Neurocomputing 455 (2021) 1–11

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Recent advances of single-object tracking methods: A brief survey

Yucheng Zhang [a], Tian Wang [a,b,*], Kexin Liu [a,c], Baochang Zhang [a,b], Lei Chen [d]

[a] School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China
[b] Institute of Artificial Intelligence, Beihang University, Beijing 100191, China
[c] Beijing Advanced Innovation Center for Big Data and Brain Computing, Beihang University, Beijing 100191, China
[d] Advanced Research Institute of Multidisciplinary Science, Beijing Institute of Technology, Beijing 100081, China

ARTICLE INFO

Article history:
Received 23 August 2020
Revised 2 May 2021
Accepted 4 May 2021
Available online 29 May 2021
Communicated by Zidong Wang

Keywords:
Single-object tracking
Correlation filters
Deep learning
Computer vision

ABSTRACT

Single-object tracking is regarded as a challenging task in computer vision, especially in complex spatio-temporal contexts. The changes in the environment and object deformation make it difficult to track. In the last 10 years, the application of correlation filters and deep learning enhances the performance of trackers to a large extent. This paper summarizes single-object tracking algorithms based on correlation filters and deep learning. Firstly, we explain the definition of single-object tracking and analyze the components of general object tracking algorithms. Secondly, the single-object tracking algorithms proposed in the past decade are summarized according to different categories. Finally, this paper summarizes the achievements and problems of existing algorithms by analyzing experimental results and discusses the development trends.

1. Introduction

Object tracking is an important research topic in the field of computer vision. There are many related directions for visual object tracking, such as single-object tracking [1,2], multi-object tracking [3,4], 3D object tracking [5] and video object segmentation [6,7]. Object tracking is widely applied in the fields of automatic driving [8], human–computer interaction [9,10], video surveillance [11–13] and so on. In the last 10 years, the application of correlation filters and deep learning improves the performance of trackers with a large margin, which makes it possible to apply in more practices.

The single-object tracking task is to follow the object labeled in a video sequence as in Fig. 1. Most existing single-object trackers carry out tracking in 4 steps. (1) Inputting images and extracting features. Features have a huge impact on tracker performance, the commonly used features are manual features and convolutional features. Deep learning has powerful feature extraction capabilities, which makes it a research hotspot. (2) Generating candidate areas. The common methods are particle filters prediction [14] and sliding window [15,16]. The former uses inference to predict candidate areas, while the latter uses exhaustive methods. (3) Building the tracking model. Building an object tracking model and accurately selecting a candidate area are the core of object tracking tasks. Existing object tracking models include generative models and discriminative models. (4) Updating the model opportunely. The model update is necessary due to changes in the environment and object itself. Hence, a suitable model update strategy greatly improves the robustness of trackers.

In most cases, the application scenes of trackers are pretty complicated, there are many interferences, including occlusion, target deformation, similar objects, scale transformation, lighting change, low resolution, fast motion and so on [17]. Therefore, accurate single-object tracking faces plenty of challenges. For the purpose of overcoming the above difficulties and improving tracking performance, many outstanding trackers have been proposed in succession. In order to grasp the development status of object tracking, analyze the bottleneck of current trackers, and explore further innovation directions, this paper summarizes and analyzes the object tracking algorithms in the past decade.

In this paper, as shown in Fig. 2, we summarize the development of single-object tracking in the past decade and divide existing algorithms into correlation filters-based algorithms [1,34–36,38–41,43–52,54,55] and deep learning-based algorithms [56–61,63–66,68,69,2,70,71,76–80,84,85]. In the early stage of object tracking, optical flow methods [86–88], filters methods [89–92] and kernel-based methods [93,94] have been used for tracking in succession. However, complex calculations and low accuracy limit further development. Correlation filters and deep learning break the development bottleneck of object tracking. The application of

* Corresponding author at: School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China.
E-mail address: wangtian@buaa.edu.cn (T. Wang).

https://doi.org/10.1016/j.neucom.2021.05.011
0925-2312/

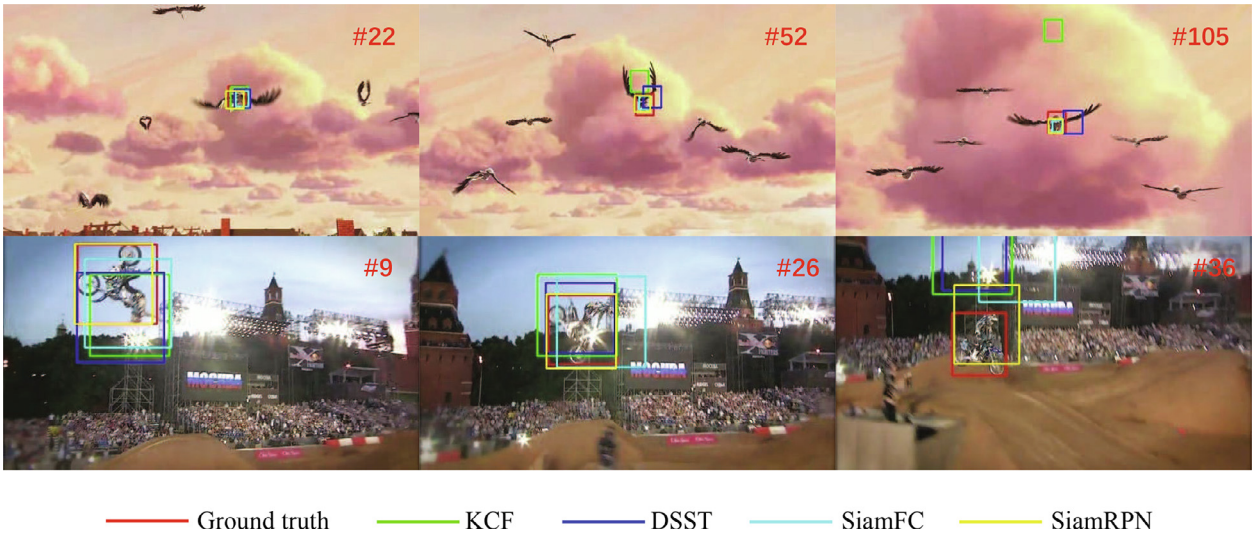


Fig. 1. We visualize the ground truth of video sequences and evaluation results of KCF [36], DSST [40], SiamFC [2], SiamRPN [70].

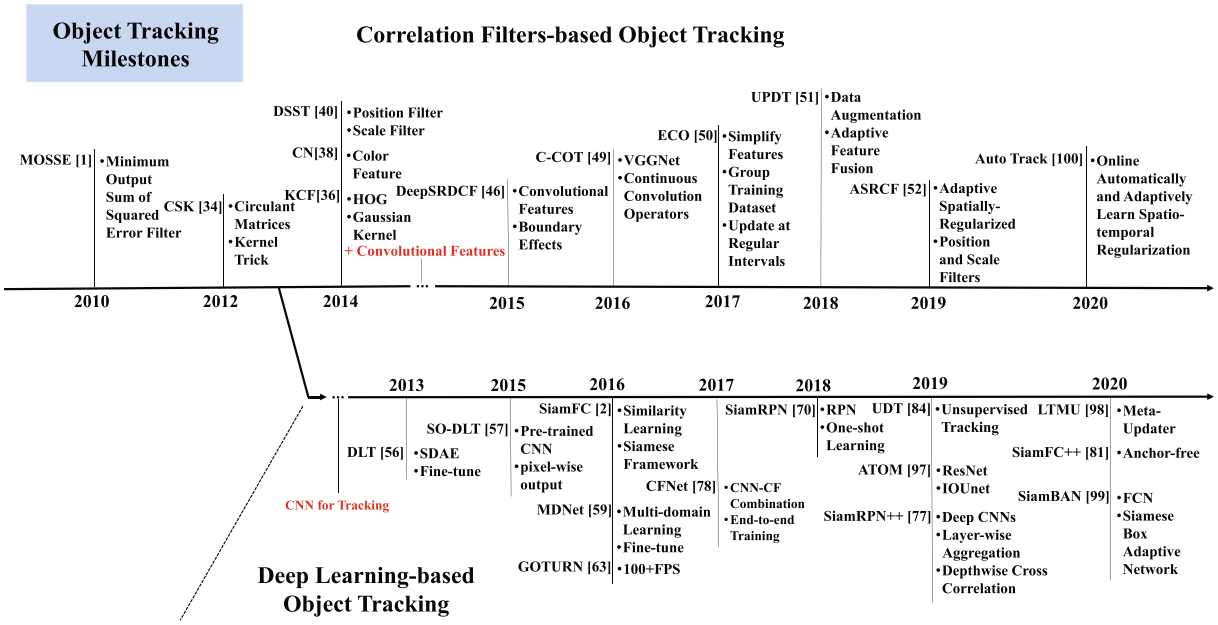


Fig. 2. Object tracking milestones. We summarize the development of single-object tracking in 10 years, and list some typical algorithms, including MOSSE [1], CSK [34], KCF [36], CN [38], DSST [40], DeepSRDCF [46], C–COT [49], ECO [50], SO-DLT [57], UPDT [51], ASRCF [52], DLT [56], SiamFC [2], MDNet [59], GOTURN [63], CFNet [78], SiamRPN [70], SiamRPN++ [77], SiamFC++ [81], UDT [84], ATOM [97], LTMU [98], SiamBAN [99], AutoTrack [100].

correlation filters improves speed and accuracy of trackers, so a considerable amount of correlation filters-based algorithms have been proposed and perform excellently. Deep learning has received widespread attention due to its powerful capabilities in images. Therefore, deep learning-based methods are proposed, and gradually achieved a balance between speed and accuracy, which became the focus of further research.

