Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2014, Article ID 189317, 12 pages
http://dx.doi.org/10.1155/2014/189317

Research Article

Online Learning Discriminative Dictionary with Label Information for Robust Object Tracking

Baojie Fan,[1] Yingkui Du,[2] and Yang Cong[2]

[1] *College of Automation, Nanjing University of Posts and Telecommunications, Nanjing 210023, China*
[2] *State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China*

Correspondence should be addressed to Baojie Fan; jobfbj@gmail.com

Received 7 March 2014; Revised 3 June 2014; Accepted 5 June 2014; Published 24 June 2014

Academic Editor: Caihong Li

A supervised approach to online-learn a structured sparse and discriminative representation for object tracking is presented. Label information from training data is incorporated into the dictionary learning process to construct a robust and discriminative dictionary. This is accomplished by adding an ideal-code regularization term and classification error term to the total objective function. By minimizing the total objective function, we learn the high quality dictionary and optimal linear multiclassifier jointly using iterative reweighed least squares algorithm. Combined with robust sparse coding, the learned classifier is employed directly to separate the object from background. As the tracking continues, the proposed algorithm alternates between robust sparse coding and dictionary updating. Experimental evaluations on the challenging sequences show that the proposed algorithm performs favorably against state-of-the-art methods in terms of effectiveness, accuracy, and robustness.

1. Introduction

Given the initialized position and size of a target in the first frame (or former frames) of a video, the goal of visual tracking is to estimate the states of the moving target in the subsequent frames. This active topic has been extensively studied in computer vision due to its important role in many applications such as automated surveillance, robot navigation, video indexing, traffic monitoring, and human-computer interaction. Despite the fact that much progress has been made in recent years [1–5], developing a robust tracking algorithm is still a challenging problem due to numerous factors: illumination, partial or full occlusions, dynamic appearance changes, scaling, abrupt motion, background clutters, pose variation, and shape deformation.

Inspired by the success of sparse representation-based face recognition [6], Mei and Ling [7] propose a novel *L1* tracker that uses a series of target templates and trivial ones to model the tracked target with the sparse constraints. In detail, the target templates are used to describe the tracked object and trivial templates are used to deal with outliers (such as occlusion). This representative scheme is robust to a wide range of image corruptions, especially moderate occlusions. Based on the milestone work, some extensions [6–20] are developed to improve the *L1* tracker in terms of both speed and accuracy. However, sparse representation-based approaches have some drawbacks. First, previous tracking algorithms construct the dictionaries naive. They directly use the sampled samples from tracked target region and its background as the dictionary atoms; they are not selected. This operation makes the dictionary redundant and ignores the discriminative and structured information from the initial training data. Second, some methods use either static dictionaries during tracking process [10] or heuristic dictionary update. Finally, many sparse coding-based trackers [6–15] seek to minimize the reconstruction error with *L2* norm to increase the representative power but ignore the discriminative ability of the learned dictionary. The data term with *L2* norm does not use the robust function in the data fitting term and might be vulnerable to large outliers and makes the tracking unstable.

In this paper, we formulate object tracking in a particle filter framework as a binary classification problem. The a priori information from training data is exploited effectively to online-learn a discriminative and reconstructive dictionary. Specifically, the class label information is incorporated into the dictionary learning process as the classification error term and idea coding regularization term, respectively. Combined with the traditional reconstruction error, a total objective function for dictionary learning is constructed with *L1* data fitting term. By minimizing the total object function, we can obtain a high quality dictionary and optimal linear classifier jointly using iterative reweighed least squares algorithm. With the help of robust sparse coding, the optimal classifier can separate the tracked object from background effectively.

The main contributions of this paper are the following.

(1) The a priori information from the training samples is exploited to construct a compact and discriminative dictionary. The learned dictionary deserves the structure information from training samples and encourages samples from the same class to have similar representations. It is a critical factor for the object tracker-based sparse representation.

(2) Learning a high quality dictionary and optimal linear classifier are accomplished jointly. All the training samples from the object and background are involved in the dictionary learning process simultaneously.

(3) Many existing sparse coding-based trackers do not use robust function in the data fitting term and might be vulnerable to large outliers. *L1* norm fitting function is incorporated into the data fitting term to overcome this problem and make the tracking reliable.

The paper is organized as follows. In Section 2, we summarize the works most related to ours. Section 3 presents the *L1* tracker and dictionary learning as the background to facilitate the introduction of our proposed model in the next section. The detailed description of the proposed tracking approach is presented in Section 4. Section 5 gives the detailed experiment setup and results. Finally, Section 6 concludes the paper.

2. Related Work

Much work has been done in object tracking. In this section, we only briefly review nominal tracking methods and those that are the most related to our tracker. We focus specifically on tracking methods that use particle filters, sparse representation, and general multitask learning methods. For a more thorough survey of tracking methods, we refer the readers to [1–5].

Existing tracking algorithms can be roughly categorized as either generative or discriminative.

2.1. The Generative Trackers. The generative methods represent the target as an appearance model. The tracking problem is formulated as searching for the regions which are the most similar to the tracked targets. These methods are based on either templates or subspace models. Popular generative trackers include eigentracker [21], mean shift tracker [22], fragment-based tracker [23], incremental tracker (IVT) [24], and VTD tracker [25]. Black and Jepson [21] learn a subspace model offline to represent target at predefined views and build on the optical flow framework for tracking. The mean shift tracker [22] is a popular mode-finding method, which successfully copes with camera motion, partial occlusions, clutter, and target scale variations. The fragment tracker [23] aims to solve partial occlusion with a representation based on histograms of local patches. The tracking task is carried out by accumulating votes from matching local patches using a template. But, this template is static, and it can not handle changes in object appearance. Ross et al. [24] learn an adaptive linear subspace online for modeling target appearance and implement tracking with a particle filter. However, IVT is less effective in handling heavy occlusion or nonrigid distortion. Kwon and Lee [25] extend the classic particle filter framework with multipledynamic observation models to account for appearanceand motion variation. Nevertheless, due to the adopted generative representation scheme, this tracker is not equipped to distinguish between the target and its local background.

2.2. Discriminative Trackers. Discriminative methods cast the tracking as a classification problem that distinguishes the tracked targets from their surrounding backgrounds. The trained classifier is online updated during the tracking procedure. Discriminative tracking algorithms use the information from both the target and the background. Examples of discriminative methods are ensemble tracking [26], online boosting (OAB) [27], semionline boosting [28], online multiple instance learning tracking [29], adaptive metric differential tracking [30], TLD [31], and CT [32].

