applied sciences

MDPI

Article

Robust Long-Term Visual Object Tracking via Low-Rank Sparse Learning for Re-Detection

Shanshan Luo [1,2], Baoqing Li [1,*], Xiaobing Yuan [1] and Huawei Liu [1]

[1] Science and Technology on Micro-System Laboratory, Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai 201800, China; lss1110@mail.ustc.edu.cn (S.L.); sinowsn@mail.sim.ac.cn (X.Y.); liuhuawei@mail.sim.ac.cn (H.L.)
[2] University of Chinese Academy of Sciences, Beijing 100049, China
* Correspondence: sinoiot@mail.sim.ac.cn

Abstract: The Discriminative Correlation Filter (DCF) has been universally recognized in visual object tracking, thanks to its excellent accuracy and high speed. Nevertheless, these DCF-based trackers perform poorly in long-term tracking. The reasons include the following aspects—first, they have low adaptability to significant appearance changes in long-term tracking and are prone to tracking failure; second, these trackers lack a practical re-detection module to find the target again after tracking failure. In our work, we propose a new long-term tracking strategy to solve these issues. First, we make the best of the static and dynamic information of the target by introducing the motion features to our long-term tracker and obtain a more robust tracker. Second, we introduce a low-rank sparse dictionary learning method for re-detection. This re-detection module can exploit a correlation among these training samples and alleviate the impact of occlusion and noise. Third, we propose a new reliability evaluation method to model an adaptive update, which can switch expediently between the tracking module and the re-detection module. Massive experiments demonstrate that our proposed approach has an obvious improvement in precision and success rate over these state-of-the-art trackers.

Keywords: visual tracking; long-term; correlation filter; low-rank sparse; motion boundary; re-detection; reliability evaluation

Citation: Luo, S.; Li, B.; Yuan, X.; Liu, H. Robust Long-Term Visual Object Tracking via Low-Rank Sparse Learning for Re-Detection. *Appl. Sci.* **2021**, *11*, 1963. https://doi.org/10.3390/app11041963

Academic Editor: Arjan Kuijper

Received: 12 January 2021
Accepted: 18 February 2021
Published: 23 February 2021

1. Introduction

Visual object tracking is one of the main tasks in computer vision and has a broad spectrum of applications [1–6] in precision guidance, automatic driving, computer-aided medicine, and so on. In a nutshell, the task of visual object tracking is to continuously locate the target in a series of video frames, on the premise that we only know the target's bounding box in the initial frame. Actually, due to the limited training data and the inevitable interference of complex scenes, visual object tracking is a complex and challenging problem in reality.

In recent years, the object tracking algorithms aimed at short-term tracking have made significant progress in enhancing the tracking performance [7–10]. Therefore, the Discriminative Correlation Filter (DCF) has drawn significant attention for its computation efficiency [11–29], and has shown a state-of-the-art performance on multiple challenging benchmark tracking datasets [30–32]. The DCF-based trackers can be formulated as a ridged regression model, which learns from a set of object appearance and background training samples obtained by cyclic shifts. They achieve extremely high computing efficiency by utilizing diagonalizing expressions with circulant matrices in the frequency domain.

Actually, the video monitoring time is very long, so we expect to track the target correctly over a long time. The DCF-based trackers are widely recognized for their excellent overall performance in short-term tracking. However, they suffer from important issues like target re-detection where once the tracking target is lost, these short-term trackers

cannot relocate the target and fail consecutively in subsequent frames due to the no re-detection module. Despite the attempts by various works to solve this issue [33–37], it is still not completely solved. The reasons include the low adaptability to noticeable appearance changes and low ability to re-detect the target. In this work, we mainly solve the problems of the long-term tracker based DCF in several aspects. First, we try to make the most of the target's static and dynamic information by introducing the motion features to our long-term tracker. Second, we study a low-rank sparse dictionary learning method for re-detection to alleviate the impact of occlusion and noise. Third, we research a new reliability evaluation method to carry out adaptive model updating or decide whether to perform re-detection according to the evaluation results.

Massive experiments on OTB-2013 [30], OTB-2015 [30], and Temple Color-128 [15] datasets prove that our proposed method has an obvious improvement compared to the baseline tracker and outperforms several representative state-of-the-art methods in success rate and precision. Besides, our proposed tracking framework can apply to other DCF-based trackers for long-term tracking.

In conclusion, the effective contributions in this paper include the following aspects.

(1) We introduce motion features as a supplement to the appearance features. By making full use of static and dynamic information, we can improve the adaptability to significant changes in target appearance during long-term tracking and obtain a more robust tracker. The extensive experiments demonstrate that this method can improve the performance of our tracker.

(2) We introduce a low-rank sparse dictionary learning method for re-detection. This re-detection module can exploit a correlation among these training samples and alleviate the impact of occlusion and noise. The comparative experiments demonstrate that the tracker's success rate and precision have been dramatically improved by introducing a re-detection module.

(3) We propose a new reliability evaluation approach to update the model adaptively by considering both the PSR value and the trajectory smoothness degree. Using this method, our proposed tracker can switch timely between the tracking module and the re-detection module. The results indicate that our approach further improve the performance of our proposed tracker.

2. Related Work

In this section, we discuss the closely related to our algorithm—DCF-based trackers and long-term trackers.

2.1. DCF-Based Trackers

Starting with the MOSSE algorithm [11], which adopts a correlation filter to a visual tracking task, discriminative correlation filter (DCF) based tracking gains extensive attention for its effectiveness and efficiency. Then, several recent works have been carried out to address the inherent limitations of the standard DCF-based tracking algorithms.

For non-linear regression, Henriques et al. [12] learn the kernelized correlation filter (KCF) via a kernel trick. KCF achieves an extremely high tracking speed, because of the diagonalization of the cyclic matrix in the frequency domain. Besides, Tang Ming et al. [38] propose a multi-kernel correlation filter (MKCF), which makes use of the invariance-discriminative power spectrums of various features and improves the performance of the tracker further. For scale estimation, a scale pool has been introduced in DSST [13] and SAMF [14]. They can learn online using the target appearance samples at the scale pool and achieve accurate scale adaptive visual tracking. For boundary effects, Danelljan et al. [17] introduce a spatial regularization component in DCF formulation to penalize the correlation filer coefficients during tracking. CSR-DCF [39] makes use of the spatial reliability map to adjust the spatial constraint and then guarantees that filter values are zero outside of the object bounding box. BACF [23] extracts real negative training samples from negative background blocks to further suppress the boundary effect. Besides, other improvements in

DCF have also made much development in context-aware [40], temporal regularisation [21], deformable parts [41].

2.2. Long-Term Trackers

Different from short-term trackers, long-term trackers need to address the problems of the re-detection of the target after tracking failure and the low adaptability to significant appearance changes. Many algorithms are proposed to acquire impressive performance in long-term tracking.