1.1. Difference from previous survey papers

We analyze surveys and reviews about object tracking in recent years. In [101], the state-of-the-art Video Object Segmentation and Tracking (VOST) methods were analysed and summarized, VOST methods divide tracking problems into video object segmentation and object tracking. [102] comprehensively investigated deep learning-based multiple object trackers on single-camera videos. [103] focused on summarizing multicue object tracking, sensors used in these trackers include not only vision but also thermal, radar and so on. [104] classified and analyzed object tracking algorithms through a large number of experiments and [105] use module-based architecture to summarize the development in 2D appearance models of visual object tracking. Compared with [101–103], our work focuses on summarizing the single-stage visual object tracking method. The tracking method using vision and a complete tracking model is a research hotspot. Although [104,105] have conducted in-depth investigations on many tracking algorithms, methods based on deep learning is rarely mentioned. Our work focuses on analyzing correlation filters-based trackers and deep learning-based trackers which are currently the most efficient trackers and have extremely high research value. Besides, we compare evaluation results of trackers on influential public datasets to get a more comprehensive summary.

1.2. Contribution

The contribution of this paper can be summarized as follows:

- This paper summarizes and analyzes object tracking algorithms based on correlation filters and deep learning in the past 10 years, and proposes a new taxonomy for algorithms involved.
- This paper comparing the performance of representative algorithms on OTB2015 [17], VOT2016 [95] and LaSOT [96]. And we analyze different types of algorithms in three aspects: speed, accuracy and robustness.
- Based on the analysis of the current algorithms, this paper discusses the potential development directions of object tracking.

The remaining parts of the paper consists of 6 chapters. In Section 2, we introduce the background and related work, including basic theory of correlation filters and deep learning. In Sections 3 and 4, as shown in Fig. 3, we introduce typical algorithms according to category, and analyze their motivations, advantages and disadvantages by summarizing each type of method. In Section 5, we analyze the evaluation results of existing algorithms in tracking datasets, and obtain the comprehensive performance of different methods. In Sections 5 and 6, we discuss the direction worthy of further research and summarize the whole paper.

2. Background and related work

2.1. Classic object tracking

The main representative methods of classic object tracking include optical flow methods, filters methods and kernel-based methods. The optical flow methods calculate the movement trend of the object by finding the corresponding pixel position between two frames, thereby obtaining tracking information. Horn et al. [86] combined a two-dimensional velocity field with grayscale features to establish an optical flow constraint equation, and proposed a calculation method for optical flow in 1981. Lucas and Kanade (L-K) [87] assumed that the optical flow is constant in the neighborhood of one pixel, and solve basic optical flow equations for each pixel in the neighborhood. The optical flow obtained by combining multiple pixels is more accurate. In order to solve the problem that the optical flow method cannot deal with fast-moving targets, Jean-Yves Bouguet [88] proposed a pyramid optical flow algorithm, which scales the image into a pyramid shape and then solves it layer by layer to obtain the original image optical flow.

The object tracking task processes continuous frames, and whether the relationship between frames is fully utilized will have a great impact on the tracking results. Welch et al. [89] proposed Kalman filters to predict the current status through the previous status and then modify the prediction result based on the observation information. The extended Kalman filters [90] perform a first-order Taylor expansion on the nonlinear model to obtain an approximate linear model, and then uses the Kalman filters to estimate status. Julier et al. [91] proposed unscented Kalman filters to solve the problem of large calculation and linear error in extended Kalman filters. Nummiaro et al. [92] proposed particle filters to cope with the situation of no determinate model.

Comaniciu [93] introduced the MeanShift to the object tracking tasks and proposed a kernel-based tracking algorithm. The algorithm extracts the color histogram of the initial object and the current candidate region, and iteratively adjusts the MeanShift vector to point to the candidate region which has the highest similarity with the original object. Bradski [94] added a scale change mechanism to the algorithm based on MeanShift, at the same time, the application of HSV color histograms and template update greatly improves performance.

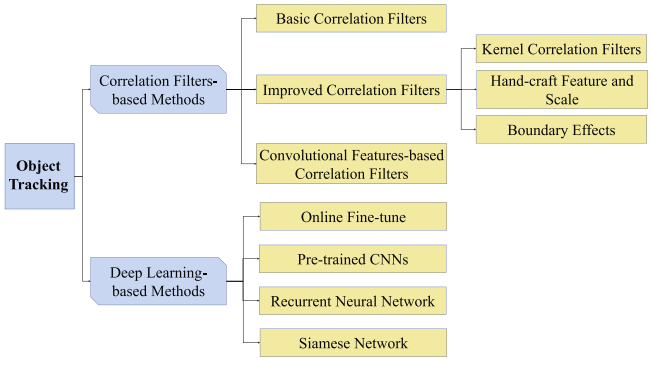


Fig. 3. We classifies object tracking algorithms into correlation filters-based methods and deep learning-based methods. According to the different technical characteristics, we classify different methods in detail.

2.2. Correlation filters

Correlation filters were early applied to signal processing. The main principle is to calculate the correlation between two signals. The correlation between signals u and v is as follows:

$$(u \otimes v)(\tau) = \int_{-\infty}^{+\infty} u^*(t)\, v(t+\tau) dt, \tag{1}$$

where $*$ denotes the complex conjugate. In the case of discrete representation, the correlation can be shown as:

$$(u \otimes v)(n) = \sum_{-\infty}^{\infty} u^*[m]\, v(m+n). \tag{2}$$

To adapt to different tasks, many innovative correlation filters methods have been proposed, such as Synthetic Discriminant Functions (SDF) [18] and Minimum Average Correlation Energy (MACE) [19]. Due to the limit of training samples, the peak height of these filters are consistent. Mahalanobis et al. [20] improved MACE by eliminating hard constrains, and proposed UMACE. Bolme et al. [21] proposed Average of Synthetic Exact Filters (ASEF), which specified the filtered output of the entire image instead of a peak during training. This method is more accurate than the previous methods in object localization.

The earliest tracking algorithm using correlation filters is MOSSE [1], after which KCF [35], DSST [40], ECO [50] etc. received widespread attention.

2.3. Deep learning

Deep learning contributes to many computer vision-related tasks. In object tracking tasks, Deep Neural Networks (DNN) have made outstanding contributions in feature extraction and position prediction. The DNNs currently used for object tracking are Autoencoders (AE), Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN).

The Autoencoder whose input is consistent with the expected output is composed of encoder and decoder, and the network is trained through the loss of input and output [22,23]. Convolutional Neural Network (CNN) contains convolutional layers, pooling layers and fully connected layers. LeNet5 [24] defines the basic structure of CNN. Krizhevsky et al. [25] built a Deep Convolutional Neural Network named AlexNet, which achieved excellent results in the visual recognition challenge and promoted the development of deep learning. Subsequently, ZFNet [26], VGGNet [27], GoogLeNet [28], ResNet [29] and DenseNet [30] continued to innovate in

terms of convolutional kernel, network structure, and network layers, which continuously improved the performance of networks. The object tracking algorithms are applied to video sequences, but CNN cannot utilize the information between frames, which causes a waste of information. Therefore, Recurrent Neural Network (RNN) is applied to object tracking because it is more suitable for sequence tasks. Hochreiter et al. [31] proposed LSTM to suppress the problem of gradient disappearance in RNN, and then Graves improved the network to make it widely used. In addition, there are many variants of LSTM, such as GRU [32] and DLSTM [33].

DLT [56] is an early well-known object tracking algorithm based on deep learning, and other representative algorithms include MDNet [59], SiamFC [2], SiamRPN [70].

3. Correlation filters-based methods

In this section, the development process of correlation filters-based single-object tracking methods is summarized. We classify the algorithms and prove the development trend by introducing some typical algorithms.

3.1. Basic correlation filters tracker

Bolme et al. [1] adopted correlation filters to accomplish object tracking tasks (MOSSE). Assuming the filter is h, the input image is c, and the correlation map is g, then,

$$g = c \otimes h, \tag{3}$$

where $\otimes$ denotes circular correlation.

In order to speed up the calculation, correlation operations are transformed into the frequency domain through Fast Fourier Transform (FFT). The Fourier transform of f is $C = \mathscr{F}(c)$ and the Fourier transform of h is $H = \mathscr{F}(h)$, and then correlation map in the Fourier domain is:

$$G = C \odot H^*, \tag{4}$$

where $\odot$ represents element-wise multiplication.

The H^* in Eq. (4) is the filter to be found. According to the principle of minimum output sum of squared error filter (MOSSE) [1], the problem is described as:

$$\min_{H^*} \sum_i |C_i \odot H^* - G_{Ti}|^2, \tag{5}$$

where C is the Fourier transform of input images c and G_T is the Fourier transform of label g_T.

Solving Eq. (5) using partial derivative equal to 0, and the result is given as follows:

$$H^* = \frac{\sum_i G_{Ti} \odot C_i^*}{\sum_i C_i \odot C_i^*}. \tag{6}$$

The filter H^* is used to perform the filtering operation on the next frame, and the maximum response obtained is the target position.

