

國立交通大學

資訊科學與工程研究所

博士論文

人臉偵測之研究

A Study on Face Detection

研究生: 井民全

指導教授: 陳玲慧教授

中華民國九十八年七月

人臉偵測之研究

A Study on Face Detection

研 究 生：井民全

Student: Min-Quan Jing

指 導 教 授：陳玲慧

Advisor: Dr. Ling-Hwei Chen

國 立 交 通 大 學

資 訊 科 學 與 工 程 研 究 所

博 士 論 文

A Dissertation

Submitted to Institute of Computer Science and Engineering

College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

in

Computer Science

June 2009

Hsinchu, Taiwan, Republic of China

中華民國九十八年七月

人臉偵測之研究

研究生:井民全

指導教授:陳玲慧博士

國立交通大學資訊學院 資訊工程系

摘要

近年來，數位安全監控技術的普遍發展，使得完全自動化的人臉系統 (fully automated face system) 越來越受到重視。人臉偵測是自動化人臉系統的基石，任何人臉應用系統都需要有一個強固的人臉偵測方法，才能達到預期的效果，因此人臉偵測的研究與分析便顯得重要而且迫切。雖然人臉偵測方法的研究已經超過 20 年，也有許多成果已經發表。可是對於多變化環境下的人臉偵測，特別是針對不同人臉角度、不同光源方向與有顏色光源的人臉位置偵測方法，依然是一項十分具挑戰性的研究領域。

一般人臉偵測方法的主要架構大致是由下列幾項要素組成：人臉候選位置的判定、臉部特徵的擷取與人臉位置的偵測。可是由於多變化的環境，導致人臉偵測的困難。例如在研判人臉可能出現的位置時，就可能受取像時的環境條件所影響，有顏色的光源與不同角度的側面光源，如不加以考量就會影響人臉候選位置的判定。另外由於攝影機與人臉之間的取像角度，使得人臉在影像中所呈現的姿勢可能為

正面、接近正面、半側面、側面等，一些重要的臉部特徵如眼睛，卻可能會因為影子、眼鏡和人臉姿勢的關係導致偵測錯誤。而不同的臉部表情也會產生不同的臉部呈現樣式。這些因素增加了人臉偵測的困難性。本論文之主要目的在於提出一個解決上述問題的人臉偵測方法。

在本論文中，我們首先將會提出一個在多變化環境下的眼睛水平線偵測方法，利用膚色資訊將人臉可能所在的區域標記出來，然後根據人類眼睛擁有的非膚色特性與高亮度(gray-level)變化特性，將眼睛的位置標示出來，最後利用人臉的幾何特性將眼睛水平線標記定位。這個方法主要的貢獻在於提出一個有效的方法，取得多變化環境下的眼睛候選位置與水平線資訊。接下來，利用得到眼睛水平線位置與眼睛可能位置座標，發展一個在多變化環境下不同姿勢的人臉偵測方法。首先將膚色資訊所標記出來的人臉區域進行肩膀區域的判定與刪除，接著利用定義好的側面人臉特徵幾何規則將人臉區域分成側面人臉或非側面人臉區域。對於非側面的人臉候選區域，利用前述方法取得眼睛水平線與眼睛可能位置座標。最後針對不同人臉姿勢與人臉亮度分佈的特性，標記出真正的眼睛位置並藉此將人臉位置標示出來。本論文所提出的方法，經過標準人臉資料庫實驗證實，可以解決人臉偵測在多變化環境下所遭遇的問題。

A Study on Face Detection

Student: Min-Quan Jing

Advisor: Dr. Ling-Hwei Chen

Institute of Computer Science and Engineering
National Chiao Tung University

ABSTRACT

While digital surveillance systems are receiving increasing concern in modern society, developing a fully automated face system is getting more and more attentions than before. However, a robust face detector is a foundation of building an automated face system. Face detection method insures a face system realizable. Thus, the demand for an efficient method to automatically detect face becomes urgent.

Although face detection has been studied for more than 20 years, developing a human face detector under various environments is still a challenging task. An automatical face-detection job would include three steps: face candidate location, facial feature extraction and face detection. Some factors make face location difficult. One is the variety of colored lighting sources, another is that facial features such as eyes may be partially or wholly occluded by shadow generated by a bias lighting direction; others are races and different face poses with/without glasses. These factors make face detection difficult. The aim of the research is to provide a solution to these problems.

In this thesis, we will firstly propose a method to extract the horizontal eye line

of a face under various environments. Based on the facts that the eye color is very different from skin color and the gray level variance of an eye is high, some eye-like regions are located. Then the horizontal eye line is extracted based on the located eye-like regions and some geometric properties in a face. The contribution of this method is providing information of eye-like region positions and human horizontal eye line under various environments. Next, a method based on the extracted eye-like region position and horizontal eye line information will be proposed to detect a face with different poses under various environments. The basic idea is, first, skin regions are extracted from an input image using skin color information and then the shoulder part is determined and cut out by using shape information. The remained head part is identified as a face candidate. For a face candidate, we apply a set of geometric features to determine if it is a profile face. If the face candidate is a non-profile face, then a set of eye-like rectangles extracted from the face candidate and the lighting distribution are used to determine. Solving the poses problem and detecting face location under various environments are the main contributions of this method. Experimental results show that the proposed method is robust under a wide range of lighting conditions, different poses and races.

誌 謝

首先要對我的指導教授陳玲慧博士獻上我最誠摯的感謝。在她細心與熱心的指導下，讓我能夠完成博士論文。從她身上我學習到豐富的知識與嚴謹的態度，更重要的是，我從她身上學習到真誠待人的處事原則。她是幫助我最多的人，在我的人生中，我很幸運能夠遇到這麼一位這麼好的老師。

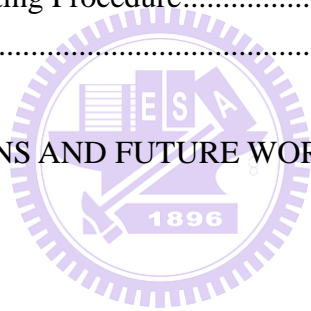
接著我要特別感謝的人是李建興學長，他在我最需要幫忙的時候伸出援手並且給我溫暖的建議與打氣。林瑞祥學長，在我困難的時候幫我排除萬難解決問題。我由衷地感謝你們，你們是我心中學長的典範。我要感謝自動化資訊處理實驗室的朋友們，惠龍我從你身上學到堅持到底的精神，我記得每個晚上你鼓勵我的話語、文超很感謝你為我做的事情、芳如我永遠記得妳下雨天幫我買便當的樣子、盈如謝謝妳幫我送論文給教授們，我知道妳一個一個去敲門、Sasami、俊旻、阿和、小胖、阿鴻、阿達、Gavin、小貝，實驗室有你們在讓我覺得很開心與溫暖。宜軒謝謝妳來實驗室看我幫我打氣，我收到了妳的祝福、阿雪謝謝妳不辭辛勞的來實驗室修改論文。合吉、朝君、小邵、致生、崇荏、佳峰，謝謝你們在我困難時陪我，你們的友誼是我重要的支柱。

最後我要感謝我的父母，在我徬徨時多次給我堅定的力量、讓我能夠勇往直前。貞如我的妻子謝謝妳多年的忍讓，使我沒有後顧之憂地追求我的夢想。這些年來辛苦妳了。我僅以最誠摯的心將本論文獻給我的父母、妻子與曾經在我生命中給予我鼓勵及協助的每一個人，謝謝你們。

TABLE OF CONTENTS

摘要.....	i
ABSTRACT.....	iii
誌謝.....	v
TABLE OF CONTENTS	vi
LIST OF TABLES	viii
LIST OF FIGURES	ix
ABBREVIATION.....	xiii
CHAPTER 1 INTRODUCTION	1
1.1 Motivation and Applications.....	1
1.2 The Main Problems and Current State.....	4
1.2.1 The Main Problem of Face Detection.....	5
1.3 Synopsis of the Dissertation.....	6
CHAPTER 2 A NOVEL METHOD FOR HORIZONTAL EYE LINE DETECTION UNDER VARIOUS ENVIRONMENTS	7
2.1 Introduction.....	7
2.2 The Proposed Method.....	10
2.2.1 The Proposed Skin Region Detector.....	10
2.2.1.1 Skin Region Extraction.....	11
2.2.1.2 Skin Region Filter.....	12
2.2.2 The Proposed Eye-Like Region Detector.....	15
2.2.2.1 Intensity Based Eye-Like Region Extraction.....	16
2.2.2.2 Non-Skin Color Based Eye Detector.....	19
2.2.3 False Eye-Like Region Removing.....	20
2.2.3.1 Overlapping/ Hair-Reflecting/ Beard/ Clothes False Region Removing.....	21
2.2.3.2 Eye Position Refining And Isolated False Region Removing.....	22
2.2.3.3 Forehead-hair False Regions Removing.....	25

2.2.4 The Proposed Horizontal Eye Line Detector	30
2.3 Experimental Results	33
CHAPTER 3 A NOVEL FACE DETECTION METHOD UNDER VARIOUS ENVIRONMENTS	36
3.1 Introduction.....	36
3.2 The Proposed Method	37
3.2.1 Profile Detection	39
3.2.1.1 Shoulder Removing Procedure	39
3.2.1.2 Profile Feature Extraction	42
3.2.2 Non-Profile Face Detection	46
3.2.2.1 Vertical Symmetric Line Location	48
3.2.2.2 False Eye-Like Rectangle Removing.....	52
3.2.3 True Eye Locating Procedure.....	53
3.3 Experimental Results	57
CHAPTER 4 CONCLUSIONS AND FUTURE WORKS	65
REFERENCES	66
PUBLICATION LIST	69
VITA	70



