

# CSFNet: Cross-Modal Semantic Focus Network for Semantic Segmentation of Large-Scale Point Clouds

Yang Luo<sup>1</sup>, Ting Han<sup>1</sup>, Graduate Student Member, IEEE, Yujun Liu<sup>2</sup>, Jinhe Su<sup>1</sup>,  
Yiping Chen<sup>1</sup>, Senior Member, IEEE, Jinyuan Li<sup>1</sup>, Yundong Wu, and Guorong Cai<sup>1</sup>, Senior Member, IEEE

**Abstract**—Semantic segmentation of large-scale point clouds is an indispensable component of outdoor scene perception, providing essential 3-D semantic insights for applications in scene reconstruction, urban planning, autonomous driving, and more. However, the discriminative capability of point clouds features declines with increasing distance from the sensor, causing current methods to usually perform poorly in segmenting distant objects. To overcome this challenge and improve the differentiation between classes with similar geometric features, we propose the cross-modal semantic focus network (CSFNet). Firstly, we design a multiscale feature dynamic fusion (MDF) module to leverage multiscale image features, thereby enriching the feature representation of point clouds with additional images color and texture information. Then, in order to extract the distinguishing features of distant and different categories of objects more efficiently, we propose a semantic focus module (SFM) that employs a multiclass contrastive learning strategy to enhance feature discrimination. Finally, we introduce cross-modal knowledge distillation (KD) to augment the model's comprehension of point clouds. Extensive experiments conducted on the SemanticKITTI and nuScenes datasets demonstrate the effectiveness of our method. Notably, our method achieves superior segmentation accuracy across multiple classes at various distances compared to current methods.

**Index Terms**—Contrastive learning, point clouds, semantic focus, semantic segmentation, urban scenes.

## I. INTRODUCTION

SEMANTIC segmentation of large-scale point clouds, a critical task in vehicle-borne and airborne remote sensing data processing [1], aims to categorize each point in 3-D

space into a distinct semantic class. This intricate process is fundamental for the accurate perception of objects and scenes, playing a crucial role in a wide array of applications such as urban planning [2], scene reconstruction [3], and autonomous driving [4]. LiDAR point clouds have become a crucial data source for urban scene semantic perception [5], [6], [7], because of their capacity to offer high-precision distance information and robustness across diverse environmental conditions. Consequently, the development of semantic segmentation algorithms for efficient LiDAR data processing is key to enhancing complex urban scene comprehension and automation. However, due to the large-scale of 3-D scenes and the complexity of urban environments, 3-D point cloud semantic segmentation faces two main challenges that significantly impact the design and performance of algorithms as follows.

- 1) *Difficulty Distinguishing Similar Classes*: Urban scenes contain a multitude of classes with similar geometric characteristics, and the subtle differences between them pose a significant challenge for semantic segmentation algorithms. While LiDAR data provides precise geometric information, the absence of color and texture further complicates the task of distinguishing between these similar classes. As illustrated in the first and second rows of Fig. 1, SPVCNN [8] frequently misclassifies sidewalks as roads and parks as sidewalks due to their geometric similarities.
- 2) *Reduced Feature Discriminability at Long Distances*: The operational principle of LiDAR sensors leads to a decrease in point cloud density as distance increases. Consequently, point cloud data becomes sparse at longer ranges, complicating the extraction of reliable geometric and intensity-based features from these sparse point clouds. In complex urban environments, accurately extracting features for distant objects is essential for precise classification. Additionally, the reduced feature discriminability at long distances hinders the algorithm's ability to clearly differentiate between objects. As seen in the third and fourth rows of Fig. 1, SPVCNN misclassifies points from the person and other vehicle categories due to their sparse representation at longer distances.

Recent research in semantic segmentation of large-scale point clouds categorizes methods into LiDAR-only and cross-modal fusion approaches. LiDAR-only methods utilize point-based, projection-based, or voxel-based strategies. Point-based methods [9], [10], [11] process raw point clouds directly,

Received 23 April 2024; revised 20 October 2024 and 18 December 2024; accepted 23 January 2025. Date of publication 28 January 2025; date of current version 27 February 2025. This work was supported in part by the Natural Science Foundation of Xiamen, China, under Grant 3502Z202373036; in part by the National Natural Science Foundation of China under Grant 42371457, Grant 42301468, and Grant 42371343; in part by the Key Project of Natural Science Foundation of Fujian Province, China, under Grant 2022J02045; in part by the Natural Science Foundation of Fujian Province, China, under Grant 2022J01337, Grant 2022J01819, Grant 2023J01801, Grant 2023J01799, Grant 2022J05157, and Grant 2022J011394; in part by the Xiamen City's Leading Project, under Grant 3502Z20231038; and in part by the Natural Science Foundation of Xiamen, China, under Grant 3502Z202371019. (Corresponding authors: Jinhe Su; Guorong Cai.)

Yang Luo, Jinhe Su, Jinyuan Li, Yundong Wu, and Guorong Cai are with the School of Computer Engineering, Jimei University, Xiamen 361021, China (e-mail: luoy1014116@gmail.com; sujh@jmu.edu.cn; lijinyuanc@gmail.com; yundongwu@jmu.edu.cn; guorongcai.jmu@gmail.com).

Ting Han and Yiping Chen are with the School of Geospatial Engineering and Science, Sun Yat-sen University, Zhuhai 519082, China (e-mail: ting.devin.han@gmail.com; chenyp79@mail.sysu.edu.cn).

Yujun Liu is with the School of Architecture and Urban Planning, Shenzhen University, Shenzhen 518061, China (e-mail: yujunliu.jmu@gmail.com).

Digital Object Identifier 10.1109/TGRS.2025.3535800

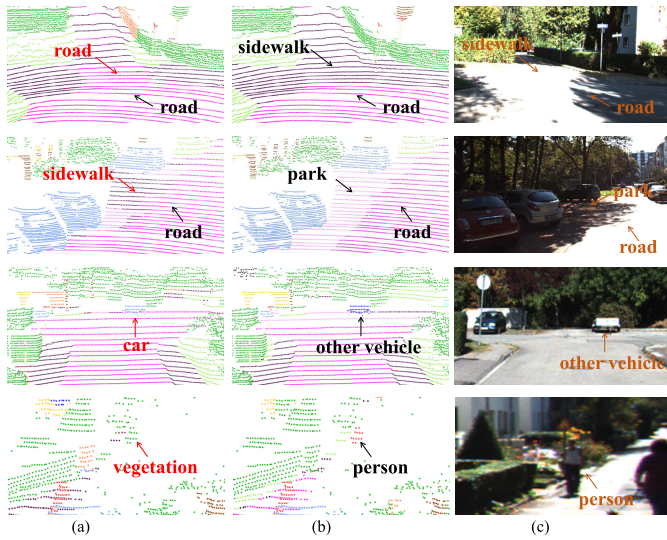


Fig. 1. Illustration of semantic segmentation challenges in urban LiDAR point clouds. Red indicates incorrect predictions, while black represents correct predictions. (a) SPVCNN. (b) Ours. (c) Image.

while projection-based approaches [12], [13] project the clouds onto a 2-D plane and use conventional segmentation networks before re-projecting the outcomes. Voxel-based methods [14], [15], [16], on the other hand, convert point clouds into structured voxel grids and apply sparse convolutions [17]. Despite their strengths, these LiDAR-only approaches often suffer from issues such as data non-uniformity and sparsity, especially in distant regions, leading to challenges in accurately distinguishing between similar object classes and segmenting faraway objects. To address these limitations, cross-modal fusion methods [18], [19], [20], [21], [22], [23], [24], [25] combine LiDAR data with RGB images, integrating visual features such as color and texture to enhance feature representation and scene understanding. While cross-modal fusion improves overall segmentation performance, existing approaches still struggle to maintain feature discriminability for distant objects due to the sparsity of LiDAR points and the difficulty in distinguishing geometrically similar classes.

In addition, contrastive learning has made rapid advancements in the field of feature learning in recent years. The remarkable success of CLIP [26] has further demonstrated that contrastive learning can significantly enhance a model's ability to discern fine-grained geometric details within point clouds. However, in the domain of point cloud semantic segmentation, the potential of contrastive learning to differentiate between point cloud features belonging to distinct semantic categories remains largely unexplored.

Motivated by these, we propose the cross-modal semantic focus network (CSFNet). Specifically, CSFNet integrates two core components: the multiscale feature dynamic fusion (MDF) module and the semantic focus module (SFM). MDF dynamically fuses multiscale features from RGB images with point cloud features, leveraging the inherent characteristics present in various depths of convolutional layers within images to enhance the model's perceptual acuity. This fusion notably improves the representational strength of point cloud features, enabling the model to capture a more comprehensive range of

detail within complex scenes. SFM implements a multiclass contrastive learning strategy that emphasizes the similarities among features within the same class and the differences among features across different classes, thereby boosting the model's discriminative capability. This targeted contrastive learning approach significantly enhances the discriminative power of point cloud features, improving recognition across diverse distances and classes. Moreover, we have introduced cross-modal knowledge distillation (KD) into CSFNet, utilizing the knowledge garnered from the fused modalities to further augment the model's comprehension of point clouds.

In evaluations conducted on the publicly available SemanticKITTI [27] and nuScenes datasets [28], our approach exhibits superior performance relative to the current state-of-the-art (SOTA) methods, achieving mean Intersection over Union (mIoU) scores of 71.5% and 80.8%, respectively. Our method has been validated across various distance ranges, consistently surpassing leading contemporary approaches. Notably, it effectively addresses challenges such as the difficulty in distinguishing similar object classes and the reduced feature discriminability at long distances, thereby providing valuable insights for applications in urban remote sensing, geographic information system (GIS), autonomous driving, and so on. Importantly, it not only attains SOTA performance but also features a reduced parameter count and shorter inference times compared to most prevailing methods.

Our main contributions are summarized as follows.

- 1) We construct MDF to dynamically fuse point clouds features with multiscale image features, enhancing the model's capacity to accurately perceive detailed and complex scenes.
- 2) We propose SFM to employ multiclass contrastive learning to amplify the similarities within classes and the differences between them, thereby boosting the discriminative power of point clouds features.
- 3) Our method achieves SOTA mIoU on the SemanticKITTI and nuScenes datasets with greater efficiency. Moreover, it shows better performance than current methods across various distances and classes.

## II. RELATED WORK

### A. LiDAR Point Clouds-Only Methods

Concerning the various representation methods for LiDAR point clouds, there are currently three main categories of LiDAR point clouds-only methods.

Point-based methods perform representation learning directly on the original point clouds. PointNet [29] and PointNet++ [30], as pioneering works in this field, employ multilayer perceptrons for point-wise feature extraction from individual points, and utilize multiscale grouping strategies to discern the local structural patterns within point clouds. Subsequent studies [31], [32], [33], [34], [35] have introduced a range of advanced sampling techniques, feature aggregation processes, and enhancement mechanisms, aiming to optimize the feature extraction of key points. Point Transformer [9] introduces self-attention layers specifically for point clouds. Building on this, Point Transformer v2

[10] enhances the model by incorporating grouped vector attention and a partition-based pooling mechanism, aimed at improving processing efficiency and model performance. Point Transformer v3 [11] further optimizes the model's simplicity and computational efficiency by organizing unstructured, irregular point cloud data into structured sequences according to specific patterns, emphasizing improvements in the model's architecture and computational efficiency.

Projection-based methods, such as RangeNet++ [36], SqueezeSeg [37], [38], [39], and SalsaNext [40], transform 3-D point clouds data into 2-D images using perspective or spherical projection. Traditional convolutional neural networks are then used to process the range images, with the semantic segmentation results subsequently being reprojected back onto the original 3-D point clouds. RangeVit [13] formulates the segmentation of range view grids as a sequence transformation problem, leveraging the powerful representational learning capabilities of the vision transformer to process range images, thereby facilitating 3-D per-point predictions.

