

SECAD-Net: Self-Supervised CAD Reconstruction by Learning Sketch-Extrude Operations (Supplementary Materials)

Pu Li^{1,2} Jianwei Guo^{1,2*} Xiaopeng Zhang^{1,2} Dong-Ming Yan^{1,2}

¹MAIS, Institute of Automation, Chinese Academy of Sciences

²School of Artificial Intelligence, University of Chinese Academy of Sciences

1. Details of Evaluation Metrics

In the main paper, we used symmetric Chamfer Distance (*CD*), Normal Consistency (*NC*), Edge Chamfer Distance (*ECD*) as evaluation metrics. Specifically, *CD* measures the squared distances between nearest neighbors of two point clouds:

$$CD = \mathbb{E}_{\mathbf{x} \in \mathbf{P}} \min_{\mathbf{y} \in \mathbf{G}} \|\mathbf{x} - \mathbf{y}\|_2^2 + \mathbb{E}_{\mathbf{y} \in \mathbf{G}} \min_{\mathbf{x} \in \mathbf{P}} \|\mathbf{x} - \mathbf{y}\|_2^2, \quad (1)$$

where \mathbf{P} and \mathbf{G} are the point clouds uniformly sampled from the prediction mesh and ground truth mesh. *ECD* is calculated in the same way as *CD*, except that the targets are points that lie on the sharp edges of the shape, forming subsets \mathbf{P}^E and \mathbf{G}^E of uniformly sampled points:

$$ECD = \mathbb{E}_{\mathbf{x} \in \mathbf{P}^E} \min_{\mathbf{y} \in \mathbf{G}^E} \|\mathbf{x} - \mathbf{y}\|_2^2 + \mathbb{E}_{\mathbf{y} \in \mathbf{G}^E} \min_{\mathbf{x} \in \mathbf{P}^E} \|\mathbf{x} - \mathbf{y}\|_2^2. \quad (2)$$

The close-to-edge points for calculating *ECD* are obtained from 12,000 surface-sampled points. We adopt the settings of distance and sharpness threshold from [1]. Also utilizing the nearest neighbors, *NC* calculates the dot product of their normal vectors:

$$NC = (\mathbb{E}_{\mathbf{x} \in \mathbf{P}} [\mathbf{n}_{\mathbf{x}} \cdot \mathbf{n}_{\mathbf{y}_{\mathbf{x}}}] + \mathbb{E}_{\mathbf{y} \in \mathbf{G}} [\mathbf{n}_{\mathbf{y}} \cdot \mathbf{n}_{\mathbf{x}_{\mathbf{y}}}]) / 2, \quad (3)$$

where \mathbf{n} represents the normal vector, $\mathbf{x}_{\mathbf{y}}$ stands for the neighbor points of point cloud \mathbf{x} in point cloud \mathbf{y} , and vice versa. We use 8,192 surface-sampled points to compute *CD* and *NC*. For easy viewing, we scale *CD* by 1,000 and *ECD* by 100.

2. Discussion of primitives and assemblies of different methods

For primitive shapes, there is a trade-off between expressiveness and ease of editing. In addition to the methods we compared in the main paper, BSP-Net [1] and CAPRI-Net [6] are two other methods that use primitive shapes as

well as CSG operations for shape reconstruction. As shown in Table 1, the primitive shapes (convexes in the original papers) of BSP-Net and CAPRI-Net are intersections of linear partitions or quadratic partitions, which cannot be edited directly in 2D. UCSG-Net and CSG-Stump adopt fixed shapes, such as boxes and spheres, as primitives. The outputs of our method, point2Cyl and ExtrudeNet contain sketches, making the cylinder primitives editable in 2D. As shown in Table 1, the property of generating primitives that can be edited in 2D makes the methods in the last four rows within a fair comparison, so we report qualitative and quantitative comparisons of such methods in the main text.

Table 1. Comparison of different shape representations for reconstruction methods based on primitives.

