



A dual projection method for semantic segmentation of large-scale point clouds

Haoying Zhao¹ · Aimin Zhou¹

Accepted: 1 April 2025 / Published online: 4 May 2025

© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2025

Abstract

3D point cloud semantic segmentation is an essential task for enhancing high-level perception in autonomous platforms. The projection-based method is an efficient approach since central projection can transform sparse point clouds into dense pseudo-2D image. Central projection suffers from scale imbalances, posing challenges for perceiving distant objects. We utilized parallel projection to adjust the distribution and scale of targets, enabling neural networks to better extract information from distant objects. We designed a joint training and inference framework to combine the perspectives of central projection and parallel projection, enhancing the final segmentation results. Our method achieves an mAcc of 89.7% and an mIoU of 59.3% on the SemanticKITTI validation set. On the SemanticKITTI test set, we achieve an mAcc of 89.0% and an mIoU of 58.7%. The experiments show that our method achieves superior segmentation performance, especially for distant targets and categories such as motor vehicles, distant road surfaces and column-like structures.

Keywords Point cloud · Semantic segmentation · Parallel projection · Autonomous driving system

1 Introduction

Semantic segmentation has attracted considerable attention in the area of autonomous driving perception system [5, 40], robotics [1] and remote sensing domains [38]. Semantic segmentation in autonomous driving scenarios can be viewed as a large-scale point cloud segmentation task, given the large number of points and the wide range of the point cloud. It provides the semantic parsing of the entire scene instead of generating the prediction position boxes for each semantic class. This form of segmentation is a fundamental task within the field of computer vision. In the context of autonomous driving scenarios, the point clouds generated by Lidar sensors constitute a representation of the vehicle's environment. It is essential to segment these point clouds into distinct parts. Deep learning-based methods have demonstrated significant success in the field of computer vision. Consequently, it is

a natural progression to apply deep learning techniques to address the challenges of point cloud semantic segmentation.

Point clouds inherently possess a disorderly and irregular nature, presenting a challenge when attempting to directly employ deep learning methods. Point-based methods [12, 14, 17, 22–24, 28, 29, 37, 39] are the pioneer methods, but constructing local spatial structures and extracting local and global features proves difficult. Because point clouds lack local geometry structure, searching for neighboring points and matching them with fixed weights requires additional computation and leads to loss of precision. Voxel-based methods [10, 19, 51] project unstructured point clouds into regular grid cells, enabling the use of 3D convolutional neural networks. However, as voxel resolution increases, the number of parameters and inference time of the model also increase significantly.

Projection-based algorithms [11, 32, 33, 41–43] have garnered significant attention for their exceptional efficiency. These methods involve initially projecting the point cloud onto a 2D image, creating a pseudo-image that encapsulates the positional information of each point. Subsequently, 2D neural networks are employed to segment these pseudo-images. Nonetheless, these pseudo-images encounter significant issues with scale imbalance and distribution irregularities concerning distant objects. This arises due to the central

✉ Aimin Zhou
amzhou@cs.ecnu.edu.cn

Haoying Zhao
hyzhao@stu.ecnu.edu.cn

¹ School of Computer Science and Technology, East China Normal University, Shanghai 200062, China

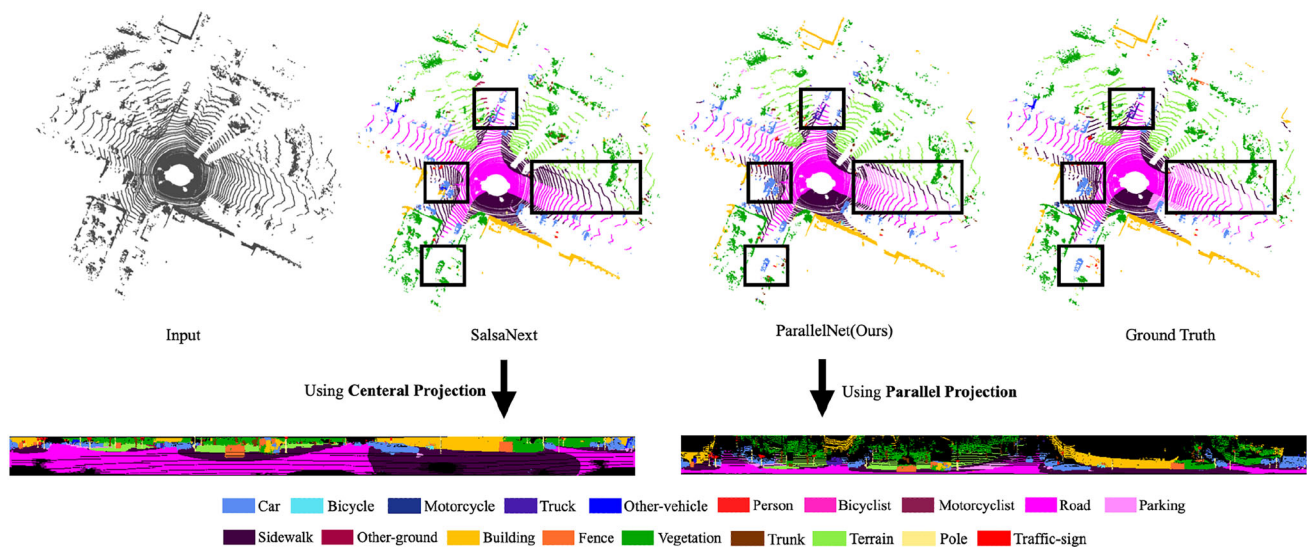


Fig. 1 Semantic segmentation results of SalsaNext [11] and our approach on SemanticKITTI [5]. The black boxes highlight the superior segmentation accuracy of our approach. Our method achieves satisfac-

tory segmentation results in the categories of vehicles and distant road surfaces. The difference in results is due to the adoption of different projection method

projection employed in the vertical direction of the method, which compresses the height of distant objects based on the number of occupied Lidar beams. Additionally, the Lidar's lower depression angle significantly outweighs its upper elevation angle, resulting in the pseudo-image being primarily dominated by nearby ground, while distant objects tend to cluster within the upper portion of the Lidar image.

To address the aforementioned limitations, this paper introduces a novel approach, the dual parallel projection method, and establishes a joint training and inferring framework. We propose the utilization of parallel projection within the context of semantic segmentation of point clouds. As depicted in Fig. 1, this approach mitigates the issue of the nearby road surface excessively dominating the scene, thereby achieving a more balanced distribution between near and distant objects. This balanced distribution eliminates the problem of height-related scale imbalance. To further enhance contextual information and optimize neighborhood feature extraction, we devise a joint training and inferring strategy that leverages the benefits of both dense central projection data and relatively sparse parallel projection data. As demonstrated in Fig. 1, our results, which is denoted as 'ParallelNet,' exhibit enhanced accuracy in segmentation, particularly noticeable in the cases of vehicles and distant road surfaces.

Overall, the main contributions of this paper can be summarized as follows:

1. We propose a novel ParallelNet that simultaneously utilizes data from parallel projection and central projection, where the parallel projection perspective works as an

enhancement of the central projection method for semantic segmentation.

2. Our designed joint training and prediction method leverages the advantages of both perspectives, allowing the relatively sparse parallel projection view to be better trained by the network.

2 Related work

3D perception involves complex tasks such as object detection [13, 25, 46, 47] and semantic segmentation [28, 29], which require computers to understand and interpret 3D information in the environment. The 3D semantic segmentation methods can be categorized into three classes: point-based methods, projection-based methods and voxel-based methods.

Point-based Method: Point-based methods typically employ farthest point sampling (FPS) [29] or random sampling [17] for downsampling. These methods establish neighborhood spaces using the KNN method [17] or spherical neighborhood search [29]. Methods based on shared MLP [17, 28, 29] extract features isotropically, while RandLA-Net also designs spatial encoding to enhance feature discriminability. Some methods employ position-aware convolutions [24, 39] or graph convolutions [27] for feature extraction. Geosegnet [7, 20] employs eigenvalue features and contrastive learning to address boundary issues.

Voxel-based Method Due to LiDAR sensors only being able to scan points on the surface of unobstructed objects, the generated point cloud is very sparse in space. The

voxels converted from these point clouds are also highly sparse. MinkowskiNet [10] introduces sparse convolution to enhance segmentation efficiency. Cylinder3D [51] designs a novel space partition method to ensure an even distribution of points in each block and introduces a new asymmetric module. SPVNAS [36] employs neural architecture search (NAS) to further enhance the network's performance.

Projection-based Method: Projection-based methods project the 3D point cloud to 2D images and then segment pseudo-2D images using existing 2D CNNs. Predominant projection techniques used to project 3D point cloud to 2D images include central projection [41], bird's-eye view (BEV) projection [6] and projection fusion-based hybrid techniques.

Many methods only utilize central projection. SqueezeSeg [41] incorporates central projection into this methodology and then employs a network composed of fire modules to segment the pseudo-images. A CRF method is utilized to refine the segmentation outcome. The PointSeg [15] utilized random sample consensus (RANSAC) during the projection of spherical images back to point cloud data to remove outliers. In SqueezeSeg v2 [42], a large convolutional kernel is adopted to expand the receptive field. The proposal of Geodesic Correlation Alignment addresses the domain shift problem. SqueezeSeg v3 [43] introduces the SAC module to handle uneven features within the pseudo-images. RangeNet++ [26] adopts the modified DarkNet backbone [31] and replaces CRF with GPU-based calculations. SalsaNext [11] introduces dilated convolution and pixel shuffle techniques to enhance the model's accuracy and efficiency.

Other methods also employ BEV projection or hybrid approaches. PolarNet [49] uses a polar-based BEV to better model the imbalanced spatial distribution of point clouds. VolMap [30] utilizes BEV projection and voxelization to generate a volumetric bird's-eye view where the height represents the different LiDAR layers. This volumetric map is used as input to train the model and further segment the point cloud. MPF [2] uses two models to handle central view and BEV independently, merging their softmax probabilities for final predictions. MVLidarNet [8] starts with predictions from the central projection image, projecting them to BEV for feature learning with feature pyramid networks. In FPS-Net, each image modality is processed through a fusion network and a 2D convolutional network, utilizing a multi-receptive field residual dense block (MRF-DB) for feature extraction and aggregation.

Central projection is dense but suffers from scale imbalance, while BEV provides scale balance but is overly sparse. We devise an oblique parallel projection method that achieves scale balance without the sparsity of BEV. Additionally, we design joint training and inference techniques to enhance neural network training and inference capabilities.

