CDPNet: Cross-Modal Dual Phases Network for Point Cloud Completion

Zhenjiang Du[1], Jiale Dou[2], Zhitao Liu[1], Jiwei Wei[1], Guan Wang[3], Ning Xie[1]*, Yang Yang[1]

[1]University of Electronic Science and Technology of China, Chengdu, China
[2] Yibin Park, University of Electronic Science and Technology of China, Yibin, China
[3]Yangtze Delta Region Institute (Huzhou), University of Electronic Science and Technology of China, Huzhou, China
{zhenjiang, doujiale}@std.uestc.edu.cn, {zl425uestc, mathematic6, wangguan12621, seanxiening, dlyyang}@gmail.com

Abstract

Point cloud completion aims at completing shapes from their partial. Most existing methods utilized shape's priors information for point cloud completion, such as inputting the partial and getting the complete one through an encoder-decoder deep learning structure. However, it is very often to easily cause the loss of information in the generation process because of the invisibility of missing areas. Unlike most existing methods directly inferring the missing points using shape priors, we address it as a cross-modality task. We propose a new Cross-modal Dual Phases Network (CDPNet) for shape completion. Our key idea is that the global information of the shape is obtained from the extra single-view image, and the partial point clouds provide the geometric information. After that, the multi-modal features jointly guide the specific structural information. To learn the geometric details of the shape, we chose to use patches to preserve the local geometric feature. In this way, we can generate shapes with enough geometric details. Experimental results show that our method achieves state-of-the-art performance on point cloud completion.

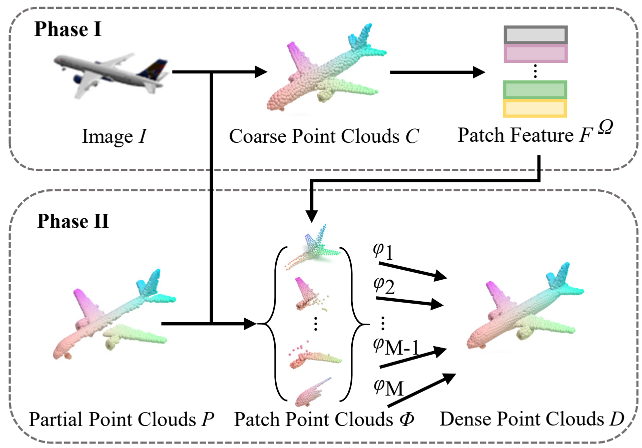

Figure 1: In phase I, we reconstruct a coarse point cloud from images and segment this point cloud into patches. In phase II, we combine the image information with the partial point cloud information to guide the generation of dense point clouds on the patches. M means the number of patches.

Introduction

Point cloud has become a popular research topic in autonomous driving, robotics, and remote sensing due to its numerous practical applications. Nevertheless, in practical implementations, the acquisition of point cloud data through 3D scanning devices may be adversely influenced by a multitude of factors, including occlusion, inadequate illumination, and suboptimal sensor resolution, which may lead to the incompleteness of point cloud (Wen et al. 2020). Consequently, point cloud completion has surfaced as an imperative research domain within 3D computer vision and computer graphics (Guo et al. 2020).

The point cloud completion task is to estimate the complete 3D point clouds based on the partial. Recent researches (Yuan et al. 2018; Groueix et al. 2018; Tchapmi et al. 2019; Xie et al. 2021) for point cloud completion successfully utilized deep-learning methods and achieved more plausible and flexible results compared with traditional geometric-based methods (Thrun and Wegbreit 2005; Pauly et al. 2005; Shao et al. 2012; Harary, Tal, and Grinspun 2014). The most commonly used approach in deep-learning methods for point cloud completion is the encoder-decoder framework. This framework involves extracting a latent code from the global partial point clouds and decoding the latent code to generate the complete point clouds. Encoding global shape representation often suffers from the information loss of some structure details on local regions of incomplete point clouds, which should be fully preserved for further inferring the missing geometric information (Wen et al. 2021). Remarkably, the human cognitive system is adept at translating visual constructs derived from 2D images into a comprehensive understanding of 3D objects and can successfully infer the shape of partial 3D objects based on 2D experiences (Aiello, Valsesia, and Magli 2022). This insight lends credence to the notion that the process of point cloud completion could benefit from integrating 2D images, thereby furnishing a more holistic depiction of the 3D shape.

In this paper, we propose a method to complete point clouds by leveraging partial point clouds and the single-view

*Ning Xie is the corresponding author.

image. Two problems must be solved to finish the cross-modal point cloud completion task. One is how to utilize the multi-modal features to do the shape completion. The other is how to generate the details of the shape.

To address these problems, we propose a novel network called Cross-Modal Dual Phases Network (CDPNet), which exploits the dual phases network to process multi-modal information to jointly guide the generation of complete point cloud shapes. We get the global information by single-view reconstruction. Besides, to obtain the local geometric information, we adopt a divide-and-conquer strategy to generate local details. Moreover, we utilize cross-modal fused information to guide the generation of shape global structural details (shown in Figure 1). Specially, our network has designed two phases (shown in Figure 2). In phase I, we extract global information from the image by the image encoder. After extraction, we use this global information to reconstruct the coarse point clouds. In order to generate enough geometric details, we segment the coarse point clouds into patches and generate dense point clouds based on patches. In phase II, we extract the fine-grained geometric information from the partial point clouds by DGCNN (Wang et al. 2019). Next, we combine the coarse local geometric information with the fine-grained geometric information and send it into the multi-patch generator (each patch corresponds to a patch generator) we designed to generate the fine-grained patch point clouds. In order to preserve the global structure, we fuse the cross-modal feature and utilize the feature to guide the multi-patch generator to preserve the shape's structural details while generating fine-grained patches. We concatenate all fine-grained patches to get the dense point clouds. In order to ensure the structural consistency between the generated point clouds and shape prior, we follow MSN (Liu et al. 2020) to merge the partial point clouds to our generated point clouds and obtain the final fine point clouds after down-sampling. This approach not only retains the original point clouds but also incorporates auxiliary information from other modalities to guide the generation of shape details.