Method	Primitives	2D editable
BSP-Net [1]	Intersection of linear partitions	✗
CAPRI-Net [6]	Intersection of quadratic partitions	✗
UCSG-Net [2]	Box, Sphere	✓
CSG-Stump [3]	Box, Sphere, Cylinder, Cone	✓
Point2Cyl [5]	Extrusion cylinder	✓
ExtrudeNet [4]	Extrusion cylinder	✓
Ours	Extrusion cylinder	✓

3. More comparison results

We show more reconstruction results of each method on the ABC and Fusion 360 datasets in Fig. 2 and Fig. 3, respectively. The modeling algorithm of the output meshes and the display order of the results are the same as in the main paper. To highlight the differences between SECAD-Net and ExtrudeNet [4], we further show the segmentation of the reconstruction results of the two methods in Fig. 1. Each color in the figure represents a primitive before the final union operation.

*Corresponding author: jianwei.guo@nlpr.ia.ac.cn

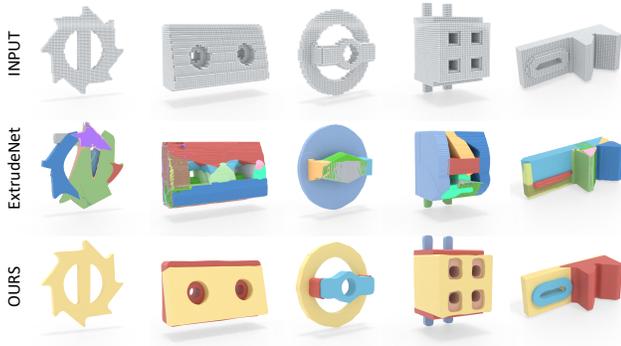


Figure 1. Visual comparison between ExtrudeNet [4] and ours. The different colors of the parts indicate different primitives.

4. Comparison results without fine-tuning

We compare the reconstruction performance of each method before fine-tuning on the ABC and Fusion datasets. Using the same evaluation metrics as the main paper, we list the quantitative results in Table 2. The results show that our method still achieves the best results in five out of six evaluations.

Table 2. Quantitative comparison between different methods without fine-tuning on ABC/Fusion 360 datasets.

Methods	CD↓	ECD↓	NC↑
UCSG-Net [2]	3.14/4.45	12.0/17.0	70.48/66.63
CSG-Stumpt [3]	3.34/5.22	3.14/6.88	70.70/66.28
ExtrudeNet [4]	3.07/5.61	2.99/6.92	71.35/68.15
Ours	2.94/4.20	3.00/5.53	71.98/68.68

5. More ablation comparisons of sketch representations

In order to demonstrate the advantages of our sketch head network (SK-head) over other classical shape representations, we further show the profiles predicted by different sketch representations in the ablation study in Fig. 6. It can be seen that our SK-head can predict smoother and more complete profiles compared to BSP and box primitives.

