QUALITY MATTERS: EMBRACING QUALITY CLUES FOR ROBUST 3D MULTI-OBJECT TRACKING

Anonymous authors
Paper under double-blind review

ABSTRACT

3D Multi-Object Tracking (MOT) has achieved tremendous achievement thanks to the rapid development of 3D object detection and 2D MOT. Recent advanced works generally employ a series of object attributes, *e.g.,* position, size, velocity, and appearance, to provide the clues for the association in 3D MOT. However, these cues may not be reliable due to some visual noise, such as occlusion and blur, leading to tracking performance bottleneck. To reveal the dilemma, we conduct extensive empirical analysis to expose the key bottleneck of each clue and how they correlate with each other. The analysis results motivate us to efficiently absorb the merits among all cues, and adaptively produce an optimal tacking manner. Specifically, we present *Location and Velocity Quality Learning*, which efficiently guides the network to estimate the quality of predicted object attributes. Based on these quality estimations, we propose a quality-aware object association (QOA) strategy to leverage the quality score as an important reference factor for achieving robust association. Despite its simplicity, extensive experiments indicate that the proposed strategy significantly boosts tracking performance by 2.2% AMOTA and our method outperforms all existing state-of-the-art works on nuScenes by a large margin. Moreover, QTrack achieves 48.0% and 51.1% AMOTA tracking performance on the nuScenes validation and test sets, which significantly reduces the performance gap between pure camera and LiDAR based trackers.

1 INTRODUCTION

3D Multi-Object Tracking (MOT) has been recently drawing increasing attention since it is widely applied to 3D perception scenes, e.g., autonomous driving, and automatic robot. The 3D MOT task aims at locating objects and associating the targets of the same identities to form tracklets. According to the used sensors, existing 3D MOT methods can mainly be categorized into two classes, i.e., camera-based and LiDAR-based schemes. In this paper, we mainly delve into the camera-only scheme since it contains semantic information and is more economical.

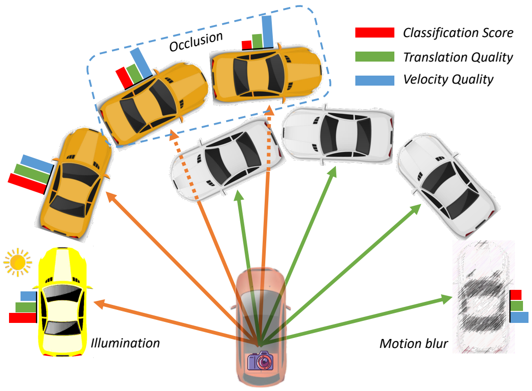
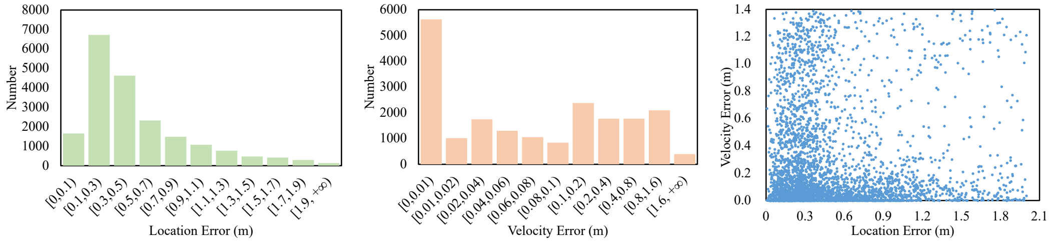


Figure 1: Illustration of three type of hard cases: (1) illumination of external, (2) occlusion, (3) motion blur. The red, green and blue pillars are organized to represent the classification score, location quality, and velocity quality, where the higher pillars indicate higher values.

Existing 3D MOT methods mostly adopt the tracking-by-detection paradigm. In this regime, a 3d detector is firstly employed to predict 3D boxes and the corresponding classification scores, and then some post-processing methods (e.g., motion-based Kalman (1960) or appearance-based) are used to line detected targets to form trajectories. In the camera scheme, it is natural to extract objects' discriminative appearance features Chaabane et al. (2021); Hu et al. (2022) to represent targets and use the features to measure the similarities among detected targets. However, the procedure of extracting the appearance feature is cumbersome since it requires predicting high-dimensional



(a) Location Quality Distribution (b) Velocity Quality Distribution (c) Correlation

Figure 2: Statistics of location and velocity quality distribution and their correlation on nuScenes *val* dataset. (a) and (b) reveal that the accuracy of all prediction locations and velocity are varying on an unfixed scale, showing irregularity. (c) shows a messy scatter plot, which reflects no relations between location and velocity results.

embedding, which is hard for joint training due to the optimization contradiction between the detection and embedding branches Yu et al. (2022b). Moreover, it is difficult to deal with the notorious occlusion and motion blur issues. Some other methods Weng et al. (2020a); Pang et al. (2021) build a motion model (Kalman Filter) to obtain some desired states of tracking clues (e.g, center position, size, size ratio, or rotation) by a linear motion assumption. Nevertheless, this process involves various hyper-parameters (e.g., initialization uncertainty of measurement, state and process, etc.) and executes complex matrix transpose operation. Different from the aforementioned methods, CenterPoint Yin et al. (2021) reasonably leverages predicted center locations and velocities of targets for building motion. In detail, it uses time lag between two moments of observations to multiply the predicted velocity for linear location prediction. Afterwards, the L2 distance among targets acts as a measurement metric for the association procedure. For simplicity, we call this tracking framework CV method. It shows effectiveness to achieve remarkable tracking performance, while only conducting a simple operation (i.e., matrix addition and multiplication) for parallel cost computation.