Our main contributions are as follows:

- We propose a new CDPNet network for point cloud completion basis multi-modal data. Our network utilizes the image to learn global information and utilizes patches to preserve the local geometric details.
- To adapt patch learning, we design a new patch generator that receives coarse patch features and fine-grained geometric information to generate fine-grained patches.
- We propose utilizing the cross-modal feature fusion module to promote the global structural generation of the shape. Experimental results show that CDPNet outperforms previous methods.

Related Works

Point-based Shape Completion. Recently, with the development of deep learning, there have been a lot of successes in various fields (Chen et al. 2016; Wang et al. 2022, 2023b; Yussif et al. 2023; Wang et al. 2023a). Therefore, applying deep learning methods to 3D has become a research hotspot. Point clouds are a set of unordered points in the 3D coordinate system that represent the 3D shape (Öngün and Temizel 2021; Du et al. 2023). Qi et al. (Qi et al. 2017a,b) propose PointNet and PointNet++ provide an end-to-end solution to extract global and local features of the point cloud to analyze shapes. Yang et al. (Yang et al. 2018) propose a folding mechanism based on point clouds for shape completion. Yuan et al. (Yuan et al. 2018) propose PCN, which is based on encoder-decoder architecture and solves the problem of point cloud completion. Relying solely on global features to reconstruct complete point clouds may result in losing local information (Zhang et al. 2023). To solve the problem, a series of methods have been proposed. Tchapmi et al. (Tchapmi et al. 2019) propose a hierarchical tree structure to generate the structured point clouds. Yu et al. (Yu et al. 2021) propose a transformer encoder-decoder architecture, which can learn the local information and generate the geometric details of the shape. However, these previous works only consider single-modal shape priors to finish the completion, which may cause the loss of information.

Cross-modal-based Shape Completion. Reconstructing shapes from single-view images has been a research hotspot and has achieved promising results (Pan et al. 2019; Nguyen et al. 2019; Li et al. 2020; Xue et al. 2022; Wen et al. 2022). Inspired by these methods, the use of cross-modal data to improve point cloud completion has been explored. Recently, Zhang et al. (Zhang et al. 2021) first proposes a method of cross-modal in the point cloud completion task. The bottleneck of this method is that the fusion of information through reconstruction techniques by estimating a rough point cloud from the image cannot fully utilize the cross-modal information. Aiello et al. (Aiello, Valsesia, and Magli 2022) propose a method that utilizes the attention mechanism to fuse multi-modal features in a latent domain. This approach may perform well in multi-modal feature fusion and alleviate the issue of matching features at different levels. However, incorporating multiple attention mechanisms can increase the learning difficulty, and this method may not always generate details of the 3D shape effectively. Our method differs from the previous methods in that we utilize image priors to provide global information and shape priors to provide geometric information. We maintain the local geometric details through patching and fuse the cross-modal features to preserve the global structural details.

Method

We formulate our cross-modal problem as generating complete point clouds based on partial point clouds and the corresponding single-view image. That is, given the partial point clouds $P \in \mathbb{R}^{N^P \times 3}$ and the single-view image $I \in \mathbb{R}^{H \times W \times 3}$, our goal is to generate the dense complete point clouds $D \in \mathbb{R}^{N^D \times 3}$. N^P and N^D denote the numbers of points in P and D. H and W denote the pixels of I. To achieve our goal, we propose a network called CDPNet. Our method can deal with the problem of cross-modal feature fusion well. Besides, it also can generate a complete point cloud with details.

