Improved Correlation Filter Tracking with Enhanced Features and Adaptive Kalman Filter

Hao Yang *, Yingqing Huang and Zhihong Xie

Department of Arms and Control Engineering, Army Academy of Armored Forces, Beijing 100072, China; huangyinqing1105@163.com (Y.H.); xiezhihong@263.net (Z.X.)
* Correspondence: y1075993780@163.com; Tel.: +86-13011153629

Received: 17 February 2019; Accepted: 1 April 2019; Published: 4 April 2019

Abstract: In the field of visual tracking, discriminative correlation filter (DCF)-based trackers have made remarkable achievements with their high computational efficiency. The crucial challenge that still remains is how to construct qualified samples without boundary effects and redetect occluded targets. In this paper a feature-enhanced discriminative correlation filter (FEDCF) tracker is proposed, which utilizes the color statistical model to strengthen the texture features (like the histograms of oriented gradient of HOG) and uses the spatial-prior function to suppress the boundary effects. Then, improved correlation filters using the enhanced features are built, the optimal functions of which can be effectively solved by Gauss–Seidel iteration. In addition, the average peak-response difference (APRD) is proposed to reflect the degree of target-occlusion according to the target response, and an adaptive Kalman filter is established to support the target redetection. The proposed tracker achieved a success plot performance of 67.8% with 5.1 fps on the standard datasets OTB2013.

Keywords: correlation filter; spatial prior; color histogram; occlusion judgement

1. Introduction

Target tracking is mainly applied in video surveillance, intelligent traffic, unmanned vehicles, and precision-guided munitions, etc. [1,2] The main task is to locate the target given only position information (usually an axis-aligned rectangle) in the video stream. So far, although research on target tracking has made great progress, it is still difficult to build a robust tracker that can meet several challenges (like light intensity changes, target deformation and rapid movement, background clutter, etc.) at the same time without prior knowledge of the target.

The discriminative correlation filter (DCF)-based algorithms [3–7] are among the most efficient tracking algorithms at present [8]. The main idea is to use the correlation operation to reflect the similarity between the candidate (target sample) and the target template (filter), and then get the most similar sample according to the correlation response peak. Moreover, the correlation operation in the time domain can be implemented in the frequency domain with element-wise operation, which makes the calculation speed of the algorithm much faster than that of other typical trackers [9–11]. However, the standard DCF algorithm requires inter-correlation operations between image blocks, so periodic convolution is used to make up the input data [3], which brings boundary effects that makes the target samples only extractable from a restricted search area. In order to solve the boundary effects caused by circulant-shift matrices, many methods have been proposed. CACF [12] adds the local context to the ridge regression model. SRDCF [13] introduces a spatial regularization function into the process of correlation filtering training. BACF [14] constructs a binary mask upon the larger search window. Nevertheless, these methods only suppress the features outside the fixed range of the sample, which means the features of the background and target within the range are not treated
differently. Therefore, trackers tend to fail when the texture feature of the target are close to those of the surroundings.

A feature-enhanced discriminative correlation filters algorithm (FEDCF) is proposed, where the color histogram is used [15], which can distinguishes target and background distractors effectively, to emphasize the texture feature differences between targets and distractors, as there is usually a prominent difference between them in a color model. Considering that part of the background and the target may have similar color model, to suppress the potential impact of this a spatial prior function is constructed to impose a spatial constraint on the sample [16], which is crucial to suppress the boundary effects and combine with color histogram on a feature level. The established weights aim to enhance the target features and suppress the background-distractor features at the same time, which helps to train the filter that is capable of more powerful distinguishing ability.

In addition, DCF-based trackers can’t handle target-occlusion effectively due to their model updating mechanisms, to address which, the average peak-response difference (APRD) which is sensitive to occlusion is proposed to tackle the target-occlusion with model updating and searching area updating. When a target is judged to be occluded according to APRD, the target model won’t update to retain its purity, and the adaptive Kalman filter [17] based on the APRD parameter is proposed to update the target search area, which aims to ensure that after occlusion the target will still be in the central area where the features are enhanced, rather than in the background area where the features are suppressed by the spatial prior function. The main contributions of the paper are as follows:

1) In order to solve the problem of insufficient expression of texture features (like HOG [18]), a spatial-adaptive weight is constructed by the color-histogram and spatial-prior function, which can increase the discrimination of the texture features between target and background-distractors.

2) Based on a correlation filter framework, a feature-enhanced correlation filter tracker (FEDCF) is proposed. The tracker combines the spatial-adaptive weights with the correlation filter on the feature level, which helps to enhance the target features while suppress the background-distractor features, and thus train the powerful discriminative filter. The Gauss-Seidel iteration [13] is applied to effectively solve the objective function.

3) For target redetection under occlusion, the APRD parameter is proposed, which can be used to reflect the degree of target occlusion and decide model updating. An adaptive Kalman filter based on APRD is proposed to update search area.

4) The proposed FEDCF was compared with the current state-of-the-art trackers on some standard datasets (OTB2013 (Figures S1–S24) [19] and OTB2015 (Figures S25–S48) [20]). The proposed algorithm runs at approximate 5 fps and achieves the top rank in term of accuracy and robustness.

2. Related Works

Target tracking aims to estimate the true trajectory based on initial target position information, and the estimation of the target at each frame may have noise [21–25]. The whole task mainly includes short-time tracking and redetection. The proposed tracker is based on a correlation filtering algorithm, which has been paid great attention because of its dense samples and computational efficiency brought by the cyclic structure. In this section, we briefly review the tracking algorithm based on correlation filtering and the classical target redetection tricks.

2.1. Correlation Filter Based Trackers

The discriminative correlation filter tracker was firstly proposed by Bolme et al. [3]. The method takes the affine transformation of the target grayscale feature as the training sample, and trains the correlation filter that correlates the sample features with the ideal Gaussian response. The target in the next frame can be obtained at the response map peak. MOSSE achieves operation speeds of 600 fps. On this basis Henriques et al. [26] proposed the Kernelized Correlation Filters (CSK) which takes
the circulant shift of the region around the target as the training sample and thus mathematically solves the problem of dense sampling. Therefore, the correlation filter can be learnt by solving the least squares in frequency domain with fast Fourier transform (FFT), and achieves real-time computing speed. Based on this classical tracker, many improved methods have been proposed. As for target features [5,8,27–31], Henriques et al. [4] used multi-channel HOG features to replace grayscale features and introduced a kernel function. Danelljan et al. [7] used a color naming feature (CN) which is dimensionality reduced with PCA. For scale estimation [5,6,32,33], Danelljan et al. [5] used a 32-dimensional scale filter to estimate the scale of the target. However, these correlation filtering algorithms are inevitably affected by the boundary effect. Samples constructed by circulant matrix (see Figure 1), especially at the boundary which can’t represent real feature information. The filter learned from this limited circulant shifted image block is more likely to drift in the case of deformation and motion blur.

![Figure 1](image_url)  
**Figure 1.** Image (b) shows the circulant samples obtained by DCF methods from original image (a), leading to a limited range of features for the filter to be learnt from. As a consequence, the template learnt is not discriminative enough.

