This WACV 2023 paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Hard to Track Objects with Irregular Motions and Similar Appearances? Make It Easier by Buffering the Matching Space

Fan Yang, Shigeyuki Odashima, Shoichi Masui, Shan Jiang
Fujitsu Research, Japan
contact: fan.yang@fujitsu.com

Abstract

We propose a Cascaded Buffered IoU (C-BIoU) tracker to track multiple objects that have irregular motions and indistinguishable appearances. When appearance features are unreliable and geometric features are confused by irregular motions, applying conventional Multiple Object Tracking (MOT) methods may generate unsatisfactory results. To address this issue, our C-BIoU tracker adds buffers to expand the matching space of detections and tracks, which mitigates the effect of irregular motions in two aspects: one is to directly match identical but non-overlapping detections and tracks in adjacent frames, and the other is to compensate for the motion estimation bias in the matching space. In addition, to reduce the risk of overexpansion of the matching space, cascaded matching is employed: first matching alive tracks and detections with a small buffer, and then matching unmatched tracks and detections with a large buffer. Despite its simplicity, our C-BIoU tracker works surprisingly well and achieves state-of-the-art results on MOT datasets that focus on irregular motions and indistinguishable appearances. Moreover, the C-BIoU tracker is the dominant component for our 2^{nd} place solution in the CVPR'22 SoccerNet MOT and the ECCV'22 MOTComplex DanceTrack challenges. Finally, we analyze the limitation of our C-BIoU tracker in ablation studies and discuss its application scope.

1. Introduction

Multiple Object Tracking (MOT) is widely applied to identify the trajectory of each object in sequential data (*e.g.*, videos). It offers important information for real-world applications which include but are not limited to autonomous driving [14], sports and dance analysis [26, 9], and animal surveys [1, 17].

Although MOT studies have been greatly developed [5, 31, 30, 35, 34, 18], a new challenge has recently attracted attention: unlike conventional MOT tasks that contain objects with distinct appearances and regular motions, MOT tasks

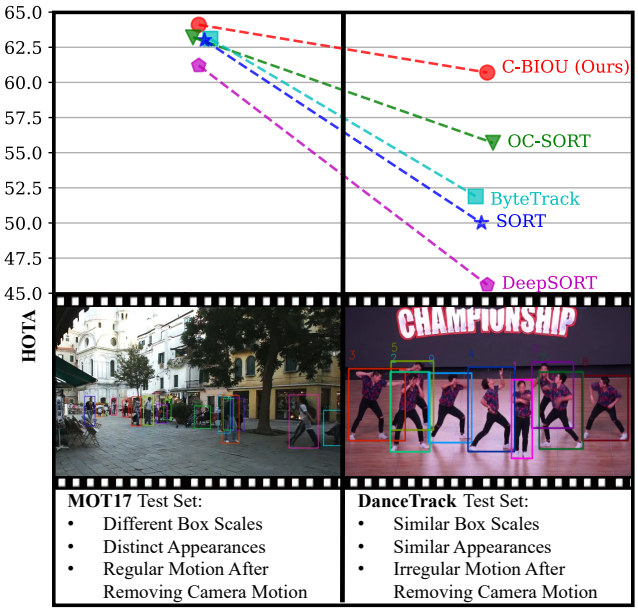

Figure 1: **Tracking performance on the test sets of MOT17 [21] and DanceTrack [26].** For a fair comparison, all methods are *online approaches* and use *the same detections* generated by YOLOX [13]. On the MOT17, our method has a similar HOTA score to other methods, whereas on the DanceTrack, our method increases the HOTA score by a remarkable margin compared to DeepSORT [31], SORT [5], ByteTrack [34], and OC-SORT [7].

that cover animals, group dancers, and sports players, may have indistinguishable appearances and irregular motions, which could cause existing MOT methods to fail. In particular, as shown in Fig. 1, several MOT methods [5, 31, 34, 7] that perform well on MOT17 [21], may experience a significant performance drop on the DanceTrack [26].

Why does the HOTA score drop significantly on the DanceTrack? **We presume that tracking failures are caused by two reasons: (i) The detections and tracks of identical objects do not overlap between adjacent frames (*e.g.*, due to the fast movement) and thus the tracking fails; (ii) After track initialization, unmatched tracks (*e.g.*, occluded objects) continue to update their**

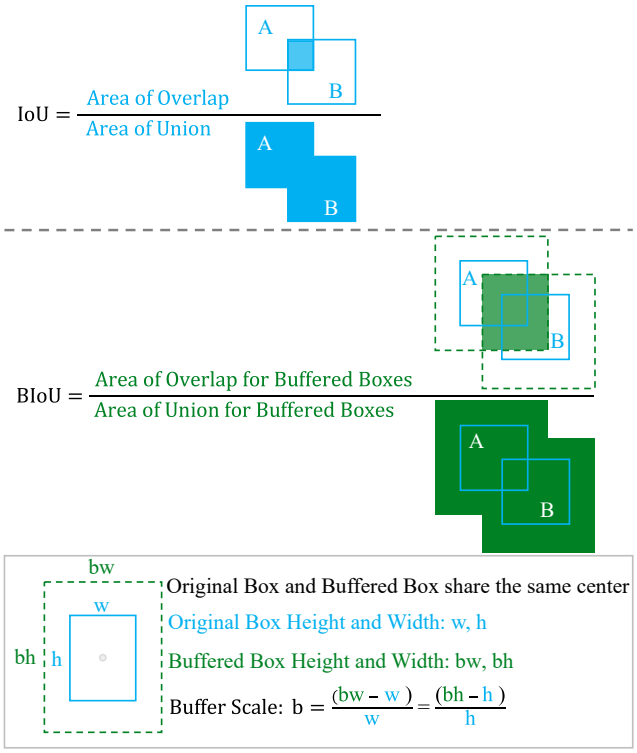

Figure 2: **Illustration of how Buffered IoU (BIoU) is calculated**. Our BIoU adds a buffer that is proportional to the original bounding box. It does not change the location center, scale ratio, and shape of the original bounding boxes but expands the original matching space.

geometric features for multiple frames, however, if their motion estimations are inaccurate (*e.g.*, due to a sudden acceleration or turning), they miss the matching opportunity when corresponding detections are available in subsequent frames. When the appearance of objects can be distinguished, appearance features could be employed to alleviate issues (i) and (ii), by matching cross-frame detections based on their appearance similarities. Nonetheless, when irregular motions are accompanied by indistinguishable appearances, most existing MOT solutions may not be able to perform a dependable tracking, so a new solution is desirable.