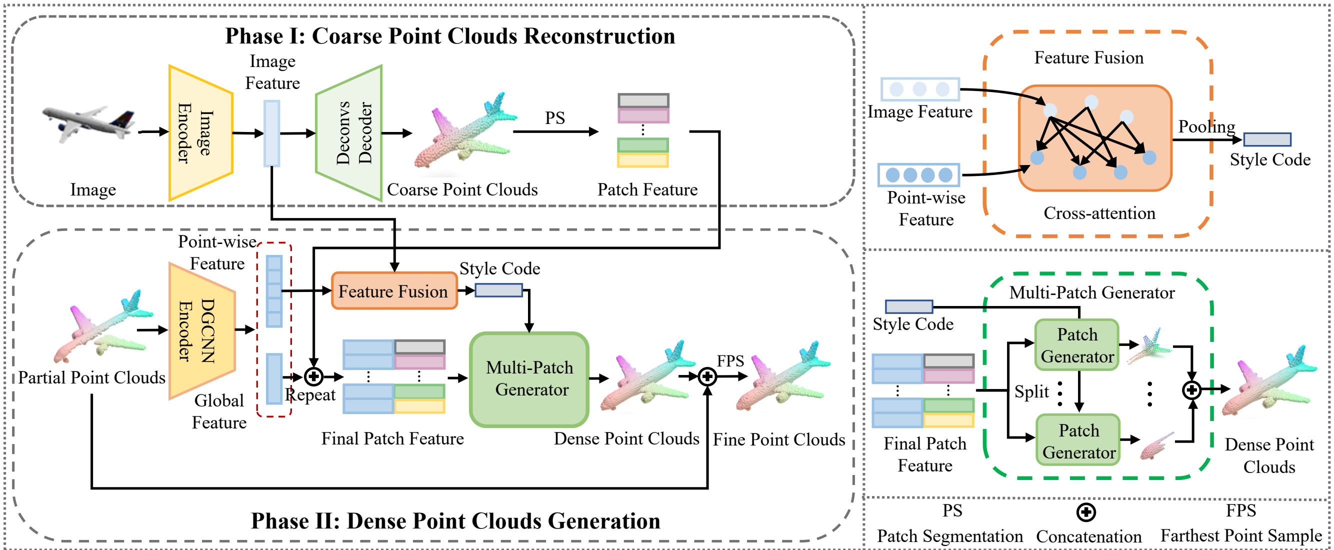

Figure 2: The architecture of CDPNet is on the left. Phase I processes the image to the coarse point clouds and performs patch segmentation on the point clouds to get M patch features. Phase II uses the patch features and the geometric feature provided by the partial point clouds to generate the dense point clouds on the patch. At the same time, the future fusion module fuses cross-modal features and is used to generate the shape's global structure details. We show the structure of (a) the Feature Fusion module and (b) the Multi-Patch Generator module on the right.

Dual Phases Architecture

Our method is divided into two phases (shown in Figure 2), the input of the phase I is the single-view image I, and the input of the second phase is the partial point clouds P. The single-view image's feature F^I is extracted through the convolutional neural network, and then deconvolution is used to generate the complete coarse point clouds C by F^I. After that, passing through patch segmentation, the coarse point clouds will be segmented into M patches. In phase II, we will feed the partial point clouds P into the DGCNN (Wang et al. 2019) to extract the point cloud's fine-grained geometric features F^P. Afterward, the extracted features F^P serve two purposes. First, after pooling, the extracted local geometric feature will be fused with the patch features of phase I and sent to the multi-patch generator we designed to generate the fine-grained patch point clouds with more geometric details. In addition, to fully use the advantages of cross-modality, we fuse the point cloud features F^P with the global image features F^I, and the fused cross-modal features to preserve the global structure details. So, we get the fine-grained patch point clouds $\Phi = \{\phi_i\}_{i=1}^{M}$ and concatenate them to get the dense point cloud D. After that, we merge the partial point clouds with the dense point clouds and do the farthest point sample (Eldar et al. 1997) to obtain the final fine point clouds.

As described above, we first reconstruct the coarse point clouds C from the image in phase I. And we will divide the coarse point clouds into M patches. So, it is very important to ensure the semantic segmentation consistency of the patches during training. Inspired by (Chen et al. 2020; Cheng et al. 2022), we will do the Patch Segmentation (PS) on the generated coarse point clouds. For the completed coarse point clouds $C = \{p_j\}_{j=1}^{N^C}$, we first leverage MLP to obtain the probability $G = \{g_{ij}\}_{i=1}^{M}{}_{j=1}^{N^C}$ that each point p_j belongs to the i'th patch. Next, we use the probability obtained in the previous to weight and sum all the point coordinates of C to get the mean value of the predicted classification results, that is, to generate key points $Y = \{y_i\}_{i=1}^{M}$. The equation is as follows:

$$y_i = \sum_{j=1}^{N^C} (g_{ij} p_j). \tag{1}$$

After that, we assign each point on C to the patch with the highest probability patch. So, coarse point clouds will be decomposed into patches, each including the points assigned to this patch. We extract the features of the points and sum the features of the points of each patch and get the final patch feature $F^\Omega = \{f_i^\omega\}_{i=1}^{M}$.

Feature Fusion Module

We fuse the acquired features of two different modalities to preserve the global structural information. The attention mechanism is suited to finding correspondences between the features of different regions and has been applied to the correspondence of features in different regions in the point cloud (Zhang et al. 2022). Inspired by this method, we use the geometric information provided by the point cloud to supplement the global information provided by the image. In specially, we project the image features to form the query, while the point cloud features are projected to form the key

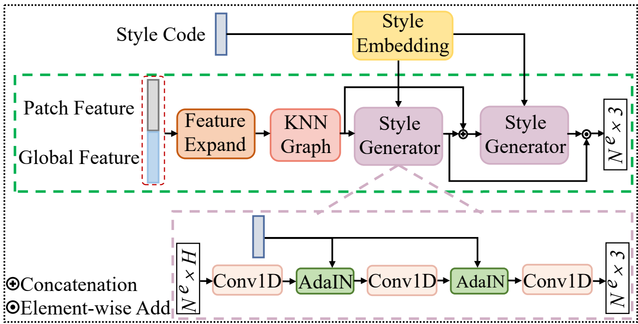

Figure 3: The structure of patch generator and its internal style generator. N^e in the above figure represents the number of points in the patch.

and value. Following that, the attention mechanism combines features from various regions of the image based on weights established by the cross-correlation between the two modalities. The equations are as follows:

$$Q = F^I W^Q, \quad K = F^P W^K, \quad V = F^P W^V, \tag{2}$$

$$F^A = SoftMax(QK^T/\sqrt{dim})V, \tag{3}$$

where W^Q, W^K, and W^V respectively denote the projected weights. dim denotes the dimension of the feature. The cross-attention fused features F^A can be regarded as the original image features enriched by the point cloud features. Subsequently, F^A will pass through a max pooling to get the global style code $F^S = \{f^s\}$ to help generate the structural details of the shape in the patch generator's generation process.

