IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON BIOLOGICALLY INSPIRED IMAGE PROCESSING CHALLENGES AND FUTURE DIRECTIONS

Received May 6, 2019, accepted May 26, 2019, date of publication May 30, 2019, date of current version June 11, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2919985

Moving Target Detection and Tracking Algorithm Based on Context Information

JING LI, JUNZHENG WANG, AND WENXUE LIU

Key Laboratory of Complex System Intelligent Control and Decision, School of Automation, Beijing Institute of Technology, Beijing 100081, China

Corresponding author: Jing Li (bitljing@bit.edu.cn)

This work was supported by the National Nature Science Foundation of China under Grant 6110 3157.

ABSTRACT To improve the robustness of target tracking algorithms in a complex environment, this paper proposes the moving target detection and tracking algorithm based on context information and closed-loop learning. A context region is composed of the target region and its current neighboring background. For every frame that follows from a video stream, the long-term tracking task is principally decomposed into four parts of synchronous operation: tracking, detection, integration, and learning. First, the tracker obtains the posterior probability of the target location and estimates the target state over succeeding frames by exploiting the spatio-temporal local information. Meanwhile, the detector searches for the target in independent frames combining with the context information of tracker, and automatically reinitialize the tracker when it fails. Then, the integrator attains the best location of the target by merging the output results of tracker and detector together through an optimal strategy. Finally, the learning process is designed as the feedback and generates training samples to update the detector according to the results of tracker and detector. Experimentally, we evaluate the performance against several latest techniques on various benchmarks, and the results demonstrate that the proposed algorithm performs remarkably in terms of robustness and tracking accuracy.

INDEX TERMS Target detection, target tracking, moving target, context information.

I. INTRODUCTION

Visual tracking is one of the most crucial issues in computer vision due to its wide range of applications such as motion analysis, image compression, monitoring, human-computer interaction, and so forth. The main challenge in visual tracking is to cope with the appearance variations of target, specifically including the intrinsic variations of pose, shape and scale, as well as variations caused by illumination changes, occlusion, background clutter and other extrinsic factors in the environment [1].

A huge amount of research has been spent on visual tracking and numerous tracking algorithms have been proposed, which can be categorized into short-term tracking and long-term tracking approaches.

Short-term tacking methods estimate the target's motion frame by frame under the assumption that the target is in absence of disappearance and complete occlusion. The research on this kind of method is focused on improving the speed, precision and robustness of tracking [2]. Traditional short-term tacking methods generally follow the target by extracting the target features, such as color [3]–[6], contour [7], texture [8], optical flow [9] and eigenbasis [1], which can perform effectively in some specific scenes. Considering the fact that these features are separately subject to the disturbance in sourrundings, some existing algorithms [10]–[12] improve the tracking performance by integration of multiple features. Other algorithms, including Kalman Fliter (KF) [13], Extended Kalman Fliter (EKF) [14], and Particle Filter (PF) [15], regard tracking as a problem of state estimation and calculate the posterior probability about observation by introducing various prediction solutions [16], [17]. Moreover, PF has been widely applied in visual tracking owing to its ability of dealing with multiple mode problems without the limitation of Gaussian hypothesis and linearity [12], [18]. Recently, a novel kind of algorithm [19]–[23] that exploits the information of target local context has achieved success and caught much attention, for instance, the Spatio-Temporal Context (STC) algorithm proposed by Zhang *et al.* [23], which can track target fast and robustly with the help of spatio-temporal context information.

The associate editor coordinating the review of this manuscript and approving it for publication was Jiachen Yang.

2169-3536 © 2019 IEEE. Translations and content mining are permitted for academic research only.
Personal use is also permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

Short-term tracking methods can only work in short image sequences and would inevitably fail as long as the target is fully occluded or disappeared, thus limiting the scope of its application. For this reason, they can not be directly applied to long-term tracking problems, where one dominating trend is to apply appearance-based detectors. With the capability of detection to a certain extent, current long-term tracking methods commonly are able to redetect the target when the target appears again after disappearence [18], [24]–[27]. Williams *et al.* [24] applied an invariant detector trained offline to evaluate the reliability of trajectory, resulting in poor adaptability to the changes of target appearance. In [18], with the detector integrated within a particle filtering framework, the method depends only on information from the past and is suitable for online applications. In [25]–[28], the tracking is deemed as a binary classification matter and realized by utilizing a strong classifier to discriminate the target from surrounding background. Meanwhile, different sorts of learning algorithms are brought in to update the classifier online [29], [30], therefore the tracker can better handle the appearance changes and short-time occlusion of the target in tacking process. Furthermore, Kalal *et al.* [31], [32] investigated long-term tracking and proposed the Tracking-Learning-Detection (TLD) algorithm by integrating the short-term tracking, detection and learning mechanism into a coherent framework, where tracking and detection are independent processes and operate synchronously.

