International Journal of Security and Its Applications
Vol. 10, No. 1 (2016), pp.361-374
http://dx.doi.org/10.14257/ijsia.2016.10.1.33

Eye Region Activity State based Face Liveness Detection System

Xu GuangZhu[1, 2], Yin PanLong[3], Lei BangJun[*2], Zou YaoBin[1, 2], Yang JiQuan[4]
[1]College of Computer Science and Information Technology, Three Gorges University, Yichang 443002, P.R.China
[2]Hubei Key Laboratory of Intelligent Vision Based Monitoring for Hydroelectric Engineering, Three Gorges University, Yichang 443002, P.R.China
[3]Edong Healthcare, Huangshi Central Hospital, Huangshi 435000, P.R. China
[4]Jiangsu key Laboratory for 3D Printing Equipment and Manufacturing, Nanjing 210046, P.R.China
xugzhu@163.com, panlonyin@163.com,bangjun.lei@ieee.com,

Abstract

In recent years, counterfeiting is posing a serious threat to face recognition based authentication system. Introducing liveness detection into a face-based authentication system can be a good solution to the problem. This paper applied eye region detection and tracking technology to anti-counterfeiting application, and designed an eye region activity based liveness detection system. First, a scale balance and a "face-to-eye" strategy were proposed to fast detect the eye region, and then KLT algorithm was used to track the detected eye region. At last, a random instruction (closing eye, moving to the left or right) was evoked to the user, and the liveness judgment was given according to the results of the users' cooperation. Experiments show that the proposed eye region activity based face liveness detection system is real-time and robust, and can meet the needs of practical applications.

Keywords*: Eye detection; Eye tracking; Scale balance; Liveness detection*

1. Introduction

In recent years, biometric authentication technologies become more mature and are widely used in real life scenarios. Commonly used biometrics technologies include voice, fingerprint, face, and behavior recognition [1-4]. Face recognition technology is becoming today's mainstream identity authentication technology and was widely used in many applications, which brought a lot of convenience to people's lives. But new challenges are also coming with it. For example in face recognition based online authentication system, counterfeiting and deception are posing serious threats to the security of authentication system.

In a face recognition based authentication system, commonly used cheating scenarios include: 1) Using the user's photo to cheat the authentication system. Counterfeiter can get photos from the true user's blog and his or her personal spaces, and then use them to deceive the authentication system. 2) Using the video clips of the true user to cheat the authentication system. A counterfeiter can get video clips via a candid way, and these video clips could be used to work as the true user logging in. It's more likely to pass the authentication system by using the candid videos. Aiming at the counterfeit deceptive problems in face recognition, face liveness detection is drawing more and more attentions.

Face liveness Detection is a technology that can effectively distinguish whether the current user is a counterfeiter (user's face photos or video clips) or not. This paper applied eye detection and tracking to face liveness detection, designed and implemented an eye

Lei BangJun is the Corresponding Author.

ISSN: 1738-9976 IJSIA
Copyright ⓒ 2016 SERSC

region activity based liveness detection system. The basic ideas of the system are as follows: first the eye region is detected and tracked, and then a random instruction (close eye, move left and move right) is evoked to the tracked eye region, according to the users' cooperation degree liveness judgment is given. Its specific process is shown in Figure 1. From Figure 1, we can see that eye region detection and tracking is the prerequisite for subsequent liveness detection, and determines whether the random instruction is executed correctly by the user is crucial in liveness detection.

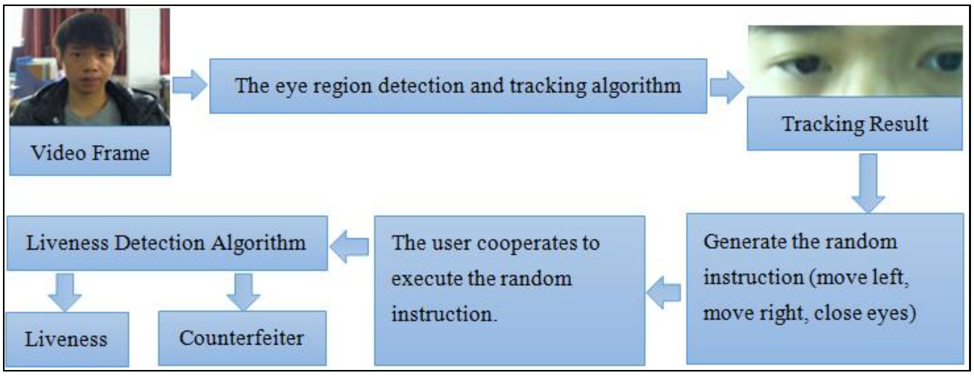


Figure 1. Flow chart of the Proposed Liveness Detection System

In order to solve the problem that for the low-quality image it is difficult to accurately detect the eye region and improve the real-time performance of the detection algorithm, on the basis of in-depth study of Viola-Jones algorithm [5, 6], this paper adopted a fast face detection algorithm based on the scale balanced strategy and an eye region detection algorithm based on face-eye combination strategy (described in section 2) [7]. And then rapid and robust eye region detection and tracking system combined with KLT tracking technique [8, 9] was designed in section 3. At last, a face liveness detection system and its was given in section 4.

2. Eye Region Search with Modified Viola-Jones Detection Algorithm

In order to deal with the problem that the eye region is not easily detected for low-quality video frames, this paper adopted a kind of 'face-eye' search strategy to detect eye regions by using face as an auxiliary feature. First, potential face regions in video frames were detected with the viola-Jones (VJ) detection algorithm [5], and then the detected face region was taken as an interest region to search potential eye regions. Because the classical VJ detection algorithm adopts sliding window based searching strategy, this will result in a large number of sub-windows to be detected with considering multi-scales objects. This results in a large computation load, and impacts its performance of real-time. But in our system, face and eye region detection is just the first stage, so it is needed to accelerate the detection method dramatically to supply enough time for subsequent processing.

