Neurocomputing 168 (2015) 327–335

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Online learning affinity measure with CovBoost for multi-target tracking

Guorong Li [a,c], Qingming Huang [a,b,c], Shuqiang Jiang [b,*], Yingkun Xu [b], Weigang Zhang [d,*]

[a] University of Chinese Academy of Sciences (CAS), Beijing 100190, China
[b] Key Lab. of Intell. Info. Process., Inst. of Comput. Tech., CAS, Beijing 100080, China
[c] Key Laboratory of Big Data Mining and Knowledge Management, CAS, China
[d] School of Computer Science and Technology, Harbin Inst. of Tech., China

ARTICLE INFO

Article history:
Received 26 August 2014
Received in revised form
23 March 2015
Accepted 27 May 2015
Communicated by Ning Wang
Available online 15 June 2015

Keywords:
Multi-target tracking
Tracklet affinity
CovBoost
Online learning

ABSTRACT

In this paper, we propose a new online learning method for measuring affinity between tracklets in multi-target tracking. As targets and background usually keep changing in the video, fixed affinity measurement could not adapt to their variations. Most existing affinity learning methods construct labeled samples based on the obtained tracklets, and then minimize a predefined loss function to get an optimal affinity measurement. However, those methods simply assume that the training error equals to testing error which is not true in many of real time tracking scenarios. Differently, we propose to learn affinity measurement through CovBoosting, which considers the evolution of the tracklets and could obtain affinity measurement with more discriminative ability. To deal with targets' disappearance and new targets' appearance, we combine tracklet affinity with contextual information to do an optimal inference. Moreover, an online updating algorithm is developed to guarantee that the learned tracklet affinity is always optimal for tracking targets in current sliding window. Experimental results on benchmark datasets demonstrate that tracklet affinity learned with our method is more discriminative and could greatly improve the performance of the multi-target tracker.

1. Introduction

Multi-target tracking has been a hot research topic for decades, for its wide applications in computer vision fields such as traffic surveillance, human–computer interaction and 3D scene reconstruction. As objects' appearance and number change, occlusions exist and so on, inferring trajectory for each target is very challenging.

With the significant improvement in object detection, many association based multi-target tracking methods e.g. [1–3] are proposed and achieve good performance. Provided detection results, they try to link the results or tracklets belonging to the same object to form its trajectory. Therefore, determining whether two tracklets should be linked is very critical. In other words, how to measure the affinity or similarity between tracklets is one of the critical problems in multi-target tracking. Some off-line methods such as [4,5] first define or learn affinity measurement offline, and then formalize multi-targets tracking into some classical problems such as maximizing network flow [6], linear programming [7] and so on. However, they assume that a fixed unified affinity measure could be used to measure the similarity between tracklets at different frames. Apparently, they neglect the variance of the targets. Generally, in a video, due to variations of illumination, background, pose and so on, the appearance of the targets can hardly remain the same. In the field of single object tracking, many online methods [8–10] have been proposed to learn appearance model for the tracked targets. Their superior performance demonstrates that adjusting appearance model and affinity measurement to adapt to target's changes are very useful for tracking.

Many online learning methods [12–16,31] are developed to learn affinity measurement or appearance model for targets in an online manner. However, those methods usually learn only with obtained training samples, neglecting that the tracklets to be analyzed in real-time tracking applications is different with them. Therefore, the learned affinity measure may be not effective for associating unlabeled tracklets. As shown in [10], in single object tracking, the distributions of the target and background in adjacent frames varies considerably. It is necessary to take this phenomenon into account and design transfer learning method to deal with it. We think this phenomenon also exists in multi-target tracking. Therefore when learning affinity measurement or appearance model, it is reasonable to assume that the distributions of obtained training samples and unlabeled samples are different. Besides the target's appearance, the motion pattern [18] is another important information for inferring target's position and linear motion model is the most popular used.

* Corresponding authors
E-mail addresses: sqjiang@jdl.ac.cn (S. Jiang), wgzhang@jdl.ac.cn (W. Zhang).

http://dx.doi.org/10.1016/j.neucom.2015.05.093
0925-2312/

As analyzed in many papers, such as [17,19–21], contextual information is very useful for predicting target's position.

Based on the above analysis, we propose a new online learning method to learn affinity measure, which is further combined with motion information, the constraints of scale changes and velocity variations. To deal with entry and disappearance of the targets, two special detections, referred as *Start* and *End* respectively, are introduced to denote target's entry and exit of the scene. The affinity between detections and them is estimated according to contextual information. The framework of our whole tracking method is shown in Fig. 1. First, like other association based methods, pedestrian detector is adopted to obtain the position of the objects. Simple tracking method is used to generate short tracklets, from which we generate many unlabeled tracklet pairs as well as a few labeled tracklet pairs. Meanwhile, with trajectory obtained in previous frames, labeled tracklet pairs are collected and used as auxiliary samples. Second, with semi-supervised CovBoosting method, appearance affinity measure between tracklets is learned. Considering motion smoothness, motion affinity between tracklets is computed and fused with appearance affinity. Finally, after determining similarity between obtained trajectory and new tracklets, multi-target tracking is formulated as an optimal matching problem. The main contributions of the proposed method are summarized as follows.

- An online appearance affinity learning method which considers differences between the distribution of the tracklets in different frames.
- Effective Gaussian models to describe the motion, size and velocity information of the trajectory of a target, which is used to measure the smoothness of two tracklets.
- An simple but effective weighted summarization approach for fusing above two similarities, in which the weights could be updated automatically.
- Contextual information is used to estimate the probability that a target exits or enters the view of the camera.

