International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Real Time Detection of Eye Corners and Iris Center from Images Acquired by Usual Camera

Guoqing Xu[1,2]* , Yangsheng Wang[1], Jituo Li[1], Xiaoxu Zhou[1]

[1]Institute of Automation,Chinese Academy of Sciences,Beijing 100190, P.R. China

[2]School of Computer Science and Engineering,Wuhan Institute of Technology,Wuhan 430074, P.R. China

* Corresponding author's Email: guoqing.xu@ia.ac.cn

Abstract: A precise iris center and eye corners detection algorithm is proposed. The work is based on common web cameras and natural light condition instead of high resolution devices and infrared illumination. The algorithm includes two parts. Firstly, the face region is detected with Adaboost method. Moreover, Active Appearance Model~~s~~ is used to locate the rough area of the eyes. Secondly, an improved local projection function and circle integral algorithm is presented. Based on this algorithm, the iris center and eye corners in the eye region can be precisely located. The experiments show a high accuracy. Compared with the existing methods, this algorithm provides a more reliable application in the gaze based aids tools for disabled people, such as cerebral palsy patients.

Keywords: Face detection; Statistical learning; Boosting; Eye detection

1. Introduction

Real time iris detection is one of the most important cues for gaze tracking. A majority of researches in this field concentrate on eye detection and gaze orientation tracking. Eye movement tracking has been proved to be useful in computer aided technology for people with physical disabilities such as Cerebral Palsy and upper limb amputation patients. It has become one of the main researches of the Human Computer Interactive (HCI). The precision and robustness of automatic iris detection is the significant task of the gaze tracker. However, it is difficult to recognize eye patterns because of the large variations in facial appearance and different sizes in video frames. The algorithm is sensitive to the pose and slant of the face and head, and the iris is usually partially occluded by the eyelids. Consequently, robust and precise iris locating and gazing tracking are still challenging work although many various approaches have been proposed in recent decades.

2. Related Work

In general, there are two approaches in the eye detection, namely, the active approach and the passive one. Active method realizes the detection of the facial features under the usual illumination whereas the passive approach employs infrared light (IR) sources. The IR illumination can yield the so-called bright pupil effect in the infrared pupil image, which makes it convenient for gaze tracking.

In an early successful application, Erica [1] used eye gaze as input device and presented a uniform processing framework. It consists of a personal computer, a near infrared light source, a light tracking video surveillance camera with infrared pass filter and a light intensity imaging board. The light source is composed of lighting emitting diodes (LEDs), which emit near the infrared light. When illuminated with the infrared light, the eye image gains two significant reflected features called the glint and the bright eye. Consequently, the direction of the subject's eye-gaze can be determined by the glint and bright eye in the camera image. Shen-Wen Shin [2] presented a novel approach in which multiple cameras and multiple point light sources are employed to estimate the optical axis of the subject's eyes. With the 3D computer vision technique, operators do not have to keep the subject's head still or spend a lot of time calibrating the gaze tracking system. There are also many researches on the active infrared illumination eye gaze detection methods, which can be found in [3] [4].

The passive gaze tracking method makes use of the image acquired by the usual camera under natural illumination condition instead of the IR lighting and special acquisition devices. Compared with the active method, it provides convenience for assembly and use. There are a lot of computer vision researches dedicated to the implementation of the facial gestures and features detection. Grauman [5] and Chau [6] provided a real-time vision system that automatically detected the subject's eye blinks and accurately measured the durations, which helped to realize communication by using blink pattern. Morris et al. [7] presented a non-contact computer operation method controlled by head and eye for people with motion difficulties. Spatio-temporal filtering and variance maps are used to locate the head and find the eye feature points respectively.

