



Scale estimation-based visual tracking with optimized convolutional activation features

Qiang Guo^{1,2} · Xuefei Cao^{1,2} · Qinglong Zou^{1,2}

Received: 15 May 2017 / Revised: 18 April 2019 / Accepted: 4 September 2019 / Published online: 12 September 2019
© Springer-Verlag GmbH Germany, part of Springer Nature 2019

Abstract

Convolutional neural networks (CNNs) have proven very effective for learning features in visual tracking. While working effectively, it is still very challenging due to the scale variations and deformation, which may cause inconsecutive tracking trajectory and distraction. In this paper, pre-train deep learning network architecture is adopted for visual tracking, by introducing a spectral pooling in the network. Then, we propose an algorithm which, by interpreting scale correlation filters as the corresponding function of convolution filters in deep neural networks, exploits multilevel CNNs activation features into a new tracking framework. Finally, two-stage fine-tuning is then introduced for updating the model to keep long-time tracking. We test the proposed tracking method on large-scale benchmark sequences. Experimental results illustrate the effectiveness of the proposed algorithm compared with other state-of-the-art methods.

Keywords Convolutional neural networks · Correlation filters · Spectrum pooling · Visual tracking

1 Introduction

Visual object tracking has long been extensively studied as it is one of the fundamental problems in computer vision. Although much progress has been made, the challenges are still remained to develop a robust tracking algorithm with efficiency in complex and dynamic environments due to many factors such as illumination variations, deformation, occlusions and significant viewpoint variation.

In general, from the perspective of representation scheme, tracking algorithms can be classified as either generative algorithms or discriminative models. The main drawback of these methods is the adaptation of low-level handcrafted feature without enough semantic data of target. In recent years, because of strong capabilities of learning feature representations, deep convolutional neural networks (CNNs) have been applied successfully throughout many computer vision tasks. Much research has been focused on CNNs methods, and a thorough recent survey can be found in [1]. Unlike

other handcrafted features such as color features [2, 3], texture features [4], the CNNs are utilized to learn deep feature representation by exploiting the multilayer perceptions which maintain more semantic information of the target. Learned features by CNNs require massive training visual data and are trained on that such as ImageNet dataset [5]. It has been shown that the deep features extracted from the convolutional layer activations can improve tracking results [6, 7]. Our approach builds on the foundation that the outputs of deeper convolutional layers contain more semantic information of targets which is effective for long-time tracking, and the scale-adaptive correlation filter is more robust for appearance variations. The convolutional activation features from three hierarchical layers are introduced for learned adaptive correlation filter tracking method [8], which has shown good performance for tracking. However, the tracking method is not robust to scale invariance or heavy occlusion of the object, also sensitive to the response position of when meeting tracking failures. Generally, global CNNs activations lack geometric invariance, which causes the CNNs tracker not to adept at handling the scale change of the targets without the adaptive template. Moreover, the max pooling process in these deep neural networks also leads to low spatial resolution.

Recently, among the discriminative tracking methods, correlation filters-based approaches [9–11] have shown

✉ Qiang Guo
royinchina@163.com

¹ Criminal Investigation Department, Criminal Investigation Police University of China, Shenyang 110854, China

² Police Terrorism Research Center, Criminal Investigation Police University of China, Shenyang 110854, China

excellent performance, and scale estimation method for discriminative correlation filters is proposed in [9, 10]. However, the target appearance is only kept in memory in a short term and tracking methods relying on correlation filters are prone to drifting due to inaccurate information updates and low-level features.

In this work, we consider CNNs architecture as translation estimation model together with a scale estimation model and learn the spatial transition of the object between the successive frames as a spatial correlation response. We also adaptively update the model for appearance variation for long-term tracking. The scale model works with deeply learned network as well as correlation filters for precisely capturing location of visual object. To summarize, the main contribution of our work is as follows.

First, by introducing a spectral pooling in the network, the optimized multilevel CNNs activation features are integrated into a new tracking framework with adaptive scale correlation filters.

Second, we propose an output update strategy of correlation response maps from different layers to calculate the location of the target.

Third, we test the proposed algorithm on a latest validation large-scale benchmark dataset [12] and achieve favorable tracking results.

2 Related work

Recent years have witnessed a popularity of the discriminative correlation filter tracker for visual object tracking. Bolme et al. [13] initially proposed the MOSSE tracker for one-dimensional feature, usually a single grayscale dimension for image description. The commonly used extension for using higher dimensional features, such as HOG or color names, can be found in Danelljan et al. [14], and Henriques et al. [15] introduced a kernelized version of the tracker. We note that Danelljan et al. [16] recently used two separate correlation filters to achieve the goal of estimating translation and scale changes. Zhang et al. [17] present the multitask correlation filter (MCF) that takes the interdependencies among different features into account to learn correlation filters jointly.

With the booming of the deep learning, some approaches have been introduced to take advantage of the CNN's feature learning capabilities for visual tracking. In general, these can be categorized into two ways to employ deep neural networks for tracking; one is to use them as a feature extractor together with a discriminative classifier [18], and the other one is to adopt a whole neural network tracking pipeline [19]. Based on the CNNs methods, Li et al. [7] exploit the CNNs end-to-end training approach to turn the tracking problem into a classification problem.

Han et al. [20] employs a CNNs for target representation with a novel regularization technique for online ensemble tracking, which has common convolutional layers but has multiple branches of fully connected layers.

