High-Performance Visual Tracking Based on High-Order Pooling Network

XINXI FENG1 AND LEI PU2

1College of Artificial Intelligence, Yangtze University, Fuzhou 350015, China
2Combat Support College, Rocket Force University of Engineering, Xi’an 710025, China

Corresponding author: Lei Pu (warmstoner@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61703423 and Grant 62072370.

ABSTRACT Convolution Neural Network (CNN) features have been widely used in visual tracking due to their powerful representation. As an important component of CNN, the pooling layer plays a critical role, but the max/average/min operation only explores the first-order information, which limits the discrimination ability of the CNN features in some complex situations. In this paper, a high-order pooling layer is integrated into the VGG16 network for visual tracking. In detail, a high-order covariance pooling layer is employed to replace the last maxpooling layer to learn discrimination features and is trained on the ImageNet and CUB200-2011 data sets. In tracking stage, the multiple levels of feature maps are extracted as the appearance representation of the target. After that, the extracted CNN features are integrated into the correlation filters framework when tracking is on-the-fly. The experimental results show that the proposed algorithm achieves excellent performance in both success rate and tracking accuracy.

INDEX TERMS Visual tracking, correlation filter, high-order pooling network, deep features, model fusion.

I. INTRODUCTION

Visual tracking is a difficult and fundamental problem in the fields of pattern recognition and computer vision. It has a wide range of applications in real life, such as traffic surveillance, motion analysis, auto-driving and human-computer interaction. Visual tracking generally refers to single-object tracking, that only the position and scale information of the first frame is given, tracking task is to estimate the state of a target in subsequent frames [28]. Although after more than three decades of development, achieving a robust visual tracking system remains a challenging task. The difficulties of this task are mainly concentrated on two aspects, one is the appearance change of the target itself, including scale variation, deformation, fast motion, etc, the other is the external environment changes, including illumination changes, background clutter, severe occlusion and so on. Researchers have shown that good feature representation plays a very important role in dealing with the above complex scenarios [51]. In order to obtain a good feature representation of the target, researchers have designed a large number of excellent hand-crafted features, such as HOG, SIFT and so on. However, these features only have good performance for some simple scenes and are difficult to deal with complex scenes.

In recent years, deep learning has been well applied in all aspects of computer vision [3], [17], [44], and has achieved great performance improvement in tasks such as image classification [23], [43], object detection [44], image segmentation [34], and object recognition [17]. Because of the strong correlation between visual tracking task and target detection, image recognition and other tasks, especially in the aspect of feature representation, it is natural to apply deep features to visual tracking.

There are many researches on using deep features for tracking tasks, which can be divided into two categories according to whether the features need to be updated online. The first type is the method that does not need online update, which can be further divided into two types. The first type is to combine the pre-trained classification network model and the correlation filter [18], [36], [42]. For example, Danelljan et al. [36] use VGG19 network as the feature extractor. At the same time, in order to improve the tracking performance, multi-layer feature fusion is adopted. However, no explanation is given on the selection of fusion weights,
and only fixed weight fusion method is used. In this regard, Danelljan et al. [42] introduced a Hedge method to realize the weight adaptation of multi-layer feature fusion. Another way is to directly train an end-to-end tracking model by using a large number of data sets. In particular, the Siamese networks are developing rapidly [2], [16], [24], [25], [56]. The tracking methods based on the Siamese network integrate the feature representation and the tracking model, and keep the same in the tracking process, achieving a faster tracking speed. The methods of online updating mainly include MDNet [40] based on multi-branches, ATOM [10] based on correlation filter updating and GradNet [27] based on gradient updating. The common problem of online updating methods is that the speed is slow and the model is relatively complex.

However, all the network structures adopted by these methods are based on the first-order pooling operation, and no further analysis of the high-order statistical information of the features is performed. In recent years, the method based on high-order statistical modeling has gained wide attention in the field of computer vision, especially in the task of object recognition [15] and fine-grained image classification [29] with high similarity of different types of objects. Meanwhile, it has obvious advantages in feature representation in the task of person re-identification [6] and semantic segmentation [5] with high similarity of same class objects. In order to improve the discrimination ability of the tracking algorithm to similar targets, this paper makes appropriate improvements to the feature extraction network, and changes the original first-order max pooling to the second-order covariance pooling method, so that the network can extract more high-order statistics about the target features and improve the model discriminative power. As shown in Fig.1, in the Biker video sequence, our deep features can be more focused on the target to be tracked, in the CarScale and Dog sequences, the traditional first-order pooling CNN model can not extract the characteristics of the target itself well, and it is easy to be disturbed by the surrounding objects. However, the deep model in this paper has better response value to the target.

