Improved TLD Algorithm for Face Tracking

Huimin Li [a], Chaojing Yu [b] and Jing Chen [c]

Chongqing University of Posts and Telecommunications, Chongqing 400065, China

[a]li.huimin666@163.com, [b]15023299065@163.com, [c]727252429@qq.com

Abstract

For the Tracking-Learning-Detection (TLD) for dynamic face tracking has the problem of tracking drift when the face is occluded seriously, this paper proposed an improved method based on TLD. After locating the face successfully, introducing the method of occlusion window to determine the degree of face occlusion. When the face is occluded slightly, using the original learner to solve it. If the face is occluded seriously, adding the direction predictor of Markov model in TLD detector to reduce the detection range and enhance the discrimination ability for similar faces. At the same time, adding Kalman filter in TLD tracker to estimate the area in current frame where the face may exist, which could reduce the number of scanning grids and improve the processing speed. The experiments show that the improved TLD algorithm has a strong robustness in different environments, can track the dynamic face quickly and accurately, and also improve the problem of tracking drift when the face is occluded seriously

Keywords

Tracking-Learning-Detection (TLD), Markov model, Kalman filter, face tracking.

1. Introduction

The problems of the increasing aging populations and children safety custody have become serious, so the tutelar home-service robots have received more and more attention these years [1]. Adding the technology of face tracking in home-service robot to realize security monitoring and human-computer interaction is an important research direction in service robot field.

Tracking-Learning-Detection (TLD) is first proposed by Zdenek kalal in 2010 as single target long term tracking algorithm [2]. TLD has good performance in real-time and stability under normal circumstances so that it attracted extensive attention in related fields. There were some improved algorithms in the basis of TLD. Dong Yongkun [3] had made some improvements to the TLD detector, using Histogram of Oriented Gridient (HOG) and incremental classifier based on Normalized Cross Correlation method (NCC) replaced the TLD detector to detect the target. Cheng et al. [4] has added Random Sample Consensus (RANSAC) in the TLD tracker to estimate the global motion model and improved the successful rate of tracking.

This paper proposed an improved face-tracking algorithm based on TLD to solve the tracking drift problem which caused by serious occlusion. A lot of experiments show that the improved algorithm proposed in this paper can solve the above problems and track the face accurately and quickly.

2. TLD algorithm

TLD is a tracking algorithm which is made up primarily of tracker, detector and learner. Fig. 1 is the framework of TLD. According to the principle of TLD, supposing the target is visible between two adjacent frames so we can estimate the target's motion state by the information of motion. The learner updates the tracker and the detector constantly through P-N learning mechanism, and assesses the detection error and update detector's target model according to the tracking result [5]. The detector searches each frame in total graph to locate the target, which is obtained according the previous learning and detection. Significantly, the detector and tracker run at the same time to determine the location of the target. To verify the accuracy of the target information, the data obtained from detector and tracker is fed back to the learner [6-7].

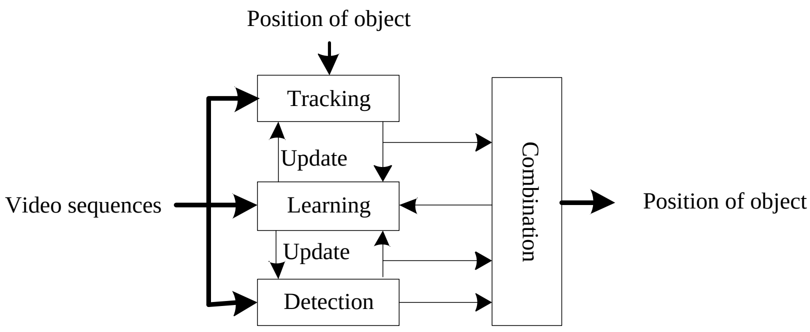


Fig.1 The framework of original TLD algorithm

3. Tracking face based on improved TLD

When the face is occluded, it is easy to have tracking drift problem for original TLD algorithm. So an improved method based on TLD would be introduced to solve this problem. To determine the degree of occlusion of the face, Occlusion Window is proposed. And then Markov prediction model [8-9] and Kalman filter [10] are added in TLD detector and tracker to deal with different occlusion situations. The operation procedure of the improved TLD tracking algorithm is exhibited in Fig. 2.

