Online Multi-Object Tracking based on Hierarchical Association Framework

Jaeyong Ju, Daehun Kim, Bonhwa Ku,
Hanseok Ko
School of Electrical Engineering
Korea University
Anam-dong, Seongbuk-gu, Seoul, Korea
{jyju, dhkim, bhku}@ispl.korea.ac.kr,
hsko@korea.ac.kr

David K. Han
Office of Naval Research
Arlington VA, USA
ctmkhan@gmail.com

Abstract

Online multi-object tracking is one of the crucial tasks in time-critical computer vision applications. In this paper, the problem of online multi-object tracking in complex scenes from a single, static, un-calibrated camera is addressed. In complex scenes, it is still challenging due to frequent and prolonged occlusions, abrupt motion change of objects, unreliable detections, and so on. To handle these difficulties, this paper proposes a four-stage hierarchical association framework based on online tracking-by-detection strategy. For this framework, tracks and detections are divided into several groups depending on several cues obtained from association results with the proposed track confidence. In each association stage, different sets of tracks and detections are associated to handle the following problems simultaneously: track generation, progressive trajectory construction, track drift and fragmentation. The experimental results show the robustness and effectiveness of the proposed method compared with other state-of-the-art methods.

1. Introduction

Tracking of multiple objects is an important topic with many computer vision applications such as video surveillance, sports analysis, and robot navigation. In particular, as the use of video camera grows explosively, it becomes increasingly important to develop the method of robustly tracking multiple objects, especially people, in real-time. However, in complex scenes, it is still challenging due to frequent and prolonged occlusions, abrupt motion change of objects, unreliable detections, and so on. Due to the significant improvements in object detectors [1, 2, 3], a lot of recent works on multi-object tracking have focused on the tracking-by-detection strategy where detections are extracted by an object detector independently in each frame and linked to build object trajectories.

This strategy can be generally categorized into offline (batch) and online (sequential) approaches. The offline approaches build multiple trajectories by globally optimizing detections within the entire video or a large sliding window in an offline step [4, 5, 6, 18]. Huang *et al.* [5] proposed a hierarchical association framework for producing longer tracklets at each level gradually. Pirsiavash *et al.* [6] solved a global data association problem using a min-cost flow algorithm in a network flow. These offline approaches usually show better performance than online approaches, because they exploit the future information. However, it requires huge computation and has significant latency limit between detection and tracking result. Thus these approaches cannot be applied to real-time applications.

On the other hand, the online approaches [7, 8, 9, 10, 11, 12, 13, 19] can be applied to real-time applications because they sequentially build object trajectories based on a frame-by-frame data association. However, these approaches are more vulnerable compared to offline approaches in complex scenes where there are frequent and prolonged occlusions, abrupt change of object motion, unreliable detections, and so on because future information is not used**.** Thus, online approaches are more likely to produce such problems as track drift, fragmentation and ID switches in complex scenes. To tackle these difficulties, Breitenstein *et al.* [9] exploited continuous confidence density by combining detector outputs and online-trained classifier based on a particle filter framework. Yan *et al.* [19] proposed to solve a data association problem hierarchically using the Hungarian algorithm with outputs of both independent trackers and detector. Shu *et al.* [12] adopted the deformable part model to handle partial occlusions, but this method is far difficult to be processed in real-time since the DPM-based detector [2] has several speed bottlenecks. In spite of their efforts, these methods still suffer from track drift problem in more complex scenarios wherein people change their motions abruptly under occlusion because of their simplified motion models that make it difficult to re-assign "drifting track" to re-appearing object from occlusion. In addition, since there is no track linking process, they are prone to cause the track fragmentation problem under long-term occlusions.

Motivated by the tracking problems described by above discussion, we propose a four-stage hierarchical association framework that can handle the following problems simultaneously: track generation, progressive trajectory

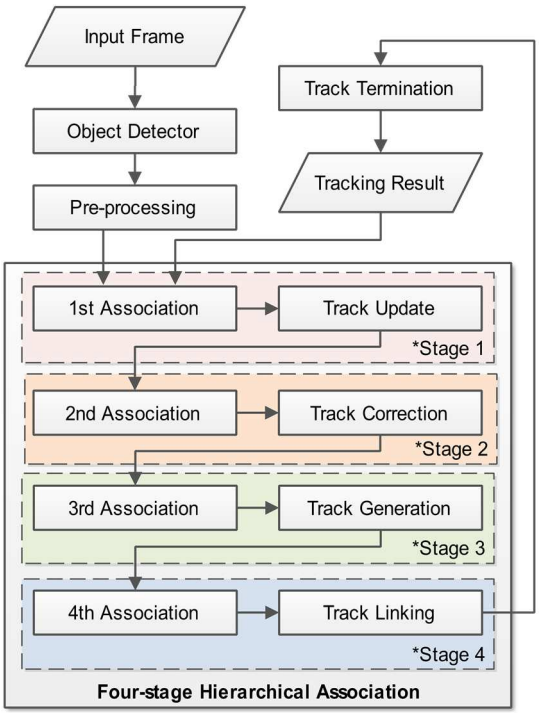


Figure 1. Flowchart of the proposed method

construction, track drift and fragmentation. To this end, first, tracks and detections are divided into several groups depending on several cues obtained from association results with the proposed track confidence. In each association stage, these different sets of tracks and detections are associated for each goal, i.e., resolution of the above problems. By following an online tracking-by-detection strategy, the proposed method can be fully applied to real-time applications while robustly tracking multiple objects in complex scenes.

