frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 06 September 2022
DOI 10.3389/fphy.2022.1003517

Multi-layer Rotation Memory Model-based correlation filter for visual tracking

Yufei Zhao[1], Yong Song[2,3]*, Guoqi Li[1], Lei Deng[1], Yashuo Bai[2,3] and Xiyan Wu[2,3]

[1]Department of Precision Instrument, Center for Brain-Inspired Computing Research, Tsinghua University, Beijing, China, [2]School of Optics and Photonics, Beijing Institute of Technology, Beijing, China, [3]Beijing Key Laboratory for Precision Optoelectronic Measurement Instrument and Technology, Beijing, China

Check for updates

OPEN ACCESS

EDITED BY
Yufei Ma,
Harbin Institute of Technology, China

REVIEWED BY
Jiamiao Yang,
Shanghai Jiao Tong University, China
Xina Cheng,
Xidian University, China

*CORRESPONDENCE
Yong Song,
yongsong@bit.edu.cn

SPECIALTY SECTION
This article was submitted to Optics and Photonics,
a section of the journal
Frontiers in Physics

RECEIVED 26 July 2022
ACCEPTED 10 August 2022
PUBLISHED 06 September 2022

CITATION
Zhao Y, Song Y, Li G, Deng L, Bai Y and Wu X (2022), Multi-layer Rotation Memory Model-based correlation filter for visual tracking.
Front. Phys. 10:1003517.
doi: 10.3389/fphy.2022.1003517

Object tracking technology is of great significance in laser image processing. However, occlusion or similar interference during visual object tracking may reduce the tracking precision or even cause tracking failure. Aiming at this issue, we propose a Multi-layer Rotation Memory Model-based Correlation Filter (MRMCF) for visual trackingin this paper. First, we establish a Multi-layer Rotation Memory (MRM) model, in which a set of three rotating concentric rings is used to simulate the three memory spacesand their updating processsimulate the memory spaces. Then we introduce the MRM model into the correlation filter tracking framework, which realizes realizing the dynamic updating of classifier parametersin the correlation filter. When the object is occluded or there is similar interference, the proposed tracker can use the Pre-occ classifier parameters stored in the memory spaces in the MRM model MRM memory spaces to retarget the object, thereby reducing the impact of these factors. The experimental results on the OTB50 dataset show that compared with trackers such as CNN-SVM, MEEM, Struck, etc., the proposed tracker achieves higher accuracy and success rate.

KEYWORDS

laser image processing, visual tracking, human visual system, memory model, correlation filter, object occlusion

1 Introduction

Object tracking technology has a wide range of application in laser spectroscopy including LiDAR (Light Detection And Ranging) image processing [1, 2], active laser detection [3, 4] and real-time laser tracker [5, 6], etc. Generally, object tracking is the task of estimating the state of an arbitrary object in each frame of a video sequence. In the most general setting, the object is only defined by its initial state in the sequence. Most current approaches address the tracking problem by constructing an object model, and these approaches are capable of differentiating between the object and background appearance [7, 8].

At present, there are mainly two types of object trackers: generative trackers and discriminative trackers. Among them, the basic idea of the generative trackers is to learn an object appearance model and search for the most similar area in the image as the object area [9–12]. The discriminative trackers (also called detection-based trackers) regard the tracking problem as a detection problem. This kind of tracker trains a classifier using the object and the background area of the current frame as the position and negative sample, respectively. And the trained classifier is adopted to find the optimal object area in the next frame.

In the classifier training process of discriminative trackers, different training methods can be used, such as Correlation Filter (CF) [13–15], Deep Learning (DL) [16–18], and Support Vector Machine (SVM) [19, 20]. Among them, CF has been widely used due to its advantages of high speed and good robustness. Specifically, in the CF method, a classifier is firstly learned from a set of training samples. Then the classifier is trained by performing a cyclic shift operation on the training sample, which allows the training and detection process can be performed in the Fourier domain. The amount of calculation is greatly reduced, thereby obtaining higher efficiency. However, CF only considers the samples of the current frame during each training. Therefore, when facing common problems such as occlusion, deformation, or background clutter [8, 21], the CF does not consider the influence of previously appeared samples, and the trained classifier is not robust enough, which may lead to tracking failure.

On the other hand, the memory mechanism in the Human Visual System (HVS) can extract old information stored in the memory space when a new similar one appears. Therefore, the memory mechanism has the potential to solve the problem of occlusion during the tracking process. In terms of object tracking, Ma et al. [22] proposed a tracker based on an adaptive CF with long-term memory and short-term memory, which achieves long-term stable memory of the appearance of the object; Wan et al. [23] introduced Long Short-Term Memory (LSTM) into the tracking process, obtaining good tracking results; Mikami et al. [24] adopted the memory model for face posture tracking, and obtained higher robustness in the complex background.

However, the above methods are based on machine memory. Unlike human memory, machine memory does not consider some key characteristics of the memory mechanism in HVS, such as the uncertainty, fuzziness, and associativity of human brain memory. When the object disappears for a long time or is interfered with by similar objects during the tracking process, the tracking accuracy will be greatly reduced.

In this paper, we established an MRM model to update the classifier parameters in the CF tracker. The MRM model consists of multiple layers of concentric rings, which simulate different levels of memory space. When multiple similar information exists in the outer ring, these data will be merged and enter the inner ring. This process simulates the memory from shallow to deep. At the same time, each ring rotates at a certain speed, which simulates the dynamic update of the information stored in the memory space. Furthermore, we proposed an MRM model-based CF tracker, which can dynamically update the classifier parameters in the CF tracker and enable the CF tracker to remember object features. When the object is occluded or interfered with by similar objects, the proposed tracker can use the reliable classifier parameters stored in the MRM model to relocate the object, thereby improving the anti-interference ability. Comparison experiments show that compared with 18 comparison trackers such as CNN-SVM, MEEM, Struck, etc., the proposed tracker has advantages in tracking accuracy and tracking success rate, the results on the OTB50 are improved 4.9% and 3.7% compared with CNN and SVM, respectively.

2 Related works

One can find various surveys that review the most current developments in visual tracking research in [8, 25]. In this section, only the works that are most relevant to our own are covered, including correlation tracking methods and memory models for visual tracking.

2.1 Correlation tracking

Due to DCFs' exceptional accuracy and efficiency, the object tracking community has been studying them extensively in recent years. The Minimum Output Sum of Squared Error (MOSSE) high-speed tracker, proposed by Bolme et al. in [26], can be regarded as the ground-breaking work that first applied correlation filters to visual tracking. Henriques et al. [13] utilized the circulant structure of training samples and HOG features to develop Kernelized Correlation Filters (KCFs) in the Fourier domain. To maintain a manageable computational cost, Danelljan et al. [27] introduced Color Name (CN) descriptors and also advanced a proposal for an adaptive dimensionality reduction technique. To manage scale fluctuations of the target, Danelljan et al. [28] introduced a Discriminative Scale Space Tracker (DSST). To reduce model drift, Mueller et al. [29] included global context information in the typical construction of CFs. Ma et al. [30]proposed a Long-term Correlation Tracking (LCT) framework featuring a redetection module. When a tracking failure took place in this system, the redetector was engaged to retrace the target's location. By merging the HOG template model with the color histogram model, Bertinetto et al. [31] created the Staple algorithm, which improved the tracking robustness.

