Supplementary: 3DN: 3D Deformation Network

Weiyue Wang[1] Duygu Ceylan[2] Radomir Mech[2] Ulrich Neumann[1]
[1]University of Southern California
Los Angeles, California
{weiyuewa,uneumann}@usc.edu
[2]Adobe
San Jose, California
{ceylan,rmech}@adobe.com

1. Implementation Details

We use an ADAM optimizer with initial learning rate 0.0005, momentum 0.9 and batch size 4. Our network is implemented with Tensorflow and trained on an Nvidia GTX1080 Ti GPU. When the input is a point cloud, we have used a point cloud of size 2048×3. No Batch Normalization layer is used. The weights for the losses are $\omega_{L_1} = 1000, \omega_{L_2} = 1, \omega_{L_3} = 10000, \omega_{L_4} = 1, \omega_{L_5} = 1000, \omega_{L_6} = 0.01, \omega_{L_7} = 1000$.

The mesh sampling operator is implemented with CuDA acceleration and Tensorflow. Since number of vertices are varying for different meshes and we need to train the network with batch size > 1, we set a maximum number of vertices and triangles to make the operation trainable with batch size > 1, and we also input the number of vertices and triangles for each sample. In the CuDA implementation, we sample the same number of points for meshes with different number of vertices. The inputs to the mesh sampling operator are $V \in \mathbb{R}^{B \times N_{Vmax} \times 3}$, $T \in \mathbb{Z}^{B \times N_{Tmax} \times 3}$, $N_V \leq N_{Vmax} \in \mathbb{Z}^{B \times 1}$, $N_T \leq N_{Tmax} \in \mathbb{Z}^{B \times 1}$, $w_{V1} \in (0,1)^{B \times N_{Tmax} \times 3}$, $w_{V2} \in (0,1)^{B \times N_{Tmax} \times 3}$, $w_{V3} \in (0,1)^{B \times N_{Tmax} \times 3}$, where B is batch size, N_{Vmax} is the max number of mesh vertices,, N_{Tmax} is the max number of mesh faces,, N_V and N_T indicate the number of vertices and triangles for each sample in the mini batch, w_{V1}, w_{V2} and w_{V3} are the random barycentric coordinate weights.

2. Network Architecture

2.1. Offset Decoder

Figure 1 illustrates the network architecture of our offset decoder.

2.2. PointNet Autoencoder for Template Selection

Figure 2 shows the network architecture of the PointNet autoencoder for template selection. “Embedding” denotes the feature vector we use to query the template source mesh.

2.3. Mid-layer fusion

Figure 3 illustrates the network architecture used in mid-layer fusion experiment.

3. Failure Cases

Figure 4 shows our failure cases when deformation is performed across different categories.

4. Qualitative Results

4.1. Point cloud as target

Figure 5, 6, 7, and 8 illustrates more qualitative results of 3DN with point cloud as target.

4.2. 2D image as target

Figure 9, 10, 11, and 12 illustrates more qualitative results of 3DN with 2D image as target.

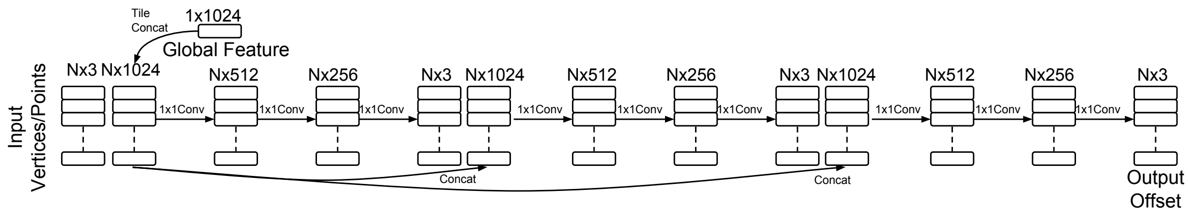

Figure 1: Offset decoder architecture. Each $1 \times 1Conv$ is followed by a ReLU layer except for the last one.

