Robust and Real-Time Visual Tracking Based on Complementary Learners

Xingzhou Luo, Dapeng Du, and Gangshan Wu(✉)

State Key Laboratory for Novel Software Technology,
Nanjing University, Nanjing, China
xzluo@smail.nju.edu.cn, dudp.nju@gmail.com, gswu@nju.edu.cn

Abstract. Correlation filter based tracking methods have achieved impressive performance in recent years, showing high efficiency and robustness to challenging situations which exhibit illumination variations and motion blur. However, how to reduce model drift phenomenon which is usually caused by object deformation, abrupt motion, heavy occlusion and out-of-view, is still an open problem. In this paper, we exploit the low dimensional complementary features and an adaptive online detector with the average peak-to-correlation energy to improve tracking accuracy and time efficiency. Specifically, we appropriately integrate several complementary features in the correlation filter based discriminative framework and combine with the global color histogram to further boost the overall performance. In addition, we adopt the average peak-to-correlation energy to determine whether to activate and update an online CUR filter for re-detecting the target. We conduct extensive experiments on challenging OTB-15 benchmark datasets, and experimental results demonstrate that the proposed method achieves promising results in terms of efficiency, accuracy and robustness while running at 46 FPS.

Keywords: Visual tracking · Correlation filter
Dimension reduction · Online detection

1 Introduction

Visual object tracking is one of the most challenging tasks in the field of computer vision and has a wide range of applications such as video surveillance, human-computer interaction, autonomous driving and robotics. In generic visual tracking, given an initial state of the object in the first frame, the goal is to estimate the trajectory of the target throughout video sequences. Despite the significant progress in visual tracking, model drift problem and scale estimation usually lead to tracking failure in challenging situations such as illumination variation, deformation, fast motion, occlusion, out of view, scale variation and background clusters.

K. Schoeffmann et al. (Eds.): MMM 2018, Part II, LNCS 10705, pp. 213–225, 2018.
https://doi.org/10.1007/978-3-319-73600-6_19

In recent years, correlation filter based discriminative methods have shown excellent performance in terms of accuracy and robustness. However, Discriminative Correlation Filter based methods [1,4,7,10,11,14,15,22] learn a correlation filter from raw pixels, Histogram of Oriented Gradients (HOG) [3] or Color Names (CN) [16] features on a set of training samples. Owing to the limitation of each type of feature in some certain scenes, those methods which employ a type of feature fail to handle scale variation accurately and deal with complex conditions. Although several methods fuse multiple features or models to learn the target appearance model, the online models tend to drift due to fast motion and occlusion. Figure 1 presents some examples of tracking failure. In addition, several complicated tracking algorithms improve performance at the price of reducing tracking speed, which limits their real-time performance in real-world applications.

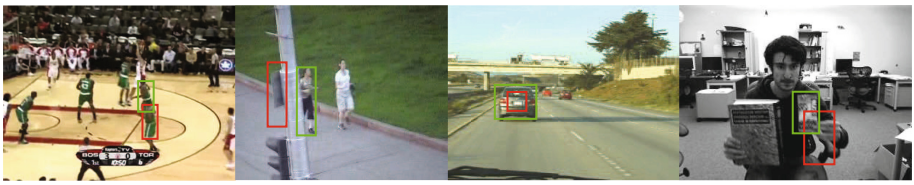

Fig. 1. Examples of tracking failure in visual object tracking (from left to right are Basketball, Jogging-1, Car24 and ClifBar). The red and green rectangles denote the incorrect tracking and ground truth bounding boxes, respectively. Model drift (column 1, 2 and 4) happens due to background clusters, fast motion and occlusion; Scale variation (column 3) occurs because of illumination change and scale variation. (Color figure online)

To overcome the aforementioned issues, we propose a robust tracking method by extending the Staple tracker with a low dimensional complementary features and use an adaptive online detector with the average peak-to-correlation energy (APCE) [17] to achieve tracking robustness and real-time performance. First, we appropriately integrate complementary features including HOG, Color Names and intensity with dimension reduction in the correlation filter based discriminative framework and combine with color histogram-based model to further boost the accuracy and efficiency of visual tracking. In addition, complementary features are extended to learn the scale filter for accurate scale estimation. Finally, we employ APCE to determine whether to activate and update an online CUR filter for re-detecting the target.

