Information Sciences 518 (2020) 238–255

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Reliable correlation tracking via dual-memory selection model

Guiji Li[a], Manman Peng[a,*], Ke Nai[a], Zhiyong Li[a], Keqin Li[a,b]

[a] College of Information Science and Engineering, Hunan University, Changsha 410082, Hunan, China
[b] Department of Computer Science, State University of New York, New Paltz, New York 12561, USA

ARTICLE INFO

Article history:
Received 26 August 2019
Revised 9 January 2020
Accepted 10 January 2020
Available online 11 January 2020

Keywords:
Correlation filter
Long-term memory
Reliability evaluation
Visual tracking

ABSTRACT

Correlation-filter-based trackers have shown favorable accuracy and efficiency in visual tracking. However, most of these trackers are prone to drift in cases of heavy occlusions and temporal tracking failures because they only maintain the short-term memory of target appearance via a highly adaptive update mode. In this paper, we propose a reliable visual tracking method based on a dual-memory selection (DMS) model to alleviate tracking drift. Considering that long-term memory is robust to heavy occlusions while short-term memory performs well in rapid appearance changes, the proposed DMS model combines these two memory patterns of the target appearance and adaptively selects a reliable memory pattern to handle the current tracking challenges via a memory selector. For each memory pattern, a memory tracker is established based on discriminative correlation filters. The short-term tracker aggressively updates the target model to capture recent appearance changes via a linear interpolation update model, while the long-term tracker conservatively updates the target model to maintain historical appearance characteristics with a memory-improved update model and a dynamic learning rate. Furthermore, a novel memory evaluation criterion (MEC) is developed to evaluate the reliability of each tracker for memory selection. From credibility and discriminability measurements considering the temporal context, the memory tracker with the highest reliability score is selected to determine the target location in each frame. Extensive experiments on public benchmark datasets demonstrate that the proposed tracking method performs favorably compared to multiple recent state-of-the-art methods.

1. Introduction

Visual tracking is a fundamental and important topic in computer vision, and it has numerous applications, ranging from video surveillance, human-machine interaction, and robotic services to automatic driving. This technique aims to estimate the trajectory of an unknown target in an image sequence with only a given initial state. Although significant progress [1,2,13,18,23,31] has been achieved over the past decades, designing an efficient and robust tracking algorithm is still quite challenging due to several factors, such as target deformations, background clutters and occlusions.

Recently, discriminative correlation filters (DCFs) have been successfully applied to visual tracking and have received extensive attention. In general, DCF-based tracking methods follow the tracking-by-detection framework, in which the training, detection and updating steps are sequentially executed during the entire tracking process. However, unlike most

* Corresponding author.
E-mail addresses: guiji.li@hnu.edu.cn (G. Li), pengmanman@hnu.edu.cn (M. Peng), naike_hnu@hnu.edu.cn (K. Nai), zhiyong.li@hnu.edu.cn (Z. Li), lik@newpaltz.edu (K. Li).

https://doi.org/10.1016/j.ins.2020.01.015
0020-0255/

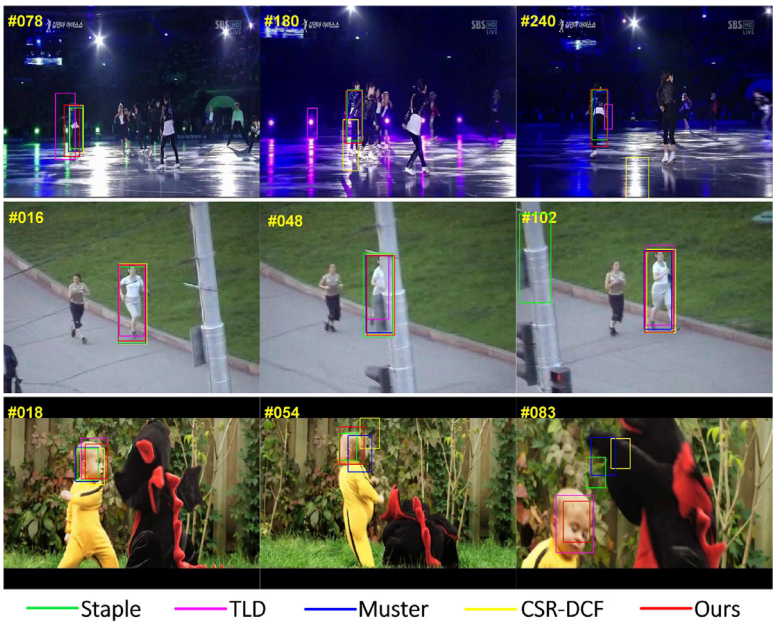

Fig. 1. Comparisons of the proposed method with other state-of-the-art methods (Staple [1], TLD [18], Muster [17], CSR-DCF [28]) on several video sequences (*Shaking, Jogging-2, Dragonbaby*) with significant deformation, heavy occlusions and out-of-plane rotation. Our method performs better than these methods by considering both the short-term memory and long-term memory of the target appearance in the proposed DMS model.

existing tracking-by-detection trackers, DCF-based trackers perform the training and detection steps more efficiently using the circular sample assumption and fast Fourier transform (FFT) technique. Moreover, the introductions of approximate dense sampling and high-dimensional features further enhance the accuracy of DCF-based tracking methods.

However, correlation-filter-based trackers are prone to drift due to their highly adaptive model update modes, especially when the target undergoes many more challenging factors, such as heavy occlusions and background clutters. Unreliable tracking results will contaminate the filter over time, which can lead to tracking failure if not immediately addressed. To mitigate the model drift problem, some researchers [4,5] design a dynamic learning rate based on the confidence of the current tracking result. However, it is not easy to robustly evaluate the tracking confidence, and this is always unfeasible in some complex scenarios. Other tracking methods [9,28] attempt to strengthen the model discrimination by reducing boundary effects. Unfortunately, they generally need to solve a complicated model formulation with a time-consuming optimization procedure, which may limit their use in many real-time applications. Recently, a number of works [27,29] focus on including a redetection scheme to refine unreliable tracking results. However, these methods always trust the redetection result without careful checking. Once the redetection result is corrupted, they will lose the chance to recover from tracking failures.

Motivated by the work in [29], we introduce the long-term memory of target appearance to alleviate the problem of model drift. Long-term memory provides more historical information of target appearance and is thus robust for handling heavy occlusions. Short-term memory is also an indispensable information resource for adapting to rapid appearance changes, and it cannot be replaced by long-term memory. In fact, these two memory patterns are complementary to each other, and cooperation between them is supposed to enhance both the adaptivity and robustness for visual tracking. Fig. 1 illustrates the specialities of trackers with different memories and the effectiveness of combining both short-term memory and long-term memory. Thanks to the maintenance of short-term memory, the Staple tracker adapts well to large appearance changes in the *Skating1* sequence, where the long-term tracker TLD fails. However, when the target suffers from heavy occlusions in the *Jogging2* sequence, the long-term tracker TLD performs more robustly than the short-term tracker Staple. By combining both short-term memory and long-term memory, our tracker and the MUSTer tracker perform favorably compared to the Staple tracker and the TLD tracker. In particular, the multistore tracker (MUSTer) also exploits both short-term memory and long-term memory to achieve better tracking performance. Despite the demonstrated success, MUSTer is computationally expensive because it needs to perform keypoint matching-tracking and RANSAC estimation based on the SIFT descriptors. Moreover, MUSTer has many parameters to carefully tune, which may weaken its generalizability in some new datasets.

In this study, we propose a dual-memory selection (DMS) model to alleviate the tracking drift problem by considering both the short-term memory and long-term memory of target appearance. The dual-memory pattern is able to provide a richer target appearance representation and enhance both the adaptivity and robustness for visual tracking. Specifically, the proposed DMS model consists of four components: a short-term tracker, a long-term tracker, the memory evaluation

criterion (MEC) and a memory selector. These four components work collaboratively to construct a reliable tracking framework. Since long-term memory is robust for handling heavy occlusions and short-term memory performs well in adapting to rapid appearance changes, we build two trackers based on correlation filters with short-term memory and long-term memory, respectively. The short-term tracker uses the linear interpolation update model to capture the recent target appearance. The long-term tracker exploits the memory-improved update model to maintain the memory of the historical target appearance. Furthermore, considering that different memory patterns have respective specialities to deal with different challenging factors, it is desirable to design a memory selector to achieve better performance in various tracking scenes. The memory selector is able to adaptively select a reliable memory pattern depending on the need for handling the current challenge. Intuitively, a direct idea for memory selection is based on the estimation of the current target state. However, it is difficult to distinguish drastic appearance changes from occlusions because they usually show similar appearance characteristics. To better perform memory selection, we propose a novel MEC that is based on the reliability evaluation of trackers with short-term memory and long-term memory. Moreover, by introducing the temporal context into the reliability evaluation, a stable output is obtained with temporal continuity. Finally, we conduct extensive evaluation experiments on the OTB-2013, OTB-2015, VOT2015 and VOT2016 datasets. Compared with various state-of-the-art DCF-based and deep learning tracking algorithms, our tracker shows superior performance in terms of accuracy and speed.

The main contributions of this paper can be summarized as follows.

1. An adaptive DMS model is proposed for alleviating the problem of tracking drift. Considering that the short-term memory and long-term memory of target appearance play different roles in addressing various challenges, the DMS model adaptively selects the most reliable memory pattern via a memory selector according to the immediate requirement.
2. A novel MEC is developed for memory selection by evaluating the reliability of trackers with short-term memory and long-term memory. Moreover, the introduction of a temporal context helps output a more stable motion trajectory with temporal continuity.
3. Extensive experiments on four large-scale benchmarks have been conducted to demonstrate the competitive performance of our tracker compared with other state-of-the-art tracking algorithms.

The remaining context of our work is organized as follows. Section 2 gives an overview of related works to ours. Section 3 presents an elaboration of our work including the dual-memory selection model (DMS), short-term tracker, long-term tracker and memory evaluation criterion (MEC). In Section 4, extensive experimental results are shown with detailed discussions. Finally, the proposed work is concluded in Section 5.

2. Related works

There are several surveys that review the recent research progress in visual tracking, which can be found in [25,37]. In this section, we only discuss the works that are the most related to ours, namely, correlation tracking methods, multiexpert tracking methods and deep learning tracking methods.

