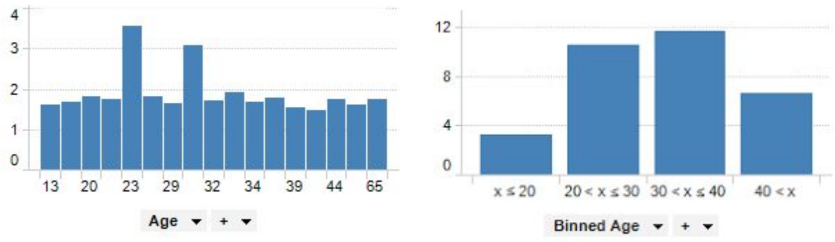
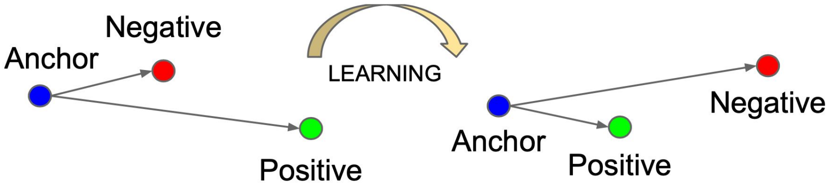
Deep Learning

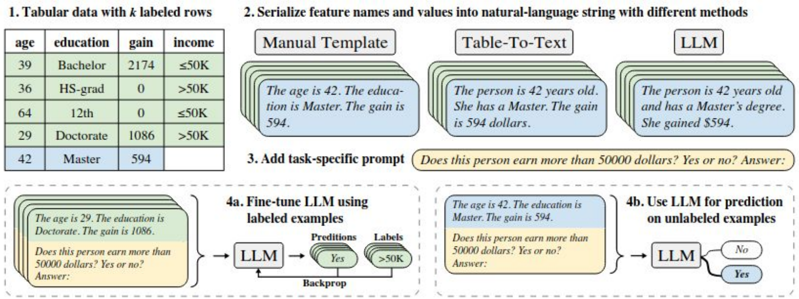
Lecture 14

Recap

- Encoding
- Pretraining
- Tabular DL
- Tabular DL as text

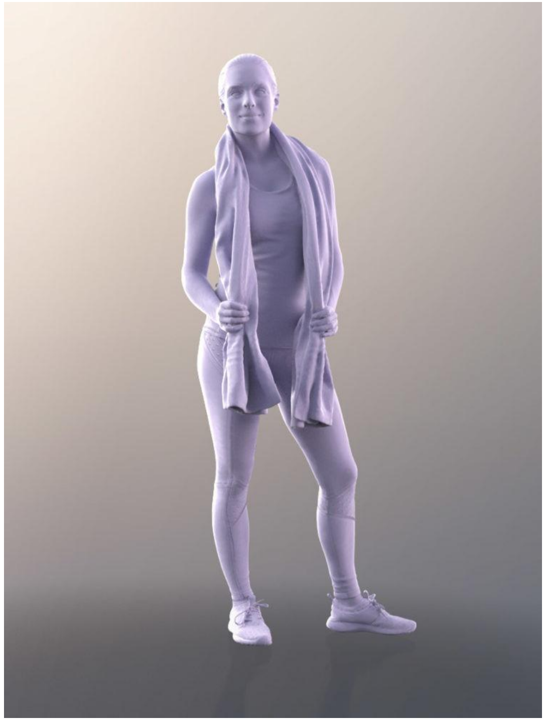


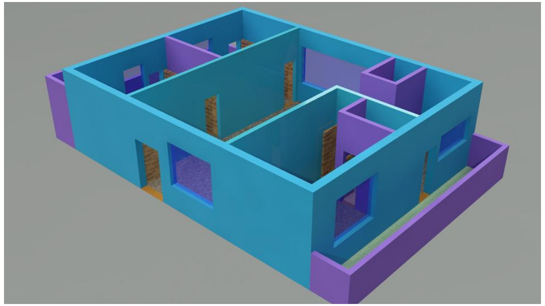




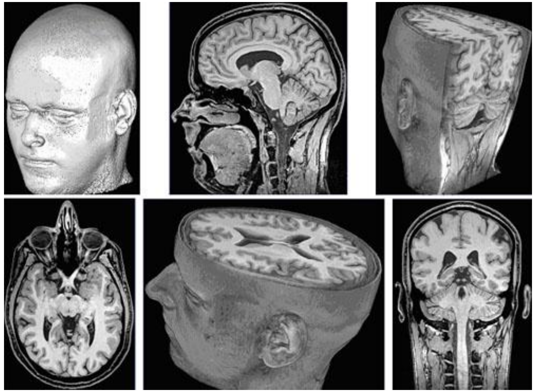
age	education	gain	income
39	Bachelor	2174	≤50K
36	HS-grad	0	>50K
64	12th	0	≤50K
29	Doctorate	1086	>50K
42	Master	594	

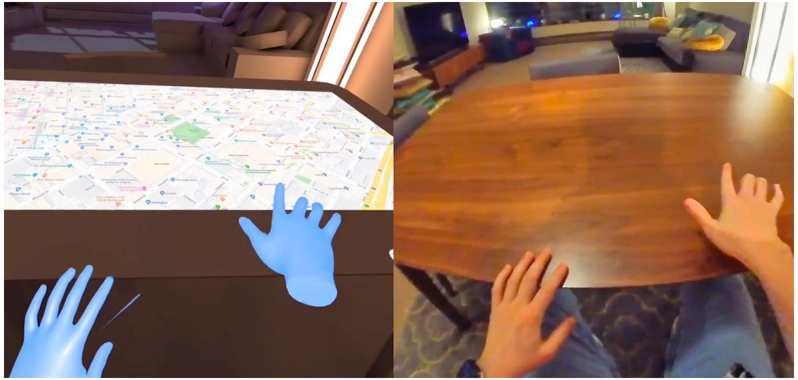
Why we need 3D?





