2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

FourStr: When Multi-sensor Fusion Meets Semi-supervised Learning

Bangquan Xie[1,2], Liang Yang[3], Zongming Yang[2], Ailin Wei[4], Xiaoxiong Weng[1] and Bing Li[2]

***Abstract*— This research proposes a novel semi-supervised learning framework *FourStr* (*Four-Str*eam formed by two two-stream models) that focuses on the improvement of fusion and labeling efficiency for 3D multi-sensor detector. FourStr adopts a multi-sensor single-stage detector named adaptive fusion network (AFNet) as the backbone and trains it through the semi-supervision learning (SSL) strategy Stereo Fusion. Note that multi-sensor AFNet and SSL Stereo Fusion can benefit each other. On the one hand, the Four-stream composed of two AFNets naturally provides rich inputs and large models for SSL Stereo Fusion. While other SSL works have to use massive augmentation to obtain rich inputs, and deepen and widen the network for large models. On the other hand, by the novel three fusion stages and Loss Pruning, Stereo Fusion improves the fusion and labeling efficiency for AFNet. Finally, extensive experiments demonstrate that FourStr performs excellently on outdoor dataset (KITTI and Waymo Open Dataset) and indoor dataset (SUN RGB-D), especially for the small contour objects. And compared to the fully-supervised methods, FourStr achieves similar accuracy with only 2% labeled data on KITTI (or with 50% labeled data on SUN RGB-D).**

I. Introduction

Recently, multi-sensor fusion has made significant progress in 3D object detection. 3D detection can be divided into camera-based, LiDAR-based, and multi-sensor-based detections according to input data type. The camera image contains rich semantic information, while it is sensitive to illumination changes and does not contain depth information. LiDAR data is immune to illumination changes and provides depth information. But it is sparse and lacks visual features. Hence, the multi-sensor-based detector should be a better solution for the complex environment.

However, the conventional two-stage multi-sensor detector usually has a complicated structure, resulting in a slow speed. Recent studies [1], [2] exhibit that the accuracy of single-stage detectors gradually approaches two-stage. Therefore, we propose to build up a multi-sensor single-stage detector.

*This work was done while Bangquan Xie was a visiting joint-training Ph.D. student at CU-ICAR, Clemson University, USA. Bangquan Xie was also supported by the International Training Program for Outstanding Young Scientific Research Talents in 2019 of Guangdong Provincial Department of Education. (Corresponding authors: Xiaoxiong Weng; Bing Li; Liang Yang.)

[1]Bangquan Xie and Prof. Xiaoxiong Weng are with the School of Civil Engineering and Transportation at South China University of Technology, Guangzhou, 510641 China. ctbqx51@mail.scut.edu.cn,ctxxweng@scut.edu.cn

[2]Bangquan Xie, Zongming Yang and Prof. Bing Li are with the Department of Automotive Engineering, Clemson University International Center for Automotive Research (CU-ICAR), Greenville, SC 29607 USA. zongmiy, bli4@clemson.edu

[3]Liang Yang is with the City College of New York, New York, 10031, USA. lyang1@ccny.cuny.edu

[4]Ailin Wei is with the Department of Bioengineering, Clemson University, Clemson, SC 29631 USA. awei@clemson.edu

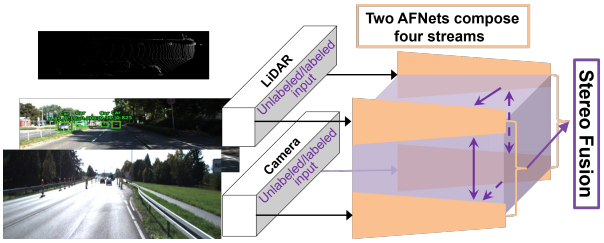

Fig. 1. Overall architecture of FourStr. Under heavy shadow and strong illumination conditions, FourStr performs excellently in detecting small contour objects, even with fewer amount of labeled data.

Besides, the weight of different sensor information needs to be changed according to different environments. For instance, camera should have a dominant impact when the effective LiDAR data is limited. LiDAR data should have a more significant contribution when the detector is affected by adverse illumination conditions such as heavy shadow or strong illumination. Therefore, two adaptive fusion modules of the proposed detector are exquisitely designed to guarantee adaptability under different environments. The proposed multi-sensor single-stage detector is named the Adaptive Fusion Network (AFNet).

Nevertheless, it is challenging for single-stage AFNet to fuse efficiently without the aid of an extra stage. Recently Zheng *et al.*[1], [3] used the semi-supervision learning (SSL) to improve performance of 3D detector. And Hinton and Chen et al. [4] indicated that rich input and large model size can improve the efficiency of SSL. Thus, inspired by [4], we proposes a novel SSL framework *FourStr* (*Four-Str*eam) that consists of two two-stream multi-sensor models. FourStr adopts the AFNet as the backbone, which is trained by an SSL strategy named the Stereo Fusion. In FourStr, multi-sensor fusion AFNet and SSL Stereo Fusion can benefit each other. As shown in Fig. 1, on the one hand, two AFNets construct four-stream to provide rich input representations and large model, which are beneficial to SSL Stereo Fusion [4]. Ingeniously, the four-stream structure can provide these two conditions for Stereo Fusion through its own structure, unlike other SSL works [4], [5] that use strong augmentations for rich input and simply deepen and widen the network for large model. On the other hand, the benefited Stereo Fusion uses the novel three fusion stages to improve the fusion and labeling efficiency for AFNet, integrating the knowledge of the different sensors and aggregating the labeled/unlabeled data information. And the proposed Loss Pruning technology is used to save computational costs and avoid over-fitting.

