BACKGROUND-SUPPRESSED CORRELATION FILTERS FOR VISUAL TRACKING

Zhihao Chen[1,3*], *Qing Guo*[2,3*], *Liang Wan*[1,3†], *Wei Feng*[2,3]

[1]School of Computer Software, Tianjin University, Tianjin, China
[2]School of Computer Science and Technology, Tianjin University, Tianjin, China
[3]Key Research Center for Surface Monitoring and Analysis of Cultural Relics, SACH, China
{zh_chen, tsingqguo, lwan, wfeng}@tju.edu.cn

ABSTRACT

Correlation filters (CF) visual object tracking is a powerful framework, with excellent tracking accuracy and beyond real-time frame rate. Its performance, however, can be severely degraded in cluttered background images. In this paper, we propose background-suppressed correlation filters (BSCF), a better CF tracking scheme, which can significantly improve the reliability and accuracy of CF trackers, without harming their beyond real-time speed. Specifically, we present a unified BSCF object function. We show that both the correlation filters and BS weight map can be efficiently and jointly solved in frequency domain. Extensive experiments on OTB-100 benchmark validate the effectiveness and generality of BS in improving multiple CF trackers with higher accuracy and robustness while maintaining their fast tracking speed. We also show BS boosted CF tracker can achieve comparable accuracy of the state-of-the-art spatially-regularized CF tracker but is 14 times faster.

Index Terms— Visual Object Tracking, Correlation Filters, Background Suppression

1. INTRODUCTION

As an important and challenging problem in computer vision, visual object tracking online estimates the position and size of an interested object through the whole video, which, ideally, is only specified at the first frame. Tracking plays a key role in many real-world applications, such as video surveillance, autonomous driving, human-machine interaction and robotics. To well adapt to the temporal variations of the object, state-of-the-art trackers usually maintain an online-updated object appearance model [1, 2, 3]. Latest discriminative trackers online learn a classifier to separate the object from background and outperform generative models on public benchmarks [1, 4]. Correlation filters (CF) based tracking is one of the most successful discriminative scheme that performs fast learning and detection in frequency domain and achieves not only accurate tracking but beyond real-time speed [5, 6]. Recently, a number of CF-based trackers have been proposed using various HOG, color or deep features [6, 7, 8, 9], robust scale estimation [10, 11], long-term memory components [12], and more effective updating strategy [3].

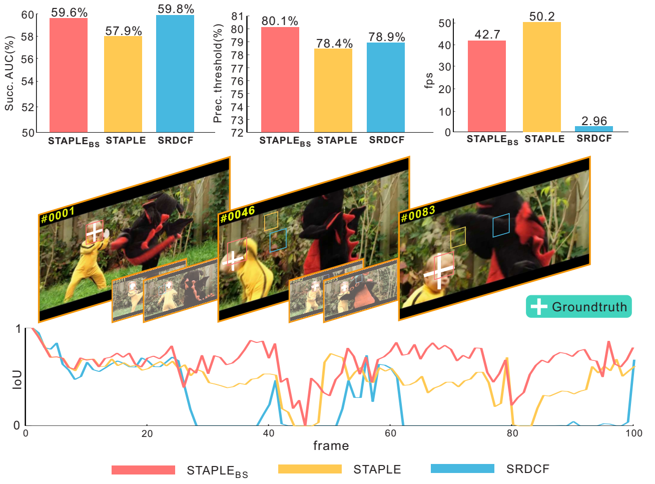

Fig. 1. Comparison of STAPLE [7], its improved version with our *background-suppressed correlation filters* (BSCF), i.e. STAPLE$_{\mathrm{BS}}$, and SRDCF [13]. The bar diagrams (top) show the tracking accuracy and average speed on OTB-100 [1]. STAPLE$_{\mathrm{BS}}$ outperforms STAPLE with 2.9% and 2.2% relative improvement in success plot AUC and precision at 20 pixels respectively while still maintaining real-time speed with only 7.5 fps deceleration. In contrast, although having similar accuracy with STAPLE$_{\mathrm{BS}}$, SRDCF runs at 2.96 fps and is 14.4 times slower than STAPLE$_{\mathrm{BS}}$, which validates the excellent performance of our BSCF. The tracking results on a challenge case, i.e. sequence of dragonbaby (middle and bottom), also present that our STAPLE$_{\mathrm{BS}}$ keeps the most accurate and tight tracking even under severe background clutter and fast motion.

Despite the balanced tracking accuracy and efficiency, most CF-based trackers tend to fail in cluttered background. As shown in Fig. 1, for instance, the fast moving character clutters the background, a state-of-the-art CF tracker STAPLE [7] easily loses the object. In addition, when multiple similar objects exist in the neighborhood, as shown in Fig. 2, classical CF-based trackers may generate a response map with

[*] are the equal contribution authors, [†] is the corresponding author. This work is supported by NSFC 61671325, 61572354, 61672376.

