

Robust visual tracking via discriminative appearance model based on sparse coding

Hainan Zhao · Xuan Wang

Published online: 22 November 2014
© Springer-Verlag Berlin Heidelberg 2014

Abstract In this paper, we formulate visual tracking as a binary classification problem using a discriminative appearance model. To enhance the discriminative strength of the classifier in separating the object from the background, an over-complete dictionary containing structure information of both object and background is constructed which is used to encode the local patches inside the object region with sparsity constraint. These local sparse codes are then aggregated for object representation, and a classifier is learned to discriminate the target from the background. The candidate sample with largest classification score is considered as the tracking result. Different from recent sparsity-based tracking approaches that update the dictionary using a holistic template, we introduce a selective update strategy based on local image patches which alleviates the visual drift problem, especially when severe occlusion occurs. Experiments on challenging video sequences demonstrate that the proposed tracking algorithm performs favorably against several state-of-the-art methods.

Keywords Visual tracking · Local sparse representation · Discriminative appearance model · Template update

H. Zhao · X. Wang (✉)
Computer Application Research Center, Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen, China
e-mail: wangxuan@cs.hitsz.edu.cn

H. Zhao
e-mail: hainan.hh@gmail.com

H. Zhao · X. Wang
Shenzhen Applied Technology Engineering Laboratory for Internet Multimedia Application, Shenzhen, China

H. Zhao · X. Wang
Public Service Platform of Mobile Internet Application Security Industry, Shenzhen, China

1 Introduction

Visual tracking is one of the most important components in computer vision which finds numerous applications in surveillance, human–computer interaction, vehicle navigation, etc. Given the initialized state (manually or by detection methods) of the target object in a frame of a video sequence, the goal of tracking is to estimate the states of the target in the subsequent frames. Although many tracking methods have been proposed and significant progress has been made within the last decades, it still suffers from difficulties in handling complex object appearance changes caused by factors such as illumination variation, partial occlusion, background clutter, and viewpoint change.

As an essential component in all tracking methods, object appearance model plays a key role in determining the tracking performance. To get an effective appearance model for visual tracking, discriminating the target from the background is a basic ability and especially important in complex scenes where the contrast between target and background is low. In literature [21], a number of discriminative tracking algorithms which formulate tracking as a binary classification problem have been proposed. Recently, sparse representation has been successfully applied to computer vision tasks, including face recognition [20], image inpainting [12], visual tracking [13], etc. And sparse learning-based method has reported superior performance in video semantic recognition [6] and region tagging [5]. In [20], a robust face recognition method is proposed, in which each test sample is linearly represented by all training samples. The recognition result of the test sample is encoded in the representation coefficients. The robustness to occlusion is achieved by introducing a representation error in the linear system. Compared to the traditional face recognition methods, it obtains superior

performance even with random features. Han et al. [7] propose a sample-based adaptive sparse representation (AdaSR) method. In this method, a sample set is constructed which gets the appearance information of both the target and the background. Each candidate sample is sparsely represented as a linear combination of templates in this set. Tracking is implemented by searching for the candidate holding the most similar AdaSR to that of the target template. The tracker performs well in complex scenes due to the background information captured in sample set which helps the tracker to discriminate the target from the background. Most recently, local features and their spatial relations get more and more attention in the field of image classification [22, 25]. In [23] and [24], both the local features and the geometric property of image are exploited for image categorization which report appealing experimental results.

Inspired by the work mentioned above, we propose a novel discriminative tracking method based on sparse coding [22]. Our method aims at discriminating the target from its background and being adapt to appearance variations of the target simultaneously. In our approach, an over-complete dictionary is constructed with a set of overlapped local image patches cropped with a fixed spatial structure inside a sample window. This over-complete dictionary captures local visual information of both the target and the background. An object is divided into a set of overlapped local image patches with the same spatial structure. We learn sparse code for each local image by sparsely representing it using the over-complete dictionary. The object is then represented by concatenating the sparse codes of all the local image patches. With this representation scheme, positive and negative samples are collected in each frame, and a linear classifier is learned to separate the target from the background. Using the classification score of a test candidate target as the likelihood, the most likely target location in each frame can be determined. To account for the target and background appearance variations, we update both the dictionary and the classifier during tracking. Unlike the holistic template update approach commonly used in [7, 13], we introduce a selective update strategy based on local patches. With this selective update strategy, the tracker alleviates the drift problem and gets more robust tracking results.

The rest of the paper is organized as follows. In the next section related work are summarized. The discriminative appearance model based on sparse coding, the classifier, and the selective update strategy are described in Sect. 3. The proposed tracking algorithm and discussion are presented in Sect. 4. We illustrate experimental results with both qualitative and quantitative evaluations in Sect. 5. Finally, concluding remarks are given in Sect. 6.

2 Related work

To deal with the various challenges occurring in the tracking process, a rich literature of object tracking methods have been proposed, which can be generally categorized into either generative or discriminative methods.

Generative methods formulate the tracking problem as searching for the region most similar to the target appearance in an image frame. These methods learn and maintain either templates or subspace models to represent the target. One of the critical issues for generative methods is how to make the tracker adapt to the inevitable appearance variation of target. Ross et al. [15] incrementally learn a low-dimensional eigenspace representation to reflect appearance changes of the target. Kwon et al. [11] decompose the observation model into multiple basic observation models. Each basic observation model covers a specific appearance variation of the target, so that the compound one can be robust to combinatorial appearance variation. Jepson et al. [8] learn a Gaussian mixture model of pixels to represent objects via an online expectation maximization (EM) algorithm. In [1], the target template is represented by multiple image patches to handle partial occlusion and pose change.