At the same time, the paper also retrained the network on ImageNet [45] and the fine-grained image CUB200-2011 data set [50], which further improved the model’s ability to distinguish similar objects.

Starting from improving the representative ability of target features, this paper proposes a visual tracking algorithm based on high-order pooling networks (HOPNet). Firstly, the high-order pooling network is constructed. Then a large number of data sets are adopted to train this model, so that the learned features can be more discriminative. Lastly, the

![FIGURE 1. The visualization comparison between the original deep feature(Base Model) and enhanced deep feature(Ours).](image-url)

In the Biker, CarScale, and Dog sequences, our deep features are more focused on the target to be tracked, while the traditional first-order pooling CNN model can not extract the characteristics of the target itself well, and it is easy to be disturbed by the surrounding objects. However, the deep model in this paper has better response value to the target.
Trained model is combined with the correlation filter algorithm, which achieved very good performance on large-scale tracking data set.

To the best of our knowledge, this algorithm is the first to extract high-order statistical information by a high-order pooling network for visual tracking tasks.

II. RELATED WORKS

In this section, we make a brief review of correlation filter based tracker with feature representation. A more detailed review can be found in [28], [47], and [8].

A. HAND-CRAFTED FEATURE BASED CF TRACKERS

The problem of visual tracking has a long history and has been widely concerned because of its wide application. Aim- ing at the slow speed of traditional visual tracking methods, in order to further improve the efficiency of the algorithm, the correlation filtering algorithm based on ridge regression came into being. Bolme et al. [4] first proposed the least squares based correlation filter algorithm (MOSSE). The algorithm is simple in design, and uses gray features as appearance representation. The tracking speed is close to 700 frames per second, which has quickly attracted wide attention. However, the algorithm still uses particle filter for sample sampling, which limits the number of available samples. Then, Hen- riques et al. Introduced the idea of circulant matrix [19], and realized dense sampling through a single sample, which greatly improved the discriminant power of the model. How- ever, the single channel gray feature limits the ability of the algorithm to deal with complex scenes. Henrique et al. Introduced a more expressive multi-channel HOG (Histogram Of Gradient) feature to enrich the expression of structural information, and derived the corresponding block image cycle mode and multi-channel correlation filter model [20]. On this basis, Liu et al. [11] constructed color attribute features (CN, color name) by analyzing RGB images, and combined them with correlation filter algorithm, which can deal with complex tracking scenes such as target deformation and rotation change. Of course, considering the natural complementarity between HOG feature based on structure information and color feature based on global statistics, Liu et al. [1] added an additional branch based on KCF algorithm, obtained the response map of tracking position by template matching using color histogram feature, and finally fused with the original response map. It can effectively improve the algorithm’s ability to deal with various complex scenes.

B. DEEP FEATURE BASED CF TRACKERS

Considering the great success of deep feature in the field of image classification, it is proved that deep feature has a strong ability to express the appearance of the target, and it is a natural direction to combine it with correlation filter algorithm [13], [18], [36], [42]. In 2015, Danelljan et al. [36], [37], [38] used the pre-training network VGG19 as the feature extractor, and combined it with the correlation filter algorithm. In order to improve the accuracy of the algorithm, the multi-layer fusion positioning method from coarse to fine was adopted, and the accuracy on OTB2013 [52] data set reached 0.891, which was far higher than other tracking methods at that time, reflecting the effectiveness of deep feature in the field of target tracking. Then, Danelljan et al. [41], [42] proposed using hedge algorithm to adaptively adjust the fusion weight of output response maps under the idea of multi-layer feature fusion algorithm. There are also a lot of follow-up work focused on feature selection and fusion, such as CFWCR [18] and UPDT [3]. The research idea also belongs to multi-feature fusion method, which mainly selects expressive subset features in a large set, and on this basis, designs adaptive fusion algorithm through the difference of representation ability of features for different targets. On the other hand, Ma et al. [33] applied the CNN model trained by the visible light data set to the thermal infrared tracking task by using the transfer learning method, to improve the discriminative capacity, a multi-level similarity model under a Siamese framework [32] and a multi-task framework [31] were proposed to learn the TIR-specific discriminative features, moreover, Liu et al. proposed a framework named self-SDCT [54], which can alleviate the demand for large annotated training samples. Subsequently, Martin Danelljan and others have proposed a series of high-performance tracking algorithms, such as DeepSRDCF [13], C-COT [14], ECO [9].

