symmetry

MDPI

Article

Cross-Correlation Fusion Graph Convolution-Based Object Tracking

Liuyi Fan [1,†], Wei Chen [2] and Xiaoyan Jiang [1,*,†]

[1] School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China
[2] School of Electrical Engineering & Automation, Jiangsu Normal University, Xuzhou 221116, China
* Correspondence: xiaoyan.jiang@sues.edu.cn
† These authors contributed equally to this work.

Abstract: Most popular graph attention networks treat pixels of a feature map as individual nodes, which makes the feature embedding extracted by the graph convolution lack the integrity of the object. Moreover, matching between a template graph and a search graph using only part-level information usually causes tracking errors, especially in occlusion and similarity situations. To address these problems, we propose a novel end-to-end graph attention tracking framework that has high symmetry, combining traditional cross-correlation operations directly. By utilizing cross-correlation operations, we effectively compensate for the dispersion of graph nodes and enhance the representation of features. Additionally, our graph attention fusion model performs both part-to-part matching and global matching, allowing for more accurate information embedding in the template and search regions. Furthermore, we optimize the information embedding between the template and search branches to achieve better single-object tracking results, particularly in occlusion and similarity scenarios. The flexibility of graph nodes and the comprehensiveness of information embedding have brought significant performance improvements in our framework. Extensive experiments on three challenging public datasets (LaSOT, GOT-10k, and VOT2016) show that our tracker outperforms other state-of-the-art trackers.

Keywords: symmetry; single-object tracking; graph attention network; Siamese networks; cross-correlation; feature fusion

check for updates

Citation: Fan, L.Y.; Chen, W.; Jiang, X.Y. Cross-Correlation Fusion Graph Convolution-Based Object Tracking. *Symmetry* **2023**, *15*, 771. https://doi.org/10.3390/sym15030771

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

Received: 25 February 2023
Revised: 16 March 2023
Accepted: 20 March 2023
Published: 21 March 2023

1. Introduction

Visual object tracking is a fundamental and challenging task in computer vision. It has wide applications in virtual reality, human-–machine interactions, image understanding, unmanned vehicles, pedestrian identification [1], object re-identification [2], and other fields. The basic task of single-object tracking is to continuously track and locate the target in each subsequent frame, which is only given the initial position of the target, without providing prior conditions about the color, shape, and size of the target in the process. This means trackers can only track by learning the target in the first frame. In general, difficulties in the tracking process mainly include: occlusion, disappearance, deformation, motion blur, and complex backgrounds. In recent years, mainstream trackers [3–9] based on Siamese networks, which have a symmetrical structure, have achieved an excellent balance between tracking accuracy and efficiency. The template features and search region features are extracted by deep networks, and a cross-correlation-based layer for information embedding is used to obtain similarity maps or response maps, which use the whole template as a convolution kernel.

In traditional Siamese trackers, the information embedding between the template and search region contains global information, in which the information transmitted from the template to the search region is limited and inadequate. Aiming at solving these issues, graph convolution networks [10–12] are widely used for object tracking. The scalability and flexibility of graph nodes achieve a part-to-part information embedding and obtain better response information. These trackers [13,14] treat pixels in feature maps as nodes and leverage graph convolution networks for information transmission. This part-to-part similarity matching can greatly alleviate the effects of deformation and occlusion. Despite their great success, the pixel-level matching-based graph convolution trackers [14] still have some drawbacks.

Graph attention networks that use part-to-part similarity information focus on local matching. These trackers ignore the integrity of nodes in objects. Within an object, the distribution between nodes is correlated and integrated. However, treating pixels in feature maps as independent nodes means that each node is a part of an object. This operation separates the global information about an object, making the object information too scattered. During the actual single object tracking, there will be a lot of interference in the environment, which is similar to the local part of the object. As Figure 1 shows, it easily makes the response to the interference environment stronger, causing tracking errors. Furthermore, the use of only local information embedding can't make full use of the information in search images and template images. How to make the node have both local information and global information is the key to tracking.

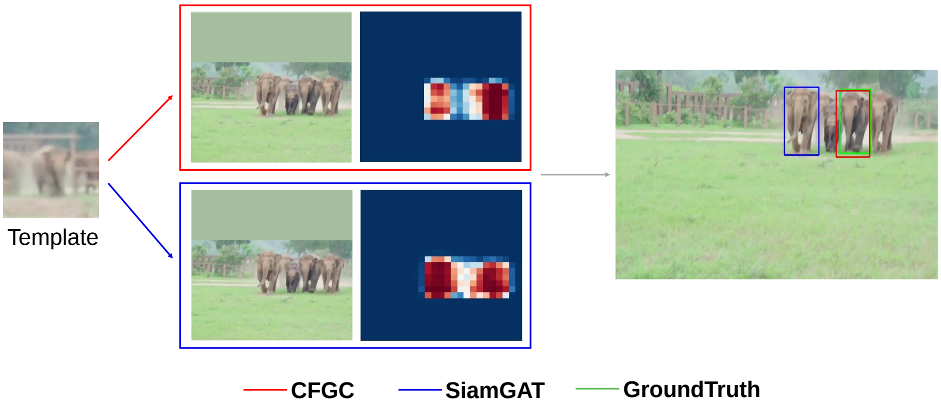

Figure 1. of our CFGC with SiamGAT on the challenging sequences from LaSOT. The baseline SiamGAT (in blue) only uses local information, while our CFGC (in red) has both local and global information in nodes. Based on the same template image and search image, our CFGC obtains a more accurate response map and tracking results. Please zoom in for a better view.

Aiming at solving these issues, we combine the graph convolution features of nodes with the cross-correlation features between the template and search regions. We obtain the template feature map and search region feature map through the template branch and search branch in Siamese network-based architectures, which have symmetry. Then, we use the whole template as a convolution kernel for a global information propagating to obtain the response map. Furthermore, we leverage graph attention networks to learn the part-level relations, and the final node information is fused with the response map as a new response map. Therefore, the information embedding between the template and search region has both global information and part-level information, combining part features and global features. It greatly alleviates the problems of similar interference, occlusion, and deformation. With the graph attention fusion module (GAF), we propose a graph attention network-based object tracking method with cross-correlation feature fusion.

