Visual tracking in complex scenes: A location fusion mechanism based on the combination of multiple visual cognition flows

Published Version:
Visual tracking in complex scenes: A location fusion mechanism based on the combination of multiple visual cognition flows / Liu, S; Huang, SC; Wang, S; Muhammad, K; Bellavista, P; Del Ser, J. - In: INFORMATION FUSION. - ISSN 1566-2535. - ELETTRONICO. - 96:(2023), pp. 281-296. [10.1016/j.inffus.2023.02.005]

Availability:
This version is available at: https://hdl.handle.net/11585/952079 since: 2024-01-04

Published:

DOI: http://doi.org/10.1016/j.inffus.2023.02.005

Visual Tracking in Complex Scenes: A Location Fusion Mechanism Based on the Combination of Multiple Visual Cognition Flows

Shuai Liu[1,2], Shichen Huang[1,2], Shuai Wang[1,2], *Khan Muhammad[3], Paolo Bellavista[4], *Javier Del Ser[5,6]

[1] Hunan Provincial Key Laboratory of Intelligent Computing and Language Information Processing, Changsha, 410000, China

[2] College of Information Science and Engineering, Hunan Normal University, Changsha, 410000, China

[3] Visual Analytics for Knowledge Laboratory (VIS2KNOW Lab), Department of Applied Artificial Intelligence, School of Convergence, College of Computing and Informatics, Sungkyunkwan University, Seoul 03063, Republic of Korea

[4] Department of Computer Science and Engineering, University of Bologna, Italy

[5] TECNALIA, Basque Research Technology Alliance (BRTA), 48160 Derio, Spain

[6] Department of Communications Engineering, University of the Basque Country (UPV/EHU), 48013 Bilbao, Spain

Corresponding Authors: Khan Muhammad (khan.muhammad@ieee.org) and Javier Del Ser (javier.delser@tecnalia.com)

Abstract

In recent years, deep learning has revolutionized computer vision and has been widely used for monitoring in diverse visual scenes. However, in terms of some aspects such as complexity and explainability, deep learning is not always preferable over traditional machine-learning methods. Traditional visual tracking approaches have shown certain advantages in terms of data collection efficiency, computing requirements, and power consumption and are generally easier to understand and explain than deep neural networks. At present, traditional feature-based techniques relying on correlation filtering (CF) have become common for understanding complex visual scenes. However, current CF algorithms use a single feature to describe the information of the target and locate it accordingly. They cannot fully express changeable target appearances in a complex scene, which can easily lead to inaccurate target locations in time-varying visual scenes. Moreover, owing to the complexity of surveillance scenes, monitoring algorithms can lose their target. The original template update strategy uses each frame with a fixed interval length as a new template, which may lead to unreliable feature extraction and low tracking accuracy. To overcome these issues, in this work, we introduce an original location fusion mechanism based on multiple visual cognition processing streams to achieve real-time and efficient visual monitoring in complex scenes. First, we propose a process for extracting multiple forms of visual cognitive information, and it is periodically used to extract multiple feature information flows of a target of interest. Subsequently, a cognitive information fusion process is employed to fuse the positioning results of different visual cognitive information flows to achieve high-quality visual monitoring and positioning. Finally, a novel feature template memory storage and retrieval strategy is adopted. When the location result is unreliable, the target is retrieved from memory to ensure robust and accurate tracking. In addition, we provide an extensive set of performance results showing that our proposed approach exhibits more robust performance at a lower computational cost compared with 36 state-of-the-art algorithms for visual tracking in complex scenes.

Keywords

Visual monitoring; Multiple visual cognition; Location fusion; Feature template memory; Complex scenes

1. Introduction

As an aspect of artificial intelligence, computer vision plays a crucial role in several applications, including biological information analysis, unmanned and autonomous driving, medical image processing, and industrial prognosis [1-6]. Computer vision can be defined as a domain that involves techniques for enabling machines to "see" the world, e.g., by using learning algorithms to imitate the functionality of the human visual cortex and obtain knowledge from image data. As such, computer vision algorithms can understand and recognize image content by simulating the cognitive criteria of human beings to analyze image information and draw valuable conclusions for a given task.

Visual monitoring is an important research topic in computer vision. Its main task is to periodically capture visual information of a specified target and then use the monitored information to predict the location of the target in each frame. The identification of the target through consecutive frames allows for the automatic real-time tracking of the target over time. In recent years, because of continual improvements and notable advancements, visual monitoring methods have been widely applied in security monitoring, smart city construction, and military guidance, among many other scenarios and applications [7-9]. However, in real-world motion scenes, visual monitoring is affected by many factors, such as changes in the shape of the target, fuzzy movement artifacts, occlusion, and illumination effects. Such contextual circumstances pose a considerable challenge to visual monitoring methods. Therefore, the development of methods for maintaining rapid and accurate monitoring of targets in complex scenarios has arguably become the main motivation for significant research investments in recent years.

At present, most visual monitoring methods either adopt a monitoring algorithm based on correlation filtering (CF) or exploit the excellent modeling capabilities of deep neural networks. Visual monitoring in complex scenes does not usually limit the type of target but only requires that the initially defined target object can be found in the video sequence. Existing deep-learning monitoring algorithms can extract rich target feature information from the flow of frames; however, they need to learn from massive datasets to distinguish the foreground from the background of video sequences. This limitation causes poor real-time monitoring quality (poor robustness) and high consumption of computing resources. In contrast, CF-based methods retain continuity between video frames, realize real-time updates of visual monitoring, and are easier to understand and explain. Consequently, they are often preferred over complex deep-learning-based alternatives. However, existing CF-based monitoring algorithms consider only a single feature (such as the histogram of oriented gradients (HOG)) to describe the visual information of the target. This limitation makes adapting to changes in the target state difficult in complex scenarios, resulting in inaccurate target positioning. In addition, CF-based methods usually update the template at fixed intervals. This template-updating method may fail to accurately re-identify the target when the tracking performance is poor, which has a negative effect on the associated results.

In this study, we address the aforementioned weaknesses of CF-based methods. Specifically, inspired by the multiple visual cognitive systems that humans use to monitor visual targets, we introduce a novel location fusion mechanism into our filtering monitoring algorithm. Moreover, considering the shortcomings of the original feature template update strategy, we propose a new feature template memory storage and retrieval strategy for the original algorithm. Our proposal aims not only to improve the accuracy of monitoring algorithms in complex scenes but also to promote the application of information fusion in visual monitoring. The contributions of this work are summarized as follows.

- Motivated by the inherent difficulty in performing visual monitoring in complex scenes, we propose a location fusion mechanism based on multiple visual cognitive processing streams. In our proposed approach, two key processes are implemented: i) *multiple visual cognitive information extraction* and ii) *cognitive information fusion*. In the former, the monitor is recycled to extract multiple complementary features from the target being monitored. In the latter, the quality of the extracted visual cognitive information is evaluated and the positioning results of different visual cognitive information flows are fused according to their quality to achieve high-quality visual monitoring and positioning. Thus, the target is recognized from a variety of cognitive perspectives. In different monitoring scenarios, various visual cognitive information flows are flexibly and effectively used to ensure high-quality target positioning. Even when dealing with highly dynamic scenarios, the monitoring accuracy of our proposed method can be steadily improved.

- The proposed approach involves a novel feature template memory storage and retrieval strategy based on monitoring confidence. The strategy is employed to calculate monitoring confidence based on visual cognitive information, which is used to evaluate the quality of feature templates so that only good feature templates are stored. This quality-driven storage policy permits the retrieval of the target from memory when the target is lost during monitoring, which may occur owing to complex artifacts and phenomena affecting the visibility of the object in the scene such as occlusion. Compared with other target-tracking methods, we provide a rationale for why our devised strategy improves the ability to find the target again after it is lost. To adapt to different monitoring scenarios, our mechanism flexibly adjusts the impact of different cognitive information flows on the monitoring process.
- The results of a comprehensive performance evaluation of our mechanism on benchmark datasets and a comparison with 36 existing advanced monitoring algorithms show that our method can effectively improve robustness in complex scenarios. The proposed approach also achieves significant improvements in terms of several evaluation metrics compared with the other algorithms on the benchmark, even under adverse contextual conditions.

The remainder of this paper is structured as follows. Section 2 introduces the relevant background for CF and visual cognition theory. Section 3 describes the proposed CF algorithm, which integrates multiple visual cognitive information flows to achieve efficient visual monitoring. Section 4 details the experimental setup. Section 5 presents the results along with a discussion from both quantitative and qualitative perspectives Section 6 provides our concluding remarks along with the directions for future research.

2. Related Work

In this section, we consider the concepts of vision cognition theory and recent developments in this direction (Section 2.1). In addition, we critically discuss the related state of the art, by considering their strengths and limitations (Section 2.2). This content is important to fully understand the core concepts and motivations behind the work presented in this manuscript.

2.1 Vision cognition theory

Visual cognition is a complex, intelligent, and efficient process that occurs in visual systems, encompassing two information-processing modes: bottom-up and top-down. Bottom-up is the feedforward process of visual cognition, which drives the visual processing in a system based on certain data or tasks. In contrast, the top-down process is the feedback of visual cognition; the visual system uses the previous knowledge and experience of the goal or target and the expectation and cognition of the future state of the goal or target. The human visual system is very powerful owing to its capacity for feedback regulation, association, memory, and other modes in the human visual cortex that effectively support decision-making in a synergic manner [10].

In recent years, researchers have applied visual cognition theories and mechanisms to several application scenarios, particularly in intelligent manufacturing and artificial intelligence, and positive results have been reported. For example, Hong et al. [20] proposed a two-component approach consisting of short-term and long-term memory storage strategies in addition to principles based on cognitive psychology to process the appearance memory of the target. Cai et al. [21] addressed the challenges of scale changes and non-rigid body deformation by analyzing the visual cognitive mechanisms of the ventral flow of the visual cortex. This mechanism simulates shallow neurons to extract low-level biologically inspired features of the appearance of a target in combination with advanced learning mechanisms to perform visual monitoring. Srivastava et al. [22] proposed a multi-object monitoring computational model designed to experimentally predict the allocation of visual attention and the effect of this allocation on an observer's ability to monitor multiple targets simultaneously. Inspired by the cognitive salience model of human attention, Zhan et al. [23] proposed a visual monitoring scheme based on significant superpixels by integrating the similarity of target appearance and cognitive salience, which plays a major role in updating the appearance model and inferring the target location.

2.2 Visual monitoring algorithm

Mainstream visual monitoring algorithms can be classified into deep-learning methods and CF algorithms. Deep-learning-based monitoring algorithms generally require large amounts of training data to learn their trainable parameters and extract meaningful features for effective learning over frame sequences with high dimensionality. Zhao et al. [43] proposed a deep-learning-based intelligent edge surveillance technique for a specific intelligent Internet of Things application. In deep-learning-based human activity recognition (HAR) performance and cost must be balanced. To this end, Shi et al. [67] proposed a smartphone-aided HAR method using a residual multi-layer perceptron. To solve the problem of online-only methods, which inherently limit the richness of the features that can be learned by a model, Bertinetto et al. [50] proposed a novel fully convolutional Siamese network trained to perform end-to-end target monitoring in video streams. Li et al. [16] introduced a Siamese region proposal network that extracted target feature information from an initial frame using a Siamese sub-network and fed it to a convolutional layer to match the extracted target feature information from other frames. Finally, a candidate region generation network was used to distinguish the target from the background for visual monitoring. Galoogahi et al. [12] proposed a manual feature background perceptron method to solve the problem that the target background may not be modeled over time.

Visual monitoring algorithms based on kernel CF convert calculations in the time domain to the frequency domain. This method utilizes the property that a cyclic matrix can be diagonalized in the frequency domain, which significantly reduces the amount of calculation and improves its speed. For example, Henriques et al. [11] optimized a method based on CF by introducing circulant matrices, which increased the number of negative samples through the circulant matrix and improved the quality of classifier training, and by adding a Gaussian kernel to the ridge regression. Nonlinear problems were converted into a high-dimensional linear space to simplify the calculations.

Furthermore, Yang et al. [13] proposed a new tracking algorithm based on CF designed to perform a novel robust estimation of similar transformations with large displacements. This method uses an efficient phase correlation scheme to simultaneously address scale and rotation changes in logarithmic polar coordinates. Although CF has high accuracy in visual monitoring, it is unfortunately impacted by the so-called *boundary effect,* in which the original sample target center circularly shifts to the edge and tracking failure is likely to occur. Danelljan et al. [54] solved this problem to some extent by imposing spatial penalties on the coefficients of correlation filters, but their workaround increased the complexity of monitoring. As detailed subsequently, our proposed memory update mechanism effectively mitigates the boundary effect by selectively updating the template and minimizing the complexity to achieve better performance.

To solve the problem of online updating, Li et al. [27] introduced time regularization in the matching process between adjacent frames to increase the number of training samples and provide a more robust model when the appearance of scenery changes significantly. Considering the problems that most monitoring algorithms encounter when scaling the search to estimate the target size (that is, the increased computational resources needed to deal with large-scale changes), Danelljan et al. [28] proposed a new scale-adaptive monitoring method designed to learn the discriminant CF used for translation and scale estimation to reduce the computational cost.

