Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)

Robust MIL-Based Feature Template Learning for Object Tracking

Xiangyuan Lan,[†] Pong C. Yuen,[†] Rama Chellappa[‡]
[†]Department of Computer Science, Hong Kong Baptist University, Hong Kong SAR, China
[‡]Center for Automation Research and ECE Department, University of Maryland, College Park, MD, USA
{lanxiangyuan, pcyuen}@comp.hkbu.edu.hk, rama@umiacs.umd.edu

Abstract

Because of appearance variations, training samples of the tracked targets collected by the online tracker are required for updating the tracking model. However, this often leads to tracking drift problem because of potentially corrupted samples: 1) contaminated/outlier samples resulting from large variations (e.g. occlusion, illumination), and 2) misaligned samples caused by tracking inaccuracy. Therefore, in order to reduce the tracking drift while maintaining the adaptability of a visual tracker, how to alleviate these two issues via an effective model learning (updating) strategy is a key problem to be solved. To address these issues, this paper proposes a novel and optimal model learning (updating) scheme which aims to simultaneously eliminate the negative effects from these two issues mentioned above in a unified robust feature template learning framework. Particularly, the proposed feature template learning framework is capable of: 1) adaptively learning uncontaminated feature templates by separating out contaminated samples, and 2) resolving label ambiguities caused by misaligned samples via a probabilistic multiple instance learning (MIL) model. Experiments on challenging video sequences show that the proposed tracker performs favourably against several state-of-the-art trackers.

1 Introduction

As an important step in intelligent motion perception, object tracking has received great research interests with the development of numerous tracking algorithms and applications. Since dramatic appearance changes, caused by illumination, pose, occlusion, etc, may occur on the tracked target during tracking, to adapt the online tracker to such appearance changes, a key problem is how to develop an adaptive model learning (updating) strategy.

With limited amount of labeled tracking samples available in the first frame, additional examples of the tracked targets which are collected by the tracker itself should be utilized for model updating (e.g. (Ross et al. 2008), (Mei and Ling 2011), (Babenko, Yang, and Belongie 2011)). While model adaptivity can be enhanced with more training samples, instability may also be increased when updating is done, leading to poor tracking performance. The problem of corrupted samples is usually encountered in tracking, which is mainly caused by two scenarios. First, large extrinsic variations such as occlusion, large illumination change may introduce outliers into target appearance, which may contaminate the tracking samples and thereby deteriorate the representation power of tracking model. Second, slight tracking inaccuracy, which is often caused by intrinsic variations (e.g. deformation and in/out of plane rotation), may lead to misaligned tracking samples, which usually introduces label ambiguity and 'confuses' the tracker itself. Updating with such samples may contribute to reduced discriminability and drift, gradually leading to tracking failure. Therefore, while maintaining adaptivity to appearance changes, the model updating strategy should be able to handle corrupted tracking samples caused by these two scenarios so as to reduce tracking drift.

However, most existing tracking algorithms which aim to reduce tracking drift are not effective in handling either or both scenarios. One kind of approaches explicitly models the outliers in the corrupted target's samples caused by occlusion or noise, such as sparsity-based trackers (Mei and Ling 2011) (Mei et al. 2013). Although they can detect and prevent contaminated samples (e.g. occluded samples) from updating which enhances their robustness to outlier samples to some extent, most of them do not explicitly handle the misaligned samples. Another kind of approaches are weakly/semi-supervised learning-based methods, such as online multiple instance learning (MIL) (Babenko, Yang, and Belongie 2011) and online semi-supervised boosting (Grabner, Leistner, and Bischof 2008), which aim to resolve the label ambiguities caused by misaligned samples within the framework of weakly/semi-supervised learning. However, the strength of their learning methods are only exploited to construct an optimal ensemble of base classifiers, and each base classifier keeps updated every frame, which is more likely to update with contaminated samples and thereby deteriorates the discriminability. Although some heuristic methods such as sample selection or sample weighting are also developed to handle these two scenarios without considering different specialties of these two issues, they may rely heavily on some prior knowledge (e.g. predefined thresholds for corrupted sample decision (Zhong, Lu, and Yang 2014), pre-defined updating rate of sample weights (Li et al. 2012)), which may not be practical and limits the trackers' flexibility to variations under different

scenarios.

To overcome the aforementioned problems, this paper proposes a novel feature template learning model based on a probabilistic multiple instance learning strategy for effective online tracking modeling updating. This model integrates the robustness of sparse representation and the flexibility of multiple instance learning into the model updating process, which enables the tracker to separate out the corrupted samples and resolving label ambiguity caused by misaligned samples in an optimal unified feature template learning framework. Within this framework, uncontaminated feature templates are learned and updated using bag of instances and contaminated samples can be separated out via sparsity modeling. Therefore, even with noisy tracking samples, the learned feature templates can capture the intrinsic characteristics and the appearance changes of the tracked target, which guarantees both the adaptivity and stability of tracking model updating. Moreover, we develop an iterative optimization algorithm to obtain the optimal solution, which guarantees the optimality of model updating.

It should be noted that several multiple-instance-learning-based tracking algorithms have been proposed (e.g. (Babenko, Yang, and Belongie 2011)). These methods formulate tracking as an online multiple instance boosting problem and aim to construct an optimal ensemble of base classifiers. However, their tracking methods update the base classifiers every frame without considering the potentially corrupted samples, which may degrade the tracking performance. Different from these methods, the proposed method explicitly considers corrupted samples, and aims to learn uncontaminated feature templates for appearance modeling. The proposed method is also different from other dictionary learning-based trackers (e.g.(Liu et al. 2016)) which may treat misaligned samples as strongly-labeled ones for model updating. The proposed method uses bag of samples to learn and update feature templates within the framework of multiple instance learning, which is less sensitive to misaligned samples. Some MIL-based feature learning methods are also proposed for other pattern classification tasks, e.g. (Shrivastava et al. 2015). However, their methods do not explicitly model the outliers present in corrupted samples, which may not be suitable for tracking problem.

The contributions of this paper are as follows:

- A feature template learning model based on multiple instance learning is proposed for updating tracking model.
- An iterative optimization algorithm is derived to build the learning model.

2 Related Work

This section briefly reviews some recent works on object tracking based on feature learning and multiple instance learning.