To evaluate the performance of the proposed method, we evaluate our method on the large-scale benchmark OTB15 datasets [20] with 100 challenging video sequences. Compared to a variety of state-of-the-art trackers, extensive experiments show that our method achieves appealing performance in terms of efficiency, accuracy and robustness. The contributions of this paper is briefly summarized as follows:

– We extend the Staple tracker by integrating the complementary features to enhance the discriminative ability to illumination variation. Besides, we employ a dimension reduction strategy to improve robustness to noise interference and tracking speed.
– We adopt the average-peak-to-correlation to determine whether to activate and update an online CUR filter, which is conducive to dealing with tracking failure effectively.

2 Related Work

Visual object tracking has been studied extensively with a variety of applications and achieved extremely excellent performance in the field of computer vision. We briefly review the relevant researches on correlation filter based tracking and tracking-by-detection approaches.

Correlation Filter. Correlation filter based tracking has widely captured researcher's attention in recent years. Minimum Output Sum of Squared Error (MOSSE), proposed by Bolme et al. [2], adopts raw pixels to model the target appearance by adaptive correlation filters. Henriques et al. [10] propose Circular Structure with Kernels tracker (CSK) that utilizes the structure of the circulant patch to learn a kernelized least squares classifier of the target from a single image patch with dense sampling, and then extend CSK by using the kernelized ridge regression and apply HOG features instead of raw pixels to Kernelized Correlation Filters (KCF) [11] to boost the performance of visual tracking. To obtain more superior performance than the CSK tracker, Danelljan et al. [7] introduce the sophisticated color features called Color Names into the framework of the CSK tracker for color sequences. Adaptive low-dimensional variant of color attributes is mainly used for accelerating tracking. Danelljan et al. [4] propose DSST to learn separate discriminative correlation filters using HOG features for translation estimation and handling the scale changes of the target during visual tracking. Scale Adaptive with Multiple Features (SAMF) [14] learns appearance model of the target by fusing both HOG and Color Names features together to facilitate robust tracking with the scale adaptive scheme. Sum of Template And Pixel-wise (Staple) [1] exploits complementary learners including the template-related learner and the histogram-related learner in the ridge regression framework to preserve robustness to color changes and deformations. However, the above mentioned CFT trackers are less effective for dealing with scale variation and model drift problem due to various challenging caused by fast motion, background cluster, long-term occlusion and out-of-view.

Tracking-by-detection. Tracking-by-detection approaches are exceedingly popular due to their high efficiency and performance. These tracking algorithms generally adopt the binary classifier which segregates the target from background

to perform visual tracking. To alleviate the stability-plasticity dilemma regarding online update in visual tracking, Kalal et al. [12] propose a novel Tracking-Learning-Detection (TLD) framework that explicitly decomposes the long-term tracking task into three components: tracking, learning and detection where the tracker provides labeled training data for training and updating detector and the detector re-initializes the tracker when tracking failure happens. Hare et al. [9] consider the spatial distribution of training samples, and integrate features and kernels into an online structured output SVM learning framework to predict the object location. Zhu et al. [22] propose the collaborative correlation tracker that jointly employ multi-scale kernelized correlation filter to learn the target appearance and introduce an efficient online CUR filter for detection which alleviates the model drift. Ma et al. [15] use discriminative correlation filters for translation and scale estimation, and develop an online random ferns classifier to redetect the target in case of tracking failure. Different from the above trackers, we utilize APCE to determine whether to activate and update the online detector.

3 Tracking Components

We aim to build a robust and real-time tracking method. Recently, the Staple [1] tracker achieves appealing performance with high speed. Due to the competitive performance and efficiency, we base our approach on the Staple tracker. In this section, we firstly review the Staple tracker. Secondly, we introduce the complementary feature used in our method to enhance robustness to illumination change. Moreover, we introduce a dimension reduction strategy to remove noise interference for target estimation and improve efficiency. Finally, to effectively deal with tracking failure, we utilize an adaptive online detection scheme with the average peak-to-correlation energy. Figure 2 shows an overview of our proposed method.