The rest of paper is organized as follows. Section 2 summarizes related works. Then in Section 3, the formalization of association based multi-target tracking is presented in detail. In Section 4, experimental results on several public datasets and analysis are provided. Finally, the conclusion and future work are given in Section 5.

2. Related works

Appearance, motion and contextual information are three kinds of information commonly used for tracking. First, simple fixed appearance model or template [19,11,17,20,29,6,7,12–14,16,18] is used, but it suffers from variations of targets and background and many methods [12–14,16,18] are proposed to learn adaptive appearance model or affinity measure online. For offline multi-target tracking, many methods first generate tracklets, and learn affinity measurement to give the tracklets that belong to the same trajectory higher affinity score. Then multi-target tracking is formalized as a global optimal association problem. Meanwhile, to better discriminate targets, some methods are proposed to online learn appearance models. They associate tracklets progressively to longer one. At each level, based on the obtained tracklets, online methods are used to learn better appearance models.

Besides appearance information, motion information [11,17–20] and contextual information [11,17,20,21] are usually used for association. Simple linear models are widely used in [14,20]. The affinity or link probability between tracklets is estimated according to how they satisfy the motion model. As targets will not move alone, they are affected by environments, other objects and so on. Contextual information is exploited to predict target's motion effectively. Meanwhile, to determine whether a detection is the start or end of a trajectory, the structure of the scene [32,33] or statistic motion information [33] is used to find sinks or sources or give one region a probability indicating the possibility of a target disappearing or appearing from it.

Different from the above methods, our approach is designed for online multi-target tracking, which can process video stream more timely. The existing online appearance model learning methods, which trained appearance model with samples in the whole video, need not consider targets' and background's variations in the streaming data. While in this paper, we assume that at time t, the trajectories of the targets before time t are already obtained and we aim to link new tracklets with those trajectories. Therefore, the samples we constructed based on trajectories may follow different distribution with that constructed with new tracklets. So we propose a semi-supervised learning method to learn discriminative appearance model. Then considering motion information, we estimate the final link affinity between trajectory and tracklets. Finally, we formulate multi-target tracking as an optimal matching problem and adopt KP [34] algorithm to solve it.

3. The approach of our association based online multi-target tracking

Provided the video, a pedestrian detector is first used to detect people. Then based on the detection results, we simply associate them to form short tracklets denoted by T_i. See Fig. 2 for example. In this process, simple object tracker such as [27,28] could be used to obtain those tracklets. After this, multi-target tracking becomes how to link those tracklets together to formulate the complete trajectory of each target.

3.1. Online appearance affinity learning for tracklets association

Before presenting our method, we first explain the important variables that are frequently used in the following paper. Table 1

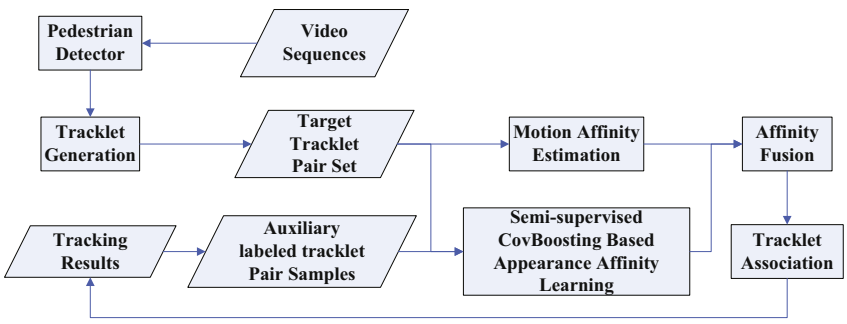


Fig. 1. The procedures of our multi-target tracking algorithm.

provides the detail illustration of those variables. Similar to other methods, our affinity measure mainly depends on motion, time and appearance information. Specifically, without loss of generality, we assume $T_i.Start \le T_j.Start$, then the affinity of two tracklets is learned by semi-supervised CovBoost [10].

3.2. Motion constraints or prediction for multi-target tracking

Motion is one of the important factors for inferring targets' states. Usually, in a short time, targets are assumed to move with uniform velocities [22]. Meanwhile, the velocities of the same target are not dependent with each other. In Fig. 3, we statistic the mean and variance of the speed at the previous frames and plot them in magenta, and the actual speed in this frame is plotted in blue. We can see that generally the speed in current frame would not deviate from the statistic mean of the speed in previous frames so much. Therefore, we could assume that the velocity sequences follow Gaussian distribution and update its mean and variation timely. Similarly, we also model the scale changes as a Gaussian Distribution sequence.

Similarity based on motion information: We predict the tracked target's (T_i) position assuming he moves at a constant velocity. As the $T_i.Velocity$ and $T_j.Velocity$ may be different, we use their mean to estimate the target's position at time $T_j.Start$. The affinity based on motion information is computed according to the following equation.

$$M_{i,j} = \exp\left\{-\frac{\| T_j.SPos - T_i.EPos - V_{i,j} * (T_j.End - T_i.Start)) \|^2}{\rho^2}\right\} \quad (1)$$

where $V_{i,j} = (0.5(T_i.Velocity + T_j.Velocity)$, ρ is the variance describing the error of our motion predictor. Apparently, if the start position is consistent with our prediction, the similarity would be high. According to the above analysis, velocity affinity is modeled as Eq. (2).

$$V_{i,j} = N(v, T_i.MVelocity; T_i.VVelocity) N(v, T_j.MVelocity; T_j.VVelocity) \quad (2)$$

Besides, the size of the same target should not varies quite differently and could be used for measuring affinity. Fig. 4 shows an example. Generally, scale of the target in the adjacent frames would not vary significantly. So we define scale affinity as Eq. (3).

$$S_{i,j} = N(T_i.EScale - Scale(i,j), T_i.MeanScale; T_i.VarScale)$$
$$N(Scale(i,j) - T_j.SScale, T_j.MeanScale; T_j.VarScale)$$
$$\exp(-KL(N(T_i.MeanScale; T_i.VarScale), N(T_j.MeanScale; T_j.VarScale))) \quad (3)$$

where $Scale(i,j)$ denotes average scale variation between the last detection of T_i and the first detection of T_j.