Recently, most algorithms decompose the tracking task into multiple modules to handle long-term tracking. For example, in [42], Zdenek et al. design a framework (TLD) that breaks the tracking task down into multiple subtasks—tracking, learning and detection. The tracker and detector can promote each other. This framework performs favourably against these existing trackers in long-term tracking. Following the idea of TLD, Ma et al. [33] teach a DCF-based tracker to estimate the translation and scale variation, and train a complementary detector by utilizing online random ferns to relocate the target. In [34], LT-FLO proposes a two-module approach—the short-term module and the long-term module. The short-term module adopts edge-based features to enhance the ability of the tracker to light changes. The long-term module combines the online model of appearance and the learned pose manifold to re-detect the target. In [35], MUSTer propose a dual-component approach that combines the long-term and short-term memory. MUSTer employs an integrated correlation filter for short-term memory and keypoint matching based RANSAC for long-term memory. In [36], Wang et al. integrate a color-based model into the discriminative correlation filter and adopt a simple sparse coding based method to acquire reliable searching areas. Since most of the aforementioned re-detection modules adopt intensive sampling, the target template and background template may contain many similar components, which may affect the performance of the long-term tracker. In this work, we investigate the re-detection module to improve the ability to distinguish background and target.

In addition, in long-term tracking, the visual feature representation has a great influence on tracker results. A majority of the DCF-based trackers only use the appearance features extracted from a single frame, such as HOG [43], color histogram [15], CN [17], convolutional neural network (CNN) features [16,18,19]. For example, Huang Bo et al. [44] adopt the HOG features extracted from background-aware samples to train multi-channel correlation filters. NL Baisa et al. [45] use the CNN features to provide better feature representation for the discriminative correlation filter. However, because the appearance target may change significantly during tracking, the target appearance model is not always good at distinguishing the target from the background. In [46–48], some studies have shown that motion features reflect the information between frames and are complementary to the appearance features. Hence, we investigate the combination of appearance features and motion features during tracking.

Besides, it is essential to identify the state of the object in long-term tracking. In [11], Bolme et al. adopt the peak-versus-noise ratio (PSR) based on the correlation response map to update the tracking model adaptively. Moreover, LMCF [49] updates the tracker by the Average Peak to Correlation (APCE) criterion, reflecting the fluctuation of the response map and the confidence level of the tracking result. However, when the background is cluttered, the adoption of PSR or APCE for reliability estimation may lead to the tracker's wrong prediction. Therefore, in this work, we investigate a better estimate of the reliability of modeling an adaptive update and timely switching between the tracking module and the re-detection module.

3. Tracking Algorithm Design

To achieve long-term stable object tracking, we put forward a long-term visual tracking approach, consisting of a tracking module, re-detection module and reliability estimation. Figure 1 shows the overall architecture of our proposed tracker.

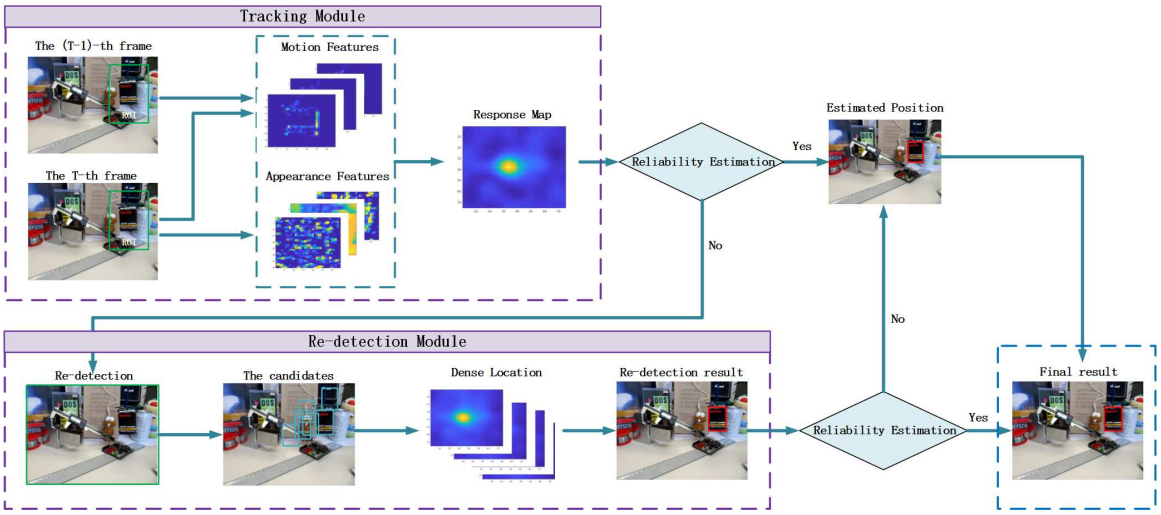


Figure 1. The framework of our proposed tracker. Our method mainly consists of tracking and re-detection modules. After the tracking module has processed the input frames, we perform a reliability estimation for the tracking result, update the tracking model adaptively, and decide whether re-detection is needed for the current result. If the re-detection process is adopted, we perform a reliability estimation for the re-detection result and decide whether the re-detected result can replace the originally detected result re-detection is needed for the current result.

3.1. Tracking Module

The DCF-based trackers have been extensively researched, thanks to their excellent performance. The goal of these trackers is to learn a DCF to infer the location of the target in the subsequent frames. Here, we introduce a multi-channel correlation filter h [40,50]. x denotes the training example, which is extracted from the video sequence images. y is a 2D Gaussian function label, representing the regression target. The filter h can be achieved by solving a ridge regression problem.

$$\varepsilon(h) = \frac{1}{2}\left\| y - \sum_{l=1}^{d} h_l \star x_l \right\|_2^2 + \frac{\lambda}{2}\sum_{l=1}^{d}\|h_l\|^2, \tag{1}$$

where $x_l \in \mathbb{R}^D$ expresses the lth channel of the vectorized image. $h_l \in \mathbb{R}^D$ indicates the lth channel of the filter. $y \in \mathbb{R}^D$ is the desired correlation response. $\sum_{l=1}^{d} h_l \star x_l$ denotes the actual correlation response. l represents the feature channel, $l \in \{1, 2, ..., d\}$. The $\star$ symbol refers to the circular correlation. λ is the weight of the regularization term.

The minimizer has a closed-form solution in the frequency domain by using Parseval's formula. It can be expressed as

$$\hat{h_l} = \frac{\hat{x_l} \odot \hat{y}}{\sum_{i=1}^{d} \hat{x_i} \odot \hat{x_i^*} + \lambda}. \tag{2}$$

Here, the hat denotes the discrete Fourier transform (DFT) of a function. $*$ symbol indicates the complex conjugation. $\odot$ symbol is element-wise multiplication.

As for scale estimation, we estimate the scale s by a single scale-space correlation filter from the discriminative scale space tracking (DSST) algorithm [13].

During tracking, the visual feature representation has a great influence on tracker results. Most trackers use only the appearance features extracted from a single frame, such as HOG [43], color histogram [15], CN [17] and CNN features [16,18,19]. However, the motion features integrate information between a pair of frames and are complementary to the appearance features. In this work, we introduce the motion features into correlation tracking.

Motion Descriptors: Optical flow extends static image features into the temporal domain, representing the motion information between two neighboring images. We can obtain optical flow (u, v) using a gradient-based optical flow algorithm [51], where u, v represent the horizontal and vertical components of optical flow respectively. Optical flow takes both target motion and camera motion into account, but taking camera motion into account may lead to errors in some cases. In most cases, the camera motion is locally translational and varies smoothly across the image plane, such as zooming, tilting. Dalal et al. [46] adopt the motion boundary histograms (MBH) descriptor to extract motion features, which can mitigate the effects of camera motion. The results of their experiment showed that MBH is more discriminative for target motion detection compared to optical flow. Therefore, to keep the information about motion boundaries and remove the constant motion information, we try to adopt the MBH descriptor to extract motion features.