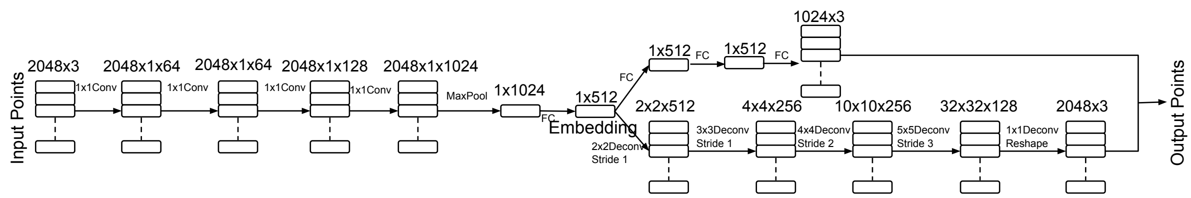

Figure 2: PointNet autoencoder architecture. Each $1 \times 1Conv$ is followed by a ReLU layer except for the last one. "Embedding" is the feature vector used to query template source mesh.

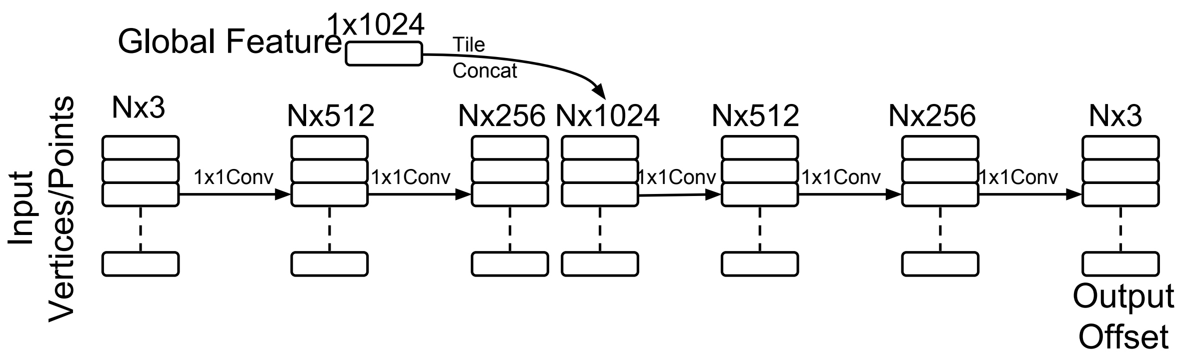

Figure 3: Offset decoder architecture for mid-layer fusion experiment. Each $1 \times 1Conv$ is followed by a ReLU layer except for the last one.

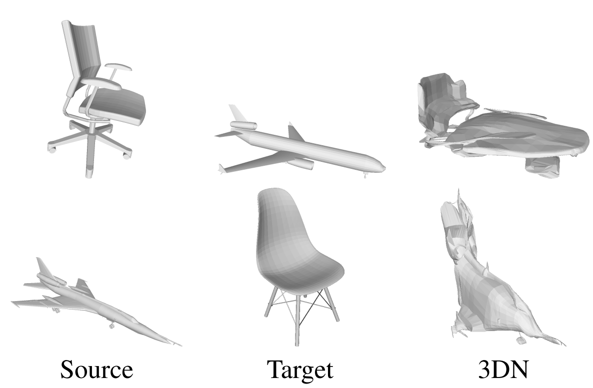

Figure 4: Failure cases.

