
Supplementary Material for “U-CAN: Unsupervised Point Cloud Denoising with Consistency-Aware Noise2Noise Matching”

Junsheng Zhou^{1*} Xingyu Shi^{1*} Haichuan Song^{2†} Yi Fang³
Yu-Shen Liu^{1†} Zhizhong Han⁴

School of Software, Tsinghua University, Beijing, China ¹

Computer Science and Technology, East China Normal University, Shanghai, China ²

Center for AI and Robotics (CAIR), NYU Abu Dhabi, UAE ³

Department of Computer Science, Wayne State University, Detroit, USA⁴

A Implementation Details

We train our denoise network with the Adam [2] optimizer. We set a peak learning rate of 1×10^{-3} that gradually decreases following a reduce learning rate on plateau scheduler. We train U-CAN with a single 3090 GPU and the optimization takes 30 epochs to converge.

B Point Cloud Denoising on Out-of-Distribution Noise Types

Dataset Settings. To evaluate the performance of U-CAN in handling out-of-distribution noises, we conduct evaluations under more types of point noises including the simulated real-world LiDAR noises, uniform noises and impulsive noises. The experiments are conducted under PUNet [7] dataset. The uniform noise data is achieved by adding uniformly distributed noises to the clean point clouds with a range of $[-0.03, 0.03]$. We generate impulsive noise data to simulate the outlier points with a ratio of 0.05.

Comparisons. We quantitatively compare the proposed U-CAN with the state-of-the-art methods for both supervised and unsupervised point cloud denoising, as shown in Tab. 1. These methods include the classic optimization-based GLR [8]; supervised learning-based methods such as DMR [3] and ScoreDenoise [4]; as well as unsupervised methods, including DMR-TTD and ScoreD-TTD. Note that U-CAN is only trained under Gaussian noises, where those noises are out of the learned distribution. The results show our superiority in handling different types of noises including real-world conditions, demonstrating the potential of applying U-CAN to real-world applications.

C More Evaluations on Point Cloud Denoising

Smaller Noise Levels. We compare U-CAN with SOTA methods including classic method GLR [8] and learning-based method DMR-TTD [3, 1] and ScoreD-TTD [4, 1] on smaller noise levels, i.e., 0.2%, 0.5% in Tab. 2. The results demonstrate that U-CAN outperforms the SOTA classic and learning-based methods under smaller point noise levels. We further evaluate the results with clean point clouds as inputs to show how the denoising methods affect the clean point clouds. Similar to the SOTA classic point cloud denoising method GLR [8], U-CAN can better keep the original perfect inputs while other learning-based methods significantly distorts them.

Larger Noise Levels. We further compare U-CAN with SOTA methods on larger noise levels, i.e., 4%, 5% in Tab. 3, where U-CAN also outperforms other baselines. The results demonstrate that

*Equal contribution. †Corresponding authors.

Table 1: Point cloud denoising results under PU-Net dataset with different types of noise.

Dataset: PU	10K, LiDAR		10K, Uniform		10K, Impulsive	
Method	CD	P2M	CD	P2M	CD	P2M
GLR [8]	3.420	1.589	3.677	1.622	2.051	1.827
DMR [3]	4.749	2.267	4.906	2.366	4.497	2.159
ScoreD [4]	3.219	1.195	3.529	1.235	1.987	1.609
DMR-TTD	8.690	5.682	9.110	5.974	6.900	4.426
ScoreD-TTD	4.189	1.985	4.451	2.025	2.740	1.894
Ours	2.994	1.495	3.252	1.623	1.421	0.854

