symmetry

MDPI

Article

Siamese Tracking Network with Spatial-Semantic-Aware Attention and Flexible Spatiotemporal Constraint

Huanlong Zhang [1], Panyun Wang [1], Jie Zhang [1,*], Fengxian Wang [1], Xiaohui Song [2] and Hebin Zhou [1]

[1] College of Electrical and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou 450003, China; hlzhang@zzuli.edu.cn (H.Z.); 332101040032@email.zzuli.edu.cn (P.W.); 2019031@zzuli.edu.cn (F.W.); yjsc@zzuli.edu.cn (H.Z.)
[2] Henan Academy of Science, Zhengzhou 450008, China; xhsong@foxmail.com
* Correspondence: 2018007@zzuli.edu.cn

Abstract: Siamese trackers based on classification and regression have drawn extensive attention due to their appropriate balance between accuracy and efficiency. However, most of them are prone to failure in the face of abrupt motion or appearance changes. This paper proposes a Siamese-based tracker that incorporates spatial-semantic-aware attention and flexible spatiotemporal constraint. First, we develop a spatial-semantic-aware attention model, which identifies the importance of each feature region and channel to target representation through the single convolution attention network with a loss function and increases the corresponding weights in the spatial and channel dimensions to reinforce the target region and semantic information on the target feature map. Secondly, considering that the traditional method unreasonably weights the target response in abrupt motion, we design a flexible spatiotemporal constraint. This constraint adaptively adjusts the constraint weights on the response map by evaluating the tracking result. Finally, we propose a new template updating the strategy. This strategy adaptively adjusts the contribution weights of the tracking result to the new template using depth correlation assessment criteria, thereby enhancing the reliability of the template. The Siamese network used in this paper is a symmetric neural network with dual input branches sharing weights. The experimental results on five challenging datasets show that our method outperformed other advanced algorithms.

Keywords: object tracking; aware attention model; spatiotemporal constraint; template updating

check for updates

Citation: Zhang, H.; Wang, P.; Zhang, J.; Wang, F.; Song, X.; Zhou, H. Siamese Tracking Network with Spatial-Semantic-Aware Attention and Flexible Spatiotemporal Constraint. *Symmetry* **2024**, *16*, 61. https://doi.org/10.3390/sym16010061

Academic Editors: João Ruivo Paulo, Cristina P. Santos and Gabriel Pires

Received: 28 September 2023
Revised: 9 November 2023
Accepted: 14 November 2023
Published: 3 January 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In recent years, the performance of target-tracking algorithms has been greatly improved with the development of artificial feature-based trackers and deep learning-based trackers. Target-tracking technology has found widespread applications in computer vision fields such as smart cities, autonomous driving, and video surveillance [1–4]. However, challenges such as target appearance changes during tracking, complex backgrounds, and the presence of similar objects can lead to tracking drift. Therefore, it is still crucial to design a robust tracking algorithm that can effectively handle the target's abrupt motion.

Siamese network-based tracking has drawn extensive attention due to its appropriate balance between accuracy and efficiency. SiamFC [5] maps the search patch into multiple scales and selects the scale with the highest classification score as the target scale for the current frame to predict the bounding box (Bbox). Zhang et al. [6] leveraged deeper and wider convolutional neural networks to further improve the tracking robustness and accuracy. However, the multiple-scale strategy is not well adapted to targets undergoing deformation while increasing the model parameters. Li et al. [7] combined the Siamese network and the region proposal network to predict the scale variation of the target, which improved the model speed and enhanced the adaptability to the deformed targets. To further simplify the model and reduce the computational complexity, some studies [8,9]

introduced the anchor-free mechanism into the tracking field, easing the tuning of complex parameters in anchor-based methods. The above studies are devoted to optimizing the feature extraction network or regression function to improve the accuracy of the Bbox and tracking efficiency. However, such methods still have some limitations.

In real tracking scenarios, a complex background can lead to deviations or even drifting of the tracking prediction box from the ground truth or toward other distractors, especially when the target undergoes drastic appearance changes or moves suddenly over long distances. To address these challenges, existing Siamese network-based trackers [10,11] introduce the centeredness or quality estimation branch independent of the classification branch to suppress excessive displacement, which solves the problem of performance degradation caused by using classification confidence for bounding box selection directly. Chen et al. [12] further proposed the Siamese center prediction network. This model predicts an object's location by correcting the target position appropriately through the offset branch. Some recent methods [13,14] build links between classification and regression, optimizing them in a synchronized manner for consistent inference. Most of these methods add extra branches or networks to improve the accuracy of target localization. In addition, in order to improve the confidence of the response map, some researchers introduced a series of fixed-window penalty functions [15–17] into the tracking model to alleviate the boundary effect, and these methods suppress the interference response to a certain extent. However, a pretrained deep network is not enough to model arbitrary forms of target features when the target state changes significantly, and the extracted target depth features may be redundant. Therefore, it is crucial to adaptively adjust the target features based on different target poses. Additionally, when the target undergoes sudden long-distance movements, the incorrect spatial penalty term can result in the response value of the distractor being higher than that of the target, significantly increasing the probability of tracking drift. Moreover, the absence of a robust target template update mechanism can lead to model degradation during the tracking process.

This work presents a Siamese-based method that addresses the aforementioned limitations. A Siamese network is a symmetric network with two input branches that share the same network structure and weight and is widely used in tracking algorithms. Our method contains a spatial-semantic-aware attention model, a flexible spatiotemporal constraint strategy, and an adaptive weight template update model. The proposed algorithm combines the response results of low-level feature maps and high-level feature maps to determine the target. While multilayer features contain richer target information, the contributions of pretrained target's deep features for visual tracking are different. We establish a spatial-semantic-aware attention model that focuses on the most informative region of the target feature map. This model strengthens feature channels with rich target semantic information by assigning them higher weights. Secondly, we observe that the fixed-window penalty function may decrease the confidence value of the correct target on the score map. To overcome this issue, we designed a flexible spatiotemporal constraint strategy which adaptively adjusts the penalty weights on the confidence map to reduce the probability of tracking failure. In order to further adapt to the target deformation, we designed an adaptive weight template updating strategy to enhance the robustness of the tracking model. The contributions of this work can be summarized as follows:

(1) A spatial-semantic-aware attention model is proposed for visual tracking. We employ a single convolutional spatially aware attention model to adaptively adjust the significance of various feature regions, thereby emphasizing the most informative location on the target feature map. Additionally, the single convolutional channel attention network is used to strengthen target-specific channels that have more target semantic information, which is achieved by increasing their weights. This approach facilitates the learning of effective feature representations for high-tracking performance.

(2) We propose a flexible spatiotemporal constraint which adaptively adjusts the constraint weights on the response map by evaluating the tracking result features. This constraint addresses the issue of the fixed-window function incorrectly penalizing the target

confidence when tracking fails. By incorporating the flexible spatiotemporal constraint, we can obtain a more reliable confidence score for the target location and avoid low-quality but high-scoring tracking results.

(3) We designed an adaptive weight template-updating strategy to mitigate model degradation caused by target appearance changes. This update mechanism evaluates the correlation between the target templates and tracking results using the depth-correlation assessment criteria and thus adaptively assigns weights to both the templates and tracking results to gather reliable template samples. Our update mechanism prevents template contamination while enriching template information.

In this paper, we first briefly review some classical tracking algorithms in recent years, especially those involving attention mechanisms and spatiotemporal constraints, and discuss some disadvantages of the current approaches. Next, we describe in detail our proposed spatial-semantic-aware attention Siamese tracking with a flexible spatiotemporal constraint. Extensive experiments on the OTB100, NFS, UAV123 VOT2016, and TC128 datasets demonstrate the superiority of our approach. Finally, the advances of the proposed methods are summarized, and its limitations are discussed.

2. Related Work

2.1. Siamese-Based Trackers

The Siamese structure proposes a similarity learning problem in which the similarity loss between two input images is calculated, and the shared parameters of two parallel convolution laminar flows are trained by backpropagation. Siamese-based trackers [18–20] solve the similarity matching problem between the target and the search area well and have become popular in the tracking field (see Table 1). SiamFC [5] performs similarity learning through deep cross-correlation, which transforms the tracking process into template matching. SiamFC has greatly improved tracking accuracy and efficiency compared with previous studies, but it is not well adapted to various challenging scenarios. To further improve the tracking performance, researchers have designed a number of Siamese trackers based on SiamFC that can adapt to more tracking challenges. SiamRPN [7] first implanted the region proposal network (RPN) [21] into the Siamese trackers to transform the global search into a region-specific detection task [22]. The bounding box regression reduces the amount of computation while improving the accuracy. In order to obtain more abundant target information, SiamMCF [20] and DSiam [18] incorporate cross-correlation on multiple layers to enhance the discriminant ability of the model. SiamBAN [8] and SiamCAR [10] designed the anchor-free strategy to avoid complex parameters caused by setting anchor boxes and further enhance the flexibility of the model.