In summary, STC algorithm can track target fast and robustly with the help of spatio-temporal context information, but it can't track when the target is occluded severely. The TLD algorithm has strong robustness to intrinsic variations and can be effectively applied to situation where the target is partly occluded or disappeared, but it has poor performance in complex environment with illumination changes and severe occlusion. Taking into account the previous points of view of related work, this work proposed the moving target detection and tracking algorithm based on context information, which comprises tracking, detection as well as learning process. Tracker takes advantage of the local spatio-temporal information to determine the target location over succeeding frames. It can stably track the target in occasions with illumination changes and partial target occlusion, whereas it would fail forever if the target disappeared. When the tracker fails, detector performs a global search in each independent frame and automatically reinitializes the tracker. Otherwise, it will make use of the context information of the tracker and consequently search in a smaller local context region. The integrator combines the results of tracker and detector into a best one through an optimal strategy. Furthermore, learning process generates new training samples to update the detector according to the output results of the tracker and detector. Making full use of local context information of the target, the proposed algorithm has not only strong recoverability after the target disappeared, but also outstanding performance in complex environment with dramatic lighting changes, severe occlusion and so on.

The contribution of this paper is to propose a robust moving target detection and tracking method based on context information and closed-loop learning framework. The proposed method can track target fast and robustly when the target undergoes rotation and scaling changes, and when the target disappears with severely occlusion and reappears, it can re-track the target stably.

The rest of the paper is organized as follows: Section II briefly reviews the tracking and detection framework based on context information and closed-loop control. On this basis, details about the proposed algorithm in this paper are discussed in Section III, where we discuss the target representation and the implementation of each module. In Section IV, we perform experiments which compare the proposed algorithm with other latest algorithms, and report the experimental results and analysis. Later, we conclude the paper in Section V.

II. TRACKING AND DETECTION FRAMEWORK BASED ON CONTEXT INFORMATION AND CLOSED-LOOP CONTROL

In this section, the tracking and detection framework based on context information and closed-loop control is presented for long-term tracking of an unknown target in video streams. As depicted in Fig. 1, the framework mainly consists of four components, that is the tracker, detector, learning process and integrator.

The *tracker* based on context information estimates the target state over consecutive frames, assuming that the target's motion between adjacent frames is limited and the target is visible. However, the tracker will fail and never recover by itself in case that the target is occluded completely or moves out of the view boundary.

The *detector* based on context information conducts a search for the target in each independent frame and localize all appearances that are similar to the target model, which contains the target's appearance information from initialization and learning process in tracking process. In addition, the output results of detector will be applied to reinitialize the tracker when it fails.

The *integrator* merges the output results of tracker and detector together through an optimal strategy and eventually attains a best bounding box that defines the target's location. If there are no outcomes in tracker and detector, the target is considered as invisible and no tracking box will be returned by integrator.

The *learning* process is designed as the feedback to evaluate detector's error and correct them. Since the detector searches for targets with known target model library, it is likely to fail when the target's appearance changes. In order to avoid the same error in the subsequent tracking, the learning process generates training samples according to the output results of tracker and detector, so that it updates the target model library and related parameters of detector in runtime.

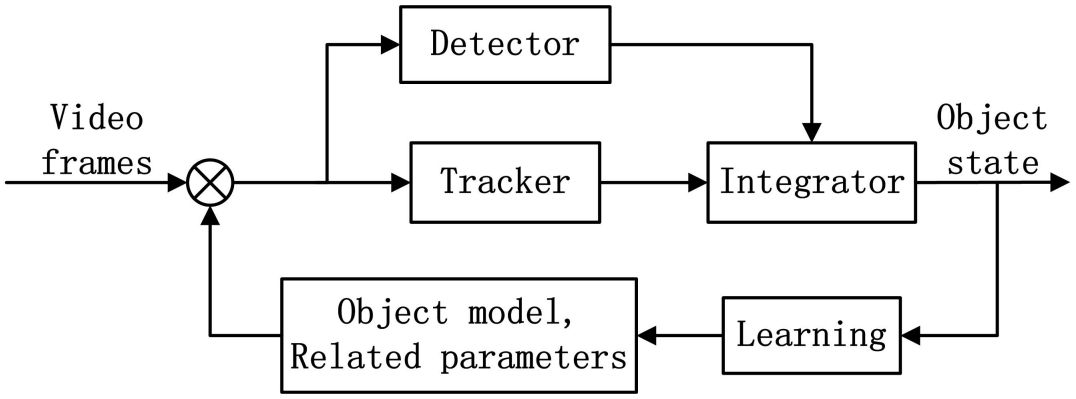


FIGURE 1. Tracking and detection framework based on closed-loop control.

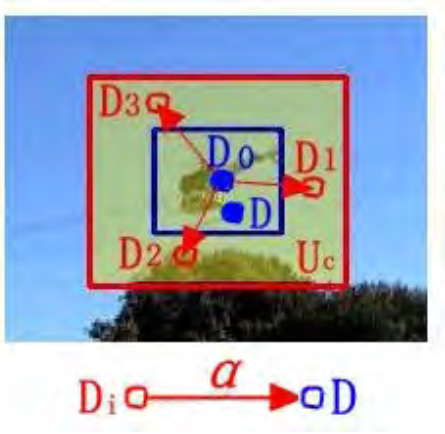


FIGURE 2. Representation of context information.

III. DETECTION AND TRACKING ALGORITHM BASED ON CONTEXT INFORMATION

A context region is composed of the target region and its current neighboring background, and its size is determined as twice the size of the target region [23]. As shown in Fig. 2, the context region of target is inside the red box, including the target region inside the blue one that centers at $I(D_0)$. The symbol D stands for possible target center of target while $D_i(i \in Z^+)$ for the points in context region. The context information of target can be represented by the coordinates $D_i(x, y)$ and corresponding intensity $I(D_i)$ of points in the context region U_c (See D_1, D_2, D_3 in Fig. 2). Additionally, vector α starts from D_i and ends at target possible center D, thus representing the relative distance and direction between target and points in context.

