

# Supplementary Material of Progressive Point Cloud Denoising with Cross-Stage Cross-Coder Adaptive Edge Graph Convolution Network

Anonymous Authors

In this supplementary document, we provide additional experimental results to further support and complement the conclusions presented in the main paper. Specifically, we provide experiments as follows:

- (1) More visual results on the PUNet dataset.
- (2) Visual Results on Real-world Scanned Data
- (3) Comparison of the running time of different denoising methods.
- (4) More comprehensive results and analyses from the ablation study are as follows:
  - Performance comparison on higher noise levels.
  - Ablation study of AEConv module on higher noise levels.
  - Ablation study of different connection architectures on higher noise levels.
  - Ablation study of the number of stages on higher noise levels.

## 1 MORE VISUAL RESULTS ON THE PUNET DATASET

In the main paper, we have presented that our method can achieve the best denoising results on the five point clouds, *i.e.*, Camel, Casting, Chair, Elk, and House, in the PUNet dataset [5]. However, due to limited page space, we could not include additional visual results in the main paper. In this section, to further demonstrate the effectiveness of our method, we have provided more visual comparison results in Figure 1. In general, we can observe that our proposed method also achieves superior visual results and better mean P2M metrics on the additional four point clouds. Therefore, compared to other state-of-the-art methods, our method can more accurately relocate noisy points onto clean surfaces, enhancing the visual quality of the point clouds.

## 2 VISUAL RESULTS ON REAL-WORLD SCANNED DATA

In the main paper, we conduct performance comparisons on real-world scanned data from the Kinect v1 and Kinect v2 datasets. To further demonstrate the effectiveness of our method, we visualize the denoising results on real-world scanned data. The visual results are shown in Figure 2. Due to the complexity of real-world noise, denoising real-world scanned data is challenging. In Figure 2, we can see that PDFlow performs poorly when faced with real-world noise. For the other comparison methods, there is little difference in their denoising capabilities in real-world scenarios. In contrast, our method can achieve better visual results on some scanned data, such as Boy. Therefore, this demonstrates that our method is effective in denoising real-world noise.

**Table 1: Running time comparison of different denoising methods on a noisy point cloud with a resolution of 50K points. The noisy point cloud contains Gaussian noise with standard deviation of 2% of the bounding sphere’s radius. The best performance results are indicated in BOLD.**

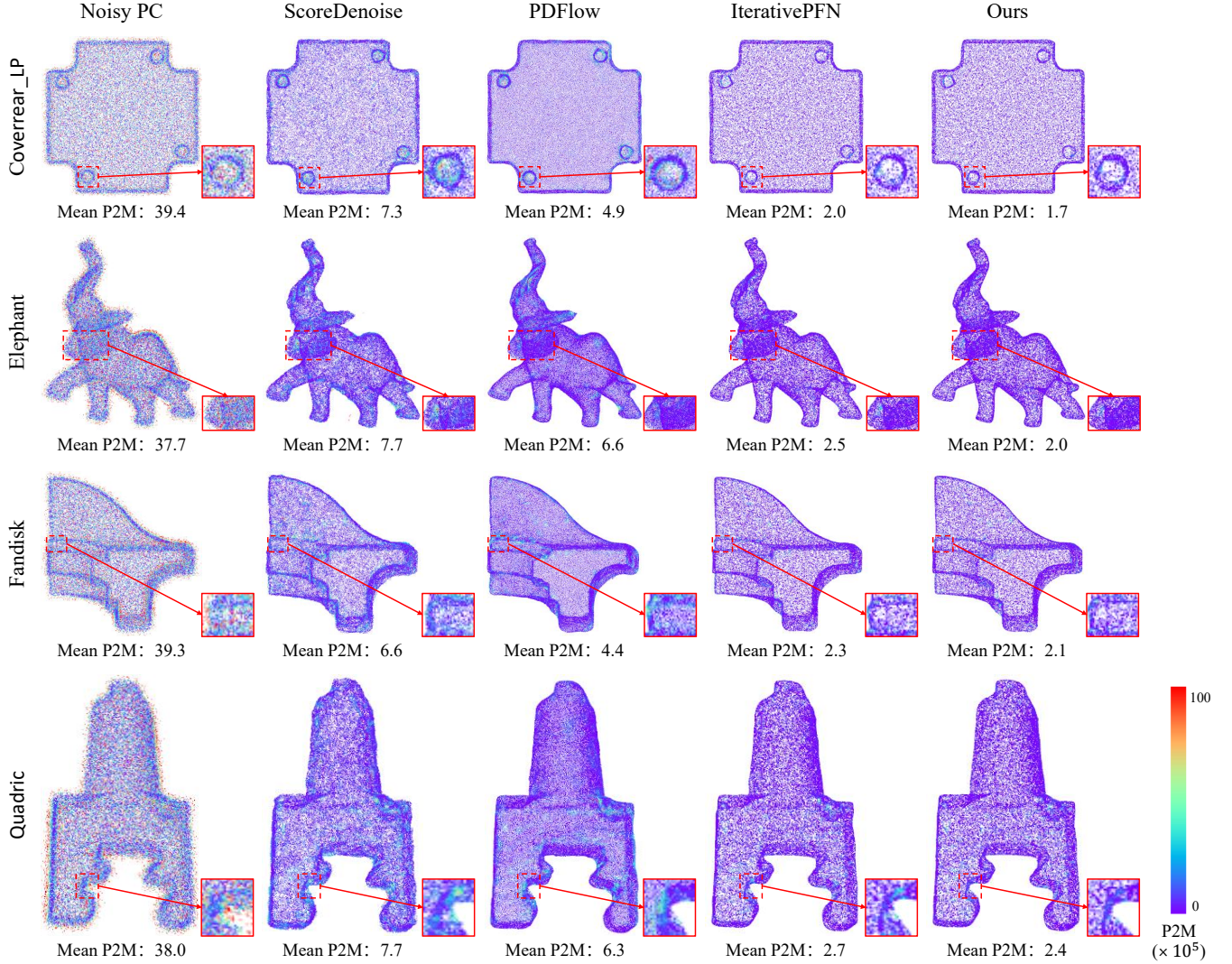
Method	Time (s)
PDFlow [3]	39.91
ScoreDenoise [2]	16.27
Pointfilter [6]	50.97
IterativePFN [1]	<b>15.83</b>
<b>C<sup>2</sup>AENet (Ours)</b>	16.14