Discriminative methods formulate tracking as a binary classification problem. The candidate which can be best separated from the background is taken as the tracking result. Avidan et al. [2] propose an ensemble tracking framework, in which a confidence map is constructed using an ensemble of weak classifiers to separate pixels that belong to the object from ones that belong to the background. The peak of the map is considered as the new position of the object. Babenko et al. [3] use multiple instance learning (MIL) instead of traditional supervised learning to learn a discriminative model for tracking. A discriminative appearance model based on superpixels is introduced in [19]. It facilitates the tracker to distinguish between the target and background. These discriminative tracking methods aim to construct a good appearance model for effectively separating the object from background. Moreover, algorithms take advantage of both generative and discriminative models which are proposed in [4, 26].

The recent development of sparse representation has attracted considerable interest in object tracking due to its robustness to occlusion and image noise. Motivated by the work in [20], Mei et al. [13, 14] apply sparse representation to visual tracking, in which each candidate sample is sparsely represented as a linear combination of target templates and trivial templates. The tracking problem is formulated as finding the candidate with the minimum reconstruction error from the target templates subspace. In their method, partial occlusion, appearance variances, and other challenging issues are considered as the error vector represented by the set of trivial templates. Wang et al. [16]

propose an online object tracking algorithm, which takes advantage of both principal component analysis (PCA) algorithm and sparse representation scheme to learn an adaptive appearance model. Compared with the holistic sparse representation methods, local sparse representation methods [9, 10] are more effective in modeling the appearance changes of the object during tracking. They are particularly useful when the appearance of target is partially changed, such as partial occlusion or deformation. Most recently, background information has been introduced into the sparse representations either in dictionary learning [7] or discriminative classifier [17, 26] which reports appealing experimental results.

3 Discriminative appearance model and classifier

In this paper, we use local sparse codes to represent objects and formulate tracking as a binary classification problem. We initialize the dictionary and classifier in the first frame after the target object is labeled manually or automatically. Both the dictionary and classifier are updated when new tracking results are available.

3.1 Discriminative appearance model based on sparse coding

Local models are important for tracking as shown in the performance improvement of local sparse representation model compared with the holistic sparse representation model [21]. In our method, we first encode the local patches inside the object region using an over-complete dictionary and then aggregate these local sparse codes for object representation. When the state of the target is estimated, we sample a set of overlapped local image patches using sliding windows inside a sample window, which is a rectangle region centered around the target. In our experiments, the sample window is empirically set to 6 or 9 times of the size of the object region, which is determined by the speed of the target object. These local patches are used as a over-complete dictionary to encode the patches inside the object region, i.e., $D = [d_1, d_2, \dots, d_M] \in \mathbb{R}^{H \times M}$, where H is the dimension of the vectorized local image, M is the number of local images in the dictionary. Most of these local images are associated with the background, we call them background basis set, denoted by D_{bg} . A few of these local images are parts of the target object, we call them target basis set, denoted by D_{tar} . For a target object, we extract local patches within it and turn them into vectors in the same way, which are denoted by $X = [\chi_1, \chi_2, \dots, \chi_m] \in \mathbb{R}^{H \times m}$, where m is the number of local images extracted from the object region. With the sparsity assumption, the local patches χ_i within the sample

window can be represented as the linear combination of only a few basis elements of the dictionary D by solving

$$b_i^* = \arg \min_{b_i} \|\chi_i - Db_i\|_2^2 + \lambda \|b_i\|_1, \quad \text{s.t. } b_i \succeq 0, \quad (1)$$

where $b_i \in \mathbb{R}^M$ is the sparse coefficient vector of the i th local patch, and λ is a regularization constant to balance reconstruction error and sparsity. $b_i \succeq 0$ indicates that all the elements of b_i are nonnegative.

With our object representation, local image patches from target and background can be represented by different bases in the dictionary. Specifically, the image patches from the target region are likely to be well reconstructed by only the target basis set, while the ones from the background region can be better represented by the span of the background basis set. Therefore, the local sparse code is more discriminative. When the local sparse codes $[b_1, b_2, \dots, b_m]$ of all the image patches from an object region are computed, we aggregate them to obtain the object representation for visual tracking. There exist many pooling methods for computing the final feature vector based on some statistics of the local codes [9, 18]. Here we directly concatenate all these local sparse codes together to represent the object, i.e., $\beta = [b_1^*, b_2^*, \dots, b_m^*]^T$. Each patch represents one part of the object; therefore, all local patches from the object with a fixed spatial relationship can reflect the object structure.

3.2 Classifier learning with sparse coding

We pose visual tracking as a classification problem, in which the aim is to separate the target object from the background. To initialize the classifier in the first frame, we draw positive and negative samples around the labeled target location. Specifically, we use a Gaussian perturbation to draw N_{pos} positive samples around the selected target location (e.g., within a radius of a few pixels), and draw N_{neg} negative samples further away from the labeled location (e.g., within an annular region a few pixels away from the target object). We make the negative samples consisting of images of both the background and parts of the target object. By this way, candidate samples containing only partial appearance of the target are treated as the negative samples, which facilitate better object localization. Both of these positive and negative sample images are set to the same size as the labeled target for generality. We then compute the concatenate sparse codes vector β for all these sample images to form the training data, $\{\beta_i, y_i\}_{i=1}^{N_{pos}+N_{neg}}$, where $y_i \in \{+1, -1\}$. With the training data, the linear classifier is learned by minimizing the following loss function

$$J(w) = \frac{1}{N_{pos} + N_{neg}} \sum_{i=1}^{N_{pos}+N_{neg}} \ell(y_i, w, \beta_i) + \frac{\lambda}{2} \|w\|_2^2, \quad (2)$$

where w is the classifier parameter, and $\ell(\bullet)$ is the logistic regression loss function with the mathematical formulae:

$$\ell(y, w, \beta) = \log(1 + e^{-yw^T\beta'}), \quad (3)$$

where $\beta' = [\beta^T, 1]^T$ is the augmented vector. Once the classifier is initialized, the classification score can be utilized as the similarity measure for tracking, which can be computed by