Recently, fully connected layers of early CNNs model have been commonly employed for image representation [21]. The network used in [21] is pre-trained by auxiliary data (80 million samples from the Tiny Image dataset) which can learn generic image features, and then transferred to online tracking. However, the architecture of the network is not sophisticated enough, and only generic feature learning hinders the capabilities of this tracker significantly. Li et al. [7] adopt a pool of CNNs trained online without off-line pre-training, and tracking was performed as foreground-background labeling process consecutively. The application of CNNs in DCF is also under progressively explored. Ma et al. [6] employed the features from multiple hierarchical convolutional layers (conv3-4, conv4-4 and conv5-4) in a hierarchical ensemble of independent DCF trackers and learned adaptive correlation filters on each CNN layer without the need for training sampling. The C-COT [22] algorithm proposes a joint learning framework to fuse deep features from different spatial pyramids to solve the learning problem in the continuous spatial domain. MDNet [23] trains a small-scale network by multidomain methods, thus separating domain-independent information from domain-specific layers. Song et al. [24] treat tracking process as convolution and apply residual learning to take appearance changes into account. Following the end-to-end ideas, some works further use a Siamese matching structure to learn a similarity measure, which regards DCF as part of the networks.

Meanwhile, the Siamese network-based trackers have received significant attention for their well-balanced tracking accuracy and efficiency. SiamRPN [25] introduces feature extraction and region proposal subnetwork including the classification branch and regression branch. Bertinetto et al. [26] propose a fully convolutional Siamese network (SiamFC) to estimate the feature similarity region-wise between two frames. The network is trained off-line and updated without any online fine-tuning for the parameters. Song et al. [27] present the VITAL algorithm to address the existing extreme class imbalance between positive and negative samples via adversarial learning, which can also enhance the ability of feature description. These trackers consider appearance features in current frame and can hardly benefit from inter-frame information. In this work, we focus on integrating convolutional features from an optimized pre-trained deep network. Our deep optimized networks learn to attend to every discriminative candidate region.

3 Proposed method

As the convolutional activation features in deep networks are discriminative and semantically meaningful, it is more robust to large translation change in long-term tracking. While the CNNs work well, correlation filter can still be a good supplement because of its efficiency in short-term tracking. Our tracking approach is based on the combination of hierarchical correlation filter and convolutional features. For convolutional activation features, we extract it through optimized deeply learned networks based on spectral pooling, which contains more structural information. As shown in Fig. 1, scale estimation process and translation estimation process composed the tracking framework. For the robustness of our tracker, we also decompose the tracking problem into translation and scale estimation. The blue sampling rectangle includes both the target and surrounding context area. We first adopt the CNNs for feature extraction. However, the pre-trained CNN is not originally designed for visual tracking task as it is more suitable for image classification. Therefore, the pre-trained network is not directly utilized in our visual task. We adopt

the spectral pooling process in pre-trained CNNs. Given a pre-trained CNN through spectral pooling, the outputs of the different pooling layer feature maps are used as multi-channel features.

We also fine-tune the network by learning generic features for all objects and specific features for tracking object sequentially. Learning generic features is trained on the validated ImageNet detection dataset. However, the fine-tuned CNNs during this phase can be activated for any object in the scene, because all layers are fine-tuned. In our work, followed by pre-training on the source task, the parameters of layers C, ReLu, P and FC are transferred to the tracking task. Due to the pooling operators used in the CNN model, the spatial resolution of the pooling layers is different. We then get each feature map.

In this section, the technical details on the CNNs layer features and scale-adaptive correlation filters are presented.

3.1 Convolutional activation features based on spectral pooling

Traditional deep CNNs offer a class of hierarchical models to learn features directly from image pixels, which combine

