Robust Visual Tracking via Semiadaptive Weighted Convolutional Features

Haijun Wang, Shengyan Zhang, Hongjuan Ge, Guo Chen, and Yujie Du

***Abstract*—In recent years, hierarchical features extracted from convolutional neural network (CNN) for robust visual tracking have been developed by several methods. As features from different layers characterize different information of target and are set fixed weighted parameters, the performance of traditional visual tracking methods based on CNN can be further improved. In this letter, we propose a novel online visual tracking method by using hierarchical convolutional features with semiadaptive weights. The responses from different layers are assessed by a novel loss function. It considers the log likelihood and the entropy term of each response. The layer with the lower loss value is set to a higher weight parameter and the layer with the higher loss value is set to a lower one. We further develop a target appearance pyramid to deal with the scale change and an online classifier to redetect targets in case of tracking failures. Extensive experiments on challenging videos demonstrate that our method can achieve better tracking results in terms of lower center location error and higher overlap rate.**

***Index Terms*—Convolutional neural network, correlation filters, visual tracking, weighted convolutional features.**

I. Introduction

VISUAL tracking is one of the important problems in computer vision [1]–[5]. It has numerous applications in video surveillance, UAV navigation, and action recognition. Although great progress [6]–[10] has been made in the past decade, visual tracking remains a challenging problem due to challenging factors including severe occlusion, background clutter, motion blur, illumination change, and scale change.

Recently, visual tracking methods via correlation filter have achieved promising tracking results in terms of both efficiency and robustness [11]–[16]. The key process of trackers based on correlation filter is that an online filter is employed to locate the target in consecutive frame by finding the position with maximum response. Dense image patches are extracted by vertical and horizontal cyclic shifts of a base image patch. Convolution operation in time domain is implemented in frequency domain by elementwise multiplication, which reduces the computation complexity greatly leading to real-time visual tracker. Bolme *et al.* [17] present a minimum output sum of squared error filter over gray-value channel for fast tracking. Henriques *et al.* [18] propose a high-speed tracking method with kernelized correlation filters (KCF), which exploits multiple channel Hog features. Danelljan *et al.* [19] demonstrate that sophisticated color features can achieve competitive tracking performance under the framework of correlation filter. Mueller *et al.* [20] adopt global context for training correlation filter and improve tracking performance under low computational cost. Bibi *et al.* [21] build a generic framework for changing target response from frame to frame, which can realize significant overall performance improvement.

Despite the significant progress has been made under the framework of correlation filter, traditional visual object tracking based on correlation filter utilizing hand-crafted features (e.g., gray, Hog, and CN) [22]–[25] cannot effectively deal with severe occlusion, fast motion, and out of views appeared in videos. In order to deal with these problems, convolutional neural networks features from different layers are adopted in correlation filter and perform favorably against the state-of-the-art methods on some challenging benchmark sequences. Wang *et al.* [26] develop a deep learning tracker using an automatically learned image representation. Hong *et al.* [27] propose an online tracking method under the framework of Bayesian filtering combining the last convolutional layer of CNN and the target-specific saliency map. Ma *et al.* [28] explore hierarchical convolutional features from VGG-Net for robust visual tracking. Qi *et al.* [29] present a novel hedge algorithm to hedge different weak CNN tracker considering historical tracking performance of each weak tracker. Bertinetto *et al.* [30] build an asymmetric siamese network for online tracking which obtains better tracking performance.

In this letter, we develop a novel loss function to assess the responses from different layers and give different weight parameters according to the value of loss function. The main contributions of our work are fourfold:

Manuscript received February 16, 2018; revised March 12, 2018; accepted March 21, 2018. Date of publication March 26, 2018; date of current version April 9, 2018. This work was supported in part by A Project of Shandong Province Higher Educational Science and Technology Program under Grant J17KA088 and Grant J16LN02, in part by the Natural Science Foundation of Shandong Province under Grant ZR2015FL009, and in part by the Key Research and Development Plan of Shandong Province under Grant 2016GGX101023. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Jun Fang. *(Corresponding author: Haijun Wang.)*

H. Wang is with the College of Civil Aviation, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China and also with the Aviation Information Technology Research and Development, Binzhou University, Binzhou 256603, China (e-mail: whjlym@163.com).

