InsFusion: Rethink Instance-level LiDAR-Camera Fusion for 3D Object Detection

Zhongyu Xia[1] **Hansong Yang**[2] **Yongtao Wang**[1✉]
[1]Wangxuan Institute of Computer Technology, Peking University
[2]Beijing Jiaotong University
{xiazhongyu,wyt}@pku.edu.cn yanghansong@bjtu.edu.cn

Abstract

Three-dimensional Object Detection from multi-view cameras and LiDAR is a crucial component for autonomous driving and smart transportation. However, in the process of basic feature extraction, perspective transformation, and feature fusion, noise and error will gradually accumulate. To address this issue, we propose InsFusion, which can extract proposals from both raw and fused features and utilizes these proposals to query the raw features, thereby mitigating the impact of accumulated errors. Additionally, by incorporating attention mechanisms applied to the raw features, it thereby mitigates the impact of accumulated errors. Experiments on the nuScenes dataset demonstrate that InsFusion is compatible with various advanced baseline methods and delivers new state-of-the-art performance for 3D object detection.

1 Introduction

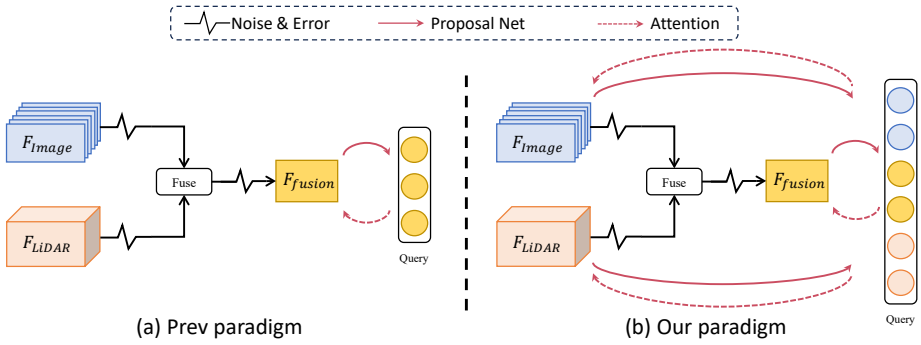

Figure 1: Comparison between the existing paradigm and our paradigm.

Multi-sensor fusion is essential for an accurate and reliable perception system, where the workflow typically entails extracting raw features from images and point clouds, transforming the 2D features into 3D representations, converting the 3D features from each modality into a consistent coordinate frame, fusing them, and extracting instance features from multimodal features. At each step of this process, various forms of noise or errors can be introduced. For instance, these may include depth estimation errors during 2D-to-3D transformation or offset prediction inaccuracies in deformable attention mechanisms, inaccuracies in extrinsic parameters during coordinate transformation, as well as information loss during the fusion process. These errors can accumulate and spread, ultimately affecting overall performance.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: UrbanAI-2025: Harnessing Artificial Intelligence for Smart Cities.

To address this issue, we introduce InsFusion, a universal instance-level fusion paradigm specifically tailored for Bird's-Eye View (BEV) LiDAR-camera fusion models. InsFusion extracts proposals from both raw and fused features and utilizes these proposals to query the raw features, thereby alleviating the impact of accumulated errors. InsFusion can be easily integrated into all existing BEV-based LiDAR-camera fusion models, enhancing their performance.

Our contributions are: (1) We propose InsFusion, a novel paradigm compatible with existing multimodal 3D object detection methods, to address the issue of accumulated errors. (2) InsFusion requires only minimal fine-tuning of the baseline models, resulting in low training cost. (3) InsFusion enhances the performance of both advanced models [1, 2] and achieves state-of-the-art results on the nuScenes dataset [3].

2 Related Work

LiDAR-only. Prior LiDAR-based works directly operate on the raw LiDAR point clouds [4, 5, 6, 7, 8, 9] to extract features. With the growing volume of LiDAR points in outdoor datasets, several works transform point clouds into Euclidean feature space, such as 3D voxels [10], range images [11, 12], and bird's eye view (BEV) plane [13, 14, 15, 11, 12].

