Detection Recovery in Online Multi-Object Tracking with Sparse Graph Tracker

Jeongseok Hyun[1*] Myunggu Kang[2] Dongyoon Wee[2] Dit-Yan Yeung[1]
[1]The Hong Kong University of Science and Technology [2]Clova AI, NAVER Corp.

In this appendix, we provide additional experiment results, visualization results, and detailed analysis of detection recovery which are not included in the main paper.

A. Additional Implementation Details

We use shorter epochs when CrowdHuman dataset is not included but only MOT or HiEve datasets are used for training. The models are trained for 30 epochs and the learning rate is dropped from $2e^{-4}$ to $2e^{-5}$ at 20 epoch. The same loss weights are adopted, but we use w_{edge} as 1, instead of 0.1. For inference, the same threshold values are used regardless of whether CrowdHuman [8] dataset is used or not. In the ablation experiments, we deploy the detection threshold values of (τ_{init}, τ_D, $\tau_{D_{low}}$) as (0.5, 0.4, 0.2) for FairMOT [13] and BYTE [12] following the official code of ByteTrack[1].

Our implementation is based on detectron2 framework[2]. Regarding training time of the ablation experiments, with two NVIDIA V100 GPUs, 3 hours are spent on training SGT without CrowdHuman dataset, while 2 days are taken if it is included. We observe that the inference speed is affected by the version of NVIDIA driver and the number of CPUs. The reported inference speed is measured on NVIDIA driver version of 460.73.01 with CUDA version of 10.1.

B. Additional Experiment Results

B.1. MOT Detection Challenge Evaluation Results

For evaluation metrics of MOT17/20 Detection benchmarks, we choose precision, recall, F1, and average precision (AP) [5]. As shown in Table 1, SGT achieves the best in every metric on MOT17/20 Detection benchmarks. SGT outperforms GSDT [10] which is also based on CenterNet [15] and GNNs. As stated in Section 2.2 of the main paper, GNNs in GSDT aggregate the current and past features to enhance the current image features. However, relational features used for association in GSDT are still limited to pairwise features while relational features in SGT are updated by GNNs to become multi-hop features. Consequently, low-scored detections are not tracked in GSDT. The high recall supports the effectiveness of detection recovery in SGT. On the other hand, the high precision indicates that detection recovery is achieved without introducing extra false positives.

Table 1. Evaluation results of the MOT17/20 Detection benchmarks.

Benchmark	Method	AP↑	Recall↑	Precision↑	F1↑
MOT 17Det	FRCNN [7]	0.72	77.3	89.8	83.1
	GSDT [10]	0.89	90.7	87.8	89.2
	YTLAB [1]	0.89	91.3	86.2	88.7
	SGT (ours)	**0.90**	**93.1**	**92.5**	**92.8**
MOT 20Det	ViPeD20 [2]	0.80	86.5	68.1	76.2
	GSDT [10]	0.81	88.6	90.6	89.6
	SGT (ours)	**0.90**	**91.6**	**92.6**	**92.1**

B.2. Visualization Comparison

Figure 1 shows qualitative comparison between SGT and others [13, 14, 10, 9] using the same detector. In MOT17, the partially occluded people have low detection confidence score and they are not used as tracking candidates in FairMOT, GSDT, and CorrTracker since they only use high-scored detections for tracking. As a result, these occluded people are missed. In MOT20, existing methods use lower detection threshold (τ_D) than MOT17 due to frequent occlusions in MOT20. However, they suffer from FPs while SGT does not have such FPs without missed detections.

B.3. Additional Ablation Experiments

Top-K vs Detection threshold. In Table 6 of the main paper, the robustness of top-K sampling in SGT is shown by the consistent performance with a wide range of K values and different K values for training and inference. Here, we experiment about using a low value of detection threshold, τ_D, that is an alternative option for choosing tracking candidates instead of top-K sampling.