To solve the problem above, this paper proposed a scale-balanced strategy based modified VJ detection algorithm, which average computation of different scales into three consecutive video frames. The underlying hypothesis to support this strategy is that the size of an eye region could not change much in consecutive video frames under common frame rates.

2.1 Face Detection with Cascade-Structured AdaBoost Algorithm

Different from face region, human eye regions have smaller size and lesser features, so it will consume a large amount of computing time and increase the possibility of false detection to directly detect human eye regions in each video frame. Therefore, in the

 Copyright ⓒ 2016 SERSC

process of the human eye region detection, we first detected the human face region by a fast and robust cascade-structured AdaBoost algorithm introduced by Viola and Jones first [5], and then searched eye regions in the detected human face area, which would guarantee the precision and timeliness of eye region detection.

The Viola-Jones (VJ) algorithm is based on cascade-structured AdaBoost learning algorithm and Haar-like features [5, 6] that can be calculated by integral image fast. VJ's core idea is training various weak classifiers with one training set, and then integrating them together to form a strong classifier. AdaBoost algorithm itself is implemented by changing the training data distribution, and it adjusts each sample's weight according to its classification correctness as well as the total classification accuracy of previous classifier trained with old data. And then it uses the new data set with modified weights to train the next classifier. Finally the obtained classifiers are integrated into a stronger classifier.

In VJ algorithm [5, 6] a cascade-structured classifier bank is adopted to detect human faces in an image. This classifier bank is made up of many strong classifiers trained with the AdaBoost learning algorithm. The strong classifier located in the front of the cascade structure contains only a few weak classifiers. The number of weak classifiers contained in the strong classifier increases in the later stage. With this cascade-structure, the anterior classifier can quickly exclude mass non-human faces, because it contains only a small number of Haar-like features about the human face. In this way, it accelerates the speed of detection remarkably. While the posterior strong classifier contains more Haar-like features, and the quantity of non-human face samples coming into this layer becomes lesser, as a result, the harder-to-classified samples can be classified more easily in this stage.

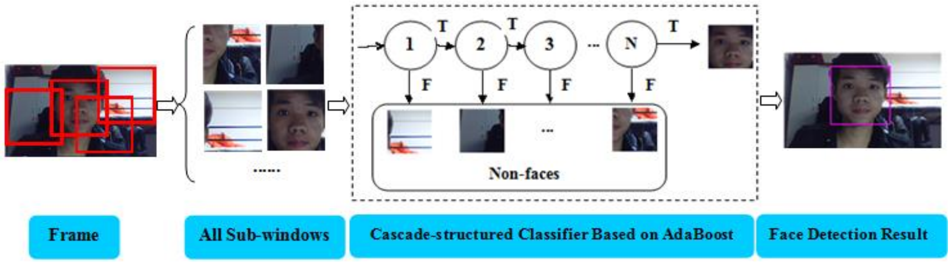


Figure 2. Diagram of Face Detection using Cascade-Structured AdaBoost Algorithm

Figure 2 is the diagram of human face detection using cascade-structured AdaBoost algorithm. First, a video frame was input and a predefined scale was chosen to get a series of sub windows based on sliding window, and then the Haar-like features of each sub-window were computed. At last a judgment of the sub-window was given by the cascade-structured classifier bank to refuse the non-face sub-window and output the face sub-window.

2.2 Using Scale Balance Strategy to Accelerate Face Detection

In order to detect human face with different sizes, VJ object detection algorithm needs to scan all sub-windows in different scales for each video frame. This greatly increased the computational complexity and largely affected the real-time processing performance of the eye detection and tracking system proposed in this paper. To solve this problem, we proposed a fast face detection algorithm based on scale balanced strategy.

The proposed scale balance strategy is to allocate the calculation of the multi-scale detection in a frame into a number of consecutive frames, and for each frame only the corresponding scale is need to be considered. Since commonly used cameras can capture more than 20 frames per second, there is little difference between the adjacent frames and the size of the detected object almost does not change in adjacent frames. This ensures that the fast detection algorithm based on scale balanced strategy can be seen as detection in the same frame and thus does not affect the results of the detection at the same time. Figure 3 shows the comparison between the traditional VJ face detection algorithm and the fast face detection algorithm based on scale balanced strategy. To compare with VJ face detection algorithm, the algorithm based on scale balanced strategy in this paper is equivalent to handle several frames at the same time, which greatly improved the detection speed.

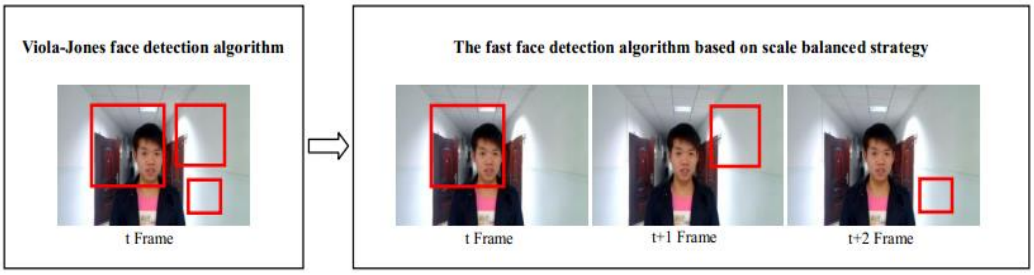


Figure 3. Comparison between VJ face detection Algorithm and the Fast Face Detection Algorithm based on Scale Balanced Strategy. The VJ Detection Algorithm made all the Scales Handled in each Frame, while the Proposed Algorithm in this Paper Equally Divided the Scales into the Adjacent Three Frames

In this paper, we test the detection performance of the VJ face detection algorithm and the proposed fast face detection algorithm based on scale balanced strategy on different data set. The experiments show that the fast face detection algorithm based on scale balanced strategy outperforms VJ face detection algorithm and can achieve real-time detection by maintaining the same accuracy. Table 1 shows the comparison of the detection performance of two algorithms in different data set. The experimental platform is: Intel(R) Core(TM) i5-3470 CPU @3.2GHZ 2.10GHZ, Microsoft Win7 operating system. The data sets for testing include the commonly used data set in computer vision and our own captured video.

