PME3D: An Adaptive and Efficient Multi-modal Feature Extraction Plug-in for 3D Object Detection

Tianyi Yu[1]
2883406Y@student.gla.ac.uk
Lei Yu (corresponding author)[2]
yulei@buaa.edu.cn

[1] School of Computing Science
University of Glasgow
Glasgow, UK

[2] Sino-French Engineer School
Beihang University
Beijing, China

Abstract

3D object detection is fundamental to autonomous driving systems, where efficiently fusing multi-modal sensor data remains a critical challenge. Although current approaches predominantly focus on improving detection accuracy through Bird's Eye View representations, they often overlook computational efficiency, a crucial factor for real-world deployment on resource-constrained automotive platforms. To bridge this gap, we propose a lightweight plug-in module that enhances feature fusion efficiency through two key mechanisms: 1) dimensionality inversion of feature extraction outputs, and 2) dynamic selection of camera features for optimal fusion with point cloud data. Our experiments on nuScenes demonstrate that this approach maintains competitive detection performance while significantly reducing computational overhead, offering a practical solution for real-time autonomous driving applications.

1 Introduction

Autonomous driving systems require precise environmental perception to operate safely in complex real-world conditions. At the core of this capability lies 3D object detection, which extends beyond conventional 2D recognition by estimating the complete 3D properties of surrounding objects. This technology has become fundamental not only for autonomous vehicles but also for robotics and augmented reality applications [15, 16, 41, 51]. The ability to accurately determine object positions, orientations, and dimensions in real time enables critical decision making for collision avoidance and path planning.

The field has witnessed a significant shift toward multimodal sensor fusion approaches [5, 25] that combine data from complementary sensors such as LiDAR and cameras. Although this fusion paradigm offers substantial improvements in detection accuracy and robustness, it introduces several technical challenges that must be addressed. With an increasing number of modalities, the complexity in feature extraction and fusion will also increase. Therefore, the demand for computing power should also be correspondingly increased. However, in real-world scenarios, 3D multimodal perception is mainly applied to edge devices

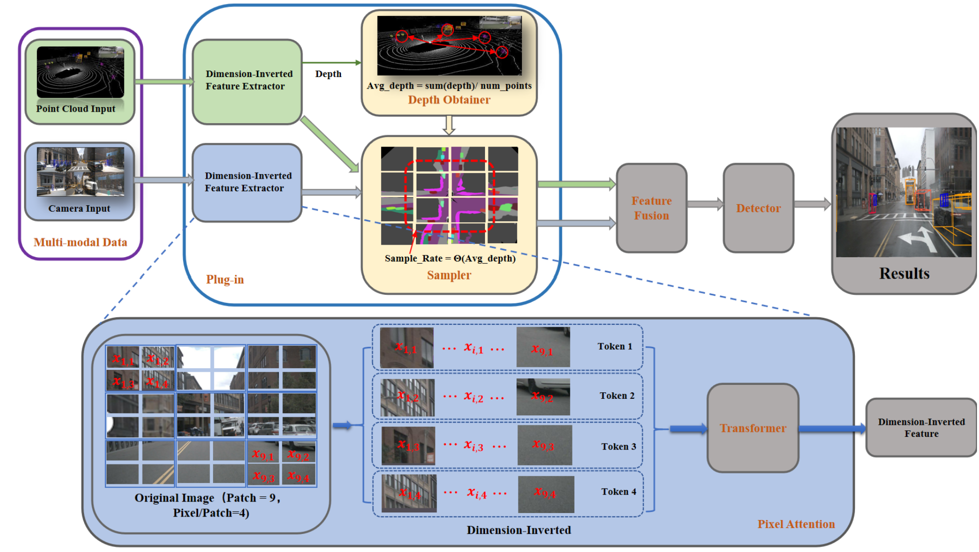

Figure 1: Our model processes camera, LiDAR, and Radar data via an inverted transformer architecture [9], extracting features along pixel dimensions instead of patches. A depth-guided sampler dynamically selects camera features using depth from point cloud features for fusion. This inverted flow and adaptive sampling ensure computational efficiency while maintaining detection accuracy, differing notably from conventional ViT-based approaches.

with limited computing power. At the same time, several research works [40], [7] have stated and shown the significance of efficiency in multimodal fusion in 3D Object Detection. However, they pose challenges, such as limitations in real-time autonomous driving deployment. Meanwhile, due to the use of different architectures in multi-modal detection models, finding a universal solution that significantly reduces computational complexity while maintaining model detection accuracy remains a challenge.

Considering that the main models [4, 27, 30] all follow the feature fusion prediction paradigm. We plan to achieve this by inserting a plug-in module into the feature extractor, which will replace the relevant features output by the original feature processor, resulting in more efficient feature fusion. We also proposed a sampler that can be plugged into the model before the fusion stage. These two parts make up the module design of our entire plug-in. The plug-in design enables seamless integration with existing detection pipelines without requiring architectural modifications. Our approach employs a plug-in module that directly replaces the original feature extractor while keeping the rest of the architecture unchanged. This design enables seamless integration with existing detection pipelines, eliminating the need for architectural modifications.