Feature learning for object tracking Recent online object tracking approaches based on feature learning include dictionary-based methods and neural network-based methods. Several dictionary-based methods have been developed to facilitate effective model updating (Liu et al. 2016), enhance the discriminability of tracking model (Lan, Zhang, and Yuen 2016) (Zhang et al. 2016), etc. However, most of them treat the tracking results in every frame as positive samples which are directly used for model updating. Once the tracking result is not precise, updating with such misaligned results may lead to drift problem. Neural network-based methods such as (Li, Li, and Porikli 2014) update a pre-trained off-line neural network online using the tracking samples to adapt the appearance changes. Such methods may not be efficient, and may still contribute to the drift once corrupted samples are used for model updating. Different from aforementioned approaches, the proposed method aims to use weakly-labeled data for feature learning without off-line large-scale training samples.

Multiple instance learning for object tracking To construct an optimal ensemble of classifiers with potential misaligned tracking samples, (Babenko, Yang, and Belongie 2011) cast object tracking as an online multiple instance boosting problem in which base classifier are selected for combination by maximizing the log likelihood of sample bags. Along this line, more variants have been developed by incorporating sample importance (Zhang and Song 2013), introducing unlabeled data (Zeisl et al. 2010), etc. However, contaminated samples (e.g. occluded samples) may be used for updating the base model.

3 Proposed Model

This section introduces the proposed robust MIL-based feature template learning model from two aspects: probabilistic contaminated feature modeling and resolving label ambiguity within a probabilistic multiple instance learning framework, and then derives the optimization procedure for solving the feature template learning model.

3.1 Robust MIL-based Feature Template Learning

Robust feature template learning via probabilistic contaminated feature modeling Let $Y = [y_1, \ldots, y_n] \in \mathbb{R}^{d\times n}$ denote the target samples obtained by shifting the bounding box by a few pixels in current frame, and n is the number of target samples in the training set. Since unpredictable large appearance changes caused by some variations (e.g. occlusion, illumination, etc) may occur during tracking, the obtained tracking samples may be contaminated. Inspired by the robustness of sparse representation (Wright et al. 2010), one objective of the learning model is to adaptively learn the uncontaminated feature templates for sparse representation of the tracked object while explicitly modeling the contaminated features as discussed below. Let

$$Y = DX + E \tag{1}$$

where $D = [D_{\cdot,1}, \ldots, D_{\cdot,p}] = [(D_{1,\cdot})^T, \ldots, (D_{d,\cdot})^T]^T \in \mathbb{R}^{d\times p}$ are the set of templates, $D_{\cdot,p'}$, and $D_{r',\cdot}$ denote the p'-th column and r'-th row of D, respectively, and $X = [X_{\cdot,1} \ldots X_{\cdot,n}]$ is the sparse coefficient matrix of target samples Y for linear combination of templates, and $E = [E_{\cdot,1} \ldots E_{\cdot,n}]$ are the separated contaminated features for the samples. As mentioned, the appearance variations (e.g. occlusion, illumination variation) come in uncertainty

and may introduce some outliers into the target samples. To explicitly model the uncertainty existing in the outliers for contaminated samples, we incorporate the probabilistic constraint that each element in E is independent and subjected to a zero-mean Laplace distribution with variance $2/b^2$, i.e.

$$P(E|\cdot;b) = (b/2)^{(nd)} \exp(-b\|E\|_1), \tag{2}$$

which implies

$$P(Y|D,C;b) = (b/2)^{(nd)} \exp(-b\|Y - DX\|_1) \tag{3}$$

Since Laplace distribution have more heavy tail than some other distributions, e.g. Gaussian distribution, it is more able to tolerate the outliers, which facilitates the insensitivity to contaminated samples. To enhance the representativeness of the learned feature templates for sparse representation and enable each template to characterize different properties of the tracked target, enforcing sparsity constraint on the coefficients is essential. As such, the prior distribution for each $X_{\cdot,j}$ is defined as: $P(X_{\cdot,j}) \propto \exp(-\lambda\phi(X_{\cdot,j}))$, where $\phi(\cdot)$ is a sparsity function and λ is a constant. Here we use ℓ_1 penalty as the sparsity function, i.e. $\phi(X_{\cdot,j}) = \|X_{\cdot,j}\|_1$. Assuming a uniform prior on the each feature template, by taking the logarithm of the joint distribution with respect to Y, D, and X, i.e. $P(Y,D,X|\cdot;b) = P(Y|D,X;b)P(D|\cdot;b)P(X|\cdot;b)$, then the maximum a posteriori (MAP) estimation of the feature templates and the sparse coefficients is the solution to the following problem:

$$\begin{aligned} &\min_{D,X} b\|Y - DX\|_1 + \lambda\|X\|_1 \\ &\text{s.t. } \|D_{\cdot,j}\|_2 \le c, j = 1,\dots,n \end{aligned} \tag{4}$$

Here we provide a better interpretation of (4). Combined with (1), (4) can be rewritten as

$$\begin{aligned} &\min_{D,C,E} b\|E\|_1 + \lambda\|X\|_1 \\ &\text{s.t. } \|D_{\cdot,j}\|_2 \le c, j = 1,\dots,n \\ &Y = DX + E \end{aligned} \tag{5}$$

From (5), we find that (4) is equivalent to minimizing the sparse regularization of the corrupted features and the coefficient vectors respectively. As in robust dictionary learning (Zhao, Wang, and Cham 2011), the first sparsity regularization aims to model the outliers present in corrupted features while the second one aims to enable different templates to capture different distinctive properties of the target for enhanced sparse representation.

Resolving label ambiguity via bag-level-based MAP Although the feature template learning model in (4) aims to learn the uncontaminated representative feature templates for sparse representation of the tracked object, it ignores the fact that misaligned samples may exist in the tracking samples. Generally, misaligned samples come from two cases. First, some intrinsic variations such as deformations, rotation may cause tracking inaccuracy, and imprecise tracker location leads to misalignment. Second, to obtain more positive samples in current frame for more effective model training, most trackers such as (Fan et al. 2014) shift the tracker location in current frame to capture more samples, which causes misalignment and thereby leads to label ambiguity.

To deal with these two cases and resolve the label ambiguity, we extend (4) to a more general framework in which feature templates are learnt using the collection of weakly-labeled samples based on a probabilistic multiple instance learning (MIL) strategy. Under the setting of MIL, only the label information for collection of samples called bag are available. A bag is positive if at least one of its samples is positive otherwise the bag is negative. For tracking problem, label ambiguity usually occurs among misaligned samples taken from a small neighborhood around the tracking position, while there is no ambiguity about negative samples which are far from the tracking position. As such, in each frame, by randomly perturbing the tracking position by a few pixels several times, multiple samples can be obtained which compose the positive bag, and each negative samples (i.e. background samples) located far from tracking position can be regarded as a negative bag. Examples of the positive bag and the negative bags in one frame are illustrated in Figure 1.