In this study, we propose a Cascaded-Buffered Intersection over Union (C-BIoU) tracker to track multiple objects that have irregular motions and indistinguishable appearances. Our BIoU (Fig. 2) is applied to alleviate issues (i) and (ii). Unlike the IoU, which only forms spatiotemporal similarities between overlapping detections and tracks, our BIoU constructs spatiotemporal similarities for originally non-overlapping detections and tracks if they are within the range of the buffers (Fig. 3). Because the buffers are proportional to the original detections and tracks, the BIoU does not change their location centers, scale ratios, and shapes but expands their matching space. With these properties,

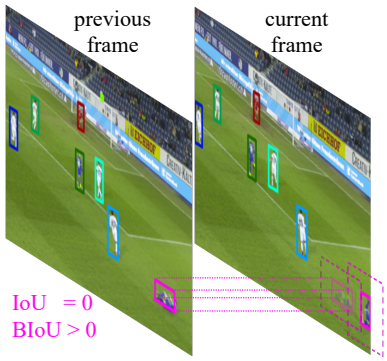

Figure 3: **An illustration of BIoU forms better cross-frame geometric consistency than IoU**. The bounding box of an identical object shares the same color. The magenta object has no overlapping detections between adjacent frames. Whether this is caused by the fast movement or incorrect motion estimation, our BIoU expands the matching space to reduce the miss matching.

our BIoU mitigates the effect of irregular motions in two aspects: one is to directly match identical but non-overlapping detections and tracks in adjacent frames, and the other is to compensate for the motion estimation bias in the matching space. Additionally, to reduce the risk of matching space overexpansion, we incorporate the BIoU into a cascaded matching scheme: first, alive tracks and detections are matched using a small buffer, and then, unmatched tracks and detections are matched again using a large buffer. To this end, our C-BIoU tracker could relieve mismatching caused by irregular motions and improve the tracking performance.

We report promising results on a variety of MOT datasets [9, 10, 1, 26] that focus on irregular motions and indistinguishable appearances. Compared with other strong MOT methods (*e.g.*, OC-SORT [7]), our C-BIoU tracker greatly improves the tracking performance, ranging from 2.6 to 7.2 in terms of the HOTA score [19]. Moreover, the C-BIoU tracker is the dominant component for our 2^{nd} place solution in the CVPR'22 SoccerNet MOT and the ECCV'22 MOTComplex DanceTrack challenges. Finally, we analyze the limitation of our C-BIoU tracker in ablation studies and discuss its application scopes.

2. Related Works

2.1. Appearance Consistency and Geometric Consistency in MOT

In MOT studies, appearance consistency and geometric consistency are two critical assumptions used for associating cross-frame detections. In general, the previous appearance of an identical object should be similar to its current appearance (*i.e.*, appearance consistency), and its previous location and shape added to its estimated motion should be approximate to its current location and shape (*i.e.*, geomet-

ric consistency).

In recent works, leveraging the appearance feature for MOT has achieved great success in conventional MOT datasets (*e.g*., MOT17 [21]). In particular, after transformers [28] have been introduced to MOT studies [27, 33, 20], the appearance similarity between cross-frame detections can be measured in a highly accurate manner, which leads to a good tracking performance. Nevertheless, the DanceTrack [26] study conducted experiments to demonstrate that appearance is not always reliable when tracking targets share a similar appearance. Other MOT datasets, such as SoccerNet [9, 10] and GMOT-40 [1], also reveal the challenge of real-world MOT tasks: tracking targets may look similar, which could fail MOT methods that achieved a state-of-the-art performance on conventional MOT datasets (*e.g*., MOT17 [21]).

Geometric matching can reduce the ambiguity caused by indistinguishable appearances. In general, the IoU is commonly used to measure geometric consistency [6, 5, 31, 30, 35, 34, 7]. The IoU scores, between detections and track predictions, are used to represent their cross-frame affinity. To estimate motions, Neural Networks [22] and Bayesian filters [2, 12] have been typically applied. While most MOT methods [5, 31, 30, 35, 34] apply the Kalman filter [15] due to its simplicity, OC-SORT [7] has enhanced the Kalman filter to handle crowded and occluded scenes. In real practice, however, motion modeling may not always be accurate. In some scenarios, for instance, soccer players and dancers may make irregular motions, which cause the motion estimation model to fail. Additionally, for a non-stationary camera, although image registration [8] can be used to calibrate camera movements, it is time-consuming, and the accuracy cannot be guaranteed. To alleviate these problems, we introduce a new geometric consistency measurement solution.

2.2. Geometric Consistency Measurement

When irregular motions are given, it is difficult to initialize and estimate the motion correctly, which may result in identical objects with no overlapping geometric features in adjacent frames. Because the IoU produces the same value of 0 for all non-overlapping geometric features (*i.e*., bounding boxes), using the IoU for geometric consistency measurement may fail tracking initialization and ongoing tracking. Thus, we propose a BIoU to expand the original matching space to measure the geometric consistency, which is robust to fast motions and motion estimation bias. Unlike the searching window in MOT [32], which applies the expanded bounding box as a spatial constraint, our BIoU takes the expanded bounding box as a matching feature. To some extent, using the GIoU [24] and DIoU [36] mitigates the same issue as our BIoU does, but we verified that our BIoU may generate better results under the same conditions (Sec. 4.3).

2.3. Cascaded Matching

After obtaining the cross-frame consistency measurements, matching (*i.e*., data association) is applied to correspond cross-frame detections. In addition to the cross-frame consistency, we can also employ other strategies to optimize the matching process. Cascaded matching is a commonly used approach in MOT studies: matching the confident and easy samples first, followed by ambiguous and difficult samples. For example, ByteTrack[34] matches confident detections earlier than unconfident detections, while DeepSORT [31] applies data association to recently matched tracks before earlier matched tracks. Since our BIoU changes the matching space, using a large buffer scale takes a higher risk of overexpansion than using a small buffer scale. We therefore integrate the BIoU and cascaded matching in our tracker (Fig. 4). We first match alive tracks and detections with a small buffer, and then match unmatched tracks and retained detections with a large buffer.

