Face Recognition Based on Multiple Facial Features

Rui Liao, Stan Z. Li
Intelligent Machine Laboratory
School of Electrical and Electronic Engineering
Nanyang Technological University, Singapore 639798

Abstract

An automatic face recognition system based on multiple facial features is described in this paper. Each facial feature is represented by a Gabor-based complex vector and is localized by an automatic facial feature detection scheme. Two face recognition approaches, named Two-Layer Nearest Neighbor (TLNN) and Modular Nearest Feature Line (MNFL) respectively, are proposed. Both TLNN and MNFL are based on the multiple facial features detected for each image and their superiority in face recognition is demonstrated.

1 Introduction

Classifying a face image as a picture of a given individual is probably the most difficult recognition task that humans carry out on a routine basis with nearly perfect rate. In recent years, many different techniques have been applied to this task, and there is a considerable literature on face recognition. It has been widely recognized that basic facial features, such as eyes, nose and mouth, and their spatial arrangement, are important for discrimination among face images that are all quantitatively similar to each other. In this paper, an automatic face recognition system based on multiple facial features is presented.

Various strategies for facial feature detection have ever been proposed, ranging from the early techniques using generalized symmetry operators [1] and multi-layer perceptrons [2], to more recent eigenfeature approach [3]. In our system, a different facial feature detection scheme, which is based on the framework of Elastic Graph Matching in [4], is utilized. In [4], most effort was put on the extraction of a face graph with as many as 48 nodes to code a face. In our scheme, however, a more accurate facial feature localization approach is adopted. Only 17 facial features, all of which have clear meanings and exact correct positions, are localized for each new face image. It provides the basis for highly accurate facial feature detection.

After facial feature detection, the complex vector at each facial feature, which is obtained by a set of convolutions with a family of complex Gabor filters, is used as the feature vector for face recognition. Two novel non-parametric classification methods, named *Two-Layer Nearest Neighbor (TLNN)* and *Modular Nearest Feature Line (MNFL)*, are proposed. In *TLNN*, only the first few facial features, which obtain the highest similarities with the corresponding facial features of model face images, are utilized for face recognition. *TLNN* is robust to changes in parts of face images by not taking those changed parts into consideration when defining image similarity, on which face recognition is based. *Nearest Feature Line (NFL)* approach was first proposed in [5]. By constructing a subspace from multiple prototypes available for each class, *NFL* generalizes the multiple prototypes each class and expands the representational capacity of available prototypes. In our system, *NFL* technique is extended to *MNFL* by constructing one subspace for each facial feature. Dramatically more kinds of variations between images could be approximated by *MNFL* than the uniform *NFL* in [5], and recognition performance is further improved.

In practical applications, hairstyle can be changed easily for a certain person and background of an image is hard to be expected and controlled. However, to date tests of most face recognition methods utilize hair information for face recognition and background of images, more or less, has impact on recognition performance. In our system, by choosing appropriate facial features, hair information is not utilized for face recognition. Influence of background on recognition performance is also minimized. Our experiment, therefore, is a test of face recognition performance based on faces only.

2 Facial Feature Representation and Detection

2.1 Facial Feature Representation

As a node in [4], in our system, a facial feature is represented by a Gabor-based complex vector, which is obtained by a set of convolutions with a family of complex Gabor filters:

$$\psi_j(\vec{X}) = \frac{\left|\vec{k}_j\right|^2}{2\pi\omega^2} \exp\left(-\frac{\left|\vec{k}_j\right|^2\left|\vec{X}\right|^2}{2\omega^2}\right) * \left[\exp(i\vec{k}_j \cdot \vec{X}) - \exp(-\frac{\omega^2}{2})\right] \quad (1)$$

where $\omega = 2\pi$, $\vec{k}_j = k_v \cos\phi_u + jk_v \sin\phi_u$, $k_v = 2^{-\frac{v+2}{2}}\pi$, $\phi_u = u\frac{\pi}{8}$, with index $j = u + 8v$. Five sizes, indexed $v \in \{0 \ldots 4\}$ and eight orientations, indexed $u \in \{0 \ldots 7\}$, are used. This representation is chosen for its biological relevance and technical properties. The Gabor kernels resemble the receptive field profiles of simple cells in the visual pathway. They are localized in both space and frequency domains and achieve the lower bound of the space-bandwidth product as specified by the uncertainty principle [6].
At each location $\vec{X}$, the coefficients are:

$$J_j(\vec{X}) = \int I(\vec{X}')\psi_j(\vec{X}-\vec{X}')\, d^2\vec{X}' \qquad (j = 0, 1, 2 \ldots 39) \quad (2)$$

where $I(\vec{X})$ is the image grey level distribution. The full wavelet transform provides 40 complex coefficients at each facial feature, forming a 40-dimensional complex vector $J(\vec{X})$ representing the corresponding facial feature.

