A Generic Face Processing Framework: Technologies, Analyses and Applications

JANG Kim-fung

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Master of Philosophy
in
Computer Science and Engineering

Supervised by

Prof. Michael R. Lyu

July 2003

Abstract

Face processing system is a vast and modern research area of computer vision and pattern recognition. This is a challenging task for integrating various techniques into an application. A detailed survey about facial feature techniques is presented. We have classified the facial features and the extraction methods into categories. The performance of face recognition methods are summarized that help researcher to have a brief introduction about current techniques and the evolution of face recognition methods. The presence of the face is important for face processing system. We propose a Face Detection Committee Machine (FDCM) to increase the accuracy of correct classification for faces. The machine combines three face detection methods that are Sparse Network of Winnows algorithm, Support Vector Machines, and Neural networks. An evaluation among the three individual approaches and FDCM are presented using the same set of training images (6,977) and testing images (24,045) in the experimental result. The result shows that the false alarm rate of FDCM (7.79%) is nearly half of the best approach (12.32%) among other three approaches. We propose two facial feature localization algorithms that based on separability filter. The separability filter is shown to be useful in improving the accuracy of eye localization. The localization algorithm for color image is presented to normalize the variation of a face image. We also develop a face processing system to demonstrate the integration of various techniques for generic purpose.

論文摘要

面孔處理系統是電腦視覺和模式識別的一個浩大和近代研究區域。將有關技術集成爲應用系統是一項富挑戰性的任務。我們提出一份關於面部特徵的詳細調查。我們首先把面部特徵和特徵的提取方法分類。報告總結了各面孔識別方法的表現，這幫助研究員對關於當前的技術和面孔識別方法的演變有一簡要介紹。面孔的存在與否對面孔處理系統是重要是的。我們提出面孔偵查委員會(FDCM)機去增加面孔分類的準確性。機器結合了三種面孔偵測方法，分別爲Sparse Network of Winnows 算法、支援向量機和神經網路。在實驗中，我們使用同樣的訓練圖像(6977)與測試圖像(24045)作出三種個別的方法和 FDCM 的評估。結果顯示， FDCM 的誤判率(7.79%) 約爲三種方法中最佳誤判率(12.32%)的一半。我們亦提出兩種根據分群過濾器的面部特徵定位算法。結果證明分群過濾器能有效改進眼睛位置的準確性。在彩色圖像上的特徵定位算法能把面孔的差異作標準處理。我們亦開發一個作爲概括目的面孔處理系統去支持我們這份論文的主旨。

Acknowledgement

I would like to take this opportunity to express my gratitude to my supervisor, Prof. Michael, R. Lyu for his generous guidance and patience given to me in my study. I am also grateful for the time and valuable suggestions that Prof. Michael, R. Lyu and Prof. Irwin King have given support and advice which are essential and valuable in my research paper (published in CAIP'2003) and my thesis.

I would like to thank my research partner, Sunny Tang, for the co-operative work and sharing of research ideas. Special thanks should be given to my fellow colleagues, Nicky Ng, Jacky Ma, M.L. Ho, C.L. Fung, Joe Lau, K.C. Chan who have helped me in solving programming problems, discussion about the research work, and have given me a wonderful university life.

Finally, I am deeply indebted to my parents for their unconditional love and support over the years.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

In the modern information age, human's information becomes valuable. Retrieval information from the video source or camera becomes many researchers' interest, especially for human face. A system can retrieve the face information, we call face processing system. Rich contents can be extracted by a face processing system. That's why it can have a wide range of usage and becomes a hot topic in recent years. It can be used for the security and social issues, multimedia database, and intelligent vision-based human computer interaction.

For the security issue, identification and authentication methods have developed into a main technology in various areas, such as entrance control in building and access control for computers. Most of these methods have a drawback with their legitimate applications. Except for the human and voice recognition, these methods almost require user to remember a password, or human action in the course of identification or authentication. However, the corresponding means are potentially being lost or forgotten, whereas fingerprints and retina scans suffer from low user acceptance rate. Face recognition has a high identification and recognition rate of greater than 90% for huge face databases with well-controlled pose and illumination conditions. This high rate can be used for replacement of lower security requirement

environment and can be successfully employed in different kind of issues such as multimedia. Video is a kind of multimedia source that contains huge information. To create a digital library, useful information like actors' name in a movie, must be retrieved and analyzed before stored in the database. Video indexing and summarization can be achieved by performing face processing. Besides the usage for security issue and multimedia issue, human computer interaction is another important system that communicates with human not only relies on mice and keyboards. New information technology and media can be given in a more effective and friendly method. A face processing system can provide more human-computer interactive services to improve the life style [53].

Automatic face processing system is a vast and modern research area of computer vision and pattern recognition, reaching from face detection, face tracking, extraction of facial features, face recognition and facial expressions. The first step of any face processing system is to detect the presence of a face and the location of the face in images. Face detection is to determine an image that contains a human face or not. Face localization is to locate the position and size of the face. In an image sequence, the movement of the head is captured by the face tracking instead of performing face detection for each image. The execution time for the tracking method is shorter than those for the face detection because the positions of face are strongly depend on the adjacent image in an image sequence. Extraction of facial feature is one of the important parts in face processing; the aim of the extraction is to find out the most discriminated feature from the face. These features can be part of human knowledge about the face, such as the positions of eyes and nose, or set of coefficients that represent a face. Several techniques apply the extracted feature for higher level classification such as face recognition and expression recognition.

Human communicate with the computer that can give an appropriate feedback to you. This ultimate goal for a human computer interaction system is extremely difficult to achieve. Some possible problems for the area have been raised by [27], [99]. In the past ten years, researchers had focused their efforts on various face processing field that include face detection, face tracking, face recognition, and expression recognition. Human computer interaction is being developed using different techniques. With the help of the evolution of technology and media, the development of computer vision systems in desktop and embedded systems becomes true [100], [79], [99], [128], [53], [56]. However, the functionality of the systems is limited and works under some specify conditions.

In this thesis, we investigate various techniques for face processing that involve in a basic framework . Lots of research work has focused on face detection, face tracking, face recognition and expression recognition. These problems are related to find out features of human face in images or image sequences. Analysis of facial feature is presented for understanding the current literature and future research direction. There are two face recognition methods, Eigenface [122] and Elastic Bunch Graph Matching [126], that give a great impact in the literature. A comparative study on these two methods helps to explain the reason for the impact. The presence of the face is important for face processing system. After the study of face detection methods, we find that using different training and testing data for the evaluation may varies the results. An experimental comparison is needed for further studies on the face pattern classification methods. Also, a possible way to improve the accuracy of classification of face pattern is presented, that is the committee machine based detection approach. Two facial feature localization algorithms are proposed for gray-scale image and color image. The algorithm used for the color image can be employed for

the image normalization. A basic face processing framework is provided for the development of a system. The framework is composed of various techniques. This is a challenging task for integrating those techniques into an application because there are limitations on the development of a face processing system.

1.2 Introduction about Face Processing Framework

Face processing systems have different purposes of usages, such as security, intelligent vision-based human computer interaction and multimedia database system. Different kinds of information are extracted from the sources in the system such as the identity of a person from the security system, the gesture of a person and the identity of an actor in a movie. All the above face processing system can be developed from a basic face processing framework. By adding different components, the basic framework can provide a suitable functionality for the system. In this section, an introduction of basic face processing framework is given. Three important modules are needed in the framework, i.e. face detection module, face tracking module and face recognition module.

1.2.1 Basic architecture

Although different face processing systems have various abilities, several basic procedures are applied to the system for information retrieval. The aim of the framework is to extract the information about human face including the location of the face, the size of the face and the identity of the human. There are two main types of systems: online system and offline system. An online system is a time critical machine that the procedures should be processed within a fixed time interval, thus, only the

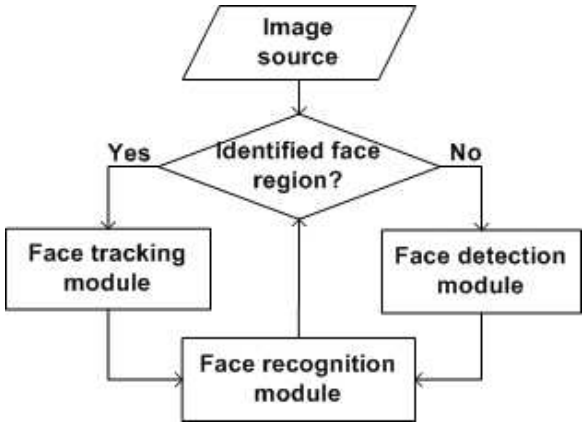

Figure 1.1: The basic framework for face processing

most important information is extracted. An offline system can have less limitation on the execution time for extracting the features, more detailed and accurate information can be retrieved for further usage. We can divide the face processing procedure into three main modules, 1) face detection module, 2) face tracking module, and 3) face recognition module. Figure 1.1 shows the basic architecture of the framework that is designed for a sequence of images. For a still image, the face tracking module can be omitted in the framework.

1.2.2 Face detection

The presence of face is one of the key factors for a face processing system. Face can appear in different conditions in an image such as user sitting in front of a web camera, passenger walking along a corridor and reporter reporting news in a roundup. Color image or gray-scale image can be the source of the system. A sequence of images provide continuously for an interactive system. We treat these images as the input of the face detection module. After processing, there are various outputs such as the number of faces, position, size, pose and orientation about the

detected faces. A normalization step can be performed using the output parameters before the face recognition module.

1.2.3 Face tracking

Tracking an object from a sequence of images is important for face processing. After the successful detection for a human face, the region of the face is processed by the face tracking module instead of the face detection module. In an image sequence, the positions of the face are strongly related to the adjacent images. Researchers have developed tracking methods based on this assumption. The initial state of the face is provided for tracking, and the restriction is held on the position and size of the face. This helps to reduce the search space for the new position in the adjacent image, and the execution time for tracking method can be shorter than those for detection. The outputs of the face tracking module are similar to those of the face detection module, except that the motion of the object can be provided by the tracking method.

1.2.4 Face recognition

After the region of the face is extracted, face recognition is applied to classify the identity of the person. Different systems employ this module to provide various functions. Security system can grant the permission to the user when the authentication result is matched with the stored profile. Video summarization system can generate a set of actor's information for multimedia indexing. The face recognition can change to other recognition method to provide more information such as expression recognition. Human computer interaction can be achieved when the human gesture classification module is employed.

1.3 The scope and contributions of the thesis

The main statement of this thesis is: knowledge of processing face can be used to develop a generic face processing system. Faces were selected as the objects to be studied because they are important and informative in many applications. This thesis employs three different phases: 1) survey about techniques applied in face processing, 2) make enhancement on current techniques, and 3) apply the obtained knowledge to develop a face processing system. The following list shows the contributions in our research work:

- A detailed survey about facial feature is presented. We classify the facial features and the feature extraction methods into categories. A summary of the performance of different face recognition methods is given. This survey helps researcher to have a brief introduction about current techniques and the evolution of face recognition methods.
- A Face Detection Committee Machine (FDCM) is introduced to enhance the accuracy of correct face classification (Paper I[1]). The machine combines three face detection approaches that are Sparse Network of Winnows (SNoW) algorithm, Support Vector Machines (SVM) and Neural networks (NN). An evaluation among the three individual approaches and FDCM is presented using experimental result.
- Two ad-hoc facial feature localization algorithms are introduced. The algorithm for gray-scale image is based on template matching and separability filter to find the position of

[1]K. F. Jang, H. M. Tang, M. R. Lyu and I. King. A Face Processing System Based on Committee Machine: The Approach and Experimental Results. Proc. of 10th International Conference on Computer Analysis of Images and Patterns (CAIP'2003), Groningen, The Netherlands. The co-author H.M. Tang participated in the writing process and the research work of face recognition committee machine. The other authors gave their useful comments.

irises and mouth. This algorithm shows that separability filter is useful in improving the accuracy of iris localization. A modified localization algorithm is proposed for color image, that is employed to normalize the face image.

- A face processing system (Paper I see footnote 1) is developed for generic purpose. Various face processing techniques are included in the system. Here it is shown that the knowledge obtained about face processing can be used to develop a face processing system. The system is success in detecting, tracking and recognizing human faces.

1.4 The outline of the thesis

The remaining chapters of this thesis are organized as follows:

Chapter 2 gives a survey about facial features that are important in the development of face processing system. We summarize and analyze the extraction techniques and the representation methods in various face processing fields. Two state-of-the-art face recognition methods, Eigenface and Elastic Bunch Graph matching are evaluated for explaining the impact.

Next, in Chapter 3, we investigate some of the face detection techniques, especially SNoW algorithm, SVM and NN. We also propose a Face Detection Committee Machine, FDCM, using the above three approaches. We combine three face detection approaches to form an expert system to improve the accuracy of classifying faces. We present the experimental results of FDCM using a data set of about 30,000 images and have a comparison among three different approaches under controlled condition. The evaluation shows that our machine performs better than the other three individual algorithms in terms of detection rate, false-alarm rate and ROC curve.

Then, two ad-hoc algorithms are proposed for facial feature

localization in Chapter 4. Template matching technique and separability filter are applied to locate the features in gray-scale image. Facial features include the position of eye region, irises and lip. Eye region is first located to reduce the search space for locating the eyes and the separability filter is applied to increase the accuracy of locating eyes. Besides, an image normalization algorithm for color image is proposed, the separability filter is applied to construct a sepmap. This sepmap is then combined with the color information to increase the accuracy of choosing eye candidates. Moreover, the location of the eyes can be found inside a non-upright face image without the predefined size. Experimental results are given to show that the separability filter can help to increase the accuracy of locating iris and the normalization algorithm can reduce the variation of the image such as rotation and scale.

We have developed a face processing system based on the generic framework, and is presented in Chapter 5. The system architecture is given in the first part of the chapter. Afterward, detailed description of each module and the techniques being applied are discussed.

Finally, conclusions are drawn about the work in Chapter 6.

□ **End of chapter.**

Chapter 2

Facial Feature Representation

Useful data are extracted from the images or image sequences to be further processed. We call these data as feature. Image analysis and pattern recognition take an important role of finding facial feature. In this chapter, we aim to summarize and analyze different facial features used in face processing regarding face recognition. We group the facial features into two main categories - pixel information and geometry information. We survey the techniques presented in various face processing fields and have a discussion about current techniques. We do not attempt to perform an exhaustive review in each field but most the-state-of-the-art approaches. Discussion about the facial feature and the development of face processing system is given.

2.1 Facial feature analysis

Although face processing technology contains many different areas, we can identify it as three basic problems. The three problems are: 1) finding out the feature, 2) coding of the feature, and 3) classifying the coded feature.

What features are useful in face processing? Before analyzing the facial feature being used in various face processing fields, we have summarized the facial feature that had been used in past work into two main categories: Geometry information and Pixel information. We do not attempt to perform an exhaustive search

Table 2.1: Category of facial feature - Pixel

		Features	Reference
Pixel	Global	Eigenface	Turk [122]
		Discrete Cosine Transform (DCT)	Eickeler [33]
		DCT-mod2	Sanderson [107]
		Discriminant Karhunen-Loeve projection (DKL)	Swets [116]
		Fisher linear discriminant (FLD)	Belhumeur [7]
		Enhanced Fisher linear discriminant (EFC)	Liu [77]
		Autocorrelation coefficients	Goudail [44]
		Direct LDA (D-LDA)	Yu [132]
		Direct fractional-step LDA (DF-LDA)	Lu [82]
		Log transform	Adini [2]
		Gabor + Enhanced FLD (GFC)	Liu [78]
		Independent component analysis (ICA)	Bartlett [5]
		Kernel PCA (KPCA)	Kim [63]
		Kernel direct discriminant analysis (KDDA)	Lu [81]
	Local	Elastic bunch graph	Wiskott [126]
		Eigennose, Eigeneye, Eigenmouth or Eigeneyebrow	Su [115]
	Color	PCA in each color channel	Torres [121]

for all the related papers to find out all the facial features, we focus on finding the features in most popular papers or the-state-of-the-art in the literature. Table 2.1 and Table 2.2 show the facial features found, and we only show one of the reference papers in the last column of the table to simplify the table.

2.1.1 Pixel information

A face photo or face sequences are represented by digital signal in computer. These signals are the prime information of facial

Table 2.2: Category of facial feature - Geometry

		Features	Reference
Geometry	Position	Iris center	Tian [119]
		Mouth position	Tian [119]
		Cheek triangular position	Tian [119]
		Nose position	Brunelli [13]
		Vertical position at the eye center	Brunelli [13]
	Shape	Iris radius	Tian [119]
		Mouth orientation	Tian [119]
		Face elliptical shape	Lanitis [70]
		Eye brow thickness	Brunelli [13]
		Four corner of the mouth	Brunelli [13]
		Eleven radii describing the face chin shape	Brunelli [13]
	Edge	Nasolabial region	Tian [119]
		Wrinkle (3 area lateral to eye outer corner)	Tian [119]
		Coarse description of brow's arches	Brunelli [13]
		Line Edge Map	Gao [41]
	Distance	Distance of two blows	Tian [119]
		Mouth width	Tian [119]
		Nose width	Brunelli [13]
		Face bigonial or zygomatic breadth	Brunelli [13]
	Motion	Muscle action	Tian [119]
		Lips motion	Tian [119]

feature. We called pixel value. Face processing technologies are mainly based on the mathematical operations with the pixel values. Within past ten years, researchers had developed numerous methods to deal with this basic information. These methods can be divided into three approaches:

1. Global approach. This approach treats the whole image as the input data. Two dimension image may transform to one dimension feature vector for further processing.

2. Local approach. This is a similar approach to the global

one except performing part of the image as the input data instead of the whole image.

3. Color information approach. Besides gray-scale intensity, a color space contain more information than the gray-scale level. There are many studies [90], [113], [65], [15], [62], [3], [112] used the color information in various fields successfully. This is one of the most efficient method for reducing the search space in face detection.

High dimensionality of the data is one of the problems in the face-related work. One of the motivations for this category is to reduce the dimension of the input data. The extracted coefficients form a feature vector whose dimension is much smaller than the original one. Besides this, the extracted vectors contain more compact features, that discard most of the useless or noisy information. Another motivation is that we do not need to pre-defined a model or apply human knowledge on selecting the features. These approaches should capture the most representative variability of facial appearance. We will discuss each method in later section.

2.1.2 Geometry information

The aim of this category is to retrieve an intermediate level of facial feature from the input data. All the methods are based on the pixel value. The characteristic of human face is being employed to select the features. This pre-defined model is represented by a set of geometry points or data that describes the information about the face. We can further summarize these methods into five sub-categories according to their level:

1. Edge. This is one of the features that we can easily retrieve from the image using edge detector like Sobel filter or Canny filter. Also cognitive psychological studies [10],

[67] show that human can recognize line drawing as the same speed and the same accuracy as gray-level pictures. This points out that edge is one of the important features.

2. Position of the feature point. The points can represent certain information about the face such as the center of the iris and the position of the mouth. The geometry position of the feature points help to build the pre-defined model. Retrieval of these feature points requires a high computational effort.

3. Distance between feature points. The relationships between feature points are recorded. The relative position of different feature can be calculated.

4. Shape of the feature. This is a group of feature points that describe the position and the outline of a feature. The feature can be eye, eyebrow, nose, mouth, face boundaries, etc.

5. Motion of the feature. Geometry movement of the feature is one of the human signals to represent their emotion. The character of each feature should be pre-defined before the extraction of effective information.

Researchers combine the human knowledge and basic pixel information to form different high-level feature. These intermediate features are employed in most of the face processing work including face detection, face recognition, expression classification, age classification, gender classification and pose estimation.

2.2 Extracting and coding of facial feature

After we explore different kinds of facial features, the next problem we need to investigate how the features are coded. The extraction techniques and representation methods are important

Table 2.3: Coding methods for Face Recognition

Type	Reference	Methods
Linear-based	Turk [122]	Principal Component Analysis (PCA)
	Brunelli [13]	Geometric feature-based matching
	Swets [116]	Discriminant Karhunen-Loeve projection (DKL)
	Belhumeur [7]	Fisher linear discriminant (FLD)
	Yu [132]	Direct LDA (D-LDA)
	Lu [82]	Direct fractional-step LDA (DF-LDA)
	Liu [77]	Enhanced Fisher linear discriminant (EFC)
	Bartlett [5]	Independent component analysis (ICA)
	Goudail [44]	Autocorrelation coefficients
Kernel-based	Eickeler [33]	Discrete Cosine Transform (DCT)
	Sanderson [107]	DCT-mod2
	Kim [63]	Kernel PCA (KPCA)
	Lu [81]	Kernel direct discriminant analysis (KDDA)
	Wiskott [126]	Elastic bunch graph matching (Gabor wavelet)
	Liu [78]	Gabor + Enhanced FLD (GFC)
Edge-based	Takacs [117]	Binary Image Metrics
	de Vel [28]	Random rectilinear line segment
	Gao [41]	Line Edge Map (LEM)
Other	Torres [121]	Skin Color
	Georghiades [42]	Illumination Cone
	Chua [20]	3D Point Signature (PS)

for classification especially for face recognition and expression recognition. We will discuss each of the area in this section.

2.2.1 Face recognition

We will discuss the face recognition problem in depth. Survey for face recognition [106], [123], [16], [4], [38] have been published about five years ago. Within the years, lots of new techniques are developed by researchers, that make significant improvement on the results. The recent methods utilized by the face recogni-

tion are shown in Table 2.3. We focus on the feature extraction techniques and face representation methods rather than the classifiers, thus we will not discuss some famous learning based face recognition methods such as Support Vector Machine [103] [96] [46], Self-Organizing Map [71], Neural Networks [71], probabilistic model [93] [109] [92] [135] and hidden Markov model [97]. In general, there are mainly four types of coding methods:

1. Linear-based approach. These are the most common methods for coding the face. Linear mathematical transformation is applied into a set of training data to find a mapping function.

2. Kernel-based approach. Using kernel functions to decompose the face, this approach performs nonlinear transformation on the face.

3. Edge-based approach. The face is represented by a set of edges that are extracted from basic image processing techniques.

4. Other approach. The previous approaches extract the feature from the gray-scale images, color information are being discarded. Color space conversion may have a nonlinear transformation for the input pixel values, that helps to extract some important feature from the color domain. Besides 2D information, researchers have started to investigate the help of 3D information.

Linear-based approach

Turk and Pentland [122] applied principal component analysis (PCA) to face recognition. This is a global method that treats the face image as a two-dimensional recognition problem. The face patterns are not randomly distributed. By using the PCA developed by Sirovich and Kirby [111], faces can be represented

in a more efficient way. They assume that faces are upright at normal to have a smaller set of two-dimensional characteristic views. They first find out the mean face image from the training set $\Gamma_1, \Gamma_2, \Gamma_3 \ldots \Gamma_M$.

$$\Psi = \frac{1}{M} \sum_{n-1}^{M} \Gamma_n, \tag{2.1}$$

and then calculate each face difference

$$\Phi_i = \Gamma_i - \Psi. \tag{2.2}$$

All the face differences form a large matrix and then perform PCA to find out the eigenvectors and eigenvalues, that represent the most expressive information from the face distribution. The covariance matrix is defined as

$$C = \frac{1}{M} \sum_{n-1}^{M} \Phi_n \Phi_n^T = AA^T \tag{2.3}$$

where matrix $A = [\Phi_1 \Phi_2 \ldots \Phi_M]$. The eigenvectors are then sorted by their associated eigenvalues, the first m eigenvectors are selected as principle components. The overall transformation is

$$Z = W_{pca}^t \Gamma \tag{2.4}$$

where W_{pca} is the projection matrix from the original image space to the lower dimensional space. The projection of resultant eigenvectors look like a face, that's why they called Eigenfaces. Some of these faces are shown in Figure 2.1. Figure 2.2 shows the original images and the reconstructed images are shown in Figure 2.3.

