Proceedings of the 2018 IEEE
International Conference on Robotics and Biomimetics
December 12-15, 2018, Kuala Lumpur, Malaysia

Robust Scalable Part-Based Visual Tracking for UAV with Background-Aware Correlation Filter

Changhong Fu[1,*], Yinqiang Zhang[1], Ran Duan[2], and Zongwu Xie[3,*]

***Abstract*— Robust visual tracking for the unmanned aerial vehicle (UAV) is a challenging task in different types of civilian UAV applications. Although the classical correlation filter (CF) has been widely applied for UAV object tracking, the background of the object is not learned in the classical CF. In addition, the classical CF cannot estimate the object scale changes, and it is not able to cope with object occlusion effectively. Part-based tracking approach is often used for the visual tracker to solve the occlusion issue. However, its real-time performance for the UAV cannot be achieved due to the high cost of object appearance updating. In this paper, a novel robust visual tracker is presented for the UAV. The object is initially divided into multiple parts, and different background-aware correlation filters are applied for these divided object parts, respectively. An efficient coarse-to-fine strategy with structure comparison and Bayesian inference approach is proposed to locate object and estimate the object scale changes. In addition, an adaptive threshold is presented to update each local appearance model with a Gaussian process regression method. Qualitative and quantitative tests show that the presented visual tracking algorithm reaches real-time performance (i.e., more than twenty frames per second) on an i7 processor with 640×360 image resolution, and performs favorably against the most popular state-of-the-art visual trackers in terms of robustness and accuracy. To the best of our knowledge, it is the first time that this novel scalable part-based visual tracker is presented, and applied for the UAV tracking applications.**

I. INTRODUCTION

Visual object tracking is an important task for the unmanned aerial vehicle (UAV) with numerous applications such as reconnaissance and surveillance [1], midair monitoring [2], wildlife protection [3], and unknown environment exploration [4]. In recent years, different visual tracking approaches have been developed for the UAV, but the vision-based UAV tracking remains as a challenging task due to the object appearance changes caused by occlusion, scale variation, illumination change, shape deformation, out-of-plane or in-plane rotation, and onboard mechanical vibration. Therefore, a more robust tracking algorithm is required to achieve higher accurate in real-time UAV tracking applications, as one example shown in Fig. 1.

In literature, UAV tracking methods are classified as either generative or discriminative approaches. Generative approach casts the UAV tracking problem as searching for the region which is the most similar to the tracked object. The tracking object is often represented by a set of templates [5] or a group of basis vectors from a subspace [6]. However, it assumes that the object appearance does not change significantly during the appearance updating procedure. Different from the generative method, discriminative approach (also refers to the tracking-by-detection method) treats tracking as a binary classification problem to distinguish the tracking object from object background. For instance, a visual tracking algorithm with structured output tracking with kernels (STRUCK) [7] is employed for UAV to achieve person following. In addition, a visual tracker, which is developed with compressive sensing [8] is used to track freewill object in UAV applications.

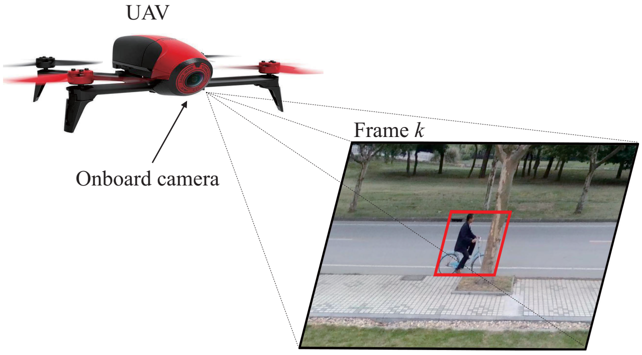


Fig. 1. Visual object tracking of UAV. The appearance of the object (i.e., Biker) is changing due to the occlusion, scale variation, illumination change, shape deformation, and onboard mechanical vibration.

Recently, the correlation filter (CF)-based discriminative method has been widely applied to various UAV tracking tasks with high-speed and promising tracking performances. A tracking system is designed in [9] for UAV to track a maneuvering target with kernelized correlation filter (KCF) [10]. The CF is implemented in [11] for UAV to achieve real-time, smooth, and long-term object following in indoor and outdoor practical scenarios. Moreover, the KCF is used to generate image patch confidence in [12], measuring object tracking reliability in the UAV tracking application. However, these classical CF-based trackers confront boundary effects and severe impacts of learning from circularly shifted samples of the foreground object. Negative examples, implicitly generated by circulant property of correlation, are actually synthetic and cannot represent true negative samples from the background, leading to suboptimal tracking results. In addition, the classical CF-based tracker cannot handle scale variation.

[1] Changhong Fu and Yinqiang Zhang are with the School of Mechanical Engineering, Tongji University, Shanghai 201804, China changhongfu@tongji.edu.cn

[2] Ran Duan is with the Interdisciplinary Division of Aeronautical and Aviation Engineering, Hong Kong Polytechnic University, HKSAR, China

[3] Zongwu Xie is with the State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin, China xiezongwu@hit.edu.cn