According to Table 2, when SGT is trained with tracking candidates sampled by top-K whose K is 100 or 300, it shows the consistent performance with both $K = \{50, 300\}$ and $\tau_D = \{0.01, 0.1\}$ while marginal degradation is ob-

*The work was partially done during an intern at Clova AI.
[1]https://github.com/ifzhang/ByteTrack
[2]https://github.com/facebookresearch/detectron2

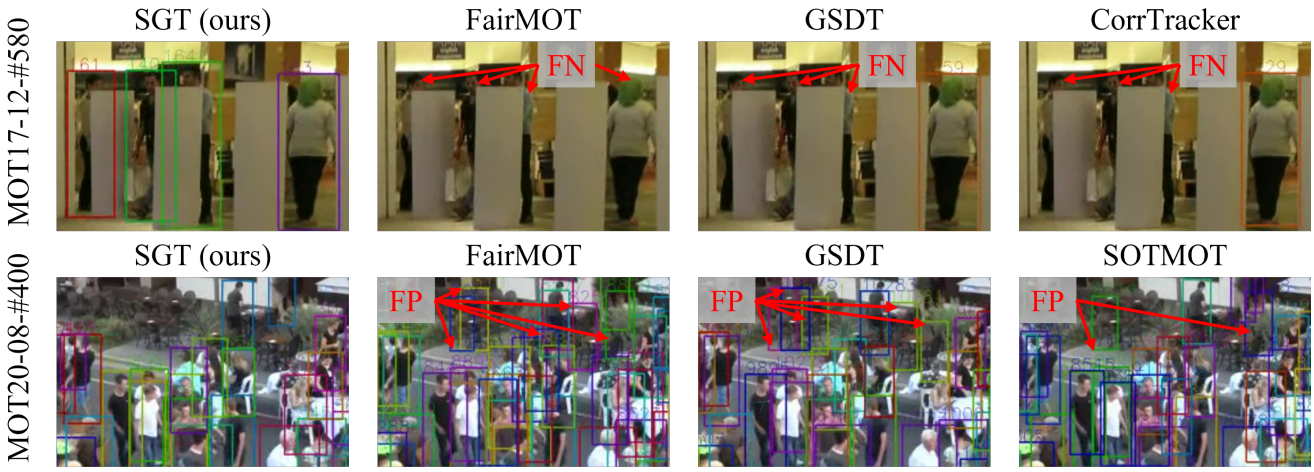

Figure 1. Qualitative comparison of CenterNet [15] based online JDT models [13, 10, 9, 14] on the MOT17/20 test datasets.

Table 2. Ablation study of threshold and top-K for choosing detections for tracking candidates. CrowdHuman dataset [8] is additionally used for training.

train	inference	MOTA↑	IDF1↑	MT↑	FP↓	FN↓	IDS↓
$K = 300$	$K = 300$	73.8	74.7	52.5	2620	11047	476
$K = 300$	$K = 50$	73.8	74.0	52.5	2531	11159	474
$K = 300$	$\tau_D = 0.01$	**74.1**	**74.9**	**53.4**	2492	**11050**	**459**
$K = 300$	$\tau_D = 0.1$	73.5	73.4	51.6	2331	11398	604
$K = 300$	$\tau_D = 0.3$	72.8	72.4	50.7	**2256**	11605	843
$K = 100$	$K = 300$	74.1	**76.5**	**53.1**	2544	10971	460
$K = 100$	$K = 50$	**74.2**	75.9	**53.1**	2505	11000	458
$K = 100$	$\tau_D = 0.01$	**74.2**	76.2	**53.1**	2530	**10961**	**451**
$K = 100$	$\tau_D = 0.1$	74.0	75.9	52.2	2391	11146	538
$K = 100$	$\tau_D = 0.3$	73.3	73.6	51.0	**2348**	11313	795
$\tau_D = 0.01$	$K = 300$	**74.4**	76.0	**53.4**	2542	**10774**	498
$\tau_D = 0.01$	$K = 50$	74.3	**77.2**	53.1	2475	10920	**485**
$\tau_D = 0.01$	$\tau_D = 0.01$	**74.4**	76.2	53.4	2543	10780	500
$\tau_D = 0.01$	$\tau_D = 0.1$	73.2	73.1	52.2	2438	11034	1002
$\tau_D = 0.01$	$\tau_D = 0.3$	71.9	71.4	51.0	**2348**	11344	1480
$\tau_D = 0.1$	$K = 300$	73.9	**76.0**	52.5	2851	10764	**470**
$\tau_D = 0.1$	$K = 50$	**74.0**	**76.0**	**53.1**	**2807**	10745	484
$\tau_D = 0.1$	$\tau_D = 0.01$	**74.0**	75.7	52.5	2842	**10722**	496
$\tau_D = 0.1$	$\tau_D = 0.1$	73.9	75.5	52.8	2849	10789	486
$\tau_D = 0.1$	$\tau_D = 0.3$	73.7	74.5	52.5	2885	10804	541

