Neural Computing and Applications
https://doi.org/10.1007/s00521-019-04413-4

ORIGINAL ARTICLE

Multi-view correlation tracking with adaptive memory-improved update model

Guiji Li[1] · Manman Peng[1] · Ke Nai[1] · Zhiyong Li[1] · Keqin Li[1,2]

Received: 13 March 2019 / Accepted: 30 July 2019

Abstract
Recently, some researchers concentrate on applying multi-view learning to the correlation filter tracking to achieve both the efficiency and accuracy. However, most of them fail to effectively collaborate multiple views to deal with more complex environment. Moreover, their methods are prone to drift in case of long-term occlusion due to the memory loss. In this paper, we propose a novel multi-view correlation filters-based tracker for robust visual tracking. First, we present an adaptive multi-view collaboration strategy to highlight different contributions of different views by jointly considering the reliability and discrimination. Second, an effective memory-improved model update rule is introduced to avoid falling into a contaminated target model. Compared with the conventional linear interpolation update rule, it can effectively deal with long-term occlusion by improving the memory of historical models. Furthermore, instead of assigning a unified learning rate for all views in each frame, we design varying learning rates for different views according to their respective evaluations on the current tracking result, which can prevent the target models of all views from being contaminated at the same time. Finally, a failure-aware scale update scheme is developed to avoid noisy scale estimation in case of temporal tracking failure. Extensive experimental results on the recent benchmark demonstrate that our tracker performs favorably against other state-of-the-art trackers with a real-time performance.

Keywords Visual tracking · Correlation filter · Multi-view learning · Scale estimation

1 Introduction

Visual tracking is of great importance for numerous computer vision applications, such as driverless [7, 44], human–computer interaction [41, 46] and video surveillance [42, 43]. Generally, model-free online tracking is desirable. Given the only initial state in the first frame with bounding box, the task is then to determine the location and size of the object in the subsequent frames. Rare prior information makes it difficult to achieve robust tracking. Besides, significant appearance changes caused by occlusion, deformation and background clutter also complicate the problem.

Recently, discriminative correlation filter (DCF)-based visual tracking methods receive extensive attention. They show remarkable performance in terms of accuracy and speed. Accuracy is improved by using multi-channel features and approximate dense sampling scheme, while high speed is strongly benefited from exploiting the fast Fourier transform (FFT) at both detection and learning stages. Based on the conventional DCF, a variety of tracking methods are developed for the further improved performance. These improvements can be broadly divided into several aspects, including reducing boundary effect [6, 10], inducing multiple kernels and templates [3, 35], changing target response [4], etc. Although excellent results have been reported, above

✉ Manman Peng
pengmanman@hnu.edu.cn

Guiji Li
guiji.li@hnu.edu.cn

Ke Nai
naike_hnu@hnu.edu.cn

Zhiyong Li
zhiyong.li@hnu.edu.cn

Keqin Li
lik@newpaltz.edu

[1] College of Information Science and Engineering, Hunan University, Changsha 410082, Hunan, China

[2] Department of Computer Science, State University of New York, New Paltz, NY 12561, USA

Published online: 08 August 2019

Fig. 1 Comparison of the proposed method (marked with the red box) with Staple (marked with the green box) on the *lemming* sequence, where the target suffers from the long-term occlusion. Although both of two methods apply multi-view learning to the correlation filter based tracking, the proposed tracker can be able to recover the target after the long-term occlusion due to our memory-improved model update rule, which improves the memory of historical models (color figure online)

trackers suffer from expensive computational cost as they usually need to solve quite complicated models in the learning stage, which precludes their applications in real-time sceneries. Moreover, single and fixed feature representation is not always powerful enough to deal with some challenging scenes, e.g., occlusions or deformation.

In order to achieve both the efficiency and robustness, some researchers focus more on applying multi-view learning to the correlation filter tracking. Li et al. [22] propose a multi-view correlation tracker to fuse several features and select the discriminative features to do tracking. Bertinetto et al. [1] solve two independent ridge regression problems, where the inherent structure of each feature is fully leveraged. By integrating multiple complementary cues into the framework of correlation filter, these works achieve robust and real-time visual tracking. However, there still exist several factors that may affect the tracking performance. (1) Fixed multi-view collaboration strategy: On the one hand, it is not an easy thing to design a suitable weight for each view. On the other hand, since the target appearance continuously changes over time, the fixed multi-view collaboration strategy will be difficult to adapt to drastic appearance variations. (2) Loss of memory: Due to the linear interpolation update rule, the target model for each view pays more attention to recent frames and decays the effect of previous frames exponentially over time. After a period of update activity, historical models will be almost forgotten by the current target model which has been updated. Therefore, the updated model is prone to drifting away when the target suffers from long-term occlusion (as can be seen in Fig. 1). (3) Consistent or constant learning rate: *Consistent* means that the target models of all views are updated with the same learning rate. Once an inaccurate tracking result occurs, all views will be contaminated and thus increase the risk of model drift. *Constant* implies that a fixed learning rate will be employed in all frames regardless of the track quality. It is obvious that a tracker with the consistent or constant learning rate is sensitive to the inaccurate tracking result.

To address above issues, we propose a novel multi-view correlation filters-based tracker,[1] which exploits both the consensus and complementary characteristics of multiple views to achieve robust visual tracking. Specifically, we obtain multiple views from multiple features like grayscale, HOG and CN features to provide a more robust and comprehensive target representation. Multiple views capture diverse appearance characteristics of the target from different perspectives; thus, a collaboration of them is supposed to boosting tracking performance. Considering that the target object undergoes varying challenging factors over time, we present an adaptive multi-view collaboration strategy to highlight different contributions of different views in each frame based on the reliability and discrimination evaluation. Furthermore, we design an effective memory-improved model update rule with view-specific learning rate to alleviate the model drift problem. On the one hand, we improve the memory of previous target observations to prevent falling into a contaminated target model. On the other hand, learning rates are dynamically adjusted for different views according to their respective evaluations on the current tracking result, which further avoids the contamination of all views at the same time. In addition, we develop a failure-aware scale update scheme to ensure accurate scale estimation. Both quantitative and qualitative experiments on the recent benchmark have been performed to validate the superiority of the proposed tracker compared to other state-of-the-art tracking methods.

1. An adaptive multi-view collaboration strategy is presented under the framework of correlation filters. Instead of assigning fixed weights, we highlight different contributions of different views by jointly considering the reliability and discrimination in each frame.
2. An effective memory-improved model update rule is proposed to avoid falling into a contaminated target model. Different from the conventional linear interpolation update rule, the proposed method updates the

[1] Note that the concept "multi-view" in this paper is different from that in 3D video domain. Generally, in visual object tracking, multiple views indicate that multiple features are extracted to capture the different appearance characteristics of the target within a single camera view. But in 3D video domain, multiple views commonly refer to multiple camera views of the same scene.

current target model in an incremental way, which improves the memory of previous observations.
3. A varying and view-specific learning rate is designed to reduce the risk of model drift. The learning rates are dynamically adjusted for different views according to their respective evaluations on the current tracking result. With the diverse learning rates, there still exists the potential to prevent the target model of a certain view from drifting, even an inaccurate tracking result occurs.
4. A failure-aware scale update scheme is developed. To avoid inaccurate scale estimation in case of failing translation estimation, we will disable the target scale estimation module once the underlying tracking failure is detected.

In the reminder of the paper, we firstly discuss the most related works to ours in Sect. 2. Then, we give a detail introduction of our multi-view correlation filters, multi-view collaboration strategy, online update model and failure-aware scale update scheme in Sect. 3. Experimental results and discussions can be found in Sect. 4. Finally, we conclude our work in Sect. 5.

2 Related works

Visual tracking [1, 10, 17, 19, 23, 25, 29, 30, 53] is a hot topic in the field of computer vision. A comprehensive review of visual tracking methods can be found in [21, 32, 39]. In this section, we only discuss the most related works to ours including correlation tracking and multi-view learning.

2.1 Correlation tracking

In recent years, correlation filters have attracted extensive attention in visual tracking due to their high-speed performance. Bolme et al. [5] model the target appearance by learning a minimum output sum of squared error (MOSSE) filter and perform tracking by correlating this filter over a search window. Henriques et al. [16] take advantage of the circulant structure of samples to solve kernel regularized least squares problem in Fourier domain. The work is further extended in KCF tracker [17] with HOG feature, which achieves the amazing performance in terms of both accuracy and efficiency. Danelljan et al. [8] propose a discriminative scale space tracker (DSST) to incorporate scale estimation in the DCF-based tracking framework. In [33], an anisotropy filter response is exploited instead of Gaussian-shaped response to promote the robustness against occlusion. To reduce the boundary effect, Danelljan et al. [10] present a novel correlation tracking method named as spatially regularized discriminative correlation filters (SRDCF) with penalizing correlation filter coefficients. Ma et al. [27] design a long-term correlation tracking framework to re-detect the target object in case of tracking failure with a detection module. Dong et al. [11] develop an occlusion-aware visual tracking algorithm to avoid the drifting problem caused by noisy updates. Li et al. [20] propose a distortion-aware correlation tracking framework, which boosts the tracking performance by solving the distortion problem in correlation filter based methods. Zhang et al. [52] exploit and complement the strength of correlation filters and particle filters to achieve more robust tracking performance. In [38], both the global and local information are considered into the correlation filters-based tracking, which significantly improves the robustness of the tracking accuracy. Sun et al. [34] carefully check the current track quality of translation correlation filters via a reverse evaluation strategy. A relocation mechanism will be activated to refine the tracking result once the track quality is low. However, these DCF-based methods mostly pay more attention to design a sophisticated model in the learning stage of the whole tracking process with a high computational cost. In this paper, we focus more on the detection and updating stage of the correlation tracking paradigm to achieve a real-time performance.