The rest of the paper is organized as follows. Section 2 describes the proposed method for sequentially and robustly tracking multiple objects. Next, Section 3 presents experimental results. Finally, conclusion is provided in Section 4.

2. Proposed tracking method

2.1. Overview

The flowchart of the proposed tracking method is shown in Fig. 1. In each frame an object detector detects objects of interest. In the pre-processing step, detection responses are pruned using the intensity difference in a bounding box region of each detection. Here, the bounding boxes overlap between tracks and detections are also used to prevent the problem that detections corresponding to standing people. This way, a lot of static false positives (e.g., trees in the background) can be removed effectively. Next, in the actual tracking process, we perform the proposed four-stage hierarchical association which finds optimal assignment of different subject pairs (i.e., track-to-detection or track-to-track) based on different affinities at each stage is performed. Here, if there is no subject pair corresponding to each association stage, we skip those stages and move on the next stage. The assignment problems of all the stages are solved by the greedy algorithm [9]. After the above process, the track termination process is performed by examining tracks which meets the condition (8) in Section 2.2 or exits the field-of-view.

2.2. Hierarchical groups of track and detection

We denote the set of all detections extracted by an object detector at the current frame t by $\mathcal{D}^t = \{d_k^t\}$. There are two subsets $\mathcal{D}_{U^1}^t$ and $\mathcal{D}_{U^2}^t$ in $\mathcal{D}^t$, i.e., $\mathcal{D}_{U^2}^t \subset \mathcal{D}_{U^1}^t \subset \mathcal{D}^t$, where $\mathcal{D}_{U^1}^t$ is the set of un-matched detections from the 1st association stage and $\mathcal{D}_{U^2}^t$ is that of un-matched detections from the 2nd association stage. These detection sets are associated at different stages of the hierarchical association framework, respectively.

Next, the set of all tracks updated at the current frame t is denoted by $\mathcal{R}^t = \{r_i^t\}$. This set is largely divided into three disjoint subsets as follows:

$$\mathcal{R}^t = \mathcal{R}_A^t \cup \mathcal{R}_I^t \cup \mathcal{R}_C^t \tag{1}$$

where $\mathcal{R}_A^t$, $\mathcal{R}_I^t$, and $\mathcal{R}_C^t$ represent the active track set, inactive track set and candidate track set, respectively. The active track set $\mathcal{R}_A^t$ includes tracks corresponding to the currently existing objects. In this set, there are three disjoint subsets as follows:

$$\mathcal{R}_A^t = \mathcal{R}_{A^n}^t \cup \mathcal{R}_{A^r}^t \cup \mathcal{R}_{A^u}^t \tag{2}$$

where $\mathcal{R}_{A^n}^t$, $\mathcal{R}_{A^r}^t$, and $\mathcal{R}_{A^u}^t$ represent the novice track (i.e., recently generated track) set, reliable track set, and un-reliable track set, respectively. Each subset is defined by:

$$\mathcal{R}_{A^n}^t = \{r_i^t | L(r_i^t) \le th_L\} \tag{3}$$
$$\mathcal{R}_{A^r}^t = \{r_i^t | L(r_i^t) > th_L,\ \Omega(r_i^t) \ge th_{\Omega,\mathrm{r}}\} \tag{4}$$
$$\mathcal{R}_{A^u}^t = \{r_i^t | L(r_i^t) > th_L,\ \Omega(r_i^t) < th_{\Omega,\mathrm{r}}\} \tag{5}$$

where th_L ($th_L = 10$ in our experiment) is the threshold on the track length $L(\cdot)$ for distinguishing whether the track is novice or not and $th_{\Omega,\mathrm{r}}$ is the threshold on the proposed track confidence $\Omega(\cdot) \in [0,1]$ described in Selection 2.3 for distinguishing whether the track is reliable or un-reliable (i.e., likely to drift or lost). In the inactive track set $\mathcal{R}_I^t$, there are two disjoint subsets as follows:

$$\mathcal{R}_I^t = \mathcal{R}_{I^o}^t \cup \mathcal{R}_{I^e}^t \tag{6}$$

where $\mathcal{R}_{I^o}^t$ and $\mathcal{R}_{I^e}^t$ represent the lost track set and

terminated track set, respectively. The lost track set $\mathcal{R}_{lo}^{t}$ includes tracks corresponding to temporally lost object due to long-term occlusion and the terminated track set $\mathcal{R}_{le}^{t}$ includes those of totally disappeared objects. Each subset is defined by:

$$\mathcal{R}_{lo}^{t} = \left\{ r_i^t \middle| L(r_i^t) > th_L,\ \Omega(r_i^t) < th_{\Omega,I},\ t - t_e^i < th_e \right\} \quad (7)$$
$$\mathcal{R}_{le}^{t} = \left\{ r_i^t \middle| L(r_i^t) > th_L,\ \Omega(r_i^t) < th_{\Omega,I},\ t - t_e^i \geq th_e \right\} \quad (8)$$

where $th_{\Omega,I}$ is the threshold for distinguishing whether the track is inactive or not, and t_e^i is the end frame of the track in active, i.e., $\Omega(r_i^{t'}) < th_{\Omega,I}$, $\forall t' \in [t_e^i, t]$. The candidate track set $\mathcal{R}_C^t$ includes the tracks waiting for enough matched detections in the 3rd stage before being the new active track. The elements of each track set can be changed based on their association results from each stage of the proposed hierarchical association framework.

