


Article

Adaptive Spatial-Temporal Regularization for Correlation Filters Based Visual Object Tracking

Fei Chen  and Xiaodong Wang

College of Computer, National University of Defense Technology, Changsha 410073, China; xdwang@nudt.edu.cn

* Correspondence: chenfei14@nudt.edu.cn; Tel.: +86-155-7085-4868

Abstract: Recently, Discriminative Correlation Filters (DCF) have shown excellent performance in visual object tracking. The correlation for a computing response map can be conducted efficiently in Fourier domain by Discrete Fourier Transform (DFT) of inputs, where the DFT of an image has symmetry on the Fourier domain. To enhance the robustness and discriminative ability of the filters, many efforts have been devoted to optimizing the learning process. Regularization methods, such as spatial regularization or temporal regularization, used in existing DCF trackers aim to enhance the capacity of the filters. Most existing methods still fail to deal with severe appearance variations—in particular, the large scale and aspect ratio changes. In this paper, we propose a novel framework that employs adaptive spatial regularization and temporal regularization to learn reliable filters in both spatial and temporal domains for tracking. To alleviate the influence of the background and distractors to the non-rigid target objects, two sub-models are combined, and multiple features are utilized for learning of robust correlation filters. In addition, most DCF trackers that applied 1-dimensional scale space search method suffered from appearance changes, such as non-rigid deformation. We proposed a 2-dimensional scale space search method to find appropriate scales to adapt to large scale and aspect ratio changes. We perform comprehensive experiments on four benchmarks: OTB-100, VOT-2016, VOT-2018, and LaSOT. The experimental results illustrate the effectiveness of our tracker, which achieved a competitive tracking performance. On OTB-100, our tracker achieved a gain of 0.8% in success, compared to the best existing DCF trackers. On VOT2018, our tracker outperformed the top DCF trackers with a gain of 1.1% in Expected Average Overlap (EAO). On LaSOT, we obtained a gain of 5.2% in success, compared to the best DCF trackers.

Keywords: Discriminative Correlation Filters; visual object tracking; adaptive spatial regularization; temporal regularization; scale space search; non-rigid deformation



Citation: Chen, F.; Wang, X. Adaptive Spatial-Temporal Regularization for Correlation Filters Based Visual Object Tracking. *Symmetry* **2021**, *13*, 1665. <https://doi.org/10.3390/sym13091665>

Academic Editor: Raúl Baños Navarro

Received: 30 July 2021

Accepted: 5 September 2021

Published: 9 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

An object tracking algorithm aims to track the object's position in a 2D or 3D input, such as wireless signal, radar (i.e., a radar echo), or camera (i.g., video frame). For example, the Bluetooth 5.1 Direction Finding standard provides the probability of high-precision and real-time tracking of targets based on Angle of Departure (AoD) and the Angle of Arrival (AoA) [1]. Visual object tracking, as the main topic of this paper, is an important area in computer vision, which estimates the trajectory of the target object with visual information from a video sequence. visual object tracking can be applied into many applications, such as video surveillance, motion analyses, human computer interaction, automatic robot navigation, and traffic monitoring.

In recent years, the Discriminative Correlation Filter (DCF) for tracking has attracted a lot of attention due to its efficiency and effectiveness, from the trackers based on hand-crafted features [2–4] to the trackers that exploit learning with deep features [5–7]. The core idea in DCF trackers is how to learn robust and discriminative filters to adapt to appearance changes online by minimizing a least-squares loss for all circular shifts of a training sample.

Compared with the end-to-end deep trackers that require large-scale training set for offline training, the DCF-based trackers are model-free and do not need offline training. Although the deep trackers obtain superior performance on challenging sequences with attributes such as deformation, rotation, and scale variation. One limitation of deep Siamese trackers, such as SiamRPN++ [8], is that the model can easily lose the target if it is occluded by the similar distractors. While state-of-the-art DCF trackers exhibit more stable performance on non-rigid objects and are even more robust in the case of heavy occlusion caused by similar distractors.

The recent advancements in this area include applying different kinds of regularization techniques [4,6,9–11], developing an efficient model update strategy [12], reducing boundary effects [10], employing background information [9,10], applying kernel tricks [13,14], and using the efficient scale estimating scheme [3].

There are a variety of regularizations, such as spatial regularization [4], spatial-temporal regularization [6], context regularization [9,10], and adaptive spatial regularization [11], which can steadily improve the tracking performance on public tracking benchmarks. As spatial-temporal regularized correlation filters (STRCF) [6] and adaptive spatially-regularized correlation filters (ASRCF) [11] merely consider limited auxiliary information, both of them still suffer from severe appearance changes on challenging video sequences.

ASRCF does not consider the temporal consistent, while STRCF does not exploit the changes of spatial constraints over time explicitly. In this paper, we first employ both the temporal information and spatial continuity during the filters learning procedure. Secondly, we study the problem of multiple features (e.g., Histogram of Oriented Gradient (HOG) [15], color features [16], and deep features [17]) combination for correlation filters learning. We found that different feature combinations lead to different impacts on the tracking performance.

The feature representation of the different layers in a deep Convolution Neural Network (CNN) are different, and one limitation of most existing DCF trackers [5–7] is the utilization of the specific layer of the CNN model. In some cases, the appearance of the target experiences significant changes caused by deformation or rotation. For example, the DeepSTRCF [6] and ASRCF [11] both fail to track the target on *MotorRolling* sequence in OTB-100 [18], in which the target rotates in the image plane and has experienced severe non-rigid deformation.

Another limitation of the published DCF trackers is that most of them, such as STRCF [6] and ASRCF [11] utilize the 1-dimensional scale space search method to find the optimal scale. By applying the scale factor, the height and width of the tracking result are adjusted simultaneously in the current frame. However, such a scaling scheme is not sufficient for severe appearance changes, such as deformation, viewpoint change, and aspect ratio change.

In this paper, we explore the scale estimation in 2-dimensional scale space to address the above limitation, in which it can adjust the height and width of target with two different scale factors. By applying the new scale space search scheme, we demonstrate that it leads to better adaptive ability to aspect ratio changes and severe deformation.

The contributions of this paper can be summarized as follows:

- We develop a novel tracker that seamlessly applies temporal and adaptive spatial regularization to learn the correlation filters, which can significantly improve the accuracy and robustness of the tracker for videos with challenging attributes.
- An efficient optimization procedure is presented, and the Alternating Direction Method of Multipliers (ADMM) algorithm is applied for solving each sub-problem.
- We propose a 2-dimensional scale space search method for finding the optimal 2-dimensional aspect ratios to adapt to target deformation, instead of the 1-dimensional scale factor. We validate our approach through sufficient experiments, and the tracking results demonstrate that our new scale search method can achieve significant improvements. On

LaSOT, our approach outperforms its counterpart DeepSTRCF in success score by 5.6% on the deformation attribute and 5.4% on the Scale Variation attribute.

- We evaluate our method on OTB-100 [18], VOT-2016 [19], VOT-2018 [20], and LaSOT [21] tracking datasets and report the state-of-the-art tracking performance. On OTB-100, our approach outperforms its counterparts ASRCF and DeepSTRCF by 0.8% and 2.3% in success. Compared to DeepSTRCF [6], our tracker improves by 5.2% in success.

The remainder of the paper is structured as follows. In Section 2, we review the most closely related work on visual object tracking, the traditional DCF tracking procedure is described in Section 3, our approach is presented in Section 4, experimental results are presented in Section 5, and our conclusions are drawn in Section 6.

2. Related Work

Most of the traditional tracking algorithms can be categorized as either generative or discriminative. The generative trackers search the region that is the most similar to the reference target in a video sequence, such as ℓ_1 tracker [22], Multi-Task Tracking (MTT) [23], and Multi-task Correlation Particle Filter (MCPF) [24]. The discriminative trackers aim to distinguish the target from the background by a detection model, such as the Multiple Instance Learning (MIL) classifier used in [25] and the kernelized structured output support vector machine (SVM) applied in Struck [26].

The tracking-by-detection [27] tracker exploited the circular structure for fast learning of the classifier and detecting with the dense sampling strategy. Henriques et al. [13] proposed kernelized correlation filters (KCF) based on linear and kernel ridge regression for fast visual object tracking. In order to utilize multi-channel RGB images or features (e.g., HOG), Kiani Galoogahi et al. proposed the Multi-channel correlation filter (MCCF) strategy for car localization, with the efficient use of memory and computations. Tang et al. [14] demonstrated that the single kernel can be decomposed into multiple kernels that can be applied to fuse multiple channel features for tracking.

Regularization methods are widely used for solving the least-squares problem, such as the spatial regularizer in [4], total generalized variation (TGV) regularizer in [28], and temporal regularizer in [6]. To solve the boundary effect problem caused by the circular assumption, Danelljan et al. [4] proposed spatially regularized discriminative correlation filters (SRDCF), which adds spatial regularization to the optimization problem via a weighted function.

The spatial weights are based on the prior information about the spatial extent of the filter. The framework in [10] allows the learning of background-aware correlation filters (BACF) from both the background and target region. In addition, Mueller et al. [9] applied context image patches around the target to learn discriminative context-aware correlation filters (CACF) for tracking.

Similar to SRDCF [4], Li et al. [6] proposed spatial-temporal regularized correlation filters (STRCF) that exploit both the spatial and temporal regularizations during model learning. The closed solution can be obtained by the ADMM algorithm, which performs the learning procedure only on a single image instead of using multiple samples in SRDCF [4]. In [29], Lukežič et al. proposed a discriminative correlation filter with Channel and Spatial Reliability (CSR-DCF) model, in which the channel and spatial reliability are combined to constrain the filter learning. The unsupervised segmentation mask is utilized as the spatial reliability.

In [30], Ma et al. applied multiple convolutional features with identical resolution for learning independent Correlation Filters (CFs) and then hierarchically estimated the final response. In [7], Danelljan et al. proposed the Continuous Convolution Operator Tracker (C-COT) to learn the CFs in a continuous domain and integrate multi-resolution shallow and deep feature maps. In [12], Danelljan proposed Efficient Convolution Operators (ECO) to reduce the computation cost while preserve the tracking performance. However, the multiple samples used for learning CFs still led to a high computation cost.

Zhang et al. proposed an ensemble tracker based on Multi-Expert Entropy Minimization (MEEM) restoration scheme to address the model drift problem. Li et al. [31] extended the MEEM to multi-expert in a collaborative way, instead of the independent entropy computation in MEEM. Similarly, Li et al. [32] utilized the unified discrete graph to model the relationship of multi-expert, while the computation load of dynamic programming increased significantly with a larger number of experts.

In [33], Wang et al. proposed a strategy to adaptively select the suitable expert from a pool of experts which was generated with different features combinations. Bhat et al. [34] combined the shallow feature model and deep feature model through adaptive weights to enhance the capacity of the model, in which the model can be obtained by solving the Quadratic Programming problem. While the limitation of it is that they only consider the two different features and the fusion happens in the model prediction stage. The weighted confidence margin is computed with the confidence scores and their locations, which does not consider the scale variation of target object.

Dai et al. [11] proposed adaptive spatially-regularized correlation filters (ASRCF) to further improve the accuracy of CFs by adding the adaptive spatial regularization, while they directly concatenate the multiple features (i.e., HOG and deep features) together for CF learning.

Although the advanced CF trackers, like STRCF [6] and ASRCF [11], have achieved state-of-the-art performance in most sequences on the OTB-100 [18] dataset, the sequences, such as *Jump*, *MotorRolling*, and *Diving* in OTB-100 [18], still pose challenges. Most existing DCF trackers estimate the scale of target by 1-dimensional scale space search, which limits the tracker's ability to adapt to large scale ratio aspect changes due to challenges, such as deformation, rotation, and viewpoint changes.