Within the Bayesian framework, target tracking can be considered to be a problem of obtaining the posterior probability $P(D)$ of target location, exactly,

$$P(D) = \sum_{D_i \in U_C} P(D|(I(D_i), D_i)) \cdot P(I(D_i), D_i). \quad (1)$$

The priori probability $P(I(D_i), D_i)$ models the appearance information of points in context with a weighted function $\omega(\bullet)$. It is noted that the closer points are to the target, the larger their weights are supposed to be, thereby $P(I(D_i), D_i)$ is defined as below (k_1 denotes a normalization constant that limits $P(I(D_i), D_i)$ into range from 0 to 1 that satisfies the definition of probability):

$$\begin{aligned} P(I(D_i), D_i) &\triangleq I(D_i) \cdot \omega(D_i - D_0) \\ &= I(D_i) \cdot k_1 e^{-\frac{|D_i - D_0|^2}{\sigma^2}}. \end{aligned} \quad (2)$$

The posterior probability $P(D)$ describes the likelihood of being the target center for every location D in the context region [33]. When D_0 is located, $P(D)$ is defined as (Similar to k_1, k_2 is a normalization constant, γ for a scale factor, and β for a shape parameter):

$$P(D) \triangleq k_2 e^{|\frac{D - D_0}{\gamma}|^\beta}. \quad (3)$$

The conditional probability $P(D|(I(D_i), D_i)$ models the spatial relationship between the target location D and its context information. It is expressed as $P(D|(I(D_i), D_i)) \triangleq O_{sc}(D - D_i) = O_{sc}(\alpha)$, which can be learned from the prior and the posteriori probability.

To update the spatio-temporal context model $O_{stc}(\alpha, t+1)$ in the $(t+1)$-th frame, we weight $O_{stc}(\alpha, t)$ and $O_{sc}(\alpha, t)$ with a weighting factor ρ, that is,

$$O_{stc}(\alpha, t+1) = (1 - \rho) O_{stc}(\alpha, t) + \rho O_{sc}(\alpha, t), \quad (4)$$

where $O_{stc}(\alpha, t)$ refers to the spatio-temporal context model and $O_{sc}(\alpha, t)$ refers to the spatio context model in the t-th frame.

A. TARGET TRACKING BASED ON CONTEXT INFORMATION

Fig. 3 illustrates the tracking process in the $(t+1)$-th frame.The main task for tracking is to learn $O_{sc}(\alpha, t)$ and $O_{stc}(\alpha, t+1)$ according to the target location $D_0(t)$ and its context $U_c(t)$. Then, tracking is formulated by obtaining the posterior probability distribution of the target location, and the point that maximizes the probability is taken as the target center.

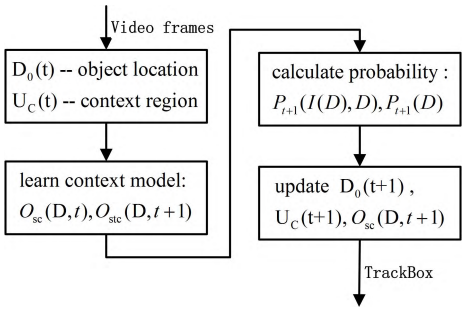


FIGURE 3. Tracking process in the $(t+1)$-th frame.

Putting equation (1), (2) and (3) together, equation (1) is formulated as

$$P(D)= k_2 e^{|\frac{D-D_0}{\gamma}|^\beta} = O_{sc}(D) \otimes (I(D) \cdot k_1 e^{-\frac{|D-D_0|^2}{\sigma^2}}),$$

where $\otimes$ denotes the convolution operator. Using fast fourier transformation (FFT) algorithm, $O_{sc}(D)$ can be solved as

$$O_{sc}(D) = F^{-1} \frac{F(k_2 e^{|\frac{D-D_0}{\gamma}|^\beta})}{F(I(D) \cdot \omega(D-D_0))}, \quad (5)$$

where F denotes the FFT function. Assembling equation (1) and (4) with the learned $O_{sc}(D)$, the posterior probability in the $(t+1)$-th frame is formulated as

$$P_{t+1}(D) = F^{-1}\left(F(O_{stc}(D, t+1)) \cdot F(I(D) \cdot \omega(\bullet))\right). \quad (6)$$

The best target location $D_0(t+1)$ is estimated by maximizing $P_{t+1}(D)$, and relevantly used to update the context region $U_c(t+1)$ and spatial context model $O_{sc}(D, t+1)$.

It is important to note that the tracker will necessarily fail if the target is completely occluded or moves out of the view boundary. To identify these occasions, the tracker is supposed to be extended with the ability of failure detection, which is realized by following method. Firstly, we obtain the Euclidean distance between the predicted $U_c(t+1)$ in current frame and the actual $U_c(t)$ in previous frame. If it is larger than a settled threshold d, that is,

$$|U_c(t+1) - U_c(t)| > d, \quad (7)$$

tracking is considered as a failure. Secondly, the confidence of tracking *tconf* is obtained by evaluating the similarity between the tracking result and target model library of detector. Similarity between two patches p_i, p_j is define as $S(p_i, p_j) = 0.5(k_n(p_i, p_j) + 1)$, where $k_n(p_i, p_j)$ is a Normalized Cross-correlation coefficient [34]. If *tconf* is less than settled threshold, tracker is also considered as unreliable and returns no tracking box. In this way, tracker is able to confirm the absence of target caused by occlusion or moving out of view.

