Image-guided Multi-level Feature Fusion Point Cloud Completion Network

Wanpeng Zhang and Ziwei Zhou*

Abstract—**To address the challenges of structural information loss in image feature extraction commonly encountered in current point cloud completion methods, as well as the limited capability of existing self-structured dual generators to effectively capture the spatial structures and fine details of point clouds, we propose an image-guided multi-level feature fusion point cloud completion model (IGMLNet). This model begins by designing an efficient residual feature extractor and integrating the ECA attention mechanism into the residual blocks of ResNet18. This integration enhances the model's ability to perceive both spatial structures and fine-grained details, thereby improving the overall feature extraction from point clouds. Next, a multi-level feature fusion module is introduced, utilizing a hierarchical attention mechanism that allows for comprehensive integration of cross-modal features from both point clouds and images. This facilitates the deep fusion and optimization of information from different modalities, enhancing the model's ability to leverage complementary information for more accurate completion. Furthermore, a Structured point cloud recovery module is employed to address the issues of missing and incomplete data in rough point clouds. Through deep refinement and structural reconstruction, this module significantly improves the completeness and quality of the reconstructed point clouds. The model has been thoroughly evaluated through extensive comparison and ablation experiments on the PCN and ShapeNet55 datasets. The experimental results demonstrate that IGMLNet achieves superior performance, maintaining high integrity while producing point clouds with rich surface details and accurate geometric features.**

Index Terms—**Point Cloud Completion, Efficient Channel Attention, Multi-level Feature Fusion, Point cloud optimization**

I. Introduction

In recent years, point clouds have gained significant attention as a representation of 3D objects. They can be easily captured by 3D scanning devices and depth cameras. However, due to limited viewpoints or occlusions, as well as sensor resolution constraints, the raw point clouds captured by 3D scanners and depth cameras are often sparse and incomplete. As a result, point cloud completion, which involves predicting a complete point cloud from a partial one, plays a crucial role in various downstream tasks in computer vision. Thanks to large-scale point cloud datasets, recent research on point cloud completion [1], [2], [3], [4], [5], has successfully leveraged deep learning methods, which have shown to outperform traditional geometry-based [6], [7], [8] and alignment-based methods [9], [10], [11], providing more reliable and flexible results. Deep learning-based point cloud completion methods have attracted increasing research interest. Although various learning-based techniques have demonstrated promising results [12], [13], [14], [15], the sparsity and structural incompleteness of the captured point clouds still limit the ability of these methods to produce satisfactory results. We identify two main challenges in this task.

The first challenge is that key semantic parts may be missing, leading to a significant gap in how point-based networks recognize global shapes and locate missing regions. Most of the papers on point cloud completion [16], [17], [18], [19], have primarily focused on unimodal problems, where only prior knowledge about the 3D shape is utilized. It was only recently that image-guided completion started receiving attention, with the expectation that point cloud completion techniques could benefit from 2D images. Some methods attempt to address this by incorporating additional color images [20], [21], but paired images are often difficult to obtain and may not be well-calibrated in terms of intrinsic parameters. ViPC uses an additional single-view image to provide global shape priors during the coarse completion stage. However, in the refinement stage, due to the simple concatenation of features learned from different modalities, it fails to recover high-frequency details and local topology of complex shapes. As a result, the shapes completed by ViPC tend to be noisy and lack finer geometric details. CSDN reformulates the fusion of image and point cloud features as a shape style transfer problem. The affine parameters in instance normalization, generated from different image features, alter the point-wise feature statistics, normalizing the output point cloud into different shapes. However, relying solely on instance normalization for statistical features may not fully capture the complex correspondences between images and point clouds, leading to the loss of some fine-grained local features.

The second challenge is how to infer detailed structures. Snowflakenet [3] and FBNet [15] use skip connections between multiple refinement steps, enabling them to better utilize previously learned shape patterns to iteratively restore finer details. However, for regions with subtle changes, complex textures, or irregular shapes, it can still be difficult to recover accurate details. VRCNet [22] enhances the

Manuscript received December 9, 2024; revised February 28, 2025.

This work was supported by the Natural Science Foundation of China (No. 61575090), the Natural Science Foundation of China Youth Fund (No. 61803189), Natural Science Foundation of Liaoning Province(2019-ZD-0031 and 2020FWDF13).

Wanpeng Zhang is a postgraduate student at the School of Computer Science and Software Engineering, University of Science and Technology LiaoNing, Anshan 114051, China (phone:86-16642299371, e-mail: 584593254@qq.com).

Ziwei Zhou* is an Associate Professor at the School of Computer Science and Software Engineering, University of Science and Technology LiaoNing, Anshan 114051, China (Corresponding author to provide phone: 86-139-4125-5680; e-mail: 381431970@qq.com).

preservation of original details by structuring relationships, which performs well in retaining original details. However, in some complex or highly sparse point clouds, the completion results may still not be perfect. Seedformer retains more local and global information through non-pooling encoding methods [23], but if the seed points and the quality of the known point cloud are poor, the completion performance may degrade significantly. SVDFormer captures global shapes from point cloud data and depth maps from multiple viewpoints. To better utilize cross-modal information, it introduces a feature fusion module that enhances interaction between views, improving recognition efficiency. It also uses a self-structural dual generator to refine coarse completions. However, the image feature extraction in SVDFormer is too simplistic, leading to the loss of important structural information from the image. Despite the fusion mechanism achieving decent results, alignment issues between different modalities may still arise, leading to dimensional mismatches and information loss. The self-structural dual generator can generate finer and smoother point clouds to some extent, but its perception of point cloud feature spatial structures and detail information needs further enhancement to better restore point cloud contours and generate more refined point clouds.

In summary, we present the Image-guided Multi-level Feature Fusion Point Cloud Completion model, or IGMLNet. This model features an efficient residual feature extractor, which incorporates the ECA attention mechanism within the residual blocks of ResNet18. This integration sharpens the model's sensitivity to spatial structures and detailed areas, thereby guiding point cloud feature extraction with enhanced efficacy. Furthermore, we introduce a multi-level feature fusion module that leverages a hierarchical attention mechanism to thoroughly amalgamate cross-modal features from point clouds and images, realizing a profound integration and optimization of information across different modalities.

To achieve point clouds that are more refined and complete, we have developed an innovative Structural Point Cloud Refinement Module. This module adeptly tackles the issues of missing and incomplete data in coarse point clouds through sophisticated refinement and structural reconstruction techniques. Employing a cascaded dual-module refinement strategy ensures the continuity and stability of the reconstruction process. The resultant point cloud model not only retains a high degree of completeness but also exhibits an abundance of surface details and precise geometric characteristics.

II. Related Jobs

Early learning-based methods [24], [25], [26], typically rely on voxelized 3D Convolutional Neural Networks (3D CNNs) to represent data. However, these methods face challenges such as high computational costs and limited resolution. Additionally, GRNet [4] and VE-PCN [27] use 3D grids as intermediate representations during the point cloud completion process.

In recent years, several end-to-end networks capable of directly processing point clouds have emerged. PCN [12], a pioneering point-based method, extracts features through a shared Multi-Layer Perceptron (MLP) and generates additional points in a coarse-to-fine manner using folding operations [28], enabling point cloud completion. Inspired by PCN, many subsequent point-based methods [18], [29], [23], have been proposed, further advancing the field.

To overcome the limitations of relying solely on point data for local shape information, some studies [30], [31], have explored the use of auxiliary modalities to enhance completion performance, known as cross-modal methods. These approaches typically combine rendered color images with partial point cloud data, using corresponding camera parameters for completion. While these methods have shown good results in experiments, they often rely on additional data inputs that are difficult to obtain in real-world applications.