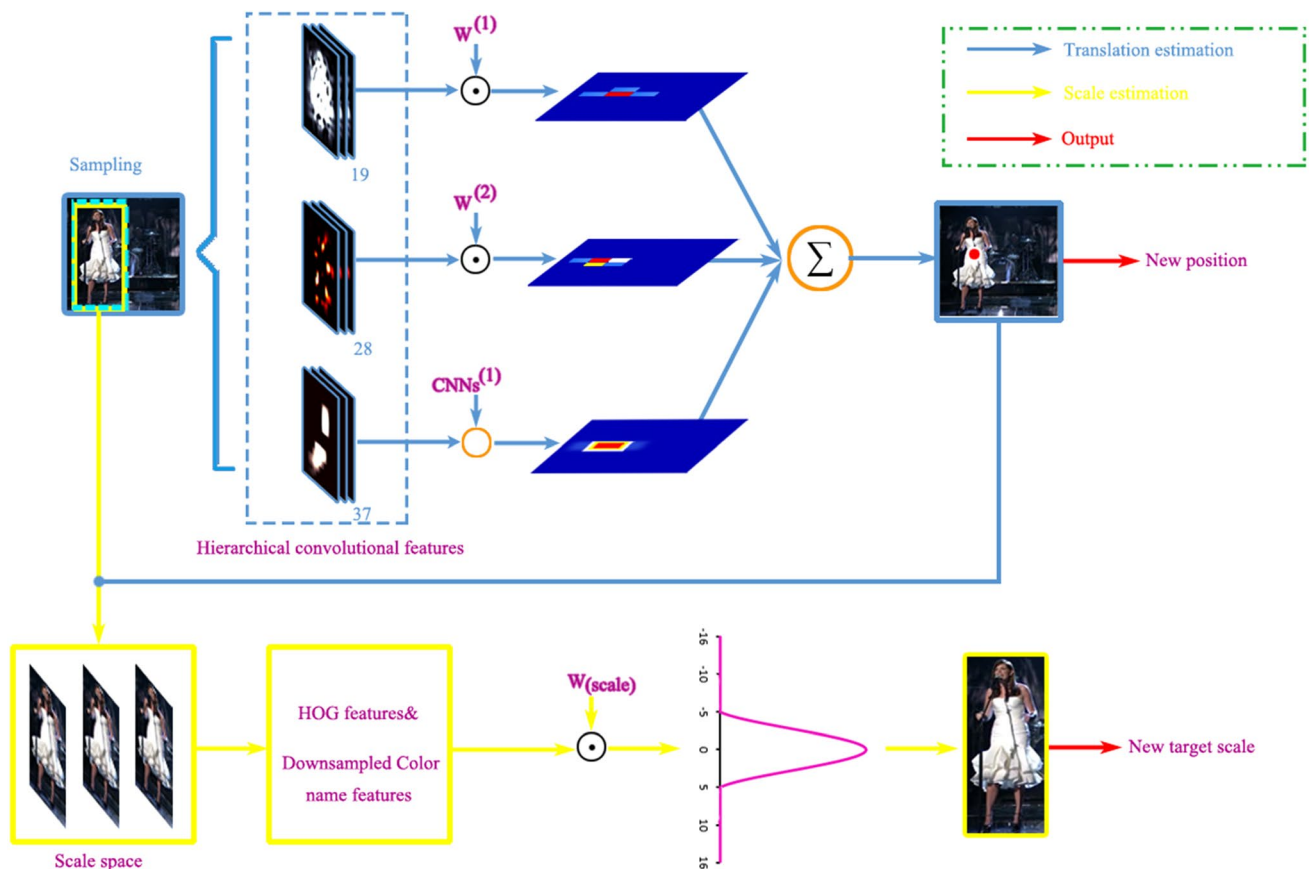


Fig. 1 Main steps of the proposed method

shared weights, subsampling and local receptive fields to ensure some degree of shift, scale and deformation invariance. The idea of shared weights and local receptive fields is organized in the convolutional layer. The feature maps can be obtained by a filter of a predefined size convolved with the input image. The max pooling process in conventional CNNs could reduce the image resolutions with a large probability, which destroys the spatial position information that is crucial to the final tracking result. Motivated by [28], we use the convolutional activation features from a spectral pooling-based CNNs model for feature extraction. Along with the CNNs forward propagation, the semantical discrimination ability is strengthened, as well as a gradual reduction in spatial resolution for precise localization.

Spectral pooling preserves considerably more information and structure for the same number of parameters (see the third row of Fig. 1) because the frequency domain provides a sparse basis for inputs with spatial structure. Spectral parameterization of CNNs is the first step in our solution, which computes the convolution of the filter with inputs. Assuming that for some layer of CNNs, we try to learn filters or convolution kernel of size $H \times W$. Each filter in our network is parameterized directly in Fourier domain or Hartley transform. Inverse DFT (discrete Fourier transformation) is calculated for obtaining spatial representation. Training proceeds quickly by computing the inputs or feature maps with Fourier transformation in mini-batches mode as standard CNNs.

The pooling process in CNNs could reduce the image resolutions with a big probability, which destroys the spatial position information that is crucial to the final tracking result. Max pooling only captures the strongest activation of the filter template with the input for each small region, which

might lose the useful information carried by these nonmaximal activations. Those steps in forward propagation of this spectral pooling are listed in Algorithm 1. The algorithm can simplify the spectral pooling by Hartley transform.

Algorithm 1: Spectral pooling

Input: feature map $x \in \mathbb{C}^{H \times W}$, dimension reduction size H^*W

1. Spectral transform by DFT or Hartley transform
2. Crop frequency representation
3. Inverse transform

Output: Pooled map

As shown in Algorithm 1, we get the pooled map approximated by taking inverse Fourier transform back into spatial domain. Given an input feature map and some desired output map dimensionality, first, we compute the spectral transform of the input by DFT or Hartley transform. We then crop the frequency representation by maintaining only the desired map dimensionality by submatrix of frequencies. After the inverse transform, the feature map can be mapped into spatial domain, which produces the results of the pooling layers in the translation estimation in Fig. 1. Since the pooling is a lossy procedure, a motivation of our work is to exploit a different pooling approach for less loss in the dimensionality reduction. Our approach preserves considerably more information per parameter than other pooling strategies and enables flexibility in the choice of pooling output dimensionality, which can keep better feature description of the feature map, as the frequency basis captures typical filter structure well in Fig. 2.

The employed network contains three convolutional layers and uses a RGB image as an input after preprocessing. Before computing the convolutional features, the image

