Filomat 39:16 (2025), 5685–5702
https://doi.org/10.2298/FIL2516685S

Published by Faculty of Sciences and Mathematics,
University of Niš, Serbia
Available at: http://www.pmf.ni.ac.rs/filomat

BoostTrack++: using tracklet information to detect more objects in multiple object tracking

Vukašin Stanojević[a,*], Branimir Todorović[a]

[a]*Department of Computer Science, Faculty of Sciences and Mathematics, University of Niš, Serbia Višegradska 33, 18000 Niš*

Abstract. Multiple object tracking (MOT) depends heavily on selection of true positive detected bounding boxes. However, this aspect of the problem is mostly overlooked or mitigated by employing two-stage association and utilizing low confidence detections in the second stage. Recently proposed BoostTrack attempts to avoid the drawbacks of multiple stage association approach and uses low-confidence detections by applying detection confidence boosting. In this paper, we identify the limitations of the confidence boost used in BoostTrack and propose a method to improve its performance. To construct a richer similarity measure and enable a better selection of true positive detections, we propose to use a combination of shape, Mahalanobis distance and novel soft BIoU similarity. We propose a soft detection confidence boost technique which calculates new confidence scores based on the similarity measure and the previous confidence scores, and we introduce varying similarity threshold to account for lower similarity measure between detections and tracklets which are not regularly updated. The proposed additions are mutually independent and can be used in any MOT algorithm.

Combined with the BoostTrack+ baseline, our method achieves near state of the art results on the MOT17 dataset and new state of the art HOTA and IDF1 scores on the MOT20 dataset.

The source code is available at: https://github.com/vukasin-stanojevic/BoostTrack.

1. Introduction

Multiple object tracking (MOT) is an important and active topic in computer vision. The main applications include human-robot interaction [46], autonomous driving [44] and surveillance [19], but it can also be applied to analyse sports videos [8], track animals [54] and even in medicine [24]. Given a video with multiple objects of interest (e.g. pedestrians), the aim is to construct a trajectory for each object. More specifically, for each frame, every object of interest should be detected and assigned an ID, which should not change during the video even if the object is not present in every frame (e.g. it can be occluded). MOT can be solved offline by processing the entire video, or online by processing one frame at a time without considering the future frames. Online MOT solutions have wider applications and can be used for real-time tracking in autonomous driving or surveillance systems. Among the online methods, tracking by detection

2020 *Mathematics Subject Classification*. Primary 68T20; Secondary 68T45, 68U10.
Keywords. multi-object tracking, detection confidence, data association.
Received: 15 July 2024; Revised: 15 January 2025; Accepted: 25 June 2025
Communicated by Marko Petković
* Corresponding author: Vukašin Stanojević
Email addresses: vukasin.stanojevic@pmf.edu.rs (Vukašin Stanojević), branimir.todorovic@pmf.edu.rs (Branimir Todorović)
ORCID iDs: https://orcid.org/0000-0002-5439-1057 (Vukašin Stanojević), https://orcid.org/0000-0002-1792-1311 (Branimir Todorović)

(TBD) methods show the best performance. In TBD paradigm, MOT is solved in two steps: detection, which outputs a set of detected bounding boxes; and association in which detected bounding boxes should be associated (matched) with currently tracked objects. Hungarian algorithm [21] is a typical choice for matching between newly detected bounding boxes and existing tracklets, i.e. current states of tracked trajectories. The cost matrix used by the algorithm can be constructed by combining different similarity measures such as intersection over union (IoU), Mahalanobious distance [31] or appearance similarity (cosine similarity between visual embedding vectors).

Before constructing a cost matrix, some method of filtering out false positive detections should be applied. This is usually achieved by discarding detections with confidence scores below a specified threshold. This simple logic results in discarding some true positive detections also. To mitigate this, ByteTrack [56] employed a two-stage association in which low-confidence detections and unmatched tracklets are used in the second association stage. Two-stage association became the standard in TBD MOT (e.g. standard benchmark methods that adopted two-stage association include [2, 11, 30]). However, multiple stage association methods can introduce identity switches (IDSWs) [42].

Recently, BoostTrack [43] used a one-stage association combined with strategies to increase (boost) the detection confidence of some detected bounding boxes before discarding detections with low confidence scores. In confidence boost based on detection of likely objects (DLO), it used intersection over union (IoU) between all detected bounding boxes and all tracklets to discover the low-confidence detections with high overlap with existing tracklet. BoostTrack boosted the confidence score of these detections assuming they correspond to currently tracked objets, which improved the tracking performance in crowded scenes with frequent occlusions (namely, on the MOT20 dataset). However, the method resulted in an increased number of new IDs, and identity switches, which indicates a more sophisticated method is required. In this paper, we extend the idea of DLO confidence boost.

In [43], shape similarity and Mahalanobis distance are used to improve (”boost”) the similarity measure to reduce ambiguity arising from using IoU only. This improved association performance. However, if a particular similarity measure improves association, it should also improve the selection of low-confidence true positive detections. We extend the idea of buffered IoU (BIoU) from [50] and create a novel soft BIoU similarity measure, which we use jointly with Mahalanobis distance and shape similarity from [43] to discover true positive low-confidence detections.

DLO confidence boost used in [43] uses only the similarity (IoU) between a detection and a tracklet to compute the boosted similarity. If similarity is above the certain threshold, the detection will be used. This means that the detection whose confidence is close to zero and the detection that has a confidence score close to the threshold (slightly below it) will be equally treated in the DLO boost procedure, i.e., both require the same high similarity to get the boost. We introduce a soft detection confidence boost which takes into account the original detection confidence score and solves the described problem.

Another weakness of the BoostTrack DLO confidence boost technique is that it treats all tracklets equally when deciding if IoU between a tracklet and a detection is high enough. However, the tracklets that are not recently updated (i.e. not matched with a detected bounding box for multiple subsequent frames due to occlusion, detector or association failure) usually have relatively low IoU with the corresponding detections once they are matched. Setting a threshold for ”high enough” similarity should be tracklet specific and depend on the number of frames since the last time the tracklet was updated. For example, if a tracklet was not matched for 30 frames, IoU of 0.8 can be considered very high. We attempt to solve this problem by introducing varying threshold based on the number of frames since the last update of a given tracklet.

Each of the proposed modifications is independent to the others and can be combined and used in any TBD MOT algorithm. We perform a detailed ablation study on MOT17 [33] and MOT20 [9] validation sets to show the effectiveness of each component. We successfully reduced the number of new IDs and identity switches, while not only retaining baseline tracking performance but surpassing it.

We name the MOT system that combines BoostTrack+ baseline with the proposed additions BoostTrack++. Among online trackers, BoostTrack++ ranks first in HOTA score on the MOT17 test set. On MOT20, BoostTrack++ ranks first in HOTA and IDF1 scores among all trackers (see Figure 1).

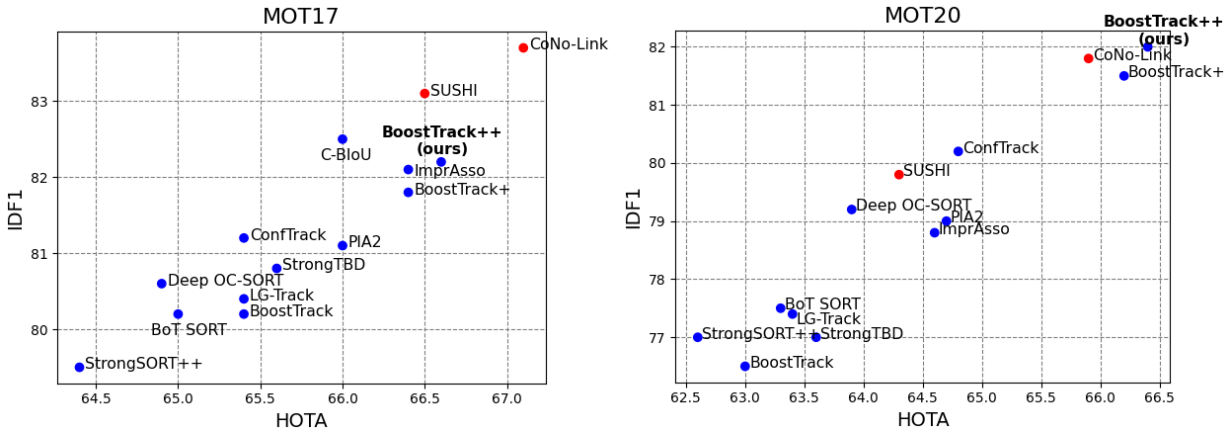

Figure 1: Results of HOTA and IDF1 metrics on MOT17 (left) and MOT20 (right) test sets. We display offline methods as red circles.

