Coupling Reranking and Structured Output SVM Co-Train for Multitarget Tracking

Yingkun Xu, Lei Qin, and Qingming Huang, *Senior Member, IEEE*

***Abstract*— Most of the previous works for multitarget tracking employ two strategies: global optimization and online state estimation. In general, global methods attempt to prevent local optimization and find the best results given global models. However, in time-critical applications, global optimization has long temporal latency. In contrast, most of the online algorithms obtain the states with greedy estimation, in which the errors are hard to be corrected. In this paper, we combine these two strategies and propose an online tracking approach with short-term storage to correct some local association errors. Based on structured output support vector machine, we propose a new framework with multiple online learners to produce multiple best local linkages, and novel features based on previously generated multiframe associations are designed for reranking of these multiple linkages. The reranking also serves as the appropriate mediator for updating of the online learners by co-train algorithm. The experimental results illustrate the advantage and robustness of this reranking algorithm, and its discrimination to find optimal ones. Comparison with some state-of-the-art methods proves that our proposed method is competitive to global optimal ones and is superior to other online tracking algorithms.**

***Index Terms*— Co-train, multitarget tracking, reranking, structured output SVM (SSVM).**

I. Introduction

VISUAL-BASED multitarget tracking is an important, but difficult topic in computer vision. Applications based on visual multitarget tracking can be roughly classified into two categories. One is related with offline analysis after events take place. Event retrieve, mining similar actions and searching specific activities are examples of this category. The other aims to react to the current time-critical scenarios, such as finding abnormal events immediately or predicting dangerous accidents. Our work aims to promote the performance of multitarget tracking for the second category applications.

According to whether detection responses of a video sequence are given as a whole or just coming frame-by-frame, two kinds of tracking approaches [1]–[8] are mainly proposed to handle the tracking problems based on associations of these detection responses. The first ones are based on global optimization within the whole sequence [1]–[4]. Considering the states of relationships between detection responses as being linked or not, detection associations can be modeled as network flow problems [1], hierarchical matching [2], or conditional random fields (CRFs) with different constrains [3], [4]. There are large feasible solution spaces for these model, thus global optimizations are employed to achieve promising performance. These global optimization algorithms considering the feasible relationships between detections in wide temporal ranges can obtain promising results. Thus, they are suitable for offline video analysis. The second ones attempt to association the detection responses in each new frames to the previously generated trajectories. They are temporal online tracking and are widely used for quick reaction for instantaneous tasks in time-critical applications. To implement online multitarget tracking, the greedy associations or bipartite matching approaches are employed using well-designed affinity metrics [5], [6].

Normally, the methods using global optimization need to infer the states of energy minimum or posterior probability maximum. These inferences are always expensive using processes like sampling [3] or graph cut [4]. Both of these algorithms try to retrieve better results step-by-step until they find the best one. These procedures can be considered as to see whether there are better solutions, then find them to correct errors. In contrast with global optimization, the online tracking processes are greedy algorithms. They try to find the best association in each frame, but errors cannot be corrected even they can be found in further steps. Therefore, how to embed the function of error correction is an important issue in online tracking process for time-critical applications.

In essence, online greedy associations are approximation of global optimization in local frames. The best solution of this local approximation is not stable to catch the tracking ground truth. By comparison, empirical studies [15], [17] show M-best solution are more powerful to cover them. For example, to track the articulated parts of person configurations, the M-best configurations of each frame are stitched and reestimated. Inspired from this idea, in online association, we can also generate multiple best solutions to recover from

Manuscript received July 14, 2014; revised December 31, 2014; accepted May 10, 2015. Date of publication May 14, 2015; date of current version June 2, 2016. This work was supported in part by the National Basic Research Program of China (973 Program) under Grant 2015CB351802 and Grant 2012CB316400 and in part by the National Natural Science Foundation of China under Grant 61332016, Grant 61025011, Grant 61133003, and Grant 61390510. This paper was recommended by Associate Editor A. Kokaram.

Y. Xu was with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China, and currently is with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, Zhejiang 310023, China (e-mail: yingkun.xu@vipl.ict.ac.cn).

L. Qin and Q. Huang are with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China. Q. Huang is also with University of Chinese Academy of Sciences, Beijing, 100049, China (e-mail: qinlei@ict.ac.cn; qmhuang@ucas.ac.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2015.2433173

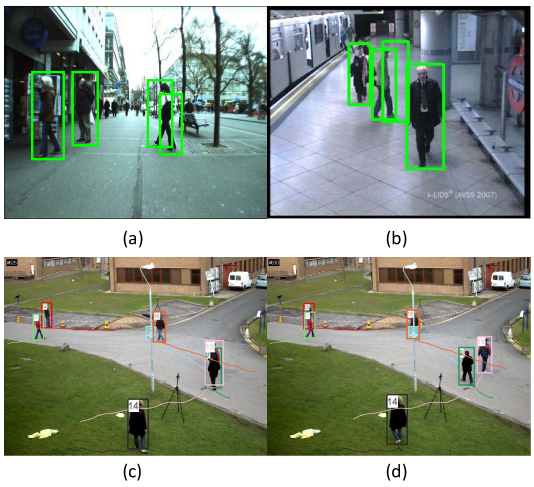


Fig. 1. (a) and (b) Detection responses are imprecise using two type detectors [23], [24]. (c) and (d) Object 15 has occluded template features.

local erroneous and ambiguous linkages, which are always caused by the false alarms, imprecise detection response, missed detection ground truth, and ambiguous description of tracked object. Fig. 1 shows some examples of these cases. In Fig. 1(a) and (b), the detectors [23], [24] obtain some imprecise results. To match with detection results in the next frame as in Fig. 1(c) and (d), the tracks may utilize their last detected features as their templates because these detections are most close to their next matched detections. However, this strategy is not suitable for the case of partial occlusion, in which the features of the last detections may be generated from the occluders rather than from tracked objects themselves. Thus, in all these cases, using one association is not a good solution for our tracking. Keeping multiple best solutions rather make it more possible to cover true association, and correct some errors.

In this paper, we propose a new online multitarget tracking method based on multiassociations. Instead of using one feature to match, we combine appearance and motional features in multiple online structured output SVMs (SSVMs) framework, and generate multiple best associations considering different features spaces between tracks and detections. The final association is ranked and determined by the multiframe linkages considering previous long-term multiframe assignments.

The insight of our approach aims to hybridize the frame-to-frame local affinity and long-term associations into one online tracking framework. We utilize weighted multiple features for frame-to-frame matching to generate and store local multiple best solutions. Through using multiple feature spaces, we can explore wider association space, and have more probability to cover correct linkages. We select the best solution using high-order multiple frame features which are more informative and discriminative than frame-to-frame affinity. Those high-order features are widely used in the CRF model [3], [4]. However, it is time consuming and even intractable to train and to infer the best association status using arbitrarily complex features in CRFs. By comparison, reranking the possible associations using these long-term multiframe features is more efficient and concise. Thus, the online solution can be designed using the strategy of local short-term multiassociation storage and global long-term reranking. This mechanism of short-term information storage and error correction are also supported well by the human vision system [30]. Evidence shows human can keep large amount of short-term information in low level to help analysis the high-level informative contents. This phenomenon accords well with our proposed method. We employ the co-train algorithm induced by the long-term association reranker to update the multiple online SSVM learners. Because tracking process will generate large amounts of new unlabeled data, semisupervised learning process is a good choice to handle them.

The main contributions of this paper lie in the following points.

1) We develop a novel online multitarget tracking framework using reranking strategy. Compared with global optimization, which directly infer the most probable posterior within the CRF framework, we can use more complex long-term association features.
2) We propose a new method to generate online multiple hypotheses without bringing in the problem of combinational explosion. These multiple hypotheses consider different feature spaces between tracks and detections, and are only different in the case of ambiguous linkage.
3) A set of discriminative features are proposed for multiframe association reranking. Experimental results prove they are effective for reranking the candidate associations based on previous tracking results.
4) The co-training process induced by reranker is employ to update the multiple online learners, to the best of our knowledge, which is the first time using co-train for online multitarget tracking.

The remainder of this paper is organized as follows. Section II introduces some related works. Sections III–V describe our proposed online learning method for data association structure. Section VI presents experimental results. Finally, we conclude this paper in Section VII.

II. Related Works

The core motivation of our paper is how to integrate global association into online tracking framework. Our strategy lies from the spirit of buffering multiple short-term solutions and find the best one using long-term information. We instantiate this strategy in the online multitarget tracking with structured output learning and reranking algorithm. We outline the related previous works in the following aspects.

Generating a small set of feasible solutions, and finding the best one after accumulation of evidences is a common strategy in the field of visual computing. It is a multistage cascading procedure in essence. The face detection algorithm [9] illustrates its advantages in performance and speed. This strategy is applied for measurement appraisal in the multiple hypotheses tracking (MHT) [7] and joint probabilistic data association filter (JPDAF) [8]. These approaches depend on the stages of finding the candidate M-best solutions and choosing the optimal one with largest sum of log probabilities of leaf branches. The cascading algorithms have also been

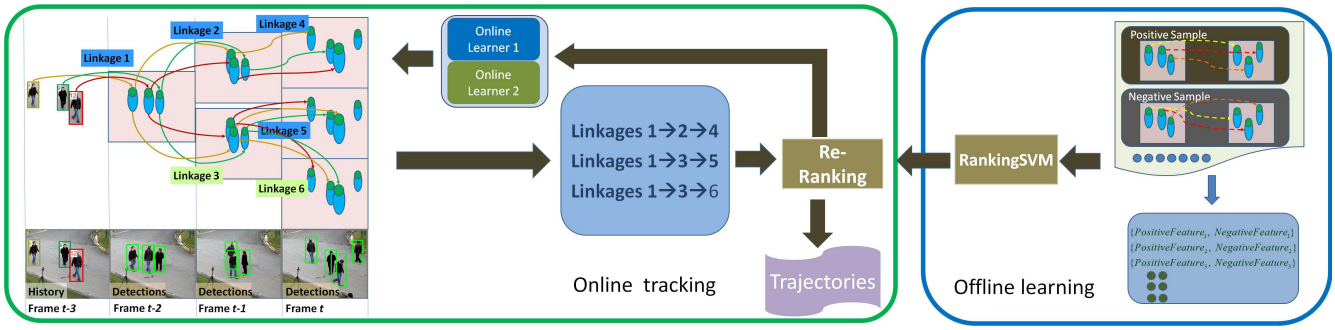


Fig. 2. Framework of our proposed method. In offline learning, the positive trajectories samples are extracted, and negative samples are constructed by degrading these positive samples. The RankingSVM is learned and used to discriminate the best one among all candidate associations in online process.

utilized within particle filter framework [10], [11]. Different from these algorithms, we combine the reranking algorithm to choose the association results of the online structured learning.