Camera-only. Early works [16, 17, 18, 19, 20, 21, 22, 23, 24] predicted 3D bounding boxes from monocular images. With the introduction of datasets such as nuScenes [3], methods for multi-view cameras have demonstrated superior performance. The core problem they address involves transforming 2D feature maps into 3D features while extracting instance features. LSS [25] and related methods [26, 27, 28, 29, 30, 31] employ depth estimation to lift 2D features into the BEV space, subsequently extracting instance features through techniques such as heatmap peak detection [15]. BEVFormer series [32, 33] acquires BEV features through Deformable Attention [34]. Subsequent approaches [35, 36, 37, 38, 39, 40, 41] directly obtain instance features by querying 2D feature maps. They streamline the feature extraction pipeline, thereby mitigating cumulative errors and consequently achieving superior performance in object detection tasks.

LiDAR-camera fusion. In contrast to early-fusion [42, 43, 44, 45] or late-fusion [46] strategies, feature-level fusion methods have demonstrated superior performance. Existing feature-level fusion methods [47, 48, 49, 50] are primarily extensions of LiDAR-only approaches, with BEV-based methods constituting the predominant paradigm. Such paradigms typically cascade multiple modules, thereby introducing cumulative errors. Subsequent methods often aim to mitigate inaccuracies in specific components of this pipeline, such as reducing depth estimation errors [51, 52] to alleviate noise in the transformation from 2D feature maps to BEV features. IS-Fusion [2] employs instance-guided Fusion to mitigate noise introduction during the BEV fusion process. FocalFormer3D [1] uses multistage heatmaps to reduce errors in the instance feature extraction process from multimodal BEV features. In contrast to previous efforts that primarily focus on refining individual components within the pipeline, InsFusion leverages minimally processed raw features to mitigate noise while maintaining full compatibility with existing methods, thereby enhancing their performance.

3 Methodology

As shown in Fig. 2, in order to extract information from features with minor cumulative noise, InsFusion takes three steps: extracting proposals from 2D feature maps, LiDAR features, and fused features; performing query alignment; and refining instance features through attention mechanisms.

3.1 Extract Queries From Raw Features

Camera Branch. For image features $F_{img} \in \mathbb{R}^{h \times w \times C_{img}}$, we first define K learnable camera queries, which are drawn from random Gaussian distributions and attached to a D_q-dim query feature to encode the rich instance characteristics. Then we adopt adaptive sampling and mixing [41] to obtain instance features $Q^{(0)}_{img} \in \mathbb{R}^{K \times D_q}$.

Lidar Branch. For LiDAR BEV features $F_{lidar_bev} \in R^{X \times Y \times C_{lidar_bev}}$, InsFusion predicts the instance heatmap and extracts proposals through peak detection. We select the highest K peaks from

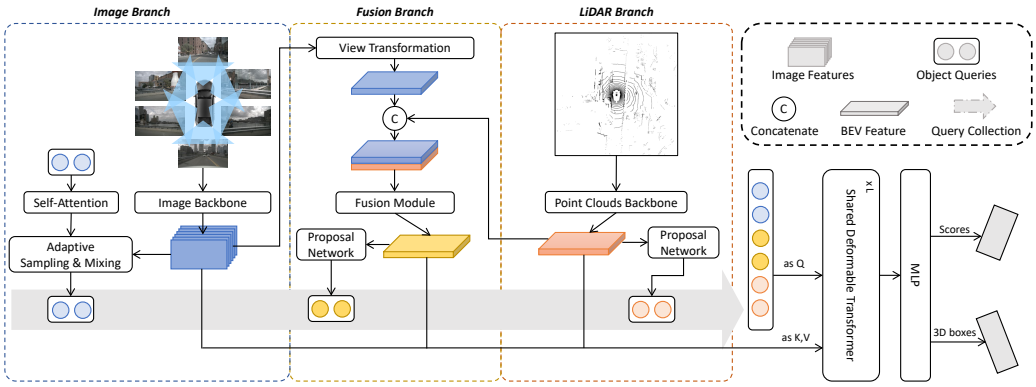

Figure 2: **Overview of the InsFusion framework.** The framework extracts proposals from raw camera features, LiDAR features, as well as fused BEV features, and then aligns and refines all proposals to predict 3D bounding boxes.