Table 1. The Comparison of Detection Performance between the Original VJ Face Detection and the Improved Fast Face Detection based on Scale Balanced Strategy

Test video	Resolution	Detection Speed (fps)	
		VJ face detection algorithm	Modified algorithm based on scale balanced strategy
David [10]	320x240	18	64
Dudek [10]		19	64
TestVideo	640x320	7	22

2.3 'Face-Eye' Searching Strategy for Eye Region Detection

In many actual applications related face or eye detection, video frames captured are in low-level quality commonly. For instance fatigue driving checking, or tracking human eyes' motion trajectories on advertisements. The video frame quality could be very poor

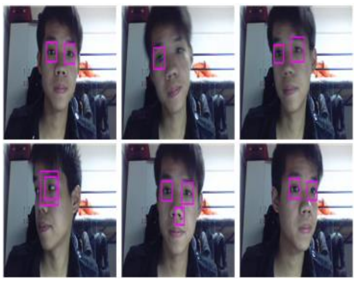
 Copyright ⓒ 2016 SERSC

due to the bad shooting environment. At the same time, human eyes are relative smaller targets in size for video frames, which are not be detected easily, let alone to be tracked.

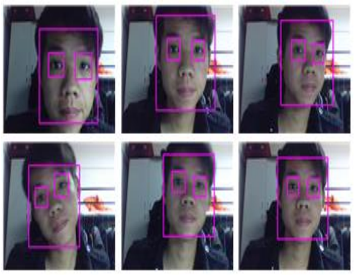
To deal with this problem, the paper proposed a 'face-eye' searching strategy which can help to detect human eye region efficiently. In the first place, the human face detection mentioned above is used to search human face region appearing in each frame. Afterwards, the obtained human face region was taken as the region of interest (ROI), then to conduct human eye detection in this ROI. The human eye detection also adopted cascade-structured AdaBoost classifier, which is same as face detection process, but the positive training samples used here are human eyes images. The approach to detect human eye is on the basis of human face region, which zooms out human eye's searching region in a video frame, and reduces the false detection rate, and increases the detection accuracy at the same time. The problem that in low-quality video sequences the detection of human eye region easily failed is solved fairly. The experiments indicate that eyes detection under the "face-eye" detection strategy is more accurate than detecting eye region directly, and can maintain little difference in the detection's elapsed time at the same time. Table 2 makes a comparison of the detection performance of the above-mentioned two methods for detecting human eyes with our testing video sequences. Figure 4 shows an example of human eye detection with the above two methods. From the Figure 4, we can see that face not only can be regarded as the auxiliary feature to detect eyes, but also is a kind of constrain information. With this constrain information, even though we only detected one eye, we can judge that this eye is the left-eye or right-eye, and can estimate the position of the other one that has not been detected at the same time.

Table 2. The Comparison of the Detection Performance Respectively Using the "Face-Eye" Detection Method and the Directly Detecting Eye Method

Method	Correct detection of two eyes	Incorrect detection of one eye	Time (s)
Eye detection without 'face-eye' strategy	79.10%	5.97%	0.25
Eye detection with 'face-eye' strategy	85.07%	1.49%	0.30



(a) Detection Results of the Method without the Auxiliary of Face



b) Detecting Results with the Auxiliary of Face Features

Figure 4. Human Eyes Detection Results in Real Life Scenario (Eye Region Tracking)

After eye regions were detected, they will be tracked to capture the user's eye motion which is useful for following liveness detection. Commonly, the movement of eye region is not relative small, and the feature points of the eye region are more obvious and relatively stable. So in this paper KLT algorithm [8, 9] was used to track the eye region

Copyright ⓒ 2016 SERSC

detected. KLT algorithm is improved on the basis of LK optical flow algorithm, and can solve the problem of aperture in LK optical flow algorithm. The KLT algorithm can achieve real-time processing speed and has a certain robustness and real-time.

3.1 KLT Tracking Algorithm

KLT tracking algorithm is first proposed by Kanade, Lucas and Tomasi. In 1981, Lucas and Kanade first proposed the LK optical flow [8, 9] computation method. The algorithm is easily applied to a set of points in the input image and later became an important method for seeking sparse optical flow. LK optical flow cannot effectively deal with the problem of aperture (*i.e*. the problem of the solution's uniqueness), and cannot accurately track large-scale movements. To solve this problem, Tomasi and Kanade improved it and proposed KLT tracking algorithm. KLT tracking algorithm has better versatility and moderate amount of computation. So it is widely used in tracking applications [11].

KLT algorithm can be divided into two steps: first, the feature points in one video frame are detected, and then the feature points detected are tracked according to the sum of square difference between the sequences of video frames with the help of three hypotheses. In order to improve the robustness of the KLT algorithm and make it achieve sub-pixel level accuracy, it usually builds a picture of the pyramid [8] to realize the feature points' tracking.

