

On Improving the Efficiency of Eigenface Using a Novel Facial Feature Localization

Aleksey Izmailov and Adam Krzyżak

Department of Computer Science and Software Engineering,
Concordia University,
Montreal, Canada H3G 1M8
izmailoff@gmail.com, krzyzak@cs.concordia.ca

Abstract. Face recognition is the most popular non-intrusive biometric technique with numerous applications in commerce, security and surveillance. Despite its good potential, most of the face recognition methods in the literature are not practical due to the lack of robustness, slow recognition, and semi-manual localizations. In this paper, we improve the robustness of eigenface-based systems with respect to variations in illumination level, pose and background. We propose a new method for face cropping and alignment which is fully automated and we integrate this method in Eigenface algorithm for face recognition. We also investigate the effect of various preprocessing techniques and several distance metrics on the overall system performance. The evaluation of this method under single-sample and multi-sample recognition is presented. The results of our comprehensive experiments on two databases, FERET and JRFD, show a significant gain compared to basic Eigenface method and considerable improvement with respect to recognition accuracy when compared with previously reported results in the literature.

1 Introduction

For many automated face recognition systems, the first step in the recognition process is to spatially locate the face in the image. This process is called *face detection*. The main challenge in face detection is the high degree of variation in facial appearance. The two main face detection methods are *feature-invariant* approach and *classification-based* approach. The former methods use those features of the face which can resist through appearance variations (such as eyes, mouth, face texture or color) as guidance to locate the face in the image. The latter methods consider a face detection problem as a binary classification, and train a classifier to filter out the non-face areas of the image. Image-based approaches were shown to be more effective in dealing with complex environments, varying illumination conditions, and cluttered backgrounds. The high computational cost is usually a major drawback of many face detection algorithms, especially for real-time applications. Viola and Jones [17], proposed a system using an ensemble learning approach which has received much attention due to its quick response time. Their

method also has some features as multiple detection, and robustness to size variation and orientation (see [20] for a survey of face detection methods). We have used this method for the detection part in our system.

There have been a variety of face recognition methods proposed in the literature (see comprehensive surveys in [15,7,1]). These methods are categorized into three groups: holistic matching, feature-based matching, and hybrid methods [5]. Holistic methods use the full face area in recognition. Feature-based matching methods usually apply a structural classifier on extracted features such as mouth, eyes, nose, ears. Finally, hybrid methods combine holistic and feature-based methods in an attempt to employ the best properties of both. One of the best known face recognition methods is the *Eigenface algorithm* [16,6,14], which is based on Principal Component Analysis (PCA). The Eigenface technique has shown to be effective on images which are properly aligned, i.e. which are aligned so that fiducial points for all images are located at the same coordinates. It is often used for frontal face recognition since it is easy to align and there is no need to deal with head rotation. Like other holistic techniques, the Eigenface approach is sensitive to illumination and light intensity. This method operates on values of grey-scale pixels which are encoded as decomposition coefficients by comparing distances between vectors. Therefore, images produced with similar light intensities will tend to have lesser distances between them and because of that property might be misclassified. The Eigenface technique by itself does not work well in realistic environments with many dynamic factors, because captured images usually contain complicated background and occlusions.

In this paper, we present a face recognition system that is able to detect human faces, crop them, and perform recognition in real time. Eigenface recognition is integrated with our system. Variable illumination, pose, and background are hard problems in recognition with Eigenface approach. We are able to reduce the impact of these factors on recognition performance by our novel face cropping algorithm as well as carefully selected preprocessing techniques and parameters. Our system shows promising results and may be used in many practical areas such as security, entertainment and business.

