Boundary Effect-Aware Visual Tracking for UAV with Online Enhanced Background Learning and Multi-Frame Consensus Verification

Changhong Fu[1,*], Ziyuan Huang[2], Yiming Li[1], Ran Duan[3], and Peng Lu[3]

***Abstract*— Due to implicitly introduced periodic shifting of limited searching area, visual object tracking using correlation filters often has to confront undesired boundary effect. As boundary effect severely degrade the quality of object model, it has made it a challenging task for unmanned aerial vehicles (UAV) to perform robust and accurate object following. Traditional hand-crafted features are also not precise and robust enough to describe the object in the viewing point of UAV. In this work, a novel tracker with online enhanced background learning is specifically proposed to tackle boundary effects. Real background samples are densely extracted to learn as well as update correlation filters. Spatial penalization is introduced to offset the noise introduced by exceedingly more background information so that a more accurate appearance model can be established. Meanwhile, convolutional features are extracted to provide a more comprehensive representation of the object. In order to mitigate changes of objects' appearances, multi-frame technique is applied to learn an ideal response map and verify the generated one in each frame. Exhaustive experiments were conducted on 100 challenging UAV image sequences and the proposed tracker has achieved state-of-the-art performance.**

I. INTRODUCTION

Visual object tracking plays a crucial role in unmanned aerial vehicle (UAV) applications such as obstacle avoidance [1], wild-life monitoring [2] and object following [3] (e.g. humans, cars, boats, etc.). Generally, UAV object tracking demands the tracker to keep track of an visual object detected by UAV. Due to online nature of learning as well as appearance changes of object, such as deformation, occlusion and illumination variation, object tracking remains a challenging task despite of recent significant progress. Especially in UAV scenarios, frequently occurrence of viewpoint change, fast movement and camera motion adds to its difficulty [4].

Correlation filter (CF) based framework has been successfully applied to tackle the aforementioned difficulties in recent years [5]–[7], due to its computational efficiency and sufficient tracking performance. It learns a correlation filter from the training samples online and calculates the correlation response in the frequency domain. The filter is then applied in the new frame to generate a response map. The object is located in this frame where the response value is the highest.

Learning CF efficiently requires, however, circular shifting operation on the training and detection samples. This generates periodic representation extension of these samples, which leads to undesired boundary effects since the search region is limited, see Fig. 1. Several approaches have been taken to mitigate boundary effects [5], [6]. Further, with limited training samples, learned models might be over-fitted and lack of negative patches will damage the accuracy of the tracker as well. In addition, many trackers based on CFs use hand-crafted features. Histogram of oriented gradients (HOG) is one of the most popular among those features [5]–[7]. Recently, convolutional-features-based trackers have demonstrated their competence in more precisely describing the object and thus achieving state-of-the-art results in visual tracking [8]–[10].

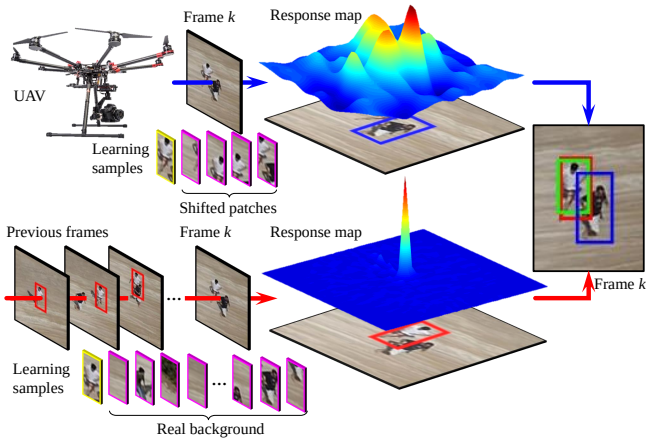

Fig. 1. Comparison between response maps of other CF based trackers and proposed tracker. Yellow samples are served as positive samples and pink ones are negative samples. Green bounding box in the right image is the ground truth. Red and blue boxes are from respectively the proposed tracker and other trackers. Real background are introduced in the proposed tracker as negative samples. Enhanced background learning and response map verification have notably suppressed the noise response in the background.

The information extracted from the response map has not received too much attention [11], [12]. However, it reveals to some extent the similarity between the model learned from previous consecutive frames and the newly detected target, which can be used to determine whether the object has differed drastically from the learned model and whether the detection result can be trusted.

This work focuses on solving boundary effect of CFs using enhanced background learning, achieving better representation of objects using convolutional features and estimate the consensus of tracking result depending on multi-frame consensus verification. A novel boundary effect-aware visual tracking approach is proposed, i.e. BEVT tracker.

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

[2]Ziyuan Huang is with the School of Automotive Studies, Tongji University, 201804 Shanghai, China tjhuangziyuan@gmail.com

[3]Ran Duan and Peng Lu are with the Adaptive Robotic Controls Lab (ArcLab), Hong Kong Polytechnic University (PolyU), Hong Kong, China peng.lu@polyu.edu.hk

Contributions of this work are listed as follows:

- A novel enhanced background learning approach is presented to learn background information and suppress the noise introduced by background at the same time. BEVT densely extracts background samples to enlarge search window, and a penalization to the object according to its spatial location is applied to suppress background noise. The effect can be seen in Fig. 1.
- A new feature extraction approach is applied. Different layers of CNN are exploited to provide both spatial and semantic information of objects, raising the preciseness of appearance models.
- A novel approach is utilized to raise the robustness of appearance model. A response map model learned from consecutive frames is compared with current-frame response map to generate a consensus score, which is used to influence the learning process so that unnecessary learning is avoided and necessary learning is enhanced.
- The tracker is evaluated on 100 challenging UAV image sequences and compared with other state-of-the-art trackers. Competitive accuracy is demonstrated in the experiments.

To the best of our knowledge, it is the first time that the presented BEVT tracker is designed in the literature and employed in UAV tracking.

II. RELATED WORKS

A. Tracking with correlation filter

CF framework has been widely applied in the field of object tracking. Its success depends on the implicit inclusion of all periodically shifted learning sample and the exceeding computational efficiency. Many trackers use CF framework such as minimum output sum of squared error (MOSSE), kernelized correlation filters [7] and multiple other trackers [5], [6], [13], [14]. However, correlation filter has a natural defect of boundary effect. Efforts have been made to investigate the problem. Spatially regularized DCF (SRDCF) utilizes spatial information to expand search regions [6], and learning background-aware CF (BACF) use negative patches from background for the correlation filter to learn [5]. However, with background introduced, more irrelevant information is also brought into the search window.

