Eye Location Based on Structure Feature for Driver Fatigue Monitoring

Qiong Wang

Abstract—One of the most important problems to solve is eye location for a driver fatigue monitoring system. This paper presents an efficient method to achieve fast and accurate eye location in grey level images obtained in the real-word driving conditions. The structure of eye region is used as a robust cue to find possible eye pairs. Candidates of eye pair at different scales are selected by finding regions which roughly match with the binary eye pair template. To obtain real one, all the eye pair candidates are then verified by using support vector machines. Finally, eyes are precisely located by using binary vertical projection and eye classifier in eye pair images. The proposed method is robust to deal with illumination changes, moderate rotations, glasses wearing and different eye states. Experimental results demonstrate its effectiveness.

Keywords—eye location, structure feature, driver fatigue monitoring

I. INTRODUCTION

THE detection of driver visual attention is very important for developing automatic systems that monitor the driver inattention and driver fatigue: a great number of fatalities occurring in motor vehicles could be avoided if these behaviors were detected and alarm signals were provided to the driver. Eye state detection is a crucial aspect in these applications: the monitoring of the attention level requires a robust eye state detection algorithm that can provide information for gaze detection, determines if the eyes are open or closed. Before eye are recognized and tracking, eyes must be located first. Therefore, eye location is an indispensable step for eye state detection in the driver fatigue monitoring system. At the first frame, eyes must be located before tracking, and to make sure the correct feature is being tracked, and eyes should be relocated periodically during the tracking. A large number of works have been published in the last decade on this subject. Generally the location of eyes can be done using active or passive techniques. Active techniques use the spectral properties of pupil under near IR illumination to highlight the eyes in the face images [1], [2], [3]. They are very simple and effective, but reveal some shortcoming: the success of these systems requires stable lighting conditions and the subject close to the camera. Passive techniques detect eyes based on their different appearance from the rest of the face. Generally they consist of two steps: locating face to extract eye regions and then eye detection from eye windows. The face detection problem has been faced up with different approaches:

Qiong Wang is with the Nanjing University of Science and Technology, Nanjing, China (e-mail: wangqiong603@163.com).

Adaboost, neural network, principal components, and skin color based methods [4], [5], [6]. On the other side many works for eye or iris detection assume either that eye windows have been extracted or rough face regions have been already located [7], [8], [9]. By our knowledge, no much works have been presented in literature that search directly eyes in whole images and are cheap both in cost and computational complexity.

The main objective of our work is to propose an eye location algorithm that is applicable in real time with standard cameras, in a real context such as people driving a car (with a complex background), and skipping the first segmentation step to extract the face region as commonly done in literature. Our eye location algorithm works on the whole image, looking for regions which have the eye pair structure. Different iris radiuses are allowed in order to face with people having different eyes dimensions and also light variations in the distance between the camera and the person. A SVM classifier is trained to obtain the real eye pair. After eye pairs are located, eyes are precisely located in by using binary vertical projection and eye classifier. Experimental results demonstrate that no matter the eyes are open or closed; the algorithm provides correct location results. Tests carried out on persons with different eyes dimensions and states, some of them wearing glasses, demonstrating the effectiveness of the proposed algorithm.

The rest of the paper is organized as follows: Section 2 describes the eye pair location algorithm. Section 3 describes the precise eye location algorithm based on correct eye pair location. The experimental results are reported in Section 4. Finally, in Section 5 conclusions and future works are presented.

II. EYE PAIR LOCATION

Eyes are the most important facial features in face detection and recognition systems. And the structure of eye region is a stable and robust feature which can distinguish eye pair from other patterns. The proposed location method finds possible eye pairs by binary template matching. Firstly, the facial image is enhanced and binaried to obtain the structure image. We define the structure image as the binary image which contains the structure feature of human face. Then a binary eye pair template is used to find eye pair candidates in the image. All the eye pair candidates are then rescaled to a fixed size and sent to the SVM classifier. An output ranging from 1 to -1 are generated, signifying the presence or absence of eye pair, respectively. Finally, the eyes can be located.

The facial images which will be processed in template matching step are binary images which contain the facial

World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:4, No:12, 2010

structure information. To obtain good-segmented binary images, four preprocessing algorithms are applied to the input images. In order to compensate the illumination and obtain more image details, the homomorphic filter is used to enhance the brightness and the contrast of the images. Then we use clustering algorithm to divide the facial feature from the skin and background, and the binaried image can be obtained through thresholding.

A. Homomorphic filtering

Homomorphic filtering is a generalized technique for nonlinear image enhancement and correction. It simultaneously normalizes the brightness across an image and increases contrast.