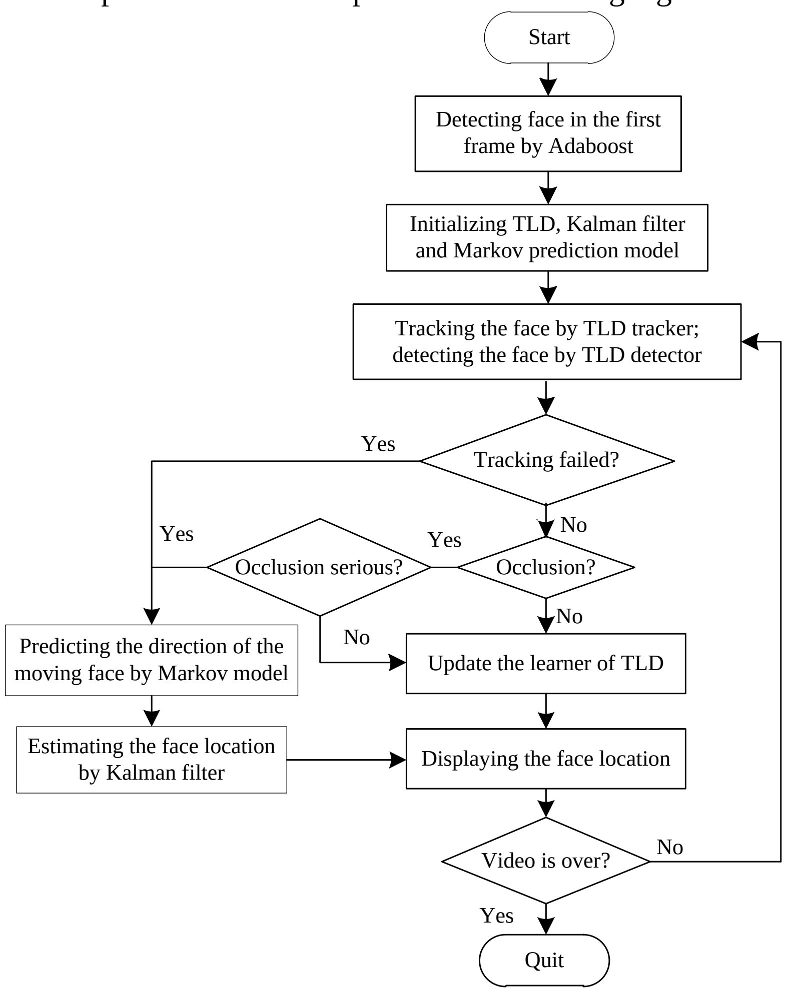


Fig.2 the flowchart of tracking face based on improved TLD

3.1 The determination of the occlusion degree of the face

The setting method of Occlusion Window is showed in the Fig. 3, which represent the situations that objects occluding face enter the window form horizontal and vertical direction. In the two pictures, yellow bold rectangles are tracking windows, and blue serif rectangles are Occlusion Windows introduced in this paper. The red lines in the middle of Occlusion Windows divide windows into two symmetrical parts of the A and A\`.

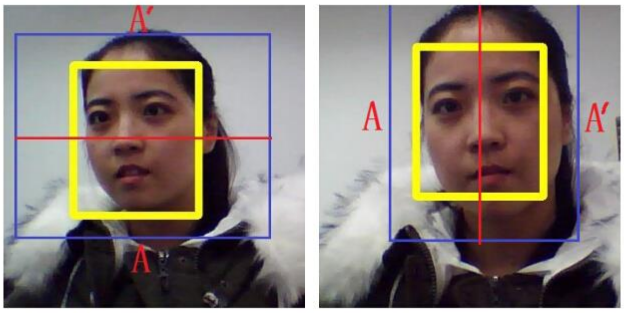


Fig.3 The settings of the occlusion window

While the occluded objects entering from different directions and occluding the faces, color histograms will be calculated corresponding the upper-down parts. And the normalization results are marked $T(k,t)$, which represents the histograms of tracking windows and Occlusion Windows. We suppose that the occluded objects enter Occlusion Window from the A side in the $t-1$th frame and enter tracking window in the t th frame to start to occlude the face. The similarity of the histograms of A and A\`'s tracking window and Occlusion Windows can be calculated at the $t-1$th and t th frame by using Bhattacharyya coefficient. The calculation formulas are showed as (1), (2), (3).