H_{lidar} and encode their spatial coordinates into initial embeddings, forming the initial LiDAR query $Q^{(0)}_{\text{lidar}} \in \mathbb{R}^{K \times D_q}$.

Fusion branch. InsFusion is a framework compatible with existing multimodal fusion methods, applicable to any feature fusion strategy, whether it employs BEV heatmaps or Transformer-based architectures to obtain the fused instance features $Q^{(0)}_{\text{fusion}}$.

Queries Alignment. In this stage, the three sets of queries ($Q_{\text{img}} \in \mathbb{R}^{K \times D_q}$, $Q_{\text{lidar}} \in \mathbb{R}^{K \times D_q}$, $Q_{\text{fusion}} \in \mathbb{R}^{K \times D_q}$) are projected into a shared latent space via modality-specific linear transformers, eliminating distribution shifts while preserving modality-specific strengths: $\hat{Q}_{\text{img}} = W_{\text{img}} \cdot Q_{\text{img}} + b_{\text{img}}$, $\hat{Q}_{\text{lidar}} = W_{\text{lidar}} \cdot Q_{\text{lidar}} + b_{\text{lidar}}$, $\hat{Q}_{\text{fusion}} = W_{\text{fusion}} \cdot Q_{\text{fusion}} + b_{\text{fusion}}$, where $W_{\text{img}}, W_{\text{lidar}}, W_{\text{fusion}} \in \mathbb{R}^{D_q \times D_q}$ are learnable projection matrices, and $b_{\text{img}}, b_{\text{lidar}}, b_{\text{fusion}} \in \mathbb{R}^{1 \times D_q}$ are bias terms. These parameters are optimized end-to-end with the entire network.

3.2 Instance Feature Refinement

After alignment, we concatenate the three aligned queries sets ($\hat{Q}_{\text{img}}, \hat{Q}_{\text{lidar}}, \hat{Q}_{\text{fusion}}$) to form a comprehensive query $Q^{(0)}$. This design integrates low-error information from raw-modality proposals and cross-modal context from fusion proposals, enabling multi-source feature interaction during refinement. Instead of using a single feature map as Key (K) and Value (V), we leverage the three core feature sources (raw camera features F_{img}, raw LiDAR BEV features F_{lidar_bev}, and fused BEV features F_{fusion_bev}) as separate Key-Value. The concatenated query Q_{concat} performs targeted queries on each of these three feature sources via a deformable transformer decoder—this ensures the query pool fully captures modality-specific strengths (e.g., image semantics, LiDAR geometry, cross-modal complementarity) while avoiding error accumulation from over-reliance on a single feature source. The refinement process follows:

$$
\begin{aligned}
Q^{(0)} &= \left[\hat{Q}\text{img}, \hat{Q}\text{lidar}, \hat{Q}\text{fusion}\right] \in \mathbb{R}^{3K \times D_q}, \\
Q^{(l)} &= \text{DeformableTransformerLayer}(Q^{(l-1)}, \text{Flatten}(F_{\text{img, raw}}), \text{Flatten}(F_{\text{lidar, raw}}), \text{Flatten}(F_{\text{fusion}})).
\end{aligned} \tag{1}
$$

This process is repeated for L iterations. Here, $[\cdot, \cdot, \cdot]$ denotes concatenation, and the deformable decoder layer independently computes attention between $Q^{(l-1)}_{\text{concat}}$ and each of the three flattened Key-Value pairs (i.e., F_{img}, F_{lidar_bev}, F_{fusion_bev}). The attention outputs from the three feature sources are aggregated via element-wise addition to update the query. This multi-source query design ensures InsFusion fully leverages the precision of raw features and the complementarity of fused features, directly addressing error accumulation by anchoring refinement to low-error, modality-specific sources.

Table 1: **Results on the nuScenes `val` set.** Applying InsFusion to FocalFormer and the state-of-the-art method IS-Fusion effectively enhances their performance.