Patch Generator Module

The patch generator shown in Figure 3 aims to create fine-grained point cloud patches. As described in Sec 3.1, phase I provides local coarse-grained patch information, while the partial point clouds provide fine-grained geometric information. To fully use these two levels of information, for each coarse-grained patch feature, the fine-grained geometric information will be repeated and contacted to obtain the final patch feature $F^{\Omega'} = \{f_i^{\omega'}\}_{i=1}^M$ that includes more detailed geometric information. Next, we expand each final patch feature $f_i^{\omega'}$ to build a patch-based local point cloud space with fully connected layers. Inspired by (He et al. 2022), we follow the similar design of the KNN-Graph module from ECG (Pan 2020) to keep the local geometric features. Through the above operations, we get the augmented local patch point cloud features $F^E = \{f_i^e\}_{i=1}^M$. And $f^e \in {}^{N^e \times H}$, where N^e represents the number of patch region points and H is the dimension of the patch points features.

Drawing inspiration from the success of StyleGAN and its application (Karras, Laine, and Aila 2019; Xie et al. 2021), we propose a style-based approach called style generator to generate more global structural details during point cloud generation by cross-modal information. In particular, we utilize an Adaptive Instance Normalization (AdaIN) module to inject f^s into the style generator's internal layers. We start with a mini-batch f^e of point activations that are linearly transformed from the input. In the next, we normalize f^e to be $\overline{f^e}$ as follows:

$$\overline{f^e} = \frac{f^e - \mu^{f^e}}{\sigma^{f^e}}, \tag{4}$$

where μ^{f^e} and σ^{f^e} are the means and standard deviations of channel-wise activations of f^e. In order to integrate the style code, we compute the new activations f^o by denormalizing it according to the style code f^s. The formula is as follows:

$$f^o = \gamma^{f^s} \otimes \overline{f^e} + \beta^{f^s}, \tag{5}$$

where γ^{f^s} and β^{f^s} are two modulation parameters which are transformed from f^s through style embedding. We follow (He et al. 2016) to do the skip connection to avoid overfitting. In the end, we get the fine-grained patch point clouds $\Phi = \{\phi_i\}_{i=1}^M$.

Training Loss Function

Fan et al. (Fan, Su, and Guibas 2017) introduce Chamfer Distance (CD) and Earth Mover's Distance (EMD), which are commonly used in point clouds. We chose EMD to compute the loss between the generated dense point clouds and the ground truth point clouds because it better ensures the consistency of the generated point cloud density. Due to the different number of points, we use CD to calculate the loss between the key points of the point cloud and the ground truth point cloud to guide the patch segmentation in phase I. So, the reconstruction loss of the point cloud is defined as follows:

$$\begin{aligned} \mathcal{L}_{REC} = \alpha\mathcal{L}_{\text{EMD}}(C,T) + \mathcal{L}_{\text{EMD}}(D,G) \\ +\eta\mathcal{L}_{CD}(Y,T), \end{aligned} \tag{6}$$

where C and D represent the generated coarse point clouds and dense point clouds, respectively. T and G represent the ground truth of coarse point clouds and dense point clouds. We also flollow (Liu et al. 2020) to borrow the expansion penalty $\mathcal{L}_{EXP}$, which discourages the points from over-expanding.

In order to ensure the semantic consistency of the patch generated in the two phases, we add an additional regularization term. Specifically, for each generated patch point cloud of phase II, we seek an average pooling to obtain the key points $\hat{Y}$. We use the Root Mean Square Error (RMSE) to calculate the loss between key points Y and $\hat{Y}$ and the loss is defined as follows:

$$\mathcal{L}_{CON} = \mathcal{L}_{RMSE}(Y, \hat{Y}), \tag{7}$$

And the total loss is defined as follows:

$$\mathcal{L}_{total} = \mathcal{L}_{REC} + \lambda\mathcal{L}_{CON} + \xi\mathcal{L}_{EXP}. \tag{8}$$

Experiments

Datasets and Implementation Details

The dataset we used in our experiments is ShapeNet-ViPC (Zhang et al. 2021). We also follow ViPC to choose

Method	Airplane	Cabinet	Car	Chair	Lamp	Couch	Table	Watercraft	Avg
FoldingNet	0.883	2.574	2.293	1.961	1.941	2.187	2.048	1.357	1.906
PCN	0.790	2.402	2.337	1.931	1.478	2.278	1.892	1.153	1.783
TopNet	0.653	2.185	2.122	1.558	1.492	1.817	1.599	0.974	1.550
MSN	0.834	2.782	2.606	1.566	1.188	2.087	1.675	0.968	1.713
GRNet	1.037	2.381	3.803	1.974	1.470	2.389	1.795	1.347	2.025
PoinTr	0.712	3.174	2.740	2.791	2.333	2.904	3.057	1.521	2.404
XMFNet	0.566	2.006	1.715	1.314	1.197	1.868	1.442	0.826	1.367
CDPNet (Ours)	**0.482**	**1.357**	**1.470**	**1.193**	**0.829**	**1.340**	**1.358**	**0.499**	**1.066**

Table 1: Quantitative results are evaluated using the Chamfer Distance, with the calculated result multiplied by 10^3. In this table, a lower value signifies a more favorable outcome. The best results are highlighted in bold. We compare with FoldingNet, PCN, TopNet, MSN, GRNet, PoinTr, and XMFNet (Yang et al. 2018; Yuan et al. 2018; Tchapmi et al. 2019; Liu et al. 2020; Xie et al. 2020; Yu et al. 2021; Aiello, Valsesia, and Magli 2022).