2.2 Facial Feature Similarity

The complex coefficient J_j has smoothly changing magnitude $a_j(\vec{X})$ and a phase $\phi_j(\vec{X})$ spatially varying with approximately the characteristic frequency of the respective Gabor filter. In our system, two different facial feature similarity functions are utilized for different tasks.
One is phase-insensitive similarity function, which is utilized mainly for facial feature discrimination:

$$S_m(J, J') = \frac{\sum_j a_j {a'}_j}{\sqrt{\sum_j a_j^2 \sum_j {a'}_j^2}} \quad (3)$$

the other is phase-sensitive similarity function, which plays the key role in facial feature localization:

$$S_p(J, J') = \frac{\sum_j a_j {a'}_j \cos(\phi_j - \phi'_j - \vec{d}\vec{k}_j)}{\sqrt{\sum_j a_j^2 \sum_j {a'}_j^2}} \quad (4)$$

where $\vec{k}_j$ is the characteristic wave vector of the respective Gabor filter and $\vec{d}$ is an estimated displacement that compensates for the rapid phase shifts. In [4], $\vec{d}$ was estimated by maximizing S_p in its Taylor expansion and only displacements within 8 pixels could be estimated. In our scheme, however, much more effort is put on the accuracy of facial feature detection. By maximizing the original S_p, rather than its Taylor expansion, the estimation range of $\vec{d}$ is increased to 18 pixels (shown in Figure 1). $\vec{d}$ is therefore relied upon as a main tool for facial feature detection in our scheme, rather than only for fixed times of small-range position adjustments of face graph nodes in [4].

2.3 Face Bunch Graph

In order to localize facial features automatically for a new face image, a data structure, called *Face Bunch Graph (FBG)*, is utilized. A collection of face images (80 images in our system) with different local properties are marked at correct facial feature points (17 facial features in our system) manually. Then a bunch of complex vectors, each derived from a different face image at the same facial feature, is stored stack-like for the corresponding facial feature. An eye bunch, for instance, may include complex vectors obtained from closed, open, female, and male eyes. By doing so, a wide range of possible variations in the appearance at that facial feature is covered. Thus, for a *FBG*, its nodes are labeled with bunches of complex vectors J_n^{FBG} and its edges are labeled with the averaged distance vectors $\Delta\vec{X}_e^{FBG}$ between facial features.

$$\Delta\vec{X}_e^{FBG} = \sum_{i=1}^{N} \Delta\vec{X}_e^{FBGi}/N \quad (5)$$

where N is the number of images utilized to form the *FBG* and $\Delta\vec{X}_e^{FBGi}$ is the distance vector between the corresponding facial features of the i^{th} image in the *FBG*. During the location of facial features in a new face image, *FBG* is matched to the new image. The complex vector extracted from a facial feature in the new image will be compared to all complex vectors in the corresponding bunch of the *FBG*, and the best fitting one will be selected.

2.4 Facial Feature Detection

For a new face image, 17 basic facial features such as pupils, nose tip, corners of mouth, ends of eyebrows and etc. are localized automatically in our system. The whole facial feature detection process consists of three stages—global face search, individual facial feature localization and graph adjusting.

Global Face Search

The first stage serves to find a face in an image and provides near-optimal starting points for the following individual facial feature localization stage. All nodes of the *FBG* are positioned tentatively over the new face image, then moved together as a rigid object, at certain step distance, over the whole image. Similarity $S_F(I, FBG) =$