LIST OF TABLES

TABLE 2.1	THE TEMPLATES FOR SIX EYE CLASSES	24
TABLE 3.1	DETECTION RESULTS ON HHI DATABASE (IMAGE SIZE 640 x 480)	61
TABLE 3.2	DETECTION RESULTS ON CHAMPION DATABASE (IMAGE SIZE ~ 150 x 220)	62
TABLE 3.3	DETECTION RESULTS ON NON-PROFILE FACES OF HHI DATABASE	62



LIST OF FIGURES

Fig. 1.1	The block diagram of the proposed face detector.	6
Fig. 2.1	Some results of applying skin region extractor on different races under various lighting environments. (a) Four images taken under normal, bias colored and high lighting environments. (b) The extracted candidate skin regions using threshold 142. (c) The extracted candidate skin regions using threshold 132.....	12
Fig. 2.2	Some examples for illustrating the skin region filter. (a) A face image. (b) Candidate skin regions using 132 as threshold. (c) The result of removing regions of small sizes and improper shapes in (b). (d) Candidate skin regions using 142 as threshold. (e) The result of removing small regions in (d). (f) The result of merging (c) and (e) and imposed on (a). (g) The result of removing uniform content regions and cross over regions in (f).	14
Fig. 2.3	The block diagram of the eye-like region detector.....	15
Fig. 2.4	Subjects taken under various lighting environments. (a) The normal lighting condition. (b) Top lighting source. (c) Colored and left bias lighting source. (d) High lighting source.....	16
Fig. 2.5	The flowchart of the intensity based eye-like region detector....	17
Fig. 2.6	An example of the bi-level thresholding results. (a) A face region. (b) The histogram of detected gray pixels within the skin region. (c) Bi-level thresholding results of (a) using different thresholds.....	18
Fig. 2.7	An example of the horizontal gray value projection for an eye. (a) The projection region extended from eye-like region. (b) The projected histogram of the projection region in (a).....	19
Fig. 2.8	The results of applying the intensity based algorithm. (a) The result of a normal lighting source face. (b) The result of a colored and left bias lighting source face.....	19
Fig. 2.9	The result of applying the proposed intensity and non-skin color based eye detectors. (a) The eye-like regions located based on non-skin color information. (b) The combined result of Fig. 2.8(b) and (a).....	20
Fig. 2.10	Five false region classes. (a) An example to show four classes	

	of false eye-like regions. (b) Another example to show forehead - hair class	21
Fig. 2.11	The result of applying the false eye-like removing procedure. (a) A bounding rectangle of a candidate skin region. (b) The horizontal projection histogram of the bounding rectangle in (a). (c) The detected upper and bottom lines. (d) The result after applying the removing procedure presented in Section 2.2.3.1 to (a).....	22
Fig. 2.12	The result of applying the eye position refining procedure. (a) An example to show an eye not located at the center of the eye-like region. (b) The result of applying the eye position refining procedure to (a).....	25
Fig. 2.13	The result of removing isolated false regions. (a) An example to show isolated regions. (b) The result of removing the isolated regions in (a).....	25
Fig. 2.14	An example to illustrate the bounding rectangle of a face. (a) The minimum rectangle containing all eye-like regions. (b) The expanding result of (a).....	26
Fig. 2.15	Two examples to show different models of gray level histograms. (a) A bias lighting face and its corresponding histogram. (b) A normal lighting face and its corresponding histogram.....	27
Fig. 2.16	Two examples to show the over-segmentation and under-segmentation. (a) Original image. (b) An over-segmentation obtained after applying the threshold process to (a). (c) Original image. (d) An under-segmentation obtained after applying the threshold process to (c).....	28
Fig. 2.17	The result of applying the forehead-hair false region removing procedure. (a) The eye-like regions before applying the removing procedure. (b) The result of applying the bi-level threshold to (a). (c) The result of removing the forehead - hair false regions in (b).....	30
Fig. 2.18	An example to explain the eye-like region resuming procedure. (a) The bounded ranges for resuming the removed eye-like regions in the shadow area of a face. (b) Three eye-like regions (marked by white rectangles) got back.....	30

Fig. 2.19	Some examples for the horizontal projection. (a) Only one valley case. (b) Two valleys case. (c) $D1 < s1 \times D2$ with two eye-like regions near the second valley line. (d) $D1 > s2 \times D2$ with two eye-like regions near the first valley line.....	33
Fig. 2.20	A part of images from HHI face database with different poses and eye-glasses.....	34
Fig. 2.21	A part of images from Champion face database with different skin colors.....	34
Fig. 2.22	The detection results for persons with shadows and various lighting sources on their faces.....	34
Fig. 2.23	The detection results of face images collected from the Internet, MPEG7 video clips and PIE database.....	35
Fig. 2.24	Some error/missing examples.....	35
Fig. 3.1	The block diagram of the proposed face detector.....	38
Fig. 3.2	Some results of applying Jing-Chen skin region extractor. (a) Three images taken under various lighting environments. (b) The extracted skin regions (white pixels) using Jing-Chen skin region detector.....	39
Fig. 3.3	An example showing two men wearing clothes with/without skin-like color. (a) A man wearing a T-shirt of skin-like color. (b) A man wearing a T-shirt of non skin-like color. (c) The detected skin region of (a). (d) The detected skin region of (b). (e) The horizontal projection of (c). (f) The horizontal projection of (d).....	40
Fig. 3.4	An example for removing shoulder from a skin region. (a) The Top-Down line for a projection histogram. (b) The neck point and the cutting line located. (c) The located cutting line on Fig. 3.3(c). (d) The result of removing shoulder from (c).....	42
Fig. 3.5	An example for illustrating the profile features. (a) Some feature points on a right profile. (b) The extracted nose peak. (c) The extracted chin scan line and chin point. (d) The extracted nose bottom scan line and nose bottom. (e) The extracted nose top.....	43
Fig. 3.6	An example to show the defined profile variables and the face rectangle. (a) The defined variables for a skin region. (b) The defined profile face rectangle.....	45

Fig. 3.7	An example for simple face model. (a) A face image. (b) The located horizontal eye line and eye-like rectangles. (c) The defined face rectangle.....	47
Fig. 3.8	Two examples for the complexity face model. (a) A man with glasses. (b) A man under bias lighting source.....	48
Fig. 3.9	The vertical histogram projection patterns and detected vertical symmetric lines. (a) A half profile pattern. (b) A bias lighting pattern. (c) A near frontal pattern with left-side pose. (d) A normal pattern.....	49
Fig. 3.10	An example to illustrate the eye-like boundary for a skin region.....	52
Fig. 3.11	Two examples for true eye and face location. (a) A case satisfying Rule 1. (b) The detected face location based on the eye locations from (a). (c) A case satisfying Rule 2. (d) The detected face location based on the eye locations of (c).....	54
Fig. 3.12	Two examples for the two kinds of eye patterns. (a) The two-layered pattern. (b) The three-layered pattern.....	56
Fig. 3.13	Some examples for true eye and face rectangle location. (a) Two-layered patterns at each side of the symmetric line. (b) The detected face location based on the true eye locations on (a). (c) One two-layered pattern detected. (d) The detected face location based on the eye locations on (c). (e) Another example for only one two-layered pattern detected at one side. (f) The detected face location based on the eye locations on (e). (g) One three-layered pattern detected. (h) The detected face location based on the eye locations on (g). (i) An example of other patterns detected. (j) The detected face location based on the eye locations on (i).....	56
Fig. 3.14	A part of images from HHI face database.....	58
Fig. 3.15	A part of images from Champion face database.....	59
Fig. 3.16	The detection results for persons with different skin colors.....	59
Fig. 3.17	The detection results for persons with different poses.....	60
Fig. 3.18	The ROC curves for our face detector on HHI and Champion databases.....	62
Fig. 3.19	The detection results of face images collected from our laboratory, the Internet, and MPEG7 video clips.....	63
Fig. 3.20	The detection results for multiple faces.....	64

ABBREVIATION

SVM	Support Vector Machine
HHI	Heinrich Hertz Institute
PIE	The CMU pose, illumination, and expression database
ROC	Receiver Operating Characteristic



CHAPTER 1

INTRODUCTION

1.1 MOTIVATION AND APPLICATIONS

Although face detection has been studied for more than 20 years, developing a human face detector under various environments is still a challenging task. Some factors make face location difficult. One is the variety of colored lighting sources, another is that facial features such as eyes may be partially or wholly occluded by shadow generated by a bias lighting direction; others are races and different face poses with/without glasses. In a drowsy driving warning system, driver's mental condition can be revealed by measuring the eye movement from the eye location [1].

How to locate the eyes of a driver under extreme lighting condition is an important issue for a successful intelligent transportation system. Detecting the eye line can help locate eyes. Moreover, the detected eye line can be used to do face detection. There are several methods are proposed to overcome face detection problems, Hjelmås and Low [2] have presented an extensive survey of feature-based and image-based algorithms for face detection. In feature-based methods, the most widely used technique is skin color detection based on one of the color models. Most image-based approaches which based on multi-resolution window scanning are proposed to detect

faces at all scales, making them computationally expensive. Also, Yang [3] classifies the approaches into four categories: knowledge-based, feature invariant, template matching, and appearance-based and address one of the difficulty of face detection is lighting source varies significantly. Several methods [4-18] had been proposed by now. Some [4-6] use eigenface, neural network and support vector machine (SVM) to detect faces of restricted poses under normal lighting condition. Chow [7] proposed an approach to locate facial features (such as eyes, eyebrows) by using an eigenmask under different lighting conditions. However, the method fails to locate the facial features for some kinds of poses (such as near-profile and profile face). Shin and Chuang [8] uses shape and contour information to locate face in a plain background and normal lighting condition. However, it is difficult to detect the predefined shape of a face in a complex background and various lighting conditions. Wu *et al.* [9] proposed an algorithm to locate face candidates using eye-analogue segments information. The algorithm would fail when a person wears glasses. Some methods [10-12] use mosaic, edge and mathematical morphology to detect eyes, however, these methods would fail to locate eyes when a face is under a poor lighting condition (such as bias-light source). Wang [13] proposed a boosted based algorithm to detect a face. But the training phase of the method is time consuming and it is only designed for frontal face. Shih [14] proposed a color based method shows reasonable

performance in terms of detection rate. However, the detection would fail under extreme lighting condition.