3 Method

We select the appropriate angle of parallel projection and combine the central projection and parallel projection using our jointly training and inferring framework to achieve a high segmentation accuracy for the segmentation targets.

This paper is the first time to introduce parallel projection in the semantic segmentation for autonomous driving, in order to reduce the scale imbalance and distribution imbalance problem. To further exploit dense neighborhood information and richer patterns, the framework of joint training and inferring is introduced, so that data from different perspectives are input into the model at the same time, and the overall accuracy of the model is improved. Experiments show that the segmentation accuracy of segmentation objects is significant improved.

3.1 Dual projection method

The projection-based algorithm initially projects the point clouds onto pseudo-images and subsequently conducts semantic segmentation on these pseudo-images. We utilize two projection methods, central projection and parallel projection, denoting it as the dual projection method. The advantage of central projection is that the projected images are dense with less occlusion. Neighborhood information can be effectively extracted. However, there is an issue of scale imbalance between near and distant objects. Extracting useful information from small-scale objects poses a challenge. In Sect. 3.1.1, we introduce the central projection method. In Sect. 3.1.2, we present the parallel projection method. We implemented both of these projections for each point cloud, producing two distinct pseudo-images.

3.1.1 Central projection method

The most common projection method is the central projection, which is determined by the way of LiDAR sensors scanning the scene and can minimize the information loss and maximize the density of projection points and reduce occlusion.

As for a point $p = (x, y, z)$ with Cartesian coordinates, the central mapping $\mathbb{R}^3 \rightarrow \mathbb{R}^2$ is used to transform them into 2D coordinates to form a range image. To form the new range view image, each points (x, y, z) in raw LiDAR is mapped to the image coordinate (u, v) as

$$\begin{aligned} u &= \frac{1}{2} * \left(1 - \frac{\arctan(\frac{y}{x})}{\pi} \right) * w \\ v &= \left(1 - \frac{\arcsin(\frac{z}{r}) + f_{\text{down}}}{f_{\text{up}} + f_{\text{down}}} \right) * h \end{aligned} \quad (1)$$

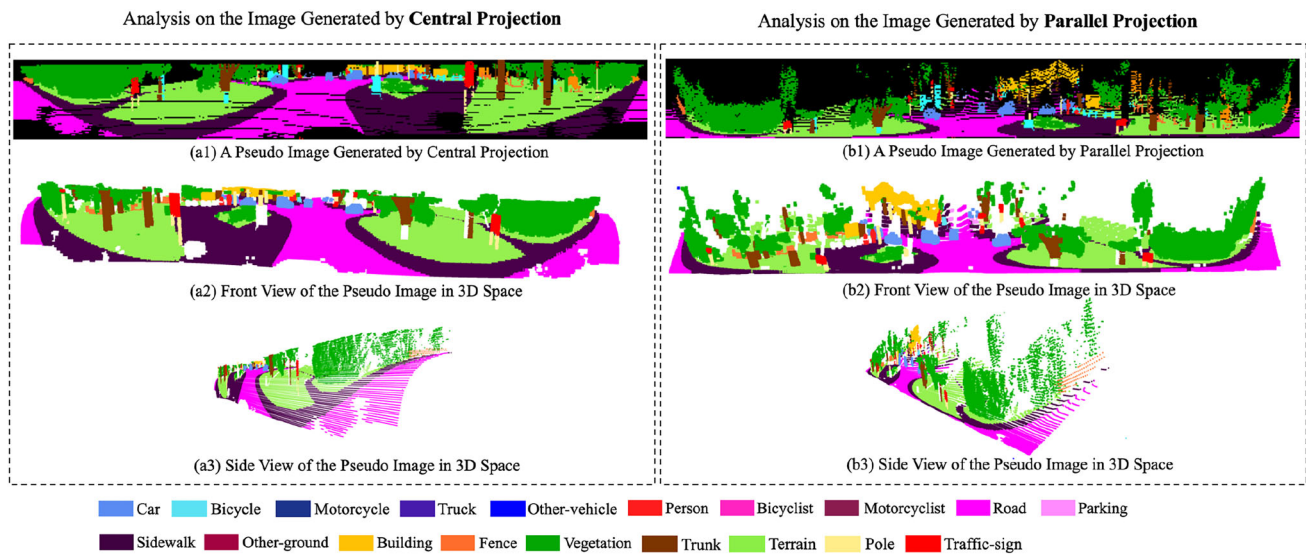


Fig. 2 A comparison of central projection and parallel projection. We compare the two images generated using different projection methods in three different ways

where (u, v) are image coordinates and (h, w) are the height and width of the desired image representation. f_{up} and f_{down} , respectively, represent the upward angle and downward angle of the vertical field of view of the sensor. And $r = \sqrt{x^2 + y^2 + z^2}$ is the range of each point.

This method replicates the scanning process of a 360-degree LiDAR. In terms of height, each scanning beam creates a pixel, effectively minimizing hidden areas. Distant objects use only a small number of LiDAR beams, resulting in height compression corresponding to distance. As shown in Fig. 2, a substantial portion of the area is projected as a road surface, rendering it challenging to discern smaller, densely distributed targets in the distance. Moreover, there is an issue with relatively high objects in the foreground, such as traffic signs, disproportionately occupying considerable space. An observation from the side view highlights that the ground assumes a prominent shape, attributed to its alignment with the arctan function. Consequently, using an inclined central projection does not offer a solution to this imbalance issue. Therefore, in addition to the central projection, a novel approach is requisite to address this concern.

3.1.2 Parallel projection method

This paper introduces a parallel projection module to address the issues of uneven distribution and scale disparities among distant objects in projection-based point cloud semantic segmentation problems.

In our paper, we directly adopts the circular inclined parallel projection of the point cloud. To form the new inclined parallel projection image, each points (x, y, z) in raw LiDAR

is mapped to the image coordinate (u, v) as

$$\begin{aligned} u &= \frac{1}{2} * \left(1 - \frac{\arctan(\frac{y}{x})}{\pi} \right) * w \\ v' &= r * \sin \left(\arcsin \left(\frac{z}{r} \right) + \alpha \right) \\ v &= \left(1 - \frac{v' - v'_{\min}}{v'_{\max} - v'_{\min}} \right) * h \end{aligned} \quad (2)$$

where (u, v) are image coordinates, (h, w) are the height and width of the desired image representation, α is the rotation angle, v' is the intermediate coordinates, $v'_{\min}$ is the minimize value among v' and $v'_{\max}$ is the maximum value among v' . And $r = \sqrt{x^2 + y^2 + z^2}$ is the range of each point. The transformation process of this projected coordinate is shown in Fig. 3

There are various methods for projecting point clouds. For instance, range images aim to retain information to the utmost degree while reducing occlusion. On the other hand, bird's-eye view (BEV) projection images are employed to maintain the original object sizes [35, 48], which is comparatively common in object detection. BEV projection also represents a form of parallel projection, functioning vertically. However, the projection images tend to exhibit significant sparsity, as depicted in Fig. 4, thereby impacting the utilization of neighboring information and the efficiency of neural network processing. Frequently, it becomes necessary to augment the projection size of BEV images to achieve favorable segmentation outcomes.

In range images projected by central projection methods, distant objects are diminished in size and concentrated in the upper regions of the images, thereby posing a challenge for

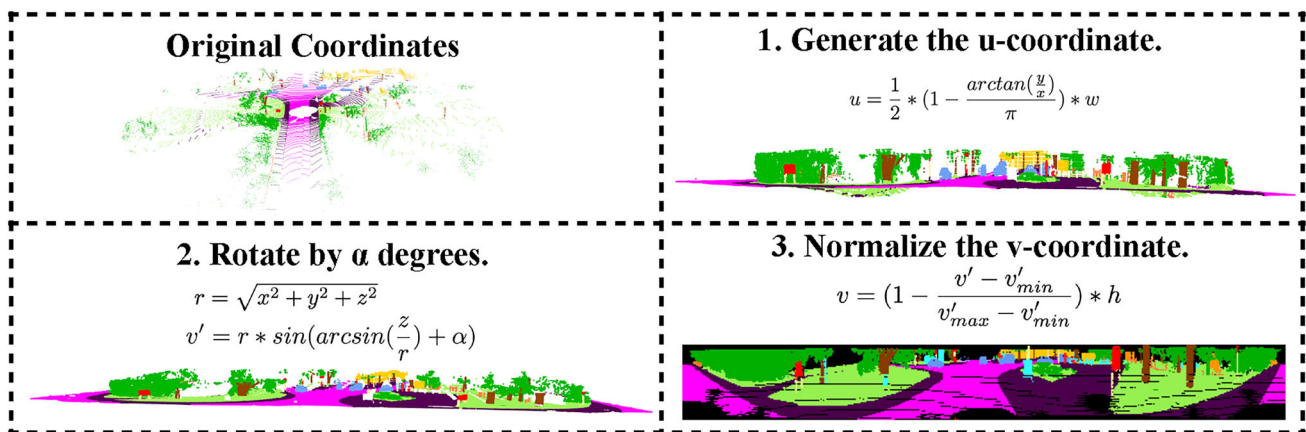
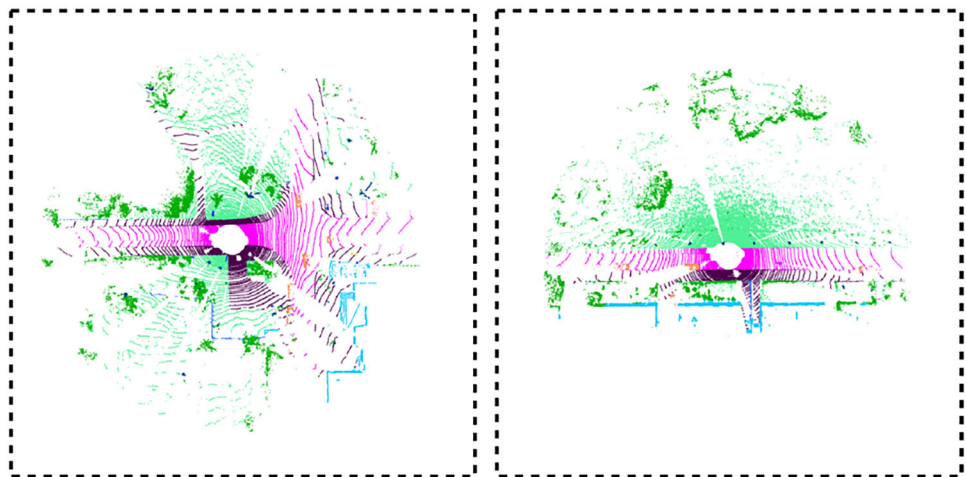


Fig. 3 Pipeline for calculating coordinates in the process of parallel projection

Fig. 4 Two samples of pseudo-images produced by BEV projection. The object scales in the pseudo-images are well balanced, but significantly sparse



neural network segmentation. In the case of BEV images, all points only occupy 30% of the projected area after projection. Such sparsity makes maintaining a substantially large image size becomes necessary. Furthermore, the absence of circular projection leads to varying degrees of sparsity across the images. We introduce the circular inclined parallel projection method to address the semantic segmentation challenges inherent in point cloud data. This approach preserves the scale of distant objects and balances the distribution of nearby and remote objects. The circular view projection also moderates the sparsity gradient along the horizontal axis, yielding a less pronounced effect than what is observed in BEV images.