Finally, by the improvement of fusion and labeling effi-

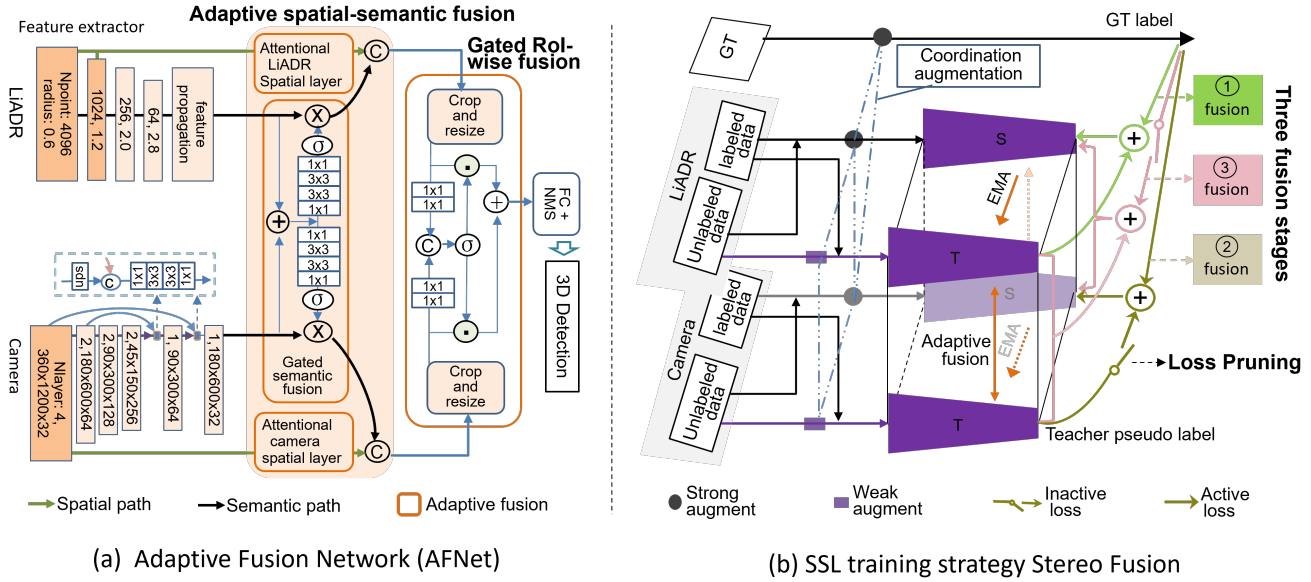

Fig. 2. Structure of FourStr. (a) is the Adaptive Fusion Network (AFNet), including three models: feature extractor, Adaptive spatial-semantic fusion, and Gated RoI-wise fusion. (b) is Stereo Fusion, receiving rich labeled/unlabeled data by the two parallel AFNets (including four streams). It uses three fusion stages to improve the fusion quality and labeling efficiency, and uses the Loss Pruning to save computational cost and avoid over-fitting.

ciency, FourStr finishs excellent results on outdoor dataset (KITTI and Waymo Open Dataset) and indoor dataset (SUN RGB-D). Even under heavy shadow and strong illumination conditions, FourStr performs excellently in detecting small contour objects (pedestrian, cyclist or distant vehicle) which have limited valid detection data, by the improvement of fusion efficiency. And FourStr achieves similar accuracy with fully-supervised methods by little labeled data, demonstrating its high labeling efficiency. The contributions of this paper are summarized as follows:

- A FourStr framework utilizing the mutual gain of multi-sensor fusion and SSL is proposed.
- A multi-sensor single-stage detector AFNet with excellent adaptive fusion modules is proposed.
- A novel SSL training strategy Stereo Fusion including the three fusion stages and Loss Pruning is proposed.

II. Related Works

3D object detection. Various multi-sensor two-stage 3D detectors have been proposed in recent years [6], [7], [8], [9], [10], but they hardly meet the real-time requirements. Recent studies *et al.* [11], [12], [13] revealed that the accuracies of single-sensor single-stage detectors gradually approach that of two-stage detectors. These works presented a single-sensor single-stage device to work as the real-time detector. However, achieving real-time performance with multiple sensors is more challenging due to its inherently complex structure. This article proposes a multi-sensor single-stage detector to tackle this challenge. **Semi-supervised learning.** Currently, there are two effective methods of SSL for 2D detection: Consistency Regularization [14], [15] and Sharpening Prediction [16], [17]. Besides, 3D detection using semi-supervised technology has become a research hotspot in recent years. Zhao *et al.*[3] designed a thorough perturbation scheme to heighten generalization of the network. Wang *et al.*[18] proposed a semi-supervised 3D object detection method leveraging IoU estimation. Zheng *et al.*[1] presented a self-ensembling single-stage object detector.

