Measurement Science
and Technology

100 year anniversary
Measurement Science and Technology

PURPOSE-LED
PUBLISHING™

PAPER

# CA-Net: a context-awareness and cross-channel attention-based network for point cloud understanding

To cite this article: Junting Lin *et al* 2025 *Meas. Sci. Technol.* **36** 045207

View the article online for updates and enhancements.

You may also like

- Omni-directional laser radar automatic acquisition system for complex industrial scenes: taking the pose measurement of hydraulic support as an example
Yichen Wang, Jiacheng Xie, Xuewen Wang et al.
- Robust multi-view PPF-based method for multi-instance pose estimation
Huakai Zhao, Yuning Gao, Mo Wu et al.
- DMS-SLAM: semantic visual SLAM based on deep mask segmentation in dynamic environments
Shuyuan Gao, Minhui Zhang, Xicheng Gao et al.


This content was downloaded from IP address 92.40.193.212 on 05/08/2025 at 22:01

IOP Publishing Measurement Science and Technology
Meas. Sci. Technol. **36** (2025) 045207 (11pp) https://doi.org/10.1088/1361-6501/adc029

# CA-Net: a context-awareness and cross-channel attention-based network for point cloud understanding

**Junting Lin[1,2,*], Jiping Zou[1], Ke Chen[2], Jinchuan Chai[3] and Jing Zuo[1]**

[1] School of Automation and Electrical Engineering, Lanzhou Jiaotong University, Lanzhou 730070, People's Republic of China
[2] Department of Mathematics and Statistics, University of Strathclyde, Glasgow G1 1XH, United Kingdom
[3] National Railway Track Test Center, China Academy of Railway Sciences Corporation Limited(CARS), Beijing 100015, People's Republic of China

E-mail: linjt@lzjtu.edu.cn, 11230422@stu.lzjtu.edu.cn, k.chen@strath.ac.uk, chaijinchuan@rails.cn and zuojing@lzjtu.edu.cn

Received 25 September 2024, revised 17 February 2025
Accepted for publication 13 March 2025
Published 24 March 2025

**Abstract**

To capture local complex shape information and fine-grained features from irregular point cloud, this paper proposes a novel local feature encoder-based network named context-awareness (CA)-Net which is used to solve the challenge of 3D object classification and segmentation. The core of the CA-Net is CA-Encoder, which is based on CA and cross-channel multi-head self-attention (CC-MSA). CA-Encoder uses contextual information awareness to aggregate local features from two levels: point cloud 3D coordinate information and high-dimensional implicitly encoded information, leveraging CC-MSA to learn channel-related information. For different point cloud benchmark tasks, CA-Net uses the local feature enhancement module (classification) and the Up-Transformer (segmentation) which includes a cascaded set of CA-Encoders to solve the problem of feature loss at non-edge points in edge sampling, so that the sampled results can both preserve the shape of the point cloud edges and reconstruct the full internal shape structure of the point cloud. The CA-Net has superior performance in experiments on ModelNet and ShapeNetPart datasets with a classification accuracy of 93.8% and a segmentation accuracy of 85.9%.

Keywords: point cloud, context awareness, attention mechanism, cascade encoder, edge sampling

## 1. Introduction

Point cloud are digitized representation on the surface of real 3D objects, and with their rich geometry, shape and structural details, they are extensively used in in several domains, including autonomous driving, intelligent welding [1, 2] and virtual reality. In practical applications, the raw point cloud data which is acquired by devices such as LIDAR or depth cameras is typically large and noisy with sparse and uneven distribution that poses a great challenge to high-level vision, including the point cloud classification, semantic segmentation, and target detection. How to downscale the point cloud while preserving its distribution and original geometric features across arbitrary scales is a basic and important task in 3D computer vision.

For the downsampling of point cloud, the classical sampling methods which are based on mathematical statistics include grid sampling [3], inverse density importance sampling [4], random sampling [5], uniform sampling (US)

* Author to whom any correspondence should be addressed.



© 2025 IOP Publishing Ltd. All rights, including for text and data mining, AI training, and similar technologies, are reserved.

[6] and farthest point sampling (FPS) [7]. FPS now is extensively used in point cloud processing tasks. Nevertheless, FPS only considers the point positions in Euclidean space, disregarding the geometric relationships among adjacent points. With the rapid advancement of deep learning technology, neural network-based point cloud sampling methods have emerged, which can overcome the limitations of traditional statistical sampling methods and further enhance the performance of downstream point cloud processing tasks. S-Net [8] generates a new small point cloud for a specific task, but there is no guarantee that the generated point cloud is a subset of the original point cloud. Over the foundation of S-Net, SampleNet [6] introduces a soft projection module to the matching step during the training process, i.e. a nearest-neighbor selection operation that makes the generated points closer to the original point cloud. CP-Net [9] proposes critical point layer, an adaptive global downsampling method with permutation-invariance and determinism, which samples points based on the importance of each point. DA-Net [10] proposes a density-adaptive downsampling method for point cloud classification tasks and improves the noise immunity of the model through local adjustment at the initial sampling points. MOPS-Net [11] generates a new sampling transform matrix as a sampled point cloud by learning a sampling transform matrix and multiplying it with the original point cloud. To summarize, the traditional sampling method lacks the adaptability to the distribution of point cloud data and cannot fully retain the key points. Deep learning methods do not consider the shape outlines of the point cloud as special features, and the local features of the point cloud are underutilized. The sampling method adopted in this paper builds upon edge sampling [12] by incorporating local feature enhancement (LFE) module. This approach effectively leverages the shape contour features of the point cloud while enhancing the local features during downsampling. As a result, the sampling outcome simultaneously can preserve both the internal structure features and the shape outlines of the point cloud.

For the point cloud upsampling, most of the earlier methods were based on optimization strategies with high computational complexity and often struggled to recover fine-grained structures. In recent years, deep neural network-based upsampling methods have emerged. PU-Net [13], a pioneering work in point cloud upsampling, introduces a hierarchical structure based on PointNet++[14] and employs Multi-Layer Perceptron (MLP) to expand the point set. EC-Net [15] minimizes the point-to-edge distance by defining an edge-aware joint loss function, which achieves edge-aware point cloud upsampling. MPU [16] proposes a multi-step progressive upsampling network that preserves local geometric features during point cloud upsampling. PU-generative adversarial network (GAN) [17] leverages a GAN to synthesize points in the latent space, but its results around fine details tend to be noise-laden. PU-CRN [18] proposes a cascaded refinement network that is both straightforward and efficient for point cloud upsampling. PU-graph convolutional network (GCN) [19] proposes a novel module named NodeShuffle which is based on GCN and further designs a feature extractor called Inception DenseGCN for multi-scale feature extraction task. In brief, early optimization-based methods have high computational complexity and lack global awareness. Although deep neural network-based methods can improve point cloud reconstruction accuracy, they are prone to lose local geometric structure features during the reconstruction process. This paper adopts a multi-encoder cascaded upsampling approach, in which the encoder leverages contextual awareness and attention mechanisms to enhance the learning capability of refined local features in point clouds. Meanwhile, the cascaded structure progressively refines the fine-grained features of the reconstructed point cloud, achieving a balance between efficiency and accuracy.