978-1-7281-0377-8/18/$31.00 © 2018 IEEE

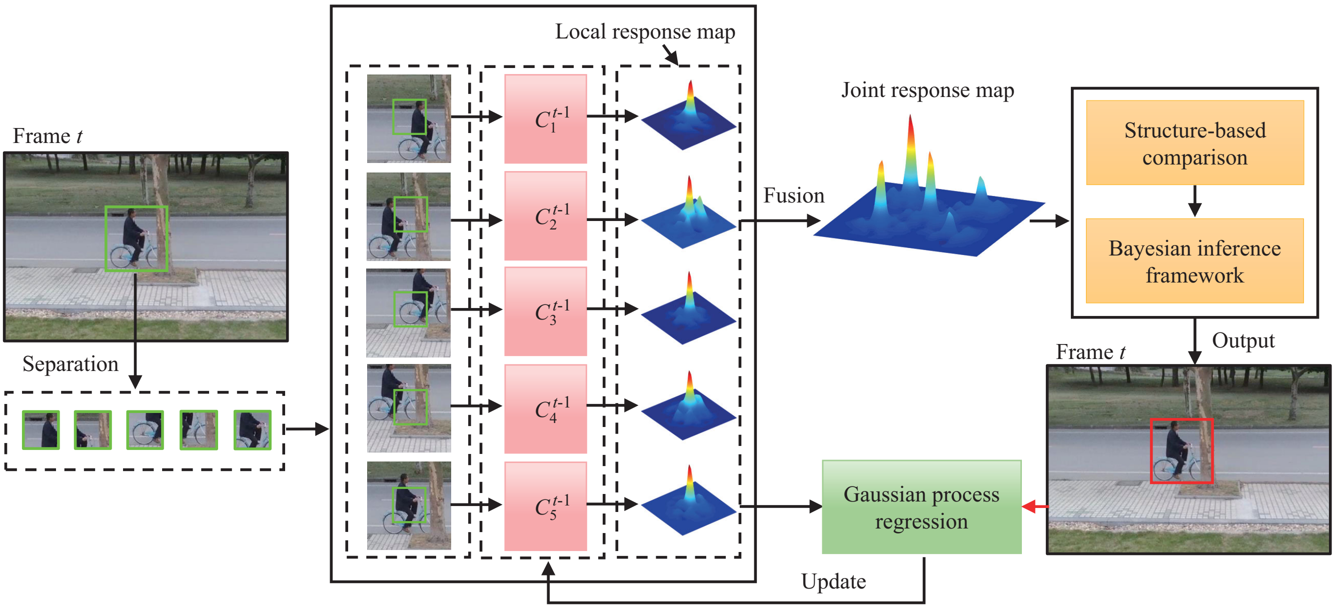


Fig. 2. Scalable part-based visual tracker for UAV with background-aware correlation filter. Adaptive classifier C_i^{t-1}, i.e., local region appearance model, is updated with an online background-aware correlation filter on the (t-1)-th frame, and then applied to estimate the local region response on the t-th frame.

In literature, several background-aware trackers are proposed to address the aforementioned issues. The correlation filters with limited boundaries (CFLB) are presented in [13] to extract real negative samples from background regions by incorporating a cropping operator, i.e., binary matrix. Additionally, a tracker with learning background-aware correlation filters (BACF) is developed in [14], which incorporates multi-channel features and employs an efficient augmented Lagrangian method for filter learning, improving the tracking robustness against background noise. Nevertheless, the CF-based tracker with holistic appearance model is prone to be dominated by occluded regions of the tracking object. Existing tracking methods solve this issue by using multiple local appearance models, which achieve promising results. Specifically, when the object is partially occluded, the remaining visible parts from the object is able to maintain visible cues for tracking. Several local appearance model-based trackers have combined the classical CF for visual tracking applications. For example, the part-based visual tracker presented in [15] separates the whole object into multiple sub-regions, and then employs the KCF trackers for each sub-region to achieve comprehensive object tracking. However, the boundary effect and severe impacts of learning from circularly shifted samples of the foreground target are still challenging issues for the object tracking performance in [15].

In this paper, we propose a novel visual tracker for UAV. The overview of the presented visual tracker is illustrated in Fig. 2. In summary, the main contributions of our work are listed as follows:

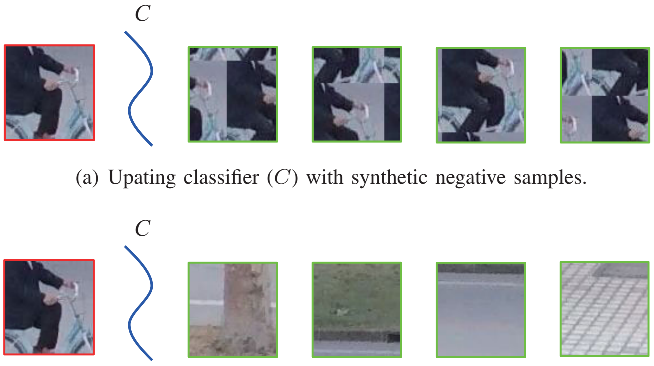
- A novel scalable part-based visual tracker is proposed, and applied for the UAV object tracking applications.
- Each part of the tracking object is tracked by using a background-aware discriminative tracking approach.
- A novel approach with coarse-to-fine strategy is presented to estimate the location and scale changes of tracking object.
- A novel adaptive threshold is proposed to update each local appearance model with a Gaussian process regression method.

Qualitative and quantitative UAV flight experiments show that the presented visual tracking algorithm achieves real-time performance (i.e., more than twenty frames per second) on an i7 processor with 640×360 image resolution, and outperforms the most popular state-of-the-art visual trackers in terms of robustness and accuracy.

The outline of the paper is organized as follows: Section II introduces the tracking approach with background-aware CF. Section III introduces the presented novel visual tracking algorithm, i.e., scalable part-based visual tracker. Section IV presents the performance evaluations and comparisons with the most popular state-of-the-art CF-based visual trackers. Finally, the concluding remarks are given in Section V.

II. TRACKING WITH BACKGROUND-AWARE CORRELATION FILTER

The classical CF trackers have benefited from the dense sampling with cyclic shifts, as the shifted samples shown in Fig. 3(a). However, such operation discards background information. In addition, it brings the boundary effect, which degrades the discriminative ability of the tracker. As a result, these trackers are prone to providing suboptimal performance in UAV tracking applications. To address these problems, a background-aware CF tracker is developed based on [14] in this work. It can provide a superior solution and improve the tracking performance for UAV. In this section, we briefly



(a) Upating classifier (C) with synthetic negative samples.

(b) Upating classifier (C) with real negative samples.

Fig. 3. Difference between the classical CF and background-aware CF.

introduce the background-aware CF (i.e., BACF) tracking method.

Remark 1: In classical CF tracker, synthetic negative training samples are generated to update the CF, as the samples with green rectangles shown in Fig. 3(a). However, the background information, i.e., real negative sample, is extracted for updating in the background-aware CF tracker, as the samples with green rectangles shown in Fig. 3(b).

