Combining Spatio-temporal context and kalman filtering for visual tracking

Yang, Haoran; Wang, Juanjuan; Miao, Yi; Yang, Yulu; Zhao, Zengshun; Wang, Zhigang; Sun, Qian; Wu, Dapeng Oliver

Published in:
Mathematics

Published: 01/11/2019

Document Version:
Final Published version, also known as Publisher’s PDF, Publisher’s Final version or Version of Record

License:
CC BY

Publication record in CityU Scholars:
Go to record

Published version (DOI):
10.3390/math7111059

Publication details:

Citing this paper
Please note that where the full-text provided on CityU Scholars is the Post-print version (also known as Accepted Author Manuscript, Peer-reviewed or Author Final version), it may differ from the Final Published version. When citing, ensure that you check and use the publisher's definitive version for pagination and other details.

General rights
Copyright for the publications made accessible via the CityU Scholars portal is retained by the author(s) and/or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights. Users may not further distribute the material or use it for any profit-making activity or commercial gain.

Publisher permission
Permission for previously published items are in accordance with publisher's copyright policies sourced from the SHERPA RoMEO database. Links to full text versions (either Published or Post-print) are only available if corresponding publishers allow open access.

Take down policy
Contact lbscholars@cityu.edu.hk if you believe that this document breaches copyright and provide us with details. We will remove access to the work immediately and investigate your claim.

Download date: 20/05/2024
Combining Spatio-Temporal Context and Kalman Filtering for Visual Tracking

Haoran Yang 1, Juanjuan Wang 1, Yi Miao 1, Yulu Yang 1, Zengshun Zhao 1,2,3,* Yulun Yang 1, Zengshun Zhao 1,2,3,*, Zhigang Wang 4, Qian Sun 1,* and Dapeng Oliver Wu 3

1 College of Electronic and Information Engineering, Shandong University of Science and Technology, Qingdao 266590, China; yohor.yanghaoran@gmail.com (H.Y.); wangjuanjuan6@163.com (J.W.); miao_y_i@163.com (Y.M.); shujin03@163.com (Y.Y.)
2 School of Control Science & Engineering, Shandong University, Jinan 250061, China
3 Department of Electrical& Computer Engineering, University of Florida, Gainesville, FL 32611, USA; dpwu@ufl.edu
4 Key Laboratory of Computer Vision and System, Ministry of Education, Tianjin Key Laboratory of Intelligence Computing and Novel Software Technology, Tianjin University of Technology, Tianjin 300384, China; wangzgtj@gmail.com

*Correspondence: zhaozengshun@sdust.edu.cn (Z.Z.); sunqian940411@163.com (Q.S.)

Received: 29 September 2019; Accepted: 30 October 2019; Published: 5 November 2019

Abstract: As one of the core contents of intelligent monitoring, target tracking is the basis for video content analysis and processing. In visual tracking, due to occlusion, illumination changes, and pose and scale variation, handling such large appearance changes of the target object and the background over time remains the main challenge for robust target tracking. In this paper, we present a new robust algorithm (STC-KF) based on the spatio-temporal context and Kalman filtering. Our approach introduces a novel formulation to address the context information, which adopts the entire local information around the target, thereby preventing the remaining important context information related to the target from being lost by only using the rare key point information. The state of the object in the tracking process can be determined by the Euclidean distance of the image intensity in two consecutive frames. Then, the prediction value of the Kalman filter can be updated as the Kalman observation to the object position and marked on the next frame. The performance of the proposed STC-KF algorithm is evaluated and compared with the original STC algorithm. The experimental results using benchmark sequences imply that the proposed method outperforms the original STC algorithm under the conditions of heavy occlusion and large appearance changes.

Keywords: spatio-temporal context algorithm; Kalman filter; object detection; target tracking

1. Introduction

While target tracking is one of the noteworthy and active research areas in the field of computer vision and machine learning, many challenges are still not resolved [1].