In these methods, a modified version of the Locas-Kanade tracking algorithm was applied. However, the eyes location was determined by the difference between current frame and the previous frame. If the eye had been still, these methods would have been failed. As a generative model of a certain visual face appearance, Active Appearance Model (AAM) [8,14] has been successfully used in the face features location. AAM was powerful in modeling and registering deformable objects. Song et al. [9] proposed a method for eye detection including the extraction of the binary edges from the gray-scale face images. Multi-resolution wavelet transform AAM was proposed in his work. An improved face region extraction and a light dots detection algorithm were used for better eye detection performance. Feng et al. [10] presented a three cues eye detection method for gray intensity image. The cues consisted of the face intensity, direction of the line joining the centers of the eyes, and the response of convolving the proposed eye variance filter with face image. Moriyama [11] proposed a system for the detailed eye region analysis based on a generative eye region model. It parameterized the fine structure and motion of an eye. The structure parameters were adjustable for the movement of the eye and represented structural individuality of the eye. It achieved accurate registration and motion recovery of eyes.

In recent decade, AAM has deserved further research and extensive application in the facial feature location, especially in the eye region location. It was robust to various environment illumination and background motion. Eyes location in the previous algorithms is usually based on the facial structure model. The position of the eyes is determined after the face is detected. This method is feasible in most cases. However, when the face tilts to left or right, or when there is complicated background, hair and eye brown shade condition, error detection and deviation occur. Therefore, this paper uses the appearance model to locate the eye in the detected face region. This method can overcome the disadvantages brought about by face tilt.

This paper is organized as follows. In Section 3, an Adaboost face location method is proposed. Considering the diversity of the training sample, the algorithm can adapt various facial poses, head slopes, and partial occluded face. Section 4 focuses on the iris center locating. A local projection function and the fast integral image are presented. In Section 5 the experiment is elaborated and a conclusion is drawn.

3. Face detection

As a general method, Adaboost [12,13] has been widely used for improving the accuracy of any given learning algorithm. Boosting realizes a classification algorithm to reweight versions of the training data and then takes a weighted majority vote of the sequence of classifiers thus produced. Therefore, dramatic improvement in performance can be achieved through simple strategy among many weak classification algorithms. It is considered to be one of the most significant developments in machine learning. Finding many weak rules of thumb is easier than finding a single, highly prediction one. Therefore, a strong classifier can be combined with many weighted weak classifiers. Kearns and Valiant proved the astonishing fact that although each weak learner performs only slightly better than random, these weak learners can be combined to form an arbitrarily good ensemble hypothesis. Viola and Jones have successfully applied Adaboost to facial recognition.

A theoretically justified Adaboost learning algorithm is termed Jesen-Shannon (JS) Boost, which used relative entropy loss as objective function for learning weak classifiers. For each iteration of Adaboost learning, a weak classifier that maximizes JS divergence is the one that minimizes relative entropy. JS divergence provides more appropriate measures of dissimilarity between two classes because it is numerically more stable than other measures such as Kullback-Leibler (KL) divergence. The JS-Boost learning algorithm is demonstrated with an application to face recognition. Results show that classifiers learned by using JS-Boost produce better results than other Adaboost variants such as RealBoost, GentleBoost and KL-Boost.

Liu and Shum [15] adopted the maximizing information gain criterion to maximize the KL divergence of the positive and negative histograms projected on the feature. The symmetric version KL divergence is computed as follows,

$$SKL(\phi_i) = \int \{[h_i^+(\phi_i(x)) - h_i^-(\phi_i(x))] \cdot \log \frac{h_i^+(\phi_i(x))}{h_i^-(\phi_i(x))}\} d\phi_i(x)\,, \tag{1}$$

where $h_i^+(\phi_i(x))$ and $h_i^-(\phi_i(x))$ are the histograms of the positive and negative examples with weights $W_i(x_i^+)$ and $W_i(x_i^-)$ respectively. However, it should be noted that KL divergence is undefined if $h_i^+(\phi_i(x)) = 0$, or $h_i^-(\phi_i(x)) = 0$. JS divergence [17], defined below, can overcome this problem.

$$SJS(\phi_i(x) = \int \{h_i^+(\phi_i(x)) \log \frac{h_i^+(\phi_i(x))}{\frac{1}{2}[h_i^+(\phi_i(x)) + h_i^-(\phi_i(x))]} + h_i^-(\phi_i(x)) \log \frac{h_i^-(\phi_i(x))}{\frac{1}{2}[h_i^+(\phi_i(x)) + h_i^-(\phi_i(x))]}\} d\phi_i(x) \tag{2}$$

From the Shannon inequality, it shows that $SJS(\phi_i) \geq 0$. $SJS(\phi_i) = 0$ if and only if $h_i^+(\phi_i(x)) = h_i^-(\phi_i(x))$, which is essential for a measure of difference.

