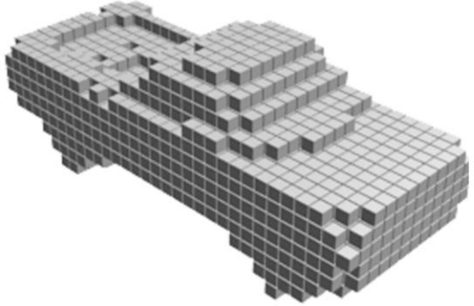
Learning Implicit Representations for Shape Generation and Analysis

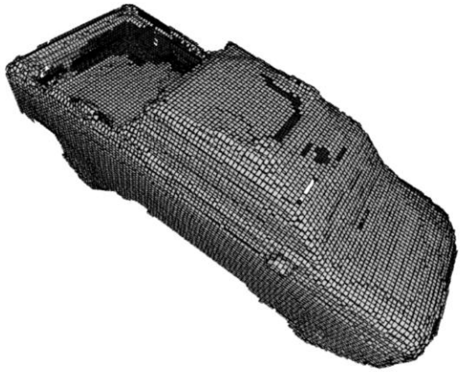
Zhiqin Chen, Ph.D. student, Simon Fraser University

3D shape representations

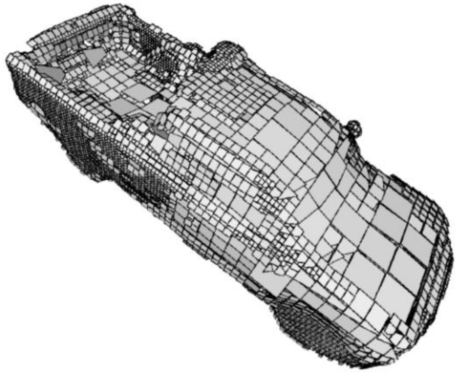
(a) Mesh



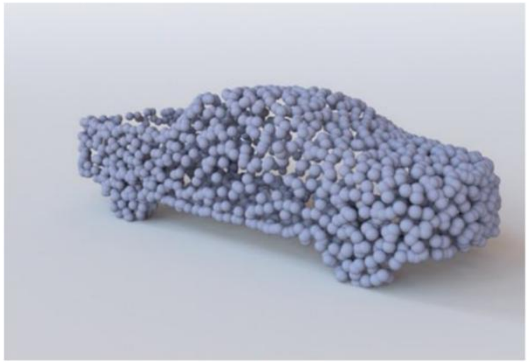
(b) Voxel [1]



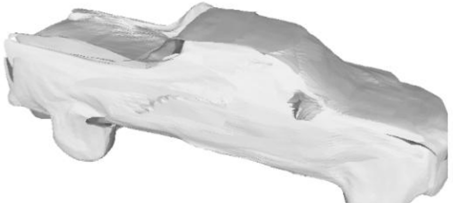
(c) O-CNN [2]
(octree-based voxel)



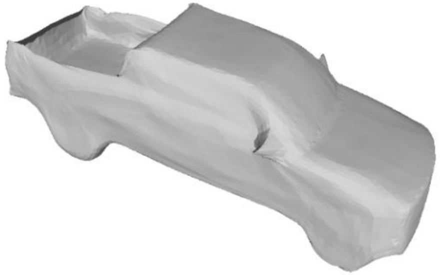
(d) Adaptive O-CNN [3]
(octree-based patch)



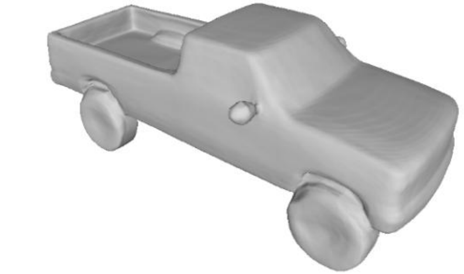
(e) Point cloud [4]



(f) AtlasNet [5]
(25 square patches)

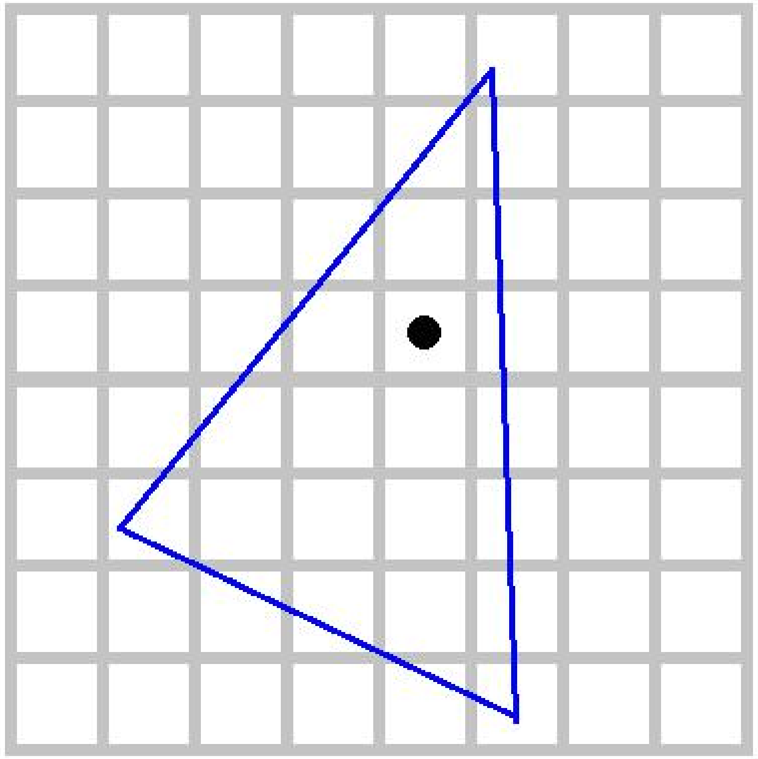


(g) AtlasNet [5]
(1 sphere patch)

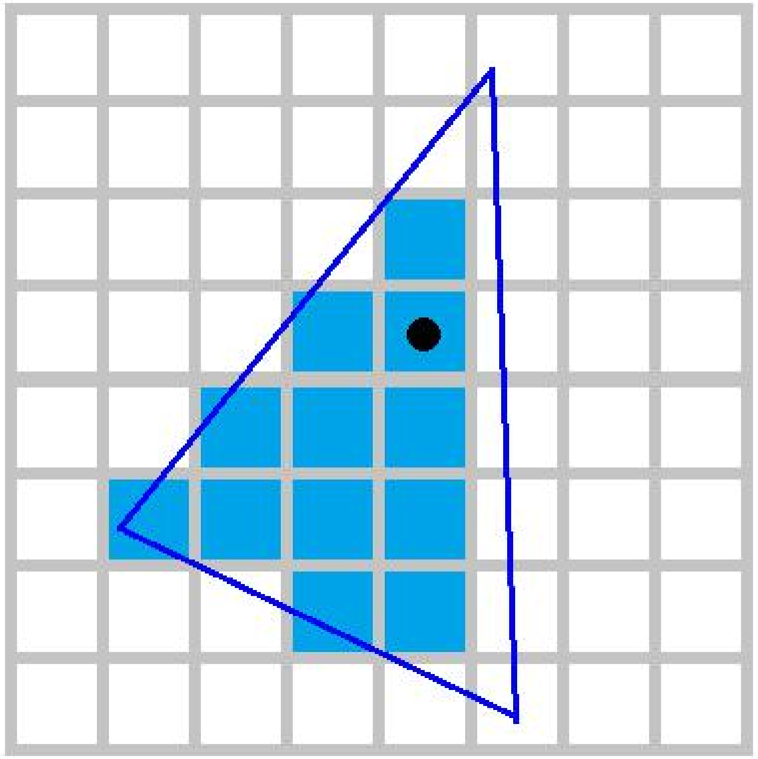


(h) Ours (IM-NET) [6]
(implicit field)

How implicit field works?

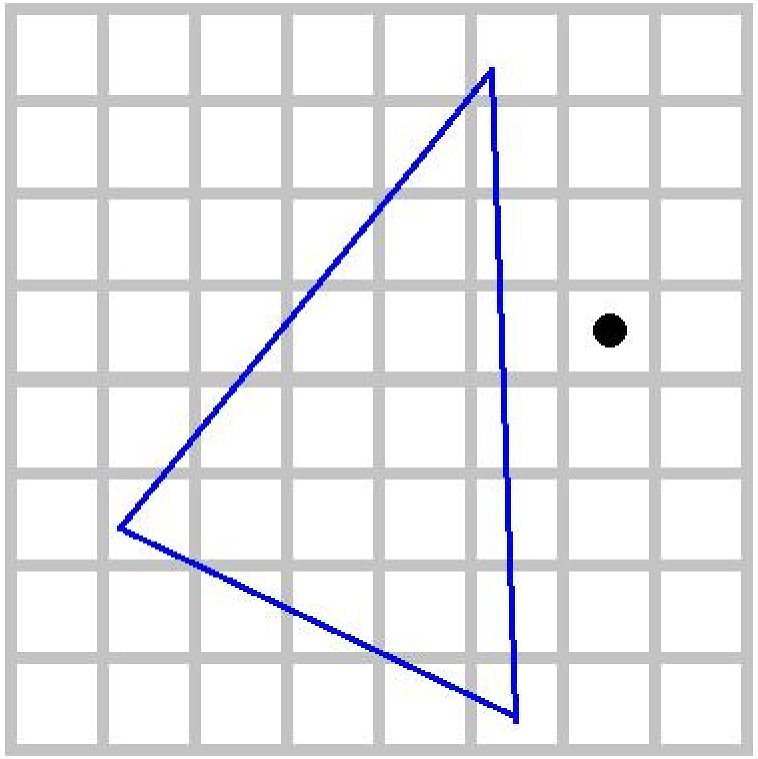


Input:
point coordinates = (5,4)

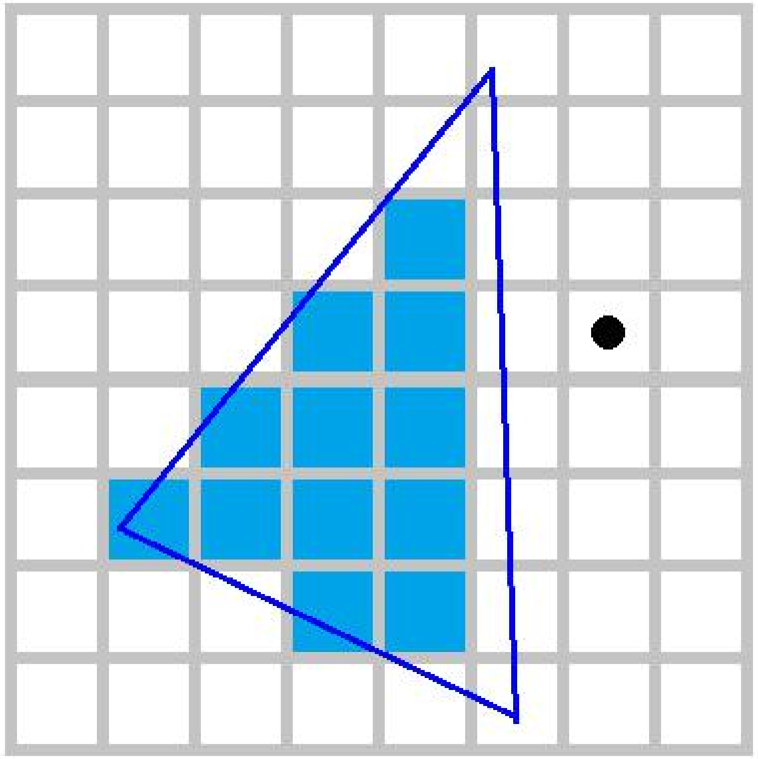


Output:
value = 1

How implicit field works?

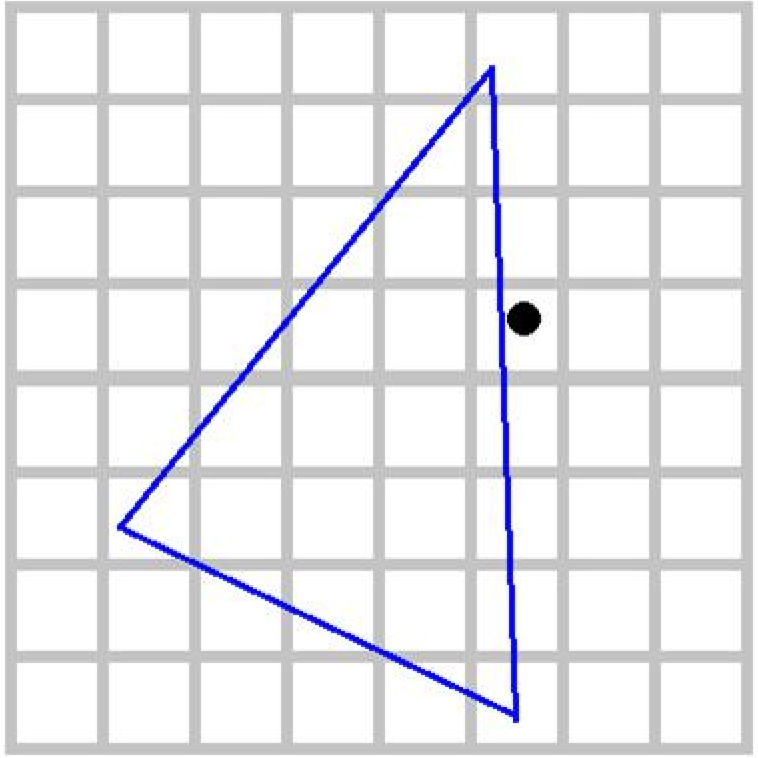


Input:
point coordinates = (7,4)

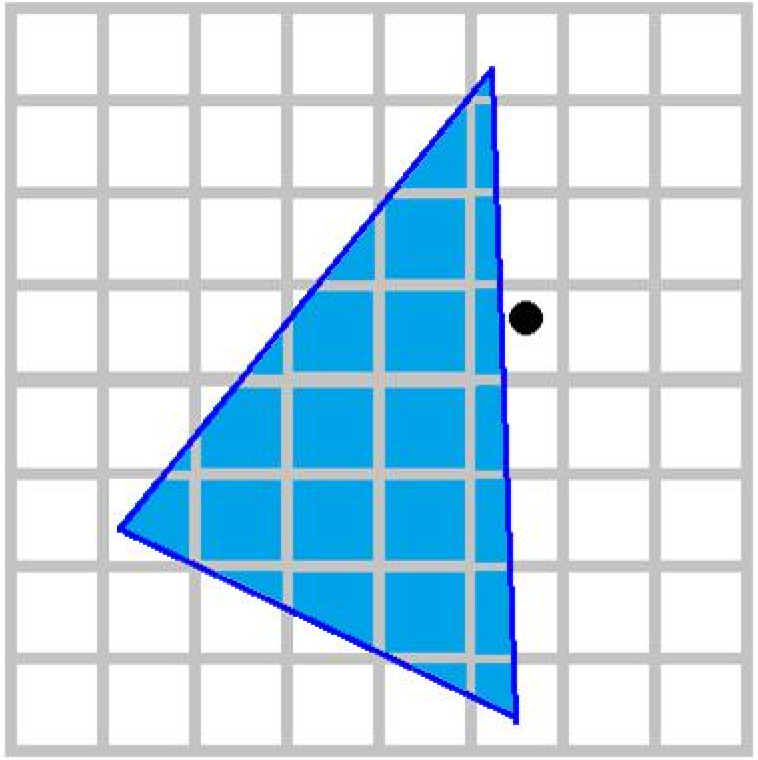


Output:
value = 0

How implicit field works?



