Impact of Occlusion Removal on PCA and LDA for Iris Recognition

Eric Anden
Department of Electrical Engineering
Stanford University
Stanford, CA, USA
eanden@stanford.edu

Abstract— **An iris recognition algorithm is implemented using morphological image processing, Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA). The suggested occlusion removal technique is implemented using erosion and dilation operators, and compared against a training set without occlusion removal and one that only keeps one half of the iris. The resulting comparison shows that occlusion removal does not have a significant impact on the iris recognition success rate, when using the PCA & LDA implementation.**

I. INTRODUCTION

As biometrics plays an increasingly important role in security applications today [1], efficient and robust iris recognition algorithms become all the more relevant and can be implemented using various image processing techniques. The iris of a human eye contains more unique features than all other biometric methods, which makes it ideal for identification [1]. However, when eyelashes and eyelids conceal the iris features in the image, the iris recognition success rate can be severely degraded [2].

This article describes an image processing algorithm that uses morphological image processing for occlusion removal. Morphological image processing is especially powerful for this application, as it can efficiently identify edges and features in the image, such as the pupil.

Once the image of the iris is processed and normalized, PCA and LDA techniques are used for iris recognition by database matching. PCA is useful as it allows for a reduction in the dataset to the most apparent features in the iris, and also allows for a more efficient computation of the Fisher LDA. The Fisher LDA is used to project the images into a space that maximizes scatter between classes, which is then used to match the test image to the training set.

II. PREVIOUS WORK

A. *Iris Recognition Techqniues*

In 1993, Daugman was the first to implement and patent an iris recognition algorithm [3], which has since been used as the foundation for iris detection and recognition. Although highly effective and with unmatched results, Daugman's algorithm is considered computationally inefficient [4]. Daugman's algorithm uses an integrodifferential operator to detect and segment the iris, and then a rubber-sheet model to normalize it [3]. Although the iris segmentation techniques vary between many implementations, the rubber-sheet model is widely used to preserve feature position with varying pupil and iris sizes. Using the normalized iris, Daugman then suggests using 2D Gabor filters to extract the phase information of the iris, and compile it a unique IrisCode for each individual. In 2009, Daugman tested his algorithm on the United Arab Emirates (UAE) border-crossing security system, with a false match rate of 1 in 200 billion [5].

Since Daugman's phased based implementation, other techniques have been suggested, including: wavelet transform using zero-crossings [6], texture analysis [7], PCA [4], Independent Component Analysis (ICA) [4], and LDA [8]. In order to implement a computationally efficient approach which maximizes the differences between the iris classes, the LDA approach as suggested by ul Haq et al. [8] is implemented in this article, which achieved results of 97% success rate using 92 subjects

B. *Occlusion Removal Techniques*

Daugman's algorithm takes into account eyelash and eyelid detection by using the intergrodifferential operator [3]. Ul Haq et al. [8] suggest either using the entire image of the iris, including occlusions, or only using the lower half of the iris, avoiding the often larger upper eyelashes. Other techniques proposed include: Canny edge detector and Hough transform [7] and finding the shortest path from two corners of the eye [9].

Several morphological techniques were suggested by Luo and Lin [10] and by Abdullah et al. [11]. Luo and Lin suggest

using a horizontal Sobel operator to detect eyelids, and then fitting a parabolic curve to the identified points to mark the eyelid border. Luo and Lin then use a difference of binary images to detect eyelashes [10]. Abdullah et al. use an 8x8 square structural element to remove eyelashes in the image by erosion [11].

III. PREPROCESSING

The first step of the iris recognition algorithm is to pre-process all of the sample images in the database. This ensures that only the iris portion of the image is compared against the testing sample, with as many occlusions as possible removed. Each of the steps A-F were processed on the first 9 samples of all 224 samples, which represents the training set. Figure 1 describes the functional flow of the pre-processing steps.

