symmetry

MDPI

Article

Multi-Type Object Tracking Based on Residual Neural Network Model

Tao Jiang [1,2], Qiuyan Zhang [3,4], Jianying Yuan [2], Changyou Wang [3,*] and Chen Li [5]

[1] School of Intelligent Medicine, Chengdu University of Traditional Chinese Medicine (CDUTCM), Chengdu 610075, China
[2] International Joint Institute of Robotics and Intelligent Systems, Chengdu University of Information Technology, Chengdu 610225, China
[3] College of Applied Mathematics, Chengdu University of Information Technology, Chengdu 610225, China
[4] School of Mathematical Science, University of Electronic Science and Technology of China, Chengdu 611731, China
[5] College of Medicine and Biological Information Engineering, Northeastern University, Shenyang 110819, China
* Correspondence: wangchangyou417@163.com

Abstract: In this paper, a tracking algorithm based on the residual neural network model and machine learning is proposed. Compared with the widely used VGG network, the residual neural network has deeper characteristic layers and special additional layer structure, which break the symmetry of the network and reduce the degradation of the neural network. The additional layer and convolution layer are used for feature fusion to represent the target. The multi-features of the object can be captured by using the developed algorithm, so that the accuracy of tracking can be improved in some complex scenarios. In addition, we defined a new measure to calculate the similarity of different image regions and find the optimal matched region. The search area is delimited according to the continuity of the target motion, which improves the real-time performance of tracking. The experimental results illustrate that the proposed algorithm achieved a higher accuracy while taking into account the real time performance, especially in dealing with some complex scenarios such as deformation, rotation changes and background clutters, in comparison with the Multi-Domain Network (MDNet) algorithm based on a convolutional neural network.

Keywords: machine learning; deep neural network; object tracking; residual network

Citation: Jiang, T.; Zhang, Q.; Yuan, J.; Wang, C.; Li, C. Multi-Type Object Tracking Based on Residual Neural Network Model. *Symmetry* **2022**, *14*, 1689. https://doi.org/10.3390/sym14081689

Academic Editor: João Ruivo Paulo

Received: 29 June 2022
Accepted: 10 August 2022
Published: 15 August 2022

1. Introduction

Object tracking is one of the fundamental problems in computer vision. It has been widely used in video monitoring [1], intelligent transportation [2], intelligent medical diagnosis [3], human-machine interaction [4] and other aspects. However, there are many challenges in some complex scenarios such as deformation (DEF), scale changes (SCs), rotation changes (RCs), background clutters (BCs) and occlusion (OCC). Many algorithms have been proposed for different application backgrounds. The existing algorithms mainly include the following categories: object tracking algorithms based on mean shift, subspace learning, detection, sparse representation, correlation filtering and deep neural network.

In mean shift algorithm [5], the kernel RGB histogram is selected as the image feature, and the similarity between different image regions is measured by calculating the Pasteur coefficient. The highest of the similarity is searched in the current frame, basing on the spatial position of the target in the previous frame. The algorithm is essentially a local optimization algorithm based on gradient rise. It has higher real-time performance, which is widely used in pattern recognition, digital image processing and computer vision. However, it has limitations in solving the problems of occlusion, background clutter and scale change. In subspace machine learning algorithm [6], the features are mapped from high-dimensional

space to low-dimensional space by a set of linear projections. It provides a compact image representation, but a large number of target templates from different perspectives and lighting conditions need to be collected in advance. Therefore, the practicability of the algorithm is not very satisfactory. Detection algorithm [7] is a discriminant algorithm. In this method, the discriminant features are selected by maximizing the variance ratio of the feature distribution between the tracked target and the surrounding background, and the robust tracking of the target is realized in the mean shift framework. Recently, the algorithm has made some progress, but it still needs to be advanced in solving the occlusion problem and online update. In the sparse representation algorithm [8], the target to be tracked is regarded as a dictionary sparse representation composed of a target template and trivial template (unit matrix). The target template is used to capture the changes of the target in the tracking process, and the trivial template is used to describe the interferences such as occlusion and illumination changes. This kind of algorithm has higher accuracy, but lower real-time performance because of solving complex convex optimization problems. The correlation filtering algorithm [9] constructs an adaptive correlation filter to model the appearance of the target. By calculating the minimum square error between the actual correlation output and the expected correlation output, the target can be detected and tracked. The algorithm not only shows good robustness to the challenges of illumination change, affine change and non-rigid deformation, but also has a good real-time performance. The main improvement direction is to add more robust features, design a more effective model updating mechanism and speed up the calculation efficiency.

The deep neural network (DNN) algorithm is first applied to the field of object tracking by Wang et al. in [10]. They combine off-line training with on-line adjustment and propose the deep learning tracking (DLT). Firstly, it uses the stacked denoising autoencoder to carry out off-line pre training on large-scale natural image data set to learn the general object representation method. Then, it uses the positive and negative samples to carry out on-line fine-tuning to make the depth network have the ability to present the characteristics of the currently tracked objects. However, the image resolution of auxiliary data set used in offline pre training stage is too low, which makes it difficult for the encoder to learn effective feature representation. Moreover, the fully connected network is not good enough to describe the target features. Therefore, the tracking effect of DLT algorithm still lower than that of traditional tracking algorithm. To make up for this deficiency and enhance the ability to characteristic representation, many researchers use the convolutional neural network (CNN) as a network model for feature extraction and classification. The CNN is pre-trained by using the ImageNet and other large-scale image datasets to obtain the efficient characteristic such as the Visual Geometry Group (VGG)-Net. The deeper the network layer is, the more accurate and comprehensive the extracted features are. However, the deepening of network layers will lead to a surge in the amount of computing, which will consume more resources. It creates an irreconcilable contradiction.