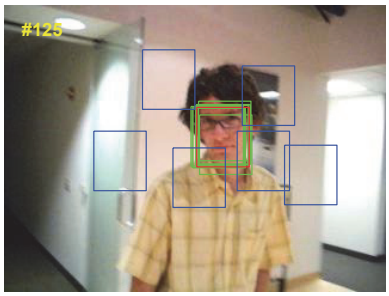

Figure 1: Illustration of sample bags in the 125th frame of *DavidIndoor*. The red bounding box denotes the tracker position, the green bounding boxes are the perturbation of the tracker position which constitute the positive bag, and the blue boxes are the negative samples each of which is a negative bag.

Let y^{ij} denote the j-th sample in the i-th sample bag, c^{ij} be the corresponding sparse coefficients. Without loss of generality, we assume $i = 1,\dots,N^+$ is the index of positive bag and $i = N^+ + 1,\dots,N$ is the index of negative bag. Based on (3) which explicitly models the outliers present in contaminated features, the probability of an instance y^{ij} belonging to the foreground (target) P^{ij} can de defined as $P^{ij} = P(y^{ij}|D,x^{ij};b) = (b/2)^d \exp(-b\|y^{ij} - Dx^{ij}\|_1)$. Then the probability of the i-th sample bag belonging to the foreground (target) can be defined as $P^i = \max_j P^{ij}$. Since only the label for the sample bag is available, inspired by (Shrivastava et al. 2014), the likelihood at bag-level is derived as follows:

$$L(\Omega) = \prod_{i=1}^{N^+}(\max_j P^{ij}) \prod_{i=N^++1}^{N} \prod_{j=1}^{S^i}(1 - P^{ij}) \tag{6}$$

where $\Omega = \{D, x^{ij}\}$ is the set of parameters to be estimated, $\prod_{j=1}^{S^i}(1 - P^{ij})$ is the probability for not being the positive bag, and S^i is the number of samples in the i-th bag. Since there is only one negative sample in the negative bag, S^i should be 1 for $i = N^+ + 1,\dots,N$. We can see that for the likelihood in (6) to be high, at least one sample in each positive bag should have high probability to be the target, while all the negative samples in the negative bag should have low probability. Therefore, unlike the feature learning model in (4) which requires each sample should have high probability to be a positive sample, the model in (6) relaxes the requirement and provide the tracker itself with more flexibility to find out the 'true' positive sample from weakly labeled data. This alleviates label ambiguity in each positive bag. For tractable optimization, we follow the standard approach for MIL (Zhang, Platt, and Viola 2005) and approximate the bag probability using

the generalized mean

$$\max_j P^i \approx \left(\frac{1}{S^i} \sum_{j=1}^{S^i} (P^{ij})^\alpha \right)^{1/\alpha} \tag{7}$$

As shown in Figure 1, the samples from a small neighborhood around the tracking position are partially overlapped, and thus they are spatially correlated. Therefore, such an approximation can utilize probabilistic information of all the correlated samples to model the bag probability. For computationally efficiency, α is set to be 1 in this paper. That is to say, the average of the instance probability is used as the approximation in the proposed model. With the same prior distribution on the feature templates and sparse coefficients as (4), by taking the logarithm, the bag-level-based MAP estimation of the feature template and sparse coefficients can be obtained by solving the following problem:

$$\begin{aligned} \min_{\Omega} & - \sum_{i=1}^{N^+} \log \left(\frac{1}{S^i} \sum_{j=1}^{S^i} (b/2)^d \exp(-b\|y^{ij} - Dx^{ij}\|_1) \right) \\ & - \beta \sum_{i=N^++1}^{N} \log \left(1 - (b/2)^d \exp(-b\|y^{i1} - Dx^{i1}\|_1) \right) \\ & + \lambda \sum_{i=1}^{N} \sum_{j=1}^{S^i} \|x^{ij}\|_1 \\ & \text{s.t. } \|D_{\cdot,j}\|_2 \le c, j = 1, \ldots, n \end{aligned} \tag{8}$$

where β is incorporated to control the effect of negative samples. As the unified feature learning framework of the proposed model, problem (8) not only shares the same merit with (4) by imposing the same probabilistic constraint on the contaminated samples, but also consider the misalignment problem via multiple instance learning strategy. We can see that the first part of the objective function (8) is identical to the reconstruction error term in (4) if there is only one sample in each bag, i.e. $S^i = 1$ for $i = 1, \ldots, N^+$. Therefore, the proposed feature template learning model provides a more general framework under the setting of multiple instance learning to handle label ambiguity existing in each sample bag. Further more, unlike (4) which only considers the reconstruction ability and aims to learn feature templates for accurate representation of target samples, minimizing the second term in (8) further enforces the background samples to be poorly represented by the learned feature templates, which implicitly strengthens the discriminability of the learning model and enables the tracker to be less sensitive to the cluttered background. We derive the optimization algorithm for (8) in the following subsection.

3.2 Optimization

As the ℓ_1 penalty function is non-differential, we approximate the penalty function by a smooth function for tractable optimization. The ℓ_1 penalty function satisfies that

$$\|y^{ij} - Dx^{ij}\|_1 = \sum_{k=1}^{d} |y^{ij}_{k,\cdot} - D_{k,\cdot}x^{ij}|, \tag{9}$$

and thereby minimizing the ℓ_1 penalty function is equivalent to minimizing the absolute loss separatively. Let $u = y^{ij}_{k,\cdot} - D_{k,\cdot}x^{ij}$. Based on the property of conjugation, we can derive

$$|u| = \max_s s \cdot u \quad \text{s.t.} -1 \le s \le 1, \tag{10}$$

According to (Nesterov 2005), the smooth version with smooth parameter θ of $|y^{ij}_{k,\cdot} - D_{k,\cdot}x^{ij}|$, denoted by $g_\theta(y^{ij}_{k,\cdot}, D_{k,\cdot}, x^{ij})$, can be given by subtracting a prox-function $p(s)$ from the objective function of (10). Here we choose $p(s) = \frac{\theta}{2}s^2$ and thus the smoothed function is

$$\begin{aligned} g_\theta(y^{ij}_{k,\cdot}, D_{k,\cdot}, x^{ij}) = \max_s & \; s \cdot (y^{ij}_{k,\cdot} - D_{k,\cdot}x^{ij}) - \frac{\theta}{2}s^2 \\ & \text{s.t.} -1 \le s \le 1 \end{aligned} \tag{11}$$

which is a convex problem. By taking the derivative of the objectives function in (11), setting it to be zero and the projection to the convex set defined by the constraints, we can obtain

$$s = \text{median} \left\{ \frac{y^{ij}_{k,\cdot} - D_{k,\cdot}x^{ij}}{\theta}, -1, 1 \right\} \tag{12}$$