3.1 The Staple Tracker

The Staple [1] tracker combines two responses—the template response is learnt from HOG feature that is insensitive to color changes, and the histogram response is learnt from the global color histogram that is robustness to shape deformation. The models are learnt by solving two independent ridge regression problems, which retains the efficiency of the correlation filter and avoids ignoring the information captured by the color histogram response.

The template response is learnt under the least-squares correlation filter formulation. Multi-channel correlation filters are learnt from a single sample of the target that consists of d-dimensional feature maps f. The optimal correlation filter h is achieved by minimizing the objective

$$\min_{h} \left\| \sum_{i}^{d} h^i \star f^i - y \right\|^2 + \lambda \sum_{i}^{d} \left\| h^i \right\|^2, \tag{1}$$

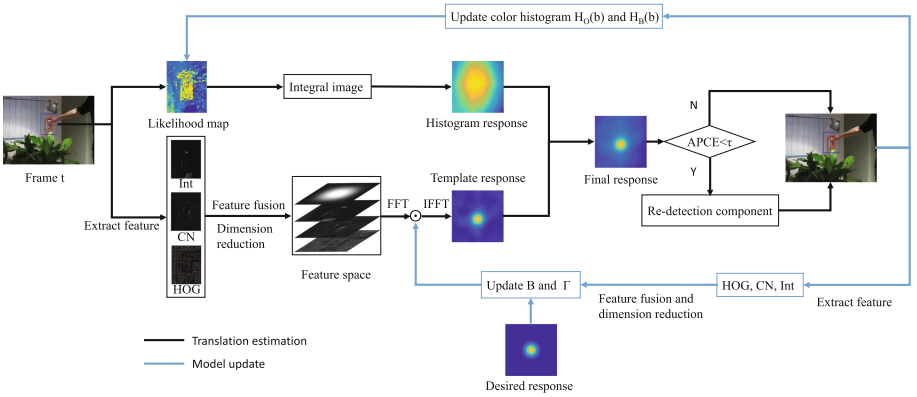

Fig. 2. The framework of our proposed method. At each frame, an image patch at estimated position and size from the previous frame is cropped as current input. First, the template response is calculated from the correlation between low dimensional complementary features and the previous learned filter ($\odot$ denotes element-wise computation). Meanwhile, the histogram response is computed by using the integral image from the target likelihood map. Then, the final response map is obtained by the linear combination of the template and histogram responses. Next, we use APCE calculated from the final response to consider whether to activate the online detector and re-detect the target. Finally, we update the template and histogram-related model parameters at estimated target state.

where f^i is the feature map of the i-th channel of f, λ is a regularization term that prevents over-fitting, y is the desired correlation output and the star $\star$ denotes the circular correlation. Further, the correlation operation is performed in the Fourier domain. As the method presented in [4], the solution of the filter is given by

$$H^i = \frac{\overline{Y} F^i}{\sum_{j=1}^{d} \overline{F^j} F^j + \lambda}, \tag{2}$$

where F^i is the discrete Fourier transform (DFT) of f^i, F^j is the DFT of f^j and $\overline{Y}$ denotes the complex conjugate of the DFT of y.

For Eq. (2), an optimal filter H_i can be achieved by solving a $d \times d$ linear system of equation per pixel, which triggers a computational bottleneck for online learning step in Discriminative Correlation Filter based tracking algorithm. To obtain a robust approximation, instead of performing expensive computation, the numerator B_t^i and the denominator Γ_t^i of the optimal filter H_t^i are updated separately as

$$\begin{aligned} B_t^i &= (1 - \alpha_{cf})\, B_{t-1}^i + \alpha_{cf} \overline{Y} F_t^i \\ \Gamma_t &= (1 - \alpha_{cf})\, \Gamma_{t-1} + \alpha_{cf} \sum_{j=1}^{d} \overline{F_t^j} F_t^j. \end{aligned} \tag{3}$$

Here, α_{cf} is the learning rate parameter of the template response. Moreover, the correlation score y_{cf} is estimated by using the inverse DFT:

$$y_{cf} = \mathscr{F}^{-1}\left(\frac{\sum_{i=1}^{d} \overline{B_{t-1}^{i}} Z_t^i}{\Gamma_{t-1} + \lambda}\right). \tag{4}$$

