EURASIP Journal on Image and Video Processing

RESEARCH Open Access

Learning attention for object tracking with adversarial learning network

Xu Cheng[1,2,3*], Chen Song[1], Yongxiang Gu[1] and Beijing Chen[1,2,3]

* Correspondence: xcheng@nuist.edu.cn
[1] School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China
[2] Jiangsu Key Laboratory of Big Data Analysis Technology, Nanjing University of Information Science and Technology, Nanjing 210044, China
[3] Engineering Research Center of Digital Forensics, Ministry of Education, Nanjing University of Information Science and Technology, Nanjing 210044, China

Abstract

Artificial intelligence has been widely studied on solving intelligent surveillance analysis and security problems in recent years. Although many multimedia security approaches have been proposed by using deep learning network model, there are still some challenges on their performances which deserve in-depth research. On the one hand, high computational complexity of current deep learning methods makes it hard to be applied to real-time scenario. On the other hand, it is difficult to obtain the specific features of a video by fine-tuning the network online with the object state of the first frame, which fails to capture rich appearance variations of the object. To solve above two issues, in this paper, an effective object tracking method with learning attention is proposed to achieve the object localization and reduce the training time in adversarial learning framework. First, a prediction network is designed to track the object in video sequences. The object positions of the first ten frames are employed to fine-tune prediction network, which can fully mine a specific features of an object. Second, the prediction network is integrated into the generative adversarial network framework, which randomly generates masks to capture object appearance variations via adaptively dropout input features. Third, we present a spatial attention mechanism to improve the tracking performance. The proposed network can identify the mask that maintains the most robust features of the objects over a long temporal span. Extensive experiments on two large-scale benchmarks demonstrate that the proposed algorithm performs favorably against state-of-the-art methods.

Keywords: Surveillance, Deep learning, Object tracking, Generative adversarial learning

1 Introduction

Nowadays, multimedia content (in particular image and video data) is being widely shared over the Internet due to the rapid development of network technologies and advent of high-end devices. Emerging technologies such as Cloud, Fog, Edge, SDN, Big Data, Internet of Things (IoT), and Deep Learning provide scalability, flexibility, agility, and ubiquity in terms of data acquisition, data storage, data management, and communications. Although a large number of multimedia forensic and security techniques have been proposed to protect multimedia data and devices and to support investigations of multimedia-related criminal cases and security incidents, a number of multimedia security issues have also emerged correspondingly, such as intelligent analysis for surveillance, copy-move forgery in digital images and videos, and biometric spoofing.

In recent year, artificial intelligence has been widely studied on solving a variety of difficult problems using deep learning network model, such as convolution neural networks for steganalysis and forensics, and generative adversarial networks for coverless steganagraphy. Surveillance technology for intelligent multimedia hiding and forensics has been a hot topic in multimedia security community. It is the basis of advanced video processing tasks such as follow-up steganography [1], data hiding [2], JPEG compressed [3], and object recognition [4] and is a necessary prerequisite for implementing high-level intelligent behavior analysis. Object tracking is one of the fundamental tasks in intelligent surveillance technology. The aim is to localize the object in a video sequence with a bounding box of the object at the first frame of the video. Although object tracking has made great progress in recent years and some effective algorithms have been proposed to solve challenging problems in specific scenarios, there are still exist many issues such as occlusion and illumination changes. So the topic deserves in-depth investigation, which is important in both academia and industry. Figure 1 shows intelligent surveillance applications.

Currently, most of the trackers based on deep learning network model use a large-scale benchmark datasets [6, 7] to train the network offline, and the sample of the first frame is used to fine-tune the parameters of network online. However, training deep network model online is challenging due to the limited training samples which cannot capture the diversity of the object appearance variations, and offline pre-training is very time consuming. In addition, the quality of the image is also an important factor for the training process. The existing tracking methods utilize the sampling scheme around the object state to obtain the training samples, such as KCF [8], DLT [9], and MCPF [10]. However, the positive samples extracted from each frame are highly overlapped, and they fail to capture rich appearance variations. In this work, we use the first ten frames of a video sequence to fine-tune the parameters of deep network model. But manual annotating the object positions from the first ten frames is always impractical and time consuming. To solve the abovementioned problems, we exploit the advantages of the pre-trained network on a large-scale benchmark datasets to predict the object position. The parameters of deep network have been pre-trained on large-scale datasets. Then we use the prediction network to track the object in the video sequences and automatically obtain the object positions of the first ten frames. The generative adversarial network has great advantages in augmenting training samples. Furthermore, the positive and negative samples are extracted from the first ten frames. Therefore, the positive and negative samples are obtained to fine-tune the generative adversarial network online. The proposed tracking algorithm can capture the changes of the object appearance in the video sequences. During the tracking, the generative model is used to occlude the image features through a randomly generated mask to enhance the diversity of positive samples. The discriminative model employs its discriminative performance to identify the object. These features are robust enough to address the challenges of object tracking. Adversarial learning network can identify masks that retain the most robust features of the object appearance. In terms of tracking accuracy, our approach obtains a relative gain of 5.9% compared to other deep learning-based tracking approaches. In this paper, the proposed tracking method can be applied to

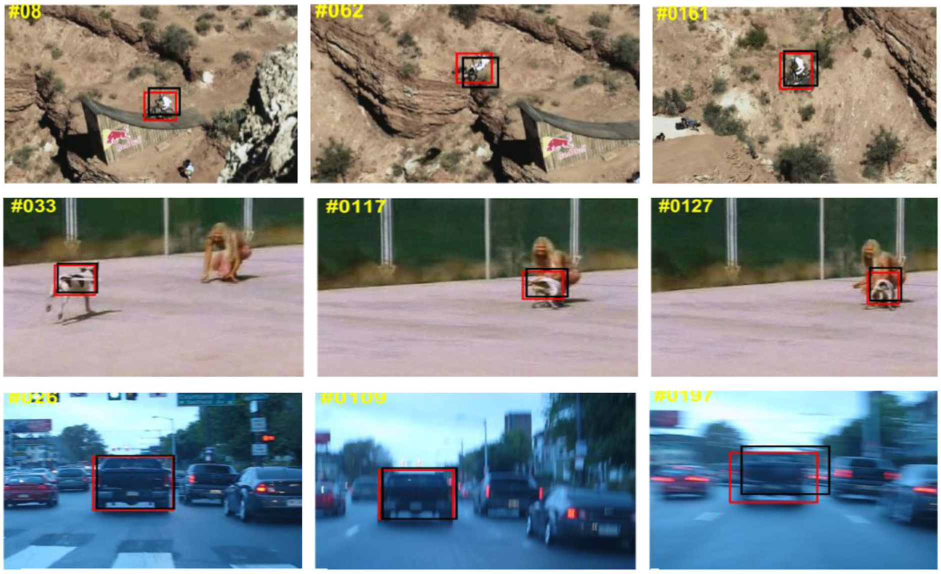


Fig. 1 Object tracking technique for intelligent surveillance analysis, tracking results of the proposed tracker (red), and the VITAL tracker [5] (black) on the OTB2015 benchmark

solve the multimedia forensics and security problems. In other words, it is possible to explore object tracking techniques for various real-time multimedia security applications, such as real-time information hiding and digital forensics. We summarize the main contributions of this work as follows:

(1) We propose an end-to-end the prediction network model for object tracking to improve the tracking accuracy and the computational complexity of training process, which can jointly train the prediction network and spatial attention model in generative adversarial network framework.
(2) An effective spatial attention mechanism is developed, which can adaptively generate the response maps. The feature representations are employed to online tracking process to alleviate over-fitting. In addition, the positive and negative samples are augmented in the feature space to capture a variety of appearance changes over a temporal span by using generative adversarial network.
(3) We conduct the extensive experiments on two popular benchmarks, which demonstrate the proposed object tracking method with learning attention significantly outperforms state-of-the-art methods.

The rest of the paper is organized as follows. In Section 2, we review related work of existing object tracking algorithms. Section 3 introduces the motivation. In Section 4, we introduce our object tracking approach for intelligent surveillance analysis. The experimental settings are presented in Section 5. In Section 6, we present experimental results and discussion in two tracking benchmarks. Finally, Section 7 concludes this paper.

2 Related work

Object tracking is one of the fundamental tasks in computer vision and has been extensively studied over the last decade. There are extensive surveys of object tracking in the literature [11–13]. Before the emergence of the object tracking algorithms based on deep learning, most tracking algorithms used particle filter framework for object tracking, such as Kalman filter [14] and particle filter [15]. However, the disadvantage of these methods is that the number of particles limits the tracking speed. In this section, we review the related advances in three research streams for visual object tracking.