1) Filter training: In the classical CF framework, the filter $\mathbf{w}$ is trained with the ridge regression approach. Its raw vectorized samples with length N are derived from image patch $\mathbf{x}$. To avoid boundary effects and gain real samples from the background, a cropping operation is employed to get training examples with a smaller size $M, M \ll N$. In the Fourier domain, the objective function $\mathcal{E}(\mathbf{w}, \hat{\mathbf{g}})$ is formulated as following to obtain desired parameters of filter $\mathbf{w}$:

$$\mathcal{E}(\mathbf{w}, \hat{\mathbf{g}}) = \frac{1}{2}\|\sum_{d=1}^{D}\hat{\mathbf{X}}_d\hat{\mathbf{g}}_d - \hat{\mathbf{y}}\|_2^2 + \frac{\lambda}{2}\sum_{d=1}^{D}\|\mathbf{w}_d\|_2^2, \quad (1)$$

where $\hat{\mathbf{y}}$ is the vectorized Gaussian regression label. The subscript d denotes the d-th one of D feature channels. $\hat{\mathbf{X}}_d$ is defined as $\hat{\mathbf{X}}_d = \mathrm{diag}(\hat{\mathbf{x}}_d)$ and $\hat{\mathbf{g}}$ is an auxiliary variable, which can be expressed as $\hat{\mathbf{g}}_d = \mathcal{F}(\mathbf{B}^\top\mathbf{w}_d)$. The symbol $\hat{}$ and $\mathcal{F}$ represent the discrete Fourier transform. An alternative formulation is $\hat{\mathbf{g}}_d = \sqrt{N}\mathbf{F}\mathbf{B}^\top\mathbf{w}_d$, where $\mathbf{F}$ is an orthonormal $N \times N$ mapping matrix for Fourier transform. The $M \times N$ binary matrix $\mathbf{B}$ implements cropping operation, which is able to crop the mid M elements from the raw signal with size N. The $\top$ denotes the conjugate transpose of a matrix or vector. λ is the coefficient for the Tikhonov regularization term. To solve the lack of closed-form solution in Eq. 1, an augmented Lagrangian method (ALM) [16] is applied. The specific Lagrangian function is able to be reformulated without single channel representation:

$$\begin{aligned}\mathcal{L}(\mathbf{w}, \hat{\mathbf{g}}, \hat{\boldsymbol{\zeta}}) = &\frac{1}{2}\|\hat{\mathbf{X}}\hat{\mathbf{g}} - \hat{\mathbf{y}}\|_2^2 + \frac{\lambda}{2}\|\mathbf{w}\|_2^2 \\ &+ \hat{\boldsymbol{\zeta}}^\top\left(\hat{\mathbf{g}} - \sqrt{N}(\mathbf{F}\mathbf{B}^\top \otimes \mathbf{I}_K)\mathbf{w}\right) \quad (2) \\ &+ \frac{\mu}{2}\|\hat{\mathbf{g}} - \sqrt{N}(\mathbf{F}\mathbf{B}^\top \otimes \mathbf{I}_K)\mathbf{w}\|_2^2,\end{aligned}$$

where μ is the trade-off penalty parameter and $\hat{\boldsymbol{\zeta}}$ is the Lagrangian parameters in the Fourier domain. $\mathbf{I}_K$ is $K \times K$ identity matrix. Using Kronecker product $\otimes$, the reformulated term is $\sum_D \hat{\mathbf{X}}_d\hat{\mathbf{g}}_d = \sqrt{N}\hat{\mathbf{X}}(\mathbf{F}\mathbf{B}^\top \otimes \mathbf{I}_K)\mathbf{w}$.

The ALM problem in Eq. 2 can be solved iteratively by alternating direction method of multipliers (ADMM). This primal problem can be separated into two subproblems, which can obtain analytic solutions, i.e., $\mathbf{w}^*$ and $\hat{\mathbf{g}}^*$, respectively. Moreover, with sparse banded property and the Sherman-Morrison formula [17], ADMM iterations can make a real-time tracking performance.

2) Object detection: The location and scale changes of the tracking object in frame t is estimated with a new image patch $\mathbf{z}^t$ and the auxiliary variable $\hat{\mathbf{g}}^{t-1}$. With multiple resolutions of the searching area, a maximum correlation filter response can be determined in order to estimate the object location and scale changes:

$$\hat{\mathbf{s}}^t = \arg\max_{\mathbf{s}}\{\hat{\mathbf{z}}^t(\mathbf{s}) \odot \hat{\mathbf{g}}^{t-1}\}, \quad (3)$$

where $\hat{\mathbf{s}}^t$ is the expected location and scale changes of tracking object. $\odot$ denotes element-wise product.

3) Filter updating: The filter is updated with below strategy:

$$\tilde{\mathbf{x}}^t = (1-\alpha)\tilde{\mathbf{x}}^{t-1} + \alpha\mathbf{x}^t, \quad (4)$$

where $\tilde{\mathbf{x}}^t$ is appearance model that is obtained from $\mathbf{x}^t$ and $\tilde{\mathbf{x}}^{t-1}$. α is a constant learning rate.

Remark 2: It is noted that the filter in the BACF tracker is updated frame-by-frame. In addition, it is not able to deal with object occlusion effectively.

III. SCALABLE PART-BASED VISUAL TRACKER WITH BACF

In UAV tracking applications, some typical challenging factors, e.g., object occlusion and scale variation, are prone to the deterioration of UAV tracking performance. To solve these challenging issues, a novel scalable part-based tracking method has been presented in this work.

A. Response Map Fusion with Adaptive Weights

As shown in Fig. 2, the tracking object is divided into multiple parts. For each part, an independent classifier, i.e., BACF, is used to provide local response map f_i^t. Finally, these local response maps are fused into a joint response map f^t to locate the tracking object. To improve the robustness of UAV tracking, adaptive weight, i.e., the importance of each local response map, is designed based on two parameters [15]: (1) peak-to-sidelobe ratio (PSR): it evaluates the sharpness of response map. (2) smooth constraint of confidence maps (SCCM): it evaluates the smoothness of response map. The adaptive weight β_i^t of each part is defined as:

$$\beta_i^t = \gamma\frac{1}{SCCM_i^t} + PSR_i^t, \quad (5)$$

where γ is a trade-off parameter between the sharpness and temporal smoothness of response map. $SCCM_i^t$ is the smoothness of the i-th response map on the t-th image frame.

PSR_i^t is the sharpness of the i-th response map on the t-th image frame. The joint response map f^t, which combines different local response maps with corresponding adaptive weights, is defined as:

$$f^t = \sum \beta_i^t f_i^t. \tag{6}$$

Remark 3: As the joint response map shown in the Fig. 2, the effects of the occluded parts can be suppressed in order to reduce their contributions for locating UAV object.

