Metadata of the chapter that will be visualized in SpringerLink

Book Title	PRICAI 2024: Trends in Artificial Intelligence	
Series Title		
Chapter Title	S2A-Attention for Multimodal 3D Semantic Segmentation Using LiDAR and Cameras in Autonomous Driving	
Copyright Year	2025	
Copyright HolderName	The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd.	
Author	Family Name	**Zhang**
	Particle	
	Given Name	**Siyu**
	Prefix	
	Suffix	
	Role	
	Division	
	Organization	Southwest University
	Address	Chongqing, China
	Email	swu040423@email.swu.edu.cn
Author	Family Name	**Guo**
	Particle	
	Given Name	**Yifu**
	Prefix	
	Suffix	
	Role	
	Division	
	Organization	South China Normal University
	Address	Guangzhou, Guangdong, China
	Division	
	Organization	University of Aberdeen
	Address	Aberdeen, UK
	Email	20223801024@m.scnu.edu.cn
		u08yg22@abdn.ac.uk
Author	Family Name	**Lu**
	Particle	
	Given Name	**Yuquan**
	Prefix	
	Suffix	
	Role	
	Division	
	Organization	South China Normal University
	Address	Guangzhou, Guangdong, China
	Division	
	Organization	University of Aberdeen

	Address	Aberdeen, UK
	Email	20223802023@m.scnu.edu.cn
		u22yl22@abdn.ac.uk
Author	Family Name	**Zeng**
	Particle	
	Given Name	**Kun**
	Prefix	
	Suffix	
	Role	
	Division	
	Organization	Sun Yat-sen University
	Address	Guangzhou, Guangdong, China
	Email	zengk29@mail2.sysu.edu.cn
Corresponding Author	Family Name	**He**
	Particle	
	Given Name	**Chao**
	Prefix	
	Suffix	
	Role	
	Division	
	Organization	South China Normal University
	Address	Guangzhou, Guangdong, China
	Email	chaohe@m.scnu.edu.cn
Corresponding Author	Family Name	**Cai**
	Particle	
	Given Name	**Lihua**
	Prefix	
	Suffix	
	Role	
	Division	
	Organization	South China Normal University
	Address	Guangzhou, Guangdong, China
	Email	lee.cai@m.scnu.edu.cn

Abstract	Adding LiDAR for an autonomous driving system will complement the weaknesses of a camera-only solution and enhance its robustness. To fully exploit the multimodal advantage, existing works have proposed various multimodal fusion algorithms to effectively combine LiDAR and camera data for scene and object recognition through 3D semantic segmentation. However, most of these methods leverage softmax-based attention modules for intra-modal feature encoding, and early fusion for inter-modal feature learning, leading to excessive computations and therefore higher latency in semantic segmentation. To mitigate this challenge, we propose the Semantic Segmentation (S2) Agent attention module for 3D semantic segmentation in autonomous driving system using LiDAR and camera. Intra-modal encoding is fully explored instead of early fusion using feature concatenation. We adopt a mid fusion strategy to further reduce computations. Experiments using open benchmark datasets nuScenes and Semantic KITTI show comparable or even better mIoUs than state-of-the-art baseline methods while obtaining better latency performance when compared to the most recent MSeg3D algorithm.

Keywords (separated by '-')	Semantic Segmentation - LiDAR - Autonomous Driving - Multi-modal Fusion

S2A-Attention for Multimodal 3D Semantic Segmentation Using LiDAR and Cameras in Autonomous Driving

Siyu Zhang[2], Yifu Guo[1,4], Yuquan Lu[1,4], Kun Zeng[3], Chao He[1(✉)], and Lihua Cai[1(✉)]

[1] South China Normal University, Guangzhou, Guangdong, China
{20223801024,20223802023,chaohe,lee.cai}@m.scnu.edu.cn
[2] Southwest University, Chongqing, China
swu040423@email.swu.edu.cn
[3] Sun Yat-sen University, Guangzhou, Guangdong, China
zengk29@mail2.sysu.edu.cn
[4] University of Aberdeen, Aberdeen, UK
{u08yg22,u22yl22}@abdn.ac.uk

Abstract. Adding LiDAR for an autonomous driving system will complement the weaknesses of a camera-only solution and enhance its robustness. To fully exploit the multimodal advantage, existing works have proposed various multimodal fusion algorithms to effectively combine LiDAR and camera data for scene and object recognition through 3D semantic segmentation. However, most of these methods leverage softmax-based attention modules for intra-modal feature encoding, and early fusion for inter-modal feature learning, leading to excessive computations and therefore higher latency in semantic segmentation. To mitigate this challenge, we propose the Semantic Segmentation (S2) Agent attention module for 3D semantic segmentation in autonomous driving system using LiDAR and camera. Intra-modal encoding is fully explored instead of early fusion using feature concatenation. We adopt a mid fusion strategy to further reduce computations. Experiments using open benchmark datasets nuScenes and Semantic KITTI show comparable or even better mIoUs than state-of-the-art baseline methods while obtaining better latency performance when compared to the most recent MSeg3D algorithm.