To solve the boundary effects, many methods have been proposed [34–38]. Galoogahi et al. [14] used larger a range of images as training samples to enrich the feature information in the search range, and introduced a binary mask to directly crop the training samples at the center where the information is not affected by the boundary. Based on ridge regression equation, Mueller et al. [12] added constraint spatial context, which essentially suppresses the influence of local context on filter training. However, the contexts selected and the confinement degree are invariant. Danelljan et al. [13] added a regularization penalty term to punish the filter coefficients over a large range surrounding the object. However, the penalty function can only suppress the background information according to its distance to the object, which is not a valid method to distinguish the distractors. Considering that the color histogram has been used for target tracking successfully [15], it could be used to recognize the background distractors as there are prominent differences between them and the target in a color model.

2.2. Target Redetection

Occluded targets can pollute the template filter of DCF-based trackers severely, which is the main reason for tracking loss. How to judge the occlusion of the target, update the template pertinently and redetect the target after occlusion, has been what researchers have focused on.

There have been several methods proposed for long-term tracking [39,40]. Kalal et al. proposed TLD [41], which decouples the tracking and detection mission, and uses two groups of independent samples to support them, respectively. In the detection stage, the sliding sub-windows are screened according to the variance to find the possible missing targets. MDP [42] applies a template matching method to build a target contour model on the basis of the TLD algorithm. When the target is lost,
the model stop updating, and when the model can match the target, the model updates to track. LCT [43] performs estimation of translation and scale for the target, respectively. When the response score is lower than the threshold, the target is judged to be occluded, and the online random ferns is applied to re-determine the optimal target location. Bolme et al. [3] used Peak to Sidelobe Ratio (PSR) to define the peak intensity, and thus to judge the occlusion of the target. However, these methods for target occlusion are either time-consuming or not accurate enough. Based on the DCF tracker, a new confidence index (APRD) is proposed to reflect the degree of occlusion, which can effectively prevent models from being polluted by occlusions. In addition, an adaptive Kalman filter with APRD is proposed. Compared with several improved particle filters [22,23] which aim to estimate the complex state space model or dynamic system, and several improved shadowing filters [21–23] which try to optimize the model’s deterministic dynamics including velocity and acceleration, the proposed Kalman filter aims not to model the trajectory of the target but to update the searching area when the model is judged not to update, and as what we will discuss in Section 3.4.2, the step would take almost no extra time. The output of the system adaptively depends on the historical observation from the feature-enhanced correlation filter and the historical predictions from the Kalman filter, which is more consistent with the changing occlusions in real life.

3. Proposed Method

In this section, we introduce our FEDCF tracker. First, in Section 3.1, the overall framework of FEDCF is overviewed. Then in Section 3.2, the feature enhancement method based on color histogram is introduced. The detailed formula of FEDCF is proposed in Section 3.3. Last in Section 3.4, the working principle of APRD confidence index and adaptive Kalman filter is demonstrated.

3.1. The FEDCF Framework

The tracking framework of FEDCF is shown in Figure 2. In the training stage, the input features are exacted and the histogram model is updated according to the searching region obtained in the previous frame. Then the histogram score is combined with the spatial-prior to form the target likelihood, which is used as target adaptive weight. The weight will be used to build new variable \( z \) rather than assigned to a feature directly.

![Diagram](image-url)

**Figure 2.** Main components of our tracking framework. The whole improved methods are represented by blue lines and purple lines. The blue lines denote the improving correlation filter with adaptive weights, which aims to enhance the target feature and suppress the background distractors. The purple
lines denote the treatment to occlusion, which can prevent the target template from occlusion pollution and ensure that the target after occlusion can be redetected at the updated searching area.

In the detection stage, the input features are correlated with the filter learned in the previous frame to obtain the response map, according to which the predicted target can be obtained. The response map will also be used to build APRD confidence and obtain the maximum value \( f_{\text{max}} \), which are designed for judging occlusion. If the APRD and \( f_{\text{max}} \) are less than their threshold value, respectively, the target is judged to be occluded, then the target model will not be updated until the APRD and \( f_{\text{max}} \) are greater than their threshold value. During the period, the adaptive Kalman filter based on APRD will update the search area continually, which aims to guarantee that the target can be located at the center of the searching area. Details of the above are given in Sections 3.2–3.4.

### 3.2. Feature Enhancement Based on Color Histogram

In order to improve the distinction of the features between target and background-distractor, we need to strengthen the target texture feature. Considering that the color features are not easily affected even in the case of rapid deformation, the confidence map can be obtained through the Bayesian classifier \([15]\) based on the color histogram, which can reflect the possibility that the pixels in the searching area belong to the foreground of the target.

Given a target-centered patch \( I \) (see Figure 3), the foreground area \( O \) is the bounding-box of size \( L_O \times H_O \) and the background area \( B \) is the searching area of size \( M \times N \). \( n_x \) represents the bin \( n \) of the color component allocated to the patch pixels \( x \in I \). The target likelihood at \( x \) can be obtained by Bayes’ Rule \([15]\):

\[
P(x \in O | O, B, n_x) \approx \frac{P(n_x|x \in O)P(x \in O)}{\sum_{\Omega \in \{O,B\}} P(n_x|x \in \Omega)P(x \in \Omega)}\tag{1}
\]

The conditional probability is obtained from the color histogram, i.e., \( P(n_x|x \in O) \approx H_O^I(n_x)/N_O \) and \( P(n_x|x \in B) \approx H_B^I(n_x)/N_B \), where \( H_O^I(n_x), H_B^I(n_x) \) denotes the pixel number of histogram \( H \) belonging to \( n_x \) in the target foreground region and the target background region respectively. \( N_O, N_B \) respectively denote the number of pixels in foreground region and background region. In addition, the prior probability can be expressed as \( P(x \in O) \approx N_O/(N_O + N_B) \). Equation (1) is rewritten as:

\[
P(x \in O | O, B, n_x) \approx \frac{H_O^I(n_x)}{H_O^I(n_x) + H_B^I(n_x)}\tag{2}
\]

Histogram score \( w_h \) is obtained from Equation (2) i.e., \( w_h = P(x \in O | O, B, n_x) \), the size of which is \( M \times N \). As Figure 3 shows, \( w_h \) can enhance the feature information which is similar to the target in the searching area. To prevent the potential influence of the enhanced background-distractor, a spatial-prior function \( w_p \) of size \( M \times N \) is defined as follow:

\[
w_p(i,j) = \frac{1}{1 + k \cdot \left( \frac{(i-x_B)^2}{L_O^2} + \frac{(j-y_B)^2}{H_O^2} \right)^2}\tag{3}
\]

where \((i,j)\) is the coordinate of pixels \( x \in I \), \((x_B, y_B)\) is the coordinate of the target center. \( L_O, H_O \) denotes the length and height of foreground region respectively. \( K \) is the rate parameter for suppressing background. As we can see, \( w_p \) almost degrades the boundary features at the rate of \( 1/(1 + k) \), while the features at the center are affected little. The final target likelihood for weighting texture features is then derived below.

\[
w_f = w_h \cdot w_p\tag{4}
\]
Using Jogging for instance (see Figure 3), it can be seen that the color histogram can obviously distinguish a target from its surrounding content. Although, the background-interference which is similar to the target is also enhanced, it can be effectively suppressed by the spatial-prior function (parameter k is set as 5). The target likelihood is finally obtained, which can enhance the target features while suppressing the surrounding-distractors.

