2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2025)
October 19-25, 2025. Hangzhou, China

KDMOS:Knowledge Distillation for Motion Segmentation

Chunyu Cao, Jintao Cheng, Zeyu Chen, Linfan Zhan, Rui Fan, Zhijian He, Xiaoyu Tang*

***Abstract*— Motion Object Segmentation (MOS) is crucial for autonomous driving, as it enhances localization, path planning, map construction, scene flow estimation, and future state prediction. While existing methods achieve strong performance, balancing accuracy and real-time inference remains a challenge. To address this, we propose a logits-based knowledge distillation framework for MOS, aiming to improve accuracy while maintaining real-time efficiency. Specifically, we adopt a Bird's Eye View (BEV) projection-based model as the student and a non-projection model as the teacher. To handle the severe imbalance between moving and non-moving classes, we decouple them and apply tailored distillation strategies, allowing the teacher model to better learn key motion-related features. This approach significantly reduces false positives and false negatives. Additionally, we introduce dynamic upsampling, optimize the network architecture, and achieve a 7.69% reduction in parameter count, mitigating overfitting. Our method achieves a notable IoU of 78.8% on the hidden test set of the SemanticKITTI-MOS dataset and delivers competitive results on the Apollo dataset.The KDMOS implementation is available at https://github.com/SCNU-RISLAB/KDMOS.**

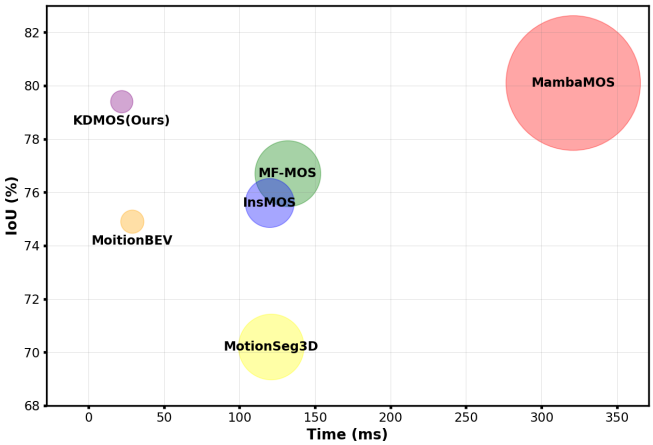

Fig. 1. The performance, inference speed, and parameter size of different MOS models on the SemanticKITTI-MOS test set are compared. The size of the circles represents the parameter size. Our knowledge distillation-based KDMOS not only outperforms MotionBEV, MF-MOS, and InsMOS in performance but also maintains a very low computational cost.

I. INTRODUCTION

Accurate separation of moving and static objects is crucial for efficient path planning and safe navigation in dynamic traffic environments [1]. Moving Object Segmentation (MOS), particularly for pedestrians, cyclists, and vehicles, reduces system errors caused by dynamic objects and improves environmental perception accuracy [2].This technology is essential for reducing uncertainties in scene flow estimation [3], [4] and path planning [5], enabling autonomous systems to make precise and reliable decisions. MOS plays a key role in real-time obstacle detection and adaptive environmental perception, making it integral to autonomous driving technology.

For the MOS task, existing solutions can be categorized into projection-based [2], [6]–[8] and non-projection-based methods [9]–[11]. Projection-based methods lose geometric information when mapping results back to the 3D point cloud space, limiting their performance. To address this, [7] proposed a two-stage approach using 3D sparse convolution to fuse information from the projection map and point cloud, reducing back-projection loss and improving accuracy.However, this approach is computationally expensive, making it challenging to balance accuracy and inference speed. Likewise, non-projection-based methods [9]–[11] face the same issue. Among them, MambaMOS [11] achieves state-of-the-art (SoTA) performance in the MOS task, but its long processing time limits real-time applicability (see Fig. 1).

Knowledge distillation (KD) [12] is an effective model compression technique that addresses real-time performance issues. Previous distillation algorithms [13], [14] have been successfully applied to LiDAR semantic segmentation, achieving remarkable results. However, most of these methods focus on model compression [13] or improving feature extraction [15], with few dedicated to generalizable knowledge distillation methods for MOS tasks.

Based on the most intuitive observation that balancing accuracy and real-time performance is fundamental to solving the MOS problem, we propose KDMOS. The core idea of KDMOS is to compress the large model and balance inference speed and accuracy. We propose Weighted Decoupled Class Distillation (WDCD), a logit-based distillation method. Following the DKD approach [16], we first decouple traditional knowledge distillation into KL losses for target and non-target classes.To mitigate the severe class imbalance in the moving category for MOS tasks and enhance distillation performance, we further decouple moving and non-moving classes, apply different distillation strategies to compute

(*Corresponding author: Xiaoyu Tang*).
Chunyu Cao, Jintao Cheng, Linfan Zhan, Zeyu Chen, and Xiaoyu Tang are with the School of Electronic and Information Engineering, South China Normal University, Foshan 528225, China. tangxy@scnu.edu.cn
Rui Fan is with the College of Electronics & Information Engineering, Shanghai Research Institute for Intelligent Autonomous Systems, the State Key Laboratory of Intelligent Autonomous Systems, and Frontiers Science Center for Intelligent Autonomous Systems, Tongji University, Shanghai 201804, China. rui.fan@ieee.org
Zhijian He is with the College of Big Data and Internet, Shenzhen Technology University, Shenzhen, China. hezhijian@sztu.edu.cn

their respective losses, and assign appropriate weights via labels.This allows the student model to effectively learn critical information among potential moving object categories, significantly reducing false positives and missed detections (see Fig. 6). Moreover, it can be broadly applied to other MOS tasks (see Fig. 5). Additionally, we introduce dynamic upsampling in the network, achieving an inference speed of 40 Hz while maintaining a balance between accuracy and speed.