3D applications



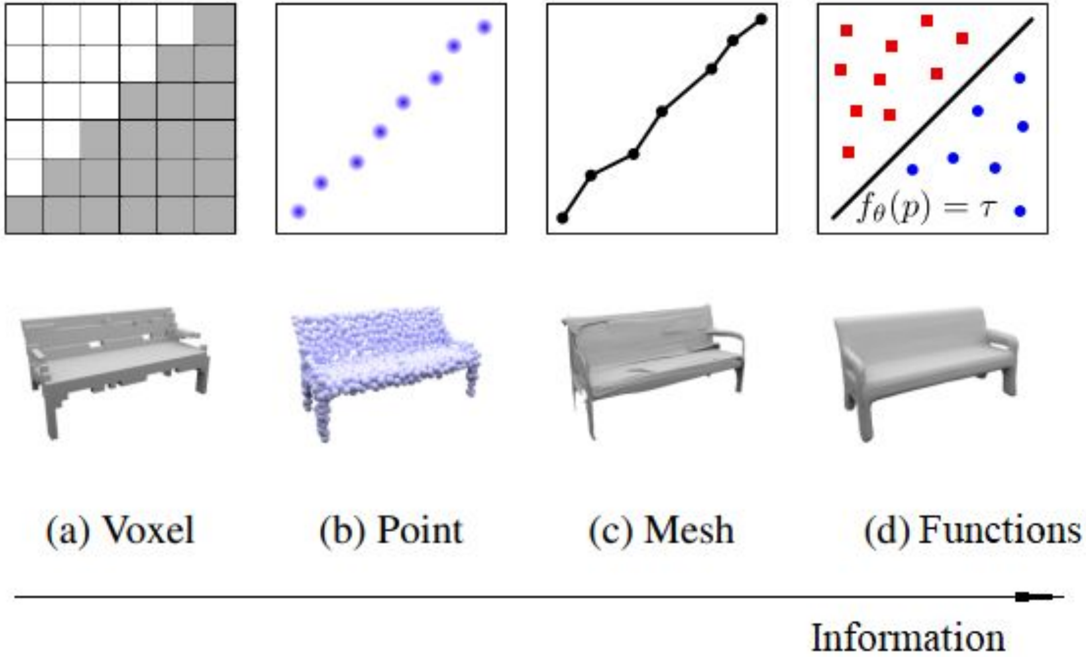


Possible 3D tasks

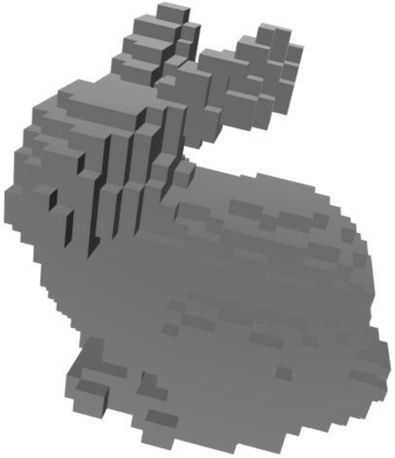
- classification, clusterization
- generation of 3D data
- 3D reconstruction of object from one or a few views
- animation of static meshes

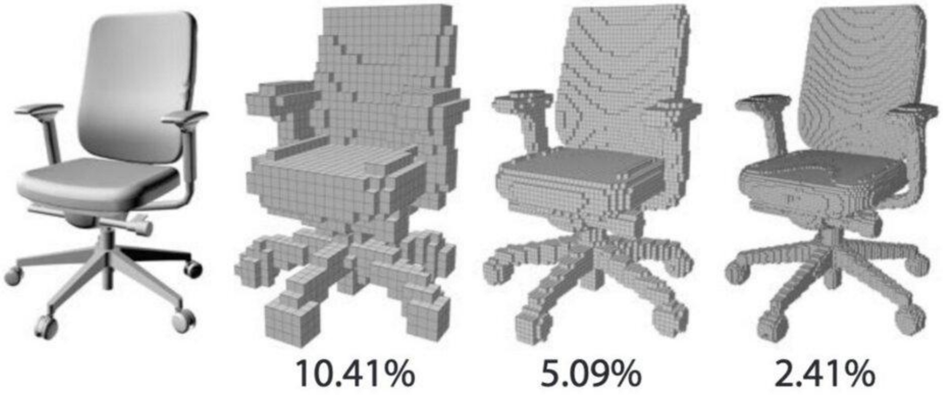
3D Reconstruction

3D data representations



Voxels (eng. volumetric + pixel)

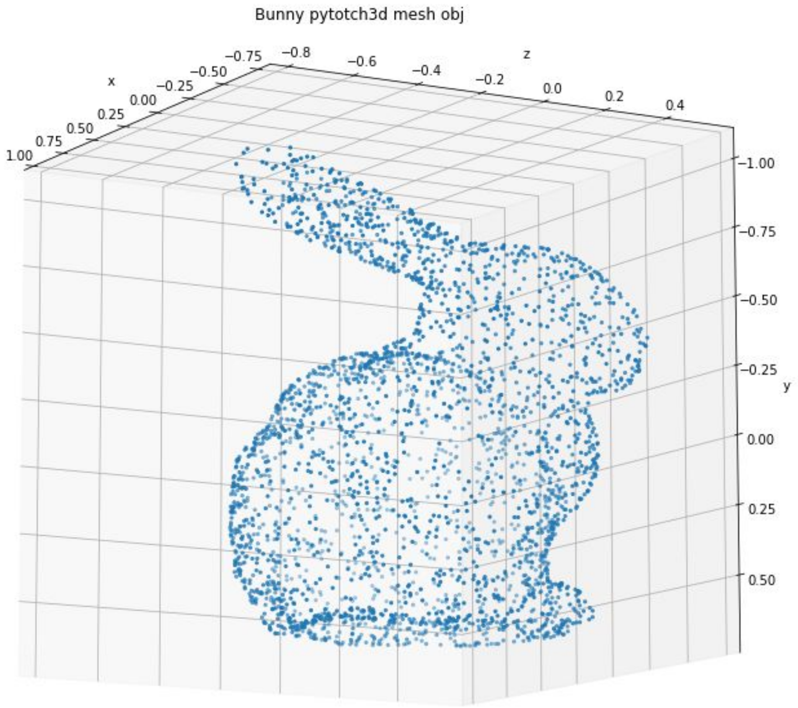




- Natural generalization of approaches that have been used for image processing.
- Can obtain from any other type of representation.
- Connection with the physical properties of objects

The higher the resolution, the better the approximation of the original shape but the lower the fraction of occupied voxels.

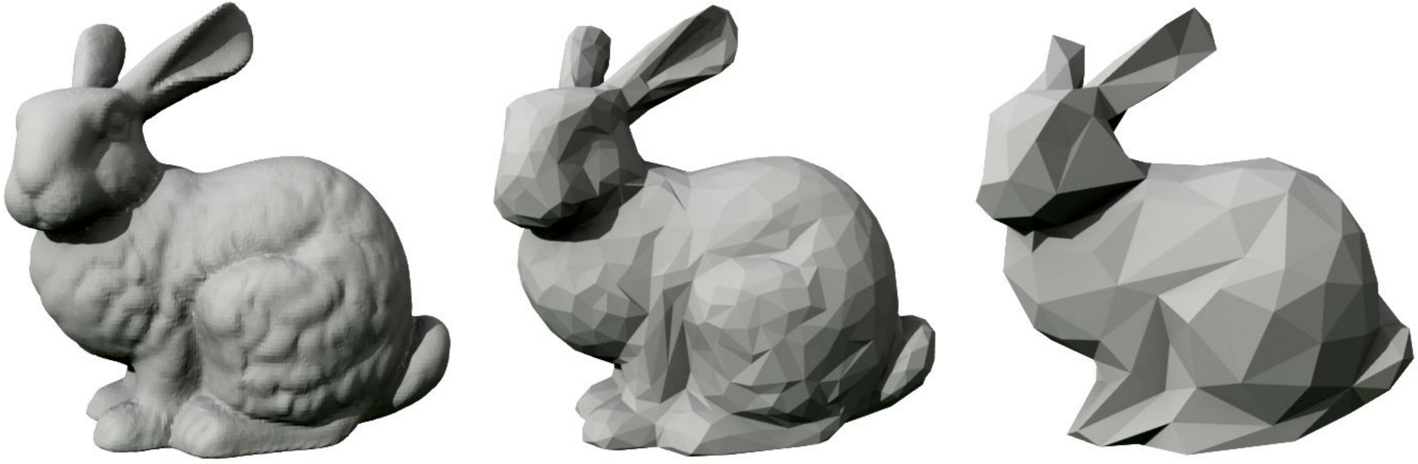
Point Clouds



- Natural data format in spatial scanning problems.
- You can multiply the points in the point list with linear transformation matrices.
- Can be easily obtained from polygonal and functional models.

❏ Data disorder.
❏ No information about the connections between points -> no topology.
❏ Hard postprocessing.

