# LEARNING CASCADED SIAMESE NETWORKS FOR HIGH PERFORMANCE VISUAL TRACKING

*Peng Gao, Liyi Xiao, Fei Wang**

Department of Electronic and Information Engineering
Harbin Institute of Technology, Shenzhen

## ABSTRACT

Visual tracking is one of the most challenging computer vision problems. In order to achieve high performance visual tracking in various negative scenarios, a novel cascaded Siamese network is proposed and developed based on two different deep learning networks: a matching subnetwork and a classification subnetwork. The matching subnetwork is a fully convolutional Siamese network. According to the similarity score between the exemplar image and the candidate image, it aims to search possible object positions and crop scaled candidate patches. The classification subnetwork is designed to further evaluate the cropped candidate patches and determine the optimal tracking results based on the classification score. The matching subnetwork is trained offline and fixed online, while the classification subnetwork performs stochastic gradient descent online to learn more target-specific information. To improve the tracking performance further, an effective classification subnetwork update method based on both similarity and classification scores is utilized for updating the classification subnetwork. Extensive experimental results demonstrate that our proposed approach achieves state-of-the-art performance in recent benchmarks.

***Index Terms*—** Visual tracking, object detection, Siamese networks, cascaded learning

## 1. INTRODUCTION

Visual tracking is a most fundamental research issue in the field of computer vision, and it is widely developed in numerous applications, such as video surveillance, drone tracking, self-driving vehicle, human-computer interaction, auxiliary medical diagnosis, and many others [1, 2]. Normally, tracking task is to estimate the trajectory of an arbitrary target in an image sequence, given only its initial location at the first frame. Despite the excellent results achieved by numerous tracking approaches [3, 4, 5, 6, 7, 8] in the past decades, visual tracking is still a challenging problem owing to complicated factors like fast motions, background clutters, motion blurs, deformations, illumination variations, low resolution, occlusions, out of views, scale variations, etc.

*Corresponding Author.

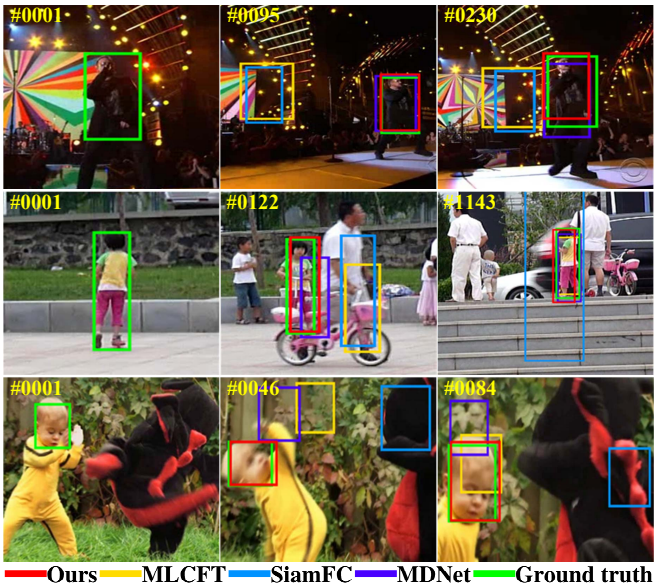


**Fig. 1**. Comparison of our proposed tracking approach with three state-of-the-art CNN based trackers: MLCFT [9], SiamFC [6] and MDNet [5] on three example sequences from OTB2015 benchmark [10], respectively. We also present the ground-truth bounding boxes of these example sequences. Best viewed in color.