Researchers have proposed many different tracking algorithms for possible occlusion, illumination changes, and pose variation during target tracking. Most of these algorithms adopt template matching [2,3], small facet tracking [4,5], particle filtering [6,7], sparse representation [8,9], contour modeling [10], and image segmentation [11]. For low resolution, target occlusion, deformation, and other complex scenes, how to achieve more robust tracking is still the current research focus.

The past decades saw an increase in academic interest in the Kalman filter [12], which integrates the promotion of the tracking algorithm. In order to enhance the stability of the Kalman filter algorithm in the target tracking process, Pouya et al. [13] proposed a tracking algorithm based on Kanade–Lucas–Tomasi (KLT) and Kalman filtering, using KLT to track targets and estimating the
tracking results of KLT with the Kalman filter algorithm. Wang and Liu [14] improved a tracking method based on the target texture features. In their method, the algorithm estimated the pose of the target in the current frame, and predicted the pose of the target in the next frame using the Kalman filter algorithm. Fu and Han [15] proposed a linear Kalman filter algorithm, which firstly adopted the background difference method to search for moving objects, and then exploited the centroid weighting method in Kalman. Wu et al. [16] introduced the normalized moment of inertia in the traditional mean shift algorithm, and used the Kalman filter to predict and estimate the target occlusion.

In recent years, the tracking framework based on the particle filter was found to be fast and effective, attracting the attention of many researchers. Particle filters (PFs) are recursive implementations of Monte Carlo methods and are ideal for analyzing highly non-linear, non-Gaussian state estimation problems where classical Kalman filter-based approaches fail [17]. Su et al. [18] boosted the visual significance model and combined it with the particle filter algorithm to solve the problem of sudden movement of the target. Liu et al. [19] put forward a tracking algorithm that is suitable for the rapid change of the target pose, which was robust to targets with large deformation and partial occlusion. Yang et al. [20] came up with a new dynamic maneuvering target model, which effectively solved the problems caused by inaccurate state models. Hu et al. [21] developed an improved resampling cellular quantum-behaved particle swarm optimization (RScQPSO) algorithm, which was a probabilistic variant of PSO, and combined it with the PF to solve the tracking problem. Sengupta and Peters [22] constructed an evolutionary particle filter with a memory-guided proposal step size update and an improved quantum-behaved particle swarm optimization (QPSO) resampling scheme for visual tracking.

When the spatio-temporal context (STC) algorithm was proposed [23], it had the advantage of speed of detection by utilizing fast Fourier transform (FFT). Subsequently, the local context, which consists of the target and its immediate surrounding background pixels, played a vital role in image processing in recent years. The strong spatio-temporal relationships between the local scenes containing the object in consecutive frames facilitate the visual tracking. This is the basic idea of STC tracking. However, the STC tracking method cannot deal with the model drift problem, which means that the target model is potentially wrongly updated after long-term occlusions. The Kalman filter can be efficiently applied to predict the state of an object to solve occlusion problems. However, when the target is severely occluded, it is difficult for the above tracking algorithm to ensure effective tracking.

Consequently, in this paper, we propose an improved spatio-temporal context tracking (STC-KF) algorithm based on a Kalman filter combined with spatio-temporal context (STC) tracking. Our approach introduces a novel formulation to address context information, which adopts the entire local information around the target, thereby preventing the remaining important context information related to the target from being lost by only using the rare key point information. In addition, the correlation between the target and the local context information is constantly updated through the learning of the spatio-temporal context model, while utilizing Kalman prediction can effectively reduce the adverse effects of noise during tracking. Experiments showed that the algorithm can show a good tracking effect when large poses and contour changes are produced in the face of the target, or when it is partially occluded.

The rest of the paper is summarized as follows: Section 2 reviews the principles of the spatio-temporal context tracking algorithm and Kalman filtering. Section 3 details the proposed approach. Section 4 elaborates on the experimental conditions and the results for the benchmark problems. Section 5 concludes the paper with possible directions for future work.