H. Ge and G. Chen are with the College of Civil Aviation, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China (e-mail: allenge@nuaa.edu.cn; chenguo_nuaa@163.com).

S. Zhang and Y. Du are with the Aviation Information Technology Research and Development, Binzhou University, Binzhou 256603, China (e-mail: zsykyx@163.com; duyujie442@163.com).

Color versions of one or more of the figures in this letter are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LSP.2018.2819622

1) We develop a novel visual tracking algorithm which combines three weak CNN trackers from hierarchical convolution layers into a stronger one.
2) We propose a semiadaptive weighted convolutional features (SACF) algorithm for visual tracking by considering performance of each weak tracker.
3) We adopt a novel detection module to recover the target from tracking failures and a target pyramid to deal with scale change.
4) Extensive experimental tracking results on OTB-2015 [31] and VOT2016 [32] demonstrate that our method can obtain the state-of-the-art performance.

II. Proposed Algorithm

In this section, we give the merits of hierarchical convolutional features by combining three weak CNN trackers into a stronger CNN tracker with semiadaptive weighted parameters.

A. Hierarchical CNN Features

Recent years have witnessed the advantages of CNN and numerous CNN models, such as AlexNet [33], CaffeNet [34], VGG [35], and RESNET [36], have been developed for image classification, object detection, and image segmentation. In this letter, we adopt convolution features from VGG-Net-19 to describe target appearance. It has been demonstrated that features extracted from different layers can provide different levels of abstract.

B. Weak CNN Trackers

Let $\mathbf{x}$ denote the lth layer of feature vector with the size $M \times N \times D$, where $M, N,$ and D stand for the width, height, and the number of channels, respectively. Training samples are generated by the cyclic shift of $\mathbf{x}$ and each sample $\mathbf{x}_{m,n}$ has a two-dimensional (2D) Gaussian function label $\mathbf{y}_{m,n}$, where $m, n \in \{0, 1, ..., M-1\} \times \{0, 1, ..., N-1\}$. The correlation filter $\mathbf{w}$ is learned by training the following minimization function [28]:

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \sum_{m,n} \|\mathbf{w} \cdot \mathbf{x}_{m,n} - \mathbf{y}_{m,n}\|^2 + \lambda\|\mathbf{w}\|_2^2 \tag{1}$$

where $\lambda > 0$ is a constant parameter. Utilizing fast Fourier transform (FFT), the closed-form solution of learned filter on the dth ($d \in \{1, ..., D\}$) channel is given by $\mathbf{W}^d = \frac{\mathbf{Y}\odot\bar{\mathbf{X}}^d}{\sum_{n=1}^{D}\mathbf{X}^i\odot\bar{\mathbf{X}}^i+\lambda}$. $\mathbf{W}, \mathbf{Y},$ and $\mathbf{X}$ stand for the Fourier transformation forms. $\bar{\mathbf{X}}$ means complex conjugation. $\odot$ is the elementwise multiplication.

When a new frame comes, a new image patch $\mathbf{z} \in M \times N \times D$ from the lth layer is cropped out at the target location of previous frame. Then the response map of the lth layer can be computed by $f_l = \mathscr{F}^{-1}\left(\sum_{d=1}^{D}\mathbf{W}^d \odot \mathbf{Z}^d\right)$. $\mathscr{F}^{-1}$ represents the inverse FFT. Then the new target position on the lth layer is estimated by searching the maximum value of the response map.