Keywords: Semantic Segmentation · LiDAR · Autonomous Driving · Multi-modal Fusion

1 Introduction

Autonomous driving is a pivotal field, in which the integration of 2D imagery with LiDAR 3D point clouds is instrumental in providing a comprehensive and robust understanding of driving scenes. On one hand, images captured by multiple cameras offer rich color and texture information about objects and environment, and have been leveraged as a mature scene understanding solution through

S. Zhang and Y. Guo—These authors contributed equally to this work.

Q. Bai et al. (Eds.): PRICAI 2024, LNAI 15284, pp. 1–12, 2025.
https://doi.org/10.1007/978-981-96-0125-7_21

semantic segmentation way before LiDAR became widely available [12,23,25]. On the other hand, point clouds generated by LiDAR can capture precise spatial information about objects, and is insusceptible to changing time and weather conditions such as darkness during night time and invisibility caused by thunderstorm. Two main categories of representation learning algorithms in point cloud data are direct methods [3,8,26] and projection-based methods [6,21].

To fully explore the superiority of combining LiDAR and camera data for semantic segmentation, deep learning-based feature encoding [5,16] and attention-driven modality fusion [9–11] have served as the mainstream methods in recent years. Specifically, various backbone networks have been adopted to extract unimodal representation from either 2D images or 3D point clouds. For example, Sun et al. proposed a high-resolution representation learning method HRNet to extract image features [16]; while Çiçek, Özgün et al. proposed the 3D U-Net that can be adopted for feature extraction in 3D point clouds [5]. With these unimodal representations, Li et al. proposed a semantic segmentation method MSeg3D for LiDAR and camera data fusion [10]. MSeg3D is able to address three common challenges existing in multi-modal segmentation model. Taking advantage of different representations in point cloud data, Liu et al. proposed the UniSeg algorithm for more robust and accurate perception in scene recognition for autonomous driving [11].

Despite the above progress, computational efficiency still has room for further improvements First, due to the massive amount of data in point clouds and the adoption of the softmax-based attention modules, the process of multimodal feature extraction and integration becomes computationally expensive [7]. In addition, during feature learning, the multimodal techniques often concatenate encodings from different layers within each modality. Although this will enrich the representations for final inference, it greatly expands the dimensions for all the input vectors to the attention modules, leading to significant increase in computational burden [14]. At the same time, early fusion of inter-modal features not only generates additional computation overhead (i.e., larger input vectors for cross-modal attention modules), but also reduces the efficiency for modal fusion, as the cross-modal attention module has to learn encodings from much lower level feature space.

To address the aforementioned problems, we adapt the agent attention module from [7] for general computer vision tasks to the 3D semantic segmentation task using LiDAR and camera. The simplified agent attention module is called **S**emantic **S**egmentation **A**gent (S2A) attention. The introduction of agent tokens within S2A attention module significantly reduces the computation complexity when compared to softmax-based attention (i.e., the typical attention module proposed in [17]). We stack multiple blocks of S2A attention modules for unimodal features extraction using input features learned by backbone neural networks instead of concatenating features from each modality to lower the dimensions of input vectors to each attention module. This intra-modality feature encoding strategy is followed by the mid fusion strategy enlightened by [14] for the voxel features and image features. With the above measures, we are able to reduce the segmentation latency while achieving comparable or even better

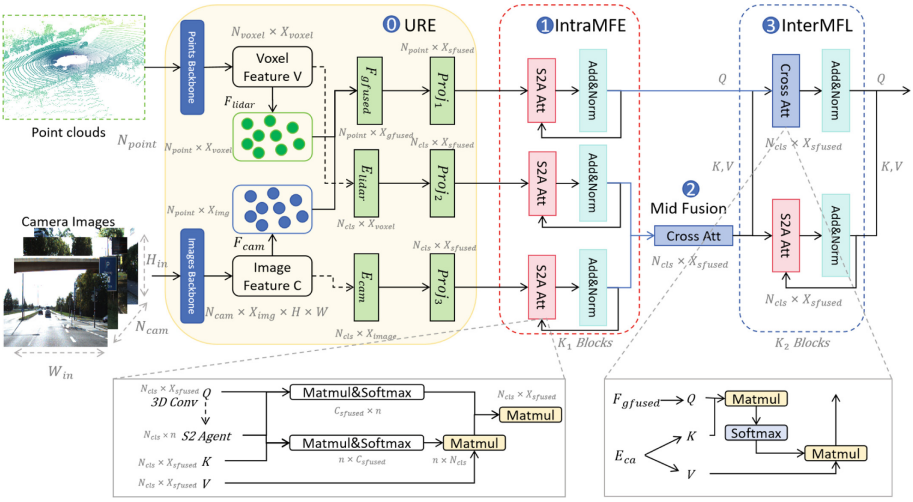

Fig. 1. Framework Overview. 0. Unimodal representation extraction (URE) obtains embeddings and spatial features of different modalities. 1. Intra-modality Feature Encoding (IntraMFE) aggregate intra-modal features by stacking S2A attention blocks. 2. Mid-Fusion fuses embeddings from different modalities via cross attention instead of concatenation to reduce computational loss. 3. Inter-modality Feature Learning (InterMFL) performs information fusion of all modality streams.