Method	Mod.	I.B.	mAP	NDS	Car	Truck	C.V.	Bus	Trailer	Barrier	Motor.	Bike	Ped.	T.C.
BEVFormer [32]	C	V2-99	41.6	51.7	61.8	37.0	12.8	44.4	17.2	52.5	42.9	39.8	49.4	58.4
BEVDet4D [26]	C	R50	40.8	52.3	62.7	33.4	12.4	39.3	15.6	58.7	38.5	39.1	46.7	61.8
SparseBEV [41]	C	V2-99	57.3	65.0	72.5	53.8	25.9	61.3	33.3	67.7	58.9	60.8	65.2	73.9
CenterPoint [15]	L	-	58.6	65.8	84.0	53.8	19.9	68.1	38.7	67.8	62.2	43.4	83.1	65.3
Transfusion-L [48]	L	-	65.1	70.1	86.6	41.1	26.1	70.6	38.1	68.5	69.4	54.7	85.5	71.9
FocalFormer3D-L [1]	L	-	66.3	70.1	87.4	59.1	27.8	76.4	44.3	71.1	75.3	60.4	87.2	74.0
FUTR3D [53]	L+C	R101	64.2	68.0	86.3	61.5	26.0	71.9	42.1	64.4	73.6	63.3	82.6	70.1
BEVFusion [49]	L+C	Swin-T	69.6	72.1	89.1	66.7	30.9	77.7	42.6	73.5	79.0	67.5	89.4	79.3
FocalFormer3D [1]	L+C	R50	70.5	73.1	**89.4**	65.4	30.5	78.8	45.6	73.2	**83.0**	**70.9**	88.7	79.5
+InsFusion (ours)	L+C	R50	71.5(+1.0)	74.2(+1.1)	88.9	**69.7**	30.5	77.7	**53.1**	**84.2**	78.4	65.0	89.0	78.9
IS-Fusion [2]‡	L+C	Swin-T	72.3	73.7	**89.4**	68.3	36.1	79.4	51.6	74.7	80.6	70.1	89.3	83.6
+InsFusion (ours)	L+C	Swin-T	**73.4**(+1.1)	**74.3**(+0.6)	**89.4**	68.1	**37.0**	**79.5**	52.9	80.1	80.3	70.4	**91.3**	**85.4**

'Mod.': Modality, 'I.B.': Image Backbone. ‡ : Reproduced with official code. Notion of modality: Camera (C), LiDAR (L).
Notion of class: Construction vehicle (C.V.), pedestrian (Ped.), traffic cone (T.C.).

Table 2: Ablation study on the number of deformable transformer layers. The metrics mAP (%) and NDS (%) are computed on the nuScenes `val` set.

Layers Number (L)	**mAP (%)**	**NDS (%)**
1	67.31	71.35
2	**69.33**	**72.35**
6	68.82	71.59

4 Experiments

We provide experimental setup and implementation details in the Appendix A.

Main Result. Experimental results on the nuScenes [3] dataset are summarized in Tab. 1. When integrated with FocalFormer3D, InsFusion yields performance gains of +1.0% mAP and +1.1% NDS; when applied to the state-of-the-art method IS-Fusion, InsFusion achieves improvements of +1.1% mAP and +0.6% NDS. These results show the effectiveness of InsFusion. Unfortunately, due to a service outage in the official nuScenes evaluation server, we were unable to evaluate the results on the test set.

Ablation study for the number of Transformer layers. As shown in Tab. 2, when $L = 1$, the single-layer deformable Transformer exhibits insufficient capability to refine concatenated proposals, which prevents the full capture of complementary information across multi-source features (raw camera, raw LiDAR, and fused BEV); and when $L = 6$, excessive iterative refinement induces the model to overfit to local noise generated during the fusion process rather than focusing on critical global instance-level information, thereby leading to performance degradation, $L = 2$ is ultimately identified as the optimal configuration, as it achieves a balanced trade-off between ensuring adequate feature refinement and avoiding overfitting to local noise.

5 Conclusion

3D object detection using multi-view cameras and LiDAR is crucial for autonomous driving and smart transportation. However, it suffers from noise and error accumulation during feature processing and fusion, which degrades performance. To solve this, we proposed InsFusion, a universal instance-level fusion paradigm for LiDAR-camera fusion. It extracts queries from both raw and fused features, and then queries these features to alleviate accumulated errors. InsFusion is compatible with existing BEV-based fusion models and only requires minimal fine-tuning (with low training cost). Experiments on the nuScenes dataset demonstrate that InsFusion can further enhance the performance of several high-performing baseline methods.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 62176007).