In contrast to these 3D data-based methods, MVCN [32] performs completion in the 2D domain using a Conditional Generative Adversarial Network (GAN). However, this approach lacks the ability to leverage ground truth data, which contains rich spatial information, to supervise the completion results. Furthermore, some methods [30], [33] attempt to supervise point cloud completion in the 2D domain by projecting the completed points onto a 2D plane and comparing them with ground truth depth maps to calculate the loss. Unlike these methods, our approach provides a more comprehensive understanding of the overall shape by analyzing the structure of the 2D input itself, allowing for shape perception during training without the need for additional information or differentiable rendering.

To generate high-quality details, many studies have introduced various strategies that optimize detail generation by learning the contextual and local spatial relationships of shapes. To achieve this, state-of-the-art methods have designed various improved modules to learn more accurate shape priors from the training data. For instance, SnowflakeNet [3] introduces Snowflake Point Deconvolution (SPD) and uses a skip-transformer to model the relationship between parent and child points. FBNet [15] employs a feedback mechanism during the refinement process to iteratively generate points. LAKe-Net [8] integrates surface skeleton representations into the refinement stage, facilitating the learning of missing topological structures.

Another class of methods focuses on retaining and utilizing local information from partial inputs. One direct approach is to combine the generated results with partial input data to predict the missing points. Since point sets can be treated as sequences of tokens, PoinTr [8] uses a transformer architecture to predict missing point proxies. SeedFormer [23] introduces a shape representation method called "patch seeds" to prevent the loss of local information during pooling operations. In addition, some methods [16], [22], [34] enhance the generated shape in the refinement stage by leveraging structural relationships. However, these strategies typically apply a uniform refinement method to all points, which limits their ability to generate detailed geometric features for individual points. SVDFormer [35] extracts features from multi-view information and learns local shape priors and alignment similarity modules to generate point clouds in a coarse-to-fine manner. These methods have improved the accuracy of point cloud completion to some extent, introducing novel ideas for feature extraction and point cloud generation. However, they

still fail to fully address issues such as the uneven distribution of points in the completed point cloud, and the restored details may still be missing or incorrect.

III. Model Design

The overall architecture of the proposed attention-based multi-level feature fusion point cloud completion model is shown in Figure 1. This architecture consists of two main parts.

The first part is the generation of the rough point cloud. In this part, PointNet++ is used to extract point cloud features, and an efficient residual feature extractor is applied to extract features from the image projections of the incomplete point cloud. Unlike the standard ResNet18, the ECA attention mechanism is integrated into the residual blocks to enhance the perception of key features. This mechanism enables adaptive optimization of channel weights and effectively improves the learning of critical information. Then, the multi-level feature fusion module, which uses two layers of cross-attention mechanisms, performs preliminary feature alignment and deep complementary information extraction. This achieves a coarse-to-fine feature fusion process, effectively handling the modal differences between point clouds and images while avoiding information loss. A layer of self-attention is used to establish long-range dependencies within the feature space, capturing global semantic information to enhance feature representation and eliminating local noise to improve feature robustness. Next, through one-dimensional transposed convolution upsampling and self-attention processing in the decoder, the global 3D shape is generated. Finally, after merging the incomplete point cloud with the global shape generated by the decoder, a resampling process is applied to obtain the final rough point cloud.

The second part is the generation of the refined point cloud. A structured point cloud refinement module is proposed, which employs a cascaded dual-module refinement strategy to perform deep refinement and structural reconstruction of the rough point cloud. The quality of the point cloud is significantly improved, resulting in a more accurate and complete point cloud.

A. Rough Point Cloud Generation Network

The rough point cloud generation network is designed to reconstruct an initial complete point cloud structure from the input incomplete point cloud. First, a three-layer PointNet++ architecture is employed to extract features from the incomplete point cloud, capturing geometric information and structural characteristics, and generating a global feature vector. The incomplete point cloud is then projected from three different viewpoints to create corresponding depth maps, and ResNet18 is utilized to extract image features. Next, a multi-level feature fusion module is applied, where a cross-attention mechanism integrates point cloud and image features, while a self-attention mechanism captures global contextual relationships, resulting in more comprehensive and expressive output features. In the decoder, the fused features undergo upsampling and further self-attention processing to generate the initial 3D point cloud. Finally, the generated 3D shape is combined with the input incomplete point cloud, followed by resampling to obtain the rough point cloud, which serves as the foundation for the subsequent fine reconstruction module. Through these steps, the network effectively extracts rich geometric and structural information, generates an initial complete point cloud, and lays a solid foundation for subsequent optimization and refinement.

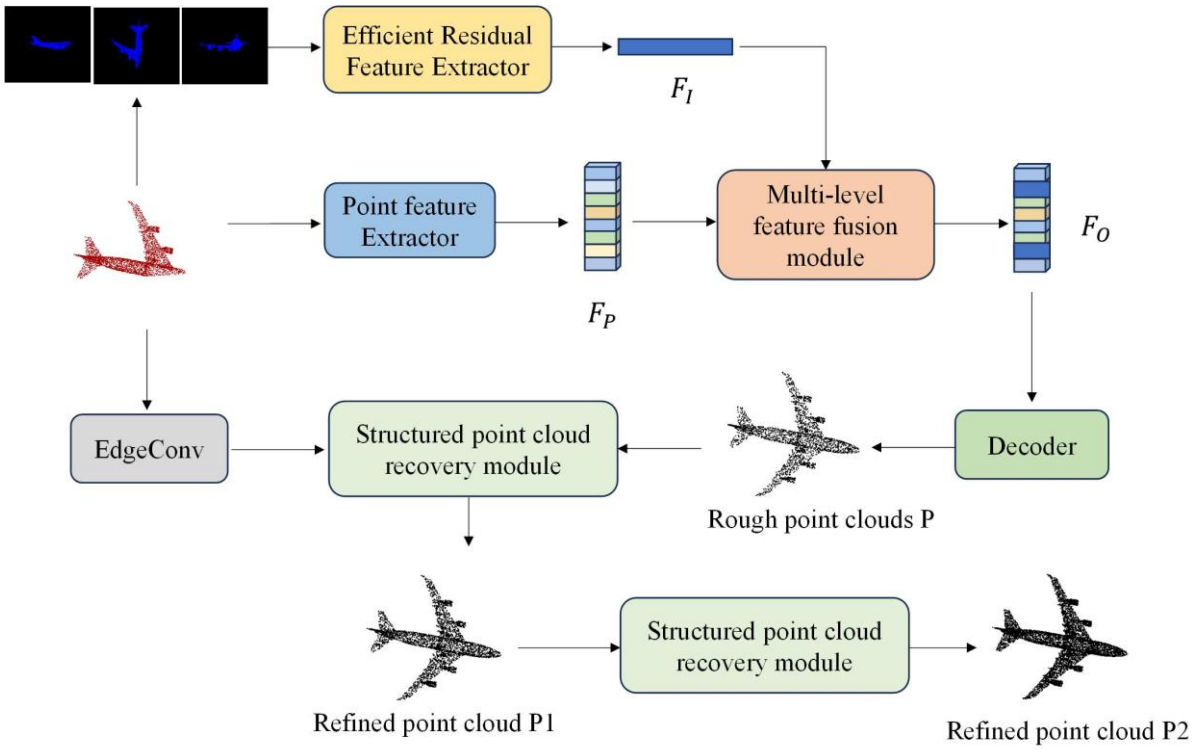

Fig. 1. The overall structural diagram of the Image-guided Multi-level Feature Fusion Point Cloud Completion Network

