Discriminative Multiple Target Tracking

Xiaoyu Wang[1] Gang Hua[2] and Tony X.Han[3]

[1] University of Missouri xw9x9@mail.missouri.edu
[2] Nokia Research Center ganghua@gmail.com
[3] University of Missouri hantx@missouri.edu

In this chapter, we introduce a metric learning framework to learn a single discriminative appearance model for robust visual tracking of multiple targets. The single appearance model effectively captures the discriminative visual information among the different visual targets as well as the background. The appearance modeling and the tracking of the multiple targets are all cast in a discriminative metric learning framework. We manifest that an implicit exclusive principle is naturally reinforced in the proposed framework, which renders the tracker to be robust to cross occlusions among the multiple targets. We demonstrate the efficacy of the proposed multiple target tracker on benchmark visual tracking sequences, and real-world video sequences as well.

1 Introduction

Visual tracking of multiple targets has been very active research in the past years [29, 7, 28, 18, 22, 21], largely due to its essentiality in video surveillance, and more emerging applications such as internet video annotation. To robustly track the multiple objects, firstly we need to *model* the visual targets, either based on contour shape or visual appearances. Then a matching algorithm match the image observation data with the models of the multiple targets. Appearance based modeling has induced a lot of attention due to its richness in representation.

For visual appearance modeling of the multiple visual targets, one may model the different visual target separately, e.g., either a generative model is built for each visual target to capture the visual variation [10, 15, 20, 27, 22, 21], or a discriminative model is built for each target to discriminate it from the background [3, 8, 4, 5]. Typical generative model for modeling visual target include appearance based subspace model [10, 15, 20] obtained using embedding methods such as principal component analysis [10, 20], or Gram-Schmidt decomposition [15], as well as Gaussian mixture model [17] learned from the Expectation-Maximization (EM) algorithm [11].

On the other hand, discriminative appearance models leveraged supervised learning algorithms to training a classification function to differentiate the appearances of the visual targets from the background. For example, support vector machine (SVM) and Boosting cascade classifier is adopted in [3] and [7], respectively, for training discriminative visual models, an ensemble classifier based on Boosting is leveraged in [4], a linear discriminative classifier is employed by [8], and a multiple instance Boosting classifier is utilized in [5]. A set of positive examples representing the target object and a set of negative examples representing the background are needed to train the discriminative model.

Compared to generative models, discriminative models aim directly on differentiating the visual target from the background clutter, hence they may be more desirable for robust visual tracking. However, separating the discriminative appearance modeling efforts for the multiple targets is problematic because each model is only focusing on differentiating the associated target with the background where the target presented. The discriminative information among the different visual targets themselves are totally ignored. Effectively capturing the discriminative information among the different visual targets may be vital in dealing with cross occlusions incurred among the multiple targets.

In this chapter, we present a discriminative formulation to learn an joint discriminative appearance model for discriminating the multiple visual targets from the background, as well as discriminating the multiple targets themselves. This formulation is cast under a discriminative metric learning framework proposed by Globerson and Roweis [12]. A nice property of this discriminative formulation is that the learning of the joint model only needs to optimize a convex function using gradient descent, where the optimal solution is guaranteed. Moreover, in our formulation, the visual matching process to track the multiple targets is also optimizing the same objective function as what we used to learn the visual model.

The visual matching process in our tracking algorithm can also be efficiently performed by gradient based optimization using any modern nonlinear optimization packages, such as the one proposed in [31]. This put our multiple target tracking algorithm into the literature of gradient based visual tracking algorithms [13, 9, 30, 14, 26]. Gradient based tracking algorithms directly match the visual model with the image observations based on the gradient of the objective function w.r.t. the motion parameters. It does not make any additional assumptions of the motion and observation models, which may often required in visual tracking algorithms based on hypothesis generation and observation verification, such as Kalman filter (KF) [19], probabilistic data association filter (PDAF) [6], and particle filter [16].

Due to the mutual discrimination of the appearance models of the different visual targets reinforced in the learning process, and the joint optimization of multiple motions, our tracking algorithm reinforces an implicit *exclusive principle* [21]. Exclusive principle, which is firstly defined by MacCormick and

Blake [21] states that no two visual targets shall account for the same image observation, which is vital to handle and being robust to occlusions when dealing with multiple objects tracking. Notice that our proposed formulation for discriminative visual modeling of multiple visual appearances may not be utilized for tracking multiple identical objects in a visual scene. Nevertheless, exact identical multiple objects are scarce in real-world videos. Therefore, this limitation may not hinder the general applicability of the proposed multiple target tracking algorithm.

When compared with previous methods for multiple target tracking algorithms, the proposed modeling and matching framework presents three advantages. Firstly, it presents a discriminative formulation to simultaneously model the appearances of multiple objects, which not only discriminates the visual target from the background, but also seeks for mutual discrimination among the different visual targets. Secondly, in our formulation, an exclusive principle is naturally reinforced, which renders it robust to handle cross occlusions among the different visual target. Thirdly, our proposed framework is easily adapted for online model updating, which is supported by a principled criterion derived from the objective function to select the optimal set of visual examples for online modeling and matching.