2.2 Multi-view learning

Multi-view learning [12, 45] is a promising paradigm to improve the tracking performance with complementary cues. How to effectively fuse multiple features to realize more robust tracking performance attracts wide attention. Zhang et al. [51] propose a novel entropy criterion-based collaborative strategy, which determines the weight of each support vector machine (SVM) by evaluating the uncertainty of the corresponding probability distribution. Similarly, entropy criterion is also employed to measure the discriminative ability of one feature in [28]. Different from the work [51], Ma et al. [28] concentrate on the combined evaluation rather than independent evaluation of each feature. Moreover, they design an iterative method to optimize the weighted entropy-based objective function. Yoon et al. [47] establish multiple trackers based on different feature representations and perform tracker interaction and selection within a probabilistic framework. Notwithstanding good performance is reported in terms of accuracy, above methods have high computation cost. To realize an effective and efficient tracker, some researchers apply multi-view learning to the correlation filters. Li et al. [23] propose a scale-adaptive kernel correlation filter tracker by directly concatenating multiple features. Li et al. [22] present a multi-view correlation filters tracker to do

tracking. Kullback–Leibler (KL) divergences between the distribution of each view and the fused distribution is minimized to obtain the fused probability distribution. Bertinetto et al. [1] develop a novel tracker called Staple, which combines the scores of both template model and color statistics models. Zhang et al. [50] jointly train an ensemble of correlation filters with multiple views to perform tracking via an efficient cotrained model. In [26], the channel reliability score is estimated for weighting per-channel in the stage of detection. However, these multi-view learning-based tracking methods either exploit fixed multi-view collaboration strategy or introduce the expensive computational cost without using efficient correlation filters. Considering that both the robustness and efficiency play important roles in visual tracking, we apply multi-view learning to the correlation filter tracking with an adaptive multi-view collaboration strategy. Moreover, we design view-specific learning rates to reduce the risk of model drift.

Algorithm 1 The Proposed Tracking Algorithm

Input:
Image I_t;
Previous target position p_{t-1} and scale s_{t-1};
Target model A^v_{t-1}, D^v_{t-1} for each view $v \in \{g, c, h\}$.
Output:
Current target position p_t and scale s_t;
Updated model A^v_t, D^v_t for each view $v \in \{g, c, h\}$.

sample the patch z_t at p_{t-1} with scale s_{t-1} from I_t.
while $v \in \{g, c, h\}$ **do**
  Extract the feature from z_t.
  Compute the response map r^v_t using Eq. (3).
  Compute the appearance similarity S_a and spatial simlarity S_s using Eq. (4) and Eq. (5).
  Compute the reliability $\zeta^v_{rel,t}$ by combining S_a and S_s via Eq. (6).
  Compute the discrimination $\zeta^v_{dis,t}$ using Eq. (7).
  Compute the weight w^v_t based on $\zeta^v_{rel,t}$ and $\zeta^v_{dis,t}$ using Eq. (8).
end while
Obtain the fused response map r_t by collaborating multi-view response maps with w^v_t using Eq. (9).
Get the target position p_t by maximizing r_t.
Check the tracking failure marker m_t using Eq. (12) and estimate the target scale s_t if $m_t = 0$.
while $v \in \{g, c, h\}$ **do**
  Extract the feature from the new sampled patch z^*_t at p_t with scale s_t.
  Compute the learning rate η^v_t using Eq. (11)
  Update the current target model A^v_t, D^v_t with η^v_t using Eq. (10).
end while

3 Our method

In this section, we first review the typical discriminative correlation filters formulation. Next, we carefully introduce our adaptive multi-view collaboration strategy, which highlights different contributions of different views by jointly considering the reliability and discrimination. Then, we elaborate the proposed memory-improved online update model. Furthermore, a varying and view-specific learning rate is designed and analyzed in detail. Last but not least, a failure-aware scale update scheme is developed to avoid inaccurate scale estimation. The whole process of our method is shown in Fig. 2 and summarized in Algorithm 1.

3.1 Discriminative correlation filters

Let f denote an image patch consisting of d feature channels. The aim of the typical DCF formulation is to learn a multi-channel discriminative correlation filter h from this single patch f, which is centered around the target object. This is achieved by minimizing the sum of squared error ε between the actual correlation response and the desired correlation response g,

$$\varepsilon = \left\| g - \sum_{l=1}^{d} h^l \star f^l \right\|_2^2 + \lambda \sum_{l=1}^{d} \left\| h^l \right\|_2^2, \tag{1}$$

where, $\star$ stands for circular correlation, $\lambda(\lambda \geq 0)$ is a regularization parameter which controls overfitting. Typically, the desired correlation response g is selected to be the Gaussian function, which implies that the correlation response output at location n is larger when the corresponding sample is closer to the target object. *Convolution Theorem* tells us that the correlation in time domain corresponds to the element-wise multiplication in Fourier

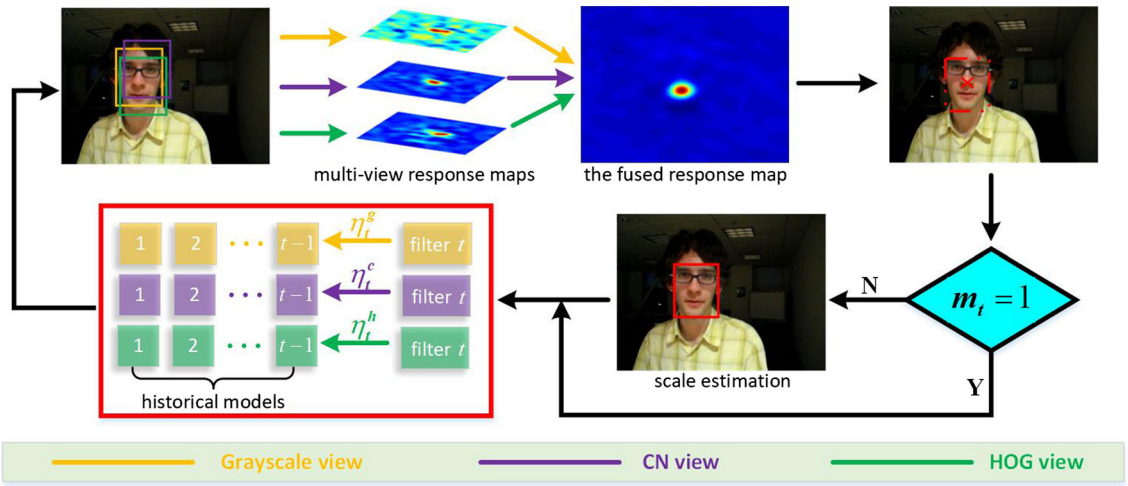

Fig. 2 The flowchart of the proposed tracking method. When a new frame arrives, we respectively calculate the response map r^v for each view $v \in \{g, c, h\}$ with the corresponding correlation filter h^v. Then we adaptively combine the response maps of all views based on the reliability and discrimination measurement. The target location is finally determined by searching for the maximal value of the fused response map r. As for the target scale, we activate the scale estimation module when $m_t = 0$. In the updating stage, we incrementally update the current target model with a new learned model for each view. Moreover, the learning rates are dynamically adjusted for different views according to their respective evaluations on the current tracking result

domain. Therefore, Eq. (1) can be efficiently minimized in the Fourier domain and the $l-$th channel of filter H is given by

$$H^l = \frac{G^* \odot F^l}{\sum_{k=1}^{d} (F^k)^* \odot F^k + \lambda}, \quad l = 1, \ldots, d, \tag{2}$$

where, $*$ refers to complex conjugation, $\odot$ is the element-wise multiplication and the fraction denotes the element-wise division. In addition, the capital letters denote the discrete Fourier transform (DFT) of the corresponding quantities. As for the derivation of Eq. (2), we refer readers to [8] for more detail.

When a new frame t arrives, the response map r_t can be computed with an image patch z_t (centered at the predicted target location) and the correlation filter H^l_{t-1} updated in the previous frame,

$$r_t = \mathcal{F}^{-1}\left(\frac{\sum_{l=1}^{d} \left(A^l_{t-1}\right)^* \odot Z^l_t}{D_{t-1} + \lambda}\right), \tag{3}$$

where $\mathcal{F}^{-1}$ represents the inverse DFT, A^l_{t-1} and D_{t-1} are the numerator and denominator of H^l_{t-1}. Then, the target can be located by searching for the peak of the response map r_t.

3.2 Multi-view collaboration strategy

Multi-view learning is helpful for accurately locating the target object since different features can provide complementary information to deal with more complex environment. Here, we employ three kinds of complementary features, i.e., grayscale, color names (CN), HOG feature to represent the object appearance. Grayscale feature provides a basic description of the object. CN feature is an important visual cue to distinguish the target from the background, while HOG feature shows its superiority in case of nondistinctive color distributions between the target and the background. For each view $v \in \{g, c, h\}$, we respectively establish an independent correlation filter h^v. These filters can work in parallel to calculate the corresponding response map r^v. The final response map r_t in the t frame can be obtained by fusing r^v_t with consideration of two important factors of each view: reliability and discrimination.

