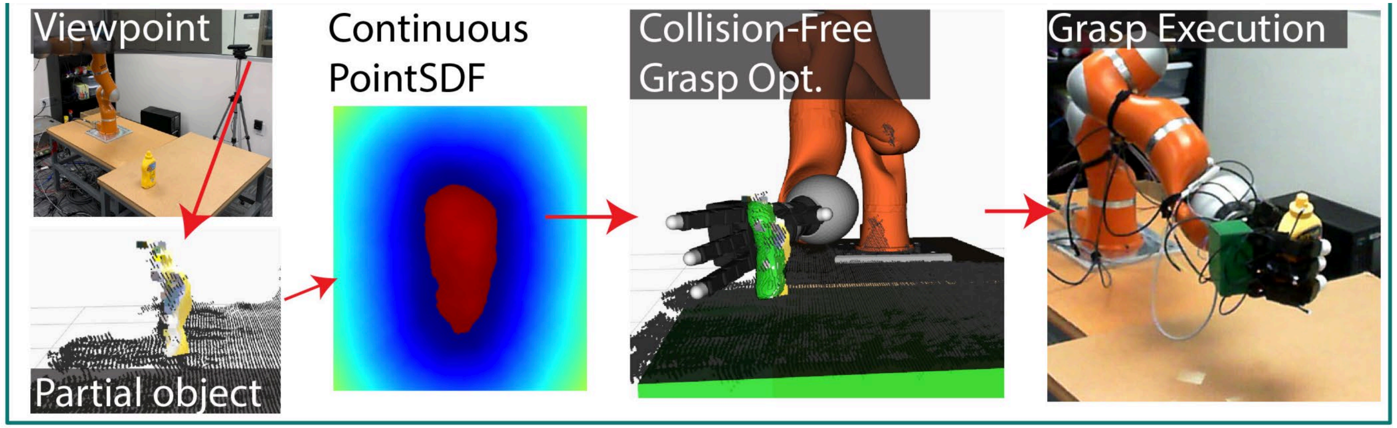


(a) Input Depth
(b) Completion (ours)
(c) Second View (ours)
(d) Ground truth
(e) 3D-EPN

