Supplementary Material

1. Abstract

We present additional experiments and ablation studies, as well as visualization results. In supplementary experiments, we give a more detailed comparison process. In ablation studies, we perform experiments using different types of setup methods. In the visualization results, we present the embedding visualizations, some results with superior performance as well as those with less than satisfactory performance, and finally analyze the corresponding reasons.

2. Supplementary Experiments

Comparasion with OUTrack. We perform a more detailed comparison with OUTrack on the MOT17 test set. We conduct the experiments under the same conditions, *i.e.*, all are trained 30 epochs based on the pre-trained model on the CrowdHuman dataset. The final feedback results are obtained from MOT benchmark. The first row of Table 1 shows the results of OUTrack, The second row is the replacement of our method with UTrack, and the third row is our unsupervised method UCSL. By comparison, we observe that although OUTrack uses an explicit occlusion estimation module and achieves advanced results on FP, our method performs better on other metrics, especially on MOTA, IDF1 and IDS.

Performance on TBD paradigm. To demonstrate the effectiveness of our method in TBD paradigm, we use the classical method DeepSort as the representative. We utilize our method, UCSL, to train the ReID network of DeepSort. To be consistent with the paper, we also use the default CenterNet as the private detector. As shown in Table 2, we test it on MOT17. As an unsupervised method, we outperform the corresponding supervised DeepSort in MOTA and HOTA metrics, and are also comparable to it in other aspects.

3. Additional Ablation Studies

In our overall loss, we do not use any weights to balance every loss, just simply add them up. We focus on the impact of the method itself rather than on improving performance by weighting the losses. In our unsupervised method, we introduce only one additional hyper-parameter, *i.e.*, the similarity threshold used in ambiguous contrast to determine whether an object is ambiguous or not. By default, we use a fixed value (0.7) for simplicity.

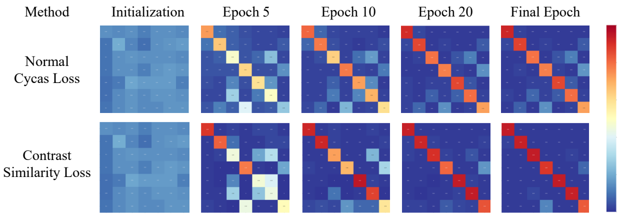

Figure 1. Comparison on object matching during training.

We explore the effect of different threshold settings, including using the fixed value and mean value, on the results in Table 3. Among given fixed values, although the results corresponding to 0.6 and 0.8 are comparable to 0.7 for some metrics, they are weaker for others. 0.7 corresponds to the strongest results in terms of overall performance. For another setting method "mean", after calculating the average similarity ($mean$), we treat the objects with similarity greater than the $mean$ and less than $1-mean$ as ambiguous objects for subsequent operations. Its results are very similar to those of the fixed value 0.8 and do not show better performance.

4. Visualization

In this section, we first show the embedding visualizations during training. Then we present the superior performance in most cases, and analyze the reasons for observed failure cases in the tracking process.

4.1. Embedding Visualization

Embedding matching. Figure 1 shows the matching results between objects according to ReID features during unsupervised training with the same input. The second column shows the results initialized with the same weights, followed by the results using the weights of the 5th, 10th, 20th and final epoch, respectively. We visualized the matching matrix in the training process for two groups in the ablation experiments, *i.e.*, using only CycAs loss and using the final contrast similarity loss to visually demonstrate the superiority of the final approach. Due to the overall trend being similar to the experimental trend in the table, we do not show the other loss combinations again here for brevity.

Method	MOTA ↑	IDF1 ↑	HOTA ↑	MT ↑	ML ↓	FP ↓	FN ↓	IDS ↓
OUTack[21]	72.1	68.8	57.3	39.5%	18.3%	**28065**	124833	4776
UCSL(UTrack[21])	71.8	70.3	**58.4**	**41.3%**	**17.1%**	35109	119130	4911
UCSL (ours)	**73.0**	**70.4**	**58.4**	40.1%	18.3%	30168	**118890**	**3540**

Table 1. Performance on MOT17. "UCSL(UTrack)" means replacing our UCSL with UTrack.