The histogram response is learnt under the color histogram based Bayes framework. Given the object region O and its surrounding region S, both the color histogram of the object and the background are calculated to obtain the histogram response. According to Bayes theorem, the object likelihood at location x is denoted as

$$P\left(x \in O|O, B\right) \approx \frac{P\left(b_x|x \in O\right) P\left(x \in O\right)}{\sum_{\Omega \in \{O,B\}} P\left(b_x|x \in \Omega\right) P\left(x \in \Omega\right)}. \tag{5}$$

Let $H_{\Omega}(\cdot)$ and b_x denote the color histogram which is calculated over the region Ω. Then, Eq. (5) simplifies to

$$P\left(x \in O|O, B\right) = \frac{H_O(b_x)}{H_O(b_x) + H_B(b_x)}. \tag{6}$$

In addition, the model parameters $H_O\left(b\right)$ and $H_B\left(b\right)$ are updated online as

$$\begin{aligned} H_{O,t}(b) &= \left(1 - \alpha_{ch}\right) H_{O,t-1}(b) + \alpha_{ch} H_{O,t}(b) \\ H_{B,t}(b) &= \left(1 - \alpha_{ch}\right) H_{B,t-1}(b) + \alpha_{ch} H_{B,t}(b). \end{aligned} \tag{7}$$

Finally, the response of color histogram y_{ch} is calculated by employing the integral image from $P\left(x \in O|O, B\right)$. The final response map g is obtained by the linear combination of the template response and histogram response as

$$g = \gamma y_{ch} + \left(1 - \gamma\right) y_{cf}. \tag{8}$$

3.2 Multiple Feature Fusion

Generally, the correlation filter only performs dot-product operation on multiple features and sums over image features in Fourier domain. The popular HOG features have been successfully applied in various practical applications [3,8,11]. Color attributes, or Color Names [16], which are linguistic color labels assigned by human to describe colors in the real world, have shown excellent results for object recognition [13]. HOG features mainly analyze image gradients while Color Names focus on color representation. To address the limitation of HOG features under illumination variation and deformation, we concatenate HOG features with complementary features Color Names and intensity into a vector to represent the target appearance model in our method.

Furthermore, we extend multiple features into the scale search procedure to achieve more accurate scale estimation. To handle scale variation, we follow the scale search strategy that the scale filter is learnt by constructing scale feature pyramid proposed by Danelljan et al. [4] at estimated target location.

3.3 Dimension Reduction

The FFT operations consume the expensive computation for the template response and scales linearly with the feature dimension. To reduce the computation cost of FFT, we introduce a dimension reduction strategy [5] that retains useful information to boost the speed of our approach.

Instead of updating the target appearance v_t, we use the learned appearance $v_t(\boldsymbol{l})$ to construct a $\hat{d} \times d$ project matrix Q_t. The project matrix Q_t is used to reconstruct the compressed target template v_t as $\hat{v}_t(\boldsymbol{l}) = Q_t v_t(\boldsymbol{l})$, where $\boldsymbol{l}$ is the tuple index that covers all elements in the target appearance v_t. The project matrix Q_t is estimated by minimizing the reconstruction error of the target template v_t

$$\epsilon = \sum_{\boldsymbol{l}} \|v_t(\boldsymbol{l}) - Q_t^T Q_t v_t(\boldsymbol{l})\|^2. \tag{9}$$

Here, the Eq. 9 is minimized under constraint $Q_t Q_t^T = I$. This is solved by performing an eigenvalue decomposition of the matrix $J_t = \sum_{\boldsymbol{l}} v_t(\boldsymbol{l}) v_t(\boldsymbol{l})^T$. The rows of the project matrix Q_t is selected as the d-eigenvectors of J_t corresponding to the largest eigenvalues. Therefore, the filters are derived as:

$$\begin{aligned} \hat{B}_t^i &= \overline{Y} \hat{V}_t^i \\ \hat{\Gamma}_t &= (1 - \alpha_{cf}) \hat{\Gamma}_{t-1} + \alpha_{cf} \sum_{j=1}^{\hat{d}} \overline{\hat{F}_t^j} \hat{F}_t^j. \end{aligned} \tag{10}$$