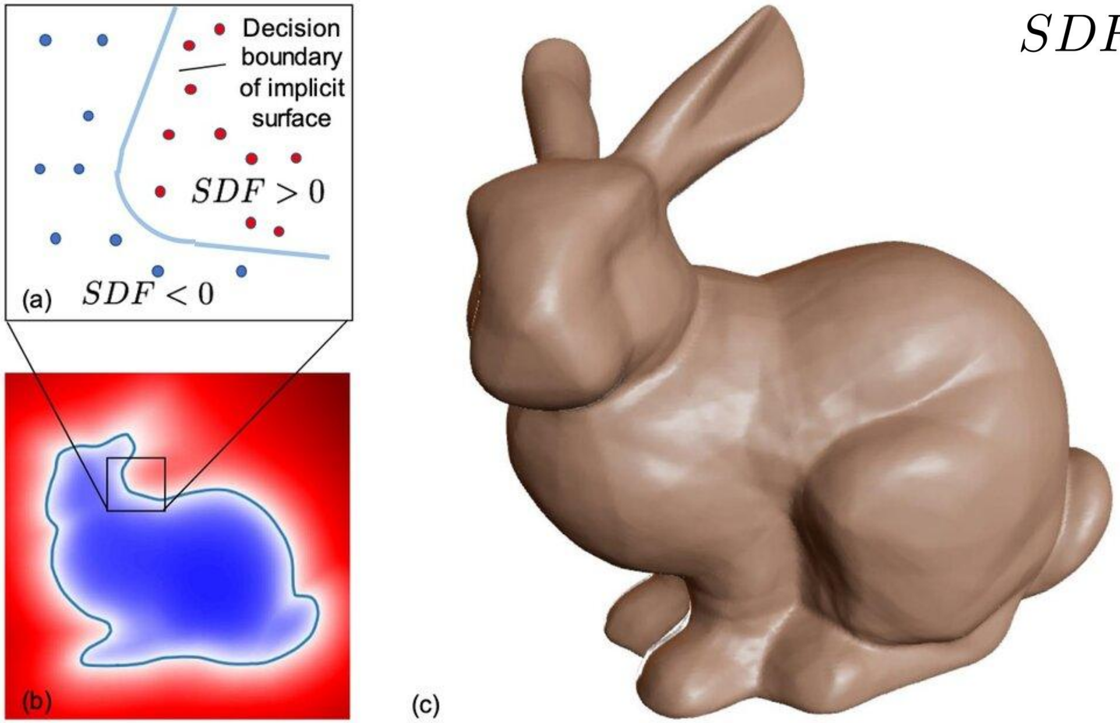
Meshes



- Natural format for use in computer graphics and games.
- Can better describe the spatial features of objects (topology, surface shape)

- ❏ Need a special mathematical apparatus for extracting features from polygonal models(convolution on graphs).
- ❏ Format is sensitive to data outliers.

Implicit 3D representations (Signed distance function)



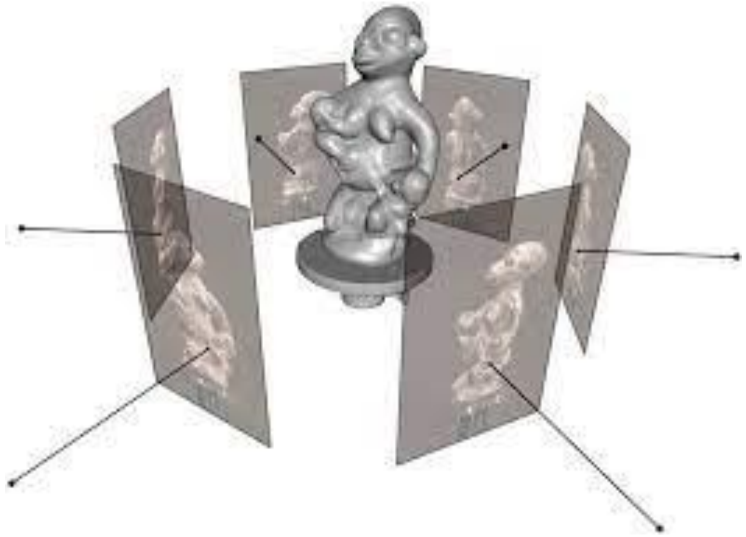
$$SDF(x) = \begin{cases} \rho(x, \partial\Omega), & x \notin \Omega \\ -\rho(x, \partial\Omega), & x \in \Omega \end{cases}$$

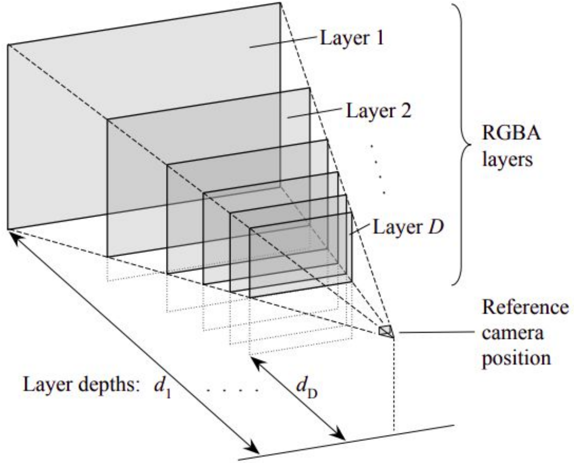
$$\rho(x, \partial\Omega) = \inf_{y \in \partial\Omega} \rho(x, y)$$

- Compact description
- Physically correct model.
- Model scalability.
- Can obtain all other formats.

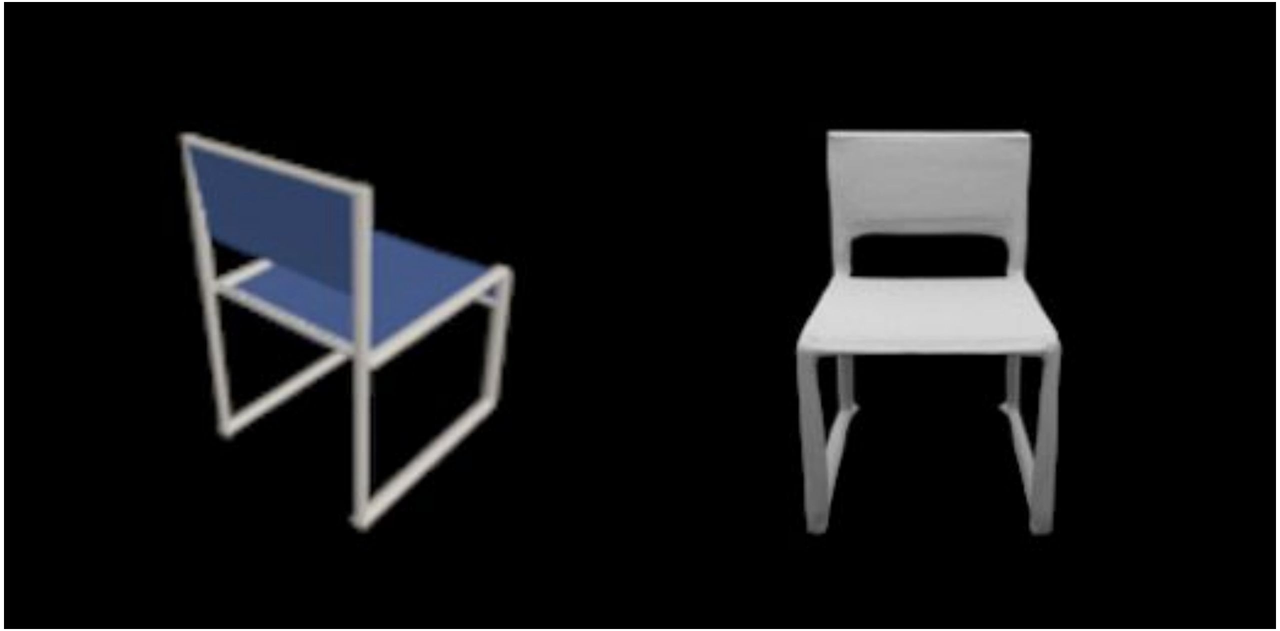
❏ Small number of datasets.
❏ Can't be obtained from other formats.
❏ difficult to work with textures.