Appearance similarity learning: Usually, the appearance of the target and background vary continually. Therefore, their data distribution does not remain the same [10]. However, most existing multi-target tracking methods [13] neglect the distribution variations when learning discriminative appearance models for targets. Similar to [10], we assume that the data distribution in previous frames and that in the current frame follow "Covariate Shift". Then we choose CovBoost to learn the appearance similarity. We use $< D_i, D_j, y_{i,j} >$ to represent the sample, where $y_{i,j} = 1$ means detection result D_i and D_j belong to the same target and $y_{i,j} = -1$ means they are detection results of different object. Then

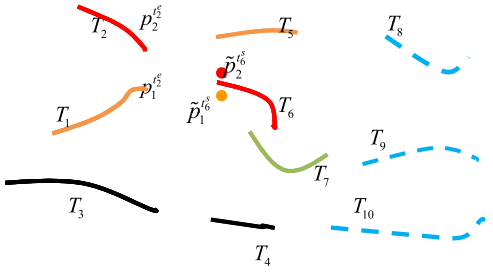


Fig. 2. Illustration for association based multi-target tracking.

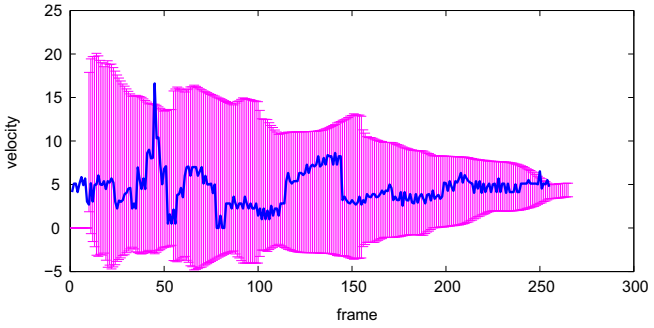


Fig. 3. Mean and variations of the velocity in the previous frames. Blue line denotes the velocity in every frame. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

Table 1
The illustrations for some variables.

Variable	Explanation
T_i	The ith tracklet
$T_i.Start$	The starting frame of T_i
$T_i.End$	The ending frame of T_i
$T_i.MeanVelocityDiff$	Mean of the difference between velocities in adjacent detections
$T_i.VarVelocityDiff$	Variance of the difference between velocities in adjacent detections
$T_i.MeanScale$	Mean of the scale changes between two adjacent detections
$T_i.VarScale$	Variance of the scale changes
$T_i.MeanMotionPreError$	Mean of the error of motion prediction model
$T_i.VarMotionPreError$	Variance of the error of motion prediction model
$T_i.NumDection$	The number of detections belonging to this tracklet
$T_i.MVelocity$	The average velocity of the target along with T_i
$T_i.VVelocity$	The variance of the velocity alongwith T_i
$T_i.SPos$	The starting position of T_i
$T_i.EPos$	The ending position of T_i
$T_i.SScale$	The scale change in the first two frames of T_i
$T_i.EScale$	The scale change in the last two frames of T_i
$T_i.Cur$	The curvature of T_i

let χ_a, χ_t and U denote auxiliary labeled samples, target labeled samples and unlabeled samples respectively. Since we assume that auxiliary samples and target samples follow different distribution, auxiliary samples are first shifted to target sample space through re-weight as shown in the first two images in Fig. 5. Similar to loss function of Adaboost, we hope our classifier (denoted by $H(\cdot)$) can classify the labeled samples correctly. For example, in the right image in Fig. 5, example x_1^+ is a positive sample, we hope $H(x_1^+)$ equal to 1. The first two items reflect penalty on inconsistency between the classification results of labeled samples and their labels. Meanwhile, we consider data consistency which is proved to be very useful in many semi-supervised learning method. That is if the observations of two samples are similar, their labels should be similar. Taken u_1,u_3 for example, u_1 is close to u_3, therefore we hope $H(u_1)$ is close to $H(u_3)$. Besides, u_3 is very similar to x_1^+, so we hope $H(u_3)$ is closer to 1. The last three items in Eq. (4) reflect penalty on data consistency. Through minimizing the loss function defined by Eq. (4), we can obtain a strong classifier $H(\cdot)$, which is the weighted sum of weak classifiers $h_{s_m}(\cdot)$.

$$L(H,\chi_a,\chi_t,U)=\underbrace{\sum_{\langle D_i,D_j,y_{ij}\rangle\in\chi_a}\frac{p_t(D_i,D_j,y_{ij})}{p_a(D_i,D_j,y_{ij})}\exp\{-y_{ij}H(D_i,D_j)\}}_{\text{loss on the auxiliary data}}$$
$$+\underbrace{\sum_{\langle D_i,D_j,y_{ij}\rangle\in\chi_t}\exp\{-y_{ij}H(D_i,D_j)\}}_{\text{loss on the target data}}$$
$$+\underbrace{\sum_{\substack{\langle D_i,D_j,y_{ij}\rangle\in\chi_a\\ \langle D_p,D_q\rangle\in U}}\frac{p_t(D_i,D_j,y_{ij})}{p_a(D_i,D_j,y_{ij})}S(i,j,p,q)\exp\{-y_{ij}H(D_p,D_q)\}}_{\text{loss measuring data consistency between auxiliary data and unlabeled data}}$$
$$+\underbrace{\sum_{\substack{\langle D_i,D_j,y_{ij}\rangle\in\chi_t\\ \langle D_p,D_q\rangle\in U}}S(i,j,p,q)\exp\{-y_{ij}H(D_p,D_q)\}}_{\text{loss measuring data consistency between target data and unlabeled data}}$$
$$+\underbrace{\sum_{\substack{\langle D_i,D_j\rangle\in U\\ \langle D_p,D_q\rangle\in U}}S(i,j,p,q)\exp\{-H(D_i,D_j)H(D_p,D_q)\}}_{\text{loss measuring data consistency between unlabeled data}}\quad(4)$$