Recently, with the development of large scale datasets, the improvement of computing power, and reliable and efficiency computing resources allocation [35], applications based on deep learning have entered a period of rapid development. Trackers based on the deep Siamese network have also drawn great attention and shown significant improvements regarding tracking performance.

In [36], Bertinetto et al. proposed the Siamese Fully-Convolutional (SiamFC) tracker based on an end-to-end fully-convolutional Siamese network and achieved high tracking speed beyond real time. In [37], the Correlation Filter Network (CFNet) was proposed for the joint learning of deep features and correlation filters in an end-to-end manner.

In the Siamese region proposal network (SiamRPN) [38], Li et al. took the advantages of the Siamese network and region proposal network (RPN) to improve the accuracy of localization and regression of the target bounding box. Furthermore, SiamRPN++ [8] employed a lightweight Depthwise Cross Correlation (DW-XCorr) layer instead of the Channel Cross Correlation (UP-XCorr) layer in SiamRPN, which achieved state-of-the-art tracking performance.

In contrast to most of the Siamese trackers, Danelljan et al. [39] proposed the Accurate Tracking by Overlap Maximization (ATOM) tracker, which combined the offline trained target estimation module and online learned target classification module for tracking. In [40], a model predictor was designed to estimate the target model, which was trained with a set of multiple samples of the sequence instead of the image pairs in Siamese approaches.

In [41], Zhang et al. designed new residual modules to build deeper and wider backbone networks, in which the cropping-inside residual (CIR) unit was utilized to remove the effect of padding operation, and the downsampling cropping-inside residual (CIR-D) unit was utilized to reduce the spatial size of the feature maps. One limitation in recent deep trackers is that they need to train the target model with large scale datasets in the offline phase.

Different from the DCF trackers that update the model with a continuous strategy in each frame [10,11] or a sparser scheme in every N frames [12], the deep trackers, such as SiamRPN++ [8] do not update the model during tracking, as they are sensitive to sudden changes caused by deformation and viewpoint change. Deep trackers (e.g., SiamRPN [38])

and ATOM [39]) also easily lose the target when there are many similar distractors around the target.

3. Revisiting Traditional Correlation Filter Tracking

In this section, we first introduce the standard discriminative correlation filters for visual object tracking. Then, we briefly describe the two baseline trackers, STRCF [6] and SRDCF [11]. These two methods utilize different regularizations for filter learning.

3.1. Revisiting Standard Correlation Filter

Given a sequence of images and its initial state, including the position and size of target, the task of standard DCF trackers is to learn CFs using a set of training samples $\{x_i, y_i\}_{i=1}^N$, where x_i denote the multiple channel feature representation of i -th image, y_i is the corresponding desired output. The main objective function can be defined as:

$$\ell = \sum_{i=1}^N \left\| \sum_{d=1}^D f^d \star x_i^d - y_i \right\|_2^2 + \frac{\lambda}{2} \|f\|_2^2 \quad (1)$$

Here, λ represents the weight of regularization term. f is the correlation filter. According to Parseval's theorem, the filter f can be efficiently formulated in the frequency domain as

$$\hat{f} = \frac{\sum_{i=1}^N \hat{y}_i \odot \text{conj}(\hat{x}_i)}{\sum_{i=1}^N \hat{x}_i \odot \text{conj}(\hat{x}_i) + \lambda I} \quad (2)$$

where \odot denotes the element-wise multiplication of two vectors. The $\hat{\cdot}$ denotes the Discrete Fourier Transform (DFT) \mathcal{F} , which possesses the property of symmetry, i.e., $\hat{x}_i = \mathcal{F}\{x_i\}$ and $\hat{y}_i = \mathcal{F}\{y_i\}$ are the DFT of the sample and its desired output. conj denotes the complex conjugation operation. Given a test image patch z , the corresponding response can be obtained by:

$$G = \mathcal{F}^{-1} \left\{ \sum_{d=1}^D \hat{z}^d \odot \text{conj}(\hat{f}^d) \right\} \quad (3)$$

where \mathcal{F}^{-1} denotes the inverse DFT operation.

3.2. Revisiting STRCF

In STRCF [6], a temporal regularization term $\|f - f_{t-1}\|_2^2$ is incorporated into the objective as follows:

$$L = \arg \min_f \frac{1}{2} \left\| \sum_{d=1}^D x_t^d \star f^d - y \right\|_2^2 + \frac{\lambda}{2} \sum_{d=1}^D \|w \odot f^d\|_2^2 + \frac{\mu}{2} \|f - f_{t-1}\|_2^2 \quad (4)$$

where w denotes the spatial regularization weight as SRDCF [4]. λ and μ are the penalty factors of the two regularization terms. The temporal regularization aims to exploit relationship between adjacent samples by passively updating the correlation filters, making the filter learned in the current frame close to the one in the previous frame.

3.3. Revisiting ASRCF

In ASRCF [11], Dai et al. introduced a regularization based on spatial weight w and its reference w^r . The objective can be formulated as:

$$L = \arg \min_f \frac{1}{2} \left\| \sum_{d=1}^D x_t^d \star (P^\top f^d) - y \right\|_2^2 + \frac{\lambda}{2} \sum_{d=1}^D \|w \odot f^d\|_2^2 + \frac{\mu}{2} \|w - w^r\|_2^2 \quad (5)$$

where λ and μ are the penalty factors of the two spatial regularization terms. P is the cropping matrix to encourage using large number of negative samples from surrounding

background for training. The third term is used to adjust the spatial weight during model learning.

4. Our Approach

In this section, we propose a novel tracking framework that simultaneously considers the spatial similarity of the target between current frame and reference frame, and the temporal consistency of filters in adjacent frames. For accurate scale estimation, we proposed a 2-dimensional scale space search method to adapt to appearance changes caused by challenging scenarios, such as deformation, target rotation, and viewpoint change. The Figure 1 shows the diagram of our tracking approach. The two sub-models are refer to two different learning rates.

First, the feature extractor is utilized to extract the features of an image patch. After obtaining the features of a target object, we obtain the appearance model-1 and appearance model-2. Secondly, we learn the two sub-models (filters) with different learning rates, which correspond to h_t^1 and h_t^2 . During the tracking process, the two sub-models are used to produce response maps. Finally, the response maps of the two sub-models are weighted aggregated together to produce the final tracking result.

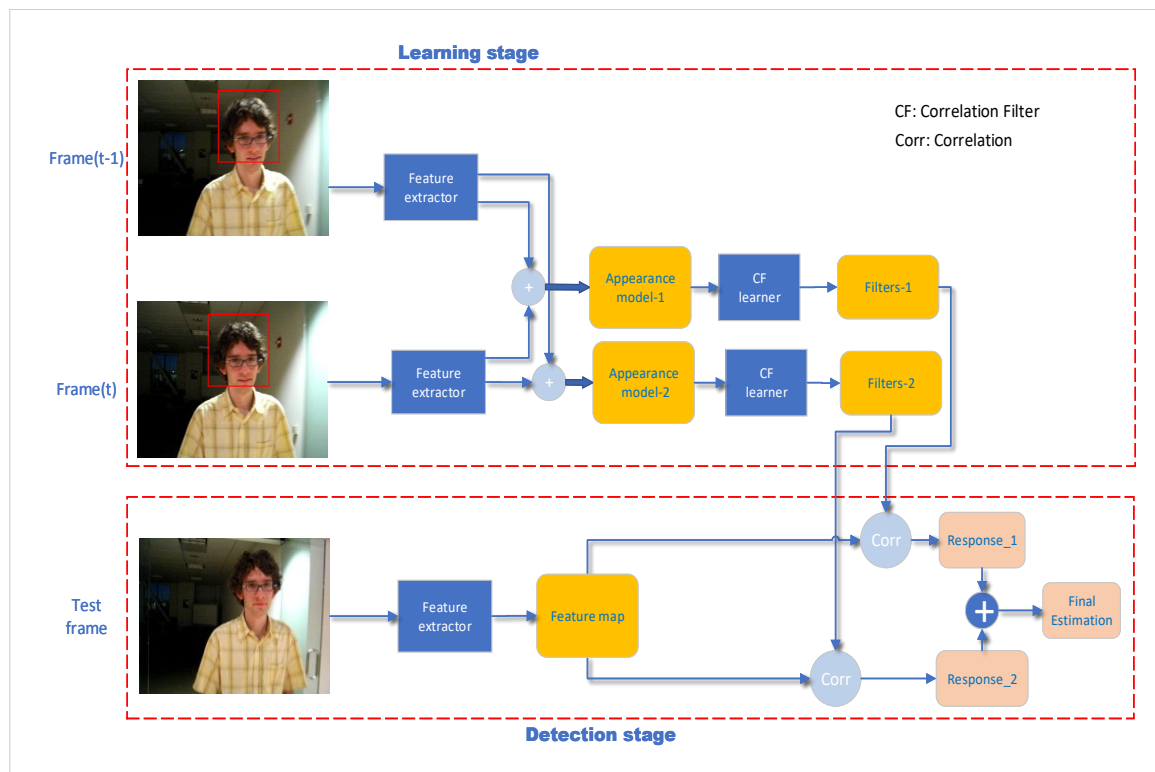


Figure 1. Overview of our visual tracking framework.

4.1. Adaptive Spatial-Temporal Regularization

Li et al. [6] introduced a temporal regularization term $\|f - f_{t-1}\|_2^2$ to make the filter learned from the current frame similar to the previous filter. In [11], Dai et al. proposed a spatially-regularized term $\|w - w^r\|_2^2$ to constrain the adaptive spatial weight w in the current frame be similar to a reference weight w^r . We found that both the temporal regularization and adaptive spatial regularization can be combined seamlessly to enhance the discriminative ability of filters. In this paper, we introduce an adaptive spatial-temporal regularization into the discriminative correlation filters framework.

In frame t , we prepare the training pair $\{x_t, y\}$, where x_t is the sampled image patch, y denotes its desired response generated with Gaussian function. By applying the adaptive

spatial-temporal regularization, the corresponding optimization problem can be defined as follows:

$$L = \arg \min_f \frac{1}{2} \left\| \sum_{d=1}^D \mathbf{x}_t^d \star (\mathbf{P}^\top f^d) - \mathbf{y} \right\|_2^2 + \frac{\lambda}{2} \sum_{d=1}^D \|\mathbf{w} \odot f^d\|_2^2 + \frac{\mu_1}{2} \|\mathbf{w} - \mathbf{w}^r\|_2^2 + \frac{\mu_2}{2} \|f - f_{t-1}\|_2^2 \quad (6)$$

where f_{t-1} denotes the learned CFs of the $t-1$ th frame. f is the CFs of the current frame to be learned, $\|\mathbf{w} \odot f^d\|_2^2$ and $\|\mathbf{w} - \mathbf{w}^r\|_2^2$ are the two spatial regularizers. \mathbf{P} is the cropping matrix as in BACF [10]. $\|f - f_{t-1}\|_2^2$ denotes the temporal regularizer. λ_1 , λ_2 , and μ are the three different penalty factors, respectively. The \top operator computes the transpose of the matrix. For a complex vector or matrix, it computes the conjugate transpose of them.