2.1. Correlation tracking

In recent years, DCFs have been extensively studied by the object tracking community due to their excellent accuracy and efficiency. In [3], Bolme et al. developed a high-speed (~ 700 FPS) tracker called minimum output sum of squared error (MOSSE), which can be regarded as the pioneering work in applying correlation filters to visual tracking. Since only grayscale features were used, the MOSSE tracker suffered from poor discrimination in challenging videos. Based on this seminal study, several works have been presented to improve the tracking precision with multidimensional features. Henriques et al. [13] learned kernelized correlation filters (KCFs) in the Fourier domain by exploiting the circulant structure of training samples and HOG features. Danelljan et al. [10] introduced color name (CN) descriptors and further proposed an adaptive dimensionality reduction technique for keeping a reasonable computational overhead. Although this method achieved impressive tracking performance, there is still much room for improvement, e.g., including scale estimation and context information, designing long-term tracking frameworks, reducing boundary effects and combining multiple features. Danelljan et al. [7] presented a discriminative scale space tracker (DSST) to handle scale variations of the target. Mueller et al. [30] incorporated global context information into the standard formulation of correlation filters to alleviate model drift. Ma et al. [29] developed a long-term correlation tracking (LCT) framework equipped with a redetection module. In this system, when a tracking failure occurred, the redetector was activated to recover the target location. In [9], Danelljan et al. proposed adopting training samples larger than the learned filter to reduce boundary effects with a spatial regularization component. Rather than using shifted sample patches, Galoogahi et al. [12] established a background-aware correlation filter from real negative training samples for enhancing the model discrimination. Bertinetto et al. [1] developed a Staple algorithm that increased the tracking robustness by combining the HOG template model and color histogram model. In [24], multiple correlation filters were learned from several complementary features to model diverse appearance characteristics of the target based on multiview learning [14,42]. Zhao et al. [50] proposed cascaded correlation filters trained with high-level and low-level convolutional features to achieve robust tracking.

2.2. Multiexpert tracking

To handle complex target appearance variations in visual tracking, an increasing number of researchers have focused on employing multiple experts to improve model diversity. Multiple experts can be established with different appearance patterns, and the best expert is carefully selected based on a certain well-designed evaluation criterion. In [46], Zhang et al. collected the current tracker and its historical snapshots to constitute an expert ensemble. Each expert reflected an updating state of the tracker at different time nodes. During contamination of the current tracker with noisy samples, a multiexpert restoration scheme was performed by selecting the best expert based on a minimum entropy criterion. Hong et al. [17] presented a multistore tracker (MUSTer) that integrated the short- and long-term stores to provide different target appearance memories. The experts with these two memory stores were designed to address different scenarios, and the final output was decided by a controller that considered the inconsistency between them. Wang et al. [36] proposed a multicue analysis framework to explore the strengths of multiple types of features. They constructed multiple experts from different views and designed an adaptive switch mechanism to select the most robust expert via pair-evaluation and self-evaluation. Nai et al. [32] developed a multipattern correlation tracker (MPCT) in which multiple experts learned diverse target appearance patterns of the target to consider drastic appearance changes. Through a two-stage selection algorithm, a suitable expert was selected in each frame for target localization.

2.3. Deep tracking

Deep learning [47] has extensive applications in various visual tasks, including image recognition [43], place recognition [45], face-pose estimation [15] and image ranking [44]. Motivated by the remarkable success in the aforementioned areas, numerous works have been developed to improve the performance of visual tracking based on deep learning. Qi et al. [33] considered different characteristics of different CNN layers and employed an online Hedge algorithm to construct a strong tracker by combining all CNN-based weak trackers. Bertinetto et al. [2] applied a fully convolutional Siamese network to learn a similarity function. The target was determined by finding the candidate with the maximum similarity to the exemplar image through an exhaustive search. Kuai et al. [22] developed a target objectness model and a target template model to solve the problem of distortion and fixed template in Siamese-network-based trackers. Fan et al. [11] proposed a parallel tracking and verification framework to further improve both the accuracy and efficiency of visual tracking. This tracker used the DSST algorithm for fast tracking inference, while the verifier employed a Siamese network for accurate verification.

The recent and popular long short-term memory (LSTM) network showed great potential in visual tracking due to its capacity to handle sequential data and learn long-term dependencies. Kim et al. [19] constructed an RLSTM tracker for spatiotemporal attention learning by combining the LSTM and a residual framework. Yang et al. [39] learned a recurrent filter and adapted it to appearance variations of the target via an LSTM network. In [40], a dynamic memory network was proposed to improve the accuracy of template-matching trackers, where the LSTM was employed to maintain target appearance variations with an addressable memory. Note that the LSTM shares some similarity with the proposed dual-memory model because both maintain short-term and long-term memory to capture and remember the target appearance. However, their working mechanisms are intrinsically different, which is mainly reflected by the memory storage modes. Based on the recurrent neural network (RNN), the LSTM introduces an additional cell state to store the long-term memory of the previous information. Furthermore, the short-term memory and long-term memory are adaptively combined into a unified memory state using an input gate and forget gate. In contrast to the LSTM, the proposed method explicitly builds the short-term and long-term memory models based on the correlation filters with different update modes and learning rates. The memory selector is then applied to select the most reliable one via the MEC.

3. Our method

In this section, we first introduce the proposed DMS model in Section 3.1, which serves as the overall framework of our method. Then, we establish the short-term tracker and long-term tracker in Section 3.2 and Section 3.3, respectively. Finally, the MEC is elaborated in Section 3.4 by considering stable credibility and discriminability measurements.

3.1. Dual-memory selection model (DMS)

The DMS model contains four important components: a short-term tracker, a long-term tracker, the MEC and a memory selector. At each frame, these components work collaboratively to provide a reliable tracking result. The short-term tracker emphasizes the memory of recent target appearance and can well adapt to drastic appearance changes. The long-term tracker maintains the memory of historical target appearance and is robust to heavy occlusions. Based on the MEC, the DMS model adaptively selects a reliable tracker that is expert for handling the current challenging factors and regards the corresponding output as the target position,

$$p_T^t = SEL\left(p_S^t, p_L^t\right), \tag{1}$$

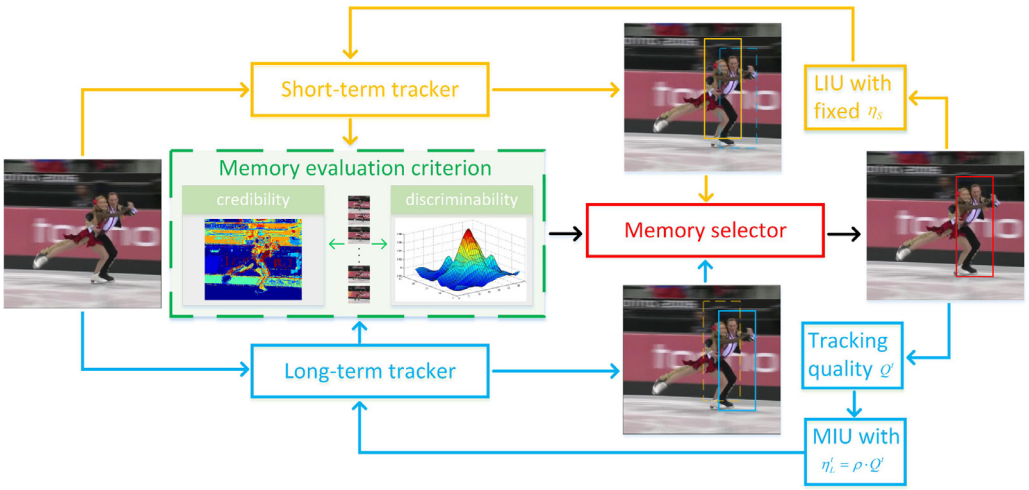

Fig. 2. The overall framework of the proposed tracking method.

where p_T^t is the target position in frame t and p_S^t and p_L^t are the positions output by the short-term tracker and long-term tracker, respectively. *SEL* is the memory selector and will be specifically described in the following section. For accuracy and efficiency, we employ DCFs as the baseline of the short-term tracker and long-term tracker, which follow the tracking-by-detection paradigm. Accordingly, the tracking position p^t is determined in image I^t by searching for the maximal score,

$$p^t = \arg\max_{p \in I^t} y\left(\delta\left(I^t, p\right), \theta^{t-1}\right), \tag{2}$$

where δ is a patch extraction function that extracts the image patch of the target size at position p. For each extracted candidate image patch, the function y assigns a score according to the model parameters θ. In the DCF-based methods, scores can be obtained from the correlation response map. Then, we obtain tracking results p_S^t and p_L^t of the short-term tracker and long-term tracker, respectively. The core module of the DMS model is the development of the MEC. It serves as the basis for *SEL* to perform memory selection by evaluating the reliability of the short-term tracker T_S^t and long-term tracker T_L^t,

$$SEL\left(p_S^t, p_L^t\right) = \begin{cases} p_S^t, & if\ R\left(T_S^t\right) > R\left(T_L^t\right) \\ p_L^t, & otherwise \end{cases} \tag{3}$$

where $R(\cdot)$ refers to the reliability evaluation function, and its concrete form can be found in Section 3.4. The overall framework of our method is shown in Fig. 2.