Reliability Reliability reveals the accuracy of the tracking result. Based on the tracking smooth assumption, we exploit the appearance similarity and the spatial similarity between two consecutive frames to measure the reliability for each view. Since the change between two consecutive frames is little, the more reliable tracking result is supposed to approach the target in the last frame in terms of the appearance and location. To this end, we first calculate the center position c^v_t of the tracking result for each view and extract the corresponding image patch p^v_t. As a single image patch is usually used to represent the target

appearance, the appearance similarity between the patch p_t^v and patch p_{t-1} can be defined as

$$\begin{aligned} S_a\left(p_t^v, p_{t-1}\right) &= \exp\left(\mathrm{NCC}\left(p_t^v, p_{t-1}\right)\right) \\ &= \exp\left(\frac{\sum_{i,j}\left(p_t^v(i,j)-\overline{p_t^v}\right)\left(p_{t-1}(i,j)-\overline{p_{t-1}}\right)}{\sqrt{\sum_{i,j}\left(p_t^v(i,j)-\overline{p_t^v}\right)^2\sum_{i,j}\left(p_{t-1}(i,j)-\overline{p_{t-1}}\right)^2}}\right). \end{aligned} \tag{4}$$

Here, $\overline{p_t^v}$ and $\overline{p_{t-1}}$ denote the mean value of matrix p_t^v and p_{t-1}. $\mathrm{NCC}\left(p_i, p_j\right)$ means the normalized correlation coefficient between p_i and p_j, which is usually used as a kind of similarity measurement between two image patches. However, the appearance similarity always becomes unreliable when the background exhibits a similar appearance compared to the target, so it is necessary to impose the distance constraint by inducing the spatial similarity. We exploit the Euclidean distance to measure the spatial similarity between the current tracking result of view v and the target in the last frame as follows,

$$S_s\left(c_t^v, c_{t-1}\right) = \exp\left(-\frac{\left\|c_t^v - c_{t-1}\right\|_2^2}{\delta^2}\right), \tag{5}$$

where, δ is a parameter which controls the steepness of the exponential function. Accordingly, the reliability of each view can be measured by combining the appearance similarity and the spatial similarity between two consecutive frames,

$$\zeta_{\mathrm{rel},t}^v = S_a\left(p_t^v, p_{t-1}\right) \cdot S_s\left(c_t^v, c_{t-1}\right). \tag{6}$$

The higher $\zeta_{\mathrm{rel},t}^v$ means the better reliability of the tracking result of view v.

Discrimination Besides reliability, a good view should have enough discriminative ability to distinguish the target object from the background. The peak-to-sidelobe ratio (PSR) measures the strength of a peak relative to the sidelobe, which can be used to evaluate the discrimination of view v in the t frame,

$$\zeta_{\mathrm{dis},t}^v = \frac{\max\left(r_t^v\right) - \mu_{\mathrm{s}}\left(r_t^v\right)}{\sigma_{\mathrm{s}}\left(r_t^v\right)}, \tag{7}$$

where, μ_{s} and σ_{s} are the mean value and standard deviation of the sidelobe area, which is defined as the response map area excluding a given window (set to 15% of response map in this paper) around the peak. From Eq. (7), we can observe that making the PSR larger needs to satisfy two conditions: (1) the peak which refers to the response value of the target object should be stronger relative to the mean response value of the sidelobe; (2) response values of the sidelobe (here refers to the response values of other samples) should be stable at a low level as much as possible. These two conditions indicate that the PSR is larger when the determination from a view is less ambiguous. Therefore, PSR can be treated as a reasonable metric to measure the discriminative ability of a view.

We argue that a view which is both reliable and discriminative should be assigned larger weight during the process of the target location. Therefore, we define the weight assignment function,

$$w_t^v = (1-\gamma)\zeta_{\mathrm{rel},t}^v + \gamma\zeta_{\mathrm{dis},t}^v, \tag{8}$$

where, γ is a trade-off between the reliability and the discrimination. Then, we calculate the final response map r_t in the t frame by a linear combination of the response maps of all views,

$$r_t = \sum_{v\in\{g,c,h\}} w_t^v r_t^v. \tag{9}$$

Note that each view has its own specialty and advantage in dealing with varying challenging factors, and this should be fully explored and applied in their collaboration. With the above linear weighted function, the collaboration of multi-view correlation filters can be well performed by considering different contributions of different views in the fusion of response maps.

3.3 Online model update

It is necessary to update the target model to adapt to appearance changes. The most commonly used linear interpolation update rule pays more attention to recent models, thereby ignoring the important historical information. We insist that all previous target models have much importance on the determination of the target location. To improve the memory of this important information, we incrementally update the current target model with a new learned model for each view as follows,

$$\begin{aligned} A_t^{v,l} &= A_{t-1}^{v,l} + \eta_t^v G^* \odot F_t^{v,l} \\ D_t^v &= D_{t-1}^v + \eta_t^v \sum_{k=1}^{d}\left(F_t^{v,k}\right)^* \odot F_t^{v,k}. \end{aligned} \tag{10}$$

Here, η_t^v is a variable which denotes the learning rate of view v in the t frame. Since the model learned from the first frame is absolutely reliable, we assign a larger learning rate for it. As for other new learned models, a common approach is to update the models of all views based on the measurement of the current tracking quality. Once the measurement is inaccurate, the target models of all views will either be contaminated with a noisy tracking result or struggle to adapt to drastic appearance variations. In contrast to above design, we dynamically adjust the learning rates for different views according to their respective

evaluations on the current tracking result. If a view is confident enough about its decision and shows good agreement with the current tracking result, it will consider a good quality track. Then, new corresponding model will be much learned. Otherwise, small learning rate should be set. Based on the above analysis, we design the varying learning rates in the t frame for different views as follows,

$$\eta_t^v = \begin{cases} \eta_0, & \text{if } t = 1, \\ L \cdot \exp\left(-\left(1 - c_t^v \cdot s_t^v\right)^2\right), & \text{otherwise}, \end{cases} \tag{11}$$

where η_0 is the learning rate for the first frame that does not depend on v and t. L is a constant learning factor. c_t^v measures the confidence of view v in the t frame, which can be defined as the peak of r_t^v, i.e., $c_t^v = \max\left(r_t^v\right)$. We denote s_t^v as the consistency score and calculate it using the overlap ratio between the bounding boxes of the current tracking result and the result of view v: $s_t^v = \frac{\left|B_t^v \cap B_t\right|}{\left|B_t^v \cup B_t\right|}$.

3.4 Scale estimation

Accurate scale estimation plays an important role in robust visual tracking. Similar to [8], we learn a one-dimensional scale correlation filter to estimate the scale changes of the target. Let $P \times Q$ be the current target size and S indicate the number of the possible target size. We sample S image patches around the target and denote the size of each patch as $sP \times sQ$, where $s \in \left\{a^{\left\lfloor -\frac{S-1}{2} \right\rfloor}, a^{\left\lfloor -\frac{S-3}{2} \right\rfloor}, \ldots, a^{\left\lfloor \frac{S-1}{2} \right\rfloor}\right\}$, a denotes the scale factor. Then, all these patches are normalized to a uniform size to learn the scale correlation filter. The final target size is determined with the maximal correlation response. However, directly updating the scale of the target is risky when a temporal tracking failure occurs. Inaccurate appearance representation will have a negative impact on the scale estimation. And the noise in scale will also contaminate the target translation model, which makes the matter worse. Therefore, it is necessary to detect the potential tracking failure before estimating the target scale. We define a tracking failure marker m_t using the consensus characteristics of multiple views,

$$m_t = \begin{cases} 0, & \frac{\left|B_t^{v_1} \cap B_t^{v_2}\right|}{\left|B_t^{v_1} \cup B_t^{v_2}\right|} > \tau, \\ 1, & \text{otherwise} \end{cases} \tag{12}$$

where, v_1 and v_2 are any two of different views, τ is a threshold which decides the current tracking state. Once the tracking failure is detected in the t frame, i.e., $m_t = 1$, the estimation of the target scale will be stopped.

4 Experimental evaluation

In this section, we evaluate the proposed method with extensive experiments on two challenging benchmark datasets, they are OTB2013 [39] and its updated version OTB2015 [40], respectively. Both quantitative and qualitative comparisons are conducted with other state-of-the-art methods. One-pass evaluation (OPE) criterion is used in all experiments.

4.1 Implemental details

The proposed tracking algorithm is implemented in Matlab and runs at 30 frames per second (meets the real-time demand) on a PC machine with Intel Core i7-6700HQ CPU 2.6 GHz and 8 G memory. The parameters used in this work are fixed in all experiments and set as follows. All related parameters in correlation filters keep the same in work [8]. The parameter $\delta^2 = 200$. The trade-off parameter γ between the reliability and the discrimination in Eq. (8) is 0.8. The learning rate η_0 for the first frame and the learning factor L in Eq. (11), respectively, set to 1.1 and 0.7. The threshold τ in Eq. (12) is set to 0.5.

4.2 Evaluation metrics

Two widely used metrics are exploited to provide a reasonable evaluation in this section: (1) precision plot, which reflects the percentage of frames whose center location error (CLE) is within a given threshold. CLE is obtained by the average Euclidean distance between the predicted target location and the ground-truth. In the precision plot, we refer the result at error threshold of 20 as distance precision (DP) and use it for comparisons; (2) success plot, which shows the ratios of successfully tracked frames whose overlap scores suppress a given threshold. The overlap score is defined as the overlap $O\left(r_g, r_p\right) = \frac{\left|r_g \cap r_p\right|}{\left|r_g \cup r_p\right|}$ between the ground truth bounding box r_g and predicted bounding box r_p. In the success plot, area under curve (AUC) is usually adopted for ranking performance.

4.3 Comparisons on OTB2013 benchmark

OTB2013 benchmark is a public challenging dataset with 51 video sequences. These sequences contain 11 challenging factors including IV (Illumination variation), SV (Scale variation), OCC (Occlusion), DEF (Deformation), MB (Motion blur), FM (Fast motion), IPR (In-plane rotation), OPR (Out-of-plane rotation), OV (Out of view), BC (Background clutters) and LR (Low resolution). Next, we

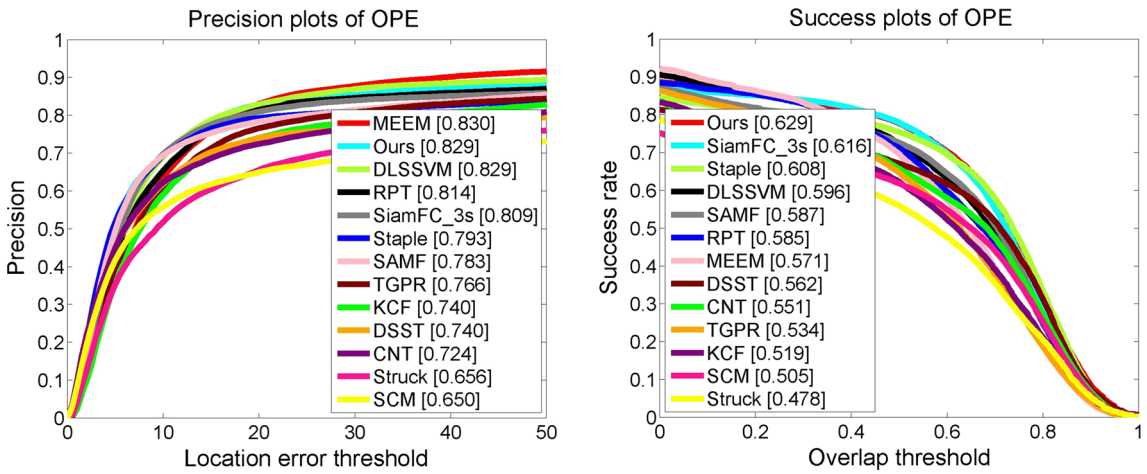

Fig. 3 The overall performance on the OTB2013 benchmark. All trackers are ranked based on the DP scores in the legend of the precision plots and the AUC scores in the legend of the success plots

will report the experimental results in terms of overall performance and attribute-based performance.