By applying the Parseval theorem, the above objective function can be expressed in the frequency for the computation efficiency as:

$$L = \arg \min_{\hat{f}} \frac{1}{2} \left\| \sum_{d=1}^D \text{conj}(\hat{\mathbf{x}}^d) \odot \hat{\mathbf{g}}^d - \hat{\mathbf{y}} \right\|_2^2 + \frac{\lambda}{2} \sum_{d=1}^D \|\mathbf{w} \odot f^d\|_2^2 + \frac{\mu_1}{2} \|\mathbf{w} - \mathbf{w}^r\|_2^2 + \frac{\mu_2}{2} \|\hat{\mathbf{g}} - \hat{\mathbf{g}}_{t-1}\|_2^2, \quad (7)$$

$$\text{s.t., } \hat{\mathbf{g}}^d = \sqrt{\text{TFP}}^\top f^d, \quad d = 1, \dots, D$$

Equation (7) can be solved with an Augmentation Lagrangian Method (ALM) as:

$$L = \arg \min_{\hat{f}} \frac{1}{2} \left\| \sum_{d=1}^D \text{conj}(\hat{\mathbf{x}}^d) \odot \hat{\mathbf{g}}^d - \hat{\mathbf{y}} \right\|_2^2 + \frac{\lambda}{2} \sum_{d=1}^D \|\mathbf{w} \odot f^d\|_2^2 + \sum_{d=1}^D (\hat{\boldsymbol{\zeta}}^d)^\top (\hat{\mathbf{g}}^d - \sqrt{\text{TFP}}^\top f^d) + \frac{\mu_1}{2} \|\mathbf{w} - \mathbf{w}^r\|_2^2 + \frac{\mu_2}{2} \|\hat{\mathbf{g}} - \hat{\mathbf{g}}_{t-1}\|_2^2 + \frac{\gamma}{2} \sum_{d=1}^D \|\hat{\mathbf{g}}^d - \sqrt{\text{TFP}}^\top f^d\|_2^2 \quad (8)$$

where the $\hat{\boldsymbol{\zeta}}$ and γ denote the Fourier Transform of the Lagrangian vector and penalty factor, respectively. Inspired from the STRCF [6], we introduce an auxiliary variable $\mathbf{h} = \frac{1}{\gamma} \hat{\boldsymbol{\zeta}}$, Equation (8) can be reformulated as,

$$L = \arg \min_{\hat{f}} \frac{1}{2} \left\| \sum_{d=1}^D \text{conj}(\hat{\mathbf{x}}^d) \odot \hat{\mathbf{g}}^d - \hat{\mathbf{y}} \right\|_2^2 + \frac{\lambda}{2} \sum_{d=1}^D \|\mathbf{w} \odot f^d\|_2^2 + \frac{\gamma}{2} \sum_{d=1}^D \|\hat{\mathbf{g}}^d - \sqrt{\text{TFP}}^\top f^d + \hat{\mathbf{h}}^d\|_2^2 + \frac{\mu_1}{2} \|\mathbf{w} - \mathbf{w}^r\|_2^2 + \frac{\mu_2}{2} \|\hat{\mathbf{g}} - \hat{\mathbf{g}}_{t-1}\|_2^2 \quad (9)$$

Then, the ADMM Algorithm [42] is adopted alternatively to solve the objective function.

Subproblem f:

$$f^d = \arg \min_{f^d} \frac{\lambda}{2} \|W f^d\|_2^2 + \frac{\gamma}{2} \|\hat{\mathbf{g}} - \sqrt{\text{TFP}}^\top f^d + \hat{\mathbf{h}}^d\|_2^2 \quad (10)$$

where W denotes the diagonal matrix: $W = \text{diag}(\mathbf{w})$. Taking the derivative of Equation (10) with respect to f^d and setting it to be zero, we can obtain the solution for f^d ,

$$f^d = (\lambda W^\top W + \gamma \text{TP})^{-1} Q, \quad Q = \gamma \text{TP}(\mathbf{g} + \mathbf{h}) \quad (11)$$

where g and h can be obtained by applying the inverse DFT \mathcal{F}^{-1} on \hat{g} and \hat{h} , i.e., $g = \mathcal{F}^{-1}\{\hat{g}\}$, $h = \mathcal{F}^{-1}\{\hat{h}\}$.

Subproblem g:

$$\begin{aligned}\hat{g}^* = \arg \min_{\hat{g}} & \frac{1}{2} \sum_{d=1}^D \|\text{conj}(\hat{x}^d) \odot \hat{g}^d - \hat{y}\|_2^2 \\ & + \frac{\gamma}{2} \sum_{d=1}^D \|\hat{g}^d - \sqrt{\text{TFP}}^\top f^d + \hat{h}^d\|_2^2 \\ & + \frac{\mu_2}{2} \|\hat{g}^d - \hat{g}_{t-1}^d\|_2^2\end{aligned}\quad (12)$$

As the j th element of \hat{y} is dependent only on the j th element across the D channels of the filters \hat{g}^d and samples \hat{x}^d , $d = 1, \dots, D$. We denote $\mathcal{V}_j(\hat{a}) = [\text{conj}(\hat{a}^1(j)), \dots, \text{conj}(\hat{a}^D(j))]^\top$ the vector of j th element of \hat{a} along all D channels. Thus, solving Equation (12) equals to solve T subproblems as:

$$\begin{aligned}\hat{g}(t)^* = \arg \min_{\hat{g}} & \frac{1}{2} \|\mathcal{V}_j(\hat{x}_t)^\top \mathcal{V}_j(\hat{g}) - \hat{y}_j\|_2^2 \\ & + \frac{\gamma}{2} \|\mathcal{V}_j(\hat{g}) + \mathcal{V}_j(\hat{h}) - \mathcal{V}_j(\sqrt{\text{TFP}}^\top f)\|_2^2 \\ & + \frac{\mu_2}{2} \|\mathcal{V}_j(\hat{g}) - \mathcal{V}_j(\hat{g}_{t-1})\|_2^2\end{aligned}\quad (13)$$

Then, we can obtain the solution of $\hat{g}(t)^*$ as:

$$\hat{g}(t)^* = (\mathcal{V}_j(\hat{x}_t)\mathcal{V}_j(\hat{x}_t)^\top + (\gamma + \mu_2)\text{TI}_k)^{-1}M \quad (14)$$

where $M = \mathcal{V}_j(\hat{x}_t)\hat{y}_j + \gamma\text{TV}_j(\sqrt{\text{TFP}}^\top f) - \gamma\text{TV}_j(\hat{h}) + \mu_2\text{TV}_j(\hat{g}_{t-1})$. Then, Equation (14) can be solved by using the Sherman Morrison formula [43], $(A + uv^\top)^{-1} = A^{-1} - \frac{A^{-1}uv^\top A^{-1}}{1 + v^\top A^{-1}u}$, where uv^\top is a rank-1 matrix. $u \in \mathbb{R}^{T \times 1}$ and $v \in \mathbb{R}^{T \times 1}$ are two column vectors, respectively. Then, Equation (14) can be reformulated as:

$$\hat{g}(t)^* = \frac{1}{(\gamma + \mu_2)\text{T}} \left(I_k - \frac{\mathcal{V}_j(\hat{x})\mathcal{V}_j(\hat{x})^\top}{(\gamma + \mu_2)\text{T} + \mathcal{V}_j(\hat{x})^\top \mathcal{V}_j(\hat{x})} \right) M \quad (15)$$

Subproblem w: The solution of spatial regularization weight w can be obtained by solving the following object function,

$$\begin{aligned}w^* &= \arg \min_w \frac{\lambda}{2} \sum_{d=1}^D \|\text{diag}(w)f^d\|_2^2 + \frac{\mu_1}{2} \|w - w^r\|_2^2 \\ &= \arg \min_w \frac{\lambda}{2} \sum_{d=1}^D \|\text{diag}(f^d)w\|_2^2 + \frac{\mu_1}{2} \|w - w^r\|_2^2 \\ &= (\lambda \sum_{d=1}^D (\text{diag}(f^d)^\top \text{diag}(f^d)) + \mu_1 I)^{-1} \mu_1 w^r\end{aligned}\quad (16)$$

Updating the Lagrangian Multiplier: We update the Lagrangian vector as:

$$\hat{h}^{i+1} = \hat{h}^i + \gamma(\hat{g}^{i+1} - \hat{f}^{i+1}) \quad (17)$$

where the penalty factor γ is updated as:

$$\gamma = \min(\gamma_{\max}, \beta\gamma) \quad (18)$$

Model Update: Instead of the online update strategy which updates the tracking model at each frame in most of the traditional DCF trackers [2,10,13], we employ the criterion called Peak to Sidelobe Ratio (PSR) [2] to measure the tracking uncertainty and decide when to update the tracking result in upcoming frames. Generally, the larger PSR value indicates the better location accuracy.

4.2. 2-Dimensional Scale Space Search Method

In visual object tracking, trackers often lose the target due to non-rigid deformation, especially for the most of DCFs trackers, which utilize the 1-dimensional scale estimation approach [3]. In this method, S number of scales with a scale factor a are used to extract an image patch J_n of size $a^n P \times a^n P$ centered around the target, where $n \in \left\{ -\lfloor \frac{S-1}{2} \rfloor, \dots, \lfloor \frac{S-1}{2} \rfloor \right\}$. During online tracking, the tracking algorithms learn their appearance models and perform tracking with initialization in the first frame.

The aspect ratio is fixed in this scale estimation approach. For example, if the best scale factor is a^* , the width and height become $[a^* \times W_{t-1}, a^* \times H_{t-1}]$, where W_{t-1} and H_{t-1} are the width and height of the target in the previous frame. The limitation of this scheme is that the width W_{t-1} and H_{t-1} are scaled with a same factor a^* . Since the new estimated state cannot cover the appearance variance of target due to severe non-rigid deformation, the variance of width and height are not the same with each other.

As shown in Figure 2, the size of the ground truth bounding boxes of target objects in raw frames are $W_1 = 133$, $H_1 = 146$, $W_2 = 90$, $H_2 = 132$, $W_3 = 95$, and $H_3 = 93$, respectively. Then, the ratios are: $a_w^1 = W_2/W_1 = 0.676$, $a_h^1 = H_2/H_1 = 0.904$, $a_w^2 = W_3/W_2 = 1.056$, and $a_h^2 = H_3/H_2 = 0.705$. The different ratios of height and width illustrate that applying an identical scale factor to scale estimation is not enough to capture the shape variation.

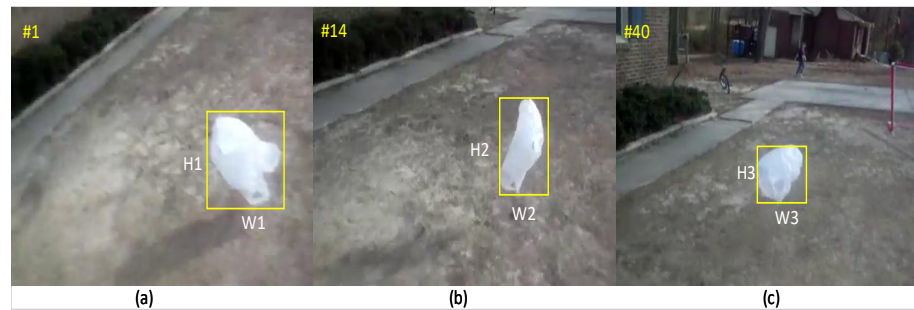


Figure 2. Examples of changes in terms of the aspect ratio. Frames are selected from the *bag* sequence in the VOT2018 dataset. (a) The frame #1; (b) The frame #14; (c) The frame #40.

To adapt to shape changes, we proposed a novel scale space search method, which allows the scale factors to change along two axis directions. As shown in Figure 3, the figure above is the scale variation in conventional trackers that use the 1-dimensional scale space search method, and the figure below is scale variation in our new method. Let the S denotes the size of scale filter, for each $m, n \in \left\{ -\lfloor \frac{S-1}{2} \rfloor, \dots, \lfloor \frac{S-1}{2} \rfloor \right\}$, we extract an image patch $J_{m,n}$ of size $a^m P \times a^n R$ centered around the previous target location for tracking, where a^m and a^n denote scale factors along two dimensions as shown in the Figure 3.

