

# Comprehensive Review of Deep Learning-Based 3D Point Cloud Completion Processing and Analysis

Ben Fei<sup>✉</sup>, Graduate Student Member, IEEE, Weidong Yang, Wen-Ming Chen<sup>✉</sup>, Member, IEEE,  
Zhijun Li<sup>✉</sup>, Fellow, IEEE, Yikang Li, Tao Ma, Xing Hu, and Lipeng Ma

**Abstract**—Point cloud completion is a generation and estimation issue derived from the partial point clouds, which plays a vital role in the applications of 3D computer vision. The progress of deep learning (DL) has impressively improved the capability and robustness of point cloud completion. However, the quality of completed point clouds is still needed to be further enhanced to meet the practical utilization. Therefore, this work aims to conduct a comprehensive survey on various methods, including point-based, view-based, convolution-based, graph-based, generative model-based, transformer-based approaches, etc. And this survey summarizes the comparisons among these methods to provoke further research insights. Besides, this review sums up the commonly used datasets and illustrates the applications of point cloud completion. Eventually, we also discussed possible research trends in this promptly expanding field.

**Index Terms**—Deep learning, point cloud, completion, 3D vision.

## I. INTRODUCTION

WITH the popularity of 3D scanning devices, including LiDAR, laser or RGB-D scanners, and so on, point clouds have become easier to capture and currently provoked a great deal of researches in the fields of robots, autonomous driving, 3D modeling and fabrication. However, the raw point clouds directly collected by these devices are primarily sparse and partial due to the occlusions, reflections, transparency, and restriction of devices' resolution and angles. Therefore, generating the complete point clouds from partial observations is crucial to boost the downstream applications.

Manuscript received 3 March 2022; revised 4 June 2022 and 21 July 2022; accepted 27 July 2022. Date of publication 16 August 2022; date of current version 5 December 2022. This work was supported by the Natural Science Foundation of China under Grant U2033029. The Associate Editor for this article was S.-H. Kong. (Corresponding authors: Wen-Ming Chen; Weidong Yang.)

Ben Fei, Weidong Yang, and Lipeng Ma are with the Shanghai Key Laboratory of Data Science, School of Computer Science, Fudan University, Shanghai 200433, China (e-mail: bfei21@m.fudan.edu.cn; wdyang@fudan.edu.cn; lpma21@m.fudan.edu.cn).

Wen-Ming Chen is with the Academy for Engineering and Technology, Fudan University, Shanghai 200433, China (e-mail: chenwm@fudan.edu.cn).

Zhijun Li is with the Department of Automation, University of Science and Technology of China, Hefei 230060, China (e-mail: zjli@ieee.org).

Yikang Li and Tao Ma are with the Shanghai AI Laboratory, Shanghai 200032, China (e-mail: liyikang@pjlab.org.cn; matao@pjlab.org.cn).

Xing Hu is with the School of Optical-Electrical Information and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China (e-mail: huxing@usst.edu.cn).

This article has supplementary downloadable material available at <https://doi.org/10.1109/TITS.2022.3195555>, provided by the authors.

Digital Object Identifier 10.1109/TITS.2022.3195555

The effectiveness of point cloud completion lies in its distinct and crucial role in various computer vision applications.

**3D reconstruction.** The generation of complete 3D scenes is the foundation and important technology for numerous computer vision tasks, including 3D map reconstruction with high-resolution in autonomous driving, 3D reconstruction in robotics, and underground mining. For instance, point cloud completion in robot applications can help route planning and decision-making by constructing a 3D scene. Moreover, the large 3D environment reconstruction in underground mining space to accurately monitor mining safety. **3D detection.** The 3D object detection relies on complete point clouds to keep state-of-the-art (SOTA) performance. For example, cars in the distance captured by LiDAR tend to be sparse and often difficult to detect. It is noted that we often need to completely segment the target point cloud to complete it. For instance, when carrying the 3D detection on the KITTI dataset, the point clouds of vehicles need to be segmented and normalized to their scale and orientation before the complete point cloud of the vehicle can be obtained. **3D shape classification.** For 3D shape classification, the complete point clouds are ultimately needed by recovering from partial observations. The partial point cloud represents a small part of the object that is usually hard to recognize. Because point cloud completion plays an essential role in plenty of practical computer vision applications, there is an instant necessity for an extensive investigation of point cloud completion.

However, there are a few surveys on the completion of point cloud and downstream tasks, while the latest progress of deep learning in point cloud completion is urgently needed to be reviewed [1]–[10]. To stimulate the developments of point cloud completion in industry and academy, we conducted a comprehensive review by summing up the rapid growth of point cloud completion technology in recent years (2017–2022), mainly including current deep-learning methods. Furthermore, we give comparisons among various deep-learning techniques.

Over the past several years, researchers have tried numerous methods to address this issue in deep learning. Early attempts on point cloud completion [12]–[17] tried to transfer mature methods from 2D completion tasks to 3D point clouds through voxelization and 3D convolution. However, these methods suffer from high computational costs with increasing spatial resolution. With the tremendous success of PointNet and PointNet++ [18], [19], direct processing of

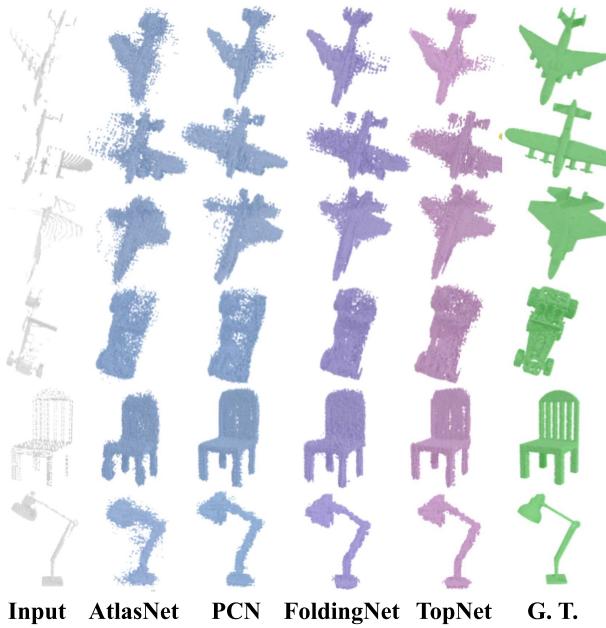


Fig. 1. Schematic diagram of completion results of commonly used point cloud algorithms [11].

3D coordinates have become the mainstream of point cloud-based 3D analysis. This technique is further applied in many pioneering works of point cloud completion [20]–[27], where an encoder-decoder scheme is designed to produce complete point clouds. In recent years, many other methods, such as point-based, view-based, convolution-based, graph-based, generative model-based, and transformer-based methods, have been springing up and achieved significant results (Fig. 1).

Contrasted with the existing papers, the main contributions of the survey could be concluded as follows:

- As far as we know, it is the first survey systematically covering nearly all DL methods on point cloud completion.
- This review introduces the latest and advanced progress in point cloud completion, together with their methods and contributions.
- Systematic comparisons of existing DL methods on a few public datasets, along with compact conclusions and profound discussions, are provided.
- We will discuss future research on DL-based point cloud completion at the end of this survey to stimulate improvement in this field.

## II. REASONS FOR INCOMPLETE POINT CLOUDS

**Definitions:** A point cloud is a set of data points in space representing a 3D shape or an object. Point clouds are generally produced by 3D scanners or by photogrammetry software, consisting of a large number of points that geometrically represent the 3D surfaces of objects. A partial or incomplete point cloud means a point cloud that has missing points. The missing of points means a part of points in point clouds is missing for various reasons.

During data acquisition, the 3D laser scanner will be affected by the characteristics of the measured object, the processing method, and the environment, inevitably leading

to the missing of points (Fig. 2). As shown in Fig. 3, the main reasons could be attributed to specular reflection, signal absorption, occlusion of external objects, self-occlusion of objects, and blind spots. The former two are due to the surface material of the objects, which might absorb or reflect the LiDAR signal in an unexpected way. The latter three are mainly due to occlusion, which can be completed with the aid of other parts of the objects or by utilizing multi-source data. Moreover, the stability of the 3D scanner in the scanning process also has a particular influence on the scanning quality.

After the data collection is completed, the point clouds also need to carry out a series of processing, such as point cloud denoising, smoothing, registration, and fusion. At the same time, these operations will significantly exacerbate the missing of points. This will not only affect the data integrity and lead to topology errors but also affect the quality of the point cloud refactoring, the 3D model reconstruction, local spatial information extraction, and subsequent processing.

## III. CHALLENGES

### A. Structural Information Challenges

The reconstruction of a complete point cloud is challenging because the structural information required for the task of point cloud completion runs counter to the disordered and unstructured nature of the point clouds. 3D object point clouds in the real world could be regarded as low-level and high-level configurations, including surfaces, semantic parts, geometrical elements, etc. Existing point cloud generation frameworks either exclude structure in their devised solutions or assume and execute a particular structure/topology for generating the complete point clouds of 3D objects, for example, a set of surfaces or manifolds. Hence, learning the structural features of the point cloud becomes critical for a better complete point cloud.

### B. Fine-Grained Complete Shapes Challenges

3D shape completion is supposed to reconstruct a reasonable fine-grained complete point cloud using relational structure information, such as geometrical symmetry, regular arrangements, and surface smoothness. Although several works have been fully exploiting the structure information through iterative refinement [28], integration of global features and local features [29], skip-connections [30], residual connections [31], etc., more efforts should be paid to generating the fine-grained complete shapes.

Therefore, this review will investigate the SOTA completion's performance and discuss the solutions they used in tackling these two significant challenges.

## IV. DATASETS

As for 3D shape completion, the datasets could be categorized into two types: artificial datasets and real-world datasets (Table I). The most studied four datasets are as follows:

**PCN** [32]: The Computer-Aided Design (CAD) dataset derived from PCN [25], totally containing 30974 3D models from 8 specific categories. The ground truth point clouds consist of 16384 uniformly sampled on the surfaces.

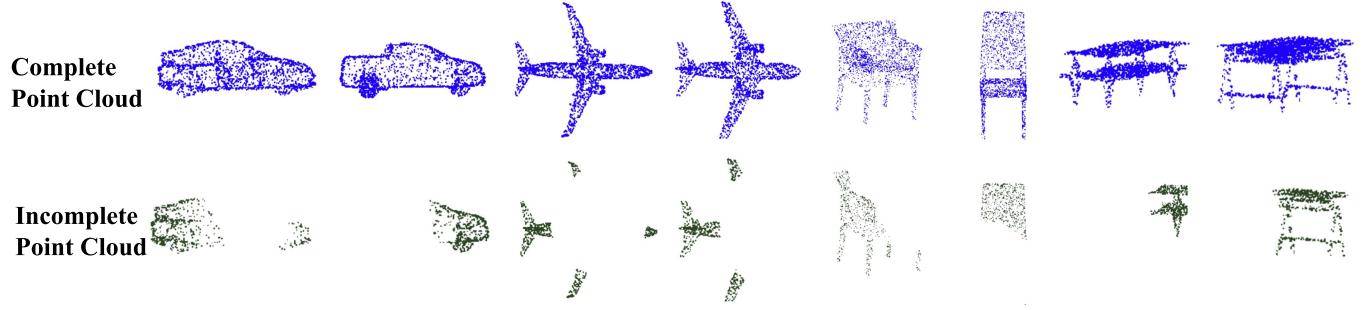


Fig. 2. Schematic of complete point cloud and missing 70% point cloud.

TABLE I  
SUMMARY OF EXISTING DATASETS FOR POINT CLOUD COMPLETION

Name	Year	Classes	Sensors or origin	Type	Views	Resolutions	Description
PCN [32]	2015	8	CAD	Synthetic	8	2048/4096/ 8192/16384	Derived from ShapeNet.
ShapeNet55 [33]	2021	55	CAD	Synthetic	all possible views	8192	Contains all the objects in ShapeNet from 55 categories.
ShapeNet34 [33]	2021	34	CAD	Synthetic	all possible views	8192	Contains 21 unseen categories and 34 seen categories.
KITTI [34]	2012	8	RGB & LiDAR	Urban (Driving)	-	depends on the network	Derived from KITTI, real-world LiDAR scans. Sparse in nature.
ModelNet [35]	2015	10 or 40	CAD	Synthetic	12	2048/16384	Proposed by princeton.
Completion3D [24]	2019	8	CAD	Synthetic	-	1024/2048/16384	Derived from ShapeNet. The partial 3D shapes are generated by back-projected 2.5D depth images from partial views into 3D space.
Multi-View Partial Point Cloud (MVP) [36]	2021	16	CAD	Synthetic	26	2048/4096/16384	Diversity of uniform views; Large-scale and high-quality; Rich categories.
Single-View Point Cloud Completion [36]	2021	16	CAD	Synthetic	1	2048/4096/16384	A large subset of MVP

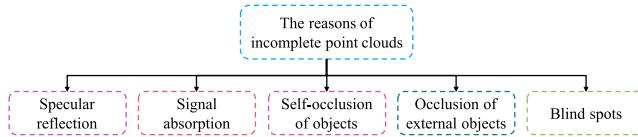


Fig. 3. Reasons for incomplete point clouds.

KITTI [34]: The dataset is collected through a Velodyne laser scanner. The odometry dataset was originally designed to evaluate the performance of stereo matching, which contains LiDAR point clouds of 22 stereo sequences. The partial cars were collected to evaluate the performance of point cloud completion methods in real-world scans in the absence of ground truth.

ModelNet40 [35]: A comprehensive set of 3D CAD models. Its objects consist of 40 categories and 13356 models.

Completion3D [24]: An online platform for evaluating shape-completion methods on the basis of a subset derived from the ShapeNet dataset. It is noted that some paired point clouds face the problem of unmatched scales, which can be tackled by the one-side CD loss proposed in [37].

In addition to the above datasets, the Shapenet 34/55 [33] and MVP datasets [36] have recently been proposed to increase the variety and number of objects, diverse viewpoints, and varying degrees of defects as close as possible to real-world objects.

## V. METRICS

For 3D point cloud completion, the Chamfer Distance (CD) [38] and Earth Mover's Distance (EMD) [38] are the

most frequently used performance criteria. CD tries to find the minimum distance between two sets of points, while EMD evaluates the reconstruction quality of the point clouds.

