electronics

MDPI

Article

Anti-Similar Visual Target Tracking Algorithm Based on Filter Peak Guidance and Fusion Network

Jing Wang [1,*], Yuan Wei [1], Xueyi Wu [1], Weichao Huang [2] and Lu Yu [1]

[1] School of Printing, Packaging and Digital Media, Xi'an University of Technology, Xi'an 710048, China; weiyuanxaut@163.com (Y.W.)

[2] School of Automation and Information Engineering, Xi'an University of Technology, Xi'an 710048, China; huangwc@xaut.edu.cn

[*] Correspondence: wangjing63@xaut.edu.cn

Abstract: Visual tracking is a key research area in computer vision, as tracking technology is increasingly being applied in daily life, it has high-research significance. Visual tracking technology usually faces various challenging interference factors, among which, a similar background is one of the factors that has a greater impact on the tracking process. Kernelized Correlation Filter (KCF) tracking algorithm can track targets quickly by using circulant matrix, and has good tracking effect, so it is widely used in the tracking field. However, when the target is interfered by similar objects, the filter template in KCF cannot effectively distinguish between the target and the interfering object. This is because the filter only uses the texture gradient feature as the description object of the target, which will make the KCF algorithm extremely sensitive to the change of the target; therefore, the filter has difficultly making a judgment in the unstable scene, cannot accurately describe the target state, and finally leads to tracking failure. Therefore, this paper fuses Color Names (CN) on the basis of the original Histogram of Oriented Gradients (HOG) feature of KCF, which can obtain a more comprehensive feature representation, and realize the application of combined features to improve the anti-interference ability of KCF in complex scenes. In addition, this paper also uses the peak response of correlation filtering as the judgment condition to determine whether the current tracking result is stable. When the filter is in an unstable tracking state, the proposed algorithm will select the value with high confidence from its multiple responses as the candidate target of the Siamese network, and the deep learning network is used as the incremental learning method of the filter. The Channel Attention is introduced into the network layer, so that the network can adaptively reason and adjust the extracted universal features, and the enhanced feature information is used as the final discriminant basis. Finally, according to the response, the target with the smallest error compared with the target template is selected from multiple candidate targets as the final tracking result. The experimental results show that the average accuracy and average success rate of the proposed algorithm are significantly improved compared with the classical tracking algorithm, especially in dealing with similar target interference.

Keywords: visual tracking; kernel correlation filter; Siamese network; Channel Attention; anti-similarity

check for updates

Citation: Wang, J.; Wei, Y.; Wu, X.; Huang, W.; Yu, L. Anti-Similar Visual Target Tracking Algorithm Based on Filter Peak Guidance and Fusion Network. *Electronics* **2023**, *12*, 2992. https://doi.org/10.3390/electronics12132992

Academic Editor: Dah-Jye Lee

Received: 28 May 2023
Revised: 22 June 2023
Accepted: 5 July 2023
Published: 7 July 2023

1. Introduction

The visual tracking algorithm predicts the position and motion state of the target in the current video or image sequence by modeling the appearance and motion information of the target [1]. Visual tracking technology has been widely used in intelligent video surveillance, unmanned driving, virtual reality, human–computer interaction and other civil and military fields. At present, with the support of hardware equipment, blockchain technology [2] and collaborative sensing technology [3], target tracking is truly developing towards intelligence and popularization. And because of that, the requirements for visual tracking technology are also getting higher and higher. However, the difficulties in the tracking process, such as occlusion, target deformation, and the superposition of similar objects, make the visual tracking algorithm a difficult and hot issue in the field of vision processing.

In order to balance the accuracy and effectiveness of the visual tracking algorithm, the visual tracking algorithm based on correlation filtering is particularly prominent. The core idea of the correlation filter tracking algorithm is to design a filtering template, which is used to perform correlation operation with the target candidate region, and the position of the maximum output response is the target position of the current frame [4]. Although the correlation filtering algorithm has the advantage of high effectiveness in many trackers, it does not meet the accuracy requirements in actual tracking. This is because the correlation filtering algorithm is sensitive to the appearance change of the target. When the target is deformed, occluded and interfered by the background, these interferences will lead to the inaccurate correlation peak, which will affect the accuracy of the tracking algorithm. An idea to optimize the correlation filter tracking algorithm is to build an accurate target representation model. Although the use of rich combination features can improve the accuracy of the model, it will also reduce the effectiveness of the algorithm. The deep learning network has made significant progress in the field of target tracking. This model can adaptively learn the high-level features of the target, output accurate characteristics for constructing the target model through different levels of the network [5], and can adapt to changes in the appearance of the target. However, deep learning models are usually limited by the quality and quantity of data, and usually require a large amount of computing resources and training data. Therefore, this paper intends to combine the correlation filtering algorithm and deep learning algorithm, and under the guidance of the correlation filtering tracking results, the deep features are fused to improve the tracking performance. The main contributions of this paper are as follows:

(1) In this paper, the response value in the kernel correlation filter theory is used as the judgment mechanism, and the judgment mechanism is designed to realize the cross association with the deep learning network. The multi-peak response generated by the filter provides a high-quality template for the Siamese network to capture multiple features about the target from the spatial level, so that the tracking algorithm can better adapt to different target characteristics and environmental conditions, so it can provide reliable tracking results.

(2) Based on the Siamese network, the channel Attention Mechanism is introduced to weight the extracted universal features, so that the features can adaptively focus on the essential information of the target in the process of fusion processing. By fusing the features, the model can comprehensively use the feature information of different levels, and improve the understanding ability and discrimination of the target model.

(3) Considering the effectiveness of the correlation filtering algorithm and the accuracy of the Siamese network, the correlation filtering algorithm tracks first, so that it can adapt to most scenes based on the fusion of color features. In order to avoid the accumulation of errors, it is determined whether the Siamese network is selected as the secondary learning mechanism to find out the potential position of the target in the image. The experimental results show that the proposed algorithm performs well, especially when the target is disturbed by similar objects.

2. Related Works

2.1. Correlation Filter-Based Trackers

Correlation filtering is a signal processing technique used to extract features or information of interest from a signal. It operates based on the correlation between the signal and a filter. The correlation filter is a window function that slides over the signal and calculates the correlation between the signal and the filter. Its basic principle is to determine whether there are features in the signal that are similar to the filter by calculating the dot product or correlation between the signal and the filter. If the signal is highly correlated with the filter, the output will have a large value; if the correlation is low, the output will be smaller. This technique is widely used in many fields, including image processing, audio processing, video processing and so on.

The early kernel correlation filtering algorithm has a relatively stable tracking effect In most motion scenes. The algorithm carries out an internal update iteration in the form of single sample closure to achieve a high level of tracking efficiency. Instead of integrating multiple samples for training, they obtain specific sample information from single samples and realize model adaptation by combining the sample information input. The Minimum Output Sum of Squared Error (MOSSE) [6] algorithm introduces a correlation filter in the Fourier domain, and optimizes the filter by minimizing the output square error, which is an efficient algorithm and is mostly used for real-time tracking. The Kernel Correlation Filter (KCF) [7] algorithm uses kernel correlation filter for object tracking, which can improve the accuracy and robustness of object tracking, and uses circulant matrix to accelerate the calculation. By only using the Histogram of Oriented Gradients (HOG) as the feature description, it is easy to be affected by external factors, and the obtained feature representation is very unstable, which is not suitable for complex scenes. The Spatially Regularized Discriminative Correlation Filter (SRDCF) [8] algorithm is a spatially regularized discriminant correlation filter. It uses a feature selection method based on sparse representation, which can effectively suppress noise and interference, and has good performance in complex scenes. ECO [9] introduces multi-party features in the input, uses convolutional network in deep learning to extract image features, and Color space features (Color Names) [10] to capture the color information of the target, and uses the histogram of oriented gradients to describe the texture and edge information of the target, in order to represent the appearance characteristics of the target more comprehensively. On this basis, the online learning method and multi-scale search strategy are applied to update the target model to adapt to the changes in the appearance of the target, which makes the model have good stability.

The above tracking models all use shallow features and show sensitivity to specific actual functions, and it is not enough to explore only the existing scale, from coarse to fine features play an important role in the tracking process [11]. KCFAPCE [12] is applied to APCE confidence as the basis for dynamic adjustment. According to the confidence level, whether the target is lost is judged, and then the update strategy of dynamically adjusting the learning factor is adopted to suppress the influence of low response value on the tracking results. CF_SIAM [13] uses a hybrid target tracking algorithm combining KCF and SiamFC, and uses SiamFC to obtain deep features, which to some extent makes up for the instability of KCF in the processing of target non-rigid changes. The IKPCA-KCF [14] algorithm is an incremental kernel principal component analysis-KCF algorithm, which gradually updates the target model to adapt to the changes of the target appearance by means of incremental learning. This paper summarizes the ideas and methods from the algorithms mentioned above, and makes innovations on the basis of the KCF algorithm. It mainly focuses on the enhancement of appearance information and the analysis of related results, and is dedicated to improving the adaptability and anti-interference of the algorithm.

2.2. Siamese Network-Based Trackers

A Siamese network is a parallel two-branch network, which receives two input features. One is the template image feature, the main content is the tracked object. The other branch is search images, which is what the model mainly identifies and locates. It is a deep learning model that compares the similarity between two inputs, encodes the inputs into a vector representation through a shared subnetwork, and uses a distance metric to measure the difference between them. Many classical Siamese networks are fully convolutional model architectures, which allows the network to effectively capture visual information [15].

