Generative Data Augmentation for Object Point Cloud Segmentation

Dekai Zhu[1,2,3]
dekai.zhu@tum.de
Stefan Gavranovic[2]
stefan.gavranovic@siemens.com
Flavien Boussuge[2]
flavien.boussuge@siemens.com
Benjamin Busam[1]
b.busam@tum.de
Slobodan Ilic[1,2]
slobodan.ilic@tum.de

[1] Technical University of Munich
Munich, Germany
[2] Siemens AG
Munich, Germany
[3] Munich Center for Machine Learning
Munich, Germany

Abstract

Data augmentation is widely used to mitigate data scarcity in deep learning. However, traditional data augmentation (TDA) typically relies on simple geometric transformations, such as random rotation and scaling, bringing limited data diversity and marginal performance improvements. Recent advances in 3D shape generation leverage diffusion models to generate realistic point clouds for 3D content creation, but these generated shapes lack point-wise semantic labels, limiting their utility for point cloud segmentation. To bridge this gap, we extend the state-of-the-art diffusion model, Lion, into a **part-aware generative model** capable of producing high-quality point clouds conditioned on given segmentation masks. Built upon this model, we propose a **3-step generative data augmentation (GDA) pipeline** for point cloud segmentation with limited supervision. GDA enhances the training set using both **generated variants** and pseudo-labeled samples, which are validated by a novel **diffusion-based pseudo-label filtering** strategy. Extensive experiments on two large-scale synthetic datasets and a real-world medical dataset demonstrate that our GDA pipeline significantly outperforms TDA and state-of-the-art semi-/self-supervised methods.

1 Introduction

Deep learning has achieved remarkable progress across many domains [12, 47, 48, 49, 50], largely driven by the availability of large-scale annotated datasets. However, in 3D computer vision, creating semantic segmentation annotations remains costly and time-consuming. Traditional data augmentation (TDA), such as random rotation or scaling, provides limited shape diversity and thus limited performance improvements. To address annotation scarcity, recent methods leverage unlabeled data via semi-, weakly-, or self-supervised learning [3, 4, 5, 6, 11, 13, 33].

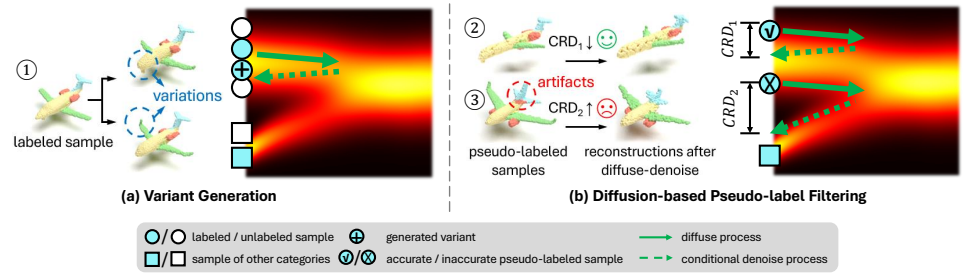


Figure 1: (a) **Variant generation** enriches the labeled data by interpolating novel samples between the existing labeled and unlabeled samples. The generated shapes exhibit diverse geometric variations. (b) **Diffusion-based pseudo-label filtering** method assesses the quality of pseudo labels based on the conditional reconstruction discrepancy (CRD), calculated by the shape deviation after the conditional diffusion-denoise process guided by the pseudo label. The sample is closely reconstructed if the pseudo label is accurate. Hence, a lower CRD indicates higher pseudo-label quality, and vice versa.

Generative models such as VAEs [14], GANs [7], and diffusion models [10] offer a promising alternative by synthesizing novel data. Recent work [39, 45] shows that diffusion-based augmentation benefits 2D tasks. However, their extension to 3D segmentation remains underexplored. Though 3D diffusion models [22, 41, 46] generate high-quality shapes, they lack part-level labels, limiting their use in segmentation training.

Therefore, this work aims to generate high-quality 3D shapes conditioned on segmentation labels, addressing the challenge of limited annotated data. We extend the state-of-the-art point cloud diffusion model Lion [41] at two levels. **(1)** At the global encoding level, we incorporate statistical semantics of parts into global features to form a richer prior. **(2)** At the local encoding level, we enhance PVCNN [20] to a **part-aware PVCNN (p-PVCNN)** that captures correlations between part labels and point-wise geometry. Based on this part-aware generative model, we propose a **3-step generative data augmentation (GDA) pipeline**: **(1)** Semi-supervised generative training of the generative model. **(2) Variant generation**, which interpolates novel labeled samples between real labeled and unlabeled data (Fig.1 (a)); **3. Diffusion-based pseudo-label filtering**, which removes inaccurate pseudo-labeled samples by measuring the **conditional reconstruction discrepancy (CRD)** (Fig.1 (b)).

We evaluate our method on two large-scale synthetic datasets, ShapeNet [38] and PartNet [24], and a real-world medical dataset, IntrA [37]. GDA consistently outperforms TDA, semi-supervised [13], and self-supervised methods [29, 30]. It generalizes well across multiple segmentation models, including PointNet [27], PointNet++ [28], Point Transformer [44], and SPoTr [26]. Furthermore, we analyze the robustness of GDA under arbitrary object orientations and study key factors affecting performance, including diffusion steps, labeled sample selection, and the quality of generated samples. In summary, the contributions of this work are the following:

- A novel part-aware generative model that generates point clouds conditioned on given segmentation masks.
- A novel 3-step GDA pipeline that generates labeled 3D shapes from a small portion of labeled 3D shapes and validates the quality of pseudo-labeled samples.
- Extensive experiments on two large-scale datasets and a challenging real-world med-

ical dataset show the state-of-the-art performance of our GDA method over TDA and related semi-supervised and self-supervised methods.

