IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 25 December 2023, accepted 15 January 2024, date of publication 22 January 2024, date of current version 13 February 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3356864

RESEARCH ARTICLE

CollabMOT Stereo Camera Collaborative Multi Object Tracking

PHONG PHU NINH, (Graduate Student Member, IEEE), AND HYUNGWON KIM
Department of Electronic Engineering, Chungbuk National University, Cheongju-si 28644, Republic of Korea

Corresponding author: Hyungwon Kim (hwkim@cbnu.ac.kr)

This work was supported in part by the National Research Foundation of Korea (NRF) for RLRC funded by the Korean Government [Ministry of Science and ICT (MSIT)], South Korea, under Grant 2022R1A5A8026986; in part by the Institute of Information & Communications Technology Planning & Evaluation (IITP) funded by the Korean Government (MSIT) through the Development of Self-Learnable Mobile Recursive Neural Network Processor Technology under Grant 2020-0-01304; and in part by MSIT under the Grand Information Technology Research Center Support Program through the Grand-ICT supervised by IITP under Grant IITP-2023-2020-0-01462.

ABSTRACT The recent advances in deep learning techniques enable 2D Multi-object tracking (MOT) to achieve remarkable performance over traditional methods. However, most 2D MOT algorithms primarily utilize only single-camera view. Therefore, they are prone to frequent tracking losses and track-ID switching under conditions due to limited viewpoints and occluded objects. To alleviate this problem, we propose a stereo-camera-based collaborated multi-object tracking (CollabMOT) method that performs online and dynamic association of multiple tracklets from baseline MOT algorithms in overlapping views of stereo cameras. CollabMOT utilizes appearance similarity to generate global tracking IDs that unify the same tracklets between viewpoints of stereo cameras. It then leverages the transitive information from these global tracking IDs to reconnect the disrupted tracklets in each camera view. CollabMOT improves the overall performance of baseline 2D MOT methods on a single camera view by mitigating the problem of ID switching. Evaluation of CollabMOT on Argoverse-HD and KITTI dataset shows improved performance over baseline MOT methods. As a result, the proposed method improves the performance of the recent state-of-the-art method on the 2D MOT task of the KITTI dataset from 79.5 to 80% on High Order Tracking Accuracy (HOTA) score for vehicles.

INDEX TERMS Multi-object tracking, stereo vision, deep learning, data association.

I. INTRODUCTION

Multi-object tracking (MOT) is the process of localizing objects and connecting them to their unique identities within a video. MOT is one of the main research areas in computer vision, with applications in surveillance, autonomous driving, and other areas. Most of the 2D MOT algorithms [1], [2], [3], [4], [5], [6], [7], [8], [9], to the best of our knowledge, only focus on a single camera with a constrained field of view. They are, therefore, vulnerable to the occlusion problem, where the algorithm becomes confused and typically gives a new identity to a tracked object that reappears, resulting in a high ratio of identity switching.

Consider the video sequence showing the left and right viewpoints of the stereo camera from KITTI dataset [10] in Figure 1. On the left camera, a vehicle with ID 26 is occluded by the automatic traffic barrier after two frames and incorrectly switches to ID 30 in the later frame. On the other hand, the right camera has an unobstructed view of the identical vehicle and consistently keeps the same ID 22. Because of their similar appearance, ID 22 from the right camera is linked to ID 26 in the previous frame and ID 30 in the current frame on the left camera. By transitive reasoning, ID 26 and 30 are an identical tracklet. The same observation can be applied to ID 25 and 21 from the right camera, which are interconnected to ID 23 from the left camera. Suppose the left camera is the primary camera in the system, the right camera could become the assistant to combat the track ID switching problem of the left camera and vice versa.

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Asif.

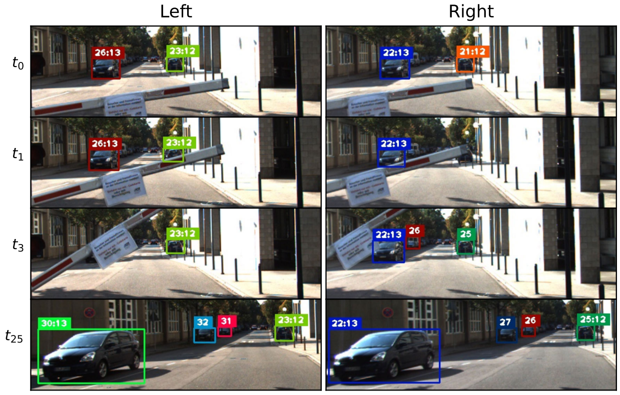


FIGURE 1. Left and right video frames from the KITTI dataset with outputs of CenterTrack [1] (number before colon sign). By utilizing the transitive reasoning between tracklets in overlapped stereo cameras, CollabMOT unites interrupted tracklets under occlusion with consistent ID (number after colon sign).

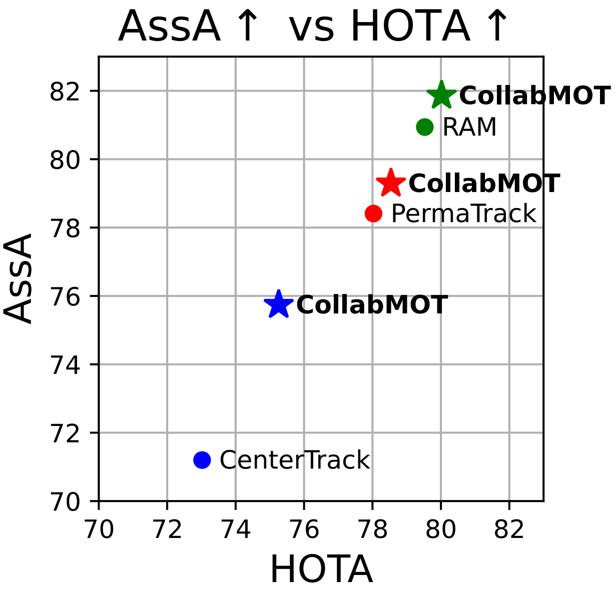


FIGURE 2. AssA (Association Accuracy)-HOTA (Higher Order Tracking Accuracy) graph comparing our proposed CollabMOT with the latest state-of-the-art trackers of the KITTI test datasets on HOTA metric [18]. The baseline tracker results are represented in circle ◦, and our proposed CollabMOT results are embodied in star ⋆ mark sharing the same color with the baseline method. The performance of baseline methods and CollabMOT is evaluated on the left camera with DeepSORT [7] is employed in the right camera. Our proposed method shows improved HOTA and AssA performance over baseline methods. Best viewed in color.

For vehicle tracking in autonomous driving or advanced driving assistance system (ADAS) applications, which is one of the main applications of MOT, cameras are mounted directly on the vehicles [10], [11], [12], [13]. Most ADAS or autonomous vehicles are equipped with multiple cameras such as those from [11] and [12]. They provide 360-degree coverage around the car, and each camera typically has a very narrow overlapping field with the others. Therefore, 3D multi-camera multi-object tracking [14], [15] spatio-temporally monitors the 3D states of tracklets across cameras. However, a narrow overlapping multi-camera setup may prove to be less effective in the event of an occluded object in a single-camera view. (Figure 1).

A common example of highly overlapped multi-cameras is a stereo camera [10], [13] as illustrated in Figure 1. Much of the work on stereo vision has been aimed at depth estimation and 3D object detection, which requires heavy calibration and complex matching algorithm [16]. Inspired by the observation from Figure 1, the 2D Multi-Object Tracking task could also benefit from the overlapping view of a stereo camera.

Most recent trackers utilize deep neural networks to learn all necessary cues to associate tracklets between frames over time. While avoiding heuristic assumptions commonly found in handcrafted appearance and motion cues, deep-learning-based tracking methods still heavily rely on well-curated and high-quality datasets. To achieve optimal performance, most methods require pre-training on large datasets before fine-tuning on the target test datasets instead of solely relying on the provided training dataset. Though such methods have the potential to become more generalized trackers, they still require significant amounts of pre-processing and complex training schemes to function accurately.

In this work, we introduce CollabMOT, which can be integrated into the existing state-of-the-art MOT tracker algorithms. Based on their appearance feature, it performs inter-camera tracklets association between the two partially overlapping views from stereo cameras. Furthermore, we utilize the inter-camera connection between tracklets to reconnect interrupted tracklets in a single-camera view.

We demonstrate the transitive relation of tracklets association in overlapping multi-cameras improves the performance of 2D multi-object tracking in each camera **without additional multi-camera annotations**. CollabMOT is extensively evaluated on KITTI [10] and Argoverse-HD [17] datasets, demonstrating improved performance over baseline MOT methods. In particular, CollabMOT improves the HOTA score from 73.02% to 75.26%, 78.03% to 78.54%, and from 79.53% to 80.02% for three state-of-the-art MOT algorithms: CenterTrack [1], PermaTrack [2], RAM [3], respectively, on the highly competitive KITTI multi-object tracking dataset as demonstrated in Figure 2. Furthermore, the proposed CollabMOT algorithm is designed to be adaptable to various appearance feature encoder profiles. In addition, it is capable of operating in a semi-online manner. As such, it represents a viable solution for real-time applications.

The rest of the paper is organized in the following way. In section II, we explore related state-of-the-art approaches and analyze their distinctive features by categorizing them into distinct classes. Then, we explain the details of our proposed CollabMOT in Section III. Next, we provide detailed experiments conducted on KITTI [10] and Argoverse-HD [17] in Section IV. Finally, we discuss the limitation and future work of CollabMOT in Section V and Section VI.

II. RELATED WORKS

A. SINGLE CAMERA 2D MULTI OBJECT TRACKING

In recent years, with the development of stronger object detection [19], [20], [21], [22], [23], tracking by detection (TBD) is widely studied and considered as one of the most effective paradigms for 2D multi-object tracking (MOT). In tracking by detection, the first step is to find objects in each image and associate them over time with similarity cues such as motion, location, and appearance. SORT [6] computes the IOU between detected boxes and predicted boxes generated by Kalman Filter [24] with a constant velocity model. DeepSORT [7] improved SORT by employing a separate appearance encoder to extract discrimination features from detected boxes as an association metric combined with IOU. Because of its modular design, TBD is compatible with various object detectors and appearance feature encoders, making it very helpful for annotating objects [25]. However, they are usually computationally expensive. Recently, several studies [4], [5] ignore appearance information and only focus on associating the bounding box generated from high-performance object detection models based on location similarity or IOU score. For pedestrian tracking, location similarity is generally satisfactory, but it is typically inferior to appearance similarity on high-velocity targets or low frame rate data [5].

Recently, several studies [1], [8], [9], [26] extended the existing object detectors into trackers and executed both tasks in a single framework. In CenterTrack [1], the author extends the anchor-less detector CenterNet [27] framework, aiming to predict the displacements of tracklets between consecutive frames. CenterTrack tends to produce short, fragmented tracklets and generally assigns long-lost, reappearing tracklets with a new identity. Based on CenterTrack, PermaTrack [3] added ConvGRU [28] as a sub-network to accumulate tracklets information in previous frames. PermaTrack learns to detect objects under occlusion by combining synthetic datasets with real datasets and heuristic label preprocessing. RAM [3], built on top of PermaTrack, additionally combines contrastive random walk [29] process to enhance further the capability to retrack the object under heavy occlusion.