segmentation performance when compared to the state-of-the-art 3D semantic segmentation baseline methods.

Our contributions in this work can be summarized as follows: 1) We propose a new agent attention module called **S**emantic **S**egmentation **A**gent (S2A) attention to reduce computation complexity for 3D semantic segmentation using LiDAR(s) and cameras. 2) We combine the intra-modality feature encoding strategy using S2A attention modules with the mid fusion strategy to improve segmentation speed in an end-to-end 3D semantic segmentation network for autonomous driving. 3) We evaluate our proposed network in two public benchmark datasets, nuScenes [2] and Semantic KITTI [1], and attain comparable performance in mIoU when compared to selective baselines. However, executing in the same hardware environment, we obtain significantly better latency when compared to the MSeg3D algorithm.

2 Related Work

2.1 3D Semantic Segmentation for Autonomous Driving

2D image-based semantic segmentation algorithms for object and scene recognition were well researched in computer vision with an early focus on CNN methods [12,21,23,25]. Wang et al. introduced the Pyramid Vision Transformer (PVT), which possesses advantages from both CNN and Transformer, and can serve as a direct replacement for CNN backbones [18]. Xia et al. proposed a novel deformable self-attention module, the flexibility of which enables it to

focus on relevant regions and capture more informative features [20]. Based on this module, they built a general backbone model called Deformable Attention Transformer for both image classification and dense prediction tasks.

The increasing adoption of 3D LiDAR in EVs led to a series of works that proposed 3D semantic segmentation deep learning algorithms on 3D point clouds.

LiDAR-only Methods. Charles et al. proposed the PointNet, which directly models each point in a point cloud for 3D classification and segmentation [3]. Zhou et al. created Cylinder3D, a 3D convolution-based framework that exploits the 3D topology relations and structures of driving-scene point clouds [26]. Cortinhal et al. introduced SalsaNext, a real time algorithm with an encoder-decoder structure for uncertainty-aware semantic segmentation [6]. Hu et al. proposed RandLA-Net, an efficient and lightweight network that can infer per-point semantics for large-scale point clouds using random point sampling [8].

LiDAR+Camera Methods. Krispel et al. proposed FuseSeg, a framework that combine LiDAR and RGB data to segment LiDAR point clouds [9]. Liu et al. leveraged all three different representations, including the point-, voxel-, and range-views of LiDAR point clouds, and RGB images from cameras to construct the UniSeg network [11]. UniSeg is designed to carry out both semantic and panoptic segmentation simultaneously. Lastly, Li et al. proposed the MSeg3D framework to address three common challenges in fusing LiDAR and camera data, namely modality heterogeneity, limited sensor field of view intersection, and multi-modal data augmentation [10].

2.2 Computational Efficiency in LiDAR-Based 3D Semantic Segmentation

Computational efficiency is of critical importance for LiDAR-based 3D semantic segmentation in an autonomous driving setting [4,13,15,19,24].

Zermas et al. proposed a fast and low complexity segmentation pipeline for 3D point cloud semantic segmentation with improved running time and comparable segmentation performance when compared with multiple baseline methods [24]. Wang et al. built the PointSeg real-time semantic segmentation method on 3D LiDAR point clouds based on the light-weight SqueezeNet with 90 frames per second (FPS) on a single GPU [19]. Milioto et al. propose the Rangenet++, a fast and accurate LiDAR semantic segmentation algorithm with sensor frame rate [13]. Chen et al. proposed the RangeSeg network, in which a shared encoder backbone with two range dependent decoders to improve computation efficiency as the heavy decoder only focuses on distant objects, and the light decoder processes the entire image [4]. Park et al. proposed the PCSCNet for fast 3D semantic segmentation on LiDAR point cloud using point convolution and sparse convolution network [15]. Among the state-of-the-art real-time models in semantic segmentation, the authors was able to show better performance in LiDAR point cloud inference.

Recognizing that voxel and fusion-based semantic segmentation models such as the newly proposed MSeg3D [10] have maintained the best overall segmentation performance in mIoU, in this work, we attempt to make improvements to

these models so that we can achieve comparable or better segmentation performance with sensor frame rate speed.