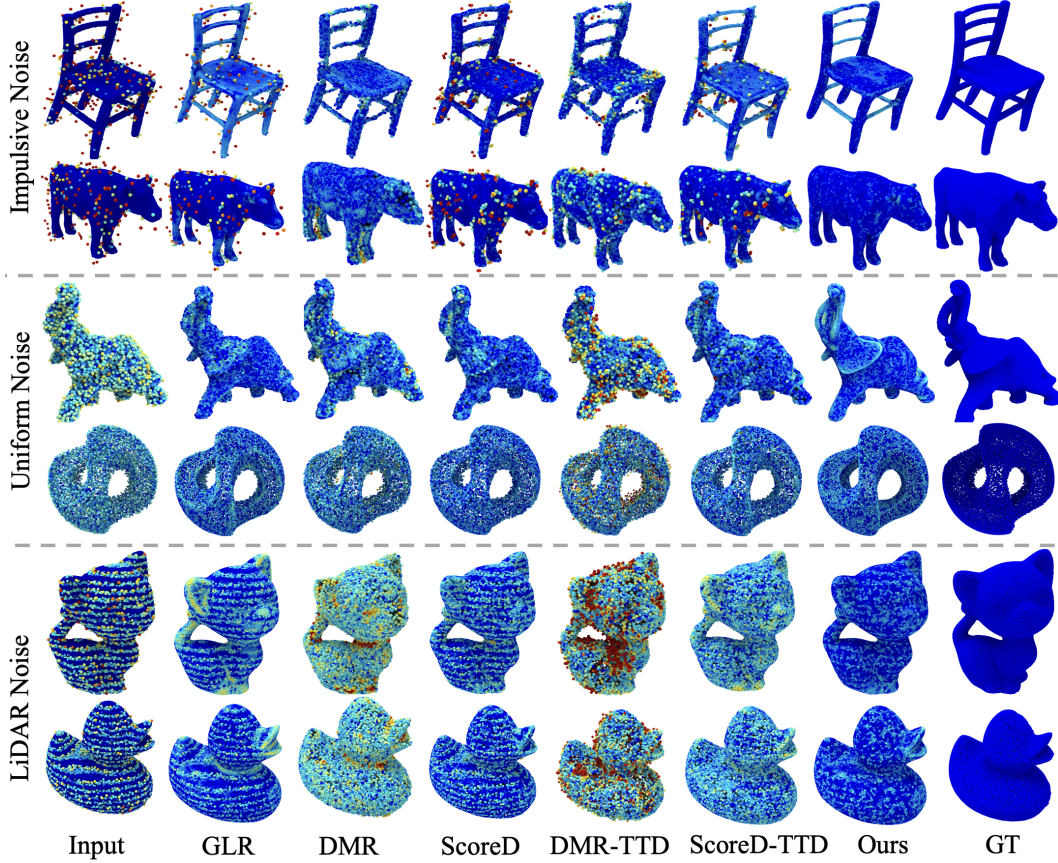


Figure 1: Comparison in point cloud denoising results under PU-Net dataset with different types of noise. The noise error at each point is shown in color, where the points closer to the ground truth surface are shown in darker blue, indicating smaller error. And those with larger error are shown in lighter red.

U-CAN performs more robustly in handling difficult noise situations, especially when compared with previous SOTA learning-based methods DMR-TTD and ScoreD-TTD.

C.1 Comparisons with Over-Fitting Methods

Efficiency comparison. We report the efficiency and numerical comparisons with the over-fitting based approach NoiseMap [5] under the setting of 3% noise of 10K points in PUNet dataset. The results in Tab. 4 demonstrate that U-CAN achieves comparable results with NoiseMap with much faster inference. NoiseMap overfits each noisy point cloud for learning a signed distance function, where each point cloud takes about more than 10 minutes to converge. On the contrary, our learning-

Table 2: Point cloud denoising results under PU-Net dataset with smaller noise levels.

Dataset: PU	10K, Clean		10K, 0.2%		10K, 0.5%	
Method	CD	P2M	CD	P2M	CD	P2M
GLR [8]	1.285	1.300	1.397	1.299	1.991	1.312
DMR-TTD [3, 1]	6.689	4.242	6.748	4.302	7.163	4.513
ScoreD-TTD [4, 1]	1.731	0.970	1.914	0.975	2.467	1.018
Ours	0.767	0.657	0.870	0.681	1.425	0.791

Table 3: Point cloud denoising results under PU-Net dataset with larger noise levels.

