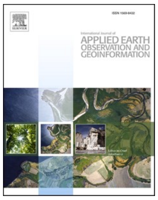
Contents lists available at ScienceDirect

International Journal of Applied Earth Observation and Geoinformation

journal homepage: www.elsevier.com/locate/jag



MVPNet: A multi-scale voxel-point adaptive fusion network for point cloud semantic segmentation in urban scenes

Huchen Li [a], Haiyan Guan [a,*], Lingfei Ma [b,*], Xiangda Lei [a], Yongtao Yu [c], Hanyun Wang [d], Mahmoud Reza Delavar [e], Jonathan Li [f,g]

[a] *School of Remote Sensing and Geomatics Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China*
[b] *School of Statistics and Mathematics, Central University of Finance and Economics, Beijing 102206, China*
[c] *Faculty of Computer and Software Engineering, Huaiyin Institute of Technology, Huaian 223003, China*
[d] *School of Surveying and Mapping, Information Engineering University, Zhengzhou 450000, China*
[e] *College of Engineering, University of Tehran, Tehran 1439951154, Iran*
[f] *Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada*
[g] *Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada*

ARTICLE INFO

Keywords:
Point cloud
Multi-scale voxel
Semantic segmentation
Geometric self-attention
Pyramid decoder

ABSTRACT

Point cloud semantic segmentation, which contributes to scene understanding at different scales, is crucial for three-dimensional reconstruction and digital twin cities. However, current semantic segmentation methods mostly extract multi-scale features by down-sampling operations, but the feature maps only have a single receptive field at the same scale, resulting in the misclassification of objects with spatial similarity. To effectively capture the geometric features and the semantic information of different receptive fields, a multi-scale voxel-point adaptive fusion network (MVP-Net) is proposed for point cloud semantic segmentation in urban scenes. First, a multi-scale voxel fusion module with gating mechanism is designed to explore the semantic representation ability of different receptive fields. Then, a geometric self-attention module is constructed to deeply fuse fine-grained point features with coarse-grained voxel features. Finally, a pyramid decoder is introduced to aggregate context information at different scales for enhancing feature representation. The proposed MVP-Net was evaluated on three datasets, Toronto3D, WHU-MLS, and SensatUrban, and achieved superior performance in comparison to the state-of-the-art (SOTA) methods. For the public Toronto3D and SensatUrban datasets, our MVP-Net achieved a mIoU of 84.14% and 59.40%, and an overall accuracy of 98.12% and 93.30%, respectively.

1. Introduction

Point cloud semantic segmentation has become the focus of many tasks, i.e., environmental perception (Du et al., 2021), autonomous driving (Guo et al., 2020; Hu et al., 2020; Xu et al., 2021a), and digital twins (Lehtola et al., 2022), due to the rapid development of light detection and ranging (LiDAR) techniques. However, because of the complexity of urban scenes, LiDAR point clouds obtained from such environments contain a large amount of noise and outliers (Liu et al., 2019). Besides, the sparse, unordered, and class-imbalance characteristics of point clouds lead to formidable challenges for accurate and efficient semantic segmentation. Thus, it is necessary to propose a semantic segmentation method that could efficiently handle the task of understanding large-scale urban scenes.

Traditional rule-guided or thresholding-guided feature extraction methods relying on prior knowledge, have limited the feature representation of point clouds (Jing et al., 2021). The development of deep learning has reduced the need for manually designing features, and researchers mainly focus on the optimization of network structures to explore in-depth feature representations of point clouds. Accordingly, early researchers projected three-dimensional (3D) point clouds into two-dimensional (2D) planes (Su et al., 2015) or spheres (Milioto et al., 2019; Lyu et al., 2022; Aksoy et al., 2020; Wu et al., 2018), and then used 2D convolutional neural networks (CNNs) to process point clouds. Moreover, some studies converted point clouds into regular voxels (Riegler et al., 2017; Wang et al., 2018; Zhou et al., 2020) and used 3D

* Corresponding authors.
E-mail addresses: lihuchen@nuist.edu.cn (H. Li), guanhy.nj@nuist.edu.cn (H. Guan), l53ma@cufe.edu.cn (L. Ma), leixd@nuist.edu.cn (X. Lei), allennessy@hyit.edu.cn (Y. Yu), why.sholar@126.com (H. Wang), mdelavar@ut.ac.ir (M.R. Delavar), junli@uwaterloo.ca (J. Li).

https://doi.org/10.1016/j.jag.2023.103391
Received 23 March 2023; Received in revised form 11 June 2023; Accepted 13 June 2023
Available online 19 June 2023

CNNs for normalization. However, all these methods inevitably lost detailed geometric information in projection or voxelization, which affected the semantic segmentation accuracy of point clouds. PointNet, a pioneer in directly applying 3D points for semantic segmentation, maintains point cloud invariance with symmetry functions and spatial transform networks (Qi et al., 2017a). Although point-based methods retain more local geometric details during the feature extraction, the data format of point clouds is unordered, which makes the random memory access time-consuming (Liu et al., 2019). Besides, the sampling grouping module embedded in PointNet++ (Qi et al., 2017b) poorly extracts local features (Hu et al., 2020).

To effectively capture the complex geometric features and the potentially relevant features of large-scale point clouds, some studies explored and used complementary information from multiple views of data. The multi-view data fusion methods, such as point-voxel fusion (Zhang et al., 2020; Liu et al., 2019; Tang et al., 2020), point-projection fusion (Liong et al., 2020), and range-point-voxel fusion (Xu et al., 2021a), enable improve segmentation accuracies compared with single-view methods. Among them, the point-voxel fusion has better complementarity, because they represent point cloud objects in 3D space with different views. Furthermore, the point-based representation is able to acquire better fine-grained spatial geometrical features, since a local feature extraction module can quickly aggregate neighborhood point features without dropping information. Correspondingly, the voxel-based representation has a regular format and ordered arrangement, which maintains the continuity of spatial information and is beneficial for acquiring coarse-grained spatial features. Besides that, using sparse convolution to extract voxel features can not only reduce computation cost but also maintain the sparsity and spatial invariance of point clouds (Tang et al., 2020; Graham et al., 2018). However, these point-voxel fusion methods only fuse single-scale features, ignoring the fact that different-scale features contain different physical dimension properties (Ye et al., 2021b), making it impossible to obtain multi-scale context information. Moreover, the ways of fusing multi-view features are relatively simple (e.g., addition and concatenation), which ineffectively use and represent features with different types. To tackle these problems, a multi-scale voxel-point adaptive fusion network (MVP-Net) is proposed in this paper for semantically segmenting point clouds in urban scenes.

MVP-Net fuses point features and voxel features at different scales via various modules and mechanisms. First, a multi-scale voxel convolution module is constructed to extract geometric features of different receptive fields. Next, to fuse multi-scale receptive field features effectively, a gating mechanism (Xu et al., 2021a) is used to adaptively aggregate voxel information according to the weights of features at different scales. Then, a point-voxel adaptive fusion module with a geometric self-attention (GSA) mechanism (Qin et al., 2022) is designed to effectively extract the potential features of points and voxels. Finally, to synthesize the fused features of each encoded layer, a pyramid decoder (Varney et al., 2022) is used to fuse the raw resolution feature maps with adaptive weighting, and a multi-scale aggregation loss function (Mao et al., 2022) is added to constrain the semantic features in the decoded layers. The main contributions of this paper are summarized as:

- We construct a multi-scale voxel gating fusion (MVGF) module that aggregates voxel features with varying resolutions while adaptively selecting context information.
- We propose a point-voxel adaptive fusion model with GSA, which can enhance model representation by using local geometric and semantic features, and then promote different-grained feature fusion.
- We employ a pyramid decoder to fuse multi-level encoded features and a multi-scale aggregation loss function to increase the supervision of the decoded layers and finally improve segmentation accuracy.

The remainder of this paper is organized as follows. Section 2 presents a systematic survey for point cloud semantic segmentation. Section 3 details our proposed MVP-Net. Section 4 performs relevant experiments on three urban scene datasets to validate and analyze the performance of each module, as well as hyperparameter analysis. Section 5 gives concluding remarks.

2. Related work

2.1. Projection-based segmentation

Projection-based methods mainly project the 3D point clouds into 2D images, including multi-view projection and spherical projection. MVCNN was one of the multi-view projection methods that first used 2D images generated from different views of a point cloud to extract single-view features and then max-pooled all views' features into global features (Su et al., 2015). SqueezeSeg (Wu et al., 2018) was a spherical projection method that used SqueezeNet to extract features from the Spherical-Front-View (SFV) and optimized the segmentation results by the conditional random field (CRF). SalsaNet (Aksoy et al., 2020) compared the contributions of both Spherical-Front-View (SFV) and Bird-Eye-View (BEV) representations in the segmentation process and showed that this method was projection-agnostic. RangeNet++ (Milioto et al., 2019) transformed the semantic results of range images back to 3D point clouds, which avoided convolutional discretization and discarding point clouds. EllipsoidNet (Lyu et al., 2022) projected the point clouds onto an ellipsoid surface, which reduced the overlap inside the points and also generated dense feature maps. The projection of 3D point clouds inevitably loses crucial structure information and the projection view setting has a significant impact on segmentation results. Hence, projection-based methods are suitable for specific small-scale scenes, such as indoor scenes, but have poor accuracy in large-scale urban scenes.

2.2. Voxel-based segmentation

The early voxel-based methods transformed point clouds into uniform voxels and applied dense 3D CNN for semantic segmentation (Wu et al., 2015; Çiçek et al., 2016). However, the computation complexity cubically grows with the increase in voxel resolution (Tang et al., 2020). To alleviate this problem, OctNet used octrees to construct non-uniform voxels for reducing spatial redundancy (Riegler et al., 2017), and MSNet used coarse-grained multi-scale voxels to fuse context information (Wang et al., 2018). Simultaneously, submanifold sparse convolution directly processed the voxel activation region through hash mapping, which greatly improved both efficiency and accuracy (Graham et al., 2018). Accordingly, Cylinder3D (Zhou et al., 2020) used 3D cylinder convolution to balance varying densities of point clouds. (AF)2-S3Net (Cheng et al., 2021) used a multi-branch attentive feature fusion module to suit point clouds with different levels of sparsity. DRINet++ (Ye et al., 2021b) regarded voxels as points and used sparse feature encoders and geometric feature enhancement to extract multi-scale features. Although sparse convolution accelerates network training and inference, lower-resolution voxels destroy the surface structure and lose critical geometric information of point clouds.

2.3. Point-based segmentation

PointNet opened a new chapter of direct processing on points, but it was lack of the local structure for extracting fine-grained features (Qi et al., 2017a). PointNet++, based on PointNet, added sampling and grouping layers to extract local context (Qi et al., 2017b). KPConv used a deformable convolutional processing method with strong descriptive power and learning ability for point clouds in dense regions (Thomas et al., 2019). RandLA-Net, a random sampling network that includes modules for local feature encoding and attention pooling, was suitable for point cloud semantic segmentation in large-scale environments (Hu et al., 2020). MappingConvSeg, inspired by KPConv, proposed a continuous convolution network to learn related spatial features (Yan et al., 2021). More recently, many local feature aggregation methods have been proposed. For instance, SCF-Net introduced a z-axis rotation-invariant description operator (Fan et al., 2021), BAAF-Net was a bilateral structure, where geometric and semantic features were learned from each other (Qiu et al., 2021), and DGFA-Net had a dilated graph feature aggregation structure (Mao et al., 2022). All these methods enriched the neighborhood feature representations. Point-based methods can accurately obtain fine-grained information, but the sparsity, disorder, and irregularity of point clouds are still common problems, which limit their abilities to achieve efficient performance in urban scenes.