3. C-BIoU Tracker

The architecture of our Cascaded-Buffered IoU (C-BIoU) tracker is illustrated in Fig. 4. It is specifically designed to track multiple objects that have indistinguishable appearances and irregular motions. We inherit part of the track management from SORT [5] and propose our C-BIoU for geometric consistency measurement.

3.1. Tracking Pipeline

Our tracking pipeline follows the tracking-by-detection paradigm—the object detector and MOT framework are separately designed. Given a video, we apply the off-the-shelf object detector (*e.g*., YOLOX [13]) to generate bounding boxes at each frame. Our C-BIoU tracker then takes those bounding boxes as inputs to produce tracking results. Such a pipeline provides great flexibility to apply our C-BIoU tracker on arbitrary detections. In our experiments (Sec. 4.2), we also show that the similar pipeline [5, 31, 34, 7] yielded strong results on our target datasets.

3.2. Buffered IoU

The Buffered IoU (BIoU) is our main contribution in this work. As shown in Fig. 2, the BIoU simply adds buffers that are proportional to the original detections and tracks for calculating the IoU. Our BIoU retains the same location centers, scale ratios, and shapes of the original detections and tracks, but it expands the matching space to measure the geometric consistency. Let $\boldsymbol{o} = (x, y, w, h)$ denote an original detection and (x, y, w, h) be the top-left coordinate, width, and height of the detection, respectively. Suppose that the buffer scale is b, we have the buffered detection

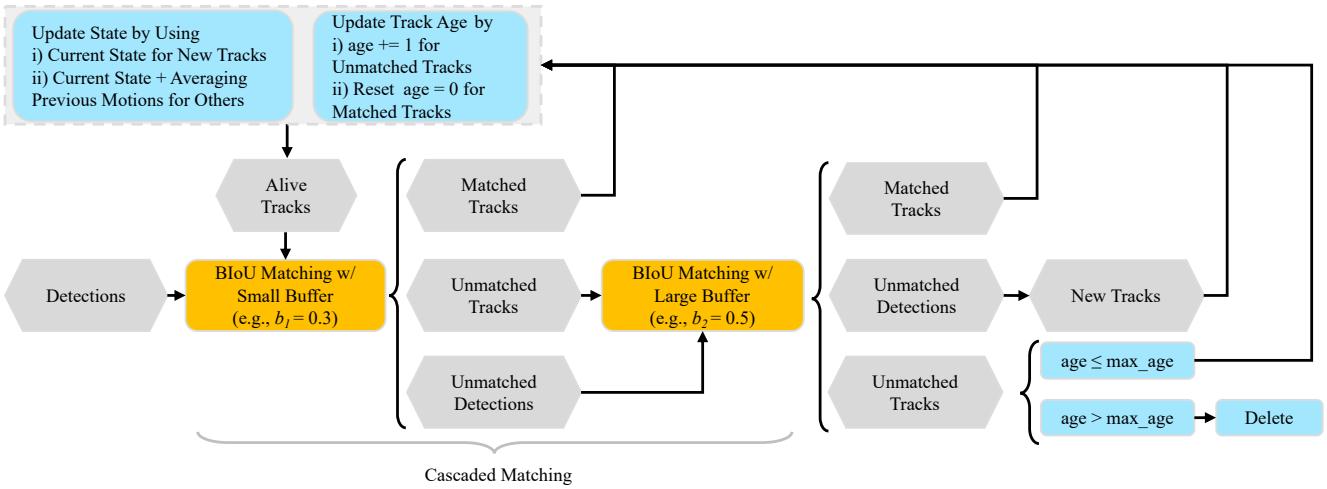

Figure 4: **Framework of Our C-BIoU Tracker.** Which initializes tracks from unmatched detections, applies the alive tracks to match new detections, and terminates a track when it has not been matched for a given amount of frames (*i.e.*, max_age). Two BIoUs, which respectively equip small and large buffers, are grouped into a cascaded matching. First, we match alive tracks and detections with the BIoU that has a small buffer (*i.e.*, b_1). Then, we continue to match unmatched tracks and detections with the BIoU that has a large buffer (*i.e.*, b_2). For the motion estimation, we simply average the speeds of recent frames to quickly respond to unpredictable motion changes.

as $\boldsymbol{o}_b = (x - bw, y - bh, w + bw, h + bh)$. To approach our cascaded matching, we apply grid research [3] to find the best combination of two buffer scales b_1 and b_2 on the training set, and then apply them to the validation set and test set. Since we have $b_1 < b_2$, when we search for the combination of b_1 and b_2 within the range of 0.1 to 0.7, the number of combinations is limited. Considering that the speed of our C-BIoU is fast (Table 2), the grid search takes an acceptable time.

3.3. Simple Motion Estimation

Unlike most MOT methods [5, 31, 30, 35, 34] that apply the Kalman filter [15] for state estimation, we simply average motions of recent frames to quickly respond to unpredictable motion changes. At frame t, suppose that a track has matched detections for more than n frames, after Δ unmatched frames, its track state $\boldsymbol{s}$ can be represented as $\boldsymbol{s}^{t+\Delta} = o^t + \frac{\Delta}{n-1}\sum_{i=t-n+1}^{t}(\boldsymbol{o}^i - \boldsymbol{o}^{i-1})$. The matched detections between frame $t-n$ to t are used to calculate motions and the average motion is applied to update the track state. We set $2 \leq n \leq 5$ by default in our experiments. The IoU score of buffered $\boldsymbol{s}_b^{t+\Delta}$ and $\boldsymbol{o}_b^{i+\Delta}$ is used for data association at the frame $t + \Delta$. Due to the simplicity of our approach, the overall tracking speed is increased for our C-BIoU tracker (Table 2).

3.4. Track Management

In an MOT framework, the function of track management is to decide how and when to initialize, update and terminate a track. We design our track management based on the mainstream solution introduced by SORT [5], which is also widely applied in other well-known MOT methods [31, 35, 30, 34, 7].