Compared with these deep learning algorithms, our proposed method does not require a large number of training samples. For a given level of monitoring accuracy, the computing demand and power consumption of our approach are significantly reduced. Compared with other tracking algorithms based on CF, our proposed solution attains higher tracking accuracy levels and can adapt better to a large set of challenging difficulties. For example, by using multiple features, our approach exhibits improved accuracy compared with the method of using a single feature provided by Galoogahi et al. [12] under conditions of occlusion, rotation, and changes in scale. Similarly, our feature template memory storage and retrieval strategy can be used to selectively update the target template, which effectively reduces the influence of boundary effects compared with a prior method [13]. A complete description of our proposed solutions, its motivations, and its advantages is provided in the following section.

3. Proposed Multiple Visual Cognition Methodology

This section consists of four subsections. Section 3.1 introduces the baseline CF algorithm. Section 3.2 presents the key defects in existing filter monitoring methods that use only a single feature to describe target information. Section 3.3 describes the two key processes of the approach: "multiple visual cognitive information extraction" and "cognitive information fusion"; it also describes the template memory strategy based on monitoring confidence. Finally, Section 3.4 explains how the filtering monitoring algorithm is combined with the location-fusion mechanism for effective visual monitoring in complex scenes.

3.1 CF monitoring algorithm

CF was first applied in the field of signal processing to describe the similarity between two signals [24]. The correlation between two signals *x* and *y* can be expressed as

$$(x \otimes y)(\alpha) = \int_{\infty}^{-\infty} f^*(t)x(t+\alpha)dt, \tag{1}$$

$$(x \otimes y)(n) = \sum_{-\infty}^{\infty} f^*\,[n]y(\beta+n), \tag{2}$$

where f^* represents the complex conjugate of f. In the context of a monitoring problem, CF can be framed by formulating the problem as the discovery of the maximum value of correlation s between a filter template, ρ, and an input image, ϖ. Mathematically,

$$\mathrm{s} = \rho \otimes \varpi. \tag{3}$$

Quantitatively performed through convolution, the Fourier transform of the cross-correlation of functions is equal to the product of the Fourier transform of the functions:

$$\mathrm{F(s)} = \mathrm{F}(\rho \otimes \varpi\,) = \mathrm{F}(\varpi\,) \cdot F(\rho)^*, \tag{4}$$

where $F(\rho)^*$is the filtering template.

Bolme et al. [25] proposed a new type of correlation filter (referred to as *minimum output sum of square error filter*) that generates a stable correlation filter during the initialization of a single frame. This correlation filter is used to calculate the least squares of each sample and can be described as

$$min_{F(\rho)^*} = \sum_{i=1}^{m} |F(\rho)^*\ \mathrm{F}(\varpi\,)_i - \mathrm{F(s)}_i|^2, \tag{5}$$

where *i* is the index of the sample for which the Fourier transform is computed. Then, the partial with regard to $F(\rho)^*$ is set equal to zero, and $F(\rho)^*$ is considered as an independent variable. The partial derivative is calculated as per Equation (5), and we obtain the closed-form expression for the new type of filter:

$$F(\rho)^* = \frac{\sum_i \mathrm{F}(\varpi\,)_i \cdot F(s)_i^*}{\sum_i \mathrm{F(s)}_i \cdot F(s)_i^*}. \tag{6}$$

3.2 Disadvantages of existing filtering monitoring methods

Existing filtering and monitoring methods extract relevant information features (mainly HOG features) of the target in the searched area of the next frame using a sliding window method. The features are then matched with the target information template of the current frame to predict the location of the next frame. For example, the target information features extracted by using the background-aware correlation filter (BACF) algorithm are as follows. First, the monitor extracts features from image region Q^i of the *i*th$(i = 1,2,\cdots,M)$ sliding window as

$$P_x(x,y) = Q^i(x+1,y) - Q^i(x,y), \tag{7}$$

$$P_y(x,y) = Q^i(x,y+1) - Q^i(x,y), \tag{8}$$

$$N(x,y) = \sqrt{P_x(x,y)^2 + P_y(x,y)^2}\,, \tag{9}$$

$$\theta(x,y) = tan^{-1}\frac{P_y(x,y)}{P_x(x,y)}, \tag{10}$$

where $Q^i(x,y)$ represents the value of the image pixel at position (x,y), P_x and P_y respectively represent the gradient values in the horizontal and vertical directions of image P, $N(x,y)$ represents the amplitude value of the gradient, and $\theta(x,y)$ denotes the direction of the gradient. The construction process of the orientation gradient histogram in an image region, Q^i, is shown in Figure 1.

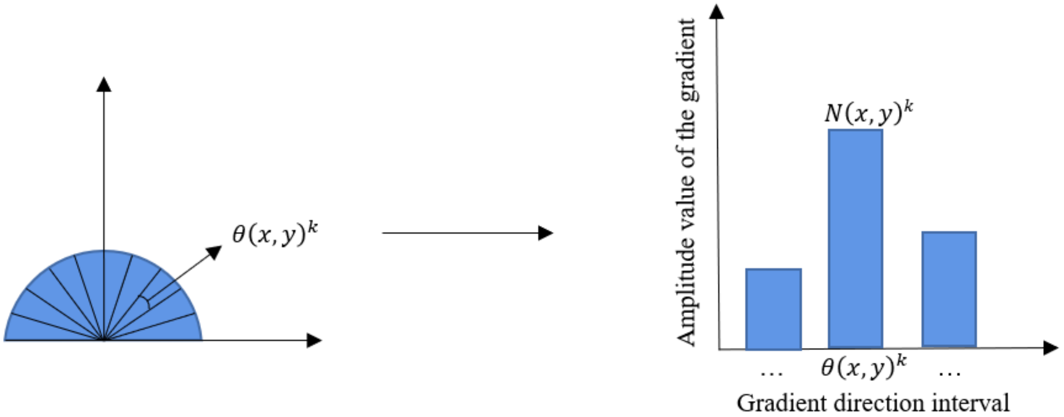


Fig. 1 Process of constructing a histogram of oriented gradients in an image region, Q^i. The left side of the figure shows that the gradient direction is divided into n intervals, and the right side shows that each interval has a corresponding gradient value.

First, we divide the gradient direction, $\theta(x,y)$, into n intervals in the image region, Q^i, so that the amplitude values of similar gradient directions are placed in the same interval, and use $A_k(k = 1,2,\cdots,n)$ to represent the k^{th} bin of the histogram. More formally,

$$A_k = \{N(x,y)^k | \theta(x,y)^k\}, \tag{11}$$

where $\theta(x,y)^k$represents the k^{th} gradient direction interval and $N(x,y)^k$ represents the amplitude value in the k^{th} gradient direction interval. n intervals are then assembled to construct the orientation gradient histogram, H^i, of the current image region, Q_i:

$$H^i = \{A_1, A_2, \cdots, A_n\}. \tag{12}$$

Finally, the M sliding window images are merged to form the HOG feature information of the images in the current search area.

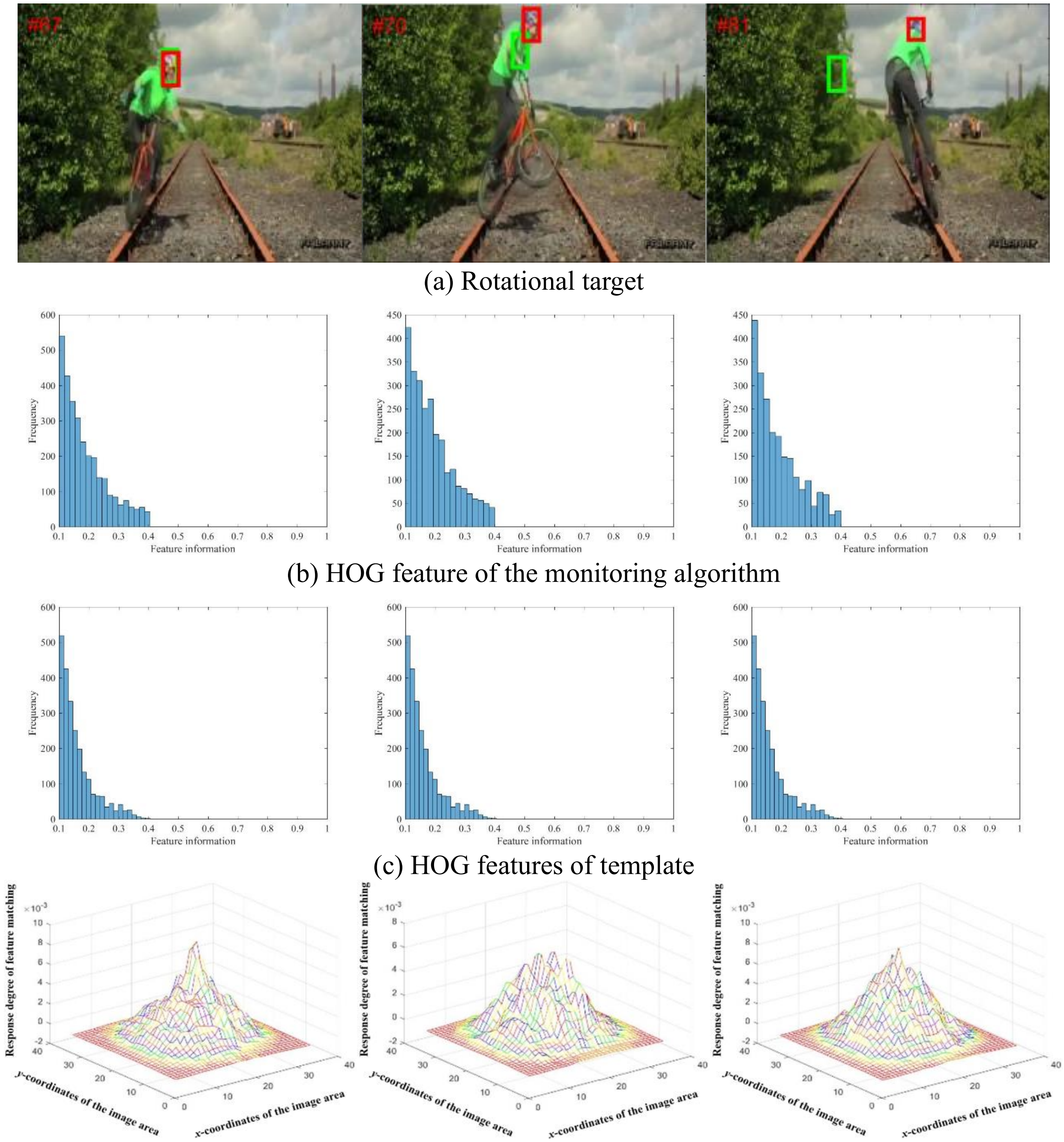
$$FT^h = \{H^1, H^2, \cdots, H^M\}. \tag{13}$$

The occurrence probability, R_h, of the HOG feature information, h, of the target in each sliding window is given as

$$R_h = f(\frac{Tot^h}{FT^h}), \tag{14}$$

where Tot^h represents the HOG feature information template of the currently used target, and f represents the matching function.

Note that when the target rotates, the feature information extracted by the monitor is an HOG feature after the rotation occurs, which exhibits poor matching performance with the current target information template, as shown in Figure 2.



(a) Rotational target

(b) HOG feature of the monitoring algorithm

(c) HOG features of template

(d) Matching response map

Fig. 2 HOG feature map and feature matching response map are extracted from a "`bike`" sequence. When the target rotates, the single feature matching response is not good, which may cause target loss for a tracking algorithm relying on such features.

Figure 2 illustrates the effect of a rotating target on the performance of the filtering-based tracking approach. Specifically, Figure 2a shows the effects of rotation on the output of the monitoring algorithm. Figure 2b presents the feature diagram extracted by the monitoring algorithm in the monitoring search area. Figure 2c displays the feature diagram extracted in the real search area, and Figure 2d presents the response matrix diagram of the feature-matching degree of the monitoring algorithm. As shown in Figure 2a, the target moves normally at frame 67, and the target feature information extracted by the monitoring algorithm can be matched with the HOG feature template information of the target of the previous frame, as shown in Figure 2d. Rotation occurs when the target moves to frame 70. At this time, the target feature information extracted by the monitor (as shown in Figure 2b) exhibits a poor matching effect with the target template information (as depicted in Figure 2d). Consequently, a large deviation exists in the monitoring of the target, resulting in a subsequent monitoring failure at frame 81. The response graph of the features extracted by the monitor and the real location features of the target at frame 81 clearly reveal that the former exhibits a large error at this time in the sequence.

Based on these observations and issues, an important conclusion can be drawn. In the existing filtering monitoring algorithms that employ only single HOG features for visual monitoring, when the target is affected by the external environment or the occurrence of target rotation and/or deformation, the target feature information extracted by the monitor cannot match the target template information well. This causes target monitoring to fail, and performing effective visual monitoring in subsequent frames of the stream also becomes difficult.

3.3 Visual monitoring strategy based on location fusion mechanism

The proposed visual monitoring strategy is implemented in three stages: multiple visual cognitive information extraction, cognitive information fusion, and feature template memory storage and retrieval strategy.

(1) *Multiple visual cognitive information extraction*: The multiple visual cognition process aims to recognize external objects/targets through multiple dimensions or angles. To complete the multiple visual cognitive information extraction process for the object/target, it is necessary to know roughly what the object/target looks like and what features it has. Therefore, multiple visual cognition processes can be regarded as continuous learning of the appearance, form, and action of external objects/targets under diverse visual perspectives.