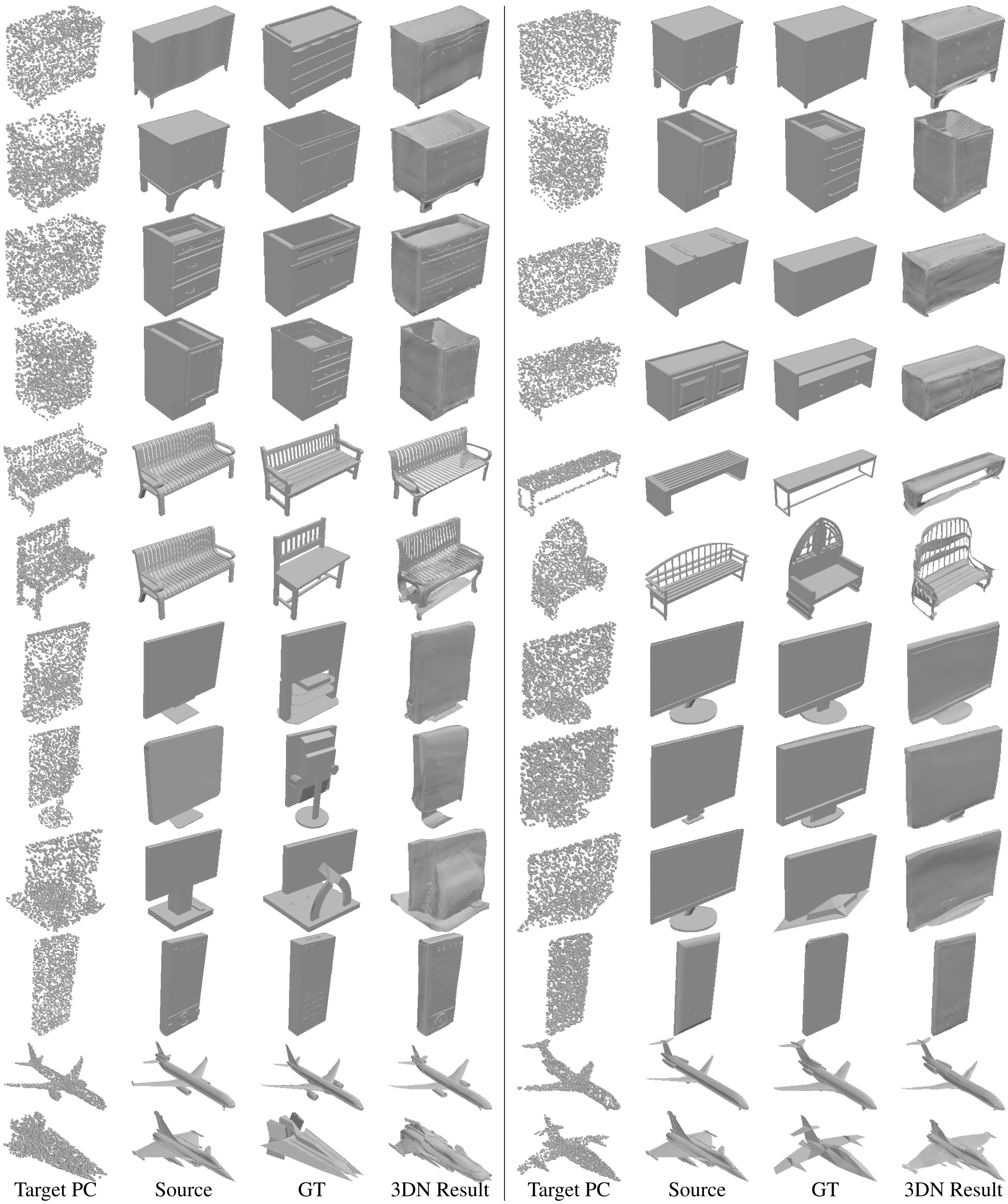

Figure 5: Qualitative results on ShapeNet point cloud. “PC” denotes point cloud. And “GT” denotes ground truth mesh.