These methods greatly reduce the dimension from the number of pixels (N) to the number of training images (m). The computation time and store capacity can be saved. The reconstruction of faces is possible when the extracted features are

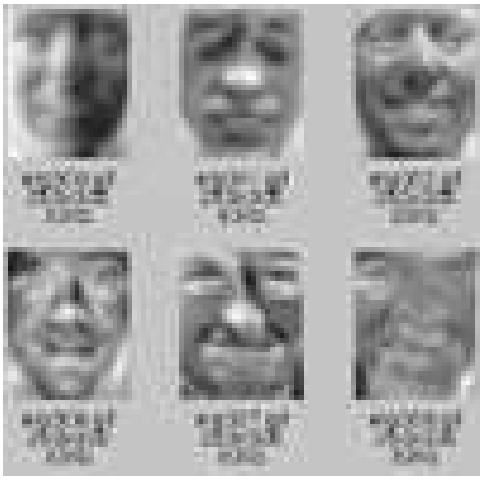

Figure 2.1: Examples of Eigenface

Figure 2.2: Original images

Figure 2.3: Reconstructed images

projected into the eigenspace plus the mean face images. This approach gives a great influence in the literature.

Brunelli and Poggio [13] is one of the early method to combine the use of geometry information and PCA method on face recognition. The idea is to extract a set of parameters to represent the facial feature such as eyes, nose, mouth and chin. They calculate the horizontal and vertical gradients to locate the position of facial features and to form a 35-D numerical vector. The PCA method is applied to increase the effectiveness of the feature vector. The extracted feature vectors from the training data are sued to train a Bayesian classifier for further recognition. They give a direction for the usage of geometry information on face recognition.

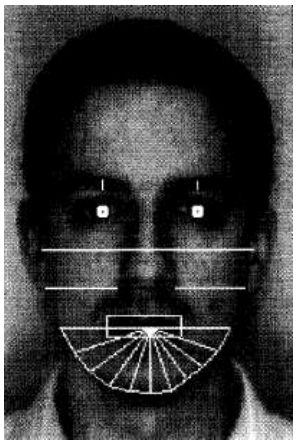

Figure 2.4: Geometric feature (white) used for face recognition. [13]

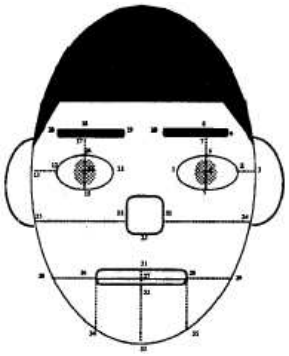

Figure 2.5: Feature points for calculation of distance vector. [24]

Mixture Distances are developed by Cox et al. [24] for face recognition. They first select a set of facial feature points (Figure 2.5) manually and then calculate a set of feature distance to form a 30-D feature vector. A mixture of mixtures technique is used to remove the uncertainty of choosing a particular order statistics or model, that reported the best result. They demonstrate the important of the use of mixture-distance instead of a simple Euclidean distance.

Swets and Weng [116] proposed a method - discriminant Karhunen-Loève (KL) projection to solve the problem of de-

generation the within-class scatter matrix in small sample size (SSS). They suggest performing two projections, i.e., the discriminant analysis projection is performed in the space of the KL projection. They first project the N-dimensional image space into m-dimensional space using the KL projection. The between-class scatter matrix is defined as

$$S_B = \sum_{n=1}^{c} N_i \Phi_n \Phi_n^T, \tag{2.5}$$

and the within-class scatter matrix is defined as

$$S_w = \sum_{n=1}^{c} \sum_{x_k \in X_n} \Phi_n \Phi_n^T \tag{2.6}$$

where Φ_n is calculated from (2.2) and N_i is the number of samples in class X_n. To have a better discriminant power from the data, the ratio of the determinant should be maximized by maximizing the between-class scatter while minimizing the within-class scatter,

$$W_{opt} = \arg\max_{W} \frac{|W^T S_B W|}{|W^T S_W W|} \tag{2.7}$$

where W is the projection matrix from image space to the most discriminating feature (MDF) space. The DKL projection to the MDF space is

$$Z = W_{opt}^t \Gamma \tag{2.8}$$

where W_{opt} is the projection matrix from the image space to the optimal space, and the final dimension of Z can be reduced to the number of sample classes. The benefits of class specific linear projection is to achieve greater between-class scatter that help to increase the correct classification rate.

However, the KDL projection has some difficulty when the within-class scatter matrix S_W is singular. Belhumeur et al. [7] modified the KDL projection to solve this problem. The method is called Fisherface, a derivative of fisher linear discriminant (FLD) [37]. They employ the standard FLD instead of

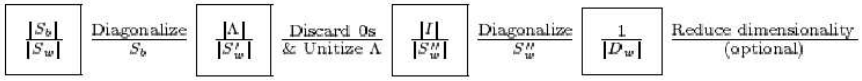

Figure 2.6: Overview of the Direct LDA algorithm [132]

the linear discriminant analysis for the second projection. The new optimal projection matrix is

$$W_{opt}^t = W_{fld}^t W_{pca}^t \tag{2.9}$$

where

$$W_{pca} = \arg\max_W |W^T S_T W|$$
$$W_{fld} = \arg\max_W \frac{|W^T W_{pca}^T S_B W_{pca} W|}{|W^T W_{pca}^T S_W W_{pca} W|}. \tag{2.10}$$

Experimental results demonstrate that Fisherface method performs better than Eigenface. Also, Fisherface method can be more robust than Eigenface when handling variation in lighting and expression. This is one of the best face recognition in the literature.

The KDL and Fisherface methods are a two-stage method using the technique of PCA and LDA-like approach. This is because the traditional LDA algorithm is computational expensive and may be infeasible when the dimension of data is too high. Yu and Yang has proposed a direct LDA (D-LDA) algorithm for high dimensional data [132]. The key idea of the algorithm is to discard the null space of S_B rather than S_W in (2.7). The S_B contains no useful information while S_W contains the most discriminative information. They reverse the diagonalization order to achieve, that mean diagonalize the matrix S_B first and then S_W. The conceptual overview is shown in Figure 2.6. Fukunaga [40] also show that Fisher's criterion can be one of variant:

$$\arg\max_W \frac{|W^T S_T W|}{|W^T S_W W|} \tag{2.11}$$

where $S_T = S_b + S_W$ is the total scatter matrix. The first step of the proposed algorithm will equivalence to PCA method and the remaining step will give the same performance with Fisherface. This algorithm provides an efficient implementation since they perform without the PCA step and eliminates the problem of singular in S_w.

Lu et al. introduce a new algorithm DF-LDA [82]. This is a combination of D-LDA and fractional step linear discriminant analysis algorithm (F-LDA) [80]. The algorithm first applies a new variant of D-LDA and then performs F-LDA to the data set. The variant of D-LDA is to change the Fisher's criterion from (2.11) to

$$\arg\max_{W} \frac{|W^T S_B W|}{|W^T S_T W|}. \tag{2.12}$$

That results in a low-dimensional SSS free subspace without discarding the most discriminant features. Besides this, a weighting function is introduced in D-LDA for sub-sequent F-LDA to reorient the subspace into a set of optimal discriminant features for face representation. The paper made a comparison among Eigenface, Fisherface, D-LDA and proposed method. The experimental results show that DF-LDA achieves the lowest error rates on two popular face databases.

Liu et al. [77] proposed an enhanced Fisher classifier (EFC) that integrated the pixel and geometry information to enhance the recognition method. A similar approach is early proposed by [69]. The Enhanced FLD Model (EFM) [75] is employed on these two features. The geometry features are selected manually from the image. Figure 2.7 shows the control points that represents the shape of a face image. The feature points include eyebrows, eyes, nose, nasolabial region, mouth and face boundary. A mean shape is first calculated by averaging the aligned face images of the training images. All the face images will be normalized to a shape-free face image by warping the original image to

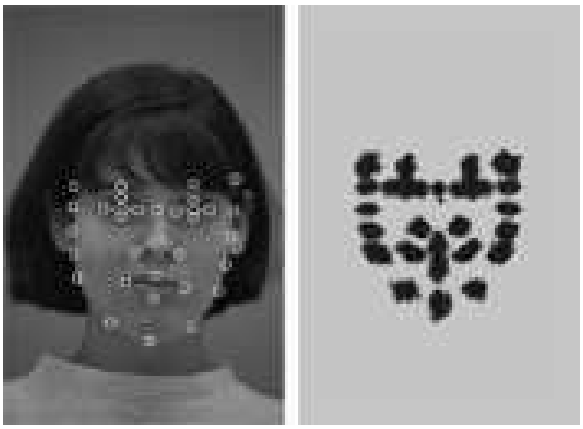

Figure 2.7: The control points for EFC. Left: the shape of a face image represented by 32 control points (white circles). Right: the distribution of the aligned shapes of the training images. [77]

the mean shape. Using PCA, the lower dimensional shape and texture features Z are then integrated using the normalization equation:

$$Z = \begin{pmatrix} \frac{Y_1^t}{\|Y_1\|} \frac{Y_2^t}{\|Y_2\|} \end{pmatrix}^t \tag{2.13}$$

where Y_1 and Y_2 are the projection matrix of shape space and texture space that calculated by (2.4), respectively. They treat the pixel and geometry information as equal important discriminating power. The within-class covariance matrix S_B of Z is calculated by (2.6) and the between-class covariance matrix S_W of Z is calculated by (2.5). EFM first diagonalizes the matrix S_W to find out the eigenvector E and the diagonal eigenvalue V. A new between-class covariance matrix is computed

$$E^{-1/2} V^t S_b V E^{-1/2} = K_b,$$

and diagonalize the new matrix K_b

$$K_b \Theta = \Theta \Lambda \ \ and \ \ \Theta^t \Theta = I$$

where Θ, Λ are the eigenvector and the diagonal eigenvalue ma-

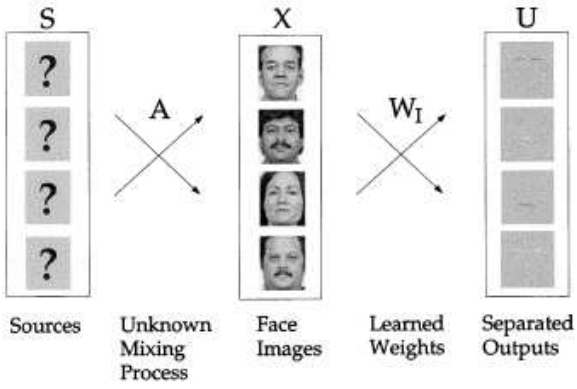

Figure 2.8: The model of ICA. The images in X are considered to be a linear combination of statistically independent basis image S, where A is an unknown mixing matrix. The basis images were estimated as the learned ICA output U. [5]

trices of K_b. The overall projection matrix of EFM is:

$$W_{elm} = EV^{-1/2}\Theta, \tag{2.14}$$

and each new face image are to compute the integrated feature Z and then the new feature X:

$$X = W_{elm}^{t} Z. \tag{2.15}$$

They show that the contour of the face is one of the important discriminative features and the pixel information takes the internal facial features as the discriminative features. This approach also yields the best recognition rate (98.5%) even only 25 features when compare with the Eigenfaces.

Independent component analysis (ICA) is being used for face recognition by Bartlett et al. [5] recently. There are several algorithms for performing ICA. The infomax algorithm developed by Bell and Sejnowski [8] are chosen in that paper. The basic concept for ICA is to define a model for the observed multivariate data from a set of training data. Figure 2.8 shows the model of ICA. Assume that the pixel values in face images were generated from a linear mixing process and have a super-Gaussian

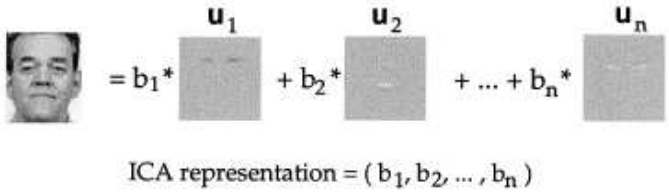

Figure 2.9: The representation of ICA. The independent basis image representation consists of the coefficients, **b**, for the linear combination of independent basis images, **u**, that comprises each face image **x**. [5]

response distribution. PCA is applied to reduce the dimension of the data and the first m principles eigenvectors of the image set P_m is chosen. The independent source images in the rows of U is produced by performing ICA on P_m^T

$$P_m^T = W_I^{-1} U. \tag{2.16}$$

By minimized the squared error of the reconstruction of the P_m, $R_m = XP + m$, the error approximation of X is obtained by $\hat{X} = R_m P_m^T$. Thus, combine with (2.16),

$$\hat{X} = R_m P_m^T = R_m W W_z^{-1} U = R_m W_I^{-1} U \tag{2.17}$$

where W_z is the sphering matrix produced by ICA and W_I is the full transform matrix of ICA. The coefficients for the linear combination of independent sources U can be retrieved from $R_m W_I^{-1}$. Figure 2.9 shows the representation of ICA. A comparative assessment of ICA for face recognition also provided by Liu and Wechsler [74] and showed that ICA can outperform the PCA method. The reason is that ICA can extract higher order statistics than PCA's second order covariance decomposition.

Goudail et al. [44] employed the local autocorrelation coefficients to represent a face. The method uses different kernels (Figure 2.10) to scan the image and the product of pixel marked in black is computed. We can formulate the equation of the co-

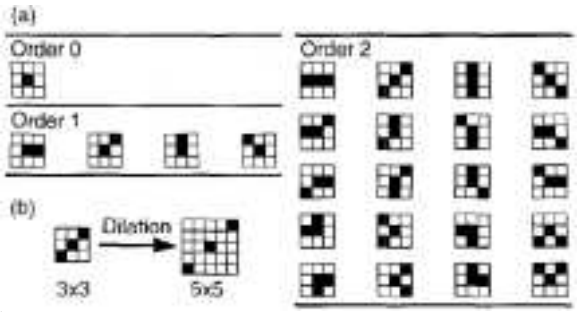

Figure 2.10: Set of autocorrelation kernels used for feature extraction. (a) The kernels used for 3×3 pixels neighborhood. The order of a kernel is its number of active pixels minus 1. (b) Example of scaling of the kernel from size of 3×3 to size of 5×5 pixels. [44]

efficient:

$$C_k = \frac{\sum_m \prod_n I_{ij} B_{ij}}{C_0} \tag{2.18}$$

where k is the kernel, m is all the possible scan window inside the image, n is all the pixel inside the scan window, I_{ij} is the intensity value of the pixel(i, j) and B_{ij} is the boolean value of the active pixel in the kernel. The storage space for representing a face is very small and do not have the problem of small sample size for training the system. The classification procedure is also efficient.

Kernel-based approach

There are several papers using the 2D Discrete Cosine Transform (DCT) coefficients as the features [97], [66]. Eickeler et al. [33] proposed one of the representative approach using 2D DCT coefficients for face recognition. An image is first compressed in JPEG standard and then calculates the observation feature vector using the DCT. Each pixel in the sampling window ($N \times N$)

is transformed according to the equation:

$$C(u,v) = \alpha(u)\alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \cos\left(\frac{(2x+1)u\pi}{2N}\right) \cos\left(\frac{(2y+1)u\pi}{2N}\right) \tag{2.19}$$

where

$$\alpha(w) = \begin{cases} \frac{1}{\sqrt{N}} & : \ for \ w = 0 \\ \frac{2}{\sqrt{N}} & : \ for \ w = 1, 2, \ldots, N-1 \end{cases}.$$

The DCT transforms the information from spatial domain to frequency that can compact the energy well. The first 15 coefficients are extracted to form the feature vector. The vector is defined as:

$$x = [\ C_0^{(u,v)} \quad C_1^{(u,v)} \quad \ldots \quad C_{M-1}^{(u,v)}\ ]^T \tag{2.20}$$

where $C_n^{(u,v)}$ is the nth DCT coefficient and M is the dimension of the vector. The method can process on the JPEG and MPEG standard to retrieve features directly. This is an efficient and powerful approach for analyzing huge multimedia resource such as video in digital library.

Sanderson and Paliwal [107] introduced the DCT-mod2 extraction technique that improve the previous method so that the extracted features are more insensitive to the illumination direction. They analyze the DCT coefficients and conclude that the 0th DCT coefficient is related to the mean intensity value inside each block window that will be affected by the illumination change directly. Furthermore, the first and second coefficients reflect the average horizontal and vertical intensity changes, respectively. They define the nth delta coefficient for the replacement of the first three DCT coefficients:

$$\Delta^z C_n^{(u,v)} = \frac{\sum_{k=-K}^{K} k h_k C_z}{\sum_{k=-K}^{K} h_k k^2} \tag{2.21}$$

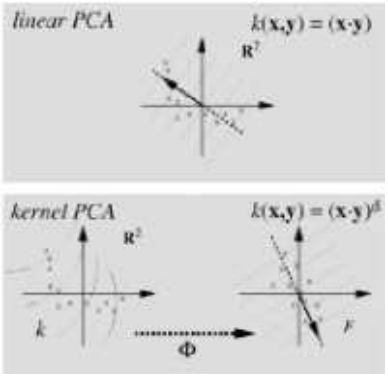

Figure 2.11: The basic idea of KPCA. [110]

where h is a $2K+1$ dimensional symmetric window vector,

$$z = \begin{cases} h & for \text{ horizontal delta coefficient} \\ v & for \text{ vertical delta coefficient} \end{cases} \text{ and}$$

$$C_z = \begin{cases} C_n^{(u,v+k)} & for \text{ horizontal delta coefficient} \\ C_n^{(u+k,v)} & for \text{ vertical delta coefficient} \end{cases}.$$

The original DCT feature vector (2.20) is changed to:

$$x = [\ \Delta^h C_0\ \Delta^v C_0\ \Delta^h C_1\ \Delta^v C_1\ \Delta^h C_2\ \Delta^v C_2\ C_3\ C_4\ \ldots\ C_{M-1}\ ]^T \tag{2.22}$$

where the superscript of (u, v) is omitted for simplification. A various light illumination testing image is used in the experiment. The result shows that it outperforms the other methods in terms of the variety of lighting direction and execution time.

Kim et al. [63] employed the kernel-based PCA (KPCA) method to extract the facial feature. This method is early introduced by [110], [95]. The basic idea of KPCA contains two steps. The first step is to map the original data $\mathbf{x}$ into a feature space F via a nonlinear mapping Φ. Figure 2.11 shows the idea of nonlinear mapping. The second step is to perform the linear PCA that we have discussed in early section in F. The

covariance matrix in F is different from (2.3),

$$C = \frac{1}{M}\sum_{j=1}^{M}\Phi(\mathrm{x}_j)\Phi(\mathrm{x}_j)^T. \tag{2.23}$$

To solve the eigenvalue equation,

$$\lambda \mathrm{V} = C\mathrm{V} \tag{2.24}$$

where $\lambda \geq 0$ and $\mathrm{V} \in F\backslash\{0\}$. The solutions V lie in the span of $\Phi(\mathrm{x}_1), \ldots, \Phi(\mathrm{x}_M)$ and the coefficients $\alpha_i (i = 1, \ldots, M)$ such that

$$\mathrm{V} = \sum_{i=1}^{M}\alpha_i\Phi(\mathrm{x}_i). \tag{2.25}$$

Consider the equivalent equation

$$\lambda(\Phi(\mathrm{x}_k)\cdot \mathrm{V} = (\Phi(\mathrm{x}_k)\cdot C\mathrm{V}) \quad \text{for all } k = 1, \ldots, M. \tag{2.26}$$

Defining an $M\times M$ matrix K that can be expressed in terms of the dot products of two mappings by

$$K_{ij} := (\Phi(\mathrm{x}_i)\cdot\Phi(\mathrm{x}_j)) = \mathrm{k}(\mathrm{x}_i, \mathrm{x}_j). \tag{2.27}$$

Now, solve the equation

$$M\lambda\alpha = K\alpha \tag{2.28}$$

for nonzero eigenvalues λ_l and eigenvectors $\alpha^l = (\alpha_1^l, \ldots, \alpha_M^l)^T$ subject to the normalization condition $\lambda_l(\alpha^l \cdot \alpha^l) = 1$. The polynomial kernel with degree d is chosen for the non-linear mapping Φ_d,

$$(\Phi_d(\mathrm{x})\cdot\Phi_d(\mathrm{y})) = \left(\sum_{i=1}^{N} x_i \cdot y_i\right)^d = (\mathrm{x}\cdot\mathrm{y})^d. \tag{2.29}$$

The input data x are projected onto the eigenvector v^l in F, where the inner products are computed. The lth nonlinear principal component is calculated by linearly combination of weights

α_i^l and Φ. One of the properties is that the number of extracted principal components can be more than the dimension of the input data. There are disadvantages of using kernel PCA to extract features, i.e., the reconstruction of the original pattern using the extracted principal components is not a easy task when compare with the linear PCA method, and the computational complexity is more expensive than its linear method.

Different kernel version of linear-based methods have been developed and applied on face recognition problem. Besides KPCA, an enhanced kernel D-LDA method (KDDA) is proposed by Lu et al. [81]. KDDA is a derived method of D-LDA and generalized discriminant analysis (GDA) [6]. This method is similar to DF-LDA [82] except the kernel function is being used. For two arbitrary class Z_l and Z_h, a $C_l \cdot C_h$ dot product matrix K_{lh} can be defined as

$$K_{lh} = k(z_{li}, z_{hj}) = \phi li \cdot \phi_{hj}. \tag{2.30}$$

Then a $L \times L$ kernel matrix K is defined for non-linear mapping

$$K = (K_{lh})_{l=1,...,C \;\; h=1,...,C}. \tag{2.31}$$

They also modified the Fisher's criterion from (2.7) to (2.12) to solve the singular problem. A low-dimensional representation y on z is

$$y = \left(E_m \Lambda_b^{-1/2} P \Lambda_w^{1/2}\right)^T (\Phi_b^T \phi(z)) \tag{2.32}$$

where P are the eigenvectors of input data. KDDA provides a nonlinear mapping for high dimensional data into a "linearized" and "simplified" feature sub-space. Like DF-LDA, the SSS problem can also be solved and retrieve the optimal discriminant subspace of the feature space without losing any significant discriminant information. The experimental result shows that the recognition rate is superior to KPCA and GDA on a multi-view face database. This may due to the non-linear property of different poses of the face images.

Another famous approach called Elastic Bunch Graph matching (EBG) is proposed by Wiskott et al. [126][127]. A similar idea is early proposed by Lades et al. [68], and has been used by [72]. They use the concept of labeled graphs [14] together with the Gabor wavelet coefficients to represent a face image. The face image is convolution with the Gabor filter, the result is shown in Figure 2.12. Position information of a face is used to deform the image plane and the edges are labeled with the distance information. Gabor wavelets are biologically motivated convolution kernels that consist of plane waves and Gaussian envelope function [26]. Each node in the labeled graph are represented by Gabor wavelet coefficients called jets. A mother kernels is defined as:

$$\Psi_j(\vec{x}) = \frac{k_j^2}{\sigma^2} \exp\left(-\frac{k_j^2 x^2}{2\sigma^2}\right) \left[\exp\left(i\vec{k}_j\vec{x}\right) - \exp\left(-\frac{\sigma^2}{2}\right)\right] \quad (2.33)$$

where $\vec{k_j}$ is the vector of plane wave and $\exp\left(-\frac{\vec{k}_j x^2}{2\sigma^2}\right)$ are the restriction of the Gaussian envelope function. A set of jets contain 5 different frequencies, index $\nu = 0, \ldots, 4$, and 8 orientations, index $\mu = 0, \ldots, 7$,

$$J_i(\vec{x}) = a_j \exp(i\phi_j) \quad (2.34)$$

where a_j is the magnitudes of the jet, that slowly vary with position, and phases ϕ_j that rotate at a rate approximately determined by the spatial frequency. The width $\frac{\sigma}{k}$ of the Gaussian is controlled by the parameter $\sigma = 2\pi$. The second term in the bracket of (2.33) makes the kernels DC-free. The value of $\vec{k}_j$ is expressed with:

$$\vec{k}_j = \begin{pmatrix} k_{jx} \\ k_{jy} \end{pmatrix} = \begin{pmatrix} k_\nu \cos\varphi_\mu \\ k_\nu \sin\varphi_\mu \end{pmatrix}, k_\nu = 2^{-\frac{\nu+2}{2}}\pi, \varphi_\mu = \mu\frac{\pi}{8}, \quad (2.35)$$

with index $j = \mu + 8\nu$. All kernels are generated from one mother wavelet by dilation and rotation because the family of kernels is

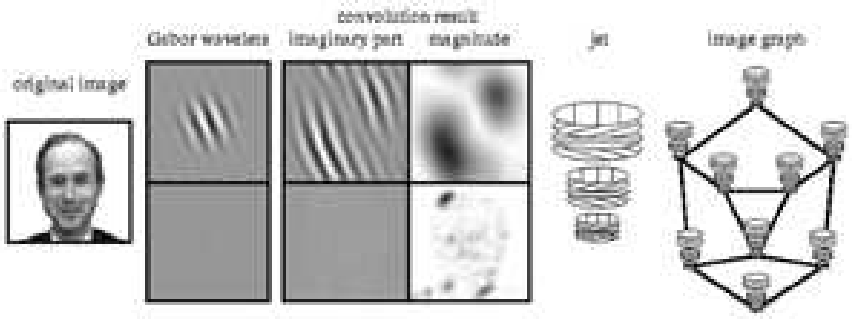

Figure 2.12: The graph representation of a face is based on the Gabor wavelet transform, a convolution with a set of wavelet kernels.