Haar-like and extended features proposed by Viola and Jones et al. [12] can easily describe the horizontal and vertical average gray gradient information quickly. However, it is hard for these features to describe the shape and edge information of face suitably, which is the main feature pattern of the face information. Nevertheless, Sobel operator is a simple and useful tool to represent edge information:

$$G_x(x,y) = S_x * I(x,y) \tag{6}$$

$$G_y(x,y) = S_y * I(x,y)\,. \tag{7}$$

Here the S_x and S_y are the filter operators of the x and y directions. The gradient strength and orientation of the pixel (x,y) are defined as

$$G(x,y) = \sqrt{G_x^2(x,y) + G_y^2(x,y)} \tag{8}$$

and

$$\phi(x,y) = \arctan(\frac{G_y(x,y)}{G_x(x,y)})\,. \tag{9}$$

The Sobel convolution is sensitive to the local noise around a single pixel. Therefore, this paper used an efficient Sobel-like feature to calculate the face features. It can be calculated through the following steps,

1. Calculate the integral image using the fast algorithm proposed by Paul Viola [13];
2. Scan the full image with a sub-window of 3 times 3 blocks, the size of each block is *m* times *n*;
3. Convolute image blocks by Sobel kernel;
4. Calculate the strength and the orientation of the gradient and then obtain the generalized features.

Figure 1 gives the Sobel-like feature filter operators. The feature space can be constituted through enumerating all of the blocks at various sizes.

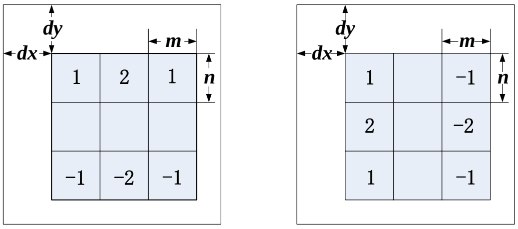


Figure. 1 Sobel-like features filter

During Sobel-like feature construction, blocks instead of the pixels of the images are convolved with Sobel operator. And a final strong classifier is trained through the features and the objective function for learning weak classifier mentioned above. Since the calculation of Sobel-like features is more time-consuming than the haar-like features, in the real time face detection, the haar-like classifiers are firstly used to discard the obvious non-face regions, and the Sobel-like classifier is used for further detailed classification. Figure 2 gives the detect results.

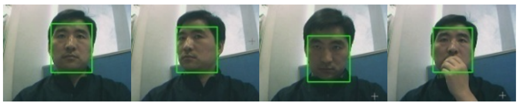


Figure. 2 Face location in different conditions

4. Location of the iris center and eye corners

4.1 Eyes location

AAM contains a statistical model of the shape and grey level appearance of a potential target. It can generalize almost any valid facial example. Matthews and Baker [14] proposed the inverse compositional image alignment algorithm. It outperformed previous approaches.

The shape of an object can be represented as a vector x. The shape model has parameters p_i controlling the shape according to the following expression

$$x = x_0 + \sum_{i=1}^{n} p_i x_i \, , \qquad (10)$$

where x_0 is the mean shape, x_i is the principal axis of shape, and p_i is a scalar. Similarly, the appearance model is represented by

$$A = A_0 + \sum_{i=1}^{m} \lambda_i A_i \, , \qquad (11)$$

where A_0 is the mean appearance, A_i is the principal axis of appearance space, and λ_i is a scalar.

Fitting an AAM to an image is considered to be a problem of minimizing the error between the input image and the closest model instance. The cost function is

$$g = \sum_x [I(W(x;p)) - T_0(W(x;\Delta p))]^2 , \quad (12)$$

where T_0 is the template, I is the input image, and W is the warp operation.