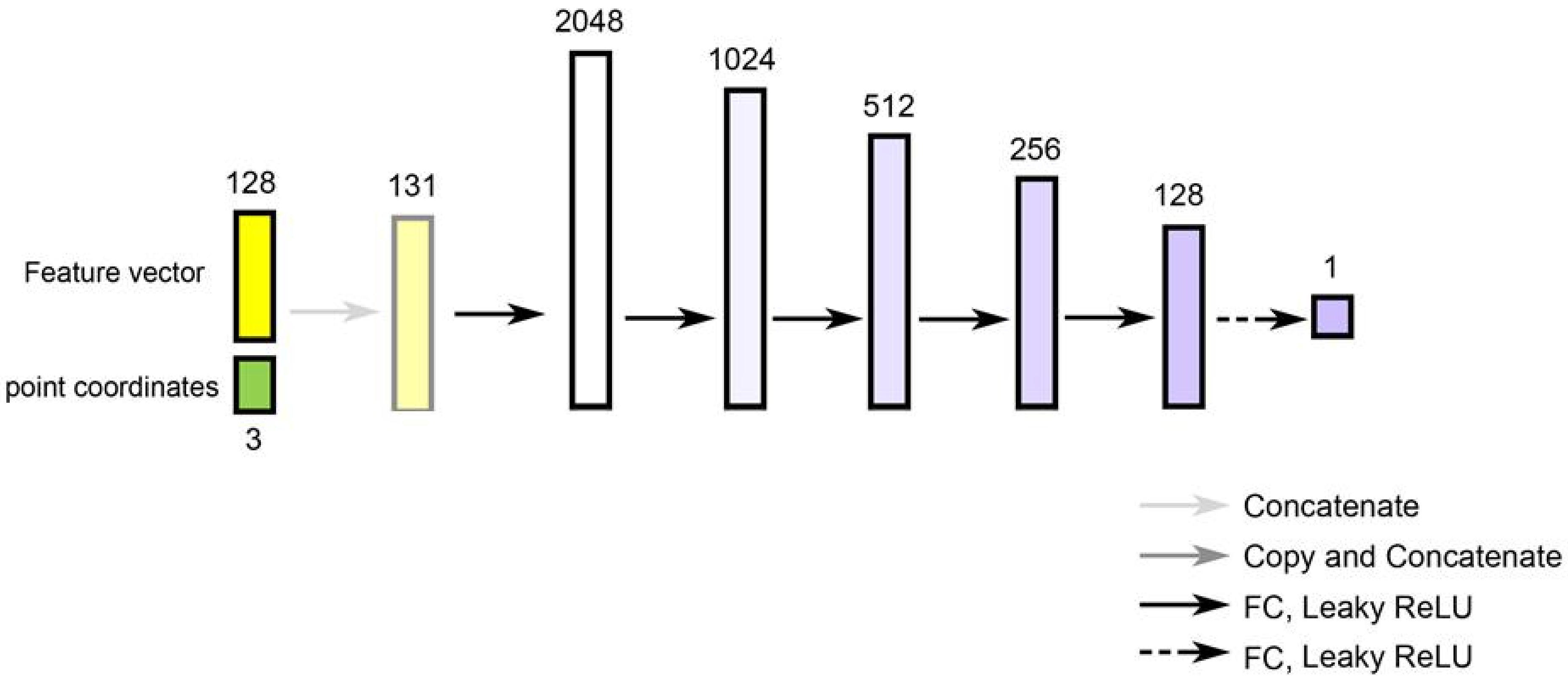
Input:
point coordinates = (5.6,3.3)



Output:
value = 0

Network - implicit field decoder

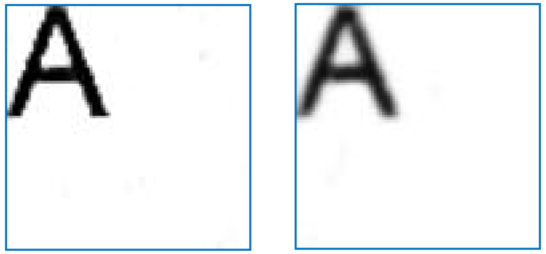
- As a whole: Input shape feature code, output a field function.



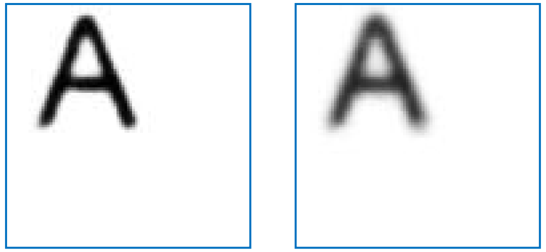
Toy 2D experiments

Toy 2D experiments

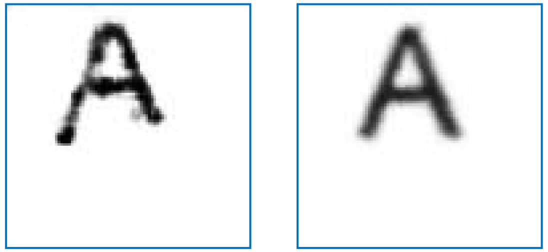
AE trained with CNN decoder:



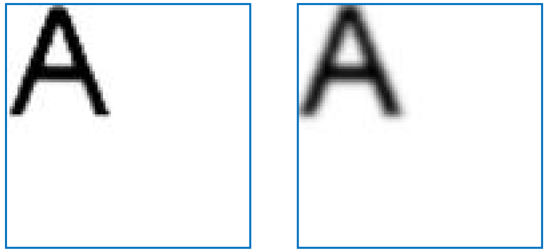
VAE trained with CNN decoder:



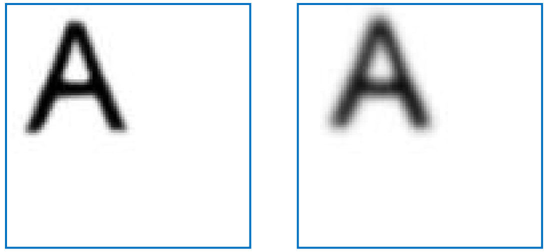
WGAN trained with CNN decoder:



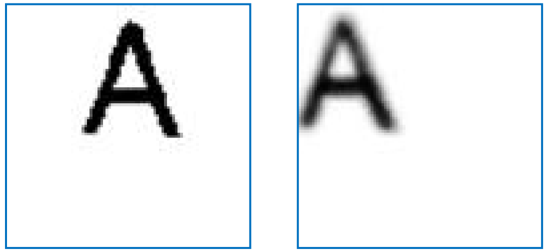
AE trained with implicit decoder:



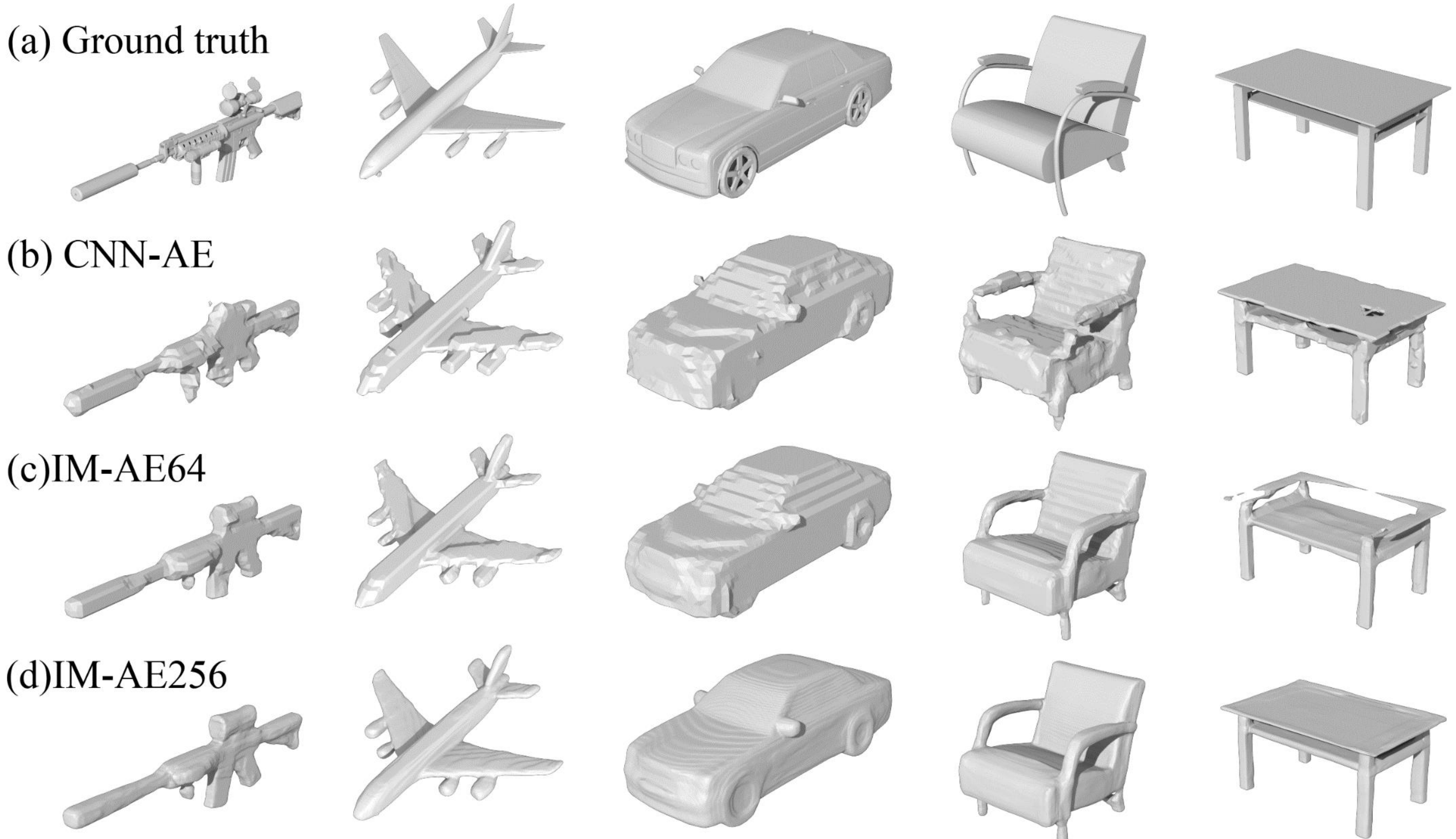
VAE trained with implicit decoder:



WGAN trained with implicit decoder:

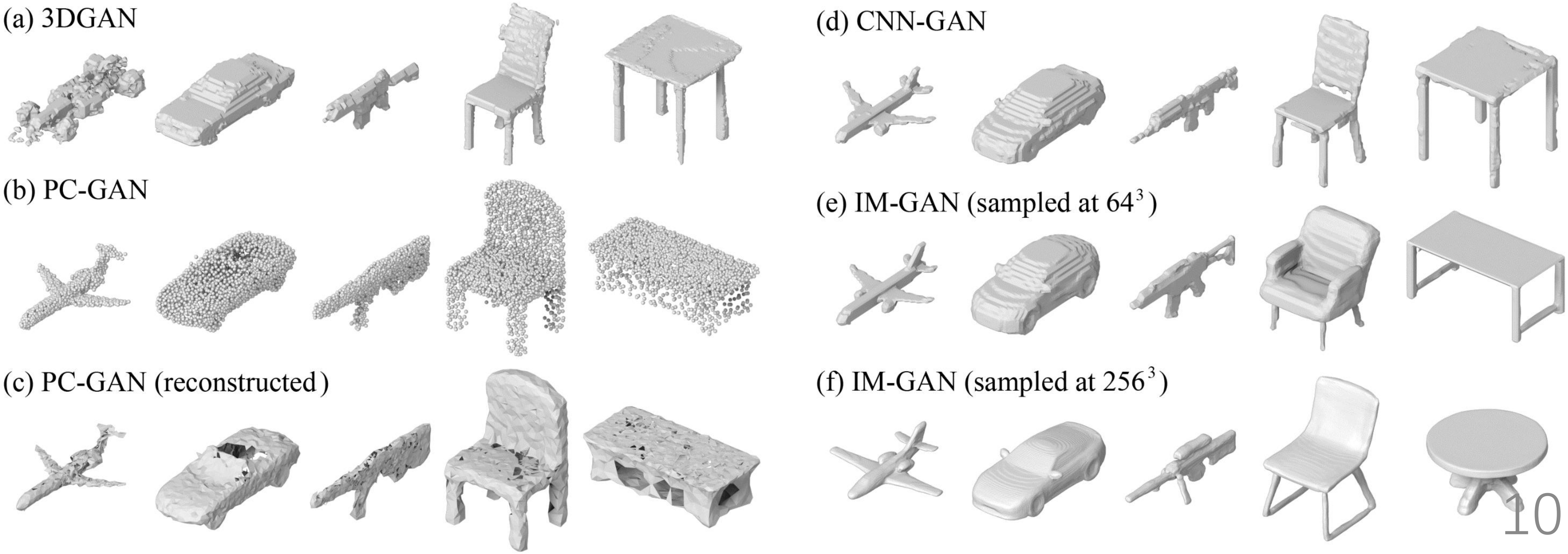


3D shape autoencoder (IM-AE) results

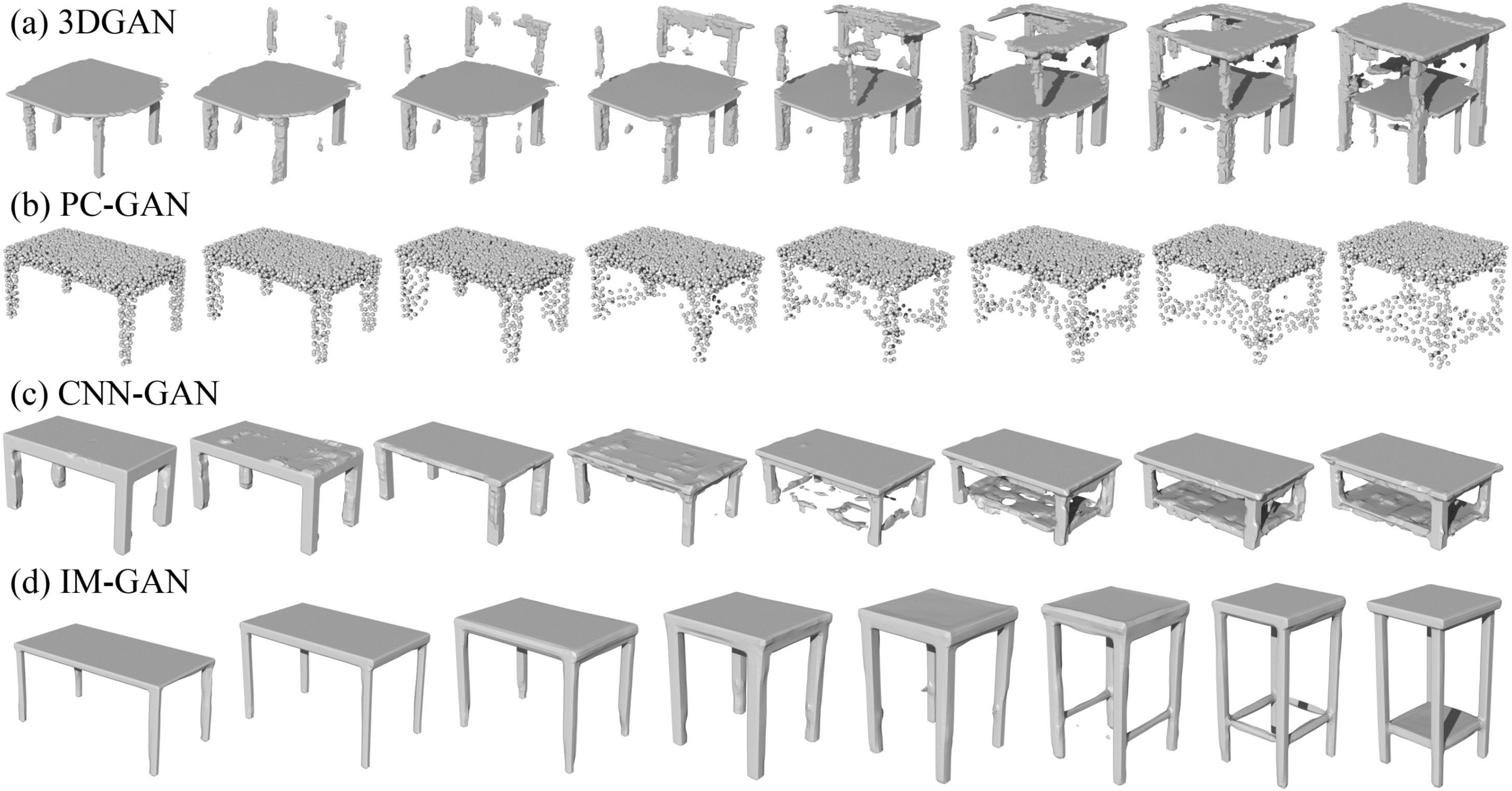


IM-GAN results

- Based on trained models of IM-AE, we train GANs on the latent codes, namely, latent GANs.