Voxel-based methods effectively partition irregular point clouds into structured voxel grids, utilizing sparse convolutions [17] to enhance network efficiency. For instance, SPVCNN [8] introduces a lightweight 3-D module called sparse point-voxel convolution (SPVConv), which combines standard sparse convolutions with a high-resolution, point-based branch. Similarly, AF2-S3Net [41] partitions continuous 3-D space into regular grids, leveraging attention mechanisms to extract features across branches with varying kernel sizes. In contrast, Cylinder3D [14] innovatively transforms conventional grid voxels into a cylindrical format. Furthermore, SDSeg3D [15] enhances point cloud encoding by modeling and preserving local relationships among points within each voxel using multihead attention. OA-CNNS [42] proposes an omni-adaptive 3-D sparse CNN that improves the performance of conventional sparse CNNs for 3-D semantic segmentation.

Despite these methods have achieved significant achievements in enhancing the accuracy and efficiency of point clouds semantic segmentation, challenges persist, particularly regarding the resolution and quality of point clouds at increased distances. As the distance from the sensor extends, the density and clarity of point clouds diminish, leading to a gradual reduction in the discriminability of extracted features. This degradation affects the algorithms' ability to precisely distinguish distant objects, posing limitations on the segmentation capabilities for objects situated at greater ranges.

## B. Cross-Modal Fusion Methods

Compared to the non-uniformity and sparsity of LiDAR point clouds, images are 2-D high-density data with rich color and texture information, offering advantages in complex scene analysis that LiDAR point clouds lack. Consequently, in recent years, researchers [25], [43], [44] have begun to explore the fusion of LiDAR point clouds and images to enhance the accuracy and robustness of point cloud semantic segmentation. PMF [18] initially uses perspective projection to map LiDAR data onto camera coordinates, then extracts features of multimodal data through a dual-stream network, and fuses

the multimodal features through multiple residual-based fusion blocks. 2DPASS [19] enhances the 3-D network by using cross-modal KD during training, leveraging the texture and color perception information from images while preserving the original 3-D information. Unlike PMF and 2DPASS, which extract modal features separately, MSeg3D [20] jointly optimizes intra-modal feature extraction and inter-modal feature fusion to maximize the correlation and complementarity between heterogeneous modalities. Moreover, by applying asymmetric transformations to both LiDAR point clouds and camera images, multimodal data augmentation is performed, introducing diverse enhancement changes to model training. UniSeg [21] introduces learnable cross-modal and cross-view association modules, achieving the fusion of various representations of point clouds and image features, and using comprehensive information to enhance perceptual robustness. PIF-Net [45] proposes a hierarchical fusion module to adaptively fuse different levels of multimodal features to ensure they are fully fused. Recent advancements in vision foundation models (VFMs) have opened up new possibilities for 3-D perception. Liu et al. [46] segment point cloud sequences by distilling VFMs, offering a novel approach to sequence-based segmentation.

To summarize, cross-modal fusion methods employ various fusion strategies to integrate image information, compensating for the inherent limitations of LiDAR point clouds. However, these methods do not scale efficiently to longer distances. Our method focuses on the similarities between the same categories and the differences between different categories, and uses multicategory contrastive learning to effectively enhance point cloud feature discrimination.

## C. Contrastive Learning in Multimodal 3-D Representation

Contrastive learning has demonstrated promising performance in scene understanding. With contrastive learning being widely used for image segmentation, it is now gradually being applied to point cloud segmentation and multimodal point cloud segmentation. PointContrast [47] uses point-wise contrastive learning to learn the geometric information. CBL [48] conducts contrastive learning between raw point cloud and subsampled sets for improving boundary segmentation. Cross-Net [49] and OpenScene [50] propose the joint learning to strengthen the cross-modal correspondence to align the 2-D and 3-D representation. Most recently, CoIn [51] constructs a rich negative sample space with a limited number of feature instances, which enhances feature discriminability and thus better identifies difficult-to-distinguish detailed features in point clouds. CLIP2Scene [52] designs a semantics-driven cross-modal contrast learning framework to train 3-D networks through semantic and spatio-temporal consistency regularization. In both PPT [53] and M3Net [54], contrastive learning is applied in the feature embedding space and the inference result space, respectively, to achieve multimodal alignment.

Despite the advances made by existing methods, there remains untapped potential in fully exploiting the semantic feature space for point cloud segmentation. To address this gap, we propose an SFM, leveraging contrastive learning to

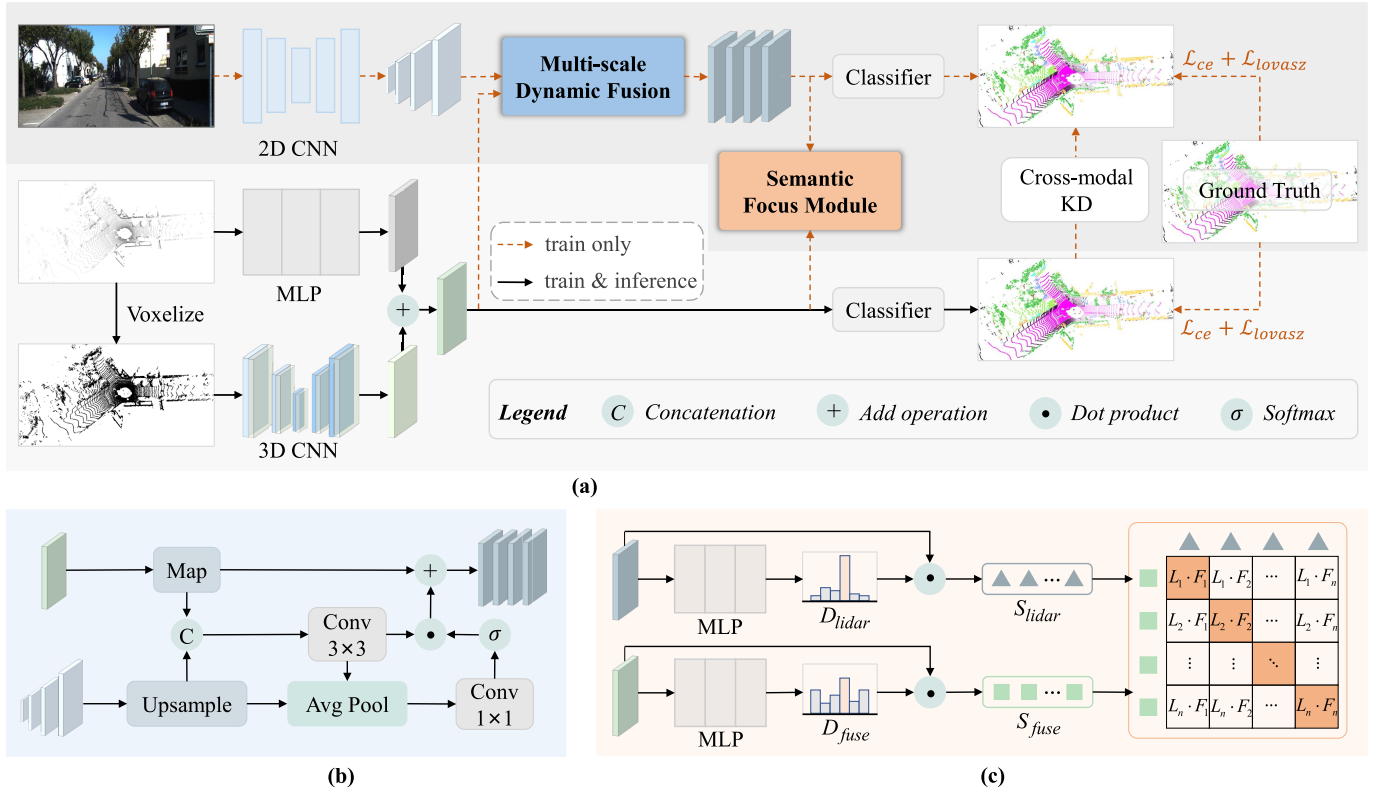


Fig. 2. Structure of the CSFNet. (a) Network architecture (Section III-A), the modules with darker backgrounds are used only during training. (b) Multiscale dynamic fusion (Section III-B), where “map” refers to the process of utilizing the mapping relationship between point clouds and images to find matching point–pixel pairs. (c) SFM (Section III-C), where  $L_n$  represents the semantic features of the  $n$ th category from the LiDAR point cloud, and  $F_n$  denotes the semantic features of the  $n$ th category after fusion.

further enhance the distinction between features belonging to different semantic categories.

### III. PROPOSED METHOD

In this section, we describe the proposed method in detail. In Section III-A, we will show an overview of our framework. Section III-B shows the multiscale feature fusion module, describing the fusion process of point clouds feature with multiscale images feature. In Section III-C, we discuss the application of multicategory contrastive learning in enhancing the discriminative power of point cloud features. Finally, we present the loss function (Section III-D) deployed for network training.

#### A. Network Architecture

CSFNet is designed to process inputs from LiDAR point clouds and RGB images, subsequently assigning semantic labels to each point. Specifically,  $P_{in} \in \mathbb{R}^{N_{point} \times 4}$  represents the LiDAR point clouds consisting of  $N_{point}$  points, each with 4-D input feature: 3-D coordinates and reflectance.  $I_{in} \in \mathbb{R}^{N_{cam} \times H \times W \times 3}$  refers to RGB images from  $N_{cam}$  cameras, each with a uniform resolution of  $H$  height and  $W$  width, encompassing 3-D RGB color information. The comprehensive architecture of our proposed CSFNet (See Fig. 2) comprises four key components: feature extraction, multiscale dynamic fusion, SFM, and cross-modal KD. These components are designed to synergize, leading to an effective and efficient

semantic segmentation process. Each of these components is explained in further detail below.

1) *Feature Extraction*: We adopt the SPVConv, as introduced by SPVCNN [8], to serve as our 3-D encoder for extracting feature from the input point clouds data. Specifically, SPVConv employs a multilayer perceptron (MLP) to directly extract per-point feature from the raw point cloud. Concurrently, the raw point cloud is voxelized, and sparse convolution operations are utilized to extract voxel feature from these voxelized representations. Subsequently, a devoxelization process is applied to these voxel features, which are then fused with the per-point feature to obtain the final comprehensive point clouds feature  $F_P$  as follows:

$$F_P = \text{MLP}(P_{in}) \oplus \psi(\text{SpConv}(\varphi(P_{in}))) \quad (1)$$

$\varphi$  and  $\psi$  are operations denoting voxelization and devoxelization, respectively. The operation SpConv stands for sparse convolution, and the symbol  $\oplus$  is used to represent element-wise addition.

In our methodology, we utilize the ResNet [55] framework as the 2-D encoder to extract multiscale image features  $F_i^s \in \mathbb{R}^{(H/(s) \times (W/(s) \times C_s)}$ . Among them,  $s = 2, 4, 8, 16$  represents the scale factor,  $C_s$  represents the number of channels in the corresponding scale feature map. Through progressive down-sampling of the image resolution, there is a corresponding increase in the number of feature channels. This approach is designed to achieve a detailed and comprehensive feature

representation across different layers of the network, capturing the nuanced characteristics of the input images.

2) *Multiscale Feature Dynamic Fusion*: The representation of distant objects in point clouds data is often sparse, posing significant challenges in extracting high-quality features from low-density point clouds. Additionally, distinguishing and segmenting objects with similar 3-D structures, such as roads and sidewalks, and cars and other vehicles, present a considerable challenge. By integrating image features from multiple scales, it is possible to supplement the missing details in point clouds, enriching them with valuable color and texture information. Large-scale image features are adept at capturing the global context and overall environment of objects, whereas small-scale features focus on local details and textures. We individually fuse point cloud features with image features at each scale, improving the richness and robustness of features for segmentation. The fusion method is detailed in Section III-B.