3.2 Eye Region Tracking based on KLT Algorithm

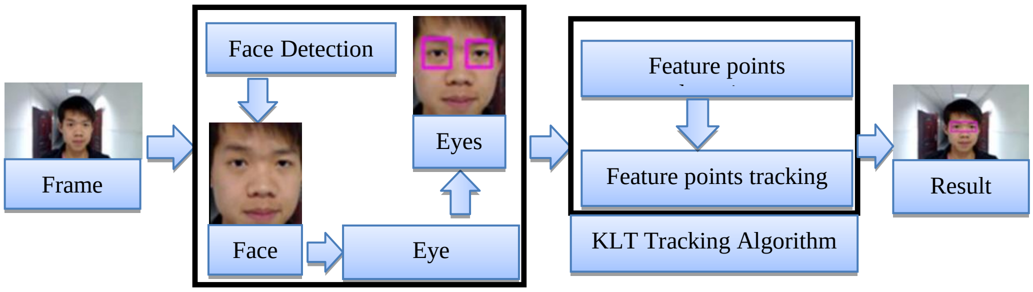


Figure 5. The Flow Chart of the Eye Region Detection and Tracking

Figure 5 shows the flow chart of eye region tracking used in our system. First, the face patch in video frames is fast detected using the AdaBoost cascade classifier and scale balance strategy proposed, and the detected face was taken as the region of interest (ROI). And then the eye region was detected in ROI rapidly and accurately. At last, the KLT algorithm was adopted to track the eye region detected.

For the eye region tracking, first the corners points in the detected eye region were used as feature points, and then KLT algorithm was adopted to track them. Because the feature points in the eye region is more evenly distributed, we can compute the position of the tracked eye region according to the following Equation (1).

$$\text{center}=\frac{\sum_{i=0}^{N} point[i]}{N} \tag{1}$$

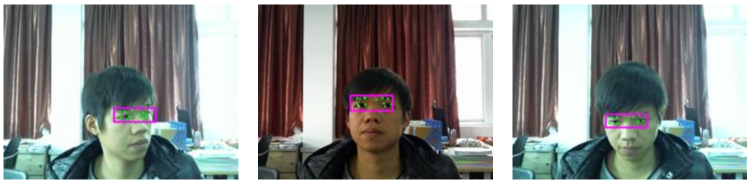
In the Equation (1), *center* represents the center coordinates of the human eye region, *point* is an array that stores the feature points' coordinates in the human eye region, *N* is the number of all feature points.

 Copyright ⓒ 2016 SERSC

(a) The Tracking Result of Eye Region on the case that face is in Front of the Camera

b) All Feature Points Tracked in Eye Region

c) The Tracking Result of Eye Region on the case that Face is turning towards Left

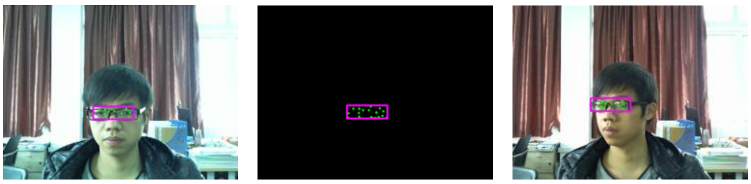


d) The Tracking Result of Eye Region on the case that the Face is turning towards Right Side

(e) The tracking Result of Eye Region on the Case that the Head is Looking Up

(f) The Tracking Result of Eye Region on the case that the Head is Looking Down

Figure 6. The Eye Region Tracking Test on the Case that Users do not Wear Glasses



(a) The Tracking Result of Eye Region on the Case that Face is in Front of the Camera

(b) All Feature Points Tracked in Eye Region

(c) The Tracking Result of Eye Region on the Case that Face is turning towards Left

(d) The Tracking Result of Eye Region on the Case that the Face is turning towards Right Side

(e) The Tracking Result of Eye Region on the Case that the Head is looking up

(f) The Tracking Result of Eye Region on the Case that the Head is Looking Down

Figure 7. The Eye Region Tracking Test on the Case that Users Wear Glasses

In order to verify the robustness and real-time performance of the proposed eye tracking method, we experimented respectively on two cases that users wear glasses and do not wear glasses in a real-world face recognition scenario. At the same time we require user to turn his or her head towards four different direction as reference [12]. The experiment platform is: Inter(R) Core(TM) Duo CPU T6500 @2.1GHZ, the operating system is Microsoft Win7. Figure 6(a) and Figure 6(b) respectively show the tracking result of eye region and its feature points. From the results we can see that the feature points are distributed evenly and tracked stably. Figure 6(c) shows the tracking result of eye region on the case that face is towards left. Figure 6(d) shows the tracking result of

Copyright ⓒ 2016 SERSC

eye region on the case that the face is towards right. Figure 6(e) and (f) show the tracking results of eye region on the case that the head is looking up and looking down. From Figure 6 we can see that KLT algorithm can track the eye region well on matter that the face is in left side, right side, looking up or looking down the camera. Figure 7 shows the tracking results in the situation that users wear glasses. All the experiments show that the proposed method can detect and track eye region well in both cases. Although the tracking result may occasionally drift during the experiment, KLT based eye region tracking algorithm is relatively robust on the whole. Because the feature points in the eye region is stable and its movement is small, KLT algorithm is very suitable for tracking in this particular case. In the experiments, the tracking algorithm can achieve an average processing speed of 20fps, and meet the needs of real-time tracking.

3. Eye Region Activity State Based Face Liveness Detection

Eye region activity state based face liveness detection is proposed in this paper by using the eye region tracking system mentioned above. The system sends random instructions (close eyes, open eyes, move left, move right), and determines whether the current face is coming from a living body according to the result that user executes the random instructions. The detection of moving left and right can be judged according to the change of the coordinate location of the eye region, and state of eyes closed can be detected with normalized cross correlation (NCC) template matching.

