Automatic Eye Detection and Its Validation

Peng Wang, Matthew B. Green, Qiang Ji
Department of Electrical,Computer and System Engineering
Rensselaer Polytechnic Institute
Troy, NY 12180

James Wayman
San José State University
San José, CA 95192

Abstract

The accuracy of face alignment affects the performance of a face recognition system. Since face alignment is usually conducted using eye positions, an accurate eye localization algorithm is therefore essential for accurate face recognition. In this paper, we first study the impact of eye locations on face recognition accuracy, and then introduce an automatic technique for eye detection. The performance of our automatic eye detection technique is subsequently validated using FRGC 1.0 database. The validation shows that our eye detector has an overall 94.5% eye detection rate, with the detected eyes very close to the manually provided eye positions. In addition, the face recognition performance based on the automatic eye detection is shown to be comparable to that of using manually given eye positions.

1 Introduction

An important issue in face recognition systems is face alignment. Face alignment involves spatially scaling and rotating a face image to match with face images in the database. It is already shown that the face alignment has a large impact on recognition accuracy [17, 15]. Currently, face alignment is usually performed with the use of eye positions. For most face recognition methods, eye positions are manually given. But for a real world application of face recognition, manually detecting eye positions is apparently not realistic. An automatic eye detection algorithm is therefore needed for a fully automatic face recognition system.

In this paper, we first propose a new real time automatic eye detection method. Our eye detection method is then validated using FRGC database [16]. The rest of this paper is organized as follows: the impact of eye location on face recognition is discussed in Section 2. The related work on automatic eye detection is reviewed in Section 3. We propose an accurate eye localization algorithm at Section 4. In Section 5, we show the experiment results of validating our eye localization for face recognition on the FRGC database. The paper concludes in Section 6.

2 Eye Detection Error on Face Recognition

To observe how the recognition performance varies according to eye localization error, the eye positions of the ground truth are artificially perturbed with random noise. Face recognition is then performed using the perturbed eye positions. The impact of eye detection on recognition accuracy is illustrated in Figure 1, where face images are aligned based on perturbed eye positions. The eye localization error in Figure 1 is the pixel error normalized by the distance between two eyes. Given a normalized error, the random noise is uniformly distributed at a circle in 2D space. The data from FRGC 1.0 and PCA baseline algorithm are used for this experiment [16].

Figure 1 clearly shows that eye location errors significantly affect the recognition accuracy. For example, about 1% (about 3 pixels for FRGC image or 0.5 pixel if the interocular distance is 50 pixels) eye location error reduces the face recognition accuracy by over 10%. When the error is about 5%, the face recognition accuracy reduces by 50%. These numbers, of course, vary, depending on the recognition methods. But still, they show the significant impact of eye position error on face recognition.

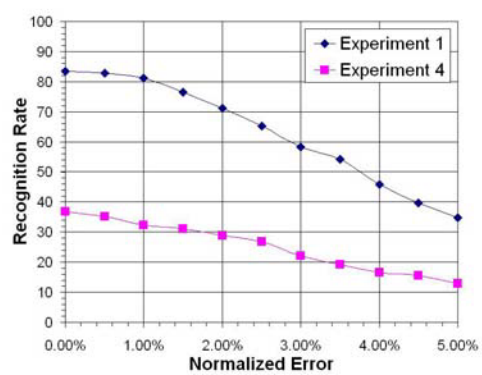


Figure 1: Face recognition accuracy vs. eye localization error in FRGC 1.0 experiments

Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)
1063-6919/05 $20.00 © 2005 IEEE

Similar conclusions have also been drawn by other researchers. In [17], the face recognition algorithms with automatic and manual alignment are evaluated. The face recognition algorithms with manual eye coordinates are referred as “partially automatic algorithms” while “fully automatic algorithms” automatically align face images. Experiment results show that the partially automatic algorithms always perform better than the fully automatic algorithms.

There are two basic strategies to address the face alignment problem. One strategy is to improve the robustness of face recognition algorithms to misalignment. The robustness of a face recognition algorithm is evaluated by its performance under misalignment[19]. [17] shows that EBGM method, which is based on Gabor wavelets representation, is more robust for fully automatic face alignment than PCA method. It is also claimed that the warping is a necessary step in face recognition to improve the robustness to imprecise alignment [15].

