This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

An End-to-End Robust Point Cloud Semantic Segmentation Network with Single-Step Conditional Diffusion Models

Wentao Qu[1], Jing Wang[2], YongShun Gong[3], Xiaoshui Huang[4*], Liang Xiao[1*]
NJUST[1], THU[2], SDU[3], SJTU[4]
{quwentao, jwang}@njust.edu.cn, huangxiaoshui@163.com, xiaoliang@mail.njust.edu.cn

Abstract

*Existing conditional Denoising Diffusion Probabilistic Models (DDPMs) with a Noise-Conditional Framework (NCF) remain challenging for 3D scene understanding tasks, as the complex geometric details in scenes increase the difficulty of fitting the gradients of the data distribution (the scores) from semantic labels. This also results in longer training and inference time for DDPMs compared to non-DDPMs. From a different perspective, we delve deeply into the model paradigm dominated by the Conditional Network. In this paper, we propose an end-to-end robust semantic **Seg**mentation **Net**work based on a **C**onditional-Noise Framework (CNF) of **DDPMs**, named **CDSegNet**. Specifically, CDSegNet models the Noise Network (NN) as a learnable noise-feature generator. This enables the Conditional Network (CN) to understand 3D scene semantics under multi-level feature perturbations, enhancing the generalization in unseen scenes. Meanwhile, benefiting from the noise system of DDPMs, CDSegNet exhibits strong robustness for data noise and sparsity in experiments. Moreover, thanks to CNF, CDSegNet can generate the semantic labels in a single-step inference like non-DDPMs, due to avoiding directly fitting the scores from semantic labels in the dominant network of CDSegNet. On public indoor and outdoor benchmarks, CDSegNet significantly outperforms existing methods, achieving state-of-the-art performance.*

1. Introduction

Point cloud, as a fundamental 3D representation, provides the most crucial data structure support for 3D tasks. Benefiting from the rapid development of 3D devices and continuous innovation in data synthesis techniques, large-scale scene point clouds have become accessible [1, 5, 10, 13]. Therefore, accurate semantic understanding of 3D scenes, applied to a wide range of 3D downstream tasks such as autonomous driving [47], robotic technology [36], and virtual reality [23], has gained increasing attention.

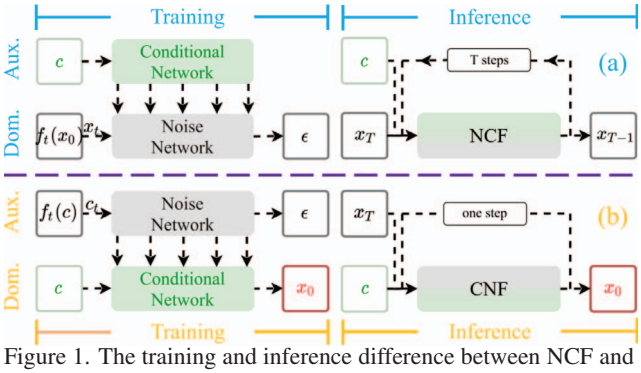

Figure 1. The training and inference difference between NCF and CNF. NCF dominated by NN, relies on the noise fitting quality, requiring extensive training and inference iterations. In contrast, CNF alleviates the noise fitting necessity by focusing on CN, cleverly avoiding this issue, alongside retaining the DDPM robustness.

Encouraged by deep learning, a large number of learnable point cloud semantic segmentation methods have achieved significant results in recent years [8, 24, 50, 52, 53, 64]. Nevertheless, these methods often overlook the fact that raw point clouds from 3D devices are usually perturbed and sparse [5, 42], making them sensitive to data noise and sparsity [21, 43, 56]. This limits the optimal segmentation accuracy, especially recognizing in object boundaries and small objects [60, 61].

Along another research line, DDPMs [16] with a strong noise-robust denoising architecture, originated from and flourished in image generation [11, 28, 41], have been explored in various 3D tasks [34, 39, 54, 66]. They typically consider the 3D task as a conditional generation problem, built upon the Noise-Conditional Framework (NCF, see Fig. 1(a)). In general, the Conditional Network (CN) extracts the conditional features for generating guidance. Meanwhile, the Noise Network (NN) predicts the scores [45] from task targets, dominating the results of tasks.

Unfortunately, this framework poses challenges when applied to real-time 3D scene understanding tasks, such as autonomous driving. This is because, to better fit the scores from task targets, DDPMs typically require more training and inference steps compared to non-DDPMs [44, 45]. Al-

*Corresponding Author. https://github.com/QWTforGithub/CDSegNet

though some methods can accelerate the DDPM sampling process [31, 44], they still require dozens of steps, alongside a sub-optimization process. Moreover, the complex distribution of 3D scenes makes it difficult for DDPMs to achieve satisfactory results in an end-to-end manner. (see Fig. 2)

To address the above problems, in this paper, we **rethink** the existing end-to-end framework of conditional DDPMs, initially revealing some key insights into the DDPM advantages and limitations in point cloud semantic segmentation.

Inspired by these core insights, we design a Conditional-Noise Framework (CNF, see Fig. 1(b)) of DDPMs, effectively preserving advantages and circumventing limitations. Unlike NCF, CNF treats NN and CN as the auxiliary network and the dominant network in 3D tasks, respectively. In this way, NN in CNF can still inherit the noise construction system of DDPMs, thus preserving the noise and sparsity robustness from DDPMs. Meanwhile, this also enables NN in CNF to relax the requirement of fitting the scores from task targets. Therefore, CNF allows the model to follow the DDPM pattern in training but not in inference, overcoming the demand for extensive iterations from DDPMs.

Furthermore, we propose an end-to-end robust point cloud semantic segmentation network based on CNF, called CDSegNet. Specifically, CDSegNet treats NN as a lightweight noise-feature generator. The noise information from NN, effectively filtered via a Feature Fusion Module (FFM), reasonably perturbs the semantic features in CN. This motivates the generalization ability of CDSegNet in unknown scenes [32, 40, 51]. Meanwhile, under CNF, CDSegNet can follow the noise addition pattern of DDPMs during training, maintaining the robustness to noise from the modeled distribution [39] and the robustness to sparse scenes and insufficient data. Moreover, benefiting from the dominance of CN in CNF, CDSegNet can be regarded as a non-DDPM during inference. This enables CDSegNet to produce the semantic labels in a single-step inference.

To the best of our knowledge, we are the first end-to-end attempt to introduce DDPMs into point cloud semantic segmentation [30, 65]. We lower the threshold and encourage more researchers to further explore applications of DDPMs to 3D tasks. Our key contributions can be summarized as:

- We systematically analyze and identify the advantages and limitations of DDPMs with a Noise-Conditional Framework in point cloud semantic segmentation, offering new knowledge for DDPMs in 3D tasks.
- We design a Conditional-Noise Framework of DDPMs, preserving strengths while circumventing shortcomings.
- We propose an end-to-end robust point cloud segmentation network based on CNF, CDSegNet, exhibiting strong robustness and requiring only a single-step inference.
- Comprehensive experiments on large-scale indoor and outdoor benchmarks demonstrate that CDSegNet achieves significant performance and strong robustness.

2. Related Works

Learnable Point Cloud Semantic Segmentation. Benefiting from the powerful data-driven capability of deep learning, directly extracting features from point clouds to understand 3D scene semantics has become possible [8, 37, 38, 48]. Inspired by the aforementioned, a multitude of methods have emerged in recent years, achieving significant success in point cloud semantic segmentation. Early methods usually utilize RNNs to establish interactions between point cloud slices, attempting to improve feature extraction effectiveness through building ordered relationships within point clouds [19, 62]. However, the substantial computational cost greatly limits the input scale. To resolve this problem, some researchers focus on large-scale scene segmentation [12, 18, 26]. Although remarkable progress has been made, the methods are still constrained by a small receptive field, limiting the further improvement in segmentation results. Lately, some methods built upon Transformers have been proposed. These methods inherit the capability of modeling long-range dependencies, overcoming the limitations of the feature receptive field and thereby achieving state-of-the-art results [14, 24, 52, 53, 59, 64].