Let u_x, u_y, v_x, v_y denote the corresponding horizontal and vertical gradients of optical flow (u, v), respectively. The gradient magnitude and orientation for each pixel are defined as follows:

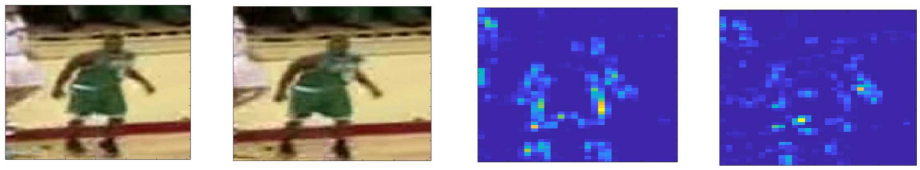
Horizontal motion boundaries:

$$mag^x = \sqrt{(u_x)^2 + (u_y)^2}, \qquad \theta^x = arctan(\frac{u_y}{u_x}). \tag{3}$$

Vertical motion boundaries:

$$mag^y = \sqrt{(v_x)^2 + (v_y)^2}, \qquad \theta^y = arctan(\frac{v_y}{v_x}). \tag{4}$$

Then, we obtain the motion boundary histograms f_{MBH_x}, f_{MBH_y} (see Figure 2), voting with mag and σ, similar to extracting the HOG features. Finally, we adopt a feature fusion scheme that concatenates the appearance features and motion features into an augmented feature vector $f = [f^T_{appearance}, f^T_{MBH_x}, f^T_{MBH_y}]^T$, which $f_{appearance}$ refers to the appearance features. In this paper, we extract the HOG [43] and color histogram [15] as the appearance features $f_{appearance}$.



(a) (b) (c) (d)

Figure 2. Illustration of the MBH. (**a**,**b**) Reference images at time t − 1 and t. (**c**) Horizontal motion boundary histograms. (**d**) Vertical motion boundary histograms.

3.2. Re-Detection Module

The DCF-based tracker may lose the target, due to the impact of occlusion, out of view, and so on. Therefore, in this section, we propose a re-detection module that can mine more candidates in the extensive background if the DCF-based tracker is unreliable. The reliability estimation method is described in Section 3.3.

In the initial frame, we densely sample a predefined number of target and background templates around the current target's location as the positive and negative training samples, respectively. Due to the dense sampling, the target and background templates may contain many similar ingredients. In addition, to alleviate the impact of occlusion and noise, we should exploit a correlation among these training samples. Here, our method utilizes sparse and low-rank representation [52–54] in a particle filter.

Suppose that the template set $A = [A_1, A_2, ..., A_N] \in R^{D \times N}$, where D is the feature dimension of training sample, N refers to the total number of training samples. Let $X = [X_1, X_2, ..., X_M] \in R^{D \times M}$ be the data matrix of all test samples. Thus, $X = AZ + E$,

where $Z = [Z_1, Z_2, ..., Z_M] \in R^{N \times M}$ is the representation matrix of X. Then, we can seek a representation of Z among all the candidates by solving the following optimization problem:

$$\begin{aligned} &min_{Z,E} \quad rank(Z) + \lambda_1 \|Z\|_0 + \lambda_2 l(E), \\ &s.t. \quad X = AZ + E, Z \geq 0, \end{aligned} \tag{5}$$

where $E = [E_1, E_2, ..., E_M] \in R^{D \times M}$ is a noise term, and $l(E) = \|E\|_{2,1} = \sum_{j=1}^{M} \sqrt{\sum_{i=1}^{D} (|E|_{ij})^2}$ [55], so minimizing $\|E\|_{2,1}$ encourages the columns of E to be zero. Since solving problem Equation (5) is NP-hard, we can relax rank function to nuclear norm as follows, and relax the l_0-norm to the l_1-norm. Therefore, Equation (5) can be rewritten as follows:

$$\begin{aligned} &min_{Z,E} \quad \|Z\|_* + \lambda_1 \|Z\|_1 + \lambda_2 \|E\|_{2,1}, \\ &s.t. \quad X = AZ + E, Z \geq 0. \end{aligned} \tag{6}$$

Here, $\|\cdot\|_*$ indicates the nuclear norm, that is, the sum of the singular values of the matrix. The parameter $\lambda_1 > 0$ and $\lambda_2 > 0$ are scalar parameters. We introduce an auxiliary variable W and convert the Equation (6) into an equivalent optimization problem:

$$\begin{aligned} &min_{Z,E} \quad \|Z\|_* + \lambda_1 \|W\|_1 + \lambda_2 \|E\|_{2,1}, \\ &s.t. \quad X = AZ + E, Z = W, Z \geq 0. \end{aligned} \tag{7}$$

We can solve the optimization problem and obtain the low-rank sparse dictionary $\hat{Z}$ and the noise matrix $\hat{E}$ by the linearized alternating direction method with the adaptive penalty (LADMAP) [52].

The augmented Lagrangian function of Equation (7) is

$$\begin{aligned} L(Z, E, W, Y_1, Y_2, \mu) =& \|Z\|_* + \lambda_1 \|W\|_1 + \lambda_2 \|E\|_{2,1} \\ &+ \langle Y_1, X - AZ - E \rangle + \langle Y_2, Z - W \rangle \\ &+ \frac{\mu}{2} (\|X - AZ - E\|_F^2 + \|Z - W\|_F^2), \end{aligned} \tag{8}$$

where μ is the corresponding penalty factor. Y_1 and Y_2 denote the Lagrangian multiplier. $\langle ., . \rangle$ is the inner product. Adopting LADMAP [52], we minimize the function L with other two variables fixed to update the variables Z, E, W alternately.

When the target needs to be re-detected, we should first obtain a certain amount of the region of interest (ROI) candidates. Unlike random sampling or exhaustive sliding window search to obtain the candidates, we use EdgeBox algorithm [56] to produce a group of candidate bounding boxes and their proposal scores, and select the top M proposals as our candidate sample set $x = \{x_1, x_2, ..., x_M\}$.

Further, we need to evaluate the reliability of these M candidate samples. For the candidate sample set x of the current frame, its coding coefficients $\langle \hat{Z}, \hat{E} \rangle$ can be obtained by Equation (7), where $\hat{Z} = [\hat{Z}_+, \hat{Z}_-]$ and $\hat{E} = [\hat{E}_+, \hat{E}_-]$. The template set A consists of the target templates A_+ and background templates A_-, that is, $A = [A_+, A_-]$. So, for the i-th candidate sample x_i, the reconstruction error between it and target templates can be expressed as $\varepsilon_p = \left\| x_i - A_+ \hat{Z}_+^i - \hat{E}_+^i \right\|_2^2$, and the reconstruction error between it and the background template is expressed as $\varepsilon_n = \left\| x_i - A_- \hat{Z}_-^i - \hat{E}_-^i \right\|_2^2$.

Then, the likelihood score of the candidate sample x_i can be measured as follows:

$$score = exp\left\{ -\frac{\varepsilon_p}{\sigma(\varepsilon_n + \gamma)} \right\}. \tag{9}$$

Here, σ and γ are set to 0.2 and 10^{-32} respectively. Obviously, the higher the score, the more reliable the candidate. Then, we obtain the position of the candidate which has

the highest likelihood score and adopt the correlation filter obtained in Section 3.1 to get a new response map. After that, we re-estimate the reliability of the response map by adopting the reliability estimation method described in Section 3.3. If the result is reliable, we replace it with the original tracking result. If not, we take the original tracking result.