In ensemble tracking [13], a set of weak classifiers are trained and combined for distinguishing the target object and the background. The features used in [26] may contain redundant and irrelevant information which affects the classification performance. To improve the classification performance, feature selection is needed. Collins et al. [33] have demonstrated that online selecting discriminative features can greatly improve the tracking performance. Inspired by the advances in face detection [34], many boosting feature selection methods have been proposed. Grabner et al. [27] propose an online boosting algorithm to select features for tracking. However, these trackers [27, 33] only use one positive sample (i.e., the current tracker location) and a few negative samples when updating the classifier. As the appearance model is updated with noisy and potentially misaligned examples, this often leads to the tracking drift problem. To better handle visual drift, Grabner et al. [28] propose an online semisupervised tracker which only labels the samples in the first frame while leaving the samples in the sequent frames unlabeled. However, this semisupervised approach discards some useful information which is very helpful in the problem domain. Babenko et al. [29] introduce multiple instance learning into online tracking where samples are considered within positive and negative bags or sets. Within the multiple instances learning (MIL) framework, several tracking algorithms have been developed [30, 35–38] in order to handle location ambiguities of positive samples for object tracking or actively select discriminative feature. Besides, Kalal et al. [31] propose the PN learning algorithm to exploit the underlying structure of positive and negative samples to learn effective classifiers for object tracking. Recently, an efficient tracking algorithm [32] based on compressive sensing theory [39] is proposed, which demonstrates that the low dimensional features randomly extracted from the high dimensional multiscale image feature space can preserve the discriminative capability, thereby facilitating object tracking.

2.3. Sparse Representation for Object Tracking. Sparse representation has been successfully applied to visual tracking [6]. Its metric is according to finding the best candidate with minimal reconstruction error using target templates and trivial ones. Most of these object tracking algorithms are in the particle filter framework. For each particle, its representation is computed independently by solving a constrained *L1* minimization problem with nonnegativity constraints, so hundreds of *L1* norm related minimization problems need to be solved for each frame during the tracking process. Besides, the solver for the *L1* norm minimizations used in [7, 8] is based on the interior point method which turns out to be too slow for tracking. A minimal error bounding strategy is introduced [8] to reduce the number of particles, equal to the number of the *L1* norm minimizations for solving. A speed-up by four to five times is reported in [8]. In [9], APG-based solution is used to improve the *L1* tracker. Liu et al. [10] integrate the dynamic group sparsity into the tracking problem and high dimensional image features are used to improve tracking robustness. Liu et al. [11] also develop a tracking algorithm based on local sparse model which employs histograms of sparse coefficients and the mean shift algorithm for object tracking. However, this method is based on a static local sparse dictionary and may fail when there is a similar object in the scenes. In Li et al. [14], dimensionality reduction and a customized orthogonal matching pursuit algorithm are adopted to accelerate the *L1* tracker. In [15], the authors propose a robust object tracking algorithm using a collaborative model that combines a sparsity-based discriminative classifier (SDC) and a sparsity-based generative model (SGM), but it adopts the naive model updating strategy and similar metric measure; this will affect the performance of the tracker. In [16], the authors develop a simple yet robust tracking method based on the structural local sparse appearance model. Its representation exploits both partial information and spatial information of the target based on a novel alignment-pooling method. In Zhang et al. [17], low-rank sparse learning is adopted to consider the correlations among particles for robust tracking. Inspired by these works, he develops the multitask tracking (MTT) algorithm [18]. However, the dictionary still includes the trivial templates; they will degrade the efficiency and effectiveness of the tracker.

3. Background

In this section, we briefly introduce the *L1* tracker and dictionary learning to facilitate the presentation of our model in the next section.

3.1. L1 Tracker. *L1* tracker and most of its extension are in the particle filter framework. Its metric is according to finding the best candidate with minimal reconstruction error using target templates and trivial ones. In each frame, *L1* tracker first generates candidate particles with the current tracking result. For each particle, its representation is computed independently by solving a constrained *L1* minimization problem with nonnegativity constraints. To adapt the appearance changes of an object, the template is updated according to both the weights assigned to each template and the similarity between templates and current estimation of target candidate.

L1 tracker can be viewed as a sparse coding process with the given dictionary (object templates and trivial ones). But *L1* and its extensions ignore the dictionary quality; they only adopt a simple strategy to construct the dictionary: take the entire positive (or negative) training set as dictionary. Sparse coding with a large dictionary is computationally expensive.

3.2. Dictionary Learning. The goal of dictionary learning is to find the optimized dictionaries that provide the representation for most statistically representative input signals. Let $Y = [y_1, y_2, \ldots, y_N] \in R^{n\times N}$ be a set of N input signals, where y_i denotes the ith input signal with n dimensional feature description. Learning a reconstructive dictionary with size K for sparse representation can be obtained by solving the following minimization problem:

$$\langle D^*, X^* \rangle = \underset{D,X}{\arg\min} \sum_{i=1}^{N} \left(\left\| y_i - D x_i \right\|_2^2 \right) + \left(\lambda \left\| x_i \right\|_1 \right), \quad (1)$$

where $D = [d_1, d_2, \ldots, d_K] \in R^{n\times K}$ is the learned dictionary and $X = [x_1, x_2, \ldots, x_N] \in R^{K\times N}$are the sparse codes of input signals. In general, the number of training samples is larger than the size of D ($N \gg K$), and x_i only uses a few dictionary atoms for its representation under the sparsity constraint. Usually, the above objective function is iteratively optimized in a two-stage manner, by alternatively optimizing with respect to D (bases) and X (coefficients) while holding the other fixed. Each stage is convex in D (while holding X fixed) and in X (while holding D fixed) but not convex in both simultaneously. The objective function in (2) only focuses on minimizing the reconstruction error and does not consider the discriminative power of a dictionary. Hence, some supervised approaches [40–47] have been proposed to improve the discriminative power of dictionary, by integrating the category label information into the objective function of dictionary learning.

4. The Proposed Tracker

Many existing online dictionary learning methods do not use the robust function in the data fitting term and might be vulnerable to large outliers. In robust statistics, *L1* fitting functions are found useful to make estimation reliable. During the process of object tracking, the challenging factors such as occlusion, illumination changes, abrupt motion, and background clutters are usually regarded as the outlier. If the *L2* norm data fitting is adopted for sparse representation-based tracker, the drift will be cumulative and result in tracking failure. However, *L1* fitting functions can overcome the above problem and make tracking reliable. Inspired by the above work [40–47], an approach to online-learn a structured sparse and discriminative representation for object tracking is presented in this section.

