electronics

MDPI

Article

A Learning Frequency-Aware Feature Siamese Network for Real-Time Visual Tracking

Yuxiang Yang, Weiwei Xing *, Shunli Zhang, Qi Yu, Xiaoyu Guo and Min Guo

School of Software Engineering, Beijing Jiaotong University, No. 3 Shangyuancun, Haidian District, Beijing 100044, China; 16112088@bjtu.edu.cn (Y.Y.); slzhang@bjtu.edu.cn (S.Z.); 16121736@bjtu.edu.cn (Q.Y.); guoxiaoyu@bjtu.edu.cn (X.G.); minguo@bjtu.edu.cn (M.G.)

* Correspondence: wwxing@bjtu.edu.cn

Received: 16 April 2020; Accepted: 19 May 2020; Published: 21 May 2020

Abstract: Visual object tracking by Siamese networks has achieved favorable performance in accuracy and speed. However, the features used in Siamese networks have spatially redundant information, which increases computation and limits the discriminative ability of Siamese networks. Addressing this issue, we present a novel frequency-aware feature (FAF) method for robust visual object tracking in complex scenes. Unlike previous works, which select features from different channels or layers, the proposed method factorizes the feature map into multi-frequency and reduces the low-frequency information that is spatially redundant. By reducing the low-frequency map's resolution, the computation is saved and the receptive field of the layer is also increased to obtain more discriminative information. To further improve the performance of the FAF, we design an innovative data-independent augmentation for object tracking to improve the discriminative ability of tracker, which enhanced linear representation among training samples by convex combinations of the images and tags. Finally, a joint judgment strategy is proposed to adjust the bounding box result that combines intersection-over-union (IoU) and classification scores to improve tracking accuracy. Extensive experiments on 5 challenging benchmarks demonstrate that our FAF method performs favorably against SOTA tracking methods while running around 45 frames per second.

Keywords: deep learning; computer version; object tracking

1. Introduction

In recent years, visual object tracking as a fundamental problem in the computer vision field has been widely studied and applied to the unmanned vehicle, traffic surveillance, and intelligent transportation. As a middle-level semantic problem, object tracking further extracts and process low-level semantic features (such as image classification) to provide reliable target location and tracking information for high-level semantic problems (such as action recognition). The tracker can analyze manually or automatically selected target in a video sequence and effectively predict the position and corresponding status of the current tracking target. However, the tracking targets have changed from traditional vehicles, pedestrians, and other large objects to random, small objects in complex scenes (such as background clutter, illumination variation, scale variation, low resolution, occlusion, and fast motion), which are harder to predict. To address this issue, strong discriminative deep learning models have been introduced to design robust and real-time tracking methods in complex scenes.

Existing deep learning-based trackers can obtain robust tracking results for deep models [1–4] have strong foreground and background discrimination ability by learning knowledge with massive parameters. However, targets are changing during the tracking process and these models perform heavy calculations to adapt to the current target, which limits the tracking speed and cannot meet the

requirements of real-time tracking. Some methods try to solve this problem by choosing lightweight models [5], but those methods usually improve tracking speed by sacrificing tracking accuracy.

The robust deep learning-based trackers use huge labeled training samples to train models. The discriminative ability of the model will be stronger with a larger training dataset. Expansion of training set requires additional manual annotation, thus some methods [6,7] try to general new training samples by the geometric transformation of original samples, which can improve the discriminative ability of the model. However, those data augmentation methods assume that samples share the same class vicinity without considering the vicinity relation across different classes, which will limit its improvement. Furthermore, classification or regression-based methods use the highest predicted score as the object position and some methods choose the object position with both higher classification and regression scores to improve the tracking accuracy. However, when classification and regression scores are conflicted, it will reduce tracking robustness and cause tracking failure.

To address those issues, a novel robust real-time tracker FAF is proposed. Different from existing tracking methods select features by different layers or channels, we innovatively introduce frequency-aware features into object tracking, which can improve the model's discrimination ability while reducing feature calculation. In order to further to improve the model's ability to distinguish between background and target, an effective training method based on data fusion is innovatively designed, which can help the model learns the vicinity relation across different classes. Finally, a joint judgment algorithm combining regression and classification scores is introduced to further improve the accuracy of the tracking model in complex scenes. Extensive experiments are evaluated on 5 famous benchmarks: OTB [8], LaSOT [9], GOT10K [10], TrackingNet [11], and VOT18 [12], which show that our tracker outperforms the state-of-the-art trackers.

The contributions of this paper include:

- This paper proposes a novel robust real-time tracker FAF with combines frequency-aware features. Different from existing tracking methods use linear combinations of shallow and deep layer features for tracking, which need a more complex network. We innovative decompose the layer feature into high-frequency and low-frequency features, then compress the redundant low-frequency features and splice them into multi-frequency features. Without increasing model complexity, the frequency-aware feature reduces feature calculations and improve feature discrimination ability.
- To enhance the ability of tracking models to distinguish between foreground and background, we innovatively design a training data fusion method to enhance the ability of the model to learn vicinity relations across different classes. Both labels and samples are used to perform weighted fusion and obtain fusion samples. By training with fusion samples, blurred boundaries between classes can improve the discriminative ability of the model.
- To improve the tracking bounding box accuracy, a joint judgment strategy combining regression and classification predicted scores are proposed. Compare with existing trackers use independent or linearly combined classification and regression scores, the proposed strategy uses confidence estimation with both predicted scores to improve the tracking accuracy. In particular, we can solve the conflict of classification and regression scores in complex environments and enhance the robustness of the model. To comprehensive verify the efficiency of FAF, extensive experiments are evaluated on 5 famous benchmarks, the results prove that the proposed FAF outperforms the state-of-the-art trackers while running at 45 fps.