B. Tracking with Structure Comparison and Bayesian Inference Framework

A novel approach with coarse-to-fine strategy is presented to estimate the location and scale changes of tracking object. Specifically, the structures of tracking object on two consecutive frames are compared for estimating the initial location and scale changes firstly, and then a Bayesian inference framework is used to obtain the final object location and scale changes.

1) Structure Comparison: After obtained the joint response map, the coarse location and scale changes of tracking object are estimated based on the local response maps. To achieve a coarse estimation, the shift vectors of all parts are applied to get the result of object translation. In details, the translation is calculated with the shift vectors $\mathbf{v}_i^t$ and their trust scores $\omega_i^t = \frac{\beta_i^t}{\sum \beta_j^t}$. The shift vector of the tracking object $\mathbf{v}^t$ is defined as:

$$\mathbf{v}^t = \sum_i \omega_i^t \mathbf{v}_i^t. \tag{7}$$

In Eq. 7, a higher ω_i^t represents a higher trust-level of this part. The translation of the tracking object is determined with shift vectors of reliable parts. The contributions of occluded parts are reduced to maintain the tracking robustness.

To estimate the scale changes, we propose a method based on the structure of all local response maps. In this approach, the scale changes of the tracking object can refer to the distribution of its reliable local response maps. Let $e_i = \|\mathbf{v}_i^t - \mathbf{v}^t\|$ be the error. The standard deviation σ_e of these errors, which represents how spread out the vectors $\mathbf{v}_i^t$ are, is calculated as the threshold to select reliable local response maps. If $e_i > \sigma_e$, the corresponding local response map will not be reliable and be discarded.

Let σ_s^t and σ_s^{t-1} denote the standard deviation of peak locations of reliable local response maps at frame t and $t-1$, the coarse estimation of scale changes for tracking object is the ratio of σ_s^t and σ_s^{t-1}, i.e., $\frac{\sigma_s^t}{\sigma_s^{t-1}}$. In this work, the tracking location and scale changes are updated initially with the shift vector $\mathbf{v}^t$ and the ratio $\frac{\sigma_s^t}{\sigma_s^{t-1}}$.

Remark 4: In the UAV tracking applications, object tracking with multiple parts mainly has three characteristics: (1) the object does not show a drastic location and scale changes in two consecutive frames. (2) all tracking parts without occlusion are constrained with a similar movement. (3) most of the parts maintain a similar location distribution. Therefore, the initial location and scale variation of the tracking object are estimated based on the structure comparison.

2) Bayesian Inference Framework: In this framework, the final object location and scale changes are estimated with the initial results obtained from structure comparison. The object state $\mathbf{s}^t$ is formulated with affine motion, it is defined as:

$$\hat{\mathbf{s}}^t = \arg\max_{\mathbf{s}_j^t} p(\mathbf{s}_j^t|\mathbf{z}^{1:t}), \tag{8}$$

where $\mathbf{z}^{1:t}$ is the measurement set with respect to the joint confidence map, i.e., $\mathbf{z}^{1:t} = \{\mathbf{z}_i, i = 1, \cdots, k\}$. $\mathbf{s}_j^t$ is the state of the j-th sample. To model the tracking process, the Chapman-Kolmogorov equation is used, i.e.:

$$p(\mathbf{s}^t|\mathbf{z}^{1:t}) \propto p(\mathbf{z}^t|\mathbf{s}^t) \int p(\mathbf{s}^t|\mathbf{s}^{t-1}) p(\mathbf{s}^{t-1}|\mathbf{z}^{1:t-1}) d\mathbf{s}^{t-1}. \tag{9}$$

In Eq. 9, system model $p(\mathbf{s}^t|\mathbf{s}^{t-1})$ is defined as:

$$p(\mathbf{s}^t|\mathbf{s}^{t-1}) \sim \mathcal{N}(\mathbf{s}^t, \tilde{\mathbf{s}}^{t-1}, \mathbf{\Psi}), \tag{10}$$

where $\tilde{s}^{t-1}$ is based on the coarse estimation of location and scale from III-B.1. $\mathbf{\Psi}$ denotes a diagonal covariance matrix whose elements are the variances of affine parameters.

Measurement model $p(\mathbf{z}^t|\mathbf{s}^t)$ in Eq. 9 is defined as:

$$p(\mathbf{z}^t|\mathbf{s}^t) = \sum f^t(\mathbf{s}_j^t) \odot \frac{M^t}{|M^t|}, \tag{11}$$

where M^t denotes the cosine window spatial mask whose peak depends on the maximum of local response maps. $|\cdot|$ is the number of the pixels in the corresponding bounding box. $f^t(\mathbf{s}_j^t)$ is the response patch of the state $\mathbf{s}_j^t$ from joint response map.

Remark 5: Calculating the maximum posterior $p(\mathbf{s}^t|\mathbf{z}^{1:t})$ in Eq. 8 is equivalent to obtain the maximum of the likelihood $p(\mathbf{z}^t|\mathbf{s}^t)$. The traditional likelihood is calculated based on a set of eigenbasis vectors or templates. Inspired by [15], response maps are applied in this work for calculating the likelihood, which significantly simplifies the computation. In Fig. 4, the summation of the response scores in a bounding box with respect to each sampling candidate is calculated directly.

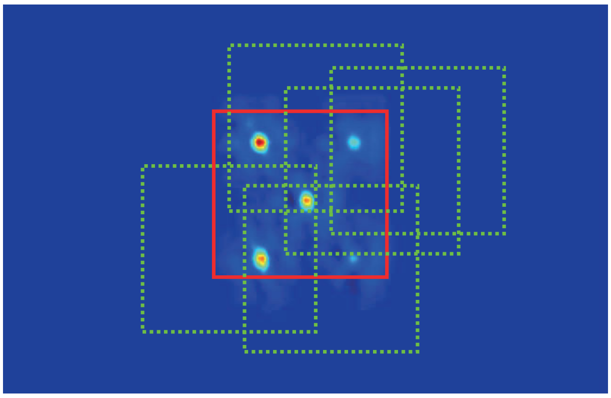


Fig. 4. Calculation of likelihood on joint response map.