3) *Semantic Focus Module*: Contrastive learning, a variant of unsupervised learning, concentrates on capturing the commonalities among akin instances and distinguishing the differences between dissimilar instances. MoCo [56] has achieved remarkable success in the domain of 2-D imagery by employing contrastive learning methodologies based on feature similarity, thereby validating the capability of contrastive learning to enhance the discriminative power of features. Inspired by this achievement, we adopt a multicategory contrastive learning strategy that leverages semantic category information of both fused features and point clouds features for comparative learning. This strategy amplifies the similarity within the same semantic categories and the distinctions between different semantic categories, thereby bolstering the discriminative ability of point clouds features and enhancing the model's proficiency in recognizing distant objects. We refer to this learning process as *semantic focus*, with the detailed computational method described in Section III-C.

4) *Cross-Modal KD*: When incorporating image features to enhance the richness and robustness of point cloud representations, a significant challenge arises from the increased model complexity and computational demands, which lead to prolonged inference times. To mitigate this issue, we employ cross-modal KD techniques. During training, the teacher network fuses both image and point cloud features, leveraging multiscale image information to enhance the representation of point cloud features. In contrast, the student network processes only point cloud data. Knowledge is effectively transferred from the teacher to the student network via KD, allowing the student network to perform semantic segmentation using point cloud data exclusively during inference. This strategy eliminates the need for image feature extraction, alignment, and fusion with point clouds during inference, significantly reducing computational overhead. Consequently, the model retains high segmentation accuracy while greatly improving inference speed, making it more efficient and practical for deployment in complex real-world applications.

## B. Multiscale Feature Dynamic Fusion

To facilitate the efficient fusion of point clouds features  $F_P$  and multiscale images features  $F_I^s$ , perspective projection is employed to establish a mapping relationship between the point clouds and image pixels. Specifically, for each point  $p_i = (x_i, y_i, z_i)$ , it is projected onto the image plane, and the mapping relationship from a point to a pixel can be expressed as follows:

$$\begin{aligned} [u_i, v_i, 1]^T &= \frac{1}{Z_i} \cdot K \cdot T \cdot [x_i, y_i, z_i, 1]^T \\ M^{p2i} &= (\lfloor v_i \rfloor, \lfloor u_i \rfloor)_{i=1}^{N_m} \end{aligned} \quad (2)$$

where  $(u_i, v_i)$  represents the coordinates of the projected point of  $p_i$ , and the values of  $u_i$  and  $v_i$  are subjected to a floor operation ( $\lfloor \cdot \rfloor$ ) to determine the corresponding pixel coordinates.  $N_m$  represents the number of point-to-pixel mappings.  $K \in \mathbb{R}^{3 \times 4}$  and  $T \in \mathbb{R}^{4 \times 4}$  represent the intrinsic matrix and extrinsic matrix of the camera, respectively. This process corresponds to the “map” in Fig. 2(b).

Based on the point-to-pixel mapping relationship, traditional methods typically concatenate single-scale image features with corresponding points one by one. However, this approach often results in insufficient spatial awareness in the system. Shallow features and deep features capture different aspects of the image: shallow features focus on basic and local details, while deep features encapsulate more global and semantic content. Therefore, as illustrated in Fig. 2(b), we first upsample the multiscale features obtained from the image branch to match the resolution of the original image blocks, ensuring that the number of feature channels is consistent with that of the point clouds features. Subsequently, by utilizing the point-to-pixel mapping, features of point-pixel pairs are concatenated along the feature dimension. Following this, we employ a simple yet effective channel attention module to dynamically select relevant fused features. Finally, by using residual connections, the fused features are added to the original point clouds features to enrich the expression of the final fused features. The computation is as follows:

$$\begin{aligned} \hat{F}_{\text{fuse}}^s &= \text{conv}(\text{concat}(M^{p2i}(F_P), U_p(F_I^s))) \\ F_{\text{fuse}}^s &= \left( \sigma \left( \text{conv} \left( f_{\text{pool}} \left( \hat{F}_{\text{fuse}}^s \right) \right) \right) \cdot \hat{F}_{\text{fuse}}^s \right) \oplus M^{p2i}(F_P) \end{aligned} \quad (3)$$

where  $\sigma$  denotes the sigmoid function,  $U_p$  denotes upsampling,  $\oplus$  represents element-wise addition within matrices,  $\text{conv}$  refers to the convolution operation, and  $f_{\text{pool}}$  signifies average pooling.

By fusing with multiscale image features, point clouds not only gain shallow features from images, such as object color, texture, and local shape, but also acquire advanced semantic features such as object categories and relationships among objects. This fusion enables the model to utilize both visual and geometric information, creating a more comprehensive and rich feature representation. As a result, the model can understand the entire scene more profoundly, enhancing recognition accuracy for long-range objects and objects with similar 3-D features.

**Algorithm 1** Semantic Focus Module

---

**Input:** Point cloud features  $\mathbf{F}_P \in \mathbb{R}^{N_{point} \times C}$  Multiscale  
Fused features  $\{\mathbf{F}_{fuse}^s \in \mathbb{R}^{N_m \times C}\}_{s \in \{2,4,6,8\}}$

**Output:** Semantic Focusing Loss  $\mathcal{L}_{sf}$

- 1  $\mathbf{D}_{lidar} = \text{softmax}(H_{point}(\mathbf{F}_P));$
- 2  $\mathbf{D}_{fuse}^s = \text{softmax}(H_{fuse}(\mathbf{F}_{fuse}^s));$  // Generate category confidence scores
- 3  $\mathbf{S}_{lidar} = \text{Transpose}(\mathbf{D}_{lidar}) \cdot \mathbf{F}_P;$
- 4  $\mathbf{S}_{fuse}^s = \text{Transpose}(\mathbf{D}_{fuse}^s) \cdot \mathbf{F}_{fuse}^s;$  // Compute semantic feature
- 5  $\mathbf{S}_{lidar} = \frac{\mathbf{S}_{lidar}}{\|\mathbf{S}_{lidar}\|_2};$
- 6  $\mathbf{S}_{fuse}^s = \frac{\mathbf{S}_{fuse}^s}{\|\mathbf{S}_{fuse}^s\|_2};$  // L2 normalization
- 7 Initialize  $\mathcal{L}_{sf} = 0;$
- 8 **for**  $s \in \{2, 4, 6, 8\}$  **do**
- 9    $V_{cs} = \mathbf{S}_{lidar} \cdot \text{Transpose}(\mathbf{S}_{fuse}^s);$  // Calculate similarity
- 10   labels =  $[0, 1, \dots, N_{class} - 1];$
- 11    $\mathcal{L}_1 = \text{CrossEntropy}(V_{cs}, \text{labels});$  // Align fused features to point cloud features
- 12    $\mathcal{L}_2 = \text{CrossEntropy}(\text{Transpose}(V_{cs}), \text{labels});$  // Align point cloud features to fused features
- 13    $\mathcal{L}_{sf} += \frac{\mathcal{L}_1 + \mathcal{L}_2}{2};$
- 14 **end**
- 15 **return** Semantic Focusing Loss  $\mathcal{L}_{sf}$

---

**C. Semantic Focus Module**

Following the semantic feature aggregation method proposed by MSeg3D [20], we converted point clouds features and fused features that incorporate images features of different scales into semantic features

$$\begin{aligned}
 \mathbf{D}_{lidar} &= \text{softmax}(H_{point}(\mathbf{F}_P)) \\
 \mathbf{S}_{lidar} &= \text{Transpose}(\mathbf{D}_{lidar}) \cdot \mathbf{F}_P \\
 \mathbf{D}_{fuse}^s &= \text{softmax}(H_{fuse}(\mathbf{F}_{fuse}^s)) \\
 \mathbf{S}_{fuse}^s &= \text{Transpose}(\mathbf{D}_{fuse}^s) \cdot \mathbf{F}_{fuse}^s.
 \end{aligned} \tag{4}$$

$H_{point}$  and  $H_{fuse}$  are MLP-based auxiliary segmentation heads used to obtain preliminary segmentation results, see Fig. 2(c).  $\mathbf{D}_{lidar} \in \mathbb{R}^{N_{point} \times N_{class}}$  and  $\mathbf{D}_{fuse}^s \in \mathbb{R}^{N_m \times N_{class}}$  are distribution matrices to represent the category confidence scores,  $D[i, j]$  closer to 1 indicates that the  $i$ th point is more likely to belong to the  $j$ th category compared to other categories. Subsequently, by transposing the distribution matrix  $\mathbf{D}_{lidar}$  and performing a matrix multiplication operation with the point cloud features  $\mathbf{F}_P$ ,  $\mathbf{D}_{fuse}^s$  and  $\mathbf{F}_{fuse}^s$  performing the same operation, we obtain the semantic features  $\mathbf{S}_{lidar}$  and  $\mathbf{S}_{fuse}^s$  in the latent semantic space. The dimensions of  $\mathbf{S}_{lidar}$  and  $\mathbf{S}_{fuse}^s$  are both  $\mathbb{R}^{N_{class} \times C}$ .

Subsequently, we compute the contrast loss between  $\mathbf{S}_{lidar}$  and  $\mathbf{S}_{fuse}^s$  when  $s = 2, 4, 8, 16$ , respectively, by multicategory contrast learning, which we call semantic focusing loss  $\mathcal{L}_{sf}$ . Specifically, we use cosine similarity as a metric to assess the degree of similarity between the vectors by calculating the

cosine value of the angle of the vectors, as illustrated in the following equation:

$$V_{cs}[i, j] = \frac{S_{lidar}^i}{\|S_{lidar}^i\|_2} \cdot \text{Transpose} \left( \frac{S_{fuse}^{s,j}}{\|S_{fuse}^{s,j}\|_2} \right) \tag{5}$$

where  $i$  and  $j$  both represent indices of semantic categories, and  $\|\cdot\|_2$  is represented as  $L2$  normalization. A  $V_{cs}$  value closer to 1 implies a higher degree of feature similarity. The training strategy is tailored to maximize the similarity among  $N_{class}$  semantic features, while concurrently minimizing the similarity between the  $N_{class}^2 - N_{class}$  distinct semantic feature combinations. The calculation method for  $\mathcal{L}_{sf}$  is as follows:

$$\begin{aligned}
 \mathcal{L}_{sf} = \sum_{s \in \{2,4,8,16\}} \sum_{i=1}^{N_{class}} \sum_{j=1}^{N_{class}} & [\delta_{ij}(1 - V_{cs}^s[i, j]) \\
 & + (1 - \delta_{ij})V_{cs}^s[i, j]]
 \end{aligned} \tag{6}$$

where  $\delta_{ij}$  is the Kronecker delta function. For  $\delta_{ij}$ , it equals 1 when  $i = j$  and 0 otherwise. The pseudo-code for the SFM is provided in Algorithm 1.

By applying cosine similarity to different semantic feature vectors, we can effectively identify and quantify the similarities and differences between features. The SFM not only helps to reveal the underlying semantic associations in the data, but also enhances the discriminative power of the features. This promotes a more accurate and reliable decision-making process in semantic segmentation through a deeper understanding of the complex relationships and distinctions between different classes of features. Extensive experiments confirm the SFM's effectiveness in improving segmentation for distant and similar objects.