Extensive experiments demonstrate the superiority of our design. In summary, the main contributions of this paper are as follows:

- We propose a general distillation framework for the MOS task, effectively balancing real-time performance and accuracy (see Fig. 1). To the best of our knowledge, this is the first application of knowledge distillation to MOS.
- The KDMOS network architecture is improved by introducing the Dysample offset, reducing model complexity and mitigating overfitting.
- Our method achieves competitive results on the SemanticKITTI [6] and Apollo [17] datasets, demonstrating its superior performance and robustness.

II. RELATED WORK

A. Knowledge Distillation

Knowledge distillation (KD) was first introduced by Hinton et al. [12]. Its core idea is to bridge the performance gap between a complex teacher model and a lightweight student model by transferring the rich knowledge embedded in the teacher. Existing methods can be categorized into two types: distillation from logits [12], [16], [18], [19] and from intermediate features [13], [14]. Hou et al. [13] proposed Point-to-Voxel Knowledge Distillation (PVKD), the first application of knowledge distillation to LiDAR semantic segmentation.They introduced a supervoxel partitioning method and designed a difficulty-aware sampling strategy. Feng et al. [14] proposed a voxel-to-BEV projection knowledge distillation method, effectively mitigating information loss during projection.Borui et al. [16] decouple the classical KD formula, splitting the traditional KD loss into two components that can be more effectively and flexibly utilized through appropriate combinations. Although feature-based methods often achieve superior performance, they incur higher computational and storage costs and require more complex structures to align feature scales and network representations. Compared to the simple and effective logit-based approach, it has weaker generalization ability and is less applicable to various downstream tasks.

B. MOS Based on Deep Learning

Recent methods tend to apply popular deep learning models and directly capture spatiotemporal features from data. Chen et al [6] proposed the LMNet, which uses range residual images as input and extracts temporal and spatiotemporal information through existing segmentation networks. Sun et al. [7] introduced a dual-branch structure that separately processes range images and residual images, utilizing a motion-guided attention module for feature fusion. Cheng et al. [2] suggested that residual images offer greater potential for motion information and proposed a motion-focused model with a dual-branch structure to achieve the decoupling of spatiotemporal information. Unlike RV projection, BEV projection represents point cloud features from a top-down view, preserving object scale consistency and improving interpretability and processing efficiency. MotionBEV [8] projects to the polar BEV coordinate system and extracts motion features through height differences over temporal windows. CV-MOS [20] proposes a cross-view model that integrates RV and BEV perspectives to capture richer motion information. Non-projection-based methods directly operate on point clouds in 3D space. 4DMOS [9] uses sparse four-dimensional convolution to jointly extract spatiotemporal features from input point cloud sequences and integrates predictions through a binary Bayesian filter, achieving good segmentation results. Mambamos [11] considers temporal information as the dominant factor in determining motion, achieving a deeper level of coupling beyond simply connecting temporal and spatial information, thereby enhancing MOS performance. Therefore, previous methods have struggled to balance real-time performance and accuracy. To address this issue, this paper proposes KDMOS based on knowledge distillation, which distills a non-projection-based large model into a projection-based lightweight model through logits distillation.

III. METHODOLOGY

In this section, we present a detailed description of KDMOS, with its overall framework illustrated in Fig. 2. We start with data preprocessing, followed by an explanation of the KDMOS network architecture and WDKD. Finally, we provide an in-depth analysis of the loss function components.

A. Input Representation

Student Input Representation. We employ an extreme bird's-eye view (BEV) for point cloud coordinate segmentation, a lightweight data representation obtained by projecting the 3D point cloud into 2D space. Following the setup from previous work [8], we project the LiDAR point cloud onto a BEV image. After obtaining the BEV images of the past N-1 consecutive frames, we align the past frames with the current frame's viewpoint using pose transformation $T \in \mathbb{R}^{4\times4}$, resulting in the projected BEV images. Meanwhile, we maintain two adjacent time windows, Q_1 and Q_2, with equal lengths, and obtain the residual image by computing the height difference between the corresponding grids in the two time windows.

$$\begin{aligned} Z_{(u,v),i} &= \{z_j \in p_j \mid p_j \in Q_{(u,v),i}, z_{\min} < z_j < z_{\max}\}, \\ I_{(u,v),i} &= \max\{Z_{(u,v),i}\} - \min\{Z_{(u,v),i}\}. \end{aligned} \tag{1}$$

where p_j is a point within the temporal window $Q_{(u,v),i}$, represented as $[x_j, y_j, z_j, 1]^T$, and each pixel value $I_{(u,v),i}$ represents the height occupied by the $(u,v)_{th}$ grid in Q_i. Following MotionBEV [8], we restrict the z-axis range