2. The Principle of Spatio-Temporal Context Tracking Algorithm and Kalman Filtering

2.1. The Basic Principle of STC Algorithm

The STC algorithm is an effective target-tracking algorithm proposed by Zhang [23]. The core idea of the STC algorithm is utilizing the target appearance model obtained in the image to acquire a spatio-temporal context model through online learning and then using the spatio-temporal context model to calculate the confidence map to obtain the most likely location of the target. There exists
a very strong spatio-temporal relationship between the object and its local context. As shown in Figure 1, the region inside the yellow rectangle is the target for tracking, while the pixels inside the red rectangle are the context information, which includes the target immediate surrounding background. Moreover, the regions inside the blue rectangles represent the learned spatio-temporal context model. The spatio-temporal context can be divided into the spatial component and the temporal component. The spatial component represents a specific relationship between the target and the background around the target. When the appearance of the target changes significantly, this relationship can help distinguish the target from the background. Furthermore, the temporal component denotes that the appearance of the target does not change very sharply from two consecutive video sequences frames. The target appearance will change largely under the circumstances from heavy occlusion. However, the local context [24] containing the target has not changed much, as the entire appearance of the red box remains similar, and it occludes only a small part of the context region. Therefore, the presence of local context information on the current frame is worthwhile to predict the target location in the next frame.

Figure 1. The illustration of the target context. (a) The definition of local context. (b) The context under heavy occlusion.

In the target tracking, the assumed target location in the initial frame has been initialized manually or detected by the certain object detection algorithms. Subsequently, we learn the spatial context model, which is applicable to update the spatio-temporal context model and to detect the object location in the next frame, and then to calculate the confidence map [25] of the frame target using the spatio-temporal context algorithm. The most important step is taking advantage of the Fourier transform and the inverse transforms to find the probability of the target spatial context conditions in the next frame. Then, the confidence map is found by convolving the conditional probability with the prior probability. The maximum value of the confidence map is the target location of the next frame [23]. Figure 2 shows the basic structure of the STC algorithm.

Figure 2. The structure of the spatio-temporal context (STC) algorithm.
In reference [22], the spatio-temporal context model is utilized as the filter in each convolutional neural network. In the initial frame, the target confidence map is exploited to update the spatio-temporal model. The target tracking problem can be described as calculating the confidence map size of tracking target location \( x \). The context feature set is defined as \( X_c = \{ c(z) = (I(z), z | z \in \Omega_c(x^*) ) \} \), where \( I(z) \) denotes the image intensity at location \( z \) and \( \Omega_c(x^*) \) is the neighborhood of location \( x^* \) (i.e., the coordinate of the tracked object center).

\[
\begin{align*}
\begin{array}{l}
\sum_{c(z) \in X} P(x,c(z)|\omega) = \sum_{c(z) \in X} P(x|c(z),\omega)P(c(z)|\omega)
\end{array}
\end{align*}
\]

where \( x \in R^2 \) is an object location and \( \omega \) denotes the object present in the scene.

From Equation (1), the confidence map consists of two parts: the conditional probability of the contextual spatial relationship \( P(x,c(z) | \omega) \) and the context prior probabilities of each point \( x \) in the local area \( P(c(x) | \omega) \).

### 2.1.1. Spatial Context Model

The conditional probability function \( P(x,c(z) | \omega) \) in Equation (1) is defined as:

\[
\begin{align*}
P(x,c(z) | \omega) = h_{sc}(x-z)
\end{align*}
\]

where \( h_{sc}(x-z) \) is a function with respect to the relative distance and direction between the target location \( x \) and its local context location \( z \), thereby encoding the spatial relationship between the target and its spatial context [26].

### 2.1.2. Context Prior Model

In Equation (1), the context prior probability is simply modeled by the following formulation:

\[
\begin{align*}
P(c(z) | \omega) = I(z)w_\sigma(z-x^*)
\end{align*}
\]

where \( I(z) \) is the grayscale features of point \( z \) in the local context of the target, and \( \omega \) is a weighted function defined by:

\[
\begin{align*}
w_\sigma(z) = ae^{-\frac{|z|^2}{\sigma^2}}
\end{align*}
\]

where the smaller the distance from \( z \) to \( x \), the greater the \( \omega \) value. The parameter \( \sigma \) is the variance of the Gaussian weight function, which determines the distance threshold. The larger the value of \( \sigma \) is, the wider the field of view will be.