3.3. Reliability Estimation

In the overall tracking framework, reliability estimation of tracking results is crucial for adaptive model updating and target re-detection. Here, we have two schemes to estimate the tracking reliability. The first is the peak sidelobe ratio (PSR) defined by the response map, and the second is the trajectory smoothness degree of the tracker.

First, the PSR of the correlation filter response map is defined as

$$PSR_t = \frac{(R^t_{max}) - \mu_i}{\sigma_t}, \tag{10}$$

where R^t_{max} indicates the maximum value of the response map in the t-th frame, μ_i and σ_i denote the mean and standard deviation of the response map, respectively. The higher the PSR value is, the more reliable the response result will be. Figure 3 reflects the relationship between the PSR value and reliability.

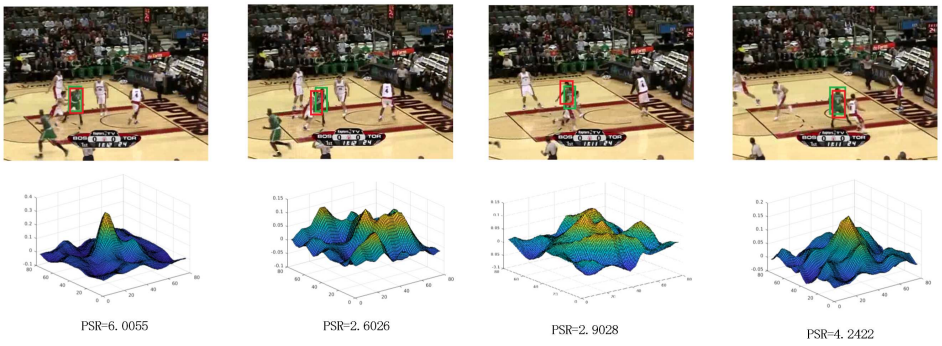


Figure 3. Peak sidelobe ratio (PSR) values significantly decrease when tracking results become less reliable (the green box represents the ground-truth target location, and the red color denotes the tracker outputs).

Second, the trajectory smoothness degree of the tracker can indicate the reliability of our tracking results as well. It can be defined as follow:

$$Smooth_t = exp(-\frac{1}{2\sigma^2} D_t^2), \tag{11}$$

where $D_t = \left\| c(B^t) - c(B^{t-1}) \right\|$, $c(B^t)$ and $c(B^{t-1})$ denote the center of the bounding box in the current frame and the center of the bounding box in the previous frame, respectively. $\sigma = \frac{1}{2}[w(B^t) + h(B^t)]$ is the average length of the width $w(B^t)$ and the height $h(B^t)$ of the bounding box. The closer the value of $Smooth_t$ is to 1, the better reliability of the tracking result. Figure 4 reflects the relationship between the trajectory smoothness degree and reliability.

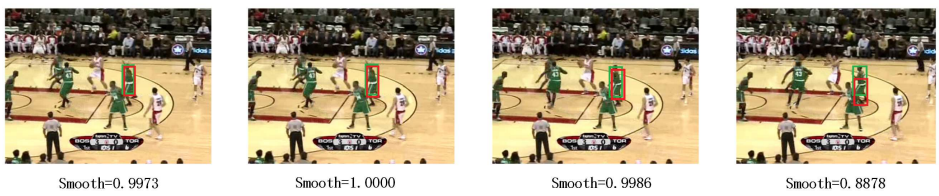


Figure 4. The values of $Smooth_t$ significantly decrease when tracking results become less reliable (the green box represents the ground-truth target location, and the red color denotes the tracker outputs).

Thereinto, derived from the response of the current frame, the PSR value reflects the reliability of the result spatially. Unlike the PSR value, the trajectory smoothness degree of the tracker considers the reliability temporally. In this paper, we consider both the PSR value and the trajectory smoothness degree.

For the switch strategy between the tracking and re-detection module, when the PSR value is lower than the threshold, or the trajectory smoothness degree is lower than the threshold, re-detection is performed. This strategy can increase the robustness of the algorithm effectively in long-term tracking. As for template updates, we update the tracking model adaptively using the reliability estimation criterion. In the current frame t, if the PSR value and the trajectory smoothness degree are higher than the thresholds, we update the template of the correlation filter $\hat{h}^t_{template} = \hat{h}^{t-1}_{template} + \hat{h}^t$, in which η denotes the learning rate. Otherwise, we do not update the model to avoid learning unreliable information. For the re-detection module, when the PSR value and the trajectory smoothness degree are higher than the thresholds, the positive and negative templates are updated.

Our proposed tracking approach is summarized in Algorithm 1.

Algorithm 1: Our proposed tracking approach

Input: Image frame I_0, initial target bounding box x_0.
Output: Position p_t, scale s_t.

for $t = 2$ *to* n **do**
  Extract the appearance and the motion features in frame t; Compute the correlation response map with Equation (2) and estimate new target position p_t and scale s_t; Compute the PSR value and the trajectory smoothness degree with Equation (10) and Equation (11);
  if $PSR_t < \tau_1 \cdot M_{PSR}$ *or* $Smooth_t < \sigma_1$ **then**
    .
  end
  Generate re-detection candidates using the EdgeBox algorithm;
  Compute candidates' confidence using low-rank sparse dictionary with Equation (9) and obtain new target position $p_t^{'}$;
  Compute the correlation response map with Equation (2) at the position $p_t^{'}$ and compute $PSR_t^{'}$ and $Smooth_t^{'}$ with Equation (10) and Equation (11);
  if $PSR_t^{'} > \tau_2 \cdot M_{PSR}$ *and* $Smooth_t^{'} > \sigma_2$ **then**
    $p_t = p_t^{'}$; $s_t = s_t^{'}$;
  end
  if $PSR_t > \tau_3 \cdot M_{PSR}$ *and* $Smooth_t > \sigma_3$ **then**
    .
  end
  Update the correlation tracker and the model;
  Update positive and negative templates;
end

4. Experiment

4.1. Experimental Setup

In this section, our proposed approach is compared to 11 state-of-the-art trackers on three challenging datasets: OTB-2013 [30], OTB-2015 [30] and Temple Color-128 [31]. These state-of-the-art trackers include the DCF-based trackers (i.e., SRDCF [17], ECOhc [18], Staple [15], DSST [13], KCF [12], CSR-DCF [39], SAMF [14], SAMF_CA [40]) and the trackers for long-term tracking (i.e., TLD [42], LCT [33]).

The comparison experiments are performed in MATLAB 2018a on a PC with an Intel i5-8400 CPU at 2.80GHz×6. In addition, to compare the performance of these trackers, we employ precision and success rate as evaluation metrics.

Parameters: In our experiments, the regularization parameter *lambda* is set to 10^{-3} in Equation (2), the regularization parameters $lambda_1$ and $lambda_2$ are set to 0.1 and 4 in Equation (7). In Equation (9), σ and γ are set to 0.2 and 10^{-32} respectively. For the reliability estimation, in Algorithm 1, $\{\tau_1, \sigma_1\}$ is set to $\{0.6, 0.8\}$, $\{\tau_2, \sigma_2\}$ is set to $\{0.7, 0.7\}$, $\{\tau_3, \sigma_3\}$ is set to $\{0.7, 0.9\}$.

