Robust Local Effective Matching Model for Multi-Target Tracking

Hao Sheng, Li Hao, Jiahui Chen, and Yang Zhang

State Key Laboratory of Software Development Environment, School of Computer Science and Engineering, Beihang University, Beijing, China,
{shenghao,haoli}@buaa.edu.cn

Abstract. Occlusion is one of the main challenges in multi-target tracking. Which causes fragments in tracking. In order to handle with fragments, various motion models were proposed. However, motion model has limited effect on dealing with long-term fragments, because the predictability of target motion declines with increase in fragment length. Thus we proposed a robust local effective matching model for partial detections to reduce fragment length first. The proposed model is integrated into a network flow based hierarchical framework to solve long-term fragments step-by-step. Initial tracklets are generated for later analysis in the first level. The robust local effective matching model is used in the second level to reduce fragment length. A motion model is utilized in the third level to solve fragments between tracklets. The benchmark results on 2D MOT 2015 dataset were compared with several state-of-the-art trackers and our method got competitive results with those trackers.

Keywords: Multi-target tracking, Network flow, Long-term fragment, Partial detection, Local effective matching model

1 Introduction

Multi-target tracking [12] is the basis of action recognition, behavior analysis. It is used in numerous applications such as visual surveillance and medical image processing. In recent years, great progress is made in multi-target tracking. However, it is still a challenging task due to false detections and occlusions.

Occlusion is one of the main challenges in multi-target tracking. It is the reason of fragments. Various motion models were proposed to solve fragments in tracking. However, motion model has limited effect on dealing with long-term fragments, because the predictability of target motion declines with increase in fragment length. We add links between tracklets and partial detections to reduce fragment length at first. Partial detections are detections smaller than the actual sizes of targets. In previous approaches, partial detections are either ignored or used without refinement. Thus we proposed a robust local effective matching model that consists of an affinity measure and a refine method for partial detections. The proposed model is integrated into a hierarchical framework for tracking. Initial tracklets are generated in the first level of the framework.

The robust local effective matching model is used in the second level to reduce fragment length. Finally, fragments are solved by a motion model in the third level. Multi-target tracking is formulated as a minimum-cost flow problem and is solved by linear programming in the proposed framework.

The main contributions of this paper are the followings :

•A robust local effective matching model is proposed to reduce fragment length. Partial detections are refined using the model.

•A hierarchical framework is proposed for tracking. Detection-detection analysis, tracklet-detection analysis and tracklet-tracklet analysis are integrated in the framework to solve fragments in multi-target tracking.

2 Related Work

The goal of multi-target tracking is to extract trajectories of interested targets from a video sequence. According to the way of data processing, multi-target tracking is categorized into online tracking and offline tracking. Online tracking [7, 18, 15] estimates current object state based on past frames. Online tracking is very fast. But it has the drawback that the solution may be trapped in local optimal value. Offline tracking utilizes a batch way to process data. All frames are used to get global optimal solution. Offline tracking [3, 9] is slower but more robust to errors than online tracking. The following discussion is about offline tracking methods.

With the remarkable advance in image-based object detection [5, 16], the task is converted to a data association optimization problem. Tracking-by-detection paradigm is proposed by Breitenstein et al. [2]. First, detection responses, which mean potential positions of interested targets, are generated by an object detector. Then, detections that belong to the same target are assigned with the same ID label. Multiple hypothesis tracking [9] builds a tree of potential track hypothesis for each target. The most likely combination of tracks is selected. However, it is time-consuming and memory intensive. Zhang et al. [19] proposed a network flow based optimization method for multiple target tracking. The minimum-cost flow to the network corresponded to the solution of tracking problem.

Object motion model is important for multi-target tracking because it predicted the potential positions of targets. The motion model assumes objects moved with constant velocity. Milan et al. used a constant motion model in [14]. McLaughlin et al. [13] incorporated a motion model into a minimum-cost network flow tracker. Their method computed the distances of estimated positions and actual positions between two tracklets, and achieved good performance on public surveillance sequences of Oxford town center [1] and PETS S2.L1 [8].