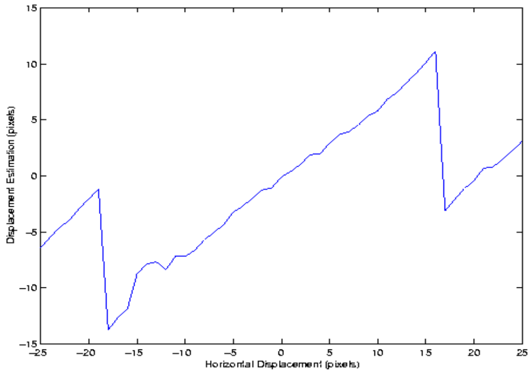


Figure 1. Displacement estimation by maximizing $\max_i S_p(J_n^I, J_n^{FBGi})$. J_n^{FBG} is the nose bunch of the *FBG*, J_n^I is taken from the pixel position of the same horizontal line as the nose tip in the face image. Only estimation in horizontal direction is shown

$\frac{1}{L}\sum_{k=1}^{L}\max_i S_m(J_k^I, J_k^{FBGi})$($L$ is the number of nodes of the *FBG*, corresponding to the L facial features to be detected) is calculated at each location and the best-fitting position serves as the starting points for the next stage.

Individual Facial Feature Localization

In this stage, each facial feature is localized individually, without taking its relative positions to other facial features into consideration. As explained earlier, at an image point, a complex vector could be obtained and a displacement $\vec{d}$ up to 18 pixels could be estimated by maximizing S_p. After the previous global face search stage, the possibility that a node of the *FBG* is within 18 pixels from the correct position of its corresponding facial feature in the new face image is relatively high. Thus, the $k^{th}(k = 1, 2 \ldots L)$ facial feature, starting from the point determined in the global face search stage, is localized individually by the following repeat process:

1. At a certain image point, a 40-dimensional complex vector J_k^I is extracted and $\vec{d}$ is estimated by maximizing $\max_i S_p(J_k^I, J_k^{FBGi})$
2. If $\left|\vec{d}\right| < 0.5$pixel or the repeat time exceeds the allowable maximum repeat time, exit the repeat process.
3. If $\left|\vec{d}\right|$ >allowable maximum one-step move distance D(an experimental value), $\vec{d} = \frac{|\vec{d}|}{D} * \vec{d}$.
4. The node is moved by $\vec{d}$ from the present point to a new point. Return to the step 1.

In our scheme, $\vec{d}$ is estimated not from all the 40 complex coefficients, but from only the last 16 complex coefficients, which are obtained by convolutions with the 16 Gabor filters with lowest center frequencies. This improvement both reduces the estimation time (by a factor of 3) by eliminating redundant information and increases the estimation accuracy by utilizing most reliable information. For all Gabor filters used in our experiments, the product of the center frequency and the scale of Gaussian window is a constant. Therefore the higher the center frequency, the smaller the effective width of the filter in spatial domain. Compared to coefficients corresponding to lower center frequencies, coefficients corresponding to higher center frequencies change more drastically with location and represent local features of a smaller area. Thus they provide less accurate information for the relative position of two points when the two points are not near to each other.

Graph Adjusting

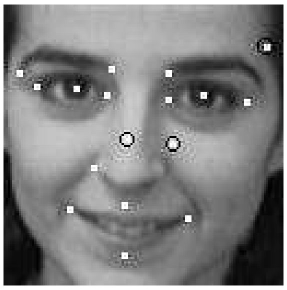
Not all 17 facial features are surely positioned correctly by individual facial feature localization process. When the starting point of a facial feature is not within the correct estimation range, it will be localized at a wrong point, which also satisfies the terminating condition $\left|\vec{d}\right| < 0.5$pixel, or at a random point when the repeat time reaches the allowable maximum repeat time. It is tested that an average of 13.2 out of 17 facial features per image could be positioned correctly by individual facial feature localization process. It testifies the high possibility of the starting point provided by global face search process falling into the correct estimation range and the validity of utilizing estimation $\vec{d}$ as a main tool for facial feature detection.

In graph adjusting stage, relative positions between facial features are utilized to localize those misplaced facial features. Correctly localized facial features are fixed, while positions of misplaced facial features are further changed to optimize the following similarity function:

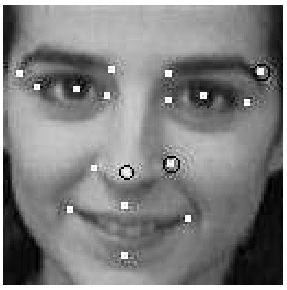
$$S(I, FBG) = \frac{1}{L}\sum_{k=1}^{L}\max_i S_m(J_k^I, J_k^{FBGi}) - \frac{\gamma}{E}\sum_{e=1}^{E}\frac{(\Delta X_e^I - \Delta X_e^{FBG})^2}{(\Delta X_e^{FBG})^2} \quad (6)$$

L is the number of nodes of *FBG* and E is the number of edges of *FBG*. γ is an experimental value that determines the relative importance of facial feature similarities and the topography term.