(2) *Cognitive information fusion*: Information fusion is an efficient way to process and combine information flows such that synergies between them towards fulfilling a task are properly leveraged. First, from the object/target-related feature information obtained in the process of multiple-vision cognitive information extraction, feature information from different cognitive angles can be collected to obtain different location information flows. Then, the quality of such different location information flows can be evaluated to support quality-aware fusion and to detect and locate the current object of interest with improved performance, thereby reducing the interference of low-quality location information on the final results.

(3) *Feature template memory storage and retrieval strategy*: Template memory is an effective means to correct poor monitoring situations. The memory of some target features can be realized through a single cognition event, but the memory of most feature contents calls for repeated cognition and feedback processes (*long-term cognitive memory*). Thus, it is more suitable for long-term monitoring in complex scenarios.

We now consider the mathematical details of the three aforementioned stages.

Multiple visual cognitive information extraction: For an object/target to be monitored, the monitor recognizes the current state and form of the object/target. To model the object more comprehensively, we chose four different cognitive perspectives that can complement each other: HOG, color, gray levels, and saliency features.

1) *Color features*: The monitor extracts color features from the image area, Q^{i}, of the i^{th} $(\mathrm{i}=1,2,\cdots,\mathrm{M})$ sliding window. It first divides the RGB of image area Q^{i} into three HSV components and quantizes them according to the sensitivity of the color change. After quantization, the value ranges of the three components

are $\{0,1,\cdots,K_H-1\}$, $\{0,1,\cdots,K_S-1\}$,and $\{0,1,\cdots,K_V-1\}$, arranged into a vector in the form of [H, S, V], ranging as follows:

$$Q^i_{HSV}=\{0,1,\cdots,K_H-1,0,1,\cdots,K_S-1,0,1,\cdots,K_V-1\}. \qquad (15).$$

The number of pixels of color component c in the sliding window image area, Q^i, is defined as I^i_c; the total number of pixels of the color component is N, and the color histogram, CO^i, of the current image region, Q^i, is then constructed. More formally,

$$CO^i=\{I^i_1,I^i_2,\cdots,I^i_N\}. \qquad (16)$$

The M sliding window images are then merged to form the color feature information of the current search area image:

$$FT^k=\{CO^1,CO^2,\cdots,CO^M\}. \qquad (17)$$

However, the operation of visual monitoring in an actual complex scene is challenging; for example, the area explored by the monitor often has one or more similar background types interfering with the tracked objects/targets, which may greatly affect the accuracy of the monitoring process. At this point, using only shape and color to describe the target would significantly affect the accuracy of the monitoring process. Therefore, we propose taking inspiration from the learning process of the human visual system [10] to learn the different characteristics of targets from more perspectives. Gray and saliency feature information are added to the algorithm to identify and track the target more accurately in the case of occlusion and rapid movement of the target.

2) *Gray features*: First, we convert image area Q^i of the *ith* $(i=1,2,\cdots,M)$ sliding window of the specific color image into a gray value to obtain the gray feature, GR^i, of the current window area:

$$GR^i(x,y)=\lambda_1 Q^i_R(x,y)+\lambda_2 Q^i_G(x,y)+\lambda_3 Q^i_B(x,y), \qquad (18)$$

where λ_1,λ_2, and λ_3 are the coefficients set by the three channels, and $Q^i_R(x,y), Q^i_G(x,y),$ and $Q^i_B(x,y)$ represent the pixel values of the current i^{th} sliding window area. Then, the M sliding window patches are merged to form the *gray feature* of the current search area image using

$$FT^s=\{GR^1,GR^2,\cdots,GR^M\}. \qquad (19)$$

3) *Saliency features*: This is produced by the visual contrast between a visual target and other areas. This can be described by using certain features. The saliency features (SF^i) of the sliding window image area Q^i are given as

$$SF^i=G^x\circledast\mathcal{F}^{-1}\left[\exp\left(\log\left(A\left(\mathcal{F}(Q^i)\right)\right)-H_n\circledast\log\left(A\left(\mathcal{F}(Q^i)\right)\right)+\mathrm{P}\left(\mathcal{F}(Q^i)\right)\right)\right]^2, \qquad (20)$$

where $\mathcal{F}$ represents the Fourier transform function, $A(*)$ and $P(*)$ represent the amplitude and phase of their argument Fourier transform, G^xrepresents a Gaussian filter smoothing saliency mapping, and $\circledast$ represents the convolutional operator. Here, H_n represents the coefficient matrix, computed as

$$H_n=\frac{1}{n^2}\begin{bmatrix}1&\cdots&1\\\vdots&\ddots&\vdots\\1&\cdots&1\end{bmatrix}. \qquad (21)$$

M sliding window images are combined to form salient features FT^f in the current search area image as

$$FT^f=\{SF^1,SF^2,\cdots,SF^M\}. \qquad (22)$$

This analysis indicates that resorting to a single feature is insufficient for visual monitoring. Therefore, mimicking the human multi-visual cognitive mechanism is essential for multiple feature extraction and integration to circumvent the problem of monitoring failures caused by insufficient target information mining. Inspired by human multiple visual cognition, we propose a visual monitoring strategy that performs cognitive information fusion, feature template memory storage, and a retrieval strategy after extracting multiple features.

Cognitive information fusion: Multiple feature information (HOG, color, gray, and saliency features) extracted in the previous stage may not match the feature information corresponding to the target in the previous frame well. In this case, obtaining accurate positioning information using only individual feature information may not be possible, as shown in Figure 2d. Therefore, the positioning information results obtained from these multiple features must be cognitively integrated. The main steps of this fusion process are detailed here.

Step 1: The matching response matrix is calculated from the color, gray, and saliency feature information as

$$R_k = f(\frac{Tot^k}{FT^k}), \tag{23}$$

$$R_s = f(\frac{Tot^s}{FT^s}), \tag{24}$$

$$R_f = f(\frac{Tot^f}{FT^f}), \tag{25}$$

where Tot^k, Tot^s, and Tot^f are respectively the color, gray, and saliency feature information templates of the current target.

Step 2: The quality levels of HOG, color, gray, and salient feature information are calculated. The quality level of the HOG feature information is expressed as

$$\rho_h = \frac{max(R_h) - \mathrm{mean}(R_h)}{std(R_h)} \ , \tag{26}$$

where $\max(*)$, $mean(*)$, and $std(*)$ represent the maximum, mean, and standard deviation, respectively. Similar expressions can be formulated for quality level ρ_k of the color feature information, quality level ρ_s of the gray feature information, and quality level ρ_f of the salient feature information.

Step 3: Finally, the response matrix obtained by adaptive integration and linear weighting is given as

$$R_F = \rho_h \cdot R_h + \rho_k \cdot R_k + \rho_s \cdot R_s + \rho_f \cdot R_f, \tag{27}$$

$$PL = \mathrm{argmax}_{x,y}(R_F), \tag{28}$$

where R_F represents the response matrix after cognitive information fusion and PL denotes the location of the current target. This process is schematically shown in Figure 3, where ρ_h, ρ_k, ρ_s, and ρ_f respectively represent the weights of the HOG, color, gray, and saliency feature information.

Feature template memory storage and retrieval strategy: After the response matrix is obtained by matching the HOG, color, gray, and salient feature information, the current cognitive features must be memorized, stored, and retrieved as needed. The monitoring confidence of the HOG feature information is expressed as

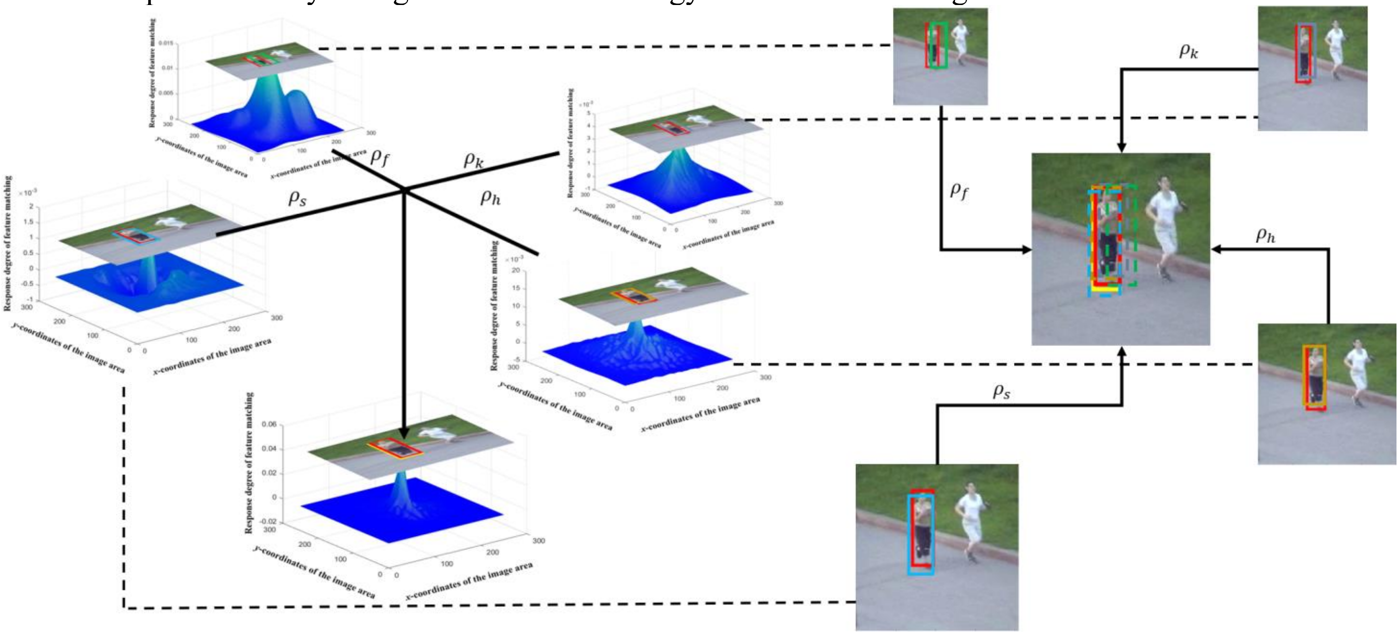
$$CF_h = \frac{|f_{max}(R_h) - f_{min}(R_h)|^2}{f_{mean}\left(\Sigma_{i,j}\left|f_{max}(R_h) - R_{h_{i,j}}\right|\right)} \ , \tag{29}$$

where $f_{min}, f_{max}, f_{mean}$represents the minimum, maximum, and mean values of the function, respectively; i, j represent the i^{th} row and j^{th} column elements of R_h; and CF_h stands for the monitoring confidence of the HOG feature information of the target. The monitoring confidence of the color feature information of the target (CF_k), that of the gray feature information (CF_s), and that of the salient feature information (CF_f) follow analogously.

Then, the feature template is stored and updated by monitoring the confidence of the feature information. For example, the update method of the HOG feature template retrieval and storage of the moving target in the current frame i is given by

$$Tot_i^h = \begin{cases} Tot_{i-1}^h, CF_h < \xi_h \\ Tot_{i-1}^h + Tot_i^{h'}, CF_h \geq \xi_h \end{cases}, \tag{30}$$

where ξ_h represents a fixed threshold (typically 0.32), and $Tot_i^{h'}$ represents the HOG information feature of the newly extracted target in the current frame i. The update method for the retrieval and storage of the target color, gray, and salient feature information is the same as that for the HOG feature information. The feature template memory storage and retrieval strategy are illustrated in Figure 4.



ρ_f
ρ_k
ρ_s
ρ_h
ρ_k
ρ_f
ρ_h
ρ_s
Response degree of feature matching
y-coordinates of the image area
x-coordinates of the image area

Fig. 3 Cognitive information fusion process. The left side of the figure depicts the quality-based fusion process of positioning results obtained from different cognitive perspectives (features), and the right side intuitively shows the location fusion process from the monitoring scene, where the image size represents the corresponding visual cognitive weight at this instant of time along the video stream. The three-dimensional coordinates of the matching response map comprise the *x*- and *y*-coordinates of the image area and the response degree of feature matching.

The specific process of the feature template memory storage and retrieval strategy is performed as follows. First, match the current frame with the four kinds of feature information extracted from the target position of the previous frame. When the matching quality is poor, we rematch the current and template frames. When the current frame matches the previous frame appropriately, the template frame is updated to the previous frame. In the monitoring scene, as shown in Figure 4, the matching response of gray features is good, so the template is updated to the previous frame. However, the matching of HOG, color, and saliency feature templates is poor, so no update is needed, and the template of the previous frame is used to match again.