2.2 Memory model for visual tracking

Due to its ability to handle sequential input and acquire long-term dependencies, the recent and well-liked Long Short-Term

Memory (LSTM) network demonstrated significant promise in visual tracking. By fusing an LSTM and a residual framework, Kim et al. [32] created an RLSTM tracker for spatiotemporal attention learning. Through the use of an LSTM network, Yang et al. [33] learned a recurrent filter and modified it to account for target appearance fluctuations. To increase the precision of template-matching trackers, a dynamic memory network was developed in [34], where the LSTM was used to maintain target appearance variations with an accessible memory.

3 MRM model

3.1 Memory mechanism

To simulate memory, the process by which the human brain encodes, stores, and extracts from the received information, Atkinson and Shiffrin [35, 36] proposed the multi-store model. They believe that the received information will experience three stages of memory, i.e., sensory memory, short-term memory, and long-term memory. In each stage, the information will go through the process of encoding, storage, and extraction. Meanwhile, information that is rarely used or extracted will be forgotten.

Among the three memory spaces, sensory memory space stores basic sensory information, which is the first step of human brain memory. Short-term memory space stores and processes complex information, which is the main space for information processing. Long-term memory space stores a large amount of prior knowledge, which enables the human brain to recall various events and recognize various patterns. Only the information that is repeatedly appeared in the short-term memory space can be transferred to the long-term memory space for storage.

However, compared with human brain memory, existing memory models lack some key functions. For example: (1) The information extracted by human brain memory is often vague, especially when the extracted information occurred a long time ago; (2) Human brain memory always links multiple related information, the human brain searches the memory space for information related to the received information and merges them; (3) The quality of the information retrieved by the human brain memory is usually related to the time and effort spent in memorizing the information; (4) In some cases, the human brain cannot recall a message at a certain time, but it may succeed after a while.

In short, the memory mechanism in HVS has some key characteristics:

Uncertainty At a certain moment, only part of the information stored in the human brain can be retrieved, but not all the information is clearly presented in the brain;

Fuzziness Similar information in the memory space will be merged, resulting in that the stored information will gradually become blurred over time;

Associativity The human brain memory will associate the related old information in the memory space with the newly received information.

Existing memory models simply simulate the memory mechanism of the human brain, but not the characteristics of uncertainty, fuzziness, and associativity. This limits its application, such as difficulty in solving the problems of object occlusion and similar object interference during tracking.

3.2 Structure of MRM model

Based on the above analysis, we established an MRM model as shown in Figure 1.

The MRM model includes a three-layer concentric ring, a Filter unit, two Compare and Merge units, a Compare unit, and some Data Storage units (including Memorized Data units and Empty Data units). At the same time, there are three different types of windows in each of the three layers of concentric rings, namely the Input window, Output window, and Observation window.

The main functions of each component are as follows:

Three-layer concentric ring: Simulates the different memory spaces of the human brain and the dynamic update of information in the human brain memory space. Each layer of concentric rings simulates a memory space. From the outer to the inner layer, they are sensory, short-term, and long-term memory space. At the same time, each layer of concentric rings rotates at a certain speed, and the speed from the outer layer to the inner layer is v_1, v_2, and v_3.

Filter unit: Filters the information entering the memory space. The filter unit compares the input information with the information that exists in the outer concentric ring (i.e., the sensory memory space). When the distance between the input information and any information in the sensory memory space is within the tolerance distance, the input information can enter the memory space.

Compare and Merge unit: Screens the information in the shallow memory space and send the qualified information into the deep memory space. The specific process includes two operations: comparison and merging. Among them, the comparison operation simulates the associativity of the human brain, and the similarity is obtained by calculating the normalized Euclidean distance between the two pieces of information. The merging operation simulates the fuzziness of the human brain and realizes the fusion of information by calculating the average value of two or more pieces of information. There is one Compare and Merge unit between the outermost layer-the middle layer and the middle layer-the innermost layer, respectively.

Compare unit: Evaluates the similarity between the output information of each layer of memory space and the initial input information, ensuring that the most similar information to the initial input information is output. The specific process is the same as the comparison operation in the Compare and Merge unit.

Data Storage unit: Mainly used to store information. Each layer of concentric rings has multiple data storage units, including memorized data units and empty data units.

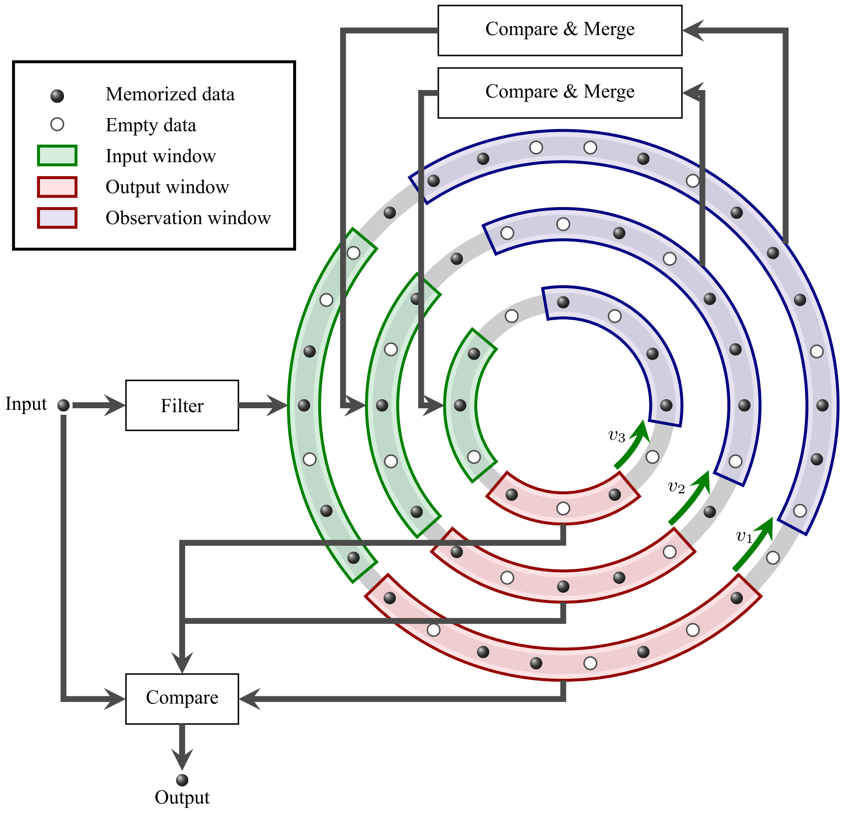

FIGURE 1
Schematic of MRM model, the input information enters three layers of concentric rings representing three different memory spaces through a filter unit, and the concentric rings rotate at different speeds, so that the input information can be transmitted to different locations in the memory space.

