Tracking without Label: Unsupervised Multiple Object Tracking via Contrastive Similarity Learning

Sha Meng*, Dian Shao*, Jiacheng Guo, Shan Gao†
Northwestern Polytechnical University, Xi'an, China
{mengsha,gjc1}@mail.nwpu.edu.cn, {shaodian,gaoshan}@nwpu.edu.cn

Abstract

Unsupervised learning is a challenging task due to the lack of labels. Multiple Object Tracking (MOT), which inevitably suffers from mutual object interference, occlusion, etc., is even more difficult without label supervision. In this paper, we explore the latent consistency of sample features across video frames and propose an Unsupervised Contrastive Similarity Learning method, named UCSL, including three contrast modules: self-contrast, cross-contrast, and ambiguity contrast. Specifically, i) self-contrast uses intra-frame direct and inter-frame indirect contrast to obtain discriminative representations by maximizing self-similarity. ii) Cross-contrast aligns cross- and continuous-frame matching results, mitigating the persistent negative effect caused by object occlusion. And iii) ambiguity contrast matches ambiguous objects with each other to further increase the certainty of subsequent object association through an implicit manner. On existing benchmarks, our method outperforms the existing unsupervised methods using only limited help from ReID head, and even provides higher accuracy than lots of fully supervised methods.

1. Introduction

As a basic task in computer vision, Multiple Object Tracking (MOT) is widely applied in a variety of fields, including robot navigation, intelligent surveillance, and other aspects [36, 33]. Currently, one of the most popular tracking paradigms is joint detection and re-identification (ReID) embeddings. In the case of supervision, ReID is regarded as a classification task. To keep track of objects, many works [39, 45] utilize appearance features for object association, where the representation ability of the ReID head will directly affect the accuracy of the object association.

However, due to limitations in various conditions such as labeled datasets, to meet the needs of researchers, there has been a growing requirement to annotate tracking datasets, which is costly and time-consuming. Therefore, unsupervised learning of visual representation has attracted great attention in tracking. Some works [35, 30] have demonstrated that training the network in the real direction can also be done even without ground truth. Some works [5, 6] directly use ReID features to cluster objects with high similarity

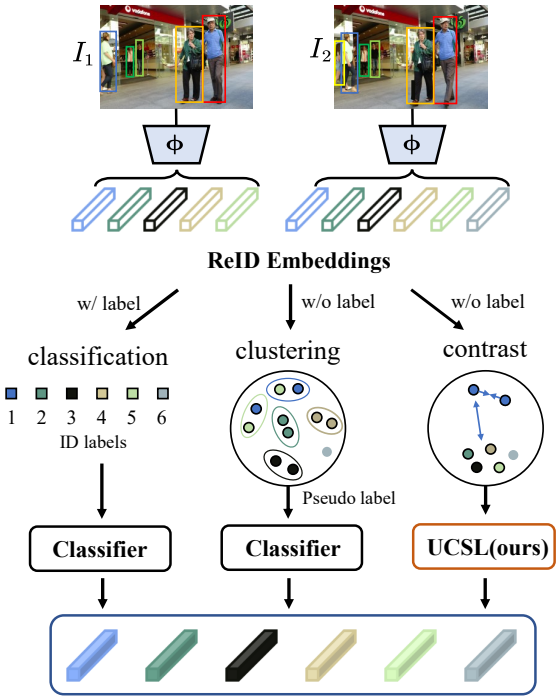

Figure 1. Supervised and Unsupervised MOT. In the joint detection and ReID embeddings framework, to obtain discriminative embeddings for tracking, the left branch is a usual method of supervised MOT training, *i.e.*, given labels, it is trained as an object classification task. The middle branch is a common method of unsupervised training, *i.e.*, it is processed by clustering, and targets with high similarity are regarded as the same class. The right branch is the proposed method with contrast similarity learning to improve the similarity of the same objects without label information.

*Equal Contribution. †Corresponding Author

into the same class, then generate pseudo-labels to train the network, as shown in the middle branch of Figure 1. But these cluster-based methods are easy to accumulate errors during the training process. In contrast, we consider using an Unsupervised Contrastive Similarity Learning (UCSL) to train the ReID branch without generating pseudo-labels, as shown in the right branch of Figure 1.

As a video task, the objects in MOT are always changing over time, which leads to inevitable problems of mutual occlusion between objects and objects, and between objects and non-objects, as well as the disappearance of old objects and the appearance of new ones. Occluded objects are not represented consistently from frame to frame due to additional interference features. Lost and emerging objects theoretically cannot be matched with other objects, and they are almost always negative for the current association stage. Thus, it is difficult for the model to determine whether arbitrary two objects are the same or not. In supervised cases, ID labels can be used to make the training more explicitly directed, while in the unsupervised case, the dual limitation of unlabeling and inherent problems makes unsupervised MOT even more challenging. So we manage to find potential connections between objects to determine if the objects are identical.

In this paper, we propose an Unsupervised Contrastive Similarity Learning (UCSL) method to solve the inherent object association problems of unsupervised MOT. Specifically, UCSL consists of three modules, self-contrast, cross-contrast and ambiguity contrast, designed to address different issues respectively. For the self-contrast, we first match between objects within frames and between objects in adjacent frames. Correspondingly, we get the direct and indirect matching results of the intra-frame objects. Then we maximize the matching probability of self-to-self to maximize the similarity of the same objects. For cross-contrast, considering theoretically the cross-frame matching results should be consistent with the final results of continuous matching, we improve the similarity of the occluded objects by making these two matching results as close as possible. For ambiguity contrast, we match between ambiguous objects, mainly containing occluded, lost, and emerging objects whose final similarity is generally low, to further determine the object identity. Our proposed method is simple but effective, which achieves outstanding performance by utilizing only the ReID embeddings without adding any additional branch such as the occlusion handling or optical-flow based cue to the detection branch.