To determine a suitable angle, we divide the range of 2 to 30 degrees into 50 intervals and compute the projection pseudo-images' density. As illustrated in Fig. 5, we select the angle that yields the densest projection pseudo-images, which is the 8th angle at 6.48 degrees. Selecting an angle that is too small can obscure distant objects, while an excessively large angle results in an excessive number of distant objects, leading to the compression of object heights. More-

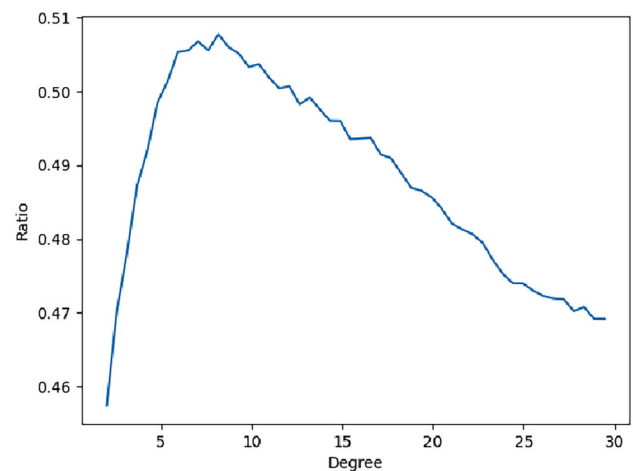


Fig. 5 Statistics of pseudo-image density at different angles. To preserve the density of the pseudo-image, we selected the 8th angle

over, an overly large angle increases sparsity and diminishes information within the unit receptive field.

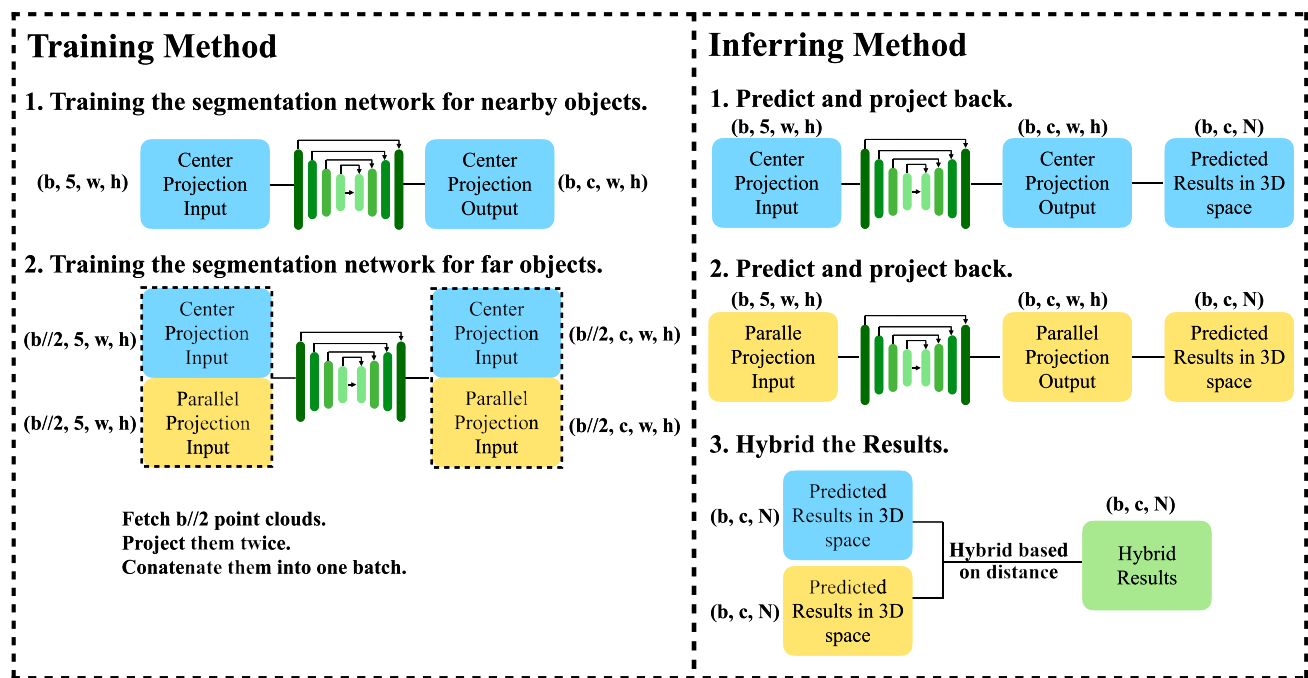


Fig. 6 Proposed joint training and inferring framework

As depicted in Fig. 2, the projection attains a uniform scale for both near and distant views, resulting in a more equitable distribution of objects. Additionally, the information content throughout the entire image is evenly spread, rendering it conducive for learning with a unified convolution kernel. Upon examining the side view, it becomes apparent that the ground adheres to a linear trajectory, effectively accomplishing the goal of achieving a balanced distribution.

3.2 Joint training and inferring framework

We devise a joint training and inferring framework to allow the network to leverage two perspectives, as depicted in Fig. 6. The two viewing angles exhibit similarity due to both employing a central annular projection in the horizontal direction. However, the central projection offers benefits in terms of density and proximity to objects, while parallel projection excels in maintaining scale and achieving distribution equilibrium.

We develop an approach to merge these perspectives into two samples within a batch for training. After extracting data in the data loader, assuming a batch size of $b/2$, we perform two projections, concatenate them to obtain data with a batch size of b , and input it into the neural network for training. Using a new projection angle in the parallel projection increases sparsity compared to the original sensor sampling angle. This situation poses challenges, especially for training convolutional neural networks, due to the scarcity of neighboring information. To enhance the learning and effective

utilization of neighborhood information, we include central projection pseudo-images in the joint training process.

To effectively leverage the benefits of both near and distant perspectives, we have formulated a joint inference framework for semantic point cloud segmentation during prediction. This framework involves merging the two segmented point clouds based on their respective distances. Specifically, the segmentation outcomes from the central projection is applied to near points, while the segmentation result from the parallel projection is utilized for far points. The fusion threshold is set at 15 ms. 15 ms is an empirical setting. Based on the sensor's pitch and yaw angles, objects beyond 15 ms face significant vertical compression issues. Given the projection's substantial depression angle of the LiDAR sensor, close objects within the central projection exhibit a larger scale and occupy significant space. Employing distance-based fusion allows us to harness the strengths of both projection methods effectively.

Our experiments conclusively demonstrate that this hybrid training approach effectively enhances the network's segmentation capability for both nearby and distant objects simultaneously. This enhancement is attributed to the fact that the pseudo-image obtained through central projection features denser neighborhoods and richer information. During the prediction phase, the pseudo-images derived from central projection predominantly encompasses nearby objects. This inherent characteristic enables the hybrid prediction method to proficiently capitalize on the benefits provided by both projection methodologies.

Table 1 Quantitative results on SemanticKITTI dataset

	Approach	Size	Car	Bicycle	Motorcycle	Truck	Other vehicles	Person	Bicyclist	Motorcyclist	Road	Parking
Point-wise	PointNet [28]	50K points	46.3	1.3	0.3	0.1	0.8	0.2	0.2	0	61.6	15.8
	SPG [22]		45.0	28.5	0.6	0.6	64.3	49.3	0.1	0.2	0.2	0.8
	PointNet++ [29]		53.7	1.9	0.2	0.9	0.2	0.9	1.0	0.0	72	18.7
	SPGraph [23]		68.3	0.9	4.5	0.9	0.8	1.0	6.0	0.0	49.5	1.7
	SPLATNet [14]		66.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	70.4	0.8
	TangentConv [37]		86.8	1.3	12.7	11.6	10.2	17.1	20.2	0.5	82.9	15.2
	RandLa-Net [17, 18]		94.2	26.0	25.8	40.1	38.9	49.2	48.2	7.2	90.7	60.3
	KPConv [39]		96.0	30.2	42.5	33.4	44.3	61.5	61.6	11.8	88.8	61.3
	LG-Net [50]		93.9	36.5	36.0	42.7	36.8	50.1	47.7	0.9	90.6	63.0
	SqueezeSeg [41]		68.3	18.1	5.1	4.1	4.8	16.5	17.3	1.2	84.9	28.4
Projection-based	DarkNet21Seg [5]	64×2048 pixels	85.4	26.2	26.5	18.6	15.6	31.8	33.6	4.0	91.4	57.0
	DarkNet53Seg [5]		86.4	24.5	32.7	25.5	22.6	36.2	33.6	4.7	91.8	64.8
	SqueezeSeg V2 [42]		81.8	18.5	17.9	13.4	14.0	20.1	25.1	3.9	88.6	45.8
	RangeNet++ [26]		91.4	25.7	34.4	25.7	23.0	38.3	38.8	4.8	91.8	65.0
	PolarNet [28]		93.8	40.3	30.1	22.9	28.5	43.2	40.2	5.6	90.8	61.7
	3D-MiniNet [3]		90.5	42.3	42.1	28.5	29.4	47.8	44.1	14.5	91.6	64.2
	SqueezeSeg V3 [43]		92.5	38.7	36.5	29.6	33.0	45.6	46.2	20.1	91.7	63.4
	LatticeNet [33]		92.9	16.6	22.2	26.6	21.4	35.6	43.0	46.0	90.0	59.4
	MPF [2]		93.4	30.2	38.3	26.1	28.5	48.1	46.1	18.1	90.6	62.3
	SalsaNext [11]		91.1	44.0	40.3	38.3	32.8	58.5	59.4	19.9	91.2	61.6
	ParallelNet (ours)		92.2	46.1	40.3	37.5	33.7	60.2	60.1	18.0	91.5	62.0