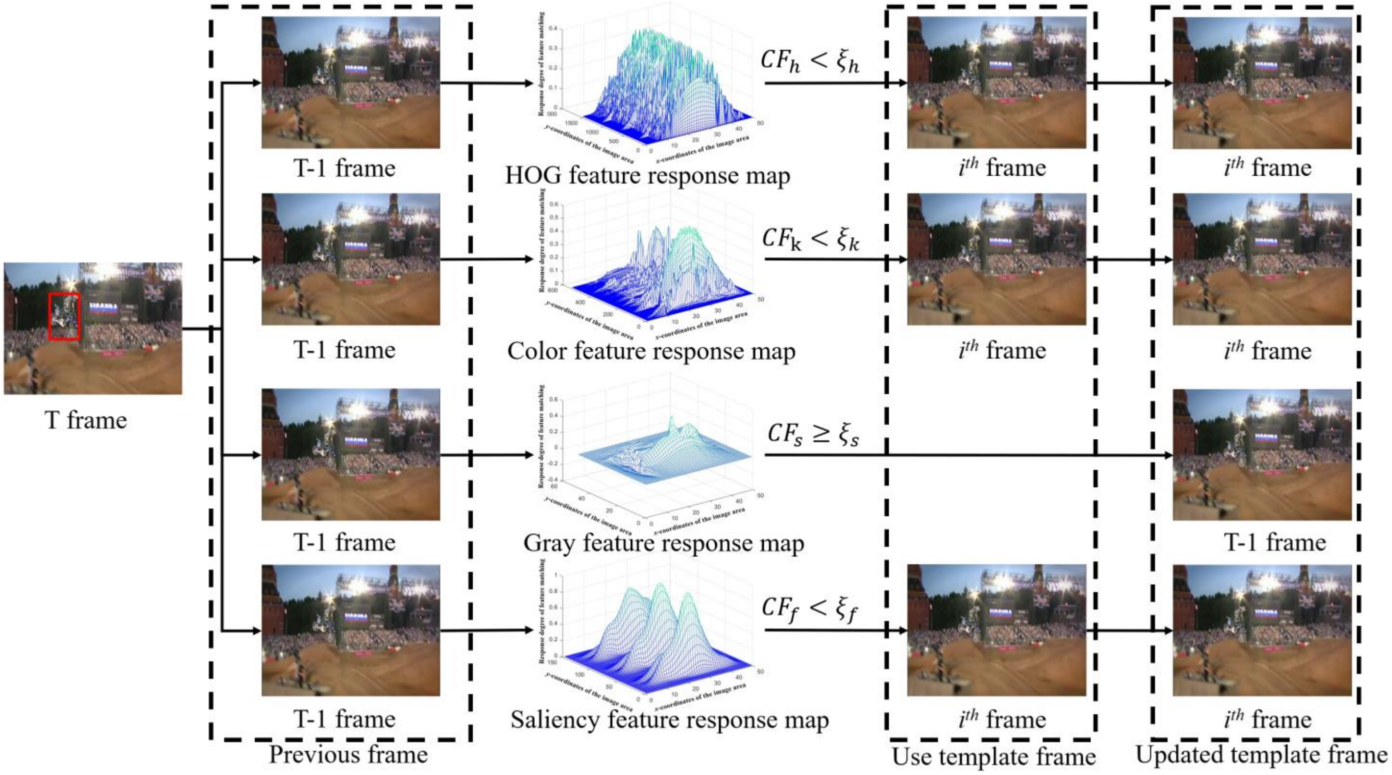


Fig. 4 Feature template memory storage and retrieval strategy. Multiple features of the target are extracted and compared with the template features, and then, whether to update the template features is decided. The three-dimensional coordinates of the matching response map include the *x*- and *y*-coordinates of the image area and the response degree of feature matching.

3.4 Filtering monitoring algorithm integrating multiple visual cognition

First, the method extracts multiple feature information flows of the targets to be tracked from the search area. Subsequently, the quality of these multiple feature information flows is used to adaptively fuse the positioning results of each feature information to achieve better visual positioning. Finally, the storage and retrieval strategy of the feature template memory is used to prevent the loss of targets during the monitoring process under visual occlusion, blockage, changes in illumination, and other artifacts that may affect the tracking process. A flow chart of the location fusion mechanism based on multiple visual cognition processes is shown in Figure 5.

According to this description, our location fusion mechanism solutions were applied to the CF monitoring algorithm, as described in Algorithm 1.

Algorithm 1: Proposed CF-assisted monitoring algorithm based on location fusion mechanism

Input: Initial target position (X_1, Y_1); set thresholds $\xi_h, \xi_f, \xi_k, \xi_s$.

Output: Prediction position (X_i, Y_i) and monitoring box of subsequent frame monitoring algorithm.

```
Repeat
  Detect:
  // Step 1: Multiple visual cognitive information extraction
      Extract HOG, color, gray, and salient feature information by formula (13), (17), (19) and (22),
      corresponding to $FT^h$, $FT^k$, $FT^s$, and $FT^f$ from the search area of the target location
      $(X_{i-1}, Y_{i-1})$in the previous frame.
```

```
        Match with the corresponding target feature template in the previous frame by formula (14), (23), (24) and (25).
        Form HOG, color, gray, and salient matching response matrix corresponding to $R_h$, $R_k$, $R_s$, and $R_f$.
    // Step 2: Cognitive information fusion
        Obtain the quality of each feature information by formula (26), which are $\rho_h$, $\rho_k$, $\rho_s$ and $\rho_f$.
        Predict the monitoring position $(X_i, Y_i)$ and monitoring box by formulas (27) and (28) in an adaptive linear weighting manner.
    // Step 3: Feature template memory storage and retrieval strategy
        Calculate the monitoring confidence of each feature information by formula (29), which are $CF_h$, $CF_k$, $CF_f$ and $CF_s$, respectively.
        If $CF_{x=h,k,s,f} \geq \xi_{x=h,k,s,f}$ then
            Update the target template of corresponding characteristic information.
Until the end of the video sequence；
```

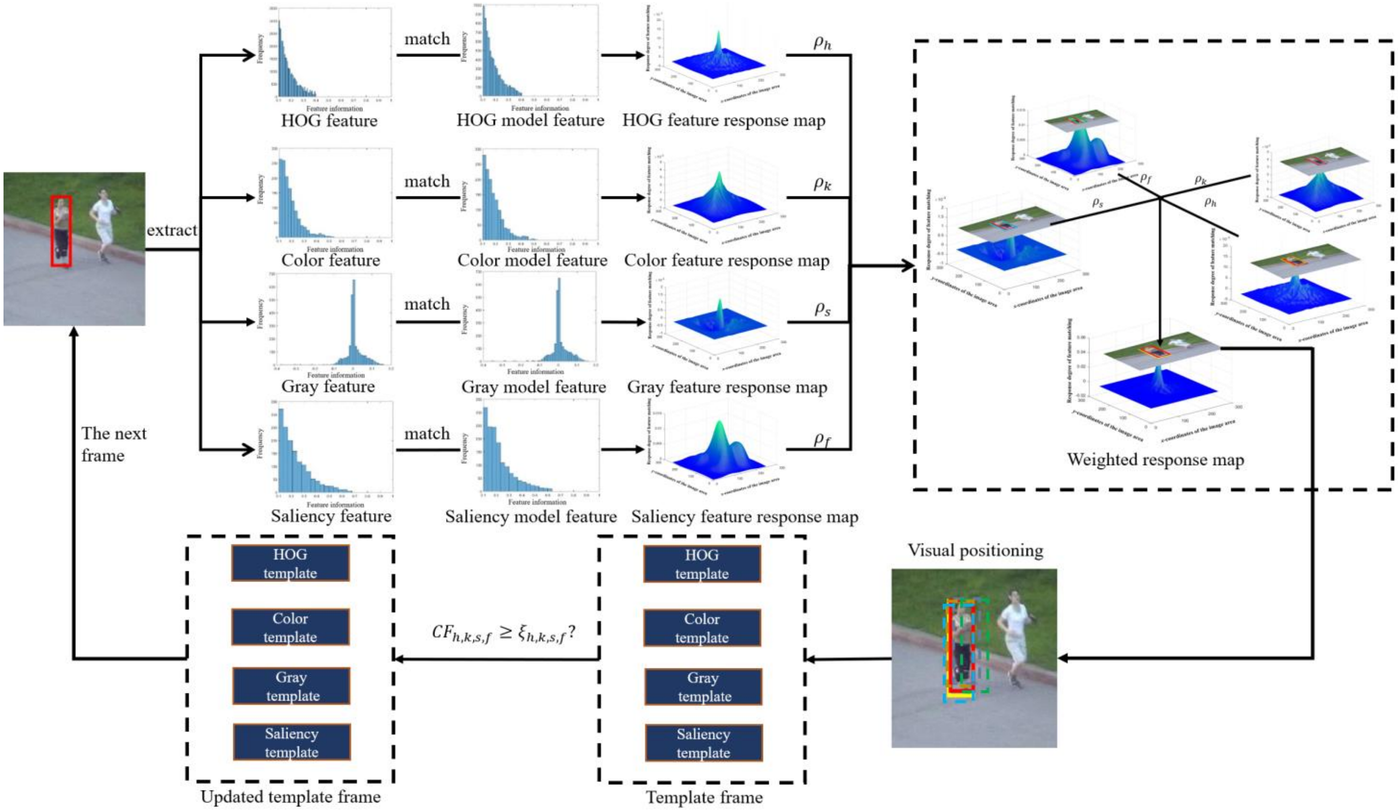


Fig. 5 Flow chart of location fusion mechanism based on multiple visual cognition. Different matching results with multi features on template are shown, and the reliability with each feature is obtained via the response map. Then, visual positioning is obtained by applying location fusion based on information quality. Finally, we confirm whether to use the feature template memory storage and retrieval strategy according to the monitoring confidence. The abscissa and ordinate of the histogram indicate feature information and frequency, respectively. The three-dimensional coordinates of the matching response map comprise the *x* and *y*-coordinates of the image area and the response degree of feature matching.

4. Experimental Setup

To assess the performance of our proposed approach, we designed an extensive experimental setup over three benchmark datasets that are widely utilized in visual monitoring: OTB100 [17], VOT2016 [18], and VOT2018 [19]. Here, we describe the challenges and performance metrics measured for each dataset.

OTB100 dataset:

There are 11 types of challenges defined in the OTB100 dataset, including fast motion, occlusion, out-of-view, deformation, motion blur, and out-of-plane rotation. There were 98 videos in this dataset, with a total of 100 test scenarios, which contained a quarter of the grayscale sequence. The evaluation indicators for the OTB100 dataset are the precision and success rate; these are computed in the context of object detection and tracking as follows.

Precision: To compute the tracking accuracy, we first calculate distance θ between the center position (X_t, Y_t) of the predicted target in each frame of the monitoring algorithm and the center position (X_i, Y_i) of the ground truth annotation of the target's position:

$$\theta = \sqrt{(X_t - X_i)^2 + (Y_t - Y_i)^2}. \tag{31}$$

We then calculate ratio Υ of the number of frames $V,$ whose distance θ is less than a certain threshold for all frames of the OTB dataset. The value of Υ reflects the search accuracy.

$$\Upsilon = \frac{\sum_{i=1}^{V} F_i}{V}, F_i = \begin{cases} 1, \theta \leq 20 \\ 0, \theta > 20 \end{cases}. \tag{32}$$

For the OTB100 dataset, a threshold score of 20 pixels was used as the representative accuracy score for each tracker.

Success rate: The computation of this second score starts by first calculating ratio ∂_i of the intersection over union measure between the target box, ω_i', of the current *i*th frame predicted by the monitoring algorithm and the true labeled target box, ω_i, namely,

$$\partial_i = \frac{|\omega_i' \cap \omega_i|}{|\omega_i' \cup \omega_i|}. \tag{33}$$

Then, we determine percentage sr of the number of frames V with ∂_i lower than a certain threshold, ϱ, to all the frames of the OTB dataset (sr is the success rate).

$$sr = \frac{\sum_{i=1}^{V} \varsigma_i}{V}, \varsigma_i = \begin{cases} 1, \partial_i > \varrho \\ 0, \partial_i \leq \varrho \end{cases}. \tag{34}$$

Using one success rate value at a specific threshold (e.g., 0.5) for tracker evaluation may not be fair. Instead, we used the area under the curve (AUC) of each successful plot to rank the tracking algorithms for comparison.

VOT2016 and VOT 2018 dataset:

The VOT2016 dataset includes six challenges: jitter blur, illumination change, scale change, and motion change. The VOT2018 dataset was updated based on the VOT2016 dataset and comprised 60 video sequences. The evaluation metrics for such datasets are the expected average overlap (EAO), accuracy, and robustness, which are defined in detail as follows.

EAO Ac_{av}): All sequences are classified according to their length, and the monitoring algorithm is tested on a sequence with a length range of $[K_h, K_l]$, to obtain the accuracy of each frame, Ac_i. They are then added together, and the average value is the average accuracy, Ac_{av}, expressed as follows:

$$Ac_{av} = \frac{1}{K_l - K_h} \sum_{i=K_h}^{K_l} Ac_i. \tag{35}$$

Accuracy (Ac_{vd}): First, we calculate the average accuracy when the *m*th monitor repeatedly runs the *t*th frame *k* times and then calculate the accuracy of each frame. Finally, the valid frames are averaged, yielding

$$Ac_{vd} = \frac{1}{N_{vd}} \frac{1}{N_r} \sum_{t=1}^{N_{vd}} \sum_{k=1}^{N_r} \phi_t(m, k), \tag{36}$$

where N_r is the number of repetitions, $\phi_t(m, k)$ is the accuracy of the *t*th frame ($t \in [1, N_{vd}]$) when the *k*th repeat sequence is executed by the m^{th} monitor, N_{vd} is the number of valid frames, and Ac_{vd} is the effective average accuracy.

Robustness (Ro_{av}): The computation of this score requires first summing the failure times of the m^{th} monitor running the k^{th} repeat sequence and then taking the average as

$$Ro_{av} = \frac{1}{N_{fa}} \sum_{k=1}^{N_{fa}} \phi_r(m,k), \tag{37}$$

where N_{fa} is the number of repetitions, $\phi_r(m,k)$ denotes the number of tracking failures, and Ro_{av} is a measure that indicates the robustness of the algorithm over time (e.g., a lower value of Ro_{av} indicates that the algorithm performs more stably over time).