III. PROPOSED METHOD

This paper proposes a robust visual tracking algorithm based on high-order pooling networks. Fig. 2 shows the pipeline of the proposed HOPNet tracker. As shown in Fig. 2, the high-order pooling network is trained on ImageNet and a large number of fine-grained image data set. After offline training, the pre-trained network parameters are retained as feature extractors. In the tracking stage, in order to improve the tracking performance, the third, the fourth and the fifth layer are used as the appearance representation of the target. By combining with the STRCF [26] algorithm, the performance is greatly improved.

A. CORRELATION FILTER

Tracking algorithms can be divided into generative trackers and discriminant trackers. As a typical representative of discriminant tracking algorithm, correlation filter has received a lot of attention and widely research because of its good discriminant ability and fast speed. The cyclic matrix and fast Fourier transform are introduced into the correlation filter algorithm, which achieves better tracking performance and higher computational efficiency. The correlation filter algorithm can be regarded as a ridge regression model as follows:

\[ \mathcal{L}(h) = \min_{h} \| f \odot h - g \|^2 + \lambda \| h \|^2 \]  

where, \( f \) represents the training samples, \( h \) denotes the filters, \( \odot \) is the spatial correlation operator. \( g \) is the ideal output, and is generally set as Gaussian window function, in which each
element corresponds to the label value of one training sample, \( \lambda \) is a regularization parameter.

## B. HIGH-ORDER POOLING NETWORK

Convolutional neural networks (CNNs) have built many excellent network architectures by stacking convolutional layers, nonlinear activation layers, and pooling layers, and are widely used in all aspects of computer vision. At present, most of the network-based improvements aim to improve the performance of the network by widening and deepening the network. There is little research work to improve the feature representation ability from the perspective of high-order information [29]. In order to enhance the discriminative power of features, this paper proposes a high-order pooling network as the target feature extractor. Next, we will briefly introduce the forward and backward propagation formulas involved in the structure, as shown in Fig. 3.

### 1) DERIVATION OF FORWARD PROPAGATION

First, the output of the last convolutional layer needs to be processed. After that, the feature matrix is expressed as \( X \in \mathbb{R}^{d \times N} \), where \( d \) is the number of feature channels, \( N = h \times w \), \( h \) and \( w \) is the feature map size of the last convolutional layer. Then we calculate the covariance matrix of the feature map:

\[
P = X \bar{X}^T
\]

(2)

where \( \bar{X} = \frac{1}{N} (I - \frac{1}{N} uu^T) \), \( u = [1, \ldots, 1]^T \) is a \( N \)-dimensional vector. Then the eigenvalue decomposition is used to process the obtained covariance matrix to obtain eigenvalues and eigenvectors:

\[
P = U \Lambda U^T
\]

(3)

where \( \Lambda = \text{diag} (\lambda_1 \ldots \lambda_d) \) is a diagonal matrix and \( \lambda_i \) is an eigenvalue. \( U = [u_1, \ldots, u_d] \) is the corresponding feature vector. Through the above eigenvalue decomposition, we can convert the power of the matrix to solve the power of the eigenvalue.

\[
Q = P^\alpha = UF (\Lambda)^{\alpha} U^T
\]

(4)

in this paper, \( \alpha = 0.5, F (\Lambda) = \text{diag} (f (\lambda_1), \ldots, f (\lambda_d)) \), \( f (\lambda_i) \) which represents the exponentiation of eigenvalues:

\[
f (\lambda_i) = \lambda_i^\alpha
\]

(5)

Now, the forward propagation of the covariance pooling layer has been deduced.

### 2) DERIVATION OF BACKWARD PROPAGATION

Next, this paper briefly deduces the backward propagation process of the covariance pooling layer. First the chain transfer process for \( U \) and \( \Lambda \) is described as follows:

\[
tr \left( \left( \frac{\partial l}{\partial U} \right)^T duU + \left( \frac{\partial l}{\partial \Lambda} \right)^T d\Lambda \right) = tr \left( \left( \frac{\partial l}{\partial Q} \right)^T dQ \right)
\]

(6)

According to formula (4), we obtain:

\[
\frac{\partial l}{\partial P} = \left( \frac{\partial l}{\partial Q} + \frac{\partial l}{\partial \Lambda} \right)^T UF
\]

(7)

\[
\frac{\partial l}{\partial X} = \alpha \left( \text{diag} (\lambda_1^{\alpha-1}, \ldots, \lambda_d^{\alpha-1}) U^T \frac{\partial l}{\partial Q} \right)_\text{diag}
\]