978-1-5386-1737-3/18/$31.00 ©2018 IEEE

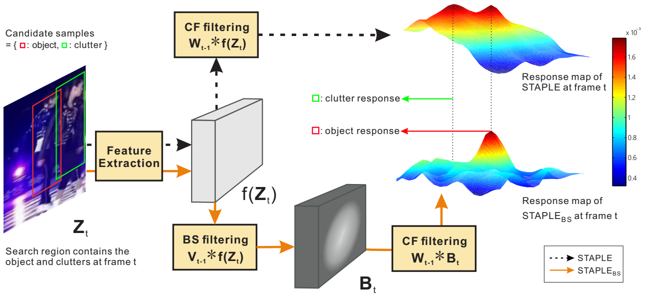

Fig. 2. Comparing a CF tracker, i.e. STAPLE [7], with its BSCF tracker, i.e. $\mathrm{STAPLE_{BS}}$ which is equipped with our *background suppression filtering*. At the frame t, there are two similar objects in the search region. The one with red bounding box is the interested target. Another is the clutter. Staple denoted as the dashed line generates a less discriminative response map having double high peaks and gets the highest value on the clutter. However, with the *background suppression filtering*, $\mathrm{STAPLE_{BS}}$ gets much more discriminative response map only having one high peak on target position.

two peaks and higher responses in the nearby background region. This is caused by two main reasons. First, the boundary effects of CF scheme introduced by the circularly shifted training sample [14] may be more harmful for cluttered background. Second, the features extracted from background regions are much less discriminative. A possible remedy is to spatially regularize the correlation filters [13] or to select effective training samples [14] to alleviate the boundary effects and boost the discriminative power of extracted features. We can also offline fine-tuned more effective deep features via a data-driven way to fix the incompetence of CF based trackers in cluttered background. However, all above solutions are either too expensive in practice, needing a large-volume of labeled data, or too slow to realize fast object tracking in mobile devices with limited computing resources [13]. As shown in Fig. 1, although getting much higher tracking accuracy than other CF trackers, spatially-regularized CF (SRDCF) [13] can only run at 2.9 fps on a common PC.

In this paper, we address the background clutter difficulty of CF tracking scheme and propose *background-suppressed correlation filters* (BSCF), which not only significantly improves the accuracy and robustness of CF trackers, but barely harms their beyond real-time speed. For example, as shown in the first row of Fig. 1, original STAPLE [7] tracker runs at 50.2fps with Succ. AUC 0.5790 and Prec. 78.4%; while BS-boosted STAPLE tracker, i.e. $\mathrm{STAPLE_{BS}}$, can run at 42.7fps with much higher accuracy (Succ. AUC 0.5960 and Prec. 80.1%). The accuracy of $\mathrm{STAPLE_{BS}}$ is comparable to that of SRDCF [13], whose frame rate is only 2.96fps, i.e. being 14 times slower than $\mathrm{STAPLE_{BS}}$. Our major contribution is three-fold. First, we propose *background suppression filtering* for CF tracking scheme, which is performed in the feature space before the complicated correlation filters optimization and can effectively exclude the cluttered background. As shown in Fig. 2, BSCF tracker gets a much more discriminative response map, thus can locate the target accurately. Second, we propose a BSCF objective function, with which BS and correlation filters can be efficiently and jointly solved in frequency domain. Third, we show the effectiveness and generality of our BS strategy on two state-of-the-art CF trackers.

2. CORRELATION FILTERS

Correlation filters (CF) are learnt via solving a linear regression objective function with dense training samples generated by circularly shifting a patch centered at the target. Such dense sampling process can be formulated as circular convolution that can be efficiently solved in frequency domain, thus leads to a high-speed tracker [6]. The general objective function for learning CF is defined as

$$\mathrm{E}(\mathbf{W}) = \left\| \sum_{l=1}^{D} (\mathbf{X}^l * \mathbf{W}^l) - \mathbf{Y} \right\|^2 + \lambda \left\| \mathbf{W} \right\|^2, \qquad (1)$$

where $\mathbf{X} \in \Re^{M \times N \times D}$ denotes features of a patch centered at the interested object; $\mathbf{W} \in \Re^{M \times N \times D}$ denotes the correlation filters we want to learn; $\mathbf{X}^l, \mathbf{W}^l \in \Re^{M \times N}$ are the lth channel of $\mathbf{X}$ and $\mathbf{W}$ with $l \in \{1, ..., D\}$; $\mathbf{Y} \in \Re^{M \times N}$ is the regression target via a 2D Gaussian function having highest value on the object location; λ is the regularization parameter to avoid over-fitting; '$*$' denotes the circulant convolution; $\|\cdot\|$ denotes the Frobenius-norm. Since the circular convolution becomes element-wise multiplication in frequency domain, $\mathbf{W}$ can be efficiently learnt by solving Eq. (1) in frequency domain. Besides, we can adopt the kernel trick to equip Eq. (1) to learn better filters, as done in [6].