B. SEMI-ONLINE MULTI-OBJECT TRACKING

According to the processing scheme, MOT algorithms could be further categorized into online, offline, and semi-online methods. Online MOT methods [1], [2], [3], [7] generate tracklets in every frame time, while offline methods [30], [31] wait for whole images sequence. On the other hand, semi-online MOT [32], [33] combines the strength of both online and offline methods as they process images on a frame-by-frame basis similar to online algorithms but correct the wrong association in the previous frames like offline methods. In this paper, we apply tracklets association on the output of online MOT algorithms to modify the wrong tracklets output in the past frames. Consequently, CollabMOT belongs to the semi-online MOT approach.

TABLE 1. Notation used throughout the paper.

Symbol	Description	Scope
I	Image frames	Local
$\mathcal{B}$	Bounding box bank	Local
b	Bounding box	Local
$\mathcal{F}$	Appearance Feature bank	Local
f	Appearance feature	Local
T	Tracklets	Local
t	Association frame	Local & Global
$\mathcal{S}$	Affinity cost	Global
$min\mathcal{S}$	Global minimum affinity cost	Global
$tmin\mathcal{S}$	Temporary minimum affinity cost	Global
$\mathcal{G}$	Global ID	Global
$t\mathcal{G}$	Temporary global ID	Global
e	Ego camera	Global
r	Remote camera	Global

C. MULTI-TARGET MULTI-CAMERA TRACKING

Most of the algorithms in Multi-target Multi-Camera Tracking (MTMCT) are designed for fixed surveillance camera systems. The performances are evaluated by aggregated global tracklets information from each local annotation in each camera view. NVIDIA has hosted the AI City challenges [34], [35], [36], [37], [38], [39] providing non-overlapping traffic camera datasets for multiple traffic analysis task such as city scale multi-target multi-camera tracking, cross camera vehicle reidentification, and more. On the other hand, highly overlapping pedestrian surveillance datasets [40], [41], [42] utilize calibration process of static cameras to map the target from image coordinates to the real-world coordinates among all cameras and evaluates the performance in projected coordinates on ground rather than the image space.

The standard workflow in MTMCT task is divided into two steps [43]: 1) Single camera Multi-Object Tracking and 2) Inter-Camera tracklets association. The first step involves generating local tracklets for each camera using a multi-object tracking algorithm. As the local tracklets are ingredients in the second step, local tracklets include target coordinates and a corresponding feature vector extracted by an independent feature encoder. In the next step, local tracklets and topology of each camera in the network are gathered and associated by their feature vector and spatiotemporal relation in the camera network into global tracklets.

Our proposed method adheres to the fundamental principles of MTMCT while introducing a distinctive processing pipeline, setting it apart from the conventional approaches. Most of the previous MTMCT methods perform offline association, which means that the Inter-Camera tracklets association conducts association after gathering all local tracklets. Our method targets real-time stereo camera applications where future frames are unavailable. We perform inter-camera tracklet association immediately after generating local tracklets in each camera. Additionally, we use positive feedback from multi-camera tracklet associations to resolve identity inconsistencies in each camera's field of view.

III. PROPOSED COLLABMOT

Table 1 summarizes the notation used in this paper. Since we work with global tracklet associations on multiple cameras

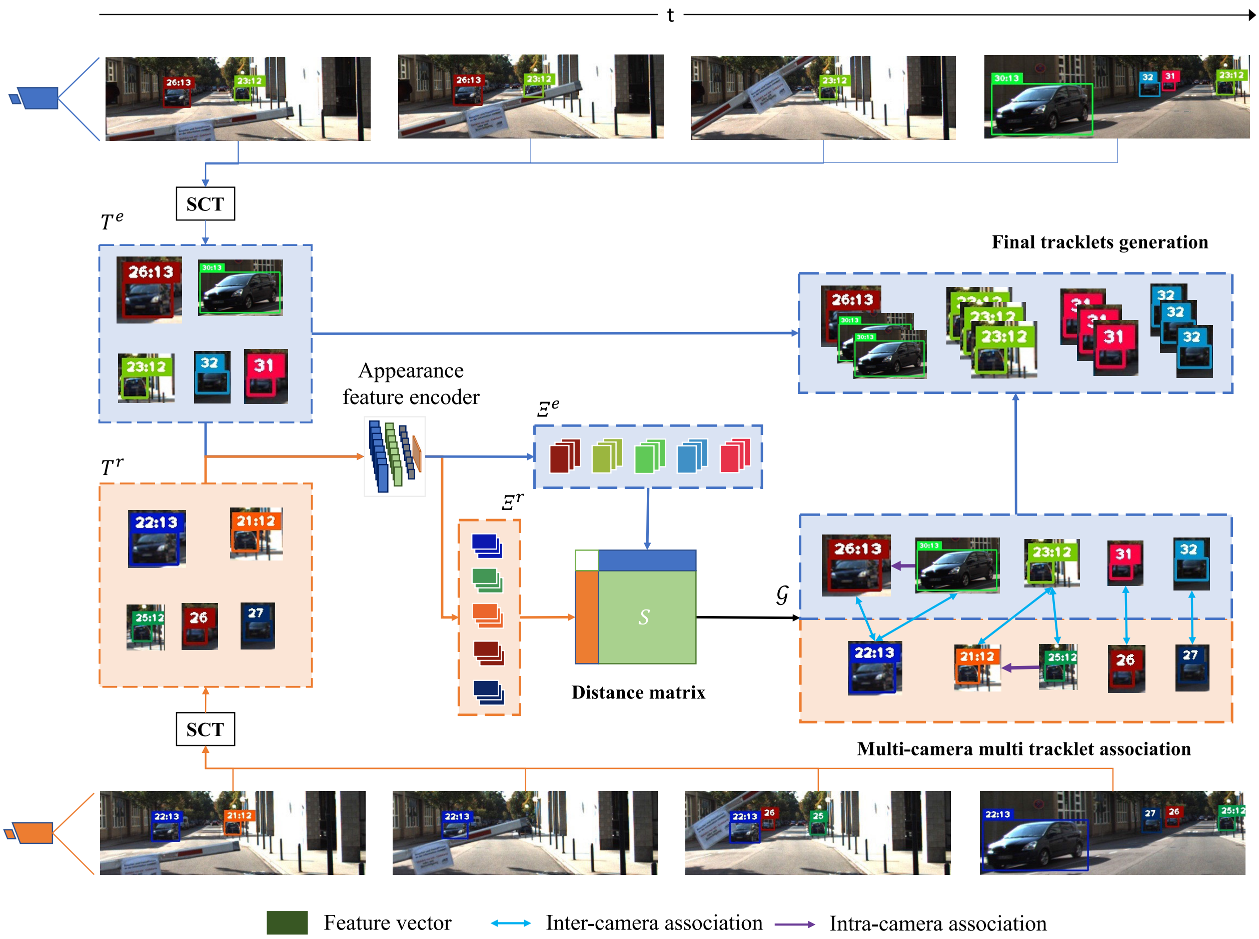


FIGURE 3. The proposed CollabMOT method consists of four major modules: 1) Single camera object tracking (SCT), 2) Appearance feature extraction, 3) Affinity cost calculation, and finally, 4) Inter-camera multi-tracklets association and intra-camera tracklets reconnection. Finally, we generate the uninterrupted tracklets combining their bounding box location in the ego camera with the global identity from the inter-camera association step. Best view in color and zoom in.

and local tracklets on each camera, we use superscripts to denote global scope and subscripts to denote local scope. For example, T_i^c is a tracklet with track id i of camera c.

Figure 3 illustrates the overall structure of the proposed method. At each time t, the Single-Camera Multi Object Tracking (SCT) module takes the synchronized input image I_t^c from camera c and generates tracklets T^c with each tracklet $T_a \in T^c$ consisting of bounding box locations $\mathcal{B} = b_i, i \in [1, t]$.

In the next step, we use an appearance feature encoder trained on vehicle reidentification dataset to generate appearance feature vector f from each bounding box $b \in \mathcal{B}$. The feature vector f forms appearance feature bank $\mathcal{F} = f_i, i \in [1, t]$, which is added to the tracklet $T_a = \{(b_i, f_i), i \in [1, t]\}$ along with $\mathcal{B}$.

Our method labels a camera that requests feedback from the multi-camera association system as an "ego camera". This terminology refers to the camera's role and relationship with the system, aiding in the comprehension of its functionality. At each ego camera, we calculate the affinity cost distance between its own tracklets and other tracklets from remote cameras by using their appearance feature bank $\mathcal{F}$ in Section III-C. Next, we determine bipartite matching between tracklets from two cameras until frame t based on the affinity cost distance in Section III-D.

The global identity of tracklets, denoted by $\mathcal{G}$, represents the unique identity of a target object across all cameras in the system. It is determined by considering both the intra-camera and inter-camera tracklet relations [43]. Intra-camera tracklet relations refer to the temporal continuity of tracklets within a single camera, while inter-camera tracklet relations refer to the spatial and temporal associations between tracklets across different cameras. By considering both types of relations, $\mathcal{G}$ provides a comprehensive representation of the identity of tracklets in a multi-camera system. Here, tracklet T_a^c from camera c is identical to tracklet $T_{a'}^{c'}$ from camera c' if $\mathcal{G}_a > 0$ and $\mathcal{G}_a = \mathcal{G}_{a'}$ with a, a' is defined in Equation 1.

$$a, a' = \begin{cases} a \in T^c, a' \in T^{c'} & \text{if } c = c' \text{ and } a \neq a' \\ a \in T^c, a' \in T^{c'} & \text{if } c \neq c' \end{cases} \qquad (1)$$

The global ID connects the ongoing tracklets between two cameras and joins the stale tracklet with the new one to form a longer tracklet in each camera by utilizing the transitive relation between tracklets. Furthermore, we interpolate missing bounding boxes of interrupted tracklet and only select unobstructed bounding boxes to complete the trajectory. Finally, we generate the final tracklets on the ego camera using the combined information from both inter-camera and intra-camera tracklet association.

A. SINGLE CAMERA TRACKING

Our proposed method provides the tracklet association over multiple cameras so it can integrate various single-camera trackers (SCTs) to produce global association over tracklets from the SCTs of the multiple cameras. Furthermore, each camera could use a different SCT algorithm to perform the task of single camera tracking if the tracking targets are similar among cameras.

B. APPEARANCE FEATURE EXTRACTOR

Data association in MOT decides the allocation of new detection boxes to tracklets based on their similarity. However, each method adopts different cues to compute the similarity in terms of the location of tracklets [1], [4], [6], and appearance information [7], [8], [9]. Due to the diversity of similarity computation techniques of MOT methods, we employ separate appearance feature encoders for all SCTs integrated into CollabMOT. Although CollabMOT does not impose any restrictions on the use of the SCT algorithm on each camera, it uses the same appearance feature encoder to ensure that the extracted feature information is comparable among tracklets. For track-by-detection SCT, which adopts a separate appearance encoder to extract feature information from detection boxes [7], we use the same appearance feature encoder for both SCT and CollabMOT. We provide more details of the appearance feature encoder in Section IV-C2.

C. DISTANCE MATRIX CALCULATION

We use Ξ^e and Ξ^r to denote, respectively, the list of tracklets from T^e and T^r whose bounding box emerge at frame t as the candidates for distance calculation at each association time t. As a small bounding box does not capture beneficial image feature data, we exclude tracklets whose bounding box height is smaller than λ_h.