Although the CV framework shows efficiency for 3D MOT tasks, it relies heavily on the predicted quality of center location and velocity. The requirement may be harsh for the 3D base detector, since estimating the center location and velocity of an object from a single image is exactly an ill-posed problem. As shown in Fig. 1, notorious occlusion, motion blur, and the illumination of external issues will significantly disturb the estimation performance. To further confirm this issue, we conduct an empirical analysis to study the predicted center location and velocity quality distribution as well as their correlations. Our study reveals two valuable points: (1) There exists a significant gap between the estimation error of 3D centers and that of velocities; (2) The predicted quality of location and velocity is extremely misaligned. The imbalanced tracking cues have little effect on the detection performance but play a dramatic role in MOT. The analysis cues motivate us to endow each predicted box with the self-diagnosis ability to tracking clues for realizing stable tracking association.

To this end, we propose to forecast the quality of tracking clues from the base 3D detector. Specifically, we introduce a Normalized Gaussian Quality (NGQ) metric with two dimensions to measure the quality of predicting center location and velocity. NGQ metric comprehensively considers the vector errors of the two predictions in a 2D vector space, which is a prerequisite for our tracking framework. Based on the quality estimation of NGQ, we design a robust association mechanism, i.e, the Quality-aware Object Association (QOA) strategy. It adopts the velocity quality to filter out low-quality motion candidates, and leverages the location quality to further rule out center positions of boxes with bad estimations. Therefore, QOA not only effectively deals with hard cases but also avoids dangerous associations. In a sense, our method is subordinate to the idea of ”Put Quality Before Quantity” principle.

By combining the proposed methods with the baseline 3D detector, we obtain a simple and robust 3D MOT framework, namely *quality-aware 3D tracker* (QTrack). We conduct extensive experiments on nuScenes dataset Caesar et al. (2020), showing significant improvements in the 3D MOT task. Comprehensively, the contributions of this work are summarized as follows:

- We conduct extensive empirical analysis to point out that the predicted quality of center location and velocity exist a large distribution gap and misalignment relationship, making an efficient CV tracking framework fall into sub-optimal performance.
- We first propose to predict the quality of velocity and location quality measured by our designed NGQ metric. Afterwards, we further introduce QOA to leverage the two qualities for insuring safe association in 3D MOT task.
- The overall 3D MOT framework (QTrack) achieves SOTA performance on nuScenes dataset which outperforms other camera-based methods by a large margin. Specially, QOA improves the baseline tracker by +2.2% AMOTA among several 3D detector settings, showing its effectiveness.

2 Related Work

2.1 3D Multi-object Tracking

Thanks to the development of 3D detection Huang et al. (2021); Li et al. (2022a); Liu et al. (2022) and 2D MOT technologies Han et al. (2022); Yu et al. (2022b;a); Zhang et al. (2022b), recent 3D MOT methods Weng et al. (2020a); Yin et al. (2021); Chaabane et al. (2021); Hu et al. (2022); Pang et al. (2021) mainly follow tracking-by-detection paradigm. These trackers following this paradigm first utilize a 3D object detector to localize the targets in the 3D space (including location, rotation, and velocity) and then associate the detected objects with the trajectories according to various cues (location or appearance).

Traditional 3D MOT usually uses a motion model (Kalman filter) to predict the location of the tracklets and then associate the candidate detections using 3D (G)IoU Weng et al. (2020a); Pang et al. (2021) or L2 distance Yin et al. (2021). Some works also utilize advanced appearance model (ReID) Chaabane et al. (2021); Weng et al. (2020b); Chaabane et al. (2021) or temporal model (LSTM) Marinello et al. (2022); Hu et al. (2022) to provide more reference cues for the association. Recently, Transformer Vaswani et al. (2017) has been used in 3D detection Wang et al. (2022) and MOT Li & Jin (2022); Zhang et al. (2022a) to learn 3D deep representations with 2D visual information and trajectory encoded. Although these methods achieved remarkable performance, when they are applied to complex scenarios (e.g., occlusion, motion blur, or light weakness), the tracking performance becomes unsatisfactory. In this work, we argue that a simple velocity clue with quality estimation can deal with the corner cases and achieve robust tracking performance. Our proposed QTrack focuses on how to assess the quality of the location and velocity prediction, and then make full use of these quality scores in the matching process.

2.2 Prediction Quality Estimation

To estimate the quality of model's prediction is non-trivial, which can be applied to tackle prediction imbalance or decision-making. In the field of object detection, advanced works Wang et al. (2021); Tian et al. (2019); Jiang et al. (2018) introduce to predict a box's centerness or IoU for perceiving the quality of prediction (3D) boxes. Huang et al. (2019) employ the method to perceive the mask predicted quality. These methods can alleviate the imbalance between classification score and location accuracy. Li et al. (2022c) introduces an uncertainty-based method to estimate the predicted quality of several depth factors, and then the quality is employed to make optimal decisions. In this paper, we introduce to predict the predicted quality of velocity and location. Afterwards, the predicted quality will be used to eliminate the non-robust association case of tracking task. To our knowledge, our work is the first effort to perceive the velocity and location qualities for the decision-making in 3D MOT task.