2.1 Correlation filter-based tracking

In recent years, correlation filters (CF) have been widely used in numerous applications such as object detection and recognition. It transfers operations into the Fourier domain as element-wise multiplication. Correlation filters [8, 16–19] have attracted considerable attention due to its computational efficiency and competitive performance. Correlation filter trackers regress all the circular shifted versions of the input features to a Gaussian function. We arrange correlation filter tracking algorithms in a hierarchy and classify them into three categories: Basic correlation filter trackers, regularized correlation filter trackers, and combination of deep learning and correlation filter trackers.

Some basic correlation filter (CF) trackers have been developed to boost performance in tracking by using scale estimation. Bolme et al. [16] propose a minimum output sum of squared error (MOSSE) tracker for object tracking on grayscale images, which encodes object appearance through an adaptive correlation filter by optimizing the output sum of squared error. MOSSE can achieve several hundreds of frames per second. In 2012, Henriques et al. [20] propose the CSK algorithm based on the improvement of MOSSE, which solves the problem of a small number of training samples in the object tracking process through the cyclic matrix and further improves the tracking accuracy of the algorithm by using the kernel technique. However, the above two algorithms adopt simple grayscale features, which are easily disturbed by the external environment, resulting in inaccurate tracking results. They are further improved by the kernelized correlation filters (KCF) [8] with HOG features in a Fourier domain. KCF performs well in OTB50 [4] benchmark in terms of tracking speed and accuracy. Scale change is also a common problem in object tracking. In [21], the DSST tracker learns adaptive multiscale correlation filters using HOG features to handle the scale and translation changes of the objects.

Regularized correlation filter trackers can improve the detection range by using different filter size and patch size. The SRDCF method [22] reduces the boundary effect problem by weighting the weight space of CF. However, its optimization is complicated and the tracking speed is slow. To improve its weakness, the CSR-DCF method [23] adds feature channel and space stability constraints based on SRDCF and uses the augmented Lagrangian scheme to facilitate fast FFT solution, which greatly improves the tracking accuracy and speed. The C-COT method [24] proposes a strategy for training continuous convolution filters, which facilitates the integration of multi-scale CNN features and achieves sub-pixel level tracking accuracy. However, the tracking model framework still adopts the SRDCF method and the computational complexity is high.

ECO [25] method further proposes a factorization convolution scheme to reduce the computational complexity of the tracking model. The UPDT [26] proposes a novel adaptive fusion approach that leverages the complementary properties of deep and shallow features to improve both robustness and accuracy. In [27], the STRCF introduces temporal regularization to SRDCF with single sample. The formulation of this method can not only serve as a reasonable approximation to SRDCF with multiple training samples, but also provide a more robust appearance model than SRDCF in the case of large appearance variations. Although these approaches achieve a satisfied performance in some constrained scenarios, they have an inherent limitation that they resort to low level hand-crafted features, which are vulnerable in dynamic situations including illumination changes, occlusion, deformations, and background clutter.

Inspired by the success of deep learning model in object detection, recognition, and classification tasks, researchers have started to focus on combining of deep learning and correlation filter. In HCF [28] and HDT [29], deep feature is used to extract the object features instead of handcrafted features. It is worth noting that CFNet [30] and DCFNet [31] achieve an end-to-end representation learning.

2.2 Deep learning-based tracking

Features' representations are important for object tracking. Experiments prove that the features designed manually (e.g., haar-like features, histogram, HOG features) are not necessarily suitable for all the objects. Deep learning in the tracking uses the adaptive selection scheme of object features instead of hand-crafted features. The popular trend is to design the deep network structures and pre-train them in order to learn object-specific features.

The main problem of deep learning in the tracking is the lack of training data [32, 33]. The object tracking only provides the object information of the first frame as training data. In this case, it is difficult to train the deep network model with small data. In [9], Wang proposes the idea of off-line pre-training deep network model and online fine-tuning tracking model called DLT tracker, which greatly solves the problem of insufficient training samples in the tracking. SO-DLT [34] continues the strategy of DLT and also greatly improves the problems of DLT by using CNN as a network model for extraction deep features and classifications. In FCNT [35], authors analyze the performance of CNN features pre-trained on ImageNet [36] and design the subsequent network structure based on the analysis results. In [37], the authors utilize a two-layer convolution neural network to learn hierarchical features from auxiliary video sequences, which takes the complex object motion and object appearance changes into account.

In recent years, the Siamese Networks [38, 39] have received more and more attention due to its two stream identical structure. In the offline training phase, a matching score function is trained through the structure of the Siamese Network. Then the matching score function is used to determine the similarity between the current object candidate state and the object template of the first frame during the tracking, which improves the tracking efficiency. In recent years, the generative adversarial network (GAN) has been widely used in many fields, such as object detection [40] and intelligent recommendation system [41]. GAN is first proposed by Ian Goodfellow in 2014, which was originally used to generate realistic-looking images [42]. The main idea behind GAN is to have two competitive neural network models. One takes noise as input and generates samples called a generator. Another model is called discriminator which receives samples from the generator and tries to discriminate true object between two sources. The generator and the discriminator are trained simultaneously by competing with each other. Realizing that there is a huge difference between image classification and tracking, TADA [43] identifies the importance of each convolutional filter and selects the object-aware features based on activations for the object representation. Then, the object-aware features are integrated into a Siamese matching network for visual tracking. Different from the existing approaches using extensive annotated data for supervised learning, UDT [44] is trained on large-scale unlabeled videos in an unsupervised manner. UDT tracker achieves the baseline accuracy of fully supervised trackers which require complete and accurate labels during the training.

In this work, we apply adversarial learning to augment training samples in the feature space to capture appearance variations in temporal domain. In addition, we can exploit robust features over the long temporal span instead of the discriminative features in individual frames.

2.3 Attention mechanisms

Attention mechanisms were first introduced in neuroscience area [45]. They have spread to image classification, multi-object tracking, etc. DAVT [46] employs a discriminative spatial attention scheme for visual tracking. CSR-DCF [47] utilizes color histograms to construct a foreground spatial map in the correlation filter framework, which learns the attention via an end-to-end deep network model. ACFN [48] chooses a subset from the associated correlation filters as an attention mechanism for visual tracking.

3 Motivation

The generative adversarial network (GAN) [42] has been widely used in object detection and semantic segmentation. The generative adversarial networks mainly consist of the generative model and discriminative model. The generative model takes a noise as an input, and the discriminative model takes samples from generative model or training data and outputs the classification probability. This learning process can be written as:

$$L = \min_{G} \max_{D} E_{x\sim p_{data(x)}}[\log D(x)] + E_{z\sim p_{noise(z)}}[\log(1 - D(G(z)))] \tag{1}$$

where G denotes the generative model, D represents the discriminative model, E is the mean operation, and $\boldsymbol{x}$ and $\boldsymbol{z}$ are two vectors from two distributions $P_{\text{data}}(\boldsymbol{x})$ and $P_{\text{noise}}(\boldsymbol{z})$, respectively.

However, the original GAN is not feasible. In our work, G will predict a weight mask which operates on the extracted features. The mask is randomly set at the beginning and gradually identifies the discriminative features by using adversarial learning. The mask is generated by G network as $G(I)$. We denote the predicted mask as $\hat{M}$ and the value of the element (i, j) as $\hat{M}_{ij}$. We define the input image as I, and the value of element (i, j, k) on image I as I_{ijk}. The dropout operation is written as follows:

$$I^0_{ijk} = I_{ijk}\hat{M}_{ij} \tag{2}$$

where I^0_{ijk} denotes the image I after the dropout operation and passed onto the classifier.

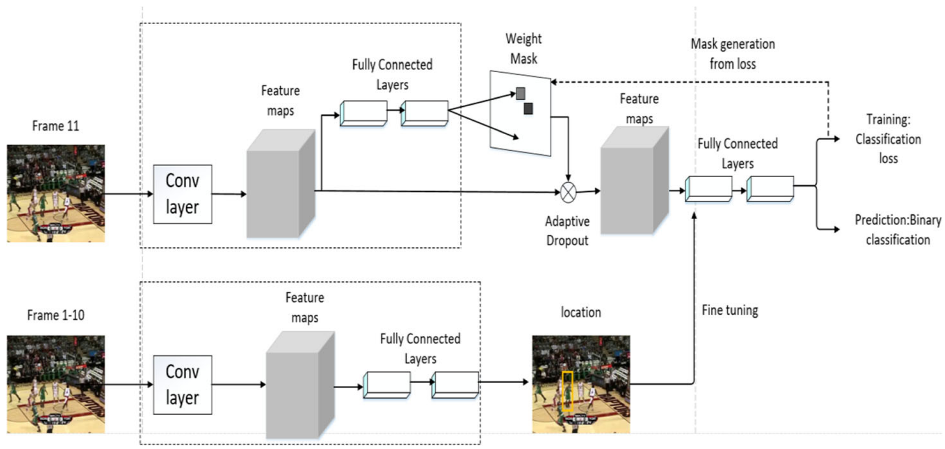


Fig. 2 The architecture of the proposed object tracking algorithm

4 Proposed object tracking method

4.1 Problem formulation

The existing trackers based on deep learning are performed as off-line training and online fine-tuning for surveillance analysis, and only use the object information of the first frame to fine-tune online the learned deep network parameters. However, it is difficult to capture previously unseen object features from one or few examples. In addition, the positive samples in each frame are highly spatially overlapped and they fail to capture rich appearance variations. On the other hand, amount of positive and negative samples for training deep learning network model are nearly impossible to meet up in real world.