- The first systematic analysis of the performance influencing factors of 3D GDA, providing valuable insights for the broader application of 3D GDA.

2 Related Works

Learning 3D segmentation from limited data can be grouped into semi-supervised, weakly-supervised, and self-supervised methods. **(1) Semi-supervised methods** annotate only a subset of samples and use contrastive learning [34] to exploit unlabeled data. For instance, Jiang et al. [13] first train on labeled data, then generate pseudo labels for unlabeled samples, using high-confidence predictions to guide contrastive learning. **(2) Weakly-supervised methods** provide partial labels per sample, e.g., sparse point annotations. They are designed for static objects [35], indoor [11, 19, 36], and outdoor [18] scenes, propagating supervision through spatio-temporal consistency [2]. **(3) Self-supervised methods** learn from unlabeled data via reconstruction or contrastive objectives. ReCon [29] combines masked autoencoding with contrastive learning, while Point-CMAE [30] further encourages consistent representations under different masking patterns.

Diffusion-based 3D generation. DPM [22] and Point-Voxel Diffusion [46] train a diffusion model directly on point clouds, while Lion [41] achieves higher generation quality by utilizing a hierarchical VAE and latent diffusion models. More recently, [31, 40, 42, 43] focus on 3D scene generation. Due to higher semantics and geometry complexity, instead of directly generating a scene, these works rearrange indoor furniture into reasonable placements.

Generative data augmentation. While augmentation techniques based on VAEs or GANs boost performance for 3D domain adaptation [15, 16, 21], these techniques do not work equally well for GDA. Recent studies [45] demonstrate the superior performance of diffusion models in GDA. Besides, [1, 9, 32, 39] investigate how to use diffusion models for generative data augmentation on 2D images.

3 Methodology

3.1 Preliminaries

Denoising diffusion probabilistic model. The diffusion model [10] generates data by simulating a stochastic discrete process: Given a sample $x_0 \sim q(x_0)$ from a distribution, the forward process gradually adds noise to the input to make it converge to a Gaussian distribution after T steps, i.e., $q(x_T) \approx \mathcal{N}(0,I)$. The training objective on the diffusion model ε_θ with parameters θ is to predict the noise ε for the perturbed sample x_t:

$$\mathcal{L} = \mathbb{E}_{t,x_0,\varepsilon}[||\varepsilon_\theta(x_t,t,a) - \varepsilon||_2^2], \tag{1}$$

where t is the time step uniformly sampled from $\{1,...,T\}$, $\varepsilon \sim \mathcal{N}(0,I)$ is the noise for diffusing x_0 to x_t, and a is the condition information. During inference, the diffusion model starts from a random sample $x_T \sim \mathcal{N}(0,I)$ and denoises it iteratively until $t = 0$. At each step, noise $\eta \sim \mathcal{N}(0,I)$ is added to x_t to increase the randomness of x_0.

Diffuse-denoise process. Unlike the standard inference process of DDPMs that starts from Gaussian noise and iterates for T steps, the diffuse-denoise process [23, 41] diffuses an

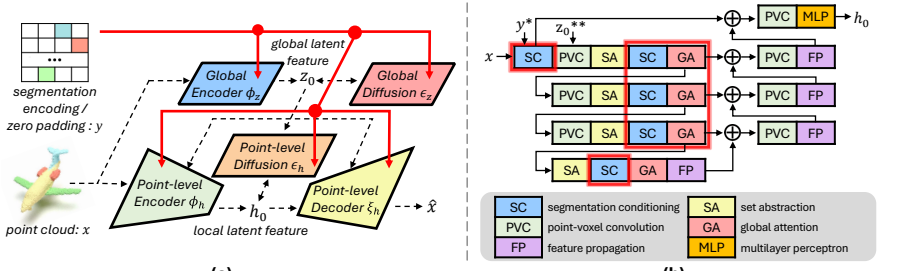


Figure 2: (a) **Architecture of the part-aware generative model.** Red solid arrows indicate the extended data flow incorporating segmentation encoding y into all modules at both global and local levels. (b) **Architecture of p-PVCNN in ϕ_h.** Red boxes highlight two extensions: (1) Segmentation conditioning (SC) modules are inserted in all layers to inject y; (2) Global attention (GA) modules are applied across all layers to enhance intra-part feature learning, especially for small parts. The decoder ξ_h and diffusion ϵ_h share the same architecture.

existing sample x_0 for τ steps ($\tau < T$) and then denoises the noisy feature x_τ backwards for same steps to obtain x'_0. Since η added at each denoising step increases the variety of x'_0, the diffuse-denoise process can be utilized to generate different variants of a given shape.

3.2 Part-aware Generative Model

Lion [41] encodes a point cloud x into a global feature z_0 and point-wise features h_0 using encoders ϕ_z and ϕ_h. Global and point-level diffusion modules, ϵ_z and ϵ_h, diffuse on z_0 and h_0, while the point-level decoder ξ_h reconstructs the point cloud based on these two latent features. However, Lion does not consider semantic part information, limiting its use in generating labeled data for segmentation. To address this, we extend Lion by conditioning all modules on segmentation labels y, making it part-aware, as shown in Figure 2 (a).

Global level encoding. The global encoder ϕ_z in Lion [41] follows PointNet [27] and uses max pooling to produce the global feature z_0, while the diffusion module ϵ_z applies stacked ResNet [8] blocks. To incorporate segmentation information y: **1.** ϕ_z takes the concatenation of x and y as input. **2.** ϵ_z takes as input the concatenation of z_t and the vector $\sigma_y \in \mathbb{R}^c$, which represents the proportion of points per part. The distribution σ_y captures semantic shape variations within a class, providing a meaningful global prior for downstream encoding. For example, in the car class, a bus has a larger roof than most other cars, while a racing car lacks a roof. More architectural details are provided in the supplementary materials.

