Supplementary Materials: ViewPCGC: View-Guided Learned Point Cloud Geometry Compression

Anonymous Authors

1 MORE DETAILS OF THE NETWORK ARCHITECTURE

To clearly present the encoder and decoder structures of ViewPCGC, we depict the detailed configurations of some modules in Fig. 1. The module structures of IRN (adopted in [2]) and SSA are demonstrated in Fig. 2. Similarly, the main module architectures within the CCEM are shown in Fig. 3. Notably, our context prediction module (CPM) incorporates MaskConv. The cube in Fig. 3 represents the convolution kernel, where the blue parts are the anchor part, and the red parts are the non-anchor part. We employ the corresponding convolution kernels to compute the convolutions for anchor part and non-anchor part, respectively.

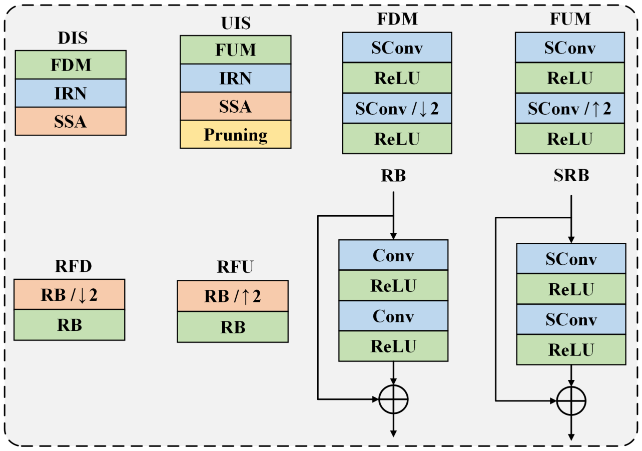

Figure 1: Detailed structures of certain modules in the encoder and decoder. "FDM" and "FUM" signify the feature downsampling and upsampling modules, respectively. “Pruning” is a selection mechanism aimed at identifying and retaining points with high probability of occurrence in the reconstruction point cloud. "SConv / ↓ 2" and "SConv / ↑ 2" describe the processes of downsampling and upsampling, respectively, employing a stride of 2 through sparse convolution. "RB / ↓ 2" and "RB / ↑ 2" denote the utilization of residual blocks for downsampling and upsampling, respectively, also with a stride of 2.

2 MORE VISUALIZATION RESULTS

To depict the reconstruction error of each point more accurately, we employ the Chamfer Distance [1] as an evaluation metric. We calculate the Chamfer Distance between each point of the compressed point cloud and the corresponding point in the original point cloud, denoted as the error map. The error map is displayed in Fig. 4. To ensure fairness, we attempt to equalize the bits per point (bpp) for all methods. Ultimately, the bpp consumption values are 0.27, 0.26, 0.22, 0.21, and 0.18, listed from left to right. It is evident that the reconstruction quality of ViewPCGC is significantly superior compared to other methods.

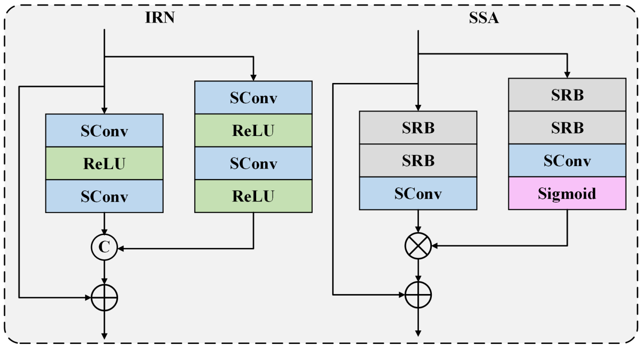

Figure 2: The architectures of IRN (adopted in [2]) and SSA.

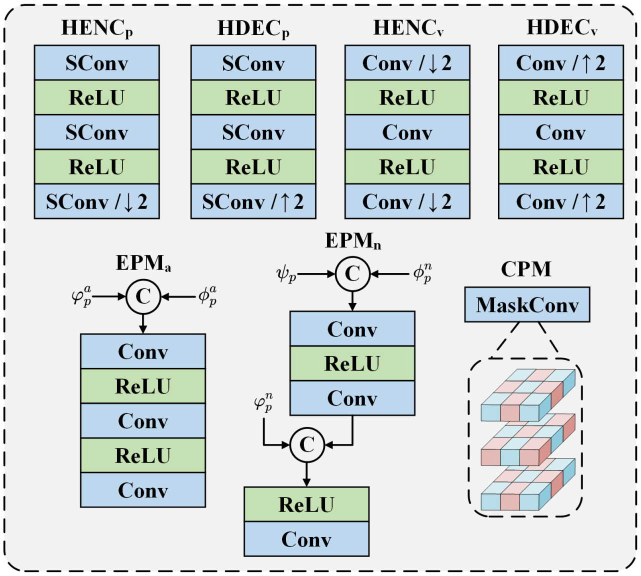

Figure 3: Detailed structures of certain modules in CCEM. "Conv / ↓ 2" and "Conv / ↑ 2" refer to the processes of downsampling and upsampling, respectively, with a stride of 2 using 2D convolution.