For some complex scenarios, it is very important to enhance the ability of feature representation. To solve the problem, we propose a new algorithm based on a new deep neural network model and machine learning. The main contribution of this work can be summarized as follows:

(1) Based on the advantages of residual network and machine learning, a new image feature extraction algorithm is proposed. In the algorithm, only two layers are extracted. One is the low-level feature, and the other is the high-level feature. It reduces the complexity of calculation caused by the increase of parameters. This allows for a trade-off between effectiveness and accuracy in some complex scenes.
(2) A new measure is defined to calculate the similarity of different image regions. This new metric skillfully transforms multiplication into addition, which greatly improves the operation speed. At the same time, it also integrates the advantages of QATM algorithm [11], taking into account the uniqueness of pairing, rather than simply evaluating the matching score.

(3) In the search algorithm, the position of the target in the previous frame is taken as the core, which appropriately reduces the search range and improves the real-time tracking to a certain extent.

In the following, we first review related work in Section 2, and present the proposed algorithm based on deep neural network and machine learning in Section 3. The experimental results in public tracking benchmark are demonstrated in Section 4. Finally, the conclusions are given in Section 5.

2. Related Work

The existing tracking algorithms can be categorized into generative and discriminative methods [12]. The generative methods describe the appearance characteristics of the target and minimize the reconstructed errors by searching the candidate target. The generative methods simply focus on the target and ignore the background information. The tracking target is easily lost if the appearance changes drastically. Recently, benchmark evaluations [13,14] suggest that using background information has noticeable effects on the object tracking results and conclude that discriminative methods provide better performance than the generative ones. The representative discriminative algorithms include boosting [15], multiple instance learning [16], tracking learning-detection (TLD) [17], structured support vector machine [18] and so on.

In recent years, the deep neural network (DNN) has been developing continuously, which has made great success in many fields such as target detection, natural language processing and unmanned driving. DNN has strong learning and expression abilities so that there are more and more tracking algorithms based on DNN [19–24]. Wang et al. [10] first apply DNN to the object tracking and propose a deep learning tracking (DLT) algorithm. The algorithm combines off-line training with on-line adjustment. To improve the tracking effect of DLT algorithm, Wang et al. [25] propose an improved depth network tracking algorithm, in which the convolutional neural network (CNN) is used as the network model to obtain features and classifications. Then, Wang et al. [26] proposed target tracking algorithm based on the full convolution network and machine learning. This algorithm uses the CNN model to extract the target features and makes a detailed attribute analysis of the features obtained by different convolution layers. Later, Gan et al. [27] first applied the recurrent neural network (RNN) to object tracking and proposed a deep machine learning tracking algorithm based on CNN and RNN. Cui et al. [28] used the multi-directional recurrent neural network to model the spatial structure relationship in the tracking area to obtain the confidence graph, which is introduced into the correlation filter as a regularization term. Ondraska et al. [29] proposed the end-to-end target tracking algorithm based on RNN. Nam et al. [30] proposed a novel target tracking algorithm based on multi domain network (MDnet). This algorithm uses labeled tracking video as pre-train CNN model directly, divides the deep network into a shared layer and the domain-specific layer. The shared layer obtains the general feature expression of different types of targets, and the specific layer learns the specific feature expression of tracking targets. Their experimental results illustrate the outstanding performance compared with seven state-of-the-art trackers on the challenging video sequences. Their MDNet algorithm is ranked first place in accuracy and first or second place in robustness. However, from the data of Table 1 in [30], it can be seen that the average score of accuracy is less than 0.65, which is not high. Through a large number of experiments, it is found that the algorithm in reference [30] can not accurately track the following three types of targets in video sequences: the first is that the tracked target deforms or rotates; the second is that there is a very similar target interference next to the tracked target; the third is that the background of the tracked target is very complex. Qi et al. [31] proposed Hedged deep tracking (HDT). VGG16 is used to train the filter with different depth features, and the idea of adaptive ensemble learning is used to integrate multiple trackers into a more robust tracker. Daneljan et al. [32] proposed a new factorized convolution operator to reduce the model parameters, simplify the training set and improve the template updating strategy. Later, Goutam et al. [33] proposed unveiling the power of

deep tracking, which processes deep and shallow features, respectively, and then performs adaptive fusion. On the other hand, many researchers [34–36] applied the siamese network to the object tracking. Wang et al. [37] proposed a residual attentional siamese network for high performance object tracking. The algorithm mitigates the over-fitting problem in deep network training and performs independent representation learning and discriminant learning to enhance the discriminant ability and adaptive ability of the algorithm. Zhang et al. [38] propose new residual modules for ResNet to eliminate the negative impact of padding and enhance tracking robustness and accuracy. A tracking framework for end-to-end off-line training is proposed in [39], which is a completely traditional anchor-free Siam network. The deep network resnet-50 is used to extract rich features to improve the accuracy. Liu et al. [40] proposed an offline universal adversarial attack called Efficient Universal Shuffle Attack for visual object tracking. Zhou et al. [41] proposed a new Siamese central perceptual network for visual tracking, which consists of the subnetwork, followed by the classification, regression, and localization branches in parallel. The high accuracy and efficiency are achieved. Ondrašovič and Tarábek [42] gave a survey of Siamese visual object tracking. Gao et al. [43] proposed a new deformable sample generator. The classifier and the deformable sample generator learned jointly, which enhanced quantitative and qualitative evaluations for the visual object tracking task.

Inspired by the above research work, especially for the shortcomings of the algorithm in reference [30], a new target tracking algorithm based on the residual neural network model and machine learning is proposed in this paper. On the premise of ensuring the real-time tracking, our algorithm can accurately track three types of targets that can not be tracked by the algorithm of reference [30]. The purpose of our method is to improve the accuracy while taking into account the real time performance, especially in dealing with some complex scenarios. We use the deeper residual neural network (ResNet50) to extract deep features, which overcomes the gradient decreasing of convolution neural network with the deepening of layers. Fusing the special additional layer structure of ResNet50 and convolution layer feature, more robust target characterization features can be obtained. Object tracking can be regarded as a special template matching. The search region is determined by the position of current frame and motion continuity of tracking target. We define a measure assessing the quality of a matching pair using soft-ranking among all matching pairs and find the highest matching quality score.

3. Object Tracking Algorithm Based on Resnet-50

In the following, we first describe the overall framework of our algorithm. Next, we detail the mechanism of residual neural network. Finally, we describe the detail of measuring the similarity of different image regions.

