Object Tracking Based on Weighted Local Sub-space Reconstruction Error

Xianyou Zeng[1,2], Long Xu[3], Shaohai Hu[1,2], Ruizhen Zhao[1,2*], Wanli Feng[1,2]

[1] Institute of Information Science, Beijing Jiaotong University, Beijing, China, 100044

[2] Key Laboratory of Advanced Information Science and Network Technology of Beijing, Beijing, China

[e-mail: 14112057@bjtu.edu.cn, shhu@bjtu.edu.cn, rzhzhao@bjtu.edu.cn, 15120327@bjtu.edu.cn]

[3] Key Laboratory of Solar Activity, National Astronomical Observatories, Chinese Academy of Sciences, Beijing, China, 100012

[e-mail: lxu@nao.cas.cn]

*Corresponding author: Ruizhen Zhao

Received December 25, 2017; revised March 15, 2018; accepted April 4, 2018; published February 28, 2019

Abstract

Visual tracking is a challenging task that needs learning an effective model to handle the changes of target appearance caused by factors such as pose variation, illumination change, occlusion and motion blur. In this paper, a novel tracking algorithm based on weighted local sub-space reconstruction error is presented. First, accounting for the appearance changes in the tracking process, a generative weight calculation method based on structural reconstruction error is proposed. Furthermore, a template update scheme of occlusion-aware is introduced, in which we reconstruct a new template instead of simply exploiting the best observation for template update. The effectiveness and feasibility of the proposed algorithm are verified by comparing it with some state-of-the-art algorithms quantitatively and qualitatively.

Keywords: visual tracking, sub-space reconstruction error, generative weights, template update

1. Introduction

With the rapid development of computer hardware level, image processing technology and artificial intelligence, computer vision has been widely applied in many fields of human activity and life, such as information retrieval (e.g. [34-36]), intelligent classification (e.g. [37-38]), decision making system and so on. Visual tracking plays a critical role in computer vision due to its wide range of applications such as motion analysis, video surveillance, vehicle navigation, human-computer interaction, aeronautics and astronautics. Although significant progress has been made in the past decades, robust object tracking is still a challenging problem due to numerous factors such as partial occlusion, illumination variation, motion blur and pose change.

The existing object tracking methods are classified into two categories: generative methods (e.g. [1-4]) and discriminative methods (e.g. [5-8], [29]). The discriminative methods transform the tracking into a classification problem and distinguish the target and the background by modeling a conditional distribution. The generative tracking methods aim to learn a visual model representing the appearance of the target being tracked and perform the tracking by looking for the image area that most matches the target object. It has been shown that generative models achieve higher generalization when training data is limited [17], while discriminative models perform better if the training set is large [18]. In addition, many hybrid tracking methods [16], [28] have been proposed to take the advantages of both generative and discriminative models.

In generative tracking methods, the object appearance representation is very important and greatly affects the likelihood estimation. Many representation schemes have been proposed, such as template-based (see [1], [4], [9]), sub-space-based (see [2], [10-11]), sparse representation-based (see [3], [12-13], [31], [33]) and feature-based (see [5-7], [15]) models. Among these representation methods, sub-space representation models provide a compact concept for the tracked object and promotes other visual tasks. Ross et al. [2] proposed an incremental visual tracking (IVT) method which is robust to in-plane rotation, illumination variation, scale change and pose change. However, it has been shown that the IVT method is sensitive to partial occlusion.

Considering the partial occlusion, quite a few attempts have been made. Adam et al. [1] proposed a fragment-based tracking approach, where the target region is partitioned into several fragments and partial occlusion is handled by combining the voting maps of these fragments. The authors of [19] extended the idea of fragment and presented local sensitive histogram to overcome multiple challenges including illumination changes and partial occlusion for robust tracking. In [20], the bag of words model was introduced into visual tracking to address partial occlusion. In [3] and [13], partial occlusion was modeled by sparse representation of trivial templates. The authors in [32] use a regularized robust sparse coding (RRSC) to robustly deal with occlusion and noise.

In this paper, a new visual tracking algorithm based on weighted local sub-space reconstruction error is proposed. First, candidate targets are represented through the PCA sub-space. Second, patch-based generative weights are computed from structural reconstruction error. Based on the patch-based representation error of the PCA sub-space and the patch-based generative weight, an effective tracking method based on particle filter is developed and used for the prediction of the tracked target. In addition, a template update scheme of occlusion-aware is introduced, which can handle appearance changes caused by occlusion or other disturbances during tracking. The main contributions of this work are

outlined below.

(1) A novel tracking algorithm based on weighted local sub-space reconstruction error is presented in this paper.

(2) A generative weight calculation method based on structural reconstruction error is inserted to deal with appearance changes in the tracking process.

(3) A template update scheme of occlusion-aware is introduced to avoid bringing noise into the template set by reconstructing a new template for template update.

The rest of the paper is arranged as follows. The related work is briefly introduced in section 2. The proposed tracking algorithm is described in detail in section 3. The comparisons between the proposed tracking algorithm and some state-of-the-art tracking algorithms are presented in section 4. Finally, the concluding remark is given in section 5.

2. Related work

A lot of works have been done in visual tracking and good reviews can be seen from [21-22]. Here, we discuss the methods that are most related to our work, namely, incremental sub-space learning based trackers and sparse representation based trackers.

2.1 incremental sub-space learning based trackers

In recent years, visual tracking based on sub-space learning ([2], [10-11], [23]) has received considerable attention. The IVT method [2] incrementally learns and updates a low dimensional PCA sub-space representation, which online adapts to the appearance changes of the target. Several experimental results demonstrate that the IVT method is effective in dealing with appearance changes caused by in-plane rotation, scale and illumination variations. However, it has the following drawbacks. Firstly, the IVT method assumes that the reconstruction error is Gaussian distributed with small variances. The assumption does not hold as partial occlusion occurs, resulting in compromised performance of tracking. Secondly, the IVT method doesn't have an effective update scheme. It directly updates the sub-space model with new observations without detecting and processing outliers. To solve partial occlusion, Lu et al. [10] introduced l_1 noise regularization into the PCA reconstruction. Wang et al. [11] utilized the linear regression with Gaussian-Laplacian assumption to deal with outliers for reliable tracking. Pan et al. [23] employed l_0 norm to regularize the linear coefficients of incrementally updated linear basis to remove the redundant features of basis vectors. Zhou et al. [39] developed a tracking algorithm based on weighted sub-space reconstruction error, which can take the advantages of sparse representation and sub-space learning model. Different from the aforementioned holistic models, a novel tracking method via weighted local sub-space reconstruction error is proposed in this paper.