Using correlation filters technology to complete object tracking tasks is the beginning of efficient tracking. MOSSE has fully explained the training and tracking process of the trackers based on correlation filters.Benefiting from the fast calculation in the frequency domain, the speed of the MOOSE tracker reached to 615 FPS. However, there is much room for improvement in both correlation filters and feature extraction, which provides the basis for a series of improvements.

3.2. Improved correlation filters trackers

3.2.1. Kernel correlation filters

The MOSSE is proposed as the beginning of correlation filters for object tracking. Kernel correlation filter is a major improvement in the correlation filters-based algorithms. Henriques et al. [34] proposed the kernel correlation filters (CSK). And then they utilized the HOG feature [35] instead of gray feature to improve the performance of CSK, and named it KCF [36]. The KCF algorithm was proposed in 2015, which is another milestone in the development of object tracking. In KCF, the authors consider the tracking tasks as classification problems of foreground and background, then described the tracking tasks as follows:

$$\min_w \sum_i (l(x_i) - g_{Ti})^2 + \lambda \|w\|, \tag{7}$$

where $l(\cdot)$ denotes linear regression function, and $l(x_i) = w^T x_i$, λ denotes regularization parameter.

Solving Eq. (7) by least square method, the result is given as follows:

$$w = \left(X^H X + \lambda I\right)^{-1} X^H g_T, \tag{8}$$

where $X = [x_1, x_2, x_3, \ldots, x_n]^T$, and X^H denotes Hermitian transpose.

The algorithm uses dense sampling, and the inverse operation is included in Eq. (8), the amount of calculation will be very large. Therefore, the author introduced a circular matrix, let $X = F\mathrm{diag}(\hat{x})F^H$, where F denotes discrete fourier matrix, and the result is expressed as follows:

$$\hat{w} = \frac{\hat{x}^* \odot \hat{g_T}}{\hat{x} \odot \hat{x}^* + \lambda}. \tag{9}$$

Besides, the author was inspired by the SVM [37] and transform the problem from linear space to non-linear space. The input x can be denoted as $\varphi(x)$ in non-linear feature-space, and the filter w can be expressed as follows:

$$w = \sum_i \alpha_i \varphi(x_i), \tag{10}$$

In Eq. (10), α is the coefficient vector, which is given by:

$$\alpha = (K + \lambda I)^{-1} g_T, \tag{11}$$

where K denotes kernel matrix. Choosing a suitable kernel can make the matrix K circulate, so that Eq. (11) can be simplified by diagonalization property and α can be solved quickly. The suitable kernels include dot-product kernel, polynomial kernel, Radial Basis Function (RBF) kernel and Gaussian kernel. And then from Eq. (10), the best filter w can be obtained by optimizing α.

Finally, for a new frame of image z, filter w is used to predict the target position:

$$l(z) = w^H \varphi(z) = \sum_i \alpha_i k(z, x_i). \tag{12}$$

where $k(z, x_i) = \varphi^H(z)\varphi(x_i)$, and can be computed by kernel function.

The number of training samples directly affects the performance of filters. Training filters with dense samples generated by the circulant matrices can solve the problem of lacking training samples. At the same time, converting the training samples into the frequency domain and using diagonalization to calculate can increase the training speed. The kernel technique can be used to map the data from low dimensions to high dimensions, thereby converting nonlinear problems into linear problems. On this basis, ridge regression can be used to solve filters. These innovations have greatly improved the performance of trackers while ensuring speed.

3.2.2. Improvement of scales and hand-crafted features

Although MOSSE and CSK have significant innovations, the shortcoming of single feature and constant scale limit further improvement. Danelljan et al. [38] added color features to CSK which improve the tracking accuracy. Li et al. [39] adopted the fusion of HOG feature, color feature and grayscale feature, and predicted bounding boxes at 7 scales, then selected the best bounding box. Danelljan et al. [40] used position filters and scale filters to accurately predict the object location. Bertinetto et al. [41] improved tracking performance by fusing HOG feature and color feature [38]. Dong et al. [42] proposed a framework with circulant structure kernels to solve occlusion, and combined CN and HOG features to improve proformance. Ma et al. [43] proposed a discriminant regression model to evaluate the confidence of tracking results, avoiding that filter update is interfered by occlusion in the process of long-term tracking.

3.2.3. Improvement of eliminating boundary effects

In the process of solving the filter, the application of Fast Fourier Transform (FFT) greatly improves the operation speed. However, the cyclic splicing operation causes the image signal to be discontinuous, which produces boundary effects. To solve this problem, Galoogahi et al. [44] marked the target position in the whole picture by a binary matrix to suppress the boundary effects. And in subsequent work, the HOG feature was used to further improve performance. Danelljan et al. [45] expanded the search range and added regular terms to penalize samples that are far away from the center, and the capability of the classifier was improved (SRDCF). Considering that the object often encounters problems such as occlusion during the tracking process, which will pollute the training dataset, Danelljan et al. [47] added sample weight parameters and regularization to the objective function, and utilized an adaptive learning rate to improve the model performance in complex environments.

3.3. Convolutional features-based correlation filters trackers

Manual features are difficult to meet the tracking requirements in complex environments, which limits the further improvement of tracking accuracy. With the superiority of deep learning in feature extraction, it has become an inevitable trend to adopt convolutional features in correlation filters framework.

Danelljan et al. [46] introduced convolutional features to further strengthen the SRDCF algorithm. Ma et al. [48] replaced HOG features with deep convolution features and fused the confidence maps obtained by filtering three layers of features respectively. Danelljan et al. [49] proposed C–COT algorithm, which combines the output features of the two convolutional layers and the original color image, and then uses continuous convolution filters to generate a response map for position prediction. Although the accuracy and robustness of C–COT are relatively good, the complexity of the model results in a slow speed, and cannot achieve real-time tracking. Danelljan et al. [50] continued to improve C–COT from three aspects: simplifying the features, grouping the training set, reducing the frequency of template updates. Through improvement, efficiency is improved. By analyzing the role of deep features and shallow features in object tracking. Bhat et al. [51] believed that shallow features are suitable for positioning and deep features contribute to the robustness of the model. Therefore, the authors improve the tracking accuracy by adaptively fusing features. Dai et al. [52] proposed a tracking method that combines deep convolution feature and adaptive spatially-regularized correlation filters. This method uses two filters to predict the position and scale of targets, which ensures accuracy and speed. Lu et al. [53] handle channel redundancy of convolution features by channel regularization, which improves the speed and accuracy of the convolutional features-based correlation filters tracker.

3.4. Discussion

Correlation filters-based object tracking is the beginning of efficient tracking. The ultra-high operating speed provides the possibility for embedded devices to run object tracking algorithms. Correlation filters with kernels improve tracking accuracy while simplifying the algorithm, which improves the theory of correlation filters-based tracking. The improvement of boundary effect, the improvement of the bounding box scale, and the rich manual features and convolution features improve the accuracy of trackers from different angles. However, most of the current high-precision correlation filters-based trackers adopt deep convolutional neural network models pre-trained by large-scale classification datasets, which leads to slow calculation and difficulty in real-time tracking. In addition, non-end-to-end training also makes it impossible to achieve optimal performance by coordinating each part of trackers.

4. Deep learning-based methods

Correlation filters-based tracking methods have better performance than traditional methods, which is more suitable for practical applications. However, manual features are difficult to cope with the changeable environment, and the superposition of various features greatly affects the speed of the algorithms. Therefore, deep learning has been gradually emphasized. In this section, we summarize deep learning-based single-object tracking methods.

4.1. Algorithms based on online fine-tune

The original idea of deep learning-based algorithms is using neural networks to extract features and then classifying the object and background. To train a suitable model for the tracking, a method of training network offline and fine-tuning model online was proposed.

DLT [56] is one of the early single object tracking algorithms using deep learning. As shown in Fig. 4, Wang et al. [56] adopted large-scale datasets to train stacked denoising autoencoder (SDAE) offline. During the tracking phase, the labeled first frame is used to fine-tune the classification network, after which the candidate area obtained through particle filtering is input into the network, and

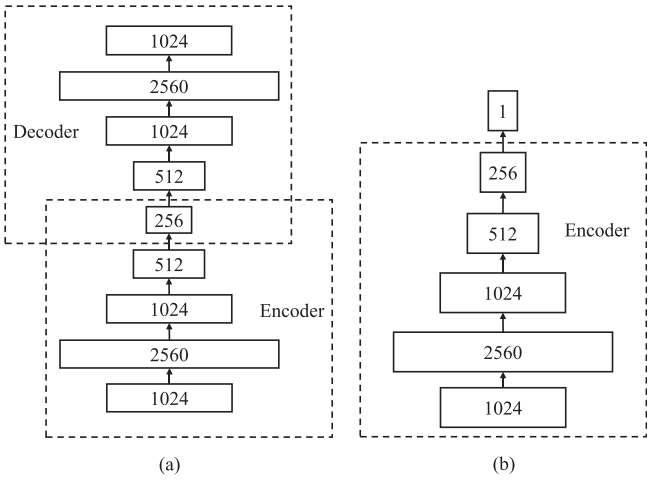


Fig. 4. The key components of DLT algorithm (the numbers in the boxes represent the number of channels in each network layer). (a) stacked denoising autoencoder; (b) The online tracking network.

the target position can be predicted. Aiming at the problem of lacking training datasets and feature extraction ability in DLT, Wang et al. [57] applied the AlexNet [25] trained offline on ImageNet [58] to obtain stronger capabilities of characterization and classification.