served with $\tau_D = 0.3$. On the other hand, large drop in MOTA and IDF1 is shown when SGT is trained with $\tau_D = 0.01$ and evaluated with $\tau_D = 0.3$. In contrast, SGT trained with $\tau_D = 0.1$ shows the consistent performance when $\tau_D = 0.3$ is used for selecting tracking candidates for inference.

Based on the results, detection threshold can be viewed as the hyperparameter that should be carefully tuned. Although K is also the hyperparameter, K is more intuitive value representing the maximum number of objects to be tracked and is easier to be decided than τ_D which is affected by many factors (*e.g.*, model architecture, training method). For this reason, we adopt top-K sampling method to include low-scored detections in SGT.

Table 3. Performance comparison of FairMOT [13] and SGT with different backbone networks: ResNet-18/101 [4], DLA-34 [11], and hourglass-104 [6]. The models are trained using the extra CrowdHuman [8] dataset.

Model	Backbone	MOTA↑	IDF1↑	MT↑	ML↓	FP↓	FN↓	IDS↓
FairMOT [13]	Res-18	66.1	**69.9**	40.1	20.4	**2036**	16029	**265**
SGT (ours)	Res-18	**68.4**	69.3	**47.2**	**15.6**	2359	**14086**	659
FairMOT [13]	Res-101	70.2	72.2	47.8	14.5	**2545**	13178	**364**
SGT (ours)	Res-101	**71.1**	**72.4**	**53.4**	**12.4**	3197	**11678**	720
FairMOT [13]	DLA-34	72.2	74.7	47.8	18.0	2660	12025	**336**
SGT (ours)	DLA-34	**74.2**	**76.3**	**53.1**	**13.3**	**2514**	**10978**	451
FairMOT [13]	HG-104	74.4	**77.1**	54.0	12.1	**2636**	10844	**344**
SGT (ours)	HG-104	**74.8**	**77.1**	**56.0**	**10.9**	2713	**10454**	428

Table 4. Effect of the number of the edges for each criterion. Center distance, IoU, and cosine similarity are three criteria used in SGT.

#edges per criterion	MOTA↑	IDF1↑	MT↑	FP↓	FN↓	IDS↓
5	71.0	72.8	47.2	2450	12590	624
10	**71.3**	**73.8**	46.6	**2190**	12742	**588**
20	71.1	73.6	**47.8**	2308	**12535**	755

Different backbone networks. Table 3 shows the performance comparison of SGT and FairMOT [13] with different backbone networks [4, 11, 6] used in CenterNet [15]. For hourglass backbone, we use the image size of $(H, W) = (640, 1152)$, instead of $(608, 1088)$ since only a multiple of 128 is allowed. SGT achieves lower FN and ML, and higher MT and MOTA than FairMOT across all backbone networks. In other words, SGT has less missed detections and more long-lasting tracklets than FairMOT. This result is corresponding to our motivation of detection recovery by tracking in SGT. Especially, SGT shows larger improvement in MOTA with the small backbone networks (*e.g.*, resnet18 and dla34). When MOT models are deployed with the limited resource of hardware, SGT can be served as an effective solution.

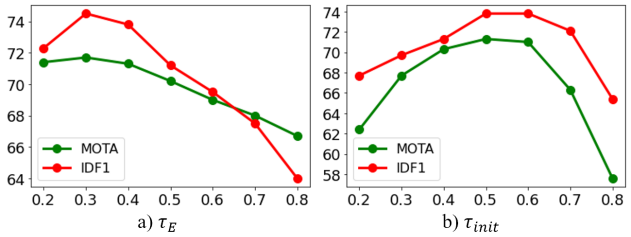

Figure 2. Sensitivity analysis of τ_E and τ_{init}.

Table 5. Ratio of recovered detections over all detections in each sequence of MOT17/20 test datasets.