2.2 sparse representation based trackers

Sparse representation has been widely studied and applied to visual tracking. Mei and Ling [3] sparsely represented each candidate object in a space spanned by target templates and trivial templates to tackle occlusion and corruption challenges. Liu et al. [24] incorporated group sparsity to boost the robustness and efficiency of the tracker. In [13], a faster version of [3] was proposed, which was further extended to handle multi-task in [25]. The works in

[26] and [27] combined sparse representation and incremental sub-space learning for object tracking by reconstructing a new template and exploiting it for template update. Our method is motivated by the works in [10], [26], [27]. We use a patch-based generative weight to adjust the patch-based reconstruction error of PCA sub-space model. To get rid of image noise, we introduce an occlusion-aware template update scheme for the object tracking.

2.3 deep networks based trackers

Recently, deep neural network has been introduced into tracking for its powerful feature learning capability. In [40], a neural network with three convolution layers was proposed for visual tracking, which learned feature representation and classifier simultaneously. In [41] and [30], a convolution neural network (CNN) was respectively pre-trained on image classification dataset, and then it was transferred to visual tracking. In [42] and [43], the authors directly trained their CNNS on large amounts of video sequences.

3. Proposed visual tracking algorithm

Object tracking can be considered as a Bayesian filtering process. Let the target state $x_t = \{l_x, l_y, \mu_1, \mu_2, \mu_3, \mu_4\}$, where l_x, l_y, μ_1, μ_2, μ_3, μ_4 denote the horizontal and vertical translations, rotation angle, scale, aspect ratio, and skew parameter respectively. Given the observation set $Y_t = \{y_1, y_2, ..., y_t\}$ up to frame t, we estimate the state of the object x_t recursively

$$p(x_t | Y_t) \propto p(y_t | x_t) \int p(x_t | x_{t-1}) p(x_{t-1} | Y_{t-1}) dx_{t-1}, \tag{1}$$

where $p(x_t | x_{t-1})$ is the motion model that represents the state transition of the object between the two consecutive frames, and $p(y_t | x_t)$ denotes the observation model that estimates the likelihood of the observation y_t at state x_t. Particle filtering is an effective implementation of Bayesian filtering. The optimal state is computed by the maximum a posterior estimation (MAP) of N samples,

$$\hat{x}_t = \arg\max_{x_t^i} p(y_t | x_t^i) p\left(x_t^i \middle| \hat{x}_{t-1}\right), \tag{2}$$

where x_t^i is the $i-th$ sample of frame t. The observation model $p(y_t | x_t^i)$ in (2) is crucial for robust tracking. In this paper, the observation model is estimated through a weighted local PCA sub-space model. **Fig. 1** shows the observation model of our method, which will be explained in detail as follows.

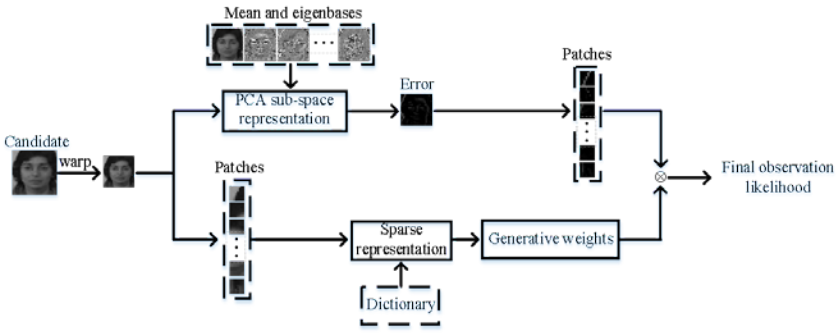

Fig. 1. The weighted local PCA sub-space observation model.

3.1 Motivation of this work

We assume that target appearance can be represented by an image sub-space with corruption,

$$y = Uz + e = \left[U\, I_q \right] \begin{bmatrix} z \\ e \end{bmatrix}, \tag{3}$$

where $y \in R^{q\times 1}$ denotes an observation vector, U represents a matrix of column basis vectors, $I_q \in R^{q\times q}$ is an identity matrix, z is the coefficient vector of basis vectors, and e indicates the error term modeled by a Laplacian noise. The coefficient vector z and the error term e can be computed by

$$[z, e] = \min_{z,e} \frac{1}{2} \left\| \bar{y} - Uz - e \right\|_2^2 + \lambda \left\| e \right\|_1, \tag{4}$$

where $\bar{y} = y - \mu$, μ is the mean vector. After obtaining z, the observation likelihood of the observation y can be measured by the reconstruction error

$$\begin{aligned} p\left(y \middle| x\right) &= \exp\left(-\left\| \bar{y} - Uz \right\|_2^2 \right), \\ &= \exp\left(-\left\| E_{PCA} \right\|_2^2 \right) \end{aligned} \tag{5}$$

where E_{PCA} is the reconstruction error. Eq. (5) is a holistic estimation method. It is usually sensitive to partial occlusion.

Inspired by local models, we reorganize the reconstruction error E_{PCA} as the connection of M local feature vectors $E_{PCA} = \left[t_1^T, t_2^T, ..., t_M^T \right]^T$, where $t_i \in R^{l\times 1}$ is a column vector denoting the $i-th$ local patch of the reconstruction error, and $M = q/l$. Then, Eq. (5) can be reformulated as

$$\begin{aligned} p(y|x) &= \exp\left(-\left\|E_{PCA}\right\|_2^2\right) \\ &= \exp\left[-\left(\left\|t_1\right\|_2^2+\left\|t_2\right\|_2^2+...+\left\|t_M\right\|_2^2\right)\right] \\ &= \exp\left[-\left(1\bullet\left\|t_1\right\|_2^2+1\bullet\left\|t_2\right\|_2^2+...+1\bullet\left\|t_M\right\|_2^2\right)\right] \end{aligned} . \quad (6)$$

From (6), we can see that the penalty weight of each part t_i is 1, which means that holistic model deals with the observation uniformly and treats each part of the observation equally regardless of the condition of each part of the observation during the tracking. It does not hold when the observation is subjected to some impulse noise, such as partial occlusion and local illumination variations. Based on the above discussion, we aim to learn a set of generative weights via sparse coding of each local patch of the observation to penalize each part of the reconstruction error E_{PCA} differently.

3.2 Weight learning by structural reconstruction error

1) Preprocessing: Each input image is adjusted to a standard size of 32×32 pixels and represented by gray-scale features. We employ a sliding window to sample a bank of non-overlapping local image patches $X=\{x_1,x_2,...,x_M\}\in R^{l\times M}$ in the input image, where x_i is the $i-th$ column local vectorized patch, l is the dimension of patch vectors and M is the number of local patches. Each patch x_i is preprocessed by l_2 normalization.