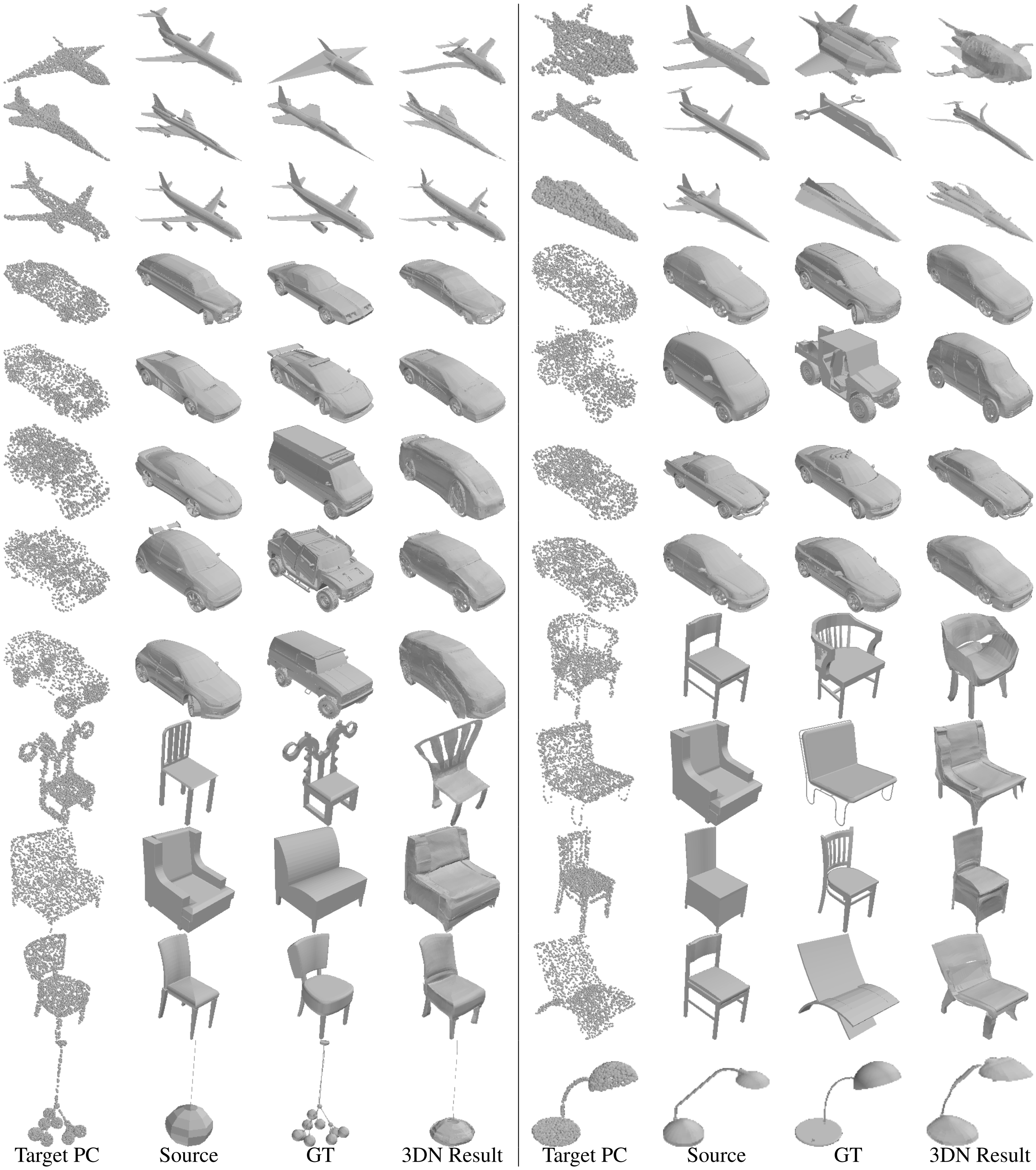

Figure 6: Qualitative results on ShapeNet point cloud. “PC” denotes point cloud. And “GT” denotes ground truth mesh.