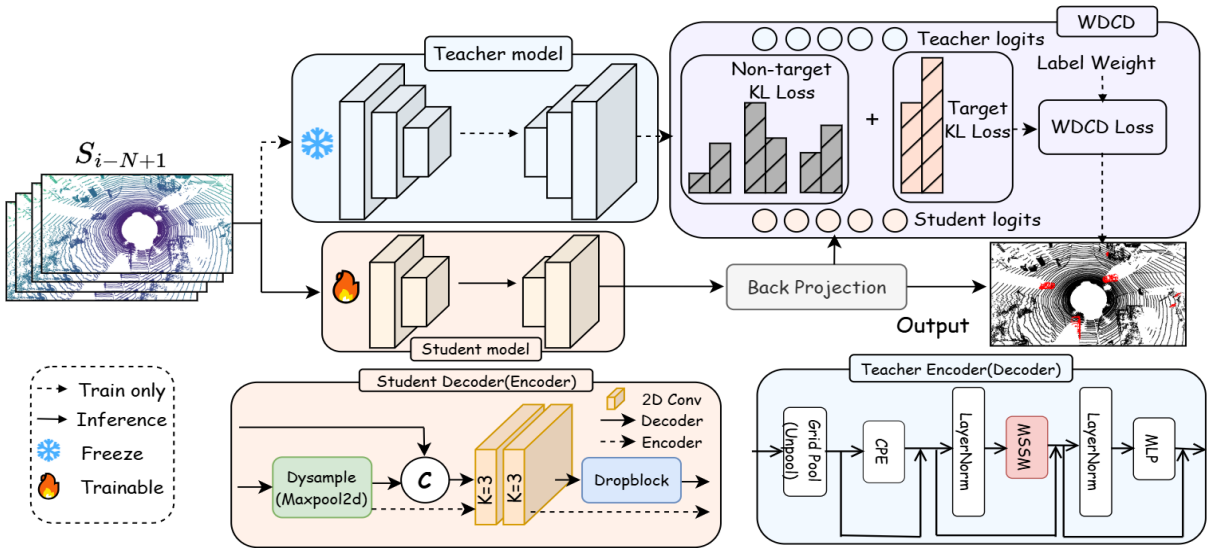

Fig. 2. The KDMOS framework comprises three main components: the teacher model, the student model, and knowledge distillation. During training, the teacher model uses pre-trained weights and remains frozen, while the student model is trained from scratch, with its parameters continuously updated through WDCD and the model's own loss. MSSM, proposed by MambaMOS [11], achieves deep coupling of temporal and spatial features.

to $(z_{\min}, z_{\max}) = (-4, 2)$. Subsequently, we compute the residuals between the projected BEV images I_1 and I_2:

$$\begin{aligned} D^0_{(x,y),i}, D^1_{(x,y),i-1}, ..., D^{N_2-1}_{(x,y),i-N_2+1} &= I_{(x,y),1} - I_{(x,y),2}, \\ D^{N_2}_{(x,y),i-N_2}, D^{N_2+1}_{(x,y),i-N_2-1}, ..., D^{N-1}_{(x,y),i-N+1} &= I_{(x,y),2} - I_{(x,y),1} \end{aligned} \tag{2}$$

where $D^k_{(u,v),i}$ represents the motion feature in the $(u,v)^{\text{th}}$ grid of the k^{th} channel for the i^{th} frame.

Teacher Input Representation. To align with the student's input, following the setup of previous work [11], we need to align the past frames with the current frame's viewpoint using the pose transformation matrix $T \in \mathbb{R}^{4\times4}$ and convert the homogeneous coordinates into Cartesian coordinates, resulting in a sequence of N continuous 4D point cloud sets. To distinguish each scan within the 4D point cloud, we add the corresponding time step of each scan as an additional dimension of the point, obtaining a spatio-temporal point representation:

$$\begin{aligned} p'_i &= \left[x_i, y_i, z_i, t_i\right]^T, \\ S' &= \{S_0, S_{1\to0}, \ldots, S_{t\to0}\} \end{aligned} \tag{3}$$

Next, We follow the setup of MambaMOS [11] to obtain sequences from unordered 4D point cloud sets.:

$$\begin{aligned} S'_i &= \Psi(S'), \\ S' &= \Psi^{-1}(S'_i) \end{aligned} \tag{4}$$

B. Network Structure

1)We propose a novel knowledge distillation network architecture, as shown in Fig. 2. The KDMOS network comprises three main components. First, the teacher model, where we select MambaMOS [11] as the teacher, which deeply integrates temporal and spatial information, delivering strong performance but with a high computational cost. We select a BEV-based method [8] as the student model.

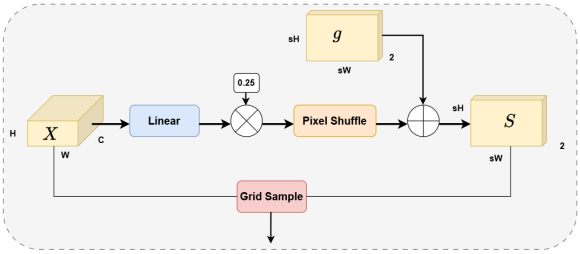

Fig. 3. Structure of the Dysample module. The input feature and original grid are denoted by X and g , respectively.