$$h(\beta) = \frac{1}{1 + e^{-w^T\beta'}}. \quad (4)$$

3.3 Selective update strategy

Since the appearances of both the target and background may change during the tracking process, it is necessary to update the appearance model, i.e., the dictionary and the classifier in our work. Many dictionary update approaches have been proposed [13, 16]. However, most of these methods replace one of the holistic templates in the dictionary using a newly tracking result or a reconstructed image every update. These holistic template update strategies are likely to lead to template drift especially when the tracking object is severely occluded. The dictionary in our method aims at modeling both of the target and the background image effectively so that the classifier separate the target object from the background as much as possible with our sparse representation. Regarding this criterion, we adopt a selective update strategy here. When the state of the target is estimated, we select a new sample window centered around the target and resample a set of overlapped local image patches in the same way as we did in the first frame. We update the templates in the dictionary using the corresponding local patch in the new sample window selectively. Our selective update strategy is developed based on the assumption that an image patch with smaller reconstruction error using the target basis set D_{tar} indicates that it is more likely to be a part of the target object, and the one with smaller reconstruction error using the background basis set D_{bg} indicates that it is occluded or be mixed with a large number of background pixels. Here we define a confidence value δ for each resampled local image patch by

$$\delta = \frac{1}{1 + \exp(\varepsilon_{\text{tar}} - \varepsilon_{\text{bg}})}, \quad (5)$$

where $\varepsilon_{\text{tar}} = \|\chi - D_{\text{tar}}^{t-1}b_{\text{tar}}\|_2^2$ is the reconstruction error of the template χ with the previous target basis set D_{tar}^{t-1} , and b_{tar} is the corresponding sparse coefficient vector calculated by Eq. (1). Similarly, $\varepsilon_{\text{bg}} = \|\chi - D_{\text{bg}}^{t-1}b_{\text{bg}}\|_2^2$ is the reconstruction error of the template χ using the previous background basis set D_{bg}^{t-1} , and b_{bg} is the related sparse coefficient vector.

For the template from the background basis set, only if its $\varepsilon_{\text{tar}} > \varepsilon_{\text{bg}}$, we update it using the corresponding local patch by

$$d_{\text{bg}}^t = \delta d_{\text{bg}}^{t-1} + (1 - \delta)d_{\text{bg}}^l, \quad (6)$$

where the new template histogram d_{bg}^t is composed of the histogram d_{bg}^{t-1} at the previous frame and the corresponding histogram d_{bg}^l last stored. In this way, the background basis set captures the appearance change of the background and meanwhile avoids being mixed with image patches from the target region. Conversely, we update the template from the target basis set, only if its $\varepsilon_{\text{tar}} < \varepsilon_{\text{bg}}$. In order to capture the new appearance of the target and recover the object from occlusions, the template histogram is updated by

$$d_{\text{tar}}^t = (1 - \delta)d_{\text{tar}}^0 + \delta d_{\text{tar}}^l, \quad (7)$$

where the new template histogram d_{tar}^t is composed of the histogram d_{tar}^0 at the first frame and the corresponding histogram d_{tar}^l last stored. In this way, target basis set adapts to the appearance variation of the target object without falsely updating with occlusion patch or patch with large number of background pixels simultaneously. To retrain the classifier using the new dictionary, training sample set is reconstructed. Specifically, the negative samples are re-collected in the same way as we did at the first frame. Somewhat differently for the positive samples, we randomly choose a sample from the positive set and replace it with the latest tracking result.

The selective updating of the dictionary ensures that most recent appearances of both the object and the background are reflected in the dictionary basis set and simultaneously prevents the target basis set being polluted by occluded or deteriorative patches, or the background basis set being mixed with part of the object. It also should be noted that only one sample in the positive sample set is replaced each time, so even a bad sample replacement during the tracking process affects little on the classifier used for calculating tracking result, which also avoids the drift problem and ensures tracking stability.

4 Proposed tracking algorithm

Our object tracking algorithm is carried out within the Bayesian inference framework. Given the observation set of target $Z_t = \{z_1, z_2, \dots, z_t\}$ up to the t th frame, we estimate the target state variable by maximizing the posteriori probability over N samples at frame t by

$$x_t^* = \arg \max_{x_t^i} p(x_t^i | Z_t), \quad i = 1, 2, \dots, N, \quad (8)$$

where x_t^i indicates the state of the i th sample at the t th frame. The posteriori probability $p(x_t|Z_t)$ (we drop the sample index i for generality) can be estimated recursively by

$$p(x_t|Z_t) \propto p(z_t|x_t) \int p(x_t|x_{t-1})p(x_{t-1}|Z_{t-1})dx_{t-1}, \quad (9)$$

where $p(x_t|x_{t-1})$ is the dynamic (motion) model between two consecutive states. We apply the affine transformation with six parameters to model the target motion, and formulate the state transition as $p(x_t|x_{t-1}) = N(x_t; x_{t-1}, \psi)$, where ψ is a diagonal covariance matrix. $p(z_t|x_t)$ is the observation model which denotes the likelihood of the observation z_t at candidate state x_t . It plays an important role in robust tracking. In our method, we formulate $p(z_t|x_t)$ using the classification score computed by Eq. (4) as