4.1 State of Eyes closed Detection Algorithm based on NCC

State of eyes closed detection is a very important part in eye region activity state based face liveness detection system. It is mainly used to determine whether the state of eyes is opened or closed, and determine whether the user executes ‘closing your eyes’ instruction correctly. For an eye region patch in a video frame, the gray value in it changes greatly for closing and opening eyes states. Take the image patch with open eyes as a template, and compare the current eye region image patch with this template to compute the similarity between them. If the similarity exceeds a certain threshold, the current state of the eye region is open, otherwise the state is closed.

There are many classic template matching methods, such as: Mean Absolute Deviation (MAD) [13], Normalize Cross-correlation (NCC) [14], Sum of Square Difference (SSD) [15], Sum of Absolute Difference (SAD) [16] and so on. Among them, the normalized cross-correlation based image matching algorithm has relatively high robustness. Therefore, we used the normalized cross-correlation algorithm to detect the closed eyes in this paper. The formula using normalized cross-correlation to calculate the similarity coefficient of two images is as following.

$$NCC(\mathrm{P}_1,\mathrm{P}_2)=\frac{1}{\mathrm{n}-1}\sum_{x=1}^{n}\frac{(\mathrm{P}_1(\mathrm{x})-\mu_1)(\mathrm{P}_2(\mathrm{x})-\mu_2)}{\sigma_1\sigma_2} \tag{2}$$

In equation (2) $NCC(\mathrm{P}_1,\mathrm{P}_2)$ is the similarity coefficient, $\mathrm{P}_1(\mathrm{x})$ is the template image, $\mathrm{P}_2(\mathrm{x})$ is the image to be compared, n is the number of the pixels of the image, μ_1 and σ_1 are respectively the mean and the standard deviation of the template image $\mathrm{P}_1(\mathrm{x})$, μ_2 and σ_2 are respectively the mean and the standard deviation of the image $\mathrm{P}_2(\mathrm{x})$ which is to be compared. With equation (2), we can compute the similarity coefficient between the two images. The value of the coefficient is in the range between [-1, 1]. If the two images are more similar, the similarity coefficient value will be closer to 1, and vice versa.

The experiments proved that the use of normalized cross-correlation algorithm can distinguish the state of open eyes and closed eyes. As shown in Figure 8, it shows the similarity coefficient of the current eye region image patch and the template of opened eye image patch in the process of blinking eyes. From Figure 8, we can see that the

 Copyright ⓒ 2016 SERSC

similarity coefficient of open eye image and closed eye image with the open eye template image is significantly different. According to similarity coefficient in Figure 8, the threshold value to distinguish the state of open eye and closed eye is set up to 0.9. If the similarity coefficient is greater than 0.9, the eye is open. If the similarity coefficient is less than 0.9, the eye is closed.

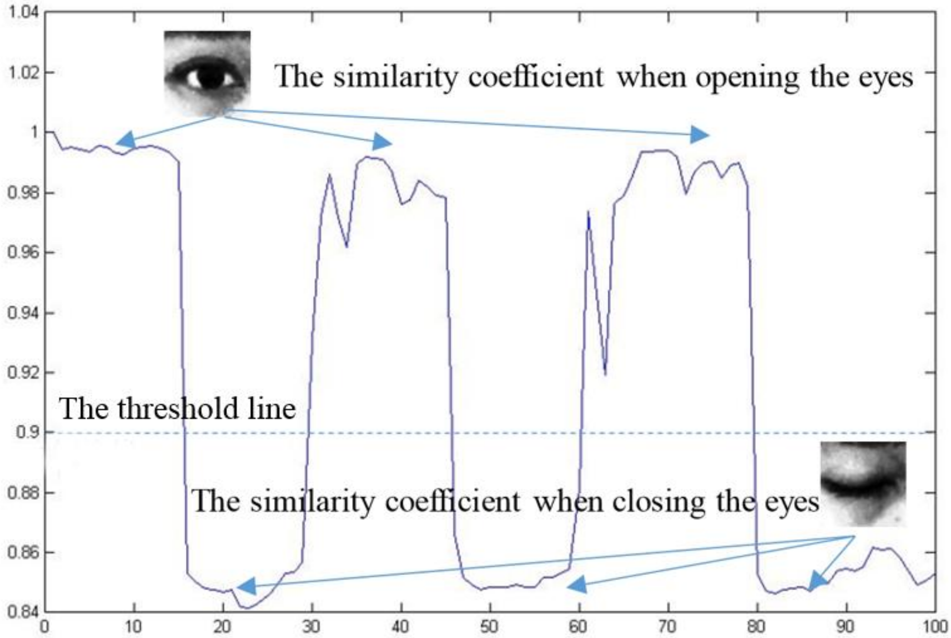


Figure 8. The Similarity Coefficient Curve during Eyes Blinking

4.2 The Implementation of Face Liveness Detection System

4.2.1 Description of the System Function: Face liveness detection can be used to distinguish current user between the true user with a counterfeiter (a video clip or photo). It has a good effect to prevent fraud from passing authentication system with a false video clip or photo, and plays a good role to protect the safety of people's life and property. The implementation of the system is based on Qt5.0.2 cross-platform application development framework and OpenCV2.4.8 (open source computer vision library). Figure 9 shows the software interface of eye region activity state based face liveness detection system.