BEV projection provides a global top-down view, intuitively representing object distribution and relative positions in the scene while maintaining low computational complexity. The final component is the knowledge distillation model, the core of our framework. With WDCD (see Fig. 4), the student model learns category similarities and differences during training, enhancing performance without additional computational cost.

2)Feature Upsampling Module:To balance accuracy and inference speed, we optimize the upsampling module with the dynamic mechanism DySample [21]. As shown in Fig. 3, the feature map is transformed through a linear layer, scaled by an offset factor to compute pixel displacement coordinates, and refined via pixel shuffle for upsampling. Displacement coordinates are added to the base grid for precise sampling. This approach replaces fixed convolution kernels with point-based sampling, reducing parameter count and model complexity, thereby lowering the risk of overfitting.

C. Weighted Decoupled Class Distillation

WDCD is a key distillation module in our framework, as shown in Fig. 4. Most distillation methods [13], [14] rely on intermediate layer features from both the teacher and student

networks.However, if the teacher and student architectures differ significantly, aligning feature scales and network capabilities requires more complex structures, making distillation more challenging and less effective. In contrast, logits-based distillation [12], [16] relies solely on the teacher's final output, bypassing its internal structure. This approach is simpler to implement, more computationally efficient, and applicable to other MOS methods.

The logits contain relational information between categories, and to further leverage this information, unlike previous MOS methods [2], [6]–[11], we divide the final predictions of points into four categories: unlabeled, static, movable, and moving.Specifically, for a training sample from the t-th class, we first compute the probabilities for the target and non-target classes for each point:

$$p_t = \frac{\exp(z_t)}{\sum_{j=1}^{4}\exp(z_j)}, \quad p_{\setminus t} = \frac{\sum_{k=1,k\neq t}^{4}\exp(z_k)}{\sum_{j=1}^{4}\exp(z_j)} \tag{5}$$

where z_j represents the logit of the j-th class, and p_t and $p_{\setminus t}$ denote the probabilities of target and non-target classes, respectively.We know that the distribution probabilities for each class can be represented as $p = [p_1, p_2, p_3, p_4]$. Meanwhile, we define $\hat{p} = [\hat{p}_1, \hat{p}_2, \hat{p}_3]$ to independently represent the probabilities for the non-target classes, excluding the target class (i.e., without incorporating its influence).Each element is computed as follows:

$$p_i = \frac{\exp(z_i)}{\sum_{j=1}^{4}\exp(z_j)}, \quad \hat{p}_i = \frac{\exp(z_i)}{\sum_{j=1,j\neq m}^{4}\exp(z_j)}. \tag{6}$$

Specifically, we use the binary probability b and the non-target probability $\hat{p}$ to represent knowledge distillation (KD), where T and S denote the teacher and student models, respectively.

$$\mathrm{KD} = \mathrm{KL}(p^T \| p^S) = p_t^T \log\left(\frac{p_t^T}{p_t^S}\right) + \sum_{i=1,i\neq t}^{C} p_i^T \log\left(\frac{p_i^T}{p_i^S}\right). \tag{7}$$

From Equations 5 and 6, we obtain $\hat{p}_i = p_i/p_{\setminus t}$. Thus, Equation 7 can be rewritten as:

$$\begin{aligned}\mathrm{KD} &= p_t^T \log\left(\frac{p_t^T}{p_t^S}\right) + p_t^T \log\left(\frac{p_{\setminus t}^T}{p_{\setminus t}^S}\right) \\ &\quad + p_{\setminus t}^T \sum_{i=1,i\neq t}^{C} \hat{p}_i^T \log\left(\frac{\hat{p}_i^T}{\hat{p}_i^S}\right) \\ &= \mathrm{KL}(\boldsymbol{b}^T \| \boldsymbol{b}^S) + (1 - p_t^T)\mathrm{KL}(\hat{\boldsymbol{p}}^T \| \hat{\boldsymbol{p}}^S)\end{aligned} \tag{8}$$

where $\mathrm{KL}(\boldsymbol{b}^T \| \boldsymbol{b}^S)$ represents the similarity between the binary probabilities of the target class for the teacher and student models, while $\mathrm{KL}(\hat{\boldsymbol{p}}^T \| \hat{\boldsymbol{p}}^S)$ denotes the similarity between the probabilities of non-target classes for the teacher and student models.In the MOS task, the severe imbalance between moving and non-moving classes results in the number of non-moving points being approximately 400 times

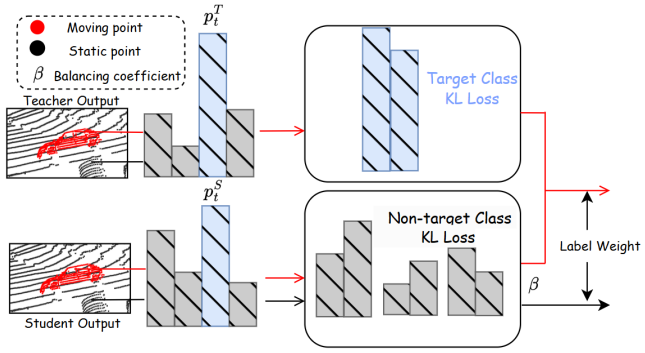

Fig. 4. The structure of WDCD, where p_t^T and p_t^S represent the teacher's and student's probabilities for the target class, respectively.

