Learnable Graph Matching: Incorporating Graph Partitioning with Deep Feature Learning for Multiple Object Tracking
Supplementary Material

Jiawei He[1,3] Zehao Huang[2] Naiyan Wang[2] Zhaoxiang Zhang[1,3,4]
[1] Institute of Automation, Chinese Academy of Sciences (CASIA) [2] TuSimple
[3] School of Artificial Intelligence, University of Chinese Academy of Sciences (UCAS)
[4] Centre for Artificial Intelligence and Robotics, HKISI_CAS
{hejiawei2019, zhaoxiang.zhang}@ia.ac.cn {zehaohuang18, winsty}@gmail.com

A. Gradients of the Graph Matching Layer

As described in Section 4.3 of our main paper, the gradients of the graph matching layer we need for backward can be derived from the KKT conditions with the help of the implicit function theorem. Here, we show the details of deriving the gradients.

For a quadratic programming (QP), the standard formulation is as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \frac{1}{2}x^\top Q(\theta)x + q(\theta)^\top x \\ \text{subject to} \quad & G(\theta)x \le h(\theta) \\ & A(\theta)x = b(\theta). \end{aligned} \tag{A}$$

So the Lagrangian is given by

$$L(x,\nu,\lambda) = \frac{1}{2}x^\top Qx + \lambda^\top(Gx-h) + q^\top x + \nu^\top(Ax-b), \tag{B}$$

where, ν and λ are the dual variables.
The (x^*,λ^*,ν^*) are the optimal solution if and only if they satisfy the KKT conditions:

$$\begin{aligned} \nabla_x L(x^*,\lambda^*,\nu^*) &= 0 \\ Qx^* + q + A^\top\nu^* + G^\top\lambda^* &= 0 \\ Ax^* - b &= 0 \\ \mathrm{diag}(\lambda^*)(Gx^* - h) &= 0 \\ Gx^* - h &\le 0 \\ \lambda^* &\ge 0. \end{aligned} \tag{C}$$

We define the function

$$g(x,\lambda,\nu,\theta) = \begin{bmatrix} \nabla_x L(x,\lambda,\nu,\theta) \\ \mathrm{diag}(\lambda)\lambda^\top(G(\theta)x - h(\theta)) \\ A(\theta)x - b(\theta) \end{bmatrix}, \tag{D}$$

and the optimal solution x^*,λ^*,ν^* satisfy the euqation $g(x^*,\lambda^*,\nu^*,\theta) = 0$.

According to the implicit function theorem, as proven in [1], the gradients where the primal variable x and the dual variables ν and λ are the optimal solution, can be formulated as

$$J_\theta x^* = -J_x g(x^*,\lambda^*,\nu^*,\theta)^{-1} J_\theta g(x^*,\lambda^*,\nu^*,\theta), \tag{E}$$

where, $J_x g(x^*,\lambda^*,\nu^*,\theta)$ and $J_\theta g(x^*,\lambda^*,\nu^*,\theta)$ are the Jacobian matrices. Each element of them is the partial derivative of function g with respect to variable x and θ, respectively.

B. Pseudo-code of Our Algorithm

To make our algorithm clear and easy to understand, we show the pseudo code of our GMTracker algorithm in Alg. A. The input of the algorithm is the detection set $\mathcal{D}^t = \{D_1^t, D_2^t, \cdots, D_{n_d}^t\}$ and tracklet set $\mathcal{T}^t = \{T_1^t, T_2^t, \cdots, T_{n_t}^t\}$, defined in Section 4.1 of our main paper. And the output is the new tracklet set $\mathcal{T}^{t+1}$ to be associated in the next frame. The motion gate κ is 9.4877. The feature similarity threshold σ is 0.6 in the videos taken by the moving camera, and 0.7 in the videos taken by the static camera. The max age δ is 100 frames.

C. Additional Experiments and Analyses

C.1. Comparison with the Oracle Tracker

To explore the upper bound of the association method, we compare our method with the ground truth association, called the Oracle tracker. The results on MOT17 *val* set are shown in Table A. There is a gap of 5.7 IDF1 and about 1000 ID Switches between our online GMTracker and the Oracle tracker.

Another observation is that on some metrics, which are extremely relevant to detection results, like MOTA, FP and FN, the gaps between the baseline, our method and the Oracle tracker are relatively small. That is why we mainly

	IDF1	MOTA	MT	ML	FP	FN	ID Sw.
Baseline	68.1	62.1	556	371	1923	124480	1135
Ours	71.5	62.3	555	375	1741	124298	1017
Oracle	77.2	62.6	545	368	1730	124287	14

Table A: Comparison between the baseline, our GMTracker and the Oracle tracker on MOT17 *val* set.