Multi-view and multi-plane representation





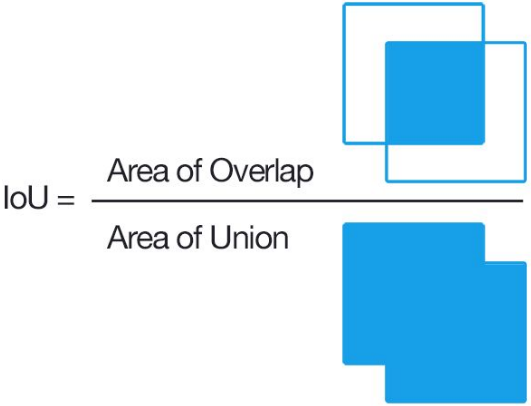
3D reconstruction from single RGB image

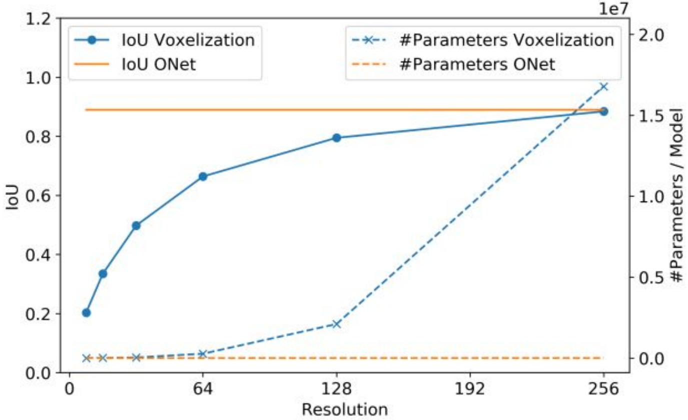


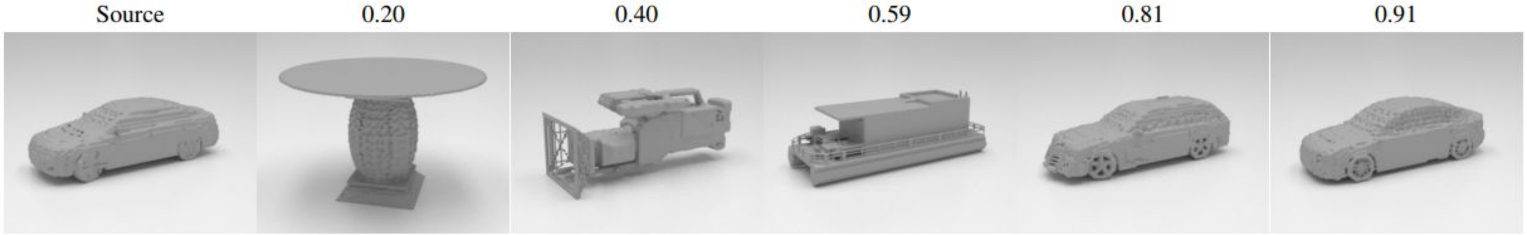
IoU metric

$$IoU(A,B) = \frac{|A \cap B|}{|A \cup B|}.$$

- Can calculate volume of the objects.
- Volume VS surface.
- Can have different insides.
- Not so good for point clouds and voxels.







Chamfer and normal loss/distance

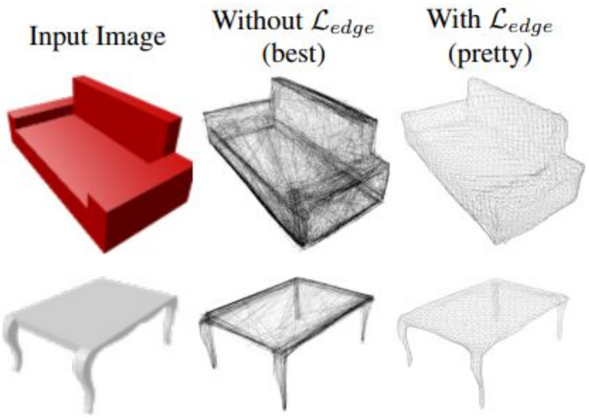
$$\Lambda_{P,Q} = \{(p, \arg\min_{q \in Q} \|p - q\|) : p \in P\},$$

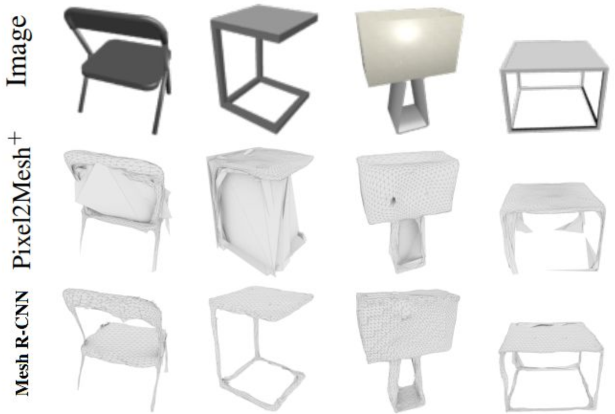
$$\mathcal{L}_{cham}(P,Q) = |P|^{-1} \sum_{(p,q)\in\Lambda_{P,Q}} \|p-q\|^2 \; + \sum_{(q,p)\in\Lambda_{Q,P}} \|q-p\|^2$$

$$\mathcal{L}_{norm}(P,Q) = -|P|^{-1} \sum_{(p,q)\in\Lambda_{P,Q}} |u_q \cdot u_p| \; - \; |Q|^{-1} \sum_{(q,p)\in\Lambda_{Q,P}} |u_p \cdot u_q|$$

Edge loss / regularizer

$$\mathcal{L}_{edge}(V, E) = \frac{1}{|E|} \sum_{(v,v')\in E} \|v - v'\|^2, \ E \subseteq V \times V.$$