In summary, we make the following contributions:

- We introduce a novel soft buffered IoU (soft BIoU) similarity measure.
- We use the combination of Mahalanobis distance, shape and soft BIoU similarity to calculate the more sophisticated similarity measure for detecting the likely objects.
- We use soft detection confidence boost which uses the original detection confidence score for calculating the boosted confidence.
- We introduce varying boost threshold based on the number of time steps since the last update of a given tracklet.
- Added to the BoostTrack+ baseline, our BoostTrack++ method achieves near state of the art results on MOT17 and sets new state of the art on MOT20 dataset.

The rest of the paper is structured as follows: In section 2, we give a brief introduction to the TBD MOT approach and cover the multiple-stage association methods, which is the standard for dealing with low-confidence detections. Since our method extends the BoostTrack, we give a short overview of BoostTrack in subsection 2.3. We discuss the drawbacks of the DLO confidence boost from BoostTrack in section 3. Section 4 covers the proposed method. First, in subsection 4.1 we introduce our novel soft buffered IoU similarity measure. In the next subsection, we introduce the average similarity measure for the detection confidence boost, after which, in subsections 4.3 and 4.4 we introduce our soft detection confidence boost and varying threshold techniques, respectively. In subsection 4.5 we introduce an algorithm which combines all of the proposed additions. In section 5, we provide the results of the ablation study and compare the proposed method with other methods on MOT17 and MOT20 test sets. We conclude the paper with section 6.

2. Preliminaries

2.1. Tracking by detection

One of the dominant approaches in MOT is to follow the TBD paradigm. In TBD, the MOT problem is solved frame-by-frame by performing two steps of processing for every frame: detection and association. Given a frame, the detector model (e.g. YOLOX [17]) outputs the set of detected bounding boxes $D = \{D_1, D_2, \ldots, D_n\}$ with the corresponding confidence scores $c_d = \{c_{d_1}, c_{d_2}, \ldots, c_{d_n}\}$. Tracking module (e.g. Kalman filter [12]) is needed to predict the state of currently tracked objects, i.e. tracklets $T = \{T_1, T_2, \ldots T_m\}$, based on the past states (one state in the case of Kalman filter).

If Kalman filter is used, the state of the object usually consists of object location and size, and the corresponding velocities. In this paper, we use the same settings for the Kalman filter as in [43].

Linear state space model of tracking is given as:

$$\begin{aligned} x_k &= F \cdot x_{k-1} + q_k \quad \text{(dynamic equation)}, \\ y_k &= H \cdot x_k + r_k \quad \text{(observation equation)}, \end{aligned} \tag{1}$$

where q_k and r_k are process and observation noises, respectively (we use the same noise setting as in [43]). The state x_k, state transition matrix F, and observation matrix H are given by:

$$\mathbf{x_k} = [u, v, h, r, \dot{u}, \dot{v}, \dot{h}, \dot{r}]^\top, \quad F = \begin{bmatrix} I_4 & I_4 \\ \mathbf{0_{4\times4}} & I_4 \end{bmatrix}, \quad H = \begin{bmatrix} I_4 & \mathbf{0_{4\times4}} \end{bmatrix}, \tag{2}$$

where u, v, h and r represent the coordinates of the object center, the height of the object and the ratio of the object's width and height, respectively. By $\dot{u}, \dot{v}, \dot{h}, \dot{r}$ we denote their corresponding velocities. When a tracklet is associated with detected bounding box D_i, that bounding box is used as the observation in Kalman update step.

Tracklets and detections can be matched using the Hungarian algorithm [21]. The cost matrix C for the matching can be constructed simply as $C = -1 \cdot S(D, T)$, where $S(D, T)$ is the similarity matrix between the detections D and tracklets T. In the simplest case, S = IoU.

Looking at TBD logic with more details, before constructing the cost matrix, a selection of "reliable" detected bounding boxes is required, i.e. false positive detections should be filtered out. The simplest method is thresholding - deciding which detections will be used based on whether the confidence score is greater than a specified threshold value τ. The matching between a detected bounding box D_i and a tracklet T_j is admissible if $S(D_i, T_j) > \tau_S$, where τ_S represents a minimal required similarity (e.g. $\tau_S = 0.3$). Unmatched detections are used to initialize new tracklets[1)].

2.2. Multiple-stage association methods

Not all low-confidence detections are false positives. The dominant approach to using low-confidence detections (and utilising the true positive detections that would have been discarded otherwise) is using a two-stage association which was first introduced in ByteTrack [56]. ByteTrack uses thresholding but does not discard low-confidence detections. Instead, the unmatched tracklets are matched with low-confidence detections in the second association stage. Followed by ByteTrack, the two-stage association approach became the standard in TBD MOT (e.g. it is used in [2, 5, 14, 23, 26, 38, 45]). Some tracking methods expanded this logic further. Tracking algorithms in [25, 49] performed three-stage association, LG-Track used four-stage association [32], while some other methods used multiple-stage association based on tracklet last update [47], or similarity between detections and tracklets [35].

However, it is demonstrated in [42] that two-stage association can introduce identity switches. The same applies to any multiple-stage association strategy which does not consider all tracklet-detection pairs in the same stage.

2.3. BoostTrack

BoostTrack is a TBD system built upon SORT [4], and it tackles three problems in MOT: finding a simple and better similarity measure to improve association performance, the selection of true positive detections, and using a simple one-stage association to avoid identity switches to which all multiple-stage association techniques are prone to [42]. To improve the association performance, BoostTrack uses three additions to IoU (to "boost" the original IoU similarity) and defines the overall similarity measure between a detected bounding box D_i and a tracklet T_j, $S(D_i, T_j)$, as

[1)] In some implementations, there is a separate threshold for tracklet creation τ_{init}, $\tau_{init} > \tau$, and only the detections with confidence scores greater than τ_{init} can be used for tracklet initialization.

$$S(D_i, T_j) = \text{IoU}(D_i, T_j) + \lambda_{IoU} \cdot c_{i,j} \cdot \text{IoU}(D_i, T_j) + \lambda_{MhD} \cdot S^{MhD}(D_i, T_j) + \lambda_{shape} \cdot S^{shape}(D_i, T_j), \quad (3)$$

where $c_{i,j}$, S^{MhD} and S^{shape} represent detection-tracklet confidence of detection D_i and tracklet T_j, Mahalanobis distance similarity and shape similarity, respectively.

To select more true positive detections for the first (and only) association stage, BoostTrack uses two detection confidence boosting techniques. Before discarding low-confidence detections, it increases the confidence score of bounding boxes which *should* be true positives. One technique relies on Mahalanobis distance to find outliers. Assumption is that the detections which do not correspond to any of the currently tracked objects are not false positives, but rather detections corresponding to previously undetected objects. The other technique, detecting the likely objects, uses IoU between a detection and all the tracklets. If low-confidence detection has a high IoU with some tracklet, the detected bounding box is considered to correspond to that tracklet, and its detection confidence score should be increased. Equation 4 shows the expression for increasing detection confidence score c_{d_i} for some detected bounding box D_i.

$$\hat{c}_{d_i} = \max\Big(c_{d_i}, \beta_c \cdot \max_j(\text{IoU}(D_i, T_j))\Big). \quad (4)$$

Hyperparameters β_c and τ implicitly define the IoU threshold required for a low-confidence detection D_i to surpass τ. Values used in [43] correspond to 0.923 and 0.8 for datasets MOT17 [33] and MOT20 [9], respectively.

The authors also proposed BoostTrack+ method which uses appearance similarity (between embedding vectors) in addition to similarity measures used in 3. BoostTrack+ outperforms BoostTrack at the expanse of longer computation time.

In Table 1, we summarize the methods used as baselines in this paper.

Table 1: Summary of various baseline methods.