Before graph adjusting process, a heuristic program utilizes some common knowledge about the relative positions of human facial features to tell whether a facial feature is localized correctly or not. The common knowledge includes, for instance, two pupils are roughly on a horizontal line, nose is below and at the middle of pupils, two mouth corners are nearly symmetric about the philtrum and etc.



(a) Result of individual facial feature localization process



(b) Result of the whole facial feature detection process

Figure 2. Results of automatic facial feature detection. It can be seen that after individual facial feature localization stage, most facial features are correctly detected. In (a), three misplaced facial features are marked out by black circles. In (b), these misplaced facial features are corrected to an acceptable extend by graph adjusting process, but not as accurate as those facial features localized only by individual facial feature localization process.

3 Face Recognition

After facial feature detection, 17 basic facial features, each labeled with a 40-dimensional Gabor-based complex vector, are detected for each new face image. Face recognition is then executed on the basis of these complex vectors, which represent local features of the areas around the multiple facial features. Two novel non-parametric classification approaches are proposed here.

3.1 Two-Layer Nearest Neighbor *(TLNN)* approach

Nearest Neighbor (NN) approach is one of the most widely used non-parametric classification methods, for its simplicity and asymptotic property [7]. In *NN* approach, distances between a query image and all images in the gallery are calculated first, and the query image is classified to the class to which the image in the gallery obtaining the smallest distance belongs.

In normal *NN* approach, each image is represented by one feature vector in feature space. In our system, however, 17 complex vectors representing local features of different facial areas are available for each image. *TLNN* approach is therefore proposed by defining similarity between two images as the average over the first H highest facial feature similarities:

$$S(I^1, I^2) = \frac{1}{H}\sum_{j=1}^{H} S_m(J^1_{k_j}, J^2_{k_j}) \qquad (7)$$

$\{k_1, k_2 \ldots k_L\}$ is a permutation of $\{1, 2 \ldots L\}$ so that $S_m(J^1_{k_i}, J^2_{k_i}) \geq S_m(J^1_{k_j}, J^2_{k_j})$ when $1 \leq i < j \leq L$. L is the number of facial features detected for each image, $H = 5$ is chosen in our experiments.

The above image similarity is robust against changes in parts of face images caused by variations in illumination, expression, head rotation and other factors. This is because similarities of facial features in changed local areas are bound to be lower than the similarities of facial features in unchanged local areas, and therefore those changed parts in the image have no weight in the image similarity. It is, in nature, an execution of first-layer *NN*, in that of two images, only those parts with highest similarities (equivalent to smallest distances) are utilized to represent the similarity between the two images (equivalent to the distance between the two images). Another layer *NN* is then executed on thus-defined image similarity. Similarities between a query image and all images in the gallery are calculated and the query image is classified to the class to which the image in the gallery obtaining the highest similarity belongs.

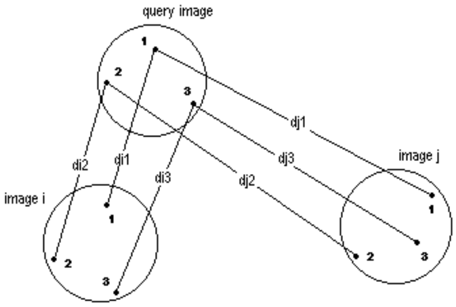


Figure 3. *Two-Layer Nearest Neighbor* **approach. The first layer** ***NN*** **is executed by choosing the smallest distance between corresponding facial features of two images as the distance between the two images, i.e. d_{i1} is taken as the distance between the query image and image i, and d_{j3} is taken as the distance between the query image and image j. The second layer** ***NN*** **is then executed on thus-defined image distance.**

3.2 Modular Nearest Feature Line *(MNFL)* approach

Nearest Feature Line (NFL) approach [5] is intended to take advantage of information contained within multiple prototypes of the same class for classification purpose, when multiple prototypes are available for each class. More specifically, in *NFL*, a linear subspace is constructed for each class by feature lines (FL) passing through each pair of prototypes in that class. Distance between a query image and its projection on to the subspace is calculated and then taken as the measure of the distance between the query image and the class.