6. More visual results of sketch interpolations

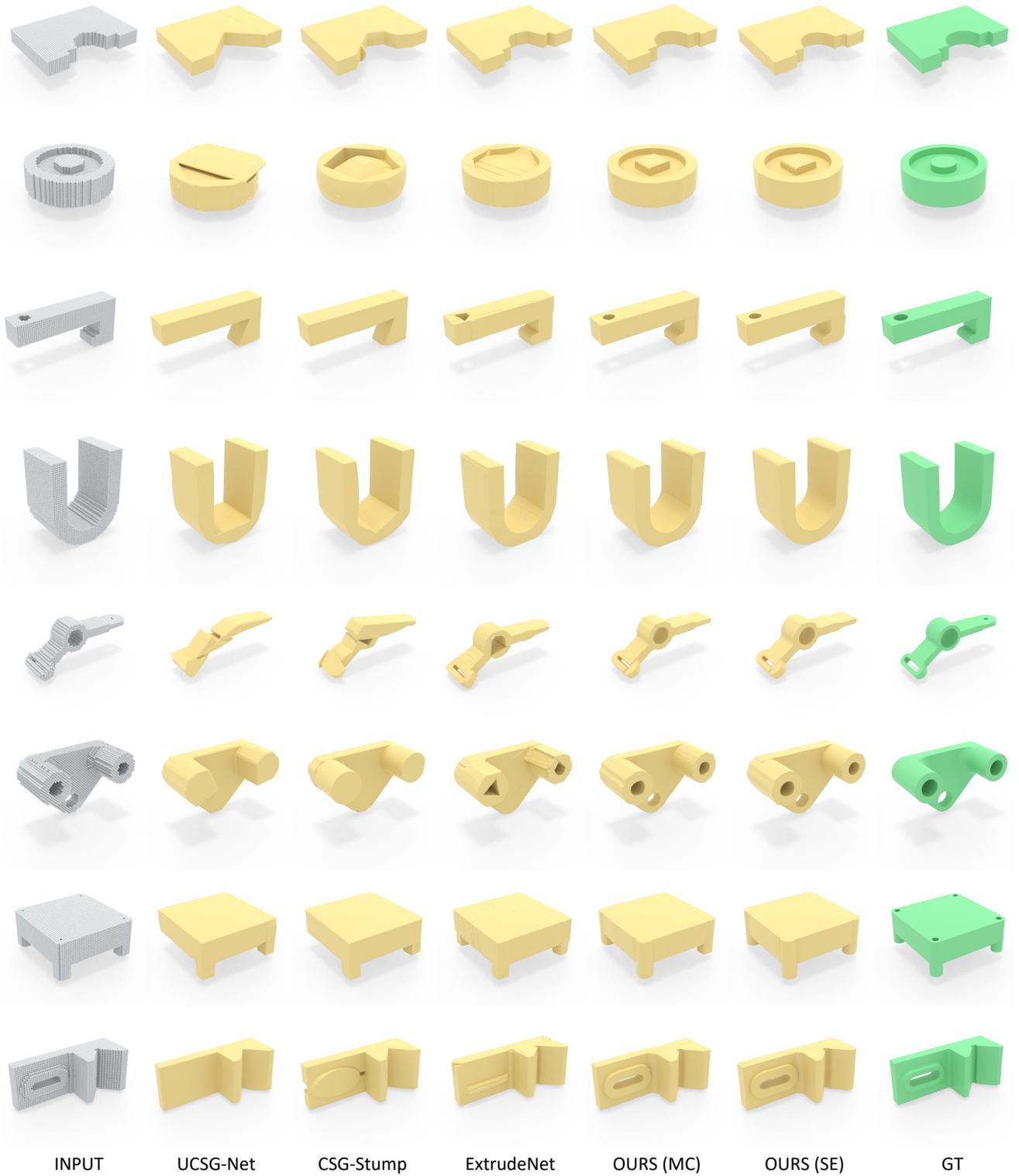
To further explore the interpretability of the proposed network, we show the 3D shapes and profiles corresponding to the linearly interpolated embedding codes in Fig. 4 and Fig. 5. The new sketches obtained by interpolation can be directly sent to CAD software for editing and re-creation.

7. CSG assembly using cylinder primitives

In our final CAD result, the smallest unit is the cylinder primitives, which can be assembled into the final shape using CSG operations. The assembly process forms a CSG tree-like structure, and we show some assembly visualization results in Fig. 7.

References

- [1] Zhiqin Chen, Andrea Tagliasacchi, and Hao Zhang. BSP-Net: Generating compact meshes via binary space partitioning. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 45–54, 2020. 1
- [2] Kacper Kania, Maciej Zieba, and Tomasz Kajdanowicz. UCSG-NET-Unsupervised discovering of constructive solid geometry tree. In *Advances in Neural Information Processing Systems*, pages 8776–8786, 2020. 1, 2, 3
- [3] Daxuan Ren, Jianmin Zheng, Jianfei Cai, Jiatong Li, Haiyong Jiang, Zhongang Cai, Junzhe Zhang, Liang Pan, Mingyuan Zhang, Haiyu Zhao, et al. CSG-stump: A learning friendly CSG-like representation for interpretable shape parsing. In *IEEE International Conference on Computer Vision (ICCV)*, pages 12478–12487, 2021. 1, 2, 3
- [4] Daxuan Ren, Jianmin Zheng, Jianfei Cai, Jiatong Li, and Junzhe Zhang. Extrudenet: Unsupervised inverse sketch-and-extrude for shape parsing. In *Computer Vision—ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*, pages 482–498, 2022. 1, 2, 3
- [5] Mikaela Angelina Uy, Yen-Yu Chang, Minhyuk Sung, Purvi Goel, Joseph G Lambourne, Tolga Birdal, and Leonidas J Guibas. Point2Cyl: Reverse engineering 3D objects from point clouds to extrusion cylinders. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 11850–11860, 2022. 1
- [6] Fenggen Yu, Zhiqin Chen, Manyi Li, Aditya Sanghi, Hooman Shayani, Ali Mahdavi-Amiri, and Hao Zhang. CAPRI-Net: Learning compact cad shapes with adaptive primitive assembly. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 11768–11778, 2022. 1