This algorithm can converge to any given accuracy from a large distance away in less iteration. Because the appearance variation is projected out, the algorithm is faster. It is significant for the real-time applications. If the input image is far from the model space, the fitting process will fail. This can always occur because of illumination variation. Therefore, building a good appearance space is very important. We propose a weighted cost function which can incorporate the intensity and edges of an image into AAMs framework.

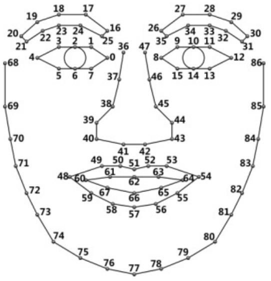
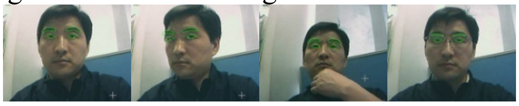


Figure. 3 Facial feature model in AAM

In this paper, each training image is marked with 87 key points manually as shown in figure 3. The keys consist of the eye regions and other facial features. In the experiment, the facial alignment time was about 20ms. This algorithm is robust to the eyes with glasses as shown in figure 4.



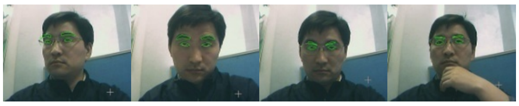


Figure. 4 Eye location in different conditions

4.2 Window Projection Function

As an effective eye location method, projection function can quickly locate the eye landmarks and improve the successive detection process. Since the eyeball is a low identity circle, the integral projection function will present a distinct valley in the integral curve. It can indicate the location of the eyeball. However, in the usual illumination, there will appear reflection highlight points on the cornea. And the integral projection will have some jumping points around the center of the eyeball. The simple use of the projection function will thus have bias. Considering that the reflection under the usual condition is not outstanding, a windowed projection function is calculated as

$$IPF_W(x) = \frac{1}{C_{GW}} \int_{\lambda \in \delta} G_\delta(\| x - \lambda \|) IPF_V(\lambda) d\lambda \, , \, (6)$$

where IPF_V is the projection function, and $G_\delta(\| x - \lambda \|)$ a filter kernel. C_{GW} is the normalization parameter. Although Gaussian kernel is a perfect filter for the noise smoothing, it also decreases the weight of the probable eyeball center projection. In other words,

$$IPF_V(x) = \frac{1}{K_W} \int_{y \in W} I(x,y) dy \qquad (7)$$

is very sensitive to the reflection on the eyeball, where IPF is gray summation of the global operations. Especially when there is hair and makeup shade, projection waveform can produce many valleys, which increases the probability of error detection. On the other hand, the traditional projection method is only a one-dimension search algorithm. After all, the vertical and the horizontal projection are two independent processes.

Projection function is always used on the whole face region. In this paper, we limit the projection function within a pre-calculated local eye region. This work is based on the face detection and an AAM algorithm, where the key points around the eyes can be detected. And if the face tilts at a small angle, these points can be robustly tracked. It is an intuitionistic choice to limit the projection region in the constant area and shape, i.e. the rectangle should be constant when the face rotates. This paper proposed a square projection to realize such algorithm which is shown in figure 5. The line connecting the point A_2 and B_2 indicates the line crossing the corner of the eye. Since the eye width is invariable when the face rotates, the projection region is set to be the same as the circle diameter denoted by the line A_2 through B_2. For convenient calculation, the exo rectangular of the circle is used as the projection region.

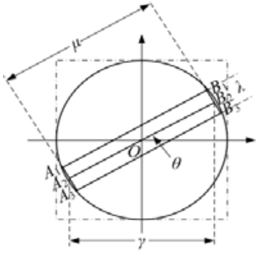


Figure. 5 Eye model

The projection image is calculated in the exo rectangular of the two eyes. Figure 6 shows the curves. From top row of the figure it can be seen that the projection curve on the horizontal axis has two valleys around the eyeballs. However, the vertical axis has more valleys especially when the two eyes are not in a horizontal line, which is very usual because the face is infrequently upright.