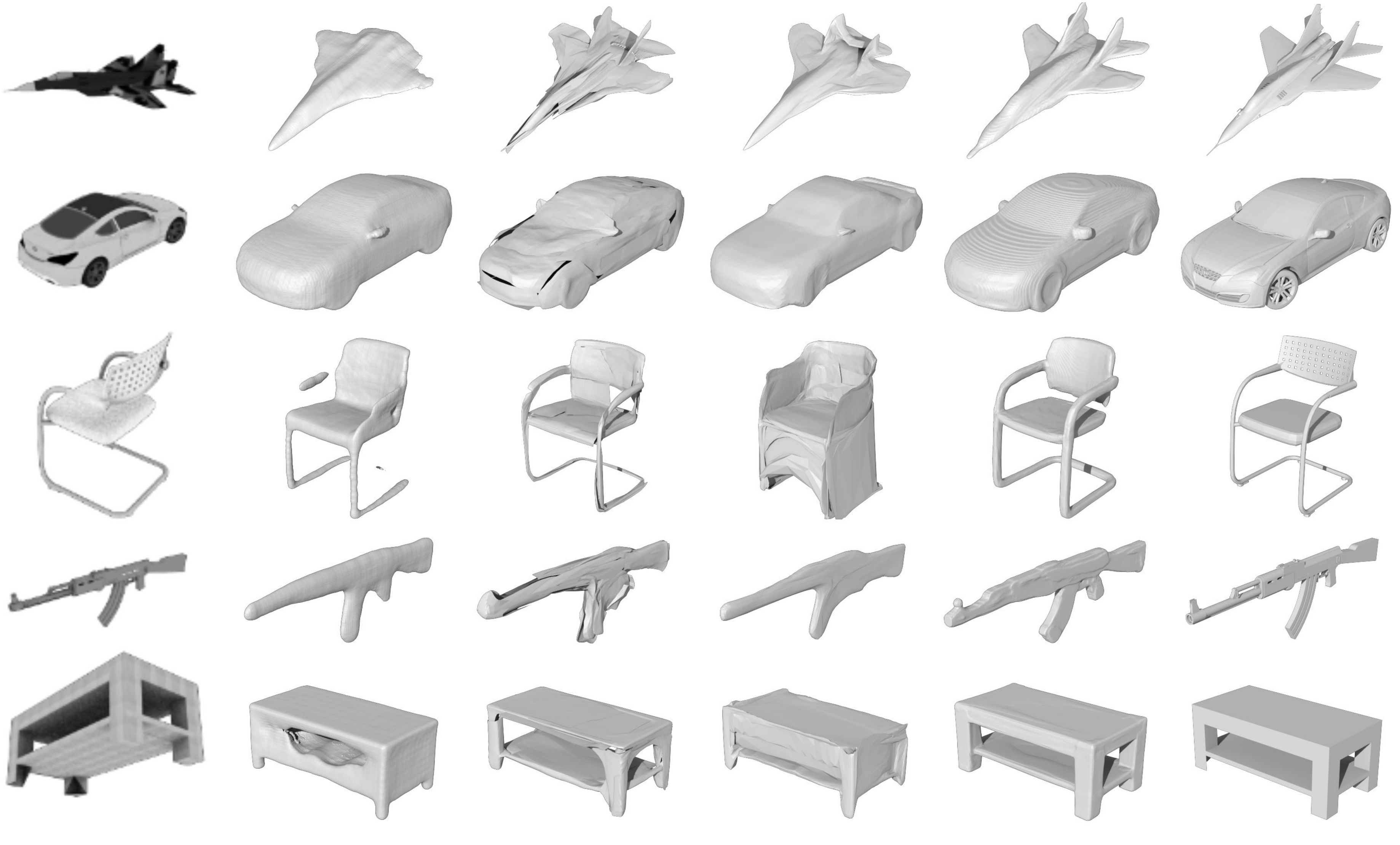


IM-GAN interpolation



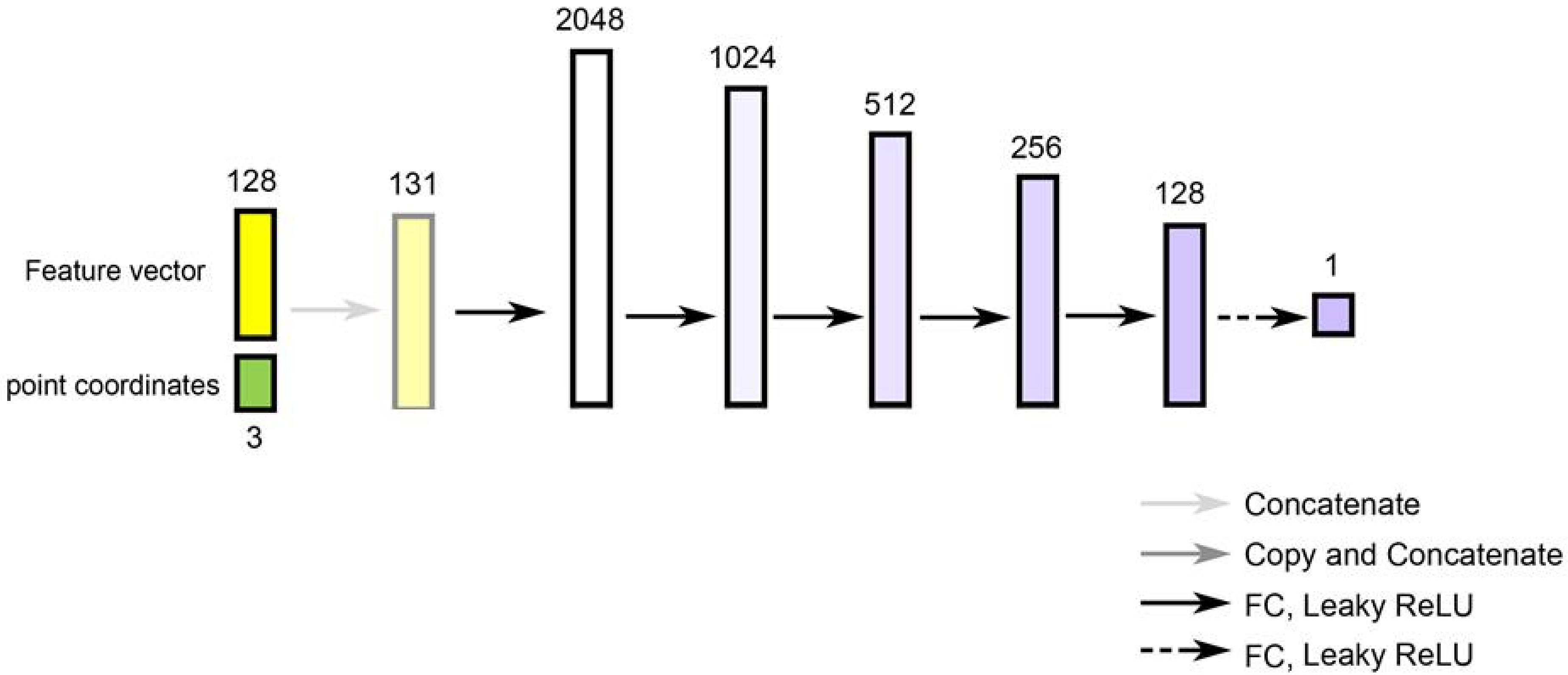
IM-GAN interpolation

IM-SVR for single view image reconstruction

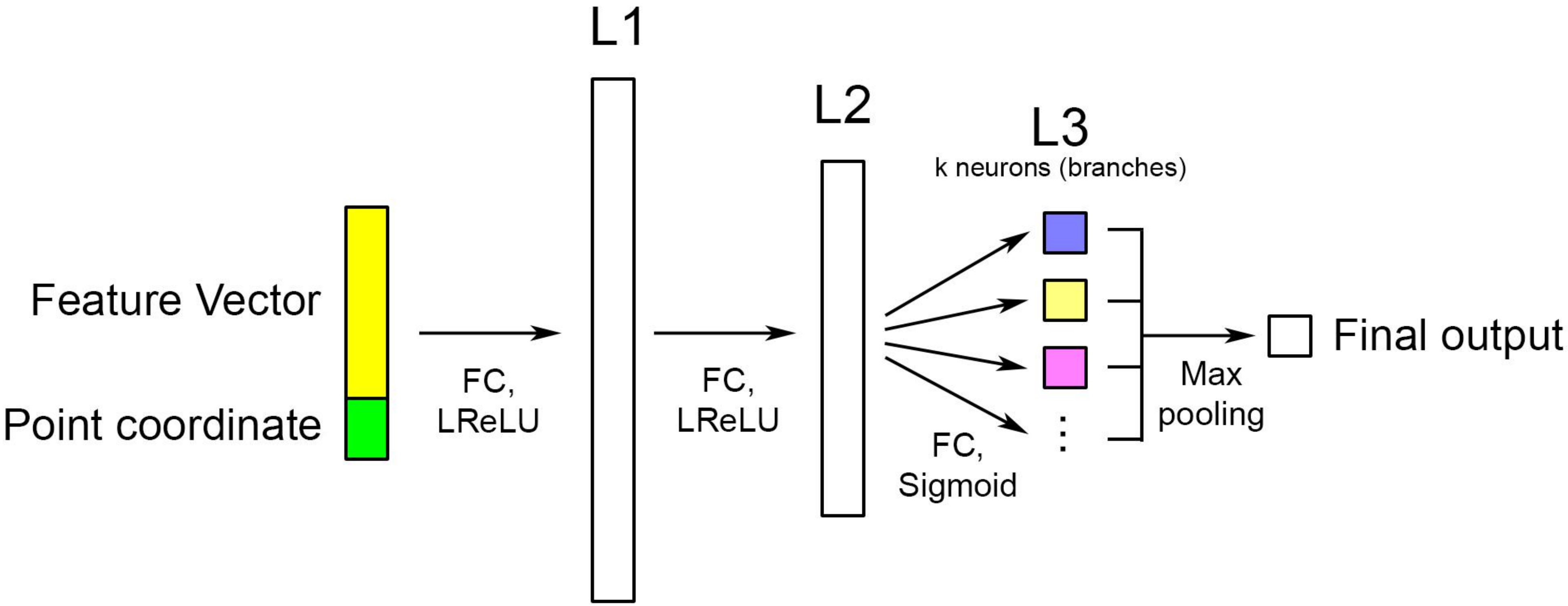


(a) Input image (b) HSP (c) AtlasNet25 (d) AtlasNetO (e) IM-SVR (e) Ground Truth

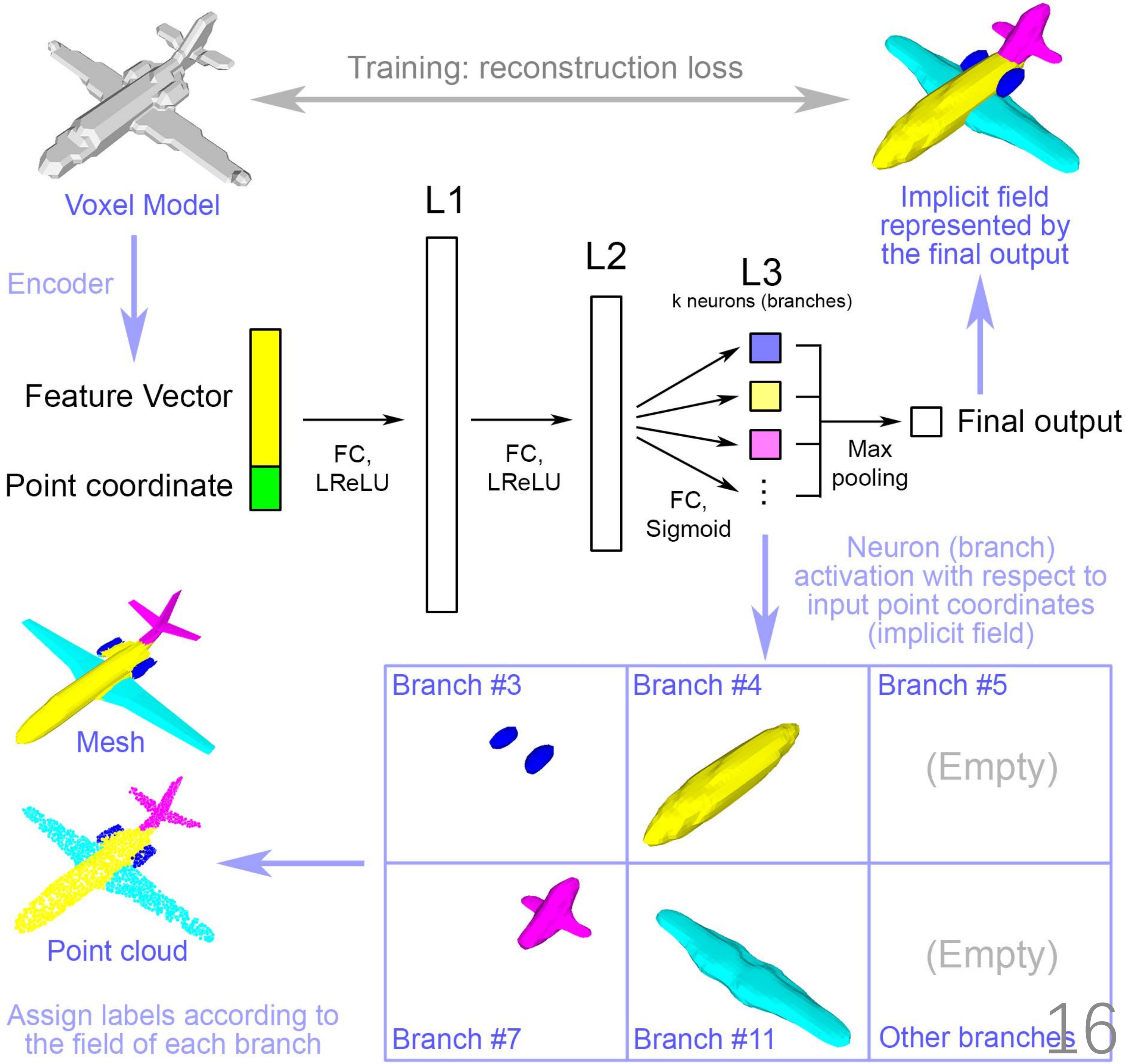
IM-NET with a single output



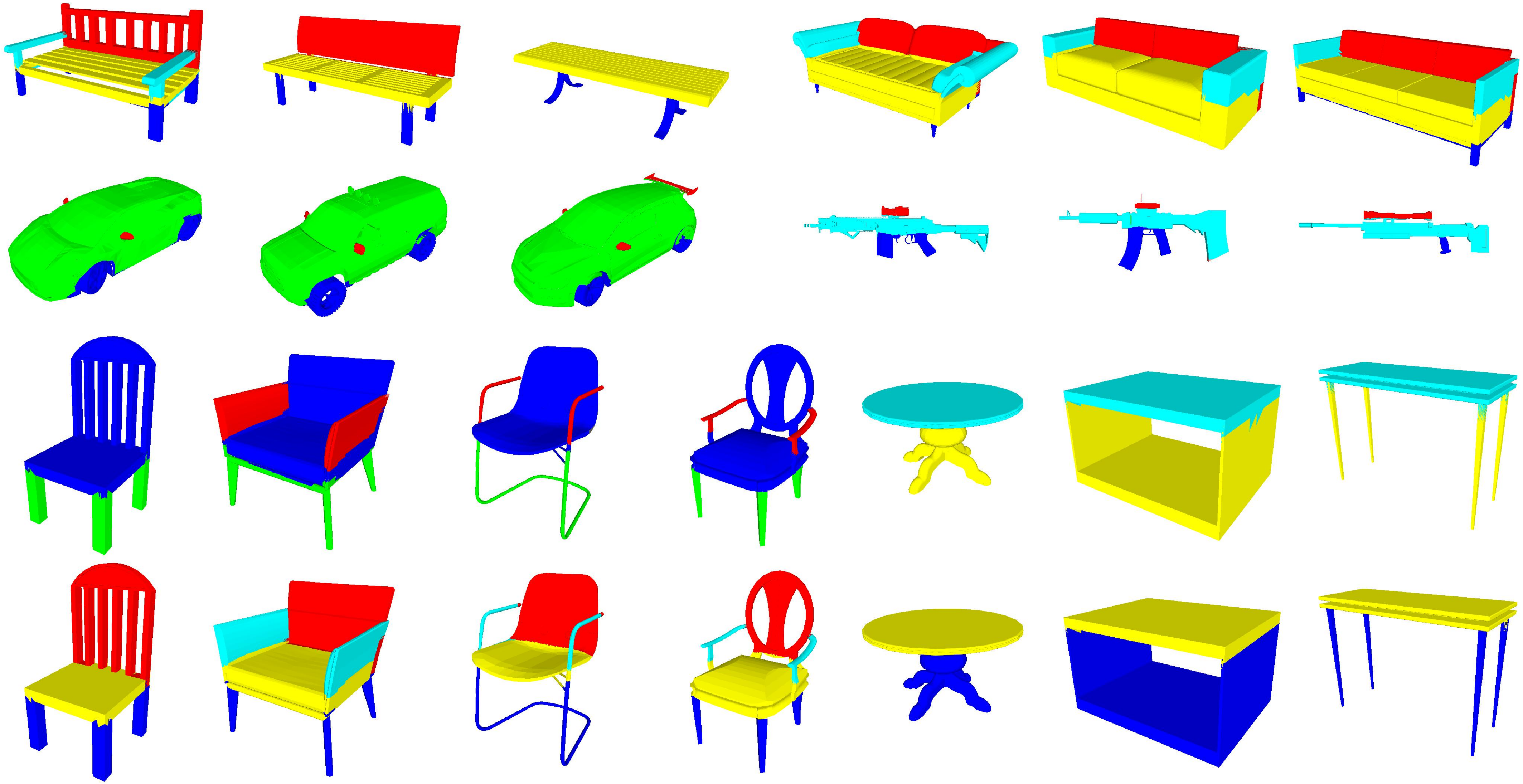
IM-NET with branched outputs



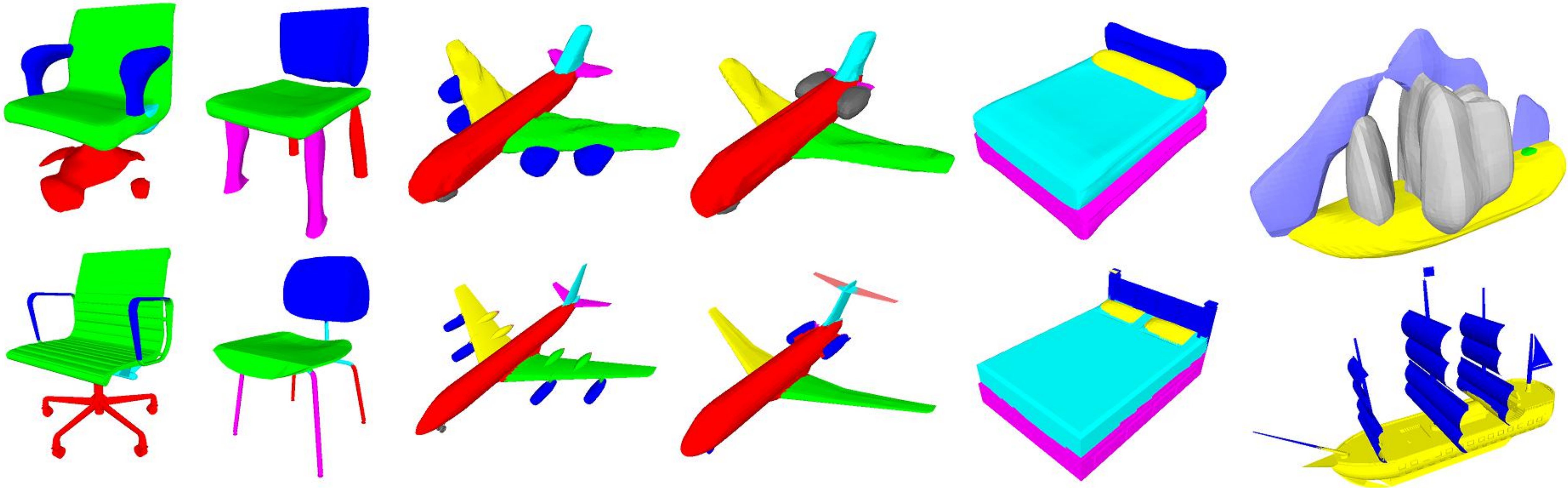
BAE-NET:
Branched Autoencoder for Shape Co-Segmentation



Unsupervised shape co-segmentation



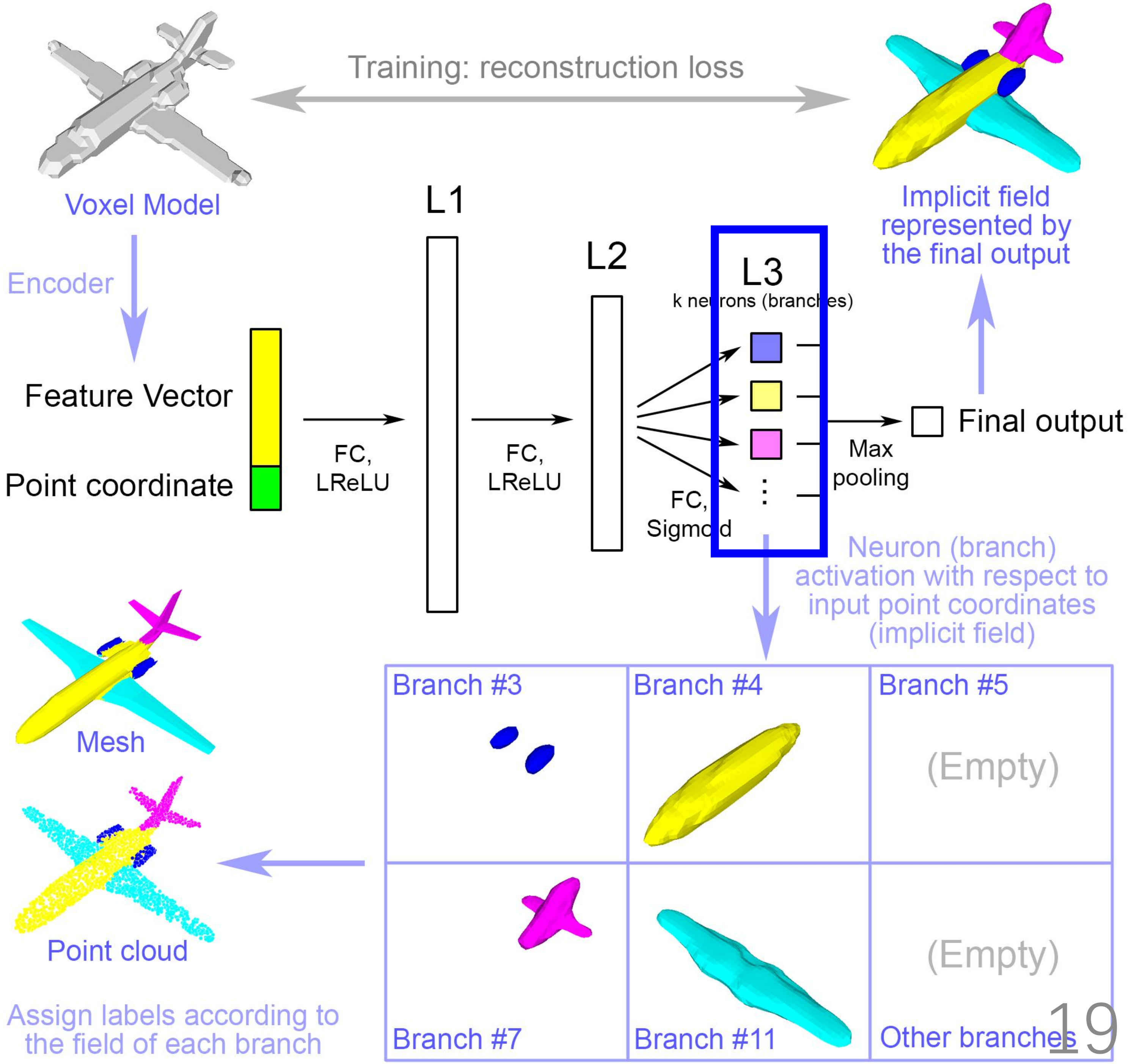
Weakly-supervised shape co-segmentation



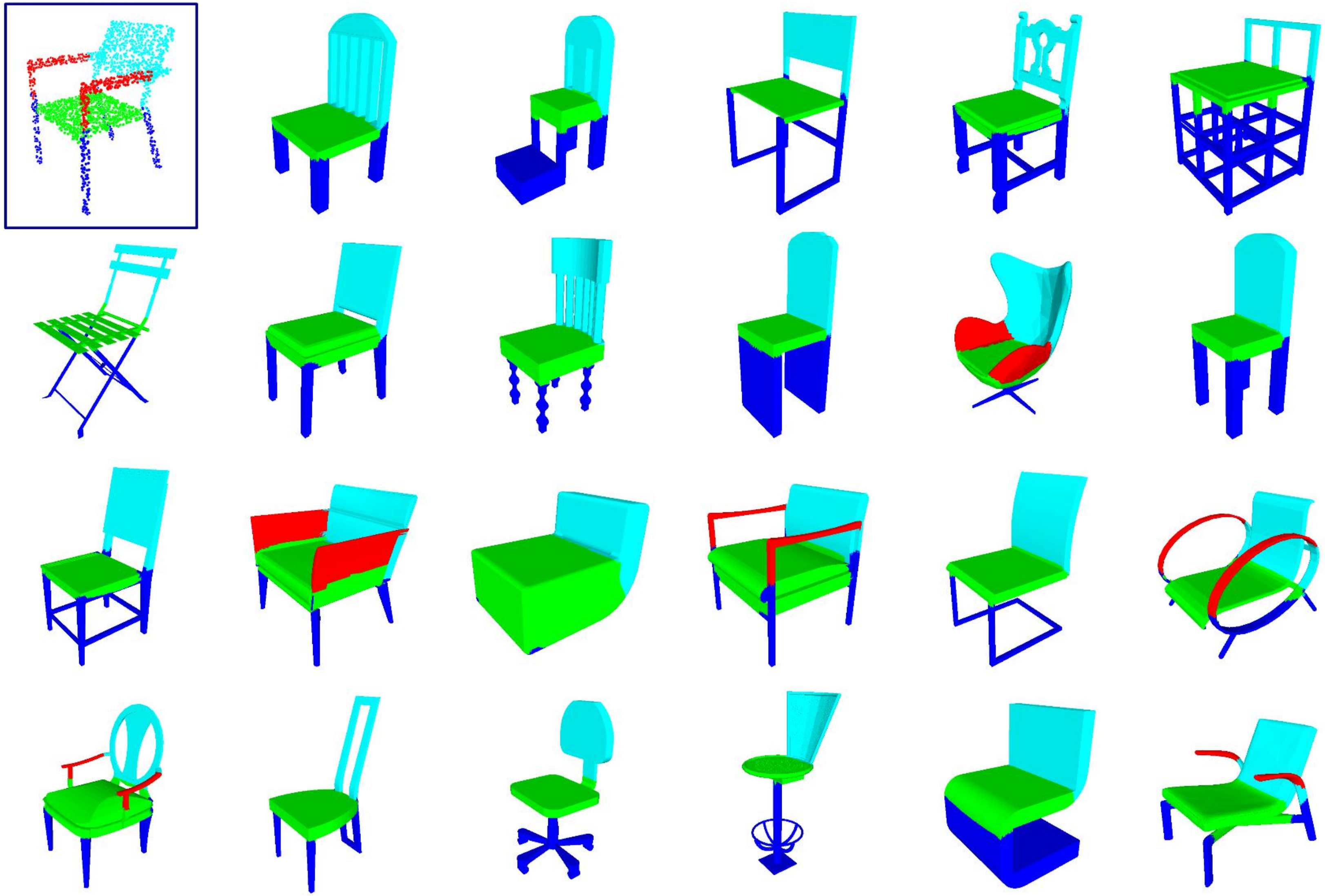
(a) Armrest (chair) (b) Back (chair) (c) Engine (airplane) (d) Head (bed) (d) Sail (ship)

One-shot training