Deep learning-based methods are capable of handling more complex 3D object geometry structures and can extract depth features from point cloud data more efficiently. PointNet [20], a seminal work in point-based methods, takes raw point cloud as input and uses a series of MLPs and symmetric functions to extract features for classification. However, PointNet relies on global pooling, which limits its ability to capture local structural information. To address this problem, subsequent work has proposed various approaches based on local feature aggregation. PointNet++ gradually aggregates global and local features by introducing a hierarchical feature aggregation structure. DGCNN [21] introduces EdgeConv, a dynamic graph convolution operator that captures local geometric relationships. These methods use FPS to partition the point cloud into different local subsets and construct local aggregation operators to learn local shape representations, ultimately constructing hierarchical structures to learn shape perception from local to global. Point Transformer [22] is the first to apply self-attention mechanism to point cloud processing andfully leveraging the capability of Transformer to capture long-range dependencies. Point cloud transformer (PCT) [23] proposes an implicit Laplace operator and an offset attention module to capture global contextual features. According to the above development history, it can be observed that most early methods relied on hierarchical structures to aggregate local and global features. However, these methods struggled to capture long-range dependencies between points. Subsequent attention-based methods [24–27] not only effectively solve this problem but also provide continuous momentum for the advancement of the point cloud processing field. The feature encoder context-awareness (CA)-Encode used in this paper achieves multi-level feature aggregation by simultaneously capturing geometric structural information and high-dimensional encoded features. Meanwhile, it enhances the ability to capture long-range relationships between points by improving the attention mechanism.

In this paper, a point cloud processing network CA-Net is proposed based on local feature encoder CA-Encoder to capture local complex shape information and fine-grained features from the point cloud. CA-Encoder, the core component of CA-Net, comprises two components: CA and cross-channel

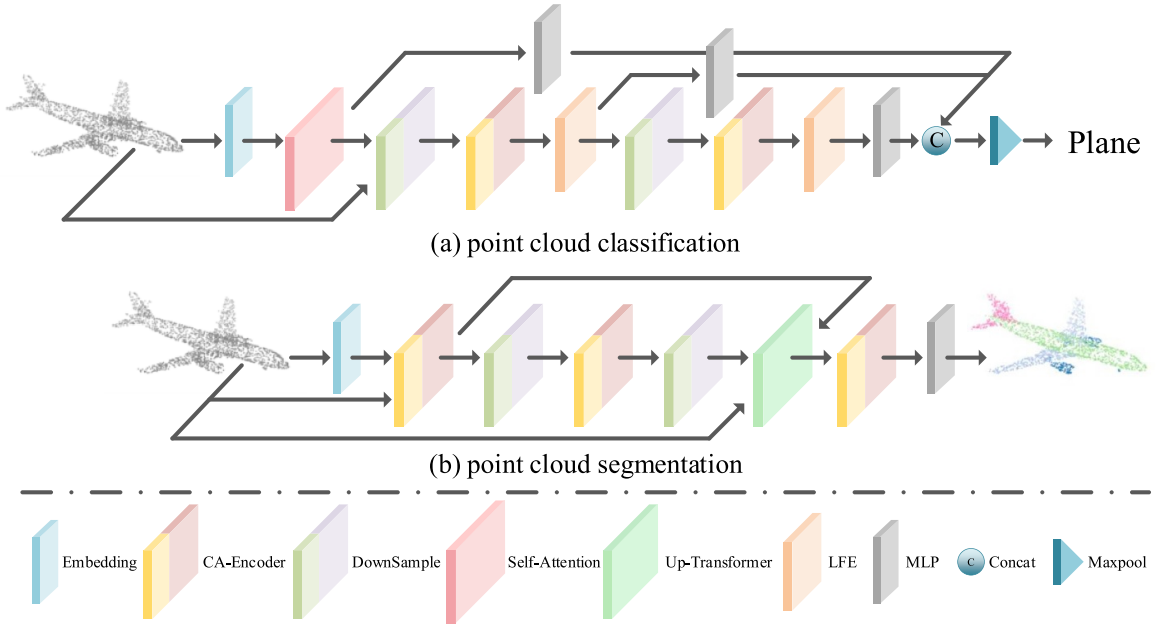



**Figure 1.** The network architecture for CA-Net.

multi-head self-attention (CC-MSA). CA initially aggregates local features by grouping and querying operations on the geometric coordinates and contextual features. Next, this module further utilizes the CC-MSA for local feature extraction. Based on the traditional attention mechanism, it enhances inter-channel information fusion by shifting attention channels. For the classification branch of point cloud processing, drawing inspiration from Edge-Conv [21], a local convolution with coordinate-based nearest neighbors is introduced as a LFE module. It performs feature enhancement on the results of edge downsampling to perceive local contextual feature information over a larger field of view. For the segmentation branch, an upsampling module (Up-Transformer) is devised based on the Transformer architecture. This module is designed to recover and reconstruct features at non-edge points leveraging the capacity of the Transformer to learn long-range dependencies.

The following is a summary of our contributions:

- This paper proposed a point cloud processing network CA-Net, which is based on a local feature encoder CA-Encoder. The objective of CA-Net is to learn rich context information and accurate shape perception from disordered and irregular point cloud.
- The core component of CA-Net is the CA-Encoder, which is based on CA and CC-MSA. In addition, the LFE module and Up-Transformer module are integrated to CA-Net to fulfill the requirements of different point cloud benchmark tasks.
- CA-Net achieved impressive results in classification and segmentation experiments on the ModelNet and ShapeNetPart datasets, with a classification accuracy of 93.8% and a segmentation accuracy of 85.9%.

## 2. Method

### 2.1. Network architecture

Figure 1(a) illustrates the network architecture for classification. First, the 3D coordinates of the original point cloud are converted into higher-dimensional implicitly encoded features through the processing of multiple convolution blocks in the embedding layer. Then, an attention mechanism is applied to aggregate local features, resulting in an initial feature representation of the point cloud. Subsequently, two rounds of downsampling (2048 $\rightarrow$ 1024 $\rightarrow$ 512) are performed using edge sampling methods. For each downsampled result, the CA-Encoder is employed to further aggregate local features, and the LFE module is used to enhance the feature map after downsampling. Finally, the feature maps are concatenated for subsequent classification tasks.

Figure 1(b) illustrates the network architecture for segmentation. First, the 3D coordinates of the original point cloud are converted into higher-dimensional implicitly encoded features through the processing of multiple convolution blocks in the embedding layer. These features are subsequently fed into the CA-Encoder to obtain an initial feature map. Following this, the point cloud undergoes two rounds of downsampling (2048 $\rightarrow$ 1024 $\rightarrow$ 512) using the edge sampling method and the 3D coordinated information from the second downsampling stage is passed into the Up-Transformer. On this basis, the coordinate and contextual feature of the original point cloud are residually connected with the upsampling results to recover the point feature information discarded during the downsampling of the point cloud. Eventually, the aggregated features are used for subsequent segmentation tasks.