Here, the compressed training sample $\hat{F}_t = \mathscr{F}\{Q_t f_t\}$ and target appearance $\hat{V}_t = \mathscr{F}\{Q_t v_t\}$. The template response is obtained by employing the compressed test sample $\hat{Z}_t = \mathscr{F}\{Q_{t-1} z_t\}$

$$y_{cf} = \mathscr{F}^{-1}\left(\frac{\sum_{i=1}^{\hat{d}} \overline{\hat{B}_{t-1}^i} \hat{Z}_t^i}{\hat{\Gamma}_{t-1} + \lambda}\right). \tag{11}$$

3.4 Online Detection

It is obvious that introducing a re-detection component is favorable for improving the robust long-term tracking algorithm in case of tracing failure. However, if the re-detection procedure is carried out at each frame in videos, the tracking algorithm will inevitably suffer the high computational complexity. The CCT [22] tracker utilizes the overlapping rate between the estimated target state and the candidate bounding box detected by the CUR filter to detect the tracking failure and alleviates model drift problem to some extent, but it hardly solves the problem of tracking failure due to the inaccuracy of translation estimation. We propose an effective method to tackle this problem.

Conventionally, correlation filters are designed to produce strong peaks for the target and the confidence degree of the response map is measured by

APCE [17]. To enable the tracker to detect tracking failure and activate re-detection module, we estimate the target state in the t-th frame by

$$APCE = \frac{|g_{max} - g_{min}|}{mean\left(\sum_{\boldsymbol{w}}(g_{(\boldsymbol{w})} - g_{min})\right)}, \tag{12}$$

where g_{max} and g_{min} is the maximum and minimum value on the correlation map, and $\boldsymbol{w}$ denotes the tuple index that covers all elements of the response map. We determine whether to activate re-detection module with the criteria APCE. In addition, the learning rates α_{cf} and α_{ch} are adjusted to $\kappa\alpha_{cf}$ and $\kappa\alpha_{ch}$ respectively if APCE is smaller than a predefined threshold, where κ is the penalty coefficient.

We adopt an online CUR filter which is firstly used in the collaborative correlation tracker for re-detect the target. Different from previous CCT, whether we update the online CUR filter depends entirely on APCE. Specially, the CUR decomposition algorithm [18,21] of a matrix $A \in R_{m\times n}$ aims to find a matrix $C \in R_{c\times r}$ with a subset of c columns of A, a matrix $R \in R_{r\times n}$ with a subset of r rows of A, and a low-rank matrix $U \in R_{c\times n}$ such that $\|A - CUR\|_\xi$ achieves minimum, where $\|\cdot\|_\xi$ is 2-norm or Frobenius norm. If APCE is above the threshold τ during tracking, we add the target appearance representation A_t into the historical object template pool A. We achieve the CUR filter D_t in current frame as follows

$$D_t = \frac{1}{c}\sum_{i=1,\ldots,c} C(i), \tag{13}$$

where C is a subset generated by the historical object template pool A with random sampling and the size c of C is approximately obtained by $c = \frac{2k}{\varepsilon}(1 + o(1))$, where k is the target rank and ε is the error probability. If APCE is below τ, we estimate the similarity between the CUR filter D_t and each possible candidate image regions in the image with convolution theorem to detect the top-k confident image regions. The target state is finally identified to locate at the maximum value among these response maps. Finally, the online CUR filter for detection is updated only when APCE exceeds the threshold τ.

4 Experiments and Analysis

4.1 Datasets and Experimental Setup

We evaluate our approach on the recently published benchmark which is widely used: the OTB15 datasets [20]. The datasets consist of 100 videos including many challenging situations: illumination variation, scale variation, motion blur, occlusion, etc. To fully evaluate our method, we follow the evaluation protocol as suggested in [19] as well as several standard evaluation metrics, namely distance precision (DP), overlap precision (OP) and tracking speed in frames per second (FPS). DP is defined as the percent of frames in a video where the Euclidean

distance between the estimated center location and the ground-truth of the target is below than a threshold. We present the result at the threshold of 20 pixels [19]. OP is computed as the percent of frames where the intersection-over-union overlap between the predicted bounding box and the ground-truth suppress a threshold. We report the result at the threshold of 0.5. We also provide the FPS for each tracker. In addition, the precision and success plots [19] of the results are given over all 100 videos. The precision and success plots show the mean distance and overlap precision over a range of thresholds, respectively. In the legend, the trackers are ranked using the average DP score at 20 pixels in precision plots and the area under the curve (AUC) in success plots.