C. Semiadaptive Strong CNN Trackers

Let $f_l(m, n)$ denotes the value of response map at the position (m, n) from the lth layer. The optimal target position (m^*, n^*) is estimated by finding the maximum value of three weighted response map, $\arg\max_{m,n} \sum_{l=3,4,5} \beta_l f_l(m, n)$, where $f_3, f_4,$ and f_5 indicate the correlation response map from the conv3-4, conv4-4, and conv5-4 layers, respectively. $\beta_3, \beta_4,$ and β_5 denote weight parameters. As the correlation response map of the conv5-4 layer encodes the semantics information and the output of conv3-4, conv4-4 give fine-grained details, it can be seen from Fig. 1 that the performance by the conv5-4 layer is better than the conv3-4, conv4-4 layer in terms of distance precision and overlap success plots on OTB-2013 [37], and we empirically set the initial weight $\beta_3, \beta_4,$ and β_5 to (0.05, 0.5, 1). Fig. 2 demonstrates that the correlation features from the conv5-4 layer do not always perform better than the other layers. Thus, the weight assigned to different layers should not be fixed and will be changed with tracking performance by different layers.

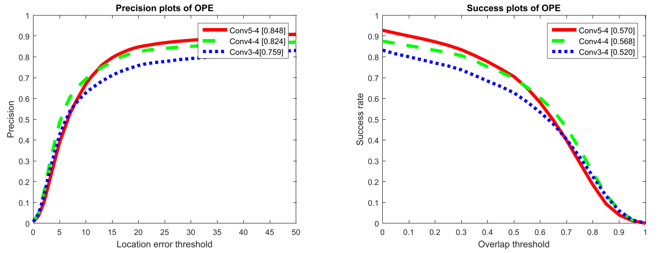

Fig. 1. Precision plots and success plots of OPE by the conv3-4, conv4-4, and conv5-4 layers from VGG-Net-19 on OTB-2013.

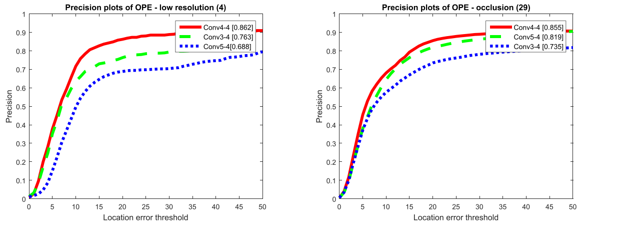

Fig. 2. Precision plots of OPE with low resolution and occlusion attributes by the conv3-4, conv4-4, and conv5-4 layers from VGG-Net-19 on OTB-2013.

In order to assess the correlation filter $\mathbf{w}_l$ from different layers thoroughly, we define a loss function ξ which was originally applied in partial-labeled learning problem [38]. Each correlation filter $\mathbf{w}_l$ of the lth layer can be considered as a nonparameter distribution and the corresponding response map $f_l(m, n)$ is viewed as the possibility of sample to be the target at the position (m, n). Define two labels ω_1 and ω_2, the normalized possibility of response map $f_l(m, n)$ is given by

$$P_l(\omega_1|f_l(m,n)) = \begin{cases} f_l(m,n), & if\ 0 \le f_l(m,n) \le 1 \\ 0, & f_l(m,n) < 0 \\ 1, & f_l(m,n) > 1 \end{cases} \tag{2}$$

and $P_l(\omega_2|f_l(m, n)) = 1 - P_l(\omega_1|f_l(m, n))$, where ω_1 and ω_2 denote the target sample and the nontarget sample, separately. Thus, the loss function ξ_l of the lth layer is defined as follows:

$$\xi_l = -L(f_l(m,n)) + \mu H(\mathbf{I}|f_l(m,n)) \tag{3}$$

where $\mathbf{I} = \{\omega_1, \omega_2\}$ is the set of labels. L is the log likelihood of $P(\omega_1|f_l(m, n))$ and is defined as

$$L(f_l(m,n)) = \max_{m,n} \log(P(\mathbf{I}_l|f_l(m,n))). \tag{4}$$

The entropy term $H(\mathbf{I}|f_l(m,n))$ is formulated as

$$H(\mathbf{I}|f_l(m,n)) = -\frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}\sum_{i=1}^{2} \times P(\omega_i|f_l(m,n))\log P(\omega_i|f_l(m,n)). \quad (5)$$

