



# Spatial and semantic convolutional features for robust visual object tracking

Jianming Zhang<sup>1,2</sup> · Xiaokang Jin<sup>1,2</sup> · Juan Sun<sup>1,2</sup> · Jin Wang<sup>1,2</sup> ·  
Arun Kumar Sangaiah<sup>3</sup> 

Received: 1 June 2018 / Revised: 30 July 2018 / Accepted: 15 August 2018 /  
Published online: 23 August 2018  
© Springer Science+Business Media, LLC, part of Springer Nature 2018

## Abstract

Robust and accurate visual tracking is a challenging problem in computer vision. In this paper, we exploit spatial and semantic convolutional features extracted from convolutional neural networks in continuous object tracking. The spatial features retain higher resolution for precise localization and semantic features capture more semantic information and less fine-grained spatial details. Therefore, we localize the target by fusing these different features, which improves the tracking accuracy. Besides, we construct the multi-scale pyramid correlation filter of the target and extract its spatial features. This filter determines the scale level effectively and tackles target scale estimation. Finally, we further present a novel model updating strategy, and exploit peak sidelobe ratio (PSR) and skewness to measure the comprehensive fluctuation of response map for efficient tracking performance. Each contribution above is validated on 50 image sequences of tracking benchmark OTB-2013. The experimental comparison shows that our algorithm performs favorably against 12 state-of-the-art trackers.

**Keywords** Object tracking · Convolutional neural networks · Correlation filter · Scale adaptive · Model updating strategy

## 1 Introduction

Visual object tracking is one of the fundamental research topics in computer vision for its numerous application, such as video surveillance, human-computer interaction, driverless

---

This work was supported in part by the National Natural Science Foundation of China under Grant 61402053, Grant 61772454, Grant 61811530332 in part by the Scientific Research Fund of Hunan Provincial Education Department under Grant 16A008, in part by the Scientific Research Fund of Hunan Provincial Transportation Department under Grant 201446, in part by the Industry-University Cooperation and Collaborative Education Project of Department of Higher Education of Ministry of Education under Grant 201702137008, in part by the Undergraduate Inquiry Learning and Innovative Experimental Fund of CSUST under Grant 2018-6-119, and in part by the Postgraduate Course Construction Fund of CSUST under Grant KC201611.

---

✉ Arun Kumar Sangaiah  
arunkumarsangaiah@gmail.com

Back Affiliation

vehicle, military field. In addition, object tracking is widely used in the multimodal data, for example, the accuracy of speech recognition would soar up significantly with tracking the lip motion simultaneously. Although many effective algorithms [9, 15, 40], and benchmark evaluations [18, 36] have been proposed in the past decade, visual tracking is still a challenging problem due to illumination variation, scale variation, partial and full occlusion, fast motion, background clutter, in-plane and out-of-plane rotation.

Nowadays, many state-of-the-art approaches learn the discriminative appearance model of the object to robust track. In general, according to the difference ways of target modeling, object tracking algorithms can be categorized as either generative or discriminative. The generative algorithms are mainly used to model the foreground, search for the candidate region by minimal reconstruction error, find the optimal matched position in the current frame, and update the target model by online learning mechanism. Compared with the generative trackers, the discriminative trackers can transform the tracking problem into a binary classification problem. The target is distinguished from its background with the trained discriminative classifier by collecting a set of positive and negative samples around the estimated target location of each frame. Such approaches [1, 5, 41] mainly rely on rich feature representations, good performance classifier, strict positive and negative samples and the robust online updating mechanism. This paper aims at robust feature representations and model updating strategy as well as solving the scale problem.

Discriminative tracking methods based on the correlation filter have become a hot topic and achieved excellent performance recently. Discriminative correlation filters (DCF<sub>s</sub>) [4, 12] use a fast Fourier transform for efficient calculations and regress the circularly shifted versions of input features to soft labels so that DCF<sub>s</sub> have excellent real-time performance. DCF<sub>s</sub> aim to train a correlation filter with discriminative ability, then find the maximum response on the confidence map as the tracking result of the object. However, most DCF<sub>s</sub> only use gray feature, HOG feature and color names feature or the fusion of multiple features. These conventional handcraft features can't meet the requirements of high performance tracking well because target tracking often requires more robust features.

With the rapid development of convolutional neural networks (CNNs), it has made outstanding achievements in computer vision such as image detection and segmentation [28, 35, 43], pattern classification [30, 44], text detection [38, 39], and medical image processing [19, 24]. The robust feature representation capability of the network makes CNNs based [7, 14, 25] a hot topic in the object tracking, and they have achieved excellent tracking performance as well. However, the CNNs model needs to be updated online because of the influences of various interference factors. Therefore, it takes much time to update the end-to-end CNNs model and the real-time performance can't be guaranteed even under the acceleration of GPU.

Therefore, combining the advantages of CNNs and DCF<sub>s</sub>, we propose a robust tracking algorithm based on spatial and semantic (double 's', DS) convolutional features. The spatial convolutional feature which we define includes the features of all the convolutional layers before the second pooling layer in the VGG-Net model shown in Fig. 3. Correspondingly, the semantic convolutional feature represents that after the second pooling layer.

The contributions of this work are as follows:

1. We construct translation correlation filters to localize the target with spatial and semantic convolutional features. The dimensions of spatial features are reduced using 2DPCA to retain most information.
2. We construct a scale correlation filter to estimate the target scale with spatial convolutional features.

3. We propose model updating strategy. The skewness is firstly introduced into object tracking. With a combination of PSR, it is more reasonable to comprehensively measure the fluctuation of response map.

Evaluation on the widely used tracking benchmark demonstrates that each contribution of proposed algorithm improves the accuracy and success rate effectively.

## 2 Related work

In this section, we mainly introduce the tracking algorithms from four aspects: tracking by CNNs, tracking by DCFs, tracking by multiple scales and online model drift prevention.

**Tracking by CNNs** Most of existing tracking algorithms use a pre-trained model to extract and classify the features of the target. Since the representation power of CNNs features shows excellent representation performance in object detection and recognition, there are some attempts to employ CNNs for visual tracking. In 2013, deep learning tracking (DLT) tracker proposed by Wang et al. [32] applies deep models to single-target tracking problems for the first time, then some state-of-the-art algorithms, like fully convolutional networks tracking (FCNT) [33], hierarchical convolutional features tracking (HCFT) [20], generic object tracking using regression networks (GOTURN) [11], fully-convolution Siamese network (SiamFC) [2], multi-domain networks tracking (MDNet) [22], convolutional residual learning tracking (CREST) [27], correlation filter network (CFNet) [31] etc. was proposed. The tracking methods based on CNNs show great performance. Different from the end-to-end framework in CNN, in this work, we exploit the correlation filters to localize the target combined with the different features to speed up the calculation and improve the accuracy.

**Tracking by DCFs** The tracking algorithms based on correlation filters have attracted widespread concern in recent years because it can transform convolution operation into element multiplication in Fourier domain, which can improve computational efficiency and achieve a good real-time performance. In 2010, Bolme et al. [4] first proposed to learn a filter by using minimum output sum of squared error (MOSSE), which was an early application of correlation filter in the field of object tracking. Then a series of improved algorithms in various forms were proposed, including circulant structure of tracking with kernels (CSK) [12], kernelized correlation filters (KCF) [13], spatio-temporal context learning (STC) [42], long-term correlation tracking (LCT) [21], joint discrimination and reliability learning tracking (DRT) [29] and sum of template and pixel-wise learners (Staple) [3]. However, these algorithms only use a single correlation filter or handcraft features, which severely limits the tracking performance. In this work, we exploit the spatial and semantic convolutional features in CNN to construct multiple correlation filters to obtain the target location and scale level respectively.