Method	Description	Pros	Cons	Improvements over prev. method
SORT	Tracker that uses IoU as similarity measure	One-stage association, real-time performance	Poor selection of true-positive detections, poor tracking performance	/
BoostTrack	Tracker focused on handling unreliable detections and improving association	One-stage association, uses all detections, real-time performance	Lacks strong cues such as visual features embedding, introduces IDs and IDSWs	Improved overall performance
BoostTrack+	Tracker focused on handling unreliable detections and improving association	One-stage association, uses all detections,	Cannot operate in real time in crowded scenes, introduces IDs and IDSWs	Improved tracking performance

2.4. Buffered IoU

To account for irregular motions, Buffered IoU (BIoU) is introduced in [50]. If the predicted state is inaccurate (due to irregular motion), the IoU between the predicted bounding box and the detected bounding box will be low (possibly 0) which makes association difficult or even impossible. The authors proposed to scale (add "buffers" to) the detected and the predicted bounding boxes, i.e. tracklets. Let $o = (x, y, w, h)$ be the original detection (or tracklet), where (x, y) represents the top-left coordinate of the bounding box, and w and h its width and height, respectively. The authors propose to use scaled detection $o_b = (x - bw, y - bh, w + 2bw, h + 2bh)$, where $b \geq 0$ is the scale parameter. More specifically, they performed two-stage association: in the first stage they used small scale parameter b_1, and in the second stage they associated remaining tracklets and detections using larger scale parameter b_2. The logic behind two-stage association is that the unmatched tracklets are more difficult to match and require larger bounding boxes for successful matching.

3. Limitations of confidence boost based on detection of likely objects

One of the contributions of BoostTrack [43] is the DLO confidence boost technique. The effectiveness of this technique is most notable in the case of the MOT20 dataset, where the DLO boost increased MOTA[2)] score by 4.8% (see table 2).

To achieve the best results, a different value of hyperparameter β_c (see equation 4) had to be specified depending on the dataset (0.65 for MOT17 and 0.5 for MOT20), which is one of the limitations of the DLO confidence boost.

The most important issue is the introduction of new IDs, which should not happen if the detections with boosted confidence truly correspond to existing tracklets. Ideally, a DLO confidence boost should produce no new IDs or result in identity switches (IDSWs). This issue is best illustrated by results on the MOT20 validation set which we present in table 2. On the MOT17 validation set (which has only 339 ground-truth IDs, as opposed to the MOT20 validation set which contains 1418 IDs) the DLO boost is less significant.

Table 2: Influence of DLO confidence boost on MOT20 validation set for various baseline methods.

Method	HOTA	MOTA	IDF1	IDSWs	IDs
SORT	56.65	69.91	73.6	1127	1894
SORT + DLO	58.58	73.28	75.09	1259 (+132)	2045 (+151)
BoostTrack - DLO	61.39	77.02	77.23	803	1867
BoostTrack	61.74	77.46	77.45	898 (+95)	2008 (+141)
BoostTrack+ - DLO	62.59	77.16	79.29	730	1852
BoostTrack+	62.58	77.7	78.93	794 (+64)	2019 (+167)

If we take SORT [4] as the most basic and representative baseline, we notice that the increase in main MOT metrics (the left side of table 2), comes at the price of increased IDs (+8%) and IDSWs (+12%).

As the idea behind the DLO confidence boost is to use the detections with similarity (IoU) above a certain threshold, two possible reasons for the method failure and places for improvement are the similarity measure used and the specified threshold.

Several papers have discussed the limitations of IoU used as a similarity measure for association and used various IoU modifications or other motion cues to construct a richer similarity measure (e.g. [6, 26, 34, 35, 43, 48]). If using IoU alone can cause ambiguities and IDSWs in association, it is also unreliable to base the DLO confidence boost solely on IoU. Using a richer similarity measure improves association and enables the discard of falsely associated detection-tracklet pairs. Since DLO confidence boost relies on calculating similarity, a richer similarity measure should also improve the selection of true positive detections.

On the other hand, using a fixed threshold (for a given dataset) for the DLO confidence boost has two drawbacks.

First, it does not take into account the original confidence score. A detection with a higher confidence score is more likely to be a true positive. If a detection has a confidence score slightly below the threshold τ, it should require only a mild conformation in similarity with existing the tracklet to boost its confidence above the τ.

Second, looking from a tracklet perspective, what can be considered a "high" similarity (e.g. IoU) between a detected bounding box?

As in BoostTrack [43], we use the Kalman filter as the tracking module. The quality of Kalman filter predictions reduces when the tracklet does not get matched with a detection, i.e. when we miss the observation and do not execute the Kalman update step. Error covariance prediction in step t, $\hat{P}_t$, is calculated as $\hat{P}_t = F \cdot P_{t-1} F^{\mathsf{T}} + Q$ (we provide the definition of the state x and state transition matrix F in equation 2).

[2)] We discuss the metrics used in subsection 5.1.

The estimate of the variance of $x \in \{u, v, h, r\}$, after not executing Kalman update for n steps (frames), if we omit Q for simplicity, becomes:

$$\hat{\text{var}}(x)_{t+n} = \text{var}(x)_t + n^2 \cdot \text{var}(\dot{x})_t. \tag{5}$$

The variance of the predictions increases quadratically with the increase of n[3)].

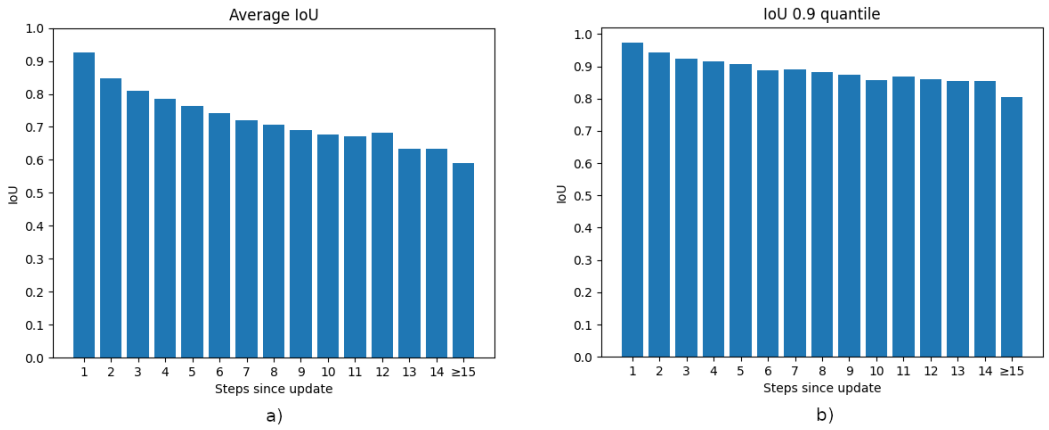

Figure 2: IoU statistics calculated on MOT17 and MOT20 validation sets for tracklets with different numbers of steps since the last update. a) Average IoU b) 0.9 quantile of IoU values.

We empirically verified that the IoU value between a detection and a tracklet decreases as the number of steps since the last successful match (last position update) of the given tracklet increases. The left side of the figure 2 shows the average IoU value (calculated on MOT17 [33] and MOT20 [9] validation sets[4)]) depending on the number of frames since the last update, while the right side of the same figure shows values of 0.9 quantiles. The 0.9 quantile gives us information about what can be considered "high" IoU depending on the number of steps since the last update of the corresponding tracklet. As the number of steps since the last update increases, we can observe a decrease in average IoU, and, more importantly, a decrease in 0.9 quantile.

The DLO confidence boost method should account for this decrease and use different tracklet-specific thresholds based on the number of frames since the last successful match of a given tracklet.

The data from table 2 and the discussion presented indicate that a more sophisticated DLO confidence boost is needed, which we attempt to offer in the following section.

4. Proposed methods

4.1. Soft BIoU

BIoU, introduced in [50], enables to match detection-tracklet pairs which are (due to the inaccurate prediction) too separated and have low or 0 IoU, by expanding bounding boxes. As such, BIoU could also be used for DLO confidence boost. However, BIoU is designed to be used in a two-stage association and cannot be used directly. In [50], the tracklets are, through two-stage association, split into two groups and the corresponding bounding boxes are scaled using hyperparameters b_1 and b_2 depending on the group.