**D. Loss Function**

The loss function assesses the difference between the predicted and ground truth disparity values. To jointly optimize all the modules in the proposed segmentation model, the overall loss function is composed of four parts: the cross-entropy loss  $\mathcal{L}_{ce}$ , the Lovasz-Softmax loss  $\mathcal{L}_{lovasz}$ , the semantic focusing loss  $\mathcal{L}_0$ , and the KD loss  $\mathcal{L}_{kd}$ . These are combined into the total loss  $\mathcal{L}$ , as shown in the following equation:

$$\mathcal{L} = \mathcal{L}_{ce} + \mathcal{L}_{lovasz} + \alpha \mathcal{L}_{sf} + \beta \mathcal{L}_{kd} \tag{7}$$

where  $\alpha$  and  $\beta$  are loss weight parameters. Where both  $\mathcal{L}_{ce}$  and  $\mathcal{L}_{lovasz}$  assess the differences between the predicted values of the point clouds and the ground truth, as well as between the predicted values of the fused modalities and the ground truth. Lovasz-Softmax loss is a loss function designed for unbalanced data in the field of semantic segmentation. The predictions of both the fusion branch and the point cloud branch will be computed  $\mathcal{L}_{ce}$  and  $\mathcal{L}_{lovasz}$  with the ground truth, see Fig. 2(a).

We utilize Kullback–Leibler (KL) divergence to calculate the distillation loss, aiming to align the point clouds' prediction results with those of the fused modalities. This approach enables the model to achieve predictions close to those obtained post-image feature fusion during inference,

using only point clouds data as input. Using  $\hat{y}_{\text{point}}$  to denote the predicted results of the point clouds and  $\hat{y}_{\text{fuse}}$  to denote the prediction results of the fused modalities, the methods for calculating  $\mathcal{L}_{\text{kd}}$  are as follows:

$$\mathcal{L}_{\text{kd}} = D_{\text{KL}}(\hat{y}_{\text{point}} \parallel \hat{y}_{\text{fuse}}). \quad (8)$$

#### IV. EXPERIMENTS

In this section, we begin by introducing the dataset and outlining the corresponding evaluation metrics. Secondly, we present the details of network implementation. Finally, we highlight the effectiveness of our method by providing both qualitative and quantitative results that demonstrate its superiority.

##### A. Datasets and Evaluation Metrics

1) *SemanticKITTI Dataset* [27]: Comprises 22 point clouds sequences, with sequences 0 to 10 (excluding sequence 8) designated as the training set and sequence 8 as the validation set. The training set includes 19 130 point clouds frames, and the validation set contains 4071 frames, amounting to over 4 billion points in total. These point clouds are annotated with 19 categories.

2) *nuScenes Dataset* [28]: Encompasses 1000 driving scenes, with 850 scenes for training and validation, and 150 for testing. In this dataset, nuScenes-lidarseg annotates 1.4 billion points with one of 32 possible semantic labels. Following the official protocol, the  $N_{\text{class}}$  is 17. Each sample contains the point clouds of 32 beams and the RGB images captured by six cameras: front, front-left, front-right, back, back-left, and back-right.

3) *Evaluation Metric*: We evaluate our method using the mIoU, which is the average IoU across all semantic categories. The IoU is the ratio of the intersection to the union of the predicted and actual values for a specific category, reflecting the consistency between the predicted results and the truth. An IoU value closer to 1 indicates a higher precision in the method's predictions for the corresponding category. Furthermore, the same as the official evaluation protocols for these datasets, which assess overall accuracy within the 50-m range, we also restrict our evaluation to this 50-m range, establishing it as a standard benchmark for comparison.

##### B. Implementation Details

In the proposed CSFNet, images are input to the network along with LiDAR point clouds only during the training phase, while the inference process deals solely with point clouds. It is important to note that the KITTI dataset employs a stereo front camera system, consisting of two color cameras with nearly overlapping fields of view. Following prior work [19], [20], for the SemanticKITTI dataset, we utilized only images from the left camera for model training. For the nuScenes dataset, images from six different viewpoints are simultaneously input into the network. During training, models are categorized as ours-small (four scales) and ours-large (six scales), depending on the number of different scale image features fused. The

epoch count is set to 120, and the model is updated using an SGD optimizer with a momentum of 0.9, an initial learning rate of 0.05, and a cosine annealing warm restarts learning rate adjustment strategy. All experiments were conducted on a single NVIDIA A100 GPU.

##### C. Comparative Study

1) *Quantitative Results on SemanticKITTI Dataset*: We compare our method with SOTA methods on SemanticKITTI in Table I. Ours-large notably achieves significantly higher mIoU scores, outperforming all LiDAR-only and fusion-based approaches. The mIoU of our method is just 0.8% lower than that of Point Transformer v3, which utilizes multidataset joint training (PTv3 + PPT [11]). In comparison with methods solely based on point clouds, such as SPVCNN [8] and FA-ResNet [58], ours-large registers mIoU improvements of 9.2% and 15.8%, respectively. This significant performance difference highlights the effectiveness of our fusion strategy in capturing spatial details and enhancing segmentation. Moreover, compared to leading fusion-based methods such as MSeg3D [20] and 2DPASS [19], our method shows mIoU enhancements of 2.5% and 1.4%. Particularly in key categories such as vehicle, pedestrian, and bicycle, our method achieves higher IoU scores. Notably, in the truck category, it achieves an impressive 96.5% mIoU, surpassing the best previous method by 5.6% and the second-best by 10.2%, thereby showcasing its exceptional performance in large-scale point clouds scene segmentation. These results strongly support the superiority of our proposed multiscale feature fusion strategy and SFM.

2) *Quantitative Results on nuScenes Dataset*: To further validate our method's effectiveness, we conducted a comparison with SOTA methods on the nuScenes dataset, as shown in Table II. It is shown that fusion-based methods generally outperform LiDAR-only methods. Among all fusion-based strategies, ours-small and ours-large consistently demonstrated superior performance across various categories. Remarkably, ours-large achieved an mIoU of 80.8%, leading all other fusion-based approaches. Moreover, mirroring its performance on the SemanticKITTI dataset, the segmentation accuracy of ours-large in crucial categories such as car, pedestrian, and traffic sign underscores its capability in managing challenging environments. Our method surpasses the latest PTv3 [11] model in mIoU performance. Notably, even when PTv3 benefits from multidataset joint training, our approach trails by 0.4% in mIoU.

3) *Performance on Car, Person, and Bicyclist*: To demonstrate the superiority of our method for long-range object segmentation, we compare the segmentation accuracies of car, person, and bicyclist at various distances. The data presented in Table III shows that our proposed CSFNet achieves high segmentation accuracies across different distance ranges. Notably, in the 40–50 m range, our method significantly outperforms the existing mainstream algorithms. Compared to SPVCNN [8], our method achieves improvements of 9.5%, 13.2%, and 11.7% for car, person, and bicyclist within 40–50 m, respectively, which is visually confirmed in Fig. 3. As the distance increases, the density of the LiDAR point cloud gradually decreases, leading to diminished

TABLE I

QUANTITATIVE RESULTS OF THE DIFFERENT METHODS ON THE VALIDATION SET OF SEMANTICKITTI. *L* INDICATES LiDAR-ONLY METHODS. LC INDICATES FUSION-BASED METHODS. THE BEST RESULT IS MARKED IN BOLD. EXPERIMENTAL DATA FOR MSEG3D [20], UNISEG [21], AND PTV3 [11] WAS OBTAINED FROM THE CORRESPONDING PAPERS

| Method                     | Modality | car         | bicycle     | motorcycle  | truck       | other vehicle | person      | bicyclist   | motorcyclist | road        | parking     | sidewalk    | other-ground | building    | fence       | vegetation  | trunk       | terrain     | pole        | traffic sign | mIoU(%)     |
|----------------------------|----------|-------------|-------------|-------------|-------------|---------------|-------------|-------------|--------------|-------------|-------------|-------------|--------------|-------------|-------------|-------------|-------------|-------------|-------------|--------------|-------------|
| RangeNet++ [36]            | L        | 89.4        | 26.5        | 48.4        | 33.9        | 26.7          | 54.8        | 69.4        | 0.0          | 92.9        | 37.0        | 69.9        | 0.0          | 83.4        | 51.0        | 83.3        | 54.0        | 68.1        | 49.8        | 34.0         | 51.2        |
| SqueezeSegV2 [38]          | L        | 82.7        | 15.1        | 22.7        | 25.6        | 26.9          | 22.9        | 44.5        | 0.0          | 92.7        | 39.7        | 70.7        | 0.1          | 71.6        | 37.0        | 74.6        | 35.8        | 68.1        | 21.8        | 22.2         | 40.8        |
| SqueezeSegV3 [39]          | L        | 87.1        | 34.3        | 48.6        | 47.5        | 47.1          | 58.1        | 53.8        | 0.0          | 95.3        | 43.1        | 78.2        | 0.3          | 78.9        | 53.2        | 82.3        | 55.5        | 70.4        | 46.3        | 33.2         | 53.3        |
| RandLaNet [32]             | L        | 92.0        | 8.0         | 12.8        | 74.8        | 46.7          | 52.3        | 46.0        | 0.0          | 93.4        | 32.7        | 73.4        | 0.1          | 84.0        | 43.5        | 83.7        | 57.3        | 73.1        | 48.0        | 27.3         | 50.0        |
| SalsaNext [40]             | L        | 90.5        | 44.6        | 49.6        | 86.3        | 54.6          | 74.0        | 81.4        | 0.0          | 93.4        | 40.6        | 69.1        | 0.0          | 84.6        | 53.0        | 83.6        | 64.3        | 64.2        | 54.4        | 39.8         | 59.4        |
| MinkowskiNet [57]          | L        | 95.0        | 23.9        | 50.4        | 55.3        | 45.9          | 65.6        | 82.2        | 0.0          | 94.3        | 43.7        | 76.4        | 0.0          | 87.9        | 57.6        | 87.4        | 67.7        | 71.5        | 63.5        | 43.6         | 58.5        |
| SPVCNN [8]                 | L        | 96.5        | 44.8        | 63.1        | 59.9        | 64.3          | 12.0        | 86.0        | 0.0          | 93.9        | 42.4        | 75.9        | 0.0          | 88.8        | 59.1        | 88.0        | 67.5        | 73.0        | 63.5        | 44.3         | 62.3        |
| Cylinder3D [14]            | L        | 96.4        | 61.5        | 78.2        | 66.3        | 69.8          | 80.8        | 93.3        | 0.0          | 94.9        | 41.5        | 78.0        | 1.4          | 87.5        | 50.0        | 86.7        | 72.2        | 68.8        | 63.0        | 42.1         | 64.9        |
| FA-ResNet [58]             | L        | 93.7        | 30.6        | 33.2        | 33.9        | 27.2          | 51.6        | 30.6        | <b>18.7</b>  | 91.3        | <b>65.2</b> | 75.5        | <b>26.0</b>  | 90.6        | 62.7        | 83.1        | 63.5        | 66.3        | 51.0        | 48.4         | 55.7        |
| PTv3 [11]                  | L        | -           | -           | -           | -           | -             | -           | -           | -            | -           | -           | -           | -            | -           | -           | -           | -           | -           | -           | -            | 70.8        |
| PTv3+PPT [11]              | L        | -           | -           | -           | -           | -             | -           | -           | -            | -           | -           | -           | -            | -           | -           | -           | -           | -           | -           | -            | <b>72.3</b> |
| PMF [18]                   | LC       | 95.4        | 47.8        | 62.9        | 68.4        | 75.2          | 78.9        | 71.6        | 0.0          | <b>96.4</b> | 43.5        | 80.5        | 0.1          | 88.7        | 60.1        | 88.6        | 72.7        | 75.3        | 65.5        | 43.0         | 63.9        |
| 2DPASS [19]                | LC       | 97.2        | 55.3        | 78.3        | 90.9        | 70.7          | 80.7        | 93.5        | 0.1          | 94.3        | 53.8        | 82.0        | 12.1         | 93.0        | 71.8        | 89.1        | 72.3        | 75.7        | 66.0        | 54.2         | 70.1        |
| MSeg3D [20]                | LC       | -           | -           | -           | -           | -             | -           | -           | -            | -           | -           | -           | -            | -           | -           | -           | -           | -           | -           | -            | 69.0        |
| UniSeg [21]                | LC       | -           | -           | -           | -           | -             | -           | -           | -            | -           | -           | -           | -            | -           | -           | -           | -           | -           | -           | -            | 71.3        |
| <b>CSFNet (Ours-small)</b> | LC       | 97.0        | <b>62.5</b> | 86.3        | 92.6        | 75.4          | <b>83.2</b> | 92.9        | 1.6          | 94.5        | 49.7        | 82.7        | 6.0          | 92.4        | 69.4        | <b>90.0</b> | 71.8        | <b>78.6</b> | <b>66.7</b> | <b>54.1</b>  | 70.9        |
| <b>CSFNet (Ours-large)</b> | LC       | <b>97.6</b> | 61.5        | <b>88.5</b> | <b>96.5</b> | <b>82.8</b>   | 82.4        | <b>93.6</b> | 2.2          | 94.8        | 52.1        | <b>83.0</b> | 0.0          | <b>93.4</b> | <b>73.7</b> | 89.1        | <b>72.8</b> | 75.6        | 66.3        | 52.3         | 71.5        |