Compared to the 1-dimensional scale space search method, our new scale search method explores different aspect ratios along two dimensions, aiming to capture severe non-rigid deformation of the target. Afterwards, the final translations and scale factors are computed by Newton's method as in SRDCF [4].

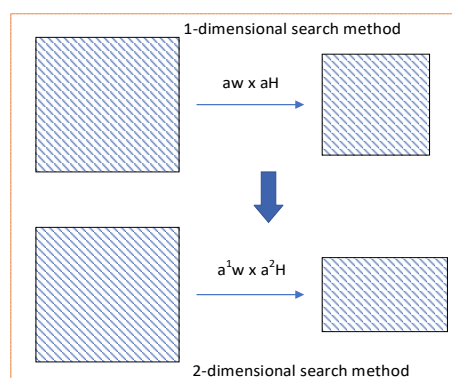


Figure 3. Comparisons of aspect ratios with different search strategies.

4.3. The Impact of Learning Rate

In traditional CF-based visual tracking methods, the learning rate is a common parameter that is used for updating the appearance model. In our framework, if we set the learning rate to 1, then the appearance model will degenerate to STRCF [6], which directly feeds the current appearance feature to the learning model. Through the experiments, we found that the learning rate also has a significant impact on the tracking accuracy of the model. For example, the *Car24* sequence in OTB-100 [18] consists of more than 3k frames, and the appearance of the *Car* changes slightly when it moves from right side to left side of the road or from far to near as in the examples we present in Figure 4. The larger learning rate implies that the appearance model updates rely more the recent features and less the previous features.

Different from the learning rate in the domain of machine learning and deep learning, which can be adjusted by the model online, we do not have the fully labeled training data to find the optimal hyper parameter, such as the learning rate. In our framework, we utilize the learning rate to update the appearance model, and the main difference is that there are two kinds of learning rates employed in our model, corresponding to two different sub-models as shown in Figure 1.

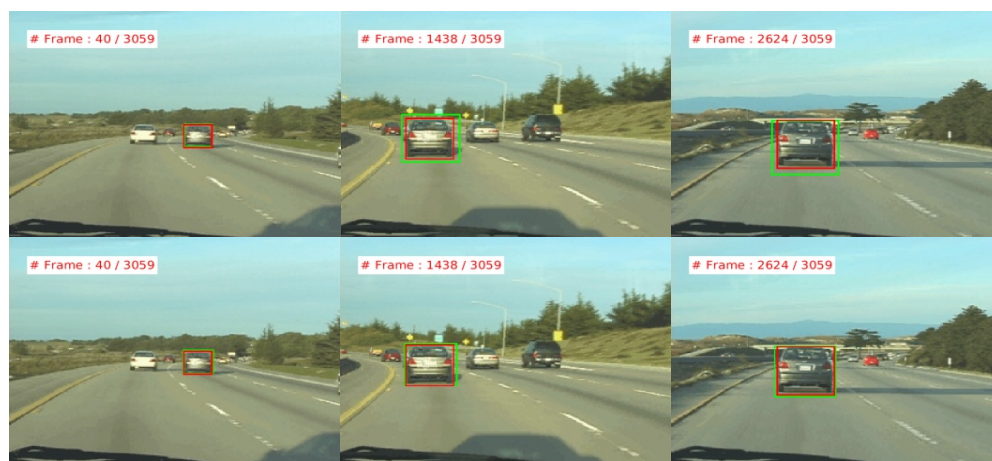


Figure 4. Tracking examples with different learning rates on *Car24* sequence, where the results in the upper row use larger learning rate and the results in the bottom row use relative small learning rate. The ground truth bounding boxes are in red.

5. Experiments

5.1. Experimental Implementations

Our approach ASTRCF is implemented with Matlab 2019b using MatConvNet on a PC with an i5-7500 CPU at 3.4 GHz, 64GB RAM, and a NVIDIA Geforce 1080ti GPU. Algorithm 1 provides a brief outline of our tracker. In order to fuse the deep and hand-

crafted feature in a proper way for filter learning, we set the standard deviation for the desired correlation output to $1/16$ of the target size as the same with [3]. The initial penalty factor γ is set as 1. We set the maximal penalty factor γ_{max} as 10,000. We set the step β as 10 to update γ by $\gamma^{i+1} = \min(\gamma_{max}, \beta\gamma^i)$. The spatial regularization factor μ_1 is set to 0.001.

Algorithm 1 The proposed tracking algorithm: iteration at time step t

Inputs: Images $\{I_t\}_{t=1}^T$, object position in previous frame p_1 , scale estimation on previous frame s_{t-1} , filters h_{t-1}^1 and h_{t-1}^2 .

Outputs: updated correlation filters h_t^1 and h_t^2 , estimated target position p_t , estimated scale s_t .

- 1: **repeat**
 - 2: extract the feature map x_t based on previous target position p_{t-1} .
 - 3: estimate the current pos p_t and scale s_t using the CFs h_{t-1}^1, h_{t-1}^2 by the 2-dimensional scale space search method.
 - 4: extract samples z_t based on current target position p_t .
 - 5: update models of h_t^1 and h_t^2 based on samples z_t and previous models of h_{t-1}^1 and h_{t-1}^2 .
 - 6: **until** end of the sequence.
-

The temporal regularization factor μ_2 is set to 15 and 8 for the two sub-models, respectively. The learning rates are set to 0.0185 and 0.003 for two sub-models, respectively. The number of scales S is set to 3 and the scale step a is set to 1.05. Then, we extract 9 image patches during tracking in our algorithm. For target representation, we combined hand-crafted features (i.g., HOG and Color-Names) with 27-th (relu-4_4) and 36-th (relu-5_4) layers in the Vgg19 network [44] for model learning.

5.2. Evaluation Methodology

We use precision and success plots as [18] to measure the performance of the trackers. The precision plot shows the percentage of frames for which the distance between estimated location and ground truth is within the specific threshold. The success plot indicates how many times the target is successfully tracked among all frames of a sequence within the IoU (Intersection-over-Union) over the estimated bounding box and the ground truth.

The VOT2016 and VOT2018 datasets evaluate the trackers with three metrics: (1) Accuracy measures the intersection over union between the predicted bounding box and ground truth. (2) Robustness measures the mean number of failures per sequence. (3) The Expected Average Overlap (EAO) integrates the accuracy and robustness and is computed as the average of the expected average overlap over a selected range of sequence lengths.

5.3. Ablation Study

An ablation study was conducted on the OTB-100 [18] dataset to validate the effectiveness of our proposed framework. For a fair comparison, we used deep features include layer-79 from ResNet-101 [45], layer-14 from VggM-2048 [17], layer-27 and layer-36 from Vgg-19 [44], and hand-crafted feature (i.g., HOG) for different trackers. First, we investigate the robustness of deeper features for dealing with the drastic deformation; secondly, we follow the same features setting to fairly compare our framework with the two baseline trackers (e.g., STRCF [6] and ASRCF [11]), conducting the challenging sequences in Table 1.

We also employ two kinds of deep features on STRCF [6] and ASRCF [11]. First, we found that more robust features (e.g., feature maps of layer-79 in ResNet101) improved the performance of STRCF [6], while not significant for ASRCF [11]. For example, the AUC success of *Ironman* sequence with deep features of ResNet101-79 layer decreases to 0.125 from 0.436 for ASRCF [11]. Secondly, through the *Singer2* sequence, we found that the low level features and high semantic features contributed differently to the tracking performance.

In such a scenario, the deep semantic features degraded the tracker's discriminative ability and make the tracker suffer from the background clutter. In order to improve

the robustness of our tracker, we constructed a collaborative tracking framework, which includes a shallow model and a deep model, operating as a parallel way during the tracking process. For the *MotorRolling* sequence, the experimental results show that both the STRCF [6] and ASRCF [11] lose the target with relatively low-level features (e.g., HOG and Vgg16-23). While our tracker with the deeper features from ResNet101 could successfully handle this problem and can achieve promising performance as the Table 1 shown.

Table 1. Average Overlap on challenging sequences of OTB-100 with different model settings.

Tracker	Ironman	MotorRolling	Matrix	Skater2	Singer2	Soccer	KitSurf	Skiing	Skater	Trans	Models
STRCF	0.117	0.094	0.010	0.639	0.039	0.189	0.716	0.088	0.601	0.570	HC+VggM-14
STRCF	0.096	0.154	0.494	0.635	0.764	0.201	0.561	0.099	0.635	0.578	HC+ResNet101-79+VggM-14
ASRCF	0.436	0.117	0.448	0.614	0.765	0.502	0.733	0.513	0.494	0.546	HC+Vgg16-23+VggM-4
ASRCF	0.125	0.119	0.441	0.610	0.788	0.449	0.712	0.515	0.537	0.563	HC+ResNet101-79+VggM-14
Ours	0.515	0.117	0.597	0.588	0.040	0.572	0.743	0.512	0.574	0.551	HC+Vgg16-23+VggM-4
Ours	0.531	0.525	0.581	0.611	0.747	0.477	0.688	0.494	0.600	0.552	HC+ResNet101-79+VggM-14
Ours	0.406	0.583	0.573	0.645	0.739	0.551	0.777	0.574	0.677	0.631	2D search method

In addition, if we increase the learning rate, then our model can obtain large improvements on sequences of the challenging attributes, for example, fast moving, illumination variation, and in-plane rotation. The results of our final model (HC+Vgg19-27+Vgg19-36) with 2-dimensional scale space search method are presented at the last line in Table 1, which outperform the other models by a large margin on these challenging sequences.

5.4. State-of-the-Art Comparison

We evaluated our framework with the selected state-of-the-art trackers, including KCF [13], DeepSRDCF [4], CCOT [7], MDNet [46], ECO [12], ECO_HC [12], DeepSTRCF [6], MCPF [24], ASRCF [11], ACFN [47], CSR-DCF [29], Staple [48], SAMF [49], MEEM [50], LSART [51], SiamRPN [38], SiamRPN++ [8], ATOM [39], and DiMP50 [40] on OTB [18], VOT-2016 [19], VOT-2018 [20], and LaSOT [21] datasets. To further illustrate the effectiveness of our approach, we also present qualitative results on these tracking datasets in Section 5.4.2.

5.4.1. Quantitative Evaluation

OTB-100 Dataset. OTB-100 [18] is an online tracking benchmark that contains 98 fully annotated sequences with 100 different target objects. Each of the *Skating2* sequence and *Jogging* sequence contains two annotated targets. The OTB-50 dataset [18] contains 49 sequences with 50 targets selected from the OTB-100 dataset. The detail tracking results are presented in Figure 5 with precision and robustness plots. Our proposed approach outperforms most of the state-of-the-art trackers, achieving an AUC score of 70.0% and a precision score of 93.6%.

Compared to DeepSTRCF, our approach achieves a gain of 2.3% AUC score and 5.3% precision score, respectively. Our approach outperforms the ASRCF by 0.8% in AUC and 1.4% in precision, respectively. Compared to other recently proposed deep neural network based trackers, such as ATOM [39], DiMP [40], and SiamRPN++ [8] which obtain AUC scores of 66.0%, 67.8%, and 69.7%, respectively, our method also achieves a competitive tracking performance, with a relative gain of 4%, 2.2%, and 0.3% over them.