Our approach is implemented in MATLAB 2015a on a desktop PC with an Intel® Core™ i7-3770 3.4 GHz CPU and 8 GB RAM. The regulation parameter is set to 0.001. The learning rates α_{cf} and α_{ch} are set to 0.01 and 0.04, respectively. We set the merge factor γ to 0.3. For online detection, we set the parameter τ to 10 to determine when to activate the detector, the penalty coefficient κ to 0.1 and the size c of template pool to 20.

4.2 Experimental Results

We compare our algorithm with 9 different state-of-the-art methods to present the excellent performance. The methods used for comparison contain Staple [1], DSST [4], SRDCF [6], CN [7], Struck [9], KCF [11], SAMF [14], LCT [15] and CCT [22]. The code or binaries for all trackers are provided by authors or the OTB datasets [20].

Table 1. Quantitative comparison of our approach with the state-of-the-art trackers on 100 challenging sequences. The results of the trackers are presented in median OP at the threshold of 0.5 and DP at threshold of 20 pixels. We also reported the average frames per second (FPS) as well. The best two results are highlighted by bold and underline. Our method performs favorably with the existing trackers.

	CN [7]	KCF [11]	Struck [9]	SAMF [14]	CCT [22]	LCT [15]	DSST [4]	SRDCF [6]	Staple [1]	Ours
OP	0.475	0.552	0.534	0.636	0.667	0.700	0.672	<u>0.728</u>	0.699	**0.774**
DP	0.595	0.697	0.655	0.740	0.739	0.762	0.696	<u>0.788</u>	0.784	**0.836**
FPS	<u>193.81</u>	**251.79**	24.37	19.30	40.12	19.43	38.6	6.99	57.63	46.05

Table 1 presents a comparison with the state-of-the-art methods above on 100 challenging sequences using OP and DP. We report the speed of the methods in average frames per second (FPS) as well. The best two results are highlighted by bold and underline in each metric. Compared to the baseline method Staple, our method improves the median OP from 69.9% to 77.4% and DP from 78.4% to 83.6%. Among the trackers in the literature, SRDCF has shown to achieve the best performance with the median OP of 72.8% and DP of 78.8%. In addition to the performance advantage with 4.6% in median OP and 4.8% in median DP, our tracker is nearly 7 times faster than SRDCF. In terms of three evaluation

metrics, our method achieves the best compared to CCT and LCT with re-detection module. Although CN and KCF obtain higher frame rate than 46.05, the proposed method is able to reach better performance with respect to them.

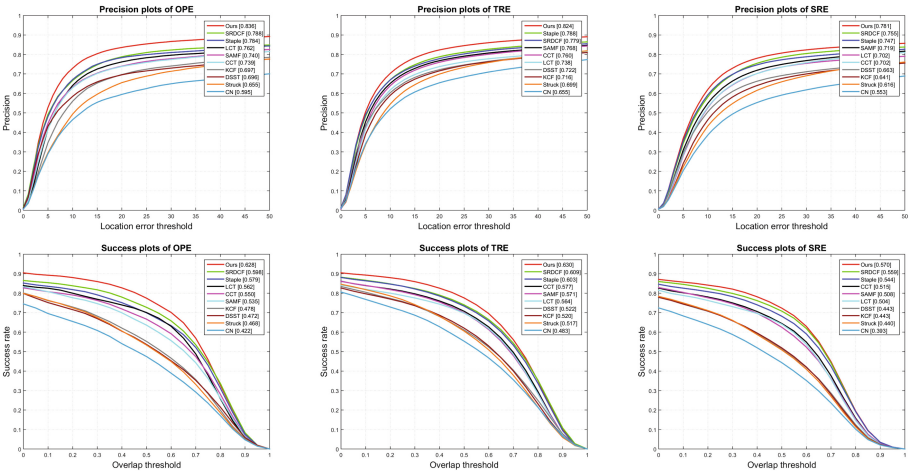

Fig. 3. The distance precision and overlap success plots using one-pass evaluation (OPE), temporal robustness evaluation (TRE) and spatial robustness evaluation (SRE) over all the 100 videos. The values indicate the mean DP score at the threshold of 20 pixels in the legend of precision plots and the legend contains the area under the curve.