![Image](image_url)

**Figure 3.** The input patch is shown as (a), where the foreground and background of the target are extracted from the red and blue bounding boxes, respectively. (b) shows histogram score which is obtained according to the color model in the search area and can reflect the differences between the target and background interference. Spatial prior at (c) indicates that the features at the center are more important than the features around. Combining the histogram score and spatial prior, the target likelihood can enhance the target features while suppressing the surrounding-distractors, which is shown as (d).

### 3.3. Learning Improved Correlation Filter

In correlation filter based algorithm, circulant samples are generated implicitly which cannot be directly weighted. This is different from traditional machine learning methods, such as SVM [44]. Given a searching area of size $M \times N$, the base sample is a one-dimensional vector $x = [x_1, x_2, \cdots, x_{M \times N}]^T$, and the cyclic shifting of x by one element is $Px = [x_{M \times N}, x_1, \cdots, x_{M \times N-1}]^T$. All the cyclic shifting of the sample constitute the training sample $X = [P^1x, P^2x, \cdots, P^{M \times N}x]$, which has d-dimensional feature vector at $x_i(m, n) \in \mathbb{R}^d$. The regression target $y = [y_1, y_2, \cdots, y_{M \times N}] \in \mathbb{R}^{M \times N}$ is of Gaussian distribution which decays smoothly from one to zero. The optimal equation of the feature-enhanced correlation filter (FEDCF) is as follows:

$$
\epsilon(f) = \sum_{k=1}^{MN} \left( \alpha_k \left\| \sum_{l=1}^d (x_k^l \cdot w_l) \ast f^l - y_k \right\|^2 \right) + \lambda \sum_{l=1}^d \left\| f^l \right\|^2
$$

(5)

where $\alpha_k$ denotes the importance of each sample on training the filter, which is uniform in the paper. And $\lambda$ is a regularization coefficient. According to Parseval’s theory, the Equation (5) can be rewritten in the frequency domain as:

$$
\epsilon(\hat{f}) = \left\| \sum_{l=1}^d \left( D(\hat{x}_l) D(\hat{w}_l) \hat{f}^l \right) - \hat{y} \right\|^2 + \lambda \sum_{l=1}^d \left\| \hat{f}^l \right\|^2
$$

(6)

where $D(\hat{w}_l)$ denotes the diagonal matrix with the elements of vector $\hat{w}_l$ in its diagonal. We define $X = [\hat{x}_1 \oplus \cdots \oplus \hat{x}_d]$ and $W = [D(\hat{w}_1) \oplus \cdots \oplus D(\hat{w}_d)]$ of size $dMN \times dMN$, and $F = [\hat{f}_1^T \cdots \hat{f}_d^T]^T$ is the vectorized filter. The Equation (6) is then rewritten in full vector form:

$$
\epsilon(\hat{f}) = \left\| \hat{X} \hat{W} \hat{F} - \hat{y} \right\|^2 + \lambda \left\| \hat{F} \right\|^2
$$

(7)
Since \( \hat{X}\hat{W} \) is not a circulant matrix which cannot obtain a closed solution directly, we define a new variable \( ˆZ = \hat{W} \cdot \hat{F} \) of size \( dMN \times 1 \) in the frequency domain. The objective function can be transformed as a cost function of \( z \):

\[
\varepsilon(z) = \| \hat{X}\hat{Z} - \hat{y} \|^2 + \lambda \| \hat{W}^{-1} \cdot ˆZ \|^2
\]  

(8)

To ensure the fast convergence, the sparse unitary matrix \( B \) of \( dMN \times dMN \) dimension is defined, by which the transformed variable is introduced as follows: \( \hat{B}\hat{X}\hat{B}^H = D, B \cdot \hat{Z} = \hat{Z}, B\hat{W}^{-1}\hat{B}^H = C, \) \( B \cdot \hat{y} = \hat{y} \). Where \( H \) represents the conjugate transpose of the matrix. The cost function (8) is then converted into full-real function:

\[
\varepsilon(z) = \| D\hat{Z} - \hat{y} \|^2 + \lambda \| C \cdot \hat{Z} \|^2
\]  

(9)

Like the way that spatially regularized correlation filter [13] is solved, \( \hat{Z} \) is obtained by solving \( A \cdot \hat{Z} = b \) with Gauss-Seidel iterative. Where:

\[
A = D^T D + \lambda C^T C
\]  

(10)

\[
b = D^T \hat{y}
\]  

(11)

\( A \) is composed of strict upper triangle \( L \) and lower triangle matrices \( U \), i.e., \( A = L + U \). The iterative process for \( \hat{Z} \) is as follows:

\[
A_t = (1 - \gamma)A_{t-1} + \gamma \left( D_t^T D_t + \lambda C_t^T C_t \right)
\]  

(12)

\[
b_t = (1 - \gamma)b_{t-1} + \gamma D_t^T \hat{y}_t
\]  

(13)

\[
L_t \hat{Z}_t^k = b_t - U_t \hat{Z}_t^{k-1}
\]  

(14)

\( K \) and \( t \) denotes the \( k_{th} \) iteration and the \( t_{th} \) frame respectively. \( \gamma > 0 \) denotes the parameter of the learning rate. At the initial frame, \( \hat{Z}_0 \) is defined as \( \left( D_1^T D_1 + \lambda C_1^T C_1 \right)^{-1} D_1^T \hat{y}_1 \) according to Equations (10) and (11), then \( \hat{Z}_t^{0} \) at \( t_{th} \) frame is assigned by \( \hat{Z}_t^{NGS} \). And \( NG_S \) is defined as the maximum number of iterations.

At the detecting stage, the sample \( \sigma \) with the highest response value is selected as the target among the sample set \( \{v_{1,r}, v_{2,r}, \cdots, v_{M \times N, r}\} \), where the sample \( \{v_{k,r} \mid \gamma \in \left\{ \left| \frac{1-S}{2} \right|, \cdots, \left| \frac{S-1}{2} \right| \} \} \) is converted from \( v_k \) at different scales \( r \). \( S \) is the total number of scales, and the target \( \sigma \) is obtained as follows:

\[
\sigma = \arg \max_{v_{k,r}} \sum_{f=1}^{d} (v_{k,r} \cdot w_f) \ast f^l = \arg \max_{\hat{v}_{k,r}} \mathcal{F}^{-1}(\hat{v}_{k,r} \cdot \hat{Z})
\]  

(15)

The whole complexity of the tracking process is \( O(dSMN \log MN + (d + K^2)dMNN_{GS}) \). Here, \( d \) denotes the feature dimension, and \( MN \) is the size of the searching area. The total number of scales is \( S \), and \( k \) denotes the non-zero term of \( \hat{w}_f \). The computational complexity is mainly derived from the scale estimation and Gauss-Seidel iterative process.

