Technical Report, IDE0852, May 2008

Multiview Face Detection Using Gabor Filters and Support Vector Machine

Bachelor's Thesis in Computer Systems Engineering

GÜL ÖNDER
AYDIN KAYACIK

School of Information Science, Computer and Electrical Engineering

Halmstad University

ABSTRACT

Face detection is a preprocessing step for face recognition algorithms. It is the localization of face/faces in an image or image sequence. Once the face(s) are localized, other computer vision algorithms such as face recognition, image compression, camera auto focusing etc are applied. Because of the multiple usage areas, there are many research efforts in face processing. Face detection is a challenging computer vision problem because of lighting conditions, a high degree of variability in size, shape, background, color, etc. To build fully automated systems, robust and efficient face detection algorithms are required.

Numerous techniques have been developed to detect faces in a single image; in this project we have used a classification-based face detection method using Gabor filter features. We have designed five frequencies corresponding to eight orientations channels for extracting facial features from local images. The feature vector based on Gabor filter is used as the input of the face/non-face classifier, which is a Support Vector Machine (SVM) on a reduced feature subspace extracted by using principal component analysis (PCA).

Experimental results show promising performance especially on single face images where 78% accuracy is achieved with 0 false acceptances.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

1. INTRODUCTION

Nowadays usage of computers has quite spread out. There is a lot of research on areas of computer vision such as face processing in visual. Face recognition research aims to teach the computer to detect and recognize the human faces. Solution of the face detection problem is the first step of such researches related to computer face processing. If faces are localized and detected in an image or video, then researchers can develop more effective and friendly methods for human computer interaction, surveillance and security systems. With the ubiquity of new information technology and media, such systems can be used massively.

1.1 Face Detection

Face detection can be simply defined as determining whether a human face is present or not in a scanned or digitalized photo, an image or a video, and reporting the location of each face(s) if there are any.

1.2 Motivation of Face Detection

Applications and difficulty of face detection, makes the problem interesting. In terms of applications, face detection is so important that the face detection problem is the first step of most applications; for example face recognition systems. Researchers have mainly focused on face recognition problem so far but recently face detection has attracted great attention, because face recognition systems primarily needs face detection systems especially for images that has a noisy background.

Although there is a lot of method for face detection, face detection problem is still difficult to solve. Because there is no exact shape of human faces and all of the faces are not so similar in an image so this makes the problem difficult. Figure 1 illustrates some example faces. The following are some of the reasons why face detection is difficult [1].

Pose: As to relative camera-face pose image of the face can vary. For example face image may be profile, frontal, and 45 degree, in this case same face seems different. Additionally, some facial features such as an eye or a nose may be partially or totally occluded.

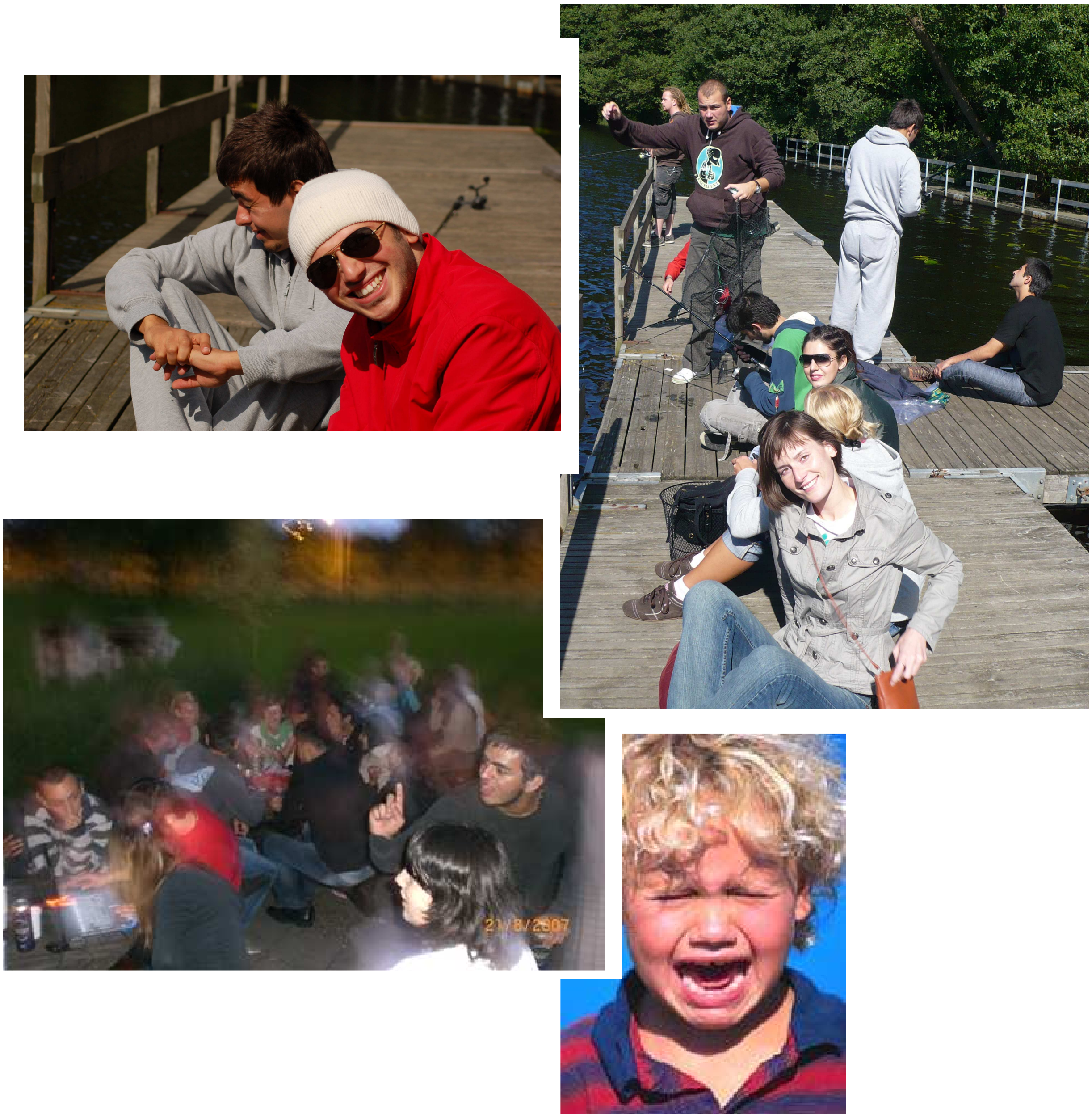


Figure 1:1 Examples of several variations that make face detection problem difficult

Structural components: Facial features such as glasses, beards, mustaches and make-up may or may not be present and these may vary excessively, also there is a great deal of variability of their shape, color and size so it affects the face image.

Facial expression: Person’s facial expression directly affects the appearance of faces. Namely when person is laughing or crying.

Occlusion: Faces in an image may not be whole, in other words faces can be partially occluded by a person or an object.

Image orientation**:** Different rotations of the camera's optical axis cause variety of face images.

Imaging conditions: Appearance of a face is affected by camera characteristics such as lenses, sensor response and lightning conditions such as spectra, source distribution and intensity.

1.3 Scope of the thesis

Face detection, as the name implies is localization of faces in images and/or video. Due to time limitation on working this thesis, the face detection system developed is only for gray scale images. This work can be extended to work on color images by first converting the color images to gray scale and then applying the algorithm. Moreover, the algorithm can work on image sequences or video by extracting each image/frame from the video and applying the algorithm.

1.4 Outline of Thesis

This thesis is organized as follows. Chapter 2 introduces background of face detection research, including techniques in gray scale images and provides some information on existing face databases. Chapter 3 describes design of Gabor filter and feature extraction from image. Chapter 4 is about Principal component analysis (PCA) that is used to reduce feature subspace. Chapter 5 introduces background of SVM and data preparation. Chapter 6 presents experimental results and chapter 7 summarizes the conclusions of this thesis and gives some future directions.

CHAPTER 2

2 BACKGROUND

Face detection is an important and first step of any system related to human face analysis. Face recognition researches started early with single face images and later face detection problem gained more attention. Research in computer vision and pattern recognition is increasingly being attracted to face detection problem. Accordingly, there are a variety of methods attributed to face detection.

Literature proves that a lot of face detection methods have been reported. Some reported segmentation schemes [2, 3] are using generalized face shape rules, motion, and color information. Beside this face detection is possible in cluttered scenes and variable scales by using probabilistic [4] and neural network methods [5].

2.1 Face Detection Methods

Several researchers grouped face detection methods in different categories. Yang, Kriegman and Ahuja [1] grouped face detection methods into four categories as illustrated on Table 2:1.

Knowledge -Based Method	Top-Down Methods
Feature -Based Method	Bottom-Up Methods
Template Matching Method	Predefined Face Templates Active Shape Model
Appearance-Based Method	Eigenfaces Distribution-Based Method Neural Network-Based Approach Support Vector Machine Sparse Network of Winnows Naïve Bayes Classifier Information-Theoretical Approach Inductive Learning

Table 2:1 Face Detection Methods

Knowledge-based method can also be called as rule-based method. This method is based on rules related to human face. These rules contain the relation of the facial features.

In ***feature-based method***, facial features are used to detect the human faces. Main idea is that the facial features are invariant and they exist even when the pose, viewpoint, or lighting conditions change.

In ***template matching methods*** some patterns are stored to describe the whole face or the facial features separately. The correlations between an input image and the stored patterns are computed for detection.

In ***appearance based methods***, the models (or templates) are learned from a set of training images in contrast to template matching method. These learned models are then used for detection.

2.1.1 Knowledge-Based Top-Down Method

In knowledge-based top-down methods, some previous knowledge, which can also be called rules, about face geometry is used like the features of a face and their relationships. For example an image contains two eyes, two eyebrows, a nose and a mouth and related to their relative distances and positions some rules must be known like each of eyes and eyebrows are symmetric etc. Known facial features are extracted in an image and location of face or facial feature candidates are determined based on facial rules coded by researcher. Finally, verification process is applied to reduce the false detections.

Although knowledge-based method seems to apply on images easily, creating rules from human faces is really difficult. If rules are so detailed program cannot be able to find all faces, but if the rules are simple, program considers non-face parts as face. Additionally if faces are in different poses, they could not be found. This is the challenging part in knowledge-based methods. But this method works well to detect the frontal faces in image.

Yang and Huang developed a system with three levels of rules to detect faces using hierarchical knowledge-based method [3]. At the higher level a predefined window is scanning the whole image and checking the part with respect the predefined rules at each location. These rules are simple facial features, detailed facial feature rules are exist in lower levels. In this research new mosaic image is created and each of the mosaics constructed by averaging the local intensity from corresponding part of the original image. Here part of image with the lowest resolution is accepted that "center part of the face has four cells with a basically uniform intensity", in other words it is a coded rule. Beside this “the upper round part of a face has a basically uniform intensity," and "the difference between the average gray values of the center part and the upper round part is significant."