4.3.1 Evaluated methods

We evaluate our tracker with 39 state-of-the-art tracking methods. They are MEEM [48], DLSSVM [31], Staple [1], CNT [49], RPT [24], TGPR [13], KCF [17], DSST [8], SAMF [23], SiamFC_3s [2], SCM [53], Struck [14] and other 27 excellent trackers which are listed in the benchmark [39]. Among these trackers, Staple, RPT, KCF, DSST and SAMF are several correlation filters based tracking algorithms. CNT and SiamFC_3s are developed with the framework of deep learning. MEEM, DLSSVM, Struck, SCM and TGPR are other different types of tracking methods. Note that we only show the most 12 competitive methods and our method in following comparisons for clarity.

4.3.2 Quantitative comparison

Overall performance Figure 3 shows the precision plots and success plots of the overall performance. All trackers are ranked by the DP scores and the AUC scores in the legend, respectively. In the precision plots, our tracker achieves the second performance with an average DP of 82.9%, which is very close to the best performing tracker MEEM. Although the performance difference exists in DP, our tracker achieves much higher speed than MEEM tracker which loses the real-time performance. In the success plots, the proposed method ranks the first and performs 1.3% better than SiamFC_3s. Note that Staple and SAMF are two correlation filters-based methods which also exploit multiple features for improving performance. But we can observe that our tracker outperforms them by a large margin. The underlying reason for the significant performance improvement is that we consider the different contributions of different features. In summary, the precision plots and success plots demonstrate that the proposed method performs favorably against other state-of-the-art trackers.

Attribute-based performance It is valuable to evaluate the performance of trackers from different perspectives. Attribute-based performance evaluation can reflect the robustness of one tracker against different challenging factors. Figures 4 and 5 illustrate the precision plots and success plots of our method and other state-of-the-art methods in different attributes, respectively. The proposed algorithm ranks within top 3 on 7 out of 11 attributes in precision plots and on 8 out of 11 attributes in success plots. Specially, our method achieves much excellent performance on the sequences with attributes "occlusion." "out of view" and "low resolution." For the sequences with attribute "low resolution," our method achieves the DP of 66.9% and AUC of 53.0%, which ranks the first in both two evaluation metrics. It's worth mentioning that our method obtains significant improvements on the sequences with attribute "occlusion" compared to other state-of-the-art trackers. The reason can be attributed to the memory-improved model update rule developed in this paper, which improves the memory of historical models to deal with model drift.

4.3.3 Qualitative comparison

We show some sampled tracking results of our method over 7 challenging sequences in Fig. 6. Besides, the tracking results of several state-of-the-art trackers (e.g., KCF, CNT, SCM and Staple) are also presented for comparisons.

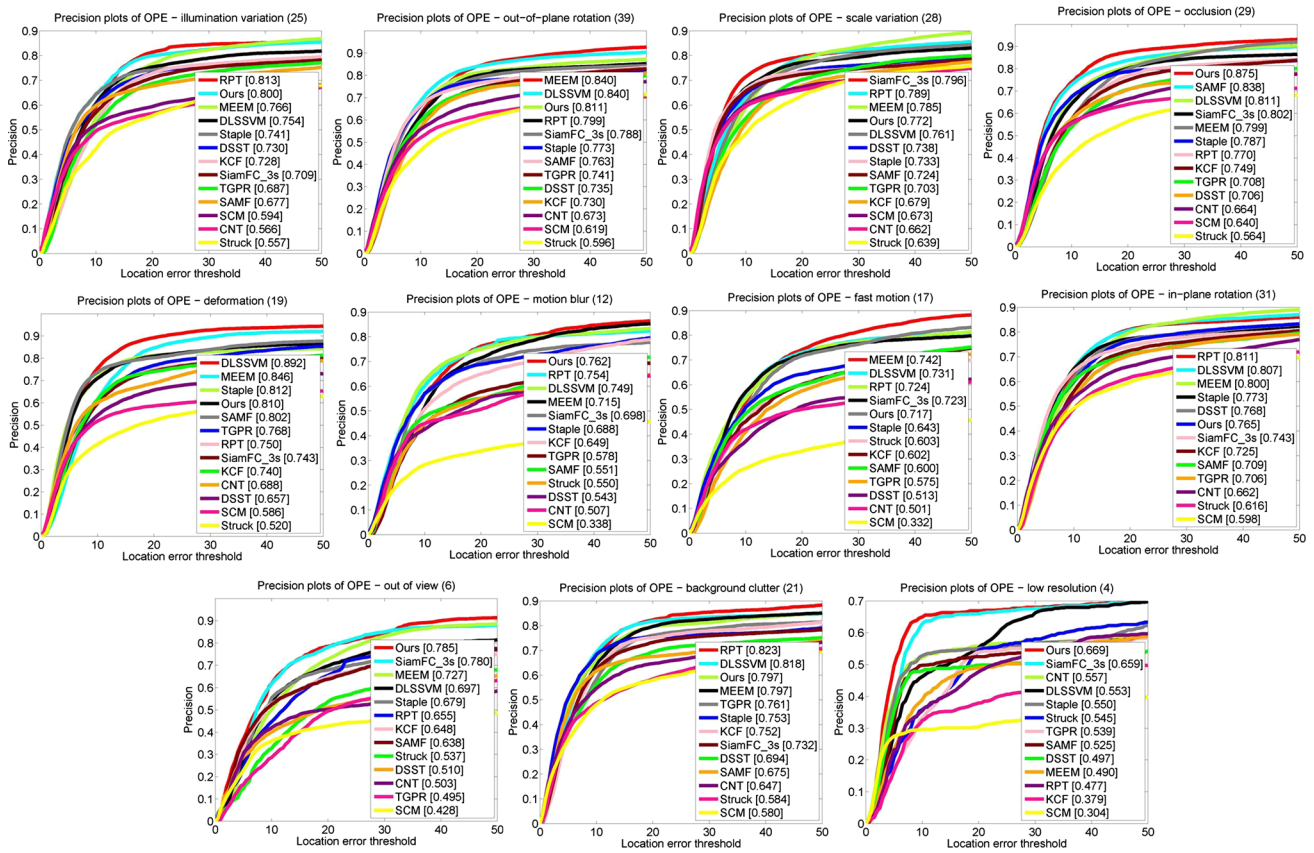

Fig. 4 Attribute-based performance evaluation on the OTB2013 benchmark with precision plots. The number of sequences for each attribute is shown in the title

Background clutter It is difficult to deal with the background clutter encountered in the *Ironman* and *Soccer* sequences, where the background near the targets has the similar color or texture as the targets through the whole process. In the *Ironman* sequence, only our tracker can track the target object well. Other trackers drift away to the background at the beginning of the sequence. In the *Soccer* sequence, the CNT, SCM and Staple trackers lose track of the target after frame 240. Although the KCF tracker can lock on the correct target at all frames, it does not handle scale variations well. Overall, only our method performs well in terms of accuracy and robustness in this sequence.

Occlusion Target objects suffer from heavy occlusions in the *Girl*, *Lemming* and *Jogging-1* sequences. In the *Girl* sequence, the KCF and Staple trackers lose the target when the head of the girl is occluded by the man at frame 433. The CNT, SCM and our trackers can track the target from the beginning to the end. In the *Lemming* sequence, the target hides behind the background for a long time (from frame 340 to frame 375). When the target reappears, only the CNT and our trackers can re-detect the target accurately. In the *Jogging-1* sequence, the target is almost fully occluded by lamp post at frame 68. The KCF, SCM and Staple trackers drift away to the background at frame 80, whereas the CNT and our trackers perform well at all frames.

Out-of-plane rotation In the *Basketball* and *Tiger2* sequences, the target objects undergo significant appearance variations like out-of-plane rotation. The KCF and our trackers achieve the best performance in the *Basketball* sequence, while the CNT, SCM and Staple trackers gradually drift away from the target at frame 62 and frame 641. In the *Tiger2* sequence, the Staple and our trackers perform better than the CNT, SCM and KCF trackers with a stable track.