B. Tracking by convolutional feature

Deep convolutional networks are receiving more attention in recent years. Some trackers directly exploit the idea of deep architecture and built upon it [11], while others extract CNN feature within CF framework [6], [9]. Since object representation is crucial in object tracking of UAV due to its requirement of online training, the application of CNN features is rapidly expanding in this field due to its comprehensive description. Both single-layer [6] and multi-layer [9] features are investigated and applied in the CF framework.

C. Verification by response map

Response map is the result generated by CF tracker in each frame and objects are believed to be where its value is the highest. Therefore, response map reveals certain information about the object to be tracked in this frame. Attentional correlation filter network for adaptive visual tracking (ACFN) uses response map to select suitable tracking module [11]. Large margin object tracking with circulant feature maps (LMCF) utilizes maximum response value and the average peak to correlation energy (APCE) to measure the confidence of each tracking result [12].

III. PROPOSED TRACKING APPROACH

In this work, a novel correlation filter with enhanced background learning and multi-frame consensus verification is proposed. Its main structure is shown in Fig. 2.

A. Overall objective

The overall objective of our tracker is to minimize the loss function with enhanced background learning, which can be expressed as the following equation:

$$\mathcal{E}(\mathbf{w}) = \frac{1}{2}\|\mathbf{y} - \sum_{d=1}^{D} \mathbf{B}\mathbf{x}^d \star \mathbf{w}^d\|_2^2 + \sum_{d=1}^{D} \|\mathbf{p}\mathbf{w}^d\|_2^2 \ , \quad (1)$$

where $\mathbf{y} \in \mathbb{R}^N$ denotes the desired correlation response, $\mathbf{x}^d \in \mathbb{R}^N (d = 1, 2, ..., D)$ denotes the dth channel of vectorized input image and $\mathbf{w}^d \in \mathbb{R}^M$ denotes the dth channel of the correlation filter. $\mathbf{B} \in \mathbb{R}^{M \times N}$ is a binary matrix, adopted to crop the middle M elements of signal $\mathbf{x}^d$. Moreover, to further mitigate the boundary effect, a penalization matrix $\mathbf{p} \in \mathbb{R}^M$ is introduced to emphasize the central part of the extracted feature.

Remark 1: The enhanced background learning not only can mitigate boundary effect by employing cropping matrix $\mathbf{B}$ to extract background samples, noise introduced by background is also suppressed in the proposed approach by spatial penalization $\mathbf{p}$. In addition, the feature vector $\mathbf{x}$ is extracted from VGG-Net [15] using conv3-4, conv4-4 and conv5-4 combined together, since early layers provide spatial details and deep layers contains semantic information to prevent deformations. Details are shown in the middle of Fig. 2.

The aforementioned objective can be equally expressed as the following equation:

$$\mathcal{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\|\mathbf{y}(n) - \sum_{d=1}^{D} \mathbf{w}^{d\top}\mathbf{B}\mathbf{x}^d[\Delta\tau_n]\|_2^2 + \sum_{d=1}^{D}\|\mathbf{p}\mathbf{w}^d\|_2^2 \ , \quad (2)$$

where $\mathbf{y}(n)$ is the ideal response and $\mathbf{B}\mathbf{x}^d[\Delta\tau_n]$ is the cropped input image sample. $[\Delta\tau_n]$ is the circular shift operator, and $\mathbf{B}\mathbf{x}^d[\Delta\tau_n]$ is the cropped input image sample that is shifted by n elements.The superscript $\top$ denotes the conjugate transpose of a complex vector or matrix.

All the summations can be expressed in matrix form, which gives,

$$\mathcal{E}(\mathbf{w}) = \frac{1}{2}\|\mathbf{y} - \mathbf{X}\left(\mathbf{I}_D \otimes \mathbf{B}^\top\right)\mathbf{w}\|_2^2 + \|\tilde{\mathbf{p}}\mathbf{w}\|_2^2 \ , \quad (3)$$

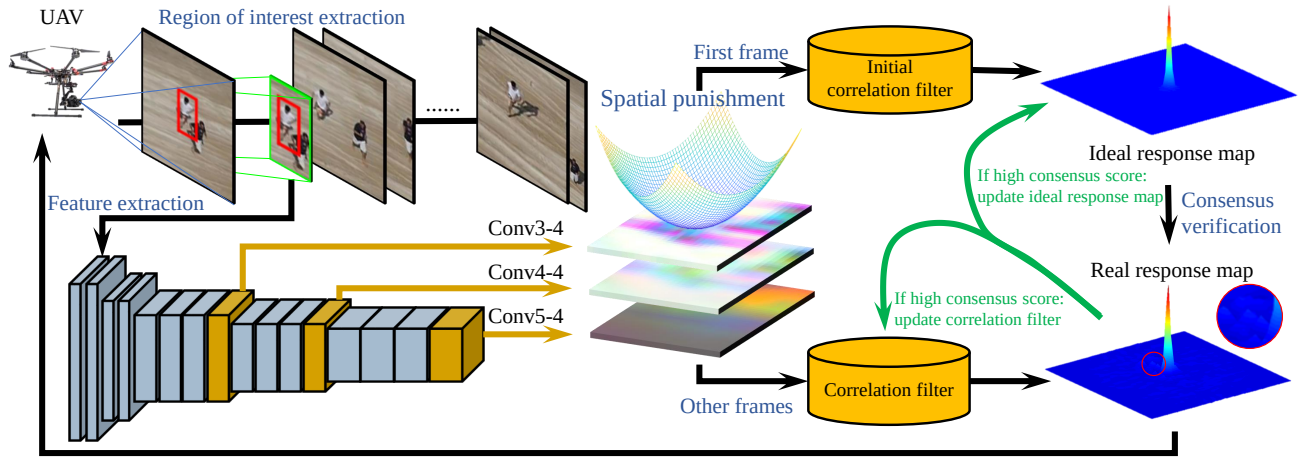

Fig. 2. Main structure of the proposed BEVT tracker. Because of enhanced background learning, background noise is further suppressed. Therefore, details in the real response map is zoomed for better clarity.

where $\mathbf{y} = \left[\mathbf{y}(1)\cdots\mathbf{y}(N)\right]^\top$, $\mathbf{w} = \left[\mathbf{w}_1^\top\cdots\mathbf{w}_D^\top\right]^\top$, $\mathbf{X} = \left[\mathbf{x}[\Delta\tau_1]^\top\cdots\mathbf{x}[\Delta\tau_N]^\top\right]^\top$, and $\tilde{\mathbf{p}} = \left[\mathbf{p}^\top\ \mathbf{p}^\top\ \cdots\ \mathbf{p}^\top\right]^\top$. $\mathbf{I}_D \in \mathbb{R}^{D\times D}$ is an identity matrix and $\otimes$ is the operator of Kronecker product.