Local level encoding. The point-level modules ϕ_h, ξ_h, and ϵ_h in Lion adopt a 4-layer PVCNN [20] backbone. Each layer consists of point-voxel convolution (PVC) modules (except the deepest one), combined with set abstraction (SA) and feature propagation (FP) [28] for hierarchical encoding. A global attention (GA) module is applied at the deepest layer for self-attention. To incorporate segmentation labels y, we introduce a **segmentation conditioning** (SC) module before each PVC block, where y is concatenated with the intermediate features. This enhances per-point geometric prediction and models inter-part relations, improving the inter-part coherence of generated shapes. We also observe that points belonging to small parts (e.g., the engine of an airplane) may be underrepresented in deeper layers due to downsampling. To address this, we extend the GA module to **all layers**, ensuring sufficient intra-part feature encoding throughout the model. The complete architecture of our extended **part-aware PVCNN** (**p-PVCNN**) is illustrated in Figure 2 (b). More details are

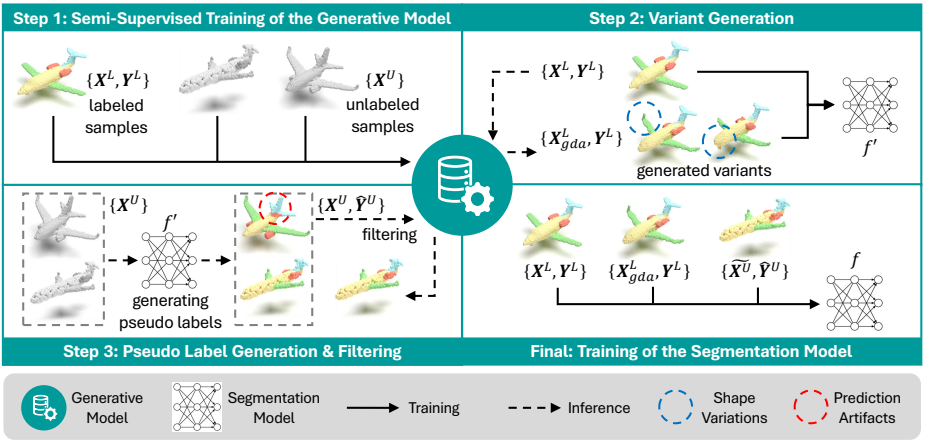


Figure 3: **Pipeline of 3-step GDA: Step 1:** Training the generative model using labeled samples $\{\mathbf{X}^L, \mathbf{Y}^L\}$ and unlabeled samples $\{\mathbf{X}^U\}$. **Step 2:** Generating variants $\mathbf{X}^L_{gda}$ via diffuse-denoise on $\mathbf{X}^L$ conditioned on $\mathbf{Y}^L$. Using $\{\mathbf{X}^L, \mathbf{Y}^L\}$ and $\{\mathbf{X}^L_{gda}, \mathbf{Y}^L\}$ to train a segmentation model f'. **Step 3:** Predicting pseudo labels $\hat{\mathbf{Y}}^U$ on $\mathbf{X}^U$ using f', then filtering low-quality samples via conditional reconstruction discrepancy. **Final:** Training the final segmentation model f using hand-labeled data and the generated samples from Steps 2 and 3.

provided in the supplementary material.

3.3 Three-step GDA Pipeline

Leveraging the part-aware generative model, we propose a 3-step GDA pipeline for the point cloud segmentation task, as illustrated in Figure 3.

Notions. Given a training set of N labeled and M unlabeled 3D point clouds ($N < M$), each consisting of n points, the goal is to train a model to generate novel shapes with point-wise segmentation labels. The labeled set contains point coordinates $\mathbf{X}^L = \{x^l_i \in \mathbb{R}^{n\times 3}\}^N_{i=1}$ and one-hot segmentation masks $\mathbf{Y}^L = \{y^l_i \in \{0,1\}^{n\times c}\}^N_{i=1}$, where c is the number of semantic parts (e.g., the car class in ShapeNet [33] has four parts: hood, roof, wheels, and body). The unlabeled set contains only coordinates $\mathbf{X}^U = \{x^u_j \in \mathbb{R}^{n\times 3}\}^M_{j=1}$, with segmentation masks padded as $\mathbf{Y}^U = \{y^u_j = \mathbf{0}^{n\times c}\}^M_{j=1}$ for unified training.

Step 1: Semi-supervised training of the generative model. Our model maps the point cloud $x \in \{\mathbf{X}^L, \mathbf{X}^U\}$ and associated labels or zero padding $y \in \{\mathbf{Y}^L, \mathbf{Y}^U\}$ into hierarchical latent feature spaces and diffuse on these latent features, composed of a global latent $z_0 \in \mathbb{R}^{d_z}$ and a point-level latent $h_0 \in \mathbb{R}^{n\times d_h}$, where d_z and d_h are the dimensions of latent spaces. The training of the generative model consists of two stages. In the first training stage, we train the hierarchical VAE, consisting of the global encoder $\phi_z : \mathbb{R}^{n\times 3} \times \mathbb{R}^{n\times c} \to \mathbb{R}^{d_z}$, point-level encoder $\phi_h : \mathbb{R}^{n\times 3} \times \mathbb{R}^{n\times c} \times \mathbb{R}^{d_z} \to \mathbb{R}^{n\times d_h}$, and point-level decoder $\xi_h : \mathbb{R}^{n\times d_h} \times \mathbb{R}^{n\times c} \times \mathbb{R}^{d_z} \to \mathbb{R}^{n\times 3}$, to maximize a variational lower bound on the data log-likelihood (ELBO):