As mentioned above, extreme lighting conditions (such as colored lighting source and bias lighting direction), different subject poses, glasses, races, and complex background are factors to make face detection difficult. To solve the problem caused by these factors, we will propose a method to extract the horizontal eye line of a face and some eye-like regions under various environments. The extracted eye line and eye-like regions can be further used to help extract the true positions of eyes and face. The proposed method contains several steps. First, we will use skin colors to extract candidate skin regions. Next, an eye-like region detector based on intensity and color information will be provided to explore all possible eye-like regions in each candidate skin region. A lighting distribution based algorithm will then be presented to remove some false eye regions. Finally, based on the extracted eye-like regions, the horizontal eye line for a candidate skin region can be located using the gray level distribution of the skin region. The eye line is used to detect a face with/without glasses under various poses and environments.

1.2 THE MAIN PROBLEMS AND CURRENT STATE

In this dissertation, we will provide methods to deal with the problems of face detection including: facial feature extraction, poses identification and face detection.

These three problems are defined as follows:

(1) *Facial feature extraction*: given an image, to develop a method extracting facial features. It is an important process for face detection since different lighting environments will impact the performance of face detection.

(2) *Pose identification*: given a face candidate, to develop a method to automatically identify what pose of the face.

(3) *Face detection under various lighting environment*: given an image, to develop a method to locate the face under various lighting environment.

Actually, the three problems are logically sequenced. The solutions to these three problems can be used as the basic components for developing a fully automatically face system. As mentioned above, most of research efforts have been spent on face detection. However, some points still remain to be solved.

1.2.1 The Main Problem of Face Detection

Although face detection has been studied for more than 20 years, developing a human face detector under various environments is still a challenging task. Some factors make face location difficult. One is the variety of colored lighting sources, another is that facial features such as eyes may be partially or wholly occluded by shadow generated by a bias lighting direction; others are races and different face poses with/without glasses. As mentioned above, extreme lighting conditions (such as colored lighting source and bias lighting direction), different subject poses, glasses, races, and complex background are factors to make face detection difficult. To solve these problems, we propose a method to detect a face with/without glasses under various poses and environments. Fig. 1.1 shows the block diagram of the proposed face detector. First, we use skin colors to extract candidate skin regions. Next, shape information is used to judge whether a skin region contains a shoulder part. If yes, a skin projection based method will be applied to remove it and the head part is labeled as a face candidate for further processing. For each face candidate, a set of geometric constrains are firstly applied to judge if the face candidate is a profile face. If no, some eye-like rectangles are first extracted. Then, based on these eye-like rectangles, a horizontal eye line and a vertical symmetric line are located to judge if the face candidate is a non-profile face.

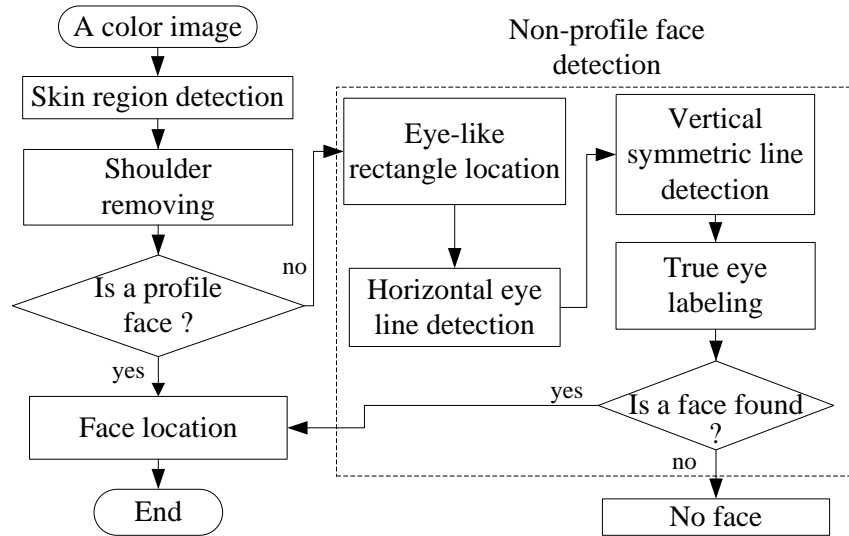


Fig. 1.1. The block diagram of the proposed face detector.

1.3 SYNOPSIS OF THE DISSERTATION

The rest of the dissertation is organized as follows. Chapter 2 describes the proposed horizontal eye line detection and eye-like region detection method. The face-pose identification method and face detection method with different poses under vary environment are proposed in Chapter 3. Some conclusions and future research directions are drawn in Chapter 4.

CHAPTER 2

A NOVEL METHOD FOR HORIZONTAL EYE LINE DETECTION UNDER VARIOUS ENVIRONMENTS

In this chapter, we will propose a method to extract the horizontal eye line of a face and some eye-like regions under various environments. It contains several steps. First, we will use skin colors to extract candidate skin regions. Next, an eye-like region detector based on intensity and color information will be provided to explore all possible eye-like regions in each candidate skin region. A lighting distribution based algorithm will then be presented to remove some false eye regions. Finally, based on the extracted eye-like regions, the horizontal eye line for a candidate skin region can be located using the gray level distribution of the skin region.

2.1 INTRODUCTION

In a drowsy driving warning system, driver's mental condition can be revealed by measuring the eye movement from the eye location [1]. How to locate the eyes of a driver under extreme lighting condition is an important issue for a successful intelligent transportation system. Detecting the eye line can help locate eyes. Moreover, the detected eye line can be used to do face detection. Although face

detection has been studied for more than 20 years, developing a human face detector under various environments is still a challenging task. Some factors make face location difficult. One is the variety of colored lighting sources, another is that facial features such as eyes may be partially or wholly occluded by shadow generated by a bias lighting direction; others are races and different face poses with/without glasses. In order to overcome this problem, Hjelmas and Low [2] have presented an extensive survey of feature-based and image-based algorithms for face detection. In feature-based methods, the most widely used technique is skin color detection based on one of the color models. Most Image-based approaches are based on multiresolution window scanning to detect faces at all scales, making them computationally expensive. Also, Yang [3] classifies the approaches into four categories: knowledge-based, feature invariant, template matching, and appearance-based and address one of the difficulty of face detection is lighting source varies significantly. Several methods [4-18] had been proposed by now. Some [4-6] use eigenface, neural network and support vector machine (SVM) to detect faces of restricted poses under normal lighting condition. Chow [7] proposed an approach to locate facial features (such as eyes, eyebrows) by using an eigenmask under different lighting conditions. However, the method fails to locate the facial features for some kinds of poses (such as near-profile and profile face). Shin and Chuang [8] uses shape

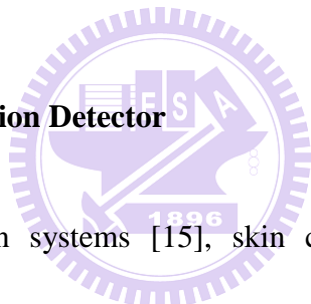
and contour information to locate face in a plain background and normal lighting condition. However, it is difficult to detect the predefined shape of a face in a complex background and various lighting conditions. Wu *et al.* [9] proposed an algorithm to locate face candidates using eye-analogue segments information. The algorithm would fail when a person wears glasses. Some methods [10-12] use mosaic, edge and mathematical morphology to detect eyes, however, these methods would fail to locate eyes when a face is under a poor lighting condition (such as bias-light source). Wang [13] proposed a boosted based algorithm to detect a face. But the training phase of the method is time consuming and it is only designed for frontal face. Shih [14] proposed a color based method shows reasonable performance in terms of detection rate. However, the detection would fail under extreme lighting condition.

As mentioned above, extreme lighting conditions (such as colored lighting source and bias lighting direction), different subject poses, glasses, races, and complex background are factors to make face detection difficult. To solve the problem caused by these factors, we will propose a method to extract the horizontal eye line of a face and some eye-like regions under various environments. The extracted eye line and eye-like regions can be further used to help extract the true positions of eyes and face.

2.2 THE PROPOSED METHOD

The proposed method contains several steps. First, we will use skin colors to extract candidate skin regions. Next, an eye-like region detector based on intensity and color information will be provided to explore all possible eye-like regions in each candidate skin region. A lighting distribution based algorithm will then be presented to remove some false eye regions. Finally, based on the extracted eye-like regions, the horizontal eye line for a candidate skin region can be located using the gray level distribution of the skin region.

2.2.1 The Proposed Skin Region Detector



In several face detection systems [15], skin color plays a major role for segmenting face regions from an image. There are several color models utilized to label pixels as skin, these include RGB, HSV and YCrCb color spaces [16-18]. Since a face image is usually taken under unpredicted lighting conditions, RGB color-model is sensitive to light and unsuitable for representing skin color. In YCbCr model, Cr represents the red degree [15]. Since human skin color tends to red, we will adopt YCbCr color model.

2.2.1.1 Skin Region Extraction

Some skin detectors [14, 15] work on skin-color distribution modeling. Different illumination levels such as indoor, outdoor, highlight and shadows will change the color of skin. However, the size of the training set and variety of samples used in the training may be impact on the skin detector performance [15]. That means the trained skin model may be required extensive training and fine tuning to archive an optimal performance for different training samples. In this dissertation, the skin detector works on explicit skin-color space thresholding.