Remark 2: Simplification process can be seen in the Appendix.

B. Transfer into frequency domain

To speed up the calculation, the correlation operations are normally carried out in the frequency domain. Therefore, (3) can be transferred into frequency domain to be the following equation:

$$\begin{aligned} &\hat{\mathcal{E}}(\mathbf{w},\hat{\mathbf{g}}) = \frac{1}{2}\|\hat{\mathbf{y}} - \hat{\mathbf{X}}\hat{\mathbf{g}}\|_2^2 + \|\tilde{\mathbf{p}}\mathbf{w}\|_2^2 \\ &s.t. \quad \hat{\mathbf{g}} = \sqrt{N}(\mathbf{I}_D \otimes \mathbf{F}\mathbf{B}^\top)\mathbf{w} \end{aligned}, \tag{4}$$

where $\hat{\mathbf{g}} \in \mathbb{C}^{DN\times 1}$ is introduced in preparation for the further optimization operations. The superscript $\hat{}$ indicates the discrete Fourier transform of a signal, i.e., $\hat{\boldsymbol{\alpha}} = \sqrt{N}F\boldsymbol{\alpha}$. Therefore, $\hat{\mathbf{y}}$, $\hat{\mathbf{X}} = [diag(\hat{\mathbf{x}}^1)^\top, \cdots, diag(\hat{\mathbf{x}}^D)^\top]$ and $\hat{\mathbf{g}}$ is respectively the Fourier form of $\mathbf{y}$, $\mathbf{X}$ and $\mathbf{g} = \left(\mathbf{I}_D \otimes \mathbf{B}^\top\right)\mathbf{w}$ in (3).

C. Optimization operations

To fully exploit the convexity of (4), we take advantage of Alternative direction method of multipliers (ADMM) [16] to calculate the optimal solution to the equation. First, we transform (4) to Augmented Lagrangian format as the following equation:

$$\begin{aligned} \mathcal{L}(\mathbf{w},\hat{\mathbf{g}},\hat{\zeta}) =& \frac{1}{2}\|\hat{\mathbf{y}} - \hat{\mathbf{X}}\hat{\mathbf{g}}\|_2^2 + \|\tilde{\mathbf{p}}\mathbf{w}\|_2^2 \\ &+ \hat{\zeta}^\top(\hat{\mathbf{g}} - \sqrt{N}\left(\mathbf{I}_D \otimes \mathbf{F}\mathbf{B}^\top\right)\mathbf{w}) \\ &+ \frac{\mu}{2}\|\hat{\mathbf{g}} - \sqrt{N}\left(\mathbf{I}_D \otimes \mathbf{F}\mathbf{B}^\top\right)\mathbf{w}\|^2 \end{aligned}, \tag{5}$$

where the penalty factor is μ and the Lagrangian vector in the Fourier domain $\hat{\boldsymbol{\zeta}} \in \mathbb{C}^{DN\times 1}$ is defined as $\hat{\boldsymbol{\zeta}} = [\hat{\boldsymbol{\zeta}}^{1\top}, \cdots, \hat{\boldsymbol{\zeta}}^{D\top}]^\top$. By employing ADMM, the equation should be divided into two subproblems, respectively the following $\hat{\mathbf{g}}^*$ and $\mathbf{w}^*$. Then (5) can be iteratively solved by solving these two subproblems alternatively.

1) Subproblem $\mathbf{w}^*$:

$$\begin{aligned} \mathbf{w}^* =& \arg\min_{\mathbf{w}} \Big\{ \|\tilde{\mathbf{p}}\mathbf{w}\|^2 + \hat{\zeta}^\top(\hat{\mathbf{g}} - \sqrt{N}\left(\mathbf{I}_D \otimes \mathbf{F}\mathbf{B}^\top\right) \\ &+ \frac{\mu}{2}\|\hat{\mathbf{g}} - \sqrt{N}\left(\mathbf{I}_D \otimes \mathbf{F}\mathbf{B}^\top\right)\mathbf{w}\|_2^2 \Big\} \\ =& \left(\frac{2\tilde{\mathbf{p}}^\top\tilde{\mathbf{p}}}{N} + \mu\right)^{-1}(\zeta + \mu\mathbf{g}) \end{aligned}, \tag{6}$$

where $\mathbf{g}$ and ζ can be calculated from $\hat{\mathbf{g}}$ and $\hat{\zeta}$ respectively through the following Inverse Fast Fourier Transform (IFFT) equations:

$$\begin{cases} \mathbf{g} = & \frac{1}{\sqrt{N}}\left(\mathbf{I}_D \otimes \mathbf{F}\mathbf{B}^\top\right)\hat{\mathbf{g}} \\ \zeta = & \frac{1}{\sqrt{N}}\left(\mathbf{I}_D \otimes \mathbf{F}\mathbf{B}^\top\right)\hat{\zeta} \end{cases}. \tag{7}$$

Remark 3: Note that inside the first pair of brackets in the calculation result of (6), $\tilde{\mathbf{p}}^\top\tilde{\mathbf{p}}$ and N are constant, and parameter μ is preset for each iteration before tracking, so the first item of $\mathbf{w}^*$ can be pre-calculated before running the algorithm so that the inverse operation won't affect the efficiency.