2 Elliptical Facial Feature Localization (EFFL) Method

Our approach to localization is based on elliptical shape which represents a person's face. To be able to do this, we first find the location of eyes using Haar-like features in conjunction with Adaptive Boosting (AdaBoost) for detection of a pair of eyes [17,8]. Based on the location of eyes we calculate a point which we shall call face center point (FCP). The FCP represents center of a human face if we assume that it has elliptical shape. This point is later used to build an ellipse surrounding the face area. The area outside of the ellipse is considered as background and does not contain information useful for recognition. We eliminate background area by setting it's pixels to one constant value such as zero, thus producing an image containing only face and black region surrounding it. We applied cascades of boosted classifiers to detect face, one eye, two

eyes, nose and lips in order to extract face area. During our experiments the most reliable results have been achieved by one classifier which was trained to detect frontal eyes. Thus we focused on face cropping based on aforementioned classifier.

The face center point is calculated by considering a rectangle $R = ABCD$ such that both eyes fall inside it (Figure 1). We denote the left and right upper corners of R as $A(x_A, y_A)$ and $B(x_B, y_B)$, respectively. These two points are the two vertices of a triangle that we form in order to find the center of the face. The third vertex of the equilateral triangle ABC , which we call the facial triangle, is point C . Point C is located in the mouth area. Figure 1 shows these stages involved in face localization and cropping. We define the centroid of ABC to be the center point of the face (O_f). We next use O_f to eliminate the background from the face image. By background we mean any part of the face image which is not a facial feature or skin area. For the purpose of background removal, we form an ellipse E around the face, which circumscribed area contains all facial features of interest such as lips, nose and eyes. Let face ellipse center O_E be the intersection point of its two major and minor axes. We choose the ellipse center O_E to be the same point as the face center O_f . The length and width of this rectangle are equal to the major and minor axes of the ellipse E as shown in Figure 1. The major and minor axes are calculated as follows:

$$\begin{aligned} \|a_1 a_2\| &= k_1 * \|AB\| \\ \|b_1 b_2\| &= k_2 * \|AB\| \end{aligned} \quad (1)$$

where k_1 and k_2 are fixed constants. These constants have been chosen carefully to build an ellipse so that we can eliminate as much background as possible, while preserving the facial features. k_1 and k_2 have been measured for an average typical person to provide a good ROI for the face recognition. We fill pixels that lie in the area outside of E and inside of ROI_C , with some constant C ($0 \leq C \leq 255$) which represents the gray scale color. The final result of this procedure can be seen in Figure 1.

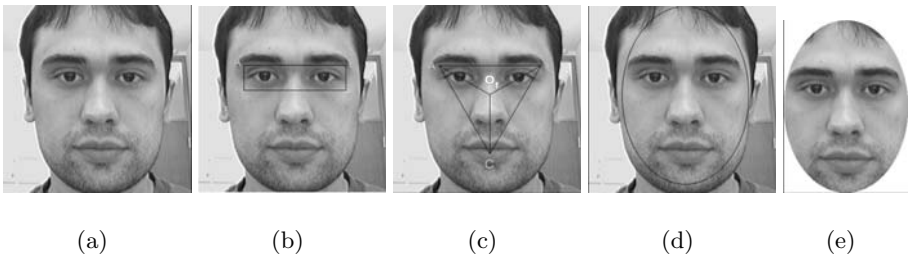


Fig. 1. The stages of face cropping procedure (a) original face image, (b) locating eyes (c) estimating face center O_f , (d) producing the face ellipse (e) cropping the face according to the face ellipse

3 Eigenface Recognition System with EFFL

In our face recognition process we use the Eigenface approach [16]. In order to recognize an unseen image, we proceed with automatic cropping of face according to the details described in Section 2. The same cropping and preprocessing should be applied to the testing and training images before recognition. Test images are represented in terms of eigenface coefficients by projecting them into face space (eigenspace) calculated during training. Test sample is recognized by measuring the similarity distance between the test sample and all samples in the training. The similarity measure is a metric of distance calculated between two vectors. We utilize some of the standard techniques used in image processing such as histogram equalization, and symmetrization. Histogram equalization redistributes the intensity values of pixels so that the distribution of intensities becomes uniform [4].