Benchmark	Sequence	Recovery Ratio (%)
MOT17	MOT17-01	12.0
	MOT17-03	1.8
	MOT17-06	6.8
	MOT17-07	10.0
	MOT17-08	**14.6**
	MOT17-12	10.3
	MOT17-14	12.6
MOT20	MOT20-04	3.8
	MOT20-06	29.0
	MOT20-07	4.9
	MOT20-08	**35.4**

The number of edges. In SGT, nodes across frames are sparsely connected if only they are close in either Euclidean or feature space. Specifically, n_{t1}^i is connected to the nodes of N_{t2} using three criteria: center distance, IoU, and cosine similarity. We choose nodes for each criterion and remove the duplicates. Table 4 shows the result of experimenting with different number of nodes for each criterion. Although the best performance is achieved with 10, there is only marginal decrease in MOTA and IDF1 with 5 and 20. Thus, this is also robust hyperparameter.

Sensitivity of τ_{init} and τ_E. The robustness of K, which is the number of tracking candidates, has been shown through the extensive ablation experiments. Here, we measure the sensitivity of τ_{init} and τ_E as well. As shown by Figure 2, τ_{init} is a sensitive threshold value since it decides initialization of new tracklets. On the other hand, τ_E is the minimum edge score used for matching. It is robust within the range between 0.2 and 0.4 since correct matching may have high edge score and the node classifier prevents false positive matching.

C. Analysis of Detection Recovery

C.1. Ratio of Recovered Detections

According to Table 5, MOT17-08 and MOT20-08 are the sequences that SGT outputs the highest ratio of recovered detections whose confidence score is lower than $\tau_{init} = 0.5$. Table 6 shows that SGT surpasses CorrTracker in MOTA by 2.7% in MOT17-08 while SGT shows lower MOTA than CorrTracker in MOT17-06 whose recovery ratio is low. In MOT20, similar trend is observed that SGT achieves larger improvement of MOTA in MOT20-08 than MOT20-07.

Table 6. Evaluation result per sequence of MOT17/20 test dataset. We choose one with high recovery ratio and the other one with low recovery ratio.

Method	MOTA↑	IDF1↑	MT↑	ML↓	FP↓	FN↓	IDS↓
MOT17-06 (6.8% recovery)							
SGT (Ours)	65.5	63.2	**48.2**	**12.2**	942	**2917**	210
FairMOT [13]	64.1	65.9	40.1	18.5	526	3533	176
GSDT [10]	63.0	62.0	40.1	21.2	681	3500	180
CorrTracker [9]	**66.2**	**68.2**	41.0	17.1	**465**	3346	**171**
MOT17-08 (14.6% recovery)							
SGT (Ours)	**52.6**	44.0	**32.9**	**14.5**	1076	**8546**	347
FairMOT [13]	42.2	42.0	22.4	28.9	**776**	11191	**237**
GSDT [10]	44.0	40.5	26.3	22.4	991	10523	323
CorrTracker [9]	49.9	**46.7**	25.0	17.1	1137	9201	250
MOT20-07 (4.9% recovery)							
SGT (Ours)	**77.9**	**71.4**	**76.6**	2.7	2277	4774	**254**
FairMOT [13]	75.6	70.0	**76.6**	**0.9**	2988	**4770**	333
GSDT [10]	75.0	68.1	64.0	1.8	**1870**	6115	282
SOTMOT [14]	72.6	71.2	**76.6**	2.7	3675	5066	317
MOT20-08 (35.4% recovery)							
SGT (Ours)	**54.1**	54.5	26.7	26.2	**2434**	32625	**468**
FairMOT [13]	27.0	49.5	**41.9**	**14.7**	32104	**23447**	981
GSDT [10]	39.4	48.9	22.5	32.5	9916	36420	608
SOTMOT [14]	43.1	**55.1**	35.6	19.9	16025	27216	863

C.2. Recall per Visibility Level

We measure the recall ratio for different visibility levels of objects and compare them of different models as shown in Figure 3. Both BYTE [12] and SGT show higher recall value for low visibility levels than FairMOT [13] since they perform association of low-scored detections. When objects are almost invisible with the visibility in the range of (0.0, 0.3), SGT outperforms BYTE in terms of the recall. These results indicate that SGT successfully tracks the low-scored detections caused by occlusion, and SGT is robust against partial occlusion. Also, the effectiveness of node classifier preventing false positive recovery is demonstrated through higher precision value of SGT than that of BYTE.