2.4. Fusion-based segmentation

Since single-representation methods are more or less problematic (Xu et al., 2021a), some researchers have recently proposed hybrid multiple-representation methods. PVCNN (Liu et al., 2019) established a point-voxel fusion network, in which the point branch extracted fine-grained features and the voxel branch obtained coarse-grained features. On the basis of PVCNN, SPVCNN (Tang et al., 2020) introduced the sparse convolution and neural architecture search to improve the computation efficiency of voxels. FusionNet (Zhang et al., 2020) built a mini-point-network structure in voxels, which can directly aggregate all point features in neighboring voxels to the target voxels. DRINet (Ye et al., 2021a) proposed a dual-representation iterative learning strategy that can flexibly transform representations between point and voxel features. RPVNet (Xu et al., 2021a) stood on the shoulders of SPVCNN and proposed a range-point-voxel fusion network with a gating mechanism to select fused features. The above methods only fuse the single-scale features and fuse them in a single way (e.g., addition). However, our proposed method not only fuses geometric information and semantic features from different receptive fields but also adaptively selects the useful parts.

3. Method

Fig. 1 shows the overall structure of the MVP-Net. The MVP-Net is a dual encoder-decoder network architecture that includes point encoders, voxel encoders, and a pyramid decoder. The original point clouds are first preprocessed to obtain the input points and voxels, respectively. For the point encoder branch, following RandLA-Net (Hu et al., 2020), a Local Point Feature Encoding (LPFE) structure is used to extract fine-grained point features. For the voxel encoder branch, a Multi-scale Voxel Gating Fusion (MVGF) structure is used to explore the coarse-grained features of different receptive fields. Next, point features and voxel features are feature-passed and fused by a Geometric Self-Attention (GSA) module. Then, each level of features in the pyramid decoder is supervised by a multi-scale aggregation loss function. Finally, the multi-scale feature maps are fused and the semantic segmentation results are produced. The following sections describe the detailed structures of the MVP-Net.

3.1. Data pre-processing

To efficiently process the large-scale raw point clouds, we use grid subsampling to reduce point density imbalance while saving on computational costs. Let denote the point cloud in space as $\mathbf{P} = \{p_i \in \mathbf{R}^{d_p}, i = 1, 2, \cdots, N_p\}$ and point features as F_p, where N_p is the number of points, and d_p is the dimension of F_p, i.e., three coordinate values (*x, y, z*), color information, intensity. Given the voxel grid $\mathbf{V} = \{v_i \in \mathbf{R}^{d_v}, i = 1, 2, \cdots, N_v\}$ and voxel features F_v, where N_v is the number of voxels, and d_v is the dimension of F_v, i.e., the average of the point features F_p in per voxel cell. The mapping functions P→V and V→P are used to represent the interconversion of points and voxels.

To reduce the differences in the spatial scales of different point clouds, we transform all the points into a local coordinate system originating at the center of gravity and normalize them to [0,1]. Then, we use the P→V function to obtain the position of the points in the corresponding voxel and set the average of all point features in the voxel cell as the voxel feature, which reduces the feature bias while improving the network training efficiency. After voxelization, the original point features in each voxel are gathered into voxel features, expanding the receptive field to some extent. Based on the regular spatial structure of voxels, we use 3D convolution to extract features. Since the fine-grained features obtained from 3D points can compensate for the geometric information lost by voxels (Ye et al., 2021b), we use low-resolution voxels to extract coarse-grained features to complement the point features while reducing the high memory consumption problem of high-resolution voxels.

3.2. Multi-scale coarse-grained voxel branch

Because of the sparsity and density variations of point clouds, the number of points in the same resolution voxel varies greatly, and the specific physical dimension properties of features in different resolution voxels are also different. Therefore, single-scale voxel features cannot

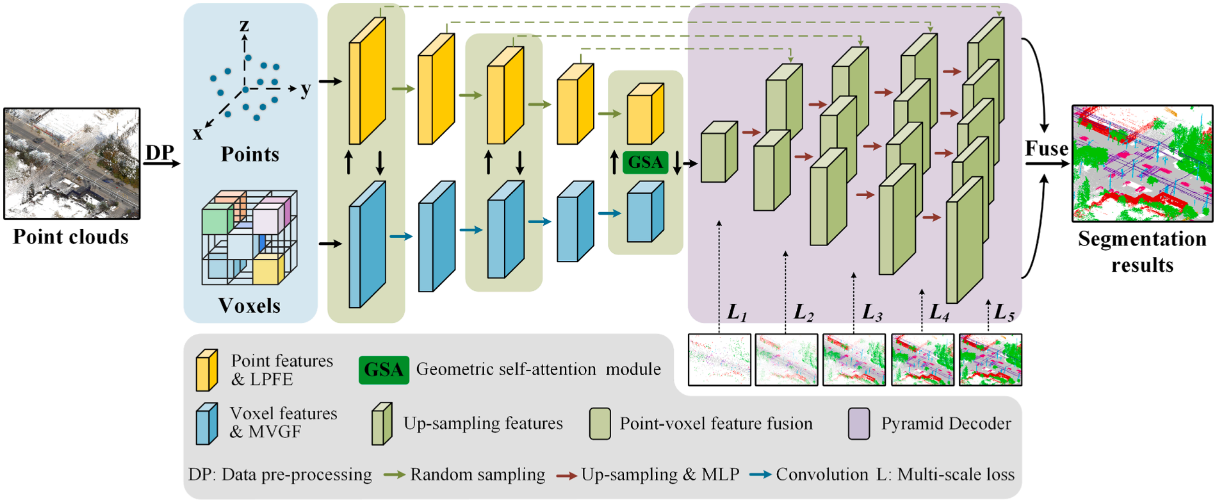


Fig. 1. Overall structure of MVP network.

satisfy accurate urban scene segmentation. Inspired by the multi-scale sparse projection method of DIRNet++ (Ye et al., 2021b), we propose a multi-scale voxel gating fusion structure to extract semantic information from different resolution voxels through different receptive fields.

As shown in Fig. 2 (blue box), in the multi-scale voxel convolution unit, we establish three voxel branches with different resolutions for extracting voxel features at small, medium, and large scales, respectively, which contain the fine details of small objects (e.g., vehicles, pedestrians, and poles) and the global contexts of large objects (e.g., buildings, roads, and vegetation) in urban scenes. Specifically, for each scale, we use two 3D convolutions to expand the receptive field, then refine the voxel resolution by the trilinear interpolation up-sampling (Liu et al., 2019) to the most fine-grained scale voxel.

Commonly used multi-scale feature fusion methods, such as addition and concatenation, ignore the semantic differences of features at different scales, resulting in reduced perception ability of small objects in the network. Therefore, we use a gating mechanism (Xu et al., 2021a) to aggregate multi-scale voxel features. This mechanism can adaptively select context information according to the importance of features and improve the reliability of the model while reducing the redundancy of features. Fig. 2 (green box) shows the detailed structure of the gating fusion mechanism. The gating map $\boldsymbol{G_i}$ is corresponded to the voxel V_i at each scale, which is calculated as follows (Xu et al., 2021a):

$$\mathbf{G_i} = \text{sigmoid}\left(\boldsymbol{W}^{G_i} \cdot V_i\right) \tag{1}$$

where $\mathbf{W}^{\mathbf{G_i}}$ is a learning weights matrix computed by the linear function, $\boldsymbol{G_i} \in [0, 1]$ is used to receive or suppress the pass-through of each scale voxel feature. For r scale voxels, the multi-scale voxel fusion features $\widetilde{F}_v$ can be computed as Eq. (2) (Xu et al., 2021a).

$$\widetilde{F}_v = \sum_{i=1}^{r} \text{split}\left[\text{softmax}\left(\sum_{i=1}^{r} \mathbf{G_i}\right)\right]_i \cdot V_i \tag{2}$$

We sum up the corresponding feature channels of the r gated graphs $\boldsymbol{G_i}$ and calculate the weights by softmax normalization. Then, we multiply the weights of the corresponding channels with the raw features and accumulate them to obtain the fused voxel features $\widetilde{F}_v$. Finally, we use the V→P function to interpolate the fused voxel features to the corresponding points for feature fusion.

3.3. Fine-grained point branch

For fine-grained point branches, we use two units, i.e., Local Spatial Encoding (LocSE) and Attentive Pooling (AP) of RandLA-Net (Hu et al., 2020), to exploit the fine-grained point features in a local space. To improve point feature representation ability, we use the Local Point Feature Encoding (LPFE) module to replace the Relative Point Position Encoding (RPPE) module in LocSE. The LPFE module consists of three parts: geometric encoding features (see Fig. 3a), matrix encoding features (see Fig. 3b), and color encoding features (see Fig. 3c).

The geometric encoding features are aggregated from the central point coordinate p_i, nearest K points coordinate p_i^k, and relative point distance dis_i^k. Specifically, we first calculate the relative point distance as follows (Fan et al., 2021):

$$dis_i^k = \sqrt{\left(x_i^k - x_i\right)^2 + \left(y_i^k - y_i\right)^2 + \left(z_i^k - z_i\right)^2} \tag{3}$$

where (x_i, y_i, z_i) are the coordinates of the central point p_i, and (x_i^k, y_i^k, z_i^k) are the coordinates of the k-th point p_i^k. Then, we extract the geometric encoding features f_i^g of points using Eq. (4) (Hu et al., 2020):

$$f_i^g = \text{mlp}\left(p_i \oplus p_i^k \oplus dis_i^k\right) \tag{4}$$

where ⊕ is the concatenation operation, mlp is Multi-Layer Perceptron.

The matrix encoding features are obtained by Sort Gram Matrix (Xu et al., 2021b), which can reduce the location ambiguity of points, constrain the inherent relationships of neighboring points, and thus obtain invariant features of arbitrary point pairs. The Sort Gram Matrix consists of two parts: the gram matrix and the sort function. The gram matrix (Eq. (5)) learns distance features and angle features of the local coordinates through the inner-product relationship of the point pairs $\mathbf{P_{ir}} \in \mathbf{R}^{3\times K}$ and maintains the point rotation invariance, while the sort function (Eq. (6)) maintains the point permutation invariance by sorting the row vectors, which are calculated as follows (Xu et al., 2021b):

$$\boldsymbol{GM}(\boldsymbol{P_{ir}}) = \boldsymbol{p_{ir}^T p_{ir}} = \begin{bmatrix} p_{ir}^{1T} p_{ir}^1 & p_{ir}^{1T} p_{ir}^2 & \cdots & p_{ir}^{1T} p_{ir}^K \\ p_{ir}^{2T} p_{ir}^1 & p_{ir}^{2T} p_{ir}^2 & \cdots & p_{ir}^{2T} p_{ir}^K \\ \vdots & \vdots & \ddots & \vdots \\ p_{ir}^{KT} p_{ir}^1 & p_{ir}^{KT} p_{ir}^2 & \cdots & p_{ir}^{KT} p_{ir}^K \end{bmatrix} \tag{5}$$

$$\boldsymbol{SGM}(\boldsymbol{P_{ir}}) = \text{f}_{sort}(\boldsymbol{GM}(\boldsymbol{P_{ir}})\,) = \begin{bmatrix} \text{f}_{sort}\left(\boldsymbol{P_{ir}^1}\right) \\ \vdots \\ \text{f}_{\text{sort}}\left(\boldsymbol{P_{ir}^K}\right) \end{bmatrix} \tag{6}$$

where $\mathbf{GM}(\mathbf{P_{ir}})$ is a semi-positive definite matrix that contains the geo-

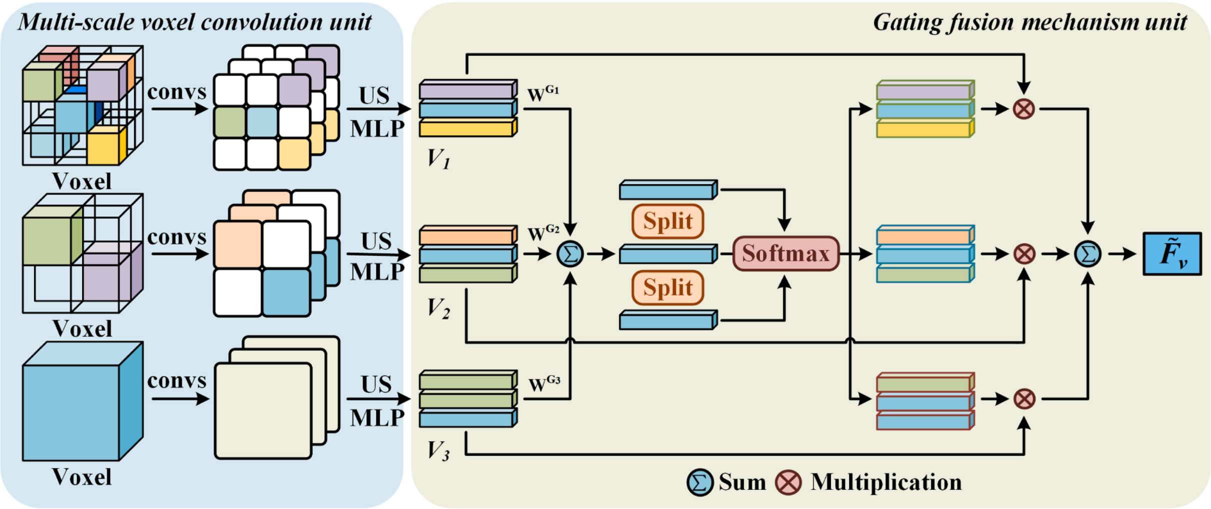


Fig. 2. Multi-scale voxel gating fusion structure. Convs: Convolutions layers. US: Up-sampling. MLP: Multi-Layer Perceptron.