2.3 Multi-View 3D Object Detection

3D object detection is the predecessor task for 3D MOT task. It can be split into two stream methods including point-based Lang et al. (2019); Yan et al. (2018); Yin et al. (2021); Shi et al. (2019; 2020); Yang et al. (2022c) and camera-based detectors Wang et al. (2021); Huang et al. (2021); Li et al. (2022a); Wang et al. (2022); Liu et al. (2022); Li et al. (2022b). In this paper, we focus on the 3D

MOT for the multi-view camera based framework, which has made tremendous advance. Transformer based methods Wang et al. (2022); Liu et al. (2022); Li et al. (2022b) introduce 3D object queries to interact with the multi-view image feature map. 3D object queries are constantly refined to predict 3D boxes and other tasks in an end-to-end manner. BEVDet Huang et al. (2021) and BEVDepth Li et al. (2022a) directly project the multi-view image feature into BEV representation and attach a center-based head Yin et al. (2021) to conduct detection task. Standing on the shoulders of giants, we aim to equip BEVDepth with the ability to perceive the quality of velocity and center locatopn, which is the key to diagnose non-robust association for tracking. Then we introduce a novel "tracking by detection" (QTrack) to endow BEVDepth with effiective and efficient tracking.

3 Methodology

3.1 Delve into the Quality Distribution

We aim to solve the task of 3D multi-object tracking (3D MOT), the goal of which is to locate the objects in the 3D space and then associate the detected targets with the same identity into the tracklets. The key challenge is how to associate the tracklets efficiently and correctly. In contrast to the motion-based and appearance-based association strategies, we argue that the simple velocity clue (CV method) is enough for the association, which is more lightweight and deployment-friendly. However, the performance of the existing CV tracking framework is not satisfactory. To analyze the reason for the limited performance of tracking with velocity, we count and visualize the distribution of the prediction error between location and velocity. As illustrated in Fig. 2 (a) and (b), we can observe that the distribution of the location and velocity quality (prediction error) is scattered, and a sizable number of low-quality boxes are included. Moreover, Fig. 2 (c) shows that the distribution correlation between the location and velocity error is nonlinear, which means the quality of the location and velocity is seriously misaligned.

Based on these observations, we conclude that the limited performance of tracking with velocity is due to the following reasons: (1) Low quality of the location or velocity. When one of the location and velocity predictions is not accurate enough, the tracker can not perform well even if the other prediction is reliable. (2) Misalignment between the quality of location and velocity. We should take both location and velocity quality into consideration. Driven by this analysis, we propose *Location and Velocity Quality Learning* to learn the quality uncertainty of the location, and velocity that can assist the tracker to select high-quality candidates for the association.

3.2 Base 3D Object Detector

Our method can be easily coupled with most existing 3D object detectors with end-to-end training. In this paper, we take BEVDepth Li et al. (2022a) as an example. BEVDepth is a camera-based Bird's-Eye-View (BEV) 3D object detector that transfers the multi-view image features to the BEV feature through a depth estimation network and then localizes and classifies the objects in the BEV view. It consists four kinds of modules: an image-view encoder, a view transformer with explicit depth supervision utilizing encoded intrinsic and extrinsic parameters, a BEV encoder and a task-specific head. The entire network is optimized with a multi-task loss function:

$$\mathcal{L}_{det} = \mathcal{L}_{depth} + \mathcal{L}_{cls} + \mathcal{L}_{reg}, \tag{1}$$

where the depth loss $\mathcal{L}_{depth}$, classification loss $\mathcal{L}_{cls}$ and regression loss $\mathcal{L}_{reg}$ remain the same setting as the original paper. As illustrated in Fig. 3, the task of the regression branch includes heatmap, offsets, height, size, rotation and velocity.

3.3 Location and Velocity Quality Learning

To effectively estimate the quality of location and velocity, it first needs to define the quality measurement metric. Technically, the box's center location is calculated by incorporating predicted heatmap and corresponding offsets so that the location quality can be simplified to offset predicted quality. Specially, the offsets and velocity are defined in a 2-dimensional vector space. We introduce a Normalized Gaussian Quality (NGQ) metric to represent their quality.

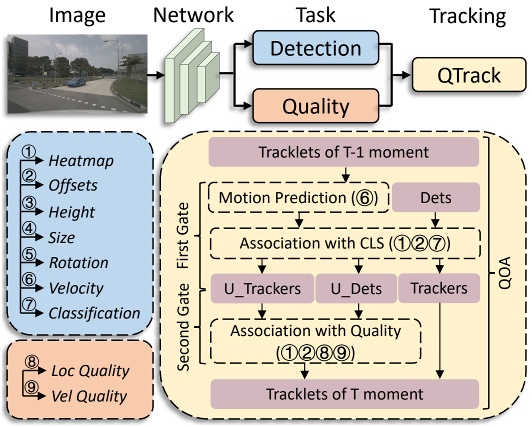


Figure 3: Overview of base 3D detector and QTrack. The multi-view images are first fed into detector, i.e., BEVDepth. Then we add two parallel branches for predicting location and velocity quality, respectively. For QTrack, it first employs velocity clue to conduct motion predicted, and then adopts heatmap, offsets, and classification score to carry out association procedure in the first gate stage. Specially, location and velocity qualities are introduced to execute this work's key module QOA for unmatched trackers and detections in the second gate stage.