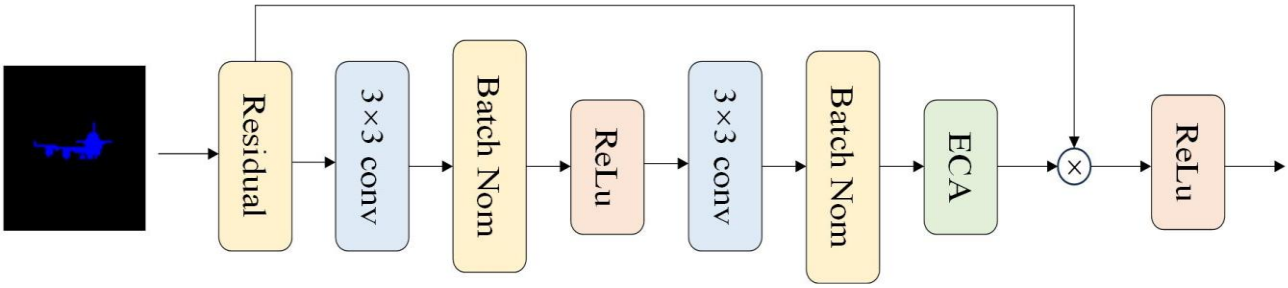

Fig. 2. The structural diagram of the Efficient Residual Feature Extractor

1) *Efficient Residual Feature Extractor:* The Efficient Residual Feature Extractor is a high-efficiency feature extraction module that combines the ResNet18 residual network with the Efficient Channel Attention (ECA) mechanism, as illustrated in Figure 2. In this module, the ECA attention mechanism is integrated into the residual blocks of ResNet18, with the goal of adaptively adjusting the weights of each channel. This enables the network to focus more effectively on features that are most relevant to the task, while suppressing irrelevant or noisy information. The ECA mechanism assigns different weights to each channel, optimizing the skip connections within the residual blocks and enhancing the network's ability to learn essential features.

This approach improves the network's sensitivity to both the spatial structure and fine details of the image, making the extracted features more complementary to the point cloud features during subsequent feature fusion. Furthermore, the introduction of the ECA attention mechanism does not significantly increase computational cost, allowing the model to improve performance while maintaining high efficiency.

Specifically, the process begins by extracting shallow features from the input 2D image I through a convolutional layer, denoted as F_L. The feature is then enhanced using Batch Normalization and the ReLU activation function, improving the model's stability and introducing non-linearity, resulting in the feature F_O, as shown in the following equation:

$$F_L = Conv3\times3(I) \tag{1}$$

$$F_O = \mathrm{Re}\,LU(BN(F_L)) \tag{2}$$

Next, a second convolution and Batch Normalization operation are applied to extract higher-level features F_H. Finally, the features are weighted using the ECA attention module to generate the final output feature Fout F_{out}, as described by the equation:

$$F_H = \mathrm{Re}\,LU(BN(Conv3\times3(F_O))) \tag{3}$$

$$F_{out} = ECA(FH) \tag{4}$$

In these equations, $Conv3\times3$ represents the convolution operation, BN stands for Batch Normalization, and $\mathrm{Re}\,LU$ ReLU denotes the ReLU activation function.

2) *Multi-level Feature Fusion Module:* Point cloud data and 2D image data typically come from the same scene, which leads to certain similarities in their features. However, as they belong to different modalities, there are significant differences in their feature representations and information structures. In practical applications, one of the key challenges is how to effectively fuse these two types of data while preserving fine-grained and learnable features. Some researchers have attempted to fuse the features by directly concatenating point cloud and image features, but experiments have shown that this approach is not optimal. The main issue lies in the inherent differences in feature representation and information characteristics between the two modalities, which makes simple concatenation insufficient to fully exploit their complementary strengths.

To address this problem, inspired by Perceiver IO, a multi-level feature fusion module is proposed, The structure diagram is shown in Figure 3. This module leverages two layers of cross-attention mechanisms and one layer of self-attention mechanism to achieve deep fusion and optimization of image and point cloud features. The cross-attention mechanism enables point cloud features to capture additional structural information from image features, thus improving their expressiveness. Meanwhile, the self-attention mechanism captures global contextual relationships within the fused feature sequence, enhancing feature consistency and semantic understanding.

In the first step of the module, the extracted point cloud features F_P and image features F_I are input into the cross-attention module. The point cloud features F_P are projected into a query tensor (Q), while the image features F_I are projected into key (K) and value (V) tensors. By calculating attention weights between the point cloud and image features, the cross-attention mechanism automatically learns the correlation and complementarity between the two modalities, generating an initial fused feature F_S. This fused feature contains both the geometric information of the point cloud and the visual information from the image. During this process, the geometric information of the point cloud is preserved, preventing information loss or excessive fusion, and thus achieving a balanced multi-modal feature integration.

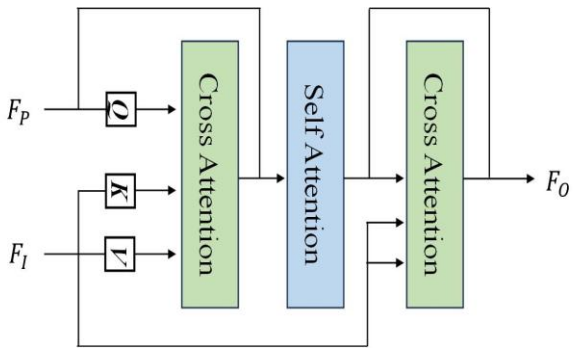

Fig. 3. The structural diagram of the Multi-level Feature Fusion Module

This fusion strategy not only enhances the semantic understanding of point cloud features but also strengthens the role of image information in point cloud completion tasks, providing a solid foundation for subsequent self-attention optimization:

$$Q = F_P W_Q^1 \tag{5}$$

$$K = F_I W_K^1 \tag{6}$$

$$V = F_I W_V^1 \tag{7}$$

$$CrossAttention(Q,K,V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \tag{8}$$

$$F_S = CrossAttention(Q,K,V) + F_P \tag{9}$$

In this context, W_Q^1, W_K^1, and W_V^1 are learnable projection weight matrices, and d_k is the dimension of the key vectors.

The fused initial feature F_S is passed into the self-attention module to further enhance its expressiveness. By capturing the global contextual relationships within the feature sequence, the self-attention mechanism makes the fused features more consistent and coherent, thus optimizing the fused feature F_E. The simplified computation formula is as follows, with the other parameters explained in the same way as in the first layer cross-attention mechanism:

$$SelfAttention = softmax(\frac{QK^T}{\sqrt{d_k}}) \tag{10}$$

$$F_E = SelfAttention(Q,K,V) \tag{11}$$

After the first layer of cross-attention and self-attention mechanisms, the enhanced feature F_E has already integrated multi-modal information from both the image and the point cloud. However, further optimization is still needed to refine the semantic relationships and structural details. Therefore, in the second layer of the cross-attention module, the enhanced feature F_E is used as the query, and image features F_I are reintroduced as the keys and values, further optimizing the alignment and fusion of cross-modal features. This step addresses any potential misalignment issues that may arise from the first cross-attention stage. It strengthens the feature transfer and optimization process, thereby improving the model's overall computational efficiency and performance. Finally, the output feature F_O generated by the second-layer cross-attention module captures deeper cross-modal relationships and complex semantic associations. The simplified computation formula is as follows:

$$CrossAttention(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}}) \tag{12}$$

$$F_O = CrossAttention(Q,K,V) + F_E \tag{13}$$

By utilizing two layers of cross-attention and one layer of self-attention, this module achieves multi-level feature fusion and optimization. First, the first layer cross-attention mechanism performs the initial cross-modal fusion, where point cloud features serve as queries and image features serve as keys and values, effectively integrating structural information and semantic details from the image into the point cloud features. This significantly enhances the expressive capability of the point cloud features. Next, the self-attention mechanism further enhances the initially fused features by capturing global contextual information and the inherent relationships between features, deepening and ensuring the consistency of the features, making the fused features more comprehensive and semantically richer. Finally, the second layer cross-attention mechanism reintroduces image features to further refine the fusion of multi-modal features, enhancing both the depth and the fine details of the fusion. This ensures that the final output features not only contain complementary multi-modal information but also possess higher expressiveness and global consistency. Through this multi-level feature fusion strategy, the model can deeply integrate information from different modalities and, through flexible feature interactions and dynamic weight distribution, significantly improve the performance and effectiveness of point cloud completion tasks.