$$\begin{aligned}\mathcal{L}(\phi_z, \phi_h, \xi_h) =& \mathbb{E}_{p(x), q_{\phi_z}, q_{\phi_h}} \{\log p_{\xi_h}(x|h_0, y, z_0) - \lambda_z D_{KL}[q_{\phi_z}(z_0|x, y)|\mathcal{N}(0, I)] \\ & - \lambda_h D_{KL}[q_{\phi_h}(h_0|x, y, z_0)|\mathcal{N}(0, I)]\},\end{aligned} \tag{2}$$

where p_{ξ_h} is the prior for reconstruction prediction, q_{ϕ_z} and q_{ϕ_h} are the posterior distribution

for sampling z_0 and h_0, and λ_z and λ_h are the hyperparameters for balancing Kullback-Leibler regularization and reconstruction accuracy. In the second training stage, we train two diffusion models $\varepsilon_z : \mathbb{R}^{d_z} \times \mathbb{R} \times \mathbb{R}^{n\times c} \to \mathbb{R}^{d_z}$ and $\varepsilon_h : \mathbb{R}^{n\times d_h} \times \mathbb{R} \times \mathbb{R}^{n\times c\times d_z} \to \mathbb{R}^{n\times d_h}$ for z_0 and h_0, respectively by minimizing the objectives:

$$\mathcal{L}(\varepsilon_z) = \mathbb{E}_{t,z_0,\varepsilon}[||\varepsilon_z(z_t,t,y) - \varepsilon||_2^2], \tag{3}$$

and

$$\mathcal{L}(\varepsilon_h) = \mathbb{E}_{t,h_0,\varepsilon}[||\varepsilon_h(h_t,t,[y,z_0]) - \varepsilon||_2^2], \tag{4}$$

where $z_0 = \phi_z(x,y)$, $h_0 = \phi_h(x,y,z_0)$, $x \in \{\mathbf{X}^L, \mathbf{X}^U\}$, $y \in \{\mathbf{Y}^L, \mathbf{Y}^U\}$. The models ε_z and ε_h are conditioned by y and $[y,z_0]$ respectively. The generative model is trained to learn the reconstruction of the point clouds with or without the segmentation labels.

Step 2: Variant generation. After training, the generative model can perform a τ-step diffuse-denoise process on $\mathbf{X}^L$. For each labeled point cloud x_i^l, we generate a set of variants $\mathbf{X}_i^l = \{x_{i,\tau}^l \in \mathbb{R}^{n\times 3} \mid 0 < \tau < T\}$ conditioned on the same segmentation label y_i^l. These variants $x_{i,\tau}^l$ preserve semantic consistency with x_i^l while exhibiting learned local deformations derived from both labeled and unlabeled data. This process effectively interpolates between labeled and unlabeled samples, producing diverse yet plausible shapes (Figure 1 (a)). Segmentation labels are naturally transferred to these new variants. We denote the generated set as $\mathbf{X}_{gda}^L = \{\mathbf{X}_i^l \mid i = 1,\ldots,N\}$. Combining $\mathbf{X}^L$ and $\mathbf{X}_{gda}^L$, we train a temporary segmentation model $f' : \mathbb{R}^{n\times 3} \to \{0,1\}^{n\times c}$ to predict pseudo labels.

Step 3: Pseudo label generation & filtering. Using f', we predict pseudo labels $\hat{\mathbf{Y}}^U = \{\hat{y}_j^u \in \{0,1\}^{n\times c} \mid j = 1,\ldots,M\}$ for the unlabeled point clouds $\mathbf{X}^U$. To mitigate the impact of inaccurate pseudo labels, we propose a diffusion-based filtering method to assess their quality. **1.** For each $(x_j^u, \hat{y}_j^u)$, we perform a τ'-step $(0 < \tau' < T)$ diffuse-denoise process with the generative model, conditioned on $\hat{y}_j^u$. This produces a reconstructed shape $\hat{x}_j^u$ from the perturbed input x_j^u, with denoising performed more deterministically (i.e., without adding noise). **2.** We define **conditional reconstruction discrepancy (CRD)** as the voxelized part-wise mIoU between x_j^u and $\hat{x}_j^u$. A low CRD (i.e., high mIoU) implies that the pseudo label $\hat{y}_j^u$ is consistent with the shape and yields a faithful reconstruction, while a high CRD indicates poor label quality. Figure 1 (b) illustrates this: in case ②, the reconstruction aligns closely with the input, while in ③, an incorrect engine label causes a large volume shift and high CRD. **3.** We discard pseudo-labeled samples with CRD above a threshold (i.e., mIoU below δ), ensuring only high-quality samples are retained for training.

Final: Training the segmentation model. After the 3-step GDA, the original labeled set $\mathbf{X}^L$ is augmented with the generated variants $\mathbf{X}_{gda}^L = \{\mathbf{X}_i^l | i = 1,\ldots,N\}$ from Step 2 and the filtered pseudo-labeled samples $\tilde{\mathbf{X}}^U \subseteq \mathbf{X}^U$ from Step 3. These combined labeled samples are used to train the final segmentation network $f : \mathbb{R}^{n\times 3} \to \{0,1\}^{n\times c}$.