Given a predicted vector $\mathbf{P} \in \mathbb{R}^2$ and ground truth vector $\mathbf{G} \in \mathbb{R}^2$, we formulate NGQ metric as:

$$\mathrm{NGQ} = e^{-\frac{\sqrt{(\mathrm{P}_x - \mathrm{G}_x)^2 + (\mathrm{P}_y - \mathrm{G}_y)^2}}{\gamma}}, \tag{2}$$

where the subscripts x and y indicate the value in the x and y directions while γ is a hyper-parameter to control the value distribution of NGQ. We set γ to 1.0 and 3.0 for location and velocity, respectively. $\mathbf{P}$ and $\mathbf{G}$ can be instantiated as predicting offset and velocity. When the prediction is equal to ground truth, NGQ = 1, while the predicted error is larger, NGQ is closer to 0.

After defining the quality, we elaborate on how to learn it. As shown in Fig. 3, we attach a 3×3 convolution layer for offset and velocity branch to predict location quality $\mathrm{NGQ}^{loc} \in \mathbb{R}^1$ and velocity quality $\mathrm{NGQ}^{vel} \in \mathbb{R}^1$, respectively. The quality supervision is conducted by binary cross entropy (BCE) loss:

$$\begin{aligned}\mathcal{L}_{quality} = -\frac{1}{N}\sum_{i=1}^{N}[\mathrm{N\hat{G}Q_i} \cdot \log \mathrm{NGQ_i} \\ + (1 - \mathrm{NGQ_i}) \cdot \log{(1 - \mathrm{N\hat{G}Q_i})}],\end{aligned} \tag{3}$$

where $\mathrm{N\hat{G}Q}$ is the ground truth quality calculated by Eq. 2. This far, the total loss for our detector is formulated as:

$$\mathcal{L}_{total} = \mathcal{L}_{det} + \mathcal{L}_{quality}. \tag{4}$$

The overall training procedure is an end-to-end manner while the quality prediction task will not damage the performance of the base detector. Moreover, the quality estimation is used in our proposed Quality-aware Object Association (QOA) module, which will be discussed next section.

3.4 Quality-aware Object Association

After obtaining the quality of the center location and velocity, we have more reference cues to achieve robust and accurate association. To this end, we propose a simple but effective quality-aware object association strategy (QOA). Specifically, QOA sets up two ”gates”. The first gate is the classification confidence score (cls score). We first separate the candidate detection boxes into high score ones and low score ones according to their cls scores. The high score candidates are first associated with the tracklets. Then the unmatched tracklets are associated with the low score candidates. These low score candidates are most caused by occlusion, motion blur, or light weakness, which are easily confused with the miscellaneous boxes. To deal with the issue, the second gate, quality uncertainty score, is introduced. After getting the second association results between the unmatched tracklets and the low score candidates, we then recheck the matched track-det pairs according to the location and velocity quality scores. Only high-quality matched track-det pairs can remain and low-quality pairs are regarded as the mismatch. The pseudo-code of QOA is shown in Algorithm 1.

Benefiting from the quality estimation, QOA does not need a complex motion or appearance model to provide association cues. A simple velocity prediction (CV) is enough (line #15). Hence, we use the velocity of the tracklet at frame $t-1$ to predict the center location at frame t and then

Algorithm 1: Pseudo-code of QOA.

```
Input: A video sequence V; object detector Det; detection score threshold $\tau$; quality score threshold $\mu_v$, $\mu_t$
Output: Tracks $\mathcal{T}$ of the video
1  Initialization: $\mathcal{T} \leftarrow \emptyset$
2  for frame $f_k$ in V do
       /* boxes & scores */
3      $\mathcal{D}_k \leftarrow$ Det($f_k$)
4      $\mathcal{D}_{high} \leftarrow \emptyset$
5      $\mathcal{D}_{low} \leftarrow \emptyset$
       /* first gate */
6      for $d$ in $\mathcal{D}_k$ do
7          if $d.score > \tau$ then
8              $\mathcal{D}_{high} \leftarrow \mathcal{D}_{high} \cup \{d\}$
9          end
10         else
11             $\mathcal{D}_{low} \leftarrow \mathcal{D}_{low} \cup \{d\}$
12         end
13     end
       /* predict location */
14     for $t$ in $\mathcal{T}$ do
15         $t \leftarrow$ CV($t$)
16     end
       /* association with high scores */
17     Associate $\mathcal{T}$ and $\mathcal{D}_{high}$ using L2 distance
18     $\mathcal{D}_{remain} \leftarrow$ remaining object boxes from $\mathcal{D}_{high}$
19     $\mathcal{T}_{remain} \leftarrow$ remaining tracks from $\mathcal{T}$
       /* association with low scores */
20     Associate $\mathcal{T}_{remain}$ and $\mathcal{D}_{low}$ using L2 distance
21     $\mathcal{T}_{sec}, \mathcal{D}_{sec} \leftarrow$ matched pairs from $\mathcal{T}_{remain}$ ,$\mathcal{D}_{low}$
22     $\mathcal{T}_{re-remain} \leftarrow$ remaining tracks from $\mathcal{T}_{remain}$
       /* second gate */
23     for $t$, $d$ in $\mathcal{T}_{sec}$, $\mathcal{D}_{sec}$ do
24         if $t.v_{score} < \mu_v$ or $d.l_{score} < \mu_t$ then
25             $\mathcal{T}_{re-remain} \leftarrow \mathcal{T}_{re-remain} \cup \{t\}$
26         end
27     end
       /* update and initialize */
28     $\mathcal{T} \leftarrow \mathcal{T} \setminus \mathcal{T}_{re-remain}$
29     for $d$ in $\mathcal{D}_{remain}$ do
30         $\mathcal{T} \leftarrow \mathcal{T} \cup \{d\}$
31     end
32 end
33 Return: $\mathcal{T}$
```