At the highest level face candidates are searched in the lowest resolution image. At the middle level local histogram equalization is performed on the face candidates constructed in the highest level, and edge detection method is applied. In the lowest level controlled with other rules that contains facial features responds to eyes or mouth. This system is tested with 60 images, and the detection rate is 83%. But the system also finds false face detections in 28 images. This detection rate is not so high but this research gives idea of using multi resolution hierarchy and rules have been used in latter face detection researches like [7].

A rule-based localization method which is similar to [57] is represented by Kotropoulos and Pitas [7]. Kanade [57] successfully used to locate the boundary of a face and with this projection method, facial features are located. Let I(x,y**)** be the intensity value of an mxn image at position (x, y), the horizontal and vertical projections of the image are defined as:

$$HI(x) = \sum_{y=1}^{n} I(x,y), \text{ and } VI(y) = \sum_{x=1}^{m} I(x,y), \qquad (1)$$

Vertical horizontal values are firstly obtained, and then with respect to this histogram, two local minima are determined. This process is made by detecting sharp changes in HI. After that it is considered that these local minima correspond to left and right side of head. Then, vertical histogram values are used to detect the locations of mouth, eyes and nose. This method is tested with ACTSM2VTS (Multimodal Verification for Teleservices and Security applications) database [8] and each image has one face and a uniform background. Detection rate of the system is 86.5%. It can be considered successful. But when the background becomes complex it is difficult to find the face in an image, additionally if input image contains multiple faces, system couldn't be able to find the faces.

2.1.2 Feature-Based Bottom-Up Methods

In bottom-up method, researchers are trying to find invariant facial features for face detection. Main idea in this approach is that human can easily detect also recognize the faces in different poses, lighting conditions etc, so there should be some properties or features related to face. Several methods are proposed to detect the invariant facial features and later to find the presence of face. Here, these methods are separated into four groups:

2.1.2.1 Facial Features

Sirohey [9] proposed a localization method whose aim is to segment a face from cluttered background for face detection. By using an edge map (Canny detector [10]) and heuristics to remove and group edges, only the edge of face can be preserved. This face contour is an elliptic curve which separates the head region from the background. In the testing stage 48 images with cluttered background is used and 80% accuracy is achieved. Chetverikov and Lerch [11] propose to use blobs and streaks (linear sequences of similarly oriented edges) instead of using edges. In this method face model, eyes, cheekbones and nose is represented with two dark blobs and three light blobs. And streaks are used to represent the outlines of the faces, eyebrows and lips. The spatial relationship among the blobs is encoded by using two triangular configurations. To make the blob detection easy, a low resolution Laplacian image is generated and then to find the triangle occurrences as a face candidate image is scanned. If there are streaks around candidates, this is considered as face.

Graf *et. al*. [12] proposed to locate the facial features in grey scale images by graph. Here band pass filter is used and by applying morphological operations, regions with high intensity are enhanced, and these regions have certain shapes like eyes, mouth etc. Generally a prominent peak can be seen in the histogram of the processed image and by considering this peak value and its width adaptive threshold values are selected and by using these values two binarized images are generated. In these two binarized images connected components are identified to determine the areas of candidate facial features. Combinations of such areas are evaluated with classifiers and then based on the evaluation it is determined whether if there is a face and if exist it is determined where the face is present. The system is tested with head-shoulder images of 40 individuals and five video sequences that contain frames between 100 and 200.

A small set of spatial image invariants is used to describe the space of face pattern by Sinha [14]. The main idea is the local structure of brightness distribution of a human face remains largely unchanged however illumination and other changes can significantly alter brightness level at different parts of face. In his scheme these observed brightness regularities are encoded as a ratio template and uses it for pattern match. A ratio template is a coarse spatial template and this spatial template corresponds to a face with a few appropriately chosen sub regions that can roughly be considered key facial features such as eyes, cheeks and foreheads. An appropriate set of pair wise brighter-darker relationships corresponding sub regions capture the brightness constraints between facial features and then if image satisfies all the pair wise brighter-darker constraints it is considered as a face.

Han *et. al.* [15] presented a morphology-based technique and uses eye-analogue segments for face detection. First, eye-analogue pixels are located in the original image. Then, some morphological operations such as closing clipped difference are applied for localization to the original image. In the following process eye-analogue segments are generated and based on these segments potential face regions are searched. These regions are considered as potential face regions if they have possible geometrical combinations of eyes, nose, eyebrows and mouth. Last process is to verify these candidates by using neural network similar to [5]. Testing results show that system works with 94% detection rate.

2.1.2.2 Texture

Augusteijn and Skufca [17] developed a method based on face texture. The underlying idea is that faces have distinct textures, so they can be distinguished from other object by using these textures. This method decides the presence of a face through the identification of face-like textures. Using second-order statistical features (SGLD) [18], the texture are computed on sub images of 16x16 pixels. Three facial features; skin, hair and others are considered. In this method cascade correlation neural network [19] for supervised classification of textures and a Kohonen self-organizing feature map [20] is used to form clusters for different texture classes. To decide the existence of face, it is suggested using votes of the occurrence of hair and skin textures. But only the texture classification result is reported. Other research [21] is made by Dai and Nakano, also used SGLD model to face detection. In this method color information and the face-texture model is combined. Their system scans the image and compares the sub regions with face texture model and the orange-like color. If color and texture model is similar with the model, sub region is considered as a face. This method enables to detect faces which are not upright or have features such as beard, glasses.

2.1.2.3 Skin Color

Human skin color is an effective feature used in many methods for face detection. It can be seen easily that different people have different skin color [1]. However, the main difference in appearance of faces lies in intensity rather than color itself [14], [15]. Some methods using skin color, utilize different color spaces like RGB [22], [23], normalized RGB [24],[25],[26], HSV(or HSI), [27][28][29], YIQ[30][21], YCrCb[32][33], YES [34], CIE XYZ[35] and CIE

LUV[2]. Generally single Gaussian or a mixture Gaussians is used to build a skin color model.

Usually skin color alone is not effective to detect the faces. In recent researches, systems are using color segmentation with shape analysis and motion information together with skin color to detect faces. The next section discusses methods that use multiple features.

2.1.2.4 Multiple Features

In recent researches, methods are combining several features for face detection. Mostly skin color, size and shape are utilized to find the candidate faces, and then this candidates are verified using local, detailed facial features like eyes, nose, hair etc. This kind of methods begins with Dai and Nakano's [20]; they detect the skin-like regions as mentioned before. Then, skin-like pixels are grouped together using methods like connected component analysis or clustering algorithms. These grouped areas considered as a face candidate if the region has an elliptic or oval shape. In the final step local features are used for verification.

A method using shape and color for face localization and feature extraction is proposed by Sobottka and Pitas [37]. Firstly skin-like regions are located by using color segmentation in HSV space. Connected components are then determined and for each of them the best fit ellipse is computed. Some of them selected as a face candidate if they are well approximated by an ellipse. In the following step candidates are searched for facial features. In an image eyes and mouths are darker so based on this observation facial features are extracted. In [38] and [39] Gaussian skin color model is utilized to classify skin color pixels [1]. A set of 11 lowest-order geometric moments are computed using Fourier and radial Mellin transforms and then this set characterize the shape of the clusters in the binary image. For detection part they trained a neural network with the extracted geometric moments. This method is tested with 100 images and returned 85% accuracy.

Kim [40] proposed a method that uses range and color for face detection. In this method disparity maps are computed and based on the assumption that background pixels have the same depth and they outnumber the pixels in the foreground objects; objects are segmented. Later segmented regions with a skin-like color are classified as faces by using a Gaussian distribution in normalized RGB color space.

2.1.3 Template Matching Method

In template matching methods a standard face pattern is utilized and this pattern is usually frontal face. This pattern is manually predefined or parameterized by a function. The correlation values with the standard patterns are computed for the face contour, eyes, nose, and mouth independently on a given input image. This approach is simple to implement however it lacks the capacity of detecting face with variations in scale, pose and shape. But other techniques such as multiresolution, multiscale, subtemplates, and deformable templates are proposed to achieve scale and shape invariance.

2.1.3.1 Predefined Templates

An earlier research is made by Sakai [41] to detect frontal faces in photographs. In this method several templates for eyes, nose, mouth etc are used and they defined in terms of line segments. These templates are matched with the input image lines. The image lines should first be extracted based on greatest gradient change. Face candidates and their locations are detected computing the correlations between sub images and contour templates and then they are matched against the sub templates. When a matching found on pixel (x, y) of the input image, a region bordered with (x+h, y+h) is searched with other templates to find the extent of the face.

2.1.3.2 Deformable Templates

Deformable templates are used by Yuille [42]. In this method facial features like eyes are modeled with fitted with *a priori* elastic model. In other words they are described by parameterized templates. An input image contains edges, peaks and valleys, and predefined energy function links them to corresponding parameters in the template. And then parameters of the energy function are minimized and it allows finding the best fit of the elastic model. In tests performance of the method was good in tracking no rigid features, but disadvantages of this approach is that the deformable template must be initialized in the proximity of the object of interest.

2.1.4 Appearance-Based Method

In appearance-based method, certain window is usually scanned through the image and then related part of an image grouped into two patterns as face or non-face. Generally a given image is resized to detect the faces which have different sizes. Alternatively the size of the sample can be calibrated appropriately to detect faces with different scales. Face detection by using this method, is based on finding the differences between face and non-face patterns. To distinguish between face and non-face patterns, many pattern recognition techniques have been used. Methods that rely on pattern recognition will be presented later. The following subsections are devoted to well-known appearance based methods.

2.1.4.1 Eigenfaces

Sirovich and Kirby developed the eigenface method for face recognition and it is used for face recognition by Matthew Turk and Alex Pentland [43]. Turk and Pentland applied the principal component analysis concept to the face detection and recognition. By using PCA performed on a training set of face images, representation of faces in face space, obtained by the eigenvectors of the training data can be generated. Face images are projected onto the subspace and this subspace is obtained by using largest M eigenvalues, then projected images are clustered. In the same way, non-face images are projected onto subspace to be clustered. Face images do not totally change when projected onto subspace, but as to images of the non-faces, they seem quite different when they are projected onto subspace. Interval between face space and image region is calculated for all locations on image for the face detection process. The distance from face space is considered as a measure of "faceness", and at every point in the image results of the calculated distance from face space is a "face map". Consequently face can be detected by considering local minima points in this face map.

2.1.4.2 Distribution-based methods

In the appearance-based methods, it can be said that face detection problem is a pattern recognition problem. Generally object detection methods works by taking into account the results of probability $P(x|\Omega=w_c)$ for x to belong to the object class w_c, where x equals to $\{x_1, x_2, \ldots\ldots, x_N\}$ that is the pixel values of an image window and Ω equals to $\{w_1, w_2, \ldots, w_M\}$ that is the set of object classes. For face detection, Ω equals to {1, -1} because there are two classes; face and non-face. In distribution-based methods, probability $P(x|\Omega=w_c)$ is tried to estimate and then maximize for x to belong to the object class w_c.