3 ABLATION STUDY OF LINEAR RESIDUAL ATTENTION

To assess the impact of using linear residual attention on RD performance and encoding/decoding time, we conduct additional experiments. The comparison involves two control groups: one using

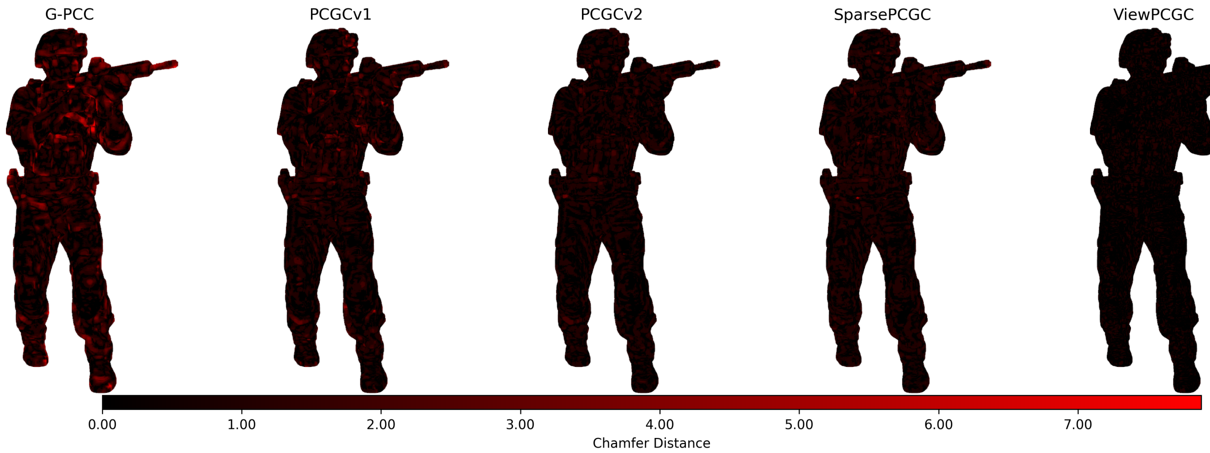

Figure 4: Error map visualization. Black represents minimal reconstruction errors, and red signifies substantial errors.

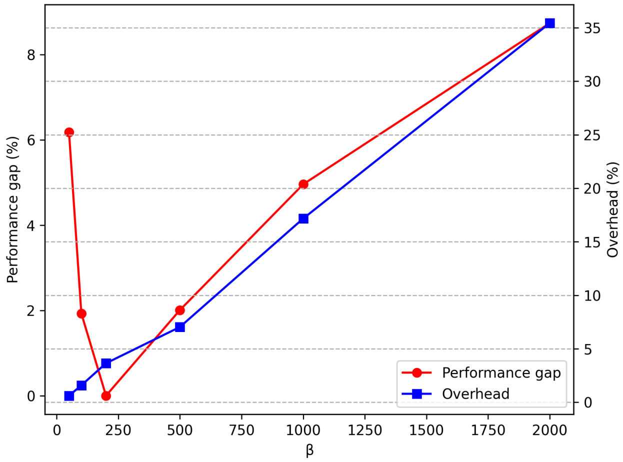

Figure 5: Experiment results for different values of β. The X-axis represents the value of β. The left Y-axis measures the performance gap under various β configurations, while the right Y-axis indicates the percentage of the view image's bitstream overhead in the total transmission bitstream.

non-local attention and another without any attention mechanism. We utilize the model with linear residual attention as the baseline, calculating the BD-rate (using D1-PSNR) for the other two models compared to the baseline. Moreover, we document the encoding and decoding time for each method, presenting the results in Table 1. The findings indicate that using non-local attention significantly increases encoding and decoding time, whereas linear residual attention does not noticeably extend the time. In terms of RD performance, models with non-local attention have a slight advantage, though the improvement is not substantial. In contrast, the absence of any attention mechanism leads to a marked decline in performance. Consequently, balancing both encoding/decoding time and RD performance, linear residual attention emerges as the optimal choice.

Table 1: Ablation study of linear residual attention.

Model	Enc/Dec (s)	BD-rate
baseline	1.33/2.15	-
w/ non-local attention	6.47/15.26	-1.53%
w/o attention	1.18/1.76	4.55%

4 ABLATION STUDY OF HYPER-PARAMETER β

In the previous section, we mention that as the view image bit stream increases, the RD performance of point cloud geometry compression will decline sharply. Conversely, the excessively low-quality view image may fail to enhance point cloud geometry compression performance. Therefore, we conduct an additional ablation study to explore the effect of improving image compression quality (i.e., increasing the value of β) on point cloud geometry compression. We train the model in four different alpha configurations ($\alpha = 1, 2, 5, 10$) to assess the performance gap across various beta configurations compared to the baseline model ($\beta = 200$ as mentioned in the training strategy). The results are illustrated in Fig. 5 as the red curve. Additionally, we calculate the ratio of the view image bit stream to the total transmission bit stream under different β configurations, shown by the blue curve in Fig. 5. The experimental results indicate that at extremely low bitrate, the poor image quality fails to provide adequate modality information for point cloud geometry compression, leading to a drop in performance. Conversely, excessive bitrate overhead becomes a major bottleneck when the image bitrate is too high, restricting further performance improvement in point cloud geometry compression.

REFERENCES

[1] Haoqiang Fan, Hao Su, and Leonidas J Guibas. 2017. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 605–613.

[2] Jianqiang Wang, Dandan Ding, Zhu Li, and Zhan Ma. 2021. Multiscale point cloud geometry compression. In *2021 Data Compression Conference (DCC)*. IEEE, 73–82.