**Tracking by multiple scales** Robust scale estimation is a challenging problem in visual object tracking. Since the size of the target changes in the process of target moving, the bounding box is difficult to adapt to the target size. Scale adaptive multiple features (SAMF) tracker proposed by Li et al [17] obtains the optimal target scale by integrating CN and HOG feature in seven different levels of scale. Discriminative scale space tracker (DSST) [6] tracker uses a discriminative correlation filter to model the target appearance, exploits two independent correlation

filters to effectively estimate the scale and localize the target respectively. Besides, spatially regularized discriminative correlation filters (SRDCF) [8], spatio-temporal context learning (STC) [42] and multi-scale compressive tracker (MSCT) [37] have realized the scale adaptation as well. In this work, we utilize the spatial convolutional features of CNN to construct a scale correlation filter, then obtain the final response of different scale levels to estimate the scale.

**Online model drift prevention** It may lead to the position drift by using the pre-trained model to localize the target, which makes the prediction inaccurate and further causes the loss of the target. Many scholars have done a lot of work to prevent model drift. MIL tracker [1] exploits multiple instance learning to form positive and negative samples into packets respectively, and trains the classifiers in packets, which enable the tracker to effectively handle the drift caused by deformation, complex background and occlusion. TLD tracker [15] combines tracking with detection to solve the problems such as deformation, occlusion, illumination variation effectively. Multiple experts using entropy minimization tracker (MEEM) [41] tracker utilizes the entropy canonical term criterion to make a reasonable decision and correct the error update of the model to improve the tracking accuracy. LMCF tracker [34] explores a multimodal target detection technique to prevent the model drift problem and establishes a model updating strategy to avoid model corruption. In this work, the PSR and skewness are introduced to measure the fluctuation of the response map, which not only optimizes the model updating strategy, but also enhances the precision under the interference of occlusion, illumination variation and complex background.

### 3 Our algorithmic overview

This section describes the framework of the proposed algorithm from following four aspects: 1) Extract DS convolutional features from the region of interest using CNN. 2) Construct the correlation filters and obtain the final response map to localize the target. 3) Estimate the multiple scale by constructing a scale filter based on the localized position. 4) Update correlation filters online with proposed model updating strategy. Fig. 1 shows the overview of our proposed algorithm.

Figure 1 shows the specific steps of the algorithm. It is described in detail as follows: Step (1): the size of candidate region generated in the current frame is set to twice the size of the target in the previous frame; Step (2): candidate regions are set as the input of the pre-trained VGG-Net-19 networks [26] to extract features. The low-level feature maps have more spatial detailed information while the high-level feature maps have abundant semantic information, so we define features drawn out from the feature maps of Conv1–2, Conv2–2, Conv3–4, Conv4–4, Conv5–4 as the features of the location filter in the forward propagation of the network; Step (3): the sizes of the feature maps are normalized to unify the sizes of drawn features. Because there are a few pooling operations of the spatial convolutional features and the sizes of the feature maps are large, we use 2DPCA to reduce the dimension of the feature maps, which remains most of the useful features. The sizes of semantic convolutional features are small, so we use resampling to enlarge the sizes of the features. We construct correlation filters for the normalized feature maps of each layer to obtain the maximum response value; Step (4), (5), (6), (7): the maximum response value of each layer is weighted, and the response map of each layer is weighted by this layer and the higher layer, so that the final response map is obtained; Step (8): the target is localized by the maximum response value of the final response map; Step

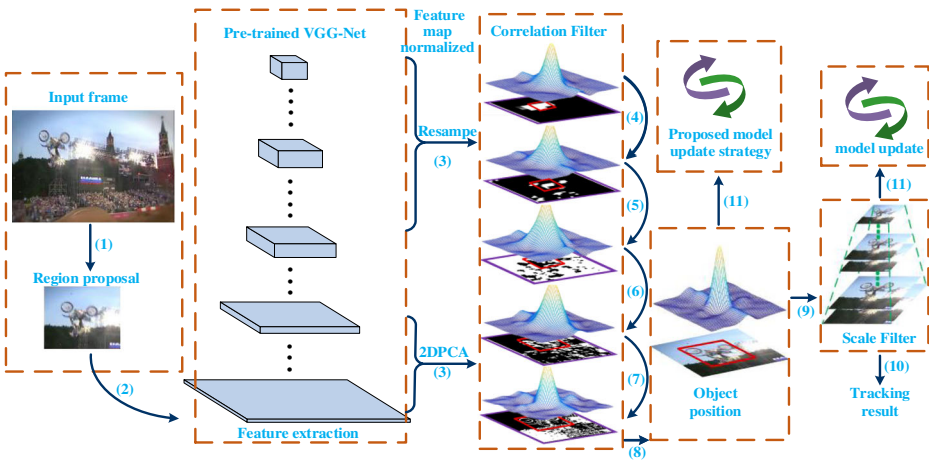


Fig. 1 Framework of the proposed algorithm

(9): the scale level of the target is determined by constructing scale pyramid model after the target is localized; Step (10): the location and size of the bounding box are finally determined; Step (11): the location and scale filters are updated with different model updating strategies, the translation filters use proposed model updating strategy while the scale filter not.

## 4 Proposed tracking algorithm

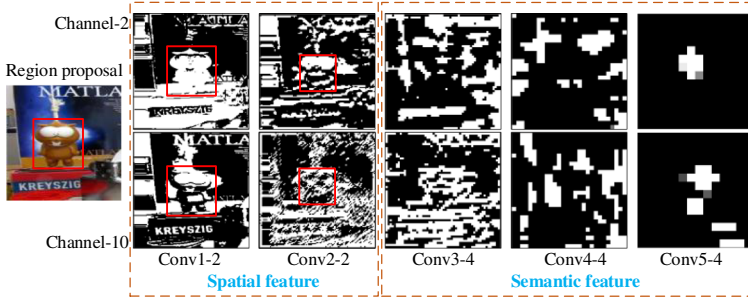
In this section, we will introduce the four aspects in details according to the algorithmic overview.

### 4.1 DS convolutional features extraction

At present, there are many popular deep network models, such as AlexNet [16], VGG-Net [26], ResNet [10] etc., which have achieved excellent performance in the field of image classification and recognition. Our algorithm mainly uses the VGG-Net network trained on the ImageNet dataset include 1.3 million images, which has an appropriate number of network layers. Compared with some shallower CNN networks, it has greater competence for feature representation and it is faster when compared with other deeper CNN networks. Considering the two main factors comprehensively, we select VGG-Net to extract the features of the region of interest.

With the forward propagation of CNN, the different types of semantic discriminative information in the image will be strengthened while spatial detailed information of the features will be weakened gradually. In terms of the task of target tracking, we pay more attention to how to exploit these features to locate the target accurately. We ignore the fully-connected layer because of its low spatial resolution, which is often a  $1 \times 1 \times n$  tensor. The resolution of the feature decreases rapidly with the increase of pooling operations in CNN, for example, in the VGG-Net, the size of the input image is  $224 \times 224$ , and the size of the output feature of pool5 layer is  $7 \times 7$ , which is  $1/32$  of the image size.