In this section, a novel attention generative adversarial network is given at first to describe the overall training architecture. The proposed generative model takes VGG network as input, which is mainly used to capture the object appearance variations of continuous video frames. The discriminative model is introduced as a supervisor and provides guidance on the advantages of the generated object appearance details. To stabilize the training of the generative adversarial networks, we present the mean squared loss to punish the classification error for each pixel. In order to improve the tracking performance, a novel spatial attention mechanism is developed to adapt the offline learned deep model to online object tracking. The VGG network is used to sense the tracked object and decode the object features into the attention response maps. At last, online tracking is described consisting of model updating and scales. The object attention maps are captured by inputting the object appearance information provided in the first ten frames and remaining video frames into the generative model. The score with maximum response score is regarded as the tracking result. This process will be continued for video frames until the end of the video sequences. Figure 2 shows the flowchart of the proposed tracker.

In Fig. 2, the generative model of GAN follows the encoder-decoder framework which attempts to encode the input of the object appearance into feature representation and decode it into corresponding outputs. The discriminative model is a standard convolutional neural network.

4.2 Network architecture

The network includes two branches, and in the lower part of the architecture which utilizes the first ten frames of a video sequences as input, which is called the prediction network. We use the prediction network to track the one to ten frames of a video sequence to obtain the object position of each frame. These features extracted from the predicted object location will be used to fine-tune the fully connected layers of the network located in the top half of the architecture. The object feature of each frame is taken as input of the network from the frame 11 to the end of video. The weight masks are applied to adaptively dropout input features. Adversarial learning identifies the weight mask that maintains the most robust features over a long temporal span while removing the discriminative features from individual frames.

It is worthy to note that the deep learning model is initialized with the weights of a VGG-16 model pre-trained on the ImageNet benchmark for object classification. Most

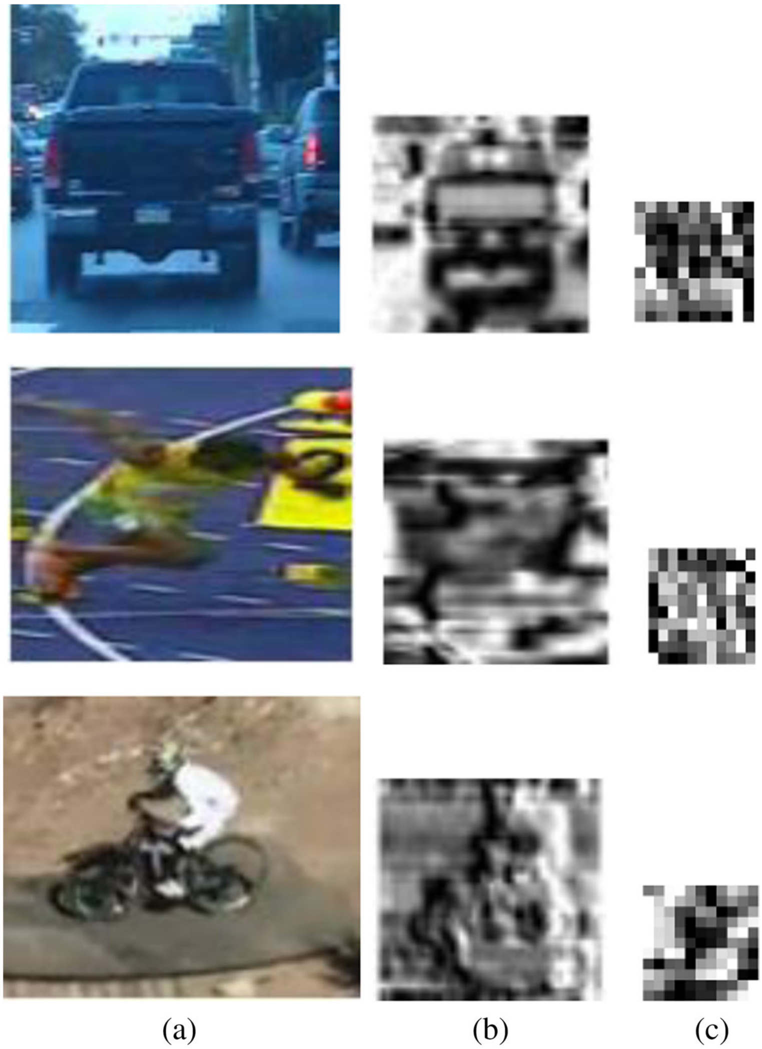


Fig. 3 Foreground response maps predict using different VGG feature maps. (**a**) Input image. (**b**) Using Conv4-1 feature only. (**c**) Using the concatenation of Conv4-2 and Conv4-3 feature with the proposed adversarial learning with attention

of deep learning-based trackers use this offline learned network and then utilize the first frame to fine-tune the network parameters during the tracking. However, it is difficult to obtain the object specific feature of a video by training the deep network model only with the sample of the first frame. On the other hand, if the deep learning network is fine-tuned by using the first n frames of a video, manually labeling the object position will be expensive and impractical. Therefore, a prediction network is introduced into deep learning framework, which can automatically predict the position of the object in the video sequence. The network structure is shown in Fig. 2, which has three convolution layers and two fully connected layers. The architecture of the prediction network is depicted in the lower part of Fig. 2. We directly use a VGG-M [49] model pretrained in the classification task from ImageNet [36], and the parameters of the convolution layers are fixed and only the fully connected layers are fine-tuned online. The cross-entropy loss is adopted for fine-tuning network parameters online. The prediction network is optimized by minimizing the cross-entropy loss function with SGD as follows:

$$\arg\min \frac{1}{N}\sum_{i=1}^{N}\left(-\sum_{j=1}^{2} p(j)\,\log(q(j))\right) \tag{3}$$

where p and q denote training samples and corresponding labels, respectively; N is the number of training samples.

The object features are extracted from the convolution layer and fed to the fully connected layer for classification. Figure 3 reports the foreground response maps predicted by using different VGG feature maps. Figure 3 is the foreground response maps predicted by using different VGG feature maps. Foreground response maps are predicted using different VGG feature maps. Conclusion of Fig. 3 is that shallow layer feature (Conv4-1 feature) focuses on object details; deep layer feature (Conv4-2 and Conv4-3) is semantic features.

Finally, the sample with the highest response score in each frame is regarded as the tracking result. This prediction network is interpreted as a generative network in generative adversarial network framework, and the samples drawn from the predicted location will be used to fine-tune the fully connected layers of the generative model.

The discriminative model is employed to make the generative model produce attention response map that is robust to occlusion, deformation, background clutter, etc. In this work, the attention response map and corresponding RGB frame of a video sequence are considered as the input of discriminative model.

4.3 Training

In our work, mean squared error (MSE) is utilized to measure the difference between estimated attention response map and ground truth map. Given an image I, and its dimension is $N = W \times H$. The mean squared loss can be formulated as:

$$L_{MSE} = \frac{1}{N}\sum_{j=1}^{N}\left(S_j - \hat{S}_j\right)^2 \tag{4}$$

where $\hat{S}$ and S denote the attention response maps and its corresponding ground truth, respectively.

However, mean squared loss function focuses on pixel-level features, and learned deep network can produce a coarse attention response maps. Therefore, training the network with the adversarial loss can be further improved the tracking performance. We iteratively train *G* and *D*, and the adversarial loss function is written as:

$$\begin{aligned} L_{AL} = & \min_G \max_D E_{(C,M)\sim P(C,M)}[\log D(M \cdot C)] \\ & + E_{C\sim P(C)}[\log(1 - D(G(C) \cdot C))] + \lambda E_{(C,M)\sim P(C,M)} \|G(C) - M\|^2 \end{aligned} \tag{5}$$

where *C* is the input image feature, *G*(*C*) is the mask generated by the *G* network, and *M* is the actual mask identifying the discriminative feature. The dot is the dropout operation on the feature *C*. As described in Eq. (5), *G* is used to predict a weight mask *G*(*C*) which operates on the extracted features. The mask is randomly initialized at the beginning and each mask represents a specific type of appearance variation. Through the adversarial learning process, *G* will gradually identify the mask that degrades the performance of classifier.