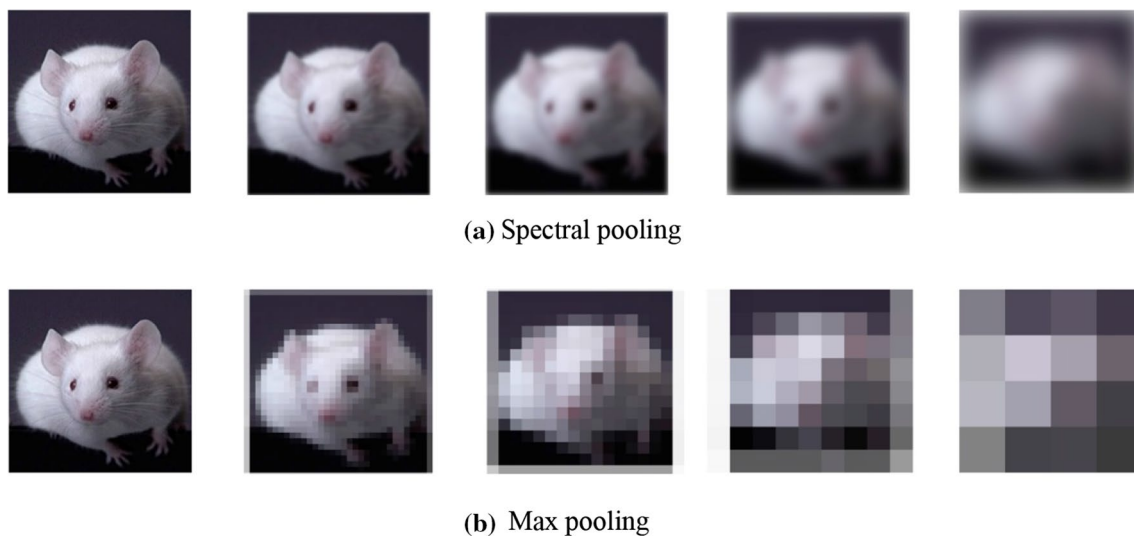


Fig. 2 Approximation of two pooling algorithms

patch is preprocessed by first resizing it to the input size 224×224 and then subtracting the mean of the network training data. The extracted features are always multiplied with a Hann window to perform a Fourier transform.

3.2 Convolutional activation features combined with correlation filter for translation estimation

Because of the computation efficiency of correlation in the frequency domain, correlation filters have been widely used in visual tracking. We adopt the scale estimation method that uses two divided correlation filters to first estimate translation and then scale variations in the object. A representative correlation filters tracker for modeling the appearance of candidate object is composed of a filter \mathbf{w} trained on the same size of image patch \mathbf{x} of $M \times N$ pixels, where all the circular shifts of $\mathbf{x}_{m,n}$, $(m,n) \in \{0, 1, \dots, M-1\} \times \{0, 1, \dots, N-1\}$ are generated as training samples with Gaussian function label $y(m,n)$. Then, the correlation filter is learned by solving the regularized minimization problem as follows:

$$\mathbf{w}^* = \arg \min_w \sum_{m,n} \|\mathbf{w} \cdot \phi(\mathbf{x}_{m,n}) - y(m,n)\|^2 + \lambda \|\mathbf{w}\|_2^2 \quad (1)$$

where ϕ denotes the mapping to a kernel space and λ is a regularization parameter ($\lambda \geq 0$). The learned filter \mathbf{w} contains the coefficients of a Gaussian ridge regression [15]. Then, the fast Fourier transformation (FFT) is taken for computing the correlation, and the objective function in (1) is minimized as:

$$\mathbf{W} = \sum_{m,n} \mathbf{a}(m,n) \phi(\mathbf{x}_{m,n}) \quad (2)$$

where \mathbf{a} is the coefficient defined by

$$\mathbf{A} = \mathcal{F}(\mathbf{a}) = \frac{\mathcal{F}(\mathbf{y})}{\mathcal{F}(\phi(\mathbf{x})\phi(\mathbf{x})) + \lambda} \quad (3)$$

In (3), $\mathcal{F}(\bullet)$ denotes the discrete Fourier operator, and $\mathbf{y}(m,n) = \{y(m,n) | (m,n) \in \{0, 1, \dots, M-1\} \times \{0, 1, \dots, N-1\}\}$. The tracking task is carried out on an image patch \mathbf{Z} in the new frame with the search window size $\mathbf{M} \times \mathbf{N}$ by computing the response map as

$$\hat{\mathbf{y}} = \mathcal{F}^{-1}(\mathbf{A} \odot \mathcal{F}(\phi(\mathbf{z}) \cdot \phi(\hat{\mathbf{x}}))) \quad (4)$$

where $\hat{\mathbf{x}}$ denotes the learned target appearance model and \odot is the Hadamard product. Therefore, the new position of the target is detected by searching for the maximum value of the response map $\hat{\mathbf{y}}$.

The above-mentioned trackers only use grayscale features. Extending this to higher dimensional features is commonly done by optimizing one filter for each feature dimension. Considering each individual feature channel, the

learned filter in frequency domain on the d th ($\mathbf{d} \in \mathbf{D}$) channel in formulation (2) and (3) can be transformed as

$$\mathbf{W}^d = \frac{\mathbf{Y} \odot \bar{\mathbf{X}}^d}{\sum_{i=1}^D \mathbf{X}^i \odot \bar{\mathbf{X}}^i + \lambda} \quad (5)$$

where \mathbf{x}^l is the l th layer of feature vector and \mathbf{Y} is the Fourier transformation form of \mathbf{y} . We denote the feature vector on the l th layer in the new frame by \mathbf{Z} . Then, the corresponding response map is computed by

$$f = \mathcal{F}^{-1} \left(\sum_{d=1}^D \mathbf{W}^d \odot \bar{\mathbf{Z}}^d \right) \quad (6)$$

As shown in Fig. 1, we incorporate scale estimation into a hierarchical convolutional network framework combined with a correlation filter process. The correlation filter for translation estimation is firstly used, which includes two response maps produced by CNNs, and one response map generated by correlation filter over the deep activation feature is complementary to each other. When the optimal translation has been found, the scale filter is then launched to estimate.

In translation estimation, the target position can be obtained by searching for the maximum value of linear combination of different response maps of the three layers. For each resized input image, it is

$$\hat{f}_{\text{final}} = \arg \max \sum_{l=1,2,3} \beta_l f_l \quad (7)$$

where \hat{f}_{final} is the final response value, f_3 denotes the convolutional activation response map, f_1 and f_2 are the correlation response maps over the pre-trained network model and β_l is the weighting parameter.