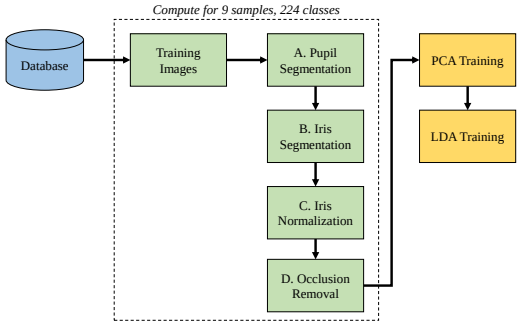

Figure 1 Pre-Processing Steps

A. *Pupil Segmentation*

Locating the pupil in the image is the most robust step in the pre-processing, and relies on morphological image processing. The pupil is often the darkest portion of the image sample, along with the eyelashes and other spots in the iris. To isolate the iris, the inverse image is binarized using a threshold of 0.9. In order to reduce the possibility of false positives, the binary image is eroded with a circular structural element, of radius 1. This helps to remove connections between the iris and the eyelashes.

The pupil can be identified by choosing the largest connected region in the binary image. If the eyelashes are particularly full and connected to the pupil, there is a possibility the pupil can be misidentified. However, the initial erosion this connection and produces the desired result. An example result is shown Figure 2.

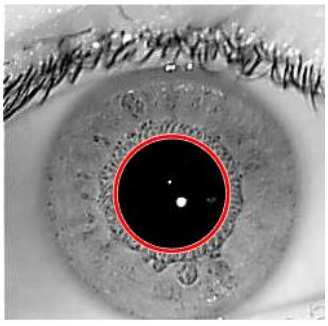

Figure 2 Example Pupil Segmentation for Class 1, Sample 1

B. *Iris Segmentation*

This step in the pre-processing uses the Circular Hough Transform (CHT) to identify circles that correspond to the iris in the image. The CHT result is effectively a convolution of the binary image with a set of circles of varying radii. The results of the convolution are stored in an accumulator array, which is used to detect the circles. This implementation uses the efficient use of edge orientation to make the computation more efficient, as suggested by Kimme et al. [12]

Using this implementation, it was found that a low edge detection threshold and high sensitivity was required to detect the iris edge. This was most likely caused by false matches for circular shapes, such as eyelids. The image was adjusted to high contrast and then processed with a median filter [8] to improve the visibility of edges. To decrease the possibility of a false detection, the CHT implementation used a predicted range. The steps below summarize in detail the steps followed.

1. Gamma correction using $\lambda = 3$ for high contrast
2. Process with median filter
3. Implement Hough Transform using predicted radii from 95 to 120 pixels
4. Verify offset from pupil center is less than 20 pixels
5. Choose the maximum radius that satisfies Step 4

If the steps outlined above do not produce an acceptable iris detection, the algorithm will default to a circle centered at the pupil with a radius twice size. A successful result is shown in Figure 3.

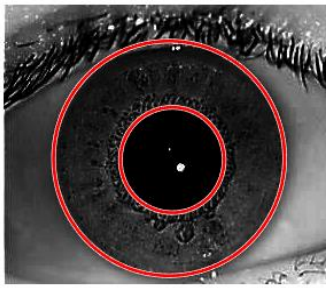

Figure 3 Iris Detection Result for Class 1, Sample 2

C. *Iris Normalization*

In order to compare the segmented iris effectively, it needs to be normalized to account for varying changes in pupil and iris size. Daugman [1] suggested a rubber-sheet model, which projects the coordinates of the iris to a rectangular shape, preserving the locations of the features in iris. This model follows Equation 1, with an example implementation shown in Figure 5.