3 Method

3.1 Problem Formulation

Let $\{L_{in}, C_{in}\}$ be a multi-modal sample, where $L_{in} \in \mathbb{R}^{N_{point} \times X_{in}}$ denotes a LiDAR point cloud containing N_{point} points, each associated with X_{in}-dimensional input features such as 3D coordinates and reflectance; $C_{in} \in \mathbb{R}^{N_{cam} \times 3 \times H_{in} \times W_{in}}$ represents RGB images captured by N_{cam} cameras. With the aid of sensor calibration, a 3D point with coordinate (x, y, z) can be mapped to the c-th local camera's image plane, resulting in a pixel coordinate (u, v). The 3D semantic segmentation task is to assign one of the N_{cls} semantic categories to each individual point within the 3D point cloud.

3.2 Unimodal Representation Extraction (URE)

The Unimodal Representation Extraction (URE) module is a replaceable and flexible component in our proposed framework, and can accommodate different multimodal data as long as it possesses multiple streams of input representations. Each stream of the representations needs not be feature encodings that are extracted from one single data modality. In our current work, we adopt the MSeg3D [10] single modality feature extraction pipeline and the Geometry-based Feature Fusion module. It uses two different backbone networks to extract LiDAR and Camera unimodal features as initial model inputs; and a Geometry-based Feature Fusion module to generate enhanced representation for field of view intersection between LiDAR and cameras. Each unimodal feature representation is also semantically enhanced by projecting it into a N_{cls} dimensional space, with each dimension representing a semantic category. More details can be referred to [10], but below we provide the necessary details relevant to our current network implementation.

Given the input $\{L_{\text{in}}, C_{\text{in}}\}$, the voxel-based LiDAR feature representation and the camera feature representation extracted from the selected backbone network can be denoted as $V \in \mathbb{R}^{N_{\text{voxel}} \times X_{\text{voxel}}}$ and $C \in \mathbb{R}^{N_{\text{cam}} \times 3 \times H_{\text{in}} \times W_{\text{in}}}$, respectively, where X_{voxel} is the LiDAR point channel dimension.

Then the extracted voxel features are devoxelized to F_{lidar} point by point, while the intercepted camera pixels are identified using bilinear interpolation, and the camera features F_{cam} are subsequently generated following similar process. We have $F_{\text{lidar}} = [f_{\text{lidar},i}]_{i=1}^{i=N_{\text{point}}} \in \mathbb{R}^{N_{\text{voxel}} \times X_{\text{voxel}}}$ and $F_{\text{cam}} = [f_{\text{cam,i}}]_{i=1}^{i=N_{\text{point}}} \in \mathbb{R}^{N_{\text{point}} \times X_{\text{img}}}$, where X_{img} is the image channel dimension.Using fully connected layers F_{lidar} and F_{cam}, we project $f_{\text{lidar},i}$ and $f_{\text{cam},i}$ into X_{int}-dimensional spaces. These projected features are then concatenated and fused through another MLP that has X_{gfused} output channels to obtain F_{gfused}.

To obtain unimodal semantic embeddings for LiDAR, a distribution matrix $D_{\text{lidar}} \in (0, 1)^{N_{\text{cls}} \times N_{\text{voxel}}}$ is derived from an intermediate segmentation $D'_{\text{lidar}} = MLP(V)$ and normalized using spatial softmax. Following the same procedure,

we obtain D_{cam} for camera. And finally we have $E_{\text{lidar}} = D_{\text{lidar}} \times V$ and $E_{\text{cam}} = D_{\text{cam}} C'$.

3.3 Semantic Segmentation Agent Attention

Inspired by the proposed agent attention in [7], which is designed for general computer vision tasks, we propose a more lightweight Semantic Segmentation Agent (S2Agent) attention module for autonomous driving tasks. The left detailed view in Fig. 1 shows the structure of S2Agent attention block. In S2Agent attention, an agent matrix A (with dimension $n << X$) is used to aggregate information from Q, as shown in Eq. 1, reducing matrix multiplication complexity. We adopt the strategy of 3D Convolution and Pooling to ensure a larger receptive field and richer semantic information on voxel embeddings.

$$A = \text{3D-}conv(Q) + Pooling(Q) \tag{1}$$

S2Agent attention enables the model to achieve a linear computational complexity of $O(N_{cls} n X)$ relative to the number of input features X, where n is much smaller than X. In contrast, the original softmax attention has a computational complexity of $O(N_{cls} X^2)$. We reduce the computation time while ensuring the ability to extract global information. As shown in Fig. 1, we apply S2Agent to extract features from different modalities, as shown in Eq. 2.

$$S2Agent\,(E) = Softmax\left(Q A_E^T\right) \cdot Softmax\left(A_E K^T\right) \cdot V \tag{2}$$

where E represents the input encoding sequence, $Q, K, V \in \mathbb{R}^{N_{cls} \times X_{sfused}}$, and $A \in \mathbb{R}^{n \times X_{sfused}}$.