CF uses the circulant convolution to achieve the dense sampling. However, most of the samples are not the real samples caused by the boundary effects. SRDCF [13] adopts spatial regularization term to solve this problem. The objective function can be expressed as

$$\mathrm{E}(\mathbf{W}) = \left\| \sum_{l=1}^{D} (\mathbf{X}^l * \mathbf{W}^l) - \mathbf{Y} \right\|^2 + \sum_{l=1}^{D} \left\| \mathbf{S} \odot \mathbf{W}^l \right\|^2, \qquad (2)$$

where $\mathbf{S} \in \Re^{M \times N}$ denotes the SR weight map which determines the confidence of each coordinate in filters $\mathbf{W}$.

Spatial regularization is a effective method but SRDCF is very slow. We can not solve Eq. (2) fast as Eq. (1) since the convolution operation that leads to a heavy computing burden still exists when Eq. (2) is transformed to the Fourier domain.

3. THE METHOD

3.1. Background-Suppressed Correlation Filters

We propose *background suppression filtering* for CF framework to address the background clutters on the feature space. As shown in Fig. 2, instead of directly filtering the features

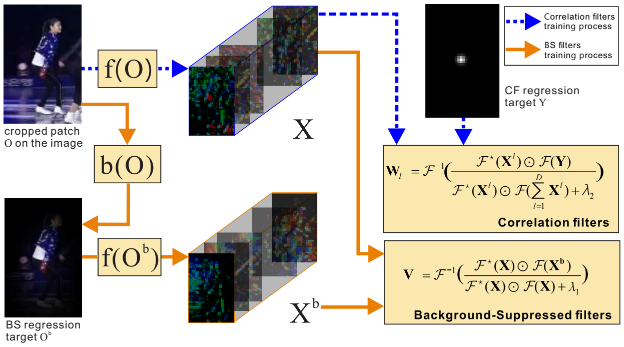
f(O)
cropped patch
O on the image
X
b(O)
f(O^b)
X^b
BS regression
target O^b
Correlation filters
training process
BS filters
training process
CF regression
target Y
$\mathbf{W}_l = \mathcal{F}^{-1}(\frac{\mathcal{F}^\star(\mathbf{X}^l)\odot\mathcal{F}(\mathbf{Y})}{\mathcal{F}^\star(\mathbf{X}^l)\odot\mathcal{F}(\sum_{l=1}^{D}\mathbf{X}^l)+\lambda_2})$
Correlation filters
$\mathbf{V} = \mathcal{F}^{-1}(\frac{\mathcal{F}^\star(\mathbf{X})\odot\mathcal{F}(\mathbf{X}^b)}{\mathcal{F}^\star(\mathbf{X})\odot\mathcal{F}(\mathbf{X})+\lambda_1})$
Background-Suppressed filters

Fig. 3. In training process, $\mathbf{O}$ represents the cropped patch centered at the target. We use the function $\mathrm{b}(\cdot)$ to generate the background-suppressed patch $\mathbf{O}^{\mathrm{b}}$. Then we respectively extracted the features $\mathbf{X}$, $\mathbf{X}^{\mathrm{b}}$ from $\mathbf{O}$, $\mathbf{O}^{\mathrm{b}}$ via the feature extraction function $\mathrm{f}(\cdot)$. Finally, we learn correlation filters and BS filters via Eq. (5) and and Eq. (6)

$\mathrm{f}(\mathbf{Z})$ of a patch $\mathbf{Z}$ with correlation filters $\mathbf{W}$, we first use BS filters, i.e. $\mathbf{V}$, to transform the $\mathrm{f}(\mathbf{Z})$ to $\mathbf{B}$ to suppress the feature values of background. The response map is finally generated by filtering $\mathbf{B}$ with traditional correlation filters, i.e. $\mathbf{W}$. Such BS filtering process helps to get a much more discriminative response map than the traditional CF framework.

Since we add an online filtering process to CF framework, it is important to explore an efficient online learning method for the new filters to realize a real-time tracker. We thus propose a novel objective function for CF with which BS and correlation filters can be jointly learnt in frequency domain. Specifically, given a training path centered at the object, i.e. $\mathbf{O} \in \Re^{M\times N\times 3}$, we first generate a background-suppressed patch (BS-patch) $\mathbf{O}^{\mathrm{b}} \in \Re^{M\times N\times 3}$ by multiplying $\mathbf{O}$ with a background-suppressed function, i.e. $\mathrm{b}(\cdot)$, that can be a 2D Gaussian function in an element-wise way. We regard the features of $\mathbf{O}^{\mathrm{b}}$ as the regression target of BS filters. We define the objective function for BSCF as following

$$\begin{aligned}\mathrm{E}(\mathbf{V},\mathbf{W}) \;&=\; \left\|\mathbf{X}*\mathbf{V}-\mathbf{X}^{\mathrm{b}}\right\|^2+\lambda_1\left\|\mathbf{V}\right\|^2 \\ &+\; \left\|\sum_{l=1}^{D}(\mathbf{X}^l*\mathbf{W}^l)-\mathbf{Y}\right\|^2+\lambda_2\left\|\mathbf{W}\right\|^2,\end{aligned} \tag{3}$$

where $\mathbf{X}^{\mathrm{b}} \in \Re^{M\times N\times D}$ is the feature of $\mathbf{O}^{\mathrm{b}}$, $\mathbf{V} \in \Re^{M\times N\times D}$ represents our BS filters. λ_1 and λ_2 are the regularization terms to avoid over-fitting.