A trivial way of using BIoU in one-stage association setup would be to use single scale value b. However, not only that "easy" and "difficult" tracklets require different b values, but not all "easy" tracklets are equally easy to match, and not all "difficult" tracklets are equally difficult.

[3)] Equality 5 follows directly from substituting F and x from equation 2 and applying prediction for n steps.
[4)] For more information on datasets and implementation details, see subsections 5.1 and 5.2.

Every predicted bounding box should be enlarged to the extent which is proportional to the uncertainty in the quality of the prediction. In [43], tracklet confidence is defined and used as a measure of reliability of the predicted tracklet state. We propose to use tracklet confidence to calculate tracklet specific scale without the need for a two-stage association. However, if we use different scales for different tracklets and comparing with detections in one stage, we cannot simply scale the tracklets and detections by the same parameter b and calculate IoU between all pairs, as done in [50]. Every detection-tracklet pair (D_i, T_j) should use a specific scale based on tracklet confidence, c_{t_j}, value.

Let $\mathbf{o} \rightarrow s$ be detection $\mathbf{o}$ scaled by s, defined (as in [50]) as

$$\mathbf{o} \rightarrow s = (x - sw, y - sh, w + 2sw, h + 2sh), \tag{6}$$

where (x, y), w, and h represent top-left corner of the bounding box, its width and height, respectively. We define soft BioU (SBioU) between D_i and T_j as

$$\text{SBIoU}(D_i, T_j) = \text{IoU}\Big(D_i \rightarrow \frac{1 - c_{t_j}}{4}, T_j \rightarrow \frac{1 - c_{t_j}}{2}\Big), \tag{7}$$

for $i \in \{1, 2, \ldots, n\}$, $j \in \{1, 2, \ldots, m\}$. As the tracklet confidence decreases, the scale increases. To calculate the scaling parameter, we divide $1 - c_{t_j}$ by 2 to match the range of scale values from [50] (for $c_{t_j} = 0.0$ it increases scale to $b_2 = 0.5$). When scaling the detection box, we use a smaller scale since we are more uncertain about the tracklet state compared to the detected bounding box position. Note that SBIoU reduces to IoU when tracklet confidence is equal to 1.

4.2. Using improved similarity measure to find likely objects

DLO confidence boost relies on IoU similarity. We propose to replace IoU in equation 4 with a more sophisticated similarity measure S, which gives:

$$\hat{c}_{d_i} = \max\Big(c_{d_i}, \beta_c \cdot \max_j(S(D_i, T_j))\Big). \tag{8}$$

Trivially, we could set S = SBIoU. However, we propose to use a richer similarity measure, of which soft BIoU is only a part.

In [43], a combination of IoU, shape and Mahalanobis distance similarity was used to construct a "boosted" similarity measure which improved association performance. If such a similarity measure helps to better distinguish objects, it should also improve the performance of detecting the likely objects.

We define similarity between a detected bounding box D_i and a tracklet T_j, $S(D_i, T_j)$, as the average of used similarity measures:

$$S(D_i, T_j) = \Big(S_1(D_i, T_j) + S_2(D_i, T_j) + \cdots S_p(D_i, T_j)\Big)/p. \tag{9}$$

Intuitively, using an average of multiple similarity measures for the DLO confidence boost means that all similarity measures need to "agree" to increase the confidence score of a given detection. We use average for simplicity and to give all summands equal weight (note that in equation 3 weights, i.e. lambdas, do not have to be equal). In our implementation, for the DLO confidence boost we use:

$$S(D_i, T_j) = \Big(\text{SBIoU}(D_i, T_j) + S^{MhD}(D_i, T_j) + S^{shape}(D_i, T_j)\Big)/3. \tag{10}$$

Note that detection-tracklet confidence scores $c_{i,j}$ used to calculate S^{shape} decrease when detection confidence is low which decreases the S^{shape} and makes it less reliable for the purposes of this work [43]. To resolve this issue, we set $c_{i,j} = 1$ when using shape similarity for the detection confidence boost.

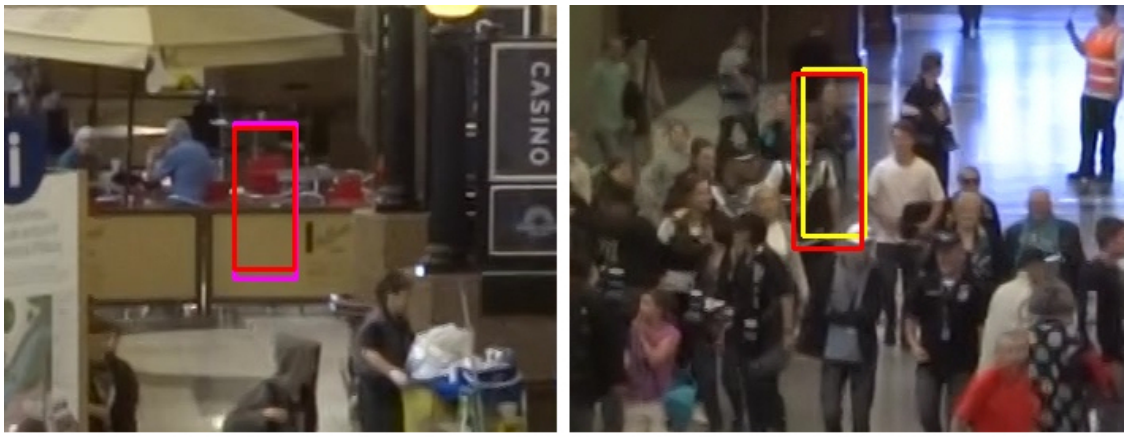

Figure 3: a) Low confidence false positive detection (in red) with high IoU between the tracklet (in purple). Mahalanobis distance similarity between the detection and the tracklet is low, and this detection will not be used. b) Low confidence true positive detection (in red) with relatively low IoU between the corresponding tracklet (in yellow). However, the average similarity measure when using SBIoU, S^{shape} and S^{MhD} is high enough.

In Figure 3 we show two examples from the MOT20 dataset of low-confidence detections.

On the left side of the figure, we show false positive detection corresponding to a ghost track. The IoU between the detection and the tracklet is high enough (IoU=0.82). However, the ghost tracks are static and the corresponding estimated covariance values are low which leads to low Mahalanobis distance similarity values even for a small mismatch ($S^{MhD} = 0.03$ in the example). This results in low average similarity and discarding the false positive detection.

On the right side of the figure, we show a true positive detection D_i ($c_{d_i} = 0.16$) with relatively low IoU between the corresponding tracklet T_j (IoU$(D_i, T_j) = 0.74$). However, SBIoU$(D_i, T_j) = 0.77$, $S^{MhD}(D_i, T_j) = 0.96$, $S^{shape}(D_i, T_j) = 0.82$, and the average similarity (from equation 9) is high enough to keep D_i for the association step.

4.3. Soft detection confidence boost

As noted in section 3, DLO confidence boost treats equally detections with very low and relatively high confidence scores. If the IoU between a given tracklet and a low-confidence detection is high enough (e.g. 0.8 for the MOT20 dataset), the detection's confidence will be increased enough to be used for the association. The detection with very low (e.g. 0.05) and a relatively high confidence score (e.g. 0.35) require the same high IoU in order for their confidence scores to be increased to a value greater than the threshold τ. However, the lower confidence detection is more likely to be false-positive and should require higher IoU compared to the bounding box with higher confidence score.

The greater the detection confidence score, the more accurate the corresponding detected bounding box, i.e. the greater IoU between the ground-truth and detected bounding box [20]. Provided that the tracking module gives accurate predictions, there is also a positive correlation between the detection confidence score and the IoU between the tracklet (i.e. predicted bounding box) and the corresponding detected bounding box. IoU between the tracklet and the detection can thus be a measure or an indicator of detection confidence. If the IoU between the detected bounding box D_i, $D_i \in \{D_1, D_2, \ldots, D_n\}$, and some tracklet is high, the detection confidence score c_{d_i} should also be high. Following the described logic, for every detected bounding box D_i, we perform soft detection confidence boost to obtain a new detection confidence score $\hat{c}_{d_i}$ as:

$$\hat{c}_{d_i} = \max\Big(c_{d_i}, \alpha \cdot c_{d_i} + (1 - \alpha) \cdot \Big(\max_j(S(D_i, T_j)\Big)^q\Big), \tag{11}$$

where $\alpha \in [0, 1]$ and $q \geq 1$ are hyperparameters and S is any similarity measure. We raise S to the power q to have better control over the entire process (all hyperparameter values are discussed in subsection 5.2). Equation 11 combines original confidence scores with the similarity and solves the problem of equal treatment of low and relatively high confidence score detections.