(8)
According to the chain derivation rule, $\frac{\partial l}{\partial p}$ and $\frac{\partial l}{\partial x}$ are as computed as follows:

$$\frac{\partial l}{\partial p} = U \left( \left( K^T \circ \left( U^T \frac{\partial l}{\partial U} \right) + \left( \frac{\partial l}{\partial Q} \right)_{\text{diag}} \right) U^T \right) \quad \text{(9)}$$

$$\frac{\partial l}{\partial x} = \hat{X} \left( \frac{\partial l}{\partial p} + \left( \frac{\partial l}{\partial P} \right)^T \right) \quad \text{(10)}$$

where matrix $K = \{K_{ij}\}$, $K_{ij}$ expressed as

$$K_{ij} = \begin{cases} 1/(\lambda_i - \lambda_j), & i \neq j \\ 0, & i = j \end{cases} \quad \text{(11)}$$

C. COMBINED WITH SPATIAL-TEMPORAL REGULARIZED CORRELATION FILTERS

In order to validate the effectiveness of the deep feature after the high-order pooling in the tracking task, this paper combines it with the spatial-temporal regularized correlation filters in a multi-layer fusion manner. Based on the general correlation filter framework, the STRCF tracker proposed a spatial regularization term to deal with boundary effects and a temporal regularization term for filter degradation. The objective function is as follows:

$$E(h) = \left( \frac{1}{D} \sum_{d=1}^{D} h^d - g \right)^2 + \lambda \sum_{d=1}^{D} \left\| w^d \right\|^2_2 + \mu \left\| h - h_{t-1} \right\|^2_2 \quad \text{(12)}$$

where, $d$ is the channel index, and $D$ is the channel number. $w$ is the spatial penalty weights. $h_{t-1}$ is a temporal regularization coefficient that represents the filter template of the previous frame. This method can effectively cope with the occlusion, fast motion and other scenes.

IV. EXPERIMENTS

In order to fully verify the effectiveness and accuracy of the proposed algorithm, we tested the proposed algorithm on a large number of data sets, including OTB2015 [53], Temple-Color128 [30] and TrackingNet [39], and made a horizontal comparison with some related algorithms. Firstly, the implementation details of the algorithm are briefly introduced.

A. IMPLEMENTATION DETAILS

The proposed tracker is implemented using MATLAB2018a and all the experiments are executed on a PC with a Intel Core i5-8400 2.8GHz CPU with a GeForce GTX Titan Xp GPU. The average testing speed of HOPNet is 10 FPS. The MatConvNet toolbox [48] is used to implement the improved VGG16 network [46] based on high-order pooling. The test parameters are set as follows: in the training stage, aiming at the problem that the dimension of the last layer of convolution is too high, this paper uses $1 \times 1$ convolution to reduce the channel dimension of the last convolution layer from 512 dimension to 256 dimension, and then send it to the covariance pooling layer for processing. In the testing stage, the third, the fourth and the fifth layer features of the network are selected as the appearance representation of the target, and the extracted features are processed by cosine window to eliminate the boundary discontinuity. The size of the target search box is a square area with the target as the center and the side length as the center, where and represent the length and width of the target respectively. For the regularization parameter in Eq. 12, $\lambda = 1$, $\mu = 16$. The scale estimation adopts the same parameter settings as those in reference [26], the weights of multi-layer fusion are set as 1, 0.5, 0.5 from fifth layer to third layer.

B. DATASETS AND COMPARED TRACKERS

This paper uses the ImageNet and the CUB200-2011 data set as the training set of the network. The OTB2015, Temple-Color128 and TrackingNet are used to test our proposed method. The OTB2015 dataset contains 100 challenging video sequences, which can be divided into 11 annotation
attributes, including fast motion, motion blur, background clutters, deformation, illumination variation, in-plane rotation, low resolution, occlusion, out of plane rotation, out of view and scale variation.

This paper compares the proposed algorithm (High Order Pooling Network, HOPNet) with other 10 recent and relevant tracking algorithms on the OTB2015 dataset [53], including:

- Deep features based CF tracker: MCPF [55], DeepSRDCF [13], DeepSTRCF [26], HDT [42], HCF [36];
- Hand-crafted features based CF tracker: STRCF [26];
- CNN based tracker: MDNet [40];
- Siamese based tracker: GradNet [27], SiamRPN [25], SiamFC [2].