2.3. Track Confidence

To evaluate the reliability of tracks, we propose a novel track confidence that takes into account the appearance affinity and the observation (detection) continuity of tracks. The proposed track confidence of track r_i at frame t is defined as the average of the confidence values of appearance and observation during the recent T_r frames:

$$\Omega(r_i^t) = \frac{1}{T_r} \sum_{t'=t-T_r+1}^{t} \Omega_a(r_i^{t'}) \cdot \Omega_o(r_i^{t'}) \quad (9)$$

where $\Omega_a(r_i^{t'})$ and $\Omega_o(r_i^{t'})$ are the confidence terms of appearance and observation for track r_i at frame t', respectively.

The appearance confidence term $\Omega_a(r_i^t) \in [0,1]$ is defined as:

$$\Omega_a(r_i^t) = [1 + \exp\{-\beta(\psi_a(r_i^t) - \tau_a)\}]^{-1} \quad (10)$$

where β is the slope parameter of the sigmoid-based function Ω_a and $\psi_a(\cdot)$ is the appearance score of track that can be obtained from the appearance affinity value of the matched pair or that value between the previous track and the predicted track in the 1st/2nd association stage as discussed with the cut-off threshold τ_a in Section 2.4. Thus, if the track is correctly associated or accurately predicted ($\psi_a \geq \tau_a$), the appearance confidence Ω_a becomes close to 1. Otherwise, ψ_a rapidly decreases due to the characteristic of a sigmoid function.

The observation confidence term $\Omega_o(r_i^t) \in [0,1]$ is defined as:

$$\Omega_o(r_i^t) = \left[1 + \exp(\eta_{r_i}^t - \delta_o)\right]^{-1} \quad (11)$$

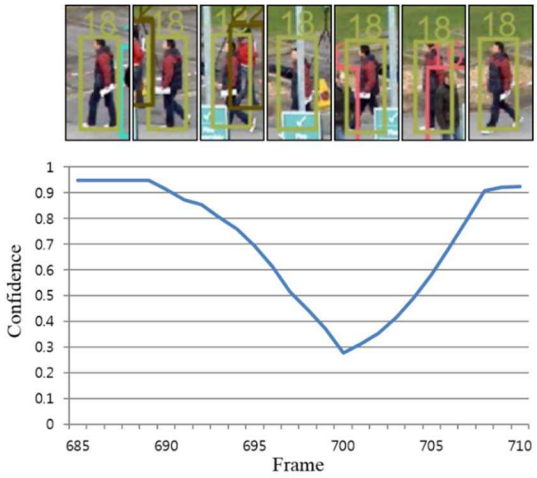


Figure 2. Confidence variation of track ID18 under occlusion (PETS 2009 S2L1).

where $\eta_{r_i}^t$ is the number of consecutively missed observations (detection responses) continued until frame t of track r_i and δ_o is the short-term occlusion tolerance. Thus, the observation confidence decreases rapidly if the detection responses of track r_i are missed continually over δ_d frames.

Consequently, the proposed track confidence has a value in the range of 0–1 and we consider a track with a high confidence value (i.e., $\Omega(r_i^t) \geq th_{\Omega,\mathrm{r}}$ where $th_{\Omega,\mathrm{r}}$ is set to 0.7 in our experiment) as a reliable track; otherwise it is considered as an un-reliable track likely to drift or fragment. Fig. 2 shows the variation of the proposed track confidence under occlusion. In Fig. 2, while track ID18 is occluded by a traffic sign or other track, the confidence of this track decreases due to a low appearance score and an increasing number of consecutively missed detections.

2.4. Hierarchical association framework

First, at the 1st stage, the association between all of the current detections and the active tracks updated at the previous frame is performed to progressively build object trajectory. After this association, track states and confidence values are updated with their association results. Next, at the 2nd stage, the association between un-reliable tracks in the active track set and detections, both of which are not associated from the 1st stage is performed to handle drifting target caused by abrupt object motion change under long-term occlusion. By re-assigning detection responses of re-appearing objects from occlusion to drifting tracks, these tracks are corrected using the associated detection. Then, at the 3rd stage, the association between candidate tracks and remaining un-matched detections after the previous association stages is performed to generate new active tracks. Finally, at the 4th stage, the association between lost tracks in the inactive track set and novice tracks is performed to link fragmented tracks of the same object.

2.4.1 Greedy association scheme. In an online tracking-by-detection framework, a frame-by-frame data association for building object trajectories can be formulated as an assignment problem that matches detections with while meeting the 1-to-1 mapping constraint. In the proposed work, the greedy algorithm [9] is used to solve the assignment problems of each association stage based on each affinity. In particular, in each association stage, the affinity matrix between the subject pairs (i.e., tracks-detections or tracks-tracks) corresponding to each stage is constructed based on the calculated affinity values. Then, the procedure that find the optimal pair with the non-zero maximum affinity value and delete the corresponding row and column from the affinity matrix is repeated until no more pairs are available.