In each iteration of the training process, object features of the input frames are extracted from convolutional layers and fed into *G* network to obtain the predicted mask m^*. Then, obtained deep features are multiplied by the predicted mask m^* and sent into *D* network. We keep the labels unchanged and train *D* through supervised learning method. *D* is trained to discriminate features from individual frames relying on more robust features over a long temporal span. Thus, it avoids the overfitting issue. *G* is used to predict different masks according to different input deep features. It enables *D* to focus on the temporal robust features without discriminative feature interference from single frame. Given an input image, multiple output features based on several random masks are created. Diversified features are performed through the dropout operation, which are sent to *D* for classification, and we choose the one with the highest loss. The corresponding mask of the selected feature is effective in decreasing the impact of the discriminative features. We set this mask as *M* in equation (5) and update *G* accordingly.

Finally, we combine the MSE loss with adversarial loss to obtain more stable and fast convergence for GAN model. The final loss function for the adversarial training can be formulated as:

$$L_{GAN} = L_{AL}(D(C, G(C)), 1) + \lambda L_{MSE} \tag{6}$$

where λ is a trade-off parameter, and we experimentally set it as 1/20 in our implementation.

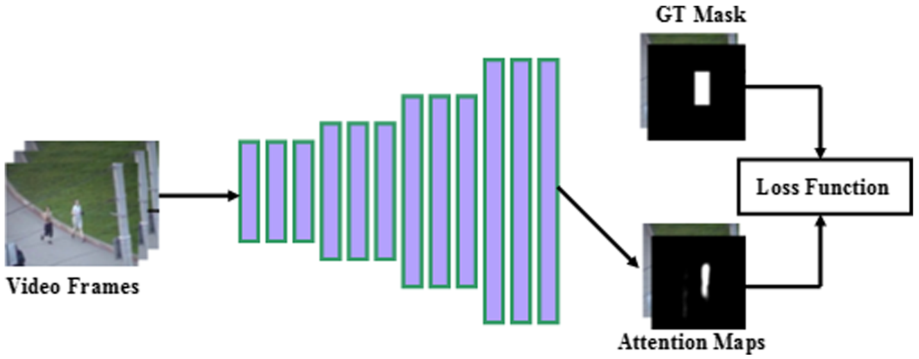


Fig. 4 Spatial attention response map

4.4 Spatial attention

Attention from the training samples can be captured to share a common attention. In practical sceneries, some attention maps are obtained by the initialization of matrix of ones. They are too restrictive to constrain all samples and the object to share a single deep network structure. Therefore, we propose a spatial attention scheme to model attention response map in Fig. 4.

The proposed attention mechanism can capture the general features and distinct the object from the background in the video. It can encode the global information of the object and has a low computational load. The output of attention module is passed through a global pooling layer to produce a channel-wise descriptor. Then three fully connected (FC) layers are added, in which learned for each channel by a self-gating mechanism based on channel dependence. This is followed by reweighting the original feature maps to generate the output of attention module. The cosine similarity is utilized to measure the similarity between current frame features φ_t (p) and the features $\varphi_{t\text{-}1}$ (p) extracted from t-1 frame.

$$w_t(p) = SoftMax\left(\frac{\phi_t(p)\cdot\phi_{t-1}(p)}{|\phi_t(p)|\cdot|\phi_{t-1}(p)|}\right) \tag{7}$$

If the current frame features is close to the features of the last frame, it is prone to the foreground object and assigned with a larger weight, otherwise, a smaller weight is assigned to background pixel.

4.5 Online tracking

In this subsection, we illustrate how our tracker works for visual object tracking. We involve the generative model during the training and remove it in the tracking stage.

We first draw the samples from the first ten frames of a video sequence to fine-tune generative model online. Then, we track the object in all videos. Given an input frame, we generate multiple candidate proposals and extract their deep features. Deep features of the candidate proposals are fed into the classifier to obtain the probability scores. During the online update, we employ these training samples jointly train the generative model and the discriminative model. The object tracking result is obtained by finding the maximum response score in the attention map.

Object appearance model updating plays a critical role in object tracking, and most of trackers update their appearance model in each frame or at a fixed interval. However, this updating strategy may introduce background information into the object appearance model when the tracking result is inaccurate due to occlusion or illumination variations.

In this paper, we need to update the object appearance model with the recently obtained object results. First, we define a fixed length sequence L to store the tracking result of each frame. When the length of L reaches a fixed number of elements, we update object appearance once. In addition, model updating is performed when the number of iteration or maximum value of response map is satisfied. The maximum response score in L is used to update the object appearance model.

Therefore, the new object appearance model is written as:

$$\mathbf{T}_{\mathrm{u}} = (1-\beta)\mathbf{T}_{\mathrm{f}} + \beta\mathbf{T}_{\mathrm{p}} \tag{8}$$

where β is a learning parameter and set empirically; $\mathbf{T}_{\mathrm{u}}$ is the updated object appearance model, which is represented by a linear combination of the initial object template $\mathbf{T}_{\mathrm{f}}$ and the last updated object appearance model $\mathbf{T}_{\mathrm{p}}$. To alleviate the drift problem during the tracking, the initial template is incorporated into the new observation template.

To handle the scale change, we follow the approach in [21] and use patch pyramid with the scale factors. The proposed object tracking algorithm can be summarized as Algorithm 1.

Algorithm 1：The Proposed Tracking Algorithm

```
Input: Initial object bounding box ($x_0$, $y_0$, $w_0$, $h_0$); video frame sequence $I_i$ ( $i$ = 1,2,3...);
Output: Estimated object state ($x_i$, $y_i$, $w_i$, $h_i$);
Initialization:
    Use the prediction network to get the object position of the first ten frames and fine-tune the
    generative adversarial network with the obtained object position;
Tracking process:
  for $i = 11$ : end
    Sample the object candidate states in frame $i$ according to the previous object position ($x_{i-1}$,
      $y_{i-1}$, $w_{i-1}$, $h_{i-1}$);
    Extract deep features of these candidates using convolution layers;
    Feed extracted deep features into the discriminative model for classification and predict the
      possibility score of each candidate;
    Estimate new object position ($x_i$, $y_i$, $w_i$, $h_i$) by the candidate with the highest score $S$;
      if max($S$) <= 0  or $i$%10 == 0  then
        Extract features from the object location of the successfully tracked frame;
        Use adversarial learning to update the discriminative model by Eq.(5);
      end
  end
```

5 Experiments

In this section, we introduce the implementation details of the proposed tracking algorithm. We then compare our tracker with state-of-the-art trackers on two benchmarks for performance evaluation. Our experiments are performed on a workstation by using MatConvNet toolbox [50] with E5 2.4 GHz CPU and Quadro K2200 GPU.

In this work, the first three convolution layers from the VGG-M model are utilized as feature extraction network. The network is pre-trained on a large-scale benchmark datasets. During the adversarial learning, both G and D are learned by the SGD scheme. The learning rate for training G and D are set to 10^{-3} and 10^{-4}, respectively. During the tracking, we draw 256 candidate samples around the object location of each frame

Table 1 Backbone architecture. Details of each building block are reported in square brackets

Block	Output size	Backbone
Conv1	125 × 125	7 × 7, 64, stride 2
Conv2_x	63 × 63	$\begin{bmatrix} 1\times 1, 64 \\ 3\times 3, 64 \\ 1\times 1, 256 \end{bmatrix} \times 3$
Conv3_x	31 × 31	$\begin{bmatrix} 1\times 1, 128 \\ 3\times 3, 128 \\ 1\times 1, 512 \end{bmatrix} \times 4$
Conv4_x	31 × 31	$\begin{bmatrix} 1\times 1, 256 \\ 3\times 3, 256 \\ 1\times 1, 1024 \end{bmatrix} \times 6$

for classification. The masks are set randomly and the resolution of each mask is the same as that of the input features. We update the generative adversarial network using 10 iterations in every 10 frames or the response score of tracked result is less than a predefined threshold. Backbone architecture is shown in Table 1.

5.1 Benchmarks

We conduct the experiments on two standard benchmarks: OTB-2013 [4] and OTB-2015 [6]. Video sequences are defined with bounding box annotations. These datasets cover various challenging aspects in visual tracking task, such as fast motion, background clutter, deformation, occlusion, illumination variations, and low resolution. The performance of all the trackers can be well tested by using two benchmarks.