**C. OVERALL PERFORMANCE**

Fig. 4 shows the comparison curve of tracking accuracy and success rate of the algorithm with other relevant algorithms in this paper. In recent years, MDNet has been the champion of the visual object tracking challenge (VOT2015) [22]. It has achieved the best performance in many datasets by offline training and online fine-tuning, but the tracking speed of MDNet is less than 1 FPS, which makes the application extremely limited. The proposed HOPNet tracker is slightly better than MDNet in precision, with a performance improvement of 1.9% in success rate and a speed of about 10 times of MDNet. Compared with HCF, HDT, deepSRDCF and MCPF, the algorithm in this paper has obvious performance advantages, and the speed is equivalent to them. In the comparison algorithm, GradNet and SiamRPN are Siamese network based tracking algorithms that proposed in recent years, which have good performance in speed and accuracy. Although the calculation speed of our method is slightly slow, we still have nearly 4% performance advantages in accuracy and success rate. Finally, compared with the STRCF algorithm, the accuracy of the algorithm is improved by 4.1%, and the success rate is improved by 3.8%, which shows a significant performance improvement.

**D. ATTRIBUTE-BASED EVALUATION**

In addition to the overall performance on the data set, we also need to focus on the tracking performance in different tracking scenarios. As mentioned above, 100 videos of OTB2015 can be divided into 11 attributes. Fig. 5 and Fig. 6 show the tracking accuracy and success rate of different algorithms under different attributes. It can be clearly seen from the Fig. 5 and Fig. 6 that the algorithm in this paper has achieved excellent tracking results on almost all attributes. Especially in dealing with background clutters (BC), the algorithm in this paper achieves 0.705 in success rate and 0.902 in tracking accuracy, which is far superior to similar tracking algorithms based on deep characteristics (HCF, HDT, etc.). This further verifies that the high-order pooling network used in this paper has a good ability to distinguish similar objects. In addition, in the low-resolution target tracking, the accuracy of the algorithm is 0.977, and the success rate is 0.702, which is mainly due to the further training of the feature extraction network in the fine-grained image data set, which improves the representation ability of the network. In complex scenes such as illumination change, rotation and motion blur, the algorithm in this paper achieves the optimal tracking accuracy and success rate.

**E. QUANTITATIVE EVALUATION**

In order to show the effectiveness of this algorithm more intuitively, it is compared with the most relevant three comparison algorithms on five challenging image videos, including Girl2, Bolt2, Biker, Lemming, Human3. As shown in Fig. 7, the algorithm in this paper can well deal with these complex scenes.

(1) **Fast Motion:** because the correlation filter algorithm is to detect and track the target in a fixed search area, it is difficult to deal with it well when the target moves fast. At the same time, fast movement will lead to obvious changes in the target’s appearance, which further aggravates the difficulty of tracking. For example, in Biker, there is a fast and obvious appearance change of the
target between frames 61 and 90. HCF and HDT algorithm based on the pre-trained first-order network lose the target. The DeepSRDCF algorithm with a large search area tracks the target, but the positioning accuracy.
FIGURE 7. Some tracking results on challenging image sequences (from top to down are Girl2, Bolt2, Biker, Lemming, Human3). We show some tracking results of DeepSRDCF [13], HCF [36], HDT [42] methods and the proposed HOPNet tracker.

and tracking success rate are lower than the algorithm in this paper.

2) **Background Clutters**: the interference of similar objects will also have a significant impact on tracking performance. There are similar objects and similar backgrounds in *bolt* and *lemming*. Because the algorithm in this paper adopts the deep feature and carries out further training on the fine-grained image, it has a better ability to distinguish the background clutter scene, and can achieve a stable tracking.

3) **Target Occlusion**: occlusion is always a difficult problem in tracking, which easily leads to tracking failure. There is occlusion in *Girl2* and *Lemming* video sequences, and HCF and HDT algorithms based on pre-trained network still fail to track. Although the algorithm in this paper does not use the corresponding occlusion detection mechanism, but thanks to better feature representation, it can accurately capture the target when the target reappears, while HCF and HDT algorithm cannot detect the target again because of their weak feature discrimination.

4) **Low Resolution**: in the field of computer vision, the detection and tracking of low resolution targets are always difficult problems. Due to the low resolution, the available information is less and the ability of feature representation is weaker. The target tracked in *Biker* and *Human3* is smaller, and *Human3* is basically a pure black object, which has a higher requirement for
feature discrimination. Because the multi-layer feature fusion method is adopted, the shallow feature contains the detailed information of the target, which has obvious advantages for processing low resolution small targets.