Symmetrization is a technique for recovering one part of an object from another symmetrical part. It can be applied to human faces to restore any shaded or occluded part. The axis of symmetry lies vertically along the nose line. Some promising results have been shown on finding this axis reliably, however there is still room for improvement [2]. Our approach to this problem, is to first find a bounding box around eyes as described in Section 2. We then build the center line as the line crosses the center of this rectangle and be parallel to short sides. Next we compute brightness of each side of the face as a function of sum of grayscale values of pixels. We compare relative brightness of the left and right side to some threshold ratio to determine if symmetrization is necessary. If the value exceeds the threshold we copy the brightest side to the darkest side, thus recovering it and making it more visible. Symmetrization works well only if the center line is determined very precisely. There are many factors that complicate the problem of finding a center line such as bad illumination, pose variation and facial expressions.

After the ellipse is extracted, we resize the resulting image to a predefined standard. This normalization step is necessary for Eigenface algorithm, as it requires that all vectors to be of the same dimensionality. We use bilinear interpolation to resize images, however other options are possible such as: nearest-neighbor interpolation, resampling using pixel area relation, and bi-cubic interpolation. Since detected features (such as eyes) for samples of the same class are practically located at the same coordinates of an image (fiducial points), we are able to automatically align images using these points and resizing. Alignment of the faces used for recognition is very important for Eigenface approach and improves recognition rate. As part of normalization we subtract average face image from all images in the training and testing set to normalize for lighting conditions [16].

Traditional Eigenface approach utilizes Euclidean distance [16]. Various experiments have been conducted by researches to measure performance of other distance measures [10]. Although in many cases Mahalanobis distance has outperformed Euclidian distance, not all studies agree on that matter. We investigate a variety of distance measures with Eigenface approach: squared Euclidean

distance (in the forms of Sum Square Error and Mean Square Error), weighted SSE, Manhattan distance, weighted Manhattan distance, Mahalanobis distance, and Simplified Mahalanobis Distance (SMD). We report our results using these metrics in Section 4.

We perform a number of experiments in order to achieve best performance that method can provide on different datasets. We investigate what area of face is important for recognition, what number of eigenvectors are better to use, how normalization improves recognition and how to optimize between recognition performance and real-time performance by selecting proper image size.

4 Empirical Evaluation

We measure performance of our biometric system using algorithm-independent approach based on: False Acceptance Rate (FAR), False Rejection Rate (FRR), and Equal Error Rate (EER) [18]. We shall define in detail each measure according to specifics of our application. Note that each measure is calculated as personal rate, and the average result is reported as the final rate.

$$\begin{aligned}
 FAR &= \frac{\# \text{mis-recognized}}{\text{total recognition attempts}}, & FRR_1 &= \frac{\# \text{frames without face}}{\text{total \#frames}} \\
 FRR_2 &= \frac{\# \text{correctly recognized above threshold}}{\text{total \#detected faces}} \\
 FRR_3 &= \frac{\# \text{frames without face} + \# \text{frames above threshold}}{\text{total \#frames}}
 \end{aligned} \tag{2}$$

We define FAR as a ratio of mis-recognized samples which contain human face to total number of attempts to recognize a person. We define three FRR measures (FRR_1, FRR_2, FRR_3). FRR_1 is designed to show rejections only due to poor image quality. This includes absence of face in the image or inability of the program to locate face of desired size and location. FRR_1 is a ratio of number of frames where face is not found to total number of frames captured. FRR_2 serves to show the rate of false rejections due to thresholding after recognition to the total number of frames, where the face is found. Finally FRR_3 shows all rejections due to both reasons depicted in FRR_1 and FRR_2 . It is defined as ratio of a number of false rejections due to thresholding plus number of frames where face is not found to total number of frames.

FRR_1 is convenient to use to see how sensitive the system is to user behavior. FRR_1 is more of a comfort criteria and can be reasonably low with collaborative user behavior. On contrary FRR_2 is a procedure that is free of quality rejection influence and thus serves as a true measure of system performance. It allows us to see how likely it is for the system to produce errors with “clean” data. FRR_3 is a good measure of overall system performance which takes into account user comfort as well. FRR and FAR by themselves do not provide a good comparison basis between systems. One usually determines a probability distribution function representing degree of similarity between test and reference samples. The most common approach is to calculate histograms which show frequency of incidence for every similarity rating.