In this paper, we focus on another strategy, which tries to improve eye localization accuracy for better face alignment. We propose a new eye detection method and validate the method using FRGC database.

3 Brief Review on Automatic Eye Detection

There are two purposes of eye detection. One is to detect the existence of eyes, and another is to accurately locate eye positions. Under most situations, the eye position is measured with the pupil center.

Current eye detection methods can be divided into two categories: active and passive eye detection [11]. The active detection methods use special types of illumination. Under IR illumination, pupils show physical properties which can be utilized to localize eyes [9, 25]. The advantages of active eye detection methods are that they are very accurate and robust. The disadvantages are that they need special lighting sources and have more false detections with an outdoor environment, where the outdoor illumination impacts the IR illumination.

Passive methods directly detect eyes from images within visual spectrum and normal illumination. Some early work extracts distinct features from eyes localization. The features include image gradients [13], projection [24], and templates [4, 12]. However, in these methods, heuristics and postprocessing are usually necessary to remove false detections, and these features are sensitive to image noise. Besides the above features, wavelets are shown to be able to localize facial features [10, 23]. Huang and Wechsler propose to select optimal Wavelet packets and classify the eye and non-eye with Radial Basis Functions (RBFs) [10]. Gabor wavelets are robust to moderate illumination change, and the similarity measurement based on Gabor wavelets is sensitive to localization change so that they can be used to detect fiducial points [23]. In [6], a two-layer Gabor wavelet network (GWN) is proposed to localize facial points from coarse to fine. The first layer localizes face region while the second layer further refines the facial points. The experiments on FERET show that about 95% eyes are located with a distance error smaller than 3 pixels. However, no face recognition accuracy is reported with this method.

Some passive methods consider eye detection as a typical two-class pattern recognition problem. In [14], eye detectors are trained with the rectangle Haar features and AdaBoost algorithm to detect eyes in images. In [5], the critical features are selected from both rectangle and center-surrounded Haar feature sets. GentleBoost is applied to construct a final eye detector. The same algorithm to train a frontal face detector. After a frontal face is detected, eyes are located inside the face region.

4 Our Eye Localization Algorithm

It is shown that applying Haar wavelet features in AdaBoost provides excellent computational efficiency with comparable accuracy with other methods for face detection [20]. A disadvantage of Haar features is their limited discriminant capability. Although the Haar features vary with different patterns, sizes and positions, they can only represent the regular rectangular shapes. However, for eye detection, the most distinguishing feature is the pupil which has a round shape.

To better represent eyes, we propose to statistically learn discriminate features to characterize eye patterns. Based on the distribution of discriminant features, we propose to learn probabilistic classifiers to separate eyes and non-eyes. Multiple classifiers are then combined in AdaBoost to form a robust and accurate eye detector. Our algorithm is briefly explained at the following paragraphs.

4.1 Discriminant Features for Eye Detection

The notations used in this paper are explained here. A training sample is denoted as (x, g_x), where x is image intensity data, and $g_x \in \{\Omega_1, \Omega_2\}$ is the sample label. In this paper, $\Omega_1 = 1$ represents the eye while $\Omega_2 = -1$ represents the non-eye. Each training sample (x, g_x) is associated with a weight w_x.

One criteria to extract a “good” feature for pattern classification is that the feature $F(x)$ can minimize the estimated Bayes error J_F:

$$J_F = \int (1 - \max_i [p(\Omega_i | F(x))]) p(x) dx \tag{1}$$

It is shown that Fisher discriminant analysis (FDA) is equivalent to Bayesian classifier if assuming Gaussian distribution and equivalent priors and covariance matrix for each class. FDA extracts the feature $z = A^T x$ by maximizing the ratio $J(A)$ of between-class covariance S_b and within-class covariance S_w (2).

$$J(A) = \frac{||A^T S_b A||}{||A^T S_w A||} \tag{2}$$

When the samples are associated with weights, the covariance matrices have the form

$$\begin{aligned} S_w &= \sum_{x \in \Omega_i} w_x (x - \mu_i)(x - \mu_i)^T \\ S_b &= \sum_{i \in \{1,2\}} P(\Omega_i)(\mu_i - \mu)(\mu_i - \mu)^T \end{aligned} \tag{3}$$

where w_x is the sample weight, μ is the mean of all samples, and μ_i is the mean of i-th class. $P(\Omega_i) = \sum_{g_x = \Omega_i} w_x$ is the weight of class i. The FDA feature can be obtained by solving the generalized eigenvalue and eigenvectors problem.