5.2 Evaluation metrics

We follow the standard evaluation metrics [6] from two benchmarks. For the OTB-2013 and OTB-2015 benchmarks, we use the one-pass evaluation (OPE) with precision and area-under-the-curve (AUC) success rate criteria. The precision metric measures the rate of frame locations within a certain threshold distance from those of the ground truth. The threshold distance is set to 20 for all the trackers. The success rate criterion measures the overlap ratio between the predicted bounding box and the ground truth bounding box.

6 Results and discussion

6.1 Quantitative evaluation

We perform quantitative evaluation on two benchmark datasets. The experimental results of the proposed tracking algorithm are reported as follows.

6.1.1 OTB-2013 benchmark

We use the OTB-2013 benchmark to confirm that our tracker is on par with the state-of-the-art trackers. The trackers that we compared included the 29 trackers from the OTB-2013 benchmark and other state-of-the-art trackers included KCF [8], MUSTer [19], DSST [21], SRDCF [22], C-COT [24], ECO [25], HCF [28],

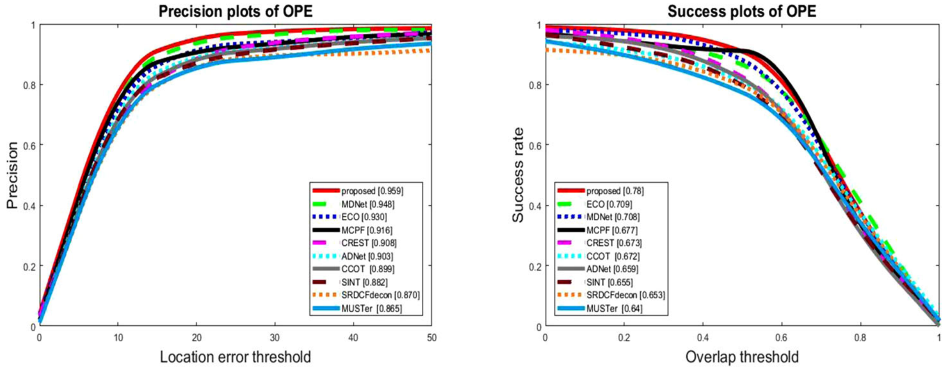


Fig. 5 Precision and success plots on the OTB-2013 dataset using the one-pass evaluation. The number in the legend indicates the average precision scores at 20 pixels and the AUC scores

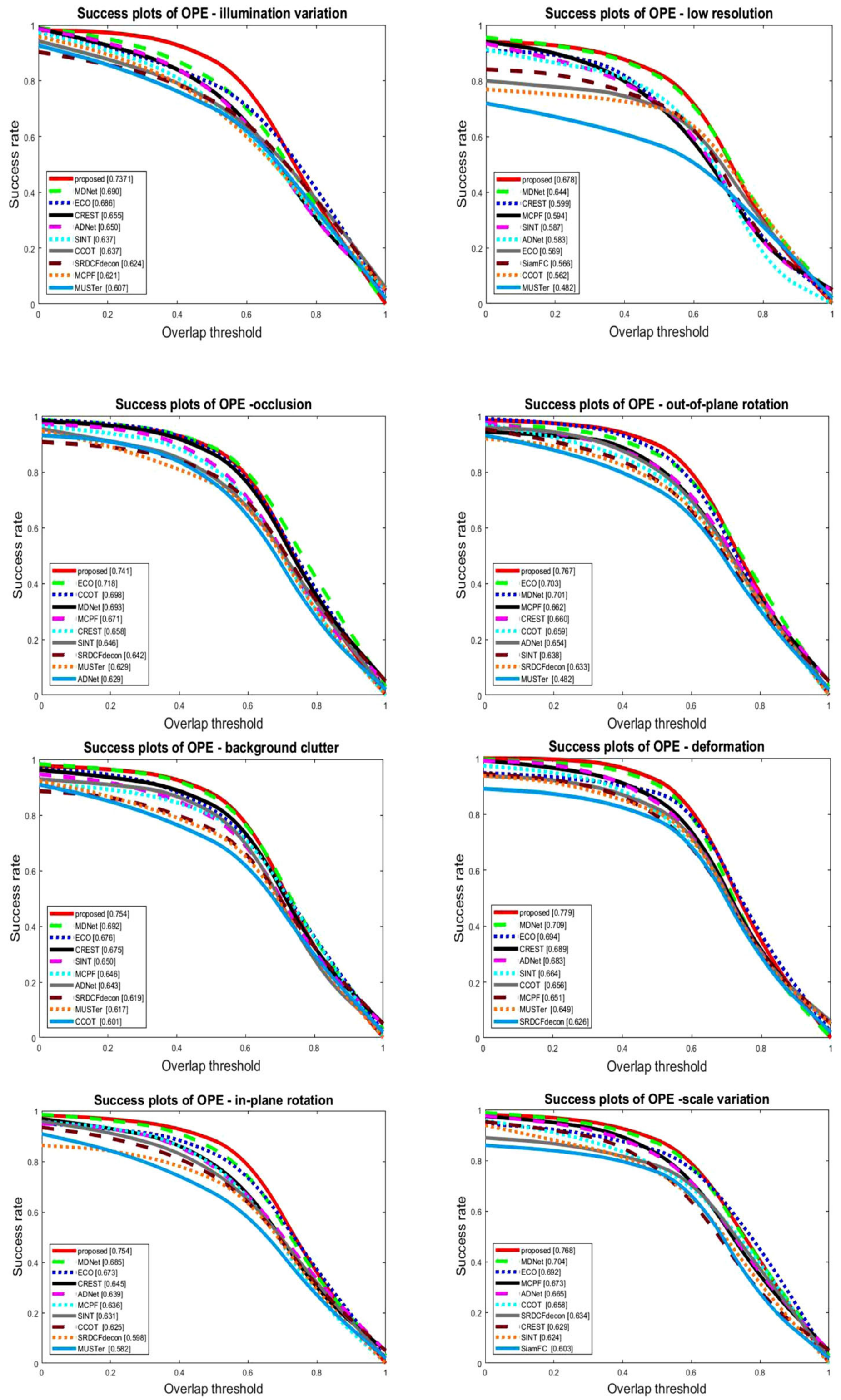


Fig. 6 The success rate plots of eight challenge attributes: illumination variation, deformation, in-plane rotation, out-of-plane rotation, background clutter, occlusion, scale variation, and low resolution. The legend contains the AUC score for each attribute

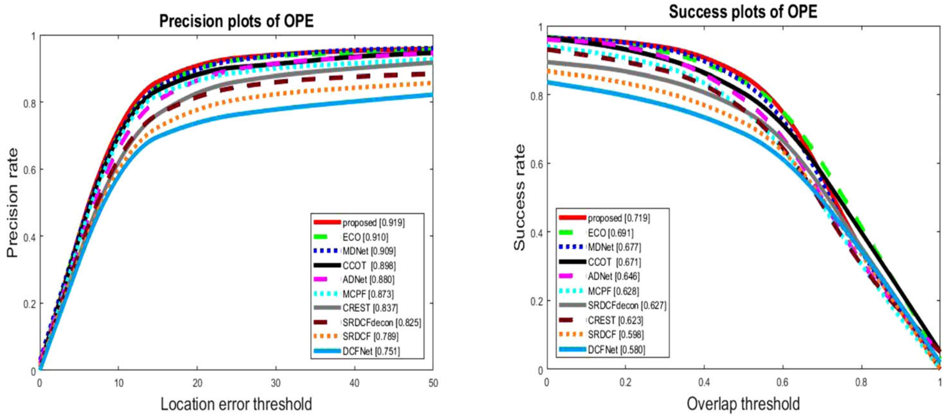
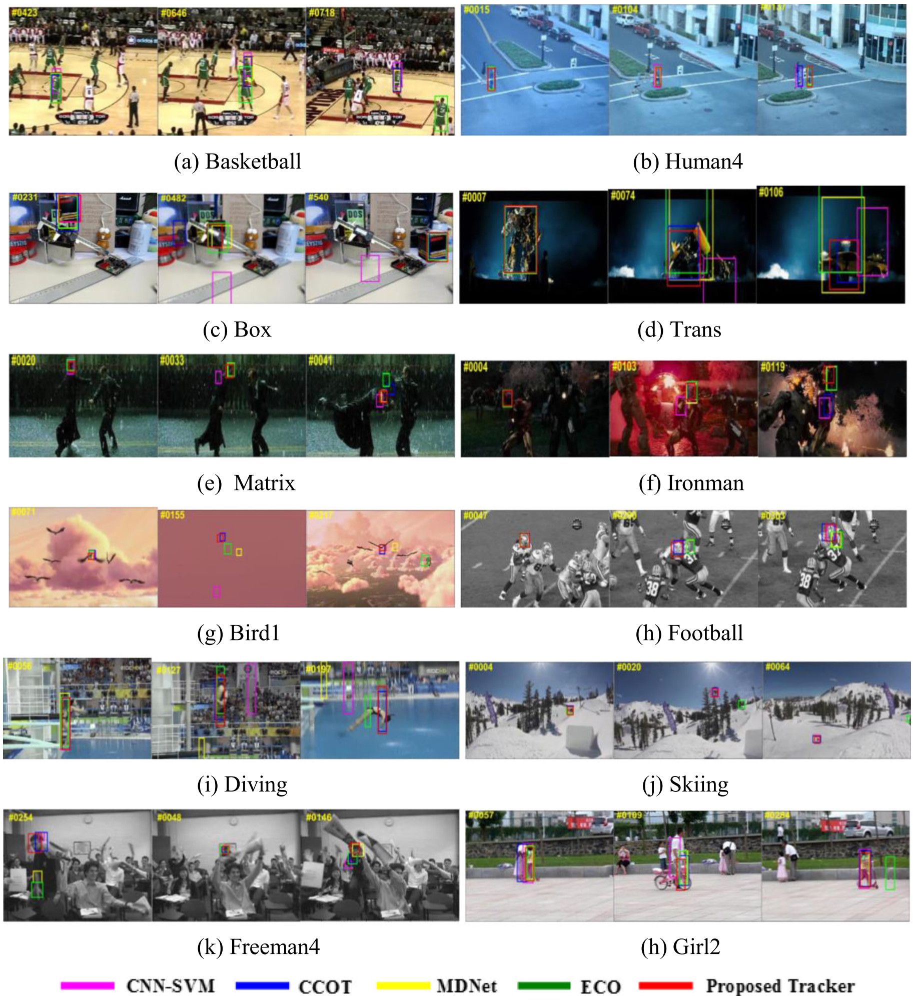