Considering the particularity of tracking tasks, adopting tracking sequences as datasets to pre-train model plays a key role in improving performance. Nam et al. [59] proposed a multi-domain learning model based on CNN (MDNet). The model contains shared layers and domain-specific layers. In the training stage, K videos are trained at the same time to get the commonality of objects, and the shared layers obtain the ability to extract common features. In the tracking stage, the weights of fc4-fc6 are fine-tuned by the first frame. Starting from the second frame, the previous frame is used to generate the object candidate samples, and an optimal object state is adjusted through bounding box regression. The structure of MDNet is shown in Fig. 5. Wang et al. [61] analyzed the effect of different feature layers. Based on the above, a feature screening network is constructed to select the channel which is most relevant to the object. Next, a general network (GNet) that obtains category information and a specific network (SNet) that distinguishes distractors are created to generate two heatmaps. When tracking, the first frame is used to initialize SNet and GNet to regress to the heat map for the target. Du et al. [62] fuses convolutional layers of multiple scales by residual structure, and proposes a shrinkage loss function to reduce the contribution of negative samples. And in the tracking phase, the first frame is used to train the regression part of model.

Since the object tracking task provides the object information in the first frame, using the first frame to initialize and fine-tune the model can help the model focus more on the current tracking task. However, online fine-tuning causes trackers to run slowly, which is difficult to meet real-time requirements. Therefore, the relationship between accuracy and speed needs to be balanced.

4.2. Algorithms based on pre-trained CNNs

Each tracking task has an annotation only on the first frame, which results in lacking training dataset. Besides, online training is inefficient. To solve these problems, CNNs are trained offline by large-scale classification datasets to extract general features, which are directly used in object tracking tasks.

Hong et al. [60] proposed a tracker composed of CNN and SVM. The main idea is training CNN offline using the ImageNet dataset, and then SVM is used to classify objects and background. Considering the requirements of real-time tracking, David et al. [63] directly regressed object location by training a neural network, which is the first deep learning-based algorithm to reach 100 fps.

The premise of most object tracking algorithms deal with the target which has small changes between two frames, but this assumption is often unreasonable, which makes it difficult to obtain good performance in situations such as occlusion, changing light, sudden movement and deformation. As shown in Fig. 6, Nam et al. [64] connected multiple pre-trained CNNs through a tree structure to form an appearance model, and the final candidate region score can be obtained by weighting the scores calculated by each CNN. When tracking online, the model adds a new CNN node every ten frames and deletes the first node. The model update method can ensure a better response to changes in complex space–time contexts.

Deep convolutional neural networks trained on large-scale classification datasets have excellent feature extraction capabilities, which is the main reason why pre-trained networks can be applied to object tracking tasks. However, the pre-trained model pays more attention to extracting semantic features of each image, while object tracking tasks pay more attention to predicting the location of the labeled object in the video sequence. Therefore, it is also a necessary process to adjust models by tracking datasets.

4.3. Algorithms based on recurrent neural network

The sequence information extracted from the video sequence has constructive guiding significance for the prediction of target motion. However, traditional object tracking algorithms can only extract the appearance features of objects by CNN. Therefore, the use of sequence information is meaningful for object tracking tasks. Due to the advantages of recurrent neural networks (RNN) in sequence-related tasks, it was utilized in object tracking.

Cui et al. [65] divided the candidate region into grids, and extract HOG features from each grid. Then RNN is used to obtain the confidence map which can weight filter to get the final target position. Ning et al. [66] utilized the structure of the YOLO algorithm [67] to roughly estimate the target position and size, and then information is inputted into the Long Short-Term Memory (LSTM) [31] to predict bounding box. The algorithm exerts the prediction ability of the RNN structure based on the existing information, but there is much room for improvement in the application of the RNN structure.

Since CNN-based models are mainly used for inter-class discrimination, objects between classes are easily confused, which results in failed tracking. Fan et al. [68] proposed a model with CNN and RNN based on MDNet, in which CNN mainly discriminates the difference between the target and background, while RNN mainly divides the target from similar objects. The features extracted by CNN and RNN are fused to enhance the discriminative ability of the tracker.

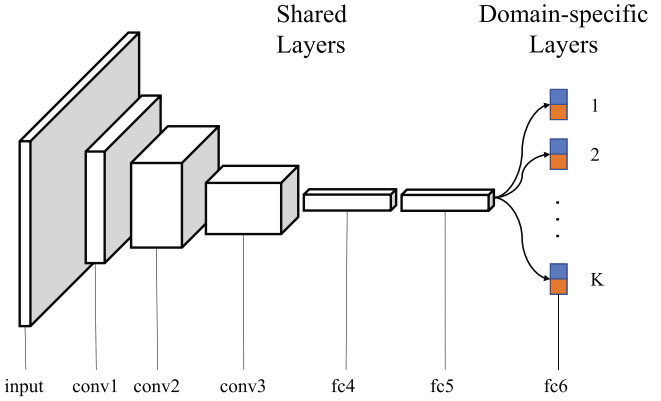


Fig. 5. Schematic diagram of MDNet algorithm.

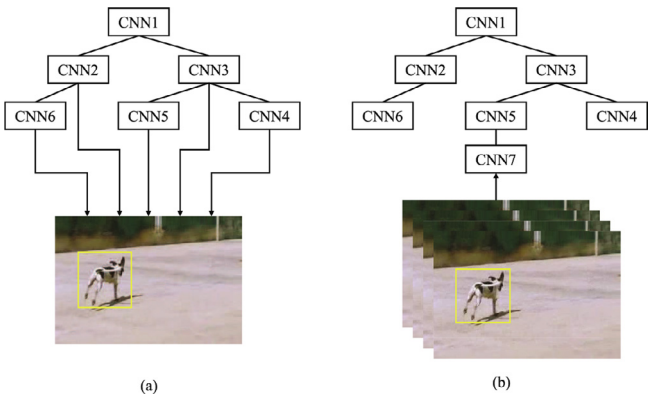


Fig. 6. Schematic diagram of TCNN algorithm.

Unlike ordinary computer vision tasks, object tracking is closely related to sequential information. Considering that RNN makes full use of the sequential information, it is suitable for object tracking tasks. The RNN-based object tracking algorithms can synthesize the information of the previous frames in the video sequence to predict position of the object in current frame. However, due to the difficulty of extracting sequential information from images, current trackers have not achieved excellent results. Although these algorithms have a clear idea, how to use sequential information between images for tracking still needs to be explored.

4.4. Algorithms based on siamese network

Most deep learning-based methods obtain the ability to extract features through offline training. However, if the objects are not known in advance, networks must be trained online for the current task, which seriously reduces the speed of trackers.

Tao et al. [69] proposed to solve the tracking problem by similar learning. In SINT algorithm [69], the model is divided into two identical branches, and the network is trained offline. The labeled box and multiple candidate boxes are entered into two branches respectively, and the matching score of each candidate box and the bounding box is obtained. Next, the candidate box with the highest score is selected. Based on SINT algorithm, SiamFC [2] which uses two completely identical network branches was proposed and trained offline. In the tracking stage, the sub-window with the labeled object is input to one branch as a template, and the current frame is input to another branch. By performing cross-correlation on two branches, the bounding box with the highest score is the optimal choice. The structure of SiamFC is shown in Fig. 7. To improve the speed and accuracy of trackers, Li et al. [70] proposed to predict the target position by region proposal network (RPN) [71]. The entire structure consists of siamese networks and RPN, and the model is trained end-to-end. In the tracking stage, the information of the bounding box containing the object is directly regressed. Dong et al. [72] designed compact latent network, which can quickly learn sequence-specific information from the first frame. Shen et al. [73] introduce the attention mechanism to highlight the features of the key parts in object, and propose a method of fusing multi-scale response maps to improve the accuracy of the tracker. Dong et al. [75,74] improves the current deep reinforcement learning method for learning the hyperparameters of the tracker, thereby improving the accuracy of the tracker.

Siamese networks have attracted great attention due to its outstanding performance. However, most of the siamese-based methods only differentiate targets from non-semantic background, which is susceptible to interference from similar objects. Zhu et al. [76] focused on designing siamese networks that can identify interference, and utilize local-to-global search strategy to re-detect the target once the tracking fails.

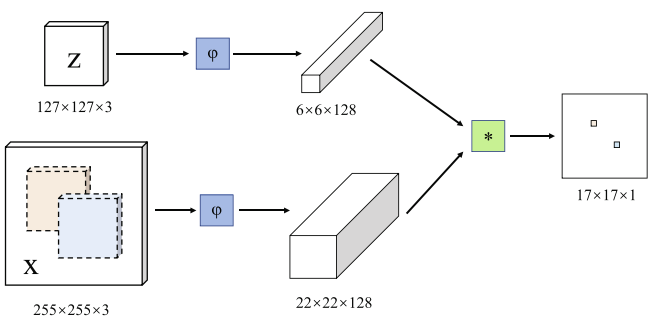


Fig. 7. Schematic diagram of SiameseFC algorithm.

After SiamFC was proposed, there were many tracking algorithms based on siamese networks, but most networks are composed of few layers. Directly using a pre-trained deep network will cause the accuracy to decrease. Li et al. [77] adopted samples with the target shifted near the center point during the training process which partly eliminates the impact of destroying the strict translation invariance, so that deep networks can be applied to tracking algorithms. Besides, the algorithm adopts multi-layer feature fusion and deep cross-correlation to improve performance.