We implement the method on the basis of FairMOT [45] using the pre-trained model on the COCO dataset [19]. Our experiments on the MOT15 [15], MOT17 [24] and MOT20 [7] datasets are conducted to evaluate the effectiveness of the proposed method. The performance of our unsupervised approach is comparable with, or outperforms, that of some supervised methods using expensive annotations.

Overall, our contributions are summarized as follows:

- We propose a contrastive similarity learning method for unsupervised MOT task, which pursues latent object consistency based only on the sample features in the ReID module given without the ID information.
- We design three useful modules to model associations between objects in different cases. To elaborate, self-contrast module matches intra-frame objects, cross-contrast module associates cross-frame objects, and ambiguity contrast module deals with those hard/corner cases (*e.g.*, occluded objects, lost objects, *etc*.)
- Experiments on MOT15[15], MOT17[24] and MOT20 [7] demonstrate the effectiveness of the proposed UCSL method. As an unsupervised method, UCSL outperforms state-of-the-art unsupervised MOT methods and even achieves similar performance as the fully supervised MOT methods.

2. Related Work

Multi-Object Tracking. Multi-object tracking is a task that localizes objects from consecutive frames and then associates them according to their identity. Thus, for a long time, the most classic tracking paradigm is tracking-by-detection [27], *i.e.*, firstly, an object detector is used to detect objects from every frame, and secondly, a tracker is used to associate these objects across frames. A large number of works [3, 40, 32, 4] in this paradigm have achieved decent performance, but the paradigm relies too much on the performance of detectors. In the past two years, the joint detection and tracking or embedding paradigm has become stronger. Some transformer-based MOT architectures [31, 23, 41] designed two decoders to perform detection and object propagation respectively. JDE [39] and FairMOT [45] directly incorporated the appearance model into a one-stage detector, and then the model can simultaneously output detection results and the corresponding embeddings. These simple but effective frameworks have been what we are looking for, so we take FairMOT [45] as our baseline.

Unsupervised Tracking. For some tasks, existing datasets or other resources cannot meet the needs of researchers. In this condition, unsupervised learning has been a popular solution and its efficiency has been demonstrated in related studies [12, 21, 35, 30]. SimpleReID [12] first used unlabeled videos and the corresponding detection sets, and generated tracking results using SORT [3] to simulate the labels, and trained the ReID network to predict the labels of the given images. It is the first demonstration of the effectiveness of the simple unsupervised ReID network

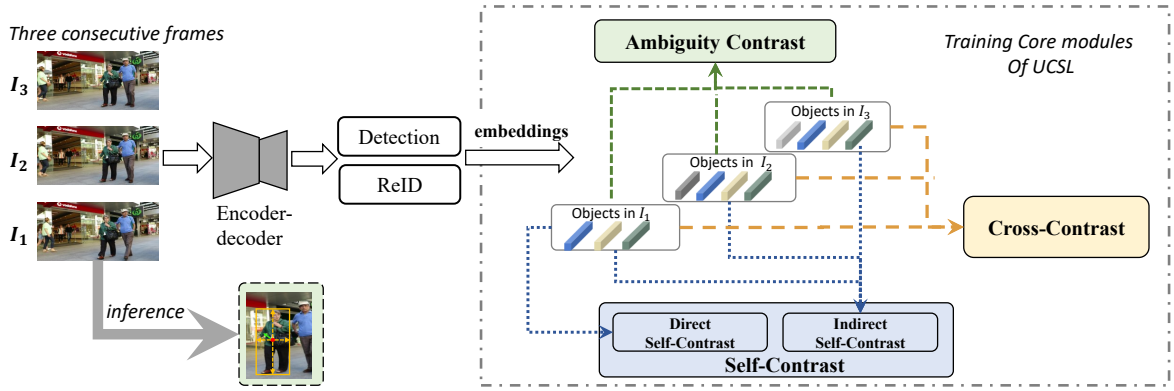

Figure 2. The overall pipeline of our proposed unsupervised contrastive similarity learning model (UCSL), which learns representations with self-contrast, cross-contrast and ambiguity contrast.

for MOT. Liu *et al*. [21] proposed a model, named OUTrack, using an unsupervised ReID learning module and a supervised occlusion estimation module together to improve tracking performance.

Re-Identification. In the field of re-identification, which is more relevant to MOT, unsupervised learning has been widely used through various means including domain adaption, clustering, *etc*. Considering the visual similarity and cycle consistency of labels, MMCL [34] predicted pseudo labels and regarded each person as a class, transforming ReID into a multi-classification problem. Some other works [5, 6, 20] also utilized clustering algorithms to generate pseudo labels and take them as ground truth to train the network. However, error accumulation is easy to occur during the iterative process. Recent methods propose self-supervised learning, Wang *et al*. [38] proposed CycAs inspired by the data association concept in multi-object tracking. By using the self-supervised signal as a constraint on the data, networks gradually strengthen the feature expression ability during the training process.

Cycle Consistency. Cycle consistency is originally proposed in Generative Adversarial Network (GAN), and widely used in segmentation, tracking, *etc*. Jabri *et al*. [10] constructed a space-time graph from the video, and cast correspondence as prediction of links. By cycle consistency, the single path-level constraint implicitly supervised chains of intermediate comparisons. Wang *et al*. [37] used cycle consistency in time as the free supervisory signal for learning visual representations from scratch. Then they used the acquired representation to find nearest neighbors across space and time in a range of visual correspondence tasks.

Contrastive Learning. Contrastive learning has shown great potential in self-supervised learning. Pang *et al*. [26] proposed QDTrack, which densely sampled hundreds of region proposals on a pair of images for contrastive learning. And they directly combined this with existing detection methods. Yu *et al*. [42] proposed multi-view trajectory contrastive learning and designed a trajectory-level contrastive loss to explore the inter-frame information in the whole trajectories. Bastani *et al*. [1] proposed to construct two different inputs for the same video sequence by hiding different information. Then they computed the trajectory of that sequence by applying the RNN model independently on each input, and trained the model using contrastive learning to produce consistent tracks.