### 2.1.3. Confidence Map

The confidence map of the target location is modeled as:

\[
\begin{align*}
c(x) = P(x|\omega) = be^{-\frac{|x-x^*|^2}{\beta}}
\end{align*}
\]

where \( b \) is a normalization constant, \( \alpha \) is a scale parameter, and \( \beta \) is a shape parameter. After verification by experiments, the optimal tracking effect is shown when \( \beta = 1 \).

### 2.1.4. Fast Learning Spatial Context Model

Our objective is to learn the spatial context model in Equation (2) based on the context prior model in Equation (3) and the confidence map shown in Function (5). Putting Equations (2), (3), and (5) together, Equation (1) becomes:
$$c(x) = be^{-|\frac{z-x}{\rho}|^{\beta}} = \sum_{z \in \Omega_c(x')} h^{sc}(x-z)I(z)\omega_{l}(z-x^*)$$  \hspace{1cm} (6)$$

where \( c(x) \) is only related to the relative distance of the target neighborhood point \( x \) to the target position \( x' \), and \( \otimes \) denotes the convolution operator.

Equation (6) can be transformed to the frequency domain so that the fast Fourier transform (FFT) algorithm can be utilized for fast convolution as the following formulation.

$$F\left(be^{-|\frac{z-x}{\rho}|^{\beta}}\right) = F(h^{sc}(x) \circ F(I(x)\omega_{l}(x-x^*)))$$  \hspace{1cm} (7)$$

where \( F \) denotes the FFT function and \( \circ \) is the element-wise product.

As long as the size of the neighborhood box is determined, the confidence graph is a constant matrix that is converted to the frequency domain for calculation and obtaining the spatial context model \( h^{sc}(x) \) [26]. The inverse Fourier transform are performed on Equation (7) to get the spatial context model.

$$h^{sc}(x) = F^{-1}\left(\frac{F\left(be^{-|\frac{z-x}{\rho}|^{\beta}}\right)}{F(I(x)\omega_{l}(x-x^*))}\right)$$  \hspace{1cm} (8)$$

where \( F^{-1} \) denotes the inverse FFT function.

We exploit the spatial context model to update the spatio-temporal context model as follows:

$$H^{stc}_{t+1} = (1 - \rho)H^{stc}_t + \rho h^{sc}_t$$  \hspace{1cm} (9)$$

where \( \rho \) is the learning parameter and \( h^{sc}_t \) is the spatial context model computed by Equation (8) at the the \( t \)-th frame. Function (9) is the temporal filtering procedure, which can be easily observed in the frequency domain:

$$H^{stc}_\omega = F_{\omega}h^{sc}_\omega$$  \hspace{1cm} (10)$$

where \( H^{stc}_\omega \triangleq \int H^{stc}_t e^{-j\omega t}dt \) is the temporal Fourier transform of \( H^{stc}_t \) and similar to \( h^{sc}_\omega \). The temporal filter \( F_{\omega} \) is set as:

$$F_{\omega} = \frac{\rho}{e^{j\omega} - (1 - \rho)}$$  \hspace{1cm} (11)$$

where \( j \) denotes the imaginary unit. It is easy to validate that \( F_{\omega} \) in Equation (11) is a low-pass filter [27,28].

2.1.5. Target Tracking

When the \((t+1)\)-th frame arrives, we crop out the local context region \( \Omega_c(x'_t) \) based on the tracked location \( x'_t \) at the \( t \)-th frame and construct the corresponding context feature set \( X^{c}_{t+1} = \{c(z) = (I_{t+1}(z), z)|z \in \Omega_c(x'_t)\} \).