In the process of object tracking, object deformation and changing light are often encountered. If the object labeled at the beginning is used as a template, it may lead to bad feature matching once the environment changes. Valmadre et al. [78] proposed to add a correlation filters layer to the template branch of the siamese structure. Through the online update of the filter, this model realizes the update of the template. In order to enable the correlation filters-based methods to learn deep convolution features through end-to-end training and give full play to the advantages of correlation filters, Wang et al. [79] treated correlation filters as a special layer in siamese framework, and responded to environmental changes by constantly updating filters. The algorithm cascades DCF modules with siamese networks, and the parameters of DCF and CNN are trained end-to-end. Although DCFNet [79] solved the problem of template update under the siamese network framework, it underutilized inter-frame information. Zhu et al. [80] proposed to use the temporal information extracted by optical flow to warp the features of 5 frames into the current frame, and utilized the space–time attention network to adjustment template features. Then the siamese framework is used for tracking.

The prediction of object positions and bounding boxes is the key to estimating object state. A series of trackers based on RPN generate a large number of anchors to predict object positions and bounding boxes. Although these methods have high accuracy in most cases, they use too much a priori knowledge when setting anchors, which causes the robustness of trackers are limited when object deformation is large. Therefore, anchor free-based trackers are proposed to solve these problems. Xu et al. [81] proposed a new algorithm based on the siamese network, in which one branch predicts the confidence of the object position by predicting each pixel in the feature map, and the other branch regresses the distance between samples and four edges of the ground truth [82]. Gao et al. [83] proposed siamese attentional keypoint network and obtain bounding boxes by predicting the coordinates of the upper left corner, center, and lower right corner of the target.

The main idea of the siamese network is to search for the position that best matches the object in the current frame. The trackers can quickly track any object without fine-tuning online according to each tracking task. At the same time, the end-to-end trained model can optimize the tracker performance to the maximum. Using this strategy ensures both accuracy and speed. The templates of trackers based on the siamese network cannot be updated which limits tracking accuracy in complex environment. Therefore, some trackers try to solve the problem by improving the model, which further improves the siamese network. In addition, pyramid-based, anchor-based, and anchor-free based methods have been successively proposed to predict object positions and bounding boxes, which continuously improves the robustness and accuracy of trackers based on the siamese network.

4.5. Discussion

Although the accuracy of deep learning-based object tracking methods is relatively high, the deep CNN and online fine-tuning result in slow calculation speed. Siamese network-based methods balance the relationship between accuracy and speed. Some scholars [78–80] proposed to design an end-to-end training model that

combines the advantages of correlation filters and deep learning, which inspired the innovation of tracking algorithms.

5. Results analysis

5.1. Datasets and indicators

To further compare the performance of different trackers in the latest 10 years, we evaluated some representative algorithms on OTB2015 [17], VOT2016 [95] and LaSOT [96]. Table 1 shows the evaluation of trackers, where *CF* denotes correlation filters and *DL* denotes deep learning.

OTB2015 is a standard dataset used for tracking testing, including 100 fully annotated sequences, which are divided into 11 attributes according to content. Success plots and precision plots are common evaluation metrics in the OTB benchmark. Success plots display the overlap scores which extend a given threshold, precision plots show the percentage whose distance between the center point of the bounding box and the ground-truth is less than the given threshold. In tables, *AUC* denotes the area under curve (AUC) of success plots, and *P* denotes precision score at a threshold of 20 pixels.

VOT2016 contains 60 video sequences, and the dataset is relabeled by automatically generating bounding boxes, making the labeling more reasonable. Evaluation metrics include A, R and *EAO*. *A* represents the tracking accuracy, which is the average of overlap scores between bounding box and groundtruth. *R* represents the robustness of tracking, and it is considered to be an metric that has the least correlation with *A*. Specifically, *R* is the rate of tracking failure in each video sequence. *EAO* represents the expected value of overlap in an image sequence without resetting.

LaSOT is a large-scale and high-quality single object tracking dataset. The dataset contains 70 categories, each category contains 20 sequences, and each sequence has an average of 2512 frames. By manually labeling each frame, approximately 3.52 million high-quality bounding box annotations are generated. The evaluation metrics are precision, normalized precision and success. Success and precision have the same meaning as the metrics in OTB datasets, and they are denoted as *AUC* and *P*. Considering the impact of object scale and image resolution on the precision metric, the normalized precision metric is proposed, and it is denoted as P_{norm}. P_{norm} is AUC of normalized precision plots.

5.2. Analysis

The precision and speed metrics reveal that the basic correlation filters-based methods are fast but low accuracy. The convolutional features greatly improve the accuracy of algorithms at the cost of slower speed. And the table also shows that the speed and accuracy of algorithms could not coexist in early deep learning-based methods. The online fine-tuning slow down the speed while improving accuracy. However, with the continuous improvement of deep learning-based algorithms, the single-object tracking methods become not only much faster but also much more accurate. The methods based on siamese framework attract broad attention due to its excellent comprehensive performance.

The robustness metrics shows that the correlation filters trackers based on convolution features is more robust than the purely correlation filters trackers. And the deep learning-based trackers, especially the siamese network-based trackers, are extremely robust. These results reflect the improvement of deep learning on trackers. In addition, the combination of deep learning and correlation filters is a direction worthy of further research.

6. Discussion on development trends

At present, deep learning-based object tracking algorithms have achieved excellent performance, but there is still much room for improvement. We believe that single-object tracking will be further developed in the following aspects:

1) Feature Representation for Object Tracking. For trackers, feature representation is directly related to tracking accuracy and speed. Early correlation filters-based algorithms use gray feature and HOG feature, and the improvement of trackers accuracy is limited by features. As the convolutional neural network improves feature quality, the performance of trackers is also greatly improved. In [77], the deep convolutional neural network is successfully applied to the tracker and further improved the tracking accuracy. In [106,107], multiple loss has been used to learn better feature representation. Therefore, for excellent trackers, the methods of feature extraction are crucial. In tracking tasks, the connection between different frames of the same video sequence is

Table 1
Performance of partial trackers.

Methods	References	OTB2015		VOT2016			LaSOT			Main Tech.	FPS	Real Time
		AUC	P	EAO	A	R	AUC	P	$\mathbf{P_{norm}}$			
MOSSE [1]	[D. S. Bolme et al. CVPR2010]	0.310	0.414	–	–	–	–	–	–	CF	615(CPU)	Y
CN [38]	[M. Danelljan et al. CVPR2014]	0.422	0.594	–	–	–	0.186	0.158	-	CF	152(CPU)	Y
KCF [36]	[J. F. Henriques et al. TPAMI2015]	0.477	0.696	0.192	0.489	0.569	0.178	0.166	0.190	CF	172(CPU)	Y
DSST [40]	[M. Danelljan et al. BMVC2014]	0.513	0.680	0.181	0.533	0.704	0.207	0.189	0.213	CF	24(CPU)	Y
SAMF [39]	[Y. Li et al. ECCVW2014]	0.541	0.752	0.186	0.507	0.587	0.233	0.203	0.239	CF	7(CPU)	N
Staple [41]	[L. Bertinetto et al. CVPR2016]	0.578	0.784	0.295	0.544	0.378	0.243	0.239	0.278	CF	80(CPU)	Y
SRDCF [45]	[M. Danelljan et al. ICCV2015]	0.598	0.789	0.247	0.535	0.419	0.245	0.219	0.248	CF	4(CPU)	N
DeepSRDCF [46]	[M. Danelljan et al. ICCVW2015]	0.635	0.851	0.276	0.528	0.326	–	–	–	CF + DL	0.3(CPU)	N
CCOT [49]	[M. Danelljan et al. ECCV2016]	0.671	0.898	0.331	0.539	0.238	–	–	–	CF + DL	0.3(CPU)	N
ECO [50]	[M. Danelljan et al. CVPR2017]	0.691	0.910	0.375	0.550	0.200	0.324	0.301	0.338	CF + DL	8(GPU)	N
UPDT [51]	[G. Bhat et al. ECCV2018]	0.713	0.932	0.378	0.532	0.182	–	–	–	CF + DL	–	–
ASRCF [52]	[K. Dai et al. CVPR2019]	0.692	0.922	0.391	0.563	0.187	0.359	0.337	–	CF + DL	28(GPU)	N
MDNet [59]	[H. Nam et al. CVPR2016]	0.678	0.909	0.257	0.541	0.337	0.397	0.373	0.460	DL	1(GPU)	N
TCNN [64]	[H. Nam et al. CVPR2016]	0.654	0.884	0.325	0.554	0.268	–	–	–	DL	1.5(GPU)	N
SINT [69]	[R. Tao et al. CVPR2016]	0.592	0.789	–	–	–	0.314	0.295	0.354	DL	–	–
SiamFC [2]	[L. Bertinetto et al. ECCV2016]	0.582	0.771	0.235	0.530	0.460	0.336	0.339	0.420	DL	86(GPU)	Y
CFNet [78]	[J. Valmadre et al. CVPR2017]	0.568	0.748	–	–	–	0.275	0.259	0.312	DL + CF	75(GPU)	Y
SiamRPN [70]	[B. Li et al. CVPR2018]	0.636	0.850	0.344	0.560	0.260	–	–	–	DL	200(GPU)	Y
DaSiamRPN [76]	[Z. Zhu et al. ECCV2018]	0.658	0.880	0.411	0.610	0.220	0.415	–	0.496	DL	160(GPU)	Y
SiamRPN++ [77]	[B. Li et al. CVPR2019]	0.696	0.914	–	–	–	0.496	–	0.569	DL	35(GPU)	Y

close, so inter-frame information needs to be further utilized to highlight object features. It is necessary to continuously improve the feature extraction methods to fully mine object features.