Therefore, the smoothed function is derived as $g_\theta(y^{ij}_{k,\cdot}, D_{k,\cdot}, x^{ij})$

$$= \begin{cases} |y^{ij}_{k,\cdot} - D_{k,\cdot}x^{ij}| - \frac{\theta}{2}, & |y^{ij}_{k,\cdot} - D_{k,\cdot}x^{ij}| > \theta \\ \frac{(y^{ij}_{k,\cdot} - D_{k,\cdot}x^{ij})^2}{2\theta}, & \text{else} \end{cases} \tag{13}$$

Accordingly, the partial derivatives of $g_\theta(y^{ij}_{k,\cdot}, D_{k,\cdot}, x^{ij})$ with respect to $D_{k,\cdot}$ and x^{ij} are derived as follows: $\frac{\partial g_\theta(\cdot,\cdot,\cdot)}{\partial D^T_{k,\cdot}} =$

$$\begin{cases} \frac{D_{k,\cdot}x^{ij} - y^{ij}_{k,\cdot}}{\theta} x^{ij} & |y^{ij}_{k,\cdot} - D_{k,\cdot}x^{ij}| \le \theta \\ \text{sign}(D_{k,\cdot}x^{ij} - y^{ij}_{k,\cdot})x^{ij} & \text{else} \end{cases} \tag{14}$$

and $\frac{\partial g_\theta(\cdot,\cdot,\cdot)}{\partial x^{ij}} =$

$$\begin{cases} \frac{D_{k,\cdot}x^{ij} - y^{ij}_{k,\cdot}}{\theta} D^T_{k,\cdot} & |y^{ij}_{k,\cdot} - D_{k,\cdot}x^{ij}| \le \theta \\ \text{sign}(D_{k,\cdot}x^{ij} - y^{ij}_{k,\cdot})D^T_{k,\cdot} & \text{else} \end{cases} \tag{15}$$

Based on (13), the ℓ_1 penalty function in (15) can be approximated by $G_\theta(y^{ij}, D, x^{ij})$ where

$$G_\theta(y^{ij}, D, x^{ij}) = \sum_{k=1}^{d} g_\theta(y^{ij}_{k,\cdot}, D_{k,\cdot}, x^{ij}). \tag{16}$$

Algorithm 1: Overall Optimization Procedure for (17)

Input: sample bags $\{[y^{ij}]_{j=1}^{S^i} | 1 \le i \le N\}$, total sample bag number N, positive sample bag number N^+, $x^{ij,t}$

Output: $\{[x^{ij}]_{j=1}^{S^i} | 1 \le i \le N\}$, D

Initialization: $t \leftarrow 1, D^t \leftarrow D^0, x^{ij} \leftarrow \mathbf{0}$

while *stopping conditions are not satisfied* **do**

- Update D^{t+1} via Algorithm (2)
- Update $x^{ij,t+1}$ via Algorithm (3)
- $t \leftarrow t + 1$
- Check stopping conditions

end

Algorithm 2: Procedure for Updating Feature Templates D in (17)

Input: sample bags $\{[y^{ij}]_{j=1}^{S^i}|1 \le i \le N\}$, total sample bag number N, positive sample bag number N^+, $\{D^t, x^{ij,t}\}$ in the $t-$th iteration of Algorithm 1

Output: D^{t+1}

Initialization: $l \leftarrow 1, D^l \leftarrow D^t, x^{ij} \leftarrow x^{ij,t}, L \leftarrow 0.2, \alpha^0 \leftarrow 1, \alpha^1 \leftarrow 1, Q \leftarrow D^t$

while *stopping conditions are not satisfied* **do**

  $r \leftarrow 1$

  $D_{k,\cdot}^{l+1} \leftarrow Q_{k,\cdot} - \frac{1}{L}\left(\nabla_{D_{k,\cdot}^T} J_1(Q_{k,\cdot})\right)^T$ **for** all k

  if $D_{\cdot,m}^{l+1} > c$ for some m **then**

    $D_{\cdot,m}^{l+1} \leftarrow \frac{D_{\cdot,m}^{l+1}}{\|D_{\cdot,m}^{l+1}\|_2}$

  end

  while $J_1(D^{l+1}) > J_1(Q) + \nabla_D^T J_1(Q)(D^{l+1} - Q) + \frac{L}{2}\|D^{l+1} - Q\|_F^2$ **do**

    $L \leftarrow 2^r \cdot L$

    $D_{k,\cdot}^{l+1} \leftarrow Q_{k,\cdot} - \frac{1}{L}\left(\nabla_{D_{k,\cdot}^T} J_1(Q_{k,\cdot})\right)^T$ **for** all k

    if $D_{\cdot,m}^{l+1} > c$ for some m **then**

      $D_{\cdot,m}^{l+1} \leftarrow \frac{D_{\cdot,m}^{l+1}}{\|D_{\cdot,m}^{l+1}\|_2}$

    end

    $r \leftarrow r + 1$

  end

  $\alpha^{l+1} \leftarrow \left(1 + \sqrt{4(\alpha^l)^2 + 1}\right)/2$

  $Q \leftarrow D^{l+1} + \frac{\alpha^l - 1}{\alpha^{l+1}}(D^{l+1} - D^l)$

  $l \leftarrow l + 1$

  Check stopping conditions: $\frac{\|D^l - D^{l-1}\|_F}{\|D^{l-1}\|_F} < \epsilon$

end

$D^{t+1} \leftarrow D^l$

The objective function in (8) can be approximated by

$$\begin{aligned} &\min_{\Omega} J_1(\Omega) + J_2(\Omega) \qquad (17)\\ &\text{s.t. } \|D_{\cdot,j}\|_2 \le c, j = 1, \ldots, n,\\ &J_1(\Omega) =\\ &-\sum_{i=1}^{N^+} \log\left(\frac{1}{S^i}\sum_{j=1}^{S^i}(b/2)^d \exp(-bG_\theta(y^{ij}, D, x^{ij}))\right)\\ &-\beta \sum_{i=N^++1}^{N} \log\left(1 - (b/2)^d \exp(-bG_\theta(y^{ij}, D, x^{ij}))\right),\\ &J_2(\Omega) = \lambda \sum_{i=1}^{N}\sum_{j=1}^{S^i} \|x^{ij}\|_1, \end{aligned}$$

