Robust Correlation Tracking for UAV Videos via Feature Fusion and Saliency Proposals

Xue, Xizhe; Li, Ying; Dong, Hao; Shen, Qiang

Published in:
Remote Sensing

DOI:
10.3390/rs10101644

Publication date:
2018

Citation for published version (APA):
Xue, X., Li, Y., Dong, H., & Shen, Q. (2018). Robust Correlation Tracking for UAV Videos via Feature Fusion and Saliency Proposals. *Remote Sensing*, *10*(10), Article 1644. https://doi.org/10.3390/rs10101644

remote sensing

MDPI

Article

Robust Correlation Tracking for UAV Videos via Feature Fusion and Saliency Proposals

Xizhe Xue [1], Ying Li [1,*], Hao Dong [1] and Qiang Shen [2]

[1] School of Computer Science and Engineering, Shaanxi Provincial Key Laboratory of Speech and Image Information Processing, Northwestern Polytechnical University, Xi'an 710129, China; xuexizhe@mail.nwpu.edu.cn (X.X.); 18729510482@163.com (H.D.)

[2] Department of Computer Science, Faculty of Business and Physical Sciences, Aberystwyth University, Aberystwyth, SY23 3DB, UK; qqs@aber.ac.uk

* Correspondence: lybyp@nwpu.edu.cn; Tel.: +86-029-8843-1532

Received: 10 September 2018; Accepted: 12 October 2018; Published: 16 October 2018

Abstract: Following the growing availability of low-cost, commercially available unmanned aerial vehicles (UAVs), more and more research efforts have been focusing on object tracking using videos recorded from UAVs. However, tracking from UAV videos poses many challenges due to platform motion, including background clutter, occlusion, and illumination variation. This paper tackles these challenges by proposing a correlation filter-based tracker with feature fusion and saliency proposals. First, we integrate multiple feature types such as dimensionality-reduced color name (CN) and histograms of oriented gradient (HOG) features to improve the performance of correlation filters for UAV videos. Yet, a fused feature acting as a multivector descriptor cannot be directly used in prior correlation filters. Therefore, a fused feature correlation filter is proposed that can directly convolve with a multivector descriptor, in order to obtain a single-channel response that indicates the location of an object. Furthermore, we introduce saliency proposals as re-detector to reduce background interference caused by occlusion or any distracter. Finally, an adaptive template-update strategy according to saliency information is utilized to alleviate possible model drifts. Systematic comparative evaluations performed on two popular UAV datasets show the effectiveness of the proposed approach.

Keywords: UAV video; visual tracking; correlation filter; saliency detection; feature fusion

1. Introduction

Recent years have witnessed significant developments in computer vision. An enormous amount of research effort has gone into vision-based tasks, such as object tracking [1–6] and saliency detection [7–10]. As a core field of computer vision, visual tracking [4–6,11] plays an active role in a wide range of applications, including driverless vehicles, robotics, traffic analysis, medical imaging, motion analysis, and many others.

It is critical to employ an efficient feature representation in order to improve the performance in object tracking. Gradient and color features are the most popular single types of feature. In particular, color features, such as color names (CN), help capture rich color characteristics, and histogram of oriented gradient (HOG) [12] features are adept in capturing abundant gradient information. Based on these feature descriptions, a variety of techniques on target tracking have been proposed. For instance, FragTrack [13] is devised to build object appearance models by exploiting multiple parts of the target. Babenko et al. [14] presented a multiple instance learning (MIL) algorithm to develop a discriminative model by bagging all ambiguous negative and positive samples. Grabner et al. [15] utilized a novel on-line Adaboost feature selection method (OAB), benefitting considerably by on-line training. In a

past paper [2], a structural local sparse representation is applied to tracking task, where both partial and spatial information are exploited. Zhang et al. [16] discovered the relationship between an object and its spatiotemporal context based on the use of a Bayesian framework. Extended Lucas Kanade (ELK) method [17] considers two log-likelihood terms that are related to information regarding object pixels or background affiliation, in addition to the standard LK template matching term. Most of the aforementioned techniques are dependent of the intensity or texture information while characterizing a given image. However, it is difficult for them to meet the requirement of processing a large number of frames per second without resorting to parallel computation on a standard PC in dealing with real-time tasks [17]. From this viewpoint, correlation filters [18–22] show their strengths both in speed and in accuracy, where tracking problem is converted from time domain to frequency domain with fast Fourier transform (FFT). In so doing, convolution can be substituted with multiplication in an effort to achieve fast learning and target detection.

Although high tracking speed may be obtained, long-time tracking can often result in model drift. To ensure the stability of model updating in object tracking, Kalal et al. [1] decomposed the ultimate task of tracking into subtasks of tracking, learning and detection (TLD), where tracking and detection reinforce each other. However, if the location of an object is predicted only with respect to the previous frame, the appearance model may suffer from noisy samples. In particular, when the object is becoming blocked by something else, the tracker will fail immediately. Having taken notice of this, Hare et al. [2] adjusted the appearance model in a more reliable way, learning a joint structured output (Struck) to predict the object location. Apart from using a correlation filter, Zhu et al. [21] introduced an additional filter for detection, which greatly alleviated the problems of location error and model drifting caused by serious occlusion. Benefiting from temporal context and online redetector, a method described previously [22] performs robustly to appearance variation.

Note that following the increasing availability of low-cost, commercially available unmanned aerial vehicles (UAVs), more and more research efforts have been focusing on object detection and tracking by UAV videos. For example, Logoglu et al. [23] designed a feature-based moving object detection method for aerial videos. Fu et al. [24] proposed a technique named ORVT, for onboard robust visual tracking of targets in aerial images using a reliable global-local object model. However, all methods mentioned above cannot cope well with challenges appearing in such videos, which typically involve illumination variation, background clutter, and occlusion. To address these issues we propose a robust tracking approach for UAV videos, which offers three main contributions: (1) Composed of the HOG and dimension-reduced CN features, fused features are introduced to correlation filter in order to improve the robustness of appearance model in describing the target. (2) To deal with background clutter and meanwhile, and to reduce the risk of model drifts caused by occlusion, saliency proposals are introduced as posterior information to relocate the object. (3) A new adaptive template update method is proposed to further alleviate the problem of model drift that is caused by occlusion or distraction. The effectiveness of this approach is demonstrated through systematic comparisons against other techniques.

The rest of this paper is organized as follows. Section 2 discusses relevant previous work on correlation filter and saliency detection. Under the general framework of correlation filter, Section 3 describes our approach. Section 4 presents an evaluation of the proposed approach and a comparative study with state-of-the-art techniques. Section 5 discusses the tracking speed of different methods and assesses the effects of each contribution made by the proposed work. Finally, Section 6 concludes this study and points out interesting further research.

2. Related Work

2.1. Correlation Filters

Because of their impressive high-speed, correlation filters have attracted a great deal of interests in object tracking. For instance, David S. Bolme et al. [25] proposed the minimum output sum of squared

errors (MOSSE) filter, which works by finding the maximum cross correlation response between the model and candidate patch. Henriques et al. [26] exploited the circulate structure and Fourier transformation in a kernel space (CSK), offering excellent performance on a range of computer vision problems. A vector correlation filter (VCF) was proposed by Boddeti et al. [27] to minimize localization errors while improving the tracking speed. Danelljan et al. [28] exploited the color attributes of an object and introduced CN features into CSK to perform object tracking. Combining techniques of kernel trick and cycle shift [26], Kernelized Correlation Filter (KCF) [29] entails more adaptive performance for diverse scenarios using multichannel HOG features. The DSST tracker [19] learns adaptive multiscale correlation filters by the use of HOG features to handle the scale change of target objects. To learn a model that is inherently robust to both color changes and deformations, Staple [30] combines two image patch representations that are sensitive to competing factors. Danelljan et al. [31] utilized a spatial regularization component in the learning process to penalize correlation filter coefficients as a function of their spatial location. Recently the authors of a past paper [20] proposed a background-aware correlation filter (BACF) that can model how background as well as foreground of an object may vary over time. To drastically reduce the number of parameter in the model, Danelljan et al. [32] proposed a factorized convolution operator. The utilization of a compact generative model of the training sample distribution significantly reduces the memory and time complexity, while providing better diversity of samples.

Whilst many methods exist as outlined above, they do not address the critical issue of online model update. As a result, such correlation trackers are susceptible to model drifting and hence, are less effective for handling important problems such as long-term occlusion and object out-of-view.

2.2. Saliency Detection

Saliency is considered to represent an object or a pixel that is more conspicuous than its neighbors. Saliency detection aims to capture the regions that stand out in an image. In terms of algorithm strategy, saliency detection approaches can be categorized into two subgroups, one is the group of bottom-up data-driven methods [9,33,34] and the other is that of top-down task-driven methods [10].

Top-down methods are task-driven which learn a supervised classifier for salient object detection. In DRFI [9], hand crafted features were extracted to classify each region. Xi at al. [10] proposed a SVM based methods with a color information as the input. On the other hand, for most bottom-up methods, low-level features are employed to calculate the saliency value. By analyzing the log-spectrum of an input image, Hou X. et al. [8] introduced a mechanism to extract the spectral residual of an image in spectral domain. They proposed a fast method for constructing the corresponding saliency map in the spatial domain which is independent of features, categories, or other forms of prior knowledge of the domain objects. To keep the structure of the objects, region-based methods were also proposed. These methods segmented images into coherent regions to obtain proper spatial structure. Goferman et al. [33] used a patch-based approach to get global properties. Cheng et al. [34] combined a soft abstraction to decompose an image into large perceptually homogeneous elements in order to achieve efficient saliency detection. Additionally, boundary cue is used to improve the saliency detection performance, with boundary prior knowledge treating image boundary regions as labeled background.

3. Proposed Approach

We aim to develop an online tracking algorithm that is adaptive to significant appearance change without being prone to drifting, in which the extracted fused features are encoded in terms of multivectors. Further, saliency information is attained to provide reliable proposals for correlation filters to redetect objects in case of tracking failure. In particular, the adaptive template updating rules are put forward in order to achieve robust performance. The flowchart of the proposed tracking approach is illustrated in Figure 1, where the speed of such a tracker is ensured using a correlation filter.

3.1. Correlation Tracking through Fused Features

Features play an important role in computer vision. For example, much of the impressive progress in object detection can be attributed to the improvement in the representation power of features [35]. Gradient and color features are the most widely exploited in object detection and tracking. Indeed, previous work [36] has verified that there exists a strong complementarity between gradient and color features.