**Table 2: Comparison results on higher noise levels on the PUNet dataset. CD is multiplied by  $10^5$ , P2M is multiplied by  $10^5$ . The best results are marked in BOLD.**

Model	10K points					
	2.75% noise		3% noise		3.25% noise	
	CD↓	P2M↓	CD↓	P2M↓	CD↓	P2M↓
PDFlow [3]	39.93	20.04	44.72	23.95	50.55	28.80
ScoreDenoise [2]	49.60	23.47	56.38	28.79	64.81	35.63
IterativePFN [1]	36.58	12.63	42.49	16.99	52.83	24.93
<b>C<sup>2</sup>AENet (Ours)</b>	<b>34.88</b>	<b>11.32</b>	<b>38.35</b>	<b>13.85</b>	<b>45.44</b>	<b>19.06</b>

## 3 COMPARISON OF THE RUNNING TIME OF DIFFERENT DENOISING METHODS

For large-resolution point clouds, the running time of denoising methods is crucial. Therefore, in this section, we conduct a running time comparison of state-of-the-art methods on a noisy point cloud with a resolution of 50K points. The results are shown in Table 1. For other state-of-the-art methods, we use their default configurations. For instance, PDFlow and Pointfilter undergo two iterations of denoising during testing. The number of gradient ascent steps for ScoreDenoise is set to 30. Both IterativePFN and our method are configured with four denoising stages. Overall, our method achieves optimal denoising performance while ensuring suboptimal running time. Compared to methods with fewer stages, our method yields superior denoising results with less running time.



**Figure 1: Visual results of four additional point clouds on the PUNet dataset, where the color of each point represents its P2M. The noisy point cloud contains Gaussian noise with standard deviation of 2% of the bounding sphere’s radius, and it has a resolution of 50K points. The mean P2M of each point cloud is directly below.**

**Table 3: Experimental results of AEConv module on higher noise levels on the PUNet database. CD is multiplied by  $10^5$ , P2M is multiplied by  $10^5$ . The best results are marked in BOLD.**

Method	10K points					
	2.75% noise		3% noise		3.25% noise	
	CD↓	P2M↓	CD↓	P2M↓	CD↓	P2M↓
EdgeConv	36.67	12.64	44.48	18.42	59.01	29.62
EdgeConv + GS	35.25	11.63	39.24	14.59	48.16	21.26
EdgeConv + GS + AEA	<b>34.88</b>	<b>11.32</b>	<b>38.35</b>	<b>13.85</b>	<b>45.44</b>	<b>19.06</b>