3. Method

In this section, we first introduce the overall pipeline, as illustrated in Figure 2, and then describe the corresponding specific concepts in detail in the subsequent parts. Finally, we introduce the whole steps of training and inference.

3.1. Contrast Similarity Learning

Given consecutive three images $\boldsymbol{I}_1, \boldsymbol{I}_2, \boldsymbol{I}_3 \in \mathbb{R}^{H\times W\times 3}$, we first feed them to the backbone, then through detection branches and ReID heads, we could get detection results and ReID feature maps, as shown in Figure 2. Based on the position of the bounding box in the ground truth, the feature embedding corresponding to each object is obtained from the corresponding feature map, which forms embedding matrices $\boldsymbol{X}_1 = \left[\boldsymbol{x}_1^0, \boldsymbol{x}_1^1, \ldots\ldots, \boldsymbol{x}_1^{N-1}\right] \in \mathbb{R}^{D\times N}$, $\boldsymbol{X}_2 = \left[\boldsymbol{x}_2^0, \boldsymbol{x}_2^1, \ldots\ldots, \boldsymbol{x}_2^{M-1}\right] \in \mathbb{R}^{D\times M}$ and $\boldsymbol{X}_3 = \left[\boldsymbol{x}_3^0, \boldsymbol{x}_3^1, \ldots\ldots, \boldsymbol{x}_3^{K-1}\right] \in \mathbb{R}^{D\times K}$, where N, M, and K are the object numbers in $\boldsymbol{I}_1$, $\boldsymbol{I}_2$ and $\boldsymbol{I}_3$, respectively, and D is the embedding dimension.

The ReID branch is connected to three contrast similarity learning branches, in which (1) Self-contrast uses intra-frame direct and inter-frame indirect self-matching to obtain discriminative representations and reduce feature interference from other objects by maximizing self-similarity. (2) Cross-contrast uses cross- and continuous-frame matching, and then adjusts similarity between objects to extract more beneficial features for object association. (3) Ambi-

guity contrast takes into account occluded, lost, and emerging objects simultaneously, and these ambiguous objects are matched with each other again to further increase the certainty of subsequent object association. We will describe the specific operation in Section 3.1.1, 3.1.2 and 3.1.3, respectively.

3.1.1 Self-Contrast Module

According to the latent knowledge that objects from the same frame must belong to different classes, we can determine that the similarity between self-to-self should be large enough. So the proposed self-contrast finally lands on a self-to-self comparison, which is a strong, deterministic self-supervised restriction. This strong restriction allows us to improve the similarity of the same targets and reduce the interference from other objects by direct and indirect self-contrast learning, as shown in the first column of Figure 3.

Direct Self-Contrast. We use current feature matrix $\boldsymbol{X}_1 = \left[\boldsymbol{x}_1^0, \boldsymbol{x}_1^1, ..., \boldsymbol{x}_1^{N-1}\right] \in \mathbb{R}^{D\times N}$ to directly compute the self-similarity matrix $\boldsymbol{S}_{ds} = \boldsymbol{X}_1^T \boldsymbol{X}_1 \in \mathbb{R}^{N\times N}$, where T represents transpose operation. Then we compute the assignment matrix with a softmax operation, as

$$\boldsymbol{S}_{dsc} = \boldsymbol{\psi}_{\text{row}}\left(\boldsymbol{S}_{ds}\right), \tag{1}$$

where $\boldsymbol{\psi}_{\text{row}}$ is row-wise softmax operation.

Indirect Self-Contrast. MOT itself operates on multiple frames, so we further perform our self-contrast similarity learning by indirect self-to-self matching. To measure similarity between objects, we calculate cosine similarity to get a similarity matrix between objects of different frames $\boldsymbol{S}_{is} = \boldsymbol{X}_1^T \boldsymbol{X}_2 \in \mathbb{R}^{N\times M}$. And similar to Eq.1, we calculate the association matrix $\boldsymbol{S}^{1\to 2} = \boldsymbol{\psi}_{\text{row}}\left(\boldsymbol{S}_{is}\right)$ and $\boldsymbol{S}^{2\to 1} = \boldsymbol{\psi}_{\text{row}}\left(\boldsymbol{S}_{is}{}^T\right)$. The corresponding results $\boldsymbol{S}^{1\to 2}$ and $\boldsymbol{S}^{2\to 1}$ are considered to match the targets in $\boldsymbol{I}_1$ to $\boldsymbol{I}_2$, and the targets in $\boldsymbol{I}_2$ to $\boldsymbol{I}_1$, respectively. Each element of $\boldsymbol{S}^{1\to 2}$ and $\boldsymbol{S}^{2\to 1}$ in the *i*-th row and *j*-th column are as follows, respectively:

$$\begin{aligned} s_{ij}^{1\to 2} &= \frac{\exp\left(\left(\boldsymbol{x}_1^i\right)^T \cdot \boldsymbol{x}_2^j / \tau\right)}{\sum_{j=0}^{M-1} \exp\left(\left(\boldsymbol{x}_1^i\right)^T \cdot \boldsymbol{x}_2^j / \tau\right)}, \\ s_{ij}^{2\to 1} &= \frac{\exp\left(\left(\boldsymbol{x}_2^j\right)^T \cdot \boldsymbol{x}_1^i / \tau\right)}{\sum_{i=0}^{N-1} \exp\left(\left(\boldsymbol{x}_2^j\right)^T \cdot \boldsymbol{x}_1^i / \tau\right)}, \end{aligned} \tag{2}$$

where τ is a temperature hyper-parameter [38].