The first term in (3) is consistent with the correlation response map. The position with the smaller value of $-L$ may be the location of target. The second term in (3) favors the distribution of correlation response map with low ambiguity. For example, if the correlation response map contains two possible labels, the entropy term gives a high confidence score to one label and gives a low confidence value to another label, which is more favored than the entropy term giving the same confidence score to both labels. Thus, if the correlation response map of the lth layer is more reliable, the corresponding loss function should be smaller. Finally, we compute the loss function of each layer. The correlation response map with minimum value is assigned to higher weight and vice versa, which means that the weight parameters $(\beta_3, \beta_4, \beta_5)$ in $\arg\max_{m,n} \sum_{l=3,4,5} \beta_l f_l(m,n)$ are not fixed. Then the final response is given by $1 \times \frac{f_{l_1}(m,n)}{\max(f_{l_1})} + 0.5 \times \frac{f_{l_2}(m,n)}{\max(f_{l_2})} + 0.05 \times \frac{f_{l_3}(m,n)}{\max(f_{l_3})}$. The subscripts $l_1, l_2, l_3 \in \{3,4,5\}$ change according to the value of loss function.

D. Model Update

In order to make our method more adaptive to severe target variation, we update the correlation filter $\mathbf{W}^d$ by interpolating the new model at the new target position with those models from the previous frame [28]

$$\mathbf{W}_t^d = \frac{(1-\eta)\mathbf{A}_{t-1}^d + \eta\mathbf{Y} \odot \bar{\mathbf{X}}_t^d}{(1-\eta)\mathbf{B}_{t-1}^d + \eta\mathbf{X}_t^i \odot \bar{\mathbf{X}}_t^i + \lambda} \quad (6)$$

where η means the update parameter and the subscript t stands for frame index.

E. Online Detector and Scale Adaptation

We introduce an online random fern classifier to recover the target from tracking failures caused by severe occlusion [39], [40]. If the maximum value of final response is less than a threshold Γ, we train an online fern classifier to detect the whole frame with sliding windows. The best classifier is obtained by $\arg\max_{c_i} \prod_{k=1}^{M} P(F_k|C = c_i)$. Here, c is the indicator of class labels. $i \in \{0,1\}$. F_k means the kth fern and $P(F_k|C = c_i)$ represents the conditional probability function. We use the Hog features to construct a target pyramid for scale estimation. Let $W \times H$ be the target size and N represents the number of scales $S = \{a^n| - \frac{N-1}{2}, -\frac{N-3}{2}, \ldots, \frac{N-1}{2}\}$. Here, $a = 1.03$. The best scale $\hat{s}$ can be estimated by $\hat{s} = \arg\max_{s}(\max(\hat{y}_1), \max(\hat{y}_2)), \ldots, \max(\hat{y}_S))$. $\hat{y}_s$ means the response map with the s scale.

III. Experiments

In this section, we give the implementation details of the proposed approach, and report the comparison results on quantitative evaluation and qualitative evaluation. We implement our method in MATLAB on an Inter i5-4570 3.20 GHz CPU with 8 G RAM. We utilize the MatConvNet toolbox for feature extraction. The fully connected layers are removed and the outputs of conv3-4, conv4-4, and conv5-4 layers from VGG-Net-19 are adopted as features. The regularization parameter λ is set to 10^{-4}. The learning rate η and the threshold Γ are set to 0.01 and 0.5, respectively. To prove the effectiveness of our method, we compare it with nine sate-of-the-art trackers including HCFT [28], HDT [29], MEEM [41], SRDCF [42], KCF [43], staple [44], SAMF [45], DSST [46], and LCT [40].

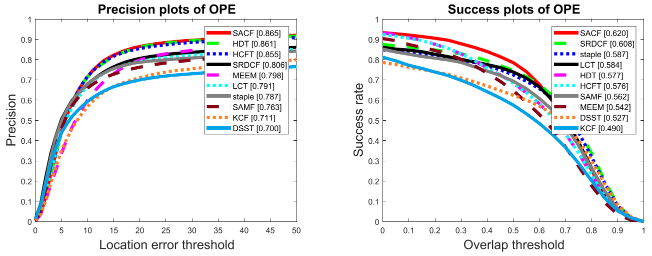

Fig. 3. Precision plots and success plots of OPE on OTB-2015.