In recent years, with the tremendous development of deep learning technology [11, 12, 13, 14], convolutional neural networks (CNN) have attracted increasing attention in the tracking community. Compared with the conventional handcrafted features based trackers [3, 4, 15, 16, 17], CNN based trackers [18, 19, 20, 5, 21, 22] can easily obtain more competitive tracking performance in multiple benchmarks [23, 10, 24]. In general, existing CNN based tracking approaches can be divided into two categories, i.e., matching based trackers and classification based trackers. The former is always pre-trained offline on the video object detection dataset of the ImageNet [25]. During tracking, it matches the candidates with the exemplar by correlating deep features and does not need online updating. In contrast, the classification based tracking approach transfers a pre-trained network as the classifier and then performs online updating by adding some particular

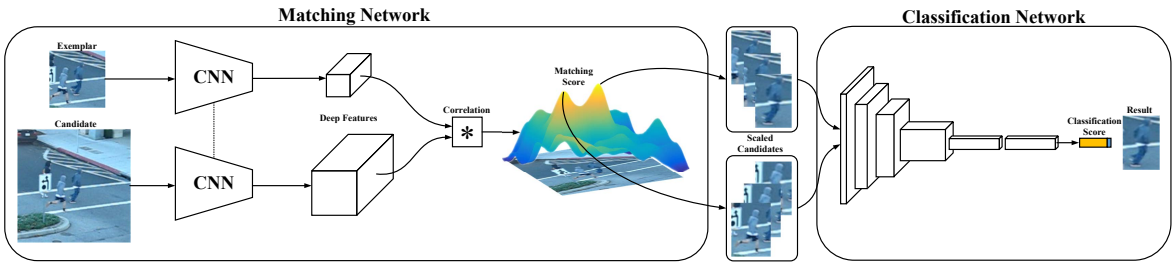


**Fig. 2**. The overall framework of our proposed approach.

layers [5]. Although all the CNN based trackers above mentioned have obtained impressive tracking results, there is still great potential to enhance performance further.

In this paper, we propose a novel cascaded Siamese network for high performance visual tracking by integrating both the matching and classification networks. First, a matching subnetwork is exploited to measure the similarity between candidate image and exemplar image and crop scaled candidate patches based on the similarity score. Then, a classification subnetwork which is cascaded with the matching subnetwork learns a target-specific classification scheme online to further determine the optimal tracking results among all scaled candidate patches based on the classification score. Finally, both similarity and classification scores are combined together to indicate whether the classification subnetwork should be updated online or not.

Our main contributions are three folds and summarized as follows:

- We propose a novel cascaded Siamese network for high performance visual tracking, which consists of a matching subnetwork and a classification subnetwork.
- We utilize an effective model update method to determine the necessity for classification subnetwork online updating.
- We conduct extensive experiments on several recent tracking benchmarks, our proposed approach achieves surprisingly good performance both in terms of accuracy and robustness, as shown in Fig. 1.

## 2. ALGORITHMIC OVERVIEW

The overall framework of our proposed approach is shown in Fig. 2. The proposed approach consists of a matching subnetwork for target localization and scaled candidate patches creation and a classification subnetwork for optimal tracking results determination. During the tracking process, an exemplar image $\mathbf{x}$ of size $127 \times 127$ and a candidate image $\mathbf{z}$ of size $255 \times 255$ both centered around the previous position of the target are first fed into the matching subnetwork. The matching subnetwork imitates the fully-convolutional Siamese architecture [6], and the similarity between the exemplar image and the candidate image is estimated by calculating the cross-correlation based on their deep features. Then, the possible target positions are chosen by searching the maximum similarity scores, and scaled candidate patches centered at all possible target positions are cropped on the candidate image. Here, the scaling method is similar to that of DSST tracker [16]. Next, the scaled candidate patches are resize to $107 \times 107$ and classified into foreground or background by the classification subnetwork, and the patch with the highest foreground score will be determined as the optimal tracking result. Finally, we update the classification subnetwork online based on the combination of both similarity and classification score corresponding to the optimal tracking result.

