# TRIT: A ROBUST TRACKER BASED ON TRIPLET NETWORK

Peng Zou and Yunfei Cai

Department of Intelligent Science and Technology, Nanjing University of Science and Technology, Nanjing, China

***ABSTRACT***

*In this paper, a target tracking algorithm, TriT(Triplet Network Based Tracker), based on Triplet network is proposed to solve the problem of visual target tracking in complex scenes. Compared with Siamese-fc algorithm, which adopts a two-way feature extraction network, TriT uses three parallel convolutional neural networks to extract the features of the target in the first frame, the target in the previous frame and the search regions of the current frame, and then obtains the high-level semantic information of the three areas. Then, the features of the target in the first frame and the target in the previous frame are respectively convolved with the features of the current search region to obtain the similarity between each position in the search area and the target in the first frame and the target in the previous frame, so as to generate two similarity score maps. Then, interpolate and enlarge the two low-resolution score maps, and use the APCE value of the score maps as the medium to fuse the two score maps, according to which the position of the tracking target in the current frame can be located. Experiments in this paper have confirmed that, compared with some other real-time target tracking algorithms such as Siamese-fc, TriT has great advantages in tracking robustness and can effectively execute tracking tasks in complex scenes, such as illumination change, occlusion and interference of similar targets. Experimental results also show that the proposed algorithm has good real-time performance.*

***KEYWORDS***

*Target Tracking, High Robustness, Triplet Network, Score Maps Fusion*

## 1. INTRODUCTION

With the wide application of video behavior analysis, automatic driving, human-computer interaction and other technologies, visual tracking technology has also attracted people's attention. In recent years, scholars have conducted a lot of research on it. In particular, the rise of deep learning technology has led to the development of many branches of visual tracking algorithms. However, due to the illumination change, deformation, rotation, background clutter, similar interference objects and uneven camera motion and other interference factors in the scene of visual target tracking [1,2], visual tracking is still a very challenging task in practice.

At present, the core of many tracking algorithms is to match the target image with the input frame. For an ideal tracking matching algorithm, it should provide a good match even if there are interference factors such as occlusion of the target, scale change, rotation, uneven illumination, and uneven camera movement. One solution is to explicitly model these distortions in matching by introducing affine transformation [3], probability matching [4], feature image [5], illumination invariant [6], occlusion detection [7] and other operations. However, the drawback of this method is that a modeled matching mechanism may well solve one kind of interference,

Natarajan Meghanathan et al. (Eds) : CSEIT, CMLA, NeTCOM, CIoT, SPM, NCS, WiMoNe, Graph-hoc - 2019
pp. 139-152, 2019. © CS & IT-CSCP 2019 DOI: 10.5121/csit.2019.91311

but it is likely to produce another kind of interference. In this article, we study a matching mechanism, rather than explicit modeling matches for specific interferences. We learn invariants from training videos containing various interfering factors. If training dataset is large enough, we can learn a general matching function apriori, which can deal with common interfering factors occurring in the video, such as target appearance change.

Based on the target tracking algorithm of Siamese-fc[8], this paper proposes a target tracking method named TriT based on Triplet network[9]. First, three parallel convolutional neural networks are used to extract the features of the input target in the first frame, the target in the previous frame and the search area in the current frame to obtain the high-level semantic information of the three areas. Then, the target features of the first frame and the previous frame are convolved with the features of the current search area, and the similarity between each position in the search area and the first target and the previous frame is obtained, thus generating two similarity score maps. Finally, interpolation and amplification were carried out for the two score maps with low resolution, and the APCE[10] value of the score maps was used as the medium to fuse the two score maps. Then we can get the syncretic score map according to which we can get the target position more precisely. All network models are obtained by offline pre-training, and the online tracking process does not update the model, so the frame rate can meet the requirements of real-time tracking. Experiments in this paper show that this method is more robust than the original algorithm, and its real-time performance is slightly reduced, but it can still meet the requirements of real-time tracking in most scenes.

In Section 1, we introduce the importance of target tracking technology and some basic target tracking algorithms. Then the TriT algorithm proposed in this paper is introduced. Finally, the article structure is introduced.Then in Section 2, we introduce the development of target tracking algorithm and the related work.In Section 3, we first introduce the target tracking algorithm based on Siamese network. Then the TriT target tracking algorithm, including theory and training steps and methods, is introduced in detail.Section 4 is experiment and analysis. TriT is compared with some other target tracking algorithms, and the experimental results are analyzed.Then, in Section 5, we analyze some deficiencies of TriT and propose some improvement directions.