Previous 3D semantic segmentation research typically emphasized Softmax attention for its strong feature extraction abilities. By seamlessly integrating Softmax and linear attention, our S2Agent attention inherits the strengths of both, achieving lower computational complexity and enhanced model expressiveness simultaneously.

3.4 Intra-Modality Feature Encoding (IntraMFE)

Most existing works fuse unimodal representations early before fully exploring a compact intra-modality representation. For example, in MSeg3D, each stream of modality representations from the URE module is directly fused with each other using cross-modal attention. A more compact encoding will significantly reduce the computation burden, particularly with attention block sequences. In order to improve learning efficiency and mitigate computation complexity, we learn intra-modality embeddings through stacking the S2Agent attention blocks. The process is expressed in Eq. 3 and 4, where h indicates the h-th S2Agent block, and modal indicates the underlying modality, and in our case could be replaced by either LiDAR or Camera.

$$E_{\text{modal}}^{h'} = MHS2AA\left(E_{\text{modal}}^{h} \oplus E_{\text{modal}}^{h-1}\right) \tag{3}$$

$$E_{\text{modal}}^{h+1} = Norm\left(E_{\text{modal}}^{h'}\right) \tag{4}$$

3.5 Mid-Fusion for LiDAR and Camera

In multi-modal task, a common paradigm is to have the early layers of the network focus on unimodal processing, and only introduce cross-modal connections in the later layers. This approach is conceptually intuitive because lower-level layers typically handle low-level features, while higher-level layers focus on learning semantic concepts. For instance, low-level visual features such as edges and corners in point clouds do not directly correspond to image features, so early fusion with images may not be beneficial. Enlightened by [14], we fused LiDAR and Camera representations using mid-fusion strategy with one single cross attention, which is shown in 5, where E_{hlidar} and E_{hcam} are the output embeddings from IntraMFE module.

$$E_{ca} = MHCA\left(E_{hlidar}, E_{hcam}, E_{hcam}\right) \tag{5}$$

3.6 Inter-Modality Feature Learning (InterMFL)

Many real world problems with multimodal sensor solutions have a hierarchical fusion structure among sensors, and are similar to our current problem. Specifically, we employ mid-fusion strategy to fuse homogeneous representations generated by the IntraMFE module. Then the mid-fusion output will need to be further fused with other representation streams that can either be from a different view or entirely different modalities. Our propose Intermodality Feature Learning (InterMFL) module aims to solve the global fusion task.

We apply both Multi-head cross-attention and S2Agent attention to fuse the field of view interception features from IntraMFE F_{hfused}, and the output features from Mid-Fusion E_{ca}. This process is illustrated in Fig. 1: InterMFL and also expressed in Eq. 6 and 7, where h indicates the h-th block in InterMFL.

$$F_{ca}^{h+1} = Norm\left(MHCA\left(F_{ca}^{h}, E_{ca}^{h}, E_{ca}^{h}\right)\right) \tag{6}$$

$$E_{ca}^{h+1} = Norm\left(MHS2AA\left(E_{ca}^{h} \oplus E_{ca}^{h-1}\right)\right) \tag{7}$$

4 Experiments

4.1 Datasets

nuScenes. The nuScenes [2] dataset consists of 28,130 training samples, 6,019 validation samples, and 6,008 testing samples. Each sample includes a sparse point cloud from a 32-beam LiDAR and RGB images from six cameras positioned around the vehicle. Due to differing vertical FOVs between the LiDAR and cameras, some points project below the images. Following the official protocol, the dataset contains 17 categories ($N_{\text{cls}} = 17$) with semantic annotations provided only for the point clouds.

SemanticKITTI. The SemanticKITTI [1] dataset is collected using a 64-beam LiDAR. As per the protocols in MSeg3D [10], sequences 00 to 10 except 08

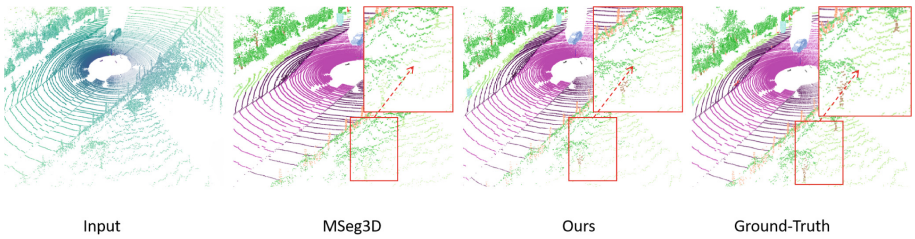

Fig. 2. LiDAr sample from the SemanticKITTI dataset. A detailed zoom-in view highlights the primary differences in classification performance between our proposed method and MSeg3D [10]. In this case, MSeg3D incorrectly classifies pedestrians (represented by brown points) on the road, while our method yields results that are significantly more consistent with the ground-truth annotations.

containing 19130 samples are used for training, while sequence 08 containing 4071 samples are used for validation. This dataset includes only the images from the front-view camera. The number of semantic categories N_{cls} is 20, and only the semantic annotations for the point clouds are provided.