compute the L2 distance between predictions and candidate detections (line #17 and line #20) as the similarity. At last, we apply the similarity with the Hungarian algorithm to get the association results. Mathematically,

$$\begin{aligned} &c_t = c_{t-1} + v_{t-1}\Delta t \\ &cost = \mathcal{L}_2(c_t, d_t) \\ &match = Hungarian(cost), \end{aligned} \tag{5}$$

where c_{t-1}, v_{t-1} represents the center location and velocity of the tracklets at frame $t-1$. d_t is the candidate detection center location at frame t and Δt is the time lag.

Methods	Modality	AMOTA ↑	AMOTP ↓	RECALL ↑	MOTA ↑	IDS ↓
		Validation Split				
CenterPoint Yin et al. (2021)	LiDAR	0.665	0.567	69.9%	0.562	562
SimpleTrack Pang et al. (2021)	LiDAR	0.687	0.573	72.5%	0.592	519
DEFT Chaabane et al. (2021)	Camera	0.201	N/A	N/A	0.171	N/A
QD3DT Hu et al. (2022)	Camera	0.242	1.518	39.9%	0.218	5646
TripletTrack Marinello et al. (2022)	Camera	0.285	1.485	N/A	N/A	N/A
MUTR3D Zhang et al. (2022a)	Camera	0.294	1.498	42.7%	0.267	3822
QTrack (Ours)	Camera	**0.511**	**1.090**	**58.5%**	**0.465**	**1144**
		Test Split				
CenterTrack Zhou et al. (2020)	Camera	0.046	1.543	23.3%	0.043	3807
DEFT Chaabane et al. (2021)	Camera	0.177	1.564	33.8%	0.156	6901
Time3D Li & Jin (2022)	Camera	0.210	1.360	N/A	0.173	N/A
QD3DT Hu et al. (2022)	Camera	0.217	1.550	37.5%	0.198	6856
TripletTrack Marinello et al. (2022)	Camera	0.268	1.504	40.0%	0.245	**1044**
MUTR3D Zhang et al. (2022a)	Camera	0.270	1.494	41.1%	0.245	6018
PolarDETR Chen et al. (2022)	Camera	0.273	1.185	40.4%	0.238	2170
SRCN3D Shi et al. (2022)	Camera	0.398	1.317	53.8%	0.359	4090
QTrack (Ours)	Camera	**0.480**	**1.107**	**56.9%**	**0.431**	1484

Table 1: Comparison with state-of-the-art methods on nuScenes validation and test dataset. Our QTrack employs VoVNet-99 initialized from DD3D as image backbone. The resolution of input image is 640 × 1600.

Methods	Backbone	AMOTA ↑	AMOTP ↓	RECALL ↑	MOTA ↑	MOTP ↓	IDS ↓	FRAG ↓	MT ↑	ML ↓
BEVDepth + KF	ResNet-50	0.303	1.337	39.7%	0.284	0.705	1290	780	1462	3344
BEVDepth + CV	ResNet-50	0.325	1.276	42.8%	0.300	0.710	903	907	1843	3299
BEVDepth + SimpleTrack	ResNet-50	0.338	1.294	43.9%	0.304	0.742	950	904	1798	3213
BEVDepth + Ours	ResNet-50	0.347	1.347	42.6%	0.309	0.722	944	1106	1758	3137
BEVDepth + KF	ResNet-101	0.301	1.345	40.2%	0.287	0.685	1444	841	1591	3156
BEVDepth + CV	ResNet-101	0.323	1.282	42.1%	0.299	0.696	807	885	2359	3256
BEVDepth + SimpleTrack	ResNet-101	0.333	1.302	42.4%	0.303	0.701	887	904	1835	3174
BEVDepth + Ours	ResNet-101	0.339	1.349	42.8%	0.309	0.691	1100	1187	1956	2890

Table 2: Comparison with different post-processing trackers on nuScenes *val* dataset. We report the tracking results with two different backbones, and the resolution of the input image is 256 × 704.

4 Experiments

4.1 Datasets and Metrics

Datasets. We mainly evaluate our QTrack on the 3D detection and tracking datasets of nuScenes. nuScenes dataset is a large-scale autonomous driving benchmark that consists of 1000 real-world sequences, 700 sequences for training, 150 for validation, and 150 for the test. Each sequence has roughly 40 keyframes, which are annotated by each sensor (e.g., LiDAR, Radar, and Camera) with a sampling rate of 2 FPS. Each frame includes images from six cameras with a full 360-degree field of view. For the detection task, there are 1.4 M annotated 3D bounding boxes from 10 categories. For the tracking task, it provides 3D tracking bounding boxes from 7 categories.

