electronics

MDPI

Article

Multiple Cues-Based Robust Visual Object Tracking Method

Baber Khan [1], Abdul Jalil [1], Ahmad Ali [2], Khaled Alkhaledi [3], Khizer Mehmood [1], Khalid Mehmood Cheema [4,5,*], Maria Murad [1], Hanan Tariq [6] and Ahmed M. El-Sherbeeny [7]

1 Department of Electrical Engineering, International Islamic University, Islamabad 44000, Pakistan; baber.khan@iiu.edu.pk (B.K.); abdul.jalil@iiu.edu.pk (A.J.); khizer.mehmood@iiu.edu.pk (K.M.); maria.murad@iiu.edu.pk (M.M.)
2 Department of Software Engineering, Bahria University, Islamabad 44000, Pakistan; ahmad.buic@bahria.edu.pk
3 Industrial and Management Systems Engineering Department, College of Engineering and Petroleum, Kuwait University, P.O. Box 5969, Kuwait City 13060, Kuwait; hf.s@ku.edu.kw
4 Department of Electrical Engineering, Khwaja Fareed University of Engineering & Information Technology, Rahim Yar Khan 64200, Pakistan
5 School of Electrical Engineering, Southeast University, Nanjing 210096, China
6 Faculty of Electrical and Control Engineering, Gdańsk University of Technology, Narutowicza 11/12, 80-233 Gdansk, Poland; hanan.tariq@pg.edu.pl
7 Industrial Engineering Department, College of Engineering, King Saud University, P.O. Box 800, Riyadh 11421, Saudi Arabia; aelsherbeeny@ksu.edu.sa
* Correspondence: kmcheema@seu.edu.cn or km.cheema@kfueit.edu.pk

Abstract: Visual object tracking is still considered a challenging task in computer vision research society. The object of interest undergoes significant appearance changes because of illumination variation, deformation, motion blur, background clutter, and occlusion. Kernelized correlation filter-(KCF) based tracking schemes have shown good performance in recent years. The accuracy and robustness of these trackers can be further enhanced by incorporating multiple cues from the response map. Response map computation is the complementary step in KCF-based tracking schemes, and it contains a bundle of information. The majority of the tracking methods based on KCF estimate the target location by fetching a single cue-like peak correlation value from the response map. This paper proposes to mine the response map in-depth to fetch multiple cues about the target model. Furthermore, a new criterion based on the hybridization of multiple cues i.e., average peak correlation energy (APCE) and confidence of squared response map (CSRM), is presented to enhance the tracking efficiency. We update the following tracking modules based on hybridized criterion: (i) occlusion detection, (ii) adaptive learning rate adjustment, (iii) drift handling using adaptive learning rate, (iv) handling, and (v) scale estimation. We integrate all these modules to propose a new tracking scheme. The proposed tracker is evaluated on challenging videos selected from three standard datasets, i.e., OTB-50, OTB-100, and TC-128. A comparison of the proposed tracking scheme with other state-of-the-art methods is also presented in this paper. Our method improved considerably by achieving a center location error of 16.06, distance precision of 0.889, and overlap success rate of 0.824.

Keywords: artificial intelligence; computer vision; visual object tracking; occlusion

check for updates

Citation: Khan, B.; Jalil, A.; Ali, A.; Alkhaledi, K.; Mehmood, K.; Cheema, K.M.; Murad, M.; Tariq, H.; El-Sherbeeny, A.M. Multiple Cues-Based Robust Visual Object Tracking Method. *Electronics* **2022**, *11*, 345. https://doi.org/10.3390/electronics11030345

Academic Editor: Donghyeon Cho

Received: 19 December 2021
Accepted: 20 January 2022
Published: 24 January 2022

1. Introduction

The vision-based object tracking problem lies in the field of computer vision. It is one of the hot topics of this field because of its large number of applications. In the past decade, the computer vision community has conducted significant work on correlation filter-based tracking algorithms. These algorithms have shown superiority in terms of computational cost. Another advantage of correlation filter-based visual object tracking is its online learning process, which allows updating the template/model in every frame of a video [1,2].

Although a lot of work has been completed on this topic, it is still demanding the attention of the computer vision research community because of associated unwanted factors that ultimately degrade any tracking algorithm's performance. These factors are deformation, partial/full occlusion, out-of-plane rotation of the object, in-plane rotation of the object, the fast and abrupt motion of the object, scale variations, and finally, illumination changes in a video.

Tracking methods may be characterized into two main categories, i.e., (i) deep feature-based methods and (ii) simple hand-crafted feature-based tracking methods. Deep feature-based tracking methods have gained the attention of the tracking community because of their higher accuracy [3,4]. The major issue with these types of tracking methods is the requirement of a higher processing unit and computational cost. Hence for practical scenarios in real-time, a simple hand-crafted feature-based tracking scheme is a better choice [5].

In a broader sense, hand-crafted feature-based tracking schemes consist of two main branches, i.e., generative and discriminative [6]. In generative visual object tracking schemes, the appearance of the target model is represented by learning a model, and then object appearance most closely related to the target model is searched, e.g., [7–9]. In contrast, discriminative approaches are designed to discriminate the target object from its background. As per the literature, discriminative methods are superior in terms of accuracy and computational cost. Some of the examples of these trackers are given in [1,10–12].

In this study, we propose a kernelized correlation filter-based tracking scheme to enhance the tracking efficiency in difficult scenarios. Our method detects the occlusion by considering multiple cues from the correlation response maps. Furthermore, we also use these cues to handle the occlusion. Adaptive learning rate strategy based on average peak correlation energy (APCE) is incorporated in the proposed tracking scheme to prevent the corrupted model, which ultimately handles the drift problem. Furthermore, this APCE is used in the scale search strategy to handle the scale variations in the incoming video frames.

The further organization of the paper is as follows. Section 2 describes the closely related work to the proposed methodology. Section 3 explains the proposed methodology. Section 4 consists of the experimental setup for the proposed methodology. Furthermore, this section also explains the performance measures used to evaluate the tracker, whereas Section 5 contains the analysis of results. At last, Section 6 concludes the study.

2. Related Work

Many tracking schemes have been proposed to address the challenges such as deformation, occlusion, scale variation, illumination changes, etc. [13–15], but correlation filter-based tracking is still a better choice because of its efficiency and less computation cost [5].

A correlation filter (CF) generates the 2-D response map of the region of interest. Maps having higher correlation values are most likely to contain the target of interest. The initial correlation filter-based tracking proposed in [16] became very famous. This tracker is based on the minimum output sum of squared error (MOSSE). It contributed to providing an adaptive online training strategy for appearance changes in the target, as there is always a tradeoff between performance and computational cost. The requirement of a large number of samples for training makes the correlation filter tracking method computationally expensive. A novel method called kernelized correlation filter (KCF) was proposed in [17] to address this issue. In this method, all the input image/patch circularly shifted versions are used as training data. Interestingly, a single image is quiet enough to provide dense sampling to train the model in this method. Furthermore, the data matrix becomes circulant when we take the cyclically shifted versions as samples. Using the kernel trick over this data significantly decreases the computational cost. This method also gained the research community's attention because of its exceptional performance in terms of

accuracy and computational cost. In subsequent years, many improvements in KCF were proposed to address the challenging issues associated with real-time videos.

In [18], the author proposed adaptive multi-scale correlation filter-based tracking to address the scale variation problem, which exists in the original KCF scheme. Another variant of the KCF-based tracker was presented in [19] to address the partial occlusion problem. This tracker uses multiple correlation filters for different parts of the object. Long-term correlation tracking was proposed in [20,21] to address the target re-detection issue. These trackers use two different correlation filters for translation and scale estimation. They also handle the occlusion by redetecting the target after disappearance using the support vector machine (SVM) classifier. Recently, a kernelized correlation filter-based tracking scheme for large-scale variation was proposed in [22]. This tracker uses a part-based scheme and divides the target into four parts. A motion-aware correlation filter tracking scheme is presented in [23]. The author tried to incorporate the Kalman filter-based prediction algorithm in the discriminative correlation filter tracking method to prevent the model drift during challenging scenarios.