2) Compressing Models. With the development of target tracking algorithms, using deep neural networks to improve accuracy has been adopted by most trackers. At the same time, the number of trackers with huge models is increasing. Although these trackers have high accuracy, they run slowly and have a large amount of calculation, which requires high equipment performance. However, the trackers are widely applied in embedded devices, limited by the ability of computing, complex networks are difficult to accomplish real-time tracking tasks. It is valuable to compress models without affecting or rarely affecting performance so that trackers can be applied in more scenes. To compress models, [108,109] have applied the method of knowledge distillation. The compression of the tracking model can remove network layers that are of little significance to tracking and focus more on the object to be tracked, thereby greatly improving the efficiency of trackers. Therefore, further research on model compression is beneficial to the development and application of object tracking algorithms.
3) Improving Unsupervised Object Tracking Algorithms. Deep learning-based methods improve the accuracy of trackers. However, a considerable amount of labeled datasets in different scenes must be used to train networks, which require a heavy workload for labeling datasets. Unsupervised methods can use unlabeled datasets to train models, which can reduce the workload of manually annotating datasets and train models with video sequences of different scenes. This is beneficial to promote trackers to various application scenarios. [110] has proposed a complete unsupervised object tracking, and used the unlabeled training datasets to train the model, which greatly expanded the training datasets of the tracker. At present, there are few researches on unsupervised object tracking. In order to reduce the training cost of trackers and expand application scenarios, more efficient trackers need to be proposed by combining advanced unsupervised learning methods and tracking models.

7. Conclusion

In the past 10 years, the performance of trackers has been continuously improved due to the application of correlation filters and deep learning in object tracking tasks. In this paper, these trackers have been divided into two categories by the characteristics of tracking models, namely, correlation filters-based trackers and deep learning-based trackers. And then, we have divided the trackers in detail by analyzing the different improvement directions.

The application of correlation filters greatly improves the speed and accuracy of trackers. However, defects in hand-crafted features can affect the accuracy of the trackers. Considering the importance of feature extraction, deep learning has received wide attention. And the combination of convolution features and correlation filters greatly improves the performance of trackers. Besides, deep learning trackers, such as siamese network-based trackers, have a reasonable structure and excellent performance, which become a basic model for improving trackers. Deep learning improves the features and models of trackers, which makes a significant contribution to object tracking. We have summarized and analyzed excellent trackers in this paper, and hope to inspire the development of object tracking.

CRediT authorship contribution statement

Yucheng Zhang: Investigation. **Tian Wang:** Writing - review & editing, Methodology, Supervision. **Kexin Liu:** Writing - review & editing. **Baochang Zhang:** Writing - review & editing. **Lei Chen:** Visualization.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work is partially supported by the National Natural Science Foundation of China (61972016, 62032016, 61903017, 62003015), Beijing Natural Science Foundation (L191007), Fundamental Research Funds for the Central Universities (YWF-20-BJ-J-612), Open Research Fund of Digital Fujian Environment Monitoring Internet of Things Laboratory Foundation (202004), and in part by China Postdoctoral Science Foundation (2020M670087).

References

[1] D.S. Bolme, J.R. Beveridge, B.A. Draper and Y.M. Lui, Visual object tracking using adaptive correlation filters, in: Proc. 2010 IEEE Conf. Computer Vision and Pattern Recognition, 2010, pp. 2544–2550..
[2] L. Bertinetto, J. Valmadre, J.F. Henriques, A. Vedaldi, P.H.S. Torr, Fully-convolutional siamese networks for object tracking, in: Proc. 2016 European Conf. Computer Vision, 2016, pp. 850–865.
[3] J. Yin, W. Wang, Q. Meng, R. Yang and J. Shen, A unified object motion and affinity model for online multi-object tracking, arXiv preprint arXiv:2003.11291, 2020..
[4] J. Shen, Z. Liang, J. Liu, H. Sun, L. Shao, D. Tao, Multiobject tracking by submodular optimization, IEEE Trans. Cybern. 49 (6) (2019) 1990–2001.
[5] U. Kart, A. Lukezic, M. Kristan, J. Kamarainen and J. Matas, Object tracking by reconstruction with view-specific discriminative correlation filters, arXiv preprint arXiv:1811.10863, 2019..
[6] X. Lu, W. Wang, M. Danelljan, T. Zhou, J. Shen and L.V. Gool, Video object segmentation with episodic graph memory networks, arXiv preprint arXiv:2007.07020, 2020..
[7] X. Lu, W. Wang, J. Shen, Y.W. Tai, D. Crandall and S.C.H. Hoi, Learning video object segmentation from unlabeled videos, arXiv preprint arXiv:2003.05020, 2020..
[8] K. Lee, J. Hwang, On-road pedestrian tracking across multiple driving recorders, IEEE Trans. Multimed. 17 (9) (2015) 1429–1438.
[9] H.X. Yang, L. Shao, F. Zheng, L. Wang, Z. Song, Recent advances and trends in visual tracking: A review, Neurocomputing 74 (18) (2011) 3823–3831.
[10] F. Boninfont, A. Ortiz, G. Oliver, Visual navigation for mobile robots: A survey, J. Intell. Rob. Syst. 53 (3) (2008) 263–296.
[11] A. Brunetti, D. Buongiorno, G.F. Trotta, V. Bevilacqua, Computer vision and deep learning techniques for pedestrian detection and tracking: A survey, Neurocomputing 300 (2018) 17–33.
[12] S. Tang, M. Andriluka, B. Andres, and B. Schiele, Multiple people tracking by lifted multicut and person re-identification, in: Proc. 2017 IEEE Conf. Computer Vision and Pattern Recognition, 2017, pp. 3701–3710..
[13] I. Haritaoglu, D. Harwood, L.S. Davis, W/sup 4/: real-time surveillance of people and their activities, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 809–830.
[14] M.S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, IEEE Trans. Signal Process. 50 (2) (2002) 174–188.
[15] J. Redmon, S.K. Divvala, R. Girshick, You Only Look Once: Unified, Real-Time Object Detection, in: Proc. 2016 IEEE Conf. Computer Vision and Pattern Recognition, 2016, pp. 779–788..
[16] J. Redmon and A. Farhadi, YOLO9000: Better, Faster, Stronger, in: Proc. 2017 IEEE Conf. Computer Vision and Pattern Recognition, 2017, pp. 6517–6525..
[17] Y. Wu, J. Lim, M.H. Yang, Object tracking benchmark, IEEE Trans. Pattern Anal. Mach. Intell. 37 (9) (2015) 1834–1848.
[18] D. Casasent, Unified synthetic discriminant function computational formulation, Appl. Opt. 23 (10) (1984) 1620–1627.
[19] A. Mahalanobis, B.V.K.V. Kumar, D. Casasent, Minimum average correlation energy filters, Appl. Opt. 26 (17) (1987) 3633–3640.
[20] A. Mahalanobis, B.V.K.V. Kumar, S. Song, S.R.F. Sims, J.F. Epperson, Unconstrained correlation filters, Appl. Opt. 33 (17) (1994) 3751–3759.