In this work, we focus on improving the performance of traditional graph attention trackers for occlusion and similar interference problems. We propose a novel cross-correlation fusion graph convolution-based object tracking (CFGC) network with GAF for general object tracking. The framework is simple yet effective, combining the graph convolution network with the cross-correlation operation directly. Instead of only local information embedding, our CFGC improves node information, enabling nodes to have both global information and part-level information. Local information embedding improves the accuracy of tracking boxes, and global information embedding weakens the interference of complex environments. These optimized information embeddings make the proposed CFGC more accurate and robust.

We evaluate our tracker on several challenge benchmarks, including the VOT2016 [15], GOT-10k [16], and LaSOT [17] datasets. Our proposed model outperforms state-of-the-art trackers. Our main contributions are as follows:

- We propose an end-to-end graph convolutional tracking framework combining traditional cross-correlation operations which has a symmetrical structure. To the best of our knowledge, this is the first work to combine them directly without complex operations or other strategies.
- We propose a graph attention fusion (GAF) method to realize both part-to-part matching and global matching for information embedding. Compared with traditional graph attention trackers, which ignore the integrity of the object, our tracker can greatly improve the anti-interference and accuracy.
- Extensive experiments on several challenging benchmarks, including the VOT2016, GOT-10k, and LaSOT datasets, show that our proposed model achieves leading performance compared to state-of-the-art trackers, which means our CFGC is accurate and robust.

2. Related Work

2.1. Object Tracking and Siamese Networks

In recent years, the object tracker based on Siamese networks has received extensive attention for its robust tracking performance, which transforms the target tracking problem into an image block matching problem and calculates the similarity of the template and the search region by training a similarity function, thus determining the location of the target. This end-to-end offline training method makes the tracking problem greatly simplified. However, it still has a good balance between tracking accuracy and efficiency. Siamese network-based architectures have a template branch and search branch, and the information embedding between them to obtain informative response maps is the key to accurate object localization.

The method SiamFC [3] constructs two symmetrical branches with shared weights for feature extraction. It feeds the generated feature map into the cross-correlation layer to generate a heatmap or response map, which takes the template features as kernels to perform a convolution operation for the first time. However, it has a problem in that the tracking box is not flexible enough. SiamRPN [4] proposes a Siamese region proposal network (RPN). It extracts candidate regions on the cross-correlation feature map, and then encodes the target appearance information on the template branch into the RPN feature to identify the foreground and background. During tracking, it treats this task as a one-shot detection task. SiamRPN makes the tracking box more accurate and saves time spent on multi-scale testing. Based on SiamRPN, DaSiamRPN [5] analyzes the features extracted by the existing Siamese network methods and their shortcomings, and then focuses on training distractor-aware Siamese networks for accurate and long-term tracking, mainly improving SiamRPN in terms of dataset expansion, training methods, and local-to-global search strategies. To make better deep networks, SiamRPN++ [6] proposes a simple and effective sampling method that breaks the spatial limitation of the deep network in depth and uses multi-layer fusion and depth-wise cross-correlation to optimize performance. However, these anchor-based trackers are sensitive to the number, scale, and aspect ratio of anchors.

To eliminate the limitations of anchors, SiamFC++ [7] adds positional regression and a quality score to SiamFC while using a variety of losses combined with training. SiamBAN [8] classifies the foreground and background by fully convolutional networks and returns to the target box. It uses multi-level information to extract the last three layers of features for feature fusion. SiamCAR [9] proposes a new tracking head, which has a centrality branch to improve tracking accuracy. However, the regression branch and the classification branch are updated independently of optimization.

To resolve the classification and regression mismatches, SiamRN [18] proposes a relation detector and a refinement module to remove interference and integrate outputs. SiamRCR [19] proposes to establish a bidirectional connection between classification and regression, which can dynamically reweight the loss of each positive sample. RBO [20] also proposes a sorting-based optimization method to solve this problem, which uses the classification loss and the IoU-guided loss as optimization constraints. Furthermore, ATOM [21] and DiMP [22], which add the update algorithm, obtain excellent performance, but their tracking speed is slow. However, Siamese networks based on cross-correlation operations cause the information embedding between the template and search region to become a global information propagating process, ignoring part-level information. Therefore, the tracker performance is negatively affected, especially if objects are deformed or occluded.

2.2. Graph Neural Networks

With the development of graph convolution networks [10–12], using the flexibility and scalability of graph nodes to embed information has also become popular in research. Graph neural networks (GNNs) [10] use neural networks in graph structures. For each node in a positional graph, the position of the neighbor can be implicitly used for storing useful information. It includes two parts: propagation and output. The propagation part combines the information of neighbor nodes and edges to obtain the state vector of the current node. The output section converts the characteristics and state vectors of the node into output vectors. To apply convolution operations to graph structures, gGraph convolutional networks (GCNs) [11] take the convolution of a node as a weighted sum of neighbors around it. Through multi-layer GCN convolutions, we can extract the information needed by each node for various classifications or analyses. However, GCNs cannot handle dynamic graphs. Furthermore, GCNs rely on specific graph structures during training, are tested on the same graph, and cannot assign different weights to each neighbor. To address this issue, graph attention networks (GATs) [12] adopt an attention mechanism, which can assign different weights to different nodes, and rely on paired neighboring nodes instead of a specific network structure when training.

Due to the flexibility of node and network structures, graph convolutional networks are widely used for object tracking [23–25] to obtain information embedding. GCT [13] trains GCNs for visual tracking for the first time. It proposes an end-to-end graph convolution tracking (GCT) method based on the Siamese framework, which can consider the time and space target information of the historical frame at the same time. SiamGAT [14] uses GATs to propose a graph attention module that establishes the part-to-part correspondence between the Siamese branches. It enables each part of the search area to aggregate the target's information. Furthermore, the algorithm proposes a target-aware template region that can adapt to different object scales and aspect ratios. However, this part-level information matching ignores the integrity of the object and associativity of object nodes, which causes a high rate of incorrect responses when the environment is similar to the parts of the object.

3. Method

As the appearance of the target continues to change and the surrounding environment becomes more complex, the traditional graph attention networks or Siamese networks that simply rely on local feature matching or global feature matching can no longer meet tracking needs. The goal of our method is to learn a more robust and accurate tracker which combines the cross-correlation operation with a graph attention network for the first time. In this section, we will describe in detail a new graph attention convolution module based on cross-correlation feature fusion. An overview of our framework is illustrated in Figure 2.