F. ABLATION STUDY

In order to further analyze the influence of different convolution layer characteristics of high-order pooling network on tracking performance, this paper extracts the characteristics of each layer and makes five groups of comparative experiments on OTB2015 data set, and the experimental results are shown in Table 1. From Table 1, it can be seen that the best tracking performance is the fourth layer feature, and the worst is the second layer feature, with more performance degradation. In addition, for the improvement of tracking performance, this paper adopt the multi-layer feature fusion method similar to HCF and HDT.

G. TEMPLE-COLOR128 AND TRACKINGNET DATASETS

In order to fully verify the tracking performance of the algorithm, the algorithm is evaluated on the Temple-Color128 dataset [30] and the TrackingNet [39]. Temple-Color128 includes 128 RGB sequences in different tracking scenarios. In this paper, the performance of the algorithm is compared by using the same evaluation metrics of OTB2015 dataset. Fig. 8 shows the accuracy and success curves of the six tracking algorithms. Among these comparison algorithms, the HOPNet tracker proposed in this paper obtains the best results (0.794, 0.588) on both evaluation metrics, which is better than the state-of-the-art C-COT [14] and MCPF tracker [55]. Meanwhile, compared with DeepSRDCF [13], this method has 5.6% performance improvement in precision rate and 5.1% improvement in success rate, which is also significantly better than CF2 tracker [36]. In general, compared with the most advanced tracking algorithm in Temple-Color128 data set, the algorithm proposed in this paper still has good performance and competitiveness.

The TrackingNet data set includes 30000 videos, covering a wide range of target categories and complex tracking scenarios. As shown in Table 2, our tracker has achieved 0.642, 0.597 and 0.738 in terms of success rate, accuracy and norm precision respectively. Compared with the correlation filter algorithm ECO [9] based on the same pre-trained model, this algorithm has obvious performance improvement, which fully verifies that the high-order pooling network has better feature expression ability. However, it is worth noting that compared with the tracking algorithms of Siamese network based and transformer based, the algorithm in this paper still has a large gap in performance. The main reason is that the algorithms of TransT [7] and SiamRPN++ [24] are trained through large-scale data sets, and the backbone networks are also better than the algorithm in this paper. In the future, we can consider combining the high-order pooling with the Siamese network.

H. FAILURE ANALYSIS

As shown in the Fig. 9, for the Twinings sequence, when the target encounters rotation, rapid change in scale and serious interference from similar objects, the algorithm in this paper
cannot quickly adjust to the current position and scale of the target due to the limited range of scale estimation and the search area, resulting in tracking failure. For the Box sequence, because the object is dark and there are many significant objects around. For Matrix sequence, because the object resolution is very low, the feature is not very significant, and the target and the scene are very similar, the algorithm in this paper is difficult to distinguish the target and the background in the tracking scenes.

I. DISCUSSION
Feature representation is a hot topic in the field of computer vision, especially in target tracking. With the wide application of convolutional neural network, its performance has been greatly improved. However, no matter whether the network is widened or deepened, the network still obtains the first-order information about the target, and its distinguishing ability is limited in the face of similar object interference. In this regard, mining high-order information about the target is a topic worthy of study. In this paper, the high-order pooling module is introduced to optimize the feature representation of the convolutional neural network, so as to further improve the performance of the tracking algorithm. Especially in the environment of similar object interference and low resolution, the performance is improved obviously. But at present, only high-order modules can be added to the deep layer of the network. How to further optimize the shallow layer will be the focus of the follow-up research.

V. CONCLUSION
In this paper, a robust visual tracking algorithm based on high-order pooling network is proposed. Firstly, on the basis of VGG16 network framework, the last first-order max pooling layer is replaced by the high-order covariance pooling layer, and the forward propagation and back propagation involved in this module are briefly deduced. Through a large number of data sets for training, the improved high-order pooling network has better ability of feature representation. Finally, the feature extraction network and correlation filter algorithm are combined to achieve the best tracking performance in large-scale data sets.

Due to the successful application of end-to-end idea in tracking tasks, we will continue to study how to introduce higher-order information into Siamese network structure to enhance the ability of feature discrimination.