### 2.2. CA-encoder

In general, a point cloud has two important types of contextual information: (1) the geometric structure in 3D space, referring to the position coordinates of each point in the original point set; (2) high-dimensional implicit encoding information, which are obtained after the original point cloud undergoes multi-layer convolution processing in the embedding module, representing the latent feature information of each point. Most of the early local feature extraction methods rely on multi-convolutional layer processing, which are limited in capturing both local geometric features and high dimensional latent features at multiple scales. Moreover, directly aggregating local features from the point cloud often fails to capture meaningful shape information. To address the aforementioned issues, this paper proposes the CA-Encoder, a local feature encoder based on attention mechanisms, which comprises two components: CA and CC-MSA.

*2.2.1. CA.* For CA of the original coordinates of the point cloud, each point $p_i \in R^3 \ (i = 1, 2, 3 \cdots n)$ in the original point set $P \in R^{n\times 3}$ is treated as a center point, the K-Nearest Neighbors (KNN) algorithm is then used to find its neighboring points. Based on the retrieved index information, a grouping operation is performed to generate a matrix $P_j \in R^{n\times k\times 3}$, which consists of the 3D coordinates of the $k$ nearest neighbors for each point. Next, expand the point set $P \in R^{n\times 3}$ and reshape it into $P_j \in R^{n\times 1\times 3}$.The relative positions $\Delta P$ can be calculated as:

$$\Delta P = P_j - P, \Delta P \in R^{n\times k\times 3}. \tag{1}$$

In addition to representing the local geometric contextual information of each point through its relative position $\Delta P$, the reshaped $P \in R^{n\times 1\times 3}$ is replicated $k$ times in the extended dimension to obtain $P \in R^{n\times k\times 3}$,which represents the global geometric contextual information. Finally, $\Delta P$ and $P$ are concatenated in the feature dimension to obtain the geometric contextual information $L_{\text{geo}}$:

$$L_{\text{geo}} = \text{concat}\left[\Delta P, P\right], L_{\text{geo}} \in R^{n\times k\times 6}. \tag{2}$$

For the high-dimensional implicitly encoded information obtained after being processed by the embedding module, a processing method similar to the one used for processing coordinate information is applied. First, the feature matrix $F_j \in R^{n\times k\times c}$ of all neighboring points is constructed through nearest neighbor search and grouping operation ($C$ is the feature dimension of implicitly encoded information). Then, expand the point feature set $F \in R^{n\times c}$ and reshape it into $F \in R^{n\times 1\times c}$. The local feature contextual information $\Delta F$ can be calculated as:

$$\Delta F = F_j - F, \Delta F \in R^{n\times k\times c}. \tag{3}$$

Next, $F \in R^{n\times 1\times c}$ is replicated $k$ times in the extended dimension to $F \in R^{n\times k\times 3}$,which represents the global feature contextual information. Finally, $\Delta F$ and $F$ are concatenated in the feature dimension to obtain the feature contextual information $\mathcal{L}_{\text{feat}}$:

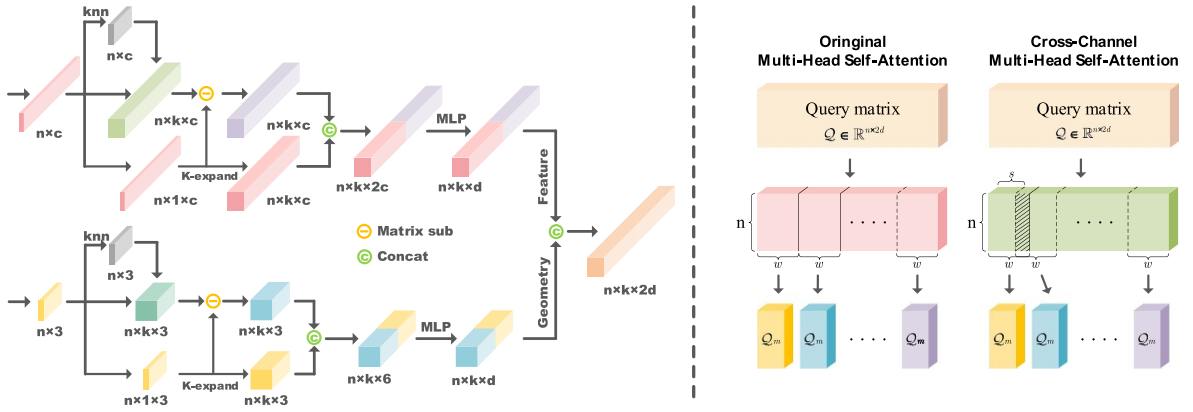
$$L_{\text{feat}} = \text{concat}\left[\Delta F, F\right], L_{\text{feat}} \in R^{n\times k\times 2c}. \tag{4}$$

On this basis, $L_{\text{geo}}$ and $L_{\text{feat}}$ are subjected to a convolution operation, and then concatenating the convolved results along the feature dimension to achieve comprehensive awareness of local contextual information at each point. Figure 2(a) illustrates the structure of the CA module.

*2.2.2. Cross-Channel Multi-head Self-Attention.* As one of the core operations in Transformer, MSA[28] mechanism facilitates the capture of capturing the dependencies between points during computation, while also segmenting independent feature channels for self-attention calculations in the corresponding heads. However, to denote the local features of the point cloud more comprehensively, the correlations between multiple segmented feature channels also needs to be considered. To achieve this objective, a new approach, namely CC-MSA is developed based on the regular MSA. First, the query matrix $\mathcal{Q}$, key matrix $\mathcal{K}$ and value matrix $\mathcal{V}$ are encoded by using a simple convolution operation. Then, splitting the corresponding submatrices $\mathcal{Q}_m$, $k_m$, $v_m$ from the initial matrix $\mathcal{Q}$, $\mathcal{K}$, $\mathcal{V}$ for each attention head. Specifically, as illustrated in figure 2(b), CC-MSA generates low-dimensional submatrices using a segmentation parameter $\varphi$, and the width of each segmentation channel can be expressed as $w = C/\varphi$ ($C$ is the dimension of the feature channel). Here, the offset of channel is set as $s = w/2 < w$. This ensures that there is a cross-over region between any two adjacent channels after segmentation. The number of attention heads can be denoted as $h = 2\varphi - 1$. On this basis, self-attention is applied to calculate the output $\mathcal{O}_m$ of each head. Due to the overlapping common areas between adjacent channels, the output $\mathcal{O}_m$ of each head contains contextual information from both its associated feature channel and neighboring feature channels, enabling the fusion of information across multiple segmentation channels. Finally, the outputs $\mathcal{O}_m$ from each head are concatenated along the feature dimension to obtain a more representative set of local features for the point cloud.