References

[1] Yilun Chen, Zhiding Yu, Yukang Chen, Shiyi Lan, Anima Anandkumar, Jiaya Jia, and Jose M Alvarez. Focalformer3d: focusing on hard instance for 3d object detection. In *ICCV*, 2023.

[2] 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*, 2024.

[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. In *CVPR*, 2020.

[4] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, 2017.

[5] C. Qi, Hao Su, Kaichun Mo, and L. Guibas. PointNet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017.

[6] C. Qi, W. Liu, Chenxia Wu, Hao Su, and L. Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *CVPR*, 2018.

[7] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. PointRCNN: 3d object proposal generation and detection from point cloud. In *CVPR*, 2019.

[8] Zetong Yang, Y. Sun, Shu Liu, and Jiaya Jia. 3DSSD: Point-based 3d single stage object detector. In *CVPR*, 2020.

[9] Zhichao Li, Feng Wang, and Naiyan Wang. Lidar r-cnn: An efficient and universal 3d object detector. In *CVPR*, 2021.

[10] Yin Zhou and Oncel Tuzel. VoxelNet: End-to-end learning for point cloud based 3d object detection. In *CVPR*, 2018.

[11] Lue Fan, Xuan Xiong, Feng Wang, Naiyan Wang, and Zhaoxiang Zhang. Rangedet: In defense of range view for lidar-based 3d object detection. In *ICCV*, 2021.

[12] Pei Sun, Weiyue Wang, Yuning Chai, Gamaleldin F. Elsayed, Alex Bewley, Xiao Zhang, Cristian Sminchisescu, and Drago Anguelov. RSN: Range sparse net for efficient, accurate lidar 3d object detection. In *CVPR*, 2021.

[13] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. PointPillars: Fast encoders for object detection from point clouds. In *CVPR*, 2019.

[14] Yue Wang, Alireza Fathi, Abhijit Kundu, David A. Ross, Caroline Pantofaru, Thomas A. Funkhouser, and Justin M. Solomon. Pillar-based object detection for autonomous driving. In *ECCV*, 2020.

[15] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Center-based 3d object detection and tracking. In *CVPR*, 2021.

[16] Yan Lu, Xinzhu Ma, Lei Yang, Tianzhu Zhang, Yating Liu, Qi Chu, Junjie Yan, and Wanli Ouyang. Geometry uncertainty projection network for monocular 3d object detection. In *ICCV*, 2021.

[17] Thomas Roddick, Alex Kendall, and R. Cipolla. Orthographic feature transform for monocular 3d object detection. In *BMVC*, 2019.

[18] Zongdai Liu, Dingfu Zhou, Feixiang Lu, Jin Fang, and Liangjun Zhang. Autoshape: Real-time shape-aware monocular 3d object detection. In *ICCV*, 2021.

[19] Abhinav Kumar, Garrick Brazil, and Xiaoming Liu. Groomed-nms: Grouped mathematically differentiable nms for monocular 3d object detection. In *CVPR*, 2021.

[20] Yunpeng Zhang, Jiwen Lu, and Jie Zhou. Objects are different: Flexible monocular 3d object detection. In *CVPR*, 2021.

[21] Yunsong Zhou, Yuan He, Hongzi Zhu, Cheng Wang, Hongyang Li, and Qinhong Jiang. Monocular 3d object detection: An extrinsic parameter free approach. In *CVPR*, 2021.

[22] Cody Reading, Ali Harakeh, Julia Chae, and Steven L Waslander. Categorical depth distribution network for monocular 3d object detection. In *CVPR*, 2021.

[23] Li Wang, Li Zhang, Yi Zhu, Zhi Zhang, Tong He, Mu Li, and Xiangyang Xue. Progressive coordinate transforms for monocular 3d object detection. In *NeurIPS*, 2021.

[24] Li Wang, Liang Du, Xiaoqing Ye, Yanwei Fu, Guodong Guo, Xiangyang Xue, Jianfeng Feng, and Li Zhang. Depth-conditioned dynamic message propagation for monocular 3d object detection. In *CVPR*, 2021.