For the first frame, we initialize all detections as new tracks. In each track, the corresponding detection is recorded in memory. Without using the appearance information, a track may need at least two tracked frames to initialize its motion estimation. For a new track, therefore, we do not predict its motion; instead, we directly assign its recorded bounding box as its current track state. As new tracks have an age of 0, they are all alive tracks and can be used to match detections. For the next frame, we apply the BIoU with a small buffer scale b_1 to calculate the geometric affinity between detections and alive tracks. Based on the geometric affinity, linear assignment (*e.g.*, Hungarian algorithm [16]) is applied to associate tracks and detections.

After the first matching, some tracks and detections could be unmatched. Besides the newly appeared and disappeared objects, we assume that some objects may have an inconsistency between their detections and states of the track. This inconsistency could be caused by large irregular motions. To alleviate this issue, we apply BIoU with a large buffer scale b_2 for the second matching. The first and second BIoU matching form a cascaded matching. After the second matching, we create new tracks from the unmatched detections and terminate a track when it has not been matched for a given amount of frames (*i.e.*, max_age). We update the state of a track by adding estimated motions to its current state. Meanwhile, we also update the age of tracks. We increase the age for the unmatched tracks and reset the age to 0 for matched tracks. This age will be compared with the threshold max_age to determine whether a track should be terminated. We repeat this progress until all frames are processed.

Note that, we only propose a simple prototype to show how to use our C-BIoU in MOT. Depending on the needs, other MOT modules can be integrated with our C-BIoU to build a more powerful MOT framework.

4. Experiments

Our experiments consist of three parts. In Sec. 4.1, we present the details of our experimental dataset and evaluation metrics. Then, in Sec. 4.2, we demonstrate the effectiveness of our C-BIoU tracker by comparing its performance to state-of-the-art methods on four MOT datasets. Next, in Sec. 4.3, we perform ablation studies to investigate (1) how our BIoU, cascaded matching, and motion modeling contribute to our final results; (2) how our dominant parameters, as the buffer scales, affect the tracking performance; and (3) how detection noise influences our C-BIoU tracker and the corresponding limitation of our C-BIoU tracker.

4.1. Dataset and Evaluation Metrics

Datasets. Four public MOT datasets are used in our experiments. MOT17 [21] covers conventional tracking scenes: most tracking targets may have distinguishable appearances, and their motions could be regular after removing camera motions. DanceTrack [26], SoccerNet [9, 10], and GMOT-40 [1] introduce another kind of realistic tracking scenario, where tacking targets share a similar texture and have irregular motions (even after removing the camera motion). Besides, compared to MOT17, more frames are included in DanceTrack, SoccerNet, and GMOT-40, which helps us make a comprehensive analysis.

Evaluation Metrics. Although MOTA [4] used to be a dominant metric for the MOT evaluation, it may favor detection over association performance. To alleviate the limitation of MOTA, the HOTA metric [19] was proposed to provide a better trade-off between detection and association performance, and thus, it is the dominant metric for recent MOT evaluations. In our experiments, we select HOTA metrics (*i.e*., HOTA, DetA and AssA) [19], CLEAR metrics (*i.e*., MOTA) [4] and Identity metrics (*i.e*., IDF1) [25] to evaluate the tracking results from various perspectives. Among them, the HOTA score is our dominant metric.

Evaluation Approaches. To evaluate the test sets of MOT17 and DanceTrack, we submit the result to their official evaluation servers to obtain the evaluation feedback. Meanwhile, we utilize the ground truth of the DanceTrack validation set, SoccerNet test set, and GMOT-40 test set to perform evaluations with the TrackEval [19] evaluation script. In our experiments, we apply the default data splitting for MOT17, DanceTrack, SoccerNet, and GMOT-40.

4.2. Main Results

4.2.1 Comparisons Using Estimated Detections

Table 1 compares our C-BIoU tracker to mainstream MOT methods on the test sets of MOT17 [21] (private detections) and DanceTrack [26]. Each score is either from previous studies (*e.g*., DanceTrack [26]) or obtained by submitting the corresponding results to official evaluation servers. Note that, since the detection quality can significantly affect the overall tracking performance, for a fair comparison, methods in the bottom block use the same detections generated by YOLOX [13]. The YOLOX weights for the MOT17 and DanceTrack datasets are offered by ByteTrack [26] and OC-SORT [7], respectively. As methods in the top block may utilize better or worse detections than ours, we list them here for reference only.

On the MOT17 test set, our method has a similar HOTA score as other methods. As analyzed in previous work [26], the main bottleneck in MOT17 is detection other than tracking. On the DanceTrack test set, our method increases the HOTA score by a remarkable margin as compared to other methods. Although DeepSORT [31], SORT [5], and ByteTrack [34] can generate comparable results on the MOT17 test set, their tracking performance largely drops on the DanceTrack test set, where more complicated object movements and similar bounding box scales are included. Compared to the second-best method (*i.e*., OC-SORT [7]), which applies the IoU, GIoU, or DIoU for its matching, our C-BIoU tracker has increased the HOTA score by 4.9 to make the new state of the art. Through the above comparisons, we prove the effectiveness of our C-BIoU tracker on conventional MOT data (*i.e*., MOT17) and our target MOT data (*i.e*., DanceTrack) that covers complicated motions and indistinguishable appearance.

In addition to the HOTA gain, our C-BIoU tracker can increase the inference speed of tracking (w/o the detection part). Table 2 reports the inference speed on the test sets of MOT17 [21] and DanceTrack [26]. Our C-BIoU tracker leverages the average speed of recent frames other than Kalman filters for its motion estimation. Therefore, we reduce the computation cost for data format transformation and other calculations used in Kalman filters. On the MOT17 and DanceTrack datasets, our C-BIoU tracker almost doubles the speed of OC-SORT [7] and is much faster than other trackers. These results reveal that our C-BIoU tracker is a practical solution for real-world applications.