### A. Chamfer Distance (CD)

$$CD(S_1, S_2) = \frac{1}{|S_1|} \sum_{x \in S_1} \min_{y \in S_2} \|x - y\|_2 + \frac{1}{|S_2|} \sum_{y \in S_2} \min_{x \in S_1} \|y - x\|_2 \quad (1)$$

By definition, CD represents the sum of the average closest distance from points in the output  $S_1$  to points in the complete point clouds  $S_2$ , and from points in  $S_2$  to points in  $S_1$ . There are two variants to CD: CD-T (CD- $\ell_1$ ) and CD-P (CD- $\ell_2$ ). The definitions of CD-T and CD-P between two point clouds  $S_1$  and  $S_2$  are as follows:

$$\begin{aligned} \mathcal{L}_{S_1, S_2} &= \frac{1}{|S_1|} \sum_{x \in S_1} \min_{y \in S_2} \|y - x\|_2^2 \\ \mathcal{L}_{S_2, S_1} &= \frac{1}{|S_2|} \sum_{y \in S_2} \min_{x \in S_1} \|x - y\|_2^2 \\ \mathcal{L}_{CD-T}(S_1, S_2) &= \mathcal{L}_{S_1, S_2} + \mathcal{L}_{S_2, S_1} \\ \mathcal{L}_{CD-P}(S_1, S_2) &= (\sqrt{\mathcal{L}_{S_1, S_2}} + \sqrt{\mathcal{L}_{S_2, S_1}}) / 2 \end{aligned} \quad (2)$$

### B. Earth Mover's Distance (EMD)

EMD aims to find out a bijection  $\phi : S_1 \rightarrow S_2$  to minimize the average distance between corresponding points from partial

and complete ones. Different from CD, the size of  $S_1$  and  $S_2$  needs to be same.

$$EMD(S_1, S_2) = \min_{\phi: S_1 \rightarrow S_2} \frac{1}{|S_1|} \sum_{x \in S_1} \|x - \phi(x)\|_2 \quad (3)$$

#### C. Fidelity Error (FD), Maximum Mean Discrepancy (MMD) and Consistency

Fidelity error (FD), Consistency, and minimal matching distance (MMD) are proposed by PCN as evaluation metrics [25]. Fidelity is utilized to measure how well the input is preserved, which calculates the average distance between the point in the input and the corresponding nearest neighbor in the output. MMD is used to measure how much the model's outputs reconstruct a typical car. Consistency aims to estimate how consistent the model's outputs are against variations in the inputs.

#### D. Density-Aware Chamfer Distance (DCD)

DCD [39] is derived from CD and it can detect disparity of density distributions. DCD pays attention to both the overall structure and local geometric details.

$$\begin{aligned} DCD(S_1, S_2) = & \frac{1}{2} \left( \frac{1}{|S_1|} \sum_{x \in S_1} \left( 1 - \frac{1}{n_{\hat{y}}} e^{-\alpha \|x - \hat{y}\|_2} \right. \right. \\ & \left. \left. + \left( \frac{1}{|S_2|} \sum_{y \in S_2} \left( 1 - \frac{1}{n_{\hat{x}}} e^{-\alpha \|\hat{x} - y\|_2} \right) \right) \right) \right) \end{aligned} \quad (4)$$

#### E. F-Score

The F-score proposed by Tatarchenko *et al.* [40] is to evaluate the distance between the surfaces of the object and is treated as the harmonic mean between precision and recall. The precision counts the percentage of reconstructed points within a certain distance to the ground truth, which stands for the accuracy of the reconstruction. On the other hand, the recall counts the percentage of points on the ground truth within a certain distance to the reconstruction, representing the completeness of the reconstruction. The distance threshold  $d$  can be leveraged to control the strictness of the F-score. The F-score can evaluate the percentage of points or surface area that are reconstructed correctly, which can be defined as follows:

$$F\text{-Score}(d) = \frac{2P(d)R(d)}{P(d) + R(d)} \quad (5)$$

where  $P(d)$  and  $R(d)$  denote the precision and recall for a distance threshold  $d$ , respectively.

$$P(d) = \frac{1}{|S_1|} \sum_{r \in S_1} \left[ \min_{t \in S_2} \|t - r\| < d \right] \quad (6)$$

$$R(d) = \frac{1}{|S_2|} \sum_{t \in S_2} \left[ \min_{r \in S_1} \|t - r\| < d \right] \quad (7)$$

where  $S_1$  is a reconstructed point set being evaluated and  $S_2$  is the ground truth.  $|S_1|$  and  $|S_2|$  are the numbers of points of  $S_1$  and  $S_2$ , respectively.

#### F. Uniformity

Uniformity [41] is usually employed to evaluate the distribution uniformity of the completed point clouds, which can be formulated as:

$$\text{Uniformity}(p) = \frac{1}{M} \sum_{i=1}^M U_{\text{imbalance}}(Q_i) U_{\text{clutter}}(Q_i) \quad (8)$$

where  $Q_i (i = 1, 2, \dots, M)$  is a point subset cropped from a patch of the output  $S_1$  using the farthest sampling and ball query of radius  $\sqrt{p}$ . The term  $U_{\text{imbalance}}$  and  $U_{\text{clutter}}$  account for the global and local distribution uniformity, respectively.

$$U_{\text{imbalance}}(Q_i) = \frac{(|Q_i| - \hat{n})^2}{\hat{n}} \quad (9)$$

where  $\hat{n} = p|S_1|$  is the expected number of points in  $Q_i$ .

$$U_{\text{clutter}}(Q_i) = \frac{1}{|Q_i|} \sum_{j=1}^{|Q_i|} \frac{(d_{i,j} - \hat{d})^2}{\hat{d}} \quad (10)$$

where  $d_{i,j}$  represents the distance to the nearest neighbor for the  $j$ -th point in  $Q_i$ , and  $\hat{d}$  is roughly  $\sqrt{\frac{2\pi p}{|Q_i|\sqrt{3}}}$  if  $Q_i$  has a uniform distribution.

## VI. METHODS

As shown in Fig. 4, based on the network structure employed in point cloud completion and generation, the existing architectures could be categorized into point-based, view-based, convolution-based, graph-based, transformer-based, generative model-based, and other methods. Nearly all milestone contributions are clearly illustrated in Fig. 5. Since most works are hybrid methods, they might belong to several methods according to their stated highlights. Therefore, we will discuss these works if they utilized the methods that can be categorized into these classes.

### A. Point-Based Methods

The point-based approaches usually model every point independently by utilizing Multi-layer Perceptrons (MLPs). The global feature is then aggregated through a symmetric function (such as Max-Pooling) because of the transformation invariance of the point clouds. Whereas the geometric information and correlations in the whole point group are still not entirely considered. As a commonly used method for processing the features, we only review the methods mainly using point-based networks in this section.

*1) Preliminary Works:* Pioneered by PointNet [18], a few works used MLP for the processing and recovering of the point clouds due to its concise and non-negligible ability of representations [33], [41]–[43]. PointNet++ [19] and TopNet [24] incorporated a hierarchical structure to take the geometric information into consideration. PointNet++ proposed two set abstraction layers, which intelligently aggregate multi-level information, while TopNet proposed a new decoder that generates a structured point cloud without supposing any particular structure or topology. Inspired by PointNet and PointNet++, Yu *et al.* [44] proposed PU-Net learn multi-scale

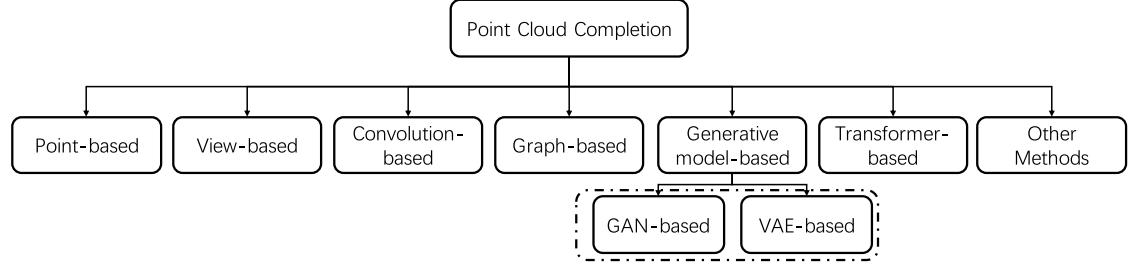


Fig. 4. Taxonomy of point cloud completion.

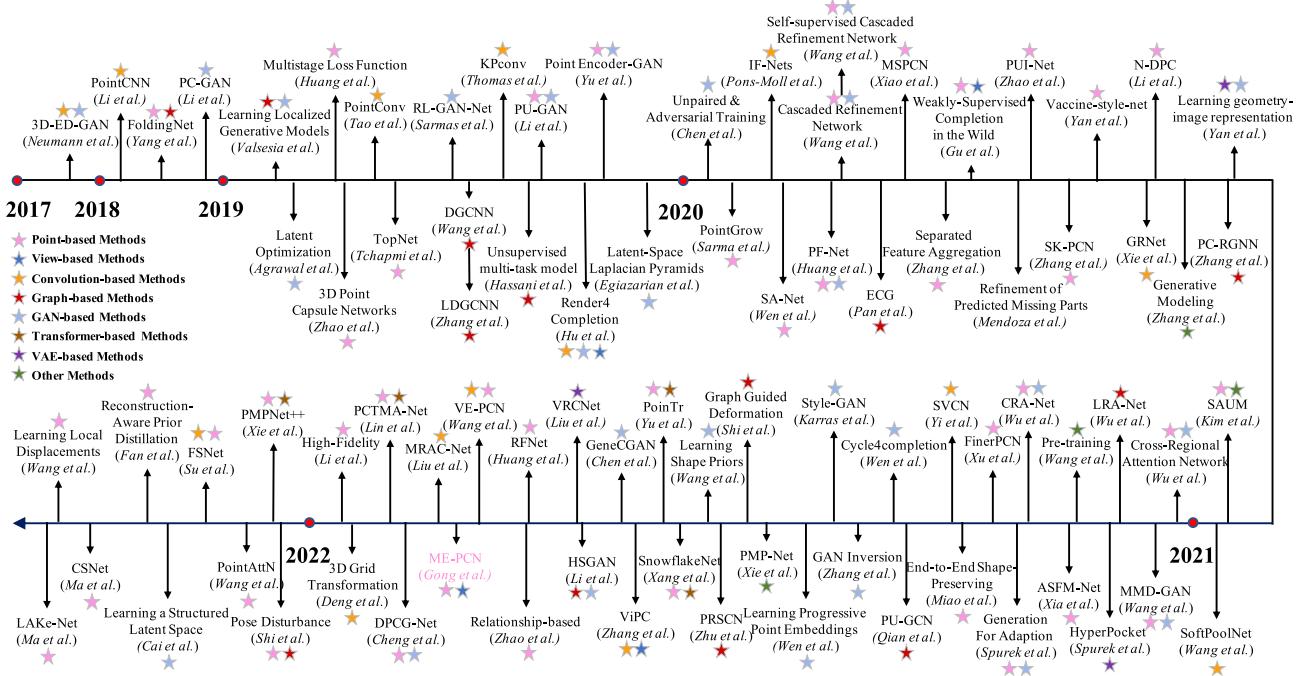


Fig. 5. Chronological summarization of the recent and most relevant DL-based methods for point cloud completion.

features by feature scaling based on sub-pixel convolutional layers. The scaling restoration method performs convolution with  $1 \times 1$  kernels on extracted features. Then, the extracted features are decomposed and reconstructed into a cluster of up-sampling points. And the joint loss function is utilized to distribute the generated point cloud on the potential surface evenly. Nevertheless, PU-Net is primarily designed to generate a single denser point cloud from a sparse point cloud rather than perform point cloud completion. It cannot fill large holes and missing parts, nor can it add meaningful points to heavily down-sampled parts of the point cloud.

To mitigate the structure loss brought by MLP, the proposed AtlasNet [20], and MSN [31] reconstruct the complete output through evaluating a set of parametric surface elements, from which the complete point clouds could be generated. Specifically, AtlasNet [20] took an additional input of a 2D point in the unit square and applied it to produce a single point on the surface. Thus, the output is a continuous image of a plane. This method could be reiterated many times to reconstruct a 3D shape from a combination of numerous surface elements. For avoiding structural loss, MSN [31] introduced the morphing-based decoder that can morph the unit squares as a set of surface elements aggregated into the coarse point cloud.

**2) PCN-Derived Methods:** For the first time, Hebert *et al.* [25] proposed a learning-based shape completion method, Point Completion Network (PCN). Unlike existing approaches, PCN straightly works on original point clouds and does not require any assumption of structural (such as symmetry) or annotation about the underlying shape (such as semantic class). It features a decoder design that allows the generation of fine-grained completions while maintaining a small number of parameters. By combining with PCN and point-wise convolution, Chang *et al.* [45] devised FinerPCN to generate complete and fine point clouds in a coarse-to-fine manner by taking into account local information. After that, Nie *et al.* [46] proposed a skeleton-bridged point completion network (SK-PCN). The SK-PCN features a 3D skeleton that is predicted to learn global information. Following that, the surface is completed by using displacements with skeletal points. In MSPCN, Zhang *et al.* [47] used a tandem of up-sampling modules to reconstruct fine-grained output and supervised each stage to generate outputs with more information and beneficial intermediate for the following phase. Furthermore, they proposed a method to identify critical sets (MVCs) for supervision by combining selected points with max-pooling and volume-down-sampling points. This MVCs could take the vital features and the overall

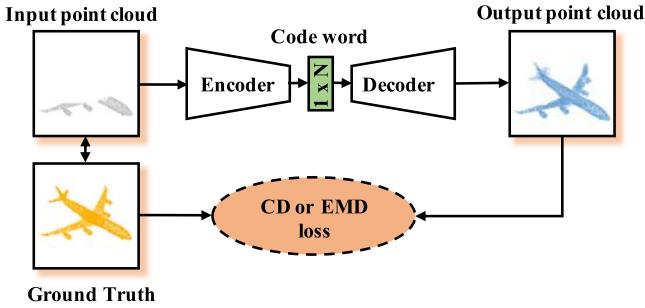


Fig. 6. The illustration of end-to-end network for point clouds completion.  $N$  represents the dimension of latent space.

shape into consideration. By regarding partial point clouds as different samples of the same distribution, the Structure Retrieval based Point Completion Network (SRPCN) [10] was devised. SRPCN uses k-means clustering to extract structure points and disperses them into distributions, and then KL Divergence is used as a metric to find the complete structure point cloud that best matches the input in a database. To obtain a good feature representation that can capture both global structure and local geometric details, FSNet [6] was presented to adaptively aggregate point-wise features into a 2D structured feature map by learning multiple latent patterns from local regions. Then, FSNet is integrated into a coarse-to-fine pipeline for point cloud completion.