Table 1 continued

	Approach	Size	Sidewalk	Other grounds	Building	Fence	Vegetation	Trunk	Terrain	Pole	Traffic sign	mIoU
Point-wise	PointNet [28]	50K points	35.7	1.4	41.4	12.9	31.0	4.6	17.6	2.4	3.7	14.6
	SPG [22]		48.9	27.2	24.6	0.3	2.7	0.1	20.8	15.9	0.8	17.4
	PointNet++ [29]		41.8	5.6	62.3	16.9	46.5	13.8	30.0	6.0	8.9	20.1
	SPGraph [23]		24.2	0.3	68.2	22.5	59.2	27.2	17.0	18.3	10.5	20.0
	SPLATNet [14]		41.5	0.0	68.7	27.8	72.3	35.9	35.8	13.8	0.0	22.8
	TangentConv [37]		61.7	9.0	82.8	44.2	75.5	42.5	55.5	30.2	22.2	35.9
	RandLa-Net [17, 18]		73.7	20.4	86.9	56.3	81.4	61.3	66.8	49.2	47.7	53.9
	KPConv [39]		72.7	31.6	90.5	64.2	84.8	69.2	69.1	56.4	47.4	58.8
	LG-Net [50]		74.2	19.1	90.6	62.3	84.6	64.1	69.7	51.5	55.0	56.3
	SqueezeSeg [41]		54.7	4.6	61.5	29.2	59.6	25.5	54.7	11.2	36.3	30.8
Projection-based	DarkNet21Seg [5]	64×2048 pixels	74.0	26.4	81.9	52.3	77.6	48.4	63.6	36.0	50.0	47.4
	DarkNet53Seg [5]		74.6	27.9	84.1	55.0	78.3	50.1	64.0	38.9	52.2	49.9
	SqueezeSeg V2 [42]		67.6	17.7	73.7	41.1	71.8	35.8	60.2	20.2	36.3	39.7
	RangeNet++ [26]		75.2	27.8	87.4	58.6	80.5	55.1	64.6	47.9	55.9	52.2
	PolarNet [28]		74.4	21.7	90.0	61.3	84.0	65.5	67.8	51.8	57.5	54.3
	3D-MiniNet [3]		74.5	25.4	89.4	60.8	82.8	60.8	66.7	48.0	56.6	55.8
	SqueezeSeg V3 [43]		74.8	26.4	89.0	59.4	82.0	58.7	65.4	49.6	58.9	55.9
	LatticeNet [33]		74.1	22.0	88.2	58.8	81.7	63.6	63.1	51.9	48.4	52.9
	MPF [2]		74.5	30.6	88.5	59.7	83.5	59.7	69.2	49.7	58.1	55.5
	SalsaNext [11]		74.8	25.9	86.2	56.8	79.6	61.1	64.2	52.9	61.7	57.9
	ParallelNet(ours)		75.5	27.8	87.1	57.6	80.7	63.6	65.1	54.8	61.9	58.7

The bold text represents the best-performing model in each category (column)

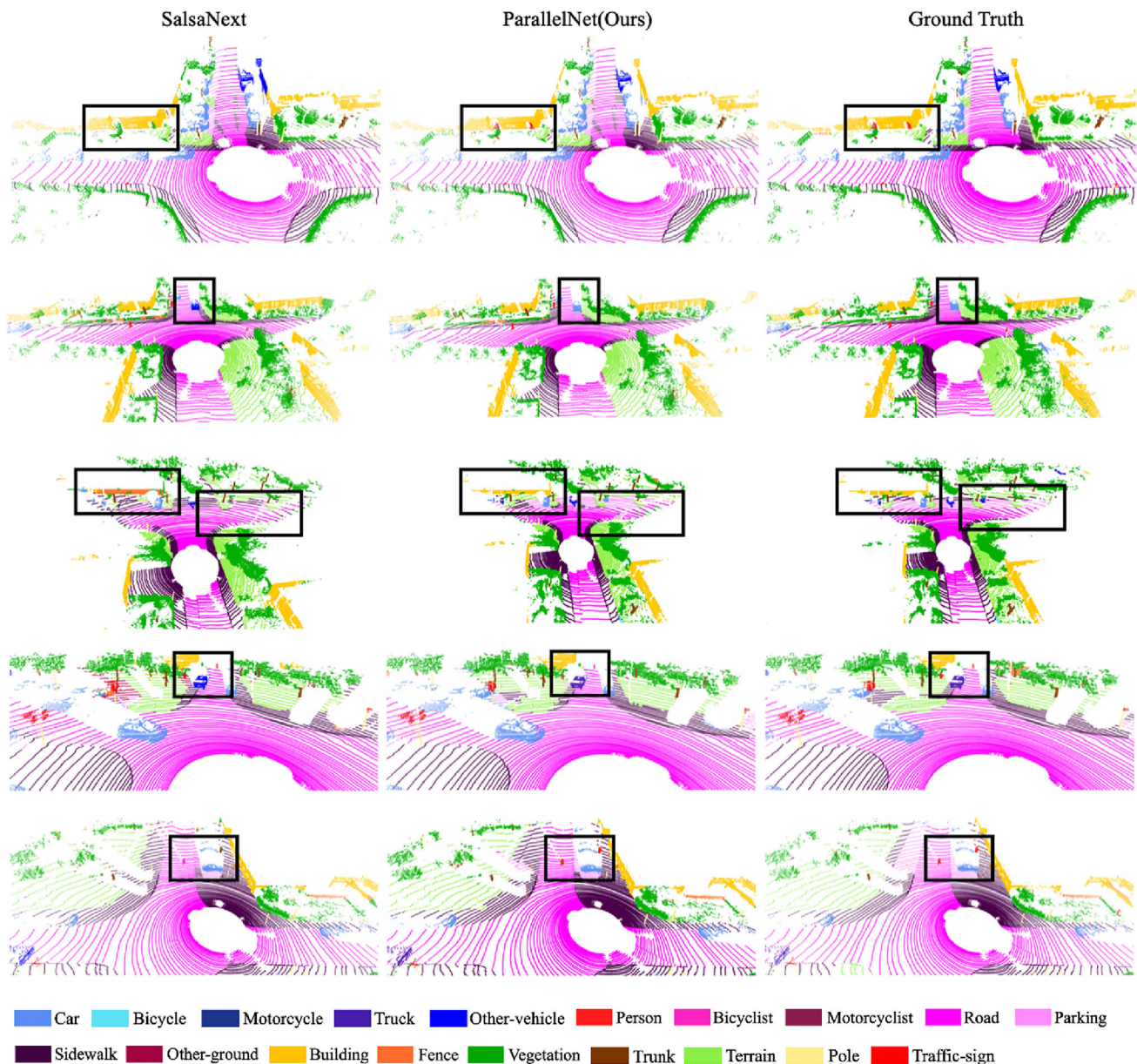


Fig. 7 Qualitative results of our method on SemanticKITTI dataset (valid). The black boxes highlight the superior segmentation accuracy of our approach

3.3 Network architecture

We employ UNet-like fully convolutional neural networks. The input size for each sample is $2048 * 64 * 5$, encompassing 5 dimensions, namely x , y , z , range and remission. The pseudo-image is generated through the original point cloud projection. The five-stage network undergoes successive downsampling, leading to feature sizes of $512 * 32$, $256 * 16$, $128 * 8$ and $64 * 4$. The number of channels in each feature stage is 32, 64, 128, 256 and 256, respectively. Following each convolution stage during downsampling, we apply dropout [16]. Additionally, in the upsampling stage, we uti-

lize pixel shuffle [34] to augment scale. Our network structure adheres to the framework of the classical projection-based point cloud semantic segmentation network, specifically, SalsaNext [11].

4 Experiments

To validate the efficacy of our designed projection method and the efficiency of the training prediction framework, we conducted experiments on the SemanticKITTI dataset [5].

Additionally, we employed an ablation study to assess the impact of our individual modules.

4.1 Dataset

SemanticKITTI Dataset: SemanticKITTI [5] was introduced based on the well-known KITTI Vision benchmark illustrating complex outdoor traffic scenes. It contains 22 stereo sequences of point clouds with 433,552 densely annotated LiDAR scans collected with a Velodyne HDL-64E Lidar, where each scan contains approximately 139k points. Each scan is a large-scale point cloud with about 10^5 points and spanning up to $160 \times 160 \times 20$ m in real 3D space. The raw 3D points have only 3D coordinates without color information and are precisely annotated in 19 semantic classes. Officially sequences 00 to 07 and 09 to 10 (19,130 scans) are used for training, sequence 08 (4071 scans) for validation and sequences 11 to 21 (20,235 scans) for online testing.

4.2 Implementation details

Our proposed method is implemented by the PyTorch deep learning framework. The experimental platform is a workstation equipped with an Intel (R) Core (TM) i7-12700KF @ 3.6 Ghz CPU a NVIDIA GeForce RTX 3090ti GPU and a pre-installed Linux system. All experiments are conducted with single-card training. In this paper, SGD is used as the optimization algorithm for the model and a total of 150 epochs are trained. The initial learning rate is set to 10^{-2} , and the learning rate decays to 10^{-4} after 150 epochs.

4.3 Metrics

Since the 3D semantic segmentation problem is essentially the point-level classification tasks, we first introduce the important concept called confusion matrix, as shown in Table 1. The confusion matrix divides the results into four categories based on the relationship between the prediction and the ground truth.