Although existing methods focusing on segmentation accuracy have achieved impressive results, they overlook the fact that raw point clouds often exhibits perturbed and sparse. This leads to them sensitive to data noise and sparsity. In this paper, we introduce DDPMs with a Conditional-Noise Framework to address the above problem. This demonstrates the strong robustness to data noise and sparsity, while avoiding extensive training and inference steps.

DDPMs for 3D Tasks. DDPMs, succeeded in image generation, have conducted some explorations in various 3D tasks. This typically transforms the 3D task as a conditional generation problem. [33] first introduces DDPMs into point cloud generation, providing inspiration for subsequent explorations. Then, some works attempt to extend DDPMs to point cloud completion [34, 66]. Subsequently, [39] undertakes preliminary investigations into DDPMs for point cloud upsampling. Furthermore, several works have integrated DDPMs into point cloud semantic segmentation using a pre-training (two-stage training) approach [30, 65].

Although some explorations demonstrate the potential of DDPMs in 3D tasks, the two-stage training requirement in scene semantic understanding tasks demonstrate that DDPMs still face the challenge of fitting the scores in complex 3D scenes. Meanwhile, the dozens or even thousands of inference steps limit practical applications in 3D tasks with real-time requirements. In this paper, we propose an end-to-end robust point cloud semantic segmentation network based on our CNF of DDPMs. Thanks to CNF, our method circumvents directly fitting the scores from semantic labels in the dominant segmentation network, requiring only a single-step inference.

3. Conditional DDPMs for PCSS

In this section, we first introduce conditional DDPMs. Next, we identify the reason behind leveraging conditional DDPMs for point cloud semantic segmentation (PCSS) and discuss the limiting factors associating with this approach.

3.1. Background

Given a target data $\boldsymbol{x_0} \sim P_{data}$, a guiding condition $\boldsymbol{c} \sim P_c$ and a latent variable $\boldsymbol{z} \sim P_{noise}$, conditional DDPMs follow an auto-regressive process [39, 44]: a predefined diffusion process q that gradually destroys the data content until $\boldsymbol{x_0}$ degrades into $\boldsymbol{z}$, and a trainable conditional generation process p_θ that slowly generates the specific result until $\boldsymbol{z}$ is recovered to $\boldsymbol{x_0}$ under the guidance of the condition $\boldsymbol{c}$. We can apply the framework to multiple 3D tasks [33, 34, 39]. In PCSS, $\boldsymbol{x_0}$ represents the target semantic label, while $\boldsymbol{c}$ means the segmented point cloud.

We consider the noise as the fitting target, due to the better performance observed in experiments [16]. Then, the training objective under specific conditions is [39]:

$$L(\theta) = \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,I)}||\boldsymbol{\epsilon} - \boldsymbol{\epsilon_\theta}(\boldsymbol{x_t}, C)||^2, \tag{1}$$

where C=$\{\boldsymbol{c}, t\}$ represents the conditions, while $t \sim \mathcal{U}(T)$ (T=1000). Therefore, unconditional generation ($\boldsymbol{c} = \emptyset$) can be viewed as a special case of conditional DDPMs conditioned on the time label t for controlling the noise level. That is, our analysis is generalizable to any type of DDPMs.

Meanwhile, under stochastic differential equations (SDEs), the target noise in conditional DDPMs can be converted into and from the score (the gradient of the data distribution) by a constant factor $\alpha = -\frac{1}{\sqrt{1-\overline{\alpha}_t}}$ [46] :

$$\alpha\boldsymbol{\epsilon_\theta}(\boldsymbol{x_t}, C) = s_\theta(\boldsymbol{x_t}, C) \approx \nabla_{\boldsymbol{x_t}} \log P_t(\boldsymbol{x_t}). \tag{2}$$

3.2. What Supports the Use of DDPMs in PCSS?

Point clouds in real-world scenes are often noisy and sparse [5, 42]. Existing methods overlook this fact, making them sensitive to data noise and sparsity [21, 43, 56]. In this paper, we introduce DDPMs to address this issue.

Noise robustness. Benefiting from the noise system, ***DDPMs inherently present the robustness to noise from the modeled distribution*** [39]. We further reveal the key sources of the robustness in the system: ***the noise samples and the noise fitting***. According to Eq. 1, DDPMs can access multi-level noise samples $\boldsymbol{x_t}$ from the modeled distribution and use the standard noise $\boldsymbol{\epsilon}$ as the fitting target. This facilitates the model to understand the task information under modeled distribution perturbations, enhancing the adaptability to the relevant distribution noise. Sec. 5.3 further discusses and supports the conclusion.

Sparsity robustness. ***DDPMs exhibit better performance on under-sampled data (sparse scenes and less data)***. In fact, this noise-adding manner can be formally regarded as a kind of ***data augmentation*** [6, 51, 57]:

$$L(\theta) = \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,I)}||\boldsymbol{\epsilon} - \boldsymbol{\epsilon_\theta}(f_t(\boldsymbol{x_0}), \boldsymbol{C})||^2, \tag{3}$$

where $f_t(\boldsymbol{x_0}) = \sqrt{1-\overline{\alpha}_t}\boldsymbol{\epsilon} + \sqrt{\overline{\alpha}_t}\boldsymbol{x_0}$ [16]. The linear function $f_t(\cdot)$ maps $\boldsymbol{x_0}$ to a more ambiguous latent distribution.

This data augmentation can significantly alleviate the model overfitting (see Tab. 7 and Fig. 9), enhancing the generalization for outdoor sparse scenes (Sec. 5.1, Sec. 5.2 and Sec. 5.5) and less data (Sec. 5.4).

3.3. What Limits the Use of DDPMs in PCSS?

Unfortunately, DDPMs may hardly be applied to tasks with high real-time requirements, due to the extensive training and inference iterations they demand (see Fig. 2).

As described in Eq. 1, the performance of DDPMs essentially lies in ***the noise fitting quality*** (the proof in the supplementary material). To better approximate the noise target, ***DDPMs require more training and inference iterations than non-DDPMs***, due to the significant error of fitting distributions with a large difference in one step [31, 44, 45].

We provide a proof, under a unified setting for DDPMs and non-DDPMs, DDPMs require more steps T to converge. For a semantic segmentation task, given a network f_θ with sufficient fitting ability and a semantic sample pair $(\boldsymbol{c}, \boldsymbol{x_0})$, the training objective of DDPMs and non-DDPMs can be consistently formulated as (assuming using MSE):

$$L_\theta = \frac{1}{T}\sum_{t=1}^{T}||\boldsymbol{y_{t-1}} - f_\theta(\boldsymbol{x_t}, \boldsymbol{c})||^2, \tag{4}$$

where the target $\boldsymbol{y_{t-1}}$ means the target noise conditioned on the time label t in DDPMs. Meanwhile, the input $\boldsymbol{x_t}$=$\boldsymbol{\mu_t} + \boldsymbol{\sigma_t\epsilon}$ [16], and we omit the time label t as part of the input.

The inference process commonly follows:

$$\boldsymbol{y'_{t-1}} = f_\theta(\boldsymbol{x_t}, \boldsymbol{c}), \quad t \in [1, T], \tag{5}$$

where $\boldsymbol{y'_{t-1}}$ means the predicted noise in DDPMs.