3) *End-to-End Mechanism*: In point-based methods, the end-to-end manner is widely used in the network architecture. In the encoder-decoder scheme (Fig. 6), the encoder in completion architecture aims at extracting both the global 3D shape features and the regional features of each point. At the same time, the decoder generates a completion point cloud and refines it. Xia *et al.* [51] devised an S2UNet network to reconstruct a more uniform and fine-grained structure from a sparse point cloud in the applications of vehicles in an end-to-end manner. Significantly, they adopt an up-sampling approach to generate a more uniform point cloud. Furthermore, they devised ASFM-Net [52], in which the asymmetrical Siamese auto-encoder (AE) generates a coarse but complete output, and the following refinement unit aims to recover a final point cloud with fine-grained details. Mendoza *et al.* [53] proposed a network with an end-to-end pattern consisting of two neural networks: the missing part prediction network and the merging-refinement network. This method predicts and integrates missing parts while preserving the existing geometry and refining the details. Miao *et al.* [54] presented a shape-preserving completion network to maintain the 3D shape and recover the fine-grained information of the reconstructed 3D shapes by designing an encoder-decoder scheme. This shape-preserving network could learn the global features and integrate regional information of adjacent points with various orientations and scales. During the decoding process, the information would be fused into latent vectors. Peng *et al.* [55] proposed a sparse-to-dense multi-encoder neural network (SDME-NET) in an end-to-end manner for completion while preserving details of the 3D shape. Notably, the defect point cloud would be completed and refined in

two stages, from sparse to dense. In the first phase, they generated coarse but complete results based on a two-layer PointNet. In the second phase, they leveraged the sparse results of the first stage using PointNet++ to yield a high-density and high-fidelity point cloud. LAKe-Net [9] was proposed as a topology-aware point cloud completion model by localizing aligned keypoints, with a novel Keypoints-Skeleton-Shape prediction manner. Cai *et al.* [4] aim to learn a unified and structured latent space that encodes both partial and complete point clouds to improve partial-complete geometry consistency in an unsupervised manner. Moreover, they applied tailored structured latent supervisions between a series of related partial point clouds to enhance the learning of the structured latent space. In this way, they could reconstruct more accurate complete point clouds equipped with better fine-grained shape details. To reduce the dependency on paired data when training a point cloud completion model, RaPD [5] was proposed as a novel semi-supervised method. A two-stage training scheme is introduced in RaPD. The first stage learns robust semantic prior through reconstruction-aware pre-training, while the second stage learns the final completion deep model by prior distillation and self-supervised completion learning. CS-Net [7] is an end-to-end network to complete the point clouds contaminated by noises or containing outliers. In CS-Net, the completion and segmentation modules work collaboratively to promote each other and benefit from the designed cascaded structure. With the help of the segmentation, the more clean point cloud is fed into the completion module. Wang *et al.* [8] proposed an architecture that relies on three layers used successively within an encoder-decoder structure, attaining significant improvements producing high resolutions reconstruction with fine-grained details. The first one carries out feature extraction by matching the point features to a set of pre-trained local descriptors. Then, to avoid losing individual descriptors as part of standard operations such as max-pooling, an alternative neighbor-pooling operation is proposed that relies on adopting the feature vectors with the highest activations. Finally, upsampling in the decoder modifies our feature extraction in order to increase the output dimension.

Moreover, two feature assemble strategies were proposed to exploit the function of multi-scale features and integrate different information to represent given parts and missing parts, respectively. The global and local feature aggregation (GLFA) and the residual feature aggregation (RFA) were dubbed [56]. These two approaches represent the two types of features and recover coordinates with the help of their combination [56]. Furthermore, a refinement module was also designed to prevent uneven distribution of the generated point cloud.

Given the scenes composed of many objects, Zhao *et al.* [57] devised a partial point cloud completion approach, which mainly emphasizes paired scenarios where two objects are very close and contextually related. And a network was designed to encode the geometry of the individual shapes and the spatial relations between different objects from paired scenes. The two-path scheme is monitored by the consistency loss between different completion sequences by the merits of conditional completion. This approach could

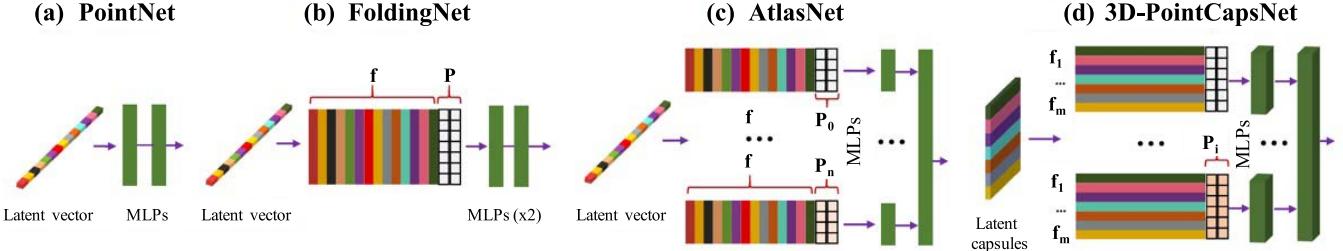


Fig. 7. Comparison of four different SOTA 3D point decoders. (a) PointNet [18] utilizes a single latent vector without surface assumption. (b) FoldingNet [48] learns a 1D latent vector together with a fixed 2D grid. (c) The AtlasNet [20] learns a deformation to transform multiple 2D configurations into local 2-manifolds. (d) The point-capsule network [49] can learn multiple latent representations, each of which can fold a distinct 2D grid onto a specific local patch.

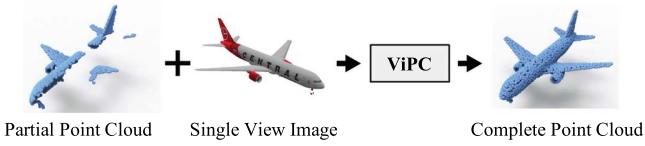


Fig. 8. ViPC is a method to complete a partial point cloud using complementary information from additional single-view images [50].

deal with the complicated case of objects severely occluding each other.

To tackle the challenging dense 3D point cloud completion problem, Li *et al.* [58] proposed a framework to perform end-to-end low-resolution recovery first, followed by a patch-wise noise aware upsampling. This method decodes a complete but sparse shape, followed by iterative refinement. Then the point cloud goes through preserving trustworthy information by symmetrization and patch-wise up-sampling. In this way, a high-fidelity dense point cloud can be achieved. Recently, a Recurrent Forward Network (RFNet) consisting of three modules (Recurrent Feature Extraction (RFE) module, Forward Dense Completion (FDC) module, and Raw Shape Protection (RSP) module) was devised [59]. The RFE extracts multiple global features from the incomplete point clouds for different recurrent levels, while the FDC produces the output in a coarse-to-fine pipeline. Further, the RSP introduces details from the original incomplete shapes to refine the completion results. In addition, a Sampling Chamfer Distance was proposed to capture the shapes of objects, and a Balanced Expansion Constraint was devised to limit the expansion distances from coarse to fine.

**4) Attention-Assisted Methods:** Attention is a flexible mechanism for learning information self-adaptively, and the accumulated important information is weighted highly. By maintaining the spatial arrangements of the partial point clouds, the 3D point-capsule network [49] was devised to handle the sparse 3D point clouds with an auto-encoder. The creation of a 3D capsule network results from the unified, universal 3D auto-encoders. As shown in Fig. 7, the capsule network chooses a promising direction, where a large number of convolution filters realize the learning of the capsule set through dynamic routing. Be integrated with an encoder-decoder architecture, and the PUI-Net [60] has the advantage of extracting features with several cascaded Attention Conv Units and concatenating multi-level features before expansion.

By using the extracted discriminative features, the dense feature map of the fine-grained point cloud is generated by a non-regional feature expansion unit. Li *et al.* [61] presented a dense point cloud completion model (N-DPC), combining self-attention units with the fusion of local features and global features. Sun *et al.* [62] proposed an auto-regressive network PointGrow with self-attention, which operates recurrently. PointGrow sampled each point according to a conditional distribution given its previously-generated points, allowing inter-point correlations to be well-exploited. PointAttN [63] employs the leverage the cross-attention and self-attention mechanisms to tackle the point completion task in a coarse-to-fine manner. It mainly comprises three modules: a feature extractor block for local geometric structure and global shape feature capturing, a seed generator block for the coarse point cloud generation, and a point generator block to produce the fine-grained point cloud.

**5) Folding-Derived Methods:** As a generic architecture firstly proved by Yang *et al.* [48], a Folding-based decoder can reconstruct arbitrary point clouds from 2D grids for objects with detailed structures with low reconstruction errors (Fig. 9, Fig. 10). The FoldingNet is like applying a “virtual force” that deforms/cuts/stretches a 2D grid lattice onto a 3D surface. This deforming force should be affected or modulated by the interconnections induced by the adjacent meshes. Due to the intermediate folding steps and training processes in the decoder could be represented by reconstruction points, the gradual variation of folding force could be seen intuitively.

Folding-based methods (KCNet [64], MSN [31], and PoinTr [33]) usually sample 2D grids from a 2D plane with fixed size and then concatenate them with the global shape representation extracted by the point cloud feature encoder. KCNet [64], AtlasNet [20], MSN [31] and SA-Net [30] reconstruct the complete object by evaluating a set of parametric surface elements and learn projections from 2D to 3D surface elements. SA-Net proposes a structure-preserving decoder with hierarchical folding for complete shape generation.

Moreover, TopNet [24] explores the hierarchical root tree architecture as a decoder to produce a random grouping of points and visually demonstrates the architecture exploited by the decoder through visualizing a node in the tree decoder as a collection of its children. To fully utilize structure details, Wen *et al.* [30] proposed Skip-Attention Network, which contributed two aspects: A skip-attention mechanism was adopted to explore the regional structure details of partial inputs, and

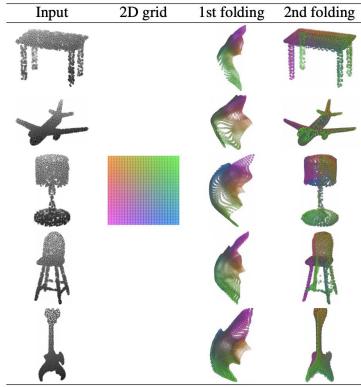


Fig. 9. Description of two-step-folding decoding. The second column shows the 2D grid to be folded when decoding. The third column consists of the results underwent one folding operation. The fourth column composes of the results after the two foldings. The final result is also the recovered point cloud. The color gradient is used to explain the correspondence between the 2D grids in the second column and the recovered point clouds in the last two ones after folding [48].

a structure-preserving decoder using hierarchical folding was proposed to make use of the selected geometric information.

Despite their limited success, the great details of an object are often missed. Existing folding-derived methods, such as PCN [25], FoldingNet [48] and TopNet [24], fail to produce the structure details of an object to certain extent. One of the reasons is that they only rely on a single global shape representation to predict the entire point cloud. In contrast, the rich local region information that helps recover the detailed geometry is not fully utilized. Zong *et al.* [11] proposed an adaptive sampling and hierarchical folding network (ASHF-Net), in which the denoising auto-encoder with an adaptive sampling module to learn local region features, while the hierarchical folding decoder with the gated skip-attention and multi-resolution completion objectives make use of local structural details. Huang *et al.* [65] combined a point-based encoder with an FC-based decoder and a folding-based decoder to produce the complete output, and this model with multistage loss function can be directly applied on the completion of point clouds.

Up-to-now, FoldingNet is the most widely used decoding block in the existing point cloud completion network. There is a drawback in FoldingNet, promoting researchers to build new decoder blocks. The folding manipulation samples the same 2D grids for each parent point, overlooking the local shape characteristics contained in the parent points.

However, there are some limitations of point-based methods.

- The point-based network mainly tackles the permutation issue. Although the point-based methods treat points independently at the local level to maintain permutation invariance, this independence overlooks the geometric relations between the points and their neighbors. It has a fundamental limitation, leading to the loss of local features.
- Most point-based methods work in a coarse-to-fine manner. They are struggling to reconstruct object details, mainly due to two reasons: 1) the coarse outputs created from global embeddings lose the high-frequency

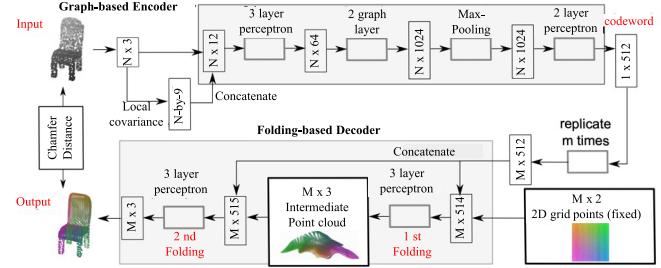


Fig. 10. The architecture of FoldingNet [48].

information for 3D shapes; 2) the second stage acts as a point up-sampling function that fails to synthesize complex topologies.

- The point-based models deal directly with the points and have an extensive computation, which is inferior to the voxel-based method in large scenarios.

### B. View-Based Methods

By the merits of the modality of the images, the key challenge to solve the completion of the point cloud is to effectively integrate the features brought by pose and regional details derived from the incomplete and the global shape information from the single-view images (Fig. 8). As a sensor fusion network, Zhang *et al.* [50] proposed ViPC, which is a view-guided architecture. The ViPC retrieves the missing global structural information from an additional single-view image. The main contribution of ViPC lies in “Dynamic Offset Predictor,” which could refine the coarse output. A multi-view consistent inference was proposed by Hu *et al.* [66] to strengthen geometric consistency in view-based 3D shape completion. And a multi-view consistency loss algorithm for inference optimization was defined, which could be implemented without ground truth supervision. Besides, the depth scan is utilized in ME-PCN [67] to make networks sensitive to shape boundaries, which enables ME-PCN to recover fine-grained surface details while keeping consistent local topology. In order to estimate the 6-DoF pose of 3D canonical shape with the help of several partial observations derived from the same object, Gu *et al.* [43] proposed a weakly-supervised approach to solve this issue. In the training process, the network uses multi-view geometric constraints to jointly optimize the canonical shapes and poses, which can deduce the complete results under the condition of a single partial input. A multi-view completion net (MVCN) [68], combining GAN and multi-view information to enhance the performance of point cloud completion.

The main character and drawback can be listed as follow:

- Different from other methods, the inputs of view-based methods are images, which might be RGB-D images or depth images.
- The performance will largely be depended on the angle and number of views due to the different information that can be acquired from these images.

### C. Convolution-Based Methods

Encouraged by the great success of convolutional neural networks (CNNs) on 2D images, several works try to utilize 3D

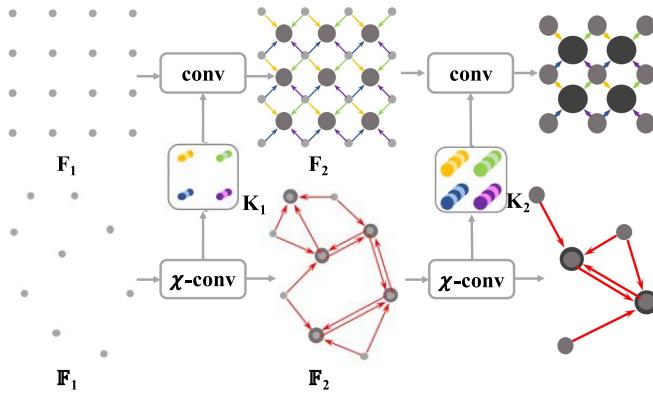


Fig. 11. The illustration of hierarchical convolution on regular grids (top) and point clouds (bottom). In regular grids, convolutions are recursively performed on local grid patches, which typically reduces the grid resolution ( $4 \times 4 \rightarrow 3 \times 3 \rightarrow 2 \times 2$ ), while increasing the number of channels. In point cloud,  $\chi$ -Conv is recursively applied to project or aggregate information from neighborhoods into fewer representative points but with richer information. ( $9 \rightarrow 5 \rightarrow 2$ ) [69].

CNNs to learn the volume representation of three-dimensional point clouds. Nevertheless, transforming a point cloud into 3D volume will bring quantization effects: (1) Loss of details; (2) Insufficient to represent fine-grained information. Therefore, as far as we know, some works directly apply CNN on irregular, partial, and defect point clouds for 3D shape completion.