2) Templates: Initially, we use the CT algorithm [7] to track the first n frames. Tracking results are used to form the templates $T=[T_1,T_2,...,T_n]$. Each template is split into local image patches. Then a dictionary $D=[d_1,d_2,...,d_{M\times n}]\in R^{l\times(M\times n)}$ can be obtained for encoding local patches of each candidate target. Each element d_i is a normalized column vector which corresponds to a local patch cropped from T.

3) Weight learning: Given a candidate target, each local image patch x_i of it can be encoded using the elements of the dictionary D by solving

$$\min_{\partial_i}\left\|x_i-D\partial_i\right\|_2^2+\lambda_1\left\|\partial_i\right\|_1, \quad (7)$$

where $\partial_i\in R^{(M\times n)\times1}$ is the sparse code of x_i, λ_1 is a control parameter.

In order to take into account the spatial layout, the dictionary D can be written as

$$\begin{aligned} D &= \left[d_1,...,d_M,d_{M+1},...,d_{2M},...,d_{(n-1)M+1},...,d_{nM}\right], \\ &= \left[D_i,D_{other}\right] \end{aligned} \quad (8)$$

where $D_i = \left[d_i, d_{M+i}, ..., d_{(n-1)M+i} \right] \in R^{l \times n}$, $1 \le i \le M$, D_{other} is made up of the other elements of D. Accordingly, the sparse code ∂_i can be denoted as $\partial_i = \left[\beta_i^T, \beta_{other}^T \right]^T$, where $\beta_i = \left[\partial_i^i, \partial_i^{M+i}, ..., \partial_i^{(n-1)M+i} \right]^T \in R^{n \times 1}$ is the sparse coefficients of the patch x_i under sub-dictionary D_i, β_{other} is the sparse coefficients of the patch x_i under sub-dictionary D_{other}.

The weight of x_i can be obtained by

$$w_i = \left\| x_i - D\left(\omega_i \otimes \partial_i \right) \right\|_2^2 + \gamma \left\| D\left(\left(1 - \omega_i \right) \otimes \partial_i \right) \right\|_1, \tag{9}$$

where $\omega_i = \left[\omega_i^1, \omega_i^2, ..., \omega_i^{(M \times n)} \right]^T$ is an indicator vector, $\otimes$ is the element-wise multiplication, and γ is a control parameter. Each element of ω_i is obtained by

$$\omega_i^j = \begin{cases} 1, j = i, M+i, ..., \left(n-1 \right) M + i \\ 0, others \end{cases}. \tag{10}$$

The flow of weight calculation is shown in Fig. 2. In (9), the first term is the reconstruction error of x_i under sub-dictionary D_i, and the second term is the sparse reconstruction of x_i under sub-dictionary D_{other} which is a penalty term. If the candidate target is perfect, both the first and the second terms on the right side of (9) are very small. Otherwise, they become very large. In this way, we can learn a set of different weights for local patches of the candidate target which satisfies $\sum_{i=1}^{M} w_i = 1$. The main advantage lies in that the structural similarity between the candidate target and templates is fully considered. Then, the observation likelihood of the candidate target can be measured by

$$p = \exp\left[-\left(\sum_{i=1}^{M} w_i \bullet \left\| t_i \right\|_2^2 + \tau \left\| e \right\|_0 \right) \right], \tag{11}$$

where $\left\| e \right\|_0$ denotes the number of the outliers of the candidate target, τ is a constant. After the observation likelihood of all candidate targets is obtained, the candidate target with the biggest observation likelihood is taken as the tracked target.

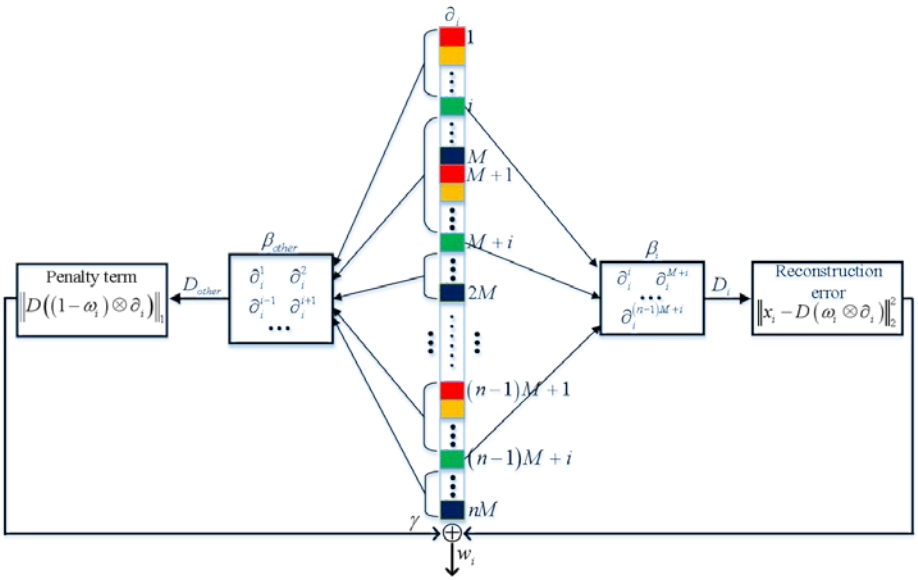

Fig. 2. Overall diagram of weight calculation.

3.3 Model updating

To adapt to the appearance change of a target object, the observation model needs to be updated dynamically. The model updating includes the updating of PCA sub-space and templates.

1) PCA sub-space updating: since the error term e can identify some outliers (e.g. partial occlusion, illumination change), we adopt the strategy proposed by [11] to update PCA sub-space (including PCA basis U and the mean vector μ). After we obtain the tracked target of each frame y_o, the tracked target is reconstructed by

$$y_r^i = \begin{cases} y_o^i, e_o^i = 0 \\ \mu_o^i, e_o^i \neq 0 \end{cases}, \tag{12}$$

where y_r is the reconstructed vector of the tracked target of each frame, e_o is the error term corresponding to the tracked target y_o. y_r is cumulated and used to incrementally update U and μ.