According to Eq. 4 and Eq. 5, the sufficient convergence for DDPMs requires at least the step size T>1 to accommodate all $\boldsymbol{y_{t-1}}$ (training) and $\boldsymbol{y'_{t-1}}$ (inference), due to the significant error in one step. However, non-DDPMs only necessitate the step size T=1 (the input $\boldsymbol{x_t}$=$\emptyset$):

$$L_\theta = ||\boldsymbol{y_0} - f_\theta(\emptyset, \boldsymbol{c})||^2, \quad \boldsymbol{y'_0} = f_\theta(\emptyset, \boldsymbol{c}), \tag{6}$$

where the target $\boldsymbol{y_0}$=$\boldsymbol{x_0}$, while $\boldsymbol{y'_0}$ means the predicted $\boldsymbol{x_0}$.

4. Methodology

4.1. Conditional-Noise Framework

To maintain the advantage sources (Sec. 3.2) and circumvent the limitation factors (Sec. 3.3), we design

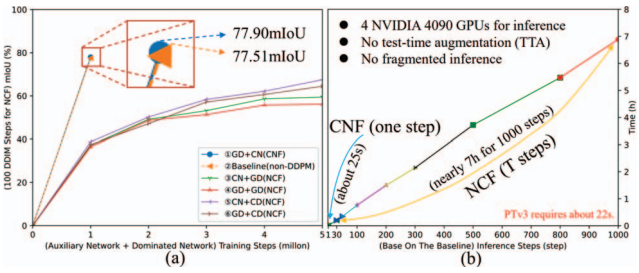

Figure 2. We try several combinations for conditional DDPMs built on the baseline (Sec. 5.1) on ScanNet in (a). GD+CD (NCF) indicates that NN and CN are modeled as Gaussian [16] and categorical [2] diffusion (details in the supplementary material). To better fit noise, these combinations dominated by NN (③,④,⑤,⑥) require more iterations to converge than CNF, but exhibiting poorer performance, due to complex scene distribution. (b) shows the inference time cost of CNF and NCF under the same baseline. CNF achieves better performance with fewer iterations.

a Conditional-Noise Framework (CNF) of DDPMs (see Fig. 1(b)). Unlike the Noise-Conditional Framework (NCF), CNF regards the Conditional Network (CN, f_ψ) as the dominant task backbone and uses the Noise Network (NN, ϵ_θ) as a auxiliary feature augmentation branch.

Specifically, to perturb the features in CN, the condition $\boldsymbol{c}$ is added with noise instead of the target $\boldsymbol{x_0}$ in NN:

$$L(\theta) = \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,I)}||\boldsymbol{\epsilon} - \epsilon_\theta(\boldsymbol{c_t}, t)||^2. \tag{7}$$

Next, a Feature Fusion Module (FFM) is used to filter the noise information from NN, ensuring the feature perturbations in a reasonable manner.

Furthermore, CN learns the task-related information under multi-level feature perturbations via FFM:

$$L(\psi) = l_{task}(\boldsymbol{x_0}, f_\psi(f_t(\boldsymbol{c}), \boldsymbol{c})), \tag{8}$$

where $f_t(\boldsymbol{c})$=$\mathcal{FFM}(\boldsymbol{c_t^{noise}})$ means a nonlinear function. $\mathcal{FFM}(\cdot)$ represents the Feature Fusion Module. $\boldsymbol{c_t^{noise}}$ means the noise feature from ϵ_θ with $\boldsymbol{c_t}$ as input. $l_{task}(\cdot)$ indicates the task-related loss function.

This simple and effective model paradigm:

- **Following Noise Construction System.** Eq. 7 indicates that the noise system of DDPMs is still retained in NN, aligning with Eq. 1, maintaining the noise robustness.
- **Aligning with Feature Augmentation.** Eq. 8 shows that CN follows the feature augmentation pattern, aligning with Eq. 3, thereby enhancing the sparsity robustness for sparse scenes and less data.
- **Relaxing Noise Fitting Requirement.** Eq. 7 and Eq. 8 mean that the model performance relies on CN rather than NN, relaxing the noise fitting requirement, avoiding excessive training and inference iterations from DDPMs.

Moreover, NN in CNF allows us to transcend the limitation of modeling Gaussian diffusion [16]. This means that CNF can use any distribution of DDPMs [3] (see Fig. 2).

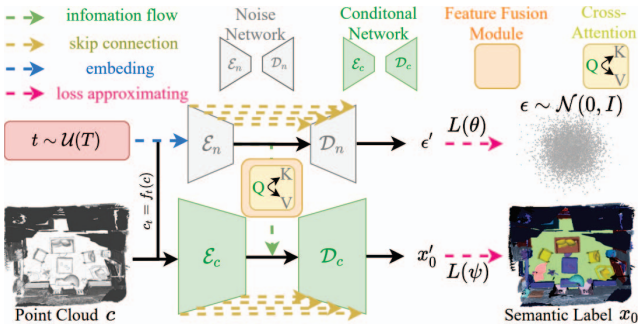

Figure 3. The overall framework of CDSegNet. The auxiliary Noise Network (NN), seen as a noise-feature generator, modeling the diffusion process conditioned on the time label, perturbs the input features at different noise levels. Meanwhile, the Feature Fusion Module (FFM) controls the noise information flow direction, achieving the semantic feature augmentation by reasonably filtering the perturbations. Furthermore, the dominant Conditional Network (CN) predicts the segmentation results in a pure manner.

4.2. Network Architecture

In this section, we introduce the overall architecture of CDSegNet aligned with CNF. This consists of three primary components: the auxiliary Noise Network (NN), the Feature Fusion Module (FFM), and the dominant Conditional Network (CN), as clearly illustrated in Fig. 3 (the parameter and optimization details in the supplementary material).

The Auxiliary Noise Network. NN constructs the noise system of DDPMs, perturbing the conditional point cloud. As mentioned in Sec. 4.1, NN should exhibit lighter, due to the insignificant noise fitting requirement. This follows the Transformer-U-Net architecture [52, 64], stacking two-stage standard Transformer blocks in the encoder and decoder. Meanwhile, similar to [52, 53], we utilize the effective grid pooling to achieve upsampling and downsampling. Moreover, the insignificant noise fitting also means that introducing the time label t in NN is sufficient to model the diffusion process without considering additional conditions.

The Feature Fusion Module. FFM directs the information flow from NN to CN at the bottleneck stage. In fact, FFM can adaptively filter the noise information, making the feature augmentation from NN in a reasonable way, as excessive perturbations may harm the performance of CN.

Specifically, FFM first performs the feature projection via MLPs, $F_{cn} \in \mathbb{R}^{N_{cn}\times C_{cn}} \rightarrow (Q) \in \mathbb{R}^{N_{cn}\times C}$, $F_{nn} \in \mathbb{R}^{N_{nn}\times C_{nn}} \rightarrow (K, V) \in \mathbb{R}^{N_{nn}\times C}$. Subsequently, FFM filters the noise information via a Cross-Attention block [49]:

$$\begin{aligned} O &= mlp(WV) + F_{cn}, \\ F &= ffn(O) + O, \end{aligned} \tag{9}$$

where $W \in \mathbb{R}^{N_{cn}\times N_{nn}} = softmax(\frac{QK^T}{\sqrt{C}})$.