REFERENCES

[1] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, “Staple: Complementary learners for real-time tracking,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Las Vegas, NV, USA, Jun. 2016, pp. 1401–1409.
[2] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, “Fully-convolutional Siamese networks for object tracking,” in Proc. Eur. Conf. Comput. Vis. Workshops (ECCVW), 2016, pp. 850–865.
[3] G. Bhat, J. Johnander, M. Danelljan, F. Shahbaz Khan, and M. Felsberg, “Unveiling the power of deep tracking,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Sep. 2018, pp. 483–498.
[4] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, “Visual object tracking using adaptive correlation filters,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2010, pp. 2544–2550.
[5] I. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu, “Semantic segmentation with second-order pooling,” in Proc. Eur. Conf. Comput. Vis. Springer, 2012, pp. 430–443.
[6] B. Chen, W. Deng, and J. Hu, “Mixed high-order attention network for person re-identification,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 371–381.
[7] X. Chen, B. Yan, J. Zhu, D. Wang, X. Yang, and H. Lu, “Transformer tracking,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2021, pp. 8126–8135.
[8] Z. Chen, Z. Hong, and D. Tao, “An experimental survey on correlation filter-based tracking,” 2015, arXiv:1509.05320.
[9] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, “ECO: Efficient convolution operators for tracking,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 6931–6939.
[10] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, “ATOM: Accurate tracking by overlap maximization,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 4660–4669.
[11] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, “Coloring channel representations for visual tracking,” in Proc. Scandian. Conf. Image Anal. Springer, 2015, pp. 117–129.
[12] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, “Learning spatially regularized correlation filters for visual tracking,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 4310–4318.
[13] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, “Convolutional features for correlation filter based visual tracking,” in Proc. IEEE Int. Conf. Comput. Vis. Workshop (ICCVW), Dec. 2015, pp. 58–66.
[14] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, “Beyond correlation filters: Learning continuous convolution operators for visual tracking,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2016, pp. 472–488.
[15] Z. Gao, J. Xie, Q. Wang, and P. Li, “Global second-order pooling convolutional networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2019, pp. 3024–3033.
[16] A. He, C. Luo, X. Tian, and W. Zeng, “A twofold Siamese network for real-time object tracking,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Las Vegas, NV, USA, Jun. 2016, pp. 770–778.
[17] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR), Las Vegas, NV, USA, Jun. 2016, pp. 770–778.
[18] Z. He, Y. Fan, J. Zhang, Y. Dong, and H. Bai, “Correlation filters with weighted convolution responses,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Oct. 2017, pp. 1992–2000.

[19] J. A. F. Henriques, R. Caseiro, P. Martins, and J. Batista, “Exploiting the circumstantial structure of tracking-by-detection with kernels,” in Proc. Eur. Conf. Comput. Vis. Berlin, Germany: Springer-Verlag, 2012, pp. 702–715.

[20] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, “High-speed tracking with kernelized correlation filters,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, no. 3, pp. 583–596, Mar. 2015.

[21] H. K. Galoogahi, A. Fagg, and S. Lucey, “Learning background-aware correlation filters for visual tracking,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 1135–1143.

[22] M. Kristan and J. M. et al., “The visual object tracking VOT2015 challenge results,” in Proc. Int. Conf. Int. Conf. Comput. Vis. Workshop (ICCV), 2015, pp. 564–586.

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in Proc. NIPS, 2012, pp. 1106–1114.

[24] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, “SiameseRN++: Evolution of Siamese visual tracking with very deep networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2019, pp. 4282–4291.

[25] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, “High performance visual tracking with Siamese region proposal network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 8971–8980.

[26] F. Li, C. Tian, W. Zuo, L. Zhang, and M.-H. Yang, “Learning spatial-temporal regularized correlation filters for visual tracking,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Feb. 2018, pp. 4904–4913.

[27] P. Li, B. Chen, W. Ouyang, D. Wang, X. Yang, and H. Lu, “GradNet: Gradient-guided network for visual object tracking,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 6162–6171.

[28] P. Li, D. Wang, L. Wang, and H. Lu, “Deep visual tracking,” Pattern Recognit., vol. 76, pp. 323–338, Apr. 2018.

[29] P. Li, J. Xie, Q. Wang, and W. Zuo, “Is second-order information helpful for large-scale visual recognition?” in Proc. IEEE Int. Conf. Comput. Vis., Oct. 2017, pp. 2070–2078.

[30] P. Liang, E. Blasch, and H. Ling, “Encoding color information for visual tracking: Algorithms and benchmark,” IEEE Trans. Image Process., vol. 24, no. 12, pp. 5630–5644, Dec. 2015.

[31] Q. Liu, X. Li, Z. He, N. Fan, D. Yuan, W. Liu, and Y. Liang, “Multi-task driven feature models for thermal infrared tracking,” in Proc. AAAI Conf. Artif. Intell., vol. 34, no. 7, Apr. 2020, pp. 11604–11611.