## 3. THE PROPOSED APPROACH

### 3.1. Matching Subnetwork

In our matching subnetwork, we adopt a fully-convolutional Siamese network which is pre-trained offline with a large video object detection dataset [25] in an end-to-end manner as the deep feature extractor [6]. Our aim is to learn a function $f(\mathbf{z}, \mathbf{x}) = g(\varphi(\mathbf{z}), \varphi(\mathbf{x}))$ to compare the exemplar image $\mathbf{x}$ with the candidate image $\mathbf{z}$ of the same size, where $\varphi(\mathbf{z})$ and $\varphi(\mathbf{x})$ represent the deep feature maps and $g$ is a similarity metric. We utilize a cross-correlation layer to measure the similarity between the output deep features,

$$f(\mathbf{z}, \mathbf{x}) = \varphi(\mathbf{z}) * \varphi(\mathbf{x}) + b \cdot \mathbb{1} \tag{1}$$

where $*$ denotes the cross-correlation operation, and $b \cdot \mathbb{1}$ indicates the bias. Thus, the output $f(\mathbf{z}, \mathbf{x})$ indicates a similarity score map corresponding to the exemplar image compared to the candidate image.

The localization of the target can be estimated at the highest peak on the similarity score map. However, since a video stream always undergoes variations such as fast motion, illumination variation and occlusion, the similarity measurement may be disturbed by similar objects or background noises in the candidate image as shown in Fig. 2, and there possibly exist multiple peaks on the similarity score map and the target may locate at one of them. If we estimate the target at

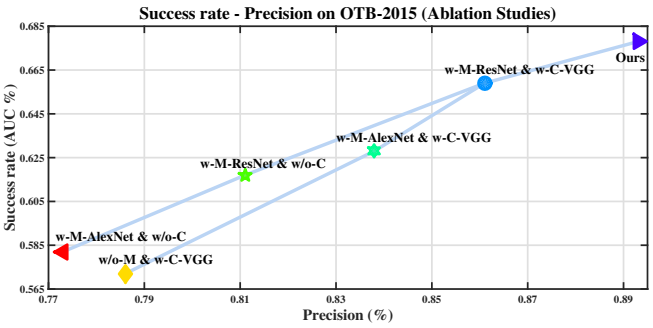


**Fig. 3**. Ablation studies plot on OTB-2015 [10]. In the legend, **M** denotes the choice of matching subnetwork and **C** denotes the choice of classification subnetwork.

wrong peaks, it will leads to inaccurate localization and tracking drift. To solve this problem, we use the classification subnetwork to further determine both the optimal target position and size among all the peaks.

### 3.2. Classification Subnetwork

In Section 3.1, we obtain a similarity score map by cross-correlating the output deep features of the feature extractor. Since the similarity score map may not be reliable enough, we treat peaks whose ratio between its score and that of the highest peak exceeding a certain threshold $\gamma_p$ as possible target positions, and the corresponding patches centered at these positions are cropped and scaled as mentioned in Section 2. After that, a series of scaled candidate patches can be obtained. Thus, we exploit a classification subnetwork for optimal tracking results determination.

The classification subnetwork architecture is similar to that of MDNet [5] which has three convolutional layers, two fully connected layers and a binary classification layer with softmax cross-entropy loss to output the probabilities of target and background classes, as shown in Fig. 2.

Finally, the candidate patch with the highest classification score in the target class will be selected as the optimal tracking result.

### 3.3. Updating Method

During tracking, the parameter of the matching subnetwork are fixed, and all the classification layer and the fully connected layers of the classification subnetwork are fine-tuning online to adapt to variations based on optimal tracking results in the current frame. However, the optimal tracking results are not always reliable for classification subnetwork updates. Inappropriate updates may break down the classification subnetwork due to the ambiguous tracking results.

In order to alleviate this issue, we utilize a simple but effective method for classification subnetwork updating. Assume the similarity and classification scores of current optimal tracking results are $S_M^t$ and $S_C^t$ respectively, and the historical scores of previous $n$ frames are $S_M = \frac{1}{n}\sum_{T=1}^{n} S_M^{t-T}$ and $S_C = \frac{1}{n}\sum_{T=1}^{n} S_C^{t-T}$. If there are no other peaks on the similarity score map that exceed a ratio $\gamma_m$ of the highest peak value, the classification subnetwork will be updated directly based on the current optimal tracking result. In contrast, if there has one or more peaks exceeds the ratio $\gamma_p$ of the highest peak value, we compare both similarity and classification scores with the historical scores. Only when these two scores $S_M^t$ and $S_C^t$ are great than $\gamma_m$ and $\gamma_c$ of their corresponding historical score $S_M$ and $S_C$ respectively, we update the last three layers of our classification subnetwork.