4.2 Evaluation Metrics

Mean Intersection over Union (mIoU). mIoU is a standard metric for evaluating semantic segmentation models, as it offers a comparable single value across models and datasets and penalizes both false positives and negatives. It is the average of Intersection over Union (IoU) values for each class, measuring the overlap between predicted and ground truth segmentation masks. The formulas for IoU and mIoU are $IoU = \frac{TP}{TP+FP+FN}$ and $mIoU = \frac{1}{C}\sum_{c=1}^{C} IoU_c$ respectively, where $TP, FP,$ andFN are true positive, false positive, and false negative respectively, and C is the number of classes.

Latency. Latency refers to the time delay between the moment point clouds and images are captured and the moment the model produces an output in semantic segmentation tasks. This latency is critical because it directly affects the vehicle's ability to make timely and accurate decisions in real-time driving scenarios.

4.3 Implementation Details

We train our model under the same schedule: AdamW optimizer and one-cycle learning rate policy with division factor 10. Momentum ranges from 0.95 to 0.85, weight decay 0.01, maximum learning rate 0.01, and each batch contains 32 random samples distributed on 4 RTX 3090 GPUs with 24 epochs.

5 Results

As shown by Table 1 and 2, for the **NuScenes** and **SemanticKITTI** datasets respectively, our proposed method achieves better result on mIoU when compared to all selected single-modal methods. When compared to other multimodal

methods such as MSeg3D [10], our model is marginally better or comparable in mIoU. Specifically, we obtain 82.5 on mIoU, which is higher than MSeg3D's 81.4, while our latency is significantly lower than MSeg3D's at about one fourth on the NuScenes dataset. On the **SemanticKITTI** dataset, Our proposed method maintain comparable mIoU with MSeg3D. Note that in both datasets, UniSeg achieves the best performance in mIoU. This is likely due to its adoption of three different point cloud representations from LiDAR, which is potentially increasing latency. Due to lack of source code from UniSeg, we can not obtain latency for it.

Figure 2 provides an example point cloud scene with semantic segmentation results from MSeg3D and our proposed method. When compared to the ground truth, we can visually see the differences between the two methods. By examining the results, it is clear that the proposed method has higher discrimination on some small details, such as fences and pedestrians.

Table 1. Quantitative comparisons on the nuScenes dataset using per-class IoU and mIoU. The latency results except MSeg3D and our proposed method are directly adopted from the original articles if they are reported, and thus only serve as references as they were computed using different computing environments.

Methods	Modality	Barrier	Bicycle	Bus	Car	C-Vehicle	Motorcycle	Pedestrian	Traffic Cone	Trailer	Truck	D-Surface	Other	Sidewalk	Terrain	Manmade	Vegetation	mIoU	Latency(s)
PolarNet	L	72.2	16.8	77.0	86.5	51.1	69.7	64.8	54.1	69.7	63.5	96.6	67.1	77.7	72.1	87.1	84.5	69.4	0.06
JS3C-Net	L	80.1	26.2	87.8	84.5	55.2	72.6	71.3	66.3	76.8	71.1	96.8	64.5	76.9	74.1	87.5	86.1	73.6	-
Cylinder3D	L	82.8	29.8	84.3	89.4	63.0	79.3	77.2	73.4	84.6	69.2	97.7	70.2	80.3	75.5	90.4	87.6	77.2	-
AMVNet	L	80.6	31.9	81.7	88.9	67.1	84.3	76.1	73.5	84.9	67.3	97.4	67.4	79.4	75.5	91.5	88.7	77.3	-
SPVNAS	L	80.0	29.9	91.9	90.8	64.7	78.9	75.6	70.9	81.0	74.6	97.4	69.2	79.9	76.1	89.3	87.1	77.3	-
Cylinder3D++	L	82.7	33.8	84.3	89.4	69.6	79.4	77.2	73.4	84.5	69.4	97.6	70.2	80.2	75.5	90.4	87.5	77.8	0.14
AF2S3Net	L	78.8	52.2	89.9	84.1	77.4	74.3	77.3	71.9	83.8	73.7	97.1	66.4	77.5	74.0	87.6	86.8	78.3	-
SPVCNN++	L	**86.3**	43.1	91.9	**92.1**	75.9	75.7	83.4	77.3	86.8	**77.4**	97.7	**71.2**	81.1	77.2	91.7	88.9	81.1	0.11
LIFusion	L+C	58.1	36.3	86.6	84.2	59.9	79.6	80.3	77.7	83.2	68.7	97.1	68.1	77.0	74.4	91.0	88.9	75.7	-
PMF	L+C	82.1	40.3	80.9	86.4	63.7	79.2	79.7	75.8	81.1	67.0	97.2	67.6	78.0	74.4	89.9	88.4	77.0	0.02
CPFusion	L+C	83.6	37.0	89.0	86.2	70.0	77.4	78.0	74.5	82.7	67.9	96.6	68.2	79.5	74.9	90.5	86.9	77.7	-
2D3DNet	L+C	83.0	**59.3**	87.9	85.0	63.7	**84.4**	81.9	75.9	84.7	71.9	96.9	67.3	79.8	75.9	92.0	89.1	79.9	-
CPGNET-LCF	L+C	84.9	63.5	94.4	92.2	79.1	85.9	85.4	78.8	86.2	76.4	**97.9**	66.5	81.0	76.4	**93.0**	**89.5**	83.2	-
Mseg3D [10]	L+C	83.1	42.5	**94.9**	92.0	67.1	78.6	85.7	80.5	**87.5**	77.3	97.7	69.8	81.2	77.8	92.4	90.1	81.4	0.45
UniSeg	L+C	85.9	71.2	92.1	91.6	**80.5**	88.0	80.9	76.0	86.3	76.7	97.7	71.8	80.7	76.7	91.3	88.8	83.5	-
Ours	L+C	85.1	53.4	93.2	91.8	78.5	78.9	**86.3**	77.5	86.8	76.2	97.8	66.6	**81.4**	**77.3**	91.2	87.7	82.5	0.12