2.4.2 Stage 1: Progressive trajectory construction. The 1st association stage solves the assignment problem between active tracks and detections to progressively build object trajectories. In the current frame t, the input pairs of this association are $\{(r_i^{t-1}, d_j^t)\}$, $\forall r_i^{t-1} \in \mathcal{R}_A^{t-1}$ *and* $\forall d_j^t \in \mathcal{D}^t$ where r_i^{t-1} is the i-th track updated in the previous frame. The affinity of the input pair for the 1st association is defined as the product of two terms based on position-size and appearance:

$$\mathcal{A}_{1st}(r_i^{t-1}, d_j^t) = \mathcal{A}_{ps}(r_i^{t-1}, d_j^t)\mathcal{A}_a(r_i^{t-1}, d_j^t), \quad \forall r_i^{t-1} \in \mathcal{R}_A^{t-1} \text{ and } \forall d_j^t \in \mathcal{D}^t \quad (12)$$

where $\mathcal{A}_{ps}(\cdot,\cdot)$ and $\mathcal{A}_a(\cdot,\cdot)$ are the position-size affinity and the appearance affinity of the input pair, respectively.

The position-size affinity of track r_i and detection d_j is defined by:

$$\mathcal{A}_{ps}(r_i, d_j) = \frac{B(r_i) \cap B(d_j)}{B(r_i) \cup B(d_j)} \quad (13)$$

where $B(\cdot) = (x, y, w, h)$ is the bounding box of a track or a detection and thus $B(r_i) \cap B(d_j)$ and $B(r_i) \cup B(d_j)$ are the intersection and union area of $B(r_i)$ and $B(d_j)$, respectively.

To evaluate the appearance affinity rapidly for real-time application, we adopted the template matching-based approach. The template of detections is constructed by a 24-bin RGI (red-green-intensity) histogram extracted from the image patch within the bounding box of detections. Here, all patches are resized to the size of 30×70 to be invariant to object scale. In addition, to take appearance variation of objects into account, each active track maintains the latest template and the historical template set which consists of at most N_H templates ($N_H = 30$ in our experiments). After the association, the templates of associated detections are updated as the latest templates of each matched track. Then, the latest template of the previous frame is added into the historical template set and the oldest template of that set is deleted to keep the up-to-date appearance model of objects. To evaluate the similarity between two templates, the Bhattacharyya coefficient $\rho(\cdot,\cdot)$ is used. Let ζ_{d_j} is the template of detection d_j, $\zeta_{r_j}^L$ is the latest template of track r_i, and $H_{r_i} = \{\zeta_{r_j}^{H(k)}\}, k \in [1, N_H]$ is the historical template set of track r_i. Finally, the appearance affinity of track r_i and detection d_j is defined by:

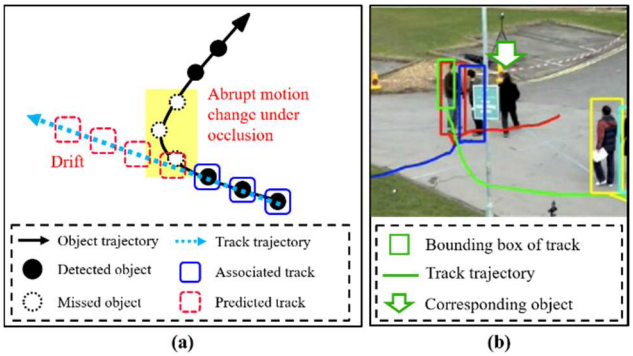


Figure 3. Track drift problem due to abrupt motion change of object under occlusion: (a) snap shot of track drift situation taking place over a window of consecutive frames, (b) tracking result including the drifting track (green box and line) of Breitenstein *et al*. [9]

$$\mathcal{A}_a(r_i, d_j) = \begin{cases} \psi_a(r_i, d_j), & \text{if } \psi_a(r_i, d_j) \geq \tau_a \\ 0, & \text{otherwise} \end{cases} \quad (14)$$

$$\psi_a(r_i, d_j) = \omega_L \cdot \rho\left(\zeta_{d_j}, \zeta_{r_j}^L\right) + (1 - \omega_L) \max_{\zeta_{r_j}^{H(k)} \in \mathcal{H}_{r_i}} \rho\left(\zeta_{d_j}, \zeta_{r_j}^{H(k)}\right) \quad (15)$$

where τ_a is the cut-off threshold to prevent incorrect associations by considering only reliable pairs and $\omega_L \in [0,1]$ is the weight for the latest appearance of the objects.

After the 1st association, the states (position, size, velocity) and the confidence values of active tracks are updated with their association results. For track state update, Kalman filter based on a constant velocity motion model is used where the states of un-associated tracks are only predicted. This filtering is effective in simple situation assuming people are not likely to change their motion abruptly. Here, the set of un-matched active tracks is denoted by $R_{U^1}^t$.