Metrics. For 3D detection task, we report nuScenes Detection Score (NDS), mean Average Prediction (mAP), as well as five True Positive (TP) metrics including mean Average Translation Error (mATE), mean Average Scale Error (mASE), mean Average Orientation Error (mAOE), mean Average Velocity Error (mAVE), mean Average Attribute Error (mAAE). For 3D tracking task, we report Average Multi-object Tracking Accuracy (AMOTA) and Average Multi-object Tracking Precision (AMOTP). We also report metrics used in 2D tracking task from CLEAR Bernardin et al. (2006), e.g., MOTA, MOTP, and IDS.

4.2 Implementation Details

Following BEVdepth, we adopt three types of backbone: ResNet-50 He et al. (2016), ResNet-101, and VoVNet-99 (Initialized from DD3D Park et al. (2021)) as the image backbone. If not specified, the image size is processed to 256×704. The data augmentation includes random cropping, random scaling, random flipping, and random rotation. In addition, we also adopt BEV data augmentations including random scaling, random flipping, and random rotation. We use AdamW as optimizer with learning rate of 2×10^{-4} and batch size of 64. When compared with other methods, QTrack is trained for 24 epochs for ResNet and 20 epochs for VoVNet with CBGS Zhu et al. (2019).

4.3 Comparision with Preceding SOTAs

Test and validation set. We compare the performance of QTrack with preceding SOTA methods on the nuScenes benchmark. The results are reported in Tab. 1. Our QTrack outperforms all current SOTA methods for the camera-based trackers by a large margin. For both validation and test sets, all reported metrics (e.g., AMOTA, AMOTP, RECALL, IDS, etc.) achieve best performance. Specially, AMOTA result of QTrack first achieves 0.511, which significantly reduces the performance gap between the pure camera and LiDAR-based trackers.

Compare with other post-processing trackers. Tab. 2 illustrates that QTrack outperforms the naive Kalman filter based method and its advanced variant from SimpleTrack Pang et al. (2021) by employing identical 3D detector and backbone settings. Moreover, our method only needs simple operations (i.e., Matrix multiplication and addition) for tracking procedure, while Kalman filter based ones need relatively complex operation like matrix transpose and the complex process for adjusting hyper-parameters. The overall tracking framework is significantly efficient and will not trigger a serious latency, which is fatal in a real perception scenario Yang et al. (2022a;b).

MF	VQ	LQ	AMOTA↑	AMOTP↓	MOTA↑	IDS↓
✗			40.4	1.266	36.9	1575
✗	✓		40.1	1.269	36.6	1680
✗		✓	40.5	1.264	**37.3**	**1445**
✗	✓	✓	**40.8**	**1.259**	37.0	1527
✓			42.2	1.236	38.1	1125
✓	✓		41.9	1.239	38.0	**999**
✓		✓	42.2	1.235	38.2	1023
✓	✓	✓	**42.6**	**1.228**	**38.3**	1076

Table 3: Ablation study of how to use velocity quality (VQ) and location quality (LQ). MF indicates using multiple frames.

4.4 Ablation Study

In this subsection, we verify the effectiveness of the proposed strategies separately through ablation studies. All the experiments are conducted on the nuScenes *val* set.

Analysis of the location and velocity quality for tracking. In this part, we conduct an in-depth analysis on the location and velocity quality score for the association process. As mentioned before, location and velocity quality scores are obtained by the quality branch. Then they are both regarded as the reference clues to filter the low classification confidence association results in QOA. We verify the performance of only using one of them as the second gate of QOA, and the results are reported in Tab. 3. As shown, only using one of the location and velocity quality scores does not contribute to the tracking performance, which confirms our analysis that the location and velocity quality is not aligned and we should take both of them into consideration.

Backbone	MF	CLS	Q.	AMOTA↑	AMOTP↓	MOTA↑	IDS↓
R50	✗			29.1	1.314	26.7	1488
R50	✗	✓		30.7	1.394	28.3	1748
R50	✗	✓	✓	31.3	1.390	28.5	1559
R50	✓			32.5	1.276	30.0	903
R50	✓	✓		34.1	1.348	30.5	1141
R50	✓	✓	✓	34.7	1.347	30.9	944
R101	✗			29.1	1.314	26.7	1488
R101	✗	✓		31.2	1.389	28.4	1622
R101	✗	✓	✓	31.8	1.386	29.1	1638
R101	✓			32.3	1.282	30.9	1100
R101	✓	✓		33.2	1.352	30.3	1053
R101	✓	✓	✓	33.9	1.349	30.9	1100
V99	✗			38.8	1.220	35.3	1670
V99	✗	✓		40.4	1.266	36.9	1575
V99	✗	✓	✓	40.8	1.259	37.0	1527
V99	✓			41.7	1.177	37.3	914
V99	✓	✓		42.2	1.236	38.1	1125
V99	✓	✓	✓	42.6	1.228	38.3	1076

Table 4: Ablation study of the components in QTrack. CLS indicates the first gate classification score while Q. indicates the second gate, i.e., quality score. MF indicates using multiple frames.