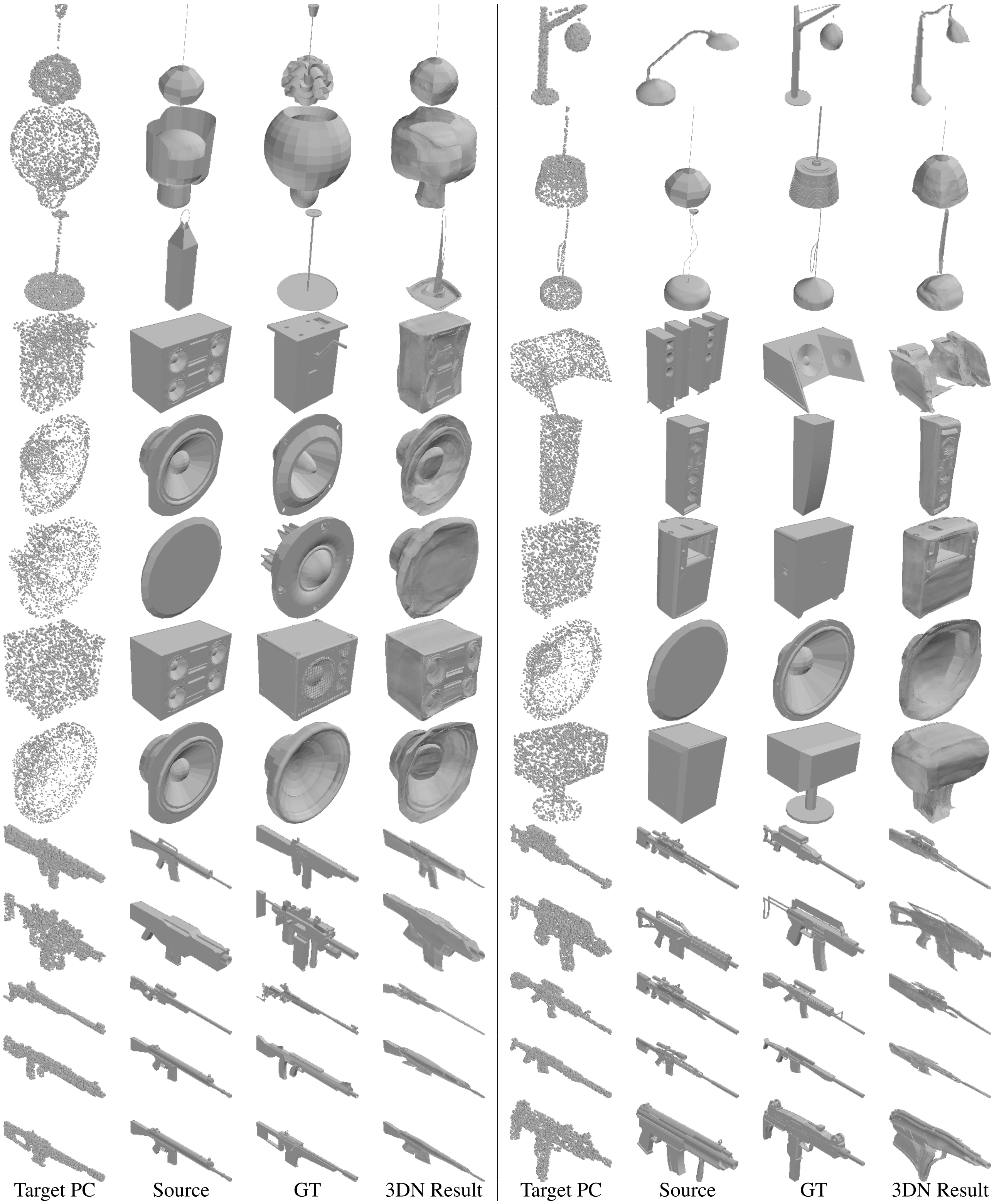

Figure 7: Qualitative results on ShapeNet point cloud. “PC” denotes point cloud. And “GT” denotes ground truth mesh.