The object location \( x'_{t+1} \) in the \((t+1)\)-th frame is determined by maximizing the new confidence map:

$$x'_{t+1} = \arg\max_{x \in \Omega_c(x'_t)} c_{t+1}(x)$$  \hspace{1cm} (12)$$

where \( c_{t+1}(x) \) is represented as:

$$c_{t+1}(x) = F^{-1}(F\left(H^{stc}_{t+1}(x)\right) \circ F(I_{t+1}(x)\omega_{l}(x-x^*))$$  \hspace{1cm} (13)$$
2.1.6. The Scale and Variance are Updated as

\[
\begin{align*}
    s'_{t_i} &= \sqrt{c_t(x^*_{t_i})} \\
    s_t &= \frac{1}{n} \sum_{i=1}^{n} s'_{t_i} \\
    s_{t+1} &= (1 - \lambda)s_t + \lambda s_t \\
    \sigma_{t+1} &= \sigma_t
\end{align*}
\]

(14)

where \(c_t(\cdot)\) is the confidence map that is computed by Equation (6), and \(s'_{t_i}\) is the estimated scale between two consecutive frames. Aiming to avoid oversensitive adaptation and to reduce the noise introduced by the estimation error, the estimated target scale \(s_{t+1}\) is obtained through filtering in which \(s_t\) is the average of the estimated scales from \(n\) consecutive frames, and \(\lambda > 0\) is a fixed filter parameter.

2.2. The Basic Principle of Kalman Filtering Algorithm

The Kalman filter assumes that the system noise and the observed noise are white noise [27]. The system state equation of the discrete dynamic equations for the non-linear systems is defined as:

\[
X_k = f(X_{k-1}) + W_{k-1}
\]

(15)

The measurement equation is set as:

\[
Z_k = h(X_k) + V_k
\]

(16)

where \(X_k \in \mathbb{R}^{nx1}\) is the target state vector, \(Z_k \in \mathbb{R}^{mx1}\) is the observation vector of dimension \(m\), \(W_k \in \mathbb{R}^{px1}\) is the state noise, \(V_k \in \mathbb{R}^{mx1}\) is the observation noise. \(W_k\) and \(V_k\) are the independent Gaussian self-noise vector sequences that are not related to each other.

\[
h(X_k) = \begin{bmatrix} r_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} \sqrt{x^2(k) + y^2(k)} \\ \arctan\left(\frac{y(k)}{x(k)}\right) \end{bmatrix}
\]

(17)

where \(h(X_k)\) is the measured equation. The next state prediction is updated by Equation (18):

\[
\hat{X}_{k|k-1} = F_{k|k-1}\hat{X}_{k-1|k-1}
\]

(18)

and the prediction variance matrix is denoted as:

\[
P_{k|k-1} = F_{k|k-1}P_{k-1|k-1}F_{k|k-1}^T + Q_{k-1}
\]

(19)

where \(F_{k|k-1}\) is the state transition matrix, as shown in Equation (29), \(\hat{X}_{i|k-1}\) is the predicted state estimator, \(P_{k|k-1}\) is the prediction estimation covariance, and \(Q_k\) is the covariance matrix.

The dynamic noise variance matrix \(Q_k\) is expressed by:

\[
Q_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\]

(20)

The filter gain matrix is signified as:

\[
K_k = P_{k|k-1}H_k^T[H_kP_{k|k-1}H_k^T + R_k]^{-1}
\]

(21)
where $R_k$ is the measurement of the noise variance matrix.

\[
R_k = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}
\]  

(22)

where $K_k$ is the optimal Kalman gain. The state estimator is represented as:

\[
\mathbf{x}_{k|k} = \mathbf{x}_{k|k-1} + K_k [Z_k - \hat{Z}_{k|k-1}]
\]

(23)

\[
\hat{Z}_{k|k-1} = h(\hat{X}_{k|k-1})
\]

(24)

which is deduced from Equation (18). The estimated error variance matrix is formulated as:

\[
P_{k|k} = [I - K_k H_k] P_{k|k-1}
\]

(25)

where $\hat{X}_{k|k}$ is the current updated state estimate, and $P_{k|k}$ is the updated covariance estimate.