4 Experiments

4.1 Experimental Settings

Datasets. We evaluate our method on two large synthetic datasets, ShapeNet [58] and PartNet [24], and a real-world dataset, IntrA [57], which contains 3D intracranial aneurysm point clouds reconstructed from MRI. We follow the official train/val/test splits of ShapeNet and PartNet. From ShapeNet, we select the categories airplane, car, lamp, motorbike, and pistol;

Table 1: Evaluation on ShapeNet [38], PartNet [24], and IntrA [37]. The **best** and second-best scores for each category are highlighted in **bold** and underlined, respectively.

Dataset	ShapeNet					PartNet					IntrA
Category	Airplane	Car	Lamp	Motorbike	Pistol	Bed	Bottle	Chair	Faucet	Table	Aneurysm
FS	80.40	76.34	72.04	67.43	82.52	38.29	52.91	60.66	53.19	58.07	-
w/o aug.	76.46	70.09	66.04	56.10	77.66	27.36	40.99	51.69	40.01	46.28	37.49
only TDA	76.84	71.48	66.71	59.03	78.91	28.19	43.06	53.25	46.20	47.89	42.29
CL [43]	76.49	71.69	66.78	59.64	79.10	29.00	47.49	53.44	47.19	50.36	43.63
ReCon [29]	77.12	74.51	66.89	65.45	80.93	30.88	49.37	58.65	47.59	51.02	47.37
Point-CMAE [30]	77.46	74.45	68.23	**65.57**	81.02	31.53	49.54	**59.23**	47.26	**51.17**	47.21
GDA (VG)	77.96	72.90	66.79	61.44	80.01	30.69	50.14	56.40	49.25	50.22	46.85
GDA (VG+FP)	**78.32**	**74.89**	**71.22**	64.67	**81.35**	**32.96**	**53.21**	58.72	**50.31**	50.42	**48.67**

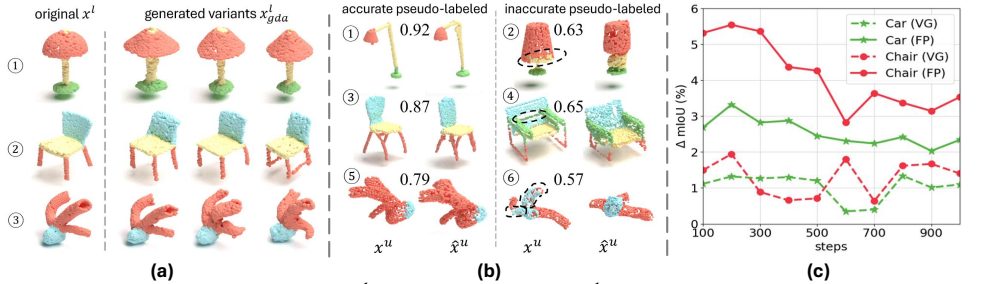


Figure 4: (a) Original samples x^l and their variants x^l_{gda}. Note: ③ is an aneurysm segment with vessels and aneurysm. (b) Pseudo-labeled samples x^u and reconstructions $\hat{x}^u$ after diffuse-denoise conditioned on $\hat{y}^u$. Accurate pseudo labels yield faithful reconstructions, while inaccurate ones result in large deviations, especially in misclassified parts (highlighted by dashed circles). Numbers denote voxelized IoU between x^u and $\hat{x}^u$, where higher IoU indicates lower CRD and better pseudo label quality. (c) Impact of diffusion steps on variant generation and pseudo-label filtering. The y-axis shows mIoU improvement over TDA.

from PartNet, we select bed, bottle, chair, faucet, and table. During training, 10% of samples in each category are labeled, and the rest are unlabeled. The IntrA dataset includes 116 labeled and 215 unlabeled aneurysm segments; we use 24 labeled and 215 unlabeled samples for training, with the remaining 92 labeled samples for testing.

Segmentation network. We primarily use PointNet [27] as the segmentation model. In the ablation studies, we additionally test PointNet++ [28], Point Transformer [44], and SPoTr [26], which is a state-of-the-art and open-source model on the ShapeNet segmentation benchmark [25].

Methods. We evaluate the performance of the following methods in the experiments:

1. Full supervision (**FS**): using the whole training set;

2. Without data augmentation (**w/o aug.**): only using 10% labeled samples;

3. Only TDA: using 10% labeled samples and TDA methods, including random rescaling (0.8, 1.2), random transfer (-0.1, 0.1), jittering (-0.005, 0.005), and random flipping;

4. CL: using 10% labeled and 90% unlabeled data with the contrastive learning method [43];

5. ReCon: pre-training on the whole training set without labels and fine-tuning on 10% labeled samples with the self-supervised method proposed in [29];

6. Point-CMAE: pre-training on the whole training set without labels and fine-tuning on 10% labeled samples with the self-supervised method proposed in [30];

7. GDA (VG): using 10% labeled data and the generated variants (VG) $\mathbf{X}^L_{gda}$;

Table 2: GDA for PointNet++ [28], Point Transformer [44], and SPoTr [26].

Backbone	Method	Airplane	Car	Lamp	Motorbike	Pistol
PointNet++ [28]	only TDA	78.10	71.52	72.64	57.88	73.24
	GDA (VG+FP)	79.92	74.66	81.15	**65.27**	79.53
PT [44]	only TDA	79.15	71.63	75.67	53.01	75.44
	GDA (VG+FP)	81.06	75.25	**81.42**	64.73	81.07
SPoTr [26]	only TDA	79.36	71.00	80.78	46.25	75.71
	GDA (VG+FP)	**81.25**	**75.73**	81.31	65.04	**81.99**

Table 3: Evaluation on ShapeNet [38] (arbitrary orientation).