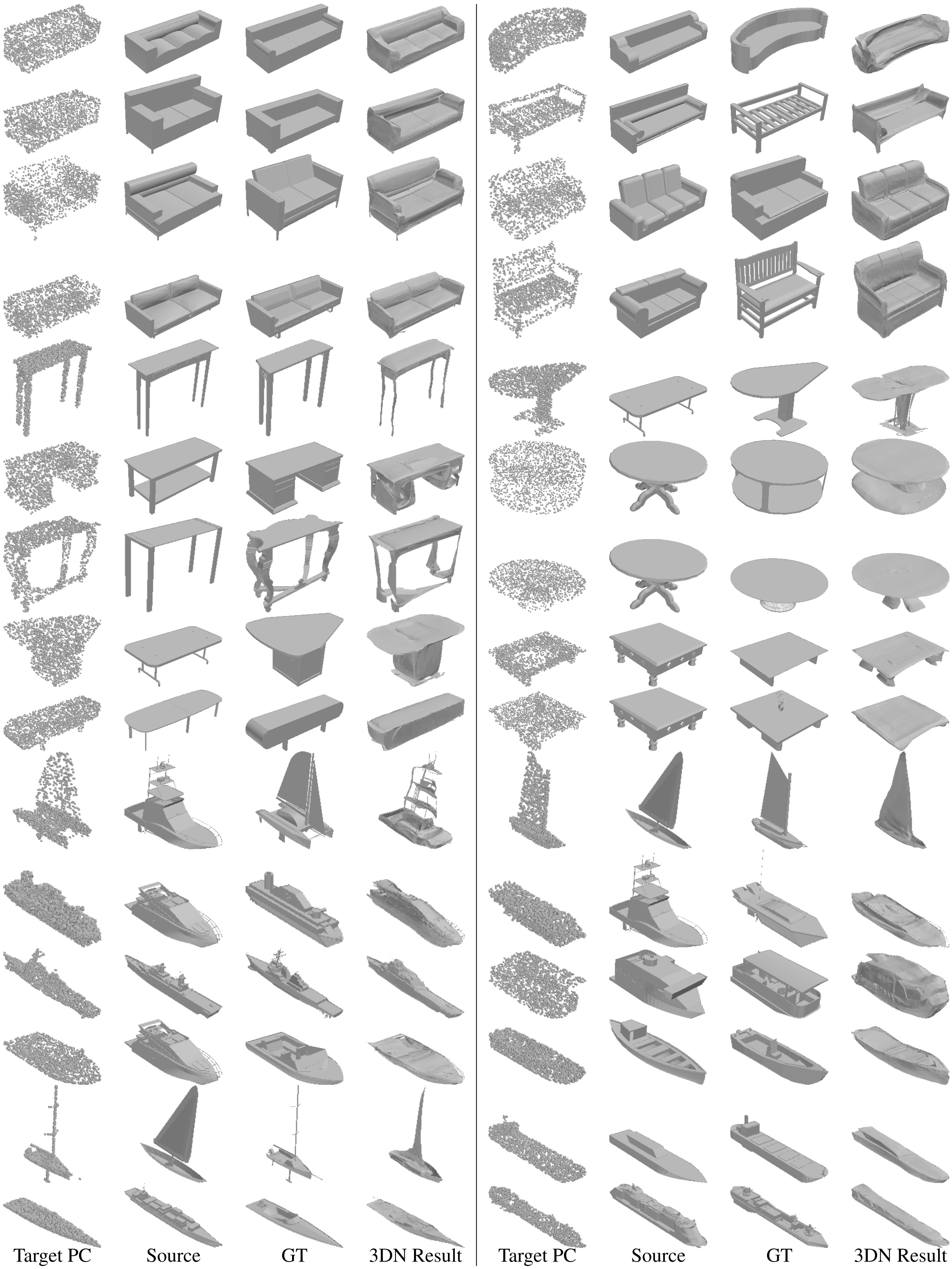

Figure 8: Qualitative results on ShapeNet point cloud. “PC” denotes point cloud. And “GT” denotes ground truth mesh.