## 2. RELATED WORK

Research on visual tracking algorithms has been very active in the field of computer vision in the past decades. From the initial particle filter [11] framework based algorithms to the subsequent correlation filter [12] based algorithms, the performance of tracking algorithms has been gradually improved. With the introduction of machine learning algorithms, especially deep learning algorithms, tracking algorithms have shown a trend of diversified development in recent years, and their performance and robustness have been significantly improved. The introduction of deep learning technology and the adoption of similarity measurement standard [13] can improve the accuracy and speed of the algorithm to a new level and achieve real-time and robust target tracking.

In 2016, Martin Danelljan proposed C-COT [14] algorithm. C-COT combines deepSRDCF and uses deep neural network VGGNet[15] as feature extraction network. It interpolates feature images with different resolution into continuous spatial domain by cubic interpolation, and then uses Hessian matrix [16] to obtain target positions with sub-pixel accuracy. It solves the problem of training filter in continuous space domain. The disadvantage of C-COT is the large training data and feature space, which leads to the low low tracking speed. In 2017, Martin Danelljan proposed ECO[17] tracking algorithm. ECO mainly solves the problem of too large model in C-COT. It speeds up the tracking speed by reducing the correlation filtering parameters, simplifying the training set, compressing the feature space and reducing the update frequency of

the model. GOTURN[18] algorithm published by Davia Held et al. in 2015 can be regarded as the pioneer of target tracking using end-to-end deep learning model. GOTURN algorithm uses ALOV300+ video data set and detection data set in ImageNet to train a convolutional neural network based on image pair as input. The network output search area changes relative to the target location in the previous frame, so as to obtain the target location in the current frame. In 2016, Luca Bertinetto proposed a new algorithm called Siamese-fc based on deep learning tracking [8]. It uses fully convolutional Siamese network for target tracking. Its structure contains two identical fully convolutional networks, and the input is a pair of images, contain the target template and the search area. Features were extracted from the two input channels through the network, and the similarity between the template image and each position in the search area was calculated by matching the two groups of features through the template. The point with the highest similarity was the position of the target. In 2018, Anfeng He et al. from Chinese academy of sciences proposed SA-Siam[19] algorithm. It changes the network structure of Siamese-fc, adopts double Siamese network, that is, adds a Siamese network to extract the semantic features of the target object, and models the target together with the features extracted from the previous network branch, so as to improve the discrimination of the model in the target tracking task. Similar to SA-Siam, RASNet[20] proposed by Qiang Wang et al also improved the similarity measurement method based on Siamese network. RASNet uses several attention mechanisms to weight the space and channel of Siamese-fc features, and decomposes the coupling of feature extraction and discriminant analysis to improve the discriminant ability.

## 3. PROPOSED METHOD

### 3.1. Siamese Network

Standard Siamese network [21] is a kind of neural network containing two or more identical subnetwork structures. These subnetworks have the same network structure, parameters and weights. By constructing some distance measures (Euclidean distance, Manhattan distance, cosine distance), Siamese networks have become an important method in measuring learning. Hu et al. [22] applied Siamese network to face recognition and achieved 97.45% accuracy in face data set LFW.

The Siamese network is mainly composed of the following two parts (Figure 1):

1) Feature extraction network: Two branch networks extract features from two input values respectively. The two networks have the same structure and share weights. It is usually implemented by convolutional neural network, which includes convolutional layer, pooling layer and activation layer of some nonlinear functions.
2) Decision network: The role of decision network is to process the output features of the feature extraction network in the next step, so as to obtain a specific form of output. There are many kinds of decision network, which can be selected according to different task forms. For example, in some tasks, the decision network is a cascade of fully connected layers, while in others, the decision network is a series of measurement functions (euceucine distance, cosine distance, etc.) and loss functions (such as cross entropy loss, contrastive loss, etc.).

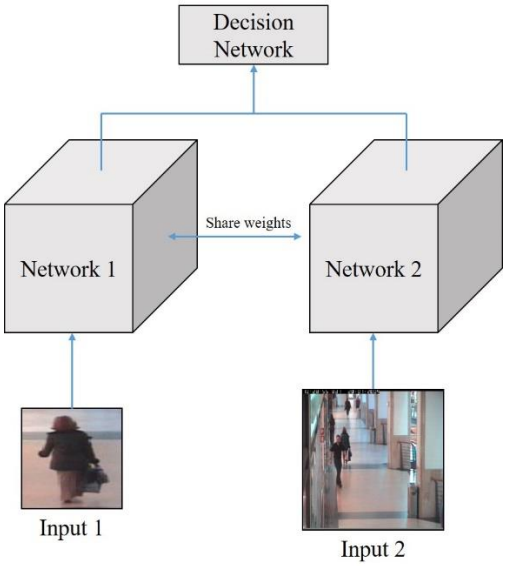


Figure1. A typical Siamese network structure

## 3.2. Tracking Algorithms Based On Siamese Network

The tracking algorithm based on twin neural network considers the process of tracking objects as a problem of similarity learning. It proposes to learn a mapping function $f(z, x)$. When the target image $z$ is similar to the candidate image $x$, the mapping function returns a higher similarity score, otherwise it returns a lower similarity score. In order to find the new position of the target in the new frame, we need to test all possible positions and select the position in the candidate image with the greatest similarity to the tracking object as the new tracking result. Similarity mapping function $f(z, x)$ is obtained by training and learning.