TABLE II

QUANTITATIVE RESULTS OF THE DIFFERENT METHODS ON THE VALIDATION SET OF nuSCENES. *L* INDICATES LiDAR-ONLY METHODS. LC INDICATES FUSION-BASED METHODS. THE BEST RESULT IS MARKED IN BOLD. EXPERIMENTAL DATA FOR MEMORYSEG [34] AND PTV3 [11] WAS OBTAINED FROM THE CORRESPONDING PAPERS

| Method                     | Modality | barrier     | bicycle     | bus         | car         | construction | motorcycle  | pedestrian  | traffic cone | trailer     | truck       | driveable   | other flat  | sidewalk    | terrain     | manmade     | vegetation  | mIoU(%)     |
|----------------------------|----------|-------------|-------------|-------------|-------------|--------------|-------------|-------------|--------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| RangeNet++ [36]            | L        | 66.0        | 21.3        | 77.2        | 80.9        | 30.2         | 66.8        | 69.6        | 52.1         | 54.2        | 72.3        | 94.1        | 66.6        | 63.5        | 70.1        | 83.1        | 79.8        | 65.5        |
| PolarNet [59]              | L        | 74.7        | 28.2        | 85.3        | 90.9        | 35.1         | 77.5        | 71.3        | 58.8         | 57.4        | 76.1        | 96.5        | 71.1        | 74.7        | 74.0        | 87.3        | 85.7        | 71.0        |
| SalsaNext [40]             | L        | 74.8        | 34.1        | 85.9        | 88.4        | 42.2         | 72.4        | 72.2        | 63.1         | 61.3        | 76.5        | 96.0        | 70.8        | 71.2        | 71.5        | 86.7        | 84.4        | 72.2        |
| Cylinder3D [14]            | L        | 76.4        | 40.3        | 91.3        | 93.8        | 51.3         | 78.0        | 78.9        | 64.9         | 62.1        | 84.4        | 96.8        | 71.6        | <b>76.4</b> | 75.4        | <b>90.5</b> | 87.4        | 76.1        |
| SFCNet [60]                | L        | 76.7        | 40.4        | 89.5        | 91.3        | 46.7         | 82.0        | 78.1        | 65.8         | 69.4        | 80.6        | 96.6        | 71.6        | 74.5        | 74.9        | 89.0        | 87.5        | 75.9        |
| RangeVit [13]              | L        | 75.5        | 40.7        | 88.3        | 90.1        | 49.3         | 79.3        | 77.2        | 66.3         | 65.2        | 80.0        | 96.4        | 71.4        | 73.8        | 73.8        | 89.9        | 87.2        | 75.2        |
| MemorySeg [34]             | L        | -           | -           | -           | -           | -            | -           | -           | -            | -           | -           | -           | -           | -           | -           | -           | -           | 76.7        |
| PMF [18]                   | LC       | 74.1        | 46.6        | 89.8        | 92.1        | 57.0         | 77.7        | 80.9        | 70.9         | 64.6        | 82.9        | 95.5        | 73.3        | 73.6        | 74.8        | 89.4        | <b>87.7</b> | 76.9        |
| 2DPASS [19]                | LC       | <b>78.4</b> | 52.5        | 95.4        | 93.4        | 62.5         | 88.9        | <b>83.1</b> | 68.1         | 75.6        | <b>89.1</b> | <b>96.9</b> | 75.5        | 76.3        | 75.7        | 89.1        | 86.9        | 80.5        |
| PTv3 [11]                  | L        | -           | -           | -           | -           | -            | -           | -           | -            | -           | -           | -           | -           | -           | -           | -           | -           | 80.4        |
| PTv3+PPT [11]              | L        | -           | -           | -           | -           | -            | -           | -           | -            | -           | -           | -           | -           | -           | -           | -           | -           | <b>81.2</b> |
| <b>CSFNet (Ours-small)</b> | LC       | 78.3        | 52.0        | 95.9        | 92.7        | 60.1         | 88.7        | <b>83.1</b> | <b>68.8</b>  | 76.6        | 85.8        | 96.8        | 73.8        | 75.8        | 75.4        | 89.3        | 87.0        | 80.0        |
| <b>CSFNet (Ours-large)</b> | LC       | 77.8        | <b>52.6</b> | <b>96.6</b> | <b>94.4</b> | <b>64.1</b>  | <b>89.3</b> | 82.9        | 67.9         | <b>77.0</b> | 87.7        | 96.8        | <b>76.6</b> | 76.2        | <b>76.1</b> | 89.3        | 86.7        | 80.8        |

TABLE III

COMPARISON OF SEGMENTATION ACCURACY (mIoU, %) FOR CAR, PERSON, AND BICYCLIST ACROSS VARIOUS DISTANCE RANGES USING DIFFERENT METHODS.  $\Delta_{40-50}$  INDICATES THE RELATIVE PERFORMANCE OF THESE METHODS COMPARED TO SPVCNN AT A DISTANCE OF 40–50 M

| Method            | Car         |             |             |             |             |                  | Person      |             |             |             |             |                  | Bicyclist   |             |             |             |             |                  |
|-------------------|-------------|-------------|-------------|-------------|-------------|------------------|-------------|-------------|-------------|-------------|-------------|------------------|-------------|-------------|-------------|-------------|-------------|------------------|
|                   | 0-10        | 10-20       | 20-30       | 30-40       | 40-50       | $\Delta_{40-50}$ | 0-10        | 10-20       | 20-30       | 30-40       | 40-50       | $\Delta_{40-50}$ | 0-10        | 10-20       | 20-30       | 30-40       | 40-50       | $\Delta_{40-50}$ |
| SPVCNN [8]        | 97.6        | 96.8        | 93.9        | 83.6        | 66.4        | +0.0             | 80.6        | 67.1        | 59.7        | 49.9        | 24.3        | +0.0             | 93.3        | 94.8        | 72.2        | 69.9        | 47.9        | +0.0             |
| MinkowskiNet [57] | 97.7        | 96.3        | 93.9        | 83.4        | 65.8        | -0.6             | 81.1        | 65.9        | 58.4        | 41.7        | 12.4        | -11.9            | 91.7        | 93.8        | 72.5        | 67.0        | 53.9        | +6.0             |
| 2DPASS [19]       | 97.5        | <b>97.5</b> | <b>95.5</b> | <b>86.1</b> | 75.1        | +8.7             | 87.9        | 75.7        | <b>72.7</b> | 64.5        | 36.6        | +12.3            | 95.4        | <b>95.3</b> | <b>80.8</b> | <b>76.4</b> | 57.4        | +10.4            |
| <b>CSFNet</b>     | <b>98.2</b> | 97.4        | 94.9        | 85.4        | <b>75.9</b> | +9.5             | <b>88.3</b> | <b>79.4</b> | 72.2        | <b>66.3</b> | <b>37.5</b> | +13.2            | <b>96.6</b> | 95.2        | 76.3        | 69.3        | <b>59.6</b> | +11.7            |

feature discriminability, which in turn affects the model's ability to segment objects accurately at longer distances. As shown in Fig. 4, current SOTA algorithms, including MinkowskiNet [57], SPVCNN [8], and 2DPASS [19],

TABLE IV

COMPARISON OF SEGMENTATION ACCURACY (mIoU, %) FOR ROAD, PARK, AND SIDEWALK ACROSS VARIOUS DISTANCE RANGES USING DIFFERENT METHODS.  $\Delta_{40-50}$  INDICATES THE RELATIVE PERFORMANCE OF THESE METHODS COMPARED TO SPVCNN AT A DISTANCE OF 40–50 M

| Method            | Road        |             |             |             |             |                  | Park        |             |             |             |             |                  | Sidewalk    |             |             |             |             |                  |
|-------------------|-------------|-------------|-------------|-------------|-------------|------------------|-------------|-------------|-------------|-------------|-------------|------------------|-------------|-------------|-------------|-------------|-------------|------------------|
|                   | 0-10        | 10-20       | 20-30       | 30-40       | 40-50       | $\Delta_{40-50}$ | 0-10        | 10-20       | 20-30       | 30-40       | 40-50       | $\Delta_{40-50}$ | 0-10        | 10-20       | 20-30       | 30-40       | 40-50       | $\Delta_{40-50}$ |
| SPVCNN [8]        | 94.0        | 93.0        | 87.3        | 79.0        | 66.9        | +0.0             | 48.1        | 54.6        | 41.1        | 25.3        | 14.2        | +0.0             | 83.4        | 75.5        | 62.0        | 47.6        | 34.0        | +0.0             |
| MinkowskiNet [57] | 94.7        | 93.2        | 88.9        | 80.5        | 66.4        | -0.5             | 52.1        | 58.6        | 45.7        | 26.6        | 16.7        | +2.5             | 84.3        | 75.6        | 63.6        | 49.3        | 35.4        | +1.4             |
| 2DPASS [19]       | 94.9        | 94.7        | 89.7        | 83.5        | 74.7        | +7.8             | <b>53.1</b> | <b>61.3</b> | 49.2        | 29.6        | 19.6        | +5.4             | 84.5        | 78.3        | 66.2        | 54.2        | 40.5        | +6.5             |
| <b>CSFNet</b>     | <b>95.6</b> | <b>94.8</b> | <b>91.0</b> | <b>86.8</b> | <b>79.3</b> | <b>+12.4</b>     | 49.8        | 61.1        | <b>56.2</b> | <b>38.4</b> | <b>21.8</b> | <b>+7.6</b>      | <b>85.3</b> | <b>78.7</b> | <b>70.3</b> | <b>60.8</b> | <b>48.7</b> | <b>+14.7</b>     |

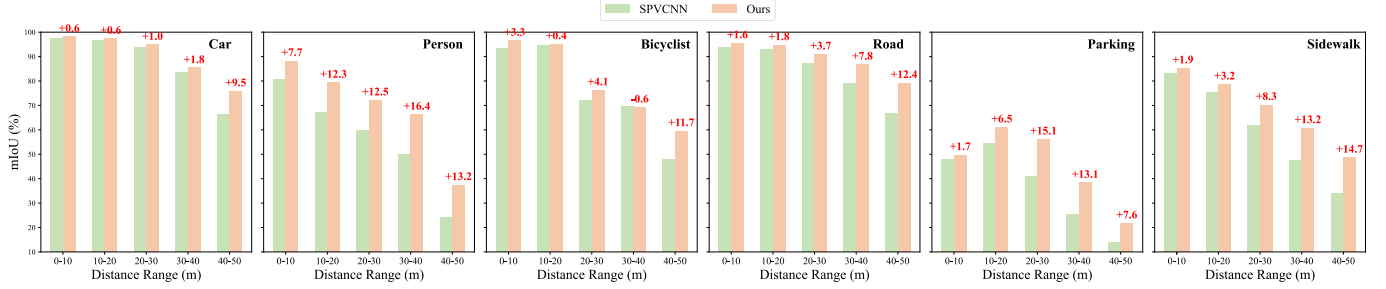


Fig. 3. Visualizing the segmentation accuracy of our method and SPVCNN at different distances for different classes. The numbers on the columns indicate the performance improvement of our method over SPVCNN.