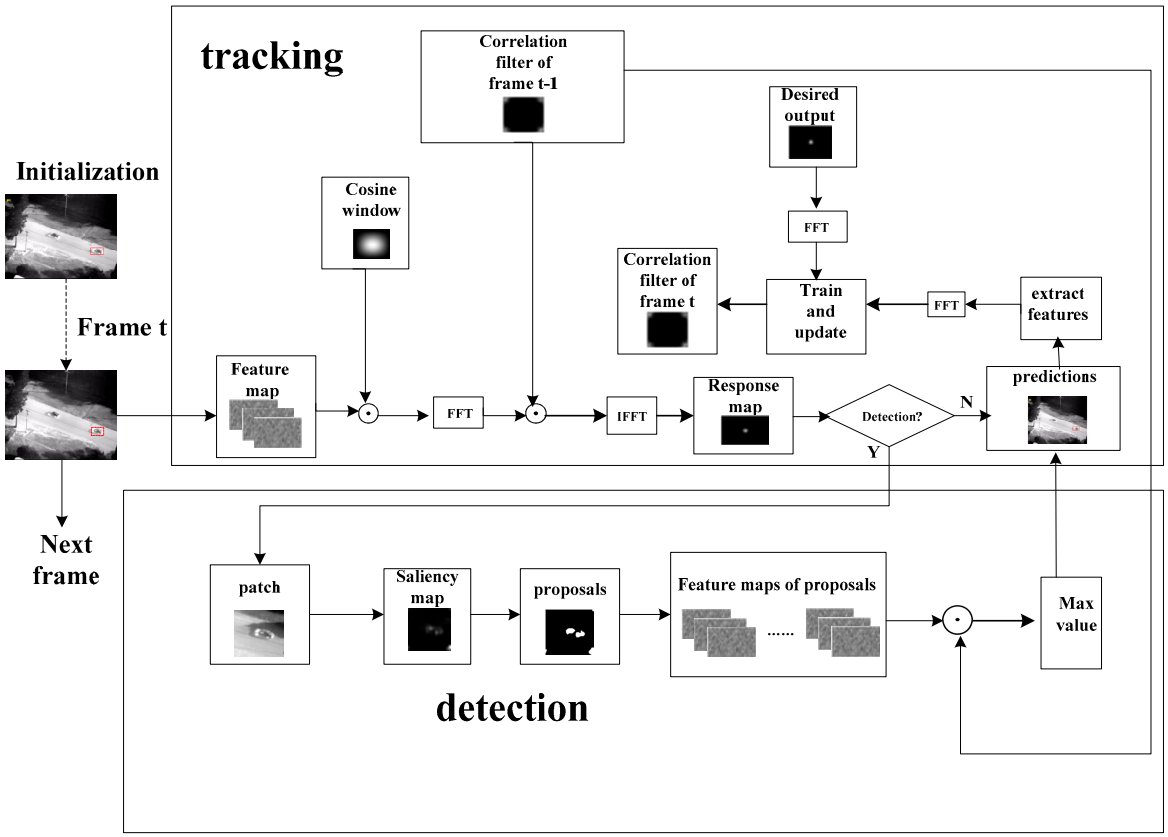


Figure 1. Flowchart of proposed tracking algorithm.

However, how to jointly utilize different features for aerial tracking is still an open question. Compared with generic visual object tracking, certain tracking challenges are amplified in aerial scenarios, including abrupt camera motion, low resolution, significant changes in scale and aspect ratio, fast moving objects, as well as partial or full occlusion. It is difficult to obtain comprehensive information of interesting objects using a single feature type like HOG or CN [37] under such circumstances. Hence, we employ fused features to achieve robust performance in aerial tracking. Inspired by CN from a linguistic viewpoint [37], which involves eleven preliminary color terms: black, blue, brown, grey, green, orange, pink, purple, red, white, and yellow, we concatenate CN features extracted from the original image and substantially reduce the number of color dimensions in an effort to enable a significant speed boost, with the support a work reported previously [28]. In addition, any given input color image is transformed into one with grey values and then, HOG features are extracted from the resulting grey image. All these features are concatenated directly to form a multivector as a fused feature descriptor.

In this paper, we utilize the multivector representation of fused features which better fits with the correlation tracking framework. More specifically, we denote x_d as the fused feature multivector of cardinality $d \in R^D$, respectively. We consider y_d as the desired correlation output corresponding to a

given sample x_d. A correlation filter w of the same dimensionality as x_d is then learned by solving the following minimization problem:

$$w* = \arg\max \sum \|w \cdot x_d - y_d\|^2 + \lambda \|w\|_2^2 \tag{1}$$

where λ is a regularization parameter. Note that the minimization problem in Equation (1) is akin to training the multivector correlation filters in a past paper [27], and can be resolved within each individual feature channel using FFT. Let the capital letters be the corresponding Fourier transformed signals. The learned filter in the frequency domain on the $d-th$ $(d \in \{1, \ldots, D\})$ channel can be written as

$$W_d = \frac{\overline{Y} \odot X^d}{\sum_{i=1}^{D} \overline{X^i} \odot X^i + \lambda} \tag{2}$$

where Y, X, W denote the discrete Fourier transforms (DFT) of y, x w, respectively; $\overline{Y}$ represents the complex conjugation of Y, and $\overline{Y} \odot X^d$ is a point-wise product. Given an image patch in the next frame (of the video sequence concerned), the fused feature multivector is denoted by $z \in R^D$. The correlation response map is computed by

$$r = F^{-1}\left(\sum_{d=1}^{D} W^d \odot \overline{Z}^d\right) \tag{3}$$

where the operator F^{-1} denotes the inverse FFT. The target location can then be estimated by searching for the position of the maximum value of the correlation response map r, such that

$$(x', y') = \arg\max_{a,b}(r(a, b)) \tag{4}$$

3.2. Object Redetection Based on Saliency Proposals

For traditional correlation filter-based trackers [26–29], the use of FFT helps greatly reduce the computational cost, demonstrating the ability of real time tracking on UAV videos. Nevertheless, two main challenges remain: (a) distraction and (b) model drift, caused by occlusion or background clutter. In DSST [19], an independent scale prediction filter is presented, but it fails perform well when serious occlusion exists, as shown in Figure 2. A common approach to handling model drift is to integrate a short-term tracker and online long-term detector, as with what is taken in the TLD algorithm [1]. However, learning an online long-term detector relies heavily on lots of well-labeled training samples which can be difficult to collect. Additionally, an exhaustive search through the entire image with sliding windows is time-consuming, especially for the case of employing complex but discriminative features.

(**a**) (**b**) (**c**)

Figure 2. Tracking results of DSST: (**a**) tracking well without occlusion; (**b**) tracking failed within occlusion; and (**c**) model drift after occlusion.

To provide relatively less proposals and suppress the background interference, in this paper we not only utilize an adaptive update strategy to learn the appearance model, but also exploit a

few pieces of reliable information from the biologically inspired saliency map. We postulate that the redetector could alleviate the model drift problem caused by occlusion or distraction.

3.2.1. Saliency Proposal Detection

Due to its simplicity and efficiency, we propose to utilize the spectral residual based saliency detection algorithm [8] to obtain saliency proposals. Then we iteratively redetect the object based on the resulting saliency proposals. Given an original image I, Fourier transform is used to extract the phase features $P(f)$ and amplitude features $A(f)$ of the image (in the frequency domain), as shown in Equations (5) and (6):

$$A(f) = R(F(I(x))) \tag{5}$$

$$P(f) = S(F(I(x))) \tag{6}$$

From this averaged spectrum is approximated by convoluting the input image $h_n(f) * L(f)$, where $L(f) = \log(A(f))$ and $h_n(f)$ denotes a local average filter to approximate the shape of $A(f)$. Thus, the spectral residual $R(f)$ can be obtained by Equation (7):

$$R(f) = L(f) - h_n(f) * L(f) \tag{7}$$

In the subsequent experimental studies, the size of $h_n(f)$, n is empirically set to 3.

The spectral residual $R(f)$ helps capture the key information contained within an image. In particular, it serves as a compressed representation of the underlying scene reflected by the image. Using inverse Fourier transform (IFT), we can construct the saliency map in the spatial domain. The saliency map contains primarily the nontrivial parts of the scene. The content of the residual spectrum can also be interpreted as the unexpected portion of the image. Thus, the value at each point in a saliency map is squared to indicate the estimation error. For better visual effects, we smooth the saliency map with a Gaussian filter $g(x)$. In sum, given an image $I(x)$, we have

$$\begin{aligned} S(x) &= g(x) * F^{-1}[\exp(R(f) + P(f))]^2 \\ g(x) &= \tfrac{1}{2\pi\sigma^2} \exp(-\tfrac{(i-k-1)^2+(j-k-1)^2}{2\sigma^2}) \end{aligned} \tag{8}$$

where $k = 4, \sigma = 2.5$, (i, j) is the coordinate of pixel x and F^{-1} denotes IFT.

Having built a saliency map $S(x)$, saliency proposals can be obtained using threshold segmentation and region connection. Specifically, the saliency map is first segmented according to the adaptive thresholding [38], and therefore generates a number of interconnected domains. Without losing generality, suppose that the connected domain corresponding to the real object does not appear at the border of the image, we can exclude the connected domains whose centers are within a certain number of pixels of the boundary in the segmented image to derive the final saliency proposals (in implementation herein, this number is set to 15).

3.2.2. Redetection Based on Saliency Proposals

The traditional correlation tracker cannot perform well when serious occlusion exists. To address this issue, we propose our tracker with a redetection approach based on saliency proposals. If the correlation response r is less than the threshold T_1 for more than L consecutive frame, it is high likely that the target is occluded seriously. So we redetect the object using saliency information at this time, otherwise the object will be located only by correlation filter.

Specifically, we consider the location of a certain object in the previous frame as a center point, around which the image patch is cropped from the original image. The image patch is of size $B \times B$, $B = \lfloor 0.08 \times num\_lost \times \sqrt{w \times h} + w + 1 \rfloor$, where num_lost is the number of frames where serious occlusion happened continuously, w and h denote the initialized horizontal width and vertical height of the interested object in the first frame, respectively, and $\lfloor \ \rfloor$ means rounding down. Such an image patch

is designed to guarantee that the longer the object is lost, the bigger the image patch is cropped from the original image.

From this we can obtain the saliency proposals in the image patch and sample paddings with a size of $3w \times 3h$ around the center of every saliency proposal. Then, correlation filtering is applied between the center of each saliency proposal and the template in the first frame, with the point of the largest response r_m taken as the center of the new object if r_m exceeds a certain value T_2. Otherwise, in order to ensure that the object remains within an image patch, the patch is expanded when repeating the redetection step in the next frame.