According to the cycle association consistency, after forward association $\boldsymbol{S}^{1\to 2}$ and backward association $\boldsymbol{S}^{2\to 1}$, each object will match itself again ideally,

$$\boldsymbol{S}_{isc} = \boldsymbol{S}^{1\to 2}\boldsymbol{S}^{2\to 1}. \tag{3}$$

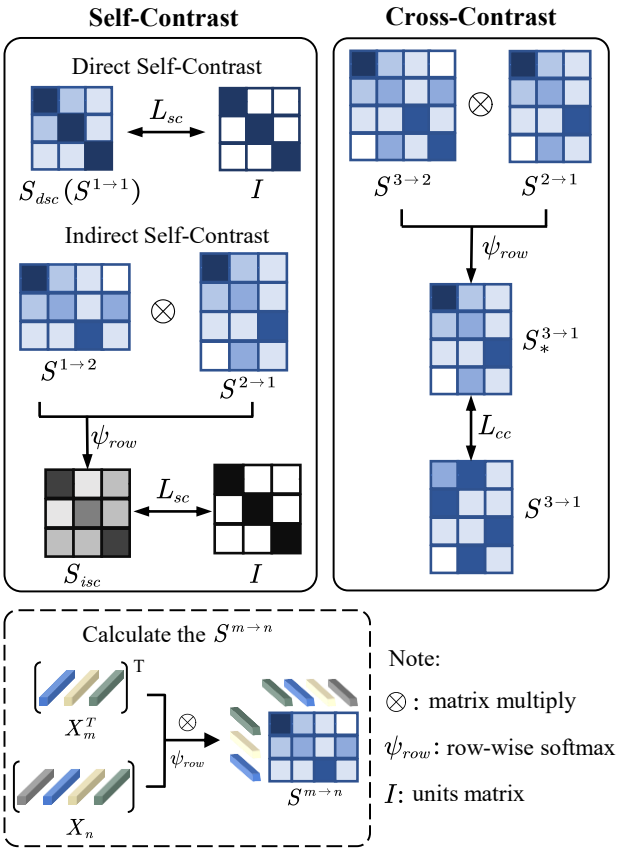

Figure 3. Self-Contrast and Cross-Contrast. We use three sets of indirect self-contrast and two sets of cross-contrast methods using different inputs. For the sake of brevity, we only show a set of specific feature calculation in each contrast.

The corresponding self-contrast loss can be formulated as:

$$L_{\text{sc}} = -\frac{1}{N}\left(\sum \log\left(\text{diag}\left(\boldsymbol{S}_{dsc}\right)\right) + \sum \log\left(\text{diag}\left(\boldsymbol{S}_{isc}\right)\right)\right), \tag{4}$$

where $\text{diag}()$ is to get a diagonal matrix.

Due to the self-contrast, it is obvious that the similarity between the same targets should be the largest, *i.e.*, the diagonal elements of S_{dsc} and S_{isc} obtained above are the largest and should be as close to 1 as possible.

3.1.2 Cross-Contrast Module

In almost all scenes of MOT, there is more or less object occlusion, and the similarity of these objects is generally low. Since MOT is an operation on multiple consecutive frames, the negative impact of these occluded objects could last for a long time. Considering theoretically the cross-frame matching results should be the same as the final results of continuous matching, we use a weaker unsupervised restriction, *i.e.*, direct (cross-frame) vs. indirect

(continuous-frame) association similarity comparison, to alleviate the above issue.

Specifically, we take three frames $\boldsymbol{I}_1, \boldsymbol{I}_2, \boldsymbol{I}_3 \in \mathbb{R}^{H\times W\times 3}$ as inputs, similar with Section 3.1.1, we calculate the target matching matrices between different frames, *i.e.*, $\boldsymbol{S}^{1\to 2}, \boldsymbol{S}^{2\to 1}, \boldsymbol{S}^{2\to 3}, \boldsymbol{S}^{3\to 2}, \boldsymbol{S}^{1\to 3}, \boldsymbol{S}^{3\to 1}$. As shown in the second column of Figure 3, we utilize $\boldsymbol{S}^{2\to 1}$ and $\boldsymbol{S}^{3\to 2}$ to compute the association matrix of $3 \to 1$, similarly use $\boldsymbol{S}^{1\to 2}$ and $\boldsymbol{S}^{2\to 3}$ to compute the association matrix of $1 \to 3$, as

$$\begin{aligned} \boldsymbol{S}_*^{1\to 3} &= \boldsymbol{\psi}_{\text{row}}\left(\boldsymbol{S}^{1\to 2}\boldsymbol{S}^{2\to 3}\right), \\ \boldsymbol{S}_*^{3\to 1} &= \boldsymbol{\psi}_{\text{row}}\left(\boldsymbol{S}^{3\to 2}\boldsymbol{S}^{2\to 1}\right). \end{aligned} \tag{5}$$

These matching matrices, which are generated indirectly through a middle frame, should be the same as direct-generated matching results.

We use relative entropy to measure the difference between the two matching distributions. KL divergence [14] is often used to compute the difference between two distributions P and Q,

$$KL(P\|Q) = \sum p(x)\log\frac{p(x)}{q(x)}, \tag{6}$$

but it is asymmetrical. We further utilize JS divergence [18] with symmetrical properties,

$$JSD(P\|Q) = \frac{1}{2}KL(P\|T) + \frac{1}{2}KL(Q\|T), \tag{7}$$

where $T = (P+Q)/2$. The corresponding cross-contrast loss is as follows,

$$L_{cc} = \frac{1}{N}JSD(\boldsymbol{S}_*^{1\to 3}\|\boldsymbol{S}^{1\to 3}) + \frac{1}{K}JSD(\boldsymbol{S}_*^{3\to 1}\|\boldsymbol{S}^{3\to 1}). \tag{8}$$

By enabling the continuous and cross-frame matching results to be close together, we use the different association results to mainly mitigate the differences in the same target caused by occlusion.

3.1.3 Ambiguity Contrast

There are occluded, lost, and emerging objects in the MOT, which will interfere with the whole learning process. We explore this problem and propose the ambiguity contrast module.

Based on the similarity between objects, we assume that objects with similarity greater than θ are the same object. The remaining objects with lower similarity are defined as ambiguous objects here. The low similarity mainly due to occlusion or the disappearance and appearance. In the occlusion case, objects of the same ID do exist, but the similarity is decreased due to the absence of original features and involvement of unrelated features. In the latter case, the similarity between the lost object and the newly emerged object is lower because there is really no target that can match it.