A. Quantitative Evaluation

We use the protocol for tracker evaluation from OTB-2013. One-pass evaluation (OPE) is utilized to measure the overall tracking performance. Fig. 3 shows the precision plots and success plots on OTB-2015 datasets. The precision plot shows the center location error between the estimated position and the ground truth for each frame. The success plot gives the overlap score at the thresholds varied from 0 to 1. It is obvious that the proposed SACF method achieves the best results both in distance precision for OPE and in the area under curve for OPE. Fig. 4 shows the results on OTB-2015 datasets of 11 challenging attributes. It can be seen that the proposed method performs better than HCFT and HDT with the use of the same CNN model. The last plot of Fig. 4 gives the comparison of our method with failure detection and without failure detection on OTB-2015. It is obvious that our method with failure detection performs better in most of the 11 challenging attributes except for low resolution. The tracking speed of our method is 1 fps which is almost the same as HCFT.

B. Qualitative Evaluation

We select two challenging sequences to demonstrate the effectiveness of our SACF method, including lemming and soccer. Fig. 5 gives the tracking results from SACF, HCFT, HDT, KCF, MEEM, LCT, staple, SAMF, DSST, and SRDCF. With the help of CNN feature, our proposed method, HCFT and HDF are able to localize the target more precisely than KCF, MEEM, LCT, staple, SAMF, DSST, and SRDCF methods with low-level handcrafted features. In the lemming sequence, HCFT, HDT, KCF, staple, SAMF, DSST, and SRDCF lose the target in the 380th frame because of the severe occlusion and background clutter. Our SACF method is able to keep up with the target movement and shows stable and more accurate tracking performance. The soccer sequence has severe illumination variation, occlusion, and background clutter. The proposed SACF method and HCFT