Let $S_{i,j}$ denote, respectively, the affinity cost of tracklet $T_i^e \in \Xi^e$ and tracklet $T_j^r \in \Xi^r$. The total affinity cost is calculated by Equation 2:

$$S_{i,j} = Sa \times \omega_a + Sp \times \omega_p \tag{2}$$

Here, Sa and Sp represent the appearance and position similarity cost, respectively, while ω_a, ω_p are weight values to balance Sa and Sp.

We follow the exponential moving average (EMA) update of a new appearance feature f_t from [9] in Equation 3:

$$\tilde{f} = \alpha \tilde{f}_{t-1} + (1 - \alpha) f_t \tag{3}$$

Here, α is the decay rate of the momentum update. We set $\alpha = 0.9$ in the experiment presented in this work. The EMA updates the appearance cost incrementally and reduces the computation burden for tracklets with a long history.

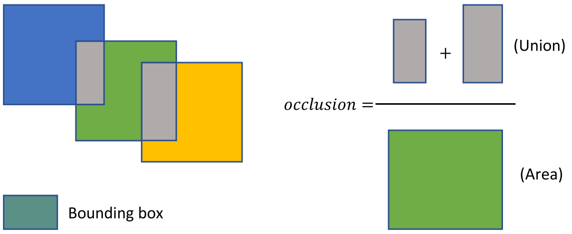


FIGURE 4. To compute the *occlusion* ratio of the bounding box, we divide the area of the union of overlapping regions (gray color) over the area of the target bounding box.

Bounding boxes with lower confidence scores tend to show lower accuracy in their localization information, leading to inaccurate appearance features. Due to the limited field of view of a single-mounted camera, bounding boxes of adjacent tracklets usually overlap with each other. To take such bounding boxes into account, we only update f_t into $\tilde{f}$ if the confidence score $conf \geq 0.3$ and $occlusion \leq \lambda_{occlusion}$. Here, we define *occlusion* ratio in Equation 4.

$$occlusion = \frac{\text{area of union of overlapping}}{\text{area of bounding box}} \tag{4}$$

Figure 4 represent the calculation process of *occlusion* in Equation 4.

For tracklets whose none of its bounding boxes satisfied these conditions until association time t, we have $\tilde{f} = \varnothing$. Finally, the appearance affinity cost is computed as follows Equation 5:

$$Sa = \begin{cases} \infty & \text{if } \tilde{f}^e = \varnothing \text{ or } \tilde{f}^r = \varnothing \\ \infty & \text{if } D_{CD}(\tilde{f}^e, \tilde{f}^r) > \lambda_{appearance} \\ D_{CD}(\tilde{f}^e, \tilde{f}^r), & \text{otherwise} \end{cases} \tag{5}$$

where D_{CD} is the cosine distance of $\tilde{f}^e, \tilde{f}^r$.

In a stereo camera setup, when camera e is on the left side of camera r, all tracklets on the left side in the viewpoint of e almost certainly do not emerge on the right side of r. The same observation can be applied to the tracklets on the right side of e if e is on the right side of r. Therefore, we set $Sp = \infty$ for any pair of T_i^e and T_j^r that fall into such cases to limit unnecessary distance calculation in Equation 2. Otherwise $Sp = 0$. More detailed calculation for Sp is provided in Appendix.

D. MUTI CAMERA TRACKLET ASSOCIATION

At each time t, an online SCT algorithm decides whether to start new tracklets, extend the existing tracklets with new detection boxes, or stop the tracklets which has not been associated with any detection box for a specified number of frames. Furthermore, it could revive the lost tracklets T_{lost} with current bounding boxes via cascaded matching [7]. In case the T_{lost} has been intra-connected with one of the ongoing tracklet $T_{current}$, we count the number of cases where the bounding boxes of T_{lost} and $T_{current}$ at each time t has IOU score greater than λ_{merged_iou}. If the number of cases

Algorithm 1 Multi-Camera Multi Tracklet Association

```
 1: Input : affinity cost matrix $S$
 2: Output : updated $\mathcal{G}$ and $minS$
 3: Handle local tracklet resurfacing.
 4: repeat
 5:     $\mathcal{A} \leftarrow$ Hungarian Algorithm [44] ($S$)
 6:     Assign $t\mathcal{G}$, $tmin\mathcal{S} = \emptyset$ # temporary empty list
 7:     for $(i, j) \in \mathcal{A}$, $S_{i,j} < \lambda_{appearance}$ do
 8:         $g_i, V_i \leftarrow$ Algorithm 2 ($T_i^e, T_j^r, T^r, S_{i,j}$)
 9:         $g_j, V_j \leftarrow$ Algorithm 2 ($T_j^r, T_i^e, T^e, S_{i,j}$)
10:         if either $V_i$ or $V_j$ is False then
11:             Set $S_{i,j} = \infty$,
12:             $\mathcal{A} \leftarrow \emptyset$ # Rerun, goto line 5
13:         end if
14:         $t\mathcal{G}_i, t\mathcal{G}_j \leftarrow$ Equation 6 ($g_i, g_j$)
15:         $tmin\mathcal{S}_i, tmin\mathcal{S}_j \leftarrow S_{i,j}$
16:     end for
17:     Update $minS \leftarrow tmin\mathcal{S}$
18:     Update $G \leftarrow t\mathcal{G}$
19:     # Finish multicamera tracklets association
20: until Reset is not triggered
```

where co-appearance is smaller than λ_{coapp}, we merge their co-appearance bounding box at each time step. Otherwise, we revoke the intra-association between T_{lost} and $T_{current}$ prior to tracklet association step. We also immediately cancel the intra-connection once the IOU score is greater than λ_{coapp} or there is more than one T_{lost} on the same camera resurfacing. In the work presented in this paper, we set the threshold $\lambda_{merged\_iou} = 0.3$ and $\lambda_{coapp} = 5$ for every experiment. CollabMOT performs local tracklet resurfacing on tracklets from all SCT before the tracklet association step.

Algorithm 2 Association Verification

```
 1: Input: $T_i^e$, $T_j^r$, $T^r$, affinity cost $S_{i,j}$
 2: Output: valid globalID $g_i$, valid check $V_i$
 3: Assign $g_i = -1$, $V_i = True$
 4: for tracklet $T_k \in T^r$, $\mathcal{G}_k > 0$, $\mathcal{G}_k = \mathcal{G}_i$ do
 5:     if $T_j^r = T_k$ then
 6:         $g_i = \mathcal{G}_k$
 7:     else
 8:         if $T_k$ and $T_j^r$ are temporal overlapped then
 9:             if $min\mathcal{S}_k < S_{i,j}$ then
10:                 $V_i = False$
11:             else
12:                 $t\mathcal{G}_i, t\mathcal{G}_k = -1$ : # reset previous global IDs
13:             end if
14:         else
15:             $g_i = \mathcal{G}_k$ # $T_k$ is lost tracklet
16:             Recover missing trajectory from $T_k$ to $T_j^r$
17:         end if
18:     end if
19: end for
```

Let's denote $\mathcal{A}$ as sets of optimal assignments between Ξ^e and Ξ^r after performing Hungarian assignment [44] on cost matrix S (Algorithm 1 line 5). Compared with offline MOT, online MOT algorithms have the disadvantage that the future frames of tracklets are unavailable at each association step. Therefore, each association $i, j \in \mathcal{A}$ represent the optimal assignment between tracklets $T_i^e \in T^r$ and $T_j^r \in T^r$ **only** at frame t. To rule out the possibility that current association i, j is not the optimal assignment compared with previous assignments in previous frames or vice versa, we proposed the dynamic association scheme that combines the current affinity cost S and the global affinity cost $minS$ which stores the minimum assignment cost of each tracklet.

We demonstrate the validation step of T_i^e and $T_j^r \in T^r$ in Algorithm 2. We iterate all the tracklet $T_k \in T^r$ which has been associated with T_i^e by same global ID, i.e $\mathcal{G}_k = \mathcal{G}_i$. We assign $gid = \mathcal{G}_k$ only when either T_k is indeed same as T_j^r (Algorithm 2 line 5) or T_k has been lost and there is no temporal overlap between T_k and T_j^r (Algorithm 2 line 15). Otherwise, if they coexist at any time, we then inspect the assignment of T_i^e, T_k by using the global affinity cost $minS$. If global affinity cost $minS_k$ is greater than $S_{i,j}$, it suggests that the previous assignment between T_k and T_i^e is not the optimal association. Hence, we revoke their connection by setting $t\mathcal{G}_i = -1$ and $t\mathcal{G}_k = -1$ (Algorithm 2 line 12). Conversely, it implies the current assignment of T_j^e and T_j^r are less favorable association than the previous pair T_i^e and T_k, so we need to invalidate the current association of i, j by setting $V = False$ (Algorithm 2 line 10).

Algorithm 2 only verifies the association of T_i^e and T_j^r on tracklets T^r. To ensure the optimal assignment of T_i^e and T_j^r on both ego and remote cameras, we additionally check the connection between T_j^r and T_i^e on tracklets T^e by using the same Algorithm 2. If the above mutual verification process detects non-optimal association (Algorithm 1 line 11, 12), in other words, either $V_i = False$ or $V_j = False$ (Algorithm 2, line 10), we invalidate i, j by setting $S_{i,j} = \infty$ and restart the procedure (Algorithm 1 line 5). Otherwise, we determine the final association identity based on g_i and g_j as described by Equation 6.

$$g_{i,j} = \begin{cases} g_i & \text{if } g_i > 0 \text{ and } g_j < 0 \\ g_j & \text{if } g_j > 0 \text{ and } g_i < 0 \\ g_i & \text{if } g_i = g_j \\ -1 & \text{if } g_i \neq g_j \\ \text{new} & \text{if } g_i < 0 \text{ and } g_j < 0 \end{cases} \tag{6}$$

To avoid updating suboptimal associations into $\mathcal{G}$ and $minS$, we designate the temporary $t\mathcal{G}$ and $tminS$ to hold the current association identity and cost of each i, j. We update $t\mathcal{G}$ to $\mathcal{G}$ and $tminS$ to $minS$ only if the reset step is not triggered.

The benefit of the above bi-directional association step is twofold. Firstly, it updates Global ID G with the optimal inter-camera association from the beginning until association time t. Second, it also reconnects the previously lost tracklets with the current ones on the single camera, increasing the SCT algorithm's performance.

E. RECOVER THE MISSING TRAJECTORY BY INTERPOLATION

Let T_i^e and T_k^e denote the current and lost tracklet on ego camera e, which has been reconnected by transitive relation

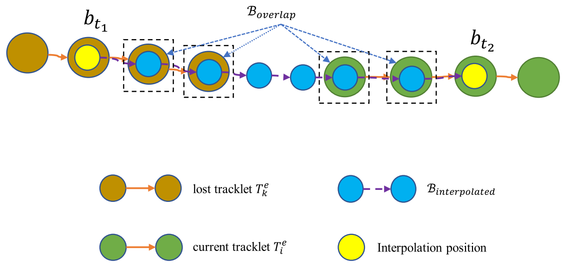


FIGURE 5. **Illustration of interpolation process between lost tracklet T_k^e and current tracklet T_i^e. We interpolate the missing bounding boxes between two bounding boxes in yellow color. We then check the IOU score of temporal overlapped interpolated and tracklets bounding box (shown in dashed rectangles) to ensure that the propagation of missing bounding boxes between two tracklets is accurate.**

with T_j^r on remote camera r. Then we have $\mathcal{G}_i = \mathcal{G}_j = \mathcal{G}_k$. Once we reconnect T_k^e with the T_i^e to create longer tracklets, we can interpolate the missing bounding box $\mathcal{B}_{interpolated}$ from the bounding boxes of T_k^e and T_i^e.