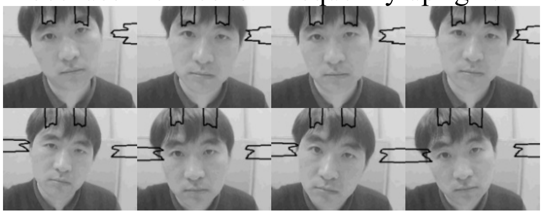


Figure. 6 Sole (top row) and respective (bottom row) vertical projection of the eyes

In order to overcome such defects, the vertical projection of the two eyes is calculated respectively. Based on the primary location, the unrelated points outside the two regions are set to zero in gray identity. The second row of figure 6 shows the horizontal and vertical projection of the two eyes, with the two curves having explicit valleys even though the face tilts in a larger scope.

Gaussian kernel is usually used as a smoothing filter. However, the whole input image has been smoothed at the image preprocessing. The noise of the local projection function curve has also been decreased. De-noise is not the main purpose of the window operation. Therefore, a rectangle window is performed to slide on the local projection curve. Such calculation is equal to a cumulative summation of the two-dimensional image space.

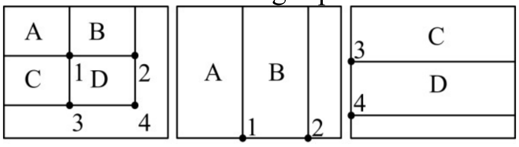


Figure. 7 Fast integral image

Paul Viola[12] used integral image to sum the pixel identity within a rectangle as shown in the left of figure 7. The value of the integral image at location 1 is calculated by the rectangle A. Location 2 is calculated by A + B, location 3 is A + C, and location 4 A + B + C + D. Therefore, the integral of region D is represented by linear combination of the four rectangles with the same start point O,

$$I_D = I_4 + I_1 - (I_2 + I_3)\,. \tag{8}$$

The projection window can also be calculated through integral image. The middle part of figure 7 shows the horizontal projection window of the whole image, and on the right is the vertical projection window of the whole image. In the usual image processing, a local window is sliding to perform the integral. Therefore, there are four image rectangles to be calculated. The main difference is that the integral is calculated in an elongated area, as shown in the middle and right part of figure 6, which can be simplified as,

$$I_B = I_2 - I_1 \tag{9}$$

$$I_D = I_4 - I_3\,. \tag{10}$$

4.3 Location of the Iris Center

In allusion to the fact that the iris is a circle, Ahlberg and Du [16, 19] used the polar coordinate integral to detect the center of the iris. The integral interval in this method was set to $[0,2\pi]$. From the eye model shown in figure 8, it can be seen that the top and bottom area of the iris are usually sheltered by the lid, and the iris will not be an intact circle. Specially, the shelter area will enlarge when user frowns, blinks, or gazes at the inner and outer eye corner. In order to reduce such effect, the integral interval should be limited.

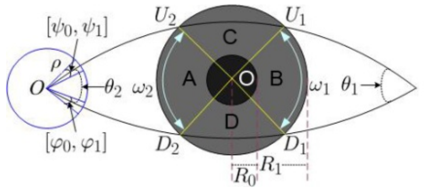


Figure. 8 Eye model

When the iris is partly covered by the lid, the iris areas composed with $D_1 - U_1$ $(\theta \in \omega_1)$ and $D_2 - U_2$ $(\theta \in \omega_2)$ are still arcs. Consequently, the integral interval can be limited to $\theta \in \Omega$, where $\Omega \in \omega_1 \cup \omega_2$.