3.2. Short-term tracker

The short-term tracker employs a highly adaptive correlation filter to capture recent appearance changes of the target object. The standard DCF formulation performs both training and detection in the Fourier domain. At the training stage, a multichannel correlation filter h is learned from an image patch f of size $M \times N$ centered around the target location. The image patch f consists of a D- dimensional feature map, and all its circular shifts $f(m, n) \in \{0, 1, \ldots, M-1\} \times \{0, 1, \ldots, N-1\}$ can be regarded as training samples for learning. Each training sample is assigned a Gaussian function label $g(m, n)$ according to its distance from the target location. The filter h is then obtained by optimizing the ridge regression model as follows:

$$\min_h \left\| g - \sum_{d=1}^{D} h^d \star f^d \right\|^2 + \lambda \sum_{d=1}^{D} \left\| h^d \right\|^2. \tag{4}$$

Here, $\star$ represents the circular correlation and $\lambda(\lambda \geq 0)$ is a regularization parameter to control overfitting. Based on the convolution theorem, Eq. (4) can be efficiently solved in the Fourier domain, and the d-*th* channel of filter H is given by

$$H^d = \frac{G^* \odot F^d}{\sum_{k=1}^{D} \left(F^k\right)^* \odot F^k + \lambda}, \quad d = 1, \ldots, D, \tag{5}$$

where $\odot$ is elementwise multiplication and the fraction denotes elementwise division. The capital letters denote the discrete Fourier transform (DFT) of the corresponding quantities, and * denotes complex conjugation. At the detection stage, an image

patch z centered around the predicted target location is extracted in a new frame. The correlation response map r of z is then calculated as follows:

$$r^t = \mathcal{F}^{-1}\left(\frac{\sum_{d=1}^{D}\left(A_{t-1}^d\right)^* \odot Z_t^d}{Y_{t-1}+\lambda}\right), \tag{6}$$

where A_{t-1}^d and Y_{t-1} are the numerator and denominator of H_{t-1}^d, respectively. $\mathcal{F}^{-1}$ denotes the inverse DFT. The target location is determined by searching for the maximum correlation response value.

The short-term tracker T_S is aggressively updated in the t frame with the linear interpolation update model to adapt to drastic appearance changes,

$$\begin{aligned} A_t^d &= (1-\eta_S)A_{t-1}^d + \eta_S G^* \odot F_t^d \\ Y_t &= (1-\eta_S)Y_{t-1} + \eta_S \sum_{k=1}^{D}\left(F_t^k\right)^* \odot F_t^k, \\ T_S^{t,d} &= \frac{A_t^d}{Y_t+\lambda} \end{aligned} \tag{7}$$

where η_S is the learning rate of the short-term tracker. It remains fixed during the entire tracking process to maintain high adaptivity. Inspired by [28], we introduce a color mask that reflects the target likelihood to training samples based on color histograms. By assigning larger weights to target pixels and smaller weights to background pixels, the color mask can effectively enhance the spatial reliability and alleviate model drift.

3.3. Long-term tracker

To ensure an efficient tracking process, the long-term tracker also exploits a discriminative correlation filter to locate the target. However, unlike the short-term tracker, the long-term tracker adopts a memory-improved update model [24] to maintain the memory of the historical target appearance. As is known, the linear interpolation update model emphasizes recent frames and reduces the effects of historical frames exponentially over time; thus, it can effectively capture recent appearance changes of the target. In fact, historical appearance characteristics are very important to recover from tracking failures when the target temporally disappears in the view or undergoes heavy occlusions. Therefore, we incrementally update the long-term tracker T_L to improve the memory of historical target appearance,

$$\begin{aligned} A_t^d &= A_{t-1}^d + \eta_L^t G^* \odot F_t^d \\ Y_t &= Y_{t-1} + \eta_L^t \sum_{k=1}^{D}\left(F_t^k\right)^* \odot F_t^k, \\ T_L^{t,d} &= \frac{A_t^d}{Y_t+\lambda} \end{aligned} \tag{8}$$

where η_L^t is the learning rate of the long-term tracker. It is adaptively adjusted according to the current tracking quality.

To measure the tracking quality more accurately, we consider the response values of both the long-term tracker and short-term tracker at the target position p_T^t (determined by the DMS model in Eq. (1)). The current tracking quality q^t is defined as follows:

$$q^t = \frac{1}{2}\left(r_S\left(p_T^t\right) + r_L\left(p_T^t\right)\right). \tag{9}$$

Here, r_S and r_L refer to response maps of the short-term tracker and long-term tracker, respectively. Note that Eq. (9) is an absolute measurement of the current tracking quality, which is unreasonable to serve as the criterion for adjusting the learning rate since response values may significantly fluctuate in different video sequences. We thus use the relative measurement Q^t of the current tracking quality by considering the average of all past frames,

$$Q^t = \frac{q^t}{\frac{1}{t-1}\sum_{i=2}^{t} q^i}. \tag{10}$$

Based on the measurement of current tracking quality, the learning rate η_L^t is adjusted adaptively as follows:

$$\eta_L^t = \rho \cdot Q^t, \tag{11}$$

where ρ is a constant learning factor.

In contrast to the highly adaptive short-term tracker, the long-term tracker is more conservative for maintaining robustness. On the one hand, the long-term tracker exploits a memory-improved update model to improve the memory of the historical target appearance. On the other hand, the long-term tracker adaptively adjusts the learning rate to reduce the effects of corrupted tracking results. By retaining both adaptivity and robustness, these two complementary trackers provide a firm basis for the DMS model.

3.4. Memory evaluation criterion (MEC)

As a core component of the DMS model, MEC plays a vital role in improving tracking performance. After establishing the short-term tracker and long-term tracker, the memory selector in the DMS model will make a suitable selection from their tracking results based on the MEC. To provide a reliable evaluation criterion for memory selection, we elaborate the formulation of MEC by considering two important measurements. These measurements are described in the following section.

The first is the credibility measurement. We employ a color-histogram-based Bayes classifier to measure the tracking credibility of the short-term tracker and long-term tracker. There are two important reasons for exploiting such a method to perform credibility measurements. On the one hand, the color histogram captures the statistical characteristics of the object appearance and is thus robust to fast deformations. On the other hand, it is efficient to obtain the target likelihood scores as credibility measurements by using a lookup table and integral image method. Let x_k denote the bin of color histograms to which pixel k belongs. To distinguish the pixels inside the target region O from those inside the background region B, we calculate the target likelihood at location k using Bayes rules as follows:

$$P\left(k \in O|O', B, x_k\right) \approx \frac{P\left(x_k|k \in O'\right)P\left(k \in O'\right)}{\sum_{\Omega \in \{O', B\}} P(x_k|k \in \Omega)P(k \in \Omega)}. \tag{12}$$

In practice, a safer foreground region O' that is slightly smaller than O will be extracted to avoid mislabeling. We can easily observe that Eq. (12) contains the prior terms and likelihood terms. The prior term can be approximated as $P\left(k \in O'\right) = \left|O'\right| / \left(\left|O'\right| + |B|\right)$, where $|\cdot|$ refers to the cardinality. For the likelihood terms, we directly estimate them from the color histograms, i.e., $P\left(x_k|k \in O'\right) \approx N_{O'}(x_k)/\left|O'\right|$ and $P(x_k|k \in B) \approx N_B(x_k)/|B|$, where $N_\Omega(x_k)$ denotes the number of pixels in region Ω with bin x_k. Then, Eq. (12) can be simplified to

$$P\left(k \in O|O', B, x_k\right) = \frac{N_{O'}(x_k)}{N_{O'}(x_k) + N_B(x_k)}. \tag{13}$$

By summing the target likelihood scores of all pixels in the corresponding tracking results, the tracking credibility of tracker T with memory $\mathrm{M} \in \{S, L\}$ can be calculated as follows:

$$e^t_{C,\mathrm{M}} = \frac{1}{\left|\Re^t_{\mathrm{M}}\right|} \sum_{k \in \Re^t_{\mathrm{M}}} l_k, \tag{14}$$

where $\Re^t_{\mathrm{M}}$ denotes the region extracted from the corresponding tracking result at frame t and l_k refers to the target likelihood at location k, e.g., $l_k = P\left(k \in O|O', B, x_k\right)$.

The second is the discriminability measurement. A discriminative tracker is supposed to have high confidence at the target location and be less ambiguous at other locations. Therefore, we apply the average peak-to-correlation energy (APCE) criterion [35] to measure the tracking discriminability,

$$e^t_{D,\mathrm{M}} = \frac{|r_{\max} - r_{\min}|^2}{mean\left(\sum_{i,j}\left(r_{i,j} - r_{\min}\right)^2\right)}. \tag{15}$$

Here, $r_{\max}$, $r_{\min}$ and r_{ij} are the maximum, the minimum and the i-th row and j-th column elements of the response map, respectively. From Eq. (15), we can observe that the APCE becomes larger if a tracker has only one sharp peak and produces smooth response values in all other areas. This behavior indicates that the tracker has strong discriminability in distinguishing the target from the background. Otherwise, the APCE becomes smaller when it is ambiguous to determine the target location.

The two aforementioned measurements are based on the current frame evaluation, which is susceptible to sudden noise and easily causes performance fluctuations. To obtain a more stable output with temporal continuity, we introduce the temporal context information into the reliability evaluation of memory. We argue that a reliable memory tracker is supposed to have the following two important properties: (1) it should maintain excellent performance over a period of time, not just at the current frame, and (2) the excellent performance should be as stable as possible. In other words, the memory tracker with better stability is preferred in our study. Specifically, we use the mean to standard deviation ratio to quantify the stability for a certain measurement (taking the tracking credibility for example),

$$E^t_{C,\mathrm{M}} = \frac{U^{\mathrm{M}}_C}{\sqrt{V^{\mathrm{M}}_C} + \xi}, \tag{16}$$

where ξ is a small constant to avoid a zero in the denominator. U^{M}_C and V^{M}_C are the weighted mean value and variance of the credibility measurement in a temporal window Δ, which can be denoted as $U^{\mathrm{M}}_C = \frac{1}{\zeta}\sum_\tau w^\tau e^\tau_{C,\mathrm{M}}$ and $V^{\mathrm{M}}_C = \frac{1}{\zeta}\sum_\tau w^\tau \left(e^\tau_{C,\mathrm{M}} - U^{\mathrm{M}}_C\right)^2$, where $\tau \in [t - \Delta + 1, t]$. $w = \left\{\omega^0, \omega^1, \ldots, \omega^{\Delta-1}\right\}$, $(\omega > 1)$ is a weight sequence that gives more focus on the recent measurements, and w^τ is the $(\tau - t + \Delta) - th$ element. ζ is a normalization scalar denoted as $\zeta = \sum_\tau w^\tau$.

Table 1
The description of evaluation datasets.