Dataset: PU	10K, 4%		10K, 5%	
Method	CD	P2M	CD	P2M
GLR [8]	5.489	3.092	6.318	3.791
DMR-TTD [3, 1]	12.24	8.716	13.70	10.06
ScoreD-TTD [4, 1]	8.865	5.637	10.79	7.32
Ours	4.355	2.380	5.605	3.376

based approach only takes about 10s to denoise a single point cloud during inference. Therefore, we do not include NoiseMap on the Tab. 1 of the main paper, since we focus on the learning-based point cloud denoising which enables a fast inference.

Visual comparison.

We provide a visual comparison with the over-fitting based approach NoiseMap under the PU-Net [7] dataset in Fig. 2. The visualizations are conducted under the setting of 10k points with Gaussian noise at a standard deviations of 3% of the bounding sphere’s radius. The denoising results of NoiseMap are obtained with the official codes. As shown, U-CAN achieves slightly more visual appealing denoising results compared to NoiseMap. Notice that NoiseMap requires more than 10 mins to overfit each noisy point cloud, where U-CAN only requires about 10s for denoising at inference. Even so, U-CAN also achieves comparable performance with NoiseMap, which demonstrates the robustness and superiority of U-CAN.

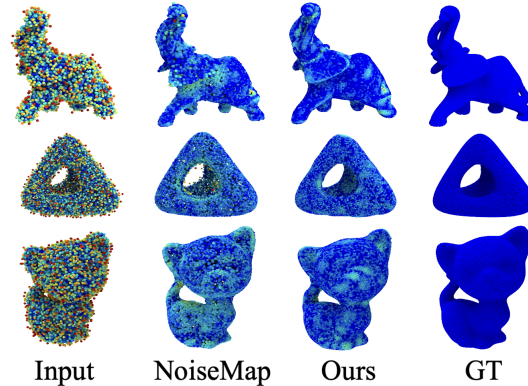


Figure 2: Visual comparison with NoiseMap.

D More Ablation Studies

The Weights of Losses. We perform detailed ablation studies to investigate the effects of different weighted combinations of the two specifically designed loss functions. As illustrated in Tab. 5, the default weighting configuration results in the highest denoising accuracy. Adjusting the weights to

Table 4: Comparisons with over-fitting method NoiseMap.

Ablation	CD	P2M	Time
NoiseMap [5]	4.221	1.847	627.0 s
Ours	3.666	1.842	10.9 s

Table 5: Ablation studies on the loss weights.

Dataset: PU		10K, 1%		10K, 2%		10K, 3%	
\mathcal{L}_{N2N}	\mathcal{L}_{DC}	CD	P2M	CD	P2M	CD	P2M
$\times 0.1$	$\times 1$	53.96	45.61	52.61	43.86	51.45	42.12
$\times 1$	$\times 0.1$	2.145	0.696	3.366	1.390	6.463	3.834
$\times 1$	$\times 1$	2.497	1.105	3.234	1.255	3.666	1.842

Table 6: Ablation studies on point cloud patch sizes.

Dataset: PU	10K, 1%		10K, 2%		10K, 3%	
Ablation	CD	P2M	CD	P2M	CD	P2M
250	2.821	1.302	3.327	1.515	4.239	2.152
500	2.843	1.432	3.390	1.703	3.898	2.004
1000	2.497	1.105	3.234	1.255	3.666	1.842
2000	2.616	1.175	3.247	1.547	3.880	1.994

favor one of the loss functions over the other leads to a decrease in performance, indicating that the balance between the two losses is critical for optimal denoising effectiveness. Note that both losses are implemented using EMD, so their magnitudes are consistent.

The Sizes of Point Cloud Patches. We conduct ablations to investigate the impact of varying the sizes of point patches on the performance. The results in Tab. 6 indicate that a patch size of 1,000 points is optimal for U-CAN. Through these studies, we demonstrate that selecting a larger or smaller patch size results in performance degradations, indicating a critical balance in the number of points per patch for optimal model performance.