2) Subproblem $\mathbf{g}^*$:

$$\begin{aligned} \hat{\mathbf{g}}^* =& \arg\min_{\mathbf{g}} \Big\{ \frac{1}{2}\|\hat{\mathbf{y}} - \hat{\mathbf{X}}\hat{\mathbf{g}}\|^2 \\ &+ \hat{\zeta}^\top\left(\hat{\mathbf{g}} - \sqrt{N}\left(\mathbf{I}_D \otimes \mathbf{F}\mathbf{B}^\top\right)\right) \\ &+ \frac{\mu}{2}\|\hat{\mathbf{g}} - \sqrt{N}\left(\mathbf{I}_D \otimes \mathbf{F}\mathbf{B}^\top\right)\mathbf{w}\|_2^2 \Big\} \end{aligned}. \tag{8}$$

Calculating (8) directly would be time-consuming but fortunately, $\hat{\mathbf{X}}$ is sparse banded, which means the computation can be accelerated through separately calculating each element of $\hat{\mathbf{y}}$, i.e. $\hat{\mathbf{y}}(n), n = 1, 2, ..., N$ because each $\hat{\mathbf{y}}$ relies solely on $\hat{\mathbf{x}}(n) = \left[\hat{\mathbf{x}}^1(n), \hat{\mathbf{x}}^2(n), ..., \hat{\mathbf{x}}^D(n)\right]^\top$ and $\hat{\mathbf{g}}(n) =$

$\left[\text{conj}\left(\hat{\mathbf{g}}^1(n)\right), \ldots, \text{conj}\left(\hat{\mathbf{g}}^D(n)\right)\right]^\top$, where conj(.) indicates the complex conjugate operation. Hence, subproblem $\mathbf{g}^*$ can be divided as N independent sub-subproblems, which are to be solved over $n = [1, 2, ...N]$:

$$\begin{aligned}\hat{\mathbf{g}}(n)^* = \arg\min_{\mathbf{g}} \Big\{ & \frac{1}{2}\|\hat{\mathbf{y}}(n) - \hat{\mathbf{x}}(n)^\top \hat{\mathbf{g}}(n)\|_2^2 \\ & + \hat{\zeta}(n)^\top \left(\hat{\mathbf{g}}(n) - \hat{\mathbf{w}}(n)\right) \\ & + \frac{\mu}{2}\|\hat{\mathbf{g}}(n) - \hat{\mathbf{w}}(n)\|^2 \Big\}\end{aligned} , \tag{9}$$

where $\hat{\mathbf{w}}(n) = \left[\hat{\mathbf{w}}^1(n), \ldots, \hat{\mathbf{w}}^D(n)\right]$ and $\hat{\mathbf{w}}^d = \sqrt{D}\mathbf{F}\mathbf{P}^\top \mathbf{w}^d$. The solution to each sub-subproblem can be expressed as:

$$\begin{aligned}\hat{\mathbf{g}}(n)^* = & \left(\hat{\mathbf{x}}^\top(n)\hat{\mathbf{x}}(n) + \mu \mathbf{I}_D\right)^{-1} \\ & \left(\hat{\mathbf{x}}^\top(n)\hat{\mathbf{y}}(n) - \hat{\zeta}(n) + \mu\hat{\mathbf{w}}(n)\right)\end{aligned} . \tag{10}$$

Unfortunately, the inverse operation here will take a huge amount of time to be carried out. In order to improve the efficiency, the Sherman-Morrison formula [17] is applied, which is $(\mathbf{A} + \mathbf{u}\mathbf{v}^\top)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{u}(\mathbf{I}_k + \mathbf{v}^\top \mathbf{A}^{-1}\mathbf{u})^{-1}\mathbf{v}^\top \mathbf{A}^{-1}$ in generalization form, where matrix u is an $n \times k$ matrix, v is an $k \times n$ matrix and A is an $n \times n$ matrix. In our case, $A = \mu \mathbf{I}_D$ and $\mathbf{u} = \mathbf{v} = \hat{\mathbf{x}}(n)$. Therefore, (10) can be equally expressed as:

$$\begin{aligned}\hat{\mathbf{g}}(n)^* = & \frac{1}{\mu}(\hat{\mathbf{y}}(n)\hat{\mathbf{x}}(n) - \hat{\zeta}(n) + \mu\hat{\mathbf{w}}(n)) \\ & - \frac{\hat{\mathbf{x}_0}(n)}{\mu b}\left(\hat{s}_{\mathbf{x}}(n)\hat{y}(n) - \hat{s}_\zeta(n) + \mu\hat{s}_{\mathbf{w}}(n)\right)\end{aligned} , \tag{11}$$

where $\hat{s}_{\mathbf{x}}(n) = \hat{\mathbf{x}}(n)^\top \hat{\mathbf{x}}(n)$, $\hat{s}_\zeta(n) = \hat{\mathbf{x}}(n)^\top \hat{\zeta}$, $\hat{s}_{\mathbf{w}}(n) = \hat{\mathbf{x}}(n)^\top \hat{\mathbf{w}}$ and $b = \hat{\mathbf{x}}(n)^\top \hat{\mathbf{x}}(n) + \mu$.

Lagrangian parameter is updated as follows:

$$\hat{\zeta}_{j+1} = \hat{\zeta}_j + \mu\left(\hat{\mathbf{g}}^*_{j+1} - \hat{\mathbf{w}}^*_{j+1}\right) , \tag{12}$$

where subscript j denotes the initial value or the value in the last iteration and subscript $j+1$ denotes the value in current iteration. So practically, we use the last(or initial) value of ζ and the current solution to the aforementioned subproblems $\hat{\mathbf{g}}^*$ and $\hat{\mathbf{w}}^*$ to update the Lagrangian parameter. Note that $\hat{\mathbf{w}}^*_{j+1} = \left(\mathbf{I}_D \otimes \mathbf{F}\mathbf{B}^\top\right)\mathbf{w}^*_{j+1}$.

D. Model update

Most currently existing trackers update appearance models and CFs on a frame-by-frame basis, ignoring the possibility that tracking result at current frame can be inaccurate. Blindly updating appearance model and correlation filters would cause the tracker to learn wrongly detected objects. Since response map reveals similarity between object and model, it is used in our work to estimate consensus so that sudden changes will not be learned.

In the first frame when UAV has detected an object, a new model is generated and an initial correlation filter is learned in this frame. The correlation filter learned is then directly used to convolute with the detected object to generate an ideal response map. In the other frames, the following criterion are used to verify the consensus score $\mathbf{C}$ of that frame:

$$\mathbf{C} = \mathbf{e}^{-\|\mathbf{M}_{\text{ideal}} - \mathbf{M}_{\text{curr}}\|_2^2} , \tag{13}$$

where $\mathbf{M}_{ideal}$ is the ideal response map and $\mathbf{M}_{curr}$ is the response map generated in the current frame.

Remark 4: Verification process in each frame (except the first) generates a consensus score and control the learning of both object appearance model and response map model. The process can be seen in Fig. 3.