We consider the unidirectional flow $F_{nn} \xrightarrow{FFM} F_{cn}$, due

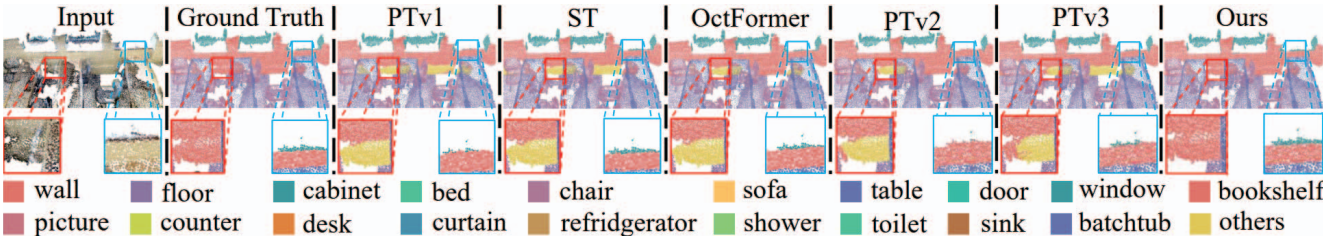

Figure 4. The visualization results on ScanNet. Our method achieves better semantic segmentation results in object boundaries, such as the boundary between the wall and the wall (red solid box) and the boundary between the wall and the window (blue solid box).

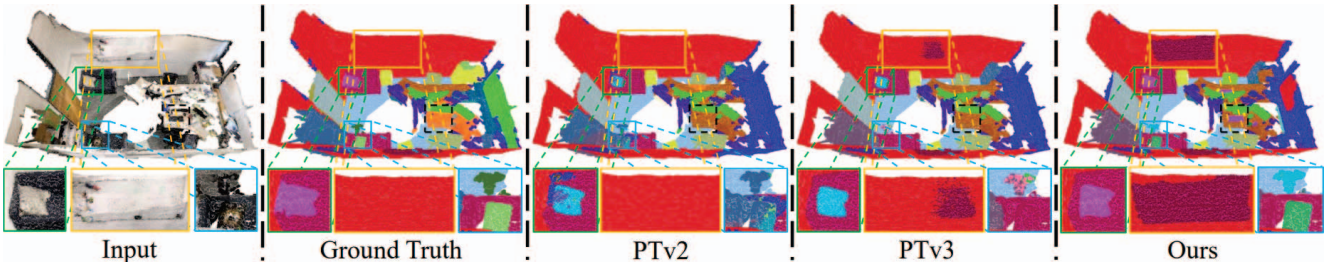

Figure 5. The visualization results on ScanNet200. Our method achieves the clearer segmentation results in small object recognition, and segments the categories not labeled in the Ground Truth, such as the keyboard (black dashed box) and the whiteboard (orange solid box).

to the better trade-off between the performance and computational cost and the sufficient diffusion modeling in NN.

The Dominant Conditional Network. CN follows the architecture of NN, with four stages for both the encoder and decoder, focusing on the segmentation results. Meanwhile, besides the feature perturbations from NN, we use only the conditional point cloud as input, ensuring that CN concentrates purely on the scene semantic understanding.

4.3. Training and Inference

Training. Following [52, 53], we constrain the semantic segmentation results in CN:

$$L(\psi) = CE(\boldsymbol{x_0}, \boldsymbol{x_0^{'}}) + \lambda Lovaz(\boldsymbol{x_0}, \boldsymbol{x_0^{'}}), \tag{10}$$

where $CE(\cdot)$ represents the cross-entropy loss, while $Lovasz(\cdot)$ follows [4]. $\lambda = 1$ means a weighting factor.

Furthermore, we can optimize CNF from a multi-task perspective (**the noise fitting in NN** and **the semantic label fitting in CN**). This applies the Geometric Loss Strategy (GLS) [7], with a geometric mean weight, to alleviate the convergence speed difference between NN and CN, regulating the perturbations from NN in a reasonable manner:

$$L_{total} = \prod_{i=1}^{n} \sqrt[n]{L_i} = \sqrt{L(\theta)} \cdot \sqrt{L(\psi)}, \tag{11}$$

where $n = 2$ means the number of tasks.

Inference. Thanks to CNF, CDSegNet can be seen a non-DDPM during inference, requesting only one step:

$$\boldsymbol{x_0^{'}} = \mathcal{D}_c(\mathcal{E}_c(\boldsymbol{c}), \mathcal{FFM}(\mathcal{E}_n(\boldsymbol{c_T}, T))), \tag{12}$$

where $\boldsymbol{c_T} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, while $\boldsymbol{x_0^{'}}$ means the predicted semantic label.

5. Experiments

5.1. Experiment Setup

Dataset. Two indoor benchmarks (ScanNet [10], ScanNet200 [42]) and one outdoor benchmark (nuScenes [5]) are used for evaluations. For ScanNet and ScanNet200, we divide train/val/test with 1201/312/100 scenes like [52, 53]. Meanwhile, the official protocol is followed to split train/val/test into 700/150/150 scenes for nuScenes. The point clouds in ScanNet, ScanNet200 and nuScenes are voxelized into 0.02m, 0.02m and 0.05m, respectively.

Baseline. To demonstrate the effectiveness of our method, we set a baseline. This eliminates the diffusion modeling in NN of CDSegNet, converting NN into a lightweight CN that approximates the input using MSE.

5.2. Comparison of Segmentation Results

Indoor Dataset. We first conduct evaluations on indoor datasets. Tab. 1 shows that CDSegNet achieves significant performance on ScanNet and ScanNet200. Meanwhile, CDSegNet outperforms all other methods across all metrics on ScanNet. This is because, benefiting from the strong noise robustness, CDSegNet can achieve better results on object boundaries with more perturbations. Fig. 4 shows the visualization on ScanNet, further supporting the viewpoint. Moreover, CDSegNet also demonstrates significant results in recognizing small objects within more complex and perturbed scenes. As shown in Fig. 5, CDSegNet

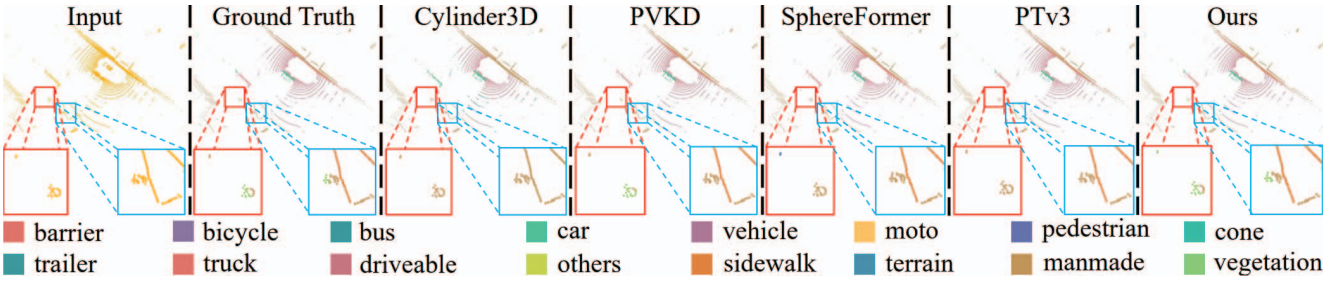

Figure 6. The visualization results on nuScenes. Our method produces remarkable results in small object recognition, such as the vegetation (red solid box and blue solid box).

can more clearly identify small objects, such as the pillow (green solid box) and the whiteboard (orange solid box).

Actually, as shown in Fig. 5, the Ground Truth sometimes includes incorrect annotations (e.g., some points of sofa are labeled as the pillow category) or may even omit annotations (e.g., the whiteboard and the keyboard). This may be one of the reasons why most models perform poorly on ScanNet200. Meanwhile, the significant results on mIoU indicate that our CDSegNet can achieve better performance in perturbed environments compared to other methods.