4.2. Results and Analysis

4.2.1. OTB-2013 Benchmark

(1) Comparison with the baseline

In this work, we estimate the OTB-2013 dataset, which includes 51 challenging video sequences. Here, we choose Staple [15] as the baseline, which extracts HOG and color histogram as the appearance feature model and enables greater accuracy. To address the issues of the target detection again after tracking failure and the interference from complex scenes, we improve this algorithm and obtain good performance for long-term tracking. Our primary contribution is introducing the motion features into correlation tracking as a complement to appearance features. The secondary contribution is introducing a re-detection module based on sparse and low-rank representation for long-term tracking. The third contribution is proposing a reliability estimation method, which takes into account the reliability of the result spatially and temporally.

In Figure 5, we analyze these contributions in detail.

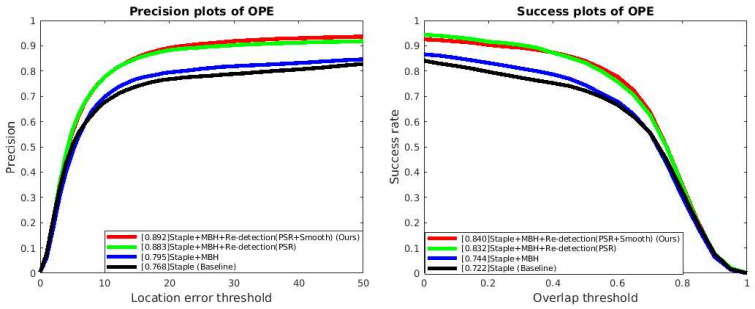


Figure 5. Precision plots (**left**) and Success plots (**right**) of OPE on OTB-2013 dataset. The values in the legend indicate the quantitative comparisons of distance precision at the threshold of 20 pixels and overlap success rate at the conventional thresholds of 0.5 ($IOU > 0.5$).

In Figure 5, on the whole, our proposed method provides an obvious performance improvement over the baseline tracker (Staple) by a success rate gain of 11.8% and a precision gain of 12.4%. More specifically, after introducing the motion features, our tracker provides a performance improvement by a success rate gain of 2.2% and a precision gain of 2.7%. On this basis, it can boost the success rate of 8.8% and the precision of 8.8% in by introducing a re-detection module based on sparse and low-rank representation for long-term tracking. Besides, our tracker can further improve the performance by a success rate of 0.8% and a precision of 0.9%. In conclusion, all of our contributions can effectively improve the tracker's performance, but the second one has the most significant improvement.

(2) Integration into Different DCF Trackers

Here, we combine multiple classical correlation filter based tracking algorithms with our proposed improvement method. In Figure 6, we compare these different DCF trackers with their improved algorithms.

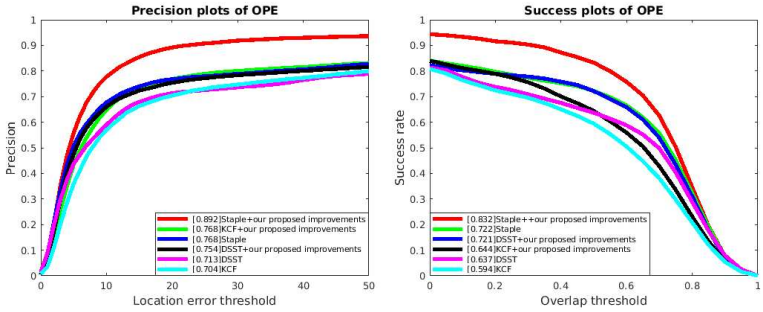


Figure 6. Precision plots (**left**) and Success plots (**right**) of OPE on OTB-2013 dataset. The values in the legend indicate the quantitative comparisons of distance precision at the threshold of 20 pixels and overlap success rate at the conventional thresholds of 0.5 ($IOU > 0.5$).

After introducing our proposed improvement method, the basic DCF based trackers (KCF [12], DSST [13], Staple [15]) all gain significant improvements. For example, the improved KCF algorithm has a success rate gain of 5.0% and a precision gain of 6.4%. The improved DSST algorithm boosts the success rate by 8.4% and the precision by 4.1%. Besides, the Staple tracker still obtains an obvious improvement by a success rate gain of 11.8% and a precision gain of 12.4%. The results show that it is feasible to improve the performance of a DCF-based tracker by introducing our proposed improvement methods.

(3) Comparison with the state-of-the-art trackers

We have done plenty of comparative experiments with these representative state-of-the-art trackers. These trackers include the DCF-based trackers (i.e., SRDCF [17], ECOhc [18], Staple [15], DSST [13], KCF [12], CSR-DCF [39], SAMF [14], SAMF_CA [40]) and the trackers for long-term tracking (i.e., TLD [42], LCT [33]).

In Figure 7, our method outperforms these state-of-the-art methods in the success rate of 83.2% and the precision of 89.2%.

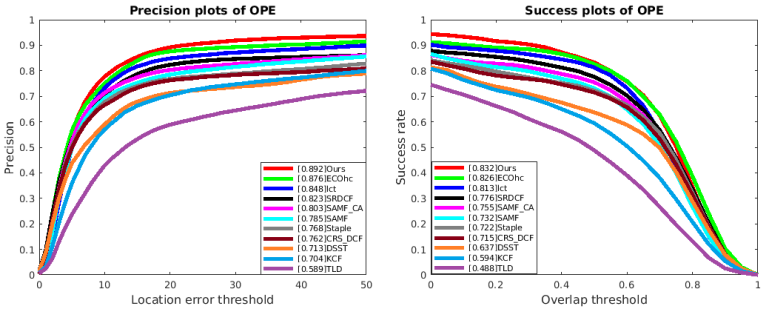


Figure 7. Precision plots (**left**) and Success plots (**right**) of OPE on OTB-2013 dataset. The values in the legend indicate the quantitative comparisons of distance precision at the threshold of 20 pixels and overlap success rate at the conventional thresholds of 0.5 ($IOU > 0.5$).

(4) Qualitative Analysis

For a further detailed analysis of these trackers, we provide the tracking results parameterized by different attributes in Figure 8, including background clutter (BC), deformation (DEF), fast motion (FM), in-plane rotation (IPR), illumination variation (IV), low resolution (LR), motion blur (MB), deformation (DEF), occlusion (OCC), out of view (OV), out-of-plane rotation (OPR) and scale variation (SV). Table 1 shows that our proposed tracker is superior to the baseline tracker in different attributes on the success rate and precision intuitively.

Table 1. The success rate (at the conventional thresholds of 0.5) of 11 attributes occurred on the OTB-2013 dataset.

	BC	MB	DEF	IV	IPR	LR	OCC	OPR	OV	SV	FM
Ours	0.796	0.732	0.871	0.766	0.761	0.628	0.848	0.815	0.865	0.775	0.723
Baseline (Staple)	0.739	0.512	0.693	0.643	0.667	0.508	0.701	0.685	0.659	0.658	0.576

Besides, our proposed approach outperforms other trackers in five attributes in Figure 9. To show the superiority of our tracker more intuitively, we conduct a qualitative analysis of these five challenging situations.