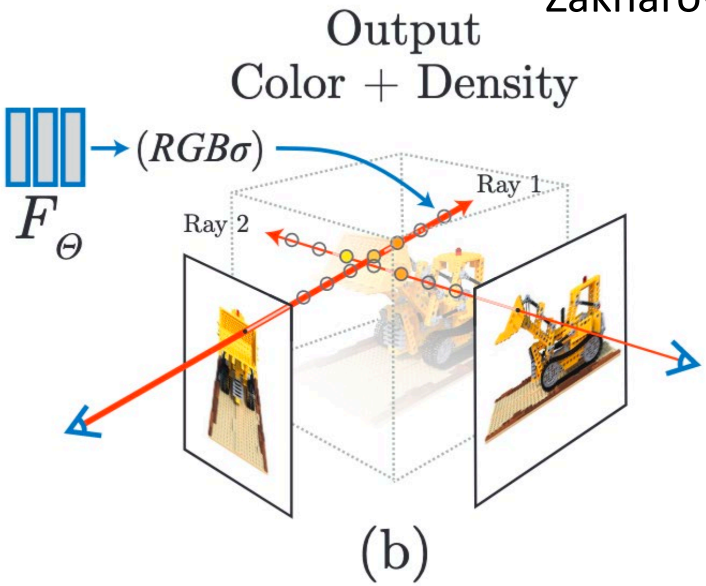
Many Follow On Works



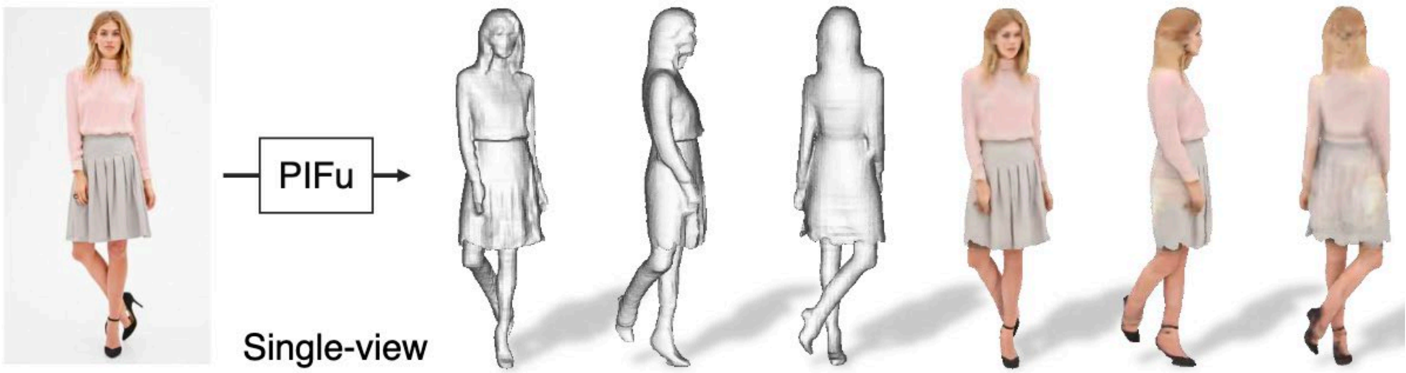
Zakharov et al. 2020



Merwe et al. 2020



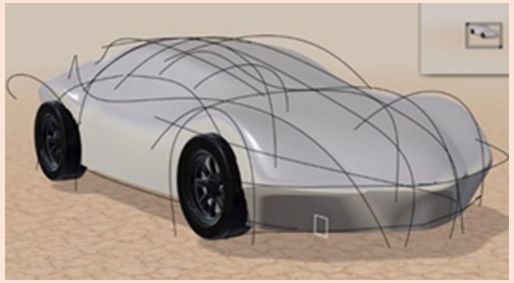
Mildenhall et al. 2020



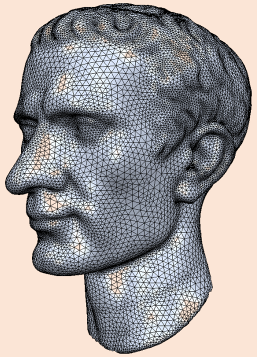
Saito et al. 2019

Shape Modeling

Designed Shapes



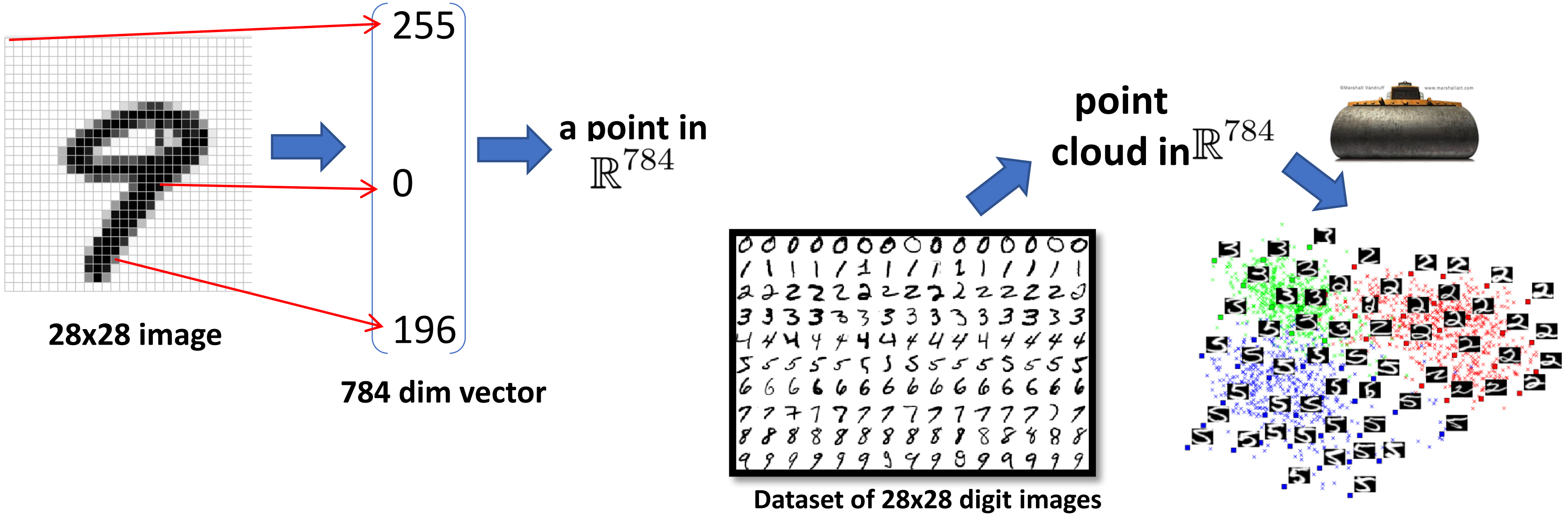
Acquired Shapes