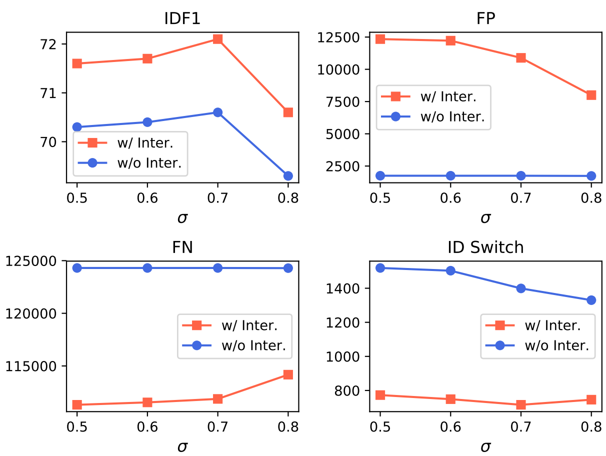

Figure A: Results on IDF1, FP, FN and ID Switch metrics under different threshold σ of the feature similarity to create a new tracklet.

concern with the metrics reflecting the association results, such as IDF1 and ID Switch.

C.2. Discussions

Tracklet born strategies. In our GMTracker, the tracklet born strategies mostly follow DeepSORT, but we also make some improvements to make these strategies more suitable for our approach, as described in Section 4.4 in our main paper. Among the three criteria to create a new tracklet, we find that the threshold σ is the most sensitive hyperparameter in our method. We conduct experiments with different σ, and its influence on IDF1, FP, FN and ID Switch is shown in Fig. A.

C.3. Detailed Performance

As shown in Table B, the results on more metrics, such as HOTA, AssA, DetA, LocA, MT, ML are provided for better comparison.

Algorithm A: GMTracker Algorithm

Input: $\mathcal{D}^t, \mathcal{T}^t$
Output: $\mathcal{T}^{t+1}$
for $D_i^t \in \mathcal{D}^t$ **do**
  $\mathbf{a}_D^{i,t} \leftarrow \texttt{MLP}_\texttt{a}(\texttt{ReID}(\mathbf{I}_i^t))$
  $\mathbf{h}_i^{(0)} \leftarrow \mathbf{a}_D^{i,t}$
for $T_j^t \in \mathcal{T}^t$ **do**
  for $D_{(j)}^k \in T_j^t$ **do**
    $\mathbf{a}_D^{(j),k} \leftarrow \texttt{MLP}_\texttt{a}(\texttt{ReID}(\mathbf{I}_{(j)}^k))$
  $\mathbf{a}_T^{j,t} \leftarrow \texttt{mean}(\mathbf{a}_D^{(j),k})$
  $\mathbf{h}_j^{(0)} \leftarrow \mathbf{a}_T^{j,t}$
for $l \leq l_{max}$ **do**
  for $D_i^t \in \mathcal{D}^t$ **do**
    $\mathbf{m}_i^{(l)} \leftarrow \mathcal{A}(\{w_{i,j}^{(l)}\mathbf{h}_j^{(l)} \mid j \in \mathcal{G}_T\})$
    $\mathbf{h}_i^{(l+1)} \leftarrow \mathcal{F}(\mathbf{h}_i^{(l)}, \mathbf{m}_i^{(l)})$
  for $T_j^t \in \mathcal{T}^t$ **do**
    $\mathbf{m}_j^{(l)} \leftarrow \mathcal{A}(\{w_{i,j}^{(l)}\mathbf{h}_i^{(l)} \mid i \in \mathcal{G}_D\})$
    $\mathbf{h}_j^{(l+1)} \leftarrow \mathcal{F}(\mathbf{h}_j^{(l)}, \mathbf{m}_j^{(l)})$
for $D_i^t, D_{i'}^t \in \mathcal{D}^t$ **do**
  $\mathbf{h}_i \leftarrow \mathbf{h}_i^{(l+1)}, \mathbf{h}_{i'} \leftarrow \mathbf{h}_{i'}^{(l+1)}$
  $\mathbf{h}_{i,i'} \leftarrow l_2([\mathbf{h}_i, \mathbf{h}_{i'}])$
for $T_i^t, T_{i'}^t \in \mathcal{T}^t$ **do**
  $\mathbf{h}_j \leftarrow \mathbf{h}_j^{(l+1)}, \mathbf{h}_{j'} \leftarrow \mathbf{h}_{j'}^{(l+1)}$
  $\mathbf{h}_{j,j'} \leftarrow l_2([\mathbf{h}_j, \mathbf{h}_{j'}])$
for $D_i^t, T_j^t \in \mathcal{D}^t, \mathcal{T}^t$ **do**
  $\mathbf{M}_e^{u,v} \leftarrow \mathbf{h}_{i,i'}^\top \mathbf{h}_{j,j'}$
  $\mathbf{B}_{i,j} \leftarrow \mathbf{h}_i^\top \mathbf{h}_j$
$\texttt{match} \leftarrow \texttt{graph\_matching}(\mathbf{M}_e, \mathbf{B})$
for $D_i^t, T_j^t \in \mathcal{D}^t, \mathcal{T}^t$ **do**
  if $\texttt{IoU}(D_i^t, T_j^t) \leq 0$ or $\texttt{d}(D_i^t, T_j^t) > \kappa$ or $\texttt{cos}(D_i^t, T_j^t) < \sigma$ **then**
    $\texttt{delete}(\texttt{match}(i,j))$
for $D_i^t, T_j^t \in \mathcal{D}_\texttt{unmatch}^t, \mathcal{T}_\texttt{unmatch}^t$ **do**
  if $\texttt{IoU}(D_i^t, T_j^t) \geq 0.3$ **then**
    $\texttt{match}_\texttt{add} \leftarrow \texttt{Hungarian}(\texttt{IoU}(D_i^t, T_j^t))$
for $D_i^t, T_j^t \in \mathcal{D}^t, \mathcal{T}^t$ **do**
  if $\texttt{match}(i,j)$ or $\texttt{match}_\texttt{add}(i,j)$ **then**
    $T_j^{t+1} \leftarrow T_j^t + \{D_i^t\}$
    $\texttt{motion}(T_j^{t+1}).\texttt{update}()$
  if $D_i^t \in \mathcal{D}_\texttt{unmatch}^t$ **then**
    $T_\texttt{new}^{t+1} \leftarrow \{D_i^t\}$
  if $T_j^t.\texttt{last\_update} > \delta$ **then**
    $\texttt{delete}(T_j^t)$
return $\mathcal{T}^{t+1}$