A distribution-based system for face detection developed by Sung and Poggio [44], [45] demonstrates how the distributions of image patterns from one object class can be learned from positive and negative examples (i.e., images) of that class. They used the multilayer perceptron classifier and distribution-based models for face and non-face patterns. Each face and non-face example is initially normalized and processed to a 19x19 pixel image and treated as a 361-dimensional vector. Then the patterns are grouped into 12 clusters, 6 for faces and 6 for non-faces, using a modified k-means algorithm.

Each cluster is considered as a multidimensional Gaussian function with a mean image and a covariance matrix. In this method two distance metrics, that are normalized Mahalanobis distance and Euclidean distance, are computed between an input image pattern and the prototype clusters. The first distance component between the test pattern and the cluster centroid, measured within a lower-dimensional subspace spanned by the cluster's 75 largest eigenvectors and second distance component is between the test pattern and its projection onto the 75-dimensional subspace. This Euclidean distance component accounts for pattern differences not captured by the first distance component. Finally multilayer perceptron (MLP) classifies face window patterns from non-face patterns. To make this classification, MLP uses the twelve pairs of distances to each face and non-face cluster. The classifier is trained using standard back propagation from a database of 47,316 window patterns, 4,150 of them are positive examples of face patterns and the rest are non-face patterns.

Yang *et al.* [46] present two methods using a mixture of linear subspaces. Common factor analysis (FA) is used in the first method. Factor analysis can be described as a statistical method to model the structure of high dimensional data using only a small number of latent variables. This analysis assumes that the variance of a single variable can be decomposed into common variance and unique variance. FA only analyzes the common variance of the observed variables. Later parameters of the model are estimated using an expectation minimization.

The second method is the Fisher Linear Discriminant that is used in [46] as second method, and this method forms the basis for the proposed method for detecting faces in gray-level images in this thesis. In [46], first, the data set is clustered into 25 face and 25 non-face classes using a Self Organizing Map (SOM), then Fisher Linear Discriminant determine a projection matrix, and this matrix is used to maximize the ratio the between-class variance and within-class variance, so the training set is projected on this subspace and Gaussian distributions are used to model each class conditional density. Finally using maximum likelihood estimation, parameters of the model are estimated. Testing results are remarkable and reported for both methods in [46]. But in this research there is no information about generic rules for selecting parameters such as number of clusters for the face and non-face class.

2.1.4.3 Neural Network-Based Approach

Rowley *et al*. proposed a method [47] using neural network. Fig 2.6 illustrates how the algorithm works. Neural network is used to distinguish face and non-face patterns. Bootstrap method mentioned before in [45] is used to generate non-face patterns. In this method with the same training process but with random initial weights and random initial non-face images, the detection results of faces don’t change. But if different algorithms are used, they give different errors or false alarms. But it is possible to decrease number of the false alarms produced by system by using multiple networks and some heuristics. This system is very effective in detecting frontal faces.

2.1.4.4 Sparse Network of Winnows

A SNoW (Sparse Network of Winnows) learning method is proposed by Yang [48] [49]. This method can detect faces with different features and expressions, in different poses, and under different lighting conditions [50]. SNoW is a sparse network of linear functions that utilizes the Winnow update rule [51]. This learning method is adopted for learning in domains in which the number of features is very large. Snow has some characteristics such as sparsely connected units, the allocation of features and links in a data driven way, the decision mechanism, and the utilization of an efficient update rule. Some face databases like Olivetti [52], UMIST [53], Harvard [54], Yale [55], and FERET [56], are used for training the system. Reported results shows that SNoW method is promising and have an error rate of 5.9%.

2.1.4.5 Naive Bayes Classifier

Schneiderman and Kanade [57] propose a method that estimates the joint probability of local appearance and position of face patterns. Based on the idea that some local patterns of an object are more unique than others, local appearance is emphasized in the method. For example the intensity patterns around the eyes are much more distinctive than the pattern found around the cheeks. This method has an advantage because of providing better estimation of the conditional density functions of these sub regions. Another reason for using this method is that a naive Bayes classifier provides a functional form of the posterior probability to capture the joint statistics of local appearance and position on the object. Decomposing a face image, four rectangular sub regions are created to be projected to a lower dimensional space using PCA and quantized into a finite set of patterns, and the statistics of each projected sub region are estimated from the projected samples to encode local appearance. Finally existence of a face is determined by comparing likelihood ratio and prior probability ratio. If the likelihood ratio is bigger than the other, it is considered that there is a face. Experimental results show 93% detection rate.

2.1.4.6 Information-Theoretical Approach

The spatial property of face pattern can be modeled through the contextual constraint which is generally specified by a small neighborhood of pixels in a face pattern. Context-dependent entities such as image pixels and correlated features are modeled using Markov random field (MRF) theory. Model is achieved by characterizing mutual influences among such entities using conditional MRF distributions. And then using histograms the face and non-face distributions can be estimated.

Huang *et. al.* [59] applied the Kullback relative information to face detection by maximizing the information-based discrimination between positive and negative examples of faces. This method analyzes the images used for training as observation of a random process and characterize with two probability functions. A family of discrete Markov processes is used to model the face and background patterns and to estimate the probability model. At this step process is an optimization problem, Markov process is selected and this maximizes the information-based discrimination between the two classes. Using the trained probability model, the likelihood ratio is computed, and with respect to this ratio value it is decided that there is a face or not.

2.1.4.7 Inductive Learning

Quinlan's C4.5 algorithm [16] is used to detect and locate the face. Huang propose a method using this algorithm to create decision tree from positive and negative examples of face patterns. These face pattern examples are 8x8 pixel windows and they are defined as a vector with 30 attributes pixel intensity values such as mean, entropy etc. C4.5 algorithm constructs a decision tree using these examples. Leaves of the decision tree indicate class identity and nodes specify tests to perform on a single attribute [1]. Using this trained tree it is decided that whether there is a face in the input image or not.

2.2 Face Image Databases

As it is explained, there are several face detection methods in literature. However, little attention is given to the development of an image database for face detection. Face databases are important because they are necessary for most methods which need to be trained with a training set. If effective face database is constructed, face detection methods may give better results, so face databases are important to test the detection methods. Face detection methods are giving different detection rates when they are tested with different databases. When two or more face detection methods are wanted to be fairly compared, they should be tested with the same database [1]. So it can be concluded that some methods are effective with specific databases. Some of the existing face databases are discussed below.

The Color FERET Database

The FERET database [56] was collected in 15 sessions between August 1993 and July 1996. The database contains 1564 sets of images for a total of 14,126 images that includes 1199 individuals and 365 duplicate sets of images. A duplicate set is a second set of images of a person already in the database and was usually taken on a different day.

For some individuals, over two years had elapsed between their first and last sittings, with some subjects being photographed multiple times. This time lapse was important because it enabled researchers to study, for the first time, changes in a subject's appearance that occur over a year.

The Yale Face Database

Yale Face Database [62] contains 165 grayscale images in GIF format of 15 individuals. There are 11 images per subject, one per different facial expression or configuration: center-light, w/glasses, happy, left-light, w/no glasses, normal, right-light, sad, sleepy, surprised, and wink.

The Yale Face Database B

The Yale Face Database [63] contains 5760 single light source images of 10 subjects each seen under 576 viewing conditions (9 poses x 64 illumination conditions). For every subject in a particular pose, an image with ambient (background) illumination was also captured.

PIE Database, CMU

PIE Database [64] is a database containing 41,368 images of 68 people. The images of each of these subjects are recorded under 13 different poses, 43 different illumination conditions, and with 4 different expressions. That's why this database is called the CMU Pose, Illumination, and Expression (PIE) database.

AT&T "The Database of Faces" (formerly "The ORL Database of Faces")

This database [67] contains ten different images of each of its 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open / closed eyes, smiling / not smiling) and facial details (glasses / no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement).

BioID Face Database

The dataset consists of 1521 gray level images with a resolution of 384x286 pixels. Each one shows the frontal view of a face of one out of 23 different test persons. For comparison reasons the set also contains manually set eye positions. The images are labeled "BioID_xxxx.pgm" where the characters xxxx are replaced by the index of the current image (with leading zeros). Similar to this, the files "BioID_xxxx.eye" contain the eye positions for the corresponding images.

Cohn-Kanade AU Coded Facial Expression Database

Cohn-Kanade AU-Coded Facial Expression Database [66] is constructed by taking photographs of 100 university students. Their ages ranged from 18 to 30 years. Sixty-five percent were female, 15 percent were African-American, and three percent were Asian or Latino. Subjects were instructed by an experimenter to perform a series of 23 facial displays that included single action units (e.g., AU 12, or lip corners pulled obliquely) and action unit combinations (e.g., AU 1+2, or inner and outer brows raised). Image sequences from neutral to target display were digitized into 640 by 480 or 490 pixel arrays with 8-bit precision for grayscale values.

MIT-CBCL Face Recognition Database

The MIT-CBCL face recognition database [68] contains face images of 10 subjects. It provides two training sets: High resolution pictures, including frontal, half-profile and profile view and the second is synthetic images (324/subject) rendered from 3D head models of the 10 subjects. The head models were generated by fitting a morphable model to the high-resolution training images. The 3D models are not included in the database. The test set consists of 200 images per subject. We varied the illumination, pose (up to about 30 degrees of rotation in depth) and the background.

Image Database of Facial Actions and Expressions - Expression Image Database

24 subjects are represented in Image Database of Facial Actions and Expressions, yielding between about 6 to 18 examples of the 150 different requested actions. Thus, about 7,000 color images are included in the database, and each has a matching gray scale image used in the neural network analysis.

CHAPTER 3

3 METHODOLOGY

3.1 GABOR FILTER

In appearance-based face detection methods, a window with a predefined size, scans the image for each location and categorizes current region with a classifier as face or non-face region. To categorize the sub images, the classifier first needs to be trained. Facial features are constructed from images which are separated before a face and non-face images, is used to train the classifier. And then it can be said that the system is ready to detect the faces in the input images. When an image is given as an input, a window scans it and for each location a facial features are constructed and then with respect to these features classifier determines if current sub image contains a face or not.

In this thesis support vector machine is used as classifier and Gabor filter is used for feature extraction. Moreover, Principle Component Analysis (PCA) method is used to reduce the dimensionality of feature vector to minimize the complexity and the computation cost. This chapter explains Gabor filters.

3.1.1 Gabor Filter Design

Gabor filter has advantages for having optimal localization properties in both spatial and frequency domain, so it is used in several object detection methods as well as face detection methods.