We present the quantitative results on different attributes of OTB-100 dataset in terms of precision and success, corresponding to Figures 6 and 7, respectively. The precision and success plots in Figures 6 and 7 illustrate that our approach outperforms other trackers on most of the attributes, such as the precision scores on deformation, fast motion, illumination variation, out-of-plane rotation, out-of-view, and low resolution attributes, and the AUC scores on background clutter, illumination variation, out-of-plane rotation, low resolution, and out of view attributes.

From the tracking results from Figures 5–7, we can conclude that our approach achieves state-of-the-art performance compared with others on OTB-100 dataset. The

success plots shows that our approach estimates scale more accurately on these dataset. For example, our tracker outperforms ASRCF [11] and DeepSTRCF [6] by a gain of 2.6% and 8.4% in success score on the attribute of in-plane rotation, respectively.

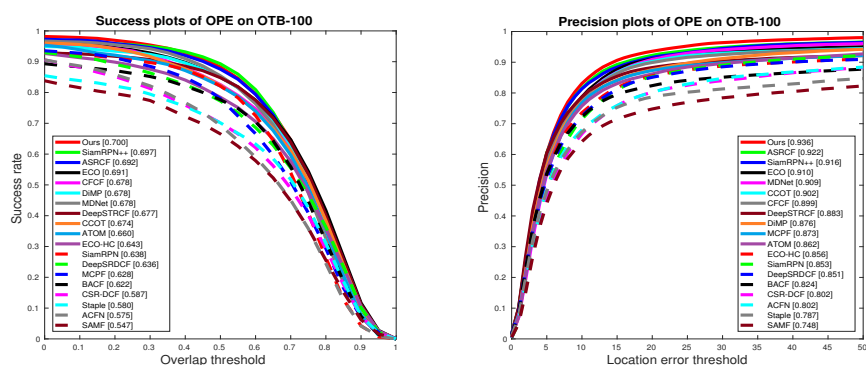


Figure 5. The overall performance of precision and success on OTB-100 [18] dataset using one-pass evaluation (OPE).

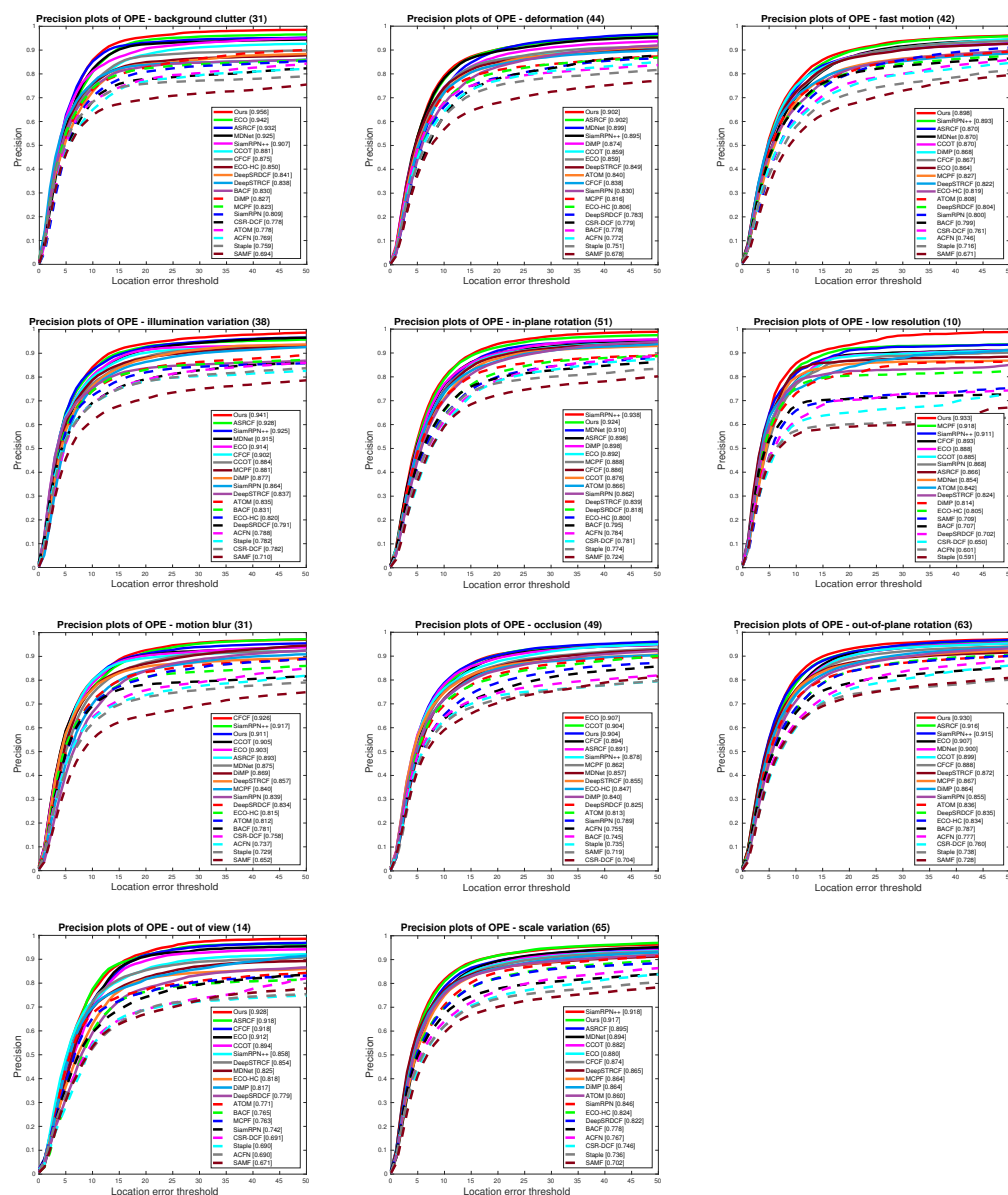


Figure 6. The precision plots on each attribute on OTB-100 [18] dataset.

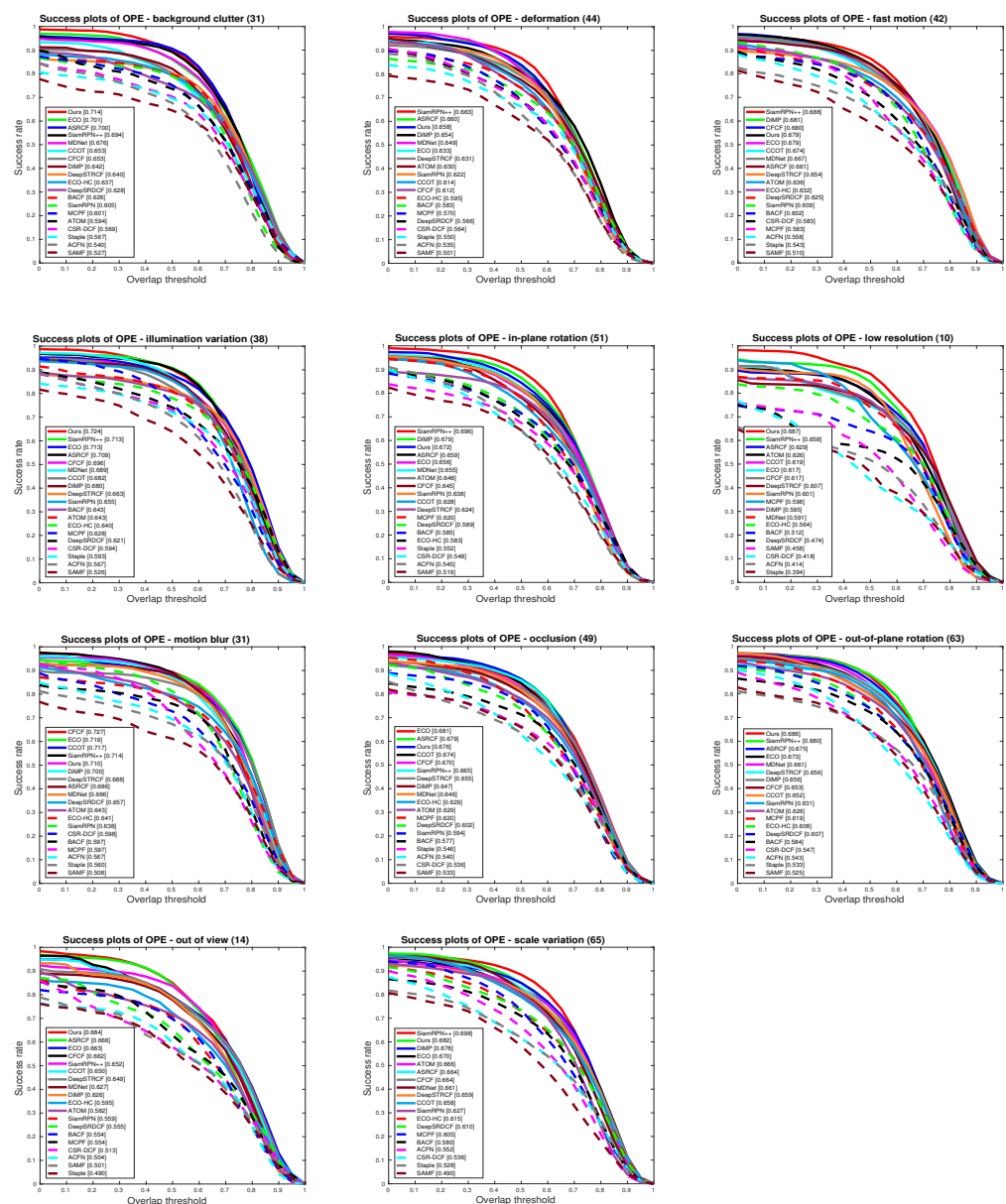


Figure 7. The success plots on each attribute on OTB-100 [18] dataset.

VOT Datasets. VOT-2016 [19] consists of 60 short sequences selected from a large pool of 356 sequences. The sequence selection method is extended to be fully automated compared to VOT2014, and the selection process focus on those sequences that are more likely challenging for tracking. VOT-2018 dataset [20] maintains the same object classes with VOT2016 but the least challenging sequences from latter were replaced by new sequences. The evaluation criteria in VOT-2018 is same as that in VOT-2016. We compare our approach with 20 state-of-the-art trackers. Table 2 shows the comparison results of different trackers on VOT-2016 and VOT-2018 datasets. Our approach outperforms most of the trackers in terms of Av, Rv, and EAO metrics. Our approach obtains an EAO score of 33.9%.

Table 2. Performance comparison among the state-of-the-art trackers on VOT 2016, and VOT 2018. The results are presented in terms of EAO, Av and Rv. The top three results are marked in red, blue, and green fonts, respectively.