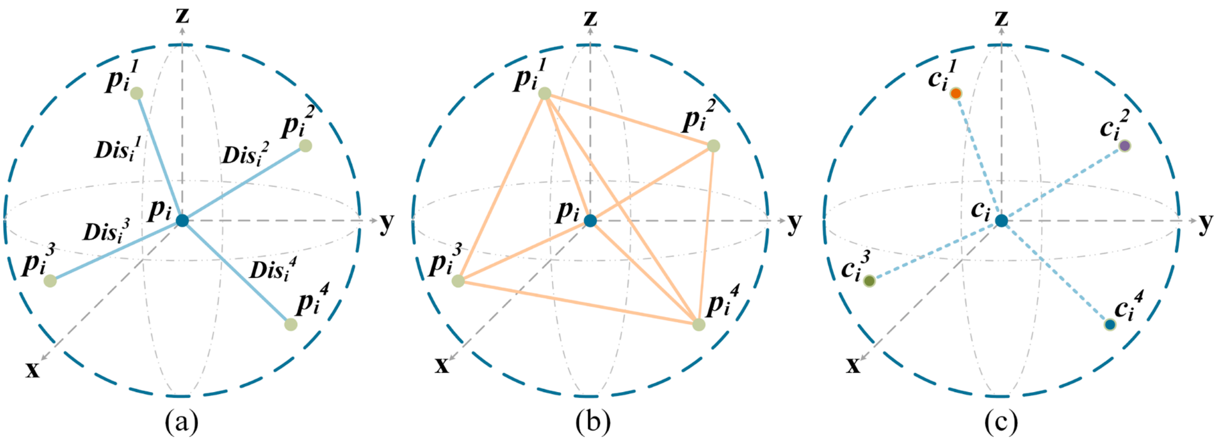


Fig. 3. Local point feature encoding structure, (a) geometric encoding features, (b) matrix encoding features, and (c) color encoding features.

metric properties of all local point pairs and forms a robust matrix structure, $\mathrm{f}_{\mathrm{sort}}(\bullet)$ is the sort-by-row function, $\mathbf{p}_{\mathbf{ir}} = \{p_{ir}^1, p_{ir}^2, \cdots, p_{ir}^K\} = \{p_i^1 - p_i, p_i^2 - p_i, \cdots, p_i^K - p_i\}$ is the relative coordinates, and $\mathbf{P}_{\mathbf{ir}}^{\mathbf{j}} = \{p_{ir}^{jT} p_{ir}^1, p_{ir}^{jT} p_{ir}^2, \cdots, p_{ir}^{jT} p_{ir}^K\} (j = 1, 2, ..., K)$ is the correlation values for each row. Then, we denote $f_i^m = \mathbf{SGM}(\mathbf{P}_{\mathbf{ir}})$ as matrix encoding features.

Since color information and geometric features are complementary (Chen et al., 2022), encoding such color information can further enhance the semantic representation of point clouds. In this paper, the color encoding features are aggregated from the color feature c_i of the central point p_i, the color features c_i^k of the nearest K points, and their variance f_i^σ. Specifically, we first calculate the color feature variance as follows (Chen et al., 2022):

$$f_i^\sigma = \frac{\sum_{k=1}^K \left(c_i^k - c_i\right)^2}{K} \tag{7}$$

The color feature variance f_i^σ helps to distinguish the boundary points, which have different labels of points in their neighborhood. Then, we concatenate c_i, c_i^k, and f_i^σ as the color encoding features f_i^c for each point using Eq. (8).

$$f_i^c = \mathrm{mlp}\left(f_i^\sigma \oplus \left(c_i^k - c_i\right) \oplus c_i\right) \tag{8}$$

Finally, the geometry-encoded features, matrix-encoded features, and color-encoded features of point p_i are concatenated together, denoted as the $R_{ec,i}$ (as shown in Eq. (9)), to encode local point features.

$$R_{ec,i} = \mathrm{mlp}\left(f_i^g \oplus \max\left(f_i^m\right) \oplus f_i^c\right) \tag{9}$$

3.4. Geometric self-attention module

Although combining multi-scale coarse-grained voxel features and fine-grained point features can improve the robustness of the proposed model, direct fusion will produce false predictions of points due to different propagation properties of the different grained features. Geometric Transformer (Qin et al., 2022) adds a geometric structure encoding embedding layers to the Vanilla self-attention mechanism, which significantly captures the internal geometric structure features and maintains the geometric consistency of point clouds. Therefore, a Geometric Self-Attention (GSA) module is designed to augment the point-voxel features and obtain global context information (see Fig. 4).

We use the LPFE structure as the geometric embedding layer in the GSA module, where geometric features and semantic information can constrain the relationship between center points and neighbor points. Firstly, the **Q, K**, and **V** matrices are calculated by the linear function of the point-voxel features $F_{pv} = F_p \oplus \widetilde{F}_v$, and the **R** matrix is calculated by the embedding of the local point encoding features. The details are as follows (Qin et al., 2022):

$$\begin{aligned} (\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) &= F_{pv} \cdot \left(\boldsymbol{W}^{\boldsymbol{Q}}, \boldsymbol{W}^{\boldsymbol{K}}, \boldsymbol{W}^{\boldsymbol{V}}\right) \\ \boldsymbol{R} &= R_{ec} \cdot \boldsymbol{W}^{\boldsymbol{R}} \end{aligned} \tag{10}$$

where **Q**, **K**, and **V** denote the query, key, and value matrices, respectively, and **R** denotes the local point encoding matrix. $\mathbf{W}^{\mathbf{Q}}$, $\mathbf{W}^{\mathbf{K}}$, $\mathbf{W}^{\mathbf{V}}$, and $\mathbf{W}^{\mathbf{R}}$ are the weights of the corresponding feature matrices, respectively.

Next, the cross-attention score **E** is calculated by multiplying the **Q**

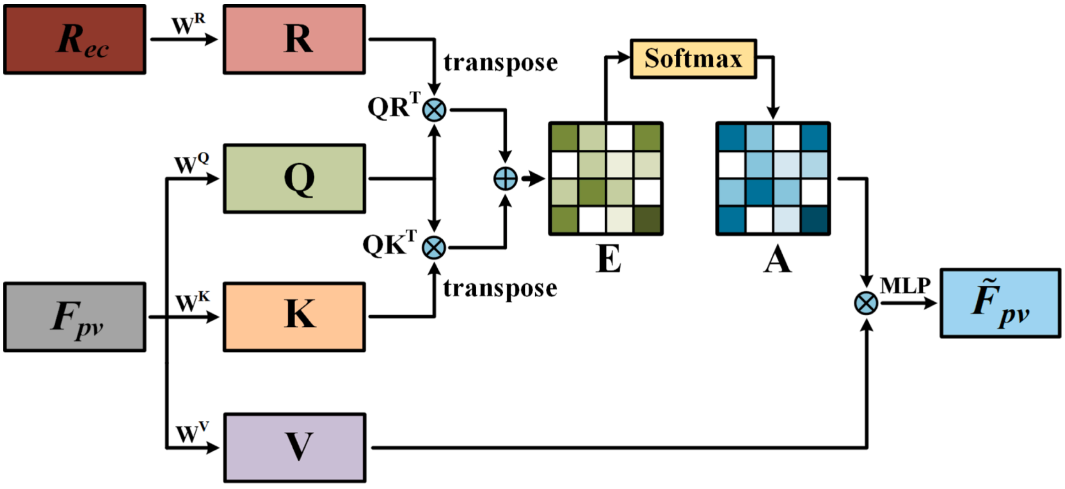


Fig. 4. Geometric self-attention module.

matrix with the **K** and **R** matrices, respectively. Ultimately, the attention weights are calculated using softmax and multiply with the **V** matrix to obtain the point-voxel fusion features $\widetilde{F}_{pv}$ as follows (Qin et al., 2022):

$$\boldsymbol{A} = \mathrm{softmax}\left(\frac{\boldsymbol{Q}\cdot\boldsymbol{R}^T + \boldsymbol{Q}\cdot\boldsymbol{K}^T}{\sqrt{dt}}\right) \tag{11}$$

$$\widetilde{F}_{pv} = \mathrm{mlp}(\boldsymbol{A}\cdot\boldsymbol{V}) \tag{12}$$

where dt is the feature dimension of the key vectors and **A** denotes the attention weight factor.

3.5. Pyramid decoder module

To fully utilize the point-voxel fusion features of different encoding layers, we introduce a pyramid decoder module (Varney et al., 2022) that allows different receptive field features to propagate between adjacent decoding layers, further promoting the reliability of the network. The pyramid decoder module is driven by the multi-scale features of fusion and multi-scale aggregation loss (see Eqs. (14) and (15)).