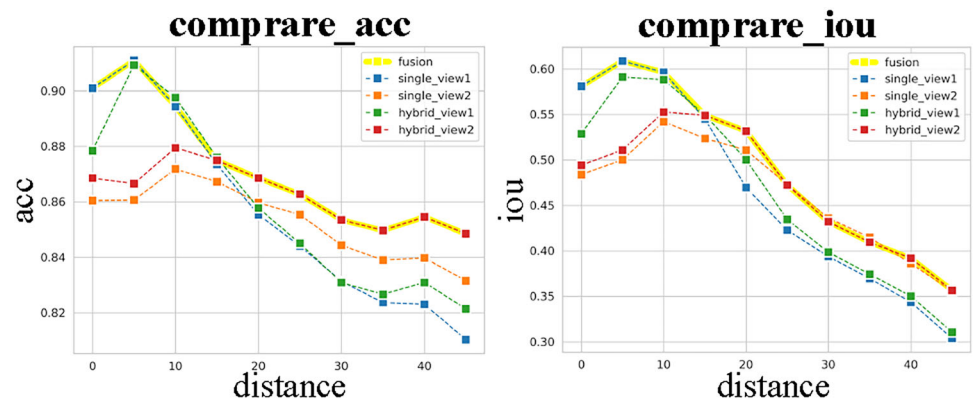
Here, true positive (TP) is the number of correctly predicted points. On the other hand, false positive (FP) and false negative (FN) are the numbers of wrongly predicted points. The former means that the prediction recognizes the point as belonging to the specific class, whereas the ground truth refers to the point as not belonging to the specific class, and the latter is vice versa. With the above knowledge, we are going to introduce two kinds of metrics, mean accuracy (mAcc) and mean intersection over union (mIoU), which are both widely applied in the area of semantic segmentation. Accuracy is a basic metric for evaluating classification tasks. Informally, accuracy is the fraction of predictions our model got right, which is calculated by dividing number of correct predictions by total number of predictions. Since our tasks

Table 2 Quantitative results on SemanticKITTI dataset for dual projection method

Approach	Car	Bicycle	Motorcycle	Truck	Other vehicles	Person	Bicyclist	Motorcyclist	Road	Parking	Sidewalk	Other grounds	Building	Fence	Vegetation	Trunk	Terrain	Pole	Traffic sign	mAcc	mIoU
Single_view1	91.4	38.4	46.8	71.6	54.7	63.7	75.3	0.0	93.5	41.5	78.7	0.3	84.5	49.7	83.9	60.8	68.9	56.1	46.3	89.5	58.5
Single_view2	85.6	32.9	31.4	63.1	34.9	57.3	66.4	0.0	86.8	28.7	69.2	0.9	85.9	44.2	81.0	56.1	62.7	47.9	43.3	86.2	51.5
Fusion_view1	90.4	36.2	45.3	68.5	34.6	60.0	70.5	0.0	93.8	43.6	79.6	0.4	85.0	44.7	83.2	61.8	67.2	58.1	47.2	89.1	56.3
Fusion_view2	86.7	31.1	29.5	65.1	41.2	57.9	62.1	0.0	87.1	34.5	68.5	0.2	86.3	42.9	82.7	60.3	65.2	52.8	46.2	86.9	52.6
Hybrid	92.2	39.7	47.1	78.4	54.4	66.5	77.1	0.0	93.9	43.1	79.4	0.4	85.8	48.6	84.1	62.7	68.8	57.4	47.1	89.7	59.3

The bold text represents the best-performing model in each category (column)

Fig. 8 A comparison of segmentation accuracy and mIoU for different views across 5 distance segments



is the multi-class semantic segmentation problem, we calculated the mean accuracy of each class as follows:

$$\text{mAcc} = \frac{1}{C} \sum_{c=1}^C \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c} \quad (3)$$

where C is the total number of class, and TP_c and FP_c is the true positive and false positive of class index c , respectively.

Since mean accuracy neglects the false negative part, it cannot fully reflect the performance. To solve this drawback, mean intersection over union is proposed to give the more comprehensive comparison, which is literally the number of overlapped point divided by the number of union points of prediction and ground truth. The formula for 3D mean intersection over union is defined as:

$$\text{mIoU} = \frac{1}{C} \sum_{c=1}^C \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c + \text{FN}_c} \quad (4)$$

where FN_c is the false negative of class index c and the rest is the same as mean accuracy. Both mAcc and mIoU are in the range of $[0, 1]$.

4.4 Results

Table 1 demonstrates the quantitative results of the proposed method and the state-of-the-art approaches on the SemanticKITTI dataset. For a fairer comparison, the results in this study were obtained without employing test time augmentation. In addition to the projection-based methods, we also compared the point-based methods. While the point-based methods have fewer parameters and a smaller network size, they lack spatial structure, which reduces the efficiency of these methods. Constructing spatial structures and matching weights for each layer in the point-based methods requires more computational resources. Despite the challenges posed by complex scenarios in traffic environments, our network denoted as ‘ParallelNet’ effectively identifies semantic labels. As evident from this table, our

method achieves the highest mIoU of 58.7% among all these methods. We outperform other advanced and competent approaches in 7 out of the 19 classes, while our performance in other classes is on par with those methods. The notable results can be attributed to our method’s more balanced distribution between near and far objects.

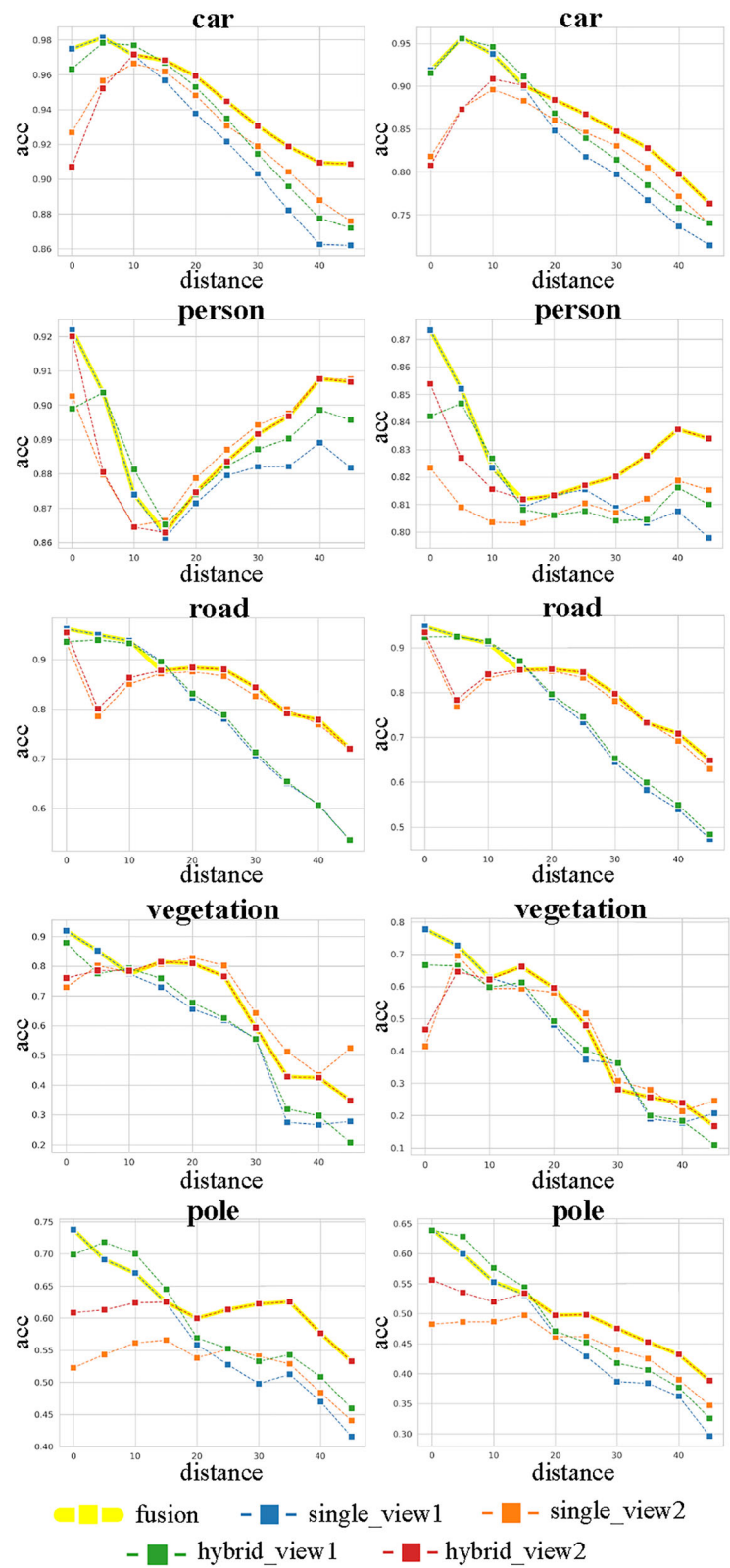
It is noteworthy that our algorithm attains favorable outcomes in both the motorcycle and people categories. Improved results are also observed in categories such as sidewalk, pole and traffic sign. In the case of vehicle and people categories, their volumes are relatively small. However, the challenge arises when segmenting distant vehicles and pedestrians due to the uneven distribution in the projection pseudo-images. Parallel projection addresses this issue by allowing distant, smaller categories to occupy more space, thereby enhancing segmentation metrics. Similarly, for pole and sidewalk, the compression of height in the projection pseudo-images make shape identification difficult. The use of parallel projection to restore height enhances segmentation accuracy in these cases.

For categories like buildings and vegetation, this method does not yield a significant advantage due to their larger size, allowing them to occupy more space even after height compression.

We also find that our approach shows advantages in segmenting smaller objects such as bicycle, motorcycle, bicyclist, pole and traffic sign by comparing with the better performing point-based methods in Table 1. This advantage is likely because smaller objects are harder to distinguish in 3D point space due to their less distinctive shape. In the projection space, the data is denser and shapes are more easily discerned. However, our method does not perform as well on larger categories like road and terrain. This may be because the original coordinates are better preserved in the 3D space, making it easier to identify planar surfaces, whereas in the 2D projection space, planar structures are stretched and harder to recognize.

Figure 7 shows the qualitative results of our method on the validation split. As depicted within in the black boxes,

Fig. 9 A comparison of segmentation accuracy and mIoU for different views across 5 distance segments in representative categories



our model can distinguish cars, trucks in a complex real scenario. Additionally, buildings at a distance exhibit better segmentation performance due to improved visibility. Distant roadside areas are also more readily identifiable, such as parking that are no longer misclassified as roads. People on the road achieve better segmentation.

In addition, some methods such as 2DPass and KPRNet utilize multimodal features after projection. KPRNet explores various ways of combining different data modalities beyond solely focusing on 2D methods, while 2DPass employs knowledge distillation to learn features extracted by 2D models. Methods like RangeViT, TransRVNet, and Rangeformer all leverage Transformers to process pseudo-images. However, they come with a significant increase in parameter count, memory consumption and higher FLOPs. These approaches typically use larger-scale models or the fusion of multiple models, resulting in more resource utilization or exploitation of different data modalities.

In addition, some methods such as 2DPASS [45] and RPVNet [44] utilize multimodal features after projection. RPVNet [44] explores various ways of combining different data modalities beyond solely focusing on 2D methods, while 2DPass [45] employs knowledge distillation to learn features extracted by 2D models. Methods like RangeViT [4], TransRVNet [9] and Rangeformer [21] all leverage Transformers to process pseudo-images. However, they come with a significant increase in parameter count, memory consumption and higher FLOPs. These approaches typically use larger-scale models or the fusion of multiple models, resulting in more resource utilization or exploitation of different data modalities.