where $S(i,j,p,q)$ denotes the similarity between $\langle D_i,D_j\rangle$ and $\langle D_p,D_1\rangle$. It is easily to obtain the solution according to our previous work [10].

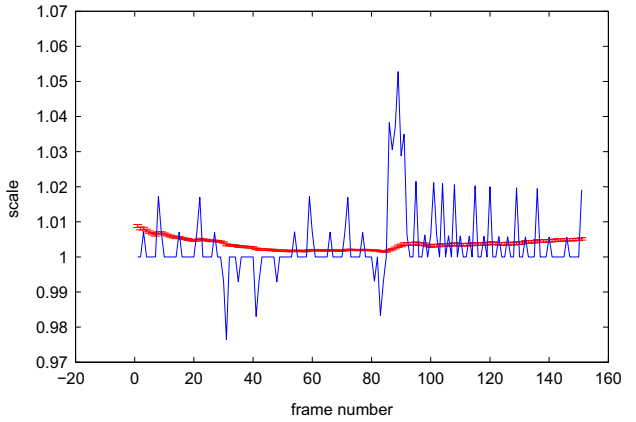


Fig. 4. Mean and variations of the scale in the previous frames. Blue line denotes the scale in every frame. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

3.3. Formulation of online affinity learning for multi-target tracking

We first associate obtained tracklets into trajectories. As time going, we obtained new tracklets, and need to associate them with trajectories as shown in Fig. 6. Let $L=l_1,l_2,\ldots,l_m$ represent the obtained trajectory and $T_n,T_{n+1},\ldots,T_{n+r}$ denote the new tracklets to be associated. We use $R(l)$ to represent the detection results set that belong to trajectory (or tracklet) l.

First, we construct sample set. Auxiliary labeled samples are collected from L. The positive auxiliary sample set $\chi_a^+=R(l_i)\times R(l_i), i=1,2,\ldots,m$, while the negative auxiliary sample set $\chi_a^-=R(l_i)\times R(l_j), i<j$. The labeled target sample set is constructed based on new tracklets: $\chi_t^+=R(T_i)\times R(T_i)$ and $\chi_t^-=R(T_i)\times R(T_j), T_j\in N(T_i)$, where $N(T_i)=\{T_j\,|\,0<T_j.Start-T_i.End<15\&\|T_j.SPos-T_i.EPos\|<R_{thresh}\}$ denotes the set of confusing tracklets that are not associated with T_i but share the similar motion information. Then the unlabeled samples are $U=R(T_i)\times R(T_j), T_j\in N(T_i)$. Then we can learn $H(\cdot)$ to compute the appearance similarity and fuse it with motion similarity through multiplying them $A(l_i,T_j)=M(l_i,T_j)H(l_i,T_j)$. To deal with target's disappearance and appearance, we introduce two nodes: *Start* and *End*.

To evaluate the affinity between a detection and *Start* or *End*, we first discover some boarder regions (denoted by B) that are in four boarders of the frame and contain people detections. See Fig. 7(a) for an example. On the right side of the frame, the region

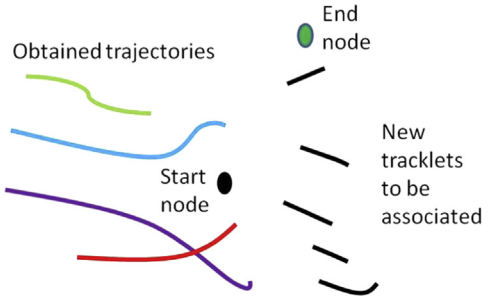


Fig. 6. Illustration for online association based multi-target tracking.

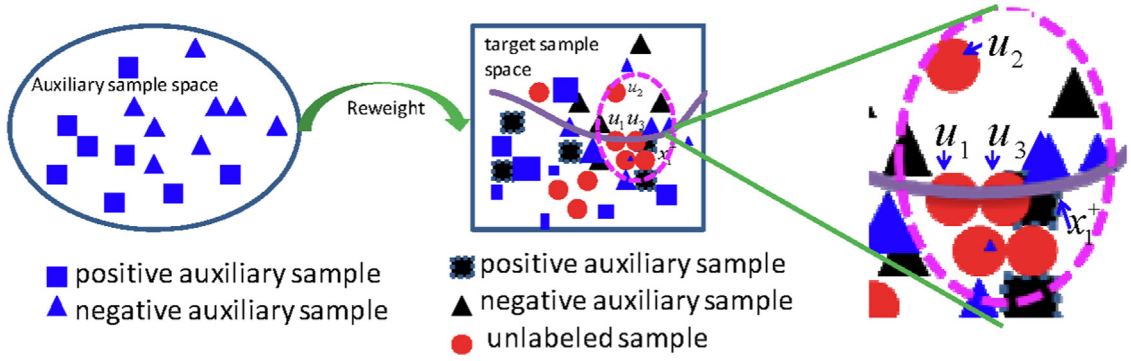


Fig. 5. Illustration for loss function.

where people labeled with red and yellow rectangles is an entry to the view of camera. We statistic the number of detections ($N_{i,j}$) in every region of the frame, then the probability that a target exits and enters from $B_{i,j}$ is estimated with $N_{i,j}(1-\exp(-r^2/\sigma^2))$. Fig. 7(b) is the probability map of Fig. 7(a). Then the tracklets whose last detection is in these regions are more likely to exit the view of the camera; the tracklets whose first detection is in these regions are more likely to be the start of a trajectory.