B. Fine-grained Point Cloud Generation Network

To obtain a more refined and complete point cloud model, this paper introduces a refinement process after generating a rough point cloud. Inspired by SVDFormer[54], we propose an innovative structured point cloud recovery module. This module effectively addresses the issues of missing and incomplete data in the rough point cloud through deep refinement and structural reconstruction. Our approach adopts a cascaded dual-module refinement strategy. The first module is primarily responsible for initial optimization of the rough point cloud, focusing on restoring the overall structure and reconstructing key geometric features. Building upon this, the second refinement module further processes the point cloud, particularly enhancing the detail quality and geometric accuracy of local regions. This progressive refinement strategy allows the system to adaptively adjust the refinement level, maintaining global structural accuracy while gradually enhancing local details. Additionally, the synergistic effect of the two refinement modules ensures the continuity and stability of the reconstruction process. The final point cloud model not only maintains high completeness but also presents rich surface details and accurate geometric features.

The structured point cloud recovery module refines the rough point cloud to obtain a high-resolution point cloud, as shown in Figure4. This process is achieved by fine-tuning and upsampling the local geometric structure of the missing point cloud. First, the rough point cloud generated by the rough point cloud generator is concatenated with the input incomplete point cloud, and farthest point sampling (FPS) [32] is applied to generate point cloud F_L. Then, Chamfer Distance (CD) embedding is used to encode the relationship between the input incomplete point cloud and the generated point cloud. This encoding describes the differences between the two in a compact and informative way, capturing their feature discrepancies. Subsequently, the encoded information is fed into a self-attention layer along with the point cloud F_L, helping the network more effectively identify and understand the shape of the missing regions, yielding intermediate features F_G. Next, EdgeConv is used to extract the local structure of the input incomplete point cloud, and a cross-attention mechanism aligns the local features with features F_G. In the cross-attention mechanism, the query

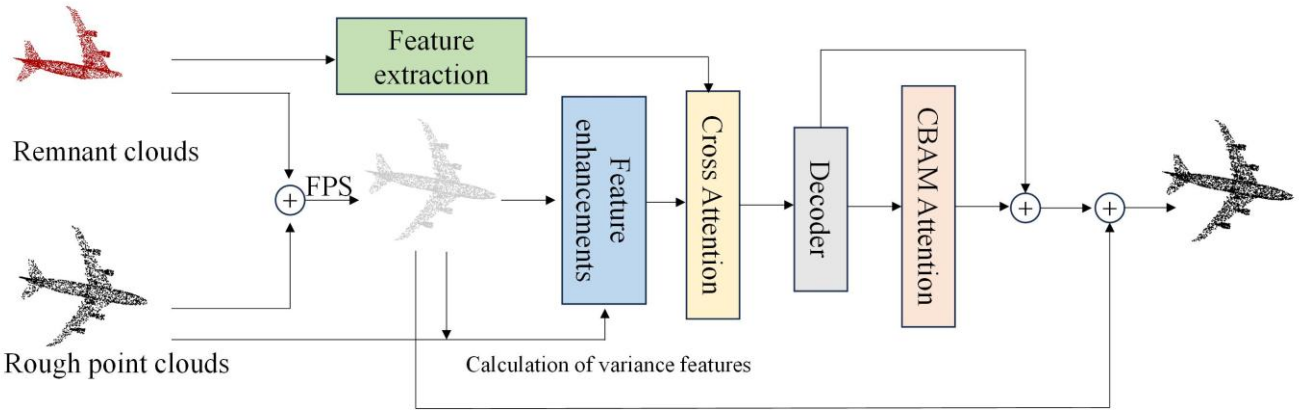

Fig. 4. The structural diagram of the Structured Point Cloud Cecovery Module

matrix comes from the results of the self-attention layer, while the key-value pairs come from the local structure of the incomplete point cloud. This alignment helps model the geometric similarity between the input incomplete point cloud and features F_G , further assisting the refinement operation.

To further improve the precision of point cloud refinement, we introduce Convolutional Block Attention Module (CBAM) to effectively capture and preserve the geometric details of point cloud features. The CBAM module emphasizes the most important feature channels for the refinement task by weighting the importance of each channel and suppressing irrelevant or redundant information, thus improving the accuracy and stability of the point cloud refinement. The CBAM module consists of two steps: channel attention and spatial attention. Finally, FQ is obtained from the decoder.

In channel attention, first, global average pooling and global maximum pooling are applied to the input features F_Q to generate two different context descriptions:

$$F_{avg} = AvgPool(F_Q) \tag{14}$$

$$F_{\max} = MaxPool(F_Q) \tag{15}$$

Where F_{avg} and $F_{\max}$ both have the dimensions of $C \times 1 \times 1$, with C being the number of channels.

Next, these two descriptions are processed through a shared fully connected layer to capture the inter-channel correlations. MLP (Multi-Layer Perceptron) is used to learn the dependencies between different channels and identify which channels are most important for the refinement task. The formula is as follows:

$$MLP(F_{avg}) = \mathrm{Re}\,LU(W_1 F_{avg}) \tag{16}$$

$$MLP(F_{\max}) = \mathrm{Re}\,LU(W_2 F_{\max}) \tag{17}$$

where W_1 and W_2 are the weight matrices of the two fully connected layers. Then, the outputs of the two MLPs are summed and passed through a sigmoid activation function to generate the channel attention weights:

$$M_c = \sigma(W_2(MLP(F_{avg}) + MLP(F_{\max}))) \tag{18}$$

where σ is the Sigmoid function, and M_c has dimensions $C \times 1 \times 1$. Finally, the channel attention weights M_c are applied to F_Q to obtain the enhanced features $F_C^{'}$:

$$F_C^{'} = M_c \otimes F_Q \tag{19}$$

where $\otimes$ denotes the element-wise multiplication. By reducing the weight of unimportant channels, noise and redundant information are minimized, significantly improving the precision and stability of point cloud refinement.