When consensus is high enough, appearance model $\hat{\mathbf{x}}$ and ideal response map $\mathbf{M}_{ideal}$ are updated to improve its robustness to pose, scale and illumination changes with learning rates respectively being η and γ as follows:

$$\begin{aligned}\hat{\mathbf{x}}^i &= (1-\eta)\hat{\mathbf{x}}^{i-1} + \eta\hat{\mathbf{x}}_{\text{curr}} \\ \mathbf{M}^i_{\text{ideal}} &= (1-\gamma)\mathbf{M}^{i-1}_{\text{ideal}} + \gamma\mathbf{M}_{\text{curr}}\end{aligned} , \tag{14}$$

where superscript i denotes current frame and $i-1$ denotes the model learned in the last learning. Subscript $curr$ denotes the feature extracted or response map generated in the current frame. Note that we use $\hat{\mathbf{x}}$ updated here to compute parameters in section III-C.

Remark 5: Compared to other CF-based trackers, which only learns object appearance model, ideal response map is also learned and updated. In addition, when the estimated consensus score is considered high enough, a reinforcement technique is used to reinforce the learning of current model, that is, the learning rate in this frame will be temporally boosted. Details can be seen in Algorithm 1.

E. Detection

Object location in the frame is detected by applying filter updated in the last allowed learning phase (controlled by consensus score) to the newly extracted convolutional features

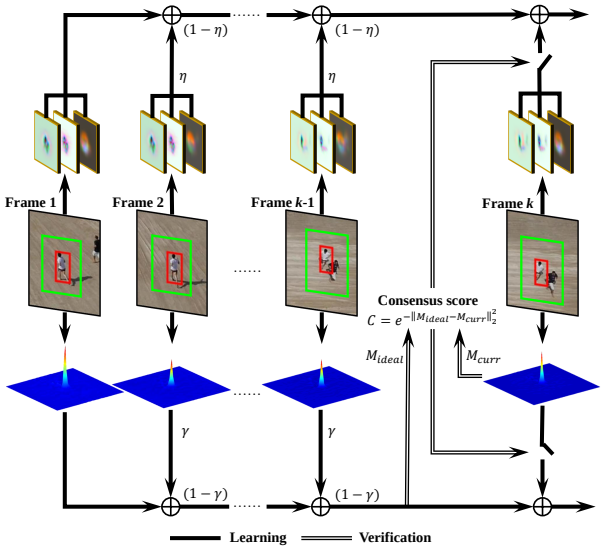

Fig. 3. Schematic diagram of verification in model update process. Top path is for learning of object appearance model (features extracted from three layers of CNN). Bottom path is for learning of response map.

on different resolutions of the searching area to estimate scale differences likewise to [6]. Interpolation strategy is applied to maximize detection scores in each correlation result. The object's scale is estimated by the scale with maximum correlation score and location is then estimated according to this scale.

F. Module evaluation

In order to verify the effectiveness of the presented modules, the proposed tracker with different modules are tested on *person2_2* image sequence.

Two evaluation criteria are employed to evaluate the performance of the proposed tracker, respectively center location error (CLE) and success rate (SR) based on one-pass evaluation (OPE).

CLE is defined as average Euclidean distance between the centers of detected bounding box of tracked objects and the manually annotated ground truth. SR shows the percentage of cases when the overlap area between annotated bounding boxes and detected ones are greater than a certain threshold.

As is shown in Fig. 4, every module added to the original framework would result in satisfying improvements in the CLE and SR.

In Section IV, more exhaustive experiments were conducted to evaluate the performance of the proposed BEVT tracker.

IV. EXPERIMENTS

The proposed BEVT tracker is evaluated extensively and thoroughly on 100 challenging UAV image sequences from well-known UAV123_10fps [4], which is specifically designed and annotated for UAV tracking including exhaustive scenarios such as person, vehicle and boat following as well as UAV following. In this section, evaluation criteria is explained and comparisons are made between the state-of-the-art trackers. Some of the examples of UAV tracking results are provided in Fig. 5.

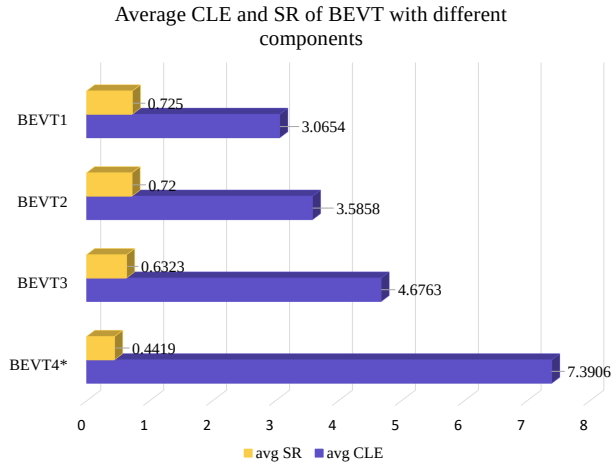

Fig. 4. Comparison between BEVT with different modules on *person2_2* image sequence. BEVT1 is the full version, i.e. BEVT. BEVT2 is BEVT without response map verification. BEVT3 is BEVT2 without enhanced background learning. BEVT4 is BEVT3 using HOG feature instead of convolutional features. Note: because BEVT4 has lost the object from frame 180, so only the result of former 180 frames is shown.

Algorithm 1: BEVT tracker

Input: Object location on frame $k-1$, Learned response model $\mathbf{M}_{\text{ideal}}^{k-1}$

Output: Estimated location on frame k

for $k = 2$ *to end* **do**
- Extract the search window in frame i centered at object location on frame $k-1$
- Represent the extracted search window using convolutional features $\hat{\mathbf{x}}_{\text{detect}}^{k}$
- Convolute learned CF $\hat{\mathbf{g}}_{k-1}$ with $\hat{\mathbf{x}}_{\text{detect}}^{k}$ on different scales to generate $\mathbf{M}_{\text{curr}}$ and detect position
- **if** *consensus score* $C > threshold_{high}$ **then**
  - η = high_learning_rate
- **else**
  - **if** *consensus score* $C > threshold_{low}$ **then**
    - η = low_learning_rate
  - **else**
    - Start detection of next frame $k+1$
  - **end**
- **end**
- Extract convolutional features where $\mathbf{M}_{\text{curr}}$ is highest
- Update object and response map model by (14)
- Learn CF $\hat{\mathbf{g}}_k$ by (6), (11) and (5)

end

A. Evaluation criteria

Besides aforementioned CLE and SR, two plots based on two criteria are employed to visualize the evaluation result. Precision plot (PP) is adopted to demonstrate percentage of frames whose CLE is within a certain threshold. Conventionally, threshold of 20 pixels is chosen to give the overall precision score [4]. Success plot (SP) demonstrates the percentage of frames whose overlap score (OS) is within a certain threshold and area under curve (AUC) is adopted to give overall overlap score [4].