1) *Preliminary Works*: In terms of the processing of point clouds, several contributions developed CNNs acting on discrete 3D grids of point cloud transformation. Hua *et al.* [70] defined convolution kernels on regular 3D grids, where the points are given the same weights when falling into the same grid. PointCNN [69] implements permutation invariance through a  $\chi$ -conv transformation. In addition to CNN on discrete spaces, several methods define convolution kernels on continuous space (Fig. 11). A rigid and deformable kernel convolution (KPConv) module was devised by Thomas *et al.* [71] to utilize a collection of learnable kernel points for 3D point clouds. The dynamic filter was extended into a convolution operator dubbed PointConv by Wu *et al.* [72]. This operator could be employed to fulfill the deep convolution architecture.

2) *Convolutional Encoder*: In this field, the point cloud will first be voxelized as input of 3D CNNs. Implicit Feature Network (IF-Nets) was devised by Chibane *et al.* [73] to deliver continuous outputs that can handle multiple topologies and complete shapes for incomplete or sparse input data. Still, critically they can also retain details when it is present in the input data and can reconstruct articulated humans. Yi *et al.* [74] devised the Sparse Voxel Completion Network (SVCN), which is composed of two U-Net-like sub-net for structure generation and refinement, respectively. The structure generation sub-net converts the input data into a set of sparse voxels by voxelization and outputting denser voxels representing the 3D surfaces. Then redundant voxels are deleted from the structure refinement network.

However, the voxelization process leads to irreversible loss of geometric information. Xie *et al.* [75] introduced a Gridding Residual Network (GRNet) and took the 3D grids as intermediate representations to process irregular point clouds.

In GRNet, Gridding and Gridding Reverse methods were designed to transform point clouds into 3D grids without any loss of structural information. And the Cubic Feature Sampling layer was presented to extract information of adjacent points and preserve context knowledge. GRNet enables the convolutions on 3D point clouds while preserving their structural and context information. However, the voxel representation of GRNet is only used to reconstruct low-resolution shapes. Therefore, Wang *et al.* [76] develop VE-PCN to transform the unordered point sets into grid representations to support edge generation and point cloud reconstruction. This multi-scale VE-PCN is able to generate fine-grained details for point cloud completion. Liu *et al.* [77] presented MRAC-Net, which includes an anisotropic convolutional encoder for extracting local and global features to enhance the model's extraction ability of latent features.

3) *Deconvolutional Decoder*: Except for feature learning, convolutions can also be utilized in reconstructing the point clouds. Wang *et al.* [78] designed SoftPoolNet, which organizes the extracted features by PointNet called soft pool according to their activation. Regional convolutions were designed to maximize the global activation entropy for the decoding stage. To recover the details of the point clouds and retain the original plane structures, Deng *et al.* [79] proposed 3D Grid Transformation Network, where the weights were calculated for the reconstructed point clouds.

Unlike PointCNN, KPConv and PointConv, in the task of point cloud completion, nearly all convolution-based methods tend to voxelize the point cloud before applying 3D convolution. Hence, we mainly discuss the limitations of these convolution-based methods in volumetric 3D data representation:

- First, not all voxels or grid representations are helpful because they contain occupied and non-occupied parts of the scanning environments. Thus, the high demand for computer storage is unnecessary within this ineffective data representations.
- Second, the voxel or grid size is hard to set, affecting the scale of input data and may disrupt the spatial relationship between points.
- Third, computation and memory requirements grow cubically with the resolution.

#### D. Graph-Based Methods

Since both point clouds and graphs can be regarded as non-Euclidean structured data, exploring the relationship between points or local regions by taking them as the vertices of some graphs is convenient (Fig. 12). Regarding every point in the inputs as the vertices, the edges could be generated by graph-based networks based on the adjacent points. Hence, graph convolutions are naturally suitable for the processing of point clouds. These methods use the advantages of graph convolutions, usually convolving the spatial neighborhoods and generating a new graph by gathering the neighborhood information of each point. Compared to point-based methods, the graph-based approaches consider regional geometric details.

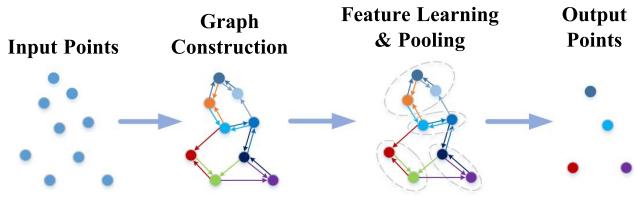


Fig. 12. An illustration of a graph-based network [2].

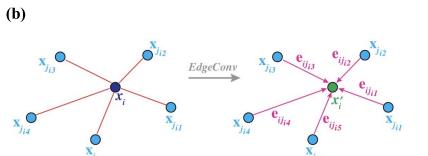
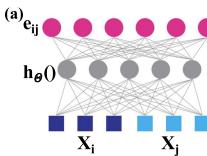


Fig. 13. (a) An edge feature  $e_{ij}$  is calculated from the pair of point  $x_i$  and  $x_j$ . (b) The EdgeConv operator. The output of EdgeConv is computed by converging the edge features of all edges emitted by each connected vertex [80].

As a pioneering work, a dynamic graph convolution was introduced by DGCNN [80]. In the dynamic graph convolution, adjacent matrices could be calculated by the relations of vertexes, which come from latent space. The graph is established in the feature space and can be dynamically updated in the DGCNN. Moreover, EdgeConv was devised to calculate graphs in every network layer dynamically and could be integrated with the existing architectures(Fig. 13). In addition, LDGCNN [81] removes the transformation and connects the multi-level features learned in different layers in DGCNN. Thus, the performance and model size can be optimized. Stimulated by DGCNN, Hassani, and Haley [82] introduced the multi-level network to exploit points and shape features for self-supervised reconstruction. Furthermore, following DGCNN, DCG [83] encodes regional links as feature vectors and refines the point clouds in a coarse-to-fine manner.

Except for the dynamic graph convolution, PointNet++ [19] and FoldingNet [48] could also be regarded as a type of method using graph convolution to exploit information from fixed adjacencies of sampled center points. Combining with graph convolution, Pan [29] designed a hierarchical encoder to refine the local geometric details by propagating multi-scale edge features, which were captured by skeleton generation. Following that, the Edge-aware Feature Expansion (EFE) module was proposed to expand/up-sample information of points by highlighting the regional edge of the point. The ECG can preserve both global structural information and local pattern features. Nodeshuffle and Inception DenseGCN [84] were proposed by Qian *et al.* The former utilizes a Graph Convolutional Network (GCN) to encode regional point features from adjacent points better, while the latter aggregates features at multiple scales. The PU-GCN is a new point up-sampling pipeline when combining the Inception DenseGCN with NodeShuffle [41]. Shi *et al.* [85] proposed a Graph Guided Deformation Network, where the input and intermediate data were considered as controlling and supporting points, respectively, and modeled the optimization guided by the graph convolutional network for the task of point cloud

completion. This network simulates the least square Laplace deformation process via mesh deformation methods, which has the adaptive ability to model the geometric details of the modeling and reduce the gap between the mesh deformation algorithm and the point cloud completion task. Zhu *et al.* [86] designed the PRSCN, which firstly uses Point Rank Sampling approaches to rate and sample feature points more objectively through local outline form. Subsequently, considering the connections among features from different scales, a Cross-Cascade block was designed to integrate features. Besides, Leap-type EdgeConv was integrated to expand the receptive field with kernel size maintained. Moreover, by utilizing global features and local features, LRA-Net [87] was proposed to recover complete point clouds with more details and smoother shapes, which are derived from the structure of PointNet and Graph Convolutional Network (GCN).

**1) Attention-Assisted GCN:** Furthermore, the attentional mechanism is also introduced into GCN. To recover fine-grained shapes, Wu and Miao [88] introduced a learning-based method. They sample local regions of partial inputs, encode their features, and combine them with exploited global features. After the graph is built, all the regional features are gathered, and the graph is convolved with multi-head attention. Graph attention mechanism enables each local feature vector to be searched across the regions and selectively absorb other local features based on relationships in high-dimensional feature space. CRA-Net [89] designed a cross-regional attention unit based on graph attention. This module quantifies underlying connections among regional features under specific contexts and is explained by global features. Given such links, every conditional regional feature vector can be searched as graph attention. In PC-RGNN [90], a graph neural network module was designed, which captures the relations between points comprehensively through the local-global attention mechanism and context aggregation based on a multi-scale graph, greatly enhancing the learning of features.

But there are two challenges for constructing graph-based networks as follows:

- First, defining an operator suitable for dynamically sized neighborhoods and maintaining the weight sharing scheme of CNNs.
- Second, exploiting the spatial and geometric relationships among each node's neighbors.

#### E. GAN-Based Methods

Compared to traditional CNN, the GAN [91] architecture utilizes a discriminator implicitly learning to estimate the point collections provided by the generator (Fig. 14). Due to the characteristics of 3D data, the integration of GAN in point cloud completion possesses several inherent challenges:

- Different from the grid structures of 2D images, where the locations of pixels are clearly defined. In contrast, point clouds with different 3D shapes are highly unstructured. In general, GANs trained on 3D shapes produce point clouds with significant inhomogeneity. That is, points are not evenly distributed on the shape's surface.

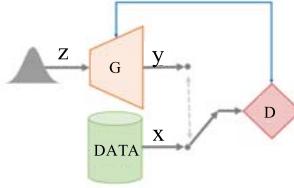


Fig. 14. An description of a generative adversarial networks. The basic architecture is on basis of antagonism between a generator ( $G$ ) and a discriminator ( $D$ ). The  $G$  is designed to produce points that seem different from real data ( $x \sim p_{data}$ ) via a random sample from a simple distribution  $z \sim p_z$  through the generator function. The task of the discriminator is to distinguish synthetic samples from real ones.

This inhomogeneity can lead to shapes with unwanted holes, undermining the integrity of predictions.

- The disorder of point clouds determines the completion task significantly different from two-dimensional image completion. In 2D image rendering, one can easily measure the reconstructed consistency between partial input visible regions and predicted outputs given mesh aligned pixel corresponding. This comparison is challenging in 3D shape completion because the corresponding regions of two 3D shapes may be located at different positions in 3D space. GAN inversion results in poor reconstruction, which jeopardizes the shape-completion mission.
- Whereas simple GANs can only yield a small size of (1024 or 2048) point collections because of the complicated point distribution and the notoriously hard training of GANs.

Therefore, the researchers greatly improved the point cloud completion based on the traditional GAN.

1) *End-to-End Mechanism*: End-to-end learning is commonly used in 3D point cloud completion. By the integration of a 3D Encoder-Decoder Generative Adversarial Network (3D-ED-GAN) and a Long-term Recurrent Convolutional Network (LRCN), Wang *et al.* [92] introduced a new structure. 3D-ED-GAN leverages an encoder to map voxelized 3D shapes into a probabilistic latent space and uses a GAN to facilitate the decoder generating the complete volumetric shapes with the help of the latent feature representations. Nevertheless, these approaches can only use 3D volumes as inputs or get a voxel representation for results. Achlioptas *et al.* [93] devised r-GAN with both generator and discriminator using fully connected layers. The AE was trained to learn the latent space. The l-GANs were trained in the latent space, which was more easily trained than simple GANs with coverage of data distributions. In the training of latent representation, the capability of multi-class GAN is almost the same as that of class-specific GAN. Gurumurthy and Agrawal [94] have devised a scheme utilizing latent GAN and AE. Nevertheless, they used a time-consuming optimization procedure for every batch of inputs to choose the best seed for GAN. Yu *et al.* [95] devised a point encoder GAN, where the max-pooling layer was utilized to tackle the irregular problem in the learning process, and two T-Nets (derived from PointNet) were added in the encoder-decoder architecture to represent the characteristics of the inputs better. And a hybrid recovery loss function was proposed to compute the diversity between two

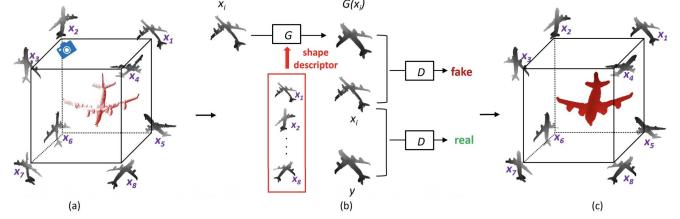


Fig. 15. An illustration of a generative multi-view representation. For example, (a) 8 depth maps of defect inputs were rendered from 8 viewpoints; (b) these introduced 8 depth maps are completed through a multi-view completion network, including adversarial loss, producing 8 completed depth maps; (c) the 8 depth maps are back-projected into a completed shape [68].

groups of disordered data. Chen *et al.* [96] proposed an end-to-end conditional GAN called GeneCGAN. From the heredity perspective, a simulated genetic (SG) layer was devised. It is executed in a hierarchical root tree using ancestor information and the connection of neighborhoods. Through a prior fusion strategy, the global feature is appended to the tree's root node as conditional information, and the conditional probability distribution of the inputs is learned.

The effective latent space representations of the point cloud provide important and fundamental information that can be utilized for 3D shape reconstruction. Wen *et al.* [97] presented Cycle4Completion with two synchronous cycle transformations between the latent spaces of complete 3D shapes and incomplete 3D shapes. The cycle transformation could facilitate the model to learn about 3D shapes through studying to produce complete or incomplete shapes with the help of complementary shapes. Chen *et al.* [98] proposed a framework for unpaired shape completion, the core of which is an adaptation network that acts as a generator to transform latent codes of the original point scans and maps them to underlying latent spaces of clean and complete object scans. These two latent spaces regularize the issue by limiting the transfer issue to respective data manifolds. Zhang *et al.* [99] presented ShapeInversion to introduce GAN inversion into shape completion for the first time. ShapeInversion leverages a GAN pre-trained on complete shapes by searching for a latent code to obtain the complete shape that best reconstructs the input of a given part. In this way, ShapeInversion eliminates the need for paired training data and can combine rich prior information in a well-trained generation model. Combining the latent-space GAN and Laplacian GAN, Egiazarian *et al.* [100] devised a multi-level network that can produce 3D objects with increasing levels of details. Based on GAN, a PU-GAN was presented by Li *et al.* [41], which is an up-sampling network that aims to learn the point distributions from the latent space. The PU-GAN can also up-sample points over patches on the surfaces of the object. An up-down-up expansion module was constructed in the generator to up-sample point features, which has an error feedback unit and self-correction functions. Moreover, a self-attention unit was also developed to increase the integration of features.

Furthermore, Li *et al.* [101] proposed a two-fold modification of GAN (PC-GAN). In PC-GAN, hierarchical Bayesian networks and implicit generative architectures are combined through hierarchical and interpretable sampling. The key of

this approach lies in the posterior inference model, which is trained for hidden variables. Besides, rather than using the SOTA Wasserstein GAN objective, a sandwiching objective was devised to result in a compacter Wasserstein distance estimate than the typically utilized dual form. Hence, PC-GAN provided a generic architecture to comprise the existing GAN easily. Wen *et al.* [102] devised a dual-generators network, in which the first generator was designed to learn point embeddings, while the second one was utilized to refine the generated point cloud based on a depth-first point embedding to generate a uniform output. To minimize the influence of noises and geometrical loss of incomplete point cloud, PF-Net [103] retains the spatial arrangement of incomplete inputs and can calculate the complicated geometry of the missing area. To this end, PF-Net uses a multi-level generation based on feature points to predict the missing parts in a hierarchical network. PF-Net also utilizes multi-stage completion and adversarial loss to produce more realistic missing regions. Among them, the adversarial loss can better solve the problem of multiple modes in the prediction. Cheng *et al.* [104] proposed an end-to-end generative adversarial network-based dense point cloud completion architecture (DPCG-Net). The DPCG-Net designed two GAN-based modules that translate point cloud completion into mapping between global feature distributions obtained by encoding partial point clouds and ground truth, respectively.