Let $a(i,j)=\alpha M_{ij}V_{ij}S_{i,j}+(1.0-\alpha)H(T_i,T_j)$. Specially, $a(0,j)$ denotes affinity between *Start* and T_j and $a(i,m+1)$ is the affinity between T_i and *End*. Then multi-target tracking could be formalized as

$$\begin{aligned}&\max_X \quad \sum_{i,j} x_{i,j}a_{i,j}\\ &\text{subject to}: \quad \sum_{j=1}^{r} x_{i,j}=1; \quad i=1,2,\ldots,m\\ &\sum_{i=1}^{m} x_{i,j}=1; \quad j=1,2,\ldots,r\\ &x_{0,0}=0, \quad x_{i,j}\in\{1,0\}\end{aligned} \tag{5}$$

This is a classical matching problem and KM [34] could be used to obtain the optimal solution.

Once we obtain the solution, we get new trajectories and could generate new labeled samples χ_a^{new}. Provided new tracklets to be associated, we construct new labeled target samples χ_t and unlabeled samples U. Then we update $H(\cdot)$ to adapt to target's variances.

Algorithm 1. The proposed multi-target tracking method.

Input: the video frame sequences $I_0, I_1, \ldots, I_F$, people detector
Output: trajectory $T_1, T_2, \ldots, T_k$

Perform object detection and initialize the simple tracker with detection results to get tracklets;
Initialization. $\chi_t = empty$, $\chi_a = empty$, $H_a(\cdot)=0$, $H_t(\cdot)=0$, $H(\cdot)=0$, the max size of χ_a: *Threshold*;
Generate training examples including labeled target samples and unlabeled samples;
Updating appearance similarity function $H(\cdot)$;
while *not reach the end of the video sequences* **do**
  Compute the affinity between tracklets according to appearance and motion information;
  Solve the matching problem using KM [34];
  Associate the tracklets according to the obtained solution;
  $\chi_a = \chi_a \cup \chi_t$;
  if *the* $|\chi_a| > Threshold$ **then**
    discard the $\chi_a - Threshold$ oldest sample;
  end
  Update $H(\cdot)$ using χ_a as auxiliary samples;
  Generate new target labeled samples χ_t according to the obtained solution;
  Update $H(\cdot)$ with new χ_t;
  Generate new tracklets and create new unlabeled samples U;
  Update $H(\cdot)$ with U;
end
return obtained trajectories $T_1, T_2, \ldots, T_k$;

To determine the weight (α) of motion information automatically, we analysis its importance with the training data. First, we set $\alpha=1$, and solve the optimization problem on the training data. The precision only using motion information is denoted as PRE_M. Similar, the precision only using appearance information is denoted as PRE_A. Then we set

$$\alpha=\frac{PRE_M}{PRE_M+PRE_A} \tag{6}$$

The complete procedures of our whole tracking algorithm are shown in Algorithm 1. As discussed in our related work [10], for each updating, the computation complexity of semi-supervised CovBoost is $O((|\chi_a|+|\chi_t|+|U|)MN_w)$, where M and N_w denote the number of selectors and the number of weak classifiers respectively. In our proposed method, when performing tracklets association, the cost time is mainly used for updating semi-supervised CovBoost classifier to infer appearance affinity and running KM matching to obtain association results. Since the computation complexity KM matching algorithm is $O(m^2)$, where m is the number of trajectories, the total computation complexity is $O((|\chi_a|+|\chi_t|+|U|)MN_w+m^3)$.

3.4. Implementation

We implement our tracking algorithm with C++ code. Some related details that readers will care about are described specifically.

Feature: Although many features such as color, texture and HoG could be used for tracking, we select Haar based feature in our experiment because it could be obtained fast and achieves excellent performance [35].

Weak Classifier: Provided two Haar feature value $f_{i,m}, f_{j,m}$ ($j=1,2,\ldots,1000$) of two image patch to be matched, the weak classifiers is defined as $h(f_{i,m},f_{j,m})=sign(p(f_{i,m}-f_{j,m}\in C_o|f_{i,m}-f_{j,m})-p(f_{i,m}-f_{j,m}\in C_b|f_{i,m}-f_{j,m}))$. C_o and C_b denote classes of positive and negative association respectively. We model $p(f_{i,m}-f_{j,m}\in C_o|f_{i,m}-f_{j,m})$, $p(f_{i,m}-f_{j,m}\in C_b|f_{i,m}-f_{j,m})$ with Gaussian distribution $N(\mu_o^i,\sigma_o^i)$ and $N(\mu_b^i,\sigma_b^i)$. During tracking, when receiving the new labeled samples, the parameters (μ_o^i,σ_o^i) are updated. In our implementation, the number (N_w) of weak classifiers is 1000 and the number (M) of selectors is 100.

Memory: We need to store auxiliary samples for updating. The maximal size of auxiliary samples is set to 1000 (*Threshold* = 1000). It is a compromise between tracking speed and accuracy.