The flow chart of the eye region activity state based face liveness detection system is shown in Figure 1. From the flow chart and the software interface of this system, we can see that the system includes the following modules:

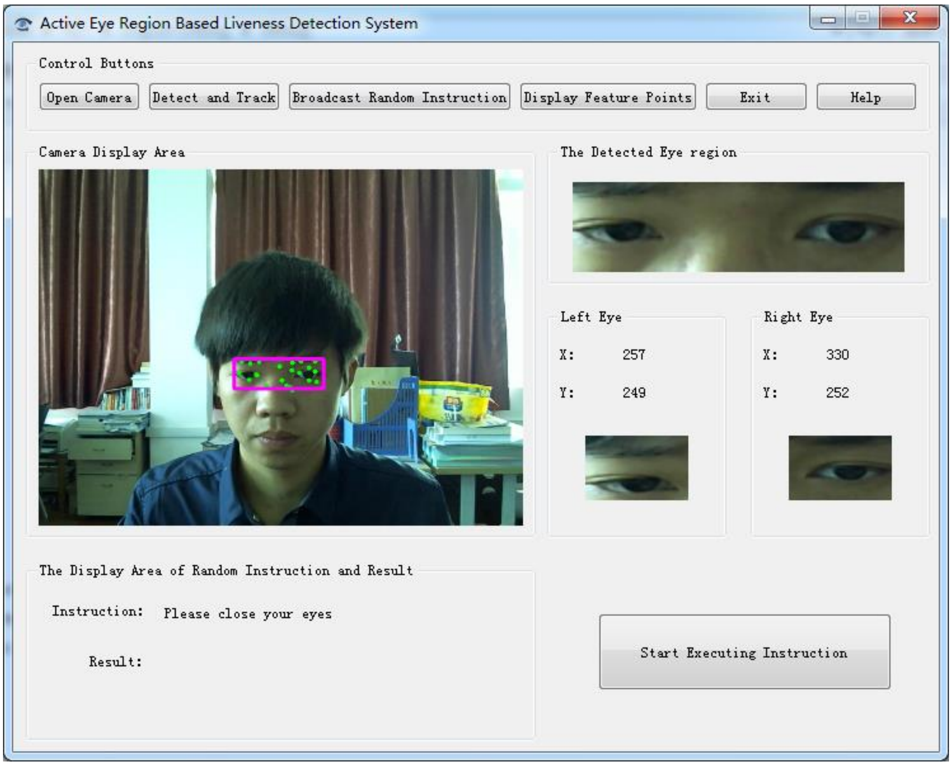


Figure 9. The Software Interface of the Eye Region Activity State based Face Liveness Detection System

1) The module for opening camera is used to open the camera, capture video frames and display it in real time.
2) The module for eye region detection and tracking. This module is one of the core modules of our system. It is mainly used to detect and track the eye region from the frame captured from the camera and make preparation for subsequent liveness detection module. The module uses the eye tracking given in section 2. As shown in Figure 10, user click the "Detect and Track" button on the software interface, and the system will begin to detect and track the eye region. Figure 9 shows the tracked eye region and the position of it.
3) The module for only showing the tracked feature points. This module is mainly used to display the tracked feature points in the eye region. Shown in Figure 9, click the "Display Feature Points" button on the software interface, the system will hide the current actual scene in the "Camera Display Area" on the software interface and show only the tracked feature points in the eye region. Figure 6(b) and Figure 7(b) show the tracked feature points in eye region.
4) The module for broadcasting random instruction. This module is mainly used to randomly generate the eye motion instructions. The system currently designed three kinds of instructions (move left, move right, close eyes). Shown in Figure 9, if a user clicks the 'Broadcast Random Instruction' button on the software interface, then the system will generate a random eye motion instruction and notifies the user in voice, and at the same time display the current random instruction on the 'The Display Area of Random Instruction and Result' region.
5) The module for liveness detection. This module is key part of our system, which is mainly used to determine whether the user correctly executes the random instruction of the eyes' motions in order to identify that the current user is in liveness or not. For determining whether the user correctly executed the random instructions, there are three situations: a) To determine whether the eye region is moved left; b) To

 Copyright ⓒ 2016 SERSC

determine whether the eye region is moved right; c) To determine whether the eye is closed. This system can judge the moving direction of the eye region according to the eye region center's coordinate changing, and determine that the eye is closed or not by using the closed eye detection algorithm based on NCC which is presented in the previous section. This system sets the similarity coefficient threshold value to 0.9. In order to avoid the impact of the eye to match in different lighting environment and different users, we select the detected eye region of current user before executing the instruction as dynamic template. As shown in Figure 9, the system can do the judgment of liveness detection by clicking the 'Start Executing the Instruction' button after the system generates the random instructions.

6) The help information module. This module is used to display the help information of the system, including the operating descriptions of it. As shown in Figure 9, click the 'Help' button to pop up the help information box.
7) The module for exiting the system. This module is used to exit the current system. As shown in Figure 9, click the 'Exit the System' button to exit the system.

4.2.2 The Performance Analysis of the Liveness Detection System In order to verify the accuracy and user experiences of proposed eye region activity state based face liveness detection system, we test the system on the following experimental platform: Inter(R) Core(TM) Duo CPU T6500 @2.10GHZ, the operating system is Microsoft Win7.

The procedure flow of this system is as follows: open the camera to show the current scene, detect and track the eye region in current scene. After tracking the region, the system randomly generates liveness instructions (moving left, moving right, and closing eyes). The user executes the appropriate action according to the instruction, and uses the liveness detection algorithm to determine whether the user executes the instruction or not. If the user executes the instruction correctly, the liveness detection is successful and the user will pass the authentication system. Otherwise the liveness detection fails, which indicates the existence of fraud and the user will not pass the authentication system.

In this paper, several tests for this system are done on multiple users. According to statistics, the current rate of the liveness detection can reach more than 98 percent, and the live detection system only fails in the case that the light changes suddenly. Because the closed eye detection algorithm in this system is based on the NCC algorithm, when the light is suddenly changed, the gray value of the image will be changed a lot, which has a great impact for the computation of the similarity coefficient, and then affects the correct rate of liveness detection. Figure 10 (a) shows the picture that the real user passes the liveness detection system by correctly executing the random instruction.