2. Related Work

In this section, we will mainly introduce two categories of tracking methods: correlation filter (CF)-based methods and deep-learning-based methods.

2.1. Correlation Filter-Based Method

CF-based methods achieve many successful applications with high-speed features calculations in recent years. [13] represents objects through hand-craft features (such as HOG) and achieves high-speed tracking performance. To adapt to object scale changes, scale CFs are also designed [14]. Ref. [6] further improves tracking speed by mapping feature calculations into Fourier space. With the development of deep learning, the discrimination ability of deep features has been improved, [15,16] takes advantage of deep features to track objects. To improve the accuracy of the model, [17,18] combines the deep features of different layers with semantic and spatial information while [7] combines hand-craft and deep features to enhance the discriminative ability of the model. However, deep features with better discrimination ability are obtained through complex calculations by larger models, which limits the speed performance of CF-based methods.

2.2. Deep Learning-Based Methods

With the rapid development of deep learning in recent years, deep learning models have been widely used in the computer vision field and their powerful learning and discrimination abilities have surpassed the traditional methods with state-of-the-art performance. For object tracking, deep learning-based methods design tracking framework through deep network models, and perform supervised or semi-supervised pre-training through massive samples to obtain robust tracking models [19]. Afterward, deep reinforcement learning and Siamese network have also been introduced into object tracking. Deep reinforcement learning-based methods [20,21] can effectively transfer the training knowledge to the tracking environments and quickly adapt to the new scenes through self-learning. To accelerate the tracking process, Siamese network [22–24] uses template matching and non-update model strategies to reduce feature calculation and model update cost. However, the existing methods mainly balance speed and accuracy by selecting different deep models without optimization for deep features. Meanwhile, complex models require massive diverse training samples while most tracking methods do not have data processing or only use geometric transformation to increase sample diversity, which also limits the robustness of the model.

3. Proposed Method

In order to solve those issues, a novel robust real-time tracking method called FAF is proposed. The proposed tracking framework consists of four modules: offline IoU modulation, online IoU predictor, online classifier, and update modules as shown in Figure 1. For the offline training stage, the offline IoU modulation is independently pre-trained with massive training datasets to learn the relation between target scale and position. For the online tracking stage, the offline IoU Modulation will guide the online IoU Predictor with the IoU regression score, and the classifier will give the classification score. The joint judgment strategy will provide an optimized target scale and position information based on the classification and regression score. Then the IoU predictor and classifier will be updated by the update module. In the proposed method, ResNet18 is chosen as backbone and pre-trained on ImageNet [25]. To improve the discrimination ability of the backbone, we innovatively optimize the original backbone through the feature decomposition and sample fusion methods.

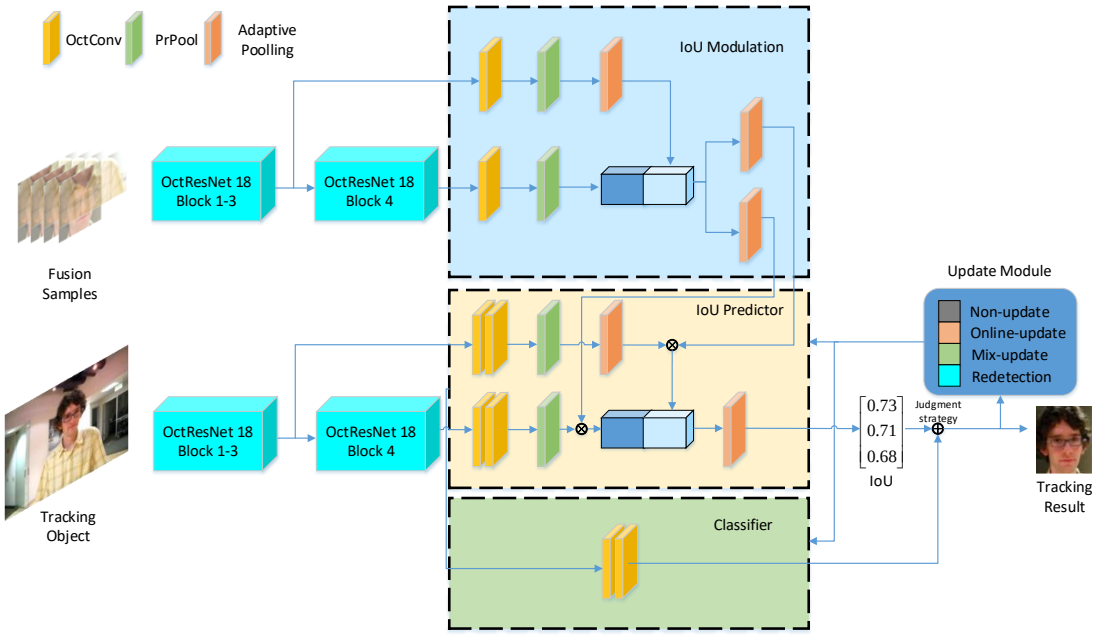

Figure 1. The framework of the proposed frequency-aware feature (FAF).