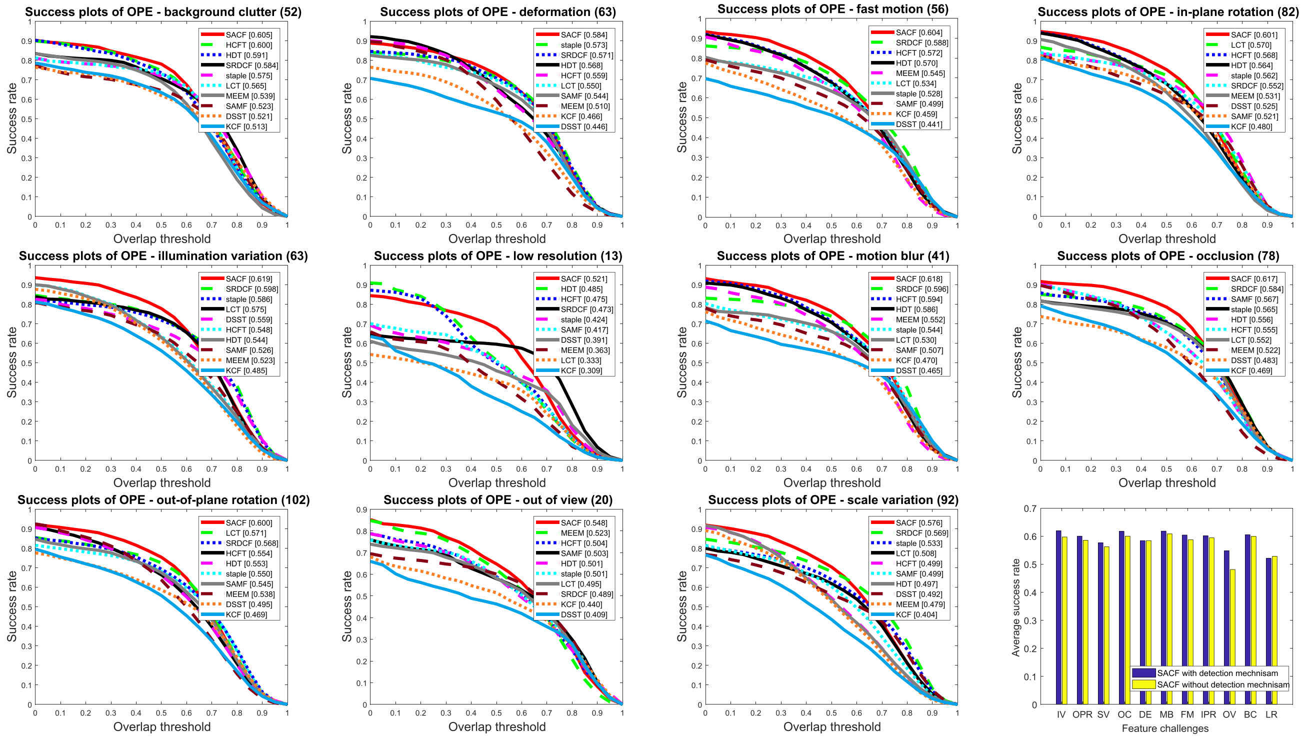

Fig. 4. Success plots of OPE of 11 challenging attributes on OTB-2015.

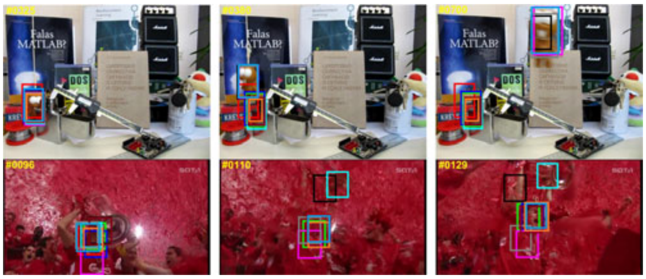

Fig. 5. Selected tracking results of our SACF method against nine state-of-the-art trackers in sequences lemming, soccer, respectively. The predicted bounding boxes of five trackers are shown with different colors, including SACF (turquoise), HCFT (red), HDT (green), KCF (blue), MEEM (black), LCT (magenta), staple (cyanine), SAMF (gray), DSST (dark red), SRDCF (orange).

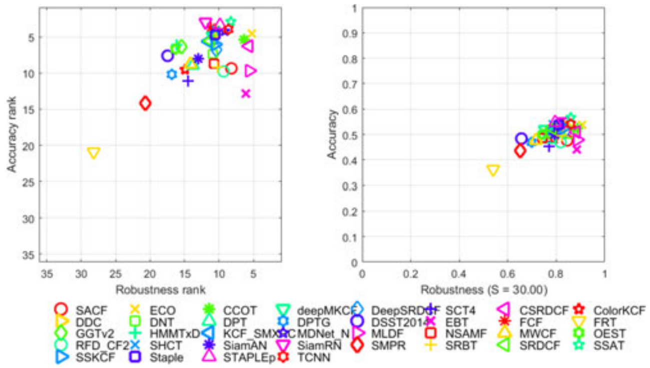

Fig. 6. Robustness-accuracy ranking plots under the baseline on the VOT2016.

are able to localize the target precisely, while HDT drift a little at frame 110 and recovers again at the following frame. Although HDT adopts CNN features, it usually loses the target on some challenging sequences. HCFT cannot obtain the same tracking performance as our method, and drifts a little at some frame (frame 380 of lemming and frame 110 of soccer), which makes HCFT give a lower contrast index in OPE. Since our method sets the different weights of three weak CNN-based trackers considering their corresponding tracking performance, with the use of redetect target and scale adaptation scheme, SACF performs well in different environments and can overcome these challenges much better than other trackers.

C. Evaluation on VOT2016

We compare our SACF method with 35 other methods mentioned on VOT2016. Fig. 6 shows the comparison results on VOT2016. Trackers closer to the top-right of the plot perform better. Although our proposed method just obtains the sixth best robustness and ranks in the middle of all algorithms in terms of accuracy, our SACF method performs better than most of the other methods taking account of both the robustness and accuracy simultaneously.