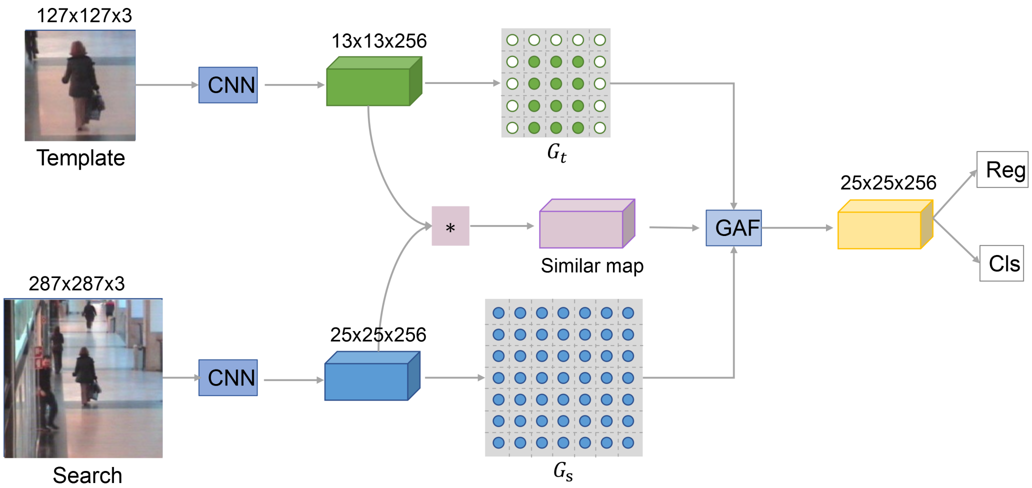

Figure 2. Overview of our proposed method. The architecture of the proposed CFGC, which consists of three core modules: a Siamese network, a cross-correlation layer, and a graph attention fusion model.

3.1. Cross-Graph Attention Convolution

Traditional graph attention networks are based on nodes and their neighbors, generally, in the same graph. The cross-graph convolution idea is used in [26], which calculates the similarity of any two node vectors between the two graphs to obtain the information interaction characteristics, but it depends on the specific graph structure during training, and the test should also be carried out on the same graph, which makes it unsuitable for the tracking of flexible targets. To optimize information embedding between the template and search region, we use cross-graph node embedding to aggregate cross-graph features.

Figure 3 shows the attention mechanism of graph attention networks. The input to it is a set of node features, $h = \left\{\vec{h}_1, \vec{h}_2, \cdots, \vec{h}_N\right\}, \vec{h}_i \in \mathbb{R}^F$, where N is the number of nodes, and F is the number of features in each node. One or more learnable linear transformations are required to adaptively learn a better representation between the nodes. Here, $W \in \mathbb{R}^{F' \times F}$, where F' is the number of dimensions to which the node feature needs to be transformed. Then, self-attention is performed on the node.

$$e_{ij} = a\left(W\vec{h}_i, W\vec{h}_j\right), \tag{1}$$

where, e_{ij} indicates the importance of the features of node j to node i. a is an attention-sharing mechanism, $a \in \mathbb{R}^{F' \times F'} \to \mathbb{R}$.

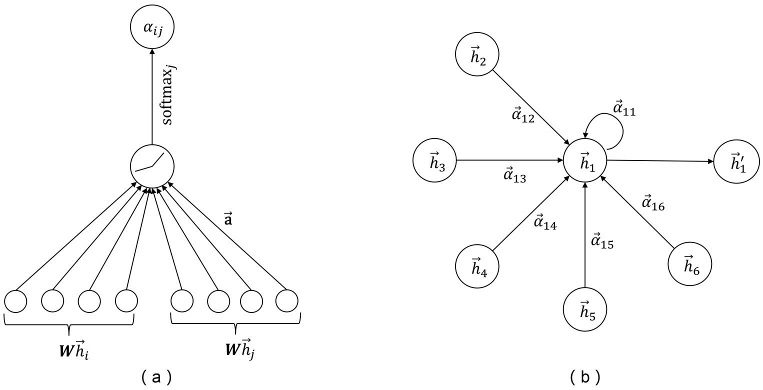

Figure 3. (**a**) is graph attention part. (**b**) is an attention by node 1 on its neighborhood. The aggregated features are $\vec{h}_1'$.

For each j corresponding to node i, graph attention networks use the softmax function to normalize the importance of all of j's feature nodes to node i and obtain the normalized attention coefficients α_{ij}.

$$\alpha_{ij} = softmax(e_{ij}) = \frac{\exp(e_{ij})}{\sum\limits_{k \in N_i} \exp(e_{ij})} . \tag{2}$$

Then, the normalized attention coefficients are used to compute a linear combination of the features corresponding to them to serve as the final output features for every node $\vec{h}'_i$.

$$\vec{h}'_i = \sigma\left(\sum_{j \in N_i} \alpha_{ij} W \vec{h}_j\right) , \tag{3}$$

where σ is nonlinearity.

Finally, a powerful feature representation with information transferring and interactions, $\vec{h}'_i$, will be obtained, which is shown in Figure 4.

However, this traditional graph attention network cannot be used directly for object tracking because the input is the node and its neighbor in the same graph. If we directly use image pixels as nodes, this leads to increased network costs and information redundancy. However, this idea of attention mechanisms remains key to information embedding.

We use cross-graph attention convolution to aggregate features. Its architecture is shown in Figure 4. We propose a graph attention convolution module based on cross-correlation feature fusion. This module integrates the features of the cross-correlation operation between the template feature map and the search feature map with the features produced by cross-graph attention features, so that the template map has both local information and global information for the information embedding of the search map. It is more robust to similar objects and complex environmental interference, and this powerful representation of information can greatly optimize tracker performance.