2.2. Trackers with the Attentional Mechanism

The attentional mechanism was first applied in neuroscience and has expanded into other fields, such as image detection [23,24] and target tracking [25]. DVAT [26] proposed the concept of 'attention regions', which applies spatial attention to two different discriminative domains (local and semi-local), effectively focusing the attention of the tracker. RTT [27] developed recurrent neural networks (RNNs) to identify components that are useful for target modeling and then used the resulting confidence map to suppress background clutter. Wang et al. [28] constructed a Residual Attentional Siamese Network (RASNet) using different types of attention mechanisms to improve the discriminative ability of the tracking model. Rahman M et al. [29] added channel attention and spatial attention behind the pretrained features to further reduce the distracting information in the target template features. In contrast to these attention mechanisms, it is proposed to learn attention through an online training deep network. We use the single convolutional attention network framework to adaptively adjust the importance of spatial and channel features and target representation.

2.3. Trackers with Spatiotemporal Constraint

Since the target motion was mostly smooth in early target tracking, researchers developed a spatiotemporal constraint strategy to suppress tracker drift. MOSSE [30] and KCF [31] avoid boundary effects in the tracking process by introducing cosine window constraints. In addition to cosine windows, other tracking algorithms also introduce window functions such as the Blackman window [15], rectangular window [16], and Hamming window [17] to prevent boundary effects. When the target displacement between two frames is small, and the background is simple, these fixed spatiotemporal constraint functions generally improve the tracking results. However, when the tracker fails to track due to the large displacement of the target between two frames in the video, these fixed spatiotemporal constraints may cause a large weight loss for the confidence of the correct target on the response map, and thus the tracker cannot be corrected. Li et al. [32] developed the NA window to suppress these customized incorrect weights, which improves the SNR of windowed ROIs by adaptively suppressing the variable noise according to the observation of similarity maps. Different from the above methods, this paper focuses on adjusting the spatiotemporal constraint under different conditions according to the quality of the tracking result box, which adaptively adjusts the penalty weights during the tracking process to reduce the probability of tracking failure.

Table 1. Related works overview.

Tracker Type	Related Tracker	Year	Peculiarity
Siamese-based trackers	SiamFC [5]	2016	Similarity learning
	SiamRPN [7]	2018	Region proposal network
	DSiam [18]	2017	Use multilayer features
	SiamMCF [20]	2018	Use multilayer features
	SiamBAN [8]	2020	Anchor-free strategy
	SiamCAR [10]	2020	Anchor-free strategy
Trackers with the attentional mechanism	DVAT [26]	2010	Local and semi-local attention regions
	RTT [27]	2016	Recurrent neural networks
	RASNet [5]	2018	Residual Attentional Siamese Network
	SCSAtt [29]	2020	Stacked channel-spatial attention
Trackers with a spatiotemporal constraint	MOSSE [30]	2010	Cosine window constraint
	KCF [31]	2014	Hamming window
	NA [32]	2020	Noise-aware framework

3. The Proposed Method

Aiming at the problem that most tracking methods easily fail during abrupt motion and target appearance changes, a tracking framework is proposed that can handle this problem. In specific tasks, the depth features acquired by the pretrained network have different importance to the target representation, resulting in a worse discrimination ability for the tracker regarding the target and background information. Immediately after, we find that the fixed-window function unreasonably weights the confidence values on the response map to produce lower-quality but higher-scoring tracking results. Finally, we develop a flexible template-updating strategy to mitigate model degradation.

Figure 1 shows the overall scheme of our proposed algorithm. It is a tracker based on classification and regression that uses ResNet-50 as the feature extraction network. In our work, Conv3, Conv4, and Conv5 from the ResNet-50 [33] network were selected to extract image features. Since the shallow features contain more spatial structure information while the deep features contain rich semantic information [34], we introduced spatial attention in Conv3 and channel attention in Conv4 and Conv5 to highlight the information that was valuable for target representation. Then, the classification (Cls) map and regression (Reg) map were obtained by correlation matching between the target template feature and the search area feature. The Cls map estimated the probability that each position in the search

area was the target, and the Reg map performed bounding box prediction. Next, the multilayer depth features of each frame-tracking result were compared with the template based on the Euclidean distance to determine whether the flexible spatiotemporal constraint strategy was activated. The flexible spatiotemporal constraint strategy was activated, which gradually increased the weight of the edge of the confidence map with the time of target loss to help the tracking recover. Finally, the adaptive weight template-updating strategy was used to generate a new template for the next frame tracking.

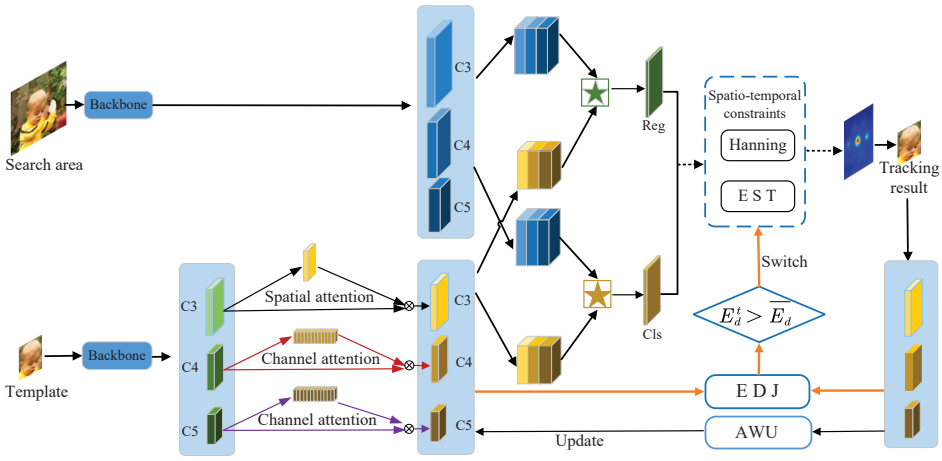

Figure 1. The proposed tracking framework, where FST represents the flexible spatiotemporal constraint strategy, EDJ represents the judgment based on the Euclidean distance, and AWU stands for adaptive weight template update.

3.1. Spatial-Semantic-Aware Attention Model

Human visual perception usually does not need to focus on the whole environment but rather on the part of the target to perceive comprehensive information about and thus understand the corresponding visual patterns [35]. The coordinate attention [36] enables mobile networks to focus on a larger area by embedding positional information into the channel domain. Yang et al. [23] proposed that dual wavelet attention can coordinate spatial and structural attention for different channels to prevent the loss of feature information and structural features. Since single-target tracking is similar to focusing on the most salient features, it is advantageous to focus on the critical regions of the target feature map. Unlike other trackers with attention mechanisms, we propose a spatial-semantic-aware attention model where the spatial-aware attention model focuses on prominent target regions in the shallow feature map, while the semantic-aware attention model distinguishes the importance of different channels of deeper features for target representation.

3.1.1. Spatial-Aware Attention Model

For the tracking target, the depth features are constructed by multiple two-dimensional feature maps. However, the contribution of all regions of the depth features obtained by the pretrained network to the tracking task is not equally important, and only the location related to the task needs to be focused upon.

Spatial attention focuses on 'where' an informative part is and enhances the informative features of the target in the image to facilitate target localization. To program this attention, we performed global max pooling $GP_{\max}(\cdot)$ and average pooling $GP_{avg}(\cdot)$ on the Conv3 feature map $F_M^{H\times W\times C}$ and fused the resulting pooling features $F_{\max}^{H\times W\times 1}$ and $F_{avg}^{H\times W\times 1}$ in the channel domain. This kind of local convolution operation can focus on the desired information on the feature map.

After fusing the doubly pooled features, we used a convolution layer $\psi_1^{3\times 3}$ to downsample the number of feature channels to one to obtain a single-channel feature map (the 3×3 convolution filter was selected as the best result through experimentation). Then, the

obtained single-channel convolution feature map was broadcasted with a sigmoid operation, and the single-channel convolutional feature map was multiplied by the previous Conv3 feature map $F_M^{H\times W\times C}$ to obtain the spatial attention feature map $S_A^{H\times W\times C}$, with the ultimate effect shown in Figure 2. The computation of the attention feature map can be described as follows:

$$F_{\max}^{H\times W\times 1} = GP_{\max}\left(F_M^{H\times W\times C}\right), \tag{1}$$

$$F_{avg}^{H\times W\times 1} = GP_{avg}\left(F_M^{H\times W\times C}\right), \tag{2}$$

$$\phi_s(\cdot)^{H\times W\times 1} = \delta\left(\psi_1^{3\times 3}\left(concat\left[F_{\max}^{H\times W\times 1}, F_{avg}^{H\times W\times 1}\right]\right)\right), \tag{3}$$

and

$$S_A^{H\times W\times C} = \phi_s(\cdot)^{H\times W\times 1} \otimes F_M^{H\times W\times C}, \tag{4}$$

where $concat[\cdot]$ represents the concatenation operation, $\psi_1^{3\times 3}$ is the convolution operation with the 3×3 kernel, the padding and stride are one, and δ represents the usual sigmoid function $f(x) = \frac{1}{1+e^{-x}}$.

