SFMNet: Sparse Focal Modulation for 3D Object Detection

Supplementary Material

6. Implementation details

6.1. Network architecture

For Argoverse2 and Waymo Open, we adopted the sparse version of CenterPoint [61] as the detection head, as used in [63]. For nuScenes, we employed TransFusion [1] as the detection head, in line with [51, 64].

Argoverse2. We begin by validating the effectiveness of our detector in the long-range detection of Argoverse2. The voxel sizes are set to $(0.1, 0.1, 0.2)$ meters. In stages 1 through 3, we employ $(0, 1, 1)$ SFM blocks, each followed by $(2, 2, 4)$ SRBs, respectively. In stage 4 and within the 2D backbone, we use $(4, 2)$ SFM blocks, each followed by $(2, 4)$ SRBs. For each SFM block, we set $L = 4$ with kernel sizes of $(3, 3, 3, 3)$ and dilation rates of $(1, 3, 5, 7)$.

Waymo Open. For the mid-range dataset, we set the voxel sizes to $(0.08, 0.08, 0.15)$ meters. In stages 1 through 3, we employ $(0, 1, 1)$ SFM blocks, each followed by $(2, 2, 4)$ SRBs, respectively. In stage 4 and within the 2D backbone, we use $(2, 2)$ SFM blocks, each followed by $(6, 6)$ SRBs. Since the Waymo scanning range is not as large as that of Argoverse2, we set $L = 4$ with kernel sizes of $(3, 5, 3, 5)$ and dilation rates of $(1, 1, 3, 3)$ for each SFM block. This configuration results in an effective receptive field (ERF) of 2 meters, compared to 3.3 meters in Argoverse2.

nuScenes. For the short-range detection dataset, we set the voxel sizes to $(0.075, 0.075, 0.2)$ meters and used the same configuration of SFM blocks and SRBs as in Waymo Open. Since nuScenes focuses on short-range detection (up to ± 54 meters), it was shown in [63] that a sparse detection head offers no advantage over a dense one, as used in [64]. Consequently, we implemented the 2D backbone with dense convolutions instead of sparse ones.

6.2. Training and inference schemes

We implemented our approach using the OpenPCDet [47] framework for the Argoverse2 and Waymo Open datasets and the MMDetection3D [10] framework for nuScenes, following the setup of state-of-the-art methods [51, 63, 64].

Argoverse2. We follow the same training schemes as [61] to optimize the network using the Adam optimizer with a weight decay of 0.05 and a one-cycle learning rate policy. The maximum learning rate is set to 5×10^{-3}, and training is performed with a batch size of 32 for 24 epochs on 8 NVIDIA L40 GPUs. Following [63], we apply ground-truth copy-paste data augmentation during training, disabling this augmentation in the final epoch as part of a fade strategy. For reporting frames per second (FPS) during inference, we use a single NVIDIA L40 GPU.

Stage2	Stage3	Stage4	2D backbone	mAP/mAPH L1	mAP/mAPH L2
				75.6/73.3	69.0/66.9
			✓	76.4/74.2	70.1/67.9
		✓	✓	**78.0/75.7**	**71.7/69.5**
	✓	✓	✓	77.5/75.2	71.2/69.0
✓	✓	✓	✓	77.9/75.6	71.6/69.4

Table 7. **Comparison of SFM module placement.** We integrate a single SFM module into various stages of the 3D backbone and the sole stage of the 2D backbone in our detector, evaluating on a 20% subset of the Waymo Open dataset. The results indicate that focusing on stage 4 of the 3D backbone and the 2D backbone is generally sufficient for optimal performance.

Waymo Open. As in Argoverse, we follow the same training schemes as [61] to optimize the network using the Adam optimizer with a weight decay of 0.05, a one-cycle learning rate policy, and a maximum learning rate of 5×10^{-3}. Training is performed with a batch size of 32 for 24 epochs on 8 NVIDIA A100 GPUs. Following [51, 63, 64], we apply ground-truth copy-paste data augmentation during training, disabling this augmentation in the final epoch as part of a fade strategy. During inference, and as described in [51, 63, 64], we use class-specific NMS with IoU thresholds of 0.7, 0.6, and 0.55 for vehicles, pedestrians, and cyclists, respectively.

nuScenes. We follow the training scheme adopted in [1]. The network is trained using the AdamW optimizer with a weight decay of 0.01, a one-cycle learning rate policy, a maximum learning rate of 1×10^{-3}, and a batch size of 32 for 20 epochs on 8 NVIDIA A100 GPUs. Similarly to [1, 51] we apply the fade strategy in the final 5 epochs.

7. Additional ablation studies

To save computational resources, we conducted ablations on 20% of the Waymo Open dataset [44], training for 12 epochs on 8 GPUs with a total batch size of 32 and reporting results on the full validation set. We used a tiny SFMNet with a single SFM module in the 3D backbone and another in the 2D backbone, followed by 2 and 4 SRBs, respectively. To study SFM placement, we positioned the module at different backbone stages, and Table 7 shows that placing it in stage 4 of the 3D backbone and in the 2D backbone is generally sufficient.