3.4. Target Redetection Mechanism

When the occlusion appears in the searching area, the target template would inevitably be corrupted, however most existing correlation filter trackers [4,5,31] do not consider it when updating the template. In general, the maximum response value of the sample can reflect the confidence of the detected target, but can’t judge occlusion effectively. Therefore, we propose a novel criterion called average peak-response difference (APRD), which can reflect the degree of occlusion sensitively, and helps to decide whether to update the template or not. In addition, according to object’s historical
position and state of occlusion, an adaptive Kalman filter is constructed based on APRD, which can provide the most likely searching area for the re-detection of the target after occlusion.

3.4.1. Model Update

The response map of all circulant samples are obtained as follows:

\[ f(x, w, f) = F^{-1}(\hat{X}\hat{W}\hat{F}) \]  

The maximum and minimum response were defined as \( f_{\text{max}} \) and \( f_{\text{min}} \) respectively. The APRD parameter is defined as:

\[ \text{APRD} = \frac{|f_{\text{max}} - f_{\text{min}}|}{\sum_x|f(x, w, f) - f_{\text{min}}|/(MN)} \]  

Equation (17) is composed of a numerator and a denominator. The denominator implies the average difference of the response map, which means the similarity of whole samples to the template. The numerator implies the maximum difference of the response map, which means the confidence of the most likely target candidates.

Using *Jogging* as an example (see Figure 4) the targets in Figure 4a,c are not occluded yet, so the response maps show one sharp peak, which indicates one definite object. Compared with which, the response map in Figure 4b, where the target suffers from occlusion, shows multiple peaks. Although the peak value does not decrease sharply, the average response difference caused by multiple peaks increases remarkably, and thus APRD decreases significantly.

\[ \text{Th}_1 = \beta_1 \text{mean}(\sum \text{APRD}) \]

\[ \text{Th}_2 = \beta_2 \text{mean}(\sum f_{\text{max}}) \]  

Figure 4. The first row are the shots of the sequence *Jogging* from OTB2013, where the red and blue bounding box denote the target and the searching area of FEDCF, respectively. The second row are response maps corresponding to the scenario above. (a), (b) and (c) respectively show the scenarios before occlusion, during occlusion and after occlusion.

Figure 5 shows that the APRD and \( f_{\text{max}} \) have consistent variation trends during the period (64th frame to 81st frame) when the target is occluded, so the occlusion can be judged by these two parameters with a certain threshold. The thresholds are defined as:

\[ \begin{align*} 
\text{Th}_1 &= \beta_1 \text{mean}(\sum \text{APRD}) \\
\text{Th}_2 &= \beta_2 \text{mean}(\sum f_{\text{max}}) 
\end{align*} \]  

**Figure 5.** The blue curve and the red curve above respectively show how \( f_{\text{max}} \) and APRD vary between the 45th frame to the 95th frame. The occlusion occurred between the 64th frame to the 81th frame.
When APRD and $f_{\text{max}}$ are smaller than their respective historical average values with ratios $\beta_1, \beta_2$, the target is judged to be occluded until the parameters are higher than their thresholds. During the period, the template stops updating and the search area is updated by the Kalman filter. After the occlusion is judged to disappear, the FEDCF tracker redetects the target in the newly updated search area.

![Figure 5](image)

**Figure 5.** The blue curve and the red curve above respectively show how $f_{\text{max}}$ and APRD varies between the 45th frame to the 95th frame. The occlusion occurred between the 64th frame to the 81th frame.

### 3.4.2. Update the Searching Area

A pure target template is the premise of the target re-detection, and updating the search area reasonably is the key for the target to be redetected successfully after occlusion. Considering that the spatial prior function in Section 3.2 is used to suppress the features surrounding the target box, it is necessary to ensure that the target feature will still be strengthened at the center of the search area after the occlusion. During the period of occlusion, the mission of target prediction could be undertaken by a Kalman filter which is good at using historical target states to predict the target location, of which the observed value are defined as the output of the FEDCF tracker. In addition, since the targets are located with different level of deviation because of their various degree of occlusion, an adaptive Kalman filter is constructed with APRD, which adaptively changes the reliability of the observed values for the Kalman filter. The mathematical model of the Kalman filter is as follows:

**State equation:**

$$X(k + 1) = A(k + 1, k)X(k) + w(k)$$  \hspace{1cm} (19)

**Prediction equations:**

$$Z(k) = H(k)X(k) + u(k)$$  \hspace{1cm} (20)

Here, $X(k)$, $Z(k)$ denote the vector of state and observation, respectively. The state transition matrix is defined as $A(k + 1, k)$, and $H(k)$ denotes the observation matrix. The vectors of system noise and observed noise are defined as, $w(k), u(k)$ respectively. The predictive equations and update equations are as follows:

**Predictive equations:**

$$\begin{align*}
X'(k + 1|k) &= A(k + 1,k)X'(k|k) \\
P(k + 1|k) &= A(k + 1,k)P(k|k)A^T(k + 1,k) + Q(k)
\end{align*}$$  \hspace{1cm} (21)

**Update equations:**

$$\begin{align*}
K(k + 1) &= P(k + 1|k)H^T(k + 1)[H(k + 1)P(k + 1|k)H^T(k + 1) + R(k + 1)]^{-1} \\
X'(k + 1|k + 1) &= X'(k + 1|k) + k(k + 1)[Z(k + 1) - H(k + 1)X'(k + 1|k)] \\
P(k + 1|k + 1) &= [I - k(k + 1)H(k + 1)]P(k + 1|k)
\end{align*}$$  \hspace{1cm} (22)

Here, $P(k)$ denotes the prior estimated covariance at the k-th frame, and $K(k + 1)$ represents the kalman gain. The covariance of the predicted noise and the observed noise at kth frame are defined as, respectively.
Because $Q(k), R(k)$ imply the credibility of the predicted value and the observed value for the Kalman filter, respectively, which can be expressed by the APRD parameter. The adaptive $Q(k), R(k)$ at frame $t$ is defined as follows:

$$
\begin{cases}
    R(t) = 1/\text{APRD}, & Q(t-1) = 1 - 1/\text{APRD} \quad \text{APRD} \geq Th_1 \\
    R(t) = \infty, & Q(t-1) = 0 \quad \text{APRD} < Th_1
\end{cases}
$$

(23)

Here, $Th_1$ is the threshold of APRD for the template to stop updating. Equation (23) shows that, when the target is judged to be occluded, the Kalman filter no longer accept the observed value coming from the FEDCF tracker, but only predict the target according to the historical state of the target. When the occlusion is not sufficient enough to occlude target, the smaller APRD is, the smaller the effect FEDCF has on the Kalman filter. When the target is judged to reappear after occlusion at t frame, we construct the search area and extraction feature patch based on the current prediction of the Kalman filter, and then enter the detection stage at the next frame. Note that, APRD and $f_{\text{max}}$ are obtained at the detection stage which takes no extra time. The Kalman filter simply uses $3 \times 3$ matrixes to make one-step prediction at each frame, so the time consumed for the redetection stage is almost negligible.

The outline of the whole tracking framework is presented in Table 1.