Dataset	Year	Number of sequences	Number of attributes	Number of frames	Number of evaluated trackers
OTB-2013	2013	51	11	29,486	29
OTB-2015	2015	100	11	59,035	31
VOT2015	2015	60	5	21,455	62
VOT2016	2016	60	5	21,455	70

In the t frame, we calculate the reliability score of tracker T with memory M by linearly combining credibility and discriminability measurements as follows:

$$R\left(T_{\mathrm{M}}^{t}\right)=(1-\mu)\cdot E_{\mathrm{C,M}}^{t}+\mu\cdot E_{\mathrm{D,M}}^{t}. \tag{17}$$

Here, $E_{\mathrm{D,M}}^{t}$ is the discriminability measurement with consideration of stability, which can be obtained as $E_{\mathrm{C,M}}^{t}$ in Eq. (16). μ is a tradeoff between credibility and discriminability measurements. Eq. (17) indicates that the tracker showing excellent and stable credibility and discriminability over a period of time will obtain a higher reliability score. Then, the target location is determined by the most reliable tracker. We summarize the proposed DMS tracking method in Algorithm 1.

Algorithm 1 The Proposed DMS Tracking Algorithm.

Input:
Image I^{t};
Target position p_{T}^{t-1} at the $t-1$ frame.
Output:
Target position p_{T}^{t} at the t frame.

Extract the image patch z^{t} at p_{T}^{t-1} from I^{t} and apply the color mask to it;
for each tracker T with memory $\mathrm{M}\in\{S,L\}$ **do**
  Compute the response map r_{M}^{t} and estimated position p_{M}^{t} using Eq.(6) and Eq. (2);
  Compute the credibility $e_{\mathrm{C,M}}^{t}$ and discriminability $e_{\mathrm{D,M}}^{t}$ using Eq. (14) and Eq. (16);
  Get $E_{\mathrm{C,M}}^{t}$ and $E_{\mathrm{D,M}}^{t}$ with consideration of the temporal context into $e_{\mathrm{C,M}}^{t}$ and $e_{\mathrm{D,M}}^{t}$ using Eq. (17);
  Evaluate the reliability $R\left(T_{\mathrm{M}}^{t}\right)$ using Eq. (17);
end for
Get the target position p_{T}^{t} using Eq. (1) and Eq. (3);
Update the short-term tracker with η_{S} using Eq. (7);
Compute the current tracking quality Q^{t} using Eq. (9) and Eq. (10);
Compute the learning rate η_{L}^{t} with Q^{t} using Eq. (11);
Update the long-term tracker with η_{L}^{t} using Eq. (8);

4. Experimental results and analysis

In this section, we first introduce implemental details of our method including experimental environments and parameter settings. Then we present extensive comparisons on the OTB benchmark [37,38] and VOT benchmark [20,21] with state-of-the-art trackers to demonstrate the superiority of the proposed method. Finally, more detailed analysis is given on the parameters. A brief description of all evaluated datasets can be found in Table 1.

4.1. Implementation details

Our DMS tracker runs at approximately 40 frames per second (FPS) on a PC with an Intel Core i7-6700HQ CPU at 2.6 GHz and 8 G memory using a MATLAB implementation. In our implementation, a combination of grayscale, HOG and CN features is used to provide a rich appearance representation for the training and testing samples. All extracted samples are further multiplied by a Hanning window to reduce boundary discontinuities. The short-term tracker and long-term tracker are constructed based on the standard DCF formulation, following the parameters recommended in [36]. However, unlike the short-term tracker, the long-term tracker exploits a memory-improved update model with an adaptive learning rate, and the learning factor ρ in Eq. (11) is set to 1. The temporal window Δ and the weighting factor ω in the weight sequence w are set to 15 and 1.2, respectively. The tradeoff parameter μ between the credibility and discriminability measurements in Eq. (17) is set to 0.3. Note that all parameters used in our experiments remain the same for all video sequences and datasets.

4.2. Evaluation on OTB benchmark

4.2.1. Experimental settings

Evaluation datasets. We conduct extensive experiments on the OTB benchmark with the OTB-2013 and OTB-2015 datasets. The OTB-2013 dataset contains 51 video sequences and is extended by the OTB-2015 dataset with 100 video sequences. These video sequences have various challenging factors with 11 annotated attributes, 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). The overall performance and attribute-based performance will be reported on the OTB-2013 and OTB-2015 datasets later.

Evaluation metrics. The one-pass evaluation (OPE) criterion is used in the OTB evaluation, which means that the tested tracker will run throughout a video sequence only with the initial state at the first frame to perform the performance evaluation. Generally, the performance of a tested tracker is quantitatively measured by two metrics: precision plots and success plots. The precision plots show the percentage of frames whose center location error $CLE(x_p, x_g) = \|x_p - x_g\|_2$ between the predicted tracking location x_p and ground-truth location x_g is smaller than a given threshold. The success plots reflect the percentage of frames whose overlap score $O(B_p, B_g) = \frac{|B_p \cap B_g|}{|B_p \cup B_g|}$ between the predicted bounding box B_p and ground-truth bounding box B_g is greater than a given threshold. As in [37], the distance precision (DP) scores at a threshold of 20 pixels in the precision plots and the area-under-curve (AUC) scores in the success plots are used to rank the performance for comparisons.

Evaluation trackers. We evaluate the proposed DMS tracker with comparison to 12 state-of-the-art trackers, including MEEM [46], KCF [13], DSST[7], SAMF [26], LCT [29], MUSTer [17], Staple [1], SRDCF [9], BACF [12], CSR-DCF [28], LMCF [35], MCCT-H [36]. These trackers are mostly based on correlation filters due to the accuracy and efficiency. In addition, we compare our tracker with other 12 deep learning-based trackers for a further evaluation including CNT [48], SiamFc [2], DeepSRDCF [8], CNN-SVM [16], ACFN [6], CFnet [34], HDT[33], MCPF[49], PTAV [11], RFL [39], MemTrack [40] and DMN [41].

4.2.2. Quantitative comparisons on OTB-2013

Overall performance. We present the overall quantitative comparisons on the OTB-2013 dataset in Fig. 3, including the precision plots and success plots of the proposed DMS tracker and other state-of-the-art trackers. The DP scores and AUC scores of all compared trackers are shown in the legend. Overall, the DMS tracker performs well on these two evaluation metrics, obtaining a DP score of 86.5% and an AUC score of 66.4%, which respectively rank the first and the second among all compared trackers. Note that the MUSTer, LCT and our methods consider both the short-term memory and long-term memory of target appearance to improve the tracking performance. The excellent results of these methods in the precision plots and success plots indicate the effectiveness of cooperation between short-term memory and long-term memory. Moreover, our method performs better than the MUSTer and LCT methods, which is mainly due to the DMS model proposed in this paper. By carefully evaluating the reliability of the short-term tracker and long-term tracker, the DMS model selects the most reliable tracker to handle the current challenging factors.

The computational speed is also an important metric for evaluating the performance of a tracker. In Table 2, we list the average FPS and the DP scores to comprehensively evaluate the efficiency and accuracy of some state-of-the-art trackers.

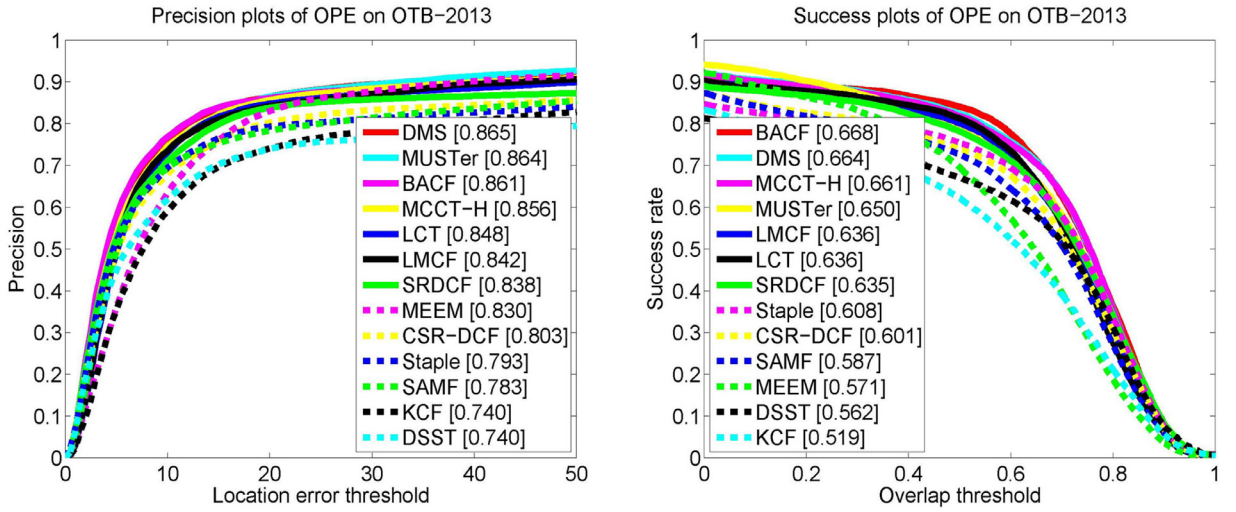

Fig. 3. Overall performance evaluation on the OTB-2013 benchmark with 51 video sequences. The DP scores and AUC scores are shown in the legend of the precision plots (left) and success plots (right) to rank all compared trackers.

Table 2
Speed comparisons (i.e., average frames per second (FPS)) of the proposed tracker with state-of-the-art trackers on the OTB-2013 benchmark. The DP scores (%) are also listed for more comprehensive comparisons. The top three performances are highlighted by the red, blue and green fonts, respectively.

Trackers	KCF	DSST	SAMF	MUSTer	LCT	SRDCF	Staple	CSR-DCF	BACF	MCCT-H	DMS
DP	74.0	74.0	78.3	86.4	84.8	83.8	79.3	80.3	86.1	85.6	86.5
FPS	277.5	34.0	17.5	4.3	32.7	3.9	60.3	8.3	25.4	25.8	39.6

From Table 2, we observe that the KCF tracker obtains the best tracking speed (277.5 FPS). Unfortunately, its accuracy has difficulty meeting the need for high-quality tracking. The SRDCF and CSR-DCF trackers significantly improve the accuracy, but at the expense of tracking speed. Overall, our DMS tracker performs favorably in both accuracy and efficiency, achieving an 86.5% DP score while still running at 39.6 FPS.