For the offline training stage as shown in Figure 1, the optimized ResNet18 obtains two-way frequency-aware features from fusion samples, as the shallow layer feature contains position information and the deep layer feature contains semantic information, and the connected features are used to learn the scale and position of the target. The conv and pooling layers are used to further improve the discrimination ability of features. The IoU Modulation is trained on large video and image datasets offline and without updates during online tracking. Our pre-training data fusion is described in Section 3.1 and the frequency-aware feature is detailed in Section 3.2.

For the online tracking stage, the first frame of the object based on data fusion will be used to initial the IoU predictor and the classifier module. Unlike the offline stage, the IoU predictor will obtain two-way features: the relevant frame guidance features from IoU Modulation and target features from the current frame. Then the IoU predictor and classifier will give the IoU and classification scores of the object in the current frame. Finally, the proposed joint judgment strategy will give the final prediction based on scores and update the IoU predictor and classifier based on the update module. The joint judgment strategy is detailed in Section 3.3.

3.1. Training Sample Fusion

Large-scale deep learning has made breakthroughs in recent years, and they have two points in common: First, more complex network structures are designed. Second, larger training datasets are proposed. Because the training dataset requires lots of manual labeling, data augmentation methods based on the existing datasets are used to increase the data. For object tracking problem, some methods apply the geometric transformation to increase data and enhance the robustness of the model. However, the existing data augmentation methods are based on the same class, and the relationship between different classes is not considered, which cannot increase the diversity of the data and limit its performance.

To solve this issue, we innovatively proposed a training sample fusion method to increase data diversity. Unlike the classification problem, the object tracking problem only contains two classes: the target and background, and pays less attention to what category the object belongs to. Inspired by [26], we enhance the data by weighting the fused samples and sample labels. With such data augmentation, the model can learn vicinity relations across examples of different classes.

To be specific, we first generate candidate samples around the ground truth bounding box by Gaussian distribution. By calculating the intersection-over-union (IoU) overlap with the ground truth, candidate samples are classified into positive and negative samples. Different from existing methods directly use the classified samples for model training, we fuse the positive and negative samples to obtain fusion samples as shown in Figure 2, the size of the fusion sample is the maximum of the two images. The details are shown in Algorithm 1.

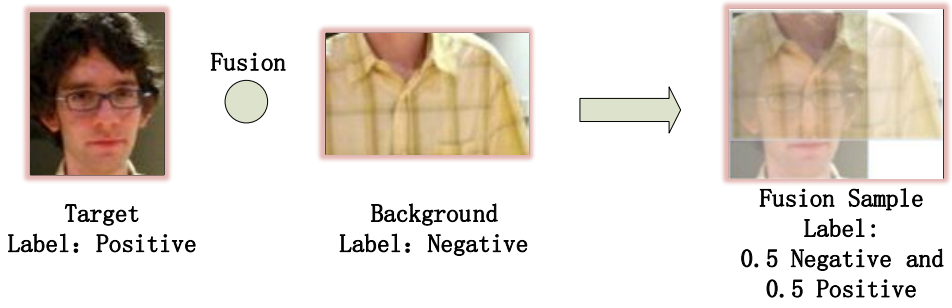

Figure 2. Training sample fusion when $\lambda = 0.5$.

Algorithm 1 Training Sample Fusion.

Input: the image M, the ground truth bound box $P(x, y, w, h)$, the number of fusion samples N_{fus}, the number of negative samples N_{neg}, the number of positive samples N_{pos}, and interpolation strength parameter α.

Generate candidate samples around $P(x, y, w, h)$ using Gaussian distribution in M
Calculate IoU for all candidate samples with ground truth
Choose N_{pos} positive samples when IoU > 0.7
Choose N_{neg} negative samples when IoU < 0.3
for $n = 0$ to N_{fus} **do**
    Random choose positive sample (x_1, y_1) and negative sample (x_2, y_2) from the corresponding sample set, respectively
    $\lambda = Beta(\alpha, \alpha)$
    $\tilde{x} = \lambda x_1 + (1.-\lambda)x_2$
    $\tilde{y} = \lambda y_1 + (1.-\lambda)y_2$
    Obtain fusion sample $(\tilde{x}, \tilde{y})$
end for
Obtain N_{fus} fusion samples
Loss = λ*criterion(outputs, y_1) + (1 - λ)*criterion(outputs, y_2)

The $\alpha \in (0, \infty)$ controls the interpolation between feature-target pairs, and generate weight λ from Beta distribution. Finally, when calculating the loss function, we calculate the loss function separately for the labels of the two samples and then perform a weighted sum of the loss functions according to the weight λ. The experiment results show that the robustness of the model can be effectively improved through data fusion.

3.2. Frequency-Aware Feature

The current models used for object tracking are fixed structures with fixed-scale convolutional layers. However, the shallow convolution contains the apparent features, while the deep convolution features contain the advanced semantic features. Therefore, the features included in traditional convolution currently have information redundancy, which increases network calculation, and the redundant information will reduce the network's ability to discriminate targets.

To address this issue, we innovatively introduce frequence-aware features into object tracking. Inspire by [27], unlike other tracking methods that distinguish between the features of different

convolution layers, we decompose the features of each convolution layer. The features in a convolution layer are divided into high-frequency features and low-frequency features, and high-frequency features contain semantic details and low-frequency features contain rough structure. By combining high-frequency features with compressed low-frequency features to reduce the network calculations and improve the network's ability to identify targets, as shown in Figure 3.