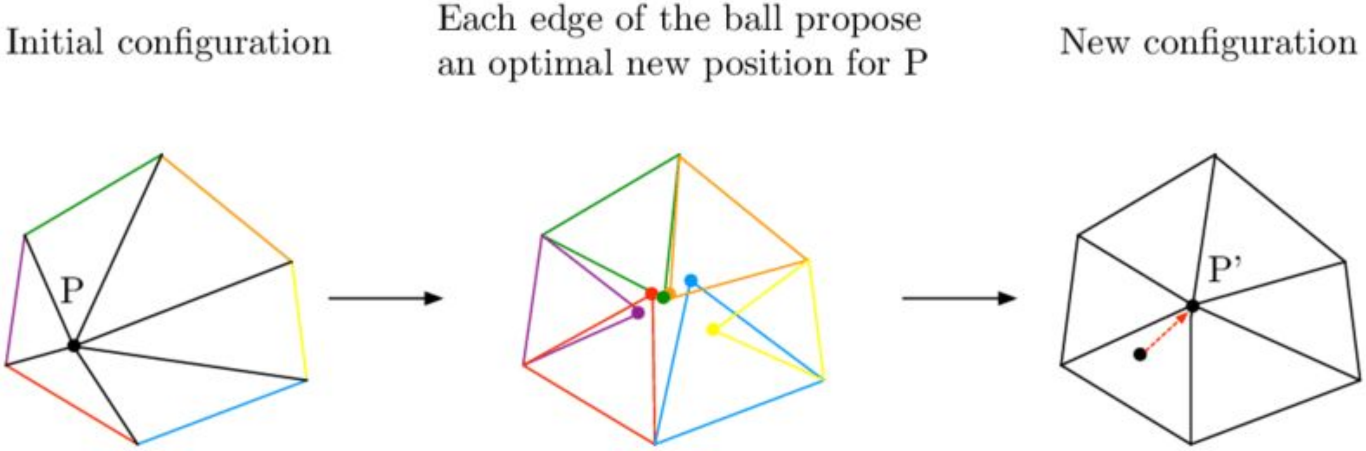


Smooth loss / regularizer

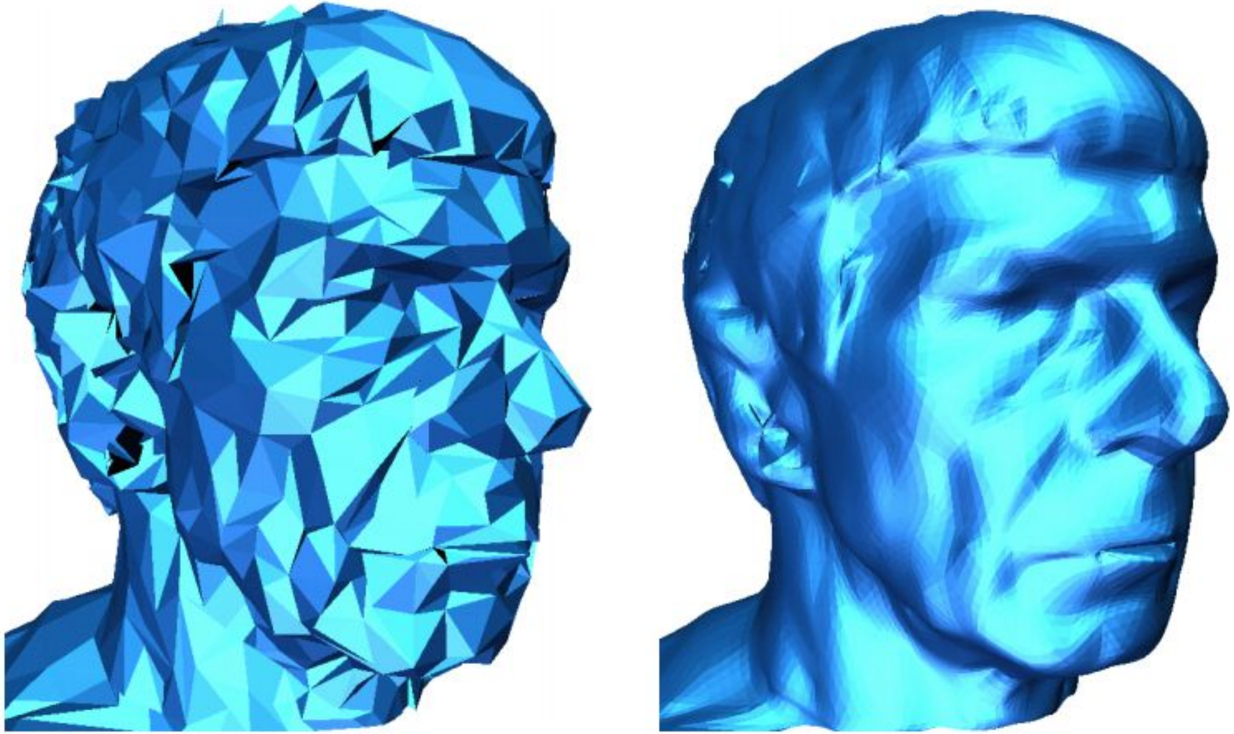
$$\mathcal{L}_{sm}(x) = \sum_{\theta_i \in \mathcal{E}} (\cos\theta_i + 1)^2.$$

Laplacian loss / regularizer

$$\mathcal{U}(\mathbf{p}) = \frac{1}{n}\sum_{i=0}^{n-1}\mathbf{p}_i - \mathbf{p} \qquad \delta_p = p - \sum_{k\in\mathcal{N}(p)} \frac{1}{\|\mathcal{N}(p)\|}k,$$