C. Updating Classifier with Gaussian Process Regression

A novel adaptive threshold is proposed to update each local appearance model, i.e., classifier. In this work, we model the relationship between PSR and $SCCM$ with Gaussian process regression (GPR) to achieve adaptive updating. This relationship is formulated as a set of functions, i.e., $g : a \in \mathbb{R} \rightarrow g(u) \in \mathbb{R}$, where $u = PSR_i^t$ and $g(u) = SCCM_i^t$. The Gaussian process (GP) model describes the distribution of this function set:

$$g(u) \sim \mathcal{GP}\big(\mathbf{m}(u), \mathbf{G}(u, u')\big), \tag{12}$$

where $\mathcal{GP}$ denotes Gaussian process. $\mathbf{m}(u)$ and $\mathbf{G}(u, u')$ are the mean function and covariance function of this set of functions. This covariance function specifies the covariance between pairs of PSR_i:

$$k(q, q') = {\sigma_f}^2 \exp\big(-\frac{1}{2l^2}(q - q')^2\big), \tag{13}$$

where σ_f and l are hyperparameters. q and q' are the inputs, i.e., PSR_i values. After the normalization of raw inputs PSR_i and outputs $SCCM_i$, the zero-mean distribution of the functions is formulated with the following prediction for each tracking part i at frame t:

$$\begin{bmatrix} \mathbf{y} \\ g_* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K}(\mathbf{a}, \mathbf{a}) + {\sigma_n}^2\mathrm{I} & \mathbf{k}(\mathbf{x}, a_*) \\ \mathbf{k}(a_*, \mathbf{a}) & k(a_*, a_*) \end{bmatrix}\right), \tag{14}$$

where $y = g(a) + \epsilon$ is the element of $\mathbf{y}$, which are the noisy observations of all functions. ϵ is Gaussian noise with variance matrix ${\sigma_n}^2\mathrm{I}$. a_* is normalized value of PSR_i^t and the elements of vector $\mathbf{a}$ are normalized values of previous PSR_i. g_* is normalized value of $SCCM_i^t$. To improve the update performance, we only select the $\mathbf{a}$ and $\mathbf{y}$ from $t - t_r$ to $t - 1$ frames, where t_r is the length of inputs memory. This approach makes the GP model focus more on the recent inputs and discard the distant ones. $\mathbf{K}(\cdot, \cdot)$ and $\mathbf{k}(\cdot, \cdot)$ denote the convariance matrix and vector of inputs, respectively. Deriving the conditional distribution $g_* | \mathbf{a}, a_*, \mathbf{y}$, the key predictive equations, which describe the distribution of the functions, are defined as:

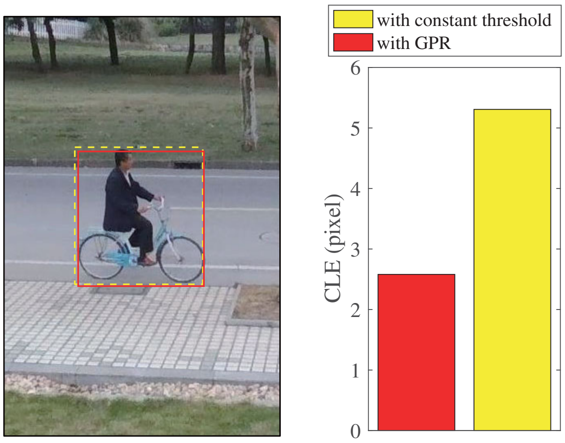
$$\bar{g}_* = \mathbf{k}(a_*, \mathbf{a})\big[\mathbf{K}(\mathbf{a}, \mathbf{a}) + {\sigma_n}^2\mathrm{I}\big]^{-1}\mathbf{y}, \tag{15}$$

$$\mathbb{V}(g_*) = \mathbf{K}(\mathbf{a}, \mathbf{a}) - \mathbf{k}(a_*, \mathbf{a})\big[\mathbf{K}(\mathbf{a}, \mathbf{a}) + {\sigma_n}^2\mathrm{I}\big]^{-1}\mathbf{k}(\mathbf{a}, a_*), \tag{16}$$

where $\bar{g}_*$ and $\mathbb{V}(g_*)$ are the mean and variance of the conditional distribution, respectively. Taking advantage of these parameters, a valid region of $SCCM$ is constructed. The upper limit of this region is $\bar{g}_* + 2\sqrt{\mathbb{V}(g_*)}$ and its lower limit is zero. On this basis, the update scheme of appearance model i at the frame t is defined as:

$$\tilde{\mathbf{x}}_i^t = \begin{cases} (1 - \alpha)\tilde{\mathbf{x}}_i^{t-1} + \alpha\mathbf{x}_i^t, & \text{if } SCCM_i^t \text{ is valid} \\ \tilde{\mathbf{x}}_i^{t-1}, & \text{else} \end{cases}, \tag{17}$$

where α is the learning rate that controls the update of the appearance model. When the calculated $SCCM_i^t$ is located in the valid region, $\tilde{\mathbf{x}}_i^t$ will be updated. Otherwise, it will not be changed.



(a) Tracking results. (b) CLE errors.

Fig. 5. Tracking comparison with GPR and constant threshold.

Remark 6: Compared to the usage of a constant threshold applied in [15], the presented adaptive threshold for updating the classifier is able to assist our tracker to achieve better tracking performance, as the center location error (CLE), which is the Euclidean distance between the ground-truth and estimated object centers, shown in Fig. 5.

IV. PERFORMANCE EVALUATION

To validate the performance of our presented tracker for UAV tracking applications, extensive UAV flight tests have been conducted with Parrot bebop 2 [1].

TABLE I
CHALLENGING FACTORS OF EACH IMAGE SEQUENCE.