4.5 Architecture design analysis

In this subsection, we carried out the following analysis to validate the efficacy of each component within our method. We examine the roles of the dual projection method, the selection of projection angles and the hybridization method in influencing the segmentation results. All ablation studies are conducted on the SemanticKITTI Validation set, employing the evaluation metrics of mIoU (mean intersection over union) and accuracy.

4.5.1 Analysis on dual projection method

We calculate the mIoU and accuracy at 10 different distances using the SemanticKITTI dataset. For the sake of convenience in expression, we denote the pseudo-images generated by central projection as ‘view1’ and those generated by parallel projection as ‘view2.’ We compare models trained on view 1, view 2 and the hybrid view as shown in Table 2.

The SemanticKITTI dataset’s maximum significant distance is 50 ms. We divide this distance into 10 equal segments

Table 3 Quantitative results on SemanticKITTI dataset for different projection angles

Approach	Car	Bicycle	Motorcycle	Truck	Other vehicles	Person	Bicyclist	Motorcyclist	Road	Parking	Sidewalk	Other grounds	Building	Fence	Vegetation	Trunk	Terrain	Pole	Traffic sign	mAcc	mIoU
Single_degree2	80.8	24.8	27.8	42.7	34.3	42.3	56.5	0.0	82.9	25.4	63.3	1.9	81.4	43.4	78.3	47.0	60.2	36.9	40.4	83.6	45.8
Single_degree5	82.8	28.8	33.8	12.6	42.3	45.9	60.0	0.0	85.0	34.0	66.2	0.9	82.4	35.9	78.0	47.3	63.4	36.7	39.8	86.2	46.1
Single_degree8	85.2	28.8	22.7	17.5	36.8	44.2	60.7	0.0	86.6	32.8	67.7	1.9	83.0	38.4	80.2	49.2	65.6	42.0	40.6	85.6	46.5
Single_degree11	85.0	25.6	23.4	17.4	27.4	42.4	56.0	0.0	86.1	32.2	67.0	1.5	82.1	34.4	78.4	48.4	64.2	39.0	39.8	84.6	44.8

The bold text represents the best-performing model in each category (column)

Fig. 10 A comparison of segmentation accuracy and mIoU for different projection angles across 5 distance segments

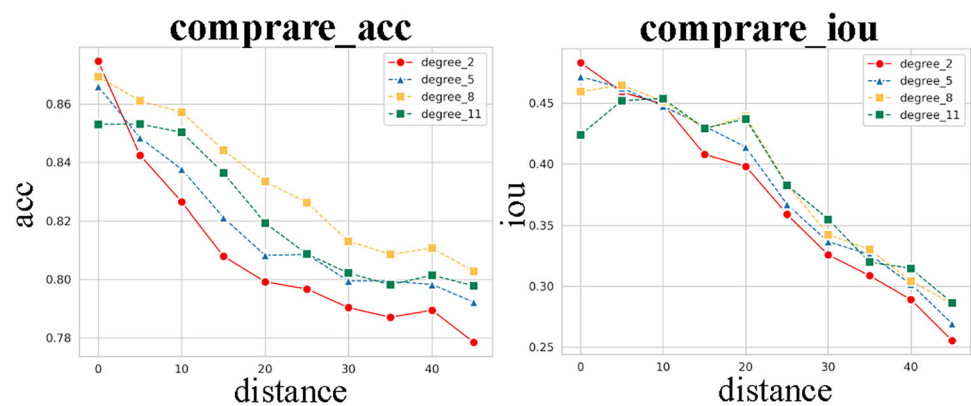
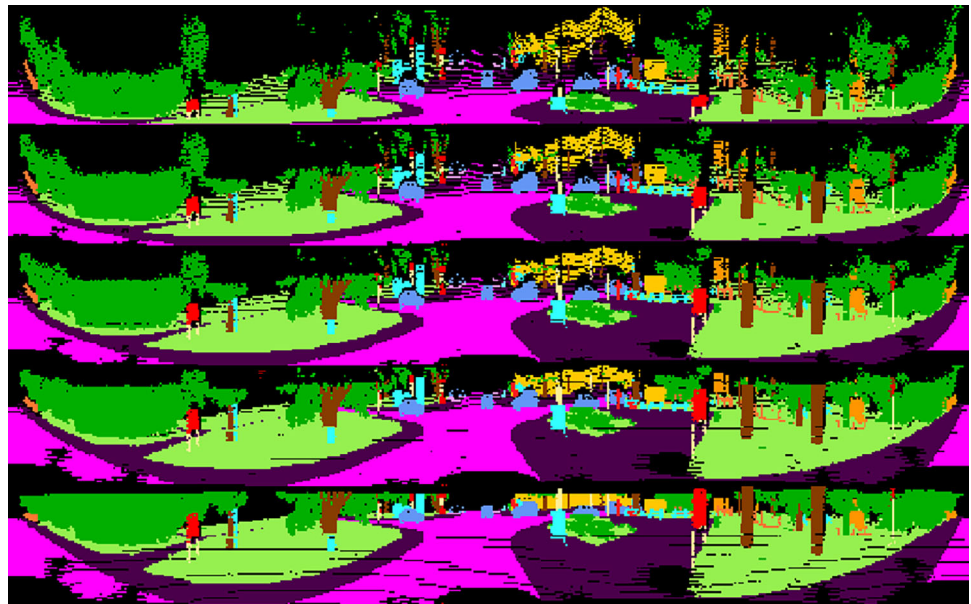


Fig. 11 Five pseudo-images with varying degrees of coordinate mixing. The two viewing coordinates are fused using mixing ratios of 0, 0.3, 0.5, 0.7 and 1.0, from top to bottom



and calculate the accuracy and mIoU at the midpoint of each segment. Figure 8 shows that as the distance increases, the overall accuracy and mIoU decrease. This reduction occurs because the height of distant objects is compressed and concentrated in the upper part of the projection, significantly affecting the segmentation accuracy of distant objects.

In Table 2, ‘Single_view1’ refers to training the network using central projections exclusively. ‘Single_view2’ involves training the network using inclined parallel projections only. ‘Fusion_view1’ employs the joint training method proposed in this paper, but solely utilizes ‘view1’ for predictions. ‘Fusion_view2’ applies the joint training method proposed in the text, but only leverages ‘view2’ for predictions. The ‘Hybrid’ category represents results obtained through our hybrid training and inferring framework.

Table 2 evidently shows that the hybrid approach has achieved favorable outcomes in the categories of motor

vehicles and pedestrians. In the road category, the impact is relatively consistent, while the vegetation category also demonstrates improved results. Similarly, in the pole and traffic sign categories, a more positive effect is observed.

We observe that when using only ‘view1,’ the segmentation of distant objects does not yield satisfactory results. When employing singular ‘view2’ training, the object mIoU at a distance remains comparable to joint training, with the accuracy in joint training being superior. At close range, both mIoU and accuracy are lower than those achieved through joint training.

Analyzing the classification visualization results as shown in Fig. 9, it becomes apparent that in the car category, the recognition accuracy of distant cars has significantly improved. Due to the reduced height of vehicles, the restoration of scale through parallel projection has a substantial impact. Furthermore, challenging-to-identify vehicles often

appear in the distance, where a more even distribution can also enhance the occupied area of vehicles.

In the case of the people category, it is evident from the figure that the segmentation accuracy and mIoU experience significant improvement at moderate distances. This improvement is attributed to the people's relatively greater height than width. The height recovery enhances the occupied area at intermediate distances, while challenges persist in long-distance segmentation due to the narrow width of pedestrians. For the road category, 'view1' exhibits high segmentation accuracy in proximity because a substantial proportion of nearby pavement is captured in this view. As the distribution imbalance is mitigated, the recognition accuracy of distant pavement is also significantly enhanced. In the vegetation category, improved distribution uniformity allows for easier differentiation of distant vegetation. In the pole category, the segmentation accuracy at a distance is effectively elevated due to the height recovery.

Based on the results from Table 2, and Figs. 8 and 9, we observe that as the distance increases, the segmentation performance for both views deteriorates. However, 'view 1' has an advantage for closer distances, while 'view 2' performs better for farther distances, with the transition point around 15 ms. Training 'view 1' alone yields better results, but 'view 2' requires joint training with 'view 1' to achieve better performance than 'view 2' on its own. Segmentation results have significantly improved for categories such as low-height vehicles, distant vegetation and road, and the narrow-width pole category.

4.5.2 Analysis on projection angle

The selection of the angle of inclination in parallel projection holds significant importance, as it determines the proportion of space occupied by near and distant objects. We begin by analyzing the angle's impact on the projection image's density, as an excessively low density can lead to voids that detrimentally affect segmentation accuracy. As illustrated in Fig. 5, we divide the range from 2 to 30 degrees into 50 equal segments and then compute the density of the projected image. This analysis reveals a higher density within the range from the second to the tenth degrees. Consequently, we chose projection angles of 2nd, 5th, 8th and 11th degrees for our experimental comparative analysis. The results are denoted as 'Single_degree2,' 'Single_degree5,' 'Single_degree8' and 'Single_degree11,' respectively. The interval of 8th degrees is the angle selected for the primary experiment in this paper. The corresponding experimental results are presented in Table 3. This experiment was conducted on the SemanticKITTI validation set with a resolution of $64 * 512$ pixels.