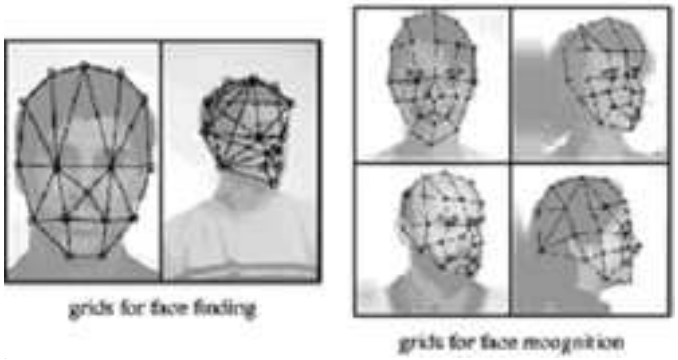

Figure 2.13: Object-adapted grids for different poses. [126]

self-similar. A set of fiducial points is used to represent the face image such as the pupils, the eyebrows, corners of the mouth and the tip of nose. A labeled graph G is object-adapted, that represents a face consists of N nodes on these fiducial points at positions $\vec{x}_n$ and E edges between them. Figure 2.13 shows some examples of the labeled graph. This deformable model and the use of Gabor wavelets can resist the variation of translation, scaling, rotation in head pose, and deformation due to facial expression in the image plane. Profile view of face images can also be recognized by these approaches. Gabor Wavelet has been employed in various pattern recognition areas [9].

Liu et al. [78], [76] combined the technique of Gabor wavelet

and EFM to form a novel Gabor-fisher Classifier (GFC) for face recognition. They first use the Gabor kernel $\Psi_{u,v}$ (2.33) to generate a set of Gabor feature $O_{u,v}(z)$ from image I,

$$O_{u,v}(z) = I(z) * \Psi_{u,v}(z) \tag{2.36}$$

where $z = (x, y)$ and $*$ denotes the convolution operator. The Gabor feature is then downsample by a factor of p to reduce the space dimension, and normalize it to zero mean and unit variance. The normalized feature vector $O_{u,v}^{(p)}$ is then concatenated to form the augmented Gabor feature $\chi^{(p)}$

$$\chi^{(p)} = \left(O_{0,0}^{(p)^t} \; O_{0,1}^{(p)^t} \; \cdots O_{4,7}^{(p)^t}\right)^t. \tag{2.37}$$

The EFM will be applied on the augmented Gabor feature $\chi^{(p)}$ to further reduce the dimension of the data and to extract the most discriminative information. Using (2.14), the final feature vector is defined as

$$\zeta^{(p)} = W_{elm}^t \chi^{(p)}. \tag{2.38}$$

Their experiment shows that the proposed method can achieve 100% recognition rate using 62 features on a data set from the FERET database. This may due to the usage of the Gabor wavelets that are optimized for face processing problems and EFM's discriminatory power.

Edge-based approach

Edge information is being used in wide range of pattern recognition tasks, however it has been neglected in face recognition for a long time.

Takacs and Wechsler [117] used the binary image metrics to compare faces. They proposed a shape comparison method that operates on edge maps and derives similarity measures using the modified Hausdorff distance [31]. Sobel edge detection is applied and thresholded using an adaptive technique to form the binary

face image. This image is scaled to $N \times M$ pixels to represent the face in the database. A simple distance similarity measure is used for classification. The dimension of extracted features can be as low as 1% of the original image and the proposed method is robust to illumination change due to the insensitivity of edge. However, this pixel-wise template matching method does not include the geometry information of features and can't deal with the image that variant in pose.

de Vel and Aeberhard [28] proposed a method of using the intensity values of line segments as the representation of faces. They first define a face boundary and then randomly choose two pixels (p_1, p_2) inside that face boundary to form a rectilinear line segments $L(p_1, p_2)$. Each face class F_k in the training set of T_k image of different views is represented by $N_k = T_k \times N_{T_k}$ lattice lines. The distance measure between two lines $L_{r,s}$ and $L_{m,n}$ is defined as:

$$D(L_{r,s}, L_{m,n}) = \sum_{q=1}^{l} ((L_{r,s}^{(q)} - (L_{m,n}^{(q)} + \Delta))^2) \tag{2.39}$$

for $r, m = 1, 2 \ldots, N_k$, and $s, n = 1, 2 \ldots, K$, where Δ is the compensate value for the illumination by shifting two lines towards the same mean value. The benefits of this approach are computationally efficient, robust to variations in scale, expression and light condition. One assumption for this approach is that face boundary is known. Thus, the occluded face area will affect the recognition result.

Cognitive psychological studies [10], [67] show that human can recognize line drawing as the same speed and the same accuracy as gray-level pictures. This points out that the edge is one of the important features. The previous method employs the Hausdorff distance that only uses the spatial information of an edge map; the location of features is being ignored. Gao and Leung [41] had modified the distance measure method and proposed a face recognition method that is based on the Line

Figure 2.14: An example of LEM. [41]

Edge Map (LEM). They also introduce a novel Line Segment Hausdorff Distance (LHD) for the distance measurement of two segments. This approach combines the geometry information and pixel information into consideration. It first groups the pixels of face edge map to line segments and then performs a thinning operation. Finally, a polygonal line fitting process is applied to form the LEM of a face. Figure 2.14 shows an example of LEM. The edge information is expected to be more robust in illumination changes. This is one of the advantages of intermediate-level features. Beside this, the face prefiltering is also possible to speed up the searching of images in the huge database while the other approaches can't be achieved. These preprocess of LEM matching give a great convenient in face identification. Experimental results show that the proposed method can perform better than the well-known Eigenface approach. The LEM method can give researcher a new concept for coding and recognition faces.

Other approach

Most of the-state-of-the-art methods are based on the gray-scale image and the color information has been discarded. However, some studies show that the color information can improve the performance of the original method.

The original Eigenface method uses the luminance component only. Torres et al. [121] stated that color information can improve the recognition rate. They apply the PCA method on different color channel independently to extract the principle components. The global distance between the test image and the training data is calculated by the weighted sum of different channels. Experiments are done using several color schemes such as Y, RGB, YUV and SV. The results show that the performance of YUV and SV are much better than the original approach (Y). This may due to the non-linear transform of the color space, that contains more discriminant information.

Skin color information also being as feature for face verification. Marcel and Bengio [84] encoded the color histograms to form a feature vector, and Multi-Layer Perceptrons (MLP) is served as classifier. They first transform the color space to normalized chrominance space (r-g) and calculate the histograms of each channel. The histograms are then quantized into 32 levels and concatenated to form the feature vector with 96 (32x3) components. This feature vector will feed into the MLP for training. The experimental results show decrease on the error rate when color information is involved.

Besides the above methods, researchers start to use or construct 3D information for face recognition. There are other approaches developed by researchers based on face synthesis. Illumination Cone is one of the novel approaches developed by Georghiades et al. [43], [42]. A set of controlled face images is needed for calculating the convex Lambertian surface of each class. The shape and albedo of the face can be reconstructed. Illumination approximation is possible in a low-dimensional subspace that enhances the robustness of recognition under a large range of pose and lighting condition.

Tanaka et al. [118] represents the faces using Extended Gaussian Image (EGI). By mapping principal curvatures and their

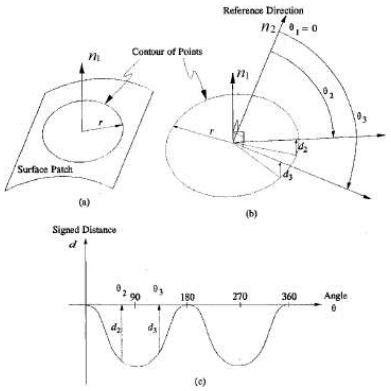

Figure 2.15: Definition of point signatures [20]

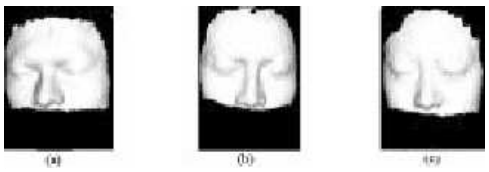

Figure 2.16: The extracted rigid face regions of three persons. [20]

directions at each pixel, two types of 3D directional facial features, ridge lines and valley lines, are extracted to form feature vector. The fisher's spherical correlation coefficient of ridge lines and valley lines are used as the similarity measure. However, the performance is not as good as other methods.

Chua et al. [20] introduce point signature (PS) [19] for face recognition problem, that is a representation for free-form surfaces. PS are represented by the signed distance and corresponding clockwise rotation angle at each point along two reference points. Figure 2.15 shows the definition of point signature. They treat the face as a 3D non-rigid surface object. A set of images with different expressions is registered to form two face surfaces. Afterward, the rigid parts are found out from the non-rigid parts. Figure 2.16 shows the extracted rigid face region. The non-rigid regions will have high registration errors that need to ignore for building the face model. The matching is accepted when all the difference between each distance pair is smaller than a threshold. Phase shifting is possible to reduce the error when more than one local maximum inside the signature. The representation of PS is a simple 1D feature vector, thus the matching process is fast and efficient. The surface of faces can

be described more completely that help for recognition. Moreover, it is invariant to translation and rotation. However, the proposed algorithm only being tested in 6 face range images and the evaluation of large data set is omitted.

3D feature points are integrated with 2D feature points for face recognition by Wang et al. [125]. Gabor wavelet filters are applied on the ten 2D feature points while point signature are used to represent the four 3D feature points. The shape and texture features are extracted from 3D feature points and 2D feature points, respectively. PCA method is applied to reduce the dimension of feature vector and they choose the SVM as the classifier for the recognition. They demonstrate the improvement of performance using fusion of 2D and 3D information.

2.2.2 Facial expression classification

Human's emotion can play an important role in interpersonal communication, this brings out the facial expression classification problem for human-computer interaction. Most of the researcher has focused on recognize the facial expression in term of action units (AUs) of the Facial Action Coding System (FACS) that is developed by Ekman and Friesen [34]. They define 44 FACS AUs and 33 of those are related to the contractions of specific facial muscles. The upper face contains 12 AUs while the lower face has 18 AUs. By combining different AUs, most of the human emotions can be expressed. The main problems are how to extract the features and to represent the features. The recent techniques can be divided into three main categories:

1. Optic flow estimation approach. Muscle contraction is one of the key factors for facial analysis. The usage of optic flow can estimate the motion of the muscles.

2. Global analysis approach. This is a similar technique that applied in face recognition. The pixel information is pro-

cessed by linear transformation or kernel decomposition to retrieve the features.

3. Local analysis approach. The difference between local analysis approach and global analysis approach is that local approach performs the filtering on part of the image as the input data instead of the whole image. The subset of the images is usually small image pitcher near the facial feature such as eyes, nose and mouth.

In the following section, we will have a general overview about different approaches for expression recognition.

Optic flow estimation approach

This is the most common method that researchers have focused on facial expression recognition. Mase [88] first used the optical flow to estimate the motion of the facial muscles. Essa et al. [35] introduced the distributed response of a set of templates to observe a given facial region. Both skin and muscle dynamics are described by a physical model, and the muscle control variables are estimated. Essa and Pentland [36] modified the previous approach by using an optimal optical flow method. A looping-system is used for the correction of the physical model. The system can learn the "ideal" spatio-temporal patterns for coding, interpretation and recognition of facial expressions.

Yacoob and Davis [129] worked without the physical model and based on the statistical distribution of the motion direction field to classify the expressions that are constructed by a mid-level symbolic representation. Each feature is represented by a rectangular window rather than a set of control points. This tracking algorithm combines the spatial and temporal information at each frame. The flow magnitudes are computed by the optical flow field and the motion vectors are quantized into eight

Figure 2.17: Feature point marking. [21]

Figure 2.18: Feature points of the model on a training image. [69]

directions. The six expressions are recognized when the satisfactory actions are detected. Rosenblum et al. [104] extended the former approach and applied the Radial Basis function Network for classifying the facial expression.

The previous approaches recognize facial expressions into a small set of prototypic expressions such as happiness or sadness. Cohn et al. [21] suggested that human's emotion expressions are complex and varied. The prototypic expressions seldom occur in daily life. Thus, they capture the full range of emotion expression by discriminating the subtle changes in facial expression that based on feature point tracking. Figure 2.17 shows the manually marked facial point for tracking. The displacement of different feature points are used to form flow vectors that will be applied the discriminant function analysis to extract the high level features.

Global analysis approach

The PCA method has been applied on the expression recognition by Lanitis et al. [69]. They proposed a generic, compact and parameterized model to represent the facial appearance. The model is generated by a set of feature points as shown in Figure

2.18. A mean shape and mean image are first calculated from the training images. All the face images will be normalized to a shape-free face image and transformed to the mean shape. The Active Shape Model search (ASM) [22] is employed to deform the original image into the shape-free image. The PCA technique is applied to extract the feature parameters both in the shape model and gray-scale model. That paper had shown the possibility of applying the generic approach that includes pose estimation, person identification, gender recognition and expression recognition.

Kimura and Yachida [64] performed PCA on the motion vectors to form a 3D emotion space. Potential Net model is used for extracting motion flow of facial expression. The displacement vector is calculated from the position of deformed grids and PCA is performed on these vectors to form the subspace. Each of the expressions is described by the top three principal components and the degree of them is estimated by the distance from the origin point.

In previous study [7], FLD can improve the recognition performance when compare with PCA. Donato et al. [30] apply the FLD method for the expression recognition. The linear assumption for facial expression classification is that the images of a facial action across different faces while the images of the same person lie in a linear subspace for face recognition. They called this method as FisherActions. The steps of the method are already explained in previous sub-section face recognition. The goal is to find a transformation matrix by (2.10).

Besides FisherActions, Donato et al. [30] also applied ICA for facial expression recognition. The procedure of performing ICA is similar to that used in face recognition. ICA can process high-order dependencies of the image set include nonlinear relationships among pixel information while PCA and FLD are based on second-order dependencies. However, the inherent or-

dering of the independent components is not known. They define the ratio of between-class to within-class variability r, for each coefficient

$$r = \frac{\sigma_{between}}{\sigma_{within}} \tag{2.40}$$

where $\sigma_{between} = \sum_j (\overline{a}_{jk} - \overline{a}_k)^2$ is the variance of the j class means and $\sigma_{within} = \sum_j \sum_i (\overline{a}_{ijk} - \overline{a}_{jk})^2$ is the sum of the variances within each class. The independent component representation is selected by the first p independent component according to the class discriminant power. ICA gave the best performance among several methods that descried in their paper.

Hong et al. [54] applied the idea of EBG for facial expression recognition. A personalized gallery, that contains images of the same person with different facial expression, is constructed for each person. Node weighting and weighted voting are used for classification of expression. Gabor feature space is also optimal in high frequency components. This provides a good method for representing feature of facial expression. Lyons et al. [83] combine the Gabor wavelet representation and the grid representation of face to form a labeled graph vector (LG vector). This LG vector is first perform the PCA to reduce the dimension and then LDA is used to clusters the facial attributes. This method can be generalized to perform on facial expression, gender and race recognition.

Zhang et al. [134] proposed a method that combine two types of features. One is the geometry position of 34 facial points and the other one is the Gabor wavelet coefficients with 3 different scales and 6 different orientations. Each image is convoluted with the Gabor kernels to form a 612(3x6x34)-dimensional vector for further process. A multi-layer perceptron is used to train the system. The experiment results show that the number of hidden nodes can be reduced when using both the Gabor coefficients and geometry positions.

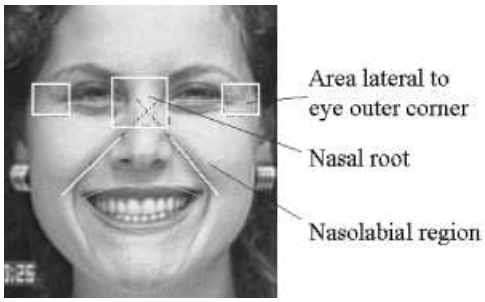

Figure 2.19: Example of nasolabial furrows, nasal root, and outer eye corners. [119]

Local analysis approach

Permanent facial features and transient facial features are extracted for expression recognition by Tian et al. [119]. The permanent features include brows, eyes and mouth, that are commonly used by other approaches. The transient features are produced by contraction of the facial muscles or by aging effect that include facial lines and furrows. Figure 2.19 shows an example of furrows near the nose region. Canny edge detector is used to quantify the amount and orientation of furrows. These features are translated to a set of parameters that describe the position, shape and orientation of the features. The parametric descriptions of features are then passed into the neural network for training and recognizing. The experimental results show that the parameterization of facial features perform well when compare with the template-based approach.

Donato et al. [30] defined local filters based on the Gabor wavelet decomposition for expression recognition. This method is developed based on Lades et al. [68]. The mother kernel of Gabor wavelet 2.33) is used to generate a set of filters to convolute with the original δ-image. The outputs jets calculated by (2.34) were downsampled by a factor q to reduce the dimension of the output data and normalized to unit length. Figure 2.20

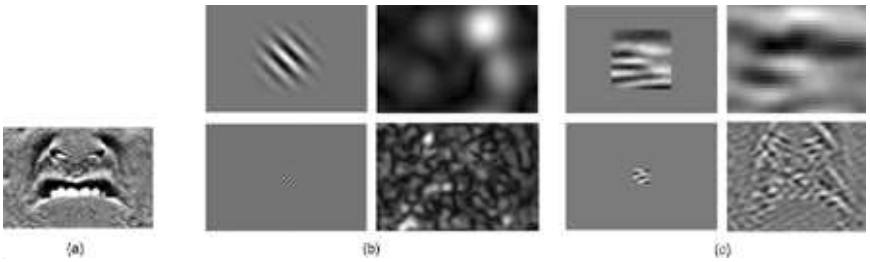

Figure 2.20: An example of the Gabor decomposition (a) Original δ-image. (b) Gabor kernels (low and high frequency) with the magnitude of the filtered image to the right. [30]

shows an example. The resulting outputs form the feature vector for further similarity measure and classification. By observing the experimental result, they conclude that higher spatial frequency bands of the Gabor filter representation contain more information than that of the lower frequency bands.

We have summarized the techniques used to extract the features for facial expression recognition. We find that the representation methods are very similar to those of face recognition except the optical flow for detecting the motion of muscle.

2.2.3 Other related work

Beside the face recognition and expression recognition, age prediction is a tough problem on face processing field. Researcher has put much effort to improve the accuracy, however, the techniques are not mutual. One argument of the problem is that even if human still cannot have confident to confirm the correctness of results.

Tiddeman et al. [120] proposed a wavelet based method for transforming facial feature. They first construct a shape model by averaging the position of the feature points,

$$x'_s = x_s + (x_{p2} - x_{p1}) \tag{2.41}$$

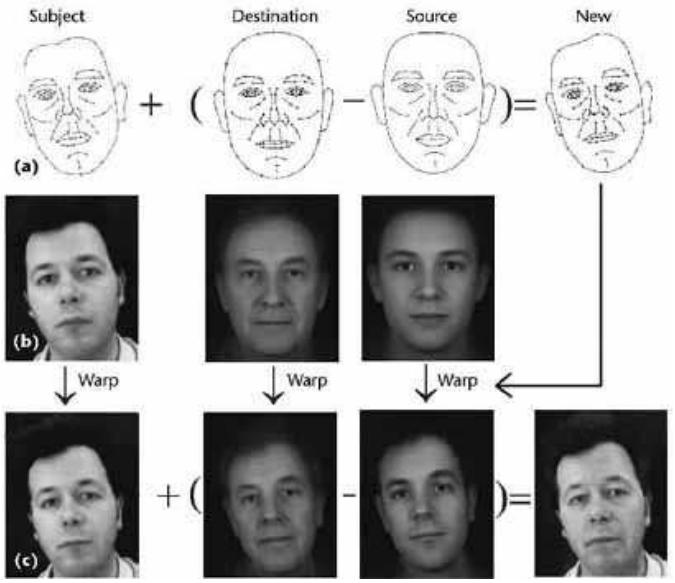

Figure 2.21: The shape and color transformation process: (a) Define new shape, (b) warp subject and prototype into new shape, and (c) transform colors at each pixel. [120]

where x_s is the original shape vector, x'_s is the transformed shape vector, and x_{p1} and x_{p1} are the normalized source and destination prototype shapes, respectively. ASM [22] are applied to place these feature points and warp the original image to the mean shape. A color model is then calculated by the mean color,

$$c'_s = c_s + (c_{p2} - c_{p1}) \tag{2.42}$$

where c_s is the warped image, c'_s is the final transformed image, and c_{p1} and c_{p1} are the warped source and destination prototype images, respectively. Figure 2.21 shows the transformation process of the shape and color from a younger adult (approximately 30 years old) to an older adult (approximately 60 years old). However, they point out that the wrinkles aren't captured in their prototype image. The texture-enhanced facial prototype is suggested by them. A wavelet-based method is early introduced in [114] for texture image. The usage of wavelet is to

adjust the amplitude of edges in the prototype image. A variety of different edge-detecting wavelet filters at different scales and orientations are applied in that paper. A locally weighted measure of the edge strength σ at each point (x, y) in a particular wavelet subband w is defined as:

$$\sigma_w(x, y) = h_x * h_y * \mid w(x, y) \mid \tag{2.43}$$

where h_x and h_y are the 1D cubic B-spline smoothing filers, used to filter ($*$ operator) along x and y axes, respectively. These values can be used to amplify the edges of the shape and color prototype $\overline{w}$ to have more representative local values:

$$v(x, y) = \frac{\overline{w}(x, y)\overline{\sigma}(x, y)}{\sigma_{\overline{w}}(x, y)} \tag{2.44}$$

where $\sigma_{\overline{w}}$ is the smoothed magnitude of subband w of the shape and color prototype at pixel (x, y) and v is the wavelet transform of the resulting texture-enhanced prototype. The texture-enhanced transformation process is shown in Figure 2.22. The experiment results show that the average perceived age for applying the texture-enhanced transformation are five times more effective.