Combination of global optimization and online updating local features is utilized in [12] and [13]. It has been proved to be effective to improve tracking performances using the selected features which are trained with large scale data set. In [12], a most discriminative feature pool is learned beforehand, and they serve as the candidate features for each gallery track segment. In [13], a deep stacked denoising autoencoder is employed to learn the robust features from training data set, and the features are updated in online tracking using fine tuning. By contrast, our proposed approach can be considered as a tradeoff between global optimization and online feature learning during online multitarget tracking.

There have been some works related with getting M-best configurations in probabilistic model [14], [15], or getting M-best solutions for energy minimization [16], [17]. The basic idea behind these algorithms is to construct the next best problem by discarding the optimal solution from the original problem. For example, in [17], the divide-and-conquer strategy is used to get solution partitions of suboptimal problems. The next best solution is chosen from these problem partitions. In [16], the relaxed optimization problem is constructed by adding the constraints of at least M dissimilarity from the previous best solution. The optimization methods like gradient descent in dual form can be used to get the suboptimal solution. In our work, differently, we use different feature spaces to get at most M-best solutions within the framework of structured learning. Thus, we bypass the problem of combinatorial explosion, and avoid the consuming searching when it is unnecessary.

Among the candidate M-best solutions, we utilize the reranking algorithm to score and select the potential best one. Reranking algorithms are transferred from natural language processing [18] and information retrieval [19] to the problems of visual tracking [20], [21]. In [20], a CRF model is constructed to represent the possible connections between detection responses. RankBoost algorithm is employed to train the model with sampling association pairs. In [21], the weakly supervised ranking algorithm is proposed to learn the weights of appearance features. The graph Laplacian is used to regularize the smoothness of similarities between samples. Different from above methods, we employ the high order and complex features from multiframes associations to preferably assess the correctness of the candidate solutions, and appraisal which one is the best.

Co-train is an important semisupervised learning algorithm when there are large amount of unlabeled data. We use co-train to update the SSVMs for online association. Co-train is first proposed to classify the Web pages with different feature spaces [31], and obtain over 96% accuracy. Later, this algorithm is extended to computer vision, such as detection [32] and tracking [33]. In [32], data from different views are extracted from the features of pedestrian and vehicle, and they are employed to train two classifiers, which are online learned using unlabeled data. In [33], the color features and histogram of gradient (HoG) are used to construct the classifiers. These classifiers are weighted to locate the final tracking result and updated using unlabeled samples. In our method, we update the different learners, which generate different association candidates, with the induction of long-term association reranker. This reranker also acts as the mediator to obtain the final tracking results.

III. Framework

In Fig. 2, we illustrate the framework of our proposed method. Our method are composed of three stages: 1) construction of multiple candidate linkages; 2) finding the best multiple linkages according to offline learned Ranking SVM; and 3) updating the multiple matching learners using co-train algorithm.

As shown in the left part of Fig. 2, at each moment t during online tracking process, we need to find the best matching between tracks and detection responses. However, as discussed in [15], utilizing one best matching with one association learner may loss the correct linkage corresponding to that in tracking ground truth. To construct more best matching results, we construct multiple matching learners. As inspiration from [33], in our method, we set up the online SSVM learners from the color appearance features and gradient texture features respectively. These independent features construct their corresponding learners to find their own matching results. In most cases, these matching results are same because the matching states are clear. However, in other ambiguous conditions, they may generate different

matching results. Thus, these different matching results are more probable to cover true linkages. In these cases, we need to choose which one should be accepted.

To find the best linkage among all candidates, multiframe linkages are more informative than that to one frame. This informs us to store local short-term linkage candidates at recent moments. To choose which one linkages should be accepted, we combine the global features of previous long-term tracking features. These features can well describe the consistency between these short-term linkages and previous long-term trajectories. To learn the best ranking learner, we employ the RankingSVM [26] to learn optimized combination of multiple features. Experimental results show the effectiveness of these combined multiple features for ranking.

The reranking procedure can not only find optimal association results for tracking but also provides the cues for online learning of multiple SSVM learners. For these multiple online learners, we employ co-train algorithm to search feasible data during tracking. Because tracking will generate large amounts of unlabeled data, this co-train learning is an appropriate process to update these multiple online learners for matching.

Detailed descriptions about our method are discussed in the following sections.

TABLE I
MAINLY USED NOTATIONS

Symbols	Description
$[t_a : t_b]$	The discrete set from t_a to t_b. If t_a and t_b is frame number, then the discrete set represents one frame range from t_a to t_b.
Δt	The length of recent short-term association frames.
ΔT	The length of previous long-term association tracks.
s_j^t	The j-th detection response at frame t.
r_i^{t-1}	The generated i-th track before frame t.
$a_{i,j}^t$	The affinity between track r_i^{t-1} and detection s_j^t.
y_t	The association result between tracks and detection responses at frame t.
$R_{[t-\Delta t-\Delta T:t]}$	The trajectory set which are overlapped with the frame range from frame $t-\Delta t-\Delta T$ to frame t.
$S_{[t-\Delta t-\Delta T:t]}$	The detection set which are detected within the frame range from frame $t-\Delta t-\Delta T$ to frame t.
$y_{[t-\Delta t:t]}$	The association between trajectory set $R_{[t-\Delta t-\Delta T:t]}$ and detection set $S_{[t-\Delta t-\Delta T:t]}$ within the frame range from $t-\Delta t$ to t.
$r_k^{[t-\Delta t-\Delta T:t]}$	The k-th trajectory in the trajectory set $R_{[t-\Delta t-\Delta T:t]}$ which has detection set within the frame range from $t-\Delta t-\Delta T$ to t.
t_s^k, t_e^k	The first frame number and the last frame number of the trajectory $r_k^{[t-\Delta t-\Delta T:t]}$ within the frame range from $t-\Delta t-\Delta T$ to t.
$s_k^{t_p}$	The detection response at the frame t_p for the trajectory $r_k^{[t-\Delta t-\Delta T:t]}$.
$B(r_k^{[t-\Delta t-\Delta T:t]})$	One B-spline curve fitting for the k-trajectory $r_k^{[t-\Delta t-\Delta T:t]}$.
$\theta_k^{t_p}$	The motion angle of the k-trajectory $r_k^{[t-\Delta t-\Delta T:t]}$ at frame t_p.
$P_k^{t_p}$	The location point of detection bounding box of the k-trajectory $r_k^{[t-\Delta t-\Delta T:t]}$ at frame t_p.
$W_k^{t_p}$	The width of detection bounding box of the k-trajectory $r_k^{[t-\Delta t-\Delta T:t]}$ at frame t_p.
$v\theta(\Delta t_\tau)_k^{t_p}$	The angle velocity of the k-trajectory $r_k^{[t-\Delta t-\Delta T:t]}$ at frame t_p.
$vP(\Delta t_\tau)_k^{t_p}$	The linear velocity of the k-trajectory $r_k^{[t-\Delta t-\Delta T:t]}$ at frame t_p.

IV. RERANKING-BASED MULTITARGET TRACKING

As discussed above, our method maintains M-best associations between tracked objects and detection responses within recent short-term frames, and then chooses the best one as tracking result considering the long-term tracking trajectories as evidences. To clarify the description of our method, we summarize the mainly used notations in Table I.

In Sections IV-A–IV-D, we first propose a multiple online SSVM learning approach to associate tracked objects and detections. Then we introduce the long-term association-based reranking and its learning algorithm. Next, we give the online co-train procedure to update the weights of online SSVM adaptively. Finally, we give the tracking flowchart using our method.

A. Multiple Online Associations Using Multilearners

Learning multiple features and combining them can enhance the robustness of online multitarget tracking. However, instead of updating features online, most of the previous works discussing how to integrate multiple features by learning the weights offline using large scale training samples [5], [6], or generate the affinities using generative models like detection confidence [11]. It is necessary to design a mechanism to update the weights of multiple features during online multitarget tracking.

Denoting the jth detection response in frame t as $s_j^t = (b_j^t, \{o_{j,k}^t\}_{k=1}^{K_1}, \{o_{j,k}^t\}_{k=1}^{K_2})$, where b_j^t is the bounding box and $o_{j,k}^t$ is its kth feature. These features are composed by two subsets $\{o_{j,k}^t\}_{k=1}^{K_1}$ and $\{o_{j,k}^t\}_{k=1}^{K_2})$, each of which corresponds to color or texture features. Because these two kinds of features are independent, we can construct different matching learners using these two subset features. Further, we can define the ith candidate track before frame t as the detection response list $r_i^{t-1} = \{s_i^{t-l}, s_i^{t-l+1}, \ldots, s_i^{t-1}\}$ from the start frame $t-l$ to frame $t-1$, as in Fig. 3. Given the set of detection responses in frame t is $S^t = \{s_j^t\}_{j=1}^n$ and the set of candidate tracks for frame t is $R^{t-1} = \{r_i^{t-1}\}_{i=1}^m$, we can calculate the K_1 or K_2-dimensional affinity $a_{i,j}^t = \phi(r_i^{t-1}, s_j^t)$ as (3) between r_i^{t-1} and s_j^t based on the K_1 or K_2 dimension features defined for each detection response. These K_1 or K_2-dimensional affinity vectors can be integrated into one affinity scalar using their weight w^t. According to these affinity scalars between tracks and detections, we can obtain the linking result.