In Figure 3, the common features are divided into high-frequency and low-frequency features. Compressing the low-frequency part, processing the data of the high-frequency and low-frequency parts, and exchanging information between them, thereby can reduce the consumption of storage and calculation by the convolution operation. The size of the low-frequency part is $(0.5h, 0.5w)$, and the length and width are exactly half of the high-frequency part (h, w). Although the low-frequency part is compressed, it also effectively expands the receptive field in the original pixel space, which can improve the recognition performance. We control the high and low-frequency feature segmentation ratio by setting the hyperparameter α as follows,

$$\begin{aligned} X &\in \mathbb{R}^{c \times h \times w} \\ X^H &\in \mathbb{R}^{(1-\alpha)c \times h \times w} \\ X^L &\in \mathbb{R}^{\alpha c \times \frac{h}{2} \times \frac{w}{2}} \end{aligned} \tag{1}$$

where X means common feature, w, and h are the width and height of the feature, c is the channel number, and X^H and X^L is high-frequency and low-frequency features, respectively.

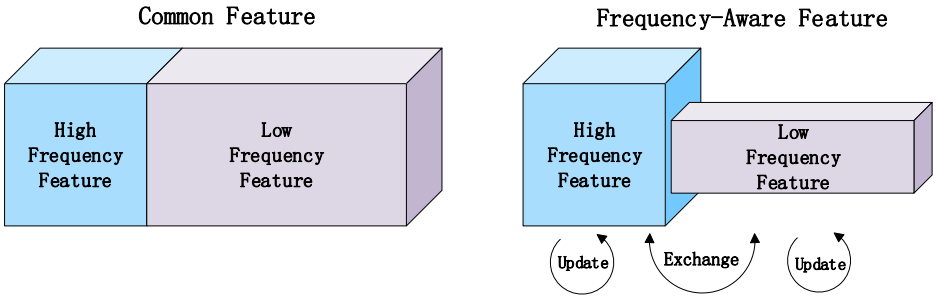

Figure 3. Frequency-aware feature.

For feature update operation, high-frequency and low-frequency features will update within the corresponding frequency. And features exchange operation will update the high-frequency and low-frequency features information between the different frequencies. Therefore, the high-frequency feature includes not only its information process, but also maps from low frequency to high frequency, and vice versa. Another advantage of the frequency-aware feature is that it has a large receptive field of low frequency-feature maps. Compared with the ordinary feature, it effectively doubles the receptive field, which will further help each frequency-aware feature capture more contextual information to improve recognition performance. As far as we know, this is the first time to design a frequency-aware feature-based Siamese network for object tracking.

3.3. Joint Judgment Strategy

The motivation of the proposed strategy comes from the classification confidence (CC) and regression confidence (RC) is separately used by tracking methods, which cannot reflect the positioning accuracy of the bounding box. Because the RC and the CC are not positively related, the existing tracking methods can only solve the high CC with high RC, but for the other three types: low CC with low RC, high CC with low RC, and low CC with high RC cannot be solved.

To solve this problem, a joint judgment strategy is designed based on [28]. Through a joint analysis of classification and regression confidence, the final prediction result has both higher classification and

regression confidences. We assume the bounding box is a Gaussian distribution $P_\Theta(x) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-x_e)}{2\sigma^2}^2}$, and the ground truth bounding box is a Dirac delta distribution $P_D(x) = \delta(x - x_g)$. The KL divergence is used to measure the asymmetry of two probability distributions. The position problem is converted to minimize the KL divergence between $P_D(x)$ and $P_\Theta(x)$, the closer the KL divergence is to 0, the more similar the two probability distributions are, which is shown as follows,

$$\hat{\Theta} = \underset{\Theta}{argmin} D_{KL}(P_D(x)||P_\Theta(x)) \tag{2}$$

where the KL divergence makes the bounding box distribute by Gaussian and closer to the ground truth. The IoU of the predicted bounding box is regarded as regression confidence. To further improve the accuracy of the bounding box, the candidate bounding boxes within the threshold IoU will be averaged based on their neighbor bounding boxes to obtain the final bounding box. Take the new $x1$ object position for ith box $x1_i$ as an example,

$$x1_i := \frac{\sum_j x1_j/\sigma^2_{x1,j}}{\sum_j 1/\sigma^2_{x1,j}} \tag{3}$$

where the final bounding box with both higher RC and higher CC is obtained. By combining the RC and the CC, we effectively solve the three situations mentioned above. Furthermore, the more accurate final bounding box will be generated based on the predicted neighbor bounding boxes, which can alleviate the loss of object due to interference information, and improve the robustness of the model in complex scenes.

4. Experiments

The proposed method is implemented in python with the PyTorch toolbox, which runs at 45 fps on a PC with a 4-cores 4.2 GHz Intel 8700k CPU and two NVIDIA 2080 Ti GPU with 11G memory. TrackingNet, OxUvA, and LaSOT datasets are used for pre-training and the network parameters remain the same for all evaluation datasets. All hyperparameters are set according to related works. The training parameters are described as follows. For the backbone network, we freeze all weight during training. For the network , the weight decay is 0.00005, and momentum is 0.9. Dropout (50%) is used in the first two fc layers. We use the mean-squared error loss function and train for 40 epochs with 64 image pairs per batch. The ADAM optimizer is employed with initial learning rate of 10^{-3}, and using a factor 0.2 decay every epochs. The experiments are carefully designed based on the same protocols and parameters.