Our comprehensive evaluations of standard benchmarks demonstrate the effectiveness of this approach. The system maintains competitive detection accuracy while achieving substantial improvements in computational efficiency.

To the best of our knowledge, our contributions to the paper are as follows.

- **A Plug-in Solution for efficient 3D multi-modal Detection:**

  We propose a novel **plug-in** architecture for efficient multi-modal 3D detection that integrates seamlessly with existing systems. Unlike methods requiring complete redesigns, our solution enables portable adaptive fusion based on existing multi-modal 3D Object Detection models.

- **Self-Adaptive Sampling for feature fusion:**

  Our plug-in features a depth-guided sampler that dynamically adjusts fusion rates using LiDAR depth cues, prioritizing computation for complex regions while maintaining accuracy.

- **Pixel attention for multimodal feature extraction:**

  Our model uses Pixel Attention to replace the vanilla channel attention, enabling more effective modeling of global temporal dependencies. Reduce computational burden with minimal computational overhead. Its inherent multi-modal compatibility enables precise LiDAR-Camera fusion.

2 Related Work

Single-modality 3D Object Detection. The field of single-modality 3D object detection has evolved through three key phases. Early breakthroughs established foundational architectures: [8] for point clouds and [50] for voxel grids. Modality-specific innovations followed, with LiDAR methods such as [21] and [36], camera-based approaches, including [35], and radar systems such as [33]. The transformer era introduced [31] for LiDAR and [26] for camera detection, while [39] advanced monocular 3D detection through 2D-3D decoupling. Recent work focuses on deployment efficiency, with [22] enabling edge inference and [46] proposing a lightweight complementary depth module that achieves state-of-the-art performance in [10].

Multi-modality 3D Object Detection. Recent advances in multimodal 3D Object Detection have evolved from early fusion approaches like [5] and [20] to sophisticated transformer-based architectures such as [1] and [26], which leverage cross-modal attention mechanisms and bird's-eye-view representations to achieve significant performance gains. The field has also progressed with innovations such as [45] for improved coordinate encoding, [37] for Radar-Camera fusion, and [6], which introduced a unified framework for processing RGB, LiDAR, and radar data via modality-agnostic sampling. Emerging techniques focus on 4D radar integration [48] and dynamic computation routing [24], collectively pushing the state-of-the-art to 72.3% NDS on nuScenes while reducing latency to 50 ms for real-time applications. These developments highlight a clear path to robust, efficient, and scalable multisensor fusion systems capable of handling various scenarios of autonomous driving. In recent years, a considerable number of scientific papers have mentioned the efficiency issue of multimodal feature fusion. [7] claimed that a combination of camera and radar features, while maintaining the fidelity of each modality, is a challenge. They proposed a novel approach that fully utilizes the depth information from the radar feature to generate the refined fused feature. [14] addressed the under-utilization of LiDAR data in non-end-to-end monocular 3D detectors by introducing the CMKD network, which distills both feature- and response-level knowledge from LiDAR to the image modality. [17] addressed the challenge that the overfitting problem will occur when there is an improper annotation method in the data annotation process, and proposed that optimizing the algorithm structure by regressing the essence of data annotation will significantly improve performance. However, researchers rarely address the challenge of balancing real-time efficiency and precision.

3 Methods

Our proposed **PME3D** is a novel **P**lug-in **M**ulti-modal architecture designed for **E**fficient seamless integration with existing mainstream **3D** object detection models as illustrated in Figure 1. This modular component can be inserted into current detection pipelines (such as [6] or [45]).

The general pipeline of our model is illustrated in Figure 1. Our plug-in module is located after the multi-modal data and before the fusion stage. In detail, the plug-in mainly consists of three components: 1) Inverted Dimension Feature Extraction. 2) Depth Obtainer and 3) Sampler. The reason we dub our approach a "plug-in" is that it has been engineered for seamless drop-in integration into virtually any existing deep learning-based 3D multimodal fusion detector, for example [45], [6], without the need to redesign the overall architecture or modify the whole architecture. In practice, our module can be inserted between the frozen feature extractors of LiDAR and camera branches and the subsequent detection head, immediately inheriting all upstream representations while only fine-tuning a few lightweight parameters. This plug-and-play characteristic means that practitioners can upgrade their current pipelines - whether they are based on BEVFusion[30], BEVDet[16], TransFusion[1] or other famous frameworks whose architecture is similar to CMT and FUTR3D [45], [6], simply adding our component, re-optimizing for a handful of epochs.

3.1 Multi-modal Data

The system accepts three sensor modalities as input, with RGB images from vehicle-mounted cameras being mandatory while operating with either LiDAR or Radar point clouds, depending on availability. These raw inputs are processed through dedicated backbone networks: for camera data, ResNet-50 [12] with the Feature Pyramid Network (FPN) [28] is used to extract multiscale visual features, while for LiDAR/Radar point cloud processing, [47] is used to generate point cloud feature maps. The backbone networks output hierarchical feature representations at multiple scales, enabling the detection of objects across varying sizes and distances.