Let a binary vector set $\mathcal{Y}_t = \{y_t | y_t = [y_{1,1}^t, \ldots, y_{m,1}^t, y_{1,2}^t, \ldots, y_{m,n}^t]^T\}$ represent the linking result candidates, in which $y_{i,j}^t$ indicates whether ith tracked object is linked to jth detection. There should be the constraints $\sum_i y_{i,j} \leq 1$ and $\sum_j y_{i,j} \leq 1$ because one candidate track is matched with one detection response at most. To solve the problem of how to obtain the online association y^t. The optimal bipartite

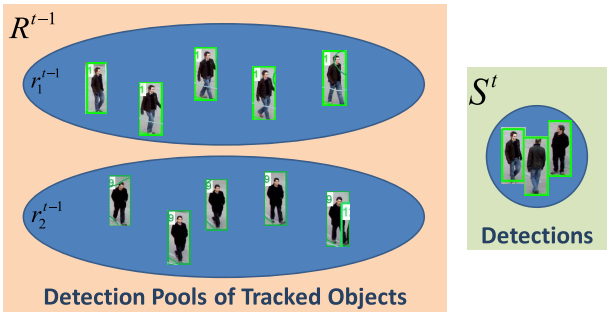


Fig. 3. Each tracked object has a pool of recent linked detection objects r_i^{t-1}. We should construct the template using the detection pool to match with the future detection responses S^t.

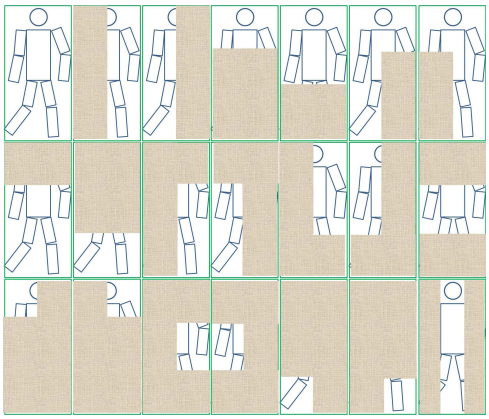


Fig. 4. Twenty-one parts of different occlusion cases defined for the features construction of detection responses.

matching method can be employed to obtain y_t by

$$y_t = \arg\max_{y \in \mathcal{Y}_t} \langle w, y^T \Phi(R^{t-1}, S^t) \rangle \tag{1}$$

where $\Phi(R^{t-1}, S^t) = [a^t_{1,1}, \ldots, a^t_{m,1}, a^t_{1,2}, \ldots, a^t_{m,n}]^T$ is the feature matching matrix.

There are three problems needed to be discussed in detail.

First, how to obtain the features for each track to match with the detections. Because the tracked trajectories are composed by a list of detection responses, we need to construct the features for each corresponding object as shown in the left part of Fig. 3. We adopt the strategy of using one typical detection response in the representative set to match with future detections. This detection response should reflect the recent features of the object, but not be degraded by occlusion. Thus, when new detection response $s^t_{j_i}$ is matched to the track r_i^{t-1}, the representative detection $s_r(r_i^t)$ of r_i^{t-1} should be updated as

$$s_r\left(r_i^t\right) = \begin{cases} s^t_{j_i} \\ s_r\left(r_i^{t-1}\right) \end{cases} \quad \exists s_k^{t-1} \in \tilde{r}_i^{t-1}, \ \text{aff}\left(s^t_{j_i}, s_k^{t-1}\right) > \varepsilon \tag{2}$$

where aff() is the affinity function, and ε is the threshold of affinity between the element detection s_k^{t-1} at frame $t-1$ and the new matched detection $s^t_{j_i}$. $\tilde{r}_i^t$ is the representative set for track i at frame t. For the first case in (2), $\tilde{r}_i^t = \tilde{r}_i^{t-1} \cup s^t_{j_i}$.

Second, we should define the affinity between tracks and detection responses considering the occlusions, imprecise detection boxes. Considering the usual occlusion states between pedestrians, we construct the different parts to form the multidimensional features, as shown in Fig. 4. Considering different cases of imprecise detections, we align each side of detection response to the track template, which are obtained from Kalman filter. Thus, we can get four affinity scalars for four alignment cases. Denote the set of these alignment cases as A_n and ith transformation of alignment case as function $\text{align}_i()$, then the affinity for kth part feature is expressed as

$$\phi_k\left(r_i^{t-1}, s_j^t\right) = \max_{\text{align}_i \in A_n} \text{aff}_k\left(s_r\left({r_i}^{t-1}\right), \text{align}_i\left(s_j^t\right)\right). \tag{3}$$

In this expression, the best subset of alignments is considered to avoid misalignment cases. Thus, we can compensate the affinity imprecision between tracks and detections due to imprecise detection responses.

Third, we utilize the online passive aggressive algorithm [22] based on current association result to update the weight w^t as

$$\begin{aligned} w^{t+1} = \arg\min_{w} \quad & \frac{1}{2}\|w - w^t\|^2 + C_1\xi^2 \\ \text{s.t.} \ & \max_{y \in \mathcal{Y}_t} \Delta(y, y_t) - \left\langle w, \delta y_t^T \Phi(R^{t-1}, S^t) \right\rangle \le \xi \\ & \delta y_t^T \Phi(R^{t-1}, S^t) \equiv y_t^T \Phi(R^{t-1}, S^t) \\ & \qquad - y^T \Phi(R^{t-1}, S^t) \end{aligned} \tag{4}$$

where the loss function between augmented inferred association y and true association is defined as $\Delta(y, y_t) = (1-y)^T y_t$ [6]. This problem has a closed-form solution [22] with robustness to noisy samples

$$w^{t+1} = w^t + \tau^t\left(y_t^T \Phi(R^{t-1}, S^t) - \tilde{y}^T \Phi(R^{t-1}, S^t)\right) \tag{5}$$

where

$$\tau^t = \frac{\Delta(\tilde{y}, y) - \left\langle w^t, y_t^T \Phi(R^{t-1}, S^t) - \tilde{y}^T \Phi(R^{t-1}, S^t) \right\rangle}{\left\| y_t^T \Phi(R^{t-1}, S^t) - \tilde{y}^T \Phi(R^{t-1}, S^t) \right\|^2 + 1/2C_1}$$

$$\tilde{y} = \arg\max_{y \in \mathcal{Y}_t} \Delta(y, y_t) + \langle w^t, y^T \Phi(R^{t-1}, S^t) \rangle.$$

The above formulations present an efficient solution for online multitarget tracking by incremental learning. Same as in [6], we define the loss function $\Delta(y, y_t) \equiv y^T(1 - y_t)$.

The formulations above describe how we construct a matching learner. We implement two match learner using two types of features. These can generate complementary solutions in online tracking. Furthermore, we need to discuss how to find the best candidate association from the view of long-term optimization.

B. Reranking of Multiple Online Associations

To cover the true association as possible as we can, we employ different matching learners using different feature spaces to generate multiple association candidates. These multiple association candidates are accumulated based on previous

ones, and obtain a huge number of associations in worst cases. However, the number of candidate association increases slowly in most times, because there are only confident matching solutions when there are no ambiguous decisions. Multiple matching learners will generate one same matching result at these times. In some ambiguous situations, these matching algorithms may generate optional associations. We need to find which one is the best and prune some low-probability candidate results.

To find the best one of the multiple alignments at the frame t, we store the candidate associations in recent short-term Δt frames based on more early previous ΔT tracking results. Because the accumulative features are more discriminative than those from one single frame, we accumulate evidences of multiple frame association in these $[t-\Delta t-\Delta T:t]$ frames. Then, we employ the reranking algorithm with the accumulative features to select the best candidate multiframe associations.

The function to score the multiframe associations based on above multiple associations is expressed as

$$f_{[t-\Delta t-\Delta T:t]}(y_{[t-\Delta t:t]}) = \alpha^T \Psi(R_{[t-\Delta t-\Delta T:t]}, S_{[t-\Delta t-\Delta T:t]}, y_{[t-\Delta t:t]}) \quad (6)$$

where α is the weight of reranking features $\Psi(R, S, y)$, $S_{[t-\Delta t-\Delta T:t]}$ is the set of detections in frames $[t-\Delta t-\Delta T:t]$, and $R_{[t-\Delta t-\Delta T:t]}$ are trajectories within frames $[t-\Delta t-\Delta T:t]$ by considering short-term associations $y_{[t-\Delta t:t]}$. The best association of y_t can be obtained by $y_t = \arg\max_{y\in\tilde{\mathcal{Y}}_t} f_{[t-\Delta t-\Delta T:t]}(y_{[t-\Delta t:t]})$, in which $\tilde{\mathcal{Y}}_t$ is the candidate association set obtained from Section IV-A considering multiassociation learners.

The key to find optimal choice of candidate associations is to design appropriate features $\Psi(R_{[t-\Delta t-\Delta T:t]}, S_{[t-\Delta t-\Delta T:t]}, y_{[t-\Delta t:t]})$ for these associations. Intuitively, the short-term associations $y_{[t-\Delta t:t]}$ in frames $[t-\Delta t:t]$ are expected to be consistent with the previous long-term associated trajectories backward to frames $[t-\Delta t-\Delta T:t]$, as shown in Fig. 5.

To achieve this point, we design the features in three aspects. First, we expect the observed detection responses in each trajectory between frames $[t-\Delta t:t]$ should have consistent appearance and motional trends as its previous part in frames $[t-\Delta t-\Delta T:t-\Delta t]$. Second, the relationship between every two trajectories should have potential consistency with the scenario, while keeping the exclusions between each other. Third, the statistical attributes of the associations should have similar distribution as those in the training data set.

We discuss in detail for these features in the following three aspects.

1) Intra-Consistency Features: The features are designed to keep intra-consistency for trajectories from the points of appearance, shape and motion. Given the trajectory set $R_{[t-\Delta t-\Delta T:t]} = \{r_k^{[t-\Delta t-\Delta T:t]}\}_{k=1}^{M}$ containing M trajectory segments, we split the frame length ΔT to N_a parts $[t-\Delta t-\Delta T_i:t-\Delta t-\Delta T_{i+1}]$, $i\in[1..N_a]$, using same log-length frame intervals. We denote the kth track

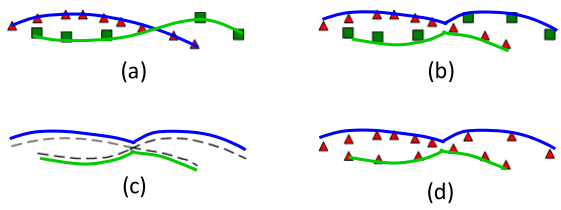


Fig. 5. Example of intra-consistency within trajectories. The inconsistency of appearance (b), trajectory shape (c), and motion trends (d) cause the association inferior to the true associations in (a) when ranking.