Figure 2 shows the visualization of feature maps with different properties. It can be found that the low-level feature maps have more obvious spatial detailed information, retain most of



**Fig. 2** Visualization of The DS feature maps. The first row shows the feature maps of the 2nd channel. The second row shows the feature maps of the 10th channel. The columns from left to right represent Conv1–2, Conv2–2, Conv3–4, Conv4–4, Conv5–4 in VGG-Net-19 respectively

the edge and texture information of the target, and facilitate the precise localization of the target. However, in the high-level feature map, a pixel corresponds to a large receptive field, and its rich semantic information can significantly improve the anti-interference ability of the target’s own postures and environmental changes during its movement. Therefore, we synthesize spatial and semantic convolutional features (double ‘s’, DS), which is helpful to improve tracking accuracy and robustness. Meanwhile, the dimensions of spatial features are reduced using 2DPCA to retain most spatial information in feature normalization.

### 4.2 Prediction location by correlation filter

Recently, discriminative correlation filters have been widely used in the field of target tracking [3, 6, 8, 13, 17]. The DCFs mainly learn a classifier and estimate the target position by searching for the maximum response location of the response map. The output of each convolutional layer is a set of multi-channel feature maps denoted by  $f_l \in \mathbb{R}^{M \times N \times D}$ , where  $M$ ,  $N$ , and  $D$  indicate the width, height, and the number of channels respectively. The  $f_l^d$  indicates the feature map of the  $d$ -th channel extracted from the  $l$ -th layer, and then a correlation filter  $h_l^d$  is constructed for each channel of the feature map. The optimal correlation filter  $h^*$  of the  $l$ -th layer is obtained by minimizing the cost function:

$$h_l^* = \arg \min \left\| \sum_{d=1}^D h_l^d \cdot f_l^d - y \right\|^2 + \lambda \|h_l\|^2 \tag{1}$$

Here,  $\lambda$  is a regularization parameter ( $\lambda \geq 0$ ), the sizes of  $h_l^d$ ,  $f_l^d$  and  $y$  are  $M \times N$ ,  $y$  denotes the corresponding Gaussian function label which is subject to a 2D Gaussian distribution,  $y$  is computed as:

$$y(m, n) = e^{-\frac{(m-M/2)+(n-N/2)^2}{2\sigma^2}} \tag{2}$$

Let  $\omega = \left\| \sum_{d=1}^D h_l^d \cdot f_l^d - y \right\|^2 + \lambda \|h_l\|^2$ , we can get the  $\Omega$  which is the representation of  $\varepsilon$  in the frequency domain:

$$\Omega = \left\| \sum_{d=1}^D H_l^d \odot \overline{F_l^d} - Y \right\|^2 + \lambda \sum_{d=1}^D \|H_l\|^2 \tag{3}$$

Here, the  $F_l^d$ ,  $Y$ , and  $H_l$  represent the discrete Fourier transform (DFT) of the  $f_l^d$ ,  $y$ , and  $h_l$  respectively. The bar means complex conjugation. The operator  $\odot$  is the Hadamard (element-wise) product in the Fourier domain. By  $\frac{\partial \tilde{\varepsilon}}{\partial H_l^d} = 0$ , the optimal filter on each feature channel  $d(d \in \{1, 2, \dots, D\})$  of the  $l$ -th layer can be computed as:

$$H_l^d = \frac{Y \odot \overline{F_l^d}}{\sum_{i=1}^D F_l^i \odot \overline{F_l^i} + \lambda} \tag{4}$$

In the prediction of the next frame, the extracted DS convolutional feature maps is denoted as  $z$ , and  $z \in \mathcal{R}^{M \times N \times D}$ . The discrete Fourier transform of  $z$  is the  $Z$ , whose complex conjugation is the  $\overline{Z}$ , the correlation response map  $E_l$  of each layer is obtained by (5) as:

$$E_l = \mathcal{F}^{-1} \left( \sum_{d=1}^D H_l^d \odot \overline{Z_l^d} \right) \tag{5}$$

Here,  $\mathcal{F}^{-1}$  denotes the Inverse DFT. We get a more reasonable position of the current layer by weighting the maximum response value of the high-level and the current layer. The optimal location of the  $(l-1)$ -th layer is computed as follows:

$$E_{l-1}(m, n) = \alpha_{l-1} E_{l-1}(m, n) + \alpha_l E_l(m, n) \tag{6}$$

Here,  $\alpha_l$  denotes the position weight of the  $l$ -th layer. We can get the final response map  $E$  by (6). The center position of the current tracking target  $p_t = (x_t, y_t)$  can be determined by searching for the maximum response value:

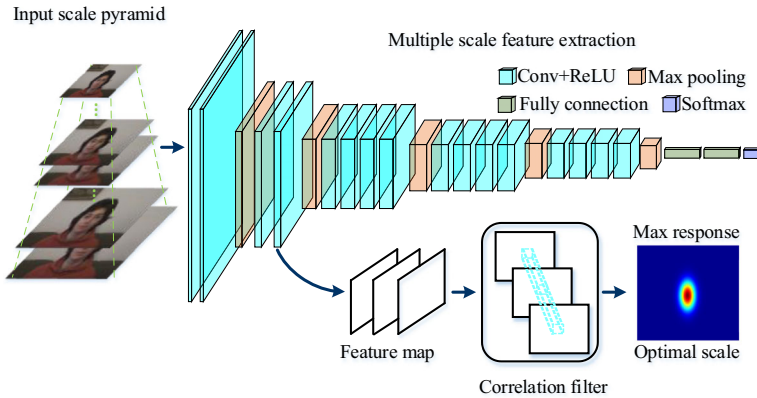
$$(x_t, y_t) = \arg_{m,n} \max E(m, n) \tag{7}$$

### 4.3 Multiple scale estimation

After the position of the target is determined, the scale pyramid model is constructed by multi-scale sampling of the target area [6] according to the target size  $s_{t-1} = (w_{t-1}, h_{t-1})$  of the previous frame. Unlike HOG or LBP feature extracted in [45, 46], in this work, the spatial features in VGG-Net is extracted to construct the scale filter to estimate the target scale. We only select  $D'$  channels of the feature maps of the Conv2–2 layer.

Figure 3 shows the implementation flow of multi-scale estimation. Let  $a$  denote the scale factor and  $S$  be the size the number of samples. As for scale level  $n = \{ \lfloor -\frac{S-1}{2} \rfloor, \dots, \lfloor \frac{S-1}{2} \rfloor \}$ , we extract multi-scale image patch of size  $a^n \cdot w_{t-1} \times a^n \cdot h_{t-1}$  around the target’s estimated central location  $p_t$  in the current frame as samples. The sampling process is shown in Fig. 3. The spatial convolutional features of each image patch are extracted, that is, the Conv2–2 layer shown in Fig. 3 to ensure that the feature maps are small enough. The extracted scale feature is denoted as  $f_s^d$ , which has  $D'$  channels of features. Given a sample label  $y$  in a 1-dimensional Gaussian function, a scale filter is obtained:

$$H_s^d = \frac{Y \odot \overline{F_s^d}}{\sum_{d=1}^{D'} F_s^d \odot \overline{F_s^d} + \lambda_s} \tag{8}$$



**Fig. 3** The above part are the scale pyramid model of target and the network structure of VGG-Net-19 which includes convolutional layers, pooling layers, fully-connection layers and softmax. The below part is a scale correlation filter constructed by spatial convolutional features. The optimal scale of the target is defined as the maximum response

