3D-RETR: End-to-End Single and Multi-View 3D Reconstruction with Transformers (Supplementary Material)

Zai Shi*[1]
zaishi@ethz.ch
Zhao Meng*[1]
zhmeng@ethz.ch
Yiran Xing[2]
yiran.xing@rwth-aachen.de
Yunpu Ma[3]
cognitive.yunpu@gmail.com
Roger Wattenhofer[1]
wattenhofer@ethz.ch

[1] ETH Zurich
[2] RWTH Aachen
[3] LMU Munich

1 3D-RETR with VQ-VAE

We describe in detail the VQ-VAE setting in our ablation study of Section 4.3 (see Figure 1). We train 3D-RETR with VQ-VAE in two separate stages.

In the first stage, we pretrain a VQ-VAE with a codebook size of 2048, where each codebook vector has 512 dimensions. The VQ-VAE Encoder and Decoder have three layers, respectively. For the VQ-VAE Decoder, we use the same residual blocks as in the CNN Decoder. The VQ-VAE Encoder encodes the $32 \times 32 \times 32$ voxel into a discrete sequence of length 64, where each element in the sequence is an integer between 0 and 2047. The VQ-VAE is trained with cross-entropy loss. The reconstruction IoU is about 0.885.

In the second stage, for every input image x and its correspondent ground-truth voxel Y, we first generate a discrete sequence D using the pretrained VQ-VAE Encoder. Then, the Transformer Encoder generates the hidden representation for the input image x, and the Transformer Decoder uses the output of the Transformer Encoder to generate another discrete sequence D'. To generate D', we use a linear layer with softmax at the output of the Transformer Decoder. We use the sequence D as the ground truth and train the Transformer Encoder and Decoder with cross-entropy loss to generate D', which should be as close as possible to D.

*Equal contribution.

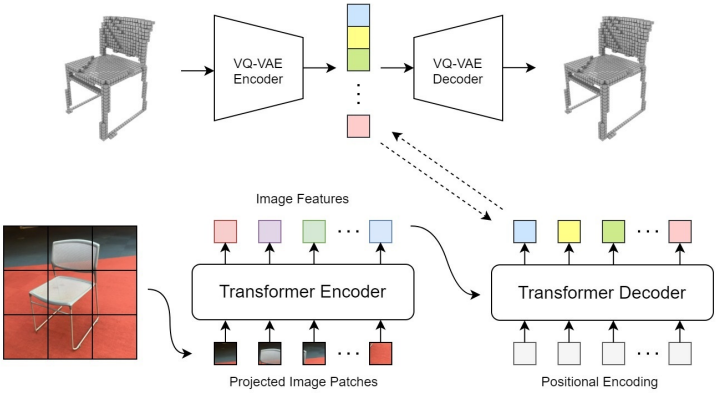

Figure 1: 3D-RETR with VQ-VAE. This corresponds to Setup 4 of our ablation study.

2 Additional Examples

We show more examples of the ShapeNet dataset and the Pix3D dataset from our 3D-RETR-B model. Table 1 shows additional examples of the Pix3D dataset. Table 2 shows examples from the ShapeNet dataset with different numbers of views as inputs. We can see a clear quality improvement when more views become available.

Table 1: Examples from the Pix3D dataset. All predictions are generated by 3D-RETR-B.

3 Model Performance with Different Views

In Table 2 of the paper, we show that 3D-RETR trained on three views still outperforms previous state-of-the-art results even when evaluated under different numbers of input views. In Table 3 and Figure 2, we give additional results on training and evaluating under different