3.1. Overview of the Developed Algorithm

The developed algorithm is based on residual neural network and machine learning. We use ResNet-50 to obtain the attribute features of the object. Object tracking can be regarded as a special template matching. Classic template matching methods often use sum-of-squared-differences or normalized cross correlation to calculate a similarity score between the template and the underlying image. These approaches work well when the transformation between the template and the target search image is simple. However, they fail in some complex scenarios, such as deformation, scale changes, rotation changes and background clutters, which is common in real life. Unlike those methods, we define a new measure in order to obtain the optimal matching regions. Based on the idea of quality aware, we give the criteria of tracking.

The general flow chart and the procedure of the developed algorithm are shown in Figure 1 and Algorithm 1. The main idea of the developed algorithm is to use the residual network (ResNet-50) to extract multiple features of the target, and then apply a new measure to match and find the optimal matching area. In order to avoid unnecessary search, the current frame of the object is used as the template frame, and the search area is

delimited according to the continuity of the target motion in the picture of the next frame. The search area is centered on the center position of the target frame of the current frame, and the size is 1.5 times more than the size of the template frame (if 1.5 times is beyond the scope of the whole picture of the next frame, search in the original picture).

Algorithm 1. Procedure of the developed algorithm for object tracking.

Input: initial object position P_1 and scale in the first frame.
Output: object position P_i and object scale in the i^{th} frame.
Draw the image patch I_1 of the object according to P_1 and scale in the first frame.
Set $i \leftarrow 2$ (initial frame number).
While ($i <$ the last frame number):
{

1. In the i^{th} frame, a box I_i^0 is intercepted with P_{i-1} as the center, and I_i^0 is 1.5 times the size of I_{i-1}.
2. Extract the features of I_i^0 and I_{i-1}:
   $T := F(I_{i-1}),: S := F(I_i^0)$, where $F(\bullet)$ is features extractor.
3. Compute the similarity measurement of T and S, and find the best matched region I_i in S.
4. Record the center position P_i of I_i.
5. Draw the image patch I_i of the objector according to P_i and scale in the i^{th} frame
6. Set $i \leftarrow i + 1$.

}

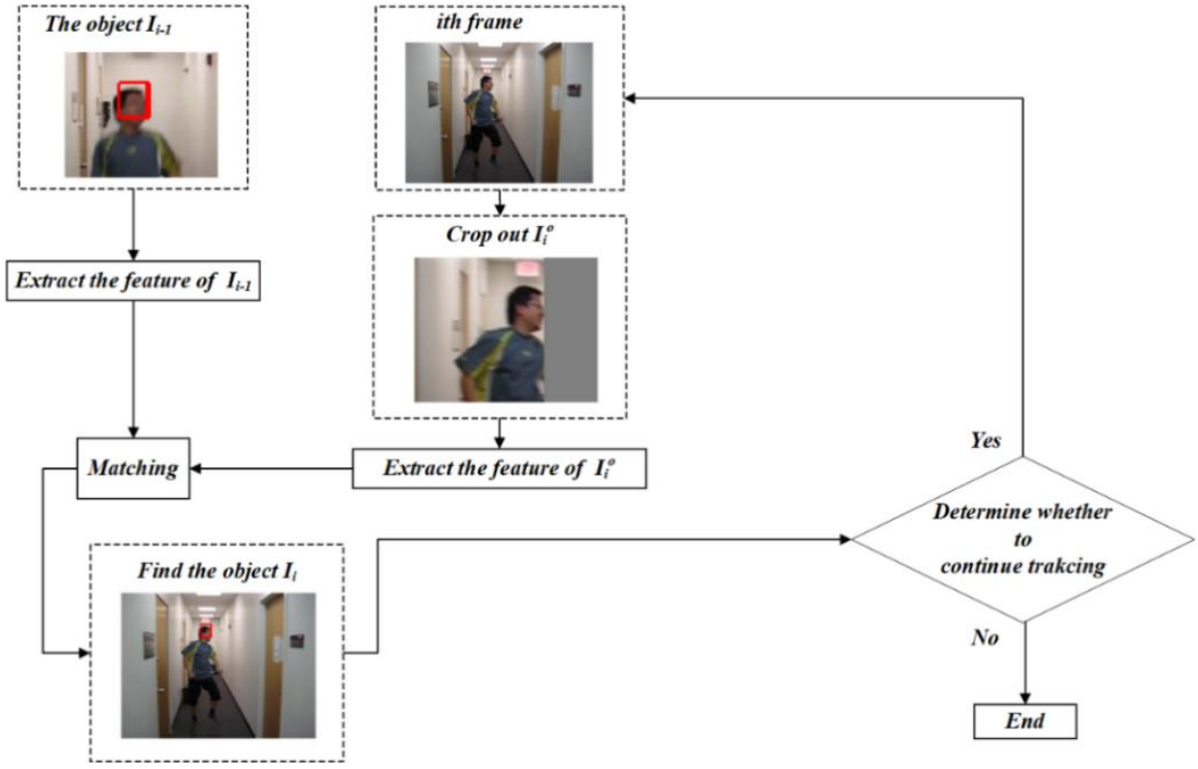

Figure 1. General flow chart.

3.2. Residual Neural Network Model

The traditional convolution network uses VGG to extract the features of the target. In theory, the expression ability of extracted features will be enhanced with the deepening of layers. However, the gradient of VGG network will disappear with the deepening of layers, resulting in the reduction of the accuracy of the extracted features. Skip connect in the residual network breaks the symmetry of the network. When the residual network is used to extract features, the problem of gradient decline caused by deep layers is completely avoided. ResNet50 is designed and trained for image classification tasks which almost pay

all attention to feature extraction. Low-level features contain rich texture information while high-level features normally reveal semanic clues. High computational complexity comes from deeper layers. As for object tracking, we have more operations than classification task after extracting features. Therefore, instead of extracting many layers in ResNet50, we extracted a low-level feature and high-level feature, respectively. Then, they are spliced to trade off effectiveness and complexity. Based on the above analysis, this paper selects the residual neural network (ResNet50) to extract target features and the network structure is shown in Table 1.