3.5.1. Multi-scale features of fusion

Given that the encoder and decoder feature maps at different layers are denoted as $\{S_1, S_2, \cdots, S_M\}$, which include $\{N_1, N_2, \cdots, N_M\}$ points, respectively, where M indicates the number of network layers. After continuous up-sampling operations, the $\{S_1, S_2, \cdots, S_M\}$ layers of feature maps in the pyramid decoder are recovered to the raw resolution feature maps $\{\widehat{S}_1, \widehat{S}_2, \cdots, \widehat{S}_M\}$, respectively. The lower feature maps (shallow layers) focus on the local details and the higher feature maps (deep layers) focus more on the global boundary information. In order to better aggregate the boundary and internal region features and reduce the perception bias between feature maps, we adaptively fuse the raw resolution feature maps. To be concrete, we learn the weights and multiply them with the feature maps, and then accumulate the weighted feature maps to obtain a fused feature map. Theoretically, it follows:

$$\lambda_m = \mathrm{softmax}(\mathrm{mlp}(\widehat{\mathbf{S}}_{\mathbf{m}})) \tag{13}$$

where $m \in (1, 2, \cdots, M)$, $\widehat{\boldsymbol{S}}_{\boldsymbol{m}}$ is the raw resolution feature map of the m-th layer, and λ_m is the weight corresponding to $\widehat{\boldsymbol{S}}_{\boldsymbol{m}}$ feature map. The adaptive weighted fusion feature map S_f is formulated as follows:

$$\mathbf{S_f} = \sum_{m=1}^{M} \lambda_m \cdot \widehat{\mathbf{S}}_{\mathbf{m}} \tag{14}$$

3.6. 2. Multi-scale aggregation loss

The adaptive weighted fusion of raw resolution feature maps improves the segmentation accuracy by aggregating features from different layers, but the semantic features lack label constraints during the up-sampling process. Therefore, we add a multi-scale aggregation loss (Mao et al., 2022) for supervising the features in the decoding layer to reduce the perturbation and further constrain the raw resolution features. The multi-scale aggregation loss function L_{ma} is calculated as follows:

$$L_{ma} = -\sum_{m=1}^{M} \varphi_m \cdot \sum_{i=1}^{N_m} \sum_{c=1}^{C} y_m^{ic} \cdot \log\left(p_m^{ic}\right) \tag{15}$$

where φ_m is the weight of the decoder features in the m layer, C is the number of classes, y_m^{ic} and p_m^{ic} denote the true label and the predicted label of point p_i as class c in the m layer, respectively. We set the weights as $\varphi_m = \{0.1, 0.1, 0.3, 0.5, 0.5\}$. In this paper, the loss function L_{total} includes the multi-scale aggregation loss function L_{ma} and the weighted cross-entropy loss function L_{wce}. L_{wce} is detailed as follows (Han et al., 2021):

$$w_{sqrt} = \frac{1}{\sqrt{N_c}\sum_i^C \frac{1}{\sqrt{N_i}}} \tag{16}$$

$$L_{wce} = -\sum_{i=1}^{N} w_{sqrt,i} \cdot \sum_{c=1}^{C} y^{ic} \cdot \log\left(p^{ic}\right) \tag{17}$$

where w_{sqrt} is the class weight of L_{wce}, N_c is the point number of the class c, y^{ic} and p^{ic} are the ground truth label and the predicted label of point p_i as class c, respectively. Hence, L_{total} is formulated as follows:

$$L_{total} = \alpha L_{wce} + \beta L_{ma} \tag{18}$$

where α and β are the constants used to balance the two loss functions. In this paper, we set $\alpha = 0.5$ and $\beta = 0.5$ (as detailed in Section 4.6).

4. Experiments and analysis

The performance of MVP-Net is evaluated on three point cloud datasets of urban scenes, namely Toronto3D, WHU-MLS, and SensatUrban, followed by the ablation analysis of each module and the hyperparametric analysis.

4.1. Experimental setup

4.1.1. Datasets

To fully evaluate the performance of MVP-Net in point cloud semantic segmentation, we used three urban scene datasets, i.e., Toronto3D, WHU-MLS, and SensatUrban, which were obtained from different platforms in different cities and contained different classes.

Toronto3D was captured by a Mobile Laser Scanning (MLS) system on Avenue Road in Toronto, Canada, covering a stretch of approximately 1.0 km with approximately 78.3 million points (Tan et al., 2020). This large-scale point cloud dataset was equally divided into 4 parts (named L001, L002, L003, and L004). Following Tan et al. (2020), the L002 was utilized for testing, while the other three parts were employed for training and validation. The raw point cloud was categorized into 8 object classes, and each point contains ten attributes, i.e., (X, Y, Z) coordinates, (R, G, B) color information, intensity, GPS Time, scan angle, and label. In this dataset, the inputs used for network training and testing are only color information and 3D coordinates. To precisely analyze the effectiveness of the proposed MVP-Net, we used Overall Accuracy (OA), per-class Intersection-over-Unions (IoUs), and mean Intersection-over-Union (mIoU) as evaluation metrics.

WHU-MLS is an MLS point cloud dataset jointly released by the Wuhan University and Shanghai Surveying and Mapping Institute (Yang et al., 2021). This dataset was divided into 40 scenes, 30 of them were used for training and 10 for testing. WHU-MLS includes more than 30 kinds of objects and 5000 typical instances in urban scenes, including ground, dynamic targets, vegetation, poles and their appurtenant structures, buildings and structural facilities, and other public amenities (Yang et al., 2021), totaling more than 300 million points. For a fair comparison, the point cloud was segmented into 17 object classes in accordance with Han et al. (2021). The inputs to our proposed network included 3D coordinates, intensity, and normals. Following Yang et al. (2021), we used IoUs and mIoU as evaluation metrics.

SensatUrban is a photogrammetric point cloud dataset covering three cities in the UK, including Birmingham, Cambridge, and York, with a total area of 7.6 km^2. Following Hu et al. (2022), 10 of the 14 tiles from Birmingham were used for training, 2 for validation, and 2 for testing, and 20 of the 29 tiles from Cambridge were used for training, 5 for validation, and 4 for testing. The raw point cloud was categorized into 13 object classes. The input to our proposed network included 3D coordinates and color information. We used OA, IoUs, and mIoU as evaluation metrics.

4.1.2. Training and inference details

For Toronto3D, WHU-MLS, and SensatUrban, the subsampling grid sizes were set to 0.04 m (Tan et al., 2020), 0.08 m (lei et al., 2022), and 0.02 m (Hu et al. 2022), respectively. In the proposed MVP-Net, the number of input points to the network was 65,536 per batch, and the training and testing batch sizes were 4 and 8, respectively. The whole network was a U-Net-like structure with 5-layer encoders and 5-layer decoders, where the encoder was divided into two branches: point branch and voxel branch. The feature dimensions of each layer in the network were {16, 64, 128, 256, 512}, respectively. For the point encoding branch, random down-sampling was performed to decrease the number of points, and the sampling rates of each layer were {1/4, 1/16, 1/64, 1/256, 1/512} of the input points, respectively. For the voxel encoding branch, the resolutions of voxels were set to {0.25 m, 0.5 m, 1 m} (detailed in Section 4.6), and each layer included two 3 × 3 × 3 convolutions. In the decoder, the nearest neighbor up-sampling was used to recover the number of input points. During the training process, the number of epochs was set to 100, the iteration steps for each epoch were set to 500, the model was updated using the Adam optimizer, the momentum was set to 0.95, the reduction after each epoch iteration was set to 5%, the number of nearest neighbor points K was set to 16, and the last training result was used for evaluation. All the experiments were performed on an Intel(R) Xeon(R) Silver 4210R CPU@2.40 GHz and a single NVIDIA GeForce RTX3090 GPU.

4.2. Evaluation on Toronto3D

Table 1 shows the quantitative results of the MVP-Net and other SOTA networks on the L002 section of the Toronto3D dataset. To ensure a fair comparison and verify the semantic feature extraction ability of the model, we divided the experiments into two groups: one group experiment used RGB information and the other did not. Experimental results showed that when not using RGB information, MVP-Net achieved a 3.15% improvement over the RandLA-Net in terms of the OA (96.10%), but a 2.54% reduction over the RandLA-Net in terms of the mIoU (75.17%), and MVP-Net achieved higher performance on roads, utility lines, and poles. Due to the stronger spatial geometric similarity of roads and road markings and the lack of color encoding information, it is difficult for MVP-Net to distinguish between roads and road markings, therefore, the IoU of MVP-Net on road markings (22.02%) was lower than that of RandLA-Net (42.62%). When using RGB information as extra inputs, MVP-Net was superior over other networks in terms of the OA (98.12%) and mIoU (84.14%), improving over the RandLA-Net by 3.75% and 2.37%, respectively. Overall, our proposed MVP-Net achieved excellent performance in five out of eight classes. They are road, road marking, natural, building, and car.

To intuitively compare the segmentation qualities of the comparative methods, we showed the visualization results obtained by MVP-Net and RandLA-Net using RGB color information as the network input features (see Fig. 5). Moreover, we presented four of the close-view regions (see Fig. 6), where the differences between MVP-Net and RandLA-Net were shown in the red boxes. From the visualization results, the segmentation effects of MPV-Net on roads, natural, poles, and cars were better than those of RandLA-Net, especially on pedestrian crossings and lane arrows (see Fig. 6c). These two classes were clearer and more complete, mainly because the LPFE module (Eq. (9)) deeply explored the geometric features and color information in the local space, which facilitated the model to distinguish scene objects with similar spatial structures. The lower resolution voxel units in MVP-Net obscured the semantic features of the fences and the surrounding buildings, resulting in the misclassification of fences in some regions, but the results (see Fig. 6d) showed that MVP-Net maintained the integrity of the overall structure of the building, which was mainly attributed to the fact that the MVGF module obtained coarse-grained features from different receptive fields and selectively aggregated the semantic features through the gating mechanism (see Fig. 2).

In addition, we compared the effects of using or not using color information on the segmentation results of road markings (see Fig. 7). From the visualization results, we can observe that the road markings without color information have scattered structures and ambiguous segmentation boundaries, which cannot be distinguished from the roads. The use of encoded color information can enhance the local spectral differences and thus facilitate the model to effectively distinguish roads and road markings.

4.3. Evaluation on WHU-MLS

The quantitative results of MVP-Net and other comparative networks on the WHU-MLS dataset are presented in Table 2. Our MVP-Net improved by 5.94% in mIoU (67.36%) over the baseline's mIoU (61.42%), which was tested under the same experimental settings. MVP-Net achieved the best performance in 14 out of 17 classes in the WHU-MLS dataset, indicating an across-the-board improvement of instances in the urban scenes. Fig. 8 shows the four scenes in the WHU-MLS test set and their close-view areas, with the red boxes indicating the model performance differences between MVP-Net and RandLA-Net. The municipal poles, telegraph poles, traffic lights, and detectors all belong to poles and their appurtenant structures, which have similar geometric structures and spatial locations (see Fig. 8a). The baseline method only used the local geometric features, which can easily cause confusion in

Table 1
Qualitative results for various methods on the L002 section of the Toronto3D dataset. The result scores of other SOTA networks came from Du et al. (2021) and Zeng et al. (2022). The scores in underlines indicate the best results for each category with or without the RGB color features, respectively, while the scores in bold are the best in all methods.

RGB	Method	OA (%)	mIoU (%)	IoUs (%)							
				road	road m.	natural	build.	util. l.	pole	car	fence
No	PointNet++	84.88	41.81	89.27	0.00	69.06	54.16	43.78	23.30	52.00	2.95
	PointNet++ (MSG)	92.56	59.47	92.90	0.00	86.13	82.15	60.96	62.81	76.41	14.43
	DGCNN	94.24	61.79	93.88	0.00	91.25	80.39	62.40	62.32	88.26	15.81
	KFCNN	95.39	69.11	94.62	0.06	96.07	91.51	87.68	81.56	85.66	15.72
	MS-PCNN	90.03	65.89	93.84	3.83	93.46	82.59	67.80	71.95	91.12	22.50
	TGNet	94.08	61.34	93.54	0.0	90.83	81.57	65.26	62.98	88.73	7.85
	MS-TGNet	95.71	70.50	94.41	17.19	95.72	88.83	76.01	73.97	94.24	23.64
	RandLA-Net	92.95	77.71	94.61	42.62	96.89	93.01	86.51	78.07	92.85	37.12
	MVP-Net (Ours)	96.10	75.17	95.15	22.02	96.55	92.80	**88.37**	**85.00**	91.89	29.58
Yes	RandLA-Net	94.37	81.77	96.69	64.21	96.92	94.24	88.06	77.84	93.37	42.86
	ResDLPS-Net	96.49	80.27	95.82	59.80	96.10	90.96	86.82	79.95	89.41	43.31
	BAAF-Net	94.20	81.20	96.80	67.30	96.80	92.20	86.80	82.30	93.10	34.00
	BAF-LAC	95.20	82.20	96.60	64.70	96.40	92.80	86.10	83.90	93.70	**43.50**
	LACV-Net	97.40	82.70	97.10	66.90	97.30	93.00	87.30	83.40	93.40	43.10
	MVP-Net (Ours)	**98.12**	**84.14**	**98.00**	**76.36**	**97.34**	**94.77**	87.69	84.61	**94.63**	39.74