As mentioned above, human skin color tends to red despite the race, we will use Cr value to determine whether a pixel is skin or not. If a pixel's Cr value is larger than a predefined threshold, then it will be considered as a skin pixel. In this dissertation, a threshold Cr_{th1} is adopted to bound skin color. On the other hand, the Cr value for skin pixels will be lowered under a high lighting with blue color biased source environment, another lower threshold Cr_{th2} will also be used. That is, for an input image, we first use Cr_{th1} (here, 142 is used) as threshold to get candidate skin regions. Then, for the same input image, we use Cr_{th2} (here, 132 is used) as threshold to get another candidate skin regions. Fig. 2.1 shows some results of applying the skin region extractor on different racial faces. From this figure, we can see that whatever the light-conditions and the human races change, most skin pixels are extracted (white

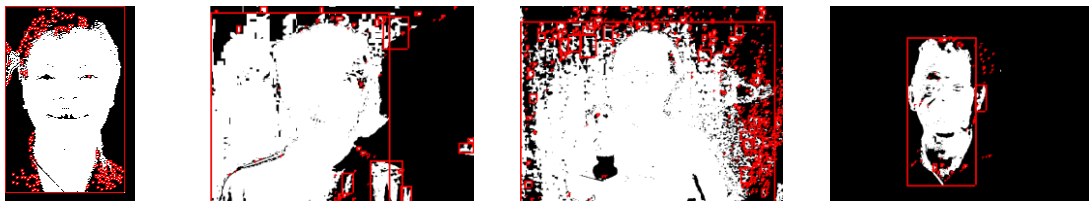
pixels). After skin pixels are extracted, those connected skin pixels are grouped to form a set of candidate skin regions, each region is bounded by a red rectangle (see Figs. 2.1(b) and 2.1(c)). From Fig. 2.1, we can see that some extracted candidate skin regions are not true skin. In the next section, we will present a filter to remove some false skin regions.



(a) Four images taken under normal, bias colored and high lighting environments.



(b) The extracted candidate skin regions using threshold 142.



(c) The extracted candidate skin regions using threshold 132.

Fig. 2.1. Some results of applying skin region extractor on different races under various lighting environments.

2.2.1.2 Skin Region Filter

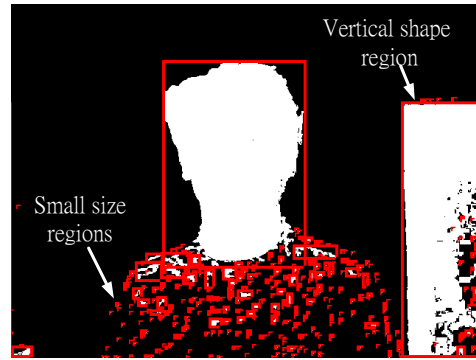
The skin region filter is designed to remove false skin regions based on four features: size, shape, uniform-content and geometric relation. Each feature can be

used to remove several false regions. The detail is described as follows.

- (1) **Size:** A candidate skin region will be removed if the width or height of its bounding rectangle is less than a threshold S_{th} . In this dissertation, we set S_{th} to be 15 (see Figs. 2.2(b) and 2.2(d)). On the other hand, if the width of its bounding rectangle is larger than a half of the input image width, the region will also be removed.
- (2) **Shape:** In general, the shape of a face should be like an ellipse. Thus, if a region looks like a horizontal or vertical thin bar, it should be removed (see Fig. 2.2(b)). Note that if a region with $H_s/W_s \geq V_{th}$, it is considered as a vertical thin bar. On the other hand, if a region with $W_s/H_s \leq H_{th}$, it is considered as a horizontal thin bar. Here, H_s and W_s are the height and the width of a region, and in the dissertation, we set V_{th} to be 2.5 and H_{th} to be 0.5.
- (3) **Uniform-Content:** A uniform content region should not be a face due to that the eyes and eyebrows are in face. Thus, if the standard derivation of the gray values in the bounding rectangle of a candidate skin region is less than a predefined threshold U_{th} (10), the region will be removed (see Fig. 2.2(f)).
- (4) **Geometric relation:** If two candidate skin regions crossover and the small one is less than a half of the big one, then the small one is removed (see Fig. 2.2(g)). On the other hand, if a small region is totally covered by another big region, then both regions are preserved (see Fig. 2.2(g)). For this case, in the later process, the small one is first used to detect the eye line. If an eye line is found, the detection for the big one will be skipped.



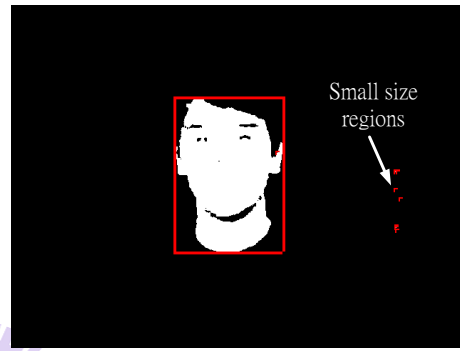
(a) A face image.



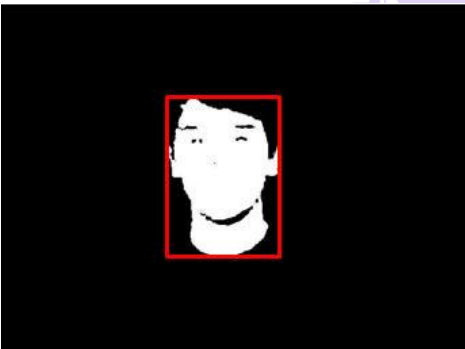
(b) Candidate skin regions using 132 as threshold.



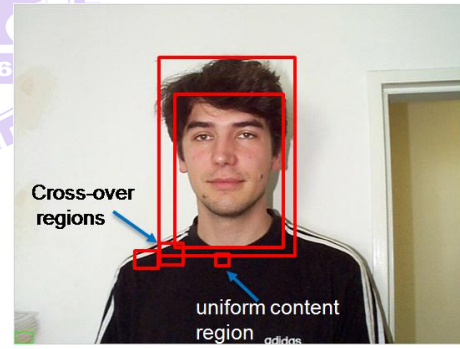
(c) The result of removing regions of small sizes and improper shapes in (b).



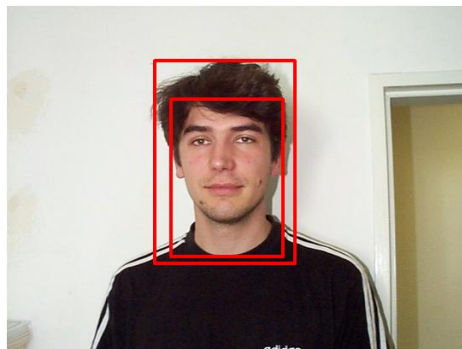
(d) Candidate skin regions using 142 as threshold.



(e) The result of removing small regions in (d).



(f) The result of merging (c) and (e) and imposed on (a).



(g) The result of removing uniform content regions and cross over regions in (f).

Fig. 2.2. Some examples for illustrating the skin region filter.

2.2.2 The Proposed Eye-Like Region Detector

In this section, we will provide a method to extract eye-like regions from those remaining candidate skin regions. Fig. 2.3 shows the block diagram of the proposed eye-like region detector. First, based on intensity and skin color information, some eye-like regions are extracted. Next, some false regions are removed. Since the true eye may not be located in the center of the eye-like region, a refining algorithm is presented to adjust the eye-like region. Finally, some false eye-like regions appearing in hair or shadow part will be removed, and isolated eye-like regions are also removed.

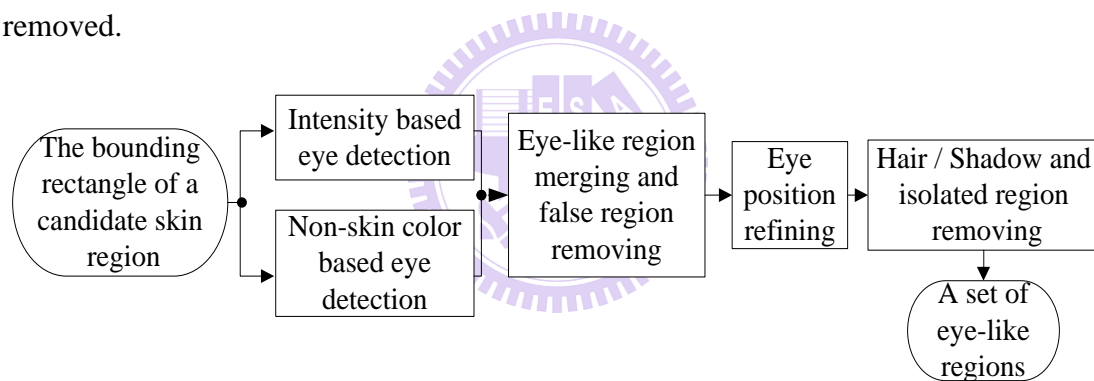


Fig. 2.3. The block diagram of the eye-like region detector.

In [7], luminance contrast and eye shape are used to detect an eye. However, under various lighting environments, unpredicted shadow appearing on face makes eye detection difficult. Fig. 2.4 shows some face images taken under various lighting environments. To treat this problem, we use two fundamental eye properties to extract eye-like regions. The first property is that the pupil and iris's gray values are lower than the gray values of skin. Based on this property, an intensity based technique is

provided to locate eye-like regions in the bounding rectangle of each candidate skin region. The second property is that under non-colored lighting source condition, the eye color will be very different from skin color. Based on this property, we consider those small non-skin color regions in the bounding rectangle of a candidate skin region as eye-like regions.



Fig. 2.4. Subjects taken under various lighting environments.