2 Appearance and motion model of multiple targets

2.1 Metric learning framework

We cast our discriminative appearance and motion model of mutliple targets by leveraging a metric learning framework similar to Globerson and Roweis [12]. Suppose we have a set of labeled training examples $\mathcal{X} = \{\mathbf{x}_{i,j} \in \mathbb{R}^N, o_{ij}\}_{j=1}^{n_i}$, where $o_{ij} = 0$ indicates background, and $o_{ij} = 1, \ldots, K$ indicates the visual samples of each of the K visual targets we are intending to track. N is the dimension of examples. Let $\mathcal{S}_0 = \{(\mathbf{x}_{0j}, o_{0j} = 0)\}_{j=0}^{n_0}$, and also let $\mathcal{S}_i = \{(\mathbf{x}_{ij}, o_{ij} = i)\}_{j=0}^{n_i}$ for any $i = 1, \ldots, K$, such that $n = \sum_{i=0}^{K}(n_i + 1)$ and $\mathcal{X} = \bigcup_{i=0}^{K} \mathcal{S}_i$. $\mathbf{x}_{ij}$ means the jth example for tracking target $i (i = 0$ implies background). In our experiments, each $\mathbf{x}_{ij}$ is usually a $w \times h$ image patch and $N = w \times h$.

We further denote $\forall i > 0$, $\mathbf{x}_{i0} = \mathbf{I}(\mathbf{m}_i)$ indicates each of the K visual targets we want to track in the current frame where $\mathbf{m}_i \in \mathbb{R}^L$ is the motion parameters we want to recover. $\mathbf{I}(\mathbf{m}_i)$ is a mapping which maps the motion parameters, affine transformation parameters for example, to image patch. Obviously, the label o_{i0} of $\mathbf{I}(\mathbf{m}_i)$ is i, since it represents the i^{th} visual target. For convenience, we will either use $\mathbf{x}_{i0}$ or $\mathbf{I}(\mathbf{m}_i)$ in our presentation depending on if we are learning for the appearance model or performing the visual matching for tracking of the multiple target. Following Globerson and Roweis [12], we propose to learn a Mahalanobis form metric, i.e.,

$$d_{\mathbf{A}}(\mathbf{x}_{ij}, \mathbf{x}_{kl}) = (\mathbf{x}_{ij} - \mathbf{x}_{kl})^T \mathbf{A} (\mathbf{x}_{ij} - \mathbf{x}_{kl}). \tag{1}$$

to achieve our unified formulation, where $\mathbf{A}$ is a positive semi-definite matrix we need to learn from data. Define, for each $\mathbf{x}_{ij} \in \mathcal{X}$, a conditional probability

$$p_{\mathbf{A}}(\mathbf{x}_{kl}|\mathbf{x}_{ij}) = \frac{1}{Z_{ij}} e^{-d_{\mathbf{A}}(\mathbf{x}_{ij}, \mathbf{x}_{kl})} = \frac{e^{-d_{\mathbf{A}}(\mathbf{x}_{ij}, \mathbf{x}_{kl})}}{\sum_{p \neq i \vee q \neq j} e^{-d_{\mathbf{A}}(\mathbf{x}_{ij}, \mathbf{x}_{pq})}}. \tag{2}$$

The ideal distribution of the optimal $\mathbf{A}$ shall collapse samples from the same class to be a single point. Specifically, the ideal distribution shall take the following form,

$$p_0(\mathbf{x}_{kl}|\mathbf{x}_{ij}) = \begin{cases} \frac{1}{n_c} & o_{ij} = o_{kl} = c \\ 0 & o_{ij} \neq o_{kl} \end{cases}. \tag{3}$$

where $c \in \{0, 1, \ldots, K\}$. Recall that $\mathbf{x}_{i0} = \mathbf{I}(\mathbf{m}_i)$. Denote $\mathcal{M} = \{\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_K\}$, we define

$$\begin{aligned} f(\mathbf{A}, \mathcal{M}) &= \sum_{i=0}^{n} KL\left(p_0(\mathbf{x}_{kl}|\mathbf{x}_{ij}) || p_{\mathbf{A}}(\mathbf{x}_{kl}|\mathbf{x}_{ij})\right) \\ &= C + \sum_{i=0}^{K} \sum_{j \neq k=1}^{n_i} \frac{1}{n_i} \left(d_{\mathbf{A}}(\mathbf{x}_{ij}, \mathbf{x}_{ik}) + \log Z_{ij}\right). \end{aligned} \tag{4}$$

where $C = \sum_{y_{ij}=y_{kl}=c} \frac{1}{n_c} \log \frac{1}{n_c}$ is a constant. To have $p_{\mathbf{A}}(\mathbf{x}_{kl}|\mathbf{x}_{ij})$ to be as close to $p_0(\mathbf{x}_{kl}|\mathbf{x}_{ij})$ as possible, we only need to proceed to minimize $f(\mathbf{A}, \mathcal{M})$. More formally, we formulate the following optimization problem,

$$\min\ f(\mathbf{A}, \mathcal{M}) \tag{5}$$

$$s.t.\ \forall \mathbf{a} \in \mathbb{R}^N, \mathbf{a}^T \mathbf{A} \mathbf{a} \geq 0. \tag{6}$$

where the constraint in Eq. 6 confines $\mathbf{A}$ to be a positive semi-definite matrix (PSD). Solving the above optimization problem would allow us to jointly obtain the optimal discriminative appearance models for all of the multiple visual targets defined by $\mathbf{A}$, and track the motions of the all of them as well, which is defined by $\mathbf{m}$. We solve both by efficient gradient based search, as we shall detail in the following sub-sections.