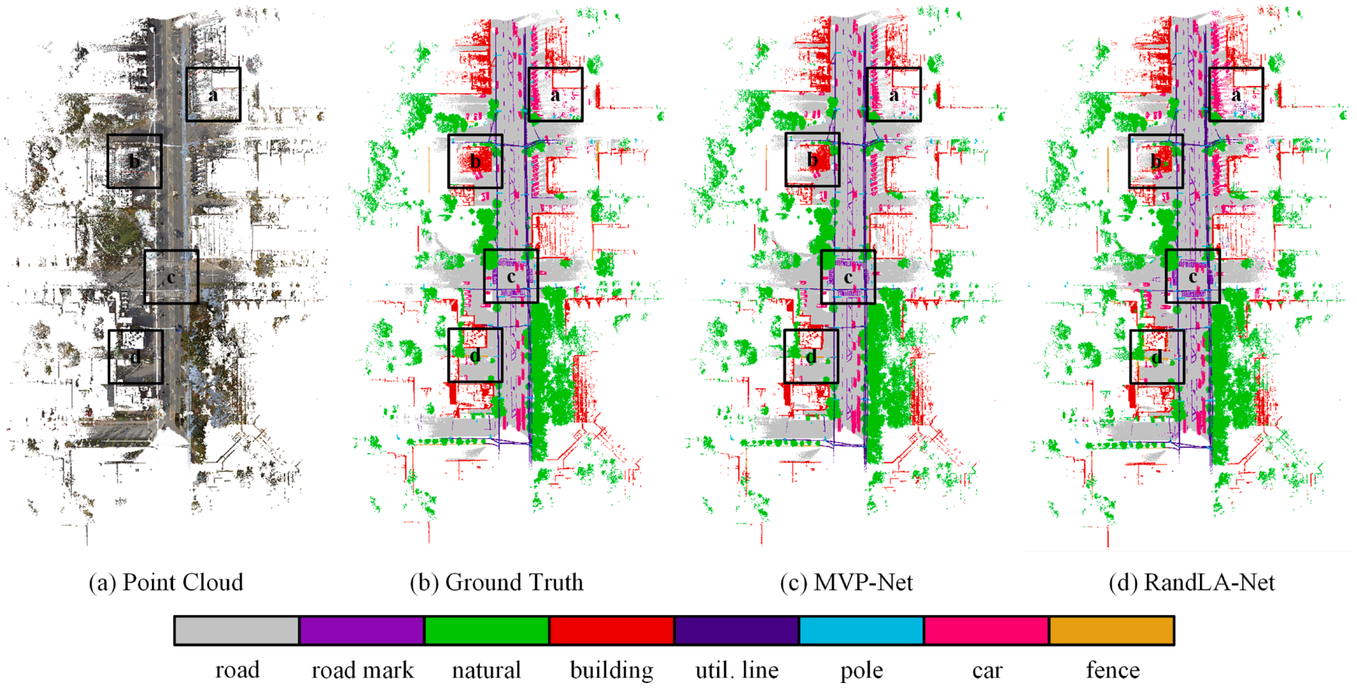

(a) Point Cloud (b) Ground Truth (c) MVP-Net (d) RandLA-Net

Fig. 5. Comparison of visualization results on the Toronto3D dataset, (a) raw point cloud with RGB, (b) ground truth labels, (c) semantic segmentation results obtained by our method, (d) semantic segmentation results obtained by RandLA-Net.

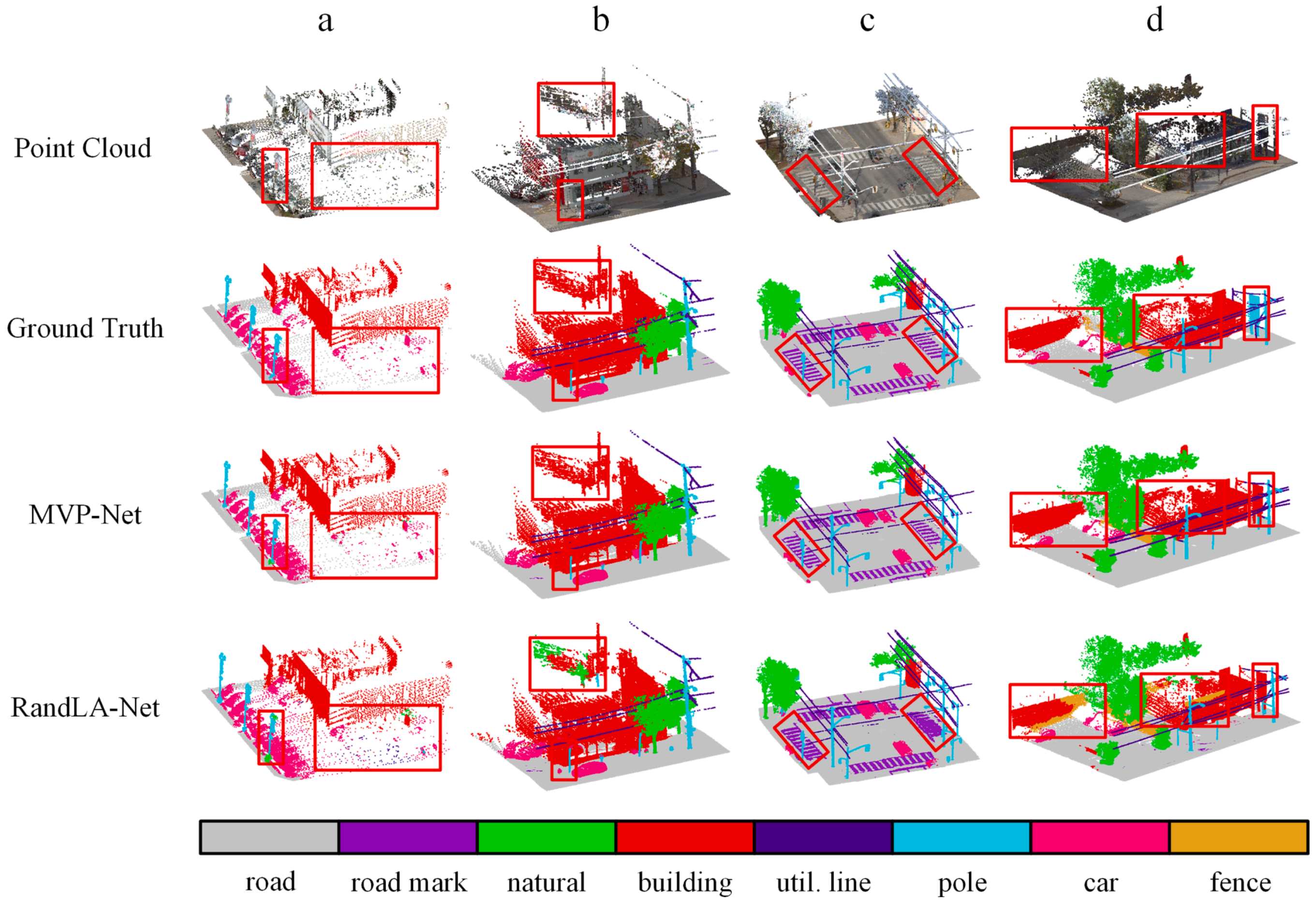


Fig. 6. Comparison of visualization results on the Toronto3D dataset, where a to d represent detailed views of the four semantic segmentation areas.

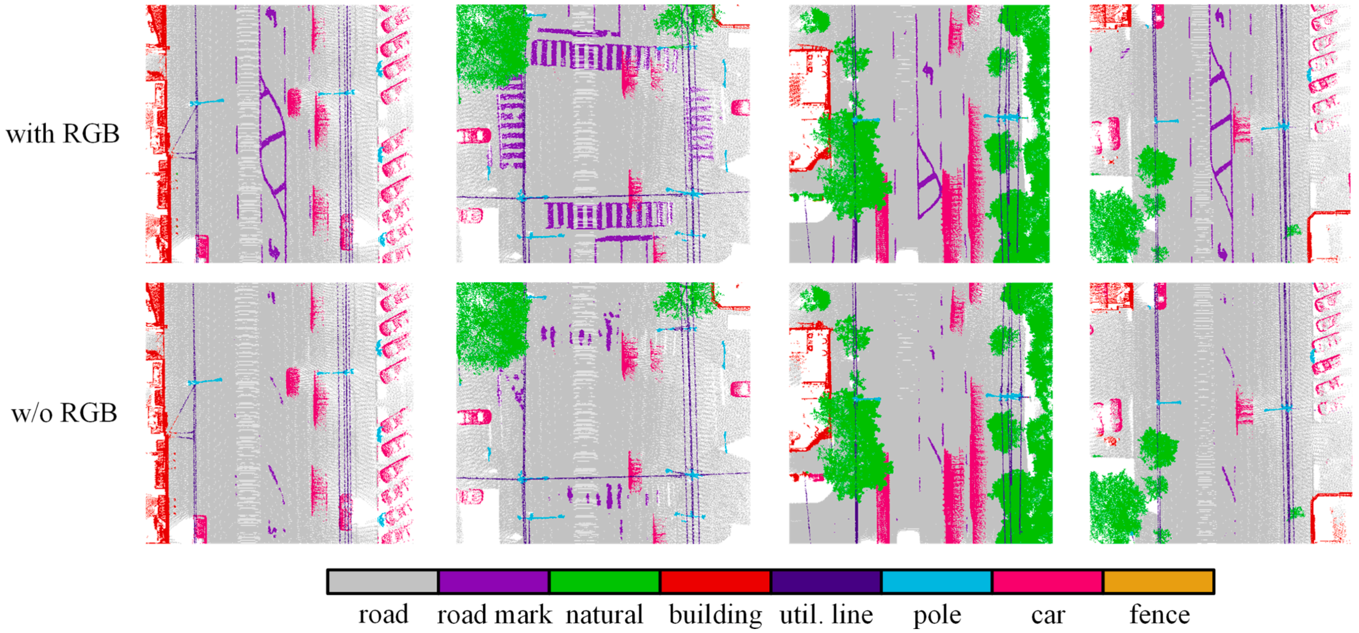


Fig. 7. Comparison of visualization results with or without RGB information as initial input on the Toronto3D dataset.

Table 2
Quantitative results of the comparative methods on the WHU-MLS dataset. The scores of other comparative methods came from Han et al. (2021). Baseline scores were acquired from RandLA-Net trained with the parameters mentioned in Section 4.1.2. Bolded scores indicate the best results in all methods, while the scores in underline are second only to the best.

Methods	mIoU (%)	IoUs (%)								
		tree roadway	nd. way rd. mark.	building vehicle	box pedestrian	light trff. light	tel. pole detector	mun. pole fence	low veg. wire	board
PointNet++	41.10	83.30	42.00	72.70	6.60	59.10	30.80	7.80	33.10	13.90
		80.00	29.50	76.70	38.90	25.00	11.00	56.30	32.70	
PointConv	46.40	85.60	48.90	73.50	28.20	59.70	35.70	20.00	32.40	16.00
		82.00	30.60	76.20	53.80	28.70	27.60	52.60	36.50	
Han's method	52.80	84.50	**58.40**	77.10	45.40	71.80	49.90	26.50	34.10	20.20
		83.60	38.10	79.10	60.80	31.00	31.30	57.90	47.20	
Baseline	61.42	89.92	52.97	84.17	54.21	70.42	45.89	25.52	32.01	36.95
		91.84	54.24	94.79	**83.40**	49.85	40.90	**69.84**	67.27	
MVP-Net (Ours)	**67.36**	**91.35**	54.25	**89.69**	**63.42**	**86.03**	**73.15**	**34.52**	**37.68**	**40.16**
		92.95	**61.43**	**95.19**	82.09	**53.69**	**44.90**	66.02	**78.66**	

the pole class (see Fig. 8c). However, the multi-scale voxel unit in MVP-Net can perceive the pole features and their appendages as a whole, which maintained the integrity of the semantic segmentation classes and also increased the discrimination between different features (see Fig. 8b). In addition, for dynamic targets (e.g., pedestrians and vehicles) with relatively independent spatial structures, MVP-Net rapidly aggregated fine-grained point features and coarse-grained voxel features, improving the segmentation accuracy of these classes.