One problem with FDA is that the single Gaussian assumption is not valid due to significant appearance variance, especially for non-eyes. Another problem is that the rank of S_b in FDA is 1 for a two-class problem, which means that only 1 effective feature can be extracted from FDA.

Nonparametric discriminant analysis (NDA) is proposed to overcome these limitations [8]. In NDA, each sample has the extra-class nearest neighbors(NNs) $x^E_{NN} = \{\hat{x} | g_{\hat{x}} \neq g_x, ||\hat{x} - x|| < c^E_x\}$ and intra-class nearest neighbors $x^I_{NN} = \{\hat{x} | g_{\hat{x}} = g_x, ||\hat{x} - x|| < c^I_x\}$ where the thresholds c^E_x and c^I_x define the extra and intra neighborhoods respectively. In calculation, the NNs are usually represented by their weighted average, i.e. $x^E = E[x^E_{NN}]$ and $x^I = E[x^I_{NN}]$. The nonparametric between-class scatter matrix is defined as (4).

$$S'_b = E_x[\gamma_x (x - x^E)(x - x^E)^T] \tag{4}$$

$$\gamma_x = \frac{min(||x - x^E||^\alpha, ||x - x^I||^\alpha)}{||x - x^E||^\alpha + ||x - x^I||^\alpha} \tag{5}$$

The NDA weight γ_x is introduced to emphasize those samples near the class boundary, and α is the control parameter. The NDA weight is close to 0.5 if the sample is near the class boundary, and tends to 0 if the sample is inside the class.

NDA obtains a full-rank between-class scatter matrix from local data so that it provides multiple features [8]. Also, the scatter matrix does not assume any distribution, but only depends on the data near class boundary. We further propose a recursive nonparametric discriminant feature

---

- Initialize sample weights.
- Repeat for $t = 1, 2, ..., N$:
  1. Extract discriminant feature $z = a^T x$ with sample weights w_x. Learn the sample distribution from z.
  2. Fit the classifier $h_t(x) = \frac{1}{2} log[\frac{P(\Omega_1|x)}{P(\Omega_2|x)}]$ from sample weights.
  3. Update $w_x \leftarrow w_x \exp[-g_x h_t(x)]$ and re-normalize the weight.
- Output the combined classifiers $sgn[\sum_{t=1}^{N} h_t(x)]$.

Figure 2: Applying discriminant features in Real AdaBoost

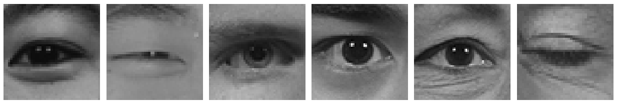
(RNDA) to speed up the NDA for object detection. For more detail, please refer to the work in [22].

4.2 Feature Selection and Classifier Construction with AdaBoost

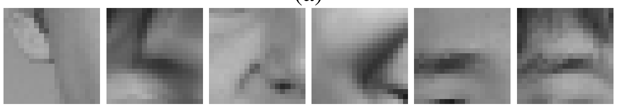
AdaBoost is very popular for object detection since its first application in face detection [21]. Basically, AdaBoost selects the critical features and train weak classifiers as well as updates the sample weights [18]. As long as the weak classifiers are slightly better than random guessing, the final classifier will have much better accuracy after combining all the weak classifiers together. The summary of AdaBoost algorithm can be found in [18, 7].

The main task in the AdaBoost is the selection of features to learn weak classifiers. We use more powerful discriminant features instead of rectangular Haar features to improve eye detection accuracy. Since the data weights in both discriminant analysis and AdaBoost represent the same distribution, they can be associated together. The algorithm applying discriminant features in AdaBoost is summarized in Figure 2.