The Input Patterns. At inference time, the input is a single noisy point cloud. During training, the inputs are several noisy observations with the same numbers of points, which can be achieved by: a) multiple noisy observations of the same object. The manner is supported by modern LiDAR systems which can capture multiple (10-30) noisy observations per second. b) several subsets of a dense noisy observation. For example, we can achieve 5 training instances of 10k points from a noisy observation of 50k points. We show the ablations of training U-CAN with these two input patterns in Tab. 7, where the results demonstrate the effectiveness of both kinds of input.

E More Visual Comparisons

E.1 Visual Comparison on Synthetic Data

For a comprehensive visual comparison with previous supervised and unsupervised approaches in point cloud denoising, we provide more visualizations under PU-Net dataset in Fig. 3. As shown, U-CAN significantly outperforms other unsupervised approaches (e.g. TTD [1], DMR-TTD [3]) and the supervised approaches PCN [6] and DMR. Moreover, U-CAN also produces comparable denoising predictions with the supervised ScoreDenoise [4].

Table 7: Ablation studies on input patterns.

Dataset: PU	10K, 1%		10K, 2%		10K, 3%	
Ablation	CD	P2M	CD	P2M	CD	P2M
Multiple \mathcal{P}	2.497	1.105	3.234	1.255	3.666	1.842
Subsets of One \mathcal{P}	2.580	1.231	3.287	1.661	3.605	1.869