$$\rho_{TR_1(k,t)} = \sum \sqrt{T(k,t)\cdot \mathrm{R}(k,t)} \tag{1}$$

$$\rho_{TR_2(k,t)} = \sum \sqrt{T(k,t)\cdot \mathrm{R}(k,t-1)} \tag{2}$$

$$\rho_{TR_3(k,t)} = \sum \sqrt{T(k,t)\cdot T(k,t-1)} \tag{3}$$

As shown above, it corresponds A or A\` part when k equals -1 or 1. When k equals -1, $\rho_{TR_1(k,t)}$ denotes the similarity of the histogram of A's tracking window and occlusion window at t; $\rho_{TR_2(k,t)}$ denotes the similarity of the histogram of A's tracking window and occlusion window at t and $t-1$. $\rho_{TR_3(k,t)}$ denotes the similarity of the histograms of A 's tracking window at t time and A 's tracking window at $t-1$. Correspondingly, the tracking window and Occlusion Window represent the part of A' when k equals 1.

$$\rho_{TR_1(k,t)} < \rho_{TR_2(k,t)} \tag{4}$$

$$\rho_{TR_3(k,t)} \leq \quad (0.5 \leq \lambda \leq 0.7) \tag{5}$$

We define when Eq. (4) is satisfied only, the face is occluded lightly; when Eq. (4) and Eq. (5) are both satisfied, the face is occluded badly. In order to improve the robustness of the occlusion judgment, we will repeat at continuous 10 frames to determine whether it meat these formulas.

3.2 The prediction of face-motion direction by Markov model

It is possible to appear some similar faces in the target area and overlapped each other on the move. In this situation, it is hard for the original TLD to track the correct face because of its poor resolution with similar faces. To solve this problem, we add Markov model in the TLD detector to estimate the motion direction of the face.

The face motion is decomposed into vertical direction and horizontal direction. Take vertical direction for example, we define 1 represents left movement and -1 represent right movement.

According to the direction predictor of Markov model, we suppose the motion state of the face in current frame only have relation with the state transition matrix and the motion state in the previous frame. The state transition matrix at time *t* can be obtained from the Eq. (6)

$$P_t = \begin{bmatrix} p(s_{t+1}=-1 \mid s_t=-1) & p(s_{t+1}=-1 \mid s_t=1) \\ p(s_{t+1}=1 \mid s_t=-1) & p(s_{t+1}=1 \mid s_t=1) \end{bmatrix} \tag{6}$$

s_t denotes the motion state of the face at t time. By means of the state transition matrix and motion state of the face at t time, the motion state of the face at time t+1 can be estimated by Eq. (7):

$$\begin{bmatrix} p(s_{t+1}=-1) \\ p(s_{t+1}=1) \end{bmatrix} = P_t \begin{bmatrix} p(s_t=-1) \\ p(s_t=1) \end{bmatrix} \tag{7}$$

In the formula, $p(s_t=-1)$ represents the probability of the face moving left at t time and $p(s_t=-1)$ represents the probability of the face moving right at t time. At t time, if the face move left, $p(s_t=-1)=1$. Else the result equals 0. At t time, $p(s_t=-1)=0.5$ if the face dose not move. The state transition matrix can be calculated by the Eq. (8) according to the face motion state.

$$P_t = \begin{bmatrix} p(s_{t+1}=-1 \mid s_t=-1) = \dfrac{n_{-1,-1}}{n_{-1}} & p(s_{t+1}=-1 \mid s_t=1) = \dfrac{n_{-1,1}}{n_1} \\ p(s_{t+1}=1 \mid s_t=-1) = \dfrac{n_{1,-1}}{n_{-1}} & p(s_{t+1}=1 \mid s_t=1) = \dfrac{n_{1,1}}{n_1} \end{bmatrix} \tag{8}$$

Where, n_{-1} (n_1) denotes the number of the frames while the face moving left (right) during 0~*t* time. $n_{-1,-1}$ ($n_{1,1}$) represents the number of the frames while the face moving left (right) both in the current and the previous frame during 0~t time. $n_{-1,-1}$ ($n_{1,1}$) represent the number of the frames while the face moves left (right) in current frame but moving right (left) in previous frame. In this paper, the difference of the center position of the face in the vertical between adjacent frames is calculated to estimate the face's moving direction.