2.2.2.1 Intensity Based Eye-Like Region Extraction

After extracting candidate skin regions, an intensity based detector will be first provided to extract eye-like regions from the bounding rectangle of each candidate skin region. Fig. 2.5 shows its flowchart.

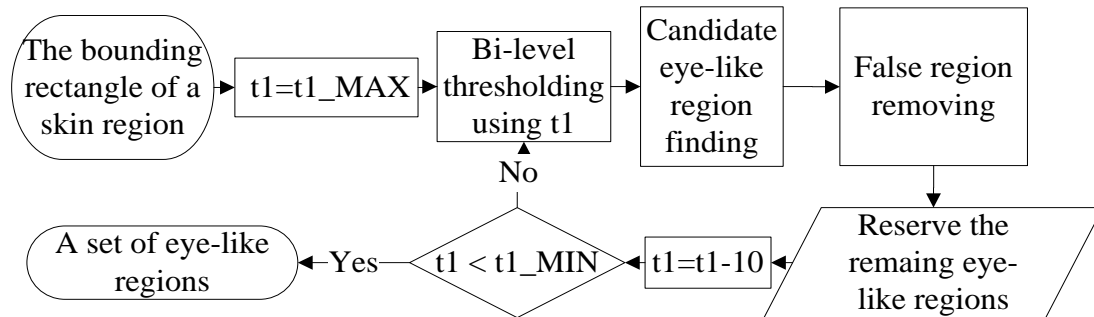
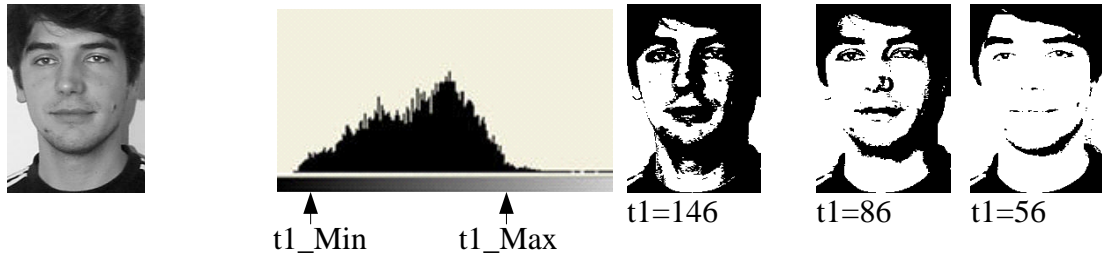


Fig. 2.5. The flowchart of the intensity based eye-like region detector.

Based on the fact that the pupil's gray value are lower than skin gray values, the detector adopts a bi-level thresholding to get the eye-like regions. First, for a candidate skin region, several different threshold values, $t1$, are used for bi-level thresholding, the maximum and minimum of $t1$ are named as $t1_MAX$ and $t1_MIN$, respectively. $t1_MAX$ is set as the average gray value of 10% pixels with maximum gray values in the candidate skin region and $t1_MIN$ is set to be the average gray value of 5% pixels with minimum gray values in the candidate skin region. At beginning, a bi-level threshold operator with $t1_MAX$ as the initial threshold is applied to extract eye-like regions, and then the operator is applied literately with $t1$ reduced by a value I_{th} (here, 10 is used) each time until $t1$ reaches $t1_MIN$. Fig. 2.6 shows an example for evaluating $t1_MAX$ and $t1_MIN$, and the result of applying the intensity based eye detector.



(a) A face region. (b) The histogram of detected gray pixels within the skin region. (c) Bi-level thresholding results of (a) using different thresholds.

Fig. 2.6. An example of the bi-level thresholding results.

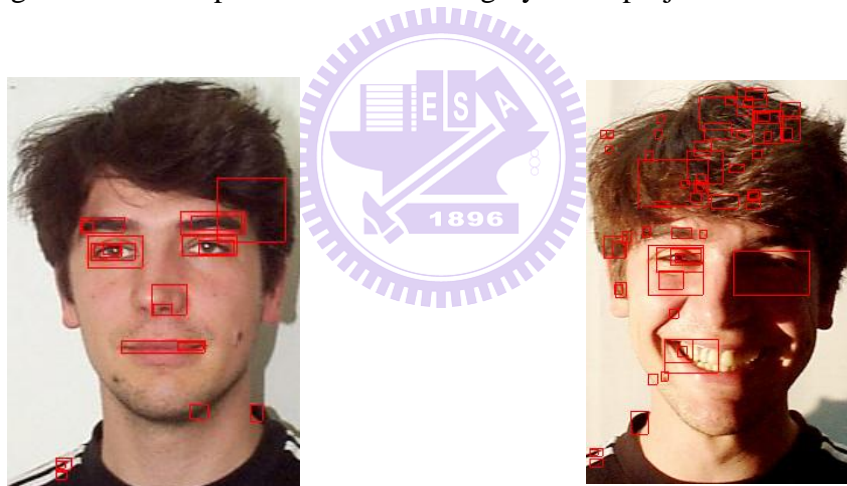
After applying a bi-level thresholding operator at each time, every resulting black region is considered as a candidate eye-like region. However, some of these black regions are false ones. Here, we provide three filters to remove these false eye-like regions. One is geometric filter, which is used to remove too small or too large regions. A black region with size less than S_{th2} (here, S_{th2} is set as 5×5) or its width (height) larger than S_{th3} (here, we set S_{th3} to be $1/4$ skin region width) will be removed. One is statistical filter, if the standard derivation of the gray values of an eye-like region is lower than a predefined threshold, the region is considered as a false eye-like region and then removed. Here the threshold we chose is 10. From the structure of an eye, we can see that the iris and pupil's gray levels are lower than skin. Based on this property, the third filter called projection filter is provided to remove each black region with its average gray value larger than that of the upper or lower neighboring areas. A horizontal gray value projection histogram (see Fig. 2.7) is used to implement the filter. If the histogram value of the middle position is larger than the histogram value

of the highest peak above or below the position, we removed the region. Fig. 2.8 shows the final result of applying the intensity based eye-like region detector to a face image, each extracted eye-like region is enclosed by a red rectangle.



- (a) The projection region extended from eye-like region. (b) The projected histogram of the projection region in (a).

Fig. 2.7. An example of the horizontal gray value projection for an eye.



- (a) The result of a normal lighting source face. (b) The result of a colored and left bias lighting source face.

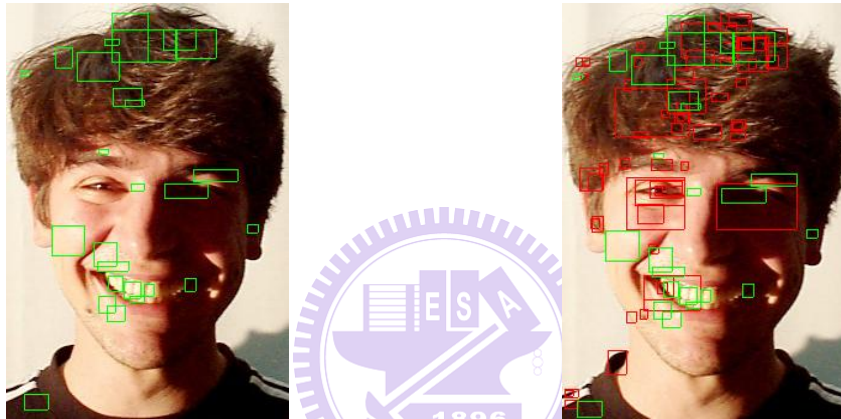
Fig. 2.8. The results of applying the intensity based algorithm.

2.2.2.2 Non-Skin Color Based Eye Detector

Under a very low lighting environment, an eye-like region extracted by intensity based detector may contain a large non-eye part (see the left eye in Fig. 2.8(b)). This will make the later process hard. To treat this problem, we use non-skin color

information to locate eyes. First, non-skin color regions within a candidate skin region are considered as eye-like regions. The three false eye-like region filters described in Section 3.1 are then used to remove some false regions. Fig. 2.9(a) shows the result of applying the non-skin color based eye detector. Each eye-like region is enclosed by a green rectangle. From this figure, we can see that the left eye was precisely located.

Fig. 2.9(b) shows the combined result of Fig. 2.8(b) and Fig. 2.9(a).

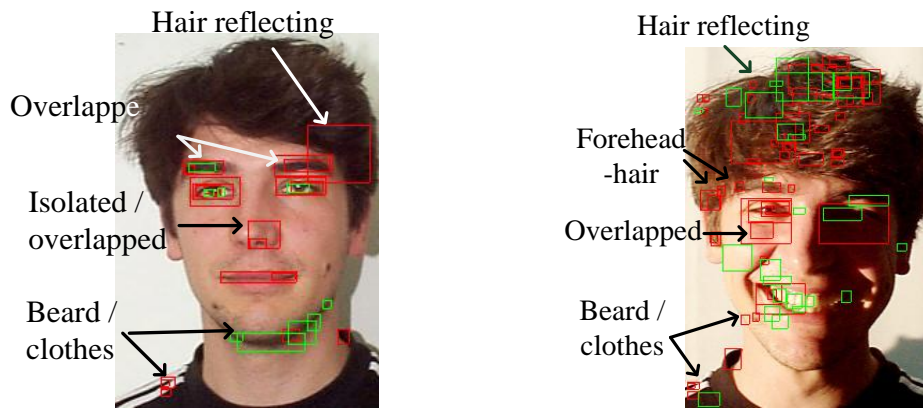


(a) The eye-like regions located based on non-skin color information. (b) The combined result of Fig. 2.8(b) and (a).

Fig. 2.9. The result of applying the proposed intensity and non-skin color based eye detectors.

2.2.3 False Eye-Like Region Removing

The obtained eye-like regions still contain some false regions; we can classify these regions into five classes: overlapped, hair reflecting, beard/clothes, isolated, and forehead-hair (see Fig. 2.10). In the following, we will provide some procedures to remove these regions.