Scale-invariant feature transform is proposed in [6]. It uses average peak correlation energy to update the scale of the target model. Due to wrong scale estimation, trackers start drifting from the actual target, and tracking failure occurs. Recent literature shows that researchers are continuously trying to handle tracking failure and redetecting the target after failure. Notable articles relevant to tracking failure detection and avoidance occlusion handling are presented in [5,24] and [25,26], respectively. Discriminative correlation filter trackers also suffer from boundary effects. To address this issue, a spatially regularized correlation filter-based tracking approach (SRDCF) is presented in [27]. This approach shows promising results but at the cost of excess computational time, as it uses more images for training. In order to decrease the computational cost while keeping the promising performance, a spatial–temporal regularized correlation filter tracking scheme (STRCF) is proposed in [28]. Another variant of SRDCF [28] with multiple kernels is presented in [29]. Collaboration of fractional Kalman with KCF is presented in [30]. Similarly, a feature-based detector module in collaboration with the KCF tracker is proposed in [31]. Researchers also applied KCF-based tracking schemes to non-RGB images. A KCF-based tracking scheme for infrared images is presented in [32].

Despite a lot of successful research on discriminative correlation filter tracking, these trackers still need improvement to enhance their robustness under challenging scenarios. This study proposes a tracking scheme based on the kernelized correlation filter (KCF) method, which performs favorably under challenging scenarios. Our main contributions are listed below.

I. A design of an occlusion detection module based on the hybridization of average peak correlation energy (APCE) and confidence of squared response map (CSRM) is presented in this study.
II. It is shown that the peak correlation score alone is not good enough to detect heavy occlusion, motion blur, scale variation, background clutter, out-of-plane rotation, and deformation. We computed multiple cues from a single response map, including peak correlation, average peak correlation energy, peak-to-sidelobe ratio, and the confidence of the squared response map. Each cue gives different insights about the target of interest, which in turn helps in accurate occlusion detection and recovery of the target. Furthermore, an efficient scale handling strategy based on multiple cues for the state-of-the-art algorithm kernelized correlation filter is also presented in this study.
III. To prevent the target model from being perverted, we adjusted the learning rate as per the value of CSRM, i.e., we update the target model with a high learning rate when the CSRM is high and with a low learning rate when the CSRM value is low thus solving the drift problem in tracking.

IV. Comprehensive evaluation and analysis of proposed algorithms with state-of-the-art methods on accepted datasets i.e., OTB-50 [33], OTB-100 [34], TC-128 [35], is carried out.

3. The Proposed Tracking Framework

3.1. Kernelized Correlation Filter (KCF)

The tracking algorithm presented in [1] builds on the MOSSE filter concept [2] by extending the filter to non-linear correlation. Linear correlation between a CF template and a test image is the inner product of the template w with a test sample z for every possible shift of the test sample z. Instead of computing the linear kernel function $w^T z$ at every shift of z, KCF computes some non-linear kernel $\kappa(w, z) = \varphi^T(w)\varphi(z)$ where κ represents a kernel function that is equivalent to mapping w and z into a non-linear space with the lifting function $\phi(\cdot)$.

In one sense, KCF can be viewed as a change away from linear correlation filters, but it can also be seen as an efficient way of solving and testing with kernel ridge regression when the training and testing data is structured in a particular way (i.e., a circulant matrix).

KCF module is presented at the top left corner of Figure 1. Henriques et al. derive KCF from the standard solution of kernelized ridge regression. For learning w, we assume the training data $X = [x_0.x_1, \ldots, x_{d-1}]$ is a $d \times d$ matrix where x_k contains the same elements as x_0 shifted by k elements. The solution to kernelized ridge regression is given by [3] as per Equation (1).

$$\alpha = (\boldsymbol{K} + \lambda \boldsymbol{I})^{-1} g, \tag{1}$$

where $\boldsymbol{K}$ is the kernel matrix such that $\boldsymbol{K}_{ij} = k(x_i, x_j)$; $\boldsymbol{I}$ is the identity matrix; λ is the regularization parameter; g is the desired correlation output; and α is the dual-space coefficient vector. The dual-space coefficients allow us to rewrite the original template w in high-dimensional dual space as given in Equation (2).

$$w = \sum_{i=1}^{N} \alpha_i \varphi(x_i) \tag{2}$$

where in terms of the dot product, the kernel function $\varphi^T(x)\varphi(x') = \kappa(x, x')$ treats all data elements equally, and kernel K and the coefficients α can be computed efficiently in the Fourier domain as follows:

$$\hat{\alpha}^* = \frac{\hat{g}}{\hat{k}^{xx'} + \lambda'}, \tag{3}$$

where $\hat{k}^{xx'}$ represents the first row of the kernel matrix $\boldsymbol{K}$, which contains the kernel function computation of x_0 with all possible shifts of another data sample denoted x': either x_0 in the training phase, or some test sample z in the testing phase, and where hat ˆ denotes the DFT of the vector.

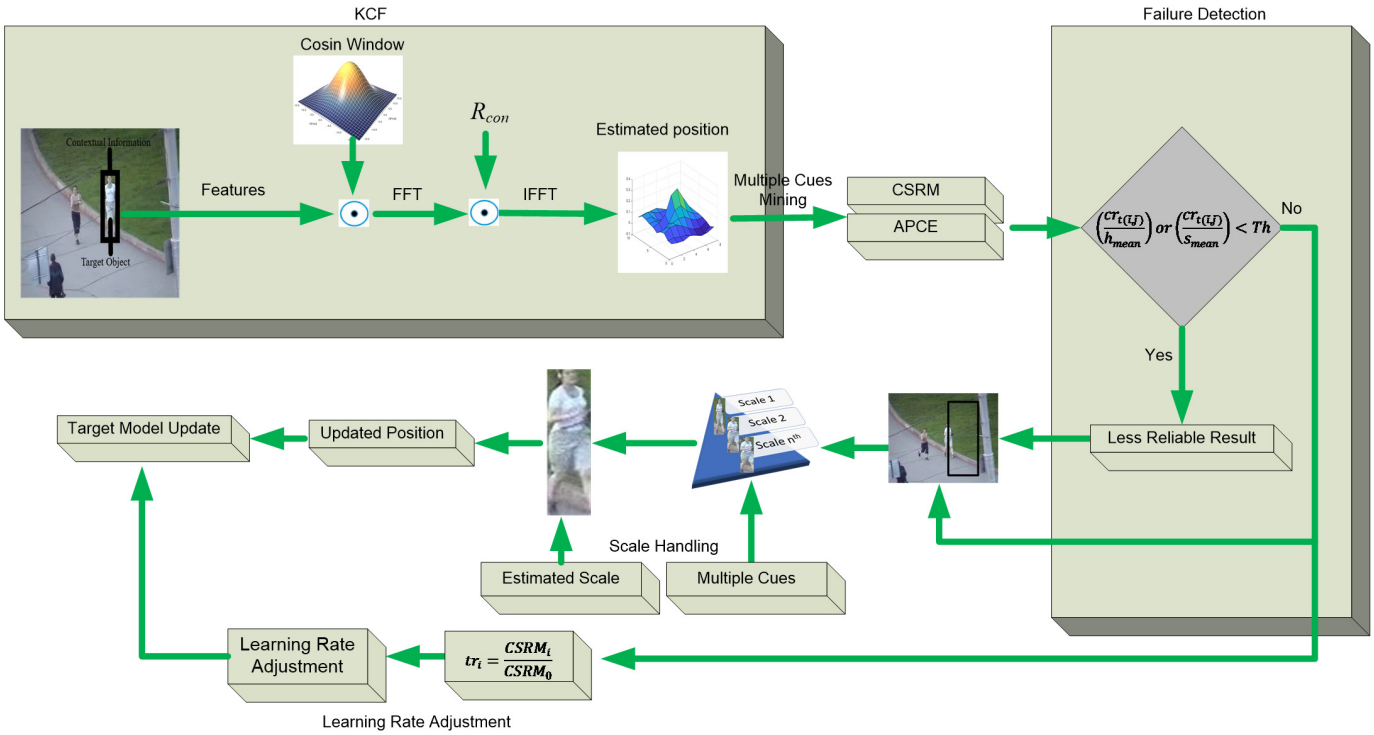


Figure 1. Graphical abstract of proposed tracking scheme. The baseline tracker is shown in the rectangle at the top left corner. The tracking failure detection (in case of occurrence of occlusion or any other issue in a frame) module is shown in the left-most rectangle, whereas the adaptive learning rate strategy is shown at the bottom of the figure. Scale handling mechanism based on multiple cues is shown below the KCF module. Furthermore, multiple cues from the response map are fed the failure detection module, and the learning rate is adjusted accordingly.