**Table 1.** The general flow of the whole FEDCF tracking algorithm.

| Input:       | Frames $\{I_t\}_{t=1}^T$, initial target location $x_1$, spatial prior $w_p$. |
|--------------|--------------------------------------------------------------------------|
| Output:      | Detected target location at each frame $\{x_t\}_{t=1}^T$.               |
| 1:           | repeat                                                                  |
| 2:           | Crop a searching patch $S_t$ which is centered at $x_t$.                |
| 3:           | Exact feature samples $V_t$ and construct the color histogram model via (2). |
| 4:           | Construct the adaptive weights $w_t$ via (4).                           |
| 5:           | Obtain APRD and $w_y$ according to (15)–(17).                          |
| 6:           | If target is judged to be occluded via (18).                           |
| 7:           | Predict the target location $x_{t+1}$ by adaptive Kalman filter via (19)–(23). |
| 8:           | Else                                                                    |
| 9:           | Update the correlation model via (12)–(13).                            |
| 10:          | Detect the target location with certain scale $x_{t+1,r}$ via (15).     |
| 11:          | End if                                                                 |
| 12:          | Until end of video sequence.                                           |

4. Experiments

We validated out FEDCF tracker on the standard datasets OTB2013 [19] and OTB2015 [20] on MATLAB R2014b. The experiments were carried out on a PC with an Intel I5 2.60 GHz CPU and 8 GB RAM. OTB2013 and OTB2015 contain 50 and 100 video sequences, respectively, with 11 challenging attributes, i.e., illumination variation (IV), out-of-plane rotation (OPR), scale variation (SV), occlusion (OCC), deformation (DEF), motion blur (MB), fast motion (FM), in-plane rotation (IPR), out of view (OPR), background clutter (BC), low resolution (LR).

The trackers are evaluated with one-pass evaluation (OPE), the evaluation criteria are the precision plots and the success plots. The precision score (PR) is calculated as the ratio of the frames where the Euclidean distance between the estimated target center and the ground-truth center is smaller than a certain threshold. The precision plot is plotted over different thresholds and the threshold in our experiment is set to 20 pixels. The success score is calculated as the percentage of the frames in which the overlap ratio is larger than a certain threshold. The overlap ratio is defined as $|a \cap b| / |a \cup b|$,.
where \( a, b \) respectively denote the estimated target bounding box and the ground-truth bounding box. The success plot ranks the trackers according the area under the curve (AUC) of it.

### 4.1. Detailed Parameters

In our proposed FEDCF method, the size of searching area is empirically set to the 2.5 times the target bounding box. The features exacted at the search area are HOG and CN features, where CN features are used to construct a color histogram with 32 bins in each channel. The parameter \( k \) in the spatial prior function is set to 5. HOG is the basic feature set used to describe the target, the cell of which is set to \( 4 \times 4 \) with 31 dimensions, which follows the setting of traditional DCF tracker \([5]\). In the detection stage, we empirically set the learning ratio \( \gamma \) to 0.025 in (12), and the regularization coefficient \( \lambda \) to 0.8. The number of iterations \( N_{GS} \) is set to 4, and the number of target scales \( S \) is 7. In the redetecting stage, \( \beta_1, \beta_2 \) determine the sensitivity of model updating to the background clutter, and when which tend to 0 or 1, the performance of redetection becomes worse. So according to the experimental tests, both \( \beta_1, \beta_2 \) are set to 0.5 in (18). Considering that the target starts moving slowly in the initial frames, the kalman filter parameters are set as follows:

\[
A(k+1,k) = \begin{bmatrix} 1 & 1 & 0.5 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}, H(k) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}, P(1|1) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 1 \end{bmatrix}
\]

The whole parameters are fixed over all video sequences, with which the proposed FEDCF is implemented on standard laptop with 5 frames/s.

### 4.2. Analysis of the FEDCF

Our proposed FEDCF can exact powerful sample features based on a correlation filter framework, where the image patch inevitably suffers from boundary effects. Our proposed improved method can be divided into two aspects, which are strengthening the target and suppressing the background-distractors. Compared with SRDCF, which can be viewed as the state-of-the-art tracker for solving boundary effects, FEDCF improves the DCF tracker at the feature level. First, FEDCF uses a color histogram to reflect the differences between targets and surroundings-distractors in the color model. As we can see from Figure 6, the target in *Bolt* is specified by the red bounding box, and the search area contains background distractors. Because not only target but also the part of distractors which are similar to the target in the statistical color model are enhanced, the spatial-prior is introduced to suppress the distractors which are enhanced by mistake, and the boundary effect is also restrained in the same way.

In order to prove that our proposed improved methods are effective even in the case of target deformation, FEDCF and SRDCF, KCF \([4]\) are compared in the scenes annotated with deformation in OTB2015. The screenshots of the tracking performances are shown as Figure 7.

From Figure 7 we can see, the target in *Bird 1* changes its shape drastically, and the targets in *Skating_1,2* suffer from interference of background-distractors. FEDCF can track the target persistently and accurately, while SRDCF and KCF drift to the background after a period of tracking. Note that the targets and background-distractors are clearly distinguishable in the statistical color model, and the color histogram of the target is very stable, even when target is undergoing deformation. Although SRDCF constrains the filter coefficients over boundary space where the background-distractors are most likely be, when the distractors surrounding the target are changing shape, SRDCF can’t overcome their interference effectively.
Figure 6. Column (a) displays the screenshots from Bolt at frames 75, 90 and 125. The target and search area are respectively specified by the red and blue bounding boxes. The effects of enhancements with histogram scores are visualized as column (b). The final weights assigned to features are shown are column (c).

Figure 7. The screenshots from Skating_1, Bird_1 and Skating_2, where the targets are the female skater, bird at the screen center and the male skater, respectively. The target results of FEDCF, SRDCF and KCF are respectively highlighted with red, green and blue bounding boxes.

In order to demonstrate the contributions of each proposed modules, we carried out an ablation experiment on OTB2015. Considering that FEDCF is an improved tracker based on DCF, and the feature-enhancement we used is similar to DAT [15] in principle, so we used DCF [4] as the baseline tracker and treated DAT as the compared tracker. The performances of the trackers are shown as Table 2.
Table 2. The evaluation of OPE in term of scores of success and precision plots on OTB2015. The top score is marked in red.

| Trackers               | Success Score (%) | Precision Score (%) | Frames Per Second |
|------------------------|-------------------|---------------------|-------------------|
| DCF                    | 47.7              | 69.5                | 240.3             |
| DCF+ enhancement       | 55.9              | 76.2                | 14.9              |
| DCF+ enhancement+ prior spatial | 62.7          | 81.6                | 5.3               |
| FEDCF                  | 65.4              | 86.4                | 5.2               |
| SRDCF                  | 59.8              | 78.8                | 5.5               |
| DAT                    | 33.3              | 43.2                | 15.6              |

As we can see, using a color histogram to enhance the HOG features can improve the DCF obviously, though the broken circulant structure obviously increases the computational complexity. However, the performance is inferior to that of SRDCF until the prior spatial function is utilized, because it’s not enough to tackle the boundary effects only by feature-enhancement. Besides which, FEDCF also introduces the adaptive Kalman filter, and the extra module takes nearly no more time, but improves the performance of the tracker in terms of success score and precision score significantly, which validates the effectiveness of our proposed redetection mechanism. Among all the trackers, DAT performs worst, which proves that although the color histogram can make texture features more robust in a complex environment, the single color model without texture features is unable to present the target, especially in the case of deformation and illumination variations where the color information is almost distorted.