Method	Airplane	Cabinet	Car	Chair	Lamp	Couch	Table	Watercraft	Avg
FoldingNet	0.925	0.671	0.668	0.758	0.800	0.689	0.765	0.862	0.767
PCN	0.933	0.619	0.613	0.723	0.813	0.621	0.759	0.864	0.743
TopNet	0.956	0.635	0.644	0.768	0.796	0.695	0.779	0.893	0.771
MSN	0.936	0.612	0.593	0.775	0.857	0.640	0.783	0.896	0.762
GRNet	0.885	0.556	0.461	0.685	0.792	0.578	0.732	0.808	0.687
PoinTr	0.946	0.615	0.614	0.707	0.793	0.632	0.726	0.831	0.733
XMFNet	0.968	0.669	0.708	0.825	0.858	0.697	0.826	0.925	0.810
CDPNet (Ours)	**0.983**	**0.807**	**0.770**	**0.853**	**0.915**	**0.809**	**0.837**	**0.976**	**0.869**

Table 2: Quantitative results are utilized the F-Score as a metric, where a higher value indicates superior performance. The best results are highlighted in bold. We compare with FoldingNet, PCN, TopNet, MSN, GRNet, PoinTr, and XMFNet (Yang et al. 2018; Yuan et al. 2018; Tchapmi et al. 2019; Liu et al. 2020; Xie et al. 2020; Yu et al. 2021; Aiello, Valsesia, and Magli 2022).

eight categories: airplane, cabinet, car, chair, lamp, couch, table, and watercraft. However, such a large amount of data brings a very large computational overhead to the experiment, so we take a subset of the dataset provided by ViPC for training. The way we choose is as follows: for each category, we randomly take about 1/3 of the total training data for training. For the test dataset, we used all the test data. For all the point cloud data, we follow (Yuan et al. 2018) to align their angles and normalize them. The input partial point clouds and ground truth point clouds both contain 2048 points. For the image data, the resolution of pixels is 224×224. We train the network with a batch size of 32. The initial learning rate is 1e-4 and decayed by 0.7 after 20 epochs. The optimization is set to stop after 150 epochs. The initial value of α is set to 1, which will change with the number of iterations. After iterating 50 epochs, we set it to 0.5. The method's hyper-parameters (η, λ, ξ) are set to: (0.1, 0.1, 0.01).

Comparision Results

In this section, we will compare our method with several previous methods for point cloud completion. We compare with the recently cross-modal point cloud completion methods XMFNet (Aiello, Valsesia, and Magli 2022). We also compare the results of several architectures FoldingNet, PCN, TopNet, MSN, GRNet, and PoinTr (Yang et al. 2018; Yuan et al. 2018; Tchapmi et al. 2019; Liu et al. 2020; Xie et al. 2020; Yu et al. 2021) for completion with single-modal data (point clouds) input when retraining on the ShapeNet-ViPC dataset. FoldingNet is an auto-encoder that uses a 2D grid, while PCN is an encoder-decoder model that employs a coarse-to-fine strategy. TopNet utilizes a tree structure network to complete the partial point cloud to preserve the topology. MSN proposes a method for point cloud completion based on parametric surfaces. GRNet introduces voxels as intermediate representations to normalize point clouds for coarse-to-fine completion. PoinTr proposes a transformer-based autoencoder for point cloud completion. For these methods, we use open-source code and parameters for retraining. We tune the number of input partial point clouds and output complete point clouds as 2048 to adjust the dataset.

Quantitative Evaluation. We follow (Zhang et al. 2021; Aiello, Valsesia, and Magli 2022) to choose the CD and F-score as the metrics for reconstruction quality. A lower CD score means better performance and a higher F-Score means better performance. The results for each category and the average are summarized in Table 1 and Table 2. The proposed method exhibits better performance compared to other methods across all eight categories, as evidenced by the improvement observed in both CD and F-Score metrics.