- Give the network a few segmented shapes.
- Supervised loss on those shapes and reconstruction loss on others.



One-shot semantic segmentation



One-shot semantic segmentation

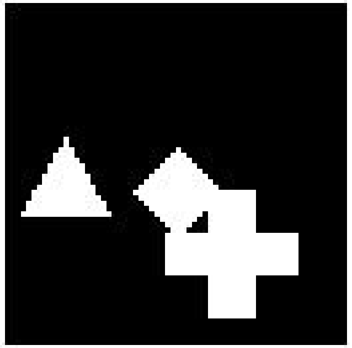
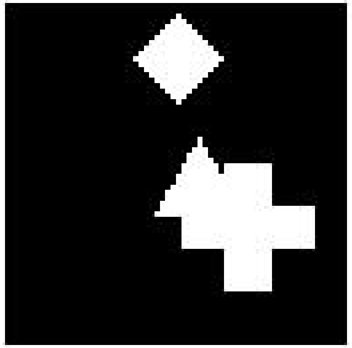
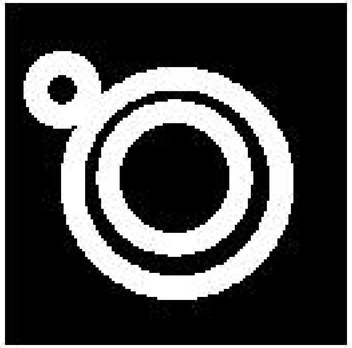
	1-exem. vs. 10% train set	2-exem. vs. 20% train set	3-exem. vs. 30% train set
Pointnet [42]	72.1	73.0	74.6
Pointnet++ [43]	73.5	75.4	76.6
PointCNN [30]	58.0	65.6	65.7
SSCN [13]	56.7	61.0	64.6
Our BAE-NET	**76.6**	**77.6**	**78.7**

Table 3. Quantitative comparison to supervised methods by average IOU over 15 shape categories, without combining parts in the ground truth. Our one-shot learning with 1/2/3 exemplars outperforms supervised methods trained on 10%/20%/30% of the shapes, respectively (on average each category has 765 training shapes).