3.2. Occlusion Handling Mechanism

The correlation response map gives multiple cues about the target in visual object tracking. For example, it contains the single distinguished peak in the case of simple sequences, while, in challenging sequences, like a blur in a video sequence or/and occlusion, the map contains multiple peaks nearly equal in height, i.e., peak value decreases whereas its adjacent values increase. Hence, target tracking failure can be predicted using this cue from the response map.

Consider the response map $h_{t(p,q)}$, of size $m \times n$, for $p = 0, 1, 2 \ldots n-1$, $q = 0, 1, 2 \ldots m-1$, at t_{th} frame. The average correlation value of the 5×5 surrounding region around (i, j) is given by Equation (4).

$$S_{t(i,j)} = \frac{1}{24}\left(\left(\sum_{p=i-2}^{i+2}\sum_{q=j-2}^{j+2} h_{t(p,q)}\right) - h_{t(i,j)}\right), \tag{4}$$

APCE tells about the degree of fluctuation of the response map. If the object undergoes fast motion, the value of *APCE* will be low. *APCE* is calculated using Equation (5).

$$APCE = \frac{|h_{max} - h_{min}|^2}{mean\left(\sum_{r,c}(h_{r,c} - h_{min})\right)}, \tag{5}$$

where, h_{max} and h_{min} denote the maximum and minimum values of the response map, respectively. $h_{r,c}$ denotes the r_{th} row and c_{th} column element of response map. Now from the response map $(\bar{i}, \bar{j})$ is the coordinate where the *APCE* value is highest, as per Equation (6).

$$(\bar{i}, \bar{j}) = \underset{t_{i,j}}{arg\ max} APCE_{t_{i,j}}, \tag{6}$$

Coordinates with the highest *APCE* are obtained by searching for the response map. Different from [5], which take the coordinates of peak correlation for further development, we use the coordinates with the highest APCE value, calculating the average and peak correlation values as $h_{t(i,j)}$ and $s_{t(i,j)}$, respectively. By taking the mean over previous z frames, we can write it in the form of Equation (7).

$$h_{mean} = \frac{1}{z} \sum_{k=t-z+1}^{t} h_{k(\bar{i},\bar{j})}, \tag{7}$$

whereas the mean of the surrounding region over previous z frames is given by Equation (8).

$$s_{mean} = \frac{1}{z} \sum_{k=t-Z+1}^{t} s_{k(\bar{i},\bar{j})}, \tag{8}$$

These two values give insight into the tracking failure, i.e., if there is a distinct gap between the peak value and surrounding peaks, this means that tracking is correct, whereas, if there is a sharp drop in peak value and an increase in surrounding peaks simultaneously, this shows that it is difficult for the tracking algorithm to find the exact target, and most probably, tracking failure will occur. Mathematically, it is shown by a conditional expression, as given in Equation (9).

$$\left(\frac{cr_{t(\bar{i},\bar{j})}}{h_{mean}}\right) or \left(\frac{cr_{t(\bar{i},\bar{j})}}{s_{mean}}\right) < Th, \tag{9}$$

where Th is set to be 0.6 [4].

3.3. Adaptive Scale Handling Mechanism

A multi-resolution translation filter scheme is implemented for scale handling. Most algorithms, such as SAMF, use Maximum response value for scale searching. In turn, this degrades the performance of the overall tracking scheme when the video sequence contains one or more challenging factors, such as scale variation, occlusion, motion blur, etc. [5]. We use Equation (5) along with Equation (10) for scale handling. Thus, incorporating multiple cues to address the issue more effectively, i.e., for any scaled sample if $CSRM$ & $APCE > Th$, only that scale is selected as the true scale of the target.

The fluctuation and peak value of the response map define the tracking reliability, i.e., the ideal response map contains the one sharp peak at the location of the target of interest and is nearly equal to zero at other locations. A sharper peak as compared to other values of the response map ensures higher localization accuracy. On the other side, when the video contains several challenging factors such as occlusion, motion blur, and scale variation, response map values will start fluctuating, and the APCE measure will decrease. Pictorial representation of scale handling strategy is shown in Figure 2.

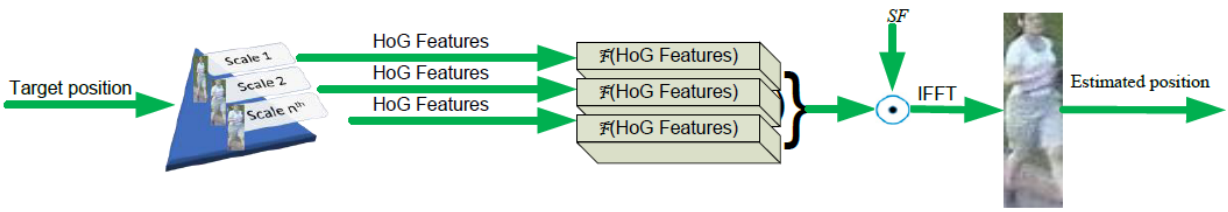


Figure 2. Scale handling mechanism. Multiple sub-windows around the estimated location are sampled. These windows are obtained by multiplying the previous target window with different scale factors. Sub window with highest APCE and CSRM value is considered as correct scale estimation of object.

We trained a simple two-dimensional KCF filter for translation estimation. Instead of using a naïve maximum response value, we use a robust APCE measure to find the true object position. The correlation response map with the highest APCE using Equation (6) is considered to be true object location. Then the multiple sub-windows around the estimated

location are sampled. These windows are obtained by multiplying the previous target window with different scale factors. The sub-window with the highest APCE value is considered the correct scale estimation of the object. Fine-tuning is applied to the previous translation estimation after getting the exact scale of the object.

3.4. Adaptive Learning Rate

Maximum response value has been used widely as a reliability measure in tracking algorithms. During occlusion, motion blur, etc., the response map changes drastically. So, using only the maximum response value as a reliability measure to detect the occlusion is not good enough. Another measure, i.e., average peak correlation energy (APCE), is presented in [6] given Equation (5).

It has been shown practically that if the target apparently appears in the detection scope, there will be a sharper peak in the response map, and the value of APCE will be smaller. However, if the target is occluded, the peak in the response map will be smoother, and the relative value of APCE becomes larger [7]. This problem is solved by squaring the response map and then finding the confidence of the squared response map [7]. The peak of the response map is represented in the nominator of Equation (10). At the same time, the denominator represents the mean square value of the response map. It is shown in Figure 1 that the input to the adaptive learning rate block is coming from the response map, i.e., we are adjusting the learning rate by fetching multiple cues from the response map. The confidence of squared response map is given by Equation (10).

$$CSRM = \frac{\left|{R_{max}}^2 - {R_{min}}^2\right|^2}{\frac{1}{MN}\sum_{r=1}^{M}\sum_{c=1}^{N}\left|{R_{r,c}}^2 - {R_{min}}^2\right|^2}, \tag{10}$$

where R_{max} and R_{min} denote the maximum and minimum values of the response map, respectively. $R_{r,c}$ denotes the r_{th} row and c_{th} column element of the response map. $M{*}N$ is the dimension of the response map.

