Exploring Simple 3D Multi-Object Tracking for Autonomous Driving
Supplementary Material

Chenxu Luo[1,2] Xiaodong Yang[1*] Alan Yuille[2]
[1]QCraft [2]Johns Hopkins University

In this supplementary material, Section 1 exemplifies the representative heuristics in matching and studies how the related hyper-parameters impact the final tracking performance for existing methods. Section 2 presents more comparisons between our approach SimTrack and the state-of-the-art method CenterPoint. Section 3 reports the inference latency of our model under different settings. Section 4 provides more results on nuScenes and Waymo.

1. Heuristic Matching and Hyper-Parameters

Existing tracking methods involve a number of hyper-parameters in heuristic matching. Some widely used ones include matching threshold, maximum number of frames to keep for a dead track, minimum number of frames before initializing a new track, to name a few.

It is known that the tracking performance is sensitive to the hyper-parameter setting in heuristic matching. For the Kalman filter based tracking, the setting of covariance matrix greatly affects the tracking result. For instance, in [6] the AMOTA on the validation set of nuScenes is 37.1 when using the default covariance matrix, but the performance boosts to 51.2 after carefully tuning the covariance matrix based on the statistics of prediction errors.

To highlight the critical role of setting hyper-parameters for the heuristic matching step, we compare the tracking results of CenterPoint [31] with different hyper-parameters in Table 5. Specifically, we exemplify with two representative hyper-parameters: maximum age and maximum distance. The former is used for a dead track to be retained for a certain number of frames before it is removed. This helps when an object is occasionally occluded in a few frames and shows up again. The latter determines the distance threshold that allows to be matched. CenterPoint tunes this threshold based on the distribution of velocity errors on the validation set. As demonstrated in Table 5, the two factors significantly impacts the tracking performance. To obtain a reasonably good result, great efforts are in need to tune these hyper-parameters. As a comparison, our approach is heuristic-free but achieves better performance.

Model	Age	Distance	AMOTA↑	AMOTP↓	IDS↓	FRAGS↓
CenterPoint	3	1.0	81.0	43.3	856	247
	3	2.0	83.1	39.5	256	184
	3	4.0	82.5	48.0	238	240
	3	∞	59.6	49.6	318	299
	0	4.0	80.0	48.0	365	352
Ours	-	-	**84.1**	**34.5**	**148**	**122**

Table 5: Impact of the representative heuristics and the setting of related hyper-parameters in the matching step of CenterPoint. We report the tracking results on the validation set (car category) of nuScenes. All results are produced by the pillar based backbone.