Method	Airplane	Car	Lamp	Motorbike	Pistol
FS	71.98	72.71	57.53	60.59	76.99
w/o aug.	20.49	17.84	22.35	16.53	31.54
only TDA	61.12	31.09	37.56	48.38	43.82
Point-CMAE [30]	68.23	62.97	55.49	**59.37**	70.02
GDA (VG+FP)	**70.47**	**67.43**	**57.76**	58.14	**71.98**

8. GDA (VG+FP): using 10% labeled samples, $\mathbf{X}^L_{gda}$, and the filtered pseudo-labeled samples (FP) $\tilde{\mathbf{X}^U}$. Note that TDA is used in **all** methods above except method 2.

Experimental Details. Our generative model is trained in two stages: the VAE modules are trained for 8k epochs, followed by 24k epochs for the latent diffusion modules. Both stages use the Adam optimizer with a learning rate of 1e-3. The VAE and diffusion modules contain 22.3M and 88.9M parameters, respectively. For pseudo-label filtering, we set the mIoU threshold δ to 0.7. Segmentation models [26, 27, 28, 44] are trained using their official default settings. All experiments are conducted on an RTX 3090 GPU.

4.2 Main Results

We report the experimental results of mIoU scores on ShapeNet [38], PartNet [24], and IntrA [37] in Table 1. Across all these datasets, GDA with variant generation and filtered pseudo labels (VG+FP) achieves the best performance in most categories. On average, GDA outperforms methods using only TDA and Point-CMAE [30] by **3.50%** and **0.74%** on ShapeNet, **5.41%** and **1.38%** on PartNet, **6.38%** and **1.46%** on IntrA dataset. Notably, PointNet trained with 10% hand-labeled data and GDA even surpasses the model trained with full supervision on the bottle class in PartNet. To prove the stability of GDA despite the stochasticity of DDPM, we repeat the experiment on car class 10 times. The results of using only TDA and GDA are **71.49±0.27** and **74.92±0.46**.

Figure 4 (a) shows the generated variants x^l_{gda} alongside their original shapes x^l. These variants exhibit realistic and diverse local deformations, such as variations in lamp shades (①), chair legs (②), and healthy vessels (③), enhancing data diversity beyond simple geometric augmentations. Figure 4 (b) visualizes the deviation between x^u and $\hat{x}^u$ after the diffuse-denoise process conditioned on pseudo labels $\hat{y}^u$. Samples (①, ③, ⑤) with accurate pseudo labels are well reconstructed, achieving high voxelized IoU scores and low CRD, indicating high pseudo-label quality. In contrast, samples (②, ④, ⑥) with misclassified parts exhibit substantial shape deviations, resulting in low IoU and high CRD. Additional examples of generated variants and reconstructions are provided in the supplementary material.

4.3 Ablation Studies

Impacts of Diffusion-based pseudo-label filtering. We use the generative model as a filter to remove samples with inaccurate pseudo labels. In this experiment, we compare our approach with other filtering methods, including: 1. Using all pseudo labels without any filtering, and 2. PseudoAugment [17], which filters out pseudo labels with low confidence scores. We conduct the experiments on the car class. The results of the above two methods are **73.13** and **73.46**, respectively, both of which are lower than the result of GDA (**74.89**).

Impacts of the number of diffusion steps. In variant generation and pseudo-label filtering, the model runs τ and τ' diffusion steps, respectively. More steps increase sample diversity

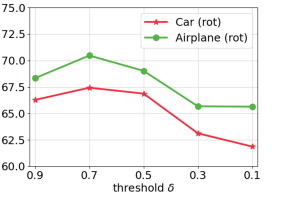
Table 4: Impact of the quality of original labeled data (C: label quality is checked).

Selection	mIoU
only TDA (C)	31.09
only TDA (w/o C)	27.53
GDA (C)	67.43
GDA (w/o C)	59.27

Table 5: Impact of the quality of generated labeled data.

Level	mIoU
only TDA	43.26
GDA-L1	50.72
GDA-L1,2	54.89
GDA-L1,2,3	54.96

Figure 5: Impacts of CRD threshold δ.



but may harm shape quality if the model is undertrained. We vary τ from 100 to 1000 on car and chair categories, and observe consistent gains across all settings (Figure 4 (c)). Thus, we use all generated variants from different τ in our experiments. For τ', both categories perform best at 200 steps, which we adopt in all pseudo-label filtering experiments.

GDA for various segmentation models. In this experiment, we test the performance of GDA using PointNet++ [28], Point Transformer [44], and SPoTr [26] on ShapeNet. The experimental results listed in Table 2 show that GDA steadily enhances the performance of these models, including the SOTA [25] model, SPoTr [26].

Impacts of CRD threshold δ. In step 3 of GDA, we filter out the pseudo-labeled samples whose voxelized mIoU between x^u and $\hat{x}^u$ falls below the threshold δ. To assess the impact of different δ values on GDA performance, we conduct experiments and present the results in Figure 5. Both the airplane and car categories (arbitrary orientation) achieve the best performance at $\delta = 0.7$. Hence, we set δ to 0.7 in all experiments.

Objects in arbitrary orientations. Objects in ShapeNet are presented in canonical poses, while maintaining the 3D object pose consistency in real world is challenging. To test the robustness of our GDA method against arbitrarily oriented objects, we randomly rotate the objects in ShapeNet [38] and repeat the experiments. The results listed in Table 3 show that GDA exhibits a larger advantage over TDA and Point-CMAE [30] in this challenging case, outperforming them by **20.76%** and **1.94%** respectively on average across categories.

Quality of hand-labeled and generated data. We observe that ShapeNet contains some annotation errors. To ensure data quality, we manually filter them when constructing $\mathbf{X}^L$. To test GDA's robustness to label noise, we re-train it on arbitrarily oriented cars using 10% labeled data without filtering. As shown in Table 4, GDA still outperforms TDA but with a larger performance drop. More discussions are provided in the supplementary material.