3.2 Feature Extractor

Our novel feature extraction module builds upon the standard Vision Transformer (ViT) framework [9] and introduces significant modifications to enhance efficiency. Pixel attention [43] focuses on the relationship between pixels within each patch, while traditional Vision Transformers (ViT) [9] concentrate on the relationship between different patches. In the context of radar or lidar perception, the primary targets are surrounding vehicles, which tend to be concentrated in one or several patches. Therefore, focusing on the relationship between pixels within a patch makes it more effective to extract vehicle characteristic information. The core innovation lies in the introduction of an inverted dimension processing scheme to achieve Pixel Attention via [29], thereby drastically reducing computational complexity. The specific implementation details are shown in the pixel attention section of Figure 1. Given an input tensor $X \in \mathbb{R}^{T \times N}$ where T represents the number of image patches and N denotes the number of pixels per patch, we strategically restructure the input as $\hat{X} \in \mathbb{R}^{N \times T}$. This dimensional inversion transforms the computation so that the complexity of self-attention is reduced from $\mathcal{O}(T^2)$ to $\mathcal{O}(N^2)$. For a typical configuration with 9 patches and 4 pixels per patch, this approach effectively decreases the computational load from $\mathcal{O}(81)$ to $\mathcal{O}(16)$ operations, representing a substantial efficiency improvement while maintaining the model's representational capacity.

3.3 Depth Estimation, Sampling and Fusion

The depth estimation module processes three parallel BEV feature streams from camera, LiDAR/Radar sensors. Based on the frameworks established in [6] and [26], we compute depth values through the coordinate transformation of the raw sensor data. The rotation matrix R and the translation vector t as shown in Equation 1 are derived using the **pyquaternion** library, enabling precise spatial alignment of multimodal features.

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = R \begin{pmatrix} x \\ y \\ z \end{pmatrix} + t \tag{1}$$

$$\bar{d} = \frac{1}{N} \sum_{i=1}^{N} d_i \tag{2}$$

Our self-adaptive sampling mechanism dynamically adjusts feature fusion rates based on scene depth complexity, the average depth $\bar{d}$ shown in Equation 1, effectively balancing computational efficiency with detection accuracy. The sampler preserves the native spatial precision of LiDAR and radar features while selectively resampling camera features according to the sampling rate $\phi(d)$ shown in Equation 3, where d_{max} means the maximum depth set in the current BEV coordinate, to improve semantic understanding. Zero-padding with PyTorch masking ensures efficient processing of unsampled regions.

$$\phi(d) = 1 - \left(\frac{d}{d_{max}} \right)^2 \tag{3}$$

The fusion module integrates the processed features through a multistage pipeline combining cross-modal attention mechanisms with spatial concatenation in BEV space. We evaluated both the conventional [6] fusion approach and our enhanced attention variant to demonstrate the effectiveness of our architectural improvements.

3.4 Loss Function

Our composite loss combines three key components: (1) a focal loss ($\gamma = 2$, $\alpha = 0.25$) to handle class imbalance, (2) Smooth L1 regression loss [11] for bounding box refinement, and (3) an IoU loss term (weight=1.0) for precise spatial alignment. Following the set prediction paradigm [41, 51], we employ bipartite matching with non-object padding to establish optimal prediction-to-ground truth assignments. This formulation maintains full compatibility with the host detectors [6, 45], while ensuring balanced gradient flow and real-time efficiency, as demonstrated in our ablation studies (Section 4).

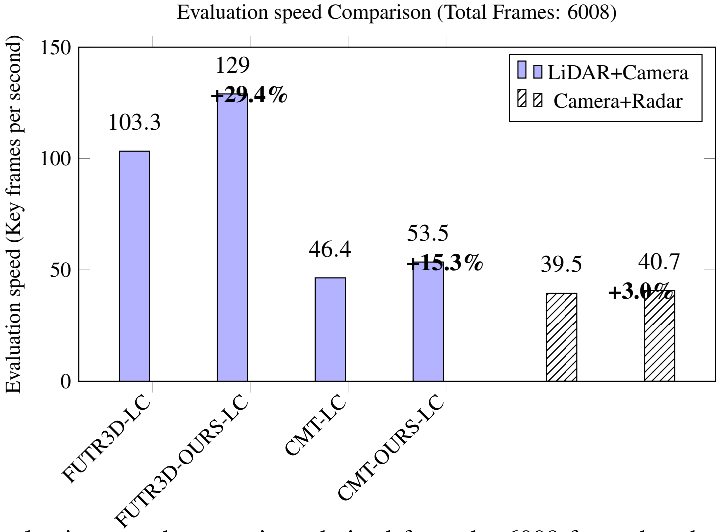

Figure 2: Evaluation speed comparison derived from the 6008-frame benchmark. (6008 indicates the total number of key frames in the nuScenes test set). Our method achieves 129 seconds in inference time (29.4% improvement) in LiDAR+Camera configuration when FUTR3D is integrated. Also, there is a 15.3% improvement when CMT is integrated. Percentages indicate inference efficiency gain over baselines.

4 Experiments

4.1 Dataset and Evaluation Protocol

We perform comprehensive evaluations on the nuScenes benchmark [3], a large-scale dataset that contains 1000 annotated scenes (700/150/150 split for train/val/test) with multimodal sensor data. Each 20-second sequence includes 40 keyframes with precise 3D annotations, allowing a rigorous assessment of temporal detection performance.