4.3. The evaluation on OTB

The proposed FEDCF was evaluated with eight state-of-the-art trackers and one color tracker on the OTB2013 and OTB2015 datasets. The compared trackers include DCF-based trackers: KCF [4], DSST [5], SAMF [6], a tracker that combines color and texture features: STAPLE [31], a long-term trackers good at handling occlusion (LCT [43]) trackers effective at solving the boundary effect like BACF [14], SRDCF [13], and CACF [12], as well as a color tracker which only uses the color histogram (DAT). KCF uses a kernel function and multiple channel HOG features to extend DCF, which can be viewed as a baseline of the DCF-based tracker. STAPLE utilizes the advantages of the color histogram and HOG in describing targets from the color model and texture feature level, and combining them on a response map, which performs better than KCF and DAT [15] individually. DSST and SAMF respectively add the estimation of scale into the target tracking framework with different strategies. Based on DCF, LCT uses an online fern classifier to redetect targets. BACF uses a binary mask on the expanded search area, to achieve samples extractions with no boundary effects. CACF introduces a spatial constraint into the filter learning, which suppresses the potential effect of the local contexts. SRDCF uses regularization function to suppress the filter coefficients with Gauss distribution. All the code of compared trackers were obtained from the respective authors’ websites or open data sources. DAT utilizes a color histogram to construct the color model, according to which it matches the target.

4.3.1. State-of-the-Art Comparison

The proposed FEDCF is evaluated with precision plots and success plots on OTB2013 with the compared trackers, the results are shown as Figure 8. FEDCF runs on average at 5.1 frames/s. This is because that FEDCF introduces adaptive weights constructed by the color histogram on a $M \times N$ searching area, which destroys the circulant matrix structure, and thus requires Gauss-Seidel iteration to solve the optimal equation. However, the contribution of this high time complexity is that FEDCF is capable of distinguishing the target and surrounding-distractors. As a result, FEDCF ranks first in terms of PR (88.9%) and AUC (67.8%) metrics, respectively. Compared with KCF which is
the baseline of the DCF-based tracker, our FEDCF outperforms it by 14.9% and 16.4% in terms of PR and AUC scores. STAPLE fuses the color histogram and HOG features on the response level, which makes the target detection robust to deformation, compared with which, the proposed FEDCF utilizes the color histogram on a feature level, where it works as the weights of the texture features, and thus the spatial prior function can effectively constrain the boundary effects. Although the speed of FEDCF is slower than that of STAPLE, FEDCF performs better than STAPLE by 10.7%/8.5% in terms of PR/AUC. SRDCF is famous for suppressing boundary effects, and our FEDCF outperforms it by 5.1%/5.0%. LCT, which is good at solving occlusion problems in long-term tracking, shows inferior performance than FEDCF by 4.1%/8.5%. In addition, the proposed FEDCF performs better than other DCF-based trackers such as SAMF, DSST, CACF and BACF. DAT which only uses color model for target tracking performs worst among all the trackers, and FEDCF surpasses it by 47.8%/36.4%, which demonstrates that the target tracking in a complex environment depends more on texture features than on color features, thus it’s reasonable to utilize the color histogram as the weights for texture features. Overall, the proposed FEDCF achieves highly competitive performance against state-of-the-art trackers on OTB2013.

For a more reliable evaluation of our FEDCF, we use OTB2015 which expands the number of challenging video sequence to 100. The overall precision plots and success plots of FEDCF and compared trackers with one pass evaluation are shown in Figure 9. The proposed FEDCF still ranks first against other seven state-of-the-art trackers. FEDCF achieves precision plot PR score of 86.4% and success plots AUC score of 65.4%, which outperforms the second tracker SRDCF by 7.6%/5.6%. As for the compared trackers, SRDCF, BACF, STAPLE perform better than others, which means our FEDCF is more robust than them.
Figure 9. The precision plots and success plots of FEDCF and compared trackers with one-pass (OPE) evaluation on OTB2015, which expands the contained sequences to 100 challenging sequences with 11 challenging attributes.

4.3.2. Attribute-Based Evaluation

For a more detailed analysis of trackers’ performances in different scenarios, the one-pass evaluations of trackers in term of precision plots on different challenging scenarios are shown as Table 3.

Table 3. The one-pass evaluation of FEDCF and compared trackers in term of PR score on OTB2015. The top three scores are highlighted in red, green and blue respectively.

|       | FEDCF | SRDCF | STAPLE | BACF | CACF | DSST | SAMF | KCF | LCT | DAT |
|-------|-------|--------|--------|------|------|------|------|-----|-----|-----|
| Iv(37)| 83.7  | 78.1   | 77.8   | 75.8 | 76.0 | 72.0 | 74.0 | 72.4| 74.3| 33.4|
| OPR(63)| 85.0 | 74.0   | 73.8   | 75.3 | 69.2 | 66.5 | 75.0 | 67.6| 74.6| 45.4|
| SV(63)| 84.0  | 74.3   | 72.4   | 75.6 | 69.4 | 65.3 | 72.2 | 63.5| 67.8| 41.4|
| OCC(48)| 81.0 | 72.7   | 72.4   | 69.6 | 66.9 | 60.2 | 73.6 | 63.2| 67.8| 42.4|
| DEF(43)| 84.1 | 73.0   | 74.7   | 72.6 | 69.2 | 56.0 | 67.1 | 61.9| 68.5| 46.9|
| MB(29)| 82.6  | 76.7   | 69.9   | 58.4 | 71.1 | 57.0 | 68.7 | 60.0| 66.9| 34.3|
| FM(39)| 80.2  | 76.9   | 71.0   | 68.9 | 72.7 | 57.5 | 70.6 | 62.1| 68.1| 38.1|
| IPR(51)| 81.1 | 74.2   | 76.8   | 76.0 | 75.1 | 71.1 | 74.4 | 70.1| 78.1| 44.3|
| OV(14)| 76.6  | 60.2   | 66.8   | 74.8 | 57.8 | 48.0 | 67.6 | 49.9| 59.2| 36.5|
| BC(31)| 87.2  | 77.5   | 74.9   | 80.1 | 76.4 | 70.4 | 72.2 | 71.2| 17.2| 37.1|
| LR(9)| 73.7  | 66.3   | 61.0   | 74.1 | 59.4 | 60.2 | 68.4 | 56.0| 53.7| 39.3|
| Overall| 86.4  | 78.8   | 78.4   | 77.1 | 74.4 | 69.3 | 76.4 | 69.5| 76.2| 43.2|

Our approach ranks first among all the trackers in ten challenging scenarios. The evaluated scores are respectively illumination variation (83.7%), out-of-plane rotation (85.0%), scale variation (84.0%), occlusion (81.0%), deformation (84.1%), motion blur (82.6%), fast motion (80.2%), in-of-plane rotation (81.1%), out-of-view (76.6%) and background clutter (87.2%). Note that, most contributed low resolution sequences are gray sequences, which do not support building a color model. Thus, both FEDCF and STAPLE, which utilize the color histogram, have worse performances in the low resolution challenge compared with their performances on the other ten challenges. However, among the eight state-of-the-art trackers, FEDCF ranks second on the low resolution challenge. Compared with SRDCF, our FEDCF performs more robustly, especially when undergoing occlusion, deformation, and motion blur, which proves that our proposed methods can significantly
improve the performance of DCF-based trackers. This is attributed to two reasons. One of the reasons is that the adaptive weights constructed from the color histogram and spatial prior could effectively enhance the target features even when undergoing deformation, and another reason is our proposed method of dealing with occlusion, which show that using the predicted target location from the adaptive Kalman filter to update the search area during occlusion is feasible, and can be applied to other DCF-based trackers which have small search area sizes. Overall, our proposed method shows more competitive performance compared with other DCF-based trackers.