2.2 Joint appearance model estimation

In formulation, discriminative appearance modeling refers to identifying the optimal $\mathbf{A}$ to define the discriminative metric between visual samples. Assume that the motion parameter $\mathbf{m}$ is fixed, following [12], it is easy to figure out that $f(\mathbf{A}, \mathbf{m})$ is a convex function of $\mathbf{A}$. Taking the derivative of $f(\mathbf{A}, \mathcal{M})$ with respect to $\mathbf{A}$, we have

$$\frac{\partial f(\mathbf{A}, \mathcal{M})}{\partial \mathbf{A}} = \sum_{i=0}^{K} \sum_{j=0}^{n_i} \sum_{k=0}^{K} \sum_{l=0}^{n_k} \omega_{ij}(kl)(\mathbf{x}_{kl} - \mathbf{x}_{ij})(\mathbf{x}_{kl} - \mathbf{x}_{ij})^T \tag{7}$$

where

$$\omega_{ij}(kl) = p_0(\mathbf{x}_{kl}|\mathbf{x}_{ij}) - p_{\mathbf{A}}(\mathbf{x}_{kl}|\mathbf{x}_{ij}). \tag{8}$$

Similar to [12], we take a gradient projection algorithm [23] to obtain the optimal $\mathbf{A}$. Specifically the following two steps are performed:

1. GRADIENT DESCENT:$\mathbf{A} = \mathbf{A} - \epsilon\frac{\partial f(\mathbf{A},\mathbf{m})}{\partial \mathbf{A}}$, where ϵ determines the step length for gradient descent.
2. PSD PROJECTION: Compute the eigen-value decomposition of $\mathbf{A}$, i.e., $\{\lambda_k, \mathbf{u}_k\}_{k=1}^N$ such that $\mathbf{A} = \sum_{k=1}^N \lambda_k \mathbf{u}_k \mathbf{u}_k^T$, set $\mathbf{A} = \sum_{k=1}^N \max(\lambda_k, 0)\mathbf{u}_k\mathbf{u}_k^T$.

The first step above performs gradient descent, and the second step reinforces the constraint to make $\mathbf{A}$ to be a positive semi-definite matrix. These two steps are iterated until convergence. Since $f(\mathbf{A}, \mathcal{M})$ is a convex function of $\mathbf{A}$ fixing $\mathcal{M}$, the iteration of these two steps is guaranteed to find the optimal solution of $\mathbf{A}$.

2.3 Motion parameter optimization

In this subsection, we fix the discriminative appearance model $\mathbf{A}$, and develop the gradient descent search for the motion parameters $\mathcal{M}$. Not losing any generality, we assume that each $\mathbf{m}_i$, $\forall i \in \{1, 2, \ldots, K\}$ is a linear motion model, i.e.,

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} a_i & b_i \\ c_i & d_i \end{bmatrix} \begin{bmatrix} x_i' \\ y_i' \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \end{bmatrix} \tag{9}$$

where $[x_i', y_i']^T$ is the canonical coordinates for the labeled examples, and $[x_i, y_i]^T$ is the coordinates in the target video frame. This linear motion model covers a wide variety of visual motions such as translation, scaling, similarity, as well as full affine motion. We proceed to derive the gradient based search for the full affine motion model.

Recall that $\mathbf{x}_{i0} = \mathbf{I}(\mathbf{m}_i)$ is the only term that involves the motion parameter $\mathbf{m}_i$, $\forall i \in \{1, 2, \ldots, K\}$, according to chain rule, we have

$$\frac{\partial f(\mathbf{A}, \mathcal{M})}{\partial \mathbf{m}_i} = \frac{\partial f(\mathbf{A}, \mathcal{M})}{\partial \mathbf{x}_{i0}} \frac{\partial \mathbf{x}_{i0}}{\partial \mathbf{m}_i}. \tag{10}$$

With some mathematical manipulations, it can be shown that

$$\frac{\partial f(\mathbf{A}, \mathcal{M})}{\partial \mathbf{x}_{i0}} = \frac{4}{n_i} \sum_{j=1}^{n_i} \mathbf{A}(\mathbf{x}_{i0} - \mathbf{x}_{ij}) - 2 \sum_{k=1}^{K} \sum_{l=0}^{n_k} \beta_{i0}(kl)\mathbf{A}(\mathbf{x}_{i0} - \mathbf{x}_{kl}). \tag{11}$$

where

$$\beta_{i0}(kl) = p_{\mathbf{A}}(\mathbf{x}_{kl}|\mathbf{x}_{i0}) + p_{\mathbf{A}}(\mathbf{x}_{i0}|\mathbf{x}_{kl}) \tag{12}$$