Method	MOTA ↑	IDF1 ↑	HOTA ↑	MT ↑	ML ↓	FP ↓	FN ↓	IDS ↓
DeepSort [40]	69.3	61.7	51.4	41.5%	16.6%	36867	129399	6882
DeepSort [40] + Ours	70.4	61.3	52.3	37.2%	18.3%	27342	131058	8379

Table 2. Performance on TBD paradigm on MOT17 dataset.

Embedding visualization. We use t-SNE to visualize embeddings acquired by our method during training. We expect embeddings of the same objects to be as close as possible and those of different objects to be as far away from each other as possible. As shown in Figure 2, different colors represent different objects. We can see that after pre-training the objects are roughly distinguished. As the training goes on, the features of identical objects are very close together and it is easier to separate different objects on the feature space, just as we expect.

4.2. Superior Performance

Figure 3 shows the tracking results of our method in certain scenarios. We take MOT17-12, MOT17-08, MOT17-07 and MOT17-03 datasets as examples, which contain a variety of common scenes. We can see that almost all objects in the figure can be detected and tracked, even if some are obscured during the movement. When objects disappear and then reappear, they still maintain the original corresponding ID. These examples demonstrate that our approach keeps trajectories consistent and thus performs well in the majority of tracking cases.

4.3. Failure Case Analysis

The previous section has demonstrated the superior performance of our method, which is able to track objects continuously in most cases. However, some tracking errors, such as FP, FN and IDS, are still unavoidable. These errors are mainly caused by the following reasons.

Occlusion. As mentioned in the previous introduction, occlusion is one of the most frequent problems in MOT, especially in crowded scenes, where occlusion can happen almost every moment. Although our approach has mitigated many of the effects of occlusion, the problem persists. As shown in Figure 4, some objects in these frames are occluded by other objects or non-objects, which combines with the fact that these objects are inherently small in size relative to the whole frame. So there are few features available after being occluded, even so few that they are overlooked. Therefore, these objects are easily missed.

Hard cases. As can be seen in the first and second images in Figure 4, the overall environment is very dark, causing some objects to almost blend in with the environment. This situation is very challenging for the network. Even though this type of video is present in the training dataset, the network balances the tracking of multiple scenes to enhance its generalization rather than focusing on this particular scenario. Therefore, only some of the more obvious objects can be detected and tracked.

In addition, the quality of the video itself greatly affects the tracking performance. The corresponding video of the third and fourth images in Figure 4 has a low resolution, so intuitively the appearance of objects is ambiguous. The learned features are also relatively ambiguous, which is not conducive to distinguishing them from other objects and affects the association.

θ	MOTA ↑	IDF1 ↑	HOTA ↑	MT ↑	ML ↓	FP ↓	FN ↓	IDS ↓
0.5	72.5	69.7	58.0	39.1%	19.4%	28989	122694	3495
0.6	72.6	**70.6**	**58.4**	38.3%	19.5%	29664	121626	3465
0.7	**73.0**	70.4	**58.4**	**40.1%**	**18.3%**	30168	**118890**	3540
0.8	72.5	70.2	58.2	38.7%	19.1%	**28788**	122868	3477
mean	72.2	70.5	**58.4**	38.7%	19.5%	29418	124158	**3399**

Table 3. Comparison of different settings for ambiguous contrast similarity threshold.