We increased the robustness of the reliability measure by considering both APCE and CSRM. First, we calculated the response using the robust APCE measure different from the MKCF [5] algorithm, which selects the response using a naïve maximum correlation value. After selecting the response with correct scale estimation, CSRM is employed to adjust the learning rate. The conditional expression given by Equation (11) is used to adjust the learning rate.

$$\begin{cases} tr_i = \frac{CSRM_i}{CSRM_0} \\ \eta_i = \eta_0, tr_i > tr_0 \\ \eta_i = \eta_0.tr_i, others \end{cases}, \tag{11}$$

where $CSRM_i$, is the value of squared response map in ith frame while $CSRM_0$ is the values of the most confident result, i.e., the result of the first frame, η_i, is the learning rate for ith frame.

4. Experiments

The proposed tracker is evaluated using both qualitative and quantitative results. A large number of experiments are performed on selected videos. Twenty-three videos were selected from three standard datasets, i.e., OTB-50 [8], OTB-100 [9], and Temple Colour-128 [10]. Visual challenges such as occlusion, out-of-plane rotation, cluttering, scale changing, deformation, fast motion and motion blur, etc. associated with these videos are presented in Table 1.

Table 1. Challenges associated with selected videos.

Sequence	OPR	IPR	OCC	LR	SV	BC	MB	IV	DEF	FM	OV
Ball_ce3					yes					yes	yes
Bike_ce1		yes			yes			yes		yes	
Boat_ce2											
Carchasing_ce1		yes	yes		yes			yes		yes	
Cardark 50						yes		yes			
Dudek	yes	yes	yes		yes				yes	yes	yes
Electricalbike_ce			yes		yes						
Guitar_ce2	yes	yes						yes		yes	
Gym 100, 128	yes	yes			yes				yes		
Hurdle_ce1							yes		yes	yes	
Man 100								yes			
Mhyang 100	yes					yes		yes	yes		
Michealjakson_ce	yes	yes						yes	yes	yes	
Motorbike_ce 128			yes			yes		yes			
Mountainbike 100	yes	yes				yes					
Railwatstation_ce		yes	yes			yes					
Redteam	yes	yes	yes	yes	yes						
Subway 100			yes			yes			yes		
Suitcase_ce			yes			yes		yes			
Sunshade 128								yes			
Suv 100		yes	yes								yes
Tiger1 100	yes	yes	yes				yes	yes	yes	yes	
Trellis 50	yes	yes			yes	yes		yes			

Evaluation Criteria

A comprehensive comparison of the proposed tracker with other latest state-of-the-art algorithms is based on three evaluation criterions, i.e., distance precision (DP), mean center location error (CLE), and overlap success rate (OSR) [11], is presented in this paper. Distance precision is defined as the distance in terms of pixels between the ground truth and estimated position. As a standard practice, it is calculated at a threshold of 20 pixels, whereas *CLE* is defined as the Euclidean distance calculated between the tracker and the ground truth of the target. Mathematically, *CLE* is given by Equation (12).

$$CLE = \sqrt{(x_i - x\ )^2 + (y_i - y\ )^2}, \tag{12}$$

where $(x_{i,}, y_i)$ are positions calculated by tracking algorithm, and (x, y) are ground truth values. The overlap success rate is defined as the area between the ground truth box and the estimated position box. Equation (13) shows the overlapping area between two boxes.

$$AuC = \frac{area\left(A_e \cap A_g\right)}{area\left(A_e \cup A_g\right)}, \tag{13}$$

where A_e is the area of the estimated bounding box, and A_g is the area of the ground truth bounding box. The numerator of Equation (13) is the intersection of two areas, whereas the denominator is the union of two bounding boxes. This overlap success rate is calculated at a threshold of 0.5. The number of frames having an overlap area greater than the threshold of 0.5 divided by the total number of frames gives an overlap success rate.

The proposed method is implemented in MATLAB (2019) on an Intel Core i7, 7th generation, 2.80 GHz processor, RAM 16 GB, a machine with a 64-bit Windows 10 operating system.

5. Results

Our proposed tracking scheme is compared and evaluated on a number of videos from three different benchmark datasets, i.e., OTB-50 [32], OTB100 [33], and TC-128 [34].

OTB 50 contains 50 videos. OTB-100 [33] contains 100 different challenging videos. Each video has one or more visual challenges, such as clutter, deformation, out-of-plane rotation, occlusion, motion blur, etc., associated with it. TC-128 [34] contains 128 challenging videos. Out of these 128 videos, 78 videos are new, and others are repeated in other datasets. We have selected 23 mixed sequences from these three datasets. Selected videos have eleven challenging attributes, namely (i) occlusion, (ii) scale variation, (iii) motion blur, (iv) fast motion, (v) out-of-plane rotation, (vi) deformation, (vii) background, (viii) in-plane rotation, (ix) intensity variation, (x) low resolution, and (xi) out-of-view movement to support and evaluate our proposed tracker. An explanation of each attribute is given in Table 2.

Table 2. Explanation of challenges associated with video sequences.

Attribute Name	Abbreviation	Explanation
Occlusion	Occ	Target is hidden behind another object
Scale variation	SV	Bounding boxes ratio of initial frame and present frame is out of range
Low resolution	LR	When the resolution becomes lower in subsequent frames
Out-of-plane rotation	OPR	Rotation of target object out of image plane
Motion blur	MB	Blurring of target region
Intensity variation	IV	Change in intensity
Fast motion	FM	Ground truth motion is greater than 20 pixels
Background clutter	BC	Target object background having similar color or texture as that of target
In-plane-rotation	IPR	Rotation of the object in the plane of image
Out-of-view movement	OV	Movement of out of the view
Deformation	DEF	Non-rigid object deformation

5.1. Quantitative Analysis

To evaluate the performance of the proposed tracker quantitatively, three performance measures were used, i.e., distance precision, overlap threshold, and center error location. Comparison based on distance precision is given in Table 3. A center location error comparison is given in Table 4, whereas an overlap success rate comparison is given in Table 5. Let us discuss the performance of the proposed tracker in comparison with other selected state-of-the-art algorithms based on each performance measure. Table 3 shows the mean distance precision at the threshold of 20 pixels of the proposed method, LCT [21], MACF [23], MKCF [5], and STRCF [28]. The proposed tracking scheme outperforms the other state-of-the-art algorithms by achieving the highest mean of 0.889. The second-best in terms of distance precision is MKCF [5], with a mean value of 0.855, whereas the third best is MACF [23], having a mean value of 0.804. A complete distance precision plot for each video is also given in Figure 3. The three most complex challenges, i.e., out-of-view movement, scale variation, and fast motion, are associated with Ball_ce3 video, and our proposed tracker achieved the highest distance precision value of 0.93 on this video sequence. A similarly proposed tracker also shows better performance for an on-suite case and gym 1 video sequences. It can be seen from Figure 3 that videos have fewer associated challenges; all the tackers have equal performance in terms of distance precision.

This paragraph explains the overlap success rate comparison. This comparison is given in Table 5. The last row shows the mean overlap success rate of 23 selected video sequences. Our proposed tracker achieved the highest mean overlap success rate of 0.824. In contrast, MACF [23] is second-best with a small difference of 0.002, while the remaining three trackers, i.e., STRCF [28], LCT [21], and MKCF [5], have a mean overlap success rate of 0.799, 0.717, and 0.645, respectively. It is noted that video sequences charchasing_ce1, Dudek, and electricalbike_ce contain severe occlusions. Our proposed tracker showed considerable performance on these sequences as per Table 5.

Table 3. Distance precision of five trackers for twenty-three video sequences at threshold of 20 pixels.