In tracking phase, get a search area $x$ centered on the center of the previous frame in the current frame, then use the feature extraction module $\varphi$ (here is the convolutional neural network) to extract the convolutional features of the search area and the target area in the first frame. The mapping function $f(z, x)$ is realized through convolution operation. And then the similarity score map can be obtained, where the position corresponding to the maximum score is the new location of the target.

The specific steps could be divided into two parts:

1) Feature extraction of the input target in first frame and the current frame using the Siamese network, serving as $\varphi_z$ and $\varphi_x$;
2) Use $\varphi_z$ and $\varphi_x$ for feature matching, and to find the feature location with the highest feature matching score. The specific matching process is implemented with convolution operation.

In the training and tracking process, the network input is an image pair containing a large image and a small image. The small image represents the real marking box in first frame (Examplar), while the large image represents the search area in current frame (Instance). Examplar extraction process takes the center of the real box as the center and extracts a box of 127*127 size. When the extraction area exceeds the image, it is filled with the average RGB value of the image. Similarly, the extraction process of Instance is in the current frame. The target center of the previous frame is set as new center, and an image area of 255*255 is extracted, and the excess part is also filled with the average RGB value of the image. The output of the network is a 17 by 17 score map, and

each position has a score (probability value) referring to the current position as the new target center position. More accurate target center position can be obtained by adopting appropriate processing methods later.

### 3.3. TriT

Although the tracking algorithm based on Siamese network can well deal with some occlusion and scale changes, it is easy to fail when the background of the tracking scene is complex and there are many similar objects interfering. This kind of algorithm can distinguish the differences between different kinds of objects well, but cannot distinguish the differences between the same kinds of objects well, so it is easy to fail in tracking in some scenarios. For example, when the background is more complex or there are more similar objects interfering, such algorithm will regard the interfering object as the object to be tracked due to the lack of discrimination ability of similar objects, leading to tracking failure. Therefore, aiming at the deficiency of Siamese network, this paper proposes an improved algorithm TriT based on the network structure of Triplet network, which can simultaneously combine the first frame and the previous frame of video to comprehensively judge the current tracking results and reduce the influence of complex background and similar object interference on the tracking algorithm.

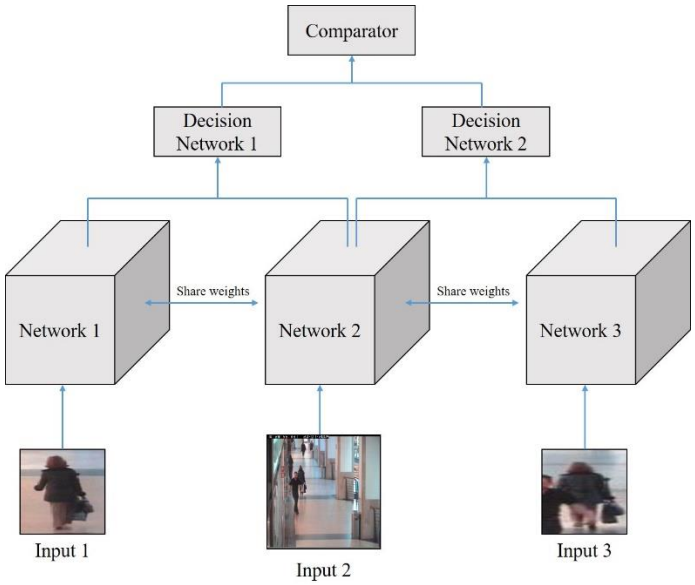


Figure 2. A Triplet network structure

Triplet network is a parallel network structure composed of three sub-neural networks. These parallel neural networks have the same network structure and the same parameters and weights. As shown in Figure 2, the Triplet network is very similar to the Siamese network, and the entire network structure can also be divided into two main parts as follows:

1) Feature extraction network: Three branch networks extract features from three input images respectively. The most commonly used network structure of feature extraction network is the classical convolutional neural network, such as LeNet model [23], AlexNet model [24] and VGGNet model. You can also customize some specific network structures for the feature extraction network here according to specific scenarios.
2) Decision network: The main function of decision network is to further process the output features of the feature extraction network to obtain a specific form of output.