### 2.3. Up-Transformer for point upsampling

The attention-based edge sampling method proposed by APES is more focused on the preservation of edge points than the downsampling method FPS used in most neural network papers. Additionally, edge sampling can achieve downsampling at arbitrary scales, but the sampled results cannot maintain the same data distribution as the original point cloud. Therefore, after the cross attention-based upsampling layer, APES still has difficulty in reconstructing the feature information of non-edge points discarded during downsampling. Unlike previous works which achieved point cloud upsampling through complex network design and dedicated up-sampling strategies, this paper proposes a cascaded




(a) The structure of context aware module. (b) A comparison diagram of MSA and CC-MSA.

**Figure 2.** The core components of CA-encoder.

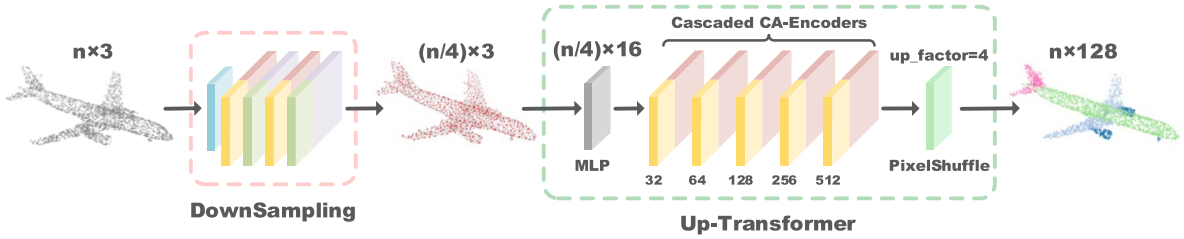



**Figure 3.** The diagrammatic picture of upsampling layer Up-Transformer.

transformer-based upsampling method. Specifically, as shown in figure 3, the 3D coordinates of the downsampled point cloud are first encoded to generate a feature map of size $n \times 16$. Subsequently, a local feature encoder CA-Encoder, is applied in a cascaded manner. It consists of two parts: CA and CC-MSA. This process gradually expands the feature dimensions of the initial feature map, with the output feature channels increasing as follows: $32 \rightarrow 64 \rightarrow 128 \rightarrow 256 \rightarrow 512$. Finally, by employing a periodic shuffling operation known as Pixel Shuffle [29], which does not require additional parameters, a dense point cloud feature map is generated with $n = 512 \times r$ and feature dimensions $C = 512/r$, where $r$ is the upsampling scale. On this basis, the 3D coordinate information of a dense point cloud are estimated using a simple MLP. Ultimately, the estimated dense 3D coordinates are fused with the original 3D coordinates through a residual link. Meanwhile, the multi-dimensional features of the dense point cloud are integrated with the preliminary local feature information, extracted from the original point cloud via the attention mechanism, serving as the final output of the 3D coordinates and local features of the point cloud.

### 2.4. LFE

At present, most deep learning-based point-and-process methods use FPS for downsampling operations. Although the edge sampling method can preserve the edge point features of the object in downsampling results at arbitrary scales, it cannot prevent the loss of non-edge point features within the object, which destroys the internal structure of the 3D object. Drawing inspiration from Edge-Conv, this paper introduces a local convolution with coordinate-based nearest neighbors as a LFE module. This module aims to compensate for the loss of features at non-edge points in edge sampling, and to perceive fine-grained local context on a larger scale. As stated in figure 4, the 3D coordinate points obtained after downsampling are treated as centroids. For each centroid, the KNN algorithm is used to search for the indices of its neighboring points. According to the index information, LFE extracts the feature of the neighboring points within the neighborhood of each point and compute the difference feature tensor between each point and its $k$ neighbors. Then, connect the initial feature tensor and the difference feature tensor along the feature dimension. Based on this, the splicing tensor is dimensionally reduced by a single layer of MLP and aggregated local features using maximum pooling. Finally, the feature tensor with the desired dimension is produced through an MLP.

## 3. Experimental design and results analysis

### 3.1. Evaluation metrics

To evaluate the classification and segmentation performance of the CA-Net, this paper adopts the main evaluation metrics commonly used in most existing point cloud benchmark tasks, including overall accuracy (OA), instance mean intersection over union (Ins. mIoU) and category mean intersection over union (Cat. mIoU), which are computed by the following formulas:

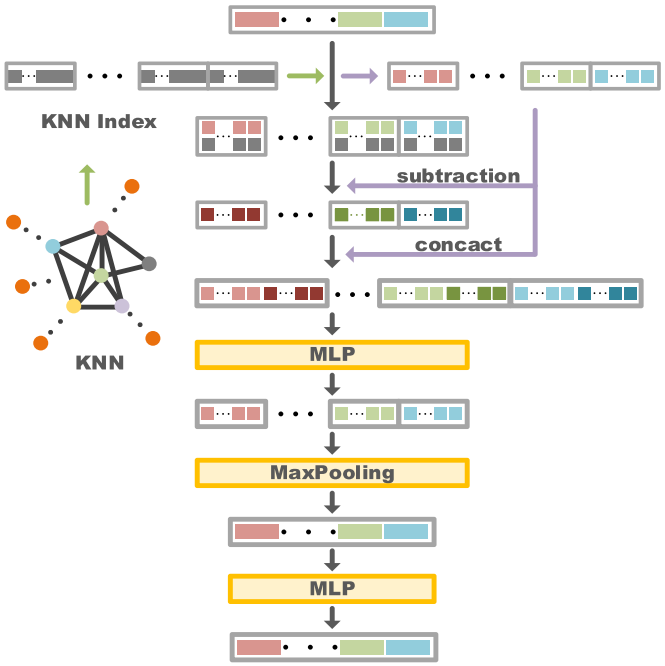



**Figure 4.** The structure of local feature enhancement (LFE).

$$\mathrm{OA} = \frac{\sum_{i=1}^{c} \mathrm{TP}_i}{\sum_{i=1}^{c} (\mathrm{TP}_i + \mathrm{FP}_i + \mathrm{TN}_i + \mathrm{FN}_i)} \tag{5}$$

$$\mathrm{Ins.mIoU} = \frac{\sum_{i=1}^{c}\sum_{j=1}^{n} \mathrm{TP}_j}{\sum_{i=1}^{c}\sum_{j=1}^{n} \mathrm{TP}_j + \sum_{i=1}^{c}\sum_{j=1}^{n} \mathrm{FP}_j + \sum_{i=1}^{c}\sum_{j=1}^{n} \mathrm{FN}_j} \tag{6}$$

$$\mathrm{Cat.mIoU} = \frac{1}{c}\sum_{i=1}^{c} \frac{\sum_{j=1}^{n} \mathrm{TP}_j}{\sum_{j=1}^{n} \mathrm{TP}_j + \sum_{j=1}^{n} \mathrm{FP}_j + \sum_{j=1}^{n} \mathrm{FN}_j}. \tag{7}$$

Among them, $c$ is the number of divided categories, $n$ is the number of points in each category. TP is a positive sample that is predicted to be a positive category, FP is a negative sample that is predicted to be a positive category, TN is a negative sample predicted to be a negative category and FN is a positive sample predicted to be a negative category.