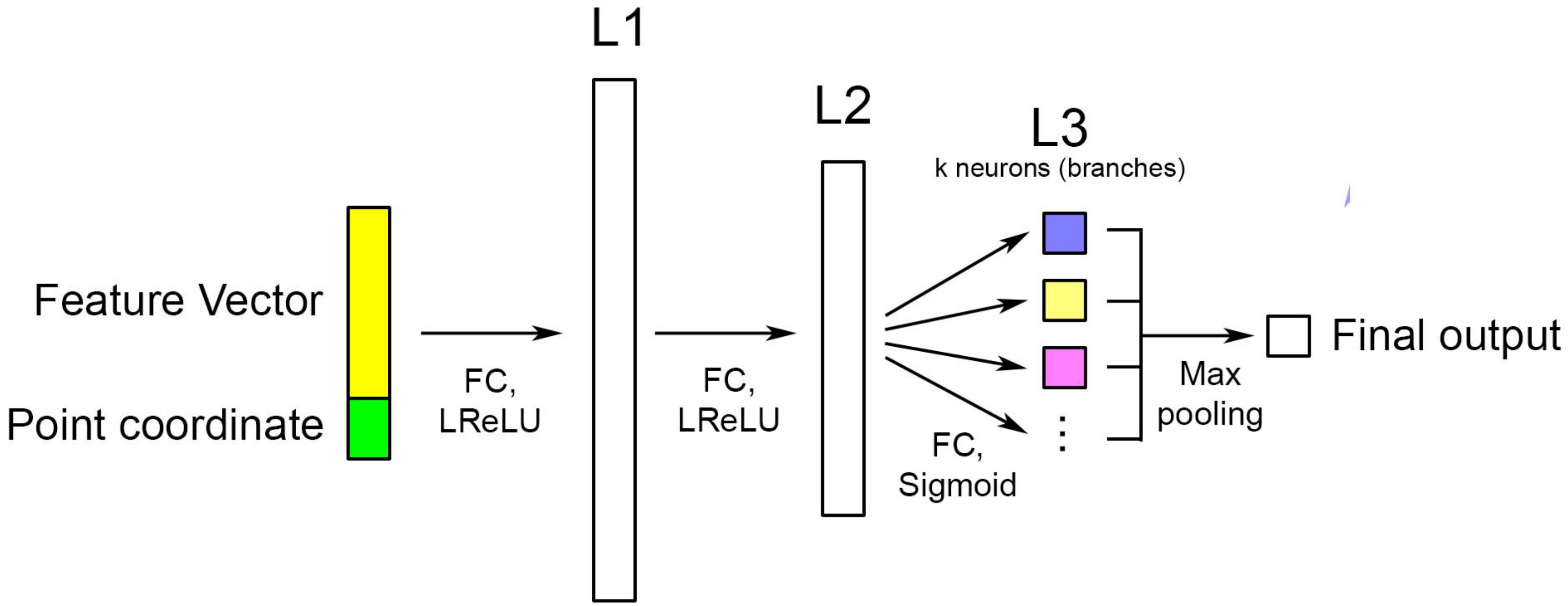
Why is it working?

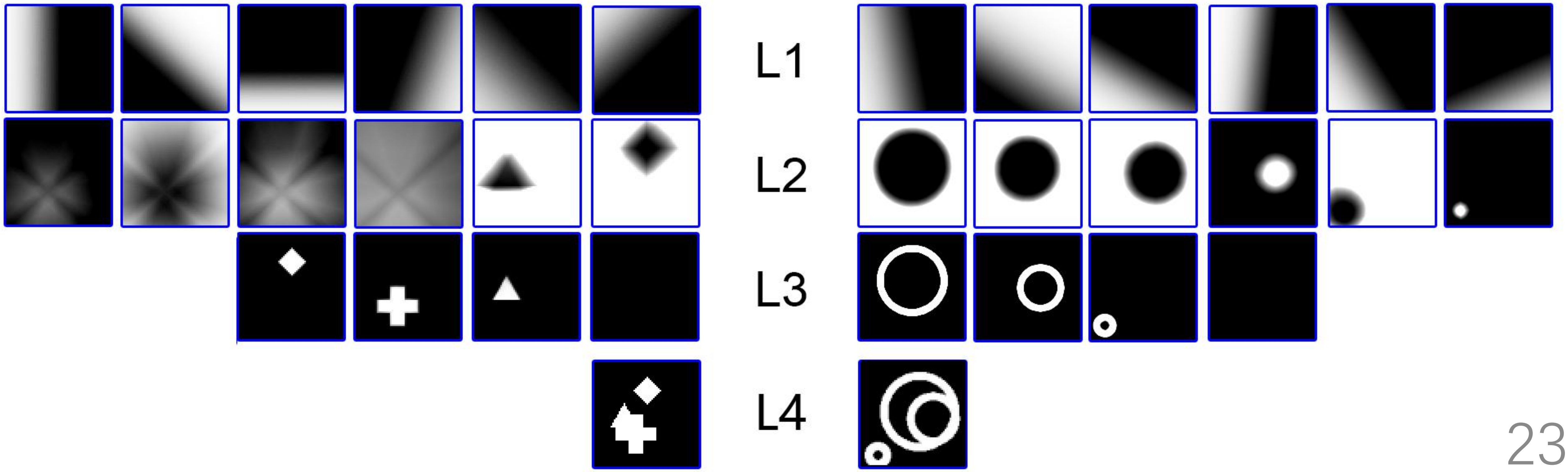
- The interpretability of our network.

- To show that, we have done toy experiment on two synthetic datasets: "Elements" and "Triple rings".