2) Occlusion-aware template updating: In this study, we give each template T_i a weight a_i which has an initial value of 1. After obtaining the reconstructed vector of the tracked target in each frame, we update the value of the weight a_i as follows.

$$a_i = a_i e^{-\theta_i}, \tag{13}$$

where θ_i is the angle between T_i and y_r. In [26] and [27], sparse representation and incremental sub-space learning are used to reconstruct a new template for the template

update, which can avoid introducing noise into templates T. Inspired by the work [26] and [27], we propose an effective template update method. The template updating method includes two operations: template replacement and weight updating. For template replacement, we first get the coefficient z of the tracked target in each frame, and then reconstruct a new template through

$$T^* = Uz + \mu. \tag{14}$$

T^* replaces the template that has the least weight. During weight updating, the median weight of the rest $n-1$ templates is used as the weight of T^*. Algorithm 1 summarizes our method and its process is shown in Fig. 3.

Algorithm 1. Our proposed tracker

Inputs: Initial target state $\hat{x}_1$, number of templates n, update interval m.

1. **Initialization:**
   template set T and reconstructed target set $\Phi = \phi$;
2. Construct the dictionary D with the local patches of templates;
3. **For** $t = n+1, ..., s$ **do**
4. Produce N candidate targets $\{x_t^i\}_{i=1}^N$ with the motion model $p\left(x_t^i \middle| \hat{x}_{t-1}\right)$;
5. Calculate weights for local patches of each candidate target using Eq. (9);
6. Estimate the observation likelihood of candidate targets according to Eq. (11);
7. Determine the tracked target $\hat{x}_t$ using the biggest observation likelihood;
8. Obtain the reconstructed vector x_r of the tracked target $\hat{x}_t$ using Eq. (12), and then $\Phi = [\Phi, x_r]$;
9. **If** size(Φ)== m **then**
10. Update the PCA sub-space with Φ, and empty Φ;
11. Get the new template T^* through Eq. (14) and update the template set;
12. Rebuild the dictionary D with local patches of the new template set;
13. **End if**
14. **End for**

Outputs: Tracked targets $\hat{x}_t$, $t = 2, ..., s$.

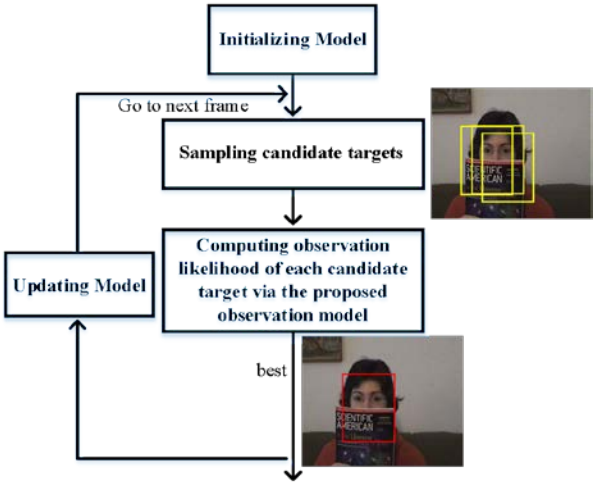

Fig. 3. Overview diagram of the proposed tracking approach.

4. Experimental results

4.1 Implementation details

The proposed algorithm is executed in MATLAB and has a running speed of 1.1 frames per second on a 3.4 GHZ i7-4770 core PC with 16GB memory. The number of templates n is 10. For all experiments, the number of patch M is 16. The variable λ in (4), λ_1 in (7), γ in (9) and τ in (11) are set to 0.1, 0.01, 0.01 and 0.05 respectively. $\{l_x, l_y, \mu_1, \mu_2, \mu_3, \mu_4\}$ is fixed to $\{6, 6, 0.01, 0, 0.005, 0\}$. The maximum number of PCA basis vectors is set to 16. The number of particles N is set to 600 for balancing effectiveness and speed. The proposed observation model is updated every 5 frames.

4.2 Quantitative evaluation

In order to evaluate the effectiveness and feasibility of the proposed tracker (WLSRE), experiments are carried out on 26 publicly available sequences [22] which contain different challenging situations (e.g. partial occlusion, illumination variation, etc.). Our WLSRE tracker is compared with seven state-of-the-art trackers: IVT [2], LSST [11], SPT [10], ASLA [26], CT [7], KCF [15], and TGPR [14].

Two metrics are measured to evaluate the proposed algorithm with other state-of-the-art methods. The first metric is the center location error which reflects the error between the center of the tracking bounding box and the center of the ground truth bounding box. The second one is the overlap rate, defined as $score = \frac{area(R_T \cap R_G)}{area(R_T \cup R_G)}$, where R_T and R_G mark the tracking bounding box and the ground truth bounding box. The average center location errors are reported in Table 1, where smaller center errors mean more accurate tracking results. The average overlap rates are listed in Table 2, where the larger the value, the more accurate the tracking result. It can be concluded from these tables that the proposed

tracker is effective and feasible.

Table 1. Average center location error (in pixels). Top three results are shown in color fonts.

sequence	LSST	SPT	IVT	ASLA	CT	KCF	TGPR	WLSRE
Basketball	107.94	5.97	107.11	82.63	89.11	8.07	9.43	10.23
doll	12.08	40.01	32.65	11.84	21.82	8.27	5.97	6.79
boy	109.29	57.62	91.25	106.07	9.03	2.67	3.38	14.69
Car4	2.66	113.84	2.15	1.59	86.03	9.47	6.11	2.29
Cardark	1.37	61.14	8.43	1.54	119.22	5.76	2.13	2.04
David	95.53	19.46	4.82	5.07	10.49	8.06	5.31	6.12
David2	5.10	50.71	1.17	1.45	76.70	2.29	2.05	2.39
David3	6.05	6.36	51.95	87.76	88.66	4.06	6.93	7.27
Dog1	4.41	18.44	3.46	4.87	6.99	4.15	5.86	5.11
Dudek	9.07	102.07	9.62	15.26	26.53	11.38	17.27	8.20
Faceocc1	15.17	30.73	18.42	78.06	25.82	15.98	13.73	14.36
Fish	3.99	22.53	5.67	3.85	10.68	4.08	5.62	4.39
Fleetface	34.79	180.02	62.23	31.09	58.43	26.37	29.22	24.08
Football	30.14	36.53	14.34	15.00	11.91	14.80	5.94	19.88
Faceocc2	10.63	50.77	7.42	19.34	18.95	7.67	7.56	13.52
Freeman1	58.21	46.19	11.64	105.66	118.72	94.62	9.34	9.43
Freeman3	3.31	60.16	35.76	3.17	65.32	19.57	88.28	1.89
Football1	11.49	10.77	24.47	12.22	20.71	5.16	11.49	7.57
Jogging2	134.95	50.07	138.22	169.86	139.30	144.03	5.39	4.16
Jumping	60.43	31.15	61.56	46.08	47.73	25.99	54.16	5.09
Lemming	177.70	172.59	181.79	178.82	32.25	77.97	150.32	18.29
Mhyang	2.10	19.62	1.87	1.70	13.28	3.92	4.43	2.64
Singer1	2.77	225.79	11.31	3.29	15.53	12.59	120.29	2.48
Singer2	14.59	241.07	175.46	175.28	127.31	10.26	10.13	15.63
Walking2	61.28	21.93	2.46	37.42	58.53	29.57	5.90	1.47
Walking	1.62	5.30	1.61	1.89	6.95	4.26	5.01	2.29
Average	37.56	64.65	41.03	46.19	50.23	21.58	22.74	8.17