Lanitis et al. [70] presented one of the first papers to focus on aging variation in a systematic way. A shape model and an intensity model are built for the approach; this idea is based on [69]. The principle component analysis is performed on these two models to reduce the data dimensionality. Detailed procedure and description are presented in [69]. The original image is first applied active appearance models (AAM) [32] to locate the landmarks for deforming the shape into the mean shape. The effect of the model parameters is shown in Figure 2.23. Aging functions $age = f(b)$ are aimed to isolate the aging variation. Three formulations: a linear, a quadratic, and a cubic are defined as

$$age = offset + w_1^T b \tag{2.45}$$

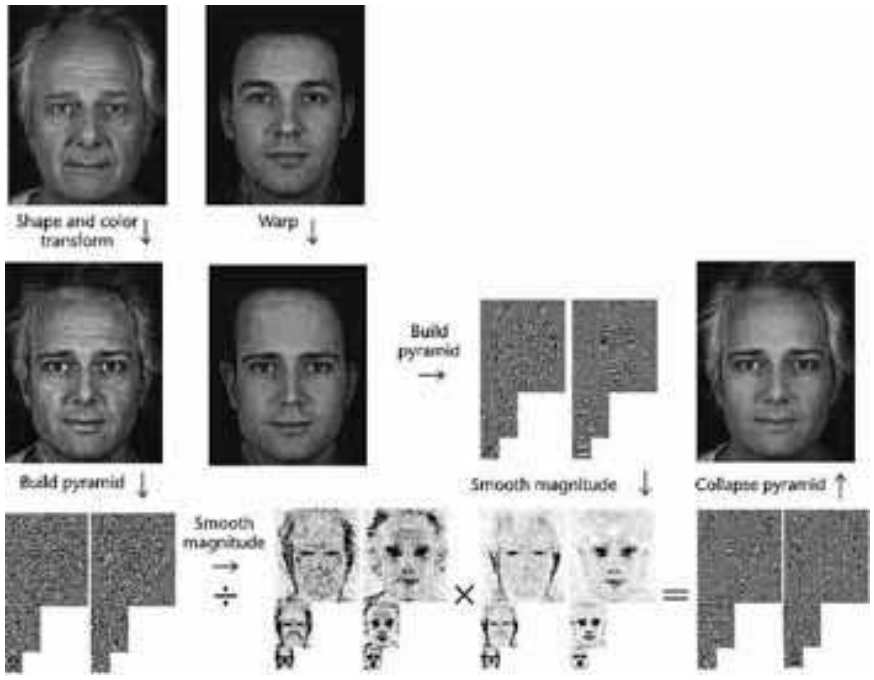

Figure 2.22: The texture-enhanced transformation process. Wavelet pyramids from these two images are built and the magnitudes are calculated. [120]

$$age = offset + w_1^T b + w_2^T (b^2) \tag{2.46}$$

$$age = offset + w_1^T b + w_2^T (b^2) + w_3^T (b^3) \tag{2.47}$$

where b^2, b^3 are vectors containing the squares and the cubes of the 50 raw model parameters, respectively, w_1, w_2, w_3 are vectors containing weights for each elements of b, b^2, b^3, respectively. Genetic algorithm is used to calculate the optimum parameters of the aging function to minimize the difference between the actual ages of training images and that of estimated one. By using the aging function, a set of raw model parameters can be converted to an estimated age:

$$b = f^{-1}(age). \tag{2.48}$$

The facial appearance of an individual can be simulated or eliminated for different ages. The new model parameters of the target

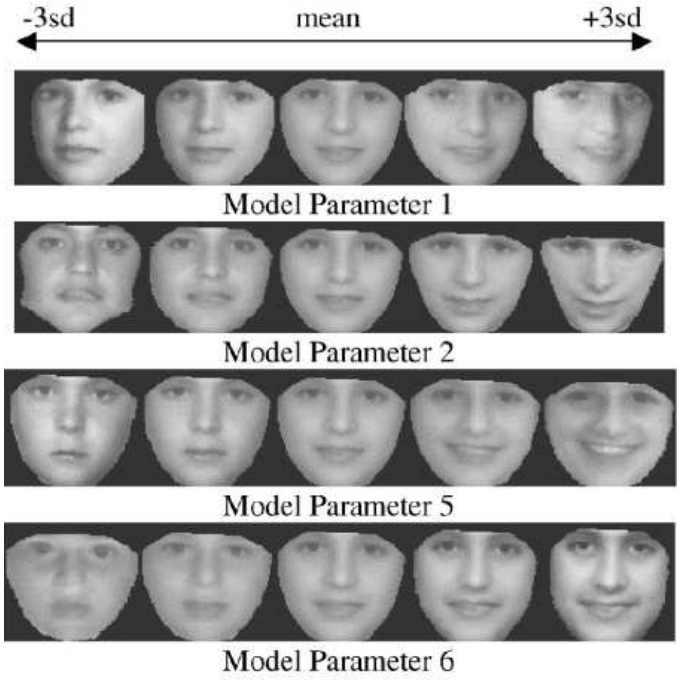

Figure 2.23: The effect of the must significant model parameter [70]

age is defined as:

$$b_{new} = b_{now} + \left[f_p^{-1}(age_{new}) - f_p^{-1}(age_{now})\right] \tag{2.49}$$

where b_{now} are the face parameters for the given face image, age_{now} is the estimated age of the subject in the image, age_{new} is the target age, b_{new} are the estimated model parameters. This approach can help to predict the facial appearance of wanted or missing persons for several years later. Also, the current records of the face database can be updated using the automatic age simulation.

2.3 Discussion about facial feature

Facial feature extraction method is the most important part for face recognition. In this section, the performance evaluation and the evolution of the face recognition are presented.

Table 2.4: The most famous face database for face recognition

Data set	Description
MIT Media Laboratory [122]	16 subjects with variants in scale, lighting and head rotation
ORL Database (AT&T)	10 different images of each of 40 distinct subjects
Yale Database	11 images per subject, 15 individuals, Images with expressions, glasses and different illumination changes
Yale Database B [42]	10 subjects each with 9 poses x 64 illumination conditions
FERET Database [102]	A large database that more than a thousand of subject and Images with different expressions, different pose and different lighting conditions
UMIST Database [45]	564 images of 20 subjects. Images with variant in poses from profile to frontal views, race, sex and appearance
University of Bern Database	300 frontal images and 150 profile images of 30 subjects
Purdue AR Database [87]	Over 4,000 color images of 126 subjects with different expressions, illumination conditions and occlusions
M2VTS Database [11]	37 different subjects and 5 multimodal shots for each subjects
The Extended M2VTS Database (XM2VTS)[91]	Four recordings of 295 subjects head shot with rotation, that taken over a period of four months. high quality colour images, video sequences and a 3d Model
University of Stirling Database	35 subjects with 3 poses and 3 expressions

Table 2.5: Other face database

Data set	Description
Harvard Database [51]	10 subjects each with 5 cropped, masked, single light source, frontal-view images
Shimon Edelman's Database	28 subjects each with at least 60 images, variants in pose, illumination and expression
The Japanese Female Facial Expression (JAFFE) Database [83]	10 Japanese female models with totally 213 images of 7 facial expressions
University of Oulu Physics-Based Database [85]	125 subjects of color images with variants in illumination, poses and with/without glasses
BioID Database [59]	Each one shows the frontal view of a face of one out of 23 different subjects. Some images contains manually set eye postions, totally 1521 images
CVL Database [1]	114 subjects each with 7 images
Max Planck Institute Database	7 views of 200 laser-scanned (Cyberware TM) heads without hair
Essex University Database	20 color images for each 395 subjects that variants in races, age, lighting condition

2.3.1 Performance evaluation for face recognition

It is important to evaluate the performance in terms of recognition rate using standard face database. Table 2.4 and Table 2.5 show some famous face databases for face recognition and the remaining face database, respectively. The resource paths

Table 2.6: The resource path of the face database for face recognition

Data set	Resource path
MIT Database [122]	ftp://whitechapel.media.mit.edu/pub/images/
ORL Database	http://www.uk.research.att.com/facedatabase.html
Yale Database	http://cvc.yale.edu/projects/yalefaces/yalefaces.html
Yale Database B [42]	http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html
FERET Database [102]	http://www.itl.nist.gov/iad/humanid/feret/
UMIST Database [45]	http://images.ee.umist.ac.uk/danny/database.html
University of Bern Database	ftp://iamftp.unibe.ch/pub/Images/FaceImages/
AR Database [87]	http://rvl1.ecn.purdue.edu/ aleix/ar.html
M2VTS Database [11]	http://www.tele.ucl.ac.be/PROJECTS/M2VTS/
XM2VTS Database [91]	http://www.ee.surrey.ac.uk/Research/VSSP/xm2vtsdb/
University of Stirling Database	http://pics.psych.stir.ac.uk/
Harvard Database [51]	ftp://ftp.hrl.harvard.edu/pub/faces
Shimon Edelman's Database	ftp://eris.wisdom.weizmann.ac.il/pub/FaceBase/
JAFFE Database [83]	http://www.mis.atr.co.jp/ mlyons/jaffe.html
Oulu Database [85]	http://www.ee.oulu.fi/research/imag/color/pbfd.html
BioID Database [59]	http://www.humanscan.de/support/downloads/facedb.php
CVL Database [1]	http://www.lrv.fri.uni-lj.si/facedb.html
Max Planck Institute Database	http://faces.kyb.tuebingen.mpg.de/
Essex University Database	http://cswww.essex.ac.uk/allfaces/

Table 2.7: Face Recognition results under different database

Method	Database - Recognition rate% (Dimension)					
	ORL	Yale	FERET⋆	UMIST	Bern	AR [87]
Eigenface [122]	93.1(22)	80.6(30)	50(25)	91.5(12)	100(20)	55.4(20)
Fisherface [7]	87.8(22)	99.4(15)	67(25)	94.8(12)	×	×
D-LDA [132]	93.7(22)	×	×	96.5(12)	×	×
DF-LDA [82]	95.7(22)	×	×	97.8(12)	×	×
EFC [77]	×	×	98.5(25)	×	×	×
ICA [5]	×	×	99.8(200)	×	×	×
DCT [33]	100.0	×	×	×	×	×
KPCA [63]	97.5(120)	×	×	86(12)	×	×
KDDA [81]	×	×	×	95(12)	×	×
EBG [126]	×	×	98	×	×	×
GFC [78]	×	×	95(25)	×	×	×
Line segments [28]	99.7∗	×	×	×	99.7∗	×
LEM [41]	×	85.4	×	×	100	96.4

Note: ∗= Combine the ORL and Ben database to form one larger database.
⋆= 600 FERET frontal face images corresponding to 200 subjects.

of each face database are given in Table 2.6. These benchmark data sets are constructed to evaluate different approaches in certain domains. We summarize the recognition results as shown in

Table 2.7. These results are getting from several papers [7], [82], [77], [33], [81], [63], [78], [5], [41]. The values inside the bracket mean that the number of feature selected for the testing. Although Table 2.7 shows the performance of these methods, we can not conclude the best representation method. There are several reasons that make the comparison become difficult.

1. The reported results are based on tuning parameters and data sets. We can observe that not all the approaches have been tested on the same data sets. Each face database has its own characteristic. The strength and weakness of different approaches can not be shown when testing on a small set of database. Besides, a set of experiments done by different set of parameters may give a variety of result. The number of features selected for the representation is an example of parameter.

2. The training and testing data may not perform the same pre-processing step before applying the methods. The most common step is to normalize the pixel values and to deform the image into pre-defined template such as the alignment of eyes. Different system architecture affects the recognition rate significantly.

3. The recognition rate may vary when different similarity measure functions are used. Cox et al. [24] show that the important of using a suitable classifier when using the same set of extracted features. Each approach has its own property, they may benefit on one similarity measure but not the others one. We can not make any assumption on this important factor when evaluate different representation methods.

Evaluation protocols such as Lausanne protocol and Brussels protocol have been used in face verification contest [89]. One of

the famous face recognition competitions FERET also provides a set of testing procedure to measure the algorithm performance [94], [102]. However, these kind of protocols are developed for performance measure of mature face recognition system. Researcher may have difficulty to perform the evaluation protocol for their prototypes. A standardized evaluation protocol should be well designed with clearly description to perform the evaluation. The description of the protocol must include how to find a representative universal test set. One suggestion [130] of evaluation on a small test set is that the test set is randomly chosen in each evaluation. Moreover, pre-processing steps are necessary to improve the quality of the images before extraction of the features. The training time and recognition time are usually ignored by most of the papers. A good evaluation protocol should keep in concern for these factors.

2.3.2 Evolution of the face recognition

After we have shown a number of different methods that have been applied on the face recognition task. Figure 2.24 shows the evolution of the face recognition methods for these ten years. We summarize the trend of the evolution into three main ideas.

1. Modify to kernel-based approach. The motivation of this modification is due to the non-linear characteristic of face problem. The comparison between linear and kernel based methods is being studied by [48]. The experimental results show that kernel-based method does not ensure better performance if the original input data is close to linearly separable. The linear-based extraction method may achieve similar result if the classifier is using a kernel function. This shows the characteristic of face is a non-linear problem and kernel-based approach is superior to linear-based method.

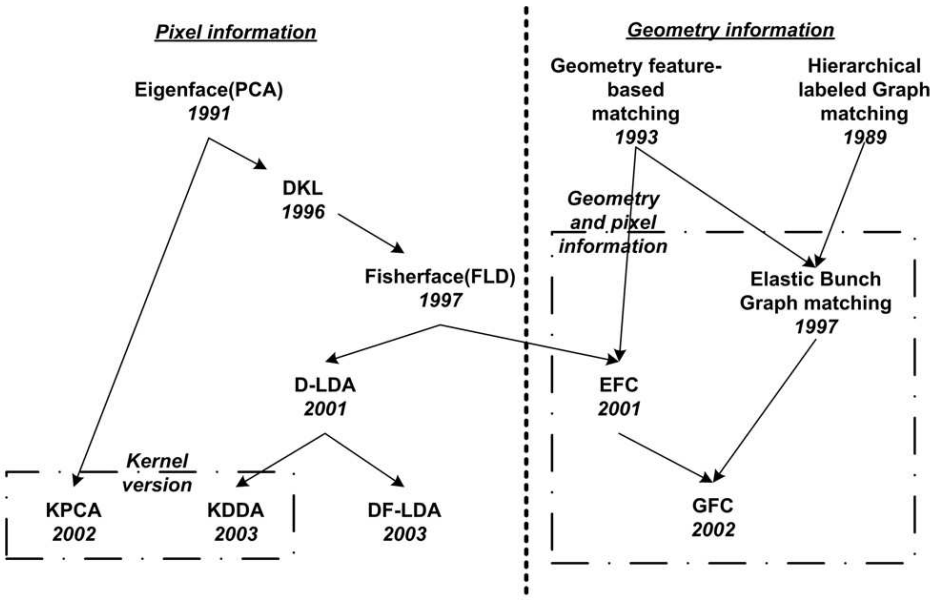

Figure 2.24: The evolution of the major face recognition approaches.

2. Adopt the deformable model. The deformable models are trained from examples that are superior to adapt a predefined template. The face image can first transform into a sharp-free image and perform the extraction method for further classification. This method can be more robust to variant in pose changes.

3. Combine the pixel information and geometry information. Researchers discovered that the geometry features contain discriminating power not less than the pixel information. They start to integrate both the pixel information and geometry information to improve the performance.

2.3.3 Evaluation of two state-of-the-art face recognition methods

There are two recently used techniques that give a great impact in the face recognition literature. The two techniques are Eigen-

Figure 2.25: Snapshot of the implementation of Eigenface (left) and Gabor wavelet (right)

face by Alex P. Pentland [122] and EBG by Laurenz Wiskott [126]. These techniques are recent and have apparently promising performances, and are the representative of new trends in face recognition. Evaluation of these two methods help to learn more about the literature.

The Eigen features and Gabor wavelet coefficients are tested in the experiment. Figure 2.26 shows the Gabor filters that applied in our experiment. The snapshot of the evaluation program is shown in Figure 2.25. The ORL face database is evaluated, that contains 40 subjects each of 10 images. In our experiments 15 eigenvectors are extracted for Eigenface and 12 fiducial points are selected for extraction of Gabor wavelet coefficient. The selected fiducial points (Figure 2.27) are:

- Eye bows (1, 2, 4 and 5)
- Pupil of eyes (3, 6)
- Nose (7, 8 and 9)
- Mouth (10, 11 and 12)

The images are divided into training set and recognition set. For the training set, the notation for the training list (ts**A**x**B**)

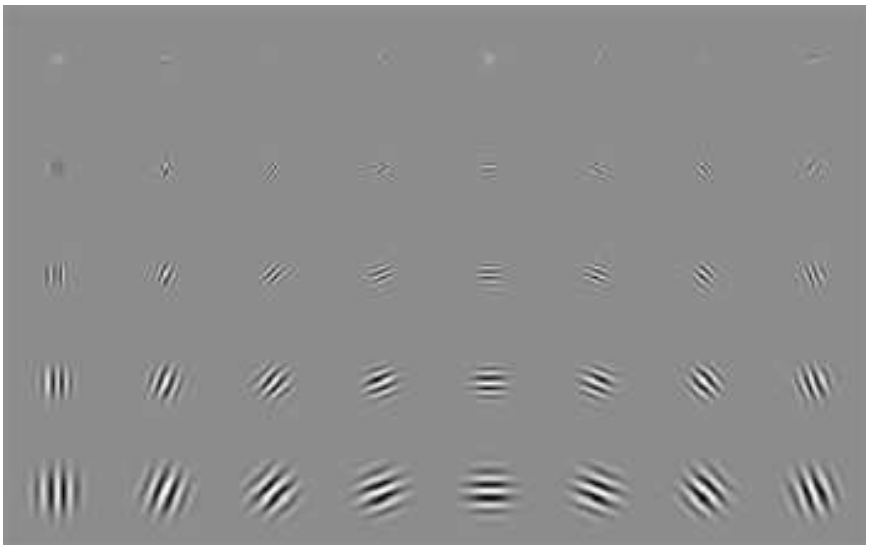

Figure 2.26: The real part of the Gabor filter with 5 frequencies and 8 orientations

means that the first **B** images of each of the first **A** subjects in the database is trained. For the recognition set, the notation for the recognition list (rs**A**_**B**) means that the **B** image of each of the first **A** subjects in the database are recognized. By using a series of training set and recognition set, different combination of the results are shown in Table 2.8 for Eigenface and Table 2.9 for Gabor wavelet coefficients.

For the ORL database, the Eigenface has a better performance (91.94%), rather than that of Gabor wavelet coefficients (80.92%). The result is not similar to that of the result from Jun Zhang [133]. There are some possible reasons for the results. First of all, the FBG formed doesn't have enough variation in the appearance of face. As the paper suggests including about 70 different variations include different age, gender and appearance. However, we have only 40 different subjects in the ORL database. Secondly, we only select 12 fiducial points for the experiments, the number of points is small when compare to the original approach. Moreover, faces are only compared with

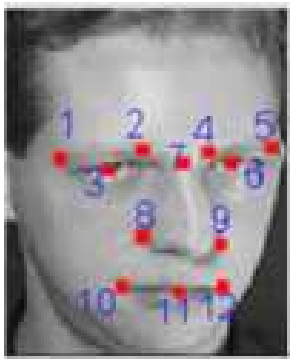

Figure 2.27: Example of selected fiducial points

Table 2.8: Results for Eigenface using 15 eigenvectors

	rs40_9			rs40_10			Average		
	Best	2nd best	3rd best	Best	2nd best	3rd best	Best	2nd best	3rd best
ts10x4	100	100	100	90	70	70	95	85	85
ts10x6	100	100	100	90	100	70	95	100	85
ts10x8	100	100	100	90	90	90	95	95	95
ts20x4	80	70	55	85	75	65	82.5	72.5	60
ts20x6	95	80	65	90	85	80	92.5	82.5	72.5
ts20x8	95	95	90	90	95	80	92.5	95	85
ts30x4	83.3	70	60	86.7	73.3	60	85	71.65	60
ts30x6	96.7	86.7	66.7	93.3	86.7	83.3	95	86.7	75
ts30x8	96.7	96.7	93.3	93.3	93.3	86.7	95	95	90
						Mean:	91.94	87.04	78.61

the Gabor wavelet coefficients while the edge information are omitted.

Eigenface is essentially a technique that use the minimum distance classifier, that is optimal if the lighting variation between the training set and recognition set can be modeled as zero-mean. It also success when the mean is nonzero but small. When the changes in lighting are large, the result will have a significant decrease in recognition rate. The reason is that the distance between two face images is dominated by the lighting factor rather than the differences between the two faces. If the pose varies, the training set needs other profile-view in order to recognize such poses. If the Eigenface is used in a practical

Table 2.9: Results for Gabor wavelet coefficient using 12 nodes

	rs40_9			rs40_10			Average		
	Best	2nd best	3rd best	Best	2nd best	3rd best	Best	2nd best	3rd best
ts10x4	90	90	50	70	60	40	80	75	45
ts10x6	80	90	60	70	70	60	75	80	60
ts10x8	90	80	70	70	70	50	80	75	60
ts20x4	85	70	60	75	60	50	80	65	55
ts20x6	85	85	65	80	80	65	82.5	82.5	65
ts20x8	85	95	75	80	80	65	82.5	87.5	70
ts30x4	80	60	53.3	83.3	60	50	81.65	60	51.65
ts30x6	83.3	80	63.3	86.7	83.3	50	85	81.65	56.65
ts30x8	80	93.3	76.7	83.3	86.7	73.3	81.65	90	75
						Mean:	80.92	77.41	59.81

system, scale, position and lighting conditions should be provided for the system to ensure high recognition rate. Eigenface can take the advantages of computational efficiency when the Eigenfaces are stored and the dimension of these vectors is not large.

EBG makes use of Gabor features, being the output of bandpass filters, and these are closely related to derivatives and are therefore less sensitive to lighting changes. Also, this approach contains features only at the key node of the image rather than the whole image. This can reduce the noise taken from the background of the face images. Together with other important advantage, it is relatively insensitive to variations in face position and facial expression. The matching procedure uses the FBG as a face template for finding the precise fiducial point, that solves the problem for automatically localization. The stored data can be easily expanded to a database for storage. When a new face image is added, no additional afford is needed to modify the templates, as it is already stored in the database. This advantage has overcome the Eigenfaces because of the recalculation of the projection.

2.4 Problem for current situation

Human visual system is the perfect face processing system that has a wide ranges of functionality. An ideal face processing system can retrieve all the features that can be captured by the human visual system. Different research work are developed based on our human visual system that is the ultimate goal of an ideal face processing system. The system is supposed to provide as much information as possible from the image or a sequence of images. There are several face processing fields that researchers currently focus on, include

- Face detection
- Face recognition
- Expression recognition
- Face tracking
- Age classification
- Gender classification
- Pose-estimation

Different techniques are developed to extract the features, and have been applied to different fields. The representation of the features is important for classification problem regarding face recognition. Several computer vision systems [99], [128], [53], [17] had integrated some of the face processing fields. There are two main types of face processing system, i.e., face recognition system for the security issue and human computer interaction system for people to communicate with the computer. The result of famous Face Recognition Vendor Test from FERET is released [101]. They conclude that those mature face recognition systems have weakness on outdoor performance and the

recognition rate decreases linearly when the elapsed time between the database and the new image increases. These show that systems are not robust to variants in lighting condition and aging effect. Although, there are various techniques to extract features. Integration of all techniques into the system is not yet success. These reduce the robustness of the developed systems due to various environmental changes. These variations include illumination changes, variant in poses, variant in facial expressions and partially occlusion of face, etc. The caricature of face is one of improvement for face recognition [23], [25]. Researchers [2], [86] start to tackle the problems and to develop an automatic face processing system.

We have summarized the facial feature from the past work into two main categories: geometry information and pixel information. Numerous new techniques have been developed to retrieve these features. The goals of this chapter are to summarize and to analyze the extraction techniques and the representation methods in various face processing fields. This helps us to know more about the strength and weakness of current technology and the general direction of development. However, various papers are evaluated using different face databases. The performance comparison among different approaches becomes difficult. Even if we have a standardized database for testing, different pre-processing steps and training procedures will affect the results. To deal with this problem, a standardized evaluation protocol should be designed to perform a subjective evaluation. The evolution of the face recognition are also presented to give the roadmap of the techniques. Human computer interaction system and automatic face recognition system are challenging work in pattern recognition area. Lots of research work have been focused on these problems. However, the developed systems work properly on the specify environment only. There are a wide ranges of improvement to increase the robustness of the

system.

□ **End of chapter.**

Chapter 3

Face Detection Algorithms and Committee Machine

Detecting face is the key factor for the success of the face processing system. There are many face detection approaches recently and are contrasted to the appearance-based methods. They rely on statistical analysis and machine learning to find the characteristics of face and non-face pattern rather than using a predefined template or rules. Turk and Pentland applied principal component analysis to face detection [122]. In [105], Rowley et al. use neural networks to learn face and non-face patterns for face detection. Support Vector Machines is used by [98] and demonstrated the success in detecting frontal faces. Roth et al. proposed Sparse Network of Winnows, SNoW algorithm [131], that proposes the primitive feature space for the learning process.

In recent years, an ensemble of classifiers has proven to provide a better estimator than the use of a single classifier. The basic idea for the committee is to train a set of estimators and combines the results to make a final decision. There are several approaches proposed by researchers in the literature such as ensemble of neural network by Hansen and Salamon [52], boosting by R. Schapire [108], gating network by Jacobs et al. [58], bagging by L. Breiman [12], hierarchical mixtures-of-experts by Jiang and Tanner [60] and designed committee by [18]. These approaches have been applied for financial prediction, e.g. [47]

[29], and face processing likes face recognition [50] and gender classification [49] but not yet in face detection.