INPUT UCSG-Net CSG-Stump ExtrudeNet OURS (MC) OURS (SE) GT

Figure 2. Visual comparison of our approach with UCSG-Net [2], CSG-Stump [3], and ExtrudeNet [4] on the ABC dataset. Our reconstruction results are generated using both marching cubes (MC) and our proposed sketch-extrude operations (SE).



Figure 3. Visual comparison between reconstruction results on Fusion 360 dataset.

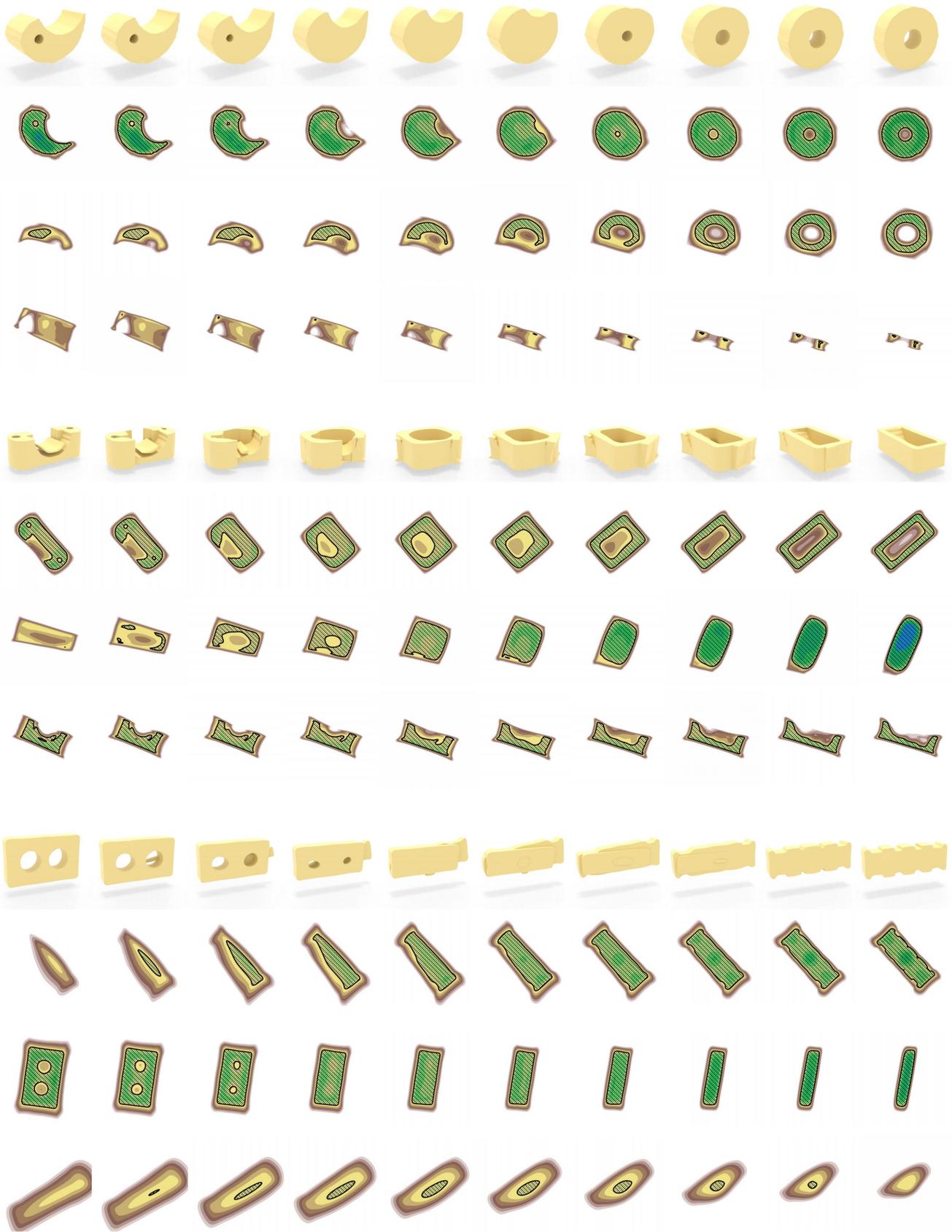


Figure 4. The leftmost and rightmost columns are the prediction results of SECAD-Net on the ABC dataset. The middle results are obtained by interpolating the latent embeddings. For each case, the top row shows the 3D shape, and each remaining row represents the sketches learned by one sketch head network.