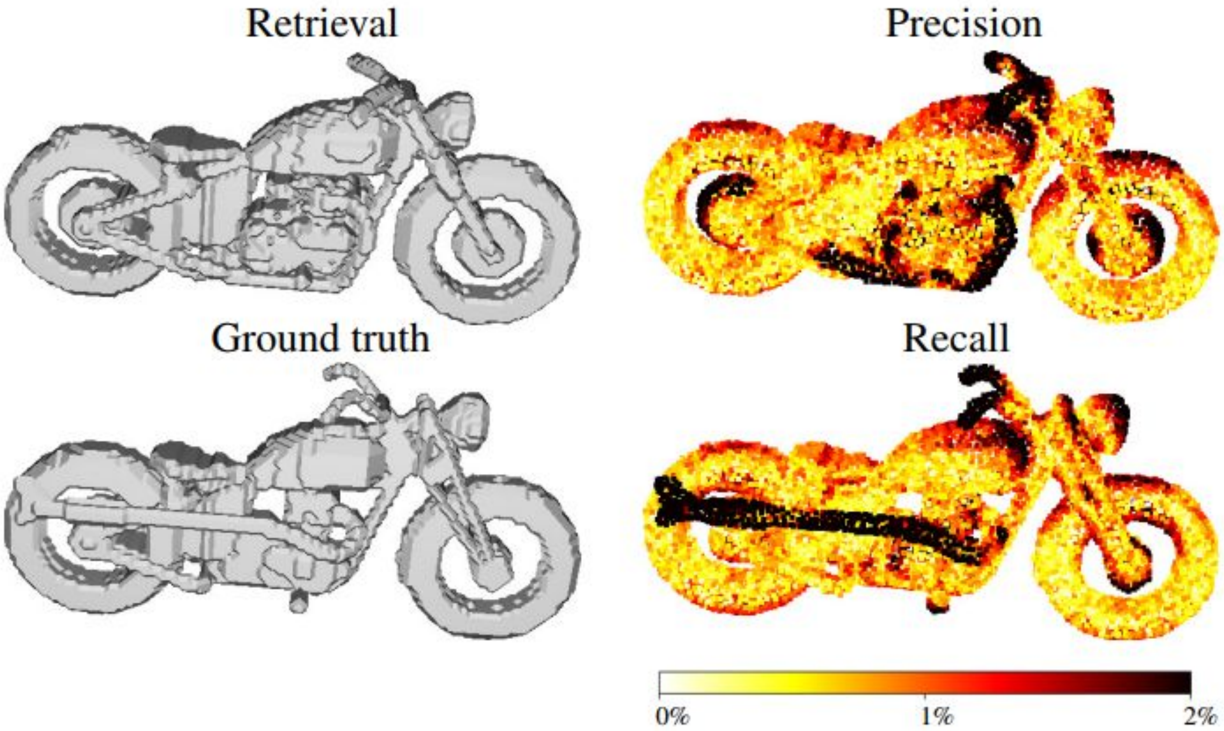


Laplacian loss / regularizer

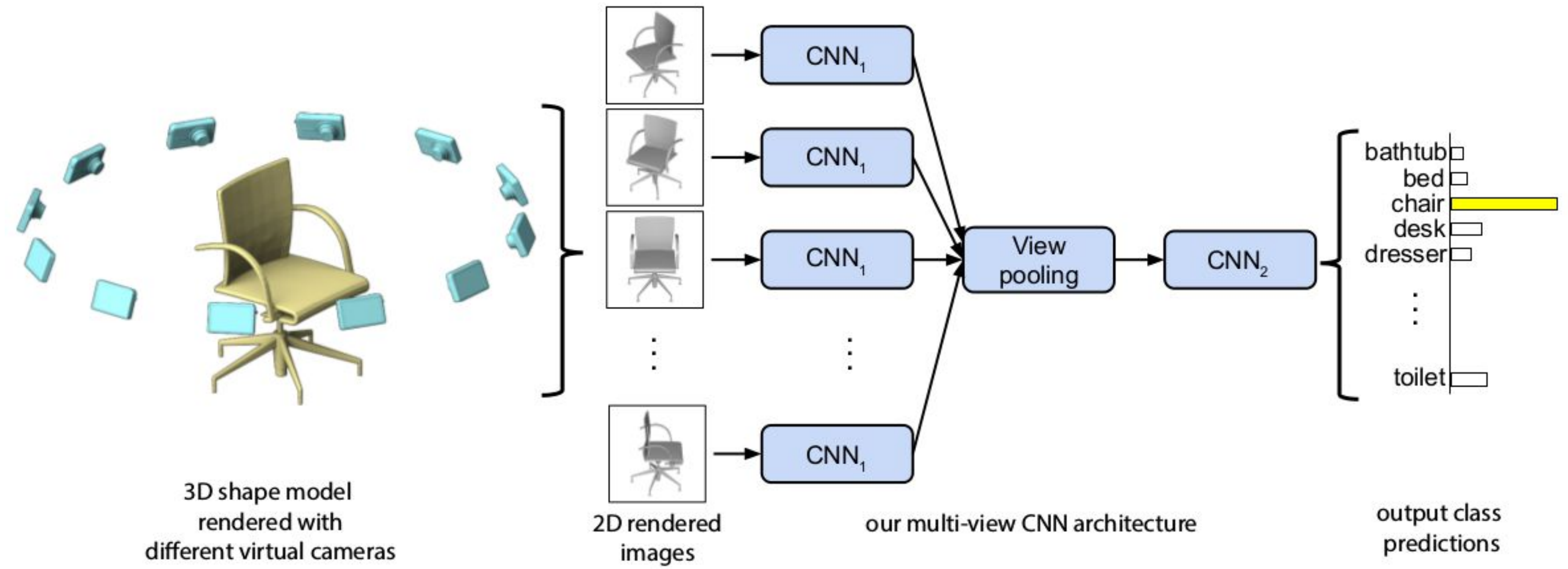


(a) (b)

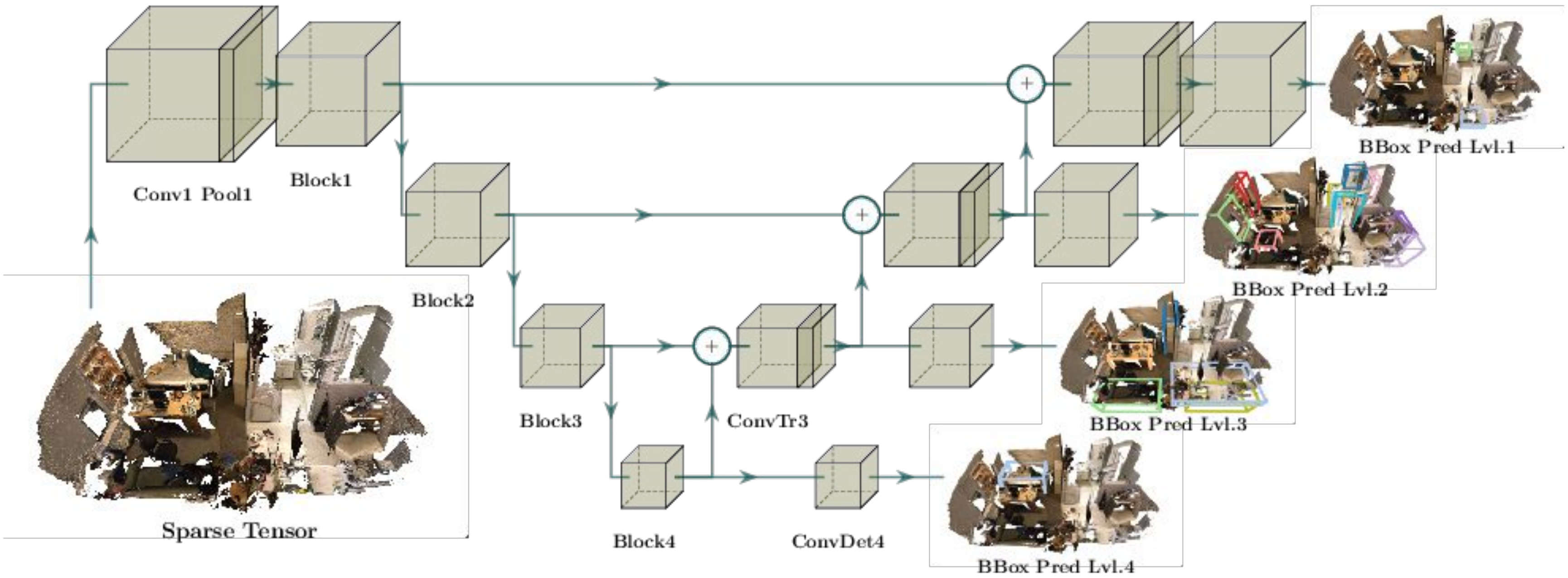
Other metrics (f1 score, Hausdorff losses, etc)