In this chapter, an introduction about face detection is given. Moreover, Face Detection Committee Machine (FDCM) is proposed, that combines SNoW algorithm, SVM and NN as our experts. Reviews for these three methods and description of FDCM are presented. Yang et al. [130] point out that there is no standard method or procedure to evaluate various face detection approaches because papers use different training and testing data for their experiment, and different infrastructure to find faces in images. There is no comparative study on different face detection methods. Thus, we present a comparison on SNoW algorithm, SVM, NN and FDCM using experimental result. The evaluation shows that our machine performs better than the other three individual algorithms in terms of detection rate, false-alarm rate and ROC curves.

3.1 Introduction about face detection

Existing face detection techniques are summarized by [130]. There are four main categories of approaches, 1) knowledge-based, 2) feature invariant, 3) template matching and 4) appearance-based.

1. Knowledge-based. This approaches are based on the human rules derived from the researcher's knowledge of human faces. It defines simple rules to describe the features of a face and their relative positions or relationships. The main problem of this approach is difficult to translate the human knowledge to computer-coded rules. If the rules are too detailed, the system may fail to detect faces and causes many false negatives. If the rules are too general, it may give many false positives. When the face is in different

poses, the rule needs to be changed and this is challenging to enumerate all possible cases.

2. Feature invariant. Researchers find invariant features of faces for detection. They find out some properties or features that exist in different poses and lighting conditions, and use these features as the indicator of presence of face. The problem of this approach is that the image feature can be corrupted or changed due to illumination, noise, and occlusion.

3. Template matching. A standard face pattern is manually predefined. Correlation is used to compute the value between input image and the standard pattern. The higher the value, the higher the probability of existence of face. This approach is easy to implement but it has been proven to be inefficient to detect face with variation in pose, scale, and orientation. Thus, deformable templates have been used to achieve scale and shape invariance.

4. Appearance-based. For the template matching methods, experts need to predefine the templates, but the "templates" used in the appearance-based method are learnt from the examples in training images. This is a kind of statistical analysis and machine learning to find the characteristics of face and non-face images. The distribution of these facial features derived from the features form models or discriminant functions. The dimension of the feature vector is usually high, thus, dimensionality reduction is needed to increase the efficiency. To detect faces, there are two main stream of methods. One is the probabilistic framework, or maximum likelihood to classify a candidate image location as face or non-face. Another approach is to find a discriminant function between face and non-face classes.

This section provides an introduction on face detection. We can conclude that the appearance-based approach is superior than the other three approaches. In general, face detection is a two class recognition problem that choose the answer between face and non-face. The accuracy of the classification result are depended on the feature extracted from the training data. However, most of the papers have shown their experimental results on different test data. A subjective comparison is not possible by using the results shown in those papers.

In next section, we propose a committee machine for face detection. Three methods from the appearance-based category are chosen as the committee members. Moreover, we perform the evaluation among these three methods and the committee machine with the same training and testing data to have a subjective comparison.

3.2 Face Detection Committee Machine

For a traditional committee machine, it applies homogeneous experts (Neural Networks or Radial Basis Function) that trained by different training data sets to arrive at a union decision. However, no one has yet focused on heterogeneous experts on face detection problems in the current status. We propose the engagement of committee machine with heterogeneous experts. This is the first effort to employ different state-of-the-art algorithms as heterogeneous experts on committee machines in face detection. We include SNoW algorithm, SVM and NN in face detection to validate any possible face images. The use of committee machine can capture more features in the same training data and overcome the shortcoming of each individual approach. The reasons why we choose those three methods are that they are both statistics approach that operate with gray-scale images and can use the same training data to train each expert.

3.2.1 Review of three approaches for committee machine

Our committee machine consists of three approaches, a) SNoW algorithm by Roth et al. [131], b) SVM [124] and c) NN by Rowley et al. [105]. A brief introduction about these three approaches is given.

SNoW

Sparse Networks of Winnow algorithm is proposed by Roth et al. [131] for face detection. This is a learning approach for detecting faces using a network of linear units over a common pre-defined feature space. The input image is encoded into a feature space using the position and intensity values of pixels. The feature index is calculated by

$$256 \times (y \times w + x) + I(x, y) \tag{3.1}$$

where w is the width of the image, (x, y) is the coordinate of the pixel, $I(x, y)$ is the intensity of the pixel [0,255]. This representation can ensure one and only one feature is active for the position (x, y) with $I(x, y)$. This is so-called primitive features. Each input image is mapped into a set of features, that are active, and propagates to the target node. The input layer of the network represents simple relation for the input features, that are connected to a target node. Let $A_t = \{i_1, \ldots, i_m\}$ be the set of active features and are linked to the target node t. Then, the target node is active:

$$\sum_{i \in A_t} w_i^t > \theta_t \tag{3.2}$$

where w_i^t is the weight on the edge connecting the ith feature to the target node t, and θ_t is the threshold of target node t. Winnow update rule is used to update the set of weight w_i^t only

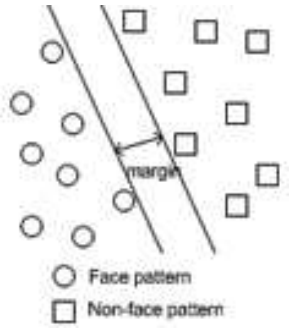

Figure 3.1: A hyperplane that separate two set of data

when have an error in prediction. It contains two steps,

$$w_i^t = \alpha * w_i^t \textbf{ for } \alpha > 1 \tag{3.3}$$

$$w_i^t = \beta * w_i^t \textbf{ for } 0 < \beta < 1 \tag{3.4}$$

where w_i^t is the weight, α is the promotion factor and β is the demotion factor. The promotion step (Eq. 3.3) is used for all the active features when the algorithm predicts 0 but the received label is 1. While the algorithm predicts 1 but the received label is 0, the demotion step is used (Eq. 3.4). The network is then trained repeatedly until the errors reduce to a pre-defined threshold.

SVM

Osuna et al. use the SVM classifier to detect faces [98]. This approach is to find out a hyperplane that can separate two classes to have the smallest generalization error. Moreover, the hyperplane leaves the maximum margin between the two classes, where the margin is the sum of the distances of the hyperplane from the closest point of the two classes (Figure **??**). The SVM classifier has the form:

$$f(x) = \textbf{sign}(\sum_i \lambda_i y_i K(\mathbf{x}, \mathbf{x_i}) + b) \tag{3.5}$$

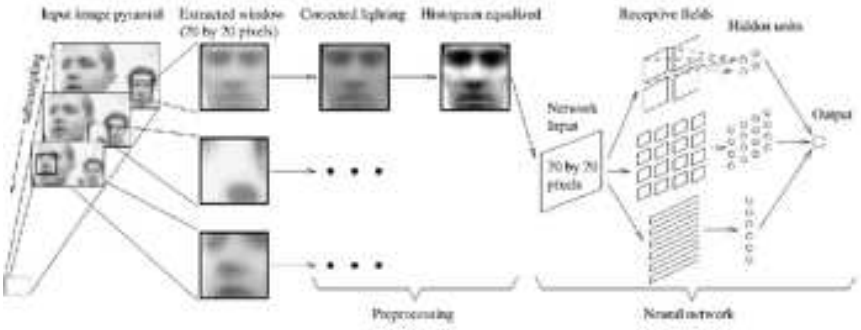

Figure 3.2: System diagram of Rowley's method. Each face is preprocessed before feeding it to neural network. [105]

where $K(x, x_i)$ is the kernel function. We can choose different function as the kernel function. In most cases, two classes can be separated using a linear function kernel. However, the problem of classification of face pattern cannot be solved well, we use polynomial kernel of second degree instead of linear kernel to solve the problem.

Neural network

Neural networks have been applied successfully in face detection. The most significant work is done by Rowley et al. [105]. They use a multi-layer neural network to learn the face and non-face patterns from images. In Figure 3.2, the first component of this method is a neural network that receives a 20×20 pixel region of an image and outputs a score ranging from [-1,1].

They use a number of simple processing elements that individually deal with pieces of a hard problem. Each processing element simply multiples inputs by a set of weights, and a nonlinearly transforms the result into an output value. The multilayer perceptron is commonly used architecture. Figure 3.3 shows the architecture, the left column is the input layer, the middle column is the hidden layer and the right column is the

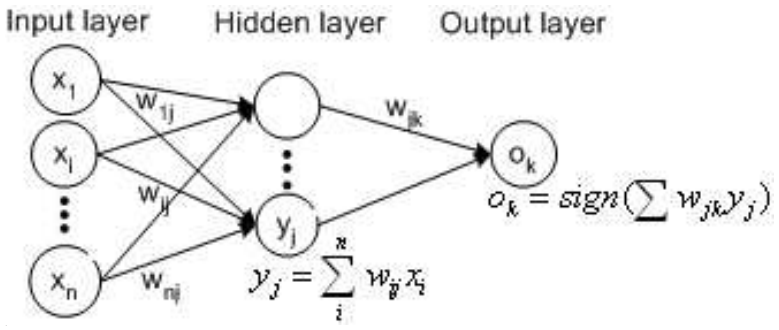

Figure 3.3: The architecture of the multilayer perceptron

output layer. The line connected between nodes represents the weighted connection. The output of the node is calculated by:

$$\sum x_{ij} * w_{ij} \tag{3.6}$$

where i is the input connection to node j, x_{ij} is the input value from i to j and w_{ij} is the weight of connection from i to j. The output of the network is compared with a desired response to produce an error. An algorithm called back propagation method is used to adjust the weights a small amount at a time to reduce the error. The network is trained by repeating this process many times. The goal of the training is to reach an optimal solution based on the performance measurement. Multi-resolution of the input image is used to detect different size of the faces and the arbitrator is used to merge the detection result.

3.2.2 The approach of FDCM

The FDCM works according to the confidence value T_i of each expert i. However, the confidence value T_i from each of the experts cannot be used directly. Figure 3.4 shows that the distribution of the confidence value of each expert varies in a large range. Thus, we need to normalize the value T_i by using the statistical information obtained from the training data:

$$\alpha_i = (T_i - \mu_i)/\sigma_i, \tag{3.7}$$

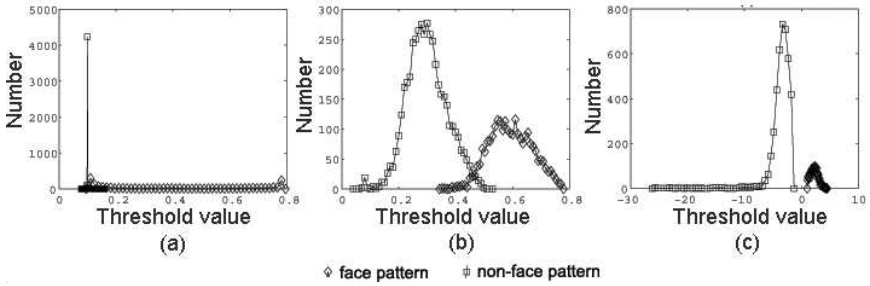

Figure 3.4: The distribution of confident value of the training data from (a) NN, (b) SNoW and (c) SVM

where T_i is the confidence value from expert i, μ_i is the mean value of training face pattern data from expert i, and σ_i is the standard derivation of training data from expert i.

One of the reasons why we need to normalize the confidence value is that they are not a uniform function. The confidence values that far away from the threshold means the stronger confidence for the decision. This information will be discarded if we first binarize the result to true or false. Another reason is that not all the experts have a fixed range of confidence value e.g., [-1,1] or [0,1]. Using statistical approach to model the problem, the information of confidence value from experts can be preserved. The output value of the committee machine can then be calculated using equation:

$$\beta = \sum_i w_i * (\alpha_i + \sigma_i * \delta_i), \tag{3.8}$$

where δ_i is the criteria factor for expert i and w_i is the weight of the expert i. The data is classified as face when the value of β is larger than 0 and non-face pattern when the value is smaller than or equal to 0.

3.3 Evaluation

We apply the CBCL face database from MIT for training and testing each of the expert systems. The training set of the database contains 2,429 face patterns and 4,548 non-face patterns, that are 19×19 pixels gray-scale and histogram normalized images. The training methods of three experts in the committee machine are explained below.

We have used the SNoW version 2.1.2 package for the implementation of SNoW approach and train the network with primitive features representation. For the SVM, the polynomial kernel of second degree is chosen to model the problem. A simple architecture for the neural network with back-propagation method is employed. All the input intensity values (19×19) are used as input units, that are connected to eight hidden units. Each hidden units are then connected to one output unit.

The experiment uses the same set of training and testing data to control the condition. We evaluate these three approaches and the committee machine without any difference in system setting. The testing set from CBCL face database contains 472 face patterns and 23,573 non-face patterns, that are 19×19 pixels gray-scale and histogram normalized images. Each testing pattern is passed into three experts and then the confidence value of the experts are passed into the committee machine.

The outputs from each single approach are determined by a threshold. When the threshold increases, the detection rate and the number of false detection will increase at the same time. For the FDCM, output value is calculated based on the experts' confidence values. When we change the value of criteria factor δ_i, the sensitivity of the committee machine will be affected. Input images are classified as face patterns will be increased if we increase the value of criteria factor δ_i. This property is shown in Fig. 3.5.

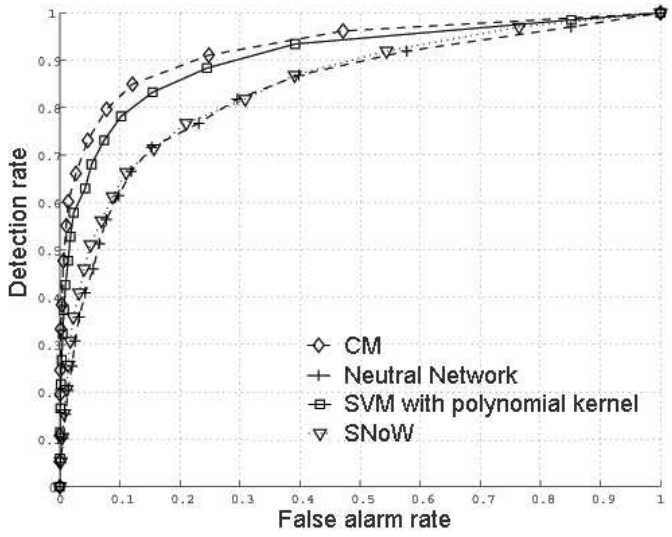

Figure 3.5: The ROC curves of three different approaches and the committee machine

The Receiver Operating Characteristics (ROC) curves are employed to show the characteristic of each approach. The area under the ROC curve provides a convenient way to compare classifiers. Table 3.1 shows the false alarm rate of each approach under the same detection rate. The false alarm rate of FDCM is nearly half of the other approach with 80% detection rate. FDCM archives lower false alarm rate than the other methods with the same detection rate. Table 3.2 lists the best operating point of each classifier. Our approach obtains the highest true positive rate while maintaining the lowest false positive rate. By observation, the statistical model of the committee machine can be engaged in the face detection problem and can improve the accuracy in classifying face and non-face pattern.

In summary, we summarize various face detection methods and propose a Face Detection Committee Machine, FDCM, using three face detection approaches, SNoW algorithm, SVM, and NN. It forms an expert system to improve the accuracy of classifying faces. The use of committee machine can capture more

Table 3.1: Experimental results on images from the CBCL testing set

	False Alarm Rate			
Detection Rate	**NN**	**SNoW**	**SVM**	**FDCM**
10%	0.56%	0.41%	0.05%	0.02%
20%	1.37%	1.09%	0.16%	0.07%
30%	2.54%	1.67%	0.44%	0.14%
40%	4.11%	2.92%	0.83%	0.41%
50%	6.32%	4.91%	1.60%	0.77%
60%	9.47%	8.47%	3.07%	1.41%
70%	13.89%	14.67%	5.98%	3.90%
80%	**26.97%**	**27.62%**	**12.32%**	**7.79%**
90%	48.95%	49.26%	28.60%	22.92%

Table 3.2: CBCL results

	True Positive	**False Positive**
NN	71.4%	15.2%
SNoW	71.6%	15.1%
SVM	81.2%	13.2%
FDCM	**84.1%**	**11.4%**

features in the same training data and can overcome the inadequacy of a single approach. We present the experimental results of committee machine using a data set of about 30,000 images and also have a comparison among three different approaches under controlled condition. The result shows our machine performs better than the other three individual algorithms in terms of detection rate, false-alarm rate and ROC curve.

□ **End of chapter.**

Chapter 4

Facial Feature Localization

Using statistical approach for face recognition can achieve high recognition rate and robust against different variation. However, the position of the facial feature point must be known before applying the statistical analysis like PCA or decomposition using wavelet. This shows the important role of facial feature localization. Although there are a number of facial feature detection algorithm, like genetic algorithm [73], their complexity is high and the effectiveness is not satisfied. In this section, we propose two algorithms to locate the facial feature points in gray-scale image and color image. For the gray-scale image, the template matching method and separability filter are applied. Color information and separability filter are combined for locating the feature points in the color image. Detailed description of the algorithms are presented in the following sections.

4.1 Algorithm for gray-scale image: template matching and separability filter

The proposed localization algorithm is based on the template matching method. Eye region is first located to reduce the search space for locating the eyes and the use of separability filter is to improve the accuracy of finding the position of iris. We assume the input image contains only one face with plain background, and faces are frontal-view and half-profile view.

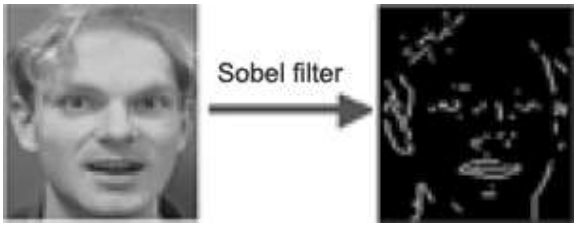

Figure 4.1: (a) Original image (b) Binary image after applied Sobel filter

Figure 4.2: Vertical projection of the binary image

4.1.1 Position of face and eye region

We focus on finding the facial feature points, thus, we simplify the segmentation to find the face candidate using the edge detection method. First, We apply the Sobel filter to the original image $I(x, y)$ and obtain a binary image $B(x, y)$, as shown in Figure 4.1. The horizontal projection is calculated to find out the left and right boundary of the head using equation:

$$V(x) = \sum_{x=0}^{N-1} B(x, y), H(y) = \sum_{y=0}^{N-1} B(x, y). \tag{4.1}$$

The position of left boundary xL, and right boundary xR are given by the smallest and largest values of x such that $V(x) \geq \frac{V(x_0)}{3}$ where x_0 denotes the column x with the largest $V(x)$. The upper boundary yU of the head is given by the smallest y such that $H(y) \geq 0.1(xR - xL)$ and the lower boundary yL of the head is given by $yU + 1.2(xR - xL)$. The extracted region is not the exact head region but all the feature points, such as eyes, nose and mouth, are included in this region.

After finding the head location, we use the vertical projection of the binary image to find the rough position of the eyes. In our experiment, we set the range of eye region to $\frac{1}{3}(xR - xL)$. Figure 4.2 shows the result of the selection. The extraction of the eye region is used to reduce the search space and to increase the accuracy of finding iris.

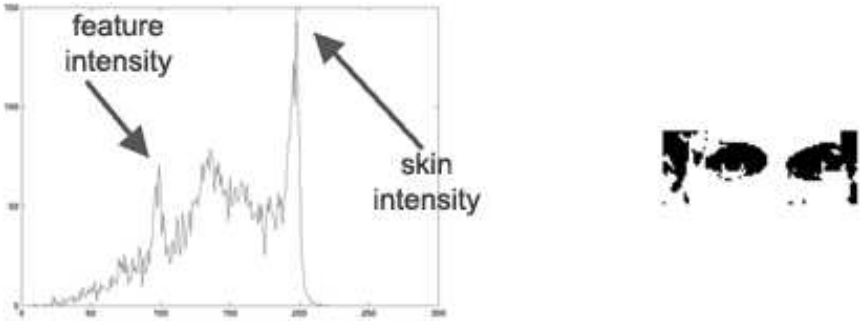

Figure 4.3: dynamic histogram thresholding technique. The histogram of the eye region (left) and the binary image using dynamic histogram thresholding technique (right).

4.1.2 Position of irises

A new binary image (Figure 4.3) is computed using dynamic histogram thresholding technique. In the histogram of the eye region (Figure 4.3), the first peak area represents the dark regions of the eye region, i.e. eyes, eyebrows. The other peak area represents the skin intensity. We choose a threshold between these two peaks for binarize the image. After choosing the threshold, we calculate the ratio of the black and white pixel (fill factor) of the new binary image and check whether this value is within an interval or not.

This technique is robust to different skin color. In Figure 4.4, comparison between three different binarization methods are shown. The standard binarization method using the fixed threshold (128 for gray-scale image) and the contour line of result image is not clear. After using the dynamic thresholding method, the contour line of two eyes can be seen. The eye and eyebrows can be clearly shown when the dynamic method with the fill factor is employed. This helps to determine the horizontal nose position that divides the eye region into two parts and each part contains one eye.

To find the horizontal nose position, the characteristic of the

Figure 4.4: An example of different thresholding technique. (a) Original image, (b) standard binarization method (c) dynamic thresholding method and (d) dynamic thresholding method with fill factor.

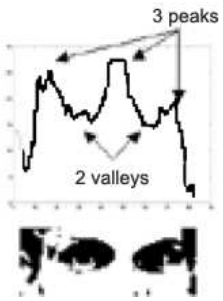

Figure 4.5: The projection of the binary eye region (upper) and the binary eye

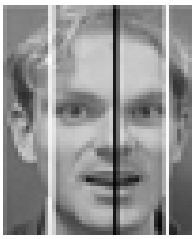

Figure 4.6: The result of the nose position. (Black line - final position of the nose, white line - the other two peaks)

horizontal projection of the binary eye region can be used. The projection of the binary image has three peaks and two valleys. The eye features form the two valleys and the peak between these two valleys is the horizontal nose position. The shape of the projection is shown in Figure 4.5. The middle peak of the projection is selected as the horizontal nose position (Figure 4.6).

Then, two eye templates are used (Figure 4.7) to compute the normalized cross-correlation. These eye templates are computed by averaging all eyes in the training set. The left part of the eye region is used to calculate the normalized cross-correlation

Figure 4.7: Eye templates used for normalized cross-correlation

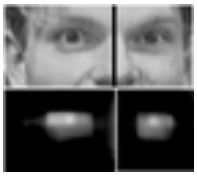

Figure 4.8: Result of two normalized cross-correlations

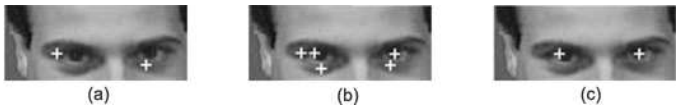

Figure 4.9: An example of eye candidates. The white cross in the image means (a) the maximum value of normalized cross-correlation, (b) the local maxima of normalized cross-correlation that greater than T_1, and (c) the minimum value of the cost function

with the left eye template. Figure 4.8 shows the result of the normalized cross-correlation.

The value of the result is ranged from [-1,1] where -1 means that the region and template are dissimilar and 1 means that they are the same. Let $r(i, j)$ be the value of the normalized cross-correlation value at pixel (i, j), the proposed algorithm selects all the local maxima whose value is larger than T_1.

$$T_1 = r_{\max} - 0.4 \times \sigma_r \tag{4.2}$$

where $r_{\max}$ is the maximum value of the normalized cross-correlation result in the region and σ_r is the standard derivation of the normalized cross-correlation result in the region. These selected candidates are called blobs. Figure 4.9 (b) shows the blobs extracted from the result.