Figure 3 shows the precision and success plots illustrating the median distance and overlap precision over all the 100 sequences. Our approach performs favorably against the mentioned trackers in OPE, TRE and SRE [8] evaluation schemes. The trackers are ranked using the median distance precision at the threshold of 20 pixels for precision plot and the area under the curve (AUC) for success plot. In precision plots, our method outperforms SRDCF by 4.8% and the baseline algorithm Staple by 5.2%. In success plots, our approach provides an improvement of 3% and 4.9% in AUC scores compared to SRDCF and Staple respectively. Additionally, we evaluate the robustness of our approach on two different types of initialization criteria temporal robustness (TRE) and spatial robustness (SRE). In both robustness evaluations, our algorithm achieves a consistent gain in performance compared to SRDCF and the baseline algorithm Staple.

4.3 Qualitative Evaluation

Here we provide a qualitative comparison of our approach with existing state-of-the-art trackers (Struck [9], KCF [11], SAMF [14], DSST [4] and Staple [1]) on eight challenging sequences in Fig. 4. The Struck uses the kernelized structured output SVM classifier and does not deal with well out-of-view, occlusion (Tiger2

and Jogging-2), scale variation (Dog1 and Doll), rotation (Rubick) and background clutters (Shaking). The KCF tracker based on correlation filter learned from HOG features does not perform well in scale variation (Dog1 and Doll) since it is not able to estimate scale changes. The KCF tracker fails to deal with background clusters (Shaking) because of the property of HOG feature and occlusion (Jogging-2 and Tiger2) due to the lack of the re-detection module in case of tracking failure. Although the SAMF tracker integrates HOG and CN features in the correlation filter framework, it is less effective in handling model drift problem caused by multiple factors (BlurOwl, Shaking and Diving). The Staple tracker performs well in scale variation (Dog1 and Doll) and out-of-view (Tiger2) due to complementary learners. However, it fails to effectively track the target for background clusters (Shaking). In addition, The Staple tracker leads to model drift (Jogging-2) since it does not deal with the partially or fully occlusion. Overall, our proposed approach performs remarkably in most challenging situations. The main reasons are as follows. First, complementary features are extracted to learn the template response which is combined with the color histogram response to improve performance. The feature fusion strategy exhibits powerful ability for handling fast motion and motion blur (BlurOwl and Tiger2), deformation (Diving) and background clusters (Shaking). Besides, our approach does well in scale variation (Dog1 and Doll) since we extend multiple powerful features to handle scale change. Finally, the online detector effectively activates re-detection module in case of tracking failure for out-of-view (Tiger2) and occlusion (Jogging-2).

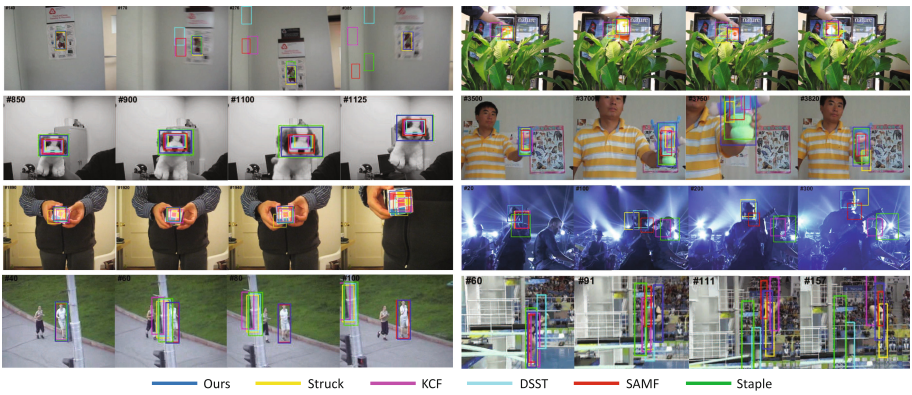

Fig. 4. A qualitative comparison of our algorithm with the five state-of-the-art trackers on eight challenging sequences (from left to right and top to down are BlurOwl, Tiger2, Dog1, Doll, Rubik, Shaking, Jogging-2 and Diving, respectively).