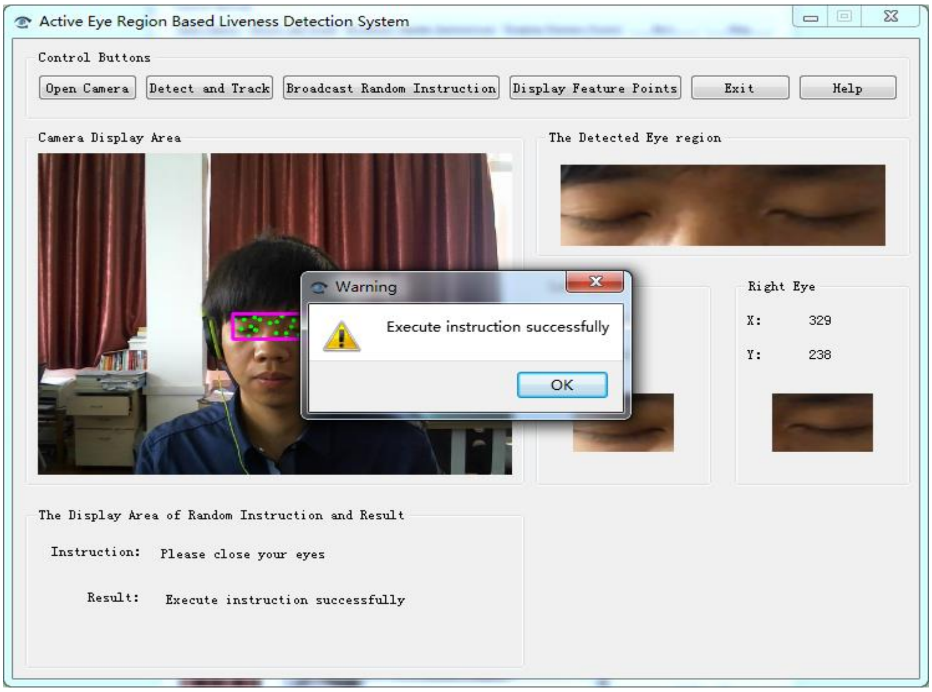
In order to verify whether the system is able to effectively prevent the counterfeit (recorded video or photos of the user), we use a printed face photo to test the system. The testing result is shown in Figure 10 (b). When the random instruction is 'close eyes' instruction, the static photo is impossible to close the eyes. So the system determined that the liveness detection failed.

All these experiments show that the system is stable and reliable. If the liveness detection system is introduced into the authentication system, it can effectively prevent the counterfeit occurred.

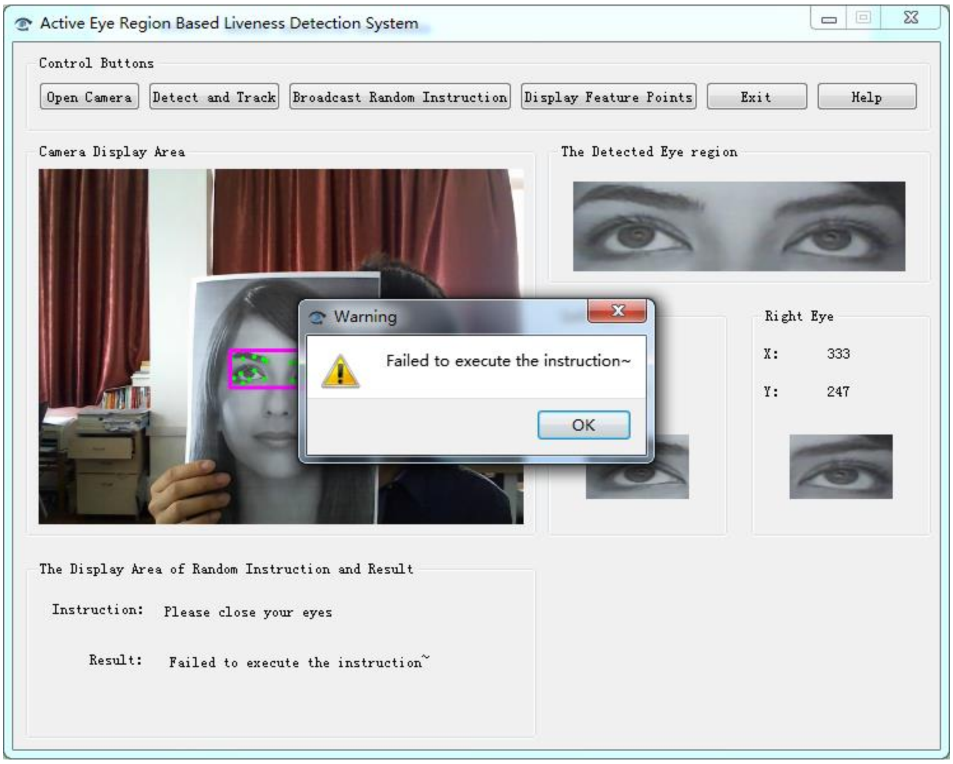
4. Summary

The paper presents an eye region activity state based face liveness detection system. First, we proposed a scale balanced strategy and a 'face-eye' strategy to detect the eye region on the basis of the VJ object detection algorithm, which solves the problem that the eye region is difficult to be accurately detected in low-quality image and improves the real-time performance of the detection algorithm.

Then the system uses the KLT algorithm to track the detected eye region. Finally it applies the tracked eye region into the liveness detection, and gives the judgment of



(a)The Experimental Result that the Real User Passed the Liveness Detection System



(b)The Experimental Result that the Face Photo Failed to Pass the Liveness Detection System

Figure 10. The Experimental Result of Liveness Detection

 Copyright ⓒ 2016 SERSC

liveness according to the results of the users' cooperation degree to execute the random instruction. For realizing the key part of the eye region activity state based face liveness detection system, we proposed to use the NCC algorithm to measure the similarity coefficient between the current eye region and the open eye template, and use it to determine the current status (close or open) of the eye region.