Samples collection: Provided the trajectory, the number of available negative samples is larger. So when we collect samples, those tracklet pairs whose frame difference is long or motion

consistency is low are filtered out. After we associate new tracklets with obtained trajectory, the labels of the samples in U could be determined. Meanwhile, χ_t become out-of-date and its samples are added to χ_a; the samples in U as well as their labels construct new χ_t.

4. Experiments

In this section, we design experiments to verify the effectiveness of our methods. To evaluate the performance quantitatively, the most widely used CLEAR MOT [30] metrics are selected as the criteria.

Precision(↑): The number of correctly matched detections divided by the number of output detections.

Recall(↑): The number of correctly matched detections divided by the total number of detections in ground truth.

MT(↑): Mostly tracked trajectories – The percentage of the obtained trajectories whose overlap with ground truth is larger than 80% divided by the number of trajectories in ground truth.

PT(↑): Partially tracked trajectories – The percentage of the obtained trajectories whose overlap with ground truth is between 20% and 80% divided by the number of trajectories in ground truth.

IDS(↓): ID Switches – The total number of times that trajectories change its ID.

FG(↓): Fragments – The total times that a whole trajectory in ground truth is segmented into different trajectories in tracking results.

↑ indicates the higher value, the better is the performance in that criteria; ↓ means just the opposite.

4.1. Analysis on appearance affinity learning algorithm

When learning appearance affinity model, we propose to use Semi-supervised CovBoost algorithm to deal with targets' variations. In this section we compare our learning algorithm with [13], which online learns discriminative appearance models with AdaBoost. To be fair, in this comparison, we use the same feature (HOG, RGB histogram, and covariance matrix). The difference is that we use semi-supervised CovBoost to learn appearance

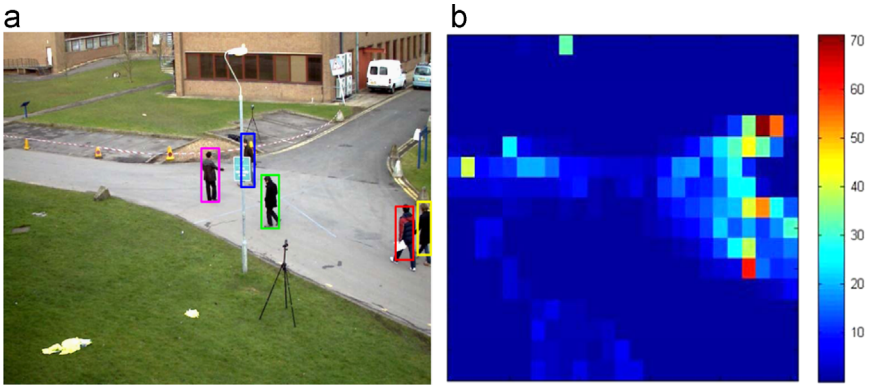


Fig. 7. Illustration for determine *Start* and *End* node using contextual information. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

Fig. 8. Some representative tracking results of SSOCBTDAM.

models for targets (referred as SSOCBTDAM). We conduct experiments on the same test sequences in [13] and display some representative tracking results in Fig. 8.

Table 2
Analysis on appearance affinity learning algorithm on public datasets.

Datasets	Methods	Recall (%)	Precision (%)	MT (%)	PT (%)	FG	IDS
CAVIAR	OLDAM	89.4	96.9	84.6	14.7	18	11
	SSOCBT DAM	90.4	97.1	86	14.5	15	9
TRECVID08	OLDAM	80.4	86.1	76.1	19.3	322	224
	SSOCBT DAM	84.3	89.2	80.2	17.2	301	202

Table 3
Information about our test videos. NOP: number of pedestrians. ANPT: average number of detections per trajectory, ANPF: average number of pedestrians per frame.

Data set	Video name	Total frames	NOP	ANPT	ANPF
PETS	PETS09_S2_L1	794	19	243	5.8
TUD	Stadmitte	179	10	111	9.3
ETH	BAHNHOF	1000	94	89	8.4
	SUNNY DAY	354	30	62	5.3

Table 4
Description of the trackers appeared in our experimental section.

Abbreviation	Description of the tracker
Online **CRF**	Online learned CRF Model for Multi-Target Tracking [14]
DC-CRF	Discrete-continuous **CRF** [27]
OLDAMs	Online Learned Discriminative Appearance Models [13]
DPAMTT	Discriminative Regularization for Online Association Learning [16]
SSOCBMTT_A	Our online multi-target tracker only using appearance affinity
SSOCBMTT	The proposed semi-supervised online CovBoost multi-target tracker

Table 2 displays quantitative comparison and we can see that SSOCBTDAM achieves better results in all criterias. This demonstrates that considering targets' variations and learning with semi-supervised CovBoost algorithm could generate more discriminative appearance models as well as affinity measurement for multi-target tracking.

4.2. Comparison with other multi-target tracking methods

In this section, we test our tracker on three widely used public datasets to show its good performance. The first sequence is PETS [23] S2.L1, showing people walks across a rotary from different directions at various velocities. The color of the coats of many people are similar. The second sequence is TUD Stadmitte, which is about a busy street recorded with a low static camera. The view of the camera is small, and the pedestrian appears large. Therefore, occlusion exists and the length of the trajectory is usually short. Pedestrian enter and exit the view of the camera frequently. Moreover, occlusion often happens. The last two sequences are BAHNHOF and SUNNY DAY from ETH Mobile Scene (*ETHMS*) dataset [24]. The detail information about the test video sequences is introduced in Tables 3. We compare with related works of state-of-the-art [14,25,26,13,15,16], whose description is provided in Table 4. For simplicity, we use abbreviation of every tracker at the following parts. To analyze whether the CovBoost algorithm is effective for appearance affinity learning, we also test the proposed method only using appearance information, referred as SSOCBMTT-A.