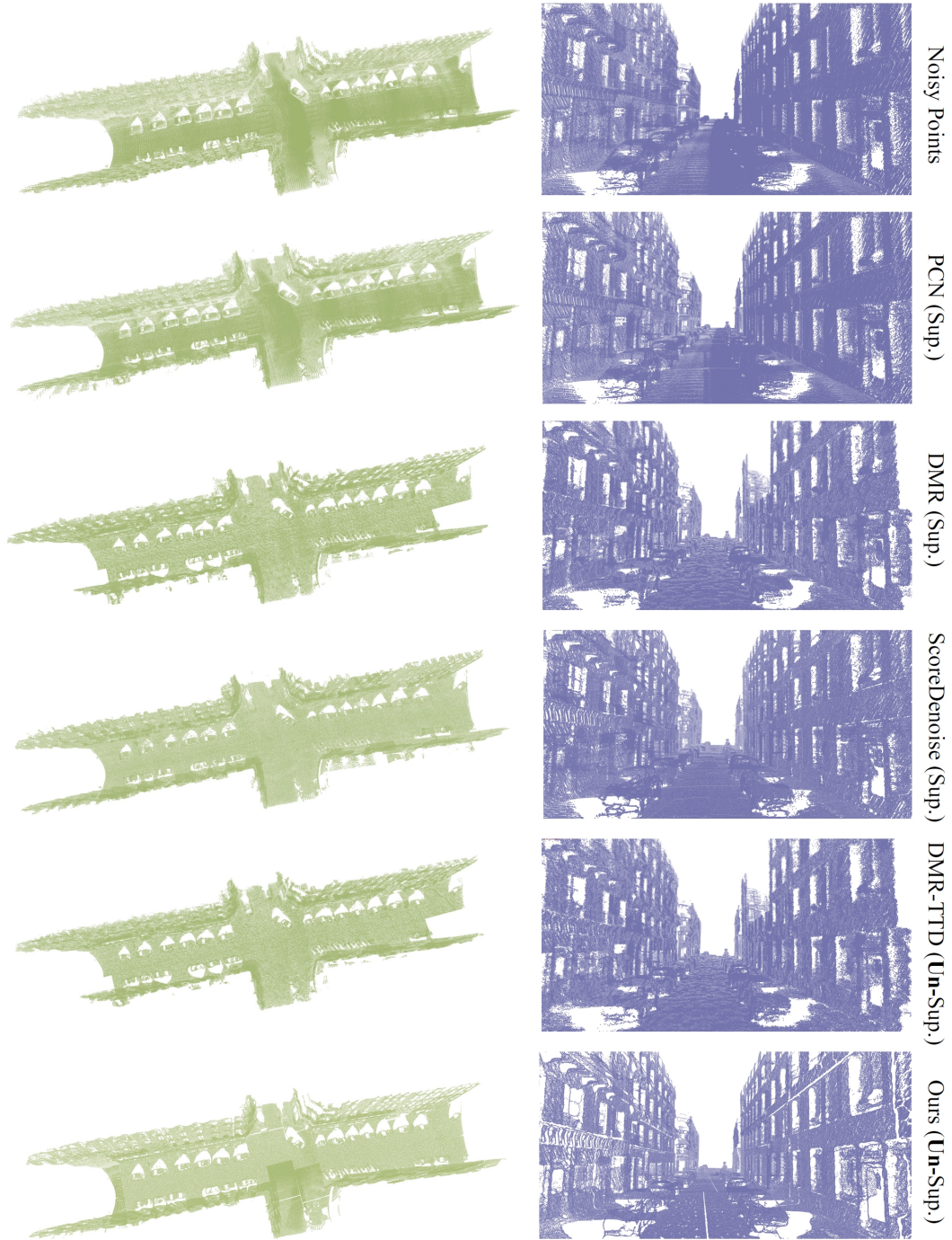


Figure 4: Denoising on the real scans under Paris-rue-Madame dataset. We show the visual comparison with the supervised (Sup.) and unsupervised (Un-Sup.) approaches on the overall scene geometries. **Left:** The noisy points and the denoised points achieved with different approaches under the whole scene. **Right:** The zoom-in visualization of the scene denoising predictions on the streets.

E.2 Visual Comparison on Scanned Data

We further provide the visual comparison with SOTA supervised and unsupervised methods on the overall scene geometries under Paris-rue-Madame dataset. As shown in Fig. 4, U-CAN significantly outperforms other baselines in terms of smoothness and cleanliness. Consequently, our method demonstrates a notable proficiency in eliminating noise artifacts, which are commonly encountered in real-world point cloud data.

F Limitations

U-CAN may occasionally produce overly smooth point cloud denoising results due to the absence of ground-truth geometric details that are typically leveraged in supervised denoising frameworks. This limitation is commonly observed in unsupervised methods across various domains, including image denoising, point cloud denoising, and audio denoising.

References

- [1] Pedro Hermosilla, Tobias Ritschel, and Timo Ropinski. Total denoising: Unsupervised learning of 3d point cloud cleaning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 52–60, 2019. 1, 3, 4
- [2] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1
- [3] Shitong Luo and Wei Hu. Differentiable manifold reconstruction for point cloud denoising. In *Proceedings of the 28th ACM international conference on multimedia*, pages 1330–1338, 2020. 1, 2, 3, 4
- [4] Shitong Luo and Wei Hu. Score-based point cloud denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4583–4592, 2021. 1, 2, 3, 4
- [5] Baorui Ma, Yu-Shen Liu, and Zhizhong Han. Learning signed distance functions from noisy 3d point clouds via noise to noise mapping. In *International Conference on Machine Learning (ICML)*, 2023. 2, 3
- [6] Marie-Julie Rakotosaona, Vittorio La Barbera, Paul Guerrero, Niloy J Mitra, and Maks Ovsjanikov. Pointcleannet: Learning to denoise and remove outliers from dense point clouds. In *Computer graphics forum*, volume 39, pages 185–203. Wiley Online Library, 2020. 4
- [7] Lequan Yu, Xianzhi Li, Chi-Wing Fu, Daniel Cohen-Or, and Pheng-Ann Heng. Pu-net: Point cloud upsampling network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2790–2799, 2018. 1, 3
- [8] Jin Zeng, Gene Cheung, Michael Ng, Jiahao Pang, and Cheng Yang. 3d point cloud denoising using graph laplacian regularization of a low dimensional manifold model. *IEEE Transactions on Image Processing*, 29:3474–3489, 2019. 1, 2, 3