4.4. Evaluation on SensatUrban

The quantitative results of MVP-Net and other comparative networks on the SensatUrban dataset are presented in Table 3. Our MVP-Net was superior over other networks in terms of the OA (93.30%) and mIoU (59.40%), improving over RandLA-Net by 3.52% and 6.71%, respectively. Furthermore, MVP-Net achieved the best performance in 5 out of 13 classes, including building, bridge, parking, traffic road, and footpath. Fig. 9 shows the results of the online evaluation on the test set of the SensatUrban dataset, with the red boxes indicating the differences between MVP-Net and RandLA-Net. From the visualization results, MVP-Net not only maintains the integrity of buildings and parking, but also better separates cars and street furniture in comparison to RandLA-Net. For tiny bridges and water, MVP-Net can also be extracted. The quantitative and visualization results show that MVP-Net can effectively improve the semantic segmentation accuracy of the point clouds in urban scenes.

4.5. Ablation studies

The experimental results on the Toronto3D, WHU-MLS, and SensatUrban datasets demonstrated the superior performance of MPV-Net. As the number of times the SensatUrban dataset could be validated online was limited, we further conducted ablation studies on the other two datasets to evaluate the effectiveness of our designed modules.

4.5.1. Effect of MVGF

The MVGF module is composed of a multi-scale voxel convolution unit and a gating fusion (GF) unit. We added the MVGF module to RandLA-Net (i.e., the baseline), and named the resultant network as Model A1. As illustrated in Table 4, in comparison to the baseline, Model A1 gained an improvement of 1.01% OA and 1.97% mIoU on the Toronto3D dataset, respectively, and obtained an increase of 0.76% OA and 2.98% mIoU on the WHU-MLS dataset, respectively. Through the experiment results, we concluded that the multi-scale voxel structure

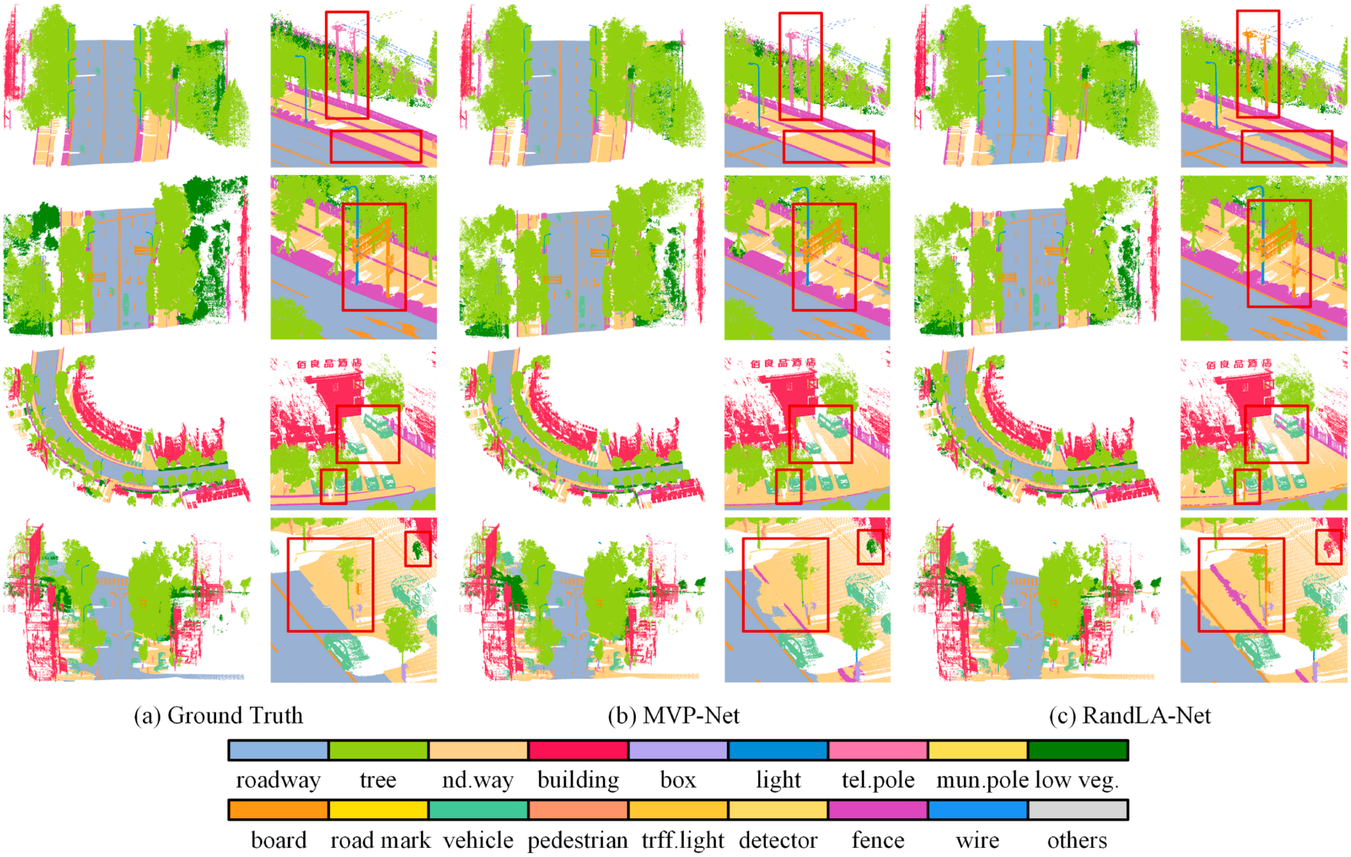


Fig. 8. Comparison of visualization results on the WHU-MLS dataset, (a) Ground truths, (b) semantic segmentation results obtained by our method, (c) semantic segmentation results obtained by RandLA-Net.

Table 3
Quantitative results of comparative methods on the SensatUrban dataset. The scores of other comparative methods came from Hu et al. (2022). Bolded scores indicate the best results in all methods, while the scores in underline are second only to the best.

Methods	OA (%)	mIoU(%)	IoUs (%)												
			ground	veg.	build.	wall	bridge	park.	rail	traffic.	street.	car	foot.	bike	water
PointNet	80.78	23.71	67.96	89.52	80.05	0.00	0.00	3.95	0.00	31.55	0.00	35.14	0.00	0.00	0.00
PointNet++	84.30	32.92	72.46	94.24	84.77	2.72	2.09	25.79	0.00	31.54	11.42	38.84	7.12	0.00	56.93
TagentConv	76.97	33.30	71.54	91.38	75.90	35.22	0.00	45.34	0.00	26.69	19.24	67.58	0.01	0.00	0.00
SPGraph	85.27	37.29	69.93	94.55	88.87	32.83	12.58	15.77	**15.48**	30.63	22.96	56.42	0.54	0.00	44.24
SparseConv	88.66	42.66	74.10	97.90	94.20	63.30	7.50	24.20	0.00	30.10	34.00	74.40	0.00	0.00	54.80
KPConv	93.20	57.58	**87.10**	**98.91**	95.33	**74.40**	28.69	41.38	0.00	55.99	**54.43**	**85.67**	40.39	0.00	**86.30**
RandLA-Net	89.78	52.69	80.11	98.07	91.58	48.88	40.75	51.62	0.00	56.67	33.23	80.14	32.63	0.00	71.31
MVP-Net (Ours)	**93.30**	**59.40**	85.10	98.50	**95.90**	66.60	**57.50**	**52.70**	0.00	**61.90**	49.70	81.80	**43.90**	0.00	78.20

can learn features from different granularity spaces and enhance the generalization capability of the network.

Additionally, we qualitatively analyzed the differences between the addition fusion strategy and the gating fusion strategy. As shown in Fig. 10, the red boxes show the differences between the two fusion strategies. From the feature maps, both feature maps are smooth and clear for large objects (e.g., roads and buildings). But for small objects, e.g., cars, poles, lights, and benches, the feature maps of the addition fusion strategy have ambiguous boundaries, while the feature maps of the gating fusion strategy have clearer boundaries and are more distinguishable from the surrounding features. Because of the different receptive fields of voxel features at different scales, the addition fusion strategy causes feature semantic confusion. The gating fusion strategy can reduce useless information, adaptively aggregate features at different scales, and maintain the integrity of spatial structure features.

4.5.2. *Effect of GSA*

The GSA module was then added to Model A1 and named the resultant network as Model A2. As can be seen in Table 4, in comparison to Model A1, on the Toronto3D dataset, Model A2 improved the OA and mIoU by 0.54% and 1.28%, respectively, while 0.26% OA and 1.14% improvement on the WHU-MLS dataset, respectively. The accuracy improvement demonstrated that the GSA module can effectively aggregate point-voxel features, and the local point encoding features contribute to a consistent geometric structure during the feature fusion process.

Further, we validated the performance of the LPFE module on the Toronto3D dataset, by providing the network only with coordinate information as the first input features (see Table 5). We conducted the following ablation experiments: (1) encoding coordinate information only, and named this network as Model B0; (2) encoding color information only, and named this network as Model B1; (3) encoding

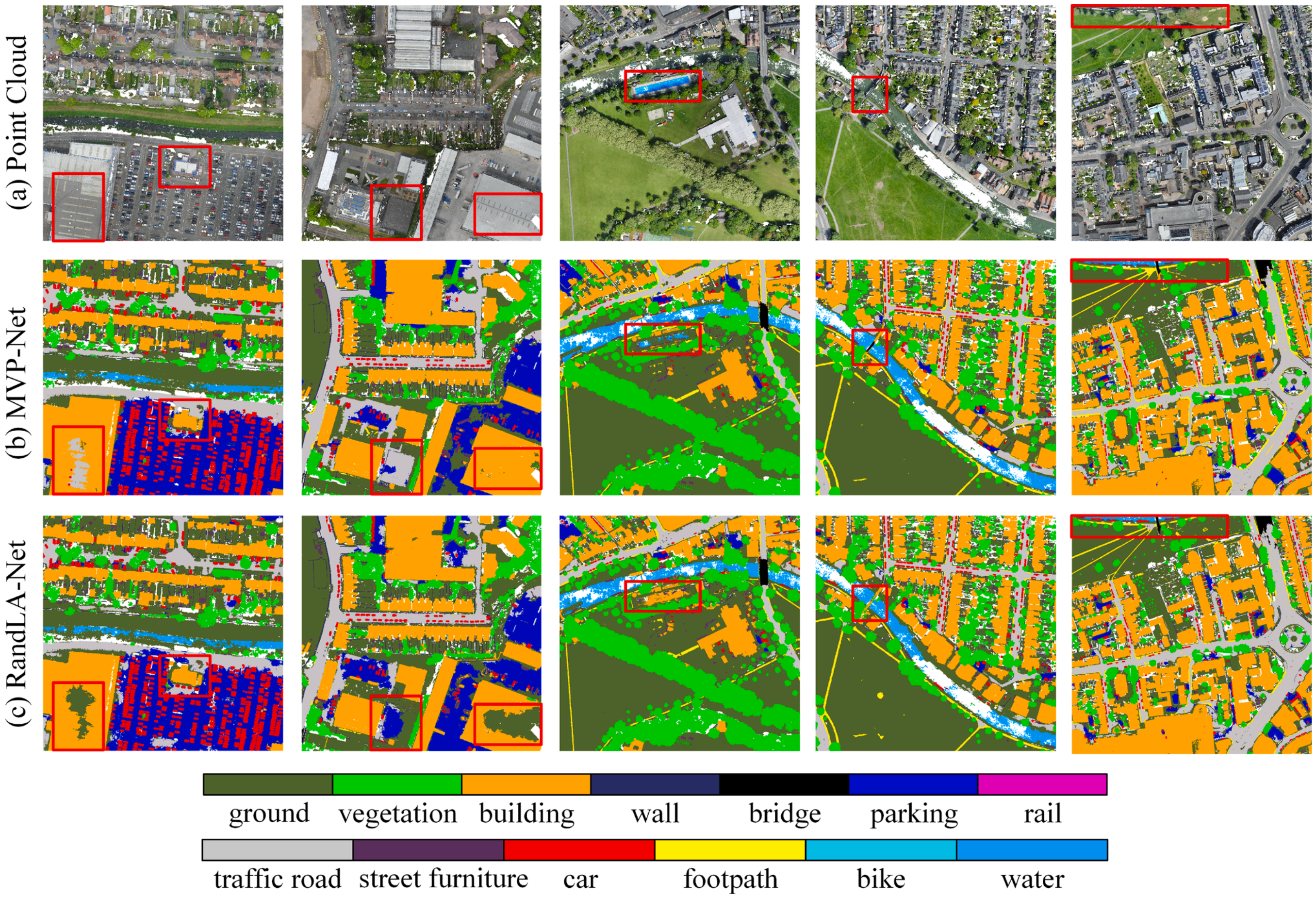


Fig. 9. Comparison of visualization results on the SensatUrban dataset, (a) raw point cloud with RGB, (b) semantic segmentation results obtained by our method, (c) semantic segmentation results obtained by RandLA-Net.