In Eq. (3), several functions, e.g. 2D Gaussian, Cosine or the simplest square wave function can serve as the $\mathrm{b}(\cdot)$ function to suppress the background clutters in original patch $\mathbf{O}$. Here, we use the 2D Gaussian function $\mathrm{G}(x,y;\sigma)$ to suppress $\mathbf{O}$ and generate the BS-patch $\mathbf{O}^{\mathrm{b}}$ via

$$\mathbf{O}^{\mathrm{b}}(x,y)=\mathrm{b}(\mathbf{O})=\mathrm{G}(x,y;\sigma)\odot\mathbf{O}(x,y), \tag{4}$$

where (x,y) denotes the coordinate in $\mathbf{O}$ and $\mathbf{O}^{\mathrm{b}}$. $\mathrm{G}(x,y;\sigma)$ is the 2D Gaussian function using variance variable σ to control the degree of background suppression.

Actually, BS filtering can be regarded as a candidate samples selection process before detection. As shown in Fig. 2, the candidate samples from background can be suppressed at feature space via BS filtering, thus leading to a more discriminative response map. Besides, the added filters in Eq. (3) do not affect the solution of correlation filters and should not harm the effectiveness of original correlation filtering.

3.2. Fast Learning Correlation and BS Filters

Similar with the solution of traditional CF [6], these two filters $\mathbf{V}$ and $\mathbf{W}$ can be jointly and efficiently solved by minimizing the objective function Eq. (3) in frequency domain. Fig.3 shows the illustration of the overall learning procedure. The solutions of $\mathbf{V}$ and $\mathbf{W}$ are

$$\mathbf{V} \;=\; \mathscr{F}^{-1}\Big(\frac{\mathscr{F}^\star(\mathbf{X})\odot\mathscr{F}(\mathbf{X^b})}{\mathscr{F}^\star(\mathbf{X})\odot\mathscr{F}(\mathbf{X})+\lambda_1}\Big), \tag{5}$$

$$\mathbf{W}^l \;=\; \mathscr{F}^{-1}\Big(\frac{\mathscr{F}^\star(\mathbf{X}^l)\odot\mathscr{F}(\mathbf{Y})}{\mathscr{F}^\star(\mathbf{X}^l)\odot\mathscr{F}(\sum_{l=1}^{D}\mathbf{X}^l)+\lambda_2}\Big), \tag{6}$$

where $\mathscr{F}(\cdot)$ and $\mathscr{F}^{-1}(\cdot)$ are the Fourier and inverse Fourier transformations. '$\star$' denotes the complex conjugate.

The regression objective function Eq. (3) can be further improved by using kernel trick, as done in [6]. Specifically, with kernel trick, we get $\mathbf{X}*\mathbf{V}=\boldsymbol{\beta}*\mathrm{K}(\mathbf{X},\mathbf{V})$ and $\mathbf{X}^l*\mathbf{W}^l=\boldsymbol{\alpha}^{\boldsymbol{l}}*\mathrm{K}(\mathbf{X}^l,\mathbf{W}^l)$, where $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ are the dual variables corresponding to the primal variables $\mathbf{W}$ and $\mathbf{V}$, respectively. $\mathrm{K}(\cdot)$ is the kernel correlation function as defined in [6] with a specified kernel function. With the kernel trick, we aim to learn $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ instead of $\mathbf{W}$ and $\mathbf{V}$. Both can also be efficiently solved in frequency domain via

$$\hat{\boldsymbol{\beta}} \;=\; \frac{\hat{\mathbf{X}}^{\mathrm{b}}}{\mathrm{K}(\hat{\mathbf{X}},\hat{\mathbf{X}})+\lambda_1}, \tag{7}$$

$$\hat{\boldsymbol{\alpha}}^l \;=\; \frac{\hat{\mathbf{Y}}}{\mathrm{K}(\hat{\mathbf{X}}^l,\sum_{l=1}^{D}\mathbf{X}^l)+\lambda_2}. \tag{8}$$

With above two functions, we can use our BSCF to improve those CF trackers using kernel trick, e.g. SAMF [11].

Eq. (5), i.e. BS filters, can be calculated fast as Eq. (6), i.e. correlational filters, due to the fast solution to the element-wise in Fourier domain. Complexity of SRDCF is $\mathcal{O}(DMN\log(MN))$ and complexity of BSCF is just $\mathcal{O}(DMN)$. So our BSCF is much faster than SRDCF.

3.3. BS Boosted CF Tracking

Background suppression, working as a fundamental part in correlation filters tracking, would be applied to extensive mainstream trackers and can be combined with other functional parts, e.g. kernel trick [6], scale detection methods [11, 10] and color histogram [7], to get a better tracker.