2) *Refinement*: Besides, the refinement strategy is also commonly integrated with GAN. Wang *et al.* [105] developed a feature alignment approach for learning the shape prior. Moreover, a coarse-to-fine method was devised to combine the shape prior with the fine phase. The loss of feature alignment comprises an L2 distance and an adversarial loss derived from Maximum Mean Discrepancy Generative Adversarial Network (MMD-GAN). Wang *et al.* [42] devised a point completion network with a cascaded refinement network (CRN) as a generator to synthesize these missing parts with high quality by using the details of inputs. Moreover, they designed a patch discriminator, which uses adversarial training to know the precise point distribution and punishes the generated shapes differently from the ground truth. Furthermore, to generate high-quality targets with detailed geometry, Wang *et al.* [28] expanded this strategy to synthesize fine-grained targets. Two self-training strategies were proposed to improve the reconstruction performances in both supervised and self-supervised settings.

3) *Multi-View GAN*: The views of the same 3D model share some common information that can be explored, including global and regional information as seen from various angles. Hu *et al.* [68] proposed a multi-view completion net (MVCN) (Fig. 15), which leverages information from all views of a 3D shape to assist the completion of every single view. Benefiting from the multi-view presentations and network architecture with conditional GAN, MVCN enhances the performance of 3D completion. Wang *et al.* [106] tried to convert the three-dimensional point cloud generation issue to a two-dimensional geometry-image generation problem and introduced an adversarial VAE to optimize the GIG proposed by combining adversarial learning with VAE. While it is easy

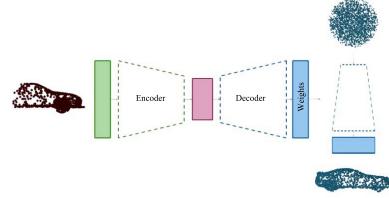


Fig. 16. HyperPocket architecture with single VAE encoder [36].

to create depth maps of 3D shapes independently, there are two drawbacks. First, they do not encourage consistency between depth maps from the same 3D object, influencing the accuracy of 3D objects obtained through back-projecting completed depth maps. Second, they could not complete a depth map and use information from other depth maps of the same 3D object. The accuracy of completing a single depth map is limited.

4) *Integrated With RL*: More recently, reinforcement learning (RL) has been integrated into GAN. Sarmad *et al.* [23] presented RL-GAN-Net, in which a reinforcement learning mechanism can control the GAN. The architecture can transform noisy defect input data into full shape with high fidelity through the control of GAN. Vaccine-Style-Net [107] was carried out in the function space of 3D surfaces, and the 3D surface is represented as the continuous decision boundary function. At the same time, an RL unit is embedded to derive the complete 3D geometry from the partial inputs.

5) *Integrated With GCN*: Except for RL, GCN is also commonly utilized with GAN. Valsesia *et al.* [108], [109] studied the unsupervised problem of generative models using graph convolution. They emphasize the generator of GAN and define graph convolution methods. They devised a network that learns to produce local features to approximate the embedding of output geometry and defined an up-sampling layer of a graph convolution to sample more efficiently using self-similarity prior. Xie *et al.* [110] presented a SpareNet by utilizing channel-attentive EdgeConv to learn the regional features and global shape. SpareNet utilizes shape features as style code to adjust the normalization layer during folding to enhance its capabilities. Moreover, a differentiable renderer is used to project the complete point cloud onto the depth map, and adversarial training is applied to promote the perception of reality from different viewpoints. Li and Baciu [111] devised a Hierarchical Self-Attention GAN (HSGAN) to use a random code and transforms it hierarchically into a representation graph by combining GCN and self-attention. In this model, the topology of the global graph is embedded into shape generation, and latent topology information is utilized to recover the geometry structure of the 3D shape.

#### F. Variational Autoencoders (VAEs)-Based Methods

Classic AEs and VAEs are trained on a complete 3D object. The model's weight is then determined to generate a latent representation of incomplete data. And the generative model completes partial inputs in conditional generative network settings. The completion production is based on the learning mode distribution extracted from the complete shape.

Spurek *et al.* [114] introduced a Variational Autoencoder architecture named HyperPocket, which is capable of disentangling latent representations and thus generating multiple variants of completed 3D point clouds (Fig. 16). The point cloud processing was split into two unconnected data streams and utilized a hyper network paradigm to fill what are known as pockets of space left by missing object parts. Pan *et al.* [36] devised a Variational Relational Point Completion Network (VRCNet) to leverage a dual-path unit and a VAE-based relational enhancement module for probabilistic modeling. And they also designed multiple relational modules that could efficiently utilize and integrate multi-scale point information, including the Point Self-Attention Kernel and the Point Selective Kernel Unit. Zamorski *et al.* [115] presented an application of three generative modeling approaches and tested the architectures of AE, VAE, and Adversarial Autoencoder both quantitatively and qualitatively. Besides, they introduced a method that uses the extended PointNet model (Double PointNet) to manipulate points based on both local features and the global shape. AutoSDF [116] proposed an autoregressive prior for 3D shapes to solve multi-modal 3D tasks such as shape completion, reconstruction, and generation. The distribution over 3D shapes is modeled as a nonsequential autoregressive distribution over a discretized, low-dimensional, symbolic grid-like latent representation of 3D shapes. This enables the network to represent distributions over 3D shapes conditioned on information from an arbitrary set of spatially anchored query locations and thus perform shape completion in such arbitrary settings.

However, the advantages and disadvantages of VAE-based methods can be concluded as follow:

- The training of VAE-based methods is more stable, compared with other (3D) generative models.
- The latent representations of VAE can be manipulated to control the shapes of the generated point cloud [114].
- The quality of VAE-based methods cannot be comparable to the GAN-based generative models, while the diversity of VAE-based methods is superior than GAN-based methods. Because of the injected noise and imperfect element-wise measures such as the squared error, the generated 3D shapes might not be smooth enough.
- The VAE-based methods are enhanced by AutoSDF [116], which is a VQ-VAE-like model. The great success of VQ-VAE [117] in image generation will also boost the VAE-based methods on point cloud completion.

#### G. Transformer-Based Methods

Transformer [118] was firstly proposed for encoding sentences in natural language processing, and after that became popular in the areas of 2D computer vision (CV) [119], [120]. Pioneered by PCT [121], Pointformer [122], and PointTransformer [123], the transformer has begun its journey in point cloud process.

By the merits of the representation learning ability of transformer, Yu *et al.* [33] regarded point cloud completion as a set-to-set translation issue and proposed a transformer

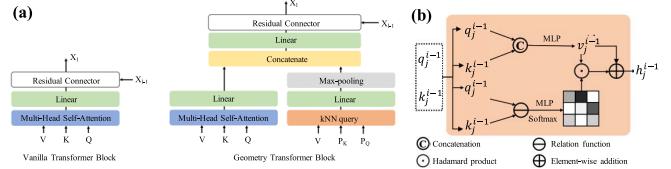


Fig. 17. (a) Compares of the transformer block and the geometry-aware transformer block [33] and (b) the detailed structure of skip-transformer [112].

encoder-decoder structure for point cloud completion. By representing the point cloud as a set of disordered points with position embeddings, The point cloud can be converted to a series of point proxies. The transformer was employed for the point cloud generation. To facilitate transformers to use better the inductive bias of 3D geometric structures of the point cloud, they further design a geometry-aware block that explicitly simulates local geometric relations (Fig. 17a).

Rather than utilizing the representation learning ability of transformer, Xiang *et al.* [112] devised SnowflakeNet with Snowflake Point Deconvolution (SPD), applying transformer-based structure to the decoding procedure. The SnowflakeNet models the generation of complete point clouds as the snowflake-like growth of points in 3D space. After each SPD, child points are gradually generated by splitting their parent points. The idea of revealing geometry details is to introduce a skip-transformer in SPD to learn point splitting modes that can best suit local regions. Skip-transformer utilizes an attention mechanism to summarize the splitting patterns used in the previous SPD layer, resulting in the splitting of the current SPD layer. The locally compact, structured point cloud produced by SPD can accurately capture the structure features of 3D shape in local patches, enabling the network to predict highly detailed geometries (Fig. 17b).

Moreover, Lin *et al.* [124] presented PCTMA-Net, where the Transformer's attention mechanism can extract the local context within a point cloud and exploit its incomplete local structure details. A morphing-atlas-based point generation network fully utilizes the extracted point Transformer feature to predict the missing region using charts defined on the shape. Improved from PMP-Net [113], PMP-Net++ [21] introduced a transformer-enhanced representation learning network, significantly improving completion performance.

However, there are some limitations of transformer-based models.

- Due to the amount of the transformer parameter, the model is too large to deploy on devices compared with other methods.
- Except for the visual interpretation of attention in SA-Net [30], the mechanism of transformer for enhanced performance is hard to interpret.

#### H. Other Methods

In addition to the methods mentioned above, researchers have also carried out studies on up-sampling and pre-train methods.

Wen *et al.* [113] designed PMP-Net to complete the point cloud by moving every point in the incomplete input to

ensure the shortest total distance of the point moving path (PMP). Therefore, PMP-Net predicts the unique PMP of each point based on the constraint of the total point movement distance. Son and Kim [125] introduced a shape completion framework to preserve both the global contexts and the local features, in which a Symmetry-Aware Up-sampling Module (SAUM) was devised to preserve the geometric details and take advantage of the symmetry of shape completion. Wang *et al.* [126] developed a pre-training mechanism named Occlusion Completion (OcCo), which works by shielding the occluded points from observations from different camera views and then optimizing the completion model. In this way, This method learns a pre-trained representation that could recognize the inherent visual constraints embedded in a real point cloud.

## VII. COMPARISON

This section summarizes the results of the latest methods on several datasets. We will compare the performance of these approaches and provide some advice for future works. The results come from the original papers. Thus the resolutions and datasets setting are varied. If the type of CD or datasets settings are not specified, the performance can be compared under the same resolution.

### A. Summary of the Performance on PCN, ModelNet, and Completion3D With Ground Truth Provided

The PCN is the most commonly utilized dataset for 3D shape completion. These three datasets all belong to synthetic benchmarks. As shown in Table II, III, V, and Fig. s2 (see the Supplementary Material), there are the results performed by various methods, and some inferences could be drawn as follow:

- The point-based models with MLP, integrated as the basic unit, are widely utilized to learn the point-wise information.
- The graph-based and GAN-based networks can fulfill excellent results on completing 3D point clouds. More attention needs to be paid to the combination of these two methods.
- Transformer-based models have recently attracted more attention because of their powerful ability to process irregular data. The SOTA methods could be credited to the latest SnowflakeNet. Nevertheless, extending transformer-based models into the spectral domain remains a challenge.

### B. Summary of the Performance on KITTI Without Ground Truth Provided

Due to the missing ground truth, the performance in the KITTI dataset is generally obtained by testing the model that trained directly on the training set of “cars” in ShapeNet. And several methods fine-tune their model on ShapeNetCars as [75] while testing the model on KITTI. Tables IV, V, and Fig. s2 (see the Supplementary Material) give the performance achieved by numerous methods on the KITTI, and from which some observations could be listed as follow:

- The KITTI dataset is derived from the real-world scans. The intrinsics challenges such as no ground truth provided and ultimately sparse in some instances bring difficulties to point cloud completion.
- As shown in Table III, the Point-based, GAN-based, and Transformer-based methods all have achieved completion effects facing such challenges.
- Besides, some works [43], [99] are devised for point cloud in the real world, and more effective should be paid on these directions.

## C. Complexity Analysis and Generalization Performance

To further gain insight into model performances, further analysis of the parameter (Params) and theoretical computation cost (FLOPs) are conducted to compare the complexity of the model and the time-consuming situation. As shown in the second and third columns of Table VI, it can be found that FoldingNet [48] owns the smallest number of parameters, while the GRNet [75] and PF-Net [103] possess a relatively great number of parameters due to the complex architecture. On the other hand, PMPNet [113] has the lowest computational cost while GRNet still has the highest one since the operator on their grids needs more parameters and computation. It is noticed that the number of parameters in SnowflakeNet [112] and PoinTr [33] is also relatively high because of the attention mechanism.

Furthermore, generalization results of models are also compared in Table VI since the performance in unknown categories is also another critical metric. Because ShapeNet55 and ShapeNet34 are proposed to measure the generalization performance, the models are trained in the seen 34 categories and evaluated on the unseen 21 categories. As we can see in the last two columns of Table VI, PoinTr [33] performs well in ShapeNet55 and ShapeNet34, proving the most remarkable generalization of PoinTr.

## VIII. APPLICATIONS

The point cloud completion is a vital technology in many applications and has accumulated several achievements. Therefore, in this section, the applications of point cloud completion in numerous fields will be introduced. We will discuss the applications where point cloud completion can be utilized at current firstly. Then, four major applications where point cloud completion will be applied are also reviewed.