The linear interpolation process of $\mathcal{B}_{interpolated}$ between two bounding boxes b_{t_1} and b_{t_2} from frame t_1 to frame t_2 is described in Equation 7:

$$\mathcal{B}_{interpolated} = \{b_{t_1} + \frac{b_{t_2} - b_{t_1}}{t_2 - t_1}(t - t_1)\} \tag{7}$$

where $t_1 < t < t_2, t \in \mathbb{I}^1$ and $b \in \mathbb{R}^4$ containing the top left and bottom right coordinates of a bounding box.

Normally, missing interpolation boxes $\mathcal{B}_{interpolated}$ between T_k^e and T_i^e are derived from the latest frame t_{last} of T_k^e to the first frame t_{first} of T_i^e [4], [45]. However, when there is no direct linear interpolation between T_k^e and T_i^e, choosing the bounding box at t_{last} and t_{first} as interpolation points can introduce the false positive bounding boxes.

Therefore, we introduce $\omega_{interpolated}$, which indicates the depth of interpolation in terms of the number of frames. In Equation 7, we assign $t_1 = t_{last-\omega_{interpolated}}$ and $t_2 = t_{first+\omega_{interpolated}}$. We define $\mathcal{B}_{overlap}$ as temporal overlapping bounding boxes positioned in frames $[first, first + \omega_{interpolated})$ and $(last - \omega_{interpolated}, last]$ from $\mathcal{B}_k^e$ and $\mathcal{B}_i^r$, respectively. We only fill the gap if all of the IOU scores between $\mathcal{B}_{interpolated}$ and $\mathcal{B}_{overlap}$ are greater than λ_{int_iou}. Through this step, we want to ensure the interpolation process produces the correct propagation of missing boxes between two tracklets. Furthermore, we set $\lambda_{int\_occlusion}$ as the threshold to discard heavy occlusion box from $\mathcal{B}_{interpolated}$ by its *occlusion* ratio calculated from Equation 4. The above process is illustrated in Figure 5.

With $\omega_{interpolated} \geq 1$, the interpolation process should wait for $\omega_{interpolated}$ frame after the intra connection between T_k^e and T_i^e was formed. If the intra-connection between T_k^e and T_i^e is revoked, (Algorithm 2 line 12), the interpolated boxes are discarded.

F. COMBINING LOCAL TRACKLET WITH GLOBAL IDENTITY

Our proposed method connects identical tracklets with different local identities among multiple cameras by assigning a singular global identity. For example, tracklet T_i^e, beside local identity i on camera e, is also represent by $\mathcal{G}_i$. However, the bounding box coordinate of T_i^e localizes the tracklet position only on the viewpoint of camera e. Therefore, at each association time t, to generate the tracklet outputs of T_i^e on ego camera e, we replace the local tracket identity i with associated $\mathcal{G}_i$ when $\mathcal{G}_i > 0$, and keep the same bounding box coordinate of original T_i, including with possible interpolated bounding boxes as described in Section III-E.

IV. EXPERIMENTS

A. DATASET

KITTI 2D multi-object tracking evaluations [10] provide 21 training and 29 test sequences captured at 10Hz with 8008 and 11095 images, respectively. Each sequence consists of two stereo-matching images from camera 2 and camera 3, where camera 3 is on the right side of camera 2. However, only the camera 2 subset has ground truth for the MOT task. While the KITTI dataset has annotations for 8 classes, it provides evaluation labels only for the 'Car' and 'Pedestrian' classes for MOT tasks. In CollabMOT, we only evaluate the performance of the 'Car' class. KITTI also does not offer validation datasets, so we follow [1], [2], and [3] to divide training images equally into training and validation sets for ablation studies.

Argoverse-HD dataset [17] extends Argoverse 3D tracking dataset [13], focus on 2D object detection for streaming perception evaluation. Argoverse-HD provides dense 2D annotations with track ID for the *ring_front_center* camera. This is because 2D annotations are more accurate than converting 3D cuboids to 2D bounding boxes. The trackid annotations are done manually to ensure quality, but some residual noise remains.[1] Argoverse-HD does not provide an official 2D MOT benchmark [17], so we follow the setup and evaluation metric from KITTI 2D MOT. To achieve a stereo camera setup similar to the KITTI dataset, we synchronize the images of *ring_front_center* (camera 2) with the *stereo_front_right* (camera 3) subset using conversion scripts from the official repository.[2] The *ring_front_center* camera captures images at 30Hz, whereas the *stereo_front_right* camera captures images at only 5Hz. After the mapping process, we end up with 2520 pairs of images. It is worth noting that the ring camera captures images at a resolution of 1920 × 1200, while the stereo camera captures images at 2056 × 2464 resolution. The baseline between two cameras is 0.14m, which is narrower than KITTI stereo camera setup (0.53m).

In the evaluation section, following the camera setup from the KITTI dataset,[3] we refer to the left and right cameras in the stereo camera setup as 'camera 2' and 'camera 3', respectively. The performance of CollabMOT is evaluated on one of the two cameras as an 'ego camera', and the other is considered as a 'remote camera'. However, since there

[1] https://github.com/mtli/sAP/blob/master/doc/data_setup.md
[2] https://github.com/argoverse/argoverse-api
[3] https://cvlibs.net/datasets/kitti/setup.php

TABLE 2. Appearance feature encoder summary. The latency is measured in milliseconds with batch size of 32.

Network	Prec@1	MACS	Params	Latency	Input Size	Network Type
EdgeNeXt-XXS [46]	0.83	256.821M	1.167M	10	256×256	Hybrid
GENet-light [47]	0.85	742.437M	6.341M	5	224×224	Convolution
Mixnet-S [48]	0.87	237.813M	2.664M	12	224×224	Convolution
MobileViT XS [49]	0.79	911.631M	1.949M	19	256×256	Hybrid
XCit [50]	0.85	2.079G	2.923M	34	224×224	Transformer

is no ground truth annotation available for 'camera 3' in the dataset, we consider 'camera 2' as the 'ego camera' for evaluation in all experiments.

B. EVALUATION METRICS

We use the official evaluation metrics of the KITTI dataset. High Order Tracking Accuracy (HOTA) [18] is the balanced combination of detection (DetA) and association (AssA) accuracy in terms of IOU score. Given predicted and ground-truth trajectories, DetA calculated the ratio of spatial intersection between predicted and ground-truth detections over their union detections. On the other hand, AssA measures the temporal intersection of identities over their union of IDs. In this work, we report HOTA and two partial AssA and DetA scores.

In addition to HOTA metrics, we also report standard benchmark metrics in the MOT task. The MOTA metric [51] has been employed as the primary MOT evaluation metric since 2006, but it places too much emphasis on detection performance over association performance [18]. As a result, we only report the IDSW score from the MOTA metric. On the other hand, the ID metric [52], designed specifically for evaluating tracking in a multi-camera setting, is the main metric for the NVIDIA AI city MTMCT benchmark [36]. Moreover, it is also applicable to standard single-camera settings [53].

C. IMPLEMENTATION DETAILS

1) HYPERPARAMETERS

Since our algorithm is focused on multi-tracklets association between stereo cameras, we employ the state-of-the-art 2D MOT method in the SCT step. As base SCT methods, we selected CenterTrack [1], PermaTrack [2], and RAM [3], which are open-source and published in peer-reviewed journals or conferences. All three methods are joint detection and tracking MOT algorithms. We also select DeepSORT [7] as one of the SCT methods for the second camera. We follow the method of [5], which uses the detection results from the baseline SCT algorithm as the input for track-by-detection SCT methods. In DeepSORT [7], we set $min\_hits = 1$ and $max\_iou\_distance = 0.9$. For other SCTs, we use the same parameters reported in their respective papers.

To demonstrate the robustness and adaptability of our proposed method, we avoid specific fine-tuning of CollabMOT when applying it to the output of the baseline SCT algorithm. In particular, for hyperparameters used in the CollabMOT, we set the following parameters used by Equation 2, 4, and 5: $\omega_a = 1$, $\omega_p = 1$, $\lambda_{appearance} = 0.1$, $\lambda_{occlusion} = 0.5$. The decay rate for EMA is configured as $\alpha = 0.9$ in Equation 3. $\omega_{interpolated} = 2$, $\lambda_{int\_iou} = 0.5$, and $\lambda_{int\_occlusion} = 0.5$ is being used in Section III-E.

2) APPEARANCE FEATURE ENCODER

Recently, with the rapid growth of high-performance neural network architectures, we have selected only a few notable architectures as the backbone for the appearance feature encoder. For a fair comparison, we designed a simple dense layer to map the output of the backbone to the desired embedding space without any middle layer. We follow [54] to train several appearance feature encoders on VeRi dataset [55] and initialize the training process with pre-trained weight on ImageNet dataset [56]. All pre-trained weights are provided by [57]. The same augmentations method is applied for every appearance encoder training. We resize all images to 300 × 300 and apply random cropping to the default input size of each network at the training step. At the inference step, we use center-cropping instead of random cropping.

In the image reidentification task, precision at n (Prec@n) is the proportion of the top-n images that are relevant to the query images. We employ Prec@1 as a metric to determine the training time. We train the ReID network until the Prec@1 is no longer improved. The default final encoder output is 64. We report the Precision at 1 (Prec@1), the number of MAC (multiply-added operation) [58], number of network parameters, latency (or throughput) in milliseconds with batch size of 32 on ONNX runtime [59], network input size and network type of all appearance feature encoders in Table 2.

All experiments are conducted on a personal computer with a single NVIDIA RTX3090 GPU, Intel i7-11700K, and 64GB of system RAM.

D. BENCHMARK RESULTS

Table 3 compares CollabMOT's performance with previous works on the KITTI Object Tracking Evaluation benchmark with 2D bounding boxes. The KITTI leaderboard provides the setup information.[4] We compare CollabMOT against

[4] https://cvlibs.net/datasets/kitti/eval_tracking.php

TABLE 3. Comparison with previous work on KITTI testing dataset. Each camera of CollabMOT is configured with an individual Single Camera Tracklet based on three state-of-the-art methods: CenterTrack [1], PermaTrack [2] and RAM [3]. Here, Laser and Stereo stand for the methods that utilize lidar-based point cloud and stereo vision information, respectively. The best results are in boldface.