Attribute-based performance. We further analyze the attribute-based performance of our DMS tracker and other trackers in Fig. 4 using precision plots and success plots. In fact, the eleven attributes annotated on the OTB-2013 dataset cover the most challenging factors in visual tracking and are very valuable for evaluating the strengths and weaknesses of a tracker from different aspects. From Fig. 4, we observe that our DMS tracker performs well against other competing trackers under most attributes. In the precision plots, our DMS tracker achieves excellent performance and ranks within the top 3 on 10 of the 11 attributes. In the success plots, the DMS tracker ranks within the top 3 on 9 of the 11 attributes. In particular, the DMS tracker obtains both the highest DP and AUC scores in cases of occlusion (87.0%/67.0%), out-of-plane rotation (87.3%/65.9%), deformation (89.4%/70.1%) and illumination variation (82.0%/64.2%). In detail, for the sequences with out-of-plane rotation attributes, our tracker outperforms the second-best tracker MCCT-H (by 1.6%) and the third-best tracker BACF (by 1.9%) in terms of DP scores. For the sequences with occlusion attributes, our tracker performs better against the MUSTer tracker (by 1.7%) in terms of DP scores and the MCCT-H tracker (by 0.9%) in terms of AUC scores. Note that in addition to our DMS tracker, the MUSTer, MCCT-H, LCT and LMCF trackers perform well in cases of occlusion. Among these trackers, the MUSTer and LCT trackers retain the historical appearance information of the target object and are thus able to detect the target when it reappears. The MCCT-H and LMCF trackers carefully evaluate the tracking confidence and adaptively adjust the updating rate to reduce the effects of unreliable samples. By integrating these two effective strategies, the proposed DMS tracker achieves the best performance among the aforementioned trackers under occlusion. In addition, the DMS tracker performs well under scale variation, in-plane rotation, fast motion and background clutters. Unfortunately, however, the DMS tracker suffers from inferior performance in cases of low resolution. The reason is that the low resolution attribute only includes four sequences, which easily causes large bias when performing the performance evaluation. At first glance, the improvement of the proposed method is not particularly significant. However, our DMS tracker achieves excellent tracking with more robust performance. We find that the two most competitive trackers, BACF and MCCT-H, perform very poorly under deformation and background clutter, which rank ninth and fifth among all trackers and are lower than our tracker, with 6.6% and 2.7% in terms of DP scores. In general, the proposed DMS method performs robustly against other state-of-the-art trackers in most challenging scenes.

4.2.3. Quantitative comparisons on OTB-2015

Overall performance. Fig. 5(a) presents the comparison results of the overall performance on the OTB-2015 dataset in terms of precision plots and success plots. From the results, we can observe that our DMS tracker achieves the best performance in the precision plots with a DP of 85.2% and the second-best performance in the success plots with an AUC of 64.1%. Compared with another top method, MCCT-H, our method obtains a 1.1% gain in DP scores and a very close performance in AUC scores. Compared with other DCF-based trackers that also intend to alleviate the model drift problem either by reducing boundary effects (e.g., SRDCF, BACF, CSR-DCF) or by inducing multiple features (e.g., SAMF, Staple), the proposed DMS tracker outperforms the best performing tracker (BACF) among them by 2.9% and 1.2% in terms of DP scores and AUC scores, respectively. The underlying reason can be attributed to the introduction of the long-term memory, which provides more historical information of target appearance to enhance the model robustness.

In addition, we compare the proposed tracker with some related trackers that also integrate the short-term memory and long-term memory of target appearance into their methods. The MUSTer tracker exploits the short-term memory to perform instant tracking and the long-term memory to control the final output. The LCT tracker utilizes the long-term memory to estimate the confidence of each tracking result obtained with the short-term memory. The other three trackers are all based on the LSTM network, which introduces the cell state and various gates to improve the long-term memory of previous information. Again, we draw the precision plots and success plots in Fig. 5(b) for the performance evaluation. Due to the robust representation ability of deep features, the DMN, MemTrack and RFL trackers significantly outperform the MUSTer and LCT trackers that use handcrafted features to represent the target appearance. However, our method using similar handcrafted features performs favorably against the MemTrack and RFL trackers and achieves performance that is comparable with that of the DMN tracker. The results show the effectiveness of the proposed DMS model in integrating short-term memory and long-term memory compared to other related trackers.

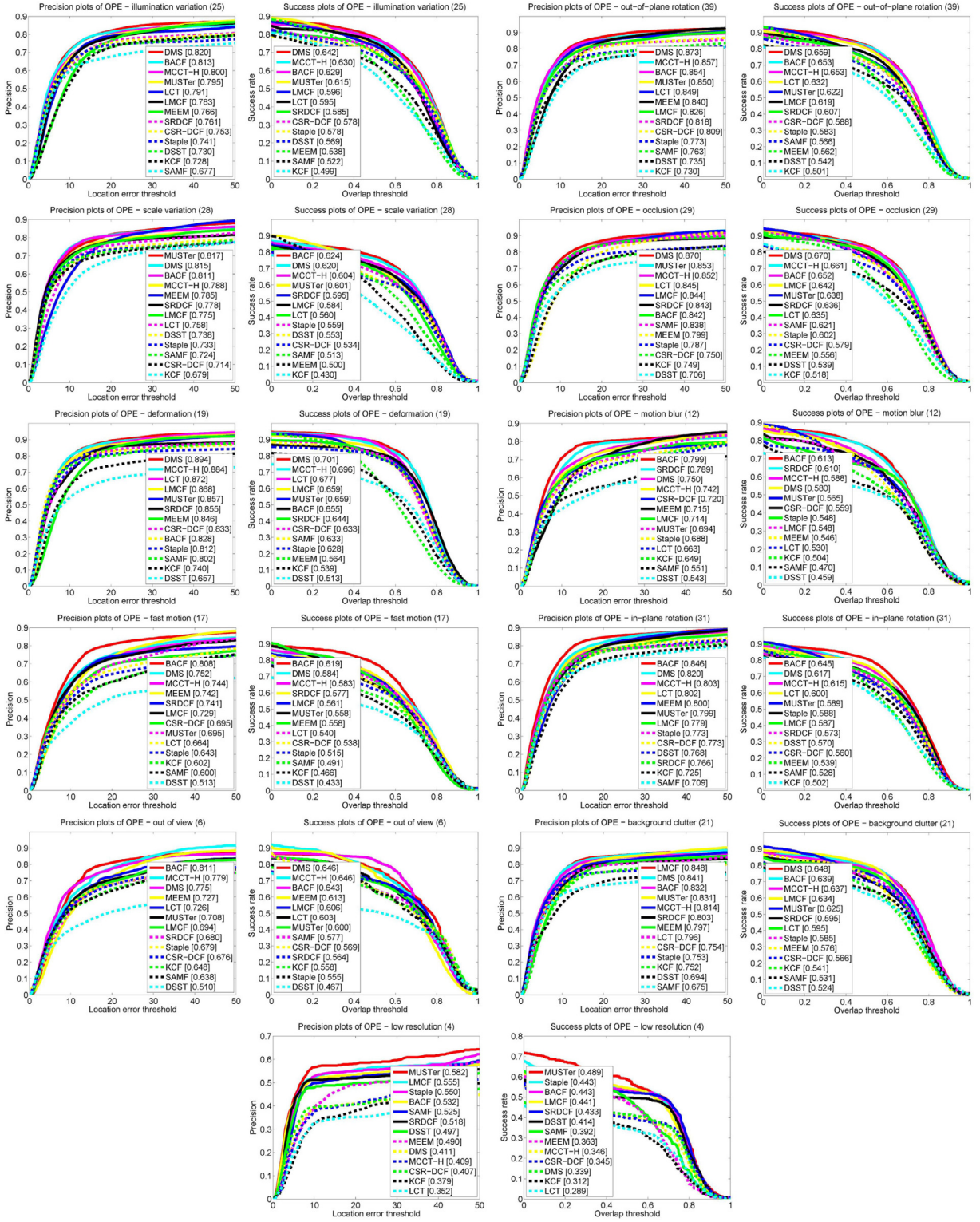

Fig. 4. Attribute-based performance evaluation on the OTB-2013 benchmark using precision plots and success plots. The number of video sequences for each attribute is shown in the title.

Table 3
Attribute-based performance comparsion on the OTB-2015 benchmark using DP scores and AUC scores. The top three performances are highlighted by the red, blue and green fonts, respectively.

	Attributes	IV	OPR	SV	OCC	DEF	MB	FM	IPR	OV	BC	LR
DP	KCF	0.713	0.677	0.640	0.630	0.627	0.598	0.629	0.693	0.494	0.712	0.554
	DSST	0.715	0.662	0.650	0.610	0.555	0.565	0.575	0.691	0.477	0.703	0.561
	SAMF	0.702	0.750	0.713	0.739	0.697	0.643	0.669	0.717	0.619	0.686	0.679
	MEEM	0.746	0.812	0.756	0.768	0.786	0.729	0.779	0.794	0.681	0.746	0.625
	MUSTer	0.779	0.766	0.728	0.760	0.716	0.676	0.716	0.773	0.587	0.784	0.667
	LCT	0.743	0.768	0.698	0.704	0.715	0.667	0.713	0.781	0.587	0.734	0.531
	Staple	0.787	0.759	0.744	0.749	0.777	0.706	0.730	0.770	0.658	0.766	0.625
	SRDCF	0.786	0.748	0.747	0.736	0.737	0.765	0.779	0.744	0.593	0.775	0.649
	CSR-DCF	0.779	0.769	0.743	0.703	0.792	0.729	0.773	0.775	0.666	0.768	0.672
	LMCF	0.795	0.784	0.746	0.757	0.753	0.728	0.756	0.755	0.689	0.822	0.673
	BACF	0.826	0.796	0.779	0.750	0.787	0.764	0.824	0.795	0.761	0.830	0.734
	MCCT-H	0.800	0.840	0.813	0.802	0.845	0.779	0.801	0.798	0.766	0.854	0.681
	DMS	0.826	0.856	0.831	0.818	0.833	0.809	0.816	0.838	0.778	0.854	0.689
	Attributes	IV	OPR	SV	OCC	DEF	MB	FM	IPR	OV	BC	LR
AUC	KCF	0.481	0.456	0.397	0.446	0.444	0.464	0.469	0.468	0.397	0.503	0.307
	DSST	0.564	0.487	0.482	0.469	0.433	0.475	0.469	0.507	0.390	0.530	0.387
	SAMF	0.531	0.544	0.497	0.552	0.517	0.524	0.521	0.522	0.490	0.534	0.435
	MEEM	0.526	0.537	0.480	0.521	0.505	0.562	0.561	0.533	0.492	0.525	0.366
	MUSTer	0.609	0.558	0.529	0.581	0.550	0.550	0.563	0.557	0.474	0.589	0.459
	LCT	0.573	0.560	0.505	0.531	0.524	0.540	0.567	0.563	0.457	0.557	0.357
	Staple	0.605	0.550	0.537	0.565	0.572	0.553	0.558	0.558	0.487	0.583	0.421
	SRDCF	0.620	0.558	0.568	0.566	0.549	0.604	0.610	0.550	0.467	0.592	0.502
	CSR-DCF	0.602	0.555	0.534	0.538	0.573	0.585	0.591	0.549	0.503	0.570	0.419
	LMCF	0.609	0.573	0.538	0.573	0.544	0.569	0.574	0.549	0.545	0.615	0.455
	BACF	0.653	0.597	0.587	0.589	0.596	0.594	0.624	0.591	0.559	0.635	0.537
	MCCT-H	0.632	0.625	0.604	0.622	0.631	0.618	0.618	0.594	0.584	0.658	0.484
	DMS	0.644	0.625	0.607	0.621	0.613	0.627	0.621	0.605	0.588	0.653	0.488