Sequence	Our Method	LCT	MACF	MKCF	STRCF
Ball_ce3	0.930	0.590	0.586	0.626	0.570
Gym1	0.966	0.940	0.955	0.953	0.940
Microbike_ce	0.998	0.238	0.238	0.966	0.238
Suitcase_ce	0.908	0.793	0.847	0.900	0.391
Railwatstation_ce	0.814	0.036	0.036	0.816	0.136
Bike_ce1	1.000	1.000	1.000	1.000	1.000
Boat_ce2	0.700	0.697	0.740	0.672	0.700
Carchasing_ce1	0.764	0.289	0.285	0.890	0.283
Cardark	1.000	1.000	1.000	1.000	1.000
Dudek	0.852	0.905	0.848	0.870	0.870
Electricalbike_ce	1.000	1.000	1.000	1.000	1.000
Guitar_ce2	0.505	0.000	0.524	0.505	0.543
Tiger1	0.780	0.890	0.974	0.147	0.990
Hurdle_ce1	0.710	0.700	0.983	0.720	0.967
Man	1.000	1.000	1.000	1.000	1.000
Mhyang	1.000	1.000	1.000	1.000	1.000
Michealjakson_ce	0.537	0.455	0.496	0.618	0.430
Mountainbike	1.000	0.996	1.000	1.000	0.978
Redteam	1.000	1.000	1.000	1.000	1.000
Subway	1.000	1.000	1.000	1.000	1.000
Sunshade	1.000	1.000	1.000	1.000	1.000
Suv	0.979	0.980	0.978	0.978	0.970
Trellis	1.000	1.000	1.000	1.000	1.000
Mean	0.889	0.761	0.804	0.855	0.784

Table 4. Center location error of proposed method, LCT, MACF, MKCF, and STRCF.

Sequence	Our Method	LCT	MACF	MKCF	STRCF
Ball_ce3	9.1665	95.2300	94.9000	71.0300	94.0000
Microbike_ce	6.2500	377.0000	253.6500	354.0000	203.0000
Michealjakson_ce	38.9436	180.0000	24.0500	351.4100	35.3300
Suitcase_ce	7.2129	32.0000	16.900	7.2300	77.3700
Sunshade	4.5964	4.5800	4.1900	4.5400	4.2800
Mhyang	2.6102	4.1200	2.3600	3.9200	2.3700
Railwatstation_ce	12.4448	328.4100	654.8900	12.4500	414.0000
Trellis	2.6922	8.7700	2.8600	7.7600	2.5000
Cardark	2.8163	2.9100	1.8300	6.0500	1.1300
Gym1	8.5791	8.1200	9.1300	7.8000	7.5100
Dudek	11.5036	15.00	10.4400	193.0000	10.9100
Bike_ce1	4.2268	4.9300	3.8600	4.1700	3.6600
Boat_ce2	42.5323	44.7300	41.4000	27.3800	45.4300
Carchasing_ce1	43.6496	60.6900	63.4700	9.4600	73.2600
Electricalbike_ce	4.8738	5.2800	5.5500	4.8000	4.6900
Guitar_ce2	58.8593	399.9100	19.0600	387.3600	18.9600
Hurdle_ce1	62.6116	23.2300	5.7200	98.6900	6.2500
Mountainbike	8.1366	9.3700	8.1000	7.6600	10.4200
Tiger1	24.3871	10.3500	7.2800	194.5800	7.2100
Redteam	3.0600	4.3800	2.7000	3.8100	2.0200
Subway	3.01400	3.1500	3.2800	2.9700	2.6500
Man	2.6466	2.2000	1.7300	2.2600	1.3100
Suv	4.6600	3.9700	3.3400	3.6500	4.1300
Mean	16.0600	70.7970	53.9400	76.7820	44.8900

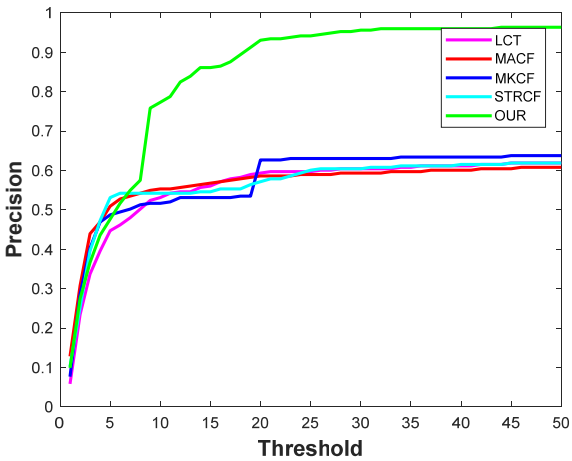
Table 5. Overlap success rate comparison of proposed tracking scheme with four other tracking algorithms.

Sequence	Our Method	LCT	MACF	MKCF	STRCF
Ball_ce3	0.696	0.530	0.553	0.520	0.540
Cardark	1.000	0.990	1.000	0.690	1.000
Microbike_ce	1.000	0.140	0.238	0.020	0.240
Railwatstation_ce	0.800	0.030	0.033	0.800	0.130
Dudek	0.9712	0.880	1.000	0.060	0.970
Mhyang	1.000	0.990	1.000	1.000	1.000
Man	1.000	1.000	1.000	1.000	1.000
Carchasing_ce1	0.710	0.290	0.283	0.730	0.280
Suitcase_ce	0.875	0.780	0.782	0.880	0.390
Subway	1.000	1.000	1.000	1.000	1.000
Electricalbike_ce	0.980	0.990	1.000	0.970	1.000
Guitar_ce2	0.524	0.000	0.980	0.030	0.970
Gym1	0.796	0.850	0.738	0.810	0.880
Hurdle_ce1	0.687	0.680	0.830	0.690	0.870
Michealjakson_ce	0.667	0.250	0.908	0.080	0.550
Bike_ce1	0.783	1.000	1.000	0.780	1.000
Mountainbike	0.990	0.990	1.000	0.990	0.950
Boat_ce2	0.517	0.610	0.660	0.460	0.630
Redteam	0.400	0.700	0.982	0.380	1.000
Sunshade	0.980	0.970	0.990	0.980	0.990
Suv	0.980	0.980	0.985	0.980	0.990
Tiger1	0.790	0.930	0.985	0.150	1.000
Trellis	0.838	0.920	0.964	0.840	0.990
Mean	0.824	0.717	0.822	0.645	0.799

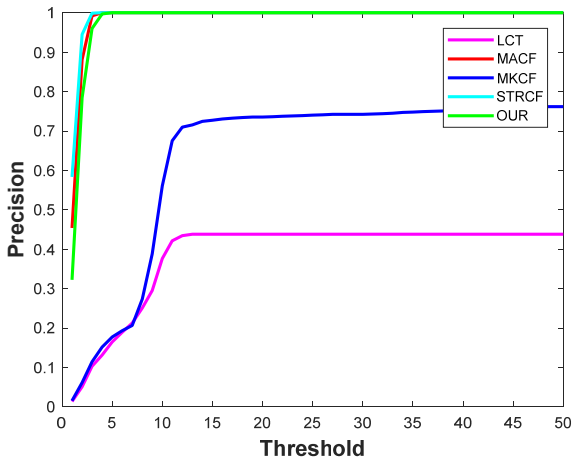
This paragraph will discuss the center location error comparison of the proposed tracker with the other four state-of-the-art algorithms. The last row of Table 4 shows the mean center location error of the proposed tracker, LCT [21], MACF [23], MKCF [5], and STRCF [28]. Our proposed tracker outperformed by achieving a mean error of 16.06. Furthermore, there is a huge gap between the second-best and our proposed tracker. The STRCF [28] scheme achieved the mean center location error of 44.89, whereas the third-best, i.e., MACF [23], achieved the mean error of 53.94 pixels. LCT [21] and MKCF [5] showed similar performance in terms of center location error by achieving 70.797 and 79.782 values, respectively. The Center location error plot of each video is also given in Figure 4. To avoid the overcrowding of plots in the paper, only six videos were selected for plots, but the error of each of the 23 videos is available in Table 4.