4.1. The Total Object Function. We construct a robust object function for online dictionary learning and the optimal classifier. To be concrete, the total objective function for the proposed tracker is defined as

$$\langle D^*, A^*, X^* \rangle = \arg\min_{D,A,X} \|Y - DX\|_1^1 + \lambda_1 \|H - AX\|_1^1 + \lambda_2 \|Q - X\|_1^1 + \lambda_3 \|X\|_1, \qquad (2)$$

where parameters $\lambda_1, \lambda_2, \lambda_3$ control the relative weight of three terms: reconstruction error term, classification error term, idea coding regularization term and norm regularization term.

Reconstruction Error Term $\|Y - DX\|_1^1$. This data fitting is robust compared with *L2* norm and can handle some outliers such as part occlusion and background in the train data. We compute the reconstruction errors of all the particles with the learned dictionary items at the same time.

Ideal Structured Regularization Term $\|Q - X\|_1^1$. This term includes the information from training samples. Q is the idea representation for Y, $Q = [q_i, q_2, \ldots, q_M] \in R^{K\times M}$. M is the number of training samples. We hope that X is very close to Q and force the samples from the same class to have similar discriminative sparse representation without losing structure information. q_i is the sparse code of an input signal y_i with the form $q_i = [q_i^1, q_i^2, \ldots, q_i^K]' = [0, \ldots, 1, 1, 1, \ldots]' \in R^K$. We cast the object tracking can be viewed as a binary classification problem: object (class T) and background (class B). If the training samples are sampled from the tracked object region, the coefficients in q_i for class T are all 1 s, while the others are all 0 s. For example, the training samples $Y = [y_1, y_2, y_3, y_4]$ include two classes: y_1, y_2 belong to object T and y_3, y_4 are from background B; the ideal representation Q for Y is as follows:

$$Q = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}. \qquad (3)$$

Classification Error Term $\|H - AX\|_1^1$. The term measures theclassification error, and it supports learning an optimal classifier. For object tracking task, we define two classes: tracked object and background. A simple linear classifier $f(A; X) = AX$ is adopted, where A is the classifier parameters. $H = [h_1, h_2, \ldots, h_N] \in R^{2\times N}$ is the class labels of training data Y. $h_i = [1, 0]^t$ is the corresponding label vector of y_i, and the nonzero position indicates the class label of y_i.

L1 Norm Regularization Term $\|X\|_1$. By adding a sparseness criterion into the objective function (2), we are able to learn a sparse and structural representation with the learned high-quality dictionary D_t. The proposed tracker is under the particle filter framework. The candidate particles are densely sampled around the current tracking target and their representations will be sparse and similar with respect to the given dictionary D_t. In other words, a few items in D_t are required to represent all the particles.

4.2. Optimization Procedure. To solve optimization problem in (2), we rewrite the proposed object function as follows.

Dictionary learning:

$$\langle D^*, A^* \rangle = \arg\min_{D,A} \left\|Y_{L1} - D_{L1}X\right\|_1^1 + \lambda_2 \|Q - X\|_1^1 + \lambda_3 \|X\|_1, \qquad (4)$$

where $Y_{L1} = [Y, \sqrt{\lambda_1}H]'$, $D_{L1} = [D, \sqrt{\lambda_1}A]'$.

Sparse coding:

$$X^* = \arg\min_{X} \left\|Y_{L2} - D_{L2}X\right\|_1^1 + \lambda_3 \|X\|_1, \qquad (5)$$

where $Y_{L2} = [Y, \sqrt{\lambda_1}Q, \sqrt{\lambda_2}H]'$ and $D_{L2} = [D, \sqrt{\lambda_1}I, \sqrt{\lambda_2}A]'$.

As in [27], iterative reweighed least squares algorithm (IRLS) is used to obtain the optimal solutions of (4) and (5). It solves the above two problems in each iteration until convergence.

Given the initial dictionary D_0, we can obtain the robust sparse coding X^* by (5). Combining D_0 and X^*, (4) can be regarded as a *L1* regression problem. The IRLS algorithm can be used to solve (4) with the known X^* and Q:

$$D_{L1}(j,:) = \arg\min_{d_{L1}} \frac{1}{n} \sum_{i=1}^{n} w_i^j \left(Y_{L1}(i,j) - d_{L1}X_i\right)^2, \qquad (6)$$

where $w_i^j = 1/\sqrt{(Y_{L1}(i,j) - d_{L1}X_i)^2 + \delta}$and δ is a small positive value. By taking derivatives for (6) and setting them to zeros, the global optimum can be reached by solving $D_{L1}(j,:)$ in the linear system

$$C^j = \sum_{i=1}^{n} w_i^j Y_{L1}(i,j) X_i^T, \qquad (7)$$

$$M^j = \sum_{i=1}^{n} w_i^j X_i X_i^T, \qquad C^j = D(j,:) M^j. \qquad (8)$$

As the trace continues, the dictionary updates with the coming data, the online versions of C^j, M^j is as follows:

$$C_N^j = C_n^j + \sum_{i=n+1}^{N} w_i^j Y_{L1}(i,j) X_i^T, \tag{9}$$

$$M_N^j = M_n^j + \sum_{i=n+1}^{N} w_i^j X_i X_i^T, \tag{10}$$

where C_n^j and M_n^j are the former data, the second terms in both (9) and (10) are the coming data.

We have learned the dictionary $D_{L1} = [D, \sqrt{\lambda_1}A]'$. For all the particles, we first compute their robust sparse codes X^* from (5) and then obtain the classification score of the particles from the optimal classifier A. The tracking is completed by the following equation:

$$\overline{X}_i = \arg\max_{X_i}(AX), \tag{11}$$

where x_i is the sparse coding of each particle with learned dictionary D. The sparse codings of all the particles form the matrix X.

4.3. Tracking Algorithm Details. For initialization, we manually choose the foreground object with the bounding box and then shift it by a few pixels to generate the positive samples. Besides, we shift the bounding box far away from the object location to generate the negative samples, which are without overlap with positive samples. The K-SVD algorithm is executed on positive and negative samples separately to learn the initial dictionary. The proposed tracking algorithm is under the particle filter framework, which recursively approximates the posterior distribution using a finite set of weighted samples. It consists of two steps: prediction and update.