To train a robust eye detector, we have collected training data from various sources. 500 pairs of eyes were collected from FERET images [17]. More eye images were collected from the web in order to include more variance from the real world. The eyes were randomly rotated with small angles. In total, thousands of eyes have been collected for training. In application, only a left eye detector is trained due to the symmetry of eyes. In detection, the images are flipped to find the right eyes. The non-eye images were randomly collected from background images. More non-eyes were collected from the false detections. Those false de-



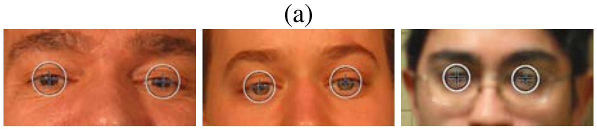
(a)



(b)

Figure 3: Some eye and non-eye images used in training. (a): some eye images. (b): typical non-eye training images

(a)



(b)

Figure 4: Localized eyes from face images. (a): face and eye detection results. (b). enlarged eye localization results.

tections were fed back for training. Some typical eye and non-eye images are shown in Figure 3.

To improve the eye detection speed, a cascade structure is applied [20]. The first layer in the cascade only has two features yet it can remove 80% of the non-eye samples. The resulting eye detector classifier uses less than 100 features.

4.3 Eye Localization

Our eye localization method follows a hierarchical principle. Firstly a face is detected, then eyes are located inside the detected face. The face detection method is also based on AdaBoost, which is introduced in [22]. Geometric constraints are applied to localize eyes, which means eyes are only searched in the top half of a face. Usually, there are multiple eyes detected around the pupil center. The final eye localization is the average of the multiple detection results. The whole systems run at around 10 fps at a P4 2.6G PC. Examples of eye detection results are shown in Figure 4.

5 Eye Detection Validation

To quantitatively validate the performance of our eye detection method, we performed two experiments. In the first experiment, we compared the detected eye positions with the manually labeled eye positions. The performance of our eye detector is characterized by the eye detection rate and eye localization error. The localization error is measured as the Euclidean distance between the detected eye positions and manual eye positions. In the second experiment, we quantify the performance of our eye detection based on its influence on face recognition accuracy of two standard baseline methods: PCA and PCA+LDA. For both experiments, FRGC 1.0 database is used.

5.1 Eye Detection Accuracy

We apply our eye detection method to all of the 2D images in the FRGC 1.0 database. The frontal face detection rate is approximately 95.0%. Usually the missing faces are caused by uncontrolled illumination. For eye detection alone, it achieves a detection rate about 99.0% on the detected faces. As a result, we have an overall 94.5% eye detection rate for FRGC 1.0. Table 1 shows the horizontal and vertical eye localization errors, as well as the total error. Additionally, Table 1 shows both pixel and normalized errors, where the normalized error is the pixel error normalized by the distance between two eyes. The average Euclidean distance between automatic eyes and ground truth is about 6.4 pixels, which accounts for 2.67% normalized error. The distribution of the Euclidean distance of detected eyes compared to the ground truth is shown at Figure 5.

Error	horizontal		vertical		Euclidean
	(mean)	(std)	(mean)	(std)	distance (mean)
Pixel	4.9914	4.5808	3.1652	2.6927	6.4016
Normalized	2.04%	1.96%	1.31%	1.35%	2.67%

Table 1: Eye localization error on FRGC 1.0.

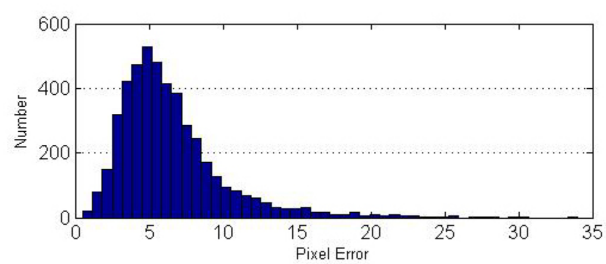


Figure 5: Distribution of eye localization pixel errors.

The comparison of eye localization using Haar features and proposed discriminant features is shown in Figure 6. In this figure, the horizontal axis is the normalized localization error, and the vertical axis is the accumulated distribution, which means the percentage of eyes with smaller normalized error than the corresponding horizontal value. From Figure 6, it is observed that the eye localization based on RNDA features has much smaller localization error than that based on Haar features. Some commercial products, such as FaceIt [1] and Viisage [2], also provide eye localizations. However, such products are unavailable to us for comparison .