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

[26] Junjie Huang, Guan Huang, Zheng Zhu, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021.

[27] 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 *AAAI*, 2023.

[28] Zengran Wang, Chen Min, Zheng Ge, Yinhao Li, Zeming Li, Hongyu Yang, and Di Huang. Sts: Surround-view temporal stereo for multi-view 3d detection. *arXiv preprint arXiv:2208.10145*, 2022.

[29] Zhuofan Zong, Dongzhi Jiang, Guanglu Song, Zeyue Xue, Jingyong Su, Hongsheng Li, and Yu Liu. Temporal enhanced training of multi-view 3d object detector via historical object prediction. In *ICCV*, 2023.

[30] Jinhyung Park, Chenfeng Xu, Shijia Yang, Kurt Keutzer, Kris Kitani, Masayoshi Tomizuka, and Wei Zhan. Time will tell: New outlooks and a baseline for temporal multi-view 3d object detection. In *ICLR*, 2023.

[31] Zhongyu Xia, Zhiwei Lin, Xinhao Wang, Yongtao Wang, Yun Xing, Shengxiang Qi, Nan Dong, and Ming-Hsuan Yang. Henet: Hybrid encoding for end-to-end multi-task 3d perception from multi-view cameras. In *European Conference on Computer Vision*. Springer, 2024.

[32] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *ECCV*, 2022.

[33] Chenyu Yang, Yuntao Chen, Hao Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Yu Qiao, Lewei Lu, et al. Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. In *CVPR*, 2023.

[34] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2021.

[35] Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *CoRL*. PMLR, 2022.

[36] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. In *ECCV*. Springer, 2022.

[37] Yingfei Liu, Junjie Yan, Fan Jia, Shuailin Li, Aqi Gao, Tiancai Wang, and Xiangyu Zhang. Petrv2: A unified framework for 3d perception from multi-camera images. In *ICCV*, 2023.

[38] Xuewu Lin, Tianwei Lin, Zixiang Pei, Lichao Huang, and Zhizhong Su. Sparse4d: Multi-view 3d object detection with sparse spatial-temporal fusion. *arXiv preprint arXiv:2211.10581*, 2022.

[39] Xuewu Lin, Tianwei Lin, Zixiang Pei, Lichao Huang, and Zhizhong Su. Sparse4d v2: Recurrent temporal fusion with sparse model. *arXiv preprint arXiv:2305.14018*, 2023.

[40] Shihao Wang, Yingfei Liu, Tiancai Wang, Ying Li, and Xiangyu Zhang. Exploring object-centric temporal modeling for efficient multi-view 3d object detection. In *ICCV*, 2023.

[41] Haisong Liu, Yao Teng, Tao Lu, Haiguang Wang, and Limin Wang. Sparsebev: High-performance sparse 3d object detection from multi-camera videos. In *ICCV*, 2023.

[42] Tengteng Huang, Zhe Liu, Xiwu Chen, and X. Bai. EPNet: Enhancing point features with image semantics for 3d object detection. In *ECCV*, 2020.

[43] Vishwanath A. Sindagi, Yin Zhou, and Oncel Tuzel. MVX-Net: Multimodal voxelnet for 3d object detection. In *ICRA*, 2019.

[44] Chunwei Wang, Chao Ma, Ming Zhu, and Xiaokang Yang. PointAugmenting: Cross-modal augmentation for 3d object detection. In *CVPR*, 2021.

[45] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Multimodal virtual point 3d detection. In *NeurIPS*, 2021.

[46] Su Pang, Daniel Morris, and Hayder Radha. Clocs: Camera-lidar object candidates fusion for 3d object detection. In *IROS*. IEEE, 2020.

[47] Jin Hyeok Yoo, Yeocheol Kim, Ji Song Kim, and J. Choi. 3D-CVF: Generating joint camera and lidar features using cross-view spatial feature fusion for 3d object detection. In *ECCV*, 2020.

[48] 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*, 2022.

[49] 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. In *NeurIPS*, volume 35, 2022.

[50] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *ICRA*, 2023.

