Technical Report for CVPR 2022 Workshop on Autonomous Driving Argoverse 3D Object Detection Competition

Jin Fang[1*], Qinghao Meng[3*], Dingfu Zhou[1], Chulin Tang[4],
Jianbing Shen[3], Cheng-Zhong Xu[2] and Liangjun Zhang[1]
[1] Robotics and Autonomous Driving Laboratory, Baidu Research
[2] University of Macau [3] Beijing Institute of Technology [4] University of California, Irvine
fangjin@baidu.com

Abstract

In this technical report, we show the details of our solution for 3D object detection competition in CVPR 2022 Workshop on Autonomous Driving. Furthermore, extending experiments are conducted to show the effectiveness of each sub-module of our solution.

1. Dataset and Metrics

First of all, we give a brief introduction to the dataset and metrics related to the competition.

Dataset. The Argoverse 2 Sensor Dataset [5] is employed for this competition, which includes 1,000 scenarios (750 for training, 150 for validation and 150 for testing) with 3D bounding box annotation. Each sequence lasts 15 seconds with 10 FPS for LiDAR frames. The synchronized camera images and HD maps are also available.

Metrics. 26 categories of objects within a 150-meter range are used for evaluation, e.g., regular vehicles and pedestrians, etc. Composite detection score (CDS) is used as the main metric, measured by the mAP and scale error, translation error, and orientation error together. More details can be found on the official website.

2. Proposed Method

The pipeline of our method is illustrated in Fig. 1. Firstly, the transformation metrics is used to align the previous 4 point cloud frames to current frame. Then, we split the merged point cloud to point clouds in the nearby region and far region. Secondly, we adopt the 3D object detection framework CenterPoint [9] with multiple resolutions and TTA / WBF fusion to obtain detection result. In the end, we reuse WBF fusion on multi-resolution and multi-range models to get the final result.

2.1. Baseline 3D Object Detector

CenterPoint [9] is a 3D object detection algorithm, which is proposed to represent and detect objects with rotationally invariant points. This method extracts feature from point cloud by a 3D sparse convolution network and then compresses the 3D feature map to 2D by height compression. Further, it adopts an elaborate anchor-free head for 3D object detection. Specifically, it involves a heatmap head and a regression head to predict the centers of objects with the help of rendered Gaussian kernels and estimate other properties of objects, including sub-voxel offset, height, sizes and rotation, respectively. In our implementation, we group the categories with similar shapes together and use a multi-head strategy to handle each of them. In addition, each heatmap head for classification only needs to focus on one class.

2.2. Multi-head Assignment

Argoverse 2 has 26 categories, and the assignment of multi-head is very important for the final performance. In Tab. 1 we show the categories for each head in our solution.

Head	Categories
1	BOX_TRUCK, TRUCK_CAB, TRUCK, ARTICULATED_BUS, BUS, SCHOOL_BUS
2	LARGE_VEHICLE, VEHICULAR_TRAILER, REGULAR_VEHICLE, MESSAGE_BOARD_TRAILER
3	BICYCLE, MOTORCYCLE
4	BICYCLIST, MOTORCYCLIST
5	CONSTRUCTION_CONE, DOG
6	BOLLARD, CONSTRUCTION_BARREL
7	PEDESTRIAN, STROLLER
8	SIGN, STOP_SIGN, MOBILE_PEDESTRIAN_CROSSING_SIGN
9	WHEELED_RIDER, WHEELED_DEVICE, WHEELCHAIR

Table 1: The categories for each head in our solution.

Since Argoverse benchmark provides multi-modal information, e.g., LiDAR, images and HDmap, the fusion of the information play a vital role in object detection. We refer to

[*] Equal contribution.

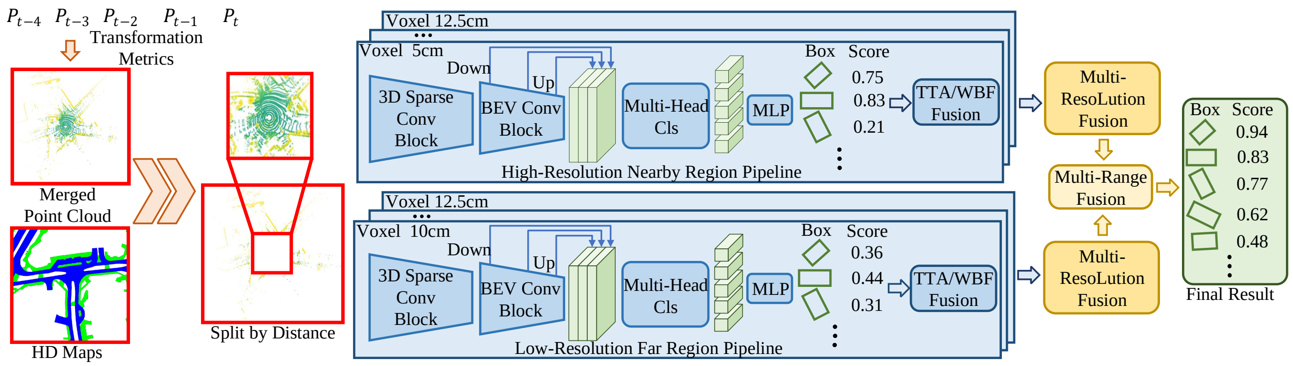

Figure 1: The pipeline of our solution.