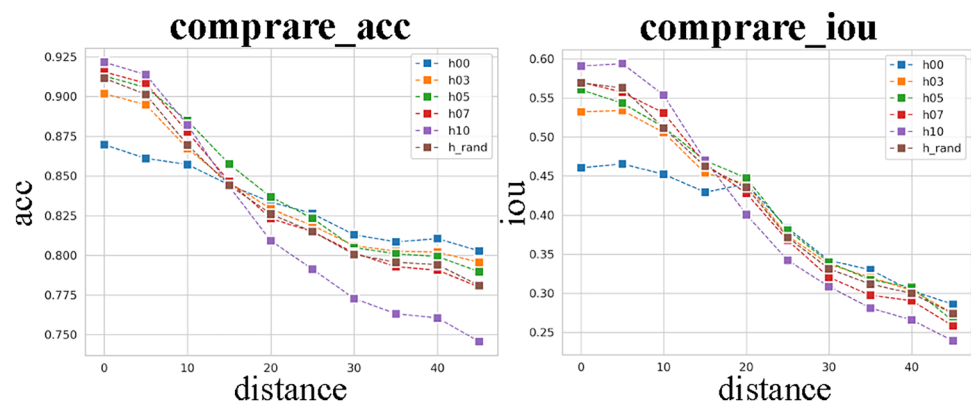
Table 3 evidently shows that the 8th angle proves to be the optimal choice. It is observable that as the angle increases, the

Table 4 Quantitative results on SemanticKITTI dataset for different levels of coordinate mixing

Approach	Car	Bicycle	Motorcycle	Truck	Other vehicles			Person	Bicyclist	Motorcyclist	Road	Parking	Sidewalk	Other grounds	Building	Fence	Vegetation	Trunk	Terrain	Pole	Traffic sign	mAcc	mIoU
h_00	85.2	28.8	22.8	17.4	36.8	44.2	61.0	0.0	0.0	0.0	86.6	33.2	67.6	1.9	83.2	38.8	80.1	49.2	65.6	42.0	40.7	85.6	46.6
h_03	89.1	36.3	35.1	51.6	27.5	46.7	64.9	0.0	0.0	0.0	91.6	46.4	75.4	0.2	83.9	44.5	81.2	50.7	67.5	45.6	38.6	87.8	51.4
h_05	90.8	35.9	46.0	43.1	40.5	53.2	71.6	0.0	0.0	0.0	92.0	47.1	76.8	0.8	83.6	46.8	83.1	51.1	73.1	43.5	40.3	88.9	53.6
h_07	90.4	38.4	44.5	51.2	40.6	46.4	73.1	0.0	0.0	0.0	93.2	48.3	77.7	0.0	83.3	49.2	82.0	53.0	70.0	45.1	41.6	88.7	54.1
h_10	88.7	35.3	44.6	70.2	36.7	51.7	72.3	0.0	0.0	0.0	93.4	52.4	78.6	0.1	81.7	49.3	82.1	54.0	71.3	45.4	40.1	88.8	55.2
h_rand	89.8	30.4	47.4	51.0	41.0	50.4	72.4	0.0	0.0	0.0	92.0	47.4	76.2	0.1	83.2	51.2	81.4	52.5	67.9	44.7	41.0	88.2	53.7

The bold text represents the best-performing model in each category (column)

Fig. 12 A comparison of segmentation accuracy and mIoU for different levels of coordinate mixing across 5 distance segments



segmentation performance of most categories improves. This improvement occurs because the distribution space for distant objects is no longer constrained. However, as the angle is further increased, the segmentation performance diminishes. This decline happens due to the reduced efficiency in utilizing space caused by the increased voids. We can observe that the accuracy of the motorcycle category improves to a certain extent with the increase in angle, yet it also introduces a higher rate of misjudgments. The segmentation efficiency of the road and vegetation categories experiences significant enhancement with increased angles. Similarly, for pole objects, the segmentation performance is also enhanced.

Analyzing the results by distance as shown in Fig. 10, it becomes evident that the segmentation effectiveness of angles 2nd and 5th is not as pronounced as that of angles 8th and 11th. While angles 8th and 11th exhibit similar segmentation outcomes at a distance, angle 8 performs better in close proximity.

4.5.3 Analysis on coordinate fusion

To examine the impact of different perspectives on various categories, we perform a coordinate-level fusion of the two perspectives. The combined result of these perspectives is illustrated in Fig. 11. As shown in the figure, the two viewing coordinates are fused using mixing ratios of 0, 0.3, 0.5, 0.7 and 1.0, from top to bottom.

Assuming that the perspective produced by the central projection is referred to as ‘view1,’ the coordinates of each point under this projection are denoted as (x_1, y_1) . Similarly, for the perspective produced by the other projection (presumably ‘view2’), the coordinates of each point are (x_2, y_2) .

The hybrid coordinates can be calculated as $(w * x_1 + (1 - w) * x_2, w * y_1 + (1 - w) * y_2)$, where w represents the mixing ratio for the central projection. We also conducted experiments with varying values of w between 0.4 and 0.6 to observe the effect of randomness on the segmentation outcome. The experimental results are presented in Table 4. This experiment was conducted on the SemanticKITTI validation set with a resolution of $64 * 512$ pixels. We demote the results generated by mixing ratios of 0, 0.3, 0.5, 0.7 and 1 and the random ratio as ‘h00,’ ‘h03,’ ‘h05,’ ‘h07,’ ‘h10’ and ‘h_rand,’ respectively.

Table 4 show that with a low addition of parallel viewing angles, the segmentation effectiveness improves in the motorcycle and person categories. As more parallel viewing angles are added, categories like vegetation, building and pole show improved performance. This improvement is due to the fact that these objects themselves have larger areas or greater heights, and the height compression caused by the increase in angle has limited impact. Categories like road and parking experience a decrease in performance as the integration proportion increases, attributed to the reduced proportion of nearby roads. Overall, randomness has minimal impact on the segmentation effect.

Upon analyzing the results based on distance as shown in Fig. 12, it is evident that an increase in the parallel projection ratio leads to an enhancement in the segmentation effect at greater distances.

4.5.4 Analysis on the design of joint training method

To better harness the features projected by parallel viewing angles and enhance the segmentation effectiveness of nearby objects, we explore various training approaches. We begin by

Table 5 Quantitative results on SemanticKITTI dataset for different training methods

Approach	vehicles					grounds							sign								
	Car	Bicycle	Motorcycle	Truck	Other	Person	Bicyclist	Motorcyclist	Road	Parking	Sidewalk	Other grounds	Building	Fence	Vegetation	Trunk	Terrain	Pole	Traffic	mAcc	mIoU
add_channel_viewall	83.9	23.0	28.5	5.3	28.4	46.6	58.5	0.0	86.2	34.8	67.3	0.9	83.6	38.8	38.2	78.2	47.2	63.5	38.7	84.7	44.8
different_batch_view1	88.6	38.2	49.3	64.0	32.4	46.9	72.0	0.0	92.9	48.1	77.4	0.0	81.2	51.5	82.1	48.3	73.0	41.5	39.4	88.6	54.0
different_batch_view2	83.9	26.1	27.4	37.6	35.0	41.8	65.4	0.0	86.3	35.2	67.0	0.4	83.2	40.1	80.4	47.1	67.0	40.4	39.2	85.6	47.6
same_batch_view1	89.0	38.0	44.0	60.7	37.7	50.5	73.3	0.0	93.6	49.0	79.0	0.1	82.8	51.0	83.0	52.6	73.5	47.3	41.1	89.3	55.1
same_batch_view2	83.9	26.0	28.7	25.1	29.7	44.2	61.0	0.0	87.3	41.1	69.3	0.1	85.0	42.3	82.2	51.2	69.9	43.8	38.6	86.9	47.9

The bold text represents the best-performing model in each category (column)

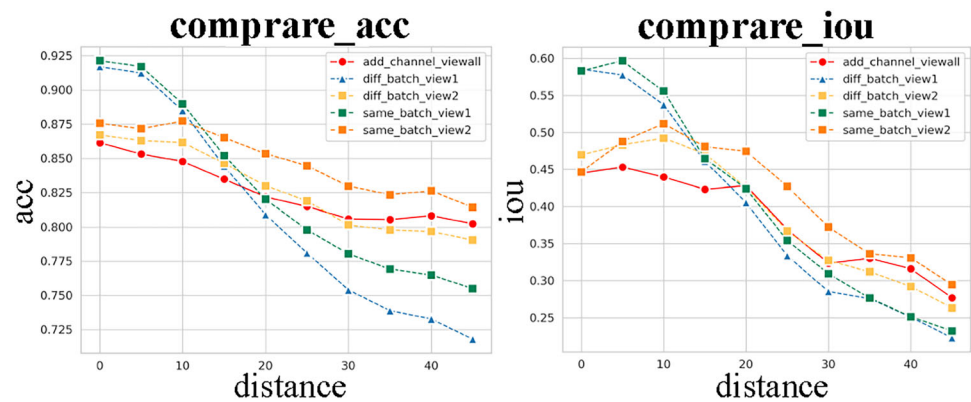
considering the amalgamation of the two graphs to augment the number of network channels. Subsequently, we examine the generation of training data with different viewing angles using the same ratio. Lastly, we contemplate the inclusion of different viewing angles within the same training batch. We denote the method of concatenating the two views during training to increase input channels as ‘add_channel_viewall.’ We also conducted experiments by training with the two views placed in separate batches. We make predictions using the two views separately, denoted as ‘different_batch_view1’ and ‘different_batch_view2.’ Lastly, we conducted experiments by training with the two views placed in the same batches. We make predictions using the two views separately, denoted as ‘same_batch_view1’ and ‘same_batch_view2.’

Observing the experimental results as shown in Table 5, it becomes evident that the approach of including both projections within a batch yields the best results. Analyzing the results based on distance as shown in Fig. 13, it is apparent that this hybrid training method not only enhances the segmentation effect for nearby distances but also improves segmentation performance at greater distances.

5 Conclusion

In summary, our method integrates parallel projection into semantic segmentation of point clouds for autonomous driving. We optimized incline angles for balanced scene representation and used joint training to enhance target features. Experiments on SemanticKITTI show improved recognition of distant objects like pedestrians and vehicles, along with better segmentation of vertical structures. Our approach also demonstrates the effectiveness of our angle selection and the enhanced model performance through joint training. Currently, there are center projection methods with density but scale imbalance, as well as BEV (bird’s-eye view) projection methods with scale balance but excessive sparsity. We are exploring new projection methods in this article and analyzing this approach.

Fig. 13 A comparison of segmentation accuracy and mIoU for different training methods across 5 distance segments



Data Availability The datasets used in this paper are public datasets and can be obtained by contacting the relevant providers. The data that support the findings of this study are available from the authors upon reasonable request.