$$p(z_t|x_t) \propto h(\beta). \quad (10)$$

Algorithm 1 Proposed tracking algorithm

- 1: Input: Image frames F_1, \dots, F_T , the initial state x_0^* .
 - 2: Output: Target state \hat{x}_t^* at frame t .
 - 3: Construct the initial over-complete dictionary $D_0 = [d_1^0, d_2^0, \dots, d_M^0]$ and learn a classifier with parameter W_0
 - 4: for $t = 1$ to T do
 - 5: Draw N candidate samples $X_i (i = 1, 2, \dots, N)$ and extract local patches $[x_i^1, x_i^2, \dots, x_i^m]$ from each candidate
 - 6: Compute sparse coefficients b_i^{j*} of each x_i^j according to Eq. 1 and concatenate them together to represent each candidate $\beta_i = [b_i^{1*}, b_i^{2*}, \dots, b_i^{m*}]^T$
 - 7: Find the tracking result \hat{x}_t^* by minimizing Eq. 4
 - 8: if update then
 - 9: Obtain a set of templates $[d_1^1, d_2^1, \dots, d_M^1]$ in the new sample window
 - 10: for $j = 1$ to M do
 - 11: Compute the confidence value δ_j of d_j^l according to Eq. 5
 - 12: if $d_j^l \in D_{bg}$ && $\varepsilon_{tar} > \varepsilon_{bg}$ then
 - 13: Update d_j according to Eq. 6
 - 14: else if $d_j^l \in D_{tar}$ && $\varepsilon_{tar} < \varepsilon_{bg}$ then
 - 15: Update d_j according to Eq. 7
 - 16: end if
 - 17: end for
 - 18: end for
-

The proposed tracking algorithm is summarized in Algorithm. It should be noted that our work bears some similarity to [17] in the form of constructing a discriminative tracker with local sparse representations. However, two key different points should be emphasized. First, being different from [17] which only use target templates in the sparse representation, we introduce both the target and background templates in

the linear system, which is capable of discriminating the target from its background preferably. For example, the image patch from part of a bad candidate object marked by the dark rectangle is shown in Fig. 1a. It contains a large number of background pixels. In this case, both of the target bases as well as the trivial bases are employed for good reconstruction in [17]; moreover, the sparse coefficients obtained are not sparse as shown in Fig. 1b. In contrast, only background bases get large coefficient values in our sparse representation, while the coefficient values corresponding to target bases tend to zero. Figure 1c shows the obtained coefficient vector. Intuitively, the coefficient vector obtained by our method is more sparser. Therefore, the coefficients are more discriminative. In addition, our updating method updates the dictionary selectively which effectively models the appearance variation of the object as well as the background with less template drift. Compared with the two-stage tracking method, we obtain favorable tracking results but with lower computational cost (details in Sect. 5).

5 Experiments

We evaluate the performance of the proposed algorithm on eight challenging image sequences. These sequences cover various challenges such as illumination variation, partial occlusion, and complex background. The proposed approach is compared with five state-of-the-art tracking

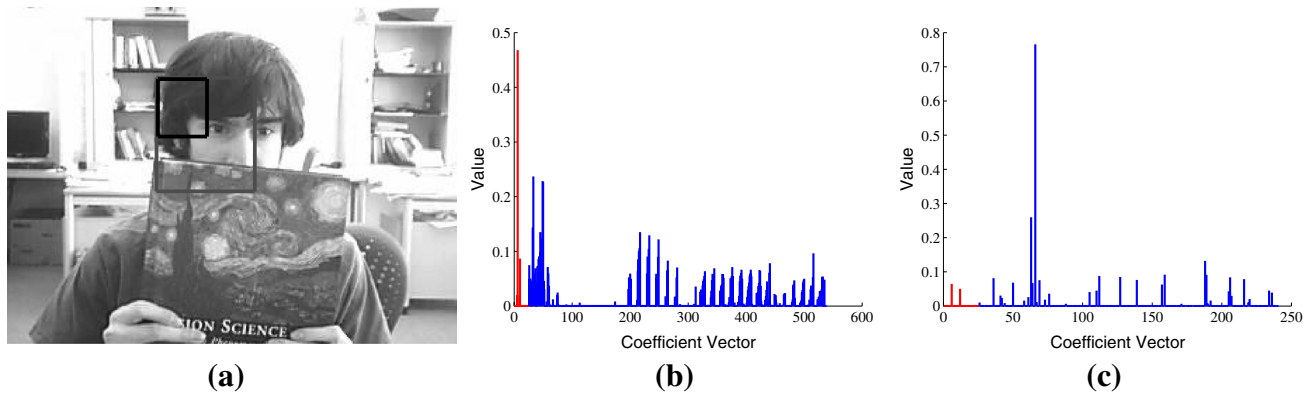


Fig. 1 Coefficient examples. **a** A ‘bad’ candidate is marked by gray rectangle and the example local patch marked by *dark rectangle* on the *lefttop*. **b** The coefficient vector of [17] using target bases and

trivial bases. **c** The coefficient vector of our method using target as well as background bases