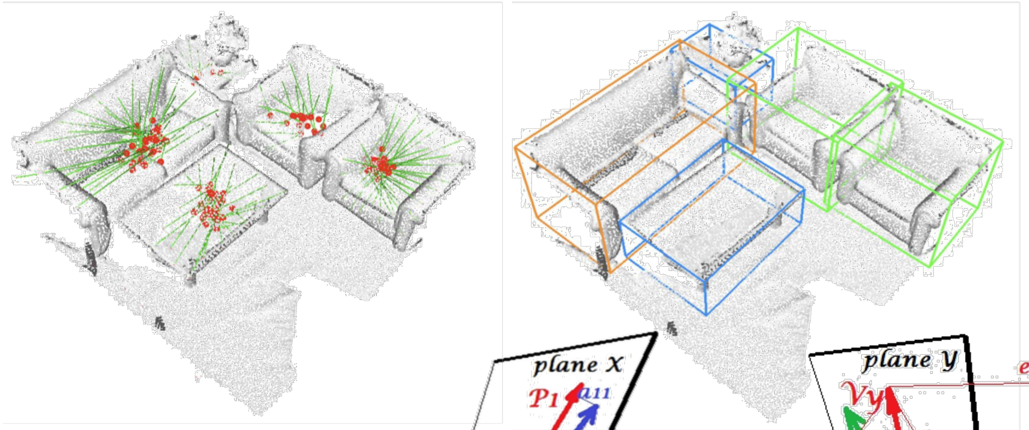
Learned Shapes

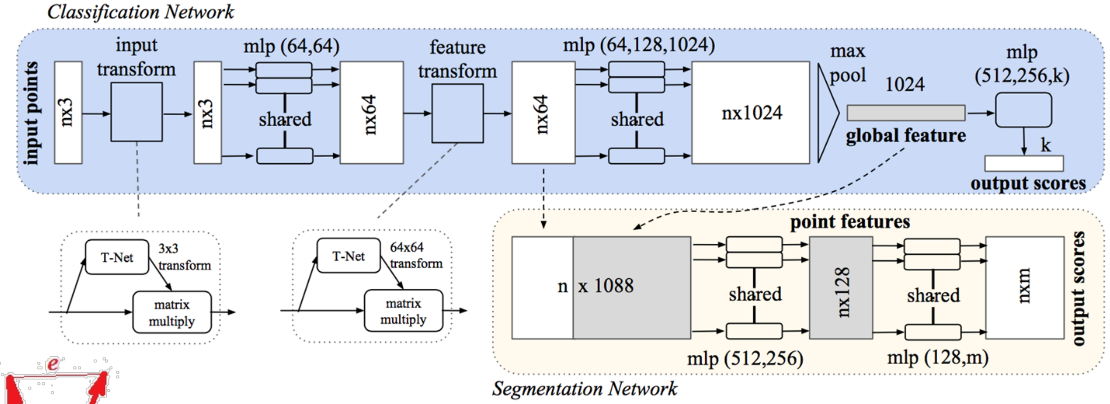
Connections to Other Areas – CS233

- Here: 3D reduced to 2D – “dimensionality reduction”
- Look up: “non-linear dimensionality reduction”
- Ways of organizing/visualizing high dim data



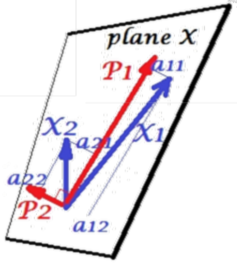
CS233: Geometric and Topological Data Analysis