TABLE V

COMPARISONS BETWEEN EFFICIENCY (RUN-TIME) AND ACCURACY (mIoU) ON THE SEMANTICKITTI VALIDATION SET. THE RESULTS OF THE OTHER METHODS ARE FROM UNISEG [21]

| Method                     | # Param(M) ↓ | Latency(ms) ↓ | mIoU(%) ↑   |
|----------------------------|--------------|---------------|-------------|
| MinkowskiNet [57]          | 21.7         | <b>48.4</b>   | 61.1        |
| Cylinder3D [14]            | 56.3         | 75.1          | 65.9        |
| SPVCNN [8]                 | 21.8         | 52.4          | 63.8        |
| UniSeg 0.2× [21]           | 28.8         | 84.6          | 67.0        |
| UniSeg 1.0× [21]           | 147.6        | 145.0         | 71.3        |
| <b>CSFNet (Ours-small)</b> | <b>7.6</b>   | 58.9          | 70.9        |
| <b>CSFNet (Ours-large)</b> | 45.4         | 122.1         | <b>71.5</b> |

TABLE VI

ABLATION STUDY CONDUCTED ON THE SEMANTICKITTI VALIDATION SET, WITH THE BEST RESULT MARKED IN BOLD

| ID  | Baseline | MDF | SFM | mIoU(%)     |
|-----|----------|-----|-----|-------------|
| I   | ✓        |     |     | 67.8        |
| II  | ✓        | ✓   |     | 70.8        |
| III | ✓        | ✓   | ✓   | <b>71.5</b> |

TABLE VII

VALIDATION OF OUR METHOD USING MINKOWSKINET ON SEMANTICKITTI VALIDATION SET: COMPARISON OF mIoU (%) ACROSS DIFFERENT DISTANCE RANGES, HIGHLIGHTING THE BEST RESULT IN BOLD

| Method                     | Distance Range (m) |             |             |             |             |
|----------------------------|--------------------|-------------|-------------|-------------|-------------|
|                            | 0-10               | 10-20       | 20-30       | 30-40       | 40-50       |
| MinkowskiNet [57]          | 66.0               | 63.3        | 59.9        | 48.9        | 37.0        |
| <b>MinkowskiNet + ours</b> | <b>69.2</b>        | <b>66.2</b> | <b>64.0</b> | <b>52.7</b> | <b>41.5</b> |

misclassify points from the “people” category as “vegetation” or “build.” Additionally, these methods exhibit significant errors when segmenting car and truck at greater distances.

In contrast, our proposed method achieves more accurate recognition of these categories. As shown in Fig. 4, the segmentation results of our method closely align with the ground truth, showcasing the effectiveness of our approach in enhancing the discriminative power of distant point cloud features.

4) *Performance on Road, Parking, and Sidewalk:* From Table IV, it is evident that CSFNet achieves the highest segmentation accuracy for road and sidewalk across all distance ranges. The segmentation accuracy of road is still as high as 79.3% at 40–50 m, which is a 12.4% improvement compared to SPVCNN. Although the accuracy of our method is slightly lower than 2DPASS for the park class in the 0–20 m range, as the distance increases, our method outperforms 2DPASS, maintaining superior segmentation performance. The bar chart in Fig. 3 clearly shows the superiority of our method in distinguishing objects with similar structures over all distance ranges. We also demonstrate the power of CSFNet through visualization. As shown in Fig. 5, MinkowskiNet, SPVCNN, and 2DPASS misclassify vegetation as fence, sidewalk as road, and park as sidewalk, while also misclassifying fence as build. The corresponding error maps clearly demonstrate that our method outperforms the current approaches.

5) *Overall Performance Comparison by Distance Range:* We compare the overall performance of SPVCNN [8], MinkowskiNet [57], 2DPASS [19], and our method across five different distance ranges on the SemanticKITTI dataset. According to the official evaluation protocol, the five distance ranges are 0–10 m, 10–20 m, 20–30 m, 30–40 m, and 40–50 m. As shown in Fig. 6, our method consistently outperforms the other three methods across all specified distance intervals. As expected, performance decreases with increasing distance—a common trend observed in all methods. However, in the 40–50 m range, our method achieves an mIoU of 45.5%, while SPVCNN [8] and MinkowskiNet [57] achieve mIoUs of

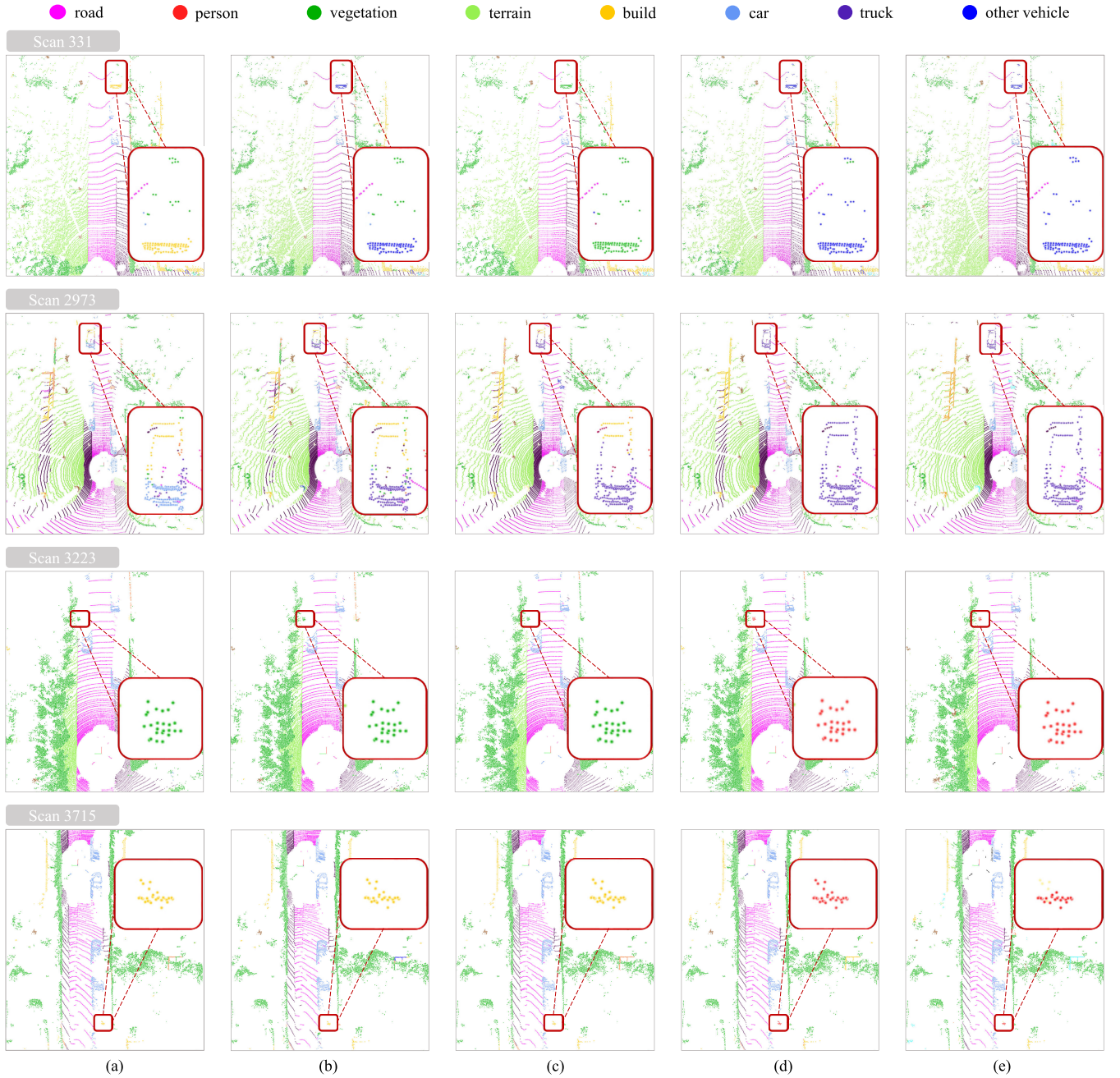


Fig. 4. Visualization of long-distance object segmentation performance on the SemanticKITTI validation set. (a) MinkowskiNet. (b) SPVCNN. (c) 2DPASS. (d) Ours. (e) Ground Truth.

37.9% and 37.0%, respectively, representing a 7.6% and 8.5% improvement over these methods.

#### D. Comparisons of Efficiency and Accuracy

Table V reports the efficiency and accuracy results of our proposed method compared to SOTA baseline methods. Ours-small offers faster inference speed while ours-large is more accurate. Compared to point clouds-only methods such as MinkowskiNet [57], Cylinder3D [14], and SPVCNN [8], ours-small not only demonstrates competitive speed but also shows mIoU improvements of 9.8%, 5.0%, and 7.1%. On the other hand, UniSeg [21] employs a fusion strategy of point

clouds and image deep features, achieving a 71.3% mIoU but at the cost of increased parameter count and prolonged inference time. Our method, utilizing image prior information only during training, facilitates prior-guided semantic focus learning across modalities. Consequently, ours-small attains comparable accuracy to UniSeg1.0 $\times$  while being 2.4 times faster in inference. The computational advantage stems from the fact that, during inference, only point cloud data is used as input, which eliminates the need for additional modalities and reduces the computational burden. Moreover, our approach avoids the time-consuming fusion and alignment operations typically required during inference, resulting in faster processing times while preserving high accuracy.