In scenes with similar objects, it is a complex task to accurately obtain the target state. The appearance features of the target are easy to be confused by the background, so effective feature extraction and scene analysis are necessary. The Siamese network combined with convolutional neural network adopts the extraction backbone form of fully convolutional structure. SiamFC [16] extracts target state information from shallow features to deep features. This method cannot guarantee effective information acquisition, because the backbone network extracts universal features. If the follow-up tracking task is only conducted according to the initial frame, and the background area or previously tracked frame information is not incorporated into the model prediction, it is difficult to establish a clear distinction between the foreground and the background. Later, Siamese network-based trackers are dedicated to precise target localization by combining anchor points. For example, SiamRPN [17] combined a Siamese network with RPN [18]. In order to improve the anti-interference ability of the model, DaSiamRPN [19] proposed a series of strategies for negative samples, which can carry out interference training consciously. Then, the optimization of the overall structure is proposed from the end-to-end structure, SiamRPN++ [20], which expands the deep spatial information on the basis of the ResNet [21] deep backbone network. SiamMask [22] added mask inference to enhance the discrimination ability of the model. Siam R-CNN [23] combines and compares the extracted regional features multiple times, and avoids the confusion of similar objects and affects the tracking accuracy by re-detection. Since then, although the structure of Siamese network has been innovative, their starting points are all traceability and relatively similar. Focusing on the interaction between channel features or letting the extracted features go through layer-by-layer comparison, information filling [24] and feature fusion [25] has become a new standard, which can make the network better equipped to deal with visual information in complex scenes.

2.3. Attention Mechanism

The Attention Mechanism is a computational model that simulates the human Attention Mechanism and is used in the way input data is processed in machine learning and deep learning. It works by assigning different weights or attention to different parts of the input in order to focus more on the important information during processing. In the Attention Mechanism, input data is usually represented as a sequence of vectors or features. The Attention Mechanism determines the importance of each input by calculating the similarity between each input vector and a learnable weight vector. These similarities are usually obtained by computing inner products or using other similarity measures.

The Attention Mechanism (AM) [26] was introduced into the field of machine learning and natural language processing in 2014, which enables the model to assign different attention weights according to different parts of the input sequence, in order to better pay attention to important information and improve the model's ability to process and express data. The channel Attention Mechanism [27] can increase the weight of feature channels related to the target object and reduce the weight of other feature channels irrelevant to the target object. VTT [28] and CTT [29] use the Attention Mechanism to integrate the extracted features, which can enhance foreground information, specifically. The SiamTPN [30] extracts the module features across layers, indicating that the interaction between shallow and deep features has a significant impact on exploring target information. The algorithm has strong recognition ability for small targets and can adapt to complex background interference. There are also algorithms [31–36] that apply spatial attention [37] to feature fusion, but this paper pursues real-time tracking, and the huge amount of data will only prolong the process time. Therefore, this paper uses deep neural network as the secondary learning mechanism of kernel correlation filtering; the template set of the search area is filtered in the multi-peak response, the information extracted from the multi-layer network is used to capture the target state, and different levels of attention layer are used to enhance the template information, in order to improve the tracking performance of the algorithm as a whole, thus achieving the level of real-time tracking.

3. Judgment Mechanism Guided by Correlation Filter Response Peaks and Multi-Template Filtering

3.1. Judgment Mechanism

The correlation filtering algorithm trains the correlation filter by extracting the target features, and judges the current search area is the potential location of the target according to the correlation response values. The higher the correlation response value, the higher the similarity degree of the signal features processed, which is the condition that the target can fully appear in the field of view. However, when the features of the target are weakened or not completely displayed, there are obvious drawbacks to local search, which are not conducive to target position inference. Especially in the correlation filtering algorithm, the guide of the filter plays a crucial role. If there is a deviation, the tracking will fail due to the accumulation of errors. This paper selects all candidate targets based on the filtering response values.

As shown in Figure 1, the response value of the tracking target is obtained after filtering, and its maximum value is the center red dot, which also represents the predicted position of the tracking target box.

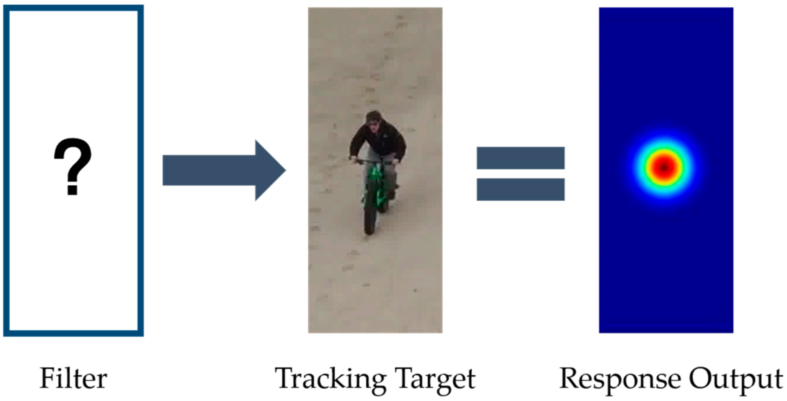
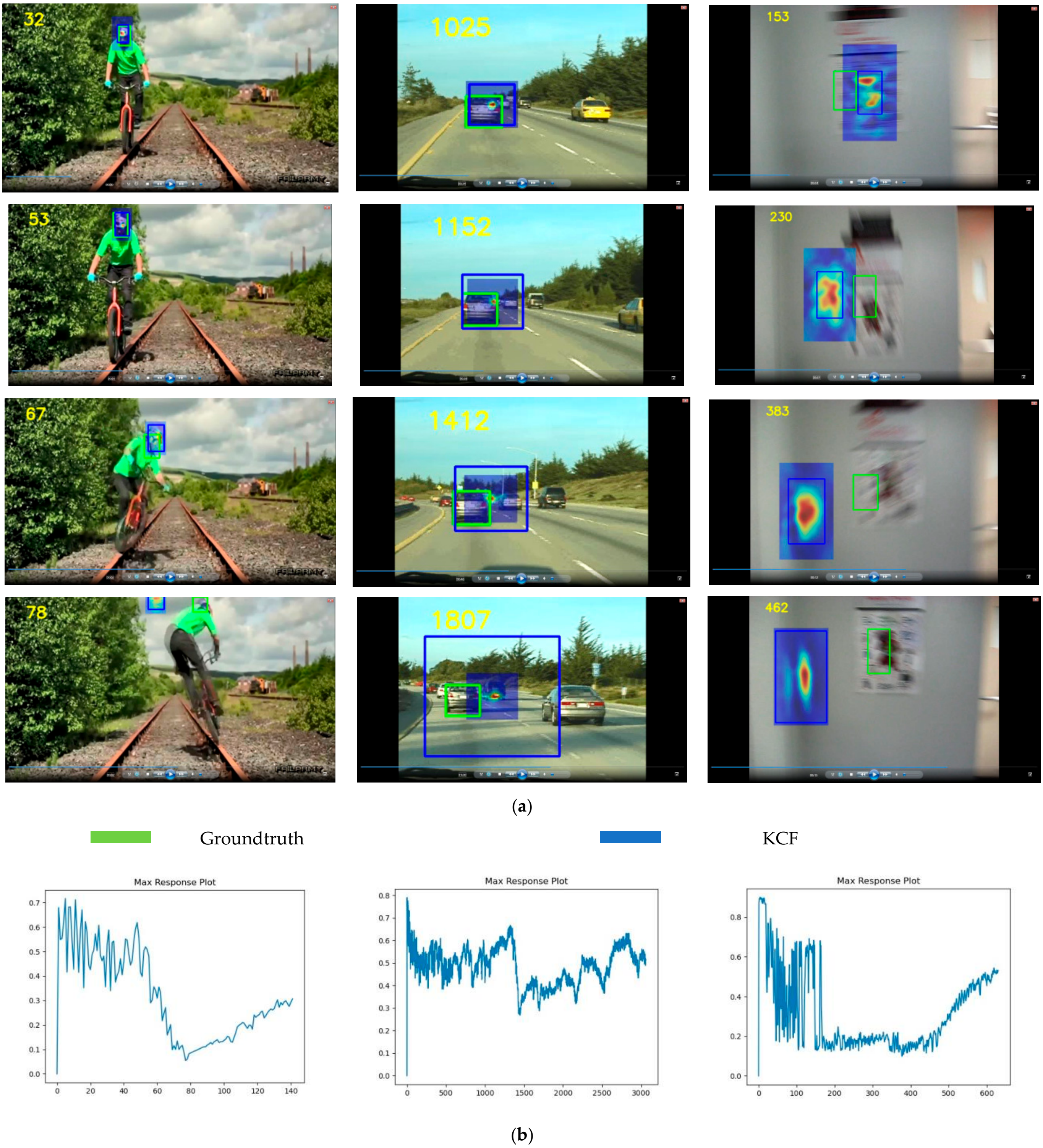


Figure 1. Correspondence between Correlation Filter Response Map and Tracking Box [7].

This experiment saved the maximum response value ($Max\_Response$) for each frame of a video sequence in the OTB-100 [38]. Based on the motion trajectory of the target in the video frames, a curve graph of the changes in the maximum response value was plotted.

In order to analyze the relationship between the correlation filtering response values and tracking results, Figure 2a shows the tracking truth values and KCF tracking results of some video sequences in the OTB-100 datasets, and the change curve of the maximum response value corresponding to each frame of KCF tracking results are shown in Figure 2b. From left to right in the figure, the sequence is: Biker video sequence, which remains stable in the early stage during tracking. When the target is suddenly lost, the maximum response is still shown in the image, but there is a large drop in the graph. In the Car24 video sequence, when the Groundtruth size changes with the target motion, the size of the tracking box stays the same, which leads to the error of video tracking and the sudden decrease of the maximum response value. In the BlurOwl video sequence, the tracking effect is good in the early stage, but the jitter increases later, which leads to the failure of tracking the target, so the Max-Response changes greatly in the later stage.

Therefore, this paper utilizes the variation of the maximum response value $Max\_Response$ in the sequence context as the judgment mechanism for the algorithm guidance condition: $Condition$, as shown in Formula (1). When the maximum response value $Max\_Response \geq \sigma$, the guidance condition is true, which indicates that the current tracking result is reliable, and the KCF-based tracker will continue to be used. When the maximum response value is $Max\_Response < \sigma$, the guidance condition is false, meaning that the current tracking results are unreliable and further network fusion tracking algorithms need to be adopted.



(**a**)

(**b**)

Figure 2. KCF Algorithm Tracking Results and Max-Response Value Change Curve. (**a**) KCF Tracking Results (Green box represents Groundtruth, Blue box represents KCF results); (**b**) Variation curve of Max-Response value.