Qualitative Evaluation. Results of the representative ex-

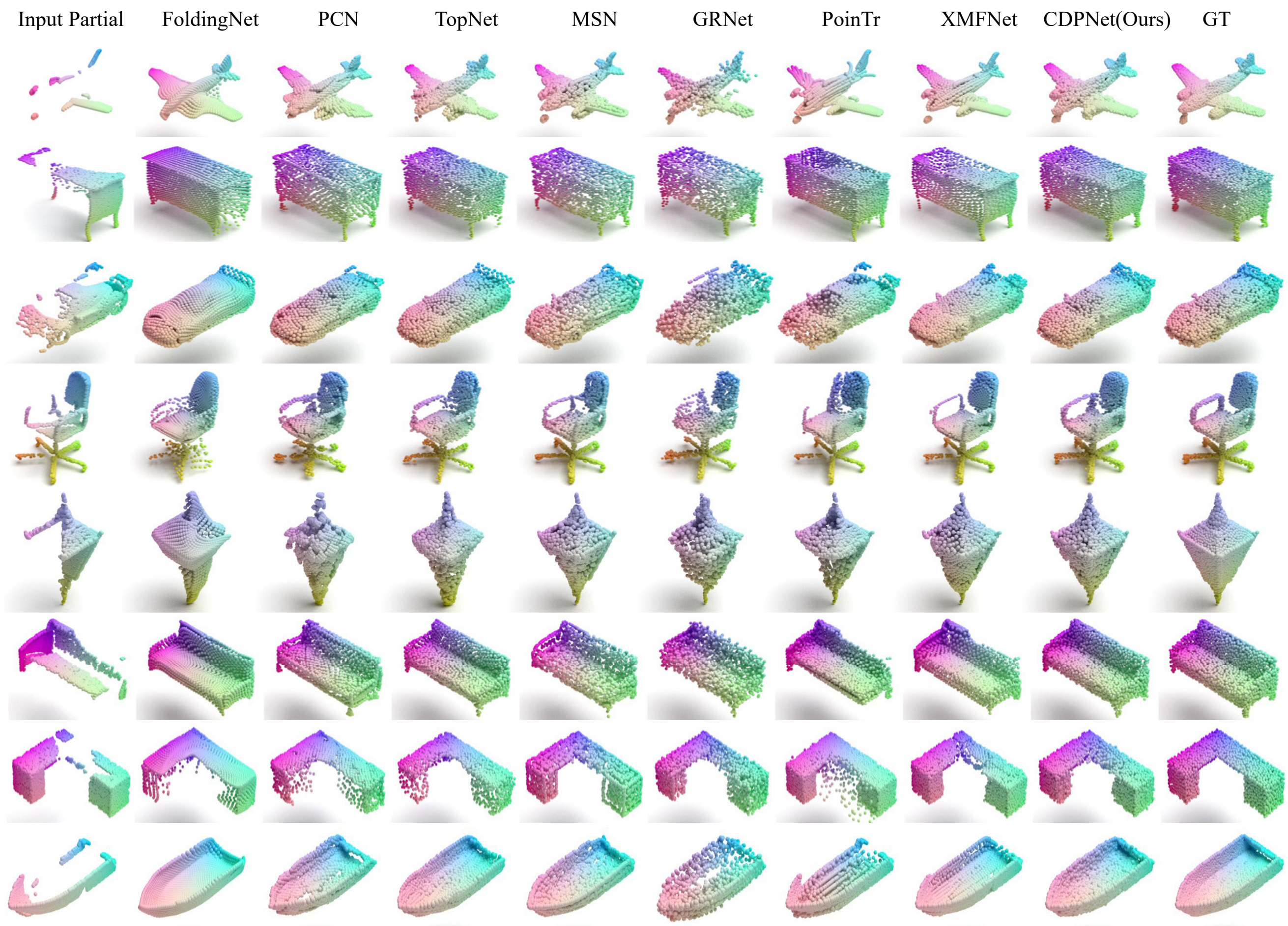

Figure 4: Qualitative comparisons with previous state-of-the-art methods. We set the resolution for partial, complete, and ground truth point clouds are 2,048. Compared with previous methods, our method can generate more details.

amples from the eight categories are shown in Figure 4. From the qualitative experimental results, it is evident that the single-modal method produces flawed results with detailed deficiencies. For FoldingNet, PCN and GRNet, although their generated results can fill in the missing parts, the whole generated results are a little messy. For TopNet and MSN, these methods can preserve some geometric information and structural information. However, shape details are still missing in the results. We can see that the results surface of planes, cars, tables, etc. generated by these methods are not smooth enough. For cabinets, couches, lamps, and watercraft, the results generated by these methods also lack details in the structure (such as the back and leg of the chair). For PoinTr, it achieves good results in high-resolution (16384 and 8192 points) single-modal point cloud completion. However, this experiment is trained and tested on the data of 2048 points, so PoinTr is likely to be overfitting, which leads to poor experimental results. Methods using cross-modal are able to preserve the geometric and structural information of shapes, as shown by XMFNet and our method. However, XMFNet still lacks some information, such as the rear of the car, and the legs of the chair. Our results show visually better performance in all eight categories than baselines.

M	Airplane	Cabinet	Car	Watercraft
4	**0.476**	1.407	1.493	0.543
8	0.482	**1.357**	**1.470**	**0.499**
16	0.487	1.381	1.502	0.508
32	0.512	1.433	1.635	0.571

Table 3: Quantitative evaluation for the different number of patch generators. And M represents the different number of patches.

Ablation Study

Analysis of the Number of the Patch Generator. We do experiments to analyze the effect of the number patch generator for the final results. We use M to represent the number of patch generators. As shown in Table 3, when the CDPNet in each number of the patch generator is fixed, more patch generators will lead to better Chamfer Distance. However, it appears to be decreased when $M \geq 8$. An increase in the number of patches may augment parameters, leading to network over-fitting. Our experiments across four classes (airplane, cabinet, car, and watercraft) reveal that with the stable network parameters, $M = 8$ achieves the optimal Chamfer Distance. Thus, we adopt $M = 8$ for our experiments.

Dual Phases	Multi-PG	FF	Score
-	-	-	1.850
✓	-	-	1.465
✓	✓	-	1.103
✓	✓	✓	**1.066**

Table 4: Module Ablation Study.

Analysis of the Effectiveness of the Module. To demonstrate the effectiveness of each module, we conduct the module ablation study. And the results are summarized in Table 4. Especially, Dual Phases represent the dual phases structure in our method. Multi-PG represents the multi-patch generator. FF represents the feature fusion module. In the first experiment, we remove all the above modules. Keeping DGCNN as the encoder and using MLP as the decoder to do the completion. In the second experiment, we add phase I. The multi-patch generator structure is used since the coarse point clouds will be segmented, and the patch generator uses MLP. In the third experiment, we replace the patch generator from the second experiment with ours but without injecting style code because of removing the feature fusion module. The final experiment used all the modules we designed. Throughout the experiments, we also used CD as the calculated score.

Our module ablation experiments reveal that the developed patch generator effectively creates fine-grained point cloud patches, while phase I of our network contributes valuable priors for final results. Additionally, the style code from the feature fusion module maintains the shape's structure and enhances performance.