Shiguang *et. al.*[31] defined 2D Gabor filter specified in space and spatial frequency domain as follows:

$$\psi_{\mathrm{u,v}}(\mathrm{Z}) = \frac{\left\|k_{u,v}\right\|^2}{\sigma^2} e^{\left(\frac{\left\|k_{u,v}\right\|^2 \left\|z\right\|^2}{2\sigma^2}\right)} \left[e^{i\bar{k}_{u,v}z} - e^{-\frac{\sigma^2}{2}} \right] \tag{1}$$

In this equation;

$k_{u,v} = k_v e^{i\phi_u}; k_v = \frac{k_{\max}}{f^v}$ gives the frequency (2)

$\phi_{u,v} = \frac{u\pi}{8}, \phi_u \in [0,\pi)$ gives the orientation (3)

where $k_{u,v} = k_v e^{i\phi_u}$ is the oscillatory wave function, and its real part is cosine function and

$e^{i\bar{k}_{u,v}z}$ is the imaginary part is sinusoid function.

In equation (2), *v* controls the scale of Gabor filters, in other words it controls the frequency and *u* controls the orientation of the Gabor filter.

3.1.2 Gabor Feature Extraction

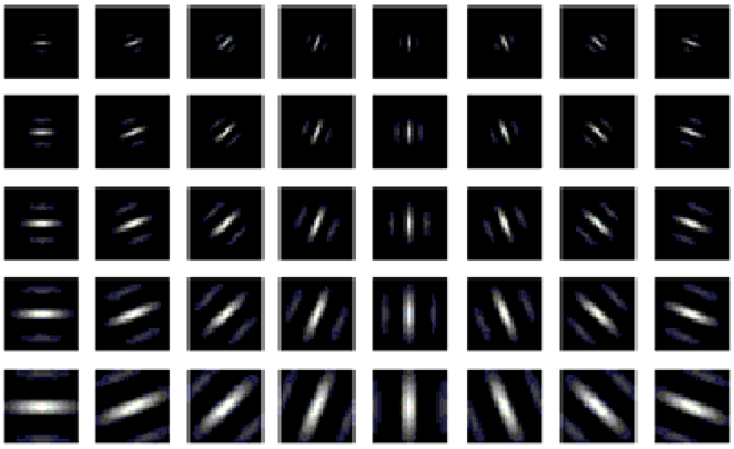
Shiguang *et. al.*[31] used Gabor filter with five frequencies corresponding to eight orientations. It can also be shown as:

$$v \in \{ 0,1,2,3,4 \} \text{ and } u \in \{ 0,1,2,3,4,5,6,7 \}$$

Also some other equations required for Gabor filter are:

$$\sigma = 2\pi,\ k_{max} = \pi/2 \quad \text{and} \quad f^{v} = \sqrt{2}$$

The real part of the Gabor filters with five frequencies and eight orientations is shown in Fig.3.1. From Fig.3.1 it can be seen that the Gabor filters exhibit strong characteristics of spatial locality and orientation selectivity. Fig 3.2 illustrates a face and its 40 Gabor filtered images.



Figur 3:1 The real part of the Gabor filters with five frequencies and eight orientations

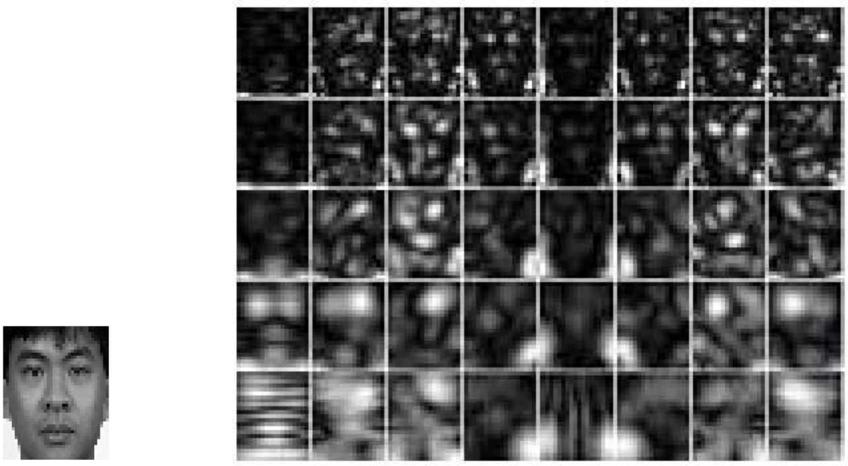


Figure 3:2 Face sample image and its forty Gabor filtered images

3.2 PRINCIPAL COMPONENT ANALYSIS

When feature vector is created with Gabor filter, its dimensions become very large. It is hard for classifier to cope with such big vector. At this point feature compression methods are used to reduce the dimensions. It is done by using linearly combining features in [47]. High-dimensional data are projected onto a lower dimensional space with these methods. There are two classical approaches to finding effective linear transformations; Principle Component Analysis (PCA) and Linear Discriminant Analysis (LDA). Difference between PCA and LDA is that, PCA seeks a projection that best represents the original data in a least-squares sense, and LDA seeks a projection that best separates the data in a least-squares sense [6]. In this thesis PCA is used.

3.2.1 Dimension reduction using PCA

Let's consider a set of N image $\{x_1, x_2 \ldots x_N\}$ which are represented by t-dimensional Gabor feature vector. PCA is used to find a linear transformation where the original t-dimensional feature space is projected onto r-dimensional feature subspace. It is clear that relation between r and t should be r << t. The new feature vector $y_i \in R^r$ is defined by;

$$\mathbf{y_i} = W_{pca}^{t} \; x_i \; (i = 1,2 \ldots N) \tag{4}$$

In this equation W_{pca}^{t} is the linear transformations matrix and x_i is the number of sample images[6].

The columns of W_{pca} are the r eigenvectors associated with the r largest eigenvalues of the scatter matrix S_T, which is defined as;

$$S_T = \sum_{i=1}^{N}(x_i - \mu)(x_i - \mu)^T \quad (5)$$

In this equation μ Є R^t is the mean image of all samples.

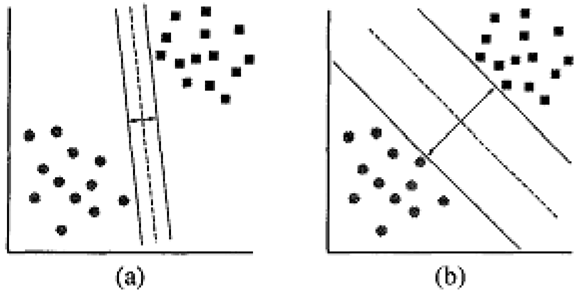
The disadvantage of PCA is that it may lose important information for discrimination between different classes.

3.3 SUPPORT VECTOR MACHINES

SVMs can be considered as a new paradigm to train polynomial function, neural networks, or radial basis function (RBF) classifiers. While most of the techniques used to train the above mentioned classifiers are based on the idea of minimizing the training error, which is usually called empirical risk, SVMs operate on another induction principle, called structural risk minimization, which minimizes an upper bound on the generalization error. An SVM classifier is a linear classifier where the separating hyper plane is chosen to minimize the expected classification error of the unseen test patterns. This optimal hyper plane is defined by a weighted combination of a small subset of the training vectors, called support vectors. Estimating the optimal hyper plane is equivalent to solving a linearly constrained quadratic programming problem.

3.3.1 Classification using SVM

Suppose there exist a dataset $D = \{(x_i, y_i)\}^l_{i=1}$ of labeled examples and y_i Є $\{-1,1\}$, the aim is to determine, among the infinite number of linear classifiers that separate the data, which one will have the smallest generalization error [61]. In this step a hyperplane that leaves the maximum margin between the two classes, where the margin is defined as the sum of the distances of the hyper plane from the closest point of the two classes, can be used. (Fig 3.3)



Figur 3:3 Separating hyperplane with a small margin (a) and a larger margin (b)

Even if the two classes are non-separable, it can be still searched for the hyper plane that maximizes the margin and that minimizes a quantity proportional to the number of misclassification errors. A positive constant C that is chosen before hand controls the trade off between margin and misclassification error. In this case it can be shown that the solution to this problem is a linear classifier

$$f(\mathbf{x}) = \operatorname{sign}(\textstyle\sum_{i=1}^{l} \lambda_i y_i \mathbf{x}^T \mathbf{x}_i + b) \qquad (6)$$

whose coefficients λ_i; are the solution of the following QP problem:

Minimize $\quad W(\Lambda) = -\Lambda^T 1 + \frac{1}{2}\Lambda^T D\Lambda$ (7)

Λ

Subject to

$$-\Lambda^T y = 0$$
$$\Lambda - C1 \leq 0$$
$$-\Lambda \leq 0$$

where $(\Lambda)_i = \lambda_i$, $(1)_i = 1$ *and* $D_i = y_i y_i x_i^T x_i$. It turns out that only a small number of coefficients λ_i are different from zero, and since every coefficient corresponds to a particular data point, this means that the data points associated to the non-zero coefficients determine the solution. These data points called support vectors are the only ones which are relevant for the solution of the problem: all the other data points could be deleted from the data set and the same solution would be obtained. The support vectors can be considered as data points and these points are located at the border between two classes. Their number is usually small, and it is showed that it is proportional to the generalization error of the classifier.

Since it is unlikely that any real life problem can actually be solved by a linear classifier, the technique has to be extended in order to allow for non-linear decision surfaces. This is easily done by projecting the original set of variables *x* in a higher dimensional feature space:

$$\mathrm{x} \in R^d \Rightarrow \mathrm{z(x)} \equiv (\phi_1(\mathrm{x}),\ldots,\phi_\mathrm{n}(\mathrm{x})) \in R^n \qquad (8)$$

and by formulating the linear classification problem in the feature space and by formulating the linear classification problem in the feature space. The solution will have the form (8) and therefore will be nonlinear in the original input variables [61]. At this point two problems arise; the choice of the features $\o_i(x)$ and the computation of the scalar product $z^T(x)z(x_i)$. The feature should be chosen in a way that allows a "rich" class of decision surfaces. If the number of features n is very large, it can be difficult to compute the scalar product. A possible solution to these problems consists in letting n going to infinity and makes the following choice [61]:

$$\mathbf{z}(\mathbf{x}) \equiv (\sqrt{\alpha_1}\psi_1(\mathbf{x}), \ldots, \sqrt{\alpha_i}\psi_i(\mathbf{x}), \ldots) \tag{9}$$

where α_i and ψ_i are the eigenvalues and eigenfunctions of an integral operator whose kernel K(x, y) is a positive definite symmetric function. With this choice the scalar product in the feature space becomes particularly simple because:

$$\mathbf{z}^T(\mathbf{x})\mathbf{z}(\mathbf{y}) = \sum_{i=1}^{\infty} \alpha_i \psi_i(\mathbf{x})\psi_i(\mathbf{y}) = K(\mathbf{x}, \mathbf{y}) \tag{10}$$

When equation (10) is chosen, it makes the scalar product in the feature space particularly simple. This equation comes from the Mercer-Hilbert-Schmidt theorem for positive definite functions [36]. The QP problem that has to be solved now is exactly the same as in equation (1), with the exception that the matrix D has now elements $D_{ij} = y_i y_j K(x_i, x_j)$ [61]. As a result of this choice, the SVM classifier has the form:

$$f(x) = \mathrm{sign}(\sum_{i=1}^{l} \lambda_i y_i K(x, x_i) + b \tag{11}$$

In Table 5.1 some choices of the kernel function are listed. Notice that how they lead to well known classifiers, whose decision surfaces are known to have good approximation properties.