2.4.3 Stage 2: Handling of drifting tracks. In complex situation where objects are occluded for a long term while changing their motion abruptly, the conventional online tracking methods based on simplified motion models (e.g., constant velocity model) [8, 9, 13] are prone to produce drift problem as shown in Fig. 3. If track drift persists in

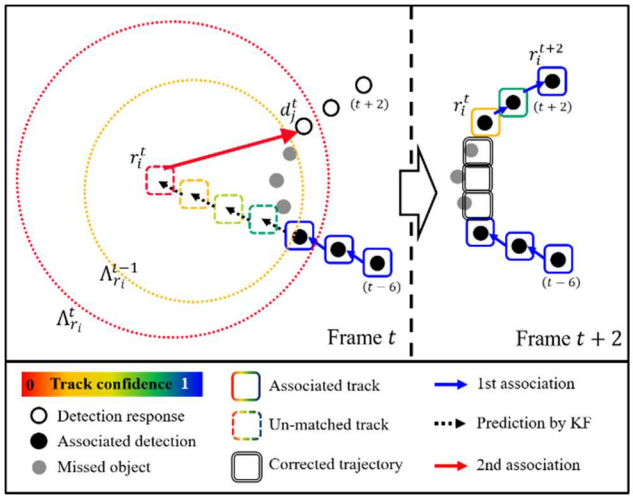


Figure 4. Snap shot of the 1st and 2nd association stage place over a window of consecutive frames (*t-6, ..., t+2*) where the color of rectangle contours indicates the track confidence.

this situation, it is more difficult to re-assign drifting tracks to detections of re-appearing objects from occlusion. In the proposed framework, as the confidence values of drifting tracks decreases continuously, these tracks are converted from reliable tracks into un-reliable tracks. To solve this drift problem, the 2nd association stage solves the re-assignment problem between un-reliable tracks and detections both of which are not associated from the 1st stage. In the current frame t, the input pairs of this association are $\{(r_i^t, d_j^t)\}$, $\forall r_i^t \in \{\mathcal{R}_{A^u}^t \cap R_{U^1}^t\}$ *and* $\forall d_j^t \in \mathcal{D}_{U1}^t$. The affinity of the 2nd association considers only an appearance term of the pair within a validation range without a position-size term because of un-reliable motion dynamics of un-reliable tracks. Thus, the affinity of the input pair for the 2nd association is defined as:

$$\mathcal{A}_{2nd}(r_i^t, d_j^t) = \begin{cases} \mathcal{A}_a(r_i^t, d_j^t), & \text{if } dist(r_i^t, d_j^t) \geq \Lambda_{r_i}^t, \\ 0, & \text{otherwise} \end{cases}$$
$$\forall r_i^t \in \{\mathcal{R}_{A^u}^t \cap R_{U^1}^t\} \text{ and } \forall d_j^t \in D_{U^1}^t \quad (16)$$

where $dist(r_i^t, d_j^t)$ is the distance between track r_i^t and detection d_j^t and $\Lambda_{r_i}^t = \alpha \cdot w_{r_i}^t \cdot (1 - \Omega(r_i^t))$ is the valid association range of un-reliable track r_i^t. Here, the smaller the track width denoted by $w_{r_i}^t$ is, the larger the valid association range is because the movement scale of objects get bigger when they get closer to camera. Also, as the track confidence decreases, the valid association range increases because the distance between a drifting track and the corresponding object can grow larger if the track drift persists. Thus, the 2nd association allow us to re-assign drifting tracks to detections of re-appearing objects which is even far away from the corresponding tracks as shown in Fig. 4.

After the 2nd association, the states and confidence values of associated tracks are re-updated with the associated detections. The trajectory of drift interval is corrected by linearly interpolating the latest missing interval.

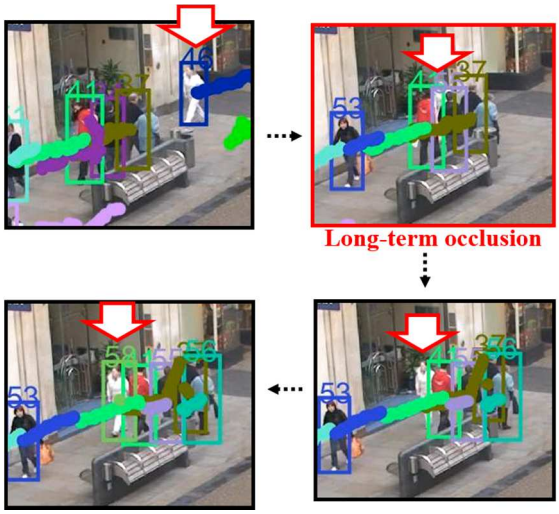


Figure 5. Track fragmentation problem under long-term occlusion.

2.4.4 Stage 3: Active track generation. The 3rd association stage solves the assignment problem between candidate tracks and remaining un-matched detections from the previous stages to generate new active tracks. Thus, the input pairs of this association in the current frame t are $\{(r_i^{t-1}, d_j^t)\}$, $\forall r_i^{t-1} \in \mathcal{R}_C^{t-1}$ *and* $\forall d_j^t \in D_{U^2}^t$. The affinity of the 3rd association is equal to that of 1st. When the candidate track is associated in certain consecutive frames (5 frames in our experiments), it is converted into a new novice track.