3.3 HOG features and name features for scale estimation

After estimate translation, we use HOG features and concatenate them with the color names features for scale estimation. We use HOG features for the translation filter and concatenate them with the compressed color names features by PCA. The training image for training scaled correlation filter model, which uses variable patch sizes around the target center, is calculated by the combined features. Let $P \times R$ denote the target size, K be the number of scales and S be the size of the scale filter. For each $s \in \left\{ \theta^n \left\lfloor \left[-\frac{S-1}{2} \right], \dots, \left[-\frac{S-1}{2} \right] \right\rfloor \right\}$, an image patch J_n of size $sP \times sR$ is extracted and centered around the target position. By maximizing the correlation output in (6), we can get the target scale differences between two consecutive frames.

3.4 Model update

We firstly update the translation model and then the scale estimation model. When the final response map \hat{y}_{final} in Eq. (7) is larger than a threshold, translation model is updated as follows:

$$\hat{W}_t^d = e^{-1/\tau} \hat{W}_{t-1}^d + (1 - e^{-1/\tau}) W_t^d \quad (8)$$

$$\hat{A}_t^d = e^{-1/\tau} \hat{A}_{t-1}^d + (1 - e^{-1/\tau}) A_t^d \quad (9)$$

4 Experiments

In this section, to show the accuracy and efficiency of our tracker in comparison with the state of the art, qualitative and quantitative experimental results are presented to evaluate the performance of the proposed tracker on the list of video sequences publicly available by [28], which contain challenging variations including background changes, pose and scale changes, occlusions and background clutter (the implementations provided by the authors used for fair comparisons). Our method is implemented in MATLAB on Intel Core I7 3.4 GHz CPU with 32G RAM. The MatConvNet toolbox in [29] is used for CNNs based feature generation and transfers the forward propagation of CNNs to a GeForce GTX Titan GPU. The average tracking speed of our tracker is 10.6 frames/s. Most computational power is spent to the forward propagation process to extract deep convolutional features employing the VGG-Net-19 model. The CNN is fine-tuned for 90 k iterations for any objects in the scene with learning rates of 0.0001 and 0.001 for fully connected layers. And the maximum number of iterations for the specific target fine-tuning in the first frame is set to be equal to 500.

Table 1 shows the tracking results for different pooling methods. The first experiment uses pooling techniques that result in data reduction of 75%, and the second experiment uses pooling techniques that result in data reduction of 93.75%. Both experiments were evaluated on tracking

accuracy. Spectral pooling method enhances the tracking success rate than max pooling or average pooling when stride and kernel size is big or small and truncation and reduction rate is high or low as well.

The success plot shows a comparison of our trackers with state-of-the-art methods on the OTB dataset containing 100 videos [28]. The area-under-the-curve (AUC) scores for the trackers are reported in the legend. For the real-data experiments, our method is compared with the other six recent relative state-of-the-art algorithms including DeepSRDCF [11], the CF2 [6], the SRDCF [30], the HDT [31], LCT [32] and KCF [15].

There are two widely used evaluation criteria in many experiments to assess the performance of the tracker. One is the success rate (SR). The overlap score of success rate is defined as score = $\frac{A \cap B}{A \cup B}$, where A is the ground-truth bounding box and B is the track result rectangle, and \cap and \cup represent the intersection and union of the two regions, respectively. Another widely used evaluation metric for object tracking is the center location error, which computes the average Euclidean distance between the center locations of the tracked targets and the manually labeled ground-truth positions of all the frames. However, the center location error only measures the pixel difference and does not reflect the size and scale of the target object. Besides, using one success rate value at a specific threshold (e.g. score = 0.5) for tracker evaluation may not be representative. Therefore, we adopt one-pass evaluation (OPE) and temporal robustness evaluation (TRE) in which an algorithm is evaluated from a particular starting frame, with the initialization of the corresponding ground-truth object state, until the end of an image sequence for the robustness evaluations of trackers.

Figures 3 and 4 illustrate overall performance of the top seven evaluated tracking algorithms which use the area under curve (AUC) of each success plot to rank the tracking algorithms as [10]. Our tracker performs well in TRE and OPES, which suggests multilevel CNNs activation features in deep neural networks yield much more stable and accurate results than other compared trackers.

Tables 2 and 3 summarize the tracking scores for state-of-the-art trackers, and the best method is highlighted in bold

Table 1 Ablation for different pooling methods on tracking performance

Pooling	Stride	Kernel size	Truncation (%)	Reduction (%)	Tracking success rate (%)
Max	2	2		75	64.1
Mean	2	2		75	62.5
DFT			75	75	65.3
Max	4	4		93.75	54.0
Mean	4	4		93.75	55.8
DFT			93.75	93.75	57.1