### 3.2. Classification

**Dataset.** ModelNet40 [30] is a widely used benchmark dataset in the field of 3D object shape recognition and classification. During the experiments, this dataset is split into two parts according to the official splitting method. A US method is applied to extract 2048 points from the surface of each object, which are then used as the original input to the classification network. For evaluation, the experiments use OA as a measure to compare with the previous work.

**Table 1.** Classification results on ModelNet40.

| Modle | OA (%) |
|---|---|
| PointNet [20] | 89.2 |
| PointNet++ [14] | 91.9 |
| SpiderCNN [31] | 92.4 |
| DGCNN [21] | 92.9 |
| PointCNN [32] | 92.2 |
| PointConv [33] | 92.5 |
| KPConv [34] | 92.9 |
| APES [12] | 93.2 |
| PCT [23] | 93.2 |
| CurveNet [35] | 93.8 |
| DeltaConv [36] | 93.8 |
| Ours | **93.8** |

**Quantitative and qualitative results.** Table 1 displays the classification quantitative comparison results (Please be advised that the voting strategy is not considered in the results). It is noteworthy that our method has obtained superior accuracy in comparison to APES global-based and local-based classification and is comparable to the SOTA classification methods. Figure 5 shows a qualitative comparison of the downsampled results for APES and CA-Net. It can be observed that the APES network overly emphasizes the sampling of edge points. Although it preserves the local sharp edge details of the object, it results in the loss of geometric features of non-edge points inside the object and fails to maintain the integrity of the edge boundaries. The downsampled results obtained by CA-Net can not only show the geometrically detailed features of the object edges excellently and ensure the integrity of the object edge lines, but also retain the structural features inside the object, which effectively improves the performance of the classification task.

### 3.3. Part segmentation

**Dataset.** The ShapeNetPart [37] dataset is a subset of ShapeNet, mainly used for segmentation task of objects on component level. ShapeNetPart comprises a total of 16 881 models in 16 categories (e.g. aircraft, cars, chairs, etc.). Each object is subdivided into different parts. For example, an airplane may be divided into different parts such as wings, fuselage and tail, and a chair may be divided into chair back, chair seat and chair legs. During the experiments, this dataset is split into two parts according to the official splitting method. For evaluation, the experiments mainly use instance mIoU as a measure to compare with the previous work.

**Quantitative and qualitative results.** Table 2 displays the segmentation quantitative comparison results. It is noteworthy that our method has superior performance in the segmentation task and improves 1% in comparison with APES. This

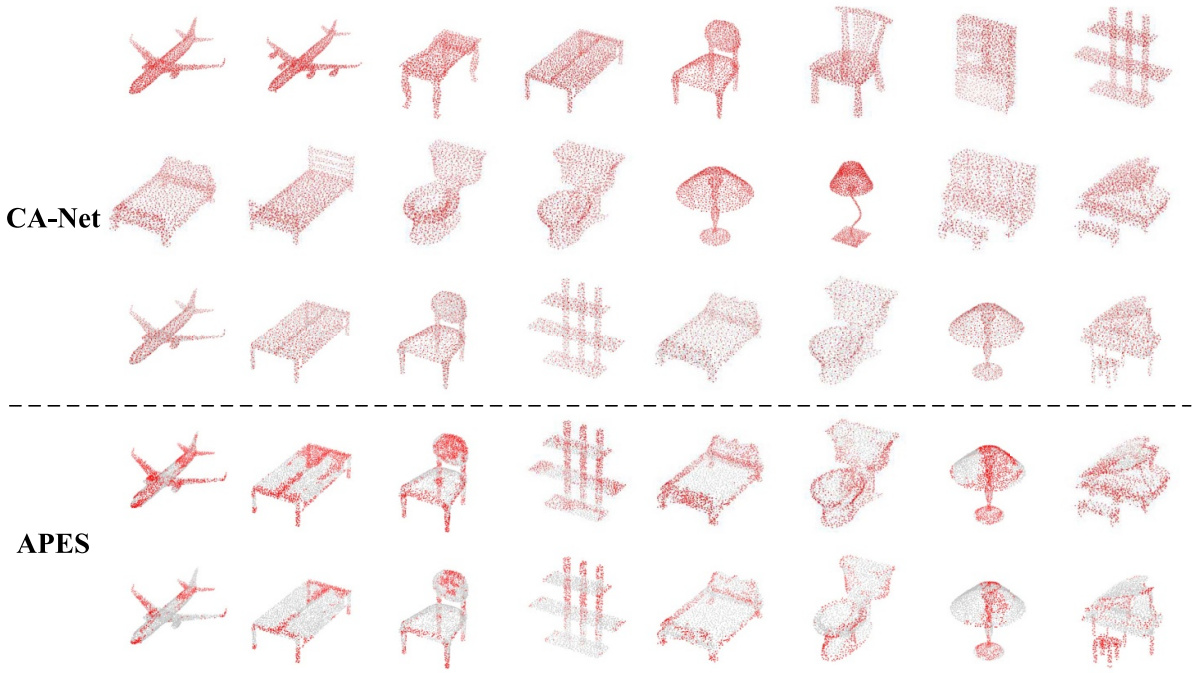



**Figure 5.** Visualization results of sampling on CA-Nets and APES in different shapes.

**Table 2.** Segmentation results on shapenetpart.

| Modle | Segmentation | |
|---|---|---|
| | Cat. mIoU (%) | Ins. mIoU (%) |
| PointNet [20] | 80.4 | 83.7 |
| PointNet++ [14] | 81.9 | 85.1 |
| SpiderCNN [31] | 82.4 | 85.3 |
| DGCNN [21] | 82.3 | 85.2 |
| PointCNN [32] | 84.6 | 86.1 |
| PointConv [33] | 82.8 | 85.7 |
| KPConv [34] | 85.0 | 86.2 |
| APES [12] | 83.1 | 84.9 |
| PCT [23] | — | 86.2 |
| CurveNet [35] | — | 86.5 |
| StratifiedTransformer [38] | **85.1** | **86.6** |
| Ours | 83.7 | 85.9 |

demonstrates Up-Transformer can effectively solve the problem that the upsampling layer in APES is not able to effectively reconstruct the dropped non-edge features. Specifically, CA-Net aggregates local features through multiple CA-Encoders, so that downsampled results obtained after two edge samplings are reconstructed to a certain extent to the same structure as the original point cloud data distribution. The downsampled point cloud reconstructed segmentation results are shown in figure 6.