Here,  $\lambda_s$  is a regularization parameter in a scale filter, and the  $F_s^d$ ,  $Y$  are DFT transformations form of the  $f_s^d$  and  $y$  respectively. The  $\overline{F}_s^d$  is the complex conjugation of the  $F_s^d$ . When tracking the target of the  $t$ -th frame in the image sequence, it is known that the target filter in the  $(t-1)$ -th frame is  $H_s(t-1)$ , then, through the multi-scale convolutional features of the target extracted by multi-scale sampling, we can get the scale correlation filter response  $E_s$  of the  $t$ -th frame:

$$E_s = \mathcal{F}^{-1} \left\{ \sum_{d=1}^{D'} H_s^d(t-1) \odot \overline{F}_s^d(t) \right\} \tag{9}$$

At this point,  $s_t$  is the maximum response with the optimal scale in response  $E_s$ , which can be calculated as:

$$s_t = \operatorname{argmax}_n E_s(n) \tag{10}$$

### 4.4 Proposed model updating strategy

Object tracking is a state estimation problem of dynamic sample, which often involves model update. In the tracking process, the target is not only influenced by its changes such as deformation and rotation, but also may be interfered by other complex factors such as illumination variations and background occlusions. These may lead to large differences of the target in different frames. Therefore, the tracking model needs to be updated in the tracking process. Some general algorithms exploit a fixed learning rate to update the correlation filter parameters for model update. However, when the target is severely affected by occlusions or the illumination variations, the tracking algorithm still regards the target whose appearance in the bounding box has changed as the real target and always update the classifier with a constant learning rate, which will lead to a decrease in the performance of the classifier and target drift or even loss. Therefore, in this work, the confidence of target location is measured by the two indexes of PSR and skewness to update the correlation filter model more reasonably.



PSR has been widely used in signal processing and can be expressed in response map as:

$$PSR_t = \frac{\max(\mathbf{E}) - \mu_t}{\sigma_t} \tag{11}$$

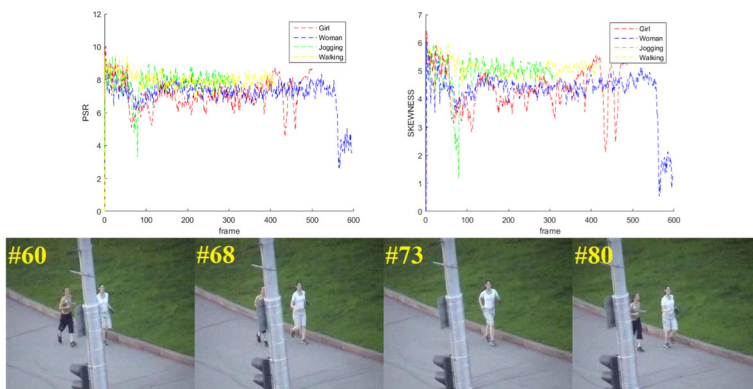
Here,  $\mathbf{E}$  is final response map after the calculation by (6), represents the maximum response value,  $\mu_t$  and  $\sigma_t$  denote the mean and standard deviation of the  $t$ -th response map  $\mathbf{E}$ . The larger the value of  $PSR_t$  is, the higher the tracking quality of  $t$ -th frame is.

Skewness is a statistic that studies data distribution symmetry. By measuring the skewness, the asymmetry and direction of data distribution can be determined. The expression applied to the response map is:

$$SK_t = \frac{\frac{1}{MN} \sum_{n=1}^N \sum_{m=1}^M (E(m, n) - \mu_t)^3}{\left[ \frac{1}{MN} \sum_{n=1}^N \sum_{m=1}^M (E(m, n) - \mu_t)^2 \right]^{\frac{3}{2}}} \tag{12}$$

Here,  $M$  and  $N$  denote the width and height of the response map respectively. The greater the value of  $SK_t$  is, the higher the tracking quality of the  $t$ -th frame is.

The distribution of PSR and skewness of the four selected image sequences is shown in Fig. 4. The value of PSR and skewness shows a similar tendency, that is, the higher the value is, the higher the confidence of the target position is. The fluctuation trend of is more sensitive and intense than PSR, so the skewness has more potential analysis effects. Fig. 4 shows the specific performance of PSR and skewness fluctuations in the *Jogging* image sequence: the target begins to approach the occlusions in the 60-th frame and is completely occluded at the 70-th frame. At the same time, the values of the PSR and the skewness drop to the lowest point. As the size of occluded part decreases, the values of the PSR and the skewness gradually increases. When their values are at the lowest point, the tracking result is not reliable. If the model is updated at this time, the model update is not reasonable. Therefore, we compare the PSR and skewness of each frame with the threshold value to judge whether the target is interfered or not, thereby formulating a new model updating strategy.



**Fig. 4** Distribution and analysis of PSR and skewness values. The above part is distribution of PSR and skewness, which fluctuate greatly in the 60-th to 80-th frame of the *jogging* image sequence, PSR on the left and skewness on the right. The below part is analysis of PSR and skewness in *Jogging* image sequence

When the model is updated normally,  $A_{t-1}^d$  and  $B_{t-1}^d$  represent the correlation filter parameters respectively for filters  $H_t^d$  at  $(t-1)$ -th frame. At  $t$ -th frame, the filter updating strategy is:

$$A_t^d = (1-\eta)A_{t-1}^d + \eta Y \odot \overline{F}_t^d(t) \quad (13a)$$

$$B_t^d = (1-\eta)B_{t-1}^d + \eta \sum_{i=1}^D F_t^i(t) \odot \overline{F}_t^i(t) \quad (13b)$$

$$H_t^d(t) = \frac{A_t^d}{B_t^d + \lambda} \quad (14)$$

Scale filter is updated according to (13), (14). When the PSR and skewness meet the requirements, the new model updating strategy is used. The updating strategy is described in Table 1. When the two conditions of  $PSR_t < \theta_1$  and  $SK_t < \theta_2$  are both satisfied, it shows that the model of the target appearance is interfered by the external factors and the response map is fluctuated intensely. Therefore, it is difficult to determine the maximum value, which makes it difficult to localize the target. In this condition, we will not update the model, the parameters of the model remain the same as them of the previous frame. Learning rate is a parameter to measure the freshness of classifiers. A lower learning rate can avoid learning more background information. Therefore, whether  $PSR_t < \theta_1$  or  $SK_t < \theta_2$  is satisfied, it shows that the response map has intense fluctuations. We need to reduce the learning of interference factors so that the model retains the parameters describing the target to the most extent. In other cases, the model is updated according to (13) and (14).

Finally, Table 2 shows the main steps of our DS target tracking algorithm.

## 5 Experimental results

In this section, we first describe the detailed implementation of the experiment, and then analyze the effectiveness of each contribution, finally compare the proposed algorithm with other state-of-art trackers on the OTB-2013 dataset. The effectiveness of the proposed tracking algorithm is verified comprehensively by quantitative evaluation, attribute-based evaluation and qualitative evaluation.