We also evaluate the quality of generated samples $\mathbf{X}^L_{gda}$. Since no ground truth is available, we rate 100 generated samples per category (airplane, car, lamp) into three levels: (1) L1: high-quality, (2) L2: minor label or shape errors, and (3) L3: low-quality with distortions. The L1/L2/L3 distributions are 49/46/5, 36/50/14, and 44/49/7, respectively. Examples are shown in the supplementary. We train segmentation models using subsets of L1, L1+2, and L1+2+3 samples. Table 5 shows that L1 and L2 samples improve performance, and the small proportion of L3 samples has negligible negative impact.

5 Conclusion

We propose a novel part-aware generative model to produce high-quality labeled point clouds from segmentation masks. Leveraging this model, we introduce the first diffusion-based GDA framework for 3D segmentation, which enhances training with diverse 3D variants from limited labeled data and validates pseudo labels using a novel diffusion-based pseudo-label filtering method. Experiments on three datasets and various segmentation models

demonstrate the robustness of GDA and its superiority over TDA and related semi-/self-supervised methods. As a pioneering work in 3D GDA, this work highlights the potential of GDA in reducing manual labeling efforts and improving segmentation performance.

References

[1] Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J Fleet. Synthetic data from diffusion models improves imagenet classification. *arXiv preprint arXiv:2304.08466*, 2023.

[2] Lennart Bastian, Daniel Derkacz-Bogner, Tony D Wang, Benjamin Busam, and Nassir Navab. Segmentor: Obtaining efficient operating room semantics through temporal propagation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 57–67. Springer, 2023.

[3] Ye Chen, Jinxian Liu, Bingbing Ni, Hang Wang, Jiancheng Yang, Ning Liu, Teng Li, and Qi Tian. Shape self-correction for unsupervised point cloud understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8382–8391, 2021.

[4] Zhiqin Chen, Kangxue Yin, Matthew Fisher, Siddhartha Chaudhuri, and Hao Zhang. Bae-net: Branched autoencoder for shape co-segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8490–8499, 2019.

[5] Mingmei Cheng, Le Hui, Jin Xie, and Jian Yang. Sspc-net: Semi-supervised semantic 3d point cloud segmentation network. In *Proceedings of the AAAI conference on artificial intelligence*, pages 1140–1147, 2021.

[6] Yan Di, Chenyangguang Zhang, Chaowei Wang, Ruida Zhang, Guangyao Zhai, Yanyan Li, Bowen Fu, Xiangyang Ji, and Shan Gao. Shapemaker: Self-supervised joint shape canonicalization, segmentation, retrieval and deformation. *arXiv preprint arXiv:2311.11106*, 2023.

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[9] Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. Is synthetic data from generative models ready for image recognition? *arXiv preprint arXiv:2210.07574*, 2022.

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

[11] Qingyong Hu, Bo Yang, Guangchi Fang, Yulan Guo, Aleš Leonardis, Niki Trigoni, and Andrew Markham. Sqn: Weakly-supervised semantic segmentation of large-scale 3d point clouds. In *European Conference on Computer Vision*, pages 600–619. Springer, 2022.

[12] Dongshuo Huang, Xiaoshui Huang, Chengdong Zhang, and Yilei Shi. Lpcg: A self-conditional architecture for labeled point cloud generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 3635–3643, 2025.

[13] Li Jiang, Shaoshuai Shi, Zhuotao Tian, Xin Lai, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Guided point contrastive learning for semi-supervised point cloud semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6423–6432, 2021.

[14] Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. *Advances in neural information processing systems*, 27, 2014.

[15] Alexander Lehner, Stefano Gasperini, Alvaro Marcos-Ramiro, Michael Schmidt, Mohammad-Ali Nikouei Mahani, Nassir Navab, Benjamin Busam, and Federico Tombari. 3d-vfield: Adversarial augmentation of point clouds for domain generalization in 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17295–17304, 2022.

[16] Alexander Lehner, Stefano Gasperini, Alvaro Marcos-Ramiro, Michael Schmidt, Nassir Navab, Benjamin Busam, and Federico Tombari. 3d adversarial augmentations for robust out-of-domain predictions. *International Journal of Computer Vision*, 132(3): 931–963, 2024.

[17] Zhaoqi Leng, Shuyang Cheng, Benjamin Caine, Weiyue Wang, Xiao Zhang, Jonathon Shlens, Mingxing Tan, and Dragomir Anguelov. Pseudoaugment: Learning to use unlabeled data for data augmentation in point clouds. In *ECCV*, pages 555–572. Springer, 2022.

[18] Minghua Liu, Yin Zhou, Charles R Qi, Boqing Gong, Hao Su, and Dragomir Anguelov. Less: Label-efficient semantic segmentation for lidar point clouds. In *European Conference on Computer Vision*, pages 70–89. Springer, 2022.

[19] Zhengzhe Liu, Xiaojuan Qi, and Chi-Wing Fu. One thing one click: A self-training approach for weakly supervised 3d semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1726–1736, 2021.

[20] Zhijian Liu, Haotian Tang, Yujun Lin, and Song Han. Point-voxel cnn for efficient 3d deep learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[21] Adrian Lopez-Rodriguez, Benjamin Busam, and Krystian Mikolajczyk. Project to adapt: Domain adaptation for depth completion from noisy and sparse sensor data. In *Proceedings of the Asian Conference on Computer Vision*, 2020.

[22] 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.

[23] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022.

[24] Kaichun Mo, Shilin Zhu, Angel X Chang, Li Yi, Subarna Tripathi, Leonidas J Guibas, and Hao Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 909–918, 2019.