2.4.5 Stage 4: Handling of fragmented tracks. In challenging situation where objects are constantly occluded by other objects or obstacles for a long-term, the track fragmentation problem likely to be occur. In the proposed framework, these tracks may be broken into two tracks, i.e., lost track (ID 46) and novice track (ID 58), under long-term occlusions as shown in Fig. 5. To link these fragmented tracks thereby building longer trajectories, the 4th association stage solves the assignment problem between lost tracks and novice tracks. Thus, the input pairs of this association in the current frame t are $\{(r_i^t, r_j^t)\}$, $\forall r_i^t \in \mathcal{R}_{I^o}^t$ *and* $\forall r_j^t \in \mathcal{R}_{A^n}^t$. The affinity of the 4th association is defined as the product of three terms based on motion as well as position-size and appearance:

$$\mathcal{A}_{5th}(r_i^t, r_j^t) = \mathcal{A}_{ps}(r_i^t, r_j^t)\mathcal{A}_a(r_i^t, r_j^t)\mathcal{A}_m(r_i^t, r_j^t),$$
$$\forall r_i^t \in \mathcal{R}_{I^o}^t \text{ and } \forall r_j^t \in \mathcal{R}_{A^n}^t \quad (17)$$

where $\mathcal{A}_m$ is the motion affinity term between tracks. The motion affinity of tracks r_i^t and r_j^t is defined as:

$$\mathcal{A}_m(r_i^t, r_j^t) = \frac{1}{2}\left(\frac{\bar{v}_{r_i}^t \cdot \bar{v}_{r_j}^t}{\|\bar{v}_{r_i}^t\|\|\bar{v}_{r_i}^t\|} + 1\right) \times \left(1 - \frac{\|\|\bar{v}_{r_i}^t\| - \|\bar{v}_{r_i}^t\|\|}{\|\bar{v}_{r_i}^t\| + \|\bar{v}_{r_i}^t\|}\right) \quad (18)$$

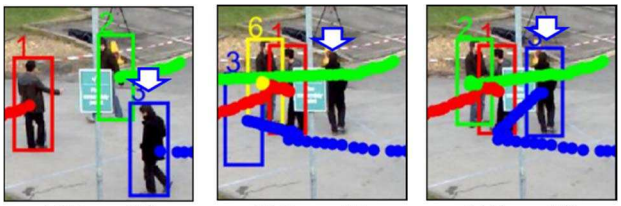
where $\bar{v}_{r_i}^t$ is the average velocity of track r_i at frame t. The first and second term measure the similarity of the motion direction and magnitude between tracks, respectively. When calculating the position-size affinity, the current state of lost track $r_i^t \in \mathcal{R}_{lo}^t$ is estimated by using the position and average velocity at the end frame t_e^i. If lost track r_i^t and novice track r_j^t are associated in this stage, these two tracks are considered as the same object and linked by linearly interpolating the trajectory in the lost interval of this object.

3. Experimental results

3.1. Dataset and Detections

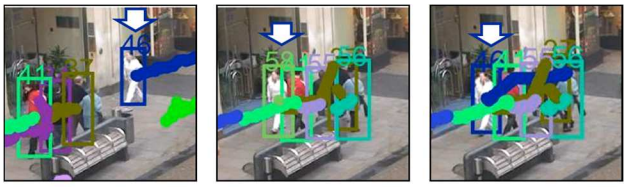
The performance of the proposed method is evaluated on the publicly available video sequences: PETS 2009 [14], Town-Centre [15], and PETS 2016 Challenge [20]. For the performance evaluation of the PETS 2009 dataset, we use the two sequences recorded from view-1 with the resolution of 768×576 pixel: S2L1 (795 frames) and S2L2 (436 frames). In these sequences, numerous pedestrians occlude each other or they are occluded by an obstacle (traffic sign) located in the middle of the intersection. In particular, some pedestrians change their motion abruptly in the occlusion area by an obstacle, thereby making tracking more difficult. Additionally, the S2L2 sequence shows a denser crowd. The Town-Centre dataset is composed of total 4500 frames with the resolution of 1980×1080 pixel. This sequence is semi-crowded with some long-term occlusions by pedestrians occluding each other or obstacles such as benches. For the performance evaluation of the PETS 2016 challenge, we use the ARENA dataset which is comprised of 7 sequences recorded from one environmental camera (ENV_RGB_3, 768×576 pixel) which provide a global view equal to the PETS 2009 sequences and two on-board cameras (TRK_RGB_1, TRK_RGB_2, 1280×960 pixel) mounted at each corner of a truck. In these sequences, there are various challenges such as scale change, occlusion, pose change, and clutter.

To detect pedestrians in the PETS 2009 and Town-Centre datasets as well as the ENV sequences of PETS 2016, we used the ACF detector [3]. On the other hand, for the TRK sequences of PETS 2016, we used the Faster R-CNN detector [1] because the ACF detector fail to respond to objects with large scale and particular poses such as falls in these sequences.



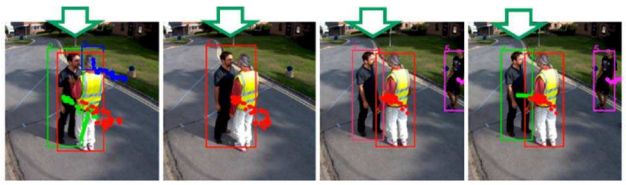
(a) Frame #18 (b) Frame #42 (c) Frame #42

Figure 6. Tracking results of the proposed method in complex situation with abrupt motion change and occlusion (PETS'09 S2L1): (a) Before this situation, (b) After stage 1, (c) After stage 2.