In addition, to make our aware attention mechanism more compatible with different targets, we adjusted the feature weights online utilizing a single convolutional. The specific method involved convolving all the samples $S_A^{H\times W\times C}$ acquired by the attention mechanism into one-dimensional features and regressing them to a Gaussian label map $Y(i,j) = e^{-\frac{i^2+j^2}{2\sigma^2}}$, where (i,j) is the offset against the target and σ is the kernel width. Then, the new aware attention weight $\phi_s(\cdot)^{H\times W\times 1'}$ was obtained by minimizing the following objective function:

$$L = \|Y(i,j) - W \odot S_A^{H\times W\times C}\|^2 + \lambda\|W\|^2, \tag{5}$$

where $\odot$ denotes the convolution operation and W is the regression weight, while λ is a regularization parameter which can inhibit the overfitting of the training process.

After online training of the target in the first frame, we could find better attention weights $\phi_s(\cdot)^{H\times W\times 1'}$. Lastly, the features which reinforced the target area were obtained by the spatial-aware attention module as follows:

$$S_A^{H\times W\times C'} = \phi_s(\cdot)^{H\times W\times 1'} \otimes F_M^{H\times W\times C}, \tag{6}$$

3.1.2. Semantic-Aware Attention Model

Some feature channels have a more prominent contribution to modeling the visual pattern of an object; that is, different channels contain different semantic information about the target. Therefore, each channel should not be treated equally when using these depth features for tracking.

For Conv4 and Conv5 obtained from the backbone network, the global average pooling operation was performed on them (the squeeze process), the detailed operations of which were as follows:

$$F_{avg}^{1\times 1\times C} = F_{sq}\left(F_M^{H\times W\times C}\right) = \frac{1}{W\times H}\sum_{i=1}^{W}\sum_{j=1}^{H} F_M^{H\times W\times C}(i,j), \tag{7}$$

where $F_{sq}(\cdot)$ represents the squeeze process, $F_M^{H\times W\times C}$ represents the Conv4 or Conv5 features, and W, H are the width and height of the feature map, respectively.

We obtained two feature vectors $F_{avg}^{1\times 1\times 1024}$ and $F_{avg}^{1\times 1\times 2048}$ through two levels of full connection (the excitation process). The first full connection fc_1 compressed C channels into $\frac{C}{r}$ channels to reduce computation, and the second full connection fc_2 reverted to C channels. The excitation process can be expressed as follows:

$$\phi_c(\cdot)^{1\times1\times C} = F_{ex}\left(F_{avg}^{1\times1\times C}\right) = \sigma\left(fc_2\left(\mathrm{ReLU}\left(fc_1\left(F_{avg}^{1\times1\times C}\right)\right)\right)\right), \tag{8}$$

where σ represents the usual sigmoid function $f(x) = \frac{1}{1+e^{-x}}$ and $\mathrm{ReLU}(\cdot)$ is a rectified linear unit layer.

Then, the $\phi_c(\cdot)^{1\times1\times1024}$ and $\phi_c(\cdot)^{1\times1\times2048}$, as feature weights, are multiplied by the corresponding channels of the features of Conv4 and Conv5 to acquire the output features for the channel attention model:

$$C_A^{H\times W\times C} = \phi_c(\cdot)^{1\times1\times C} \otimes F_M^{H\times W\times C}, \tag{9}$$

Similar to Section 3.1.1, we assigned different weights to each channel utilizing a single convolutional. We convolved all the multi-channel $C_A^{H\times W\times C}$ values acquired by the channel attention mechanism into one-dimensional features and regressed them to a Gaussian label map. The better aware attention weight $\phi_c(\cdot)^{H\times W\times1'}$ was obtained by minimizing the following objective function:

$$L = \|Y(i,j) - W \odot C_A^{H\times W\times C}\|^2 + \lambda\|W\|^2, \tag{10}$$

Lastly, the target semantic features obtained by the semantic-aware attention module were as follows:

$$C_A^{H\times W\times C'} = \phi_c(\cdot)^{1\times1\times C'} \otimes F_M^{H\times W\times C}, \tag{11}$$

Figures 2 and 3 show our spatial-aware attention model and semantic-aware attention model frameworks, respectively. The method enhances the effective features online and weakens those that are redundant or even interfering with the tracking.

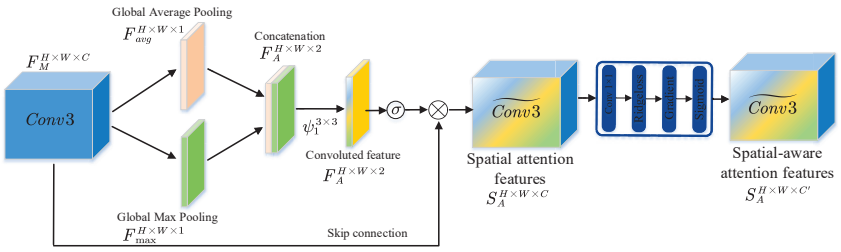

Figure 2. Proposed spatial-aware attention module.

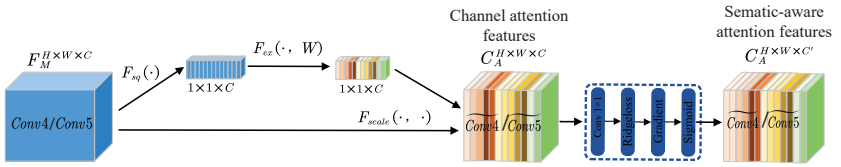

Figure 3. Proposed semantic-aware attention module.

3.2. Flexible Spatiotemporal Constraint

Most of the existing trackers were proposed under the assumption of smoothness; that is, researchers assume that the target displacement between two frames will not be too large, and thus various window functions were proposed to punish the final response graph (assign a value [0,1] according to the distance between the sample center and the target in the previous frame). This can improve the confidence of the target response to a certain extent. But in the actual tracking scene, there will always be some similar targets or other interference information that leads to tracker drift. Once the tracking fails, the response of the correct target location will be continuously suppressed under the action of the fixed-window function, resulting in low-quality but high-scoring tracking results. The fixed-window function (Hanning window) fails to correct the tracker when the target

deviates too far from the center of the search area, as shown in Figure 4. Therefore, to reduce the continuous negative impact of fixed spatio-temporal constraints on the target when the tracker fails, we developed a flexible spatiotemporal constraint strategy.

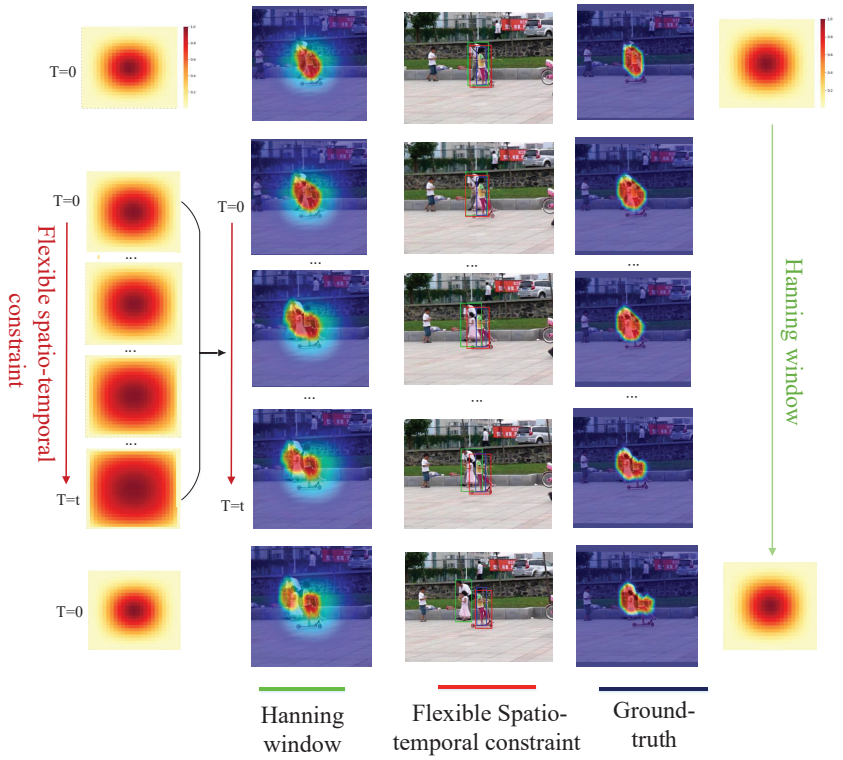

Figure 4. Tracking results and target-scored heat map visualization results under the Hanning window and flexible spatiotemporal constraint.