In the TriT tracking model proposed in this paper, the similarity between the target in the first frame and the target in the previous frame and the search area of the current frame is calculated at the same time, and the target location is determined by merging the two score maps. The algorithm flow is shown in Figure 3.

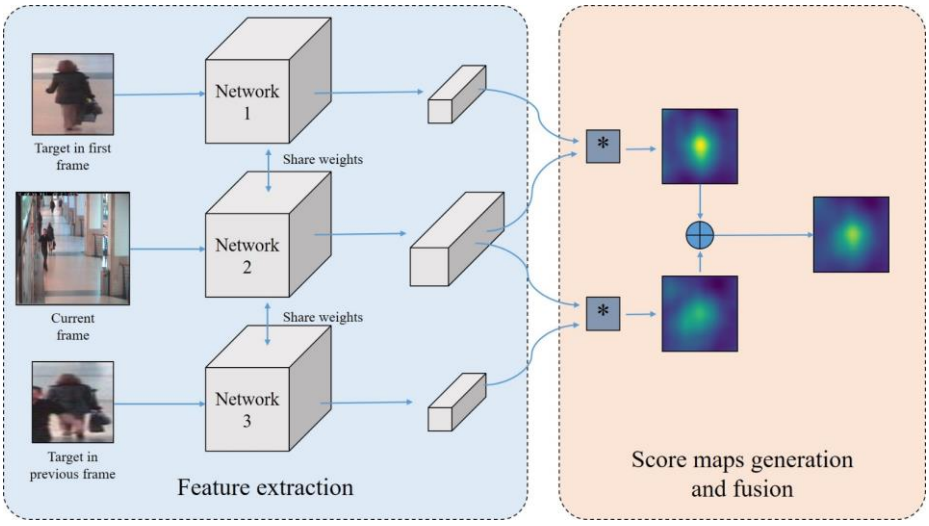


Figure 3. TriT tracking algorithm network structure

**Feature Extraction.** With respect to the given three inputs, namely the first frame target area z, the previous frame target area $z'$and the current frame search area x, we use three same fully convolutional neural networks to execute the feature extraction. Then we can get the feature output $\varphi(z)$、$\varphi(z')$and $\varphi(x)$. The three networks appear as three parallel network structures in the model.

The advantage of fully convolutional network is that we can provide a larger search image as the input of the network, rather than the candidate images of the same size. It can calculate the similarity between all candidate sub-windows and the tracking target in one evaluation and output the result as matrix. In fact, the weights of the fully connected layer in common CNN can be remoulded into the convolutional kernel of the convolutional layer, and the fully connected layer can be transformed into the convolutional layer, so as to realize the fully convolutional network.

The feature extraction network in this paper refers to the AlexNet network structure proposed by Krizhevsky et al in [24], as shown in Fig. 4. The first convolutional layer, conv1, uses a large convolution kernel whose size is 11*11, conv2 uses the convolution kernel whose size is 5*5, and conv3, conv4, conv5 uses a small convolution kernel of 3*3. The purpose of adopting such network structure is that to use a large convolution kernel in the shallow convolutional layer can quickly reduce the feature dimension and increase the receptive field, while in the deeper convolutional layer, the input feature is not too large, so the smaller convolution kernel is adopted to obtain richer semantic information. In addition, a 3*3 pooling layer is added after conv1 and conv2, with the maximum pooling and step size of 2, for further reducing the feature dimensions and maintaining the rotation invariance of the features. Except for the last layer, the network output of each layer is processed by ReLU activation function. The padding operation is not applied to the input of each layer.

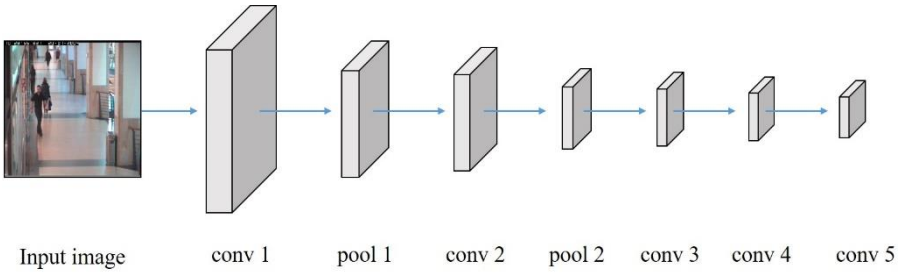


Figure 4. The feature extraction network in TriT

**Training Process.** The loss function uses logistic loss, in the form of:

$$\ell(y, v) = log(1 + exp(-yv)) \quad (1)$$

Where v represents the score of the corresponding candidate regions, $y\epsilon\{+1, -1\}$, represents its real label. During the training, take the processing of the target in the first frameand the input image as an example. After feature extraction network, multiple candidate boxes in the input image will get multiple confidence scores, and then output them in the form of score mapv: $D \rightarrow R$. Finally, the average value of logistic loss of each candidate box will be adopted as the total loss function in the form as follows:

$$L(y, v) = \frac{1}{D}\sum_{u\epsilon D} \ell(y[u], v[u]) \quad (2)$$

Where y[u]and v[u] represent the real label of position u in the input image and the confidence score calculated by network.