In spatial attention, first, the features are weighted in the spatial dimension to focus on key spatial regions of the point cloud, ensuring that the model attends to points or regions with important geometric information. The enhanced features $F_C^{'}$, after the channel attention, are pooled in the spatial dimension using both average pooling and maximum pooling to generate two spatial descriptions:

$$F_{avg}^{'} = AvgPoolchannel(F_C^{'}) \tag{20}$$

$$F_{\max}^{'} = MaxPoolchannel(F_C^{'}) \tag{21}$$

where $F_{avg}^{'}$ and $F_{\max}^{'}$ have dimensions $1 \times H \times W$, with H and W representing the height and width of the feature map. Then, the average pooling and maximum pooling results are concatenated along the channel dimension to form a $2 \times H \times W$ feature map:

$$F_S^{'} = Concat(F_{avg}^{'} + F_{\max}^{'}) \tag{22}$$

By combining the average pooling and maximum pooling results, both global average information and extreme value information are fused, helping to better understand the important spatial regions and improve the accuracy of locating critical areas. A convolution operation is then applied to the concatenated feature map to capture the spatial dependencies.

Similarly, the weights of each spatial position are generated via the sigmoid activation function:

$$M_s = \sigma(F_S^{'}) \tag{23}$$

where σ is the Sigmoid function, and M_s has dimensions $1 \times H \times W$.

Finally, the spatial attention weights M_s are applied to the channel-attention-enhanced features $F_C^{'}$, yielding the final enhanced features $F_{out}^{'}$:

$$F_{out}^{'} = M_s \otimes F_C^{'} \tag{24}$$

The spatial attention mechanism helps the model accurately locate the key spatial regions in the point cloud, ensuring that these regions receive sufficient attention during the refinement process, thus improving the ability to capture and retain geometric details. The original features and the enhanced features are concatenated and passed through a

mapping layer to obtain the final coordinate offsets, which are then used to generate the refined point cloud.

C. Loss Function and Evaluation Metrics

To evaluate the difference between the generated point cloud and the ground truth P_{gt}, we choose Chamfer Distance (CD) as the loss function. Chamfer Distance is a commonly used metric for point cloud generation and matching tasks, which measures the similarity between two point sets by calculating the distance from each point to its nearest neighbor. In recent years, Chamfer Distance has been widely applied in point cloud completion, generation, and reconstruction tasks due to its strong performance in capturing local structures of point clouds.

Additionally, to support a coarse-to-fine generation process, we regularize the loss function during training. This regularization strategy helps guide the model to maintain global consistency throughout the generation process, while ensuring that details are gradually refined, thus improving the quality and accuracy of the generated point clouds. Through this approach, we are able to balance the recovery of global structure with the fine reconstruction of local details during training. The calculation formula is as follows:

$$L = L_{CD}(P_c, P_{gt}) + L_{CD}(P_i, P_{gt}) \tag{25}$$

Here, $i = 1, 2$. During the calculation of the loss, the ground truth point cloud P_{gt} is downsampled to the same density as the predicted point sets P_c and P_i.

IV. Experiments and Results Analysis

It is recommended that footnotes be avoided (except for the unnumbered footnote with the receipt date on the first page). Instead, try to integrate the footnote information into the text and the reference part.

A. Datasets

The PCN dataset originates from the ShapeNet database and covers eight object categories (such as airplanes, cars, chairs, tables, etc.), comprising over 30,000 pairs of point cloud data. Each object's point cloud data is divided into partial point clouds (2048 points) and complete point clouds (16,384 points). The partial point clouds are generated by simulating common occlusions and noise found in real-world scenarios, representing the kind of incomplete point clouds that sensors might encounter in practical applications. In contrast, the complete point clouds are high-density, lossless representations. The partial point clouds in this dataset are created through the backprojection of 2.5D depth images, effectively mimicking the missing features that sensors experience when capturing point cloud data. Due to its diverse object categories and realistic generation of missing data, the PCN dataset has become a benchmark for evaluating algorithm performance in the field of point cloud completion.

The ShapeNet-55 dataset is a subset of the ShapeNet database, containing point clouds from 55 different object categories, totaling 52,470 point cloud models. The point cloud data in this dataset are generated through automated 3D scanning and manual CAD modeling, ensuring the diversity and high quality of the data. Each point cloud model features a complete point cloud with 8,192 points, while partial point clouds are created by masking out specific areas, retaining 2,048 points. The ShapeNet-55 dataset includes a variety of object categories such as furniture, vehicles, and everyday items, making it suitable for testing the generalization capability of algorithms across a broader range of object types. As one of the largest publicly available 3D datasets, ShapeNet-55 provides a wealth of diverse data, facilitating the study of point cloud completion tasks and aiding in the evaluation of algorithm performance and stability on large-scale datasets. Both datasets are widely used in point cloud completion research for their representativeness and standardization, offering a comprehensive test of algorithm performance on real-world data.

B. Experimental Environment and Parameter Settings

The experiments in this paper are conducted on an Ubuntu 20.04 operating system, with hardware consisting of an i9-13900KF processor, 32GB of RAM, and an NVIDIA GeForce RTX 4060TI GPU. The model is implemented using the PyTorch and CUDA frameworks, and trained with the AdamW optimizer. The initial learning rate is set to 5×10^(-4) , with an appropriate weight decay value.

During training, the batch size for the PCN dataset is set to 32, with a total of 300 epochs. The learning rate is decayed by a factor of 0.9 every 20 epochs. For the ShapeNet-55 dataset, the batch size is adjusted to 48, and training is carried out for 200 epochs. At the end of each epoch, the best model is selected based on the validation set and evaluated on the test set. The network parameters are iteratively updated by optimizing the loss function, and the best model is chosen for final evaluation.

C. Comparative Experimental Analysis

1) *Comparison Results on PCN Dataset:* To comprehensively validate the effectiveness of the algorithm in this chapter, we compare it with several classical and advanced point cloud completion methods. The models for comparison include the baseline model SVDFormer and other models such as PCN [12], GRNet [4], PoinTr [8], SnowflakeNet [3], SDT [36], PMP-Net++ [37], and Seedformer [23]. The experimental results are presented in Table I. The Chamfer Distance (Manhattan norm, cd-l1) and F-Score (F1) are used to evaluate the quality of point cloud completion on the PCN dataset. The bolded values in the table represent the best result for each row.

As shown in Table I, the algorithm proposed in this chapter outperforms other methods in terms of both the mean CD distance and F1 score across eight object categories. In particular, the mean CD distance is 1.2% better than the best-performing SVDFormer model. This indicates that the IGMLNet algorithm proposed in this chapter has stronger capabilities in learning missing geometric features from incomplete point clouds.

To visually demonstrate the performance of different algorithms in handling point cloud data, the completion results of several models are shown in Figure 5. The visualization clearly reflects the differences in detail recovery, global structure reconstruction, and filling of missing regions. The results show that IGMLNet performs the best among all methods. This model not only precisely restores local details but also maintains the coherence of global structure, demonstrating exceptional performance, particularly in completing complex shapes. For instance, in the detailed

regions such as the legs of a chair and the wings of an airplane, the point clouds generated by IGMLNet better preserve the geometric features and avoid over-smoothing or the introduction of artificial structures. Compared to other methods, IGMLNet's completion results are more natural, with better detail recovery and more balanced overall structure. Furthermore, during the completion process, it can accurately restore object edges, concavities, and convexities, and when faced with large missing areas, it can reasonably fill in the gaps, avoiding significant geometric distortions or unnatural transitions. These advantages stem from IGMLNet's innovative design in multi-level feature optimization, enabling it to refine completion results at multiple levels. Overall, the model demonstrates significant advantages in detail recovery, structural coherence, and global consistency, generating more natural and realistic completed point clouds.