An image can be expressed as the product of illumination and reflectance:

$$f(x,y) = i(x,y) \cdot r(x,y) \qquad (1)$$

When the illumination is uniform, $i(x, y)$ is considered to be a constant and the image is considered to be the reflectance of the object. However, the lighting condition is usually uneven. The illumination component tends to vary slowly and its frequency fastens on low part in the frequency domain; the reflectance tends to vary rapidly and its frequency is in high part. If two components can be operated separately, the illumination problem will be solved and the image will be enhanced. Hence, the log transform is used to equation (1):

$$ln\, f(x,y) = ln\, i(x,y) + ln\, r(x,y) \qquad (2)$$

Then Fourier transform is used to equation (2), which makes the succedent operation is in frequency domain. The illumination and relectance turn to additive though log transform. Then 2-D Fourier transform is used and the coordinate variables are u and v. $H(u,v)$ is the homomorphic filter function applied to the illumination and reflectance, respectively. After taking the inverse Fourier transform and exponent transform, we get enhanced image $g(x,y)$. $H(u,v)$ used in this paper has the following form:

$$H(u,v) = (H_H - H_L) \cdot (1 - exp(-C \cdot \frac{D}{D_0})) + H_L \qquad (3)$$

If the parameters H_L and H_H are chosen to be $H_L < 1$ and $H_H > 1$, then the filter $H(u,v)$ will decrease the contribution of the low frequency (illumination) and amplify the contribution of mid and high frequencies (reflectance). As shown in Fig.1, Fig.1(a) is the input image with low contrast due to the illumination, and we can't obtain good segmentation if using it without enhancement. Fig.1(b) demonstrates the image enhanced by homomorphic filtering, the contrast is improved and the details in face region are enhanced.

B. Structure image segmentation

We divide the features of interest from the skin and the background by clustering the gray level image into three clusters through the K-Mean Clustering algorithm. The lightest gray level representing the background or other light pixels is set to 255, the intermediate representing the skin is set to 128, and the darkest representing both the features and other dark pixels of the image (for example the hair and the beard) is set to

0. Fig.1(c) shows the facial image processed by K-Mean clustering algorithm.

After clustering, a threshold is set to 128. Then a binary image is obtained, which obviously reflects the face structure. Considering the non-face area can influence the speed and the results of template matching, the oversize black area which is useless in the binary image is eliminated by the conventional connected components labeling process. Then the final

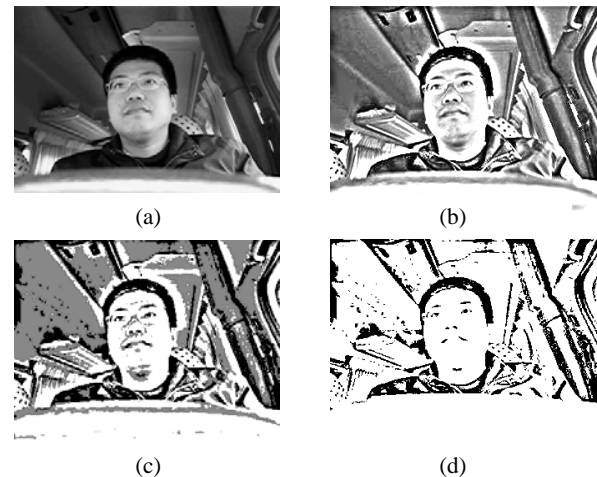

(a)　　　　　　　　　　(b)

(c)　　　　　　　　　　(d)

Fig. 1 An example of preprocessing (a) Original image (b) Enhanced image (c) Clustered image (d) Structure image

structure image is obtained, as shown in Fig.1(d).

C. Binary template matching

In order to determine the set of rows which contains the eyes, we apply the template matching to the structure image in order to search possible eye pairs [10]. We adopt a binary template which models the two eyes in a very rough way but clearly embody the structure of eye region. A single template has been used for all the images which are of different size, thus showing a desirable scale independence property.

Among the positions with the high cross correlation, all the eye pair candidates are extracted. In the proposed method, some in depth rotation of the face depth or rotation on-the-plane of the image are permitted as long as both eyes of the face are visible. Fig.2 illustrates the eye pair candidates selection.

Fig. 2 Eye pair candidates selection

D. Binary template matching

For the purpose of getting the successful eye location rate, the proposed method is followed by a simple eyes verifier. We use eyes verifier instead of face verifier proposed in some

World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:4, No:12, 2010

literatures [11], because the eye pattern is considered reliable enough to confirm the existence of the face. We also consider that there may be occlusions on the region below eyes, and the exaggerated facial expressions will also influence the results if using face verifier.

In this paper, we choose the SVM as the classifying function. One distinctive advantage this type of classifier has over traditional neural networks is that SVMs achieve better generalization performance.