OTB-2013 was proposed by Wu et al. in 2013. OTB-2013 contains 50 video sequences involving 11 interference attributes, such as motion blur (MB), deformation (DEF), fast motion (FM), out-of plane rotation (OPR), scale variation (SV), occlusion (OCC), illumination variation (IV), background clutter (BC), out-of-view (OV), in-plane rotation (IPR) and low

**Table 1** The Proposed model updating strategy

Satisfied conditions	Updating strategies
$PSR_t < \theta_1$ and $SK_t < \theta_2$	not necessary to update the model at $t$ -th frame
$PSR_t < \theta_1$ or $SK_t < \theta_2$	reduce the learning rate and update model: $\eta_{new} = 0.8 \times \eta$
else	update model normally

**Table 2** The main steps of our ds tracking algorithm

Algorithm: proposed tracking approach: iteration at $t$ -th step
<b>Input:</b> Image sequence: $I_1, I_2, \dots, I_n$ . Initial target position: $p_0 = (x_0, y_0)$ , and initial target scale: $s_0 = (w_0, h_0)$
<b>Output:</b> The estimated position of target: $p_t = (x_t, y_t)$ and estimated scale: $s_t = (w_t, h_t)$
<b>For</b> $t = 2: n$
<b>Locate the position of Target:</b>
1: Crop out the region proposal in frame $t$ centered at $p_{t-1}$ , and extract the DS convolutional features;
2: Build the correlation filters using (4) and (5) for five convolutional layers;
3: Weight five correlation response maps using (6), and obtain the compositive response map;
4: Localize the center of the target $p_t$ using (7);
<b>Estimate the Scale of Target:</b>
5: Obtain the multi-scale samples in frame $t$ based on $p_t$ and $s_{t-1}$ ;
6: Build scale filters by extracting convolutional features from the above samples;
7: Compute the scale filters response using (8) and (9);
8: Estimate the optimal scale $s_t$ of the target using (10);
<b>Model Update:</b>
9: Calculate PSR and skewness using (11) and (12), update the position filters according table1;
10: Update the scale filters using (13), (14);
<b>END</b>

resolution (LR). We use the One-Pass Evaluation (OPE) to evaluate the proposed algorithm and adopt two performance indicators: success plot and precision plot. Success plot illustrates the percentage of frames whose overlaps ratio are larger than the given threshold. Overlap ratio is defined as  $s = \text{area}(R_T \cap R_G) / \text{area}(R_T \cup R_G)$ , where  $R_T$  refers to the tracked bounding box and  $R_G$  refers to the ground truth. The area under curve (AUC) of each success plot is used to rank the evaluated trackers. Precision plot is defined as the average Euclidean distance between the center locations of the tracked bounding box and the ground truth. To rank the trackers, the commonly used distance threshold is 20 pixels.

## 5.1 Implementation details

We exploit the VGG-Net-19 for the feature extraction of target. We crop out a bounding box that is 2 times the size of the target on the image as the candidate region and input it into the network. In the process of forward propagation, we extract the features of Conv1–2, Conv2–2, Conv3–4, Conv4–4 and Conv5–4 and set the corresponding weight for each layer respectively to balance spatial and semantic information. Our implementation runs at 2.26 frames per second on a computer with an Intel I7-6700 K 4.0 GHz CPU, 16GB RAM, and a GeForce GTX980Ti GPU card which is only used to compute the CNN features. The version of Cuda is 8.0. We implement our tracker in MATLAB using Matconvnet toolbox.

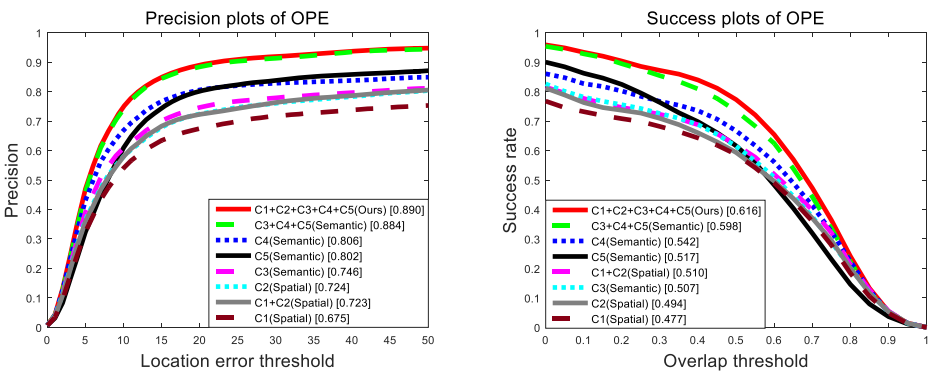
In the experiment, some parameters should be set to fixed values in advance. We set the regularization parameter of (1) to  $\lambda = 10^{-4}$ . The width of the Gauss kernel in (2) is set to  $\sigma = 0.1$ . The number of multi-scale feature channel selected in (8) is  $D = 20$ . The learning rate in (13) is set to  $\eta = 0.01$ . The thresholds in the updating strategy are set to  $\theta_1 = 5.6$  and  $\theta_2 = 3.2$ . The weights of each layer from the bottom to the top, are correspondingly set to (0.1, 0.15, 0.3, 0.5, 1).

## 5.2 Effectiveness analysis of contributions

**The analysis of DS features** To verify the validity of the proposed DS features, we draw out the features of different layers and compare them on the benchmark, the results are shown in Fig. 5. First, we use the features of single layer. As the number of layer increases in turn, the performance of using 5 layers feature is the most notable. The main reason is that we draw out the feature maps before the pooling layer, which ensures rich spatial details and facilitates localization, while semantic information can help eliminate the interference of adverse factors.

**The analysis of scale feature** Similarly, we analyze the effects of spatial and semantic features derived from different layers on scale variations, and the results are shown in Fig. 6. The features of Conv2–2 have the best performance in solving scale problems. Because the size of feature map required for the scale estimation is very small, better spatial detailed information is required. However, the semantic features of the higher layers contain a lot of information around it, which is not conducive to the determination of the scale, and the effect is poor, so it is not shown in the results. The results of six layers including spatial features and semantic features and the results using HOG feature are presented. Conv2–2 has spatial information and contains less semantic information. After resampling, most of the detailed information is retained, thus shows a better performance.

**The analysis of proposed model updating strategy** We analyze the effectiveness of the model by whether adding our proposed updating strategy or not, and we also verify it in other algorithms, such as DSST [6]. Their performance on benchmark datasets is shown in Fig. 7. By adding our proposed model updating strategy, the tracking accuracy and success rate can be improved to a certain extent. Therefore, our online updating strategy can also be extended to other correlation filter trackers. It also shows that PSR and skewness can effectively measure the fluctuation of the response map.



**Fig. 5** A comparison of localization performance using different features. Each single layer C1, C2, C3, C4, C5 represents Conv1–2, Conv2–2, Conv3–4, Conv4–4, Conv5–4 respectively. C1 + C2 represents the combination of C1 and C2, similarly, C3 + C4 + C5 represents the combination of C3, C4, C5 and C1 + C2 + C3 + C4 + C5 represents the combination of C1, C2, C3, C4, C5

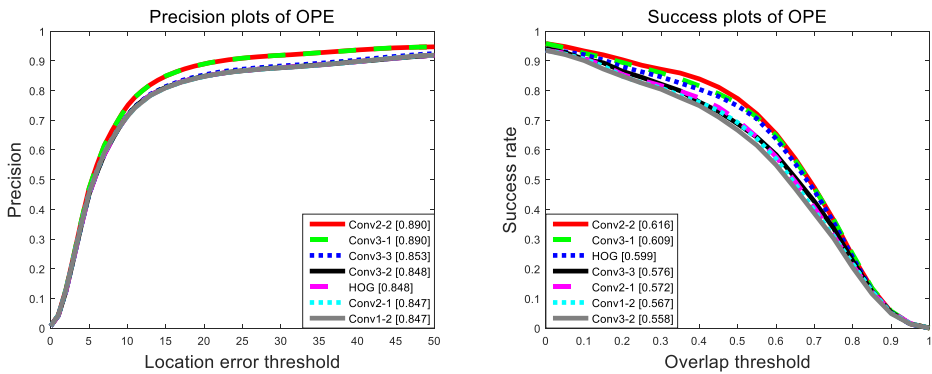


Fig. 6 A comparison of scale estimation performance using features of different layers and HOG feature

### 5.3 Comparison with state-of-the-art trackers

Furthermore, we synthesize all contributions as our proposed algorithm and compare them with 13 current state-of-the-art trackers. These methods are mainly divided into three categories: (1) trackers based on CNN include HCF [20], SiamFC [2], FCNT [33], CNN-SVM [14], DLT [32]; (2) trackers based on the correlation filter include SRDCF [8], Staple [3], DSST [6], KCF [13], CSK [12]; (3) Single or multiple online classifier trackers include DLSSVM [23], MEEM [41], Struck [9].