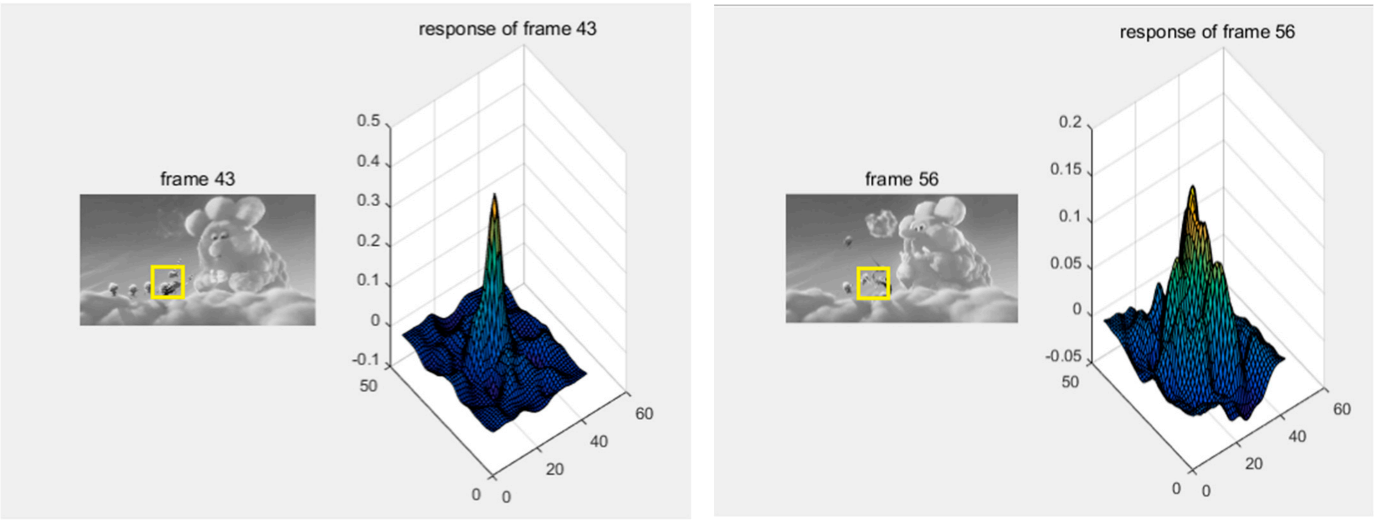
$$Condition = \begin{cases} True, & Max\_Response \geq \sigma \\ False, & Max\_Response < \sigma \end{cases} \tag{1}$$

where σ is the threshold value, and this parameter is set to 0.6 in the experiment.

3.2. Multi-Templates Filtering

The KCF uses the maximum response value to determine the current tracking result. However, once there is interference from similar objects or full occlusion, the maximum response will cause the tracking results to move with similar objects, even leading to tracking lost. When the target is disturbed by similar objects, the corresponding filtering response value will change from the obvious peak value to multiple peaks with relatively close values, as shown in Figure 3. Therefore, in order to reduce the impact of similar object interference, this paper uses multi-response values to select a more accurate target candidate templates T_c to prevent tracking drift and other issues. Assuming R_s represents the response values sorted from high to low, R_s^i is the i-th response value, i can take up to m values, and θ is the difference threshold of the response values, then the filtered multi-templates T_c are obtained by calculating whether the difference between the maximum response value and other response values is within the threshold range, as shown in Formula (2).

$$T_c(i) = \left\{ i \middle| \left| R_s^1 - R_s^i \right| \leq \theta, i = 2 \cdots\cdots m \right\} \tag{2}$$



(**a**) Response of frame 43

(**b**) Response of frame 56

Figure 3. Comparison of Tracking Results and Response Maps at Different Frames on Bird Video Sequence. The yellow box indicates the gtracking target of the current frame.

4. The Proposed Algorithm

The proposed algorithm is mainly composed of KCF and a Siamese network. KCF is an algorithm that can meet the real-time tracking requirements, while the Siamese network uses deep learning features to improve the accuracy of target description. The algorithm uses a peak-guided decision mechanism and adopts the KCF tracking framework when the correlation filter response is good. When similar object interference occurs, the target model features in KCF are difficult to distinguish between the foreground and background; the algorithm introduces a fusion attention Siamese network to improve the ability to distinguish similar targets. In addition, the algorithm uses multi-template filtering to further improve tracking accuracy. An overview of the proposed algorithm is shown as Figure 4.

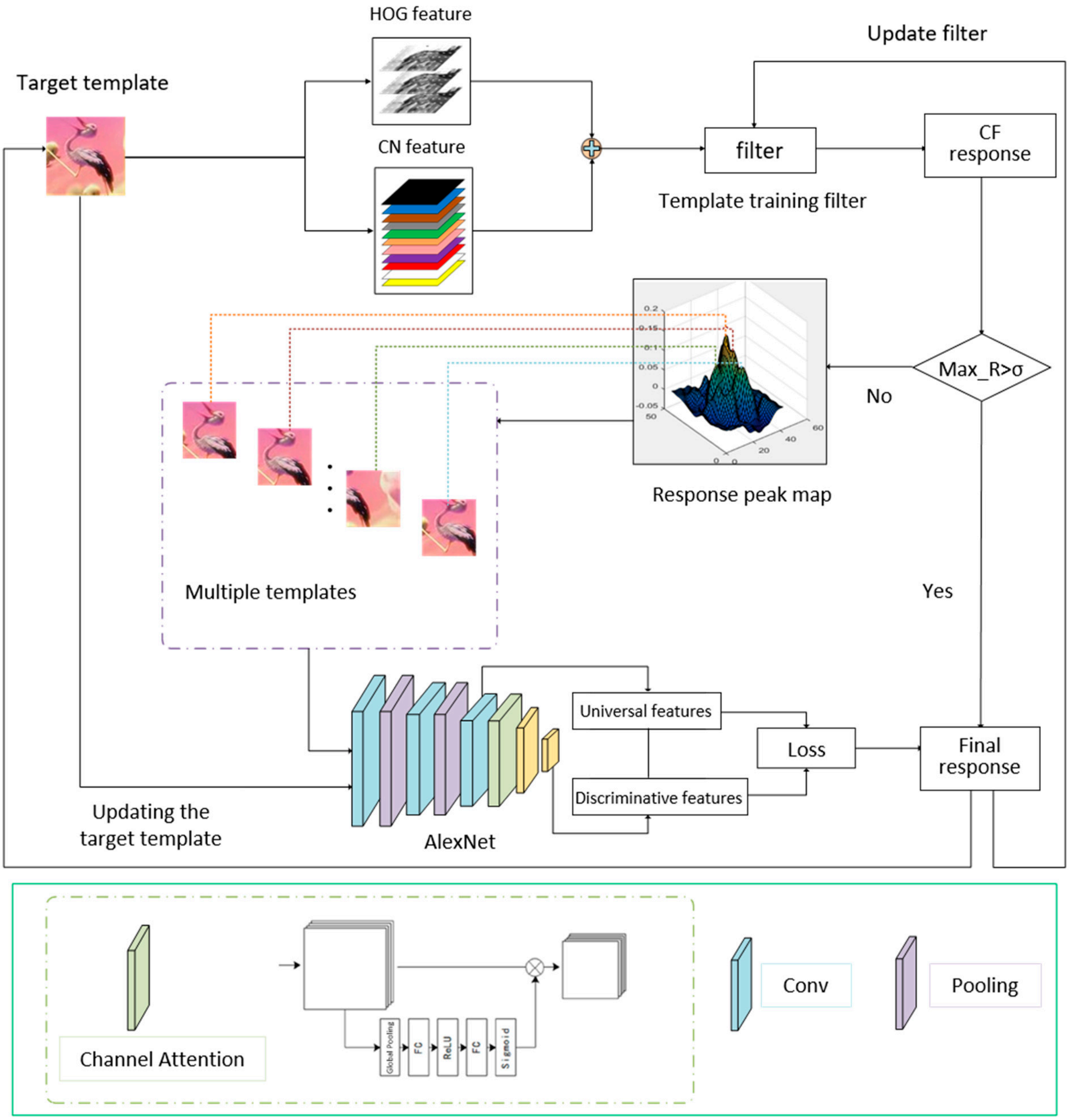


Figure 4. An overview of the proposed algorithm.

4.1. KCF Based on Combined Features

In this paper, the structure of the original kernelized correlation filter (KCF) algorithm is used, and the main improvement is the feature extraction method in the original algorithm. The color space feature is added on the basis of the gradient histogram. The KCF algorithm only uses the HOG feature as the target appearance representation, which has certain limitations. This is because in some complex situations, such as illumination changes, irregular appearance deformation or scale transformation of the target itself [39], unilateral extraction of the HOG gradient histogram feature cannot accurately describe the shape and spatial structure of the target, because the feature is sensitive to the changes of the scene, and it mainly focuses on the local gradient direction [40] in the image.

This paper chooses to incorporate color information as the compensatory information for the target model in kernel correlation filtering. The Color Names feature (CN) utilizes multiple colors to describe the appearance model of the target, and the global nature of this feature can also well-supplement the defects of the HOG feature that only describes the local region of the target. Therefore, this article combines the CN and HOG features to describe the appearance feature of the object, in order to enhance the discrimination ability of the filter. Circulant matrix was used in the subsequent process to reduce the amount of calculation. The kernel space is added to make the data linearly separable, and the tracking response can be calculated quickly and the tracking rate can be improved.

The tracking process applied by the optimized kernel correlation filter algorithm is as follows.

(1) The expression of the appearance model is enhanced, and the HOG and CN features are combined to extract the features of the picture, which can be obtained as

$$xx = \{x_H, x_C\}, \tag{3}$$

where x_H represents the 31 × 1 dimensional feature value of HOG, x_C represents 11 × 1 dimensional feature value of CN, and xx represents the 42 × 1 dimensional combined feature of these two features. With the HOG feature, which is more likely to describe the contour of the target, and the CN feature, which is more likely to describe the color information of the target, the tracking target can be described from both the gradient aspect and the target space feature aspect. The HOG feature can describe the contour of the target and its gradient change, while the CN feature focuses on describing the color information and has an excellent ability to distinguish the deformed target. This method can improve the discrimination accuracy of the filter for the foreground and background in the feature extraction of the target region.