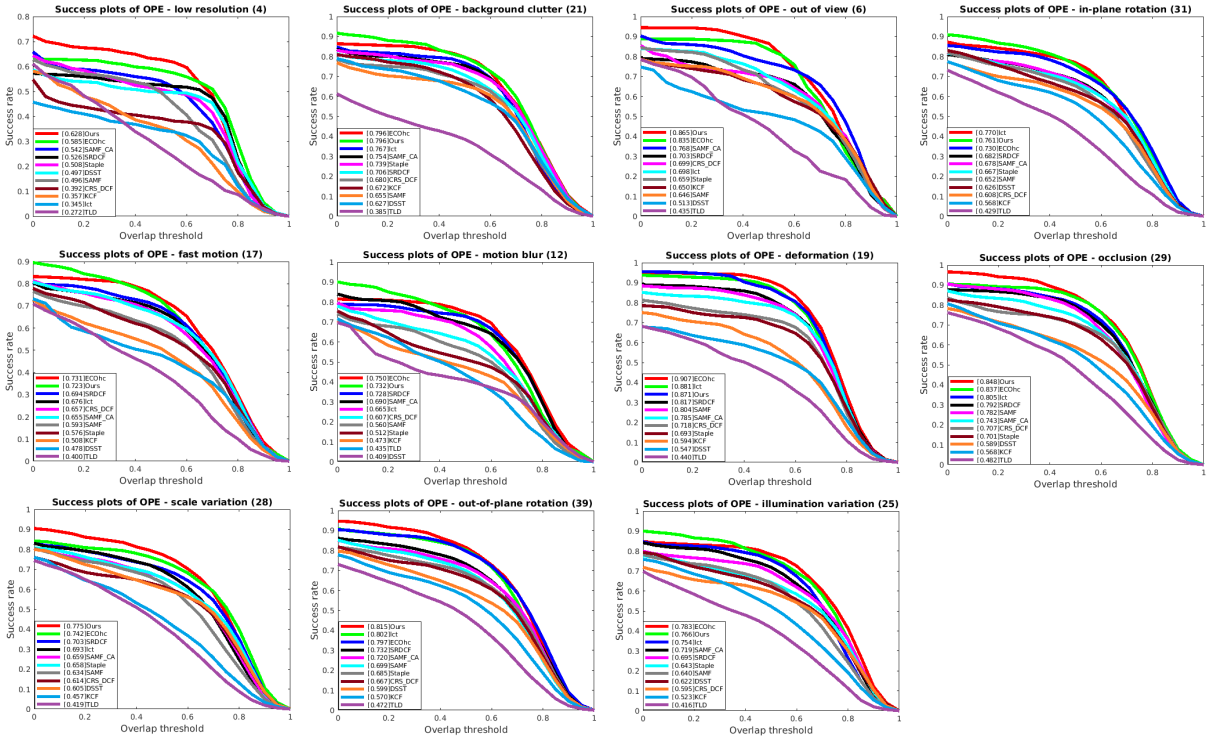


Figure 8. Success plots over 11 challenging attributes on OTB-2013 dataset. The values in the legend indicate the overlap success rate at the conventional thresholds of 0.5 ($IOU > 0.5$).

(a) Out of view

Figure 9a visualizes the tracking results on the box video sequence. We can observe that most trackers fail after the box in the video sequence moves out of view. Nevertheless, in Frame #495, our proposed tracker can re-detect the target accurately.

(b) Occlusion

Figure 9b shows that the freeman video sequence encounters occlusion over Frames #40 to #270. It is evident that occlusion may cause drift of the appearance model and lead to failure eventually. Among these trackers, our proposed tracker can achieve accurate tracking and outperforms the baseline (Staple) and most trackers.

(c) Scale variation

In practical applications, the relative movement between target and camera causes the change of the target scale. Figure 9c shows that KCF, SAMF, SAMF_CA, LCT and TLD cannot adapt to the change of the target scale well. However, our proposed approach can track the singer precisely.

(d) Out-of-plane rotation

The out-of-plane rotation may cause the appearance of the target to change greatly. In Figure 9d, only the tracker proposed by us can successfully track the ironman.

(e) Low resolution

Low resolution may make it impossible to obtain detailed information about the target, which is a challenge for accurate tracking. In Figure 9e, our tracker runs in low resolution. Our proposed tracker, TLD and ECOhc can track the panda precisely.

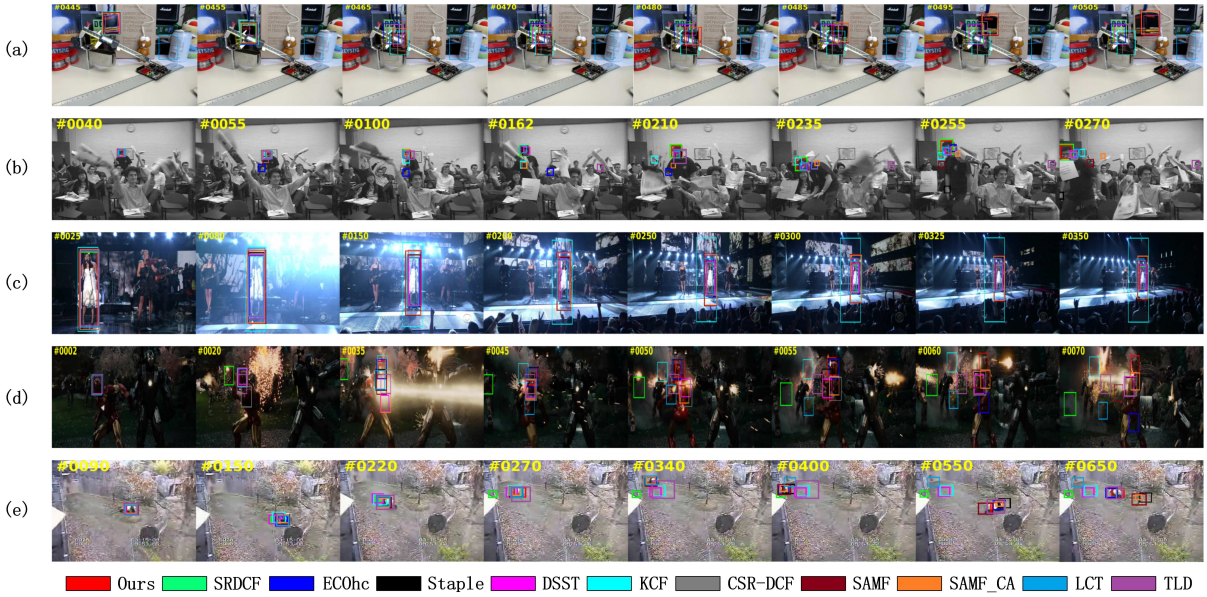


Figure 9. Qualitative comparison of our proposed tracker with state-of-the-art trackers on the box, freeman, singer, ironman and panda videos, under out of view, occlusion, scale variation, out-of-plane rotation and low resolution, respectively.

4.2.2. OTB-2015 Benchmark

In Figure 10, we present results on the OTB-2015 dataset. It is observed that our tracker achieves a prominent improvement over the baseline (Staple) with a success rate gain of 12.9% and a precision gain of 12.4%. Our proposed approach provides the best performance with a success rate of 80.3% and a precision of 87.1%.