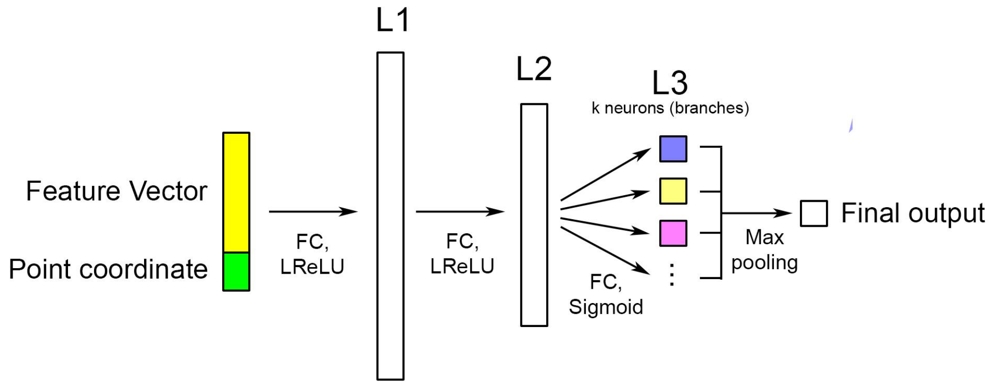




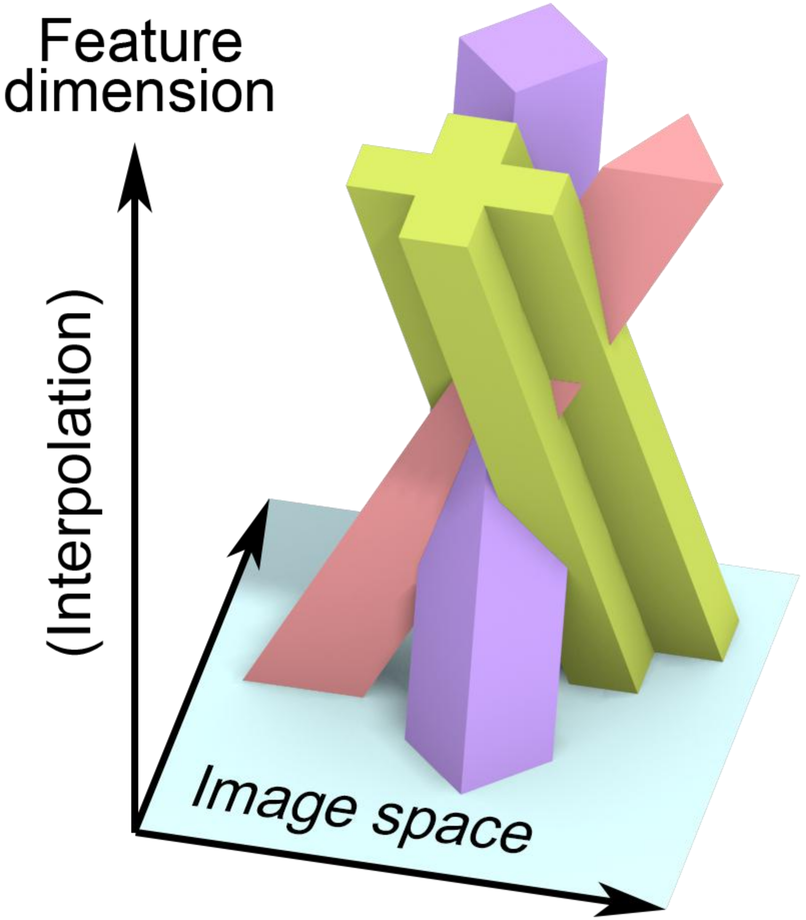

Interpretability



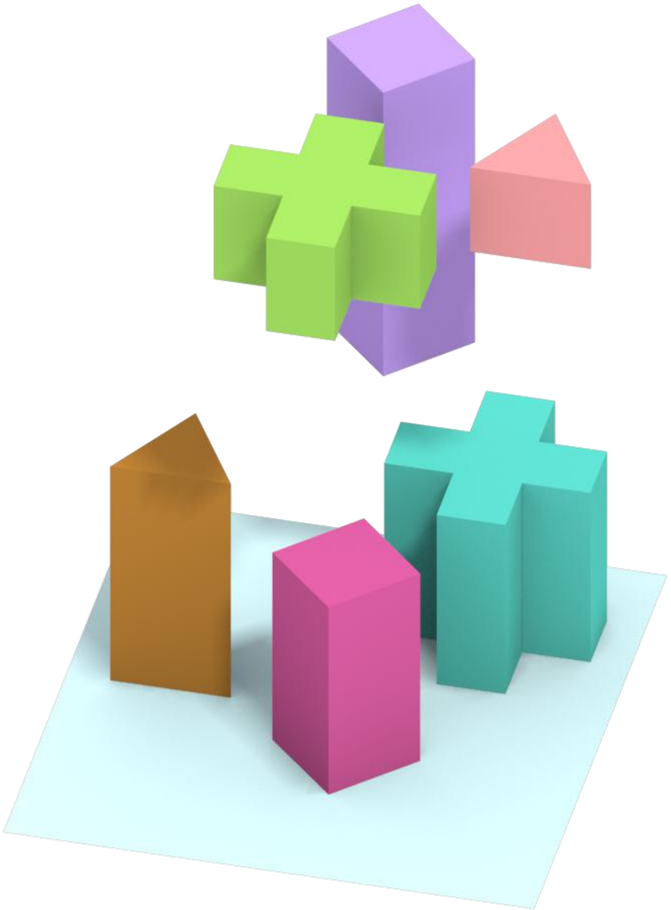


Hyperdimensional shape



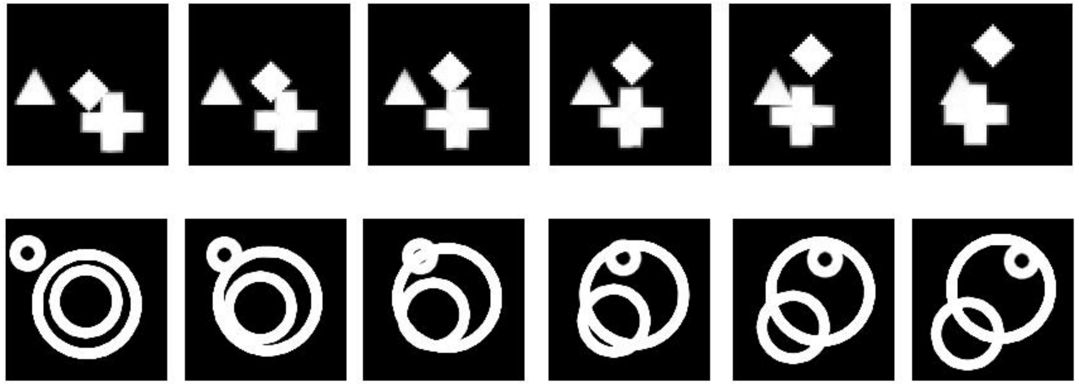


(a) Joint space of our model

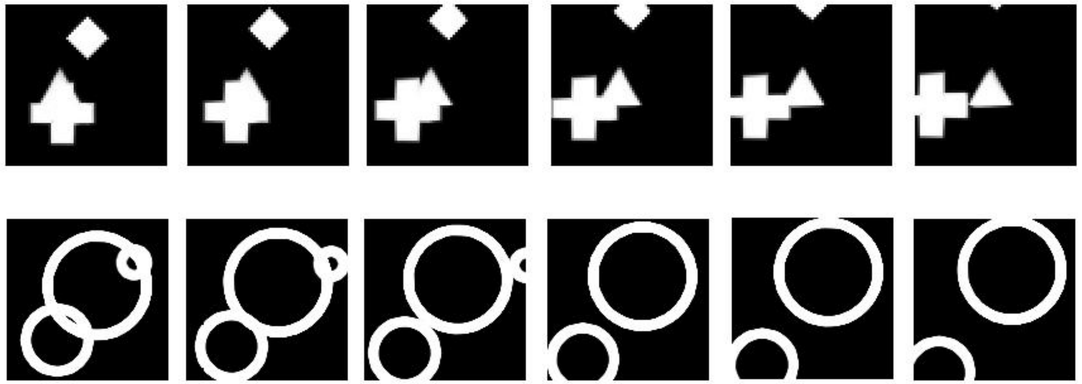


(b) Joint space of a CNN AE

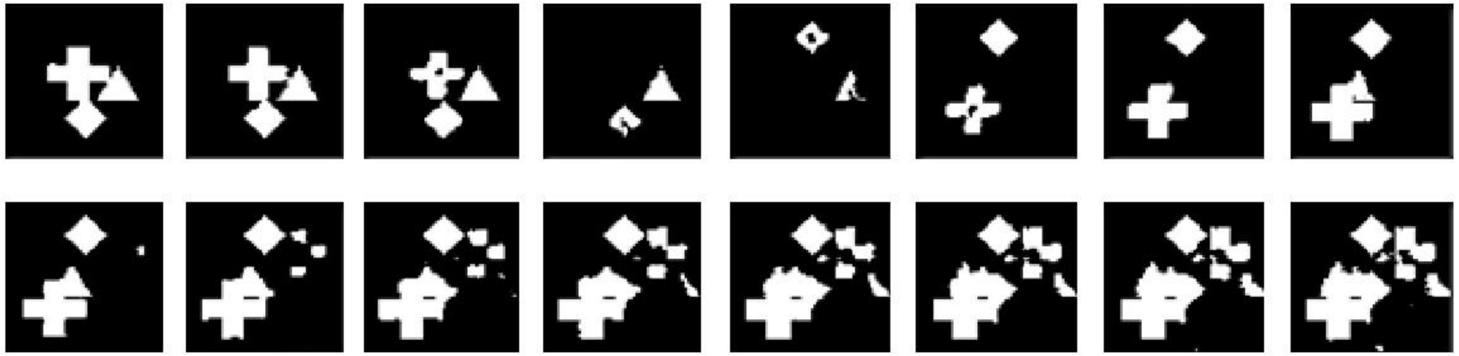
Interpolations



3-layer
interpolation



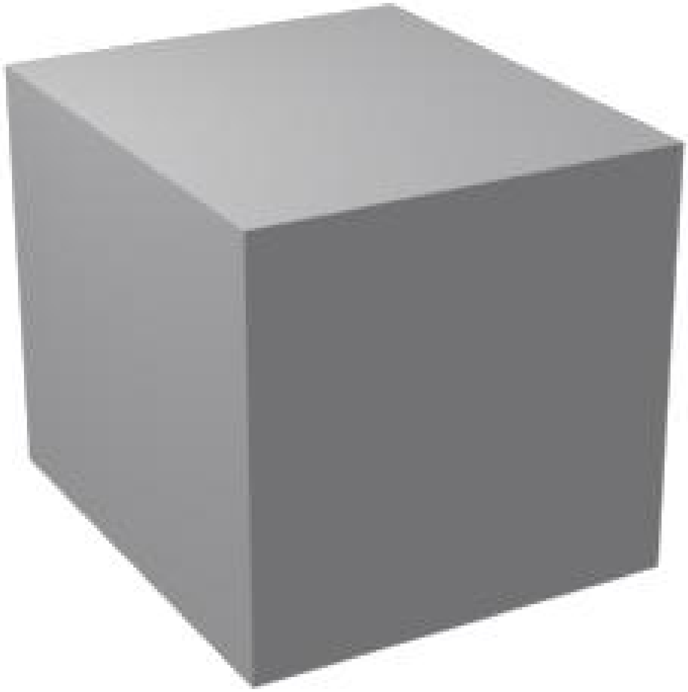
3-layer
extrapolation



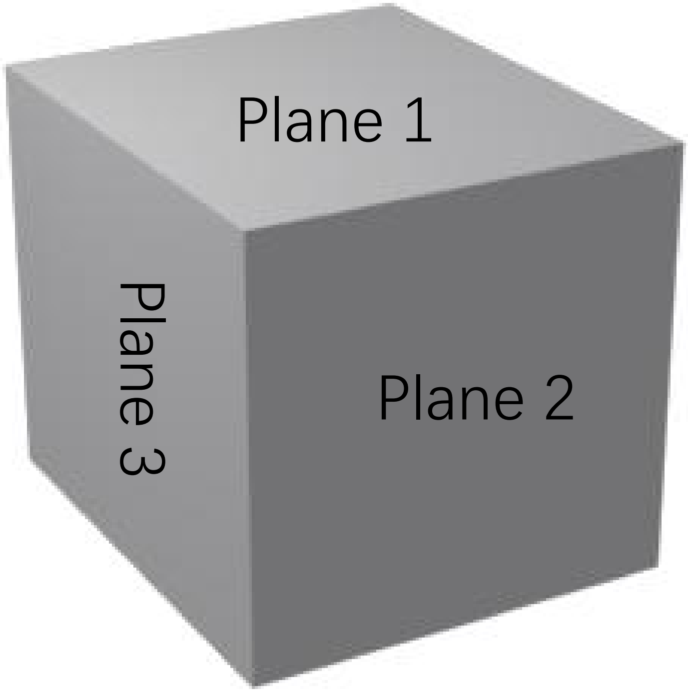
CNN interpolation

CNN extrapolation

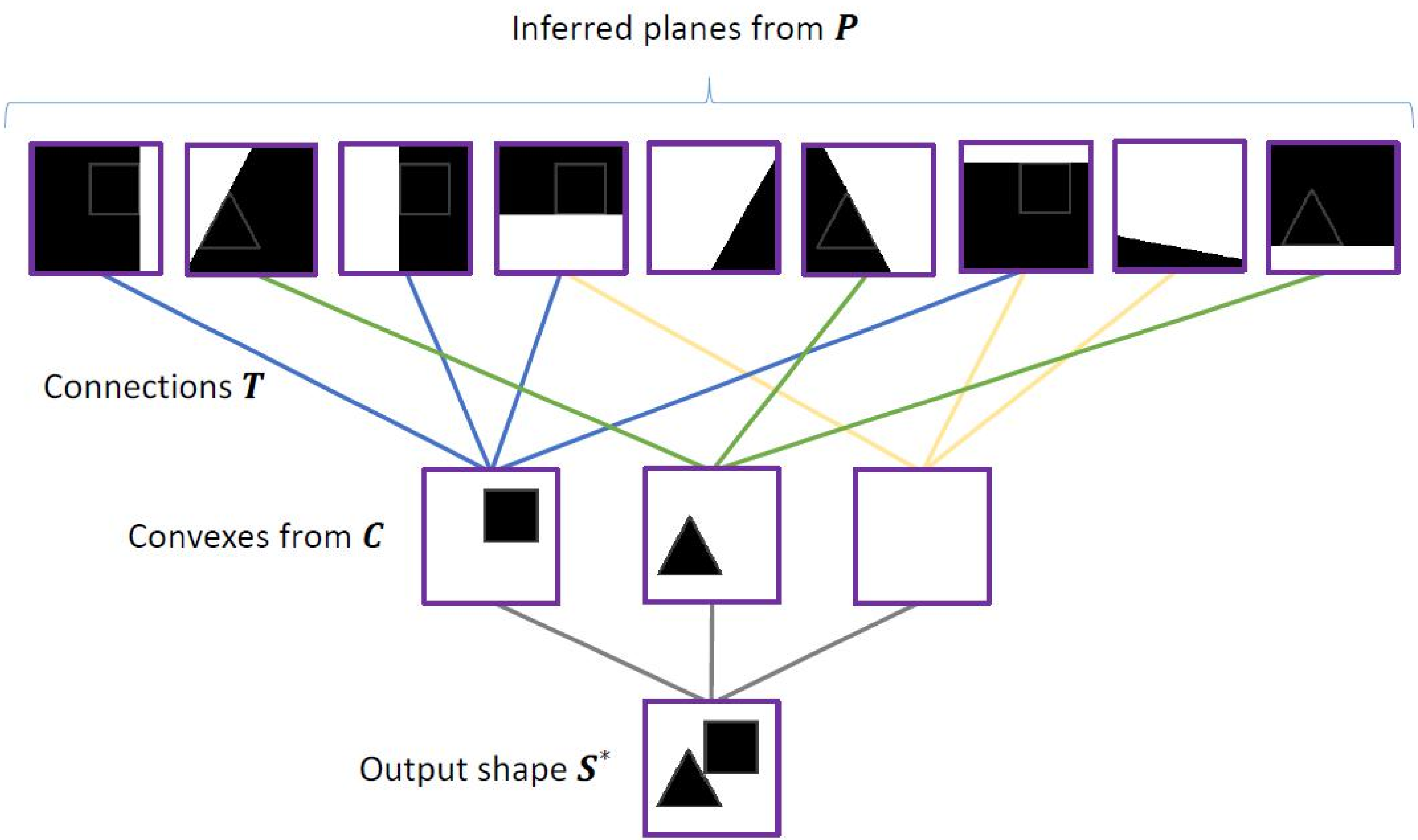
How to represent a cube?



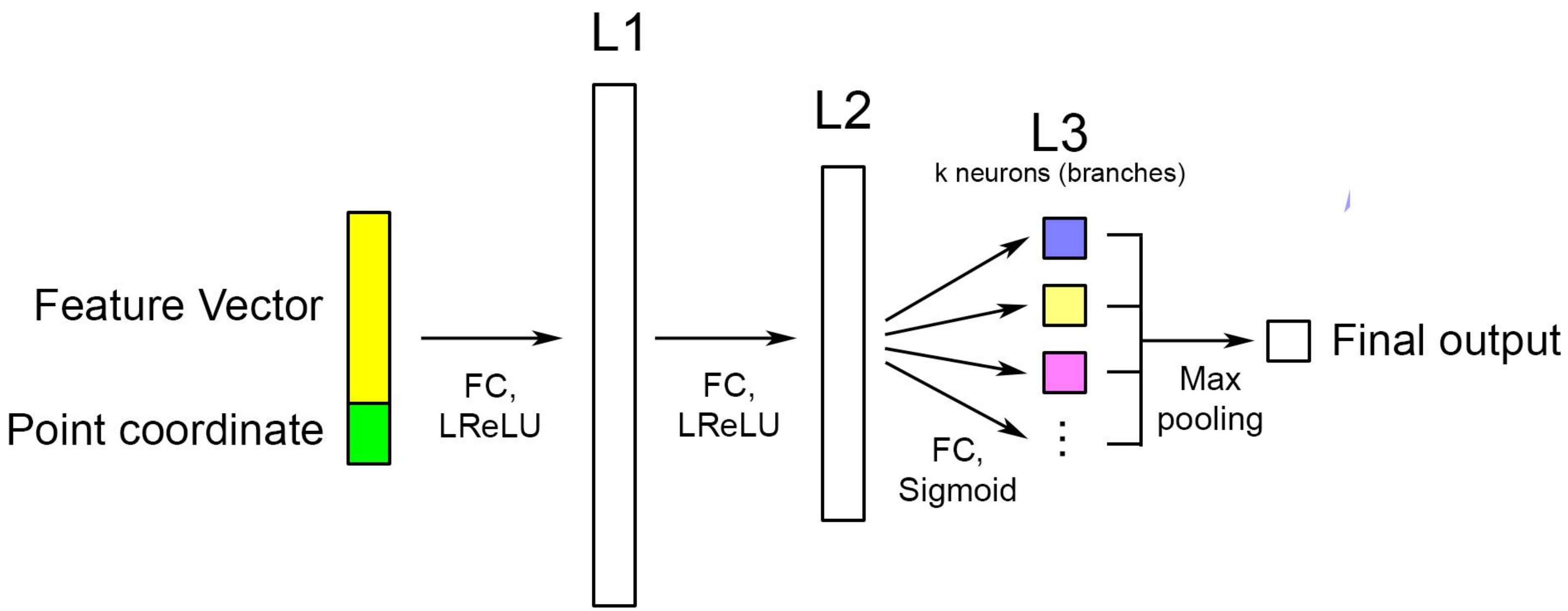
How to represent a cube?

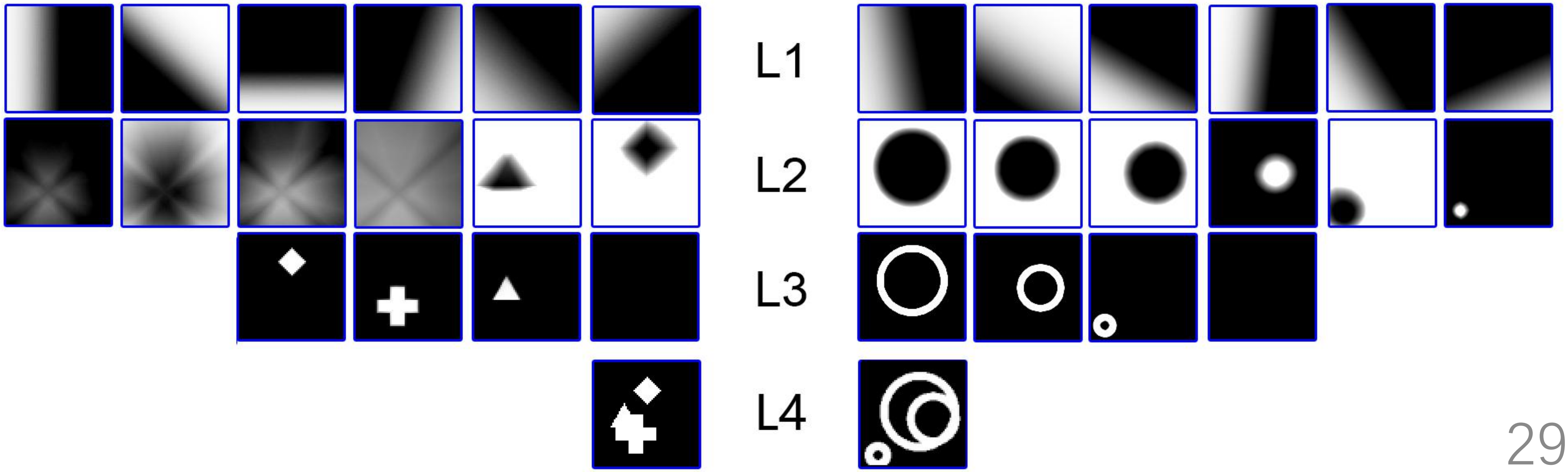


Represent shapes as Binary Space Partitioning tree



Interpretability



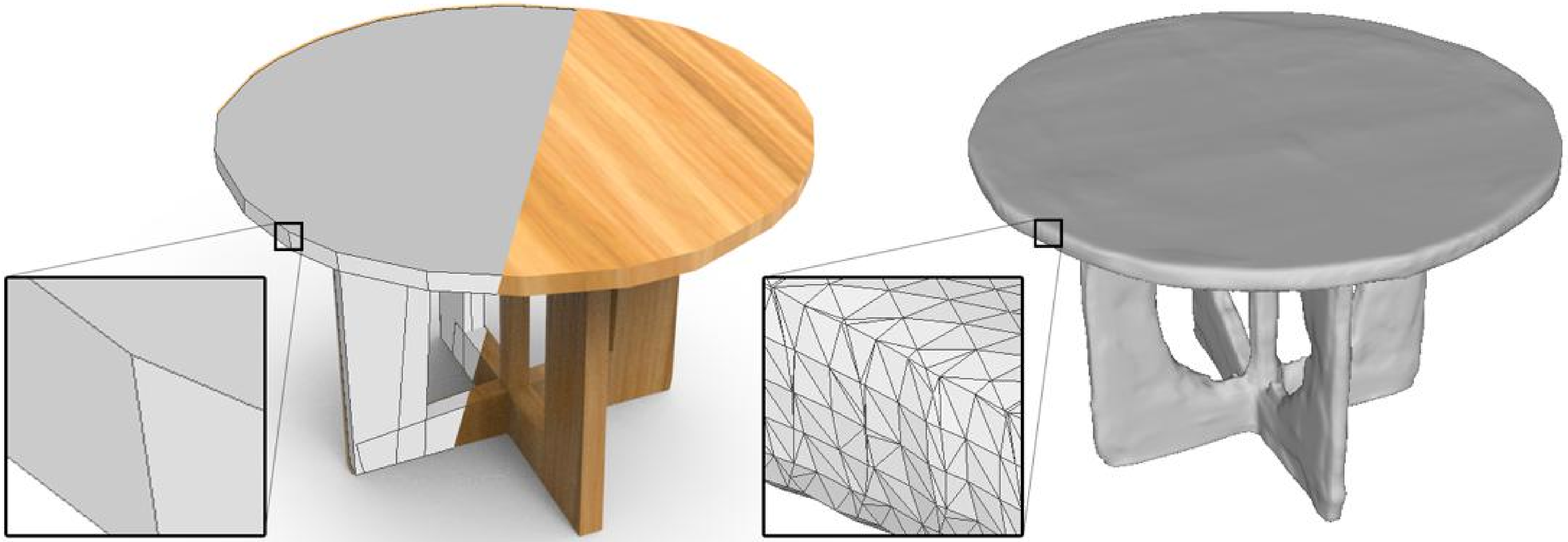


BSP-Net: Generating Compact Meshes via Binary Space Partitioning

Zhiqin Chen
Simon Fraser University

Andrea Tagliasacchi
Google Research

Hao Zhang
Simon Fraser University



(a) BSP-Net output
(392 vertices, 219 polygons or 600 triangles)

(b) IM-NET output
(sampled at 256^3, 91,542 vertices, 183,096 triangles)

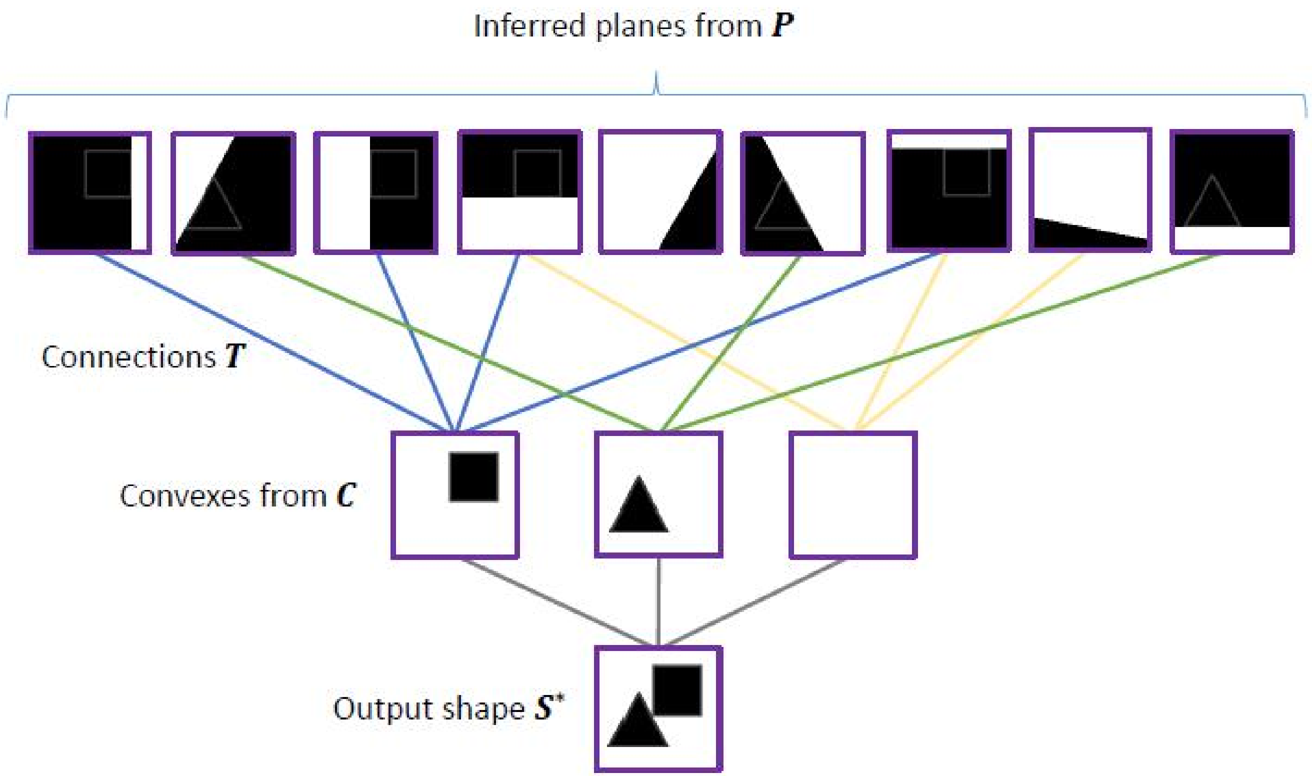


Figure 2: An illustration of "neural" BSP-tree.