Method	Unsup	MOTA↑	IDF1↑	HOTA↑	MT ↑	ML ↓	FP ↓	FN ↓	IDS ↓
MOT17									
TrackFormer [23]	No	74.1	68.0	57.3	47.2%	10.4%	34602	108777	2829
TransTrack [31]	No	75.2	63.5	54.1	55.3%	10.2%	50157	86442	3603
TransCenter [41]	No	73.2	62.2	54.5	40.3%	18.5%	23112	123738	4614
QDTrack [26]	No	68.7	66.3	53.9	40.6%	21.9%	26589	146643	3378
JDE [39]	No	56.7	55.0	45.1	25.7%	30.1%	35700	202824	5526
CSTrack [17]	No	74.9	72.6	59.3	41.5%	17.5%	23847	114303	3567
FairMOT [45]	No	73.7	72.3	59.3	43.2%	17.3%	27507	117477	3303
SimpleReID* [12]	Yes	61.7	58.1	46.9	27.2%	32.3%	**16872**	197632	**1864**
SimpleReID [12]	Yes	69.0	60.7	50.4	**41.5%**	**16.7%**	36933	129852	8112
UTrack [21]	Yes	71.8	70.3	**58.4**	41.3%	17.1%	35109	119130	4911
UCSL (ours)	Yes	**73.0**	**70.4**	**58.4**	40.1%	18.3%	30168	**118890**	3540
MOT15									
EAMTT [28]	No	53.0	54.0	42.5	35.9%	19.6%	7538	20590	7538
TubeTK [25]	No	58.4	53.1	42.7	39.3%	18.0%	5756	18961	854
RAR15 [8]	No	56.5	61.3	46.0	45.1%	14.6%	9386	16921	428
MTrack [42]	No	58.9	62.1	47.9	38.1%	17.5%	6314	18177	750
FairMOT [45]	No	55.0	60.2	45.9	39.5%	13.5%	8635	18045	946
UCSL (ours)	Yes	**59.1**	59.2	46.3	**46.7%**	**11.8%**	8358	**15742**	1013
MOT20									
TransCenter [41]	No	58.5	49.6	54.1	48.6%	14.9%	64217	146019	4695
MTrack [42]	No	63.5	69.2	55.3	68.8%	7.5%	96123	86964	6031
FairMOT [45]	No	55.7	64.6	52.5	67.4%	6.9%	131548	90421	7018
SimpleReID* [12]	Yes	53.6	50.6	41.7	30.3%	25.0%	**6439**	231298	**4335**
SimpleReID [12]	Yes	61.8	54.8	45.5	60.4%	8.8%	78101	110594	9107
UCSL (ours)	Yes	**62.4**	**63.0**	**52.3**	**68.0%**	**7.0%**	104164	**84799**	5459

Table 4. 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.

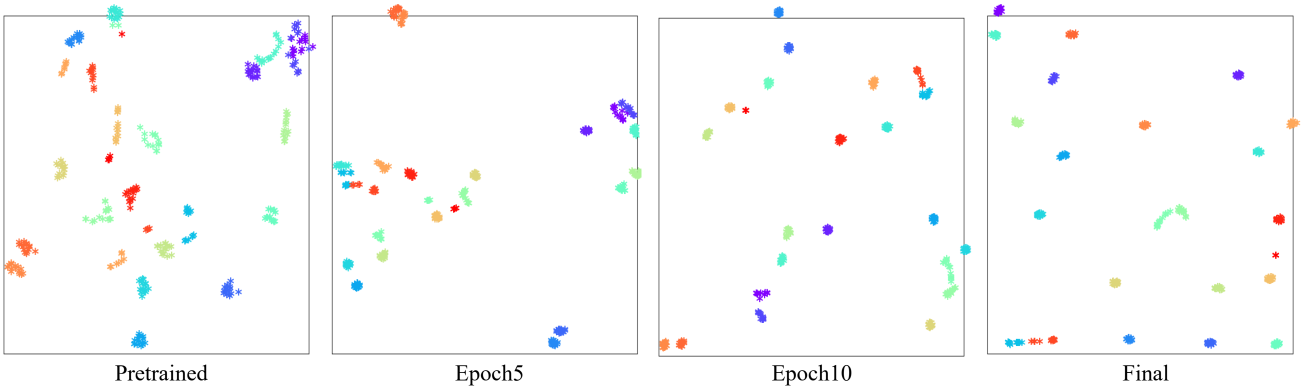

Figure 2. Visualization of instance embeddings using t-SNE.