Table 4
Comparison of the experimental results of different models on the Toronto3D and WHU-MLS datasets. The bolded scores are the best in all models.

Model	MVGF	GSA	PD	MALoss	Toronto3D		WHU-MLS	
					OA (%)	mIoU (%)	OA (%)	mIoU (%)
Baseline					95.91	77.88	89.35	61.42
A1	√				96.92	79.85	90.11	64.40
A2	√	√			97.46	81.13	90.37	65.54
A3	√	√	√		97.89	83.17	91.26	66.92
MPV-Net	√	√	√	√	**98.12**	**84.14**	**91.32**	**67.36**

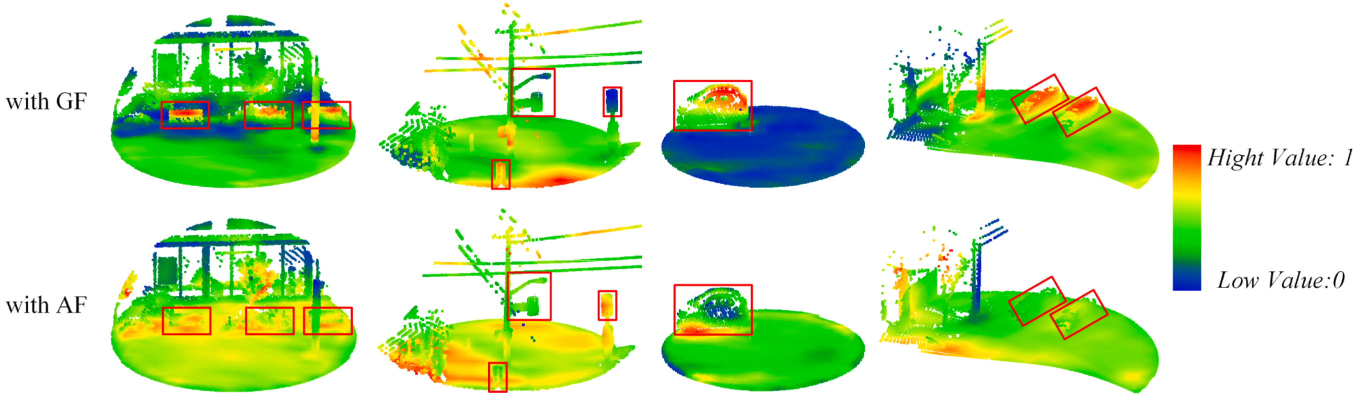


Fig. 10. Comparison of visualization results of feature maps on the Toronto3D dataset. GF: Gating Fusion. AF: Addition Fusion.

Table 5
Comparison of different encoding features on the Toronto3D dataset, only the coordinate information is used as the original feature of the network. The scores in bold are the best in all the models.

Model	Encoding	Toronto3D	
		OA (%)	mIoU (%)
B0	xyz	95.60	72.45
B1	rgb	92.55	64.99
B2	xyz + rgb	96.30	78.73
B3	xyz + matrix	95.97	73.62
B4	rgb + matrix	97.21	80.17
B5	xyz + rgb + matrix	**97.25**	**80.64**

coordinate information and color information, and named this network as Model B2; (4) encoding coordinate information and matrix information, and named this network as Model B3; (5) encoding color information and matrix information, and named this network as Model B4; and (6) encoding all coordinate, color, and matrix information, and named this network as Model B5. We could draw the following three conclusions from Table 5. (1) Spatial coordinate features have the greatest influence on the model performance, and geometric features in the neighborhood space are beneficial for capturing context information. (2) Color information and geometric features are complementary, and color variance helps to improve the segmentation accuracy of boundary points. (3) Encoding geometric features between any point pairs in local space can improve model performance even further.

4.5.3. Effect of pyramid decoder

We added a pyramid decoder (PD) to Model A2 and named the resultant network as Model A3. Compared to Model A2, Model A3 gained an improvement of 0.43% and 2.04% on OA and mIoU, respectively, on the Toronto3D dataset, as well as an increase of 0.89% and 1.38% on OA and mIoU, respectively, on the WHU-MLS dataset (see Table 4). The statistical results indicated that the pyramid decoder can effectively enhance the robustness of the network by fusing multi-layer feature maps. Then, we visualized the fused feature maps of using or not using the adaptive weighted fusion module (as detailed in Eq. (14)). It can be seen that the adaptive weighted fusion of multi-layer feature maps is beneficial to reduce the conflict of different level features and improve the effectiveness of the pyramid decoder (see Fig. 11).

4.5.4. Effect of multi-scale loss function

We incorporated the weighted cross-entropy loss function with the multi-scale aggregation loss (MALoss) function. As shown in Table 4, MVP-Net outperformed Model A3 with an OA of 0.23% and a mIoU of 0.97%, respectively, on the Toronto3D dataset, as well as with an OA of 0.06% and a mIoU of 0.44%, respectively, on the WHU-MLS dataset. The experimental results demonstrated that the multi-scale aggregation loss function constrains the semantic features after the nearest neighbor upsampling operation, which substantially lowers the uncertainty of the pyramidal feature map fusion and improves the accuracy of Model A3.

4.6. Hyperparameter analysis

On the Toronto3D dataset, we verified the effect of the multiscale voxel resolution $r = \{r_1, r_2, r_3\}$ and the weight parameters α and β in Eq. (18).

As shown in Fig. 12, r represents different combinations of multi-scale voxel resolutions, α and β are constants used to balance the magnitude of the loss function. When $r = \{0.25,0.5,1\}$, regardless of the values of α and β, the OA and mIoU were close to or exceeded the other values of r. The span of these three voxel resolutions was wide enough to cover the target objects in urban scenes, which makes the MVGF module extract and fuse different grained features efficiently. Therefore, we used the set of resolutions $r = \{0.25, 0.5, 1\}$ to extract coarse-grained voxel features.

When $(\alpha,\beta) = (0.5,0.5)$, the OA and mIoU were comparable with or outperformed the other values of α and β at $r = \{0.25,0.5,0.75\}$, and $r = \{0.25, 0.5, 1\}$. However, when $(\alpha,\beta) = (0.6, 0.4)$, under $r = \{0.35, 0.5, 0.65\}$, and $r = \{0.35, 0.5, 1\}$, both the OA and mIoU were close to or outperformed the other values of α and β. To efficiently extract the urban scene features, we used the optimum results obtained under $r = \{0.25, 0.5, 1\}$, i.e., $(\alpha,\beta) = (0.5, 0.5)$ to balance the loss functions and extract features.

5. Conclusion

In this paper, we proposed a multi-scale voxel-point adaptive fusion network, MVP-Net, for semantically segmenting large-scale LiDAR point clouds in urban scenes. MVP-Net first used the multi-scale voxel gating fusion module to acquire multi-scale semantic features, which facilitated the diverse representation of model features. Then, based on the geometric self-attention mechanism, the point-voxel adaptive fusion module aggregated both the fine-grained and coarse-grained features to fully extract contextual semantic information. Finally, the pyramid decoder extracted feature maps at different scales and fused the full-sized feature maps to obtain semantic segmentation results. MVP-Net has been extensively evaluated on three urban scene datasets. On the Toronto3D, WHU-MLS, and SensatUrban datasets, MVP-Net improved by 2.37%, 5.94%, and 6.71% of mIoU compared to the baseline, i.e., RandLA-Net, respectively. Quantitative results and visual inspection demonstrated that the proposed MVP-Net achieved a promising point cloud semantic segmentation performance in large-scale and complex urban scenes with variedly-scaled road objects.

CRediT authorship contribution statement

Huchen Li: Conceptualization, Software, Methodology, Writing – original draft, Writing – review & editing. **Haiyan Guan:** Conceptualization, Methodology, Writing – original draft, Writing – review &

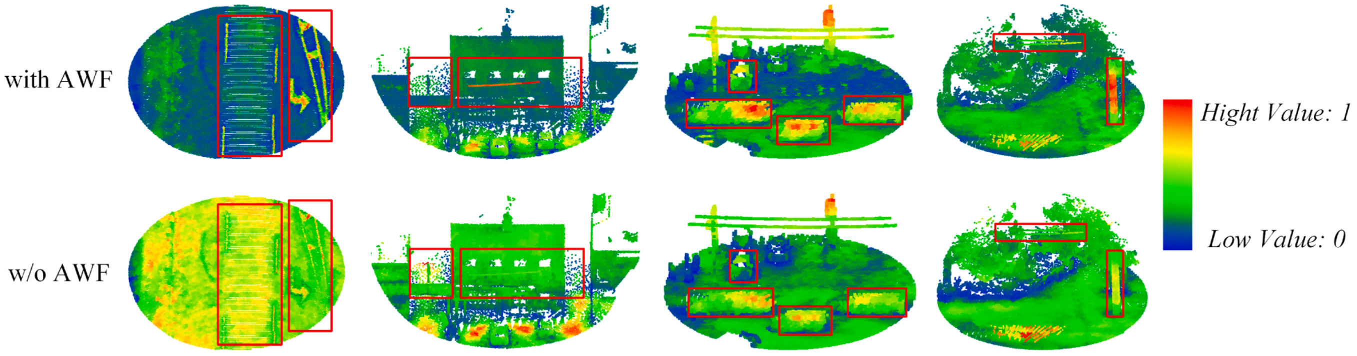


Fig. 11. Comparison of visualization results of multi-layer feature maps on the Toronto3D dataset. AWF: Adaptive Weighted Fusion.