that of moving points. During training, the high accuracy of non-moving classes significantly reduces the effectiveness of $\mathrm{KL}(\boldsymbol{b}^T|\boldsymbol{b}^S)$, which may even become detrimental (as shown in Table V). Meanwhile, $\mathrm{KL}(\hat{\boldsymbol{p}}^T\|\hat{\boldsymbol{p}}^S)$ is also influenced by p_t^T, severely impairing the effectiveness of knowledge distillation.

Therefore, in the MOS task, we further decouple moving and non-moving classes. For moving classes, both losses are computed as usual, while for easily trainable non-moving classes, only $\mathrm{KL}(\hat{\boldsymbol{p}}^T|\hat{\boldsymbol{p}}^S)$ is applied. We define this approach as Decoupled Class Distillation (DCD).

$$\mathrm{DCD} = \begin{cases} \mathrm{KL}(\boldsymbol{b}^T\|\boldsymbol{b}^S) + \beta\mathrm{KL}(\hat{\boldsymbol{p}}^T\|\hat{\boldsymbol{p}}^S), & \text{class} = \text{moving} \\ \beta\mathrm{KL}(\hat{\boldsymbol{p}}^T\|\hat{\boldsymbol{p}}^S), & \text{class} \neq \text{moving} \end{cases} \tag{9}$$

where β is the balancing coefficient. Based on label-assigned weighting, our final WDCD is formulated as:

$$\begin{aligned} W^i &= \mathrm{Content}[\mathrm{label}], \\ \mathrm{WDCD} &= \mathrm{DCD}/W^i \end{aligned} \tag{10}$$

Where Content represents the ratio of points from different categories in the i^{th} frame, and label represents the ground truth labels of points in the i^{th} frame.

D. Loss Function

During the training process, the total loss function of this algorithm includes both the segmentation loss and the knowledge distillation loss:

$$\mathcal{L}_{\mathrm{Total}} = \mathcal{L}_{\mathrm{Student}} + \gamma\mathcal{L}_{\mathrm{LDCD}}, \tag{11}$$

where L_{Student} represents the segmentation loss generated by the student network, L_{LDCD} is the losses of Knowledge Distillation and γ is the balancing coefficient.The student network loss consists of the cross-entropy loss L_{wce} and the Lovász-Softmax loss L_{ls}. The loss function is defined as follows:

$$\mathcal{L}_{\mathrm{Student}} = \mathcal{L}_{\mathrm{wce}} + \mathcal{L}_{\mathrm{ls}}, \tag{12}$$

TABLE I

COMPARISONS RESULT ON SEMANTICKITTI-MOS DATASET.

Methods	Publication	Validation(%)	Test(%)
LMNet [6]	ICRA 2021	63.8	60.5
LiMoSeg [22]	ArXiv 2021	52.6	-
Cylinder3D [23]	CVPR 2021	66.3	61.2
RVMOS [24]	RAL 2022	71.2	74.7
4DMOS [9]	RAL 2022	77.2	65.2
MotionSeg3D [7]	IROS 2022	71.4	70.2
InsMOS [10]	IROS 2023	73.2	75.6
MotionBEV [8]	RAL 2023	76.5	75.8
SSF-MOS [4]	TIM 2024	70.1	-
Mambamos [11]	ACM MM 2024	**82.3**	**80.1**
MF-MOS [2]	ICRA 2024	76.1	76.7
Ours	-	79.4	78.8

TABLE II

CROSS VAL IOU PERFORMANCE OF DIFFERENT METHODS ON APOLLO DATASET.

Methods	Validation(%)
LMNet	16.9
MotionSeg3D	7.5
MotionBEV	25.1
MF-MOS	49.9
4DMOS	**73.1**
Ours	68.2

TABLE III

ABLATION EXPERIMENTS WITH PROPOSED MODULES.

Methods	Component		IoU(%)	params(M)
	Dysample	WDCD		
MotionBEV			76.5	4.42
KDMOS(i)	-	-	76.2	4.42
KDMOS(ii)	✓	-	76.7	4.08
	-	✓	77.8	4.42
	✓	✓	**79.4**	**4.08**

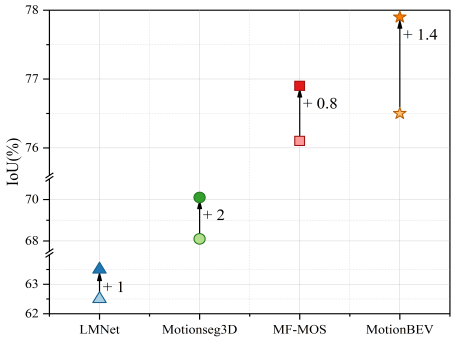

Fig. 5. The performance of the proposed WDCD module on other MOS methods (IoU%).