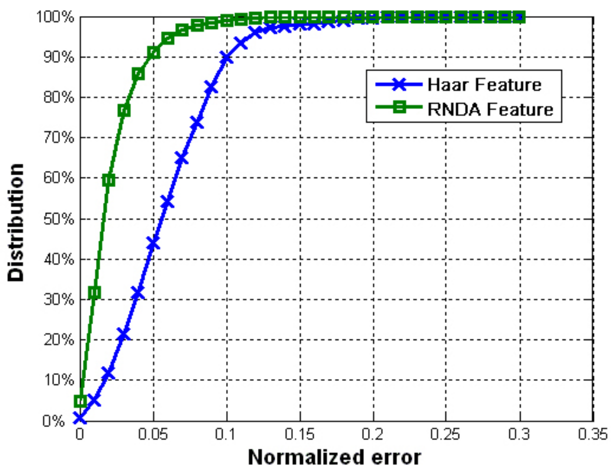


Figure 6: Comparison of accumulated eye localization errors with Haar feature and RNDA feature.

5.2 Face Recognition Experiment

Since the accuracy of manually provided eye positions is not confirmed and that nobody knows where the real eye positions are, it is not convincing to quantify the performance of our eye detector using the distance between the manual eye positions and the detected eye positions. To further validate the performance our eye detector, we decide to apply it to face recognition and use the accuracy of face recognition to judge its performance. Specifically, face recognition testing was performed using experiments 1, 2 and 4 from FRGC 1.0. In each of the three experiments, 608 probe images were compared against 152 gallery images. In experiment 1, single controlled still images were compared against single controlled images. The experiment utilized a set of 183 training images to train the classifiers. In Experiment 2 multiple controlled images were compared against multiple controlled images. In this experiment 732 images were used for training. Finally, experiment 4 compared uncontrolled single images to controlled single images. This experiment used 366 images for training.

For this experiment, a baseline PCA method, and a baseline PCA+LDA method was used. Both methods are based on Colorado State University's Face Identification Evaluation System [3]. In both PCA and PCA+LDA, a simple cutoff filter was used to retain 40% of the eigenvectors.

Using these face recognition methods, the faces and eyes are automatically localized from the FRGC images. For the purpose of validating our eye localization method, we eliminated the effects of the missed face detections. When the face detector failed to locate a face, a face region was simulated with the given eye positions. In the FRGC experiments, there are approximately 95.0% automatic face detections and 5.0% simulating face detections. Based on the face detection results, eye localization was successful on over 99.0% of the FRGC images. If an eye was missed, the manually marked eye position was used. In total, less than 15 eyes out of all 4715 FRGC 1.0 samples were manually marked.

Using the location of the eyes, the images were first clipped, rotated, and scaled to a fixed image size. An elliptical mask was then used to remove extraneous background components. Finally, a standard histogram equalization was performed on the faces, and then the recognition algorithms were applied.

The comparison of face recognition results with automatic and manual eye localizations are summarized in Figure 7. The CMC curves for each experiment are shown in Figure 8.(a)-(c). The ROC curve is given in Figure 8.(d).

5.2.1 Experiment 1

The recognition results of experiment 1 are shown in Figure 8.(a). In FRGC 1.0 experiment 1, the automatic recognition results are very similar to the manually marked recognition results. The PCA face recognition result for manually marked eye locations was 83.30%, and the fully automated eye locations resulted in a recognition rate of 81.75%. The difference between automatic and manually marked points is only due to recognition errors in 10 out of 608 images. By looking at the ROC curve we can confirm that the results for automatically marked eye positions produce very similar results to manually marked points.

The PCA + LDA recognition results were slightly poorer than the pure PCA results. However, the automatically marked points still had very similar performance when compared to the manually marked points.

5.2.2 Experiment 2

The recognition results of experiment 2 are shown in Figure 8.(b). FRGC 1.0 experiment 2, had remarkably similar performance between the automated and the manual recognition results. The PCA face recognition result for manually

Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)
1063-6919/05 $20.00 © 2005 IEEE

marked eye locations was 97.04% and the fully automated eye locations resulted in a recognition rate of 96.38%.