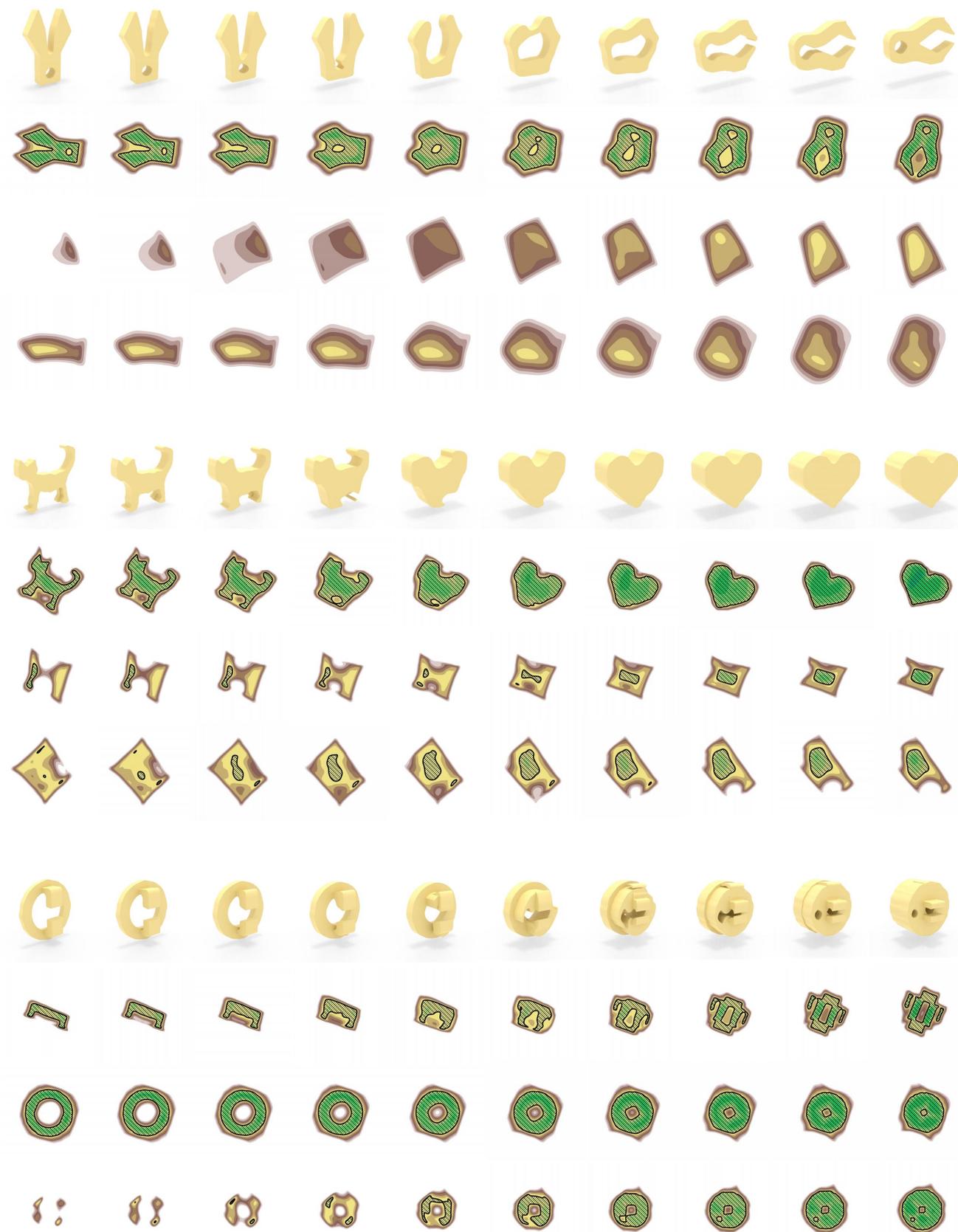


Figure 5. The leftmost and rightmost columns are the prediction results of SECAD-Net on the ABC dataset. The middle results are obtained by interpolating the latent embeddings. For each case, the top row shows the 3D shape, and each remaining row represents the sketches learned by one sketch head network.