Table 5 shows the results of our methods, comparing with the results of other methods, which are reported in the related papers. We can see that although SSOCBMTT could not achieve the best performance in Precision and Recall, its *F*-score is the highest. Moreover, the sums of MT and PT of SSOCBMTT on PETS and TUD are almost one on all the test sequences, meaning that the percentage of the obtained trajectories whose overlap with ground truth is larger than 20%, is nearly 100%. Compared to SSOCBMTT-A, SSOCBMTT makes improvement in all the criterias, proving that incorporating motion information could be very useful. However, the improvement of SSOCBMTT on ETHMS is not so significantly, because the video sequences in ETHMS are recorded with a moving camera and the motion information would be not so accurate.

Table 5
Quantitative comparisons of different tracking methods on public test sequences.

Datasets	Methods	Recall (%)	Precision (%)	Fscore	MT (%)	PT (%)	FM	IDS
PETS_ 09_S2_L1	CRF [14]	91.8	99.0	0.953	89.5	10.5	9	0
	DC-CRF [27]	–	–	–	94.7	5.3	15	22
	Milan [26]	92.4	98.4	0.954	91.3	4.3	6	11
	DPAMTT [16]	96.1	97.5	0.968	94.7	5.3	3	2
	SSOCBMTT_A	96.0	97.3	0.967	94.6	5.2	2	1
	SSOCBMTT	96.8	98.4	0.976	95.1	4.9	1	0
TUD_Stadmitte	CRF [14]	87.0	96.7	0.916	70.0	30.0	1	0
	DC-CRF [25]	–	–	–	40.0	60.0	13	15
	Milan [26]	84.7	86.7	0.857	77.8	22.2	3	4
	DPAMTT [16]	90.6	96.3	0.934	80.0	20.0	4	2
	SSOCBMTT_A	90.8	96.3	0.935	81.2	18.3	5	2
	SSOCBMTT	91.2	97.3	0.942	84.2	15.8	3	1
ETHMS	CRF [14]	79.0	90.4	0.843	68.0	29.0	19	11
	DC-CRF [25]	–	–	–	–	–	–	–
	Milan [26]	77.3	87.2	0.820	66.4	25.4	69	57
	DPAMTT [16]	79.3	87.6	0.832	67.0	26.6	26	21
	SSOCBMTT_A	80.1	85.2	0.826	68.3	27.3	23	16
	SSOCBMTT	82.1	87.3	0.846	68.7	27.6	20	11

5. Conclusions

In this paper, we propose to use semi-supervised CovBoost learning algorithm to learn appearance affinity measurement for multi-target tracking. By considering target's variations, it can learn more discriminative appearance models (or affinity measurement) for tracklets association. To infer the target's entry or exit, contextual information is used to estimate the probability that a tracklet is the start or end of a trajectory. To exploit motion information, we use Gaussian distribution to model the changes of velocity and scale, which could further improve the performance of multi-target tracking.

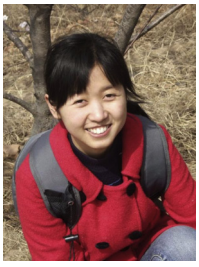
In our future work, we will focus on how to discover more accurate contextual information and how to use them for predicting target's motion.

Acknowledgements

This work was supported in part by National Basic Research Program of China (973 Program): 2012CB316400, by National Natural Science Foundation of China: 61303153, 61025011, 61332016 and 61202322, China Postdoctoral Science Foundation: 2014T70111, President Foundation of UCAS.