$$I(x(r,\theta), y(r,\theta)) \rightarrow I(r,\theta) \quad (1)$$

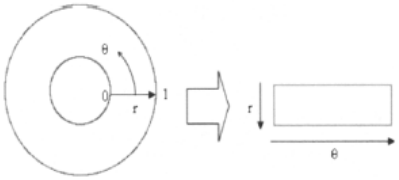

Figure 4 Daugman's Rubber-Sheet Model

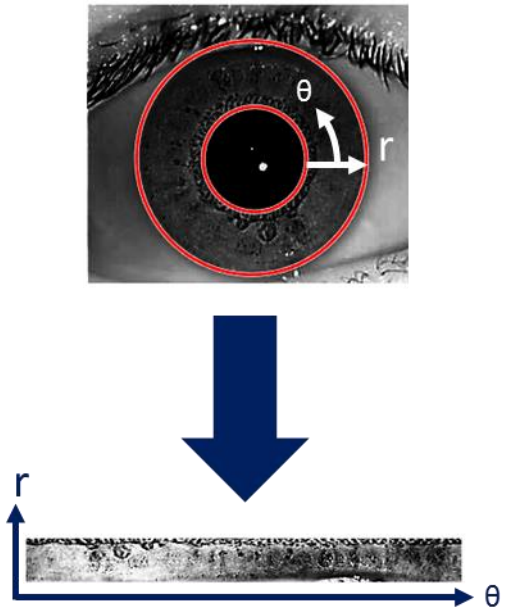

Figure 5 Rubber-Sheet model implementation on Class 1, Sample 2

D. *Occlusion Removal*

The occlusion removal implementation uses two steps: eyelash detection and exclusion by parabolic projection. This is founded on a combination of the methods described by Luo and Lin [10] and Abdullah, et al. [10].

The process used for this eyelash detection follow the steps outlined below.

1. Inverse the normalized iris image, from Section III-C, and convert to binary using a threshold T=0.9
2. Erode image using a circular structural element with radius 1, to remove spots in the iris and other outliers
3. Remove any connected components with a total number of pixels less than 10, to further remove outliers
4. Dilate the image with a 4x4 square structural element

The steps above ensure that the only remaining pixels in the binary image belong to the eyelashes, as shown in Figure 6 and Figure 7.

Figure 6 Eyelash Detection Result for Class 2, Sample 2

Figure 7 Eyelash Detection Result for Class 7, Sample 2

Figure 7 exhibits artifacts that were not removed as part of the eyelash detection in Steps 1-4. These artifacts are taken into account for the parabolic projection outlined in the steps below. A threshold of 500 pixels was chosen as an entry criteria to the parabolic projection. The remaining steps are shown below.

5. Divide the image into left and right sides, as referenced from the vertical center line of the picture.
6. For the left side, estimate the coefficients a, b, c which minimize the least-square error with the parabolic equation:

$$ax^2 + bx + c \tag{2}$$

7. To ensure outliers above the eyelid do not perturb the parabolic estimate, remove with outliers with error less than -30 are removed (-y-direction represent the pixels at the upper part of the image).
8. Repeat Step 6 using only the pixels with the lowest y-value. This helps to ensure the upper contour of the eyelashes is captured in the parabola.
9. Repeat step 8 for the right side. The upper eyelash (right side) tends to have more pixels in the upper part of the image, so these outliers do not need to be removed.
10. Remove any pixels below (greater y-value) the parabolas estimated in Step 7 and 8 of the image, as well as any features detected after Step 4.

A before and after comparison is shown in Figure 8 and Figure 9. This shows that most of the non-iris regions in the image are removed by the algorithm. These pixels are assigned to the mean value of the image, so that they are not identified as a feature in the iris recognition. Finally, the contrast of the normalized iris is improved by using histogram equalization, as suggested by ul Haq et al. [8].

Figure 8 Normalized Iris Image before Occlusion Removal (Class 2, Sample 2)

Figure 9 Normalized Iris Image after Occlusion Removal, identified by yellow pixels (Class 2, Sample 2)

E. *PCA Training*

Once the iris is properly segmented and occlusions removed, the database of pre-processed training images can be projected on a lower dimensional space using Principal Component Analysis (PCA). The pre-processed images are mapped linearly using the PCA such that the in-class variance of the data in each class is maximized [4]. The PCA eigenvectors are computed using the Sirovic and Kirby algorithm [13] as described in Equation 3.

$$S^H S\vec{v_i} = \lambda_i \vec{v_i} \tag{3}$$

where S represents the covariance matrix of the pre-processed images, with the dataset mean subtracted. The *k* largest eigenvectors from Equation 3 are used to form the *PCA Matrix*, which is used to project images to the low-dimensional space. The integer k is chosen as the dimensionality of the space, and is adjusted as a parameter.