### *3.4. Computational efficiency analysis*

This section compares CA-Net with five other classic point cloud networks and analyzes computational efficiency based on the comparison results. The comparison includes network parameters, training and testing time for classification and segmentation tasks. All experiments in this section were conducted in an Ubuntu 20.04 (64-bit) environment with Python 3.7, PyTorch 1.13.1, and a GeForce RTX 4090 Ti GPU. Training Parameters: epoch = 200, batch size = 8.

According to table 3, compared to other networks, especially APES, CA-Net achieves higher classification and segmentation accuracy without significantly increasing training and testing time, demonstrating competitive computational efficiency. Additionally, the smaller number of parameters reflects the lightweight nature of the network architecture. Although KPConv has higher computational efficiency and segmentation accuracy, it requires a longer training period (max epoch = 500) to reach peak performance in both classification and segmentation tasks. Overall, CA-Net achieves a well-balanced trade-off between computational efficiency and processing performance.

## 4. Ablation study

In this section, multiple ablation studies are conducted to explore the design choices of the point cloud processing network architectures.

### *4.1. Classification on ModelNet*

**Feature learning layer.** In this paper, the feature learning layer which is used CA-Net is CA-Encoder, a local feature encoder combining local CA and CC-MSA. APES uses the feature learning layer N2PAttention in a branch which is local classification. CA-Net can also use other layers designed for similar purposes in literature, e.g. EdgeConv, KpConv. The

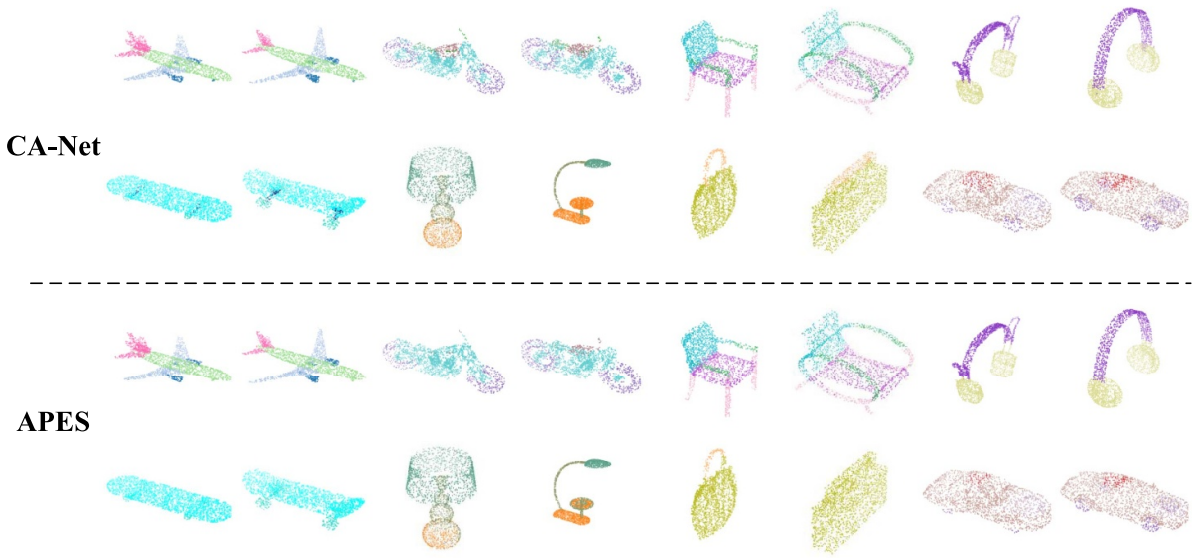



**Figure 6.** Visualization results of part segmentation after point cloud sampling.

**Table 3.** Computational efficiency of different networks on classification and segmentation tasks.

| | | Classification on ModelNet40 | | | Segmentation on ShapeNet | | |
|---|---|---|---|---|---|---|---|
| Model | Parameters (*M*) | Train (s) | Test (s) | OA (%) | Train (s) | Test (s) | Ins. mIoU (%) |
| PointNet | **3.48** | 37 | 6 | 89.2 | 49 | 12 | 83.7 |
| PointNet++ | 1.48 | 80 | **19** | 91.9 | **184** | **28** | 85.1 |
| DGCNN | 1.84 | 84 | 11 | 92.9 | 115 | 17 | 85.2 |
| KPConv | 1.73 | 32 | 11 | 92.9 | — | — | **86.2** |
| APES | 1.67 | 88 | 12 | 93.4 | 160 | 20 | 84.9 |
| Ours | 1.67 | **92** | 12 | **93.8** | 163 | 21 | 85.9 |

**Table 4.** Ablation study on the use of different feature learning layers.

| Method | Feature learning layer | OA (%) |
|---|---|---|
| APES [12] | N2PAttention | 93.4 |
| DGCNN [21] | EdgeConv | 92.5 |
| KPConv [34] | KpConv | 93.1 |
| CA-Net | CA-Encoder | 93.8 |

**Table 5.** Ablation study of using a different number of embedding dimensions.

| Embedding dimension | OA (%) |
|---|---|
| 64 | 92.3 |
| 128 | 93.8 |
| 192 | 93.2 |

results obtained by using different feature learning layers for classification experiments are shown in table 4. It can be observed that the best performance of the point cloud processing network is achieved by using CA encoder as the feature learning layer.

**Embedding dimension.** The raw data in the experimental dataset consists of point sets with three-dimensional coordinate information, characterized by sparsity and disorder. To facilitate more effective feature extraction and learning in subsequent network layers, the raw point cloud data is usually mapped to a high-dimensional feature space through an embedding module to generate a preliminary point cloud feature map. Therefore, the choice of embedding dimensions is a key factor affecting subsequent performance. Currently, most point cloud processing networks achieve better performance when using larger embedding dimensions. Table 5 compares the classification results with embedding dimensions of 64, 128, and 192. Based on the comparison, it can be observed that the classification performance of the proposed CA-Net is optimal when the embedding dimension is set to the default value of 128.

**Network architecture.** Comparing CA-Net with the classification network architecture of APES, CA-Net has two main differences: (1) feature learning layer. The feature learning layer applied on local classification of APES is N2PAttention. The feature learning layer used in this paper is CA-Encoder, a local feature encoder which combines local CA and cross-channel multicast self-attention. (2)Network architecture. CA-Net

