Research Article

An Improved Kernelized Correlation Filter Based Visual Tracking Method

Jianjun Ni ^{1,2,3}, Xue Zhang,¹ Pengfei Shi ^{1,2} and Jinxiu Zhu^{1,3}

¹College of IOT Engineering, Hohai University, Changzhou 213022, China

²Jiangsu Universities and Colleges Key Laboratory of Special Robot Technology, Hohai University, Changzhou 213022, China

³Jiangsu Provincial Collaborative Innovation Center of World Water Valley and Water Ecological Civilization, Nanjing 210098, China

Correspondence should be addressed to Jianjun Ni; njhhuc@gmail.com

Received 8 July 2018; Revised 29 September 2018; Accepted 22 November 2018; Published 17 December 2018

Guest Editor: Ayed A. Salman

Copyright © 2018 Jianjun Ni et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Correlation filter based trackers have received great attention in the field of visual target tracking, which have shown impressive advantages in terms of accuracy, robustness, and speed. However, there are still some challenges that exist in the correlation filter based methods, such as target scale variation and occlusion. To deal with these problems, an improved kernelized correlation filter (KCF) tracker is proposed, by employing the GM(1,1) grey model, the interval template matching method, and multiblock scheme. In addition, a strict template update strategy is presented in the proposed method to accommodate the appearance change and avoid template corruption. Finally, some experiments are conducted. The proposed method is compared with the top state-of-the-art trackers, and all the tracking algorithms are evaluated on the object tracking benchmark. The experimental results demonstrate obvious improvements of the proposed KCF-based visual tracking method.

1. Introduction

Visual tracking is one of the most fundamental tasks in the field of computer vision, which has several practical applications including unmanned aerial vehicles, traffic control, human computer interaction, and video surveillance [1–4]. There are various issues that should be considered in designing a tracking system, such as real-time tracking, adaptivity, and robustness [5]. In recent years, although the visual tracking technology has progressed significantly [6], it remains challenging for long-term visual object tracking due to a mass of factors, such as occlusion, deformation, illumination variation, scale variation, and fast motion of targets [7, 8].

Various methods have been proposed to deal with the visual tracking problem. For example, Ma et al. [9] proposed a joint blur state estimation and multitask reverse sparse learning framework to solve the motion blur problem in visual tracking. Zhang et al. [10] presented a robust correlation tracking algorithm with discriminative correlation filters to handle target scale variations in complex scenes. The popular

tracking algorithms can be categorized into generative and discriminative methods [11]. The generative method focuses on learning a perfect appearance model to represent the target and then to find the most similar candidate by matching [12, 13]. Among these methods, KLT [14] and NCC [15] are the original methods. They are simple and efficient; however, they tend to drift as the search area enlarges. Different from the generative method, the discriminative method formulates the tracking problem as a binary classification task and distinguishes the target from the background by using a discriminative classifier [16]. Usually, the classifier is trained and updated by positive and negative samples. In particular, the correlation filter based discriminative method has been proven to have high efficiency and recently attracted a considerable amount of research attention [17]. But there are also a lot of limitations in the correlation filter based method [18], such as the scale variation and occlusion.

To deal with the problems existing in the correlation filter based method, many improved methods have been proposed. For example, Henriques et al. [19] presented the Circulant Structure tracker with Kernels (CSK), which solved

the expensive computation of dense sampling through a cyclically shifted sampling method. This study also proved that the kernel matrix of the samples also had circulant structure. Based on CSK, the kernelized correlation filter (KCF) is proposed, which adds multichannel features to the correlation filter pipeline [20]. The KCF method adopts HOG features instead of the raw pixel values used in CSK, which makes the tracking results more accurate. Additionally, many algorithms are designed to handle the target scale variation in tracking. For example, the Discriminative Scale Space Tracker (DSST) [21] uses a feature pyramid and a 3D correlation filter to estimate scale. The Scale Adaptive with Multiple Features tracker (SAMF) [22] uses a fixed scaling pool to sample the candidates at different sizes. These methods can deal with some limitations of the correlation filter based trackers. However, there are still some problems that are not solved very well, such as the speed and occlusion issue.

In this paper, an improved KCF-based visual tracking method is proposed, which is focused on the problems associated with occlusion and scale variation in the correlation filter based trackers. The main contributions of this paper are summarized as follows. (1) A general long-term motion model is established based on the interval template matching in redetection. (2) A surveyed area is proposed for the detector searching. In addition, the GM(1,1) model is used to estimate the position of the target. When a full occlusion occurs, the GM(1,1) model is also used to predict the position of the target based on the previous behaviors of the target. (3) Two scale variables of the target are defined to measure the scale variation during the tracking. Then a novel method is proposed to estimate the scale variables by using a part-based model. (4) Aiming at the problem of the offset of target position which will reduce the precision and success rate in the process of tracking, the edge information of target and the maximum response value are utilized to correct the target position.