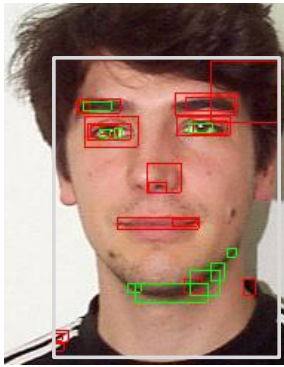


(a) An example to show four classes of false eye-like regions. (b) Another example to show forehead - hair class.

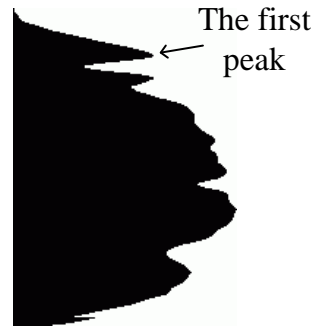
Fig. 2.10. Five false region classes.

2.2.3.1 Overlapping/ Hair Reflecting/ Beard/ Clothes False Region Removing

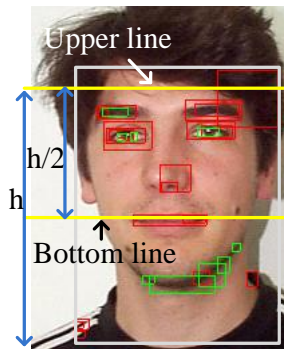
For the overlapping class, if one eye-like region totally covers a small eye-like region, the covering region is removed. To remove hair reflection regions, first, a horizontal gray level projection histogram (see Fig. 2.11(b)) is created. Then, the first peak location from the top of the projection histogram is defined as an upper line which usually indicates the forehead. Each eye-like region intersecting the upper line is removed. For the beard/clothes class, we first define a bottom line with distance $h/2$ from the upper line, where h is the distance between the upper line and the bottom of the skin region. Then all eye-like regions below the line are removed. Fig. 2.11(d) shows the result of applying the removing procedure to Fig. 2.11(a).



(a) A bounding rectangle of a candidate skin region.



(b) The horizontal projection histogram of the bounding rectangle in (a).



(c) The detected upper and bottom lines.



(d) The result after applying the removing procedure presented in Section 2.2.3.1 to (a).

Fig. 2.11. The result of applying the false eye-like removing procedure.

2.2.3.2 Eye Position Refining And Isolated False Region Removing

For a non-rotated human face, an eye pair should be located at a near-horizontal line. Thus for an eye-like region, if we can not find another region at its left or right, then this region is called isolated region and should be removed. However, under various environments, a true eye may not be located at the center of the extracted eye-like region (see Fig. 2.12(a)), we are then unable to know where the eye is. In order to treat this problem, we provide an eye position refining method to locate the true eye position in the eye-like region. First, we classify the frontal images of HHI










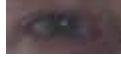











into three classes according to the lighting condition: normal, high light and low light.

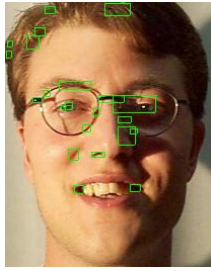
Next, one image is selected randomly from each class. Considering the effect of glasses, the normal class has two images selected with/without wearing glasses. For each image, three templates are extracted (see Table 2.1). An eye area including eyelid is taken as the first template. In the first template, an area only containing eye is then taken as the second template. In the second template, an area only containing eye ball is taken as the third template. After obtaining the twelve templates, we use each template to mask each eye-like region by sliding the template pixel by pixel in the region to find the best matched area for each class. Covariance is used to measure the similarity. If an eye-like region with the located eye center does not lie in the true eye, we take three new templates from the image containing the region using the previous method. The procedure is repeated until all eye-like regions are processed. After the procedure is finished, we find that nine new templates (see Table 2.1) belonging to three new classes are extracted. These three new classes are left-side (the eye sees the left side), right-side (the eye sees the right side) and bias lighting. Since the images in HHI have different face sizes, the extracted templates will also have different sizes. After we make the eye templates, a mentioned similarity matching procedure is adopted to find the best matched area for testing image. We use the similarity function is defined as follows.

$$\text{cov}(A, B) = \frac{\sum_{i=1}^n (a_i - A_{mean})(b_i - B_{mean})}{n}, \quad (2-1)$$

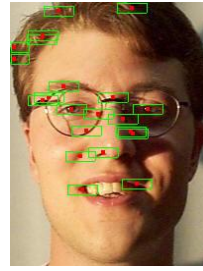
where $A = [a_1, a_2, \dots, a_n]$ and $B = [b_1, b_2, \dots, b_n]$ are two areas, and A_{mean} and B_{mean} are the mean values for each area. Thus, we can obtain six best matched areas for each eye-like region. Then the center of the best matched area with the highest Cb/Cr value (the most possible non-skin area) is considered as the best eye center in the eye-like region (see the red points in Fig. 2.12(b)). Finally, all eye-like regions are refined to be a $w \times h$ region centering at the best eye center. In the dissertation, we set w to 30, h to 10 (see the green rectangles in Fig. 2.12(b)). After an eye-like region is refined, we judge if it is an isolated region. If yes, the region will be removed. Fig. 2.13(b) shows the result of removing those refined isolated false regions.

Table 2.1 The templates for six eye classes.

Eye Type	Template Images					
Normal						
High light						
Low light						
Left-side						
Right-side						
Eye ball						

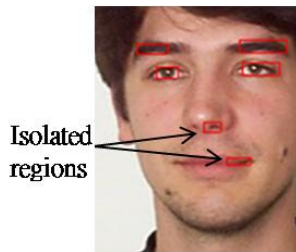


(a) An example to show an eye not located at the center of the eye-like region.

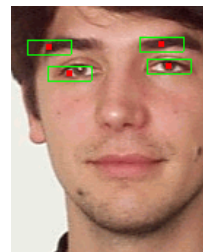


(b) The result of applying the eye position refining procedure to (a).

Fig 2.12. The result of applying the eye position refining procedure.



(a) An example to show isolated regions.



(b) The result of removing the isolated regions in (a).

Fig. 2.13. The result of removing isolated false regions.

2.2.3.3 Forehead-hair false regions removing

Before describing the proposed forehead-hair false region removing algorithm, we will define the bounding rectangle for a face. Let R be the minimum rectangle containing all eye-like regions, then R is expanded in the vertical direction to form a square rectangle (see Fig. 2.14). The square rectangle is defined as the bounding rectangle of the face.

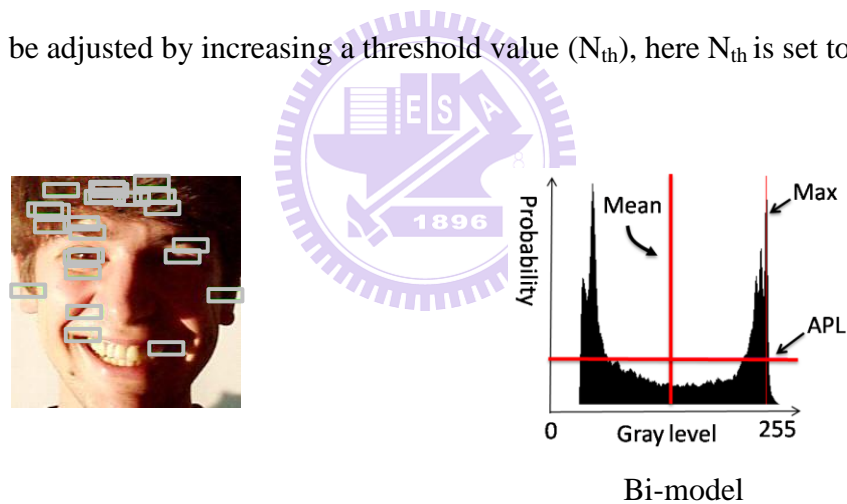


(a) The minimum rectangle containing all eye-like regions. (b) The expanding result of (a).

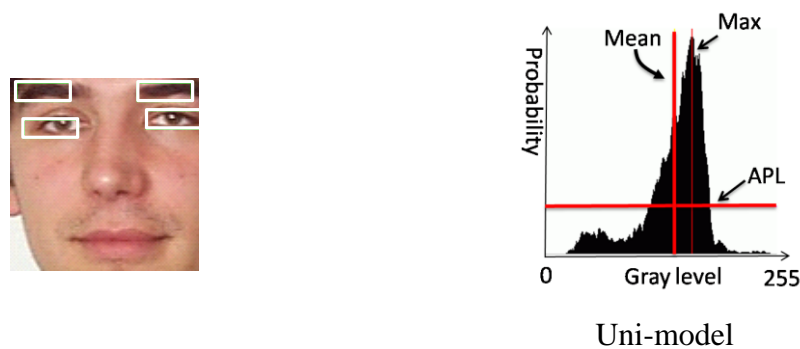
Fig. 2.14. An example to illustrate the bounding rectangle of a face.