F. LDA Training

The final step of the image pre-processing is to compute the Fisher LDA projections for each image. The Fisher LDA minimizes the scatter of the data within a class, while maximizing the scatter between classes [14]. The LDA is computed by finding W which maximizes the following equation:

$$W_{opt} = \underset{W}{\arg max}\left(\det(WRW^H)\right) = \underset{W}{\arg max}\left(\frac{\det(WR_BW^H)}{\det(WR_WW^H)}\right) \quad (4)$$

$$\text{where } R_B = \sum_{i=1}^{c} N_i(\vec{\mu_i} - \vec{\mu})(\vec{\mu_i} - \vec{\mu})^H \quad (5)$$

$$\text{and } R_B = \sum_{i=1}^{c}\sum_{\vec{\Gamma_l}\in Class(i)}(\vec{\Gamma_l} - \vec{\mu_i})(\vec{\Gamma_l} - \vec{\mu_i})^H \quad (6)$$

The problem is solved by finding the J largest eigenvectors in the following equation:

$$R_B\vec{w_i} = \lambda_i R_w\vec{w_i}, i = 1,2,\dots,J \quad (7)$$

To calculate the LDA eigenvectors, this implementation uses the matrices R_B and R_w from Equations 5 and 6 with the images projected on the PCA space as defined below.

$$PCA\ Projection = (Sample\ Image - Mean) * PCA\ Matrix$$

where the *PCA Matrix* is calculated in Section III-E. The LDA eigenvectors can then be used to project the class mean PCA projections on to the LDA space. The resulting LDA projections form a *k* x 224 matrix, where each column represents a class, and each row represents the LDA feature projections for that class. Figure 10 illustrates an example implementation using 2 Classes and 10 samples, where the green line show the LDA Eigenvector which separates the two classes.

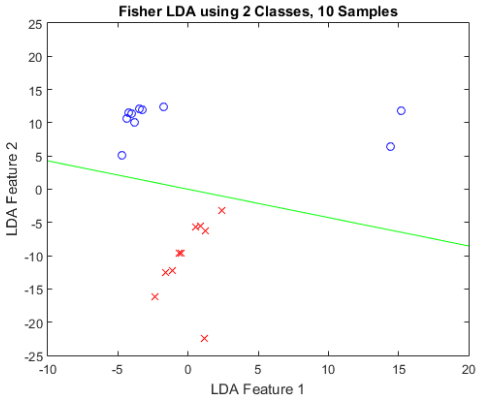

Figure 10 LDA Projection using 2 Classes, 10 Samples

IV. Iris Recognition

To evaluate the iris recognition algorithm, the 10th sample from each class is tested. The algorithm computes the LDA projections for each testing image, and use the minimum mean-squared error to identify the class (person) of the image.

A. Preprocessing

For each testing image, Sections III-A-D outlined in Section III need to be repeated. Because the PCA matrix and the LDA vectors are already computed, Sections III-E-F do not need to be repeated.

B. Iris Scoring

The method in Section III-F is used to project the mean-subtracted image on the LDA space. The resulting LDA projections are used to calculate the mean-squared error, as shown in Equation 8.

$$j_{identified} = \underset{j}{\arg min}\sum_{i=1}^{LDA\ Dim}\left(C_{ij} - \widehat{C_i}\right) \quad (8)$$

where *j* represents the class. The class that minimizes the mean squared error is the identified class.

V. Results and Recommendations

The algorithm described in this article was implemented on a dataset of 224 different subjects, with 10 samples for each [15]. For each subject, the first 9 samples were used to train the PCA and LDA, while the 10th sample was used to test the algorithm.

A. Success Rate

The success rate of the three different occlusion removal options is shown in Figure 11. It is defined by the number of correctly identified iris divided by the total number of irises in that class. The results show that all three options show high success rates around 30-40 samples, but drops off steeply after that and asymptotes around 70%. For these results, the dimension of the Fisher LDA was chosen to be the same size as the number of classes test.