Table 1. Residual network framework for feature extraction.

Name	Patch Size/Stride	Output Size
Conv1	$7 \times 7/2$	$112 \times 112 \times 64$
Maxpool	$3 \times 3/2$	$56 \times 56 \times 64$
Residual2_x	$\begin{bmatrix} 1 \times 1/1 \\ 3 \times 3/2 \\ 1 \times 1/1 \end{bmatrix} \times 3$	$56 \times 56 \times 256$
Residual3_x	$\begin{bmatrix} 1 \times 1/1 \\ 3 \times 3/2 \\ 1 \times 1/1 \end{bmatrix} \times 4$	$28 \times 28 \times 512$
Residual4_x	$\begin{bmatrix} 1 \times 1/1 \\ 3 \times 3/2 \\ 1 \times 1/1 \end{bmatrix} \times 1$	$14 \times 14 \times 1024$
Combination	Residual3_x⊕Residual4_x	1536

From Table 1, we can see that the network structure of the developed algorithm is as follows: The Conv1 layer and Maxpool layer represent the first layer of convolution and the second layer of convolution, respectively. The Residual2_x layer represents three residual blocks with nine layers of convolution. The Residual3_x layer represents four residual blocks with twelve layers of convolution. The Residual4_x layer represents one residual block with three layers of convolution. The Combination layer splices the features output by Residual2_x layer and Residual4_x layer. Therefore, our network structure has a total of 26 layers, and the feature dimension of the image is 1536. These features are used to calculate the similarity and determine the location of the object.

3.3. Measuring Algorithm

To compute a similarity score between the template and the underlying image, traditional template matching algorithms generally utilize sum-of-squared-differences (SSD) or normalized cross correlation (NCC). When the transformation between the template and the target search picture is basic, these algorithms perform effectively. However, these approaches start to fail when it is in the real-world since the transformation is complicated or non-rigid. For complex situations, such as partial occlusion and color change, the weaknesses of these algorithms will be highlighted.

In [11], Cheng et al. proposed a novel quality-aware template matching (QATM) method, that can be easily embedded into any deep neural network. Five different template matching cases are considered as shown in Table 2, where "1-to-1 matching" implies exact matchings (two matched objects), "1-to-N" and "M-to-1" indicates s or t is a homogeneous patch causing multiple matches, e.g., white paper to white wall," M-to-N" means many patterned patches. It is clear that "1-to-1" matching case is the most important. A quantitative assessment of QATM as shown in Equation (1)

$$R^* = \underset{R}{\operatorname{argmax}} \left\{ \sum_{r \in R} \max\{Quality(r,t) | t \in T\} \right\} \quad (1)$$

such that the region R in search areas that maximizes the overall matching quality.

Table 2. Template matching cases from QATM.

	Matching Cases				**Not Matching**
	1-to1	**1-toN**	**M-to-1**	**M-to-N**	
Quality	High	Medium	Medium	Low	Very Low
QATM(s,t)	1	1/N	1/M	1/MN	$1/\|T\|\|S\| \approx 0$

Because the addition operation is faster than the multiplication operation, we propose a new measurement method, that is, to evaluate the matching quality between the object template and the search object, as shown below:

$$LNQATM\ (r,t) = \ln P(t|r) + \ln P(r|t) \tag{2}$$

where

$$P(t|r) = \frac{\exp\{\rho(f_t, f_r)\}}{\sum_{t \in T} \exp\{\rho(f_t, f_r)\}} \tag{3}$$

In (3), f_t and f_r are the feature representation of patch t and r, respectively. $\rho(\cdot,\cdot)$ is a predefined similarity measure between two patches, e.g., cosine similarity, which can be computed through the tensorflow.einsum in DNN. Therefore, in (2), $P(t|r)$ can be considered as the likelihood function that a template patch t is matched in S, and symmetrically, $P(r|t)$ can be considered as the likelihood function that a search patch r is matched in T. $P(t|r)$ can be interpreted as a soft-ranking of the current patch compared to all other patches in the template image in terms of matching quality. We can see that the maximum of the measurement function is related to both $\ln P(t|r)$ and $\ln P(r|t)$. The likelihood function $P(t|r)$ in (3) can be regarded as the softmax activation, which is a standard DNN layer. Procedure of our algorithm for matching quality between two images are shown in Algorithm 2.

Algorithm 2. Procedure of the developed algorithm for matching quality between two images.

LNQATM: measure matching quality between object template and search object.
1: Given the object template I_T and search object I_S. Where $Func(\cdot|I)$ indicates doing operation along axis of I.
2: $T \leftarrow F(\mathrm{I}_T), S \leftarrow F(I_S)$, Where $F(\cdot)$ indicates features extractor.
3: $\rho_{st} \leftarrow Patch{-}\mathrm{w}iseSimilarity(T,S)$.
4: $L(s|t) \leftarrow Soft\max(\rho_{st}|T)$,
5: $L(t|s) \leftarrow Soft\max(\rho_{st}|S)$.
6: $LNQATM \leftarrow \ln L(\mathrm{s}|t) + \ln L(\mathrm{t}|s)$.
7: $S_{map} \leftarrow Max(LNQATM|\mathrm{T})$.
8: $T_{map} \leftarrow Max(LNQATM|\mathrm{S})$.

Remark 1. *Compared with the existing feature extraction methods based on the shallow and deep NNs have been widely applied to solve this type of problem, the advantage of the developed feature extraction method in this paper is that the fine-tuning training is not required on the new data set when we use shallow and deep networks. In additional, we cut some layers at the end of ResNet-50 to reduce the computational complexity caused by similarity measurement in our algorithm.*

4. Experimental Results and Analysis

We evaluated the developed algorithm on the public benchmark dataset Object Tracking Benchmark (OTB) 2015 [14], which contains 100 sequences with the ground-truth labels and covers various challenging scenes such as DEF, SCs, RCs, BCs and so on. Since the algorithm (MDnet) in [30] is the representative CNN-based tracking method and it demonstrates much better accuracy than other seven state-of-the-art tracking algorithms, we choose it as the compared method.