Generally speaking, due to the smoothness assumption, the depth features of the target will not change greatly between adjacent frames. Therefore, when the tracker produces low-quality tracking results, the depth features of the tracking result will be significantly different from the template features. Based on this, we can consider whether to switch the spatiotemporal constraint by evaluating the depth features of both the tracking result and the target template. We expanded the tracking results to the same size as the target template and used the backbone network ResNet-50 to obtain the three-layer depth features of the tracking results. For the tracking result and target template, we compared the depth features of their corresponding layers based on the Euclidean distance. We will switch the spatiotemporal constraints when Equation (12) is met:

$$\|T^G_{0,L} - T^R_{t,L}\|_2 > \overline{E_d}, \tag{12}$$

$$\|T^G_{0,L} - T^R_{t,L}\|_2 = mean\left[\sum_l \sqrt{\sum_{x,y}\left(T^G_{l,x,y} - T^R_{l,x,y}\right)^2}\right], \tag{13}$$

$$\overline{E_d} = \frac{\sum_{t=2}^{t-1}\|T^G_{0,L} - T^R_{t-1,L}\|_2}{t-1}, t > 1, \tag{14}$$

where $T^G_{0,L}$ is the initial template feature, $T^R_{t,L}$ is the tracking result feature (t is the sequence number of frames, while L is the layer of the features index), and $T^G_{l,x,y}$ and $T^R_{l,x,y}$ are the feature pixel values of the template and tracking result, respectively (l is the channel ordinal number, while x and y represent the pixel position index).

We observed that when the tracking error was caused by a change in target appearance, although the confidence score of the correct target was higher on the response graph without

applying the window penalty function, due to the fixed spacetime constraints, the response of the target far away from the center of the search area would be suppressed, and thus the tracker could not recover to the correct target. However, in most cases, the window function could reduce the likelihood that the tracker would track similar objects far from the center point of the search area. Based on this, we established a flexible spatiotemporal constraint to penalize the target confidence score S_{con} on the response map. More details are shown in Figure 4. Our strategy is defined as follows:

$$S_{con}^{new} = \rho_{con} \times S_{con} + (1 - \rho_{con}) \times Q_i^{new}, \tag{15}$$

$$Q_i^{new} = Q_0 \times e^{-\alpha(t+\sigma)}, \tag{16}$$

$$\alpha = \ln\left(\frac{Q_i}{Q_f}\right) \times \frac{1}{n}, \tag{17}$$

$$\sigma = \ln\left(\frac{Q_i}{Q_0}\right) \times \left(-\frac{1}{\alpha}\right), \tag{18}$$

where ρ_{con} is a predefined hyperparameter and determines the degree to which the flexible spatiotemporal constraint affects the original response map. If ρ_{con} is set to a large value, then the flexible spatiotemporal constraint has minimal impact on the original response map, which may cause the response value far from the center of the response map to be too large, resulting in the boundary effect. On the other hand, if ρ_{con} is set to a small value, then the final response map is primarily determined by the flexible spatiotemporal constraint, and the initial response map output by the tracker is largely disregarded. This will greatly reduce the confidence of the response map, where α is the expansion rate indicating the distance penalty, σ represents the amount of translation to the left, which allows the value to continue expanding from any position without having to start from Q_0 (Q_0 is one), Q_i, n, and Q_f represent the initial value, the expansion time length, and the final value, respectively, and each Q_i represents the spatiotemporal constraint weight of the original position, in which different Q_i values form different expansion curves.

3.3. Adaptive Weight Template Updating

In practical tracking tasks, most tracker models continuously degrade due to the constant change in target appearance, resulting in tracker drift. Some Siamese trackers utilize the target state given in the first frame to obtain an initial template and do not update it again [5,7]. Most update functions are limited to linear combinations with previous templates, and fixed combination weights severely limit the universality of the update mechanism. In order to make the template dynamically update to reduce model degradation and prevent contamination of the template from undifferentiated updates, we developed an adaptive weight template-updating strategy which can dynamically fuse the tracking results to generate the cumulative template for subsequent frame tracking.

First, the object defined by the ground truth in the initial frame has its most reliable original information, and thus we used the appearance features of the initial template as a baseline for the tracking results of the subsequent frames to generate cumulative templates by using a convolutional neural network to learn the target information that the initial template had. A new cumulative template was updated for each frame during the tracking process. For each frame to be tracked, its corresponding template T_t^C was generated from three components: the initial template, the cumulative template T_{t-1}^C, and the tracking result T_{t-1}^R for the previous frame. This would give the template richer temporal information. The generation process can be formalized as follows:

$$T_t^C = T_0^G + conv\left[T_{t-1}^R, T_{t-1}^C, T_0^G\right], \tag{19}$$

where $conv[\cdot]$ represents the convolutional operation and t is the sequence index of the frame.

Furthermore, as can be seen in Figure 5, the tracking result features of different frames differed significantly from the initial template features due to the constant changes in the target appearance. Even for the same object, updates to the target template needed to change dynamically based on the tracking state. If all tracking results were utilized indiscriminately to update the template, then this may have led to redundancy or contamination of the template.

Therefore, we estimated the correlation between the cumulative template and the initial template and assigned weights to both by means of depth correlation assessment criteria. Since the depth features of different layers had different contributions to the final response map, we conducted depth cross-correlation between the three depth features of the initial template and the corresponding layers of the cumulative template. Then, we found the ratio with the autocorrelation of the initial template features to generate the weights of the corresponding layers. According to the weight of the corresponding layer, the feature of the tracking result and the feature of the cumulative template were fused to generate a new cumulative template for the tracking of the next frame. Note that the target from the first frame provided the most reliable information, and therefore we set the shrink parameter so that the template retained more of the initial information of the target. The following is thte recursive formula for the template update:

$$T^C_{t,L} = T^G_{0,L} + \lambda_{t,L} \cdot conv\left[T^R_{t-1,L}, T^C_{t-1,L}, T^G_{0,L}\right], \tag{20}$$

$$\lambda_{t,L} = \frac{conv\left[T^R_{t-1,L}, T^C_{t-1,L}, T^G_{0,L}\right] \circledast T^G_{0,L}}{T^G_{0,L} \circledast T^G_{0,L}}, \tag{21}$$

where L is the layer of the features index and t is the sequence index of the frame. The operator $\circledast$ denotes the cross-correlation operation (i.e., the former is used as a convolution kernel to perform convolution operations on the latter).

It can be seen in Figure 5 that our model can improve the template degradation caused by target deformation or target background changes.

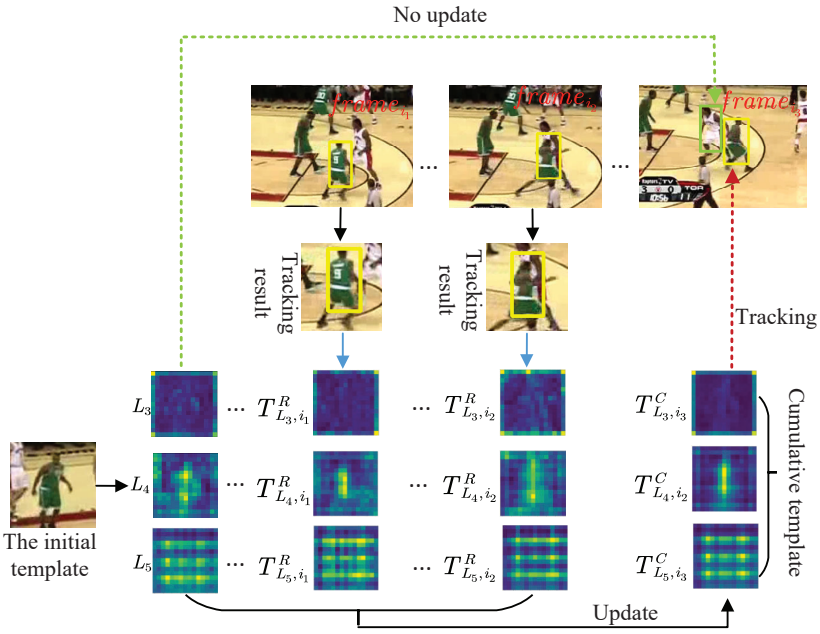

Figure 5. Visualization of the features of the initial template, tracking results, and the cumulative templates. The green box represents the tracking result when the template is not updated. The yellow box represents the tracking result under our updating mechanism.