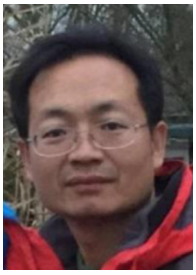
[21] D.S. Bolme, B.A. Draper, and J.R. Beveridge. Average of synthetic exact filters, in: Proc. 2009 IEEE Conf. Computer Vision and Pattern Recognition, 2009, pp. 2105–2112..
[22] Y. Lecun, Y. Bengio, G.E. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.
[23] T. Wang, M. Qiao, Z. Lin, C. Li, H. Snoussi, Z. Liu, C. Choi, Generative Neural Networks for Anomaly Detection in Crowded Scenes, IEEE Trans. Inf. Foren. Secur. 14 (5) (2019) 1390–1399.
[24] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.
[25] A. Krizhevsky, I. Sutskever and G. Hinton, ImageNet classification with deep convolutional neural networks, in: Proc. 2012 IEEE Int. Conf. Neural Information Processing Systems, 2012, pp. 1097–1105..
[26] M.D. Zeiler and R. Fergus, Visualizing and understanding convolutional networks, in: Proc. 2014 European Conf. Computer Vision, 2014, pp. 818–833..
[27] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: Proc. 2014 IEEE Conf. Computer Vision and Pattern Recognition, 2014..
[28] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, Going deeper with convolutions, in: Proc. 2015 IEEE Conf. Computer Vision and Pattern Recognition, 2015, pp. 1–9..
[29] K.M. He, X.Y. Zhang, S.Q. Ren, and J. Sun, Deep residual learning for image recognition, in: Proc. 2016 IEEE Conf. Computer Vision and Pattern Recognition, 2016, pp. 770–778..
[30] G. Huang, Z. Liu, L.V.D. Maaten, and K.Q. Weinberger, Densely connected convolutional networks, in: Proc. 2017 IEEE Conf. Computer Vision and Pattern Recognition, 2017, pp. 2261–2269..
[31] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.
[32] K. Cho, B.V. Merrienboer, D. Bahdanau, and Y. Bengio. On the properties of neural machine translation: Encoder-decoder approaches, arXiv preprint arXiv:1409.1259, 2014..
[33] Y. Shi, K. Yao, L. Tian, L. Tian, and D. Jiang, Deep LSTM based feature mapping for query classification, in: Proc. 2016 Conf. North American Chapter of the Association for Computational Linguistics, 2016, pp. 1501–1511..
[34] J.F. Henriques, R. Caseiro, P. Martins, J. Batista, Exploiting the circulant structure of tracking-by-detection with kernels, in: Proc. 2012 European Conf. Computer Vision, 2012, pp. 702–715.
[35] N. Dalal and B. Triggs, Histograms of oriented gradients for human detection, in: Proc. 2005 IEEE Conf. Computer Vision and Pattern Recognition, 2005, pp. 886–893..
[36] J.F. Henriques, R. Caseiro, P. Martins, High-speed tracking with kernelized correlation filters, IEEE Trans. Pattern Anal. Mach. Intell. 37 (3) (2015) 583–596.
[37] C. Cortes, V. Vapnik, Support-vector networks, Mach. Learn. 20 (3) (1995) 273–297.
[38] M. Danelljan, F.S. Khan and M. Felsberg, Adaptive color attributes for real-time visual tracking, in: Proc. 2014 IEEE Conf. Computer Vision and Pattern Recognition, 2014, pp. 1090–1097..
[39] Y. Li, J. Zhu, A scale adaptive kernel correlation filter tracker with feature integration, in: Proc. 2014 European Conf. Computer Vision Workshops, 2014, pp. 254–265.
[40] M. Danelljan, G. Hager, F.S. Khan, M. Felsberg, Accurate scale estimation for robust visual tracking, in: Proc. 2014 British Machine Vision Conference, 2014.
[41] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik and P.H.S. Torr, Staple: Complementary learners for real-time tracking, in: Proc. 2016 IEEE Conf. Computer Vision and Pattern Recognition, 2016, pp. 1401–1409..
[42] X. Dong, J. Shen, D. Yu, W. Wang, J. Liu, H. Huang, Occlusion-aware real-time object tracking, IEEE Trans. Multi. 19 (4) (2017) 763–771.
[43] C. Ma, X. Yang, Chongyang Zhang and M. Yang, Long-term correlation tracking, in: Proc. 2015 IEEE Conf. Computer Vision and Pattern Recognition, 2015, pp. 5388–5396..
[44] H.K. Galoogahi, T. Sim and S. Lucey, Correlation filters with limited boundaries, in: Proc. 2015 IEEE Conf. Computer Vision and Pattern Recognition, 2015, pp. 4630–4638..
[45] M. Danelljan, G. Hager, F.S. Khan and M. Felsberg, Learning spatially regularized correlation filters for visual tracking, in: Proc. 2015 IEEE Int. Conf. Computer Vision, 2015, pp. 4310–4318..
[46] M. Danelljan, G. Hager and F.S. Khan, Convolutional features for correlation filter based visual tracking, in: Proc. 2015 IEEE Int. Conf. Computer Vision Workshops, 2015, pp. 58–66..
[47] M. Danelljan, G. Hger, F.S. Khan, M. Felsberg, Adaptive decontamination of the training set: A unified formulation for discriminative visual tracking, in: Proc. 2016 IEEE Conf. Computer Vision and Pattern Recognition, 2016.
[48] C. Ma, J. Huang, X. Yang and M. Yang, Hierarchical convolutional features for visual tracking, in: Proc. 2015 IEEE Int. Conf. Computer Vision, 2015, pp. 3074–3082..
[49] M. Danelljan, A. Robinson, F.S. Khan, M. Felsberg, Beyond correlation filters: Learning continuous convolution operators for visual tracking, in: Proc. 2016 European Conf. Computer Vision, 2016, pp. 472–488.
[50] M. Danelljan, G. Bhat, F.S. Khan and M. Felsberg, ECO: Efficient convolution operators for tracking, in: Proc. 2017 IEEE Conf. Computer Vision and Pattern Recognition, 2017, pp. 6931–6939..
[51] G. Bhat, J. Johnander, M. Danelljan, F.S. Khan, and M. Felsberg, Unveiling the power of deep tracking, in: Proc. 2018 European Conf. Computer Vision, Munich, Germany, 2018, pp. 493–509..
[52] K. Dai, D. Wang, H. Lu, C. Sun, and J. Li, Visual tracking via adaptive spatially-regularized correlation filters, in: Proc. 2019 IEEE Conf. Computer Vision and Pattern Recognition, 2019, pp. 4670–4679..
[53] X. Lu, C. Ma, B. Ni and X. Yang, Adaptive region proposal with channel regularization for robust object tracking, IEEE Trans. Circuits Syst. Video Technol. https://doi.org/10.1109/TCSVT.2019.2944654..
[54] F. Tang, Q. Ling, Spatial-aware correlation filters with adaptive weight maps for visual tracking, Neurocomputing 358 (2019) 369–384.
[55] X. Lu, J. Li, Z. He, W. Wang, H. Wang, Distracter-aware tracking via correlation filter, Neurocomputing 348 (2019) 134–144.
[56] N.Y. Wang and D.Y. Yeung, Learning a deep compact image representation for visual tracking, in: Proc. 2013 IEEE Int. Conf. Neural Information Processing Systems, 2013. pp. 809–817..
[57] N.Y. Wang, S.Y. Li, A. Gupta and D.Y. Yeung, Transferring rich feature hierarchies for robust visual tracking, in: Proc. 2015 IEEE Conf. Computer Vision and Pattern Recognition, 2015..
[58] J. Deng, W. Dong, R. Socher, L. Li, K. Li and F.F. Li, ImageNet: A large-scale hierarchical image database, in: Proc. 2009 IEEE Conf. Computer Vision and Pattern Recognition, 2009, pp. 248–255..
[59] H. Nam and B. Han, Learning multi-domain convolutional neural networks for visual tracking, in: Proc. 2016 IEEE Conf. Computer Vision and Pattern Recognition, 2016, pp. 4293–4302..
[60] S. Hong, T. You, S. Kwak, and B. Han, Online tracking by learning discriminative saliency map with convolutional neural network, in: Proc. 2015 IEEE Int. Conf. Machine Learning, 2015, pp. 597–606..
[61] L. Wang, W. Ouyang, X. Wang and H. Lu, Visual tracking with fully convolutional networks, in: Proc. 2015 IEEE Int. Conf. Computer Vision, 2015, pp. 3119–3127..
[62] X. Lu, C. Ma, B. Ni, X. Yang, I. Reid, M.H. Yang, Deep regression tracking with shrinkage loss, in: Proc. 2018 European Conf. Computer Vision, 2018, pp. 369–386.
[63] D. Held, S. Thrun, S. Savarese, Learning to track at 100 FPS with deep regression networks, in: Proc. 2016 European Conf. Computer Vision, 2016, pp. 749–765.
[64] H. Nam, M. Baek and B. Han, Modeling and propagating CNNs in a tree structure for visual tracking, in: Proc. 2016 IEEE Conf. Computer Vision and Pattern Recognition, 2016, pp. 2137–2155..
[65] Z. Cui, S. Xiao, J. Feng and S. Yan, Recurrently target-attending tracking, in: Proc. 2016 IEEE Conf. Computer Vision and Pattern Recognition, 2016, pp. 1449–1458..
[66] G. Ning, Z. Zhang, C. Huang, Z. He, X. Ren, and H.Wang, Spatially supervised recurrent convolutional neural networks for visual object tracking, in: Proc. 2017 IEEE Conf. International Symposium on Circuits and Systems, 2017, pp. 1–4..
[67] J. Redmon, S.K. Divvala, R. Girshick, and A. Farhadi, You only look once: Unified, real-time object detection, in: Proc. 2016 IEEE Conf. Computer Vision and Pattern Recognition, 2016, pp. 779–788..
[68] H. Fan, H.B. Ling, SANet: Structure-aware network for visual tracking, in: Proc. 2017 IEEE Conf. Computer Vision and Pattern Recognition Workshops, 2017, pp. 2217–2224..
[69] R. Tao, E. Gavves, and A.W.M. Smeulders, Siamese instance search for tracking, in: Proc. 2016 IEEE Conf. Computer Vision and Pattern Recognition, 2016, pp. 1420–1429..
[70] B. Li, J. Yan and W. Wu, High performance visual tracking with siamese region proposal network, in: Proc. 2018 IEEE Conf. Computer Vision and Pattern Recognition, 2018, pp. 8971–8980..
[71] X. Wang, A. Shrivastava, and A. Gupta, A-Fast-RCNN: Hard positive generation via adversary for object detection, in: Proc. 2017 IEEE Conf. Computer Vision and Pattern Recognition, 2017, pp. 3039–3048..
[72] X. Dong, J. Shen, S. Porikli, CLNet, A compact latent network for fast adjusting siamese trackers, in: Proc. 2020 European Conf. Computer Vision, 2020, pp. 378–395.
[73] J. Shen, X. Tang, X. Dong, L. Shao, Visual Object Tracking by Hierarchical Attention Siamese Network, IEEE Trans. Cybernet. 50 (7) (2020) 3068–3080.
[74] X. Dong, J. Shen, W. Wang, Y. Liu, L. Shao and F. Porikli, Hyperparameter Optimization for Tracking with Continuous Deep Q-Learning, in: Proc. 2018 IEEE Conf. Computer Vision and Pattern Recognition, 2018, pp. 518–527..
[75] X. Dong, J. Shen, W. Wang, L. Shao, H. Ling and F. Porikli, Dynamical Hyperparameter Optimization via Deep Reinforcement Learning in Tracking, IEEE Trans. Pattern Anal. Mach. Intell. https://doi.org/10.1109/TPAMI.2019.2956703..
[76] Z. Zhu, Q. Wang, B. Li, Distractor-aware siamese networks for visual object tracking, in: Proc. 2018 European Conf. Computer Vision, 2018, pp. 103–119.
[77] B. Li, W. Wu and Q. Wang, SiamRPN++: Evolution of siamese visual tracking with very deep networks, in: Proc. 2019 IEEE Conf. Computer Vision and Pattern Recognition, 2019..
[78] J. Valmadre, L. Bertinetto, J.F. Henriques, A. Vedaldi, and P.H.S. Torr, End-to-end representation learning for correlation filter based tracking, in: Proc. 2017 IEEE Conf. Computer Vision and Pattern Recognition, 2017, pp. 5000–5008..
[79] Q. Wang, J. Gao, and J. Xing, DCFNet: Discriminant correlation filters network for visual tracking, arXiv preprint arXiv:1704.04057, 2017..