At the frame specially, let affine parameters $X = (x, y, s, r, \theta, \lambda)$ represent the target state, where x and y are the coordinates, s and r are the scale and the aspect, θ is the rotation angle, and λ is the skew. $Y_{1:t-1} = \{Y_1, Y_2, \ldots, Y_{t-1}\}$ denotes the observation of the target from the first frame to the frame $t-1$. Particle filters tracking estimates and propagates the probability by recursively performing prediction,

$$p(X_t \mid Y_{1:t-1}) = \int p(X_t \mid X_{t-1})\, p(X_{t-1} \mid Y_{1:t-1})\, dX_{t-1} \tag{12}$$

and updating

$$p(X_t \mid Y_{1:t}) = \frac{p(Y_t \mid X_t)\, p(X_t \mid Y_{1:t-1})}{p(Y_t \mid Y_{1:t-1})}. \tag{13}$$

The optimal state for the frame t is obtained according to the maximal approximate posterior probability:

$$X_t^* = \arg\max_{X_t} p(X \mid Y_{1:t}) = \arg\max(AX). \tag{14}$$

Table 1: All the tested image sequences.

Video sequence	Total frame	Challenging factor
Animal	71	Abrupt motion and background clutter
Car11	393	Illumination variation, background clutter, and scale and pose change
Girl	501	Occlusion and scale and pose change
Jumping	313	Abrupt motion
Cliffbar	471	Scale change, background clutter, and abrupt motion
Caviar	500	Occlusion, scale change, and background clutter
Football	362	Similar object and background clutter
Woman	550	Occlusion, scale change, and similar object

This inference is governed by the model $p(X_t \mid X_{t-1})$, which describes the temporal correlation of the tracking results in consecutive frames, and it is modeled to be Gaussian with the dimensions of X_t assumed independent. The observation model $p(Y_t \mid X_t)$ reflects the similarity between a target candidate and dictionary templates. In this paper, $p(Y_t \mid X_t)$ is proportional to the classifier scores.

5. Experiments

In this section, we make a thorough comparison on challenging image sequences between our proposed trackers and state-of-the-art tracking methods. Our trackers are evaluated on 8 challenging tracking sequences (e.g., car11, cliffbar, and woman sequences) that are publicly available online. Table 1 lists all the evaluated image sequences; these videos are recorded in indoor and outdoor environments and include the abovementioned challenging factors in visual tracking. We evaluate the proposed tracker against ten state-of-the-art visual tracking algorithms including: ONND [12], LSST [13], SCM [15], ASLA [16], MTT [18], CT [32], VTD [25], MIL [29], PN [31], IVT [24], and *L1* [6]. These trackers are implemented using publicly available source codes or binaries provided by the authors. They are initialized using their default parameters.

5.1. Parameter Setting. The proposed algorithm is implemented in MATLAB R2011b on a Pentium 2.3 GHz Dual Core laptop with 2 GB memory. For each sequence, the location of the target object is manually labeled in the first frame. Each image sample from the target and background is normalized to a 32 × 32 patch. We set the parameters $\lambda_1, \lambda_2, \lambda_3$ in (5) to be 2, 4, and 0.01, respectively. The parameter β in (10) is set to 0.01, and $\sigma = 0.95$. The numbers of positive templates and negative templates are 200 and 600, respectively. The learned dictionary includes 200 items.

5.2. Quantitative Comparison. For quantitative performance comparison, two popular evaluation criteria are used, namely, center location error (CLE) and tracking success rate (TSR). The CLE is computed as the distance between the predicted

Table 2: Average center location error (in pixel). The best three results are marked by *, †, and ‡.

	Animal	Car11	Girl	Jumping	Cliffbar	Caviar	Football	Woman
IVT	127.5	2.1058	48.4739	36.8024	24.8112	65.9575	13.61	173.28
L1	171.4	33.252	62.4351	92.3931	49.6003	65.6717	18.17	130.615
PN	25.65	25.113	23.1583	3.5891^*	11.2504	44.4463	13.54	17.9335
VTD	11.92	27.055	21.4425	62.9881	34.5553	58.2016	$4.300^\ddagger$	118.49
MIL	66.46	43.465	32.2088	9.8941	13.3477	100.186	13.66	124.51
Frag	92.09	63.922	18.0463	58.4481	48.6741	116.06	17.21	100.41
MTT	15.86	2.8024	23.8883	34.4735	46.1711	64.9936	9.842	134.02
SCM	10.02	2.5200	10.0169^*	$4.0973^\ddagger$	$7.7062^\ddagger$	$2.8980^\dagger$	$3.899^\dagger$	143.59
ASLA	7.284^*	$1.7824^\ddagger$	$16.1827^\ddagger$	4.2797	$5.6058^\dagger$	5.5937	14.94	2.42^*
CT	19.85	8.3523	32.9341	42.9961	23.4202	35.7958	8.138	114.83
LSST	10.05	1.8700	73.1139	4.7716	23.3066	$3.0729^\ddagger$	$7.574^\dagger$	116.43
ONND	$8.443^\ddagger$	$1.5816^\dagger$	27.8825	36.6116	29.6067	63.3374	20.37	$7.1114^\dagger$
Ours	$8.021^\dagger$	1.3649^*	$10.3670^\dagger$	$4.0786^\dagger$	2.6542^*	2.4861^*	3.845^*	$7.9432^\ddagger$

Table 3: Average tracking success rate. The best three results are marked by *, †, and ‡.

	Animal	Car11	Girl	Jumping	Cliffbar	Caviar	Football	Woman
IVT	0.2166	0.8077	0.4262	0.2826	0.5648	0.1435	0.5573	0.0777
L1	0.0386	0.4353	0.3263	0.0927	0.1993	0.1387	0.5732	0.1262
PN	0.4118	0.3761	0.5770	0.6904	0.3798	0.1632	0.5049	0.4616
VTD	0.5771	0.4320	0.5125	0.0797	0.3292	0.1519	0.6165	0.1181
MIL	0.2129	0.1745	0.5197	0.5267	0.4622	0.1330	0.5760	0.1290
Frag	0.0764	0.0857	$0.6887^\dagger$	0.1383	0.1337	0.1334	0.5210	0.1270
MTT	0.5185	0.7537	0.6338	0.2318	0.3073	0.1420	0.6643	0.1251
SCM	0.6081	0.6949	$0.6791^\ddagger$	$0.7174^\dagger$	$0.6533^\dagger$	$0.8332^\ddagger$	$0.8296^\dagger$	0.1728
ASLA	0.6198^*	0.7930	0.6491	$0.7121^\ddagger$	$0.6197^\ddagger$	0.6721	0.6362	0.7929^*
CT	0.5250	0.5306	0.5108	0.1531	0.3852	0.1727	$0.6994^\ddagger$	0.1202
LSST	0.5750	$0.8106^\ddagger$	0.1199	0.6540	0.5648	$0.8510^\dagger$	0.6888	0.1622
ONND	$0.6095^\ddagger$	0.8425^*	0.4205	0.1357	0.3483	0.0519	0.4076	$0.6731^\dagger$
Ours	$0.6124^\dagger$	$0.8376^\dagger$	0.6908^*	0.7201^*	0.7986^*	0.8540^*	0.8308^*	$0.6593^\ddagger$