Local optical flow based affinity measure had been previously explored by Choi et al. [3]. Their aggragated optical flow descriptor encoded how interest points in a detection box moved with respect to another detection box. Their work was focused on using interest points trajectories to measure detection similarity. We used tracking state of interest points to refine partial detections and add links between refined partial detections and tracklets.

3 Robust Local Effective Matching Model

The robust local effective matching model has two components, an affinity measure and a method to refine the partial detections. Partial detection covers only a local part of the target. So the affinity measure between partial detections and tracklets should be able to associate the local part with full target. The proposed model uses the PTP(percent of tracked points) score as affinity measure and refined partial detections using tracking state of interest points.

3.1 PTP Score

If detection O_i and detection O_j belong to the same target, interest points in O_i could also be tracked in O_j. Let $\mathcal{P}^{ij} = \{p_k^{ij} : k = 1, 2, ...m\}$ be the interest point set selected from O_i and tracked in O_j. m is the number of selected points. Each interest point is $p_k^{ij} = \left(x_k^i, x_k^j, y_k^i, y_k^j, f_k^{ij}\right)$, where x_k^i and y_k^j are coordinates of p_k^{ij} in O_i, x_k^j and y_k^j are coordinates of p_k^{ij} in O_j, f_k^{ij} is a binary indicator that shows whether the point is tracked.

$$f_k^{ij} = \begin{cases} 1 & \text{if } p_k^{ij} \text{ is tracked} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

The PTP score between detection O_i and detection O_j takes the max value calculated in two directions, from O_i to O_j and from O_j to O_i. The definition of PTP score is as follows:

$$\mathcal{S}_{O_i,O_j} = \max\{\frac{\sum_{p_k^{ij} \in \mathcal{P}_{ij}} f_k^{ij}}{\sum_{p_k^{ij} \in \mathcal{P}_{ij}} 1}, \frac{\sum_{p_k^{ji} \in \mathcal{P}_{ji}} f_k^{ji}}{\sum_{p_k^{ji} \in \mathcal{P}_{ji}} 1}\} \tag{2}$$

In this paper, interest points are selected and tracked using KLT [11]. Local optical flow is the optical flow inside detection bounding boxes.

3.2 Refinement Algorithm for Partial Detection

Partial detection is FP in evaluation, because the overlap between partial detection and ground truth is too small. So partial detections need to be refined.

If $\mathcal{S}_{O_i,O_j} \geq \mathcal{S}_{min}$, $\{O_i, O_j\}$ is known as a matched pair. $\mathcal{S}_{O_i,O_j}$ is the PTP score of O_i and O_j as described in Sec.3.1. $\mathcal{S}_{min}$ is a minimum threshold. For a matched pair $\{O_i, O_j\}$, O_i is a partial detection if $h_i < \alpha * h_j$. h_i is the height of O_i, h_j is the height of O_j and α is a factor that controls acceptable height difference. Here are two assumptions. One is that the detection size of same target will not change much in neighboring frames. The other is at least one detection of the matched pair has an appropriate size. The first assumption is satisfied with adequate frame rate. The second assumption is satisfied by setting constraints.

Algorithm 1 Refinement Algorithm for Partial Detection

Input: $O_i = \{t_i, x_i, y_i, w_i, h_i\}, O_j = \{t_j, x_j, y_j, w_j, h_j\}, \mathcal{P}^{ij} = \{p_k^{ij} : k = 1, 2, ...m\}$, O_i is a partial detection.
Output: O_i'

for all $p_k^{ij} = \left(x_k^i, x_k^j, y_k^i, y_k^j, f_k^{ij}\right) \in \mathcal{P}_{ij}$ **do**
  if f_k^{ij} is 1 **then**
    $\Delta x_k \leftarrow x_k^j - x_k^i$, $\Delta y_k \leftarrow y_k^j - y_k^i$
  else
    $\Delta x_k \leftarrow 0$, $\Delta y_k \leftarrow 0$
  end if
end for
$\Delta x \leftarrow \frac{\sum_{k=1,2,\ldots,m} \Delta x_k}{\sum_{k=1,2,\ldots,m} f_k^{ij}}$, $\Delta y \leftarrow \frac{\sum_{k=1,2,\ldots,m} \Delta y_k}{\sum_{k=1,2,\ldots,m} f_k^{ij}}$
return $O_i' = \{t_i, x_j - \Delta x, y_j - \Delta y, w_j, h_j\}$