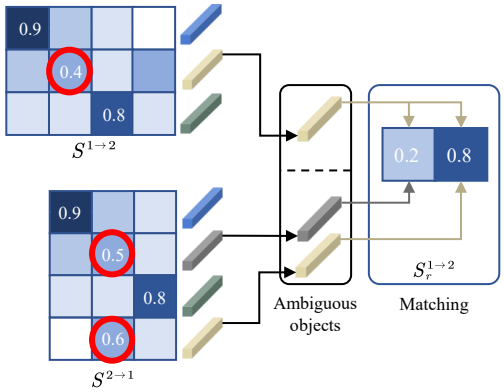

Figure 4. Ambiguity Contrast. For brevity, we only give the maximum similarity for each row, where certain objects have lower similarity to all other targets, *i.e.*, even the maximum similarity is below the threshold value, which is indicated by red circles in the figure. The corresponding feature embeddings are extracted and then matched again.

Our proposed method for ambiguous objects in the unsupervised training process is shown in Figure 4. We find the ambiguous objects in $\boldsymbol{I}_1$ according to the matching matrix of $\boldsymbol{S}^{1\to 2}$ based on the similarity. Similarly, we get ambiguous objects in $\boldsymbol{I}_2$ based on the matching result of $\boldsymbol{S}^{2\to 1}$. Then these objects are again subjected to similarity calculation to get the similarity matrix $\boldsymbol{S}_r{}^{1\to 2} \in \mathbb{R}^{D\times N_r}$ and $\boldsymbol{S}_r{}^{2\to 1} \in \mathbb{R}^{D\times M_r}$, where N_r and M_r are the number of ambiguous objects in $\boldsymbol{I}_1$ and $\boldsymbol{I}_2$, respectively. Finally, the loss of the module is calculated by minimum entropy:

$$\begin{aligned} L_{ac} = -\frac{1}{|N_r - M_r| + 1}\Bigg(&\frac{1}{N_r}\boldsymbol{S}_r{}^{1\to 2}\log\left(\boldsymbol{S}_r{}^{1\to 2}\right) \\ &+\frac{1}{M_r}\boldsymbol{S}_r{}^{2\to 1}\log\left(\boldsymbol{S}_r{}^{2\to 1}\right)\Bigg). \end{aligned} \tag{9}$$

When the number of ambiguous objects in two frames is equal, considering the two frames are very close to each other, we can assume that there are no disappearing or emerging objects, only occlusion exists, and the entropy should be as small as possible at this time. When the number of ambiguous objects in the two frames is not equal, it must contain disappearing or emerging objects. They are less similar to other objects because they cannot be matched, so we dynamically weaken the loss by adaptive coefficients.

3.2. UCSL for Unsupervised MOT

We apply UCSL on FairMOT [45], which composes of a backbone network, a detection head, and a re-identification head. For simplicity, the setting of the backbone and detec-

tion head follows FairMOT [45]. The overall architecture of UCSL is illustrated in Figure 2.

In the training stage, we follow the three contrast learning sub-modules in Section 3.1, and the complete loss function of ReID can be defined as follows:

$$L\left(I_t, I_{t-1}, I_{t-2}\right) = L_{sc} + L_{cc} + L_{ac} \tag{10}$$

where three consecutive frames I_t, I_{t-1} and I_{t-2} denote inputs. L_{sc}, L_{cc}, and L_{ac} denote self-contrast, cross-contrast, and ambiguity contrast losses mentioned above, respectively.

In the inference stage, video frames are fed into the network one by one. Then we obtain the corresponding detection results and ReID embeddings. We use the detection bounding boxes in the first frame to initialize multiple trajectories, and then use two-stage matching to complete object association. The overall association idea is also similar to FairMOT [45], using Kalman Filter [11] to predict the position of the objects and match bounding boxes with existing trajectories using embedding distance. For the trajectories and detections that are not matched, we match them using iou distance. Finally, the remaining unmatched detections are initialized as new objects, and the unmatched trajectories are saved for 30 frames and matched when they appear again.

4. Experiments

In this section, the proposed UCSL is evaluated on the MOT17 [24], MOT15 [15] and MOT20 [15]. The description of the datasets and the experimental setup is as follows, and next, we compare UCSL with the advanced approaches. Then, we show the evaluation of the effect of our model with ablation experiments.

4.1. Datasets

The proposed method is evaluated on MOT15, MOT17 and MOT20. MOT15 is the first dataset provided by MOT Challenge. It contains 22 video sequences, 11 of which are used for training and 11 for testing. The MOT15 is derived from older datasets and has different characteristics, such as fixed or moving cameras, different lighting environments, *etc.* MOT17 consists of 14 video sequences in total, 7 of which are used for training and 7 for testing, which is the most frequently used in MOT by far. MOT20 contains 4 training videos and 4 testing videos with more complex environments and greater crowd density, so MOT20 is more challenging than any previous datasets.

To evaluate our method, we use the standard MOT challenge metrics [2, 16, 22], mainly including Multi-Object Tracking Accuracy (**MOTA**), ID F1 Score (**IDF1**), Higher Order Tracking Accuracy (**HOTA**), Mostly Tracked objects (**MT**), Mostly Lost objects (**ML**), Number of False Positives (**FP**), Number of False Negatives (**FN**) and Number of Identity Switches (**IDS**), where the higher the first four items the better, the lower the last four items the better, and we use "↑" and "↓" to represent respectively.

4.2. Implementation Details

By default, UCSL is implemented based on the basis of the original FairMOT [45]. We take DLA-34 [43] as the backbone of the model and take the detection branch of the COCO dataset [19] pre-trained model to initialize our model parameters. We follow the most hyper-parameters settings of FairMOT [45].

We use conventional data enhancement approaches such as rotation, random cropping and horizontal flip, scale transformation, color jittering, *etc.*, and resize the input image size to 1088×608. We use the Adam optimizer [13] with the initial learning rate set to 10^{-4}, and the batch size set to 8. The similarity threshold θ in ambiguity contrast is 0.7. The model iterates 60 epochs on the MOT17 training set in the internal ablation experiments. We eventually train the corresponding dataset for 30 epochs on the basis of a pre-trained model of the CrowdHuman [29] dataset. The learning rate decays to 10^{-5} at the 20th epoch. Finally, we train our model on 4 RTX2080ti GPUs in about 10 hours.