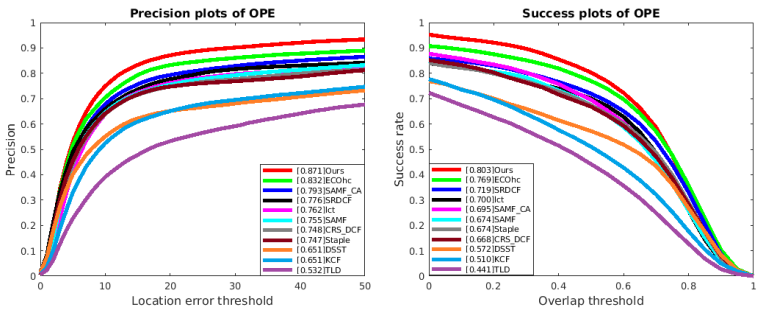


Figure 10. Precision plots (**left**) and Success plots (**right**) of OPE on OTB-2015 dataset. The values in the legend indicate the quantitative comparisons of distance precision at the threshold of 20 pixels and overlap success rate at the conventional thresholds of 0.5 ($IOU > 0.5$).

4.2.3. Temple Color-128 Benchmark

The Temple Color-128 dataset contains 128 color video sequences. In Figure 11, our proposed method obtains a substantial improvement over the baseline (Staple), with a success rate gain of 7.1% and a precision gain of 9.1%. Besides, our tracker outperforms all of the others in success rate and precision.

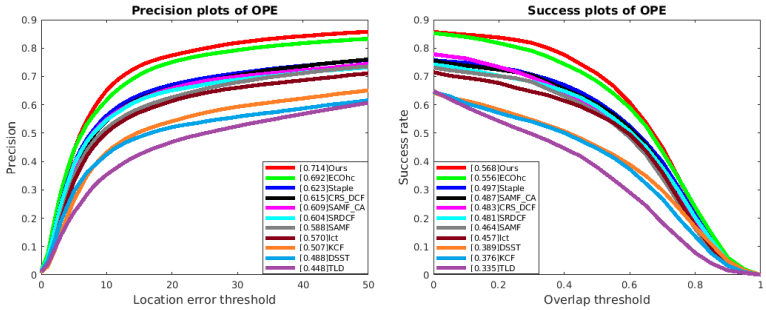


Figure 11. Precision plots (**left**) and success plots (**right**) of OPE on Temple color-128 dataset. The values in the legend indicate the quantitative comparisons of distance precision at the threshold of 20 pixels and overlap success rate at the conventional thresholds of 0.5 ($IOU > 0.5$).

4.2.4. Speed Analysis

For a full comparison, Table 2 lists the average tracking speed of these trackers on the OTB-2015 dataset. Our proposed tracker has an average speed of 19.1 FPS, which maintains a sufficient balance between speed and accuracy.

Table 2. The speeds of different trackers on OTB-2015 dataset.

	Ours	**ECOhc**	**STAPLE**	**SRDCF**	**BACF**	**SAMF _CA**	**CRS _DCF**	**KCF**	**SAMF**	**DSST**	**TLD**	**LCT**
Avg.FPS (CPU)	19.1	37.6	67.7	12.8	41.6	27.7	17.8	315.1	29.4	63.9	60.6	33.5

5. Conclusions

A novel long-term tracking method is proposed to achieve more robust long-term tracking in this paper. By introducing motion features as a supplement to the appearance features, our tracker can effectively improve the discrimination ability of the tracker. In our re-detection module, we introduce a low-rank sparse dictionary learning to re-detect targets. For reliability evaluation, our tracker can update the model adaptively, considering both the PSR value and the trajectory smoothness degree. Numerous experiments demonstrate that our proposed tracker overmatches other state-of-the-art trackers, and our proposed framework can apply to other DCF-based trackers in long-term tracking.

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

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: We use the public datasets (OTB-2013, OTB-2015, Temple Color-128) to verify my algorithm, and their links include http://cvlab.hanyang.ac.kr/tracker_benchmark (accessed on 12 January 2021) and http://www.dabi.temple.edu/hbling/data/TColor-128.html (accessed on 12 January 2021).

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

References

1. Smeulders, A.W.; Chu, D.M.; Cucchiara, R.; Calderara, S.; Dehghan, A.; Shah, M. Visual tracking: An experimental survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *36*, 1442–1468.
2. Lu, H.; Li, P.; Wang, D. Visual object tracking: A survey. *Pattern Recognit. Artif. Intell.* **2018**, *31*, 61–76.
3. Kim, I.S.; Choi, H.S.; Yi, K.M.; Choi, J.Y.; Kong, S.G. Intelligent Visual Surveillance—A Survey. *Int. J. Control. Autom. Syst.* **2010**, *8*, 926–939. [CrossRef]
4. Reddy, K.R.; Priya, K.H.; Neelima, N. Object Detection and Tracking—A Survey. In Proceedings of the 2015 International Conference on Computational Intelligence and Communication Networks (CICN), Jabalpur, India, 12–14 December 2015; pp. 418–421.
5. Deori, B.; Thounaojam, D.M. A Survey On Moving Object Tracking in Video. *Int. J. Inf. Theory* **2014**, *3*, 31–46. [CrossRef]
6. Pan, Z.; Liu, S.; Fu, W. A Review of Visual Moving Target Tracking. *Multimed. Tools Appl.* **2017**, *76*, 16989–17018. [CrossRef]
7. Baker, S.; Matthews, I. Lucas-Kanade 20 Years on: A Unifying Framework. *Int. J. Comput. Vis.* **2004**, *56*, 221–255. [CrossRef]
8. Oron, S.; Bar-Hillel, A.; Levi, D.; Avidan, S. Locally Orderless Tracking. *Int. J. Comput. Vis.* **2015**, *111*, 213–228. [CrossRef]
9. Ross, D.A.; Lim, J.; Lin, R.S.; Yang, M.H. Incremental Learning for Robust Visual Tracking. *Int. J. Comput. Vis.* **2008**, *77*, 125–141. [CrossRef]
10. Danelljan, M.; Shahbaz Khan, F.; 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, Columbus, OH, USA, 24–27 June 2014; pp. 1090–1097.
11. 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, San Francisco, CA, USA, 13–18 June 2010; pp. 2544–2550.
12. Henriques, J.F.; Caseiro, R.; Martins, P.; Batista, J. High-speed tracking with kernelized correlation filters. *IIEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *37*, 583–596. [CrossRef] [PubMed]
13. Danelljan, M.; Häger, G.; Khan, F.S.; Felsberg, M. Discriminative Scale Space Tracking. *IEEE Trans. Ppattern Anal. Mach. Intell.* **2016**, *39*, 1561–1575. [CrossRef]
14. Li, Y.; Zhu, J. A Scale Adaptive Kernel Correlation Filter Tracker with Feature Integration. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 5–12 September 2014; pp. 254–265.
15. 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, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 1401–1409.
16. Danelljan, M.; Hager, G.; Shahbaz, K.F.; Felsberg, M. Convolutional Features for Correlation Filter based Visual Tracking. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015; pp. 58–66.
17. Danelljan, M.; Hager, G.; Shahbaz, K.F.; Felsberg, M. Learning Spatially Regularized Correlation Filters for Visual Tracking. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015; pp. 4310–4318.
18. Danelljan, M.; Bhat, G.; Shahbaz Khan, F.; Felsberg, M. Eco: Efficient Convolution Operators for Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 22–25 July 2017; pp. 6638–6646.
19. 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, 8–16 October 2016; pp. 472–488.
20. 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, Beach, CA, USA, 16–20 June 2019; pp. 4670–4679.
21. 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, Salt Lake City, UT, USA, 19–21 June 2018; pp. 4904–4913.
22. Xu, T.; Feng, Z.H.; Wu, X.J.; Kittler, J. Learning Adaptive Discriminative Correlation Filters via Temporal Consistency Preserving Spatial Feature Selection for Robust Visual Object Tracking. *IEEE Trans. Image Process.* **2019**, *28*, 5596–5609. [CrossRef] [PubMed]
23. Kiani Galoogahi, H.; Fagg, A.; Lucey, S. Learning Background-aware Correlation Filters for Visual Tracking. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 19–22 October 2017; pp. 1135–1143.
24. Luo, S.; Li, B.; Yuan, X. An Anti-Drift Background-aware Correlation Filter for Visual Tracking in Complex Scenes. *IEEE Access* **2019**, *7*, 185857–185867. [CrossRef]
25. Xue, X.; Li, Y.; Shen, Q. Unmanned Aerial Vehicle Object Tracking by Correlation Filter with Adaptive Appearance Model. *Sensors* **2018**, *18*, 2751. [CrossRef] [PubMed]
26. Yang, Y.; Zhang, Y.; Li, D.; Wang, Z. Parallel Correlation Filters for Real-time Visual Tracking. *Sensors* **2019**, *19*, 2362. [CrossRef] [PubMed]
27. Shin, J.; Kim, H.; Kim, D.; Paik, J. Fast and Robust Object Tracking Using Tracking Failure Detection in Kernelized Correlation Filter. *Appl. Sci.* **2020**, *10*, 713. [CrossRef]
28. He, W.; Li, H.; Liu, W.; Li, C.; Guo, B. rStaple: A Robust Complementary Learning Method for Real-Time Object Tracking. *Appl. Sci.* **2020**, *10*, 3021. [CrossRef]
29. Wang, W.; Liu, C.; Xu, B.; Li, L.; Chen, W.; Tian, Y. Robust Visual Tracking Based on Fusional Multi-Correlation-Filters with a High-Confidence Judgement Mechanism. *Appl. Sci.* **2020**, *10*, 2151. [CrossRef]