Table 2. Average overlap rate. Top three results are shown in color fonts.

sequence	LSST	SPT	IVT	ASLA	CT	KCF	TGPR	WLSRE
Basketball	0.0828	0.7609	0.1051	0.3839	0.2563	0.6716	0.6484	0.6420
doll	0.6960	0.3856	0.4345	0.8262	0.4529	0.5348	0.5746	0.6703
boy	0.3579	0.3057	0.2602	0.3694	0.5902	0.7863	0.7570	0.6366
Car4	0.8808	0.1249	0.8755	0.7536	0.2135	0.4846	0.4978	0.8817
Cardark	0.8357	0.1883	0.6632	0.8492	0.0031	0.6224	0.8073	0.8017
David	0.1732	0.4065	0.6449	0.7485	0.4956	0.5384	0.5863	0.6429
David2	0.4892	0.1372	0.7015	0.8964	0.0025	0.8177	0.8224	0.7065
David3	0.3965	0.7550	0.4806	0.4324	0.3064	0.7748	0.7363	0.6922

Dog1	0.7613	0.4417	0.7411	0.7232	0.5352	0.5519	0.6031	0.7446
Dudek	0.7956	0.4055	0.7528	0.7366	0.6470	0.7284	0.7011	0.7903
Faceocc1	0.7680	0.6336	0.7264	0.3186	0.6369	0.7539	0.7772	0.7754
Fish	0.7109	0.4914	0.7715	0.8505	0.7157	0.8394	0.8151	0.7060
Fleetface	0.6125	0.0324	0.4574	0.5657	0.5214	0.5847	0.5592	0.6391
Football	0.3425	0.1678	0.5566	0.5309	0.6100	0.5447	0.6978	0.5557
Faceocc2	0.4630	0.2702	0.7273	0.6455	0.6078	0.7511	0.7699	0.6754
Freeman1	0.2585	0.1897	0.4256	0.2652	0.1445	0.2151	0.4091	0.5285
Freeman3	0.6678	0.2734	0.3943	0.7460	0.0025	0.3220	0.0534	0.6167
Football1	0.4805	0.5147	0.5572	0.4927	0.2270	0.7232	0.6160	0.5892
Jogging2	0.1342	0.1438	0.1440	0.1422	0.1054	0.1258	0.7708	0.6920
Jumping	0.1260	0.1831	0.1223	0.2266	0.0431	0.2761	0.0859	0.6523
Lemming	0.1439	0.1183	0.1386	0.1448	0.5492	0.3836	0.2205	0.6335
Mhyang	0.8336	0.5641	0.7963	0.9156	0.6002	0.7966	0.7696	0.8555
Singer1	0.8768	0.1861	0.5738	0.7918	0.3477	0.3549	0.2282	0.7701
Singer2	0.6166	0.0351	0.0429	0.0438	0.0826	0.7315	0.7272	0.6226
Walking2	0.3519	0.2973	0.7948	0.3713	0.2658	0.3874	0.5964	0.7997
Walking	0.7571	0.5973	0.7660	0.7717	0.5205	0.5298	0.5937	0.7208
Average	0.5236	0.3311	0.5252	0.5593	0.3647	0.5704	0.5932	0.6939

4.3 Qualitative evaluation

We choose some tracking results from the test sequences for qualitative evaluation. The results are shown in **Figs. 4-9**, which exhibit the feasibility and effectiveness of the proposed method.

Heavy occlusion: We test several sequences (Faceocc1, Faceocc2, Jogging2, David3, Walking2) with heavy or long-time partial occlusion. In the Faceocc1 sequence, a woman frequently uses a book to occlude her face. With the exception of ASLA and SPT, the remaining six trackers perform well. For the Faceocc2 sequence, IVT, KCF, TGPR and our WLSRE tracker can successfully track the target. In the Jogging2 sequence, the target meets with heavy occlusion. LSST, KCF, CT, ASLA, IVT and SPT are unable to recapture the target and suffer significant deviation when the person passes through the obstacle and reappears (see #62, #150 and #307). In contrast, TGPR and our WLSRE tracker precisely track the target in this sequence. In the David3 sequence, TGPR, KCF, SPT and our tracker successfully deal with heavy occlusion and perform well in this sequence (seen from #83, #150 and #220). In the Walking2 sequence, the walking woman is occluded by a man for a long period of time. Only IVT, TGPR and our WLSRE tracker successfully complete the tracking task, which can be seen from #100, #240, #400 and #500. The robustness of our WLSRE tracker against occlusion can be attributed to two reasons: (1) patch-based weights impose larger penalties on occluded parts and reduce the influence of occlusion; (2) occlusion-aware template update scheme effectively prevents noise from entering the template set.

Fig. 4. Screenshots of the tracking results on 5 sequences with occlusion.

Illumination variations: **Fig. 5** provides the results of some sequences with illumination changes. In the Cardark sequence, CT can't perform well (seen from #100, #250, #300 and #393). SPT loses the tracked object after 200 frames (seen from #250 and #300). IVT drifts away from the correct location of the target at last (see #393). KCF, TGPR, ASLA, LSST and our WLSRE tracker successfully capture the target trajectory of all frames. In the Fish sequence, the illumination changes obviously. All the methods except SPT robustly overcome this difficulty and achieve accurate tracking. For the Mhyang sequence, IVT and the proposed WLSRE method are superior to other methods and obtain better tracking results. In the David sequence, LSST cannot follow the tracked object rightly during tracking (seen from #409 and #499). CT and SPT exhibit a small deviation in some frames, which can be seen from #499 and #749 respectively. TGPR, KCF, ASLA, IVT and our WLSRE tracker successfully track the target throughout this sequence and ASLA achieves the best performance in terms of both location and scale. In the Car4 sequence, SPT and CT drift off the target when there is a large illumination variation at frames #200 and #240. Moreover, the target undergoes scale variation. While TGPR and KCF can successfully estimate the location of the target, they do not deal with scale changes of it well (seen from #400 and #500). Due to the use of incremental PCA sub-space, IVT, LSST, ASLA and the proposed algorithm achieve good performance in dealing with the appearance change caused by illumination and scale changes.