Methods	Setup	HOTA ↑	AssA ↑	DetA ↑	IDSW ↓
CIWT [60]	**O**nline+**S**tereo	54.9	50.0	60.6	491
MOTSFusion [61]	**S**tereo	68.7	66.2	72.2	415
TrackMPNN [62]	**O**nline	72.3	70.6	74.7	481
CenterTrack [1]	**O**nline	73.0	71.2	75.6	254
CollabMOT (Left: CenterTrack [1], Right: CenterTrack [1])	semi-**O**nline+**S**tereo	73.5	72.1	75.6	246
CollabMOT (Left: CenterTrack [1], Right: DeepSORT [7])	semi-**O**nline+**S**tereo	75.3	75.7	75.5	227
DeepFusionMOT [63]	**O**nline+**S**tereo+**L**aser	75.5	80.0	71.5	**84**
OC-SORT [5]	**O**nline	76.5	76.4	77.2	250
PermaTrack [2]	**O**nline	78.0	78.4	78.3	258
CollabMOT (Left: PermaTrack [2], Right: PermaTrack [2])	semi-**O**nline+**S**tereo	78.2	78.7	78.4	246
CollabMOT (Left: PermaTrack [2], Right: DeepSORT [7])	semi-**O**nline+**S**tereo	78.5	79.3	78.4	249
CollabMOT (Left: RAM [3], Right: RAM [3])	semi-**O**nline+**S**tereo	79.5	80.8	**78.8**	210
RAM [3]	**O**nline	79.5	80.9	**78.8**	210
CollabMOT (Left: RAM [3], Right: DeepSORT [7])	semi-**O**nline+**S**tereo	**80.0**	**81.9**	**78.8**	207

other multi-object tracking algorithms utilizing either 2D images or stereo setups. All the performance scores are collected from the KITTI evaluation website. KITTI hides annotations and limits multiple submissions to the evaluation server to prevent the authors from tuning their hyperparameters for better results. For each submission to the KITTI leaderboard, CollabMOT is configured by selecting one of the three baseline tracker algorithms for both the left camera (Camera 2) and the right camera (Camera 3). In addition to the similar baseline trackers setup, the right camera (Camera 3) also employs DeepSORT [7] in some combinations. We use the same hyperparameters and appearance feature encoder based on MixNet-S [48] for each submission.

At Table 3, we have made significant enhancements to the association accuracy (AssA) of the baseline tracker's performance across all combinations of CollabMOT. Our findings demonstrate that CollabMOT achieves superior results when combined with different trackers compared to similar baseline trackers. We attribute this to the concept of ensemble machine learning [64], where the integration of results from multiple models generally yields positive outcomes and leads to better performance. Our method does sacrifice the accuracy of the detection in order to complete the trajectory of interrupted tracklets, especially when the MOT algorithm with lower detection accuracy is adopted. With the improvement of AssA, CollabMOT substantially improves the HOTA score of all three baseline methods. In Table 3 and Figure 2, CollabMOT shows significant improvement in AssA and HOTA scores on the baseline of CenterTrack as CenterTrack prioritizes the short-term association of tracklets in consecutive frames rather than a long-term association of re-emerged tracks. The combination of RAM and DeepSORT in CollabMOT outperforms other variations of CollabMOT, as the baseline performance of RAM is better than PermaTrack and CenterTrack. In particular, RAM [2] is a state-of-the-art public online MOT on 2D images ranked at the top of the list in KITTI leaderboard.

In Table 4, we report the performance of CollabMOT on the Argoverse-HD validation dataset with the baseline of DeepSORT with StreamYOLO [65] detector trained on Argoverse-HD [17] train dataset. Compared to KITTI dataset, we obtain an inferior increase in terms of HOTA and AssA scores. The baseline for two cameras in the KITTI dataset is 0.53 m. In contrast, the Argoverse-HD dataset has a baseline of approximately 0.14 m. The narrow baseline and different resolutions may contribute marginally to the improvement compared to the KITTI dataset. In addition, we also provide results for CollabMOT using Yolov8 [21] and YoloX [22], both trained on COCO [66] dataset. The effectiveness of CollabMOT can be seen on both optimized and non-optimized detectors.

Apart from tracking performance, inference speed is also a crucial metric. CollabMOT inherits the two-step hierarchical setup commonly found in MTMCT [43], considering that each camera performs its own MOT tracking algorithm. To compute the average latency reported in Table 5, we use an additional system with a similar specification to run the baseline MOT and feature encoder on the right camera and provide the output to the system that executes the left camera (ego camera). The latency of Baseline SCT and Feature Extractor is averaged across left and right cameras. Secondary SCT represents the additional latency in the right camera if a different MOT (DeepSORT [7]) is used. For all combinations of SCT of the CollabMOT, the latency of the distance calculation in Section III-C and multi-camera tracklets association in Section III-D (D+A), which is only executed in ego camera, is approximately 2.79∼3.16 ms per frame. Therefore, CollabMOT is a good solution to improve tracking performance, as low-cost stereo-camera systems are increasingly adopted in self-driving vehicles and ADAS systems.

E. ABLATION STUDY

In this section, we show the ablation study of the proposed algorithm. We compare the performance of CollabMOT with the left camera (camera 2) is selected as ego camera against the baseline results from CenterTrack [1], PermaTrack [2], and RAM [3], respectively.

TABLE 4. Evaluation of CollabMOT on Argoverse datasets with the left camera is selected as the ego camera. DeepSORT [7] is employed for the right SCT of CollabMOT. The best results are marked by boldface.

Detector	Method	HOTA ↑	AssA ↑	DetA ↑	IDF1 ↑	IDSW ↓
StreamYOLO [65]	DeepSORT [7]	50.38	53.38	**48.53**	56.71	1344
	CollabMOT	**50.44**	**53.52**	**48.53**	**56.76**	**1341**
Yolov8-L [21]	DeepSORT [7]	45.46	56.72	**36.71**	50.84	542
	CollabMOT	**45.52**	**56.89**	36.70	**50.97**	**539**
YoloX-L [22]	DeepSORT [7]	43.29	56.74	**33.28**	48.42	385
	CollabMOT	**43.37**	**56.94**	**33.28**	**48.52**	**382**

TABLE 5. Latency analysis of each component of CollabMOT on 11095 images from KITTI testing datasets for each submission. The unit is in milliseconds. D+A is the combined latency of the distance calculation in Section III-C and multi-camera tracklets association in Section III-D.

Left	Right	Baseline SCT	Secondary SCT	Feature extractor	D+A	Totals
CenterTrack [1]	CenterTrack [1]	61.11	0.00	5.14	2.79	69.05
	DeepSORT [7]	61.11	0.93	5.36	2.94	70.34
PermaTrack [2]	DeepSORT [7]	78.77	0.95	5.37	2.88	87.97
	PermaTrack [2]	78.77	0.00	5.42	3.16	87.36
RAM [3]	RAM [3]	84.90	0.00	5.40	3.04	93.35
	DeepSORT [7]	84.90	0.93	5.44	2.89	94.17

1) COMBINATION OF DIFFERENT METHODS OF SCT

CollabMOT aims to provide a stronger association by combining the results of individual SCT on a stereo camera system. In Table 6, noticeable improvements can be seen in the cases of CollabMOT. This result demonstrates that the tracking performance can benefit from the use of stereo cameras. For all four notable baseline SCT algorithms, combining two different SCT gives better performance over a unified SCT algorithm, similar to the results in Table 3. For example, HOTA, AssA and IDF1 scores for CenterTrack [1] increase by 1.12%, 2.15% and 2.12%, respectively, when CenterTrack is combined with DeepSORT. These results indicate that CollabMOT can significantly improve SCT's results with multiple baseline MOT algorithms.

2) TRACKLETS INTERPOLATION

To evaluate the contribution of the proposed interpolation method discussed in Section III-E, we compare the performance of CollabMOT in two modes: with and without interpolation mode in Table 7. Without interpolation, for all three baseline methods, DetA score of CollabMOT does not increase because CollabMOT does not generate any bounding box to complete the interrupted trajectory. When interpolation is enabled, DetA score slightly increases. In particular, the DetA score improvement of CenterTrack [1] is the smallest among the three methods. This is attributed to the fact that interpolated bounding box accuracy depends on the accuracy of the detected bounding boxes, and baseline CenterTrack [1] has the lowest detection accuracy among three baseline MOTs. Other than DetA, CollabMOT significantly enhances the performance of various baseline MOT algorithms by improving IDSW, IDF1, AssA, and HOTA scores. This improvement is consistent across all methods, regardless of whether they use interpolation or not. Other papers such as [4] and [45] consider interpolation as a post-processing step and thus, the interpolation is conducted after the tracking is all finished for the last image in each sequence. On the other hand, our method performs linear interpolation right after it reconnects the interrupted tracklets based on inter-camera multi-tracklet transitive relations.

3) EVALUATION ON VARIOUS APPEARANCE ENCODER NEURAL NETWORKS

We evaluate CollabMOT on the KITTI half-validation dataset using various lightweight backbone networks for appearance feature encoder. The result is shown in Table 8. The detailed information of each encoder is described in Table 2. The results of this experiment indicate that CollabMOT can be used for various appearance encoder profiles.

Furthermore, we evaluate CollabMOT performance with several embedding sizes on MixNet-S [48]. From Table 9, CollabMOT enhances the tracking performance over the base method for all embedding sizes. As reported by [8], it has been known that lower dimensional re-ID features usually cause less harm to the tracking accuracy. These results also help us to choose the optimal embedding size for the appearance feature encoder.

4) THRESHOLD SENSITIVITY

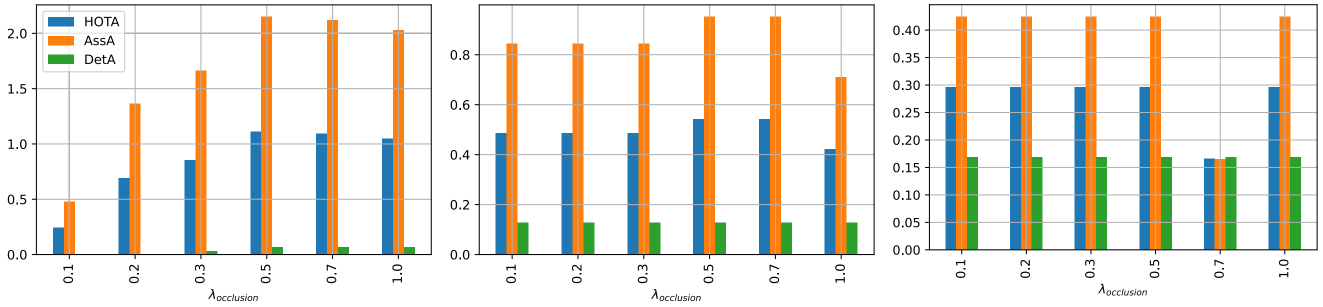
In Figure 6, we demonstrate our proposed method under different values of $\lambda_{occlusion}$, as defined in Equation 4, for all three baseline methods. The lower the value of $\lambda_{occlusion}$, the fewer qualified bounding boxes in tracklets are selected

TABLE 6. Evaluation of CollabMOT with different combinations of MOT methods on Right (Camera 3) on *validation* set. Each subtable presents the baseline MOT performance in the first row. In many combinations of CollabMOT, the performance is improved by more than +0.5 points, which are highlighted by boldface. For baseline DeepSORT [7], we use the detection results from PermaTrack [2].