Following the discussion in subsection 4.2, S can be defined as in equation 9.

Figure 4 shows a frame from MOT17 (sequence 11) and a detected bounding box (in red) with confidence score 0.56 which is slightly below the standard threshold 0.6 used for MOT17. The IoU between the detection and the corresponding tracklet is relatively low (0.82, which is low compared to the threshold value 0.923 used for MOT17 in [43]). However, since the original confidence score is only slightly below the threshold, the imperfect IoU of 0.82 is enough to boost the detection confidence above the threshold.

Figure 4: A detected bounding box with confidence (in red) with a confidence score slightly below a threshold, and the corresponding tracklet (in green). The IoU between the detection and the tracklet is relatively low, but enough for the soft detection confidence boost to increase the confidence score above the threshold.

4.4. Varying similarity threshold

All the similarity measures discussed previously depend upon tracking performance, i.e. the quality of predictions. As noted in section 3, the quality of predictions reduces with the increase of the number of steps since the last tracklet update. Reduced quality of predictions leads to lower IoU value when objects get matched (see figure 2) and indicates that a single threshold value for DLO confidence boost cannot provide the best results.

To address the described issue, we propose to use different threshold values when searching for the likely objects. The threshold should be higher for the IoU (or any other similarity measure) between a detection and a recently updated tracklet.

Let last_update(T_j) be the number of steps since the last update of tracklet T_j, $j \in \{1, 2, \ldots, m\}$. We decrease the threshold β_j corresponding to a tracklet T_j linearly from the starting value β_{high} to the final value β_{low}, and define our varying threshold β_j as:

$$\beta_j = \max(\beta_{low}, \beta_{high} - \gamma \cdot (\text{last\_update}(T_j) - 1)). \tag{12}$$

Using the varying similarity threshold, the boosted confidence of the detected bounding box D_i, $i \in \{1, 2, \ldots n\}$, can be obtained as:

$$\hat{c}_{d_i} = \begin{cases} \max(c_{d_i}, \tau), & \text{if } S(D_i, T_j) \geq \beta_j \text{ for any } j \in \{1, 2, \ldots, m\}, \\ c_{d_i}, & \text{otherwise,} \end{cases} \tag{13}$$

where S is any similarity measure, and τ (as noted earlier) threshold for discarding low-confidence detections.

An example of low-confidence detected bounding box D_i with high enough IoU between some tracklet T_j is shown in Figure 5.

Figure 5: A detected bounding box D_i (in red) with an original confidence score of 0.35. A predicted bounding box (in blue) after not matching the tracklet T_j for 27 frames. For the given tracklet, threhold $\beta_j = 0.8$. Since $\text{IoU}(D_i, T_j) = 0.83 \geq \beta_j$, the c_{d_i} is increased, and D_i will not be discarded.

4.5. Combining the proposed methods

All proposed additions are mutually independent and can be used separately or jointly. Algorithm 4.1 shows the combined usage of all proposed elements and can be used as a replacement for the DLO confidence boost used in BoostTrack [43]. Parameters *useS*, *useSB* and *useVT* control whether we should use improved similarity measure, soft detection confidence boost and varying threshold, respectively.

5. Experiments and results

5.1. Datasets and metrics

Datasets. We perform experiments and evaluation of our method on standard MOT benchmark datasets: MOT17 [33] and MOT20 [9]. MOT17 contains pedestrian videos filmed with static and moving camera. The training set consists of 7 sequences and 5316 frames in total (frame rate ranges from 14 to 30 depending on the video), while the test set contains a continuation of the same sequences and has 5919 frames in total.

MOT20 consists of 8 sequences of crowded scenes with changing lighting conditions filmed at 25 FPS. The training set contains 4 sequences (8931 frames) and the test set consists of remaining sequences (4479 frames).

As in previous works (e.g. [2, 30, 56]), we use a custom detector instead of the provided dataset detections and perform experiments under private detection protocol. We use the second half of each training sequence as a validation set.

Metrics. We use standard metrics to assess the performance of our method. Namely, we use:

- Multi-Object Tracking Accuracy (MOTA) metric [3], which penalizes false positive and false negative detections and is primarily used to evaluate detection performance.
- IDF1 [39], which is primarily used as a measure of association performance.
- Higher Order Tracking Accuracy (HOTA) [29], which combines localization accuracy, association and tracking performance and tends to assess the entire MOT performance.

In addition to the mentioned metrics, in the ablation study, we also monitor the number of IDs used and the number of identity switches (IDSW). We monitor IDs because detection confidence boosting can result in an increased number of IDs. IDWS is important because new detections should not cause additional IDSWs.

Algorithm 4.1 Improved detection confidence boost

procedure IDCBoost($D, T, useS, useSB, useVT$) ▷ $D = \{D_1, D_2, \dots D_n\}$, $T = \{T_1, T_2 \dots, T_m\}$
  if useS **then**
    S := compute_similarity(D, T) ▷ Using equation 10.
  else
    S := IoU(D, T)
  end if
  if not useSB and not useVT **then**
    for each i **do**
      $\hat{c}_{d_i} := \max\Big(c_{d_i}, \beta_c \cdot \max_j(S(D_i, T_j))\Big)$. ▷ Applying equation 8
    end for
  else
    if use SB **then**
      for each i **do**
        $c_{d_i} := \max\Big(c_{d_i}, \alpha \cdot c_{d_i} + (1-\alpha) \cdot \Big(\max_j(S(D_i, T_j))\Big)^q\Big)$ ▷ Applying equation 11
      end for
    end if
    if use VT **then**
      for each (i, j) **do**
        $beta_j := \text{compute\_threshold}(T_j, \gamma, \beta_{low}, \beta_{high})$ ▷ Using equation 12.
        if $S(D_i, T_j) \geq beta_j$ **then**
          $c_{d_i} := \max(c_{d_i}, \tau)$
          break
        end if
      end for
    end if
  end if
  return $c_{d_1}, c_{d_2}, \dots, c_{d_n}$ ▷ Outputs boosted confidence scores $\hat{c}_{d_1}, \hat{c}_{d_2}, \dots, \hat{c}_{d_n}$.
end procedure

5.2. Implementation details

MOT and BoostTrack specific settings. We extend the work done in [43] and use the same additional components and settings. We provide a brief overview of the most important components used[5)]. Namely, we use YOLOX-X [17] as the detector with weights from [56]; we apply Enhanced correlation coefficient maximization from [13] for camera motion compensation (and use the implementation from [11]); we use FastReID [18] for computing visual embedding (used for calculating visual appearance similarity); for postprocessing of the results, we use gradient boosting interpolation (GBI) from [53]. We calculate shape mismatch (used in S^{shape}) between a tracklet T_j and a detection D_i, $ds_{i,j}$ as:

$$ds_{i,j} = \frac{|D_i^w - T_j^w| + |D_i^h - T_j^h|}{\max(D_i^w, T_j^w)}. \tag{14}$$

BoostTrack++ specific settings. We run a grid search to find optimal parameters q and α (used for soft detection confidence boost introduced in subsection 4.3). Specifically, we tested settings $(q, \alpha) \in \{1, 1.25, 1.5, 1.75, 2\} \times \{0, 0.05, 0.1, \dots, 0.95, 1\}$ on MOT17 validation set (we used S = IoU for simplicity) and choose $q = 1.5, \alpha = 0.65$ as the best trade-off between different metrics and different baseline settings (e.g. whether we use camera motion compensation, postprocessing, appearance similarity).

[5)] We instruct readers to [43] for more information on minor implementation details.

Based on empirical results displayed in figure 2, as our varying similarity threshold setting we used $\beta_{high} = 0.95$, which we reduce to $\beta_{low} = 0.8$ over 20 frames ($\gamma = (\beta_{high} - \beta_{low})/20 = 0.0075$).