[32] Q. Liu, X. Li, Z. He, N. Fan, D. Yuan, and H. Wang, “Learning deep multi-level similarity for thermal infrared object tracking,” IEEE Trans. Multimedia, vol. 23, pp. 2114–2126, 2020.

[33] Q. Liu, X. Lu, Z. He, C. Zhang, and W.-S. Chen, “Deep convolutional neural networks for thermal infrared object tracking,” Knowl.-Based Syst., vol. 134, pp. 189–198, Oct. 2017.

[34] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2015, pp. 3431–3440.

[35] A. Lukeźiń, T. Vojíř, L. C. Zajc, J. Matas, and M. Kristan, “Discriminative correlation filter tracker with channel and spatial reliability,” Int. J. Comput. Vis., vol. 126, pp. 671–688, Jul. 2018.

[36] C. Ma, J. Huang, X. Yang, and M. Yang, “Hierarchical convolutional features for visual tracking,” in Proc. Int. Conf. Comput. Vis. (ICCV), 2015, pp. 3074–3082.

[37] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, “Adaptive correlation filters with long-term and short-term memory for object tracking,” Int. J. Comput. Vis., vol. 126, no. 8, pp. 771–796, Aug. 2018.

[38] C. Ma, Y. Xu, B. Ni, and X. Yang, “When correlation filters meet convolutional neural networks for visual tracking,” IEEE Signal Process. Lett., vol. 23, no. 10, pp. 1455–1458, Oct. 2016.

[39] M. Müller, A. Bibi, G. Giancola, S. Alishahi, and B. Ghanem, “TrackingNet: A large-scale dataset and benchmark for object tracking in the wild,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 300–317.

[40] H. Nam and B. Han, “Learning multi-domain convolutional neural networks for visual tracking,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 4293–4302.

[41] Y. Qi, S. Zhang, L. Qin, Q. Huang, H. Yao, J. Lim, and M.-H. Yang, “Hedging deep features for visual tracking,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 5, pp. 1116–1130, May 2018.

[42] Y. Qi, S. Zhang, L. Qin, H. Yao, Q. Huang, J. Lim, and M. Yang, “Hedged deep tracking,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 4303–4311.

[43] R. Waseem and Z. Wang, “Deep convolutional neural networks for image classification: A comprehensive review,” Neurocomput., vol. 29, no. 9, pp. 2352–2449, Aug. 2017.

[44] J. Redmon and A. Farhadi, “YOLO9000: Better, faster, stronger,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 6517–6525.

[45] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and F. Li, “ImageNet large scale visual recognition challenge,” Int. J. Comput. Vis., vol. 115, no. 3, pp. 211–252, Dec. 2015.

[46] K. Simonyn and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2014, arXiv:1409.1556.

[47] A. W. M. Smudeiders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, “Visual tracking: An experimental survey,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 36, no. 7, pp. 1442–1468, Jul. 2014.

[48] A. Vedaldi and K. Lenc, “MatConvNet: Convolutional neural networks for MATLAB,” in Proc. 25th Annu. ACM Int. Conf. Multimedia, 2015, pp. 689–692.

[49] P. Voigtlaender, J. Luiten, P. H. Torr, and B. Leibe, “Siam R-CNN: Visual tracking by re-detection,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2020, pp. 6578–6588.

[50] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, “The Caltech-UCSD birds-200–2011 dataset,” Tech. Rep., 2011.

[51] N. Wang, J. Shi, D. Yeung, and J. Jia, “Understanding and diagnosing visual tracking systems,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 3101–3109.

[52] Y. Wu, J. Lim, and M. Yang, “Online object tracking: A benchmark,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2013, pp. 2411–2418.

[53] Y. Wu, J. Lim, and M. H. Yang, “Object tracking benchmark,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, no. 9, pp. 1834–1848, Sep. 2015.

[54] D. Yuan, X. Chang, P. Huang, and Q. Liu, “Self-supervised deep correlation tracking,” IEEE Trans. Image Process., vol. 30, pp. 976–985, 2020.

[55] T. Zhang, C. Xu, and M.-H. Yang, “Multi-task correlation particle filter for robust object tracking,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 4819–4822.

[56] Y. Zhang, L. Wang, J. Qi, D. Wang, M. Feng, and H. Lu, “Structured Siamese network for real-time visual tracking,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Sep. 2018, pp. 351–366.