This paper is organized as follows. The proposed KCF-based tracker is described in Section 2. Section 3 gives out the experiments and the results analysis. Some discussions on the performance of the proposed approach are given in Section 4. Finally, Section 5 contains our conclusions.

2. The Proposed Approach

In this paper, the problem of visual tracking based on KCF method is studied, and an improved method is proposed. The work flow of the proposed method is shown in Figure 1, where the KCF tracker is the key part, and another two important parts are the long-term tracking method and the scale estimation method. The general KCF tracking algorithm has been presented in other literature (see [20, 23] for details), so here only the necessary description of the KCF-based method and the improvements of it will be given out.

2.1. The Proposed Long-Term Tracking Method. The general KCF tracking algorithm includes two main phases, namely, the training and detection phases. In the training phase, it involves training classifier $y = f(\mathbf{x})$ with a set of samples $\mathbf{X} = [\mathbf{x}_1 \cdots \mathbf{x}_n]^T$. In detection phase, a new sample $\mathbf{z}$ is detected by

$$f(\mathbf{z}) = \mathcal{F}^{-1} \left(\left(\hat{\mathbf{k}}^{\mathbf{z}\mathbf{x}} \right) \odot \hat{\alpha} \right) \quad (1)$$

where $\hat{\mathbf{k}}^{\mathbf{z}\mathbf{x}}$ is the kernel correlation of $\mathbf{x}$ and $\mathbf{z}$, α is a parameter of f , and $\wedge$ stands for the Fourier transform operation.

In this paper, the success of the tracking result is determined based on the maximum response value in $f(\mathbf{z})$ and a flag *Flag1* is defined, namely,

$$\text{If } \max f(\mathbf{z}) \leq T_a \text{ Then } \text{Flag1} = 1 \quad (2)$$

where T_a is a predefined threshold and *Flag1* = 1 means that the target was not tracked successfully.

To deal with the problem of tracking failure, the target position for the KCF tracker is reinitialized using the grey prediction model GM(1,1) and the interval template matching method. The basic idea of the proposed long-term tracking method is as follows. (1) The grey model GM(1,1) is used to predict the target's position in the first five frames when the target is lost, and the forecast position is used to update the KCF tracker. (2) If the KCF tracker still cannot accurately track the target after the update, then the template matching is performed. In this study, the detector searches for the target in a surveyed area instead of in the whole image. (3) When a full occlusion is occurring, neither KCF tracker nor the detector can find the target. Because the full occlusion usually takes place for a short time, the position of the target can be estimated based on its previous position by the GM (1, 1) method. After leaving the occlusion situation, the target can be found by the detector again and the tracking process is continued. The details of the proposed method are introduced as follows.

2.1.1. The Position Estimation Based on GM(1,1). The grey model GM(1, 1) is a forecast model in grey system, which has a series of characteristics including less data, low computational complexity, and accurate forecasting [24, 25]. So it is suitable for the real-time visual target tracking. In the grey prediction model, the initial data set is defined as

$$\mathbf{x}^{(0)} = \left(x^{(0)}(1), x^{(0)}(2), \dots, x^{(0)}(n) \right) \quad (3)$$

where n is the total number of the initial data set. The 1-AGO (accumulated generating operation, AGO) is defined as

$$\mathbf{x}^{(1)} = \left(x^{(0)}(1), \sum_{k=1}^2 x^{(0)}(k), \dots, \sum_{k=1}^n x^{(0)}(k) \right) \quad (4)$$

According to GM (1, 1), the first-order grey differential equation is donated:

$$\frac{d\mathbf{x}^{(1)}}{dt} + a\mathbf{x}^{(1)} = u \quad (5)$$

where a and u are the grey parameters, which can be solved by the least-square method, namely,

$$(a, u)^T = (B^T B)^{-1} B^T Y \quad (6)$$

FIGURE 1: Flow diagram of the improved KCF-based approach for target tracking.

where the coefficient vector Y and the accumulated matrix B are expressed by

$$B = \begin{bmatrix} -0.5(x^{(1)}(2) + x^{(1)}(1)) & 1 \\ -0.5(x^{(1)}(3) + x^{(1)}(2)) & 1 \\ \vdots & \vdots \\ -0.5(x^{(1)}(n) + x^{(1)}(n-1)) & 1 \end{bmatrix}, \quad (7)$$

$$Y = \begin{bmatrix} x^{(0)}(2) \\ x^{(0)}(3) \\ \vdots \\ x^{(0)}(n) \end{bmatrix}$$

Then the solution of (4) is obtained as follows:

$$\hat{x}^{(1)}(k) = \left(x^{(0)}(1) - \frac{u}{a}\right) e^{-a(k-1)} + \frac{u}{a} \quad (8)$$

By applying the inverse accumulated generation operation, the predicted equation will be obtained:

$$\hat{x}^{(0)}(k) = \left(x^{(0)}(1) - \frac{u}{a}\right) (1 - e^a) e^{-a(k-1)}, \quad (k > 1) \quad (9)$$

where $\hat{x}^{(0)}(k)$ is called the predicted value of the GM(1,1) model.