$r_k^{[t-\Delta t-\Delta T:t]}$ as the detection list $\{s_k^{t_s^k},\ldots,s_k^{t_e^k}\}$. For each interlaced frame $t_j\in[t-\Delta t:2:t]$, the appearance intra-consistency feature can be expressed as $\Psi_{i,j}^1$

$$\Psi_{i,j}^1 = \frac{1}{M}\sum_{k=1}^{M} I\left(t_s^k < t-\Delta t-\Delta T_i\right) \max_{t_p\in[t-\Delta t-\Delta T_i:t-\Delta t-\Delta T_{i+1}]} \mathrm{Aff}\left(s_k^{t_p}, s_k^{t_j}\right) \quad (7)$$

where $I(.)$ is the indicator function, and $\mathrm{Aff}(s_k^{t_p}, s_k^{t_j})$ is the similarity between two detections $s_k^{t_p}$, $s_k^{t_j}$ using Bhattacharyya coefficient. Using three types of appearance information [HSV, local binary pattern (LBP), and RGB color], we can obtain 45D features for appearance similarities based on three interlaced short-term frames and 64 frames of previous tracking results ($N_a=5$).

To express the intra-consistency of the shape for trajectories, we expect the smoothness of fitting curves is good as possible as those in ground truth. Thus, fitting the detection points and computing the errors of detection points in the short-term Δt frames is very important clues. To allow motional changes, we fit the curve for detections within different frames, with the length as half of the previous length. For example, looking backward 64 frames of previously tracking results, there are four groups of curves $B_i(r_k^{[t-\Delta t-\Delta T:t-\Delta t]})$, where the ith group of curves is obtained from temporal range $[t-\Delta t-2^i\Delta\tau:t-\Delta t]$, $i\in[1..4]$, and τ is minimal fitting length. Thus, for the ith group of curves, the smoothness for all tracks can be

$$\Psi_i^2 = \frac{1}{M}\sum_{k=1}^{M} I\left(t_s^k < t-\Delta t-2^{i-1}\Delta\tau\right) \times \frac{1}{\Delta t}\sum_{t_j\in[t-\Delta t:t]} \exp\left(-\lambda_1 \left\| P_k^{t_j} - B_i\left(r_k^{[t-\Delta t-\Delta T:t]}\right)\right\|_2\right) \quad (8)$$

where $\|P-B_i(r_k^{[t-\Delta t-\Delta T:t-\Delta t]})\|_2$ is the Euclidean distance from the center point P to the curve $B_i(r_k^{[t-\Delta t-\Delta T:t-\Delta t]})$.

To illustrate intra-consistency of motional trends for the trajectories, we consider the motional direction and velocity, respectively. Given the directions and locations of the trajectory $r_k^{[t-\Delta t-\Delta T:t]}$ as $\{\theta_k^{t_s^k},\ldots,\theta_k^{t_e^k}\}$ and $\{P_k^{t_s^k},\ldots,P_k^{t_e^k}\}$, we can compute their angular velocity and linear velocity at each moment as $\{v\theta(\Delta t_\tau)_k^{t_i^k} = (\theta_k^{t_i^k} - \theta_k^{t_i^k+\Delta t_\tau})/\Delta t_\tau\}$ and

$\{\upsilon P(\Delta t_\tau)_k^{t_i^k} = (P_k^{t_i^k} - P_k^{t_i^k+\Delta t_\tau})/\Delta t_\tau\}$ by consideration of different temporal interval Δt_τ. It is helpful to tolerate the misalignments of detections using different Δt_τ because the small perturbations exist for detection responses. Assuming the angular velocities obey von Mises distribution as in [25] and the linear velocities obey Gaussian distribution, the motion consistencies based on temporal interval Δt_τ are expressed as

$$\Psi^3_{\Delta t_\tau} = \frac{1}{M}\sum_{k=1}^{M}\frac{1}{\Delta t} \times \sum_{t_j\in[t-\Delta t:t]} \frac{1}{2\pi I_0(\lambda_2)}\exp\left(\lambda_2\cos\left(\upsilon\theta(\Delta t_\tau)_k^{t_j}\right)\right) \quad (9)$$

$$\Psi^4_{\Delta t_\tau} = \frac{1}{M}\sum_{k=1}^{M}\frac{1}{\Delta t} \times \sum_{t_j\in[t-\Delta t:t]} \exp\left(-\frac{\left(\upsilon P(\Delta t_\tau)_k^{t_j} - \upsilon P(\Delta t_\tau)_k^{t_j-1}\right)^2}{2\sigma_k^2(\upsilon P(\Delta t_\tau)_k)}\right) \quad (10)$$

where I_0 is modified Bessel function of order 0. Using three different values for Δt_τ, e.g., (1, 2, 4), we can get 6D features.

2) Intertrajectory Features: The features are expected to obtain the information between every two trajectories. These features should reflect the information of mutual exclusion among the trajectories, as well as coincident with context. For example, people always walk along the limited paths since there are not too many roads in one scenario. Thus, some persons will pass along similar paths when there are lots of people walking through. We design the features from the aspects of trajectory shapes. To obtain the features between the trajectory shapes, we compute the chamfer distance between every two trajectories. We keep its five-bin histogram after normalization by the width of bounding box $\{W_k^{t_s^k}, \ldots, W_k^{t_e^k}\}$. The value of i-bin is

$$\Psi_i^5 = \frac{1}{M(M-1)} \times\sum_{u\neq v}^{M} I\left(L(i) < \mathrm{Chd}\left(r_u^{[t-\Delta t-\Delta T:t]}, r_v^{[t-\Delta t-\Delta T:t]}\right) < H(i)\right) \quad (11)$$

where

$$\mathrm{Chd}(r_u, r_v) = \frac{1}{|r_u|}\sum_{t_j\in[t_t^u:t_e^u]}\min_{t_l\in[t_t^v:t_e^v]}\left(\left\|P_u^{t_j} - P_v^{t_l}\right\|/W_u^{t_j}\right)$$

where $L(i)$ and $H(i)$ are the low and high boundary for bin i. We set the values as (0, 0.5, 1, 2, 4) and (0.5, 1, 2, 4, $+\infty$) for five bins, respectively.

3) Statistic Features: The statistic features of the trajectory group in the temporal range $[t - \Delta t - \Delta T : t]$ are calculated according to the ground truth. We expect the features obtained from the test associated trajectories are matched to these statistic values as much as possible. We utilize three statistic features. The first is the average occluded length. Inspired by [4], we use Cauchy–Lorentz distribution to model the consecutive occluded frames of each trajectory. The average weighted value is

$$\Psi^6 = \frac{1}{M}\sum_{k=1}^{M}\prod_{\Delta j\in\mathrm{gaps}\left(r_k^{[t-\Delta t-\Delta T:t-\Delta t]}\right)}\frac{\lambda_3}{\left|r_k^{\Delta j}\right|^2+\lambda_3^2} \quad (12)$$

where $\mathrm{gaps}(r)$ is the occluded segments of track r, and $|r_k^{\Delta j}|$ is the consecutive frame length for occluded segment Δj.

The second statistic information is the number of started trajectory and the number of terminated trajectory between the temporal range $[t - \Delta t : t]$. We assume they are modeled by exponential distribution. Thus, we obtain the features related with these numbers as

$$\Psi^7 = \exp\left(-\lambda_4\frac{1}{M}\sum_{k=1}^{M} I\left(t-\Delta t <= t_s^k < t\right)\right) \quad (13)$$

$$\Psi^8 = \exp\left(-\lambda_5\frac{1}{M}\sum_{k=1}^{M} I\left(t-\Delta t <= t_e^k < t\right)\right). \quad (14)$$

The third statistic information is the length of trajectory. We hope the trajectory extends as long as possible. Thus, too short trajectories are unexpected because they are always obtained by abnormal linkages. We calculate this attribution as the following feature:

$$\Psi^9 = \exp\left(-\lambda_6\frac{1}{M}\sum_{k=1}^{M}\frac{1}{t_e^k - t_s^k}\right). \quad (15)$$

Above nine types of features are appended to appraisal whether the generated trajectories are better or not. The parameters $(\lambda_1, \lambda_2, \ldots, \lambda_6)$ are estimated by training data, as described in Section VI-A. Setting $\Delta t = 5$ and $\Delta T = 64$, there are 64D features by considering all above three aspects' features. By experiments, they are suitable for discrimination between positive associated tracks and negative ones.

C. Ranking SVM Learning of Multionline Associations

Because our goal is to appraise multiframe associations obtained using multiple matching learners, we need to prioritize them using one score function $f_{[t-\Delta t-\Delta T:t]}(.)$. This is a problem of reranking learning for structured output as discussed in [26]. Given one pair of association results $y^i_{[t-\Delta t:t]}$ and $y^j_{[t-\Delta t:t]}$ such that the former has higher priority than the latter, denoting as $y^i_{[t-\Delta t:t]} \succ y^j_{[t-\Delta t:t]}$, we hope their values using score function have relationship

$$y^i_{[t-\Delta t:t]} \succ y^j_{[t-\Delta t:t]} \Leftrightarrow f_{[t-\Delta t-\Delta T:t]}\left(y^i_{[t-\Delta t:t]}\right) > f_{[t-\Delta t-\Delta T:t]}\left(y^j_{[t-\Delta t:t]}\right). \quad (16)$$