For any parameter $\xi_i \in \mathbf{m}_i$, again, applying chain rule, we have

$$\frac{\partial \mathbf{x}_{i0}}{\xi_i} = \frac{\partial \mathbf{I}(\mathbf{m}_i)}{\partial \xi_i} = \frac{\partial \mathbf{I}(\mathbf{m}_i)}{x_i} \frac{\partial x_i}{\partial \xi_i} + \frac{\partial \mathbf{I}(\mathbf{m}_i)}{y_i} \frac{\partial y_i}{\partial \xi_i}, \tag{13}$$

where $\frac{\partial \mathbf{I}(\mathbf{m}_i)}{x_i}$ and $\frac{\partial \mathbf{I}(\mathbf{m}_i)}{y_i}$ represents the image gradient in the target frame in horizontal and vertical directions, respectively. For ease of notation, we denote them as $\mathbf{I}_{x_i}$ and $\mathbf{I}_{y_i}$ respectively. Following Eq. 13, we have, $\forall i \in \{1, 2, \ldots, K\}$

$$\frac{\partial \mathbf{x}_{i0}}{\partial a_i} = \mathbf{I}_{x_i} x_i' \ \frac{\partial \mathbf{x}_{i0}}{\partial b_i} = \mathbf{I}_{x_i} y_i' \tag{14}$$

$$\frac{\partial \mathbf{x}_{i0}}{\partial c_i} = \mathbf{I}_{y_i} x_i' \ \frac{\partial \mathbf{x}_{i0}}{\partial d_i} = \mathbf{I}_{y_i} y_i' \tag{15}$$

$$\frac{\partial \mathbf{x}_{i0}}{\partial e_i} = \mathbf{I}_{x_i} \quad \frac{\partial \mathbf{x}_{i0}}{\partial f_i} = \mathbf{I}_{y_i} \tag{16}$$

Therefore, we may easily calculate the gradient of $f(\mathbf{A}, \mathcal{M})$ with respect to $\mathbf{m}_i$ by applying Eq. 10 to Eq. 16. Then we can take a gradient descent step to recover the optimal motion parameter $\mathbf{m}_i$, $\forall i \in \{1, 2, \ldots, K\}$, i.e.,

$$\mathbf{m}_i = \mathbf{m}_i - \eta \frac{\partial f(\mathbf{A}, \mathbf{m}_i)}{\partial \mathbf{m}_i} \tag{17}$$

where the step length η could be estimated, for example, by a quasi-Newton method such as L-BFGS [31].

3 Online matching and updating multiple models

Another challenge in appearance model based multiple target tracking is to robustly adapt the model to the visual environment. This adaptation may be indispensable for robust tracking since the target objects may go through drastic visual changes from environmental conditions such as extreme lighting, occlusions, casting shadows, and pose and view changes. The metric learning formulation we proposed in Eq. 5 enables us to naturally fulfill this task. We proceed to present it in a more formal way.

Extended from the notation of Sec. 2, let $\mathcal{X}^{(t)} = \bigcup \mathcal{S}_i^{(t)}$ be the set of n labeled examples we maintain at time instance t. We also let $\mathbf{A}_t$ be the current discriminative appearance model, and $\mathcal{M}_t = \{\mathbf{m}_i^{(t)}\}_{i=1}^K$ be the motion parameters we need to recover. Hence we have $\mathbf{x}_{i0}^{(t)} = \mathbf{I}^{(t)}(\mathbf{m}_i^{(t)})$. At each time instant t, given $\mathcal{X}^{(t)}$ and $\mathbf{A}_t$, we run the gradient descent optimization algorithm outlined in Sec. 2.3 to obtain the optimal motion parameters $\hat{\mathbf{m}}_i^{(t)}$, $\forall i \in \{1, 2, \ldots, K\}$. This fulfills our visual matching and tracking task. Then we perturb each $\hat{\mathbf{m}}_i^{(t)}$ in turn to generate a set of α background samples $\mathcal{S}_{0\alpha}^{(t+1)}$ to replace the oldest α samples subset $\mathcal{S}_{0\alpha}^{(t)}$ in $\mathcal{X}^{(t)}$. In practice, we sample examples around the current tracked target with a relative bigger distance to replace old background examples. This results in the new labeled examples $\mathcal{X}^{(t+1)}$, i.e.,

$$\mathcal{X}^{(t+1)} = (\mathcal{X}^{(t)} \setminus \mathcal{S}_{0\alpha}^{(t)}) \cup \mathcal{S}_{0\alpha}^{(t+1)}. \tag{18}$$