IV. EXPERIMENTS

In this section, we conduct extensive experiments to comprehensively evaluate KDMOS. Section IV-A introduces the datasets and evaluation metrics. In Section IV-B, we present quantitative and qualitative comparisons between our method and other state-of-the-art approaches on SemanticKITTI-MOS [6] and Apollo [17]. In Section IV-C, we conduct ablation experiments to evaluate the effectiveness of our knowledge distillation module. In Section IV-D, we perform a qualitative analysis to intuitively compare our algorithm with other SoTA approaches. Finally, in Section IV-E, we present the runtime performance of our method.

A. Experiment Setups

The SemanticKITTI-MOS dataset [25] is the largest and most authoritative dataset for the MOS task, derived from the original SemanticKITTI dataset. We follow the setup from previous work [8] for training, validation, and testing. Additionally, to evaluate the generalization ability of our method across different environments, we tested it on another dataset, Apollo, following the setup by Chen et al. [6].

Our code is implemented in PyTorch. Experiments were conducted on a single NVIDIA RTX 4090 GPU with a batch size of 8. We trained KD-MOS for 150 epochs using Stochastic Gradient Descent (SGD) to minimize L_{wce}, L_{ls}, and L_{WDCD}, with a momentum of 0.9 and a weight decay of 0.0001. The initial learning rate was set to 0.005 and decayed by a factor of 0.99 after each epoch. The offset factor for DySample a was set to 0.25, and the weight of L_{WDCD} was set to 0.3. The teacher network used MambaMOS pre-trained weights with frozen parameters. The student network was trained from scratch without pre-trained weights, and training continued until the validation loss converged. We used the Jaccard Index, also known as the Intersection-over-Union (IoU) metric [26], to evaluate MOS performance for moving objects.

B. Comparison with State-of-the-Art Methods

First, we report the validation and test results on the SemanticKITTI-MOS dataset [6] in Tab. I. Competitive results were achieved on the validation set. For the test set, we submitted the moving object segmentation results to the benchmark server, which also demonstrated excellent performance, further confirming the model's robustness.As shown in Tab. I, our mIoU score is lower than that of MambaMOS [11] in both the validation and test benchmarks. However, as shown in Tab. VI, while non-projection-based methods achieve higher accuracy, they often struggle to balance accuracy and inference speed. In contrast, our method effectively combines the advantages of both projection-based and non-projection-based approaches, demonstrating superior overall performance.Specifically, compared to the baseline MotionBEV, the performance improved by 3.9% on the validation set and 2.9% on the test set.

To evaluate the generalization ability of our method in different environments, we also report the validation results on the Apollo dataset [17] in the Tab. II. Following the standard settings of previous methods [10], [27],the data in the Tab. II does not use any domain adaptation techniques or retraining. Compared to other projection-based methods, our KDMOS performed the best. Although the results on the Apollo dataset are not as good as those of 4DMOS [9], as shown in the table, Our method demonstrates significantly faster inference speed and outperforms 4DMOS on large-scale datasets such as SemanticKITTI-MOS.

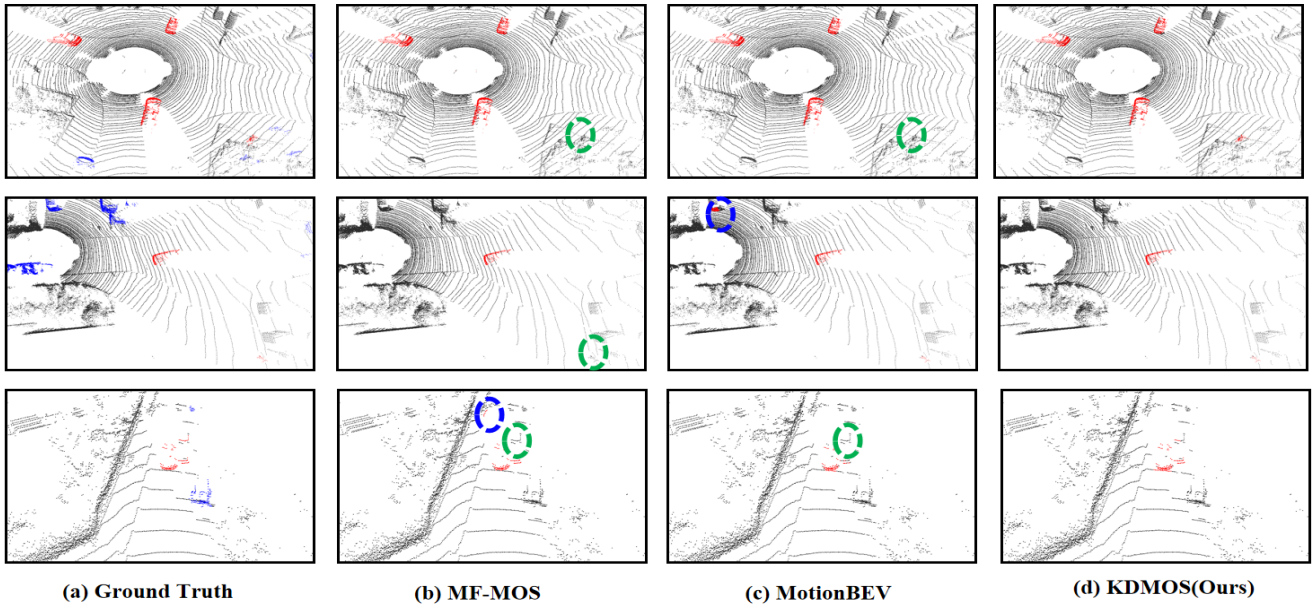

Fig. 6. Qualitative results of LiDAR-MOS on the SemanticKITTI-MOS validation set using different methods. Green circles indicate false negatives, while blue circles indicate false positives.