4.3.4 Comparison with deep learning trackers

To provide a more comprehensive experimental evaluation of our method, we compare the proposed tracker with other 7 deep learning trackers on the OTB2013 benchmark in this section. The comparison results are shown in Table 1 with DP scores and AUC scores. From the results, we find that our method achieves satisfactory performance among all compared deep learning trackers. Note that our method only exploits hand-craft features such as grayscale, HOG

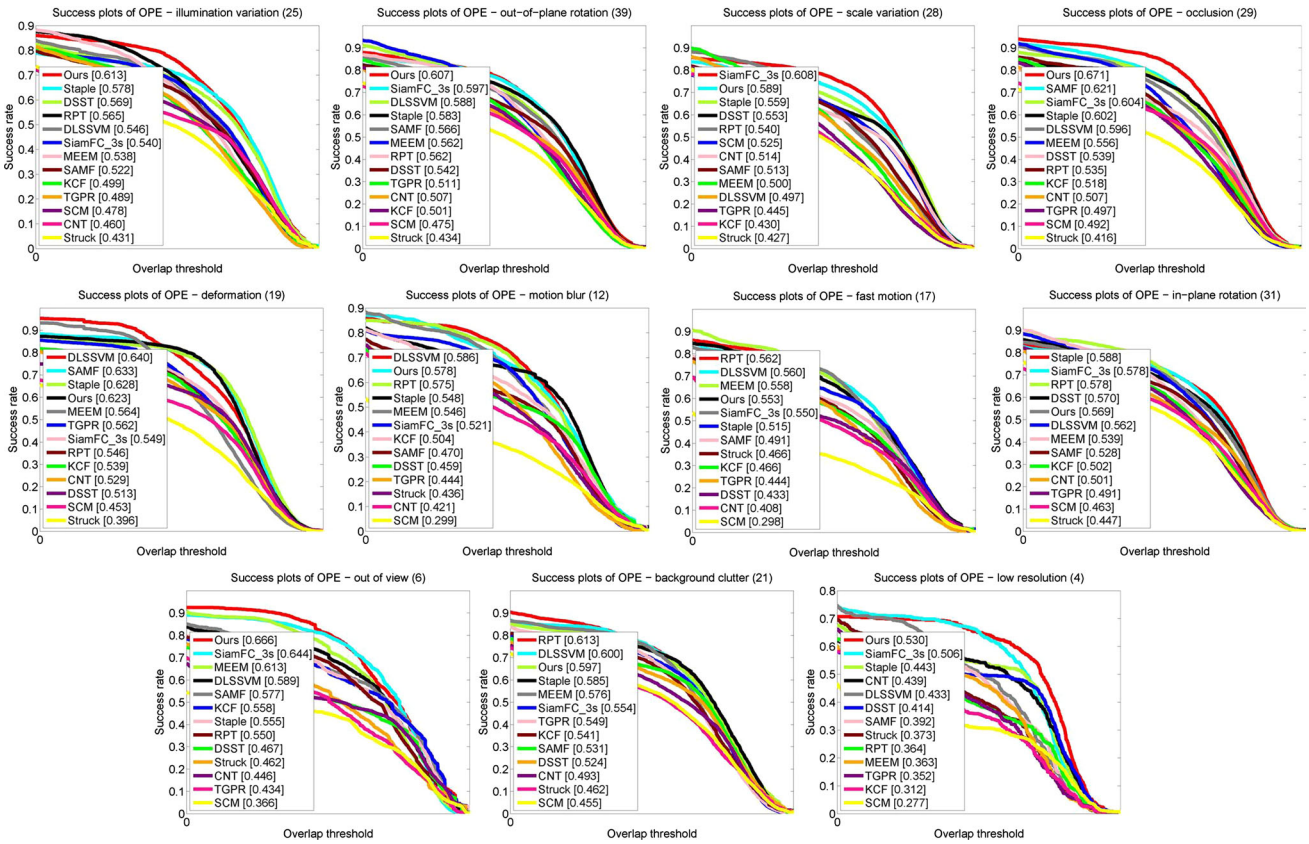

Fig. 5 Attribute-based performance evaluation on the OTB2013 benchmark with success plots. The number of sequences for each attribute is shown in the title

and CN features. Although deep learning trackers like DeepSRDCF and CNN-SVM using powerful deep features are more robust to occlusions and deformations, they suffer from expensive computational cost.

4.4 Comparisons on OTB2015 benchmark

OTB2015 benchmark is an updated version of the OTB2013 benchmark with 100 video sequences. Similarly, OTB2015 benchmark is also annotated with 11 attributes. Due to the space limitation, we only list the DP scores and the AUC scores of the overall performance and attribute-based performance of all evaluated methods.

4.4.1 Evaluated methods

We evaluate the proposed methods with 9 competing trackers and other 29 state-of-the-art trackers. The 9 competing trackers are MEEM [48], LCT [27], DLSSVM [31], RPT [24], KCF [17], DSST [8], SAMF [23], SCM [53] and Struck [14]. Other 29 baseline trackers are listed in [40]. For clarity, only the most 9 compared methods and our method are presented.

4.4.2 Quantitative comparison

Overall performance We show the DP scores and the AUC scores of the overall performance in Table 2. It can be observed that our tracker ranks second in the DP scores, which is slightly lower than the best performing tracker MEEM. While in the AUC scores, our method achieves the best performance among all evaluated methods. We can also observe that MEEM and DLSSVM perform better in the DP scores than in the AUC scores. This is because they are not equipped with the scale estimation. Overall, the proposed method performs well against other state-of-the-art trackers.

Attribute-based performance Tables 3 and 4 summarize the comparison results in terms of the DP scores and the AUC scores of attribute-based performance. The proposed method ranks within top 3 on 8 out of 11 attributes in the DP scores and the first on almost all of attributes in the AUC scores, respectively. As can be seen in Tables 3 and 4, our tracker performs well in most challenging scenes, especially in case of illumination variation, occlusion, out of view and low resolution. On the low resolution sequences, our method has absolute advantages with the

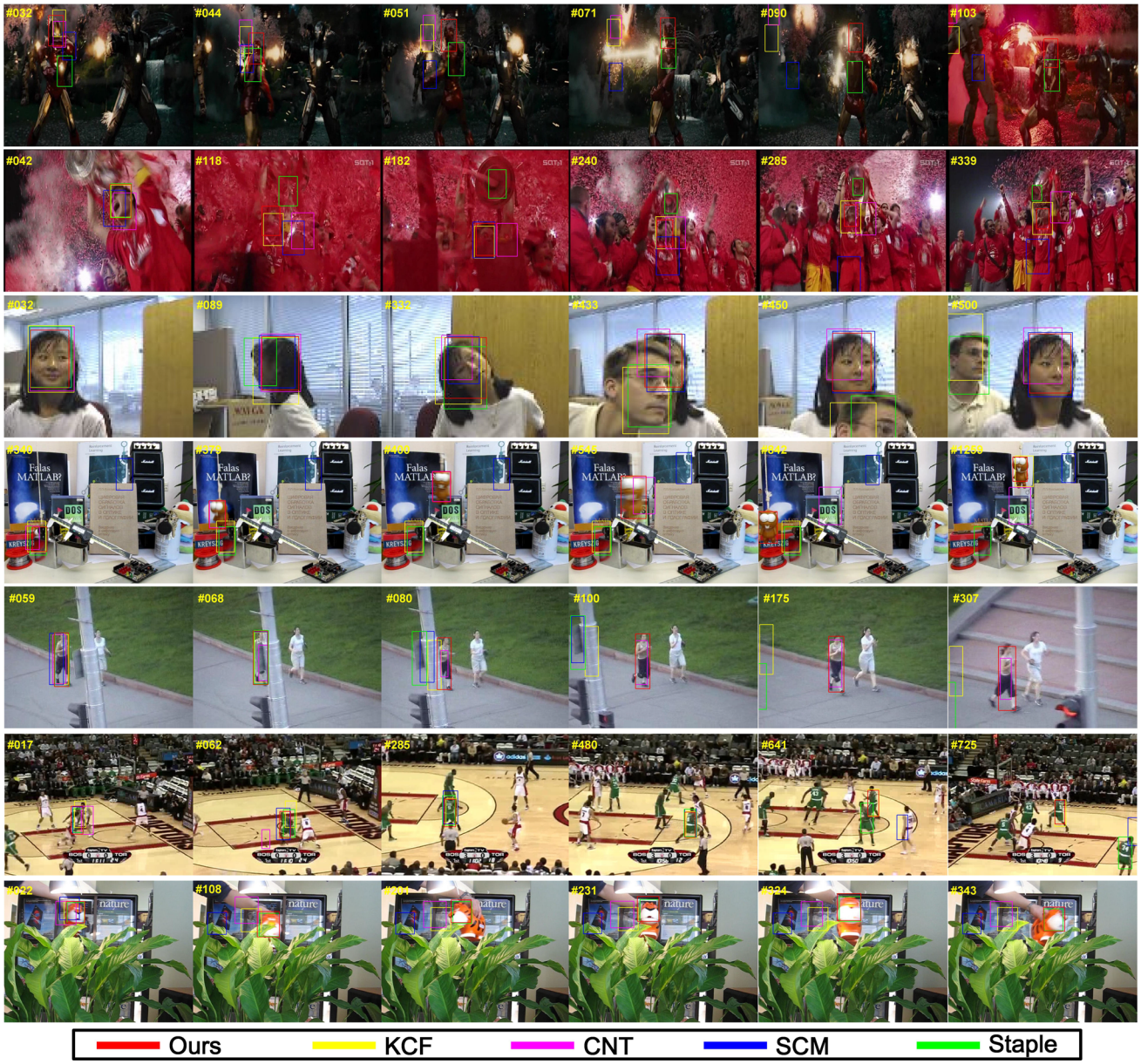

Fig. 6 Some sampled tracking results of 5 trackers in the *Ironman*, *Soccer*, *Girl*, *Lemming*, *Jogging-1*, *Basketball*, *Tiger2* sequence

Table 1 The DP scores and AUC scores of the proposed method and some deep learning trackers on the OTB2013 benchmark

Trackers	SiamFC_3s [2]	CFnet [36]	SO-DLT [37]	CNT [49]	Goturn [15]	DeepSRDCF [9]	CNN-SVM [18]	Ours
DP	0.809	0.807	0.810	0.724	0.620	**0.849**	*0.852*	***0.829***
AUC	***0.616***	0.611	0.595	0.551	0.444	*0.641*	0.597	**0.629**

The best performance is marked with the italic fronts, the second best performance is marked with the bold fronts, the third best performance is marked with the bold italic fronts

DP of 77.6% and the AUC of 50.0%, which significantly outperforms the second best method by 9.7% and 6.5%. These advantages benefit from the proposed multi-view collaboration strategy, which effectively fuses multiple features to provide more rich information. On the out of view sequences, our tracker and MEEM achieve impressive performance. It can be attributed to the use of historical models. Even if the target leaves the view, our

Table 2 The DP scores and AUC scores of the proposed method and other state-of-the-art trackers on the OTB2015 benchmark