**Quantitative evaluation** Figure 8 shows the OPE results of our proposed algorithm and 13 trackers on 50 complete image sequences. Our tracker is far ahead in success rate, not only 0.1% behind HCF in accuracy, but also ahead of other algorithms. In conclusion, our algorithm has excellent performance.

**Attribute-based evaluation** We further analyze the performance of the tracker on image sequences with different attributes. Figs. 9 and 10 show the accuracy and success rate of the tracker respectively, and the proposed algorithm performs well in all attributes. Firstly, our method presents excellent performance in complex background, deformation and low resolution, which can be contributed to the rich spatial and semantic information of the features of different layers extracted in the deep network. Secondly, our proposed DS features have good

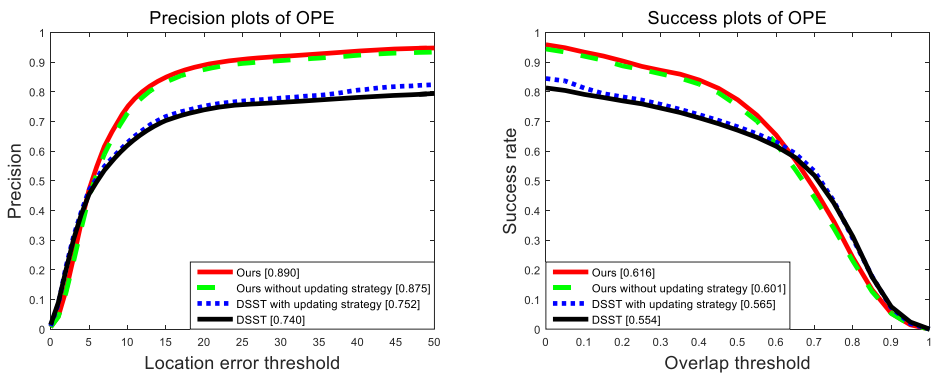
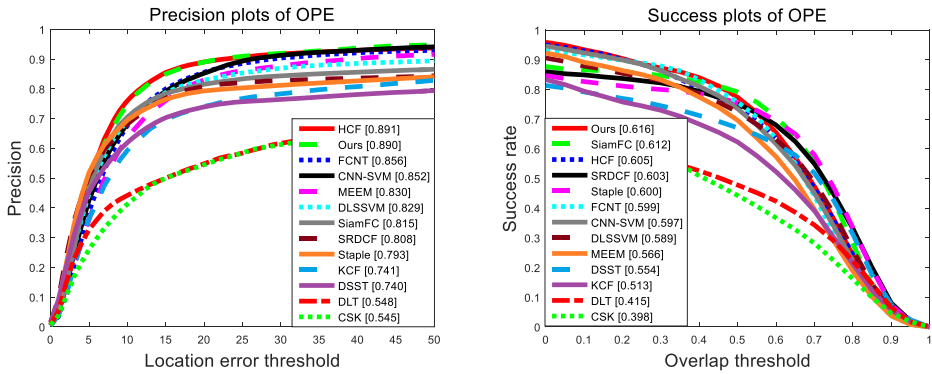


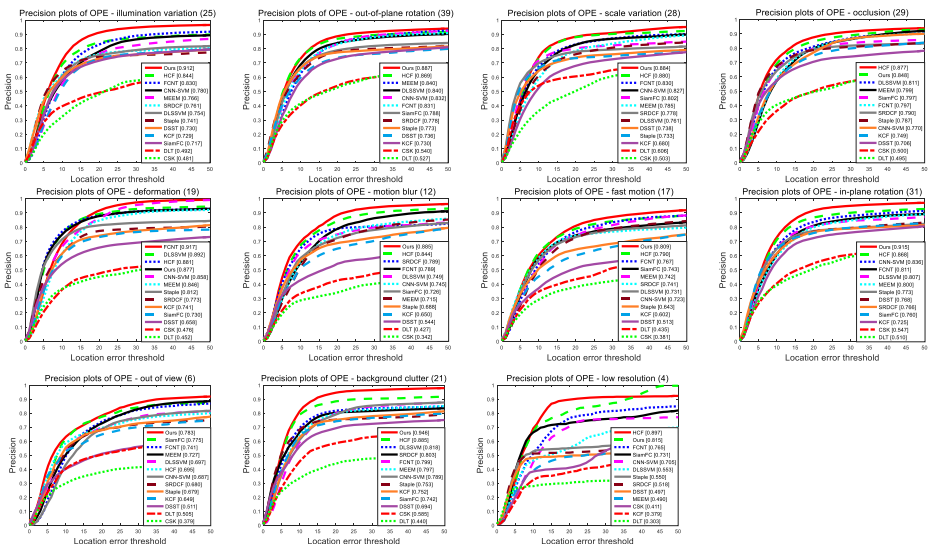
Fig. 7 A comparison of the accuracy and success rate of the algorithm based on whether to join our proposed model updating strategy or not



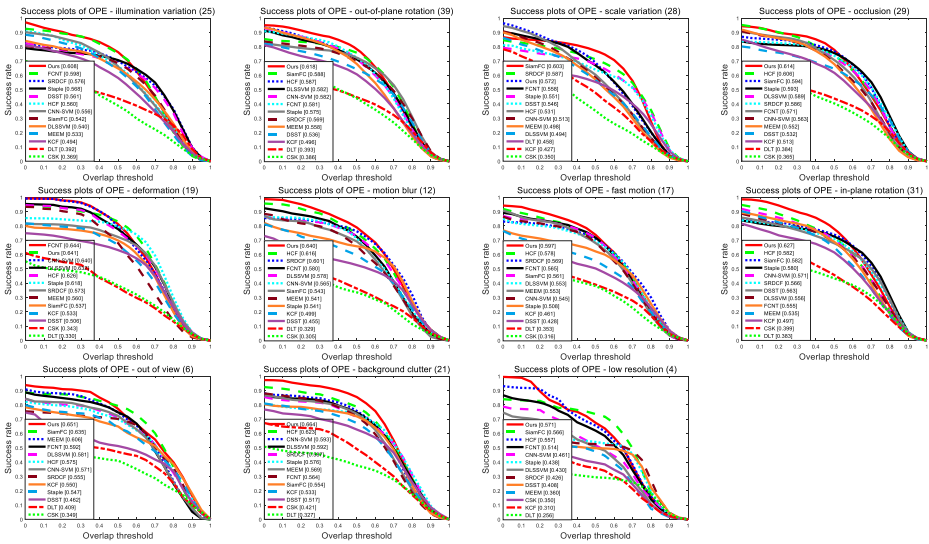
**Fig. 8** Evaluation results on 50 sequences with 13 start-of-the-art trackers

performance in solving the scale problem, which is mainly due to the strong representation ability of the deep features and the complete spatial information of the traction-bottom features. Finally, the excellent performance in solving occlusion, fast motion and out of view is mainly due to the proposed model updating strategy, and the introduction of PSR and skewness can better express the distribution of response map, thus providing guidance for model update.