4.3. Performance and Comparison

Comparison on MOT17. In this part, we compare our method with some other supervised and unsupervised methods on MOT17. In general, the performance of supervised methods is more advantageous purely in terms of metrics. As an unsupervised approach, we expect it to be as close as possible to state-of-the-art results. As shown in Table 1, we list some popular methods of joint detection and tracking or embeddings, and our method achieves considerable results, especially on IDF1 and HOTA. As the results provided by SimpleReID [12] are based on public detections, for a fairer comparison, we use the detection results of the same detector, i.e., CenterNet [46], to obtain the corresponding private detection-based results of simpleReID [12]. Since UTrack [21] is not tested on MOT17 test set, we replace it with our designed UCSL and conduct experiments under the same hardware conditions on MOT17. The results are shown in the tenth result row of Table 1. Based on the same FairMOT+CycAs model, although UTrack [21] and UCSL are very close on IDF1 and HOTA, our model improves 1.2 in terms of MOTA. Our model outperforms UTrack [21] in terms of ReID feature extraction with the same detection branch. We notice that IDS is not better compared to other methods, which may attribute to that UCSL tracks more trajectories and has a higher recall.

Performance on Other Datasets. In addition to MOT17, we also conduct experiments on MOT15 and MOT20, as shown in Table 1. Since FairMOT [45] uses additional MIX datasets for training besides the CrowdHuman

Method	Unsup	MOTA↑	IDF1↑	HOTA↑
MOT17				
TrackFormer [23]	No	74.1	68.0	57.3
TransTrack [31]	No	75.2	63.5	54.1
TransCenter [41]	No	73.2	62.2	54.5
QDTrack [26]	No	68.7	66.3	53.9
JDE [39]	No	56.7	55.0	45.1
CSTrack [17]	No	74.9	72.6	59.3
FairMOT [45]	No	73.7	72.3	59.3
SimpleReID* [12]	Yes	61.7	58.1	46.9
SimpleReID [12]	Yes	69.0	60.7	50.4
UTrack [21]	Yes	71.8	70.3	**58.4**
UCSL (ours)	Yes	**73.0**	**70.4**	**58.4**
MOT15				
EAMTT [28]	No	53.0	54.0	42.5
TubeTK [25]	No	58.4	53.1	42.7
RAR15 [8]	No	56.5	61.3	46.0
MTrack [42]	No	58.9	62.1	47.9
FairMOT [45]	No	55.0	60.2	45.9
UCSL (ours)	Yes	**59.1**	59.2	46.3
MOT20				
TransCenter [41]	No	58.5	49.6	54.1
MTrack [42]	No	63.5	69.2	55.3
FairMOT [45]	No	55.7	64.6	52.5
SimpleReID* [12]	Yes	53.6	50.6	41.7
SimpleReID [12]	Yes	61.8	54.8	45.5
UCSL (ours)	Yes	**62.4**	**63.0**	**52.3**

Table 1. Performance on MOT17, MOT15 and MOT20 test sets. "Unsup" means unsupervised training. "*" denotes using public detections. Bold and underline indicate unsupervised and supervised best metrics, respectively.

dataset, we train and test this method under the same conditions for a fair comparison. On MOT15, the performance of our unsupervised UCSL is metrically stronger than the supervised methods on MOTA, and achieves comparable overall performance on other metrics.

MOT20 is more complex than the scenarios in MOT15 relatively and has a larger amount of data, so the results of MOT20 are improved over those on MOT15. Our model outperforms the unsupervised SimpleReID [12] largely, especially on IDF1 and HOTA. Compared with supervised methods, the results show that our method is already comparable to them.

Performance under JDE paradigm. Our method is based on the JDE paradigm, considering FairMOT [45] as the baseline by default. We show the results of classical and our methods under the same paradigm, as shown in Table 2. Since the JDE [39] does not provide results on the MOT17 test set, we retest them under the same conditions. Due to the same paradigm, our approach also can be applied in other JDE-based methods, *e.g.*, JDE [39].

Method	MOTA ↑	IDF1 ↑	HOTA ↑
JDE(yolov5s) [39]	70.2	66.6	54.1
FairMOT [45]	73.7	72.3	59.3
CSTrack [17]	74.9	72.6	59.3
JDE(yolov5s) + Ours	69.6	68.0	55.7
FairMOT + Ours	73.0	70.4	58.4

Table 2. Mehtods on MOT17 under the same paradigm, JDE (joint detection and embeddings). "yolov5s" denotes detection branch baseline. The upper and lower parts are supervised and unsupervised methods, respectively.

Method	MOTA↑	IDF1↑	HOTA↑
YOLOX [9] + BYTE [44]	78.8	77.0	62.7
CenterNet [46] + BYTE [44]	73.1	70.0	58.9
CenterNet [46] + UCSL(ours)	73.0	70.4	58.4

Table 3. Comparison with ByteTrack [44] on MOT17. For a more intuitive comparison, we use YOLOX + BYTE to represent ByteTrack directly.

Comparison with TBD. Due to the contradiction between detection and ReID, compared with JDE, indeed, TBD (tracking-by-detection) paradigm could achieve a higher performance limit. But joint training methods output detections and embeddings simultaneously, balancing the accuracy and speed. So under JDE paradigm, we focus on exploring the impact of the unsupervised approach on it, rather than aiming for the state-of-the-art performance. To compare our method with TBD, we consider ByteTrack [44] as the representative for advanced TBD methods, First, it should be noticed that ByteTrack [44] uses trajectories interpolation on MOT17 dataset, which turns it into an offline approach. So we test ByteTrack [44] without interpolation on MOT17, as shown in the first result row of Table 3. In our approach, the detection branch uses CenterNet [46] by default, so the comparison between the second and third result lines of Table 3 demonstrates that the performance impact of our unsupervised approach is comparable to that of BYTE [44] with the same detector.