Let $I_\Omega(x_0, y_0, r)$ be the integral of the circle around the pixel (x_0, y_0) where the radius equals r and the angle belongs to set Ω.

$$I_\Omega(x_0, y_0, r) = \int_{\theta\in\Omega} I(x_0 + r\cos\theta, y_0 + r\sin\theta)d\theta$$

(11)

The integral has maximal gradient around the contour of the iris (x_0, y_0, r). Therefore, the location of the center and radius of the iris can be denoted by the partial differential maximum value $I_\Omega(x_0, y_0, r)$,

$$(x_0, y_0, r) = \arg\max(G(r) * \frac{\partial}{\partial r} I_\Omega(x_0, y_0, r)), \quad (12)$$

where $G(r)$ is a Gaussian convolution kernel.

4.4 Location of the eye corners

The corner of the eye can be located on the binary image. However, there are noises in the local area around the eye corners. The direct projection has bias. The corner of the eye is the contour cross point of the top and bottom lid. Therefore, the contour of the lids around the eye corner can be included in a sector area by setting up a polar coordinate pointing to the eye corner:

$$l_u : r \in [0, \rho], \alpha \in [\psi_0, \psi_1] \quad (13)$$

$$l_d : r \in [0, \rho], \alpha \in [\varphi_0, \varphi_1]. \quad (14)$$

In order to reduce the searching area and increase the accuracy, initial candidate range of integral can be set to

$$g(x, y), x \in [x_l, x_r], y \in [y_u, y_d], \quad (15)$$

where y_u is the top start point coordinate in the detection area. Under the usual vision, the eye corners are lower than the iris center, so coordinate y_u can be set to be the same as the vertical point of the iris center. Gray projection is processed in this rectangle area. If the vertical projection value satisfies $I_v(x, y) > \sigma_v$, the point (x, y) can be a candidate of the iris center, where σ_v is the threshold. In order to exclude the noise, the integral area is limited in the sector

$$I_u(x, y) = \iint_{(x,y)\in\omega_u} I(x, y)dxdy$$
$$= \iint_{r\in[0,\rho],\theta\in[\psi_0,\psi_1]} I(r, \theta)rdrd\theta$$

(16)

$$I_d(x, y) = \iint_{(x,y)\in\omega_d} I(x, y)dxdy$$
$$= \iint_{r\in[0,\rho],\theta\in[\varphi_0,\varphi_1]} I(r, \theta)rdrd\theta.$$

(17)

5. Experiment

The experiment is processed under natural lighting condition. The computer configuration is Pentium4 2.4GHz CPU and 512Mbit memory. Since the target of the iris detection is to locate the gaze orientation of the person in a real time common camera video, the test is performed in a low resolution image database, e.g. the BioID and FERET face database. These two databases consist of frontal face images respectively. The size of each image in BioID is 384×288 pixels, and the FERET is 256×384 pixels. The eyes positions of the facial images have been manually marked. It is convenient for the calculation of the eye center detection algorithm accuracy. As the base of the whole work, face detection is firstly executed. The face position is calculated by using the boosted cascade face detection algorithm given by Viola and Jones [12]. The classifier can correctly distinguish most faces. However, there are also some failures. The main reason includes the reflected illumination, attitude, occlusion and bushy beard. The whole detection ratio is about 70%.

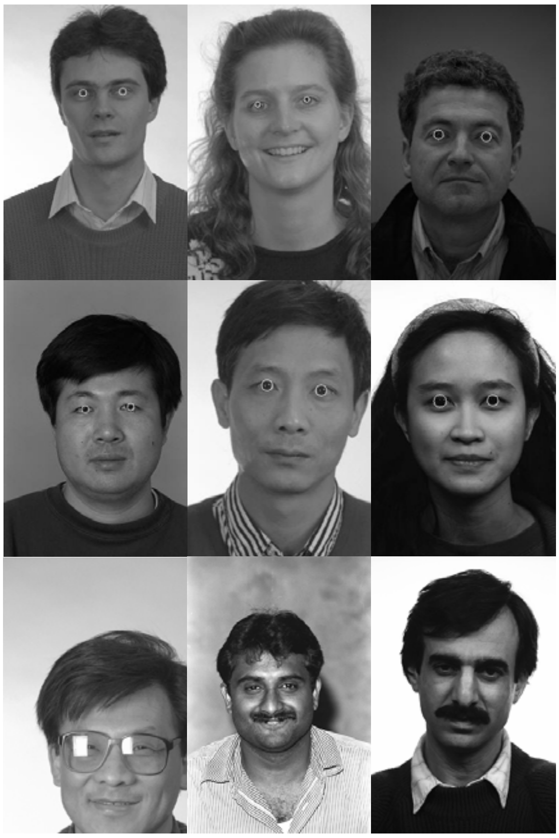


Figure. 9 Successful (top 2 rows) and unsuccessful (bottom row) iris center detection on the FERET face Database