Most similarity measures such as Euclidean distance or Mahalanobis distance return results which show how far one instance is from another one. Thus by setting threshold at a minimum distance value provided from false acceptance tests we can be "statistically" sure that there will be no false acceptance cases, i.e. if $FAR = 0$ then FRR most likely will be quite high. If we set threshold to obtain zero FRR then FAR might be unacceptably high. We also used Equal Error Rate EER , as a threshold representing the error at a point i where $FAR(i) = FRR(i)$.

4.1 Data

In our experiments we used two databases: FERET [3] and the extended version of JRFD [12]. JRFD consists of approximately 100 people. This database contains 16 frontal face images per individual (men and women of different age and race) taken by cellular phone camera. Images are taken in natural environment and faces are cropped to rectangular areas of 350x350 pixels. Images have prevalence of ambient light with left or right areas of the face occasionally being less illuminated.

4.2 Results

In this section we present final results on combination of methods tested in order to achieve highest recognition rate. We experiment with single and multi instance algorithms. A set of preprocessing techniques were applied to both databases and different criteria were examined as presented below.

Eigenvectors: Number of eigenvectors used for purposes of decomposition as well as similarity measure comparison is chosen. We select 150 top eigenvectors since this number produced the best result for both databases.

ROI: Cropping the whole face rather than eyes showed better performance. We believe that area containing both frontal eyes might be used for recognition only in very well controlled environment because the shape of the eyes changes due to different emotions, lighting, or blinking. These peculiarities produce high variance in pictures of eyes and make them less suitable for recognition.

Size: We select optimal size of the face. Since results of the experiments didn't vary much with image size we decided to choose the minimum size of the cropped face from JRFD (200x280) in order to gain recognition speed. It is also twice larger than that of FERET but smaller than it's maximum size.

Background: Our experiments show that the best value to fill out pixels outside the face area with, is zero. We present results for both all and averaged coefficients of a training class. For purposes of real-time system the use of averaged coefficients reduces time to compare test image coefficients to all training instances when finding smallest distance. However on modern computers with relatively small face databases the difference is hardly noticeable.

Brightness: We observed an improvement in accuracy by applying histogram equalization procedure to the cropped face. Another option besides regular intensity normalization techniques would be removing first eigenvector which provides less improvement in recognition rate. Hence we choose histogram equalization approach as a preprocessing step after face cropping.

Similarity measure: Euclidean distance, Mahalanobis distance and Manhattan distance showed good performance in different cases. We performed experiments using all three of these distances and compared results.