Method	Left	Right	HOTA ↑	AssA ↑	DetA ↑	IDF1 ↑	IDSW ↓
Baseline SCT	CenterTrack [1]		76.72	76.54	77.26	85.48	64
CollabMOT	CenterTrack [1]	CenterTrack [1]	77.37 (**+0.65**)	77.82 (**+1.28**)	77.26	86.68 (**+1.20**)	61
		DeepSORT [7]	77.84 (**+1.12**)	78.69 (**+2.15**)	77.32 (+0.06)	87.60 (**+2.12**)	**55**
Baseline SCT	DeepSORT [7]		78.46	77.65	79.61	86.28	111
CollabMOT	DeepSORT [7]	DeepSORT [7]	79.21 (**+0.75**)	78.96 (**+1.31**)	79.78 (+0.17)	87.50 (**+1.22**)	101
		PermaTrack [2]	79.36 (**+0.90**)	79.24 (**+1.59**)	79.80 (+0.19)	87.81 (**+1.53**)	**98**
Baseline SCT	PermaTrack [2]		80.36	80.71	80.32	88.79	59
CollabMOT	PermaTrack [2]	DeepSORT [7]	80.90 (**+0.54**)	81.66 (**+0.95**)	80.45 (+0.13)	89.73 (**+0.94**)	58
		PermaTrack [2]	80.92 (**+0.56**)	81.70 (**+0.99**)	80.45 (+0.13)	89.79 (**+1.00**)	**57**
Baseline SCT	RAM [3]		80.39	81.19	79.93	90.61	33
CollabMOT	RAM [3]	RAM [3]	80.61 (+0.22)	81.46 (+0.27)	80.10 (+0.17)	90.88 (+0.27)	**32**
		DeepSORT [7]	80.69 (+0.30)	81.62 (+0.43)	80.10 (+0.17)	91.05 (+0.44)	**32**

TABLE 7. Evaluation of CollabMOT with and without interpolation mode on the KITTI validation datasets. The best results are marked by boldface. DeepSORT [7] is employed for the right SCT of CollabMOT. The text highlights the best IDSW metric results in bold and shows improvements of CollabMOT over the baseline method in other metrics in brackets, with additional bold highlights for improvements over +0.5 points.

Left	Method	Interpolation	HOTA ↑	AssA ↑	DetA ↑	IDF1 ↑	IDSW ↓
CenterTrack [1]	Baseline SCT		76.72	76.54	77.26	85.48	64
	CollabMOT	**No**	77.77 (**+1.05**)	78.62 (**+2.08**)	77.26	87.57 (**+2.09**)	**55**
		Yes	77.84 (**+1.12**)	78.69 (**+2.15**)	77.32 (+0.06)	87.60 (**+2.12**)	**55**
PermaTrack [2]	Baseline SCT		80.36	80.71	80.32	88.79	59
	CollabMOT	**No**	80.79 (+0.43)	81.56 (**+0.85**)	80.32	89.61 (**+0.82**)	**57**
		Yes	80.90 (**+0.54**)	81.66 (**+0.95**)	80.45 (+0.13)	89.73 (**+0.94**)	58
RAM [3]	Baseline SCT		80.39	81.19	79.93	90.61	33
	CollabMOT	**No**	80.53 (+0.14)	81.46 (+0.27)	79.93	90.90 (+0.29)	**32**
		Yes	80.69 (+0.30)	81.62 (+0.43)	80.10 (+0.17)	91.05 (+0.44)	**32**



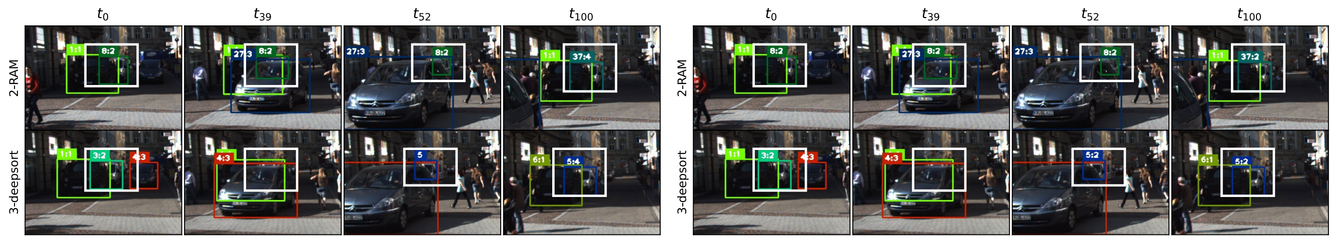
(a) CenterTrack

(b) PermaTrack

(c) RAM

FIGURE 6. The bar graph illustrates the improved performance of CollabMOT over three baseline MOT algorithms on the left camera in the validation set of the KITTI dataset, with a different value of $\lambda_{occlusion}$. DeepSORT [7] is employed as SCT for the right camera.

for EMA update in Equation 3. The results obtained from the three baseline methods reveal that the benchmark metrics gradually increase with an increase in the $\lambda_{occlusion}$. However, it is important to note that a higher $\lambda_{occlusion}$ can result in the selection of more occluded bounding boxes, which may lead to the inclusion of noisy and non-discriminative appearance features in the EMA process. It is observed that the improvement in all three baseline methods reaches its peak when $\lambda_{occlusion}$ is set to 0.5, and tends to decrease beyond this value. Therefore, choosing an optimal value for $\lambda_{occlusion}$ is essential to obtain the best performance in the EMA process. In Figure 6c, the metric drops when the $\lambda_{occlusion}$ is set to 0.7 but increases at 1.0. This behavior is explained by Figure 7. At t_{52}, ID 5 and ID 4 overlap in camera 2, resulting in ID 5 not being able to associate with ID 8. Consequently, the transitive relationship between ID 5



(a) $\lambda_{occlusion} = 0.7$ **(b)** $\lambda_{occlusion} = 1.0$

FIGURE 7. Qualitative examples demonstrate the CollabMOT performance with two threshold values of $\lambda_{occlusion}$ on similar sequence from KITTI dataset and similar SCT setup in each camera.

TABLE 8. Evaluation on Kitti half validation dataset with multiple feature encoder. We use the same embedding size of 64. DeepSORT [7] is employed for the right SCT of CollabMOT. Each subtable presents the baseline MOT performance in the first row. The text highlights the best IDSW metric results in bold and shows improvements of CollabMOT over the baseline method in other metrics in brackets, with additional bold highlights for improvements over +0.5 points.

Left	Method	Encoder Network	HOTA ↑	AssA ↑	DetA ↑	IDF1 ↑	IDSW ↓
CenterTrack [1]	Baseline SCT	n/a	76.7	76.5	77.3	85.5	64
	CollabMOT	Mixnet-S [48]	77.8 (**+1.1**)	78.7 (**+2.2**)	77.3	87.6 (**+2.1**)	55
		XCit [50]	77.4 (**+0.7**)	77.8 (**+1.3**)	77.3	87.2 (**+1.7**)	59
		MobileViT XS [49]	78.1 (**+1.4**)	79.3 (**+2.8**)	77.3	88.2 (**+2.7**)	**52**
		EdgeNeXt-XXS [46]	77.7 (**+1.0**)	78.4 (**+1.9**)	77.3	87.2 (**+1.7**)	56
		GENet-light [47]	77.7 (**+1.0**)	78.5 (**+2.0**)	77.3	87.5 (**+2.0**)	54
PermaTrack [2]	Baseline SCT	n/a	80.4	80.7	80.3	88.8	59
	CollabMOT	Mixnet-S [48]	80.9 (**+0.5**)	81.7 (**+1.0**)	80.4 (+0.1)	89.7 (**+0.9**)	58
		XCit [50]	81.2 (**+0.8**)	82.2 (**+1.5**)	80.4 (+0.1)	90.4 (**+1.6**)	**56**
		MobileViT XS [49]	80.9 (**+0.5**)	81.7 (**+1.0**)	80.4 (+0.1)	89.7 (**+0.9**)	58
		EdgeNeXt-XXS [46]	80.9 (**+0.5**)	81.7 (**+1.0**)	80.4 (+0.1)	89.7 (**+0.9**)	58
		GENet-light [47]	81.3 (**+0.9**)	82.5 (**+1.8**)	80.4 (+0.1)	90.6 (**+1.8**)	**56**
RAM [3]	Baseline SCT	n/a	80.4	81.2	79.9	90.6	33
	CollabMOT	Mixnet-S [48]	80.7 (+0.3)	81.6 (+0.4)	80.1 (+0.2)	91.0 (+0.4)	**32**
		XCit [50]	80.6 (+0.2)	81.4 (+0.2)	80.1 (+0.2)	90.8 (+0.2)	**32**
		MobileViT XS [49]	80.7 (+0.3)	81.6 (+0.4)	80.1 (+0.2)	91.0 (+0.4)	**32**
		EdgeNeXt-XXS [46]	80.7 (+0.3)	81.6 (+0.4)	80.1 (+0.2)	91.0 (+0.4)	**32**
		GENet-light [47]	80.7 (+0.3)	81.6 (+0.4)	80.1 (+0.2)	91.0 (+0.4)	**32**

and ID 3 via ID 8 is lost. In a later frame, although ID 8 is switched to ID 37 and ID 37 is interconnected with ID 5, ID 37 is still unable to intra-connect with ID 8. However, when $\lambda_{occlusion}$ is set to 1.0, ID 5 and ID 8 are interconnected, and the transitive relationship is maintained, allowing ID 37 and ID 8 to reconnect.

F. QUALITATIVE RESULTS

Figure 8 visualizes a difficult tracking example where conventional baseline MOT algorithms poorly perform, while CollabMOT can improve the performance of baseline MOT algorithms. More qualitative examples are provided in the supplementary video.[5]

V. DISCUSSION AND LIMITATION

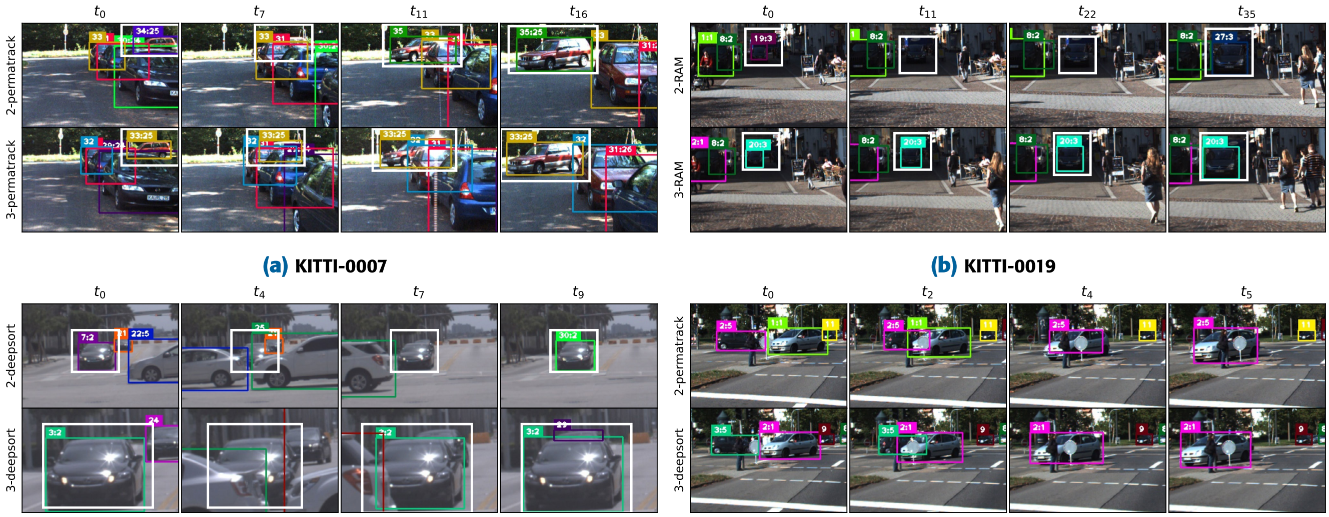
In the proposed CollabMOT, we utilize the highly overlapping view from a stereo camera setup to improve the system's performance of a single-camera MOT. The CollabMOT design takes inspiration from our binocular vision system, where humans use both eyes to perceive the world. This system significantly improves our ability to perceive the surroundings around us. By relying on both eyes, we can see the world with greater clarity and depth, which cannot be achieved by using only one eye [67]. While the stereo camera setup is common in autonomous driving [10], [17], other applications of MOT in surveillance only utilize a single camera setup [53], [68], [69]. As demonstrated in the paper, CollabMOT is capable of functioning without multi-camera annotation. However, the lack of a stereo view poses a constraint on its deployment on the MOT benchmark, which provides only a monocular view. We hoped that the implementation of CollabMOT could attract more attention and suggest the creation of a comprehensive MOT benchmark with stereo data.