## 4. EXPERIMENTS

In this section, we conduct extensive experiments to validate the effectiveness of our proposed cascaded Siamese network. We first detail the implementation of our approach. Then, we investigate the impact of the architecture of the matching and classification subnetworks as well the update method. Finally, we compare our approach with nine state-of-the-art trackers including ECO [7], CCOT [20], MLCFT [9], CACT [4], Staple [17], MDNet [5], SiamFC [6], KCF [3] and DSST [16] on three tracking benchmarks: OTB-2013 [23], OTB-2015 [10] and VOT-2016 [24]. The experiments on OTB benchmarks are exploiting two metrics: distance precision and overlap success rate, while the expected average overlap (EAO) is exploited in the VOT dataset.

### 4.1. Implementation Details

**Network Architecture.** In the matching subnetwork, we exploit ResNet [13] for deep feature extraction, which followed by a cross-correlation layer. The convolutional layers of the classification network are identical to the corresponding parts of VGG-M [12], the fully connected layers have 512 output units and the classification layer output 2 scores as described in MDNet [5].

**Offline Training.** For the training process of both matching and classification subnetworks, sample pairs are selected from the ImageNet video object detection dataset [25] with random interval. The exemplar and candidate images are picked from the same video. We first load the pre-trained networks to initialize our approach. Then, we apply stochastic gradient descent (SGD) with the learning rate set from $10^{-3}$ to $10^{-4}$ and the momentum of 0.9 to train the networks end-to-end, respectively. More details about the training methods can be found in [6] and [5].

**Online Tracking.** During the tracking process, we only update the parameters of the last three layers of the classification subnetwork, and others are fixed. The candidate image is cropped approximately four times the target size centered at the previous position. The certain thresholds $\gamma_p$, $\gamma_m$ and $\gamma_c$ are set to 0.75, 0.8 and 0.6, respectively. The number of historical frames $n$ is set to 6. Moreover, we exploit three scales

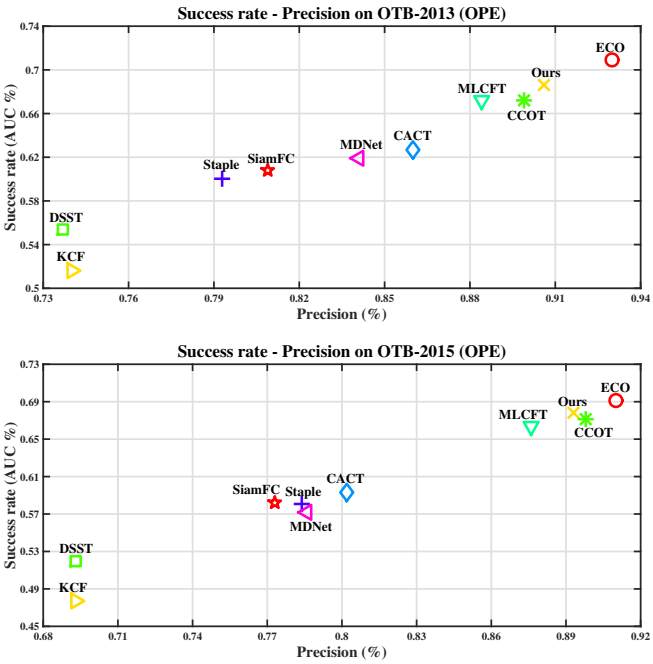


**Fig. 4**. Success rate-Precision ranking plots of our approach and nine state-of-the-art trackers on OTB-2013 [23] (top) and OTB-2015 [10] (bottom). The better performance a tracker achieves, the closer to the top-right corner of the graph.