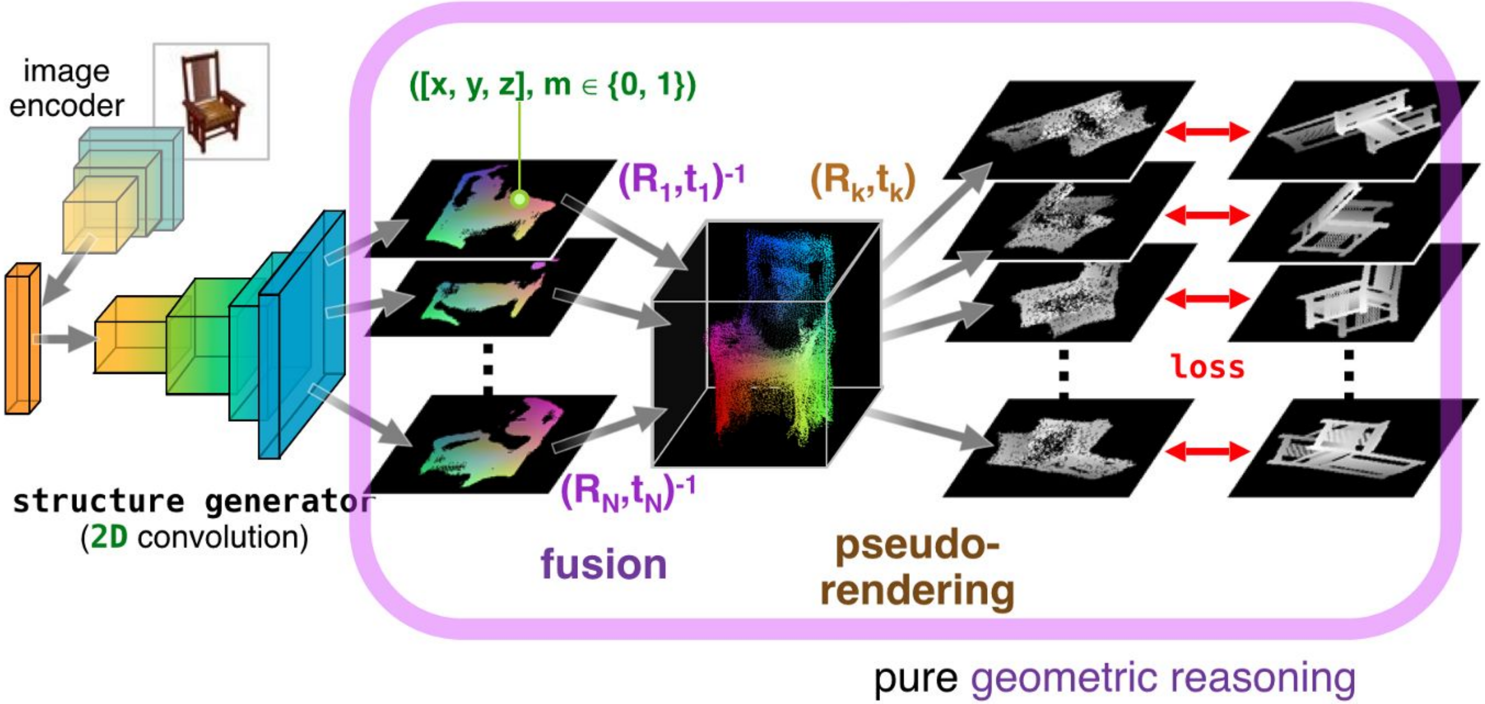
Possible networks



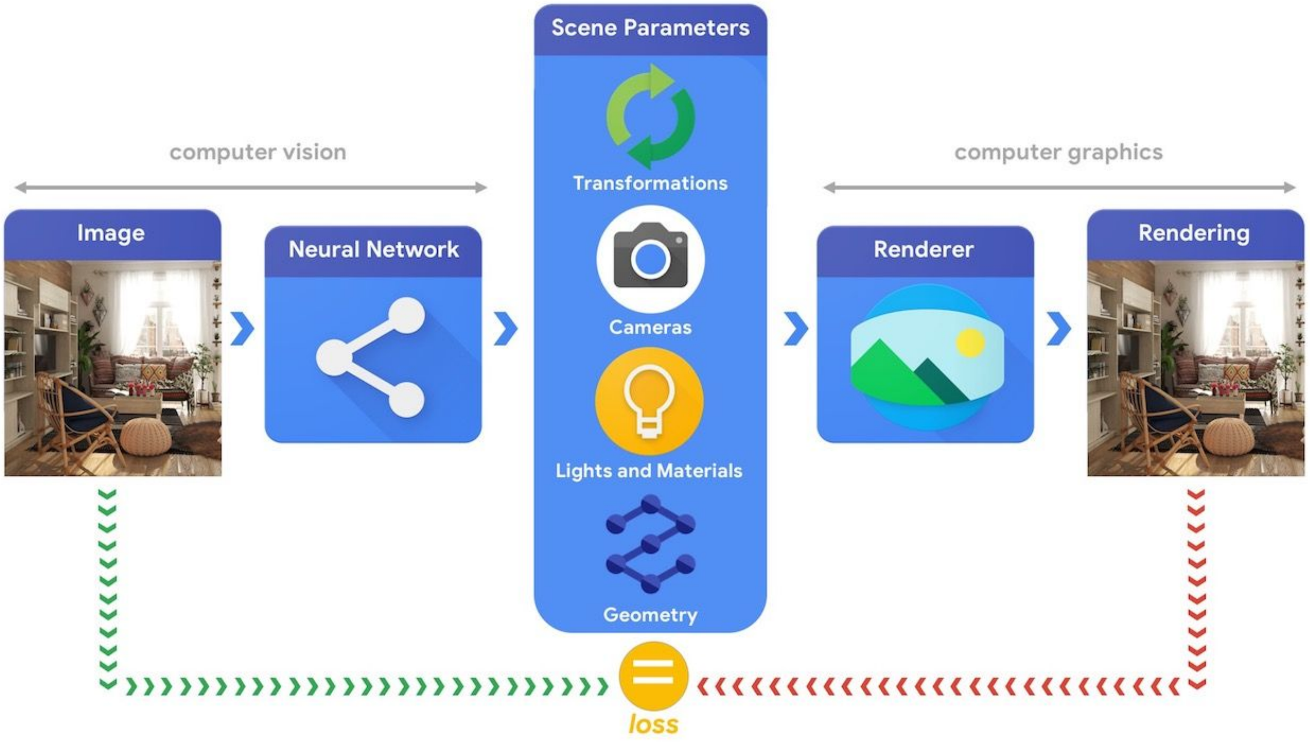
Possible networks



Possible Networks



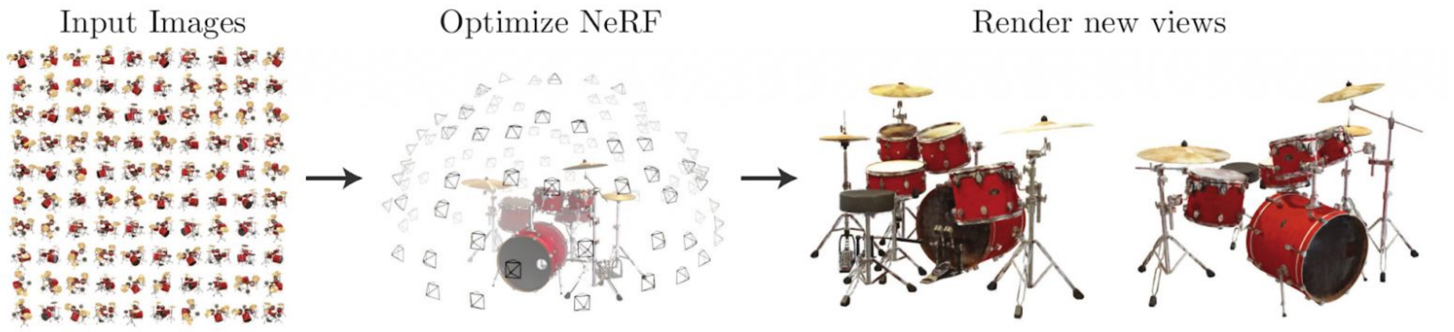
Rendering

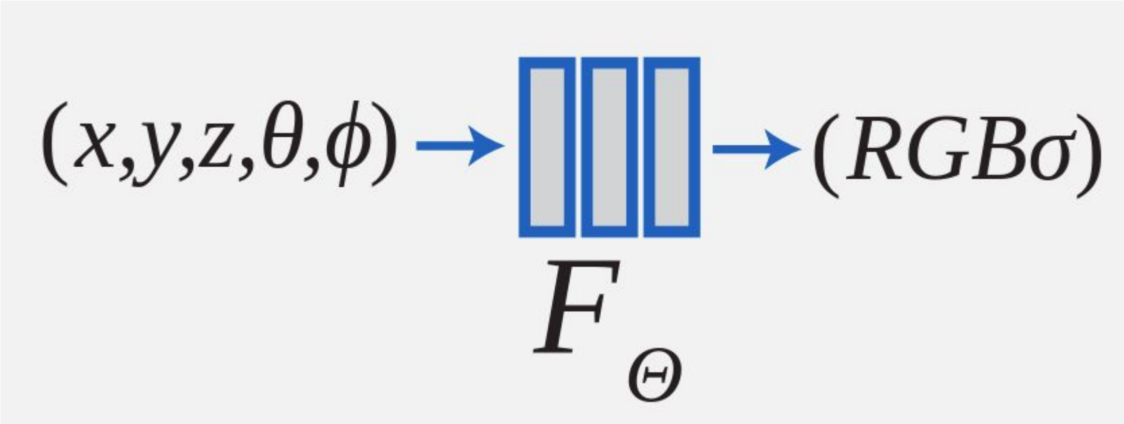


Scene Parameters
Transformations
Cameras
Lights and Materials
Geometry
computer vision
computer graphics
Image
Neural Network
Renderer
Rendering
loss