Figure 9 shows the successful iris center location samples. A white circle in the eye indicates the location of the iris. The circle center is the recognition eye center. For the distinct show of the black iris area, the eye center is not drawn. As a usual evaluation, the normalized error of the eye center XY coordination is calculated. This measure was firstly presented by Jesorsky [18] as

$$\text{err} = \frac{\max(d_l, d_r)}{d_{lr}}, \qquad (18)$$

where d_l and d_r are the Euclidean distances between the detected and the ground truth left and right eye. d_{lr} is the Euclidean distance between the eyes in the ground truth.

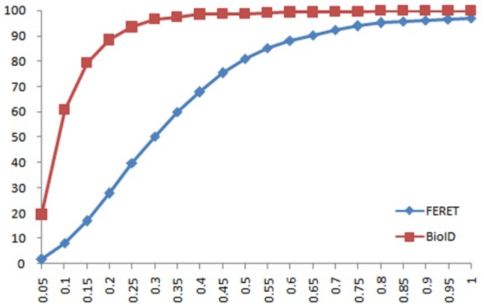


Figure. 10 Iris detection result on the two face databases

Figure 10 gives the iris center location results on the two face databases. The vertical axis is the location ratio under the corresponding err ratio of the normalized error of the eye center calculated by equation 18. It performs better on the FERET than BioID since each image in latter is not always a standard front upright face, and the illumination is another main reason. When err<0.25, FERET performs a detection ratio larger than 95%.

The eye center tracking performance is evaluated by a usual web camera in the natural illumination. The resolution of the camera image is 320 × 240 pixels. The subject is sitting in front of the camera, at the distance of about 50cm. Figure 11 shows that the algorithm can robustly locate and track the eye center especially when the user moves or rotates the head. The processing speed is 25 fps.

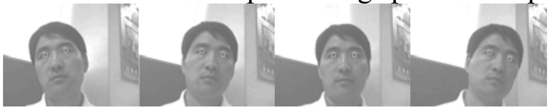


Figure. 11 Real time iris track with a web camera

Thanks to the existing statistics, three videos captured by the usual camera are tested. These videos have 297 × 392 and 428 frames respectively. Each video includes eyeball rotation, blink and slight head shaking. The features to be detected are eye corners, iris center and iris radius. In each video, the correctly detected feature numbers are counted and drawn in top of figure 12. It shows that the high correction appears in eye corners and iris center. However, the iris radius detection rate is low. Frequent winks and shelter are the main reason. The bottom part of figure 12 shows an approximate linear relationship between the frame numbers and the correctly detected feature numbers. It also indicates that the correction rate of eye corners and iris center detection is high, and iris radius rate is low.