The developed algorithm is implemented in Python, basing on the TensorFlow framework, and runs at around 1 fps with eight cores of 1.60 GHz Intel(R) Core (TM) i5-8250U and NVIDIA GeForce MX150.

Remark 2. *As shown in Figure 2, Figure 3, Figure 4 some detailed qualitative tracking results show the superiority of the developed algorithm. In the three figures, # number on the left corner of each image denotes the frame index, and the green, blue and red boxes respectively represent the real position, the position predicted by the algorithm (MDnet) in article [30] and the position predicted by the developed algorithm.*

Figure 2. Tracking bounding boxes obtained by two different algorithms on Boy challenging sequences.

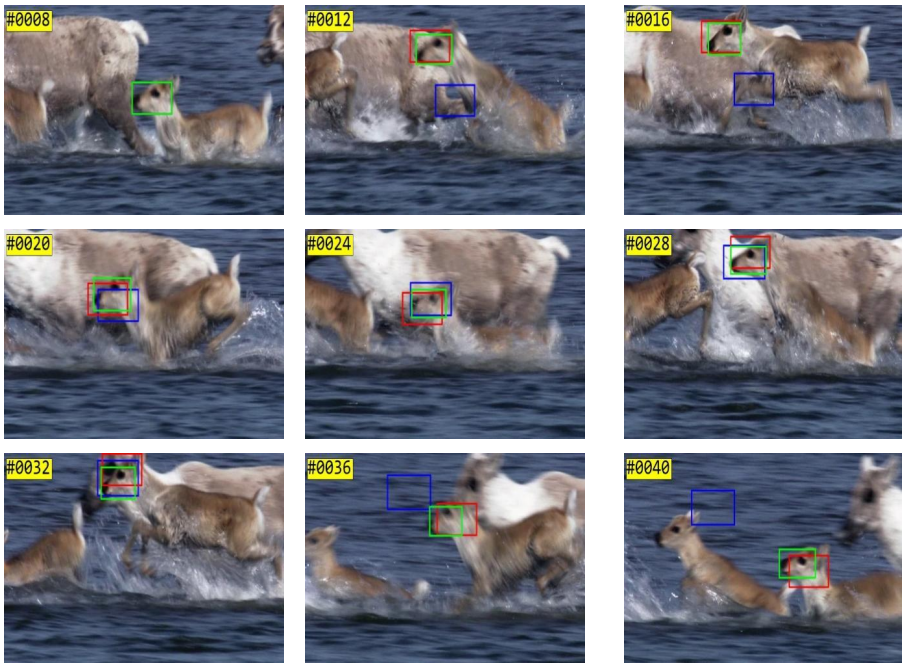

Figure 3. Tracking bounding boxes obtained by two different algorithms on Deer challenging sequences.

Figure 4. Tracking bounding boxes obtained by two different algorithms on Lemming challenging sequences.

To evaluate quantitatively the performance of our approach, we use two common evaluation metrics: the center location error (CLE) and the bounding box overlap. The center location error is the Euclidean distance between the center of the tracking result and ground-truth bounding box. The precision metric is defined as the percentage of correct tracking frames, whose center location error is less than the corresponding threshold. Generally, the threshold is often set to 20 pixels. The bounding box overlap is computed as:

$$S = |RT \cap RG| / |RT \cup RG|,$$

where RT and RG are the tracked and the ground-truth bounding boxes, respectively, and $|\cdot|$ represents the number of pixels in the region. A frame is considered successful if the overlap ratio is larger than the corresponding threshold, which is generally set to 0.5. The success rate is defined as the percentage of successful frames. The results are ranked by the area under curve for the success plot. We employ one-pass evaluation (OPE) to report the overall performance.

Case 1: The tracked target deforms or rotates

In Figure 2, the MDnet algorithm fails to estimate the position of the boy's head in frame 560, frame 565, frame 580, frame 585, frame 590 and frame 595, in which there are more deformation and rotation on the face of the boy. The developed algorithm can better estimate the object position in comparison with MDnet [30]. Figure 5 shows the precision and success plots based on center location error and bounding box overlap ratio, respectively. From Figure 5, we can see that the developed algorithm achieves a precision score of 0.90 and success score of 0.82, which exhibits improvements in the precision and success scores by 11% and 18%, respectively compared to MDnet.

Case 2: There is a very similar target interference next to the tracked target

In the Deer sequence, the object has the similar image features as its background. As shown in Figure 3, the deer is not tracked by MDnet in four frames, i.e., frame 12, frame 16, frame 36 and frame 40. The developed algorithm can reliably track the object throughout.

As is shown in Figure 6, the developed algorithm achieves a precision score of 0.54 and success score of 0.38, which exhibits improvements in the precision and success scores by 7% and 15%, respectively compared to MDnet [30].

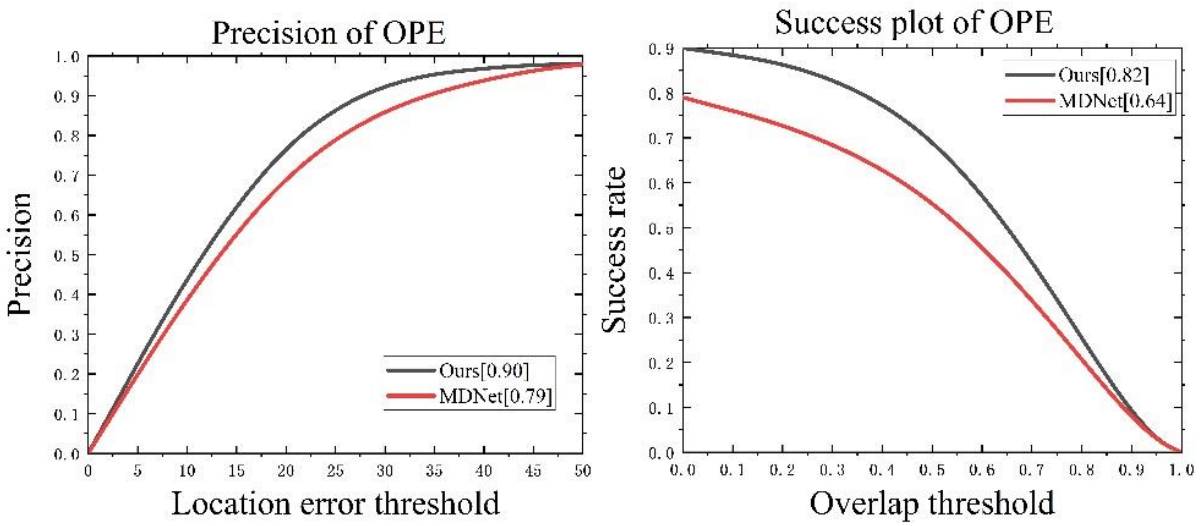

Figure 5. Precision and success plots of the MDnet [30] and the developed algorithm for Boy challenging sequences.