Considering the linear weighted formulation defined in (6) and features defined in the previous section, we need to obtain the optimal weight vector α. This reranking for structured output can be solved efficiently using cutting-plane algorithm [28]. We employ the scaling slack form to learn the

weight vector

$$\alpha = \arg\min_{\alpha} \frac{1}{2}\|\alpha\|^2 + C_2 \sum_{i\in[1:|Q|]} \zeta^i \tag{17}$$
$$\text{s.t. } \alpha^T \left(\Psi\left(R_{[t_i-\Delta t-\Delta T:t_i]}, S_{[t_i-\Delta t-\Delta T:t_i]}, y^{\circ}_{[t-\Delta t:t]}\right) - \Psi(R_{[t_i-\Delta t-\Delta T:t_i]}, S_{[t_i-\Delta t-\Delta T:t_i]}, y_{[t-\Delta t:t]})\right) \geq 1 - \frac{\zeta^i}{L(y^{\circ}_{[t-\Delta t:t]}, y_{[t-\Delta t:t]})} \quad \forall \zeta^i \geq 0$$
$$\forall y_{[t-\Delta t:t]} \neq y^{\circ}_{[t-\Delta t:t]} \ \wedge \ y_{[t-\Delta t:t]} \in Y_{[t-\Delta t:t]}.$$

In our method, we extract the training sample set Q from the ground truth. Each sample in this set is composed by the detections $S_{[t_i-\Delta t-\Delta T:t_i]}$ with at most length $\Delta T + \Delta t$, their association results $y^{\circ}{}_{[t-\Delta t:t]}$ by changing some linkages in ground truth as introduced in experiments, and the associated tracks $R_{[t_i-\Delta t:t_i]}$. The loss $L(y^{\circ}{}_{[t-\Delta t:t]}, y_{[t-\Delta t:t]})$ between two associations $y^{\circ}{}_{[t-\Delta t:t]}$ and $y_{[t-\Delta t:t]}$ are defined as the Hamming loss of the $y_{[t-\Delta t:t]}$ when $y^{\circ}{}_{[t-\Delta t:t]}$ is same with ground truth. Otherwise we calculate the Hamming distances for $y_{[t-\Delta t:t]}$ and $y^{\circ}{}_{[t-\Delta t:t]}$ with ground truth, and define the loss function as the difference between these two Hamming distances. We use efficient one-slack algorithm [27] to train the weight vector α.

D. Online Association Learners Co-Train With Ranking SVM

Our method employs two different feature sets $\{o^t_{j,k}\}_{k=1}^{K_1}$ and $\{o^t_{j,k}\}_{k=1}^{K_2}$ to describe each detection response s^t_j. The first features explore the color spaces, including the HSV and RGB color histograms. The second features explore the gradient and texture-based features, including the HoG and LBPs. These two feature sets reflect independent features and construct two online association learners from their own aspects. We employ the online learning algorithms to update the weights as in (5). One important problem is how to obtain the appropriate samples during online association.

Similarly with tracking a single object, we cannot confirm the labels of samples during data association between tracks and detections. Learning with audacious labeled samples will lead to problems of identity switching and appearance confusion like the drifting issue. Thus, a well-designed semi-supervised learning algorithm is preferred. Considering the mechanism of multionline association learners and long-term reranking-based selection, a natural strategy is to utilize the co-train algorithm. The co-train algorithm is first proposed by Blum and Mitchell [31]. It augments the training set using unlabeled data because labeling samples are expensive. In the co-train method, the features are divided into two independent views. These two views have sufficient attributes such that learning algorithm can be done rightly using either of them. During online learning, one learner can attach the labels to update the another weak learner. In [31], it has been proved that co-train process can rightly update the weak learners, which are initiated with small labeled samples, if the weak learners are learnable from the view of probably approximately correct (PAC) model and their corresponding attributes are conditionally independent given the labels. These co-train learners have low generalization errors than combining the features directly.

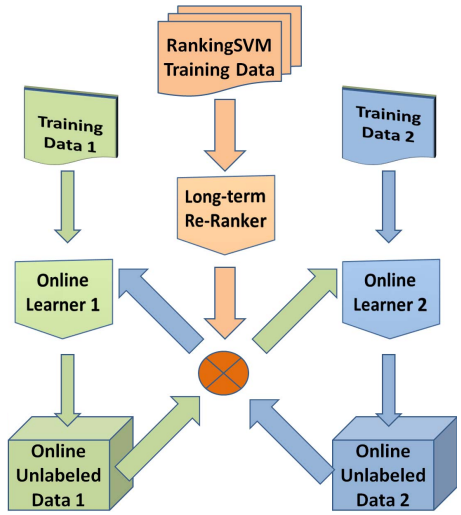


Fig. 6. Co-train procedure to train the two different online association learners.

The basic assumption in [31] is that two views are sufficient and independent. Other studies extend the basic co-train learning algorithm, so that they do not need to satisfy these assumptions strictly any more [34], [35]. Empirical results also show the co-train process works well even when the assumptions do not hold strictly. Zhou and Li [35] show co-training with more than two weak learners can work well even the weak learners are completely different.

In our method, we combine the online association learners and long-term reranking into the co-train learning module, as shown in Fig. 6. Similar to [33], we utilize two independent feature sets to construct two online association learners. Like the reranking learners, these online learners are trained first using a set of offline extracted samples from frame-to-frame association ground truth. To fit the learners to the current concrete tracking context, we need to update the two association learners during online tracking. Although the reranker is not independent with these two online learners, we can still update the two association learners with co-train strategy using association data at current moment. To handle the noisy samples, similar to [35], the reranker acts as the third learner to mediate whether the sample can be believed or not. If the first learner agrees with the reranker, it is safe to use the sample to update the second learner, and vice versa. Besides, we set two constraints to choose the eligible associations samples. One constraint is the early stored association samples in short-term association memory have priority to be samples for online training. That is, the longer we keep the association in short-term memory, the more reliable it is. Another constraint is the successively selected association samples have priority used as training samples. Thus, we choose this association sample in short-term memory which is either kept longest or selected frequently by reranker as the current sample for co-train learning at each round.

V. Algorithm

We have discussed about multiple short-term associations reranking based on the features of consistency with previous long-term associated tracks. Although there are some similarities between multitarget tracking based on detection association and single object tracking using the mechanism of tracking-by-detection, there are large differences between them. Two of differences are how to initialize one tracked object and how to terminate its tracking. While tracking single object always does not consider this problem, tracking multitarget based on data association must do. In this paper, we use the concept of life cycle to describe the stages from initiation to termination of all of tracked objects uniformly. Our life cycle model for tracking can be described in Fig. 7.

As in Fig. 7, we create a tracked object when it is associated for more than six frames continuously, otherwise too short associations are treated as false alarms. To decide whether a object has been out of camera field or not, we need to estimate each object's position. This mechanism works to not only handle the problem of short-term occlusions but also obtain redetecting spatial ranges for occluded objects. As usual, we calculate an estimated box location using Kalman filter for each object because it is concise and runs fast. When the redetecting spatial range is overlapped with next detections, the matching affinity is calculated, otherwise it is set to null.

Combining the life cycle variables and parameters for Kalman filter, we obtain candidate set of tracked object list U^t. This set is composed by all tracking candidates, each of which is a list of being tracked objects for matching. We describe one object using its representative detection $s_r(r_i^t)$ as in Section IV-A with its life cycle variables and tracking parameters. This candidate set buffer all our current information for different matching. The recent candidate associations are stored in short-term association candidate set A_s, which memorizes all recent possible best short-term associations. When long-term tracking has enough evidence to correct possible errors, these short-term candidate associations can provide the candidates of the next best answer.

Algorithm 1 summarizes our method. In the offline training process, the reranker using long-term associated trajectories is learned using artificial training data set, and the frame-to-frame matching samples are extracted from ground truth to train the two online association learners initially. In the online training process, the short-term association candidate set and tracked object list candidate set are updated at each frame. The best recent associations are found to correct potential errors and construct the overall tracked trajectories.

VI. Experiments

The main purpose of our method is to promote the online tracking performance by correcting some errors with long-term association reranking. Thus, it is important to design the stable and reliable reranking algorithm. The first part of our experiments verifies the long-term association features for reranking, and the performance of Ranking SVM training.

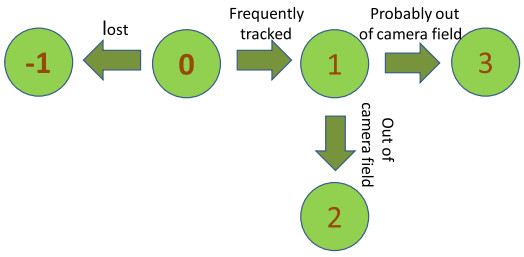


Fig. 7. Life cycle for one tracked object. When one detection response does not find matched tracked object, it becomes one candidate object with state 0. This candidate object may be treated as one false alarm with state -1 when it is lost for some frames or initialized as one true object with state 1 after being tracked frequently. A true object keeps its state until it is estimated as out of camera field with state 3 or really leaves the camera field as state 2.

To evaluate the performance of our multitarget tracking approach, we first utilize three public data sets to verify the performance: 1) PETS2009-S2-L1; 2) ETHMS; and 3) TUD. The sequences in these data sets contain different visual conditions, such as using static camera and moving camera, partial occlusion and full occlusion, pose variation, and illumination changing. More importantly, the detection results and tracking ground truth of these data sets are opened to public. Thus, we can compare with other methods fairly. Then, we compare with other state-of-the-art methods in the other two public data sets: 1) CAVIAR and 2) AVSS iLIDS. These data sets are captured from the real scenarios corridors of one mall and one subway station. We illustrate the performance of our method in these real-world situations.

We set the hyper-parameters C_1 and C_2 in (4) and (17) as 0.1 and 50.0, and estimate other parameters related with specific distribution by training data set, as described below.