**Qualitative evaluation** We select several tracking results for current popular trackers on some challenging image sequences, the detailed performance of which is shown in Fig. 11. To present clearly, we select the better performance trackers, including DSST, Staple, SRDCF, SiamFC, HCF and our tracker. Our algorithm has better performance and precise localization. However, other methods cannot solve all challenges well and only perform well on some attributes. The DSST tracker performs well in scales and illumination variation (*basketball*, *dudek*, *skating1*, *trellis*), but it will lead to tracking failure caused by occlusion, motion blur,



**Fig. 9** Attribute-based evaluation of our tracker with 13 state-of-the-art trackers results in accuracy



**Fig. 10** Attribute-based evaluation of our tracker with 13 state-of-the-art trackers results in success rate

rotation and other factors interfere (*soccer, matrix, jogging, motorrolling*). The Staple tracker combines the local HOG feature and the global color feature for tracking, so it has good real-



**Fig. 11** A comparison of bounding boxes on challenging image sequences (from left to right and top to down are *soccer, matrix, skating1, trellis, jogging, football1, dudek, basketball, carscale, motorrolling, coke, sylvester*)

time performance. The Staple tracker also presents a similar characteristic as DSST, which is better in dealing with partial occlusion and scale variation (*jogging*), but it is not ideal in background clutter (*basketball*). The SRDCF adopts color names and HOG feature to overcome the boundary effect, which has good effectiveness and real-time performance. The SRDCF has good performance when it comes to motion blur, illumination variation and scale variation (*soccer*, *trellis*, *carscale*), but it is unsatisfactory in sequences with fast motion, occlusion, rotation and background clutter (*matrix*, *jogging*, *football1*, *motorrolling*). SiamFC uses a Siamese Networks to match targets for tracking and has better performance. It is an algorithm based on matching, therefore it has very good effect in solving background clutter, occlusion, scale variation and rotation (*jogging*, *football1*, *Dudek*, *carscale*, *motorrolling*). However, because the tracking model is not updated online, there are still defects when it comes to illumination variation, motion blur, fast motion and so on (*soccer*, *matrix*, *skating1*, *basketball*). The HCF adopts hierarchical convolutional features and correlation filter to achieve better performance. Because it doesn't have a good scale adaptive mechanism, so the effect is not ideal in scale variation, motion blur and fast motion (*soccer*, *matrix*, *trellis*, *dudek*, *carscale*), but is better in illumination variation, occlusion, rotation and background clutter (*skating1*, *jogging*, *football1*, *basketball*, *motorrolling*).

We fail to track targets in two image sequences named *coke* and *sylvester* respectively. In *coke*, the target is fully occluded in the previous frames. When it appears again in the following frames, the proposed algorithm fails to find it. In *sylvester*, the target has complex background in most images, then the algorithms learn much redundant background information. Therefore, most tracking algorithms lost the target including the proposed algorithm. However, the SiamFC tracker can track the target successfully.

## 6 Conclusion

In this paper, we propose a robust target tracking algorithm based on spatial and semantic convolutional features, which can not only localize the target accurately, but also adapt to scale changes effectively. Our tracking algorithm constructs multiple correlation filters for target location by combining multi-layer convolutional features with rich spatial and semantics information. The deep network is used to extract the features of the target pyramid model, effectively overcoming the problem that scale cannot be adaptive. Moreover, two indexes, PSR and skewness are introduced to measure the fluctuation of the response map, so that a more reasonable online update model is achieved and the anti-interference ability of external factors is improved. Experiments on large benchmark datasets have shown that each of our proposed contributions is reasonable, and the proposed algorithm has achieved state-of-the-art effects on qualitative analysis, attribute-based evaluation, and quantitative analysis. In the future, we would further improve real-time performance and the efficiency of our tracking algorithm by using surveillance video sequences from multimodal (infrared and visible) cameras. Part of the difficulty is the correct registration of video sequences from multimodal cameras.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



## References

- Babenko B, Yang MH, Belongie S (2011) Robust object tracking with online multiple instance learning. *IEEE Trans Pattern Anal Mach Intell* 33(8):1619–1632
- Bertinetto L, Valmadre J, Henriques JF, et al. (2016) Fully-convolutional siamese networks for object tracking. in *Proc. Eur. Conf. Comput. Vis., Amsterdam*. 850–865
- Bertinetto L, Valmadre J, Golodetz S, et al. (2016) Staple: Complementary learners for real-time tracking. in *Proc. Eur. Conf. Comput. Vis., Las Vegas*. 1401–1409
- Bolme DS, Beveridge JR, Draper BA, et al. (2010) Visual object tracking using adaptive correlation filters. in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit., San Francisco*. 2544–2550
- Danelljan M, Khan FS, Felsberg M, et al (2014) Adaptive Color Attributes for Real-Time Visual Tracking. in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Columbus*. 1090–1097
- Danelljan M, Häger G, Khan F, et al. (2014) Accurate scale estimation for robust visual tracking. In *Proc. Br. Mach. Vis. Conf.* 1–5
- Danelljan M, Hager G, Shahbaz Khan F, et al. (2015) Convolutional features for correlation filter based visual tracking. in *Proc IEEE Int Conf Comput Vis., Santiago, Chile*. 58–66
- Danelljan M, Hager G, Shahbaz Khan F, et al. (2015) Learning spatially regularized correlation filters for visual tracking. in *Proc IEEE Int Conf Comput Vis., Santiago*. 4310–4318
- Hare S, Saffari A, Torr PHS (2016) Struck: structured output tracking with kernels. *IEEE Trans Pattern Anal Mach Intell* 38(10):2096–2109
- He K, Zhang X, Ren S, et al. (2016) Deep residual learning for image recognition. in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Las Vegas*. 770–778
- Held D, Thrun S, Savarese S (2016) Learning to track at 100 fps with deep regression networks. in *Proc. Eur. Conf. Comput. Vis., Amsterdam*. 749–765
- Henriques JF, Caseiro R, Martins P, et al. (2012) Exploiting the circulant structure of tracking-by-detection with kernels. in *Proc. Eur. Conf. Comput. Vis., Florence*. 702–715
- Henriques JF, Caseiro R, Martins P et al (2015) High-speed tracking with kernelized correlation filters. *IEEE Trans Pattern Anal Mach Intell* 37(3):583–596
- Hong S, You T, Kwak S, et al. (2015) Online tracking by learning discriminative saliency map with convolutional neural network. *Int. Conf. Mach. Learn., Lile*. 597–606
- Kalal Z, Mikolajczyk K, Matas J. Tracking-learning-detection. *IEEE Trans Pattern Anal Mach Intell*, vol. 34, no. 7, pp. 1409–1422, July. 2012
- Krizhevsky A, Sutskever I, Hinton GE (2017) Imagenet classification with deep convolutional neural networks. *Commun ACM* 60(6):84–90
- Li Y, Zhu J (2014) A Scale Adaptive Kernel Correlation Filter Tracker with Feature Integration. in *Proc. Eur. Conf. Comput. Vis., Zurich*. 254–265
- Li P, Wang D, Wang L et al (2018) Deep visual tracking: review and experimental comparison [J]. *Pattern Recogn* 76:323–338
- Lv Y (2018) Alcoholism detection by data augmentation and convolutional neural network with stochastic pooling. *J Med Syst* 42(1):2
- Ma C, Huang JB, Yang X, et al. (2015) Hierarchical convolutional features for visual tracking. in *Proc IEEE Int Conf Comput Vis., Santiago*. 3074–3082
- Ma C, Yang X, Zhang C, et al. (2015) Long-term correlation tracking. in *Proc. Eur. Conf. Comput. Vis., Boston*. 5388–5396
- Nam H, Han B (2016) Learning multi-domain convolutional neural networks for visual tracking. in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Las Vegas*. 4293–4302
- Ning J, Yang J, Jiang S, et al. (2016) Object tracking via dual linear structured SVM and explicit feature map. in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* 4266–4274
- Pan C (2018) Abnormal breast identification by nine-layer convolutional neural network with parametric rectified linear unit and rank-based stochastic pooling. *J Comput Sci* 27:57–68
- Qi Y, Zhang S, Qin L, et al. (2016) Hedged deep tracking. in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Las Vegas*, 4303–4311
- K Simonyan, A Zisserman (2014) Very deep convolutional networks for large-scale image recognition. [Online]. Available: <https://arxiv.org/abs/1409.1556>
- Song Y, Ma C, Gong L, et al. (2017) Crest: Convolutional residual learning for visual tracking. in *Proc IEEE Int Conf Comput Vis., Venice*. 2574–2583
- Sun J (2017) Polarimetric synthetic aperture radar image segmentation by convolutional neural network using graphical processing units. *J Real-Time Image Proc.* <https://doi.org/10.1007/s11554-017-0717-0>
- Sun C, Wang D, Lu H, et al. (2018) Correlation Tracking via Joint Discrimination and Reliability Learning [C]. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 489–497

