Height-Fidelity Dense Global Fusion for Multi-modal 3D Object Detection

Supplementary Material

A. Overview

We begin by outlining the structure of this supplementary material. §B elaborates more details of Mamba and discusses the difference between Mamba and transformer. §C supplements more details of multi-modality representation learning. §D delves deeper into the analysis of other components within our framework, providing both qualitative and quantitative evaluations of our Modality Aligner (HMB-M) and BEV Space Fusion (HMB-B). §E provides results of different resolutions and backbones. §F presents an efficiency analysis based on parameter count and inference speed. §G reports further results on the nuScenes test dataset, including additional metrics such as mASE and a detailed per-category mAP. §H delivers extensive evaluations on Waymo and BEV segmentation that underscore the generalizability of our method.

B. More Details of Mamba

Long-range modeling of Mamba. Rooted in linear systems theory [9], the State Space Model (SSM) [4, 5, 7, 8] provides a robust framework for representing dynamical systems. The continuous-time SSM is:

$$\mathbf{h}'(t) = \mathbf{A}\mathbf{h}(t) + \mathbf{B}\mathrm{x}(t), \tag{1}$$

$$\mathbf{y}(t) = \mathbf{C}^{\top}\mathbf{h}(t) + \mathbf{D}\mathbf{x}(t), \tag{2}$$

where $\mathbf{h}(t) \in \mathbb{R}^C$ is the hidden state, $\mathbf{y}(t) \in \mathbb{R}^L$ is the output, $\mathbf{A}$ governs system dynamics, $\mathbf{B}$ insert the input $\mathbf{x}(t)$ to the state, $\mathbf{C}$ projects the state to output, and $\mathbf{D}$ allows residual connection. To make continuous-time models feasible for real-world applications, they are approximated in discrete time using matrices $\overline{\mathbf{A}}$ and $\overline{\mathbf{B}}$, derived from their continuous counterparts with time step Δ. After that, Mamba [5] introduces an input-dependent selection mechanism, enabling the system to adaptively extract information from the input sequence. In particular, $\overline{\mathbf{B}}$, $\overline{\mathbf{C}}$, and $\boldsymbol{\Delta}$ are predicted with discrete input $\mathbf{x}_t$. The matrix $\overline{\mathbf{A}}$, derived from HiPPO [6], effectively captures long-range dependencies by transforming global features into a compressed representation. $\overline{\mathbf{A}}$ is a matrix with following structure:

$$A_{nk} = \begin{cases} \sqrt{(2n+1)}\sqrt{(2k+1)}, & \text{if } n > k \\ n+1, & \text{if } n = k \\ 0, & \text{if } n < k \end{cases} \tag{3}$$

HiPPO enjoys favorable theoretical properties: it is invariant to input space/time scale [6]. In point cloud processing, non-uniform spatial intervals arise from occlusion and varying distances, complicating the application of many linear attention mechanisms. Such long-range context modeling broadens the scope of contextual reasoning, facilitating accurate discrimination between high-semantic distractors and targets and mitigating spurious or missed detections arising from occlusion or other challenges.

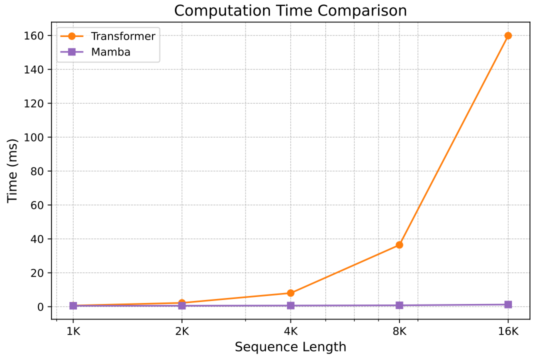


Figure 1. **Time comparison between Mamba and Transformer.**

Content-based v.s. Position-based. A content-based interaction focuses on the content of the context. In contrast, a position-based interaction emphasizes the relationship between the query position and the context position. Unlike the Transformer, which employs content-based interaction, Mamba utilizes position-based interaction, which relies more heavily on positional information. This distinction underscores the importance of accurate spatial cues, thus motivating our design of Height-Fidelity LiDAR Encoding and the Hybrid Mamba Block.