In the proposed approach, if the target cannot be tracked successfully based on the ordinate and horizontal coordinates of the target in the previous four frames, the corresponding horizontal coordinate prediction model $x^{(0)} = (x^{(0)}(1), x^{(0)}(2), x^{(0)}(3), x^{(0)}(4))$ and the ordinate coordinate prediction model $y^{(0)} = (y^{(0)}(1), y^{(0)}(2), y^{(0)}(3), y^{(0)}(4))$ are respectively established, and then two models are used to realize the prediction of the horizontal and ordinate coordinates of the target. The center position of the target (x_p, y_p) can be expressed by

$$(x_p, y_p) = \begin{cases} (x^{(0)}(5), y^{(0)}(5)) & \sigma_x > T_b \text{ \& } \sigma_y > T_b \\ (\mu_x, \mu_y) & \text{else} \end{cases} \quad (10)$$

FIGURE 2: The detection strategy: (a) exhaustive search, (b) based on the proposed surveyed area.

where $x^{(0)}(5)$ and $y^{(0)}(5)$ are the GM(1,1) model prediction values by using the position of the target in the previous four frames, σ_x and σ_y are the variance of $x^{(0)}$ and $y^{(0)}$, T_b is a threshold to estimate the magnitude of the change in several numbers in $x^{(0)}(k)$ and $y^{(0)}(k)$, and μ_x and μ_y are the average of $x^{(0)}$ and $y^{(0)}$, respectively.

2.1.2. Occlusion Handling Based on Interval Template Matching Method. Redetection involves recovering the tracker when it fails. If the object is lost, the detector facilitates the reinitialization of the target position for the KCF tracker, so the performance of the KCF tracker is improved for long-term visual object tracking in challenging conditions. In the proposed method, the detector is based on a sliding window approach and template matching approach. In order to implement the proposed algorithm in real time, the grey model GM(1,1) is used to predict the target's position in the first five frames after the target is lost, and then the forecast position is used to update the KCF tracker.

If the KCF tracker still cannot accurately track the target after the update, then template matching is performed. In this paper, the Bhattacharyya coefficient $BC(T, C)$ is used to calculate similarity between the target and the candidate model during the template matching process, namely [26],

$$BC(T, C) = \sum_{x \in X} \sqrt{T(x)C(x)} \quad (11)$$

where T, C stand for the feature vector of grayscale histogram of the target model and the candidate model, if the original image is grayscale. Otherwise, they are the reconstruction feature vectors of R, G, B channel histograms of the target model and the candidate model. The value of $BC(T, C)$ is within the range of $[0, 1]$, and the larger it is, the more similar the two models are.

In the template matching process, if the detector performs the search on the entire area of an image, it will take a significant amount of time to find the target. Hence, it is indispensable to detect only on a small region instead of the entire region of an image. And the detector should extract an area with high probability of containing the target. Based on the statistical information obtained from the GM (1, 1)

model introduced above, a surveyed area S_Area is defined as follows:

$$S_Area = (x_p - 0.5 * H : x_p + 0.5 * H, y_p - 0.5 * W : y_p + 0.5 * W) \quad (12)$$

where H and W are the height and width of the matching template. The detection strategy based on the proposed surveyed area is shown in Figure 2.

2.2. Scale Estimation Using Multiblock Scheme. In visual tracking, one of the main drawbacks of the KCF tracker is that it does not address the issue of the target scale variation. When the target scale changes, the KCF tracker is prone to drift and cannot efficiently locate the target. To deal with this problem, a part-based model to estimate the target scale is proposed. In the proposed method, a global block is first used to cover the entire target. Then the main four parts are divided out from the global block (see Figure 3). The splitting direction is simply determined by the height, width, and the modified position of target, and two scale variables ($x1$ and $x2$) are defined to estimate the height and width variation of the target, respectively, which are calculated by

$$x1 = \frac{(xu + xd)}{2} \quad (13)$$

$$x2 = \frac{(xr + xl)}{2}$$

where xu, xd, xr , and xl are defined as

$$\begin{aligned} xu &= \frac{couu_t}{couu_{t-1}} \\ xd &= \frac{coud_t}{coud_{t-1}} \\ xr &= \frac{cour_t}{cour_{t-1}} \\ xl &= \frac{coul_t}{coul_{t-1}} \end{aligned} \quad (14)$$

where $couu, coud, coul$, and $cour$ are the number of pixel points in the top, bottom, left, and right four blocks, respectively. If there are more black pixels than white pixels, the

FIGURE 3: The procedure of the proposed multiblock scheme.