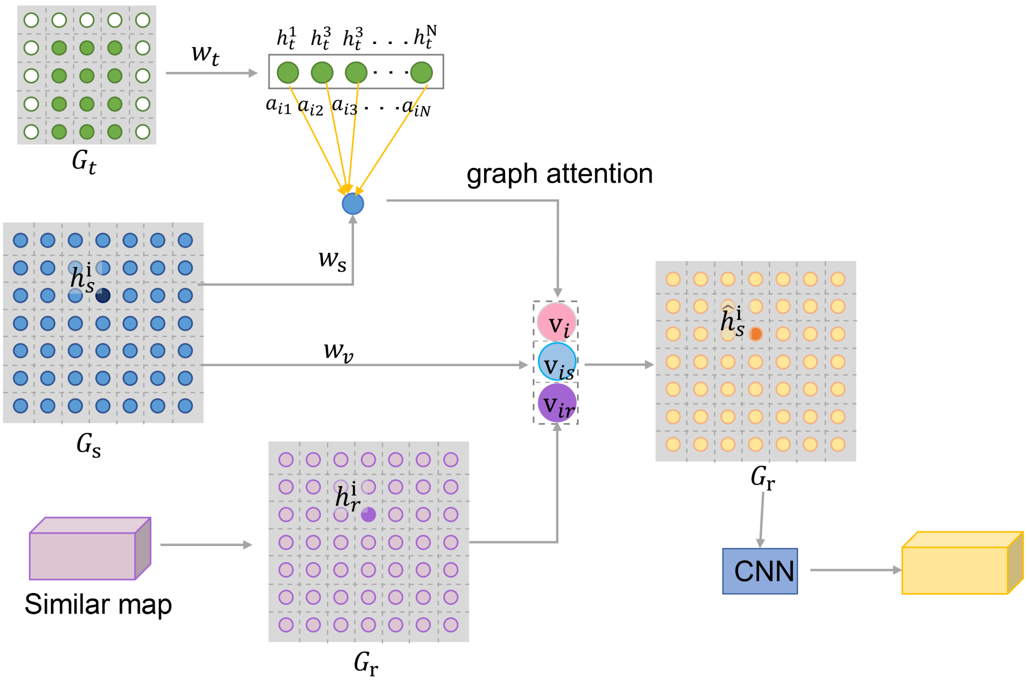

Figure 4. The architecture of GAF model, which combines graph attention network with cross-correlation feature. The representation of each search node is reconstructed by template target nodes with attention mechanism. Then, it fuses the cross-correlation similarity map. The final new node is fed into the tracking head for target location prediction.

Given two images of template patch T and search S, two feature maps Ft and Fs are first obtained using the Siamese feature extraction network. We consider each 1 × 1 × c grid of the feature map as a node (part), where c represents the number of feature channels. Next, the boundless weightless graph is generated.

Based on the graph, we carry out cross-graph attention convolution operations on nodes in the template and search regions. V_t is the node set of G_t, and V_s is the node set of G_s. h_t^j is the j node of the template feature map, and h_s^i is the i node of the search feature map.

The input features are transformed into higher-level features through two learnable linear transformations, w_s and w_t , $w_s, w_t \in \mathbb{R}^{256\times256}$. Self-attention is then performed on the node, and we simply use the inner product between features as a measure of similarity. e_{ij} shows the importance of the features of node j to node i.

$$e_{ij} = f\left(w_s h_s^i, w_t h_t^j\right) . \tag{4}$$

$$f\left(h_s^i, h_t^j\right) = \left(w_s h_s^i\right)^T \left(w_t h_t^j\right) . \tag{5}$$

Then, we obtain the attention coefficients a_{ij} according to Equation (2), which measures how much attention or weight the tracker should pay to part j in the template graph based on the i nodes in the search graph.

Information interaction between template nodes and search region nodes is carried out according to the attention coefficient. v_i is an aggregated representation of node i that uses the attention weight of all nodes in G_t to the i node in G_s. With the learned attention coefficient, each search node can effectively aggregate the target information from the template. We use Equation (3) to obtain v_i.

Finally, through cross-graph attention convolution, the information part-to-part information propagation of template graph and search graph node features is effectively realized. We obtain v_i as Figure 4 shows. The representation of each search node v_i is reconstructed by aggregating information and attention mechanisms from all neighboring target nodes.

Using the cross-graph attention convolution, the part-to-part correspondence is established, and each part of the search region aggregates the target information so as to obtain a very effective information embedding mapping. v_i will be concatenated with the cross-correlation feature as a part of feature fusion.

3.2. Cross-Correlation Features and Feature Fusion

Due to the complex environment, simply relying on part-level features can no longer meet tracking needs. Although each part of the search region aggregates the target information in the cross-graph attention convolution, it has the disadvantage of ignoring the overall information of the object, causing tracking errors in occlusion and environmental interference.

Inspired by the cross-correlation layer, we use the whole template as a convolution kernel to make cross-correlation operations in search region for each channel, which evaluates and calculates the similarity of all translational subwindows (candidate regions) on the dense mesh. Additionally, it obtains the matching degree of each position in a large search area with a target in the template. This information embedding based on the cross-correlation layer realizes the global information propagation between the template and the search region. We use this global information to improve the limitations of part-level information in graph attention networks.

Given the two images of template patch T and search S, we use the cross-correlation layer to obtain a similar map that has a global information embedding.

This similar map is upsampled to the same shape as G_s. Then, each pixel of the graph is treated as a node, and the node information is the feature information of the multi-channel of the pixel. G_r is a boundless and weightless graph.

$$v_{ir} = h_r^i \,, \tag{6}$$

where v_{ir} contains the information interaction between the corresponding area of the node and the whole target.

To obtain deeper information, we apply linear transformations to the node features.

$$v_{is} = w_v h_s^i \,, \tag{7}$$

where v_{is} is the linear transformations in the search region.

The fusion of local feature information and global feature information is the key to our tracker. The cross-graph attention convolution features are fused with node features of search features and cross-correlation features to obtain a more powerful feature representation, $\hat{h}_s^i$.

$$\hat{h}_s^i = ReLu(v_i||v_{is}||v_{ir}) \,. \tag{8}$$

This feature fusion operation realizes the local information embedding and global information embedding of the template graph for the search graph, which greatly improves the tracking performance in occlusion and similar interference situations.

The fused feature map, which is the response map obtained by the information finally embedded in the template branch and the search branch, will be fed into the tracking head [9] for target localization. The tracking head contains a classification branch and a regression branch. Both branches share the same response mapping from GAF.