## 4 MORE COMPREHENSIVE RESULTS AND ANALYSES FROM THE ABLATION STUDY

### 4.1 Performance Comparison on Higher Noise Levels

In Table 2, we further investigate the performance of different denoising methods on higher noise levels. All models in the experiment are trained on Gaussian noise with standard deviations ranging from 0.5% to 2% of the bounding sphere’s radius. Following that, the models are tested on Gaussian noise with standard deviations of 2.75%, 3 %, and 3.25% of the bounding sphere’s radius, respectively. As a result, the noise level during testing is significantly higher than during training. From the experimental results, we observe that ScoreDenoise performs less satisfactorily as the



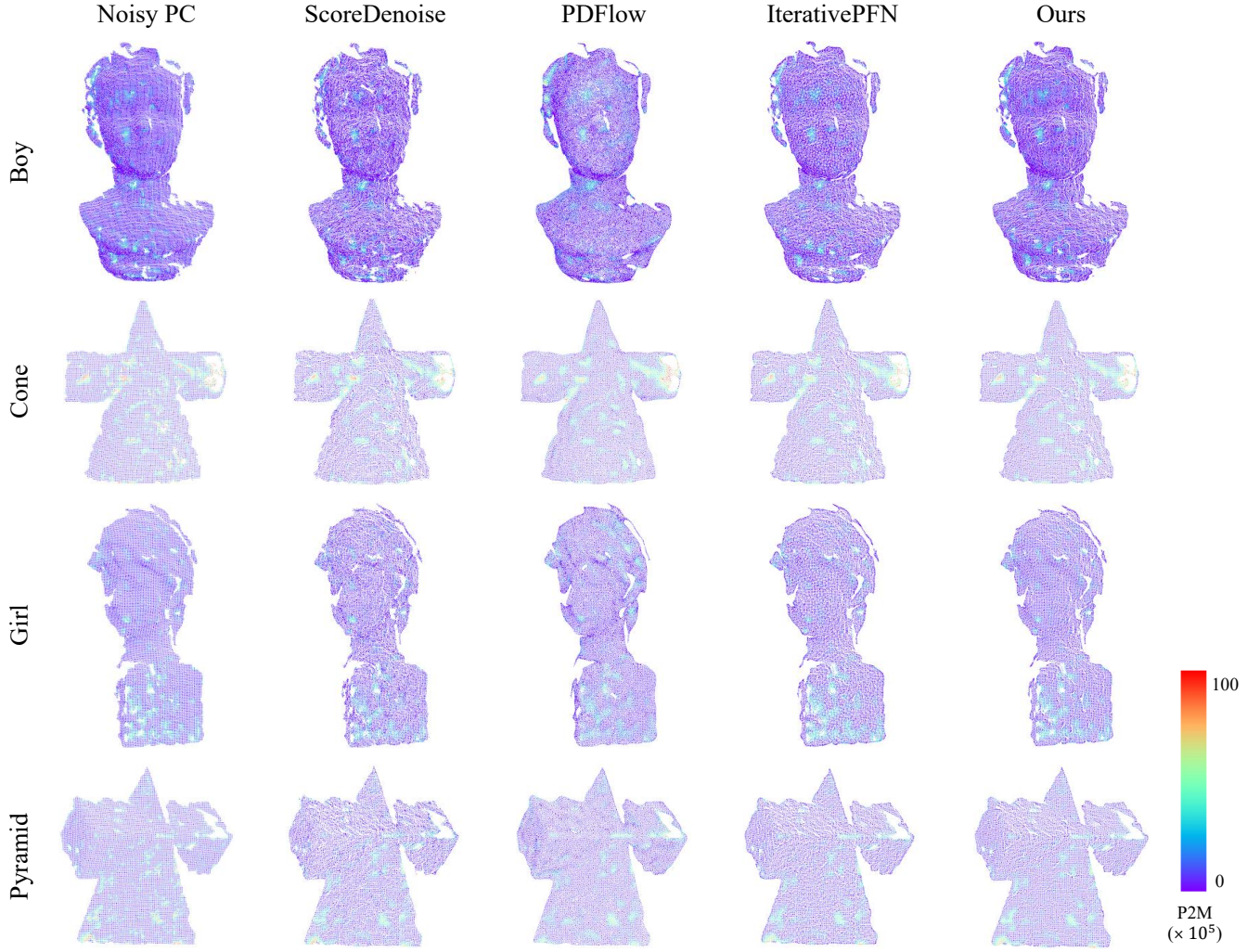


Figure 2: Visual results of four scans on the Kinect v1 dataset, where the color of each point represents its P2M. The real-world scanned data includes Boy, Cone, Girl, and Pyramid.

noise level increases. Compared to other state-of-the-art denoising methods, our method consistently outperforms them across all high noise levels. This demonstrates the effectiveness and robustness of  $C^2$  AENet in challenging scenarios.