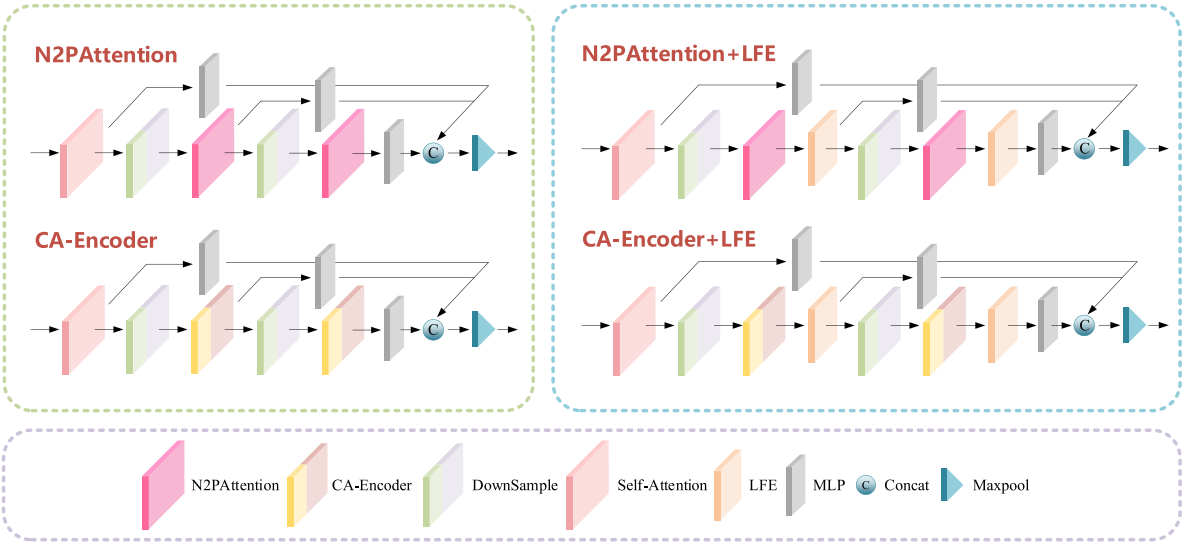



**Figure 7.** Four different network architectures for the classification task.

**Table 6.** Ablation study of using the different network architectures.

| | CA-Encoder | LFE | OA (%) |
|---|---|---|---|
| Network Architecture Design | × | × | 93.1 |
| | × | √ | 93.4 |
| | √ | × | 93.5 |
| | √ | √ | 93.8 |

**Table 7.** Ablation study of using a different number of neighbors for local-based feature encoder.

| $k$ | 16 | 20 | 30 | 32 | 40 |
|---|---|---|---|---|---|
| OA (%) | 93.2 | 93.4 | 93.8 | 93.6 | 93.1 |

adds LFE after using an attention mechanism for preliminary feature aggregation of the edge sampling results.

There are four different network structures which are designed to explore the effect of LFE and CA-Encoder on the classification results. The network structures are shown in figure 7. The classification results which are obtained from different network structures are shown in table 6. Comparing the classification results, it is revealed that the combination of LFE and CA-Encoder can effectively improve the performance of the network.

**Choice of $k$ in nearest neighbor search.** The local feature encoder which is proposed in this paper needs to use the KNN algorithm to search the neighbors for each point $p_i \in R^3(i = 1,2,3\cdots n)$ in the original point set $P$ when performing CA. The number of neighbors $k$, as a critical parameter, determines the size of the local feature awareness region for a point. A comparison of the results obtained for setting different numbers of neighbors $k$ is shown in table 7.

According to the comparison, it can be observed that as the value of $k$ increases from 16 to 30, the receptive field of the CA module continues to expand, enabling the network to extract more comprehensive local features, leading to improved classification accuracy. However, when $k$ is increased beyond 30, the receptive field becomes excessively large, introducing noise and redundant information that interferes with the extraction of fine-grained local features, ultimately degrading performance. Therefore, setting $k = 30$ achieves the optimal balance between receptive field size and local feature extraction in our experiments.

### 4.2. Segmentation on ShapeNetPart

**Selection of segmentation parameter $\varphi$.** The local feature encoder which is proposed in this paper needs to select an appropriate segmentation parameter $\varphi$ to segment the feature channels when enhancing features based on CC-MSA with contextual information. Then, the offset $s$ of the channel is set by using $\varphi$ as a reference and compute the number of attention heads $h$. As $\varphi$ increases, the segmentation channel width $\omega$ decreases and the number of attention heads $h$ increases. Therefore, it can help the model in accurately capturing features and improve the ability of the model to learn fine-grained features. However, when $\varphi$ is set to an excessively large value, the features obtained by each attention head are limited and the model is difficult to effectively capture the global features and cross-channel dependencies of the input data. At the same time, the computational complexity of the model increases significantly. So, the selection of $\varphi$ is not the larger the better. The segmentation results obtained by using different sizes of the segmentation parameter $\varphi$ are shown in table 8. Where $C$ is the number of feature channels of the point cloud.

**Upsampling layer.** In addition to the feature learning layer, the difference between the segmentation networks of CA-Net and APES is that the two segmentation networks use different upsampling layers. APES uses an upsampling layer which is based on cross-attention, CA-Net uses an upsampling layer Up-Transformer. Specifically, Up-Transformer progressively

**Table 8.** Ablation study on the effect of different segmentation parameters in cross-channel multi-head self-attention.

| $\varphi$ | $\omega$ | $h$ | Ins. mIoU(%) |
|---|---|---|---|
| 2 | $w = C/\varphi$ | $h = 2\varphi - 1$ | 85.5 |
| 4 | | | 85.9 |
| 6 | | | 85.4 |

**Table 9.** Ablation study of using different upsampling layers.

| Method | Ins. mIoU (%) |
|---|---|
| APES [12] | 84.9 |
| iPUNet [39] | 84.3 |
| CRNet [19] | 83.7 |
| Up-Transformer | 85.9 |

expands the feature dimension of the initial feature map via the cascaded CA-Encoders and aggregates a more comprehensive feature map. Similarly, the network can use other upsampling methods in the literature to reconstruct the features of down-sampled point cloud. A comparison of the instance mIoU that are obtained by using different up-sampling methods for point cloud segmentation is shown in table 9. The comparison reveals that the best performance is achieved by the point cloud segmentation network which uses the Up-Transformer as the upsampling layer.

## 5. Conclusion

This paper proposes a novel point cloud understanding network CA-Net, based on a local feature encoder called CA-Encoder. As the core component of the network, CA-Encoder consists of two components: CA and CC-MSA. CA can capture both the geometric structure and the encoded feature of the point cloud, enabling the network to aggregate fine-grained local features more effectively. Built upon the conventional attention mechanism, CC-MSA introduces offsets to the feature channels, adding correlation information among multiple channels into attention output. It further enhances the feature learning capability of the network.

For the point cloud classification task, the network introduces an LFE module to enhance the features of the down-sampled results, compensating for the loss of non-edge point features during the sampling process. For the segmentation network, a cascaded upsampling module (Up-Transformer) with multiple feature encoders is employed to achieve fine-grained point cloud feature reconstruction.

In summary, the network fully considers the balance between edge point and non-edge point feature information during the sampling process, achieving significant performance improvements in both point cloud classification and segmentation tasks. In future work, it would be worthwhile to further investigate upsampling strategies tailored to the edge-sampled points, whose distribution differs from that of the original point cloud, to achieve better segmentation performance.

## Data availability statement

The data cannot be made publicly available upon publication because they are owned by a third party and the terms of use prevent public distribution. The data that support the findings of this study are available upon reasonable request from the authors.

## Acknowledgment

This work was supported by the National Natural Science Foundation of China (No.52162050).