Fig. 5. The comparison of qualitative results on 5 sequences with illumination changes.

Scale variations: **Fig. 6** shows some of the results of four sequences containing scale variations. The target in the doll sequence experiences a long time scale change and rotation. SPT drifts away and finally loses the target (seen from #1000, #2250 and #3500). CT and IVT fail to precisely locate the target at the end (see #3500). Except for the three methods, the other five methods perform well. For the Dog1 and Walking sequences, LSST, IVT, ASLA and our WLSRE tracker are superior to others. The Singer1 sequence is very difficult due to large changes in light and scale. SPT runs poorly (see #200, #250 and #351). IVT slightly deviates from the target location (see #250 and #351). TGPR is incapable of tracking the target properly when drastic illumination changes occur (see #100, #200 and #351). As can be seen from frames #250 and #351, KCF and CT can't deal with scale change well. ASLA, LSST and our WLSRE tracker robustly overcome the challenges caused by the changes in illumination and scale, and accurately locate the target over the whole sequence.

Fig. 6. The representative results when the tracked targets experience scale variation.

Background clutters: Fig. 7 gives some representative tracking results of Football, Singer2, Basketball and Dudek sequences, where the targets are disturbed by background clutters. The target in the Football sequence not only has a very similar appearance to the background, but also is affected by occlusion and rotation. SPT can't track the target accurately (seen from #100, #140 and #200). LSST drifts to the background (e.g. #200). CT, IVT, ASLA, KCF, TGPR and our WLSRE method can successfully track most frames. For the Singer2 sequence, the target being tracked goes through numerous challenges including background clutters, illumination variations, deformation and rotation. SPT, IVT, ASLA and CT fail to track when the target rotates (e.g. #89). Instead, KCF, TGPR, LSST and our WLSRE method win mentioned challenges and exactly keep track of the target on this sequence (seen from #89, #200 and #300). In the Basketball sequence, TGPR, KCF, SPT and our WLSRE tracker persistently track the target, while other methods fail. The Dudek sequence involves a number of challenges of background clutters, occlusion and pose change. SPT fails in many frames (seen from #600, #800 and #1070). Except for SPT, other methods can stably track the target, among which LSST and our WLSRE method run best.

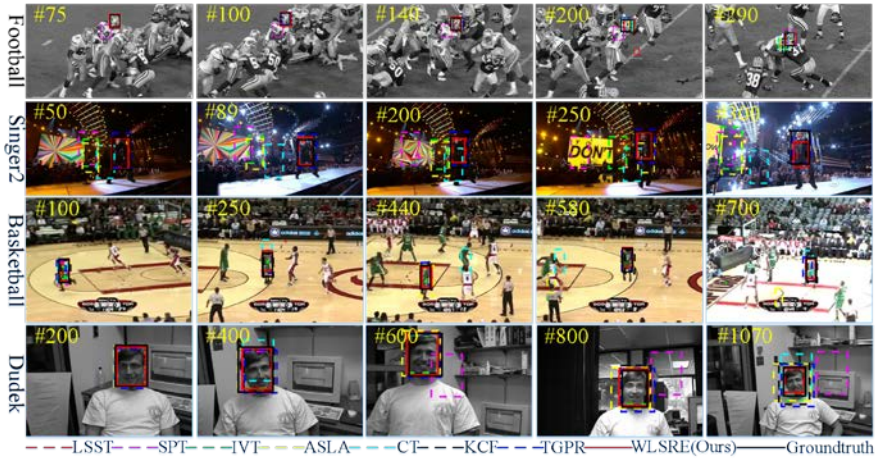

Fig. 7. The results of all evaluated trackers on 4 sequences with background clutters.

Fast motion: Fast motion of the target object leads to blurred image appearance which is difficult to tackle in tracking task. Fig.8 illustrates the tracking results on the Fleetface, boy, Jumping and Lemming sequences. For Fleetface sequence, most trackers can successfully track most of the frames except for SPT. In the boy sequence, the target suffers from fast motion, motion blur, as well as rotation. SPT, ASLA, IVT and LSST lose track of the target when motion blur occurs, whereas KCF, TGPR and our WLSRE method perform favorably (see #375 and #570). In the Jumping sequence, the target moves so drastically that it is difficult to predict its location. Only the proposed tracker successfully tracks the target in the entire sequence (seen from #34, #175 and #313). The Lemming sequence is very challenging for visual tracking as the target meets with multiple challenges of fast motion, heavy occlusion, together with out-of-plane rotation. We can note that CT and our method perform more excellently than other methods (e.g. #400, #850 and #1336).

Fig. 8. Qualitative evaluation of different tracking algorithms on 4 sequences with fast motion.

Rotation: **Fig. 9** presents a few results for four sequences with rotation challenge. In the David2 sequence, KCF, TGPR, IVT, ASLA and our WLSRE method perform well and achieve outstanding performance. In the Freeman1 sequence, the face of a man experiences large scale changes and rotation. Only IVT, TGPR and our WLSRE tracker can track the target of most frames, and our WLSRE method achieves the best overlap rate. The Freeman3 sequence includes scale variation, in-plane and out-of-plane rotations, which makes this tracking task difficult. We can see that only LSST, ASLA and our WLSRE tracker can win these difficulties and exactly track the target throughout the frames, which has been verified on frames #146, #278, #350 and #460. There are in-plane rotation and out-of-plane rotation in the Football1 sequence. Along with rotation is background clutter. SPT performs unsteadily and shakes around the target position (seen from #14, #40 and #74). CT doesn't track well from the beginning (see frame #14). LSST is unable to lock the tracked object well in the latter half of the sequence (see #40 and #74). IVT, KCF and TGPR drift away to the background region at the end (see #74). Our WLSRE method stably tracks the target till the end.

Fig. 9. Sample results of all compared trackers on several sequences with rotation.

4.4 Evaluation on OTB-50

In order to make the experiments more convincing, we also run the proposed method on the object tracking benchmark (OTB) [22]. The trackers that are compared with our method include KCF [15], TGPR [14], VTD [4], DLT [30], ASLA [26], IVT [2], LSST [11], SPT [10], CT [7], FCNT [41], and boostingtrack [27]. Precision and success plots are used for the evaluation of the performance of all compared trackers. Fig. 10 reports the performance (in terms of precision plot, success plot, precision score and success score) of the 12 trackers on 50 videos. We can observe that our method obtains more satisfying and more promising results than holistic models such as LSST, SPT and IVT.