Real-world experimental results show that the system is real-time and robust, and can effectively distinguish current user involved from the true user with counterfeiters (videos or photos). It has a good effect to prevent fraud behavior from passing authentication system by using prepared videos or photos. For using video clips with user blinking scenario, the proposed random instruction mechanism can ensure our system security, because there are too much instruction combinations to try all cases in a limited time for counterfeiters. On the other hand, our system can be set to deadlock when three failures for implementing random instructions.

Acknowledgment

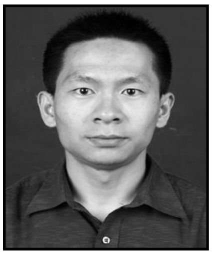
This work was financially supported by national natural science foundation of china (61402259, U1401252), Jiangsu key laboratory for 3D printing equipment and manufacturing open found project (L20140713), Hubei key laboratory of intelligent vision based monitoring for hydroelectric engineering open found project (2014KLA10, 2015KLA05), Engineering Research Center of Eco-environment in Three Gorges Reservoir Region, Ministry of Education, China Three Gorges University open found project (KF2015-10).

Reference

[1] S. G. Smith, R. J .Mills, R. T. Morton and M. E. Davis, "Voice recognition for performing authentication and completing transactions in a systems interface to legacy systems", US, Patent 8,831,949, **(2014)** September 9.
[2] A. K. Jain, H. Lin, S. Pankanti and R. Bolle, "An identity-authentication system using fingerprints", Proceedings of the IEEE, vol.85, no.9, **(1997)**, pp. 1365-1388.
[3] W. Y. Zhao, R. Chellappa, P. J. Phillips and A. Rosenfeld, "Face recognition: A literature survey", ACM Computing Surveys (CSUR), vol. 35, no.4, **(2003)**, pp. 399-458.
[4] P. Dollár, V. Rabaud, G. Cottrell and S. Belongie, "Behavior recognition via sparse spatio-temporal features", Proceedings of the 2nd IEEE International Workshop Visual Surveillance and Performance Evaluation of Tracking and Surveillance, Beijing, China, **(2005)** October 15-16.
[5] P.Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features", Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Cauai, USA, **(2001)** December 8-14.
[6] P. Viola, Jones and M. J. Jones, "Robust Real-time Object Detection", Proceedings of the Second International Workshop on Statistical and Computational Theories of Vision-Modeling, Learning, Computing, and Sampling, Vancouver, Canada, **(2001)** July 13.
[7] Panlong Yin, Guangzhu Xu, Bangjun Lei and Weihua Cao, "A Rapid Automatic Eye Detection and Long-time Tracking System", Proceedings of the International Conference on Computer Science and Artificial Intelligence, Chengdu, China, **(2013)** November 16-17.
[8] B. D. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision", Proceedings of the 7th International Joint Conference on Artificial Intelligence, Vancouver, Canada, **(1981)** August 24-28.
[9] F. Bourel, C. C. Chibelushi, A. A. Low and S. S.Dg, "Robust Facial Feature Tracking", Proceedings of the British Machine Vision Conference, Bristol, UK, **(2000)** September 11-14.
[10] David and Dudek face video sequence, http://www.cs.toronto.edu/~dross/ivt/
[11] P. Tissainayagam and D. Suter, "Assessing the performance of corner detectors for point feature tracking applications", Image and Vision Computing, vol.22, no. 8, **(2004)**, pp. 663-679.
[12] A. Fathi and M. T. Manzuri, "Eye Detection and Tracking in Video Streams", Proceedings of the IEEE International Symposium on Communications and Information Technology, Sapporo, Japan, **(2004)** October 26-29.
[13] W. F.Sharpe, "Mean Absolute Deviation Characteristic Lines for Securities and Portfolios", Management Science, **(1971)**, vol. 18, no. 2, pp. 1-13.
[14] J. P. Lewis, "Fast Normalized Cross-Correlation", Proceedings of Vision interface, Quebec, Canada, **(1995)** May 18-19.

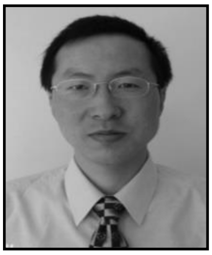
[15] L. M. Po and K. Guo, "Transform-Domain Fast Sum of the Squared Difference Computation for H.264/AVC Rate-Distortion Optimization", IEEE Transactions on Circuits and Systems for Video Technology , vol. 17, no. 6, **(2007)**, pp. 765 - 773.
[16] J. Vanne, E. Aho, T. D. Hamalainen and K. Hamalainen, "A High-Performance Sum of Absolute Difference Implementation for Motion Estimation", IEEE Transaction on Circuits System Video Technology, vol. 16, no. 7, **(2006)**, pp. 876-883.

Authors



Xu GuangZhu, (1979.1-) He is currently an associate professor at University of China Three Gorges University. He received his M.S. and Ph.D. degrees in Circuits & Systems and Radio Physics at Lanzhou University in 2004 and 2007 respectively. His main research interests are visual object detection and tracking, medical image processing and biometrics.

Yin PanLong, (1990.9-) He received his master degree from China Three Gorges University in 2015. Now he is a staff of Huangshi central hospital of Edong healthcare of Hubei province. His research interests include visual object detection and tracking, image processing and information manage system design.



Lei BangJun, (1963.6-) He received his Ph.D from TUDelft, the Netherlands in 2003. He is currently a professor at University of China Three Gorges University. His majors include low-level image processing, 3-D imaging and multi-dimensional intelligent computer vision. He is an IEEE senior member and have published more than 60 papers.

 Copyright ⓒ 2016 SERSC