Trackers	SCM [53]	Struck [14]	KCF [17]	DSST [8]	SAMF [23]	RPT [24]	MEEM [48]	LCT [27]	DLSSVM [31]	Ours
DP	0.570	0.640	0.691	0.679	0.749	0.756	*0.781*	0.761	***0.763***	**0.771**
AUC	0.451	0.465	0.479	0.519	***0.557***	0.542	0.534	**0.569**	0.544	*0.582*

The best performance is marked with the italic fronts; the second best performance is marked with the bold fronts; the third best performance is marked with the bold italic fronts

Table 3 The DP scores of the proposed method and other state-of-the-art methods in different attributes on the OTB2015 benchmark

Trackers	SCM [53]	Struck [14]	KCF [17]	DSST [8]	SAMF [23]	RPT [24]	MEEM [48]	LCT [27]	DLSSVM [31]	Ours
IV	0.608	0.557	0.713	0.715	0.702	*0.806*	***0.746***	0.743	0.727	**0.783**
OPR	0.575	0.603	0.677	0.662	0.750	0.742	*0.812*	0.768	***0.780***	**0.785**
SV	0.570	0.614	0.640	0.650	0.713	***0.723***	*0.756*	0.698	0.719	**0.732**
OCC	0.574	0.551	0.630	0.610	***0.739***	0.694	**0.768**	0.704	0.730	*0.772*
DEF	0.562	0.549	0.627	0.555	0.697	***0.735***	*0.786*	0.715	**0.746**	0.698
MB	0.275	0.584	0.598	0.565	0.643	0.683	*0.729*	0.667	**0.728**	***0.711***
FM	0.321	0.638	0.629	0.575	0.669	***0.728***	*0.779*	0.713	**0.732**	0.706
IPR	0.537	0.633	0.693	0.691	0.717	0.745	*0.794*	**0.781**	***0.776***	0.744
OV	0.426	0.482	0.494	0.477	0.619	0.580	*0.681*	0.587	***0.621***	**0.668**
BC	0.582	0.559	0.712	0.703	0.686	*0.789*	***0.746***	0.734	0.728	**0.764**
LR	0.596	0.666	0.554	0.561	**0.679**	0.589	0.625	0.531	***0.670***	*0.776*

The best performance is marked with the italic fronts; the second best performance is marked with the bold fronts; the third best performance is marked with the bold italic fronts

Table 4 The AUC scores of the proposed method and other state-of-the-art methods in different attributes on the OTB2015 benchmark

Trackers	SCM [53]	Struck [14]	KCF [17]	DSST [8]	SAMF [23]	RPT [24]	MEEM [48]	LCT [27]	DLSSVM [31]	Ours
IV	0.502	0.431	0.481	***0.564***	0.531	0.553	0.526	**0.573**	0.530	*0.605*
OPR	0.437	0.429	0.456	0.487	***0.544***	0.521	0.537	**0.560**	0.542	*0.572*
SV	0.443	0.410	0.397	0.482	***0.497***	0.491	0.480	**0.505**	0.472	*0.539*
OCC	0.441	0.402	0.446	0.469	**0.552**	0.490	0.521	***0.531***	0.522	*0.587*
DEF	0.414	0.393	0.444	0.433	0.517	0.505	0.505	**0.524**	*0.527*	***0.521***
MB	0.269	0.467	0.464	0.475	0.524	0.526	***0.562***	0.540	*0.578*	**0.569**
FM	0.299	0.478	0.469	0.469	0.521	0.554	**0.561**	*0.567*	***0.559***	0.556
IPR	0.411	0.453	0.468	0.507	0.522	0.531	0.533	*0.563*	***0.537***	**0.538**
OV	0.345	0.375	0.397	0.390	***0.490***	0.464	**0.492**	0.457	0.472	*0.528*
BC	0.473	0.437	0.503	0.530	0.534	*0.577*	0.525	***0.557***	0.523	**0.575**
LR	***0.412***	0.361	0.307	0.387	**0.435**	0.364	0.366	0.357	0.403	*0.500*

The best performance is marked with the italic fronts, the second best performance is marked with the bold fronts, the third best performance is marked with the bold italic fronts

tracker and MEEM can still locate the correct target when it reappears. In addition, we observe that the proposed method does not perform favorably in sequences with the attribute of fast motion. It may be because the search region we set is small for computational efficiency. Generally, our tracker achieves the excellent performance in most challenging scenes compared to other competing trackers.

4.4.3 Qualitative comparison

Figure 7 illustrates some sampled tracking results of our method and several state-of-the-art methods in 5 challenging sequences. The selected trackers include DSST, SCM, LCT and Struck. The DSST and LCT trackers fail to deal with the deformation after frame 330 in the *Girl2* sequence. The Struck, DSST and SCM trackers drift away

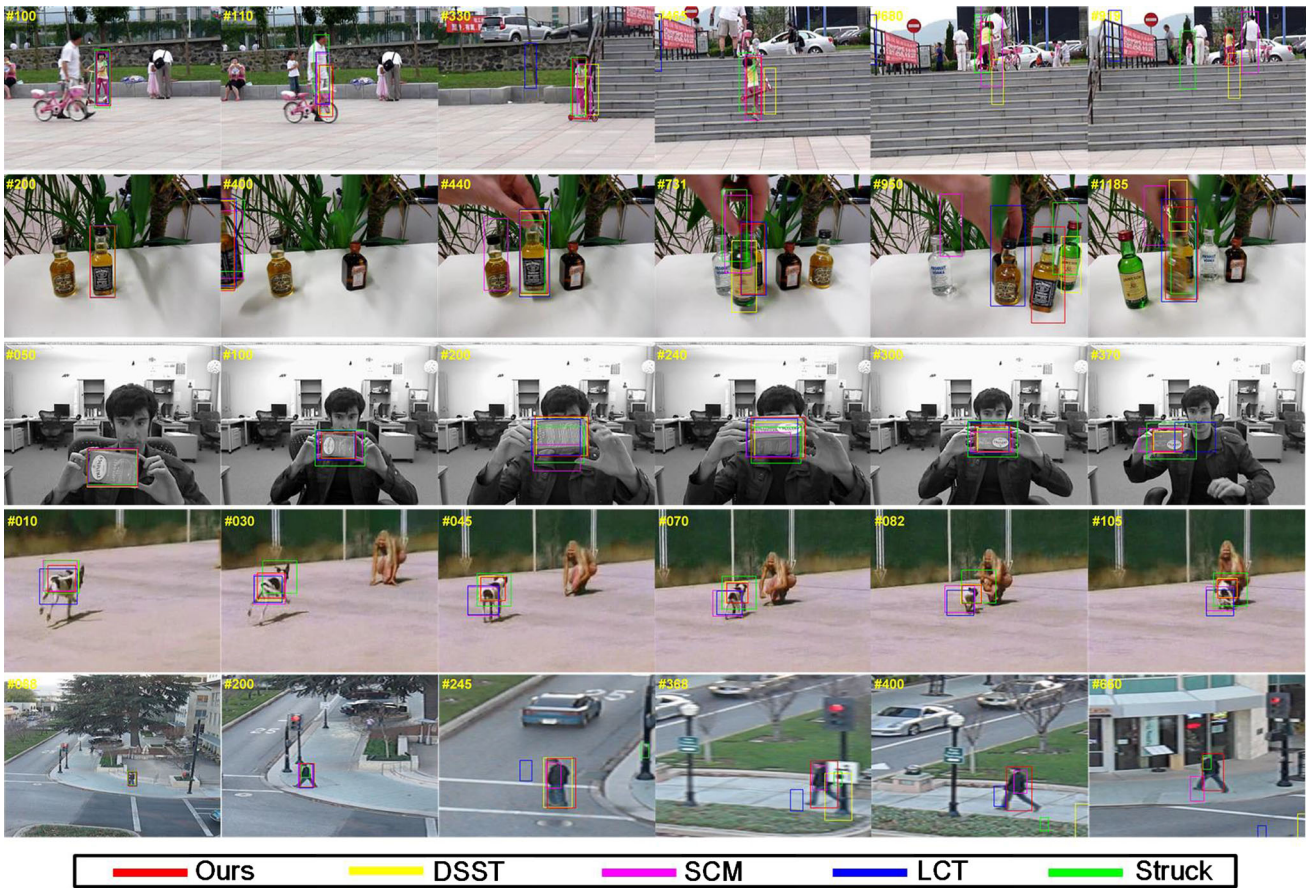

Fig. 7 Some sampled tracking results of 5 trackers in the *Girl2*, *Liquor*, *Twinning*, *Dog*, *Human6* sequences

from the target due to the challenge of occlusions at frame 731 in the *Liquor* sequence. The SCM tracker is less robust to the scale variation and out-of-plane rotation at frame 200 in the *Twinning* sequence and frame 45 in the *Dog* sequence. Moreover, all compared trackers do not perform well in the *Human6* sequences, where the target undergoes various appearance variations including fast motion, occlusion and out of view. Overall, our tracker achieves the excellent performance in above mentioned sequences.

4.5 Ablation analysis

In this section, we carry out the ablation analysis to give a deeper understand of the proposed method. All experiments are conducted in the OTB2013 benchmark and the comparison results are illustrated in Fig. 8.

We evaluate the effectiveness of each component of the proposed method to prove its contribution. Based on the proposed method, we present four variants with different configurations for comparisons: (1) the **Ours_WACS** method: our method without the adaptive multi-view collaboration strategy; (2) the **Ours _WPUM** method: our method without the proposed online update model. It means that the conventional linear interpolation update rule and the consistent and constant learning rate are used in this method; (3) the **Ours_WFASU** method: our method without the failure-aware scale update scheme; (4) the **Ours_WALL** method: our method without all components proposed in this paper. From Fig. 8, we have the following observations. First, by adaptively collaborating multiple views with consideration of the reliability and discrimination, our method improves the "**Ours_WACS**" method by 6.7% and 4.6% in the precision plots and success plots. Second, our method improves the "**Ours_WPUM**" by 8.0% and 5.5% in the DP scores and the AUC scores, which demonstrates the effectiveness of the proposed online update model. Third, our method obtains the performance gains compared with the "**Ours_WFASU**" method in both DP scores and AUC scores. It can be demonstrated that our scale update method is more robust against the direct scale update fashion by detecting the underlying tracking failure. Fourth, our method outperforms the "**Ours_WALL**" method by a large margin (12.1% in the precision plots and 7.9% in the success plots) by combining all components into a unified framework.