Table 2. Quantitative comparisons on the semanticKITTI dataset using mIoU.

Method	Modality	mIoU
SalsaNext	L	59.4
SPVNAS	L	62.3
Cylinder3D	L	64.9
PointPainting	L+C	54.5
PMF	L+C	63.9
UniSeg	L+C	75.2
CPGNet-LCF	L+C	67.1
MSeg3D [10]	L+C	66.7
Ours	L+C	67.3

Table 3. Robustness analysis on Nuscenes by removing some cameras as malfunction.

#-Camera	6	5	4	3	2	1	0
mIoU	82.5	79.8	78.6	77.9	76.7	75.8	74.8

Table 4. The impacts of adopting different attention mechanisms on mIoU and latency on the NuScenes dataset.

Attention	IntraMFE	InterMFL	mIoU	#Params(M)	Latency(s)
Softmax	✓	✓	81.3	87.34	0.415
	✓	S2A	81.7	72.88	0.359
	S2A	✓	82.0	66.27	0.223
Linear	✓	✓	75.3	48.84	0.108
	✓	S2A	77.4	49.42	0.110
	S2A	✓	79.6	50.11	0.113
S2Agent	✓	✓	82.5	51.85	0.117

Table 5. Robustness analysis on point cloud frame per second using the Nuscenes dataset. "#L-Frame" is the frame number per second for LiDAR.

#-Camera	×	×	×	×	×	×	✓
#L-Frame	1	10	20	25	30	40	25
mIoU(MSeg3D)	72.0	74.7	75.4	75.8	75.3	75.2	81.2
mIoU(Ours)	73.5	75.1	76.4	76.8	76.3	75.8	82.5

Table 6. Effect of different fusion strategies on mIoU and latency on NuScenes.

Fusion strategy	mIoU	#Params(M)	Latency(s)
Early-Fusion	80.4	47.64	0.099
Late-Fusion	82.2	52.71	0.131
Mid-Fusion	82.5	51.85	0.117

Ablation Analyses

1. Attention Modules: Table 4 provides a summary of the softmax, linear, and the S2Agent attention modules on mIoU, number of weight parameters, and latency on our proposed method. Specifically, S2Agent attention leads to best mIoU with a lower latency than when using softmax attention. **2. Robustness Analysis on Camera Failure**: Simulation experiments under different camera failure conditions are carried out and results are shown in Table 3. With all cameras functioning normally, we achieve 82.5 on mIoU. In contrary, only 74.8 on mIoU is obtained when all six cameras fail. **3. Robustness Analysis on Point Cloud Frame Per Second:** Using the nuScenes dataset and following the methodology of [22], we aggregate multiple previous frames into the current frame based on the provided ego-vehicle motion information. As is shown in Table 5, we obtain best performance on mIoU when the number of LiDAR frames per second is 25. **4. Fusion Strategy**: Table 6 summarizes the impact of various fusion strategies on mIoU, the number of weight parameters, and latency within

our proposed method. Notably, the Mid-Fusion strategy achieves the highest mIoU performance while maintaining significantly lower latency when compared to other fusion strategies.

