SupFusion: Supervised LiDAR-Camera Fusion for 3D Object Detection

1. Implementation Details

Training Settings. We implement the MVXNet [4], PV-RCNN-LC [3], Voxel-RCNN-LC [1], and SECOND-LC [5] based on pytorch on 8 Tesla A100 GPUs. Here, the PV-RCNN and Voxel-RCNN are the lidar-based detections, we follow VFF [2], which is a state-of-the-art LiDAR-Camera fusion method based on PV-RCNN and Voxel-RCNN on KITTI benchmark to obtain the corresponding multi-model detectors. We also re-conduct SECOND-LC based on BEVFusion fusion strategy and SECOND lidar detector on nuScenes benchmark. The detector and optimizer settings including voxel size, epoch, and learning rate (LR) follow the open-sourced code and are shown in Tab. 1.

Table 1: Detectors and optimizers settings

Detector	Voxel Size	Epoch	LR
MVXNet	[0.05, 0.05, 0.1]	40	0.003
PV-RCNN-LC	[0.05, 0.05, 0.1]	80	0.05
Voxel-RCNN-LC	[0.05, 0.05, 0.1]	80	0.05
SECOND-LC	[0.075, 0.075, 0.2]	20	0.0001

SupFusion. We employ the SupFusion training strategy and the deep fusion module. In Tab. 3 in the paper, we conduct experiments based on three fusion strategies, including sum, concat, and ours to indicate summation, concatenation, and our deep fusion. For the model with summation fusion, we directly add the lidar and camera features as the fusion feature and feed it into the detection head for final prediction. To employ our proposed SupFusion, we mimic the fusion feature to the high-quality feature and then add it to the original fusion features for the final prediction, which is a simple residual module. The concatenation applys the same strategy (how to train the baseline model and apply SupFusion) as summation except for the fusion method. For our deep fusion, as shown in Fig. 2 in the paper, we introduce a 2D learner and a 2D3D learner to obtain the fusion features, we prune the $\mathcal{L}_{sim}$ loss to conduct experiments without SupFusion in Tab. 3.

2. Feature Visualization

We compare the feature map response on the nuScenes dataset in Fig. 1. We could observe that the background regions of the image branch in yellow cycle are alleviated, and the object regions in cyan cycle are highlighted by the model with SupFusion.

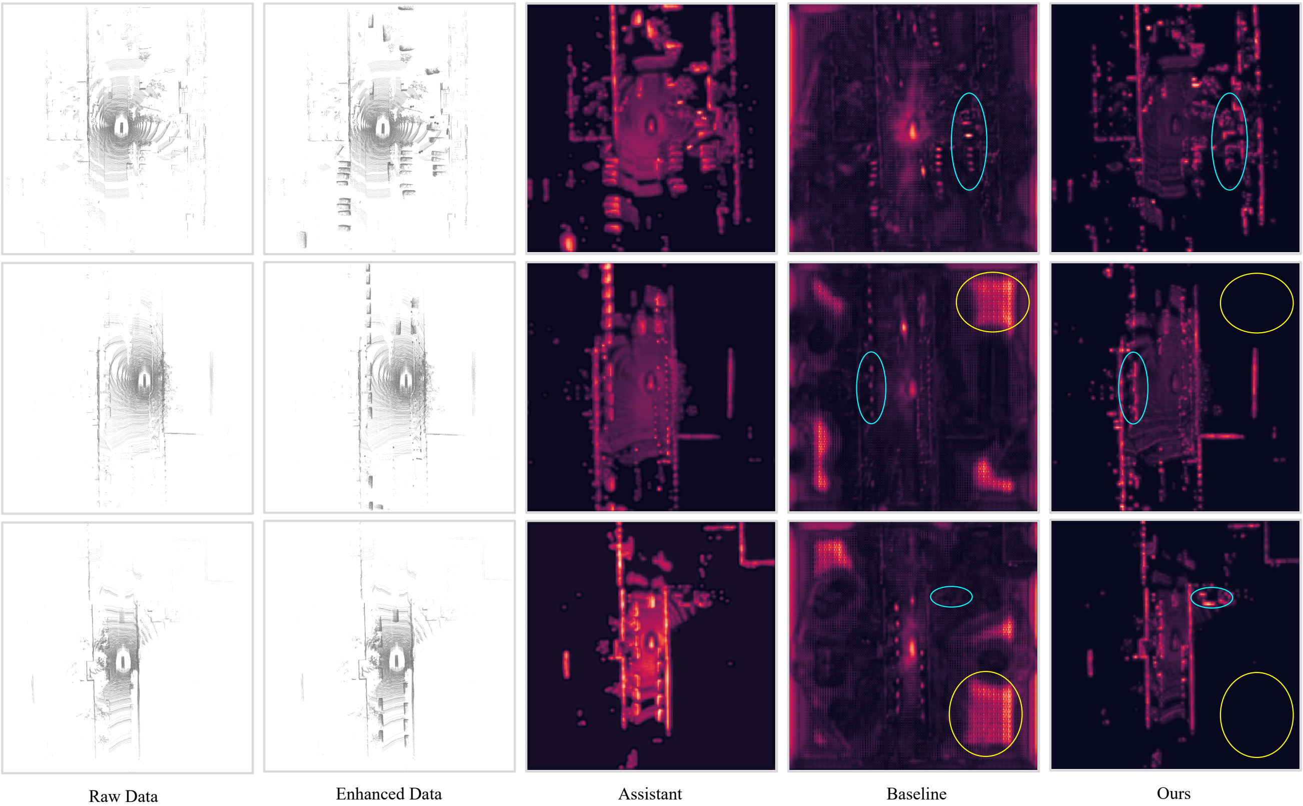

Figure 1: Visualization results. The first column shows the raw sparse lidar data and the second column displays the enhanced data. We show the feature maps of the assistant model, baseline model, and our SupFusion model following, respectively.

References

[1] Jiajun Deng, Shaoshuai Shi, Peiwei Li, Wengang Zhou, Yanyong Zhang, and Houqiang Li. Voxel r-cnn: Towards high performance voxel-based 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1201–1209, 2021.

[2] Yanwei Li, Xiaojuan Qi, Yukang Chen, Liwei Wang, Zeming Li, Jian Sun, and Jiaya Jia. Voxel field fusion for 3d object detection. In *CVPR*, pages 1120–1129, 2022.

[3] 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. In *CVPR*, pages 10529–10538, 2020.

[4] Vishwanath A Sindagi, Yin Zhou, and Oncel Tuzel. Mvx-net: Multimodal voxelnet for 3d object detection. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 7276–7282. IEEE, 2019.

[5] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018.