Figure 3 shows that an object is redetected based on saliency proposals. As can be seen from this figure, when the object is lost, we can gradually relocate its approximate position by saliency detection in the area where the object may appear. Following this, the object can then be relocated accurately using correlation filtering.

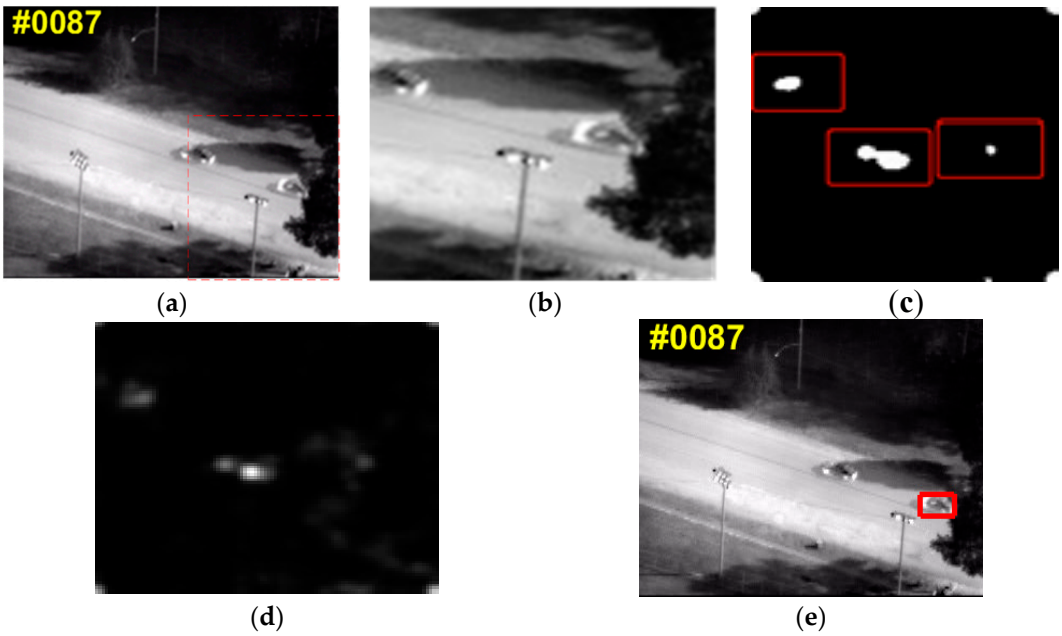


Figure 3. Object redetection based on saliency proposals. (**a**) Original image; (**b**) zoomed-in image patch cropped from (**a**); (**c**) zoomed-in saliency map; (**d**) zoomed-in saliency proposals; and (**e**) redetected object (marked with a red rectangle).

3.3. Adaptive Model Updating

To obtain a robust and efficient approximation, we update the numerator A^d and the denominator B^d of the correlation filter W^d in Equation (2) separately, using a moving average:

$$A_t^d = (1-\eta)A_{t-1}^d + \eta Y \odot \overline{X}_t^d \tag{9}$$

$$B_t^d = (1-\eta)B_{t-1}^d + \eta \sum_{i=1}^{D} X_t^i \odot \overline{X}_t^i \tag{10}$$

$$W_t^d = \frac{A_t^d}{B_t^d + \lambda} \tag{11}$$

where t is the frame index and the learning rate η is set to 0.025 empirically.

If the object position is relocated according to saliency information, we update the template according to Equation (12):

$$\begin{aligned} A &= (1 - 10 \times \eta) \times A^l + 10 \times \eta \times init\_A^l \\ B &= (1 - 10 \times \eta) \times B + 10 \times \eta \times init\_B \end{aligned} \tag{12}$$

Then, the previous templates and the first frame template are combined to update the target template, thereby minimizing potential model drift.

4. Experimental Results

We provide representative experimental results in this section. The proposed tracker is implemented in Matlab2014 on a PC with a 3.4 GHz processor and 16 GB RAM without involving any sophisticated program optimization. In order to present an objective evaluation regarding the performance of the proposed approach, we conduct experiments on two datasets, namely, the VIVID dataset [39] and the UAV123 dataset [40], for both qualitative and quantitative evaluations. In these experiments, the parameters are fixed for all of the sequences, in which T_1 and T_2 are set to 0.2 and 0.25, respectively. In addition, L is set as 7 and the candidate region size for the correlation filter is set to three times as big as that of the object under tracking.

We compare the proposed tracker with a range of excellent state-of-art trackers, including TLD [1], DSST [19], BACF [20], ORVT [24], Staple [30], SRDCF [31], ECO_HC [32], KCFDP [41], BIT [42], and fDSST [43]. Among these trackers, TLD introduces the detection method into the tracking problem, which performs well when occlusion exists, while DSST, KCFDP, SRDCF, Staple, BACF, ECO_HC, and fDSST involve the use of correlation filters to improve the speed of tracking. In particular, ORVT is an onboard robust aerial tracking algorithm working by the use of a reliable global-local object model. Additionally, BIT is a biologically inspired tracker that extracts low-level biologically inspired features while imitating an advanced learning mechanism to combine generative and discriminative models for target location. Note that we employ publicly available codes of compared trackers for fair comparison.

We follow the standard evaluation metrics for object tracking algorithms in two aspects: the precision rate and success rate. The precision rate shows the percentage of successfully tracked frames on which the center location error (CLE) of a tracker is within a given threshold (e.g., 20 pixels), with CLE defined as the average Euclidean distance between the center locations of the targets and the manually labeled ground truths. A tracking result in a frame is considered successful if $\frac{|r_d \cap r_t|}{|r_d \cup r_t|} > \theta$ for a threshold $\theta \in (0, 1]$, where r_d and r_t denote the areas of the bounding boxes of the tracked region and the ground truth, respectively, $\cap$ and $\cup$ represent the intersection and union of two regions, respectively, and $|\cdot|$ denotes the number of pixels in the region. Thus, the success rate is defined as the percentage of frames where the overlap rates are greater than a threshold θ. Normally, the threshold θ is set to 0.5.

We present the results under one-pass evaluation (OPE) using the average precision and success rate over all sequences. OPE is the most common evaluation method which runs trackers on each sequence for once. It initializes the trackers with the state of the ground truth object in the first frame and reports the average precision or success rates across all the results obtained.

4.1. Experiments on VIVID Dataset

There are eleven video sequences in the VIVID dataset. Apart from motion blur and fast motion, these video sequences also suffer from further difficulties such as occlusion, scale variation, background clutter, low resolution, etc. In the VIVID dataset, the ground truth is given every ten frames. To evaluate the trackers more accurately, we mark the entire eleven sets of videos' ground truths, referring to the official data, for quantitative evaluation.

The experimental results on these nine videos are summarized in Tables 1 and 2, which show the overall rates of the success plots and those of the precision plots, respectively. As can be seen from these tables, our tracker performs reliably and can achieve optimal outcomes overall. In particular, regarding the first three video sequences where occlusion occurs seriously, our method exhibits an excellent performance benefitting from saliency based redetection and adaptive template updating, while the other trackers lost the targets under these circumstances. However, the remaining video sequences are frequently affected by scale change, rotation and similar objects which led to a decline in the performance of our algorithm also.

Table 1. Overall rates of precision plots on different sequences of VIVID dataset.

	Ours	**DSST**	**SRDCF**	**KCFDP**	**BACF**	**Staple**	**BIT**	**fDSST**	**TLD**	**ECO_HC**	**ORVT**
pktest01	**0.870**	0.477	0.600	0.606	0.459	0.353	0.356	0.359	0.850	0.411	0.390
pktest02	**0.814**	0.329	0.566	0.448	0.745	0.475	0.329	0.315	0.712	0.556	0.544
pktest03	**0.867**	0.576	0.605	0.615	0.786	0.661	0.563	0.570	0.617	0.554	0.511
egtest01	0.845	0.274	0.275	0.274	**0.910**	0.909	0.861	0.836	0.433	0.887	0.849
egtest02	0.786	0.898	0.931	0.873	0.927	0.928	0.907	**0.941**	0.911	0.691	0.632
egtest03	0.825	0.864	0.855	0.863	0.845	0.859	0.870	0.867	**0.919**	0.849	0.864
egtest04	0.803	0.391	0.940	0.918	0.927	0.928	0.907	0.941	0.216	**0.953**	0.384
egtest05	0.712	0.731	0.734	0.730	0.731	0.734	0.729	0.736	**0.786**	0.729	0.726
Redteam	0.869	0.953	0.968	0.936	**0.964**	0.962	0.911	0.943	0.931	0.944	0.946
Overall	**0.821**	0.610	0.719	0.696	0.810	0.757	0.715	0.723	0.708	0.730	0.650

Table 2. Overall rates of success plots on different sequences of VIVID dataset.

	Ours	**DSST**	**SRDCF**	**KCFDP**	**BACF**	**Staple**	**BIT**	**fDSST**	**TLD**	**ECO_HC**	**ORVT**
pktest01	**0.533**	0.188	0.173	0.165	0.175	0.170	0.147	0.172	0.493	0.173	0.142
pktest02	**0.510**	0.101	0.285	0.114	0.286	0.098	0.090	0.097	0.420	0.284	0.141
pktest03	**0.462**	0.267	0.239	0.231	0.246	0.277	0.216	0.274	0.194	0.278	0.199
egtest01	0.559	0.086	0.083	0.080	0.553	**0.584**	0.466	0.446	0.178	0.583	0.466
egtest02	0.497	0.655	0.680	0.604	0.640	0.691	0.606	0.697	0.655	0.750	**0.875**
egtest03	0.538	0.643	0.531	0.618	0.551	0.643	0.653	0.646	**0.709**	0.601	0.646
egtest04	0.283	0.244	0.575	0.511	0.640	0.691	0.606	**0.697**	0.117	0.508	0.232
egtest05	0.359	0.397	**0.404**	0.395	0.397	0.392	0.394	0.402	0.398	0.126	0.391
redteam	0.542	0.580	**0.859**	0.738	0.781	0.597	0.561	0.672	0.716	0.721	0.626
Overall	**0.476**	0.351	0.425	0.384	0.474	0.460	0.415	0.456	0.431	0.447	0.413

Figure 4 shows the qualitative evaluation on the VIVID dataset. Figure 4a illustrates the performance of our approach and compared algorithms on the sequence *pktest01*. Only our method keeps the virtue of robust tracking after more than 100 frames of occlusion. It is evident that through redetecting object by saliency information, the proposed tracker is more robust than the other trackers. In the sequence *pktest03*, in addition to motion blur and fast motion, the other main challenges for tracking are illustration variation, serious occlusion, and background clutter. From the last picture of Figure 4b, it is obvious that the full occlusion with the car is handled well by our tracker, while the other methods have a shift for the target. This implies that saliency detection makes an important contribution to achieve such an outstanding performance. In addition, almost every frame is subject to a varying degree of background clutter. Note that the scale of the target is too small to recognize, it is almost integrated with the background with certain texture and other details lost. It can be seen from the results that only our algorithm can successfully deal with the problem of background clutter as other methods fail to track the target completely. There is no doubt that fused features help improve the robustness of the proposed appearance model. In addition, the adaptive model update strategy also helps reduce model drift. Both of the above measures lead to the excellent performance of our method.