MapFusion [1] to fuse the map information during training.

2.3. Training Details

Hyper-parameter. The epoch number is set as 20, the optimization algorithm is AdamW with a one-cycle learning rate policy, and the max learning rate for CenterPoint is 0.003, while the weight decay is 0.01 and momentum is 0.9. 3D IoU Loss [10] is employed during training.

Data Augmentation. Data augmentation is applied to guarantee data diversity and improve the robustness of the network. Our data augmentation strategies include 1) random rotation from $[-\frac{\pi}{4}, \frac{\pi}{4}]$ around the gravity axis, 2) random flipping, 3) random scaling from the uniform distribution 0.9 to 1.1 and 4) GT-aug [7] which copy the objects from other frame and paste into the current frame. Furthermore, 4 previous LiDAR sweeps are accumulated (temporal augmentation) to leverage the temporal information.

Multi-resolution Training. The resolution for voxelization affects the performance, higher resolution with smaller voxels tends to be easier to detect the small size objects (e.g., pedestrians, bicyclists), while the higher resolution with larger voxels has better behavior to detect the large size objects (e.g., buses, trucks). We use a multi-resolution strategy while training the model,more specifically, 4 resolutions (i.e. 0.050m, 0.075m, 0.100m, 0.125m) are used for the voxelization.

Multi-range Training. The evaluation range is 150m. It is GPU memory-consuming to load all the points and the voxels with high resolution into the memory. Thus we propose to divide the point cloud into two different ranges, i.e., 0 ˜ 60m and 60m ˜ 150m. The detection results from two ranges will be simply merged as final results.

2.4. Inference Details

Test Time Augmentation (TTA) and Weighted Box Fusion (WBF) are employed to obtain better results for each model trained with different voxel resolutions. Then the results are fused with WBF and filtered with a score threshold (e.g., 0.04).

Test Time Augmentation (TTA). During the inference process, we perform data augmentation on each input point cloud, we find this enforces the network to give more robust predictions. Specifically, three augmentation strategies are applied on the point cloud, 1) double flip (e.g., along X, Y or XY axes); 2) multiple rotations (e.g., ±11.25°, ±22.5°); 3) frame scaling (e.g., 0.95, 1.0, 1.05). The output boxes are transformed reversely and aggregated by WBF. We find this technique effectively improves the orientation estimation, which is of crucial importance under the metrics.

Weighted Box Fusion (WBF). Weighted Box Fusion [3] is an effective way to aggregate bounding boxes from different sources, which is originally proposed for 2D object detection. Firstly, it performs clustering on detected boxes according to the intersection-over-union (IoU). Then, it generates a new box from each cluster, whose center and size are calculated by weighted sums of each box in the cluster. As for the rotation, the one with the highest score will be directly adapted. And the final confidence score of the updated box is the average confidence of all the boxes in the cluster.

3. Experimental Results

In this section, we first show the results on the testing server and then we evaluate the effectiveness of each submodule in our solution on validation split.

3.1. Evaluation on Testing Server

We submit our results to the testing server [1], and the result is shown in Tab. 2. Our solution outperforms the baseline with 0.20 for mCDS and 0.23 for mAP.

3.2. Ablation Studies

In this section, we search for the best combination of the hyper-parameters and evaluate the effectiveness of the

[1] https://eval.ai/challenge/1710/leaderboard/4078

Method	mCDS	mAP	mATE	mASE	mAOE
Baseline	0.14	0.18	0.49	0.34	0.72
Ours	**0.34**	**0.41**	0.40	0.30	0.54

Table 2: The evaluation result on the testing split of Argoverse benchmark. The result of Baseline is provided by the official team.

techniques. All the results are evaluated on validation split.

Multi-resolution training. In Tab. 3, we can find that compared with the results with different resolutions, the fusion result achieves the best performance.

Method	mCDS	mAP	mATE	mASE	mAOE
voxel (0.050m)	0.229	0.282	0.406	0.308	0.563
voxel (0.075m)	0.226	0.278	0.405	0.308	0.588
voxel (0.100m)	0.210	0.260	0.424	0.306	0.611
Fusion	**0.238**	**0.292**	0.396	0.304	0.570

Table 3: Evaluation results with different voxel resolutions, the method is based on CenterPoint without TTA and temporal augmentation. All the models are trained on the full training data within 60m range.