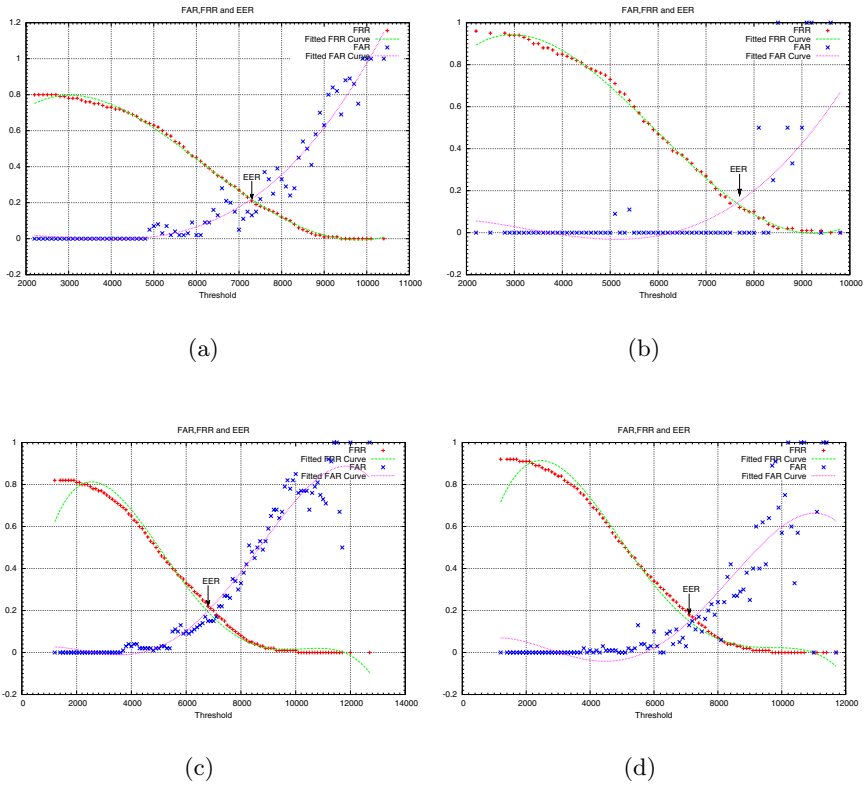


Fig. 2. Estimation of Equal Error Rate (EER) and tradeoffs of false acceptance and false rejection (FAR, FRR). Real data points as well as fitted curve are represented. (a) real person, single instance; (b) real person, multi instance; (c) database image, single instance; (d) database image, multi instance.

Multi instance: Recognition based on results combined from multiple images by selecting minimum weighted distance (described in Multi Instance Recognition section) showed best results. We present results for both single and multi instance recognition algorithms. Table 1 provides brief summary and statistics on our experiments.

Table 1. Experimental results for all performance rates defined in our experiments. EER estimated without curve fitting.

rate	test object	frames	threshold	EER	AVG	STDEV	MIN	MAX
<i>FAR</i>	person	multi	-	-	0.0529	0.1354	0.0000	0.4091
<i>FAR</i>	person	single	-	-	0.1861	0.1647	0.0234	0.5461
<i>FAR</i>	picture	multi	-	-	0.0923	0.1277	0.0000	0.7955
<i>FAR</i>	picture	single	-	-	0.1792	0.1466	0.0000	0.7744
<i>FRR</i> ₁	person	multi	-	-	0.0000	0.0000	0.0000	0.0000
<i>FRR</i> ₁	person	single	-	-	0.3278	0.1488	0.0298	0.5464
<i>FRR</i> ₁	picture	multi	-	-	0.0000	0.0000	0.0000	0.0000
<i>FRR</i> ₁	picture	single	-	-	0.1236	0.1565	0.0000	0.7583
<i>FRR</i> ₂	person	multi	8019	0.0954	0.1235	0.2287	0.0000	0.6818
<i>FRR</i> ₂	person	single	7445	0.1845	0.1697	0.2263	0.0138	0.7190
<i>FRR</i> ₂	picture	multi	7363	0.1491	0.0664	0.0922	0.0000	0.4286
<i>FRR</i> ₂	picture	single	7128	0.1804	0.1845	0.1785	0.0000	0.7143
<i>FRR</i> ₃	person	multi	8019	0.0954	0.1235	0.2287	0.0000	0.6818
<i>FRR</i> ₃	person	single	7445	0.1845	0.4523	0.1627	0.2914	0.8046
<i>FRR</i> ₃	picture	multi	7363	0.1491	0.1533	0.1804	0.0000	0.8333
<i>FRR</i> ₃	picture	single	7128	0.1804	0.2823	0.2025	0.0596	0.9205
<i>FAR</i> thresh	person	multi	8019	0.0954	0.0074	0.0222	0.0000	0.0667
<i>FAR</i> thresh	person	single	7445	0.1845	0.0524	0.0588	0.0000	0.1675
<i>FAR</i> thresh	picture	multi	7363	0.1491	0.0325	0.0951	0.0000	0.6591
<i>FAR</i> thresh	picture	single	7128	0.1804	0.0404	0.0941	0.0000	0.5219

We can conclude that our system has practical use and error proof even in realistic environment with complicated background and pose variations. Although experimental data shows that we can achieve $FAR = 0$ at tolerable FRR level we suggest long-term tests and further investigation to find possible problems in recognition.