A. Evaluation of Reranking Features for Multiframe Associations

In the offline training process, we learn the weights α for association features $\Psi(R_{[t_i-\Delta t-\Delta T:t_i]}, S_{[t_i-\Delta t-\Delta T:t_i]}, y_{[t-\Delta t:t]})$. We employ the structured output RankingSVM algorithm. First, we need to construct the training data set Q which is composed by multiframe association lists $\{(y^{q0}_{[t-\Delta t:t]}, y^{q1}_{[t-\Delta t:t]}, \ldots, y^{q10}_{[t-\Delta t:t]})\}_{q=1}^{|Q|}$. In each frame range $[t-\Delta t:t]$, we extract a list of different associations $y^{GT}_{[t-\Delta t:t]}$, $y^{q_i}_{[t-\Delta t:t]}$, $i=1,\ldots,10$, where the first association $y^{GT}_{[t:t+\Delta t]}$ is same as ground truth, and the other 10 associations are obtained by three different transformation operations to $y^{GT}_{[t:t+\Delta t]}$: switching some detection segments between two tracks, drifting some detections for some tracks, or adding some faked tracks into the association. Using these operations, the constructed associations have such relationship $y^{GT}_{[t:t+\Delta t]} \succ y^{q_1}_{[t:t+\Delta t]} \succ \cdots \succ y^{q_10}_{[t:t+\Delta t]}$ that we can construct different association lists as samples. Thus, the size of training data set is 10 times of the number of temporal ranges which we can separate training data set into. Because the scenarios in PETS2009 and ETHMS are extremely different, we construct 4680 and 3650 samples for

Algorithm 1 Online Multitarget Tracking Based on Multiple Short-Term Associations Reranking

Input:
Candidate set of tracked object list, U^0;
Short-term association candidate set, A_s;
Long-term trajectory set, T_r;
Output:
current trajectories set, MTT;

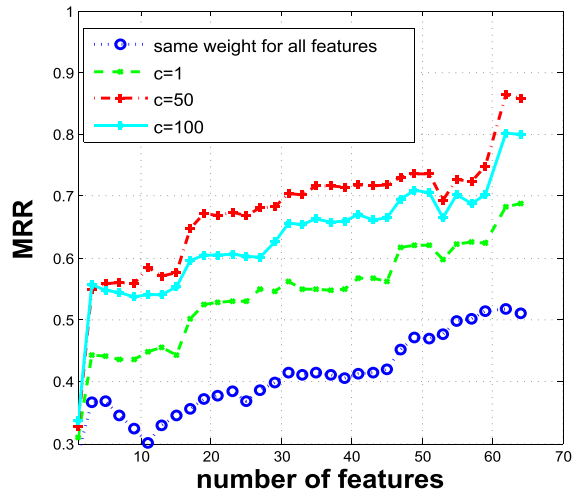
Learning RankingSVM and two online association learners:
Extract the training set with long-term association trajectories and their associations, and train the re-ranker using RankingSVM.
Extract the frame-to-frame training data, and learn the two online association learners using two type of features.
Online Association at frame $t(t = 1...N)$**:**
In each frame t, collect the detection response set as S^t, set $U^t = []$.
for each candidate object list u in U^{t-1}. **do**
  Get non-replicated matching results using all online association learners.
  for each matching result **do**
    Find matched objects R^{t-1} in u, and update their life states, estimated boxes.
    Add the matching to the short-term association set A_s.
    Generate candidate objects with unmatched detections, and add to u.
    Add u to U^t.
  end for
end for
Find the best short-term associations from A_s using re-ranker considering T_r.
Find the eligible associations sample in A_s, and delete it from A_s.
Update T_r using association sample which it is reliable enough.
According to the feature type of the association sample, update the association learners with co-train process.
Generate the current trajectory list MTT according to the best short-term associations and previous trajectory list T_r.

these two data sets, respectively, and verify the reranking results by cross validation between them.

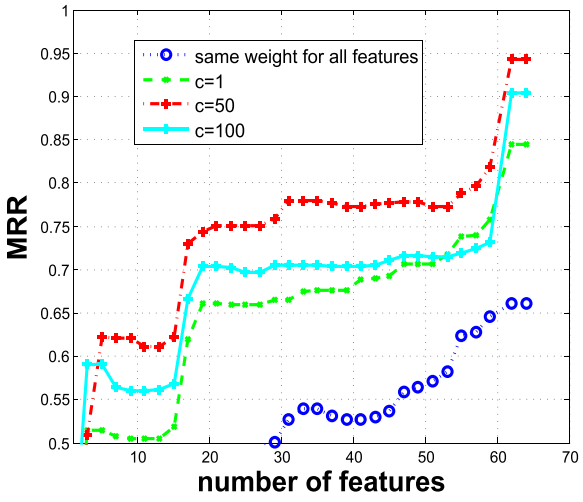
To evaluate the quality of the learned weights, we utilize one common statistic metric for ranking, mean reciprocal rank (MRR) [36], to measure how well the weighted features discriminate the ranking for positive samples in the validation set. The MRR is defined as the mean of reciprocal ranks for each positive sample in its corresponding sample list [36]

$$\mathrm{MRR} = \frac{1}{|Q|}\sum_{q\in Q}\frac{1}{\mathrm{rank}_q} \tag{18}$$

where rank_q is the number of rank order for the only true association in each sample association list. Different from mean average precision and normalized discounted cumulative gain [36], it reflects the order of the only true instance being queried. In this paper, it evaluates the rank of the only true association in each sample list.



(a)



(b)

Fig. 8. MRR curves for the different number of features. (a) Test results on PETS2009 based on the ETHMS as training data set. (b) Test results on ETHMS based on PETS2009 as training data set.

To check the different values for tradeoff parameter C_2, we get four results for two group tests. The four results are composed using three different values 1, 50, and 100 for C_2 and all same weights for α. The two group tests are running to train the RankingSVM using PETS2009, or ETHMS as training data sets with another as validation data sets, respectively. This cross-validation-like methods can verify the appropriate values for hyper parameters and show the robustness of our designed features in different scenarios. In each group experiments, we obtain the MRR curves on validation sets using different number of features, thus we can prove that our designed features are effective to rank the higher prioritized association to the lower one. Fig. 8(a) and (b) gives experimental results on PETS2009 using weights trained on ETHMS data sets and experimental results on ETHMS data sets using PETS2009 as the training data set, respectively.

In our experiments, the correctness of the ranking results using training data sets ETHMS and PETS2009 arrives nearly 0.95 and 0.85. These metrics do not drop obviously when the weights are verified on the cross-validation sets PETS2009 and ETHMS, respectively, as in Fig. 8(a) and (b). This shows the robustness of our features to the scenario changes since ETHMS data set has the parallel viewpoint from the pedestrian and moving camera, while the PETS2009 is taken from static camera on the high building with top viewpoint. Besides, our features appear insensitive to the hyper-parameters changes. The results are better and stable when the hyper parameter is set to 50, which is adopted in our next experiments. However, the number of features employed in our experiments affects the results manifestly. When we combine all features together, the performance arrives best, which proves the effectiveness of these features.

B. Comparison With State-of-the-Art Methods

Our method aims to combine the reranking multicandidate associations into online tracking strategy. As discuss above, we need the cross-validate strategy to evaluate our method using different training sets for reranking learning. We call our method as reranking-based online multitarget tracking (ReRankingCoMT). Specifically, the methods based on training data sets, ETHMS, PETS09, and TUD, are named ReRankingCoMT-E, ReRankingCoMT-P, and ReRankingCoMT-T, respectively. To evaluate the performance of our proposed method and show the advantage of this combining strategy, we need to compare with the methods using online strategy and offline optimization. Using the common ground truth and detection response input, we compare with five state-of-the-art methods. The first two methods are recognition-based tracking (PIRMPT) [12] and SSVM-based method (SSVMMOT) [6]. Similar with our methods, these two algorithms combine the offline learned features to online tracking process. The last three methods are energy-based algorithm (EnergyMIN) [28], the method considering exclusion between detections and trajectories (ExcTracking) [4] and the online CRF-based method (OnlineCRF) [3]. These three methods construct different CRFs with different constraints like the trajectory smoothness, the exclusion between trajectories, and the difference between closed objects. Thus, the global optimizations are executed to finding the best association results. It is convincible for our proposed method to compare with these five methods both in aspect of offline feature learning and long-term association optimization.

To evaluate the quantitative performance, we employ the VACE metrics [3]. These metrics are mainly composed by detection recall (RECALL), detection precision (PREC), the percentage of the mostly tracked (MT) objects, the percentage of the partial tracked (PT) objects, the percentage of the mostly lost (ML) objects, the number of trajectories' interruption by tracking (Frag), and the number of real identities' changes for tracked trajectory (IDS). Because ML is redundant with MT and PT, we omit this item. These metrics can be calculated using public tools [3]. Moreover, we use the harmonic mean (F) of RECALL and PRECISION to reflect the overall metric.

TABLE II
COMPARISON WITH STATE-OF-THE-ART METHODS

Datasets	METHOD	RECALL	PREC	F	MT	PT	Frag	IDS
PETS-2009-S2-L1	PIRMPT[12]	89.5%	99.6%	0.94	78.9%	21.1%	23	1
	SSVMMOT[6]	97.2%	93.7%	0.95	94.7%	5.3%	19	4
	OnlineCRF[3]	93.0%	95.3%	0.94	89.5%	10.5%	13	0
	ExcTracking[4]	—	—	—	94.7%	5.3%	15	22
	EnergyMIN[28]	92.4%	98.4%	0.95	91.3%	4.3%	6	11
	ReRankingCoMT-E	99.1%	97.7%	0.98	100.0%	0.0%	3	2
	ReRankingCoMT-T	99.0%	97.7%	0.98	100.0%	0.0%	5	3
TUD Stadt-mitte	PIRMPT[12]	81.0%	99.5%	0.89	60.0%	30.0%	0	1
	SSVMMOT[6]	80.0%	96.7%	0.88	80.0%	20.0%	11	0
	OnlineCRF[3]	87.0%	96.7%	0.92	70.0%	30.0%	1	0
	ExcTracking[4]	—	—	—	40.0%	60.0%	13	15
	EnergyMIN[28]	84.7%	86.7%	0.86	77.8%	22.2%	3	4
	ReRankingCoMT-E	88.5%	99.5%	0.94	80.0%	20.0%	2	2
	ReRankingCoMT-P	91.0%	94.5%	0.93	80.0%	20.0%	3	2
ETHMS	PIRMPT[12]	76.8%	86.6%	0.81	58.4%	33.6%	23	11
	SSVMMOT[6]	78.4%	84.1%	0.81	62.7%	29.6%	72	5
	OnlineCRF[3]	79.0%	90.4%	0.84	68.0%	24.8%	19	11
	ExcTracking[4]	77.3%	87.2%	0.82	66.4%	25.4%	69	57
	ReRankingCoMT-P	82.3%	83.6%	0.83	70.2%	24.5%	34	42
	ReRankingCoMT-T	81.7%	84.7%	0.83	68.1%	25.5%	45	56

Table II gives the results of comparison. By comparison with the similar online algorithms, PIRMPT and SSVMMOT, which utilized offline learned features in the online tracking manner. Our method exceeds them both in recall and integrity of the trajectories. Most true detections are linked to our tracked results and trajectories are generated in more complete. By comparison with global optimization methods, EnergyMIN, ExcTracking, and OnlineCRF, our approach shows competitive results. In the scenario of the static camera, such as in PETS2009, our approach even outperforms these global optimizing algorithms. There are less identity switches and false-alarming tracking fragments. As proved in the above discussion, these attribute to the stable distinguishability in most ambiguous cases. Besides, different from these global optimizing methods which all induce long-time delay when used for online tracking applications, our method does not have this problem.