Table 1: The comparison experiments following the experiment above. We added the experiment on FPS and memory in the training stage for comparison.

Method	FPS-500	FPS-3500	FPS-5000	FPS-full
UVTR [23]	-	-	-	1.77
SparseFusion [24]	-	-	-	4.38
BEVFusion [30]	-	-	-	9.58
TransFusion [1]	-	-	-	6.51
DAL-Base [18]	-	-	-	10.69
CMT-R50 [45]	-	-	-	10.72
CMT-OURS	15.82	15.5	16.44	**24.90**

4.2 Evaluation Metrics

Our analysis employs only one official nuScenes metric: **NDS**, which evaluates the performance, error rate, and inference efficiency of the model. When evaluating the efficiency of

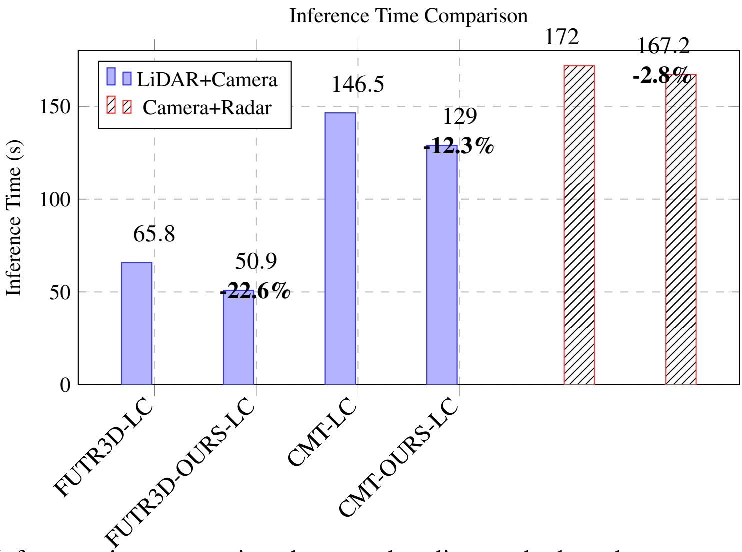

Figure 3: Inference time comparison between baseline methods and our approach. Negative percentages indicate the improvement over corresponding baselines. FUTR3D-OURS achieves 22.6% faster inference than FUTR3D in LiDAR+Camera configuration. Note that OURS indicates the model with our plug-in integrated into the baseline models.

each model, we introduce inference time. Inference time refers to the period that starts from the initiation of a test until the moment when the bounding boxes are successfully generated. Suppose that the model can perform the corresponding results in detection tasks, but gives a lower inference time. In that case, the model has lower complexity in detection tasks and, therefore, can be more adapted to real-time driving scenarios.

4.3 Implementation Details

All experiments were carried out on NVIDIA RTX 4090/3090 and A100 GPUs using CUDA 12.3. Only one GPU is used each time we train or test the model. We evaluated two state-of-the-art baseline architectures. CMT (Cross-modal Transformer) [45] and FUTR3D [6], selected for their representative fusion approaches.

Backbone Selection. All models employ ResNet [13] series backbones to ensure a fair comparison, leveraging their well-established advantages, including effective gradient flow through residual connections, robust multiscale feature extraction capabilities, balanced computational efficiency, and reproducible benchmarking through widespread adoption in the literature. This consistent selection of the backbone isolates the impact of our architectural innovations while maintaining compatibility with existing detection frameworks.

Sensor Configuration. Our experimental setup employs six 1600×900 RGB cameras (front/rear/left/right views), simulated 4-beam LiDAR point clouds via angular subsampling (range γ, inclination θ, azimuth ϕ), and filtered radar data (200-300 points/frame containing spatial coordinates, velocity, and intensity). Following [3]' methodology, all radar data was pre-processed using official filtering tools.

4.4 Training Details

We conducted experiments on nuScenes using [6] and [45] as our integration models. For [6], we trained separate LiDAR+Camera and Radar+Camera models (20 epochs each) on 4090/A100 hardware (Table 3), employing AdamW (lr= 1.0×10^{-4}), ResNet-101 backbone (256×704 input), and BEVDet-style augmentation [16]. For [45], we used ResNet50 with a learning rate equal to 3.5×10^{-5} (20 epochs per configuration), obtained based on the ratio of the original initial learning rate to the batch size of the training processes, following their original learning policy.

Table 2: Comprehensive Evaluation of Sensor Fusion Algorithms for 3D Object Detection on the nuScenes Test Dataset. CMT/FUTR3D-OURS indicates the model with our Plug-in module integrated.