4.1. Evaluation Otb100

The proposed FAF is first evaluated on a famous benchmark dataset OTB100 dataset. Eight state-of-the-art trackers are compared with the proposed method, including ECO [7], MDNet [19], ATOM [29], DeepSRDCF [30], CF2 [17], HDT [18], and KCF [6]. These methods include CF-based methods, deep learning-based methods, and reinforcement learning-based methods.

The tracking results of state-of-the-art methods under one-pass evaluation (OPE) on OTB100. As shown in Figure 4, the proposed FAF exhibits high precision and success rates. Compared with state-of-the-art real-time tracker ATOM with 30 FPS, our tracker achieves 90.1% and 67.3% in the precision and success rates, which are 1.9% and 1.4% higher than ATOM. KCF uses a handcraft feature and can track at 160 FPS. However, due to the weak discrimination ability, lower tracking accuracy is obtained. ECO and MDNet both use deep models with optimization, and achieve better tracking performance, but they cannot meet the real-time tracking requirements. In addition, our tracker outperforms them in both speed and accuracy in the following datasets experiments.

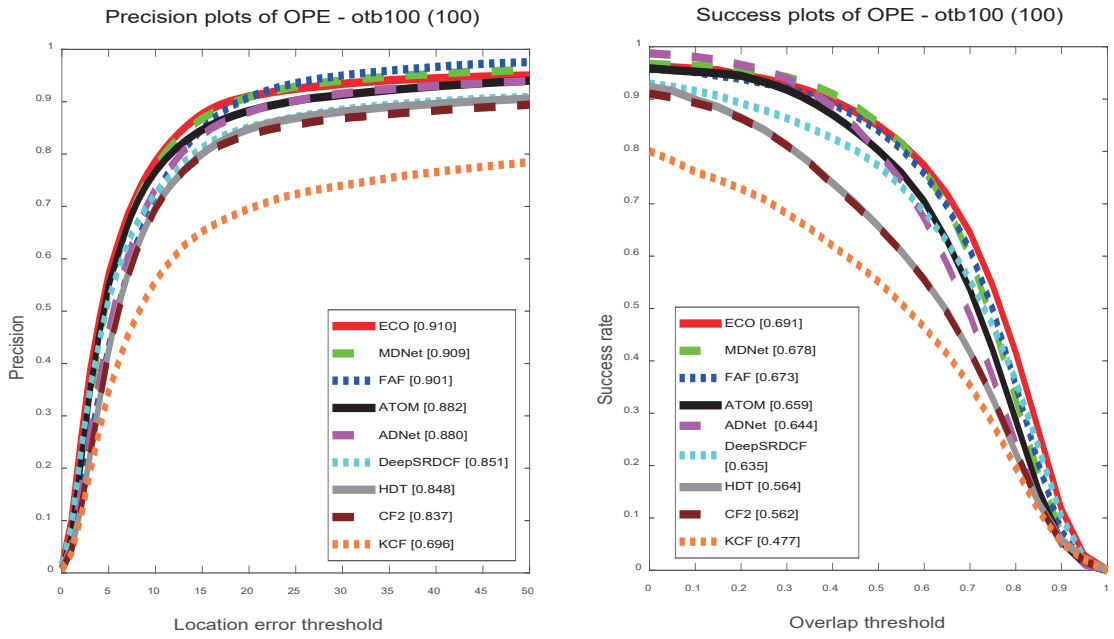

Figure 4. The precision and success rate plots on the OTB100 dataset are performed using the one pass evaluation (OPE) method. The proposed method performs well compared to state-of-the-art methods.

4.2. Ablation Analysis

To analysis the accuracy and speed depend on alpha, we compare different alpha value on OTB100 dataset. As shown in Table 1, we only increase positive or negative samples and no mixup samples obtained when alpha is 0 or ∞. The tracker speed will be improved without mixup process. The tracker performs better when alpha = 1, it gains 0.015 and 0.013 improvement than alpha = 0.5 on precision and AUC rates, respectively. For mixup samples are hard samples when when alpha = 1, it can help the trained model to have better robustness.

Table 1. Analysis the accuracy and speed of the proposed method depend on α. The best results are in bold.

	0	**0.5**	**1**	**10**	**∞**
Prec.	0.875	0.886	**0.901**	0.891	0.883
AUC	0.654	0.660	**0.673**	0.664	0.659
FPS	**48**	45	45	45	**48**

To demonstrate the effectiveness of each component in the proposed method FAF, ablation experiments are performed on OTB2015. The baseline means the original model without any optimization, "**I**" means the baseline with training sample fusion optimization, and "**I + II**" denotes the baseline with both training sample fusion and frequency-aware feature optimizations. For the version of the full components "**I + II + III**" denotes the complete model with all training sample fusion, frequency-aware feature, and joint judgment strategy optimizations. The performance of all those variations is shown in Table 2, and every component can improve the performance of the proposed method.

Table 2. Ablation results of the FAF on the OTB100, which shows the effectiveness of each component of the proposed method. The best results are in bold.