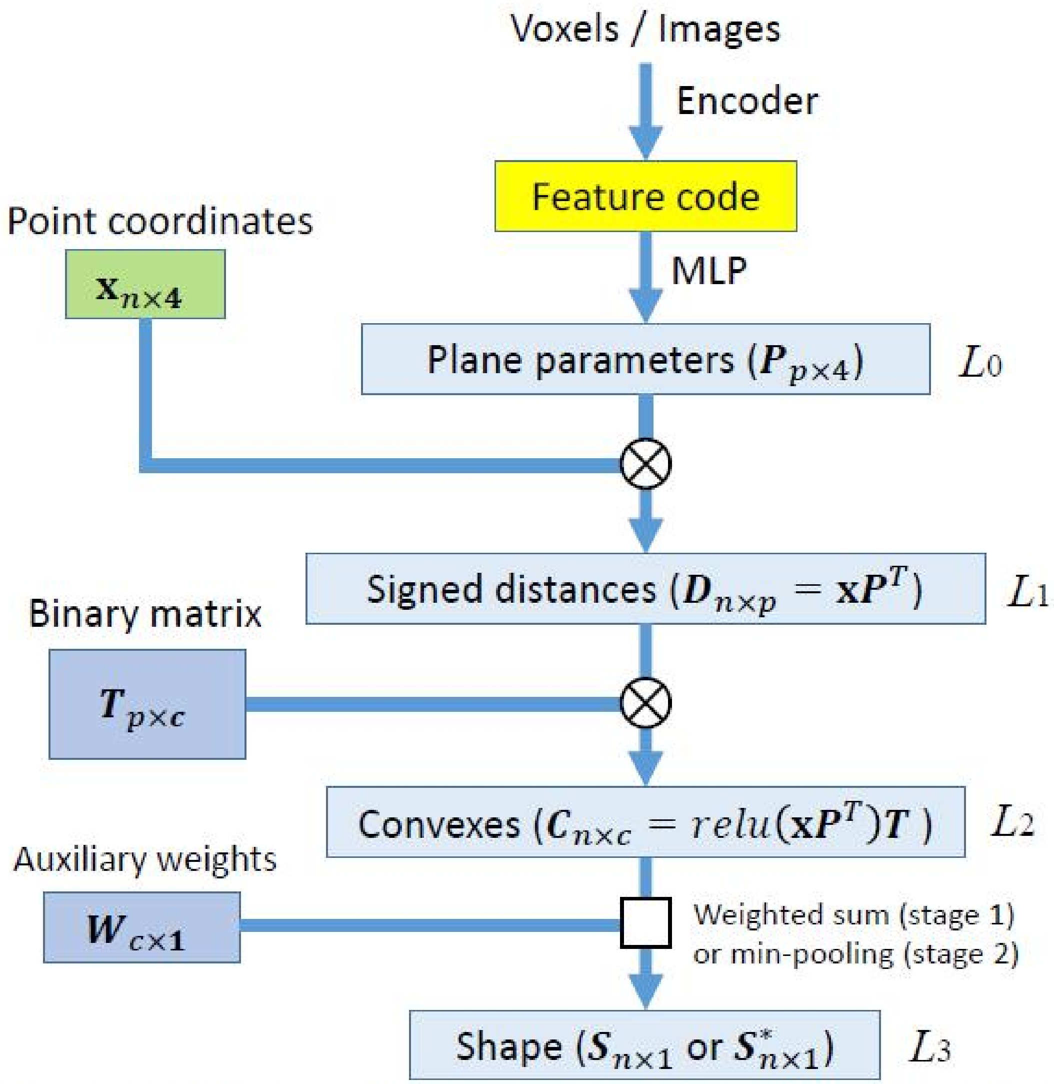
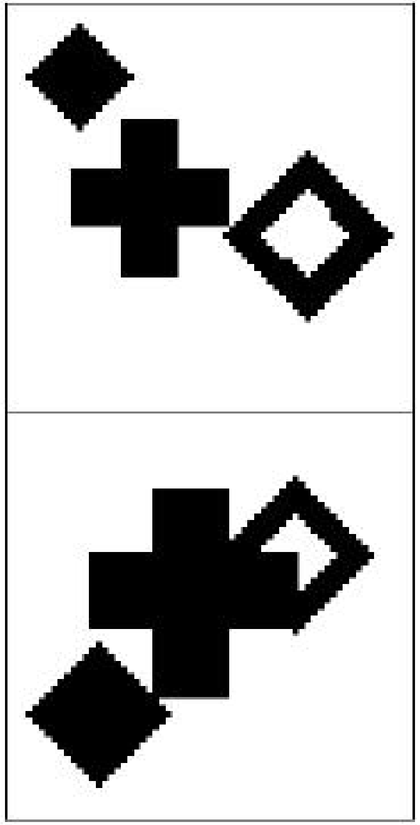
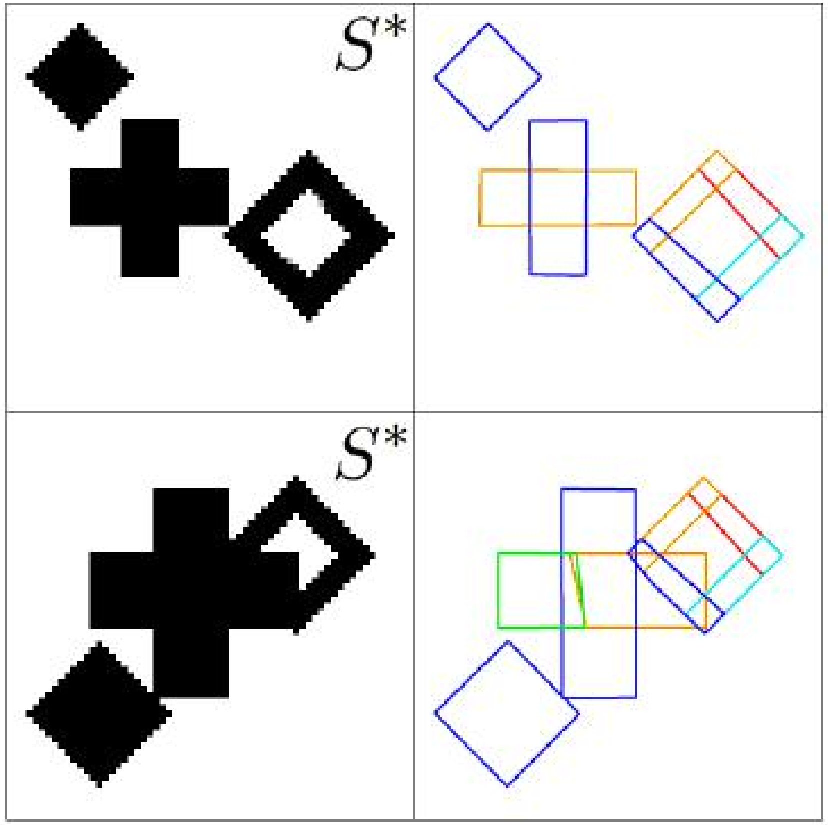


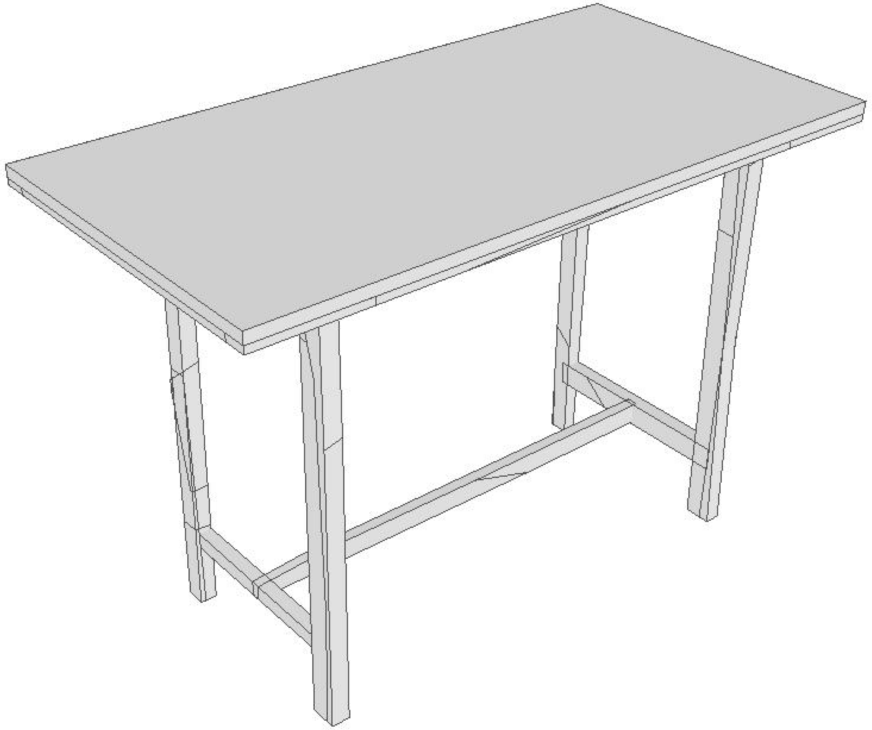
Figure 3: The network corresponding to Figure 2.