These results are slightly worse than those presented by ul Haw et al. [8]. This discrepancy could be explained by the different use of databases (ul Haq et al. used CASIA 1.0) as well as the use of a threshold discard irises that were not recognized.

The results in Figure 11 also show that the success rate does not vary greatly between the different occlusion removal options. In fact, it appears that the using no occlusion removal at all may be just as good as Option B or C. This is explained by the loss of feature information by Option B and C. In Option B, less information about the iris is available to train the LDA. In Option C, additional artifacts can be introduced if the occlusion removal process miscalculates the position of the eyelashes and eyelids.

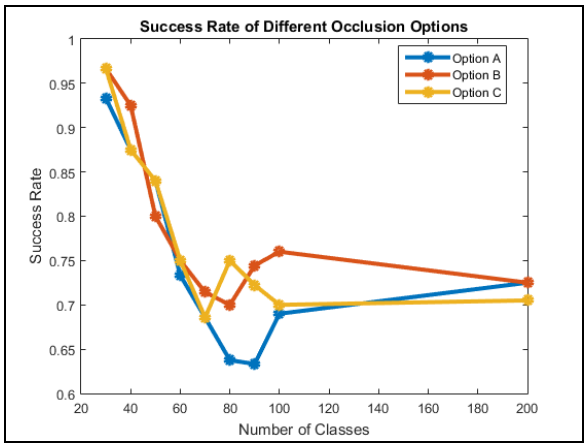

Figure 11 Success Rates of Occlusion Removal Options

B. *Dimensionality*

In order to gauge the effect of the choice of dimensionality on the success rate of the algorithm, dimensions from 2 to 200 were chosen to test irises using a class size of 50. The results shown in Figure 12 illustrate that that success rate increases rapidly between dimensions 2 to 10, but then only gradually increases from 20 to 100. This justifies the choice to use 100 as dimension size in Section V-A, and also suggests that a faster implementation of the algorithm could be achieved at the expense of only modest decreases in success rate. This could be significant for a real-time iris recognition implementation. A dimension size of 200 could have been implemented to repeat the results in Figure 11 to improve the success rate of Option C, but the test time required would prove to be too lengthy and would lose the value of the PCA/LDA approach.

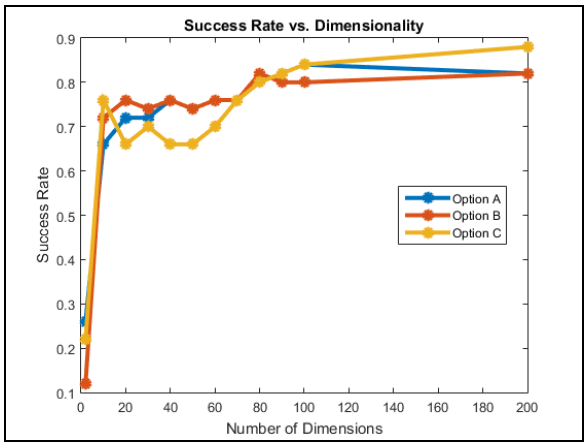

Figure 12 Success Rates vs Dimensionality for Class Size 50

C. *Conclusions & Future Improvements*

The results in this article are significant because they show the impact of occlusion removal has little or no impact on an iris recognition algorithm that uses PCA and LDA. This can provide some benefits to an application, as the occlusion removal steps can be removed, as long as an image of the iris is captured correctly.

Instead, the occlusion techniques described in Section III can be used as a threshold to determine if the image of the iris is valid. This can help improve the fidelity of the samples and effectively improve the success rate of any iris recognition algorithm.

In a future implementation, this image processing algorithm could be improved by further refinement of the morphological processes used. This implementation required adjusting pixel thresholds and safeguards against false matches, which can be further optimized. Additionally, higher dimensions could be used to evaluate the success of Option C. To do this, high performance computing assets are recommended.