5.2. Qualitative Analysis

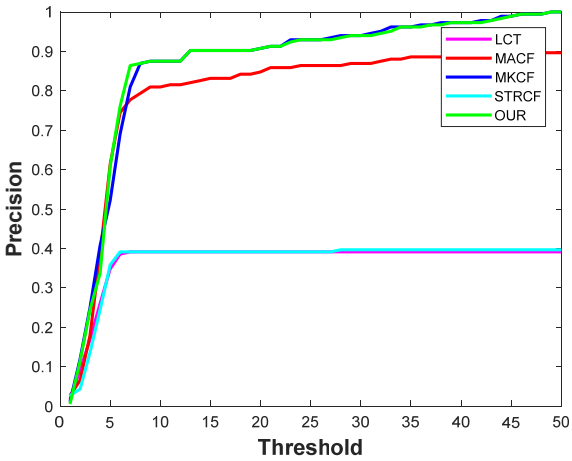
To evaluate and support our proposed tracker, the qualitative analysis is given in this section. For the qualitative analysis, the results of five trackers, i.e., our proposed, MKCF [5], MACF [23], STRCF [28], and LCT [21], over five video sequences are presented in Figure 5. Three frames of each video are shown in Figure 5. The top to bottom rows of Figure 5 contains the ball_ce3, car1, microbike_ce, railwaystation_ce, and suitcase_ce video sequences. In the first row of Figure 5, all the trackers successfully track the target until frame number 137. In frame number 174, the object of interest i.e., the ball is out of view of the tracker; hence, all the trackers have a bounding box at some wrong position. In frame number 256, the object again came back into view. In this frame, the proposed tracker is the only one to track the ball correctly. The green bounding box can be seen in the third column of the first row of Figure 5.



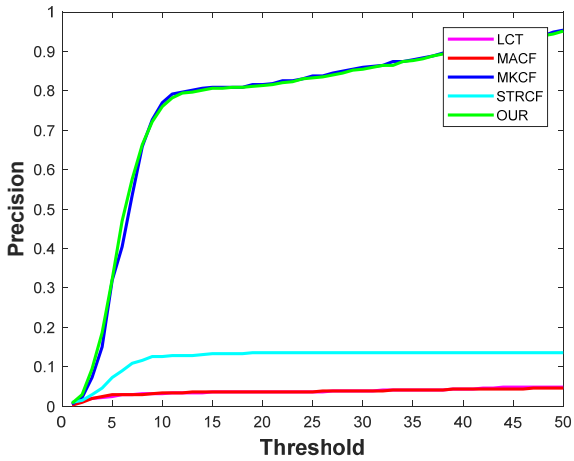
(**a**) Ball_ce3



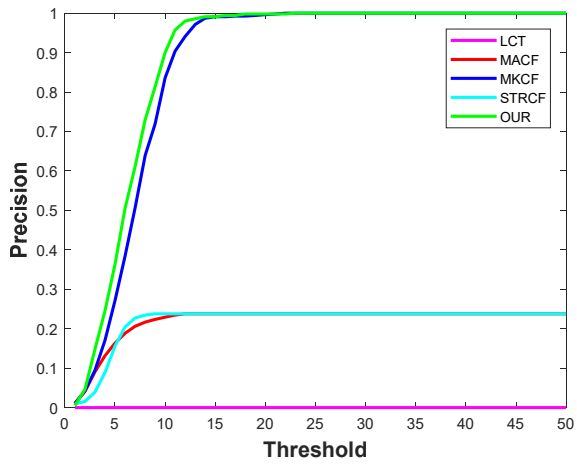
(**b**) Car1



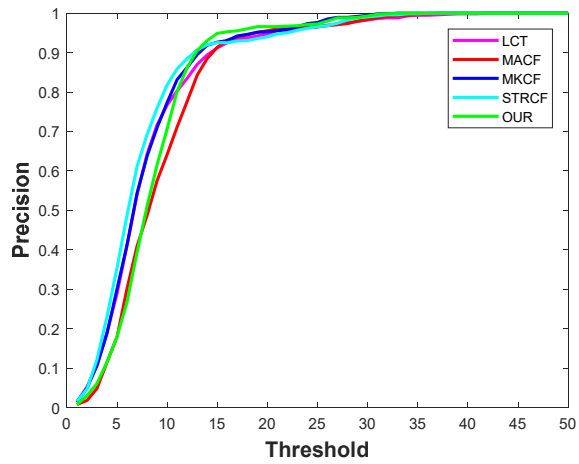
(**c**) Suitcase_ce



(**d**) Railwaystation_ce

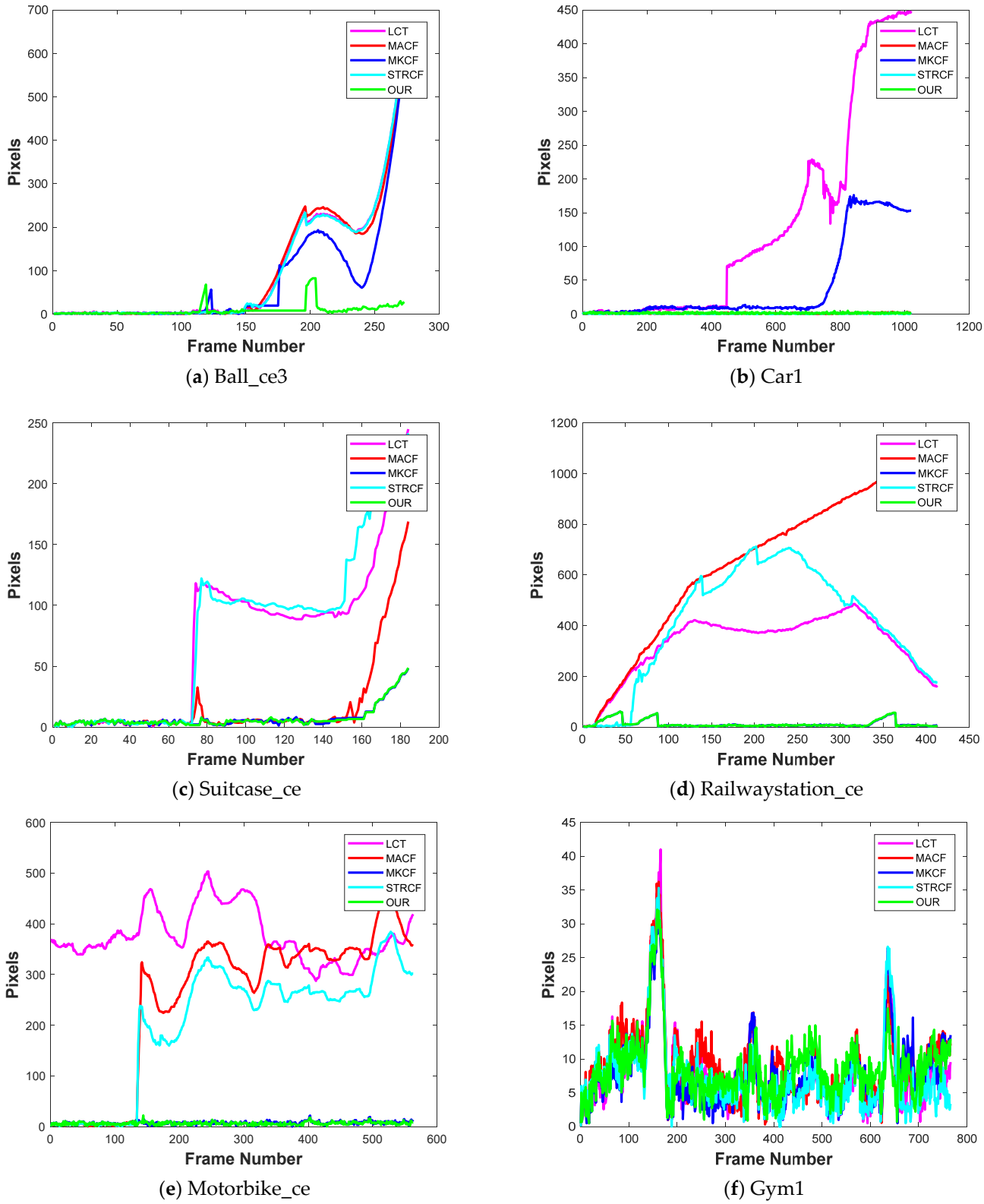


(**e**) Motorbike_ce



(**f**) Gym1

Figure 3. Quantitative analysis: comparison based on distance precision over a threshold of 20 pixels. Six videos selected from OTB-50, OTB-100, and Colour-128 datasets.