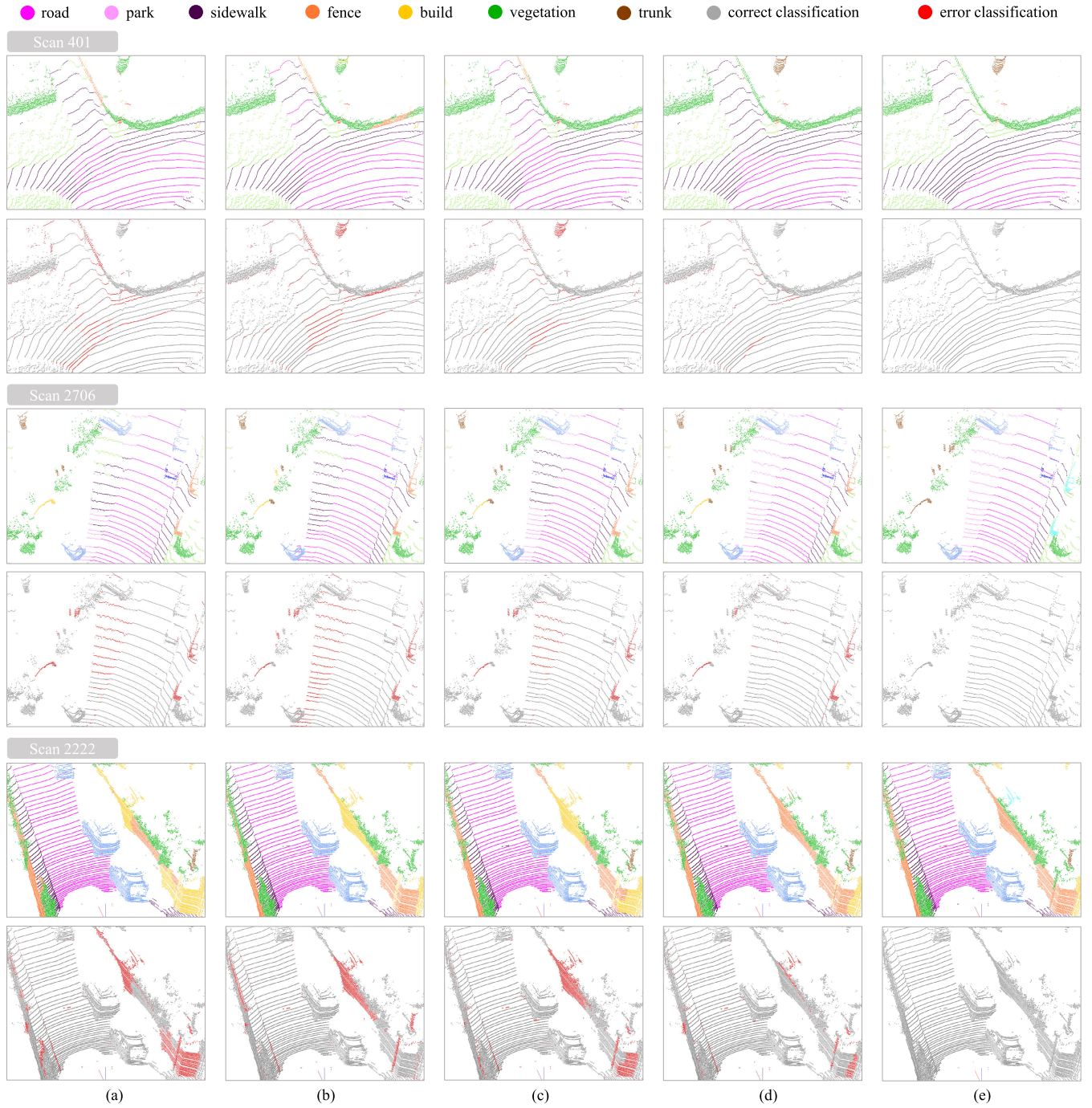


Fig. 5. Visualization of object segmentation performance for similar geometric structures on the SemanticKITTI validation set. (a) MinkowskiNet. (b) SPVCNN. (c) 2DPASS. (d) Ours. (e) Ground Truth. For each of the three scenes, the first row presents segmentation results, and the second row displays the corresponding error maps. Correctly predicted points are shown in gray, while incorrectly predicted points are in red.

### E. Ablation Study

We conducted extensive ablation studies on the SemanticKITTI dataset to validate the effectiveness of each module in our method. We established a baseline model that only inputs point cloud data, achieving an mIoU of 67.8% with the aid of test-time augmentation (TTA), as detailed in Experiment I (see Table VI). Comparing Experiment II with Experiment I, the inclusion of our proposed MDF led to a 3% increase in mIoU. Following this, the implementation

of SFM in Experiment III resulted in an additional 0.7% enhancement in model precision.

We further demonstrate the effectiveness of the two proposed modules by visualizing their performance in comparison. As shown in the first row of Fig. 7 (highlighted within the white box), point clouds in distant regions exhibit reduced density, and the fusion strategy employed by 2DPASS struggles to effectively delineate object boundaries, resulting in noticeable noise in the confidence map. In contrast, our method

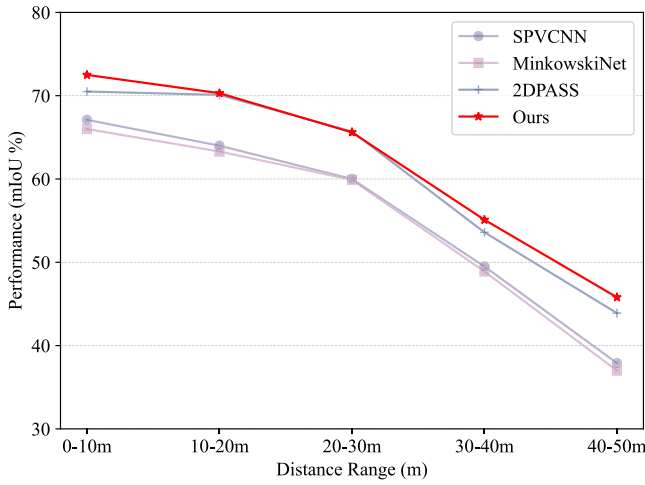


Fig. 6. Overall performance comparison by distance range on SemanticKITTI dataset. The x-axis represents the range of distances, while the y-axis indicates the average mIoU scores within the corresponding distance ranges.

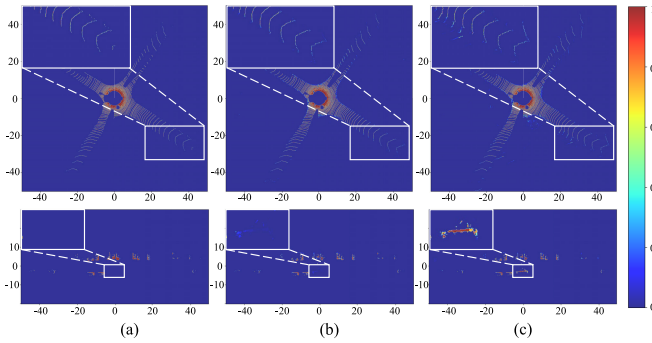


Fig. 7. Bird's eye view confidence maps comparing the fusion strategy of 2DPASS and our proposed MDF module on the SemanticKITTI dataset. Red indicates a segmentation confidence of 1, while blue represents 0. The color map is shown on the right. The horizontal and vertical axes represent spatial coordinates in meters. (a) Ground truth. (b) Ours. (c) 2DPASS.

achieves higher confidence and fewer misclassifications in these regions. Additionally, in the second row (within the white box), our approach significantly enhances the recognition accuracy for objects with similar 3-D features, such as cars and trucks, thereby demonstrating the effectiveness of our MDF module in distinguishing geometrically similar objects. To further validate the performance of the SFM, we present error maps that visually compare the results of models with and without the SFM (see Fig. 8). These error maps clearly highlight the improved delineation of object boundaries and a reduction in segmentation errors, particularly in challenging regions with complex class boundaries.

Moreover, we evaluate the integration of our method with MinkowskiNet, with experimental results summarized in Table VII. MinkowskiNet [57] shows a 4.5% improvement in segmentation accuracy for the 40–50 m distance range by incorporating our module during training, with the mIoU scores also improving across all other distance ranges.

## V. CHALLENGES AND LIMITATIONS

### A. Distance Range Limitation

The performance of LiDAR point cloud semantic segmentation at long distances is influenced by multiple factors,

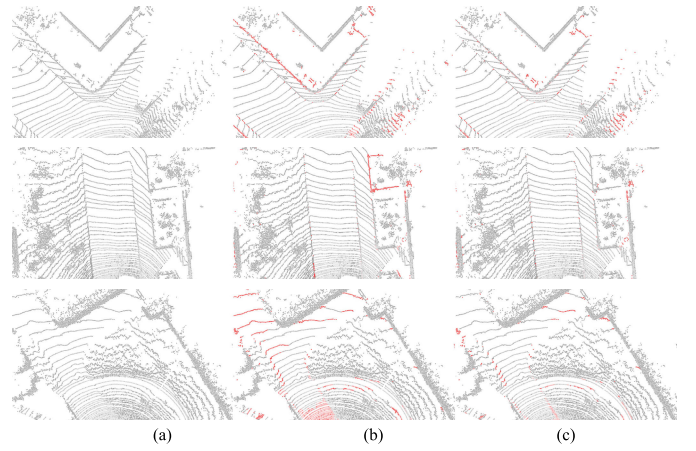


Fig. 8. Error maps comparing segmentation results of models with and without the SFM. Correctly predicted points are shown in gray, while incorrectly predicted points are in red. These maps illustrate the SFM's impact on reducing errors, particularly at boundary regions. (a) Ground truth. (b) w/o SFM. (c) w/ SFM.

including point cloud density, noise, and the complexity of the segmentation task. As discussed in the introduction of this article, the feature discriminability of LiDAR point clouds diminishes at longer ranges due to lower point density, hindering the accurate classification of distant objects. This challenge is particularly pronounced in complex urban environments, where subtle differences between similar objects (e.g., sidewalks and roads, or different types of vehicles) are crucial for precise semantic segmentation. To address this issue, our method leverages multiscale image features to enhance discriminative power and improve semantic classification by strengthening the ability to distinguish between different semantic categories in the semantic feature space.

In this study, however, we focused our evaluation on distances up to 50 m, primarily due to the characteristics of the datasets used (e.g., SemanticKITTI and nuScenes) and their predefined evaluation protocols. While this constraint provided a well-defined framework for our analysis, we recognize the importance of extending the range of distances in future work. To that end, we plan to incorporate additional datasets with extended distance ranges and further refine our algorithmic approach to handle sparser and less reliable point clouds at greater distances. Such an extension will enhance the robustness and applicability of our method in real-world scenarios, such as large-scale 3-D urban mapping and autonomous driving, where accurate segmentation of distant objects is crucial. Expanding the distance range, we aim to improve the generalizability of our approach while maintaining high segmentation accuracy in urban environments with a diverse array of object types.

### B. Calibration Dependency Challenges

One significant limitation of CSFNet is its reliance on the calibration relationship between images and point clouds. As we focused our analysis on datasets such as SemanticKITTI and nuScenes, which provide this calibration, our method may not perform effectively on other multimodal datasets lacking such relationships. This dependence restricts the

generalizability of our findings and suggests that further exploration is necessary to adapt our framework for datasets without direct calibration. Our future work will aim to address this limitation by investigating alternative techniques that could enable effective fusion of point clouds and images without requiring precise calibration.

## VI. CONCLUSION

In this article, we present a CSFNet for semantic segmentation of large-scale point clouds. This network is particularly effective in segmenting distant objects and those exhibiting similar geometric features. Extensive experiments conducted on two publicly available large-scale urban LiDAR point cloud datasets demonstrate that CSFNet achieves SOTA performance. By effectively addressing the challenges posed by long-range distance and data sparsity, our method provides significant advancements in remote sensing tasks, with promising applications in Earth observation and other critical fields. Moreover, our approach utilizes complementary information from both point clouds and images, offering a novel framework for integrating multimodal data in large-scale remote sensing applications.

Despite the substantial improvement in segmentation accuracy for long-distance objects, some categories still exhibit segmentation accuracies below 50% when distances exceed 40 m. Future work will focus on further investigating the impact of distance on segmentation performance and overcoming the inherent limitations of point cloud segmentation accuracy, with the aim of advancing intelligent point cloud processing for remote sensing applications.

## ACKNOWLEDGMENT

The authors would like to thank the editor and the anonymous reviewers for their careful, serious, and valuable comments that helped improve this article.