Computational Complexity. Fig. 1 demonstrates that as the token number surpasses 4K, the computational cost per transformer layer escalates sharply, rendering it impractical for perception tasks involving more than 16K tokens. In contrast, the Mamba method exhibits remarkable scalability, maintaining a nearly constant processing time even when handling sequences of up to 16K tokens.

C. Multi-modality Representation Learning

This section provides a detailed description of the development of a fusion framework incorporating the proposed modules. Specifically, it covers: 1) the use of the HMB to align the feature distributions of the two modalities, 2) the application of height-fidelity point cloud features to construct fusion in the raw 3D space using HMB, and 3) the integration of features in the BEV space through with HMB.

Modality Aligner. The point cloud and image modalities differ fundamentally in both their raw data structures and the characteristics of the features they extract. As a result, directly fusing these heterogeneous features can lead to suboptimal performance. To address this issue, we propose routing the features from both modalities through a shared Hybrid Mamba Block (HMB), allowing them to exchange information and align their feature distributions. This process is akin to the behavior of ”normalization” layers.

$$\hat{\mathbf{F}}_{\mathrm{L}}, \hat{\mathbf{F}}_{\mathrm{I}} = \mathrm{HybridMamba}(\mathbf{F}_{\mathrm{L}},\ \mathbf{F}_{\mathrm{I}},\ \mathcal{C}_{\mathrm{P}},\ \mathcal{C}_{\mathrm{I}}), \tag{4}$$

where $\mathcal{C}_{\mathrm{I}} \in \mathbb{R}^{N_{\mathrm{img}} \times W \times H \times 2}$ is coordinates of image features $\mathbf{F}_{\mathrm{L}} \in \mathbb{R}^{N_{\mathrm{voxel}} \times C}$, $\mathcal{C}_{\mathrm{P}} \in \mathbb{R}^{N_{\mathrm{voxel}} \times 3}$ is 3D coordinates of voxel features $\mathbf{F}_{\mathrm{I}} \in \mathbb{R}^{N_{\mathrm{img}} \times W \times H \times C}$.

Raw Space Fusion. After obtaining the features of consistent distribution, it is imperative to establish a unified coordinate system to fuse them. We first conduct fusion in the image coordinate system, in which we project the 3D voxel features with coordinates in raw space to the image planes with the transformation matrices $\mathbf{E}_{\mathrm{P}\rightarrow\mathrm{I}} \in \mathbb{R}^{N_{\mathrm{img}} \times 4 \times 4}$. Specifically, we can get the projected coordinates of point cloud features $\mathcal{C}^{\mathrm{j}}_{\mathrm{P}\rightarrow\mathrm{I}}$ on the j-th image plane,

$$\mathcal{C}^{\mathrm{j}}_{\mathrm{P}\rightarrow\mathrm{I}} = \mathbf{E}^{\mathrm{j}}_{\mathrm{P}\rightarrow\mathrm{I}} \mathcal{C}_{\mathrm{P}}. \tag{5}$$

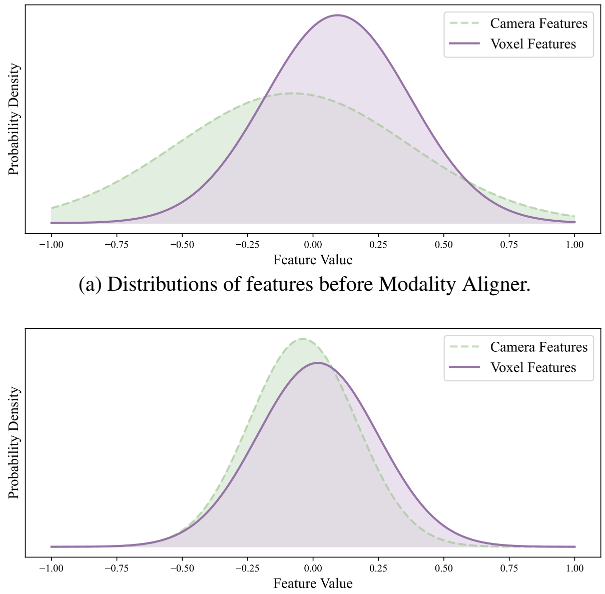
By establishing this unified coordinate representation, we facilitate the seamless integration of features from both modalities within our HMB,

$$\tilde{\mathbf{F}}_{\mathrm{L}}, \tilde{\mathbf{F}}_{\mathrm{I}} = \mathrm{HybridMamba}(\hat{\mathbf{F}}_{\mathrm{L}}, \hat{\mathbf{F}}_{\mathrm{I}}, \mathcal{C}_{\mathrm{P}\rightarrow\mathrm{I}}, \mathcal{C}_{\mathrm{I}}). \tag{6}$$