Analysis of the components of QTrack. In this part, we verify the effectiveness of various components in QTrack through an ablation study. As shown in Tab. 4, the first row of the table shows baseline performance for tracking when using BEVDepth detections followed by a simple velocity association step (CV method). We can observe that the two gates of QOA can both develop the tracking performance in the all settings (ResNet-50, ResNet-101 or VoVNet-99, single-frame or multi-frame), which means that the filter for the low-quality association results is necessary. Furthermore, we can observe that the metric of IDS increases when applying the first gate by classification confidence score. This phenomenon shows that only considering confidence score inevitably introduces low-quality bounding boxes, which causes bad association cases. Therefore, the second gate, quality score, can provide a fine-grained reference to achieve a better association trade-off.

Extra Branch	mAP↑	NDS↑	CV	SimpleTrack
None	0.3579	0.4826	0.326	0.337
Appearance Relative Drop	0.3522 -0.57%	0.4798 -0.38%	0.315 -1.1%	0.328 -0.9%
Quality Relative Drop	0.3585 **+0.06%**	0.4831 **+0.05%**	0.325 -0.1%	0.338 **+0.1%**

Table 5: Influence of extra branch on performance and tracking detection. For tracking performance of CV and SimpleTrack, we report the AMOTA metric for comparison.

Influence on base 3D detector. As shown in Tab. 5, it proves that adding quality prediction branch does not affect the performance of base 3D detector. This is an extremely important property since post-processing trackers normally rely on the super performance of detector. Going one step further, we report the tracking performance by employing existing CV and SimpleTrack scheme. It reveals that tracking performance will not be affected by our quality branch, which agrees with our designing purpose of Sec. 1. Then, we explore to append an appearance branch for extracting instance wised appearance embedding, which implement is the same as Zhang et al. (2021). The results show that slight performance degradation (nearly 0.5%) is triggered on detection task, but it significantly damages the performance of tracking task by nearly 1.0%. It reflects that our method is more effective and efficient.

4.5 Discussion and Future Work

Inspired by Jiang et al. (2018); Wu et al. (2020); Yang et al. (2022d), we explore to incorporate velocity quality V with classification score C as M, which is adopted to act as threshold metric in NMS procedure. Technically, we formulate M in Eq. 6, in which α is a hyper-parameter to control the contribution of V.

$$M = V^{(1-\alpha)} \cdot C^{\alpha}, \tag{6}$$

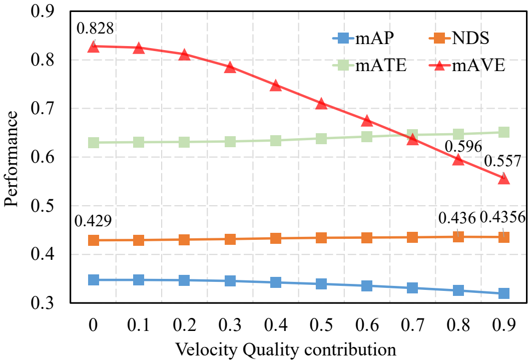


Figure 4: Influence of velocity quality on detection performance in NMS procedure. The abscissa indicates α in Eq. 6.

As shown in Fig. 4, we plot the four performance metrics of detection task by controlling α. It reflects that as contribution of V becomes bigger, mAVE drops dramatically. However, it also brings about inevitable performance degradation for mAP and mATE metrics. NDS, as a comprehensive metric, becomes better and then gets worse as α changes larger, which is actually a trade-off between location error and velocity error. This phenomenon agrees with our viewpoint in Sec. 1, i.e., the quality of these two prediction tasks are not aligned. Combining the performance of detection and tracking tasks with respect to above imbalance issue, it exposes a challenge: *how to design a method to simultaneously predict location (or 3D box) and velocity well?* This challenge can help further boost performance of 3D detection task or other downstream tasks like 3D MOT.

5 Conclusion

In this paper, we analyze the imbalance prediction quality distribution of location and velocity. It motivates us to propose a Quality-aware Object Association (QOA) method to alleviate the imbalance issue for 3D multi-object tracking (3D MOT). To this end, we introduce Normalized Gaussian Quality (NGQ) metric to measure the predicted quality of location and velocity, and structure an effective module for quality learning. Afterwards, we further present QTrack, an "tracking by detection" framework for 3D MOT in multi-view camera scene, which incorporats with QOA to perform tracking procedure. The extensive experiments demonstrate the efficacy and robustness of our method. Finally, we release a challenge to inspire more research to focus on the imbalance between localization and velocity qualities for both 3D detection and tracking tasks.

REFERENCES

Keni Bernardin, Alexander Elbs, and Rainer Stiefelhagen. Multiple object tracking performance metrics and evaluation in a smart room environment. In *Sixth IEEE International Workshop on Visual Surveillance, in conjunction with ECCV*, volume 90. Citeseer, 2006.

Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11621–11631, 2020.

Mohamed Chaabane, Peter Zhang, J Ross Beveridge, and Stephen O'Hara. Deft: Detection embeddings for tracking. *arXiv preprint arXiv:2102.02267*, 2021.