center position and the ground truth center position. Clearly, we hope the CLE is small. Figure 1 presents the relative position errors (in pixels) between the ground truth center and the tracking results. Table 2 summarizes the average center location errors in pixels.

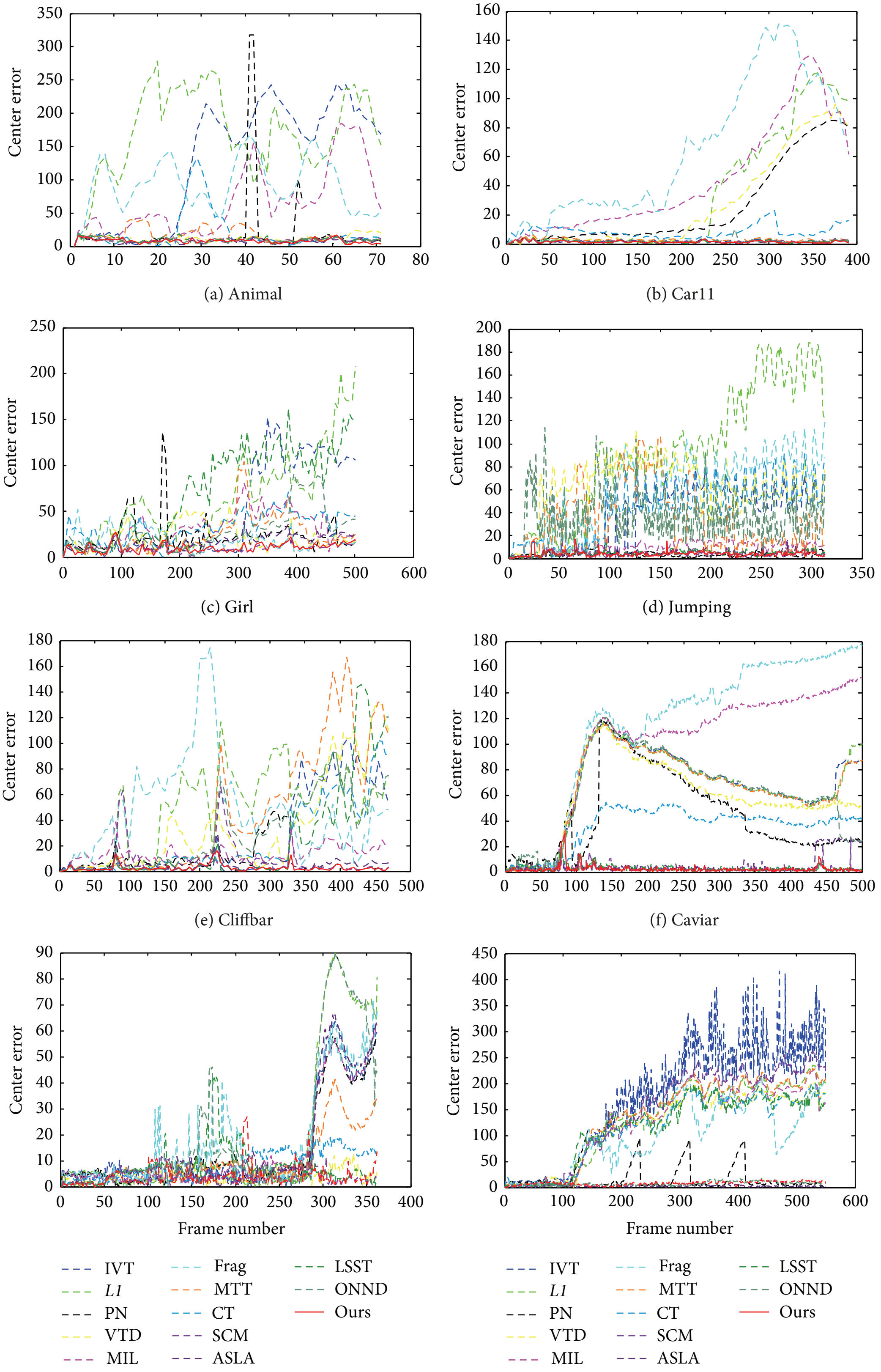
The TSR is computed as the ratio of the number of frames the target is successfully tracked to the number of frames in the sequence. To define whether the target is successfully tracked at a frame, we use the score in the PASCAL VOC challenge [48], which can be computed as

$$\text{score} = \frac{\text{area}\left(R_T \cap R_G\right)}{\text{area}\left(R_T \cup R_G\right)}, \tag{15}$$

where R_T is the current tracking result and R_G is the ground truth. Table 3 and Figure 2 give the average tracking success rates and relative tracking success rates, respectively. Overall, the proposed tracker performs well against the other state-of-the-art algorithms.

5.3. Qualitative Comparison. There are blurred images in animal sequence, which is difficult for most trackers to solve this situation. From Figure 3, we can see that the head of the fawn becomes blurred at the frame 25 or 42; the appearance of the tracked object is indistinguishable. Most tracking algorithms fail to follow the target, such as MIL, PN, and Frag. The proposed algorithm successfully tracks the target object throughout the sequence. Its located accuracy and overlap rate are better than SCM, LSST, and ONND and less than ASLA.