BEV Space Fusion. We also apply the proposed Hybrid Mamba Block (HMB) to fuse the visual and point cloud features in the BEV space, with the goal of generating a unified multi-modal representation for the subsequent detection head. Specifically, we employ the Lift-Splat-Shoot (LSS) transformation [14] to project the multi-view image features into the BEV space, obtaining $\tilde{\mathbf{F}}_{\mathrm{I}\rightarrow\mathrm{B}} \in \mathbb{R}^{W_{\mathrm{b}} \times H_{\mathrm{b}} \times C}$. Given that many positions in the BEV space lack valuable information, therefore, to ensure computational efficiency, we downsample the spatial resolution of the visual BEV features by half. The point cloud features are also transformed to BEV space. It is worth noting that we only remove the height information of the sparse point cloud features and do not fill the vacant positions in the BEV space, thus no additional computation is introduced. Finally, we concatenate the BEV features of the two modalities and fuse their information with our proposed HMB,

$$\mathbf{F}_{\mathrm{bev}} = \mathrm{HybridMamba}(\tilde{\mathbf{F}}_{\mathrm{P}\rightarrow\mathrm{B}}, \tilde{\mathbf{F}}_{\mathrm{I}\rightarrow\mathrm{B}}, \mathcal{C}_{\mathrm{P}\rightarrow\mathrm{B}}, \mathcal{C}_{\mathrm{I}\rightarrow\mathrm{B}}), \tag{7}$$

after which $\mathbf{F}_{\mathrm{bev}}$ is fed into the detection head.



(a) Distributions of features before Modality Aligner.

(b) Distributions of features after Modality Aligner.

Figure 2. Visualization analysis of Modality Aligner.

	Modality Aligner	mAP	NDS
①	Without Aligner	71.2	73.7
②	LayerNorm	71.0	73.2
③	HMB-M	71.9	74.3

Table 1. Ablation on different Modality Aligner.

D. Analysis of Other Components.

Analysis of the Modality Aligner. We evaluate the efficacy of our feature alignment module through both visualization and empirical experiments. As illustrated in Fig. 2, we present the feature distributions obtained from the initial modality tokenizers and compare them to the distributions after applying the Modality Aligner. Owing to the intrinsic differences between the two modalities, their feature distributions are initially disparate, as shown in Fig. 2(a). However, effective feature fusion necessitates that these features reside within similar distributions to facilitate seamless integration. To achieve this alignment, we use the Modality Aligner, which brings the distributions of the two modalities closer together. As depicted in Fig. 2(b), the distributions of voxel and camera features become significantly more aligned after alignment. This visualization highlights the capability of our proposed component to effectively harmonize feature spaces across modalities. Beyond visual analysis, we conduct quantitative experiments to substantiate the effectiveness of the Modality Aligner. According to Tab. 1, incorporating the Modality Aligner enhances performance metrics from 71.2 mAP and 73.7 NDS to 71.9

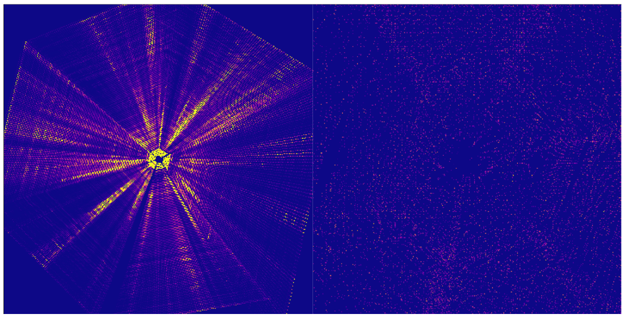


Figure 3. Visualization of camera features in BEV space before with LSS (left) and with UniTR approach (right).