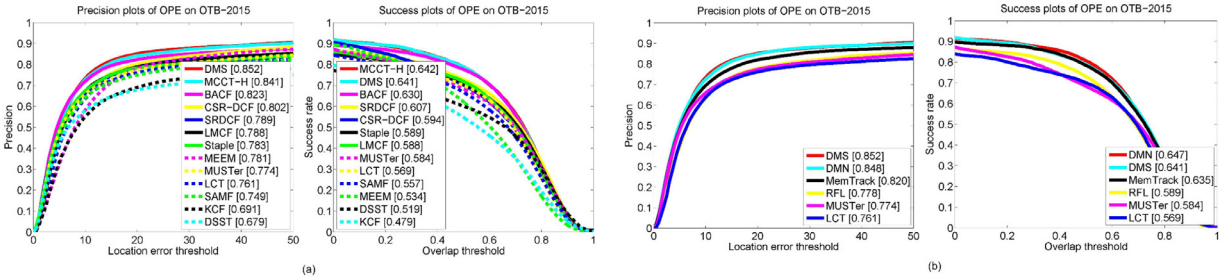

Fig. 5. Overall performance evaluation on the OTB-2015 benchmark with (a) state-of-the-art trackers, (b) some related trackers. The DP scores and AUC scores are shown in the legend of the precision plots (left) and success plots (right) to rank all compared trackers.

Attribute-based performance. We report the DP scores and AUC scores of our method and 12 other compared methods under different attributes in Table 3. As shown in Table 3, our method achieves the best results on 8 of the 11 attributes in terms of DP scores and impressive results on all attributes in terms of AUC scores. These outstanding results demonstrate the robustness of the proposed method regarding different challenging attributes. Specifically, for the sequences with

Table 4
Comparisons of the proposed tracker with some state-of-the-art deep learning trackers on the OTB-2013 and OTB-2015 benchmarks. The DP scores and AUC scores are used to rank all compared trackers. The top three performances are highlighted by the red, blue and green fonts, respectively.

	Trackers	CNT	SiamFc	DeepSRDCF	CNN-SVM	ACFN	CFnet	HDT	MCPF	PTAV	DMS
OTB-2013	DP	0.724	0.809	0.849	0.852	0.860	0.807	0.889	0.916	0.894	0.865
	AUC	0.551	0.616	0.641	0.597	0.607	0.611	0.603	0.677	0.663	0.664
OTB-2015	DP	0.577	0.772	0.851	0.814	0.802	0.748	0.848	0.873	0.849	0.852
	AUC	0.451	0.583	0.635	0.554	0.575	0.568	0.564	0.628	0.635	0.641

occlusion and out-of-view attributes, the DMS tracker performs favorably against other state-of-the-art trackers such as BACF and Staple. This result is mainly because the DMS tracker enhances the long-term memory of target appearance via the memory-improved update model. For the sequences with deformation and in-plane rotation attributes, the DMS tracker obtains high scores in both DP and AUC. This can be attributed to the maintenance of the short-term memory, which is an indispensable information resource for adapting to the rapid appearance variations of the target. Moreover, with the help of the cooperation between short-term memory and long-term memory in the proposed DMS model, our tracker achieves excellent adaptivity and robustness to deal with various challenging factors, e.g., fast motion, background clutter and motion blur.

4.2.4. Qualitative comparisons

Fig. 6 shows some qualitative results of the proposed DMS tracker and several state-of-the-art trackers, including SAMF, MEEM, MUSTer, SRDCF, Staple, and MCCT-H, on 9 challenging video sequences. These sequences are obtained from the OTB-2015 dataset with various challenging factors. The MEEM tracker contains multiple experts with different target states and performs robustly in motion blur (*Blurowl*), but it fails to adapt to the scale variation (*Human2*) because it does not equip a scale estimator. Although the SAMF tracker is able to deal with the scale variation (*Box*) with a scale adaptive kernelized correlation filter, it performs poorly in fast motion (*Blurowl*) due to the limited search area. The SRDCF tracker increases negative samples and the search area by exploiting training and testing samples with larger spatial supports, leading to excellent performance in background clutter (*Soccer*) and fast motion (*Blurowl*). However, it cannot accurately locate the target undergoing heavy occlusions (*Girl2* and *Box*). This is because the SRDCF tracker aggressively updates the target model without careful checking. With the introduction of long-term memory, the MUSTer tracker is able to recover from tracking failures caused by heavy occlusions (*Girl2* and *Box*). However, it fails to adapt to the significant appearance changes, such as out-of-plane and in-plane rotation (*Dragonbaby* and *Human2*). The Staple tracker is robust to deformation (*Singer2* and *Tiger2*) with the combination of HOG and color features but is less effective in handling motion blur (*Blurowl*) and background clutter (*Shaking*). The MCCT-H tracker achieves high robustness in most challenging video sequences except for the *Soccer* sequence, where the target undergoes various appearance changes, e.g., variations in illumination and scale, background clutter, occlusions and so on. Overall, our tracker performs favorably in all these sequences. By strategically utilizing the short-term memory and long-term memory of the target appearance with the DMS model, our tracker performs adaptively and robustly to deal with various challenging factors.

4.2.5. Comparisons with deep learning trackers

To provide a comprehensive evaluation, we compare our DMS method with 9 deep-learning-based tracking methods on the OTB-2013 and OTB-2015 datasets. The comparison results can be found in Table 4 with the DP scores and AUC scores. On the OTB-2013 dataset, our method has a certain gap compared with the best-performing method MCPF (with DP of 91.6% and AUC of 67.7%) and achieves results that are comparable with those of the PTAV method (with DP of 89.4% and AUC of 66.3%) and HDT method (with DP of 88.9% and AUC of 60.3%). On the OTB-2015 dataset, our method ranks second and first on the DP and AUC metrics, respectively, showing impressive performance compared with other deep-learning-based methods. Note that our method only exploits the conventional handcrafted features, while others use powerful deep features to obtain a performance improvement. However, exacting deep features is very computationally expensive. Among all compared deep learning methods, only the SiamFc, CFnet and PTAV methods achieve real-time performance. However, the SiamFc and CFnet methods perform worse than our method by a large margin. Although the PTAV method performs slightly better than our method on the OTB-2013 dataset, it is less effective on the OTB-2015 dataset and runs slower than our method.

4.3. Evaluation on VOT benchmark

4.3.1. Experimental settings

Evaluation datasets. The VOT2015 and VOT2016 datasets are used to evaluate the proposed DMS tracker on the VOT benchmark. The VOT2015 dataset contains 60 video sequences that are selected from a large sequence pool via a fully automatic selection methodology. The VOT2016 dataset uses the same sequences as the VOT2015 dataset but with a further

Fig. 6. Some sampled tracking results of the proposed DMS tracker with 6 state-of-the-art trackers in the *Blurowl, Box, Dragonbaby, Girl2, Human2, Shaking, Singer2, Soccer, Tiger2* sequences, where the target undergoes vaiours challenging factors, e.g., heavy occlusions, background clutters and drasitic defomation.

refined annotation. These carefully selected video sequences are more challenging to track due to the optimized diversity in visual attributes.

Evaluation metrics. In contrast to the OTB evaluation, the VOT benchmark applies a reset-based evaluation mechanism to fully leverage the dataset, which means that the tested tracker will be reinitialized once a failure is detected. Following the protocol in [20,21], we exploit three evaluation metrics to analyze the tracking performance: accuracy, robustness and expected average overlap (EAO). The accuracy measures the average overlaps $A = \frac{1}{N_{valid}} \sum_{t=1}^{N_{valid}} O\left(B_p^t, B_g^t\right)$ between the predicted and ground-truth bounding boxes in valid frames. The robustness reflects the number of failures N_F during tracking, where a failure is detected when the overlap between the predicted bounding box and ground-truth bounding box is zero. The EAO decides the final performance ranks of the compared trackers, which is actually a no-reset average overlap estimator on the OTB evaluation but with reduced bias and variance. It is computed as the average of the expected average overlap curve values over a sequence length range $[N_{lo}, N_{hi}]$,

$$EAO = \frac{1}{N_{hi} - N_{lo}} \sum\nolimits_{N_S = N_{lo}:N_{hi}} \hat{\Phi}_{N_S}, \tag{18}$$

where $\hat{\Phi}_{N_S}$ averages the average overlaps $\Phi_{N_S} = \frac{1}{N_S} \sum_{t=1}^{N_S} O\left(B_p^t, B_g^t\right)$ of all N_S-frame-long sequences. In addition, the average overlap (AO) in the OTB benchmark is reused in the VOT2016 benchmark for extended evaluation.

Evaluation trackers. There are 62 and 70 trackers participating in the VOT2015 and VOT2016 challenges, respectively. For presentation clarity, we only compare our tracker with some baseline and top-ranked trackers in the corresponding challenge. These trackers come from various classes, e.g., correlation filters-based trackers, deep learning-based trackers and structured SVM-based trackers.