4.2.2 Comparisons Using Oracle Detections

To focus only on the tracking, we perform experiments using oracle detections from the DanceTrack validation set [26], SoccerNet test set [9, 10], and GMOT-40 test set [1]. The results in Table 3 indicate that our C-BIoU tracker can significantly surpass the other methods [5, 31, 34, 26, 7], improving the tracking performance ranging

Table 1: **Results on the test sets of MOT17 [21] and DanceTrack [26].** For a fair comparison, methods in the bottom block use the same detections generated by YOLOX [13]. On MOT17, our method has a similar HOTA score to other methods, whereas, on the DanceTrack, our method increases the HOTA score with a remarkable margin.

Tracker	MOT17 Test Set					DanceTrack Test Set				
	HOTA↑	DetA↑	AssA↑	MOTA↑	IDF1↑	HOTA↑	DetA↑	AssA↑	MOTA↑	IDF1↑
Using Other Detections										
FairMOT [35]	59.3	60.9	58.0	73.7	72.3	39.7	66.7	23.8	82.2	40.8
QDTrack [23]	53.9	55.6	52.7	68.7	66.3	45.7	72.1	29.2	83.0	44.8
TransTrack [27]	54.1	61.6	47.9	75.2	63.5	45.5	75.9	27.5	88.4	45.2
MOTR [33]	57.2	58.9	55.8	71.9	68.4	54.2	73.5	40.2	79.7	51.5
GTR [37]	59.1	61.6	57.0	75.3	71.5	48.0	72.5	31.9	84.7	50.3
Using Detections Generated by YOLOX-x [13] with Input Size of [800, 1440]										
DeepSORT [31]	61.2	63.1	59.7	78.0	74.5	45.6	71.0	29.7	87.8	47.9
SORT [5]	63.0	64.2	62.2	80.1	78.2	50.0	75.5	33.2	90.4	52.0
ByteTrack [34]	63.1	64.5	62.0	80.3	77.3	51.9	80.1	33.8	90.9	52.0
OC-SORT [7]	63.2	63.2	63.2	78.0	77.5	55.7	**81.7**	38.3	**92.0**	54.6
C-BIoU Tracker	**64.1**	**64.8**	**63.7**	**81.1**	**79.7**	**60.6**	81.3	**45.4**	91.6	**61.6**

Table 2: **Comparison of the tracking inference speed (w/o the detection part) using an Intel Xeon Silver 4216 CPU**. The unit is FPS (Frames Per Second). Because our C-BIoU utilizes the average speed of recent frames other than the Kalman filter for its motion estimation, it is faster than other trackers. Note that, the speed of tracker is proportional to the number of tracking objects, and when the number of objects increases, the speed of the tracker drops.

Tracker	MOT17	DanceTrack
SORT [5]	144	271
ByteTrack [34]	118	207
OC-SORT [7]	185	341
C-BIoU Tracker	**361**	**680**

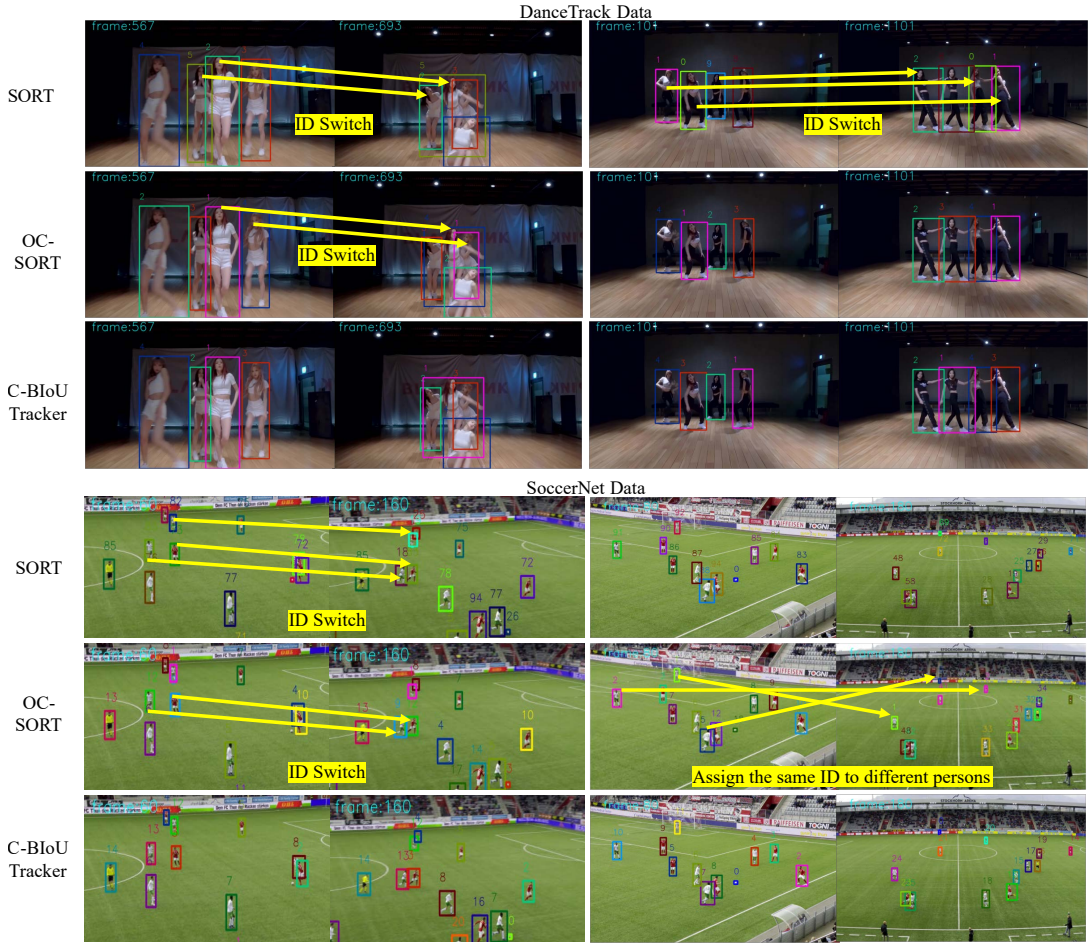

Figure 5: **Example results on the DanceTrack validation set [26] and SoccerNet test set [9, 10].** Our C-BIoU tracker generates fewer tracking errors than SORT [5] and OC-SORT [7].

from 2.6 to 7.2 in terms of the HOTA score. To obtain a more comprehensive look at the tracking performance, we plot the tracking results on multiple datasets for SORT [5], OC-SORT [7], and our C-BIoU tracker in Fig. 5.