IV. Conclusion

In this letter, we propose a novel and effective tracking algorithm for robust visual tracking. By investigating the performance among different tracking methods based on different features from different layers, we set different weight parameters for different weak CNN trackers and obtain a stronger one. We further introduce an online detector to recover the target from tracking failure and build a target pyramid to deal with scale variation. Extensive experiments on OTB-2015 and VOT2016 show that our method can achieve more favorable tracking performance than several other methods.

References

[1] Z. Chi, H. Li, H. Lu, and M. Yang, "Dual deep network for visual tracking," *IEEE Trans. Image Process.*, vol. 26, no. 4, pp. 2005–2015, Apr. 2017.

[2] K. Zhang, L. Zhang, and M. Yang, "Fast compressive tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 10, pp. 2002–2015, Oct. 2014.

[3] C. Sun, D. Wang, and H. Lu, "Occlusion-aware fragment-based tracking with spatial-temporal consistency," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3814–3825, Aug. 2016.

[4] M. Danelljan, A. Robinson, F. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Proc. Eur. Conf. Comput. Vision*, 2016, pp. 472–488.

[5] M. Danelljan, G. Bhat, F. Khan, and M. Felsberg, "ECO: Efficient convolution operators for tracking," in *Proc. Comput. Vision Pattern Recog.*, 2017, pp. 6931–6939.

[6] S. Zhang, X. Lan, Y. Qi, and P. Yuen, "Robust visual tracking via basis matching," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 3, pp. 421–430, Mar. 2017.

[7] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *Proc. Int. Conf. Comput. Vision Workshop*, 2015, pp. 621–629.

[8] A. Lukezic, T. Vojir, L. Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter with channel and spatial reliability," in *Proc. Comput. Vision Pattern Recog.*, 2017, pp. 4847–4856.

[9] S. Zhang, Y. Qi, F. Jiang, X. Lan, P. Yuen, and H. Zhou, "Point-to-set distance metric learning on deep representations for visual tracking," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 1, pp. 187–198, Jan. 2018.

[10] S. Zhang, X. Lan, H. Yao, H. Zhou, D. Tao, and X. Li, "A biologically inspired appearance model for robust visual tracking," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2357–2370, Oct. 2017.

[11] H. Galoogahi, A. Fagg, and Si. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proc. Int. Conf. Comput. Vision*, 2017, pp. 1135–1143.

[12] F. Li, Y. Yao, P. Li, D. Zhang, W. Zuo, and M. Yang, "Integrating boundary and center correlation filters for visual tracking with aspect ratio variation," in *Proc. Int. Conf. Comput. Vision*, 2017, pp. 2001–2009.

[13] J. Valmadre, L. Bertinetto, F. Henriques, and A. Vedaldi, "End-to-end representation learning for Correlation Filter based tracking," in *Proc. Comput. Vision Pattern Recog.*, 2017, pp. 2805–2813.

[14] S. Zhang, H. Zhou, F. Jiang, and X. Li, "Robust visual tracking using structurally random projection and weighted least squares," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 11, pp. 1749–1760, Nov. 2015.

[15] Q. Hu, Y. Guo, Y. Chen, and J. Xiao, "Correlation filter tracking: Beyond an open-loop system," in *Proc. Brit. Mach. Vision Conf.*, 2017, pp. 1–12.

[16] K. Zhang, L. Zhang, M. Yang, and D. Zhang, "Fast tracking via spatio-temporal context learning," in *Proc. Eur. Conf. Comput. Vision*, 2014, pp. 127–141.

[17] D. Bolme, J. Beveridge, B. Draper, and Y. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. Comput. Vision Pattern Recog.*, 2010, pp. 2544–2550.

[18] J. 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.

[19] M. Danelljan, F. Khan, M. Felsberg, and J. Weijer, "Adaptive color attributes for real-time visual tracking," in *Proc. Comput. Vision Pattern Recog.*, 2014, pp. 1090–1097.

[20] M. Mueller, N. Smith, and B. Ghanem, "Context-aware correlation filter tracking," in *Proc. Comput. Vision Pattern Recog.*, 2017, pp. 1396–1404.