TABLE IV
COMPARISONS OF DIFFERENT KD METHODS.

Method	Publication	IoU(%)
KD [12]	NIPS 2015	77.2
DKD [16]	CVPR 2022	77.6
KD+Std [28]	CVPR 2024	77.4
Ours	-	**79.4**

TABLE V
ABLATION OF DKD MODULES ON NON-MOVING CLASSES USING THE SEMANTICKITTI-MOS VALIDATION SET

Component		IoU(%)
NCKD	TCKD	
-	-	76.5
-	✓	76.2
✓	-	77.3
✓	✓	77.1

TABLE VI
COMPUTATION RESOURCE COMPARISON.

Method	FPS	ms	params(M)	size(MB)
4DMOS	15	65	**1.84**	**7**
MambaMos	3	321	184.78	564
MF-MOS	8	121	35.59	132
MotionBEV	34	29	4.42	17
KDMOS	**40**	**25**	4.08	15

C. Ablation Experiment

In this section, we conduct ablation experiments on the proposed KDMOS and its various components. All experiments are performed on the SemanticKITTI validation set (sequence 08). As shown in Tab. III. It is noteworthy that our proposed WDCD shows significant improvement (+1.3% IoU) compared to MotionBEV without increasing the number of parameters. To further demonstrate the indispensability of each component, we conducted ablation experiments with different component combinations in Setting ii. Each proposed component consistently improves baseline performance to varying degrees.The last row shows that our complete KDMOS achieves the best performance, with a significant accuracy improvement (+2.9% IoU) and a 7.69% reduction in parameter count compared to the baseline.

To evaluate the generalization ability of the proposed WDCD, we applied it to other MOS methods [2], [6]–[8], using them as baseline models and training from scratch. The experimental results are shown in Fig. 5. The proposed module also enhances the performance of other MOS models. Notably, due to the nature of logits-based knowledge distillation, it introduces no additional parameters while improving the model's performance without any loss. Furthermore, to explore the advantages brought by our method,we compared WDCD with other KD algorithms [12], [16], [28], [29], As shown in Tab. IV. Our WDCD demonstrates superior performance over other methods in MOS task.

To further validate our argument in Section III-C, we conduct an ablation study on each distillation module for non-moving classes using the SemanticKITTI-MOS validation set, while applying WDCD to the moving class as usual. The results are shown in Table V, where TCKD and NCKD represent the distillation of the teacher and student models on the target and non-target classes, respectively.Since the number of non-moving classes is significantly higher than that of moving classes, they achieve higher accuracy during training. As a result, TCKD has a limited effect and may even be detrimental (-0.3% IoU). In contrast, applying NCKD alone proves more effective than combining both.

D. Qualitative Analysis

To provide a more intuitive comparison between our algorithm and other SoTA algorithms, we conducted a visual qualitative analysis on the SemanticKITTI-MOS dataset. As shown in Fig. 6, both MF-MOS and MotionBEV exhibit misclassification of movable objects and missed detection of moving objects. Compared to SoTA algorithms, the knowledge distillation-based model effectively mitigates the impact of moving targets and accurately captures them.

E. Evaluation of Resource Consumption

We evaluate the inference time (FPS, ms), memory usage (size), and the number of learnable parameters (params) of our method and SoTA methods on Sequence 08, using 112 Intel(R) Xeon(R) Gold 6330 CPUs @ 2.00GHz and a single NVIDIA RTX 4090 GPU. As shown in Tab. VI. We achieved real-time processing speed and outperformed MotionBEV in terms of performance, achieving a balance between accuracy and inference speed.

V. CONCLUSIONS

This paper presents an efficient knowledge distillation framework tailored for MOS, enabling effective transfer of knowledge from a large-parameter teacher model to a smaller student model. The introduction of Dysample reduces model complexity and overfitting risks. Experimental results show that: 1) The proposed KDMOS model achieves high accuracy and fast inference on the SemanticKITTI-MOS dataset, balancing accuracy and speed; 2) The framework exhibits strong performance and generalization, improving various MOS methods without loss.

REFERENCES

[1] X. Chen, S. Li, B. Mersch, L. Wiesmann, J. Gall, J. Behley, and C. Stachniss, "Moving object segmentation in 3d lidar data: A learning-based approach exploiting sequential data," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6529–6536, 2021.

[2] J. Cheng, K. Zeng, Z. Huang, X. Tang, J. Wu, C. Zhang, X. Chen, and R. Fan, "Mf-mos: A motion-focused model for moving object segmentation," *arXiv preprint arXiv:2401.17023*, 2024.

[3] S. A. Baur, D. J. Emmerichs, F. Moosmann, P. Pinggera, B. Ommer, and A. Geiger, "Slim: Self-supervised lidar scene flow and motion segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 13 126–13 136.