Software. We build our code on top of code from [43], which uses codes from [4, 11, 30, 53, 56].

Hardware. All experiments are performed on the desktop with 13th Gen Intel(R) Core(TM) i9-13900K CPU and NVIDIA GeForce RTX 3080 GPU.

We use TrackEval [28] to evaluate the results on validation sets. The results on the test sets are evaluated on the official MOT Challenge server [1].

5.3. Ablation study

The goal of the proposed method is to replace or improve the performance of the DLO confidence boost technique used in [43]. Ideally, our method should not increase the number of used IDs compared to any baseline which does not apply DLO (because it should only boost the confidence score of the detections which correspond to existing objects). Furthermore, a successful DLO confidence boost should improve the overall tracking performance: not only the MOTA score which penalizes usage of false positive detections, but the values of HOTA, IDF1 and IDWS metrics should also improve because the quality of selected bounding boxes is crucial for the successful matching.

5.3.1. Influence of different similarity measures.

Since the DLO confidence boost resulted in substantial performance improvement on the MOT20, but also in an increased number of IDs and IDWSs, we study the effect of various similarity measures from equation 10 on MOT20 validation set. We performed experiments on three baselines: SORT+DLO, BoostTrack and BoostTrack+. Table 3 shows the results[6] of using SORT+DLO and BoostTrack+ as baselines (as the most basic and the more advanced methods, respectively). We display the results of using the BoostTrack baseline in Appendix B.

The first two rows of table 3 are the baselines. The first row shows results of the baseline method without DLO confidence boost, while the second uses a simple IoU based DLO confidence boost which we try to improve (see equation 4). A successful DLO method should keep IDs and IDSW values as close to the values from the first row, and at the same time improve the HOTA, MOTA and IDF1 values from the second row.

Table 3: The effect of various similarity measures used for DLO confidence boost on the MOT20 validation set (best in bold).

Setting				SORT baseline					BoostTrack+ baseline				
DLO	SBIoU	S^{MhD}	S^{shape}	HOTA	MOTA	IDF1	IDSW	IDs	HOTA	MOTA	IDF1	IDSW	IDs
✗	✗	✗	✗	56.65	69.91	73.6	1127	1894	62.59	77.16	79.29	730	1852
✓	✗	✗	✗	**58.58**	73.28	75.09	1259	2045	62.58	77.7	78.93	794	2019
✓	✗	✗	✓	57.55	**73.4**	72.59	2008	2729	61.99	75.63	77.51	1081	2736
✓	✗	✓	✗	57.76	71.3	74.87	1188	1932	62.79	77.71	79.37	739	1917
✓	✗	✓	✓	57.73	71.33	74.81	1186	**1926**	62.8	77.75	79.35	**723**	**1914**
✓	✓	✗	✗	58.52	73.19	74.94	1228	1981	62.53	**77.84**	78.77	780	1964
✓	✓	✗	✓	58.5	73.22	74.97	1273	2040	62.63	77.7	78.82	782	2038
✓	✓	✓	✗	57.87	71.3	75.08	**1158**	1934	**62.84**	77.78	**79.49**	736	1928
✓	✓	✓	✓	57.95	71.38	**75.2**	1162	1929	62.82	77.8	79.42	729	1918

As results from the table show, using an average of SBIoU, S^{shape} and S^{MhD} offers the best trade-off between the various metrics. It substantially reduces IDs and IDSW values compared to the DLO baseline, while increasing HOTA, MOTA and IDF1 values (the only exception is MOTA score when using the SORT baseline).

[6] We consider the best IDs and IDWS values that are the closest to non-DLO baseline.

5.3.2. Influence of other components

We study the influence of proposed components on BoostTrack+ baseline on MOT17 and MOT20 validation sets. In Appendix C we show the results of the ablation study where we used SORT [4] and BoostTrack as baselines.

Table 4 shows the influence of adding each of the proposed components to the BoostTrack+ baseline: namely, new similarity measure for the confidence boost (S column), soft (detection confidence) boost (SB column), and using varying threshold (VT column) for the DLO confidence boost. Note, when we use SB or VT, and not S, the similarity used in SB and VT is IoU. For more details, see algorithm 4.1. Again, in the first row, we show results of the baseline method without any DLO confidence boost, while the second row shows results of the original DLO confidence boost from BoostTrack.

Table 4: Ablation study on the MOT17 and MOT20 validation sets for different additional components (best in bold). Components are added to the BoostTrack+ baseline.

Setting				MOT17					MOT20				
DLO	S	SB	VT	HOTA	MOTA	IDF1	IDSW	IDs	HOTA	MOTA	IDF1	IDSW	IDs
✗	✗	✗	✗	72.17	81.02	84.86	79	388	62.59	77.16	79.29	730	1852
✓	✗	✗	✗	**72.41**	80.95	85.09	83	402	62.58	77.7	78.93	794	2019
✓	✗	✗	✓	72.13	80.88	84.76	84	**395**	62.8	77.3	79.56	**712**	1852
✓	✗	✓	✗	72.05	80.67	84.91	**80**	402	62.86	77.84	79.23	812	2043
✓	✗	✓	✓	72.02	80.58	85.08	81	412	62.82	77.86	79.16	813	2042
✓	✓	✗	✗	71.88	80.96	84.82	89	398	62.82	77.8	79.42	729	1918
✓	✓	✗	✓	72.17	81.15	84.75	83	398	62.75	77.35	79.44	722	**1848**
✓	✓	✓	✗	72.24	81.23	85.44	81	396	**63.2**	**78.03**	**80.05**	734	1912
✓	✓	✓	✓	72.22	**81.33**	**85.45**	84	396	63.16	78.02	79.97	732	1906

The results from table 4 show that using SB and VT with IoU as a similarity measure (without the proposed average similarity) increases the number of IDs even more than baseline DLO. However, combined with S, we get the best of both worlds: we get a substantial improvement in tracking performance, outperforming the original DLO confidence boost, while at the same time only slightly increasing the number of IDs and IDSWs: +5 IDSWs (6.3%), +8 IDs (2%) on the MOT17 and +2 IDSWs (0.3%), +54 IDs (2.9%) on the MOT20.

5.4. Comparison with other methods

To evaluate our method on MOT17 [33] and MOT20 [9] test sets (under private detection protocol), we used all proposed additions combined (S+SB+VT setting). The results on the test sets are provided in the table 5. We mark offline methods with '*'[7)].

Our BoostTrack++ method achieves improvement compared to the BoostTrack+ on the MOT17 test set: +0.2 HOTA, +0.1 MOTA, +0.4 IDF1, -24 IDSWs. On more challenging MOT20, BoostTrack++ shows slightly better improvement compared to the BoostTrack+: +0.2 HOTA, +0.5 MOTA, +0.5 IDF1, -65 IDSWs. Improvement in MOTA indicates the increase in the use of true positive detections, which improves HOTA and IDF1 scores.

Compared to the online trackers, our method ranks first in HOTA score (66.6) on the MOT17 test set, while on the MOT20, our method ranks first in HOTA (66.4) and IDF1 (82.0) among both online and offline methods.

[7)]Note that our method, as most "online" methods (e.g. [2, 20, 30, 42, 56]), applies postprocessing to the results (which are obtained online), and could be best described as "semi-online". However, the official MOT Challenge [1] distinguishes only offline and online methods, and methods like ours are considered online.

Table 5: Comparison with other MOT methods on the MOT17 test set (best in bold). Offline methods are marked with '*'.