Window: Including input window, output window, and observation window. Three kinds of windows exist on each layer of the concentric ring. As the concentric ring rotates, any information will only appear in a specific window at any time. This simulates the uncertainty of human brain memory, that is, not all the stored information can be used at any time.

3.3 Workflow of MRM model

Figure 2 shows the workflow of the MRM model. The detailed process is as follows:

1) The filter unit evaluates whether the initial input information can enter the memory space. If it is permitted to enter, go to step (2). Otherwise, go to step (6);
2) The initial input information enters the sensory memory space and is named sensory input information. If there exists an empty data storage unit in the input window of the sensory memory space at the current moment, the sensory input information is directly stored in the corresponding unit. Otherwise, calculate the normalized Euclidean distance between the sensory input information and each piece of information in the input window, and merge it with the information with the smallest distance. At the same time, the normalized Euclidean distance between each information in the output window and the sensory input information is compared, and the information with the smallest distance will be let out as the sensory output information;
3) Use Compare and Merge unit to merge two or more pieces of information within the tolerance distance in the observation window of the sensory memory space. The merged information is fed into the short-term memory space, called short-term input information;
4) The process of inputting and outputting information in the short-term memory space is similar to step (2), and its output is called short-term output information;
5) Similar to step (3) and step (4), use Compare and Merge unit to merge two or more pieces of information within the tolerance distance in the observation window of the short-term memory space. Then input them into the long-term memory space, which is called long-term input information. At the same time, output long-term output information;

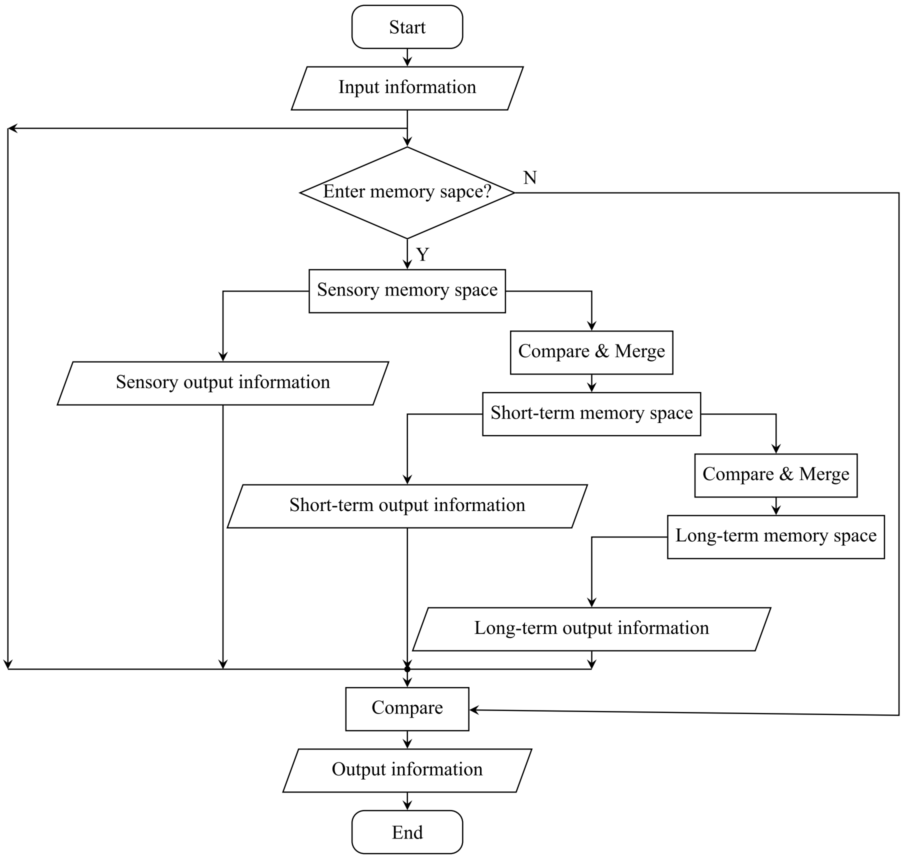

FIGURE 2
Workflow of the MRM model.

6) If there is no output in the three memory spaces, the initial input information is treated as the final output information. Otherwise, the distances between three output information and the initial input information are calculated separately, and the output information with the smallest distance from the initial input information is regarded as the final output information.

4 MRM-based CF tracker

After the establishment of the MRM model, we introduce it into the CF tracking framework and propose an MRM-based CF tracker.

4.1 CF tracking framework

A typical CF tracking framework mainly implements object tracking by repeating the detection-training-update process for each frame of the input image. When any frame of image is input, the search window of the current frame is first determined according to the predicted position in the previous frame, then the feature map of the search image is extracted. Next, the previously learned classifier is used to convolve the feature map to generate a response map. The position of the maximum value on the response map is regarded as the object position of the current frame. Finally, the classifier parameters are trained and updated according to the feature map at the current object position.

Let $(\hat{m}_{t-1}, \hat{n}_{t-1}, a, b)$ be the position and size information of the object in the $(t-1)$th frame of image, where $\hat{m}_{t-1}$ and $\hat{n}_{t-1}$ are the center coordinate of the tracking box, a and b are the width and height of the tracking box. Expand the tracking box to create a search window $(\hat{m}_{t-1}, \hat{n}_{t-1}, \rho a, \rho b)$ for the tth frame of image, where ρ is the expansion factor.

Extract the deep convolution feature map in the search window of the tth frame image, and use x_t to denote the cyclic shift of the feature map with size of $M \times N \times D \times L$ in the tth frame, where M, N, D, and L respectively represent the width, height, the number of channels and layer of the feature map. Then x_t $[d, l]$ represents the feature map of channel d in the lth layer of the tth frame of image, where $d \in \{1, \ldots, D\}$, $l \in \{1, \ldots, L\}$.

For the feature map of the lth layer, the corresponding classifier $w_{t-1}[d,l]$ of the $(t-1)$th frame image and the feature map $x_t[d,l]$ of the tth frame image are respectively subjected to Fourier transform, after the dot multiplication, sum along the channel, and the sub-response map $f_t[l]$ of this layer can be obtained through inverse Fourier transform, as shown in Eq. 1,

$$f_t[l] = \mathcal{F}^{-1}\left(\sum_{d=1}^{D}\mathcal{F}(w_{t-1}[d,l]) \odot \mathcal{F}(x_t[d,l])\right), \tag{1}$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ represent DFT (Discrete Fourier Transform) and inverse DFT, respectively, and $\odot$ represents Hadamard product.