4. Experiments

4.1. Settings and Datasets

The method presented in this article was implemented with Pytorch. The experiment was conducted on a PC with 16.0 GB of RAM, an Intel(R) Core(TM) i7-10700 CPU 2.90 GHz, and an NVIDIA GeForce GTX 1660 SUPER GPU. Our tracker was evaluated on four datasets: OTB100 [37], UAV123 [38], NFS [39], VOT2016 [40], and TC128 [41]. The Conv3, Conv4, and Conv5 datasets on ResNet-50 [33] were used to extract the depth features of the target template and the search region. We enhanced the target representation by introducing spatial attention and channel attention, which could suppress the depth features of interference information. The number of iterations during online training affected the quality of the aware attention features. If the number of iterations is too low, then the loss value will not decrease effectively. Consequently, the distractor information in the target template features will not be adequately suppressed, and the tracking success rate will not improve. Conversely, if there are too many iterations, then the training time will be prolonged, thereby negatively affecting the tracking speed. To determine the optimal number of iterations, we conducted a comparative experiment within the range of 100–800 iterations. The experiment utilized the OTB100 dataset, and the results are presented in Table 2. Considering the balance between tracking accuracy and speed, and taking into account the resource limitations, we set the maximum number of iterations during training of the aware attention models to 500 based on the experimental findings. The learning rates of the neural network were 0.02 (Conv3), 0.05 (Conv4), and 0.0006 (Conv5).

Table 2. The experimental results for different iteration numbers on OTB100.

Iteration Number	Success Rate (%)	FPS
100	68.4	10.2
200	68.6	9.3
300	68.7	8.5
400	69.0	8.1
500	69.2	7.9
600	69.3	7.3
700	69.0	7.0

4.2. Results on OTB100

OTB100 is one of the most widely used datasets in the field of tracking, and it consists of 100 video sequences. The test sequence includes deformation illumination variation, out-of-plane rotation, scale variation, in-plane rotation, occlusion, motion blur, fast motion, and other challenging aspects. The evaluation was based on two metrics: success and precision plot metrics. The precision plot metric is the percentage curve of video frames, whose center position error is less than a given threshold. The success plot metric is the percentage curve of video frames with border overlap greater than a given threshold. We compared our tracker with some state-of-the-art trackers (SiamFC++ [11], SiamBAN [8], DaSiamRPN [42], GradNet [43], DeepSRDCF [44], SiamRPN [7], SiamDW-FC [6], SRDCF [45], SiamFC [5], and fDSST [46]). As shown in Figure 6, the performance of our tracker on both benchmarks was at an advanced level. Our tracker provided varying degrees of gain compared with the SiamBAN tracker.

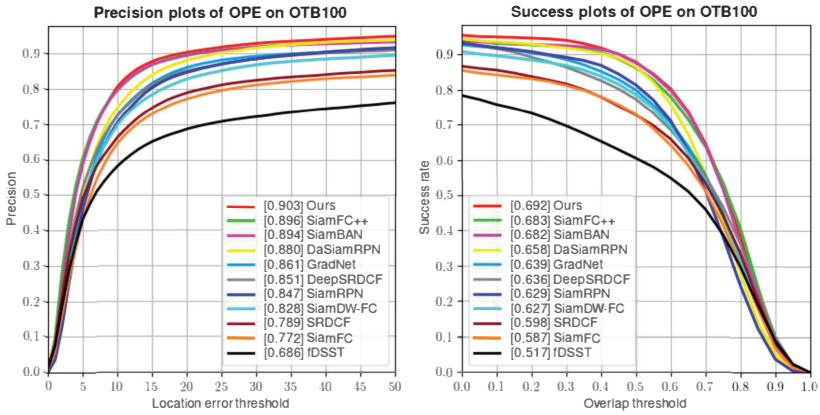

Figure 6. Success and precision plots on OTB100.

4.3. Attribute-Based Comparison

We tested the tracking results of the proposed method on the OTB100 dataset under six common challenges: background clutters, fast motion, motion blur, occlusion, being out of view, and scale variation. Figure 7 shows the comparison results between our tracker and other trackers for the six different attributes. It can be seen that the proposed method coped better with various tracking challenges. In particular, compared with SiamBAN [8], our tracker performed better under the background clutter challenge, with a 1.6% improvement in the success rate and 1.4% higher accuracy for the background clutters, which can indicate that the spatial-semantic attention learning model effectively distinguished the targets from the background and similar objects. Moreover, when the target had fast motion and scale variation, the excellent results produced by our method show that the flexible spatiotemporal constraint can reduce the error response caused by the abrupt change in the target.

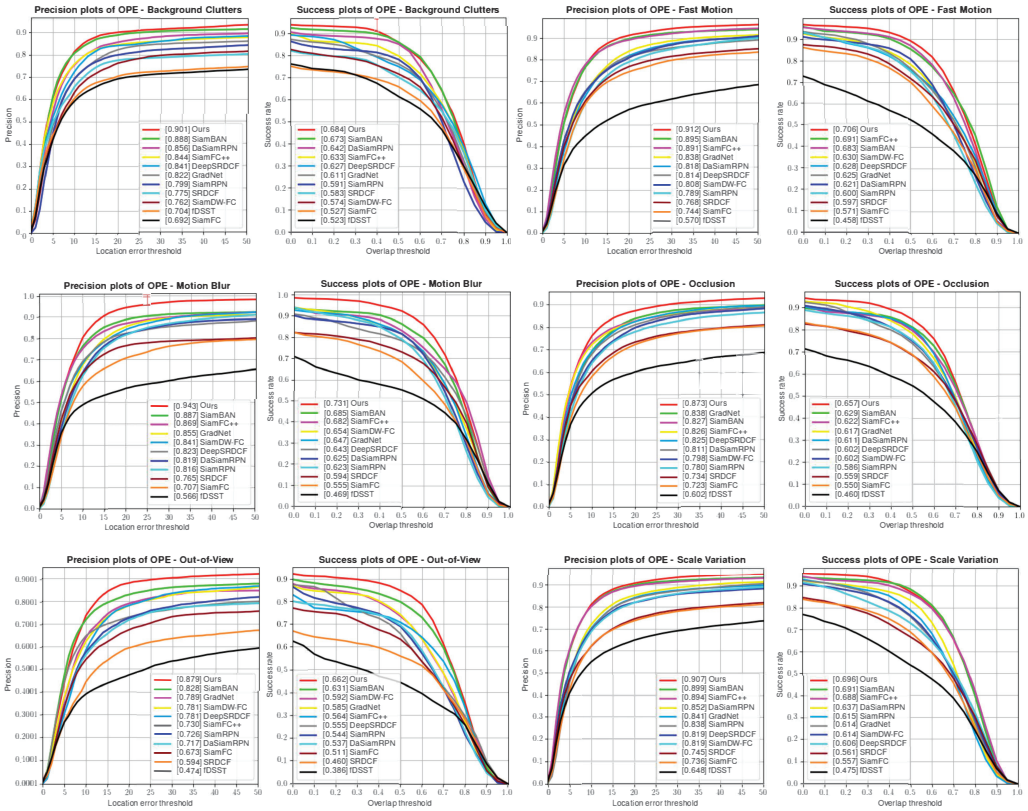

Figure 7. The precision plots and success plots for six challenging attributes on the OTB100 dataset.

4.4. Results on UAV123

The UAV123 dataset has video captured by a low-altitude drone, containing 123 videos characterized by a large number of viewpoint changes. We tested our algorithm on the UAV123 dataset using the same evaluation metric as OTB. (The precision is the percentage of video frames whose center position error is less than a given threshold, and the success rate is the percentage of video frames with border overlap greater than a given threshold). Table 3 shows the comparison of our method with ATOM [47], SiamBAN [8], SiamRPN++ [48], DaSiamRPN [42], SiamRPN [7], ECO [49], TCTrack++ [49], SRDCF [45], SiamTHN [50], LGFF [51], and SAMF [52]). The results show that our tracker had 1.9% higher accuracy and a 2.6% higher success rate compared with SiamBAN. However, ATOM had better performance in terms of precision compared with most classification regression-based trackers. This was due to the fact that ATOM introduced an online trained classification component, which allowed the network to estimate the target state with higher classification confidence. Therefore, the centroid of the bounding box was closer to the true position.

Table 3. Comparison of results of other trackers with ours on UAV123.