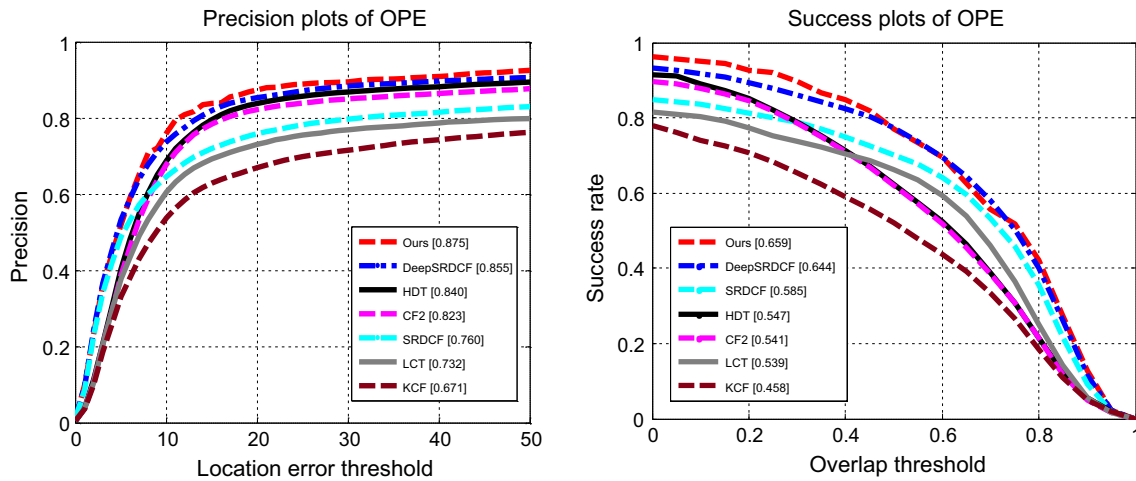


Fig. 3 Precision plots and success plots of OPE over the TB-100 sequences

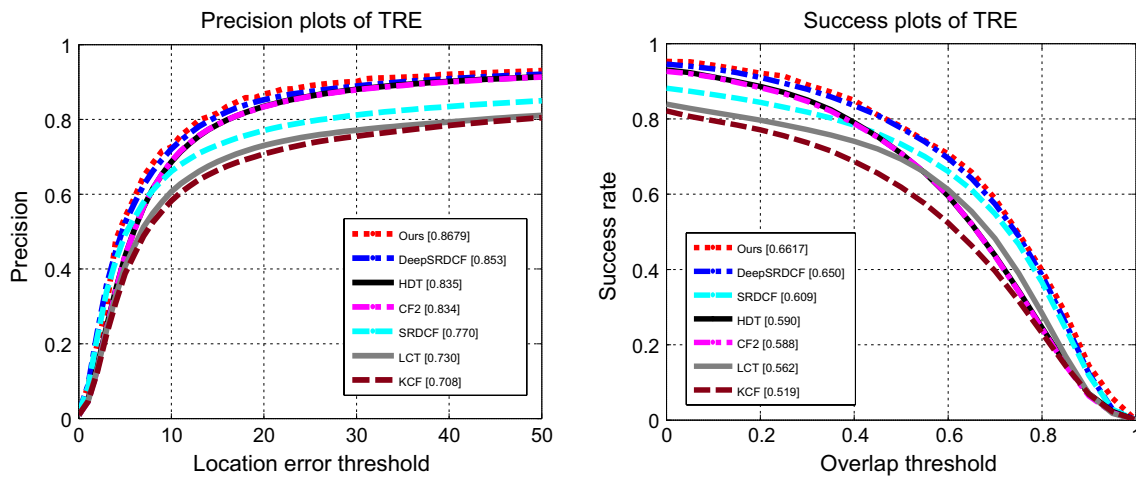


Fig. 4 Precision plots and success plots of TRE over the TB-100 sequences

Table 2 Average scores of OPE for different trackers on the OTB100 dataset

	Ours	DeepSRDCF	HDT	SRDCF	CF2	LCT	KCF
Precision	0.875	0.855	0.840	0.760	0.823	0.732	0.671
Success	0.659	0.644	0.585	0.547	0.541	0.539	0.458

Table 3 Average scores of TRE for different trackers on the OTB100 dataset

	Ours	DeepSRDCF	HDT	SRDCF	CF2	LCT	KCF
Precision	0.868	0.863	0.835	0.770	0.834	0.730	0.708
Success	0.662	0.660	0.590	0.609	0.588	0.562	0.519

color. It can be seen from Tables 2 and 3 that our algorithm performs better than other state-of-art related algorithms. It should be noted that the proposed method exploits only convolutional activation features to learn the object and background, in which the scale correlation filters is also adopted

with low computational complexity, yet it outperforms DeepSRDCF that resorts to complicate filter techniques in terms of both accuracy and efficiency.

For fair comparison, we compare our tracker with the recent state-of-the-art methods in Table 4. Results are summarized in

Table 4 Performance of newest trackers for each individual attribute on the VOT2015 dataset

Tracker	Illumination change		Motion change		Occlusion		Scale change		Redetection		
	A	R	A	R	A	R	A	R	A	F	S (%)
MDNet	0.68	1.07	0.56	15.73	0.54	13.93	0.56	11.20	0.64	×	0
C_COT	0.66	1.00	0.49	19.00	0.51	18.00	0.51	12.00	0.38	×	0
SiamRPN	0.70	1.04	0.57	18.25	0.56	16.56	0.57	14.05	0.64	×	0
SiameFC	0.67	2.00	0.51	30.00	0.47	20.00	0.51	20.00	0.63	×	0
Ours	0.66	2.87	0.59	15.98	0.55	13.28	0.58	11.00	0.65	0	2

'A' denotes the accuracy score, 'R' the failure times, 'F' the results of the redetection experiment with the average number of frames required for redetection (frames) and 'S' the percentage of sequences with successful redetection (success); long-term attribute for last column is tested on the VOT2018 [33]

The bold values denote the best performance tracker in each column

Table 4. MDNet, C_COT and SiameFC trackers confirm their short-term design. Our tracker is pseudo-long-term tracker because of small search window and its inefficient updating model.