Kernel Function	**Type of Classifier**
$K(x, x_i) = \exp(-\| x-x_i\|^2)$	Gaussian RBF
$K(x, x_i) = (x^T x_i + 1)^d$	Polynomial of Degree d
$K(x, x_i) = \tanh(x^T x_i - \Theta)$	Multilayer Perception

Table 3:1 Some possible kernel functions and the type of decision surface they define

CHAPTER 4

4 EXPERIMENTAL RESULTS

This system detects faces by normalizing an image patterns at 19 x 19 and classifying them using SVM to determine the appropriate class (face/non-face). We use 1000 images to extract face samples, which contain 1000 real faces from BioID database. Moreover, we use 471 face images from CMU database. Each manually cropped face box is normalized into 19 × 19 pixels and gives one face sample. For non-face patterns 2500 non-face images are used from CMU Database.

We can divide our system into two parts; first is training and second is test part.

4.1 Preparing Database

Support Vector Machine classification technique studied to classify a 19x19 window of pixels as a face/non-face. To correctly detect a face, the face must fit into the window and occupy all of it for the classifier; it must not be larger or smaller than the window. The system slides this 19x19 window across the image. We use face patterns from Bioid and CMU databases. For BioID face images, first step is cropping face patterns and second is normalizing them into 19x19 pixels.

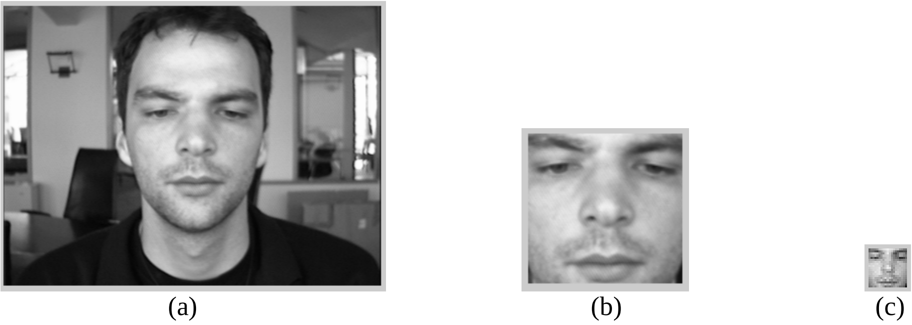


Figure 4:1 (a) Face Image from BioID database, (b) cropped image and (c) normalized image(19x19)

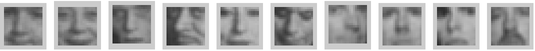


Figure 4:2 Face images from CMU database (19x19)

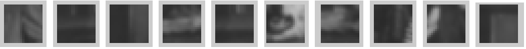


Figure 4:3 Nonface images from CMU database (19x19)

Gabor feature extraction

Gabor filter can capture salient visual properties such as spatial localization, orientation selectivity, and spatial frequency characteristics. Considering these excellent capacities and its great success in face detection and face recognition, we choose Gabor features to represent the face image.

The Gabor representation of an image is the convolution of the image with the Gabor filter. Based on the Gabor representations, a feature vector is formed. In our experiment we use the Gabor filters with the following parameters: five scales v ϵ {0 1 2 3 4} and eight orientations u ϵ {0 1 2 3 4 5 6 7}.

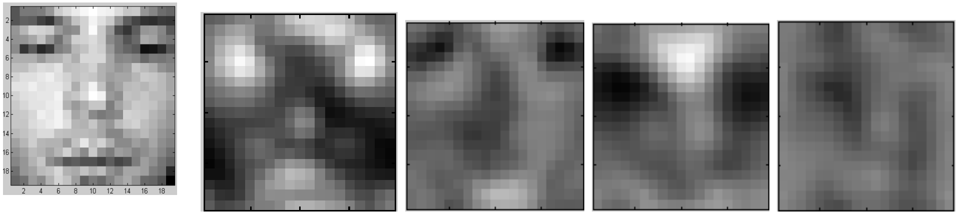


Figure 4:4 Face sample image and its Gabor filtered images.

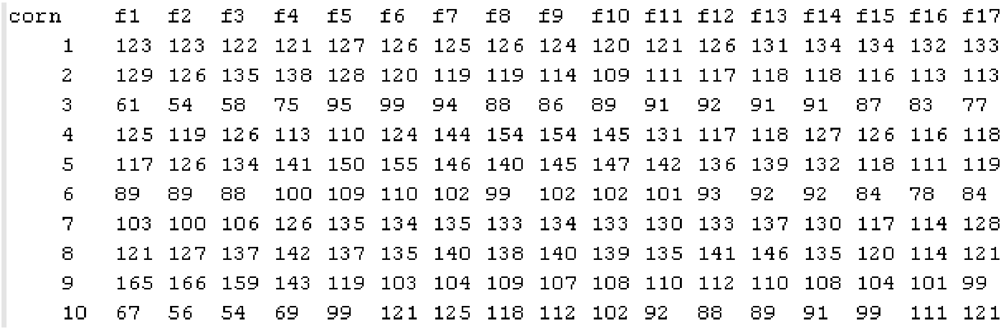
Figure 4.4 shows the Gabor filtered images of a face sample. The filtered images are visualized by coding the output values of Gabor filter in gray levels. We can see that the orientation properties of face pattern are well represented by the Gabor filtered images.

4.2 Feature reduction using PCA

We use PCA to reduce the dimensionality by linearly combining features. Linear methods project the high dimensional data onto a lower dimensional space (feature compression).With PCA we try to improve classification accuracy. From Gabor features we have 19x19x8x5 possible features; we reduce this feature size to 19x19x7 for an image.

4.3 Classifying by SVM

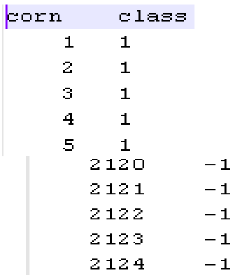
To classify the patterns we use Gist SVM software. The SVM server takes as input three files: a training data set, a corresponding set of classification labels, and a test data set. Each row in each of these files corresponds to one example. We prepared two files for classification patterns. For format of files; the first row should contain the name of each feature, and the first column should contain the name of each element in the data set.



corn	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	f11	f12	f13	f14	f15	f16	f17
1	123	123	122	121	127	126	125	126	124	120	121	126	131	134	134	132	133
2	129	126	135	138	128	120	119	119	114	109	111	117	118	118	116	113	113
3	61	54	58	75	95	99	94	88	86	89	91	92	91	91	87	83	77
4	125	119	126	113	110	124	144	154	154	145	131	117	118	127	126	116	118
5	117	126	134	141	150	155	146	140	145	147	142	136	139	132	118	111	119
6	89	89	88	100	109	110	102	99	102	102	101	93	92	92	84	78	84
7	103	100	106	126	135	134	135	133	134	133	130	133	137	130	117	114	128
8	121	127	137	142	137	135	140	138	140	139	135	141	146	135	120	114	121
9	165	166	159	143	119	103	104	109	107	108	110	112	110	108	104	101	99
10	67	56	54	69	99	121	125	118	112	102	92	88	89	91	99	111	121

Figure 4:5 Example of our training file for SVM.

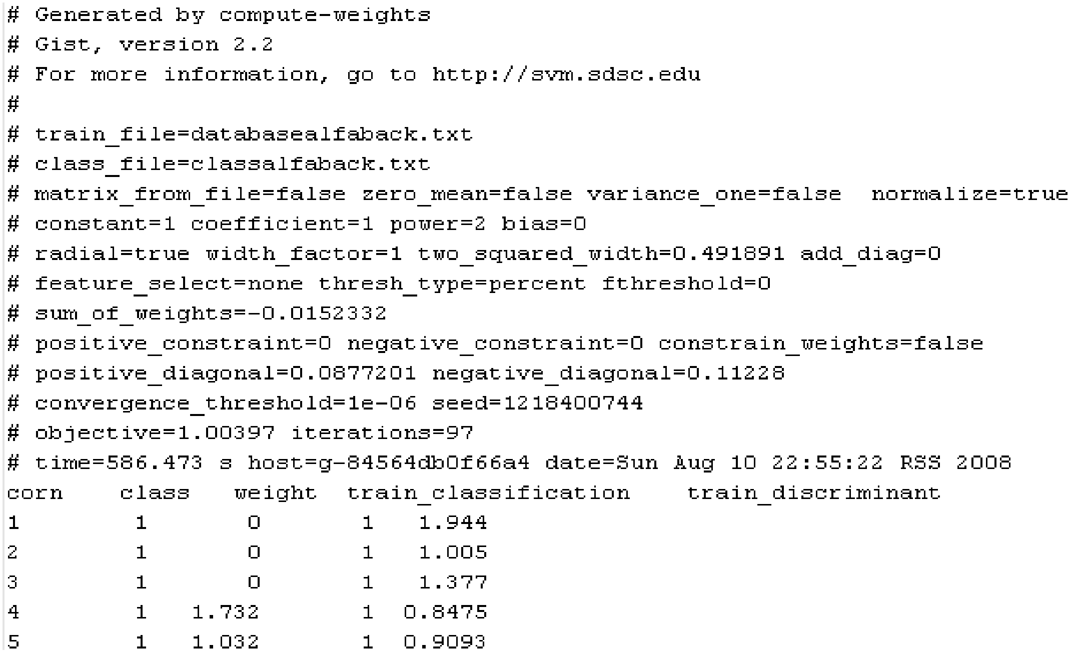
As figure 4.5 above, we use 19x19x7 features for 1471 face and 2500 non-face patterns. Then a database of face and non-face patterns, assigned to classes +1 and -1 respectively (Figure 4.6), is used to train a SVM with a 2nd-degree polynomial as kernel function.



corn	class
1	1
2	1
3	1
4	1
5	1
2120	-1
2121	-1
2122	-1
2123	-1
2124	-1

Figure 4:6 Example of our class file for SVM

From these files, we generate model file of face and non-face database that contains a weight associated with each training set. In addition to the weights, the file contains the classification label, a predicted classification label (which indicates which side of the hyperplane the example lies on), and a discriminant value (which is proportional to the distance between the example and the hyperplane).