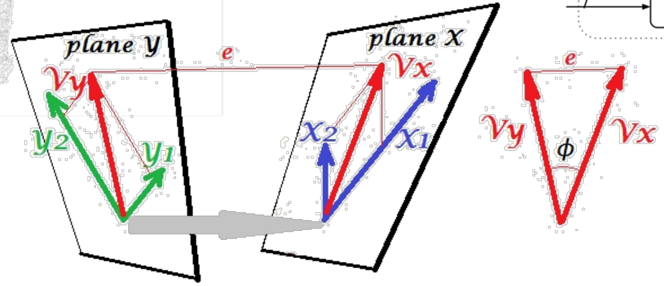




PointNet – Deep Networks for Point Cloud Data

Spring 2020-21

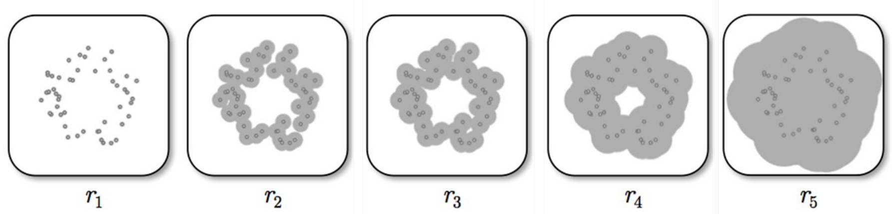




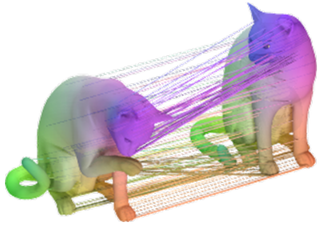
Principal Components Analysis

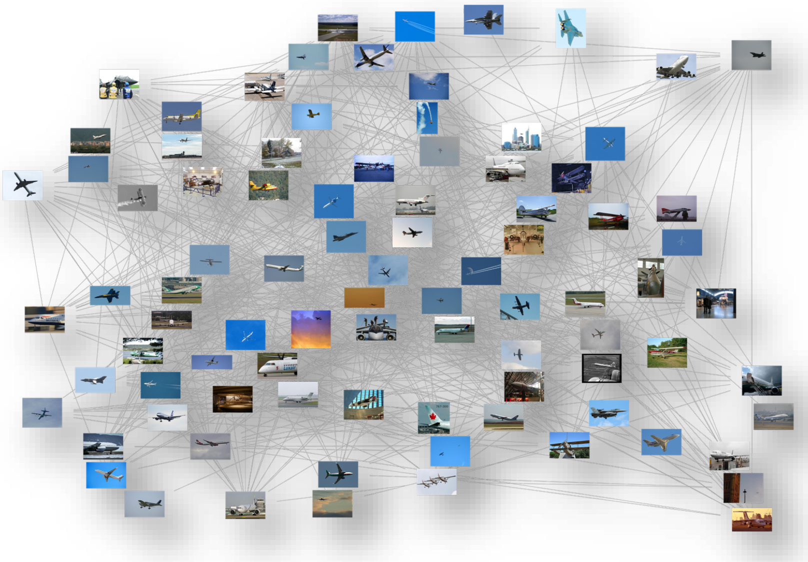
Canonical Correlation Analysis

The Shape of Data

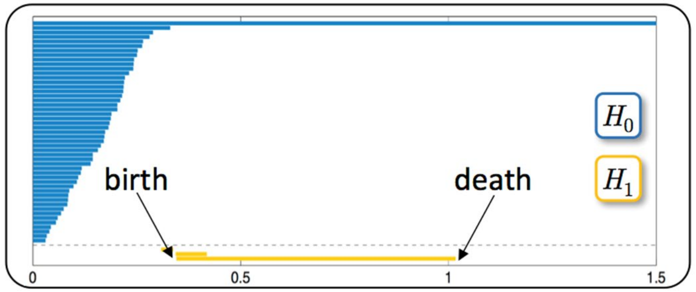


Functional Maps and
Functional Map Networks





TDA



Persistent Homology

That’s All