30. Tu Y, Lin Y, Wang J et al (2018) Semi-supervised learning with generative adversarial networks on digital signal modulation classification [J]. *Comput Material Continua* 55(2):243–254
31. Valmadre J, Bertinetto L, Henriques J, et al. (2017) End-to-End Representation Learning for Correlation Filter Based Tracking. in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Honolulu. 5000–5008
32. Wang N, Yeung DY (2013) Learning a deep compact image representation for visual tracking. *Adv. neural inf. proces. syst.*, Lake Tahoe. 809–817
33. Wang L, Ouyang W, Wang X, et al. (2015) Visual Tracking with fully convolutional networks. in Proc IEEE Int Conf Comput Vis., Santiago. 3119–3127
34. Wang M, Liu Y, Huang Z (2017) Large margin object tracking with circulant feature maps. in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Honolulu. 4800–4808
35. Wang SH, Sun J, Phillips P et al Polarimetric synthetic aperture radar image segmentation by convolutional neural network using graphical processing units [J]. *J Real-Time Image Proc* 2017(4):1–12
36. Wu Y, Lim J, Yang M-H (2015) Object tracking benchmark. *IEEE Trans Pattern Anal Mach Intell* 37(9): 1834–1848
37. Wu Y, Jia N, Sun J (2015) Real-time multi-scale tracking based on compressive sensing. *Vis Comput* 31(4):471–484
38. Yan C, Xie H, Liu S et al (2018) Effective Uyghur language text detection in complex background images for traffic prompt identification [J]. *IEEE Trans Intell Transp Syst* 19(1):220–229
39. Yan C, Xie H, Chen J et al (2018) An effective Uyghur text detector for complex background images [J]. *IEEE Trans Multimed.* <https://doi.org/10.1109/TMM.2018.2838320>
40. Zhang K, Zhang L, Yang MH (2012) Real-time compressive tracking. in Proc. Eur. Conf. Comput. Vis., Florence, Italy. 864–877
41. Zhang J, Ma S, Sclaroff S (2014) MEEM: robust tracking via multiple experts using entropy minimization. in Proc. Eur. Conf. Comput. Vis., Zurich, Switzerland. 188–203
42. Zhang K, Zhang L, Yang MH, et al. (2014) Fast tracking via spatio-temporal context learning. in Proc. Eur. Conf. Comput. Vis., Zurich. 127–141
43. Zhang YD, Zhang Y, Hou XX et al (2018) Seven-layer deep neural network based on sparse autoencoder for voxelwise detection of cerebral microbleed. *Multimed Tools Appl* 77(9):10521–10538
44. Zhang YD, Muhammad K, Tang C (2018) Twelve-layer deep convolutional neural network with stochastic pooling for tea category classification on GPU platform. *Multimed Tools Appl.* <https://doi.org/10.1007/s11042-018-5765-3>
45. Zhang S, Wang H, Huang W et al (2018) Plant diseased leaf segmentation and recognition by fusion of superpixel, K-means and PHOG [J]. *Optik-Int J Light Electron Optics* 157:866–872
46. Zhang S, Wang H, Huang W et al (2018) Combining modified LBP and weighted SRC for palmprint recognition [J]. *SIVIP.* <https://doi.org/10.1007/s11760-018-1246-4>



**Jianming Zhang** received the B.S. and M.S. degree in 1996 and 2001 respectively from Zhejiang University and the National University of Defense Technology, China. He received the Ph.D. in 2010 from Hunan University, China. Currently, he is an associate professor and the deputy dean in the School of Computer and Communication Engineering at Changsha University of Science and Technology, China. His main research interests lie in the areas of computer vision, data mining, and wireless ad hoc & sensor networks. He has published more than 60 research papers. He is a member of ACM and CCF.



**Xiaokang Jin** received the B.S. degree from the Changsha University of Science and Technology in 2016, China. He is currently pursuing the M.S. degree in computer science and technology at Changsha University of Science and Technology. His research interests include computer vision, deep learning and object tracking.



**Juan Sun** is an undergraduate of School of Computer and Communication Engineering, Changsha University of Science and Technology, China. She was just accepted as a graduate student by the Changsha University of Science and Technology. Her research is mainly about object tracking.



**Jin Wang** received the B.S. and M.S. degree from Nanjing University of Posts and Telecommunications, China in 2002 and 2005, respectively. He received Ph.D. degree from Kyung Hee University Korea in 2010. Now, he is a professor in the School of Computer & Communication Engineering, Changsha University of Science & Technology. His research interests mainly include wireless communications and networking, performance evaluation and optimization etc. He is a Senior Member of IEEE and a member ACM.



**Arun Kumar Sangaiyah** received the M.S. degree in computer science and engineering from the Government College of Engineering, Tirunelveli, Anna University, India. He received the PhD degree in computer science and engineering from the VIT University, Vellore, India. He is presently working as an associate professor in the School of Computer Science and Engineering, VIT University, India. His area of interest includes software engineering, computational intelligence, wireless networks, bioinformatics, and embedded systems. He has authored more than 100 publications in different journals and conference of national and international repute. His current research work includes global software development, wireless ad hoc and sensor networks, machine learning, cognitive networks, and advances in mobile computing and communications. Also, he has registered one Indian patent in the area of Computational Intelligence. Besides, he is an editorial board member/associate editor of various international journals.

## Affiliations

Jianming Zhang<sup>1,2</sup> · Xiaokang Jin<sup>1,2</sup> · Juan Sun<sup>1,2</sup> · Jin Wang<sup>1,2</sup> · Arun Kumar Sangaiah<sup>3</sup>

<sup>1</sup> Hunan Provincial Key Laboratory of Intelligent Processing of Big Data on Transportation, Changsha University of Science and Technology, Changsha 410114, China

<sup>2</sup> School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha 410114 Hunan Province, China

<sup>3</sup> School of Computer Science and Engineering, Vellore Institute of Technology, Vellore 632014, India