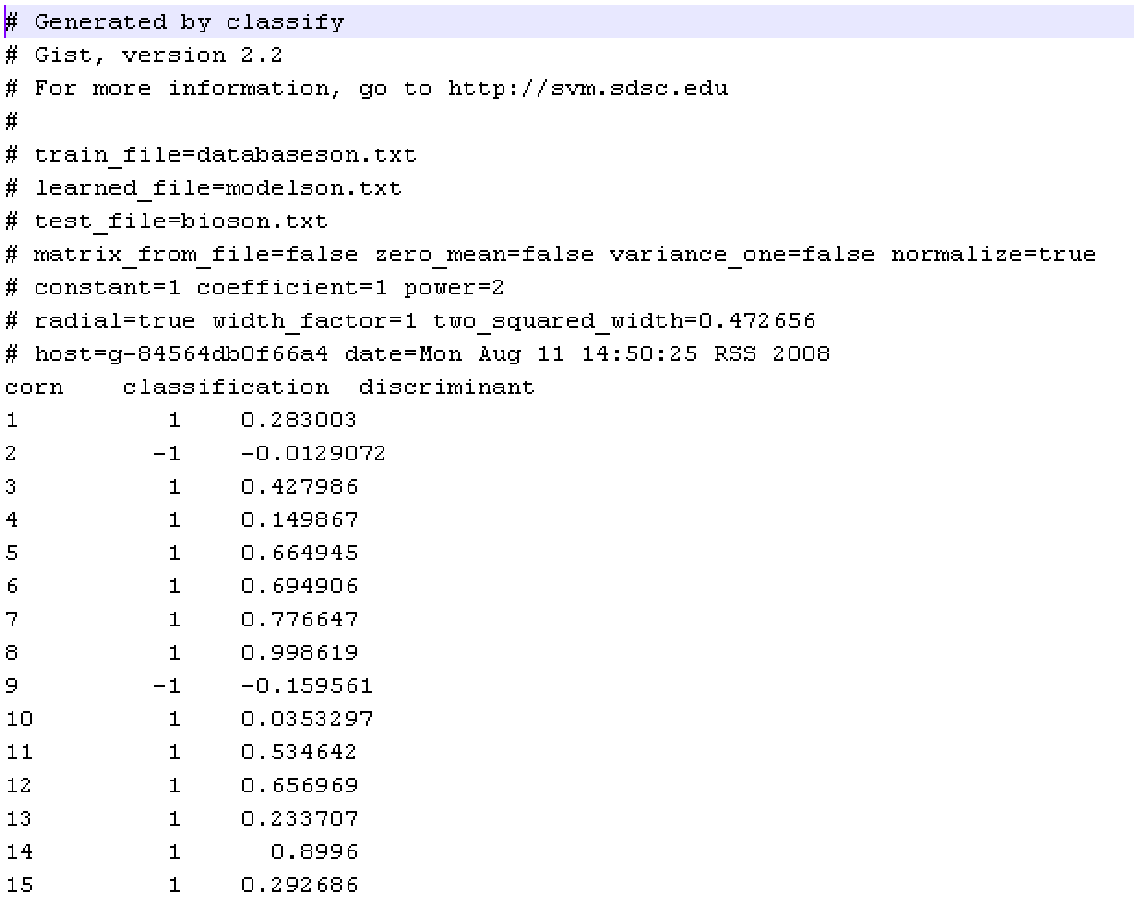
```
# Generated by compute-weights
# Gist, version 2.2
# For more information, go to http://svm.sdsc.edu
#
# train_file=databasealfaback.txt
# class_file=classalfaback.txt
# matrix_from_file=false zero_mean=false variance_one=false  normalize=true
# constant=1 coefficient=1 power=2 bias=0
# radial=true width_factor=1 two_squared_width=0.491891 add_diag=0
# feature_select=none thresh_type=percent fthreshold=0
# sum_of_weights=-0.0152332
# positive_constraint=0 negative_constraint=0 constrain_weights=false
# positive_diagonal=0.0877201 negative_diagonal=0.11228
# convergence_threshold=1e-06 seed=1218400744
# objective=1.00397 iterations=97
# time=586.473 s host=g-84564db0f66a4 date=Sun Aug 10 22:55:22 RSS 2008
corn    class   weight  train_classification    train_discriminant
1        1       0       1   1.944
2        1       0       1   1.005
3        1       0       1   1.377
4        1   1.732       1  0.8475
5        1   1.032       1  0.9093
```

Figure 4:7 Output of our model file

Training

To evaluate the performance of our system we use 520 different face images from BioID database and 1000 non-face images from CMU database. Images sizes normalized to 19x19 pixels.

For face images we get 119 false result based on 520 face images. A detection rate 78% is achieved. For non-face images we use 1000 images patterns from CMU database and we get true result for all patterns.



```
# Generated by classify
# Gist, version 2.2
# For more information, go to http://svm.sdsc.edu
#
# train_file=databaseson.txt
# learned_file=modelson.txt
# test_file=bioson.txt
# matrix_from_file=false zero_mean=false variance_one=false normalize=true
# constant=1 coefficient=1 power=2
# radial=true width_factor=1 two_squared_width=0.472656
# host=g-84564db0f66a4 date=Mon Aug 11 14:50:25 RSS 2008
corn    classification  discriminant
1          1    0.283003
2         -1    -0.0129072
3          1    0.427986
4          1    0.149867
5          1    0.664945
6          1    0.694906
7          1    0.776647
8          1    0.998619
9         -1    -0.159561
10         1    0.0353297
11         1    0.534642
12         1    0.656969
13         1    0.233707
14         1      0.8996
15         1    0.292686
```