Our experiments have revealed additional limitations of CollabMOT. For instance, in Figure 8d, which illustrates

[5] https://youtu.be/zXakz9p97Ss

TABLE 9. Evaluation of CollabMOT on multiple feature encoder embedding sizes of same architecture [48]. DeepSORT [7] is employed for the right SCT of CollabMOT. The text highlights the best IDSW metric results in bold and shows improvements of CollabMOT over the baseline method in other metrics in brackets, with additional bold highlights for improvements over +0.5 points.

Left	Method	Size	HOTA ↑	AssA ↑	DetA ↑	IDF1 ↑	IDSW ↓
CenterTrack [1]	Baseline SCT	n/a	76.7	76.5	77.3	85.5	64
	CollabMOT	32	77.8 (**+1.1**)	78.7 (**+2.2**)	77.3	87.6 (**+2.1**)	**55**
		64	77.8 (**+1.1**)	78.7 (**+2.2**)	77.3	87.6 (**+2.1**)	**55**
		128	77.7 (**+1.0**)	78.4 (**+1.9**)	77.3	87.4 (**+1.9**)	58
		256	77.5 (**+0.8**)	78.0 (**+1.5**)	77.3	87.0 (**+1.5**)	60
PermaTrack [2]	Baseline SCT	n/a	80.4	80.7	80.3	88.8	59
	CollabMOT	32	80.9 (**+0.5**)	81.7 (**+1.0**)	80.4 (+0.1)	89.7 (**+0.9**)	**58**
		64	80.9 (**+0.5**)	81.7 (**+1.0**)	80.4 (+0.1)	89.7 (**+0.9**)	**58**
		128	80.6 (+0.2)	81.1 (+0.4)	80.4 (+0.1)	89.2 (+0.4)	59
		256	80.6 (+0.2)	81.1 (+0.4)	80.4 (+0.1)	89.2 (+0.4)	59
RAM [3]	Baseline SCT	n/a	80.4	81.2	79.9	90.6	33
	CollabMOT	32	80.7 (+0.3)	81.6 (+0.4)	80.1 (+0.2)	91.0 (+0.4)	**32**
		64	80.7 (+0.3)	81.6 (+0.4)	80.1 (+0.2)	91.0 (+0.4)	**32**
		128	80.7 (+0.3)	81.6 (+0.4)	80.1 (+0.2)	91.0 (+0.4)	**32**
		256	80.7 (+0.3)	81.6 (+0.4)	80.1 (+0.2)	91.0 (+0.4)	**32**



(a) KITTI-0007

(b) KITTI-0019

(c) ArgoverseHD

(d) KITTI-0002

FIGURE 8. Visualization of CollabMOT. The border color and the unique number before colon mark inside each box represent the local identity of the tracklets in each camera view, and the number after colon mark represents the global identity of the tracklets in a multi-camera system. The targets are fixed by CollabMOT and are highlighted by white boxes. Best view in color and zoom in.

the output of Permatrack on camera 2, the tracklet of the vehicle with ID 2 was wrongly merged into the tracklet with ID 1 in a later frame. While DeepSort successfully separated the two tracklet instances on the right camera, ID 2 was inter-camera associated with ID 1 on the left camera. Because of its inter-camera connection to ID 1 in the previous frame, indicated by the global ID after the colon mark, CollabMOT refuses to allow ID 2 on the right camera to connect with ID 2 on the left camera. This example highlights a limitation of our method when the baseline algorithm propagates tracklets incorrectly to different objects. CollabMOT respects and follows the decision of the baseline algorithm.

VI. CONCLUSION

In this paper, we proposed a novel inter-camera tracklet association algorithm called CollabMOT, which utilizes stereo cameras to improve the performance of multi-object tracking. CollabMOT dynamically associates tracklets generated from separate multi-object tracking algorithms on each camera based on their appearance feature and feedback the global association information to help combat the identity-switching problem on each camera. Our experiments demonstrated the improvement in terms of HOTA metrics on KITTI 2D tracking and Argoverse-HD dataset when applying CollabMOT to the output of state-of-the-art published MOT algorithms. CollabMOT also adapts to multiple setups and a

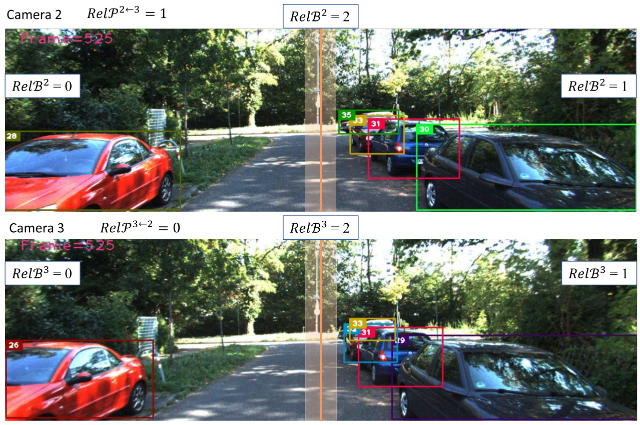


FIGURE 9. Illustration of the relative position of tracklet in each camera perspective. The transparent rectangle indicates the center area in each camera view.

TABLE 10. Truth table for association cost between two tracklet T^e and T^r which is derived by their respective bounding box between stereo camera e and r.

$Rel\mathcal{P}^{e\leftarrow r}$	$Rel\mathcal{B}^e$	$Rel\mathcal{B}^r$	Cost
0	1	0	∞
1	0	1	∞

variety of appearance feature encoders. Through this work, we have shown that CollabMOT utilizes stereo cameras to further increase the performance of multi object tracking in ADAS or autonomous driving systems. Future work will focus on improving the performance and increasing the adaptation of CollabMOT on different multi-object tracking domains.

APPENDIX

POSITION COST CALCULATION

The position affinity cost Sp rules out the infeasible association between tracklets by considering the relative position of two cameras and the bounding box location of each tracklet in each respective camera perspective. We denote the relative position of remote camera r as $Rel\mathcal{P}^{e\leftarrow r}$ with respect to ego camera e. The relative position of the bounding box of tracklets in each camera perspective is denoted by $Rel\mathcal{B}$. $Rel\mathcal{P}^{e\leftarrow r}$ is defined in Equation 8.

$$Rel\mathcal{P}^{e\leftarrow r} = \begin{cases} 0 : r & \text{on the left side of } e \\ 1 : r & \text{on the right side of } e \end{cases} \tag{8}$$

For stereo cameras, $Rel\mathcal{P}^{e\leftarrow r}$ is a fixed value. For example in KITTI dataset [10], we have $Rel\mathcal{P}^{3\leftarrow2} = 0$ and $Rel\mathcal{P}^{2\leftarrow3} = 1$ for camera 2 and camera 3, respectively. This follows from the fact that camera 3 is on the right side with respect to camera 2, so the vehicle objects from camera 3 are on the right side with respect to the same vehicles captured by camera 2. The examples are shown in Figure 9.

The relative position of tracklets on each camera viewpoint can be grouped into three areas: left, center, and right. We denote left by 0, right by 1, and center by 2. The relative position of the bounding box of a tracklet in frame t is defined by Equation 9.

$$Rel\mathcal{B} = \begin{cases} 0 : x + w/2 < centerline - \lambda_{bbox} \\ 1 : x > centerline + \lambda_{bbox} \\ 2 : \text{otherwise} \end{cases} \tag{9}$$

Here, x and w are the center location and width of bounding box B, respectively. At the same time, $centerline$ is a vertical line dividing the camera view to left and right sections based on intrinsic camera parameters, and λ_{bbox} is a margin value. We set $\lambda_{bbox} = 10$ on every experiments. For each pair of T_i and T_j, we have the truth table for $RelP^r$, $RelB^e$, $RelB^r$ in Table 10.

We set $Sp = \infty$ for any pair of T_i^e and T_j^r whose $RelP^r$, $RelB^e$, $RelB^r$ are described in Table 10. Otherwise $Sp = 0$.

REFERENCES

[1] X. Zhou, V. Koltun, and P. Krähenbühl, "Tracking objects as points," in *Computer Vision—ECCV*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Cham, Switzerland: Springer, 2020, pp. 474–490.

[2] P. Tokmakov, J. Li, W. Burgard, and A. Gaidon, "Learning to track with object permanence," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 10860–10869.

[3] P. Tokmakov, A. Jabri, J. Li, and A. Gaidon, "Object permanence emerges in a random walk along memory," in *Proc. ICML*, 2022, pp. 21506–21519.

[4] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, and X. Wang, "ByteTrack: Multi-object tracking by associating every detection box," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2022, pp. 1–21.

[5] J. Cao, J. Pang, X. Weng, R. Khirodkar, and K. Kitani, "Observation-centric SORT: Rethinking SORT for robust multi-object tracking," 2022, *arXiv:2203.14360*.

[6] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 3464–3468.

[7] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 3645–3649.

[8] Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu, "FairMOT: On the fairness of detection and re-identification in multiple object tracking," *Int. J. Comput. Vis.*, vol. 129, no. 11, pp. 3069–3087, Nov. 2021.

[9] Z. Wang, L. Zheng, Y. Liu, Y. Li, and S. Wang, "Towards real-time multi-object tracking," in *Computer Vision—ECCV*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Cham, Switzerland: Springer, 2020, pp. 107–122.

[10] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[11] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "NuScenes: A multimodal dataset for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11618–11628.

[12] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, and Y. Chai, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*. Seattle, WA, USA: IEEE, Jun. 2020, pp. 2443–2451.

[13] M.-F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan, and J. Hays, "Argoverse: 3D tracking and forecasting with rich maps," 2019, *arXiv:1911.02620*.

[14] Z. Pang, J. Li, P. Tokmakov, D. Chen, S. Zagoruyko, and Y.-X. Wang, "Standing between past and future: Spatio-temporal modeling for multi-camera 3D multi-object tracking," 2023, *arXiv:2302.03802*.

[15] T. Zhang, X. Chen, Y. Wang, Y. Wang, and H. Zhao, "MUTR3D: A multi-camera tracking framework via 3D-to-2D queries," 2022, *arXiv:2205.00613*.

[16] N. Kemsaram, A. Das, and G. Dubbelman, "A stereo perception framework for autonomous vehicles," in *Proc. IEEE 91st Veh. Technol. Conf. (VTC-Spring)*, May 2020, pp. 1–6.