Then, take γ_l as the weight coefficient to add the sub-response maps $f_t[l]$ of all layers to get the total response map f_t, as shown in Eq. 2,

$$f_t = \sum_{l=1}^{L}\gamma_l \cdot f_t[l] = \sum_{l=1}^{L}\gamma_l\mathcal{F}^{-1}\left(\sum_{d=1}^{D}\mathcal{F}(w_{t-1}[d,l]) \odot \mathcal{F}(x_t[d,l])\right). \tag{2}$$

The position of the maximum value in the total response map f_t is the center position of the tracking box in the tth frame, as shown in Eq. 3,

$$(\hat{m}_t, \hat{n}_t) = \arg\max_{m,n} f_t(m,n), \tag{3}$$

where $(m, n) \in \{1, \ldots, M\}\times\{1, \ldots, N\}$.

Create a training sample set x_t' by cyclic shift at the object position $(\hat{m}_t, \hat{n}_t)$ of the tth frame image. Each sample has a 2-D Gaussian label, which can be expressed by Eq. 4,

$$y_{u,v} = \exp\left[-\frac{(u-M/2)^2 + (v-M/2)^2}{2\varepsilon^2}\right], \tag{4}$$

where $(u, v) \in \{1, \ldots, M\}\times\{1, \ldots, N\}$, and ε represents bandwidth.

Next, the new classifier $w_t'[l]$ of the lth layer in the tth frame image can be obtained by minimizing ℓ_2 loss function of the output $w_t[d,l]{*}x_t'[d,l]$ and the corresponding Gaussian label $y_{u,v}$, that is,

$$w_t'[l] = \arg\min_{w_t[l]}\left\|\sum_{d=1}^{D} w_t[d,l]{*}x_t'[d,l] - y\right\|^2 + \lambda\sum_{d=1}^{D}\|w_t[d,l]\|^2, \tag{5}$$

where λ is the regularization coefficient of ℓ_2, and $*$ represents the correlation operation, that is, the operation shown in Eq. 1.

It can be solved by DFT

$$w_t'[d,l] = \mathcal{F}^{-1}\left(\frac{\mathcal{F}(y)^* \odot \mathcal{F}(x_t'[d,l])}{\sum_{i=1}^{D}(\mathcal{F}(x_t'[i,l]))^* \odot \mathcal{F}(x_t'[i,l]) + \lambda}\right), \tag{6}$$

where * represents conjugation, and $\odot$ represents Hadamard product.

By performing the above operations on each frame of the input image, the position of the object in each frame will be obtained, and the classifier can be updated at the same time.

4.2 Design of MRM-based CF tracker

Two MRM models are introduced in the CF framework to form an MRM-based CF tracker, which enhances the classifier's ability to resist the occlusion of objects and interference from similar objects.

Specifically, in the MRM-based CF tracker, when processing a new image, after the classifier training process in the conventional CF framework, the trained classifier is input into the MRM model and updated according to the MRM model update rules. Then a reliable classifier for the current frame will be output. This reliable classifier integrates the characteristics of similar classifiers stored in the MRM model, and can better deal with object occlusion and interference from similar objects in the tracking process. The overall framework of the proposed algorithm is shown in Figure 3.

On the other hand, considering that the classifier includes a large number of parameters, its update process in the MRM model includes many operations, which will lead to excessive calculations and affect the tracking speed. Therefore, we design a "Command-Follow" mechanism to update the classifier, as shown in Figure 3. In the command channel, send the histogram feature into an MRM model to get its update process. Then, in the follow channel, the classifier can be updated in another MRM model only according to the same update process, without participating in the calculation. Since the data amount of the histogram feature is usually much smaller than the classifier, the dynamic update of the classifier can be realized with less calculation.

The specific steps of the MRM-based CF tracker are as follows:

(1) Initialization: At the first frame, initialize the search window and the two MRM models in the command and follow channels. Then extract the histogram feature q_1 and deep features x_1 of the search window, and train a classifier W_1 at the same time;
(2) Classifier updating: When tracking the tth $(t > 1)$ frame image, first extract the histogram feature of the search window in the $(t-1)$th frame q_{t-1}. Then, update q_{t-1} with the MRM model in the command channel. According to the "Command-Follow" mechanism, the classifier W_{t-1} can be updated with the same update process with the MRM model in the follow channel. Finally, a reliable classifier $\hat{W}_{t-1}$ will be obtained;
(3) Object locating: Extract the deep feature of the search window in the $(t-1)$th frame image x_{t-1}, and calculate the response map by Eq. 2. Then the object locating location of the tth frame image can be obtained by Eq. 3;

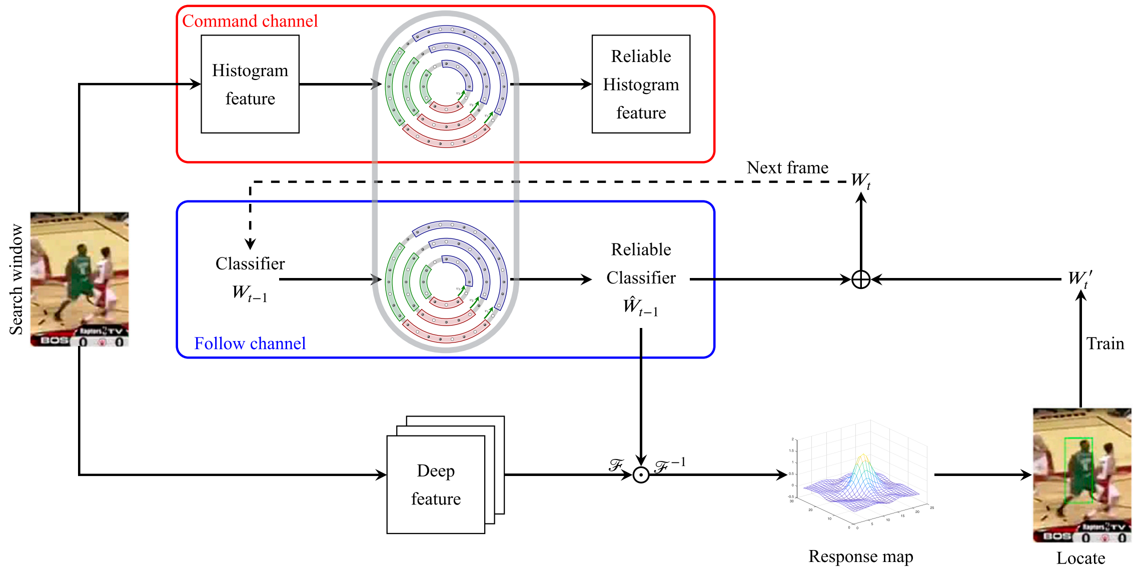

FIGURE 3
Pipeline of the MRM-based CF Tracker, the MRM model in the command channel gets the memory information update rule, and the follow channel updates the classifier in the CF tracker according to the rule, thereby improving the performance of the classifier.