III. Method

Fig. 2 shows the structure of FourStr for 3D detection. It includes two mutually reinforcing components: AFNet and Stereo Fusion. In Fig. 2 (a), a multi-sensor single-stage 3D detector AFNet was proposed (Section A). It includes three modules: feature extractor, Adaptive spatial-semantic fusion, and Gated RoI-wise fusion (RoI: regions of interest). Note that the last two fusion modules are exquisitely designed as the adaptive modules to guarantee adaptability. In Fig. 2 (b), Stereo Fusion employs two parallel AFNets (including four streams) to receive rich labeled/unlabeled data. Stereo Fusion uses three fusion stages to improve the fusion quality and labeling efficiency, and uses Loss Pruning to save computational cost and avoid over-fitting (Section B).

A. *Adaptive fusion network (AFNet)*

In Fig. 2 (a), we use simplified Pointnet++ [19] as backbone for LiDAR extractor, and the modified ResNet-50 as backbone for camera extractor. See more in Appendix A.

Adaptive spatial-semantic fusion. The Adaptive spatial-semantic fusion includes the Attentional LiDAR/camera spatial layer and Gated semantic fusion.

In Fig. 2 (a), we use the Attentional LiDAR/camera spatial layer to extract the low-level feature in the spatial block (dark orange boxes) of the LiDAR/camera extractor. This spatial feature is useful for small contour objects detection. For the Attentional LiDAR/camera spatial layer, in Fig. 3 (a), the

features of the first two blocks in the LiDAR extractor are connected and learned adaptively by a modified PointNet [20] and two 1×1 convolutions. The modified PointNet is lightweight by dropping the upsampling layers and the refinement stage. Batch-norm and ReLU operations are applied to normalize the scales of the feature. Then a lightweight attention model is used to reweight the feature. In Fig. 3 (b), the Attentional camera spatial layer was used to extract the low-level image feature from the "thick" spatial block (dark orange box) of the camera extractor. "thick" means the layer number (4) of the spatial block is bigger than that of the other blocks (2). Since the dimension (360×1200×32) of the spatial block input is large, the Attentional camera spatial layer is lightweight to avoid high calculation costs.

To extract semantic features from both camera and LiDAR data adaptively, we use a Gated semantic fusion that selectively combines the feature maps according to the relevance to object detection condition, as shown in Fig. 2 (a). The semantic features are gated by applying the attention maps:

$$F_l^g = F_l \otimes \alpha(Conv(F_l \oplus F_c)), F_c^g = F_c \otimes \alpha(Conv(F_c \oplus F_l)), \quad (1)$$

where F_l and F_c represent the LiDAR and camera features, respectively. $\otimes$ is the element-wise operation, and $\oplus$ is the channel-wise concatenation operation. F_l^g and F_c^g are the corresponding gated feature. The $\alpha(Conv(F_l \oplus F_c))$ is the element of the attention maps, indicating the relative importance between the camera and LiDAR features.

As shown in Fig. 2 (a), the Adaptive spatial-semantic fusion model fuses the features of the spatial path (the green arrow line) and semantic path (the black arrow line), and inputs them into the Gated RoI-wise fusion.

Gated RoI-wise fusion. In Fig. 2 (a), Gated RoI-wise fusion receives two views with low-level spatial and high-level semantic information processed by Adaptive spatial-semantic fusion. The crops from both input views are resized to 7×7, then fused with a gated element-wise mean operation. Next, three task-specific FC (Fully Connected) layers and the NMS (non maximum suppression) are deployed to classify and regress the anchors directly for performing classification, regression, and orientation tasks of object detection.

In short, we innovatively designed the Adaptive spatial-semantic fusion and Gated RoI-wise fusion to ensure the adaptability of AFNet. The modified feature extractors are lightweight. Hence, AFNet is an efficient multi-sensor single-stage detector based LiDAR-camera fusion. And the Stereo Fusion is used to further improves its the fusion quality and labeling efficiency below.

B. *Semi-supervised learning by Stereo Fusion*

Coordination augmentation and four-stream for rich inputs. For the consistent learning of SSL, it is essential to implement different intensities of data enhancement. Inspired by [21], we apply strong augmentation for student CNN. Note that teacher CNN is input by the unlabeled data with weak augmentation, and the labeled data without augmentation reducing the disturbation of labeled data. It ensures

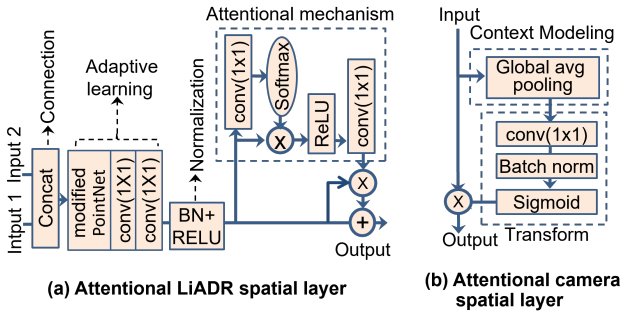

Fig. 3. The structure of Attentional LiDAR/camera spatial layer. (a) is the Attentional LiADR spatial layer. (b) is Attentional camera spatial layer.

that teacher CNN produces high-quality pseudo-labels and secures a reliable performance margin over the student CNN during the training, as shown in Fig. 2 (b).