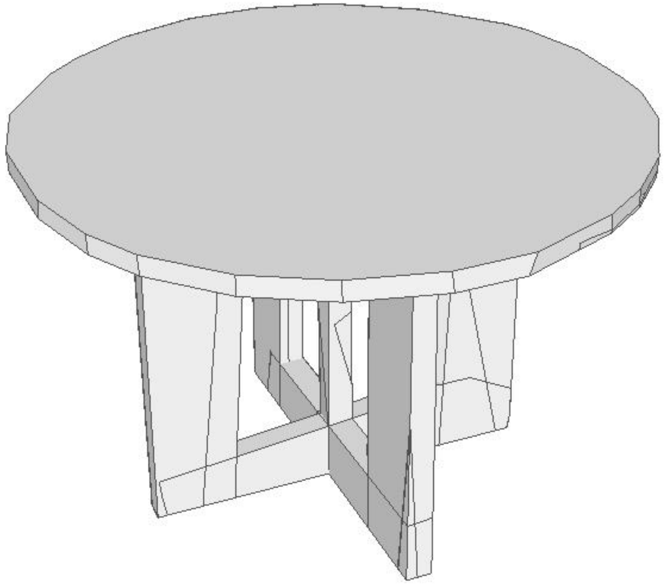


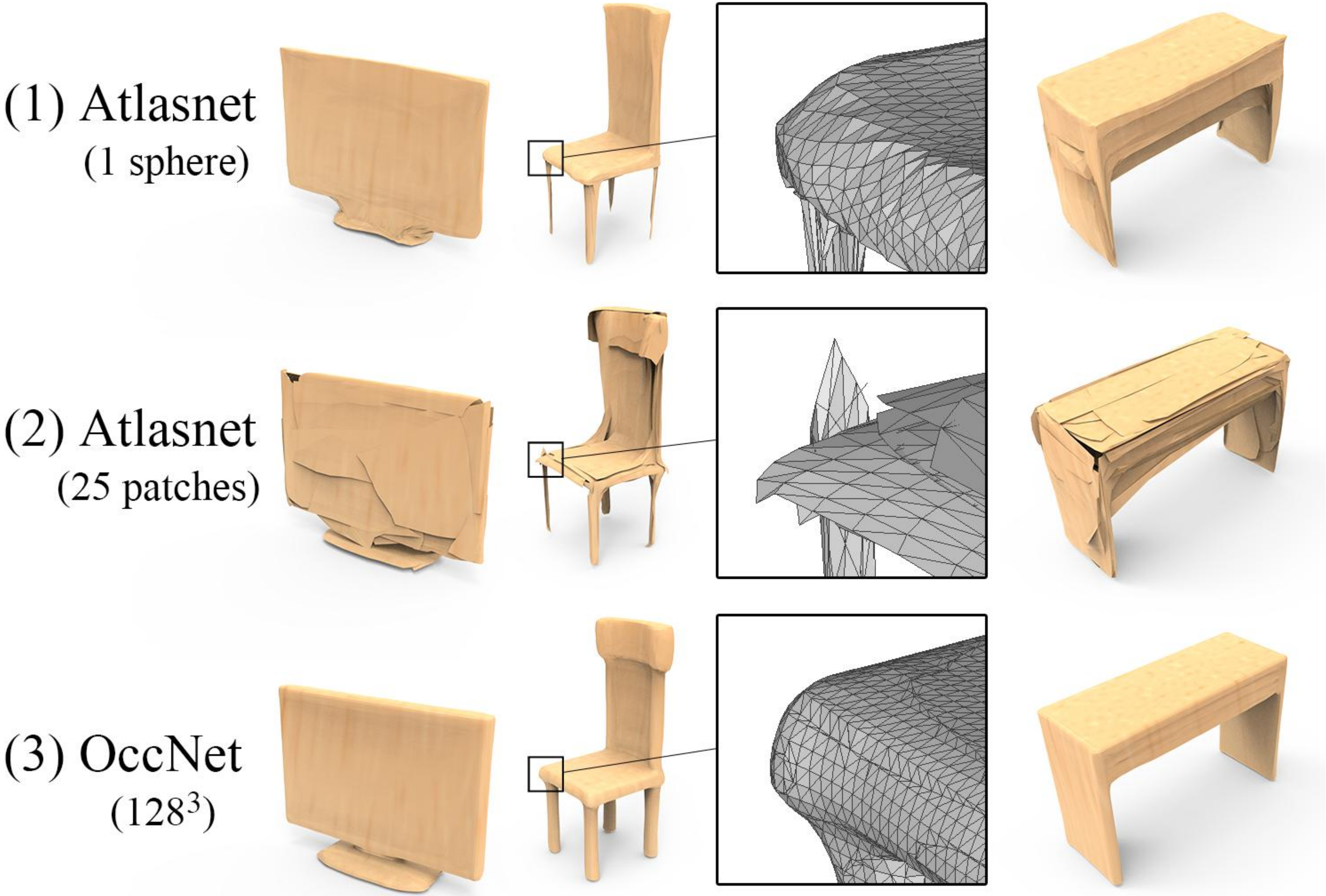
(a) Input

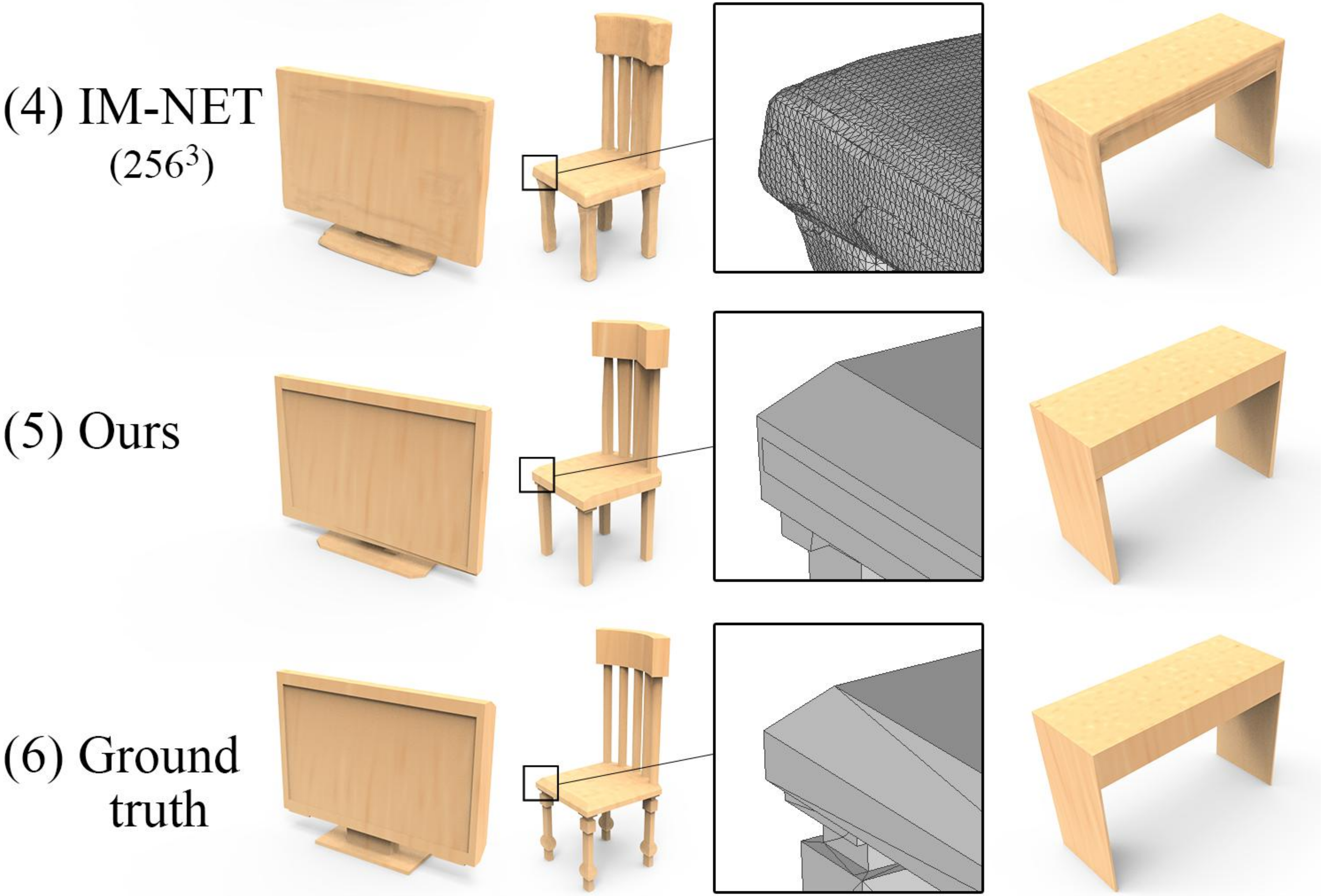


(d) Stage 2 - Discrete w/ $\mathcal{L}^*_{\text{overlap}}$

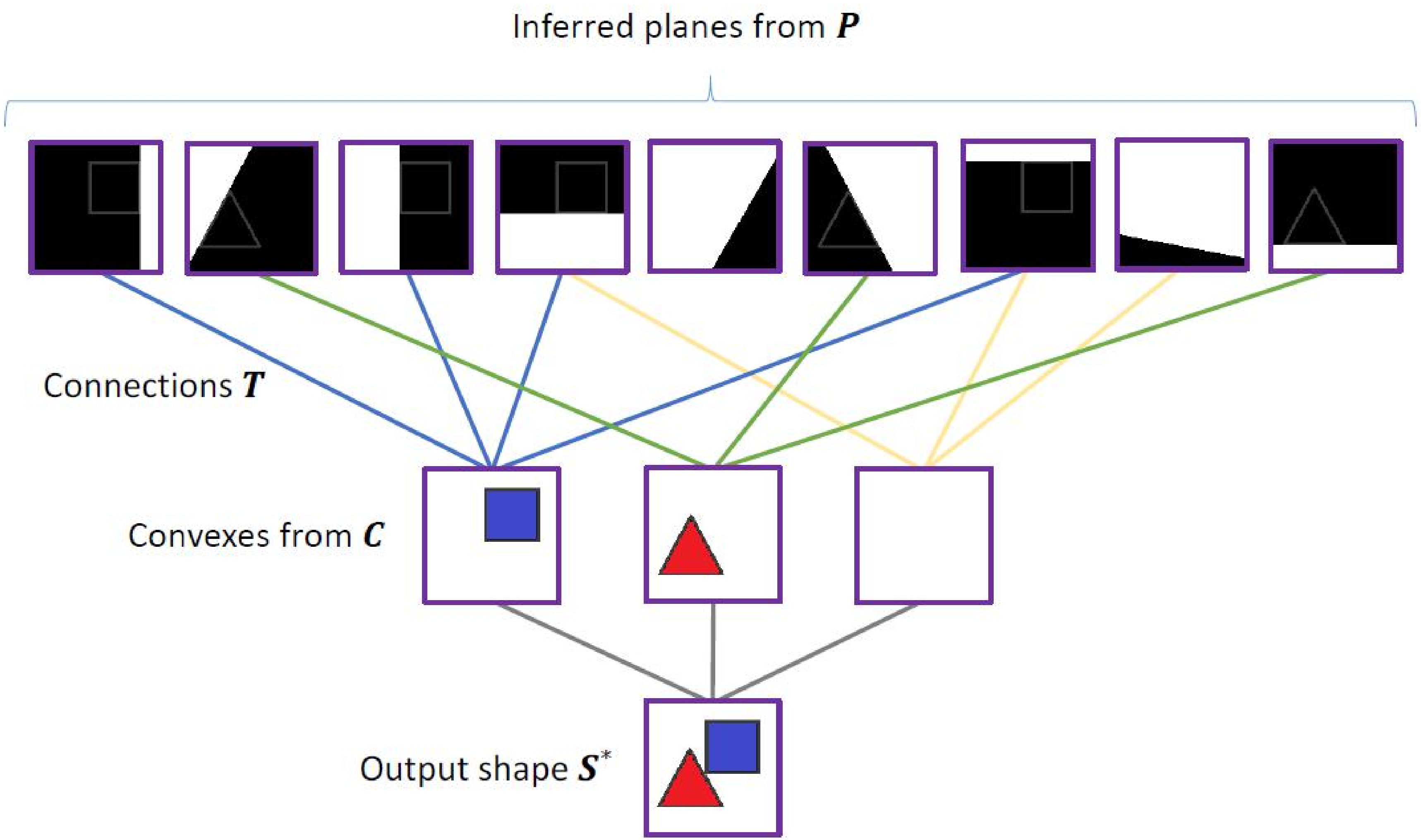


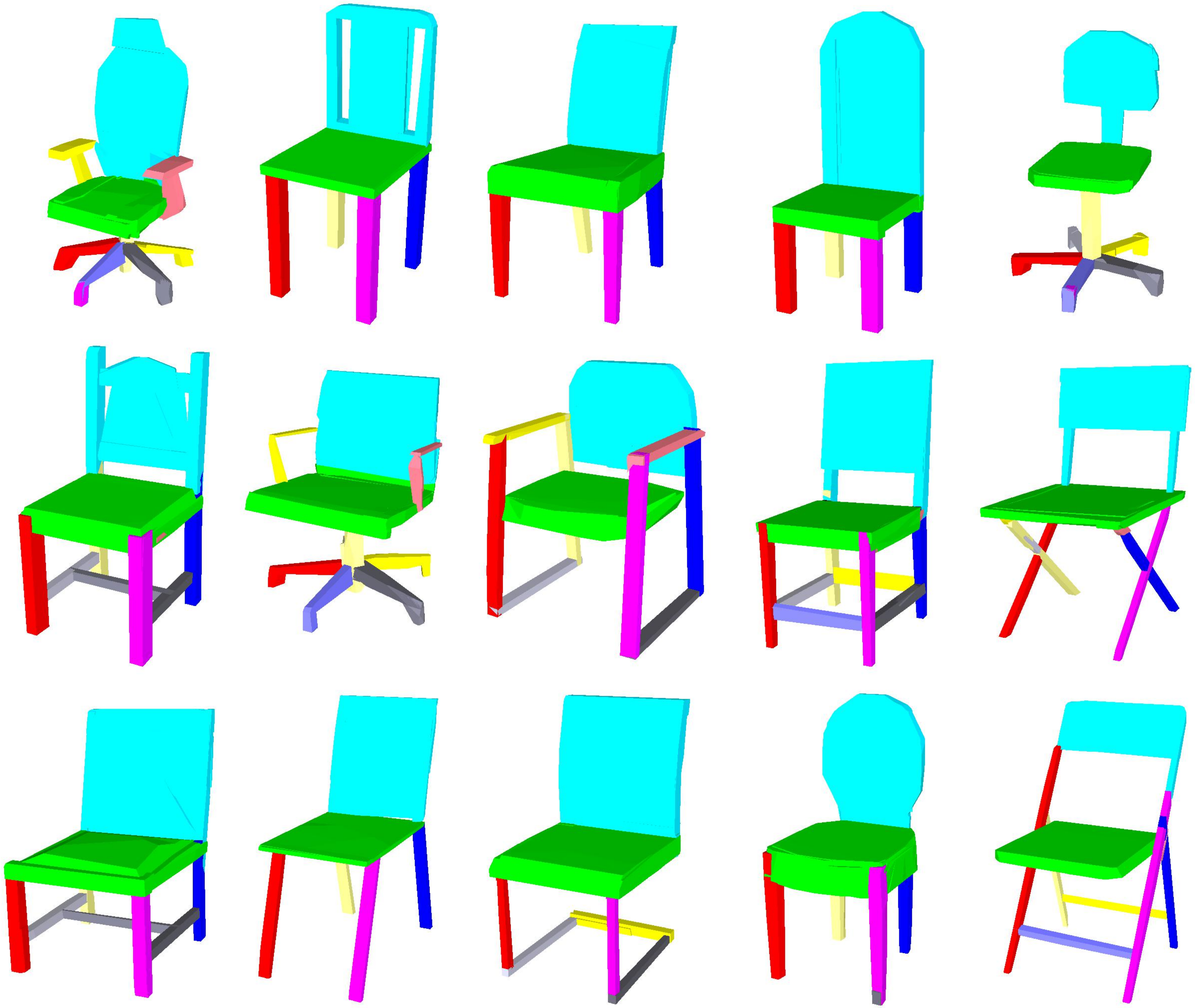




(1) Atlasnet
(1 sphere)
(2) Atlasnet
(25 patches)
(3) OccNet
(128^3)

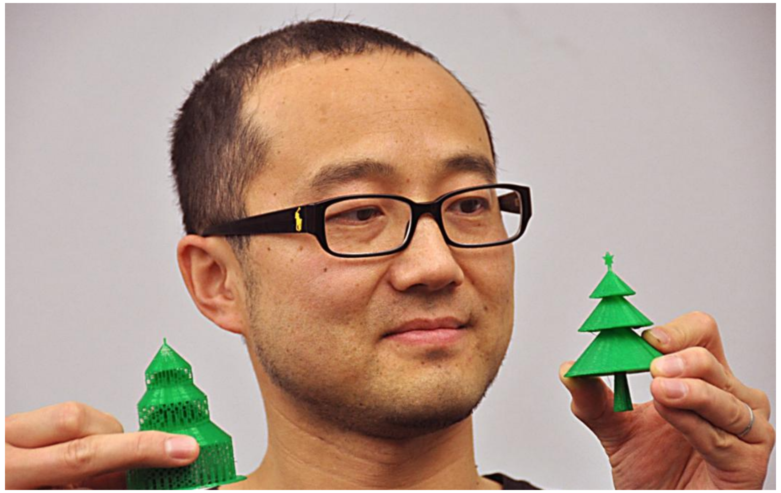

(4) IM-NET
(256³)
(5) Ours
(6) Ground truth

Represent shapes as Binary Space Partitioning tree





Acknowledgements

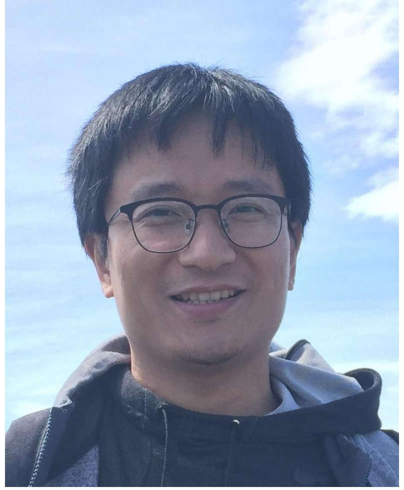


Hao (Richard) Zhang
Simon Fraser University
(my supervisor)

Siddhartha Chaudhuri
Adobe Research,
IIT Bombay

Matthew Fisher
Adobe Research

Andrea Tagliasacchi
Google Research

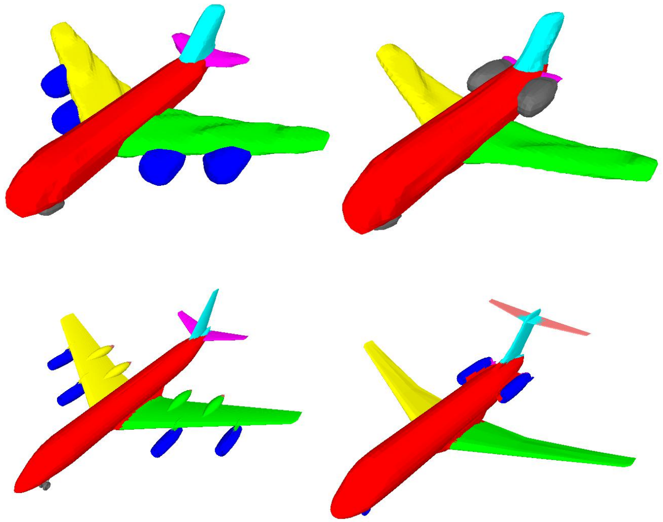


Kangxue Yin
Ph.D. student
Simon Fraser University

Thank you. Q&A

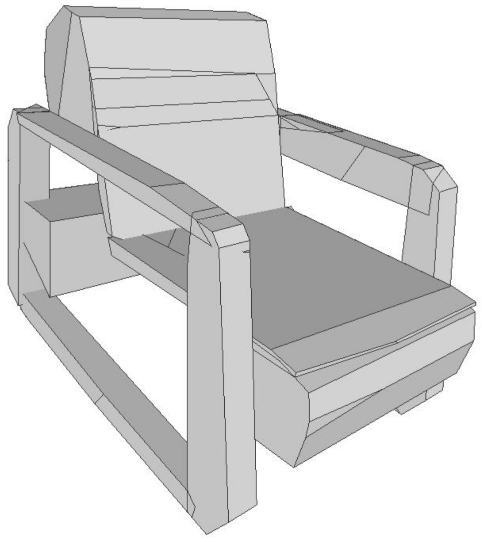
IM-NET

Learning Implicit Fields for Generative Shape Modeling (CVPR19)



BAE-NET

BAE-NET: Branched Autoencoder for Shape Co-Segmentation (ICCV19)



BSP-NET

BSP-Net: Generating Compact Meshes via Binary Space Partitioning (CVPR20)