Figure 4:8 Output of classifier for face patterns from BioID database

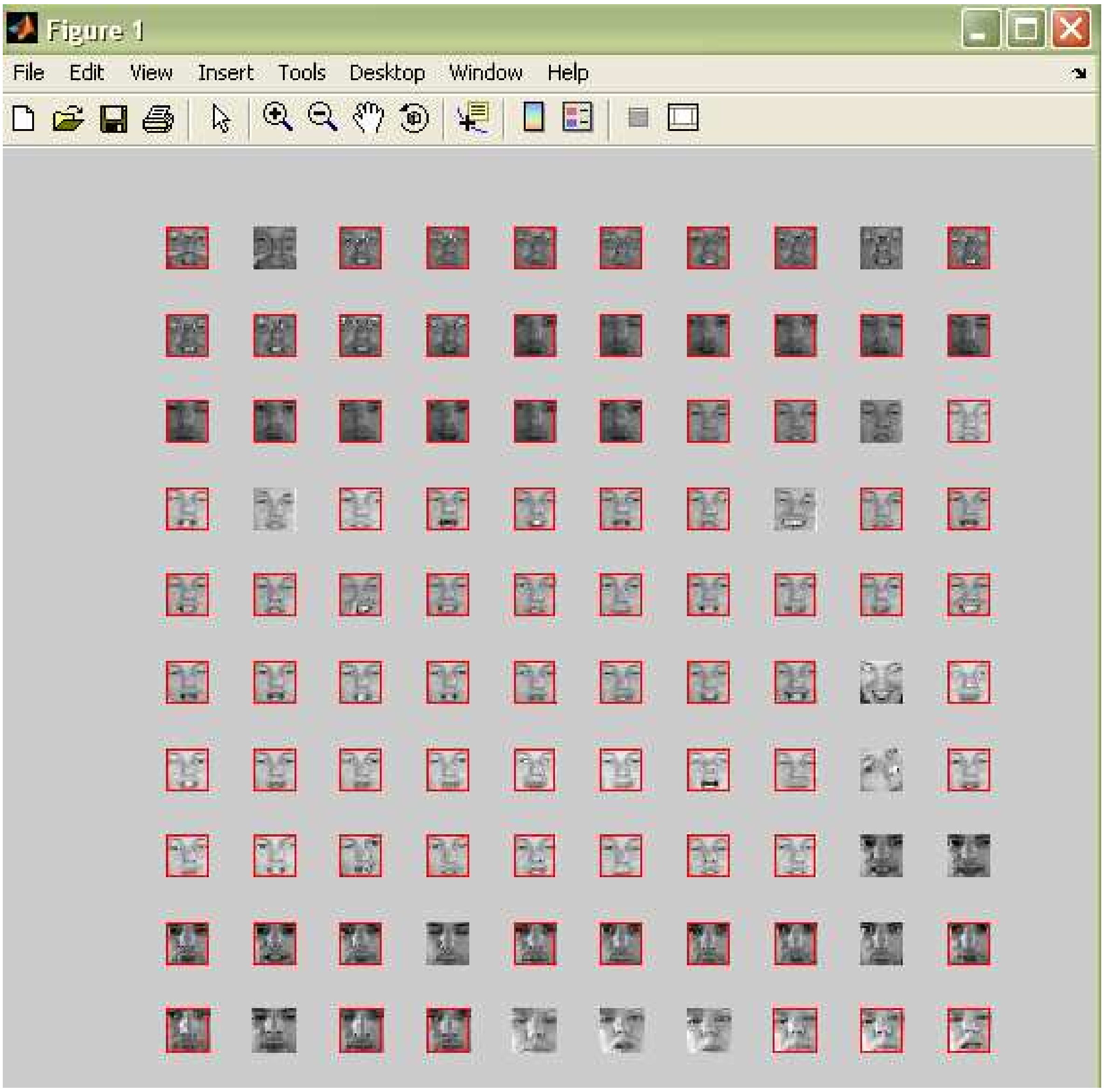


Figure 4:9 Result of 100 face images from BioID database

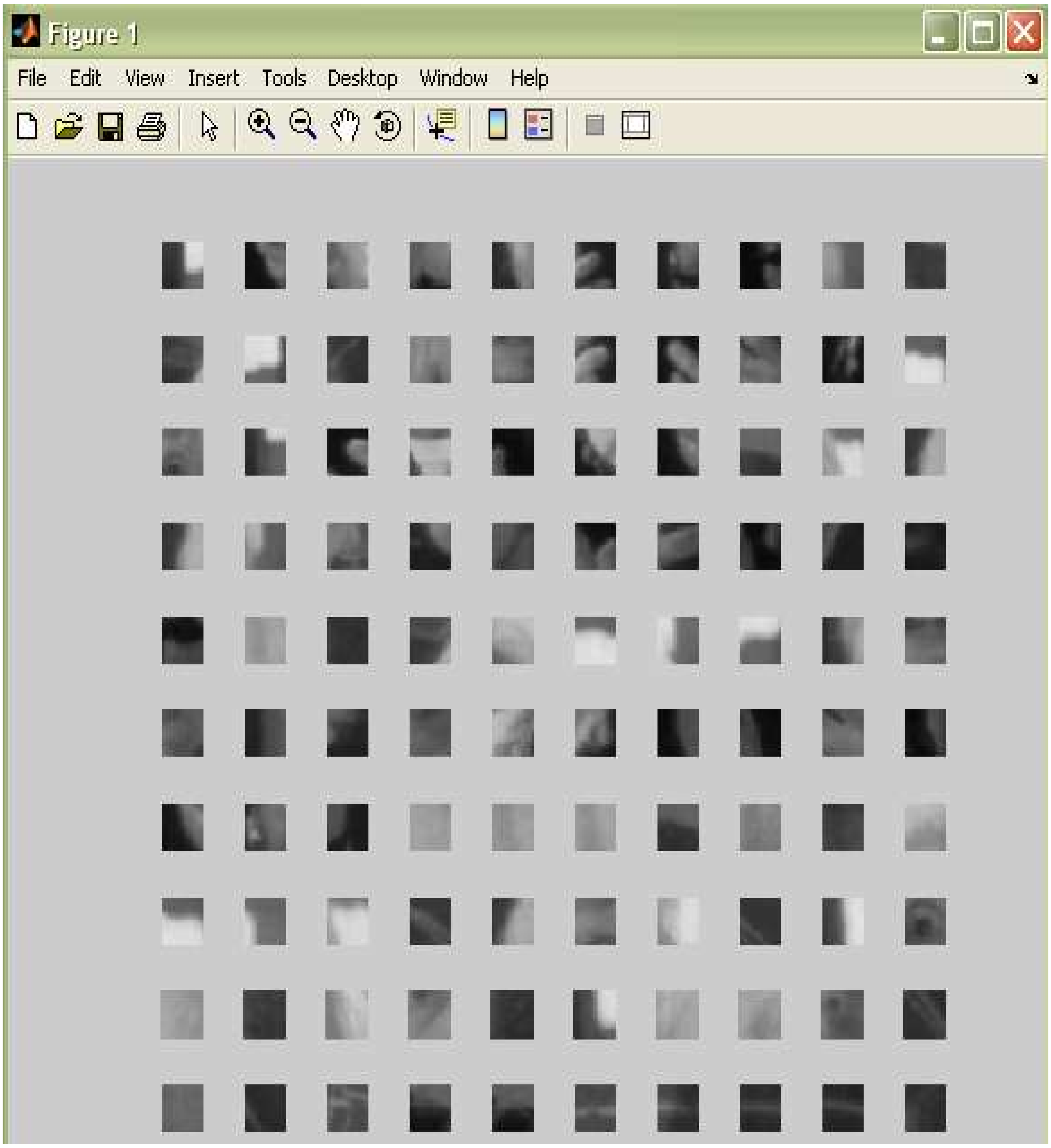


Figure 4:10 Result of 100 non-face images from CMU database

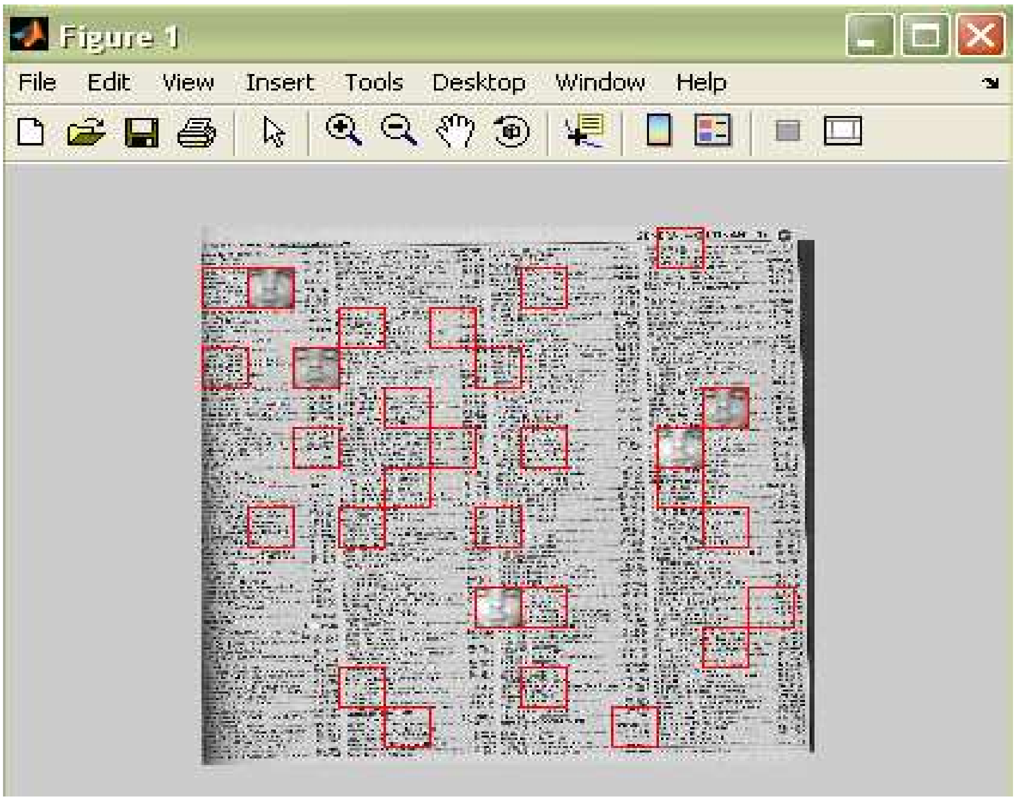


Figure 4:11 Example of our results

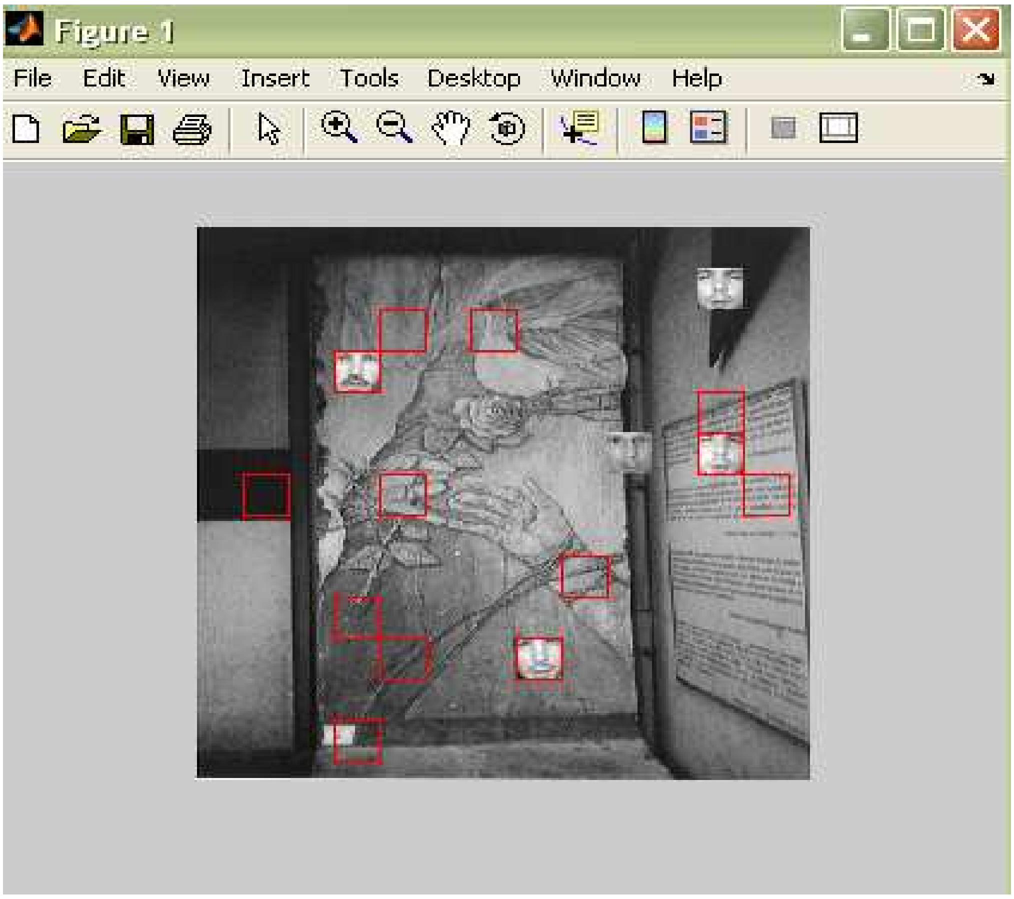


Figure 4:12 Example of our results

CHAPTER 5

5 CONCLUSIONS AND RECOMMENDATIONS

In this thesis, a new approach to face detection with Gabor features is presented. The method uses Gabor filters for extracting the feature vectors. From the experimental results we get 119 false results based on 520 face images and a detection rate 78% is achieved. It can be seen that proposed method achieves good results. In the proposed algorithm, the facial features are compared using a general structure. Proposed method is also robust to illumination changes mainly attributed to the property of Gabor features, which is the main problem with the eigenface approaches.

PCA can significantly reduce the dimensionality of the original feature without loss of much information in the sense of representation, but it may lose important information for discrimination between different classes. So for increase performance in future works additional methods such as combination of PCA and LDA can be applied.

Furthermore, experimental results in the training phase showed that SVM face classifier is able to separate examples included in face and non-face training data from each other. We used free parameters of SVM that are the positive constant C(1) that controls the trade off between margin and misclassification error and the parameter associated to the kernel (the degree of the polynomial) (power 2).

REFERENCES

[1] Ming Hsuan Yang, David J. Kriegman, and Narendra Ahuja. *Detecting Faces in Images: A Survey*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 1, pp. 34–58, January 2002.

[2] Ming-Hsuan Yang Ahuja, N. *Detecting Human Faces in Color Images*, Proc. of the 1st IEEE Conf. on Image Processing, vol. 1, pp. 127-130, 4-7 Oct 1998

[3] G. Yang and T. S. Huang, Human Face Detection in A Complex Background, Pattern Recognition, vol. 27, no. 1, pp. 53-63, 1994.

[4] H. Schneiderman and T. Kanade. A Statistical Method for Object Detection Applied to Faces and Cars. In International Conference on Computer Vision and Pattern Recognition, pp. 1746-1759, 2000.

[5] H. A. Rowley, S. Baluja, and T. Kanade, Neural Network-Based Face Detection, IEEE Trans. Pattern Analysis Machine Intelligence, vol. 20 no. 1, pp. 23-38, January 1998

[6] Hong-Bo Deng, Lian-Wen Jin, Li-Xin Zhen, Jian-Cheng Huang, A New Facial Expression Recognition Method Based on Local Gabor Filter Bank and Pca Plus LDA, International Journal of Information Technology, vol. 11, no. 11, pp. 86-96, 2005

[7] C. Kotropoulos and I. Pitas, Rule-Based Face Detection in Frontal Views, Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 97), vol. 4, pp.2537-2540, 1997

[8] S. Pigeon and L. Vandendrope, The M2VTS Multimodal Face Database, Proceedings of the First International Conference on Audio- and Video-Based Biometric Person Authentication, vol. 1206, pp. 403-409, 1997

[9] S.A. Sirohey, Human Face Segmentation and Identification, Technical Report CS-TR-3176, Univ. of Maryland, pp. 1-33,1993.

[10] J. Canny, A Computational Approach to Edge Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 8, no. 6, pp. 679-698, June 1986.