B. TARGET DETECTION BASED ON CONTEXT INFORMATION

According to the size of the initial tracking box, the detector generates 21 bounding boxes of different scales with step of 1.2. Each of the boxes traverses the whole image with horizontal step of width's 10 percent and vertical step of height's 10 percent, thus producing all image patches that may contain the target.

If the tracker fails, the detector has to perform a global search in all image patches. Otherwise, the posterior probability $P_t(D)$ derived in tracking can be used to narrow the search of detector. To be specific, the detector acquires the median value P_m of $P_t(D)$ as

$$P_m = Median(P_t(D)), \quad (8)$$

and afterwards limit the search scope to patches whose posterior probability at center is larger than P_m.

The detector is realized by a cascade classifier, which is constituted of three stages: variance classifier (VC), ensemble classifier (EC), and nearest neighbor classifier (NNC). Fig. 4 shows the block diagram of the detector. The input image patche is firstly resampled to a normlized resolution, and then the three classifiers respectively reject the patch or pass it to the next stage.

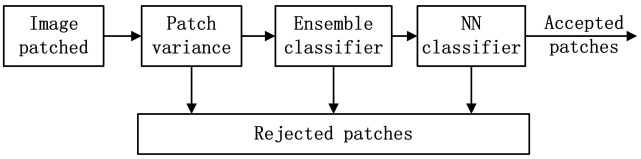


FIGURE 4. Block diagram of detector.

VC is the first stage of the cascade. It computes the image variance for input patches through integral image algorithm [35], and filters out all patches whose gray-value variance is smaller than 50 percent of variance of the patch in the initial tracking box.

EC is the second stage of the cascade and composed of n base classifiers which are based on a set of binary features [36], [37]. To extract features, each base classifier performs a set of pixel comparisons on the patch respectively. The discrete pixel coordinates of pixel comparisons are generated randomly in initialization and allocated averagely into base classifiers and remain unchanged in runtime. Each comparison returns 0 or 1 and these measurements are concatenated into a binary code x which indexes to an array of posterior probability distributions $P_i(y|x)$, where $y \in \{0, 1\}$. If there are k pixel comparisons performed by each base classifier, the distribution would have 2^k entries. The posteriors are initialized as zero and estimated as $P_i(y|x) = \frac{P_C}{P_C+N_C}$, where P_C and N_C respectively stand for the count of positive and negative patches that are assigned the same binary code x in learning process. EC eliminates all patches whose average posterior of individual base classifiers is less than 0.5.

NNC is used to classify the patches that have passed the first two stages. It separately obtain their similarity with positive and negative samples in the target model library by means of normalized cross-correlation method [34], thereby determining the classification and relevant confidence of the patches. Also, NNC is applied to evaluate the confidence of tracker's result *tconf*.

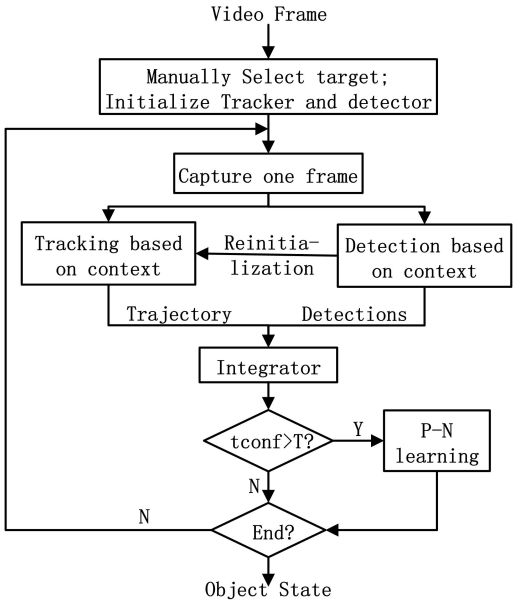


FIGURE 5. Flow chart of the proposed algorithm.

C. INTEGRATING AND TARGET LEARNING

The tracker and the detector operate in parallel and output bounding boxes of target individually. The tracker outputs only one box in a frame, where detector may localize several ones. The targetive of the integrator is to obtain a optimal bounding box from the results of the tracker and the detector. To this end, integrator clusters the output boxes of detector into several classes and assesses their confidence *dconf* through NNC of detector. If there exists a class with *dconf* higher than *tconf*, integrator will take it as the final output regardless of tracker's result; Otherwise, integrator determines the final output by weighting tracker's result and the class with the highest *dconf*.

The learning process is designed to evaluate the error of detector by using P-N learning algorithm [32], and generate labeled samples to update its EC and NNC in runtime. When trajectory is fairly reliable, that is, the confidence *tconf* is larger than settled threshold, it can be used as the basis of samples selection. For EC, samples are selected from the patches in the neighbourhood of integrator's output result. If EC classifies these samples incorrectly, the corresponding P_C and N_C are updated, which consequently updates $P_i(y|x)$. For NNC, samples includes all the patches that passed the EC. Similarly, the samples that are classified incorrectly by NNC will be added into the target model library. In this way, tracker will bring in new data for detector whenever it finds new appearances of the target.