Methods	ScanNet [10]			ScanNet200 [42]		
	mIoU	mAcc	allAcc	mIoU	mAcc	allAcc
PTv1 [64]	70.8	76.4	87.5	29.8	40.5	78.4
MinkUNet [8]	72.3	79.4	89.1	28.3	39.8	77.9
ST [24]	74.3	82.5	90.7	-	-	-
OctFormer [50]	75.0	83.1	91.3	32.9	42.4	81.2
PTv2 [52]	75.5	82.9	91.2	31.4	42.0	80.7
PTv3 [53]	77.6	85.0	92.0	35.3	**46.0**	83.3
Baseline	77.5	85.1	91.9	35.4	45.5	83.6
Ours	**77.9**	**85.2**	**92.2**	**36.3**	45.9	**83.9**

Table 1. The results on ScanNet and ScanNet200. Our method surpasses other methods in almost all metrics.

Outdoor Dataset. We also conduct the validation on the outdoor benchmark. Compared to indoor scenes, the spatial distance between points in outdoor point clouds is larger. This causes more significant data sparsity. Tab. 2 shows that CDSegNet exhibits the excellent results, significantly outperforming existing methods on the mIoU. As mentioned in Sec. 4.1, CNF aligns with a feature augmentation strategy, improving generalization on sparse scenes. Fig. 6 further demonstrates the superiority of CDSegNet in small object recognition in large outdoor sparse scenes, such as the vegetation (red solid box and blue solid box).

5.3. Validation for Noise Robustness

We further validate the noise robustness of CDSegNet. This adds a Gaussian perturbation $\boldsymbol{n_G} \sim \mathcal{N}(\boldsymbol{n_G}; \mathbf{0}, \tau \boldsymbol{I})$ to the normalized inputs of models [15, 39], i.e., $\boldsymbol{c'} = \boldsymbol{c} + \boldsymbol{n_G}$.

Fig. 7 demonstrates the strong noise robustness of CDSegNet on ScanNet and ScanNet200. Meanwhile, we can observed that Ours and Ours-x_0 are stronger than the baseline, validating the conclusion in Sec. 3.2. However, Ours-x_0 performs significantly weaker than Ours. Although Ours-x_0 retains the noise sample $\boldsymbol{x_t}$ during training, the fitting target is $\boldsymbol{x_0}$, not $\boldsymbol{\epsilon}$. This leads to the model lacking the further noise adaptability in scenes. We believe that **the noise fitting in the noise system of DDPMs contributes mainly to the noise robustness** (we hope to validate the conclusion in the future).

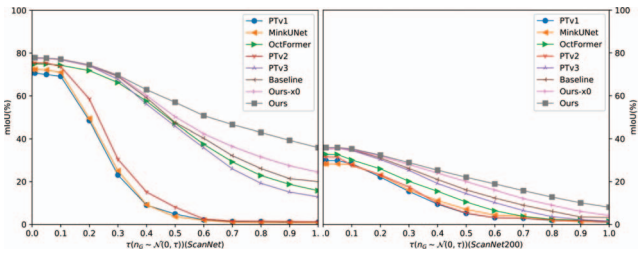

Figure 7. The noise robustness results on ScanNet and ScanNet200. Ours-$\boldsymbol{x_0}$ means the fitting target is $\boldsymbol{x_0}$ in NN of CDSegNet. Our method consistently outperforms all other methods.

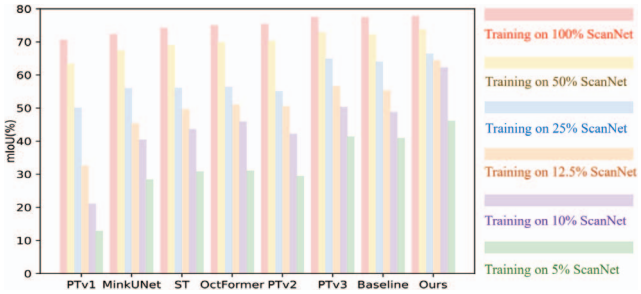

Figure 8. The sparsity robustness results on under-sampled ScanNet. Our method demonstrates the best results in all cases.

Furthermore, we investigate the robustness to noise from other distributions. Tab. 3 shows that although CDSegNet maintains strong performance across noise from other distributions, the performance is slightly reduced compared to Gaussian noise. Combined with Eq. 2, [39] provides an intuitive explanation from the perspective of the gradient of the data distribution, i.e., the predicted noise guiding $\boldsymbol{x_t} \xrightarrow{\boldsymbol{\epsilon}} \boldsymbol{x_{t-1}}$, $\boldsymbol{\epsilon} \sim p_{noise}(\boldsymbol{\epsilon}|\boldsymbol{x_t}, C)$. In this paper, we provide a simpler explanation that is consistent with the source.

According to Sec. 3.2, since DDPMs can see multi-level noise samples and fit the noise target from the modeled distribution during training, this makes that they can adapt to

Methods	mIoU	barrier	bicycle	bus	car	vehicle	moto	pedestrian	cone	trailer	truck	drivable	others	sidewalk	terrain	manmade	vegetation
RangeNet53++ [35]	65.5	66.0	21.3	77.2	80.9	30.2	66.8	69.6	52.1	54.2	72.3	94.1	66.6	63.5	70.1	83.1	79.8
PolarNet [63]	71.0	74.7	28.2	85.3	90.9	35.1	77.5	71.3	58.8	57.4	76.1	96.5	71.1	74.7	74.0	87.3	85.7
Salsanext [9]	72.2	74.8	34.1	85.9	88.4	42.2	72.4	72.2	63.1	61.3	76.5	96.0	70.8	71.2	71.5	86.7	84.4
AMVNet [29]	76.1	79.8	32.4	82.2	86.4	**62.5**	81.9	75.3	**72.3**	83.5	65.1	**97.4**	67.0	**78.8**	74.6	90.8	87.9
Cylinder3D [67]	76.1	76.4	40.3	91.2	93.8	51.3	78.0	78.9	64.9	62.1	84.4	96.8	71.6	76.4	75.4	90.5	87.4
PVKD [17]	76.0	76.2	40.0	90.2	**94.0**	50.9	77.4	78.8	64.7	62.0	84.1	96.6	71.4	76.4	76.3	90.3	86.9
RPVNet [58]	77.6	78.2	43.4	92.7	93.2	49.0	85.7	80.5	66.0	66.9	84.0	96.9	73.5	75.9	76.0	90.6	88.9
SphereFormer [25]	79.5	78.7	46.7	95.2	93.7	54.0	88.9	81.1	68.0	**74.2**	**86.2**	97.2	74.3	76.3	75.8	**91.4**	**89.7**
PTv3 [53]	80.3	**80.5**	**53.8**	95.9	91.9	52.1	88.9	**84.5**	71.7	74.1	84.5	97.2	75.6	77.0	76.2	91.2	89.6
Baseline	80.4	80.1	53.2	95.9	92.0	56.5	89.2	84.1	71.2	73.1	84.4	96.9	**76.5**	77.2	75.8	91.3	89.4
Ours	**81.2**	80.1	53.5	**97.0**	92.3	62.3	**89.7**	84.2	71.7	72.2	85.9	97.2	**76.5**	77.8	**76.9**	**91.4**	**89.7**

Table 2. The results on nuScenes. Our method significantly outperforms other methods in the term of mIoU.

related distribution noise during inference. That is, DDPMs are definitely robust to noise from the modeled distribution compared to non-DDPMs. Moreover, the closer the noise distribution is to the modeled distribution, the better the noise robustness (the Laplace distribution); otherwise, this deteriorates (the Poisson distribution).