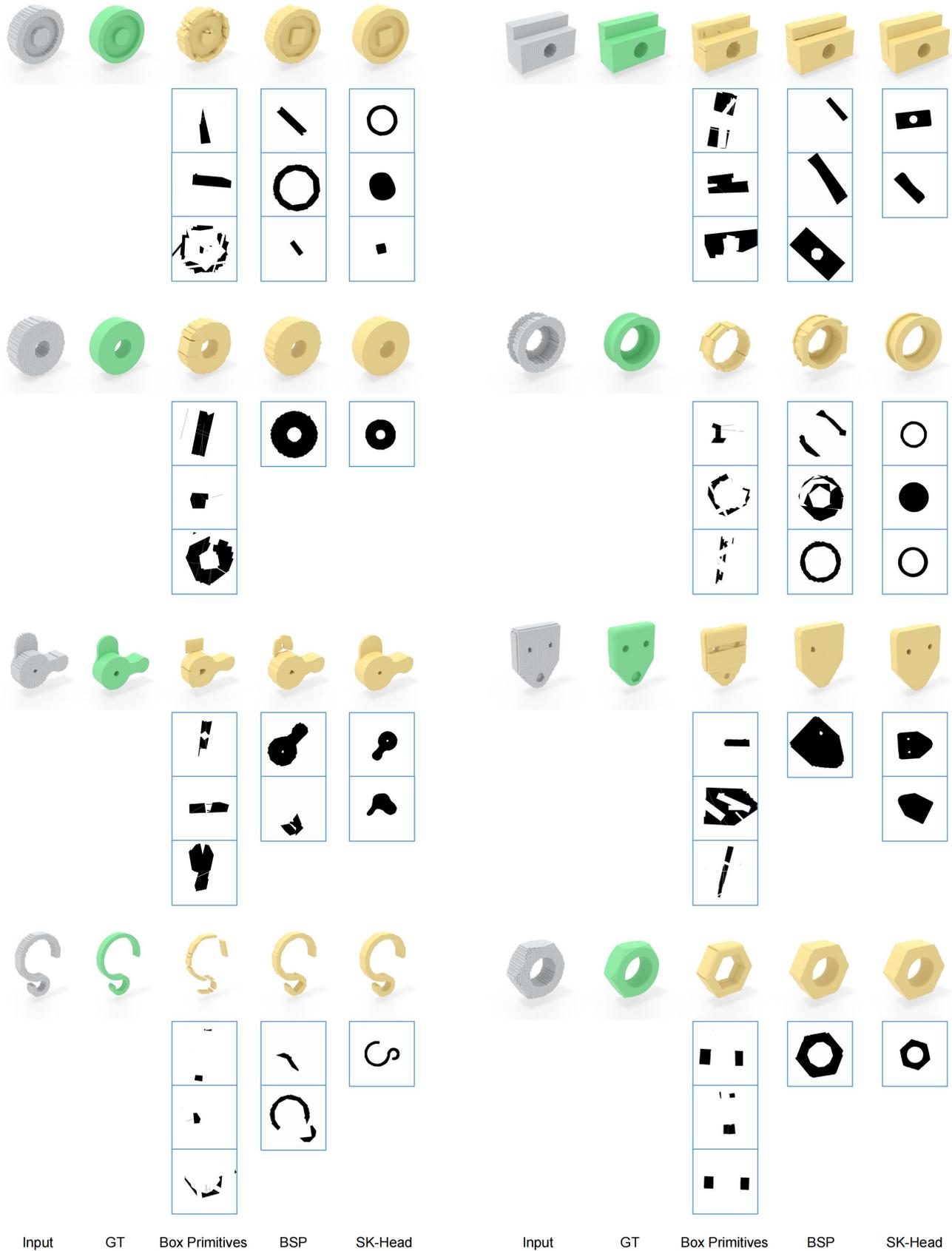


Figure 6. Visual comparison results for ablation study on sketch representation. Each case contains 3D shapes modeled with different sketch representations and the corresponding binary profile images.