In the training process, the stochastic gradient descent method is used to optimize the network parameters.

**Score Map Generation.** After feature extraction of the inputs through the fully convolutional network, the location of the target in current frame can be calculated by feature matching. We take calculating the similarity between $\varphi(z)$ and $\varphi(x)$ as an example, we multiplied the corresponding positions of the first small area of 6*6*128 of $\varphi(x)$ and $\varphi(z)$ of size 6*6*128 and then summed it, namely the cube convolution operation. And then we can get a similarity value, representing the similarity of the first region of $\varphi(x)$and $\varphi(z)$. In turn, the calculation of the similarity of all 6*6*128 in $\varphi(x)$and $\varphi(z)$ will lead to a similarity score map m1. Similarly, the calculation process was used to obtain a score map m2 with the similarity score of$\varphi(z')$and $\varphi(x)$. This calculation process is similar to the convolution operation of image, but it is changed from 2D to 3D. Therefore, the convolution calculation method in the convolutional neural network can be directly used for rapid implementation.

**Score Map Fusion.** In last section, the similarity score maps m1 and m2 are obtained by the method of convolution. Because the dimensions of extracted features are small, the score map generated is also small. This is not conducive to accurate locating. Therefore, the interpolation algorithm is firstly used to enlarge the score map to a larger dimension. In this paper, the bicuric interpolation algorithm is adopted to enlarge the score map of 17*17 by 16 times to 272*272, resulting in the enlarged score maps M1 and M2. Finally, the final score map M is obtained by merging the two score maps:

$$M = \lambda * M1 + (1 - \lambda) * M2 \quad (3)$$

Where $\lambda$ represents the weight of the score map. The peak position of M is the target position calculated by the network.

In this paper, $\lambda$ is 0.5.

## 4. EXPERIMENT AND ANALYSIS

In order to verify the effectiveness of the tracking algorithm TriT proposed in this paper, relevant experiments were carried out on the target tracking data set OTB100[2], and a total of 94 video sequences were tested. At the same time, the comparison experiment with the current mainstream and well-worked algorithms is carried out to draw a more convincing conclusion. Experimental environment: Ubuntu16.04 system, Intel Core i7 7800X processor (3.5ghz), 48GB of memory, NVIDIA GeForce GTX 1080Ti graphics card, TensorFlow1.4 deep learning framework, and Python programming language. In this paper, the threshold of overlap rate to judge whether it is a successful tracking is set as 0.5.

The OTB100 dataset classifies video sequences according to the challenging factors in visual target tracking, as shown in Table 1.

Table 1. Challenging factors in visual tracking.

| Factor | Description |
|---|---|
| IV | Illumination Variation |
| SV | Scale Variation |
| OCC | Occlusion |
| DEF | Deformation |
| OV | Out-of-View |
| BC | Background Clusters |
| LR | Low Resolution |
| FM | Fast Motion |
| MB | Motion Blur |
| IPR | In-Plane-Rotation |
| OPR | Out-Plane-Rotation |

### 4.1. Tracking Effect Evaluation Indicators

In this paper, two indicators are adopted to measure the experimental effect: Distance Precision (DP) and Overlap Precision (OP). DP is defined as follows: in a video sequence, the proportion of the number of frames in the video sequence whose average Euclidean distance between the tracking target location center and the real target center (marking value) is less than the set threshold. In this experiment, the threshold is set to 20 pixels. OP is defined as follows:

$$\text{score} = \frac{A_g \cap A_p}{A_g \cup A_p} \tag{4}$$

The OP reflects the overlap between the calculated location and its real location. In Equation (4), $A_g$ represents the real position of the tracking target in the image, $A_p$ represents the tracking target position output by the algorithm. The values of $A_g$ and $A_p$ are the area of the rectangular box. And the score reflects the overlap degree. The higher the value is, the higher the tracking accuracy is. Schematic diagram is shown in Figure 5.