4.4. Ablation Studies

We conduct ablation experiments on the MOT17 test set, in which we test all contrast losses mentioned above as well as some settings about input frame interval and output ReID dimension.

Baseline. We are inspired by the method CycAs [38]. As shown in the first row in Table 4, only the triple loss of the original CycAs [38] is used for modeling, aiming to make the probability of the object matching back to itself reach a credible level to ensure cycle consistency. In this method, IDF1 and HOTA are 59.1 and 49.6, respectively.

Self-Contrast Loss. We use both the direct and indirect

L_{sc}		L_{cc}	L_{ac}	IDF1 ↑	HOTA ↑	MOTA ↑	MT ↑	ML ↓	FP ↓	FN ↓	IDS ↓
L_{dsc}	L_{isc}										
	CycAs [38]			59.1	49.6	69.5	38.7%	19.0%	31309	132816	7839
✓				61.5	50.8	69.9	39.3%	19.8%	**29475**	132954	7314
	✓			66.6	54.9	69.8	38.6%	18.1%	31341	133173	5997
✓	✓			67.2	55.0	69.4	38.6%	18.2%	32502	134808	5544
✓	✓	✓		**68.4**	**55.6**	69.8	39.6%	19.2%	32619	132228	5595
✓	✓	✓	✓	68.2	55.5	**70.5**	**40.8%**	**16.2%**	40569	**125004**	**5208**

Table 4. Performance with different losses on MOT17 test set. "CycAs" represents utilizing original loss function in CycAs [38]. L_{sc} represents self-contrast loss, where L_{dsc} and L_{isc} represents direct and indirect self-contrast loss, respectively. L_{cc} represents cross-contrast loss, L_{ac} represents ambiguity contrast loss.

Interval	MOTA ↑	IDF1 ↑	MT ↑	ML ↓
7	68.5	66.5	38.2%	20.7%
3	68.7	67.7	37.8%	20.6%
1	**70.5**	**68.2**	**40.8%**	**16.2%**

Table 5. Comparison of different input frame intervals. Based on the current frame, three consecutive frames are taken as input according to the number of frame intervals listed in the table. For the current frame t, for example, when the interval is 1, the inputs are frames t, $t-1$ and $t-2$.

ReID Dim	MOTA ↑	IDF1 ↑	MT ↑	ML ↓
64	69.8	**68.9**	38.2%	20.2%
128	70.5	68.2	**40.8%**	**16.2%**
256	**70.7**	68.4	37.7%	20.6%

Table 6. Comparison of different output ReID dimensions.

self-contrast losses to construct the model to extract ReID embeddings better. In the direct and indirect self-contrast subparts, we both use the intra-frame cross-entropy loss to construct the loss function, bringing the same objects closer together in the feature space and different targets further apart in the feature space. As seen from the second, third and fourth rows of Table 4, both direct and indirect self-contrast learning have little effect on MOTA, while have significantly improved the IDF1 and HOTA metrics and reduced IDS, demonstrating that our self-contrast similarity learning extracts more discriminative ReID embeddings.

Cross-Contrast Loss. We use cross- and consecutive-frames matching for cross-contrast similarity learning, with the aim of reducing the effect caused by mutual occlusion between objects. As can be seen from the fifth row in Table 4, on basis of self-contrast similarity learning, the cross-contrast improves IDF1 and HOTA metrics to 68.4 and 55.6 respectively, and there are also different degrees of improvement on other metrics.

Ambiguity Contrast Loss. In order to consider both the occluded, disappearing and emerging objects in MOT, we use ambiguity contrast to match these ambiguous objects again. From the last row of Table 4, one can see that after adding the ambiguity contrast based on the above two losses, the result has a more obvious improvement mainly in MOTA, MT and IDS, indicating that the method does have a positive effect on maintaining the object's trajectory.

Input Frame Interval. In our model, the default input is three consecutive frames. To show its superiority, we set different input intervals to train and test the corresponding model on MOT17, as shown in Table 5. Generally speaking, occlusion will last for a long time, but we find the larger the frame interval the weaker the performance, which may be surprising but explainable. Large interval is more suitable for supervised settings, where objects between any frames can be well matched with annotated ID labels. However, long intervals may cause drastic object changes without ID labels, making matching hard and errors accumulated. In addition, during training, there is an intersection between each input frame group. So, long-term temporal relation is taken into consideration just in an implicit manner.

Output ReID Dimension. In Table 6, we compare three different ReID embedding dimensions. As we can see, compared to the 64-dimension ReID embeddings, the 128-dimension performs better in terms of MOTA and MT metrics. The 256-dimension features have a similar improvement effect on the MOTA and IDF1 as the 128-dimension but consume more space and slow down the training and inference speed. For all these reasons, we choose the 128-dimension as the output dimension of the ReID branch.

5. Conclusions

We propose a simple but effective unsupervised method based on Contrastive Similarity Learning (UCSL). Specifically, we construct three learning types: self-contrast, cross-contrast and ambiguity contrast learning. Combining these sub-modules, the network is able to learn discriminative features consistently and reliably, and handle with occluded, lost and emerging objects simultaneously. Our unsupervised method outperforms existing unsupervised methods, and even surpasses some advanced supervised methods.