Remark 6: The experiments were carried out strictly according to standard procedure and standard evaluation criteria in the visual object tracking field.

B. Comparison with state-of-the-art trackers

In this section, the tracking performances of the proposed tracker is demonstrated based on the aforementioned criteria. In oder to achieve a comprehensive evaluation, the BEVT tracker is horizontally compared with 20 state-of-the-art trackers, i.e. HCF [9], BACF [5], KCF [7], StapleCA [18], Staple [19], Struck [20], ASLA [21], CSK [22], IVT [23], PTAV [24], TLD [25], MUSTER [26], MCCT [27], MCCT-H [27], DCF [7], STRCF [28], SRDCF [6], MEEM [29], SAMF [30], DSST [14], on 100 challenging UAV image sequences

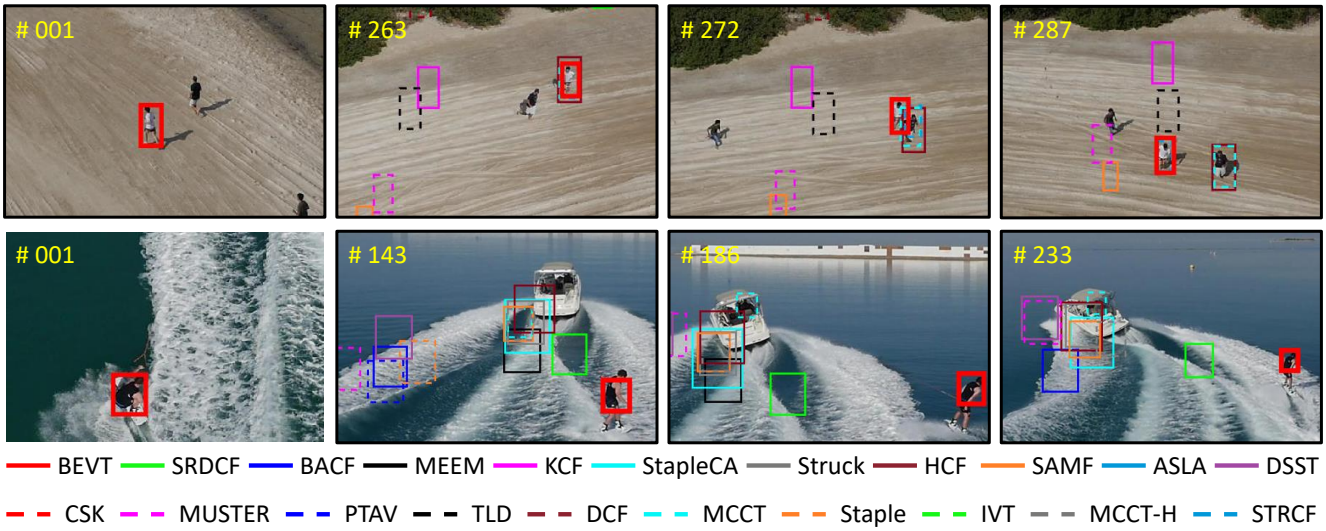

Fig. 5. Examples of UAV tracking results. The first row is *waterboard4* and the second is *group2_2* image sequences. Tracking code and video can be found here: http://github.com/vision4robotics/BEVT-tracker and https://youtu.be/wjMQmx1_qkw

captured from a low-altitude aerial perspective [4], which are nearly identical to the real-world UAV following cases. Fig. 6 and Fig. 7 shows all the results of PPs and AUCs of 21 trackers on 100 challenging UAV image sequences. In terms of precision and success ratio, the proposed BEVT tracker has demonstrated a superiority in overall performance.

Remark 7: All trackers are implemented with MATLAB R2017a and all the experiments were run on the computer with an i7-8700K processor (3.7GHz), 48GB RAM and NVIDIA Quadro P2000 GPU.

Besides overall performance, the proposed tracker and 20 other top trackers also compare on 12 attributes categorized by 100 challenging UAV image sequences, i.e. aspect ratio change (ARC), background clutter (BC), camera motion (CM), fast motion (FM), full occlusion (FOC), illumination variation (IV), low resolution (LR), out-of-view (OV), partial occlusion (POC), scale variation (SV), similar object (SOB) and viewpoint change (VC) [4]. Table I and II show the scores in CLE and AUC evaluation on mentioned 12 attributes. The proposed tracker has achieved the best performance in both CLE and AUC evaluation in all aspects but SOB, in which BEVT both achieved second place. Two violin plots Fig. 8 and Fig. 9 visualizes the results of PP and AUC of all evaluated trackers respectively.

Remark 8: Among all the attributes that our tracker performs satisfactory, in the evaluation attribute CM, FM, LR and VC, the proposed BEVT tracker outperforms the other tracker substantially. The superiority of performance of BEVT tracker in CM, FM and VC probably results

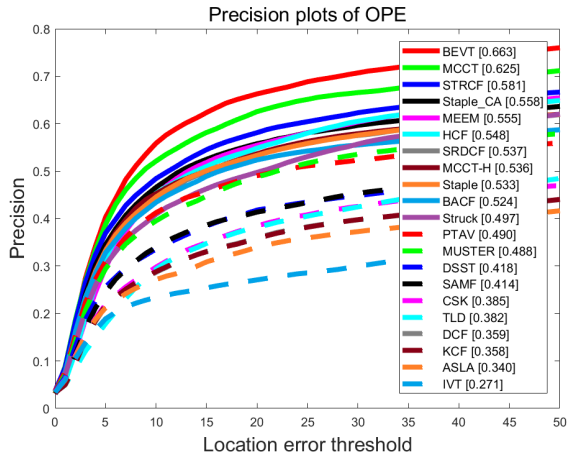

Fig. 6. Precision plots of 21 trackers on 100 challenging UAV image sequences. The BEVT tracker has a superiority of respectively 3.8% and 8.2% in comparison with the second best tracker MCCT (2018 CVPR) and third best tracker STRCF (2018 CVPR).