4. Experiments

4.1. Implementation Details

We trained and tested our tracker using RTX-3090 cards with Python version 3.6 and Pytorch version 1.2. We used GoogLeNet [27] as our backbone, which allows for the learning of multiscale feature representations with fewer parameters and faster inference speed. The training process of the tracker involved two stages: in the first stage, we froze the parameters in the backbone network and trained the GAF model and the tracking head network. After several epochs, we moved on to the second stage, where we thawed the parameters in the backbone network and adjusted them.

Our experiments used COCO [28], ImageNet [29], and YouTube-BB [30] for the training in VOT2016 [15]. For the fairness of experiments, we only used the official training and testing sets to evaluate our tracker on the GOT-10k [16] and LaSOT [17] datasets. During training, the template patch size was set to 127 × 127 pixels, and the search region size was 287 × 287 pixels. The interval epoch was set to 10. The template frame was the initial frame of the sequence and was fixed.

4.2. Ablation Study

In this section, we perform extensive analysis of our proposed model on the LaSOT dataset. The AUC score is adopted for evaluation.

Cross-graph attention. We evaluated the importance of the part-level information embedding. Table 1 shows the performance of only using global feature embedding under the same Siamese network architecture. By replacing the cross-correlation operation with graph attention networks, the success is improved by 1.4% from 51.6% to 53.0%. This is because the graph attention networks treat the feature map pixels as independent nodes, which realizes part-to-part information propagating instead of global information propagating using the whole template as a convolution kernel. Learning the part-level relations can better adapt to deformation and occlusion. However, too scattered of matching of information can also lead to tracking errors and poor robustness to similar interference.

Cross-correlation. We also evaluated the importance of cross-correlation operations to improve performance. As can be seen in Table 1, the success performance of this module is improved by 1.6% from 53.0% to 54.6%, and the precision is improved by 2.2% when compared to only a simple graph attention layer. These results demonstrate that the supplement of global information can better enrich node information, which makes trackers more robust to occlusion, deformation, and similar interference. We concatenate the node features together and reduce dimensionality through convolution neural networks to fuse features. However, the balance of global and local information is also key.

Table 1. Ablation analysis of the proposed ranking-based optimization, consisting of cross-graph attention and cross-correlation layer, on LaSOT.

Dataset	Cross-Graph Attention	Cross-Correlation	Success	Precision
LaSOT	✘	✔	0.516	0.524
	✔	✘	0.530	0.524
	✔	✔	0.546	0.546

4.3. Experiments on LaSOT

To evaluate the generalization of our trackers, we used the official LaSOT [17] dataset for training and testing. LaSOT is a high-quality benchmark for large-scale single-object tracking. It contains 1400 video sequences and includes different object collections from 70 categories, each containing the same number of videos. Its video sequence attributes include 14 attributes, including full occlusion, partial occlusion, and deformation. The LaSOT dataset is by far the largest tracking benchmark with high-quality annotations.

We used the official training set and test set to train or test our proposed model on the LaSOT dataset. We evaluated our tracker by the success rate and precision, and the success plot and precision plot are drawn in Figures 5 and 6. Compared to the state-of-the-art trackers, including SiamFC, ATOM, SiamDW [31], SiamRPN++, SiamBAN, SiamCAR, CLNet [32], SiamGAT, USOT [33], SiamRN [18], and RBO [20], our tracker ranks first in both success rate and precision. Our proposed method especially has significant performance improvements in partial occlusion, full occlusion, and out-of-view, which is beneficial to the complementation of the overall cross-correlation feature of the object to the dispersed part-level features. For these factors, we demonstrate improvements of 1.5%, 1.9%, and 3.5% compared to baseline. Compared to the baseline SiamGAT, our model's performance improves by 1.6%, 2.2%, and 1.6% in terms of success, precision, and normalized precision overall.

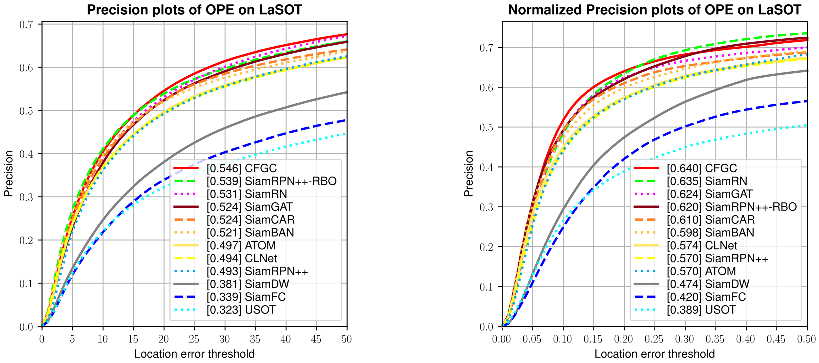

Figure 5. Comparison with the state-of-the-art trackers on the LaSOT dataset in terms of the normalized precision and precision plots of OPE.