Method	MOT17				MOT20			
	HOTA	MOTA	IDF1	IDSW	HOTA	MOTA	IDF1	IDSW
FairMOT [57]	59.3	73.7	72.3	3303	54.6	61.8	67.3	5243
MOTR [52]	62.0	78.6	75.0	2619	/	/	/	/
ByteTrack [56]	63.1	80.3	77.3	2196	61.3	77.8	75.2	1223
QuoVadis [10]	63.1	80.3	77.7	2103	61.5	77.8	75.7	1187
BPMTrack [16]	63.6	81.3	78.1	2010	62.3	78.3	76.7	1314
SuppTrack* [55]	/	/	/	/	61.9	78.2	75.5	1325
UTM [51]	64.0	81.8	78.7	1431	62.5	78.2	76.9	1228
FineTrack [37]	64.3	80.0	79.5	1272	63.6	77.9	79.0	980
StrongSORT++ [11]	64.4	79.6	79.5	1194	62.6	73.8	77.0	770
BASE* [22]	64.5	81.9	78.6	1281	63.5	78.2	77.6	984
Deep OC-SORT [30]	64.9	79.4	80.6	1023	63.9	75.6	79.2	779
BoT_SORT [2]	65.0	80.5	80.2	1212	63.3	77.8	77.5	1313
SparseTrack [27]	65.1	81.0	80.1	1170	63.5	78.1	77.6	1120
MotionTrack [36]	65.1	81.1	80.1	1140	62.8	78.0	76.5	1165
LG-Track [32]	65.4	81.4	80.4	1125	63.4	77.8	77.4	1161
StrongTBD [40]	65.6	81.6	80.8	954	63.6	78.0	77.0	1101
C-BIoU [50]	66.0	**82.8**	82.5	1194	/	/	/	/
PIA2 [41]	66.0	82.2	81.1	1026	64.7	78.5	79.0	1023
ImprAsso [42]	66.4	82.2	82.1	**924**	64.6	**78.6**	78.8	992
SUSHI* [7]	66.5	81.1	83.1	1149	64.3	74.3	79.8	706
ConfTrack [20]	65.4	80.0	81.2	1155	64.8	77.2	80.2	**702**
BoostTrack+ [43]	66.4	80.6	81.8	1086	66.2	77.2	81.5	827
CoNo-Link* [15]	**67.1**	82.7	**83.7**	1092	65.9	77.5	81.8	956
BoostTrack++ (**ours**)	66.6	80.7	82.2	1062	**66.4**	77.7	**82.0**	762

6. Conclusions

In this paper, we identified the drawbacks of the DLO confidence boost introduced in BoostTrack and proposed a method to mitigate the identified issues. Our goal was to utilize the benefits of DLO confidence boost but avoid its drawbacks - namely, causing IDSWs and introducing new IDs. To this end, we proposed a novel soft BIoU similarity measure and three plug-and-play additions, each of which attempts to provide better control and utilize richer tracklet and detection information to improve the selection of true positive detected bounding boxes.

Using our methods with BoostTrack+ baseline, our BoostTrack++ method ranks first in HOTA and IDF1 metrics on the MOT20 dataset and achieves comparable to the state of the art results on the MOT17 dataset.

However, the achieved MOTA score is still relatively low, indicating the need for even better algorithms for selecting true positive detections in a one-stage TBD MOT paradigm.

Appendix A. List of abbreviations

In table A.6, we show the list of all abbreviations used in the paper.

Table A.6: List of abbreviations

Abbreviation	Definition
TBD MOT	Tracking by detection multiple object tracking
IoU	Intersection over union
BIoU	Buffered IoU from [50]
SBIoU	Soft BIoU introduced in subsection 4.1
DLO	Detection of likely objects used to boost confidence scores in [43]
IDSW	Identity switch
S (ablation study)	Similarity measure from equation 10
SB (ablation study)	Soft (detection confidence) boost from subsection 4.3
VT (ablation study)	Varying similarity threshold from subsection 4.4

Appendix B. Influence of different similarity measures

In table B.7 we show the results of applying various similarity measures instead of IoU for the DLO confidence boost. We used BoostTrack as the baseline method. As with SORT or BoostTrack+ baselines (displayed in table 3), using the average of all three similarity measures provides the best trade-off between introducing additional identities and overall tracking performance.

Table B.7: The effect of various similarity measures used for the DLO confidence boost on the MOT20 validation set (best in bold).

Setting				BoostTrack baseline				
DLO	SBIoU	S^{MhD}	S^{shape}	HOTA	MOTA	IDF1	IDSW	IDs
✗	✗	✗	✗	61.39	77.02	77.23	803	1867
✓	✗	✗	✗	61.74	77.46	77.45	898	2008
✓	✗	✗	✓	59.64	75.34	73.85	1252	2637
✓	✗	✓	✗	61.7	77.38	77.52	856	1911
✓	✗	✓	✓	61.69	77.52	77.57	862	1911
✓	✓	✗	✗	61.74	**77.65**	77.58	865	1949
✓	✓	✗	✓	61.17	77.5	76.63	915	2026
✓	✓	✓	✗	61.8	77.6	77.7	834	1917
✓	✓	✓	✓	**61.88**	77.57	**77.89**	**825**	**1902**

Appendix C. Ablation study on SORT and BoostTrack

To show the robustness of the proposed components, we performed additional ablation experiments using SORT and BoostTrack as baseline methods. Table C.8 shows the results of various settings on MOT20 validation set. Note that, ideally, our method should not introduce new IDs and IDSWs, and it should at the same time improve overall tracking performance. Using only VT with BoostTrack as the baseline provides a good example of trade-offs required. Adding VT even reduced IDs and IDWSs by 2. However, the overall tracking performance is only slightly improved by adding VT only, while S or S+SB+VT settings provide a better trade-off between various metrics and the number of additional IDs and IDSWs.

Table C.8: Ablation study on the MOT20 validation sets for different additional components (best in bold).

Setting				SORT baseline					BoostTrack baseline				
DLO	S	SB	VT	HOTA	MOTA	IDF1	IDSW	IDs	HOTA	MOTA	IDF1	IDSW	IDs
✗	✗	✗	✗	56.65	69.91	73.6	1127	1894	61.39	77.02	77.23	803	1867
✓	✗	✗	✗	58.58	**73.28**	75.09	1259	2045	61.74	77.46	77.45	898	2008
✓	✗	✗	✓	56.62	70.13	73.6	**1130**	1901	61.53	77.2	77.46	**801**	**1865**
✓	✗	✓	✗	**58.07**	73.06	74.55	1315	2059	61.29	77.61	76.88	902	2052
✓	✗	✓	✓	58.03	73.06	75.0	1325	2060	61.25	77.59	76.84	904	2048
✓	✓	✗	✗	57.95	71.38	**75.2**	1162	1929	**61.88**	77.57	**77.89**	825	1902
✓	✓	✗	✓	56.9	70.41	73.97	1148	**1891**	61.54	77.23	77.48	819	1862
✓	✓	✓	✗	57.61	71.65	74.53	1192	1933	61.57	77.67	77.43	866	1901
✓	✓	✓	✓	57.65	71.64	74.6	1200	1935	61.56	**77.71**	77.38	864	1895