5. Results and Discussion

This section is structured into three sections. In the first part, we describe a quantitative evaluation of the proposed tracking method in comparison with state-of-the-art algorithms in terms of multiple evaluation metrics over the three datasets (OTB100, VOT2016, and VOT2018) described previously. Considering the importance of quality-oriented evaluation in this field, in Section 5.2, we report and discuss the qualitative aspects of our tracking method against other counterparts included in the benchmark. Finally, possible limitations in the design and performance of our proposed solution are discussed in Section 5.3, to highlight important future research directions that align with this work.

5.1 Quantitative analysis

We begin by evaluating, from different perspectives, the performance of the proposed tracking method (referred to as BACF_HMC hereinafter) against 36 advanced tracking algorithms on the OTB100, VOT2016, and VOT2018 datasets.

A. OTB100

In this section, we describe an objective evaluation and analysis of our BACF_HMC algorithm and the latest existing state-of-the-art monitoring algorithms (SiamFC+ [30], ECO_HC [31], MenTrack [32], RFL [33], OMFL [34], LMCF [35], AutoTrack [36], TB_BiCF [37], DCF_CA [38], ACFN [39], Staple [40], LUDT+ [41], StructSiam [42], RT-MDNet [44], Siam-tri [45], SACF [46], HP [47], UDT [48], CFNet [49], SiamFC [50], KCF [11], LDES [13], and BACF [12]) on the OTB100 dataset. The results of the overall objective analysis are presented in Figure 6 and Table 1.

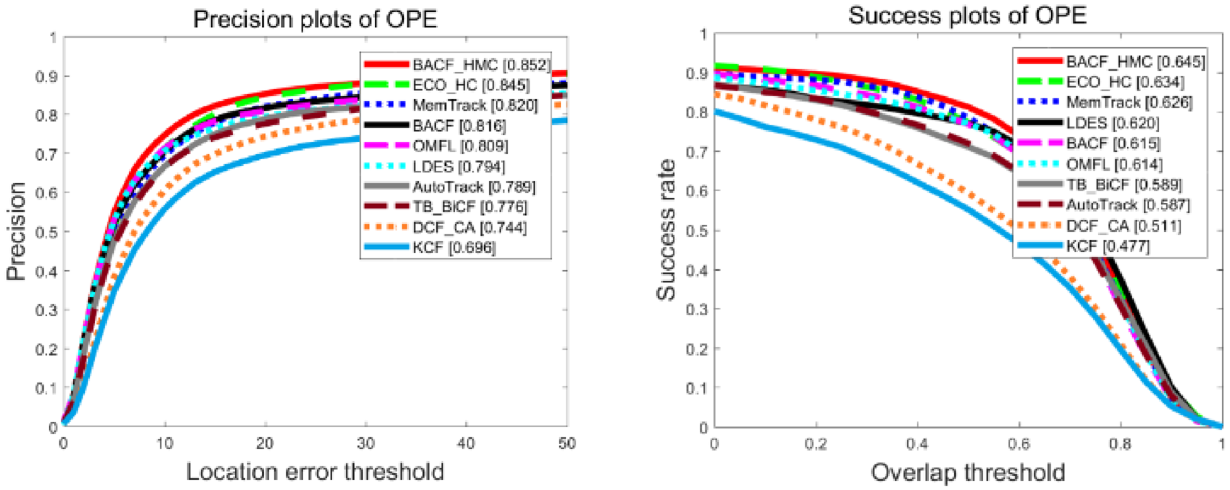


Fig. 6 Overall precision and success plots of BACF_HMC and other monitoring algorithms, namely, ECO_HC, MenTrack, BACF, OMFL, LDES, AutoTrack, TB_BiCF, DCF_CA, and KCF, over the OTB100 dataset. References of each method considered in these plots can be found in Table 1. For the sake of clarity

in these plots, only the top-10 tracking algorithms are considered in these plots.

BACF_HMC outperformed all the other algorithms on the benchmark. In particular, the precision of the original BACF monitoring algorithm was 0.816 and the success rate was 0.615. The precision and success rates of the BACF HMC algorithm with multiple visual cognition mechanisms were 0.852 and 0.645, respectively. Compared to the precision and success rate of the original BACF, the performance of BACF_HMC with multiple visual cognitive mechanisms was better by 4.41% and 4.88%, respectively. For the latest monitoring algorithm, LUDT+, published in 2021, the precision and success rates were 0.843 and 0.639, respectively. The precision and success rate of BACF_HMC improved on the performance of state-of-the-art methods by 1.07% and 0.94%, respectively. For 2020, the precision of the monitoring algorithm that performed best at the time (AutoTrack) was 0.789, whereas its success rate was 0.587. BACF_HMC surpasses these scores considerably, with higher precision and a better success rate (7.98% and 9.88%, respectively). As shown by these results, by virtue of BACF_HMC's integration of multiple visual cognitive flows, it achieved significant improvements in performance compared with the best approaches reported so far for this dataset.

Table 1: Comparison of BACF_HMC to 23 state-of-the-art tracking algorithms over the OTB100 dataset. The top three results are shown in orange, green, and blue, respectively.

Tracking algorithms [ref] (Venue)	**Precision**	**AUC**	**Tracking algorithms [ref] (Venue)**	**Precision**	**AUC**
KCF [11] (TPAMI 2015)	0.696	0.477	MenTrack [32] (ECCV 2018)	0.820	0.626
SiamFC [50] (ECCV 2016)	0.771	0.582	SACF [46] (ECCV 2018)	0.839	0.633
Staple [40] (CVPR 2016)	0.784	0.581	Siam-tri [45] (ECCV 2018)	0.781	0.592
ACFN [39] (CVPR 2017)	0.799	0.573	StructSiam [42] (ECCV 2018)	0.851	0.621
BACF [12] (ICCV 2017)	0.816	0.615	LDES [13] (AAAI 2019)	0.794	0.620
CFNet [49] (CVPR 2017)	0.748	0.568	OMFL [34] (MDPI 2019)	0.809	0.614
DCF_CA [38] (CVPR 2017)	0.744	0.511	SiamFC+ [30] (CVPR 2019)	0.581	0.640
ECO_HC [31] (CVPR 2017)	0.845	0.634	UDT [48] (CVPR 2019)	0.760	0.594
LMCF [35] (CVPR 2017)	0.789	0.580	AutoTrack [36] (CVPR 2020)	0.789	0.587
RFL [33] (ICCVW 2017)	0.778	0.581	TB_BiCF [37] (ICRA 2020)	0.776	0.589
HP [47] (CVPR 2018)	0.796	0.601	LUDT+ [41] (IJCV 2021)	0.843	0.639
MDNet [44] (ECCV 2018)	0.885	0.650	BACF_HMC (our)	0.852	0.645

To further illustrate the precision of the BACF_HMC algorithm fused with multiple visual cognition mechanisms, we compared the monitoring algorithms for other attributes on the OTB100 dataset. The experimental results are shown in Figure 7.

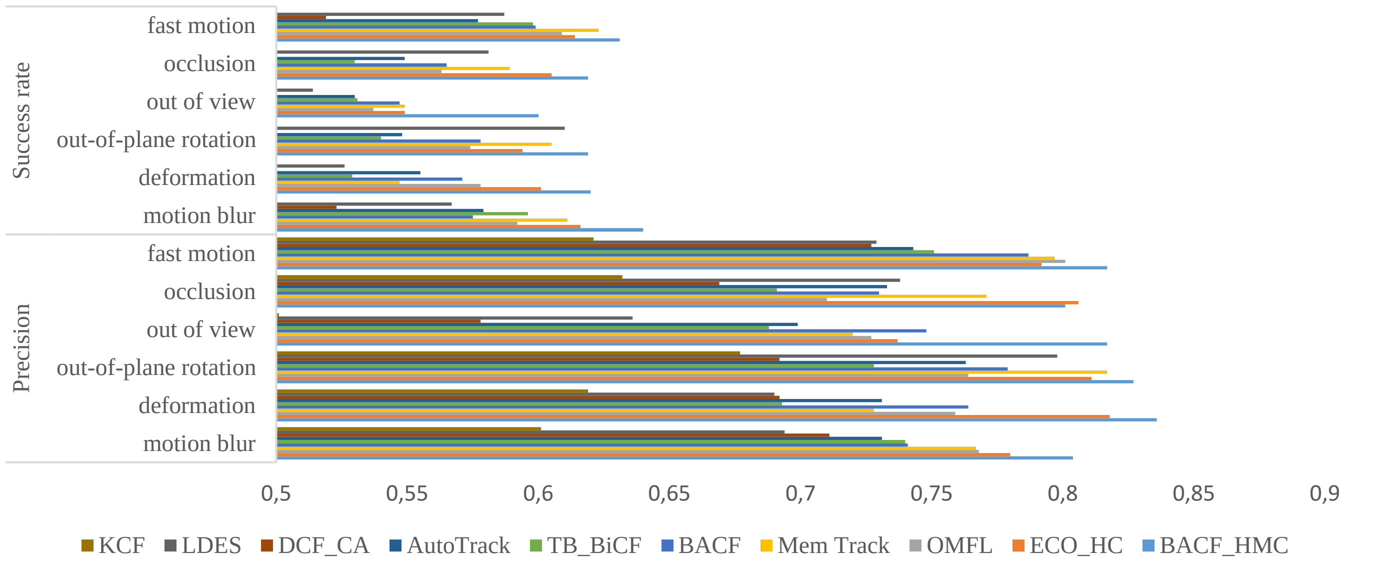


Fig. 7 Comparison of the success rate and precision of BACF_HMC with those of the other state-of-the-art tracking algorithms (ECO_HC [31], OMFL [34], MemTrack [32], BACF [12], TB_BiCF [37], DCF_CA [38], LDES [13], and KCF [11]) considering different categories of sequences defined in the OTB100 dataset.

Figure 7 shows the success rate and precision comparison between our BACF_HMC and other relevant monitoring algorithms in terms of fuzzy motion, low resolution, out-of-plane rotation, out of view, occlusion, fast movement, and other challenging attributes.

As depicted by the scores in Figure 7, the performance of the proposed BACF_HMC approach is considerably superior to those of ECO_HC, OMFL, MemTrack, BACF, TB_BiCF, DCF_CA, LDES, and KCF under fuzzy motion, low resolution, out-of-plane rotation, out of field of view, occlusion, fast movement, and other challenging conditions defined in the challenge. When dealing with video sequences with out of view events, BACF_HMC significantly outperforms BACF and ECO_HC, with relative gains of 6.9 % (precision) and 5.1 % (success rate). This is achieved owing to the utilization of multiple features to improve the overall performance, while using a memory update mechanism to retrace the target after it is lost.

B. *VOT2016 and VOT2018*

In the second part of this study, BACF_HMC is compared with monitoring algorithms proposed over the years for the VOT2016 and VOT2018 datasets:

- VOT2016: BWRR [58], HCF [59], DSST [52], SCT [60], MOSSE_CA [38], RFL [33], and monitoring algorithms such as SRDCF [54], HCFT [62], TGPR [63], SAMF [53], and Struck [65].
- VOT2018: RT-MDNET+RandAtt [51], BACF [12], KCF [11], DSST [52], SAMF [53], SRDCF [54], CSR-DCF [55], WSCF_ST [56], DeepSRDCF [57], SiamFC [50], and LUDT [41].

Table 2 presents the results. For the VOT2018 dataset, BACF_HMC outperforms the very recent LUDT approach proposed in [41] with a relative gain of 33.12% and 9.57%, respectively, in terms of the EAO and accuracy, whereas its robustness decreases by 17.39%. Compared with the RT-MDNET +RandAtt algorithm, the values of the EAO and accuracy of BACF_HMC are greater by 49.64% and 0.19%, respectively, whereas the robustness decreases by 34.48%. Similarly, WSCF_ST is surpassed by BACF_HMC with performance levels boosted by 7.14% (EAO) and 11.41% (accuracy) and a decay of 73.73% in terms of robustness.

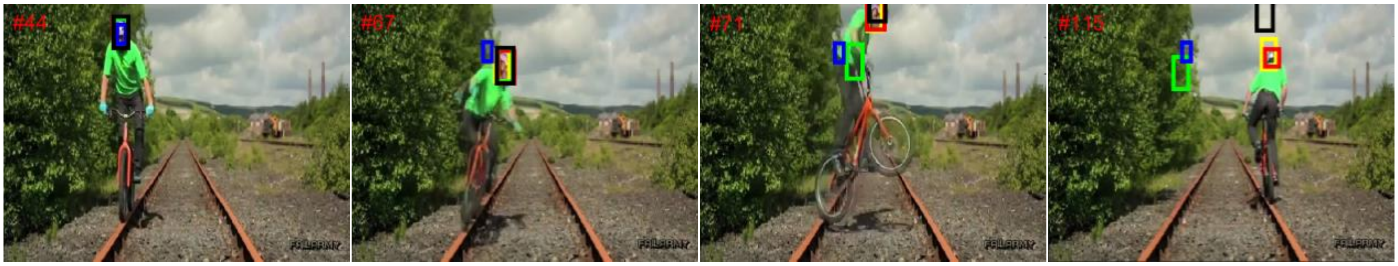
Table 2: Comparison of our BACF_HMC on VOT2016 and VOT2018 datasets with other latest algorithms. The top three results are shown in orange, green, and blue.