TABLE I
CHAMFER DISTANCE OF DIFFERENT MODELS UNDER THE PCN (CD-L1)

Categories	PCN	GRNet	PoinTr	SnowflakeNet	SDT	PMP-Net++	Seedformer	SVDFormer	IGMLNet
plane	5.50	6.45	4.75	4,29	4.60	4.39	3.85	3.62	3.54
Cabinet	22.70	10.37	10.47	9.16	10.05	9.96	9.05	8.79	8.63
Car	10.63	9.45	8.68	8.08	8.16	8.53	8.06	7.46	7.28
Chair	8.70	9.41	9.39	7.89	9.15	8.09	7.06	6.91	6.86
Lamp	11.00	7.96	7.75	6.07	8.12	6.06	5.21	5.33	5.19
Sofa	11.34	10.51	10.93	9.23	10.65	9.83	8.85	8.49	8.35
Table	11.68	8.44	7.78	6.55	7.64	7.17	6.05	5.90	5.72
Watercraft	8.59	8.04	7.29	6.40	7.66	6.52	5.85	5.83	5.78
CD-Avg	12.15	8.83	8.38	7.21	8.24	7.56	6.74	6.54	6.42
F1	0.695	0.708	0.745	0.801	0.754	0.781	0.818	0.841	0.852

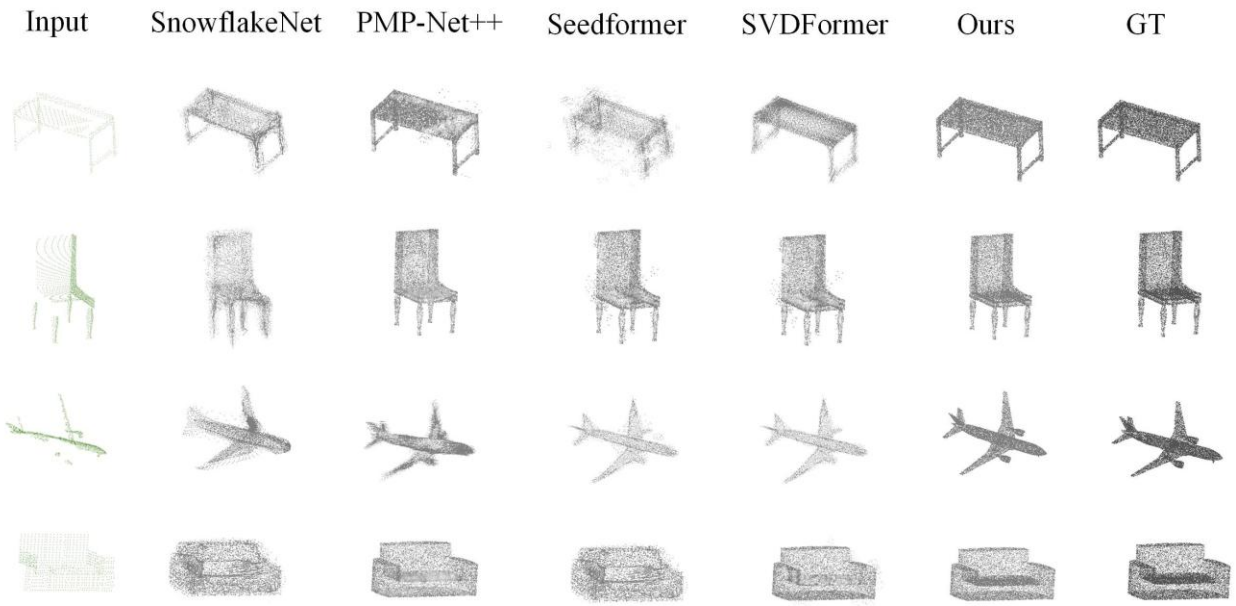

Fig. 5. Visualization results of the PCN datase

TABLE II
COMPARISION RESULTS OF SHAPENET55 DATASET UNDER DIFFERENT DIFFICULTY LEVELS (CD-L2)

	FoldingNet	PCN	GRNet	PoinTr	Seedformer	SVDFormer	IGMLNet
CD-S	2.67	1.94	1.35	0.58	0.50	0.48	0.46
CD-M	2.66	1.96	1.71	0.89	0.77	0.70	0.67
CD-H	4.05	4.08	2.85	1.79	1.49	1.30	1.28
CD-Avg	3.12	3.12	1.97	1.09	0.72	0.83	0.80
F1	0.082	0.133	0.238	0.464	0.472	0.451	0.468