Fig. 7 Precision and success rate plots on the OTB-2015 benchmark by using the one-pass evaluation. The number in the legend indicates the average distance precision scores at 20 pixels and AUC success scores


(a) Basketball

(b) Human4

(c) Box

(d) Trans

(e) Matrix

(f) Ironman

(g) Bird1

(h) Football

(i) Diving

(j) Skiing

(k) Freeman4

(h) Girl2

Fig. 8 Qualitative evaluation of the proposed tracker, CNN-SVM, C-COT, MDNet, and ECO on 12 challenging sequences

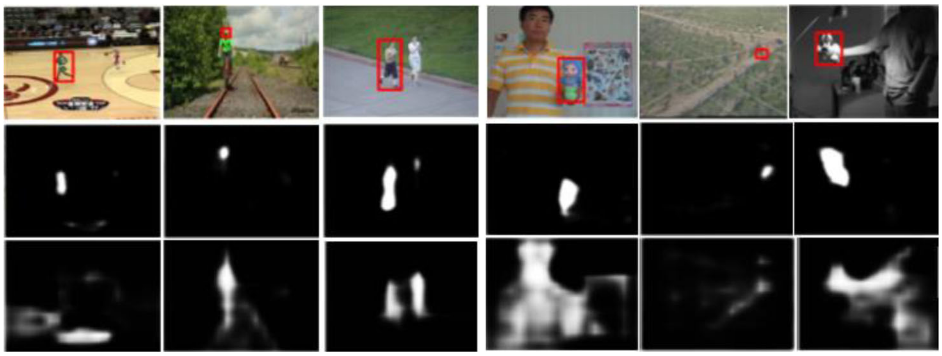


Fig. 9 The maps generated by the proposed spatial attention scheme (middle row) and saliency detection algorithm [63, 64] Deep Saliency (bottom row)

HDT [29], CFNet [30], FCNT [35], SiameFC [38], SINT [39], MDNet [51], LCT [52], VITAL [5], CREST [53], TCDL [54], Staple [55], MCPF [10], DLS-SVM [56], CNN-SVM [57], GOTURN [58], SRDCFdecon [59], DeepSRDCF [60], SCT [61], and ADNet [62].

We evaluate all the trackers on 50 video sequences using the one-pass evaluation with distance precision and overlap success metrics. Figure 5 shows the tracking results from all compared trackers. We only show the top ten trackers for presentation clarity. The number listed in the legend indicates the AUC overlap success rate and precision score at 20 pixels. Overall, it clearly illustrates that our tracking method outperforms the state-of-the-art trackers significantly in both evaluation measures. The OTB-2013 dataset has 11 attributes (e.g., background clutter, occlusion, deformation, scale variation, illumination variation) to describe the different challenges in the tracking. These attributes are useful for analyzing the performance of trackers in different aspects. Figure 6 shows the results of different tracking algorithms on eight main challenging attributes. It demonstrates that our tracker can effectively handle the challenges and achieve

Table 2 Tracking performance and frame per second (FPS) of the state-of-the-art approaches on OTB-100 benchmark. "-"denotes invalid state; the bold fonts indicate the best results

Algorithms	Techinique	Code type	Precision	Success rate	FPS
Ours	Deep Learning	MATLAB & C++	**0.919**	**0.719**	14.8
MUSTer	Correlation Filter	MATLAB & C++	0.774	0.575	6.1
DSST	Correlation Filter	MATLAB & C++	0.693	0.52	35.5
SiamFC	Deep Learning	MATLAB & C++	0.771	0.691	31.2
CCOT	Correlation Filter	MATLAB & C++	0.691	0.682	2.6
MDNet	Deep Learning	MATLAB	0.788	0.678	1.4
SIT	Deep Learning	MATLAB	0.732	0.575	**225.5**
PCOM		MATLAB & C++	-	-	27.6
KCF	Correlation Filter	MATLAB	0.690	0.477	124.1
CN		MATLAB & C++			65.4
Struck	SVM	C++	-	-	9.6
TLD	Boosting	MATLAB & C++	-	-	2.7
MIL	Boosting	C++	-	-	31.8

Table 3 Results of different features

ConvLayer	2	3	4	3, 4	2, 3, 4
Precision	0.624	0.705	0.764	**0.821**	0.795
Success rate	0.502	0.524	0.557	**0.587**	0.566

leading performance. The proposed method performs favorably against the state-of-the-art trackers when evaluating with eight challenging factors.

6.1.2 OTB-2015 benchmark

For more detailed analysis, we also compare our tracker with the state-of-the-art trackers on the OTB-2015 benchmark. Figure 7 shows that the proposed tracker performs well. Although the ECO tracker has achieved a good performance, the proposed tracker uses the samples of the first ten frames to train deep network model, so both precision and success rate are leading.

6.2 Qualitative evaluation

In Fig. 8, we qualitatively report the results of other four state-of-the-art trackers (such as, CNN-SVM, C-COT, MDNet, ECO) and the proposed tracker on 12 challenging video sequences.

In most of the video sequences, CNN-SVM is unable to locate the object position due to the limited performance of SVM classifier. MDNet improves CNN-SVM through an end-to-end CNN network formulation, and it performs well on deformation (Trans), low resolution (Skiing), and fast motion (Diving). However, it does not perform well in handling out-of-plane rotation (Ironman) and occlusion (Human4). The correlation filter-based trackers such as C-COT and ECO use deep features for visual object tracking, but they fail to exploit more sophisticated deeper architectures. They perform well in handling occlusion (Human4, Box) and deformation (Trans). However, the

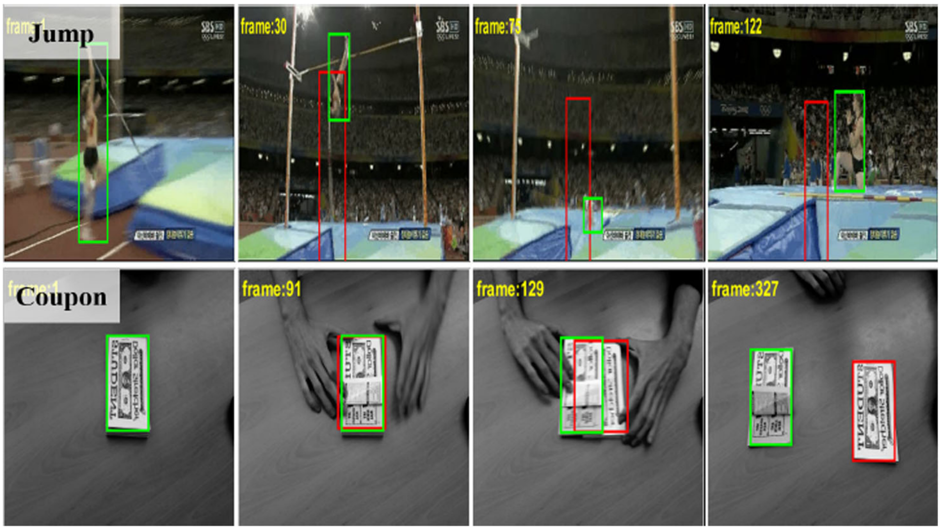


Fig. 10 Failure cases of our method on "Jump" and "Coupon". Green and red bounding means the ground truth and the results from our tracker, respectively

tracked object drifts when it undergoes heavy occlusions (Bird1). Overall, our tracker captures the appearance variations of the object by fine-tuning the network and the adversarial learning scheme enhances the discriminative ability of the classifier. Therefore, the proposed tracker performs well in estimation both the scale and position of the object on these challenging video sequences. The proposed tracker performs favorably against state-of-the-art.