Table 2. Comparison of recognition performance between different methods on FERET database

method	Accuracy%
Eigenface-(gray) [19]	35.00
Eigenface-(Gabor) [19]	36.05
TCF-CFA(N=3) [19]	94.18
Eigenface	48.89
Our method-single instance	84.18
Our method-multi-weighted	86.30

Table 3. Comparison of recognition performance between different methods on JRFD database

method	Accuracy%
multi-biometric [12](20 & 9 face & hand feature points)	99.54
Face [13](12 feature points)	98.85
Eigenface	89.46
Our method-single instance	98.34
Our method-multi-weighted	100.00

Table 2 shows results obtained on FERET database by other methods. First two eigenface methods are standard eigenface approaches. The sets of images used for these experiments are very similar but not identical. In all cases frontal face images were used. Although TCF-CFA method shows good performance it is based on precise locations of fiducial points which are established manually. Furthermore as reported its computational complexity is high. We would like to point out improvements made by our approach over the standard algorithm by

Table 4. JRFD - final recognition results with cropped face 200x280, 0 pixels - background, histogram equalized, with averaging coefficients, 150 eigenvectors

Distance Measure	Coef.Avg.	Accuracy(%)	Accuracy(%)	Accuracy(%)	Accuracy(%)
		train-test:249-1025	train-test:332-949	train-test:2056-949	train-test:949-2056
		JRFD	JRFD	FERET	FERET
Euclidean	no	97.66	98.20	82.51	68.24
Mahalanobis		97.46	97.77	82.51	67.70
Manhattan		98.34	97.66	84.18	70.18
Multi-weighted		100.00	100.00	84.18	86.30
Euclidean	yes	97.56	97.66	78.61	68.24
Mahalanobis		97.37	97.45	79.35	67.70
Manhattan		97.76	97.77	83.03	70.18
Multi-weighted		100.00	100.00	83.03	86.30

comparing results in the last three rows of Table 2. Original method performed poorly because no alignment or size normalization of the facial area is done. We would also like to note that 1-NN method is not restricted by thresholding in these experiments and accuracy can be further improved at the cost of FRR rate. We compare results on JRFD in Table 3 with the most statistically significant results reported in [11]. As you can see our fully automated system is only 0.51% less accurate than the manual face recognition system in [11]. Also it is only 1.2% worse than multi-biometric system. Our system outperforms [11] in case of multi-instance recognition. Overall we observe a drastic improvement in eigenface approach itself by almost 36% on FERET and 9% on JRFD databases. Moreover our results compare well to the state-of-the-art methods while those methods are not fully automatic.

In course of experiments using a set of preprocessing techniques on both databases we observed similar behavior of different techniques. Some researchers report in the literature significant improvements after applying symmetrization [9]. We did not obtain very precise results of symmetrization. This is mainly because minor shifts in center line would cause face to look less similar to original by shrinking or stretching lips and nose areas and changing the distance between eyes.

5 Conclusions

In this paper we developed a real-time face recognition response with a new automated face detection, which we called EFFL. It should be noted that very few existing applications automatically locate a face in the image and do automatic alignment. For many developed methods, faces should be manually located. This makes such systems impractical in real world applications. In our approach for face detection, if there exist more than one face in the image, our system selects the face that appears in the largest size or closest to the camera, for recognition or authentication. The face is then cropped and aligned by using our introduced background elimination method. An ellipse is fitted to the face area in order to minimize the effect of different hair styles while it preserves all fiducial features.

The face recognition module in our system uses Eigenface methodology. The Eigenface method has the shortcoming of reflecting variations of the background as well as variations of the face which can increase false matching. Our proposed face cropping procedure offers an efficient solution to this problem. We showed that performing face cropping prior to the recognition significantly improved the recognition accuracy (about 7% on JRFD database). In order to achieve a better performance, we also tried different distance metrics instead of the traditionally used Euclidian distance. Our results on two different face databases (FERET and JRFD) suggest that Manhattan distance is slightly superior. We also showed that performing histogram equalization on the cropped faces improves the overall system performance. We plan to continue our investigations with new symmetrization methods and eigenvectors. We also intend to expand our experiments to other databases and applications.