Method	Modality	NDS ↑
3D-CVF[49]	L+C	62.3
PointPainting[38]	L+C	61.0
FUTR3D[6]	L+C	66.1
FUTR3D-OURS	L+C	66.3
MVP[42]	L+C	70.5
FusionPainting[44]	L+C	71.6
TransFusion[1]	L+C	71.7
CMT[45]	L+C	70.1
CMT-OURS	L+C	62.0
TransCAR[34]	C+R	52.2
CRAFT[19]	C+R	48.3
FUTR3D[6]	C+R	50.3
CenterFusion[32]	C+R	44.9
FUTR3D-OURS	C+R	48.5

We initially trained the model with both the fusion sampler and the iTransformer (using dimension inversion). Our ablation study removed these components to assess their individual contributions. Following [6], we set a depth limit of 50 m for LiDAR/Radar data based on the detector ranges. For CMT [45], we integrated our Self-adaptive Sampler and iTransformer, using the original data set splits. Inference benchmarks were conducted on the 6008-frame test set using trained checkpoints.

4.5 Results

Inference efficiency. As shown in Figure 2 (6008-frame benchmark) and 3, our method achieves substantial efficiency gains in all sensor configurations. For LiDAR+Camera (LC) fusion, we observe a 29.4% frame rate improvement over FUTR3D (129 vs. 103.3, along with a 22. 6% reduction in processing time (50.9 s vs. 65.8 s). For Camera+Radar (CR) mode, performance remains stable with a gain in evaluation speed of +3.0%. In particular, LC consistently outperforms CR in both metrics (129 vs. 40.7; 50.9 s vs. 167.2 s), underscoring LiDAR's value for real-time tasks.

To address inference-time inaccuracies caused by data loading, we introduce FPS as a new metric and evaluate on four nuScenes test subsets (500, 3500, 5000, 6000 key frames, selected sequentially to mitigate noise). As shown in Table 1, our CMT-OURS model achieves FPS values of 15.82, 15.5, 16.44, and 24.90 in subsets - ranking first among all methods. Unlike baselines that rely on key frames, CMT-OURS leverages full-frame data, demonstrating superior efficiency without sacrificing accuracy. Although a slight performance drop is observed in Table 2, our model maintains competitive accuracy while providing state-of-the-art efficiency (SOTA). These results validate our architecture's ability to balance temporal optimization and cross-modal compatibility, outperforming SOTA methods in both inference time and FPS.

Performance. Our approach achieves competitive LiDAR+Camera performance (66.3 NDS, comparable to FUTR3D 66.1 NDS) with 22.6% faster inference (50.9 s vs. 65.8 s). For CMT, we almost retain performance while accelerating by 12.3%. We selected FUTR3D and CMT for their balanced fusion strategies, excluding TransFusion [1] (LiDAR-centric limitations) and FusionPainting [44] (computationally intensive two-stage fusion). Using attention-based fusion without auxiliary networks, we balance accuracy and efficiency.

Performance drops during integration arise from real-time depth estimation errors and BEV-world coordinate misalignment. In camera-radar fusion, our plug-in module matches the accuracy of FUTR3D (38.9 NDS) with 2.8,s faster inference (167.2 s vs. 172 s). The 1.8-point NDS trade-off (48.5 vs. 50.3) represents a relative loss of 3.6% for efficiency gains. For LiDAR camera setups, we maintain +0.2 NDS accuracy with 22.6% faster inference (50.9 s vs. 65.8 s).

As shown in Table 2, TransCAR [34] is excluded due to its distinct early-level fusion (triple transformer decoders vs. single cross-modality decoders in CMT/FUTR3D), avoiding architectural incompatibility biases. These results validate our method's ability to optimize efficiency across sensor modalities with minimal accuracy trade-offs.

Table 3: We benchmark our plug-in module against FUTR3D and CMT baselines on the nuScenes test dataset, analyzing inference time and detection metrics. Controlled experiments ensure fair comparison by using identical hardware configurations. In this table, OURS means the model with our plug-in integrated. L+C represents LiDAR+Camera configuration, while C+R represents Camera+Radar configuration.

Method	NDS	NDS Gap	Modality	GPU	Time (s)	Time Gap (%)
FUTR3D	66.1	-	L+C	4090	65.8	-
FUTR3D-OURS	66.3	+0.30	L+C	4090	50.9	-22.6
CMT	70.8	-	L+C	3090	146.5	-
CMT-OURS	62.1	-12.1	L+C	3090	129	-12.3
FUTR3D	50.3	-	C+R	A100	172	-
FUTR3D-OURS	48.5	-3.58	C+R	A100	167	-2.8

Table 2 benchmarks our plug-in against FUTR3D and CMT on nuScenes, with controlled hardware. For LiDAR + camera (L + C): FUTR3D-OURS gains +0.30% NDS (66.3 vs 66.1) and -22.6% time (50.9s vs 65.8s, GPU 4090); CMT-OURS sacrifices 12.1% NDS (62.1 vs 70.8) for 12.3% time improvement (129s vs 146.5s, GPU 3090). For Camera+Radar (C+R, GPU A100): FUTR3D-OURS sees -3.58% NDS (48.5 vs 50.3) but -2.8% time (167s vs 172s). These show that our method balances efficiency (up to 22.6% speedup) and accuracy trade-offs, validating generalization ability across modalities and hardware, with NDS gaps quantifying precision-efficiency trade-offs.

4.6 Ablation Study

In this section, we discuss the impact of the Self-Adaptive Sampler (S) and the iTransformer (i) on the detection performance of our model, using ResNet50 (CMT) and ResNet101

(FUTR3D) as the backbone.