Algorithm 1: BS boosted CF Tracking

```
Input: Initial bounding box $\mathbf{box}_1 \in \Re^4$ of interested
        object, image set $\mathbf{I}$ of video
Output: $\{\mathbf{box}_t | t = 2, ..., \mathrm{nFrames}\}$
Initialize position $\mathbf{p}_1$, scale $\mathbf{s}_1$ and regression target $\mathbf{Y}$;
while $t <= \mathrm{nFrames}$ do
    Background suppression:
    1: Crop the $\mathbf{Z}_t$ from $\mathbf{I}_t$ according to $\mathbf{p}_{t-1}$ and $\mathbf{s}_{t-1}$;
    2: Extract the $\mathrm{f}(\mathbf{Z}_\mathrm{t})$ from $\mathbf{Z}_t$;
    3: Generate $\mathbf{B}_t$ with $\mathrm{f}(\mathbf{Z}_\mathrm{t})$ and $\mathbf{v}_{t-1}$ via
       Eqs. (9) & (12);
    Position estimation:
    1: Compute the response map $\mathbf{R}_t$ with $\mathbf{B}_t$ and $\mathbf{W}_{t-1}$
       via Eqs. (11) & (13);
    2: Set $\mathbf{p}_{t+1}$ to the object position that maximizes $\mathbf{R}_t$;
    Model update:
    1: Extract the $\mathbf{X}_t$ from $\mathbf{I}_t$ according to $\mathbf{p}_t$ and $\mathbf{s}_t$;
    2: Crop $\mathbf{X}_t^b$ from $\mathbf{X}_t$ with a proper BS strategy;
    3: Update the BS filters and correlation filters via
       Eqs. (5) & (7) & (8);
    $t = t + 1$;
end
```

We present the general process of BS boosted CF tracking in Algorithm 1 whose two key processes, i.e. detection and updating, are detailed in following.

Detection via BSCF. During the detection, we first preprocess the features $\mathrm{f}(\mathbf{Z})$ extracted by patch $\mathbf{Z}$ through the trained BS filters $\mathbf{V}$, then we can finally get the new background-suppressed patch $\mathbf{B} \in \Re^{M \times N \times D}$. The formula in the Fourier domain is as below

$$\hat{\mathbf{B}} = \hat{\mathbf{V}} \odot \hat{\mathrm{f}}(\mathbf{Z}), \tag{9}$$

where '$\hat{}$' denotes the Fourier domain representation of a signal; $\hat{\mathbf{B}} \in \Re^{M \times N \times D}$ denotes the output of background suppression at frame $t+1$ in Fourier domain; $\hat{\mathbf{V}} \in \Re^{M \times N \times D}$ denotes BS filters learned at frame t in Fourier domain;

The strategy of cropping the background clutter completely is not enough robust due to the edge effect of Fast Fourier Transform (FFT). So, we fuse $\mathrm{f}(\mathbf{Z})$ to $\mathbf{B}$ with a proper fusing rate so as to eliminate the boundary effect of FFT. We adopt coefficient γ to control the fusing rate. After fusing the part of original, we can obtain the final $\hat{\mathbf{B}}'$, i.e.

$$\hat{\mathbf{B}}' = (1-\gamma)\hat{\mathrm{f}}(\mathbf{Z}) + \gamma\hat{\mathbf{B}}, \tag{10}$$

Then, as the background-suppressed production of $\hat{\mathrm{f}}(\mathbf{Z})$, $\hat{\mathbf{B}}' \in \Re^{M \times N \times D}$ becomes the new input of traditional CF detection. Let $\hat{\mathbf{R}} \in \Re^{M \times N}$ represent the response map in Fourier domain. We can rewrite the formula as follow

$$\hat{\mathbf{R}} = \sum \hat{\mathbf{W}} \odot \hat{\mathbf{B}}', \tag{11}$$

where $\hat{\mathbf{W}} \in \Re^{M \times N}$ donates the traditional correlation filters in Fourier domain.

The detection can also be solved in the dual domain. Here, in order to express simply, we combine the Eq. (9) with Eq. (10) and transform them into the dual domain. We adopt γ to control the fusing rate, then we have

$$\hat{\mathbf{B}}' \;=\; (1-\gamma)\hat{\mathrm{f}}(\mathbf{Z}) + \gamma(\mathrm{K}(\hat{\mathrm{f}}(\mathbf{Z}), \hat{\mathbf{X}}^{\mathrm{m}}) \odot \hat{\boldsymbol{\beta}}), \tag{12}$$

$$\hat{\mathbf{R}} \;=\; \mathrm{K}(\hat{\mathbf{B}}', \hat{\mathbf{X}}^{\mathrm{m}}) \odot \hat{\boldsymbol{\alpha}}, \tag{13}$$

where $\hat{\mathbf{X}}^{\mathrm{m}}$ donates the feature map model in Fourier domain that needs to be updated at each frame.