(a) Frame #638 (b) Frame #683 (c) Frame #683

Figure 7. Tracking results of the proposed method under long-term occlusion situation (Town-Centre): (a) Before this situation, (b) After stage 3, (c) After stage 4.



(a) Frame #437 (b) Frame #452 (c) Frame #480 (d) Frame #480

Figure 8. Tracking results of the proposed method in severe inter-object occlusion situation (PETS'16 A1_ARENA-15_06_TRK_RGB_2): (a) Before this situation, (b) Track loss (c) After stage 3, (d) After stage 4.

3.2. Evaluation metrics

To evaluate the tracking performance, the CLEAR MOT metrics [16] is used. This metric consists of two metrics, the multi-object tracking precision (MOTP ↑) which evaluates the alignment of true positive trajectories w.r.t. the ground truth and the multi-object tracking accuracy (MOTA ↑) which calculates the accuracy composed of false positives, false negatives, and identity switches. Furthermore, the metrics proposed by Li *et al* [17] were computed: the ratio of mostly tracked (MT ↑) and mostly lost (ML ↓) trajectories as well as identity switches (IDS ↓). Here, the arrow symbol ↑ represents that higher scores indicate better results while ↓ means that lower scores indicate better tracking results.

3.3. Quantitative analysis

First, the tracking results for verifying the effectiveness of each proposed association stage are shown in Fig. 6-8.

Table 1: Performance comparison between the proposed method and other state-of-art methods

Dataset		Methods	MOTP (%)	MOTA (%)	MT (%)	ML (%)	IDS
PETS'09 S2L1	Online	Proposed	75.3	90.4	**94.7**	**0.0**	**6**
	Online	Breitenstein *et al.* [9]	56.3	79.7	-	-	-
	Online	Yang *et al.* [7]	53.8	75.9	-	-	-
	Offline	Berclaz *et al.* [18]	72.0	80.3	73.9	8.7	13
	Offline	Andriyenko *et al.* [4]	**80.2**	**90.6**	91.3	4.3	11
PETS'09 S2L2	Online	Proposed	**71.8**	56.8	**39.6**	**0.0**	230
	Online	Breitenstein *et al.* [9]	51.3	50.0	-	-	-
	Offline	Andriyenko *et al.* [4]	59.4	**56.9**	37.8	16.2	**99**
Town-Centre	Online	Proposed	**74.7**	66.3	56.1	8.7	260
	Online	Pellegrini *et al.* [8]	70.7	63.4	**59.1**	7.0	288
	Online	Yamaguchi *et al.* [13]	71.1	63.3	58.1	**6.5**	302
	Offline	Leal-Taixe *et al.* [17]	71.5	**67.3**	58.6	7.0	**165**

Fig. 6 shows the tracking results of the proposed method in complex situation where the pedestrian pointed by an arrow is constantly occluded by a traffic sign while changing its motion abruptly. While track ID3 corresponding to this pedestrian has drifted due to missing detections by occlusion before the 2nd association as shown in Fig. 6(b), this pedestrian become correctly tracked after applying the 2nd stage and re-update on track ID3 as shown in Fig. 6(c). Fig. 7 shows our tracking results in complex situation where the pedestrian pointed by an arrow is occluded by multiple pedestrians for a long-term. While the trajectory corresponding to this pedestrian is broken into two tracks ID46 and ID58 before the 4th association as shown in Fig. 7(b), this trajectory become restored after linking both fragmented tracks by the 4th stage as shown in Fig. 7(c). Additionally, the other result of handling of a fragmented track by the 4th association is shown in Fig. 8. During the interval #434~#476, the detection responses corresponding to the human pointed by an arrow are constantly missed because of the other human (track ID1) who sticks to this human. Thus, track ID2 becomes a lost track during this interval as shown in Fig. 8(b). In the Fig. 8(c) and (d), after the end of this situation, novice track ID7 of the same human is generated after stage 3 and it is linked with lost track ID2 by stage4. Therefore, these results in Fig. 6-8 confirm the effectives of the 2nd and 4th stage of the proposed hierarchical association framework.

The performance comparison results on the PETS 2009 and Town-Centre datasets are shown in Table 1 where the best results are highlighted in bold. In the PETS 2009 dataset, the proposed method provides better performance compared to other online tracking methods [7, 9] in terms of the MOTP and MOTA. Also, our method shows almost the same or even better performance compared to the offline tracking methods [4, 18]. Notably, this improvement of our method is achieved without future information, latency between detection and tracking result. In the Town-Centre dataset, the performance of the proposed method is shown best compared to the other online methods [13, 17] in terms of MOTP, MOTA and IDS.

The illustrative tracking results of the proposed method for each dataset are shown in Fig. 9. Overall, the proposed method tracks objects robustly while building longer trajectories even in complex situations where there are frequent and prolonged occlusions by object interactions and obstacles (e.g., traffic sign, benches) and abrupt motion changes of objects. However, in the TRK sequences of PETS 2016 dataset, the proposed method generally failed to track humans who are far away from the camera or walked closely together because of the critical failure of detecting these objects from the adopted detector [1]. Since the proposed method follows an online tracking-by-detection strategy, the performance of the proposed tracking method, inevitably, depends on that of an object detector.