### A. Current Applications

This paper discusses the techniques of point cloud completion are object-level completion. Therefore, these methods are excellent in single object completion in real-world scenes, for example, reconstructions of a chair in real-world scenes produced by the HyperPocket [36] model. Relationship-based methods [57] complete partial scans of pairwise scenes by conditioning the completion of each object with the partial scan of the other object. Both of them can complete the real-world scenes for the convenience of indoor robots. Further, Gu *et al.* [43] proposed a weakly supervised method

TABLE II

COMPARISON OF 3D POINT CLOUDS COMPLETION PERFORMANCES ON THE PCN. HERE, “CD” REPRESENTS THE MEAN CHAMFER DISTANCE, AND ‘EMD’ REPRESENTS THE MEAN EARTH MOVER’S DISTANCE. THE “-” REPRESENTS THE PERFORMANCES THAT ARE UNREACHABLE. (THE CD LOSS IS SCALED BY 1000, AND EMD LOSS IS SCALED BY 100. IF NOT SPECIFIED, THE CD MEANS CD- $\ell_1$ )

Methods		PCN		
		CD (Value/Resolution)	EMD (Value/Resolution)	F-Score@1% (Value/Resolution)
Point-based Methods	N-DPC [61]	10.25/16384	6.05/16384	-
	MSN [31]	10.00/8192	3.78/8192	-
	MSPCN w/ MVCS [47]	0.94/16384 (CD- $\ell_2$ )	-	-
	PCN [25]	10.02/16384	6.40/16384	-
	SK-PCN [46]	0.18/2048 (CD- $\ell_2$ )	0.50/2048	-
	ASFm-Net [52]	12.09/4096	-	-
	FinerPCN [45]	16.65/2048	6.14/2048	-
	Refinement of Predicted Missing Parts [45]	1.28/2048	-	-
	SDME-Net [55]	17.00/4096	4.33/4096	-
	ASHF-Net [11]	0.26/16384	-	-
	TopNet [24]	0.97/2048	-	-
	SA-Net [30]	0.77/2048	-	-
	Multistage Loss Function [65]	8.75/1024	5.02/1024	-
	Multi-View Consistent Inference [66]	8.05/2048	-	-
View-based Methods	Weakly-Supervised 3D Shape Completion [43]	26.40/8196 (3 categories)	-	-
	MVCN (Render4Completion) [68]	8.30/16384	-	-
	ViPC [50]	3.31/2048 (ShapeNet-ViPC)	-	0.59/2048 (ShapeNet-ViPC)
	GRNet [75]	0.27/16384 (CD- $\ell_2$ )	-	0.71/16384
Convolution-based Methods	SoftPoolNet [78]	5.94/16384	4.80/1024	-
	ECG [29]	1.02/2048	-	-
Graph-based Methods	Graph-Guided Deformation Network [85]	0.60/2048	-	-
	PRSCN [86]	4.48/2048 (ShapeNet-part(14)) 4.78/2048 (ShapeNet-part(16))	-	-
	DCG [83]	3.28/2048	-	-
	LRA-Net [87]	26.04/2048	11.21/2048	-
	Graph Guided Deformation [85]	0.60/2048	-	-
GAN-based Methods	CRN [42]	8.51/16384	-	-
	Latent-Space Laplacian Pyramids [100]	0.34/2048 (3 categories)	4.16/2048 (3 categories)	-
	MMD-GAN [105] (Learning Shape Priors)	8.57/2048 (CD- $\ell_2$ )	-	-
	NSFA [56]	0.81/16384	-	-
	PF-Net [103]	1.01/2048	-	-
	SCRN [28]	8.51/16384	-	-
	Cycle4Completion [97]	1.41/2048	-	-
	ShapeInversion [99]	1.49/2048	-	0.84/2048
	GencCGAN [96]	2.63/2048	2.70/2048	-
	SpareNet [110]	0.52/2048	0.19/2048	-
Transformer-based Methods	SnowflakeNet [112]	1.86/2048	-	-
	PoinTr [33]	1.09/8192 (CD- $\ell_2$ ) (ShapeNet-55)	-	0.46/8192 (ShapeNet-55)
Other Methods	PMP-Net [113]	8.66/2048	-	-

to estimate both 3D canonical shape and 6-DoF pose for alignment, given multiple partial observations associated with the same instance. This method enables the completion toward objects in the wild, which might benefit the high-precision of 3D object detection or tracking in outdoor scenes. Considering many works being powerful for synthesis datasets, significant efforts should be made to enable object-level point cloud completion in more practical scenarios.

### B. Construction

Because of the enormous benefits, the completed point cloud is urgently needed by industries to enhance productivity, such

as the manufacturing [127], [128] and construction industry [129], [130]. For instance, as shown in Fig. 18a [131], the point cloud was acquired at a precast concrete manufacturing plant. Compared with traditional measurement approaches, such as manual or other measurements based on equipment, the point cloud data captured by sensors has the merits of a higher measurement rate and higher measurement accuracy.

### C. Mining Space

Nowadays, 3D point cloud processing technology has been applied frequently in mining. For example, the national robotics engineering center of the United States can successfully draw a high-precision, 3D map of underground roads using

TABLE III

COMPARISON OF 3D POINT CLOUDS COMPLETION PERFORMANCES ON THE MODELNET AND COMPLETION3D. “CD” REPRESENTS THE MEAN CHAMFER DISTANCE AND ‘EMD’ REPRESENTS THE MEAN EARTH MOVER’S DISTANCE. THE “—” STANDS FOR THE PERFORMANCES ARE UNREACHABLE. (THE CD LOSS IS SCALED BY 1000 AND EMD LOSS IS SCALED BY 100)

Methods		ModelNet		Completion3D	
		CD (Value/Resolution)	EMD (Value/Resolution)	CD (Value/Resolution)	EMD (Value/Resolution)
Point-based Methods	ASFm-Net [52]	-	-	0.668/2048	-
	An End-to-End Shape-Preserving [54]	-	-	26.900/1024	3.370/1024
	N-DPC [61]	-	-	1.695/2048	-
Convolution-based Methods	GRNet [75]	1.064/2048	-	1.046/2048	-
	SoftPoolNet [78]	-	-	1.19/2048	-
Graph-based Methods	PRSCN [86]	4.649/2048	-	-	-
	Cross-Regional Attention Network [89]	27.00/2048 (6 categories without normalization)	11.700/16384	-	-
GAN-based Methods	CRN [42]	2.293/2048	-	0.921/2048	-
	Cascaded Refinement Network [28]	2.472/16384	-	0.914/16384	-
VAE-based Methods	HyperPocket [36]	-	-	1.791/2048	-
Transformer-based Methods	SnowflakeNet [112]	-	-	0.760/2048	-
	PointAttN [63]	-	-	0.663/2048	-
Other Methods	PMP-Net [113]	-	-	0.923/2048	-

TABLE IV

COMPARISON OF 3D POINT CLOUDS COMPLETION PERFORMANCES ON THE KITTI. “MMD” REPRESENTS THE MINIMAL MATCHING DISTANCE AND “FD” REPRESENTS THE FIDELITY DISTANCE. THE “—” STANDS FOR THE PERFORMANCES ARE UNREACHABLE

Methods		KITTI		
		MMD (Value/Resolution)	FD (Value/Resolution)	Consistency(Value/Resolution)
Point-based Methods	PCN [25]	0.0185/16384	0.028/16384	0.01163/16384
	ASFm-Net [52]	-	0.014/4096	0.020/4096
	ASHF-Net [11]	0.541/16384	0.773/16384	0.298/16384
Convolution-based Methods	GRNet [75]	-	-	0.313/16384
	SoftPoolNet [78]	0.01465/16384	0.02171/16384	0.00992/16384
GAN-based Methods	MMD-GAN [105]	-	0.034/2048	-
	NSFA [56]	-	0.0261/16384	-
	SpareNet	0.368/2048	1.461/2048	0.249/2048
Transformer-based Methods	PoinTr [33]	-	0.526/8192	-

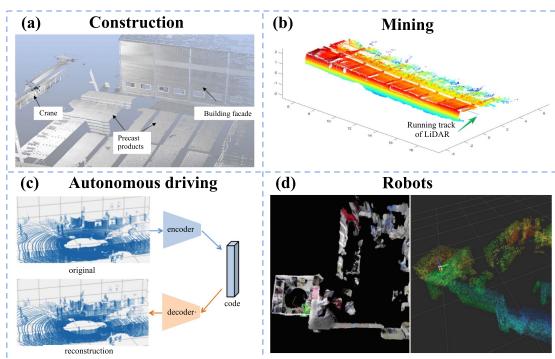


Fig. 18. Point cloud completion in (a) construction [131], (b) fully mechanized mining equipment [132], (c) 3D reconstruction [133], and (d) localization of autonomous indoor robots [134].

the point cloud data obtained by a 3D laser scanner and then propose an intelligent mining model based on the 3D

map [135]. The 3D point cloud data is used to describe and draw the whole fully-mechanized mining face, accurately and intuitively reflecting the spatial position relationship between the coal wall and fully-mechanized mining equipment. This method provides directional information for the scraper conveyor to adjust the displacement of the hydraulic support in time (Fig. 18b) [132]. Significantly, the complete point cloud will provide more accurate information for mining space.

#### D. Autonomous Driving

On the one hand, the main task of autopilot is to find a compact 3D point cloud representation and maintain the capacity for reconstruction; As shown in Fig. 18c, reconstruction helps to store data in autonomous driving. Because every autonomous vehicle (AV) has to store high-definition maps and collect real-time LiDAR sweeps, data storage would

TABLE V  
SUMMARIZATION OF MILESTONE DL NETWORKS BASED ON THE POINT CLOUD PROCESSING METHODS

Methods		Highlights	limitation
Point-based methods	AltasNet [20]	AtlasNet regards a 3D shape as a collection of parametric surface elements and infers a surface representation.	The reconstruction largely depends on the reiterated many times.
	MSN [31]	MSN predicts a set of parametric surface elements and undergoes a combination with the partial input by a sampling algorithm.	Fail to generate fine-grained details of object shape.
	PCN [25]	Combining the fully connected network and FoldingNet, PCN performs the coarse-to-fine completion.	Incapable of synthesizing shape details.
	ASFM-Net [52]	ASFM-Net is an asymmetrical Siamese auto-encoder model to learn a shape prior information.	The visualization performance of completion can be further improved.
	SK-PCN [46]	SK-PCN predicts the 3D skeleton to acquire the global structure and completes the surface by learning skeletal points' displacements.	The meso-skeleton only focus on the overall shapes.
	FoldingNet [48]	A two-stage generation process is commonly used to assume that 3D objects can be recovered from a 2D-manifold.	The implicit intermediate is hard to be constrained explicitly.
	SA-Net [30]	SA-Net proposes hierarchical folding in the multi-stage points generation decoder.	It is difficult to interpret and constrain the implicit representation of the target shape from the intermediate layer to help refine the shape in the local region.
Convolution-based methods	ASHF-Net [11]	ASHF-Net proposes a hierarchical folding decoder with the gated skip-attention and multi-resolution completion target to exploit the local structure details of the incomplete inputs.	The surface of the results is not smooth. Unstructured predictions are obtained by the decoder [112].
	GRNet [75]	GRNet introduces 3D grids as intermediate representations to regularize unordered point clouds.	It is still subject to the resolution. Higher resolution will bring about great computational cost
Graph-based methods	VE-PCN [76]	VE-PCN incorporates the structure information into the shape completion by leveraging edge generation.	The edges only focus on high frequency components.
	PRSCN [86]	PRSCN proposes a Point Rank Sampling to select feature points and eliminate the influence of outlier points. Point Rank Sampling pays more attention to the local relative importance and generates high fidelity geometrical shapes.	It can only predict the missing parts. The Point Rank Sampling (PRS) module is susceptible to the density distribution. The generalization ability needs to be improved, facing data from other domains with different data distribution and object patterns
	ECG [29]	ECG applies several modules with graph convolutions for edge-aware feature learning and/or preserving.	There are noises around the completion result.
	DCG [83]	DCG employs the point set auto-encoder first to produce a sparsely coarse shape and then refines it by encoding neighborhood connectivity on a graph representation.	The performance can be further improved, due to the noises around the final output. And the unevenly distributed points might lead to high EMD.
View-based methods	CRA-Net [88]	The CRA module facilitates each local feature vector to search the regions within a fully-connected graph and selectively absorbs other local features based on their relationships with graph convolution.	Inensitive to edge information. Failed to fully exploit the local features, which is soloved by [89]
	ViPC [50]	An extra single-view image explicitly provides the global structural prior information for completion.	The surface of the results is not that smooth. And the performance is largely depended on the angles of images.
GAN-based methods	PF-Net [103]	PF-Net uses a feature-points-based multi-scale generating network and combines multi-stage completion loss and adversarial loss to generate realistic missing regions.	PF-Net neglects effective feature fusion, resulting the shape lacks local geometric details.
	ShapeInversion [99]	ShapeInversion addresses the domain gaps between virtual and real-world partial scans and various simulated partial shapes through GAN inversion.	Both shape completion and manipulation are conducted on a model pre-trained with a single category
	MMD-GAN (Learning Shape Priors)	MMD-GAN proposes 3D feature alignment methods to learn the shape priors from complete and partial point clouds.	Inensitive to edge information. The performance might be further enhanced by training a model in one class as [99].
	NSFA [56]	NSFA proposes two separated feature aggregation namely GLFA and RFA, considers the existing known part and the missing part separately.	Suffer from the information loss of structure details.
	Cascaded Refinement Network [42]	The generator of CRN is a cascaded refinement network, exploiting the details of the partial inputs and synthesizing the missing parts with high quality.	The performance can be further enhanced. There are a few pairs of partial and complete point cloud with unmatched scales in CRN dataset.
Transformer-based methods	PoinTr [33]	PoinTr regards point cloud completion as a set-to-set translation issue and employs a transformer encoder-decoder architecture.	The model is relatively large due to transformer's large number of parameters.
	SnowflakeNet [112]	SnowflakeNet introduces a skip-transformer to learn splitting patterns in Snowflake Point Deconvolution for progressively increasing the number of points.	The resulting point cloud is not evenly distributed. It focuses more on structured output rather than topological information [9].

be expensive for a large fleet of AVs. While no mature compression standard is available to deal with large-scale open scene 3D point clouds [133], reconstruction technology can provide 3D point cloud compression and reduce the cost of data storage. Point cloud completion can be used for reconstruction to obtain a higher quality point cloud.

On the other hand, the production of high-definition maps is relatively expensive and impractical for every scene. Hence, semantic scene completion is proposed to complete the sparse LiDAR sweeps. In this area, convolution-based methods have been widely applied. However, the details of scenes are missing due to the voxelization of the point

cloud [136]–[138]. To tackle this issue, the completion of sparse LiDAR sweeps through point-based methods might be a solution.

Images, point clouds, and radar data could be combined to produce precise, geo-referenced, and information-rich cues for AVs' navigation and decision-making [139]. Data from low-end LiDAR and high-end LiDAR are also fused. At the same time, there are some difficulties in merging these data. Most importantly, in the fusion of cross-source data, the sparsity of point cloud leads to inconsistent and missing data. Therefore, the point cloud completion can be utilized to tackle the sparsity of real-time LiDAR sweeps.

TABLE VI

**COMPLEXITY ANALYSIS.** THE NUMBER OF THE PARAMETER (PARAMS) AND THEORETICAL COMPUTATION COST (FLOPS) OF EXISTING METHODS ARE REPORTED. THE AVERAGE L2 CHAMFER DISTANCES OF ALL CATEGORIES IN SHAPENET-55 AND UNSEEN CATEGORIES IN SHAPENET34 ARE ALSO PROVIDED

Methods	Param.	FLOPS	ShapeNet55	ShapeNet34
			-CD- $\ell_2$	-CD- $\ell_2$
FoldingNet [48]	2.30 M	27.58 G	3.12	3.62
PCN [32]	5.04 M	15.25 G	2.66	3.85
TopNet [24]	5.76 M	6.72 G	2.91	3.50
PF-Net [103]	73.05 M	4.96 G	5.22	8.16
GRNet [75]	73.15 M	40.44 G	1.97	2.99
ECG [29]	13.77 M	9.32 G	1.76	2.09
CRN [42]	5.00 M	8.41 G	2.01	3.30
ASFM-Net [52]	10.75 M	17.41 G	1.70	2.67
PMPNet [113]	5.18 M	3.05 G	1.47	2.15
SnowflakeNet [112]	18.43 M	5.20 G	1.21	2.20
PoinTr [33]	30.9 M	10.41 G	1.07	2.05

### E. Robotics

Point cloud technology has been widely used in robotics in recent years. Localization and mapping are crucial for autonomous mobile robot navigation in unknown environments. **Localization** An accurate 6-degree of freedom (6-DoF) pose is ideal for unmanned aerial vehicles (UAVs) or humanoid robots and robots performing missions. However, localization with RGB-D cameras in a 3D environment still presents some challenges: (1) Robots often take a long time to orient themselves in a 3D environment; (2) There are changes in the 3D environment. For instance, as shown in Fig. 18d, Luo *et al.* presented a method to determine the 6-DoF global positioning of the robot without a given initial pose, which is dubbed the Fast Scene Recognition and Alignment (FSRA) system [134]. **Mapping** More attention has been paid to the three-dimensional environment point cloud maps in public and engineering space. 3D environment maps are helpful for autonomous mobile robots in indoor environments without GPS. However, these precise locations and mappings still require a complete point cloud as a prerequisite.