[11] D. Chetverikov and A. Lerch, Multiresolution Face Detection, Theoretical Foundations of Computer Vision, vol. 69, pp. 131-140, 1993.

[12] H.P. Graf, T. Chen, E. Petajan, and E. Cosatto, Locating Faces and Facial Parts, Proc. First Int’l Workshop Automatic Face and Gesture Recognition, pp. 41-46, 1995.

[13] T.K. Leung, M.C. Burl, and P. Perona, Finding Faces in Cluttered Scenes Using Random Labeled Graph Matching, Proc. Fifth IEEE Int’l Conf. Computer Vision, pp. 637-644, 1995.

[14] P. Sinha, Object Recognition Via Image Invariants: A Case Study, Investigative Ophthalmology and Visual Science, vol. 35, no. 4, pp. 1735-1740, 1994

[15] C.-C. Han, H.-Y. M. Liao, K.-C. Yu, and L.-H. Chen, Fast Face Detection Via Morphological-Based Pre-Processing, in Proceedings of the Ninth International Conference on Image Analysis and Processing, pp. 469-476, 1998

[16] J.R. Quinlan, C4. 5: Programs for Machine Learning, 1993.

[17] M.F. Augusteijn and T.L. Skujca, Identification of Human Faces through Texture-Based Feature Recognition and Neural Network Technology, Proc. IEEE Conf. Neural Networks, pp. 392-398, 1993.

[18] R.M. Haralick, K. Shanmugam, and I. Dinstein, Texture Features for Image Classification, IEEE Trans. Systems, Man, and Cybernetics, vol. 3, no. 6, pp. 610-621, 1973.

[19] S. Fahlman and C. Lebiere, The Cascade-Correlation Learning Architecture, Advances in Neural Information Processing Systems 2, pp. 524-532, 1990.

[20] T. Kohonen, Self-Organization and Associative Memory, 1989.

[21] Y. Dai and Y. Nakano, Face-Texture Model Based on SGLD and Its Application in Face Detection in a Color Scene, Pattern Recognition, vol. 29, no. 6, pp. 1007-1017, 1996.

[22] T.S. Jebara and A. Pentland, Parameterized Structure from Motion for 3D Adaptive Feedback Tracking of Faces, Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 144-150, 1997.

[23] T.S. Jebara, K. Russell, and A. Pentland, Mixtures of Eigenfeatures for Real-Time Structure from Texture, Proc. Sixth IEEE Int’l Conf. Computer Vision, pp. 128-135, 1998.

[24] Y. Miyake, H. Saitoh, H. Yaguchi, and N. Tsukada, Facial Pattern Detection and Color Correction from Television Picture for Newspaper Printing, Journal of Imaging Technology, vol. 16, no. 5, pp. 165-169, 1990.

[25] J.L. Crowley and J.M. Bedrune, Integration and Control of Reactive Visual Processes, Proc. Third European Conf. Computer Vision, vol. 2, pp. 47-58, 1994.

[26] J.L. Crowley and F. Berard, Multi-Modal Tracking of Faces for Video Communications, Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 640-645, 1997

[27] D. Saxe and R. Foulds, Toward Robust Skin Identification in Video Images, Proc. Second Int’l Conf. Automatic Face and Gesture Recognition, pp. 379-384, 1996

[28] J. Sobottka and I. Pitas, Segmentation and Tracking of Faces in Color Images, Proc. Second Int’l Conf. Automatic Face and Gesture Recognition, pp. 236-241, 1996.

[29] K. Sobottka and I. Pitas, Face Localization and Feature Extraction Based on Shape and Color Information, Proc. IEEE Int’l Conf. Image Processing, pp. 483-486, 1996.

[30] Y. Dai and Y. Nakano, Extraction for Facial Images from Complex Background Using Color Information and SGLD Matrices, Proc. First Int’l Workshop Automatic Face and Gesture Recognition, pp. 238-242, 1995.

[31] Jie Chen, Shiguang Shan, Peng Yang, Shengye Yan, Xilin Chen and Wen Gao1, Novel Face Detection Method Based on Gabor Features, Sınobıometrıcs 2004 : Chinese conference on biometric recognition, vol. 3338, pp. 90-99, 2004

[32] D. Chai and K.N. Ngan, Locating Facial Region of a Head-and-Shoulders in Color Image, Proc. 3rd Int’l Conf. Automatic Face and Gesture Recognition, pp. 124-129, 1998.

[33] H.Wang andS.-F.Chang, A Highly Efficient System for Automatic Face Region Detection in MPEG Video, IEEE Trans. Circuits and Systems for Video Technology, vol. 7, no. 4, pp. 615-628, 1997.

[34] E. Saber and A.M. Tekalp, Frontal-View Face Detection and Facial Feature Extraction Using Color, Shape and Symmetry Based Cost Functions, Pattern Recognition Letters, vol. 17, no. 8, pp. 669-680, 1998.

[35] Q. Chen, H. Wu, and M. Yachida, Face Detection by Fuzzy Matching, Proc. Fifth IEEE Int’l Conf. Computer Vision, pp. 591-596, 1995.

[36] F. Riesz and B. Sz.-Nagy. Functional Analysis. , 1955.

[37] K. Sobottka and I. Pitas, Face Localization and Feature Extraction Based on Shape and Color Information, Proc. IEEE Int’l Conf. Image Processing, pp. 483-486, 1996

[38] J.C. Terrillon, M. David, and S. Akamatsu, Automatic Detection of Human Faces in Natural Scene Images by Use of a Skin Color Model and Invariant Moments, Proc. Third Int’l Conf. Automatic Face and Gesture Recognition, pp. 112-117, 1998.

[39] J.C. Terrillon, M. David, and S. Akamatsu, Detection of Human Faces in Complex Scene Images by Use of a Skin Color Model and Invariant Fourier-Mellin Moments, Proc. Int’l Conf. Pattern Recognition, pp. 1350-1355, 1998.

[40] S.-H. Kim, N.-K. Kim, S.C. Ahn, and H.-G. Kim, Object Oriented Face Detection Using Range and Color Information, Proc. Third Int’l Conf. Automatic Face and Gesture Recognition, pp. 76-81, 1998

[41] T. Sakai, M. Nagao, and S. Fujibayashi, Line Extraction and Pattern Detection in a Photograph, Pattern Recognition, vol. 1, pp. 233-248, 1969.

[42] A. Yuille, P. Hallinan, and D. Cohen, Feature Extraction from Faces Using Deformable Templates, Int’l J. Computer Vision, vol. 8, no. 2, pp. 99-111, 1992.

[43] M. Turk and A. Pentland, Eigenfaces for Recognition, J. Cognitive Neuroscience, vol. 3, no. 1, pp. 71-86, 1991.

[44] K.-K. Sung, Learning and Example Selection for Object and Pattern Detection, PhD thesis, Massachusetts Inst. of Technology, 1996.

[45] K.-K. Sung and T. Poggio, Example-Based Learning for View-Based Human Face Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 20, no. 1, pp. 39-51, Jan. 1998.

[46] M.-H. Yang, N. Ahuja, and D. Kriegman, Face detection using mixtures of linear subspaces, Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition, 2000.

[47] R. O. Duda, P. E. Hart, D. G. Stork, Pattern Classification, 2001

[48] D. Roth, Learning to Resolve Natural Language Ambiguities: A Unified Approach, Proc. 15th Nat'l Conf. Artificial Intelligence, pp. 806-813, 1998.

[49] A. Carleson, C. Cumby, J. Rosen, and D. Roth, The SNoW Learning Architecture, Technical Report UIUCDCS-R-99-2101, Univ. of Illinois at Urbana-Champaign Computer Science Dept., 1999.

[50] M.-H. Yang, D. Roth, and N. Ahuja, A SNoW-Based Face Detector, Advances in Neural Information Processing Systems 12, pp. 855-861, 2000

[51] N. Littlestone, Learning Quickly when Irrelevant Attributes Abound: A New Linear-Threshold Algorithm, Machine Learning, vol. 2, pp. 285-318, 1988

[52] F.S. Samaria, Face Recognition Using Hidden Markov Models, PhD thesis, Univ. of Cambridge, 1994

[53] D.B. Graham and N.M. Allinson, Characterizing Virtual Eigensignatures for General Purpose Face Recognition, Face Recognition: From Theory to Applications, vol. 163, pp. 446-456, 1998

[54] P. Hallinan, A Deformable Model for Face Recognition Under Arbitrary Lighting Conditions, PhD thesis, Harvard Univ., 1995

[55] P. Belhumeur, J. Hespanha, and D. Kriegman, Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 711-720, 1997.

[56] P.J. Phillips, H. Moon, S.A. Rizvi, and P.J. Rauss, The FERET Evaluation Methodology for Face-Recognition Algorithms, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22, no. 10, pp. 1090-1034, Oct. 2000

[57] H. Schneiderman and T. Kanade, Probabilistic Modeling of Local Appearance and Spatial Relationships for Object Recognition, Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 45-51, 1998.

[58] T. Rikert, M. Jones, and P. Viola, A Cluster-Based Statistical Model for Object Detection, Proc. Seventh IEEE Int'l Conf. Computer Vision, vol. 2, pp. 1046-1053, 1999

[59] J. Huang, S. Gutta, and H. Wechsler, Detection of Human Faces Using Decision Trees, Proc. Second Int'l Conf. Automatic Face and Gesture Recognition, pp. 248-252, 1996

[60] Hong-Bo Deng, Lian-Wen Jin, Li-Xin Zhen, Jian-Cheng Huang, A New Facial Expression Recognition Method Based on Local Gabor Filter Bank and PCA plus LDA

[61] Edgar Osuna and Robert Freund and Federico Girosi, Training Support Vector Machines: An Application to Face Detection, pp. 130-136, 1997

[62] P. N. Bellhumer, J. Hespanha, and D. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence, Special Issue on Face Recognition, Vol. 17, No. 7, pp. 711--720, 1997

[63] Georghiades, A.S. and Belhumeur, P.N. and Kriegman, D.J. From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose. IEEE Trans. Pattern Anal. Mach. Intelligence Vol. 23, No. 6, pp. 643-660, 2001

[64] T. Sim, S. Baker, and M. The CMU Pose, Illumination, and Expression Database IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25, No. 12, pp. 1615–1618, December 2003

[65] F. Samaria and A. Harter. Parameterisation of a stochastic model for human face identification. In 2nd IEEE Workshop on Applications of Computer Vision, Sarasota, FL, 1994.

[66] Takeo Kanade, Comprehensive database for facial expression analysis. pp. 46-53. 2000

[67] Ferdinando Samaria, Andy Harter, Parameterisation of a Stochastic Model for Human Face Identification, Proceedings of 2nd IEEE Workshop on Applications of Computer Vision, December 1994

[68] B. Weyrauch, J. Huang, B. Heisele, and V. Blanz, Component-based Face Recognition with 3D Morphable Models, First IEEE Workshop on Face Processing in Video, 2004.