Support vector machine is a patter classification algorithm developed by V. Vapnik and his team [12]. It is a binary classification method that finds the optimal linear decision surface based on the concept of structural risk minimization. The decision surface is a weighted combination of elements of the training set. These elements are called support vectors and characterize the boundary between the two classes. Given a set of N examples:

$$(x_1,y_1), ...(x_i,y_i), ...(x_N,y_N) \quad x_i \in \mathbb{R}^N, y_i \in \{-1,1\}$$

In case of linear separable data, maximum margin classification aims to separate two classes with hyperplane that maximizes distance of supports vectors. This hyperplane is called OSH (Optimal Separating Hyperplane). OSH can be expressed as in equation (4):

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i (x_i^T x) + b \qquad (4)$$

This solution is defined in terms of subset of training samples (supports vectors) whose α_i is non-zero.

In the case of linearly non-separable patterns, SVM is to perform non-linear mapping of input vector into high dimensional dot product space F. This is called the feature space. In this feature space, we can exploit the linear algorithm mentioned in the previous part, but with a difference. The separating hyperplane is now defined as a linear function of vectors drawn from the feature space rather than the original input space. In general, however, the dimension of the feature space is very large, so we have the technical problem of computing high dimensional spaces. Kernel method gives the solution to this problem. In equation (4), substituting $x_i^T x$ to $\varphi^T(x_i)\varphi(x)$ leads to the following formula:

$$f(x) = sgn[\sum_{i=1}^{N} y_i \alpha_i \varphi^T(x_i)\varphi(x) + b] \qquad (5)$$

In SVM, all mappings occur in form of inner product. Hence, kernel method is simply applied to SVM by replacing all occurrences of $\varphi^T(x_i)\varphi(x)$ by $k(x_i,x)$ rather than using mapping φ explicitly. This kernel method is backed up by Mercer's theorem. Thus the formula for non-linear SVM with kernel is

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i k(x_i,x) + b \qquad (6)$$

The requirement on the kernel $k(x_i,x)$ is to satisfy Mercer's theorem. Within this requirement there are some possible inner product kernels. There are Gaussian Radial Basis Functions (RBFs), polynomial functions, and sigmoid polynomials whose decision surfaces are known to have good approximation properties. In this paper, we choose Gaussian radial basis function as the kernel function.

All the eye pair candidates (gray level images) are extracted according to the results of binary template matching. Then they are normalized into the size of 20×8 pixels and verified by using SVM to obtain real eye pairs.

The training data used for generating eye verification SVM consists of 400 images of each class (eye-pair and non-eye-pair). Non-eye-pair images are more various than eye pair images, so it is difficult to select standard images. Selection of proper non-eye-pair images is very important to train SVM because performance of SVM is influenced by what kind of non-eye-pair images are used. In the initial stage of training SVM, we use non-eye-pair images similar to eye pair such as eyebrows, nostrils and other eye-pair-like patches as eye pair, as shown in Fig. 3. Fig.4 shows the results of eye pair location. After the real eye pairs are obtained, the eye regions are located according to the position of each eye pair.
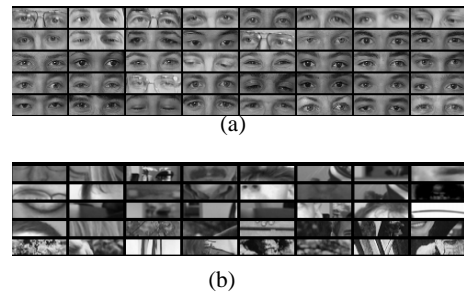

(a)

(b)

Fig. 3 Training data for eye pair classifier (a) Positive samples (b) Negative samples

Fig. 4 Eye pair location

III. PRECISE EYE LOCATION

After eye pairs are located, two eyes can be precisely located in the eye pair images by using vertical projection and eye classifier.

A. Binary vertical projection

We apply binary vertical projection to binary eye pair images. Binary eye pair images embody the structure of eye region clearly. Therefore, vertical centers of two eyes can be located precisely according to the projection results. The vertical center is the center of two boundaries, as shown in Fig.5.