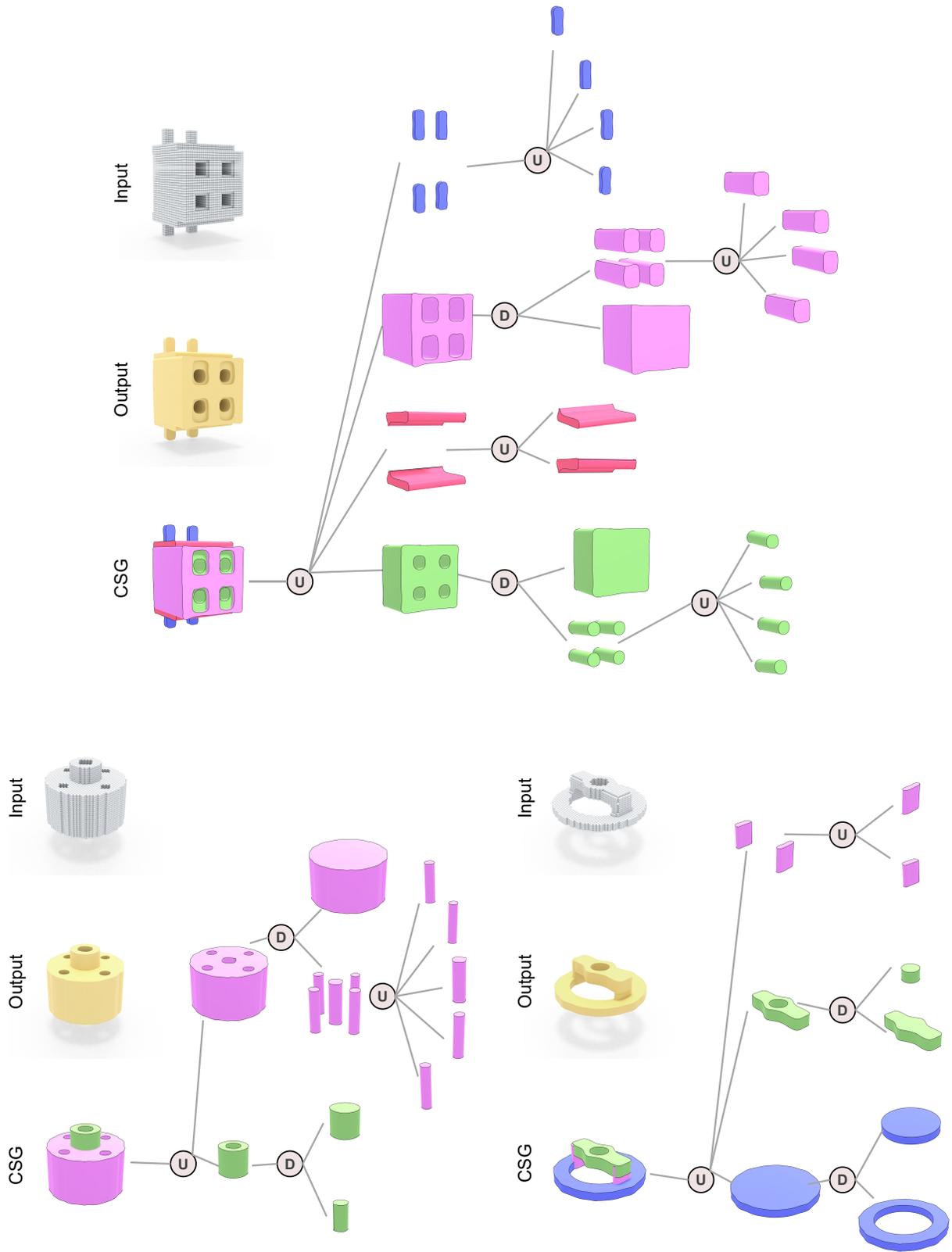


Figure 7. Visualization of assembling cylinder primitives into complete shapes with Boolean operations. Therefore, the final output CAD models are represented as a CSG tree-like structure.