As shown in Figure 4c–e, where there is no significant occlusion, our methods can always follow the target as with other trackers. It works even when similar cars appear in the sequence *egtest02*.

However, when scale variation and rotation occur, the calculated scales of bounding boxes are not sufficiently accurate causing a decrease in the accuracy of our tracker. For the sequences *egtest01* and *redteam*, the background is similar with the edge of the target. If the response of the correlation filter is less than the threshold for a long time, our tracker will automatically try to relocate the target by exploiting the vision saliency. Of course, this strategy may gradually introduce certain noise from the background around the target to the template, leading to slight model drift.

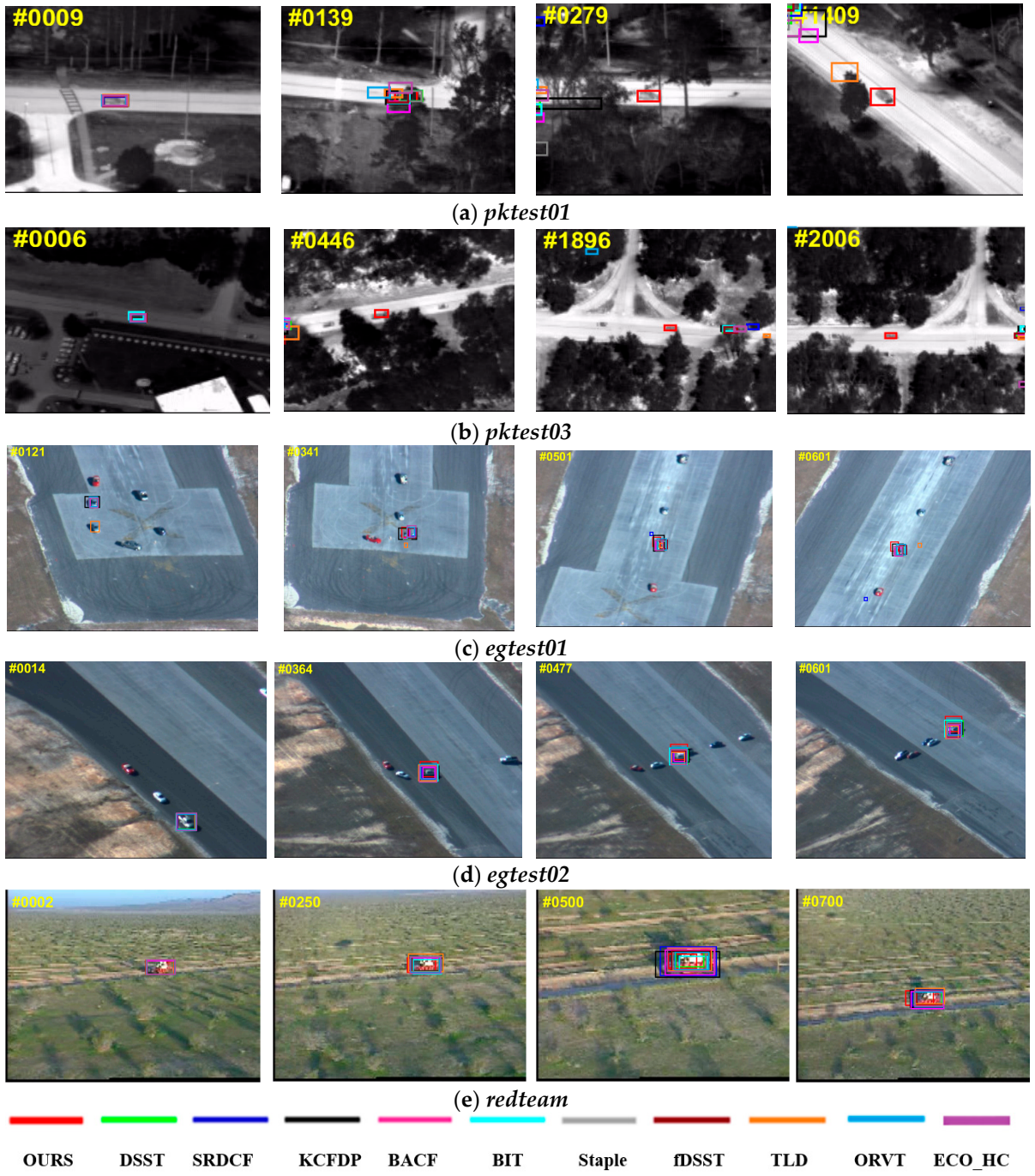


Figure 4. Qualitative evaluation of tracking results on VIVID dataset.

4.2. Experiments on UAV123 Dataset

In order to evaluate the performance of our proposed approach, we conduct experiments on twenty challenging video sequences selected from the UAV123 dataset for both quantitative and

qualitative analysis. The UAV123 dataset provides a facility for the evaluation of different trackers on a number of fully annotated HD videos captured from a professional grade UAV. It complements those benchmarks establishing the aerial component of tracking while providing a more comprehensive sampling of tracking nuisances that are ubiquitous in low-altitude UAV videos. Apart from aspect ratio change (ARC) and fast motion (FM), these video sequences are also affected by several adverse conditions such as background clutter (BC), camera motion (CM), full occlusion (FOC), illumination variation (IV), low resolution (LR), out of view (OV), partial occlusion (POC), similar object (SOB), scale variation (SV), and viewpoint change (VC). Thus, the experiments carried out herein include all typical challenges involved in real-world aerial tracking problems.

Ranging from 535 to 1783 frames, the twenty selected sequences used here involve all the challenging factors in the UAV123 dataset with different resolutions. Various scenes exist in these sequences, such as roads, buildings, field, beaches, and so on. The targets include aerial vehicles, person, trucks, boats, cars, etc. Detailed information of these sequences is listed in Table 3.

Table 3. Description of sequences selected from UVA123 for experimental investigations.

Sequence	Size	# Frames	Challenge
truck3	1280 × 720	535	LR,POC,BC
bike2	1280 × 720	553	ARC,BC,CM,FOC,IV,OV,POC
boat6	1280 × 720	805	SV
wakeboard3	1280 × 720	823	SV,ARC,LR,VC,CM
building3	1280 × 720	829	SV.SOB
group2_2	1280 × 720	865	SV, FOC,POC,VC,CM,SOB
person14_1	1280 × 720	847	SV,ARC,LR,FOC,POC,BC,CM
boat1	1280 × 720	901	SV
group2_3	1280 × 720	913	SV,ARC,LR, FOC,POC, BC,IV,CM,SOB
uav1_3	720 × 480	997	SV,ARC,LR,FM, FOC,POC,OV, BC,IV,VC,CM
person10	1280 × 720	1021	SV,FOC,POC,OV,VC,CM,SOB
car17	1280 × 720	1057	SV,ARC,LR,VC,CM
person16	1280 × 720	1147	SV,ARC,FOC,POC,BC,IV,CM
car14	1280 × 720	1327	SV,ARC,LR, FOC,POC,OV,VC,CM
group1_1	1280 × 720	1333	SV,POC,SOB
person18	1280 × 720	1393	SV,ARC, POC,OV,VC,CM
car10	1280 × 720	1405	SV,POC,SOB
person2_2	1280 × 720	1434	POC,OV,CM
car3	1280 × 720	1717	SV,LR,POC,OV,CM,SOB
person20	1280 × 720	1783	SV, ARC,POC,OV, VC, CM,SOB

Tables 4 and 5 exhibit the overall rates of the success plots and those of the precision plots on the twenty sequences, respectively. It can be seen that our tracker achieves the best performance on average, demonstrating its robustness in dealing with object tracking tasks involving different challenging factors and various background types.

Table 4. Overall rates of precision plots on different sequences of UAV123 dataset.

	Ours	DSST	SRDCF	KCFDP	BACF	Staple	BIT	fDSST	TLD	ECO_HC	ORVT
truck3	0.967	**0.972**	0.969	0.931	0.967	0.969	0.907	0.961	0.915	0.970	0.949
bike2	0.275	0.354	**0.360**	0.163	0.274	0.263	0.144	0.270	0.144	0.358	0.336
boat6	**0.951**	0.825	0.837	0.838	0.950	0.923	0.571	0.818	0.768	0.905	0.823
building3	0.956	0.937	0.938	0.919	0.954	0.948	0.898	0.923	0.808	**0.961**	0.899
group2_2	**0.936**	0.902	0.914	0.635	0.931	0.593	0.895	0.563	0.357	0.923	0.887
person14_1	0.204	0.209	0.208	0.207	0.208	0.207	0.207	0.207	**0.823**	0.208	0.209
boat1	**0.816**	0.687	0.722	0.603	0.815	0.784	0.779	0.777	0.329	0.780	0.526
wakeboard3	0.928	0.257	0.860	0.259	0.928	**0.935**	0.253	0.265	0.395	0.932	0.875
group2_3	**0.900**	0.682	0.843	0.757	0.898	0.875	0.759	0759	0.274	0.868	0.775
uav1_3	**0.389**	0.090	0.155	0.090	0.198	0.091	0.090	0.090	0.195	0.341	0.090
person10	0.339	0.335	0.341	0.336	0.329	0.342	0.338	0.312	**0.541**	0.340	0.339
car17	0.329	0.217	0.104	0.145	0.329	0.230	0.144	0.272	0.315	**0.507**	0.247