ACKNOWLEDGMENT

The author would like to thank Professor Gordon Wetzstein and Kushagr Gupta for teaching this course and providing insightful help and mentorship when asked.

The database of images used were accessed from IIT Delhi Iris Database (Version 1.0) with the generous permission of Ajay Kumar.

REFERENCES

[1] Chowhan, S.S.; Shinde, G.N., "Iris Biometrics Recognition Application in Security Management," in *Image and Signal Processing, 2008. CISP '08. Congress on* , vol.1, no., pp.661-665, 27-30 May 2008.

[2] H. Proenca and L. A. Alexandre, "A method for the identification of inaccuracies in pupil segmentation," in *Proc. ARES 2006*, 2006.

[3] J.Daugman, "High confidence visual recognition of person by a test of statistical independence" IEEE transaction, Pattern Analysis and Machine intelligence, vol. 15, pp. 1148-1161, Nov 1993.

[4] Jin-Xin Shi; Xiao-Feng Gu, "The comparison of iris recognition using principal component analysis, independent component analysis and Gabor wavelets," in *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on* , vol.1, no., pp.61-64, 9-11 July 2010.

[5] Daugman, J., "Probing the Uniqueness and Randomness of IrisCodes: Results From 200 Billion Iris Pair Comparisons," in *Proceedings of the IEEE* , vol.94, no.11, pp.1927-1935, Nov. 2006.

[6] Boles W. and Boashash B. A human identification technique using images of the iris and wavelet transform. IEEE Trans. on Signal Processing, 46(4):1185 – 1188, 1998.

[7] Wildes R.P, "Iris Recognition: An Emerging Biometric Technology," Proc. IEEE, vol. 85,no. 9, pp. 1348-1363, September 1997.

[8] Emad ul Haq, Q.; Javed, M.Y.; Sami ul Haq, Q., "Efficient and robust approach of iris recognition through Fisher Linear Discriminant Analysis method and Principal Component Analysis method," in *Multitopic Conference, 2008. INMIC 2008. IEEE International* , vol., no., pp.218-225, 23-24 Dec. 2008

[9] M. Pardas, "Extraction and tracking of the eyelids," Proc. IEEE Symp. International conference on Acoustics ,Speech and Signal Processing(ICASSP '00), IEEE Press ,June. 2000, vol. 4, pp. 2357-2360.

[10] Zhongliang Luo; Tusheng Lin, "Detection of Non-iris Region in the Iris Recognition," in *Computer Science and Computational Technology, 2008. ISCSCT '08. International Symposium on* , vol.2, no., pp.45-48, 20-22 Dec. 2008.

[11] Abdullah, M.A.M.; Dlay, S.S.; Woo, W.L., "Fast and accurate method for complete iris segmentation with active contour and morphology," in *Imaging Systems and Techniques (IST), 2014 IEEE International Conference on* , vol., no., pp.123-128, 14-17 Oct. 2014Peter N. Belhumeur Joao P. Hespanha David J. Kriegman, "Eigenfaces vs.

Fisherfaces: Recognition Using Class Specific Linear Projection" presented in IEEE Trans. on PAMI, July 1997.

[12] C. Kimme, D. Ballard, J. Sklansky, Finding circles by an array ofaccumulators, Proc. ACM 18 (1975) 120–122.

[13] Kirby, M.; Sirovich, L., "Application of the Karhunen-Loeve procedure for the characterization of human faces," in *Pattern Analysis and Machine Intelligence, IEEE Transactions on* , vol.12, no.1, pp.103-108, Jan 1990

[14] R.A. Fisher, “The Use of Multiple Measures in Taxonomic Problems,” Ann. Eugenics, vol. 7, pp. 179-188, 1936

[15] Kumar, Ajay. "IIT Delhi Iris Database (Version 1.0)." *IIT Delhi Iris Database*. Hong Kong Polytechnic University, n.d. Web. 18 Nov. 2015. <http://www4.comp.polyu.edu.hk/~csajaykr/IITD/Database_Iris.htm>.