Methods	Smalle τ (mIoU)			Big τ (mIoU)		
	τ=0.01	τ=0.05	τ=0.1	τ=0.5	τ=0.7	τ=1.0
	Gaussian Noise					
PTv2 [52]	75.5	75.4	73.7	8.7	1.5	1.2
PTv3 [53]	77.6	77.4	76.9	45.8	26.0	12.9
Ours	**77.9**	**77.7**	**77.2**	**57.0**	**46.7**	**35.9**
	Uniform Noise					
PTv2 [52]	75.5	75.5	75.2	51.2	45.7	20.6
PTv3 [53]	77.6	77.6	77.5	74.3	70.6	56.5
Ours	**77.9**	**77.9**	**77.8**	**74.8**	**70.9**	**56.8**
	Laplace Noise					
PTv2 [52]	75.4	75.2	73.9	22.4	7.4	3.2
PTv3 [53]	77.6	77.4	76.2	30.1	14.5	8.3
Ours	**77.8**	**77.6**	**76.7**	**43.0**	**26.7**	**14.3**
	Possion Noise					
PTv2 [52]	75.5	74.2	61.2	2.4	1.2	1.0
PTv3 [53]	77.6	**77.1**	**73.6**	5.7	2.8	1.5
Ours	**77.8**	76.8	73.3	**6.5**	**3.0**	**1.9**

Table 3. The results of multiple distribution noise robustness on ScanNet. Our method performs well on the Gaussian and Laplace noise (approximating the modeled distributions), but performs sightly poorly on the Poisson noise (far from the one).

5.4. Validation for Sparsity Robustness

We also conduct sparsity robustness experiments on the under-sampled ScanNet. This first randomly samples 5%, 10%, 12.5%, 25%, and 50% from the training and validation set, respectively. Subsequently, the model is trained and fitted on the under-sampled training and validation set, while performing inference on the entire validation set.

Fig. 8 illustrates the results. CDSegNet demonstrates the significant superiority on under-sampled data. As mentioned in Sec. 3.2, CNF aligns a learnable feature augmentation, mapping the features to more complex distributions, enhancing the overfitting resistance of models on less data.

5.5. Generalization for CNF

As mentioned in Sec. 4.1, CNF is a new network framework that introduces DDPMs into 3D tasks. Therefore, we further conduct the generalization experiments of CNF.

Other backbones. We first conduct the experiments for introducing CNF into MinkUNet and PTv3. We only add FFM and NN of CDSegNet to MinkUNet (using MinkUNet achieving NN) and PTv3. In Tab. 4, by introducing CNF, MinkUNet and PTv3 exhibit the better results in multiple benchmarks, due to the feature augmentation through reasonable perturbations. Moreover, benefiting from the lightweight NN, the additional inference time and memory consumption introduced by CNF in MinkUnet and PTv3 are negligible. Furthermore, we observe that PTv3 is surprisingly slower than CDSegNet on nuScenes, as PTv3 utilizes time-consuming point-level element-wise addition for skip connections, while CDSegNet uses channel concatenation.

Methods	Params	mIoU	τ=0.1	τ=0.5	τ=1.0	IT/MM
			ScanNet [10]			
MinkUNet [8]	37.9M	72.3	70.0	3.6	0.1	63s/0.9G
PTv3 [53]	46.2M	77.6	77.5	45.8	12.9	53s/1.1G
Baseline	101.4M	77.5	76.9	46.1	13.0	55s/1.9G
MinkUNet+CNF	47.1M	73.5	73.4	35.4	12.2	63s/1.0G
PTv3+CNF	59.4M	77.7	77.4	**60.1**	33.2	53s/1.1G
CDSegNet	101.4M	**77.9**	**77.6**	57.0	**35.9**	56s/1.9G
			ScanNet200 [42]			
MinkUNet [8]	37.9M	28.3	27.8	1.2	0.0	64s/0.9G
PTv3 [53]	46.2M	35.3	34.0	10.2	1.0	53s/1.1G
Baseline	101.4M	35.4	33.9	11.1	1.2	56s/1.9G
MinkUNet+CNF	47.1M	30.6	29.4	15.6	6.4	64s/0.9G
PTv3+CNF	59.4M	35.9	35.5	**21.2**	**7.0**	53s/1.1G
CDSegNet	101.4M	**36.3**	**35.7**	20.6	5.5	57s/1.9G
			nuScenes [5]			
MinkUNet [8]	37.9M	73.4	45.4	1.0	1.0	733s/1.0G
PTv3 [53]	46.2M	80.3	63.9	1.1	1.1	163s/1.0G
Baseline	101.4M	80.4	64.5	1.1	1.1	128s/1.8G
MinkUNet+CNF	47.1M	75.7	55.3	1.3	1.3	735s/1.1G
PTv3+CNF	59.4M	81.0	64.8	1.3	1.3	126s/1.1G
CDSegNet	101.4M	**81.2**	**66.2**	**3.8**	**3.8**	129s/1.8G

Table 4. The results of backbones on multiple benchmarks. 'IT' and 'MM' mean the inference time and the mean memory. We run on an NVIDIA 3090 GPU with batch size=1, without using test-time augmentation (TTA) and fragmented inference.

Methods	Params	Val	Test	Only using training data?	IT/MM
PTv3 [53]	46.2M	80.3	81.2	✓	163s/1.0G
PTv3+PPT [53]	97.3M	**81.2**	**83.0**	✗	-/-
PTv3+CNF	59.4M	81.0	82.8	✓	126s/1.1G
CDSegNet	101.4M	**81.2**	82.0	✓	129s/1.9G

Table 5. The results on the nuScenes test set. Our method shows excellent generalization ability on the nuScenes test set.

On Test set. We further evaluated on the nuScenes test set. PTv3+PPT adopts a multi-dataset joint training strategy (For a fair comparison, we avoid deliberately optimizing the results, as we cannot know how many datasets and tricks were actually used). In Fig. 5, PTv3+CNF, despite having nearly half the number of parameters and being trained only on the training set, performs slightly worse than PTv3+PPT.

Other 3D tasks. We also introduce CNF into the other 3D tasks, classification. PointNet [37] and PointNet++ [38]

are selected as backbones. We use an additional PointNet++ branch for modeling the diffusion process. Tab. 6 show that by introducing CNF, they exhibit significant improvements in classification, alleviating the sensitivity to noise.

Methods	PointNet [37]				PointNet++ [38]			
	CA	IA	τ=0.5	25%	CA	IA	τ=0.5	25%
Without CNF	86.9	90.5	2.4	19.8	90.4	92.3	2.5	20.0
With CNF	**88.1**	**91.9**	**15.6**	**25.3**	**91.3**	**93.2**	**16.1**	**28.7**

Table 6. The classification results on ModelNet40 [55]. 'CA' means the class accuracy, while 'IA' stands for the instance accuracy. Introducing CNF improves the results in classification.

5.6. Ablation Study

The diffusion modeling. We first conduct the ablation study for the diffusion modeling in CDSegNet. Tab. 7 shows the results on ScanNet. Ours-CN surprisingly experiences a significant drop across all cases. We believe that the excessive parameters make the model overfit on ScanNet, resulting in the poor generalization. In fact, reducing the parameter number will transform Ours-CN into PTv3. This demonstrates that the reasonable perturbations can significantly enhance the overfitting resistance of CN. Fig. 9(a) further supports the conclusion.