Since $\mathbf{m}_i^{(t)}$ has been recovered, for ease of presentation, we abuse the notation to temporally define $\mathbf{x}_{i0}^{(t+1)} = \mathbf{I}^t(\mathbf{m}_i^{(t)})$, $\forall i \in \{1, 2, \ldots, K\}$. With $\mathcal{X}^{(t+1)}$ We can then run the gradient projection optimization algorithms outlined in Sec. 2.2 to obtain the optimal $\mathbf{A}_{t+1}$. To proceed with the next matching step to identify the optimal $\mathbf{I}^{t+1}(\mathbf{m}_i^{(t+1)})$, $\forall i \in \{1, 2, \ldots, K\}$, we need to retire one example for each visual target in the current $\mathcal{X}^{(t+1)}$ to update the example set, we propose a least consistent criterion based on the contribution of each of the target examples to the unified cost function $f(\mathbf{A}_{t+1}, \mathcal{M}_t)$. Indeed, fixing $\mathbf{A}_{t+1}$ and $\mathcal{M}_t$, $f(\mathbf{A}_{t+1}, \mathcal{M}_t)$ is a function of $\mathcal{X}^{(t+1)}$, i.e., $f(\mathbf{A}_{t+1}, \mathcal{M}_t) = g(\mathcal{X}^{(t+1)})$. We can similarly define a $g(\cdot)$ function for any subset of $\mathcal{X}^{(t+1)}$ based on Eq. 4. Therefore, for each $\mathbf{x}_{ij} \in \mathcal{X}^{(t+1)}$, a consistent criterion can be defined as

$$c(\mathbf{x}_{ij}) = g\left(\mathcal{X}^{(t+1)}\right) - g\left(\mathcal{X}^{(t+1)} \setminus \{\mathbf{x}_{ij}\}\right). \tag{19}$$

It is easy to understand that the larger $c(\mathbf{x}_{ij})$ is, the more contribution $\mathbf{x}_{ij}$ has made to $f(\mathbf{A}_{t+1}, \Updownarrow_t)$. If the label $o(x_{ij}) = i$, a larger $c(\mathbf{x}_{ij})$ indicates that $\mathbf{x}_{ij}$ is not very compatible to the rest of the visual samples of target i, and hence should be retired from the sample set. More formally, we select

$$\mathbf{x}_i^\star = \arg\max_{\mathbf{x} \in \mathcal{X}^{(t+1)}, o(\mathbf{x})=i} c(\mathbf{x}) \tag{20}$$

to retire from $\mathcal{X}^{(t+1)}$, for each $i \in \{1, 2, \ldots, K\}$. In real operation, we only need to change the numbering of $\mathbf{x}_{i0}^{(t+1)} = \mathbf{I}^t(\mathbf{m}_t)$ to the numbering of $\mathbf{x}_i^\star$, then we reset $\mathbf{x}_{i0}^{(t+1)} = \mathbf{I}^{t+1}(\mathbf{m}_i^{(t)})$, $\forall i \in \{1, 2, \ldots, K\}$, which are initialized to kick off the matching process to recover the optimal motion parameter $\mathcal{M}_{t+1}$.

The above steps will be iterated from time instant t to time instant $t+1$. Therefore we track the multiple visual targets and estimate the joint discriminative visual appearance model in an online fashion, which are all based on efficient gradient based optimization. Most previous approaches resort to heuristics or the oldness of visual samples to select the optimal set of online training examples. While our proposed selection criterion for positive examples in Eq. 20 is derived directly from the objective function of the proposed formulation in a principled fashion. It manifests another benefit of our proposed metric learning framework for discriminative appearance modeling and matching of multiple visual objects.

To initialize the tracking algorithm, we can either run an object detector if it applies, such as a face detector [24] or a human detector [25], if we are tracking a number of faces or persons, or request the users to manually specify the tracking rectangles for the multiple visual target. Then the initialized tracking rectangles are perturbed to form the initial set of labeled examples $\mathcal{X}^{(1)}$. More specifically, perturbed rectangles with sufficient overlap with the initial rectangles are regarded as the visual samples of the corresponding targets, while those perturbed rectangles which are deviated too much from the initial rectangles are deemed as visual samples of the background. This bootstraps learning for the optimal discriminative appearance model $\mathbf{A}_2$, which

is then adopted to obtain the optimal motion parameter $\mathcal{M}_2$. This processes will be repeated as described above.

Last but not least, when maintaining the labeled example set $\mathcal{X}^{(t)}$, we fix a small set of β background and β visual examples extracted from the initialization frame for each of the visual target in the working set, i.e., we never replace them with new examples. This treatment is very important to keep our discriminative appearance model stable and avoid it to be drifted too drastically in the visual tracking process.