$couu$, $coud$, $coul$, and $cour$ are the number of black pixel points; otherwise, they are the number of white pixel points. The basic work flow is as follows. When a new frame comes, the t -th frame image patches are captured in several scale spaces based on the $(t - 1)$ -th frame image patches. The four blocks are converted into binary image blocks, and the number of black and white pixel points in the four blocks is figured out. In the t -th frame, the larger the target, the more black pixel points are counted in the four blocks. Therefore, x_1 and x_2 not only reflect the scale change of target but also can be used to update the height and width of the target separately. In this paper, the following rule is used to judge whether there is a scale change of the target:

$$\begin{aligned} & \text{If } T_c < x_1 < T_d \quad (x_1 \neq 1) \\ & \text{and } T_c < x_2 < T_d \quad (x_2 \neq 1), \end{aligned} \quad (15)$$

then $Flag2 = 1$

where T_c and T_d are two predefined parameters, and $Flag2 = 1$ means the scale change occurs. In order to more accurately estimate the change of the target scale, a new position $pos_{fl}(x, y)$ is constructed as

$$pos_{fl}(x, y) = \beta pos_{eg}(x, y) + (1 - \beta) pos_{mv}(x, y) \quad (16)$$

where $pos_{eg}(x, y)$ is the average value of each pixel point which is detected by using the Sobel operator on target [27], β is the learning rate, and $pos_{mv}(x, y)$ is the target position detected by the KCF tracker. Then, the height $scale_ht_t$ and width $scale_wt_t$ of target can be updated by

$$\begin{aligned} scale_ht_t &= x_1 \times scale_ht_{t-1} \\ scale_wt_t &= x_2 \times scale_wt_{t-1} \end{aligned} \quad (17)$$

By the proposed approach, the target scale can be estimated recursively in every frame during the tracking. The procedure of the proposed target scale estimation based on the multiblock scheme is abstracted as follows and an example of it is shown in Figure 3.

Step 1. Deploy the Sobel operator to modify the center position of target.

Step 2. Determine the multiblock scheme by the height, width, and the modified position of target.

Step 3. Calculate two scale variables by using the multiblock scheme.

Step 4. Judge whether there is a scale change of the target.

Step 5. Update the scale of the target recursively in every frame, if the scale of the target changes.

2.3. Model Updating for Tracker. In the KCF tracker, to ensure that the tracking process can adapt to the target variations at the following input frames, the parameter $\hat{\alpha}$ and the feature $\hat{\mathbf{x}}$ in (1) should be updated as follows:

$$\begin{aligned} \hat{\alpha}(t+1) &= (1 - \varphi) \hat{\alpha}(t) + \varphi \hat{\alpha} \\ \hat{\mathbf{x}}(t+1) &= (1 - \varphi) \hat{\mathbf{x}}(t) + \varphi \hat{\mathbf{x}} \end{aligned} \quad (18)$$

In the detector model, the target model describes the similarity between the candidate model and the target. Therefore, the target model should also be constantly updated


```

Load the initial the parameters;
For t= 1 to N % N is the number of frames in the sequence.
    If t = 1 then
        Get  $patch_1$ ; Set  $\hat{\alpha}_1$  and  $\hat{\mathbf{x}}_1$ ;
        Get  $cour_1, coul_1, couu_1$  and  $coud_1$ ;
    End
    If t > 1, then
        Calculate  $\max f(\mathbf{z})$ ;
        Get  $cour_t, coul_t, couu_t$  and  $coud_t$ ;
        Calculate  $x_1$  and  $x_2$ ;
        If  $T_c < x_1 < T_d(x_1 \neq 1)$  and  $T_c < x_2 < T_d(x_2 \neq 1)$ , then
            Flag2=1, and update the target position and scale;
        End
        If  $\max f(\mathbf{z}) \leq T_a$ , then
             $M=M+1$ ; Call GM (1, 1) %Prediction and update KCF tracker's target position.
            If mod(M,5)= 0, then
                Call GM(1,1);
                Set up the surveyed area; Call Tem_matching( );
                Get max(BC(T, C)) and update KCF tracker's target position;
            End
        End
        Update  $patch_{t+1}, \hat{\alpha}_{t+1}$  and  $\hat{\mathbf{x}}_{t+1}$ ;
    End
End

```

ALGORITHM 1: The pseudocode of the proposed tracking method.

TABLE 1: The parameters of the proposed method.