Tracker	Precision	Success Rate
ATOM	0.856	0.643
SiamRPN++	0.840	0.642
Ours	0.849	0.648
SiamTHN	0.836	0.635
LGFF	0.834	0.632
SiamBAN	0.833	0.631
DaSiamRPN	0.781	0.569
SiamRPN	0.768	0.557
TCTrack++	0.731	0.519
ECO	0.741	0.525
SRDCF	0.676	0.464
SAMF	0.592	0.395

4.5. Results on NFS

The Need for Speed (NFS) dataset consists of 100 video sequences captured from real-world scenarios, with a total frame count of up to 380,000. All sequences are pre-annotated with different visual attributes such as occlusion, fast motion, and background clutter. We evaluated our tracker in the 30 FPS version of the dataset. The area under the curve (AUC) score of the success plot reflects the overall tracking results. Table 4 lists the evaluation results of our approach as well as MDNet [53], ECO [54], C-COT [55], UPDT [56], ATOM [47], SiamBAN [8], and LGFF [51]. Our tracker ranked second out of all the methods that participated in the comparison.

Table 4. Comparison with state-of-the-art trackers on the NFS dataset in terms of AUC.

	MDNet	ECO	C-COT	UPDT	ATOM	SiamBAN	Ours	LGFF
AUC	0.422	0.466	0.488	0.537	0.584	0.594	0.602	0.610

4.6. Results on VOT2016

VOT-2016 consists of video sequences in 60 different scenarios. Unlike OTB's evaluation system, VOT-2016's evaluation metrics include robustness (failure rate, where a lower value is best), accuracy (average overlap during successful tracking periods, where a higher value is best), and EAO (expected average overlap, which quantitatively reflects both robustness and accuracy, where a higher value is best). In the VOT evaluation protocol, the tracker will be reinitialized when no overlap between the prediction Bbox and the ground truth is detected. We compared our tracker with others, namely SiamRPN [7],

C-COT [55], MDNet [53], SiamRN [57], D3S [58], ROAM [59], SPS [60], SiamRNE [61], SiamTHN [50], and SiamBAN [8]. Table 5 shows the evaluation results for each tracker. The EAO of our tracker was further improved compared with SiamBAN, and the failure rate was also reduced, which shows that our spatial-semantic-aware attention model, adaptive template updating, and flexible spatiotemporal constraint strategy can effectively reduce the probability of tracking failure. However, the accuracy of our tracker was worse than that of D3S, which was limited by the fact that the bounding box in the VOT evaluation system is rotatable, while the bounding box that our tracker predicts is flush with the image boundary.

Table 5. Details on the state-of-the-art trackers in VOT2016.

Tracker	EAO	Accuracy	Robustness
SiamRPN	0.344	0.56	1.08
C-COT	0.331	0.53	0.85
MDNet	0.257	0.54	1.2
SiamRN	0.277	0.55	1.37
D3S	0.493	0.660	0.13
SiamBAN	0.505	0.632	0.149
SPS	0.459	0.625	0.158
ROAM	0.441	0.559	0.131
SiamRNE	0.300	0.540	1.120
SiamTHN	0.510	0.625	0.126
Ours	0.515	0.636	0.140

4.7. Results on TC128

To further demonstrate the generality of the proposed tracking method in various scenarios, we tested our tracker on the TC128 dataset, which is more complex than OTB. TC128 has 128 color video sequences, and its tracking scenario is more variable than that of OTB. We compared our method with other trackers, including SiamBAN [8], ADMT [62], SiamCAR [10], SiamGAT [25], MEEN [63], and Struck [64]. Figure 8 shows the precision plots and success plots of seven trackers on the TC128 dataset. It can be seen that the scores of the proposed method on AUC and DP were 79.9% and 58.0%, respectively. Compared with the second-best tracker (SiamBAN), our tracking method increased by 1.6% and 1.8% in terms of the AUC and DP, respectively.

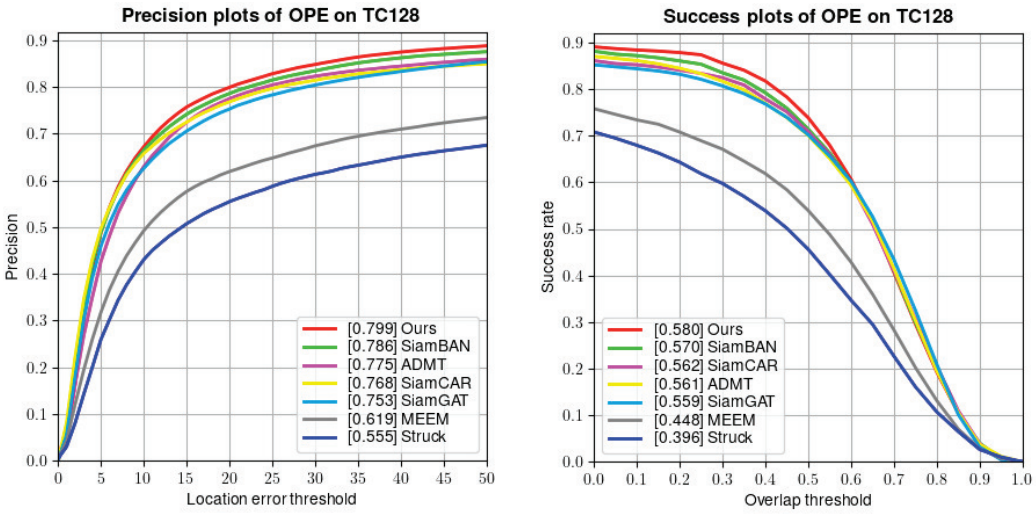

Figure 8. Success and precision plots on TC128.

4.8. Visual Evaluation

To further demonstrate the performance of our tracker in the face of various challenges, we visualized the tracking results of our tracker versus other trackers in real-world scenarios. The scenario in Figure 9 includes tracking challenges such as occlusion (in Matrix and Soccer), scale variation (in Biker, Soccer, Walking2, and Trans), illumination changes (in Matrix and Soccer), deformation (in Trans), and other changes in the appearance of the target, as well as fast motion (in Matrix and Biker) and background clutters (in Matrix, Soccer, and Walking2). It can be seen that trackers such as SiamBAN [8], DaSiamRPN [42], and ADMT [62] would cause the prediction box to be less accurate and even track drift when faced with the above challenges. In contrast, our tracking method can better adapt to target appearance changes and various challenges due to the introduction of the spatial-semantic-aware attention model and spatiotemporal constraint strategy.

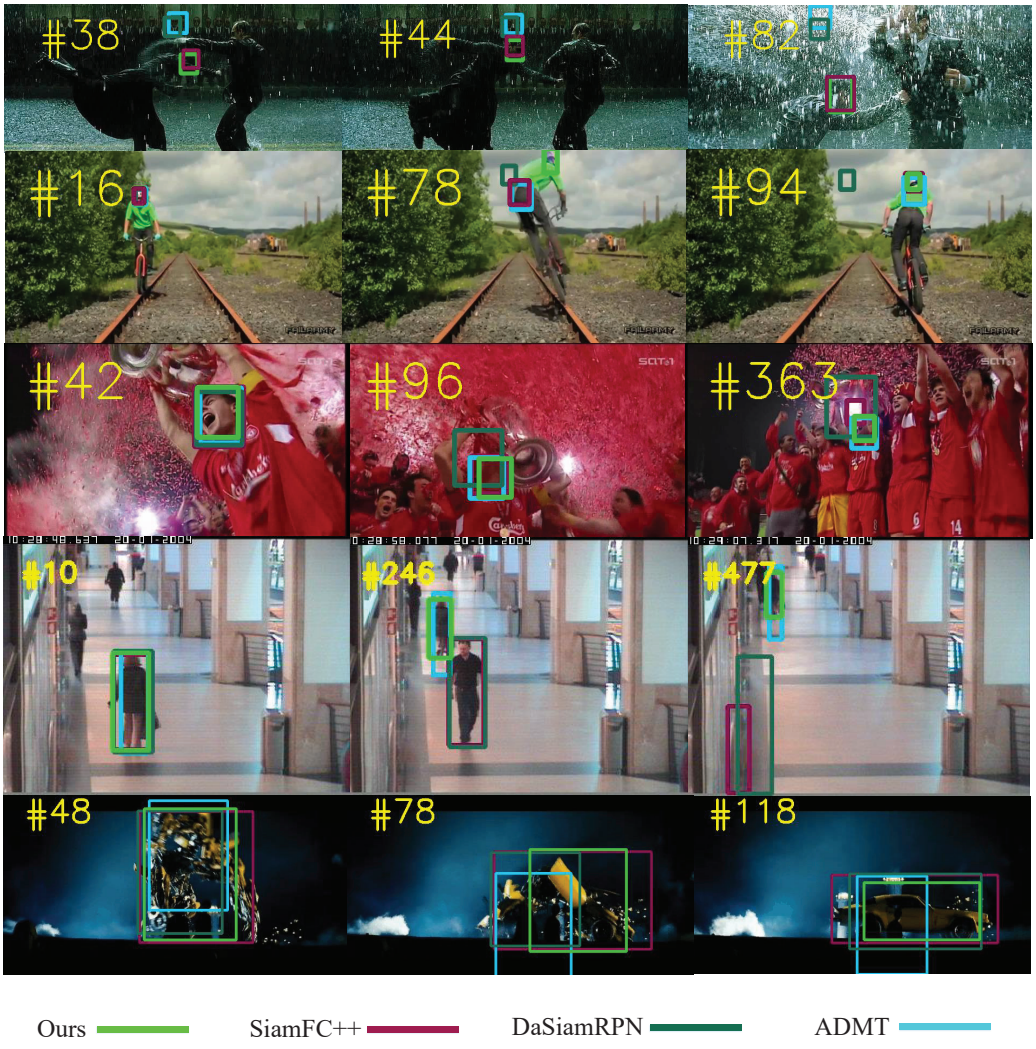

Figure 9. The qualitative results of our approach and other trackers on four challenging real-world scenarios (from top to bottom: Matrix, Biker, Soccer, Walking2, and Trans).