D. THE PROPOSED ALGORITHM

The flow chart of the proposed algorithm is illustrated in Fig. 5, and the algorithm flow is described as follows:

Manually select the target, and initialize the detector and the tracker in the first frame;

1). The tracker obtains the target location by utilizing local spatio-temporal information and evaluates its confidence *tconf* in NNC of detector;

2). When the tracker fails, detector performs a global search for targets and reinitialize tracker; otherwise it performs a local search based in context region of tracker; detector clusters its outputs into several classes and assesses their confidence *dconf* through NNC;

3). The integrator merges the trajectory and detections together through an optimal strategy and eventually produces a best bounding box; if integrator has no outputs, it directly jumps to 1);

4). If *tconf* is larger than threshold T, the learning process generates new training samples to update EC and NNC of detector according to the output results of the tracker and detector. Otherwise, it will be skipped;

5). If the video doesn't end, it goes to the next frame and the procedure returns to 1).

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. EXPERIMENTAL RESULTS

In this section, the proposed algorithm is evaluated by four typical public benchmark videos, which contain scenes of various challenging factors, such as disappearance, scale and pose variations of target, drastic illumination changes, severe occlusion, and so on. Table 1 shows the detailed information of benchmarks(''Frames'' stands for the total frame number of of a sequence, ''OP'' for the number of frames where the target presents, ''OCC/DIS'' for the number of frames with occlusion/disappearance, ''I/P/S'' for whether there exists illumination/position/scale changes in a sequence), and the regions inside bounding boxes in the screenshots presents the target we need to track.

TABLE 1. Detailed information of benchmark videos.

Sequence	Frames	OP	OCC/DIS	I/P/S
1. David	761	761	0/0	Y/Y/Y
2. Pedestrian	184	156	0/28	N/Y/Y
3. Car	945	945	85/0	N/Y/Y
4. Plane	782	782	290/0	N/Y/Y
5. Plane with occlusion	1262	1170	92/92	N/Y/Y

We compare our method with other latest algorithms, namely, TLD [31], STC [23], CT [38] and WMT [28]. Some experimental results of these methods are illustrated in Fig. 6-Fig. 9. Additionally, the frame numbers of screenshots appear in the upper left corner, and the presence of a colored dot indicates that the corresponding methods returned nothing in the frame.

Fig. 6 displays the tracking results on the *David* sequence, which contains drastic illumination changes (See #100, #194, #283, #338, #393) as well as gradual pose and scale variations

FIGURE 6. Screenshots of tracking results in the video *David*.

FIGURE 7. Screenshots of tracking results in the video *Pedestrian*.

FIGURE 8. Screenshots of tracking results in the video *Car*.

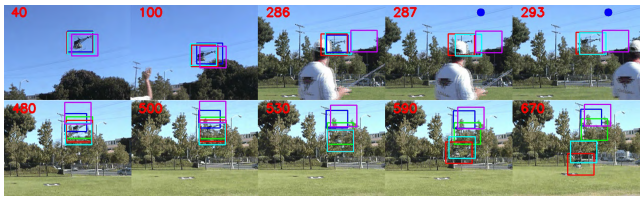


FIGURE 9. Screenshots of tracking results in the video *Plane*.

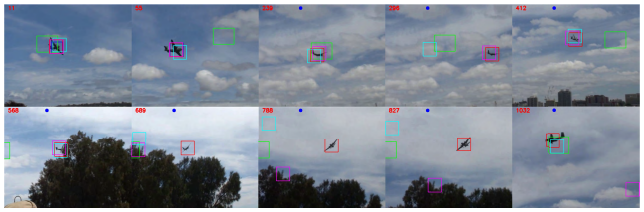


FIGURE 10. Screenshots of tracking results in the video *Plane with occlusion*.

(See #440, #455, #473, #500). STC, CT and proposed method perform favorably on this sequence, and WMT gradually deviates from the target, whereas TLD either loses the target or returns wrong bounding boxes in most of the time.

The tracking results on the *Pedestrian* sequence are shown in Fig. 7. The target disappears in #54 frame and reappears in #80 frame (See #54, #57, #80) due to the camera motion. Owing to the capability of detection, TLD and the proposed are able to recapture the target target when it appears again, but the others fail in tracking as long as the target disappears. Besides, our method does not export any bounding box due to its ability of effective failure detection.

Fig. 8 shows the experiment results on the *Car* sequence, which contains partial or several occlusion and pose variation at times. The target partly moves out of the view boundary in #40 frame, and all methods are able to track successfully. In the subsequence, the target is severely occluded by trees between #505 frame and #570 frame (See #512, #520, #544, #550) and experiences partial occlusion with pose variation (See #689, #788). In this situation, only the proposed method achieves favorable performance during most of the periods.

In the *Plane* sequence, the target undergoes in-plane rotation and temporary occlusion in the cluttered background, which is shown in the Fig. 9. As it is close to the cluttered background in #100 frame, TLD and WMT intend to deviate from the target. There follows temporary occlusion in the subsequence (See #286, #287). As a result, only STC and our method perform well while the others drift to background. Afterwards, the target encounters the cluttered background with in-plane rotation. Since the texture is pretty similar to that of the target (See #480, #500, #530, #590, #286, #287), most trackers fail to track the target. Since the tracker of our method brings in new appearance information of local context background and update detector constantly by learning. As a consequence, our method can stably track the target.