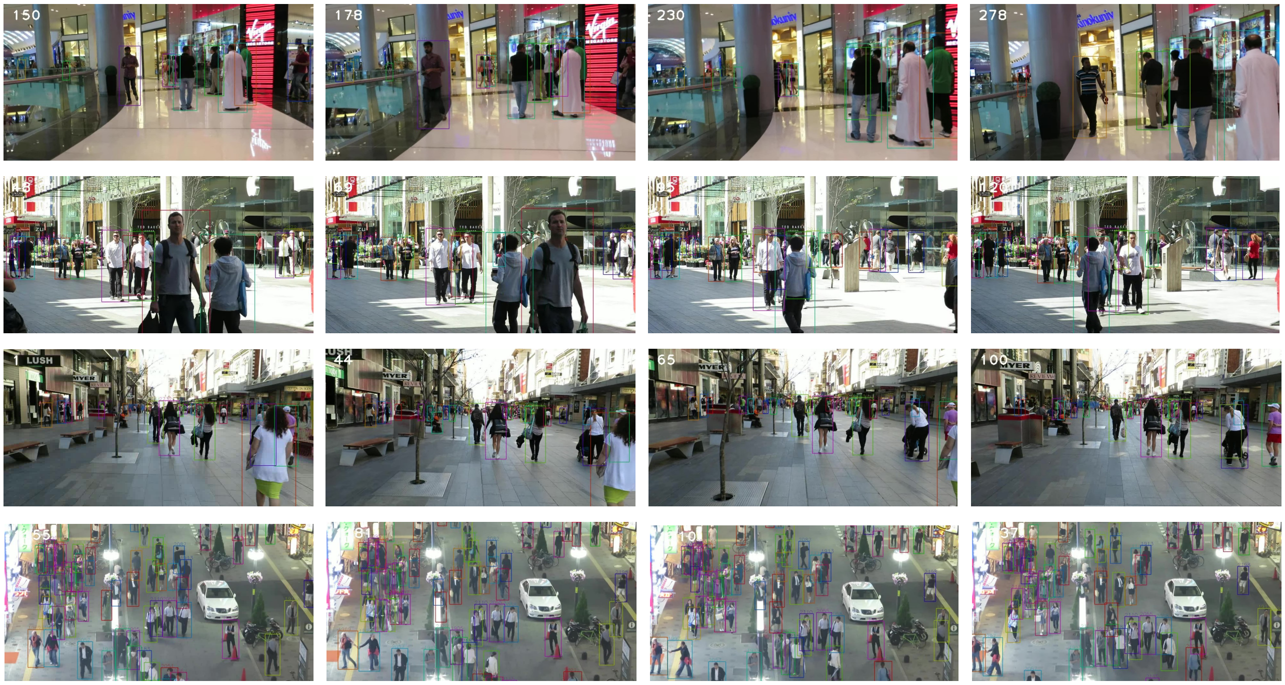

Figure 3. Superior performance.

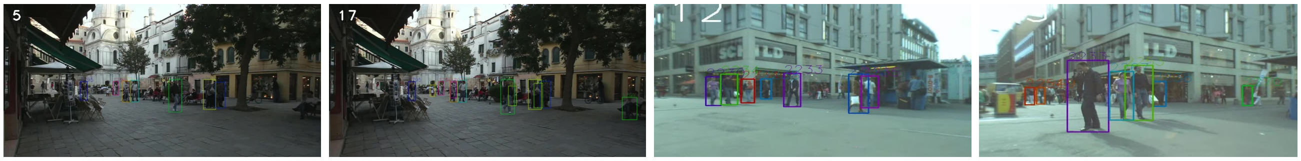

Figure 4. Failure cases.