VOT2016				**VOT2018**			
Tracking algorithms [Ref] (Venue)	**A ↑**	**R ↓**	**EAO ↑**	**Tracking algorithms [Ref.] (Venue)**	**A ↑**	**R ↓**	**EAO ↑**
Struck [65] (ICCV 2011)	0.439	3.37	0.142	DSST [52] (BMVC 2014)	0.388	5.36	0.083
DSST [52] (BMVC 2014)	0.537	0.52	0.181	SAMF [53] (ECCV 2014)	0.463	2.33	0.151
SAMF [53] (ECCV 2014)	0.498	37.79	0.186	DeepSRDCF [57] (ICCVW 2015)	0.492	0.71	0.154
TGPR [63] (ECCV 2014)	0.452	41.01	0.181	KCF [11] (TPAMI 2015)	0.440	0.75	0.137
HCF [59] (ICCV 2015)	0.467	1.39	0.231	SRDCF [54] (ICCV 2015)	0.477	3.77	0.122
HCFT [62] (ICCV 2015)	0.471	1.38	0.220	SiamFC [50] (ECCV 2016)	0.500	0.58	0.188
SRDCF [54] (ICCV 2015)	0.527	1.50	0.241	BACF [12] (ICCV 2017)	0.492	0.76	0.141
SCT [60] (CVPR 2016)	0.480	0.55	0.188	CSR-DCF [55] (CVPR 2017)	0.487	1.25	0.268
MOSSE_CA [38] (CVPR 2017)	0.420	0.71	0.162	RT-MDNet [51] (ECCV 2020)	0.503	0.87	0.137
RFL [33] (ICCVW 2017)	0.524	0.59	0.223	WSCF_ST [56] (TIP 2020)	0.540	2.17	0.184
BWRR [58] (TMM 2021)	0.540	1.37	0.289	LUDT [41] (IJCV 2021)	0.460	0.69	0.154
BACF_HMC (our)	0.547	0.45	0.246	BACF_HMC (our)	0.504	0.57	0.205

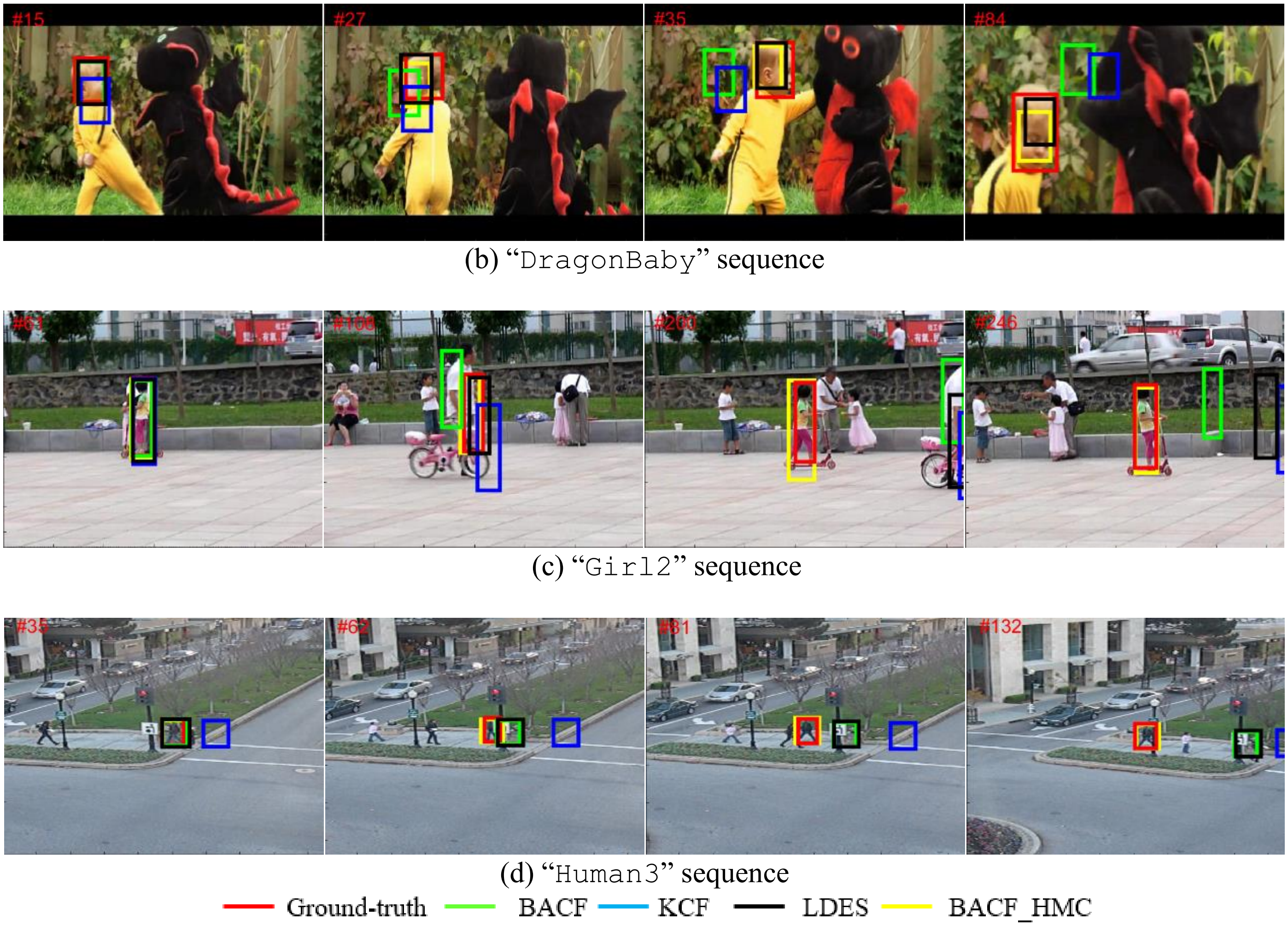
Similar conclusions can be drawn from the results obtained for the VOT2016 challenge, as shown in Table 2. Significant performance improvements compared to the state-of-the-art methods were observed with our proposed approach. First, BACF_HMC outperformed the BWRR algorithm published in 2021, with a relative gain of 17.48% (EAO) and 1.3% (accuracy), whereas the robustness score of BACF_HMC decreased by 67.15%. When compared to MOSSE_CA, EAO and accuracy improvements were noted for BACF_HMC, scoring 51.85% and 30.24% higher for these scores, respectively. Robustness decreased by 36.62%, indicating a more robust behavior of BACF_HMC. Similarly, compared with RFL, the EAO and accuracy scores of BACF_HMC were improved by 10.31% and 4.39%, respectively, whereas robustness was enhanced by 23.73%.

5.2 Qualitative analysis

In addition to the quantitative analysis given in the previous subsection, we selected key frames of several representative video sequences (`"bike,"` `"DragonBaby,"` `"Girl2,"` and `"Human3"`) for qualitative analysis. We aimed to compare the performance of our BACF_HMC with that of the naïve BACF algorithm [12]. Figure 8 shows a qualitative comparison between the output of BACF_HMC, naïve BACF, and other algorithms from the recent literature (specifically KCF and LDES).



(a) “`Bike`” sequence



(b) "`DragonBaby`" sequence

(c) "`Girl2`" sequence

(d) "`Human3`" sequence

Fig. 8 Comparison of monitoring effects among BACF_HMC, BACF [12], KCF [11], and LDES [13] in different sequences.

In Figure 8, the red box represents the ground truth of the target, green box represents the target box monitored by the BACF [12] algorithm, blue box represents the target box monitored by the KCF [11] algorithm, black box represents the target box monitored by the LDES [13] algorithm, and yellow box represents the target box monitored by the BACF_HMC algorithm.

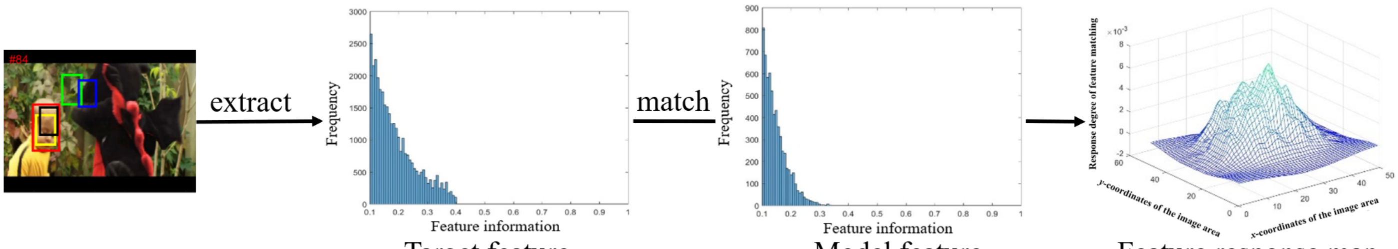
In the "`bike`" sequence shown in Figure 8a, the target involves five difficult situations: target rotation, fast movement, scale change, passing out of view, and low resolution. The target in frame 44 is in normal motion, and all monitoring algorithms included in this qualitative analysis can track it accurately. However, the target starts to rotate, suffers from scale changes, and moves rapidly at the 67th frame. Consequently, KCF fails to monitor it properly beyond this point of the frame sequence. In frame 71, the contextual changes (target rotation, scale change, and fast movement) are exacerbated sharply. At this point, the target information features extracted by the naïve BACF approach cannot match the target template information well, thus causing monitoring failures and loss of the target. However, LDES and BACF_HMC perform accurate monitoring, even under these circumstances. When the target continues to move to the 115th frame and the target rotation, scale change, and fast movement continue to increase, such that the LDES fails to monitor the target accurately at that time. By contrast, BACF_HMC can perform robust monitoring under these circumstances by leveraging its three core functionalities (multiple visual cognitive information extraction, cognitive information fusion, and feature template memory storage and retrieval strategy).

In Figure 8b, the target of the "DragonBaby" sequence is subject to four complex imagery artifacts that pose a challenge for tracking algorithms: scale change, occlusion, low resolution, and plane rotation. At the 15th frame, the target remains in a normal walking state, and all the algorithms can track it accurately until this point. However, in the 27th frame and the subsequent 35 frames, plane rotation, scale change, and low resolution occur, affecting the target. As a result, the KCF and naïve BACF algorithms cannot sustain accurate visual monitoring. However, the LDES and BACF_HMC maintain good tracking performance. At the 84th frame, the scale of the target begins to vary. At this point, the feature information extracted by KCF and the naïve version of BACF is mismatched with the target template information, so that the monitored target box exhibits a large deviation. However, BACF_HMC performs robustly at a good tracking quality because it uses color features and other information flows for cognitive monitoring. This assists BACF_HMC in accommodating complex visual artifacts and events, such as scale changes, occlusions, low resolution, and plane rotation.

As shown in Figure 8c, the target of the "Girl2" sequence suffers from scale change, occlusion, fuzzy motion, plane rotation, and low resolution. At the 61st frame, the target moves normally, and all algorithms can accurately monitor it. However, in the 108th frame, the target is occluded, and the position estimation of the KCF and naïve BACF deviates significantly with respect to the real location of the target in the frame. Only LDES and BACF_HMC perform accurate monitoring. At the 200th frame, the LDES is unable to conduct real-time monitoring because of the occlusion of the target and the low resolution and rotation observed at this time. However, BACF_HMC still performs reliably when monitoring the target, making it a suitable tracker for sequences subject to target scale change, occlusion, fuzzy motion, plane rotation, and low resolution, particularly for complex scenes.