Moreover, the accurate prediction of spatial attention response map is a key factor in our tracker. Thanks to the utilization of mean squared loss and adversarial loss, the predicted attention response maps are robust for most challenging cases. Even if the attention is not precise, the tracking results will not be largely affected in our experiments. Figure 9 shows the robustness of the proposed tracking method. The red bounding box is our results.

In addition, a major concern of the proposed tracker is its computational efficiency. Our tracker largely reduces the computational burden in learning and tracking. The parameters of deep network model can also be pre-computed in the training phase. Its tracking error grows proportionally as the number of index increases. During the tracking, the parameters of deep network model will be updated in a fixed interval time. This greatly accelerates the tracking process. The runtime of our tracker against other trackers is shown in Table 2.

6.3 Feature comparison

We compare the feature effects of different layers of deep learning network model on the OTB-2015 benchmark, which is shown in Table 3. We can see that the combination of the features extracted from conv3 and conv4 layers achieves the best results, which verifies the rationality of the feature selection strategy of the proposed tracking algorithm. The best results are in bold.

6.4 Failure cases

Although the proposed tracking algorithm can achieve a satisfied performance, a few failure cases occur when object suffers from the long-term occlusions. When the tracked object reappears and becomes very small, the proposed tracking method fails to follow the object due to the limited pixels and appearance variations, which can result in poor tracking performance. A feature selection implementation strategy using the feature from conv2 is able to track the object, because the features of conv2 layer have higher resolution than the features from deeper layers. For the Biker sequence, the object suddenly moves violently beyond the search area of the proposed tracking method. Many single object trackers are not able to cope with this challenge problem in this sequence.

Our tracker fails to track objects when they have very similar appearance (e.g., result in “Coupon” sequence) and experience dramatic topology changes (e.g., result in “Jump” sequence) in Fig. 10. Another limitation of our tracker is that the running speeds (14.8 fps on OTB-100 dataset) are far below real-time usage, which cannot be easily employed in other products, such as mobile phone and many embedded devices. We leave these issues for further studies.

7 Conclusion

In this paper, we propose an effective object tracking method with learning attention. We design a prediction network which is pre-trained off-line and used to predict the object positions of a video sequence. The object positions of the first ten frames are employed to fine-tune prediction network for obtaining rich appearance variations. The positive and negative samples are also augmented. Furthermore, these object locations are captured to mine the domain-specific information through fine-tuning the adversarial generative network. We adaptively use dropout to mine the discriminative features which are originally diminished during the training process. The adaptive dropout is achieved via adversarial learning to find discriminative features according to different inputs. In addition, we present a spatial attention mechanism to improve the tracking performance. Compared with the state-of-the-art, the proposed tracking method achieves outstanding performance in two large public tracking benchmarks. Further research directions include applying the spatial attention into multi-modal applications.

Abbreviations

DL: Deep learning; CF: Correlation filter; CNN: Convolutional neural network; AUC: Area under the curve; MOSSE: Minimum output sum of squared error; HOG: Histogram of oriented gradient; OPE: One pass evaluation

Acknowledgements

The authors would like to thank the anonymous reviewers for useful and constructive comments that help improve the quality of this paper.

Authors' contributions

All authors took part in the discussion of the work described in this paper. All authors read and approved the final manuscript.

Funding

This work is supported in part by the National Natural Science Foundation of China (Grant No. 61802058, 61911530397); in part by the Equipment Advance Research Foundation Project of China (Grant No. 61403120106); in part by the Startup Foundation for Introducing Talent of Nanjing University of Information Science and Technology (Grant No. 2018r057); in part by the Project funded by the China Postdoctoral Science Foundation (Grant No. 2019M651650); and in part by the Open Project Program of the State Key Lab of CAD&CG (Grant No. A1919), Zhejiang University, and the PAPD fund.

Availability of data and materials

We used publicly available dataset in order to illustrate and test our methods. The OTB dataset can be found in http://cvlab.hanyang.ac.kr/ trackerbenchmark/datasets.html.

Competing interests

The authors declare that they have no competing interests.

Received: 25 February 2020 Accepted: 4 October 2020
Published online: 11 November 2020

References

1. Y. Zhang, X. Luo, Y. Guo, et al., *Multiple Robustness Enhancements for Image Adaptive Steganography in Lossy Channels*, IEEE Transactions on Circuits and Systems for Video Technology, Published online (Early Access) (2019). https://doi.org/10.1109/TCSVT.2019.2923980
2. C. Qin, W. Zhang, F. Cao, X. Zhang, C. Chang, Separable reversible data hiding in encrypted images via adaptive embedding strategy with block selection. Signal Processing **153**, 109–122 (2018)
3. J. Wang, H. Wang, J. Li, X. Luo, Y. Shi, S. Jha, *Detecting double JPEG compressed color images with the same quantization matrix in spherical coordinates*, IEEE Transactions on Circuits and Systems for Video Technology. https://doi.org/10.1109/TCSVT.2019.2922309
4. X. Wu, C. Luo, Q. Zhang, J. Zhou, H. Yang, Y. Li, Text detection and recognition for natural scene images using deep convolutional neural networks. Comput Mater Continua **61**(1), 289–300 (2019)
5. Y. Song, C. Ma, X. Wu, et al., *VITAL: Visual tracking via adversarial learning*, IEEE Conference on Computer Vision and Pattern Recognition (2018), pp. 8990–8999
6. Y. Wu, J. Lim, M.-H. Yang, *Online object tracking: a benchmark, in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA* (2013), pp. 2411–2418
7. Y. Wu, J. Lim, M. Yang, Object tracking benchmark. IEEE Trans. Pattern Anal. Mach. Intell **37**(9), 1834–1848 (2015)
8. J.F. Henriques, R. Caseiro, P. Martins, et al., High-speed tracking with kernelized correlation filters. IEEE Trans. Pattern Anal. Machine Intell. **37**(3), 583–596 (2015)
9. N. Wang, D.-Y. Yeung, *Learning a deep compact image representation for visual tracking*, The Annual Conference on Neural Information Processing Systems (2013), pp. 809–817