Although we achieve the best performance on the three datasets, our tracking results are still imperfect even using

Table 3: **Comparisons on the DanceTrack validation set [26], SoccerNet test set [9, 10], and GMOT-40 test set [1]**. Where "App." and "Mo." represent the appearance feature and motion estimation, respectively.

Tracker	HOTA↑	DetA↑	AssA↑	MOTA↑	IDF1↑
DanceTrack Validation Set [26]. Using Oracle Detections.					
DanceTrack (IoU) [26]	72.8	**98.9**	53.6	98.7	63.5
DanceTrack (IoU+Mo.) [26]	69.4	87.9	54.8	99.4	71.3
DanceTrack (App.) [26]	59.7	82.5	43.2	97.2	60.5
DanceTrack (IoU+Mo.+App.) [26]	68.0	97.7	47.4	97.9	58.7
DeepSORT [31]	66.8	86.1	51.8	97.4	68.3
SORT [5]	67.6	86.6	52.8	98.1	69.6
OC-SORT [7]	79.1	97.7	64.0	**99.6**	76.1
C-BIoU Tracker	**81.7**	97.6	**68.4**	99.3	**80.5**
SoccerNet Test Set [9, 10]. Using Oracle Detections.					
ByteTrack [34] (reported by [9])	71.5	84.3	60.7	94.6	-
DeepSORT [31] (reported by [9])	69.6	82.6	58.7	94.8	-
SORT [5]	74.7	87.2	64.0	96.1	75.6
OC-SORT [7]	82.0	98.6	67.9	98.3	76.3
C-BIoU Tracker	**89.2**	**99.4**	**80.0**	**99.4**	**86.1**
GMOT-40 Test Set [1]. Using Oracle Detections.					
DeepSORT [31]	86.4	87.9	84.9	94.2	88.6
SORT [5]	87.8	90.9	84.8	97.6	89.6
OC-SORT [7]	92.4	99.3	86.0	98.5	90.0
C-BIoU Tracker	**96.4**	**99.7**	**93.2**	**99.6**	**95.6**

Table 4: **Ablation experiments on the DanceTrack validation set [26], SoccerNet test set [9, 10], and GMOT-40 test set [1]**. Where "C.M." and "Mo." represent the cascaded matching and motion estimation, respectively. We remove the cascaded matching and motion estimation in Fig. 4 to construct a unified framework for the IoU, GIoU [24], DIoU [36], and BIoU. The best results obtained by tuning the parameters are reported. Our BIoU performs better than the GIoU and DIoU. Using the C-BIoU setting is better than that using the BIoU alone. The motion estimation contributes to better HOTA scores.

Tracker	C.M.	Mo.	HOTA↑	DetA↑	AssA↑	MOTA↑	IDF1↑
DanceTrack Validation Set [26]. Using Oracle Detections.							
IoU Tracker	✗	✗	76.6	97.5	60.2	99.2	73.6
GIoU Tracker	✗	✗	77.1	**97.6**	60.9	99.2	74.0
DIoU Tracker	✗	✗	75.1	97.0	58.2	99.2	72.9
BIoU Tracker	✗	✗	80.0	97.5	65.7	**99.3**	78.2
C-BIoU Tracker	✓	✗	80.2	97.5	65.9	**99.3**	79.3
C-BIoU Tracker	✓	✓	**81.7**	**97.6**	**68.4**	**99.3**	**80.5**
SoccerNet Test Set [9, 10]. Using Oracle Detections.							
IoU Tracker	✗	✗	81.9	99.4	67.5	**99.8**	75.7
GIoU Tracker	✗	✗	79.8	**99.7**	63.8	97.8	73.4
DIoU Tracker	✗	✗	84.3	**99.7**	71.2	99.2	79.9
BIoU Tracker	✗	✗	87.7	**97.7**	77.1	99.4	83.0
C-BIoU Tracker	✓	✗	88.9	99.5	79.4	99.5	85.2
C-BIoU Tracker	✓	✓	**89.2**	99.4	**80.0**	99.4	**86.1**
GMOT-40 Test Set [1]. Using Oracle Detections.							
IoU Tracker	✗	✗	93.0	99.6	86.8	98.1	90.1
GIoU Tracker	✗	✗	93.4	**99.8**	87.4	98.5	90.2
DIoU Tracker	✗	✗	93.6	99.7	87.8	99.2	91.7
BIoU Tracker	✗	✗	96.2	99.5	93.0	**99.6**	95.4
C-BIoU Tracker	✓	✗	96.3	99.7	93.1	**99.6**	95.5
C-BIoU Tracker	✓	✓	**96.4**	99.7	**93.2**	**99.6**	**95.6**

oracle detections. Therefore, in the current research, it is useful to construct baselines using oracle detections and focus on improving the data association performance. We hope our baselines can motivate related research.

4.3. Ablation Experiments

We perform ablation studies to investigate the effect of individual modules and buffer scales in our C-BIoU tracker, as well as the effect of noisy detections.

4.3.1 Effect of Each Module in the C-BIoU Tracker

Table 4 shows the influence of each module in our C-BIoU tracker. In detail, we present the following analysis.

Effect of the BIoU. As a comparison, we apply the BIoU matching only once and remove the motion estimation in Fig. 4 to construct the BIoU tracker. Using the same framework, the tracker equipped with BIoU achieves a higher HOTA score than other trackers equipped with IoU, GIoU [24], or DIoU [36]. Although the GIoU and DIoU can incorporate non-overlapping boxes for geometric consistency measurement, they may not generate comparable results as our BIoU does.

Effect of Integrating Cascaded Matching and the BIoU. On the DanceTrack and GMOT-40, integrating cascaded matching and BIoU can slightly improve the performance as compared to using BIoU alone, with a HOTA gain of 0.2 and 0.1, respectively. While on SoccerNet, the improvement from integrating cascaded matching and the BIoU is more significant, with a HOTA gain of 1.2. In the SoccerNet dataset, since the non-stationary camera can add extremely fast motion to objects, the use of cascade matching is more robust in this case.