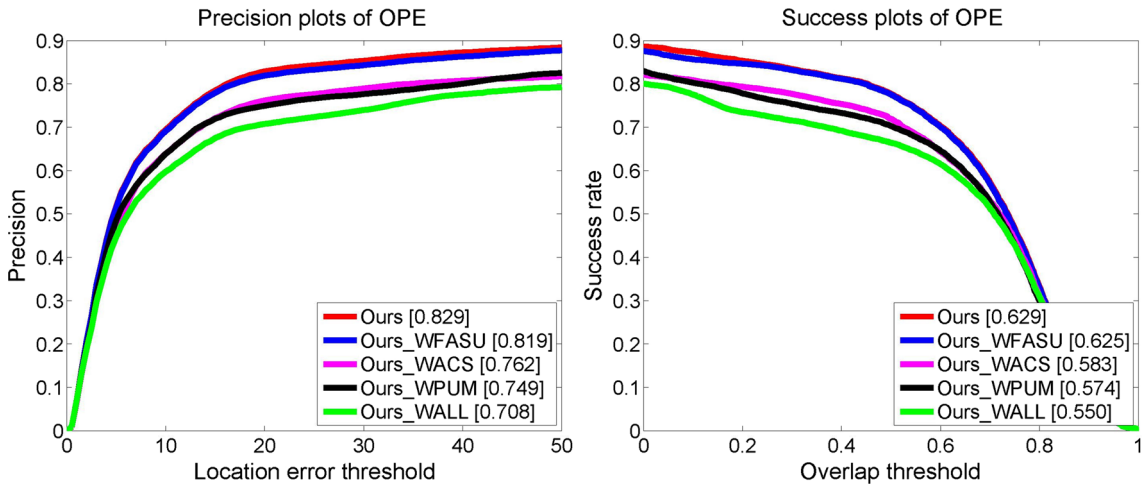

Fig. 8 The precision plots and success plots of our method with different configurations

These results prove the contribution of each component proposed in this paper.

4.6 Parameter analysis

There are several important parameters influencing the performance of our tracker, such as the regularization parameter λ in Eq. (1), trade-off parameter γ in Eq. (8), the learning parameters η_0 and L in Eq. (11) and the threshold τ in Eq. (12). In this section, we analyze the effects of these parameters on the OTB2013 benchmark with DP scores.

1. Effect of λ: The parameter λ in Eq. (1) is a regularization parameter which controls overfitting by governing the relative importance of the regularization term compared with the error term in the ridge regression model. The larger λ means the heavier penalization imposed on the filter coefficients. Table 5 shows the corresponding DP scores when we set λ to 0, 0.01, 0.1, 1 and 10, respectively. The best tracking performance is achieved with λ at 0.01.
2. Effect of γ: the parameter γ is a trade-off between the reliability and the discrimination in the weight assignment function Eq. (8). To evaluate the effect of γ, we parameterize it by a discrete set {0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0}. Figure 9a shows the corresponding DP scores for different γ. It can be observed that too small γ results in a degraded tracking performance, which implies the importance of the discrimination in collaborating multiple views. While too large γ is also not a proper selection as reliability is an indispensable factor to measure a view as well. Therefore, we set a moderate value of γ to 0.8 because it achieves the best performance.

Table 5 The DP scores using different λ values

	$\lambda = 0$	$\lambda = 0.01$	$\lambda = 0.1$	$\lambda = 1$	$\lambda = 10$
DP	0.827	**0.829**	0.828	0.823	0.817

Bold value indicates the best performance

3. Effect of τ: The parameter τ in Eq. (12) is the threshold which controls the update of the target scale. Once the underlying tracking failure occurs, we will stop estimating the target scale to avoid inducing noises. Similarly, we exploit the same method as evaluating γ to analyze τ. We parameterize it by a discrete set {0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9} and the corresponding DP scores are shown in Fig. 9b. From Fig. 9b, we find that the greatest DP score is achieved when we set τ to 0.5.
4. Effects of η_0 and L: The η_0 and L in Eq. (11) decide the learning rates in each frame. Specifically, η_0 reflects the importance of the memory for the first frame, while L controls the learning rates for subsequent frames. To better explore the relationship between this two learning parameters, we list Table 6 by exploiting different η_0 and L. As can be seen in Table 6, the tracking performance is degraded when η_0 is set too large. This is because excessive memory for the first frame will make the tracker less robust to drastic appearance variations. Our tracker achieves the best tracking performance by setting η_0 to 1.1 and L to 0.7.

5 Conclusion

In this paper, we propose a novel multi-view correlation filters-based tracking algorithm to achieve both robustness and efficiency. First, to better collaborate multiple views to

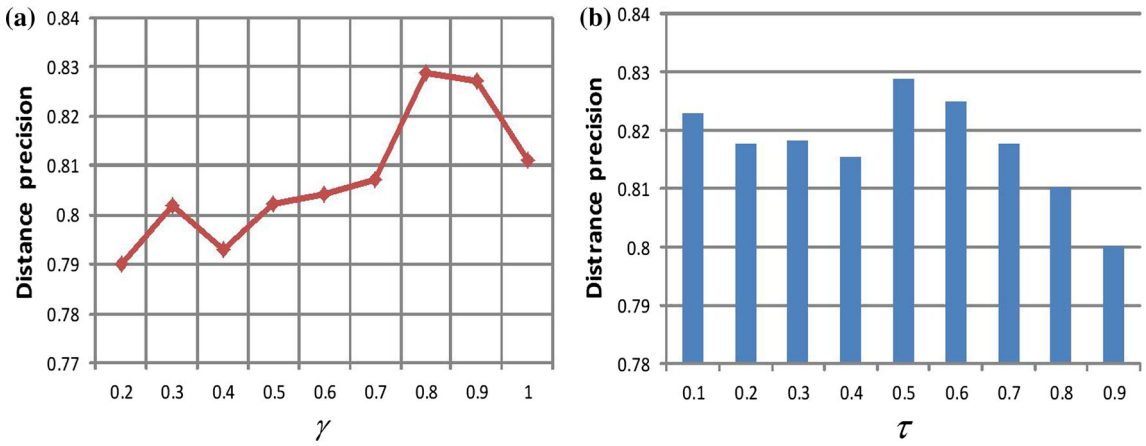

Fig. 9 Effects of **a** γ and **b** τ with different values

Table 6 The DP scores using different η_0 and L values

	$\eta_0 = 0.9$	$\eta_0 = 1.0$	$\eta_0 = 1.1$	$\eta_0 = 1.2$	$\eta_0 = 1.3$
$L = 0.5$	0.804	0.806	0.796	0.791	0.781
$L = 0.6$	0.805	0.808	0.801	0.806	0.796
$L = 0.7$	0.823	0.806	**0.829**	0.803	0.801
$L = 0.8$	0.805	0.821	0.806	0.805	0.807
$L = 0.9$	0.794	0.805	0.825	0.823	0.809

Bold value indicates the best performance

deal with more complex environment, we present an adaptive multi-view collaboration strategy by jointly considering the reliability and discrimination. Then considering that the conventional linear interpolation update model looses the memory of historical models over time, we propose an effective memory-improved model update rule to maintain these important information. Furthermore, we design dynamic and diverse learning rates to prevent all models from being contaminated at the same time. Last but not least, a failure-aware scale update scheme is introduced to reduce the impact of failing translation estimation. Extensive experiments are performed on the recent benchmark with an excellent performance against several state-of-the-art trackers.

Acknowledgements This work was supported by the National key R&D Program of China (Grant Nos. 2017YFB0202901, 2017YFB0202905). The corresponding author of this paper is Manman Peng (pengmanman@hnu.edu.cn).

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Bertinetto L, Valmadre J, Golodetz S, Miksik O, Torr PHS (2016) Staple: complementary learners for real-time tracking. In: 2016 IEEE conference on computer vision and pattern recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016, pp 1401–1409
2. Bertinetto L, Valmadre J, Henriques JF, Vedaldi A, Torr PHS (2016) Fully-convolutional siamese networks for object tracking. In: Computer vision-ECCV 2016 Workshops, Amsterdam, The Netherlands, October 8–10 and 15–16, 2016, proceedings, part II, pp 850–865
3. Bibi A, Ghanem B (2015) Multi-template scale-adaptive kernelized correlation filters. In: 2015 IEEE international conference on computer vision workshop, ICCV workshops 2015, Santiago, Chile, December 7–13, 2015, pp 613–620
4. Bibi A, Mueller M, Ghanem B (2016) Target response adaptation for correlation filter tracking. In: Computer vision-ECCV 2016—14th European conference, Amsterdam, The Netherlands, October 11–14, 2016, proceedings, part VI, pp 419–433
5. Bolme DS, Beveridge JR, Draper BA, Lui YM (2010) Visual object tracking using adaptive correlation filters. In: The twenty-third IEEE conference on computer vision and pattern recognition, CVPR 2010, San Francisco, CA, USA, 13–18 June 2010, pp 2544–2550
6. Chen K, Tao W, Han S (2017) Visual object tracking via enhanced structural correlation filter. Inf Sci 394:232–245
7. Chen W, An J, Li R, Fu L, Xie G, Bhuiyan MZA, Li K (2018) A novel fuzzy deep-learning approach to traffic flow prediction with uncertain spatial-temporal data features. Future Gener Comput Syst 89:78–88
8. Danelljan M, Häger G, Khan FS, Felsberg M (2014) Accurate scale estimation for robust visual tracking. In: British machine vision conference, BMVC 2014, Nottingham, UK, September 1–5, 2014
9. Danelljan M, Häger G, Khan FS, Felsberg M (2015) Convolutional features for correlation filter based visual tracking. In: 2015 IEEE international conference on computer vision workshop, ICCV workshops 2015, Santiago, Chile, December 7–13, 2015, pp 621–629
10. Danelljan M, Häger G, Khan FS, Felsberg M (2015) Learning spatially regularized correlation filters for visual tracking. In: 2015 IEEE international conference on computer vision, ICCV 2015, Santiago, Chile, December 7–13, 2015, pp 4310–4318