Sequences	**OC**	**SV**	**IV**	**DE**	**IR**	**OR**	**CB**	**MV**	**AF**
Biker	√	√			√	√			√
BlueMan		√		√		√		√	
Car	√	√	√		√	√		√	√
Driver		√	√		√		√	√	
Logo		√	√		√	√	√	√	
OccMan1	√	√	√	√		√		√	√
OccMan2	√			√	√	√			
boat1		√			√		√	√	
boat2		√			√	√		√	
boat5			√		√	√	√	√	
building5			√		√	√	√	√	√
car10	√	√			√	√		√	√
car3	√	√				√	√	√	
car9		√	√		√	√		√	√
group1-1	√		√	√	√	√	√	√	√
person1				√		√	√	√	√

A. Evaluation Dataset and Challenging Factor

In this paper, 7 challenging tracking sequences from our flight tests are randomly selected for the validation. In addition, 9 challenging tracking sequences from the well-known UAV123 dataset [18] have been also chosen randomly

[1] http://www.parrot.com/

as the baseline to evaluate the presented tracking algorithm. The challenging factors of these 16 image sequences include partial or full occlusion (OC), scale variation (SV), illumination variation (IV), deformation (DE), in-plane rotation (IR), out-of-plane rotation (OR), cluttered background (CB), mechanical vibration (MV), and aggressive flight (AF), as shown in Tab. I.

B. Evaluation Criterion

For tracking performance evaluation criterions, one-pass evaluation (OPE) [19] with the CLE and success rate (SR) are employed. For the CLE, it has been introduced in the **Remark 6**. For the SR, it depends on success score (SS), which is defined as:

$$\mathrm{SS} = \frac{|ROI_E \cap ROI_{GT}|}{|ROI_E \cup ROI_{GT}|}, \tag{18}$$

where $|\ast|$ is the number of pixels in a region. $\cup$ and $\cap$ are the union and intersection operators. ROI_{GT} and ROI_E are the ground-truth and estimated regions of the tracking object. If the SS is larger than a threshold ϱ (ϱ=0.5 in this work) in one frame, the tracking on this frame is considered as a success. Therefore, the SR is defined as the ratio between the successful frame number and total image frame number.

C. Tracker for Comparison

Since our tracker is based on the CF, the most popular state-of-the-art CF-based trackers, i.e., KCF [10], BACF [14], and MCCT [20], have been employed to compare with our presented tracking algorithm, i.e, SPBACF. For all these well-known trackers, we have used the open source or binary programs provided by the authors with default parameters. It is noted that convolutional features from a convolutional neural network, e.g., VGG-Net [21], have been presented for object tracking. However, the tracker with convolutional features often requires high-cost computation. Since the UAV always has limited computing capability, it is difficult for UAV to achieve the real-time performance with convolutional features. Therefore, the HOG feature [22] is used in the MCCT tracker instead of convolutional features, i.e., MCCT-H. In addition, all visual trackers are initialized with the same parameters, e.g., initial object location and scale.

D. Setup of Our Tracker

In this work, we separate the holistic object into 5 different parts. Specifically, the bounding box is divided into 4 parts without overlaps, and the fifth one locates at the object center with the same size as other 4 parts, as shown in the Fig. 2. The parameters of background-aware CF tracker are the same as the parameters in the BACF tracker [14]. In the adaptive weighting step, we set the trade-off coefficient γ between the PSR and $SCCM$ as 10^{-4}. To achieve a balance between the performance and tracking speed, the number of particles is set to 300. The covariance matrix $[\sigma_x, \sigma_y, \sigma_{sr}, \sigma_{sc}, \sigma_\theta, \sigma_\phi]$ is [4,4,0,0.01,0,0]. In GPR model, we set the hyperparameters as following: the length of kernel l is equal to 0.5. The deviations of signal σ_f and noise σ_n are 0.05 and 0.001, respectively.

E. Evaluation Result and Discussion

Table II and III provide average CLE and SR for different trackers on all challenging tracking sequences. From the Tab. II, our tracker has achieved the best tracking performance among all trackers in terms of the average CLE. Although the average FPS of our tracker is ranking as *No.* 2, it is fast enough to close the control loop for UAV navigation. In addition, it is noted that the code of our tracker is not optimized in this work. From the Tab. III, our tracker has also obtained the best tracking performance in terms of the average SR. Therefore, we can find that the presented tracker performs favorably against KCF, BACF, MCCT-H trackers in

TABLE II

Center location error (CLE) (in pixels) and frames per second (FPS). Red and blue fonts indicate the best and second-best performances in all visual trackers.

Sequences	SPBACF	KCF	BACF	MCCT-H
Biker	3.46	37.19	4.12	5.99
BlueMan	3.10	60.52	1.93	2.98
Car	14.91	208.83	23.54	24.47
Driver	9.31	11.56	6.35	4.45
Logo	3.30	2.51	1.19	1.69
OccMan1	14.09	599.41	603.74	835.16
OccMan2	5.40	329.57	327.39	2.98
boat1	11.68	9.25	4.93	14.54
boat2	3.02	5.09	3.36	6.26
boat5	5.40	11.77	11.46	22.61
building5	3.98	24.72	3.28	2.94
car10	1.89	2.81	2.07	2.55
car3	1.88	2.56	1.62	2.03
car9	3.84	234.99	3.79	5.37
group1-1	8.39	26.90	68.55	4.70
person1	3.80	411.88	6.02	4.31
Average CLE	***6.09***	***123.72***	***67.08***	***58.94***
Average FPS	***23.29***	***161.14***	***22.08***	***7.17***

TABLE III

Success rate (SR) (%) (ϱ=0.5). Red and blue fonts indicate the best and second-best performances in all visual trackers.

Sequences	SPBACF	KCF	BACF	MCCT-H
Biker	100.00	29.15	100.00	51.28
BlueMan	100.00	15.33	100.00	100.00
Car	97.50	29.50	88.50	77.50
Driver	100.00	46.70	100.00	100.00
Logo	100.00	69.86	100.00	99.29
OccMan1	99.17	35.95	40.12	39.88
OccMan2	97.80	15.80	16.20	98.20
boat1	100.00	23.92	100.00	100.00
boat2	100.00	60.67	100.00	100.00
boat5	88.76	29.59	78.11	35.50
building5	100.00	89.44	100.00	100.00
car10	99.57	98.08	99.57	99.57
car3	100.00	81.50	100.00	100.00
car9	98.25	6.54	98.24	97.93
group1-1	97.53	67.87	19.33	97.08
person1	90.26	16.10	100.00	100.00
Average SR	***98.05***	***44.75***	***83.75***	***87.26***