As is shown in Alg.1. x_i and y_i are the coordinates of upper left corner of O_i, t_i is the frame of O_i, w_i is the width of O_i and h_i is the height of O_i. The notations used in O_j have similar meanings. Other notations are described in Sec.3.1. Suppose O_i is a partial detection in matched pair $\{O_i, O_j\}$, we need to get a proper size and position to refine it. Since the detections of same target will not change much in neighboring frames, we use the size of O_j as the refined size. The refined position is calculated using the average displacement of interest points and position of O_j.

4 Hierarchical Tracking

Our tracking method works in a hierarchical way. The first level analyzes detection-detection relationship and generates tracklets for later analysis. The robust local effective matching model is used in second level to reduce fragment length. In third level, a motion model is utilized to solve fragments between tracklets.

4.1 Tracklets Generation

Since tracklets generated in this stage are the basis for later analysis, we chose a conservative strategy to get them. We want tracklets with id switches and false alarms as less as possible. These two kinds of mistakes in this stage might persist in later stages, while false negatives and fragments can be solved later.

Given a detection set $\mathcal{O} = \{O_1, O_2, ...O_n\}$, the pairwise cost between detection O_i and detection O_j is defined as :

$$C(O_i, O_j) = \begin{cases} \mathcal{C}(V_{i,j}, V_{max}^{i,j}) + 1 - cos(f_i, f_j) & (O_i, O_j) \in \mathcal{A} \\ +\infty & \text{otherwise} \end{cases} \tag{3}$$

where $V_{i,j}$ is the velocity between detection O_i and detection O_j, $V_{max}^{i,j}$ is the max velocity between O_i and O_j, $V_{max}^{i,j} = h_a * \beta$, h_a is the average height of

O_i and O_j and β is a factor. f_i is a 256-dimensional feature vector of detection O_i trained by deep learning and f_j is the feature vector of O_j, $cos(f_i, f_j)$ is the cosine distance of f_i and f_j, function $\mathcal{C}$ is defined as:

$$\mathcal{C}(\gamma, \lambda) = 1 - e^{\sqrt{\frac{\gamma}{\lambda}}} \tag{4}$$

$\mathcal{A} = \{(O_m, O_n) : m, n \text{ satisfy that } \Delta T_{m,n} \leq T_{max}, V_{m,n} \leq V_{max}^{m,n}, Overlap(m,n) \geq Overlap_{min}\}$. $\Delta T_{m,n}$ is the absolute time difference between detection O_m and detection O_n, T_{max} is the maximum threshold for time gap between detections. $Overlap(m,n)$ is the bounding box overlap between detection O_m and detection O_n, $Overlap_{min}$ is the minimum threshold for overlap.

We get conservative tracklets for later processing by setting strict threshold in this stage. This reduces the search space in the meanwhile.

4.2 Tracklet-Detection Analysis

In order to reduce fragment length of tracklets, we use the robust local effective matching model to explore evidence of targets in unused detections.

Given tracklets set $\mathcal{T} = \{T_1, T_2, \ldots, T_m\}$ and detections set $\mathcal{O}$, we get an unused detection set U. The pairwise cost between tracklet $T_i = \{O_{i1}, O_{i2}, \ldots, O_{il_i}\}$($l_i$ is the length of T_i) and an unused detection O_j is defined as follows:

$$C(T_i, O_j) = \begin{cases} 1 - \mathcal{S}_{O_{i1},O_j} & 0 < t(O_{i1}) - t(O_j) < T_{max}, \mathcal{S}_{O_{i1},O_j} \geq \mathcal{S}_{min} \\ 1 - \mathcal{S}_{O_{il_i},O_j} & 0 < t(O_j) - t(O_{il_{T_i}}) < T_{max}, S_{O_{il_i},O_j} \geq \mathcal{S}_{min} \\ +\infty & \text{otherwise} \end{cases} \tag{5}$$

where O_{i1} is the first detection of T_i, O_{il_i} is the last detection of T_i, $t(O_{i1})$ is the frame of O_{i1}, $t(O_{il_i})$ is the frame of O_{il_i}, $\mathcal{S}_{O_{i1},O_j}$ is the PTP score of O_{i1} and O_j, $\mathcal{S}_{O_{il_i},O_j}$ is the PTP score of O_{il_i} and O_j, $\mathcal{S}_{min}$ is the minimum threshold.