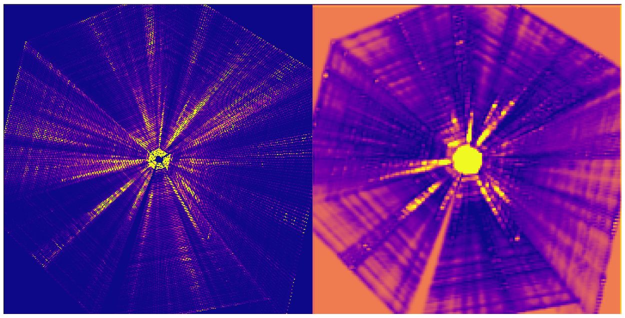


Figure 4. Visualization of camera features in BEV space before (left) and after (right) downsample operation.

mAP and 74.3 NDS (① *vs.* ③). Furthermore, we observe that simply applying LayerNorm to unify distributions not only fails to yield improvements but actually degrades performance (① *vs.* ②). This underscores that our Modality Aligner offers a more sophisticated and effective approach to feature alignment than basic normalization techniques.

Analysis of BEV Space Fusion. As illustrated in Fig. 3, the LSS method yields more precise feature localization in BEV space compared to the original approaches utilized in UniTR. However, mapping image features into BEV space via LSS substantially increases the number of features, resulting in a highly sparse representation with numerous zero-valued positions. To enhance the efficacy of feature fusion, we apply downsample to the BEV features prior to the fusion process. As shown in Fig. 4, this downsample ameliorates the sparsity issue by increasing the information density of the features. Nonetheless, excessive downsampling can blur object boundaries, thereby degrading performance. Tab 2 demonstrates that a downsample stride of 2 achieves optimal performance, underscoring the crucial importance of selecting an appropriate downsample stride for efficient feature fusion in our model.

	Downsample Stride	mAP	NDS
①	Without HMB-B	71.4	73.6
②	1	71.3	73.8
③	2	71.9	74.3
④	4	71.0	73.5

Table 2. Ablation on different downsample stride.

Resolution	mAP	NDS
1056 × 384	72.3	74.5
704 × 256	71.9	74.3

Resolution	HFL	mAP	NDS
Swin-T+VoxelNet	✗	70.2	72.7
Swin-T+VoxelNet	✓	71.7	74.0
Ours	✗	70.3	73.1
Ours	✓	71.9	74.3

Table 3. Different resolutions(left) and backbones(right).

E. Different Resolutions and Backbones.

We evaluated how image resolution and backbone design affect overall performance. As shown in Tab. 3(left), raising the resolution by a factor of 1.5 results in a 2.25-fold increase in the number of tokens, yet yields only marginal performance improvements. This observation highlights the advantage of adopting a lower resolution to enhance computational efficiency. Furthermore, Tab. 3(right) reveals that the primary performance gains originate from Height-Fidelity LiDAR Encoding (HFL) and our fusion strategy, rather than from modifications to the backbone. These findings emphasize the efficacy of our specialized feature representation and fusion design.

	BEVFusion (MIT)	DAL	IS-Fusion	FusionMamba (Ours)
Params(M)	40.84	47.77	48.32	30.47
FPS	4.2	4.3	3.2	4.7

Table 4. Params(M) and FPS with current SOTA methods.

F. Efficiency Analysis

Tab. 4 highlights the efficiency of our method in terms of both parameter count and inference speed. The results demonstrate that FusionMamba achieves state-of-the-art performance without relying on a larger model or sacrificing inference speed, underscoring that our performance gains stem from a more effective design rather than increased model complexity. Additionally, its compact and efficient nature facilitates better transferability to other tasks, enabling broader applicability.

Parameter Efficiency: FusionMamba maintains a lightweight architecture with only 30.47M, which is significantly lower than BEVFusion (40.84M), DAL (47.77M), and IS-Fusion (48.32M). FusionMamba achieves superior results without excessive parameter growth, demonstrating that efficient feature representation and fusion strategies play a more crucial role than sheer scale.