4.9. Ablation Study

In this section, we perform an ablation analysis of the spatial-semantic attention model and flexible spatiotemporal constraint strategy as well as the adaptive weight template-updating model. To visually illustrate our proposed components' effectiveness, we analyzed our algorithm on the OTB100 dataset using one-pass evaluation. The baseline method adopted the original deep features of Conv3, Conv4, and Conv5 from ResNet. The precision (DP) and success rate (OP) are shown in Table 6. Siambase denotes the basic

tracker used by the algorithm. SiamSA denotes the addition of the spatial-semantic-aware attention model. SiamSAST denotes the tracker for the combination of the spatial-semantic-aware attention model and the flexible spatiotemporal constraint strategy. SiamSDP denotes the final tracker with the combination of the three components. In Table 5, it can be seen that both the DP and OP gradually improved after adding each of the three components to the basic tracker. This shows that our spatial-semantic-aware attention model can effectively enhance the sensitive features of the target and improve the ability of the tracker to distinguish the target and background information. Thanks to the introduction of the flexible spatiotemporal constraint strategy, the tracker can react in case of tracking drift. Therefore, the DP plot using SiamSAST was higher than that for the basic method. After adding the adaptive weight template update, the accuracy and success rate of SiamSDP increased by 0.33% and 0.29%, respectively, compared with SiamSAST, indicating that the adaptive weight template update can adapt to the appearance changes in the target and reduce the degradation of the tracker. In addition, we further tested the tracking speed of the proposed algorithm. Since both our spatial-semantic-aware attention model and adaptive weight template update model contain convolutional neural networks, it can be seen that the number of parameters of our model and the amount of computation rose compared with the base tracker, which led to a decrease in tracking speed. Limited by the resource environment and tracking speed, our method has not been considered for application to real-time tracking.

Table 6. Ablation studies on the OTB100 dataset.

Tracker	DP	OP	Params (M)	FLOPs (M)	FPS
SiamBase	0.894	0.682	53.932	5569.01	11.2
SiamSA	0.897	0.686	54.619	6232.32	8.5
SiamSAST	0.899	0.688	54.619	6232.32	8.4
SiamSDP	0.903	0.692	59.801	6495.10	7.9

5. Conclusions

This paper proposes Siamese tracking with spatial-semantic-aware attention and adaptive template updating to suppress irrelevant information about an object's appearance and reduce model degradation. We used the spatial-semantic-aware attention model to enhance the feature representation ability and improve the tracking performance. The proposed spatial-aware attention module is responsible for highlighting the location of the target, and the semantic-aware attention module focuses on important feature channels. Then, the flexible spatiotemporal constraint strategy was proposed to remove the incorrect penalty of the fixed spatiotemporal constraint strategy on the correct target response in case of tracking failure. Finally, we proposed an adaptive weight template-updating strategy to adapt to changes in target appearance during tracking. It can adaptively generate new reliable templates using the tracking results of each frame. We conducted extensive experiments on several challenging datasets such as OTB100, VOT2016, NFS, UAV123, and TC128 to validate the effectiveness of the proposed method.

In this work, our primary focus was on addressing the challenges associated with target tracking in scenarios involving abrupt motion. While our spatial-semantic-aware attention model improved the tracking accuracy, it is important to note that the global average pooling and convolutional network utilized in the model may result in the loss of certain feature information. Additionally, the increased number of model parameters can lead to a decrease in tracking speed. In future works, we will explore alternative attention mechanisms and consider developing lightweight models to reduce the overall number of model parameters. Furthermore, tracking models trained and tested on specific datasets have limitations in their generalization ability, and our study is no exception. Although the dataset samples used for training of the proposed method are sufficiently varied, the capturing device acquires video sequences under unbalanced illumination, a

certain viewing angle, etc., which may lead to capture bias. Ambiguous definitions of visual semantic facts can also lead to labeling and category bias. The limited nature of the dataset when confronted with new, unseen samples may lead to erroneous conclusions. Studying the differences between existing datasets and debiasing methods to improve the generalization ability of tracking algorithms will be our future research direction.

Author Contributions: Conceptualization, H.Z. (Huanlong Zhang) and P.W.; methodology, H.Z. (Huanlong Zhang) and P.W.; software, J.Z. and P.W.; validation, F.W., H.Z. (Huanlong Zhang), and X.S.; formal analysis, F.W.; investigation, J.Z.; resources, H.Z. (Huanlong Zhang); data curation, H.Z. (Hebin Zhou); writing—original draft preparation, P.W.; writing—review and editing, H.Z. (Huanlong Zhang); visualization, H.Z. (Hebin Zhou); supervision, X.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (grant numbers 62272423, 62072416, 62102373, and 62006213), the Program for Science & Technology Innovation Talents in Universities of Henan Province, China (grant number 21HASTIT028), and the Natural Science Foundation of Henan Province, China (grant number 202300410495).

Data Availability Statement: The data presented in this study are available in the article.

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

References

1. Shirzadeh, M.; Asl, H.J.; Amirkhani, A.; Jalali, A.A. Vision-based control of a quadrotor utilizing artificial neural networks for tracking of moving targets. *Eng. Appl. Artif. Intell.* **2017**, *58*, 34–48. [CrossRef]
2. Fernandez-Sanjurjo, M.; Bosquet, B.; Mucientes, M.; Brea, V.M. Real-time visual detection and tracking system for traffic monitoring. *Eng. Appl. Artif. Intell.* **2019**, *85*, 410–420. [CrossRef]
3. Zhang, J.; Liu, J.; Wang, Z. Convolutional neural network for crowd counting on metro platforms. *Symmetry* **2021**, *13*, 703. [CrossRef]
4. He, Z.; He, H. Unsupervised multi-object detection for video surveillance using memory-based recurrent attention networks. *Symmetry* **2018**, *10*, 375. [CrossRef]
5. Bertinetto, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H. Fully-convolutional siamese networks for object tracking. In Proceedings of the Computer Vision—ECCV 2016 Workshops, Amsterdam, The Netherlands, 8–10 and 15–16 October 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 850–865.
6. Zhang, Z.; Peng, H. Deeper and wider siamese networks for real-time visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4591–4600.
7. Li, B.; Yan, J.; Wu, W.; Zhu, Z.; Hu, X. High performance visual tracking with siamese region proposal network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8971–8980.
8. Chen, Z.; Zhong, B.; Li, G.; Zhang, S.; Ji, R. Siamese box adaptive network for visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 6668–6677.
9. Li, L.; Liu, Y.; Chen, Z. SiamCenter: An Anchor-free Siamese Network for Object Tracking. In Proceedings of the 2020 9th International Conference on Computing and Pattern Recognition, New York, NY, USA, 30 October–1 November 2020; pp. 460–466.
10. Guo, D.; Wang, J.; Cui, Y.; Wang, Z.; Chen, S. SiamCAR: Siamese fully convolutional classification and regression for visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 6269–6277.
11. Xu, Y.; Wang, Z.; Li, Z.; Yuan, Y.; Yu, G. Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12549–12556.
12. Chen, D.; Tang, F.; Dong, W.; Yao, H.; Xu, C. SiamCPN: Visual tracking with the Siamese center-prediction network. *Comput. Vis. Media* **2021**, *7*, 253–265. [CrossRef]
13. Peng, J.; Jiang, Z.; Gu, Y.; Wu, Y.; Wang, Y.; Tai, Y.; Wang, C.; Lin, W. Siamrcr: Reciprocal classification and regression for visual object tracking. *arXiv* **2021**, arXiv:2105.11237.
14. Zhang, J.; Miao, M.; Zhang, H.; Wang, J.; Zhang, J.; Qiu, Z. Siamese reciprocal classification and residual regression for robust object tracking. *Digit. Signal Process.* **2022**, *123*, 103451. [CrossRef]
15. Kaiser, J.; Schafer, R. On the use of the I 0-sinh window for spectrum analysis. *IEEE Trans. Acoust. Speech Signal Process.* **1980**, *28*, 105–107. [CrossRef]
16. Mohanty, N.C. *Signal Processing: Signals, Filtering, and Detection*; Springer: Berlin/Heidelberg, Germany, 2012.
17. Bergen, S.W.; Antoniou, A. Design of ultraspherical window functions with prescribed spectral characteristics. *Eurasip J. Adv. Signal Process.* **2004**, *2004*, 196503. [CrossRef]