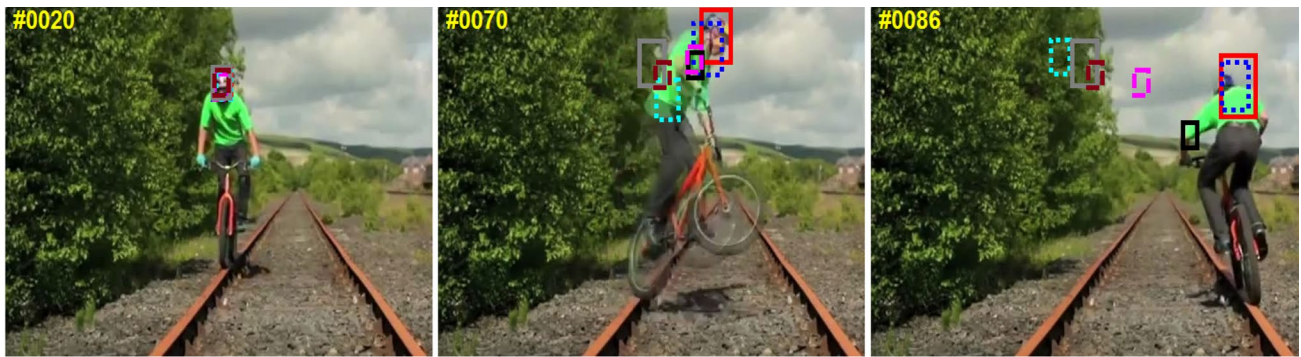
We also qualitatively evaluate the performances of tracking results of some widely used representative video sequences. We show some representative tracking results on the sequences for DeepSRDCF, HDT, CF2, SRDCF, LCT and KCF tracking methods presented in Fig. 5. Figure 5a, e shows the tracking results of Biker sequences and Human6 sequences with Out-of-View, Out-of-Plane Rotation and deformation attributes. Our tracker can handle deformation well due to its adaptive learned correlation filters on each convolutional layer to explore the target appearance. The target in Fig. 5d, f, videos where the target maintains the same size seems to be the optimal operating point for all trackers. In Fig. 5d, e, f, the visual object suffers scale changes. When large-scale variation occurs, our tracker obtains higher precision score. It is somewhat expected because the tracker algorithm incorporates scale estimation during tracking compared to ours. There is a significant dip in performance around $6\times$ variation in scale for the KCF. Figure 5b shows the tracking results of challenging sequences with fast motion attributes and deformation attributes. Our tracker can handle deformation and motion variation well due to its pooling method and translation model to

explore the target appearance. The target in Fig. 5c suffers illumination variation; ours shows favorable performance to tackle these challenges, which is attributed to the online fine-tuning.

Overall, ours shows favorable performance to tackle these challenges, which is attributed to the adaptive scale appearance model and output update strategy of correlation response maps.

5 Conclusion

In this paper, we presented an efficient deep learning-based tracking algorithm implemented by integrating improved multilevel CNNs activation features with scale correlation filters, in which new architectural network composed of multilevel response map and correlation filters is utilized to search the best match object region with higher response score. We introduced the scaled discriminative correlation filters to overcome the scale variations in the tracing scenery. Experimental results on challenging video sequences with partial occlusion, illumination change, deformation and other aspects show that our tracker achieves favorable performance against other three similar states-of-the-art algorithms.



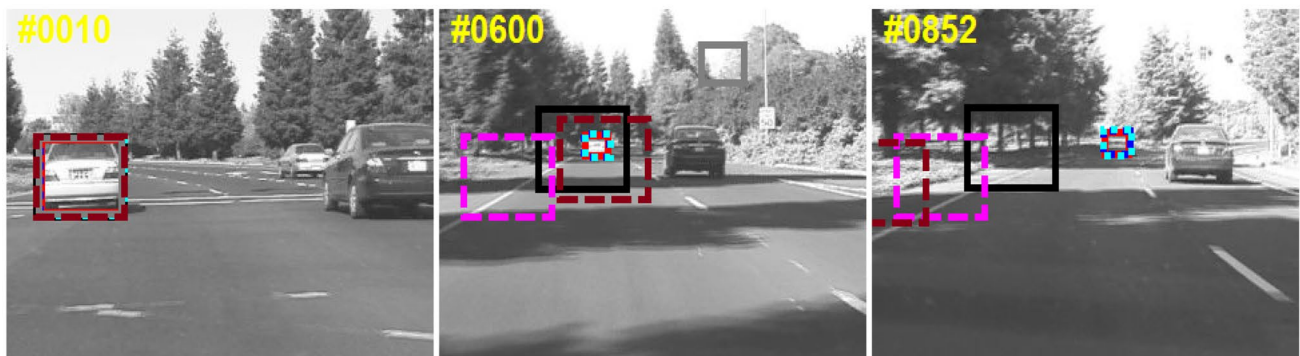
(a) Biker



(b) Bird2



(c) KiteSurf



(d) Car1

Fig. 5 A comparison of our method with the state-of-the-art trackers in some representative sequences



(e) Human6



(f) Singer1



Fig. 5 (continued)

Acknowledgements This work is supported by the National Natural Sciences Foundation of China under Grant Nos. 61603415, 61602322, 61503274 and the Fundamental Research Funds for the Central Universities under Grant No. D2019021.