Shaoyu Chen, Xinggang Wang, Tianheng Cheng, Qian Zhang, Chang Huang, and Wenyu Liu. Polar parametrization for vision-based surround-view 3d detection. *arXiv preprint arXiv:2206.10965*, 2022.

Shoudong Han, Piao Huang, Hongwei Wang, En Yu, Donghaisheng Liu, and Xiaofeng Pan. Mat: Motion-aware multi-object tracking. *Neurocomputing*, 476:75–86, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hou-Ning Hu, Yung-Hsu Yang, Tobias Fischer, Trevor Darrell, Fisher Yu, and Min Sun. Monocular quasi-dense 3d object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

Junjie Huang, Guan Huang, Zheng Zhu, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021.

Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask scoring r-cnn. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6409–6418, 2019.

Borui Jiang, Ruixuan Luo, Jiayuan Mao, Tete Xiao, and Yuning Jiang. Acquisition of localization confidence for accurate object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 784–799, 2018.

Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering 82(Series D)*, pp. 35–45, 1960.

Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12697–12705, 2019.

Peixuan Li and Jieyu Jin. Time3d: End-to-end joint monocular 3d object detection and tracking for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3885–3894, 2022.

Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. *arXiv preprint arXiv:2206.10092*, 2022a.

Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. *arXiv preprint arXiv:2203.17270*, 2022b.

Zhuoling Li, Zhan Qu, Yang Zhou, Jianzhuang Liu, Haoqian Wang, and Lihui Jiang. Diversity matters: Fully exploiting depth clues for reliable monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2791–2800, 2022c.

Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. *arXiv preprint arXiv:2203.05625*, 2022.

Nicola Marinello, Marc Proesmans, and Luc Van Gool. Triplettrack: 3d object tracking using triplet embeddings and lstm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4500–4510, 2022.

Ziqi Pang, Zhichao Li, and Naiyan Wang. Simpletrack: Understanding and rethinking 3d multi-object tracking. *arXiv preprint arXiv:2111.09621*, 2021.

Dennis Park, Rares Ambrus, Vitor Guizilini, Jie Li, and Adrien Gaidon. Is pseudo-lidar needed for monocular 3d object detection? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3142–3152, 2021.

Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 770–779, 2019.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10529–10538, 2020.

Yining Shi, Jingyan Shen, Yifan Sun, Yunlong Wang, Jiaxin Li, Shiqi Sun, Kun Jiang, and Diange Yang. Srcn3d: Sparse r-cnn 3d surround-view camera object detection and tracking for autonomous driving. *arXiv preprint arXiv:2206.14451*, 2022.

Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9627–9636, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 913–922, 2021.

Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *Conference on Robot Learning*, pp. 180–191. PMLR, 2022.

Xinshuo Weng, Jianren Wang, David Held, and Kris Kitani. 3d multi-object tracking: A baseline and new evaluation metrics. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 10359–10366. IEEE, 2020a.

Xinshuo Weng, Yongxin Wang, Yunze Man, and Kris M Kitani. Gnn3dmot: Graph neural network for 3d multi-object tracking with 2d-3d multi-feature learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6499–6508, 2020b.

Shengkai Wu, Xiaoping Li, and Xinggang Wang. Iou-aware single-stage object detector for accurate localization. *Image and Vision Computing*, 97:103911, 2020.

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

Jinrong Yang, Songtao Liu, Zeming Li, Xiaoping Li, and Jian Sun. Real-time object detection for streaming perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5385–5395, 2022a.

Jinrong Yang, Songtao Liu, Zeming Li, Xiaoping Li, and Jian Sun. Streamyolo: Real-time object detection for streaming perception. *arXiv preprint arXiv:2207.10433*, 2022b.

Jinrong Yang, Lin Song, Songtao Liu, Zeming Li, Xiaoping Li, Hongbin Sun, Jian Sun, and Nanning Zheng. Dbq-ssd: Dynamic ball query for efficient 3d object detection. *arXiv preprint arXiv:2207.10909*, 2022c.

Jinrong Yang, Shengkai Wu, Lijun Gou, Hangcheng Yu, Chenxi Lin, Jiazhuo Wang, Pan Wang, Minxuan Li, and Xiaoping Li. Scd: A stacked carton dataset for detection and segmentation. *Sensors*, 22(10):3617, 2022d.

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*, pp. 11784–11793, 2021.

En Yu, Zhuoling Li, and Shoudong Han. Towards discriminative representation: Multi-view trajectory contrastive learning for online multi-object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8834–8843, 2022a.

En Yu, Zhuoling Li, Shoudong Han, and Hongwei Wang. Relationtrack: Relation-aware multiple object tracking with decoupled representation. *IEEE Transactions on Multimedia*, 2022b.

Tianyuan Zhang, Xuanyao Chen, Yue Wang, Yilun Wang, and Hang Zhao. Mutr3d: A multi-camera tracking framework via 3d-to-2d queries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4537–4546, 2022a.

Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *International Journal of Computer Vision*, 129(11):3069–3087, 2021.

Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022b.

Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Tracking objects as points. In *European Conference on Computer Vision*, pp. 474–490. Springer, 2020.

Benjin Zhu, Zhengkai Jiang, Xiangxin Zhou, Zeming Li, and Gang Yu. Class-balanced grouping and sampling for point cloud 3d object detection. *arXiv preprint arXiv:1908.09492*, 2019.