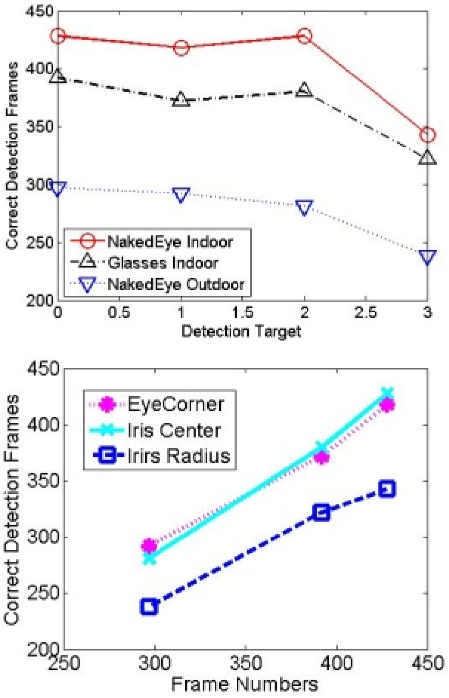


Figure. 12 Accuracy and efficiency of detection

6. Conclusion

In this paper, a robust iris center location is presented. It can stably detect the eye center in various conditions such as background movement and environment illumination changing. A local projection integral is presented to increase the location accuracy. Compared with the direct projection, the local window method can effectively adjust the tilt of the face and the local noise point around the iris center. With the improved integral image and the circle integral image algorithm, the eye center and the iris radius can be precisely located and tracked. The future work will focus on the processing of illumination variation, which can cause some highlight reflection on the cornea and increase the difficulty of robust detection of the eye center.

Acknowledgments

This work has been funded by National Key Technology R&D Program in the 11th Five Year Plan of China under the program of Research on Disabled People Function Rehabilitation Aids (Grant No. 2009BAI71B07). This program aims to foster innovation to make technology accessible for the disabled.

References

[1] T.E. HUTCHINSON and K.P. WHITE. Human computer interaction using eye-gaze input. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1527-1534, 1989.

[2] S.W. Shin and J. Liu. A novel approach to 3d gaze tracking using stereo cameras. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, 34(1):234-245, 2004.

[3] Kang Ryoung Park. A real-time gaze position estimation method based on a 3-d eye model. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, 37(1):119-212, 2007.

[4] A. Villanueva and R. Cabeza. A novel gaze estimation system with one calibration point. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, 38(4):1123–1138, 2008.

[5] K. Grauman and M. Betke. Communication via eye blinks - detection and duration analysis in real time. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1010-1017, 2000.

[6] Michael Chau and Margrit Betke. Real time eye tracking and blink detection with usb cameras. *Boston University Computer Science Technical Report*, (12), 2005.

[7] T. Morris and P. Blenkhom. Blink detection for real time eye tracking. *Journal of Network and Computer Applications*, 25(2):129–143, 2002.

[8] T.F. Cootes and C.J. Edwards. Active appearance models. In: *Proc. of European Conf. On Computer Vision*, 2:484-498, 1998.

[9] J. Song and Z. Chi. A robust eye detection method using combined binary edge and intensity information. *Pattern Recognition*, (39):1110–1125, 2006.

[10]G.C. Feng and P.C. Yuen. Multi-cues eye detection on gray intensity image. *Pattern Recognition*, (34):1033-1046, 2001.

[11]T. Moriyama and T. Kanade. Meticulously detailed eye region model and its application to analysis of facial images. *IEEE Transaction on pattern analysis and machine intelligence*, 28(5):738-752, 2006.

[12]P. Viola and M.J. Jones. Rapid object detection using a boosted cascade of simple features. *Computer Vision and Pattern Recognition*, 1:8-14, 2001.

[13]P. Viola and M.J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137-154, 2004.

[14]Iain Matthews and Simon Baker. Active appearance models revisited. *Journal of Computer Vision*, 60(2):135–164, 2004.

[15]C. Liu and H. Y. Shum. Kullback-Leibler Boosting. In Proc. *IEEE Conf. on Compuer Vision and Pattern Recognition*, pp. 407-411, May 2003.

[16]J. Ahlberg. A system for face localization and facial feature extraction. In: *Proc. of Siggraph* 98, pp. 124–139, 1999.

[17]Y. Bubner, J. Puzicha, C. Tomasi, and J. M. Buhmann. Empirical evaluation of dissimilarity measures for color and texture. *Computer Vision and Image Understanding*, 84:25 – 43, 2001.

[18]O. Jesorsky, K.J. Kirchbergand, and R. Frischholz. *Robust face detection using the Hausdorff distance*. Audio and Video Biom. Pers. Auth., 90-95,1992.

[19]Du Zhijun, Wang Yangsheng. Eye location Algorithm for Frontal View Face Image. *Journal of Computer-Aided De-sign and Computer Graphics*. 21(6):763-769,2009.