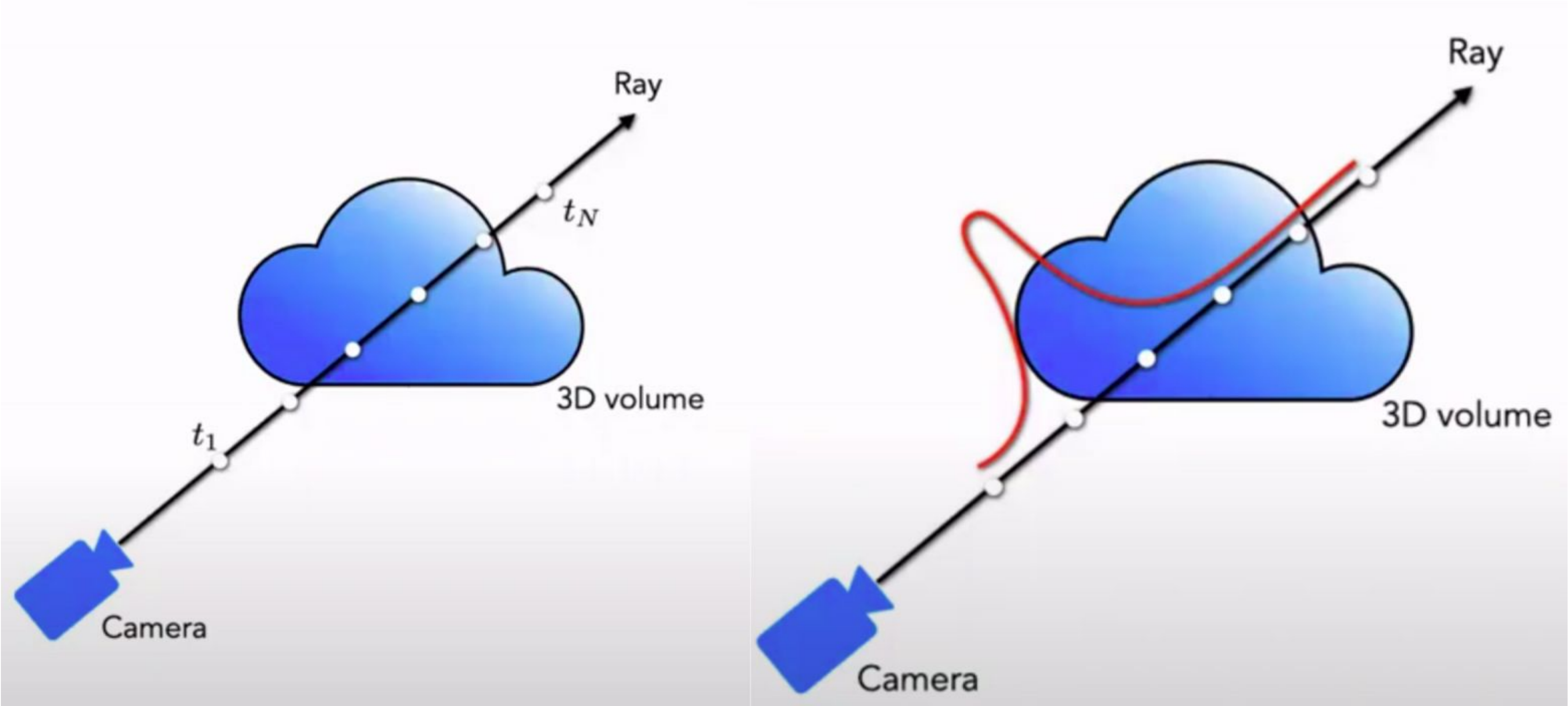
NeRF

NeRF(Neural Radiance Fields)

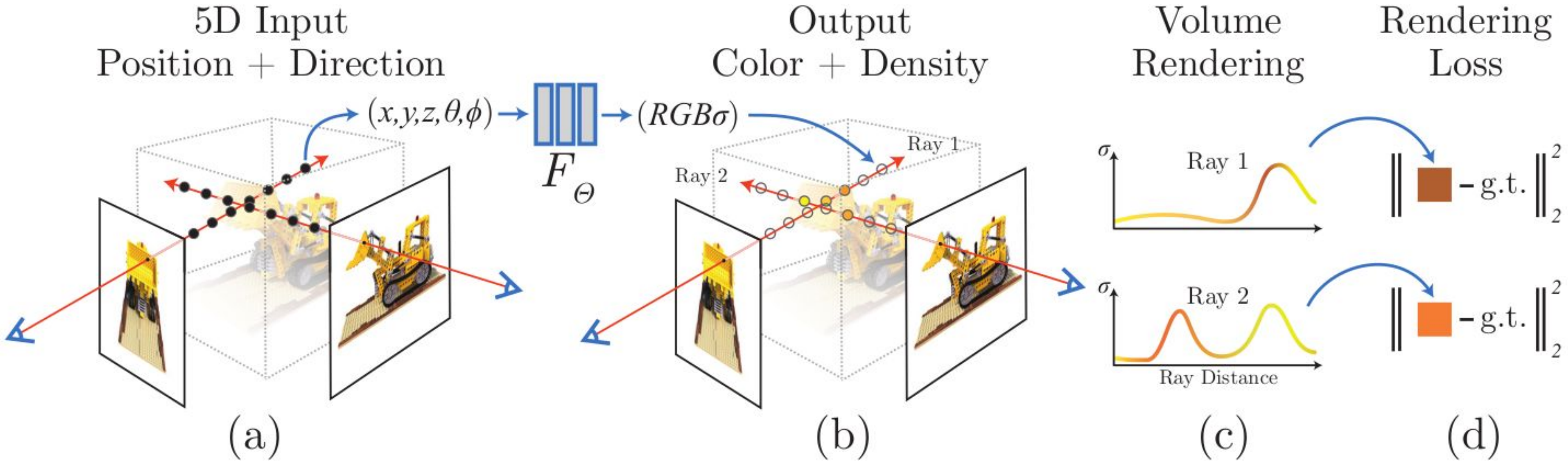




NeRF(Neural Radiance Fields)



NeRF



$$\mathcal{L} = \sum_{\mathbf{r} \in \mathcal{R}} \left[\left\| \hat{C}_c(\mathbf{r}) - C(\mathbf{r}) \right\|_2^2 + \left\| \hat{C}_f(\mathbf{r}) - C(\mathbf{r}) \right\|_2^2 \right]$$

NeRF

Recap

- 3D applications
- 3D data representations
- Some losses for 3D tasks
- One really popular model (NeRF)