TABLE 1 Parameter settings of the MRM model.

	Outer ring	Middle ring	Inner ring
Number of data storage	20	15	10
Rotation speed	2	1	1
Input window size	5	3	2
Observation window size	6	5	3
Output window size	5	5	3

(4) Classifier training: Train the classifier to obtain the new classifier of the tth frame image W_t, as shown in Eqss 5, 6.

5 Experiment

5.1 Experiment setup

A PC (Intel (R) Xeon E5-2620 (2.10 GHz) × 2 CPU, NVIDIA Quadro P2000 GPU, 64 GB memory) is used to carry out a comparative experiment of the proposed tracker.

The comparative experiment is based on the OTB50 [8] dataset, including 50 image sequences, each of which has different attributes. These image sequence attributes are factors that easily occur in the tracking process and affect the tracking accuracy. There are 11 types, namely: Illumination Variation (IV), Out-of-Plane Rotation (OPR), Scale Variation (SV), Occlusion (OCC), Deformation (DEF), Motion Blur (MB), Fast Motion (FM), In-Plane Rotation (IPR), Out-of-View (OV), Background Clutter (Background Clutter, BC) and Low Resolution (LR). In addition, the specific parameter settings of the memory space of each layer in the comparison experiment are shown in Table 1, and the tolerance distance in the algorithm is set to 0.35.

The tracking precision and success rate were evaluated by precision plot and success plot, respectively.

The tracking precision is the percentage of frames whose estimated locations lie in a given threshold distance to ground-truth centers. By setting a series of different thresholds, the corresponding tracking precision values can be calculated to generate a curve, i.e., a precision plot. Generally, the value obtained when the threshold is 20 pixels is treated as the tracking precision of the tracker.

As for the success rate, let axdenote the area of the tracking box and bydenote the ground truth. An Overlap Score (OS) can be defined by $OS = |a \cap b|/|a \cup b| OS = |x \cap y|/|x \cup y|$where $\cap$ and $\cup$ are the intersection and union of two regions, and |X| counts the number of pixels in the corresponding area X. Afterward, a frame whose OS is larger than a certain threshold is referred to as a successful frame, and the ratios of successful frames at the thresholds ranging from 0 to one are plotted in success plots. Generally, the value when the threshold is 0.5 is used as the tracking success rate of the tracker.

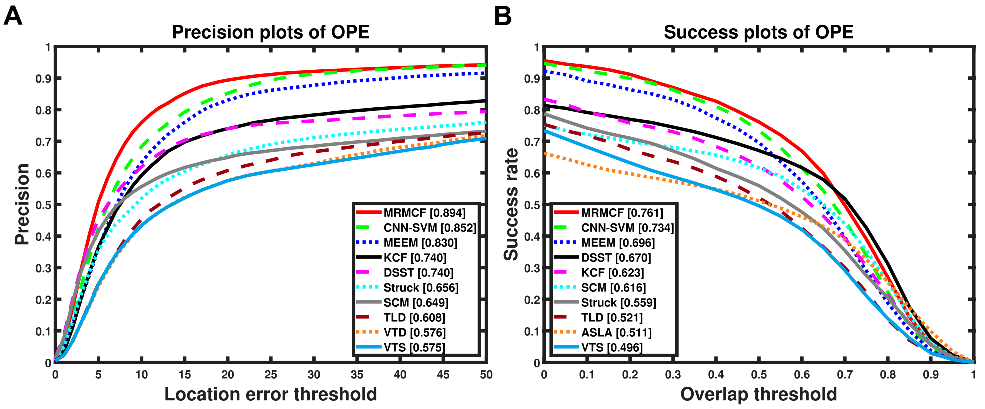

FIGURE 4
OPE comparison results of the proposed tracker and the comparison methods on OTB50. **(A)** Precision plots, **(B)** Success plots.

5.2 Results

5.2.1 Overall performance

The proposed MRMCF is compared with 18 trackers, including CNN-SVM [37], MEEM [20], KCF [13], DSST [28], Struck [38], SCM [39], TLD [40], VTD [41], VTS [42], CCT [43], ASLA [44], LSK [45], PCOM [46] etc.

Figure 4 is the overall comparison results under one pass evaluation (OPE). For readability, only the first 10 trackers are plotted. It can be seen that the tracking precision of the proposed MRMCF reaches 89.4%, and the tracking success rate reaches 76.1%, which is higher than comparison trackers, indicating that the proposed tracker has better tracking performance.

Figure 5 shows some of the tracking results of the proposed tracker and comparison trackers including sequences *coke*, *deer*, *football*, *freeman4*, *girl*, *lemming*, and *matrix* of the OTB50 [8]. In order to show the tracking boxes more clearly, only the top five trackers are shown in the figure.

Sequence *coke* has six attributes, including IV, OCC, FM, IPR, OPR, and BC. In the early stage of the tracking process (#0050 and #0116), the object keeps moving smoothly, so all five trackers are able to track the object steadily. Then, the object rotates in frame #0210, and the tracking boxes of DSST and MEEM drifted. Next, the object is blocked by a plant in frame #0260, and the tracking boxes of DSST and CNN-SVM drifted greatly. In the end, the object reappeared in frame #0270, while DSST, MEEM, and KCF could not find the object back;

Sequence *deer* has five attributes, including MB, FM, IPR, BC, and LR. Similarly, in the early stage, the five trackers all performed well (#0010). In the following tracking process, due to the fast movement of the object and the interference of similar objects, KCF lost the object in frames #0030, #0036, and #0050, DSST lost the object in frame #0030 and #0040, and the tracking box of MEEM has a small drift in #0040;

Sequence *football* has four attributes, including OCC, IPR, OPR and BC. In the beginning, all five trackers can track the object well (#0100). With the rapid movement of the object, the tracking boxes of DSST, MEEM, and KCF drifted (#0200). Due to the interference of similar objects, the tracking boxes of DSST, MEEM, CNN-SVM, and KCF have drifted (#0283 and #0290). Then, in frame #297, DSST, CNN-SVM, and KCF lost their targets, while the proposed MRMCF was not disturbed, maintaining good tracking performance.