(**a**) Ball_ce3 (**b**) Car1

(**c**) Suitcase_ce (**d**) Railwaystation_ce

(**e**) Motorbike_ce (**f**) Gym1

Figure 4. Quantitative analysis: comparison based on center location error. Six videos are selected from OTB-50, OTB-100, and Colour-128 datasets.

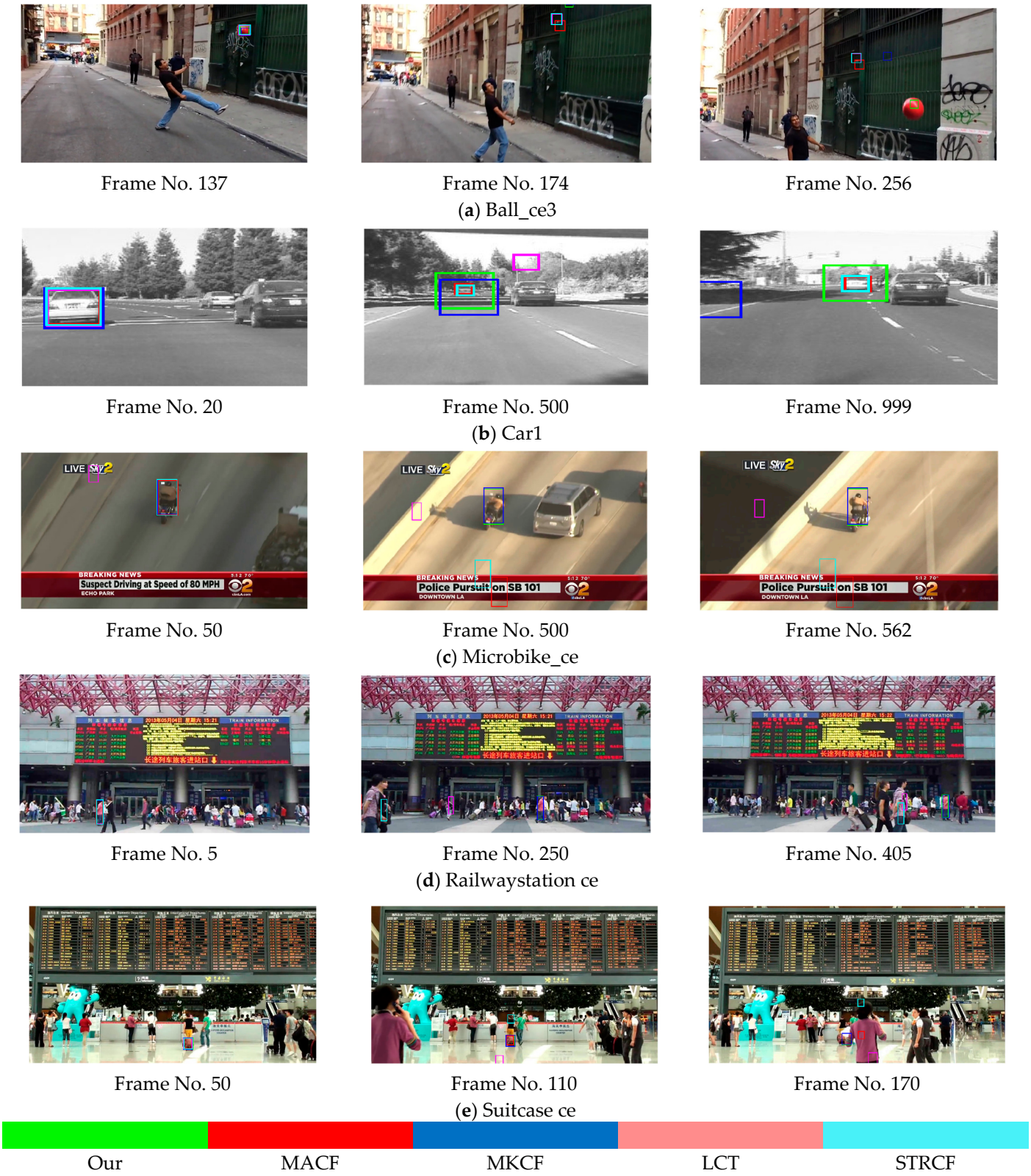


Figure 5. Qualitative analysis over six selected videos from OTB-50, OTB-100, and Colour-128 datasets.

This is a very challenging issue, for the object-tracking community to track an object when it comes back into view after out-of-view movement. Our proposed tracker successfully handled this situation. In the second row of Figure 5, the car1 sequence is shown. All

the trackers successfully track the target up until frame number 20. LCT [21] lost the target in frame number 500, whereas MKCF [5] lost the target in frame number 999.

Our proposed tracker, along with MACF [23] and STRCF [28], tracks the target successfully until the end of the video sequence. The third row of Figure 5 represents the microbike_ce video sequence.

In this sequence, LCT [21] fails at the start of the video sequence, which can be seen in frame number 50. While STRCF [28] and MACF [23] fail to track the object in frame number 500, our proposed tracker tracks the target correctly until the end of the video sequence. The fourth row of Figure 5 shows the railwaystation_ce video sequence. This sequence contains the clutter background, in-plane rotation, and occlusion. Our proposed tracker successfully handles the challenges associated with this video sequence and tracks the object successfully, as shown in frames number 5250 and 405. MKCF [5] also shows similar performance on this sequence, whereas all other tracker schemes fail to track the object.

The last row shows the suitcase_ce sequence from the Colour-128 dataset. The object to track in this sequence is a suitcase held in the hand of the girl. This sequence contains the clutter background, intensity variation, and occlusion. In this sequence, again the proposed tracker achieves a better result by tracking the target successfully even after the occlusion. MKCF [5] showed a similar performance to our proposed algorithm on this sequence, whereas MACF [23], STRCF [28], and LCT [21] were unable to track the correct object.

6. Conclusions

Most of the tracking algorithms use a single cue fetched from the response map for the training and detection phase of the filter. Like other tracking methods, our baseline tracker KCF also uses a single cue from the response map, such as peak correlation or peak to sidelobe ratio. This single cue could not give much insight into the tracking result, which causes the algorithm to suffer in challenging scenarios such as scale variation, occlusion, illumination variation, and motion blur. Similarly, simple KCF cannot detect the reliability of tracking results, which causes a drift problem. In the proposed tracking methodology, different cues such as average peak correlation energy, the confidence of squared response map, peak correlation value, and, last but not least, novel differences of peak correlation from single response map are used to handle the challenging issues of video sequences.

A comparison of the proposed scheme with four other state-of-the-art algorithms is presented. For a fair comparison, twenty-three different video sequences are selected from three standard visual object tracking datasets, i.e., OTB-50, OTB-100, and TC-128. The proposed tracking scheme shows favorable quantitative as well as qualitative results. For all three performance measures, our method achieved the highest accuracy.

Response map computation is a mandatory step for correlation filter-based tracking schemes. Tracking efficiency may be further enhanced without a significant increase in computation cost with the help of multiple cues mined from the response map. Furthermore, multiple cues also give better insight into the target/tracking result.