The proposed method was implemented using MATLAB on a PC with a quad core 3.4GHz CPU and 8GB memory without any parallel programming and GPU processing. The average speeds of the proposed method for each crowd density scenario are about 149 fps for low density (PETS 2016), 39 fps for middle density (PETS09-S2L1), and 8 fps for high density (PETS09-S2L2, Town-Center), excluding detection costs. Hence, these experimental results indicate that the proposed tracking method with efficient object detectors can be applicable to real-time applications.

4. Conclusion

In this paper, a novel online multi-object tracking method is proposed for real-time applications. To track objects robustly in complex scenarios, we proposed a four-stage hierarchical association framework with the track confidence which can simultaneously handle the problems of track generation, progressive trajectory construction, track drift and fragmentation. Each association stage solves the different assignment problem composed of different

Figure 9. Tracking results of the proposed method for each dataset. At each frame, bounding boxes, recent trajectories (50 frames), and IDs of tracks are illustrated in color boxes, lines, and numbers.

detection and track sets based on their confidence for each goal that is to handle the above problems. Representative experimental results showed the improved performance of the proposed method, compared with other state-of-the-art methods.

References

[1] S. Ren, K. He, R. Girshick, J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks", In Advances in Neural Information Processing Systems, 2015, pp. 91-99

[2] P. F. Felzenszwalb, F. Fleuret, E. Turetken, et al, "Object detection with discriminatively trained part-based models", IEEE Trans. Pattern Anal. Mach. Intell., 2010, 32, (9), pp. 1627-1645

[3] P. Dollar, R. Appel, S. Belongie, et al, "Fast feature pyramids for object detection", IEEE Trans. Pattern Anal. Mach. Intell., 2014, 36, (8), pp. 1532-1545

[4] A. Andriyenko, S. Roth, K. Schindler, "Continuous energy minimization for multitarget tracking", IEEE Trans. Pattern Anal. Mach. Intell., 2014, 36, (1), pp. 58-72

[5] C. Huang, Y. Li, R. Nevatia, "Multiple target tracking by learning-based hierarchical association of detection responses",. IEEE Trans. Pattern Anal. Mach. Intell., 2013, 35, (4), pp. 898-910.

[6] H. Pirsiavash, D. Ramanan, C. C. Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects", Proc. IEEE Conference on Computer Vision and Pattern Recognition, Providence, USA, Jun. 2011, pp. 1201–1208

[7] J. Yang, Z. Shi, P. Vela, J. Teizer, "Probabilistic multiple people tracking through complex situations", Proc. IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, Miami, USA, Jun. 2009, pp. 79-86

[8] S. Pellegrini, A. Ess, K. Schindler, L. Van Gool, "You"ll never walk alone: modeling social behavior for multitarget tracking", Proc. IEEE International Conference on Computer Vision, Kyoto, Japan, Sep. 2009, pp. 261-268

[9] M. D. Breitenstein, F. Reichlin, B. Leibe, et al, “Online multiperson tracking-by-detection from a single, uncalibrated camera”, IEEE Trans. Pattern Anal. Mach. Intell., 2011, 33, (9), pp. 1820-1833
[10] Y. Cai, N. de Freitas, and J. J. Little, “Robust visual tracking for multiple targets”, in Proc. European Conference on Computer Vision, Graz, Austria, pp. 107-118, May. 2006.
[11] B. Wu, R. Nevatia, “Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors”, International Journal of Computer Vision, 2007, 75, (2), pp. 247-266.
[12] G. Shu, A. Dehghan, et al, “Part-based multiple-person tracking with partial occlusion handling”, in Proc. IEEE CVPR, Providence, RI, USA, 201
[13] K. Yamaguchi, A. Berg, L. Ortiz, and T. Berg, “Who are you with and where are you going?”, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, Providence, USA, Jun. 2011, pp. 1345-1352
[14] PETS’09 Dataset, http://www.cvg.reading.ac.uk/PETS2009/
[15] Town-Centre Dataset, http://www.robots.ox.ac.uk/ActiveVision/Research/Projects/2009bbenfobb_headpose/project.html
[16] K. Bernardin and R. Stiefelhagen, “Evaluating multiple object tracking performance: the CLEAR MOT metrics”, EURASIP J. Image and Video Processing, 2008, 2008, (1), pp. 1-10
[17] Y. Li, C. Huang, R. Nevatia, “Learning to associate: HybridBoosted multi-target tracker for crowded scene”, Proc. IEEE Conference on Computer Vision and Pattern Recognition, Miami, USA, Jun. 2009, pp. 2953-2960
[18] J. Berclaz, F. Fleuret, E. Turetken, *et al*, “Multiple object tracking using K-shortest paths optimization”, IEEE Trans. Pattern Anal. Mach. Intell., 2011, 33, (9), pp. 1806-1819
[19] X. Yan, X. Wu, I. Kakadiaris, S. Shah, “To track or to detect? An ensemble framework for optimal selection”, Proc. European Conference on Computer Vision, Florence, Italy, Oct. 2012, pp. 594–607
[20] PETS 2006 Challenge, http://www.pets2016.net/