C.3. Visualization of Recovered Detections

Figure 4 shows the cases of detection recovery in the MOT20 test dataset. In the first row, people indicated by the blue and brown bounding boxes occlude each other. In the frame #33, their detection scores are higher than τ_{init} which is detection threshold value to initialize new tracklet. However, from frame #34 to #37, their detection scores are between 0.3 or 0.4 which are lower than τ_{init}, nevertheless, SGT successfully tracks them. If only high-scored detections are used for association, they are failed to track, leading to missed detections and disconnected tracklets. The

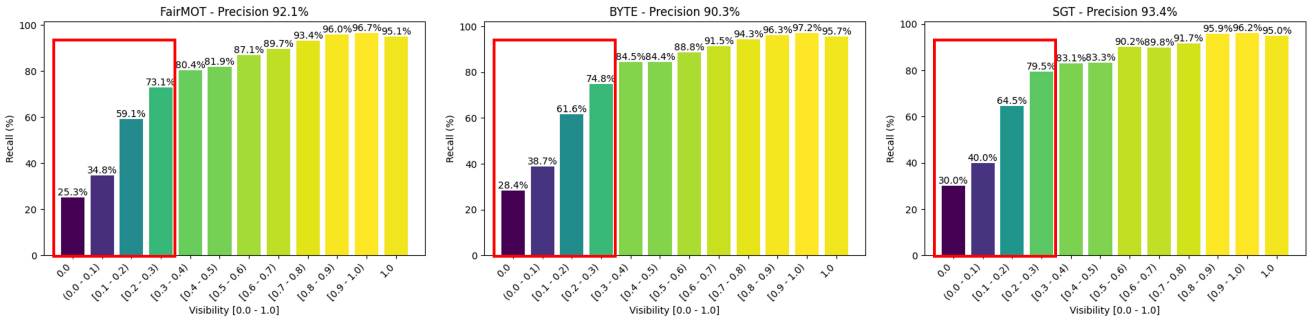

Figure 3. Recall ratio comparison of FairMOT [13], BYTE [12] on top of FairMOT [13], and SGT for different visibility levels of objects in MOT17 validation dataset.

Figure 4. Detection recovery cases in MOT20 test dataset [3]. We show the annotation of each bounding box in the format of "{id}-{detection score}". The tracklets in the red circles are recovered in the next frames.

Figure 5. Example of ID switch caused by non-human occluder.

second row of Figure 4 is another example of detection recovery.

D. Discussion

High IDS in MOT16/17. In Section 4.2 of the main paper, we stated that non-human occluders in MOT16/17 result in high IDS in MOT16/17 compared to low IDS in MOT20. Figure 5 shows the example whose video is taken from a department store, where non-human occluders commonly exist.

References

[1] Zhaowei Cai, Quanfu Fan, Rogerio S Feris, and Nuno Vasconcelos. A unified multi-scale deep convolutional neural network for fast object detection. In *ECCV*, pages 354–370. Springer, 2016.

[2] Luca Ciampi, Nicola Messina, Fabrizio Falchi, Claudio Gennaro, and Giuseppe Amato. Virtual to real adaptation of pedestrian detectors. *Sensors*, 20(18):5250, 2020.

[3] 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.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[5] 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 *ECCV*, pages 740–755. Springer, 2014.

[6] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, pages 483–499. Springer, 2016.

[7] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *NeurIPS*, 28, 2015.

[8] 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.

[9] Qiang Wang, Yun Zheng, Pan Pan, and Yinghui Xu. Multiple object tracking with correlation learning. In *CVPR*, pages 3876–3886, 2021.

[10] Yongxin Wang, Kris Kitani, and Xinshuo Weng. Joint object detection and multi-object tracking with graph neural networks. In *ICRA*, pages 13708–13715, 2021.

[11] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *CVPR*, pages 2403–2412, 2018.

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

[13] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *IJCV*, 129(11):3069–3087, 2021.

[14] Linyu Zheng, Ming Tang, Yingying Chen, Guibo Zhu, Jinqiao Wang, and Hanqing Lu. Improving multiple object tracking with single object tracking. In *CVPR*, pages 2453–2462, 2021.

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