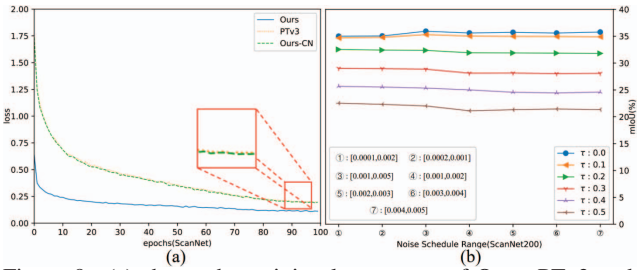

Figure 9. (a) shows the training loss curves of Ours, PTv3 and Ours-CN on ScanNet. The loss curve of Ours-CN (with more parameters compare) is nearly identical to that of PTv3, but the poorer results during inference, indicating overfitting. However, Ours demonstrates the lower loss curve and better performance, as the reasonable perturbations from NN alleviates the model overfitting, enhancing the generalization. In (b), the linear schedule noise range ③ demonstrates the best trade-off.

The noise schedule range. The noise schedule range is crucial for CDSegNet, controlling the perturbation degree. Generally, the noise schedule range is positively correlated with the robustness, while showing a negative correlation with the performance. This produces the trade-off between the segmentation performance and the noise robustness.

In Fig. 9(b), ③ yields the best trade-off on ScanNet200. Meanwhile, the cosine noise schedule range $[0, 1000]$ and the linear schedule noise range ⑤ correspond to the best results for ScanNet and nuScenes, respectively. This inspires us to choose **a smaller schedule range for the more complex (ScanNet200) or sparser (nuScenes) scenes**. Conversely, **a larger range should be selected (ScanNet)**.

The multi-task optimization. Benefiting from the dual branch architecture [20, 39], CDSegNet has two fitting objectives: the noise fitting and the semantic label fitting, coming from NN and CN, respectively. However, as mentioned in Sec. 3.3, the noise fitting typically converges more slowly, due to more fitting targets. This may cause the unreasonable noise information from NN to impair the ability of CN for understanding the scene semantics. Therefore, in addition to FFM, we also apply an effective loss balancing strategy to further alleviates the unreasonable noise perturbations (more optimizations in the supplementary material).

We consider several loss balancing strategies: 1) Equal Weighting (EW). 2) Random Loss Weighting (RLW) [27]. 3) Uncertainty Weights (UW) [22]. 4) Geometric Loss Strategy (GLS) [7]. Tab. 8 shows that GLS produces the best results. Moreover, we observed that after balancing with GLS, the loss values from CN and NN become closer and exhibit smaller fluctuations. **This can further inspire us to optimize the model with a multi-branch framework from a multi-task perspective.** Meanwhile, to achieve better performance, the loss values among multiple tasks should be more compact and stable.

Methods	Params	Performance			Robustness	
		mIoU	mAcc	allAcc	τ=0.5	25%
PTv3	46.2M	77.6	85.0	92.0	45.8	64.3
Ours-CN	88.1M	76.6	84.6	91.6	43.4	61.8
Baseline	101.4M	77.5	85.1	91.9	46.1	64.0
Ours	101.4M	**77.9**	**85.2**	**92.2**	**57.0**	**66.5**

Table 7. Ablation study of the diffusion modeling in CDSegNet on ScanNet. The baseline means removing the diffusion modeling in NN. Meanwhile, Our-CN represents removing entire NN in CDSegNet, retaining only CN. This results demonstrate the importance of the diffusion modeling for CDSegNet.

Methods	Performance			Robustness	
	mIoU	mAcc	allAcc	τ=0.5	25%
EW	77.6	85.1	92.0	55.5	64.4
RLW [27]	77.4	85.0	91.9	54.1	63.9
UW [22]	77.6	85.0	92.1	55.6	65.1
GLS [7]	**77.9**	**85.2**	**92.2**	**57.0**	**66.5**

Table 8. Ablation study of multi-task optimization on ScanNet. GLS shows the better performance and robustness for CDSegNet.

6. Conclusion

In this paper, we systematically revealed some key insights of DDPMs in point cloud semantic segmentation. Meanwhile, to preserve the advantages while overcoming the limitations, a Conditional-Noise Framework of DDPMs is designed. Based on this, we further proposed an end-to-end robust point cloud segmentation network, CDSegNet. CDSegNet demonstrated the strong robustness to the data noise and sparsity, while requiring a single-step inference. Moreover, we provided a more understandable explanation for the noise robustness from DDPMs. Overall, we have lowered the barrier for applying DDPMs, with the hope of encouraging broader extensions of DDPMs in 3D tasks.

Acknowledgments. This work was supported in part by the Frontier Technologies R&D Program of Jiangsu under grant BF2024070, in part by the National Natural Science Foundation of China under Grant 62471235, and in part by Hunan Natural Science Foundation Project (No. 2025JJ50338) and Shanghai Education Committee AI Project (No. JWAIYB-2).

References

[1] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1534–1543, 2016. 1

[2] Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34:17981–17993, 2021. 4

[3] Arpit Bansal, Eitan Borgnia, Hong-Min Chu, Jie Li, Hamid Kazemi, Furong Huang, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Cold diffusion: Inverting arbitrary image transforms without noise. *Advances in Neural Information Processing Systems*, 36, 2024. 4

[4] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4413–4421, 2018. 5

[5] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 1, 3, 5, 7

[6] Xinlei Chen, Zhuang Liu, Saining Xie, and Kaiming He. Deconstructing denoising diffusion models for self-supervised learning. *arXiv preprint arXiv:2401.14404*, 2024. 3

[7] Sumanth Chennupati, Ganesh Sistu, Senthil Yogamani, and Samir A Rawashdeh. Multinet++: Multi-stream feature aggregation and geometric loss strategy for multi-task learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 0–0, 2019. 5, 8

[8] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3075–3084, 2019. 1, 2, 6, 7

[9] Tiago Cortinhal, George Tzelepis, and Eren Erdal Aksoy. Salsanext: Fast, uncertainty-aware semantic segmentation of lidar point clouds. In *Advances in Visual Computing: 15th International Symposium, ISVC 2020, San Diego, CA, USA, October 5–7, 2020, Proceedings, Part II 15*, pages 207–222. Springer, 2020. 7

[10] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 1, 5, 6, 7

[11] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 1

[12] Siqi Fan, Qiulei Dong, Fenghua Zhu, Yisheng Lv, Peijun Ye, and Fei-Yue Wang. Scf-net: Learning spatial contextual features for large-scale point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14504–14513, 2021. 2

[13] Whye Kit Fong, Rohit Mohan, Juana Valeria Hurtado, Lubing Zhou, Holger Caesar, Oscar Beijbom, and Abhinav Valada. Panoptic nuscenes: A large-scale benchmark for lidar panoptic segmentation and tracking. *IEEE Robotics and Automation Letters*, 7(2):3795–3802, 2022. 1

[14] Meng-Hao Guo, Jun-Xiong Cai, Zheng-Ning Liu, Tai-Jiang Mu, Ralph R Martin, and Shi-Min Hu. Pct: Point cloud transformer. *Computational Visual Media*, 7:187–199, 2021. 2

[15] Yun He, Danhang Tang, Yinda Zhang, Xiangyang Xue, and Yanwei Fu. Grad-pu: Arbitrary-scale point cloud upsampling via gradient descent with learned distance functions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5354–5363, 2023. 6

[16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1, 3, 4

[17] Yuenan Hou, Xinge Zhu, Yuexin Ma, Chen Change Loy, and Yikang Li. Point-to-voxel knowledge distillation for lidar semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8479–8488, 2022. 7

[18] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11108–11117, 2020. 2

[19] Qiangui Huang, Weiyue Wang, and Ulrich Neumann. Recurrent slice networks for 3d segmentation of point clouds. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2626–2635, 2018. 2

[20] Shengyu Huang, Zan Gojcic, Mikhail Usvyatsov, Andreas Wieser, and Konrad Schindler. Predator: Registration of 3d point clouds with low overlap. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 4267–4276, 2021. 8