Discussion on the Single-View Image

We study the impact of the auxiliary image input on the complete performance. For each partial point clouds, we produce 24 complete point clouds, each generated with the reference of an image from the 24 rendered views. We demonstrate some representative results in Figure 5. In the figure, we respectively show the single-view image with the best completion result and the single-view image with the worst completion result. The results show that the addition of the image input provides a significant improvement in performance. Besides, we can observe from the results that the more the image contains missing parts, the better the effect of completion (Such as cabinet, the first picture contains more missing information, so better results are obtained). We also report performance for the best views, worst views, and random views in the test dataset, as shown in Table 5. This highlights

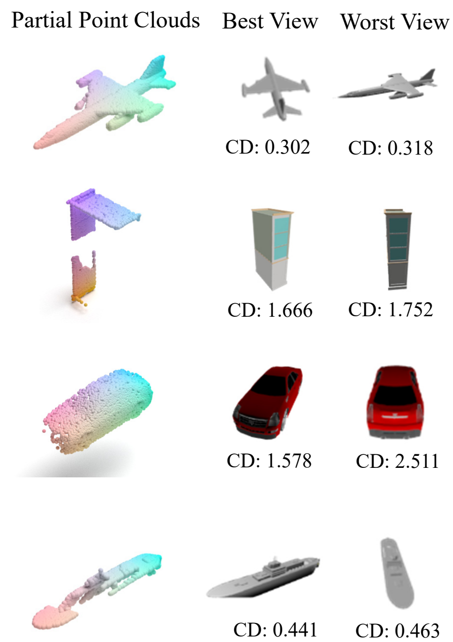

Figure 5: Different views provide different complementary performances. Each input partial point cloud is shown on the left, and the CD measures quantitative.

Method	Airplane	Cabinet	Car	Watercraft
Worst View	0.490	1.378	1.513	0.509
Random view	0.482	1.357	1.470	0.499
Best view	**0.474**	**1.323**	**1.450**	**0.486**

Table 5: Quantitative evaluation for views.

the potential of 'good' views to provide supplementary information, while 'bad' views offer limited insight. Identifying the 'good' views for each partial point cloud represents a promising direction for future exploration.

Conclusion

We propose CDPNet, a novel cross-modal point cloud completion framework that leverages an extra single-view image to recover missing global information and preserve the local geometric details through the patches. We divide the coarse point clouds into multiple patches and generate dense point clouds based on the patches. Our approach also enriches image information with point cloud information to obtain the style code to guide the generation of shape structural details. We compare our method with existing single-modal and cross-modal point cloud completion methods and demonstrate the performance improvements.

Acknowledgments

This work was supported by the National Key RD Program of China (Grant No. 2022YFB3104600), Chengdu Science and Technology Project (2019-YF08-00285-GX), and the

National Natural Science Foundation of China under Grant NO. 61976156, and partially funded by Grant SCITLAB-30005 of Intelligent Terminal Key Laboratory of Sichuan Province, and partially supported by the National Natural Science Foundation of China under grant U20B2063, 62220106008 and 62306067.

References

Aiello, E.; Valsesia, D.; and Magli, E. 2022. Cross-modal Learning for Image-Guided Point Cloud Shape Completion. In *Advances in Neural Information Processing Systems*.

Chen, N.; Liu, L.; Cui, Z.; Chen, R.; Ceylan, D.; Tu, C.; and Wang, W. 2020. Unsupervised learning of intrinsic structural representation points. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9121–9130.

Chen, Z.; Dai, C.; Jiang, L.; Sheng, B.; Zhang, J.; Lin, W.; and Yuan, Y. 2016. Structure-aware image inpainting using patch scale optimization. *Journal of Visual Communication and Image Representation*, 40: 312–323.

Cheng, A.-C.; Li, X.; Liu, S.; Sun, M.; and Yang, M.-H. 2022. Autoregressive 3d shape generation via canonical mapping. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, 89–104. Springer.

Du, Z.; Lu, Y.; Wang, G.; Xie, N.; and Yang, Y. 2023. GT-Net: Variational Autoencoder Networks based on Graph Transformer for 3D Shape Learning. In *2023 IEEE International Conference on Multimedia and Expo (ICME)*, 918–923. IEEE.

Eldar, Y.; Lindenbaum, M.; Porat, M.; and Zeevi, Y. Y. 1997. The farthest point strategy for progressive image sampling. *IEEE Transactions on Image Processing*, 6(9): 1305–1315.

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

Groueix, T.; Fisher, M.; Kim, V. G.; Russell, B. C.; and Aubry, M. 2018. A papier-mâché approach to learning 3d surface generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 216–224.

Guo, Y.; Wang, H.; Hu, Q.; Liu, H.; Liu, L.; and Bennamoun, M. 2020. Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(12): 4338–4364.

Harary, G.; Tal, A.; and Grinspun, E. 2014. Context-based coherent surface completion. *ACM Transactions on Graphics (TOG)*, 33(1): 1–12.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

He, Q.; Xie, N.; Du, Z.; Ren, K.; and Dou, J. 2022. PDP-NET: Patch-Based Dual-Path Network for Point Cloud Completion. In *2022 IEEE International Conference on Multimedia and Expo (ICME)*, 1–6. IEEE.

Karras, T.; Laine, S.; and Aila, T. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4401–4410.

Li, X.; Liu, S.; Kim, K.; De Mello, S.; Jampani, V.; Yang, M.-H.; and Kautz, J. 2020. Self-supervised single-view 3d reconstruction via semantic consistency. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, 677–693. Springer.