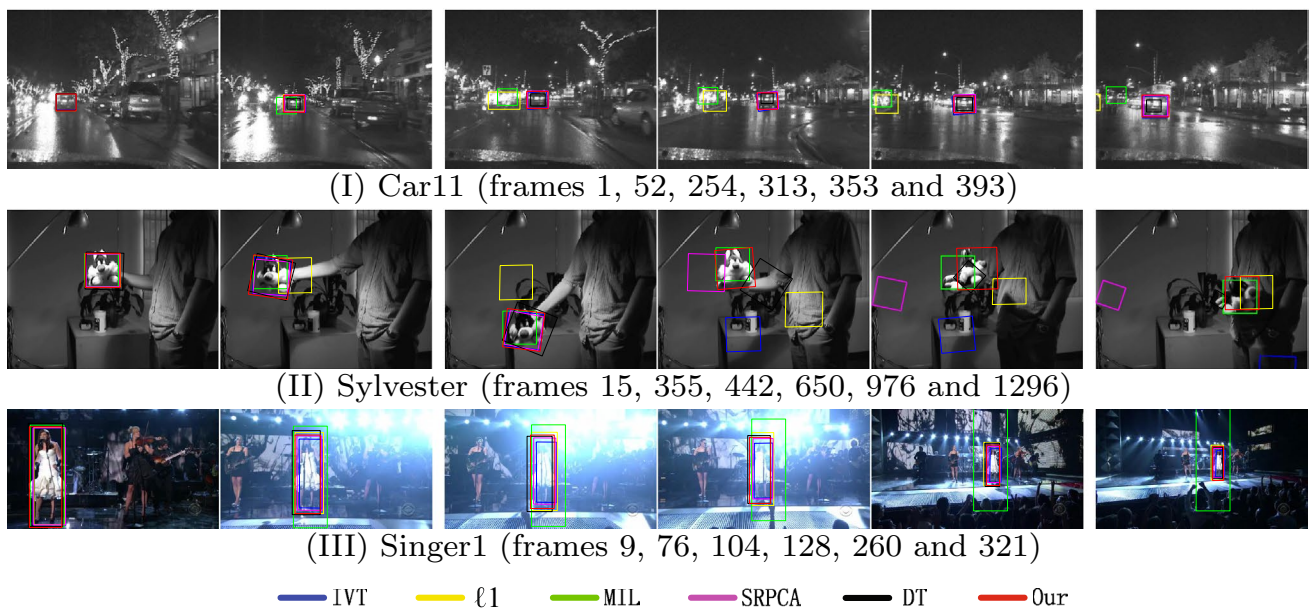


Fig. 2 Tracking results when there is large illumination variation

methods including incremental visual tracking (IVT) method [15], ℓ_1 tracker (ℓ_1) [13, 14], multiple instance learning (MIL) method [3], tracking method with sparse prototypes (SRPCA) [16], and the discriminative object tracking (DT) [17]. For fair comparison, we use the source or binary codes provided by the authors with tuned parameters for best performance.

Our tracker is implemented in MATLAB, which runs at 4.7 frames per second (fps) on a PC with Intel Core i7-3770 CPU (3.4 GHz) with 16 GB memory. The target image observation is normalized to 32×32 pixels from which overlapping 16×16 patches with a shift of 8 pixels are extracted. The local image patches used in the dictionary are normalized to the same size for efficiency, however, the

sliding step of which is set to 4 pixels for good reconstruction. A number of 400 particles are used in our experiment and the dictionary and classifier are updated every 5 frames for computational efficiency. Only gray scale information is used in our experiments. For each sequence, the location of the target object is manually labeled in the first frame. Both qualitative and quantitative evaluations are presented in this section.

5.1 Qualitative evaluation

Illumination change: Fig. 2 shows results from challenging sequences with significant change of illumination. In the Car11 sequence, the target object is small with low contrast

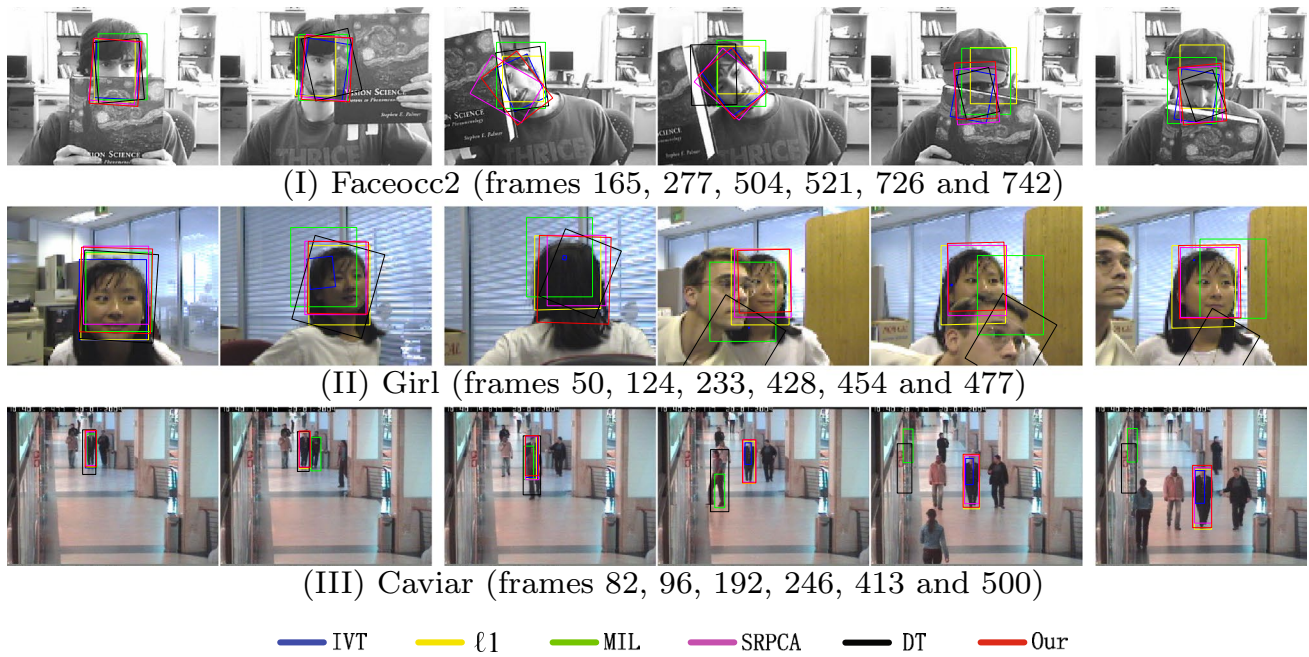


Fig. 3 Tracking results when there is occlusion

and drastic illumination change. The IVT tracking method achieves good results in this sequence. It can be attributed to the fact that subspace learning method is robust to illumination changes. Our tracker performs well in spite of the low contrast between the foreground and the background due to the discriminative appearance model. The DT and SRPCA method also keep tracking the vehicle throughout the video sequence, while the other methods drift to the cluttered background or other vehicles when drastic illumination variation occurs. In the Sylvester sequence, an animal doll moves with significant pose and lighting variation. As shown in Fig. 2II, the ℓ_1 and the SRPCA methods gradually drift away from the target object and the IVT method also fails to track the target in frame 650 and does not recover later. Our tracker is capable of tracking the doll all the time even when it changes pose drastically. The MIL tracker also achieves comparable performance. In the singer1 sequence, the stage light changes drastically seen from frame 70 to frame 190. The MIL method has large tracking errors since it does not estimate the scale change of the target well, the other trackers are able to locate the target object in this sequence.