References

[1] Favyen Bastani, Songtao He, and Samuel Madden. Self-supervised multi-object tracking with cross-input consistency. *Advances in Neural Information Processing Systems*, 34:13695–13706, 2021. 3

[2] 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. 6

[3] 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. 2

[4] Guillem Brasó and Laura Leal-Taixé. Learning a neural solver for multiple object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6247–6257, 2020. 2

[5] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision*, pages 132–149, 2018. 1, 3

[6] Zuozhuo Dai, Guangyuan Wang, Weihao Yuan, Xiaoli Liu, Siyu Zhu, and Ping Tan. Cluster contrast for unsupervised person re-identification. *arXiv preprint arXiv:2103.11568*, 2021. 1, 3

[7] Patrick Dendorfer, Hamid Rezatofighi, Anton Milan, Javen Shi, Daniel Cremers, Ian Reid, Stefan Roth, Konrad Schindler, and Laura Leal-Taixé. Mot20: A benchmark for multi object tracking in crowded scenes. *arXiv preprint arXiv:2003.09003*, 2020. 2

[8] Kuan Fang, Yu Xiang, Xiaocheng Li, and Silvio Savarese. Recurrent autoregressive networks for online multi-object tracking. In *IEEE Winter Conference on Applications of Computer Vision*, pages 466–475, 2018. 7

[9] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 7

[10] Allan Jabri, Andrew Owens, and Alexei Efros. Space-time correspondence as a contrastive random walk. *Advances in neural information processing systems*, 33:19545–19560, 2020. 3

[11] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960. 6

[12] Shyamgopal Karthik, Ameya Prabhu, and Vineet Gandhi. Simple unsupervised multi-object tracking. *arXiv preprint arXiv:2006.02609*, 2020. 2, 6, 7

[13] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[14] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951. 5

[15] Laura Leal-Taixé, Anton Milan, Ian Reid, Stefan Roth, and Konrad Schindler. Motchallenge 2015: Towards a benchmark for multi-target tracking. *arXiv preprint arXiv:1504.01942*, 2015. 2, 6

[16] Yuan Li, Chang Huang, and Ram Nevatia. Learning to associate: Hybridboosted multi-target tracker for crowded scene. In *IEEE conference on computer vision and pattern recognition*, pages 2953–2960, 2009. 6

[17] Chao Liang, Zhipeng Zhang, Xue Zhou, Bing Li, Shuyuan Zhu, and Weiming Hu. Rethinking the competition between detection and reid in multiobject tracking. *IEEE Transactions on Image Processing*, 31:3182–3196, 2022. 7

[18] Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151, 1991. 5

[19] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755, 2014. 2, 6

[20] Yutian Lin, Xuanyi Dong, Liang Zheng, Yan Yan, and Yi Yang. A bottom-up clustering approach to unsupervised person re-identification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 8738–8745, 2019. 3

[21] Qiankun Liu, Dongdong Chen, Qi Chu, Lu Yuan, Bin Liu, Lei Zhang, and Nenghai Yu. Online multi-object tracking with unsupervised re-identification learning and occlusion estimation. *Neurocomputing*, 483:333–347, 2022. 2, 3, 6, 7

[22] 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*, 129(2):548–578, 2021. 6

[23] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixe, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8844–8854, 2022. 2, 7

[24] 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. 2, 6

[25] Bo Pang, Yizhuo Li, Yifan Zhang, Muchen Li, and Cewu Lu. Tubetk: Adopting tubes to track multi-object in a one-step training model. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6308–6318. 7

[26] 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. 3, 7

[27] Deva Ramanan and David A Forsyth. Finding and tracking people from the bottom up. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages II–II, 2003. 2

[28] Ricardo Sanchez-Matilla, Fabio Poiesi, and Andrea Cavallaro. Online multi-target tracking with strong and weak detections. In *European Conference on Computer Vision*, pages 84–99, 2016. 7

[29] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*, 2018. 6

[30] Qiuhong Shen, Lei Qiao, Jinyang Guo, Peixia Li, Xin Li, Bo Li, Weitao Feng, Weihao Gan, Wei Wu, and Wanli Ouyang. Unsupervised learning of accurate siamese tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8101–8110, 2022. 1, 2

[31] Peize Sun, Jinkun Cao, Yi Jiang, Rufeng Zhang, Enze Xie, Zehuan Yuan, Changhu Wang, and Ping Luo. Transtrack: Multiple object tracking with transformer. *arXiv preprint arXiv:2012.15460*, 2020. 2, 7

[32] ShiJie Sun, Naveed Akhtar, HuanSheng Song, Ajmal Mian, and Mubarak Shah. Deep affinity network for multiple object tracking. *IEEE transactions on pattern analysis and machine intelligence*, 43(1):104–119, 2019. 2

[33] Hideaki Uchiyama and Eric Marchand. Object detection and pose tracking for augmented reality: Recent approaches. In *18th Korea-Japan Joint Workshop on Frontiers of Computer Vision*, 2012. 1

[34] Dongkai Wang and Shiliang Zhang. Unsupervised person re-identification via multi-label classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10981–10990, 2020. 3

[35] Ning Wang, Yibing Song, Chao Ma, Wengang Zhou, Wei Liu, and Houqiang Li. Unsupervised deep tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1308–1317, 2019. 1, 2

[36] Xiaogang Wang. Intelligent multi-camera video surveillance: A review. *Pattern recognition letters*, 34(1):3–19, 2013. 1

[37] Xiaolong Wang, Allan Jabri, and Alexei A Efros. Learning correspondence from the cycle-consistency of time. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2566–2576, 2019. 3

[38] Zhongdao Wang, Jingwei Zhang, Liang Zheng, Yixuan Liu, Yifan Sun, Yali Li, and Shengjin Wang. Cycas: Self-supervised cycle association for learning re-identifiable descriptions. In *European Conference on Computer Vision*, pages 72–88, 2020. 3, 4, 7, 8

[39] 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, 2020. 1, 2, 7

[40] 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. 2

[41] Yihong Xu, Yutong Ban, Guillaume Delorme, Chuang Gan, Daniela Rus, and Xavier Alameda-Pineda. Transcenter: Transformers with dense queries for multiple-object tracking. *arXiv preprint arXiv:2103.15145*, 2021. 2, 7

[42] En Yu, Zhuoling Li, and Shoudong Han. Towards discriminative representation: Multi-view trajectory contrastive learning for online multi-object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8834–8843, 2022. 3, 7

[43] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2403–2412, 2018. 6

[44] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII*, pages 1–21, 2022. 7

[45] 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. 1, 2, 5, 6, 7

[46] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 6, 7