On the basis of the Markov model, we can estimate the direction of the face moving in the current frame. According to the actual position in the last frame, the detection area of the face in the current frame can be obtained. And those faces which appear in this detected area meet the conditions of Markov model. So we can exclude the interference and overlapping problem of other faces. Fig. 4 represents that Markov predicted the movement direction of face. In this picture, the position of the face in the previous frame is represented by the yellow rectangle. And by the use of the Markov model, we can forecast the face has a tendency to move to the right in the current frame. So on the right side of the red thread is the part which is sent to the TLD detector. Conversely the left part is no longer sent to the TLD detector. So we can narrow the detection area.

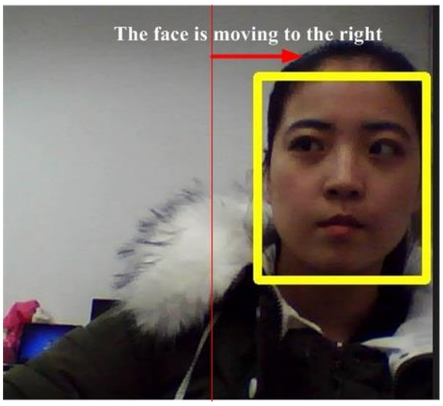


Fig.4 Markov predicted the movement direction of face

3.3 The estimation to the face location by Kalman filter

When the state of the face satisfies Eq. (4), we determine it as slight occlusion and send it to the TLD learner. When it satisfies both Eq. (4) and Eq. (5), we determine it as serious occlusion and add Kalman filter in TLD tracker to predict the position of the face in the current frame. With these operations, the search area can be narrowed, and the processing time can be depressed.

The central position of the face in each frame is obtained to construct the Kalman filter's observation matrix and state matrix, which are showed in Eq. (9) and Eq. (10):

$$O_k = \begin{bmatrix} c_x & c_y \end{bmatrix}^T \tag{9}$$

$$S_k = \begin{bmatrix} c_x & c_y & s_x & s_y \end{bmatrix}^T \tag{10}$$

In these formulas, c_x , c_y denote the coordinate components of the face center in horizontal and vertical positions. s_x , s_y denote the speed of the moving face in horizontal and vertical direction. We suppose the face moving at a constant velocity between two adjacent frames because the adjacent frames are very closed. And its state transition matrix and observation matrix are showed as Eq. (11):

$$A_{\mathrm{k,k-1}} = \begin{bmatrix} 1 & 0 & t & 0 \\ 0 & 1 & 0 & t \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, H_{\mathrm{k}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{11}$$

Where, t denotes interval time between two adjacent frames.

The original TLD tracker needs to scan all the sub-windows in the image which may contain human faces to determine the position of the face. It is not efficient for the original TLD because it has not limit the search scope, which caused the sub scanning grids too much to reduce efficiency. To improve this problem, the Kalman filter is added into TLD tracker to predict the possible position of the face in current frame, reduce the number of scanning grids. Steps of the improved TLD proposed in this paper are elaborated as follows:

Step 1 Predict the central position of the face in current frame by using Kalman filter;

Step 2 A big rectangle, centering on this position, whose ratio of length to width is same with the face window in previous frame and its area is 6 times as much as the face window in previous frame, is defined as Search Scope(S-Scope). And S-Scope is the area where the face may appear in current frame.

Step 3 Send the sub-windows to TLD tracker, which are intersected with S-Scope, to determine if the face exists.

Fig. 5 represents the scanning grids are reduced by the use of Kalman filter. In this picture, the blue thick rectangular frame is the area which is obtained by Kalman filter and the face may appear in. We will send the rectangle 1, 3 intersected or contained by the blue rectangle to TLD detector. As opposed to rectangle 2, 4, 5. So we can reduce the number of scanning grids of TLD.