If first condition in Eq.5 is met and $h_j < \alpha * h_{i1}$ or second condition in Eq.5 is met and $h_j < \alpha * h_{il_i}$, then O_j is a partial detection and is refined using the algorithm in Sec.3.2. h_j is the height of O_j, h_{i1} is the height of O_{i1}, h_{il_i} is the height of O_{il_i} and α is a factor that controls acceptable height difference. The cost is one minus PTP score of O_j and its nearest-frame detection in T_i when the PTP score exceeds a minimum threshold and T_i and O_j are not overlapped in time. Otherwise the cost is infinity.

4.3 Tracklet-Tracklet Analysis

After the detection-detection analysis and tracklet-detection analysis, no more detection information could be explored to solve fragment. Thus a motion model is used to analyze relationship between tracklets.

For a tracklet $T_k = \{O_{k1}, O_{k2}, \ldots, O_{kl_k}\}$, l_k is the length of T_k, we use detections in the first 1s to backwardly estimate its velocity and use detections in the last 1s to forwardly estimate its velocity. Linear regression is performed over

x,y coordinates of detections with time as the predictor variable. The forwardly predicted position of T_k in Δt frames later is written as $T_k^f(t(O_{kl_k})+\Delta t)$, $t(O_{kl_k})$ is the frame of O_{kl_k}. The backwardly predicted position of T_k in Δt frames earlier is written as $T_k^b(t(O_{k1})-\Delta t)$, $t(O_{k1})$ is the frame of O_1.

The pairwise cost between tracklets T_i and T_j (T_i happens before T_j) is defined as:

$$C(T_i,T_j)=\begin{cases}1-e^{\sqrt{\frac{E_{i,j}}{E_{max}}}} & \Delta T_{il_i,j1}\le T_{max}, E_{i,j}\le E_{max}\\ +\infty & \text{otherwise}\end{cases} \tag{6}$$

where $E_{i,j}$ is the energy to link T_i and T_j, $\Delta T_{il_i,j1}$ is the absolute time difference between O_{il_i} and O_{j1}, O_{il_i} is the last detection of T_i, O_{j1} is the first detection of T_i, T_{max} is the maximum threshold for time difference, E_{max} is the maximum energy threshold. $E_{i,j}$ is defined as:

$$\begin{aligned}E_{i,j}=&\frac{1}{F}\sum_{t'=1}^{F}|T_i^f(t(O_{j1})+t')-T_j(t(O_{j1})+t')|+\\ &\frac{1}{F}\sum_{t'=1}^{F}|T_j^b(t(O_{il_k})-t')-T_j(t(O_{il_k})-t')|\end{aligned} \tag{7}$$

where F is the window length used to calculate energy, F equals to the value of FPS(frames per second) in our experiments. The cost function is designed to encourage tracklet pairs with small residual between predicted position and actual position of tracklets.

4.4 Minimum-cost Network Flow Optimization

The MOT problem is formulated as a minimum-cost network flow problem. The objective function for the problem is:

$$\begin{aligned}\mathcal{T}^*&=\underset{\mathcal{T}}{\operatorname{argmin}}\sum_{T_k\in\mathcal{T}}-\log P(T_k)+\sum_i-\log P(O_i|\mathcal{T})\\ &=\underset{\mathcal{T}}{\operatorname{argmin}}\sum_i C_{en,i}f_{en,i}+\sum_{i,j}C_{i,j}f_{i,j}+\sum_i C_{ex,i}f_{ex,i}+\sum_i C_i f_i\\ &\quad s.t.\ f_{en,i}+\sum_j f_{j,i}=f_i=f_{ex,i}+\sum_j f_{i,j},\forall i\end{aligned} \tag{8}$$