[25] Papers with Code. 3d part segmentation on shapenet-part. https://paperswithcode.com/sota/3d-part-segmentation-on-shapenet-part, 2025. URL https://paperswithcode.com/sota/3d-part-segmentation-on-shapenet-part. Accessed: 2025-03-06.

[26] Jinyoung Park, S. Lee, Si Hyeon Kim, Yunyang Xiong, and Hyunwoo J. Kim. Self-positioning point-based transformer for point cloud understanding. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21814–21823, 2023. URL https://api.semanticscholar.org/CorpusID:257804915.

[27] C. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 77–85, 2016. URL https://api.semanticscholar.org/CorpusID:5115938.

[28] C. Qi, L. Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Neural Information Processing Systems*, 2017. URL https://api.semanticscholar.org/CorpusID:1745976.

[29] Zekun Qi, Runpei Dong, Guofan Fan, Zheng Ge, Xiangyu Zhang, Kaisheng Ma, and Li Yi. Contrast with reconstruct: Contrastive 3d representation learning guided by generative pretraining. In *International Conference on Machine Learning*, pages 28223–28243. PMLR, 2023.

[30] Bin Ren, Guofeng Mei, Danda Pani Paudel, Weijie Wang, Yawei Li, Mengyuan Liu, Rita Cucchiara, Luc Van Gool, and Nicu Sebe. Bringing masked autoencoders explicit contrastive properties for point cloud self-supervised learning. In *Proceedings of the Asian Conference on Computer Vision*, pages 2034–2052, 2024.

[31] Jiapeng Tang, Yinyu Nie, Lev Markhasin, Angela Dai, Justus Thies, and Matthias Nießner. Diffuscene: Scene graph denoising diffusion probabilistic model for generative indoor scene synthesis. *arXiv preprint arXiv:2303.14207*, 2023.

[32] Brandon Trabucco, Kyle Doherty, Max Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models. *arXiv preprint arXiv:2302.07944*, 2023.

[33] Lingjing Wang, Xiang Li, and Yi Fang. Few-shot learning of part-specific probability space for 3d shape segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4504–4513, 2020.

[34] Saining Xie, Jiatao Gu, Demi Guo, Charles R Qi, Leonidas Guibas, and Or Litany. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 574–591. Springer, 2020.

[35] Xun Xu and Gim Hee Lee. Weakly supervised semantic point cloud segmentation: Towards 10x fewer labels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13706–13715, 2020.

[36] Cheng-Kun Yang, Ji-Jia Wu, Kai-Syun Chen, Yung-Yu Chuang, and Yen-Yu Lin. An mil-derived transformer for weakly supervised point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11830–11839, 2022.

[37] Xi Yang, Ding Xia, Taichi Kin, and Takeo Igarashi. Intra: 3d intracranial aneurysm dataset for deep learning. In *CVPR*, pages 2656–2666, 2020.

[38] L. Yi, Vladimir G. Kim, Duygu Ceylan, I-Chao Shen, Mengyan Yan, Hao Su, Cewu Lu, Qi-Xing Huang, Alla Sheffer, and Leonidas J. Guibas. A scalable active framework for region annotation in 3d shape collections. *ACM Transactions on Graphics (TOG)*, 35:1 – 12, 2016. URL https://api.semanticscholar.org/CorpusID:2880712.

[39] Zebin You, Yong Zhong, Fan Bao, Jiacheng Sun, Chongxuan Li, and Jun Zhu. Diffusion models and semi-supervised learners benefit mutually with few labels. In *Proc. NeurIPS*, 2023.

[40] Sihao Yu, Fei Sun, Jiafeng Guo, Ruqing Zhang, and Xueqi Cheng. Legonet: A fast and exact unlearning architecture. *arXiv preprint arXiv:2210.16023*, 2022.

[41] Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. Lion: Latent point diffusion models for 3d shape generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[42] Guangyao Zhai, Evin Pinar Örnek, Dave Zhenyu Chen, Ruotong Liao, Yan Di, Nassir Navab, Federico Tombari, and Benjamin Busam. Echoscene: Indoor scene generation via information echo over scene graph diffusion. In *European Conference on Computer Vision*, 2024.

[43] Guangyao Zhai, Evin Pınar Örnek, Shun-Cheng Wu, Yan Di, Federico Tombari, Nassir Navab, and Benjamin Busam. Commonscenes: Generating commonsense 3d indoor scenes with scene graphs. *Advances in Neural Information Processing Systems*, 36, 2024.

[44] 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.

[45] Chenyu Zheng, Guoqiang Wu, and Chongxuan Li. Toward understanding generative data augmentation. *Advances in Neural Information Processing Systems*, 36, 2024.

[46] 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.

[47] Dekai Zhu, Guangyao Zhai, Yan Di, Fabian Manhardt, Hendrik Berkemeyer, Tuan Tran, Nassir Navab, Federico Tombari, and Benjamin Busam. Ipcc-tp: Utilizing incremental pearson correlation coefficient for joint multi-agent trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5507–5516, 2023.

[48] Dekai Zhu, Qadeer Khan, and Daniel Cremers. Multi-vehicle trajectory prediction and control at intersections using state and intention information. *Neurocomputing*, 574: 127220, 2024.

[49] Dekai Zhu, Yan Di, Stefan Gavranovic, and Slobodan Ilic. Sealion: Semantic part-aware latent point diffusion models for 3d generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 11789–11798, 2025.

[50] Dekai Zhu, Yixuan Hu, Youquan Liu, Dongyue Lu, Lingdong Kong, and Slobodan Ilic. Spiral: Semantic-aware progressive lidar scene generation. *arXiv preprint arXiv:2505.22643*, 2025.