However, it is essential to coordinate the input data during the distinction data augmentation for SSL, denoted as the coordination augmentation, since we use both LiDAR and camera. Data augmentation with excessive distinction does not guarantee that the modifications applied to the LiDAR points are reflected in the camera images. Hence, the weak augmentation of teacher CNN and strong augmentation of student CNN adopt the same methods. Only the hyper-parameters of these methods are different. For example, strong augmentation uses global scaling with a factor within [0.8,1.2] while weak augmentation uses it with factor within [0.95,1.05]. Similarly, we use the same strong augmentation for both ground truth and student CNN to ensure their coordination.

Since excessive data augmentation will reduce the effect of multi-sensor fusion, we cannot use it to provide rich inputs that is favorable for SSL like other SSL works [4], [5]. Cleverly, four-stream composed of two AFNets provides rich inputs for SSL by its structure. Hence, the data augmentation method of our coordination augmentation only includes the global scaling and global rotation.

Three fusion stages for fusion and labeling efficiency. Except for the rich input, four-stream provides a large multi-sensor model, which is another advantage for SSL strategy. However, training and integration of the large model is challenging. We first train the four-stream by the supervised losses as the pre-training (See more in Appendix B). Then the model is trained in a semi-supervised manner by inputting a mixture of labeled and unlabeled samples during each training batch. For labeled samples, the model is supervised by ground truth like in the pre-training. For unlabeled samples, the model is supervised by the pseudo-labels produced from teacher CNN.

Moreover, Stereo Fusion uses three fusion stages to fully integrate the knowledge of the different sensors and aggregate the labeled/unlabeled data information. As shown in Fig. 2 (b), the two top streams obtain two LiDAR data with varying augmentation intensity, increasing the weight averages of all layer outputs and aggregating the labeled/unlabeled

TABLE I

RESULTS ON KITTI VAL SET WITH VARYING RATIOS OF LABELED DATA, EVALUATED BY MAP WITH 40 RECALL POSITIONS FOR MODERATE LEVEL. THE CASE WITH LABEL RATIO OF 100% MEANS THAT IT IS TRAINED BY FULLY-SUPERVISED METHOD.

Method	1%			2%			100%		
	car	ped.	cyc.	car	ped.	cyc.	car	ped.	cyc.
PVR.[22]	78.6	48.1	62.2	80.9	46.7	63.8	84.83	-	-
3DIoU.[18]	80.7	54.4	67.3	82.0	54.6	69.5	84.8	60.2	74.9
GraR-Po [23]	-	-	-	-	-	-	83.18	-	-
SFD[24]	-	-	-	-	-	-	88.56	-	-
FourStr/AFNet	**84.9**	**61.5**	**74.8**	**88.1**	**62.6**	**77.2**	**89.8/86.9**	**67.9/63.1**	**81.5/76.2**

TABLE II

COMPARISON ON THE WAYMO OPEN DATASET WITH 202 VALIDATION SEQUENCES FOR THREE CATEGORIES. L1: LEVEL_1, L2: LEVEL_2. IMPRO. OF S: IMPROVEMENT OF FOURSTR.

Method	Vehicle(L1)		Vehicle(L2)		Ped.(L1)		Ped.(L2)		Cyc.(L1)		Cyc.(L2)	
	mAP	mAPH	mAP	mAPH	mAP	mAPH	mAP	mAPH	mAP	mAPH	mAP	mAPH
PV-RCNN[22]	77.51	76.89	68.98	68.41	75.01	65.65	66.04	57.61	67.81	66.35	65.39	63.98
M3DETR[25]	77.66	77.09	70.54	69.98	68.20	58.50	60.64	52.03	67.28	65.69	65.31	63.75
Center.[26]	76.7	76.2	68.8	68.3	79.0	72.9	71.0	65.3	-	-	-	-
Graph-Ce [23]	80.77	80.28	72.55	72.10	82.35	76.64	74.44	69.02	75.28	74.21	72.52	71.49
FourStr	**80.94**	**80.96**	**74.88**	**73.98**	**82.84**	**77.24**	**78.82**	**71.23**	**75.89**	**75.27**	**76.82**	**73.93**

information by the exponential moving average (EMA) [15], denoted as ① fusion. Similarly, the two bottom streams get two camera data and aggregate the feature of all layers by EMA, denoted as ② fusion. And the four streams integrate the single sensor data with varying augmentation intensity by EMA and fuse the multi-sensor data by the fusion modules of AFNet simultaneously, denoted as ③ fusion. Generally, the ① fusion and ② fusion integrate the LiDAR and camera knowledge, respectively. And the ③ fusion is a comprehensive integration. In practice, Stereo Fusion is an end-to-end process, which gets the best results in the order of ① fusion, ② fusion, and ③ fusion.