The difference between automatic and manually marked points is only due to recognition errors in 4 out of 608 images. In fact, the automatic face detection missed only 22 out of 608 total faces. The difference between automatic and manually marked eyes for PCA + LDA was nearly the same as those for regular PCA.

5.2.3 Experiment 4

The recognition results of experiment 4 are shown in Figure 8.(c). Experiment 4 was significantly more difficult for face recognition than the other two experiments. The PCA face recognition result for manually marked eye locations was 36.84%, and the fully automated eye locations resulted in a recognition rate of 25.82%. The results for automated detection in this experiment compared to manually marked points are poorer than the results of the previous two experiments. The PCA + LDA recognition differences between automated and manually marked points are once again similar to the PCA results.

5.3 Discussion

We can conclude from the experiments that the recognition results for the fully automated eye localization and face recognition are comparable to the manually marked tests. The recognition results of those two tests are very close, e.g. 83.30% vs. 81.76% for experiment 1 and 97.04% vs. 96.38% for experiment 2. Considering that usually the subspace methods are sensitive to misalignment, our proposed eye localization method is very successful in these two experiments.

Compared with experiments 1 and 2, experiment 4 shows poorer recognition results. In experiment 4, probe images are taken under uncontrolled environments while gallery images are taken under controlled environment. For this experiment, the eye positions for probe images have higher errors, which results in the decrease of recognition accuracy. In many applications, the quality of gallery images can often be controlled, which could alleviative this problem.

We observed that in experiment 1, even with about 2.67% normalized errors, our automatic eyes show better recognition accuracy than the synthetic eyes with the uniformly distributed noise in Figure 1. This is because the errors produced by the eye detector are more consistent than the uniform noise so that the smaller variance of eye detections can produce better recognition accuracy.

In addition, we also noticed that a few images with poorly detected eyes can often decrease the average performance of our eye detection. The average eye detection can be improved if those poor eye detections can be identified and removed from subsequent face recognition. Our future research will address this problem by associating a confidence measurement with each detected eye.

6 Conclusion

In this paper, we introduce an automatic eye detection technique. Experimental results are then provided to show the validation of the eye detector using FRGC 1.0 database. The results show that face recognition based on the automatic eye localization has comparable accuracy with the face recognition based on manual eye positions. This demonstrates that our proposed eye localization method can be incorporated into a fully automatic face recognition system. Future work will further improving our eye localization method under uncontrolled environments.

References

[1] *http://www.identix.com*.

[2] *http://www.viisage.com*.

[3] Ross Beveridge, David Bolme, Marcio Teixeira, and Bruce Draper, *The csu face identification evaluation system users guide:version 5.0*, Computer Science Department, Colorado State University, May 2003.

[4] T. d'Orazio, M. Leo, G. Cicirelli, and A. Distante, *An algorithm for real time eye detection in face images*, ICPR, 2004, pp. 278–281.

[5] Ian Fasel, Bret Fortenberry, and Javier Movellan, *A generative framework for real time object detection and classification*, Computer Vision and Image Understanding **98** (2005), no. 1, 182–210.

[6] R.S. Feris, J. Gemmell, K.Toyama, and V. Kruger, *Hierarchical wavelet networks for facial feature localization*, IEEE International Conference on Automatic Face and Gesture Recognition, 2002, pp. 118–123.

[7] J. Friedman, T. Hastie, and R. Tibshirani, *Additive logistic regression: a statistical view of boosting*, The Annals of Statistics **28** (2000), no. 2, 337–374.

[8] K. Fukunaga, *Nonparametric discriminant analysis*, IEEE Transactions on Pattern Analysis and Machine Intelligence **5** (1984), no. 1, 671–678.

[9] A. Haro, M. Flickner, and I. Essa, *Detecting and tracking eyes by using their physiological properties, dynamics, and appearance*, IEEE International Conference on Computer Vision and Pattern Recognition, vol. 1, 2000, pp. 163–168.