## Conflict of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## ORCID iDs

Junting Lin https://orcid.org/0000-0002-5763-5256
Jiping Zou https://orcid.org/0009-0007-5471-5094
Ke Chen https://orcid.org/0000-0002-6093-6623
Jinchuan Chai https://orcid.org/0000-0003-0714-0172
Jing Zuo https://orcid.org/0000-0002-3696-4544

## References

[1] Guo Q, Yang Z, Xu J, Jiang Y, Wang W, Liu Z, Zhao W and Sun Y 2024 Progress, challenges and trends on vision sensing technologies in automatic/intelligent robotic welding: state-of-the-art review *Robot. Comput. Integr. Manuf.* **89** 102767
[2] Wang H, Rong Y, Xu J, Huang Y and Zhang G 2025 Application and trends of point cloud in intelligent welding: state of the art review *J. Manuf. Syst.* **79** 48–72
[3] Jaderberg M, Simonyan K and Zisserman A 2015 Spatial transformer networks *Advances in Neural information Processing Systems* vol 28 (https://doi.org/10.48550/arXiv.1506.02025)
[4] Groh F *et al* 2018 Flex-convolution—million-scale point-cloud learning beyond grid-worlds *Asian Conf. on Computer Vision* (https://doi.org/10.48550/arXiv.1803.07289)
[5] Hu Q *et al* 2020 Randla-net: efficient semantic segmentation of large-scale point clouds *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition* (https://doi.org/10.48550/arXiv.1911.11236)
[6] Lang I *et al* SampleNet: differentiable point cloud sampling 2019 *2020 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 7575–85
[7] Zhao H *et al* 2019 PointWeb: enhancing local neighborhood features for point cloud processing *2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 5560–8
[8] Dovrat O *et al* 2018 Learning to Sample *2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 2755–64
[9] Nezhadarya E *et al* 2019 Adaptive hierarchical down-sampling for point cloud classification *2020 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 12953–61

[10] Fu J *et al* 2018 Dual attention network for scene segmentation *2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 3141–9
[11] Qian Y *et al* 2020 MOPS-net: a matrix optimization-driven network fortask-oriented 3D point cloud downsampling (arXiv:2005.00383)
[12] Chengzhi W *et al* 2023 Attention-based point cloud edge sampling *2023 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 5333–43
[13] Yu L *et al* 2018 PU-net: point cloud upsampling network *2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition* pp 2790–9
[14] Qi C *et al* 2017 PointNet++: deep hierarchical feature learning on point sets in a metric space *Neural Information Processing Systems* (https://doi.org/10.48550/arXiv.1706.02413)
[15] Yu L *et al* 2018 EC-Net: an edge-aware point set consolidation network.*European Conf. on Computer Vision* (https://doi.org/10.48550/arXiv.1807.06010)
[16] Wang Y *et al* 2018 Patch-based progressive 3D point set upsampling *2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 5951–60
[17] Li R *et al* 2019 PU-GAN: a point cloud upsampling adversarial network *2019 IEEE/CVF Int. Conf. on Computer Vision (ICCV)* pp 7202–11
[18] Du H *et al* 2022 Point cloud upsampling via cascaded refinement network *Asian Conf. on Computer Vision* (https://doi.org/10.48550/arXiv.2210.03942)
[19] Qian G *et al* 2021 Pu-gcn: point cloud upsampling using graph convolutional networks *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition* (https://doi.org/10.48550/arXiv.1912.03264)
[20] Qi C *et al* 2016 PointNet: deep learning on point sets for 3D classification and segmentation *2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 77–85
[21] Wang Y *et al* 2018 Dynamic graph CNN for learning on point clouds *ACM Trans. on Graphics (TOG)* 38 pp 1–12
[22] Zhao H *et al* 2020 Point transformer *2021 IEEE/CVF Int. Conf. on Computer Vision (ICCV)* pp 16239–48
[23] Guo M-H, Cai J-X, Liu Z-N, Mu T-J, Martin R R and Hu S-M 2020 PCT: point cloud transformer *Comput. Vis. Media* **7** 187–99
[24] Lu D, Yang L, Wang B, Wu K, Li Y and Zheng X 2022 3DGTN: 3-D dual-attention glocal transformer network for point cloud classification and segmentation *IEEE Trans. Geosci. Remote Sens.* **62** 1–13
[25] Zhang D *et al* 2020 Feature pyramid transformer *Computer Vision–ECCV 2020: 16th European Conf. Proc. Part XXVIII 16 (Glasgow, UK, 23–28 August 2020)* (Springer) (https://doi.org/10.48550/arXiv.2007.09451)
[26] Wu X *et al* 2022 Point transformer v2: grouped vector attention and partition-based pooling *Advances in Neural Information Processing Systems* 35 pp 33330–42
[27] Xumin Y *et al* 2022 Point-bert: pre-training 3d point cloud transformers with masked point modeling *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition* (https://doi.org/10.48550/arXiv.2111.14819)
[28] Vaswani A *et al* 2017 Attention is all you need *Neural Information Processing Systems* (https://doi.org/10.48550/arXiv.1706.03762)
[29] Shi W *et al* 2016 Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network *2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 1874–83
[30] Wu Z *et al* 2014 3D shapenets for 2.5 d object recognition and next-best-view prediction (arXiv:1406.5670 2.4)
[31] Xu Y *et al* Spidercnn: deep learning on point sets with parameterized convolutional filters 2018 *Proc. of the European Conf. on Computer Vision (ECCV)* (https://doi.org/10.48550/arXiv.1803.11527)
[32] Li Y *et al* 2018 PointCNN: convolution on X-transformed points *Neural Information Processing Systems* (https://doi.org/10.48550/arXiv.1801.07791)
[33] Wu W *et al* 2018 PointConv: deep convolutional networks on 3D point clouds *2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 9613–22
[34] Thomas H *et al* 2019 KPConv: flexible and deformable convolution for point clouds *2019 IEEE/CVF Int. Conf. on Computer Vision (ICCV)* pp 6410–9
[35] Muzahid A A M, Wan W, Sohel F, Wu L and Hou L 2021 CurveNet: curvature-based multitask learning deep networks for 3D object recognition *IEEE CAA J. Autom. Sin.* **8** 1177–87 (available at: https://api.semanticscholar.org/CorpusID:22658218)
[36] Wiersma R 2022 DeltaConv: anisotropic geometric deep learning with exterior calculus (https://doi.org/10.48550/arXiv.2111.08799)
[37] Yi L *et al* 2016 A scalable active framework for region annotation in 3D shape collections *ACM Trans. on Graphics (TOG)* **35** 1–12
[38] Lai X *et al* 2022 Stratified transformer for 3D point cloud segmentation *2022 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 8490–9
[39] Wei G, Pan H, Zhuang S, Zhou Y and Li C 2023 iPUNet: iterative cross field guided point cloud upsampling *IEEE Trans. Vis. Comput. Graph.* **30** 6089–103