4 Discriminant exclusive principle

We argue that the proposed joint formulation for multiple object tracking naturally incorporates an exclusive principle [22] in the matching process. Therefore it is robust to handle occlusions among the different visual objects. The exclusive principle states that no two visual tracker shall occupy the same image observation. Our proposed algorithm naturally achieved it because of the joint discriminative appearance model $\mathbf{A}$, which reinforces the mutual discrimination of the appearances between two visual targets $\mathbf{I}(\mathbf{m}_i)$ and $\mathbf{I}(\mathbf{m}_j)$. To see this more clearly, given an optimal $\mathbf{A}$, if $\mathbf{I}(\mathbf{m}_i)$ and $\mathbf{I}(\mathbf{m}_j)$ occupy similar image regions (a.k.a, $\mathbf{m}_j \doteq \mathbf{m}_i$), and thus have very similar visual appearances, the mutual discriminative information encoded in $\mathbf{A}$ would incur a large value for $f(\mathbf{A}, \mathcal{M})$. Therefore, $\mathbf{m}_j = \mathbf{m}_i$ is not an optimal solution to $\mathcal{M}$. In other words, the optimal motion parameter $\mathcal{M}$ is more likely to occur when $\forall 1 \leq i < j \leq K$, $\mathbf{m}_j \neq \mathbf{m}_i$. Therefore, the exclusive principle among the different visual targets are naturally reinforced, which makes our proposed framework for multiple target tracking to be more robust to cross occlusions among the different visual targets.

5 Experiments

5.1 Visualization of learned appearance model

The appearance model $\mathbf{A}$ defines an discriminative embedding to differentiate the multiple visual objects from the background. Each eigenvector of $\mathbf{A}$ is corresponding to one basis vector of the embedding. To have a better understanding on how the appearance model $\mathbf{A}$ functions, in Fig. 1, we visualize the top 12 eigenvectors of an optimal $\mathbf{A}$ estimated at frame 512 when tracking three persons in the CAVIAR sequence. As clearly observed, they encode the contour and shape information of the target objects. It is quite sensible because A is used in our discriminative framework for discriminating the multiple objects from the background,and also reinforce the mutual discriminations among the different objects. The shape information is probably the most reliable one for the achieving that.

5.2 Multiple target tracking for different video sequences

We evaluate the tracker on two datasets: the CAVIAR [1] videos and the ETH Mobile Scence(ETHMS) [2]. For each single object, we randomly extract 20 positive examples to form a positive set tightly around the initial bounding box of the object. The number of negative examples around a single object is also set to be 20. We will have $20 * N$ negative examples for each object, supposing that we have N objects to be tracked. A confliction solving procedure is employed to avoid extracting a positive example from one object as a negative of another. After obtaining the matched patch in the current frame, negative examples would be generated by randomly selecting patches with a minimum and maximum distance toward the positive. The motion parameters (affine parameters) are kept the same in this step. Half of the positive and negative examples would be kept without updating to help the tracker recover from big changes and occlusions. The normalized pixel intensity is used as the feature. We downsample the image patches to 20×20, regardless of their original dimensions(The feature dimension for each object must be the same to fit into the metric framework). This procedure is implemented by solving a warping equation instead of directly sampling the image patch, which will provide a smoother objective function for the gradient descent optimization in the second step of the iteration.

Figure 2 shows the tracking results for a video from CAVIAR in which three persons walk on the corridor with big scale changes and occlusions. The objects encounter big occlusion by a crossing person from key frame 816. We present the sample results obtained by our tracker, the ILT [20] tracker and the Meanshift [9]. Our tracker shows quite robust responses.The ILT tracker loses the target when it's occluded by a person crossing the corridor. The Meanshift tracker shifts because it cannot deal with big scale change.

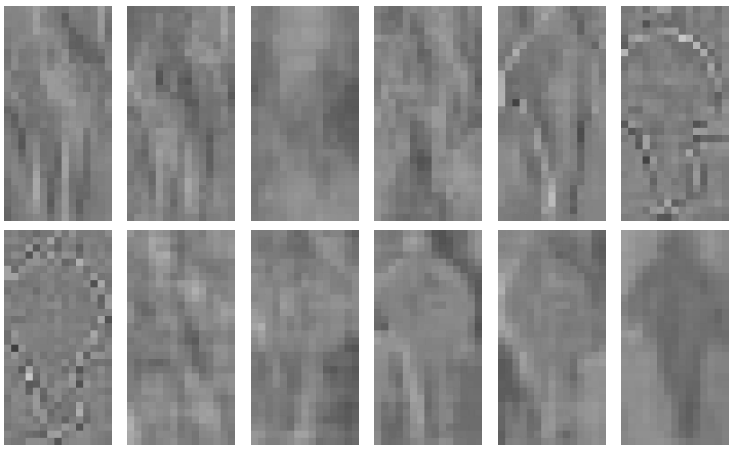

Fig. 1. The top 12 eigenvectors (with the descent order from top left to bottom right) for the discriminative matrix A.

In order to give quantitative performance comparison with these two works, we employ a criterion called Average Tracking Precision(ATP) to do the evaluation, enlighted by the PASCAL grand challenge. More formally, for each tracking task, a ground truth mask for the object of interest is labeled in each frame j. The mask is represented as a point set $\mathcal{G}_j$ which is a collection of all points in the ground truth bounding box. The tracking result is represented as a point set $\mathcal{T}_j$ at frame j. $(x_i, y_i) \in \mathcal{G}_j$ or $\mathcal{T}_j$ indicates that the pixel at (x_i, y_i) is associated with the target. For an ideal tracker, $\forall j$, $\mathcal{G}_j = \mathcal{T}_j$.