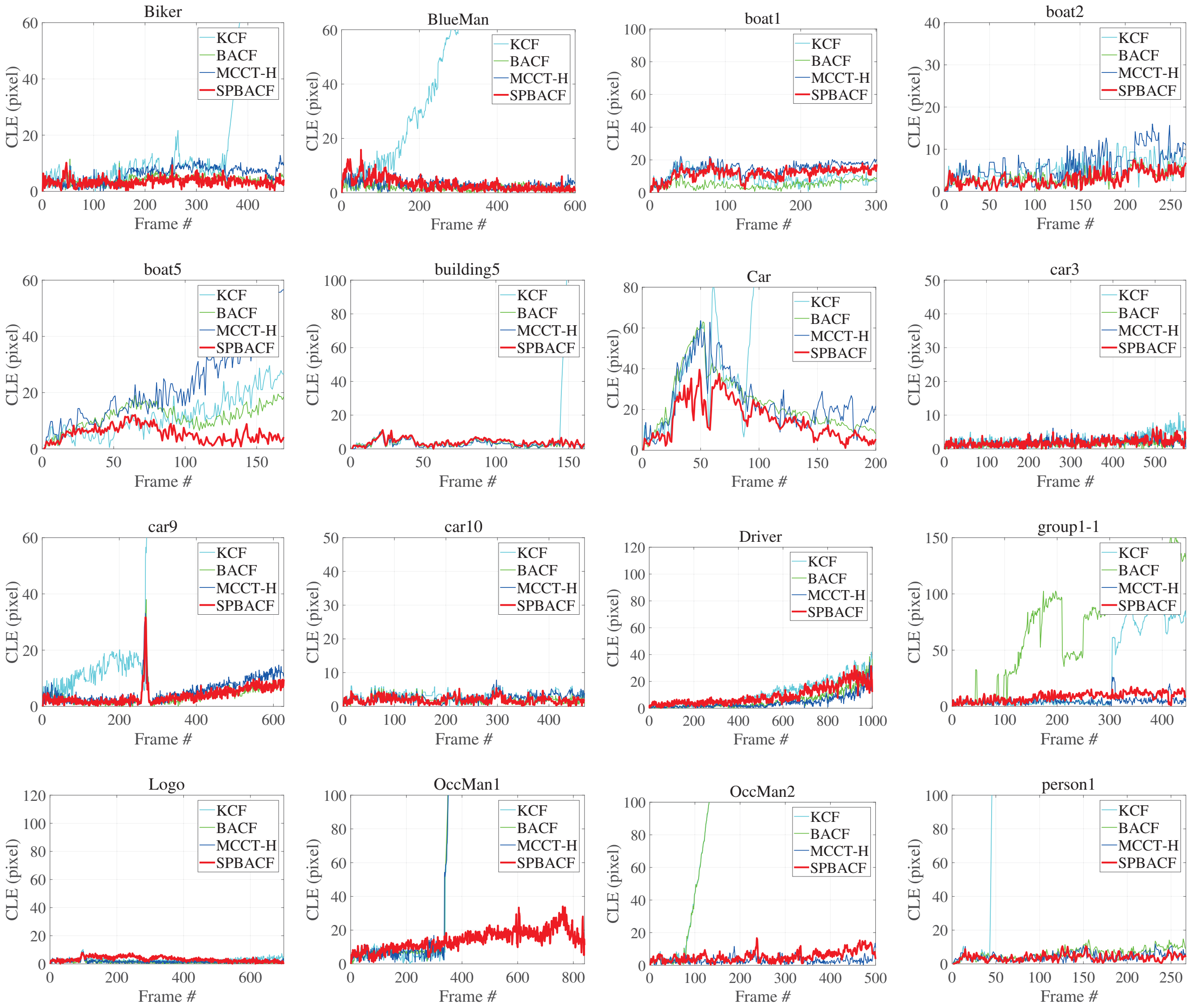


Fig. 6. CLE error evolution plots of all trackers in different challenging tracking sequences.

terms of efficiency, robustness, and accuracy.

Figure 6 shows the CLE error evolution plots of all trackers in different challenging tracking sequences. Compared to other 4 trackers, our tracker can maintain reasonable tracking performance without obvious drifts. Figure 7 provides some tracking examples with all trackers in different challenging image sequences. Especially, our tracker is more robust to the occlusion and scale variation compared to other 4 trackers.

V. CONCLUSIONS

In this paper, a novel robust real-time visual tracker has been presented, and applied for the UAV to track freewill 2D or 3D object. Specifically, the background-aware CF tracker is used to achieve better tracking performance compared to the classical CF tracker. An effective coarse-to-fine strategy with structure comparison and Bayesian inference framework is developed to improve the estimation of the tracking object location and scale changes. In addition, an adaptive threshold is established to update each local appearance model with a Gaussian process regression approach. The extensive UAV flight tests show that our presented visual tracker outperforms the most promising state-of-the-art visual trackers, and overcome the object appearance change caused by different challenging situations. We believe our approach will open the doors to their wider use in real-world UAV tracking tasks.

ACKNOWLEDGMENT

The work was supported by the State Key Laboratory of Robotics and System (HIT) (No. SKLRS-2018-KF-03) and the Fundamental Research Funds for the Central Universities (No.22120180009).

REFERENCES

[1] M. Mueller, G. Sharma, N. Smith, and B. Ghanem, “Persistent Aerial Tracking system for UAVs,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 1562–1569.