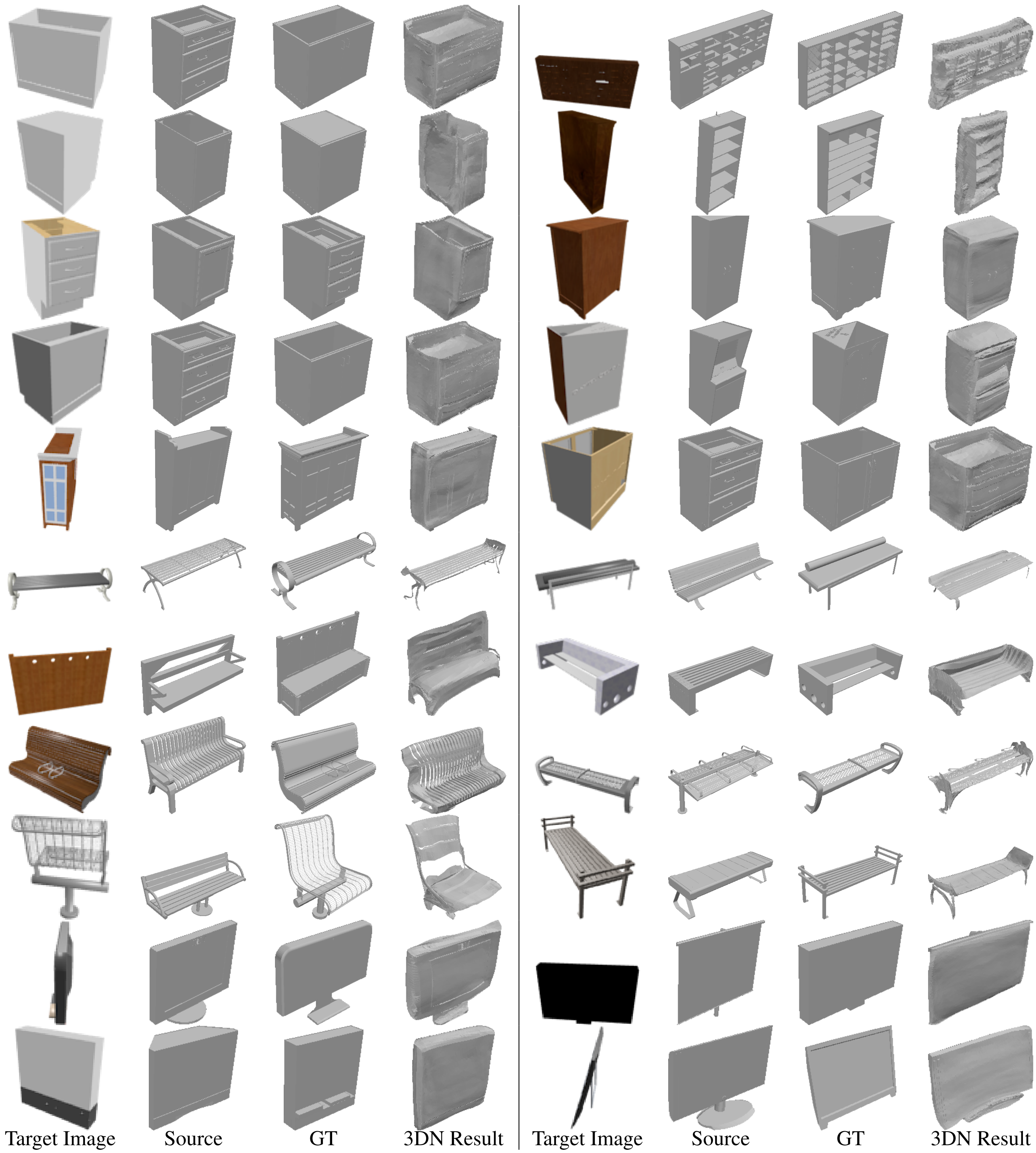

Figure 9: Qualitative results on ShapeNet 2D images. “GT” denotes ground truth mesh.

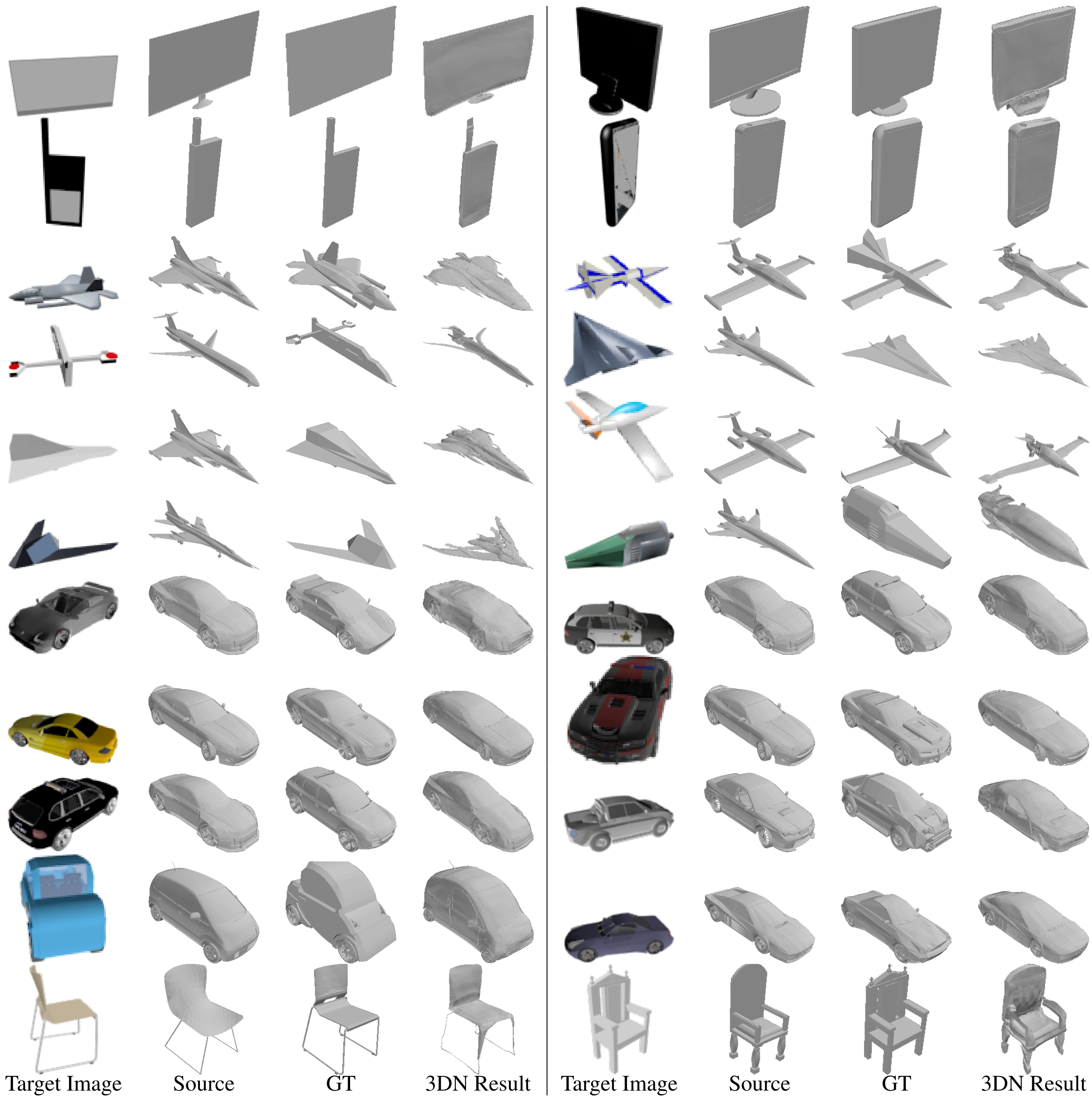

Figure 10: Qualitative results on ShapeNet 2D images. "GT" denotes ground truth mesh.