Parameters	Values	Remarks
φ	0.02	The initial learning rate (see (18))
β	0.1	The learning rate (see (16))
η	0.9	The update rate (see (19))
T_a	0.25	The threshold in (2)
T_b	5	The threshold in (10)
T_c	0.95	The value of the parameter in (15)
T_d	1.05	The value of the parameter in (15)

to strictly represent the target. In this paper, the target model $patch_t$ is updated by the following rule:

$$\begin{aligned}
 & patch_t \\
 &= \begin{cases} (1 - \eta) patch_{t-1} + \eta patch_1 & \max f(\mathbf{z}) > T_a \\ patch_{t-1} & \text{otherwise} \end{cases} \quad (19)
 \end{aligned}$$

where η is the update rate and $patch_1$ is the initial template of the target.

The pseudocode of the proposed tracking method is summarized in Algorithm 1

3. Evaluation of the Proposed Method

In this section, the proposed method is evaluated on the object tracking benchmark OTB50 [28, 29]. These sequences are recorded in various scenarios and contain different

challenges such as scale variation, background clutter, illumination variation, and occlusion. The experiments are implemented with MATLAB, on a computer with Intel Core i7-6500U and 2.50GHz CPU with 4.096GB RAM. In this paper, the HOG cell size is 4 * 4 and the number of orientation bins is nine. To mitigate the boundary effect, the extracted features are multiplied by a cosine window. The parameters of the proposed method are listed in Table 1.

To objectively evaluate the proposed algorithm (SLKCF), which is compared with other state-of-the-art tracking methods, including the Multistore Tracker (MUSTER) [30], Discriminative Scale Space Tracker (DSST) [21], Scale Adaptive with Multiple Features tracker (SAMF) [22], KCF tracker [20], CSK tracker [19], Struck [31], Tracking-Learning-Detection method (TLD) [32], Compressive Tracker (CT) [33], and Distribution Fields for Tracker (DFT) [34].

3.1. Quantitative Evaluation. In this study, three performance criteria often used in other literature are chosen to evaluate

TABLE 2: The distance precision of different trackers in eleven representative videos.

The tested sequences	The methods									
	SLKCF	KCF	DSST	SAMF	CSK	MUSTER	CT	TLD	Struck	DFT
Girl	100	87.40	92.80	100	55.40	99.00	60.80	91.80	100	29.60
walking2	100	63.40	100	77.00	46.80	100	43.20	42.60	98.20	40.20
car4	100	94.99	100	100	35.51	100	28.07	87.41	99.24	26.25
carScale	97.22	80.56	75.79	76.59	65.08	76.19	71.83	85.32	64.68	65.08
football	100	79.56	79.83	80.11	79.83	79.83	79.83	80.39	75.14	84.25
freeman1	93.25	39.26	38.34	38.96	55.52	89.26	39.57	53.99	80.06	94.17
freeman4	90.81	53.00	95.76	25.80	18.73	91.17	6.36	40.99	37.46	30.04
Shaking	95.62	2.74	99.73	4.11	56.44	98.36	4.66	40.55	19.18	83.01
Soccer	92.86	27.55	69.39	79.08	13.52	46.17	21.94	11.48	25.26	22.45
woman	95.98	93.80	93.80	93.80	24.96	93.97	20.44	19.11	100	95.14
Jogging1	99.35	23.13	23.13	97.39	22.80	95.44	23.13	97.39	24.10	21.50
Average	96.83	58.67	78.96	70.26	43.14	88.13	36.35	59.18	65.76	53.79

TABLE 3: The success rate of different trackers in eleven representative videos.

The tested sequences	The methods									
	SLKCF	KCF	DSST	SAMF	CSK	MUSTER	CT	TLD	Struck	DFT
Girl	96	83.2	67	100	39.8	58	17.8	76.4	98	25.2
walking2	98	41.4	41.4	51	38.8	100	38.4	34	43.4	38.2
car4	90.74	36.42	39.91	100	27.62	100	27.47	79.21	39.76	25.8
carScale	96.83	44.44	44.84	59.92	44.84	73.81	44.84	43.65	43.25	44.84
football	85.36	70.17	72.65	67.68	65.75	62.71	78.45	41.16	66.02	84.25
freeman1	23.01	16.26	15.03	28.22	14.42	62.27	10.12	21.17	21.47	17.79
freeman4	28.98	18.37	38.16	16.61	16.96	29.68	0.35	26.86	15.55	18.02
Shaking	91.23	1.37	96.99	1.37	58.08	98.9	4.11	40	16.71	82.47
soccer	47.19	23.21	33.16	33.93	13.78	16.58	20.15	12.24	15.56	21.94
woman	91.96	93.63	93.47	91.96	24.46	93.3	15.91	16.58	93.47	93.47
Jogging1	97.07	22.48	22.48	96.74	22.48	94.79	22.48	96.74	22.48	21.5
Average	76.94	41.00	51.37	58.86	33.36	71.82	25.46	44.36	43.24	43.04