For each frame j, the tracking precision r_j is defined as: $r_j = |\mathcal{G}_j \cap \mathcal{T}_j| / |\mathcal{G}_j \cup \mathcal{T}_j|$. Noticing that $r_j \in [0, 1]$, the ATP for a tracker of an object in a video clip is defined as:

$$ATP = \frac{1}{N}\sum_{j=1}^{N} r_j = \frac{1}{N}\sum_{j=1}^{N} \frac{|\mathcal{G}_j \cap \mathcal{T}_j|}{|\mathcal{G}_j \cup \mathcal{T}_j|}, \tag{21}$$

where N is the running length of the video clips in frame number. For an ideal tracker, $ATP \equiv 1$. We use it as the exclusive quantitative measure to compare the performance of the TUDAMM with other state-of-the-art trackers.

Because neither of the other two algorithms support multiple object tracking, we track the objects independently to obtain results from the two trackers. Figure 3 shows the ATP curve. The TUDAMM tracker gives the best performance, with an ATP above 0.7. Recall that in PASCAL grand challenge, a detection with an overlap bigger than 0.5 with ground truth would be treated as a true detection. The ATP value 0.7 implies perfect tracking performance.

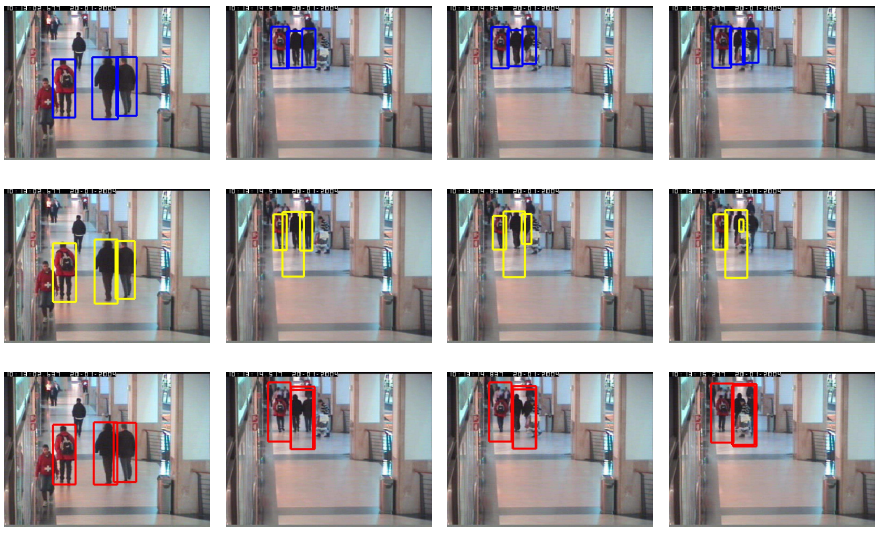

Fig. 2. Sampled multiple object tracking results on the Caviar dataset. Key frame NO.:513,809,817,828. The first row: our tracking results; the second row: tracking results of ILT; the third row: tracking results of Meanshift.

Figure 4 presents sampled key frames from the result of tracking three persons on a street [2]. The person with red coat is occluded by a tree during the tracking. Figure 5 presents sampled tracking results for a video from

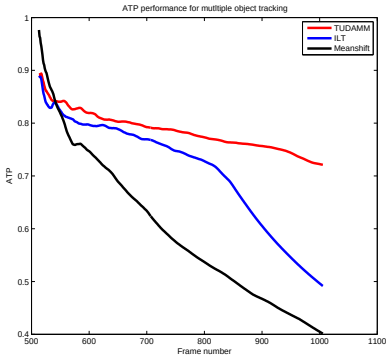

Fig. 3. Tracking performance comparison using ATP. Red curve: TUDAMM;Blue curve: ILT; Black curve: Meanshift

Fig. 4. Multiple Tracking result on the ETH dataset

CAVIAR dataset [1].

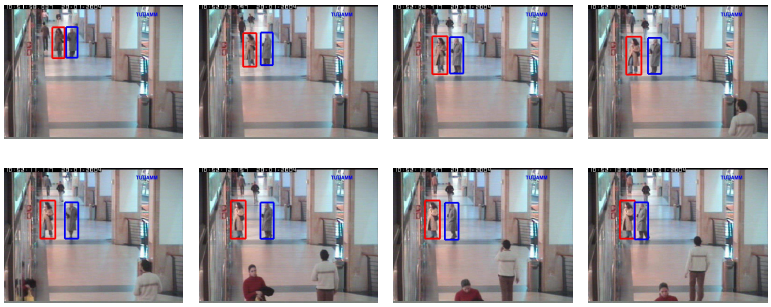

Fig. 5. Multiple Tracking result on the Caviar dataset

Figure 6 shows the tracking result for a horse racing video in which cross occlusion happens frequently. Our tracker show excellent performance. The ILT tracker cannot locate the object very well and the fifth(left to right) horse is completely lost during the tracking process. The Meanshift tracker is not good at solving cross occlusions and the bounding box shifts drastically.