(2) Given a pixel patch (x_i,y_i), the linear regression function is

$$f(x_i) = \omega^{\mathrm{T}} xx_i, \tag{4}$$

(3) Performing a cyclic operation on a vector can result in its cyclic matrix, and the calculation formula is:

$$x = Fdiag(\hat{x})F^H, \tag{5}$$

The filter is solved by substituting x into the circulant matrix:

$$\hat{\omega} = \frac{\hat{x}^* \cdot \hat{y}}{\hat{x}^* \cdot \hat{x} + \lambda}, \tag{6}$$

The response of the filter is obtained as follows:

$$\hat{y} = \hat{x} \cdot \hat{\omega}, \tag{7}$$

(4) When the Gaussian kernel is selected for solving, the kernel function can be obtained as follows:

$$K^{xx} = \exp(-\frac{1}{\sigma^2}\left(\|xx\|^2 + \|xx'\|^2\right) - 2F^{-1}(x\hat{x} \cdot x\hat{x}'^*)), \tag{8}$$

Calculate the filter coefficients as follows:

$$a^{xx*} = \frac{y}{K^{xx} + \lambda I}, \tag{9}$$

Finally, we obtain a quick response:

$$\hat{f} = \hat{k}^{xxz} \cdot \hat{a}^{xx}, \tag{10}$$

4.2. Siamese Network with Attention Fusion

4.2.1. Backbone Network

The SiamFC fully convolutional Siamese object-tracking algorithm mainly uses the AlexNet [41] neural network as the backbone model in its network architecture. AlexNet is a lightweight five-layer convolutional network structure that is simple in structure and easy to apply. Its structure is shown in Table 1:

Table 1. Structure diagram of AlexNet network convolutional embedding.

Layer Name	Convolution Kernel	Stride	Target Template Feature Map Size	Search Template Feature Map Size	Number of Channels
			127 × 127	255 × 255	3
conv1	11 × 11	2	59 × 59	123 × 123	96
pool1	3 × 3	2	29 × 29	61 × 61	96
conv2	5 × 5	1	25 × 25	57 × 57	256
pool2	3 × 3	2	12 × 12	28 × 28	256
conv3	3 × 3	1	10 × 10	26 × 26	192
conv4	3 × 3	1	8 × 8	24 × 24	192
conv5	3 × 3	1	6 × 6	22 × 22	128

The reason why the AlexNet network is selected for the SiamFC tracking algorithm is that object tracking does not require fine classification of the target like object classification or object detection. Instead, it focuses on extracting as much relevant content as possible within the target range. Object tracking is mainly to do with binary classification of targets and non-targets, and the five-layer convolutional neural network of AlexNet is sufficient to meet the parameter requirements in tracking. Moreover, the lightweight network size ensures fast network operation. But the SiamFC algorithm only uses the depth features of the first three layers of AlexNet, which are not precise enough to describe foreground or background and cannot handle interference from similar objects. Therefore, this paper proposes a combination of universal features and discriminative features to distinguish similar targets.

In this tracking algorithm, the target template and search template are input into the following five-layer network for calculation. The features obtained from the first three convolutional layers are selected as universal features (blue dashed box in Figure 5), which can obtain the generalized features of the target. The discriminative features (orange dashed box in Figure 5) are extracted from the universal features using the last two convolutional layers, which can distinguish different similar targets by utilizing the appearance description of deeper features. The response of these two types of features is combined and processed to determine the final position of the target.

4.2.2. Attention Module

Using the features extracted by AlexNet is insufficient to adapt to complex scene tracking, especially excluding the interference of similar objects. To address this problem, the proposed algorithm incorporates channel Attention Mechanism into the Siamese network; Channel Attention is used to process multi-channel input data. According to the correlation between different channels in the input data, the weight of each channel is dynamically adjusted to better capture the important information in the input. In this paper, the output of the third layer of AlexNet is selected as the general feature, and the feature map at this time may contain redundant or irrelevant information. On this basis, the Channel Attention Mechanism is applied, and the model selectively focuses on and weights important channels, in order to realize feature selection and compression. From Figure 6, the specific implementation process of the channel Attention Mechanism can be seen, which extracts features from the input data, uses convolution operation to capture spatial, frequency domain or other types of features, uses global pooling operation to obtain the correlation value of each channel, calculates each channel of the input data according to the correlation value, and uses the activation function to normalize the correlation value into weight. Finally, the weight is applied to each channel of the input data, and the features of different channels are weighted by the multiplication operation to generate the final output.

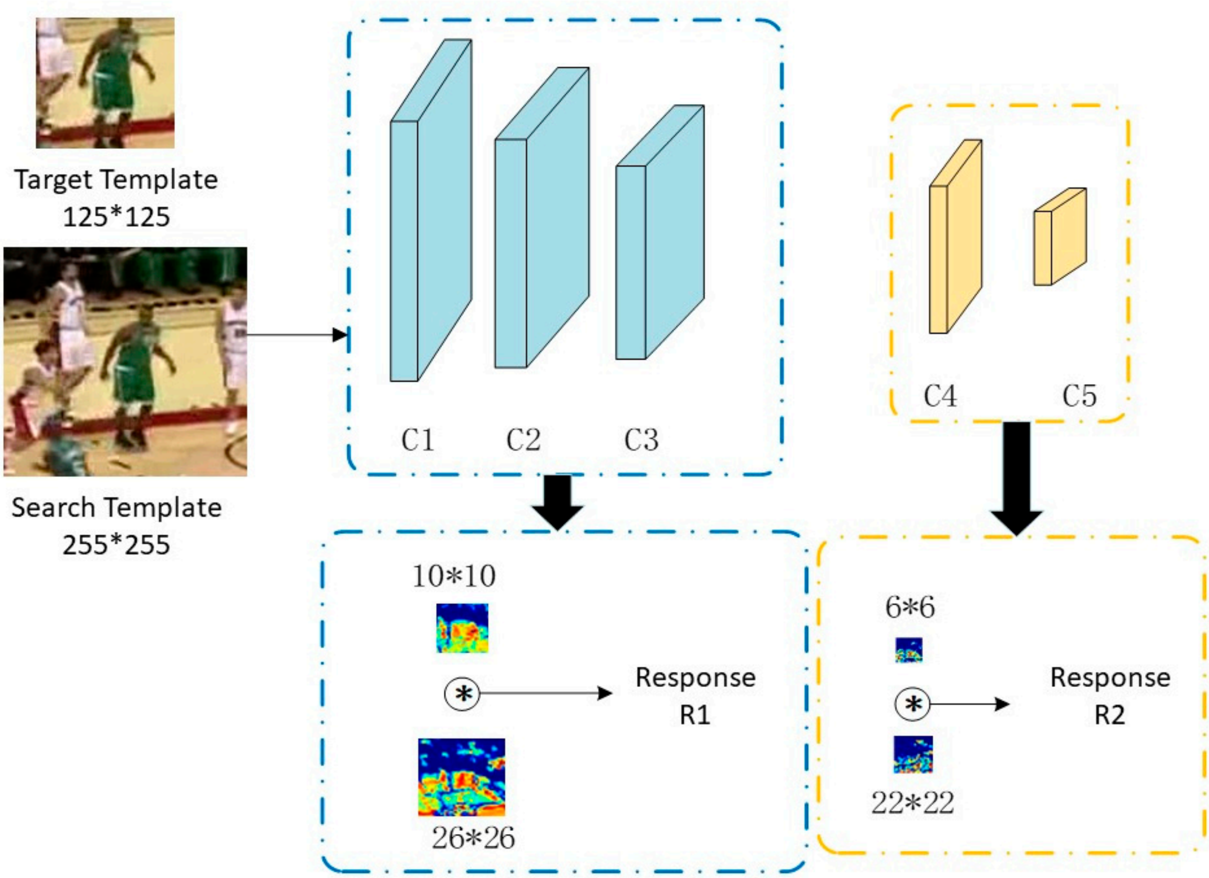


Figure 5. Schematic diagram of universal and discriminative feature extraction.

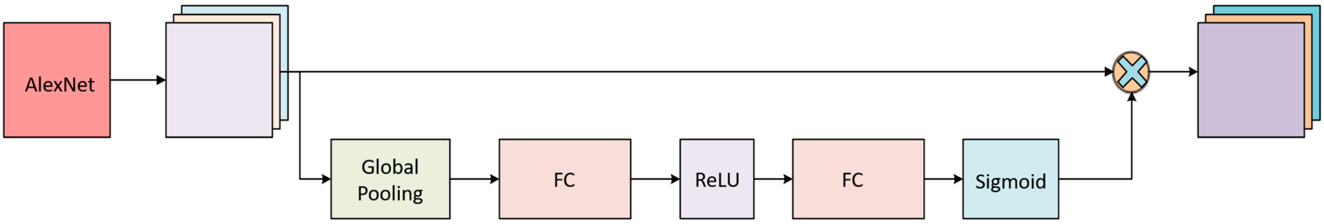


Figure 6. Network Architecture of AlexNet with the Channel Attention Module.

By using the features of Channel Attention to represent the target more accurately and comprehensively, it can adapt to different tasks and input data. The importance of the target location in each channel of the third layer features is more prominent. After the convolution operation of the last two layers of the AlexNet network, the model has extracted a more compact and useful feature representation. Therefore, in this paper, the features of the third layer are used as general features to highlight the target area. The features enhanced by attention are used as discriminative features to obtain the key feature representation of the target itself. Therefore, the Channel Attention layer can adaptively focus on the channel related to the target, according to the context information of the target, improve the discrimination ability and robustness of the target, and improve the interference ability of similar objects.