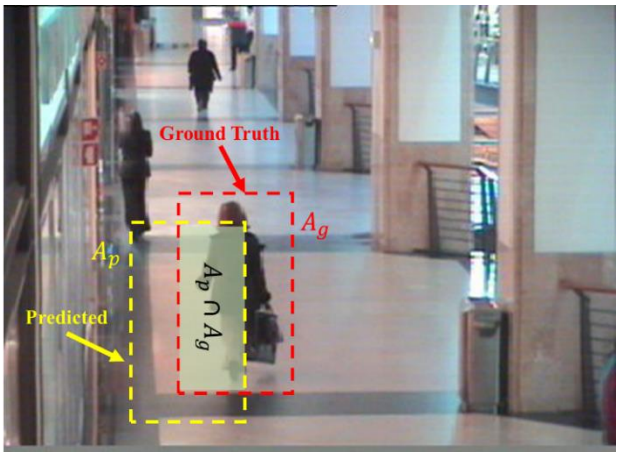


Figure 5. Schematic diagram of overlap precision

## 4.2. Comparison Experiment Between Trit and Siamese-Fc

The TriT tracking algorithm proposed in this paper is improved on the basis of Siamese-fc algorithm. In order to verify the effectiveness and improvement of TriT, the experimental results of comparison with the Siamese-fc algorithm are firstly analyzed.

To verify the robustness of TriT tracking algorithm, a data set containing all the challenging factors in target tracking was tested and analyzed. In Girl2, Basketball, Walking2 and Soccer video sequences, not only the appearance of the target is distorted, but also interference objects very similar to tracking targets appear in video. Figure 6 lists the comparison of TriT and Siamese-fc in Girl2, Basketball, Walking2 and Soccer video sequences respectively.

In the Girl2 video sequence, similar types of tracking target interference are generated in the image as the girl being followed passes right to left through the adult on the right. But since the little girl was not covered, both Siamese-fc and TriT were able to effectively track her. At around the 100$^{th}$ frame, the little girl was blocked by passers-by. After that, Siamese-fc misjudged the tracking object, but TriT was still able to effectively track the girl. In the Basketball video sequence, in 471$^{st}$ frame, the positions of two Basketball players with similar appearance overlap and then staggered. At this time, the Siamese-fc algorithm makes a misjudgment, treating the other player as the tracking player. By contrast, TriT does not make a misjudgment, and can still effectively track the original tracking object. In the Walking2 video sequence, a similar situation occurring in the Basketball video sequence occurs again. When a man similar to the woman tracked appears in the image, TriT can still effectively track the target, but the Siamese-fc algorithm misjudges. In Soccer video sequence, the red celebration ribbon and the scene lights have a very big interference to the tracking face. Take 292$^{nd}$ frame and 350$^{th}$ frame as examples, it can be seen that Siamese-fc algorithm wrongly locates the tracking object on another face and the cup, while TriT does not misjudge the tracking object. In addition, TriT can also be found to have a significantly higher tracking overlap accuracy than Siamese-fc in the frame where no tracking target is lost or misjudged.

Since the input of TriT algorithm contains not only the tracking information of the target in the first frame, but also the tracking information of the network output in the previous frame. On the one hand, the distortion of the target in the current frame relative to the target in the previous frame is smaller than the distortion of the target in the current frame relative to the target in the first frame. On the other hand, with the location of the target in the previous frame, the algorithm is not easy to misjudge even if there are multiple interference objects similar to the target in the

background. Therefore, TriT theoretically has more robust tracking performance than Siamese-fc, which is verified by our experiments.


Figure 6. Comparison of TriT and Siamese-fc tracking performance in Girl2, Basketball, Walking2 and Soccer video sequences in OTB100 dataset. The yellow box represents the real location of the tracking object, the blue box represents the location marked by the TriT algorithm, and the red box represents the location marked by the Siamese-fc algorithm..

## 4.3. Quantitative Comparisons

Section In order to comprehensively evaluate the performance of TriT algorithm, experiments are carried out on the OTB100 data set. With 94 video participating in the test, the DP and OP values of the algorithm are obtained and compared with the Siamese-fc algorithm and other mainstream real-time target tracking algorithms, including LCT[25], Staple[26], KCF[27] and Struck[28] algorithms. Eight representative video sequences were selected in the experiment, and the performance of each algorithm was compared. The experimental results were shown in Table 2 and Table 3.