The tracking results on the *Plane with occlusion* sequence are shown in Fig. 10. The plane target undergoes rotation and scaling changes before the occlusion occurs. Both the proposed method and the WMT method can stably track the target. When the target disappears in #568 frame and reappears in #689 frame, the others fail in tracking as long as the target disappears, only the proposed method can re-track the target stably(See #788, #827, #1032).

From the experimental results above, it can be seen that our method has ability of failure detection and a strong recoverability after the target disappeared. Furthermore, it is still able to track the target stably even in complex environments with a variety of challenges, such as drastic illumination, pose variation, several occlusion and cluttered background.

B. QUANTITATIVE ANALYSIS

In this section, our method is quantitatively evaluated in terms of success rate (SR) of tracking and center location error (CLE). The tracking in one frame is considered as a success if its overlap degree (OD) is larger than 0.5, where OD is formulated as

$$(ROI_t \cap ROI_{gt})/(ROI_t \cup ROI_{gt}), \tag{9}$$

where ROI_t and ROI_{gt} respectively denote a tracked bounding box and a ground truth one. SR is defined as $SR = SF/TF$, where SF denotes the number of successful frames and TF denotes the total number of frames in the sequence. CLE

TABLE 2. Success Rate (SR)(%) of each method on different video sequences.

	TLD [31]	STC [23]	CT [38]	WMT [28]	Proposed
1	15.4	**97.5**	61.8	65.8	76.1
2	**78.8**	29.5	25.5	45.2	75.5
3	89.3	71.4	43.7	32.8	**98.0**
4	48.5	66.5	52.7	13.3	**85.2**
4	32.6	37.4	43.1	85.3	**93.8**

TABLE 3. Center Location Error (CLE)(pixels) of each method on different video sequences.

	TLD [31]	STC [23]	CT [38]	WMT [28]	Proposed
1	20.0	**4.8**	12.5	14.5	12.7
2	**4.8**	24.0	53.2	11.1	7.8
3	6.1	23.8	62.4	35.8	**5.3**
4	22.6	20.8	14.3	55.7	**7.2**
5	52.5	19.2	10.7	8.9	**4.6**

refers to the pixel distance between the centers of ROI_t and ROI_{gt}.

1) COMPARISON OF TACKING RATES

Considering the randomness of the algorithms, we repeat experiments on each video sequence many times, and then take the experimental results in average. Table 2 shows SR of each method on different benchmarks. The bold figures indicate the highest SR in each sequence.

Compared with TLD, the SR of our method is slightly lower in sequence 2, but certainly much higher in other sequences. Since the sequence 2 contains disappearance of target, STC and CT can not redetect the target after it disappeared, resulting in a much lower SR. Compared with STC, our method obtains a lower SR in sequence 1 due to the illumination changes that have adverse effects on appearance-based detector.Meanwhile, our method performs better than the other methods in all benchmarks.

Generally, our method achieves the best performance in two sequences and the second best in others, thus achieving the best comprehensive performance. This can be ascribed to the utilization of context information of target. Despite the appearance of target changes significantly due to various factors, most of the local context surrounding has little change over succeeding frames, which helps to predict the target location in the next frame.

2) COMPARISONS OF TACKING ERROR

Table 3 shows CLE of each method on different benchmarks. As it is with the Table 2, the italic figures with underline denote the best performance in each sequence. It can be seen from Table 2 and Table 3 that, there exist a negative correlation between SR and CLE. Compared with the experiment results of [39], our proposed method also shows excellent performance.

In order to directly show the changes of tracking error in tracking process, the curve graphs of OD and CLE for TLD, STC, CT, WMT and our method are illustrated by using the experimental results in sequence 3 and 4, which relatively contain more frames challenges. Note that CLE and OD are respectively set to 150 and 0 when tracking fails according to the experimental results.

The curve graph of OD and CLE in sequence 3 is shown in Fig. 11. In interval 1 where the target is heavily occluded,

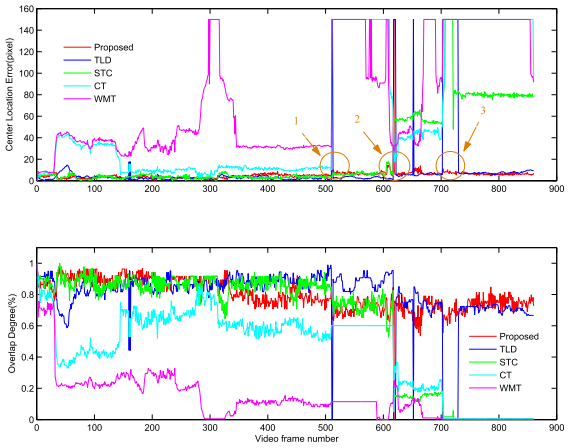


FIGURE 11. OD and CLE in *car* sequence.