Table 4: Overall detection performance metrics, including NDS, and Time for models utilizing ResNet101 and ResNet50 backbones, conducted on the validation set, under different configurations of Self-adaptive Sampler (S) and iTransformer (i).

Modality	Baseline Model	S	i	NDS	Time (s)	Evaluation Speed (1/s)
L+C	FUTR3D	×	×	66.2	65.8	91.31
L+C	FUTR3D	×	✓	67.0	49.6	121.35
L+C	FUTR3D	✓	✓	66.3	50.9	118.25
R+C	FUTR3D	×	×	50.3	80.8	74.49
R+C	FUTR3D	×	✓	45.8	58.0	103.77
R+C	FUTR3D	✓	✓	48.5	77.2	78.0
L+C	CMT	×	×	70.3	146.5	41.1
L+C	CMT	×	✓	63.4	132	45.6
L+C	CMT	✓	✓	62.1	129.6	45.2
L+C	CMT	✓	×	61.5	129.1	46.4

Table 4 presents the overall detection performance (NDS) and efficiency metrics (time, evaluation speed) of the models, tested on the validation set under different configurations of S and i. For LiDAR + Camera (LC) fusion, the baseline FUTR3D achieves 66.2 NDS with neither S nor i, while allowing i alone increases NDS to 67.0 and cuts time from 65.8s to 49.6s; combining S and i slightly drops NDS to 66.3 but further reduces the time to 50.9 s. For Radar + Camera (R + C) fusion, FUTR3D sees the NDS fluctuate—from 50.3 (no S/i) to 45.8 (only i) and 48.5 (both S/i), with corresponding time changes (80.8, 58.0, 77.2 s). For CMT-based L + C models: The baseline (no S/i) hits 70.3 NDS but takes 146.5s; enabling i reduces NDS to 63.4 but cuts the time to 132s. Combining S and i further adjusts NDS (62.1/61.5) with a marginal time change (129.6s/129.1s). In general, iTransformer shows significant time saving potential in all modalities, while the interaction between S and i impacts NDS differently for the FUTR3D and CMT backbones, highlighting trade-offs between detection accuracy and inference efficiency under varying architectural configurations.

5 Conclusion and Future Work

We present a plug-in architecture for efficient multimodal 3D object detection fusion, combining a self-adaptive sampler with dimension-inverted transformers. Experimental results on nuScenes demonstrate that our method maintains competitive detection accuracy while achieving significant speed improvements across different sensor configurations and hardware platforms. The approach shows particular effectiveness in LiDAR-based detection while maintaining robust performance with radar data. Although improvement is limited, these advances in balancing accuracy and efficiency represent an important step toward practical real-time perception systems for autonomous driving. Future work includes extending to more datasets and real-scene applications. Also, we are considering extending the idea of self-adaptive sampling to a broader range of baseline models.

References

[1] Xuyang Bai, Zeyu Hu, Xinge Zhu, Qingqiu Huang, Yilun Chen, Hongbo Fu, and Chiew-Lan Tai. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers, 2022. URL https://arxiv.org/abs/2203.11496.

[2] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster, 2023. URL https://arxiv.org/abs/2210.09461.

[3] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. *CoRR*, abs/1903.11027, 2019. URL http://arxiv.org/abs/1903.11027.

[4] Yanlu Cai, Weizhong Zhang, Yuan Wu, and Cheng Jin. Fusionformer: A concise unified feature fusion transformer for 3d pose estimation. In Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan, editors, *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 900–908. AAAI Press, 2024. doi: 10.1609/AAAI.V38I2.27849. URL https://doi.org/10.1609/aaai.v38i2.27849.

[5] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving, 2017. URL https://arxiv.org/abs/1611.07759.

[6] Xuanyao Chen, Tianyuan Zhang, Yue Wang, Yilun Wang, and Hang Zhao. FUTR3D: A unified sensor fusion framework for 3d detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023 - Workshops, Vancouver, BC, Canada, June 17-24, 2023*, pages 172–181. IEEE, 2023. doi: 10.1109/CVPRW59228.2023.00022. URL https://doi.org/10.1109/CVPRW59228.2023.00022.

[7] Xiaomeng Chu, Jiajun Deng, Guoliang You, Yifan Duan, Houqiang Li, and Yanyong Zhang. Racformer: Towards high-quality 3d object detection via query-based radar-camera fusion, 2025. URL https://arxiv.org/abs/2412.12725.

[8] Longyuan Ding, Yong Cai, Jian Zhang, Ya Gao, Junjue Wang, Chenyu Zheng, Lei Lei, and Ailong Ma. Pointnet: Learning point representation for high-resolution remote sensing imagery land-cover classification. In *IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2021, Brussels, Belgium, July 11-16, 2021*, pages 4956–4959. IEEE, 2021. doi: 10.1109/IGARSS47720.2021.9554009. URL https://doi.org/10.1109/IGARSS47720.2021.9554009.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning*

Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

[10] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[11] Ross Girshick. Fast r-cnn, 2015. URL https://arxiv.org/abs/1504.08083.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.90. URL https://doi.org/10.1109/CVPR.2016.90.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.90. URL https://doi.org/10.1109/CVPR.2016.90.