4.2.3. Activation Function

The Rectified Linear Unit (ReLU) [42] non-saturating neuron was used as the activation function in the network, and its image is shown in Figure 7. ReLU inserts a non-linear factor as a correction unit into the network, and the function is obtained by Formula (11).

$$f(x) = \max(0, x), \tag{11}$$

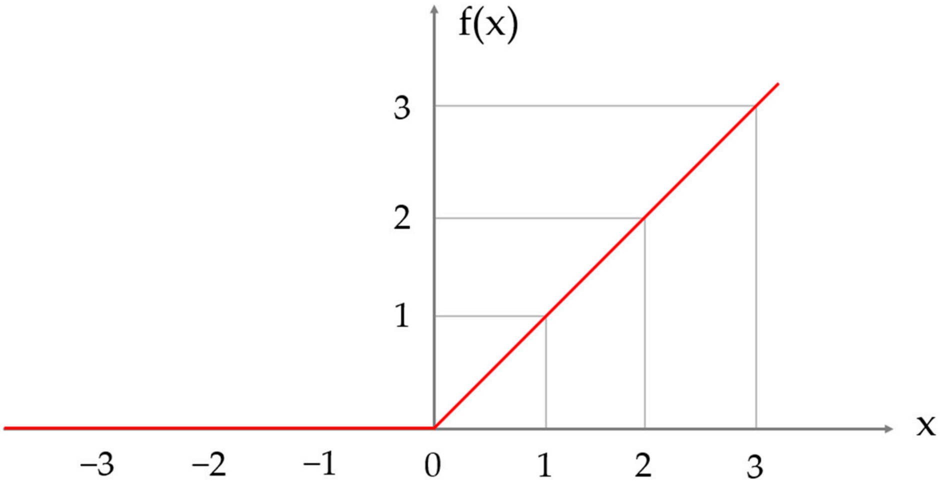


Figure 7. Image of ReLU function.

When the input x is greater than 0, the output is x. When the input x is less than 0, then the output is 0. Compared with the traditional neural network activation function, the ReLU function has more efficient gradient descent and back propagation, and reduces the computational cost by utilizing the sparsity of function activation, resulting in better performance.

4.2.4. The Loss Function

SiamFC network uses Logistic loss function [43], which is defined as follows:

$$l(y, v) = \log(1 + \exp(-yv)), \tag{12}$$

where v represents the response value of a single candidate sample output by the network, and y represents the label of the actual response value Groundtruth, and y $\in [-1, 1]$. The loss of the final score map is defined as the average of the individual losses as follows:

$$L(y, v) = \frac{1}{|D|} \sum_{u \in D} l(y[u], v[u]), \tag{13}$$

D represents the generated heatmap, u for a certain value in D. $|D|$ represents the size of the heatmap. While for the Groundtruth of the heatmap, it is labeled according to the following Formula (14):

$$y[u] = \begin{cases} 1, & k\|u - c\| \le R \\ -1, & otherwise \end{cases}, \tag{14}$$

where c represents the central position of the target in the heatmap, u represents the position of any point in the heatmap, $\|u - c\|$ represents the Euclidean distance between two points u and c, R is the threshold value set for the distance, and k is the reduction factor of the heatmap after passing through the network.

From the network structure, it can be seen that three layers of convolution or pooling have a stride of 2, so the variation of pixels containing object information is reduced by a factor of $2^3 = 8$. Operations with a stride of 1 do not affect pixels containing object information. The final output of the network is actually a discriminative method, trained with positive and negative samples. Each candidate sub-window in the search image x is actually a sample, and its score output is the probability of it being a positive or negative sample. If logistic regression is used to represent it, this is a typical binary classification problem using logistic regression. When the stride of the network is k, if the elements in the score map are within the radius R of the center, they are considered positive samples; otherwise, they are negative samples.

4.3. Overall Tracking Framework

As shown in Figure 4, the overall tracking framework of this paper comprises a judgment mechanism and a Siamese network stage. When the response score value of correlation filtering does not meet the result requirements, it enters the Siamese network stage, and obtains the extreme target boxes in different regions according to the multi-peak response. Starting from these target boxes, the correlation degree was calculated in the deep network, and the target box with the highest correlation was the final response. When the response score reaches the set threshold, the correlation filter tracking is continued. The specific steps are as follows:

Step 1: Firstly, the Groundtruth is obtained, and its 2.5 times target area is input into the KCF algorithm as the search area to obtain its combination feature $xx = \{x_H, x_C\}$, and calculate the response value in the filter.

Step 2: Discriminate using the maximum response value and threshold input into the judgment mechanism to select the next step to be taken.

Step 3: When the judgment is true, continue to use the feature fusion algorithm based on KCF algorithm for tracking.

Step 4: When the judgment is false, it indicates that the tracking effect of KCF in the current frame is poor. The input template of KCF and the multi-peak target box are used as the network input to the Siamese neural network to extract the universal features and discriminative features of the search area and the target template. The feature map is upsampled from 17 × 17 × 17 to 272 × 272 × 272 by bicubic interpolation (because the original image is relatively rough, this can obtain more accurate positioning), and the scale function of the Siamese network (four templates) is used to obtain the new tracking position. The final result is calculated as follows.

$$f(x) = \begin{cases} \hat{k}^{xz} \cdot \hat{a}, \;\; \max\_R > \sigma \\ \min\_Loss = corr(template_x, template\_z), else' \end{cases} \tag{15}$$

Step 5: The results obtained by the tracker are fed back to the filter to update the coefficients and the template of the next frame, in order to obtain a more robust tracking algorithm.

5. Experiments

5.1. Environment and Dataset

5.1.1. Environment

This algorithm mainly uses Ubuntu-18.0 system as the experimental basic environment, establishes Pytorch1.6 deep learning framework, and uses Python scripting language for programming implementation. At the same time, development modules such as cuda 10.0, cudnn 7.5, python 3.7, pytorch 1.0.0, and python-opencv are used. The software and hardware parameters of the computer are as follows: an eight-core Core·i7-7700 CPU with a main frequency of 3.60 GHz, Geforce·Ground_TruthX-1080·GPU with 32 GB memory. In terms of training optimization, the network was trained iteratively for 50 epochs, the batch size of training data was set to 8, and the learning rate was set to 10^{-2}–10^{-8}.

In the multi-peak guided, anti-similar object tracking experiment based on the judgment mechanism, the following parameters are adjusted:

(1) Module selection for Attention Mechanism: In this paper, Channel Attention is used to discriminate the target position in different channels of features, in order to improve the saliency of the target region and reduce the importance of non-target regions.
(2) A basic temporal constraint is incorporated, limiting the object search to a time range of approximately four times the previous size. Additionally, a cosine window is added to the score map to penalize large temporal offsets. To track objects in a large-scale space, the search pattern is processed in several scaled versions. Penalization is applied to changes in all scales, while changes in the current scale are suppressed.
(3) Stride setting: The method with one as the quantization stride does not have an impact on the image containing the object information, but on the network, the score map will be reduced by a multiple after passing through the network. It can be known from the network structure that the convolution and pooling with three layers take two as the quantization step.

5.1.2. Dataset and Evaluation Metrics

In this paper, the OTB-50 [44] dataset is used to test the experimental results. The OTB dataset contains the most common video sequences in our daily life. This dataset provides a rich set of challenges including object scale changes, occlusions, fast motion, illumination changes, and more. Each video sequence is provided with an initial bounding box and complete annotation of the object, as well as an evaluation metric used to evaluate the performance of the tracker. The following metrics are used to evaluate performances of tracking methods:

(1) Precision is an indicator that measures the overlap between the predicted bounding box and the Groundtruth bounding box of the tracker at a given frame. It represents the accuracy of the tracker on the target position.
(2) Success rate is an indicator that measures the proportion of successful tracking of targets by a tracker across the entire dataset. It indicates whether the tracker's tracking results on different frames were successful. The success rate is usually calculated by calculating the ratio of the number of frames successfully tracked to the total number of frames.

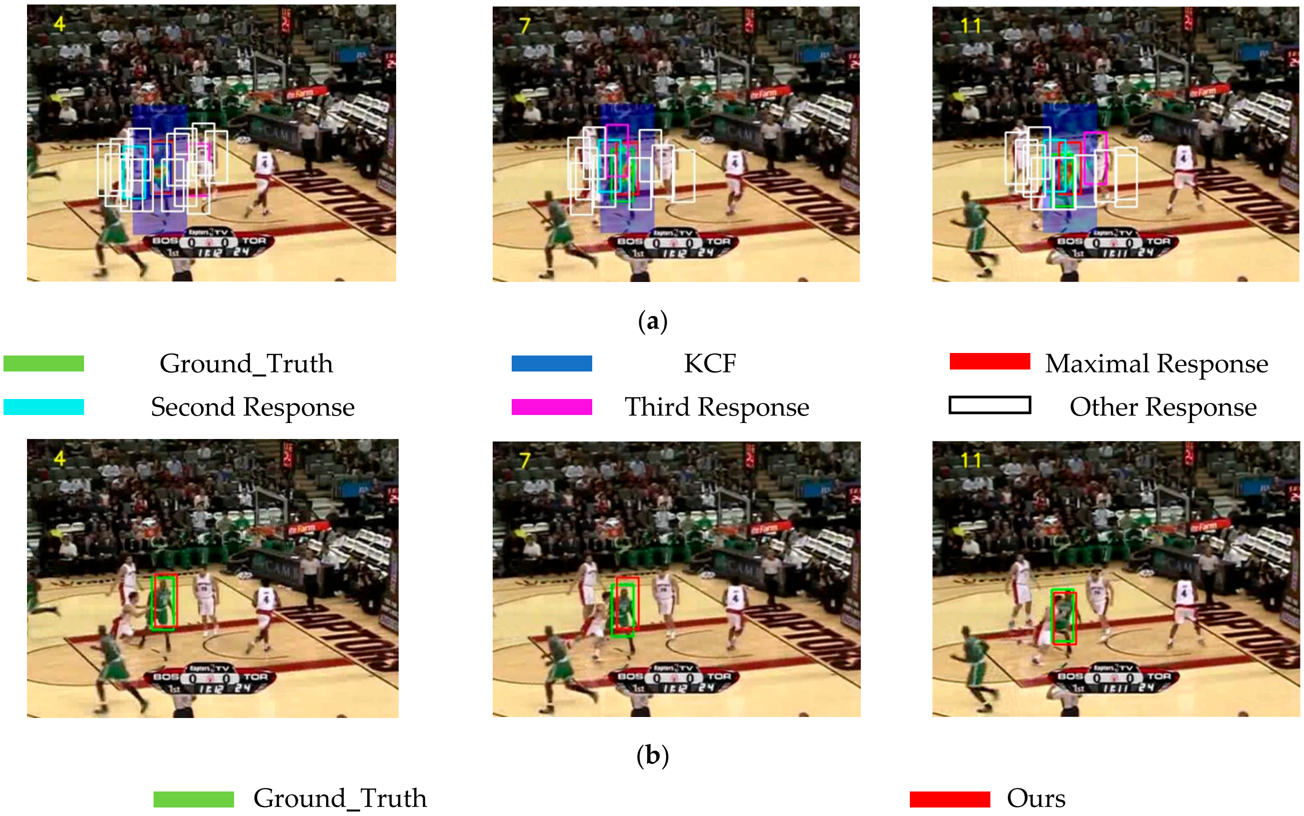
5.2. Analysis of Experimental Results

In order to prove whether the proposed method is effective in object tracking, the OTB-50 dataset is selected for verification, and the video sequences in the dataset cover 11 different tracking challenge attributes. We selected some video sequences from the above dataset to demonstrate the experimental results, and selected subjective and objective evaluation metrics in this section to analyze our algorithm.