Effect of the Motion Estimation. According to the results, motion estimation plays an important role in our C-BIoU tracker. Since our BIoU can compensate the matching space for incorrect motion estimation, using a simple motion estimation (*i.e.*, averaging previous motions) yields better HOTA scores than that without using motion estimation.

4.3.2 Effect of Buffer Scales in the C-BIoU Tracker

In our C-BIoU tracker, the buffer scales b_1 and b_2 are critical hyperparameters. Here, we perform ablation studies to investigate how buffer scales affect the tracking performance. On the DanceTrack validation set [26], we form the combination of b_1 and b_2 ranging from 0.1 to 0.7 and evaluate their tracking performance. Since we have $b_1 < b_2$, we only need to check 21 combinations. As shown in Fig. 6, the combination of $[0.3, 0.4]$ gives the maximum HOTA score. In real practice, we perform a similar approach to select the best combination on the training dataset and apply them to the test dataset. Note that, although the variation of buffer scales affects the tracking performance remarkably, using the IoU tracker can only achieve a HOTA score of 76.6,

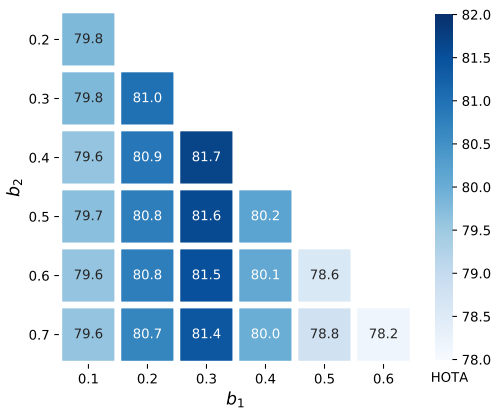

Figure 6: **Results of applying various buffer-scale combinations on the DanceTrack validation set [26].** For buffer scales b_1 and b_2, since we have $b_1 < b_2$, we only check the lower triangle of the combination matrix.

which is lower than using any of the above buffer combinations.

4.3.3 Effect of the Detection Noise

We have shown the superiority of our C-BIoU tracker in the previous experiments, however, we need to discuss about its limitations. Accordingly, we conduct the following analysis.

In the previous experiments, our C-BIoU tracker significantly outperforms other MOT methods when using either high-quality detections generated by YOLOX [13] or oracle detections. Nonetheless, assuming that we only have low-quality detections, the robustness of our C-BIoU tracker needs to be studied. We inject noise (*i.e.*, False Negatives and False Positives) to the oracle detections of the DanceTrack validation set [26] and form noisy detections that have quantitatively defined noise ratios. To inject detection noise, we first remove detections to generate False Negatives, and then add detections to non-target locations to form False Positives. Both of them have the same ratios.

The results in Table 5 reveal the influence of noisy detections on the tracking performance by considering noise ratios together. To date, such an ablation study had not been taken into account in existing studies. When the noise ratio is not higher than 20%, our C-BIoU tracker can maintain the best performance. However, a higher noise ratio, such as 40%, could lead to a worse performance of our C-BIoU tracker than the normal IoU tracker. The result is attributed to low-ratio noisy detections, which avoids the overlapping of the track and detection of an object in a *short* interval of frames. Therefore, using BIoU matching to expand the matching space can result in more samples being correctly matched than IoU matching. However, for high-ratio noisy detections, the track and detection of an object do not overlap in a *large* interval of frames. Consequently, both IoU matching and BIoU matching may generate tracking errors. In addition, the expansion of the matching space by BIoU leads to more aggressive matching, which increases the risk of missed matches with False Positives. For these reasons, the robustness of our C-BIoU tracker decreases when extremely noisy detections are given. Fortunately, as reported in previous works [26, 9, 1], high-quality detections can be obtained in our target MOT datasets, since the similar appearance may ease the object detection. Thus, our C-BIoU tracker is applicable to real-world applications despite its limitations.

Table 5: **The influence of the detection quality.** We inject different levels of noises to the oracle detections of the DanceTrack validation set [26] to quantitatively investigate the influence of detection quality. IoU tracker and OC-SORT [7] are used as baselines. We apply IoU matching only once in Fig. 4 to construct the IoU tracker.

Noise Ratio	Tracker	HOTA↑	DetA↑	AssA↑	MOTA↑	IDF1↑
0%	OC-SORT [7]	79.1	**97.7**	64.0	**99.6**	76.1
	IoU Tracker	76.6	97.5	60.2	99.2	73.6
	C-BIoU Tracker	**81.7**	97.6	**68.4**	99.3	**80.5**
20%	OC-SORT [7]	61.4	78.3	48.1	79.3	65.3
	IoU Tracker	57.6	**79.5**	41.7	**81.7**	59.6
	C-BIoU Tracker	**62.3**	78.3	**49.5**	79.2	**66.0**
40%	OC-SORT [7]	28.0	40.4	19.4	41.4	34.3
	IoU Tracker	**38.3**	**58.6**	**25.0**	**60.4**	**40.8**
	C-BIoU Tracker	29.2	58.0	14.7	57.7	29.1

5. Conclusion and Limitation Discussion

We present a novel Cascaded-Buffered IoU (C-BIoU) tracker to track multiple objects that have indistinguishable appearances and irregular motions. Experiments are conducted on related MOT datasets, and our C-BIoU tracker outperforms most existing methods by a notable margin. These results suggest that our C-BIoU tracker is generalizable and promising for tracking multiple objects with indistinguishable appearances and irregular motions. The good performance of our C-BIoU tracker can be attributed to its buffered matching space, which mitigates the effect of irregular motions in two aspects: one is to directly match identical but non-overlapping detections and tracks in adjacent frames, and the other is to compensate for the motion estimation bias in the matching space.

As a limitation, our C-BIoU tracker may not be robust to extremely noisy detections (Sec. 4.3.3). However, with advancements in object detection, existing studies hint that good detections can be obtained in most MOT tasks. In addition, for other applications such as semi-automatic MOT annotations (*e.g.*, [11, 29]), human factors are introduced to correct detections before tracking. Hence, our C-BIoU tracker remains a capable solution for real-world applications due to its simplicity, fast speed, and good tracking performance.

References

[1] Hexin Bai, Wensheng Cheng, Peng Chu, Juehuan Liu, Kai Zhang, and Haibin Ling. Gmot-40: A benchmark for generic multiple object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6719–6728, 2021.