these trackers. The first one is the distance precision (DP), which is defined as the ratio of the number of correctly tracked frames to the total number of frames for a range of distance thresholds at 20 pixels in the sequence. If the distance precision is higher at low thresholds, it means that the results are more accurate. The second one is the success rate (SR), which shows the percentage of successfully tracked frames. The standard for successful tracking is that the bounding box overlap between the tracked object and the ground-truth is bigger than a threshold value (which is set to 0.5 in this study). The third one is the tracking speed, which is also a vital criterion to evaluate a tracker. If a tracker can obtain a speed of 25 frame per second (FPS), this tracker is regarded to run in real time. In the quantitative evaluation experiments, eleven representative videos are selected from the object tracking benchmark OTB50 and used to show the performances of the proposed approach. The average distance precision and success rate plots of one-pass evaluation (OPE) in these eleven videos are shown in Figure 4. The precision

and the success rate of all compared trackers in these eleven videos are listed in Tables 2 and 3. To further test the proposed approach, all the sequences of OTB50 are used to compare the deferent methods and the results are listed in Table 4.

The results of Figure 4 show that the proposed method (SLKCF) achieves outstanding performance in the distance precision and the success rate, which indicate that the SLKCF tracker has the best overall performance using both the two metrics. Also the results in Figure 4 show that the proposed approach significantly outperforms the traditional KCF tracker. The results in Tables 2 and 3 reveal that the proposed occlusion handling model and scale estimation using multiblock scheme can help the KCF tracker achieve outstanding performance. The average precision of the proposed approach is 96.83% but only 58.67% of the standard KCF method. The average rate of the proposed approach is 76.94% but only 41.00% of the standard KCF method. Table 4 shows the proposed approach has the best performance with DP=86.29% and SR=78.31% for all the sequences of OTB50.

TABLE 4: The performances of different trackers in all videos of OTB50.

Average value	The methods									
	SLKCF	KCF	DSST	SAMF	CSK	MUSTER	CT	TLD	Struck	DFT
DP	86.29	70.97	72.60	75.65	54.46	86.13	40.57	60.75	65.61	49.6
SR	78.31	60.30	58.91	67.33	44.26	77.98	34.13	52.09	55.93	44.38
FPS	85.46	168.41	35.18	14.79	235.62	5.83	43.87	22.53	17.89	7.97

FIGURE 4: The average distance precision and success rate plots of OPE in eleven representative videos.

Though the proposed method is slower than the KCF and the CSK tracker, it is faster than the other trackers and can also track the target in real time.

3.2. Qualitative Evaluation. In order to evaluate the proposed algorithm more intuitively in comparison with other trackers, Figure 5 displays the results of different trackers for several representative sequences. In Figure 5(a), the sequences of jumping show the original images are grey and the scene with fast motion due to the target. Owing to the integration of the GM(1,1) grey model and the interval template matching method, the proposed SLKCF obtains good performance. The KCF tracker does well when the target moved smoothly but drifts when fast motion occurs. Figure 5(b) shows a full occlusion in which the target overlaps with the pole. Before the full occlusion, the target can be tracked by every tracker successfully. However, when the full occlusion is occurring, only the proposed method tracks the target. At this moment, the GM(1,1) model helps correct the tracking process. In Figure 5(c), the results are obtained when background clutter, illumination changes, motion blur, occlusion are presented. At this moment, the proposed SLKCF tracker can also deal with these challenging factors well like other methods. And the proposed mechanisms in this study make it more efficient and consume less computational time. The sequences in

Figure 5(d) depict scale variation accompanied with partial occlusion. The ratio of the maximal target size to the minimal one is more than 30 when the target vehicle approaches the camera from far away. DSST, SAMF, MUSTER, and the proposed method can adapt to the target scale change. It is obvious that the four trackers above work well when the scale changes slightly. After the second hundredth frame, DSST, SAMF, and MUSTER only can capture part of the car. In contrast, the proposed method in this paper can accurately track the entire car. The results in Figures 5(e)-5(f) show that the proposed approach can deal with the problem efficiently so that the target undergoes scale variation, illumination changes, and occlusion.

4. Discussion

The videos in the object tracking benchmark are annotated with different attributes, which represent challenges in each video. The attributes include the occlusion (OCC), illumination variation (IV), deformation (DEF), scale variation (SV), fast motion (FM), background clutter (BC), in-plane rotation (IPR), out-of-plane rotation (OPR), and out of view (OV). In order to further discuss the validity of the proposed algorithm in these challenging situations, some experiments are conducted. The results of distance precision for the challenging attributes are shown in Table 5 and Figure 6.