Table 4. *Cont.*

	Ours	DSST	SRDCF	KCFDP	BACF	Staple	BIT	fDSST	TLD	ECO_HC	ORVT
person16	**0.911**	0.215	0.215	0.219	0.216	0.215	0.214	0.214	0.202	0.215	0.205
car14	0.678	0.630	0.646	**0.701**	0.641	0.666	0.676	0.637	0.508	0.644	0.623
group1_1	0.847	0.778	0.902	0.870	0.887	0.630	0.833	0.396	0.183	**0.909**	0.908
person18	0.555	0.427	0.504	0.357	0.550	0.513	0.431	0.520	0.126	**0.563**	0.146
car10	0.942	0.912	0.943	0.947	**0.955**	0.946	0.939	0.916	0.633	0.950	0.952
person2_2	0.933	0.925	0.920	0.915	0.923	0.897	0.922	0.876	0.655	**0.937**	0.925
car3	0.963	0.958	0.956	0.944	0.953	0.952	0.666	0.930	0.090	**0.965**	0.926
person20	0.468	0.372	0.438	**0.508**	0.468	0.504	0.238	0.344	0.111	0.400	0.225
Overall	**0.698**	0.586	0.639	0.567	0.670	0.624	0.545	0.553	0.419	0.682	0.606

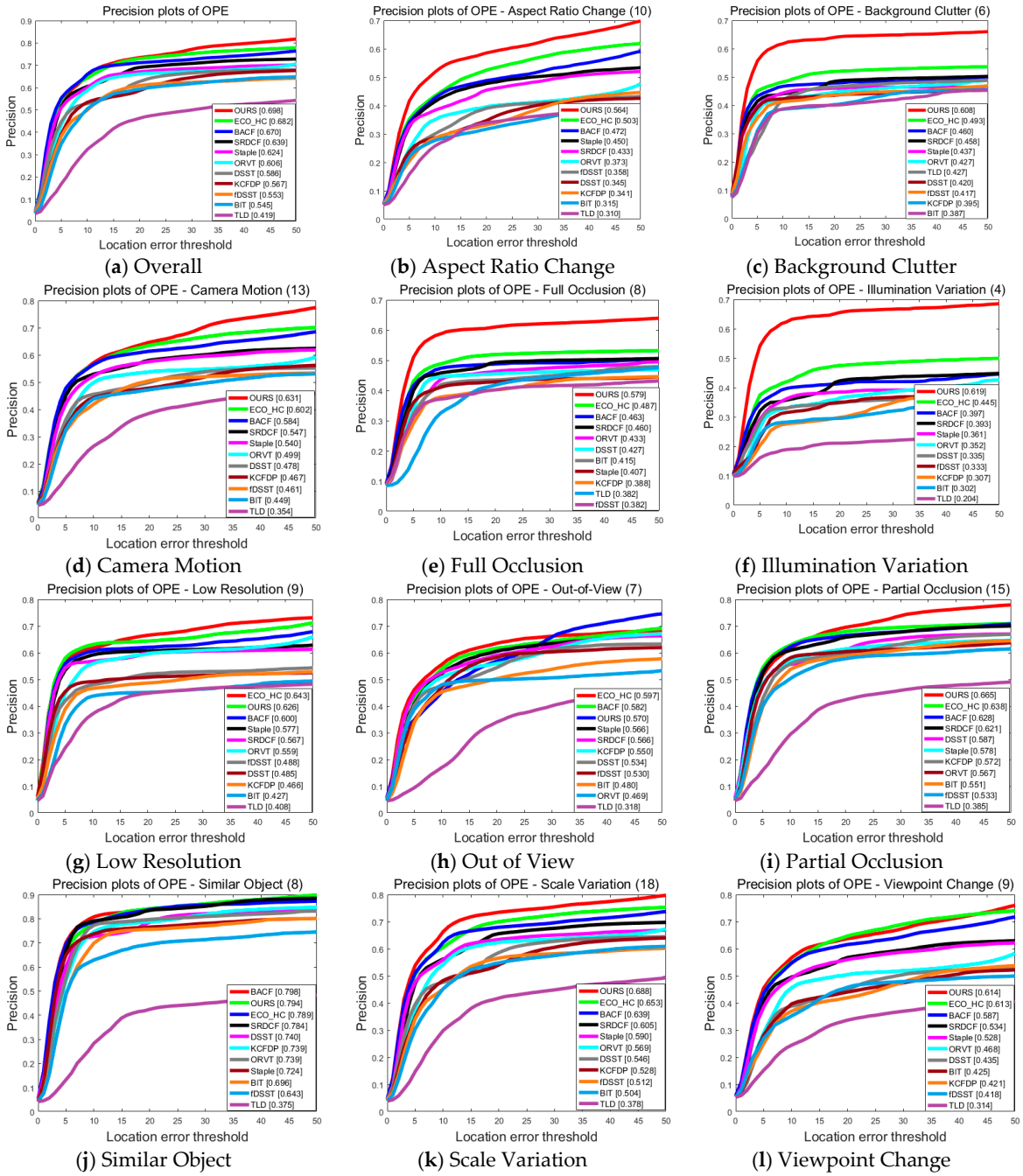
Table 5. Overall rates of success plots on different sequences of UAV123 dataset.

	Ours	DSST	SRDCF	KCFDP	BACF	Staple	BIT	fDSST	TLD	ECO_HC	ORVT
truck3	0.763	0.757	0.575	0.569	0.763	0.753	0.552	0.759	0.609	**0.809**	0.729
bike2	0.137	**0.162**	0.142	0.043	0.111	0.129	0.017	0.134	0.016	0.151	0.143
boat6	**0.810**	0.338	0.661	0.657	0.800	0.343	0.182	0.603	0.438	0.706	0.380
building3	**0.774**	0.544	0.669	0.615	**0.774**	0.646	0.517	0.638	0.424	0.745	0.538
group2_2	**0.689**	0.610	0.684	0.498	0.685	0.439	0.600	0.439	0.276	0.683	0.603
person14_1	0.118	0.137	0.131	0.130	0.136	0.134	0.132	0.136	**0.529**	0.135	0.137
boat1	**0.788**	0.376	0.765	0.460	0.786	0.689	0.376	0.761	0.494	0.767	0.768
wakeboard3	0.366	0.186	0.532	0.175	0.366	0.560	0.182	0.182	0.271	**0.622**	0.492
group2_3	**0.530**	0.388	0.467	0.376	0.528	0.504	0.400	0.309	0.104	0.487	0.402
uav1_3	**0.145**	0.001	0.047	0.001	0.071	0.001	0.001	0.001	0.069	0.147	0.001
person10	0.153	0.147	0.158	0.148	0.141	0.160	0.143	0.133	**0.295**	0.159	0.154
car17	**0.380**	0.090	0.072	0.054	**0.380**	0.098	0.055	0.104	0.268	0.130	0.215
person16	**0.626**	0.098	0.099	0.104	0.100	0.096	0.097	0.098	0.085	0.094	0.087
car14	0.433	0.368	0.384	**0.492**	0.433	0.439	0.375	0.453	0.314	0.443	0.418
group1_1	0.725	0.621	0.618	0.694	0.714	0.474	0.663	0.314	0.156	**0.770**	0.744
person18	0.659	0.505	0.615	0.601	0.659	**0.666**	0.507	0.619	0.185	0.660	0.352
car10	0.711	0.803	0.793	0.810	0.824	0.818	0.781	0.750	0.389	**0.832**	0.823
person2_2	0.764	0.761	0.675	0.764	0.766	0.729	0.763	0.719	0.432	0.776	**0.779**
car3	0.710	0.690	0.638	0.628	0.653	0.699	0.396	0.670	0.062	**0.738**	0.657
person20	**0.720**	0.333	0.645	0.685	0.718	0.693	0.337	0.582	0.212	0.653	0.331
Overall	**0.531**	0.396	0.470	0.425	0.522	0.454	0.345	0.420	0,282	0.526	0.426

We also perform an attribute-based comparison with other methods on this subset of the UAV123 dataset. Figures 5 and 6 show the success plots and precision plots of twelve respective attributes on the precision and success rates, respectively. As can be seen from these results, our tracker always performs reliably and can achieve the optimal, or at least a close to optimal solution in most cases. Specifically, for the amplified challenging factors in aerial tracking, including CM, BC, SV, ARC, FM, IV, FOC, and VC, our tracker is able to achieve promising results, benefitting from the robustness of fused features as well as from the employment of the appearance template and model updating strategy. For videos with fast moving objects, camera motion, and background clutter, the fused features have stronger abilities to capture the information from the objects and, therefore, lead to better results as compared to the classic single-feature trackers. In addition, when the aspect ratio of an object changes significantly, our adaptive appearance template updating strategy can adjust the template to the appearance of the object. Moreover, thanks to the high confidence model updating method background noise is suppressed as much as possible when serious occlusion exists in aerial videos. Nevertheless, our tracker may not perform equally well when dealing with images of low resolution and targets that are out of view. It is likely due to the fact that such challenging factors usually create very serious problems for saliency detection, resulting in model drift.

Figure 7 illustrates qualitative evaluations on the application of different trackers to example sequences selected from the UAV123 dataset. In the sequence *person16*, the background has the similar color with person, making it difficult for the trackers to successfully function to a different extent.

Owing to the use of saliency information, our tracker is able to relocate the object after it has been occluded by the tree and outperforms the state-of-the-art tracking methods. As shown in Figure 7b, the sequence *uav1_3* contains almost all the possible challenges in aerial tracking, especially low resolution and serious background clutter. Benefiting from the target redetection strategy, our tracker can track the target successfully all the time, while the others locate the target correctly only once in a while. Of course, the robustness of fused features also helps ensure the good performance of our tracker. However, for certain sequences with serious scale variation and similar objects, for example the sequence *car10*, our tracker slightly underperforms in comparison to several state-of-art algorithms (e.g., ECO_HC, BACF, and Staple). Under such circumstances, our tracker may incur small model drift but it does not lose the target.



(**a**) Overall (**b**) Aspect Ratio Change (**c**) Background Clutter

(**d**) Camera Motion (**e**) Full Occlusion (**f**) Illumination Variation

(**g**) Low Resolution (**h**) Out of View (**i**) Partial Occlusion

(**j**) Similar Object (**k**) Scale Variation (**l**) Viewpoint Change

Figure 5. Precision plots of proposed tracker compared with state-of-the-art approaches on different attributes of UAV123 dataset.