Trackers	VOT2016			VOT2018		
	Av	Rv	EAO	Av	Rv	EAO
Ours	0.528	0.177	0.402	0.507	0.225	0.339
ASRCF [11]	0.568	0.187	0.390	0.492	0.234	0.328
DeepSTRCF [6]	0.55	0.257	0.313	0.531	0.272	0.299
ATOM [39]	0.617	0.189	0.424	0.589	0.201	0.401
SA-SIAM [52]	0.543	0.337	0.291	0.5	0.459	0.236
CSR-DCF [29]	0.524	0.239	0.338	0.491	0.356	0.256
CFCF [53]	0.560	0.169	0.384	0.511	0.286	0.283
CCOT [7]	0.541	0.239	0.331	0.494	0.318	0.267
SiamRPN [38]	0.56	0.314	0.340	0.490	0.464	0.244
SiamRPN++ [8]	0.637	0.178	0.478	0.6	0.234	0.414
ECO [12]	0.54	0.201	0.374	0.484	0.276	0.281
ECO_HC [12]	0.53	0.304	0.322	0.494	0.435	0.238
DSST [54]	0.535	0.707	0.181	0.395	1.452	0.079
LSART [51]	0.495	0.215	0.323	0.495	0.276	0.323
MEEM [50]	0.490	0.515	0.194	0.463	0.534	0.193
Staple [48]	0.547	0.379	0.295	0.530	0.688	0.169
SRDCF [4]	0.536	0.421	0.247	0.490	0.974	0.119
DeepSRDCF [5]	0.507	0.326	0.276	0.492	0.707	0.154
SiamFC [36]	0.532	0.461	0.88	0.503	0.585	0.188
SAMF [49]	0.507	0.590	0.186	0.484	1.302	0.093
KCF [13]	0.491	0.571	0.192	0.447	0.773	0.135

In addition, our tracker outperforms ASRCF with a relative gain of 1.5% in Av and 1.1% in EAO on VOT-2018 dataset. Our approach improves the DeepSTRCF by 4% in EAO on VOT-2018 and by 8.9% in EAO on VOT-2016. Our approach also outperforms the deep Siamese trackers, such as SiamRPN [38] and SA-SIAM [52] in Rv and EAO. SiamRPN++ [8] obtains a high EAO score of 41.4%, with the Av score of 60%. The deep neural network based trackers, such as SiamRPN++ and ATOM, employ the bounding box regression branch to estimate the scale variation of the target, which promotes the IoU between estimated target state and its ground truth, comparing with the search strategy applied in the DCF trackers.

LaSOT Dataset. LaSOT [21] dataset contains 1400 sequences with 3.52 million frames and an average of 2512 frames per sequence, among which, the test set contains 280 sequences. There are 70 different object categories in the whole dataset and each category consists of 20 sequences. This dataset provides high-quality dense bounding box annotations for each sequence. Moreover, this dataset categorizes the sequences into 14 attributes, including aspect ratio change (ARC), background clutter (BC), camera motion (CM), deformation (DEF), fast motion (FM), full occlusion (FOC), illumination variation (IV), low resolution (LR), motion blur (MB), out-of-view (OV), partial occlusion (POC), rotation (ROT), scale variation (SV), and viewpoint change (VC).

As shown in Figure 8, our approach achieves the highest performance in terms both the robustness and accuracy compared to previous methods. Our approach obtains a gain of 5.7% and 9.3% in AUC over ASRCF [11] and DeepSTRCF [6], respectively. In addition, our approach obtains a gain of 6.8% and 10.1 % in precision compared with its counterparts ASRCF [11] and DeepSTRCF [6], respectively. Figures 9 and 10 show the success plots and precision plots on LaSOT with different attributes.

Our approach obtained improvements for all attributes in the precision score and success score compared to ASRCF [11] and DeepSTRCF [6]. Our approach outperforms its counterparts ASRCF and DeepSTRCF by a large gain of 5.7% and 5.8% in success on aspect ratio change attribute, respectively. In addition, Our approach outperforms its counterparts

ASRCF and DeepSTRCF by a gain of 7% and 5.6% in success on the deformation attribute, respectively. This illustrates the ability of our tracker to deal with challenging sequences.

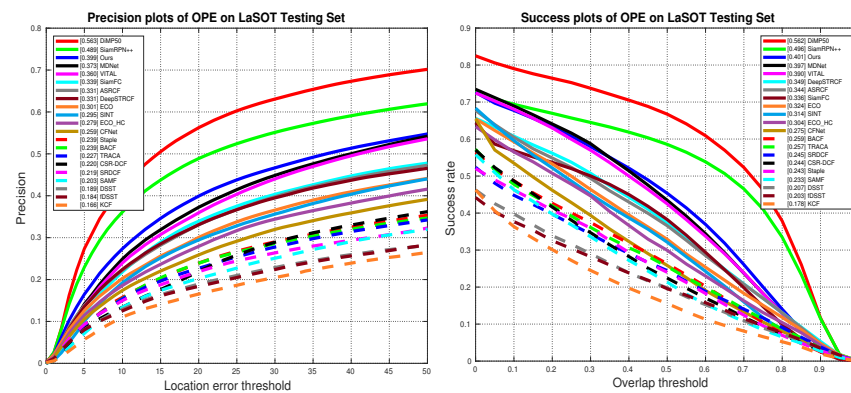


Figure 8. The overall performance of precision and success plots on the LaSOT dataset using one-pass evaluation (OPE).

Our approach still has a gap compared with the end-to-end trained deep tracker SiamRPN++ [8] and DiMP [40], which benefit from the large-scale offline training and bounding box regression network.

5.4.2. Qualitative Results

In this section, we present the qualitative tracking results of different trackers on the OTB-100, VOT-2018, and LaSOT datasets, corresponding to Figures 11–13, respectively. The qualitative results illustrate that our approach is capable of tracking scale changes and appearance deformation compared with other DCF trackers. In Figure 11, we observed that the target in the *MotorRolling* sequence flipped in plane, and the scale also changed dramatically over time. The STRCF [31] and ASRCF [11] both lost the target, while our approach can successfully track the target among most of the frames and obtain relative high AUC score. Furthermore, our tracker achieves promising performance and is on par with the ATOM tracker. For the example of *Trans* sequence, the target transformed from a robot to a car with the severe appearance changes, as the frames #16, #61, and #117 shown. The results of our tracker are closer to the ground truth than other trackers.

As shown in Figure 12, our tracker outperforms the DCF trackers and deep Siamese tracker SiamRPN++ [8] in frame #119 and #145 in sequence *fish1* when the posture of fish changes significantly. In *bag* sequence, our approach can adjust the width and height properly to adapt to the deformation of *bag*. In Figure 13, our approach can still capture the shape of the dog in frame #209 and frame #230, while most of the other DCF trackers have lost the target or merely tracked the part of dog. For the *cat-3* sequence, only our approach and SiamRPN++ [8] can successfully track the target object, which also illustrates the effectiveness of our method.

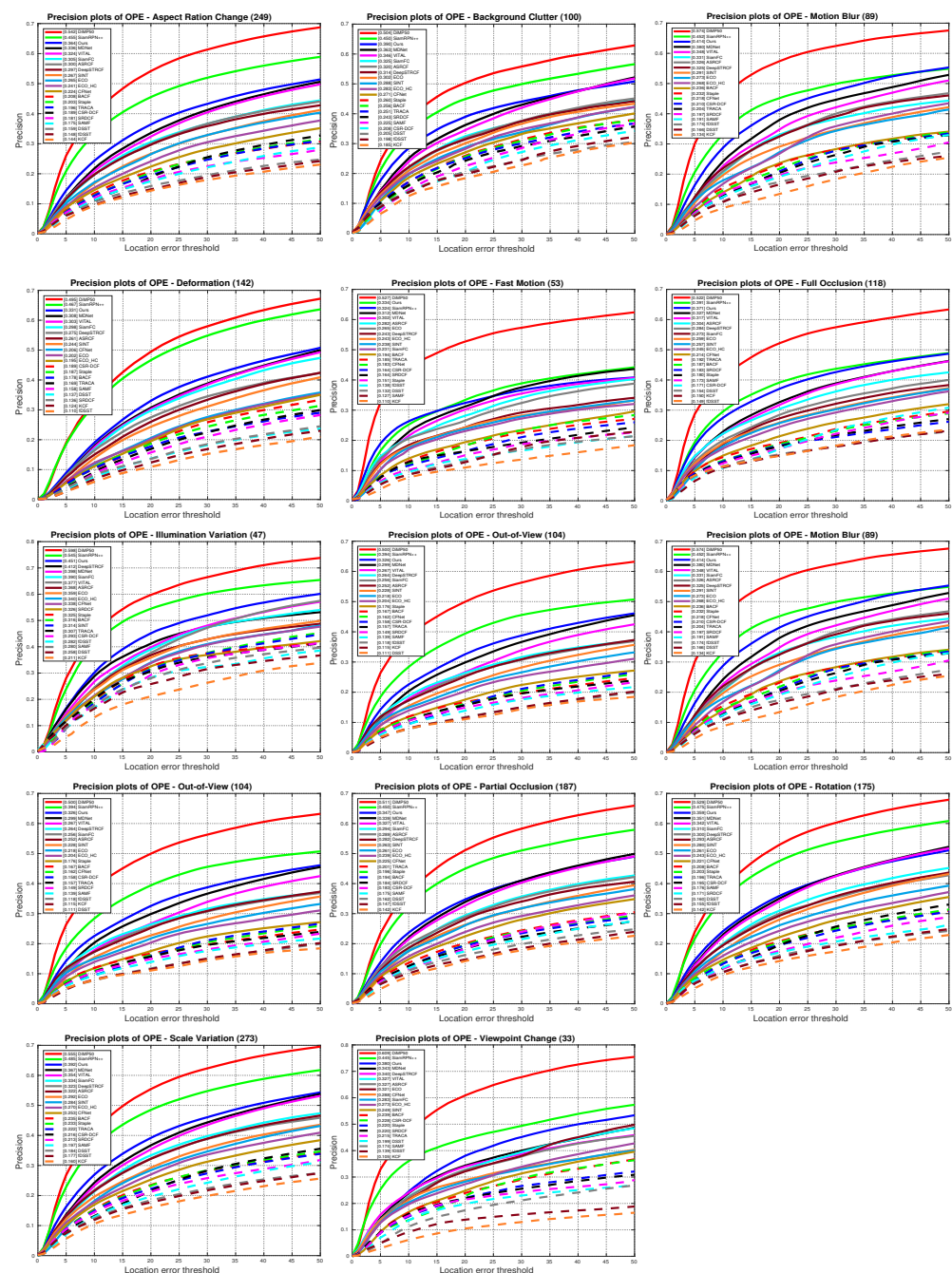


Figure 9. The precision plots on LaSOT [21] dataset for each attribute.

The OTB-100 dataset contains more sequences that with rigid target objects, such as car, box, and football. While VOT-2018 and LaSOT contain more sequences with non-rigid target objects. Our approach yields favorable performance on OTB-100 dataset with more stable sequences. However, the end-to-end trained deep trackers exhibits better performance on sequences with more deformable targets. The deep trackers employ deep convolutional network as the backbone to extract features, while our tracker utilizes both the HOG and deep features for correlation filters learning. SiamRPN++ [8] tends to lose the target which was occluded by similar distractor, as the *Walking2* sequence shown in Figure 11.