References

1. Abate, A.F., Nappi, M., Riccio, D., Sabatino, G.: 2d and 3d face recognition: A survey. *Pattern Recog. Lett.* 28(14), 1885–1906 (2007)
2. Chen, X., Flynn, P.J., Bowyer, K.W.: Fully automated facial symmetry axis detection in frontal color images. In: *AUTOID 2005: Proceedings of the Fourth IEEE Workshop on Automatic Identification Advanced Technologies*, Washington, DC, USA, pp. 106–111. IEEE Computer Society, Los Alamitos (2005)
3. FERET (October 2008), <http://www.frvt.org/feret/default.htm>
4. Gonzalez, R.C., Woods, R.E.: *Digital Image Processing*, 3rd edn. Prentice-Hall, Inc., Upper Saddle River (2006)
5. Hongtao, S., Feng, D.D., Rong-chun, Z.: Face recognition using multi-feature and radial basis function network. In: *VIP 2002: Selected papers from the 2002 Pan-Sydney workshop on Visualisation*, pp. 51–57. Australian Computer Society, Inc., Darlinghurst (2002)
6. Kirby, M., Sirovich, L.: Application of the Karhunen-Loeve procedure for the characterization of human faces. *IEEE Trans. Pattern Anal. Mach. Intell.* 12(1), 103–108 (1990)
7. Kong, S.G., Heo, J., Abidi, B.R., Paik, J., Abidi, M.A.: Recent advances in visual and infrared face recognition: a review. *Comput. Vis. Image Underst.* 97(1), 103–135 (2005)
8. Lienhart, R., Kuranov, A., Pisarevsky, V.: Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In: Michaelis, B., Krell, G. (eds.) *DAGM 2003*. LNCS, vol. 2781, pp. 297–304. Springer, Heidelberg (2003)
9. Nakao, N., Ohyama, W., Wakabayashi, T., Kimura, F.: Automatic detection of facial midline as a guide for facial feature extraction. In: *PRIS*, pp. 119–128 (2007)
10. Perlibakas, V.: Distance measures for pca-based face recognition. *Pattern Recog. Lett.* 25(6), 711–724 (2004)
11. Rokita, J.: Multimodal biometric system based on face and hand images taken by a cell phone. Master's thesis, Computer Science Dept., Concordia University, Montreal, Quebec, Canada (March 2008)
12. Rokita, J., Krzyżak, A., Suen, C.Y.: Cell phones personal authentication systems using multimodal biometrics. In: *International Conference on Image Analysis and Recognition*, pp. 1013–1022 (2008)

13. Rokita, J., Krzyżak, A., Suen, C.Y.: Multimodal biometrics by face and hand images taken by a cell phone camera. *International Journal of Pattern Recognition and Artificial Intelligence*, 411–429 (2008)
14. Sirovich, L., Kirby, M.: Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America* 4(3), 519–524 (1987)
15. Tan, X., Chen, S., Zhou, Z.-H., Zhang, F.: Face recognition from a single image per person: A survey. *Pattern Recog.* 39(9), 1725–1745 (2006)
16. Turk, M., Pentland, A.: Eigenfaces for recognition. *Journal of Cognitive Neuroscience* 3(1), 71–86 (1991)
17. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: *CVPR*, vol. 1, pp. 511–518 (2001)
18. Woodward, J.D., Orland, N.M.: *Biometrics*. McGraw-Hill, Inc., New York (2002)
19. Yan, Y., Zhang, Y.-J.: Tensor correlation filter based class-dependence feature analysis for face recognition. *Neurocomputing* 71(16-18), 3434–3438 (2008)
20. Yang, M.-H., Kriegman, D.J., Ahuja, N.: Detecting faces in images: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* 24(1), 34–58 (2002)