[17] M. Li, Y.-X. Wang, and D. Ramanan, "Towards streaming perception," 2020, *arXiv:2005.10420*.

[18] J. Luiten, A. Osep, P. Dendorfer, P. Torr, A. Geiger, L. Leal-Taixé, and B. Leibe, "HOTA: A higher order metric for evaluating multi-object tracking," *Int. J. Comput. Vis.*, vol. 129, no. 2, pp. 548–578, Feb. 2021.

[19] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[20] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[21] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLOv8," Version 8.0.0, Ultralytics, Madrid, Spain, Jan. 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[22] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO series in 2021," 2021, *arXiv:2107.08430*.

[23] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 2022, *arXiv:2207.02696*.

[24] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, Mar. 1960.

[25] Z. Tang, M. Naphade, M.-Y. Liu, X. Yang, S. Birchfield, S. Wang, R. Kumar, D. Anastasiu, and J.-N. Hwang, "CityFlow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification," 2019, *arXiv:1903.09254*.

[26] P. Bergmann, T. Meinhardt, and L. Leal-Taixé, "Tracking without bells and whistles," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 941–951.

[27] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," 2019, *arXiv:1904.07850*.

[28] N. Ballas, L. Yao, C. Pal, and A. Courville, "Delving deeper into convolutional networks for learning video representations," 2015, *arXiv:1511.06432*.

[29] A. Jabri, A. Owens, and A. A. Efros, "Space–time correspondence as a contrastive random walk," 2020, *arXiv:2006.14613*.

[30] X. Wang, E. Türetken, F. Fleuret, and P. Fua, "Tracking interacting objects using intertwined flows," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 11, pp. 2312–2326, Nov. 2016.

[31] S. Tang, B. Andres, M. Andriluka, and B. Schiele, "Subgraph decomposition for multi-target tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5033–5041.

[32] W. Choi, "Near-online multi-target tracking with aggregated local flow descriptor," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*. Santiago, Chile: IEEE, Dec. 2015, pp. 3029–3037.

[33] L. Lan, X. Wang, G. Hua, T. S. Huang, and D. Tao, "Semi-online multi-people tracking by re-identification," *Int. J. Comput. Vis.*, vol. 128, no. 7, pp. 1937–1955, Jul. 2020.

[34] M. Naphade, D. C. Anastasiu, A. Sharma, V. Jagrlamudi, H. Jeon, K. Liu, M.-C. Chang, S. Lyu, and Z. Gao, "The NVIDIA AI city challenge," in *Proc. IEEE SmartWorld, Ubiquitous Intell. Comput., Adv. Trusted Comput., Scalable Comput. Commun., Cloud Big Data Comput., Internet People Smart City Innov. (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, Santa Clara, CA, USA, Aug. 2017, pp. 1–6.

[35] M. Naphade, M.-C. Chang, A. Sharma, D. C. Anastasiu, V. Jagarlamudi, P. Chakraborty, T. Huang, S. Wang, M.-Y. Liu, R. Chellappa, J.-N. Hwang, and S. Lyu, "The 2018 NVIDIA AI City Challenge," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 530–537.

[36] M. Naphade, Z. Tang, M.-C. Chang, D. C. Anastasiu, A. Sharma, R. Chellappa, S. Wang, P. Chakraborty, T. Huang, J.-N. Hwang, and S. Lyu, "The 2019 AI City Challenge," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR) Workshops*, Jun. 2019.

[37] M. Naphade, S. Wang, D. C. Anastasiu, Z. Tang, M.-C. Chang, X. Yang, L. Zheng, A. Sharma, R. Chellappa, and P. Chakraborty, "The 4th AI City Challenge," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 2665–2674.

[38] M. Naphade, S. Wang, D. C. Anastasiu, Z. Tang, M.-C. Chang, X. Yang, Y. Yao, L. Zheng, P. Chakraborty, C. E. Lopez, A. Sharma, Q. Feng, V. Ablavsky, and S. Sclaroff, "The 5th AI City Challenge," 2021, *arXiv:2104.12233*.

[39] M. Naphade, S. Wang, D. C. Anastasiu, Z. Tang, M.-C. Chang, Y. Yao, L. Zheng, M. S. Rahman, A. Venkatachalapathy, A. Sharma, Q. Feng, V. Ablavsky, S. Sclaroff, P. Chakraborty, A. Li, S. Li, and R. Chellappa, "The 6th AI City Challenge," 2022, *arXiv:2204.10380*.

[40] L. Cai, L. He, Y. Xu, Y. Zhao, and X. Yang, "Multi-object detection and tracking by stereo vision," *Pattern Recognit.*, vol. 43, no. 12, pp. 4028–4041, Dec. 2010.

[41] J. Ferryman and A. Shahrokni, "PETS2009: Dataset and challenge," in *Proc. 12th IEEE Int. Workshop Perform. Eval. Tracking Surveill.*, Dec. 2009, pp. 1–6.

[42] T. Chavdarova, P. Baque, S. Bouquet, A. Maksai, C. Jose, T. Bagautdinov, L. Lettry, P. Fua, L. Van Gool, and F. Fleuret, "WILDTRACK: A multi-camera HD dataset for dense unscripted pedestrian detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.* Salt Lake City, UT, USA: IEEE, Jun. 2018, pp. 5030–5039.

[43] T. I. Amosa, P. Sebastian, L. I. Izhar, O. Ibrahim, L. S. Ayinla, A. A. Bahashwan, A. Bala, and Y. A. Samaila, "Multi-camera multi-object tracking: A review of current trends and future advances," *Neurocomputing*, vol. 552, Oct. 2023, Art. no. 126558.

[44] H. W. Kuhn, "The Hungarian method for the assignment problem," *Nav. Res. Logistics Quart.*, vol. 2, nos. 1–2, pp. 83–97, 1955.

[45] N. Aharon, R. Orfaig, and B.-Z. Bobrovsky, "BoT-SORT: Robust associations multi-pedestrian tracking," 2022, *arXiv:2206.14651*.

[46] M. Maaz, A. Shaker, H. Cholakkal, S. Khan, S. W. Zamir, R. M. Anwer, and F. S. Khan, "EdgeNeXt: Efficiently amalgamated CNN-transformer architecture for mobile vision applications," 2022, *arXiv:2206.10589*.

[47] M. Lin, H. Chen, X. Sun, Q. Qian, H. Li, and R. Jin, "Neural architecture design for GPU-efficient networks," 2020, *arXiv:2006.14090*.

[48] M. Tan and Q. V. Le, "MixConv: Mixed depthwise convolutional kernels," 2019, *arXiv:1907.09595*.

[49] S. Mehta and M. Rastegari, "MobileViT: Light-weight, general-purpose, and mobile-friendly vision transformer," 2021, *arXiv:2110.02178*.

[50] A. El-Nouby, H. Touvron, M. Caron, P. Bojanowski, M. Douze, A. Joulin, I. Laptev, N. Neverova, G. Synnaeve, J. Verbeek, and H. Jegou, "XCiT: Cross-covariance image transformers," 2021, *arXiv:2106.09681*.

[51] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The CLEAR MOT metrics," *EURASIP J. Image Video Process.*, vol. 2008, pp. 1–10, 2008.

[52] E. Ristani and C. Tomasi, "Features for multi-target multi-camera tracking and re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.* Salt Lake City, UT, USA: IEEE, Jun. 2018, pp. 6036–6046.

[53] P. Dendorfer, H. Rezatofighi, A. Milan, J. Shi, D. Cremers, I. Reid, S. Roth, K. Schindler, and L. Leal-Taixé, "MOT20: A benchmark for multi object tracking in crowded scenes," 2020, *arXiv:2003.09003*.

[54] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," 2017, *arXiv:1703.07737*.

[55] X. Liu, W. Liu, H. Ma, and H. Fu, "Large-scale vehicle re-identification in urban surveillance videos," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2016, pp. 1–6.

[56] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[57] R. Wightman, "PyTorch image models," HuggingFace, GitHub Repository, Manhattan, NY, USA, 2019. [Online]. Available: https://github.com/rwightman/pytorch-image-models, doi: 10.5281/zenodo.4414861.

[58] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[59] *ONNX Runtime*, ONNX Runtime Developers, Washington, DC, USA, 2021.

[60] A. Osep, W. Mehner, M. Mathias, and B. Leibe, "Combined image- and world-space tracking in traffic scenes," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 1988–1995.

[61] J. Luiten, T. Fischer, and B. Leibe, "Track to reconstruct and reconstruct to track," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 1803–1810, Apr. 2020.

[62] A. Rangesh, P. Maheshwari, M. Gebre, S. Mhatre, V. Ramezani, and M. M. Trivedi, "TrackMPNN: A message passing graph neural architecture for multi-object tracking," 2021, *arXiv:2101.04206*.

[63] X. Wang, C. Fu, Z. Li, Y. Lai, and J. He, "DeepFusionMOT: A 3D multi-object tracking framework based on camera-LiDAR fusion with deep association," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 8260–8267, Jun. 2022.

[64] C. Zhang and Y. Ma, *Ensemble Machine Learning: Methods and Applications*. New York, NY, USA: Springer, 2012.

[65] J. Yang, S. Liu, Z. Li, X. Li, and J. Sun, "Real-time object detection for streaming perception," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*. New Orleans, LA, USA: IEEE, 2022, pp. 5375–5385.

[66] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. ECCV*, 2014, pp. 740–755.

[67] R. K. Jones and D. N. Lee, "Why two eyes are better than one: The two views of binocular vision," *J. Exp. Psychol., Human Perception Perform.*, vol. 7, no. 1, pp. 30–40, 1981.

[68] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler, "MOTChallenge 2015: Towards a benchmark for multi-target tracking," 2015, *arXiv:1504.01942*.

[69] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, "MOT16: A benchmark for multi-object tracking," 2016, *arXiv:1603.00831*.

PHONG PHU NINH (Graduate Student Member, IEEE) received the B.S. degree in electronic engineering from the Tech University of Korea, South Korea, in 2017. He is currently pursuing the Ph.D. degree in electronics engineering with Chungbuk National University, Cheongju-si, South Korea. His current research interests include communication systems, computer vision, multi object tracking, and anomaly detection for smart factory applications.

HYUNGWON KIM received the B.S. and M.S. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), in 1991 and 1993, respectively, and the Ph.D. degree in electrical engineering and computer science from the University of Michigan, Ann Arbor, MI, USA, in 1999. In 1999, he joined Synopsys Inc., Mountain View, CA, USA, where he developed electronic design automation software. In 2001, he joined Broadcom Corporation, San Jose, CA, USA, where he developed various network chips, including a WiFi gateway router chip, a network processor for 3G, and 10gigabit ethernet chips. In 2005, he founded Xronet Corporation, a Korea-based wireless chip maker, where he was the CTO and the CEO, he managed the company to successfully develop and commercialize wireless baseband and RF chips and software, including WiMAX chips supporting IEEE802.16e and WiFi chips supporting IEEE802.11a/b/g/n. Since 2013, he has been with Chungbuk National University, Cheongju-si, South Korea, where he is currently a Full Professor with the Department of Electronics Engineering. His current research interests include artificial intelligence, deep learning, image recognition, CNN accelerator chip design, autonomous driving with AI and V2X, wireless sensor networks, mixed signal SoC designs, and low power sensor circuits.