4.3.2. Evaluation on VOT2015

We show the sequence-pooled AR-raw and AR-rank plots on the VOT2015 benchmark in Fig. 7. The EAO plot is also shown in Fig. 7 for ranking the overall performance of our tracker and of other compared trackers. According to the AR-raw and AR-rank plots, the best accuracy and robustness are achieved by the MDNet tracker, which is also the winner of the VOT2015 challenge. We observe that our DMS tracker is close to the winner in terms of accuracy and achieves competitive performance among the top-ranked trackers in terms of robustness. In the EAO plot, our DMS tracker ranks third among all participating trackers in the VOT2015 challenge, where the previous three top-performing trackers are the MDNet, DeepSRDCF and EBT trackers, respectively. Note that both the first-ranked and the second-ranked trackers utilize deep features to obtain high performance.

4.3.3. Evaluation on VOT2016

The comparison results of our method with the top fifteen trackers on the VOT2016 benchmark are summarized in Table 5. According to the VOT2016 benchmark, any tracker that exceeds the average performance (0.255) of the 15 tested trackers can be considered state-of-the-art. Clearly, all compared trackers here are state-of-the-art. As shown in Table 5, our DMS tracker obtains an EAO of 0.314, which ranks fourth among all participating trackers in the VOT2016 challenge. In terms of accuracy, the proposed DMS tracker performs significantly better than most of the compared trackers but is slightly inferior to the best-performer SSAT. Regarding robustness, our tracker achieves performance comparable with that of deep learning trackers such as C-COT and TCNN. In addition, the AO score of our tracker is 0.451, which outperforms

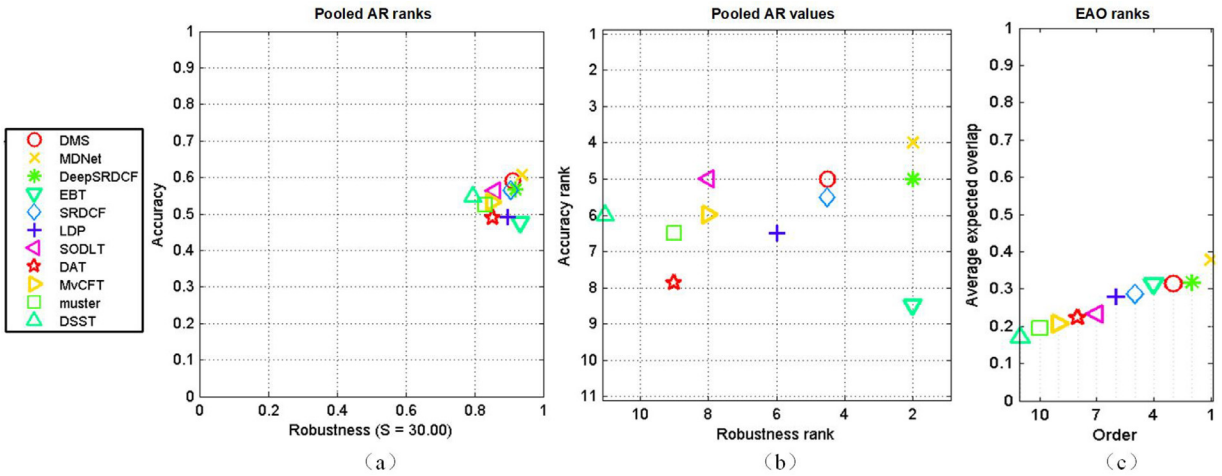

Fig. 7. Performance evaluation on the VOT2015 benchmark with (a) AR raw plots, (b) AR rank plots (the most upper tracker has the best accuracy rank and the most right tracker has the best robustness rank) and (c) expected average graph (the most right tracker is the best-performing).

Table 5
Performance evaluation on the VOT2016 benchmark in terms of EAO, accuracy, robustness and AO metrics.

Trackers	EAO	Accuracy	Robustness	AO
C-COT	0.331	0.539	0.238	0.469
TCNN	0.325	0.554	0.268	0.485
SSAT	0.321	0.577	0.291	0.515
DMS	**0.314**	**0.562**	**0.308**	**0.451**
MLDF	0.311	0.490	0.233	0.428
Staple	0.295	0.544	0.378	0.388
DDC	0.293	0.541	0.345	0.391
EBT	0.291	0.465	0.252	0.370
SRBT	0.290	0.496	0.350	0.333
STAPLE+	0.286	0.557	0.368	0.392
DNT	0.278	0.515	0.329	0.427
SSKCF	0.277	0.547	0.373	0.391
SiamFC-R	0.277	0.549	0.382	0.421
DeepSRDCF	0.276	0.528	0.326	0.427
SHCT	0.266	0.547	0.396	0.392
MDNet_N	0.257	0.541	0.337	0.457

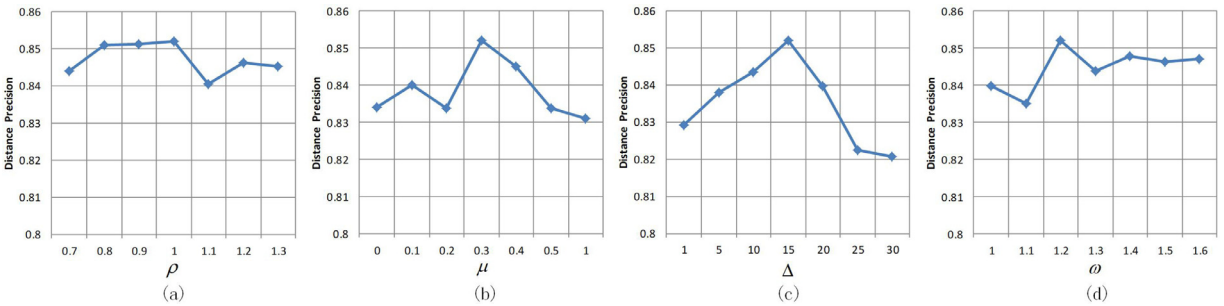

Fig. 8. Effects of (a) ρ, (b) μ, (c) Δ and (d) ω with different values.

the correlation-filter-based trackers Staple and DeepSRDCF by 6.3% and 2.4%. Overall, our proposed DMS tracker achieves state-of-the-art performance on the VOT2016 benchmark in various aspects.

4.4. Parameter analysis

In this section, we analyze the effects of several important parameters on the tracking performance using DP scores. These parameters include the learning factor ρ in Eq. (11), the tradeoff parameter μ in Eq. (17), the temporal window Δ and the weighting factor ω in the weight sequence w. All parameter experiments are conducted on the OTB-2015 dataset.

(1) Effect of ρ: The parameter ρ in Eq. (11) is a learning factor that decides the basic learning rate of the long-term tracker. Combined with ρ, the relative tracking quality measurement Q^t adaptively adjusts the learning rate in frame t to reduce the effects of corrupted tracking results. A larger ρ means that more information of subsequent frames can be updated in the long-term target model and vice versa. Fig. 8(a) depicts the corresponding DP scores when we set ρ to 0.7, 0.8, 0.9, 1.0, 1.1, 1.2 and 1.3. We observe that the best tracking performance is achieved by setting a moderate value of ρ to 1.0.

(2) Effect of μ: The parameter μ in Eq. (17) is a tradeoff between credibility and discriminability measurements, which play different roles in the reliability evaluation function. The credibility measures the target likelihood of the current tracking result of a tracker, whereas the discriminability measures the capacity to distinguish the target from the background of a tracker. To better balance these two important measurements, we analyze how the tracking performance of the proposed DMS tracker is influenced by different μ. As shown in Fig. 8(b), the lack of any measurement in the reliability evaluation function leads to a degraded tracking performance when we set μ to 0 or 1. The highest DP score was obtained with μ at 0.3.

(3) Effects of Δ and ω: To avoid performance fluctuations in the reliability evaluation function, we introduce the temporal context information for keeping more stable measurements with a weight sequence w. There are two parameters Δ and ω in the weight sequence w. On the one hand, Δ controls the size of the temporal window, and more historical frames will be considered with the increase in Δ. Fig. 8(c) shows the DP scores for different Δ. We can easily find that the tracking performance significantly decreased when we only consider the current frame with Δ at 1. The result indicates the importance of temporal context information for maintaining performance stability. In addition, we find

that too much historical information also causes poor tracking performance when we increase Δ to 30. Therefore, we set Δ to 15 in our experiments because it achieves the greatest performance. On the other hand, ω decides the importance of each historical frame. It assigns larger weights to recent frames as recent measurements are more valuable for performing the reliability evaluation. Particularly, equal weights will be assigned for each historical frame by setting $\omega = 1$. As shown in Fig. 8(d), the DP score is maximized by setting ω to 1.2.

5. Conclusion

In this paper, we consider both the short-term memory and long-term memory of the target appearance for enhancing the adaptivity and robustness of visual tracking and further propose a DMS model to select a reliable memory pattern to handle the current tracking challenges. Specifically, we establish a memory tracker for each memory pattern based on DCFs. Furthermore, to perform a robust reliability evaluation for memory selection, an MEC is presented by considering the credibility and discriminability of each memory tracker with temporal continuity. Comprehensive experimental comparisons and analyses are conducted on multiple tracking benchmarks to demonstrate the superiority of our method against other state-of-the-art methods.

Declaration of Competing Interest

We declare that we have no conflicts of interest to this work.

CRediT authorship contribution statement

Guiji Li: Conceptualization, Methodology, Software, Validation, Writing - original draft. **Manman Peng:** Writing - original draft, Supervision, Project administration. **Ke Nai:** Validation, Formal analysis. **Zhiyong Li:** Investigation, Writing - review & editing. **Keqin Li:** Writing - review & editing.

Acknowledgements

This work is supported by the National key R&D Program of China (Grant 2017YFB0202901, 2017YFB0202905). This work is supported by the National Nature Science Foundation of China(Grant Number: 61906167). The corresponding author of this paper is Manman Peng (pengmanman@hnu.edu.cn).