Training and updating models. We update the filter model with a proper learning rate instead of completely covering formerly trained filter. We denote learning rate to η, μ

$$\hat{\mathbf{V}}_{\mathrm{t}} \;=\; (1-\eta)\hat{\mathbf{V}}_{\mathrm{t-1}} + \eta\hat{\mathbf{V}}'_{\mathrm{t}}, \tag{14}$$

$$\hat{\mathbf{W}}_{\mathrm{t}} \;=\; (1-\mu)\hat{\mathbf{W}}_{\mathrm{t-1}} + \mu\hat{\mathbf{W}}'_{\mathrm{t}}, \tag{15}$$

where, $\hat{\mathbf{V}}'_{\mathrm{t-1}}$, $\hat{\mathbf{W}}'_{\mathrm{t-1}}$ are obtained by Eq. (5) at frame t; $\hat{\mathbf{V}}'_{\mathrm{t}}$, $\hat{\mathbf{W}}'_{\mathrm{t}}$ represents the historical filter model at frame $t-1$; $\hat{\mathbf{V}}_{\mathrm{t}}$, $\hat{\mathbf{W}}_{\mathrm{t}}$ represents the final filter model in frame t after updating with learning rate η, μ.

4. EXPERIMENT RESULTS

4.1. Setup

Dataset and metrics. We use the object tracking benchmark OTB-100 [1] to validate the effectiveness of our method. OTB-100 contains 100 sequences that can be grouped into 11 subsets according 11 attributes. We then benchmark them against their baseline versions. In addition, we use the object tracking benchmark OTB-100 to evaluate the performance of all trackers. There are two metrics included, i.e. bounding box overlap ratio and center location error. By setting a success threshold for each metric, we can get the precision and success plots, which quantitatively measure the performance of different trackers on OTB-100. For the fair comparison, all trackers are run on the same workstation (Intel Core i7-2600 3.4GHz 8GB RAM) using MATLAB thereby we can compare their fps under the same conditions.

Baselines. We select two representative CF trackers STAPLE and SAMF as baselines. We only choose trackers that are based on the CF formulation (Eq. (1)) and these trackers tend to adopt the different features and different ways of implementation. We apply our background-suppressed method to these trackers and denote them $\mathrm{STAPLE_{BS}}$, $\mathrm{SAMF_{BS}}$. To put the tracking performance into perspective, we also compare the baseline trackers (STAPLE [7] and SAMF [11]) and their background-suppressed versions ($\mathrm{STAPLE_{BS}}$ and $\mathrm{SAMF_{BS}}$) to the most recent state-of-art trackers (HCFT [8] and CFNet [15]), which are not necessarily CF based. In addition, we include the real-time CF trackers (SRDCF [13])(LMCF [16] and MEEM [17]) and other recent popular trackers (DLSSVM [18] and LCT [12]).

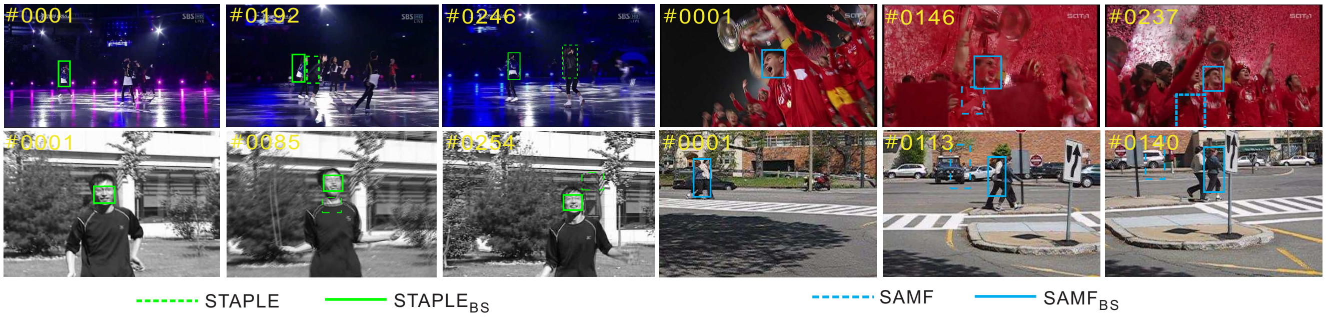

Fig. 6. Trackers results of two baseline trackers against our background-suppressed versions. The trackers and corresponding videos chosen from the OTB-100 are (from top to bottom): STAPLE: skating1, jumping; SAMF: soccer, couple. Obviously, our improved BS versions tend to track the target more tight and robust when it comes to the background clutters and fast motion.

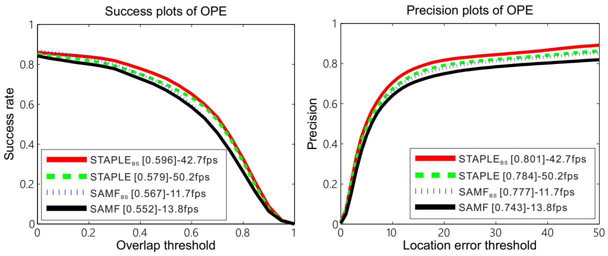

Fig. 4. Success and precision plots of OPE (one pass evaluation) on OTB-100. The numbers in the legend indicate the area-under-curve (AUC) score for success plots and the representative precisions at 20 pixels for precision plots, respectively. Our method improves baselines by 2-3% in performance just slows the speed by 13-15%.