Inference Speed: FusionMamba delivers an inference speed of 4.7 FPS, outperforming the other methods (BEVFusion at 4.2 FPS, DAL at 4.3 FPS, and IS-Fusion at 3.2

Method	Present at	mATE	mASE	mAOE	mAVE	mAAE	mAP	NDS
PointPainting [15]	CVPR'20	38.0	26.0	54.1	29.3	13.1	54.1	61.0
PointAugmenting [16]	CVPR'21	25.3	23.5	35.4	26.6	12.3	66.8	71.1
TransFusion [1]	CVPR'22	25.9	24.3	32.9	28.8	12.7	68.9	71.7
AutoAlignV2 [3]	ECCV'22	24.5	23.3	31.1	25.8	13.3	68.4	72.4
UVTR [11]	NeurIPS'22	30.6	24.5	35.1	22.5	12.4	67.1	71.1
BEVFusion (MIT) [12]	ICRA'23	26.1	23.9	32.9	26.0	13.4	70.2	72.9
DeepInteraction [19]	NeurIPS'22	25.7	24.0	32.5	24.5	12.8	70.8	73.4
BEVFusion (ADLab) [13]	NeurIPS'22	25.0	24.0	35.9	25.4	13.2	71.3	73.3
CMT [18]	ICCV'23	27.9	23.5	30.8	25.9	11.2	72.0	74.1
SparseFusion [22]	ICCV'23	25.8	24.3	32.9	26.5	13.1	72.0	73.8
DAL [10]	ECCV'24	25.3	23.9	33.4	17.4	12.0	72.0	74.8
UniTR [17]	ICCV'23	24.1	22.9	25.6	24.0	13.1	70.9	74.5
MambaFusion-Lite	Ours	23.7	22.7	27.8	22.4	13.0	72.0	75.0
MambaFusion-Base	Ours	23.3	22.3	26.8	21.5	13.2	73.2	75.9

Table 5. Comparisons on nuScenes test dataset. We present the Average Translation Error (mATE), mean Average Scale Error (mASE), mean Average Orientation Error (mAOE), mean Average Velocity Error (mAVE), and mean Average Attribute Error (mAAE) of each method. All present methods use a single model without any test time augmentation.

Method	Modality	mAP	NDS	Car	Truck	C.V.	Bus	T.L.	B.R.	M.T.	Bike	Ped.	T.C.
PointAugmenting [16]	L+C	66.8	71.0	87.5	57.3	28.0	65.2	60.7	72.6	74.3	50.9	87.9	83.6
MVP [21]	L+C	66.4	70.5	86.8	58.5	26.1	67.4	57.3	74.8	70.0	49.3	89.1	85.0
TransFusion [1]	L+C	68.9	71.7	87.1	60.0	33.1	68.3	60.8	78.1	73.6	52.9	88.4	86.7
AutoAlignV2 [3]	L+C	68.4	72.4	87.0	59.0	33.1	69.3	59.3	78.0	72.9	52.1	87.6	85.1
UVTR [11]	L+C	67.1	71.1	87.5	56.0	33.8	67.5	59.5	73.0	73.4	54.8	86.3	79.6
BEVFusion (PKU) [12]	L+C	69.2	71.8	88.1	60.9	34.4	69.3	62.1	78.2	72.2	52.2	89.2	85.5
DeepInteraction [19]	L+C	70.8	73.4	87.9	60.2	37.5	70.8	63.8	80.4	75.4	54.5	91.7	87.2
BEVFusion (MIT) [13]	L+C	70.2	72.9	88.6	60.1	39.3	69.8	63.8	80.0	74.1	51.0	89.2	86.5
CMT [18]	L+C	72.0	74.1	88.0	63.3	37.3	75.4	65.4	78.2	79.1	60.6	87.9	84.7
SparseFusion [22]	L+C	72.0	73.8	88.0	60.2	38.7	72.0	64.9	79.2	78.5	59.8	90.9	87.9
ObjectFusion [2]	L+C	71.0	73.3	89.4	59.0	40.5	71.8	63.1	80.0	78.1	53.2	90.7	87.7
IS-FUSION [20]	L+C	73.0	75.2	88.3	62.7	38.4	74.9	67.3	78.1	82.4	59.5	89.3	89.2
DAL [10]	L+C	72.0	74.8	89.1	60.2	34.6	73.3	65.8	80.6	81.7	58.5	89.6	86.6
UniTR [17]	L+C	70.9	74.5	87.9	60.2	39.2	72.2	65.1	76.8	75.8	52.2	89.4	89.7
FusionMamba-Lite	L+C	72.0	75.0	88.9	62.5	37.1	74.1	66.4	73.3	78.8	58.0	90.9	90.2
FusionMamba-Base	L+C	73.2	75.9	89.4	63.8	39.1	75.1	68.0	75.0	80.7	59.5	91.4	90.6