[21] Xiaoshui Huang, Sheng Li, Wentao Qu, Tong He, Yifan Zuo, and Wanli Ouyang. Frozen clip model is efficient point cloud backbone. *arXiv preprint arXiv:2212.04098*, 2022. 1, 3

[22] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491, 2018. 8

[23] Abderrazzaq Kharroubi, Rafika Hajji, Roland Billen, and Florent Poux. Classification and integration of massive 3d points clouds in a virtual reality (vr) environment. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42(W17), 2019. 1

[24] Xin Lai, Jianhui Liu, Li Jiang, Liwei Wang, Hengshuang Zhao, Shu Liu, Xiaojuan Qi, and Jiaya Jia. Stratified transformer for 3d point cloud segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8500–8509, 2022. 1, 2, 6

[25] Xin Lai, Yukang Chen, Fanbin Lu, Jianhui Liu, and Jiaya Jia. Spherical transformer for lidar-based 3d recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17545–17555, 2023. 7

[26] Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4558–4567, 2018. 2

[27] Baijiong Lin, Feiyang Ye, Yu Zhang, and Ivor W Tsang. Reasonable effectiveness of random weighting: A litmus test for multi-task learning. *arXiv preprint arXiv:2111.10603*, 2021. 8

[28] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 300–309, 2023. 1

[29] VE Liong, TNT Nguyen, S Widjaja, D Sharma, and ZJ Chong. Amvnet: Assertion-based multi-view fusion network for lidar semantic segmentation. arxiv 2020. *arXiv preprint arXiv:2012.04934*. 7

[30] Chang Liu, Aimin Jiang, Yibin Tang, Yanping Zhu, and Qi Chen. 3d point cloud semantic segmentation based on diffusion model. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4375–4379. IEEE, 2024. 2

[31] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35:5775–5787, 2022. 2, 3

[32] Canjie Luo, Yuanzhi Zhu, Lianwen Jin, and Yongpan Wang. Learn to augment: Joint data augmentation and network optimization for text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13746–13755, 2020. 2

[33] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2837–2845, 2021. 2, 3

[34] Zhaoyang Lyu, Zhifeng Kong, Xudong Xu, Liang Pan, and Dahua Lin. A conditional point diffusion-refinement paradigm for 3d point cloud completion. *arXiv preprint arXiv:2112.03530*, 2021. 1, 2, 3

[35] Andres Milioto, Ignacio Vizzo, Jens Behley, and Cyrill Stachniss. Rangenet++: Fast and accurate lidar semantic segmentation. In *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 4213–4220. IEEE, 2019. 7

[36] Anh Nguyen and Bac Le. 3d point cloud segmentation: A survey. In *2013 6th IEEE conference on robotics, automation and mechatronics (RAM)*, pages 225–230. IEEE, 2013. 1

[37] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 2, 7, 8

[38] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 2, 7, 8

[39] Wentao Qu, Yuantian Shao, Lingwu Meng, Xiaoshui Huang, and Liang Xiao. A conditional denoising diffusion probabilistic model for point cloud upsampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20786–20795, 2024. 1, 2, 3, 6, 8

[40] Sylvestre-Alvise Rebuffi, Sven Gowal, Dan Andrei Calian, Florian Stimberg, Olivia Wiles, and Timothy A Mann. Data augmentation can improve robustness. *Advances in Neural Information Processing Systems*, 34:29935–29948, 2021. 2

[41] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1

[42] David Rozenberszki, Or Litany, and Angela Dai. Language-grounded indoor 3d semantic segmentation in the wild. In *European Conference on Computer Vision*, pages 125–141. Springer, 2022. 1, 3, 5, 6, 7

[43] Zhenbo Shi, Zhi Chen, Zhenbo Xu, Wei Yang, Zhidong Yu, and Liusheng Huang. Shape prior guided attack: Sparser perturbations on 3d point clouds. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8277–8285, 2022. 1, 3

[44] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 1, 2, 3

[45] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 1, 3

[46] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. 3

[47] Liujie Sun, Tengfei Zeng, Jingxing Fan, and Wenju Wang. Double-view feature fusion network for lidar semantic segmentation. *Journal of Image and Graphics*, 29(1):205–217, 2024. 1

[48] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6411–6420, 2019. 2

[49] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 4

[50] Peng-Shuai Wang. Octformer: Octree-based transformers for 3d point clouds. *ACM Transactions on Graphics (TOG)*, 42(4):1–11, 2023. 1, 6

[51] Zekai Wang, Tianyu Pang, Chao Du, Min Lin, Weiwei Liu, and Shuicheng Yan. Better diffusion models further improve adversarial training. In *International Conference on Machine Learning*, pages 36246–36263. PMLR, 2023. 2, 3

[52] Xiaoyang Wu, Yixing Lao, Li Jiang, Xihui Liu, and Hengshuang Zhao. Point transformer v2: Grouped vector attention and partition-based pooling. *Advances in Neural Information Processing Systems*, 35:33330–33342, 2022. 1, 2, 4, 5, 6, 7

[53] Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler faster stronger. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4840–4851, 2024. 1, 2, 4, 5, 6, 7

[54] Yang Wu, Kaihua Zhang, Jianjun Qian, Jin Xie, and Jian Yang. Text2lidar: Text-guided lidar point cloud generation via equirectangular transformer. In *European Conference on Computer Vision*, pages 291–310. Springer, 2024. 1

[55] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015. 8

[56] Chong Xiang, Charles R Qi, and Bo Li. Generating 3d adversarial point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9136–9144, 2019. 1, 3

[57] Weilai Xiang, Hongyu Yang, Di Huang, and Yunhong Wang. Denoising diffusion autoencoders are unified self-supervised learners. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15802–15812, 2023. 3

[58] Jianyun Xu, Ruixiang Zhang, Jian Dou, Yushi Zhu, Jie Sun, and Shiliang Pu. Rpvnet: A deep and efficient range-point-voxel fusion network for lidar point cloud segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16024–16033, 2021. 7

[59] Yu-Qi Yang, Yu-Xiao Guo, Jian-Yu Xiong, Yang Liu, Hao Pan, Peng-Shuai Wang, Xin Tong, and Baining Guo. Swin3d: A pretrained transformer backbone for 3d indoor scene understanding. *arXiv preprint arXiv:2304.06906*, 2023. 2

[60] Shuquan Ye, Dongdong Chen, Songfang Han, and Jing Liao. Learning with noisy labels for robust point cloud segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6443–6452, 2021. 1

[61] Shuquan Ye, Dongdong Chen, Songfang Han, and Jing Liao. Robust point cloud segmentation with noisy annotations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(6):7696–7710, 2022. 1

[62] Xiaoqing Ye, Jiamao Li, Hexiao Huang, Liang Du, and Xiaolin Zhang. 3d recurrent neural networks with context fusion for point cloud semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 403–417, 2018. 2

[63] Yang Zhang, Zixiang Zhou, Philip David, Xiangyu Yue, Zerong Xi, Boqing Gong, and Hassan Foroosh. Polarnet: An improved grid representation for online lidar point clouds semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9601–9610, 2020. 7

[64] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16259–16268, 2021. 1, 2, 4, 6

[65] Xiao Zheng, Xiaoshui Huang, Guofeng Mei, Yuenan Hou, Zhaoyang Lyu, Bo Dai, Wanli Ouyang, and Yongshun Gong. Point cloud pre-training with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22935–22945, 2024. 2

[66] Linqi Zhou, Yilun Du, and Jiajun Wu. 3d shape generation and completion through point-voxel diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5826–5835, 2021. 1, 2

[67] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9939–9948, 2021. 7