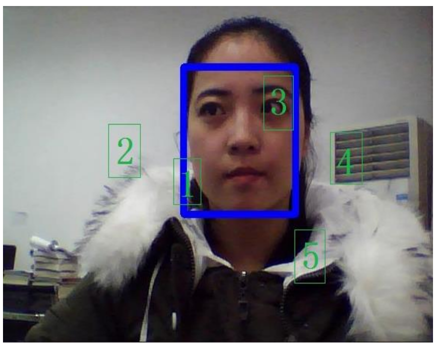


Fig.5 Kalman reduces the number of child windows

4. The experiment results and analysis

4.1 Robustness test

In order to verify the robustness and effectiveness of the proposed algorithm in different experimental environment, 3 methods are used to compare the proposed algorithm, including the original TLD algorithm, the TLD algorithm joining the Markov direction predictor only(M-TLD), the TLD algorithm joining the Kalman filter only(K-TLD), the improve TLD algorithm proposed in this paper (Imp-TLD).

The experiments are finished in 4 different environments as shown as Fig. 6. Label 1 represents normal illumination with simple background. Label 2 shows poor illumination with simple background. Label 3 shows normal illumination with complex background. And label 4 shows poor illumination with complex background. As shown below, table 1 denotes the number of frames containing faces in 4 cases and table 2 denotes the experimental by using 4 different algorithms in the four different environments.

Fig.6 Experimental environments

Table 1. The frames of the video images (frames)

Environment	Total frames	Face is occluded or disappear	Normal face
1	313	0	313
2	310	19	291
3	311	23	288
4	305	31	274

Table 2. Test results of the video

Number	The frames tracking Correctly				Frame rate (frames/s)			
	TLD	M-TLD	K-TLD	Imp-TLD	TLD	K-TLD	M-TLD	Imp-TLD
1	311	310	299	312	22.132	27.035	27.703	30.663
2	267	266	268	285	21.693	26.303	26.685	28.807
3	255	255	253	279	21.005	24.891	25.351	28.096
4	238	237	238	262	20.553	24.421	24.706	27.615

From table 2 we can see that the results of the 4 algorithms are similar in the normal illumination and simple background environment without occlusion. They all can track the face accurately. Comparatively, the original TLD has the problem of tracking drift in the normal or poor illumination environment with complex background. Under the same environment, M-TLD and K-TLD's successful tracking frames are less than the original TLD because of the reduction of search sub-windows. As for the frame rate, M-TLD and K-TLD have a higher processing speed than the original TLD. Imp-TLD proposed in this paper not only enhances the judgment to similar faces but also narrows the search area by introducing Occlusion Window, Markov prediction model and Kalman filter. So Imp-TLD has better performance than the other three algorithms synthetically. Through the analysis and comparison of the experimental data, we can prove that Imp-TLD is strongly robust to different environments, and it can track to face accurately and fast in the same time.

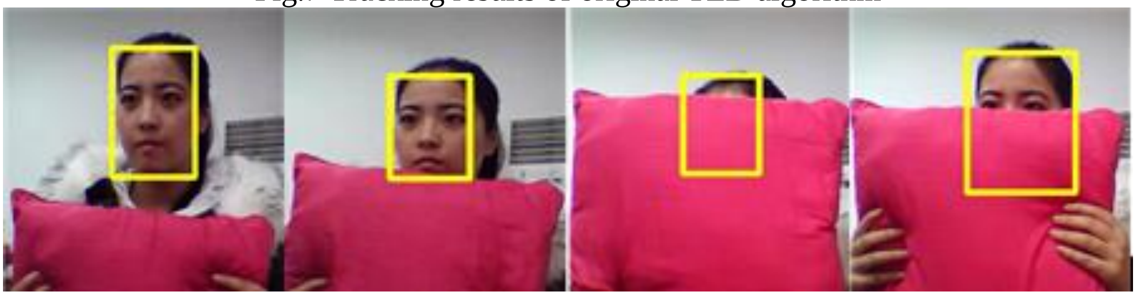
4.2 Occlusion test

In this paper, we have conducted occlusion test. Imp-TLD and the original TLD are performed under the complex environment with object interference to. Fig. 7 denotes the test result of the original TLD, and Fig. 8 denotes the result of Imp-TLD. In these figures, (a),(b),(c),(d) represent the face in the situations of no occlusion, slight occlusion, serious occlusion and the face reappearance, respectively.