Sequence *freeman4* has four attributes, including SV, OCC, IPR, OPR. Due to the object being occluded, object rotation, and other factors, the tracking boxes of the comparison trackers have drifted to different degrees. For example, in frame #0080, MEEM and KCF drifted, in frame #0150, DSST, CNN-SVM and KCF drifted, and MEEM drifted in a smaller range, and in frame #0270, all four comparison trackers drifted. Meanwhile, the proposed MRMCF maintained a good tracking result throughout the tracking process;

Sequence *girl* has four attributes, including SV, OCC, IPR and OPR. In the early stages, all the five trackers have good tracking performance, only DSST has a small range of drift (#0300 and #0420). When interfered with by a similar object, KCF, DSST, MEME, and CNN-SVM are affected. Among them, the tracking frame of KCF has a large drift, while DSST, MEME, and CNN-SVM have a small drift (#0438). With time, the tracking box of DSST also drifted widely (#0470 and #0490). In the whole process, the proposed MRMCF can achieve stable tracking without being disturbed by a similar object;

Sequence *lemming* has six attributes, including IV, SV, OCC, FM, OPR, and OV. Similarly, in the early stages of the tracking process, all five trackers perform well (#0200). When the object

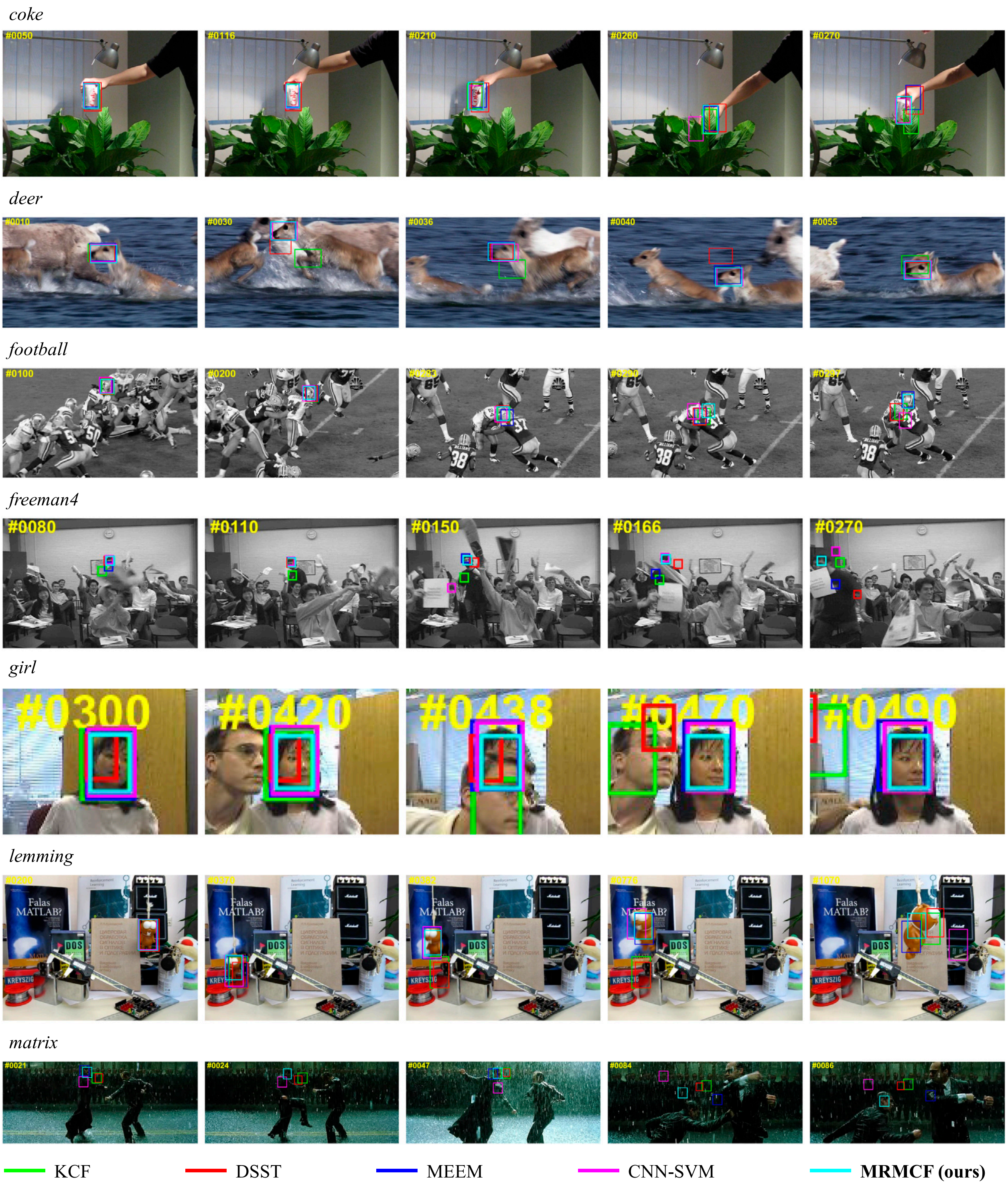

FIGURE 5
Part of tracking results of the proposed tracker and comparison trackers on the OTB50.

rotates in frame #0370, the tracking boxes of DSST, MEEM, and KCF all drift in a small range. Later, as the object moved quickly, DSST and KCF lost the object, and the tracking boxes of CNN-SVM and MEEM also drifted (#0382 and #0776). In frame #1070, the posture and scale of the object changed, and the tracking boxes of the five trackers could not fully contain the object, while

TABLE 2 AUC values of success plot corresponding to each attribute (%), the highest values are highlighted in bold.

	IV	OPR	SV	OCC	DEF	MB	FM	IPR	OV	BC	LR
MRMCF	**60.3**	**59.5**	**55.5**	**58.7**	62.7	**58.5**	**57.0**	**58.6**	56.5	**64.1**	**49.4**
CNN-SVM	55.6	58.2	51.3	56.3	**64.0**	56.5	54.5	57.1	57.1	59.3	46.1
MEEM	53.3	55.8	49.8	55.2	56.0	54.1	55.3	53.5	**60.6**	56.9	36.0
DSST	56.1	53.6	54.6	53.2	50.6	45.5	42.8	56.3	46.2	51.7	40.8
KCF	49.3	49.5	42.7	51.4	53.4	49.7	45.9	49.7	55.0	53.5	31.2
SCM	47.3	47.0	51.8	48.7	44.8	29.8	29.6	45.8	36.1	45.0	27.9
Struck	42.8	43.2	42.5	41.3	39.3	43.3	46.2	44.4	45.9	45.8	37.2
LSK	37.1	40.0	37.3	40.9	37.7	30.2	32.8	41.1	43.0	38.8	23.5
VTD	42.0	43.4	40.5	40.3	37.7	30.9	30.2	43.0	44.6	42.5	17.7
TLD	39.9	42.0	42.1	40.2	37.8	40.4	41.7	41.6	45.7	34.5	30.9
VTS	42.9	42.5	40.0	39.8	36.8	30.4	30.0	41.6	44.3	42.8	16.8
CCT	28.6	36.4	33.5	37.8	34.5	31.2	33.1	35.5	46.7	38.5	18.9
FOT	28.6	36.4	33.5	37.8	34.5	31.2	33.1	35.5	46.7	38.5	18.9
PCOM	28.6	36.4	33.5	37.8	34.5	31.2	33.1	35.5	46.7	38.5	18.9
ASLA	42.9	42.2	45.2	37.6	37.2	25.8	24.7	42.5	31.2	40.8	15.7

the position of the tracking box of the proposed MRMCF is relatively accurate;

Sequence *matrix* has seven attributes, including IV, SV, OCC, FM, IPR, OPR, and BC. In frame #0021, DSST, CNN-SVM, and KCF lost the object, and MEEM drifted in a small range. In frame #0024, DSST, CNN-SVM, and KCF lost their targets. In frame #0047, CNN-SVM also lost the target, while DSST, MEEM, and KCF drifted in a small range. In frame #0084, all trackers lost the target due to the rapid movement of the object. While in the following frame #0086, the proposed MRMCF retrieved the object again.