[51] Haotian Hu, Fanyi Wang, Jingwen Su, Yaonong Wang, Laifeng Hu, Weiye Fang, Jingwei Xu, and Zhiwang Zhang. Ea-lss: Edge-aware lift-splat-shot framework for 3d bev object detection. *arXiv preprint arXiv:2303.17895*, 2023.

[52] Yun Zhao, Zhan Gong, Peiru Zheng, Hong Zhu, and Shaohua Wu. Simplebev: Improved lidar-camera fusion architecture for 3d object detection. *arXiv preprint arXiv:2411.05292*, 2024.

[53] Xuanyao Chen, Tianyuan Zhang, Yue Wang, Yilun Wang, and Hang Zhao. Futr3d: A unified sensor fusion framework for 3d detection. *arXiv preprint arXiv:2203.10642*, 2022.

[54] MMDetection3D Contributors. MMDetection3D: OpenMMLab next-generation platform for general 3D object detection. `https://github.com/open-mmlab/mmdetection3d`, 2020.

A Experimental Setup.

Dataset. Our experimental evaluation is performed on the nuScenes dataset, a standard benchmark for autonomous driving research that provides comprehensive multi-sensor data across diverse urban environments. This dataset enables rigorous evaluation of 3D detection systems under challenging real-world conditions.We adopt the official evaluation metrics: NuScenes Detection Score (NDS) and mean Average Precision (mAP), which comprehensively assess detection accuracy, localization precision, and orientation prediction.

Implementation Details. Our main implementation relies on the open-source MMdetection3D [54] framework. When deploying InsFusion on the FocalFormer and IS-Fusion baselines, we maintain all the original settings of the baseline model, including the LiDAR backbone, camera backbone, and the original fusion network, to ensure a fair comparison.

For the key parameters of InsFusion: the number of layers for the Shared Deformable Transformer is set to $L = 2$; the initial query count for both the LiDAR branch and camera branch is 300. We train our network using a two-stage training mode: Firstly, train the baseline model (skip this step if official weights are available and use the official weights directly), and then train the "extract proposals from img" network. When training the InsFusion-enhanced model, load and freeze the baseline weights, and load the weights of the "extract proposals from img" network.

For optimization, we use the Adam optimizer with a one-cycle learning rate policy (max learning rate 2×10^{-5}) and a weight decay of 0.01. All experiments are conducted on 8 NVIDIA RTX 8000 GPUs, with a batch size of 16 and a training duration of 6 epochs.

B Limitation.

A limitation of InsFusion is that it introduces additional computational overhead compared to baseline models, which manifests as a slight reduction in inference efficiency. However, as shown in Tab. 3, this efficiency cost is insignificant and well within acceptable limits for practical use: when integrated into the FocalFormer baseline, the inference FPS decreases by only 9.3%; when applied to the IS-Fusion baseline, the FPS reduction is even smaller, at just 6.6%. Based on the inference FPS metric alone, the resulting decrease in inference latency remains within an acceptable range, indicating that its computational overhead does not impede practical deployment in 3D detection applications.

Table 3: Computational Overhead Comparison Between InsFusion-Enhanced Models and Baselines

Model Name	**Inference FPS↑**
FocalFormer	1.29
+InsFusion	1.17
IS-Fusion	0.91
+InsFusion	0.85

C Broader Impacts Statement.

All datasets we used are published datasets. We do not see potential privacy-related issues. This study may inspire future research on LiDAR-camera fusion model for 3D perception.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the paper's core contributions.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of this work in Appendix. B.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: This paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide the implementation details in Appendix. A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Code will be released after the paper is accepted.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide the implementation details in Appendix. A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The paper does not report error bars, confidence intervals, or statistical significance information for experiments supporting main claims, due to the high complexity and cost of constructing error bars.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide the information for compute resources in Appendix. A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research in the paper conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We provide the discussion of broader impacts in Appendix. C.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The benchmark and models utilized in this study do not pose such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All models, code, and data utilized in this study are properly cited to acknowledge their original owners. We have strictly complied with all applicable licenses governing the use of these models, code, and data.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not release any new assets.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing activities or research conducted on human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core methods of this paper are unrelated to LLMs and do not involve LLMs as any component.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.