[10] Jeffrey Huang and Harry Wechsler, *Eye detection using optimal wavelet packets and radial basis functions (rbfs).*, International Journal of Pattern Recognition and Artificial Intelligence **13** (1999), no. 7, 1009–1026.

[11] Qiang Ji, Harry Wechsler, Andrew Duchowski, and Myron Flickner, *Special issue: eye detection and tracking*, Computer Vision and Image Understanding (2005), 1–3.

[12] T. Kawaguchi, D. Hidaka, and M. Rizon, *Detection of eyes from human faces by hough transform and separability filter*, ICIP, vol. 1, 2000, pp. 49–52.

[13] R. Kothari and J.L. Mitchell, *Detection of eye locations in unconstrained visual images*, ICIP, vol. 3, 1996, pp. 519–522.

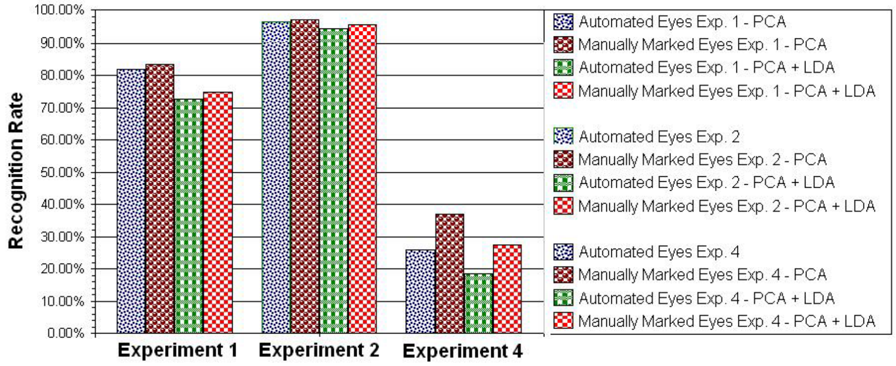
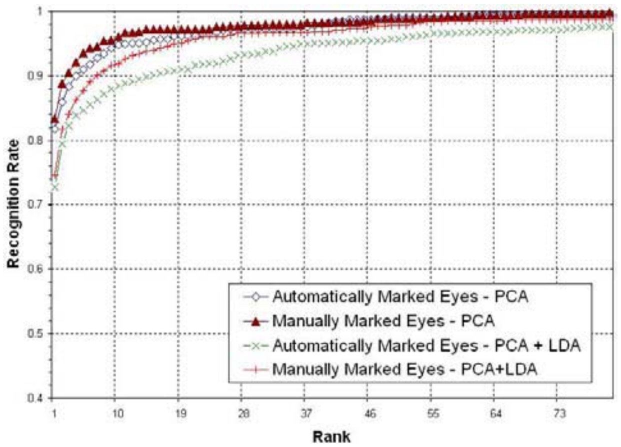
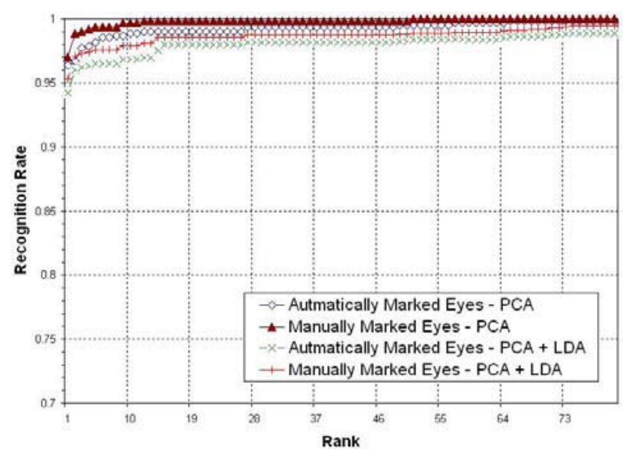


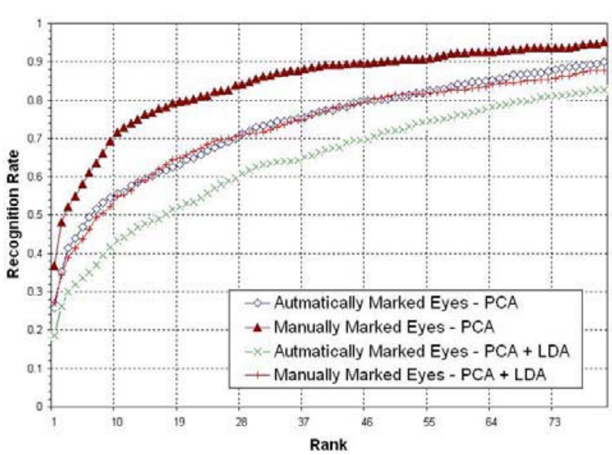
Figure 7: Recognition summary with manual and automatic eye positions: PCA and PCA+LDA baseline methods.