6 Conclusion

In this work, we introduce a new agent attention module, the Semantic Segmentation Agent (S2A) attention module, which is designed to reduce computational complexity in 3D semantic segmentation tasks using LiDAR and cameras. By integrating intra-modality feature encoding with S2A attention modules and employing a mid-fusion strategy, we have improved segmentation speed in an end-to-end 3D semantic segmentation network tailored for autonomous driving. Our evaluation on two public benchmark datasets, nuScenes and Semantic KITTI, demonstrates that our network achieves comparable performance in terms of mIoU against selected baselines. Notably, when executed in the same hardware environment, our approach significantly reduces latency when compared to the newly proposed MSeg3D.

References

1. Behley, J., et al.: Semantickitti: a dataset for semantic scene understanding of lidar sequences. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
2. Caesar, H., et al.: nuscenes: a multimodal dataset for autonomous driving. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
3. Charles, R.Q., Su, H., Kaichun, M., Guibas, L.J.: Pointnet: deep learning on point sets for 3D classification and segmentation. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
4. Chen, T.H., Chang, T.S.: Rangeseg: range-aware real time segmentation of 3D lidar point clouds. IEEE Trans. Intell. Veh. **7**(1), 93–101 (2021)
5. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3D U-Net: learning dense volumetric segmentation from sparse annotation. In: Ourselin, S., Joskowicz, L., Sabuncu, M.R., Unal, G., Wells, W. (eds.) MICCAI 2016. LNCS, vol. 9901, pp. 424–432. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46723-8_49
6. Cortinhal, T., Tzelepis, G., Aksoy, E.: Salsanext: fast semantic segmentation of lidar point clouds for autonomous driving (2020)
7. Han, D., Ye, T., Han, Y., Xia, Z., Song, S., Huang, G.: Agent attention: on the integration of softmax and linear attention. arXiv preprint arXiv:2312.08874 (2023)
8. Hu, Q., et al.: Randla-net: efficient semantic segmentation of large-scale point clouds. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
9. Krispel, G., Opitz, M., Waltner, G., Possegger, H., Bischof, H.: Fuseseg: Lidar point cloud segmentation fusing multi-modal data. Cornell University - arXiv (2019)
10. Li, J., Dai, H., Han, H., Ding, Y.: MSeg3D: multi-modal 3D semantic segmentation for autonomous driving (2023)

11. Liu, Y., et al.: Uniseg: a unified multi-modal lidar segmentation network and the openpcseg codebase. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 21662–21673 (2023)
12. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
13. Milioto, A., Vizzo, I., Behley, J., Stachniss, C.: Rangenet++: fast and accurate lidar semantic segmentation. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4213–4220. IEEE (2019)
14. Nagrani, A., Yang, S., Arnab, A., Jansen, A., Schmid, C., Sun, C.: Attention bottlenecks for multimodal fusion. Adv. Neural. Inf. Process. Syst. **34**, 14200–14213 (2021)
15. Park, J., Kim, C., Kim, S., Jo, K.: Pcscnet: fast 3D semantic segmentation of lidar point cloud for autonomous car using point convolution and sparse convolution network. Expert Syst. Appl. **212**, 118815 (2023)
16. Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5693–5703 (2019)
17. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
18. Wang, W., et al.: Pyramid vision transformer: a versatile backbone for dense prediction without convolutions. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV) (2021)
19. Wang, Y., Shi, T., Yun, P., Tai, L., Liu, M.: Pointseg: real-time semantic segmentation based on 3D lidar point cloud. arXiv preprint arXiv:1807.06288 (2018)
20. Xia, Z., Pan, X., Song, S., Li, L.E., Huang, G.: Vision transformer with deformable attention. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4794–4803 (2022)
21. Xu, C., et al.: SqueezeSegV3: Spatially-Adaptive Convolution for Efficient Point-Cloud Segmentation, pp. 1–19 (2020)
22. Yin, T., Zhou, X., Krahenbuhl, P.: Center-based 3D object detection and tracking. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021)
23. Yuan, Y., Chen, X., Wang, J.: Object-Contextual Representations for Semantic Segmentation, pp. 173–190 (2020)
24. Zermas, D., Izzat, I., Papanikolopoulos, N.: Fast segmentation of 3D point clouds: a paradigm on lidar data for autonomous vehicle applications. In: 2017 IEEE International Conference on Robotics and Automation (ICRA), pp. 5067–5073. IEEE (2017)
25. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
26. Zhou, H., et al.: Cylinder3d: an effective 3D framework for driving-scene lidar semantic segmentation. arXiv, Computer Vision and Pattern Recognition (2020)

Author Queries

Chapter 21

Query Refs.	Details Required	Author's response
AQ1	This is to inform you that corresponding authors have been identified as per the information available in the Copyright form.	
AQ2	Please check and confirm if the authors and their respective email address have been correctly identified. Amend if necessary.	
AQ3	Per Springer style, both city and country names must be present in the affiliations. Accordingly, we have inserted the city name in affiliation. Please check and confirm if the inserted city name is correct. If not, please provide us with the correct city name.	