Author Contributions: Conceptualization, B.K., A.J. and A.A.; methodology, B.K., A.J. and A.A.; software, B.K., K.M. and K.M.C.; validation, K.M., K.M.C. and M.M.; formal analysis, B.K., K.M. and K.M.C.; investigation, B.K., K.M.C. and M.M.; resources, K.A., A.M.E.-S. and K.M.C.; writing—original draft preparation, B.K. and K.M.C.; writing—review and editing, K.M. and M.M.; visualization, H.T., A.J., A.A. and K.M.C.; supervision, A.J. and A.A.; project administration, K.A., A.M.E.-S., K.M.C. and H.T.; funding acquisition, K.A., A.M.E.-S., H.T. and K.M.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors extend their appreciation to King Saud University for funding this work through the Researcher Support Project number (RSP-2021/133), King Saud University, Riyadh, Saudi Arabia.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Henriques, J.F.; Caseiro, R.; Martins, P.; Batista, J. Exploiting the circulant structure of tracking-by-detection with kernels. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7575 LNCS, pp. 702–715. [CrossRef]
2. Kim, Y.; Park, H.; Paik, J. Robust Kernelized Correlation Filter using Adaptive Feature Weight TT. *IEIE Trans. Smart Process. Comput.* **2018**, *7*, 433–439. [CrossRef]
3. Chen, K.; Tao, W. Once for All: A Two-Flow Convolutional Neural Network for Visual Tracking. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *28*, 3377–3386. [CrossRef]
4. Hadfield, S.J.; Lebeda, K.; Bowden, R. The visual object tracking VOT2014 challenge results. In Proceedings of the European Conference on Computer Vision (ECCV) Visual Object Tracking Challenge Workshop, Zurich, Switzerland, 6 September 2014.
5. Shin, J.; Kim, H.; Kim, D.; Paik, J. Fast and Robust Object Tracking Using Tracking Failure Detection in Kernelized Correlation Filter. *Appl. Sci.* **2020**, *10*, 713. [CrossRef]
6. Ma, H.; Acton, S.T.; Lin, Z. SITUP: Scale Invariant Tracking Using Average Peak-to-Correlation Energy. *IEEE Trans. Image Process.* **2020**, *29*, 3546–3557. [CrossRef]
7. Ross, D.A.; Lim, J.; Lin, R.-S.; Yang, M.-H. Incremental Learning for Robust Visual Tracking. *Int. J. Comput. Vis.* **2008**, *77*, 125–141. [CrossRef]
8. Zhou, S.K.; Chellappa, R.; Moghaddam, B. Visual tracking and recognition using appearance-adaptive models in particle filters. *IEEE Trans. Image Process.* **2004**, *13*, 1491–1506. [CrossRef]
9. Mei, X.; Ling, H. Robust visual tracking using ℓ1minimization. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 1436–1443. [CrossRef]
10. Possegger, H.; Mauthner, T.; Bischof, H. In defense of color-based model-free tracking. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 2113–2120. [CrossRef]
11. Hare, S.; Saffari, A.; Torr, P.H.S. Struck: Structured output tracking with kernels. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 263–270. [CrossRef]
12. Tang, M.; Yu, B.; Zhang, F.; Wang, J. High-speed tracking with multi-kernel correlation filters. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018. [CrossRef]
13. Babenko, B.; Yang, M.H.; Belongie, S. Robust object tracking with online multiple instance learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 1619–1632. [CrossRef]
14. Zhong, W.; Lu, H.; Yang, M.-H. Robust object tracking via sparsity-based collaborative model. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 1838–1845.
15. Zhang, T.; Jia, K.; Xu, C.; Ma, Y.; Ahuja, N. Partial occlusion handling for visual tracking via robust part matching. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1258–1265.
16. Bolme, D.S.; Beveridge, J.R.; Draper, B.A.; Lui, Y.M. Visual object tracking using adaptive correlation filters. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 2544–2550. [CrossRef]
17. Henriques, J.F.; Caseiro, R.; Martins, P.; Batista, J. High-Speed Tracking with Kernelized Correlation Filters. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *37*, 583–596. [CrossRef]
18. Danelljan, M.; Hager, G.; Khan, F.S.; Felsberg, M. Discriminative Scale Space Tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1561–1575. [CrossRef]
19. Liu, T.; Wang, G.; Yang, Q. Real-time part-based visual tracking via adaptive correlation filters. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 4902–4912. [CrossRef]
20. Ma, C.; Yang, X.; Zhang, C.; Yang, M.H. Long-term correlation tracking. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5388–5396. [CrossRef]
21. Ma, C.; Huang, J.-B.; Yang, X.; Yang, M.-H. Adaptive Correlation Filters with Long-Term and Short-Term Memory for Object Tracking. *Int. J. Comput. Vis.* **2018**, *126*, 771–796. [CrossRef]
22. Lian, G. A novel real-time object tracking based on kernelized correlation filter with self-adaptive scale computation in combination with color attribution. *J. Ambient Intell. Humaniz. Comput.* **2020**, 1–9. [CrossRef]
23. Zhang, Y.; Yang, Y.; Zhou, W.; Shi, L.; Li, D. Motion-Aware Correlation Filters for Online Visual Tracking. *Sensors* **2018**, *18*, 3937. [CrossRef] [PubMed]
24. Khan, B.; Ali, A.; Jalil, A.; Mehmood, K.; Murad, M.; Awan, H. AFAM-PEC: Adaptive Failure Avoidance Tracking Mechanism Using Prediction-Estimation Collaboration. *IEEE Access* **2020**, *8*, 149077–149092. [CrossRef]
25. Mehmood, K.; Jalil, A.; Ali, A.; Khan, B.; Murad, M.; Khan, W.U.; He, Y. Context-Aware and Occlusion Handling Mechanism for Online Visual Object Tracking. *Electronics* **2020**, *10*, 43. [CrossRef]

26. Mehmood, K.; Jalil, A.; Ali, A.; Khan, B.; Murad, M.; Cheema, K.; Milyani, A. Spatio-Temporal Context, Correlation Filter and Measurement Estimation Collaboration Based Visual Object Tracking. *Sensors* **2021**, *21*, 2841. [CrossRef]
27. Gao, L.; Li, Y.; Ning, J. Improved kernelized correlation filter tracking by using spatial regularization. *J. Vis. Commun. Image Represent.* **2018**, *50*, 74–82. [CrossRef]
28. Li, F.; Tian, C.; Zuo, W.; Zhang, L.; Yang, M.-H. Learning spatial-temporal regularized correlation filters for visual tracking. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
29. Su, Z.; Li, J.; Chang, J.; Song, C.; Xiao, Y.; Wan, J. Learning spatial-temporally regularized complementary kernelized correlation filters for visual tracking. *Multimed. Tools Appl.* **2020**, *79*, 25171–25188. [CrossRef]
30. Mehmood, K.; Ali, A.; Jalil, A.; Khan, B.; Cheema, K.M.; Murad, M.; Milyani, A.H. Efficient Online Object Tracking Scheme for Challenging Scenarios. *Sensors* **2021**, *21*, 8481. [CrossRef]
31. Tseng, D.-C.; Chen, C.-H.; Chen, Y.-M. Autonomous Tracking by an Adaptable Scaled KCF Algorithm. *Int. J. Mach. Learn. Comput.* **2021**, *11*, 48–54. [CrossRef]
32. Yang, X.; Li, S.; Yu, J.; Zhang, K.; Yang, J.; Yan, J. GF-KCF: Aerial infrared target tracking algorithm based on kernel correlation filters under complex interference environment. *Infrared Phys. Technol.* **2021**, *119*, 103958. [CrossRef]
33. Wu, Y.; Lim, J.; Yang, M.-H. Online object tracking: A benchmark. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2411–2418.
34. Wu, Y.; Lim, J.; Yang, M.-H. Object Tracking Benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1834–1848. [CrossRef] [PubMed]
35. Liang, P.; Blasch, E.; Ling, H. Encoding Color Information for Visual Tracking: Algorithms and Benchmark. *IEEE Trans. Image Process.* **2015**, *24*, 5630–5644. [CrossRef] [PubMed]