References

[1] *MOT Challenge*, https://motchallenge.net/, Accessed: 2024-06-04.
[2] N. Aharon, R. Orfaig, B. Bobrovsky, *BoT-SORT: Robust Associations Multi-Pedestrian Tracking*, ArXiv preprint abs/2206.14651, (2022).
[3] K. Bernardin, R. Stiefelhagen, *Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics* J. Image. Video. Proc. (2008), 1–10.
[4] A. Bewley, Z. Ge, et al. *Simple online and realtime tracking*, ICIP, (2016), 3464–3468.
[5] D.C. Bui, N.L. Hoang *CAMTrack: a combined appearance-motion method for multiple-object tracking* Mach. Vis. Appl. **35** (2024).
[6] J. Cao, J. Pang, et al, *Observation-Centric SORT: Rethinking SORT for Robust Multi-Object Tracking*, CVPR, (2023), 9686–9696.
[7] O. Cetintas, G. Braso, L. Leal-Taixe, *Unifying Short and Long-Term Tracking with Graph Hierarchies*, CVPR, (2023), 22877–22887.
[8] Y. Cui, C. Zeng, et al *Sportsmot: A large multi-object tracking dataset in multiple sports scenes*, CVPR, (2023), 9921–9931.
[9] P. Dendorfer, A. Ošep, et al. *MOTChallenge: A Benchmark for Single-Camera Multiple Target Tracking*, Int. J. Comput. Vis. **129** (2021), 845–881.
[10] P. Dendorfer, V. Yugay, et al. *Quo Vadis: Is Trajectory Forecasting the Key Towards Long-Term Multi-Object Tracking?*, Adv. Neural Inf. Process. Syst. **35** (2022), 15657–15671.
[11] Y. Du, Z. Zhao, et al. *Strongsort: Make deepsort great again*, IEEE Trans. Multimedia **25** (2023), 8725–8737.
[12] R. E. Kalman, *A New Approach to Linear Filtering and Prediction Problems*, J. Basic Eng. **82** (1960), 35–45.
[13] G.D. Evangelidis, E. Z. Psarakis, *Parametric Image Alignment Using Enhanced Correlation Coefficient Maximization*, IEEE Trans. Patt. Anal. Mach. Intel. **30** (2008), 1858–1865.
[14] W. Feng, L. Bai, et al. *Towards Frame Rate Agnostic Multi-Object Tracking*, Int. J. Comput. Vis. **132** (2022), 1443–1462.
[15] Y. Gao, H. Xu, et al. *Multi-Scene Generalized Trajectory Global Graph Solver With Composite Nodes for Multiple Object Tracking*, AAAI **38** (2024), 1842–1850.
[16] Y. Gao, H. Xu, et al. *BPMTrack: Multi-Object Tracking With Detection Box Application Pattern Mining*, IEEE Trans. Image Process **33**, (2024), 1508–1521.
[17] Z. Ge, S. Liu, et al. *YOLOX: Exceeding YOLO Series in 2021*, ArXiv preprint abs/2107.08430, (2021).
[18] L. He, X. Liao, et al. *Fastreid: A pytorch toolbox for general instance re-identification*, ACM-MM, (2023), 9664–9667.
[19] S. Jha, C. Seo, et al. *Real time object detection and trackingsystem for video surveillance system*, Multimed. Tools. Appl. **80** (2021), 3981–3996.
[20] H. Jung, S. Kang, et al. *ConfTrack: Kalman Filter-Based Multi-Person Tracking by Utilizing Confidence Score of Detection Box*, WACV, (2024), 6583–6592.
[21] H. Kuhn, *The Hungarian method for the assignment problem*, Nav. Res. Logistics Quart. **2** (1955), 83–97.
[22] M. Larsen, S. Rolfsjord, et al. *BASE: Probably a Better Approach to Visual Multi-Object Tracking*, VISIGRAPP **4**, (2024) 110–121.
[23] Y. Li, Y. Youyu *A lightweight scheme of deep appearance extraction for robust online multi-object tracking*, Vis. Comput. **40** (2024), 2049–2065.
[24] X. Li, P. Yin, et al. *Research on Multi-Object Tracking Algorithm for Thyroid Nodules Based on ByteTrack*, EEBDA, (2024), 1589–1592.
[25] Y. Li, L. Wu, et al. *Motion estimation and multi-stage association for tracking-by-detection*, Complex Intell. Syst. **10** (2024), 2445–2458.
[26] J. Li, Y. Ding, H. Wei, (2022) *SimpleTrack: Rethinking and Improving the JDE Approach for Multi-Object Tracking*, Sensors, **22**.
[27] Z. Liu, X. Wang, et al. *SparseTrack: Multi-Object Tracking by Performing Scene Decomposition based on Pseudo-Depth*, ArXiv preprint abs/2306.05238, (2023).
[28] J. Luiten, A.Hoffhues, *Trackeval*, https://github.com/JonathonLuiten/TrackEval (2020)
[29] J. Luiten, A. Ošep, et al. *HOTA: A Higher Order Metric for Evaluating Multi-object Tracking*, Int. J. Comput. Vis. **129** (2021), 548–578.
[30] G. Maggiolino, A. Ahmad, et al. *Deep OC-Sort: Multi-Pedestrian Tracking by Adaptive Re-Identification*, ICIP, (2023), 3025–3029.
[31] P. Mahalanobis, *On the generalized distance in statistics*, Proceedings Of The National Institute Of Sciences (Calcutta), **2** (1936), 49–55.

[32] T. Meng, C. Fu, *Localization-Guided Track: A Deep Association Multi-Object Tracking Framework Based on Localization Confidence of Detections*, ArXiv preprint abs/2309.09765, (2023).
[33] A. Milan, L. Leal-Taixé, et al. *MOT16: A benchmark for multi-object tracking*, ArXiv preprint abs/1603.00831, (2016).
[34] M. Morsali, Z. Sharifi, et al. *SFSORT: Scene Features-based Simple Online Real-Time Tracker*, ArXiv preprint abs/2404.07553, (2024).
[35] M. Nasseri, M. Babaee, et al. *Online relational tracking with camera motion suppression*, J. Vis. Commun. & Im. Repr. **90** (2023).
[36] Z. Qin, S. Zhou, et al. *MotionTrack: Learning Robust Short-Term and Long-Term Motions for Multi-Object Tracking*, CVPR, (2023), 17939–17948.
[37] H. Ren, S. Han, et al. *Focus On Details: Online Multi-Object Tracking with Diverse Fine-Grained Representation*, CVPR, (2023), 11289–11298.
[38] K. Ren, C. Hu, H. Xi, *Rlm-tracking: online multi-pedestrian tracking supported by relative location mapping*, Int. J. Mach. Learn. & Cyber. **15** (2024), 2881–2897.
[39] E. Ristani, F. Solera, et al. *Performance Measures and a Data Set for Multi-target, Multi-camera Tracking*, ECCVW, (2016), 17–35.
[40] D. Stadler *A Detailed Study of the Association Task in Tracking-by-Detection-based Multi-Person Tracking*, Proceedings of the 2022 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory, **62**, (2023) 59–85.
[41] D. Stadler, J. Beyerer, *Past Information Aggregation for Multi-Person Tracking*, ICIP, (2023), 321–325.
[42] D. Stadler, J. Beyerer, *An Improved Association Pipeline for Multi-Person Tracking*, CVPRW, (2023), 3170–3179.
[43] V. Stanojevic, B. Todorovic, *BoostTrack: boosting the similarity measure and detection confidence for improved multiple object tracking*. Mach. Vis. Appl. **35** (2024).
[44] Y. Wang, Z. Wang, et al. *A Tracking-By-Detection Based 3D Multiple Object Tracking for Autonomous Driving*, ICAUS, (2021), 3414–3423.
[45] Y. Wang, J. Hsieh, et al. *SMILEtrack: SiMIlarity LEarning for Occlusion-Aware Multiple Object Tracking*, AAAI, (2024), 5740–5748.
[46] T. Wengefeld, S. Müller, et al. *A multi modal people tracker for real time human robot interaction*, RO-MAN, (2019), 1–8.
[47] N. Wojke, A. Bewley, D. Paulus, *Simple online and realtime tracking with a deep association metric*, ICIP, (2017), 3645–3649.
[48] M. Yang, G. Han, et al. *Hybrid-sort: Weak cues matter for online multi-object tracking*, AAAI **38** (2024), 6504–6512.
[49] J. Yang, Y. Ban, J. Liu, *Local many-to-many matching via ROI feature decomposition for multi-object tracking*, SIViP (2024), 1–17.
[50] F. Yang, S. Odashima, et al. *Hard to Track Objects with Irregular Motions and Similar Appearances? Make It Easier by Buffering the Matching Space*, WACV, (2023), 4788–4797.
[51] S. You, H. Yao, et al. *UTM: A Unified Multiple Object Tracking Model with Identity-Aware Feature Enhancement*, CVPR. (2023), 21876–21886.
[52] F. Zeng, B. Dong, et al. *MOTR: End-to-End Multiple-Object Tracking with Transformer*, ECCV (2022), 659–675.
[53] K. Zeng, Y. You, et al. *NCT:noise-control multi-object tracking*, Complex Intell. Syst. **9** (2023), 4331—4347.
[54] L. Zhang, J. Gao, et al *Animaltrack: A benchmark for multi-animal tracking in the wild*, Int. J. Comput. Vis. **131** (2023), 496–513.
[55] Y. Zhang, H. Chen, et al. *Handling Heavy Occlusion in Dense Crowd Tracking by Focusing on the Heads*, AI 2023: Advances in Artificial Intelligence **14471** (2023), 79–90.
[56] Y. Zhang, P. Sun, et al. *ByteTrack: Multi-Object Tracking By Associating Every Detection Box*, ECCV, (2022), 1–21.
[57] Y. Zhang, C. Wang, et al. *FairMOT: On the Fairness of Detection and Re-identification in Multiple Object Tracking*, Int. J. Comput. Vis. **129** (2021), 3069–3087.