[21] A. Bibi, M. Mueller, and B. Ghanem, "Target response adaptation for correlation filter tracking," in *Proc. Eur. Conf. Comput. Vision*, 2016, pp. 419–433.

[22] H. Jiang, J. Li, D. Wang, and H. Lu, "Multi-feature tracking via adaptive weights," *Neurocomput.*, vol. 207, pp. 189–201, 2016.

[23] D. Huang, L. Luo, M. Wen, Z. Chen, and C. Zhang, "Enable scale and aspect ratio adaptability in visual tracking with detection proposals," in *Proc. Brit. Mach. Vision Conf.*, 2015, pp. 185.1–185.12.

[24] S. Liu, T. Zhang, X. Cao, and C. Xu, "Structural correlation filter for robust visual tracking," in *Proc. Conf. Comput. Vision Pattern Recog.*, 2016, pp. 4312–4320.

[25] P. Li, D. Wang, L. Wang, and H. Lu, "Deep visual tracking: Review and experimental comparison," *Pattern Recog.*, vol. 76, pp. 323–338, 2018.

[26] N. Wang and D. Yeung, "Learning a deep compact image representation for visual tracking," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 809–817.

[27] S. Hong, T. You, S. Kwak, and B. Han, "Online tracking by learning discriminative saliency map with convolutional neural network," in *Proc. 32nd Int. Conf. Mach. Learn.*, 2015, pp. 597–606.

[28] C. Ma, J. Huang, X. Yang, and M. Yang, "Hierarchical convolutional features for visual tracking," in *Proc. Int. Conf. Comput. Vision*, 2015, pp. 3074–3082.

[29] Y. Qi *et al.*, "Hedged deep tracking," in *Proc. Comput. Vision Pattern Recog.*, 2016, pp. 4303–4311.

[30] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. Torr, "Fully-convolutional siamese networks for object tracking," in *Proc. Eur. Conf. Comput. Vision*, 2016, pp. 850–865.

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

[32] M. Kristan *et al.*, "The visual object tracking VOT2016 challenge results," in *Proc. Eur. Conf. Comput. Vision Workshop*, 2016, pp. 777–823.

[33] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[34] Y. Jia *et al.*, "CAFFE: Convolutional architecture for fast feature embedding," in *Proc. 22th ACM Int. Conf. Multi.*, 2014, pp. 675–678.

[35] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional networks," in *Proc. Brit. Mach. Vision Conf.*, 2014, pp. 1–12.

[36] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. Comput. Vision Pattern Recog.*, 2016, pp. 770–778.

[37] Y. Wu, J. Lim, and M. Yang, "Online object tracking: A benchmark," in *Proc. Comput. Vision Pattern Recog.*, 2013, pp. 2411–2418.

[38] Y. Grandvalet and Y. Bengio, "Semi-supervised learning by entropy minimization," in *Proc. 17th Int. Conf. Neural Inf. Proc. Syst.*, 2004, pp. 529–536.

[39] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1409–1422, Jul. 2012.

[40] C. Ma, X. Yang, C. Zhang, and M. Yang, "Long-term correlation tracking," in *Proc. Comput. Vision Pattern Recog.*, 2015, pp. 5388–5396.

[41] J. Zhang, S. Ma, and S. Sclaroff, "MEEM: Robust tracking via multiple experts using entropy minimization," in *Proc. Eur. Conf. Comput. Vision*, 2014, pp. 188–203.

[42] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proc. Int. Conf. Comput. Vision*, 2015, pp. 4310–4318.

[43] J. Henriques, C. Rui, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Proc. Eur. Conf. Comput. Vision*, 2012, pp. 702–715.

[44] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, "Staple: Complementary learners for real-time tracking," in *Proc. Comput. Vision Pattern Recog.*, 2016, pp. 1401–1409.

[45] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *Proc. Eur. Conf. Comput. Vision Workshop*, 2014, pp. 254–265.

[46] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proc. Brit. Mach. Vision Conf.*, 2014, pp. 65.1–65.11.