Parameter settings. All baseline trackers are run with the standard parameters provided by the authors. In our improved trackers, we keep it as the standard if used. We set the regularization factor λ_1 to $\{1,1\}$, the update rule with learning rate $\{0.3,0.2\}$, the BS fusing rate γ $\{0.2,0.4\}$ and adopt the proper variance σ in Gaussian window to $\{0.3,0.25\}$.

4.2. BS boosted CF vs. CF

Overall results. Fig.4 shows the performance of two baseline trackers and their BS versions on OTB-100. Our $\mathrm{STAPLE_{BS}}$ improves the STAPLE by 2.9% in success plot and 2.1% in precision plot. Our $\mathrm{SAMF_{BS}}$ improves the SAMF by 2.7% in success plot and 4.6% in precision plot. The extensive experiments demonstrate that our method not only helps to make the position of center more accurate but also get more tight bounding box, i.e. is beneficial to the scale detection. Due to adding the other BS filters, our BS version trackers must be slower than the original versions. However, since the BS filters also are solved fast in the Fourier domain just like raw correlation filters, we do the extra BS at a very limited cost of speed. On average, our BS version trackers slow down the raw version trackers by 15%.

Attribute based comparison. While our method improves the overall performance in most scenarios there are certain categories that benefit more than others. As the Fig.5 shown, the most major improvement is achieved in the cases of background clutter, fast motion, occlusion and motion blur.

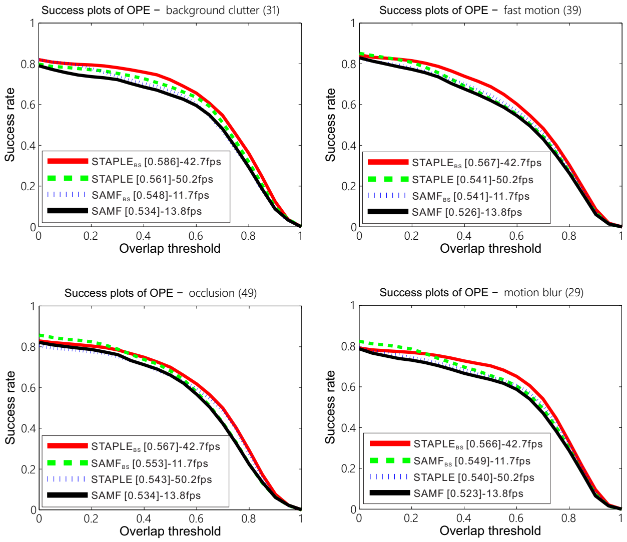

Fig. 5. All these figures indicate the area-under-curve (AUC) score for each attribute. We choose 4 attributes that includes background clutter, fast motion, occlusion and motion blur. In these attributes, our BS version improves the raw tracker by 4%-7% in performance.

Qualitative results. To visualize the impact of our proposed method on tracking performance, we show some examples of each baseline tracker compared to its background-suppressed version of sample videos from OTB-100 in Fig.6. Obviously, our BS versions have better performance than baseline trackers in fast motion (jumping, couple) and background clutter (skating1, soccer). For instance, in the 'skating1' sequence, there exist many similar skaters. At frame #190, two skaters including the interested object start to be very close. STAPLE get the double high responses to object and clutter. Due to the background-suppressed method, our $\mathrm{STAPLE_{BS}}$ instead get only one high response to our object. With STAPLE, after frame #192 the response score of clutter surpasses the score of object thereby the subsequent tracking is failed. However, our $\mathrm{STAPLE_{BS}}$ maintain the accurate and tight tracking all the time because the response of clutter tend to be suppressed in detection.

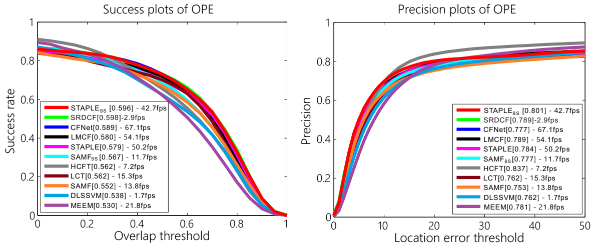

Fig. 7. Success and precision plots of OPE on OTB-100. These two figures show the average performance of state-of-art trackers and our BS trackers. It is obvious that our BS tracker $\mathrm{STAPLE}_{\mathrm{BS}}$ obtains the second highest overlap AUC 0.596 and keep the fast real-time speed at the same time.