10. T. Zhang, C. Xu, M.H. Yang, *Multi-task correlation particle filter for robust object tracking*, IEEE Conference on Computer Vision and Pattern Recognition (2017), pp. 4335–4343
11. A. Yilmaz, O. Javed, M. Shah, Object tracking: a survey. ACM Computing Surveys (CSUR) **38**(4), 13 (2006)
12. J. Wang, T. Li, X. Luo, Y. Shi, S. Jha, Identifying computer generated images based on quaternion central moments in color quaternion wavelet domain. IEEE Trans. Circ. Syst. Video Technol **29**(9), 2775–2785 (2018)
13. A.W.M. Smeulders, D.M. Chu, R. Cucchiara, et al., Visual tracking: an experimental survey. IEEE Trans. Pattern Anal. Machine Intell **36**(7), 1442–1468 (2014)
14. F. Ababsa, *Robust extended kalman filtering for camera pose tracking using 2D to 3D lines correspondences*, IEEE/ASME International Conference on Advanced Intelligent Mechatronics (2009), pp. 1834–1838
15. M.S. Arulampalam, S. Maskell, N. Gordon, et al., A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Trans. Signal Processing. **50**(2), 174–188 (2002)
16. D.S. Bolme, J.R. Beveridge, B.A. Draper, et al., *Visual object tracking using adaptive correlation filters*, IEEE Conference on Computer Vision and Pattern Recognition (2010), pp. 2544–2550
17. X. Cheng, Y. Zhang, L. Zhou, Y. Zheng, Visual tracking via auto-encoder pair correlation filter. IEEE Trans. Industrial Electronics. **67**(4), 3288–3297 (2020)
18. R. Cheng, YATA: Yet another proposal for traffic analysis and anomaly detection, Computers. Mater. Continua. **60**(3), 1171–1187 (2019)
19. Z. Hong, Z. Chen, C. Wang, et al., *Multi-store tracker (MUSTer): a cognitive psychology inspired approach to object tracking*, IEEE Conference on Computer Vision and Pattern Recognition (2015), pp. 749–758
20. J.F. Henriques, R. Caseiro, P. Martins, et al., *Exploiting the circulant structure of tracking-by-detection with kernels*, European Conference on Computer Vision (2012), pp. 702–715
21. M. Danelljan, G. Häger, F. Khan, et al., *Accurate scale estimation for robust visual tracking* (British Machine Vision Conference, Nottingham, 2014), pp. 1–5
22. M. Danelljan, G. Hager, F. Shahbaz Khan, M. Felsberg, *Learning spatially regularized correlation filters for visual tracking*, IEEE International Conference on Computer Vision (2015), pp. 4310–4318
23. A. Lukezic, T. Vojir, L. C. Zajc, et al. Discriminative Correlation Filter with Channel and Spatial Reliability. IEEE Conference on Computer Vision and Pattern Recognition. (2017).
24. M. Danelljan, A. Robinson, F. Khan, M. Felsberg, *Beyond correlation filters: Learning continuous convolution operators for visual tracking*, European Conference on Computer Vision (2016), pp. 472–488
25. M. Danelljan, G. Bhat, F.S. Khan, M. Felsberg, *ECO: Efficient Convolution Operators for Tracking*, IEEE Conference on Computer Vision and Pattern Recognition (2017), pp. 6931–6939
26. G. Bhat, J. Johnander, M. Danelljan, et al., *Unveiling the power of deep tracking*, European Conference on Computer Vision (2018), pp. 483–498
27. F. Li, C. Tian, W. Zuo, et al., *Learning spatial-temporal regularized correlation filters for visual tracking*, IEEE Conference on Computer Vision and Pattern Recognition (2018), pp. 4904–4913
28. C. Ma, J.B. Huang, X. Yang, et al., *Hierarchical convolutional features for visual tracking*, IEEE international conference on computer vision (2015), pp. 3074–3082
29. Y. Qi, S. Zhang, L. Qin, et al., *Hedged deep tracking*, IEEE conference on computer vision and pattern recognition (2016), pp. 4303–4311
30. J. Valmadre, L. Bertinetto, J. Henriques, et al., *End-to-end representation learning for correlation filter based tracking*, IEEE Conference on Computer Vision and Pattern Recognition (2017), pp. 2805–2813
31. Q. Wang, J. Gao, J. Xing, M. Zhang, W. Hu, *Dcfnet: Discriminant correlation filters network for visual tracking. arXiv preprint arXiv:1704.04057* (2017)
32. G. Wang, C. Luo, X. Sun, Z. Xiong, and W. Zeng, Tracking by instance detection: A meta-learning approach, IEEE Conference on Computer Vision and Pattern Recognition. (2020).
33. M. Zhang, W. Ren, Y. Piao, Z. Rong, H. Lu, *Select, Supplement and Focus for RGB-D Saliency Detection, CVPR* (2020).
34. N. Wang, S. Li, A. Gupta, et al., Transferring rich feature hierarchies for robust visual tracking. arXiv preprint arXiv:1501.04587, (2015).
35. L. Wang, W. Ouyang, X. Wang, et al., *Visual tracking with fully convolutional networks*, IEEE International Conference on Computer Vision (2015), pp. 3119–3127
36. J. Deng, W. Dong, R. Socher, et al., *ImageNet: a large-scale hierarchical image database*, IEEE conference on computer vision and pattern recognition (2009), pp. 248–255
37. K. Zhang, Q. Liu, Y. Wu, et al., Robust visual tracking via convolutional networks without training. IEEE Trans. Image Processing **25**(4), 1779–1792 (2016)
38. L. Bertinetto, J. Valmadre, J.F. Henriques, et al., *Fully-convolutional siamese networks for object tracking*, European conference on computer vision (2016), pp. 850–865
39. R. Tao, E. Gavves, A.W.M. Smeulders, *Siamese instance search for tracking*, IEEE conference on computer vision and pattern recognition (2016), pp. 1420–1429
40. X. Wang, A. Shrivastava, A. Gupta, *A-fast-rcnn: Hard positive generation via adversary for object detection*, IEEE Conference on Computer Vision and Pattern Recognition (2017), pp. 2606–2615
41. W. Jiang, J. Chen, Y. Jiang, Y. Xu, Y. Wang, L. Tan, G. Liang, A new time-aware collaborative filtering intelligent recommendation system. Comput. Mat. Continua. **61**(2), 849–859 (2019)
42. I. Goodfellow, J. Pouget-Abadie, M. Mirza, et al., *Generative adversarial nets. Advances in neural information processing systems* (2014), pp. 2672–2680
43. X. Li, C. Ma, B. Wu, et al. Target-Aware Deep Tracking. IEEE Conference on Computer Vision and Pattern Recognition, (2019).
44. N. Wang, Y. Song, C. Ma, W. Zhou, et al. Unsupervised Deep Tracking. IEEE Conference on Computer Vision and Pattern Recognition, (2019).
45. B.A. Olshausen, C.H. Anderson, D.C. Van Essen, A neurobiological model of visual attention and invariant pattern recognition based on dynamic routing of information. J. Neurosci. **13**(11), 4700–4719 (1993)
46. J. Fan, Y. Wu, S. Dai, *Discriminative spatial attention for robust tracking*, In European Conference on Computer Vision (2010), pp. 480–493

47. A. Lukezic, T. Vojir, L. Cehovin Zajc, J. Matas, and M. Kristan. Discriminative correlation filter with channel and spatial reliability. In IEEE Conference on Computer Vision and Pattern Recognition, 2017.
48. J. Choi, H.J. Chang, S. Yun, T. Fischer, Y. Demiris, *Attentional correlation filter network for adaptive visual tracking*, IEEE Conference on Computer Vision and Pattern Recognition (2017), pp. 4807–4816
49. K. Simonyan, A. Zisserman, *Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556* (2014)
50. A. Vedaldi, K. Lenc, *Matconvnet: Convolutional neural networks for MATLAB*, The 23rd ACM international conference on Multimedia. ACM (2015), pp. 689–692
51. H. Nam, B. Han, *Learning multi-domain convolutional neural networks for visual tracking*, IEEE Conference on Computer Vision and Pattern Recognition (2016), pp. 4293–4302
52. C. Ma, X. Yang, C. Zhang, et al., *Long-term correlation tracking*, IEEE conference on computer vision and pattern recognition (2015), pp. 5388–5396
53. Y. Song, C. Ma, L. Gong, et al., *CREST: Convolutional residual learning for visual tracking*, IEEE International Conference on Computer Vision (2017), pp. 2555–2564
54. X. Cheng, Y. Zhang, J. Cui, et al., Object Tracking via Temporal Consistency Dictionary Learning. IEEE Trans. Syst. Man. Cybernet Syst. **47**(4), 628–638 (2017)
55. L. Bertinetto, J. Valmadre, S. Golodetz, et al., *Staple: Complementary learners for real-time tracking*, IEEE conference on computer vision and pattern recognition (2016), pp. 1401–1409
56. J. Ning, J. Yang, S. Jiang, et al., *Object tracking via dual linear structured SVM and explicit feature map*, IEEE conference on computer vision and pattern recognition (2016), pp. 4266–4274
57. S. Hong, T. You, S. Kwak, et al., *Online tracking by learning discriminative saliency map with convolutional neural network*, International conference on machine learning (2015), pp. 597–606
58. D. Held, S. Thrun, S. Savarese, *Learning to track at 100 fps with deep regression networks*, European Conference on Computer Vision (2016), pp. 749–765
59. M. Danelljan, G. Hager, K.F. Shahbaz, et al., *Adaptive decontamination of the training set: A unified formulation for discriminative visual tracking*, IEEE Conference on Computer Vision and Pattern Recognition (2016), pp. 1430–1438
60. M. Danelljan, G. Hager, K.F. Shahbaz, et al., *Convolutional features for correlation filter based visual tracking*, IEEE International Conference on Computer Vision Workshops (2015), pp. 58–66
61. J. Choi, J. Chang, J. Jeong, et al., *Visual tracking using attention-modulated disintegration and integration*, IEEE Conference on Computer Vision and Pattern Recognition (2016), pp. 4321–4330
62. S. Yun, J. Choi, Y. Yoo, et al., *Action-decision networks for visual tracking with deep reinforcement learning*, IEEE Conference on Computer Vision and Pattern Recognition (2017), pp. 2711–2720
63. Y. Piao, Z. Rong, M. Zhang, H. Lu, *Exploit and Replace: An Asymmetrical Two-Stream Architecture for Versatile Light Field Saliency Detection, AAAI* (2020)
64. Y. Piao, W. Ji, J. Li, et al., *Depth-Induced Multi-Scale Recurrent Attention Network for Saliency Detection*, International conference on computer vision (2019), pp. 7254–7263

Publisher's Note

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