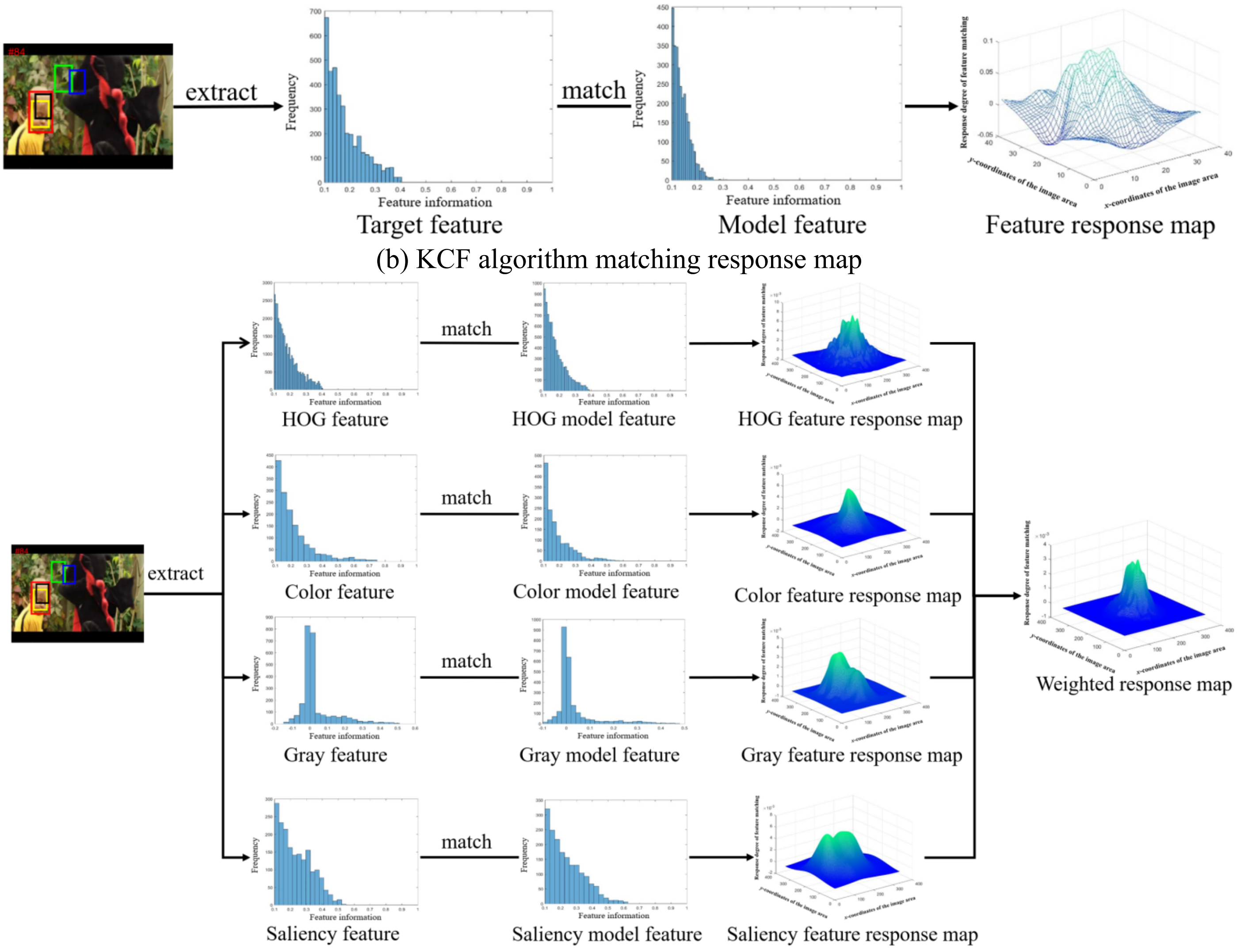
In Figure 8d, the target of the "Human3" sequence is shown to pass through background clutter, occlusion, and low-resolution artifacts. At the 35th frame, the target is in background clutter with low resolution, and the naïve KCF method can no longer perform accurate monitoring. When the target moves at the 62nd frame, the occlusion problem occurs again, and BACF and LDES cannot extract multiple information features of the target, resulting in cognitive monitoring errors. However, our proposed BACF_HMC tracker can leverage multiple visual cognitive information flows to perform accurate monitoring under such circumstances. In the subsequent 81st and 132nd frames, BACF_HMC still accurately tracks the target, demonstrating its good performance under conditions with background clutter, occlusion, and low resolution, which are common in complex scenes.

As shown in Figure 9, we select the 84th frame of the "DragonBaby" sequence for an experimental comparative analysis. This sequence covers the challenges of deformation, occlusion, motion blur, fast motion, out-of-plane rotation, and out of view, which are representative of the types of events experienced by drone monitoring, air early warning, and traffic supervision.



Target feature Model feature Feature response map

(a) BACF algorithm matching response map



(b) KCF algorithm matching response map

(c) BACF_HMC algorithm matching the response map

Fig. 9 Feature extraction and feature matching of KCF [11], BACF [12], and BACF_HMC algorithms in frame# 84 of the "`DragonBaby`" sequence. The abscissa and ordinate of the histogram are “feature information” and “frequency” respectively. The three-dimensional coordinates of the matching response map are the *x*- and *y*-coordinates of the image area and the response degree of feature matching.

As displayed in Figure 9a, the matching response graph obtained by matching the target feature information with the template feature information shows the matching response result of the naïve BACF algorithm in frame 84th. As shown in Figure 9b, the matching response result of this algorithm in the mentioned frame is obtained by matching the target feature information with template feature information. As depicted in Figure 9c, different target feature information and template feature information are matched to obtain matching response results of different quality levels, and these matching response results are fused based on such levels of quality to yield the final matching response results. The matching response map in Figure 9c shows the matching response result of the BACF_HMC algorithm in the 84th frame.

The matching response result reveals that the extracted features and the template feature matching obtained by naïve BACF and KCF are not good, resulting in a monitoring failure, as shown in Figure 9b, owing to the change in scale occurring in the 84th frame. The feature information extracted by KCF and BACF cannot match the target template information. In this case, the BACF_HMC algorithm achieves realistic matching results by increasing the proportion of color features and by using them along with other information flows for cognitive monitoring.

Naïve BACF and KCF use only single-feature template matching. Therefore, when abnormal situations beyond the characteristic domain occur in a scene, matching results often fail. In contrast, BACF_ HMC analyzes the quality of different features in different scenes through its response matrix and resorts to a variety of visual cognitive mechanisms to address tracking problems derived from such abnormal visual artifacts, such as changes in scale, occlusion, fuzzy motion, and background clutter. Therefore, this limitation can be significantly reduced. This enables BACF_HMC to maintain a superior level of monitoring performance in complex and nonstationary scenes.

5.3 Limitations

Figure 10 shows the limitations of the LUDT+ algorithm compared with BACF_HMC. LUDT+ fails more easily when monitoring under variable lighting conditions. As may be observed in the singer2 sequence, when BACF_HMC and LUDT+ lose the target at the same time, our algorithm can quickly recover and monitor it again owing to its feature template memory storage and retrieval strategy. In the case of rotation and occlusion, the monitoring performance of LUDT+ is worse than that of our algorithm.

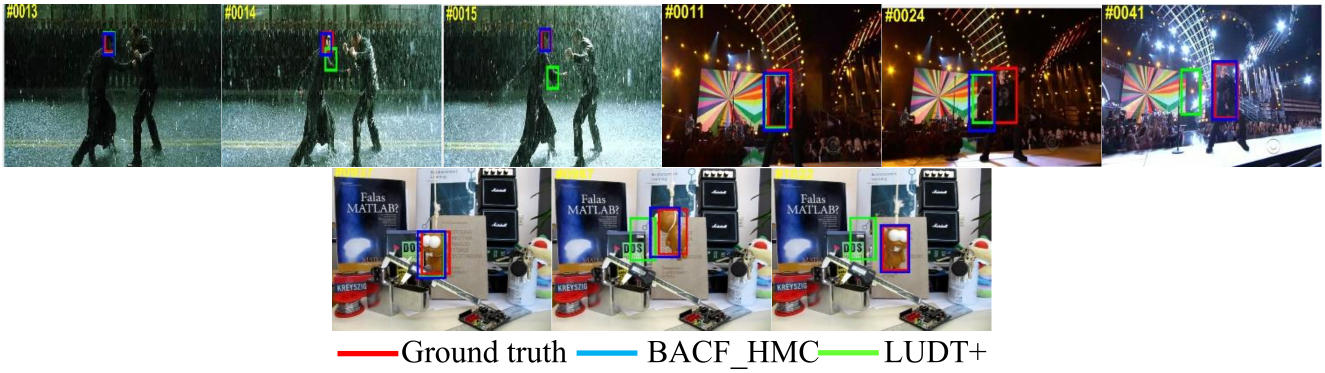


Fig. 10 Failure cases of the LUDT+ [41] tracking algorithm. The first video is a short sequence from the film The Matrix (`matrix`), the second video is referred to as `singer2`, and the third video is `lemming`. When illumination and rotation occur together, the proposed BACF_HMC approach performs better than LUDT+.

Figure 11 highlights the possible limitations of BACF_HMC. First, unlike the four features used in RGB image tracking, our algorithm can use only HOG and gray features when the tracking sequence is a gray image. Neither color nor salience features can be extracted from a grey image. Consequently, tracking cannot be optimally performed. In any case, future extensions that use other manually crafted features such as local binary pattern features with gray invariance can also be considered. Such new features can be combined with HOG and gray cognitive information flows to improve monitoring performance. Second, BACF_HMC could not adapt well to drastic changes in the target aspect ratio. When the aspect ratio changed in a continuous fashion, it inevitably introduced a substantial amount of non-semantic information, resulting in the algorithm being unable to track the target precisely. At this time, the feature template memory storage and retrieval strategy did not work, which may be because the target changed significantly in a short time and our memory strategy only stored the features of a single frame. At this time, the template features stored in the memory and the features of the previous frame were not able to match the features of the current monitoring frame well, resulting in a loss. Even if the occurrence of this drastic change in the aspect ratio of the target is relatively rare in real-world settings, possible extensions can be directed towards introducing additional filters to deal with these sharp scale changes, thus improving the resiliency of BACF_HMC at the cost of a higher computational complexity. In subsequent work, we will consider whether this strategy can further improve performance by adding memory-stored templates.

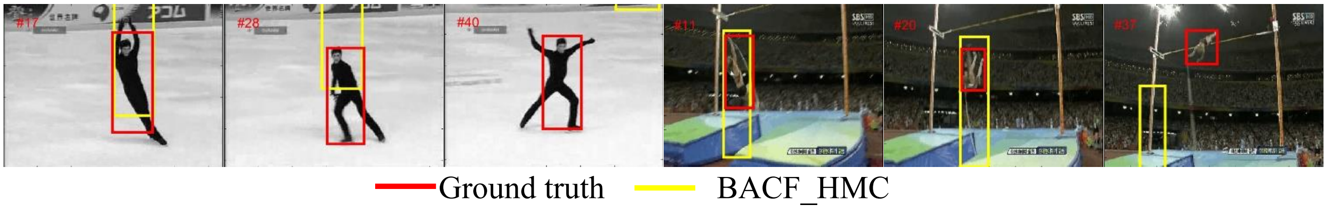


Fig. 11 Failure cases of the proposed BACF_HMC tracking algorithm. The top and bottom videos are `skater2` and `jump`. When the gray image and the aspect ratio change dramatically, our method does not perform well and fails to correctly track the monitored target.

6. Conclusion

Recently, deep learning has been widely applied in most areas of computer vision. Visual monitoring is no exception to this trend. However, deep-learning algorithms used for visual monitoring can only be locally generalized and adapted to new situations whenever the visual data are similar to past data. By contrast, visual cognition can be extremely generalizable and can adapt to unique and novel situations because of abstract modeling. Although existing CF-based methods can quickly use a single feature to obtain the target location for visual monitoring, when facing complex scenes, these methods cannot fully express the modeling of changeable target appearances, ultimately leading to inaccurate monitoring and positioning. Moreover, the CF-based methods do not consider that the tracker may lose the target. Thus, retrieving the target using the original template update strategy may be difficult, leading to unreliable feature extraction, which can have a negative impact on the tracking results.

Based on these considerations and to address the associated limitations of the current state-of-the-art in the field, we introduced the theory of location fusion based on multiple visual cognition and combined it with state-of-the-art CF-based monitoring algorithms to yield a novel visual monitoring approach that performs efficiently in complex scenes by leveraging the inherent benefits of simpler and more understandable approaches than deep-learning methods. Thus, the stages involved in the proposed approach (multiple visual cognitive information extraction, cognitive information fusion, and feature template memory storage and retrieval strategy) are mathematically simple, do not require high computational resources, and reliably use information fusion theory to complete high-quality visual monitoring. The results of extensive experiments and a performance comparison with 36 recent visual monitoring algorithms were conclusive. The proposed method not only improved the robustness of monitoring algorithms in complex scenes but also highlighted a promising research path in terms of the application of information fusion to the design of computational methods for visual monitoring. The limitations of the proposed algorithm were identified and explained by using illustrative examples, tracing different research directions aimed at circumventing them effectively.

In summary, the location fusion mechanism based on multiple visual cognitions does not depend on specific features but instead combines multiple features and related memories to achieve the abstract modeling ability of human cognition in case of hypothetical scenes. Our approach identifies and improves the shortcomings of the model over time. Visual monitoring is more complicated than simply stacking additional layers or using more training data to expand the scope of applications. In future work, we need to find more representative features to further improve the accuracy of monitoring. We also need to consider ways to solve the problem of computational slowdown in the use of more diverse features to resiliently track targets under particularly challenging visual conditions.

Acknowledgements

This work was supported by the Natural Science Foundation of China with No. 62207012; Natural Science Foundation of Hunan Province with No. 2020JJ4434, 2020JJ5368, Key Scientific Research Projects of Department of Education of Hunan Province with No. 19A312; Key Research Project on Degree and

Graduate Education Reform of Hunan Province with No. 2020JGZD025; the National Social Science Foundation of China with No. AEA200013. This work was also supported in part by the National Research Foundation of Korea Grant funded by the Korean Government (MSIT) under Grant 2019M3F2A1073179. J. Del Ser acknowledges funding support from the Basque Government through ELKARTEK and EMAITEK funds as well as the Consolidated Research Group MATHMODE (IT1456-22).

References

[1] Y. Himeur, B. Rimal, A. Tiwary, & A. Amira, (2022), "Using artificial intelligence and data fusion for environmental monitoring: A review and future perspectives," Information Fusion (INF), 86, 44-75.

[2] A. -M. Yang, J. -M. Zhi, K. Yang, J. -H. Wang and T. Xue, (2021), "Computer Vision Technology Based on Sensor Data and Hybrid Deep Learning for Security Detection of Blast Furnace Bearing," in IEEE Sensors Journal (JSEN), 21(22), 24982-24992.

[3] M. Afifi, A. Abdelhamed, A. Abuolaim, A. Punnappurath and M. S. Brown, (2021), "CIE XYZ Net: Unprocessing Images for Low-Level Computer Vision Tasks," in IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 44(9), 4688-4700.