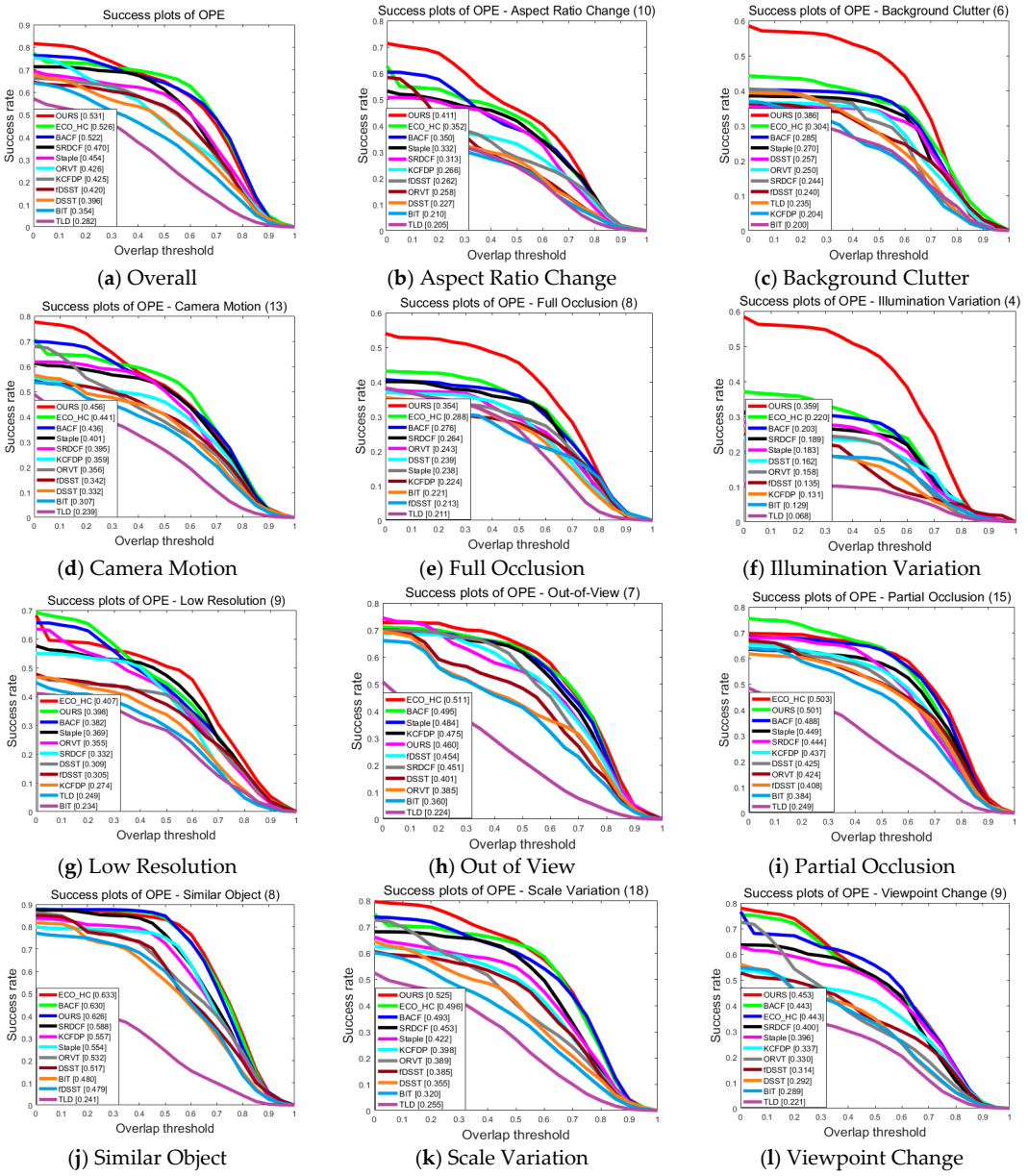


Figure 6. Success plots of tracker compared with state-of-the-art approaches on different attributes of proposed UAV123 dataset.

(**a**) *person16*

Figure 7. *Cont.*

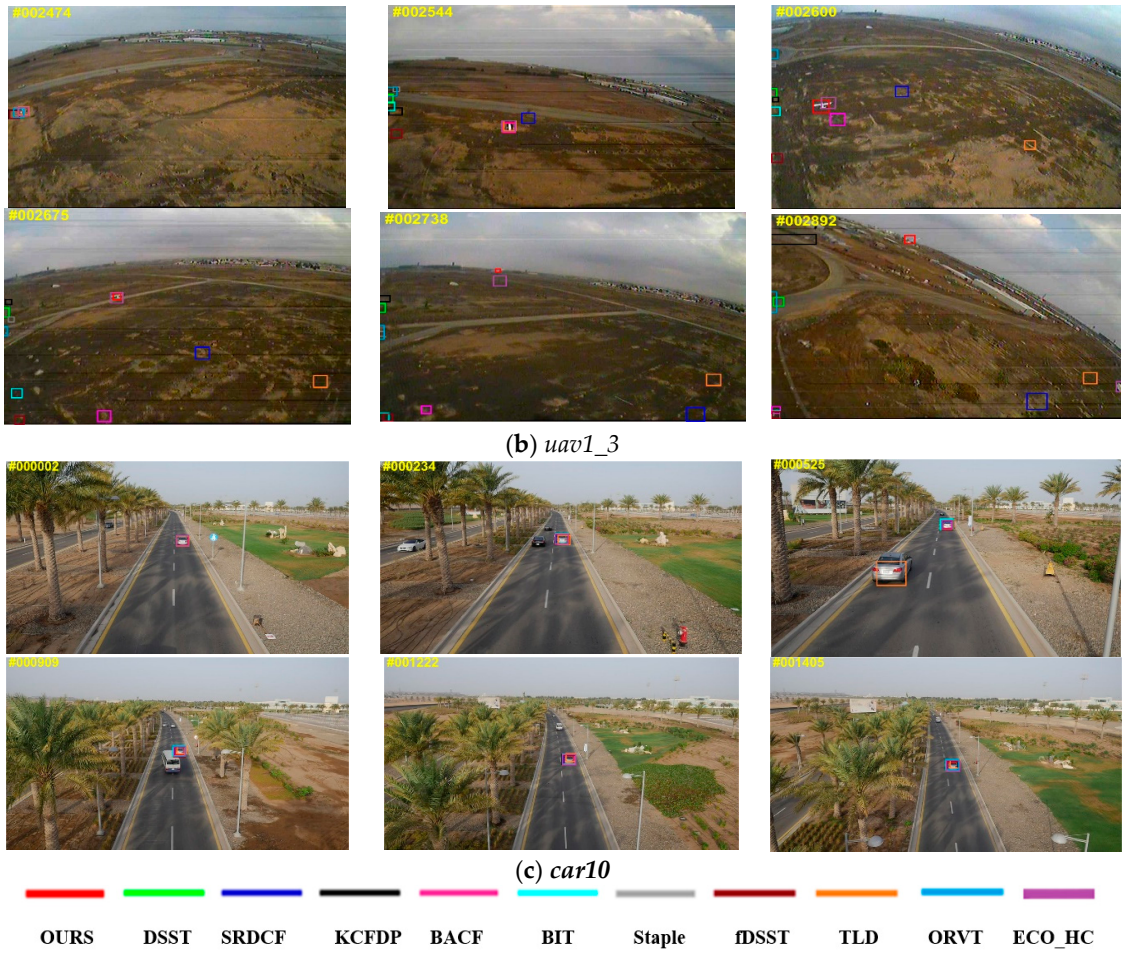


Figure 7. Qualitative evaluation of tracking results on UAV123 dataset.

5. Discussion

In this section, we discuss the tracking speed of different methods and assess the effect of each technical contribution incorporated within the proposed approach. All the experimental results are again taken on the twenty selected sequences of the UAV123 dataset, as indicated previously.

5.1. Speed Analyse

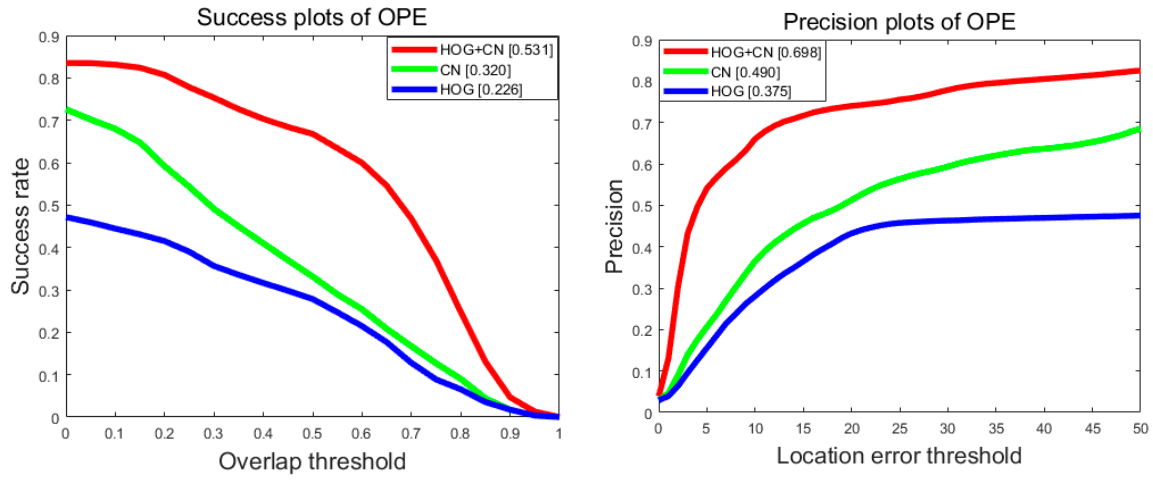
For practical applications of aerial tracking, the computational efficiency of a given tracker also needs to be considered. Table 6 lists the running speed of each compared tracker and the average speeds over all of the sequences are shown in the last row. As we can see, the fDSST tracker achieves the fastest running speed which is almost 133 fps and the biologically inspired BIT tracker performs well in terms of running efficiency, too. Mainly due to the low cost of computing the color histogram, the Staple tracker also has a good performance on tracking speed. However, SRDCF and BACF trackers show low running efficiencies on all of the twenty test sequences, which are approximately 10.79 fps and 9.65 fps, respectively, which still may not meet the standard of real-time running. It is worthwhile to note that our tracker can meet the real-time requirements, while gaining highly satisfactory results on both success rate and precision rate. This owes much to the robustness of fused features and the efficacy of saliency detection. To further strengthen the performance of our proposed tracker, we are trying to find an optimization method to speed it up.

Table 6. Running speed (frame per second) of each tracker on sequences from the UAV123 dataset.