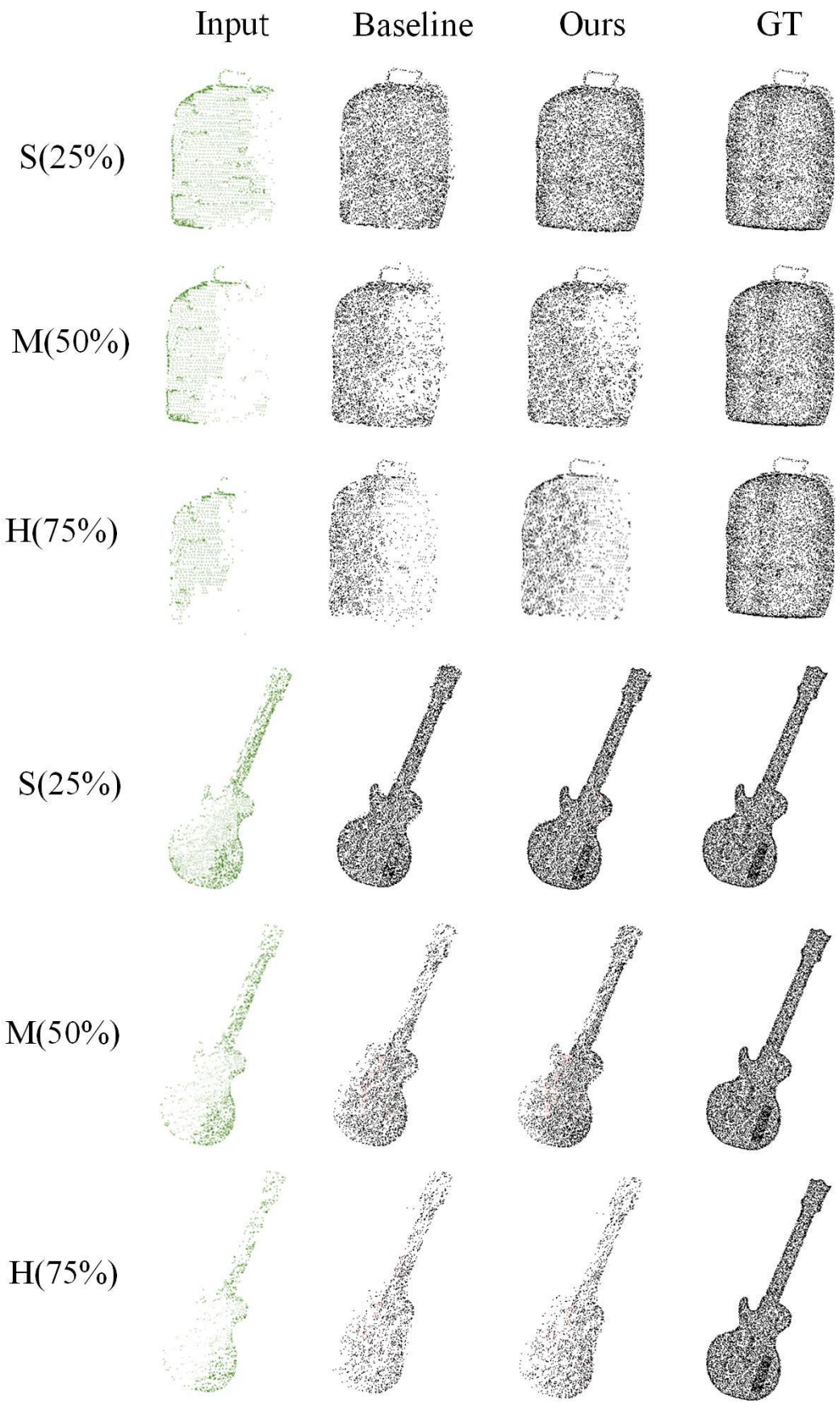

Fig. 6. Visualization results of the ShapeNet55 datase

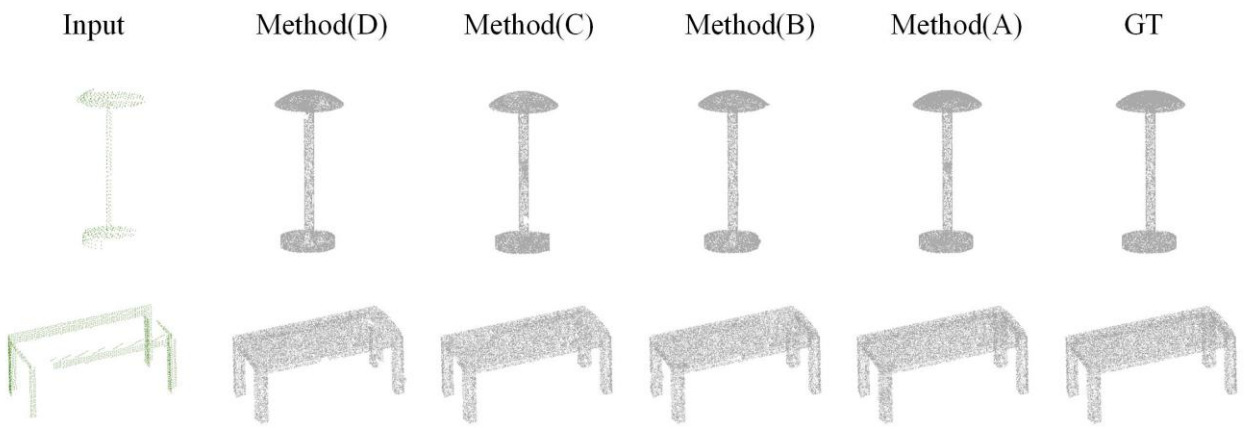

Fig. 7. Visualization results of ablation experiments

2) *Comparison Results on ShapeNet55 Dataset:* The experiments on the ShapeNet-55 dataset categorize the point cloud incompleteness into three difficulty levels: easy (25% missing), medium (50% missing), and hard (75% missing), to provide a clear illustration of the model's performance at different levels of missing data. To ensure a comprehensive and accurate evaluation, Chamfer Distance (Euclidean norm, CD-L2) and F-Score are used as evaluation metrics. The former measures the geometric distance between the completed point cloud and the ground truth, while the latter provides a comprehensive assessment of the accuracy and completeness of the point cloud distribution. The experimental results are shown in Table II.

The results show that the IGMLNet model performs exceptionally well across all difficulty levels, particularly in the hard-level point cloud completion, where both CD and F-Score values significantly outperform those of other models. In the easy setting, the model achieves an excellent CD value of 0.46; in the medium setting, the CD value is 0.67; and in the hard setting, the CD value is 1.28. Although the model's performance decreases with higher levels of incompleteness, it still maintains a high level of accuracy, demonstrating its robustness across different difficulty settings. Additionally, the IGMLNet model achieves an F-Score of 0.468, showing strong recall and effective detail recovery. Comparisons with other models further highlight IGMLNet's advantages across various missing patterns and incompleteness levels. During the completion process, the model excels at recovering global structures and accurately capturing local details, particularly when completing complex geometric shapes. Moreover, as shown in the visual results in Figure 6, the proposed model demonstrates a balanced performance in terms of point cloud uniformity and detail recovery, which contributes to its strong adaptability and superiority in point cloud completion tasks.

D. Ablation Experimental Analysis

To further assess the impact of the efficient residual feature extractor, multi-level feature fusion module, and structured point cloud recovery module on the experimental results, we conducted four ablation experiments on the PCN dataset. In each experiment, one of the aforementioned modules is removed, and the model performance is compared. The final results are then compared with those of the full model. The comparison results of the ablation experiments are shown in Table III.

TABLE III
COMPARISON OF RESULTS OF ABLATION EXPERIMENTS (CD-L1)

Method	Description	CD
(A)	complete model	6.42
(B)	without efficient residual feature extractor	6.68
(C)	without multi-level feature fusion module	6.81
(D)	without structured point cloud recovery module	7.05

The experimental results indicate that removing any of the modules leads to a decline in both model performance and point cloud completion quality. Specifically, removing the efficient residual feature extractor (B) and the multi-level feature fusion module (C) caused the average CD values to increase by 2.6% and 3.9%, respectively. When using the baseline model decoding structure (D), the average CD value increased by 6.3%. The complete IGMLNet model (A) achieved the lowest values across all evaluation metrics. These results demonstrate that the efficient residual feature extractor, multi-level feature fusion module, and structured point cloud recovery module each significantly affect point cloud completion accuracy, confirming their crucial contribution to the network's performance and their mutually reinforcing role.

The visual results of the ablation experiments are shown in Figure 7. Method (D), which lacks the structured point cloud recovery module, results in less smooth predictions, as seen in the airplane and lamp examples. Method (C), which does not use the multi-level feature fusion module to optimize and integrate multi-level features, shows blurred boundaries in parts of the chair and lamp, with less defined local details. Method (B), which only employs the ResNet18 residual network for feature extraction, suffers from the loss of detailed features, leading to poorer completion performance.

V. CONCLUSION

This paper proposes a novel point cloud completion network, IGMLNet. The network uses image guidance and employs an efficient residual feature extractor to deeply extract image features, capturing more prominent spatial structures and details. These enhanced features provide

stronger support for guiding subsequent point cloud features. In terms of feature fusion, IGMLNet utilizes a multi-level feature fusion module that progressively merges point cloud and image features, generating fused features rich in structural information, thereby effectively enhancing the expressive power of the features. To further improve completion performance, the network introduces a cascaded structured point cloud recovery module, which enhances local details while maintaining global structural accuracy, effectively addressing the issues of missing and incomplete data in coarse point clouds.

We conduct comprehensive evaluations of IGMLNet on widely used PCN and ShapeNet55 datasets. Experimental results demonstrate that, compared to existing state-of-the-art methods, the proposed model achieves significant performance improvements in point cloud completion tasks, proving its superiority in practical applications.

REFERENCES

[1] Yin-Yue Nie, Yi-Qun Lin, Xiao-Guang Han, Shi-Hui Guo, Jian Chang, Shu-Guang Cui, and J. Zhang, "Skeleton-bridged point completion: From global inference to local adjustment," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020, pp. 16119–16130.

[2] Xin Wen, Peng Xiang, Zhi-Zhong Han, Yan-Pei Cao, Peng-Fei Wan, Wen Zheng, and Yu-Shen Liu, "PMP-Net: Point cloud completion by learning multi-step point moving paths," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 7443–7452.

[3] Piang Xiang, Xin Wen, Yu-Sheng Liu, Yan-Pei Cao, Peng-Fei Wan, Wen Zheng, "SnowflakeNet: Point cloud completion by snowflake point deconvolution with skip-transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2021, pp. 5499–5509.