In the repeated 40 experiments, the results are basically consistent with Fig. 7 and Fig. 8. As the results indicate, when the face is slightly occluded, both the original TLD and Imp-TLD can work successfully because the slightly occluded part is added to original TLD learner as a positive sample. However when the face is occluded seriously, the original TLD is very likely to track the face failed due to the model drifting. On the contrary, because of Markov prediction model added in the TLD detector and Kalman filter added to the TLD tracker. Imp-TLD has better performance in the situation that the face is occluded seriously.

(a) No occlusion (b) Slight occlusion (c) Serious occlusion (d) the face reappearance

Fig.7 Tracking results of original TLD algorithm



(a) No occlusion (b) Slight occlusion (c) Serious occlusion (d) the face reappearance

Fig.8 Tracking results of improved TLD algorithm

5. Conclusion

To solve the problem that it is likely to track the face failed when the face is occluded seriously for original TLD, a improved TLD algorithm is proposed for tracking face in this paper. After located the face, the degree of occlusion is evaluated by the Occlusion Window which is introduced in this paper. If the face is slightly occluded, the original TLD processes still. But if the face is occluded seriously, the direction predictor of Markov model is added to narrow search area and strengthen the resolution to similar faces. In the same time, Kalman filter is added in the TLD tracker to estimate the face's position in the next frame and reduce the running time. Through a lot of experiments, it is proved that Imp-TLD can track the face accurately and quickly. At the same time, Imp-TLD has good performance to deal with the tracking drift problem when the face is occluded seriously.

Acknowledgements

This work was supported by the Chongqing Science and Technology Commission (CSTC) (No. cstc2015jcyjBX0066).

References

[1] Xu F, Zhang X, Du Z. Home service robot industry development present situation investigation report of our country [J]. Robot Technique and Application, 2009, 02:14-19(in Chinese)

[2] Kalal Z, Matas J, Mikolajczyk K. P-N learning: bootstrapping binary classifiers by structural constraints[C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. New York: IEEE Press, 2010: 49-56

[3] Dong Y, Wang Chun-Xi, Xue L, et al. Pedestrian detection and tracking based on TLD framework [J]. Journal of Huazhong University of Science and Technology(Natural Science Edition), 2013,S1:226-228+232(in Chinese)

[4] Cheng S, Liu G W, Sun J X. Robust and fast Tracking-Learning-Detection. In: Proceedings of International Conference On Computer Science and Intelligent Communication (CSIC). Zhengzhou, china: ACSR, 2015. 449--452

[5] Zhou X, Qian Q, Ye Y, Wang Cong-Qing, Improved TLD visual target tracking algorithm [J]. Journal of Image and Graphics, 2013,09:1115-1123

[6] Chen L Y, Zhang D, Zhao S Y, et al. An Algorithm of Visual Tracking Based on Tracking-learning-detection [J]. Science Technology and Engineering, 2013, 9: 2382-2386(in Chinese)

[7] Gopalan R, Hong T, Shneier M, et al. A learning approach towards detection and tracking of lane markings [J]. IEEE transactions on intelligent transportation systems, 2012, 13(3): 1088-1098

[8] Lamberti R, Septier F, Salman N, et al. Sequential Markov Chain Monte Carlo for multi-target tracking with correlated RSS measurements [C], 2015 IEEE 10th International Conference on Intelligent Sensors, April 7-9, 2015, Singapore, 2015:1-6

[9] Liu Y L, Yu Y X. An Effective Method to Formulate State Transition Probability Matrix of Markov Model of Large-Scale System[J]. Journal of Tianjin University: Natural Science and Engineering Technology, 2013, 9:791-798(in Chinese)

[10] Jiang C, Zhang Y. Reduced-order Kalman filtering for state constrained linear systems [J]. Journal of Systems Engineering and Electronics, 2013, 4:674-682.