References

- Kristan, M., Matas, J., Leonardis, A., et al.: The visual object tracking vot2015 challenge results. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 1–23 (2015)
- Danelljan, M., Khan, F.S., Felsberg, M., van de Weijer, J.: Adaptive color attributes for real-time visual tracking. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, pp. 1090–1097 (2014)
- Possegger, H., Mauthner, T., Bischof, H.: In defense of color-based model-free tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2113–2120 (2015)
- Lebeda, K., Hadfield, S., Matas, J., et al.: Texture-independent long-term tracking using virtual corners. *IEEE Trans. Image Process.* **25**(1), 359–371 (2016)
- Russakovsky, O., Deng, J., Su, H., et al.: Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **115**(3), 211–252 (2015)
- Ma, C., Huang, J.B., Yang, X., et al.: Hierarchical convolutional features for visual tracking. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3074–3082 (2015)
- Li, H., Li, Y., Porikli, F.: Deeptack: Learning discriminative feature representations by convolutional neural networks for visual tracking. In: Proceedings of the British Machine Vision Conference. BMVA Press (2014)
- Ma, C., Xu, Y., Ni, B., et al.: When correlation filters meet convolutional neural networks for visual tracking. *IEEE Signal Process. Lett.* **23**(10), 1454–1458 (2016)

9. Danelljan, M., Häger, G., Khan, F., et al.: Accurate scale estimation for robust visual tracking. In: British Machine Vision Conference, Nottingham, September 1–5, 2014. BMVA Press (2014)
10. Li, Y., Zhu, J.: A scale adaptive kernel correlation filter tracker with feature integration. In: European Conference on Computer Vision. Springer, pp. 254–265 (2014)
11. Danelljan, M., Hager, G., Khan, F.S., et al.: Convolutional features for correlation filter based visual tracking. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 58–66 (2015)
12. Wu, Y., Lim, J., Yang, M.H.: Object tracking benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**(9), 1834–1848 (2015)
13. Bolme, D.S., Beveridge, J.R., Draper, B.A., Lui, Y.M.: Visual object tracking using adaptive correlation filters. In: CVPR, 2010 (2010)
14. Danelljan, M., Khan, F.S., Felsberg, M., van de Weijer, J.: Adaptive color attributes for real-time visual tracking. In: CVPR, 2014 (2014)
15. Henriques, J.F., Caseiro, R., Martins, P., et al.: High-speed tracking with kernelized correlation filters. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**(3), 583–596 (2015)
16. Li, Y., Zhang, Y., Xu, Y., et al.: Robust scale adaptive kernel correlation filter tracker with hierarchical convolutional features. *IEEE Signal Process. Lett.* **23**(8), 1136–1140 (2016)
17. Zhang, T., Xu, C., Yang, M.-H.: Multi-task correlation particle filter for robust object tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4335–4343 (2017)
18. Wang, N., Li, S., Gupta, A., Yeung, D.-Y.: Transferring rich feature hierarchies for robust visual tracking. *arXiv preprint arXiv:1501.04587* (2015)
19. Wang, N., Yeung, D.-Y.: Learning a deep compact image representation for visual tracking. In: Burges, C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K. (eds.) *Advances in Neural Information Processing Systems 26*, pp. 809–817. Red Hook, Curran Associates (2013)
20. Han, B., Sim, J., Adam, H.: Branchout: regularization for online ensemble tracking with convolutional neural networks. In: CVPR, 2017 (2017)
21. Wang, L., Ouyang, W., Wang, X., et al.: Visual tracking with fully convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3119–3127 (2015)
22. Danelljan, M., Robinson, A., Khan, F.S., Felsberg, M.: Beyond correlation filters: learning continuous convolution operators for visual tracking. In: Proceedings of the European Conference on Computer Vision, pp. 472–488 (2016)
23. Nam, H., Han, B.: Learning multi-domain convolutional neural networks for visual tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4293–4302 (2016)
24. Song, Y., Ma, C., Gong, L., Zhang, J., Lau, R.W.H., Yang, M.-H.: Crest: convolutional residual learning for visual tracking. In: IEEE International Conference on Computer Vision (2017)
25. Li, B., Yan, J., Wu, W., Zhu, Z., Hu, X.: High performance visual tracking with siamese region proposal network. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
26. Bertinetto, L., Valmadre, J., Henriques, J.F., Vedaldi, A., Torr, P.H.S.: Fully-convolutional siamese networks for object tracking. In: Proceedings of the European Conference on Computer Vision Workshops, pp. 850–865 (2016)
27. Song, Y., Ma, C., Wu, X., Gong, L., Bao, L., Zuo, W., Shen, C., Rynson, L., Yang, M.-H.: Vital: visual tracking via adversarial learning. In: CVPR, 2018 (2018)
28. Rippel, O., Snoek, J., Adams, R.P.: Spectral Representations for Convolutional Neural Networks. *Comput. Sci* (2015)
29. Vedaldi, A., Lenc, K.: Matconvnet: convolutional neural networks for matlab. In: Proceedings of the 23rd ACM International Conference on Multimedia. ACM, pp. 689–692 (2015)
30. Danelljan, M., Hager, G., Khan, F.S., et al.: Learning spatially regularized correlation filters for visual tracking. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4310–4318 (2015)
31. Qi, Y., Zhang, S., Qin, L., et al.: Hedged deep tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4303–4311 (2016)
32. Ma, C., Yang, X., Zhang, C., et al.: Long-term correlation tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5388–5396 (2015)
33. Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pfugfelder, R., Zajc, L.C., Vojir, T., Bhat, G., Lukezic, A., Eldesokey, A., Fernandez, G., et al.: The sixth visual object tracking vot2018 challenge results. In: ECCV Workshops, 2018 (2018)

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

Qiang Guo received master degree in 2007 and doctorate degree in 2016 from the College of Information Science and Engineering of Northeastern University. Now he is currently a lecturer in Criminal Investigation Police University of China. His recently main research focuses on abnormal behavior recognition and predicting terrorism with machine learning.

Xuefei Cao is an associate professor at the Criminal Investigation Police University of China. He is a PhD candidate at Charles Sturt University of Australia. His current research focuses on policing anti-terrorism with artificial intelligence.

Qinglong Zou is a lecturer at the Criminal Investigation Police University of China (CIPUC). He received his MS degree from CIPUC.