In order to intuitively show the compared performances of the trackers on challenging video sequences in the OTB2015 dataset, we show screenshots of eight challenging videos from OTB2015 (see Figure 10). Because of space limitations, we only compare our FEDCF with the top four trackers: BACF, SRDCF and Staple, and their estimated bounding boxes are respectively shown as the red, green, black and blue rectangular boxes. The sequences in first two row (Blurcar1, BlurOwl) mainly suffer from serious motion blur. In the Blurcar1 image, BACF and STAPLE lost the target at the beginning and drift to the background, and only our FEDCF and SRDCF finish the tracking mission with great ability by utilizing HOG features. The target in BlurOwl undergoes even more serious motion blur where the target feature is kind of similar to the background, and BACF still loses the target at the start. As we can see, STAPLE hardly catches the target by matching with the color histogram. As for SRDCF, which only relies on texture features, the estimation of scale is not accurate enough, however, our FEDCF can keep up with the target accurately. The sequences of the next four rows (Box, Human3, Jogging, Lemming) can mainly be considered occlusion scenes. When the target in Jogging is occluded, only our FEDCF and SRDCF redetect the target successfully after its reappearance. The success of SRDCF can be attributed to its larger searching area which is four times the target size and that the moving distance of the target is not too long in the plane because of the moving camera. Although the search area of our FEDCF is smaller (2.5 times the target area), it still performs well. In the Human3 scene where the camera is fixed, not only BACF and STAPLE, but also SRDCF lost target the finally. Our proposed FEDCF is the only tracker which doesn’t lose the target. This is because FEDCF utilizes the Kalman filter to update the search area which ensures that the target will still be located at the center of the search area where the features are enhanced after occlusion. The target in Lemming undergoes a brief occlusion, which only FEDCF and BACF can tackle well. The scene of Box is similar to Lemming, however the target suffers from occlusion for over 10 frames. Only FEDCF catches the target successfully all the way, which is due to the fact that FEDCF uses APRD to stop updating the template immediately when the target is occluded and effectively prevents the template from being corrupted. The sequences of the last two rows (Shaking, Skating _1) mainly suffer from illumination variation and deformation. In the Shaking scene, SRDCF and STALE lost the target after drastic lighting changes, while our FEDCF and BACF show great resistance. As to Skating _1, BACF and STAPLE lost its target which undergoes deformation, then SRDCF drifts to the background due to the illumination variation. Only our FEDCF keeps its tracking box around the target, which proves that under challenging scenarios like illumination and deformation, the texture features like HOG are insufficient to express the target. However, colors histogram and spatial priors can effectively enhance the difference between the target and background-distractors, which is hardly affected by deformation and illumination variation. In conclusion, our proposed methods of adaptive weights and redetection methods can successfully handle different challenges, especially occlusion, deformation, and motion blur.
Figure 10. The screenshots of the tracking process on eight challenging sequences. The results of trackers (FEDCF, BACF, STAPLE and SRDCF) are respectively marked with red, green, blue and black rectangles.

5. Discussion and Conclusions

A feature-enhanced discriminative filter (FEDCF) is proposed for visual tracking. A novel method using color histograms and spatial prior functions to construct the adaptive features is designed to enhance the differences between the target and background distractors on a feature level. In addition, an effective target redetection mechanism is proposed, which uses the proposed APRD parameter to judge model updating and construct an adaptive Kalman filter to effectively handle target occlusion.
Comparative experiments demonstrate that FEDCF can handle the most challenging scenes in OTB2013 and OTB2015. Compared with SRDCF, STAPLE and BACF, the proposed approach performs better, especially under the scenarios of deformation, motion blur and occlusion. Moreover, the proposed method for target occlusion can be applied to other DCF-based trackers to improve their robustness in complex environments. Our future work will focus on simplifying the optimal function and improving the speed of the algorithm.

Supplementary Materials: The following are available online at http://www.mdpi.com/1424-8220/19/7/1625/s1. Both the evaluations on OTB2013 and OTB2015 are provided as 24 figures. Figures S1–S11: Precision plots of OPE on 11 challenging attributes. Figures S12–S24: success plots of OPE on 11 challenging attributes.

Author Contributions: Validation, Y.H.; Writing—original draft, H.Y.; Writing—review & editing, Z.X.

Funding: This research received no external funding.

Acknowledgments: This work is supported by the National Natural Science Foundation of China under Grant 61772109 and 61771165.

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

References

1. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–25 June 2005; Schmid, C., Soatto, S., Tomasi, C., Eds.; IEEE Computer Soc.: Los Alamitos, CA, USA, 2005; Volume 1, pp. 886–893.

2. Yilmaz, A.; Javed, O.; Shah, M. Object tracking: A survey. ACM Comput. Surv. 2006, 38, 45. [CrossRef]

3. Lee, K.-H.; Hwang, J.-N. On-Road Pedestrian Tracking Across Multiple Driving Recorders. IEEE Trans. Multimed. 2015, 17, 1429–1438. [CrossRef]

4. Bolme, D.S.; Beveridge, J.R.; Draper, B.A.; Lui, Y.M. Visual object tracking using adaptive correlation filters. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision & Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010.

5. 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]

6. Danelljan, M.; Häger, G.; Khan, F.; Felsberg, M. Accurate scale estimation for robust visual tracking. In Proceedings of the British Machine Vision Conference, Nottingham, UK, 1–5 September 2014.

7. Li, Y.; Zhu, J. A Scale Adaptive Kernel Correlation Filter Tracker with Feature Integration. In Computer Vision—ECCV 2014 Workshops; Agapito, L., Bronstein, M.M., Rother, C., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2015; Volume 8926, pp. 254–265.

8. Danelljan, M.; Khan, F.S.; Felsberg, M.; Weijer, J.V.D. Adaptive Color Attributes for Real-Time Visual Tracking. In Proceedings of the IEEE Conference on Computer Vision & Pattern Recognition, Columbus, OH, USA, 23–28 June 2014.

9. Kristan, M.; Leonardis, A.; Matas, J.; Felsberg, M. The Visual Object Tracking VOT2017 challenge results. In Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops, Santiago, Chile, 7–13 December 2017; pp. 1949–1972.