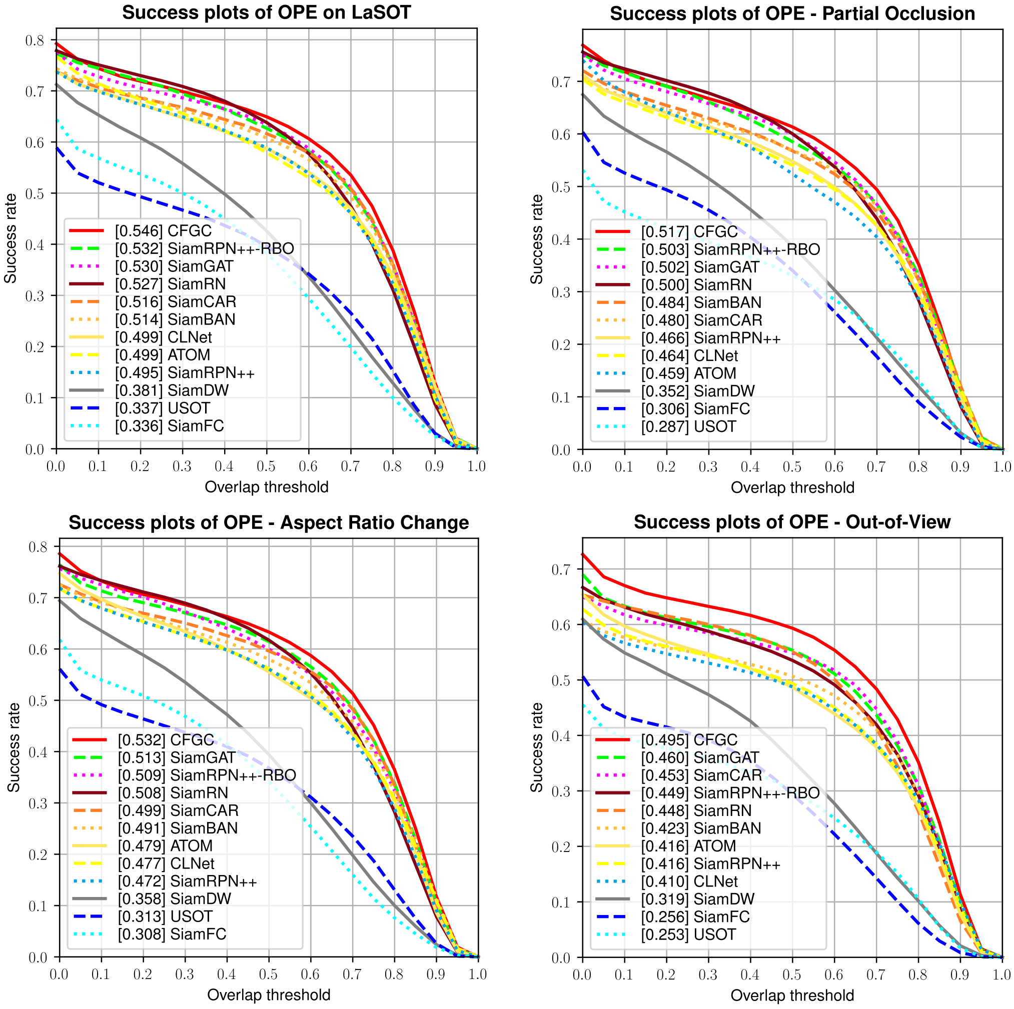

Figure 6. Comparison with the state-of-the-art trackers on the LaSOT dataset in terms of success plot of OPE including all attributes and three individual attributes.

Compared with RBO, which is a recent tracker, our tracker surpasses it by 0.7% in precision and 1.4% in success. The results show that the preposed tracker still has excellent performance in long-term tracking tasks, and the fusion of part-level information and global information is effective.

We also performed a qualitative experimental analysis of the proposed tracker. As shown in Figure 7, the four video sequences are Motorcycle-3, Pool-7, Elephant-12, and Volleyball-18. This sequence group contains complex scenes such as deformation, occlusion, background clutter, motion blur, illumination variation, and so on. Figure 7a,c shows that SiamRPN++, CLNet, and SiamRPN++-RBO have poor tracking when the target is occluded and violently deformed. Figure 7b,d show that SiamGAT has a reduction in tracking performance in an environment that is similar to the part of object. Compared with other trackers, our CFGC is able to produce more accurate target bounding boxes due to the proper integration of part-level and global information between the template and search regions. The part-to-part similarity matching in the graph network improves the accuracy of the tracker, and the extraction of cross-correlation features can weaken the interference of the surrounding environment. Therefore, the proposed CFGC is robust and efficient.

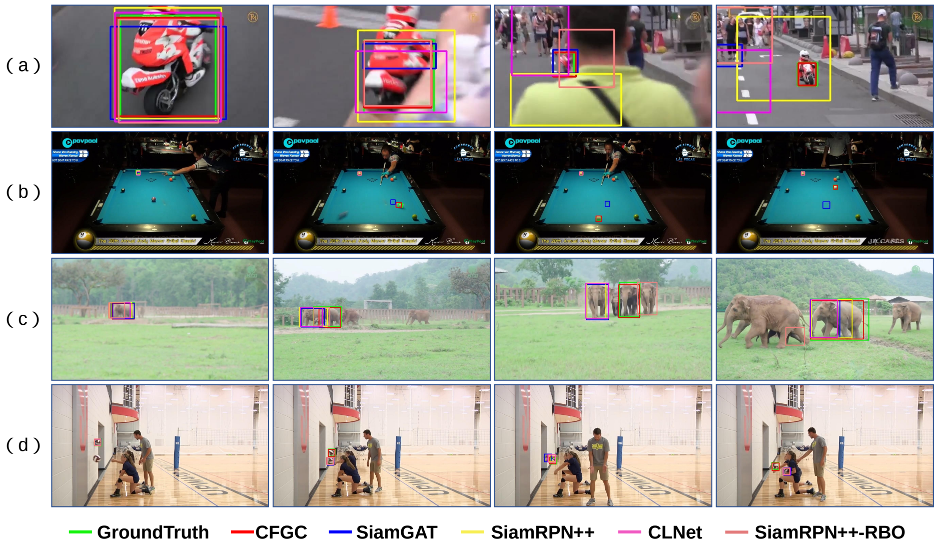

Figure 7. Comparison of our CFGC with state-of-the-art trackers on the four challenging sequences from LaSOT. (**a**): Motorcycle-3, (**b**): Pool-7, (**c**): Elephant-12, (**d**): Volleyball-18. Our tracker is able to handle occlusion, similar interference, and deformation due to the embedding of part-level and global information. As shown in the graph, our tracker (in red) significantly outperforms the baseline SiamGAT (in blue), SiamRPN++ (in yellow), ClNet (in purple), SiamRPN++-RBO (in orange), and GroundTruth (in green). Please zoom in for a better view.

4.4. Evaluation on GOT-10k

We validated our proposed trackers on the GOT-10k dataset [16], which consists of more than 10,000 fully annotated sequences. The training and test classes of the GOT-10k have zero overlap. This avoids the bias of evaluation results towards familiar objects and promotes the generalization of tracker development. Furthermore, GOT-10k offers a wide range of object types and forms of motion, providing a wider coverage of real-world moving objects, and the scale and diversity of data of data significantly increases the reliability of assessments.