Table 2. DP errors of TriT and other mainstream real-time tracking algorithms in the OTB100 data set (in pixels).

| | **TriT** | **Siamese-fc** | **LCT** | **Staple** | **KCF** | **Struck** |
|---|---|---|---|---|---|---|
| **Tiger2** | 11.3 | 25.3 | 16.7 | 13.7 | 45.1 | 20.3 |
| **Bird1** | 12.2 | 146.8 | 100.7 | 58.7 | 142.1 | 145.0 |
| **Soccer** | 14.8 | 47.4 | 62.3 | 68.6 | 39.2 | 81.2 |
| **Box** | 21.5 | 26.9 | 151.9 | 56.3 | 90.0 | 120.7 |
| **CarScale** | 16.2 | 5.3 | 53.2 | 33.1 | 88.0 | 101.6 |
| **Couple** | 15.1 | 5.1 | 19.0 | 28.5 | 44.9 | 23.2 |
| **Jump** | 48.2 | 57.8 | 165.4 | 189.5 | 129.6 | 135.7 |
| **Skating2-1** | 30.5 | 48.8 | 36.9 | 53.2 | 43.1 | 38.1 |

Table 3. OP rate of TriT and other mainstream real-time tracking algorithms in the OTB100 data set (in percent).

| | **TriT** | **Siamese-fc** | **LCT** | **Staple** | **KCF** | **Struck** |
|---|---|---|---|---|---|---|
| **Tiger2** | 60.4 | 44.6 | 56.9 | 61.4 | 31.5 | 49.0 |
| **Bird1** | 45.3 | 14.1 | 20.4 | 26.1 | 4.8 | 8.3 |
| **Soccer** | 54.8 | 21.3 | 12.6 | 20.2 | 39.4 | 17.3 |
| **Box** | 59.7 | 59.3 | 9.9 | 33.7 | 28.6 | 19.4 |
| **CarScale** | 76.9 | 77.0 | 67.9 | 76.0 | 41.7 | 41.1 |
| **Couple** | 59.3 | 68.7 | 41.7 | 51.4 | 19.6 | 50.9 |
| **Jump** | 28.6 | 20.9 | 4.3 | 4.8 | 8.5 | 9.6 |
| **Skating2-1** | 61.4 | 29.7 | 55.2 | 39.3 | 51.8 | 54.1 |

In Table 2 and Table 3, the sequence of CarScale, Couple and Tiger2 video sequences represents tracking in a relatively simple environment. Although the appearance of the target changes greatly, there is less interference such as objects similar to the tracked object occurring in the background. The performance of TriT algorithm is close to that of Siamese-fc algorithm, but with a slight lead in most video sequences and a performance advantage over other non-Siamese network methods in most cases. In video sequences with complex backgrounds represented by Soccer and Bird1, the tracking algorithm receives disturbances such as objects with similar appearance of tracking objects, constantly changing backgrounds, large deformation and fast change of tracking objects themselves. At this time, TriT algorithm shows great advantages over Siamese-fc and other algorithms in terms of center point error and overlap rate.

In terms of tracking speed, under the experimental conditions in this paper, the average frame rate of TriT and Siamese-fc tracking is shown in Table 4.

Table 4. Comparison of running speed between TriT and Siamese-fc (frame/second).

| **Tracker** | TriT | Siamese-fc |
|---|---|---|
| **Average speed** | 52 | 61 |

As can be seen from Table 4, TriT algorithm has a slower tracking speed, but it can still meet the requirements of real-time tracking in most scenarios. From the analysis of network structure, in the phase of feature extraction, TriT has 50% more computational load than Siamese-fc, so the tracking speed is slower than Siamese-fc.

## 5. CONCLUSIONS

Aiming at the problem of visual tracking in complex environment, this paper proposes a highly robust target tracking algorithm TriT. Based on the Siamese-fc algorithm, TriT adds the information of tracking target output by the previous frame to the input of the algorithm, and adopts three parallel fully convolutional neural networks for feature extraction, which is equivalent to two parallel Siamese-fc. Then the two score graphs are fused to determine the location of the tracking object in the current frame. Experiments on OTB100 data set show that TriT algorithm can still perform very robust tracking in complex environments such as illumination change, tracking object appearance change and occlusion. By contrast, Siamese-fc algorithm without the previous frame as input is very easy to misjudge the tracking object in the tracking process under a complex background. And the center point error of the target position and overlap rate calculated by TriT in the tracking process are generally better than that of Siamese-fc. TriT's tracking speed is slower than Siamese-fc due to the additional way of input, but our experiments show that TriT can still meet the requirements of real-time tracking in general tracking scenarios.

This paper mainly provides a new idea of visual tracking algorithm. Due to the simple network structure, the experimental effect is not as good as the best tracking algorithm. In later work, the method of fusing correlation filtering can be considered to update the model online, which will make the tracking algorithm more robust. Meanwhile, the feature extraction network in TriT can be improved to improve the speed of the algorithm. More reasonable loss function design can also improve the robustness of the algorithm.