5.4. Quantitative Comparison. In the car11 sequence, a car is driven into a very dark environment. The contrast between the tracked target and its surrounding background is low, and the ambient light changes significantly. Furthermore, the low image resolution of the target object makes tracking difficult. The tracking results are illustrated in Figure 3. Due to changes in lighting, Frag and MIL algorithms start to drift around frame 60. *L1* method starts to fail in frame 250. IVT, SCM, ASLA, LSST, MTT, and ONND algorithms perform well as our tracker in the whole sequence. However, the accuracy and robustness of these methods are less than our proposed algorithm. However, the other methods drift away when



(a) Animal

(b) Car11

(c) Girl

(d) Jumping

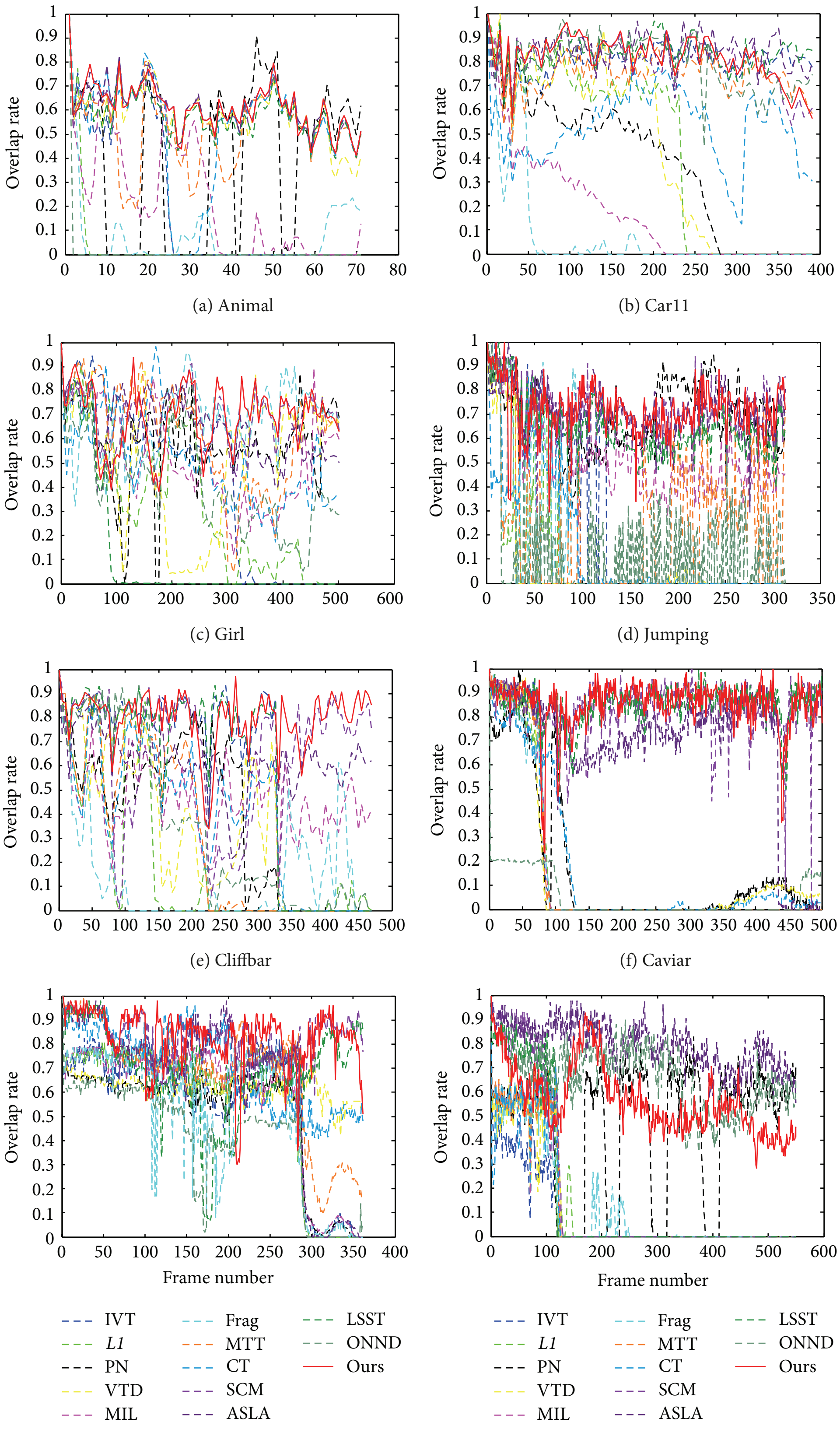
(e) Cliffbar

(f) Caviar

(g) Football

(h) Woman

FIGURE 1: Central-pixel error. This figure shows central-pixel error for ten tested video clips. Our algorithm is compared with ten state-of-the-art methods.