[4] K. Muhammad, S. Khan, J. D. Ser and V. H. C. d. Albuquerque, (2021), "Deep Learning for Multigrade Brain Tumor Classification in Smart Healthcare Systems: A Prospective Survey," in IEEE Transactions on Neural Networks and Learning Systems (TNNLS), 32(2), pp. 507-522.

[5] A. Diez-Olivan, J. Del Ser, D. Galar, & B. Sierra, (2019), "Data fusion and machine learning for industrial prognosis: Trends and perspectives towards Industry 4.0," Information Fusion (INF), 50, 92-111.

[6] K. Muhammad, A. Ullah, J. Lloret, J. D. Ser and V. H. C. de Albuquerque, (2021), "Deep Learning for Safe Autonomous Driving: Current Challenges and Future Directions," in IEEE Transactions on Intelligent Transportation Systems (TITS), 22(7), pp. 4316-4336.

[7] W. Ren, X. Wang, J. Tian, Y. Tang and A. B. Chan, (2021) "Tracking-by-Counting: Using Network Flows on Crowd Density Maps for Tracking Multiple Targets," in IEEE Transactions on Image Processing (TIP), vol. 30, pp. 1439-1452.

[8] J. Wang, D. Rodriguez, A. Mishra, P. R. Nallabolu, T. Karp and C. Li, (2021), "24-GHz Impedance-Modulated BPSK Tags for Range Tracking and Vital Signs Sensing of Multiple Targets Using an FSK Radar," in IEEE Transactions on Microwave Theory and Techniques (TMTT), 69(3),1817-1828.

[9] Y. Wang, Y. Wu and Y. Shen, (2020), "Cooperative Tracking by Multi-Agent Systems Using Signals of Opportunity," in IEEE Transactions on Communications (TCOMM), 68(1), 93-105.

[10] A. C. Huk, & D. J. Heeger, (2002), "Pattern-motion responses in human visual cortex," Nature neuroscience, 5(1), 72-75.

[11] J. F. Henriques, R. Caseiro, P. Martins and J. Batista, (2015), "High-Speed Tracking with Kernelized Correlation Filters," in IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 37(3), 583-596.

[12] H. K. Galoogahi, A. Fagg and S. Lucey, (2017), "Learning Background-Aware Correlation Filters for Visual Tracking," in IEEE International Conference on Computer Vision (ICCV), pp. 1144-1152.

[13] Li, Yang, et al. (2019) "Robust Estimation of Similarity Transformation for Visual Object Tracking. " Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 33(1), 8666–8673.

[14] Y. Cao, X. Luo, J. Yang, Y. Cao, & M. Y. Yang, (2022), "Locality guided cross-modal feature aggregation and pixel-level fusion for multispectral pedestrian detection, " Information Fusion (INF), 88, pp. 1-11.

[15] H. Zhang, H. Xu, X. Tian, J. Jiang, & J. Ma, (2021). "Image fusion meets deep learning: A survey and perspective, " Information Fusion (INF), 76, pp. 323-336.

[16] B. Li, J. Yan, W. Wu, Z. Zhu and X. Hu, (2018), "High Performance Visual Tracking with Siamese Region Proposal Network," IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8971-8980.

[17] Y. Wu, J. Lim and M. Yang, (2015), "Object Tracking Benchmark," in IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 37(9),1834-1848.

[18] Kristan, Matej, et al. (2016), "The Visual Object Tracking VOT2016 Challenge Results. " Proceedings, European Conference on Computer Vision (ECCV) Workshops, vol. 9914, pp. 777–823.

[19] Kristan M. et al. (2019), The Sixth Visual Object Tracking VOT2018 Challenge Results. In: Leal-Taixé L., Roth S. (eds) Computer Vision – ECCV 2018 Workshops. ECCV 2018. Lecture Notes in Computer Science, vol 11129. Springer, Cham.

[20] Z. Hong, Z. Chen, C. Wang, X. Mei, D. Prokhorov, & D. Tao, (2015) "Multi-store tracker (muster): A cognitive psychology inspired approach to object tracking, " In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 749-758.

[21] B. Cai, X. Xu, X. Xing, K. Jia, J. Miao and D. Tao, (2016), "BIT: Biologically Inspired Tracker," in IEEE Transactions on Image Processing (TIP), 25(3), 1327-1339.

[22] Srivastava, Nisheeth, and Ed Vul. (2016) "Attention Modulates Spatial Precision in Multiple - Object Tracking." Topics in Cognitive Science, 8(1), 335-348.

[23] J. Zhan, H. Zhao, P. Zheng, et al. (2021), Salient Superpixel Visual Tracking with Graph Model and Iterative Segmentation. Cogn Comput 13, 821–832.

[24] Kumar B V K V, Mahalanobis A, Juday R D. (2005), Correlation pattern recognition[M]. Cambridge university press.

[25] D. S. Bolme, J. R. Beveridge, B. A. Draper and Y. M. Lui, (2010), "Visual object tracking using adaptive correlation filters," IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2544-2550.

[26] X. Zhang, P. Ye, H. Leung, K. Gong, & G. Xiao, (2020). "Object fusion tracking based on visible and infrared images: a comprehensive review, " Information Fusion (INF), 63, pp. 166-187.

[27] F. Li, C. Tian, W. Zuo, L. Zhang and M. Yang, (2018), "Learning Spatial-Temporal Regularized Correlation Filters for Visual Tracking," IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4904-4913.

[28] M. Danelljan, G. Häger, F. S. Khan and M. Felsberg, (2017), "Discriminative Scale Space Tracking," in IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 39(8), 1561-1575.

[29] H. Fan et al., (2019), "Lasot: A high-quality benchmark for large-scale single object tracking," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5374-5383.

[30] Z. P. Zhang, H. W. Peng. (2019), "Deeper and Wider Siamese Networks for Real-Time Visual Tracking. " IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4591–4600.

[31] Danelljan, Martin, et al. (2017), "ECO: Efficient Convolution Operators for Tracking." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6931–6939.

[32] T. Y. Yang, Antoni B. Chan. (2018), "Learning Dynamic Memory Networks for Object Tracking. " Proceedings of the European Conference on Computer Vision (ECCV), pp. 153–169.

[33] T. Y. Yang, Antoni B. Chan. (2017) "Recurrent Filter Learning for Visual Tracking." IEEE International Conference on Computer Vision Workshops (ICCVW), pp. 2010–2019.

[34] C. H. Fu, et al. (2019), "Correlation Filter-Based Visual Tracking for UAV with Online Multi-Feature Learning." Remote Sensing, 11(5), 549.

[35] M. Wang, Y. Liu and Z. Huang, (2017), "Large Margin Object Tracking with Circulant Feature Maps," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4800-4808

[36] Y. M. Li, et al. (2020), "AutoTrack: Towards High-Performance Visual Tracking for UAV With Automatic Spatio-Temporal Regularization." IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 11923–11932.

[37] F. L. Lin, et al. (2020), "BiCF: Learning Bidirectional Incongruity-Aware Correlation Filter for Efficient UAV Object Tracking." IEEE International Conference on Robotics and Automation (ICRA), pp. 2365–2371.

[38] M. Mueller, N. Smith and B. Ghanem, (2017), "Context-Aware Correlation Filter Tracking," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.1387-1395.

[39] J. Choi, H. J. Chang, S. Yun, T. Fischer, Y. Demiris and J. Y. Choi, (2017), "Attentional Correlation Filter Network for Adaptive Visual Tracking," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4828-4837.

[40] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik and P. H. S. Torr, (2016), "Staple: Complementary Learners for Real-Time Tracking," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1401-1409.

[41] Wang, Ning, et al. (2021), "Unsupervised Deep Representation Learning for Real-Time Tracking. " International Journal of Computer Vision (IJCV), 129(2), pp. 400–418.

[42] Y. H. Zhang, et al. (2018), "Structured Siamese Network for Real-Time Visual Tracking. " Proceedings of the European Conference on Computer Vision (ECCV), pp. 355–370.

[43] Y. Zhao, Y. Yin, & G. Gui. (2020). "Lightweight deep learning based intelligent edge surveillance techniques." IEEE Transactions on Cognitive Communications and Networking, 6(4), 1146-1154.

[44] Jung, Ilchae, et al. (2018), "Real-Time MDNet. " Proceedings of the European Conference on Computer Vision (ECCV), pp. 89–104.

[45] X. P. Dong, J. B. Shen. (2018), "Triplet Loss in Siamese Network for Object Tracking. " Proceedings of the European Conference on Computer Vision (ECCV), pp. 472–488.

[46] M. D. Zhang, et al. (2018), "Visual Tracking via Spatially Aligned Correlation Filters Network. " Proceedings of the European Conference on Computer Vision (ECCV), pp. 469–485.

[47] X. Dong, J. Shen, W. Wang, Y. Liu, L. Shao and F. Porikli. (2018), "Hyperparameter Optimization for Tracking with Continuous Deep Q-Learning," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 518-527.

[48] N. Wang, Y. Song, C. Ma, W. Zhou, W. Liu and H. Li. (2019), "Unsupervised Deep Tracking," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.850-865.

[49] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi and P. H. S. Torr. (2017), "End-to-End Representation Learning for Correlation Filter Based Tracking," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5000-5008.

[50] L. Bertinetto, J. Valmadre, J. Henriques, A. Vedaldi, P. H. S. Torr. (2016), "Fully-Convolutional Siamese Networks for Object Tracking, " European Conference on Computer Vision (ECCV), pp. 850–865.

[51] S. Jia, C. Ma, Y. B. Song and X. K. Yang. (2020), "Robust Tracking Against Adversarial Attacks, " European Conference on

Computer Vision (ECCV), pp. 69–84.

[52] M. Danelljan, G. Häger, F. Khan, and M. Felsberg. (2014), "Accurate Scale Estimation for Robust Visual Tracking, " in Proceedings of the British Machine Vision Conference (BMVC), pp. 1-5.

[53] Y. Li, J. K. Zhu. (2015), "A Scale Adaptive Kernel Correlation Filter Tracker with Feature Integration. " European Conference on Computer Vision (ECCV), pp. 254–265.

[54] M. Danelljan, G. Häger, F. S. Khan and M. Felsberg. (2015), "Learning Spatially Regularized Correlation Filters for Visual Tracking," 2015 IEEE International Conference on Computer Vision (ICCV), pp. 4310-4318.

[55] A. Lukežic, T. Vojír, L. C. Zajc, J. Matas and M. Kristan. (2017), "Discriminative Correlation Filter with Channel and Spatial Reliability," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4847-4856.

[56] R. Han, W. Feng and S. Wang. (2020), "Fast Learning of Spatially Regularized and Content Aware Correlation Filter for Visual Tracking," in 2020 IEEE Transactions on Image Processing (TIP), vol. 29, pp. 7128-7140.

[57] M. Danelljan, G. Häger, F. S. Khan and M. Felsberg. (2015), "Convolutional Features for Correlation Filter Based Visual Tracking," 2015 IEEE International Conference on Computer Vision Workshop (ICCVW), pp. 621-629.

[58] H. Zhu, H. Peng, G. Xu, L. Deng, Y. Cheng and A. Song. (2021), "Bilateral Weighted Regression Ranking Model with Spatial-Temporal Correlation Filter for Visual Tracking," in 2021 IEEE Transactions on Multimedia, (TMM), 24, pp. 2098-2111.

[59] C. Ma, J. Huang, X. Yang and M. Yang. (2015), "Hierarchical Convolutional Features for Visual Tracking," 2015 IEEE International Conference on Computer Vision (ICCV), pp. 3074-3082.

[60] J. Choi, H. J. Chang, J. Jeong, Y. Demiris and J. Y. Choi. (2016), "Visual Tracking Using Attention-Modulated Disintegration and Integration, " 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4321–4330.

[61] R Gutiérrez, V Rampérez, Paggi, H. , Lara, J. A., & Soriano, J. (2022), "On the use of information fusion techniques to improve information quality: taxonomy, opportunities and challenges, " Information Fusion (INF), 78, pp. 102-137.

[62] C. Ma, J. B. Huang, X. K. Yang and M. H. Yang. (2015), "Hierarchical Convolutional Features for Visual Tracking," 2015 IEEE International Conference on Computer Vision (ICCV), pp. 3074-3082.

[63] J. Gao, H. B. Ling, W. M. Hu, J. L. Xing. (2014), "Transfer Learning Based Visual Tracking with Gaussian Processes Regression, " European Conference on Computer Vision (ECCV), pp. 188–203.

[64] Y. F. Zhang, C. Y. Wang, X. J. Wang, W. J. Zeng and W. Y. Liu. (2021), "Fairmot: On the fairness of detection and re-identification in multiple object tracking, " International Journal of Computer Vision (IJCV), 129(11), 1-19.

[65] S. Hare, S. Golodetz, A. Saffari, et al. (2016), "Struck: Structured Output Tracking with Kernels," in IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 38(10), 2096-2109.

[66] Y. H. Zhang, L. J. Wang, D. Wang, J. Q. Qi, H. C. Lu. (2021), "Learning Regression and Verification Networks for Robust Long-term Tracking, " International Journal of Computer Vision (IJCV), 129(9), 1-12.

[67] S. Shi, Y. Wang, H. Dong, G. Gui, & T. Ohtsuki. (2022). Smartphone-Aided Human Activity Recognition Method Using Residual Multi-Layer Perceptron. In IEEE INFOCOM 2022-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 1-6.