where $J_1(\Omega)$ is differential while $J_2(\Omega)$ is non-smooth. Problem (17) is not jointly convex with D and $\{c^{ij}\}$, but it is convex with respect to each of them when the other is fixed. It is difficult to derive the analytical solution to (17). Therefore, we derive an iterative optimization algorithm to solve this problem. We employ fast proximal gradient method with line search (Nesterov 2013) to update the optimal variables iteratively. Therefore, we first derive the gradient of $J_1(\Omega)$ with respect to $D_{k,\cdot}^T$ and x^{ij} based on (14) and (15) as follows:

$$\begin{aligned} &\nabla_{D_{k,\cdot}^T} J_1(\Omega) = \sum_{i=1}^{N^+} b\nabla_{D_{k,\cdot}^T} g_\theta(y_{k,\cdot}^{ij}, D_{k,\cdot}, x^{ij}) + \qquad (18)\\ &\sum_{i=N^++1}^{N} \frac{F(y^{ij}, D, x^{ij}) b\nabla_{D_{k,\cdot}^T} g_\theta(y_{k,\cdot}^{ij}, D_{k,\cdot}, x^{ij})}{F(y^{ij}, D, x^{ij}) - 1},\\ &\text{For } 1 \le i \le N^+,\ \nabla_{x_{ij}} J_1(\Omega) =\\ &\sum_{i=1}^{N^+} b \sum_{k=1}^{r} \nabla_{x^{ij}} g_\theta(y_{k,\cdot}^{ij}, D_{k,\cdot}, x^{ij}),\\ &\text{For } N^+ + 1 \le i \le N,\ \nabla_{x^{ij}} J_1(\Omega) =\\ &\sum_{i=N^++1}^{N} \frac{F(y^{ij}, D, x^{ij}) b \sum_{k=1}^{r} \nabla_{x^{ij}} g_\theta(y_{k,\cdot}^{ij}, D_{k,\cdot}, x^{ij})}{F(y^{ij}, D, x^{ij}) - 1} \end{aligned}$$

where $F(y^{ij}, D, x^{ij}) = (b/2)^d \exp(-bG_\theta(y^{ij}, D, x^{ij}))$. The gradient information in (18) is employed to update the optimal variables iteratively until the relative changes of the objective function value in the adjacent iterations are less than a threshold or the maximum iteration number is reached. The optimization procedures for solving (17) are summarized in Algorithms 1, 2 and 3.

Algorithm 3: Procedure for Updating Sparse Coefficients x^{ij} in (17)

Input: sample bags $\{[y^{ij}]_{j=1}^{S^i}|1 \le i \le N\}$, total sample bag number N, positive sample bag number N^+, $\{D^{t+1}, x^{ij,t}\}$ in the $(t+1)-$th and $t-$th iteration of Algorithm 1

Output: $\{[x^{ij,t+1}]_{j=1}^{S^i}|1 \le i \le N\}$

Initialization: $l \leftarrow 1, D \leftarrow D^{t+1}, x^{ij,l} \leftarrow x^{ij,t}, \alpha^0 \leftarrow 1, \alpha^1 \leftarrow 1, u^{ij} \leftarrow x^{ij,l}, L \leftarrow 0.2$

while *stopping conditions are not satisfied* **do**

  $r \leftarrow 1$

  $x^{ij,l+1} \leftarrow \mathbf{prox}_{\lambda\|\cdot\|_1}\left(u^{ij} - \frac{1}{L}\nabla_{x^{ij}} J_1(u^{ij})\right)$ **for** all i, j

  while $J_1(X^{l+1}) > J_1(U) + \nabla_X^T J_1(U)(X^{l+1} - U) + \frac{L}{2}\|X^{l+1} - U\|_F^2$ **do**

    $L \leftarrow 2^r \cdot L$

    $x^{ij,l+1} \leftarrow \mathbf{prox}_{\lambda\|\cdot\|_1}\left(u^{ij} - \frac{1}{L}\nabla_{x^{ij}} J_1(u^{ij})\right)$ **for** all i, j

    $r \leftarrow r + 1$

  end

  $\alpha^{l+1} \leftarrow \left(1 + \sqrt{4(\alpha^l)^2 + 1}\right)/2$

  $U \leftarrow X^{l+1} + \frac{\alpha^l - 1}{\alpha^{l+1}}(X^{l+1} - X^l)$

  $l \leftarrow l + 1$

  Check stopping conditions: $\frac{\|X^l - X^{l-1}\|_F}{\|X^{l-1}\|_F} < \epsilon$

end

$x^{ij,t+1} \leftarrow x^{ij,l}$ **for** all i,j

4 Implementation Details

4.1 Target Representation and Decision

Given that sparsity constraints on coefficient vectors are incorporated into the feature template learning model in (8), which en-

Table 1: Video-by-Video Success Rate. The best three results are shown in red, blue and green.

Sequence	Struck	MIL	VTD	IVT	SCM	L1T	CXT	SemiT	ASLA	MTT	Proposed Method
Woman	**0.93**	0.19	0.18	0.19	**0.87**	0.2	0.21	0.14	0.19	0.2	**0.54**
Bolt	0.02	0.01	**0.24**	0.01	0.01	0.01	0.02	**0.07**	0.01	0.01	**0.98**
Faceocc2	**1**	0.94	**0.99**	0.92	0.88	0.81	**0.95**	0.56	0.82	0.9	0.9
Faceocc1	**1**	0.77	0.93	0.98	**1**	**1**	0.78	0.71	0.31	**1**	**1**
DavidOutdoor	0.34	**0.69**	0.49	**0.64**	0.48	0.46	0.14	0.18	0.51	0.1	**0.98**
Crossing	0.96	**0.99**	0.42	0.24	**1**	0.25	0.34	0.88	**1**	0.23	0.98
Mountain-Bike	0.86	0.58	**1**	**0.99**	0.96	0.93	0.28	0.29	0.9	0.97	**0.99**
Shaking	0.17	0.23	**0.94**	0.01	**0.9**	0.04	0.12	0.01	0.39	0.01	**0.5**
Trellis	0.78	0.24	0.5	0.32	**0.85**	0.16	0.82	0.2	**0.86**	0.2	**0.96**
Car4	0.41	0.28	0.35	**1**	0.97	0.3	0.3	0.25	**1**	0.32	**1**
Sylvester	**0.93**	0.55	0.81	0.68	**0.89**	0.43	0.76	0.43	0.75	0.83	**0.98**
Lemming	**0.65**	**0.81**	0.5	0.17	0.17	0.17	**0.61**	0.15	0.17	0.37	0.38
Dudek	**0.98**	0.86	**1**	0.97	**0.98**	0.8	0.92	0.46	0.9	0.93	0.96
Car11	**1**	0.18	0.68	0.7	**1**	**1**	0.69	0.93	**1**	**1**	0.88
Subway	**0.94**	0.81	0.22	0.21	**1**	0.23	0.23	0.38	0.22	0.08	**0.88**
Soccer	0.16	0.16	**0.23**	0.17	**0.24**	**0.2**	0.13	0.07	0.13	0.18	0.17
Football	0.67	0.74	**0.78**	0.72	0.6	0.69	0.66	0.18	0.65	**0.78**	**0.78**
DavidIndoor	0.24	0.25	0.7	0.8	**0.92**	0.7	0.87	0.21	**0.96**	0.29	**0.97**
Couple	0.55	**0.67**	0.08	0.09	0.11	**0.6**	0.57	0.41	0.09	**0.61**	0.53
Doll	0.65	0.45	0.81	0.44	**0.99**	0.34	**0.98**	0.15	**0.92**	0.52	**0.92**
Average	**0.66**	0.52	0.59	0.51	**0.74**	0.47	0.52	0.33	0.59	0.48	**0.81**

