CS482/682 Final Project Report Group 3

Self-Supervised 3D Shape Reconstruction from Single-View 2D Images

Honglin Zhu hzhu55, Kangyun Chen kchen127, Wenyu Jiao wjiao2, Zhengyuan Zhang zzhan266

May 9, 2023

1 Introduction

Background Nowadays, three-dimensional (3D) Reconstruction from two-dimensional (2D) images is widely used in many fields, including computer graphics, medical imaging, and so on. Besides, plenty of 2D images and 3D ground truth are required for a satisfactory 3D reconstruction result. However, to release the stress of collecting images from different viewpoints and 3D supervising annotations, here comes the importance of a single-view input source and self-supervision approach. Given single-view 2D inputs, our project aims to reconstruct 3D shapes in mesh representation, which consists of polygons with vertices and faces.

Related Work Liu et al. [1] propose a framework called Soft Rasterizer to reconstruct 3D shapes from 2D single-view images under self-supervision. The model comprises an encoder-decoder-based 3D polygon mesh generator and a differentiable renderer. Given two images from two viewpoints, a 3D mesh is generated based on the first image, while loss is obtained by comparing the difference between the second image and the 3D-rendered 2D output from the 3D mesh under a self-supervised learning technique.

2 Methods

Dataset Our baseline dataset [1] contains approximately 10k unique 3D models from 13 object categories in ShapeNetCore [2]. Each object is rendered from 24 azimuth angles with image resolution 64 × 64 × 4, with a fixed elevation angle under the same camera and lighting setup. Each object is also accompanied by a 3D ground truth voxel representation. For fair comparison, we adopted the same intra-categorical train/validate/test split as in [1], where the three datasets are already provided. To meet our current resources, we selected 6 categories of 3D objects from the baseline dataset (see Supplementary). Following the baseline setting, only 2 viewpoints were applied in our task and the ground truth voxel representation was only involved in the validation and test process.

(a) 2D Input (b) 2D Reference (c) 3D Truth

Figure 1: A Graphical Depiction of the Dataset

Baseline Model We choose Soft Rasterizer [1] based reconstruction network as our baseline model. According to the provided codebase, for each object to be reconstructed, two 2D images from two viewpoints are taken, one is for input and the other is for reference. Given the input 2D image, a 3D mesh representation is generated first, fed into a differentiable renderer to get a rendered 2D output. In the end, we will calculate the loss based on the difference between the rendered 2D image and the ground truth reference 2D image, where the differentiable render-

ing procedure ensures end-to-end backpropagation. However, 3D shapes are reconstructed only in shapes without color or texture information.

Differentiable Renderer In order to ensure self-supervised training for 3D reconstruction, a renderer is required, where a 2D image is generated given a 3D shape. However, in traditional rendering procedures, rasterization and z-buffering are not differentiable and cannot flow gradient information. However, Soft Rasterizer ensures differentiable rasterization and z-buffering by replacing them with probability maps computation and aggregations respectively.

Loss Denoting the rendered output and the reference images as I_{ren} and I_{ref} respectively, the Silhouette Loss L_s is defined as follows:

$$L_s = 1 - \frac{||I_{ren} \otimes I_{ref}||_1}{||I_{ren} \oplus I_{ref} - I_{ren} \otimes I_{ref}||_1}$$

where $\otimes$ and $\oplus$ denote element-wise product and sum respectively. Moreover, two weighted regularization terms are involved, which are Laplacian Loss L_l and Flatten Loss L_f. Therefore, the total loss is

$$L = L_s + \lambda_l L_l + \lambda_f L_f$$

Evaluation Even though 3D shapes are provided but not capitalized during training, they can still be used for performance evaluation. We evaluate the performance based on the 3D Intersection over Union (IoU) score. The higher the 3D IoU we get, the better performance the model obtains.

Custom Tweaks In order to improve the reconstruction performance beyond the provided baseline model, we have tried two different encoder structures:

1. ResNet-Like Encoder: with residual connections

2. Attention-Involved Encoder: with an additional Transformer encoder layer

3 Results

After implementing two different architectures to the model encoder, we got two new models. We trained these models and compared their performance with the baseline model. The testing result shows that the baseline model performs best (lowest loss and highest IoU). The specific results are shown in **Table 1**. We also compared the 3D mesh reconstruction objects generated by these different models. The result shows that all three models generate high-quality 3D objects. **Figure 2** shows the mesh generation outputs, where the three 3D mesh images are presented from different viewpoints for our generated object.

	Mean Loss	Mean IoU
Baseline	**0.204286**	**0.546209**
ResNet-Like Encoder	0.209622	0.540347
Attention-Involved Encoder	0.205251	0.538869

Table 1: Comparison of different encoder structures

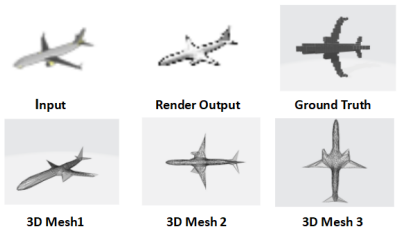

Figure 2: Baseline Model Input and Outputs

4 Discussion

The result shows that the baseline model has the best performance. We think one of the reasons is that the dataset used for training and testing might not be complex enough to benefit from the advanced architectural features in the ResNet-like and attention-involved models. The baseline model might perform better due to its simpler architecture and fewer parameters.

References

[1] S. Liu, T. Li, W. Chen, and H. Li, "Soft rasterizer: A differentiable renderer for image-based 3d reasoning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7708–7717, 2019.

[2] X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee, "Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision," *Advances in neural information processing systems*, vol. 29, 2016.