Table 6. Comparison of different methods on mAP, NDS, and mAP of varying object categories. 'C.V.', 'T.L.', 'B.R.', 'M.T.', 'Ped.', and 'T.C.' indicate the construction vehicle, trailer, barrier, motorcycle, pedestrian, and traffic cone, respectively.

FPS). This efficiency ensures that our model maintains high processing speed while achieving superior performance.

G. More Results of Test Set.

To rigorously evaluate perception algorithms in autonomous driving, the nuScenes dataset provides a comprehensive suite of metrics. The mean Average Translation Error (mATE) quantifies the average positional discrepancy between predicted objects and ground truth, assessing localization accuracy. The mean Average Scale Error (mASE) measures errors in estimating object sizes, reflecting the precision of scale estimation. The mean Average Orientation Error (mAOE) evaluates discrepancies in orientation angles, indicating the model's ability to accurately predict object headings. The mean Average Velocity Error (mAVE) calculates errors in velocity predictions, providing insights into the accuracy of motion estimation for dynamic objects. Lastly, the mean Average Attribute Error (mAAE) assesses errors in predicting object attributes, such as activity or state, thus evaluating the model's capability to infer additional semantic information. Collectively, these metrics offer a multidimensional evaluation framework that captures various critical aspects of perception performance in autonomous driving systems.

As presented in Tab 5, our proposed method surpasses existing approaches, particularly in mATE and mASE, indicating superior accuracy in estimating object locations and sizes. Moreover, our method achieves higher NDS scores, reflecting enhanced overall performance. Additionally, as shown in Tab 6, we report the mAP scores across different categories, where our method attains top-tier performance in most categories. These results substantiate the effectiveness and robustness of our approach.

H. Extended Experiments

Method	Modality	mAP/mAPH (Waymo L2)	mAP (Argo.)
Baseline-L (UniTR)	L	74.0/72.1	38.6
UniTR	L+C	74.9/73.6	41.2
Baseline-L (Ours)	L	74.4/72.5	39.1
MambaFusion (ours)	L+C	**76.5/75.4**	**43.3**

Table 7. Results on Waymo and Argoverse2.

Extend to Waymo Dataset. Beyond our evaluation of the nuScenes dataset, which employs a 32-beam LiDAR system, we extend our approach to the Waymo dataset which is one of the most extensive and challenging benchmarks in autonomous driving, offering 64-beam LiDAR data and diverse environmental conditions. As detailed in Tab. **??**, our method consistently improves detection performance on Waymo. In experiments using 20% of the Waymo data, our single-modality baseline (Baseline-L (Ours)) achieves mAP/mAPH scores of 71.3/69.8, closely matching the LiDAR-only baseline of UniTR (71.2/69.5). When incorporating the camera modality, the UniTR method attains scores of 72.3/70.9, whereas our proposed MambaFusion method significantly enhances performance to 73.7/72.4. This enhancement underscores the strong generalization capability of our multi-modal fusion strategy, demonstrating its applicability across diverse autonomous driving datasets.

Method	Mean IoU
UniTR	63.7
MambaFusion (ours)	65.3

Table 8. BEV map segmentation results (25% data).

Extend to BEV Segmentation. Improved extraction of global contextual information is also essential for accurate BEV segmentation. In the BEV segmentation conducted on 25% of the nuScenes, our approach attains a Mean Intersection-over-Union (IoU) of 65.3, surpassing the 63.7 achieved by UniTR. This advancement demonstrates that our fusion method efficiently captures comprehensive global features, thereby significantly enhancing segmentation precision, particularly regarding boundary delineation and holistic scene understanding.