References

[1] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, P.H.S. Torr, 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, 2016, pp. 1401–1409.
[2] L. Bertinetto, J. Valmadre, J.F. Henriques, A. Vedaldi, P.H.S. Torr, 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, 2016, pp. 850–865.
[3] D.S. Bolme, J.R. Beveridge, B.A. Draper, Y.M. Lui, 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, 2010, pp. 2544–2550.
[4] K. Chen, W. Tao, S. Han, Visual object tracking via enhanced structural correlation filter, Inf. Sci. 394 (2017) 232–245.
[5] S. Chen, B. Liu, C.W. Chen, A structural coupled-layer tracking method based on correlation filters, in: MultiMedia Modeling - 23rd International Conference, MMM 2017, Reykjavik, Iceland, January 4–6, 2017, Proceedings, Part I, 2017, pp. 65–76.
[6] J. Choi, H.J. Chang, S. Yun, T. Fischer, Y. Demiris, J.Y. Choi, Attentional correlation filter network for adaptive visual tracking, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21–26, 2017, 2017, pp. 4828–4837, doi:10.1109/CVPR.2017.513.
[7] M. Danelljan, G. Häger, F.S. Khan, M. Felsberg, Accurate scale estimation for robust visual tracking, in: British Machine Vision Conference, BMVC 2014, Nottingham, UK, September 1–5, 2014, 2014.
[8] M. Danelljan, G. Häger, F.S. Khan, M. Felsberg, 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, 2015, pp. 621–629, doi:10.1109/ICCVW.2015.84.
[9] M. Danelljan, G. Häger, F.S. Khan, M. Felsberg, Learning spatially regularized correlation filters for visual tracking, in: 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7–13, 2015, 2015, pp. 4310–4318.
[10] M. Danelljan, F.S. Khan, M. Felsberg, J. van de Weijer, Adaptive color attributes for real-time visual tracking, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23–28, 2014, 2014, pp. 1090–1097.
[11] H. Fan, H. Ling, Parallel tracking and verifying: A framework for real-time and high accuracy visual tracking, in: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22–29, 2017, 2017, pp. 5487–5495, doi:10.1109/ICCV.2017.585.
[12] H.K. Galoogahi, A. Fagg, S. Lucey, Learning background-aware correlation filters for visual tracking, in: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22–29, 2017, 2017, pp. 1144–1152, doi:10.1109/ICCV.2017.129.
[13] J.F. Henriques, R. Caseiro, P. Martins, J. Batista, High-speed tracking with kernelized correlation filters, IEEE Trans. Pattern Anal. Mach. Intell. 37 (3) (2015) 583–596, doi:10.1109/TPAMI.2014.2345390.
[14] C. Hong, J. Yu, D. Tao, M. Wang, Image-based three-dimensional human pose recovery by multiview locality-sensitive sparse retrieval, IEEE Trans. Industr. Electron. 62 (6) (2015) 3742–3751, doi:10.1109/TIE.2014.2378735.
[15] C. Hong, J. Yu, J. Zhang, X. Jin, K.-H. Lee, Multimodal face-pose estimation with multitask manifold deep learning, IEEE Trans. Industr. Inform. 15 (7) (2019) 3952–3961.
[16] S. Hong, T. You, S. Kwak, B. Han, 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, 2015, pp. 597–606.
[17] Z. Hong, Z. Chen, C. Wang, X. Mei, D.V. Prokhorov, D. Tao, Multi-store tracker (muster): A cognitive psychology inspired approach to object tracking, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7–12, 2015, 2015, pp. 749–758, doi:10.1109/CVPR.2015.7298675.
[18] Z. Kalal, K. Mikolajczyk, J. Matas, Tracking-learning-detection, IEEE Trans. Pattern Anal. Mach. Intell. 34 (7) (2012) 1409–1422, doi:10.1109/TPAMI.2011.239.

[19] H. Kim, R. Park, Residual LSTM attention network for object tracking, IEEE Signal Process. Lett. 25 (7) (2018) 1029–1033, doi:10.1109/LSP.2018.2835768.
[20] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R.P. Pflugfelder, L. Cehovin, T. Vojír, G. Häger, The visual object tracking VOT2016 challenge results, in: Computer Vision - ECCV 2016 Workshops - Amsterdam, The Netherlands, October 8–10 and 15–16, 2016, Proceedings, Part II, 2016, pp. 777–823.
[21] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, L. Cehovin, G. Fernández, T. Vojír, G. Häger, G. Nebehay, R.P. Pflugfelder, The visual object tracking VOT2015 challenge results, in: 2015 IEEE International Conference on Computer Vision Workshop, ICCV Workshops 2015, Santiago, Chile, December 7–13, 2015, 2015, pp. 564–586.
[22] Y. Kuai, G. Wen, D. Li, Masked and dynamic siamese network for robust visual tracking, Inf. Sci. 503 (2019) 169–182, doi:10.1016/j.ins.2019.07.004.
[23] G. Li, M. Peng, K. Nai, Z. Li, K. Li, Visual tracking via context-aware local sparse appearance model, J. Visual Commun. Image Represen. 56 (2018) 92–105.
[24] G. Li, M. Peng, K. Nai, Z. Li, K. Li, Multi-view correlation tracking with adaptive memory-improved update model, Neural Comput. Appl. (2019), doi:10.1007/s00521-019-04413-4.
[25] P. Li, D. Wang, L. Wang, H. Lu, Deep visual tracking: review and experimental comparison, Pattern Recognit 76 (2018) 323–338.
[26] Y. Li, J. Zhu, 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, 2014, pp. 254–265, doi:10.1007/978-3-319-16181-5_18.
[27] C. Liu, P. Liu, W. Zhao, X. Tang, Robust tracking and redetection: collaboratively modeling the target and its context, IEEE Trans. Multimedia 20 (4) (2018) 889–902, doi:10.1109/TMM.2017.2760633.
[28] A. Lukezic, T. Vojir, L.C. Zajc, J. Matas, M. Kristan, 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, 2017, pp. 4847–4856.
[29] C. Ma, X. Yang, C. Zhang, M. Yang, Long-term correlation tracking, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7–12, 2015, 2015, pp. 5388–5396.
[30] M. Mueller, N. Smith, B. Ghanem, Context-aware correlation filter tracking, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21–26, 2017, 2017, pp. 1387–1395.
[31] K. Nai, Z. Li, G. Li, S. Wang, Robust object tracking via local sparse appearance model, IEEE Trans. Image Processing 27 (10) (2018) 4958–4970.
[32] K. Nai, D. Xiao, Z. Li, S. Jiang, Y. Gu, Multi-pattern correlation tracking, Knowl.-Based Syst. (2019), doi:10.1016/j.knosys.2019.05.032.
[33] Y. Qi, S. Zhang, L. Qin, H. Yao, Q. Huang, J. Lim, M. Yang, Hedged deep tracking, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016, 2016, pp. 4303–4311, doi:10.1109/CVPR.2016.466.
[34] J. Valmadre, L. Bertinetto, J.F. Henriques, A. Vedaldi, P.H.S. Torr, 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, 2017, pp. 5000–5008.
[35] M. Wang, Y. Liu, Z. Huang, Large margin object tracking with circulant feature maps, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21–26, 2017, 2017, pp. 4800–4808.
[36] N. Wang, W. Zhou, Q. Tian, R. Hong, M. Wang, H. Li, Multi-cue correlation filters for robust visual tracking, in: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18–22, 2018, 2018, pp. 4844–4853.
[37] Y. Wu, J. Lim, M. Yang, Online object tracking: A benchmark, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23–28, 2013, 2013, pp. 2411–2418, doi:10.1109/CVPR.2013.312.
[38] Y. Wu, J. Lim, M. Yang, Object tracking benchmark, IEEE Trans. Pattern Anal. Mach. Intell. 37 (9) (2015) 1834–1848.
[39] T. Yang, A.B. Chan, Recurrent filter learning for visual tracking, in: 2017 IEEE International Conference on Computer Vision Workshops, ICCV Workshops 2017, Venice, Italy, October 22–29, 2017, 2017, pp. 2010–2019, doi:10.1109/ICCVW.2017.235.
[40] T. Yang, A.B. Chan, Learning dynamic memory networks for object tracking, in: Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part IX, 2018, pp. 153–169, doi:10.1007/978-3-030-01240-3_10.
[41] T. Yang, A.B. Chan, Visual tracking via dynamic memory networks, IEEE Trans. Pattern Anal. Mach. Intell. (2019), doi:10.1109/TPAMI.2019.2929034.
[42] J. Yu, Y. Rui, D. Tao, Click prediction for web image reranking using multimodal sparse coding, IEEE Trans. Image Processing 23 (5) (2014) 2019–2032, doi:10.1109/TIP.2014.2311377.
[43] J. Yu, M. Tan, H. Zhang, Y. Rui, D. Tao, Hierarchical deep click feature prediction for fine-grained image recognition, IEEE Trans. Pattern Anal. Mach. Intell. (2019), doi:10.1109/TPAMI.2019.2932058.
[44] J. Yu, X. Yang, F. Gao, D. Tao, Deep multimodal distance metric learning using click constraints for image ranking, IEEE Trans. Cybernetics 47 (12) (2017) 4014–4024, doi:10.1109/TCYB.2016.2591583.
[45] J. Yu, C. Zhu, J. Zhang, Q. Huang, D. Tao, Spatial pyramid-enhanced netvlad with weighted triplet loss for place recognition, IEEE Trans. Neural Netw. Learning Syst. (2019), doi:10.1109/TNNLS.2019.2908982.
[46] J. Zhang, S. Ma, S. Sclaroff, 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, 2014, pp. 188–203.
[47] J. Zhang, J. Yu, D. Tao, Local deep-feature alignment for unsupervised dimension reduction, IEEE Trans. Image Process. 27 (5) (2018) 2420–2432, doi:10.1109/TIP.2018.2804218.
[48] K. Zhang, Q. Liu, Y. Wu, M. Yang, Robust visual tracking via convolutional networks without training, IEEE Trans. Image Processing 25 (4) (2016) 1779–1792, doi:10.1109/TIP.2016.2531283.
[49] T. Zhang, C. Xu, M. Yang, 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, 2017, pp. 4819–4827, doi:10.1109/CVPR.2017.512.
[50] D. Zhao, L. Xiao, H. Fu, T. Wu, X. Xu, B. Dai, Augmenting cascaded correlation filters with spatial-temporal saliency for visual tracking, Inf. Sci. 470 (2019) 78–93, doi:10.1016/j.ins.2018.08.053.