## REFERENCES

- [1] M. Dai et al., "Multiprototype relational network for few-shot ALS point cloud semantic segmentation by transferring knowledge from photogrammetric point clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–17, 2024.
- [2] J. Shao, W. Yao, P. Wang, Z. He, and L. Luo, "Urban GeoBIM construction by integrating semantic LiDAR point clouds with as-designed BIM models," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–12, 2024.
- [3] Z. Wang et al., "Deep learning-based semantic segmentation and surface reconstruction for point clouds of offshore oil production equipment," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–18, 2024.
- [4] M. Jindal, A. Jha, and L. R. Cenkeramaddi, "Bollard segmentation and position estimation from LiDAR point cloud for autonomous mooring," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–9, 2022.
- [5] C. Liu et al., "Context-aware network for semantic segmentation toward large-scale point clouds in urban environments," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5703915.
- [6] Y. Zang et al., "LCE-NET: Contour extraction for large-scale 3-D point clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–13, 2023.
- [7] Z. Luo, Z. Zeng, W. Tang, J. Wan, Z. Xie, and Y. Xu, "Dense dual-branch cross attention network for semantic segmentation of large-scale point clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–16, 2024.
- [8] H. Tang et al., "Searching efficient 3D architectures with sparse point-voxel convolution," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 685–702.
- [9] H. Zhao, L. Jiang, J. Jia, P. Torr, and V. Koltun, "Point transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 16259–16268.
- [10] X. Wu, Y. Lao, L. Jiang, X. Liu, and H. Zhao, "Point transformer v2: Grouped vector attention and partition-based pooling," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 35, Jan. 2022, pp. 33330–33342.
- [11] X. Wu et al., "Point transformer v3: Simpler, faster, stronger," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 4840–4851.
- [12] L. Kong et al., "Rethinking range view representation for LiDAR segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 228–240.
- [13] A. Ando, S. Gidaris, A. Bursuc, G. Puy, A. Boulch, and R. Marlet, "RangeViT: Towards vision transformers for 3D semantic segmentation in autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 5240–5250.
- [14] X. Zhu et al., "Cylindrical and asymmetrical 3D convolution networks for LiDAR segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9939–9948.
- [15] J. Li, H. Dai, and Y. Ding, "Self-distillation for robust LiDAR semantic segmentation in autonomous driving," in *Proc. Eur. Conf. Comp. Vis. (ECCV)*, Jan. 2022, pp. 659–676.
- [16] J. Xu, R. Zhang, J. Dou, Y. Zhu, J. Sun, and S. Pu, "RPVNet: A deep and efficient range-point-voxel fusion network for LiDAR point cloud segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 16024–16033.
- [17] B. Graham, M. Engelcke, and L. van der Maaten, "3D semantic segmentation with submanifold sparse convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9224–9232.
- [18] Z. Zhuang, R. Li, K. Jia, Q. Wang, Y. Li, and M. Tan, "Perception-aware multi-sensor fusion for 3D LiDAR semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 16280–16290.
- [19] X. Yan et al., "2DPASS: 2D priors assisted semantic segmentation on LiDAR point clouds," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 677–695.
- [20] J. Li, H. Dai, H. Han, and Y. Ding, "MSeg3D: Multi-modal 3D semantic segmentation for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 21694–21704.
- [21] Y. Liu et al., "UniSeg: A unified multi-modal LiDAR segmentation network and the OpenPCSeg codebase," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 21662–21673.
- [22] Y. Li et al., "Deepfusion: LiDAR-camera deep fusion for multi-modal 3D object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2022, pp. 17182–17191.
- [23] Y. Jiao, Z. Jie, S. Chen, J. Chen, L. Ma, and Y.-G. Jiang, "MSMDFusion: Fusing LiDAR and camera at multiple scales with multi-depth seeds for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 21643–21652.
- [24] J. Yan et al., "Cross modal transformer: Towards fast and robust 3D object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 18268–18278.
- [25] X. Bai et al., "TransFusion: Robust LiDAR-camera fusion for 3D object detection with transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 1090–1099.
- [26] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, vol. 139, 2021, pp. 8748–8763.
- [27] J. Behley et al., "SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9297–9307.
- [28] H. Caesar et al., "nuScenes: A multimodal dataset for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11621–11631.
- [29] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 652–660.
- [30] C. R. Qi, Y. Li, H. Su, and L. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 30, Jan. 2017.
- [31] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegeui, F. Goulette, and L. J. Guibas, "KPConv: Flexible and deformable convolution for point clouds," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6411–6420.

- [32] Q. Hu et al., "RandLA-Net: Efficient semantic segmentation of large-scale point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11108–11117.
- [33] Y. Xu et al., "NeiEA-NET: Semantic segmentation of large-scale point cloud scene via neighbor enhancement and aggregation," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 119, May 2023, Art. no. 103285.
- [34] E. Li, S. Casas, and R. Urtasun, "MemorySeg: Online LiDAR semantic segmentation with a latent memory," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 745–754.
- [35] T. Han et al., "Point cloud semantic segmentation with adaptive spatial structure graph transformer," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 133, Sep. 2024, Art. no. 104105.
- [36] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss, "RangeNet++: Fast and accurate LiDAR semantic segmentation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 4213–4220.
- [37] B. Wu, A. Wan, X. Yue, and K. Keutzer, "SqueezeSeg: Convolutional neural nets with recurrent CRF for real-time road-object segmentation from 3D LiDAR point cloud," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 1887–1893.
- [38] B. Wu, X. Zhou, S. Zhao, X. Yue, and K. Keutzer, "SqueezeSegV2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a LiDAR point cloud," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 4376–4382.
- [39] C. Xu et al., "SqueezeSegv3: Spatially-adaptive convolution for efficient point-cloud segmentation," in *Proc. 16th Eur. Conf. Comput. Vis. (ECCV)*, Aug. 2020, pp. 1–19.
- [40] T. Cortinhal, G. Tzelepis, and E. E. Aksoy, "SalsaNext: Fast, uncertainty-aware semantic segmentation of LiDAR point clouds," in *Advances in Visual Computing*, Jan. 2020, pp. 207–222.
- [41] R. Cheng, R. Razani, E. Taghavi, E. Li, and B. Liu, "(AF)<sup>2</sup>-S3Net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 12542–12551.
- [42] B. Peng et al., "OA-CNNs: Omni-adaptive sparse CNNs for 3D semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 21305–21315.
- [43] G. Krissel, M. Opitz, G. Waltner, H. Possegger, and H. Bischof, "FuseSeg: LiDAR point cloud segmentation fusing multi-modal data," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2020, pp. 1863–1872.
- [44] D. Peng, Y. Lei, W. Li, P. Zhang, and Y. Guo, "Sparse-to-dense feature matching: Intra and inter domain cross-modal learning in domain adaptation for 3D semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 7088–7097.
- [45] Z. Guo, R. Xu, C.-C. Feng, and Z. Zeng, "PIF-Net: A deep point-image fusion network for multimodality semantic segmentation of very high-resolution imagery and aerial point cloud," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 11270–11282, 2024.
- [46] Y. Liu et al., "Segment any point cloud sequences by distilling vision foundation models," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 36, Jan. 2023, pp. 37193–37229.
- [47] S. Xie, J. Gu, D. Guo, C. R. Qi, L. Guibas, and O. Litany, "PointContrast: Unsupervised pre-training for 3D point cloud understanding," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 574–591.
- [48] L. Tang, Y. Zhan, Z. Chen, B. Yu, and D. Tao, "Contrastive boundary learning for point cloud segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 8479–8489.
- [49] M. Afham, I. Dissanayake, D. Dissanayake, A. Dharmasiri, K. Thilakarathna, and R. Rodrigo, "CrossPoint: Self-supervised cross-modal contrastive learning for 3D point cloud understanding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 9902–9912.
- [50] S. Peng, K. Genova, C. Jiang, A. Tagliasacchi, M. Pollefeys, and T. Funkhouser, "OpenScene: 3D scene understanding with open vocabularies," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 815–824.
- [51] Q. Xia et al., "CoIn: Contrastive instance feature mining for outdoor 3D object detection with very limited annotations," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 6231–6240.
- [52] R. Chen et al., "CLIP2Scene: Towards label-efficient 3D scene understanding by CLIP," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 7020–7030.
- [53] X. Wu et al., "Towards large-scale 3D representation learning with multi-dataset point prompt training," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 19551–19562.
- [54] Y. Liu et al., "Multi-space alignments towards universal LiDAR segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 14648–14661.
- [55] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
- [56] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9729–9738.
- [57] C. Choy, J. Gwak, and S. Savarese, "4D spatio-temporal ConvNets: Minkowski convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3075–3084.
- [58] L. Zhan, W. Li, and W. Min, "FA-ResNet: Feature affine residual network for large-scale point cloud segmentation," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 118, Apr. 2023, Art. no. 103259.
- [59] Y. Zhang et al., "PolarNet: An improved grid representation for online LiDAR point clouds semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9601–9610.
- [60] Y. Zheng, G. Wang, J. Liu, M. Pollefeys, and H. Wang, "Spherical frustum sparse convolution network for LiDAR point cloud semantic segmentation," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Jan. 2024, pp. 121827–121858.



**Yang Luo** received the B.S. degree from the Department of Electronic and Optical Engineering, Nanjing University of Posts and Telecommunications, Nanjing, China, in 2022. He is currently pursuing the M.S. degree in computer engineering with Jimei University, Xiamen, China.

His research interests include semantic segmentation, image processing, and its application in remote sensing and autonomous driving.



**Ting Han** (Graduate Student Member, IEEE) is currently pursuing the Ph.D. degree with the School of Geospatial Engineering and Science, Sun Yat-sen University, Zhuhai, Guangdong, China.

He has published several articles in top journals including IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, *ISPRS Journal of Photogrammetry and Remote Sensing*, and *International Journal of Applied Earth Observation and Geoinformation*. His research interests

include image and point cloud processing, semantic segmentation, 3-D reconstruction, deep learning theory in 3-D vision, and its applications in remote sensing and GeoAI.



**Yujun Liu** received the M.S. degree in information and communication engineering from Jimei University, Xiamen, China, in 2024. He is currently pursuing the Ph.D. degree with the School of Architecture and Urban Planning, Shenzhen University, Shenzhen, China.

His research interests include image and point cloud processing, building reconstruction, stereo matching, and deep learning.



**Jinhe Su** received the Ph.D. degree from the University of Chinese Academy of Sciences, Beijing, China, in 2018.

He is currently an Associate Professor with the School of Computer Engineering, Jimei University, Xiamen, Fujian, China. His main research interests include machine learning, computer vision, remote sensing data processing, image processing, and virtual reality.



**Yundong Wu** received the Ph.D. degree from Information Engineering University of the People's Liberation Army, Zhengzhou, Henan, China, in 2000.

From 2004 to 2008, he was a Professor with the Department of Remote Sensing Information Engineering, Information Engineering University of the People's Liberation Army. He is currently a Professor with the College of Computer Engineering, Jimei University, Xiamen, Fujian, China. His main research directions are terms of photogrammetry and

remote sensing, and LiDAR data analysis 3-D reconstruction.



**Yiping Chen** (Senior Member, IEEE) received the Ph.D. degree in information communication engineering from the National University of Defense Technology, Changsha, China, in 2011.

From 2007 to 2011, she was an Assistant Researcher with The Chinese University of Hong Kong, Hong Kong. She is currently an Associate Professor with the School of Geospatial Engineering and Science, Sun Yat-sen University, Zhuhai, China. Her main research interests include image processing, mobile LiDAR data analysis, and GeoAI.

Dr. Chen is an Associate Editor of IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING and the Co-Chair of the ISPRS WG I/4 on LiDAR, Laser Altimetry and Sensor Integration (2022–2026).



**Jinyuan Li** received the B.S. degree in software engineering from Jimei University, Xiamen, China, in 2023, where he is currently pursuing the M.S. degree in software engineering.

His research interests include semantic segmentation, image processing, and its application in autonomous driving.



**Guorong Cai** (Senior Member, IEEE) received the Ph.D. degree from Xiamen University, Xiamen, Fujian, China, in 2013.

He is currently a Professor with the College of Computer Engineering, Jimei University, Xiamen. His main research directions are computer vision and point cloud processing, including 3-D reconstruction, image and point cloud-based object detection and recognition, stereo vision, deep learning theory, and its applications.