	Baseline	**I**	**I + II**	**I + II + III**
Prec.	0.882	0.887	0.893	**0.901**
AUC	0.659	0.662	0.667	**0.673**
FPS	30	30	**45**	**45**

Training sample fusion: Training sample fusion increases the diversity of samples and enhances the ability of the model to learn vicinity relation across different classes, which can enhance the discrimination ability of model without extra cost. The results show that 1.1% and 0.8% have been improved on precision and AUC rates, respectively.

Frequency-aware feature: Frequency-aware feature increases the precision and AUC rates by 2.0% and 2.2%, and dramatically improves tracking speed by 1.5 times. Because we innovatively decompose the layer feature into high-frequency and low-frequency features, and compress the redundant low-frequency feature and splice them into multi-frequency features. Without increasing model complexity, the frequency-aware feature can reduce redundant in low-frequency feature calculations and further improves the feature discrimination ability of the proposed model.

Joint judgment strategy: Finally, to obtain a more accurate target position, the joint judgment strategy is proposed by considering both classification and regression results. As shown in Table 2, the precision and AUC rates are improved by 0.7% and 0.6%, respectively.

4.3. State-Of-The-Art Comparison

We compare our tracker FAF with state-of-the-art methods on four challenging tracking datasets.

VOT2018: VOT2018 consists of 60 test video sequences and the performance are evaluated by failure rate (R), average overlap (A), and Expected Average Overlap (EAO) to provide the overall performance ranking. We choose short-term tracking tests with state-of-the-art methods for comparison. As shown in Table 3, we compare our method with the five top methods in the VOT2018 dataset. Our method achieves the best R and EAO scores while having a competitive A score. Among the top trackers, only SiamRPN++ achieves a 0.003 higher accuracy score than the proposed method. Compared with ATOM, our method obtains 2.1%, 2.5%, and 0.7% improvements on EAO, R, and A score, respectively.

GOT10K: GOT10K includes more than 10,000 video sequences and the target frames are over 1.5 million, all of which are manually annotated. The data set consists of five categories: animals, man-made objects, people, natural scenery, and part, which can be subdivided into 563 target categories. Only the GOT10K dataset is used to train model and 180 test video sequences are used to evaluate the performance of FAF with five state-of-the-art methods. As shown in Table 4. FAF achieves the best scores with 0.581, 0.453 and 0.672 on AUC, precision (0.5) and precision (0.75) rates. Compared with non-real-time methods ECO and MDNet, the proposed method achieves huge improvements in all three evaluation indexes.

TrackingNet: TrackingNet uses the video sequences in Youtube-BB and divides the original 23 categories into 27 categories. The video sequence is divided into 15 attributes by automatically estimated and visually inspected. Use the DCF tracker to label missing target boxes. There are 12 chunks of 2511 sequences for the training and 1 chunk of 511 sequences for the testing. Table 5 shows the results in terms of precision, normalized precision, and AUC. In terms of precision, normalized precision, and AUC, C-RPN achieves scores of 0.619, 0.749, and 0.669, respectively. The proposed method FAF outperforms the second method ATOM with 1.9%, 1.5%, and 2.4% in terms of precision, normalized precision, and AUC rates, respectively.

LaSOT: LaSOT collects 1,400 sequences and 3.52 million frames of YouTube videos with an average video length of 2512 frames. It contains 70 categories and each category contains 20 sequences, the training subset contains 1120 videos, 2.83m frames, and the test subset contains 280 sequences,

690k frames. We evaluate the proposed method with five state-of-the-art methods on the test dataset with 280 sequences. The results in terms of normalized precision and success are shown in Table 6. Among those state-of-the-art methods, FAF achieves the best AUC and precision scores with 0.537 and 0.601. Compared with SiamRPN++, our method significantly improves the AUC and precision rates with 4.1% and 3.2%, respectively.

Table 3. Comparison with state-of-the-art trackers on the VOT 2018 dataset. The results are presented in terms of expected average overlap (EAO), accuracy value (A), and robustness value (R). The best and second results are in red and blue, respectively.

	SiamRPN++ [22]	ATOM	UPDT [31]	DaSiamRPN [32]	DRT [33]	FAF
EAO	0.414	0.401	0.378	0.383	0.356	0.422
R	0.234	0.204	0.184	0.276	0.201	0.179
A	0.6	0.59	0.536	0.586	0.519	0.597
FPS	35	30	-	160	-	45

Table 4. Comparison with the state-of-the-art trackers on the GOT10K dataset. The results are presented in terms of precision (0.5), precision (0.75), and robustness value (R). The best and second results are in red and blue, respectively.

	ATOM	SiamFC [34]	ECO	MDNet	CCOT [35]	FAF
Prec.(0.5)	0.634	0.404	0.309	0.303	0.328	0.672
Prec.(0.75)	0.402	0.144	0.111	0.099	0.104	0.453
AUC	0.556	0.374	0.316	0.299	0.325	0.581
FPS	30	80	8	1	1	45

Table 5. Comparison with state-of-the-art trackers on the TrackingNet dataset. The results are presented in terms of precision, normal.precision, and AUC. The best and second results are in red and blue, respectively.

	ATOM	GFS-DCF [36]	UDT [37]	C-RPN [23]	CACF [38]	FAF
Prec.	0.648	0.566	0.557	0.619	0.536	0.667
Norn.Prec	0.771	0.718	0.702	0.749	0.467	0.786
AUC	0.703	0.609	0.611	0.669	0.608	0.727
FPS	30	8	55	32	35	44

Table 6. Comparison with state-of-the-art trackers on the LaSOT dataset. The results are presented in terms of precision and AUC. The best and second results are in red and blue, respectively.