[4] Hao-Zhe Xie, Hong-Xun Yao, Shang-Chen Zhou, Jia-Geng Mao, Sheng-Ping Zhang, and Wen-Xiu Sun, "GRNet: Gridding residual network for dense point cloud completion," in *Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 365–381.

[5] Xu-Min Yu, Yong-Ming Rao, Zi-Yi Wang, Zu-Yan Liu, Ji-Wen Lu, and Jie Zhou, "PoinTr: Diverse point cloud completion with geometry-aware transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2021, pp. 12498–12507.

[6] Feng Han and Song-Chun Zhu, "Bottom-up/top-down image parsing with attribute grammar," *IEEE Trans*. Pattern Anal. Mach. Intell., vol. 31, no. 1, pp. 59–73, Jan. 2008.

[7] M. Pauly, N. J. Mitra, J. Giesen, M. H. Gross, and L. J. Guibas, "Example-based 3D scan completion," in *Symp. Geometry Process.*, 2005, pp. 23–32.

[8] Tian-Jia Shao, Wei-Wei Xu, Kun Zhou, Jing-Dong Wang, Dong-Ping Li, and Bai-Ning Guo, "An interactive approach to semantic modeling of indoor scenes with an RGBD camera," *ACM Trans. Graph. (TOG)*, vol. 31, no. 6, pp. 1–11, Nov. 2012.

[9] E. Kalogerakis, S. Chaudhuri, D. Koller, and V. Koltun, "A probabilistic model for component-based shape synthesis," *ACM Trans. Graph. (TOG)*, vol. 31, no. 4, pp. 1–11, Oct. 2012.

[10] V. G. Kim, W. Li, N. J. Mitra, S. Chaudhuri, S. DiVerdi, and T. Funkhouser, "Learning part-based templates from large collections of 3D shapes," *ACM Trans. Graph. (TOG)*, vol. 32, no. 4, pp. 1–12, Aug. 2013.

[11] A. Martinovic and L. Van Gool, "Bayesian grammar learning for inverse procedural modeling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2013, pp. 201–208.

[12] Wen-Tao Yuan, T. Khot, D. Held, C. Mertz, and M. Hebert, "PCN: Point completion network," in *Proc. 2018 Int. Conf. 3D Vision (3DV)*, Verona, Italy, 2018, pp. 728–737.

[13] Zi-Tian Huang, Yu-Kuan Yu, Jia-Wen Xu, Feng Ni, and Xin-Yi Le, "PF-Net: Point fractal network for 3D point cloud completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, 2020, pp. 7662–7670.

[14] Wen-Xiao Zhang, Qing-An Yan, and Chun-Xia Xiao, "Detail preserved point cloud completion via separated feature aggregation," in *European Conf. Comput. Vis. (ECCV)*, 2020, pp. 512–528.

[15] Xue-Jun Yan, Hong-Yu Yan, Jing-Jing Wang, Hang Du, Zhi-Hong Wu, Di Xie, Shi-Liang Pu, and Li Lu, "FBNet: Feedback network for point cloud completion," in *European Conf. Comput. Vis. (ECCV)*, 2022, pp. 676–693.

[16] Liang Pan, "ECG: Edge-aware Point Cloud Completion with Graph Convolution," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4392–4398, 2020.

[17] Liang Pan, Xin-Yi Chen, Zhou-Gang Cai, Jun-Zhe Zhang, Hai-Yu Zhao, Shuai Yi, and Zi-Wei Liu, "," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 8524–8533.

[18] X. Wang, M. H. Ang, and G. Lee, "Cascaded refinement network for point cloud completion with self-supervision," *IEEE Trans. Pattern Anal. Mach. Intell*., vol. 7, no.3, pp.112–51, 2021.

[19] H. Mittal, B. Okorn, A. Jangid, and D. Held, "Self-supervised point cloud completion via inpainting," in British Mach. Vis. Conf., 2021, pp. 22–16.

[20] X. Zhang, Y. Feng, S. Li, C. Zou, H. Wan, X. Zhao, Y. Guo, and Y. Gao, "View-guided point cloud completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit*., 2021, pp. 15890–15899.

[21] X. Zhang, Y. Feng, S. Li, C. Zou, H. Wan, X. Zhao, Y. Guo, and Y. Gao, "Vew-guided point cloud completion,"in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15890–15899.

[22] L. Pan, X. Chen, Z. Cai, J. Zhang, H. Zhao, S. Yi, and Z. Liu, "Variational relational point completion network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8524–8533.

[23] H. Zhou, Y. Cao, W. Chu, J. Zhu, T. Lu, Y. Tai, and C. Wang, "Seedformer: Patch seeds based point cloud completion with upsample transformer," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 416–432.

[24] A. Dai, C. Ruizhongtai Qi, and M. Nießner, "Shape completion using 3D-encoder-predictor CNNs and shape synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*., 2017, pp. 5868–5877.

[25] X. Han, Z. Li, H. Huang, E. Kalogerakis, and Y. Yu, "High-resolution shape completion using deep neural networks for global structure and local geometry inference," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 85–93.

[26] D. Stutz and A. Geiger, "Learning 3D shape completion from laser scan data with weak supervision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1955–1964.

[27] Xiao-Gang Wang, M. H. Ang, and G. H. Lee, "Voxel-based network for shape completion by leveraging edge generation," in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2021, pp. 13189–13198.

[28] Y. Yang, C. Feng, Y. Shen, and D. Tian, "FoldingNet: Point cloud auto-encoder via deep grid deformation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 206–215.

[29] M. Liu, L. Sheng, S. Yang, J. Shao, and S. M. Hu, "Morphing and sampling network for dense point cloud completion," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, 2020, pp. 11596–11603.

[30] E. Aiello, D. Valsesia, and E. Magli, "Cross-modal learning for image-guided point cloud shape completion," in *Advances in Neural Information Processing Systems*, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, Eds., 2022, pp. 1–10

[31] S. Y. Huang, H.-Y. Hsu, and Y.-C. F. Wang, "SPoVT: Semantic-prototype variational transformer for dense point cloud semantic completion," in *Advances in Neural Information Processing Systems*, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, Eds., 2022.

[32] T. Hu, Z. Han, A. Shrivastava, and M. Zwicker, "Render4Completion: Synthesizing multi-view depth maps for 3D shape completion," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops*, 2019, pp. 0–0.

[33] C. Xie, C. Wang, B. Zhang, H. Yang, D. Chen, and F. Wen, "Style-based point generator with adversarial rendering for point cloud completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 4619–4628.

[34] Wen-Xiao Zhang, Zhen Dong, Jun Liu, Qin-Gan Yan, Chun-Xia Xiao, et al. "Point cloud completion via skeleton-detail transformer," I*EEE Transactions on Visualization and Computer Graphics*, vol. 2, no. 5, pp. 10-11, 2022.

[35] Zhe Zhu, Hong-Hua Chen, Xing He, et al., "SVDformer: Complementing point cloud via self-view augmentation and self-structure dual-generator," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 14508–14518.

[36] Wenxiao Zhang, Zhen Dong, Jun Liu, Qingan Yan, Chunxia Xiao, et al. "Point cloud completion via skeleton-detail transformer," in *IEEE Transactions on Visualization and Computer Graphics*, 2022, pp.1322

[37] Xin Wen, Peng Xiang, Zhizhong Han, Yan-Pei Cao, Pengfei Wan, Wen Zheng, et al., "PMP-Net++: Point Cloud Completion by Transformer-Enhanced Multi-step Point Moving Paths," in *IEEE Transactions on Pattern Analysis and Ma- chine Intelligence*, 2022, pp. 0–0.