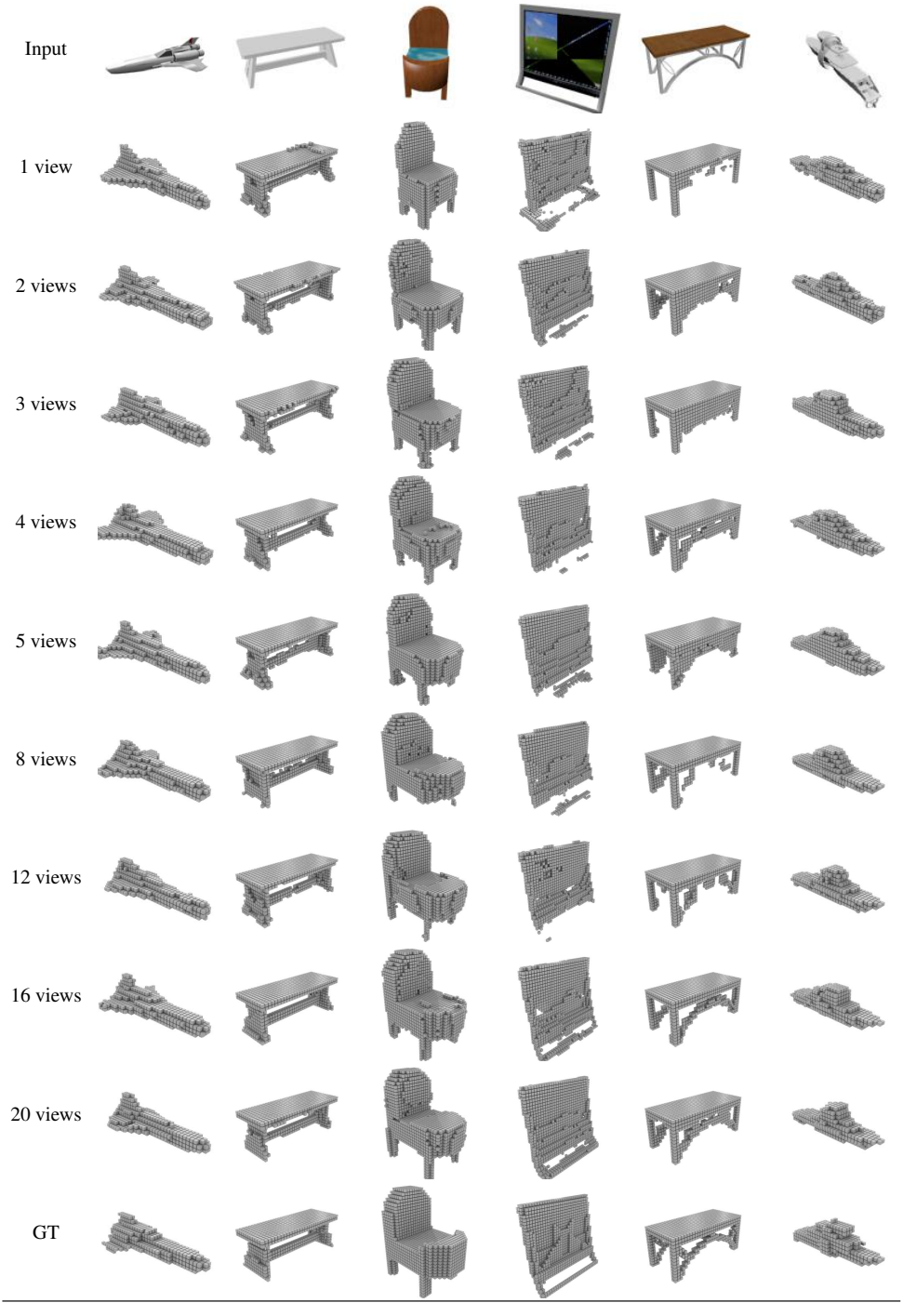

Table 2: Examples from the ShapeNet dataset. All predictions are generated by 3D-RETR-B.

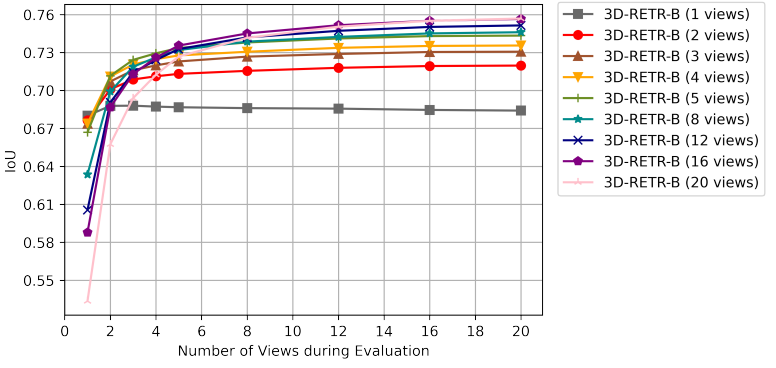

Figure 2: Models performance with different views.

numbers of views. We can observe that more views during evaluation can boost model performance. Another observation is that models trained with more views are not necessarily better than models trained with fewer views, especially when the number of views available during evaluation is far fewer than the number of available views during training. For example, when only one view is available, the model trained with one view reaches an IoU of 0.680, while the model trained with 20 views only reaches an IoU of 0.534.

Train \ Eval	1 view	2 views	3 views	4 views	5 views	8 views	12 views	16 views	20 views
1 view	**0.680**	0.688	0.688	0.687	0.687	0.686	0.686	0.685	0.684
2 views	0.676	0.701	0.709	0.711	0.713	0.716	0.718	0.719	0.720
3 views	0.674	0.707	0.716	0.720	0.723	0.729	0.729	0.730	0.731
4 views	0.674	0.711	0.721	0.725	0.728	0.731	0.734	0.735	0.736
5 views	0.667	**0.712**	**0.724**	**0.729**	0.734	0.738	0.741	0.743	0.743
8 views	0.634	0.699	0.719	0.726	0.732	0.739	0.742	0.745	0.746
12 views	0.606	0.691	0.714	0.724	0.733	0.742	0.747	0.750	0.751
16 views	0.588	0.687	0.713	0.726	**0.735**	**0.745**	**0.752**	**0.755**	**0.757**
20 views	0.534	0.657	0.694	0.712	0.727	0.742	0.750	**0.755**	**0.757**

Table 3: Model performance with different views during training and evaluation. **Bold** indicates the best performance in an evaluation setting.