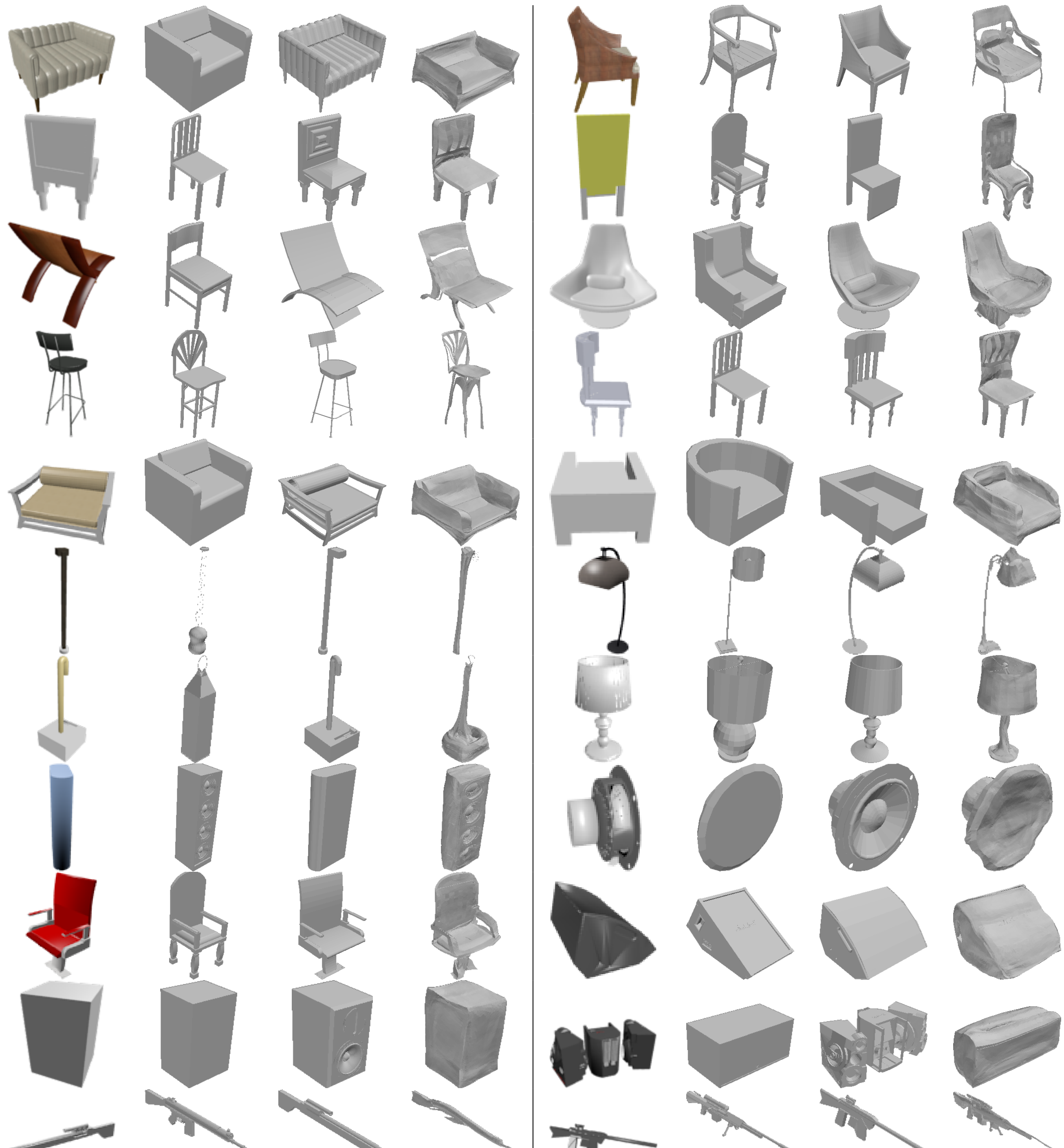

Figure 11: Qualitative results on ShapeNet 2D images. "GT" denotes ground truth mesh.

Figure 12: Qualitative results on ShapeNet 2D images. “GT” denotes ground truth mesh.