In summary, for the above-mentioned typical sequences, the proposed MRMCF tracker shows better performance.

5.2.2 Attribute-based evaluation

For detailed analyses, an attribute-based evaluation in OTB50 is also conducted. The Area Under Curve (AUC) scores of MRMCF and the comparison trackers under 11 image sequence attributes are shown in Table 2. The results demonstrate that MRMCF performs well on most attributes, especially on occlusion, scale variation, illumination variation, background clutter, and out-of-plane rotation, etc.

5.3 Ablation studies

We performed an ablation analysis for different MRM structures. As shown in Figure 1, the proposed MRM consists

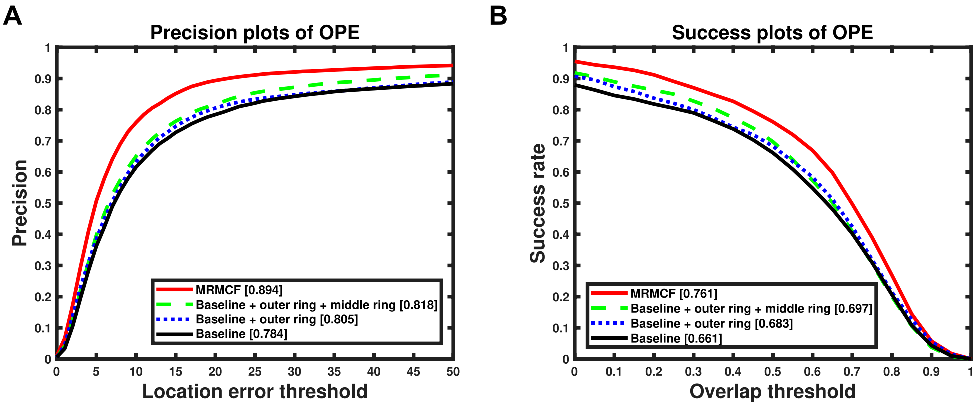

FIGURE 6
OPE comparison results of trackers with different configuration MRM models on OTB50. **(A)** Precision plots, **(B)** Success plots.

of three rings, representing the sensory, short-term and long-term memory space, respectively. In the ablation analysis, we take the CF tracker without the MRM model as the baseline tracker and compared it with three different MRM models with the following configurations: the MRM model including only the outer ring, the MRM model including the outer and middle rings and the MRM model including three rings (i.e. the proposed MRMCF), the results shown in Figure 6.

As shown in Figure 6, compared with the baseline tracker, the performance of the tracker incorporating the MRM model has improved. Moreover, the improvement of the tracker performance is limited by the incomplete MRM model, while the performance improvement of the tracker with the complete MRM model is very obvious.

5.4 generality analysis

The MRM model proposed in this paper is a relatively independent module in the whole tracker, which is mainly used to strengthen the connection with the previous classifier in the update process in the CF tracking framework. In most tracking-by-detection visual tracking methods, the classifier or object template update process is involved, so the proposed MRM model can be added to the update process of these trackers. Through reasonable parameter settings, performance improvement such as anti-occlusion similar to the tracker proposed in this paper can be finally achieved.

6 Conclusion

In this paper, an MRMCF tracker is proposed. Firstly, an MRM model based on the memory mechanism of HVS is established. By introducing the MRM model into the CF framework, the MRMCF tracker is formed, which realizes the dynamic update of CF classifier parameters. Under conditions such as occlusion or similar interferences, MRMCF can extract the reliable classifier parameters stored in the memory space of the MRM model to relocate the object, thereby achieving accurate object tracking. The experimental results based on the OTB50 show that compared with the comparison trackers, the proposed MRMCF has advantages in tracking precision and success rate, especially under various challenging conditions such as object occlusion and image clutter.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: http://cvlab.hanyang.ac.kr/tracker_benchmark/benchmark_v10.html.

Author contributions

YZ: Conceptualization, Methodology, Software, Validation, Writing-Original Draft. YS: Supervision, Project administration, Funding acquisition. GL: Supervision, Writing-review and editing, Investigation. LD: Software, Writing-review and editing, Visualization. YB: Data curation, Writing-review and editing. XW: Investigation.

Funding

This work was supported by the National Natural Science Foundation of China (NSFC) (81671787).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher’s note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Zhang J, Xiao W, Coifman B, Mills JP. Vehicle tracking and speed estimation from roadside lidar. *IEEE J Sel Top Appl Earth Obs Remote Sens* (2020) 13:5597–608. doi:10.1109/jstars.2020.3024921

2. Wang Y, Zhu Y, Liu H. Research on unmanned driving interface based on lidar imaging technology. *Front Phys* (2022) 10. doi:10.3389/fphy.2022.810933

3. He S, Meng Y, Gong M. Active laser detection system for recognizing surveillance devices. *Opt Commun* (2018) 426:313–24. doi:10.1016/j.optcom.2018.05.069

4. Li H, Zhang X. Laser reflection characteristics calculation and detection ability analysis of active laser detection screen instrument. *IEEE Trans Instrum Meas* (2021) 71:1–11. doi:10.1109/tim.2021.3129223

5. Sahu RK. A review on application of laser tracker in precision positioning metrology of particle accelerators. *Precision Eng* (2021) 71:232–49. doi:10.1016/j.precisioneng.2021.03.015

6. Iñigo B, Ibabe A, Aguirre G, Urreta H, López de Lacalle LN. Analysis of laser tracker-based volumetric error mapping strategies for large machine tools. *Metals* (2019) 9:757. doi:10.3390/met9070757

7. Marvasti-Zadeh SM, Cheng L, Ghanei-Yakhdan H, Kasaei S. *Deep learning for visual tracking: A comprehensive survey*. IEEE Transactions on Intelligent Transportation Systems (2021).

8. Wu Y, Lim J, Yang MH. Online object tracking: A benchmark. In: Proceedings of the IEEE conference on computer vision and pattern recognition; Washington, DC, USA. IEEE (2013). p. 2411–8.

9. Zhang T, Ghanem B, Liu S, Ahuja N. Robust visual tracking via multi-task sparse learning. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition; Washington, DC, USA. IEEE (2012). p. 2042–9.

10. Sevilla-Lara L, Learned-Miller E. Distribution fields for tracking. In: 2012 IEEE Conference on computer vision and pattern recognition; Washington, DC, USA. IEEE (2012). p. 1910–7.