## REFERENCES

[1] Smeulders, A. W., Chu, D. M., Cucchiara, R., Calderara, S., Dehghan, A., & Shah, M. (2013). Visual tracking: An experimental survey. *IEEE transactions on pattern analysis and machine intelligence*, *36*(7), 1442-1468.

[2] Wu, Y. , Lim, J. , & Yang, M. H. . (2013). Online Object Tracking: A Benchmark. *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE.

[3] Lucas, B. D. , & Kanade, T. . (1997). An Iterative Image Registration Technique with an Application toStereo Vision. *Proceedings of the $7^th$ International Joint Conference on ArtificialIntelligence*. Morgan Kaufmann Publishers Inc.

[4] Comaniciu, D. , Ramesh, V. , & Meer, P. . (2002). Real-time tracking of non-rigid objects using mean shift. *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No.PR00662)*. IEEE.

[5] Ross, D. A. , Lim, J. , Lin, R. S. , & Yang, M. H. . (2008). Incremental learning for robust visual tracking. *International Journal of Computer Vision,77*(1-3), 125-141.

[6] Nguyen, H. T. , & Smeulders, A. W. M. . (2006). Robust tracking using foreground-background texture discrimination. *International Journal of Computer Vision,69*(3), 277-293.

[7] Pan, J. , & Hu, B. . (2007). Robust Occlusion Handling in Object Tracking. *IEEE Conference on Computer Vision & Pattern Recognition*. IEEE.

[8] Bertinetto, L. , Valmadre, J. , Henriques, João F., Vedaldi, A. , & Torr, P. H. S. . (2016). Fully-convolutional siamese networks for object tracking.

[9] Hoffer, E. , & Ailon, N. . (2014). Deep metric learning using triplet network.

[10] Wang, M. , Liu, Y. , & Huang, Z. . (2017). Large margin object tracking with circulant feature maps.

[11] Dai, Y. , & Liu, B. . (2015). Robust video object tracking using particle filter with likelihood based feature fusion and adaptive template updating. *Computer Science*.

[12] Bolme, D. S. , Beveridge, J. R. , Draper, B. A. , & Lui, Y. M. . (2010). Visual object tracking using adaptive correlation filters. *The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010, San Francisco, CA, USA, 13-18 June 2010*. IEEE.

[13] Anfeng He∗†, Chong Luo‡, Xinmei Tian†, & Wenjun Zeng‡. (2018). A twofold siamese network for real-time object tracking.

[14] Danelljan, M., Robinson, A., Khan, F. S., & Felsberg, M. (2016, October). Beyond correlation filters: Learning continuous convolution operators for visual tracking. In *European Conference on Computer Vision* (pp. 472-488). Springer, Cham.

[15] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[16] Mansoori, S. A. H. , Mirza, B. , & Fazel, M. . (2015). Hessian matrix, specific heats, nambu brackets, and thermodynamic geometry. *Journal of High Energy Physics,2015*(4), 115.

[17] Danelljan, M., Bhat, G., Shahbaz Khan, F., & Felsberg, M. (2017). Eco: Efficient convolution operators for tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6638-6646).

[18] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Berg, A. C. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, *115*(3), 211-252.

[19] Anfeng He∗†, Chong Luo‡, Xinmei Tian†, & Wenjun Zeng‡. (2018). A twofold siamese network for real-time object tracking.

[20] Wang, Q., Teng, Z., Xing, J., Gao, J., Hu, W., & Maybank, S. (2018). Learning attentions: residual attentional siamese network for high performance online visual tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4854-4863).

[21] Hong, S., You, T., Kwak, S., & Han, B. (2015, June). Online tracking by learning discriminative saliency map with convolutional neural network. In *International conference on machine learning* (pp. 597-606).

[22] Hu, J., Lu, J., & Tan, Y. P. (2014). Discriminative deep metric learning for face verification in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1875-1882).

[23] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324.

[24] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).

[25] Ma, C., Yang, X., Zhang, C., & Yang, M. H. (2015). Long-term correlation tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5388-5396).

[26] Bertinetto, L., Valmadre, J., Golodetz, S., Miksik, O., & Torr, P. H. (2016). Staple: Complementary learners for real-time tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1401-1409).

[27] Henriques, J. F., Caseiro, R., Martins, P., & Batista, J. (2014). High-speed tracking with kernelized correlation filters. *IEEE transactions on pattern analysis and machine intelligence*, *37*(3), 583-596.

[28] Hare, S., Golodetz, S., Saffari, A., Vineet, V., Cheng, M. M., Hicks, S. L., & Torr, P. H. (2015). Struck: Structured output tracking with kernels. *IEEE transactions on pattern analysis and machine intelligence*, *38*(10), 2096-2109.