18. Guo, Q.; Feng, W.; Zhou, C.; Huang, R.; Wan, L.; Wang, S. Learning dynamic siamese network for visual object tracking. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1763–1771.
19. Held, D.; Thrun, S.; Savarese, S. Learning to track at 100 fps with deep regression networks. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 749–765.
20. Morimitsu, H. Multiple context features in siamese networks for visual object tracking. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8–14 September 2018.
21. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*. [CrossRef]
22. Huang, L.; Zhao, X.; Huang, K. Bridging the gap between detection and tracking: A unified approach. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 3999–4009.
23. Yang, Y.; Jiao, L.; Liu, X.; Liu, F.; Yang, S.; Li, L.; Chen, P.; Li, X.; Huang, Z. Dual wavelet attention networks for image classification. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *33*, 1899–1910. [CrossRef]
24. Wang, F.; Jiang, M.; Qian, C.; Yang, S.; Li, C.; Zhang, H.; Wang, X.; Tang, X. Residual attention network for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3156–3164.
25. Guo, D.; Shao, Y.; Cui, Y.; Wang, Z.; Zhang, L.; Shen, C. Graph attention tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 9543–9552.
26. Fan, J.; Wu, Y.; Dai, S. Discriminative spatial attention for robust tracking. In Proceedings of the European Conference on Computer Vision; Springer: Berlin/Heidelberg, Germany, 2010; pp. 480–493.
27. Cui, Z.; Xiao, S.; Feng, J.; Yan, S. Recurrently target-attending tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1449–1458.
28. 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 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4854–4863.
29. Rahman, M.M.; Fiaz, M.; Jung, S.K. Efficient visual tracking with stacked channel-spatial attention learning. *IEEE Access* **2020**, *8*, 100857–100869. [CrossRef]
30. 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.
31. Henriques, J.F.; Caseiro, R.; Martins, P.; Batista, J. High-speed tracking with kernelized correlation filters. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *37*, 583–596. [CrossRef] [PubMed]
32. Li, S.; Zhao, S.; Cheng, B.; Chen, J. Noise-aware framework for robust visual tracking. *IEEE Trans. Cybern.* **2020**, *52*, 1179–1192. [CrossRef] [PubMed]
33. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
34. Ma, C.; Huang, J.B.; Yang, X.; Yang, M.H. Hierarchical convolutional features for visual tracking. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3074–3082.
35. Corbetta, M.; Shulman, G.L. Control of goal-directed and stimulus-driven attention in the brain. *Nat. Rev. Neurosci.* **2002**, *3*, 201–215. [CrossRef] [PubMed]
36. Hou, Q.; Zhou, D.; Feng, J. Coordinate attention for efficient mobile network design. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 13713–13722.
37. Wu, Y.; Lim, J.; Yang, M.H. Object Tracking Benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1834–1848. [CrossRef] [PubMed]
38. Mueller, M.; Smith, N.; Ghanem, B. A benchmark and simulator for uav tracking. In Proceedings of the European Conference on Computer Vision; Springer: Berlin/Heidelberg, Germany, 2016; pp. 445–461.
39. Kiani Galoogahi, H.; Fagg, A.; Huang, C.; Ramanan, D.; Lucey, S. Need for speed: A benchmark for higher frame rate object tracking. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1125–1134.
40. Kristan, M.; Matas, J.; Leonardis, A.; Felsberg, M.; Cehovin, L.; Fernandez, G.; Vojir, T.; Hager, G.; Nebehay, G.; Pflugfelder, R. The visual object tracking vot2016 challenge results. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Las Vegas, NV, USA, 27–30 June 2016; pp. 1–23.
41. Liang, P.; Blasch, E.; Ling, H. Encoding color information for visual tracking: Algorithms and benchmark. *IEEE Trans. Image Process.* **2015**, *24*, 5630–5644. [CrossRef]
42. Zhu, Z.; Wang, Q.; Li, B.; Wu, W.; Yan, J.; Hu, W. Distractor-aware siamese networks for visual object tracking. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 101–117.
43. Li, P.; Chen, B.; Ouyang, W.; Wang, D.; Yang, X.; Lu, H. Gradnet: Gradient-guided network for visual object tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6162–6171.

44. Danelljan, M.; Hager, G.; Shahbaz Khan, F.; Felsberg, M. Convolutional features for correlation filter based visual tracking. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Santiago, Chile, 7–13 December 2015; pp. 58–66.
45. Danelljan, M.; Hager, G.; Shahbaz Khan, F.; Felsberg, M. Learning spatially regularized correlation filters for visual tracking. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 4310–4318.
46. Danelljan, M.; Häger, G.; Khan, F.S.; Felsberg, M. Discriminative scale space tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1561–1575. [CrossRef]
47. Danelljan, M.; Bhat, G.; Khan, F.S.; Felsberg, M. Atom: Accurate tracking by overlap maximization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4660–4669.
48. 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 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4282–4291.
49. Cao, Z.; Huang, Z.; Pan, L.; Zhang, S.; Liu, Z.; Fu, C. Towards Real-World Visual Tracking with Temporal Contexts. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 15834–15849. [CrossRef]
50. Bao, J.; Chen, K.; Sun, X.; Zhao, L.; Diao, W.; Yan, M. SiamTHN: Siamese Target Highlight Network for Visual Tracking. *IEEE Trans. Circuits Syst. Video Technol.* **2023**. [CrossRef]
51. Ni, X.; Yuan, L.; Lv, K. Efficient Single-Object Tracker Based on Local-Global Feature Fusion. *IEEE Trans. Circuits Syst. Video Technol.* **2023**. [CrossRef]
52. Li, Y.; Zhu, J. A scale adaptive kernel correlation filter tracker with feature integration. In Proceedings of the European Conference on Computer Vision; Springer: Berlin/Heidelberg, Germany, 2014; pp. 254–265.
53. Nam, H.; Han, B. Learning multi-domain convolutional neural networks for visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4293–4302.
54. Danelljan, M.; Bhat, G.; Shahbaz Khan, F.; Felsberg, M. Eco: Efficient convolution operators for tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6638–6646.
55. Danelljan, M.; Robinson, A.; Shahbaz Khan, F.; Felsberg, M. Beyond correlation filters: Learning continuous convolution operators for visual tracking. In Proceedings of the European Conference on Computer Vision; Springer: Berlin/Heidelberg, Germany, 2016; pp. 472–488.
56. 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 (ECCV), Munich, Germany, 8–14 September 2018; pp. 483–498.
57. Cheng, S.; Zhong, B.; Li, G.; Liu, X.; Tang, Z.; Li, X.; Wang, J. Learning to filter: Siamese relation network for robust tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 4421–4431.
58. Lukezic, A.; Matas, J.; Kristan, M. D3s-a discriminative single shot segmentation tracker. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 7133–7142.
59. Yang, T.; Xu, P.; Hu, R.; Chai, H.; Chan, A.B. ROAM: Recurrently optimizing tracking model. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 6718–6727.
60. Hu, Q.; Zhou, L.; Wang, X.; Mao, Y.; Zhang, J.; Ye, Q. Spstracker: Sub-peak suppression of response map for robust object tracking. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Voluem 34, pp. 10989–10996.
61. Fan, N.; Liu, Q.; Li, X.; Zhou, Z.; He, Z. Siamese residual network for efficient visual tracking. *Inf. Sci.* **2023**, *624*, 606–623. [CrossRef]
62. Zhang, H.; Liang, J.; Zhang, J.; Zhang, T.; Lin, Y.; Wang, Y. Attention-Driven Memory Network for Online Visual Tracking. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**. [CrossRef]
63. Zhang, J.; Ma, S.; Sclaroff, S. MEEM: Robust tracking via multiple experts using entropy minimization. In Proceedings of the Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014; Proceedings, Part VI 13; Springer: Berlin/Heidelberg, Germany, 2014; pp. 188–203.
64. 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.* **2015**, *38*, 2096–2109. [CrossRef]

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.