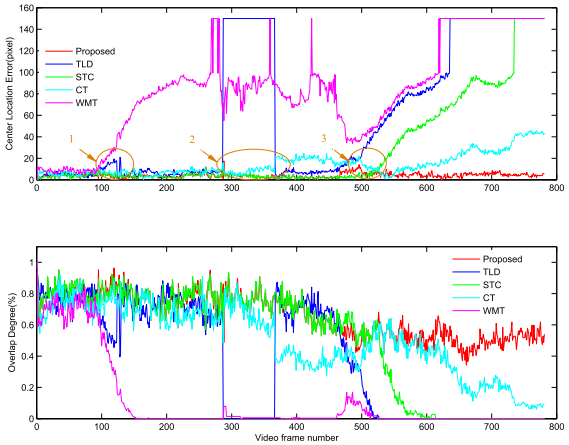


FIGURE 12. OD and CLE in *plane* sequence.

our method and STC can still track the target effectively with the tracking error increasing a little, but TLD comes to a failure due to the occlusion. In interval 2 and 3, the target is temporarily occluded with pose variation. As a result, STC, CT and WMT almost lose the target with a large error, but our method are able to acquire the target again after occlusion with a smaller fluctuation of error than TLD.

Fig. 12 shows the changes of OD and CLE in sequence 4. The target is approaching the cluttered background in interval 1, giving rise to an increased tracking error of TLD. Similarly, TLD fails to track in interval 2 on account of severe

occlusion while STC and our method are able to track accurately. Afterwards, the target begins to enter into cluttered background in interval 3. Affected by this, TLD, STC and CT start to drift to the background and the tracking error of them increase evidently, and the error of WMT is relatively small. Overall, our method performs robustly in the whole process of tracking despite the minor fluctuation in interval 3.

In summary, our method has higher success rate as well as smaller tracking error than TLD, STC, CT and WMT at the same time, and performs remarkably in complex environments that are characterized by various challenges.

V. CONCLUSION

In this paper, we investigated the problem of long-term tracking of an unknown target in a video stream. Aimed at improving the robustness of target tracking algorithms in complex environment, this work proposes the moving target detection and tracking algorithm based on context information. It mainly decomposes the long-term tracking task into four parts of synchronous operation: tracking, detection, integration and learning, which are described in previous sections. Five typical public benchmark videos are selected to evaluate the proposed algorithm, which is compared to other latest algorithms. Experimental results demonstrate that the proposed method not only has strong recoverability after target disappeared, but also tracks effectively and robustly in complex environment with dramatic lighting changes, severe occlusion, cluttered background and so on. On the other hand, it is noted that the learning process only updates detector while tracker is kept invariant in the framework. Hence, in future work we plan to investigate the update of tracker in long-term tracking process.

REFERENCES

[1] J. Li and J. Wang, "Adaptive object tracking algorithm based on eigenbasis space and compressive sampling," *IET Image Process.*, vol. 6, no. 8, pp. 1170–1180, Nov. 2012.

[2] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 564–577, May 2003.

[3] Y. Ren, C.-S. Chua, and Y.-K. Ho, "Color based tracking by adaptive modeling," in *Proc. 7th Int. Conf. Control, Autom., Robot. Vis. (ICARCV)*, vol. 3, Dec. 2002, pp. 1597–1602.

[4] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 8, pp. 790–799, Aug. 1995.

[5] J. Yang, K. Sim, X. Gao, W. Lu, Q. Meng, and B. Li, "A blind stereoscopic image quality evaluator with segmented stacked autoencoders considering the whole visual perception route," *IEEE Trans. Image Process.*, vol. 28, no. 3, pp. 1314–1328, Mar. 2019. doi: 10.1109/TIP.2018.2878283.

[6] J. Yang, K. Sim, W. Lu, and B. Jiang, "Predicting stereoscopic image quality via stacked auto-encoders based on stereopsis formation," *IEEE Trans. Multimedia*, to be published. doi: 10.1109/TMM.2018.2889562.

[7] A. Yilmaz, X. Li, and M. Shan, "Contour-based object tracking with occlusion handling in video acquired using mobile cameras," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 11, pp. 1531–1536, Nov. 2004.

[8] G. Paschos, "Perceptually uniform color spaces for color texture analysis: An empirical evaluation," *IEEE Trans. Image Process.*, vol. 10, no. 6, pp. 932–937, Jun. 2001.

[9] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artif. Intell.*, vol. 17, nos. 1–3, pp. 185–203, Aug. 1981.

[10] L. Jing, W. Junzheng, and W. Lipeng, "On vision servo tracking and control based on multi-cue adaptive integration," in *Proc. 31st Chin. Control Conf. (CCC)*, Jul. 2012, pp. 3681–3685.

[11] B. Jiang, J. Yang, Q. Meng, B. Li, and W. Lu, "A deep evaluator for image retargeting quality by geometrical and contextual interaction," *IEEE Trans. Cybern.*, to be published. doi: 10.1109/TCYB.2018.2864158.

[12] H. Rezaee, A. Aghagolzadeh, and H. Seyedarabi, "Vehicle tracking by fusing multiple cues in structured environments using particle filter," in *Proc. IEEE Asia Pacific Conf. Circuits Syst. (APCCAS)*, Dec. 2010, pp. 1001–1004.

[13] D. Salmond, "Target tracking: Introduction and Kalman tracking filters," in *Proc. IEEE Target Tracking, Algorithms Appl.*, vol. 2, Oct. 2002, pp. 1-1–1-16.