3. STC-KF Target Tracking Algorithm

In the target tracking, the rectangular regions of the tracking object are manually marked in the initial frame, and the confidence map of the frame target is calculated by the spatio-temporal context algorithm. The spatio-temporal context algorithm, as a fast method, mainly adopts fast Fourier transform to calculate each local context region. However, its drawback is that it can easily cause the drift problem. Depending on the advantages and disadvantages of the spatio-temporal context and the Kalman filter algorithm, an improved algorithm combines these two algorithms, which is denoted as STC-KF. Using Fourier transform and inverse transform, we find out the probability of the spatial context condition of the target in the next frame. Then, tracking is conducted in the next frame by calculating the confidence map as a convolution problem, which incorporates the spatio-temporal context information, and the best target location can be estimated by maximizing the confidence map or by the prediction value of the Kalman filter. Specifically, the Kalman filter utilizes the state transition matrix to determine the predicted value of the status with the achieved system status of the $(k - 1)$-th frame, and the state transition matrix is also relevant to the state estimation at the $k$-th frame. The prediction value of the Kalman filter can be updated as the Kalman observation of the object position and marked on the next frame. Figure 3 is the algorithm flowchart of the STC-KF algorithm.

![Figure 3](image-url)  

Figure 3. The implementation process of the spatio-temporal context and Kalman filtering (STC-KF) algorithm.
Based on the estimation function in Equation (25) and the spatio context prior model in Equation (9), our objective is to improve the tracker. It can be formulated as:

$$H_{t+1}^{stick} = (1 - \rho)H_t^{stick} + \rho h_{sc}^{st}(x) + P_{kj}$$  \hspace{1cm} (26)

The Gaussian function is introduced into the context prior model in Equation (3) as:

$$P(c(z)|\theta) = I(z) \frac{1}{2\pi \sigma^2} e^{-\frac{|z|^2}{2\sigma^2}}$$  \hspace{1cm} (27)

where $I(z)$ is the image intensity of the $z$ point. The parameter $\sigma$ is the variance of the Gaussian weight function, which determines the distance threshold.

Therefore, our spatio-temporal context model can theoretically effectively filter out the image noise introduced by the appearance variations, thereby leading to more stable results.

When the target is severely obstructed, the Euclidean distance between the grayscales of the $t$-th frame and the $(t+1)$-th frame can be calculated by Equation (28), and the result can be measured as a judgment of whether the target is occluded:

$$d_E(I_t(z), I_{t-1}(z)) = \sqrt{\sum_{r \in r} (I_t(z_i) - I_{t-1}(z_i))^2}$$  \hspace{1cm} (28)

When $d_E$ is greater than 17% of the target area and the target center is unchanged but the target is occluded, the program begins with Kalman filter prediction [28].

The Jacobian matrix is shown as:

$$F_{kj} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$  \hspace{1cm} (29)

The measurement equation is updated as:

$$m_k = \begin{bmatrix} r_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} \sqrt{(s_x)^2 + (s_y)^2} \\ \arctan(s_y/s_x) \end{bmatrix}$$  \hspace{1cm} (30)

where $r_k$ is the scope of the observation and $\theta_k$ is the observation angle. According to Equation (29), the Jacobian matrix that measures the updated equation is:

$$H_k = \frac{\partial (m_k)}{\partial X_k} \bigg|_{X_k} = \begin{bmatrix} \cos(\theta_k) & \sin(\theta_k) & 0 & 0 \\ -\sin(\theta_k)/r_k & \cos(\theta_k)/r_k & 0 & 0 \end{bmatrix}$$  \hspace{1cm} (31)

After updating the error variance matrix in Equations (25) through (31), we can locate the position of the target in each frame by Equation (26), which efficiently reduces the risk of missing the target and improves the stability for tracking.