If we only use the global maximum of the normalized cross-correlation value as the position of the iris, the result may be sensitive to the noise and yeilds an inaccuracy result, e.g. the white cross in Figure 4.9 (a). To make an improvement of the

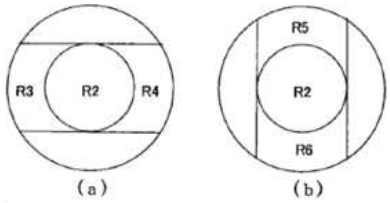

Figure 4.10: The templates used to compute $C_1(i)$ and $C_2(i)$

accuracy, we introduce a cost function C to evaluate the cost of each blob B_i. The idea of cost function is getting from [61] and separability filter is used. The equation of the separability [39] between regions R_1 and R_2 is:

$$\eta = \frac{\delta_b^2}{\delta_T^2}, \delta_T^2 = \sum_{i=1}^{N} \left(P_i - P_m\right)^2, \delta_b^2 = n_1 \left(P_1 - P_m\right)^2 + n_2 \left(P_2 - P_m\right)^2 \tag{4.3}$$

where $n_k\,(k = 1, 2)$ is the number of pixels inside R_k, N is the total number of pixels inside the union of R_1 and R_2, $P_k\,(k = 1, 2)$ is the average intensity inside R_k, P_m, is the average intensity inside the union of R_1 and R_2, and P_i is the intensity values of pixel i. Using equation (4.3), (4.4) and (4.5) to compute the cost of each blob:

$$C(i) = C_1(i) + C_2(i) + C_3(i) + \frac{1}{R(i)} \tag{4.4}$$

$$C_1(i) = \frac{|\eta_{23}(i) - \eta_{24}(i)|}{\eta_{23}(i) + \eta_{24}(i)}, C_2(i) = \frac{|\eta_{25}(i) - \eta_{26}(i)|}{\eta_{25}(i) + \eta_{26}(i)}, C_3(i) = \frac{U(i))}{U_{\mathbf{av}}} \tag{4.5}$$

where η_{xy} is the separability between region R_x and R_y, $U(i)$ is the average intensity inside B_i, U_{av} is the average of $U(i)$ over all blobs, and $R(i)$ is the value of the normalized cross-correlation result. We use the template of Figure 4.10 at the position (x, y) on $I(x, y)$ and then compute the separability $\eta_{23}, \eta_{24}, \eta_{25}$ and η_{26}.

After calculating the cost of all blobs, the position of the iris is selected by having the minimum cost value of the blob.

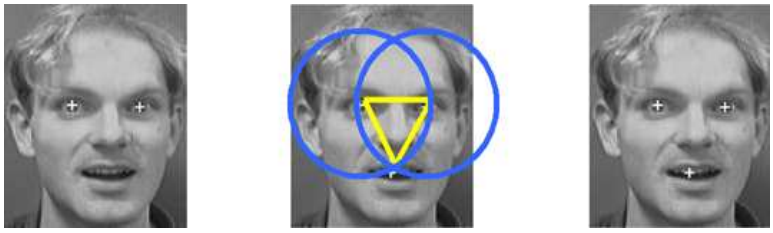

Figure 4.11: The estimation of the mouth position. Two irises are found using previous method (left), The geometry operation to locate the roughly mouth position (center), and the third point is the central mouth position (right).

In Figure 4.9(c), the two irises are chosen using the proposed method. This information is used to find the position of the mouth in the next section.

4.1.3 Position of lip

One of another important facial feature is the position of mouth. The positions of two irises are used to roughly locate the position of the mouth. Let d_e be the distance between two irises and d_m be the distance between the iris and the central part of the mouth. The distance d_m can use the aspect ratio of d_e and d_m (around 1.0-1.3 d_e) to estimate,

$$d_e \leq d_m \leq 1.3 \times d_e. \tag{4.6}$$

We set the distance d_m to be $1.1 \times d_e$ and use geometric operation to calculate the position of the central of the mouth $m(x, y)$ as shown in Figure 4.11. Then, a mouth region is extracted around $m(x, y)$ and the dynamic thresholding technique is applied to binarize the region (Figure 4.12). The minimum value of the projection of the binary image is the vertical position of mouth (Figure 4.13).

All the result found by the proposed algorithm is shown in Figure 4.13 (right), the white box is the eye region, two white

Figure 4.12: The calculation of the mouth position. The extracted mouth region (left), the binary image of mouth region (center), and vertical projection of the binary image (right).

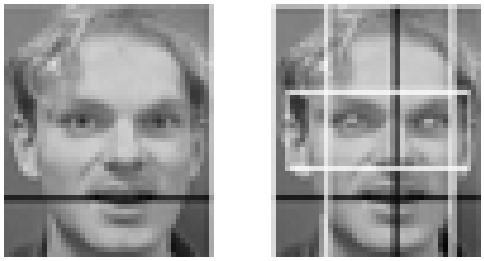

Figure 4.13: The location of different facial features. The vertical position of the mouth (left), and the entire facial feature selected by our algorithm include the two eye region, horizontal nose position and vertical mouth position (right).

crosses are two irises, the vertical black line is the horizontal nose position and the horizontal black line is the vertical mouth position. The exact position of the left corner, right corner and central part of the mouth is still in development. For the proposed algorithm, we focus on the execution time. The performance of this proposed algorithm is evaluated in the next section.

4.2 Algorithm for color image: eyemap and separability filter

We first apply the color information and separability filter to find the candidates of eyes and mouth. The color information gives an efficient way for finding the position of the eyes and mouth. The idea of the extraction algorithm is based on [55]. The separability filter is applied to construct a sepmap, which is then combined with the color information to increase the accuracy of choosing eye candidates. Each pair of eye candidates and one mouth candidate are then combined to form a face candidate. The final result of the feature points are selected by verifying each face candidate.

4.2.1 Position of eye candidates

The characteristic of chrominance components of the eye shows that high C_b and low C_r values are found around the eyes. We find the chrominance eyemap EC by

$$EC = \frac{1}{3}\left\{(C_b)^2 + (255 - C_r)^2 + (\frac{C_b}{C_r})\right\} \tag{4.7}$$

where C_b and C_r are the two chrominance components of YCrCb color space, $C_b^2, (255 - C_r)^2, C_b/C_r$ are normalized to the range of [0,255]. The construction of the chrominance eyemap and luma eyemap are shown in 4.14 and 4.15, respectively.

For the luminance component, morphological operations are performed on the Y channel to find the luma eyemap EL,

$$EL = \frac{(Y \oplus F)}{Y \ominus F + 1} \tag{4.8}$$

where Y is the luminance component of YCrCb color space, F is an ellipse structuring element, and $\oplus$ and $\ominus$ are the gray-scale dilation and erosion operations, respectively.

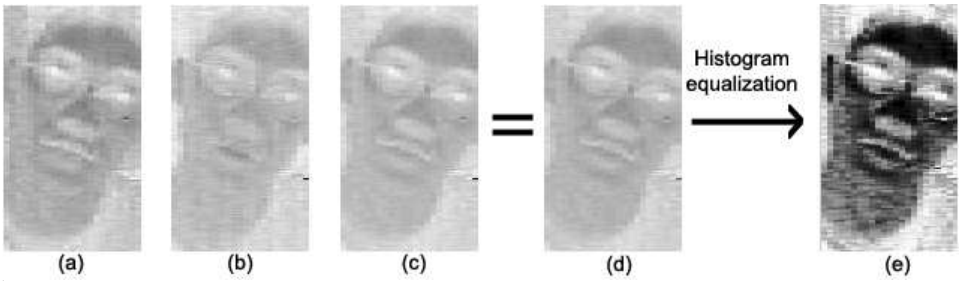

Figure 4.14: The construction of EC: (a)C_b^2, (b)$(255 - C_r)^2$, (c)$\frac{C_b}{C_r}$, (d) chrominance eyemap EC, and (e) EC after histogram equalization.

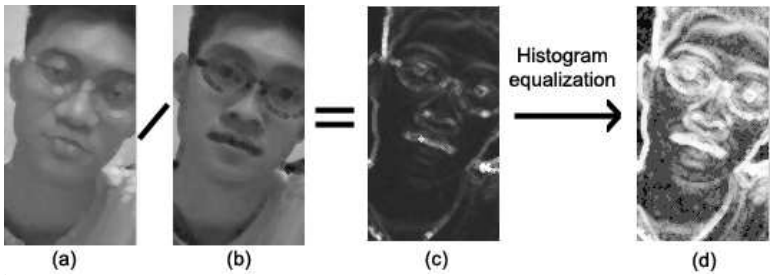

Figure 4.15: The construction of EL: (a)Upper part of the luma eyemap, (b)loweer part of the luma eyemap , (c) the luma eyemap EL, and (d) EC after histogram equalization.

Separability filter is applied to the luminance component. The description of the separability filter is given in previous section. The coefficients of separability filter form the sepmap SEP, that are calculated by equation (4.3) and (4.5). After finding two eyemaps (EC and EL), histogram equalization are performed to normalize the values of these two eyemaps. The resultant eyemap EM is calculated by the formula,

$$EM = ((EC \text{ AND } EC) \text{ AND } EL) \text{ AND } SEP \qquad (4.9)$$

where AND is the multiplication operation. The local maximum values of the EM are chosen as the eye candidates. The calculation of the resultant eyemap is shown in Figure 4.16.

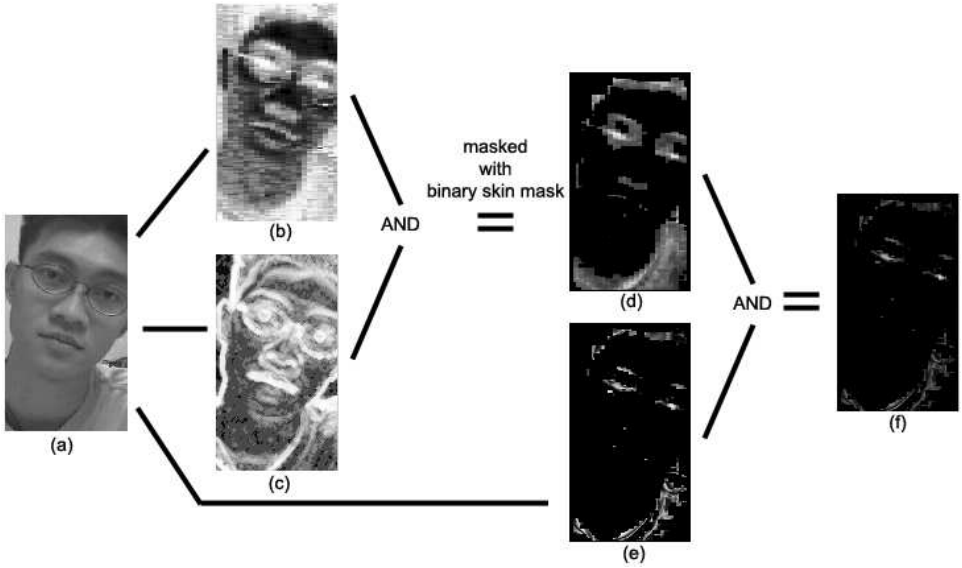

Figure 4.16: The calculation of EM. (a) The original image, (b) EC after histogram equalization, (c) EL after histogram equalization, (d) the masked intermediate image of ((a) AND (b)), (e) the sepmap SEP, and (f) the resultant eyemap EM.

4.2.2 Position of mouth candidates

After the eye candidates are found, mouth candidates are chosen from the mouthmap using the method proposed by [55]. The color of the mouth is red in color, that contains high C_r and low C_b values of chrominance components. The mouthmap MM is constructed as follows:

$$MM = C_r^2 \cdot (C_r^2 - \epsilon \cdot \frac{C_r}{C_b})^2 \tag{4.10}$$

$$\epsilon = 0.95 \cdot \frac{\mu_{C_r^2}}{\mu_{\frac{C_r}{C_b}}} \tag{4.11}$$

where C_r^2 and C_r/C_b are normalized to the range [0,255], $\mu_{C_r^2}$ and $\mu_{\frac{C_r}{C_b}}$ are the mean value of C_r^2 and $\frac{C_r}{C_b}$ inside the face mask, respectively. Figure 4.17 shows an example of the mouthmap.

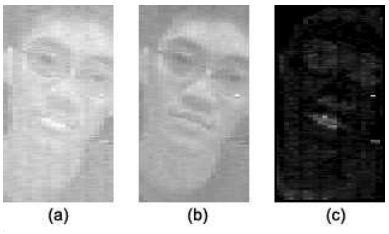

Figure 4.17: The calculation of MM. (a)C_r^2, (b)$\frac{C_r}{C_b}$, and (c) Mouthmap (MM)

The mouth candidates are selected by each local maximum in the MM.

4.2.3 Selection of face candidates by cost function

Each pair of eye candidates and one mouth candidate are combined to form a face candidate. Each face candidate is first verified by the geometry constraint such as the minimum distance between two eyes, and the ratio of the distance between eyes and mouth. This verification step can eliminate most of the impossible face candidate. Afterward, the cost of each eye candidate is calculated,

$$C(i) = EM(x,y) + SEP(x,y) + U(x,y) + V(x,y) \qquad (4.12)$$

where i is the index of eye candidate, (x,y) are the coordinate of the eye candidate i, EM is the value of eyemap, SEP is the value of sepmap, U is the average intensity of the blob and V is the sum of the value of the blob in EL.

For each face candidate, the original image is normalized by the position of two eye candidates to form a 19×19 image $I_N(i,j)$. These include rotation, re-scaling and histogram equalization operations. The face candidate is then evaluated by a cost function, that consists of different factors such as the confidence value of face detection classifier and the symmetric

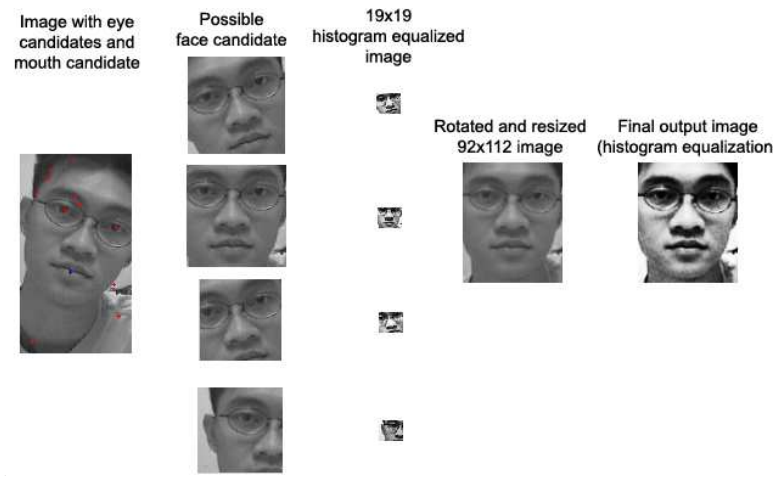

Figure 4.18: The selection of face candidate and the final output image. Red cross is eye candidate and blue cross is mouth candidate.

property,

$$C(i,j,k) = C(i) + C(j) + SVM(i,j) + SYM(i,j) \qquad (4.13)$$

where i and j are the index of two eye candidates, k is the index of mouth candidate, $C(i)$ is the cost of eye candidate i, $SVM(i,j)$ is the confidence value of the SVM classifier using input image $I_N(i,j)$, and $SYM(i,j)$ is the coefficient of the normalized cross-correlation of the image $I_N(i,j)$ and the mirror of the image $I_N(i,j)$. The location of the eyes and mouth are then determined by the face candidate having the maximum cost value. Figure 4.18 illustrates the selection among the face candidates and the final normalized face image. The advantage of this method is the location of the eyes can be found inside a non-upright face image without the predefined size.

4.3 Evaluation

In this section, we will have the performance evaluation for two facial feature localization algorithms. One gray-scale face

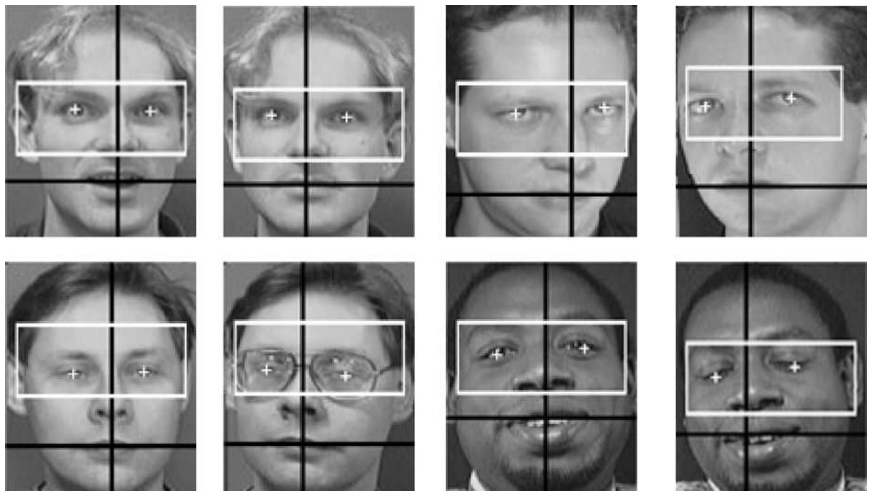

Figure 4.19: Experimental results of the localization algorithm. Four different people with different poses, skin intensities, open/closed eyes and with/without glasses

database and one color face database are used for the experiments.

4.3.1 Algorithm for gray-scale image

We have implemented the proposed algorithm on a SUN Ultra 5/400 workstation under Unix environment. Figure 4.19 shows some of the experimental results. Moderate subject size database, Olivetti Research Lab, is used. There are 40 subjects, each with 10 variations to form a set of 400 images. All the images have the same size (92×112 pixels), contained frontal-view and half-profile view. We can locate both of the irises and the horizontal position of the mouth precisely using the proposed algorithm. The face images are varying in poses, lighting, facial expressions, and facial details, such as with and without glasses, open and closed eyes.

We have used different cost functions to find the position of irises and calculate the Euclidean distance between the position

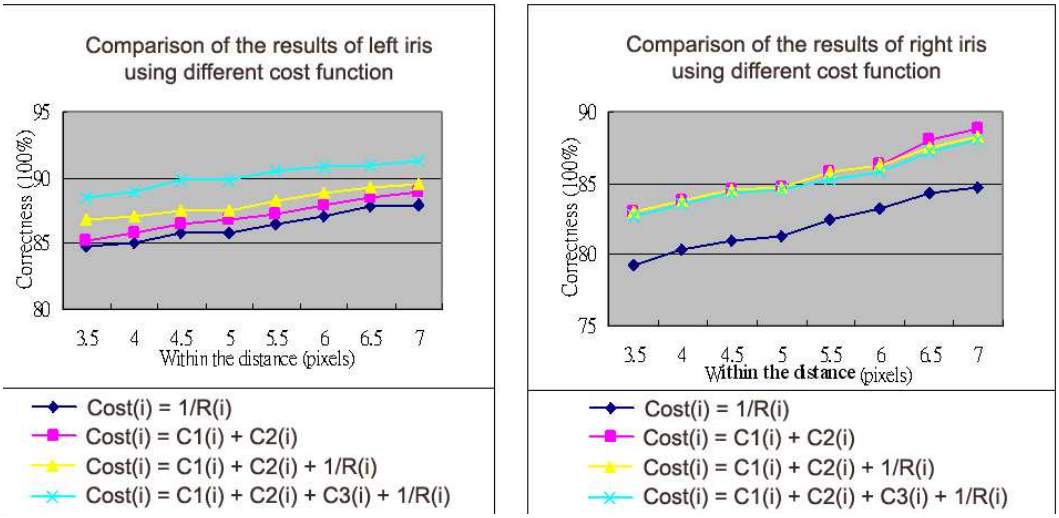

Figure 4.20: Comparison of the result of left/right iris using different cost function.

found by the proposed method and manually selected position from the image. Figure 4.20 shows the correctness of the left iris and right iris. By observation, the cost function using both the normalized cross-correlation and separability obtains the highest correction rate. We can conclude that the separability can help to increase the accuracy of finding the position of iris. To use the separability, we need to set the radius of the template (Figure 4.10) in the calculation. We have set the radius of the template from the range of [2, 7] and apply the algorithm. Increasing in radius will increase the execution time significantly, however the correctness of the result will not be better. From the results (Figure 4.21), we know that radius 3 have the best performance because the radius of the iris in the input image is about 3 pixels.

Table 4.1 shows the results of the experiments. The execution time of the proposed algorithm is about 0.5s. Our proposed method is faster than the original template matching method and another method that use Hough transform and separability filter [61]. Facial points such as eye, eye bows, nose and mouth

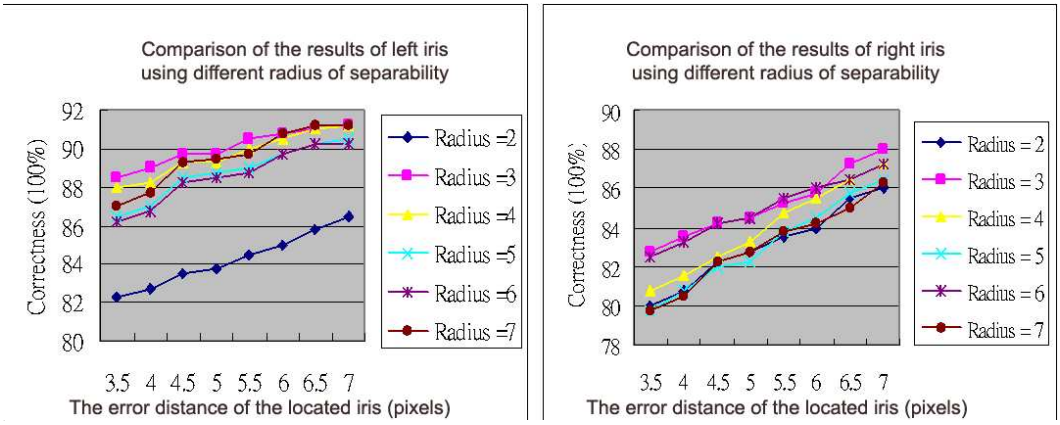

Figure 4.21: Comparison between the separability using different radius.

Table 4.1: The comparison of the performances of the proposed method, the template matching and the method using Hough transform and separability

Algorithm	Correct rate(%)	CPU time
Proposed method	**88.5**	0.5
Proposed method without separability	84.75	**0.3**
Template matching	63.75	0.6
Hough transform and separability filter method	56.5	19.3

take an important role for face recognition. We propose an algorithm that is based-on template matching and the separability filter to extract the feature points in a fast manner.

4.3.2 Algorithm for color image

The previous gray-scale face database lack of color information and can not be used for the evaluation. Thus, we have chosen the AR color face database for the experiments. There are about 130 different subjects and each subject contains two sets of images that are taken in two sessions. We select 4 subsets of the images of each subjects for the test, that are varying in poses, facial expressions, appearance with and without glasses, open and closed eyes. Figure 4.22 shows some example of set of

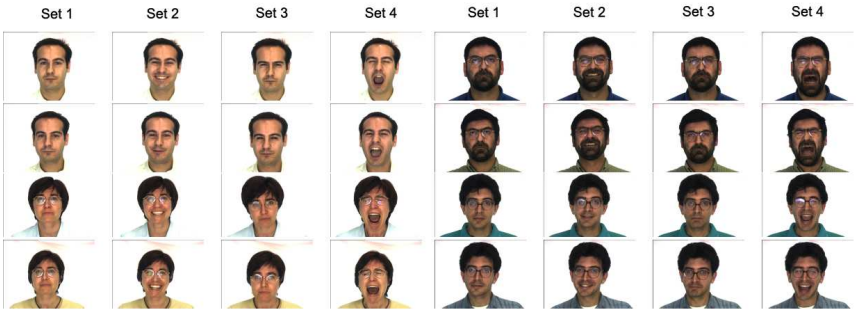

Figure 4.22: Examples of set of images for testing.

Table 4.2: The results of AR face database

	w and w/o glasses			w/o glasses		
	Set 1-4	Set 1-3	Set 1	Set 1-4	Set 1-3	Set 1
# of images	1,020	765	255	736	552	184
# of errors	220	124	26	97	48	6
Correct rate(%)	78.4	83.8	89.8	86.8	91.3	**96.7**

images. The description of each subset of images is as follow:

1. Neutral expression
2. Smile
3. Anger
4. Scream

This algorithm will be used to normalize a face image into an upright image with 112×92 pixels. The upright face that contains two eyes and one mouth is classified as correct extraction.

The algorithm is evaluated with different experimental setup. We have divided the face images into two groups, image with glasses and image without glasses. Figure 4.23 shows some examples of the correctness of the result image and The experimental results with these two setting are shown in Table 4.2.