Table 2: Video-by-Video Average Overlapping Rate. The best three results are shown in red, blue and green.

Sequence	Struck	MIL	VTD	IVT	SCM	L1T	CXT	SemiT	ASLA	MTT	Proposed Method
Woman	**0.73**	0.16	0.15	0.15	**0.67**	0.16	0.2	0.11	0.15	0.17	**0.5**
Bolt	0.01	0.01	**0.37**	0.01	0.02	0.01	0.02	**0.06**	0.01	0.01	**0.79**
Faceocc2	**0.79**	0.68	0.74	0.73	0.73	0.69	**0.75**	0.48	0.65	**0.75**	0.71
Faceocc1	0.73	0.6	0.69	0.73	**0.8**	**0.75**	0.64	0.57	0.32	0.7	**0.77**
DavidOutdoor	0.29	**0.54**	0.41	**0.48**	0.4	0.38	0.12	0.15	0.44	0.1	**0.69**
Crossing	0.69	**0.74**	0.32	0.31	**0.79**	0.21	0.37	0.69	**0.79**	0.2	0.71
Mountain-bike	0.71	0.46	0.7	**0.73**	0.68	**0.74**	0.23	0.23	**0.73**	**0.75**	0.68
Shaking	0.35	0.43	**0.71**	0.03	**0.69**	0.08	0.12	0.01	0.47	0.04	**0.57**
Trellis	0.62	0.25	0.46	0.26	**0.68**	0.2	0.66	0.2	**0.8**	0.22	**0.69**
Car4	0.5	0.26	0.36	**0.88**	**0.76**	0.25	0.31	0.23	**0.76**	0.45	**0.87**
Sylvester	**0.73**	0.53	0.62	0.52	**0.69**	0.41	0.6	0.34	0.6	0.65	**0.74**
Lemming	**0.48**	**0.65**	0.44	0.14	0.14	0.14	**0.46**	0.12	0.15	0.29	0.35
Dudek	0.73	0.71	**0.8**	0.75	**0.77**	0.69	0.73	0.38	0.74	0.76	**0.77**
Car11	**0.9**	0.2	0.55	0.67	**0.85**	**0.89**	0.57	0.84	**0.85**	0.83	0.65
Subway	**0.66**	**0.66**	0.16	0.17	**0.73**	0.16	0.18	0.29	0.19	0.07	0.63
Soccer	0.19	0.17	**0.33**	0.16	**0.24**	0.17	0.13	0.07	0.11	0.18	**0.3**
Football	0.55	**0.59**	0.57	0.56	0.49	0.56	0.55	0.15	0.54	**0.58**	**0.59**
DavidIndoor	0.24	0.43	0.56	0.65	**0.73**	0.54	0.65	0.25	**0.75**	0.3	**0.69**
Couple	**0.54**	**0.5**	0.07	0.07	0.1	0.47	**0.49**	0.35	0.08	**0.49**	0.46
Doll	0.53	0.47	0.65	0.44	**0.83**	0.45	**0.75**	0.12	**0.83**	0.39	0.7
Average	**0.55**	0.45	0.48	0.42	**0.59**	0.4	0.43	0.28	0.5	0.4	**0.64**

Table 3: Video-Set-Based Comparison in Terms of Success Rate and Precision.

	Struck	MIL	VTD	IVT	SCM	L1T	CXT	SemiT	ASLA	MTT	Proposed Method
Success Rate	**0.541**	0.445	0.475	0.414	**0.577**	0.392	0.419	0.354	0.488	0.390	**0.636**
Precision	**0.720**	0.554	0.614	0.576	**0.751**	0.496	0.565	0.431	0.591	0.457	**0.862**

ables each template to capture different distinctive properties of the target, to better utilize the representation power of the learned feature templates, we adopt the sparse representation scheme for target representation for the sake of its robustness and effectiveness (Zhang et al. 2013a) (Lan, Ma, and Yuen 2014) (Lan et al. 2015). To further enhance the adaptivity, the learned feature templates are augmented with recently obtained important samples, which are denoted by D' and updated by the same scheme in (Mei and Ling 2011). Given the learned feature templates D' and M target candidates $\{x_i\}_{i=1}^{M}$ with their states $\{s_i\}_{i=1}^{M}$ sampled by particle filtering, the sparse representations of the target candidates can be obtained via solving the following problem:

$$w^i = \arg\min_{w} \|x_i - D'w\|_2^2 + \lambda_1 \|w\|_1 \tag{19}$$

where λ_1 is the tradeoff between the reconstruction error and the sparseness. After the sparse coefficients of each target candidate with respect to the feature template are obtained, the target state at frame t, denoted as S^t can be decided as follows:

$$S_t = s_k \tag{20}$$
$$\text{s.t. } k = \arg\min_{i} \|x_i - Dw_i\|_2^2$$

5 Experiments

The section describes the experimental setting, and reports the quantitative and qualitative experimental results, respectively.