### 4. Experimental Results and Analysis

We evaluate the proposed STC-KF tracking algorithm using three representative benchmarks of OTB50/100. In the entire tracking experiment, the platform used a Windows 7 operating system, 3G memory, and 2.20 GHz computer, and was simulated on Matlab2014a platform.
4.1. Database Introduction

OTB50, also named as OTB2013 [29], is a performance evaluation database supplied by CVPR2013. The database OTB 100, also known as database OTB2015 [30], is given in CVPR2015. OTB50 (2013) and OTB100 (2015) respectively include a large dataset with ground-truth object positions and extents for tracking experiments. Specifically, OTB100 contains the video sequences of OTB50, but these two databases are considered as two different video sequences due to different labeling objects. The full benchmark contains 100 sequences from the recent literature. The video set used in this experiment includes three scenes: scene Car with occlusion, scene David with illumination changes, and scene Motocross with pose and contour variation. Among these three scenes, there are 450 frames in Car with the resolution of 290 × 217, 770 frames in David with the resolution of 320 × 240, and 190 frames in Motocross with the resolution of 470 × 310.

4.2. Experimental Results

4.2.1. Scene with Occlusion Condition

In the experiment based on the Car dataset, the position of the initial frame of the target is marked as (140,90,55,31).

As shown in Figure 4, before the 90th frame, the car can be tracked normally, and the STC algorithm can correctly track the target without blocking or drifting. However, when tracking to frame 91 and frame 92, due to the fast-moving speed of the target, the tracking target frame started to shift to the incorrect position. Correspondingly, from frame 97 to frame 105, the target frame continuously missed the correct target. At frame 105, the target box completely missed the target and the target remained in the lost state.

Figure 5 depicts the tracking effect of the STC-KF algorithm on the Car dataset, where the yellow rectangle and red rectangle indicate the target location obtained by the STC-KF algorithm of the STC algorithm and the Kalman filter, respectively. In frame 91 and frame 92, when the background of the car changed drastically, the STC-KF algorithm could normally track the target, which was better than the original STC algorithm. Besides, from frame 165 to 169, the STC-KF algorithm can correctly predict the target when the object was occluded, which solved the target occlusion problem. From frame 358 to frame 365, although the target was in the low light intensity background, the algorithm in this paper still depicted a robust tracking effect.

![Figure 4. Target tracking results under occlusion condition via the original STC algorithm, where the red rectangle represents the target position.](image-url)
4.2.2. Scene with Illumination Changes Condition

In the experiment based on the David dataset, the position of the initial frame of the target is marked as (161, 65, 75, 95).

It can be seen from the comparison between Figures 6 and 7 that in the 50th frame, both algorithms can track the target in the case of low illumination intensity, but the STC algorithm contains too many redundant regions. Thus, at 100 frames, tracking the face frame was slightly offset; at frame 150, the target can be tracked correctly, but the tracking frame was slightly offset. The STC-KF algorithm was more accurate in tracking the face area of the target, and fewer redundant areas, which indicated that the STC-KF algorithm is superior.

4.2.3. Scene with Pose and Contour Variation Condition

In the experiment based on the Motocross dataset, the position of the initial frame of the target is marked as (288, 313, 36, 78).

Figure 8 depicts the simulation of the Motocross by the STC algorithm. The STC algorithm cannot track the target correctly due to the target vertical gap between two adjacent frames being too large. Figure 9 reveals the simulation of the motocross by the STC-KF algorithm. Indeed, the tracking effect
was slightly better than the STC algorithm, but there were occasions when the tracking target was occasionally lost by these two consecutive frames, which have great variation in pose and contour. However, in the vertical direction, the STC-KF algorithm can determine the target position more effectively than the STC algorithm.

![Figure 8. Target tracking results under pose and contour variation condition via the original STC algorithm, where the red rectangle represents the target position.](image)

![Figure 9. Target tracking results under pose and contour variation condition via the STC-KF algorithm, where the yellow rectangle and red rectangle indicate the target location obtained by the STC-KF algorithm of the STC and the Kalman filter, respectively.](image)

5. Performance Analysis

Table 1 shows the correct number of target tracking frames for the STC algorithm and the STC-KF algorithm in the Car experiment. From the table, we can see that in terms of the number of correct tracks frames, the accuracy of the STC-KF algorithm (88.3%) is significantly higher than the accuracy of the STC algorithm (22.2%). Obviously, the proposed algorithm achieves the better performance in terms of success rate.