$1.02^{\{-1,0,1\}}$ to crop candidate pathes at each possible target position.

Our approach is implemented using MXNet [26] on an Amazon EC2 instance with an Intel Xeon E5 CPU, 61GB RAM and a NVIDIA K80 GPU, 12GB VRAM. It is worth to mention that we retrained MDNet [5] on ImageNet [25] since the original MDNet is training with tracking videos that may cause unfair performance over other tracking approaches.

### 4.2. Ablation Studies

To verify the effectiveness of our designed matching and classification subnetwork as well the update method in our cascaded Siamese network, we conduct ablation studies on OTB-2015 benchmark. The result is shown in Fig. 3.

It is clear that the performances of all the variations which are implemented using the components indicated in the plot legend are not as good as our full approach, and each component in our tracking framework is helpful to improve performance. A noteworthy is only our final implementation, denoted by *Ours*, employs the update method.

### 4.3. Results on OTB

We show the success rate-precision ranking plots on OTB-2013 and OTB-2015 benchmarks [23, 10] in Fig. 4. It illustrates that the proposed tracker performs better than other

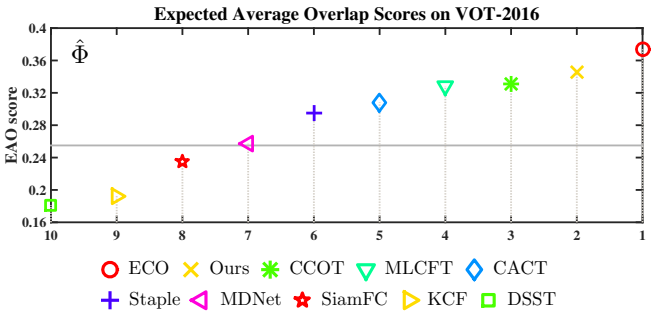


**Fig. 5**. EAO graph of our approach and nine state-of-the-art trackers on VOT-2016 [24]. The better performance a tracker achieves, the closer to the right of the graph.

re-detection trackers MLCFT and CACT, but is less effective than ECO which exploits continuous convolutional filters.

Overall, our approach attains surprisingly excellent performance both in terms of accuracy and robustness.

### 4.4. Results on VOT

We also evaluate our proposed approach on the VOT-2016 dataset [24] as shown in Fig. 5. The horizontal grey line indicates the state-of-the-art bound according to the VOT committee. Our tracker ranks second in overall performance evaluations based on the EAO measure. Specifically, the performance of our approach excels the CCOT [20] tracker which achieves the best results in the original VOT-2016 challenge.

SiamFC [6] and MDNet [5] are the baselines of the proposed approach. Compared to them, our tracker not only learns a matching subnetwork to search the possible target positions, but also benefits from the classification subnetwork to determine the optimal tracking results. What is more, the effective classification subnetwork updating method ensure the robustness of the tracker. Therefore, our cascaded Siamese network outperforms them with a large margin.

## 5. CONCLUSION

In this paper, we propose a cascaded Siamese network for high performance visual tracking. Our proposed approach consists of the matching subnetwork for similarity learning and the classification subnetwork for optimal target result determination. Extensive experiments on three recent tracking benchmarks demonstrate competing performance of the proposed tracker over a number of state-of-the-art approaches.

## 6. ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China under Grant No. 31701187, the Guangdong Provincial Science and Technology Planning Program under Grant No. 2016B090918047, and Promotional Credit from Amazon Web Service, Inc.

## 7. REFERENCES


[1] Alper Yilmaz, Omar Javed, and Mubarak Shah, "Object tracking: A survey," *ACM Computing Surveys*, vol. 38, no. 4, pp. 13, 2006.

[2] Arnold WM Smeulders, Dung M Chu, Rita Cucchiara, Simone Calderara, Afshin Dehghan, and Mubarak Shah, "Visual tracking: An experimental survey," *IEEE TPAMI*, vol. 36, no. 7, pp. 1442–1468, 2014.