FIGURE 5: The tracking results of different trackers on six videos.

TABLE 5: The distance precision of the different trackers in nine challenging attributes.

Different attributes	The methods									
	SLKCF	KCF	DSST	SAMF	CSK	MUSTER	CT	TLD	Struck	DFT
FM	0.954	0.673	0.797	0.832	0.345	0.721	0.381	0.386	0.633	0.609
BC	0.962	0.366	0.83	0.544	0.499	0.748	0.355	0.441	0.399	0.632
MB	0.944	0.607	0.816	0.864	0.192	0.701	0.212	0.153	0.626	0.588
DEF	0.977	0.585	0.585	0.956	0.239	0.947	0.218	0.582	0.621	0.583
IV	0.961	0.548	0.907	0.692	0.326	0.846	0.188	0.396	0.609	0.567
IPR	0.957	0.529	0.788	0.578	0.492	0.829	0.407	0.578	0.574	0.584
OCC	0.97	0.635	0.788	0.787	0.409	0.852	0.409	0.586	0.656	0.485
OPR	0.961	0.541	0.743	0.662	0.436	0.855	0.365	0.579	0.584	0.584
SV	0.962	0.603	0.851	0.661	0.413	0.882	0.33	0.526	0.693	0.54
Average	0.961	0.565	0.789	0.731	0.372	0.82	0.318	0.47	0.599	0.575

FIGURE 6: Performance of 10 trackers in nine challenging attributes (the legend contains the distance precision score at a threshold of 20 for each tracker in OPE).

The results show that the proposed approach ranks the first with a large margin comparing to other trackers. These promising results show that the proposed tracker (SLKCF) is more effective than the other tracker for these challenging attributes.

5. Conclusion

In this paper, the target tracking task is studied and an improved KCF-based tracking algorithm that can be implemented in real time is proposed. In the proposed approach, the KCF tracker is combined with the GM(1,1) grey model, the interval template matching method, and part-based model, which enable the proposed algorithm to track the target in challenging environments containing occlusion and object scale variation successfully. These performances of the proposed approach above are superior to the traditional KCF-based tracker. The experimental results show that the proposed method outperforms the other trackers in terms of distance precision and success rate. In future work, the tracking speed of the proposed approach should be further improved and some novel intelligent methods will be studied to improve the performance of the tracking algorithm.

Data Availability

All data supporting this study are openly available from the website of the Visual Tracker Benchmark at http://cvlab.hanyang.ac.kr/tracker_benchmark/datasets.html.

Conflicts of Interest

The authors declared that they have no conflicts of interest in this work.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61873086, 61801169, and 61573128) and the Fundamental Research Funds for the Central Universities (2018B23214).