5.2.1. Qualitative Analysis

Figure 8 shows some frames from the Basketball video sequence, comparing the multi-peak object bounding box of the KCF feature fusion algorithm with the subjective effect tested by our proposed algorithm. In the previous summary of the tracking results of the KCF algorithm in this paper, the Basketball video sequence is mainly about the target moving continuously, which is affected by the appearance of similar objects (similar athletes) around and partially occluded by similar moving objects during the process of movement. During the tracking process, the tracker is affected by the appearance of similar objects, and when the similar object is too close to the target, it may even occlude the original target in the search area, causing the filter coefficients to change and the tracker to easily fail. Figure 8a shows the multi-peak tracking results after the original KCF algorithm is improved by feature fusion. Figure 8b shows the tracking result of the Siamese network solution algorithm after extracting the depth features of the multi-template object bounding box in (a). It can be seen from the figure that the target box moves with the movement of the original tracking target, and is not affected by similar objects, and is very close to the

Groundtruth. Even when the similar object partially occludes the target, the original target can be distinguished well, maintaining the robustness of the tracker.



(**a**)

(**b**)

Figure 8. Multi-Peak Object bounding Box and Tracking Results of the Proposed Algorithm on Basketball. (**a**) Multi-peak target box; (**b**) Algorithm in this paper.

Figures 9–13 are partial video displays of the comparison of tracking results of KCF algorithm, SiamFC algorithm and the proposed algorithm on OTB dataset, respectively. It can be seen that in Figure 9, the size of the target is always changing, and the background is an object with a very close texture to the target. Under the interference of continuous motion and similar objects, the proposed algorithm can continuously track the target position, while SiamFC and KCF drift. In Figure 10, the effective images of the target appear less due to the perspective change, and it is easy to confuse with the surrounding pedestrians. The proposed algorithm can maintain stable positioning and expand the prediction range as much as possible under the condition of limited target observability, so that the target is always included in the prediction box and the large drift phenomenon is avoided. In Figure 11, the environment around the target changes greatly, and the existence of surrounding objects has great interference on the prediction of the target. It can be seen that the proposed algorithm can quickly redetect the correct target in the case of error, while SiamFC and KCF track the wrong target for a long time, resulting in a large deviation. In Figure 12, there are multiple similar objects around the target, and the target is in a state of continuous activity and is occluded by multiple similar objects. The prediction range of the proposed algorithm is the closest to the Groundtruth, and the prediction result of SiamFC is also guaranteed to be a complete target object, but there is still a small range of offset in the prediction result of KCF. It can be seen in Figure 13 that the target is always moving rapidly, and there are similar objects that are extremely close to the target around. The proposed algorithm has good robustness, and can achieve accurate positioning even if the distance is extremely close to the same type of object. Compared with SiamFC, which mistakenly identifies similar objects as the target to be tracked, the KCF algorithm still shows the phenomenon of prediction box drift. Compared with KCF and SiamFC, the proposed algorithm can keep a good tracking effect in the environment with similar objects. The tracking results are more accurate and the anti-similar object ability is stronger.

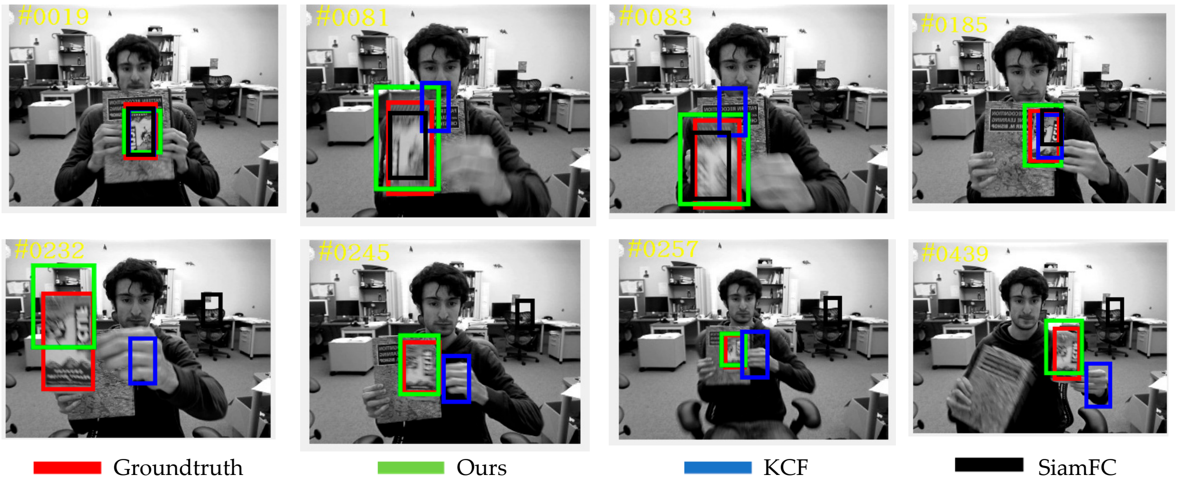


Figure 9. Subjective Results of Different Algorithms on the Coke Sequence.

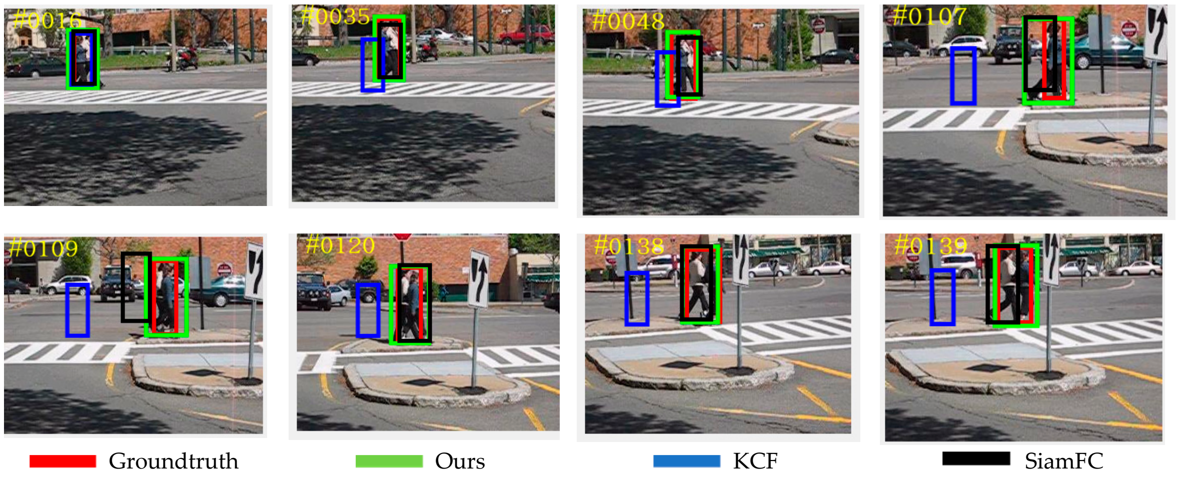


Figure 10. Subjective Results of Different Algorithms on the Couple Sequence.

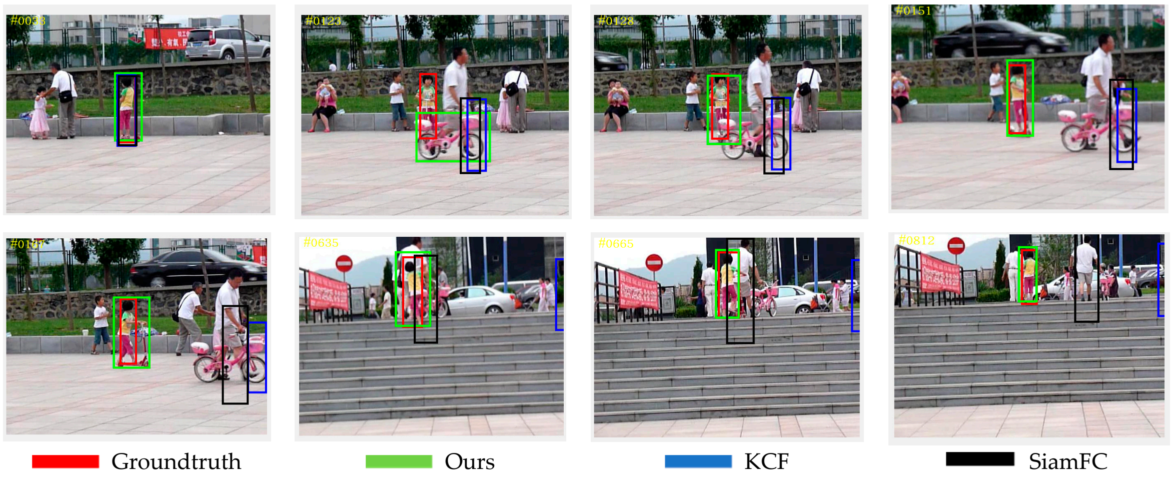


Figure 11. Subjective Results of Different Algorithms on the Girl2 Sequence.