Loss Pruning for saving computational cost and avoiding over-fitting. Training the three fusion stages in a semi-supervised manner requires a lot of computational costs. And FourStr does not use too much data augmentation for the large four-stream model, which leads to easy over-fitting in training. Hence, we use Loss Pruning to save computational costs and avoid over-fitting. In Fig. 2 (b), the six losses (illustrated as the six input arrows on either side of the three ⊕ symbols) can be divided into two categories: inactive loss and active loss. We randomly deactivate some losses during training to change the weight distribution of the six losses, saving computational cost and speeding up convergence. For example, during the ① fusion training of student CNN, if we do not activate the loss generated by the teacher pseudo label, ground truth will play a dominant role. Experiments show Loss Pruning not only saves the computational cost and avoids over-fitting but also improves the accuracy, achieving a similar effect as dropout [27].

Moreover, we "prune" the composition of loss to further save computational cost. Similar to supervised training, we consider semantic categories L_{cla}, object locations L_{center}, and sizes L_{size} to build the semi-supervised loss. The ③ fusion is trained by minimize the $\sigma L_{cla}+\varsigma L_{center}+\tau L_{size}+L^{supe}_{student}$, where σ, ς, and τ are the loss weights. However, we find that L_{cla} result has a slight advantage in the experiment. To save computational cost, ① fusion and ② fusion are trained by minimizing the $L_{cla}+L^{supe}_{student}$ only, pruning the rest two losses. See more about the three semi-supervised losses in Appendix C.

IV. EXPERIMENTS

Our method is evaluated on the outdoor dataset KITTI and Waymo Open Dataset, and on indoor dataset SUN RGB-D. See more about the datasets in Appendix D.

A. *Experiments on outdoor datasets*

Implementation details. For two outdoor datasets, the mean average precision (mAP) and mean average precision weighted by heading (mAPH) are used for 3D and BEV (birds-eye view) tasks evaluation, and the rotated IoU (Intersection over Union) threshold is set as 0.7 for vehicle and 0.5 for pedestrian/cyclist. We pre-train AFNet by minimizing the supervised loss and then initialize the student and teacher CNNs with pre-trained weights. Then the model is trained by Stereo Fusion for minimizing semi-supervised consistency loss. We use 4 V100 GPUs to train the network with the ADAM optimizer. Due to different scales of the two datasets, we use a batch size of 4 with a learning rate of 0.01 for KITTI, and a batch size of 8 with the cosine annealing learning rate strategy for Waymo Open Dataset.

Comparisons on KITTI. On the KITTI validation set, Table I exhibits FourStr achieves notable improvement across all categories on the Moderate level. Compared to semi-supervised model [18], with the 1%, 2%, and 100% labeled data, FourStr increases by average of 5.1 on car and 7.3 on pedestrian and cyclist, showing its excellently performance for detecting small contour objects. Compared to fully-supervised training model [24], FourStr achieves s similar result on car (88.1 & 88.56) by only 2% of labeled data, demonstrating its labeling efficiency. Besides, the AFNet

TABLE III

RESULTS ON SUN RGB-D VAL SET WITH VARYING RATIOS OF LABELED DATA. BASED ON 3 RUNS WITH RANDOM SAMPLING, MAP@0.25 ARE REPORTED AS MEAN ± STANDARD DEVIATION. THE BOLD INDICATES THE TOP TWO.

Model	10%	20%	30%	40%	50%	70%	100%
VoteNet[28]	34.01	42.04	47.08	50.24	52.72	56.71	57.7
SESS[3]	42.87	47.87	53.17	54.73	56.37	58.97	61.1
3DIoU.[18]	**45.50**	49.70	-	-	-	-	61.50
FourStr	**48.30**	**52.42**	**58.72**	**61.98**	**64.26**	**65.86**	**66.68**

TABLE IV

COMPARISON WITH FULLY-SUPERVISED METHODS BY 100% LABELS ON SUN RGB-D VAL SET. PT: PROCESSING TIME.

Method	mAP-@0.25	PT (s)	Size-(MB)
VoteNet[28]	57.7	0.10	11.2
GroupFree[29]	63.0	-	-
FCAF3D[30]	64.2	0.03	-
FourStr	**66.7**	**0.03**	**9.1**

TABLE V

EFFECT OF AFNET CONDUCTED ON THE CAR OF THE KITTI VALID SET. L_O: USING LIDAR ONLY, AS: ATTENTIONAL LIDAR/CAMERA SPATIAL LAYER, GSF: GATED SEMANTIC FUSION.