Heavy occlusion: Fig. 3 demonstrates how the proposed method performs when the target undergoes heavy occlusion. For the Faceocc2 sequence, many trackers drift apart from the target or do not scale well when the face is heavily occluded. Although the MIL tracker is able to track the target, it is not able to estimate the in-plane rotation due to its design. Our and the SRPCA method are able to track

the target accurately throughout the video sequence. This can be attributed to the update schemes of the two trackers, both of which prevent the dictionary being contaminated by occlusion. For the Girl sequence, the target girl's face undergoes occlusion from a man's face passing in front of it. The IVT method drifts away quickly. This result demonstrates that the IVT method based on PCA subspace representation is sensitive to out-of-plane rotation. The ℓ_1 , SRPCA, and our methods successfully track the target, while other trackers drift apart when occlusion occurs. Figure 3c shows the tracking results in a surveillance video. This video is challenging due to scale change, partial occlusion, and similar objects. The MIL tracker does not perform well because it takes the Haar-like features for object representation, which is sensitive to similar objects occlude each other. The DT method drifts away from the target after it is occluded. The ℓ_1 , SRPCA, and our algorithm keep tracking the target throughout the sequence. And our method obtains better tracking accuracy even when the target is heavily occluded.

Complex background: Fig. 4 presents the tracking results where the target objects appear in complex background. The Stone sequence is challenging as there are numerous stones of different shapes and colors. The MIL, ℓ_1 , and DT methods drift away, whereas the IVT, SRPCA, and our trackers successfully keep tracking the target throughout the sequence. The Bolt sequence is challenging as the target object undergoes large pose variation, occlusion and background clutters. Many trackers lost the target

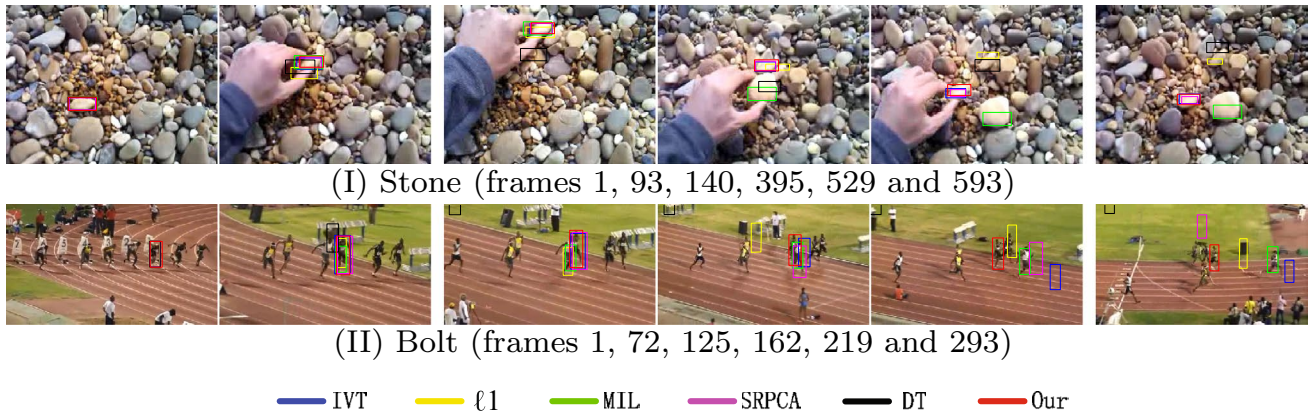


Fig. 4 Tracking results when the targets appear in complex backgrounds

Table 1 Average center error (in pixels)

Image sequence	IVT	ℓ_1	MIL	SRPCA	DT	Ours
Car11	2.3	33.3	43.5	2.2	2.1	1.9
Sylvester	70.5	60.3	<i>11.1</i>	56.2	19.4	9.4
Singer1	9.7	4.6	15.2	5.7	4.5	3.7
Faceocc2	4.9	11.1	14.1	4.0	6.9	3.6
Girl	13.2	4.9	13.9	3.5	16.7	2.2
Caviar	8.4	3.4	69.8	2.1	65.4	2.5
Stone	2.2	19.2	32.3	3.0	28.1	2.8
Bolt	57.7	41.1	38.6	37.8	204.1	9.6
Average	21.1	22.2	29.8	<i>14.3</i>	43.4	4.5