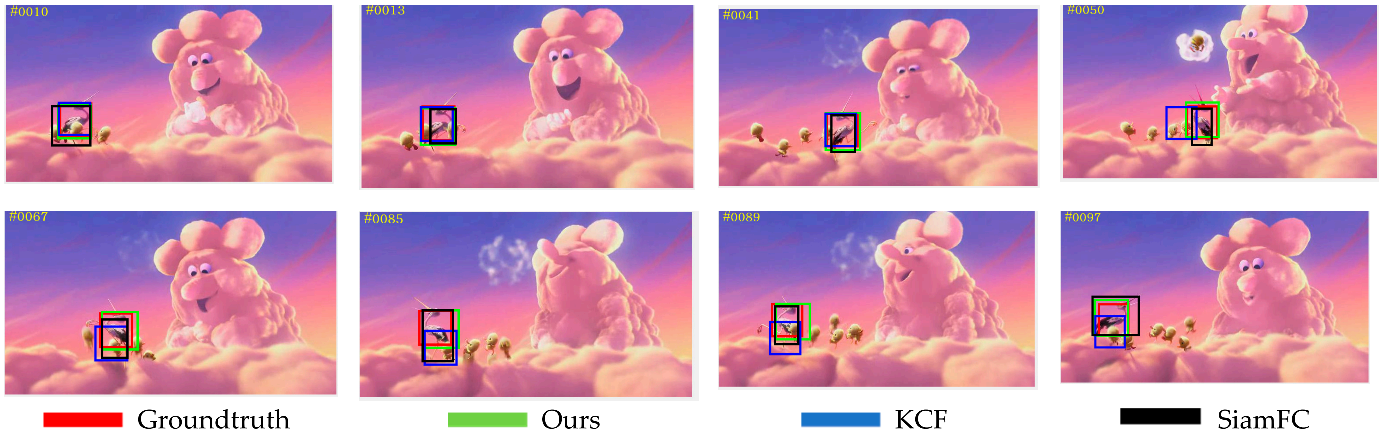


Figure 12. Subjective Results of Different Algorithms on the Bird2 Sequence.

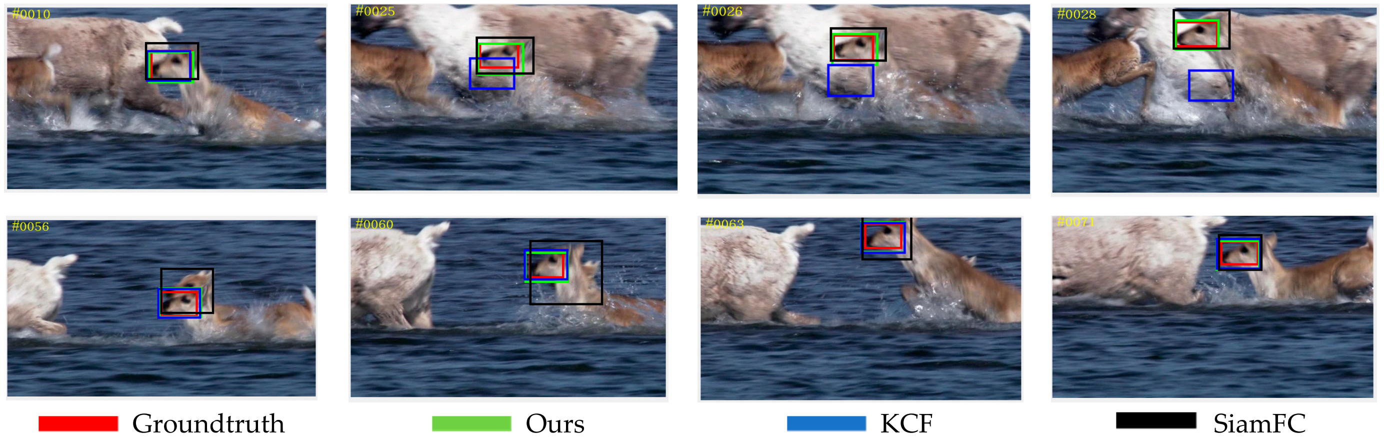


Figure 13. Subjective Results of Different Algorithms on the Deer Sequence.

5.2.2. Quantitative Analysis

Apart from subjective evaluation, this article employs two objective evaluation metrics to analyze the performance of object tracking algorithms. The tracking experiments were conducted on the OTB-50 dataset, and comparative experiments were carried out on video sequences containing similar objects.

Table 2 shows the results of tracking tests on nine videos selected from the OTB dataset using the KCF tracking algorithm, SiamFC tracking algorithm, and the proposed algorithm. According to Table 2, the proposed algorithm achieves an average precision improvement of 29.7% and an average success rate improvement of 29% compared to the KCF algorithm. Similarly, the proposed algorithm outperforms the SiamFC algorithm with an average precision improvement of 16.4% and an average success rate improvement of 16.7% on the same video sequences with similar objects interferences. Therefore, the proposed algorithm demonstrates better tracking and recognition capabilities, as well as stability, compared to the KCF and SiamFC tracking algorithms. It effectively addresses the issue of failures caused by the presence of similar targets in object tracking.

Table 3 shows the average precision and success rate results of the proposed algorithm compared to KCF, SiamFC, DeepSRDCF, and MDNet tracking algorithms on the OTB-50 dataset. Compared to KCF, the proposed algorithm improved precision by 25.9%, and compared to DeepSRDCF, SiamFC, MDNet, SiamR-CNN, and SiamGAT, it improved precision by 10.2%, 3.2%, 2.8%, 1.1%, and 0.2%, respectively. In terms of success rate, the proposed algorithm improved by 23.3% compared to KCF, and by 7.3%, 6.8%, 0.5%, and 0.3% compared to DeepSRDCF, SiamFC, MDNet, and SiamR-CNN, respectively. However, the success rate of the proposed algorithm was lower than that of SiamGAT.

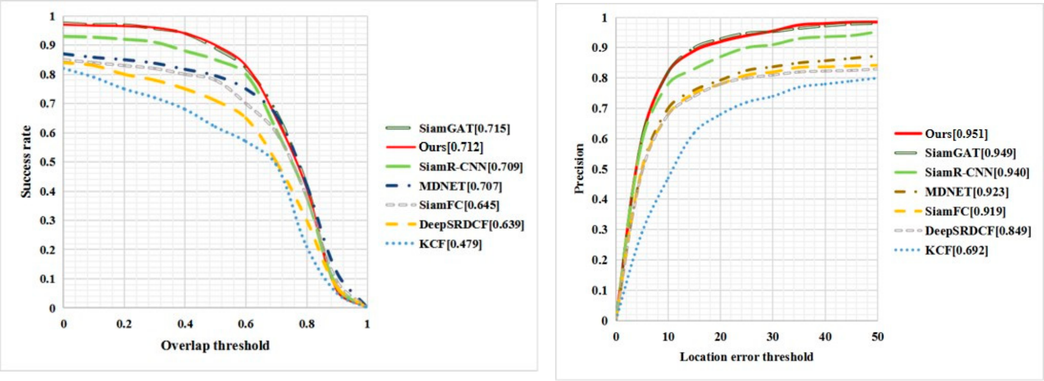
Table 2. Comparison of Tracking Results between KCF Algorithm, SiamFC Algorithm, and the Method Proposed in this Article.

Video Sequence	KCF		SiamFC		Ours	
	Precision (%)	Success (%)	Precision (%)	Success (%)	Precision (%)	Success (%)
Basketball	48.0	32.7	33.6	24.7	86.3	70.9
CarDark	90.1	73.5	84.6	68.5	93.5	75.1
BlurCar1	89.9	64.0	76.1	71.0	90.1	88.9
Deer	29.9	26.3	56.4	49.0	81.1	67.1
Soccer	13.7	14.4	13.1	11.9	24.2	20.8
Bird2	57.9	49.8	84.3	72.8	87.3	75.3
Coke	69.2	55.2	77.3	59.2	87.4	79.4
Couple	31.0	27.7	89.5	68.4	94.5	90.7
Girl2	6.18	9.1	40.4	37.7	58.7	45.2
Average	48.4	39.2	61.7	51.5	78.1	68.2

Table 3. Precision and Success Rate Results of Various Algorithms on OTB-50.

Algorithm Name	Precision (%)	Success (%)
KCF [7]	69.2	47.9
DeepSRDCF [8]	84.9	63.9
SiamFC [9]	91.9	64.5
MDNet [45]	92.3	70.7
SiamR-CNN [23]	94.0	70.9
SiamGAT [46]	94.9	71.5
Ours	95.1	71.2

Plotting the data as a curve can better illustrate the comparison of algorithm data. Figure 14 shows the comparison curve of the proposed algorithm and several classic algorithms in the table above. Figure 14a shows the success rate curve at a given center position error threshold, and Figure 14b shows the precision curve at a given center position error threshold. It can be seen that the area surrounded by the proposed algorithm curve and the coordinate axis is the largest among all the algorithms, indicating that the proposed algorithm has significantly improved tracking performance compared to KCF and SiamFC algorithms. Moreover, compared to classic algorithms in the correlation filter, deep learning, and combined deep learning and correlation filter algorithms, the proposed algorithm has outstanding performance and high-practical significance.



(**a**) (**b**)

Figure 14. Tracking results on OTB-50. (**a**) Success; (**b**) Precision.

6. Conclusions

The algorithm proposed in this paper is characterized by the design of a judgement mechanism that combines the related filtering algorithm and Siamese network. In the related filtering algorithm, a multi-peak guidance method is used to screen out the maximum response target box in each region. In the Siamese network, Channel Attention is introduced to enhance feature fusion processing, enabling the model to adaptively focus on useful information at different scales and improve its discriminative ability in special scenarios. Especially in the presence of similar object interference, the accuracy of candidate region prediction is improved by screening multiple templates based on correlation filtering algorithm, and the combination of universal features and discriminative features extracted by the Siamese network can better distinguish foreground and background information. This allows the model to achieve stable prediction processing in real-time tracking, even in the presence of similar background interference. After comparison in the OTB dataset, the proposed algorithm in this paper achieved a 29.7% increase in precision compared to the KCF algorithm, and a 3.2% increase compared to SiamFC. The success rate value also increased by 23.3% compared to KCF and 6.8% compared to SiamFC. In terms of similar object interference, the proposed improvement scheme in this paper achieved an accuracy and success rate improvement of over 20% based on the original results of KCF and SiamFC. This indicates that the proposed algorithm in this paper has better tracking and recognition capabilities, as well as stability.