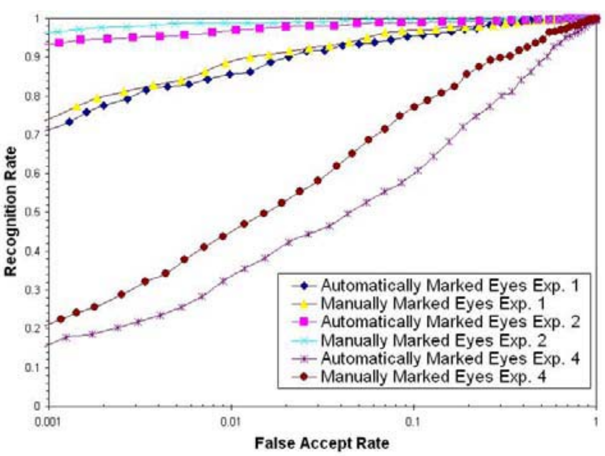
(a)



(b)



(c)



(d)

Figure 8: Recognition results with manual and automatic eye positions on FRGC 1.0. (a):PCA and LDA baseline methods for experiment 1. (b):PCA and LDA baseline methods for experiment 2. (c):PCA and LDA baseline methods for experiment 4. (d):ROC curves for automated and manually located eyes with PCA baseline method

[14] Y. Ma., X. Ding, Z. Wang, and N. Wang, *Robust precise eye location under probabilistic framework*, IEEE International Conference on Automatic Face and Gesture Recognition, 2004, pp. 339–344.

[15] A.M. Martinez, *Recognizing imprecisely localized, partially occluded, and expression variant faces from a single sample per class*, PAMI, IEEE Transactions on **24** (2002), no. 6, 748–763.

[16] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, *Overview of the face recognition grand challenge*, IEEE International Conference on Computer Vision and Pattern Recognition, 2005.

[17] P.J. Phillips, Hyeonjoon Moon, S.A. Rizvi, and P.J. Rauss, *The feret evaluation methodology for face-recognition algorithms*, IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000), no. 10, 1090–1104.

[18] R. E. Schapire, *A brief introduction to boosting*, Proc. of the Sixteenth International Joint Conference on Artificial Intelligence, 1999, pp. 246–252.

[19] Shiguang Shan, Wen Gao, Yizheng Chang, Bo Cao, and Pang Yang, *Review the strength of gabor features for face recognition from the angle of its robustness to mis-alignment*, International Conference on Pattern Recognition, 2004, pp. 338–341.

[20] Paul Viola and Michael Jones, *Robust real-time object detection*, International Journal of Computer Vision **57** (2004), no. 2, 137–154.

[21] ———, *Robust real-time object detection*, Interntional workshop on statistical and computational theories of vision **57** (2004), no. 2, 137–154.

[22] Peng Wang and Qiang Ji, *Learning discriminant features for multiview face and eye detection*, IEEE International Conference on Computer Vision and Pattern Recognition, 2005.

[23] L. Wiskott, J.-M.Fellous, N. Kruger, and C. von der Malsburg, *Face recognition by elastic bunch graph matching*, IEEE Transactions on Pattern Analysis and Machine Intelligence (1997), 775–779.

[24] Z.H. Zhou and X. Geng, *Projection functions for eye detection*, Pattern Recognition **37** (2004), no. 5, 1049–1056.

[25] Zhiwei Zhu, Qiang Ji, and Kikuo Fujimura, *Combining kalman filtering and mean shift for real time eye tracking under active ir illumination*, International Conference on Pattern Recognition, 2002, pp. 318–321.