11. Oron S, Bar-Hillel A, Levi D, Avidan S. Locally orderless tracking. *Int J Comput Vis* (2015) 111:213–28. doi:10.1007/s11263-014-0740-6

12. Zhang K, Zhang L, Liu Q, Zhang D, Yang MH. Fast visual tracking via dense spatio-temporal context learning. In: European conference on computer vision; Berlin, Heidelberg. Springer (2014). p. 127–41.

13. Henriques JF, Caseiro R, Martins P, Batista J. High-speed tracking with kernelized correlation filters. *IEEE Trans Pattern Anal Mach Intell* (2015) 37: 583–96. doi:10.1109/tpami.2014.2345390

14. Marvasti-Zadeh SM, Ghanei-Yakhdan H, Kasaei S. Adaptive exploitation of pre-trained deep convolutional neural networks for robust visual tracking. *Multimed Tools Appl* (2021) 80:22027–76. doi:10.1007/s11042-020-10382-x

15. Ma C, Huang JB, Yang X, Yang MH. Hierarchical convolutional features for visual tracking. In: Proceedings of the IEEE international conference on computer vision; Washington, DC, USA. IEEE (2015). p. 3074–82.

16. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition; Washington, DC, USA. IEEE (2016). p. 770–8.

17. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. *Computer Sci* (2014).

18. Zhang K, Liu Q, Yi W, Yang MH. Robust visual tracking via convolutional networks without training. *IEEE Trans Image Process* (2016) 25:1779–92. doi:10.1109/tip.2016.2531283

19. Hare S, Golodetz S, Saffari A, Vineet V, Cheng MM, Hicks SL, et al. Struck: Structured output tracking with kernels. *IEEE Trans Pattern Anal Mach Intell* (2015) 38:2096–109. doi:10.1109/tpami.2015.2509974

20. Zhang J, Ma S, Sclaroff S. Meem: Robust tracking via multiple experts using entropy minimization. In: European conference on computer vision; Berlin, Heidelberg. Springer (2014). p. 188–203.

21. Wu Y, Lim J, Yang MH. Object tracking benchmark. *IEEE Trans Pattern Anal Mach Intell* (2015) 37:1834–48. doi:10.1109/tpami.2014.2388226

22. Ma C, Huang JB, Yang X, Yang MH. Adaptive correlation filters with long-term and short-term memory for object tracking. *Int J Comput Vis* (2018) 126: 771–96. doi:10.1007/s11263-018-1076-4

23. Wan X, Wang J, Kong Z, Zhao Q, Deng S. Multi-object tracking using online metric learning with long short-term memory. In: 2018 25th IEEE International Conference on Image Processing (ICIP); Washington, DC, USA. IEEE (2018). p. 788–92.

24. Mikami D, Otsuka K, Yamato J. Memory-based particle filter for face pose tracking robust under complex dynamics. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition; Washington, DC, USA. IEEE (2009). p. 999–1006.

25. Li P, Wang D, Wang L, Lu H. Deep visual tracking: Review and experimental comparison. *Pattern Recognition* (2018) 76:323–38. doi:10.1016/j.patcog.2017.11.007

26. Bolme DS, Beveridge JR, Draper BA, Lui YM. Visual object tracking using adaptive correlation filters. In: 2010 IEEE computer society conference on computer vision and pattern recognition. IEEE (2010). p. 2544–50.

27. Danelljan M, Shahbaz Khan F, Felsberg M, Van de Weijer J. Adaptive color attributes for real-time visual tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2014). p. 1090–7.

28. Danelljan M, Häger G, Khan FS, Felsberg M. Accurate scale estimation for robust visual tracking. In: British Machine Vision Conference; Berlin, Heidelberg. Springer (2014).

29. Mueller M, Smith N, Ghanem B. Context-aware correlation filter tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2017). p. 1396–404.

30. Ma C, Yang X, Zhang C, Yang MH. Long-term correlation tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2015). p. 5388–96.

31. Bertinetto L, Valmadre J, Golodetz S, Miksik O, Torr PH. Staple: Complementary learners for real-time tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2016). p. 1401–9.

32. Kim HI, Park RH. Residual lstm attention network for object tracking. *IEEE Signal Process Lett* (2018) 25:1029–33. doi:10.1109/lsp.2018.2835768

33. Yang T, Chan AB. Recurrent filter learning for visual tracking. In: Proceedings of the IEEE International Conference on Computer Vision Workshops (2017). p. 2010–9.

34. Yang T, Chan AB. Learning dynamic memory networks for object tracking. Proceedings of the European conference on computer vision (ECCV) (2018), 152–67.

35. Atkinson RC, Shiffrin RM. Human memory: A proposed system and its control processes. *Psychol Learn Motiv* (1968) 2:89–195. doi:10.1016/s0079-7421(08)60422-3

36. Shiffrin RM, Atkinson RC. Storage and retrieval processes in long-term memory. *Psychol Rev* (1969) 76:179–93. doi:10.1037/h0027277

37. Hong S, You T, Kwak S, Han B. Online tracking by learning discriminative saliency map with convolutional neural network. *Computer Sci* (2015) 597–606.

38. Hare S, Saffari A, Torr PHS. Struck: Structured output tracking with kernels. In: International Conference on Computer Vision; Washington, DC, USA. IEEE (2011).

39. Wei Z. *Robust object tracking via sparsity-based collaborative model*. Washington, DC, USA: Computer Vision & Pattern RecognitionIEEE (2012).

40. Kalal Z, Matas J, Mikolajczyk K. Pn learning: Bootstrapping binary classifiers by structural constraints. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; Washington, DC, USA. IEEE (2010). p. 49–56.

41. Kwon J, Lee KM. Visual tracking decomposition. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; Washington, DC, USA. IEEE (2010). p. 1269–76.

42. Kwon J, Lee KM. Tracking by sampling and integratingmultiple trackers. *IEEE Trans Pattern Anal Mach Intell* (2013) 36:1428–41. doi:10.1109/TPAMI.2013.213

43. Danelljan M, Hager G, Shahbaz Khan F, Felsberg M. Convolutional features for correlation filter based visual tracking. In: Proceedings of the IEEE International Conference on Computer Vision Workshops; Washington, DC, USA. IEEE (2015). p. 58–66.

44. Jia X, Lu H, Yang MH. Visual tracking via adaptive structural local sparse appearance model. In: IEEE Conference on computer vision and pattern recognition; Washington, DC, USA. IEEE (2012). p. 1822–9.

45. Liu B, Huang J, Yang L, Kulikowsk C. *Robust tracking using local sparse appearance model and k-selection*. Washington, DC, USA: CVPR 2011IEEE (2011). p. 1313–20.

46. Dong W, Lu H. Visual tracking via probability continuous outlier model. In: IEEE Conference on computer vision and pattern recognition; Washington, DC, USA. IEEE) (2014).