30. Wu, Y.; Lim, J.; Yang, M.H. Online Object Tracking: A benchmark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 25–27 June 2013; pp. 2411–2418.
31. Liang, P.; Blasch, E.; Ling, H. Encoding Color Information for Visual Tracking: Algorithms and Benchmark. *IEEE Trans. Image Process.* **2015**, *24*, 5630–5644. [CrossRef] [PubMed]
32. Kristan, M.; Leonardis, A.; Matas, J.; Felsberg, M.; Pflugfelder, R.; Čehovin 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 (ECCV), Munich, Germany, 8–14 September 2018. [CrossRef]
33. Ma, C.; Yang, X.; Zhang, C.; Yang, M.H. Long-term Correlation Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 8–10 June 2015; pp. 5388–5396.
34. Lebeda, K.; Hadfield, S.; Matas, J.; Bowden, R. Long-term Tracking through Failure Cases. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Sydney, Australia, 1–8 December 2013; pp. 153–160.
35. Hong, Z.; Chen, Z.; Wang, C.; Mei, X.; Prokhorov, D.; Tao, D. Multi-store Tracker (muster): A Cognitive Psychology Inspired Approach to Object Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 8–10 June 2015; pp. 749–758.
36. Wang, N.; Zhou, W.; Li, H. Reliable Re-detection for Long-term Tracking. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *29*, 730–743. [CrossRef]
37. Chen, Z.; Liu, P.; Du, Y.; Luo, Y.; Guo, J. Long-term Correlation Tracking via Spatial-temporal Context. *Vis. Comput.* **2020**, *36*, 425–442. [CrossRef]
38. Tang, M.; Feng, J. Multi-kernel Correlation Filter for Visual Tracking. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015; pp. 3038–3046.
39. Lukezic, A.; Vojir, T.; Čehovin Zajc, L.; Matas, J.; Kristan, M. Discriminative Correlation Filter with Channel and Spatial Reliability. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6309–6318.
40. 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.
41. Lukežič, A.; Zajc, L.Č.; Kristan, M. Deformable Parts Correlation Filters for Robust Visual Tracking. *IEEE Trans. Cybern.* **2017**, *48*, 1849–1861.
42. Kalal, Z.; Mikolajczyk, K.; Matas, J. Tracking-learning-detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *34*, 1409–1422. [CrossRef] [PubMed]
43. Felzenszwalb, P.F.; Girshick, R.B.; McAllester, D.; Ramanan, D. Object Detection with Discriminatively Trained Part-based Models. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *32*, 1627–1645. [CrossRef] [PubMed]
44. Huang, B.; Xu, T.; Liu, B.; Yuan, B. Context Constraint and Pattern Memory for Long-term Correlation Tracking. *Neurocomputing* **2020**, *377*, 1–15. [CrossRef]
45. Baisa, N.L.; Bhowmik, D.; Wallace, A. Long-term Correlation Tracking using Multi-layer Hybrid Features in Sparse and Dense Environments. *J. Vis. Commun. Image Represent.* **2018**, *55*, 464–476. [CrossRef]
46. Dalal, N.; Triggs, B.; Schmid, C. Human Detection Using Oriented Histograms of Flow and Appearance. In Proceedings of the European Conference on Computer Vision, Graz, Austria, 7–13 May 2006; pp. 428–441.
47. Chunyu, Y.; Jun, F.; Jinjun, W.; Yongming, Z. Video Fire Smoke Detection Using Motion and Color Features. *Fire Technol.* **2010**, *46*, 651–663. [CrossRef]
48. Tu, F.; Ge, S.S.; Tang, Y.; Hang, C.C. Robust Visual Tracking via Collaborative Motion and Appearance Model. *IEEE Trans. Ind. Inform.* **2017**, *13*, 2251–2259. [CrossRef]
49. Wang, M.; Liu, Y.; Huang, Z. Large Margin Object Tracking with Circulant Feature Maps. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4021–4029.
50. Zeng, H.; Peng, N.; Yu, Z.; Gu, Z.; Liu, H.; Zhang, K. Visual Tracking using Multi-channel Correlation Filters. In Proceedings of the 2015 IEEE International Conference on Digital Signal Processing (DSP), Singapore, 21–24 July 2015; pp. 211–214.
51. Horn, B.K.; Schunck, B.G. Determining Optical Flow. *Tech. Appl. Image Underst.* **1981**, *281*, 319–331. [CrossRef]
52. Lin, Z.; Liu, R.; Su, Z. Linearized Alternating Direction Method with Adaptive Penalty for Low-rank Representation. *arXiv* **2011**, preprint arXiv:1109.0367.
53. Du, H.; Hu, Q.; Qiao, D.; Pitas, I. Robust Face Recognition via Low-rank Sparse Representation-based Classification. *Int. J. Autom. Comput.* **2015**, *12*, 579–587. [CrossRef]
54. Ding, Y.; Chong, Y.; Pan, S. Sparse and Low-rank Representation with Key Connectivity for Hyperspectral Image Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 5609–5622. [CrossRef]
55. Liu, J.; Ji, S.; Ye, J. Multi-task Feature Learning via Efficient L2, 1-norm Minimization. *arXiv* **2012**, preprint arXiv:1205.2631.
56. Zitnick, C.L.; Dollár, P. Edge boxes: Locating Object Proposals from Edges. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 391–405.