where $\mathcal{T}$ is the hypothesis set, T_k is a tracklet hypothesis, O_i is a detection. $C_{en,i}$ is the cost of entry edge between source node and O_i, $C_{ex,i}$ is the cost of exit edge between sink node and O_i, C_i is the cost of detection edge of O_i, $C_{i,j}$ is the cost of transition edge between detection O_i and detection O_j, $f_{en,i}$, $f_{ex,i}$, f_i, $f_{i,j}$ are binary indicators show whether those edges are selected.

The transition edge cost between detection nodes is defined as above three subsections. The cost of detection edge is negative normalized confidence of the

detection. The entry/exit edge cost takes a constant zero. Number of targets is obtained using a Fibonacci search in the possible value interval.

In order to avoid the accumulation of errors, we use tracklet-detection analysis and tracklet-tracklet analysis to add some new edges between detections to the network, instead of building new network of tracklets.

5 Experiments

The MOT challenge benchmark provides a standardized evaluation of multiple target tracking methods. So we conducted our experiments on 2D MOT2015[10] datasets. The selected datasets have a total of 22 sequences. We used public detections for excluding the influence of detection quality. Raw detections were pre-processed using a non-maximum suppression method.

5.1 Implementation Details

The feature vectors used in the first level were trained by deep learning. In our first level to generate tracklets, parameters were set as follows: $\beta = 3$, $T_{max} = 2$ frames, $Overlap_{min} = 0.5$. In our second level, $T_{max} = 4$ frames, $\mathcal{S}_{min} = 0.8$, $\alpha = 0.8$. In our third level, $E_{max} = 300$, $T_{max} = 4s$. In the calculation of PTP score, we observed that if total number of selected points are too small, wrong pairs will be produced. So we set a minimum value min_{ns} of total number of selected points. In our experiments, $min_{ns} = 12$. Matlab built-in solver `linprog` finished the linear programming.

5.2 Evaluation Metrics

MOTA is multiple object tracking accuracy. MOTP is multiple object tracking precision. Each track in ground truth is classified to MT($\geq$ 80%),PT or ML($\leq$ 20%) according to total percent of successfully tracked parts. FP(false positives or false alarms), FN(false negatives), IDs(ID switches) and FM(fragments) are different errors made in tracking progress.

5.3 Framework Verification

Stage I is the result of first level. Stage II is the result with first level and second level. Stage III is the result with all three levels. A histogram of fragment length is shown in Fig.1. As the figure shows, number of long-term fragments (Length$\geq 1s$) decreases while number of short fragments(Length$< 1s$) increases in stage II. This is because the robust local effective matching model reduces fragment length at first. In stage III, there is a sharp decline in number of short fragments, because the fragments are solved using the motion model. Decline in number of long-term fragments is much smaller than decline in number of short fragments, which proves that the longer the fragment is, the worse the motion model works.

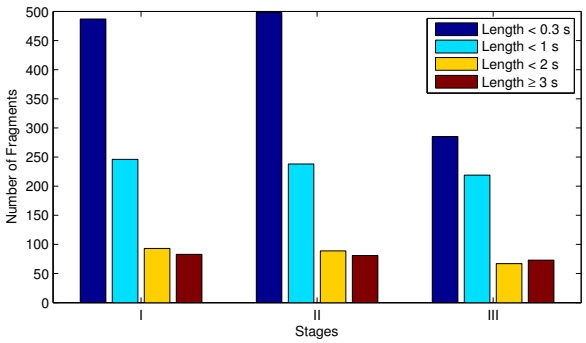

Fig. 1. Fragment statistics on 2D MOT2015 training set.

Table 1. Main evalution metrics on training set.