We followed the GOT-10k protocol and trained our proposed model using only a subset of its training set. We evaluated our tracker on the GOT-10k dataset and compared it with state-of-the-art trackers, including SiamFC, ATOM, DiMP, SiamFC++, Ocean [34], D3S [35], SiamCAR, SiamGAT, PACNet [36], SiamRCR [19], SAOT [37], LightTrack [38], STARK [39], ViTCRT [40], and RBO. We selected the widely used average overlap (AO) and success rate (SR) as evaluation metrics. As shown in Table 2, our proposed tracker performs well, and compared to the baseline SiamGAT, its performance improves by 1.3%, 1.2% and 1.9% in terms of AO, $SR_{0.5}$, and $SR_{0.75}$, respectively. The results demonstrate the robustness and stability of our tracker. Furthermore, we make a comparison about the speed, flops and params in our CFGC, SiamGAT, and STARK. As Table 3 shows, our CFGC has a smaller number of parameters than STARK, which means less hardware requirements. In comparison with the baseline SiamGAT, our CFGC improves tracking performance at a small cost. However, the information redundancy remains a challenge.

Table 2. Comparison of our tracker with state-of-the-art trackers on the GOT-10k dataset.

Tracker	AO (%)	$SR_{0.5}$(%)	$SR_{0.75}$(%)
ATOM	55.6	63.4	40.2
SiamCAR	56.9	67	41.5
PACNet	58.2	68.5	44.3
SiamFC++	59.5	69.5	47.9
D3S	59.7	67.6	47.2
SiamRPN++-RBO	60.2	71.8	44.6
Ocean-online	61.1	72.1	47.3
DiMP-50	61.1	71.7	49.2
LightTrack	62.3	72.6	-
SiamRCR	62.3	75.2	46
SiamGAT	62.7	74.4	48.8
STARK	67.2	76.1	61.2
ViTCRT	65.6	75.0	59.8
CFGC	64.0	75.6	50.7

Table 3. Comparison of the speed, FLOPs, and Params.

Tracker	Speed (fps)	FLOPs (G)	Params (M)
STARK	31.7	20.4	47.2
SiamGAT	37.5	19.57	14.23
CFGC	37.03	19.61	14.29

4.5. Evaluation on VOT2016

We also compared our tracker on VOT2016 [15] in Table 4. VOT2016 contains 60 challenging sequences. We used three metrics, accuracy (Acc), robustness ((Rob), and expected average overlap rate (EAO), to report tracking performance. The higher the accuracy and expected average overlap rate, the lower the robustness score and the tracker performance. As Figure 8 shows, our proposed tracker outperforms the state-of-the-art trackers, including SiamDW [31], ASRCF [41], ROAM [42], ULAST [43], USOT [33], S2SiamFC [44], SiamMask-box [45], SiamFP [46], SiamFC-ACM [47], MetaRTT [48], and others. The proposed tracker achieves 0.438 in EAO and 0.646 in Acc, which ranks first. Furthermore, the robustness score ranks third. Benefiting from the rich information embedding of nodes, our tracker achieves a better robustness score than SiamFC-ACM. The results show that our tracker is more accurate and robust. However, the balance between robustness and accuracy remains a challenge.

Table 4. Comparison of our tracker with the state-of-the-art trackers on the VOT2016 dataset.

Tracker	EAO (%) ↑	Acc (%) ↑	Rob (%) ↓
S2SiamFC	21.5	49.3	63.9
SiamFC	23.5	53.2	46.1
SiamFP	33.5	53.7	38.1
SiamRPN	33.7	57.8	31.2
SiamFC-ACM	33.8	53.5	29.4
MetaRTT	34.6	-	-
SiamDW	37.1	58.0	24.0
ROAM	38.4	55.6	13.8
ASRCF	39.1	56.0	18.7
USOT	40.2	60.0	23.3
SiamMASK-box	41.2	62.3	23.3
ULAST	41.7	60.3	21.4
CFGC	43.8	64.6	19.6

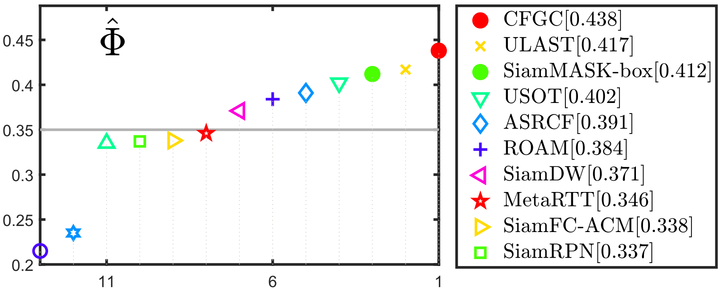

Figure 8. Expected averaged overlap performance on VOT2016.

5. Conclusions

In this paper, we have presented a novel graph attention network-based object tracking with cross-correlation feature fusion, termed CFGC. The symmetry of the template branch and search branch ensures the uniformity and representativeness of feature extraction. Our CFGC combines the graph attention network with a cross-correlation layer. The part-to-part similarity matching in graph networks improves the accuracy of the tracker, and the extraction of cross-correlation features can weaken the interference of the surrounding environment. Instead of only local or global information, the information embedding between the template and search region has both global information and part-level information on GAF, which greatly alleviates the problems of similar interference, occlusion, and deformation. Our CFGC enables more generalizable visual tracking. The proposed tracker outperforms state-of-the-art trackers on the VOT2016, GOT-10k and LaSOT datasets, proving that our tracker has better robustness and accuracy. However, a large amount of information fusion easily leads to redundant information. In the future, we intend to explore feature filtering strategies to improve the information redundancy of local and global feature fusion. Furthermore, a dynamic template update strategy is also a good way to explore.

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

Funding: The work is supported by the following projects: National Natural Science Foundation of China (NSFC), Essential projects, Nr.: U2033218, 61831018.

Data Availability Statement: Not applicable.

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

References

1. Suljagic, H.; Bayraktar, E.; Celebi, N. Similarity based person re-identification for multi-object tracking using deep siamese network. *Neural Comput. Appl.* **2022**, *34*, 18171–18182. [CrossRef]
2. Bayraktar, E.; Wang, Y.; DelBue, A. Fast re-obj: Real-time object re-identification in rigid scenes. *Mach. Vis. Appl.* **2022**, *33*, 97. [CrossRef]
3. Cen, M.; Jung, C. Fully convolutional siamese fusion networks for object tracking. In Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 7–10 October 2018.
4. 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 (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.
5. 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.
6. 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 (CVPR), Long Beach, CA, USA, 15–20 June 2019.