[80] Z. Zhu, W. Wu, W. Zou, and J. Yan, End-to-end flow correlation tracking with spatial-temporal attention, in: Proc. 2018 IEEE Conf. Computer Vision and Pattern Recognition, USA, 2018, pp. 548–557..
[81] Y. Xu, Z. Wang, Z. Li, Y. Yuan and G. Yu, SiamFC++: Towards robust and accurate visual tracking with target estimation guidelines, arXiv preprint arXiv:1911.06188..
[82] T. Wang, M. Qiao, M. Zhang, Y. Yang, H. Snoussi, Data-driven prognostic method based on self-supervised learning approaches for fault detection, J. Intell. Manuf. 31 (7) (2020) 1611–1619.
[83] P. Gao, R. Yuan, F. Wang, L. Xiao, H. Fujita, Y. Zhang, Siamese attentional keypoint network for high performance visual tracking, arXiv preprint arXiv: 1904.10128..
[84] N. Wang, Y. Song, C. Ma, W. Zhou, W. Liu, and H. Li, Unsupervised Deep Tracking, in: Proc. 2019 IEEE Conf. Computer Vision and Pattern Recognition, Long Beach, 2019, pp. 1308-1317..
[85] X. Lu, B. Ni, C. Ma, X. Yang, Learning transform-aware attentive network for object tracking, Neurocomputing 349 (2019) 133–f144.
[86] B.K.P. Horn, B.G. Schunck, Determining optical flow, Artif. Intell. 17 (1–3) (1981) 185–203.
[87] B.D. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, in: Proc. Int. Joint Conf. Artificial Intelligence, 1981, pp. 674–679..
[88] J.Y. Bouguet, Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm, Intel Corporation, Microprocessor Research Labs, OpenCV Documentation, 2001.
[89] P.S. Maybeck, The Kalman filter: An introduction to concepts, Autonomous Robot Vehicles (1990).
[90] B.F.L. Scala, R.R. Bitmead, Design of an extended Kalman filter frequency tracker, IEEE Trans. Signal Process. 44 (3) (1994) 739–742.
[91] S.J. Julier, Unscented filtering and nonlinear estimation, Proc. IEEE 92 (3) (2004) 401–422.
[92] K. Nummiaro, E. Koller-Meier, L.V. Gool, An adaptive colorbased particle filter, Image Vis. Comput. 21 (1) (2003) 99–110.
[93] D. Comaniciu, P. Meer, Mean shift: A robust approach toward feature space analysis, IEEE Trans. Pattern Anal. Mach. Intell. 24 (5) (2002) 603–619.
[94] G.R. Bradski, Computer vision face tracking for use in a perceptual user interface, Intel Technol. J. 2 (2) (1998) 12–21.
[95] M. Kristan, A. Leonardis, J. Matas, et al., The visual object tracking vot2016 challenge results. in, in: Proc. 2016 European Conf. Computer Vision Workshops, 2016, pp. 1–45.
[96] H. Fan, L. Lin, F. Yang, et al., LaSOT: A high-quality benchmark for large-scale single object tracking, in: Proc. 2018 IEEE Conf. Computer Vision and Pattern Recognition, 2018..
[97] M. Danelljan, G. Bhat, F.S. Khan and M. Felsberg, ATOM: Accurate Tracking by Overlap Maximization, in: Proc. 2019 IEEE Conf. Computer Vision and Pattern Recognition, 2019, pp. 4655–4664..
[98] K. Dai, Y. Zhang, D. Wang, J. Li, H. Lu and X. Yang, High-performance long-term tracking with meta-updater, in: Proc. 2020 IEEE Conf. Computer Vision and Pattern Recognition, 2020, pp. 6298–6307..
[99] Z. Chen, B. Zhong, G. Li, S. Zhang, R. Ji, Siamese box adaptive network for visual tracking, in: Proc. IEEE Conf. 2020 Computer Vision and Pattern Recognition, 2020, pp. 6667–6676..
[100] Y. Li, C. Fu, F. Ding, Z. Huang and G. Lu, AutoTrack: Towards high-performance visual tracking for UAV with automatic spatio-temporal regularization, in: Proc. 2020 IEEE Conf. Computer Vision and Pattern Recognition, 2020, pp. 11920–11929..
[101] R. Yao, G. Lin, S. Xia, et al., Video object segmentation and tracking: a survey, ACM Trans. Intell. Systems and Tech. 11 (4) 2020..
[102] G. Ciaparrone, F.L. Snchez, S. Tabik, et al., Deep learning in video multi-object tracking: A survey, Neurocomputing 381 (2019) 61–88.
[103] G.S. Walia, R. Kapoor, Recent advances on multicue object tracking: a survey, Artif. Intell. Rev. 46 (1) (2016) 1–39.
[104] A.W.M. Smeulders, D.M. Chu, R. Cucchiara, et al., Visual tracking: an experimental survey, IEEE Trans. Pattern Anal. Mach. Intell. 36 (7) (2014) 1442–1468.
[105] X. Li, W. Hu, C. Shen, et al., A survey of appearance models in visual object tracking, ACM Trans. Intell. Systems and Tech. 4 (4) (2013) 58:1–58:48..
[106] X. Dong, J. Shen, D. Wu, K. Guo, X. Jin, F. Porikli, Quadruplet network with one-shot learning for fast visual object tracking, IEEE Trans. Image Processing 28 (7) (2019) 3516–3527.
[107] X. Dong, J. Shen, Triplet loss in siamese network for object tracking, in: Proc. 2018 European Conf. Computer Vision, 2018, pp. 472–488..
[108] Y. Liu, X. Dong, X. Lu, F.S. Khan, J. Shen and S. Hoi, Teacher-students knowledge distillation for siamese trackers, arXiv preprint arXiv:1907.10586..
[109] N. Wang, W. Zhou, Y. Song, C. Ma, H. Li, Real-time correlation tracking via joint model compression and transfer, IEEE Trans. Image Processing 29 (2020) 6123–6135.
[110] N. Wang, W. Zhou, Y. Song, C. Ma, W. Liu and H. Li, Unsupervised deep representation learning for real-time tracking, arXiv preprint arXiv:1904.01828..

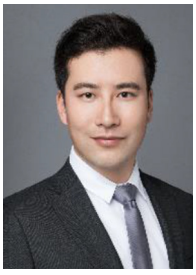
Yucheng Zhang received B.S. degree from University of Jinan, Jinan, China, in 2017 .He is currently pursuing the M.S. degree with the School of Automation Science and Electrical Engineering, Beihang University, Beijing, China. His research interests include machine learning and computer vision.

Tian Wang received the B.S. degree and M.S. degree from Xi'an Jiaotong University, China, in 2007 and 2010, respectively. He received Ph.D. degree from University of Technology of Troyes, France, in 2014. He is an associate professor at the Institute of Artificial Intelligence, Beihang University. His research interests include artificial intelligence, machine learning, computer vision and pattern recognition.

Kexin Liu received the M.Sc. degree in control science and engineering from Shandong University, Jinan, China in 2013, and Ph. D degree in System Theory from Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, China in 2016, respectively. From 2016 to 2018, he was a postdoctoral fellow in Peking University, Beijing, China. Currently, he is an associated professor with the School of Automation Science and Electrical Engineering, Beihang University, Beijing, China. His research interests include multi-agent systems and complex networks.



Baochang Zhang is currently an Professor with Beihang University, Beijing, China. His research interests include pattern recognition, machine learning, face recognition, and wavelets.



Lei Chen received the Ph.D. degree in control theory and engineering from Southeast University, Nanjing, China, in 2018. He was a visiting Ph.D. student with the Royal Melbourne Institute of Technology University, Melbourne, VIC, Australia, and Okayama Prefectural University, Soja, Japan. He is currently a Post-Doctoral Fellow with the School of Automation Science and Electrical Engineering, Beihang University, Beijing, China. His current research interests include complex networks, characteristic model, spacecraft control, and network control.