Methods	Refined Det	IDF1 ↑	HOTA↑	MOTA ↑	MT↑	ML↓	FP ↓	FN ↓	IDS ↓	AssA↑	DetA↑	LocA↑
					MOT17							
GNMOT (O*) [7]	-	47.0	-	50.2	19.3	32.7	29316	246200	5273	-	-	-
FAMNet (O) [4]	-	48.7	-	52.0	19.1	33.4	14138	253616	3072	-	-	-
JBNOT (O*) [5]	-	50.8	41.3	52.6	19.7	35.8	31572	232659	3050	39.8	43.3	80.2
Tracktor++ (O) [2]	Tracktor	52.3	42.1	53.5	19.5	36.6	12201	248047	2072	41.7	42.9	80.9
Tracktor++v2 (O) [2]	Tracktor	55.1	44.8	56.3	21.1	35.3	8866	235449	1987	45.1	44.9	81.8
GNNMatch (O) [9]	Tracktor	56.1	45.4	57.0	23.3	34.6	12283	228242	1957	45.2	45.9	81.5
GSM_Tracktor (O) [8]	Tracktor	57.8	45.7	56.4	22.2	34.5	14379	230174	**1485**	47.0	44.9	80.9
CTTrackPub (O) [12]	CenterTrack	59.6	48.2	61.5	**26.4**	**31.9**	14076	200672	2583	47.8	49.0	81.7
GMTracker(Ours) (O)	Tracktor	63.8	49.1	56.2	21.0	35.5	**8719**	236541	1778	53.9	44.9	81.8
GMT_CT(Ours) (O)	CenterTrack	**66.9**	**52.0**	**61.5**	26.3	32.1	14059	**200655**	2415	**55.1**	**49.4**	**81.8**
TPM [10]	-	52.6	41.5	54.2	22.8	37.5	13739	242730	1824	40.9	42.5	80.0
eTC17 [11]	-	58.1	44.9	51.9	23.1	35.5	36164	232783	2288	47.0	43.3	79.4
MPNTrack [3]	Tracktor	61.7	49.0	58.8	28.8	33.5	17413	213594	1185	51.1	47.3	81.5
Lif_TsimInt [6]	Tracktor	65.2	50.7	58.2	28.6	33.6	16850	217944	**1022**	54.9	47.1	81.5
LifT [6]	Tracktor	65.6	51.3	60.5	27.0	33.6	14966	206619	1189	54.7	48.3	81.3
GMT_simInt (Ours)	Tracktor	65.9	51.1	59.0	29.0	33.6	20395	209553	1105	55.1	47.6	81.2
GMT_VIVE (Ours)	Tracktor	65.9	51.2	60.2	26.5	33.2	**13142**	209812	1675	55.1	47.8	81.3
GMTCT_simInt (Ours)	CenterTrack	**68.7**	**54.0**	**65.0**	**29.4**	**31.6**	18213	**177058**	2200	**56.4**	**52.0**	**81.5**
					MOT16							
Tracktor++v2 (O) [2]	Tracktor	54.9	44.6	56.2	20.7	35.8	2394	76844	617	44.6	44.8	82.0
GNNMatch (O) [9]	Tracktor	55.9	44.6	56.9	22.3	35.3	3235	74784	564	43.7	45.8	81.7
GSM_Tracktor (O)[8]	Tracktor	58.2	45.9	57.0	22.0	34.5	4332	73573	**475**	46.7	45.4	81.1
GMTracker(Ours) (O)	Tracktor	63.9	48.9	55.9	20.3	36.6	**2371**	77545	531	53.7	44.6	**82.1**
GMT_CT (Ours) (O)	CenterTrack	**68.6**	**53.1**	**62.6**	**26.7**	**31.0**	5104	**62377**	787	**56.3**	**50.4**	81.8
TPM [10]	-	47.9	36.7	51.3	18.7	40.8	**2701**	85504	569	34.6	39.3	79.1
eTC [11]	-	56.1	42.0	49.2	17.3	40.3	8400	83702	606	44.5	39.9	78.8
MPNTrack [3]	Tracktor	61.7	48.9	58.6	27.3	34.0	4949	70252	354	51.1	47.1	81.7
Lif_TsimInt [6]	Tracktor	64.1	49.6	57.5	25.4	34.7	4249	72868	**335**	53.3	46.5	**81.9**
LifT [6]	Tracktor	64.7	50.8	61.3	27.0	34.0	4844	65401	389	53.1	48.9	81.4
GMT_simInt (Ours)	Tracktor	66.2	51.2	59.1	27.5	34.4	6021	68226	341	55.1	47.7	81.5
GMT_VIVE (Ours)	Tracktor	66.6	51.6	61.1	26.7	33.3	3891	66550	503	55.3	48.5	81.5
GMTCT_simInt (Ours)	CenterTrack	**70.6**	**55.2**	**66.2**	**29.6**	**30.4**	6355	**54560**	701	**57.8**	**53.1**	81.5