[4] T. Song, Y. Liu, Z. Yao, and X. Wu, "Ssf-mos: Semantic scene flow assisted moving object segmentation for autonomous vehicles," *IEEE Transactions on Instrumentation and Measurement*, 2024.

[5] H. Thomas, B. Agro, M. Gridseth, J. Zhang, and T. D. Barfoot, "Self-supervised learning of lidar segmentation for autonomous indoor navigation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 14 047–14 053.

[6] X. Chen, S. Li, B. Mersch, L. Wiesmann, J. Gall, J. Behley, and C. Stachniss, "Moving object segmentation in 3d lidar data: A learning-based approach exploiting sequential data," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6529–6536, 2021.

[7] J. Sun, Y. Dai, X. Zhang, J. Xu, R. Ai, W. Gu, and X. Chen, "Efficient spatial-temporal information fusion for lidar-based 3d moving object segmentation," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 11 456–11 463.

[8] B. Zhou, J. Xie, Y. Pan, J. Wu, and C. Lu, "Motionbev: Attention-aware online lidar moving object segmentation with bird's eye view based appearance and motion features," *IEEE Robotics and Automation Letters*, 2023.

[9] B. Mersch, X. Chen, I. Vizzo, L. Nunes, J. Behley, and C. Stachniss, "Receding moving object segmentation in 3d lidar data using sparse 4d convolutions," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7503–7510, 2022.

[10] N. Wang, C. Shi, R. Guo, H. Lu, Z. Zheng, and X. Chen, "Insmos: Instance-aware moving object segmentation in lidar data," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 7598–7605.

[11] K. Zeng, H. Shi, J. Lin, S. Li, J. Cheng, K. Wang, Z. Li, and K. Yang, "Mambamos: Lidar-based 3d moving object segmentation with motion-aware state space model," in *ACM International Conference on Multimedia (MM)*, 2024.

[12] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[13] Y. Hou, X. Zhu, Y. Ma, C. C. Loy, and Y. Li, "Point-to-voxel knowledge distillation for lidar semantic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 8479–8488.

[14] F. Jiang, H. Gao, S. Qiu, H. Zhang, R. Wan, and J. Pu, "Knowledge distillation from 3d to bird's-eye-view for lidar semantic segmentation," in *2023 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2023, pp. 402–407.

[15] X. Yan, J. Gao, C. Zheng, C. Zheng, R. Zhang, S. Cui, and Z. Li, "2dpass: 2d priors assisted semantic segmentation on lidar point clouds," in *European Conference on Computer Vision*. Springer, 2022, pp. 677–695.

[16] B. Zhao, Q. Cui, R. Song, Y. Qiu, and J. Liang, "Decoupled knowledge distillation," in *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, 2022, pp. 11 953–11 962.

[17] W. Lu, Y. Zhou, G. Wan, S. Hou, and S. Song, "L3-net: Towards learning based lidar localization for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6389–6398.

[18] J. H. Cho and B. Hariharan, "On the efficacy of knowledge distillation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 4794–4802.

[19] S. I. Mirzadeh, M. Farajtabar, A. Li, N. Levine, A. Matsukawa, and H. Ghasemzadeh, "Improved knowledge distillation via teacher assistant," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 04, 2020, pp. 5191–5198.

[20] X. Tang, Z. Chen, J. Cheng, X. Chen, J. Wu, and B. Xue, "Cv-mos: A cross-view model for motion segmentation," *IEEE Transactions on Instrumentation and Measurement*, 2024.

[21] W. Liu, H. Lu, H. Fu, and Z. Cao, "Learning to upsample by learning to sample," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 6027–6037.

[22] S. Mohapatra, M. Hodaei, S. Yogamani, S. Milz, H. Gotzig, M. Simon, H. Rashed, and P. Maeder, "Limoseg: Real-time bird's eye view based lidar motion segmentation," *arXiv preprint arXiv:2111.04875*, 2021.

[23] X. Zhu, H. Zhou, T. Wang, F. Hong, Y. Ma, W. Li, H. Li, and D. Lin, "Cylindrical and asymmetrical 3d convolution networks for lidar segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 9939–9948.

[24] J. Kim, J. Woo, and S. Im, "Rvmos: Range-view moving object segmentation leveraged by semantic and motion features," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8044–8051, 2022.

[25] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, "Semantickitti: A dataset for semantic scene understanding of lidar sequences," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9297–9307.

[26] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, pp. 303–338, 2010.

[27] X. Chen, B. Mersch, L. Nunes, R. Marcuzzi, I. Vizzo, J. Behley, and C. Stachniss, "Automatic labeling to generate training data for online lidar-based moving object segmentation," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6107–6114, 2022.

[28] S. Sun, W. Ren, J. Li, R. Wang, and X. Cao, "Logit standardization in knowledge distillation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 15 731–15 740.

[29] Z. Hao, J. Guo, K. Han, Y. Tang, H. Hu, Y. Wang, and C. Xu, "One-for-all: Bridge the gap between heterogeneous architectures in knowledge distillation," *Advances in Neural Information Processing Systems*, vol. 36, 2024.