(a) Animal

(b) Car11

(c) Girl

(d) Jumping

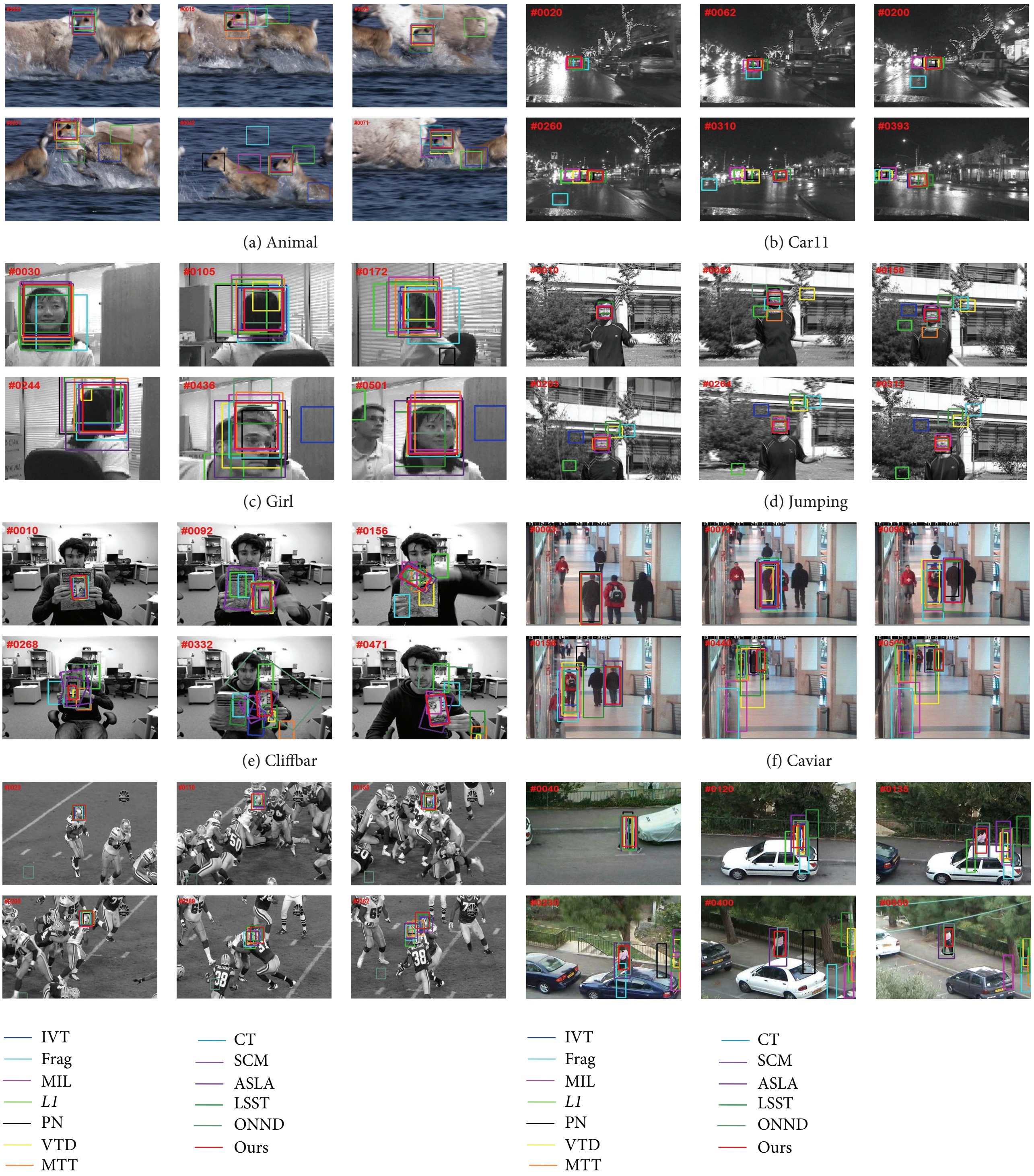
(e) Cliffbar

(f) Caviar

(g) Football

(h) Woman

Figure 2: Overlap rate evaluation. This figure shows overlap rates for ten tested video clips. Our algorithm is compared with ten state-of-the-art methods.



(a) Animal

(b) Car11

(c) Girl

(d) Jumping

(e) Cliffbar

(f) Caviar

(g) Football

(h) Woman

Figure 3: Comparison of 13 trackers on 8 video sequences in terms of bounding box reported.

drastic illumination variation occurs (e.g., #0200 and #0250) or when similar objects appear in the scene (e.g., #0305), especially the car makes a turn at about frame 260.

The tracking object in the girl sequence undergoes occlusion (complete occlusion of the girl's face as she swivels in the chair), large pose change, and scale variation with in-plane and out-of-plane rotations (from large to small and from small to large). The tracking results are shown in Figure 3. The experimental results demonstrate that our method achieves the best performance in this sequence.

Other trackers experience drift at different instances: Frag at frame 248, IVT at frame 436, and VTD at frame 477.

There is abrupt motion in jumping sequences, so it is difficult to predict the location of tracked target in the blurry images. Furthermore, it is rather challenging to account for drastic appearance change caused by motion blur and properly update these appearance models. Figure 3 shows that most tracking algorithms fail to follow the target right at the beginning of this sequence (e.g., #13 and 25). The proposed algorithms SCM, ASLA, LSST, and PN successfully track the target object throughout the sequence.

In the cliffbar video, the background has similar texture to the target. Moreover, the target undergoes scale variance, in-plane rotation, and abrupt motion as shown in Figure 3. The Frag, *L1*, IVT, CT, MIL, LSST, ONND, and SCM methods drift to the cluttered background, while our proposed tracker has the best performance on this sequence; it can adapt to the scale and rotation change of the target and overcome the influence of similar background and motion blur.

In the caviar sequence, the target is occluded by two people at times and one of them is similar in color and shape to the target. Numerous methods fail to track the target because there are similar objects around it when heavy occlusion occurs. In contrast, our tracker achieves stable performance in the entire sequence when there is a large scale change with heavy occlusion at frame 442.

The football sequence is challenging due to the cluttered background, because there are many football players with the similar helmets in appearance to the tracked object in this scene. When the tracked target approaches other football players, some trackers are not robust and begin to drift, as shown in frames 76, 113, and 150 in Figure 3.When the two football players collide at frame 290, most tracking methods especially cannot locate the target correctly. Only our tracker, CT, VTD, and ONND overcome this problem and successfully locate the correct object in the whole sequence. The accuracy of our method is the highest.

In the woman sequence, the walking woman undergoes pose variation together with long-time partial occlusion. The difficulty lies in the fact that the woman is greatly occluded by the parked cars. Most trackers fail and lock on a car with similar color to the trousers when the legs of the woman are heavily occluded from frame 110 to 130. Only ONNDL, our tracker, and ASLA can overcome this difficulty and follow the target accurately. Although PN tracker can find the tracked target again after the trace fails, it is vulnerable for occlusion and always loses the target as shown in Figure 3.

6. Conclusions

In this paper, we present a supervised approach to learn and update a structured, sparse, and discriminative representation for object tracking. Label information from training data is incorporated into the dictionary learning process to construct a discriminative structured dictionary. This is accomplished by adding an ideal-code regularization term and classification error term to the total objective function. By minimizing the objective function, we can obtain a high quality dictionary and optimal linear classifier simultaneously. This approach exploits the strength of label information and encourages images from the same class to have similar representations. Experimental results on challenging image sequences demonstrate that our tracking algorithm performs favorably against several state-of-the-art algorithms. Possible future work includes online and robust discriminative dictionary learning and structured low-rank representations for real-time object tracking.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work has been funded by the Natural Science Foundation of China (Grant no. 61203270), 2012 State Key Laboratory of Robotics Open Project, and the Introduction Foundation for the Talent of Nanjing University of Tele. and Com (no. NY212028). The project was supported by the Jiangsu Key Laboratory of Image and Video Understanding for Social Safety (Nanjing University of Science and Technology), Grant no. 30920140122007.

References

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: a survey," *ACM Computing Surveys*, vol. 38, no. 4, article 13, 2006.

[2] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. van den Hengel, "A survey of appearance model in visual object tracking," *Transactions on Intelligent Systems and Technology*, vol. 4, no. 4, article 58, 2013.

[3] Y. Wu, L. Jongwoo, and Y. Ming-Hsuan, Online object tracking: a Benchmark, in CVPR, 2013.