	Ours	SRDCF	ECO_HC	KCFDP	BACF	Staple	BIT	fDSST	TLD	ORVT	DSST
truck3	45.99	19.28	79.73	131.08	13.04	99.41	139.15	**219.96**	2.90	26.04	145.87
bike2	42.59	26.27	82.51	102.42	14.67	102.45	179.93	**295.03**	1.44	36.43	238.96
boat6	52.76	16.83	78.13	53.81	12.26	101.18	142.74	**163.75**	10.03	32.59	157.43
building3	32.60	12.81	77.04	49.20	11.31	93.08	108.00	**165.08**	5.81	12.78	102.79
group2_2	16.46	7.78	76.02	53.29	9.21	86.66	81.79	**124.44**	12.64	28.91	60.64
person14_1	26.28	9.56	77.33	46.77	8.36	87.81	97.64	**157.11**	29.87	21.47	93.65
boat1	10.13	5.41	51.93	18.95	7.69	**51.28**	5.71	12.02	13.90	12.05	3.12
wakeboard3	37.78	12.90	76.47	55.53	11.73	98.71	106.63	**136.41**	21.68	25.01	104.89
group2_3	7.49	13.28	78.38	58.80	9.68	96.05	103.95	**170.87**	26.06	21.66	105.19
uav1_3	30.18	14.58	77.34	42.85	10.48	90.58	114.98	**176.30**	32.74	23.35	162.47
person10	8.67	4.96	74.40	39.20	5.51	**73.84**	49.09	68.28	27.15	24.13	25.72
car17	47.52	20.95	86.30	196.04	13.03	103.27	165.52	**273.97**	10.93	33.98	220.91
person16	17.18	5.63	72.06	43.89	6.87	82.86	78.10	**116.52**	17.95	14.49	51.68
car14	29.91	6.29	88.89	42.54	9.26	92.75	67.67	**112.94**	13.50	17.79	44.73
group1_1	11.45	5.39	75.11	32.94	8.25	**70.77**	50.82	34.82	14.56	25.56	25.42
person18	16.1	4.91	56.42	22.75	6.33	**30.78**	12.18	18.10	25.25	26.80	4.50
car10	26.63	7.68	76.76	33.40	8.55	86.67	84.75	**133.25**	16.81	19.19	73.95
person2_2	11.75	5.40	74.03	29.18	8.29	71.69	53.54	**88.44**	16.16	25.82	29.51
car3	37.27	12.16	80.86	54.14	12.99	93.77	124.41	**186.74**	17.86	28.73	111.87
person20	4.26	3.85	**50.52**	31.09	5.53	19.8	26.39	20.36	16.08	17.84	9.09
Average	25.65	10.79	74.51	56.89	9.65	81.67	89.64	**133.71**	16.66	23.73	88.61

5.2. Effect of Fused Features

Computationally, feature construction is an essential part of our tracker as it provides sufficient information for the correlation filter. We perform an experimental study to show the advantage of feature fusion. In particular, we test our tracker with fused features against a version of the tracker using only HOG or CN features. The results are reported in Figure 8. It is obvious that fused features lead to better performance in terms of both the precision rate and the success rate.



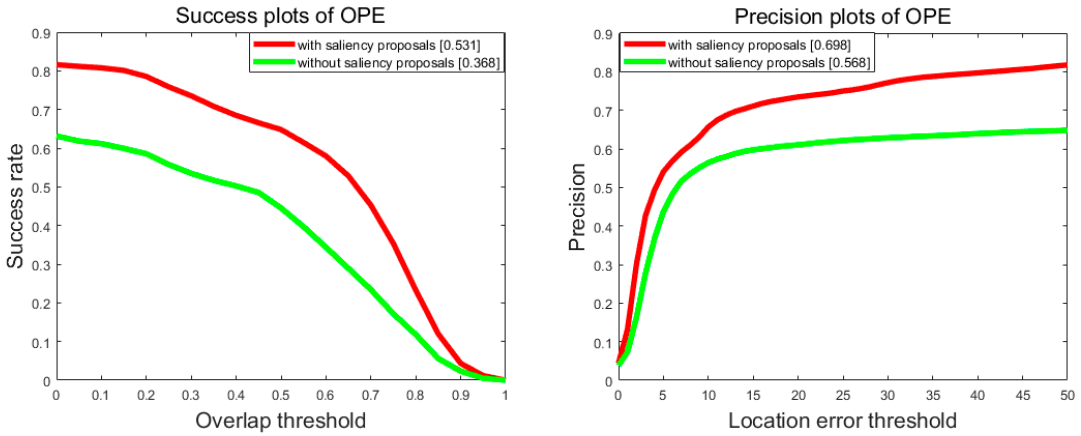
(**a**) Success plot (**b**) Precision plot

Figure 8. Tracking results using fused, color names (CN) or histograms of oriented gradient (HOG) features on 20 sequences from UAV123 dataset.

5.3. Effect of Saliency Proposals

To demonstrate the effectiveness of our saliency proposals in the detection stage, we evaluate the quantitative performance of our tracker with and without saliency proposals respectively. Note that almost all the sequences used for the experiments suffer from partial or full occlusion. The results are shown in Figure 9. Compared with the version without saliency proposals, the one utilizing saliency obtained exceedingly better performance. In addition, these results demonstrate that the

tracking-by-detection mechanism is very helpful once integrated with correlation-based tracking for occlusion-dominated scenes.



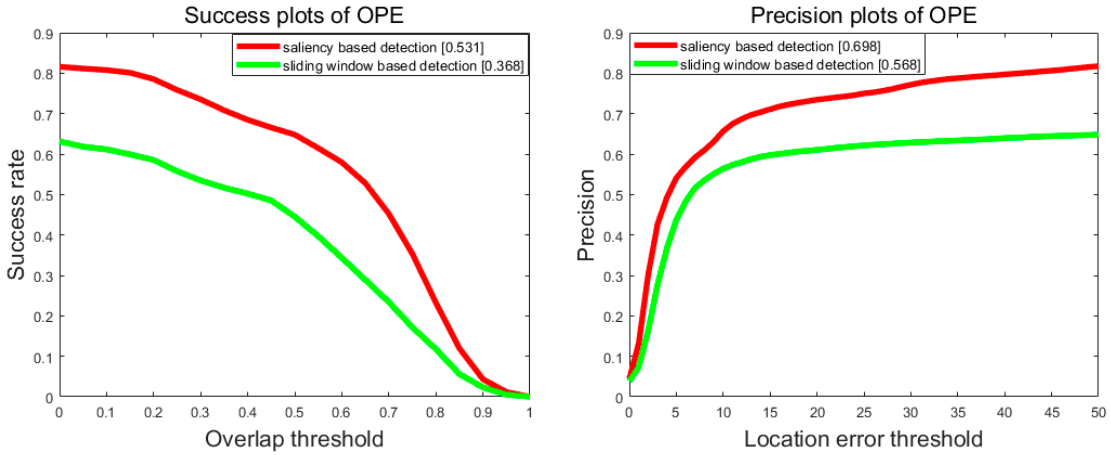
(**a**) Success plot (**b**) Precision plot

Figure 9. Tracking results with or without saliency proposals on 20 sequences from UAV123 dataset.

5.4. Comparison of Saliency-Based Detection and Sliding Widow Based Detection

To further testify the contribution of the saliency-based mechanisms, we use the traditional sliding window-based detection in substitution of the saliency-based within the generic framework of our tracking algorithm. Specifically, the detector is applied to the entire frame with sliding windows when $\max(\mathrm{r}) < \mathrm{T}_1$. In our implementation, the detector is trained by a random fern classifier [1], where each fern performs a number of pixel comparisons on an image patch with two feature vectors that point to the leaf-node with a certain posterior probability. The posteriors from all ferns are averaged as the target response and the detection is based on the use of the scanning window strategy. We use a k-nearest neighbor (KNN) classifier to select the most confident tracked results as positive training samples, e.g., a new patch is predicted as the target if k nearest feature vectors in the training set all have positive labels (k = 5 in this work).

Figure 10 presents the success plot and precision plots of these two trackers on the testing sequences. Obviously, our tracker performs significantly better on both evaluations. Due to fast motions of the UAV, great changes can occur to the scale and appearance of the target in the videos, which may reduce the similarity between the target and the corresponding tracking templates. Hence, it is hard for methods using a sliding window to obtain satisfactory results, which work by discriminating the target according to similarity measures between windows. What should not be ignored is that object tracking is closely related to attentional tasks in the biological world. Inspired by this observation, we exploit the abundant saliency information in the videos. Then, the adaptive template updating strategy ensures that new templates obtained by saliency detection can be introduced in time. This helps minimize the occurrence of possible model drift when the appearance of the target changes drastically.



(**a**) Success plot (**b**) Precision plot

Figure 10. Tracking results with saliency based detection and sliding window based detection on 20 sequences from UAV123 dataset.

Furthermore, we compare the speeds of these two methods. The results are illustrated in Figure 11. It can be seen that with the introduction of saliency information, the proposed approach achieves a higher running speed on the majority of testing sequences as compared to the version with sliding window-based detection. This can be expected because the proposed approach is intended to imitate biological vision systems that are able to pop-out the salient locations in the visual field [44] even under the most adverse conditions (e.g., highly cluttered scenes, low-light, etc.). These salient locations become the focus of attention for the post-attentive stages of visual processing, which can effectively provide proposals for target relocation. However, for the detector without the use of saliency detection, every tracking outcome is computed via running a sliding window, inevitably at the expense of costing more computing resource.

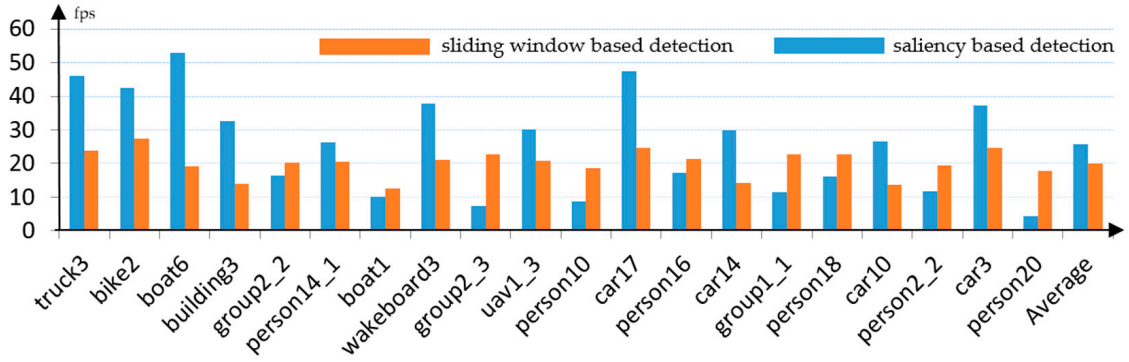


Figure 11. Running speeds of tracking methods with saliency-based detection and sliding window based detection on 20 sequences from UAV123.