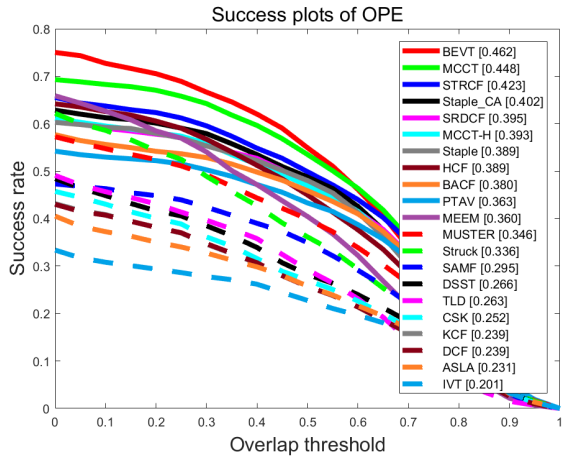

Fig. 7. AUCs of 21 trackers on 100 challenging UAV image sequences. The BEVT tracker has a superiority of respectively 1.4% and 5.0% in comparison with the second best tracker MCCT (2018 CVPR) and third best tracker STRCF (2018 CVPR).

TABLE I

SCORES OF PP (THRESHOLD AT 20 PIXELS). RED AND BLUE FONTS RESPECTIVELY INDICATES BEVT'S BEST AND SECOND BEST PERFORMANCES AMONG ALL TRACKERS.

	ARC	BC	CM	FM	FOC	IV	LR	OV	POC	SV	SOB	VC
BEVT	59.5	54.4	67.3	56.7	49.1	58.7	55.2	56.1	59.9	62.5	62.4	60.5
Best of other trackers	55.3	51.3	60.7	48.6	45.3	55.7	46.7	54	58.2	58.4	65.3	54.4
Avg. of other trackers	39.565	34.68	41.33	29.465	34.915	37.035	36.33	36.515	42.15	43.43	50.45	38.555

TABLE II

SCORES OF SR (AUC). RED AND BLUE FONTS RESPECTIVELY INDICATES BEVT'S BEST AND SECOND BEST PERFORMANCES AMONG ALL TRACKERS.

	ARC	BC	CM	FM	FOC	IV	LR	OV	POC	SV	SOB	VC
BEVT	39.5	34.3	46.9	34.2	25	40.5	28.4	39.3	40.4	43	44.1	42
Best of other trackers	37.8	33.4	44.6	31.5	22.8	38.8	25.3	38.9	39.7	41.1	45.5	39.9
Avg. of other trackers	26.875	22.235	29.775	18.275	17.025	26.055	17.94	26.33	28.255	29.97	34.16	28.06

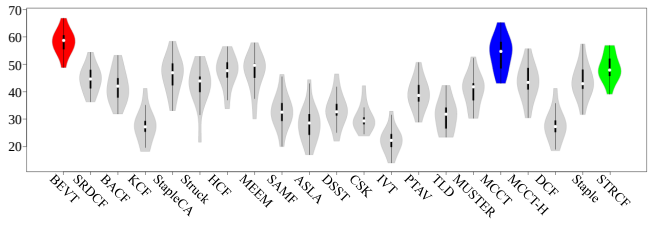

Fig. 8. 12 attributes based on PP results of 21 trackers. Red, blue and green fonts respectively indicates the best, second best and third best performances among all trackers.

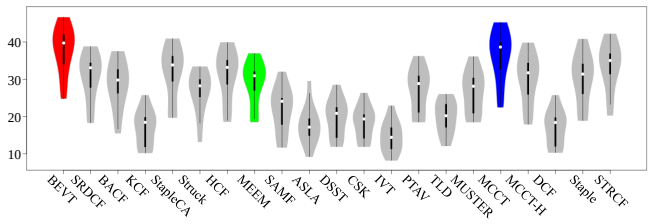

Fig. 9. 12 attributes based on AUC results of 21 trackers. Red, blue and green fonts respectively indicates the best, second best and third best performances among all trackers.

from the large search area combined with strong background noise suppressing ability since all these scenarios include large-scale motion of tracked objects. Due to the ability of consensus verification module to verify tracking result and interfere with learning process, BEVT tracker is able to learn a more robust and accurate object appearance model even when there are few information and many noises in LR scenarios. All these scenarios appear extremely frequently. Our lead in FM, CM and VC demonstrates the trackers ability to follow the object in higher relative speed and the lead in LR means the proposed tracker can follow smaller objects (farther) or in bad weather. It can be thus concluded that our tracker are more suitable than other trackers for UAV to perform object tracking at higher speed and in harsher environments.

C. Limitations

Background clutter performance: Due to the inclusion of more real background information, similar objects are more likely to be included in the detection phase. Therefore, the BEVT tracker's performance is limited when there is similar objects in the background, especially in cases when the target is occluded, as is shown in Table I and Table II.

Spatial penalization disturbance: What's more, in order to simplify calculation, matrix multiplication was used instead of element-wise multiplication of penalization term $\mathbf{p}$ and correlation filter $\mathbf{w}$ to enforce spatial penalization. This would introduce unwanted disturbance in spatial punishment. Therefore, a bowl-shaped penalization term might not be the best penalization term. A more suitable penalization matrix can be more carefully selected in the future.

Speed: In addition, the BEVT tracker is implemented on MATLAB platform with no optimization. Therefore, the frame per second achieves only on average 0.653 on the 100 challenging UAV image sequences on the platform mentioned before. Fortunately, the used DJI S1000+ is capable of carrying high-performance GPU and CPU. With proper optimization for GPU and implementation of parallel computation, BEVT tracker can be applied to UAV tracking.

V. CONCLUSIONS

In this work, tracker with online enhanced background learning and multi-frame consensus verification (response map verification) is proposed. Enhanced background learning solves the boundary effect of BEVT tracker and reduces the noise introduced by background. Convolutional features improve the description of object by making it more comprehensive with spatial details and semantic information. The introduced consensus verification further adds to the robustness of learned model. The BEVT tracker uses the ADMM to optimize the calculation of correlation filter. After performing exhaustive experiments and comparisons on 100 challenging UAV image sequences, it is proven that despite our limitation in speed, the best tracking performance is achieved among all state-of-the-art trackers. Addressing boundary effect and strengthening representation of models enables UAV to track more complex objects in complicated environments. In our perspective, the results of this work further improves the CF framework in terms of limitation in boundary effect and extend the implementation of response map, thus promoting object tracking for UAV scenarios.