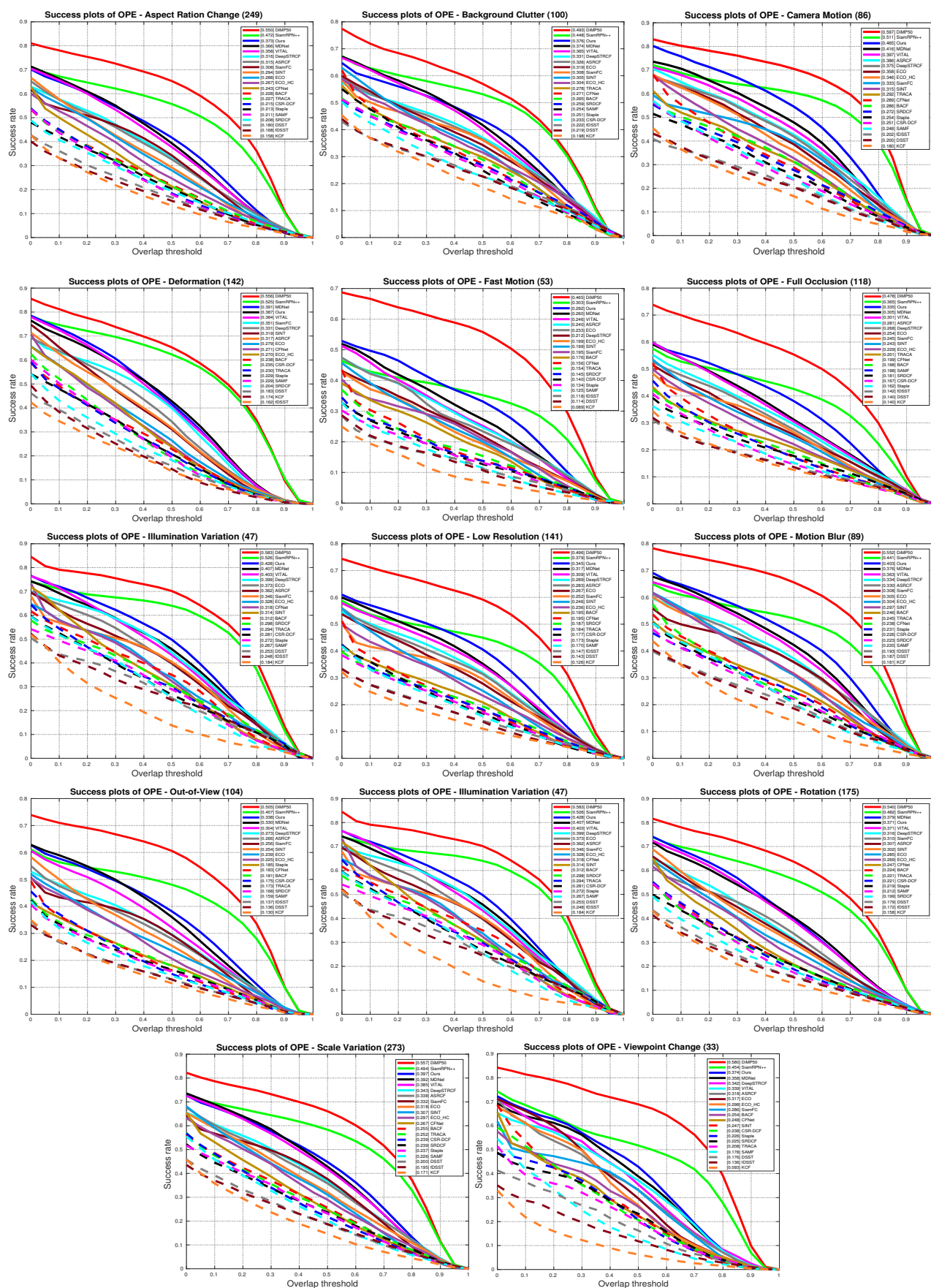


Figure 10. The success plots on LaSOT [21] dataset for each attribute.

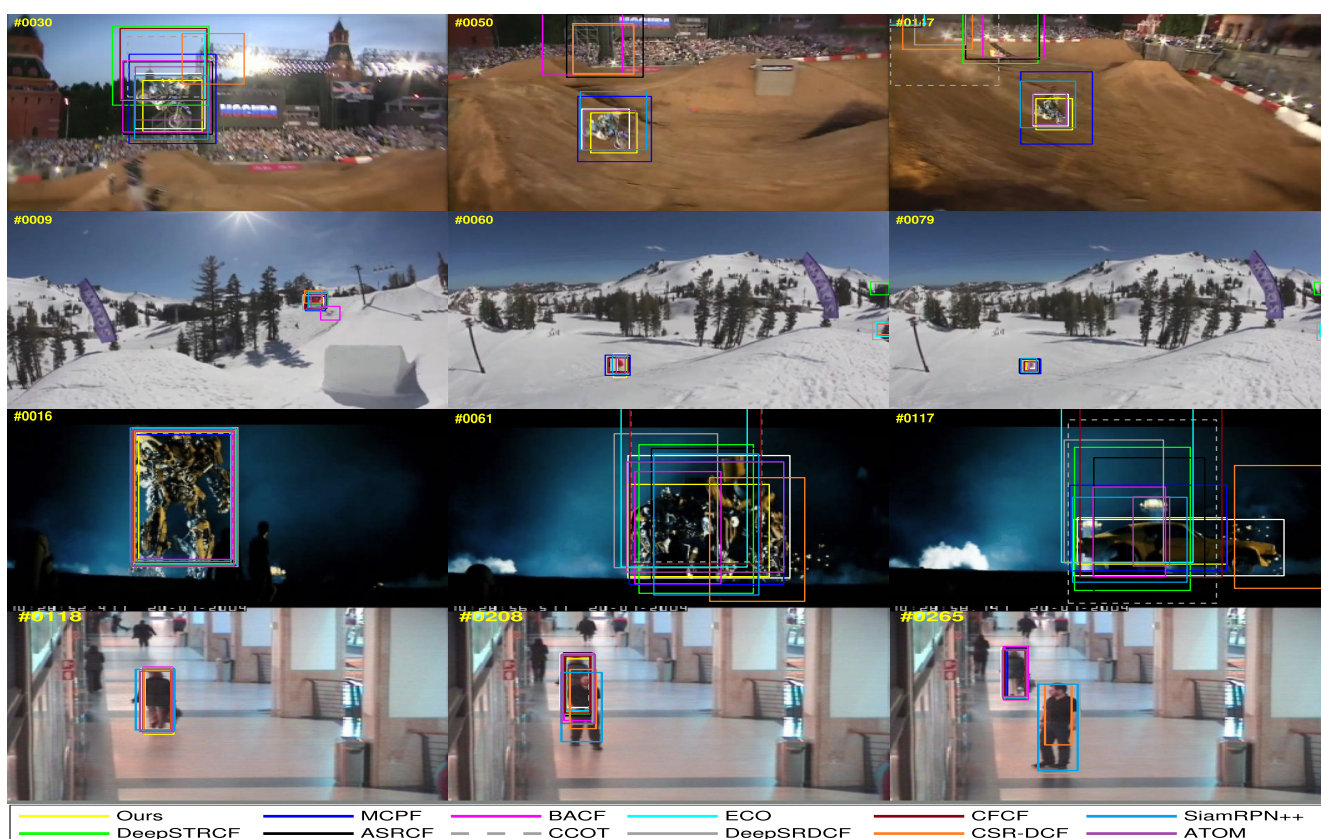


Figure 11. Qualitative results on sequences of *MotorRolling*, *Skiing*, *Trans*, and *Walking2* in OTB-100 dataset. The ground truth bounding boxes are in white.

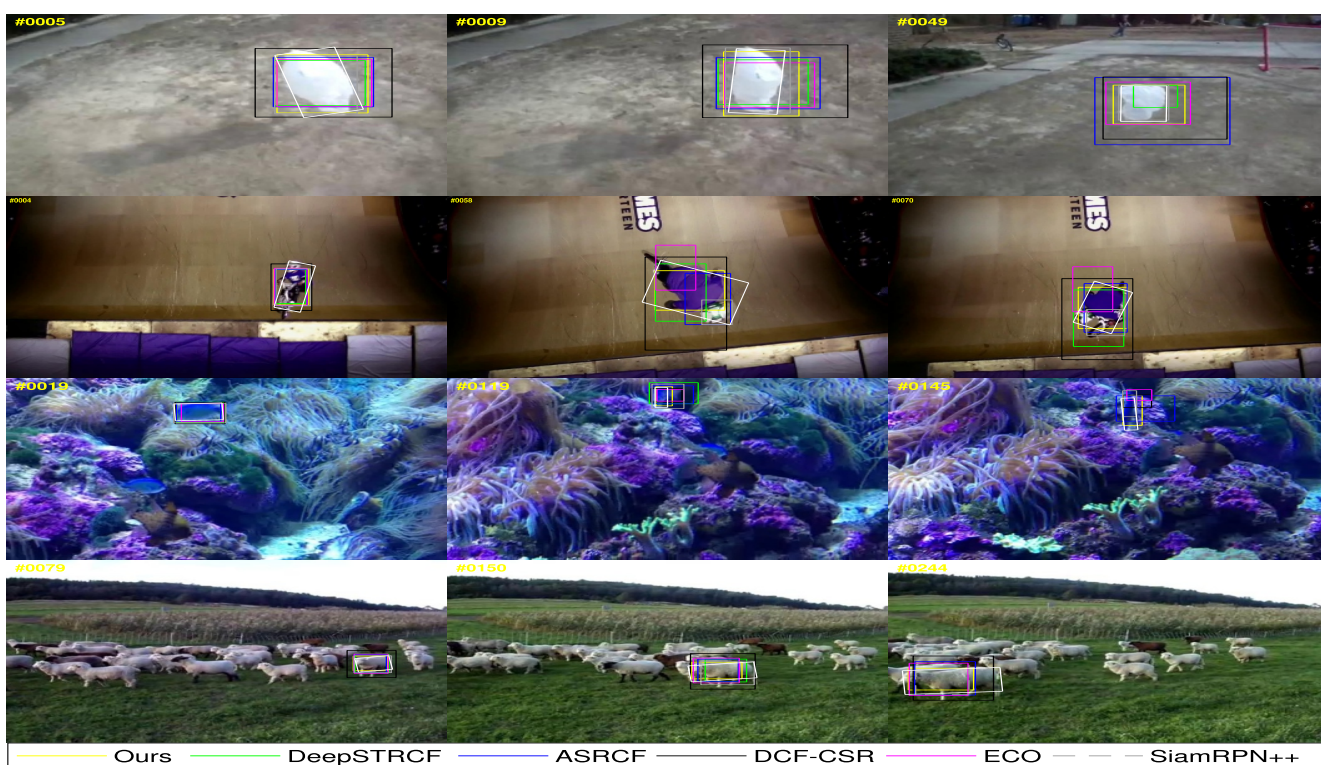


Figure 12. Qualitative results on sequences of *bag*, *bmx*, *fish1*, and *sheep* in VOT-2018 dataset. The ground truth bounding boxes are in white.

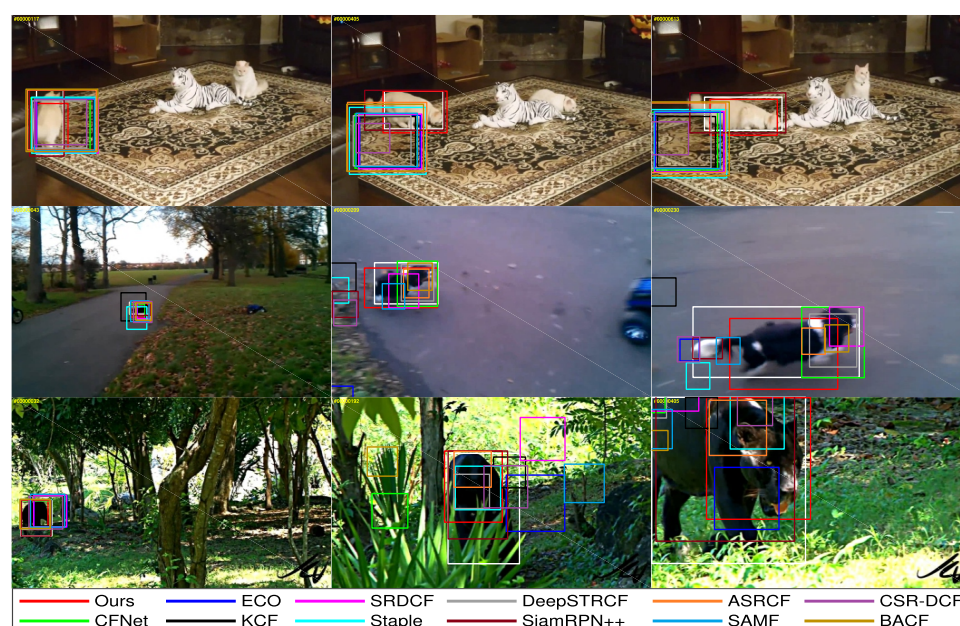


Figure 13. Qualitative results on sequences of *cat-3*, *dog-1*, and *leopard-1* in LaSOT dataset. The ground truth bounding boxes are in white.