[14] Yu Hong, Hang Dai, and Yong Ding. Cross-modality knowledge distillation network for monocular 3d object detection. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part X*, volume 13670 of *Lecture Notes in Computer Science*, pages 87–104. Springer, 2022. doi: 10.1007/978-3-031-20080-9_6. URL https://doi.org/10.1007/978-3-031-20080-9_6.

[15] Junjie Huang and Guan Huang. Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. *CoRR*, abs/2203.17054, 2022. doi: 10.48550/ARXIV.2203.17054. URL https://doi.org/10.48550/arXiv.2203.17054.

[16] Junjie Huang, Guan Huang, Zheng Zhu, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *CoRR*, abs/2112.11790, 2021. URL https://arxiv.org/abs/2112.11790.

[17] Junjie Huang, Yun Ye, Zhujin Liang, Yi Shan, and Dalong Du. Detecting as labeling: Rethinking lidar-camera fusion in 3d object detection, 2023. URL https://arxiv.org/abs/2311.07152.

[18] Junjie Huang, Yun Ye, Zhujin Liang, Yi Shan, and Dalong Du. Detecting as labeling: Rethinking lidar-camera fusion in 3d object detection. In Ales Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol, editors, *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part XXII*, volume 15080 of *Lecture Notes in Computer Science*, pages 439–455. Springer, 2024. doi: 10.1007/978-3-031-72670-5_25. URL https://doi.org/10.1007/978-3-031-72670-5_25.

[19] Youngseok Kim, Sanmin Kim, Jun Won Choi, and Dongsuk Kum. CRAFT: camera-radar 3d object detection with spatio-contextual fusion transformer. In Brian Williams, Yiling Chen, and Jennifer Neville, editors, *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications*

of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023, pages 1160–1168. AAAI Press, 2023. doi: 10.1609/AAAI.V37I1.25198. URL https://doi.org/10.1609/aaai.v37i1.25198.

[20] Jason Ku, Melissa Mozifian, Jungwook Lee, Ali Harakeh, and Steven Waslander. Joint 3d proposal generation and object detection from view aggregation, 2018. URL https://arxiv.org/abs/1712.02294.

[21] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds, 2019. URL https://arxiv.org/abs/1812.05784.

[22] Jinyu Li, Chenxu Luo, and Xiaodong Yang. Pillarnext: Rethinking network designs for 3d object detection in lidar point clouds, 2023. URL https://arxiv.org/abs/2305.04925.

[23] Yanwei Li, Yilun Chen, Xiaojuan Qi, Zeming Li, Jian Sun, and Jiaya Jia. Unifying voxel-based representation with transformer for 3d object detection, 2022. URL https://arxiv.org/abs/2206.00630.

[24] Yiheng Li, Hongyang Li, Zehao Huang, Hong Chang, and Naiyan Wang. Sparsefusion: Efficient sparse multi-modal fusion framework for long-range 3d perception, 2024. URL https://arxiv.org/abs/2403.10036.

[25] Yingwei Li, Adams Wei Yu, Tianjian Meng, Ben Caine, Jiquan Ngiam, Daiyi Peng, Junyang Shen, Bo Wu, Yifeng Lu, Denny Zhou, Quoc V. Le, Alan Yuille, and Mingxing Tan. Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection, 2022. URL https://arxiv.org/abs/2203.08195.

[26] Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. In Brian Williams, Yiling Chen, and Jennifer Neville, editors, *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pages 1477–1485. AAAI Press, 2023. doi: 10.1609/AAAI.V37I2.25233. URL https://doi.org/10.1609/aaai.v37i2.25233.

[27] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. *CoRR*, abs/2203.17270, 2022. doi: 10.48550/ARXIV.2203.17270. URL https://doi.org/10.48550/arXiv.2203.17270.

[28] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection, 2017. URL https://arxiv.org/abs/1612.03144.

[29] Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=JePfAI8fah.

[30] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L. Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *IEEE International Conference on Robotics and Automation, ICRA 2023, London, UK, May 29 - June 2, 2023*, pages 2774–2781. IEEE, 2023. doi: 10.1109/ICRA48891.2023.10160968. URL https://doi.org/10.1109/ICRA48891.2023.10160968.

[31] Ishan Misra, Rohit Girdhar, and Armand Joulin. An end-to-end transformer model for 3d object detection, 2021. URL https://arxiv.org/abs/2109.08141.

[32] Ramin Nabati and Hairong Qi. Centerfusion: Center-based radar and camera fusion for 3d object detection. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, page 1526–1535. IEEE, January 2021. doi: 10.1109/wacv48630.2021.00157. URL http://dx.doi.org/10.1109/WACV48630.2021.00157.

[33] Felix Nobis, Maximilian Geisslinger, Markus Weber, Johannes Betz, and Markus Lienkamp. A deep learning-based radar and camera sensor fusion architecture for object detection, 2020. URL https://arxiv.org/abs/2005.07431.

[34] Su Pang, Daniel D. Morris, and Hayder Radha. Transcar: Transformer-based camera-and-radar fusion for 3d object detection. In *IROS*, pages 10902–10909, 2023. doi: 10.1109/IROS55552.2023.10341793. URL https://doi.org/10.1109/IROS55552.2023.10341793.