[14] A. E. Nordsjo, "A constrained extended kalman filter for target tracking," in *Proc. IEEE Radar Conf.*, Apr. 2004, pp. 123–127.

[15] P. Pérez, J. Vermaak, and A. Blake, "Data fusion for visual tracking with particles," *Proc. IEEE*, vol. 92, no. 3, pp. 495–513, Mar. 2004.

[16] D. Shi, T. Chen, and L. Shi, "An event-triggered approach to state estimation with multiple point- and set-valued measurements," *Automatica*, vol. 50, no. 6, pp. 1641–1648, 2014.

[17] D. Shi, T. Chen, and L. Shi, "On set-valued Kalman filtering and its application to event-based state estimation," *IEEE Trans. Autom. Control*, vol. 60, no. 5, pp. 1275–1290, May 2015.

[18] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool, "Robust tracking-by-detection using a detector confidence particle filter," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Oct. 2009, pp. 1515–1522.

[19] H. Grabner, J. Matas, L. Van Gool, and P. Cattin, "Tracking the invisible: Learning where the object might be," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 1285–1292.

[20] L. Wen, Z. Cai, Z. Lei, D. Yi, and S. Z. Li, "Robust online learned spatio-temporal context model for visual tracking," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 785–796, Feb. 2014.

[21] T. B. Dinh, N. Vo, and G. Medioni, "Context tracker: Exploring supporters and distracters in unconstrained environments," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 1177–1184.

[22] B. Jiang, J. Yang, Z. Lv, and H. Song, "Wearable vision assistance system based on binocular sensors for visually impaired users," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1375–1383, Apr. 2019. doi: 10.1109/JIOT.2018.2842229.

[23] K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M.-H. Yang, "Fast visual tracking via dense spatio-temporal context learning," in *Proc. 13th Eur. Conf. Comput. Vis.*, vol. 8693, Sep. 2014, pp. 127–141.

[24] O. Williams, A. Blake, and R. Cipolla, "Sparse Bayesian learning for efficient visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1292–1304, Aug. 2005.

[25] C. Leistner, H. Grabner, and H. Bischof, "Semi-supervised boosting using visual similarity learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–8.

[26] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 983–990.

[27] B. Zeisl, C. Leistner, A. Saffari, and H. Bischof, "On-line semi-supervised multiple-instance boosting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, Jun. 2010, p. 1879.

[28] K. Zhang and H. Song, "Real-time visual tracking via online weighted multiple instance learning," *Pattern Recognit.*, vol. 46, no. 1, pp. 397–411, Jan. 2013.

[29] A. Saffari, C. Leistner, J. Santner, M. Godec, and H. Bischof, "On-line random forests," in *Proc. IEEE 12th Int. Conf. Comput. Vis. Workshops (ICCV Workshops)*, Sep./Oct. 2009, pp. 1393–1400.

[30] A. Wang, G. Wan, Z. Cheng, and S. Li, "An incremental extremely random forest classifier for online learning and tracking," in *Proc. 16th IEEE Int. Conf. Image Process. (ICIP)*, Nov. 2009, pp. 1449–1452.

[31] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1409–1422, Jul. 2012.

[32] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: Bootstrapping binary classifiers by structural constraints," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 49–56.

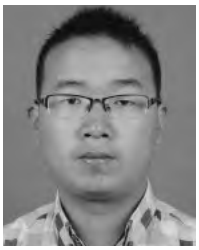
[33] K. Bai, "Adaptive confidence map fusion in visual object tracking," in *Proc. Int. Conf. Inf. Eng. Comput. Sci.*, Dec. 2009, pp. 1–4.

[34] J. Wu, H.-J. Yue, Y.-Y. Cao, and Z.-M. Cui, "Video object tracking method based on normalized cross-correlation matching," in *Proc. 9th Int. Symp. Distrib. Comput. Appl. Bus. Eng. Sci. (DCABES)*, Aug. 2010, pp. 523–527.

[35] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Dec. 2001, pp. I-511–I-518.
[36] A. Bosch, A. Zisserman, and X. Munoz, "Image classification using random forests and ferns," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.
[37] M. Ozuysal, P. Fua, and V. Lepetit, "Fast keypoint recognition in ten lines of code," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.
[38] K. Zhang, L. Zhang, and M. Yang, "Fast compressive tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 10, pp. 2002–2015, Oct. 2014.
[39] X. Tian, H. Li, and H. Deng, "Object tracking algorithm based on improved context model in combination with detection mechanism for suspected objects," *Multimedia Tools Appl.*, pp. 1–16, Jan. 2019.

JING LI was born in 1982. She received the M.S. degree in engineering from the Shandong University of Technology, in 2007, and the Ph.D. degree in engineering from the Beijing Institute of Technology, in 2011, where she is currently an Associate Professor with the School of Automation. Her research interests include image detection technology, and target detection and tracking.

JUNZHENG WANG was born in 1964. He received the M.S. and Ph.D. degrees in engineering from the Beijing Institute of Technology, in 1990 and 1994, respectively, where he is currently a Professor and a Tutor of Ph.D. Student of the School of Automation. His current research interests include motion drive and control, image detection and tracking, and the static and dynamic performance testing control systems.



WENXUE LIU was born in China, in 1992. He received the B.S. degree from the Beijing Institute of Technology, in 2017. His research interest includes target detection and tracking.