The correctness of the face with neutral expression can obtain nearly 90%. The accuracy of the algorithm decreases when the

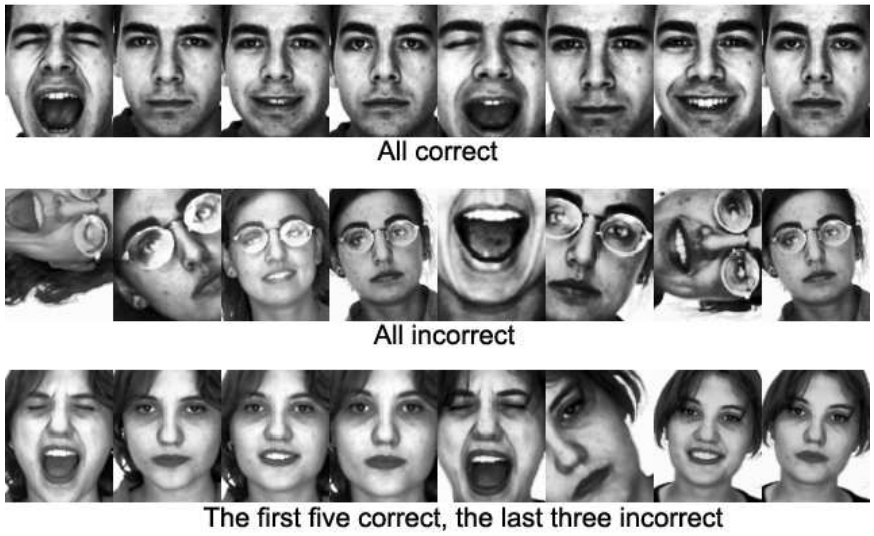

Figure 4.23: Examples of the correctness of the images.

Figure 4.24: Example of images with glasses.

variation of the image increases. Most of the images in set 4 are closed eyes, the information from the luma eyemap is reduced. This results in decreasing the correct rate to 78.4%. Besides, we find that lots of images with glasses have strong reflected light around the eyes (Figure 4.24). The values of chrominance components are unusual around the eyes. These affect the accuracy of locating the eye candidates. Therefore, we select all the images without glasses for another testing. The results are shown in the right-hand side of the Table 4.2. We see that the accu-

racy increases by 6-7% for all the three results when compared with the former one. This shows that the reflected light from the glasses degrades the performance of the algorithm. For the image with neural expression, the correctness of the localization algorithm can achieve **96.7%**.

In this chapter, we have proposed two facial feature localization algorithms. For the gray-scale image, the eye region is first located to reduce the search space for locating the eyes and the separability filter is applied to increase the accuracy of locating eyes. For the color image, the separability filter is applied to construct a sepmap. This sepmap is then combined with the color information to increase the accuracy of choosing eye candidates. Moreover, the location of the eyes can be found inside a non-upright face image without the predefined size. Evaluation of our proposed algorithms are given in this chapter.

□ **End of chapter.**

Chapter 5

Face Processing System

We have investigated several technologies for face processing that include face detection, face recognition and feature extraction. An integration work for different technologies is needed to develop a face processing system. In this chapter, we first introduce the purpose and the architecture of our system. There are some limitations that restrict our system development. Explanation and solution for the problems are discussed. Furthermore, detailed implementation for each module is described.

5.1 System architecture and limitations

We have developed a face processing system that detects and tracks a face automatically. The purpose of the system is to identify the person who sits in front of a web camera. The system consists of four main components: 1) Pre-processing, 2) face detection, 3) face tracking, and 4) face recognition. The system architecture is shown in Figure 5.1. The raw image data is first pre-processed to find out all the face candidates. Presence of face is known in the face detection module using the SVM classifier. Face tracking is then performed instead of face detection in consecutive frames to improve the efficiency of locating the position of the detected face. Face recognition is applied to identify the person.

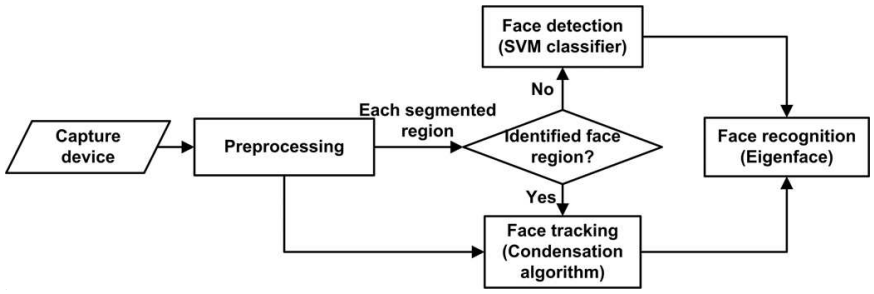

Figure 5.1: System architecture of the face processing system

For an online system, the process must finish within a fixed time interval. The execution time for each module is concerned. We have a trade off between the performance and accuracy. The main problem we faced is the processing time of face detection module. Although we have proposed a Face Detection Committee Machine that improves the accuracy of classifying the face pattern, we cannot choose the proposed method for the detection module due to the execution time of the FDCM. To solve this problem, we take the advantage of using color information to process the input data and then the SVM classifier is employed in the face detection module. Detailed descriptions about each module are presented in the following section.

5.2 Pre-processing module

This module is applied to reduce the noisy components from the capture device, and to reduce the search space of face detection and face tracking. In the pre-processing module, the system is activated by motion detection method, and the skin segmentation is applied to find out all the possible face candidate regions. The flow of the pre-processing module is shown in Figure 5.2.

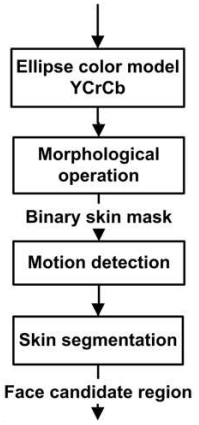

Figure 5.2: The flow of the pre-processing module

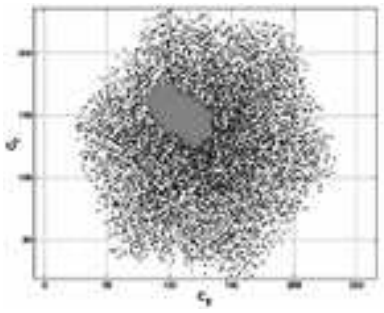

Figure 5.3: 2D projection in the CrCb subspace (gray dots represent skin color samples and black dots represent non-skin tone color)[55]

5.2.1 Ellipse color model

Firstly, image obtained from the capture device or video file is transformed from RGB color model to YCrCb color model. The luminance component (Y) is used for the face detection and chrominance component (Cr and Cb) are used in other modules. In various studies, human skin color is not a uniform distribution around the whole color space. Human skin color is one of the effective features to detect face in a complex background. Although different people have different skin color, there are several papers shown that the major difference is varies in intensity but not their chrominance. There are various color spaces have been used, such as RGB, normalized RGB, HSV, YCrCb, YIQ, etc. Kjeldsen and Kender defined a color predicate in HSV color space to separate skin regions from background [65]. Most of human skin tone can be represented by an elliptical equation in the YCrCb color space (Figure 5.3) [55]. We employ the ellipse color model to locate all the flesh color to form a binary mask. Morphological operation is then applied to the binary mask to form a new binary mask. This step can reduce noisy components especially for images taking from the web camera.

Figure 5.4: The motion detection step: (a) The image at time-step t, (b) the image at time-step $t+1$, (c) the binary skin mask of (a), (d) the binary skin mask of (b), and (e) the motion map.

Motion detection

Motion detection has been used to detect the human face by [62]. We apply this technique to detect the moving segment for further processing. Once the system runs, the motion detection starts to detect any movement of skin color between two images. The two images first employ the ellipse color model to form two morphological binary masks. Let $B_t(x, y)$ and $B_{t+1}(x, y)$ be the morphological binary masks at time-step t and $t+1$, respectively. The motion map $M(x, y)$ is defined as

$$M(x,y) = \begin{cases} 0 & \text{if } B_t(x,y) = 0 \text{ and } B_{t+1}(x,y) = 0 \\ 1 & \text{if } B_t(x,y) \neq B_{t+1}(x,y) \\ 2 & \text{if } B_t(x,y) > 0 \text{ and } B_{t+1}(x,y) > 0 \end{cases} . \quad (5.1)$$

Figure 5.4 shows the process of motion detection. For the motion map, the values of 0, 1, 2 are represented by black, white and gray in Figure 5.4(e), respectively. The moving segment is found by counting the pixels within the boundary of the white region only.

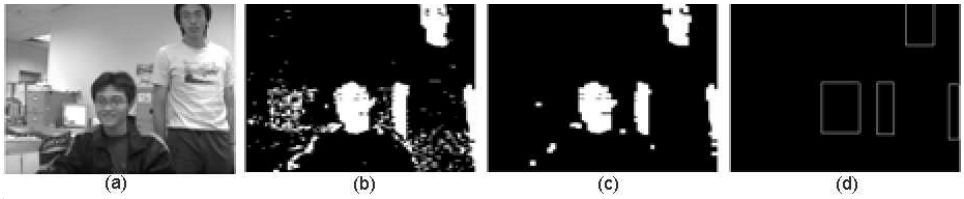

Figure 5.5: Skin segmentation step: (a) original image (b) binary skin mask, (c) binary skin mask after morphological operation, and (d) face candidates

Skin segmentation

The skin segmentation is performed on the image by applying the new binary mask to find out the face candidates regions from the image. The skin segmentation steps are shown in Figure 5.5. This module uses the color information to reduce the search space for finding face candidates. New face candidate regions are passed into the face detection module.

The color model helps to detect the skin color like region that can be applied for motion detection and segmentation of face candidate region. Motion of skin color triggers the system to have further processing.

5.3 Face detection module

After the face candidates are found, the candidate regions are verified by face detection module.

5.3.1 Choosing the classifier

Besides the accuracy for detecting faces, execution time is an important factor for choosing a suitable model for the system. Each model has different execution time, detection rate and false alarm rate. To choose the most suitable model for the system, we need a performance evaluation for each model.

Table 5.1: Performance evaluation for different classifiers using CBCL testing set (24,045 patterns)

	(a)	(b)	(c)	(d)	(c)+(d)
Method	**Time(s)**	**False alarm rate**	**Normalized (a)**	**Normalized (b)**	**Performance**
NN	**35.5732**	26.97%	-0.8517	0.6813	-0.1703
SNoW	63.6025	27.62%	-0.2889	0.7490	0.46018
SVM (polynomial)	65.2234	12.32%	-0.2563	-0.8440	**-1.1002**
SVM (linear)	61.1298	27.43%	-0.3385	0.7292	0.3907
FDCM	164.3991	**7.79**%	1.7352	-1.3156	0.4196

We first calculate the average execution time for classifying ten trials of 24,045 testing images, note that the time for loading the image file and data file is excluded. The statistics are shown in Table 5.1. Most of the models consume around 63 seconds to finish the testing process except the Neural Networks, that uses around 36 seconds only. The false alarm rates of each model with the same detection rate are selected for the calculation of the performance. The performance of each model is defined as

$$Performance_i = \frac{t_i - \mu_t}{\sigma_t} + \frac{f_i - \mu_f}{\sigma_f} \tag{5.2}$$

where t_i and f_i are the execution time and the false alarm rate of classifier i, respectively, μ_t and σ_t are the mean and standard derivation of the execution time of five classifiers, respectively, and μ_f and σ_f are the mean and standard derivation of the false alarm rate of five classifiers, respectively. The score of the SVM model with polynomial kernel of second degree obtain the lowest score (-1.1002). Thus, the SVM model with polynomial kernel of second degree is chosen as the face detection classifier for our system.

5.3.2 Verifying the candidate region

To verify the candidates, the SVM classifier is applied for detecting a face in the region. The whole set of training and testing

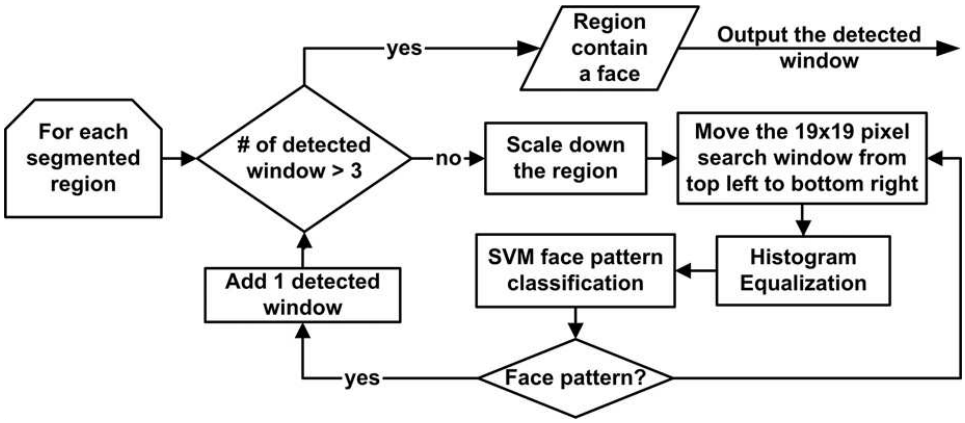

Figure 5.6: The flow of the face detection module

image of the CBCL face database are used for training the SVM classifier, that totally includes 2,901 face patterns and 28,121 non-face patterns. The flow of the face detection module is shown in Figure 5.6. Each extracted candidate region is resized to various scales to detect different size of faces. A 19×19 search window is searching around the regions from top left of the resized region to bottom right. Histogram equalization is performed on the window, that compensates the camera variations such as lighting variations. Figure 5.7 shows an example of re-scaling and histogram equalization.

Each search window is classified into face pattern or non-face pattern using the SVM classifier. The scale down approach can help to avoid the process of excess search window. For the same size of region, detecting a large face processes a smaller number of search window than detecting a small face. The number of detected face pattern is accumulated to reduce the false detection of the system. We only accept the region containing a face when the number of classified face pattern is more than three. After the first presence of a face at the larger scale, the pixels in the detected window are extracted and then pass into the face tracking module or face recognition module.

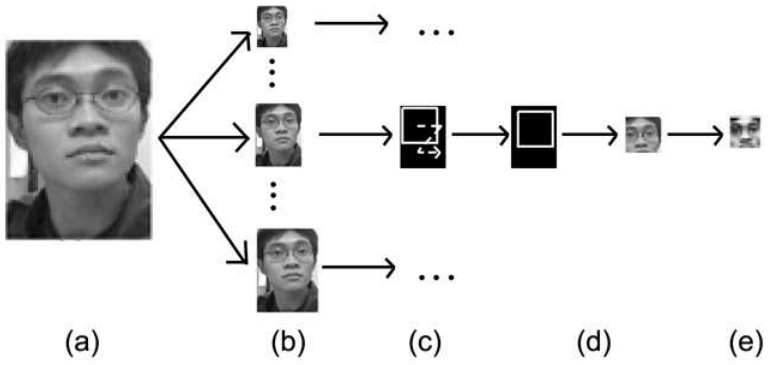

Figure 5.7: The procedure of face detection: (a) Original face candidate region, (b) resize the region into different scales, (c) the zigzag movement of the 19×19 pixel search window, (d) one of the search window, and (e) histogram equalization on the search window.

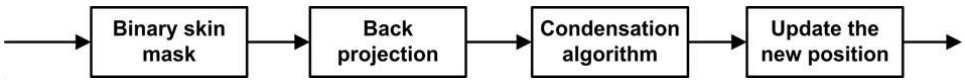

Figure 5.8: The flow of the face tracking module.

5.4 Face tracking module

More information is needed to extract from the detected face. We start to track the face region using face tracking method instead of face detection method. The flow of the face tracking module is shown in Figure 5.8. The main process for tracking the detected object is the Condensation algorithm.

5.4.1 Condensation algorithm

Conditional Density Propagation (Condensation) algorithm [57] is a sampling-based tracking method. It represents the posterior probability density $p(x_t|Z_t)$ using a set of N random samples, denoted $\{S_t^{(n)}, n = 1, \cdots, N\}$ with weights $\pi_t^{(n)}$. There are three phases to compute the density iteratively at each time step t for the set of random samples to track the movement:

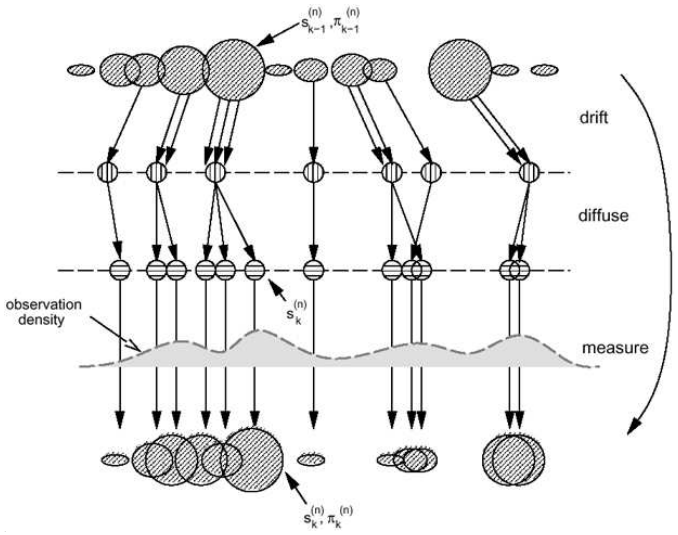

Figure 5.9: One time-step of the Condensation algorithm [57]

1. Selection phase. The element with high weights in the set $S_t^{(n)}$ has a higher probability to be chosen to the predictive steps.

2. Predictive phase. The new sample set $S_{t+1}^{(n)}$ for the new time-step *t+1* is generated by independent Brownian motion of the selected elements in the previous phase. The weights $\pi_{t+1}^{(n)}$ are approximately from the effective prior density $p(x_{t+1}|Z_t)$ for time-step *t+1*.

3. Measurement (Update) phase. A measurement model is given in terms of likelihood $p(z_{t+1}|x_{t+1})$, and the posterior PDF $p(x_{t+1}|Z_{t+1})$ is calculated by

$$p(x_{t+1}|Z_{t+1}) = c_{t+1}p(z_{t+1}|x_{t+1})p(x_{t+1}|Z_t) \tag{5.3}$$

   where c_{t+1} is a normalization factor.

Figure 5.9 shows an example of the three steps of the probabilistic propagation process. By using the above steps, the moment of the tracked object can be estimated by the newly formed sample set.

5.4.2 Tracking the region using Hue color model

Condensation algorithm is employed for face tracking in our system. A measurement model needs to be defined for the algorithm. There are several color models can act as the measurement model.

1. HSV color model. The transformation of HSV is non-linear and the Hue channel represents the most significant characteristic of the color.

2. RGB color model. All the channel information should be included for representing the skin color.

3. YCrCb color model. The distribution of skin color is represented by two channels Cr and Cb without the brightness [55].

To reduce the processing time, we choose one dimension model rather than two or three dimension models. Thus, the model is expressed as a histogram of the Hue channel in HSV color space.

The region detected by the face detection module is used to calculate the histogram of the color model. A new sample set (100) is randomly selected, then, a set of random number is generated and is added into the sample set. Back-projection algorithm translates the color histogram to probability distributions. The probability distributions are transformed from the dynamics and formed the weight of each sample point. The mean position of the tracked face is calculated by the moments of the sample set.

If only the Hue channel is applied in the tracking algorithm, the boundary of tracked face is not precise, and may be diverged to the background when the histogram is similar to the tracked object. To solve this problem, the input image is masked by the binary skin mask that is calculated in the pre-process module. The localization period can be reduced in the Condensation

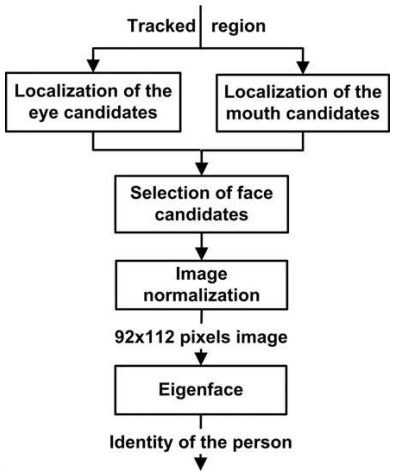

Figure 5.10: The flow of the face recognition module

algorithm. Because the probability of masked region being selected as the sample is decreased. The tracker can compute more quickly and is more insensitive to the background noise.

5.5 Face recognition module

The last module for the system to extract information is face recognition module. Two main steps for the face recognition module are shown in Figure 5.10. Normalization is applied on the face region, and the identity of the resultant image is recognized using the Eigenface method that has been discussed in previous chapter.

5.5.1 Normalization

To improve the accuracy of the recognition result, normalization is performed on the tracked region. These include rotation, scaling and pixel normalization. Reference points, like the position of the eyes, provide the standard method for the normalization. Our proposed localization algorithm for color image is applied

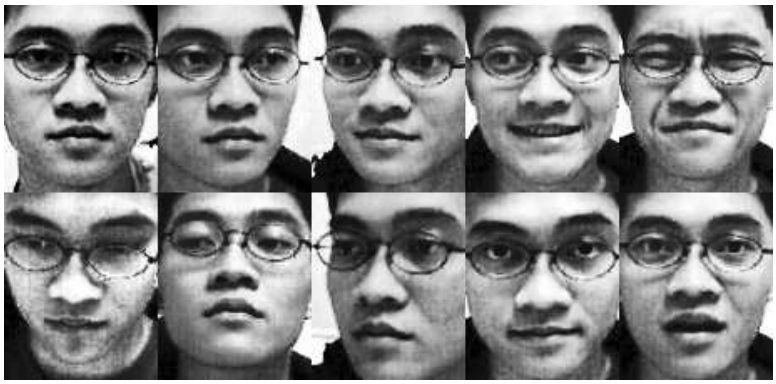

Figure 5.11: A set of training image of one subject.

for the normalization, that use the color information and the separability filter for locating the position of eyes and mouth. The resultant image are normalized to 92×112 pixels image for further processing.

5.5.2 Recognition

Eigenface is used to recognize the image. We have discussed about the property that variation in the pixel values will affect the recognition result. To minimize the effect of the pixel variation, the normalized image is first extracted by the method described in the previous section.

In our system, we divide the recognition process into two phases, i.e., training and recognition. For the training phase, each subject contains 10 normalized images that are captured from our system (Figure 5.11). The set of training images contains variation in the facial expression and different angles from the frontal view. Size and orientation variations are not required in the training set because the normalization step provides a standard image. Thus, the training set can be more compact when compare to those without normalization.

After Eigenface is applied on the training set, the first 30

eigenvectors are selected as principle components. Each of the training images is then projected onto the principle components to extract 30 eigen coefficients. When the system has captured a new image, this image is first projected onto the principle components to calculate 30 eigenvalues. A simple distance measurement, Euclidian distance, is chosen for the similarity measure. The identity of the test image is classified by the face class having the minimum Euclidian distance between the test image and the training image.

5.6 Applications

The system can be modified based on the system architecture to suit different purposes. The modifications include changing the capture device and recording the status of the tracked person.

1. Recognition system. The name of the person can be recognized if that person has a personal record in our system. The system can have two types of applications, naming the person in 1) video conference and 2) news report. The video stream from the video conference becomes the capture device. The position of the person at different time can be recorded using the face tracking module, and the identity of the person can be extracted using the face recognition module. Information can be extracted from the multimedia by the system. The location, size and identity of a person can be found from a photo or video file.

2. Authentication system. The identity given by the user can be verified by our system. When the resultant identity matches the given identity, permission will be granted to that user. For example, the web camera is placed in front of the door as the input device. The user can provide the user identity to the system for verification. The authentication

of the door entry will grant to that user upon the successful checking.

In this chapter, we provide the detailed description about the implementation of our system. These include the system architecture, the process flow of each module and the technologies we applied in our system. Several applications can be developed based on the modification of our system.

□ **End of chapter.**

Chapter 6

Conclusion

This thesis investigates the knowledge of face processing. The technologies of face processing are studied including face detection, face tracking, face recognition and expression recognition. A detailed survey about facial feature is presented. We have classified the facial features and the feature extraction methods into categories. The performance of face recognition methods are summarized that help researcher to have a brief introduction about current techniques and the evolution of face recognition methods.