Declarations

Conflict of interest We declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Ahmed, S.M., Tan, Y.Z., Chew, C., Mamun, A.A., Wong, F.S.: Edge and corner detection for unorganized 3D point clouds with application to robotic welding. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 7350–7355 (2018)
2. Alnaggar, Y.A., Afifi, M., Amer, K., ElHelw, M.: Multi projection fusion for real-time semantic segmentation of 3D LiDAR point clouds. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1799–1808 (2021)
3. Alonso, I., Riazuelo, L., Montesano, L., Murillo, A.C.: 3D-MiniNet: learning a 2D representation from point clouds for fast and efficient 3D LiDAR semantic segmentation. *IEEE Robotics Autom. Lett.* **5**(4), 5432–5439 (2020)
4. Ando, A., Gidaris, S., Bursuc, A., Puy, G., Boulch, A., Marlet, R.: RangeViT: Towards vision transformers for 3d semantic segmentation in autonomous driving. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5240–5250 (2023)
5. Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Gall, J., Stachniss, C.: Towards 3D LiDAR-based semantic scene understanding of 3D point cloud sequences: The semantickitti dataset. *Int. J. Robotics Res.* **40**(8–9) (2021)
6. Beltrán, J., Guindel, C., Moreno, F.M., Cruzado, D., García, F., de la Escalera, A.: BirdNet: a 3D object detection framework from LiDAR information. In: International Conference on Intelligent Transportation Systems (ITSC), pp. 3517–3523 (2018)
7. Chen, C., Wang, Y., Chen, H., Yan, X., Ren, D., Guo, Y., Xie, H., Wang, F.L., Wei, M.: GeoSegNet: point cloud semantic segmentation via geometric encoder-decoder modeling. *The Visual Computer*, pp. 1–15 (2023)
8. Chen, K., Oldja, R., Smolyanskiy, N., Birchfield, S., Popov, A., Wehr, D., Eden, I., Pehserl, J.: MVLidarNet: real-time multi-class scene understanding for autonomous driving using multiple views. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2288–2294 (2020)
9. Cheng, H., Han, X., Xiao, G.: TransRVNet: LiDAR semantic segmentation with transformer. *IEEE Trans. Intell. Transp. Syst.* **24**(6), 5895–5907 (2023)
10. Choy, C.B., Gwak, J., Savarese, S.: 4D spatio-temporal convnets: Minkowski convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3075–3084 (2019)
11. Cortinhal, T., Tzelepis, G., Aksoy, E.E.: SalsaNext: Fast, uncertainty-aware semantic segmentation of LiDAR point clouds. In: Proceedings of the International Symposium on Visual Computing (ISVC), vol. 12510, pp. 207–222 (2020)
12. Fan, S., Dong, Q., Zhu, F., Lv, Y., Ye, P., Wang, F.: SCF-Net: Learning spatial contextual features for large-scale point cloud segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 14504–14513 (2021)
13. Feng, T., Wang, W., Wang, X., Yang, Y., Zheng, Q.: Clustering based point cloud representation learning for 3d analysis. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 8249–8260 (2023)
14. Graham, B., Engelcke, M., van der Maaten, L.: 3D semantic segmentation with submanifold sparse convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9224–9232 (2018)
15. He, Q., Peng, J., Jiang, Z., Hu, X., Zhang, J., Nie, Q., Wang, Y., Wang, C.: PointSeg: A training-free paradigm for 3D scene segmentation via foundation models. *CoRR arXiv:2403.06403* (2024)
16. Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Improving neural networks by preventing co-adaptation of feature detectors. *CoRR arXiv:1207.0580* (2012)
17. Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A.: RandLA-Net: Efficient semantic segmentation of large-scale point clouds. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 11105–11114 (2020)
18. Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A.: Learning semantic segmentation of large-scale point clouds with random sampling. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**(11), 8338–8354 (2022)
19. Huang, J., You, S.: Point cloud labeling using 3D convolutional neural network. In: Proceedings of the 23rd International Conference on Pattern Recognition (ICPR), pp. 2670–2675 (2016)
20. Jiang, J., Lu, X., Wanli, Ouyang, Wang, M.: Unsupervised contrastive learning with simple transformation for 3D point cloud data. *Vis Comput* 1–18 (2023)
21. Kong, L., Liu, Y., Chen, R., Ma, Y., Zhu, X., Li, Y., Hou, Y., Qiao, Y., Liu, Z.: Rethinking range view representation for lidar segmen-

- tation. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 228–240 (2023)
22. Landrieu, L., Simonovsky, M.: Large-scale point cloud semantic segmentation with superpoint graphs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4558–4567 (2018)
 23. Landrieu, L., Simonovsky, M.: Large-scale point cloud semantic segmentation with superpoint graphs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4558–4567 (2018)
 24. Li, Z., Wang, J., Qu, X., Xiao, J.: 3D point cloud segmentation for complex structure based on PointSIFT. In: Proceedings of the Pattern Recognition and Computer Vision (PRCV), vol. 12305, pp. 660–670 (2020)
 25. Meng, Q., Wang, W., Zhou, T., Shen, J., Jia, Y., Gool, L.V.: Towards a weakly supervised framework for 3d point cloud object detection and annotation. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**(8), 4454–4468 (2022)
 26. Milioto, A., Vizzo, I., Behley, J., Stachniss, C.: RangeNet ++: fast and accurate LiDAR semantic segmentation. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4213–4220. IEEE (2019)
 27. Phan, A.V., Nguyen, M.L., Nguyen, Y.L.H., Bui, L.T.: DGCNN: a convolutional neural network over large-scale labeled graphs. *Neural Netw.* **108**, 533–543 (2018)
 28. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: PointNet: deep learning on point sets for 3D classification and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 77–85 (2017)
 29. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: PointNet++: deep hierarchical feature learning on point sets in a metric space. In: Advances in Neural Information Processing Systems, pp. 5099–5108 (2017)
 30. Radi, H., Ali, W.: VolMap: A real-time model for semantic segmentation of a LiDAR surrounding view. *CoRR arXiv: 1906.11873* (2019)
 31. Redmon, J., Farhadi, A.: YOLOv3: an incremental improvement. *CoRR* (2018)
 32. Rethage, D., Wald, J., Sturm, J., Navab, N., Tombari, F.: Fully-convolutional point networks for large-scale point clouds. In: Proceedings of the European conference on computer vision (ECCV), pp. 625–640 (2018)
 33. Rosu, R.A., Schütt, P., Quenzel, J., Behnke, S.: LatticeNet: fast spatio-temporal point cloud segmentation using permutohedral lattices. *Auton. Robots* **46**(1), 45–60 (2022)
 34. Shi, W., Caballero, J., Huszar, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1874–1883 (2016)
 35. Simon, M., Milz, S., Amende, K., Gross, H.: Complex-YOLO: an Euler-region-proposal for real-time 3D object detection on point clouds. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 197–209 (2018)
 36. Tang, H., Liu, Z., Zhao, S., Lin, Y., Lin, J., Wang, H., Han, S.: Searching efficient 3D architectures with sparse point-voxel convolution. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 685–702 (2020)
 37. Tatarchenko, M., Park, J., Koltun, V., Zhou, Q.: Tangent convolutions for dense prediction in 3D. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3887–3896 (2018)
 38. Tateno, K., Tombari, F., Navab, N.: Real-time and scalable incremental segmentation on dense SLAM. In: Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS), pp. 4465–4472 (2015)
 39. Thomas, H., Qi, C.R., Deschaud, J., Marcotegui, B., Goulette, F., Guibas, L.J.: KPConv: Flexible and deformable convolution for point clouds. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 6410–6419 (2019)
 40. Wang, Y., Chao, W., Garg, D., Hariharan, B., Campbell, M.E., Weinberger, K.Q.: Pseudo-LiDAR from visual depth estimation: Bridging the gap in 3D object detection for autonomous driving. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8445–8453 (2019)
 41. Wu, B., Wan, A., Yue, X., Keutzer, K.: SqueezeSeg: Convolutional neural nets with recurrent CRF for real-time road-object segmentation from 3D LiDAR point cloud. In: Proceedings of the International Conference on Robotics and Automation (ICRA), pp. 1887–1893 (2018)
 42. Wu, B., Zhou, X., Zhao, S., Yue, X., Keutzer, K.: SqueezeSegV2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a LiDAR point cloud. In: Proceedings of the International Conference on Robotics and Automation (ICRA), pp. 4376–4382 (2019)
 43. Xu, C., Wu, B., Wang, Z., Zhan, W., Vajda, P., Keutzer, K., Tomizuka, M.: SqueezeSegV3: Spatially-adaptive convolution for efficient point-cloud segmentation. In: Proceedings of the European Conference on Computer Vision (ECCV), vol. 12373, pp. 1–19 (2020)
 44. Xu, J., Zhang, R., Dou, J., Zhu, Y., Sun, J., Pu, S.: RPVNet: A deep and efficient range-point-voxel fusion network for LiDAR point cloud segmentation. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 16004–16013 (2021)
 45. Xu, Y., Gao, J., Zheng, C., Zheng, C., Zhang, R., Cui, S., Li, Z.: 2DPASS: 2D priors assisted semantic segmentation on LiDAR point clouds. In: Proceedings of the European conference on computer vision (ECCV), pp. 677–695 (2022)
 46. Yin, J., Fang, J., Zhou, D., Zhang, L., Xu, C., Shen, J., Wang, W.: Semi-supervised 3d object detection with proficient teachers. In: Proceedings of the European conference on computer vision (ECCV), pp. 727–743 (2022)
 47. Yin, J., Zhou, D., Zhang, L., Fang, J., Xu, C., Shen, J., Wang, W.: Proposalcontrast: Unsupervised pre-training for lidar-based 3d object detection. In: S. Avidan, G.J. Brostow, M. Cissé, G.M. Farinella, T. Hassner (eds.) Proceedings of the European conference on computer vision (ECCV), pp. 17–33 (2022)
 48. Zeng, Y., Hu, Y., Liu, S., Ye, J., Han, Y., Li, X., Sun, N.: RT3D: real-time 3-D vehicle detection in LiDAR point cloud for autonomous driving. *IEEE Robot. Autom. Lett.* **3**(4), 3434–3440 (2018)
 49. Zhang, Y., Zhou, Z., David, P., Yue, X., Xi, Z., Gong, B., Foroosh, H.: PolarNet: An improved grid representation for online LiDAR point clouds semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9598–9607 (2020)
 50. Zhao, Y., Ma, X., Hu, B., Zhang, Q., Ye, M., Zhou, G.: A large-scale point cloud semantic segmentation network via local dual features and global correlations. *Comput. Graph.* **111**, 133–144 (2023)
 51. Zhu, X., Zhou, H., Wang, T., Hong, F., Ma, Y., Li, W., Li, H., Lin, D.: Cylindrical and asymmetrical 3D convolution networks for LiDAR segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9939–9948 (2021)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the

author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.



Haoying Zhao received the B.Sc degree from East China Normal University. She is currently a Ph.D. candidate in the School of Computer Science and Technology, East China Normal University, Shanghai, China. Her research interests include computer vision and machine learning.



Aimin Zhou received the B.Sc. and M.Sc. degrees from Wuhan University, China, in 2001 and 2003, respectively, and the Ph.D. degree from University of Essex, UK, in 2009. He is currently a Professor with Shanghai Institute of AI for Education, and School of Computer Science and Technology, East China Normal University, Shanghai, China. He has authored over 80 peer-reviewed papers. His current research interests include evolutionary optimization and learning, explainable machine learning and intelligent education. Dr. Zhou is an Associate Editor of *Swarm and Evolutionary Computation* and an Editorial Board Member of *Complex and Intelligent Systems* and *Chinese Journal of Electronics*.