Stage	Rcll↑	Prec↑	ML↓	FP↓	FN↓	IDs↓	FM↓	MOTA↑	MOTP↑
I	37.7	81.5	278	3407	24863	786	693	27.2	73.9
II	39.4	80.8	266	3735	24163	741	681	28.2	74.0
III	45.8	76.8	246	5513	21632	469	461	30.8	73.8

Tracking result on ETH-Bahnhof is shown in Fig.2. Odd frames are omitted in the figure. The fragment of target exists in frames from 71 to 79 in top row. In bottom row, partial detections(red boxes) in frames 71 to 73 and frames 78 to 89 are refined (green boxes). The fragment length reduced by 4 frames. Then the fragment is solved by a motion model.

The total result on training set is listed in Tab.1. ↑ is a positive indicator meaning the higher the value, the better, while ↓ means the lower the value, the better. FN reduces in stage II, because partial detections are refined and used to reduce the length of fragments. FN reduces in stage III because fragments between tracklets are solved using the motion model. FP increases in stage II and stage III, because some wrong links were added to the network. Increase of MOTA shows that both the robust local effective matching model and the motion model improves the tracker performance.

5.4 Benchmark Results

In order to compare with state-of-the-art trackers, we ran the proposed tracker on 2D MOT 2015 test set. The parameters are trained on training set and listed in Sec.5.1. The benchmark results are listed in Tab.2.

The table shows the comparison of SegTrack[18], JDPA_m[6], LINF1[4], ELP[13], LP_SSVM[17] and our method LFNF. LFNF outperforms other state-of-the-art trackers in both terms of MOTA and MOTP. It is noteworthy that our method achieves the lowest false alarms per frame too. Our approach is most close related to ELP[13], because both work used network flow optimization and motion

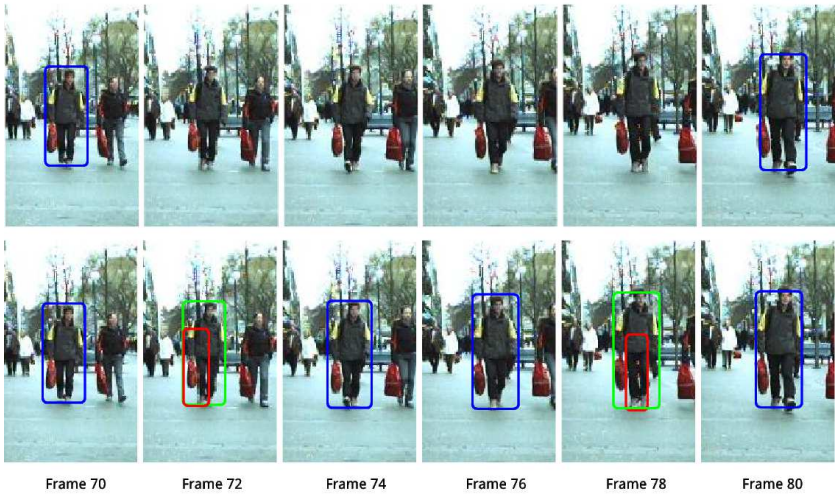

Fig. 2. Tracking results of stage I (top) and stage III (bottom) on ETH-Bahnhof.

Table 2. Comparison on 2D MOT 2015 Benchmark. All trackers use public detections only. FAF: the average number of false alarms per frame. Hz: tracker speed in frames per second. (accessed on 5/4/2017)

Tracker	MOTA	MOTP	FAF	MT	ML	FP	FN	IDs	FM	Hz
SegTrack	22.5	71.7	1.4	5.8%	63.9%	7890	39020	697	**737**	0.2
JDPA_m	23.8	68.2	1.1	5.0%	58.1%	6373	40084	365	869	32.6
LINF1	24.5	71.3	1.0	5.5%	64.6%	5864	40207	**298**	744	7.5
ELP	25.0	71.2	1.3	**7.5%**	**43.8%**	7345	37344	1396	1804	5.7
LP_SSVM	25.2	71.7	1.4	5.8%	53.0%	8369	**36932**	646	849	**41.3**
LFNF(Ours)	**26.1**	**72.5**	**0.8**	5.0%	51.9%	**4487**	39872	1075	1165	4.0

model. Our approach achieves fewer fragments than ELP, which proves that our approach handles fragments more effectively.