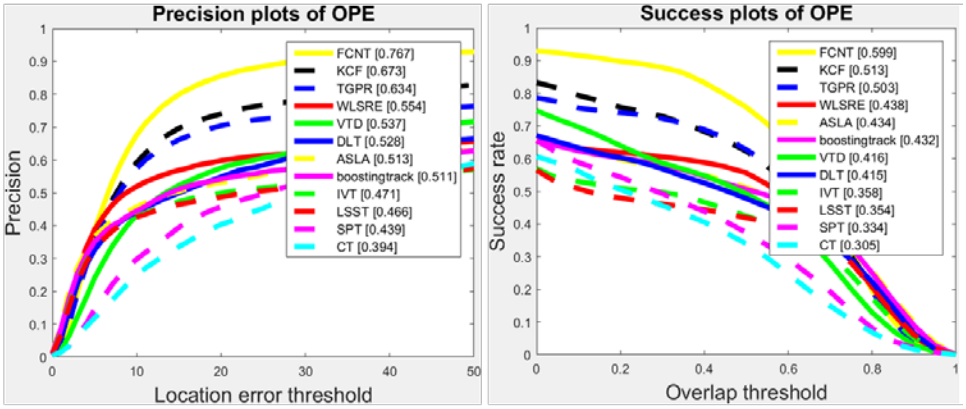

Fig. 10. Performance evaluation of the 12 trackers on OTB-50.

5. Conclusions

This paper presents a novel tracking algorithm based on weighted local sub-space reconstruction error. In this work, we explicitly take partial occlusion and other interference factors into account by learning a set of weights for local patches of PCA sub-space reconstruction error. Under a generative model, the weights are calculated through the structural errors and reflect the spatial similarity between the candidate targets and the templates. At the same time, an occlusion-aware template update method is introduced to enhance the performance of the tracker. Extensive evaluation demonstrates the effectiveness and feasibility of the proposed algorithm. Our future work will focus on integrating effective detection modules for persistent tracking. Moreover, a particle selection mechanism will be introduced to accelerate our tracker.

References

[1] A. Adam, E. Rivlin and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 798-805, June 17-22, 2006. Article (CrossRef Link)

[2] 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, May, 2008. Article(CrossRef Link)

[3] X. Mei and H. Ling, "Robust visual tracking using l1 minimization," in *Proc. of International Conference on Computer Vision*, pp. 1436-1443, September 29-October 2, 2009. Article(CrossRef Link)

[4] J. Kwon and K. M. Lee, "Visual tracking decomposition," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1269-1276, June 13-18, 2010. Article(CrossRef Link)

[5] B. Babenko, M. H. 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, August, 2011. Article(CrossRef Link)

[6] S. Hare, S. Golodetz and A. Saffari, "Struck: Structured output tracking with kernels," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38 no. 10, pp. 2096-2109, October, 2016. Article(CrossRef Link)

[7] K. Zhang, L. Zhang and M. H. Yang, "Fast Compressive Tracking," *IEEE Transactions on Pattern* Analysis and Machine Intelligence, vol. 36, no. 10, pp. 2002-2015, October, 2014. Article(CrossRef Link)

[8] F. Yang, H. Lu and M. H. Yang, "Robust superpixel tracking," *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1639-1651, April, 2014. Article(CrossRef Link)

[9] Q. Wang, F. Chen and W. Xu, "Tracking by third-order tensor representation," *IEEE Transactions on Systems Man and Cybernetics*, vol. 41, no. 2, pp. 385-396, April, 2011. Article(CrossRef Link)

[10] D. Wang, H. Lu and M. H. Yang, "Online object tracking with sparse prototypes," *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 314-325, January, 2013. Article(CrossRef Link)

[11] D. Wang, H. Lu and M. H. Yang, "Robust Visual Tracking via Least Soft-thresold Squares," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 9, pp. 1709-1721, September, 2016. Article(CrossRef Link)

[12] B. Liu, J. Huang, L. Yang and C. Kulikowsk, "Robust tracking using local sparse appearance model and k-selection," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1313-1320. June 20-25, 2011. Article(CrossRef Link)

[13] C. Bao, Y. Wu, H. Ling and H. Ji, "Real time robust l1 tracker using accelerated proximal gradient approach," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1830-1837, June 16-21, 2012. Article(CrossRef Link)

[14] J. Gao, H. Ling, W. Hu and J. Xing, "Transfer learning based visual tracking with gaussian processes regression," in *Proc. of European Conference on Computer Vision*, pp. 188-203, September 6-12, 2014. Article(CrossRef Link)

[15] J. F. Henriques, R. Caseiro, P. Martins and J. Batista. "High-speed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583-596, March, 2015. Article(CrossRef Link)

[16] W. Zhong, H. Lu and M. H. Yang, "Robust object tracking via sparse collaborative appearance model," *IEEE Transactions on Image Processing*, vol. 23, no. 5, pp. 2356-2368, May, 2014. Article(CrossRef Link)

[17] J. Xue and D. M. Titterington, "Comment on "On Discriminative vs. Generative Classifiers: A Comparison of Logistic Regression and Naive Bayes"," *Neural Processing Letters*, vol. 28, no. 3, pp. 169-187, October, 2008. Article(CrossRef Link)

[18] J. A. Lasserre, C. M. Bishop and T. P. Minka, "Principled hybrids of generative and discriminative models," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 87-94, June 17-22, 2006. Article(CrossRef Link)

[19] S. He, Q. Yang, R. W. Lau, J. Wang and M. H. Yang, "Visual tracking via locality sensitive histograms," in *Proc. of International Conference on Computer Vision*, pp. 2427-2434, June 23-28, 2013. Article(CrossRef Link)

[20] F. Yang, H. Lu, W. Zhang and G. Yang, "Visual tracking via bag of features," *IET Image Processing*, vol. 6, no. 2, pp. 115-128, March, 2012. Article(CrossRef Link)

[21] A. W. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara and A. Dehghan, "Visual Tracking: An experimental survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1442-1468, July, 2014. Article(CrossRef Link)

[22] Y. Wu, J. Lim and M. H. Yang, "Online Object Tracking: A Benchmark," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2411-2418, June 23-28, 2013. Article(CrossRef Link)

[23] J. Pan, J. Lim, Z. Su and M. H. Yang, "L0-Regularized Object Representation for Visual Tracking," in *Proc. of British Machine Vision Conference*, September 1-5, 2014. Article(CrossRef Link)