## IX. FUTURE DIRECTION AND OPEN QUESTIONS

Based on the above discussions, there are two problems to be solved: (1) To achieve high precision and robust completion by overcoming the above challenges. (2) Fast operation speed and high accuracy guarantee. In this part, we propose several future research directions to enhance the performance of DL-based point cloud network as follows:

- Unlike images, point cloud cannot find the corresponding “points” in the paired point cloud. If this issue can be tackled, many methods in image generation can also be applied in point cloud completion.
- View-based methods can be utilized to be combined with recently popular Neural Radiance Fields (NeRF) to reconstruct more realistic 3D shapes from the images.
- Novel generative models need to be integrated into point cloud completion, such as the diffusion model. Although a few works [37] have been made, the speed of generating samples is demanded to be improved.

- Moreover, establishing the novel loss functions is a significant challenge to be solved in the future, such as DCD loss [39].
- Although the DL-based point cloud completion has achieved impressive results, almost all the existing networks are conducted in the current datasets, such as PCN, ModelNet, and Completion3D. These datasets are derived from the CAD. Therefore, it is urgent to develop new datasets captured in the real world to make the networks more robust in the wild. Although unsupervised methods [99], [108], [109] have been developed, more effects should be paid on it because the captured point clouds from real-world could not obtain the ground truth.
- Due to the disorder and irregularity of point cloud, the early processing of point cloud is mainly voxelization. Still, this processing method will lead to the loss of effective information of the point cloud and increased computational complexity. Although feature extraction networks have been designed, such as PointNet and GCN, more effects should be paid to feature learning. In the decoder design, there are only fully connected networks, FoldingNet, and the newly-proposed transformer-based decoder network.
- Although there are remarkable achievements in 3-D DL models, including PointNet [18], PointNet++ [19], PointCNN [69], DGCNN [80], FoldingNet [48], PF-Net [103], PoinTr [33] and other work [113], [125], [126]. As transformer outperforms various of methods in computer vision, the transformer-based methods will be widely studied in the next few years.
- Limited networks can fulfill robust real-time completion tasks. In addition, the network training process is time-consuming. Research emphasis should be focused on lightweight and compact structure design.

## X. CONCLUSION

This paper has carried out a systematic review of the approaches for 3D point completion. Moreover, a comprehensive taxonomy and performance comparison of these approaches has been summed up. The advantages and limitations of each method are introduced, and the possible research directions are listed. This paper details DL’s research challenges and opportunities in point cloud completion to promote the potential developments. Nowadays, nearly all methods are striving to tackle the two main challenges: fully exploit the structural information and predict fine-grained complete 3D shapes. Although significant progress has been made, many efforts still need to be paid to make point cloud completion practical.

## REFERENCES

- [1] X. Huang, G. Mei, J. Zhang, and R. Abbas, “A comprehensive survey on point cloud registration,” 2021, *arXiv:2103.02690*.
- [2] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun, “Deep learning for 3D point clouds: A survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 12, pp. 4338–4364, Dec. 2021.
- [3] H. Lu and H. Shi, “Deep learning for 3D point cloud understanding: A survey,” 2020, *arXiv:2009.08920*.

- [4] Y. Cai, K.-Y. Lin, C. Zhang, Q. Wang, X. Wang, and H. Li, "Learning a structured latent space for unsupervised point cloud completion," 2022, *arXiv:2203.15580*.
- [5] Z. Fan, Y. He, Z. Wang, K. Wu, H. Liu, and J. He, "Reconstruction-aware prior distillation for semi-supervised point cloud completion," 2022, *arXiv:2204.09186*.
- [6] Z. Su, H. Huang, C. Ma, H. Huang, and R. Hu, "Point cloud completion on structured feature map with feedback network," 2022, *arXiv:2202.08583*.
- [7] C. Ma, Y. Yang, J. Guo, C. Wang, and Y. Guo, "Completing partial point clouds with outliers by collaborative completion and segmentation," 2022, *arXiv:2203.09772*.
- [8] Y. Wang, D. J. Tan, N. Navab, and F. Tombari, "Learning local displacements for point cloud completion," 2022, *arXiv:2203.16600*.
- [9] J. Tang, Z. Gong, R. Yi, Y. Xie, and L. Ma, "LAKe-Net: Topology-aware point cloud completion by localizing aligned keypoints," 2022, *arXiv:2203.16771*.
- [10] K. Zhang, X. Yang, Y. Wu, and C. Jin, "SRPCN: Structure retrieval based point completion network," 2022, *arXiv:2202.02669*.
- [11] D. Zong, S. Sun, and J. Zhao, "ASHF-Net: Adaptive sampling and hierarchical folding network for robust point cloud completion," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, 2021, pp. 3625–3632.
- [12] A. Dai, C. R. Qi, and M. NieBner, "Shape completion using 3D-encoder-predictor CNNs and shape synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5868–5877.
- [13] 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. (ICCV)*, Oct. 2017, pp. 85–93.
- [14] A. Sharma, O. Grau, and M. Fritz, "VConv-DAE: Deep volumetric shape learning without object labels," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 236–250.
- [15] D. Stutz and A. Geiger, "Learning 3D shape completion from laser scan data with weak supervision," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1955–1964.
- [16] Y. Liu, B. Fan, S. Xiang, and C. Pan, "Relation-shape convolutional neural network for point cloud analysis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8895–8904.
- [17] Z. Liu, H. Tang, Y. Lin, and S. Han, "Point-voxel CNN for efficient 3D deep learning," 2019, *arXiv:1907.03739*.
- [18] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 652–660.
- [19] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," 2017, *arXiv:1706.02413*.
- [20] T. Groueix, M. Fisher, V. G. Kim, B. C. Russell, and M. Aubry, "A papier-mâché approach to learning 3D surface generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 216–224.
- [21] X. Wen *et al.*, "PMP-Net++: Point cloud completion by transformer-enhanced multi-step point moving paths," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, May 15, 2022, doi: 10.1109/TPAMI.2022.3159003.
- [22] P. Mandikal and V. B. Radhakrishnan, "Dense 3D point cloud reconstruction using a deep pyramid network," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2019, pp. 1052–1060.
- [23] M. Sarmad, H. J. Lee, and Y. M. Kim, "RL-GAN-Net: A reinforcement learning agent controlled GAN network for real-time point cloud shape completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5898–5907.
- [24] L. P. Tchapmi, V. Kosaraju, H. Rezatofighi, I. Reid, and S. Savarese, "TopNet: Structural point cloud decoder," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 383–392.
- [25] W. Yuan, T. Khot, D. Held, C. Mertz, and M. Hebert, "PCN: Point completion network," in *Proc. Int. Conf. 3D Vis. (3DV)*, Sep. 2018, pp. 728–737.
- [26] J. Shi, L. Xu, P. Li, X. Chen, and S. Shen, "Temporal point cloud completion with pose disturbance," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4165–4172, Apr. 2022.
- [27] J. Zhang, W. Chen, Y. Wang, R. Vasudevan, and M. Johnson-Roberson, "Point set voting for partial point cloud analysis," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 596–603, Apr. 2021.
- [28] X. Wang, M. H. Ang, Jr., and G. H. Lee, "Cascaded refinement network for point cloud completion with self-supervision," 2020, *arXiv:2010.08719*.
- [29] L. Pan, "ECG: Edge-aware point cloud completion with graph convolution," *IEEE Robot. Autom. Lett.*, vol. 5, no. 3, pp. 4392–4398, Jul. 2020.
- [30] X. Wen, T. Li, Z. Han, and Y.-S. Liu, "Point cloud completion by skip-attention network with hierarchical folding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1939–1948.
- [31] 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.
- [32] A. X. Chang *et al.*, "ShapeNet: An information-rich 3D model repository," 2015, *arXiv:1512.03012*.
- [33] X. Yu, Y. Rao, Z. Wang, Z. Liu, J. Lu, and J. Zhou, "PoinTr: Diverse point cloud completion with geometry-aware transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 12498–12507.
- [34] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, 2013.
- [35] Z. Wu *et al.*, "3D ShapeNets: A deep representation for volumetric shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1912–1920.
- [36] L. Pan *et al.*, "Variational relational point completion network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 8524–8533.
- [37] Z. Lyu, Z. Kong, X. Xu, L. Pan, and D. Lin, "A conditional point diffusion-refinement paradigm for 3D point cloud completion," 2021, *arXiv:2112.03530*.
- [38] H. Fan, H. Su, and L. Guibas, "A point set generation network for 3D object reconstruction from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 605–613.
- [39] T. Wu, L. Pan, J. Zhang, T. Wang, Z. Liu, and D. Lin, "Density-aware chamfer distance as a comprehensive metric for point cloud completion," 2021, *arXiv:2111.12702*.
- [40] M. Tatarchenko, S. R. Richter, R. Ranftl, Z. Li, V. Koltun, and T. Brox, "What do single-view 3D reconstruction networks learn?" in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3405–3414.
- [41] R. Li, X. Li, C.-W. Fu, D. Cohen-Or, and P.-A. Heng, "PU-GAN: A point cloud upsampling adversarial network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7203–7212.
- [42] X. Wang, M. H. Ang, and G. H. Lee, "Cascaded refinement network for point cloud completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 790–799.
- [43] J. Gu *et al.*, "Weakly-supervised 3D shape completion in the wild," in *Proc. 16th Eur. Conf. Comput. Vis.* Glasgow, U.K.: Springer, Aug. 2020, pp. 283–299.
- [44] L. Yu, X. Li, C.-W. Fu, D. Cohen-Or, and P.-A. Heng, "PU-Net: Point cloud upsampling network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2790–2799.
- [45] Y. Chang, C. Jung, and Y. Xu, "FinerPCN: High fidelity point cloud completion network using pointwise convolution," *Neurocomputing*, vol. 460, pp. 266–276, Oct. 2021.
- [46] Y. Nie *et al.*, "Skeleton-bridged point completion: From global inference to local adjustment," 2020, *arXiv:2010.07428*.
- [47] W. Zhang, C. Long, Q. Yan, A. L. H. Chow, and C. Xiao, "Multi-stage point completion network with critical set supervision," *Comput. Aided Geometric Des.*, vol. 82, Oct. 2020, Art. no. 101925.
- [48] Y. Yang, C. Feng, Y. Shen, and D. Tian, "FoldingNet: Point cloud auto-encoder via deep grid deformation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 206–215.
- [49] Y. Zhao, T. Birdal, H. Deng, and F. Tombari, "3D point capsule networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1009–1018.
- [50] X. Zhang *et al.*, "View-guided point cloud completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 15890–15899.
- [51] Y. Xia, W. Liu, Z. Luo, Y. Xu, and U. Still, "Completion of sparse and partial point clouds of vehicles using a novel end-to-end network," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 2020, pp. 933–940, Aug. 2020.
- [52] Y. Xia, Y. Xia, W. Li, R. Song, K. Cao, and U. Still, "ASFM-Net: Asymmetrical Siamese feature matching network for point completion," 2021, *arXiv:2104.09587*.
- [53] A. Mendoza, A. Apaza, I. Sipiran, and C. Lopez, "Refinement of predicted missing parts enhance point cloud completion," 2020, *arXiv:2010.04278*.