Fig. 6. Multiple Tracking results for a horse racing video. Key frame No: 3026,3143, 3202,3341. The first row:TUDAMM tracker;the second row:the Meanshift tracker; the third row: the ILT tracker.

As we can clearly observe, our discriminative multiple targets tracker present very robust tracking results under drastic visual variations induced by illumination changes, scale changes, pose articulations, as well as mutual occlusions.

6 Discussions, Conclusion and Future Work

We proposes a discriminative metric learning framework for robust tracking of multiple targets. It not only seeks for appearance models to discriminate the multiple foreground targets from the background, but also try to recover subtle discriminations between two different visual targets. Our experiments on a set of challenging real-world video sequences demonstrated the robustness of the proposed tracking algorithms in dealing with large visual variations and cross-occlusions.

Future work may include further exploration of different type of filters for further improving the robustness of the tracker under the same formulation. Meanwhile, as discussed above, this framework may encounter problem if objects are nearly identical. We will further investigate this issue and explore means of mitigate this issue. It may be addressed by posing strong dynamic models learned online, we will defer all this to our future work.

References

1. "http://homepages.inf.ed.ac.uk/rbf/caviardata1".
2. B. A.Ess and L. Gool. Depth and appearance for mobile scene analysis. In *ICCV*, 2007.
3. S. Avidan. Support vector tracking. In *CVPR*, 2001.
4. S. Avidan. Ensemble tracking. In *CVPR*, 2005.
5. B. Babenko, M.-H. Yang, and S. Belongie. Visual tracking with online multiple instance learning. In *CVPR*, 2009.
6. Y. Bar-Shalom. *Tracking and data association*. Academic Press Professional, Inc., San Diego, CA, USA, 1987.
7. Y. Cai, N. de Freitas, and J. J. Little. Robust visual tracking for multiple targets. In *The 9th European Conf. on Computer Vision*, volume 4, pages 107–118, Graz, Austria, May 2006.
8. R. T. Collins and Y. Liu. On-line selection of discriminative tracking features. In *ICCV*, volume 1, pages 346–352, 2003.
9. D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *CVPR*, volume 2, pages 142–149, 2000.
10. T. Cootes, G. Edwards, and C. Taylor. Active appearance models. In *ECCV*, pages 484–498, 1998.
11. A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society Series B*, 39(1):1–38, 1977.
12. A. Globerson and S. T. Roweis. Metric learning by collapsing classes. In *NIPS*, 2005.
13. G. D. Hager and P. N. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(10):1025–1039, 1998.
14. G. D. Hager, M. Dewan, and C. V. Stewart. Multiple kernel tracking with ssd. In *CVPR*, volume 1, pages 790–797, 2004.
15. J. Ho, K.-C. Lee, M.-H. Yang, and D. Kriegman. Visual tracking using learned subspaces. In *CVPR*, volume 1, pages 782–789, 2004.

16. M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *ECCV*, volume 1, pages 343–356, 1996.
17. A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi. Robust online appearance models for visual tracking. In *CVPR*, pages 415–422.
18. Z. Khan, T. Balch, and F. Dellaert. Mcmc data association and sparse factorization updating for real time multitarget tracking with merged and multiple measurements. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(12):1960–1972, December 2006.
19. J. W. Lee, M. S. Kim, and I. S. Kweon. A kalman filter based visual tracking algorithm for an object moving in 3d. In *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Vol. 1*, pages 342–347, 1995.
20. J. Lim, D. Ross, R.-S. Lin, and M.-H. Yang. Incremental learning for visual tracking. In *NIPS*, pages 801–808, 2005.
21. J. MacCormick and A. Blake. A probabilistic exclusion principle for tracking multiple objects. In *ICCV*, pages 572–587, 1999.
22. J. MacCormick and M. Isard. Partitioned sampling, articulated objects, and interface-quality hand tracking. In *ECCV*, pages 3–19, 2000.
23. J. B. Rosen. The gradient projection method for nonlinear programming. part i. linear constraints. *Journal of the Society for Industrial and Applied Mathematics*, 8(1):181–217, March 1960.
24. P. Viola and M. J. Jones. Robust real-time face detection. *Int. J. Comput. Vision*, 57(2):137–154, 2004.
25. X. Wang, T. X. Han, and S. Yan. An hog-lbp human detector with partial occlusion handling. In *ICCV*, 2009.
26. Y. Wu and J. Fan. Contextual flow. In *CVPR*, 2009.
27. M. Yang and Y. Wu. Tracking non-stationary appearances and dynamic feature selection. In *CVPR*, 2005.
28. Q. Yu, G. Medioni, and I. Cohen. Multiple target tracking using spatio-temporal markov chain monte carlo data association. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Minneapolis, Minnesota, June 2007.
29. T. Yu and Y. Wu. Collaborative tracking of multiple targets. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, volume I, pages 834–841, Washington, DC, June 2004.
30. Q. Zhao, S. Brennan, and H. Tao. Differential emd tracking. In *ICCV*, 2007.
31. C. Zhu, R. H. Byrd, P. Lu, and J. Nocedal. Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transaction Mathematical Software*, 23(4):550–560, 1997.