5.1 Experimental Setting

Twenty publicly available image sequences, which cover various kinds of challenging scenarios, e.g. occlusion, abrupt illumination changes, large pose variations, etc., are used for evaluation. The proposed tracker is compared with ten state-of-the-art trackers, i.e. semi-supervised learning-based tracker: SemiB (Grabner, Leistner, and Bischof 2008), multiple instance learning-based tracker: MIL (Babenko, Yang, and Belongie 2011), sparsity-based trackers which explicitly models the noise/outliers: L1T (Mei and Ling 2011), MTT (Zhang et al. 2013b), SCM (Zhong, Lu, and Yang 2014), ASLA (Jia, Lu, and Yang 2012), feature learning-based method: IVT (Ross et al. 2008), and other state-of-the-arts: VTD (Kwon and Lee 2010), CXT (Dinh, Vo, and Medioni 2011), STRUCK (Hare, Saffari, and Torr 2011). The source codes provided by the authors are used, and they are set with the same initialization parameters for fair comparison.

We empirically set β in (8) to be 0.01, λ in (8) to be 0.01, λ_1 in (19) to be 0.01, b in (8) to be 1, c in (8) to be 1, the number of

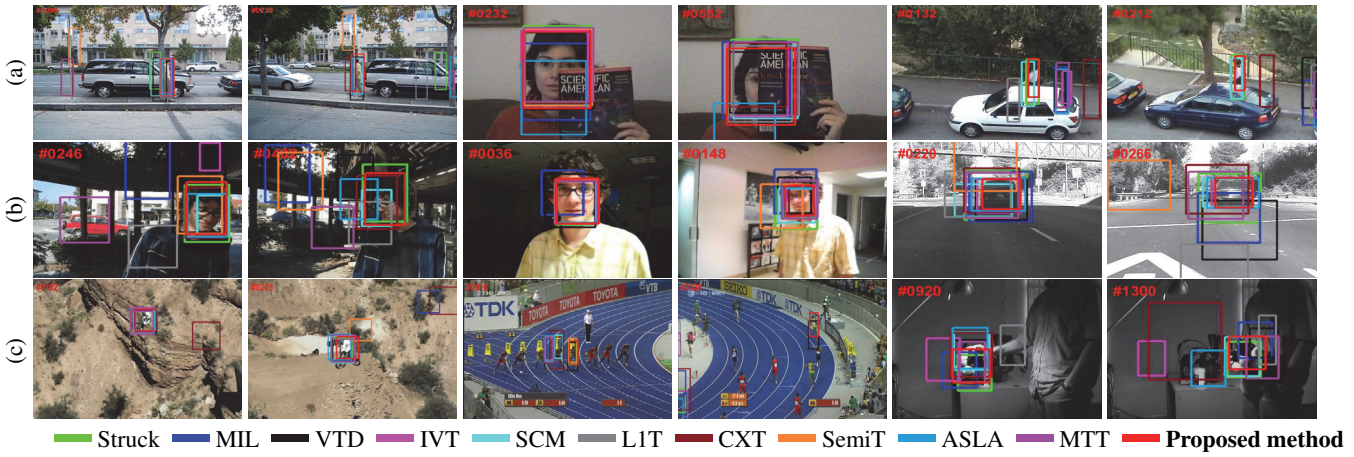

Figure 2: Qualitative results on some typical frames of several videos with some challenging factors. (a) Occlusion (*DavidOutdoor, Faceocc, Woman*). (b) illumination (*Trellis, DavidIndoor, Car4*). (c) pose and cluttered background (*Mountain-bike, Bolt, Sylvester*).

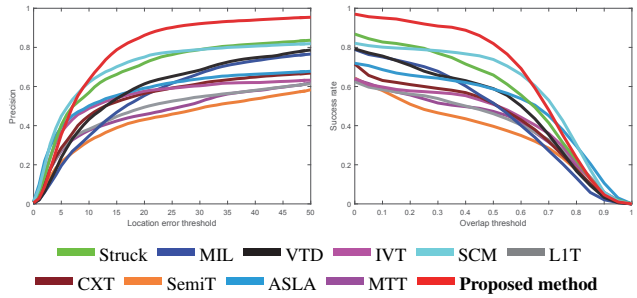

Figure 3: Video-set-based comparison of the 11 trackers on the whole video set in terms of precision (left) and success rate (right)

the learned feature templates to be 20. In every frame, 8 potentially positive samples and 50 samples in the first frame compose the positive bag for training, which means N^+ in (8) is set to 1 , while 10 samples are used to construct 10 negative bags. To obtain the features of each sample for model learning, we use the grey scale feature of 8-by-8 down-sampled image patch, extract HOG features, and then concatenate them into a single feature vector. The feature templates are initialized using K-SVD algorithm (Aharon, Elad, and Bruckstein 2006).

5.2 Experimental Results

We quantitatively evaluate the proposed tracker from two aspects: video-by-video comparison and video set-based comparison. For video-by-video quantitative comparison, two widely accepted metrics: success rate and average overlapping rate are adopted. The overlapping rate is defined as $\frac{area(B_T \bigcap B_G)}{area(B_T \bigcup B_G)}$ where B_G and B_T are the bounding boxes of the ground-truth and the tracker. A success of a tracking result means the overlapping rate is larger than 0.5. Table 2 and Table 1 record the overlapping rate and success rate respectively. The quantitative results show that the proposed tracker outperforms most of other trackers on most videos in terms of average overlapping rate and success rate. The average overlapping rate ranks in top three on 13 videos while the success rate ranks in top three on 12 videos. Since the proposed tracker runs stably on these videos, it achieves the best average performance in terms of average overlapping rate and success rate. In particular, superior performances are achieved by the proposed tracker on some videos which cover pose variation (e.g. *Sylvester, DavidIndoor, Bolt*), large illumination variations(e.g. *Trellis, Car4*), cluttered background (e.g. *Shaking, Football*), occlusion (e.g. *Faceocc1, DavidOutDoor*). etc.. This is because the feature template learning model, which explicitly models the contaminated features, make it more effective to deal with corrupted samples caused by large illumination variations, occlusion, etc.. In addition, by learning the tracking model using weakly-labeled samples organized in the form of sample bags, the proposed tracker is less sensitive to misaligned samples which is usually caused by some pose variations, rotation, etc..

To reduce the risk that performance evaluation is trapped by the peculiarity of single video, video-set based evaluation is performed under two evaluation methods: precision plot and success rate. The precision plot is defined based on center location error (CLE) which measures the Euclidean distance between the centers of ground-truth and bounding box. The precision plot shows the percentage of frames where the CLE is within a given threshold which changes from 0 to 50. The success rate plot shows percentage of success frames whose overlapping rate is larger than a given threshold changing from 0 to 1. The performance scores for precision plot and success rate plot is defined as the precision with threshold 20 and the average success rate, respectively. Figure 3 illustrates the precision plot and success rate plot for the evaluation video set. We can see that the proposed tracker maintain a higher precision and success rate than those of all other trackers in most cases. Table 3 records the overall performance scores, demonstrating that the proposed tracker outperforms other trackers.