TTA. The results of TTA can be found in Tab. 4, from where we can draw the conclusion that TTA can effectively improve performance. Notice that with TTA (× 60), the overall mCDS is improved but also brings damage to mASE. We guess that is caused by the scaling augmentation, so we remove the frame scaling in TTA for the final submission.

Method	mCDS	mAP	mATE	mASE	mAOE
voxel (0.100m)	0.210	0.260	0.424	0.306	0.611
+TTA (× 4)	0.222	0.273	0.413	0.302	0.604
+TTA (× 60)	**0.236**	**0.291**	0.300	0.617	0.236

Table 4: Evaluation results on TTA strategy. TTA (× 4) uses only frame flipping along X,Y and XY axes while TTA (× 60) uses combinations with the augmentation strategies introduced in Sec. 2. All the models are trained on the full training data within 60m range.

Multi-heads Assignment. In Tab. 5, we can find that our multi-head assignment can significant improve the performance. The baseline is from the official code [2].

Method	mCDS	mAP	mATE	mASE	mAOE
Baseline multi-head	0.139	0.181	0.419	0.288	0.690
Our multi-head	**0.195**	**0.246**	0.388	0.285	0.639

Table 5: Evaluation results on different multi-head assignments. All the models are trained on the 10% training data in 60m range without TTA and temporal augmentation.

Temporal Augmentation. Tab 6 shows that with the temporal augmentation, the performance can get extra gain.

Method	mCDS	mAP	mATE	mASE	mAOE
voxel (0.100m)	0.195	0.246	0.388	0.285	0.639
+ Temporal Aug	**0.211**	**0.265**	0.424	0.277	0.508

Table 6: Evaluation results on temporal augmentation strategy. All the models are trained on the 10% training data in 60m range without TTA.

Multi-range Fusion. Tab 7 shows the fusion result by the models trained with two different ranges, 0 ˜ 60m and 60m ˜ 150m. It is obvious that model (0 ˜ 60m) contributes most of the performance, and there is still a huge improvement space for the far-distance objects.

Method	mCDS	mAP	mATE	mASE	mAOE
[0 ˜ 60m]	0.195	0.246	0.388	0.285	0.639
[60m ˜ 150m]	0.018	0.026	0.673	0.423	1.164
[0 ˜ 60m] + [60m ˜ 150m]	**0.201**	**0.262**	0.449	0.308	0.776

Table 7: Evaluation results on multi-range fusion. All the models are trained with voxel (0.100m) setting on the 10% training data without TTA.

4. Future Works

PointPainting [4] and FusionPainting [6] are two potential methods to fuse the images and semantic information, which will be investigated in the future. In addition, there are still a large number of unlabeled scenes in the dataset, which can further improve the generalization ability of the model in training. WS3D [2] and Auto4D [8] are also potential methods for training the network with incomplete annotation in the future.

References

[1] Jin Fang, Dingfu Zhou, Xibin Song, and Liangjun Zhang. Mapfusion: A general framework for 3d object detection with hdmaps. In 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 3406–3413. IEEE, 2021. 2

[2] Qinghao Meng, Wenguan Wang, Tianfei Zhou, Jianbing Shen, Yunde Jia, and Luc Van Gool. Towards a weakly supervised framework for 3d point cloud object detection and annotation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021. 3

[3] Roman Solovyev, Weimin Wang, and Tatiana Gabruseva. Weighted boxes fusion: ensembling boxes for object detection models. arXiv e-prints, 2019. 2

[4] Sourabh Vora, Alex H Lang, Bassam Helou, and Oscar Beijbom. Pointpainting: Sequential fusion for 3d object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 4604–4612, 2020. 3

[5] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes,

[2] https://github.com/benjaminrwilson/torchbox3d

Deva Ramanan, Peter Carr, and James Hays. Argoverse 2: Next generation datasets for self-driving perception and forecasting. In Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks 2021), 2021. 1

[6] Shaoqing Xu, Dingfu Zhou, Jin Fang, Junbo Yin, Zhou Bin, and Liangjun Zhang. Fusionpainting: Multimodal fusion with adaptive attention for 3d object detection. In 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), pages 3047–3054. IEEE, 2021. 3

[7] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. Sensors, 18(10):3337, 2018. 2

[8] Bin Yang, Min Bai, Ming Liang, Wenyuan Zeng, and Raquel Urtasun. Auto4d: Learning to label 4d objects from sequential point clouds. arXiv preprint arXiv:2101.06586, 2021. 3

[9] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 11784–11793, 2021. 1

[10] Dingfu Zhou, Jin Fang, Xibin Song, Chenye Guan, Junbo Yin, Yuchao Dai, and Ruigang Yang. Iou loss for 2d/3d object detection. In 2019 International Conference on 3D Vision (3DV). IEEE, 2019. 2