[35] Dennis Park, Rares Ambrus, Vitor Guizilini, Jie Li, and Adrien Gaidon. Is pseudo-lidar needed for monocular 3d object detection?, 2021. URL https://arxiv.org/abs/2108.06417.

[36] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection, 2021. URL https://arxiv.org/abs/1912.13192.

[37] Lukas Stäcker, Shashank Mishra, Philipp Heidenreich, Jason R. Rambach, and Didier Stricker. Rc-bevfusion: A plug-in module for radar-camera bird's eye view feature fusion. In Ullrich Köthe and Carsten Rother, editors, *Pattern Recognition - 45th DAGM German Conference, DAGM GCPR 2023, Heidelberg, Germany, September 19-22, 2023, Proceedings*, volume 14264 of *Lecture Notes in Computer Science*, pages 178–194. Springer, 2023. doi: 10.1007/978-3-031-54605-1_12. URL https://doi.org/10.1007/978-3-031-54605-1_12.

[38] Sourabh Vora, Alex H. Lang, Bassam Helou, and Oscar Beijbom. Pointpainting: Sequential fusion for 3d object detection. *CoRR*, abs/1911.10150, 2019. URL http://arxiv.org/abs/1911.10150.

[39] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. FCOS3D: fully convolutional one-stage monocular 3d object detection. In *IEEE/CVF International Conference on Computer Vision Workshops, ICCVW 2021, Montreal, BC, Canada, October 11-17, 2021*, pages 913–922. IEEE, 2021. doi: 10.1109/ICCVW54120.2021.00107. URL https://doi.org/10.1109/ICCVW54120.2021.00107.

[40] Yiran Wang, Jiaqi Li, Chaoyi Hong, Ruibo Li, Liusheng Sun, Xiao Song, Zhe Wang, Zhiguo Cao, and Guosheng Lin. Tacodepth: Towards efficient radar-camera depth estimation with one-stage fusion, 2025. URL https://arxiv.org/abs/2504.11773.

[41] Yue Wang, Vitor Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. DETR3D: 3d object detection from multi-view images via 3d-to-2d queries. In Aleksandra Faust, David Hsu, and Gerhard Neumann, editors, *Conference on Robot Learning, 8-11 November 2021, London, UK*, volume 164 of *Proceedings of Machine Learning Research*, pages 180–191. PMLR, 2021. URL https://proceedings.mlr.press/v164/wang22b.html.

[42] Longhui Wei, Lingxi Xie, Wengang Zhou, Houqiang Li, and Qi Tian. Mvp: Multimodality-guided visual pre-training, 2022. URL https://arxiv.org/abs/2203.05175.

[43] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. CBAM: convolutional block attention module. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VII*, volume 11211 of *Lecture Notes in Computer Science*, pages 3–19. Springer, 2018. doi: 10.1007/978-3-030-01234-2_1. URL https://doi.org/10.1007/978-3-030-01234-2_1.

[44] Shaoqing Xu, Dingfu Zhou, Jin Fang, Junbo Yin, Bin Zhou, and Liangjun Zhang. Fusionpainting: Multimodal fusion with adaptive attention for 3d object detection. *CoRR*, abs/2106.12449, 2021. URL https://arxiv.org/abs/2106.12449.

[45] Junjie Yan, Yingfei Liu, Jianjian Sun, Fan Jia, Shuailin Li, Tiancai Wang, and Xiangyu Zhang. Cross modal transformer: Towards fast and robust 3d object detection. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 18222–18232. IEEE, 2023. doi: 10.1109/ICCV51070.2023.01675. URL https://doi.org/10.1109/ICCV51070.2023.01675.

[46] Longfei Yan, Pei Yan, Shengzhou Xiong, Xuanyu Xiang, and Yihua Tan. Monocd: Monocular 3d object detection with complementary depths, 2024. URL https://arxiv.org/abs/2404.03181.

[47] Yan Yan, Yuxing Mao, and Bo Li. SECOND: sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018. doi: 10.3390/S18103337. URL https://doi.org/10.3390/s18103337.

[48] Yanlong Yang, Jianan Liu, Tao Huang, Qing-Long Han, Gang Ma, and Bing Zhu. Ralibev: Radar and lidar bev fusion learning for anchor box free object detection systems, 2024. URL https://arxiv.org/abs/2211.06108.

[49] Jin Hyeok Yoo, Yecheol Kim, Ji Song Kim, and Jun Won Choi. 3d-cvf: Generating joint camera and lidar features using cross-view spatial feature fusion for 3d object detection. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXVII*, volume 12372 of *Lecture Notes in Computer Science*, pages 720–736. Springer, 2020. doi: 10.1007/978-3-030-58583-9_43. URL https://doi.org/10.1007/978-3-030-58583-9_43.

[50] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 4490–4499. Computer Vision Foundation / IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00472. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Zhou_VoxelNet_End-to-End_Learning_CVPR_2018_paper.html.

[51] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: deformable transformers for end-to-end object detection. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=gZ9hCDWe6ke.