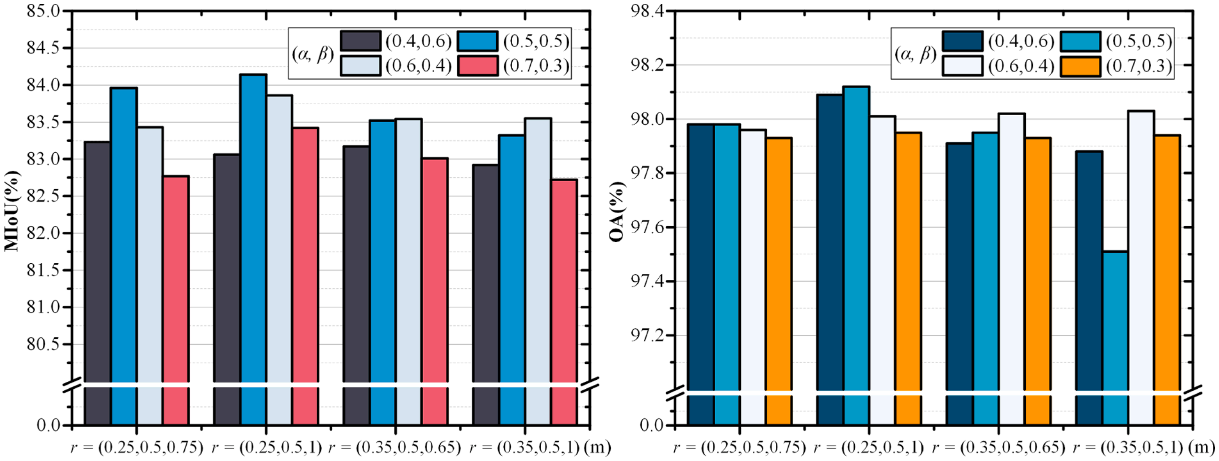


Fig. 12. The OA and mIoU obtained with different values of *r*.

editing, Project administration, Funding acquisition. **Lingfei Ma:** Validation, Formal analysis, Investigation, Supervision, Writing – review & editing, Funding acquisition. **Xiangda Lei:** Data curation, Methodology, Software, Writing – review & editing. **Yongtao Yu:** Writing – review & editing, Funding acquisition. **Hanyun Wang:** Writing – review & editing, Funding acquisition. **Mahmoud Reza Delavar:** Writing – review & editing. **Jonathan Li:** Resources, Supervision, Writing – review & editing.

Declaration of Competing 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.

Data availability

The authors do not have permission to share data.

Acknowledgment

This research was supported in part by the National Natural Science Foundation of China [41971414, 62076107, 42101451, 42271457], in part by the Postgraduate Research & Practice Innovation Program of Jiangsu Province [KYCX22_1212], and also in part by the Emerging Interdisciplinary Project of Central University of Finance and Economics in China.

References

Aksoy, E.E., Baci, S., Cavdar, S., 2020. Salsanet: Fast road and vehicle segmentation in lidar point clouds for autonomous driving. In: Proc. IV, pp. 926-932. https://doi.org/10.1109/IV47402.2020.9304694.

Chen, C., Wang, Y., Chen, H., Yan, X., Ren, D., Guo, Y., Wei, M., 2022. GeoSegNet: Point Cloud Semantic Segmentation via Geometric Encoder-Decoder Modeling. arXiv preprint arXiv:2207.06766.

Cheng, R., Razani, R., Taghavi, E., Li, E., Liu, B., 2021. (AF)2–S3Net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network. In Proc. CVPR 12547–12556. https://doi.org/10.48550/arXiv.2102.04530.

Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O., 2016. 3D U-Net: learning dense volumetric segmentation from sparse annotation. In Proc. MICCAI 424–432. https://doi.org/10.48550/arXiv.1606.06650.

Du, J., Cai, G.R., Wang, Z.Y., Huang, S.F., Su, J.H., Junior, J.M., Smit, J., Li, J., 2021. ResDLPS-Net: Joint residual-dense optimization for large-scale point cloud semantic segmentation. ISPRS J. Photogramm. Remote Sens. 182, 37–51.

Fan, S., Dong, Q., Zhu, F., Lv, Y., Ye, P., Wang, F. Y., 2021. SCF-Net: Learning spatial contextual features for large-scale point cloud segmentation. In: Proc. CVPR, pp. 14504-14513.

Graham, B., Engelcke, M., Van Der Maaten, L., 2018. 3D semantic segmentation with submanifold sparse convolutional networks. In: Proc. CVPR, pp. 9224-9232. https://doi.org/10.48550/arXiv.1711.10275.

Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., Bennamoun, M., 2020. Deep learning for 3d point clouds: A survey. IEEE Trans. Pattern Anal. Mach. Intell. 43 (12), 4338–4364.

Han, X., Dong, Z., Yang, B., 2021. A point-based deep learning network for semantic segmentation of MLS point clouds. ISPRS J. Photogramm. Remote Sens. 175, 199–214.

Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A., 2020. RandLA-Net: Efficient semantic segmentation of large-scale point clouds. In: Proc. CVPR, pp. 11105-11114. https://doi.org/10.1109/CVPR42600.2020.01112.

Hu, Q., Yang, B., Khalid, S., Xiao, W., Trigoni, N., Markham, A., 2022. Sensaturban: Learning semantics from urban-scale photogrammetric point clouds. Int. J. Comput. Vis. 130 (2), 316–343. https://doi.org/10.1007/s11263-021-01554-9.

Jing, Z., Guan, H., Zang, Y., Ni, H., Li, D., Yu, Y., 2021. Survey of Point Cloud Semantic Segmentation Based on Deep Learning. J. Front. Comp. Sci. Tech. 15 (1), 1–26. https://doi.org/10.3778/j.issn.1673- 9418.2006025.

Lehtola, V.V., Koeva, M., Elberink, S.O., Raposo, P., Virtanen, J.-P., Vahdatikhaki, F., Borsci, S., 2022. Digital twin of a city: Review of technology serving city needs. Int. J. Appl. Earth Obs. Geoinf. 114, 102915.

Lei, X., Guan, H., Ma, L., Yu, Y., Dong, Z., Gao, K., Delavar, M., Li, J., 2022. WSPointNet: A multi- branch weakly supervised learning network for semantic segmentation of large-scale mobile laser scanning point clouds. Int. J. Appl. Earth Obs. Geoinf. 115, 103129 https://doi.org/10.1016/j.jag.2022.103129.

Liong, V. E., Nguyen, T. N. T., Widjaja, S., Sharma, D., Chong, Z. J., 2020. AMVNet: Assertion-based multi-view fusion network for lidar semantic segmentation. arXiv preprint arXiv:2012.04934.

Liu, Z., Tang, H., Lin, Y., Han, S., 2019. Point-voxel CNN for efficient 3D deep learning. Adv. NeurIPS 32.

Lyu, Y., Huang, X., Zhang, Z., 2022. EllipsoidNet: Ellipsoid representation for point cloud classification and segmentation. In Proc. WACV 854–864. https://doi.org/10.48550/arXiv.2103.02517.

Mao, Y., Sun, X., Diao, W., Chen, K., Guo, Z., Lu, X., Fu, K., 2022. Semantic Segmentation for Point Cloud Scenes via Dilated Graph Feature Aggregation and Pyramid Decoders. In: arXiv preprint arXiv:2204.04944.

Milioto, A., Vizzo, I., Behley, J., Stachniss, C., 2019. RangeNet++: Fast and accurate lidar semantic segmentation. In: Proc. IROS, pp. 4213-4220. https://doi.org/10.1109/IROS40897.2019.8967762.

Qi, C.R., Su, H., Mo, K., Guibas, L.J., 2017a. PointNet: deep learning on point sets for 3D classification and segmentation. In: Proc. CVPR, pp. 77-85. https://doi.org/10.1109/CVPR.2017.16.

Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017b. PointnNet++: Deep hierarchical feature learning on point sets in a metric space. In: Proc. NeurIPS, pp. 5099-5108. http://arxiv.org/abs/1706.02413.

Qin, Z., Yu, H., Wang, C., Guo, Y., Peng, Y., Xu, K., 2022. Geometric transformer for fast and robust point cloud registration. In: Proc. CVPR, pp. 11143-11152. https://doi.org/10.48550/arXiv.2202.06688.

Qiu, S., Anwar, S., Barnes, N., 2021. Semantic segmentation for real point cloud scenes via bilateral augmentation and adaptive fusion. In: Proc. CVPR, pp. 1757-1767. https://doi.org/10.48550/arXiv.2103.07074.

Riegler, G., Osman Ulusoy, A., Geiger, A., 2017. OctNet: Learning deep 3d representations at high resolutions. In Proc. CVPR 3577–3586. https://doi.org/10.48550/arXiv.1611.05009.

Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E., 2015. Multi-view convolutional neural networks for 3D shape recognition. In Proc. ICCV 945–953. https://doi.org/10.1109/ICCV.2015.114.

Tan, W., Qin, N., Ma, L., Li, Y., Du, J., Cai, G., Yang, K., Li, J., 2020. Toronto-3D: A large-scale mobile lidar dataset for semantic segmentation of urban roadways, in. In: Proc. CVPR Workshops, pp. 797- 806. https://doi.org/10.1109/CVPRW50498.2020.00109.

Tang, H., Liu, Z., Zhao, S., Lin, Y., Lin, J., Wang, H., Han, S., 2020. Searching efficient 3d architectures with sparse point-voxel convolution. In: Proc. ECCV, pp. 685-702. https://doi.org/10.1007/978-3- 030-58604-1_41.

Thomas, H., Qi, C.R., Deschaud, J.E., Marcotegui, B., et al., 2019. KPConv: Flexible and deformable convolution for point clouds. In: Proc. ICCV, pp. 6411–6420. https://doi.org/10.1109/ICCV.2019.00651.

Varney, N., Asari, V. K. 2022. Pyramid point: A multi-level focusing network for revisiting feature layers. IEEE Geosci. Remote Sens. Lett., https://doi.org/10.48550/arXiv.2011.08692.

Wang, L., Huang, Y., Shan, J., He, L., 2018. MSNet: multi-scale convolutional network for point cloud classification. Remote Sens. 10 (4), 612.

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J., 2015. 3D ShapeNets: A deep representation for volumetric shapes. In: Proc CVPR, pp. 1912-1920.

Wu, B., Wan, A., Yue, X., Keutzer, K., 2018. SqueezeSeg: Convolutional neural nets with recurrent CRF for real-time road-object segmentation from 3D LiDAR point cloud. In Proc. ICRA, pp. 1887-1893. https://doi.org/10.1109/ICRA.2018.8462926.

Xu, J., Zhang, R., Dou, J., Zhu, Y., Sun, J., Pu, S., 2021a. RPVNet: A deep and efficient range-point-voxel fusion network for lidar point cloud segmentation. In: Proc. ICCV, pp. 16024-16033. https://doi.org/10.48550/arXiv.2103.12978.

Xu, J., Tang, X., Zhu, Y., Sun, J., Pu, S., 2021b. SGMNet: Learning rotation-invariant point cloud representations via sorted Gram matrix. In: Proc. ICCV, pp. 10448-10457.

Yan, K., Hu, Q., Wang, H., Huang, X., Li, L., Ji, S., 2021. Continuous mapping convolution for large- scale point clouds semantic segmentation. IEEE Geosci. Remote Sens. Lett. 19, 1–5.

Yang, B., Han, X., Dong, Z., 2021. Point cloud benchmark dataset WHU-TLS and WHU-MLS for deep learning. J. Remote Sens. 25 (1), 231–240.

Ye, M., Xu, S., Cao, T., Chen, Q., 2021a. DRINet: A dual-representation iterative learning network for point cloud segmentation. In: Proc. ICCV, pp. 7447-7456. https://doi.org/10.48550/arXiv.2108.04023.

Ye, M., Wan, R., Xu, S., Cao, T., Chen, Q., 2021b. DRINet++: Efficient Voxel-as-point Point Cloud Segmentation. In: arXiv preprint. https://doi.org/10.48550/arXiv.2111.08318.

Zeng, Z., Xu, Y., Xie, Z., Tang, W., Wan, J., Wu, W., 2022. LACV-Net: Semantic Segmentation of Large- Scale Point Cloud Scene via Local Adaptive and Comprehensive VLAD. arXiv preprint arXiv:2210.05870.

Zhang, F., Fang, J., Wah, B., Torr, P., 2020. Deep fusionnet for point cloud semantic segmentation. In Proc. ECCV 644–663. https://doi.org/10.1007/978-3-030-58586-0_38.

Zhou, H., Zhu, X., Song, X., Ma, Y., Wang, Z., Li, H., Lin, D., 2020. Cylinder3D: An effective 3D framework for driving-scene lidar semantic segmentation. In: arXiv preprint. https://doi.org/10.48550/arXiv.2008.01550.