In general, pixels in hair or shadow regions caused by a bias lighting have lower gray levels than pixels in a face. Based on this property, we find that if the bounding rectangle of a face contains a part of lower gray region (hair or side shadow), its gray-level histogram, $h(x)$, will be a bi-model distribution; else the histogram will be a uni-model distribution (see Fig. 2.15). To distinguish these two models, an average probability horizontal line (APL) $Y = T$ with $T = \frac{1}{256} \sum_{x=0}^{255} h(x)$ is first defined. If the number of the intersection line segments between the APL and the histogram is two, a bi-model distribution is identified (see Fig. 2.15(a)). Otherwise, if only one line segment exists, a uni-model distribution is identified (see Fig. 2.15(b)). For a face bounding rectangle, those eye-like regions appearing in the hair area should be removed. Based on the above discussion, a procedure is proposed to remove forehead-hair false regions. To identify the hair part, we set the initial bi-level threshold t_0 as average gray value of the face rectangle for a bi-model histogram. If the face is a uni-model, we set the initial threshold t_0 as

$$\min \left\{ t \mid ACC(t) > 5\% \times \sum_{x=0}^{255} h(x) \right\}$$
 with $ACC(t) = \sum_{x=0}^t h(x)$. The pixels with gray levels less than t_0 are labeled as black pixels. However, the threshold may be improper such that over-segmentation or under-segmentation (see Fig. 2.16) will occur. To treat this problem, the threshold t_0 will be adjusted according to the following procedure. After applying the bi-level thresholding, if no two eye-like regions contain over O_{th} (30%) black pixels, an over-segmentation is detected (see Fig. 2.16(b)). The t_0 should be adjusted to be larger. The adjustment rule is to take the nearest right valley to the current t_0 on the histogram as a new threshold t_0 . If there is no valley is found, then the t_0 will be adjusted by increasing a threshold value (N_{th}), here N_{th} is set to be 5.



(a) A bias lighting face and its corresponding histogram.



(b) A normal lighting face and its corresponding histogram.

Fig. 2.15. Two examples to show different models of gray level histograms.



(a) Original image.



(b) An over-segmentation obtained after applying the threshold process to (a).



(c) Original image.



(d) An under-segmentation obtained after applying the threshold process to (c).

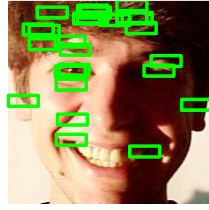
Fig. 2.16. Two examples to show the over-segmentation and under-segmentation.

On the other hand, if all eye-like regions are located in a big black area, an under-segmentation is detected (see Fig. 2.16(d)), $t0$ will be refined to smaller. The adjustment rule is to take the nearest left valley to the current $t0$ on the histogram as a new threshold $t0$. If there is no valley was found, then the $t0$ will be decreased by N_{th} .

The above procedure is repeated until no over or under segmentation occurs. In order to prevent a dangling situation, any used valley will be removed to avoid reusing.

Then the bi-level thresholding operator with the final threshold is applied to identify the dark regions in the face bounding rectangle (see Fig. 2.17(b)). If a dark area is hair, it should be large in the face bounding rectangle. If a dark area is iris, it should be small. So, for a dark region with area larger than $\frac{1}{4}$ face bounding rectangle, we

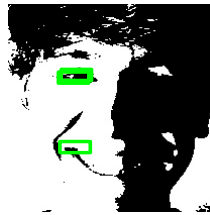
consider it as a hair region, and all eye-like regions in the hair region are removed. In addition, if a dark region has the ratio of the height over the width larger than a threshold 2.5, we will remove the region. On the other hand, since the iris and eyelids have lower gray level than the other part of an eye, if an eye-like region contains less than 3% dark pixels, the region is also removed. Note that any eye-like region locating at the shadow part will also be removed by the above procedure (see Fig. 2.17(c)). To resume these regions, we use a previously mentioned fact if an eye is found, then the other eye should be found at its right side or left side. For each remaining eye-like region, R , we first draw three lines, $L1$, $L2$ and $L3$, passing through the center of R . $L1$ is a horizontal line, the angles between $L1$ and $L2$ (or $L3$) is a predefined angle A_{th} (here, $\pm 10^\circ$ is used) (see Fig. 2.18(a)) for a bi-model histogram face. Then a bounded range, BR , is defined to be the area bounded by $L2$ and $L3$. Based on the bounded range, all eye-like regions in the bounded range removed by the forehead-hair false region removing procedure will be got back (see Fig. 2.18(b)). Now the remaining eye-like regions are used to form a new bounding rectangle (see the dotted rectangle in Fig. 2.18(b)) which will be used to determine the true horizontal eye line.



(a) The eye-like regions before applying the removing procedure.

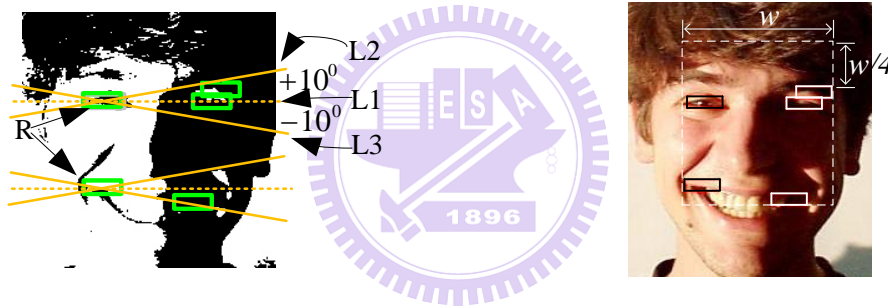


(b) The result of applying the bi-level threshold to (a).



(c) The result of removing the forehead - hair false regions in (b).

Fig. 2.17. The result of applying the forehead-hair false region removing procedure.



(a) The bounded ranges for resuming the removed eye-like regions in the shadow area of a face.

(b) Three eye-like regions (marked by white rectangles) got back.

Fig. 2.18. An example to explain the eye-like region resuming procedure.

2.2.4 The Proposed Horizontal Eye Line Detector

In this section, we will propose a novel eye line detection algorithm. Because eyes and eyebrows are regions with low gray-level, their locations will correspond to the local valley of the horizontal projection histogram. So the eye line detection

algorithm can be reduced as a valley finding procedure on the horizontal gray value projection histogram. In the valley finding procedure, all valleys are taken first, then the small valleys with value smaller than a threshold D_{th} (here, we set D_{th} to be 3) will be removed. If some valleys are located at similar locations with distance between two neighboring ones smaller than a threshold B_{th} (here, B_{th} is set to be 3), the deepest valley will be kept, others are discarded.

The detail is described as follows. For a given bounding rectangle of a face, if there is only one valley in the projection histogram and there is a pair of eye-like regions near the valley (see Fig. 2.19(a)), then the horizontal line (the green line in Fig. 2.19(a)) is defined as the line passing through the vertical middle location between the two eye-like regions. If there are two valleys, and if the distance between these two valleys is less than a threshold and there are at least one pair of eye-like regions near the two horizontal lines passing through these two valleys, respectively, then the upper one (the red line in Fig. 2.19(b)) is considered as the eyebrow line and the other (the green line in Fig. 2.19(b)) is the eye line. If there are 3 or 4 valleys, we will execute the following procedure (called procedure A). First, we define the distance between the first and second valleys as $D1$ and between the second and third valleys as $D2$. Then we check $D1$ and $D2$ to see if they satisfy the following two conditions.

Condition 1: $D1 < s1 \times D2$.

Condition 2: $D1 > s2 \times D2$.

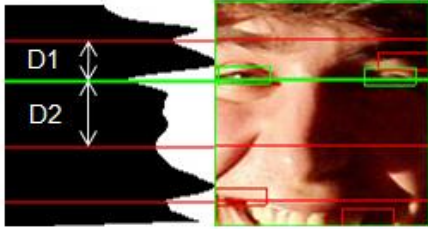
Note that $s1$ and $s2$ are set as 1.2 and 3.0 in the dissertation.

If valleys satisfy Condition 1 and there exist two eye-like regions near the second valley line, then the second valley line (the green line in Fig. 2.19(c)) is considered as eye line. On the other hand, if Condition 2 is satisfied and there exist two horizontal eye-like regions near the first valley line, then the first valley line (the green line in Fig. 2.19(d)) is considered as the eye line. If both conditions are not satisfied, we will apply procedure A described above to the next three valleys (the second, third and fourth). If no eye line is found, we conclude that the region is not a face. If an eye line is found, any eye-like region which distance from the eye line larger than 1.5 height of the eye-region will be removed. Then, the remaining regions will be used to outline a face rectangle where the left/right position is the most left/right region. The top position of the rectangle is defined as the upper location of the most top region from 3 height of an eye-like region. And the height of the rectangle is defined as 1.3 width of the rectangle.

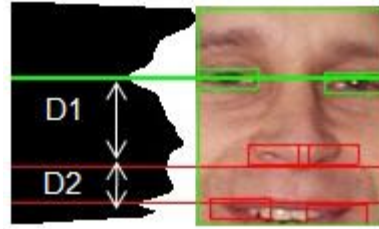


(a) Only one valley case.

(b) Two valleys case.



(c) $D1 < s1 \times D2$ with two eye-like regions near the second valley line.



(d) $D1 > s2 \times D2$ with two eye-like regions near the first valley line.

Fig. 2.19. Some examples for the horizontal projection.

2.3 EXPERIMENTAL RESULTS

In order to show the effectiveness of the proposed method, we apply the proposed method to HHI face database [19] of 206 images (see Fig. 2.20) and Champion face database [20] of 227 images (see Fig. 2.21). We also collect some other images from Internet, MPEG7 video clips, PIE database [21] to evaluate the performance. These contain face images of different racial persons under different kinds of lighting conditions (such as overhead, side and color lightings). Since one of our main applications of the method is for a drowsy driving warning system and in most cases, a driver's head position is usually frontal or near profile. Since the head is rarely in-plane rotated when people are driving, in dissertation only detect the horizontal eye line and allowed in-plane rotated in $A_{th} (\pm 10^\circ)$ degrees. The system is

implemented by Java language on a laptop with Pentium(R) M processor 1.4 G Hz CPU. In Fig. 2.21, some successful results of applying our method to some faces with different skin colors. The successful results for faces with different poses and eye glasses are shown in Fig. 2.20. Even there is shadow on a face; the eye line can also be detected (see Fig. 2.22).



Fig. 2.20. A part of images from HHI face database with different poses and eye-glasses.