Consider a change in the image space from point X_1 to X_2. The degree of the change may be measured by $\delta X = \|X_1 - X_2\|$. When $\delta X \rightarrow 0$, the locus of query image q due to the change can be approximated well enough by a straight-line segment between X_1 and X_2. The straight line passing through X_1 and X_2, denoted $\overline{X_1X_2}$, is called a feature line (FL) of the class that X_1 and X_2 both belong to.

The query image q is projected onto the FL $\overline{X_1X_2}$ as point p, as illustrated in Figure 4. FL distance between query image q and $\overline{X_1X_2}$ is defined as: $d_{FL}(q, \overline{X_1X_2}) = \|q - p\|$ where $\|\cdot\|$is some norm.

The projection point p could be represented as:

$$p = X_1 + \mu(X_2 - X_1) \tag{8}$$

where $\mu = \frac{(q-X_1)\cdot(X_2-X_1)}{(X_2-X_1)\cdot(X_2-X_1)}$ and $''.''$ stands for dot product.

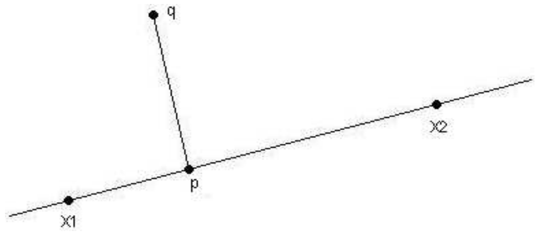


Figure 4. Generalizing two prototypes X_1 and X_2 by the feature line $\overline{X_1X_2}$. The query image q is projected onto the line as the projection point p.

FL $\overline{X_1X_2}$ provides information about linear variants of the two prototypes X_1 and X_2 , and virtually provides an infinite number of prototypic points of the class that X_1 and X_2 both belong to. The representational capacity of the prototype set of this class therefore is significantly expanded. Suppose there are T prototypes, represented by $X_i(i = 1, 2 \cdots T)$, in a certain class C. Then $C_T^2 = \frac{T(T-1)}{2}$ feature lines can be constructed for this class by one feature line passing through one pair of prototypes within the class. NFL distance between q and class C then is defined as:

$$D_{NFL}(q, C) = \min_{1 \le i < j \le T} d_{FL}(q, \overline{X_iX_j}) \tag{9}$$

In our system, *NFL* approach is extended to *Modular Nearest Feature Line (MNFL)* by constructing a subspace for each facial feature of each person class. Suppose the complex vector at the k^{th} facial feature of a query image is represented by J_k^q, and the complex vector at the corresponding k^{th} facial feature of the i^{th} image in a person class is represented by J_k^{Pi}. Then a subspace could be constructed from J_k^{Pi} ($i = 1, 2 \ldots T$, T is the number of images in this person class) by passing feature lines through each pair of J_k^{Pi} and J_k^{Pj}, denoted $\overline{J_k^{Pi}J_k^{Pj}}$ $(1 \le i < j \le T)$. NFL distance between J_k^q and this subspace then can be calculated as:

$$D_{NFL}^k(q, P) = \min_{1 \le i < j \le T} d_{FL}(J_k^q, \overline{J_k^{Pi}J_k^{Pj}}) \tag{10}$$

NFL distance between the query image and the person class, therefore, is defined as:

$$D_{NFL}(q, P) = \frac{1}{L}\sum_{k=1}^{L} D_{NFL}^k(q, P) \tag{11}$$

L is the number of facial features detected for each image.

MNFL treats possible variations in each local facial area independently. For example, suppose there is a query face image with glaring eyes and slightly opened mouth. There are other two images of the same person as the query image, one with closed eyes and closed mouth, and the other with normal open eyes and a laugh. Then, the complex vector extracted from the eye of the query image is an extrapolating point of the corresponding complex vectors extracted from the eyes of the later two images. The complex vector extracted from the mouth of the query image is an interpolating point between the corresponding complex vectors extracted from the mouths of the later two images. *MNFL* approach, therefore, is capable of covering dramatically more kinds of possible variations between images than uniform NFL approach in [5], where only one subspace is constructed for each person class

4 Experimental Results

Two experiments were done on two sets of images from different face image database. The first experiment was done on 200 images (5 images per person * 40 persons) from Cambridge database. These images mainly subjected to variations in viewpoint within a person class, and variations in race, age and gender between person classes. There were also slight expression variations within these images. The second experiment added another 100 images, (5 images per person * 15 persons) from Yale database and (5 images per person * 5 persons) from Harvard database. Images from Yale database mainly contained significant expression variations, while images from Harvard database mainly subjected to variations in illuminating angles. The purpose of adding these images was to introduce more significant variations in background, expression, and illumination, so that the difficulty of face recognition was further increased.