#### 4.2 Ablation Study of AEConv Module on Higher Noise Levels

To further validate the effectiveness and contribution of the AEConv module, we conduct an additional ablation studies. Unlike the experiments presented in the main text, this study evaluates the contributions of the EdgeConv baseline [4], multi-level graph structure, and adaptive edge attention (AEA) module on higher noise levels. The results are presented in Table 3. In the table, we use GS to represent multi-level graph structures. We observe that the multi-level graph structure leveraging different neighborhood information is beneficial for denoising. On this basis, the AEA module

further achieves a significant performance leap, demonstrating its ability to focus the network on critical denoising regions. Therefore, the AEConv module remains effective in capturing the complex relationships between data even on higher noise levels.

#### 4.3 Ablation Study of Different Connection Architectures on Higher Noise Levels

In Table 4, we conduct additional experiments to assess the performance of different connection architectures on higher noise levels. We find that the baseline achieved suboptimal performance on higher noise levels. Adding the cross-stage connections of the encoder and decoder separately or simultaneously to the baseline resulted in unsatisfactory performance on high noise levels. Ultimately, the linear mapping of cross-stage features effectively solves this issue, making the transferred information more suitable for learning in the next stage.

**Table 4: Experimental results of five different connection architectures on higher noise levels on the PUNet database. CD is multiplied by  $10^5$ , P2M is multiplied by  $10^5$ . The best results are marked in BOLD.**

Method	10K points					
	2.75% noise		3% noise		3.25% noise	
	CD↓	P2M↓	CD↓	P2M↓	CD↓	P2M↓
Base	36.79	12.60	41.97	16.42	51.61	23.73
Base+Cross-Stage (Encoder)	37.28	12.93	44.49	18.28	58.61	29.13
Base+Cross-Stage (Decoder)	38.48	13.79	47.98	20.97	61.94	31.90
Base+Cross-Stage Cross-Coder	39.02	14.14	49.29	21.86	64.46	33.79
Base+Cross-Stage Cross-Coder (FC)	<b>34.88</b>	<b>11.32</b>	<b>38.35</b>	<b>13.85</b>	<b>45.44</b>	<b>19.06</b>

**Table 5: Experimental results of different number of stages on higher noise levels on the PUNet database. CD is multiplied by  $10^5$ , P2M is multiplied by  $10^5$ . The best results are marked in BOLD.**

Method	10K points					
	2.75% noise		3% noise		3.25% noise	
	CD↓	P2M↓	CD↓	P2M↓	CD↓	P2M↓
Our network with 1 stage	44.43	18.13	51.91	23.88	61.59	31.57
Our network with 2 stages	37.61	13.07	42.92	16.97	51.33	23.66
Our network with 4 stages	<b>34.88</b>	<b>11.32</b>	<b>38.35</b>	<b>13.85</b>	<b>45.44</b>	<b>19.06</b>
Our network with 8 stages	37.16	12.74	43.91	17.73	55.69	26.77

#### 4.4 Ablation Study of the Number of Stages on Higher Noise Levels

Table 5 shows the experimental results of the multi-stage denoising scheme on higher noise levels. It primarily evaluates our network with 1, 2, 4, and 8 stages. The results indicate that the performance gains of the multi-stage denoising scheme on higher noise levels align with the trends observed in the main paper. The optimal performance is attained when configuring the network with 4 stages. This further demonstrates the importance of multi-stage denoising scheme in enhancing denoising capabilities.

#### REFERENCES

- [1] Dasith de Silva Edirimuni, Xuequan Lu, Zhiwen Shao, Gang Li, Antonio Robles-Kelly, and Ying He. 2023. IterativePFN: True Iterative Point Cloud Filtering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13530–13539.
- [2] Shitong Luo and Wei Hu. 2021. Score-based point cloud denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4583–4592.
- [3] Aihua Mao, Zihui Du, Yu-Hui Wen, Jun Xuan, and Yong-Jin Liu. 2022. PD-Flow: A point cloud denoising framework with normalizing flows. In *European Conference on Computer Vision*. Springer, 398–415.
- [4] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. 2019. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (tog)* 38, 5 (2019), 1–12.
- [5] Lequan Yu, Xianzhi Li, Chi-Wing Fu, Daniel Cohen-Or, and Pheng-Ann Heng. 2018. PU-Net: Point Cloud Upsampling Network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [6] Dongbo Zhang, Xuequan Lu, Hong Qin, and Ying He. 2020. Pointfilter: Point cloud filtering via encoder-decoder modeling. *IEEE Transactions on Visualization and Computer Graphics* 27, 3 (2020), 2015–2027.