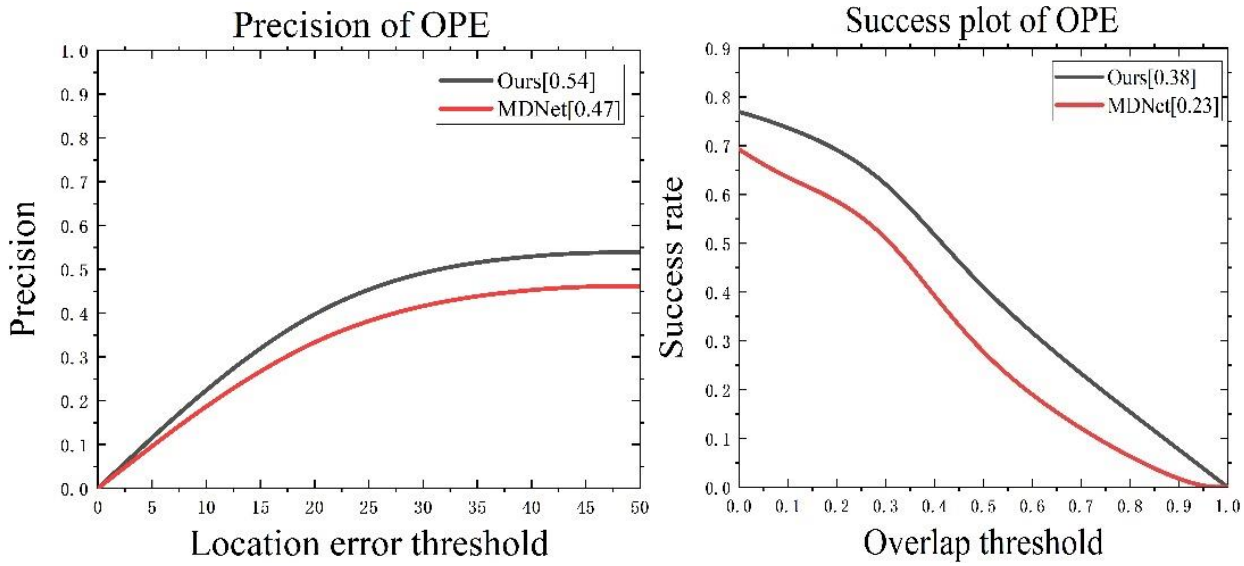

Figure 6. Precision and success plots of the MDnet [30] and the developed algorithm for Deer challenging sequences.

Case 3: The background of the tracked target is very complex

In Figure 4, the Lemming sequence has more complex background. Except for Frame 512 and Frame 542, MDnet loses the object in all the remaining seven frames. However, the developed algorithm performs reliably throughout the whole video. Figure 7 illustrates that the developed algorithm achieves a precision score of 0.85 and success score of 0.71, which exhibits improvements in the precision and success scores by 5% and 2%, respectively compared to MDnet [30].

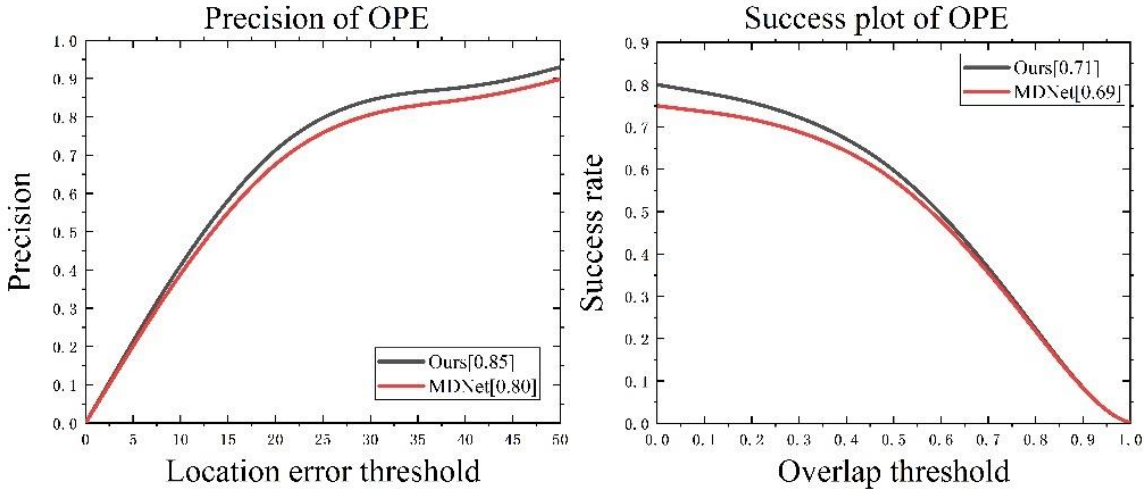

Figure 7. Precision and success plots of the MDnet [30] and the developed algorithm for Lemming challenging sequences.

Moreover, in Figure 8, the average precision and success scores for three challenging sequences are shown. We can see that the developed algorithm achieves an overall precision score of 0.74 and an overall success score of 0.64, whereas MDNets are 0.66 and 0.52, respectively. The developed algorithm is obviously superior to MDnet [30].