L_o	C_o	AS	GSF	Mod.	Easy	Hard
✓				78.9	87.8	76.9
✓	✓			81.2	88.0	79.2
✓	✓	✓		84.1	89.8	79.9
✓	✓		✓	82.6	88.3	78.9
✓	✓	✓	✓	86.9	92.2	81.5

TABLE VI

EFFECT OF STEREO FUSION CONDUCTED ON CAR CATEGORY OF THE KITTI VALID SET WITH 1% LABELED DATA. ①F: ① FUSION.

①F	②F	③F	L_{cen}	L_{cla}	L_{size}	Mod.
		✓	✓			81.5
		✓		✓		81.7
		✓			✓	81.3
		✓	✓	✓	✓	82.1
✓		✓	✓	✓	✓	83.9
	✓	✓	✓	✓	✓	83.4
✓	✓	✓	✓	✓	✓	84.9

TABLE VII

EFFECT OF LOSS PRUNING CONDUCTED ON THE CAR OF THE KITTI VALID SET. W. LP: WITH LOSS PRUNING, ACC. : ACCURACY, EP. : EPOCH.

w. LP		w/o. LP	
acc.	ep.	acc.	ep.
86.9	100	84.8	140

(Not trained by Stereo Fusion) completes excellent results 86.9 mAP on car category.

Comparisons on Waymo Open Dataset. In Table II, FourStr finishs the improvements over all categories and exhibits more significant gains in small classes. Since camera data makes up for the lack of valid point clouds in LEVEL_2, the improvements of FourStr in LEVEL_2 are more significant than that in LEVEL_1. In LEVEL_2, FourStr completes a 4.38 mAP improvement for pedestrian than [26], and outperforms [25] by 4.3 for cyclist.

B. *Experiment on indoor dataset SUN RGB-D*

Implementation details. Like on KITTI, we pre-train AFNet and then train the model with Stereo Fusion and AD strategy on SUN RGB-D. Unlike on KITTI, the detection radius of the AFNet variants adopted on SUN RGB-D is two times smaller than that on KITTI, because indoor dataset has a shorter detection distance than the outdoor dataset. The Adam optimizer trains the model with batch size 8 and an initial learning rate of 0.001. The learning rate decreases by 0.1 after 60 epochs and then reduces by another 0.1 after 40 epochs. Training the model to convergence takes 100 epochs on the Nvidia Tesla V100 GPU.

Comparisons on SUN RGB-D. In Table III, FourStr has better results than [3] and [18]. The smaller the proportion of labeled data, the more significant advantage FourStr has over other methods. This reveals that FourStr learns more knowledge from unlabeled data when the number of labeled data is limited. In Table IV, FourStr is compared with the fully-supervised methods by training with 100% labeled data. Compared with [30], FourStr improves 2.5 mAP with the similar processing speed (0.03 s). Besides, the accuracy (64.26 mAP shown in Table III) of FourStr trained by 50% labeled data is similar with the accuracy of [30] trained by 100% labeled data (64.2 mAP shown in Table IV).

C. *Ablation Study*

Effect of AFNet: In Table V, the ablation study is conducted on the car of the KITTI valid set. We use the LiDAR_only (only input the LiDAR data) as the baseline. For the second row, the improvement in the Hard level (2.3) is significant while in the Easy level (0.3) is marginal. Since the objects from the Hard level usually contain only a few point cloud, adding camera could benefit more. For the third and fourth rows, the attentional LiDAR/camera spatial layer (AS) and Gated semantic fusion (GSF) improve the Moderate level by 6.5%, and 4.7%, respectively, indicating they effectively extract spatial and semantic information.

Effect of Stereo Fusion: This ablation study is conducted on car category of the KITTI valid set with 1% labeled data. In Table VI, the results of the first three rows indicate L_{cla} loss achieves the best result. The fourth row demonstrates that better performance can be achieved by combining three losses. The last three rows indicate that the integration of the three fusions gains a better result.

Effect of Loss Pruning: The ablation study is conducted on the car of the KITTI valid set. In Table VII, during the Stereo Fusion stages, Loss Pruning not only saves the computational cost but also improves the accuracy. Loss Pruning improves the accuracy of 2.1 mAP while reducing training time by 29%.

D. *Qualitative analysis*

The results of qualitative analysis are produced by FourStr. In Fig. 4 (a) and (b), FourStr works well in detecting distant cars under strong illumination and heavy shadow conditions, which is a big challenge for the single-sensor detector. Fig. 4 (c) and (d) shows the visualization results of pedestrians and cyclists on the street. Note that the two people, sitting in the middle of Fig. 4 (d), are not detected, which could be caused by the limited number of this similar training samples.

(a) (b) (c) (d)

Fig. 4. Visualizations of results on the KITTI validation set (Please zoom in to see better). The upper part is the 2D detection box with detection IoU (right) and classification score (left); the lower part is the 3D detection box. red: The ground-truth boxes, green: The car detection boxes, blue: the pedestrian detection box, yellow: the cyclist detection box.