[2] Yaakov Bar-Shalom, Thomas E Fortmann, and Peter G Cable. Tracking and data association, 1990.

[3] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2), 2012.

[4] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008:1–10, 2008.

[5] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *IEEE international conference on image processing*, pages 3464–3468, 2016.

[6] Erik Bochinski, Volker Eiselein, and Thomas Sikora. High-speed tracking-by-detection without using image information. In *International Workshop on Traffic and Street Surveillance for Safety and Security at IEEE AVSS 2017*, Lecce, Italy, Aug. 2017.

[7] Jinkun Cao, Xinshuo Weng, Rawal Khirodkar, Jiangmiao Pang, and Kris Kitani. Observation-centric sort: Rethinking sort for robust multi-object tracking. *arXiv preprint arXiv:2203.14360*, 2022.

[8] Anthony Cioppa, Adrien Deliege, Floriane Magera, Silvio Giancola, Olivier Barnich, Bernard Ghanem, and Marc Van Droogenbroeck. Camera calibration and player localization in soccernet-v2 and investigation of their representations for action spotting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 4537–4546, June 2021.

[9] Anthony Cioppa, Silvio Giancola, Adrien Deliege, Le Kang, Xin Zhou, Zhiyu Cheng, Bernard Ghanem, and Marc Van Droogenbroeck. Soccernet-tracking: Multiple object tracking dataset and benchmark in soccer videos. *arXiv preprint arXiv:2204.06918*, 2022.

[10] Adrien Deliege, Anthony Cioppa, Silvio Giancola, Meisam J Seikavandi, Jacob V Dueholm, Kamal Nasrollahi, Bernard Ghanem, Thomas B Moeslund, and Marc Van Droogenbroeck. Soccernet-v2: A dataset and benchmarks for holistic understanding of broadcast soccer videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4508–4519, 2021.

[11] Jaime B Fernandez, GM Venkatesh, Dian Zhang, Suzanne Little, and Noel E O'Connor. Semi-automatic multi-object video annotation based on tracking, prediction and semantic segmentation. In *2019 International Conference on Content-Based Multimedia Indexing (CBMI)*, pages 1–4. IEEE, 2019.

[12] Pierre F Gabriel, Jacques G Verly, Justus H Piater, and André Genon. The state of the art in multiple object tracking under occlusion in video sequences. In *Advanced Concepts for Intelligent Vision Systems*, pages 166–173. Citeseer, 2003.

[13] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021.

[14] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012.

[15] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960.

[16] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[17] Jessy Lauer, Mu Zhou, Shaokai Ye, William Menegas, Steffen Schneider, Tanmay Nath, Mohammed Mostafizur Rahman, Valentina Di Santo, Daniel Soberanes, Guoping Feng, et al. Multi-animal pose estimation, identification and tracking with deeplabcut. *Nature Methods*, 19(4):496–504, 2022.

[18] Shuai Li, Yu Kong, and Hamid Rezatofighi. Learning of global objective for network flow in multi-object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8855–8865, June 2022.

[19] Jonathon Luiten, Aljosa Osep, Patrick Dendorfer, Philip Torr, Andreas Geiger, Laura Leal-Taixé, and Bastian Leibe. Hota: A higher order metric for evaluating multi-object tracking. *International Journal of Computer Vision*, pages 1–31, 2020.

[20] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixe, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. *arXiv preprint arXiv:2101.02702*, 2021.

[21] Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016.

[22] Anton Milan, Seyed Hamid Rezatofighi, Anthony R. Dick, Ian D. Reid, and Konrad Schindler. Online multi-target tracking using recurrent neural networks. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 4225–4232. AAAI Press, 2017.

[23] Jiangmiao Pang, Linlu Qiu, Xia Li, Haofeng Chen, Qi Li, Trevor Darrell, and Fisher Yu. Quasi-dense similarity learning for multiple object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 164–173, 2021.

[24] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 658–666, 2019.

[25] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *ECCV*, pages 17–35. Springer, 2016.

[26] Peize Sun, Jinkun Cao, Yi Jiang, Zehuan Yuan, Song Bai, Kris Kitani, and Ping Luo. Dancetrack: Multi-object tracking in uniform appearance and diverse motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 0–10, June 2022.

[27] Peize Sun, Yi Jiang, Rufeng Zhang, Enze Xie, Jinkun Cao, Xinting Hu, Tao Kong, Zehuan Yuan, Changhu Wang, and Ping Luo. Transtrack: Multiple-object tracking with transformer. *arXiv preprint arXiv:2012.15460*, 2020.

[28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[29] Paul Voigtlaender, Michael Krause, Aljosa Osep, Jonathon Luiten, Berin Balachandar Gnana Sekar, Andreas Geiger, and Bastian Leibe. Mots: Multi-object tracking and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7942–7951, 2019.

[30] Zhongdao Wang, Liang Zheng, Yixuan Liu, Yali Li, and Shengjin Wang. Towards real-time multi-object tracking. In *European Conference on Computer Vision*, pages 107–122. Springer, 2020.

[31] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *IEEE international conference on image processing*, pages 3645–3649, 2017.

[32] Fan Yang, Zheng Wang, Yang Wu, Sakriani Sakti, and Satoshi Nakamura. Tackling multiple object tracking with complicated motions—re-designing the integration of motion and appearance. *Image and Vision Computing*, 124:104514, 2022.

[33] Fangao Zeng, Bin Dong, Tiancai Wang, Cheng Chen, Xiangyu Zhang, and Yichen Wei. Motr: End-to-end multiple-object tracking with transformer. *arXiv preprint arXiv:2105.03247*, 2021.

[34] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. *arXiv preprint arXiv:2110.06864*, 2021.

[35] 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.

[36] Zhaohui Zheng, Ping Wang, Wei Liu, Jinze Li, Rongguang Ye, and Dongwei Ren. Distance-iou loss: Faster and better learning for bounding box regression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12993–13000, 2020.

[37] Xingyi Zhou, Tianwei Yin, Vladlen Koltun, and Phillip Krähenbühl. Global tracking transformers. *arXiv preprint arXiv:2203.13250*, 2022.