The best and second best results are shown in bold and italic fonts

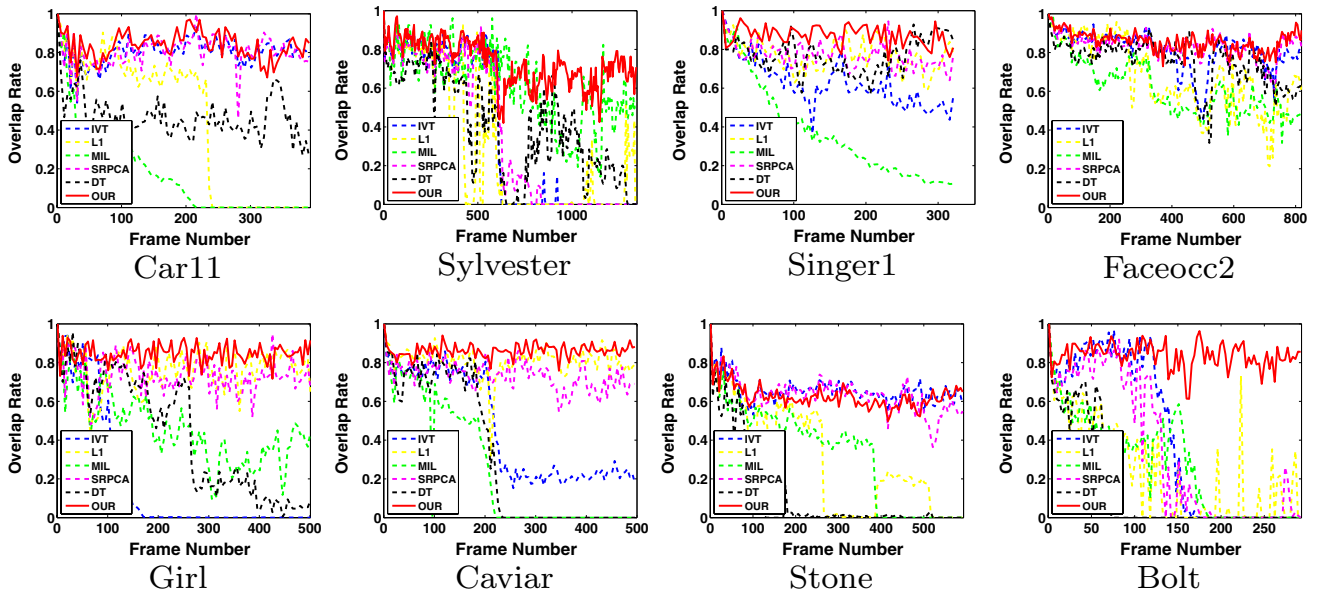


Fig. 5 Quantitative comparison of the trackers in terms of overlap rate

Table 2 Average overlap rate

Image sequence	IVT	ℓ_1	MIL	SRPCA	DT	Ours
Car11	0.83	0.44	0.17	<i>0.81</i>	0.47	<i>0.81</i>
Sylvester	0.36	0.30	<i>0.62</i>	0.38	0.40	0.73
Singer1	0.63	0.77	0.36	0.77	<i>0.79</i>	0.87
Faceocc2	0.81	0.67	0.61	<i>0.83</i>	0.75	0.85
Girl	0.18	<i>0.76</i>	0.48	0.73	0.43	0.84
Caviar	0.45	<i>0.80</i>	0.26	0.73	0.32	0.86
Stone	0.66	0.29	0.32	<i>0.63</i>	0.10	0.61
Bolt	<i>0.40</i>	0.23	0.27	0.35	0.14	0.77
Average	0.54	0.53	0.38	<i>0.65</i>	0.42	0.79

The best and second best results are shown in bold and italic fonts

successively after frame 72 as holistic representations are not effective in handling objects with pose shape variations. Our tracker locks on the target in the whole sequence because the appearance model of our tracker is based on discriminative local feature which is insensitive to non-rigid shape deformation and background clutter.

5.2 Quantitative evaluation

Performance evaluation is an important issue that requires sound criteria in order to fairly assess the strength of tracking algorithms. We employ two typical evaluation criteria to quantitatively assess the performance of these trackers. The first one is center location error which is approximated by the distance between the central position of the tracking result and that of the manually labeled ground truth. Table 1 summarizes the results in terms of average center location error. The second criterion is the tracking overlap rate which indicates stability of each algorithm as taking the size and pose of the target object into account. Figure 5 shows the overlap rates of each tracking algorithm for all the sequences and Table 2 presents the average overlap rates. Overall, the proposed tracker performs favorably against state-of-the-art methods.

6 Conclusion

This paper presents a sparse coding-based discriminative appearance model for visual tracking. Two key contributions of this work are emphasized: First, we introduce structure information of both target and background in the local sparse representation of the object. This makes the sparse coding of the target object sparser and more discriminative, therefore enhancing the ability of the classifier in separating the object from the background. Second, to adapt our tracker to account for appearance change of both the target and the background and to alleviate the drift problem, we propose a selective update strategy which

prevents the dictionary being contaminated and keeps a good discriminative ability of the classifier for long-term tracking. Experimental results compared with several state-of-the-art methods on challenging sequences demonstrate the effectiveness and robustness of the proposed algorithm.

Acknowledgments This work was partially supported by Shenzhen Applied Technology Engineering Laboratory for Internet Multimedia Application under Grants Shenzhen Development and Reform Commission No. 2012720; Public Service Platform of Mobile Internet Application Security Industry under Grants Shenzhen Development and Reform Commission No. 2012720; Research on Key Technology in Developing Mobile Internet Intelligent Terminal Application Middleware under Grants No. JC201104210032A; Research on Key Technology of Vision Based Intelligent Interaction under Grants No. JC201005260112A.