References

- [1] C. Kanellakis and G. Nikolakopoulos, "Survey on Computer Vision for UAVs: Current Developments and Trends," *Journal of Intelligent & Robotic Systems*, vol. 87, no. 1, pp. 141–168, 2017.
- [2] C.-C. Chiang, M.-C. Ho, H.-S. Liao, A. Pratama, and W.-C. Syu, "Detecting and recognizing traffic lights by genetic approximate ellipse detection and spatial texture layouts," *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 12, pp. 6919–6934, 2011.
- [3] D. Wei, "Discrete-State-Based Vision Navigation Control Algorithm for One Bipedal Robot," *Mathematical Problems in Engineering*, vol. 2015, Article ID 168645, 12 pages, 2015.
- [4] M. Crocco, M. Cristani, A. Trucco, and V. Murino, "Audio surveillance: A systematic review," *ACM Computing Surveys*, vol. 48, no. 4, 2016.
- [5] H. Yang, L. Shao, F. Zheng, L. Wang, and Z. Song, "Recent advances and trends in visual tracking: a review," *Neurocomputing*, vol. 74, no. 18, pp. 3823–3831, 2011.
- [6] A. Smeulder, D. Chu, R. Cucchiara, S. Calderara, A. Deghan, and M. Shah, "Visual tracking: an experimental survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1442–1468, 2013.
- [7] T. Zhou, M. Zhu, D. Zeng, and H. Yang, "Scale adaptive kernelized correlation filter tracker with feature Fusion," *Mathematical Problems in Engineering*, vol. 2017, Article ID 1605959, 8 pages, 2017.
- [8] C. Liu, P. Liu, W. Zhao, and X. Tang, "Robust Tracking and Redetection: Collaboratively Modeling the Target and Its Context," *IEEE Transactions on Multimedia*, vol. 20, no. 4, pp. 889–902, 2018.
- [9] B. Ma, L. Huang, J. Shen, L. Shao, M.-H. Yang, and F. Porikli, "Visual tracking under motion blur," *IEEE Transactions on Image Processing*, vol. 25, no. 12, pp. 5867–5876, 2016.
- [10] L. Zhang, Y. Wang, H. Sun, Z. Yao, and S. He, "Robust visual correlation tracking," *Mathematical Problems in Engineering*, vol. 2015, Article ID 238971, 13 pages, 2015.
- [11] Q. Liu, X. Zhao, and Z. Hou, "Survey of single-target visual tracking methods based on online learning," *IET Computer Vision*, vol. 8, no. 5, pp. 419–428, 2014.
- [12] Y. Yuan, H. Yang, Y. Fang, and W. Lin, "Visual Object Tracking by Structure Complexity Coefficients," *IEEE Transactions on Multimedia*, vol. 17, no. 8, pp. 1125–1136, 2015.
- [13] X. Mei and H. Ling, "Robust visual tracking and vehicle classification via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 11, pp. 2259–2272, 2011.
- [14] S. Baker and I. Matthews, "Lucas-Kanade 20 years on: a unifying framework," *International Journal of Computer Vision*, vol. 56, no. 3, pp. 221–255, 2004.
- [15] A. Banharnsakun and S. Tanathong, "Object detection based on template matching through use of best-so-far ABC," *Computational Intelligence and Neuroscience*, vol. 2014, Article ID 919406, 8 pages, 2014.
- [16] B. Babenko, M. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1619–1632, 2010.
- [17] Y. Xu, Y. Li, J. Wang, Z. Miao, H. Li, and Y. Zhang, "Feature adaptive correlation tracking," *IEICE Transaction on Information and Systems*, vol. E100D, no. 3, pp. 594–597, 2017.
- [18] J. Ma, H. Luo, B. Hui, and Z. Chang, "Robust scale adaptive tracking by combining correlation filters with sequential Monte Carlo," *Sensors*, vol. 17, no. 3, 2017.
- [19] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 7575, no. 4, pp. 702–715, 2012.
- [20] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2015.
- [21] M. Danelljan, G. Häger, F. Shahbaz Khan, and M. Felsberg, "Accurate Scale Estimation for Robust Visual Tracking," in *Proceedings of the British Machine Vision Conference*, pp. 65.1–65.11, Nottingham, UK, 2014.

- [22] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 8926, pp. 254–265, 2015.
- [23] S. Jeong, G. Kim, and S. Lee, "Effective visual tracking using multi-block and scale space based on kernelized correlation filters," *Sensors*, vol. 17, no. 3, 2017.
- [24] Q. Yu, J. Lyu, L. Jiang, and L. Li, "Traffic anomaly detection algorithm for wireless sensor networks based on improved exploitation of the GM(1,1) model," *International Journal of Distributed Sensor Networks*, vol. 12, no. 7, 2016.
- [25] L. Han, W. Tang, Y. Liu, J. Wang, and C. Fu, "Evaluation of measurement uncertainty based on grey system theory for small samples from an unknown distribution," *Science China Technological Sciences*, vol. 56, no. 6, pp. 1517–1524, 2013.
- [26] Y. El merabet, Y. Ruichek, S. Ghaffarian et al., "Maximal similarity based region classification method through local image region descriptors and Bhattacharyya coefficient-based distance: Application to horizon line detection using wide-angle camera," *Neurocomputing*, vol. 265, pp. 28–41, 2017.
- [27] S. Chen, X. Yang, Z. You, and M. Wang, "Innovation of aggregate angularity characterization using gradient approach based upon the traditional and modified Sobel operation," *Construction and Building Materials*, vol. 120, pp. 442–449, 2016.
- [28] Y. Wu, J. Lim, and M.-H. Yang, "Object tracking benchmark," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1834–1848, 2015.
- [29] M. Dai, S. Cheng, X. He, and D. Wang, "Object tracking in the presence of shaking motions," *Neural Computing and Applications*, 2018.
- [30] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Adaptive Correlation Filters with Long-Term and Short-Term Memory for Object Tracking," *International Journal of Computer Vision*, pp. 1–26, 2018.
- [31] S. Hare, S. Golodetz, A. Saffari et al., "Struck: structured output tracking with kernels," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 10, pp. 2096–2109, 2015.
- [32] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1409–1422, 2012.
- [33] B. Wang, L. Tang, J. Yang, B. Zhao, and S. Wang, "Visual Tracking Based on Extreme Learning Machine and Sparse Representation," *Sensors*, vol. 15, no. 10, pp. 26877–26905, 2015.
- [34] T. Zhang, S. Liu, N. Ahuja, M.-H. Yang, and B. Ghanem, "Robust visual tracking via consistent low-rank sparse learning," *International Journal of Computer Vision*, vol. 111, no. 2, pp. 171–190, 2014.