11. Dong X, Shen J, Yu D, Wang W, Liu J, Huang H (2017) Occlusion-aware real-time object tracking. IEEE Trans Multimed 19(4):763–771
12. Fang Y, Zhang H, Ye Y, Li X (2014) Detecting hot topics from twitter: a multiview approach. J Inf Sci 40(5):578–593
13. Gao J, Ling H, Hu W, Xing J (2014) Transfer learning based visual tracking with gaussian processes regression. In: Computer vision-ECCV 2014—13th European conference, Zurich, Switzerland, September 6–12, 2014, proceedings, part III, pp 188–203
14. Hare S, Saffari A, Torr PHS (2011) Struck: structured output tracking with kernels. In: IEEE International conference on computer vision, ICCV 2011, Barcelona, Spain, November 6–13, 2011, pp 263–270. https://doi.org/10.1109/ICCV.2011.6126251
15. Held D, Thrun S, Savarese S (2016) Learning to track at 100 FPS with deep regression networks. In: Computer vision-ECCV 2016—14th European conference, Amsterdam, The Netherlands, October 11–14, 2016, proceedings, part I, pp 749–765
16. Henriques JF, Caseiro R, Martins P, Batista JP (2012) Exploiting the circulant structure of tracking-by-detection with kernels. In: Computer cision-ECCV 2012—12th European conference on computer vision, Florence, Italy, October 7–13, 2012, proceedings, part IV, pp 702–715. https://doi.org/10.1007/978-3-642-33765-9_50
17. Henriques JF, Caseiro R, Martins P, Batista J (2015) High-speed tracking with kernelized correlation filters. IEEE Trans Pattern Anal Mach Intell 37(3):583–596. https://doi.org/10.1109/TPAMI.2014.2345390
18. Hong S, You T, Kwak S, Han B (2015) Online tracking by learning discriminative saliency map with convolutional neural network. In: Proceedings of the 32nd international conference on machine learning, ICML 2015, Lille, France, 6–11 July 2015, pp 597–606
19. Li G, Peng M, Nai K, Li Z, Li K (2018) Visual tracking via context-aware local sparse appearance model. J Vis Commun Image Represent 56:92–105
20. Li H, Wu H, Zhang H, Lin S, Luo X, Wang R (2017) Distortion-aware correlation tracking. IEEE Trans Image Process 26(11):5421–5434
21. Li X, Hu W, Shen C, Zhang Z, Dick AR, van den Hengel A (2013) A survey of appearance models in visual object tracking. ACM Trans Intell Syst Technol 4(4):58:1–58:48. https://doi.org/10.1145/2508037.2508039
22. Li X, Liu Q, He Z, Wang H, Zhang C, Chen W (2016) A multi-view model for visual tracking via correlation filters. Knowl Based Syst 113:88–99
23. Li Y, Zhu J (2014) A scale adaptive kernel correlation filter tracker with feature integration. In: Computer vision-ECCV 2014 workshops—Zurich, Switzerland, September 6–7 and 12, 2014, proceedings, part II, pp 254–265. https://doi.org/10.1007/978-3-319-16181-5_18
24. Li Y, Zhu J, Hoi SCH (2015) Reliable patch trackers: robust visual tracking by exploiting reliable patches. In: IEEE conference on computer vision and pattern recognition, CVPR 2015, Boston, MA, USA, June 7–12, 2015, pp 353–361
25. Li Z, Gao S, Nai K (2017) Robust object tracking based on adaptive templates matching via the fusion of multiple features. J Vis Commun Image Represent 44:1–20. https://doi.org/10.1016/j.jvcir.2017.01.012
26. Lukezic A, Vojir T, Zajc LC, Matas J, Kristan M (2017) Discriminative correlation filter with channel and spatial reliability. In: 2017 IEEE conference on computer vision and pattern recognition, CVPR 2017, Honolulu, HI, USA, July 21–26, 2017, pp 4847–4856
27. Ma C, Yang X, Zhang C, Yang M (2015) Long-term correlation tracking. In: IEEE conference on computer vision and pattern recognition, CVPR 2015, Boston, MA, USA, June 7–12, 2015, pp 5388–5396
28. Ma L, Lu J, Feng J, Zhou J (2015) Multiple feature fusion via weighted entropy for visual tracking. In: 2015 IEEE international conference on computer vision, ICCV 2015, Santiago, Chile, December 7–13, 2015, pp 3128–3136
29. Nai K, Li Z, Li G, Wang S (2018) Robust object tracking via local sparse appearance model. IEEE Trans Image Process 27(10):4958–4970
30. Nai K, Xiao D, Li Z, Jiang S, Gu Y (2019) Multi-pattern correlation tracking. Knowl Based Syst. https://doi.org/10.1016/j.knosys.2019.05.032
31. Ning J, Yang J, Jiang S, Zhang L, Yang M (2016) Object tracking via dual linear structured SVM and explicit feature map. In: 2016 IEEE conference on computer vision and pattern recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016, pp 4266–4274
32. Smeulders AWM, Chu DM, Cucchiara R, Calderara S, Dehghan A, Shah M (2014) Visual tracking: an experimental survey. IEEE Trans Pattern Anal Mach Intell 36(7):1442–1468. https://doi.org/10.1109/TPAMI.2013.230
33. Sui Y, Zhang Z, Wang G, Tang Y, Zhang L (2016) Real-time visual tracking: Promoting the robustness of correlation filter learning. In: Computer vision-ECCV 2016—14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, proceedings, part VIII, pp 662–678
34. Sun S, An Z, Jiang X, Zhang B, Zhang J (2019) Robust object tracking with the inverse relocation strategy. Neural Comput Appl 31:123–132
35. Tang M, Feng J (2015) Multi-kernel correlation filter for visual tracking. In: 2015 IEEE international conference on computer vision, ICCV 2015, Santiago, Chile, December 7–13, 2015, pp 3038–3046
36. Valmadre J, Bertinetto L, Henriques JF, Vedaldi A, Torr PHS (2017) End-to-end representation learning for correlation filter based tracking. In: 2017 IEEE conference on computer vision and pattern recognition, CVPR 2017, Honolulu, HI, USA, July 21–26, 2017, pp 5000–5008
37. Wang N, Li S, Gupta A, Yeung D (2015) Transferring rich feature hierarchies for robust visual tracking. CoRR arXiv:1501.04587
38. Wang X, Hou Z, Yu W, Pu L, Jin Z, Qin X (2018) Robust occlusion-aware part-based visual tracking with object scale adaptation. Pattern Recognit 81:456–470
39. Wu Y, Lim J, Yang M (2013) Online object tracking: a benchmark. In: 2013 IEEE conference on computer vision and pattern recognition, Portland, OR, USA, June 23–28, 2013, pp 2411–2418. https://doi.org/10.1109/CVPR.2013.312
40. Wu Y, Lim J, Yang M (2015) Object tracking benchmark. IEEE Trans Pattern Anal Mach Intell 37(9):1834–1848
41. Xie G, Zeng G, Jiang J, Fan C, Li R, Li K (2017) Energy management for multiple real-time workflows on cyber-physical cloud systems. Future Gener Comput Syst. https://doi.org/10.1016/j.future.2017.05.033
42. Xie K, Li X, Wang X, Xie G, Wen J, Cao J, Zhang D (2017) Fast tensor factorization for accurate internet anomaly detection. IEEE ACM Trans Netw 25(6):3794–3807
43. Xie K, Li X, Wang X, Cao J, Xie G, Wen J, Zhang D, Qin Z (2018) On-line anomaly detection with high accuracy. IEEE ACM Trans Netw 26(3):1222–1235
44. Xie K, Peng C, Wang X, Xie G, Wen J, Cao J, Zhang D, Qin Z (2018) Accurate recovery of internet traffic data under variable rate measurements. IEEE ACM Trans Netw 26(3):1137–1150
45. Xu C, Tao D, Xu C (2013) A survey on multi-view learning. CoRR arXiv:1304.5634

46. Yang B, Li Z, Jiang S, Li K (2018) Envy-free auction mechanism for VM pricing and allocation in clouds. Future Gener Comput Syst 86:680–693
47. Yoon JH, Yang M, Yoon K (2016) Interacting multiview tracker. IEEE Trans Pattern Anal Mach Intell 38(5):903–917
48. Zhang J, Ma S, Sclaroff S (2014) MEEM: robust tracking via multiple experts using entropy minimization. In: Computer vision-ECCV 2014—13th European conference, Zurich, Switzerland, September 6–12, 2014, proceedings, part VI, pp 188–203
49. Zhang K, Liu Q, Wu Y, Yang M (2016) Robust visual tracking via convolutional networks without training. IEEE Trans Image Process 25(4):1779–1792. https://doi.org/10.1109/TIP.2016.2531283
50. Zhang L, Suganthan PN (2017) Robust visual tracking via co-trained kernelized correlation filters. Pattern Recognit 69:82–93
51. Zhang S, Yu X, Sui Y, Zhao S, Zhang L (2015) Object tracking with multi-view support vector machines. IEEE Trans Multimed 17(3):265–278. https://doi.org/10.1109/TMM.2015.2390044
52. Zhang T, Xu C, Yang M (2017) Multi-task correlation particle filter for robust object tracking. In: 2017 IEEE conference on computer vision and pattern recognition, CVPR 2017, Honolulu, HI, USA, July 21–26, 2017, pp 4819–4827
53. Zhong W, Lu H, Yang M (2014) Robust object tracking via sparse collaborative appearance model. IEEE Trans Image Process 23(5):2356–2368