References

[1] A.A. Butt, R.T. Collins, Mutli-target tracking by Lagrangian relaxation to min-cost network flow, in: CVPR, 2013.
[2] B. Leibe, K. Schindler, L.V. Gool, Coupled detection and trajectory estimation for multi-object tracking, in: ICCV, 2006.
[3] Q. Yu, G. Medioni, I. Cohen, Multiple target tracking using spatio-temporal Markov chain Monte Carlo data association, in: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 2007.
[4] Y. Li, C. Huang, R. Nevatia, Learning to associate: hybridboosted multi-target tracker for crowded scene, in: CVPR, 2009 pp. 2953–2960.
[5] A.G.A. Perera, C. Srinivas, A. Hoogs, G. Brooksby, W.S. Hu, Multi-object tracking through simultaneous long occlusions and split-merge conditions, in: CVPR, 2006, pp. 666–673.
[6] L. Zhang, Y. Li, R. Nevatia, Global data association for multi-object tracking using network flows, in: CVPR, 2008, pp. 1–8.
[7] H. Jiang, S. Fels, J. Little, A linear programming approach for multiple object tracking, in: CVPR, 2007, pp. 1–8.
[8] H. Grabner, C. Leistner, H. Bischof, Semi-supervised on-line boosting for robust tracking, in: ECCV, 2008.
[9] B. Zeisl, C. Leistner, A. Saffari, H. Bischof, On-line semi-supervised mulitple-instance boosting, in: CVPR, 2010.
[10] G.R. Li, Q.M. Huang, L. Qin, S.Q. Jiang, SSOCBT: a robust semisupervised online covboost tracker that uses samples differently, IEEE Trans. Circuits Syst. Video Technol. 23 (2013) 695–709.
[11] G.R. Li, W. Qu, Q.M. Huang, A multiple targets appearance tracker based on object interaction model, IEEE Trans. Circuits Syst. Video Technol. 22 (2012) 450–464.
[12] B. Yang, R. Nevatia, Online learned discriminative part-based appearance models for multi-human tracking, in: ECCV, 2012.
[13] C.H. Kuo, C. Huang, R. Nevatia, Multi-target tracking by on-line learned discriminative appearance models, in: CVPR, 2010.
[14] B. Yang, R. Nevatia, An online learned crf model for multi-target tracking, in: CVPR, 2012.
[15] S. Kim, S. Kwak, J. Feyereisl, B. Han, Online multi-target tracking by large margin structured learning, in: ACCV, 2012.
[16] Y.K. Xu, L. Qin, G.R. Li, Q.M. Huang, Online discriminative structured output SVM learning for multi-target tracking, IEEE Signal Process. Lett. 21 (2014) 190–194.
[17] J. Liu, P. Carr, R. Collins, Y. Liu, Tracking sports players with context-conditioned motion models, in: CVPR, 2013.
[18] B. Yang, R. Nevatia, Multi-target tracking by online learning of non-linear motion patterns and robust appearance models, in: CVPR, 2012.
[19] R. Collins, Multitarget data association with higher-order motion models, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 1744–1751.
[20] S. Pellegrini, A. Ess, K. Schindler, G.L. Van, You'll never walk alone: modeling social behavior for multi-target tracking, in: ICCV, 2009.
[21] P. Cui, L.F. Sun, F. Wang, S.Q. Yang, Contextual mixture tracking, IEEE Trans. Multimed. 11 (2009) 333–341.
[22] C. Dicle, M. Sznaier, O. Camps, The way they move: tracking targets with similar appearance, in: ICCV, 2013.
[23] J.M. Ferryman, A. Shahrokni, Pets2009: dataset and challenge, in: Winter-PETS, 2009.
[24] A. Ess, B. Leibe, K. Schindler, L.V. Gool, A mobile vision system for robust multi-person tracking, in: CVPR, 2008.
[25] A. Milan, K. Schindler, S. Roth, Detection- and trajectory-level exclusion in multiple object tracking, in: CVPR, 2013.
[26] A. Mila, S. Roth, K. Schindler, Continuous energy minimization for multitarget tracking, IEEE Trans. Pattern Anal. Mach. Intell. 36 (2014) 58–72.
[27] B.D. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, in: International Joint Conference on Artificial Intelligence, 1981.
[28] M. Isard, A. Blake, Condensation-conditional density propagation for visual tracking, Int. J. Comput. Vis. 29 (1998) 5–28.
[29] B.N. Zhong, Y. Chen, Y.J. Shen, Y.W. Chen, R.R. Ji, X.T. Yuan, D.S. Chen, W. B. Chen, Robust tracking via patch-based appearance model and local background estimation, Neurocomputing 123 (2014) 344–353.
[30] K. Bernardin, R. Stiefelhagen, Evaluating multiple object tracking performance: the CLEAR MOT metrics, Image Video Process. 1 (2008) 1C10.
[31] B. Wang, G. Wang, K.L. Chan, L. Wang, Tracklet association with online target-specific metric learning, in: CVPR, 2014.
[32] S. Ali, M. Shah, Floor fields for tracking in high density crowd scenes, in: Proceedings of European Conference on Computer Vision, 2008.
[33] X. Song, X. Shao, H. Zhao, J. Cui, R. Shibasaki, H. Zha, An online approach: learning-semantic-scene-by-tracking and tracking-by-learning-semantic-scene, in: CVPR, 2010.
[34] H.W. Kuhn, Variants of the Hungarian method for assignment problems, Naval Res. Logist. Q. 3 (1956) 253–258.
[35] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, in: CVPR, 2001.



Guorong Li received her B.S. degree in technology of computer application from Renmin University of China, in 2006 and Ph.D. degree in technology of computer application from the Graduate University of the Chinese Academy of Sciences in 2012.

Now, she is a associate professor within the Graduate University of Chinese Academy of Sciences. Her research interests include object tracking, video analysis, pattern recognition and corss-media analysis.

Qingming Huang (SM08) received the B.S. degree in computer science and Ph.D. degree in computer engineering from Harbin Institute of Technology, Harbin, China, in 1988 and 1994, respectively.

He is currently a Professor with the University of the Chinese Academy of Sciences (CAS), Beijing, China, and an Adjunct Research Professor with the Institute of Computing Technology, CAS. He has been granted by China National Funds for Distinguished Young Scientists in 2010. He has authored or coauthored more than 200 academic papers in prestigious international journals and conferences. His research areas include multimedia video analysis, video adaptation, image processing, computer vision, and pattern recognition.

Huang is a reviewer for the IEEE Transactions on Multimedia and the IEEE Transactions on Circuits and Systems for Video Technology. He has served as a TPC member for well-known conferences, including ACM Multimedia, CVPR, ICCV, and ICME.

Shuqiang Jiang (SM08) received the M.S. degree from the College of Information Science and Engineering, Shandong University of Science and Technology, Shandong, China, in 2000, and the Ph.D. degree from the Institute of Computing Technology, CAS, Beijing, China, in 2005.

He is currently a Prefessor with Digital Media Research Center, Institute of Computing Technology, CAS, Beijing, China. He is also with the Key Laboratory of Intelligent Information Processing, CAS. His research interests include multimedia processing and semantic understanding, pattern recognition, and computer vision.

Yingkun Xu received the Master degree of computer science and technology from Nanjing University in 2005. He is now pursuing Ph.D degree in the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS). His research interests include machine learning, computer vision and video technology.

Weigang Zhang received the M.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 2005. He is currently working toward the Ph.D. degree at the School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China. He is also a faculty member with School of Computer, Harbin Institute of Technology at Weihai, Weihai, China. His research interests include image processing, video analysis and pattern recognition.