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. In *CVPR*, pages 1090–1099, 2022. 4

[2] Qi Cai, Yingwei Pan, Ting Yao, Chong-Wah Ngo, and Tao Mei. Objectfusion: Multi-modal 3d object detection with object-centric fusion. In *ICCV*, pages 18067–18076, 2023. 4

[3] Zehui Chen, Zhenyu Li, Shiquan Zhang, Liangji Fang, Qinhong Jiang, and Feng Zhao. Deformable feature aggregation for dynamic multi-modal 3d object detection. In *ECCV*, pages 628–644. Springer, 2022. 4

[4] Tri Dao and Albert Gu. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality. *arXiv preprint arXiv:2405.21060*, 2024. 1

[5] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023. 1

[6] Albert Gu, Tri Dao, Stefano Ermon, Atri Rudra, and Christopher Ré. Hippo: Recurrent memory with optimal polynomial projections. *NIPS*, 33:1474–1487, 2020. 1

[7] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021. 1

[8] Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *NeurIPS*, 34:572–585, 2021. 1

[9] Joao P Hespanha. *Linear systems theory*. Princeton university press, 2018. 1

[10] Junjie Huang, Yun Ye, Zhujin Liang, Yi Shan, and Dalong Du. Detecting as labeling: Rethinking lidar-camera fusion in 3d object detection. In *ECCV*, pages 439–455. Springer, 2025. 4

[11] Yanwei Li, Yilun Chen, Xiaojuan Qi, Zeming Li, Jian Sun, and Jiaya Jia. Unifying voxel-based representation with transformer for 3d object detection. *NeurIPS*, 35:18442–18455, 2022. 4

[12] Tingting Liang, Hongwei Xie, Kaicheng Yu, Zhongyu Xia, Zhiwei Lin, Yongtao Wang, Tao Tang, Bing Wang, and Zhi Tang. Bevfusion: A simple and robust lidar-camera fusion framework. *NeurIPS*, 35:10421–10434, 2022. 4

[13] 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. pages 2774–2781. IEEE, 2023. 4

[14] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *ECCV*, pages 194–210. Springer, 2020. 2

[15] Sourabh Vora, Alex H Lang, Bassam Helou, and Oscar Beijbom. Pointpainting: Sequential fusion for 3d object detection. In *CVPR*, pages 4604–4612, 2020. 4

[16] Chunwei Wang, Chao Ma, Ming Zhu, and Xiaokang Yang. Pointaugmenting: Cross-modal augmentation for 3d object detection. In *CVPR*, pages 11794–11803, 2021. 4

[17] Haiyang Wang, Hao Tang, Shaoshuai Shi, Aoxue Li, Zhenguo Li, Bernt Schiele, and Liwei Wang. Unitr: A unified and efficient multi-modal transformer for bird's-eye-view representation. In *ICCV*, pages 6792–6802, 2023. 4

[18] Junjie Yan, Yingfei Liu, Jianjian Sun, Fan Jia, Shuailin Li, Tiancai Wang, and Xiangyu Zhang. Cross modal transformer via coordinates encoding for 3d object dectection. *arXiv preprint arXiv:2301.01283*, 2(3):4, 2023. 4

[19] Zeyu Yang, Jiaqi Chen, Zhenwei Miao, Wei Li, Xiatian Zhu, and Li Zhang. Deepinteraction: 3d object detection via modality interaction. *NeurIPS*, 35:1992–2005, 2022. 4

[20] Junbo Yin, Jianbing Shen, Runnan Chen, Wei Li, Ruigang Yang, Pascal Frossard, and Wenguan Wang. Is-fusion: Instance-scene collaborative fusion for multimodal 3d object detection. In *CVPR*, pages 14905–14915, 2024. 4

[21] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Multimodal virtual point 3d detection. *NeurIPS*, 34:16494–16507, 2021. 4

[22] Zhizhuo Zhou and Shubham Tulsiani. Sparsefusion: Distilling view-conditioned diffusion for 3d reconstruction. In *CVPR*, pages 12588–12597, 2023. 4