6 Conclusion

In order to cope with long-term fragments, we proposed a robust local effective matching model for partial detetctions to reduce fragment length first. The propsoed model is integrated into a hierarchical framework to solve fragments. The first level generates initial tracklets for later analysis. The second level utilizes the robust local effective matching model to reduce fragment length and refine partial detections. The third level solves fragments between tracklets by using a motion model. Experiments were conducted on 2D MOT 2015. Results on training set showed that our method improves tracking performance, especially in terms of Rcll, FN, FM and MOTA. Benchmark results on test set showed that our method get competitive results with other state-of-the-art trackers.

Acknowledgments.

This study is partially supported by the National Natural Science Foundation of China(No.61472019), the National Science Technology Pillar Program (No.2015BAF14B01), the Programme of Introducing Talents of Discipline to Universities, the Open Fund of the State Key Laboratory of Software Development Environment under grant SKLSDE-2017ZX-09 and HAWKEYE Group.

References

1. Benfold, B., Reid, I.: Stable multi-target tracking in real-time surveillance video. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. pp. 3457–3464. IEEE (2011)
2. Breitenstein, M.D., Reichlin, F., Leibe, B., Koller-Meier, E., Van Gool, L.: Robust tracking-by-detection using a detector confidence particle filter. In: 2009 IEEE 12th International Conference on Computer Vision. pp. 1515–1522. IEEE (2009)
3. Choi, W.: Near-online multi-target tracking with aggregated local flow descriptor. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3029–3037 (2015)
4. Fagot-Bouquet, L., Audigier, R., Dhome, Y., Lerasle, F.: Improving multi-frame data association with sparse representations for robust near-online multi-object tracking. In: European Conference on Computer Vision. pp. 774–790. Springer (2016)
5. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE transactions on pattern analysis and machine intelligence 32(9), 1627–1645 (2010)
6. Hamid Rezatofighi, S., Milan, A., Zhang, Z., Shi, Q., Dick, A., Reid, I.: Joint probabilistic data association revisited. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3047–3055 (2015)
7. Hong Yoon, J., Lee, C.R., Yang, M.H., Yoon, K.J.: Online multi-object tracking via structural constraint event aggregation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1392–1400 (2016)
8. Izadinia, H., Saleemi, I., Li, W., Shah, M.: 2t: Multiple people multiple parts tracker. In: European Conference on Computer Vision. pp. 100–114. Springer (2012)
9. Kim, C., Li, F., Ciptadi, A., Rehg, J.M.: Multiple hypothesis tracking revisited. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4696–4704 (2015)
10. Leal-Taixé, L., Milan, A., Reid, I., Roth, S., Schindler, K.: Motchallenge 2015: Towards a benchmark for multi-target tracking. arXiv preprint arXiv:1504.01942 (2015)
11. Lucas, B.D., Kanade, T., et al.: An iterative image registration technique with an application to stereo vision (1981)
12. Luo, W., Xing, J., Zhang, X., Zhao, X., Kim, T.K.: Multiple object tracking: A literature review. arXiv preprint arXiv:1409.7618 (2014)
13. McLaughlin, N., Del Rincon, J.M., Miller, P.: Enhancing linear programming with motion modeling for multi-target tracking. In: 2015 IEEE Winter Conference on Applications of Computer Vision. pp. 71–77. IEEE (2015)

14. Milan, A., Roth, S., Schindler, K.: Continuous energy minimization for multitarget tracking. IEEE Transactions on Pattern Analysis & Machine Intelligence 36(1), 58–72 (2014)
15. Possegger, H., Mauthner, T., Roth, P.M., Bischof, H.: Occlusion geodesics for online multi-object tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1306–1313 (2014)
16. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–1 (2015)
17. Wang, S., Fowlkes, C.C.: Learning optimal parameters for multi-target tracking with contextual interactions. International Journal of Computer Vision pp. 1–18 (2016)
18. Wen, L., Du, D., Lei, Z., Li, S.Z., Yang, M.H.: Jots: Joint online tracking and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2226–2234 (2015)
19. Zhang, L., Li, Y., Nevatia, R.: Global data association for multi-object tracking using network flows. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. pp. 1–8. IEEE (2008)