6. Conclusions

In this paper, we proposed a new visual object tracking algorithm based on discriminative correlation filters. Our approach learns the discriminative correlation filters (DCF) via not only adaptive spatial regularization but also temporal regularization for estimating the target state. A 2-dimensional scale space search method was proposed to adapt to appearance variations of the target object and to improve the accuracy of the tracking results.

From the ablation study on OTB-100, we found that the features from deeper neural networks can improve the robustness of the tracker. We also analyzed the impact of different learning rates on learning process of the target model. Comprehensive experiments and analysis were performed on the OTB-100, VOT2016, VOT2018, and LaSOT datasets. The qualitative and quantitative results illustrated the effectiveness of our tracking framework.

Our tracking framework outperformed most of the existing state-of-the-art DCF trackers on challenging sequences and achieved competitive performance compared with end-to-end deep trackers. The results on challenging sequences of three datasets further demonstrate that our novel scale space search method can improve the accuracy of scale estimation.

In future research, we could apply trainable end-to-end deep convolutional neural networks to represent the target during learning of DCFs in the future. An interesting direction for future work is to incorporate bounding box regression technologies in many deep learning based detectors, such as [55], to further improve the performance.

Author Contributions: Investigation, F.C.; writing—original draft preparation, F.C.; writing—review and editing, X.W.; supervision, X.W.; funding acquisition, X.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Science and Technology Foundation of State Key Laboratory grant number 6142110180405.

Data Availability Statement: All datasets evaluated in the paper can be found on official websites, OTB-100: http://cvlab.hanyang.ac.kr/tracker_benchmark/datasets.html accessed on 1 September 2021, VOT-2016: <https://www.votchallenge.net/vot2016/> accessed on 1 September 2021, VOT-2018: <https://www.votchallenge.net/vot2018/dataset.html> accessed on 1 September 2021, LaSOT: <https://cis.temple.edu/lasot/> accessed on 1 September 2021. In addition, all data generated during the study are available from the corresponding author by request.

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

References

1. Pau, G.; Arena, F.; Gebremariam, Y.E.; You, I. Bluetooth 5.1: An Analysis of Direction Finding Capability for High-Precision Location Services. *Sensors* **2021**, *21*, 3589. [[CrossRef](#)] [[PubMed](#)]
2. Bolme, D.S.; Beveridge, J.R.; Draper, B.A.; Lui, Y.M. Visual Object Tracking using Adaptive Correlation Filters. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 13–18 June 2010; pp. 2544–2550.
3. Danelljan, M.; Häger, G.; Khan, F.; Felsberg, M. Accurate scale estimation for robust visual tracking. In Proceedings of the British Machine Vision Conference (BMVC), Nottingham, UK, 1–5 September 2014.
4. Danelljan, M.; Hager, G.; Shahbaz Khan, F.; Felsberg, M. Learning Spatially Regularized Correlation Filters for Visual Tracking. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 11–18 December 2015; pp. 4310–4318.
5. Danelljan, M.; Hager, G.; Khan, F.S.; Felsberg, M. Convolutional Features for Correlation Filter Based Visual Tracking. In Proceedings of the IEEE International Conference on Computer Vision Workshop, Santiago, Chile, 7–13 December 2015; pp. 621–629.
6. Li, F.; Tian, C.; Zuo, W.; Zhang, L.; Yang, M.H. Learning spatial-temporal regularized correlation filters for visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 4904–4913.
7. 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 (ECCV), Amsterdam, The Netherlands, 8–16 October 2016; pp. 472–488.
8. 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 (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 4282–4291.
9. Mueller, M.; Smith, N.; Ghanem, B. Context-aware Correlation Filter Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 1387–1395.
10. Galoogahi, H.K.; Fagg, A.; Lucey, S. Learning Background-aware Correlation Filters for Visual Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 21–26.
11. Dai, K.; Wang, D.; Lu, H.; Sun, C.; Li, J. Visual Tracking via Adaptive Spatially-Regularized Correlation Filters. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 4670–4679.
12. Danelljan, M.; Bhat, G.; Khan, F.S.; Felsberg, M. ECO: Efficient Convolution Operators for Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6931–6939.
13. 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](#)]
14. Tang, M.; Feng, J. Multi-kernel Correlation Filter for Visual Tracking. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 11–18 December 2015; pp. 3038–3046.
15. Dalal, N.; Triggs, B. Histograms of Oriented Gradients for Human Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Diego, CA, USA, 20–25 June 2005.
16. Danelljan, M.; Khan, F.S.; Felsberg, M.; Van De Weijer, J. Adaptive Color Attributes for Real-Time Visual Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 1090–1097.
17. Chatfield, K.; Simonyan, K.; Vedaldi, A.; Zisserman, A. Return of the Devil in the Details: Delving Deep into Convolutional Nets. In Proceedings of the British Machine Vision Conference (BMVC), Nottingham, UK, 1–5 September 2014.
18. Wu, Y.; Lim, J.; Yang, M.H. Object Tracking Benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1834–1848. [[CrossRef](#)] [[PubMed](#)]
19. Kristan, M.; Leonardis, A. *The Visual Object Tracking VOT2016 Challenge Results*; Springer: Amsterdam, The Netherlands, 2016; Volume 8926, pp. 191–217.
20. Kristan, M.; Leonardis, A.; Matas, J. The sixth visual object tracking vot2018 challenge results. In Proceedings of the European conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
21. Fan, H.; Ling, H.; Lin, L.; Yang, F.; Chu, P.; Deng, G.; Yu, S.; Bai, H.; Xu, Y.; Liao, C. LaSOT: A High-Quality Benchmark for Large-Scale Single Object Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 5374–5383.
22. Mei, X.; Ling, H. Robust Visual Tracking and Vehicle Classification via Sparse Representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 2259–2272. [[PubMed](#)]
23. Zhang, T.; Ghanem, B.; Liu, S.; Ahuja, N. Robust Visual Tracking via Multi-task Sparse Learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012; pp. 2042–2049.
24. Zhang, T.; Xu, C.; Yang, M.H. Multi-task Correlation Particle Filter for Robust Object Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 4819–4827.

25. Babenko, B.; Yang, M.H.; Belongie, S. Robust Object Tracking with Online Multiple Instance Learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 1619–1632. [[CrossRef](#)] [[PubMed](#)]
26. Hare, S.; Golodetz, S.; Saffari, A.; Vineet, V.; Cheng, M.M.; Hicks, S.L.; Torr, P.H. Struck: Structured Output Tracking with Kernels. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 2096–2109. [[CrossRef](#)] [[PubMed](#)]
27. Henriques, J.o.F.; Caseiro, R.; Martins, P.; Batista, J. Exploiting the Circulant Structure of Tracking-by-detection with Kernels. In Proceedings of the European Conference on Computer Vision (ECCV), Firenze, Italy, 7–13 October 2012; pp. 702–715.
28. Wen, L.; Ma, Q.; Yu, S.; Chui, K.T.; Xiong, N. Variational Regularized Tree-Structured Wavelet Sparsity for CS-SENSE Parallel Imaging. *IEEE Access* **2018**, *6*, 61050–61064.
29. Lukežič, A.; Vojř, T.; Zajc, L.Č.; Matas, J.; Kristan, M. Discriminative Correlation Filter with Channel and Spatial Reliability. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 4847–4856.
30. Ma, C.; Huang, J.; Yang, X.; Yang, M. Hierarchical Convolutional Features for Visual Tracking. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 11–18 December 2015; pp. 3074–3082.
31. Li, J.; Hong, Z.; Zhao, B. Robust Visual Tracking by Exploiting the Historical Tracker Snapshots. In Proceedings of the IEEE International Conference on Computer Vision Workshop (ICCVW), Santiago, Chile, 7–13 December 2015; pp. 604–612.
32. Li, J.; Deng, C.; Xu, R.Y.D.; Tao, D.; Zhao, B. Robust Object Tracking With Discrete Graph-Based Multiple Experts. *IEEE Trans. Image Process.* **2017**, *26*, 2736–2750. [[CrossRef](#)] [[PubMed](#)]
33. Wang, N.; Zhou, W.; Tian, Q.; Hong, R.; Wang, M.; Li, H. Multi-cue Correlation Filters for Robust Visual Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 4844–4853.
34. 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 (ECCV), Munich, Germany, 8–14 September 2018; pp. 483–498.
35. Wang, H.; Wang, L.; Zhou, Z.; Tao, X.; Pau, G.; Arena, F. Blockchain-Based Resource Allocation Model in Fog Computing. *Appl. Sci.* **2019**, *9*, 5538. [[CrossRef](#)]
36. Bertinetto, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H. Fully-convolutional Siamese Networks for Object Tracking. In Proceedings of the European conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 8–16 October 2016; pp. 850–865.
37. Valmadre, J.; Bertinetto, L.; Henriques, J.; Vedaldi, A.; Torr, P.H. End-to-end Representation Learning for Correlation Filter Based Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 5000–5008.
38. 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, 18–22 June 2018; pp. 8971–8980.
39. 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 (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 4660–4669.
40. Bhat, G.; Danelljan, M.; Van Gool, L.; Timofte, R. Learning Discriminative Model Prediction for Tracking. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Long Beach, CA, USA, 27 October–2 November 2019; pp. 6182–6191.
41. Zhang, Z.; Peng, H. Deeper and wider siamese networks for real-time visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 4591–4600.
42. Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; Eckstein, J. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Found. Trends Mach. Learn.* **2011**, *3*, 1–122. [[CrossRef](#)]
43. Sherman, J.; Morrison, W.J. Adjustment of an Inverse Matrix Corresponding to a Change in One Element of a Given Matrix. *Ann. Math. Stat.* **1950**, *21*, 124–127. [[CrossRef](#)]
44. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
45. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
46. 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 (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 4293–4302.
47. Choi, J.; Chang, H.J.; Yun, S.; Fischer, T.; Demiris, Y.; Choi, J.Y. Attentional Correlation Filter Network for Adaptive Visual Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 4828–4837.
48. Bertinetto, L.; Valmadre, J.; Golodetz, S.; Miksik, O.; Torr, P.H. Staple: Complementary Learners for Real-time Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 1401–1409.
49. Li, Y.; Zhu, J. A Scale Adaptive Kernel Correlation Filter Tracker with Teature Integration. In Proceedings of the European Conference on Computer Vision (ECCV), Zurich, Switzerland, 6–12 September 2014; pp. 254–265.
50. Zhang, J.; Ma, S.; Sclaroff, S. MEEM: Robust Tracking via Multiple Experts using Entropy Minimization. In Proceedings of the European Conference on Computer Vision (ECCV), Zurich, Switzerland, 6–12 September 2014; pp. 188–203.

-
51. Sun, C.; Wang, D.; Lu, H.; Yang, M.H. Learning Spatial-aware Regressions for Visual Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 8962–8970.
 52. He, A.; Luo, C.; Tian, X.; Zeng, W. A Twofold Siamese Network for Real-time Object Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 4834–4843.
 53. Gundogdu, E.; Alatan, A.A. Good Features to Correlate for Visual Tracking. *IEEE Trans. Image Process.* **2018**, *27*, 2526–2540. [[CrossRef](#)] [[PubMed](#)]
 54. Danelljan, M.; Hager, G.; Khan, F.S.; Felsberg, M. Discriminative Scale Space Tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1561–1575. [[CrossRef](#)] [[PubMed](#)]
 55. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 580–587.