APPENDIX

Here is the deduction from (2) to (3). The summation term in the first norm in (2) can be simplified as follows:

$$\begin{aligned}
&\sum_{d=1}^{D} \mathbf{w}^{d\top}\mathbf{B}\mathbf{x}^{d}[\Delta\tau_n] \\
&= \begin{bmatrix}\mathbf{w}^{1\top}\mathbf{B} & \mathbf{w}^{2\top}\mathbf{B} & \cdots & \mathbf{w}^{D\top}\mathbf{B}\end{bmatrix} \cdot \\
&\quad \begin{bmatrix}\mathbf{x}^{1}[\Delta\tau_n]^{\top} & \mathbf{x}^{2}[\Delta\tau_n]^{\top} & \cdots & \mathbf{x}^{D}[\Delta\tau_n]^{\top}\end{bmatrix}^{\top}, \\
&= \begin{bmatrix}\mathbf{w}^{1\top} & \mathbf{w}^{2\top} & \cdots & \mathbf{w}^{D\top}\end{bmatrix} (\mathbf{I}_D \otimes \mathbf{B}) \cdot \\
&\quad \begin{bmatrix}\mathbf{x}^{1}[\Delta\tau_n]^{\top} & \mathbf{x}^{2}[\Delta\tau_n]^{\top} & \cdots & \mathbf{x}^{D}[\Delta\tau_n]^{\top}\end{bmatrix}^{\top}
\end{aligned} \tag{15}$$

which is a scalar, so it can be further simplified as

$$\begin{aligned}
&\sum_{k=1}^{K} \mathbf{h}_k^{T}\mathbf{P}\mathbf{x}_k[\Delta\tau_j] \\
&= \mathbf{x}[\Delta\tau_n]\,(\mathbf{I}_D \otimes \mathbf{B})\,\mathbf{h}
\end{aligned}, \tag{16}$$

where $\mathbf{x}[\Delta\tau_n] = \begin{bmatrix}\mathbf{x}^{1}[\Delta\tau_n]^{\top} & \mathbf{x}^{2}[\Delta\tau_n]^{\top} & \cdots & \mathbf{x}_D[\Delta\tau_n]^{\top}\end{bmatrix}$ and $\mathbf{h} = \begin{bmatrix}\mathbf{w}^{1\top}\mathbf{w}^{2\top}\cdots\mathbf{w}^{D\top}\end{bmatrix}^{\top}$.

Therefore, (2) can be simplified as follows:

$$\mathcal{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\|\mathbf{y}(n) - \mathbf{x}[\Delta\tau_n]\left(\mathbf{I}_D \otimes \mathbf{B}^{\top}\right)\mathbf{w}\|^2 + \|\tilde{\mathbf{p}}\mathbf{w}\|^2\,, \tag{17}$$

which is (3) after definition in Section III.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China (No. 61806148) and the Fundamental Research Funds for the Central Universities (No. 22120180009).

REFERENCES

[1] C. Fu, A. Carrio, M. A. Olivares-Mendez, R. Suarez-Fernandez, and P. Campoy, "Robust real-time vision-based aircraft tracking from unmanned aerial vehicles," in *2014 ieee international conference on robotics and automation (ICRA)*, 2014, pp. 5441–5446.

[2] M. Olivares-Mendez, C. Fu, P. Ludivig, T. Bissyandé, 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.

[3] M. Mueller, G. Sharma, N. Smith, and B. Ghanem, "Persistent aerial tracking system for uavs," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 1562–1569.

[4] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for uav tracking," in *European conference on computer vision*. Springer, 2016, pp. 445–461.

[5] H. Kiani Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1135–1143.

[6] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4310–4318.

[7] 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.

[8] Y. Qi, S. Zhang, L. Qin, H. Yao, Q. Huang, J. Lim, and M.-H. Yang, "Hedged deep tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4303–4311.

[9] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3074–3082.

[10] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual tracking with fully convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3119–3127.

[11] J. Choi, H. Jin Chang, S. Yun, T. Fischer, Y. Demiris, and J. Young Choi, "Attentional correlation filter network for adaptive visual tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4807–4816.

[12] M. Wang, Y. Liu, and Z. Huang, "Large margin object tracking with circulant feature maps," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4021–4029.

[13] J. Van De Weijer, C. Schmid, J. Verbeek, and D. Larlus, "Learning color names for real-world applications," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1512–1523, 2009.

[14] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *British Machine Vision Conference, Nottingham, September 1-5, 2014*. BMVA Press, 2014.

[15] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[16] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[17] J. Sherman and W. J. Morrison, "Adjustment of an inverse matrix corresponding to a change in one element of a given matrix," *The Annals of Mathematical Statistics*, vol. 21, no. 1, pp. 124–127, 1950.

[18] M. Mueller, N. Smith, and B. Ghanem, "Context-aware correlation filter tracking," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 1387–1395.

[19] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. Torr, "Staple: Complementary learners for real-time tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1401–1409.

[20] S. Hare, S. Golodetz, A. Saffari, V. Vineet, M.-M. Cheng, S. L. Hicks, and P. H. Torr, "Struck: Structured output tracking with kernels," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 10, pp. 2096–2109, 2016.

[21] X. Jia, H. Lu, and M.-H. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *2012 IEEE Conference on computer vision and pattern recognition*. IEEE, 2012, pp. 1822–1829.

[22] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *European conference on computer vision*. Springer, 2012, pp. 702–715.

[23] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *International journal of computer vision*, vol. 77, no. 1-3, pp. 125–141, 2008.

[24] H. Fan and H. Ling, "Parallel tracking and verifying: A framework for real-time and high accuracy visual tracking," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5486–5494.

[25] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 7, pp. 1409–1422, 2012.

[26] Z. Hong, Z. Chen, C. Wang, X. Mei, D. Prokhorov, and D. Tao, "Multi-store tracker (muster): A cognitive psychology inspired approach to object tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 749–758.

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

[28] F. Li, C. Tian, W. Zuo, L. Zhang, and M.-H. Yang, "Learning spatial-temporal regularized correlation filters for visual tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4904–4913.

[29] J. Zhang, S. Ma, and S. Sclaroff, "Meem: robust tracking via multiple experts using entropy minimization," in *European conference on computer vision*. Springer, 2014, pp. 188–203.

[30] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *European conference on computer vision*. Springer, 2014, pp. 254–265.