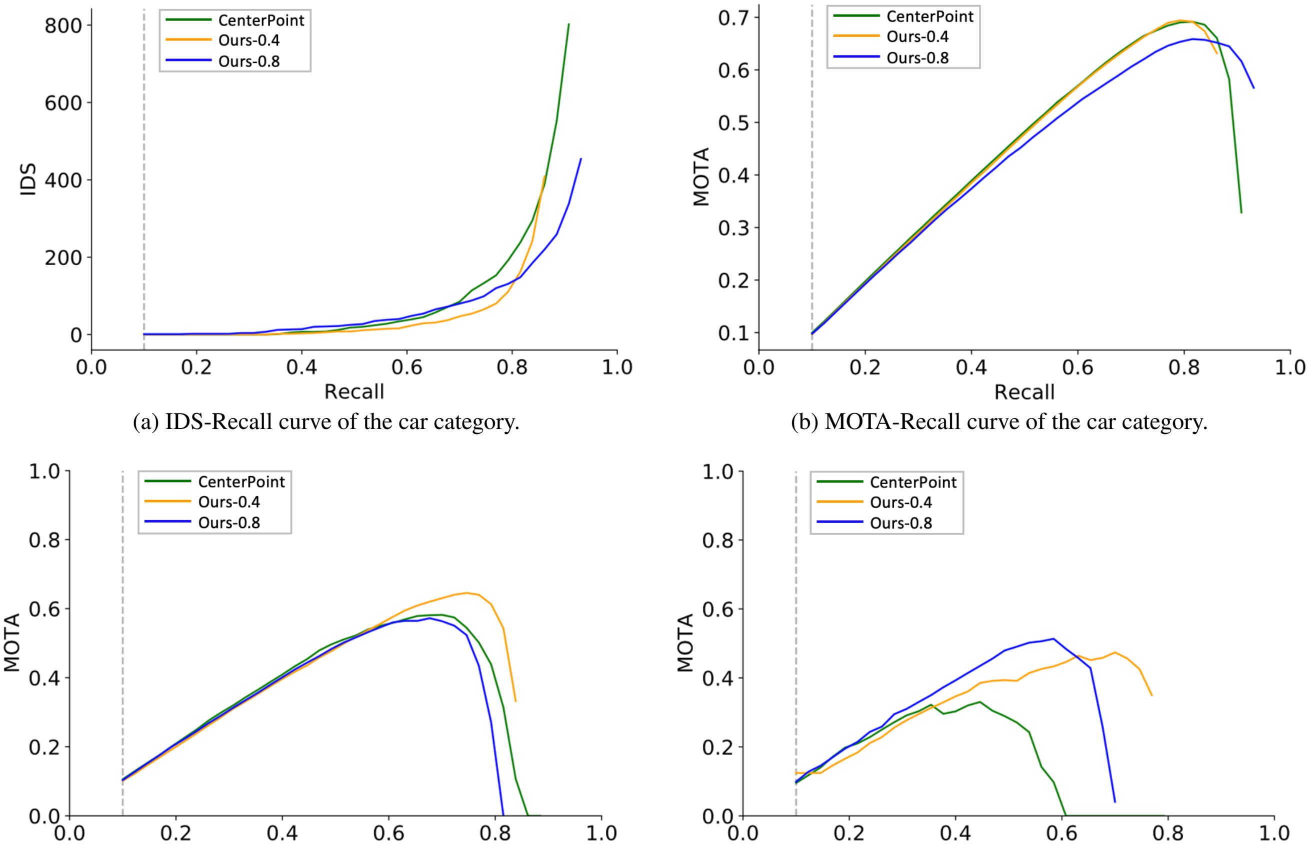
2. More Comparisons with CenterPoint

Here we provide more detailed comparisons on MOTA and IDS between SimTrack and CenterPoint under different recall rates. As shown in Figure 4a, our model has much less identity switch under high recall rates. This is because the heuristic matching based tracking methods like CenterPoint suffer from the large amount of false positive detections, while SimTrack is less vulnerable to false positives thanks to our joint detection and tracking design. This advantage makes our approach more robust and stable in particular for the scenarios where a high recall rate is desired. Figures 4b-4d respectively plot the curves of MOTA-Recall for car, pedestrian and motorcycle. Overall, our model achieves superior MOTA at high recall rates.

3. Inference Latency

Our joint detection and tracking design is flexible to incorporate in a 3D object detection network and only introduces a small computational overhead to to the backbone network. Table 6 compares the inference latency between a detection-only model and our joint detection and tracking model using different centerness map resolutions with the pillar and voxel based backbones. As shown in this table, our approach only slightly increases the inference latency of the detection-only model by 1-2ms. We report the inference time on a single TITAN RTX GPU.

[*] Correspondence to xiaodong@qcraft.ai



(a) IDS-Recall curve of the car category.

(b) MOTA-Recall curve of the car category.

(c) MOTA-Recall curve of the pedestrian category.

(d) MOTA-Recall curve of the motorcycle category.

Figure 4: Comparisons on IDS and MOTA between SimTrack and CenterPoint under different recall rates. All results are produced by the pillar based backbone. Ours-0.4 (0.8) denote the resolution of centerness map: 0.4m×0.4m (0.8m×0.8m). Ours-0.8 is the default resolution. See Section 4.5 in the paper for more details about the resolution.

Resolution	Pillar Backbone	Voxel Backbone
0.4m×0.4m	36ms / 38ms	65ms / 67ms
0.8m×0.8m	33ms / 34ms	63ms / 64ms

Table 6: Comparison of inference latency of the detection-only model vs. our joint detection and tracking model using different centerness map resolutions and backbones.

4. More Results on nuScenes and Waymo

In addition to simplify and improve tracking, SimTrack can also boost the detection accuracy. Table 7 compares the detection results of SimTrack and CenterPoint. We report mAP and NDS of all classes on nuScenes. Note the result on the test set of CenterPoint is also based on its enhanced version as described in the paper. Our joint detection and tracking model can significantly improve the detection performance. In Table 8, we provide more results of SimTrack on Waymo. We employ the pillar based backbone and adopt the dynamic voxelization proposed in [1] to replace the hard voxelization as used in all other experiments.

Method	PointPillars (v)		VoxelNet (v)		VoxelNet (t)	
	mAP	NDS	mAP	NDS	mAP	NDS
CenterPoint	50.3	60.2	56.4	64.8	58.0	65.5
Ours	**55.5**	**64.9**	**60.1**	**67.6**	**61.3**	**67.6**

Table 7: Comparison of the 3D object detection results on the validation (v) and test (t) sets of nuScenes.

Class	MOTA↑	Miss↓	Miss Match↓	FP↓
Vehicle	54.3 / 50.7	34.6 / 38.8	0.20 / 0.19	10.9 / 10.4
Pedestrian	58.3 / 53.9	31.5 / 35.2	0.60 / 0.57	10.5 / 10.3

Table 8: We report the tracking performance using dynamic voxelization on the validation set of Waymo, and the numbers are in the format LEVEL_1 / LEVEL_2.

References

[1] Yin Zhou, Pei Sun, Yu Zhang, Dragomir Anguelov, Jiyang Gao, Tom Ouyang, James Guo, Jiquan Ngiam, and Vijay Vasudevan. End-to-end multi-view fusion for 3D object detection in lidar point clouds. In *CoRL*, 2020. 2