7. 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 (AIII), New York, NY, USA, 10 February 2020.
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 (CVPR), Seattle, WA, USA, 14–19 June 2020.
9. 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 (CVPR), Seattle, WA, USA, 14–19 June 2020.
10. Scarselli, F.; Gori, M.; Tsoi, A.C.; Hagenbuchner, M.; Monfardini, G. The graph neural network model. *IEEE Trans. Neural Net.* **2008**, *20*, 61–80. [CrossRef] [PubMed]
11. Zhang, S.; Tong, H.; Xu, J.; Maciejewski, R. Graph convolutional networks: A comprehensive review. *Comput. Soc. Net.* **2019**, *6*, 1–23. [CrossRef]
12. Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. *Stat* **2018**, *1050*, 4.
13. Gao, J.; Zhang, T.; Xu, C. Graph convolutional tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
14. 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 (CVPR), Nashville, TN, USA, 20–25 June 2021.
15. Kristan, J.M.M.; Leonardis, A.; Felsberg, M. The visual object tracking vot2016 challenge results. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 November 2016.
16. Huang, L.; Zhao, X.; Huang, K. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE Trans. Pattern Anal. Mach. Intell. (Tpami)* **2019**, *43*, 1562–1577. [CrossRef]
17. Fan, H.; Lin, L.; Yang, F.; Chu, P.; Deng, G.; Yu, S.; Bai, H.; Xu, Y.; Liao, C.; Ling, H. Lasot: A high-quality benchmark for large-scale single object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
18. 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 (CVPR), Nashville, TN, USA, 20–25 June 2021.
19. 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. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), Montréal, QC, Canada, 19–20 August 2021.
20. Tang, F.; Ling, Q. Ranking-based siamese visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022.
21. 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 (CVPR), Long Beach, CA, USA, 15–20 June 2019.
22. Bhat, G.; Danelljan, M.; Gool, L.V.; Timofte, R. Learning discriminative model prediction for tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019.
23. Wang, Y.; Kitani, K.; Weng, X. Joint object detection and multi-object tracking with graph neural networks. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021.
24. Cao, P.; Zhu, Z.; Wang, Z.; Zhu, Y.; Niu, Q. Applications of graph convolutional networks in computer vision. *Neural Comput. Appl.* **2022**, *34*, 13387–13405. [CrossRef]
25. Dai, P.; Weng, R.; Choi, W.; Zhang, C.; He, Z.; Ding, W. Learning a proposal classifier for multiple object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021.
26. Wang, R.; Yan, J.; Yang, X. Learning combinatorial embedding networks for deep graph matching. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019.
27. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
28. Lin, T.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the European Conference on Computer Vision (ECCV), Zurich, Switzerland, 6–12 September 2014.
29. Deng, J.; Dong, W.; Socher, R.; Li, L.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami, FL, USA, 20–25 June 2009.
30. Real, E.; Shlens, J.; Mazzocchi, S.; Pan, X.; Vanhoucke, V. Youtube-boundingboxes: A large high-precision human-annotated data set for object detection in video. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–27 July 2017.
31. 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 (CVPR), Long Beach, CA, USA, 15–20 June 2019.
32. Zheng, Z.; Wan, Y.; Zhang, Y.; Xiang, S.; Peng, D.; Zhang, B. Clnet: Cross-layer convolutional neural network for change detection in optical remote sensing imagery. *ISPRS J. Photogramm. Remote. Sens. (JPRS)* **2021**, *175*, 247–267. [CrossRef]
33. Zheng, J.; Ma, C.; Peng, H.; Yang, X. Learning to track objects from unlabeled videos. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021.
34. Zhang, Z.; Peng, H.; Fu, J.; Li, B.; Hu, W. Ocean: Object-aware anchor-free tracking. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020.

35. 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 (CVPR), Seattle, WA, USA, 13–19 June 2020.
36. Ko, K.; Lee, J.-T.; Kim, C.-S. Pac-net: pairwise aesthetic comparison network for image aesthetic assessment. In Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 7–10 October 2018.
37. Zhou, Z.; Pei, W.; Li, X.; Wang, H.; Zheng, F.; He, Z. Saliency-associated object tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021.
38. Yan, B.; Peng, H.; Wu, K.; Wang, D.; Fu, J.; Lu, H. Lighttrack: Finding lightweight neural networks for object tracking via one-shot architecture search. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021.
39. Yan, B.; Peng, H.; Fu, J.; Wang, D.; Lu, H. Learning spatio-temporal transformer for visual tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021.
40. Nardo, E.D.; Ciaramella, A. Tracking vision transformer with class and regression tokens. *Inf. Sci.* **2023**, *619*, 276–287. [CrossRef]
41. Dai, K.; Wang, D.; Lu, H.; Sun, C.; Li, J. Visual tracking via adaptive spatially-regularized correlation filters. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
42. 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 (CVPR), Seattle, WA, USA, 13–19 June 2020.
43. Shen, Q.; Qiao, L.; Guo, J.; Li, P.; Li, X.; Li, B.; Feng, W.; Gan, W.; Wu, W.; Ouyang, W. Unsupervised learning of accurate siamese tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022.
44. Sio, C.H.; Ma, Y.; Shuai, H.; Chen, J.; Cheng, W. S2siamfc: Self-supervised fully convolutional siamese network for visual tracking. In Proceedings of the 28th ACM International Conference on Multimedia, Seattle, WA, USA, 12–16 October 2020.
45. Wang, Q.; Zhang, L.; Bertinetto, L.; Hu, W.; Torr, P.H.S. Fast online object tracking and segmentation: A unifying approach. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
46. Zhao, W.; Deng, M.; Cheng, C.; Zhang, D. Real-time object tracking algorithm based on siamese network. *Appl. Sci.* **2022**, *12*, 7338. [CrossRef]
47. Han, W.; Dong, X.; Khan, F.S.; Shao, L.; Shen, J. Learning to fuse asymmetric feature maps in siamese trackers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021.
48. Jung, I.; You, K.; Noh, H.; Cho, M.; Han, B. Real-time object tracking via meta-learning: Efficient model adaptation and one-shot channel pruning. In Proceedings of the AAAI Conference on Artificial Intelligence (AIII), New York, NY, USA, 7–12 February 2020; pp. 11205–11212.

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.