4.3. Comparison with State-of-the-art Trackers

Fig.7 illuminates the results of the overlap of success plot and the location error of precision plot in all mentioned trackers. Those trackers adopt different objective functions, different features or other methods. We can easily find that our improved tracker $\mathrm{STAPLE}_{\mathrm{BS}}$ achieves the second best performance in both success and precision plots. Our $\mathrm{STAPLE}_{\mathrm{BS}}$ keep the fast real-time speed as CFNet, LMCF but surpass them by 1.2%, 2.8% in success plot and 3.1%, 1.5% in precision plot. SRDCF achieves the rank 1 in success plot and HCFT achieves rank 1 in precision plot. however SRDCF, HCFT have just 2.9fps and 7.2fps respectively that are far below our $\mathrm{STAPLE}_{\mathrm{BS}}$ tracker thereby they can not satisfy the demand of real-time. In addition, our improved tracker $\mathrm{SAMF}_{\mathrm{BS}}$ keep the primary speed of SAMF and surpass the LCT in both success and precision plots by 0.8% and 1.9%.

5. CONCLUSION

In this paper, we have proposed *background-suppressed correlation filters* (BSCF), a general and effective strategy to significantly improve the accuracy and robustness of CF tracking scheme in handling cluttered background. With a unified BSCF objective function, both the correlation filters and BS weight map can be efficiently and jointly solved in frequency domain. Extensive experiments on benchmark dataset show the superiority and generality of BSCF over classical CF trackers with higher accuracy and comparable beyond real-time speed. Since BS can be used as a general tool for the CF tracking scheme, in the future, we are interested to investigate its role in boosting more recent CF trackers, e.g. [9, 19] and to compare its performance with spatial regularization. Besides,object structure from superpixels [20] and segmentation [21, 22] may help get better tracking accuracy.

6. REFERENCES

[1] Y. Wu, J. Lim, and M.-H. Yang, “Object tracking benchmark,” *IEEE TPAMI*, vol. 37, no. 9, pp. 1834–1848, 2015.

[2] Q. Guo, W. Feng, C. Zhou, C.-M. Pun, and B. Wu, “Structure-regularized compressive tracking with online data-driven sampling,” *IEEE TIP*, vol. 26, no. 12, pp. 5692–5705, 2017.

[3] Q. Guo, W. Feng, C. Zhou, R. Huang, L. Wan, and S. Wang, “Learning dynamic Siamese network for visual object tracking,” in *ICCV*, 2017.

[4] M. Kristan and et al., “The visual object tracking vot2015 challenge results,” in *ICCVW*, 2015.

[5] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, “Visual object tracking using adaptive correlation filters,” in *CVPR*, 2010.

[6] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, “High-speed tracking with kernelized correlation filters,” *IEEE TPAMI*, vol. 37, no. 3, pp. 583–596, 2015.

[7] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, “Staple: Complementary learners for real-time tracking,” in *CVPR*, 2016.

[8] C. Ma, J. B. Huang, X. Yang, and M. H. Yang, “Hierarchical convolutional features for visual tracking,” in *ICCV*, 2015.

[9] R. Z. Han, Q. Guo, and W. Feng, “Content-related spatial regularization for visual object tracking,” in *ICME*, 2018.

[10] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, “Accurate scale estimation for robust visual tracking,” in *BMVC*, 2014.

[11] Y. Li and J. Zhu, “A scale adaptive kernel correlation filter tracker with feature integration,” in *ECCVW*, 2014.

[12] C. Ma, X. Yang, Chongyang Zhang, and M. H. Yang, “Long-term correlation tracking,” in *CVPR*, 2015.

[13] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, “Learning spatially regularized correlation filters for visual tracking,” in *ICCV*, 2015.

[14] H. K. Galoogahi, T. Sim, and S. Lucey, “Correlation filters with limited boundaries,” in *CVPR*, 2015.

[15] J. Valmadre, L. Bertinetto, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, “End-to-end representation learning for correlation filter based tracking,” in *CVPR*, 2017.

[16] M. Wang, Y. Liu, and Z. Huang, “Large margin object tracking with circulant feature maps,” in *CVPR*, 2017.

[17] J. Zhang, S. Ma, and S. Sclaroff, “Meem: Robust tracking via multiple experts using entropy minimization,” in *ECCV*, 2014.

[18] J. Ning, J. Yang, S. Jiang, L. Zhang, and M. H. Yang, “Object tracking via dual linear structured svm and explicit feature map,” in *CVPR*, 2016.

[19] R. Huang, W. Feng, and J. Sun, “Color feature reinforcement for cosaliency detection without single saliency residuals,” *IEEE SPL*, vol. 24, no. 5, pp. 569–573, 2017.

[20] L. Li, W. Feng, L. Wan, and J. Zhang, “Maximum cohesive grid of superpixels for fast object localization,” in *CVPR*, 2013.

[21] Q. Guo, S. Sun, X. Ren, F. Dong, B. Z. Gao, and W. Feng, “Freqeuncy-tuned active contour model,” *Neurocomputing*, vol. 275, no. 31, pp. 2307–2316, 2018.

[22] W. Feng, J. Jia, and Z. Liu, “Self-validated labeling of markov random fields for image segmentation,” *IEEE TPAMI*, vol. 32, pp. 1871–1887, 2010.