Liu, M.; Sheng, L.; Yang, S.; Shao, J.; and Hu, S.-M. 2020. Morphing and sampling network for dense point cloud completion. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 11596–11603.

Nguyen, A.-D.; Choi, S.; Kim, W.; and Lee, S. 2019. GraphX-convolution for point cloud deformation in 2D-to-3D conversion. In *Proceedings of the ieee/cvf international conference on computer vision*, 8628–8637.

Öngün, C.; and Temizel, A. 2021. LPMNet: latent part modification and generation for 3D point clouds. *Computers & Graphics*, 96: 1–13.

Pan, J.; Han, X.; Chen, W.; Tang, J.; and Jia, K. 2019. Deep mesh reconstruction from single rgb images via topology modification networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9964–9973.

Pan, L. 2020. ECG: Edge-aware point cloud completion with graph convolution. *IEEE Robotics and Automation Letters*, 5(3): 4392–4398.

Pauly, M.; Mitra, N. J.; Giesen, J.; Gross, M. H.; and Guibas, L. J. 2005. Example-based 3d scan completion. In *Symposium on Geometry Processing*, CONF, 23–32.

Qi, C. R.; Su, H.; Mo, K.; and Guibas, L. J. 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 652–660.

Qi, C. R.; Yi, L.; Su, H.; and Guibas, L. J. 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30.

Shao, T.; Xu, W.; Zhou, K.; Wang, J.; Li, D.; and Guo, B. 2012. An interactive approach to semantic modeling of indoor scenes with an rgbd camera. *ACM Transactions on Graphics (TOG)*, 31(6): 1–11.

Tchapmi, L. P.; Kosaraju, V.; Rezatofighi, H.; Reid, I.; and Savarese, S. 2019. Topnet: Structural point cloud decoder. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 383–392.

Thrun, S.; and Wegbreit, B. 2005. Shape from symmetry. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 2, 1824–1831. IEEE.

Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S. E.; Bronstein, M. M.; and Solomon, J. M. 2019. Dynamic graph cnn for learning on point clouds. *Acm Transactions On Graphics (tog)*, 38(5): 1–12.

Wang, Z.; Gao, Z.; Wang, G.; Yang, Y.; and Shen, H. T. 2023a. Visual Embedding Augmentation in Fourier Domain for Deep Metric Learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 1–11.

Wang, Z.; Gao, Z.; Xu, X.; Luo, Y.; Yang, Y.; and Shen, H. T. 2022. Point to Rectangle Matching for Image Text Retrieval. In *Proceedings of the 30th ACM International Conference on Multimedia*, 4977–4986.

Wang, Z.; Xu, X.; Wang, G.; Yang, Y.; and Shen, H. T. 2023b. Quaternion Relation Embedding for Scene Graph Generation. *IEEE Transactions on Multimedia*, 1–12.

Wen, X.; Li, T.; Han, Z.; and Liu, Y.-S. 2020. Point cloud completion by skip-attention network with hierarchical folding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1939–1948.

Wen, X.; Xiang, P.; Han, Z.; Cao, Y.-P.; Wan, P.; Zheng, W.; and Liu, Y.-S. 2021. Pmp-net: Point cloud completion by learning multi-step point moving paths. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7443–7452.

Wen, X.; Zhou, J.; Liu, Y.-S.; Su, H.; Dong, Z.; and Han, Z. 2022. 3D shape reconstruction from 2D images with disentangled attribute flow. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3803–3813.

Xie, C.; Wang, C.; Zhang, B.; Yang, H.; Chen, D.; and Wen, F. 2021. Style-based point generator with adversarial rendering for point cloud completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4619–4628.

Xie, H.; Yao, H.; Zhou, S.; Mao, J.; Zhang, S.; and Sun, W. 2020. Grnet: Gridding residual network for dense point cloud completion. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX*, 365–381. Springer.

Xue, Y.; Chen, J.; Zhang, Y.; Yu, C.; Ma, H.; and Ma, H. 2022. 3D Human Mesh Reconstruction by Learning to Sample Joint Adaptive Tokens for Transformers. In *Proceedings of the 30th ACM International Conference on Multimedia*, 6765–6773.

Yang, Y.; Feng, C.; Shen, Y.; and Tian, D. 2018. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 206–215.

Yu, X.; Rao, Y.; Wang, Z.; Liu, Z.; Lu, J.; and Zhou, J. 2021. Pointr: Diverse point cloud completion with geometry-aware transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 12498–12507.

Yuan, W.; Khot, T.; Held, D.; Mertz, C.; and Hebert, M. 2018. Pcn: Point completion network. In *2018 International Conference on 3D Vision (3DV)*, 728–737. IEEE.

Yussif, S. B.; Xie, N.; Yang, Y.; and Shen, H. T. 2023. Self-Relational Graph Convolution Network for Skeleton-Based Action Recognition. In *Proceedings of the 31st ACM International Conference on Multimedia*, 27–36.

Zhang, S.; Liu, X.; Xie, H.; Nie, L.; Zhou, H.; Tao, D.; and Li, X. 2023. Learning Geometric Transformation for Point Cloud Completion. *International Journal of Computer Vision*, 1–21.

Zhang, W.; Dong, Z.; Liu, J.; Yan, Q.; Xiao, C.; et al. 2022. Point cloud completion via skeleton-detail transformer. *IEEE Transactions on Visualization and Computer Graphics*.

Zhang, X.; Feng, Y.; Li, S.; Zou, C.; Wan, H.; Zhao, X.; Guo, Y.; and Gao, Y. 2021. View-guided point cloud completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15890–15899.