	GradNet [39]	ATOM	SiamRPN++ [22]	SPM [40]	C-RPN	FAF
Prec.	0.351	0.576	0.569	0.471	0.459	0.601
AUC	0.365	0.515	0.496	0.485	0.455	0.537
FPS	80	30	35	120	32	44

4.4. Failure Case Analysis

As shown in Figure 5, the first row is the Singer2 sequence, and the second row is the Tran sequence, the proposed method does not perform well on those two sequences. For the Singer2 sequence, the the target and the background are too similar, the proposed method does not distinguish between them accurately and looses the target. For the Tran sequence, the scale and appearance of the target has changed drastically during the tracking process. Our tracker does not learn the target characteristics accurately during the rapid and dramatic change of the target, which eventually caused the target to be lost. We will try to design size-aware module and use handcraft features to solve those problems in future work.

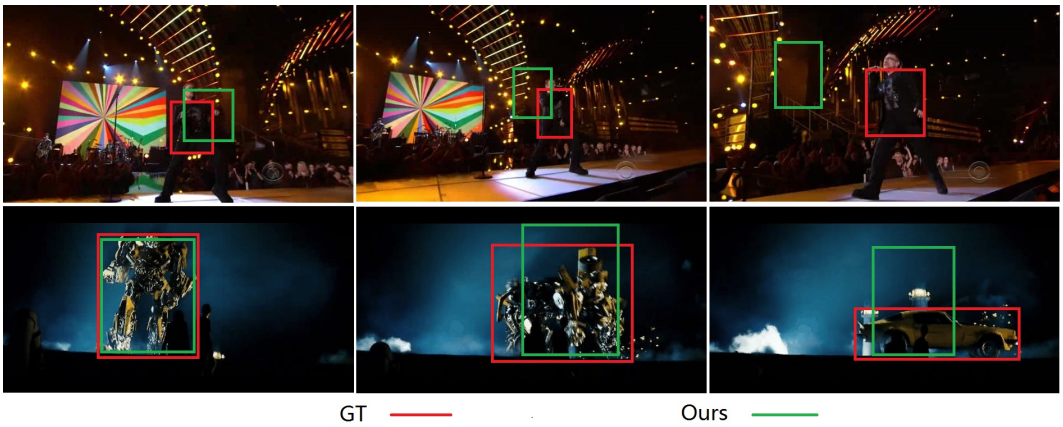

Figure 5. Failure case analysis. The red and green bounding boxes are the ground truth and results of the proposed method, respectively.

5. Conclusions

In this paper, we present a novel tracking method FAF based on frequency-aware feature and sample fusion. Our method innovatively factorizes feature map into different frequency features and reduce the redundant information. The frequency-aware feature can improve the discrimination ability by enlarging the receptive field of layers, while reducing calculations by compressing the low-frequency feature. Further, our method designs a data-independent augmentation for object tracking model training. The model can learn vicinity relations across different classes by convex combination of both tags and images, which can improve the discrimination ability of model. Finally, a joint judgment strategy based on regression and classification scores is proposed to fine-tune the bounding box of the target, which can solve the conflict of regression and classification scores in complex scenes and improve the robustness of the model. Extensive experiments on five famous benchmarks show that our proposed FAF performs favorably against SOTA tracking methods while running around 45 frames per second.

Author Contributions: Conceptualization, Y.Y. and W.X.; methodology, Y.Y. and S.Z.; software, Y.Y. and X.G.; validation, Y.Y., Q.Y. and M.G.; formal analysis, W.X.; investigation, W.X., Q.Y.; resources, S.Z., X.G. and Q.Y; data curation, Q.Y. and M.G.; writing—original draft preparation, Y.Y., W.X. and S.Z.; writing—review and editing, W.X. and S.Z.; visualization, Y.Y. and M.G.; supervision, Q.Y. and X.G.; project administration, W.X. and S.Z.; funding acquisition, W.X. and S.Z. 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 (No.61876018, No.61601021, No.61976017), the Beijing Natural Science Foundation (L172022).

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

References

1. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proceedings of the International Conference for Learning Representations San Diego, CA, USA, 7–9 May 2015.
2. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
3. Sheng, M.; Wang, W.; Qin, H.; Wan, L.; Li, J.; Wan, W. A Novel Changing Athlete Body Real-Time Visual Tracking Algorithm Based on Distractor-Aware SiamRPN and HOG-SVM. *Electronics* **2020**, *9*, 378. [CrossRef]
4. Yang, Y.; Xing, W.; Zhang, S.; Gao, L.; Yu, Q.; Che, X.; Lu, W. Visual Tracking With Long-Short Term Based Correlation Filter. *IEEE Access* **2020**, *8*, 20257–20269. doi:10.1109/ACCESS.2020.2968125. [CrossRef]
5. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.