10. Zhang, K.H.; Zhang, L.; Liu, Q.S.; Zhang, D.; Yang, M.H. Fast Visual Tracking via Dense Spatio-temporal Context Learning. In Computer Vision—ECCV 2014; Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T., Eds.; Lecture Notes in Computer Science; Springer International Publishing Ag: Cham, Switzerland, 2014; Volume 8693, pp. 127–141.

11. Comaniciu, D.; Meer, P. Mean shift: A robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell. 2002, 24, 603–619. [CrossRef]

12. Nummiaro, K.; Koller-Meier, E.; Van Gool, L. An adaptive color-based particle filter. Image Vis. Comput. 2003, 21, 99–110. [CrossRef]

13. Mueller, M.; Smith, N.; Ghanem, B. IEEE: Context-Aware Correlation Filter Tracking. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; IEEE: New York, NY, USA, 2017; pp. 1387–1395.
14. Danelljan, M.; Hager, G.; Khan, F.S.; Felsberg, M. IEEE: Learning Spatially Regularized Correlation Filters for Visual Tracking. In Proceedings of the 2015 IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 4310–4318.
15. Galoogahi, H.K.; Fagg, A.; Lucey, S. IEEE: Learning Background-Aware Correlation Filters for Visual Tracking. In Proceedings of the 2017 IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1144–1152.
16. Possegger, H.; Mauthner, T.; Bischof, H. In Defense of Color-based Model-free Tracking. In Proceedings of the 2015 IEEE Conference on Computer Vision & Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
17. Du, F.; Liu, P.; Zhao, W.; Tang, X.L. Spatial-temporal adaptive feature weighted correlation filter for visual tracking. Signal Process.-Image Commun. 2018, 67, 58–70. [CrossRef]
18. Shen, Y.Q.; Gong, H.J.; Xiong, Y. Application of Adaptive Extended Kalman Filter for Tracking a Moving Target. Comput. Simul. 2007, 24, 210–213.
19. Wu, Y.; Lim, J.; Yang, M.-H. IEEE: Online Object Tracking: A Benchmark. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2411–2418.
20. Zaitouny, A.; Stemler, T.; Small, M. Tracking a single pigeon using a shadowing filter algorithm. Ecol. Evol. 2017, 7, 4419–4431. [CrossRef] [PubMed]
21. Barrett, L.; Henzi, S.P. Particle Filters for Positioning, Navigation and Tracking. IEEE Trans. Signal Process. 2002, 50, 425–437.
22. Miguez, J. Analysis of selection methods for cost-reference particle filtering with applications to maneuvering target tracking and dynamic optimization. Digit. Signal Process. 2007, 17, 787–807. [CrossRef]
23. Zaitouny, A.A.; Stemler, T.; Judd, K. Tracking Rigid Bodies Using Only Position Data: A Shadowing Filter Approach based on Newtonian Dynamics. Digit. Signal Process. 2017, 67, 81–90. [CrossRef]
24. Zaitouny, A.; Stemler, T.; Algar, S.D. Optimal Shadowing Filter for a Positioning and Tracking Methodology with Limited Information. Sensors 2019, 19, 931. [CrossRef] [PubMed]
25. Henriques, J.F.; Rui, C.; Martins, P.; Batista, J. Exploiting the Circulant Structure of Tracking-by-Detection with Kernels. In European Conference on Computer Vision; Springer: Berlin/Heidelberg, Germany, 2012; pp. 702–715.
26. Tang, M.; Feng, J. IEEE: Multi-kernel Correlation Filter for Visual Tracking. In Proceedings of the 2015 IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3038–3046.
27. Bibi, A.; Ghanem, B. IEEE: Multi-Template Scale-Adaptive Kernelized Correlation Filters. In Proceedings of the 2015 IEEE International Conference on Computer Vision Workshop, Santiago, Chile, 7–13 December 2015.
28. Choi, J.; Chang, H.J.; Jeong, J.; Demiris, Y.; Choi, J.Y. IEEE: Visual Tracking Using Attention-Modulated Disintegration and Integration. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4321–4330.
29. Li, X.; Liu, Q.; He, Z.; Wang, H.; Zhang, C.; Chen, W.-S. A multi-view model for visual tracking via correlation filters. Knowl.-Based Syst. 2016, 113, 88–99. [CrossRef]
30. Bertinetto, L.; Valmadre, J.; Golodetz, S.; Miksik, O.; Torr, P.H.S. IEEE: Staple: Complementary Learners for Real-Time Tracking. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1401–1409.
31. Montero, A.S.; Lang, J.; Laganiere, R. IEEE: Scalable Kernel Correlation Filter with Sparse Feature Integration. In Proceedings of the 2015 IEEE International Conference on Computer Vision Workshop, Santiago, Chile, 7–13 December 2015.
32. 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]
33. Galoogahi, H.K.; Sim, T.; Lucey, S. IEEE: Correlation Filters with Limited Boundaries. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 4630–4638.
34. Gundogdu, E.; Alatan, A.A. IEEE: Spatial windowing for correlation filter based visual tracking. In Proceedings of the 2016 IEEE International Conference on Image Processing ICIP, Phoenix, AZ, USA, 25–28 September 2016; pp. 1684–1688.
35. Lukeziec, A.; Vojir, T.; Zajc, L.C.; Matas, J.; Kristan, M. IEEE: Discriminative Correlation Filter with Channel and Spatial Reliability. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; IEEE: New York, NY, USA, 2017; pp. 4847–4856.
36. Hu, H.; Ma, B.; Shen, J.; Shao, L. Manifold Regularized Correlation Object Tracking. *IEEE Trans. Neural Netw. Learn. Syst.* 2018, 29, 1786–1795. [CrossRef] [PubMed]

37. Danelljan, M.; Robinson, A.; Khan, F.S.; Felsberg, M. Beyond Correlation Filters: Learning Continuous Convolution Operators for Visual Tracking. In *Computer Vision—ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2016; Volume 9909, pp. 472–488.

38. Ma, C.; Huang, J.-B.; Yang, X.; Yang, M.-H. Adaptive Correlation Filters with Long-Term and Short-Term Memory for Object Tracking. *Int. J. Comput. Vis.* 2018, 126, 771–796. [CrossRef]

39. Hong, Z.; Chen, Z.; Wang, C.; Mei, X.; Prokhorov, D.; Tao, D. IEEE: MUlti-Store Tracker (MUSTer): A Cognitive Psychology Inspired Approach to Object Tracking. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 749–758.

40. Kalal, Z.; Mikolajczyk, K.; Matas, J. Tracking-Learning-Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 2012, 34, 1409–1422. [CrossRef] [PubMed]

41. Xiang, Y.; Alahi, A.; Savarese, S. Learning to Track: Online Multi-object Tracking by Decision Making. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015.

42. Ma, C.; Yang, X.; Chongyang, Z.; Yang, M.-H. Long-term correlation tracking. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.

43. Kim, H.-U.; Lee, D.-Y.; Sim, J.-Y.; Kim, C.-S. IEEE: SOWP: Spatially Ordered and Weighted Patch Descriptor for Visual Tracking. In Proceedings of the 2015 IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3011–3019.

44. Wu, Y.; Lim, J.; Yang, M.-H. Object Tracking Benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.* 2015, 37, 1834–1848. [CrossRef] [PubMed]