[24] B. Liu, L. Yang, J. Huang, P. Meer and L. Gong, "Robust and fast collaborative tracking with two stage sparse optimization," in *Proc. of European Conference on Computer Vision*, pp. 624-637, September 5-11, 2010. Article(CrossRef Link)

[25] T. Zhang, B. Ghanem, S. Liu and N. Ahuja, "Robust visual tracking via multi-task sparse learning," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2042-2049, June 16-21, 2012. Article(CrossRef Link)

[26] X. Jia, H. Lu and M. H. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1822-1829, June 16-21, 2012. Article(CrossRef Link)

[27] B. Ma, J. Shen, Y. Liu and H. Hu, "Visual tracking using strong classifier and structural descriptors," *IEEE Transactions on Multimedia*, vol. 17, no. 10, pp. 1818-1828, October, 2015. Article(CrossRef Link)

[28] X. Zeng, L. Xu, L. Ma, R. Zhao and Y. Cen, "Visual tracking using global sparse coding and local convolutional features," *Digital Signal Processing*, vol. 72, pp. 115-125, January, 2018. Article(CrossRef Link)

[29] W. Feng, Y. Cen, X. Zeng, Z. Li, M. Zeng and V. Voronin, "Object tracking based on adaptive updating of a spatial-temporal context model," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 11, pp. 5459-5473, November, 2017. Article(CrossRef Link)

[30] N. Wang and D. Y. Yeung, "Learning a deep compact image representation for visual tracking," in *Proc. of Advances in Neural Information Processing Systems*, pp. 809-817, December 5-10, 2013. Article(CrossRef Link)

[31] Y. Qi, L. Qin, J. Zhang, S. Zhang, Q. Huang and M. H. Yang, "Structure-aware local sparse coding for visual tracking," *IEEE Transactions on Image Processing*, vol. pp, no. 99, pp. 1-1, January, 2018. Article(CrossRef Link)

[32] P. P. Dash and D. Patra, "Efficient visual tracking using multi-feature regularized robust sparse coding and quantum particle filter based localization," *Journal of Ambient Intelligence and Humanized Computing*, vol. 2018, no. 5, pp. 1-14, January, 2018. Article(CrossRef Link)

[33] Y. Zhou, J. Han, X. Yuan, Z. Wei and R. Hong, "Inverse Sparse Group Lasso Model for Robust Object Tracking," *IEEE Transactions on Multimedia*, vol. 19, no. 8, pp. 1798-1810, August, 2017. Article(CrossRef Link)

[34] N. Ali, K. B. Bajwa, R. Sablatnig and Z. Mehmood, "Image retrieval by addition of spatial information based on histograms of triangular regions," *Computers & Electrical Engineering*, vol. 54, no. c, pp. 539-550, August, 2016. Article(CrossRef Link)

[35] N. Ali, K. B. Bajwa, R. Sablatnig and S. A. Chatzichristofis, "A novel image retrieval based on visual words integration of SIFT and SURF," *Plos One*, vol. 11, no. 6, pp. e0157428, June, 2016. Article(CrossRef Link)

[36] N. Ali, D. A. Mazhar, Z. lqbaland and R. Ashraf, "Content-Based Image Retrieval Based on Late Fusion of Binary and Local Descriptors," *International Journal of Computer Science & Information Security*, vol. 14, no. 11, pp. 821-837, March, 2017. Article(CrossRef Link)

[37] L. Ye, L. Wang, Y. Sun, L. Zhao and Y. Wei, "Parallel multi-stage features fusion of deep convolutional neural networks for aerial scene classification," *Remote Sensing Letters*, vol. 9, no. 3, pp. 294-303, December, 2017. Article(CrossRef Link)

[38] Q. Liu, R. Hang, H. Song and Z. Li, "Learning Multiscale Deep Features for High-Resolution Satellite Image Scene Classification," *IEEE Transactions on Geoscience & Remote Sensing*, vol. 56, no. 1, pp. 117-126, January, 2018. Article(CrossRef Link)

[39] T. Zhou, K. Xie, J. Zhang, J. Yang and X. He, "Robust object tracking based on weighted subspace reconstruction error with forward: backward tracking criterion," *Journal of Electronic Image*, vol. 24, no. 3, pp. 033005, 2015. Article(CrossRef Link)

[40] H. Li, Y. Li and F. Porikli, "Robust online visual tracking with a single convolutional neural network," *in Proc. of the 12-th Asian Conference on Computer Vision*, pp. 194-209, November 1-5, 2014. Article(CrossRef Link)

[41] L. Wang, W. Ouyang, X. Wang and H. Lu, "Visual tracking with fully convolutional networks," *in Proc. of IEEE International Conference on Computer Vision*, pp. 3119-3127, December 7-13, 2015. Article(CrossRef Link)

[42] R. Tao, E. Gavves and A. W. M. Smeulders, "Siamese instance search for tracking," *in Proc. of Computer Vision and Pattern Recognition*, June 27-30, 2016. Article(CrossRef Link)

[43] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," *in Proc. of Computer Vision and Pattern Recognition*, June 27-30, 2016. Article(CrossRef Link)

Xianyou Zeng received the M.S. degree from South China Normal University. He is currently a Ph.D. student in the College of Computer and Information Technology at Beijing Jiaotong University. His research interests include image processing and visual tracking.

Long Xu received his Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences. After that, he was a postdoc with City University of Hong Kong, Chinese University of Hong Kong, and Nanyang Technological University. Currently, he is with the Key Laboratory of Solar Activity, National Astronomical Observations, Chinese Academy of Sciences. His research interests include image/video processing, wavelet, machine learning and computer vision.

Shaohai Hu received the B.S. degree and M.S. degree in electronic engineering from the Beihang University, Beijing, China, in 1985 and 1988 respectively. He received the Ph.D. degree in signal and information processing from Beijing Jiaotong University, Beijing, China, in 1991. He is a professor of Beijing Jiaotong University, Beijing, China. His research interests including image processing, information infusion and neural network.

Ruizhen Zhao received his Ph.D. degree from Xidian University. After that he was a postdoctrol fellow in Institute of Automation, Chinese Academy of Sciences. He is currently a professor and supervisor of doctor student of Beijing Jiaotong University. His research interests include wavelet transform and its applications, algorithms of image and signal processing, compressive sensing and sparse representation.

Wanli Feng received the BS degree from Northwest Normal University in 2015. He has been pursuing his master's degree in signal and information processing at the Beijing Jiaotong University since 2015. His current research interests include visual tracking and machine learning.