Before the development of the face processing framework, techniques about different modules are studied. We have proposed an ensemble approach for face detection module. FDCM is applied to increase the accuracy of correct classification for faces. The machine combines three face detection methods that are SNoW algorithm, SVM, and NN. An evaluation among the three individual approaches and FDCM is presented using the same set of training images (6,977) and testing images (24,045) in the experimental result. The result shows that the false alarm rate of FDCM (**7.79%**) is nearly half of the best approach (12.32%) among three approaches.

The position of human eye is an important information for face tracking and face recognition. Two localization algorithms for facial features are proposed for gray-scale image and color image. The algorithm based on template matching and separa-

bility filter is described to find the position of irises and mouth. The separability filter is shown to be useful in improving the accuracy of eye localization. This algorithm combines the color information to normalize the face image, and works satisfactorily in our system.

Lastly, the knowledge obtained from the literature is shown to be useful in the development of a face processing system. A face processing system is developed for generic purpose. Various face processing techniques are employed in the system, that include face detection, face tracking, facial feature localization and face recognition. A wide range of applications can be developed based on the proposed architecture by adding or modifying the function of modules.

These results support our aim of this thesis: knowledge of processing face can be used to develop a generic face processing system.

□ **End of chapter.**

Bibliography

[1] Acknowledgement to computer vision laboratory, faculty of computer and information science, university of ljubljana, ljubljana, slovenia and scv, pters, velenje, slovenia, http://www.lrv.fri.uni-lj.si/facedb.html.

[2] Y. Adini, Y. Moses, and S. Ullman. Face recognition: the problem of compensating for changes in illumination direction. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):721–732, Jul 1997.

[3] A. Albiol, L. Torres, and E. Delp. An unsupervised color image segmentation algorithm for face detection applications. In *Proc. Int'l Conf. on Image Processing*, volume 2, pages 681–684, 2001.

[4] W. Barrett. A survey of face recognition algorithms and testing results. In *Conf. Record of the Thirty-First Asilomar Conf. on Signals, Systems and Computers*, volume 1, pages 301–305, Nov 1997.

[5] M. Bartlett, J. Movellan, and T. Sejnowski. Face recognition by independent component analysis. *IEEE Trans. Neural Networks*, 13(6):1450–1464, Nov 2002.

[6] G. Baudat and F. Anouar. Generalized discriminant analysis using a kernel approach. *The Journal of Neural Computation*, 12(10):2385–2404, 2000.

[7] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. fisherfaces: recognition using class specific linear projection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):711–720, Jul 1997.

[8] A. Bell and T. Sejnowski. Blind separation and blind deconvolution: an information-theoretic approach. In *Int'l Conf. on Acoustics, Speech, and Signal Processing*, volume 5, pages 3415–3418, May 1995.

[9] X. W. B. Bhanu. Gabor wavelet representation for 3-d object recognition. *IEEE Trans. Image Processing*, 6(1):47–64, Jan 1997.

[10] I. Biederman and G. Ju. Surface versus edge-based determinants of visual recognition. In *Cognitive Psychology*, volume 20, pages 38–64, 1988.

[11] J. Bigun, G. Chollet, and G. Borgefors. The m2vts multimodal face database. In *Lecture Notes in Computer Science: Audio- and Video-Based Biometric Person Authentication*, volume 1206, pages 403–409, 1997.

[12] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[13] R. Brunelli and T. Poggio. Face recognition: features versus templates. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(10):1042–1052, Oct 1993.

[14] J. Buhmann, J. Lange, and C. von der Malsburg. Distortion invariant object recognition by matching hierarchically labeled graphs. In *Int'l Joint Conf. on Neural Networks*, pages 155–159, Jun 1989.

[15] T. Chang, T. Huang, and C. Novak. Facial feature extraction from color images. In *Proc. of the 12th IAPR*

Int'l Conf. on Pattern Recognition, volume 2, pages 39–43, 1994.

[16] R. Chellappa, C. Wilson, and S. Sirohey. Human and machine recognition of faces: a survey. *Proc. of the IEEE*, 83(5):705–741, 1995.

[17] J. Choi, S. Lee, C. Lee, and J. Yi. A real-time face recognition system using multiple mean faces and dual mode fisherfaces. In *Proc. of IEEE Int'l Symposium on Industrial Electronics*, volume 3, pages 1686–1689, 2001.

[18] S. W. Christensen, I. Sinclair, and P. A. S. Reed. Designing committees of models through deliberate weighting of data points. *Journal of Machine Learning Research*, 4:39–66, Apr 2003.

[19] C. Chua and R. Jarvis. Point signatures: A new representation for 3d object recognition. *Int'l Journal of Computer Vision*, 25(1):63–85, 1997.

[20] C.-S. Chua, F. Han, and Y.-K. Ho. 3d human face recognition using point signature. In *Proc. of IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pages 233–238, 2000.

[21] J. Cohn, A. Zlochower, J. Lien, and T. Kanade. Feature-point tracking by optical flow discriminates subtle differences in facial expression. In *Proc. of IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pages 396–401, Apr 1998.

[22] T. Cootes, C. Taylor, and A. Lanitis. Multi-resolution search with active shape models. In *Proc. of the 12th IAPR Int'l Conf. on Pattern Recognition*, pages 610–612, Oct 1994.

[23] N. Costen, I. Craw, T. Kato, G. Robertson, and S. Akamatsu. Manifold caricatures: on the psychological consistency of computer face recognition. In *Proc. of Int'l Conf. on Automatic Face and Gesture Recognition*, pages 4–9, Oct 1996.

[24] I. Cox, J. Ghosn, and P. Yianilos. Feature-based face recognition using mixture-distance. In *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 209–216, Jun 1996.

[25] I. Craw, N. Costen, T. Kato, and S. Akamatsu. How should we represent faces for automatic recognition? *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21(8):725–736, Aug 1999.

[26] J. Daugman. Complete discrete 2-d gabor transforms by neual networks for image analysis and compression. *IEEE Trans. Acoustics, Speech and Signal Processing*, 36(7):1169–1179, Jul 1988.

[27] J. Daugman. Face and gesture recognition: overview. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):675–676, Jul 1997.

[28] O. De Vel and S. Aeberhard. Line-based face recognition under varying pose. *IEEE Trans. Pattern Analysis and Machine Intelligenc*, 21(10):1081–1088, Oct 1999.

[29] P. Disorntetiwat and C. Dagli. Simple ensemble-averaging model based on generalized regression neural network in financial forecasting problems. In *IEEE Adaptive Systems for Signal Processing, Communications, and Control Symposium*, pages 477–480, 2000.

[30] G. Donato, M. Bartlett, J. Hager, P. Ekman, and T. Sejnowski. Classifying facial actions. *IEEE Trans. Pattern*

Analysis and Machine Intelligence, 21(10):974–989, Oct 1999.

[31] M.-P. Dubuisson and A. Jain. A modified hausdorff distance for object matching. In *Proc. of Int'l Conf. on Pattern Recognition*, volume 1, pages 566–568, Oct 1994.

[32] G. Edwards, C. Taylor, and T. Cootes. Interpreting face images using active appearance models. In *Proc. of IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pages 300–305, Apr 1998.

[33] S. Eickeler, S. Muller, and G. Rigoll. High quality face recognition in jpeg compressed images. In *Int'l Conf. on Image Processing*, volume 1, pages 672–676, Dec 1999.

[34] P. Ekman and W. Friesen. *The Facial Action Coding System: A Techniue for the Measurement of Facial Movement.* Consulting Psychologists Press, San Francisco, 1978.

[35] I. Essa, T. Darrell, and A. Pentland. Tracking facial motion. In *Proc. of IEEE Workshop on Motion of Non-Rigid and Articulated Objects*, pages 36–42, Nov 1994.

[36] I. Essa and A. Pentland. Coding, analysis, interpretation, and recognition of facial expressions. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):757–763, Jul 1997.

[37] R. Fisher. The use of multiple measures in taxonomic problems. In *Ann. Eugenics*, pages 179–188, 1936.

[38] T. Fromherz. Face recognition: a summary of 1995 - 1997. Technical Report TR-98-027, Berkeley, CA, 1998.

[39] K. Fukui and O. Yamaguchi. Facial feature point extraction method based on combination of shape extraction and

pattern matching. *IEICE Trans.*, J80-D-II(8):2170–2177, 1997.

[40] K. Fukunaga. *Introduction to statisticalPattern Recognition.* Academic Press, New York, second edition, 1990.

[41] Y. Gao and M. Leung. Face recognition using line edge map. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(6):764–779, Jun 2002.

[42] A. Georghiades, P. Belhumeur, and D. Kriegman. From few to many: illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. Pattern Analysis and Machine Intelligence*, pages 643–660, 2001.

[43] A. Georghiades, D. Kriegman, and P. Belhurneur. Illumination cones for recognition under variable lighting: faces. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 52–58, Jun 1998.

[44] F. Goudail, E. Lange, T. Iwamoto, K. Kyuma, and N. Otsu. Face recognition system using local autocorrelations and multiscale integration. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(10):1024–1028, Oct 1996.

[45] N. Graham, D.B.; Allinson. Face recognition using virtual parametric eigenspace signatures. In *Int'l Conf. on Image Processing and Its Applications*, volume 1, pages 106–110, Jul 1997.

[46] G. Guo, S. Li, and K. Chan. Face recognition by support vector machines. In *Proc. of IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pages 196–201, 2000.

[47] J.-J. Guo and P. Luh. Market clearing price prediction using a committee machine with adaptive weighting coef-

ficients. In *IEEE Power Engineering Society Winter Meeting*, volume 1, pages 77–82, 2002.

[48] H. Gupta, A. Agrawal, T. Pruthi, C. Shekhar, and R. Chellappa. An experimental evaluation of linear and kernel-based methods for face recognition. In *Proc. of IEEE Workshop on Applications of Computer Vision*, pages 13–18, 2002.

[49] S. Gutta, J. Huang, P. Jonathon, and H. Wechsler. Mixture of experts for classification of gender, ethnic origin, and pose of human faces. *IEEE Trans. on Neural Networks*, 11(4):948–960, Jul 2000.

[50] S. Gutta, J. Huang, B. Takacs, and H. Wechsler. Face recognition using ensembles of networks. In *Proc. of Int'l Conf. on Pattern Recognition*, volume 3, pages 25–29, Aug 1996.

[51] P. Hallinan. *A deformable model for face recognition under arbitrary lighting conditions*. PhD thesis, Harvard University, 1995.

[52] L. Hansen and P. Salamon. Neural network ensembles. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(10):993–1001, Oct 1990.

[53] I. Haritaoglu and M. Flickner. Attentive billboards. In *Proc. 11th Int'l Conf. on Image Analysis and Processing*, pages 162–167, 2001.

[54] H. Hong, H. Neven, and C. von der Malsburg. Online facial expression recognition based on personalized galleries. In *Proc. of IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pages 354–359, Apr 1998.

[55] R.-L. Hsu, M. Abdel-Mottaleb, and A. Jain. Face detection in color images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(5):696–706, May 2002.

[56] C. Iordanoglou, K. Jonsson, J. Kittler, and J. Matas. Wearable face recognition aid. In *Proc. of IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing*, volume 6, pages 2365–2368, 2000.

[57] M. Isard and A. Blake. Condensation - conditional density propagation for visual tracking. *Int'l J. Computer Vision*, 1998.

[58] R. Jacobs, M. Jordan, S. Nowlan, and G. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3:79–87, 1991.

[59] O. Jesorsky, K. J. Kirchberg, and R. W. Frischholz. Robust face detection using hausdorff distance. In *Proc. of Int'l Conf. on Audio- and Video-Based Biometric Person Authentication*, pages 90–95, Jun 2001.

[60] W. Jiang and M. Tanner. On the asymptotic normality of hierarchical mixtures-of-experts for generalized linear models. *IEEE Trans. on Information Theory*, 46(3):1005–1013, May 2000.

[61] T. Kawaguchi, D. Hidaka, and M. Rizon. Detection of eyes from human faces by hough transform and separability filter. In *Proc. Int'l Conf. on Image Processing*, volume 1, pages 49–52, 2000.

[62] J. Kawato, S.and Ohya. Automatic skin-color distribution extraction for face detection and tracking. In *Proc. Fifth Int'l Conf. on Signal Processing Proceedings*, volume 2, pages 1415–1418, 2000.

[63] K. I. Kim, K. Jung, and H. J. Kim. Face recognition using kernel principal component analysis. In *IEEE Signal Processing Letters*, volume 9, pages 40–42, Feb 2002.

[64] S. Kimura and M. Yachida. Facial expression recognition and its degree estimation. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 295–300, Jun 1997.

[65] R. Kjeldsen and J. Kender. Finding skin in color images. In *Proc. of the Second Int'l Conf. on Automatic Face and Gesture Recognition*, pages 312–317, 1996.

[66] V. Kohir and U. Desai. Face recognition using a dct-hmm approach. In *Proc. Workshop on Applications of Computer Vision*, pages 226–231, Oct 1998.

[67] K. Konen. Comparing facial line-drawings with gray-level images: A case study on phantomas. In *Proc. of The Internet Corporation for Assigned Names and Numbers*, 1996.

[68] M. Lades, J. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R. Wurtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE Trans. Computers*, 42(3):300–311, Mar 1993.

[69] A. Lanitis, C. Taylor, and T. Cootes. Automatic interpretation and coding of face images using flexible models. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):743–756, Jul 1997.

[70] A. Lanitis, C. Taylor, and T. Cootes. Toward automatic simulation of aging effects on face images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(4):442–455, 2002.

[71] S. Lawrence, C. Giles, A. C. Tsoi, and A. Back. Face recognition: a convolutional neural-network approach. *IEEE Trans. Neural Networks*, 8(1):98–113, Jan 1997.

[72] R. Liao and S. Li. Face recognition based on multiple facial features. In *Proc. of IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pages 239–244, 2000.

[73] C.-H. Lin and J.-L. Wu. Automatic facial feature extraction by genetic algorithms. *IEEE Trans. Image Processing*, 8(6):834–845, Jun 1999.

[74] C. Liu and H. Wechsler. Comparative assessment of independent component analysis. In *Proc. of Int'l Conf. on Audio- and Video-Based Biometric Person Authentication*, pages 22–24, Mar 1999.

[75] C. Liu and H. Wechsler. Robust coding schemes for indexing and retrieval from large face databases. *IEEE Trans. Image Processing*, 9(1):132–137, Jan 2000.

[76] C. Liu and H. Wechsler. A gabor feature classifier for face recognition. In *Proc. of IEEE Int'l Conf. on Computer Vision*, volume 2, pages 270–275, 2001.

[77] C. Liu and H. Wechsler. A shape- and texture-based enhanced fisher classifier for face recognition. *IEEE Trans. Image Processing*, 10(4):598–608, Apr 2001.

[78] C. Liu and H. Wechsler. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. *IEEE Trans. Image Processing*, 11(4):467–476, Apr 2002.

[79] E. Lizama, D. Waldoestl, and B. Nickolay. An eigenfaces-based automatic face recognition system. In *IEEE Int'l*

Conf. on Systems, Man, and Cybernetics, pages 174–177, Oct 1997.

[80] R. Lotlikar and R. Kothari. Fractional-step dimensionality reduction. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(6):623–627, Jun 2000.

[81] J. Lu, K. Plataniotis, and A. Venetsanopoulos. Face recognition using kernel direct discriminant analysis algorithms. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(1):116–126, Jan 2003.

[82] J. Lu, K. Plataniotis, and A. Venetsanopoulos. Face recognition using lda-based algorithms. *IEEE Trans. Neural Networks*, 14(1):195–200, Jan 2003.

[83] M. Lyons, J. Budynek, and S. Akamatsu. Automatic classification of single facial images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21(12):1357–1362, Dec 1999.

[84] S. Marcel and S. Bengio. Improving face verification using skin color information. In *Proc. of Int'l Conf. on Pattern Recognition*, volume 2, pages 378–381, 2002.

[85] E. Marszalec, B. Martinkauppi, M. Soriano, and M. Pietikainen. A physics-based face database for color research. *The Journal of Electronic Imaging*, 9(1):32–38, 2000.

[86] A. Martinez. Recognizing imprecisely localized, partially occluded, and expression variant faces from a single sample per class. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(6):748–763, Jun 2002.

[87] A. Martinez and R. Benavente. The ar face database. Tech. Report CVC 24, Purdue University, 1998.

[88] K. Mase. Recognition of facial expression from optical flow. *IEICE Trans. E*, 74(10):3474–3483, 1991.

[89] J. Matas, M. Hamouz, K. Jonsson, J. Kittler, Y. Li, C. Kotropoulos, A. Tefas, I. Pitas, T. Tan, H. Yan, F. Smeraldi, J. Bigun, N. Capdevielle, W. Gerstner, S. Ben-Yacoub, Y. Abeljaoued, and E. Mayoraz. Comparison of face verification results on the xm2vtfs database. In *Proc. of Int'l Conf. on Pattern Recognition*, volume 4, pages 858–863, 2000.

[90] B. Menser and F. Muller. Face detection in color images using principal components analysis. In *Seventh Int'l Conf. on Image Processing and Its Applications*, volume 2, pages 620–624, 1999.

[91] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre. Xm2vtsdb: The extended m2vts database. In *Int'l conf. on Audio and Video-Based Biometric Person Authentication*, pages 72–77, 1999.

[92] B. Moghaddam, T. Jebara, and A. Pentland. Bayesian face recognition. *The Journal of the Pattern Recognition*, 33:1771–1782, 2000.

[93] B. Moghaddam, W. Wahid, and A. Pentland. Beyond eigenfaces: probabilistic matching for face recognition. In *Proc. Third IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pages 30–35, 1998.

[94] H. Moon and P. Phillips. The feret verification testing protocol for face recognition algorithms. In *Proc. of IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pages 48–53, Apr 1998.

[95] K.-R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf. An introduction to kernel-based learning

algorithms. *IEEE Trans. Neural Networks*, 12(2):181–201, Mar 2001.

[96] C. Nakajima, N. Itoh, M. Pontil, and T. Poggio. Object recognition and detection by a combination of support vector machine and rotation invariant phase only correlation. In *Proc. of 15th Int'l Conf. on Pattern Recognition*, volume 4, pages 787–790, 2000.

[97] A. Nefian and I. Hayes, M.H. Hidden markov models for face recognition. In *Int'l conf. on Acoustics, Speech, and Signal Processing*, volume 5, pages 2721–2724, May 1998.

[98] E. Osuna, R. Freund, and F. Girosit. Training support vector machines: an application to face detection. In *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 130–136, 1997.

[99] A. Pentland. Looking at people: sensing for ubiquitous and wearable computing. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(1):107–119, Jan 2000.

[100] A. Pentland and T. Choudhury. Face recognition for smart environments. *The Journal of Computer*, 33(2):50–55, Feb 2000.

[101] P. Phillips, P. Grother, R. Micheals, D. Blackburn, E. Tabassi, and J. Bone. Frvt 2002: Overview and summary, Mar 2003.

[102] P. Phillips, H. Moon, S. Rizvi, and P. Rauss. The feret evaluation methodology for face-recognition algorithms. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(10):1090–1104, 2000.

[103] P. J. Phillips. Support vector machines applied to face recognition. In *Advances in Neural Information Processing Systems*, volume 11, pages 803–809, 1998.

[104] M. Rosenblum, Y. Yacoob, and L. Davis. Human expression recognition from motion using a radial basis function network architecture. *IEEE Trans. Neural Networks*, 7(5):1121–1138, Sep 1996.

[105] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20(1):23–38, Jan 1998.

[106] A. Samal and P. Iyengar. Automatic recognition and analysis of human faces and facial expressions: A survey. *The Journal of Pattern Recognition*, 25(1):65–77, Jan 1992.

[107] C. Sanderson and K. Paliwal. Fast feature extraction method for robust face verification. In *Electronics Letters*, volume 38, pages 1648–1650, 5 Dec 2002.

[108] R. E. Schapire. The strength of weak learnability. *Machine Learning*, 5(2):197–227, 1990.

[109] H. Schneiderman and T. Kanade. Probabilistic modeling of local appearance and spatial relationships for object recognition. In *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 45–51, 1998.

[110] B. Scholkopf, A. Smola, and K. Muller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319, 1998.

[111] L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization of human faces. *J. Opt. Soc. Am. A*, 4(3):519–524, 1987.

[112] S. Srisuk and W. Kurutach. A new robust face detection in color images. In *Proc. Fifth IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pages 306–311, 2002.

[113] H. Stern and B. Efros. Adaptive color space switching for face tracking in multi-colored lighting environments. In *Proc. Fifth IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pages 236–241, May 2002.

[114] E. Stollnitz, A. DeRose, and D. Salesin. Wavelets for computer graphics: a primer. 2. *IEEE Trans. Computer Graphics and Applications*, 15(4):75–85, Jul 1995.

[115] G. Su, C. Zhang, R. Ding, and C. Du. Mmp-pca face recognition method. In *Electronics Letters*, volume 38, pages 1654–1656, 5 Dec 2002.

[116] D. Swets and J. Weng. Using discriminant eigenfeatures for image retrieval. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(8):831–836, Aug 1996.

[117] B. Takacs and H. Wechsler. Face recognition using binary image metrics. In *Proc. of IEEE Int's Conf. on Automatic Face and Gesture Recognition*, pages 294–299, Apr 1998.

[118] H. Tanaka, M. Ikeda, and H. Chiaki. Curvature-based face surface recognition using spherical correlation. principal directions for curved object recognition. In *Proc. of IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pages 372–377, Apr 1998.

[119] Y.-I. Tian, T. Kanade, and J. Cohn. Recognizing action units for facial expression analysis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 23(2):97–115, Feb 2001.

[120] B. Tiddeman, M. Burt, and D. Perrett. Prototyping and transforming facial textures for perception research. *IEEE Computer Graphics and Applications*, 21(5):42–50, Sep/Oct 2001.

[121] L. Torres, J. Reutter, and L. Lorente. The importance of the color information in face recognition. In *Proc. of Int'l Conf. on Image Processing*, volume 3, pages 627–631, 1999.

[122] M. Turk and A. Pentland. Face recognition using eigenfaces. In *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 586–591, 1991.

[123] D. Valentin, H.Abdi, A. O'Toole, and G. Cotrell. Connectionist models of face processing: A survey. *The Journal of Pattern Recognition*, pages 1209–1230, 1994.

[124] V. Vapnik. *Statistical Learning Theory.* John Wiley and Sons, Inc., 1998.

[125] Y. Wang, C.-S. Chua, Y.-K. Ho, and Y. Ren. Integrated 2d and 3d images for face recognition. In *Proc. of Int'l Conf. on Image Analysis and Processing*, pages 48–53, Sep 2001.

[126] L. Wiskott, J.-M. Fellous, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):775–779, Jul 1997.

[127] L. Wiskott, J.-M. Fellous, and C. von der Malsburg. Face recognition by elastic bunch graph matching. In *Proc. Int'l conf. on Image Processing*, volume 1, pages 139–132, 1997.

[128] W. Wolf, B. Ozer, and T. Lv. Smart cameras as embedded systems. *The Journal of Computer*, 35(9):48–53, Sep 2002.

[129] Y. Yacoob and L. Davis. Recognizing human facial expressions from long image sequences using optical flow. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(6):636–642, Jun 1996.

[130] M.-H. Yang, D. Kriegman, and N. Ahuja. Detecting faces in images: a survey. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(1):34–58, Jan 2002.

[131] M.-H. Yang, D. Roth, and N. Ahuja. A snow-based face detector. In *Advances in Neural Information Processing Systems*, volume 12, pages 862–868. The MIT Press, 1999.

[132] H. Yu and J. Yang. A direct lda algorithm for high-dimensionaldata - with application to face recognition. *The Journal of the Pattern Recognition*, 34(10):2067–2070, Oct 2001.

[133] J. Zhang, Y. Yan, and M. Lades. Face recognition: eigenface, elastic matching, and neural nets. In *Proc. of the IEEE*, volume 85, pages 1423–1435, Sep 1997.

[134] Z. Zhang, M. Lyons, M. Schuster, and S. Akamatsu. Comparison between geometry-based and gabor-wavelets-based facial expression recognition using multi-layer perceptron. In *Proc. of IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pages 454–459, Apr 1998.

[135] S. Zhou and R. Chellappa. A robust algorithm for probabilistic human recognition from video. In *Proc. Int'l Conf. on Pattern Recognition*, pages 226–229, 2002.