[4] S. Zhang, H. Yao, X. Sun, and X. Lu, "Sparse coding based visual tracking: review and experimental comparison," *Pattern Recognition*, vol. 46, no. 7, pp. 1772–1788, 2013.

[5] A. Smeulder, D. Chu, R. Cucchiara, S. Calderara, A. Deghan, and M. Shah, "Visual tracking: an experimental survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013.

[6] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[7] X. Mei and H. Ling, "Robust visual tracking and vehicle classification via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 11, pp. 2259–2272, 2011.

[8] X. Mei, H. Ling, Y. Wu, E. P. Blasch, and L. Bai, "Efficient minimum error bounded particle resampling L1 tracker with occlusion detection," *IEEE Transactions on Image Processing*, vol. 22, no. 7, pp. 2661–2675, 2013.

[9] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real time robust *l*1 tracker using accelerated proximal gradient approach," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 1830–1837, Providence, RI, USA, June 2012.

[10] B. Liu, J. Huang, L. Yang, and C. Kulikowsk, "Robust tracking using local sparse appearance model and K-selection," in

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11), pp. 1313–1320, June 2011.

[11] B. Liu, L. Yang, J. Huang, P. Meer, L. Gong, and C. Kulikowski, "Robust and fast collaborative tracking with two stage sparse optimization," in *Proceedings of the European Conference on Computer Vision*, pp. 1–14, 2011.

[12] N. Wang, J. Wang, and D. Yeung, "Online robust non-negative dictionary learning for visual tracking," in *Proceedings of the International Conference on Computer Vision (ICCV '13)*, 2013.

[13] D. Wang, H. Lu, and M.-H. Yang, "Least soft-threshold squares tracking," *in Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR '013)*, pp. 2371–2378, 2013.

[14] H. Li, C. Shen, and Q. Shi, "Real-time visual tracking using compressive sensing," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 1305–1312, June 2011.

[15] W. Zhong, H. Lu, and M.-H. Yang, "Robust object tracking via sparsity-based collaborative model," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 1838–1845, Providence, RI, USA, June 2012.

[16] X. Jia, H. Lu, and M. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 1822–1829, June 2012.

[17] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Low-rank sparse learning for robust visual tracking," in *Proceedings of the 12th European Conference on Computer Vision (ECCV '12)*, pp. 1–8, Florence, Italy, October 2012.

[18] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Robust visual tracking via multi-task sparse learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 2042–2049, Providence, RI, USA, June 2012.

[19] X. Chen, W. Pan, J. T. Kwok, and J. G. Carbonell, "Accelerated gradient method for multi-task sparse learning problem," in *Proceedings of the 9th IEEE International Conference on Data Mining (ICDM '09)*, pp. 746–751, December 2009.

[20] X. Yuan and S. Yan, "Visual classification with multi-task joint sparse representation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 3493–3500, San Francisco, Calif, USA, June 2010.

[21] M. Black and A. Jepson, "Eigentracking: robust matching and tracking of articulated objects using a view-based representation," in *Proceedings of the European Conference on Computer Vision*, pp. 329–342, 1996.

[22] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564–577, 2003.

[23] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, pp. 798–805, June 2006.

[24] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 1–3, pp. 125–141, 2008.

[25] J. Kwon and K. M. Lee, "Visual tracking decomposition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 1269–1276, June 2010.

[26] S. Avidan, "Ensemble tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 2, pp. 261–271, 2007.

[27] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via online boosting," in *Proceedings of the British Machine Vision Conference (BMVC'06)*, pp. 47–56, Edinburgh, Scotland, September 2006.

[28] H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised online boosting for robust tracking," in *Proceedings of the European Conference on Computer Vision*, pp. 234–247, 2008.

[29] B. Babenko, M. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1619–1632, 2010.

[30] N. Jiang, W. Liu, and Y. Wu, "Adaptive and discriminative metric differential tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 1161–1168, Providence, RI, USA, June 2011.

[31] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: bootstrapping binary classifiers by structural constraints," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 49–56, June 2010.

[32] K. Zhang, L. Zhang, and M.-H. Yang, "Real-time compressive tracking," in *Proceedings of the 12th European Conference on Computer Vision (ECCV '12)*, pp. 864–877, 2012.

[33] R. T. Collins, Y. Liu, and M. Leordeanu, "Online selection of discriminative tracking features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1631–1643, 2005.

[34] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. I511–I518, December 2001.

[35] C. Leistner, A. Saffari, and H. Bischof, "Miforests: multiple-instance learning with randomized trees," in *Proceedings of the European Conference on Computer Vision (ECCV '10)*, pp. 29–42, 2010.

[36] B. Zeisl, C. Leistner, A. Saffari, and H. Bischof, "On-line semi-supervised multiple-instance boosting," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 1879–1886, San Francisco, Calif, USA, June 2010.

[37] K. Zhang, L. Zhang, and M.-H. Yang, "Real-time object tracking via online discriminative feature selection," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4664–4677, 2013.

[38] K. Zhang, L. Zhang, M.-H. Yang, and Q. H. Hu, "Robust object tracking via active feature selection," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 23, no. 11, pp. 1957–1967, 2013.

[39] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: universal encoding strategies?" *Institute of Electrical and Electronics Engineers. Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, 2006.

[40] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[41] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Discriminative learned dictionaries for local image analysis," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '08)*, pp. 1–8, Anchorage, Alaska, USA, June 2008.

[42] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *Proceedings of the IEEE*

Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10), pp. 2691–2698, June 2010.

[43] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Supervised dictionary learning," in *Proceedings of the 23rd Annual Conference on Neural Information Processing Systems (NIPS '09)*, pp. 1033–1040, Vancouver, Canada, December 2009.

[44] J. Mairal, M. Leordeanu, F. Bach, M. Hebert, and J. Ponce, "Discriminative sparse image models for class specific edage detection and image interpretation," in *Proceedings of the 10th European Conference on Computer Vision (ECCV '08)*, pp. 43–56, Marseille, France, 2008.

[45] H. Zhang, A. Berg, M. Maire, and J. Malik, "SVM-KNN: discriminative nearest neighbor classification for visual category recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 2, pp. 2126–2136, June 2006.

[46] C. Lu, J. Shi, and J. Jia, "Online robust dictionary learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'13)*, pp. 415–422, Portland, Ore, USA, June 2013.

[47] Z. Jiang, Z. Lin, and L. S. Davis, "Label consistent K-SVD: learning a discriminative dictionary for recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2651–2664, 2013.

[48] M. Everingham, L. van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.