Table B: Detailed comparison with state-of-the-art methods on MOT16 and MOT17 *test* set. (O) denotes online methods. (O*) denotes near-online methods.

References

[1] Shane Barratt. On the differentiability of the solution to convex optimization problems. *arXiv preprint arXiv:1804.05098*, 2018. 1

[2] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixe. Tracking without bells and whistles. In *ICCV*, pages 941–951, 2019. 3

[3] Guillem Brasó and Laura Leal-Taixé. Learning a neural solver for multiple object tracking. In *CVPR*, pages 6247–6257, 2020. 3

[4] Peng Chu and Haibin Ling. FAMNet: Joint learning of feature, affinity and multi-dimensional assignment for online multiple object tracking. In *ICCV*, pages 6172–6181, 2019. 3

[5] Roberto Henschel, Yunzhe Zou, and Bodo Rosenhahn. Multiple people tracking using body and joint detections. In *CVPR Workshops*, pages 770–779, 2019. 3

[6] Andrea Hornakova, Roberto Henschel, Bodo Rosenhahn, and Paul Swoboda. Lifted disjoint paths with application in multiple object tracking. In *ICML*, 2020. 3

[7] Jiahe Li, Xu Gao, and Tingting Jiang. Graph networks for multiple object tracking. In *WACV*, pages 719–728, 2020. 3

[8] Qiankun Liu, Qi Chu, Bin Liu, and Nenghai Yu. GSM: Graph similarity model for multi-object tracking. In *IJCAI*, pages 530–536, 2020. 3

[9] Ioannis Papakis, Abhijit Sarkar, and Anuj Karpatne. GCNNMatch: Graph convolutional neural networks for multi-object tracking via sinkhorn normalization. *arXiv preprint arXiv:2010.00067*, 2020. 3

[10] Jinlong Peng, Tao Wang, Weiyao Lin, Jian Wang, John See, Shilei Wen, and Erui Ding. TPM: Multiple object tracking with tracklet-plane matching. *Pattern Recognition*, page 107480, 2020. 3

[11] Gaoang Wang, Yizhou Wang, Haotian Zhang, Renshu Gu, and Jenq-Neng Hwang. Exploit the connectivity: Multi-object tracking with trackletnet. In *ACM MM*, pages 482–490, 2019. 3

[12] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Tracking objects as points. In *ECCV*, pages 474–490, 2020. 3