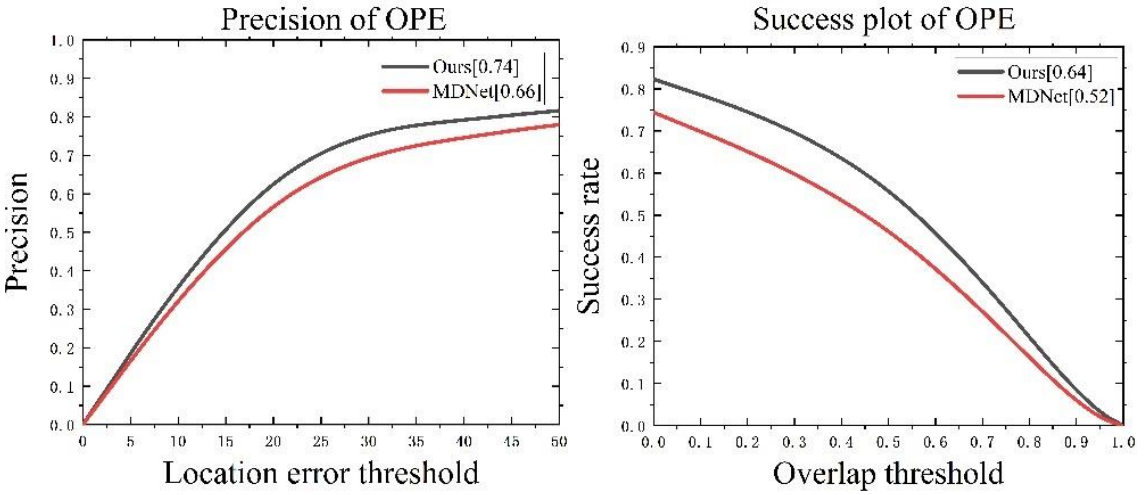

Figure 8. Average precision and success plots of the MDnet [30] and the developed algorithm for three challenging sequences.

5. Conclusions

A large number of experiments show that the algorithm in [30] can not accurately track the targets in the following three cases in video sequences: the tracked target deforms or rotates; there is a very similar target interference next to the tracked target; the background of the tracked target is very complex. To make up for these shortcomings, a new target tracking algorithm based on the residual neural network model and machine learning is proposed in this paper. Compared with the widely used VGG network, residual neural network has deeper characteristic layers and special additional layer structure. The additional layer and convolution layer are used for feature fusion to represent the target. In this paper, we extract the texture features and semanic clues features information of the target, respectively. Therefore, the multi-features of the object can be captured by the developed algorithm, so that the accuracy of tracking can be improved in some complex scenarios. In addition, a new measure is defined to calculate the similarity of different image regions to

find the optimal matched region. The search area is delimited according to the continuity of the target motion, which improves the real-time performance of tracking to a certain extent. The experimental results illustrate that the developed algorithm achieved a higher accuracy while taking into account the real time performance, especially the developed algorithm can accurately track three types of targets that can not be tracked by the algorithm of reference [30].

In addition, we need to point out the limitations of the developed algorithm. Compared with [30], the real-time advantage of the developed algorithm is not obvious. In the next work, we plan to optimize the network structure and improve the similarity function to obtain a better target tracking algorithm.

Author Contributions: T.J., Q.Z., J.Y., C.W. and C.L. contributed equally to each part of this work. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Scientific Research Fund of Sichuan Provincial Science & Technology Department under Grants (Nos. 2021YFH0069, 2021YFG0295, 2021YFQ0057, 2021YFG0133, 2022YFS0565, 2022YFN0020), the Scientific Research Fund of Chengdu Science and Technology Burea (Nos. 2022-YF05-01186-SN, 2022-YF05-01128-SN), the Scientific Research Fund of Chengdu University of Information Technology (No. KYTZ201820) of China, and the National Natural Science Foundation of China (No. 12101090).

Institutional Review Board Statement: No applicable.

Informed Consent Statement: Not applicable.

Acknowledgments: The authors would like to thank the referees for their valuable suggestions, which are of great value to improve the level of this paper.

Conflicts of Interest: The authors declare that they have no competing interests.

References

1. Venkatesan, R.; Raja, P.D.A.; Ganesh, A.B. Video surveillance based tracking system. In *Artificial Intelligence and Evolutionary Algorithms in Engineering Systems*; Springer: New Delhi, India, 2015; pp. 369–378.
2. Bhattacharyya, A.; Bandyopadhyay, S.; Pal, A. ITS-light: Adaptive lightweight scheme to resource optimize intelligent transportation tracking system-customizing CoAP for opportunistic optimization. In *Mobile and Ubiquitous Systems: Computing, Networking, and Services*; Springer: Cham, Switzerland, 2013; pp. 1949–1955.
3. Revell, J.; Mirmehdi, M.; McNally, D. Computer vision elastography: Speckle adaptive motion estimation for elastography using ultrasound sequences. *IEEE Trans. Med. Imaging* **2005**, *24*, 755–766. [CrossRef]
4. Rautaray, S.S.; Agrawal, A. Vision based hand gesture recognition for human computer interaction: A survey. *Artif. Intell. Rev.* **2015**, *43*, 1–54. [CrossRef]
5. Comaniciu, D.; Ramesh, V.; Meer, P. Real-time tracking of non-rigid objects using mean shift. In Proceedings of the 2000 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Hilton Head Island, SC, USA, 15 June 2000; pp. 142–149. [CrossRef]
6. Black, M.J.; Jepson, A.D. EigenTracking: Robust Matching and Tracking of Articulated Objects Using a View-Based Representation. *Int. J. Comput. Vis.* **1998**, *26*, 63–84. [CrossRef]
7. Collins, R.T.; Liu, Y.; Leordeanu, M. On-Line Selection of Discriminative Tracking Features. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1631–1643. [CrossRef]
8. Mei, X.; Ling, H. Robust visual tracking using $\ell 1$ minimization. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 1436–1443. [CrossRef]
9. 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. [CrossRef]
10. Wang, N.; Yeung, D.Y. Learning a Deep Compact Image Representation for Visual Tracking. In *Advances in Neural Information Processing Systems*; Curran Associates Inc.: Lake Tahoe, India, 2013; pp. 809–817. [CrossRef]
11. Cheng, J.; Wu, Y.; AbdAlmageed, W.; Natarajan, P. QATM: Quality-Aware Template Matching for Deep Learning. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 11545–11554. [CrossRef]
12. Liu, T.; Wang, G.; Yang, Q. Real-time part-based visual tracking via adaptive correlation filters. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, USA, 8–10 June 2015; pp. 4902–4912. [CrossRef]