Figure 2 demonstrates some qualitative results on some typical frames which covers occlusion, clutter background, variations in illumination and pose. We see that the proposed tracker runs stably without drifting in most cases, e.g. large illumination variations (e.g. Car4#220, Trellis#246), occlusion (e.g. David#238, woman#212), pose variations (e.g. Bolt#18), which shows the effectiveness of the proposed feature template learning model in appearance modeling.

6 Conclusion

This paper proposes a novel MIL-based feature template learning for object tracking. By explicitly modeling the contaminated samples and resolving the label ambiguity within a probabilistic multiple instance learning framework, the proposed model is able to effectively perform model updating with corrupted samples and alleviate the tracking drift problem. Comparison experiment with other ten state-of-the-art trackers show its effectiveness.

Acknowledgements

This work was supported in part by Hong Kong RGC General Research Fund HKBU12254316. The authors would like to thank the anonymous reviewers for their suggestions on improving the quality of this paper.

References

Aharon, M.; Elad, M.; and Bruckstein, A. 2006. K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. Signal Process.* 54(11):4311–4322.

Babenko, B.; Yang, M.; and Belongie, S. 2011. Robust object tracking with online multiple instance learning. *IEEE Trans. Pattern Anal. Mach. Intell.* 33(8):1619–1632.

Dinh, T. B.; Vo, N.; and Medioni, G. 2011. Context tracker: Exploring supporters and distracters in unconstrained environments. In *Proc. CVPR*, 1177–1184.

Fan, B.; Du, Y.; Gao, H.; and Wang, B. 2014. Online discriminative dictionary learning via label information for multi task object tracking. In *Proc. ICME*, 1–6.

Grabner, H.; Leistner, C.; and Bischof, H. 2008. Semi-supervised on-line boosting for robust tracking. In *Proc. ECCV*, 234–247.

Hare, S.; Saffari, A.; and Torr, P. H. 2011. Struck: Structured output tracking with kernels. In *Proc. ICCV*, 263–270. IEEE.

Jia, X.; Lu, H.; and Yang, M.-H. 2012. Visual tracking via adaptive structure local sparse model. In *Proc. CVPR*, 1822–1829.

Kwon, J., and Lee, K. M. 2010. Visual tracking decomposition. In *Proc. CVPR*, 1269–1276.

Lan, X.; Ma, A. J.; Yuen, P. C.; and Chellappa, R. 2015. Joint sparse representation and robust feature-level fusion for multi-cue visual tracking. *IEEE Trans. Image Process.* 24(12):5826–5841.

Lan, X.; Ma, A. J.; and Yuen, P. C. 2014. Multi-cue visual tracking using robust feature-level fusion based on joint sparse representation. In *Proc. CVPR*, 1194–1201.

Lan, X.; Zhang, S.; and Yuen, P. C. 2016. Robust joint discriminative feature learning for visual tracking. In *Proc. IJCAI*, 3403–3410.

Li, X.; Shen, C.; Shi, Q.; Dick, A. R.; and van den Hengel, A. 2012. Non-sparse linear representations for visual tracking with online reservoir metric learning. In *Proc. CVPR*, 1760–1767.

Li, H.; Li, Y.; and Porikli, F. 2014. Deeptrack: Learning discriminative feature representations by convolutional neural networks for visual tracking. In *Proc. BMVC*.

Liu, R.; Lan, X.; Yuen, P. C.; and Feng, G. C. 2016. Robust visual tracking using dynamic feature weighting based on multiple dictionary learning. In *Proc. EUSIPCO*, 2166–2170.

Mei, X., and Ling, H. 2011. Robust visual tracking and vehicle classification via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* 33(11):2259–2272.

Mei, X.; Ling, H.; Wu, Y.; Blasch, E. P.; and Bai, L. 2013. Efficient minimum error bounded particle resampling L1 tracker with occlusion detection. *IEEE Trans. Image Process.* 22(7):2661–2675.

Nesterov, Y. 2005. Smooth minimization of non-smooth functions. *Math. Prog.* 103(1):127–152.

Nesterov, Y. 2013. Gradient methods for minimizing composite functions. *Math. Prog.* 140(1):125–161.

Ross, D. A.; Lim, J.; Lin, R.; and Yang, M. 2008. Incremental learning for robust visual tracking. *Int. J. Comput. Vis.,* 77(1-3):125–141.

Shrivastava, A.; Pillai, J. K.; Patel, V. M.; and Chellappa, R. 2014. Dictionary-based multiple instance learning. In *Proc. ICIP*, 160–164.

Shrivastava, A.; Patel, V. M.; Pillai, J. K.; and Chellappa, R. 2015. Generalized dictionaries for multiple instance learning. *Int. J. Comput. Vis.* 114(2-3):288–305.

Wright, J.; Ma, Y.; Mairal, J.; Sapiro, G.; Huang, T. S.; and Yan, S. 2010. Sparse representation for computer vision and pattern recognition. *Proceedings of the IEEE* 98(6):1031–1044.

Zeisl, B.; Leistner, C.; Saffari, A.; and Bischof, H. 2010. On-line semi-supervised multiple-instance boosting. In *Proc. CVPR*, 1879–1879.

Zhang, K., and Song, H. 2013. visual tracking via weighted multiple instance learning. *Pattern Recognit.* 46(1):397–411.

Zhang, S.; Yao, H.; Sun, X.; and Lu, X. 2013a. Sparse coding based visual tracking: Review and experimental comparison. *Pattern Recognit.* 46(7):1772–1788.

Zhang, T.; Ghanem, B.; Liu, S.; and Ahuja, N. 2013b. Visual tracking using structured multi-task learning. *Int. J. Comput. Vis.* 101(2):367–383.

Zhang, S.; Lan, X.; Yao, H.; Zhou, H.; Tao, D.; and Li, X. 2016. A biologically inspired appearance model for robust visual tracking. *IEEE Trans. Neural Netw. Learn. Syst.* DOI:10.1109/TNNLS.2016.2586194.

Zhang, C.; Platt, J. C.; and Viola, P. A. 2005. Multiple instance boosting for object detection. In *Proc. NIPS*, 1417–1424.

Zhao, C.; Wang, X.; and Cham, W. 2011. Background subtraction via robust dictionary learning. *EURASIP J. Image Video Process.* 2011.

Zhong, W.; Lu, H.; and Yang, M. 2014. Robust tracking via sparse collaborate appearance model. *IEEE Trans. Image Process.* 23(5):2356–2368.