6. Henriques, J.F.; Caseiro, R.; Martins, P.; Batista, J. High-speed tracking with kernelized correlation filters. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 583–596. [CrossRef] [PubMed]
7. Danelljan, M.; Bhat, G.; Khan, F.S.; Felsberg, M. ECO: Efficient Convolution Operators for Tracking. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
8. Wu, Y.; Lim, J.; Yang, M.H. Object tracking benchmark. *IEEE Trans. Pattern Anal. Mach. Intel.* **2015**, *37*, 1834–1848. [CrossRef] [PubMed]
9. 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 Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 5374–5383.
10. Huang, L.; Zhao, X.; Huang, K. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *arXiv* **2018**, arXiv:1810.11981.
11. Muller, M.; Bibi, A.; Giancola, S.; Alsubaihi, S.; Ghanem, B. Trackingnet: A large-scale dataset and benchmark for object tracking in the wild. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 300–317.
12. Kristan, M.; Leonardis, A.; Matas, J.; Felsberg, M.; Pflugfelder, R.; Cehovin Zajc, L.; Vojir, T.; Bhat, G.; Lukezic, A.; Eldesokey, A.; et al. The sixth visual object tracking vot2018 challenge results. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018.
13. Zhang, J.; Ma, S.; Sclaroff, S. MEEM: robust tracking via multiple experts using entropy minimization. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 188–203.
14. Danelljan, M.; Häger, G.; Khan, F.; Felsberg, M. Accurate scale estimation for robust visual tracking. In Proceedings of the British Machine Vision Conference, Nottingham, UK, 1–5 September 2014.
15. Hong, S.; You, T.; Kwak, S.; Han, B. Online tracking by learning discriminative saliency map with convolutional neural network. In Proceedings of the IEEE International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 597–606.
16. Lee, D.H. Fully Convolutional Single-Crop Siamese Networks for Real-Time Visual Object Tracking. *Electronics* **2019**, *8*, 1084. [CrossRef]
17. Ma, C.; Huang, J.B.; Yang, X.; Yang, M.H. Hierarchical convolutional features for visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3074–3082.
18. Qi, Y.; Zhang, S.; Qin, L.; Yao, H.; Huang, Q.; Lim, J.; Yang, M.H. Hedged deep tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4303–4311.
19. 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.
20. Chen, B.; Wang, D.; Li, P.; Wang, S.; Lu, H. Real-time'Actor-Critic'Tracking. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 318–334.
21. Yun, S.; Choi, J.; Yoo, Y.; Yun, K.; Choi, J.Y. Action-Decision Networks for Visual Tracking with Deep Reinforcement Learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1349–1358.
22. 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 Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–21 June 2019; pp. 4282–4291.
23. Fan, H.; Ling, H. Siamese cascaded region proposal networks for real-time visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–21 June 2019; pp. 7952–7961.
24. Liu, P.; Li, X.; Liu, H.; Fu, Z. Online Learned Siamese Network with Auto-Encoding Constraints for Robust Multi-Object Tracking. *Electronics* **2019**, *8*, 595. [CrossRef]
25. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.

26. Zhang, H.; Cisse, M.; Dauphin, Y.N.; Lopez-Paz, D. mixup: Beyond empirical risk minimization. In Proceedings of the International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
27. Chen, Y.; Fang, H.; Xu, B.; Yan, Z.; Kalantidis, Y.; Rohrbach, M.; Yan, S.; Feng, J. Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution. *arXiv* **2019**, arXiv:1904.05049.
28. He, Y.; Zhang, X.; Savvides, M.; Kitani, K. Softer-nms: Rethinking bounding box regression for accurate object detection. *arXiv* **2018**, arXiv:1809.08545.
29. Danelljan, M.; Bhat, G.; Khan, F.S.; Felsberg, M. Atom: Accurate tracking by overlap maximization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–21 June 2019; pp. 4660–4669.
30. Danelljan, M.; Hager, G.; Shahbaz Khan, F.; Felsberg, M. Convolutional features for correlation filter based visual tracking. In Proceedings of the International Conference on Computer Vision Workshop, Santiago, Chile, 7–13 December 2015; pp. 58–66.
31. Bhat, G.; Johnander, J.; Danelljan, M.; Shahbaz Khan, F.; Felsberg, M. Unveiling the power of deep tracking. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 483–498.
32. 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, Munich, Germany, 8–14 September 2018; pp. 101–117.
33. Sun, C.; Wang, D.; Lu, H.; Yang, M.H. Correlation tracking via joint discrimination and reliability learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 489–497.
34. Bertinetto, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H. Fully-convolutional siamese networks for object tracking. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 850–865.
35. Danelljan, M.; Robinson, A.; Khan, F.S.; Felsberg, M. Beyond correlation filters: Learning continuous convolution operators for visual tracking. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 472–488.
36. Xu, T.; Feng, Z.H.; Wu, X.J.; Kittler, J. Joint group feature selection and discriminative filter learning for robust visual object tracking. In Proceedings of the International Conference on Computer Vision, Cardiff, Wales, 9–12 September 2019; pp. 7950–7960.
37. Wang, N.; Song, Y.; Ma, C.; Zhou, W.; Liu, W.; Li, H. Unsupervised deep tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–21 June 2019; pp. 1308–1317.
38. Mueller, M.; Smith, N.; Ghanem, B. Context-aware correlation filter tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1396–1404.
39. Li, P.; Chen, B.; Ouyang, W.; Wang, D.; Yang, X.; Lu, H. Gradnet: Gradient-guided network for visual object tracking. In Proceedings of the International Conference on Computer Vision, Cardiff, Wales, 9–12 September 2019; pp. 6162–6171.
40. Wang, G.; Luo, C.; Xiong, Z.; Zeng, W. Spm-tracker: Series-parallel matching for real-time visual object tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–21 June 2019; pp. 3643–3652.