V. Conclusion

In this paper, a FourStr framework utilizing the mutual gain of multi-sensor fusion AFNet and SSL strategy Stereo Fusion is proposed. The novel and adaptive AFNet completes excellent results. And Stereo Fusion further to improve the fusion and labeling efficiency of AFNet. Although semi-supervised training on four-stream is expensive, the processing time of the trained two-stream model satisfies the real-time requirements. With simple changes, Stereo Fusion can improve the accuracy of other two-stream 3D detectors.

APPENDIX

Appendix A: Feature extractor

LiDAR extractor. For point cloud, we utilize simplified Pointnet++ [19] as our backbone . Four set-abstraction modules with multiscale grouping are used to subsample points into groups with sizes of 4096, 1024, 256, and 64. Then feature propagation modules are applied to get the point-wise feature vectors for proposal generation. For the target detection task, we have to regress precise target locations and classify each regressed bounding box as a positive/negative sample. In such processes, it is necessary to consider both low-level spatial information and high-level abstract semantic information. In Fig. 2, LiDAR extractor contains the spatial block (dark orange boxes) and the semantic block (light orange boxes), reducing the number of detection points and increasing the detection radius gradually. **Camera extractor.** For camera images, we utilize a modified ResNet-50 as the encoder using the image size (3, H, W) as input. A bottom-up decoder is employed to upsample the feature map back to multi-scale. The output feature maps of the encoder are upsampled with the bi-linear interpolation and concatenated the corresponding feature maps from the encoder. Then the feature map passed through two 1x1 and two 3x3 convolutional layers. Like the LiDAR extractor, our camera extractor contains spatial and semantic blocks.

Appendix B: Supervised losses. We use the classical Focal loss [31] (denoted L_{cla}^{supe}) for the bounding box classification to alleviate the class imbalance problem. Besides, we use the Orientation-aware Distance-IoU loss [1] to focus more attention on the alignment of box centers and orientations between the predicted and ground-truth bounding boxes. Distance-IoU loss is formulated as: $L_{reg}^{supe} = 1 - IoU(b_p, b_g) + \frac{d^2}{l^2} + \rho(1 - |cos(\Delta o)|)$, where b_p and b_g is the predicted and ground-truth bounding boxes, respectively, d denotes the distance between 3D centers of two bounding boxes, l denotes the diagonal length of the minimum cuboid that encloses both bounding boxes; Δo denotes BEV orientation difference between b_p and b_g; and ρ is a hyper-parameter weight. Hence, the overall loss of 3D supervised training is: $L_{student}^{supe} = \lambda L_{cla}^{supe} + \mu L_{reg}^{supe}$, where λ and μ is the loss weights.

Appendix C: Three semi-supervised losses. The $C_s = c_s$ and $C_t = c_t$ denote the class probabilities of the predicted objects from the student and the teacher CNN, respectively. The aligned $C_s^a = c_s^a$ is easily obtained based on minimum center distance. We define the L_{cla} consistency loss as the Kullback-Leibler (KL) divergence between C_s^a and C_t: $L_{cla} = \frac{1}{C_t} \sum D_{KL}(c_s^a || c_t)$.

Let $M_s = m_s$ and $M_t = m_t$ denotes the centers of the predicted 3D bounding boxes from the student to teacher CNN. We further use M_t^a to denote the elements from M_s that are aligned with each element in M_t . Thus, we propose the L_{center} consistency loss: $L_{center} = \frac{\sum_{m_s} \|m_s - m_t^a\|_2 + \sum_{m_t} \|m_t - m_s^a\|_2}{|M_s| + |M_t|}$,to minimize the alignment errors between the teacher and student CNN.

The sizes of the bounding boxes predicted by the student and the teacher networks are denoted as $B_s = b_s$ and $B_t = b_t$, respectively. We use the same minimum center distance to get the aligned $B_s^a = b_s^a$. The L_{size} consistency loss can now be computed as the Mean Square Error (MSE) between B_s^a and B_t: $L_{size} = \frac{1}{|B_t|} \sum (b_s^a - b_t)^2$.

Finally, the total consistency loss is a weighted sum of all the three consistency terms described earlier:

$$L_{consistency} = \lambda L_{class} + \mu L_{center} + \nu L_{size},$$

where λ, μ, and ν are the weights to control the importance of the corresponding consistency term.

Appendix D: Experiment datasets

KITTI dataset. The training set of KITTI includes 7,481 images/point clouds, divided into the train split (3,712 samples) and validation split (3,769 samples). The testing set contains 7,518 image/point clouds. **Waymo Open Dataset.** It includes 798 training sequences with around 158,361 LiDAR samples, and 202 validation sequences with 40,077 LiDAR samples. It annotates the objects in the full 360° field instead of 90° on KITTI. **SUN RGB-D Datasets.** It is a single-view and indoor RGB-D dataset for 3D detection. It includes 5,285 RGB-D training images annotated with amodal oriented 3D bounding boxes for 37 object categories.

REFERENCES

[1] L. J. C.-W. F. Wu Zheng, Weiliang Tang, "Se-ssd: Self-ensembling single-stage object detector from point cloud," in *CVPR*, pp. 14494–14503, 2021.