<table>
<thead>
<tr>
<th>Algorithm Name</th>
<th>Video Name</th>
<th>Number of Frames</th>
<th>Correctly Track Frames</th>
</tr>
</thead>
<tbody>
<tr>
<td>STC</td>
<td>car</td>
<td>410</td>
<td>91</td>
</tr>
<tr>
<td>KF-STC</td>
<td>car</td>
<td>410</td>
<td>362</td>
</tr>
</tbody>
</table>

We can analyze the advantages of our algorithm from the position of the center point.

Table 2 shows the location information of the David video of each target by the STC algorithm and the STC-KF algorithm. According to Table 2, the average error of the STC algorithm is 5.87 pixels, and the average error of the STC-KF algorithm is 2.83 pixels. This shows that the algorithm of this paper is more effective and accurate than the original STC algorithm in target tracking.

<table>
<thead>
<tr>
<th>Number of Frames</th>
<th>STC Tracking Coordinates</th>
<th>STC-KF Tracking Coordinates</th>
<th>STC Pixel Error</th>
<th>STC-KF Pixel Error</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(120,62)</td>
<td>(124,63)</td>
<td>(121,61)</td>
<td>4.1</td>
</tr>
<tr>
<td>100</td>
<td>(164,63)</td>
<td>(169,68)</td>
<td>(160,60)</td>
<td>7.1</td>
</tr>
<tr>
<td>150</td>
<td>(142,44)</td>
<td>(144,46)</td>
<td>(143,42)</td>
<td>2.8</td>
</tr>
<tr>
<td>200</td>
<td>(128,31)</td>
<td>(136,31)</td>
<td>(129,29)</td>
<td>8</td>
</tr>
<tr>
<td>250</td>
<td>(180,74)</td>
<td>(180,70)</td>
<td>(182,73)</td>
<td>4</td>
</tr>
<tr>
<td>300</td>
<td>(126,60)</td>
<td>(120,55)</td>
<td>(130,59)</td>
<td>9.2</td>
</tr>
</tbody>
</table>
In view of the shortcomings of the STC algorithm in the tracking process, this paper combines the STC and the Kalman filter to form the STC-KF algorithm; this algorithm mainly solves the problems of pose and contour variations and the target occlusion. We compare the proposed algorithm of this paper with the original STC algorithm, and the experiments on challenging video sequences show that the proposed STC-KF algorithm achieves favorable performance in terms of accuracy, robustness, and speed.

6. Conclusions

In this paper, we presented a fast and robust algorithm that combines the STC and the Kalman filter to form the STC-KF algorithm. The algorithm mainly solved the problems related to heavy occlusion, illumination changes, and pose and contour variation. By tracking experiments, compared with the STC algorithm, the STC-KF algorithm is robust to severe occlusion, self-position change, and illumination intensity change under the premise of ensuring tracking accuracy. Consequently, the target tracking performance of the proposed algorithm under occlusion condition is superior to that of the STC algorithm.

However, in view of the inaccuracy of tracking in the experiment, there is still room to advance in the tracking effect of strenuous moving targets in serious pose and contour variation condition. In recent years, some sophisticated algorithms have adopted detection algorithms [12,31] or deep learning algorithms [32–34], and future work can focus on those algorithms to further improve robustness and tracking accuracy.


Funding: This research was funded by the National Natural Science Foundation of China, grant number 61403281; the Natural Science Foundation of Shandong Province, grant number ZR2014FM002; China Postdoctoral Science Special Foundation Funded Project, grant number 2015T80717; Youth Teachers’ Growth Plan of Shandong Province, grant number 201701.

Acknowledgments: Many thanks for the reviewer and the editors for the insightful comments and suggestions.

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

References


15. Fu, Z.X.; Han, Y. Centroid weighted Kalman filter for visual object tracking. *Measurement* 2012, 45, 650–655. [CrossRef]


© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).