Fig. 9 shows some tracking examples. The first row illustrates some results from the static camera as in PETS2009-S2-L1 sequence. There are interactions and short term partial occlusions, such as persons 1 (green box) and 2 (blue box) in the frames from 13 to 65. They are meeting and crossing each other. They are occluded each other nearly complete at frame 26 until frame 65 the third person walks around them. The appearance features at 65 is more discriminative than that in frame 26, thus the reranking works at this situation to correct the ambiguities at frame 26 and obtain a correct tracking results for them, so as for frame 733, in which three persons cross each other. Our method appears robust for these partial occlusions, even in the long-time occlusion case for person 3 in the 65th frame. The second and third rows illustrate the examples when camera is parallel with view field in TUD and ETHMS sequences. There are many inaccurate detection results in TUD sequence, and person 3, person 2, and person 8 pass behind others causing some full occlusions. Our method is

Fig. 9. Tracking examples for PETS09, TUD, and ETHMS bahnnof from top to bottom row.

TABLE III
EXPERIMENTAL RESULTS ON CAVIAR

METHOD	RECALL	MT	PT	Frag	IDS
Ours	90.2%	87.5%	11.1%	14	9
Zhang et al.[1]	76.4%	85.7%	10.7%	20	15
Huang et al.[2]	86.3%	78.3%	14.7%	54	12
Li et al. [37]	89.0%	84.6%	14.0%	17	11
Kuo et al.[38]	89.4%	84.6%	14.7%	18	11
Kuo et al. [12]	88.1%	86.0%	13.3%	17	4

robust to handle these occlusions, when they appear again, except the ID of 4 is hijacked at frame 85 near the border of the right side, which causes one ID switch.

By contrast with TUD, the sequence of ETHMS is taken from moving camera, and there are lots of full occlusions when the persons pass by. Our method can handle them in most cases like in frame 108 to frame 305, where the person 11 pass behind person 7, 8, and 10 until disappears at the behind of person 17.

C. Experiments on Other Data Sets

We utilize the other two data sets, CAVIAR and AVSS iLIDS, to verify the performance of our proposed method further in this section.

The CAVIAR data set utilized in our experiment is composed of 26 clips taken from corridor view. The resolution of the sequences is 384 × 288 pixels, and contains about 36 296 frames and 193 trajectories in all. The AVSS iLIDS AB data set contains three sequences with resolution of 720 × 576 pixels. The three sequences, named EASY, MEDIUM, and HARD, are about 14 863 frames in length and contain about 130 ground-truth trajectories totally. These data sets are all captured from the real-world surveillance systems, and have more complex scenarios than the previous three data sets.

First, we evaluate the performance using CAVIAR data set with the same metrics VACE as in the previous section. We compare with five state-of-the-art methods: the methods in [1], [2], [12], [37], and [38]. Table III shows the experimental results. Our method is more liable to obtain complete trajectories compared with other methods.

As shown in Fig. 10, our method is robust to occlusions. The long-term reranking is helpful to discriminate the true association from the ambiguity of occlusion and overlapping between objects as from frame 75 to frame 139 in Fig. 10(a) or frame 167 to frame 270 in Fig. 10(b). However, these occlusion cases cannot be handled well without the support of reranking evidence as in Fig. 10(c) in which the template of occluded objects are disturbed by the front past persons. Thus, the erroneous colliding trajectories are constructed. These colliding trajectories can be corrected when we considering the long-term reranking features, such as motional consistency, as expressed in (8)–(10).

Second, we utilize the iLIDS data set to verify our methods. We use the metrics CLEAR MOT [39] to compare with [17], [40], and [41].

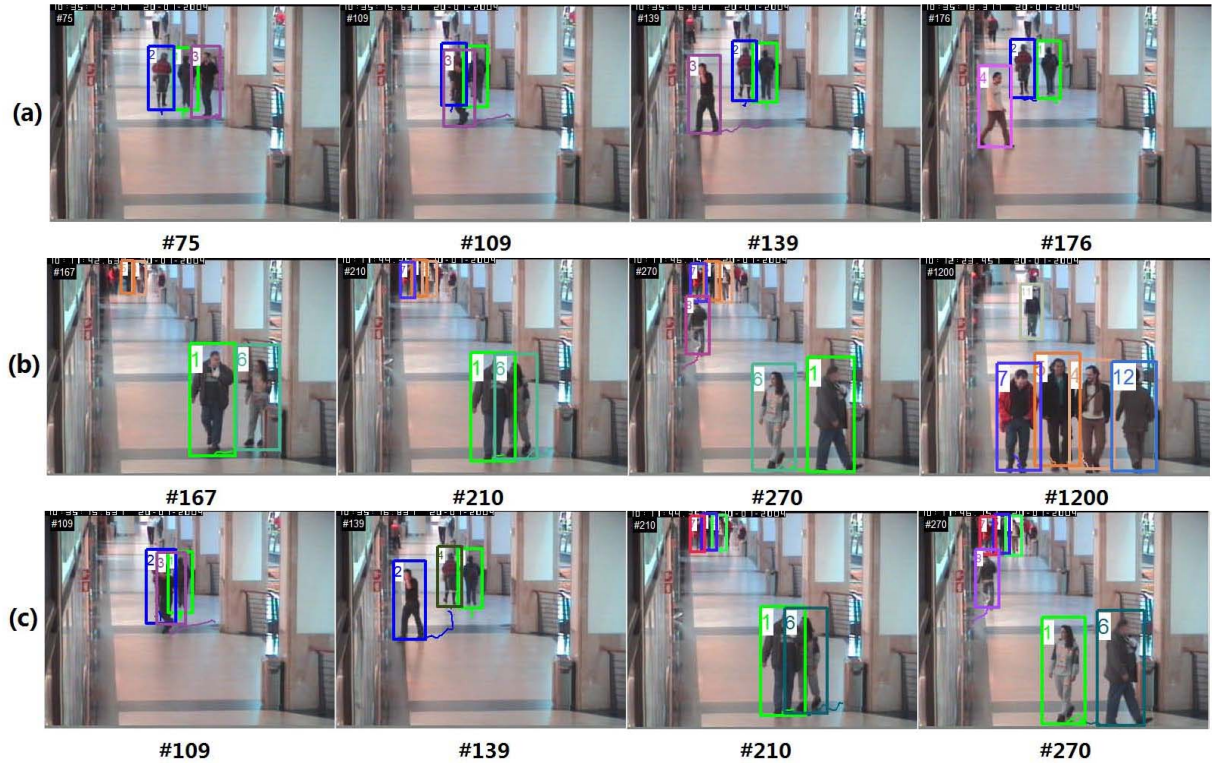


Fig. 10. Tracking examples on two of CAVIAR sequences. (a) *EnterExitCrossingPaths1cor*. (b) *ThreePastShop1cor*. (c) Occlusions are not handled well if we only obtain the online association results without considering reranking.

Fig. 11. Tracking examples on three iLIDS sequences. (a) iLIDS easy. (b) iLiDS medium. (c) iLIDS hard.

The CLEAR MOT [39] contains two intuitive metrics for tracking results named MOTP and MOTA. The former shows the tracking precision independent on the assignments, and the latter shows the rightness of mapping between the estimated trajectories and their ground truth. In each frame i, let GTD_i, be the number of ground truth detections, and let MIS_i, FTR_i,

TABLE IV
TRACKING RESULTS ON iLIDS DATA SET

METHOD	MOTP	MOTA	REC	PRC
Ours	79.1%	77.8%	83.7%	94.5%
Huang et al.[17]	—	68.4%	71.0%	86.3%
Brendel et al.[40]	70.0%	78.6%	80.6%	98.5%
Benfold et al.[41]	73.6%	59.3%	82.0%	80.3%

TRD_i, and IDS_i be the number of missed tracked detections, falsely tracked detections, rightly tracked detections, and identity switches respectively. Let OVL_{ji} be the overlap ratio of detection j to its matched tracking result of frame i, then the multitargets tracking precision metric MOTP and tracking accuracy MOTA can be defined as

$$\mathrm{MOTP} = \frac{\sum_{i,j} \mathrm{OVL}_{ji}}{\sum_i \mathrm{TRD}_i} \tag{19}$$

$$\mathrm{MOTA} = 1 - \frac{\sum_i (\mathrm{MIS}_i + \mathrm{FTR}_i + \mathrm{IDS}_i)}{\sum_i \mathrm{GTD}_i} \tag{20}$$

where the first two elements are missing ratio per frame, and false-alarming ratio per frame. Here, we employ the other two metrics with REC and PRC as the following definition:

$$\mathrm{REC} = 1 - \frac{\sum_i \mathrm{MIS}_i}{\sum_i \mathrm{GTD}_i} \tag{21}$$

$$\mathrm{PRC} = 1 - \frac{\sum_i \mathrm{FTR}_i}{\sum_i \mathrm{GTD}_i}. \tag{22}$$

Table IV gives our tracking results and comparison between the other three methods. Our method has competitive performance.

Fig. 11 shows some tracking examples using our proposed method. It can be seen our method is robust to the short-term full occlusions due to a pillar at the right side as in Fig. 11(a). This is beneficial from the prediction of Kalman filter during the life-cycle for each object. Our method is also robust for partial occlusions such as in frame 400 to frame 433 as in Fig. 11(b) and in frame 179 to frame 284 as in Fig. 11(c), which is due to the online co-train process based on multiparts features, although some ID switches exist in crowded situations when detections are not accurate anymore such as in frame 400 as in Fig. 11(b).

We implement our code using MATLAB 2013 with Intel i7-3632QM platform, which is equipped with 2.2-GHz CPU and 4G ROM. Our tracking system runs at 2.1 frames/s. This speed is not fast due to two reasons.