[2] M. F. Mozifian, *Real-time 3d object detection for autonomous driving*. PhD thesis, University of Waterloo, 2018.

[3] N. Zhao, T.-S. Chua, and G. H. Lee, "Sess: Self-ensembling semi-supervised 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11079–11087, 2020.

[4] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. Hinton, "Big Self-Supervised Models are Strong Semi-Supervised Learners," *arXiv:2006.10029 [cs, stat]*, June 2020. arXiv: 2006.10029.

[5] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations," *arXiv:2002.05709 [cs, stat]*, June 2020. arXiv: 2002.05709.

[6] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-View 3D Object Detection Network for Autonomous Driving," *arXiv:1611.07759 [cs]*, June 2017. arXiv: 1611.07759.

[7] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. Waslander, "Joint 3D Proposal Generation and Object Detection from View Aggregation," *arXiv:1712.02294 [cs]*, July 2018. arXiv: 1712.02294.

[8] M. Liang, B. Yang, Y. Chen, R. Hu, and R. Urtasun, "Multi-task multi-sensor fusion for 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7345–7353, 2019.

[9] J. H. Yoo, Y. Kim, J. Kim, and J. W. Choi, "3d-cvf: Generating joint camera and lidar features using cross-view spatial feature fusion for 3d object detection," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*, pp. 720–736, Springer, 2020.

[10] C.-H. Wang, H.-W. Chen, and L.-C. Fu, "Vpfnet: Voxel-pixel fusion network for multi-class 3d object detection," *arXiv preprint arXiv:2111.00966*, 2021.

[11] C. He, H. Zeng, J. Huang, X.-S. Hua, and L. Zhang, "Structure aware single-stage 3d object detection from point cloud," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11873–11882, 2020.

[12] W. Zheng, W. Tang, S. Chen, L. Jiang, and C.-W. Fu, "Cia-ssd: Confident iou-aware single-stage object detector from point cloud," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 3555–3562, 2021.

[13] Z. Liang, Z. Zhang, M. Zhang, X. Zhao, and S. Pu, "Rangeioudet: Range image based real-time 3d object detector optimized by intersection over union," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7140–7149, 2021.

[14] S. M. Laine and T. O. Aila, "Temporal ensembling for semi-supervised learning," Apr. 12 2018. US Patent App. 15/721,433.

[15] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, (Red Hook, NY, USA), p. 1195–1204, Curran Associates Inc., 2017.

[16] D. Berthelot, N. Carlini, E. D. Cubuk, A. Kurakin, K. Sohn, H. Zhang, and C. Raffel, "Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring," *arXiv preprint arXiv:1911.09785*, 2019.

[17] Q. Xie, Z. Dai, E. Hovy, M.-T. Luong, and Q. V. Le, "Unsupervised Data Augmentation for Consistency Training," *arXiv:1904.12848 [cs, stat]*, June 2020. arXiv: 1904.12848.

[18] H. Wang, Y. Cong, O. Litany, Y. Gao, and L. J. Guibas, "3dioumatch: Leveraging iou prediction for semi-supervised 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14615–14624, 2021.

[19] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[20] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660, 2017.

[21] K. Sohn, D. Berthelot, C.-L. Li, Z. Zhang, N. Carlini, E. D. Cubuk, A. Kurakin, H. Zhang, and C. Raffel, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," *arXiv preprint arXiv:2001.07685*, 2020.

[22] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li, "Pv-rcnn: Point-voxel feature set abstraction for 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10529–10538, 2020.

[23] H. Yang, Z. Liu, X. Wu, W. Wang, W. Qian, X. He, and D. Cai, "Graph r-cnn: Towards accurate 3d object detection with semantic-decorated local graph," *arXiv preprint arXiv:2208.03624*, 2022.

[24] X. Wu, L. Peng, H. Yang, L. Xie, C. Huang, C. Deng, H. Liu, and D. Cai, "Sparse fuse dense: Towards high quality 3d detection with depth completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5418–5427, 2022.

[25] T. Guan, J. Wang, S. Lan, R. Chandra, Z. Wu, L. Davis, and D. Manocha, "M3detr: Multi-representation, multi-scale, mutual-relation 3d object detection with transformers," *arXiv preprint arXiv:2104.11896*, 2021.

[26] T. Yin, X. Zhou, and P. Krahenbuhl, "Center-based 3d object detection and tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11784–11793, 2021.

[27] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.

[28] C. R. Qi, O. Litany, K. He, and L. J. Guibas, "Deep hough voting for 3d object detection in point clouds," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9277–9286, 2019.

[29] Z. Liu, Z. Zhang, Y. Cao, H. Hu, and X. Tong, "Group-free 3d object detection via transformers," *arXiv preprint arXiv:2104.00678*, 2021.

[30] D. Rukhovich, A. Vorontsova, and A. Konushin, "Fcaf3d: Fully convolutional anchor-free 3d object detection," *arXiv preprint arXiv:2112.00322*, 2021.

[31] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.