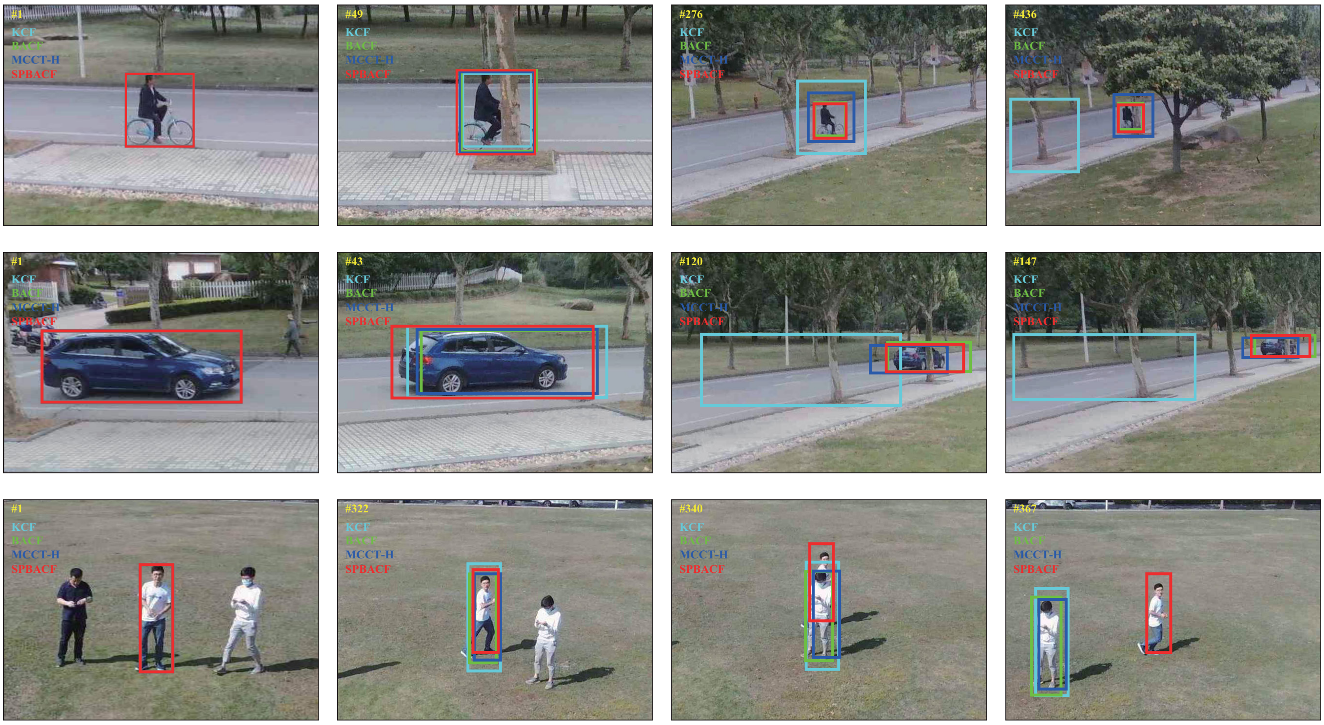


Fig. 7. UAV tracking examples. The first, second, third rows show the Biker, Car, OccMan1 image sequences. The source code and related tracking video can be checked at: `https://github.com/vision4robotics/SPBACF-Tracker` and `https://youtu.be/qm9StK3jqN4`, respectively.

[2] Y. Lin and S. Saripalli, “Sense and avoid for Unmanned Aerial Vehicles using ADS-B,” in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 6402–6407.

[3] M. A. Olivares-Mendez, C. Fu, P. Ludivig, T. F. Bissyande, S. Kannan, M. Zurad, A. Annaiyan, H. Voos, and P. Campoy, “Towards an Autonomous Vision-Based Unmanned Aerial System against Wildlife Poachers,” *Sensors*, vol. 15, no. 12, pp. 31 362–31 391, 2015.

[4] N. Imanberdiyev, C. Fu, E. Kayacan, and I. Chen, “Autonomous navigation of UAV by using real-time model-based reinforcement learning,” in *Proceedings of International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 2016, pp. 1–6.

[5] C. Martinez, T. Richardson, and P. Campoy, “Towards Autonomous Air-to-Air Refuelling for UAVs using visual information,” in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2013, pp. 5756–5762.

[6] C. Fu, A. Carrio, M. A. Olivares-Mendez, and P. Campoy, “Online learning-based robust visual tracking for autonomous landing of unmanned aerial vehicles,” in *Proceedings of International Conference on Unmanned Aircraft Systems (ICUAS)*, 2014, pp. 649–655.

[7] H. Lim and S. N. Sinha, “Monocular Localization of a moving person onboard a Quadrotor MAV,” in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 2182–2189.

[8] C. Fu, R. Suarez-Fernandez, M. A. Olivares-Mendez, and P. Campoy, “Real-time Adaptive Multi-Classifier Multi-Resolution Visual Tracking Framework for Unmanned Aerial Vehicles,” *IFAC Proceedings Volumes*, vol. 46, no. 30, pp. 99–106, 2013.

[9] H. Cheng, L. Lin, Z. Zheng, Y. Guan, and Z. Liu, “An autonomous vision-based target tracking system for rotorcraft unmanned aerial vehicles,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 1732–1738.

[10] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, “High-Speed Tracking with Kernelized Correlation Filters,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2015.

[11] R. Li, M. Pang, C. Zhao, G. Zhou, and L. Fang, “Monocular Long-Term Target Following on UAVs,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2016, pp. 29–37.

[12] Z. Qu, X. Lv, J. Liu, L. Jiang, L. Liang, and W. Xie, “Long-term reliable visual tracking with UAVs,” in *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2017, pp. 2000–2005.

[13] H. K. Galoogahi, T. Sim, and S. Lucey, “Correlation filters with limited boundaries,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 4630–4638.

[14] H. K. Galoogahi, A. Fagg, and S. Lucey, “Learning Background-Aware Correlation Filters for Visual Tracking,” in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 1144–1152.

[15] T. Liu, G. Wang, and Q. Yang, “Real-time part-based visual tracking via adaptive correlation filters,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 4902–4912.

[16] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers,” in *Foundations and Trends in Machine Learning*, vol. 3, 2010, pp. 1–122.

[17] J. Sherman and W. J. Morrison, “Adjustment of an Inverse Matrix Corresponding to a Change in One Element of a Given Matrix,” *Annals of Mathematical Statistics*, vol. 21, no. 1, pp. 124–127, 1950.

[18] N. S. M. Mueller and B. Ghanem, “A Benchmark and Simulator for UAV Tracking,” in *Proceedings of European Conference on Computer Vision (ECCV)*, 2016.

[19] Y. Wu, J. Lim, and M.-H. Yang, “Object Tracking Benchmark,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1834–1848, 2015.

[20] N. Wang, W. Zhou, Q. Tian, R. Hong, M. Wang, and H. Li, “Multi-cue correlation filters for robust visual tracking,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4844–4853.

[21] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” in *Proceedings of International Conference on Representation Learning (ICLR)*, 2015, pp. 1–14.

[22] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 2005, pp. 886–893.