13. Liu, Q.; Zhao, X.; Hou, Z. Survey of single-target visual tracking methods based on online learning. *IET Comput. Vis.* **2014**, *8*, 419–428. [CrossRef]
14. Wu, Y.; Lim, J.; Yang, M. Object Tracking Benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1834–1848. [CrossRef]
15. Grabner, H.; Leistner, C.; Bischof, H. *Semi-Supervised On-Line Boosting for Robust Tracking*; ECCV: Marseille, France, 2008; pp. 234–247. [CrossRef]
16. Babenko, B.; Yang, M.; Belongie, S. Robust Object Tracking with Online Multiple Instance Learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 1619–1632. [CrossRef]
17. Kalal, Z.; Mikolajczyk, K.; Matas, J. Tracking-Learning-Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 1409–1422. [CrossRef] [PubMed]
18. Hare, S.; Golodetz, S.; Saffari, A.; Vineet, V.; Cheng, M.-M.; Hicks, S.L.; Torr, P.H. Struck: Structured Output Tracking with Kernels. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 2096–2109. [CrossRef]
19. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105. [CrossRef]
20. Simonyan, K.; Zisserman, A. Very deep convolutional networks for largescale image recognition. *arXiv* **2014**, arXiv:1409.1556.
21. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 580–587. [CrossRef]
22. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 8–10 June 2015; pp. 3431–3440. [CrossRef]
23. Taigman, Y.; Yang, M.; Ranzato, M.; Wolf, L. DeepFace: Closing the Gap to Human-Level Performance in Face Verification. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1701–1708. [CrossRef]
24. Toshev, A.; Szegedy, C. DeepPose: Human Pose Estimation via Deep Neural Networks. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1653–1660. [CrossRef]
25. Wang, N.; Li, S.; Gupta, A.; Yeung, D.-Y. Transferring Rich Feature Hierarchies for Robust Visual Tracking. Computer Science. *arXiv* **2015**, arXiv:1501.04587.
26. Wang, L.; Ouyang, W.; Wang, X.; Lu, H. Visual Tracking with Fully Convolutional Networks. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 3119–3127. [CrossRef]
27. Gan, Q.; Guo, Q.; Zhang, Z.; Cho, K. First Step toward Model-Free, Anonymous Object Tracking with Recurrent Neural Networks. Computer Science. *arXiv* **2015**, arXiv:1511.06425.
28. Cui, Z.; Xiao, S.; Feng, J.; Yan, S. Recurrently Target-Attending 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. 1449–1458. [CrossRef]
29. Ondruska, P.; Posner, I. Deep Tracking: Seeing Beyond Seeing Using Recurrent Neural Networks. *arXiv* **2016**, arXiv:1602.00991. [CrossRef]
30. 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. [CrossRef]
31. Qi, Y.; Zhang, S.; Qin, L.; Yao, H.; Huang, Q.; Lim, J.; Yang, M.-H. Hedged Deep 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. 4303–4311. [CrossRef]
32. Danelljan, M.; Bhat, G.; Shahbaz Khan, F.; 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. [CrossRef]
33. Bhat, G.; Johnander, J.; Danelljan, M.; Khan, F.S.; Felsberg, M. Unveiling the Power of Deep Tracking. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 493–509.
34. Jiang, C.; Xiao, J.; Xie, Y.; Tillo, T.; Huang, K. Siamese network ensemble for visual tracking. *Neurocomputing* **2018**, *275*, 2892–2903. [CrossRef]
35. Bertinetto, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H. Fully-Convolutional Siamese Networks for Object Tracking. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 8–10 and 15–16 October 2016; pp. 850–865. [CrossRef]
36. Guo, Q.; Feng, W.; Zhou, C.; Huang, R.; Wan, L.; Wang, S. Learning Dynamic Siamese Network for Visual Object Tracking. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 24–27 October 2017; pp. 1781–1789. [CrossRef]
37. Wang, Q.; Teng, Z.; Xing, J.; Gao, J.; Hu, W.; Maybank, S. Learning Attentions: Residual Attentional Siamese Network for High Performance Online Visual Tracking. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4854–4863. [CrossRef]

38. Zhang, Z.; Peng, H. Deeper and Wider Siamese Networks for Real-Time Visual Tracking. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 4586–4595. [CrossRef]
39. Han, G.; Du, H.; Liu, J.; Sun, N.; Li, X. Fully Conventional Anchor-Free Siamese Networks for Object Tracking. *IEEE Access* **2019**, *7*, 123934–123943. [CrossRef]
40. Liu, S.; Chen, Z.; Li, W.; Zhu, J.; Wang, J.; Zhang, W.; Gan, Z. Efficient Universal Shuffle Attack for Visual Object Tracking. In Proceedings of the ICASSP 2022—2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022; pp. 2739–2743. [CrossRef]
41. Zhou, W.; Wen, L.; Zhang, L.; Du, D.; Luo, T.; Wu, Y. SiamCAN: Real-Time Visual Tracking Based on Siamese Center-Aware Network. *IEEE Trans. Image Process.* **2021**, *30*, 3597–3609. [CrossRef]
42. Ondrašovič, M.; Tarábek, P. Siamese Visual Object Tracking: A Survey. *IEEE Access* **2021**, *9*, 110149–110172. [CrossRef]
43. Gao, X.; Zhou, Y.; Huo, S.; Li, Z.; Li, K. Robust object tracking via deformation samples generator. *J. Vis. Commun. Image Represent.* **2022**, *83*, 103446. [CrossRef]