References

- Adam, A., Rivlin, E., Shimshoni, I.: Robust fragments-based tracking using the integral histogram. In: Computer vision and pattern recognition, 2006 IEEE Computer Society Conference on, vol 1. IEEE, pp. 798–805 (2006)
- Avidan, S.: Ensemble tracking. *Comput. Vision Pattern Recognit.* **2**, 494–501 (2005). doi:[10.1109/CVPR.2005.144](https://doi.org/10.1109/CVPR.2005.144)
- Babenko, B., Yang, M.H., S. Belongie: Visual tracking with online multiple instance learning. In: Computer vision and pattern recognition, pp. 983–990 (2009). doi:[10.1109/CVPRW.2009.5206737](https://doi.org/10.1109/CVPRW.2009.5206737)
- Dinh, T.B., Medioni, G.: Co-training framework of generative and discriminative trackers with partial occlusion handling. In: Workshop on applications of computer vision, pp. 642–649 (2011). doi:[10.1109/WACV.2011.5711565](https://doi.org/10.1109/WACV.2011.5711565)
- Han, Y., Wu, F., Shao, J., Tian, Q., Zhuang, Y.: Graph-guided sparse reconstruction for region tagging. In: Computer vision and pattern recognition (CVPR), 2012 IEEE Conference on, pp. 2981–2988. IEEE (2012)
- Han, Y., Yang, Y., Yan, Y., Ma, Z., Sebe, N., Zhou, X.: Semisupervised feature selection via spline regression for video semantic recognition. *Neural Netw. Learn. Syst. IEEE Trans.* (2014). doi:[10.1109/TNNLS.2014.2314123](https://doi.org/10.1109/TNNLS.2014.2314123)
- Han, Z., Jiao, J., Zhang, B., Ye, Q., Liu, J.: Visual object tracking via sample-based adaptive sparse representation (adasr). *Pattern Recognit.* **44**(9), 2170–2183 (2011)
- Jepson, A.D., Fleet, D.J., El-Maraghi, T.F.: Robust online appearance models for visual tracking. *Pattern Anal. Mach. Intell. IEEE Trans.* **25**(10), 1296–1311 (2003)

9. Jia, X., Lu, H., Yang, M.H.: Visual tracking via adaptive structural local sparse appearance model. In: Computer vision and pattern recognition (CVPR), 2012 IEEE Conference on, pp. 1822–1829. IEEE (2012)
10. Kwak, S., Nam, W., Han, B., Han, J.H.: Learning occlusion with likelihoods for visual tracking. In: International conference on computer vision, pp. 1551–1558 (2011). doi:[10.1109/ICCV.2011.6126414](https://doi.org/10.1109/ICCV.2011.6126414)
11. Kwon, J., Lee, K.M.: Visual tracking decomposition. In: Computer vision and pattern recognition, pp. 1269–1276 (2010). doi:[10.1109/CVPR.2010.5539821](https://doi.org/10.1109/CVPR.2010.5539821)
12. Mairal, J., Bach, F., Ponce, J., Sapiro, G., Zisserman, A.: Non-local sparse models for image restoration. In: Computer vision, 2009 IEEE 12th International Conference on, pp. 2272–2279. IEEE (2009)
13. Mei, X., Ling, H.: Robust visual tracking using ℓ_1 minimization. In: Computer vision, 2009 IEEE 12th International Conference on, pp. 1436–1443. IEEE (2009)
14. Mei, X., Ling, H.: Robust visual tracking and vehicle classification via sparse representation. *Pattern Anal. Mach. Intell. IEEE Trans.* **33**(11), 2259–2272 (2011)
15. Ross, D.A., Lim, J., Sung Lin, R., Hsuan Yang, M.: Incremental learning for robust visual tracking. *Int. J. Comput. Vision* **77**, 125–141 (2008). doi:[10.1007/s11263-007-0075-7](https://doi.org/10.1007/s11263-007-0075-7)
16. Wang, D., Lu, H., Yang, M.H.: Online object tracking with sparse prototypes. *Image Process. IEEE Trans.* **22**(1), 314–325 (2013)
17. Wang, Q., Chen, F., Xu, W., Yang, M.H.: Online discriminative object tracking with local sparse representation. In: Workshop on applications of computer vision, pp. 425–432 (2012). doi:[10.1109/WACV.2012.6162999](https://doi.org/10.1109/WACV.2012.6162999)
18. Wang, Q., Chen, F., Yang, J., Xu, W., Yang, M.H.: Transferring visual prior for online object tracking. *Image Process. IEEE Trans.* **21**, 3296–3305 (2012). doi:[10.1109/TIP.2012.2190085](https://doi.org/10.1109/TIP.2012.2190085)
19. Wang, S., Lu, H., Yang, F., Yang, M.H.: Superpixel tracking. In: International Conference on Computer Vision, pp. 1323–1330 (2011). doi:[10.1109/ICCV.2011.6126385](https://doi.org/10.1109/ICCV.2011.6126385)
20. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. *Pattern Anal. Mach. Intell. IEEE Trans.* **31**(2), 210–227 (2009)
21. Wu, Y., Lim, J., Yang, M.H.: Online object tracking: A benchmark. In: Computer vision and pattern recognition (CVPR), 2013 IEEE Conference on, pp. 2411–2418. IEEE (2013)
22. Zhang, L., Gao, Y., Xia, Y., Dai, Q., Li, X.: A fine-grained image categorization system by cellet-encoded spatial pyramid modeling. *Ind. Electron. IEEE Trans.* (2014). doi:[10.1109/TIE.2014.2327558](https://doi.org/10.1109/TIE.2014.2327558)
23. Zhang, L., Han, Y., Yang, Y., Song, M., Yan, S., Tian, Q.: Discovering discriminative graphlets for aerial image categories recognition. *IEEE Trans. Image Process.* **22**(12), 5071–5084 (2013)
24. Zhang, L., Ji, R., Xia, Y., Zhang, Y., Li, X.: Learning a probabilistic topology discovering model for scene categorization. *Neural Netw. Learn. Syst. IEEE Trans.* (2014). doi:[10.1109/TNLS.2014.2347398](https://doi.org/10.1109/TNLS.2014.2347398)
25. Zhang, L., Song, M., Liu, X., Bu, J., Chen, C.: Fast multi-view segment graph kernel for object classification. *Signal Process.* **93**(6), 1597–1607 (2013)
26. Zhong, W., Lu, H., Yang, M.H.: Robust object tracking via sparsity-based collaborative model. In: Computer vision and pattern recognition (CVPR), 2012 IEEE Conference on, pp. 1838–1845. IEEE (2012)