6. Conclusions

In this paper, we have proposed a robust tracking method for UAV videos via fused feature based correlation filter and saliency detection. The correlation filter that combines the HOG and dimension-reduced CN features leads to significant contribution in tracking performance while dealing with challenging factors such as occlusion, noise and illumination. To handle serious occlusion, this work has introduced saliency information into the tracker as redetection, thereby reducing background interference. Moreover, an adaptive model update strategy is adopted to alleviate possible model drifts, which is both robust and computationally efficient. Experimental investigations have demonstrated, both quantitatively and qualitatively, that our approach achieves favorable results on the

average performance for two popular aerial tracking datasets in comparison with the state-of-the-art methods. Given its reliability and robustness, the proposed tracker can be successfully employed in a wide variety of UAV video applications (beyond those related to surveillance), such as wild-life monitoring, activity control, navigation/localization, and obstacle/object avoiding, especially when real-time processing is mandatory, as in the case of rescue or defense purposes.

As a generic approach for aerial videos, we plan to further develop more robust fused features and to reinforce the fast nature of the redetect methods in future, while operating in real-time. Also, in this work, it has been assumed that each-channel feature is independent of the rest and hence, no interaction between such features has been considered. As such, a channel-wise filter was successfully adopted. However, it would be interesting to explore the interconnections among the information contents conveyed by different channels and to introduce a general linear filter to deal with such cases.

Author Contributions: All the authors made significant contributions to this work. X.X. and Y.L. devised the approach and analyzed the data; Q.S. provided advice for the preparation and revision of the work; X.X. performed the experiments; and H.D. helped with the experiments.

Funding: This work was supported by the National Natural Science Foundation of China (61871460, 61876152), the National Key Research and Development Program of China (2016YFB0502502), and the Foundation Project for Advanced Research Field of China (614023804016HK03002).

Acknowledgments: The authors would like to thank the editors and the anonymous referees for their constructive comments which have been very helpful in revising this paper.

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

References

1. Kalal, Z.; Matas, J.; Mikolajczyk, K. Tracking-Learning-Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 1409–1422. [CrossRef] [PubMed]
2. Hare, S.; Saffari, A.; Torr, P.H.S. Struck: Structured Output Tracking with Kernels. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Barcelona, Spain, 6–13 November 2011; pp. 263–270.
3. Lu, H.; Jia, X.; Yang, M.H. Visual tracking via adaptive structural local sparse appearance model. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012; pp. 1822–1829.
4. Blake, A.; Isard, M. *Active Contours: The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion*; Springer Science Business Media: Berlin, Germany, 2012.
5. Battiato, S.; Farinella, G.M.; Furnari, A.; Puglisi, G.; Snijders, A.; Spiekstra, J. An integrated system for vehicle tracking and classification. *Expert Syst. Appl.* **2015**, *42*, 7263–7275. [CrossRef]
6. Andriluka, M.; Roth, S.; Schiele, B. People-tracking-by-detection and people-detection-by-tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.
7. Zingoni, A.; Diani, M.; Corsini, G. A Flexible Algorithm for Detecting Challenging Moving Objects in Real-Time within IR Video Sequences. *Remote Sens.* **2017**, *9*, 1128. [CrossRef]
8. Hou, X.; Zhang, L. Saliency Detection: A Spectral Residual Approach. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Minneapolis, MN, USA, 18–23 June 2007; pp. 1–8.
9. Jiang, H.; Wang, J.; Yuan, Z.; Wu, Y. Salient Object Detection: A Discriminative Regional Feature Integration Approach. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 2083–2090.
10. Li, X.; Li, Y.; Shen, C.; Dick, A.; Hengel, A.V.D. Contextual hypergraph modeling for salient object detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Sydney, Australia, 1–8 December 2014; pp. 3328–3335.
11. Wan, M.; Gu, G.; Qian, W.; Ren, K.; Chen, Q.; Zhang, H.; Maldague, X. Total Variation Regularization Term-Based Low-Rank and Sparse Matrix Representation Model for Infrared Moving Target Tracking. *Remote Sens.* **2018**, *10*, 510. [CrossRef]

12. Dalal, N.; Triggs, B. Histograms of Oriented Gradients for Human Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Diego, CA, USA, 20–26 June 2005; pp. 886–893.
13. Adam, A.; Rivlin, E.; Shimshoni, I. Robust Fragments-Based Tracking Using the Integral Histogram. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), New York, NY, USA, 17–22 June 2006; pp. 798–805.
14. Babenko, B.; Yang, M.-H.; Belongie, S. On-line boosting and vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami, FL, USA, 20–26 June 2009; pp. 983–990.
15. Grabner, H.; Bischof, H. On-line boosting and vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), New York, NY, USA, 17–22 June 2006; pp. 260–267.
16. Zhang, K.; Zhang, L.; Liu, Q. Fast visual tracking via dense spatio-temporal context learning. In Proceedings of the 2014 European Conference on Computer Vision (ECCV), Zurich, Switzerland, 6–12 September 2014; pp. 127–141.
17. Oron, S.; Bar-Hillel, A.; Avidan, S. Extended Lucas-Kanade Tracking. In Proceedings of the 2014 European Conference on Computer Vision (ECCV), Zurich, Switzerland, 6–12 September 2014; pp. 142–156.
18. Yang, R.; Wei, Z. Real-Time Visual Tracking through Fusion Features. *Sensors* **2016**, *16*, 949.
19. Danelljan, M.; Hager, G.; Khan, F.S.; Felsberg, M. Accurate scale estimation for robust visual tracking. In Proceedings of the British Machine Vision Conference (BMVC), Nottingham, UK, 1–5 September 2014; pp. 65.1–65.11.
20. Galoogahi, H.K.; Fagg, A.; Lucey, S. Learning Background-Aware Correlation Filters for Visual Tracking. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1144–1152.
21. Zhu, G.; Wang, J.; Wu, Y.; Lu, H. Collaborative Correlation Tracking. In Proceedings of the British Machine Vision Conference, Swansea, UK, 7–10 September 2015; pp. 184.1–184.12.
22. Ma, C.; Yang, X.; Zhang, C.; Yang, M.H. Long-term correlation tracking. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5388–5396.
23. Logoglu, K.B.; Lezki, H.; Yucel, M.K. Feature-Based Efficient Moving Object Detection for Low-Altitude Aerial Platforms. In Proceedings of the IEEE International Conference on Computer Vision Workshop, Venice, Italy, 22–29 October 2017; pp. 2119–2128.
24. Fu, C.; Duan, R.; Kircali, D. Onboard Robust Visual Tracking for UAVs Using a Reliable Global-Local Object Model. *Sensors* **2016**, *16*, 1406. [CrossRef] [PubMed]
25. Bolme, D.S.; Beveridge, J.R.; Draper, B.A.; Lui, Y.M. Visual object tracking using adaptive correlation filters. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 13–18 June 2010; pp. 2244–2250.
26. Henriques, F.; Caseiro, R.; Martins, P.; Batista, J. Exploiting the Circulant Structure of Tracking-by-Detection with Kernels. In Proceedings of the European Conference on Computer Vision (ECCV), Florence, Italy, 7–13 October 2012; pp. 702–715.
27. Boddeti, V.N.; Kanade, T.; Kumar, B.V. Correlation filters for object alignment. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 2291–2298.
28. Danelljan, M.; Khan, F.S.; Felsberg, M.; van de Weijer, J. Adaptive Color Attributes for Real-Time Visual Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 1090–1097.
29. Henriques, 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]
30. Bertinetto, L.; Valmadre, J.; Golodetz, S.; Miksik, O.; Torr, P.H.S. 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. Danelljan, M.; Hager, G.; Khan, F.S.; Felsberg, M. Learning Spatially Regularized Correlation Filters for Visual Tracking. In Proceedings of the 2016 IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2016; pp. 4310–4318.

32. Danelljan, M.; Bhat, G.; Khan, F.S.; Felsberg, M. ECO: Efficient Convolution Operators for Tracking. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6931–6939.
33. Goferman, S.; Zelnik-Manor, L.; Tal, A. Context-aware saliency detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 1915–1926. [CrossRef] [PubMed]
34. Cheng, M.M.; Warrell, J.; Lin, W.Y.; Zheng, S.; Vineet, V.; Crook, N. Efficient Salient Region Detection with Soft Image Abstraction. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 1529–1536.
35. Benenson, R.; Omran, M.; Hosang, J.; Schiele, B. Ten Years of Pedestrian Detection, What Have We Learned? In Proceedings of the European Conference on Computer Vision Workshops, Zurich, Switzerland, 6–7 September 2014; pp. 613–627.
36. Khan, R.; Weijer, J.V.D.; Khan, F.S.; Muselet, D.; Ducottet, C.; Barat, C. Discriminative Color Descriptors. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 2866–2873.
37. Berlin, B.; Kay, P. *Basic Color Terms: Their Universality and Evolution*; University of California Press: Berkeley, CA, USA, 1991.
38. Roth, D.B.G. Adaptive Thresholding using the Integral Image. *J. Graph. Tools* **2007**, *12*, 13–21.
39. VIVID Tracking Evaluation Web Site. Available online: http://vision.cse.psu.edu/data/vividEval/datasets/datasets.html (accessed on 22 April 2018).
40. Mueller, M.; Smith, N.; Ghanem, B. A Benchmark and Simulator for UAV Tracking. In Proceedings of the 2016 European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016; pp. 445–461.
41. Huang, D.; Luo, L.; Wen, M.; Chen, Z. Enable Scale and Aspect Ratio Adaptability in Visual Tracking with Detection Proposals. In Proceedings of the 2015 British Machine Vision Conference, Swansea, UK, 7–10 September 2015; pp. 185.1–185.12.
42. Cai, B.; Xu, X.; Xing, X.; Jia, K.; Miao, J.; Tao, D. BIT: Biologically Inspired Tracker. *IEEE Trans. Image Process.* **2016**, *25*, 1327–1339. [CrossRef] [PubMed]
43. Danelljan, M.; Häger, G.; Khan, F.S.; Felsberg, M. Discriminative Scale Space Tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1561–1575. [CrossRef] [PubMed]
44. Mahadevan, V.; Nuno, V. Saliency-based discriminant tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami, FL, USA, 20–25 June 2009; pp. 1007–1013.