- [54] Y. Miao, L. Zhang, J. Liu, J. Wang, and F. Liu, "An end-to-end shape-preserving point completion network," *IEEE Comput. Graph. Appl.*, vol. 41, no. 3, pp. 20–33, May 2021.
- [55] Y. Peng *et al.*, "Sparse-to-dense multi-encoder shape completion of unstructured point cloud," *IEEE Access*, vol. 8, pp. 30969–30978, 2020.
- [56] W. Zhang, Q. Yan, and C. Xiao, "Detail preserved point cloud completion via separated feature aggregation," in *Proc. 16th Eur. Conf. Comput. Vis.* Glasgow, U.K.: Springer, Aug. 2020, pp. 512–528.
- [57] X. Zhao, B. Zhang, J. Wu, R. Hu, and T. Komura, "Relationship-based point cloud completion," *IEEE Trans. Vis. Comput. Graph.*, early access, Sep. 3, 2021, doi: [10.1109/TVCG.2021.3109392](https://doi.org/10.1109/TVCG.2021.3109392).
- [58] R.-W. Li, B. Wang, C.-P. Li, L.-X. Zhang, and L. Gao, "High-fidelity point cloud completion with low-resolution recovery and noise-aware upsampling," 2021, *arXiv:2112.11271*.
- [59] T. Huang *et al.*, "RFNet: Recurrent forward network for dense point cloud completion," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 12508–12517.
- [60] Y. Zhao, J. Xie, J. Qian, and J. Yang, "PUI-Net: A point cloud upsampling and inpainting network," in *Proc. Chin. Conf. Pattern Recognit. Comput. Vis. (PRCV)*. Cham, Switzerland: Springer, 2020, pp. 328–340.
- [61] G. Li, Y. Chen, M. Cheng, C. Wang, and J. Li, "N-DPC: Dense 3D point cloud completion based on improved multi-stage network," in *Proc. 9th Int. Conf. Comput. Pattern Recognit.*, Oct. 2020, pp. 274–279.
- [62] Y. Sun, Y. Wang, Z. Liu, J. E. Siegel, and S. E. Sarma, "PointGrow: Autoregressively learned point cloud generation with self-attention," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2020, pp. 61–70.
- [63] J. Wang, Y. Cui, D. Guo, J. Li, Q. Liu, and C. Shen, "PointAttN: You only need attention for point cloud completion," 2022, *arXiv:2203.08485*.
- [64] Y. Shen, C. Feng, Y. Yang, and D. Tian, "Mining point cloud local structures by kernel correlation and graph pooling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4548–4557.
- [65] H. Huang, H. Chen, and J. Li, "Deep neural network for 3D point cloud completion with multistage loss function," in *Proc. Chin. Control Decis. Conf. (CCDC)*, Jun. 2019, pp. 4604–4609.
- [66] T. Hu, Z. Han, and M. Zwicker, "3D shape completion with multi-view consistent inference," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 10997–11004.
- [67] B. Gong, Y. Nie, Y. Lin, X. Han, and Y. Yu, "ME-PCN: Point completion conditioned on mask emptiness," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 12488–12497.
- [68] 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. Workshop (ICCVW)*, Oct. 2019, pp. 1–9.
- [69] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "PointCNN: Convolution on X-transformed points," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 820–830.
- [70] B.-S. Hua, M.-K. Tran, and S.-K. Yeung, "Pointwise convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 984–993.
- [71] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, and L. Guibas, "KPCConv: Flexible and deformable convolution for point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6411–6420.
- [72] W. Wu, Z. Qi, and L. Fuxin, "PointConv: Deep convolutional networks on 3D point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9621–9630.
- [73] J. Chibane, T. Alldieck, and G. Pons-Moll, "Implicit functions in feature space for 3D shape reconstruction and completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6970–6981.
- [74] L. Yi, B. Gong, and T. Funkhouser, "Complete & label: A domain adaptation approach to semantic segmentation of LiDAR point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 15363–15373.
- [75] H. Xie, H. Yao, S. Zhou, J. Mao, S. Zhang, and W. Sun, "GRNet: Gridding residual network for dense point cloud completion," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 365–381.
- [76] X. 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. (ICCV)*, Oct. 2021, pp. 13189–13198.
- [77] S. Liu, D. Li, W. Huang, Y. Cao, and S. Chen, "MRAC-Net: Multi-resolution anisotropic convolutional network for 3D point cloud completion," in *Proc. Pacific Rim Int. Conf. Artif. Intell.* Cham, Switzerland: Springer, 2021, pp. 403–414.
- [78] Y. Wang, D. J. Tan, N. Navab, and F. Tombari, "SoftPoolNet: Shape descriptor for point cloud completion and classification," in *Proc. 16th Eur. Conf. Comput. Vis.* Glasgow, U.K.: Springer, Aug. 2020, pp. 70–85.
- [79] X. Deng, X. Hu, N. E. Buris, P. An, and Y. Chen, "3D grid transformation network for point cloud completion," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2021, pp. 3642–3646.
- [80] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 1–12, 2019.
- [81] K. Zhang, M. Hao, J. Wang, C. W. de Silva, and C. Fu, "Linked dynamic graph CNN: Learning on point cloud via linking hierarchical features," 2019, *arXiv:1904.10014*.
- [82] K. Hassani and M. Haley, "Unsupervised multi-task feature learning on point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8160–8171.
- [83] K. Wang, K. Chen, and K. Jia, "Deep cascade generation on point sets," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, vol. 2, Aug. 2019, p. 4.
- [84] G. Qian, A. Abualshour, G. Li, A. Thabet, and B. Ghanem, "PU-GCN: Point cloud upsampling using graph convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11683–11692.
- [85] J. Shi, L. Xu, L. Heng, and S. Shen, "Graph-guided deformation for point cloud completion," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 7081–7088, Oct. 2021.
- [86] L. Zhu, B. Wang, G. Tian, W. Wang, and C. Li, "Towards point cloud completion: Point rank sampling and cross-cascade graph CNN," *Neurocomputing*, vol. 461, pp. 1–16, Oct. 2021.
- [87] H. Wu and Y. Miao, "LRA-Net: Local region attention network for 3D point cloud completion," in *Proc. 13th Int. Conf. Mach. Vis.*, Jan. 2021, Art. no. 1160519.
- [88] H. Wu and Y. Miao, "Cross-regional attention network for point cloud completion," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 10274–10280.
- [89] H. Wu, Y. Miao, and R. Fu, "Point cloud completion using multi-scale feature fusion and cross-regional attention," *Image Vis. Comput.*, vol. 111, Jul. 2021, Art. no. 104193.
- [90] Y. Zhang, D. Huang, and Y. Wang, "PC-RGNN: Point cloud completion and graph neural network for 3D object detection," 2020, *arXiv:2012.10412*.
- [91] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–9.
- [92] W. Wang, Q. Huang, S. You, C. Yang, and U. Neumann, "Shape inpainting using 3D generative adversarial network and recurrent convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2298–2306.
- [93] P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas, "Learning representations and generative models for 3D point clouds," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 40–49.
- [94] S. Gurumurthy and S. Agrawal, "High fidelity semantic shape completion for point clouds using latent optimization," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2019, pp. 1099–1108.
- [95] Y. Yu, Z. Huang, F. Li, H. Zhang, and X. Le, "Point encoder GAN: A deep learning model for 3D point cloud inpainting," *Neurocomputing*, vol. 384, pp. 192–199, Apr. 2020.
- [96] C. Chen, D. Liu, C. Xu, and T.-K. Truong, "GeneCGAN: A conditional generative adversarial network based on genetic tree for point cloud reconstruction," *Neurocomputing*, vol. 462, pp. 46–58, Oct. 2021.
- [97] X. Wen, Z. Han, Y.-P. Cao, P. Wan, W. Zheng, and Y.-S. Liu, "Cycle4Completion: Unpaired point cloud completion using cycle transformation with missing region coding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 13080–13089.
- [98] X. Chen, B. Chen, and N. J. Mitra, "Unpaired point cloud completion on real scans using adversarial training," 2019, *arXiv:1904.00069*.
- [99] J. Zhang *et al.*, "Unsupervised 3D shape completion through GAN inversion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 1768–1777.
- [100] V. Egiazarian *et al.*, "Latent-space Laplacian pyramids for adversarial representation learning with 3D point clouds," 2019, *arXiv:1912.06466*.
- [101] C.-L. Li, M. Zaheer, Y. Zhang, B. Poczos, and R. Salakhutdinov, "Point cloud GAN," 2018, *arXiv:1810.05795*.

- [102] C. Wen, B. Yu, and D. Tao, "Learning progressive point embeddings for 3D point cloud generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 10266–10275.
- [103] Z. Huang, Y. Yu, J. Xu, F. Ni, and X. Le, "PF-Net: Point fractal network for 3D point cloud completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 7662–7670.
- [104] M. Cheng, G. Li, Y. Chen, J. Chen, C. Wang, and J. Li, "Dense point cloud completion based on generative adversarial network," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–10, 2022.
- [105] X. Wang, M. H. Ang, and G. H. Lee, "Point cloud completion by learning shape priors," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 10719–10726.
- [106] L. Wang, Y. Huang, P. Tao, Y. Hou, and Y. Liu, "Learning geometry-image representation for 3D point cloud generation," 2020, *arXiv:2011.14289*.
- [107] W. Yan, R. Zhang, J. Wang, S. Liu, T. H. Li, and G. Li, "Vaccine-style-net: Point cloud completion in implicit continuous function space," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 2067–2075.
- [108] D. Valsesia, G. Fracastoro, and E. Magli, "Learning localized generative models for 3D point clouds via graph convolution," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–15.
- [109] D. Valsesia, G. Fracastoro, and E. Magli, "Learning localized representations of point clouds with graph-convolutional generative adversarial networks," *IEEE Trans. Multimedia*, vol. 23, pp. 402–414, 2021.
- [110] 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. (CVPR)*, Jun. 2021, pp. 4619–4628.
- [111] Y. Li and G. Baciu, "HSGAN: Hierarchical graph learning for point cloud generation," *IEEE Trans. Image Process.*, vol. 30, pp. 4540–4554, 2021.
- [112] P. Xiang *et al.*, "SnowflakeNet: Point cloud completion by snowflake point deconvolution with skip-transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 5499–5509.
- [113] X. Wen *et al.*, "PMP-Net: Point cloud completion by learning multi-step point moving paths," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 7443–7452.
- [114] P. Spurek *et al.*, "HyperPocket: Generative point cloud completion," 2021, *arXiv:2102.05973*.
- [115] M. Zamorski, M. Zięba, and J. Świątek, "Generative modeling in application to point cloud completion," in *Proc. Int. Conf. Artif. Intell. Soft Comput.* Cham, Switzerland: Springer, 2020, pp. 292–302.
- [116] P. Mittal, Y.-C. Cheng, M. Singh, and S. Tulsiani, "AutoSDF: Shape priors for 3D completion, reconstruction and generation," 2022, *arXiv:2203.09516*.
- [117] P. Esser, R. Rombach, and B. Ommer, "Taming transformers for high-resolution image synthesis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 12873–12883.
- [118] A. Vaswani *et al.*, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
- [119] A. Dosovitskiy *et al.*, "An image is worth  $16 \times 16$  words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.
- [120] N. Parmar *et al.*, "Image transformer," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 4055–4064.
- [121] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R. R. Martin, and S.-M. Hu, "PCT: Point cloud transformer," *Comput. Vis. Media*, vol. 7, no. 2, pp. 187–199, Jun. 2021.
- [122] X. Pan, Z. Xia, S. Song, L. E. Li, and G. Huang, "3D object detection with pointformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 7463–7472.
- [123] H. Zhao, L. Jiang, J. Jia, P. H. Torr, and V. Koltun, "Point transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 16259–16268.
- [124] J. Lin, M. Rickert, A. Perzylo, and A. Knoll, "PCTMA-Net: Point cloud transformer with morphing atlas-based point generation network for dense point cloud completion," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2021, pp. 5657–5663.
- [125] H. Son and Y. M. Kim, "SAUM: Symmetry-aware upsampling module for consistent point cloud completion," in *Proc. Asian Conf. Comput. Vis.*, 2020, pp. 1–17.
- [126] H. Wang, Q. Liu, X. Yue, J. Lasenby, and M. J. Kusner, "Unsupervised point cloud pre-training via occlusion completion," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9782–9792.
- [127] C. Bradley, G. W. Vickers, and M. Milroy, "Reverse engineering of quadric surfaces employing three-dimensional laser scanning," *Proc. Inst. Mech. Eng., B, J. Eng. Manuf.*, vol. 208, no. 1, pp. 21–28, Feb. 1994.
- [128] S. Son, H. Park, and K. H. Lee, "Automated laser scanning system for reverse engineering and inspection," *Int. J. Mach. Tools Manuf.*, vol. 42, no. 8, pp. 889–897, 2002.
- [129] C. Fröhlich and M. Mettenleiter, "Terrestrial laser scanning—New perspectives in 3D surveying," *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 36, no. 8, p. W2, 2004.
- [130] M. J. Olsen, F. Kuester, B. J. Chang, and T. C. Hutchinson, "Terrestrial laser scanning-based structural damage assessment," *J. Comput. Civil Eng.*, vol. 24, no. 3, pp. 264–272, 2010.
- [131] Q. Wang, Y. Tan, and Z. Mei, "Computational methods of acquisition and processing of 3D point cloud data for construction applications," *Arch. Comput. Methods Eng.*, vol. 27, pp. 479–499, Apr. 2020.
- [132] Z. Xing, S. Zhao, W. Guo, X. Guo, and Y. Wang, "Processing laser point cloud in fully mechanized mining face based on DGCNN," *ISPRS Int. J. Geo-Inf.*, vol. 10, no. 7, p. 482, Jul. 2021.
- [133] S. Schwarz *et al.*, "Emerging MPEG standards for point cloud compression," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 9, no. 1, pp. 133–148, Mar. 2018.
- [134] R. C. Luo, V. W. S. Ee, and C.-K. Hsieh, "3D point cloud based indoor mobile robot in 6-DoF pose localization using fast scene recognition and alignment approach," in *Proc. IEEE Int. Conf. Multisensor Fusion Integr. Intell. Syst. (MFIS)*, Sep. 2016, pp. 470–475.
- [135] S. Fekete, M. Diederichs, and M. Lato, "Geotechnical and operational applications for 3-dimensional laser scanning in drill and blast tunnels," *Tunnelling Underground Space Technol.*, vol. 25, no. 5, pp. 614–628, Sep. 2010.
- [136] R. Cheng, C. Agia, Y. Ren, X. Li, and L. Bingbing, "S3CNet: A sparse semantic scene completion network for LiDAR point clouds," 2020, *arXiv:2012.09242*.
- [137] X. Yan *et al.*, "Sparse single sweep LiDAR point cloud segmentation via learning contextual shape priors from scene completion," 2020, *arXiv:2012.03762*.
- [138] L. Roldao, R. de Charette, and A. Verroust-Blondet, "LMSCNet: Lightweight multiscale 3D semantic completion," in *Proc. Int. Conf. 3D Vis. (3DV)*, Nov. 2020, pp. 111–119.
- [139] M. Liang, B. Yang, S. Wang, and R. Urtasun, "Deep continuous fusion for multi-sensor 3D object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 641–656.



**Ben Fei** (Graduate Student Member, IEEE) received the M.S. degree from the Department of Material Science, Fudan University, Shanghai, China, in 2021, where he is currently pursuing the Ph.D. degree with the School of Computer Science. His research interests include generative model, point cloud process, transformer, and 3D vision. He is a Young Professionals of IEEE.



**Weidong Yang** received the Ph.D. degree in software engineering from Xidian University in 1999. He was a Post-Doctoral Researcher from the School of Computer Science, Fudan University, Shanghai, China, from 1999 to 2001. He is a Professor with the School of Computer Science, Fudan University. His research interests include big data, knowledge engineering, database and data mining, and software engineering.



**Wen-Ming Chen** (Member, IEEE) received the Ph.D. degree in bio-engineering from the National University of Singapore (NUS), Singapore, in 2011. From 2011 to 2016, he was a Research Fellow with The University of Melbourne, Australia. He has worked on both experimental and computational approach to the neuro-musculoskeletal systems of human locomotion. He is currently an Associate Professor with the Institute of Biomedical Engineering and Technology, Fudan University, Shanghai. He was a recipient of the Martyn Shorten Award by the International Society of Biomechanics (ISB) in 2011 and the Yamaguchi Medal by the Asia-Pacific Association for Biomechanics in Asia in 2015.



**Zhijun Li** (Fellow, IEEE) received the Ph.D. degree in mechatronics from Shanghai Jiao Tong University, Shanghai, China, in 2002. From 2003 to 2006, he was a Post-Doctoral Fellow with the University of Electro-Communications, Tokyo, Japan; and the National University of Singapore, Singapore. He is currently a Chair Professor with the University of Science and Technology, China, where he has been the Vice Dean of the School of Information Science and Technology since 2019. He has published over 400 articles, where the prestigious contributions were *Wearable Robotics* and *Bio-Mechatronics Systems*. He was a Distinguished Lecturer of RA, the 2016 National Distinguished Young Scholar (NSFC), and the Web of Science Highly Cited Researcher in 2019 and 2021. He received award at the 2018 National Ten-Thousand Talents Program in China.



**Yikang Li** received the B.S. degree in electronic engineering from Tsinghua University, Beijing, in 2016, and the Ph.D. degree in electronic engineering from The Chinese University of Hong Kong, in 2020. From 2019 to 2021, he was the Deputy Director of research and development on autonomous driving with SenseTime Technology Ltd. Since 2021, he has been a Research Scientist with the Shanghai AI Laboratory, Intelligent Transportation Research Center, China. His research interests include the 3D scene understanding, multi-modal fusion, scene reconstruction, autonomous driving simulation, and decision intelligence. His awards and honors include the National Scholarship from the Chinese Ministry of Education, the Hong Kong Ph.D. Fellowship from Hong Kong Government, and Microsoft Research Asia Ph.D. Fellowship.



**Tao Ma** received the B.E. degree in information engineering and the master's degree in pattern recognition from Northwestern Polytechnical University, Xi'an, China, in 2017 and 2020, respectively. He is currently a full-time Researcher with the Shanghai AI Laboratory, Autonomous Driving Group. His research interests include 3D object perception and multi-modal fusion.



**Xing Hu** received the Ph.D. degree in control science and engineering from Shanghai Jiao Tong University in 2016. He is an Associate Professor with the School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, China. His current research interests include image processing, computer vision, and machine learning.



**Lipeng Ma** received the B.S. degree in information security from the Nanjing University of Posts and Telecommunications (NUPT), Nanjing, China, in 2019. He is currently pursuing the Ph.D. degree in computer science with Fudan University. His research interests include knowledge graph application, pretrained language model, and AIOps.