The horizontal projection is not used to obtain the horizontal center of eyes and locate two eyes, because left eye and right eye are not in a straight line when driver's head slants, and then the horizontal projection is not applicable.
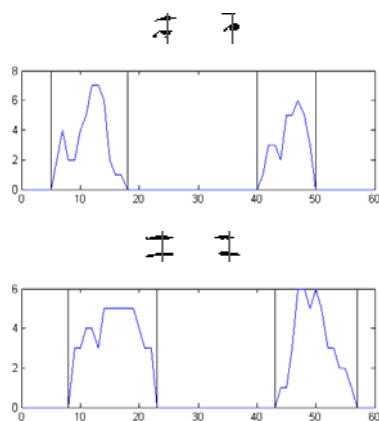
World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:4, No:12, 2010

Fig. 5 Binary vertical projection

B. *Precise eye location by using eye classifier*

We also apply SVM classifier to precisely locate the eyes. A square window is moving from top to down along the left vertical center lines in grey level eye pair image. Then the window image is sent to the SVM classifier so that the left eye is located. Another window is moving from top to down along the right vertical center line in grey level eye pair image to locate right eye in the same way. The side length of the window is 2/3 of the width of eye pair image.

The training data used for generating eye classification SVM consists of 600 images of each class (eye and non-eye). Selection of proper non-eye images is very important to train SVM because performance of SVM is influenced by what kind of non-eye images is used. We use non-eye images similar to eyes such as eyebrows, and other eye-like patches, as shown in Fig. 6.
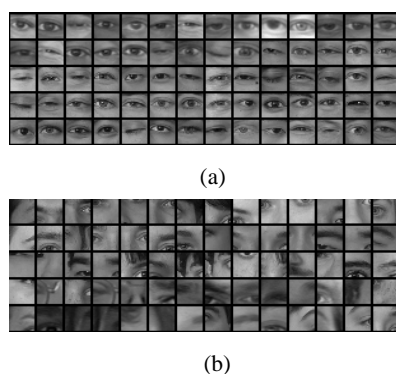

(a)

(b)

Fig. 6 Training data for eye classifier (a) Positive samples (b) Negative samples

In this section, we present our experiment results. The proposed method is tested on the grey level face images obtained in the real-word driving conditions. The images are 320×240 pixels.

From the database, we randomly selected 200 face images of people, different poses, illuminations and expressions, which two eyes are present in the images.

IV. EXPERIMENTAL RESULTS

Experimental results show that the proposed method is robust enough to work well in different poses, illuminations and different eye states. Some results are shown in Fig.7. The eye pair location rate is 96.7%. Our method has the high accuracy with a correct eye location rate of 99.2% based on eye pair location. The high location rate will lead to accurate eye states recognition and eye tracking.

Fig. 7 Experimental results

V. CONCLUSION

In this paper we have presented an efficient eye location algorithm for driver fatigue monitoring, which is applicable in the real-world condition. The structure of eye region is used as a robust cue for finding eye pair candidates. Eye pair candidates are then sent to classifier to get real eye pair. Then eyes are precisely located by using binary vertical projection and eye classifier. The proposed method can deal with illumination changes, moderate rotations, glasses wearing and different eye states. The future research will focus on eye tracking and eye state recognition for driver fatigue monitoring, which are based on eye location.

REFERENCES

[1] S.Y. Zhao, and R.R. Grigat, "Robust eye detection under active infrared illumination," Proceedings of International Conference on Pattern Recognition, vol.4, pp. 481-484, 2006.
[2] H. Huang, Y.S. Zhou, and et al., "An optimal eye locating and tracking systems for driver fatigue monitoring", Proceedings of the International Conference on Wavelet Analysis and Pattern Recognition, pp. 1144 -1148, 2007.
[3] W. Di, R.B. Wang, and et al., "Driver eye feature extraction based on Infrared illumination", Proceedings of IEEE Intelligent Vehicles Symposium, pp.330-334, pp. 2009.
[4] R Hsu, M. Mottleb, and A. K. Jain, "Face detection in color images", IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 4, No. 5, pp. 696–706, 2002.

[5] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 20, No. 1, pp 23–38, 1998.

[6] P.Viola, M.J.Jones, "Robust real-time face detection", International Journal of Computer Vision. Vol.57, No.2, pp.137-154, 2004.

[7] T. Kawaguchi, and M. Rizon, "Iris detection using intensity and edge information," Pattern Recognition. vol. 36, pp.549–562, 2003.

[8] A. Tauseef, and K. Intaek, "An improved eye location algorithm using multi-cue facial Information", Proceedings of International Conference on Computer, Control and Communication, pp.1-6, 2009.

[9] JF Ren, and XD. Jiang, "A method for accurate localization of facial features", Proceedings of International Conference on Image Processing, pp.2733-2736, 2009.

[10] Q. Wang, WK Yang, and et al., "Face detection using binary template matching and SVM", LNAI4099, pp.1237-1241, 2006.

[11] J.X Wu, and Z.H Zhou, "Efficient face candidates selector for face Detection," Pattern Recognition, 2003, Vol. 36, pp. 1175–1186.

[12] V. Vapnik, "The nature of statistical learning theory," New York: Springer-Verlag, 1995.