[3] João F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista, "High-speed tracking with kernelized correlation filters," *IEEE TPAMI*, vol. 37, no. 3, pp. 583–596, 2015.

[4] Peng Gao, Yipeng Ma, Chao Li, Ke Song, Yan Zhang, Fei Wang, and Liyi Xiao, "Adaptive object tracking with complementary models," *IEICE Transactions on Information and Systems*, vol. E101-D, no. 11, 2018.

[5] Hyeonseob Nam and Bohyung Han, "Learning multi-domain convolutional neural networks for visual tracking," in *CVPR*, 2016.

[6] Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr, "Fully-convolutional siamese networks for object tracking," in *ECCV*, 2016.

[7] Martin Danelljan, Goutam Bhat, Shahbaz Fahad Khan, and Michael Felsberg, "Eco: Efficient convolution operators for tracking," in *CVPR*, 2017.

[8] Peng Gao, Yipeng Ma, Ke Song, Chao Li, Fei Wang, Liyi Xiao, and Yan Zhang, "High performance visual tracking with circular and structural operators," *Knowledge-Based Systems*, vol. 161, pp. 240–253, 2018.

[9] Yipeng Ma, Chun Yuan, Peng Gao, and Fei Wang, "Efficient multi-level correlating for visual tracking," in *ACCV*, 2018.

[10] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang, "Object tracking benchmark," *IEEE TPAMI*, vol. 37, no. 9, pp. 1834–1848, 2015.

[11] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NeurIPS*, 2012.

[12] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556v6*, 2015.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[14] Jie Hu, Li Shen, and Gang Sun, "Squeeze-and-excitation networks," in *CVPR*, 2018.

[15] Peng Gao, Yipeng Ma, Chao Li, Ke Song, Fei Wang, and Liyi Xiao, "A complementary tracking model with multiple features," in *IVPAI*, 2018.

[16] Martin Danelljan, Gustav Häger, Fahad Khan, and Michael Felsberg, "Accurate scale estimation for robust visual tracking," in *BMVC*, 2014.

[17] Luca Bertinetto, Jack Valmadre, Stuart Golodetz, Ondrej Miksik, and Philip HS Torr, "Staple: Complementary learners for real-time tracking," in *CVPR*, 2016.

[18] Peng Gao, Yipeng Ma, Ke Song, Chao Li, Fei Wang, and Liyi Xiao, "Large margin structured convolution operator for thermal infrared object tracking," in *ICPR*, 2018.

[19] Ran Tao, Efstratios Gavves, and Arnold W. M. Smeulders, "Siamese instance search for tracking," in *CVPR*, 2016.

[20] Martin Danelljan, Andreas Robinson, Fahad Shahbaz Khan, and Michael Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *ECCV*, 2016.

[21] Peng Gao, Yipeng Ma, Ruyue Yuan, Liyi Xiao, and Fei Wang, "Siamese attentional keypoint network for high performance visual tracking," *arXiv preprint arXiv:1904.10128*, 2019.

[22] Jack Valmadre, Luca Bertinetto, Joao Henriques, Andrea Vedaldi, and Philip H. S. Torr, "End-to-end representation learning for correlation filter based tracking," in *CVPR*, 2017.

[23] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang, "Online object tracking: A benchmark," in *CVPR*, 2013.

[24] Matej Kristan, Ales Leonardis, Jiri Matas, Michael Felsberg, and Roman Pflugfelder, "The visual object tracking vot2016 challenge results," in *ECCV*, 2016.

[25] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C Berg, and Fei-Fei Li, "Imagenet large scale visual recognition challenge," *IJCV*, vol. 115, no. 3, pp. 211–252, 2015.

[26] Tianqi Chen, Mu Li, Yutian Li, Min Lin, Naiyan Wang, Minjie Wang, Tianjun Xiao, Bing Xu, Chiyuan Zhang, and Zheng Zhang, "Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems," *arXiv preprint arXiv:1512.01274*, 2015.