Prospective direction: (1) Global information: This paper seeks the local area where the target may exist according to the peak influence, and there are still errors in the inference of its information. Perhaps introducing spatial attention in the whole space and performing similarity comparison between global information can obtain a more comprehensive difference between the target value and the background value. (2) Temporal information: In this paper, only the initial frame is used as the tracking basis, and large differences have been generated after the target has been transformed for a long time. If the target information in the interval frame can be introduced, the appearance deformation of the target can be well adapted.

Author Contributions: Conceptualization, J.W.; methodology, J.W.; software, Y.W.; validation, J.W. and Y.W.; formal analysis, J.W.; investigation, X.W. and W.H.; resources, J.W. and Y.W.; data curation, W.H.; writing—original draft preparation, Y.W.; writing—review and editing, J.W.; visualization, J.W. and Y.W.; supervision, J.W., X.W., W.H. and L.Y.; project administration, J.W., X.W. and L.Y.; funding acquisition, W.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the National Natural Science Foundation (NNSF) of China NO. 62073258), the Scientific Research Program funded by the Shaanxi Education Department (NO. 21JK0787).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wang, J.; Zhu, H.J.I.J.W.M.I.P. Object tracking via dual fuzzy low-rank approximation. *Int. J. Wavelets Multiresolution Inf. Process.* **2019**, *17*, 1940003. [CrossRef]
2. Wang, W.; Xu, H.; Alazab, M.; Gadekallu, T.R.; Han, Z.; Su, C. Blockchain-Based Reliable and Efficient Certificateless Signature for IIoT Devices. *IEEE Trans. Ind. Inform.* **2022**, *18*, 7059–7067. [CrossRef]
3. Ren, S.; Chen, S.; Zhang, W. Collaborative Perception for Autonomous Driving: Current Status and Future Trend. In Proceedings of the 2021 5th Chinese Conference on Swarm Intelligence and Cooperative Control, Singapore, 19–22 November 2021; pp. 682–692.
4. Xia, R.; Chen, Y.; Ren, B. Improved anti-occlusion object tracking algorithm using Unscented Rauch-Tung-Striebel smoother and kernel correlation filter. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 6008–6018. [CrossRef]
5. Wang, J.; Zhu, H.; Yu, S.; Fan, C. Object tracking using color-feature guided network generalization and tailored feature fusion. *Neurocomputing* **2017**, *238*, 387–398. [CrossRef]

6. Bolme, D.S.; Beveridge, J.R.; Draper, B.A.; Lui, Y.M. Visual object tracking using adaptive correlation filters. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 2544–2550.
7. Henriques, J.F.; Caseiro, R.; Martins, P.; Batista, J. High-Speed Tracking with Kernelized Correlation Filters. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 583–596. [CrossRef] [PubMed]
8. Danelljan, M.; Häger, G.; Khan, F.S.; Felsberg, M. Learning Spatially Regularized Correlation Filters for Visual Tracking. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 4310–4318.
9. Danelljan, M.; Bhat, G.; Khan, F.S.; Felsberg, M. ECO: Efficient Convolution Operators for Tracking. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6931–6939.
10. Liu, P.; Du, J.; He, S.; Ren, G. A Real-Time Target Tracking Algorithm Based on Improved Kernel Correlation Filter. In Proceedings of the 2021 5th International Conference on Imaging, Signal Processing and Communications (ICISPC), Kumamoto, Japan, 23–25 July 2021; pp. 5–9.
11. Du, S.; Wang, S. An Overview of Correlation-Filter-Based Object Tracking. *IEEE Trans. Comput. Soc. Syst.* **2022**, *9*, 18–31. [CrossRef]
12. Hou, W.; Li, H.; Su, J.; Cui, H.; Luo, X. Target tracking algorithm based on image matching and improved kernel correlation filter. In Proceedings of the 2021 International Conference on Electronic Information Engineering and Computer Science (EIECS), Changchun, China, 23–25 September 2021; pp. 252–257.
13. Hou, Y.; Lin, X.; Li, J. Correlation Filter and Deep Siamese Network Hybrid Algorithm for Visual Object Tracking. In Proceedings of the 2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP), Xi'an, China, 9–11 April 2021; pp. 73–76.
14. Zhao, F.; Hui, K.; Wang, T.; Zhang, Z.; Chen, Y. A KCF-Based Incremental Target Tracking Method with Constant Update Speed. *IEEE Access* **2021**, *9*, 73544–73560. [CrossRef]
15. Ondrašovič, M.; Tarábek, P. Siamese Visual Object Tracking: A Survey. *IEEE Access* **2021**, *9*, 110149–110172. [CrossRef]
16. Cen, M.; Jung, C. Fully Convolutional Siamese Fusion Networks for Object Tracking. In Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 7–10 October 2018; pp. 3718–3722.
17. Li, B.; Yan, J.; Wu, W.; Zhu, Z.; Hu, X. High Performance Visual Tracking with Siamese Region Proposal Network. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8971–8980.
18. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
19. Zhu, Z.; Wang, Q.; Li, B.; Wu, W.; Yan, J.; Hu, W. Distractor-Aware Siamese Networks for Visual Object Tracking. In Proceedings of the Computer Vision—ECCV 2018, Munich, Germany, 8–14 September 2018; pp. 103–119.
20. Li, B.; Wu, W.; Wang, Q.; Zhang, F.; Xing, J.; Yan, J. SiamRPN++: Evolution of Siamese Visual Tracking with Very Deep Networks. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 4277–4286.
21. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
22. Wang, Q.; Zhang, L.; Bertinetto, L.; Hu, W.; Torr, P.H.S. Fast Online Object Tracking and Segmentation: A Unifying Approach. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 1328–1338.
23. Voigtlaender, P.; Luiten, J.; Torr, P.H.S.; Leibe, B. Siam R-CNN: Visual Tracking by Re-Detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 6577–6587.
24. Sosnovik, I.; Moskalev, A.; Smeulders, A. Scale Equivariance Improves Siamese Tracking. In Proceedings of the 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), Virtual, 5–9 January 2021; pp. 2764–2773.
25. Luo, Y.; Cai, Y.; Wang, B.; Wang, J.; Wang, Y. SiamFF: Visual Tracking with a Siamese Network Combining Information Fusion with Rectangular Window Filtering. *IEEE Access* **2020**, *8*, 119899–119910. [CrossRef]
26. Bahdanau, D.; Cho, K.; Bengio, Y.J.C. Neural Machine Translation by Jointly Learning to Align and Translate. Neural machine translation by jointly learning to align and translate. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015—Conference Track Proceedings, San Diego, CA, USA, 7–9 May 2015.
27. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the Computer Vision—ECCV 2018, Munich, Germany, 8–14 September 2018; pp. 3–19.
28. Bian, T.; Hua, Y.; Song, T.; Xue, Z.; Ma, R.; Robertson, N.; Guan, H. VTT: Long-term Visual Tracking with Transformers. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 9585–9592.
29. Yang, C.; Zhang, X.; Song, Z. CTT: CNN Meets Transformer for Tracking. *Sensors* **2022**, *22*, 3210. [CrossRef]
30. Xing, D.; Evangeliou, N.; Tsoukalas, A.; Tzes, A. Siamese Transformer Pyramid Networks for Real-Time UAV Tracking. In Proceedings of the 2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 4–8 January 2022; pp. 1898–1907.

31. Yu, Y.; Xiong, Y.; Huang, W.; Scott, M.R. Deformable Siamese Attention Networks for Visual Object Tracking. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 6727–6736.
32. Chen, X.; Yan, B.; Zhu, J.; Wang, D.; Yang, X.; Lu, H. Transformer Tracking. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Virtual, Online, 19–25 June 2021; pp. 8122–8131.
33. Yu, B.; Tang, M.; Zheng, L.; Zhu, G.; Wang, J.; Feng, H.; Feng, X.; Lu, H. High-Performance Discriminative Tracking with Transformers. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021; pp. 9836–9845.
34. Gu, F.; Lu, J.; Cai, C. RPformer: A Robust Parallel Transformer for Visual Tracking in Complex Scenes. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 5011214. [CrossRef]
35. Chen, X.; Kang, B.; Wang, D.; Li, D.; Lu, H. Efficient Visual Tracking via Hierarchical Cross-Attention Transformer. In Proceedings of the Computer Vision—ECCV 2022 Workshops, Tel Aviv, Israel, 23–27 October 2022; pp. 461–477.
36. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B.J.I.C.I.C.o.C.V. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021; pp. 9992–10002.
37. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *arXiv* **2017**, arXiv:1706.03762.
38. Wu, Y.; Lim, J.; Yang, M.H. Object Tracking Benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1834–1848. [CrossRef] [PubMed]
39. Felzenszwalb, P.F.; Girshick, R.B.; McAllester, D.; Ramanan, D. Object Detection with Discriminatively Trained Part-Based Models. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 1627–1645. [CrossRef] [PubMed]
40. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; Volume 881, pp. 886–893.
41. Kristan, M.; Matas, J.; Leonardis, A.; Felsberg, M.; Cehovin, L.; Fernandez, G.; Vojir, T.; Hager, G.; Nebehay, G.; Pflugfelder, R.; et al. The Visual Object Tracking VOT2015 Challenge Results. In Proceedings of the 2015 IEEE International Conference on Computer Vision Workshop (ICCVW), Santiago, Chile, 11–18 December 2015; pp. 564–586.
42. Glorot, X.; Bordes, A.; Bengio, Y. Deep Sparse Rectifier Neural Networks. In Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, Proceedings of Machine Learning Research (AISTATS), Lauderdale, FL, USA, 11–13 April 2011; pp. 315–323.
43. Bertinetto, L.; Valmadre, J.; Henriques, J.; Vedaldi, A.; Torr, P. Fully-Convolutional Siamese Networks for Object Tracking. In Proceedings of the European Conference on Computer Vision(ECCV), Amsterdam, The Netherlands, 11–14 October 2016; pp. 850–865.
44. Wu, Y.; Lim, J.; Yang, M.H. Online Object Tracking: A Benchmark. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2411–2418.
45. Nam, H.; Han, B. Learning Multi-domain Convolutional Neural Networks for Visual Tracking. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 4293–4302.
46. Guo, D.; Shao, Y.; Cui, Y.; Wang, Z.; Zhang, L.; Shen, C. Graph Attention Tracking. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Virtual, 19–25 June 2021; pp. 9538–9547.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.