1) The code is not optimized, the multibest associations are calculated with some redundant codes, which can be compressed.
2) There are many feature recalculations, which take most of times, whereas the online co-train process is fast due to the closed form of updating formulation (5).

VII. CONCLUSION

In this paper, we propose a new online tracking framework. Our work generates the short-term multibest associations based on multiple association learners. A long-term association-based reranking is trained to discriminate the best one from these associations, by which the multiple online association learners are update with co-train process. The experimental results show the discrimination of the weighted long-term reranking features by offline training. By comparison with more than five state-of-the-art algorithms in five public data sets using common evaluation metrics, our method outperforms the other online tracking algorithms and is competitive with global optimal ones.

Our work aims to correct the errors in determinate online association with help of the long-term features. This idea is consistent with human vision system. Besides, it can be extended to promote the performance of the other online tracking algorithms. In the future work, we can try to promote the performance by adding more discriminative features and covering the true association as much as possible.

REFERENCES

[1] L. Zhang, Y. Li, and R. Nevatia, "Global data association for multi-object tracking using network flows," in *Proc. IEEE Conf. CVPR*, Jun. 2008, pp. 1–8.
[2] C. Huang, B. Wu, and R. Nevatia, "Robust object tracking by hierarchical association of detection responses," in *Proc. 10th ECCV*, 2008, pp. 788–801.
[3] B. Yang and R. Nevatia, "Multi-target tracking by online learning a CRF model of appearance and motion patterns," *Int. J. Comput. Vis.*, vol. 107, no. 2, pp. 203–217, 2014.
[4] A. Milan, K. Schindler, and S. Roth, "Detection- and trajectory-level exclusion in multiple object tracking," in *Proc. IEEE Conf. CVPR*, Jun. 2013, pp. 3682–3689.
[5] X. Yan, X. Wu, I. A. Kakadiaris, and S. K. Shah, "To track or to detect? An ensemble framework for optimal selection," in *Proc. 12th ECCV*, 2012, pp. 594–607.
[6] S. Kim, S. Kwak, J. Feyereisl, and B. Han, "Online multi-target tracking by large margin structured learning," in *Proc. 11th ACCV*, 2012, pp. 98–111.
[7] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Autom. Control*, vol. 24, no. 6, pp. 843–854, Dec. 1979.
[8] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*. San Diego, CA, USA: Academic, 1988.
[9] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137–154, 2004.
[10] Y. Li, H. Ai, T. Yamashita, S. Lao, and M. Kawade, "Tracking in low frame rate video: A cascade particle filter with discriminative observers of different life spans," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 10, pp. 1728–1740, Oct. 2008.
[11] S. Stalder, H. Grabner, and L. Van Gool, "Cascaded confidence filtering for improved tracking-by-detection," in *Proc. 11th ECCV*, 2010, pp. 369–382.
[12] C.-H. Kuo and R. Nevatia, "How does person identity recognition help multi-person tracking?" in *Proc. IEEE Conf. CVPR*, Jun. 2011, pp. 1217–1224.
[13] N. Wang and D.-Y. Yeung, "Learning a deep compact image representation for visual tracking," in *Advances in Neural Information Processing Systems 26*. Red Hook, NY, USA: Curran Associates, 2013.
[14] M. Fromer and A. Globerson, "An LP view of the M-best MAP problem," in *Advances in Neural Information Processing Systems 22*. Red Hook, NY, USA: Curran Associates, 2009.
[15] C. Yanover and Y. Weiss, "Finding the M most probable configurations using loopy belief propagation," in *Advances in Neural Information Processing Systems 16*. Cambridge, MA, USA: MIT Press, 2004.
[16] D. Batra, P. Yadollahpour, A. Guzman-Rivera, and G. Shakhnarovich, "Diverse M-best solutions in Markov random fields," in *Proc. 12th ECCV*, 2012, pp. 1–16.
[17] D. Park and D. Ramanan, "N-best maximal decoders for part models," in *Proc. IEEE ICCV*, Nov. 2011, pp. 2627–2634.
[18] M. Collins, "Discriminative reranking for natural language parsing," in *Proc. 17th ICML*, 2000, pp. 175–182.
[19] T.-Y. Liu, "Learning to rank for information retrieval," *Found. Trends Inf. Retrieval*, vol. 3, no. 3, pp. 225–331, 2009.

[20] B. Yang, C. Huang, and R. Nevatia, "Learning affinities and dependencies for multi-target tracking using a CRF model," in *Proc. IEEE Conf. CVPR*, Jun. 2011, pp. 1233–1240.
[21] Y. Bai and M. Tang, "Robust tracking via weakly supervised ranking SVM," in *Proc. IEEE Conf. CVPR*, Jun. 2012, pp. 1854–1861.
[22] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer, "Online passive-aggressive algorithms," *J. Mach. Learn. Res.*, vol. 7, pp. 551–585, Mar. 2006.
[23] C. Huang and R. Nevatia, "High performance object detection by collaborative learning of joint ranking of granules features," in *Proc. IEEE Conf. CVPR*, Jun. 2010, pp. 41–48.
[24] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.
[25] B. Song, T.-Y. Jeng, E. Staudt, and A. K. Roy-Chowdhury, "A stochastic graph evolution framework for robust multi-target tracking," in *Proc. 11th ECCV*, 2010, pp. 605–619.
[26] T. Joachims, "Training linear SVMs in linear time," in *Proc. 12th Int. Conf. KDD*, 2006, pp. 217–226.
[27] T. Joachims, T. Finley, and C.-N. J. Yu, "Cutting-plane training of structural SVMs," *Mach. Learn.*, vol. 77, no. 1, pp. 27–59, 2009.
[28] A. Milan, S. Roth, and K. Schindler, "Continuous energy minimization for multitarget tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 1, pp. 58–72, Jan. 2014.
[29] (2014). *Bo Yang's Homepage*. [Online]. Available: http://iris.usc.edu/people/yangbo/downloads.html
[30] E. Averbach and G. Sperling, "Short term storage of information in vision," in *Information Theory*, C. Cherry, Ed. Washington, DC, USA: Butterworth, 1961, pp. 196–211.
[31] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proc. 11th Annu. Conf. Workshop Comput. Learn. Theory (COLT)*, 1998, pp. 92–100.
[32] O. Javed, S. Ali, and M. Shah, "Online detection and classification of moving objects using progressively improving detectors," in *Proc. IEEE Conf. CVPR*, vol. 1. Jun. 2005, pp. 696–701.
[33] F. Tang, S. Brennan, Q. Zhao, and H. Tao, "Co-tracking using semi-supervised support vector machines," in *Proc. IEEE 11th ICCV*, Oct. 2007, pp. 1–8.
[34] S. A. Goldman and Y. Zhou, "Enhancing supervised learning with unlabeled data," in *Proc. 17th ICML*, 2000, pp. 327–334.
[35] Z.-H. Zhou and M. Li, "Tri-training: Exploiting unlabeled data using three classifiers," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 11, pp. 1529–1541, Nov. 2005.
[36] D. R. Radev, H. Qi, H. Wu, and W. Fan, "Evaluating Web-based question answering systems," in *Proc. LREC*, 2002, pp. 1–4.
[37] Y. Li, C. Huang, and R. Nevatia, "Learning to associate: HybridBoosted multi-target tracker for crowded scene," in *Proc. IEEE Conf. CVPR*, Jun. 2009, pp. 2953–2960.
[38] C.-H. Kuo, C. Huang, and R. Nevatia, "Multi-target tracking by on-line learned discriminative appearance models," in *Proc. IEEE Conf. CVPR*, Jun. 2010, pp. 685–692.
[39] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The CLEAR MOT metrics," *J. Image Video Process.*, vol. 2008, pp. 1–10, Feb. 2008.
[40] W. Brendel, M. Amer, and S. Todorovic, "Multiobject tracking as maximum weight independent set," in *Proc. IEEE Conf. CVPR*, Jun. 2011, pp. 1273–1280.
[41] B. Benfold and I. Reid, "Stable multi-target tracking in real-time surveillance video," in *Proc. IEEE Conf. CVPR*, Jun. 2011, pp. 3457–3464.

Yingkun Xu received the master's degree in computer science and technology from Nanjing University, Nanjing, China, in 2005, and the Ph.D degree in technology of computer application from the Institute of Computing Technology (ICT), University of Chinese Academy of Sciences (UCAS) in 2015.

He is currently a faculty member with College of Computer Science and Technology, Zhejiang University of Technology. His research interests include machine learning, computer vision, and video technology.

Lei Qin received the B.S. and M.S. degrees in mathematics from Dalian University of Technology, Dalian, China, in 1999 and 2002, respectively, and the Ph.D. degree in computer science from Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2008.

He is an Associate Professor with Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences. He has authored or co-authored over 40 technical papers in the area of computer vision. His research interests include image/video processing, computer vision, and pattern recognition.

Dr. Qin is a Reviewer of IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and IEEE TRANSACTIONS ON CYBERNETICS. He has served as a TPC Member of various conferences, including the European Conference on Computer Vision, the International Conference on Pattern Recognition, the International Conference on Multimedia and Expo, the Pacific Rim Symposium on Image and Video Technology, the International Conference on Internet Multimedia Computing and Service, and the Pacific-Rim Conference on Multimedia.

Qingming Huang (SM'08) received the bachelor's degree in computer science and the Ph.D. degree in computer engineering from Harbin Institute of Technology, Harbin, China, in 1988 and 1994, respectively.

He is a Professor with the University of Chinese Academy of Sciences, Beijing, China, and an Adjunct Research Professor with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing. He has co-authored more than 300 academic papers in international journals, including IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON MULTIMEDIA, and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and top-level conferences, such as ACM Multimedia, International Conference on Computer Vision, Computer Vision and Pattern Recognition, and European Conference on Computer Vision. His current research areas include multimedia content analysis, image processing, computer vision, pattern recognition, and machine learning.

Dr. Huang served as the Program Chair and Organization Committee or TPC Member in various well-known international conferences, including ACM Multimedia, the International Conference on Computer Vision, the International Conference on Multimedia and Expo, and the Pacific Rim Symposium on Image and Video Technology.