5 Conclusion

In this paper, we propose a robust and real-time visual tracking object method. We extend the Staple tracker by integrating the complementary features to enhance the discriminative power to illumination change. We also extend multiple features to the procedure of learning scale filter to achieve accurate scale estimation and improve robustness and efficiency by reducing feature in dimension. In addition, we employ APCE to determine whether to activate and update the CUR filter to improve robustness to tracking failure. Extensive experiments demonstrate that our method achieves superior performance in terms of accuracy, robustness and speed.

Acknowledgments. This work is supported by the National Science Foundation of China under Grant No. 61321491, and Collaborative Innovation Center of Novel Software Technology and Industrialization.

References

1. Bertinetto, L., Valmadre, J., Golodetz, S., Miksik, O., Torr, P.H.S.: Staple: complementary learners for real-time tracking. In: CVPR, pp. 1401–1409 (2016)
2. Bolme, D.S., Beveridge, J.R., Draper, B.A., Lui, Y.M.: Visual object tracking using adaptive correlation filters. In: CVPR, pp. 2544–2550. IEEE (2010)
3. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR, vol. 1, pp. 886–893. IEEE (2005)
4. Danelljan, M., Häger, G., Khan, F.S., Felsberg, M.: Accurate scale estimation for robust visual tracking. In: BMVC. BMVA Press (2014)
5. Danelljan, M., Häger, G., Khan, F.S., Felsberg, M.: Discriminative scale space tracking. TPAMI **39**(8), 1561–1575 (2017)
6. Danelljan, M., Häger, G., Shahbaz Khan, F., Felsberg, M.: Learning spatially regularized correlation filters for visual tracking. In: ICCV, pp. 4310–4318 (2015)
7. Danelljan, M., Shahbaz Khan, F., Felsberg, M., Van de Weijer, J.: Adaptive color attributes for real-time visual tracking. In: CVPR, pp. 1090–1097 (2014)
8. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. TPAMI **32**(9), 1627–1645 (2010)
9. Hare, S., Saffari, A., Torr, P.H.S.: Struck: structured output tracking with kernels, pp. 263–270 (2011)
10. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: Exploiting the circulant structure of tracking-by-detection with kernels. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7575, pp. 702–715. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33765-9_50
11. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. TPAMI **37**(3), 583–596 (2015)
12. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-learning-detection. TPAMI **34**(7), 1409–1422 (2012)
13. Khan, F.S., Van de Weijer, J., Vanrell, M.: Modulating shape features by color attention for object recognition. IJCV **98**(1), 49–64 (2012)

14. Li, Y., Zhu, J.: A scale adaptive kernel correlation filter tracker with feature integration. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8926, pp. 254–265. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16181-5_18
15. Ma, C., Yang, X., Zhang, C., Yang, M.H.: Long-term correlation tracking. In: CVPR, pp. 5388–5396 (2015)
16. Van De Weijer, J., Schmid, C., Verbeek, J., Larlus, D.: Learning color names for real-world applications. TIP **18**(7), 1512–1523 (2009)
17. Wang, M., Liu, Y., Huang, Z.: Large margin object tracking with circulant feature maps. In: CVPR (2017)
18. Wang, S., Zhang, Z.: Improving CUR matrix decomposition and the Nyström approximation via adaptive sampling. JMLR **14**(1), 2729–2769 (2013)
19. Wu, Y., Lim, J., Yang, M.H.: Online object tracking: a benchmark. In: CVPR, pp. 2411–2418 (2013)
20. Wu, Y., Lim, J., Yang, M.H.: Object tracking benchmark. TPAMI **37**(9), 1834–1848 (2015)
21. Xu, M., Jin, R., Zhou, Z.: CUR algorithm for partially observed matrices. In: ICML, pp. 1412–1421 (2015)
22. Zhu, G., Wang, J., Wu, Y., Lu, H.: Collaborative correlation tracking. In: BMVC, p. 184-1 (2015)