All images in our experiments were of frontal-viewed human faces and were warped to a standard size (128*128 pixels). In order to adjust for differences in lighting and camera setting, a technique, called histogram equalization, was utilized. In this technique, a grey value histogram was computed for each image. Depending on the shape of the histogram, a non-linear grey scale transfer function was computed and applied, to spread out intensity levels near histogram peaks and compress them near troughs.

In our experiments, when one image was chosen as the query image, the other four images in the same person class as the query image constituted a four-prototype person class, while other person classes were still made up of five images. Each image in the gallery was chosen as the query image once in our experiments.

No facial features above eyebrows or on the contour of a face were chosen in our system. Hair information therefore was not utilized for face recognition, and the effect of image background on recognition performance was also minimized.

Four recognition criteria were evaluated: 1) *NN*, the averages over all 17 facial feature similarities between the query image and all images in the gallery were calculated, and the query image was classified to the class to which the image in the gallery obtaining the highest average similarity belonged. 2) *TLNN*. 3) *NFL*, all 17 40-dimensional complex vectors were concatenated as a feature vector and *NFL* approach in [5] was executed. 4) *MNFL*. It can be seen that recognition accuracy of *TLNN* and *MNFL* is apparently higher than that of *NN* and *NFL* for both image sets. It testifies the superiority of the two recognition approaches proposed in this paper, both of which take advantage of the fact that in our system, multiple complex vectors, which represent local features of multiple facial areas, are available for each image.

Method	NN	TLNN	NFL	MNFL
Experimental I	93.0%	95.5%	94.0%	96.0%
Experimental II	91.6%	93.6%	92.6%	94.3%

Table 1. Comparison of face recognition accuracy of four recognition criteria

5 Conclusions

In this paper, an automatic face recognition system based on multiple facial features is presented. All facial features are detected automatically and significant improvements are made to achieve higher accuracy and efficiency of facial feature detection. A Gabor-based complex vector is utilized to represent each facial feature and two novel face recognition approaches, both of which are executed on the basis of these complex vectors, are proposed. *TLNN* is robust against changes in parts of face images by executing first-layer *NN* in defining image similarity, on which face recognition (second-layer *NN*) is based. *MNFL* covers dramatically more kinds of variation between images than uniform *NFL* by constructing a subspace for each facial feature. Experimental results show that *TLNN* and *MNFL* achieve apparently higher recognition accuracy than *NN* and *NFL* methods.

Both *TLNN* and *MNFL* are general non-parametric classification approaches and by no means specific to face recognition. They can be applied naturally to other object recognition task, provided multiple features of the same weight are available for each object.

References

[1] Reisfeld, D., Wolfson, H., and Yeshurun, Y., *Detection of Interest Points Using Symmetry*, ICCV'90, Osaka, Japan, Dec.1990

[2] Vincent, J. M., Waite, J. B., and Myers, D. J., *Automatic Location of Visual Features by a System of Multilayered Perceptrons*, IEE Proceedings, Vol. 139, no. 6, Dec. 1992

[3] Pentland, A., Moghaddam, B., and Starner, T., *View-based and Modular Eigenspaces for Face Recognition*, IEEE Proceedings, Computer Vision and Pattern Recognition, 1994

[4] Laurenz Wiskott, Jean-Marc Fellous, Norbert Kruger, and Christoph von der Malsburg, *Face Recognition by Elastic Graph Matching*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, no. 7, July 1997

[5] S. Z. Li, J. Lu. *Face Recognition Using the Nearest Feature Line Method*, IEEE Transactions on Neural Networks, 10(2):439-443, March,1999

[6] R.N.Bracewell, *The Fourier Transform and Its Applications*, New York, McGraw-Hill, 1978

[7] Keinosuke Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press INC. Orlando, Florida, 1972