Fig. 2.21. A part of images from Champion face database with different skin colors.

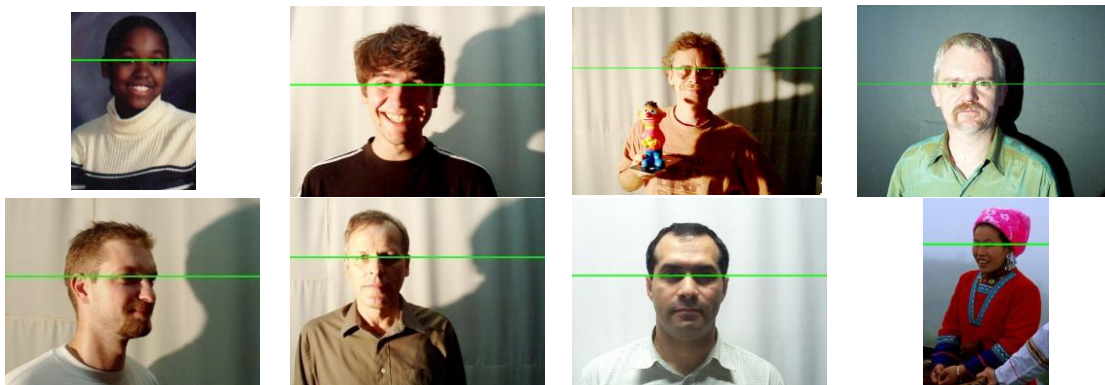


Fig. 2.22. The detection results for persons with shadows and various lighting sources on their faces.

If the detected eye line is located under eye brow and near eyes, we consider it to be a correct detection; otherwise it is an error detection. The tolerance value is the height of an eye-like region. If no horizontal eye line is detected for a face image, we call it a missing detection. Fig. 2.23 shows the successful detection results for a set of images from the Internet, MPEG7 video clips and PIE database images. For HHI database, the correct detection rate is 95.63% and the correct detection rate on Champion database is 94.27%. Fig. 2.24 shows the error and missing detection results.

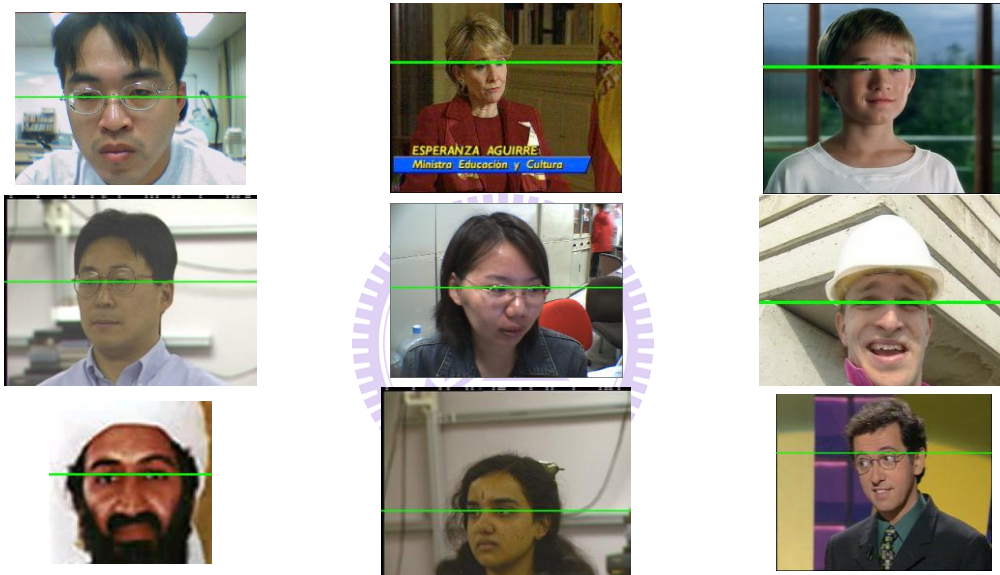


Fig. 2.23. The detection results of face images collected from the Internet, MPEG7 video clips and PIE database.



Fig. 2.24. Some error/missing examples.

CHAPTER 3

A NOVEL FACE DETECTION METHOD UNDER VARIOUS ENVIRONMENTS

In this chapter, we will propose a method to detect a face with different poses under various environments. It consists of three phases: profile feature extraction, non-profile face detection and the true eye locating.

3.1 INTRODUCTION

Detecting a face from an image is the first step for face applications. Even it has been studied for many years; detecting a face under various environments is still a challenge work. Some factors make face detection difficult. One is the variety of colored lighting sources, another is that facial features such as eyes may be partially or wholly occluded by shadow generated by a bias lighting direction; others are races and different face poses with/without glasses. Several methods [4-18] had been proposed to detect faces. Some [4-6] use eigenface, neural network and support vector machine (SVM) to detect faces of restricted poses under normal lighting condition. Hsu *et al.* [18] proposed a face detection algorithm that uses a nonlinear transform to overcome the variance of the skin tones. Some defined chroma

properties are used to extract the facial features and applied to verify a face. Chow *et al.* [7] proposed an approach to locate a face candidate under different lighting conditions and then applied an eigenmask based method to verify the face. However, these methods fail to locate faces with some kinds of poses (such as near-profile and profile face). Shin and Chuang [8] uses shape and contour information to locate face in a plain background and normal lighting condition. However, it is difficult to detect a face in a complex background and various lighting conditions. Wu and Zhou [9] proposed an algorithm to locate face candidates using eye-analogue segment information. The algorithm would fail when a person wears glasses. Some methods [10-12] use mosaic, edge and mathematical morphology to detect eyes, however, these methods would fail to locate eyes when a face is under a poor lighting condition (such as bias-light source).

3.2 THE PROPOSED METHOD

As mentioned above, various lighting conditions, different head poses, glasses, races, and complex background are factors to make face detection difficult. To solve these problems, we propose a method to detect a face with/without glasses under various poses and environments. Fig. 3.1 shows the block diagram of the proposed face detector. First, we use skin colors to extract candidate skin regions. Next, shape

information is used to judge whether a skin region contains a shoulder part. If yes, a skin projection based method will be applied to remove it and the head part is labeled as a face candidate for further processing. For each face candidate, a set of geometric constrains are firstly applied to judge if the face candidate is a profile face. If no, some eye-like rectangles are first extracted. Then, based on these eye-like rectangles, a horizontal eye line and a vertical symmetric line are located to judge if the face candidate is a non-profile face.

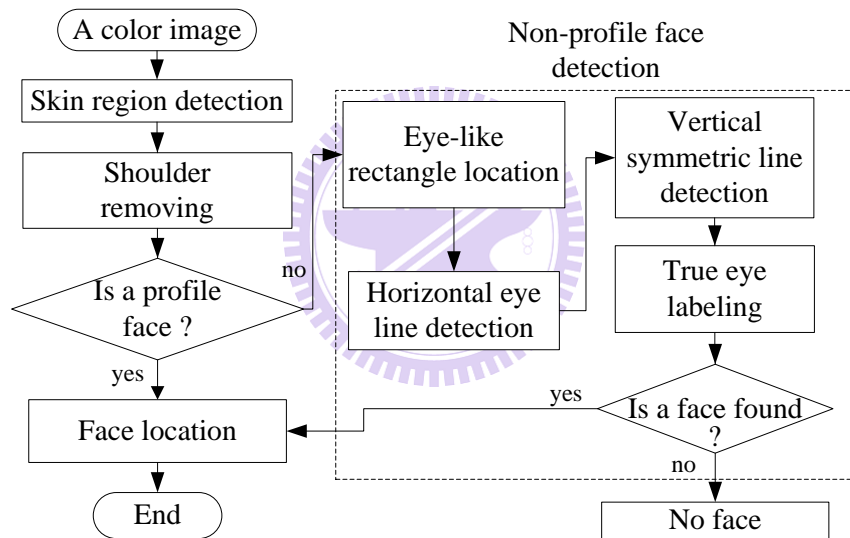


Fig. 3.1. The block diagram of the proposed face detector.

The remaining of this chapter is described as follows. In Section 3.2.1, a shape based algorithm is proposed to identify a profile face. Section 3.2.2 presents an algorithm to detect a non-profile face. Section 3.2.3 provides the true eye locating procedure. In Section 3.3, some experimental results based on HHI and Champion face databases are given to demonstrate the effectiveness of the proposed method.

3.2.1 Profile Detection

In this section, we will provide a method to determine if an image contains a profile face. First, the skin region detector proposed by Jing and Chen [22] is adopted to extract skin regions from an input image. Fig. 3.2 shows the extracted results from some face images with different poses taken under various lighting environments. Note that some pixels in clothes with color similar to skin are also labeled as skin. In the following, we will provide a method to remove these pixels.



Fig. 3.2. Some results of applying Jing-Chen skin region extractor.

3.2.1.1 Shoulder Removing Procedure

Fig. 3.3 shows an example of two men, one in Fig. 3.3(a) wears a T-shirt of skin-like color; and the other in Fig. 3.3(b) wears a T-shirt of non-skin color. Figs.

3.3(c) and 3.3(d) show the results of applying Jing-Chen method to Figs. 3.3(a) and 3.3(b), respectively. In Fig. 3.3(c), pixels on shoulder are labeled as skin pixels. Here, a projection-based procedure is provided to remove the shoulder part.

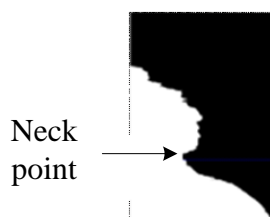


(a) A man wearing a T-shirt of skin-like color (b) A man wearing a T-shirt of non skin-like color.



(c) The detected skin region of (a).

(d) The detected skin region of (b).



(e) The horizontal projection of (c).

(f) The horizontal projection of (d).

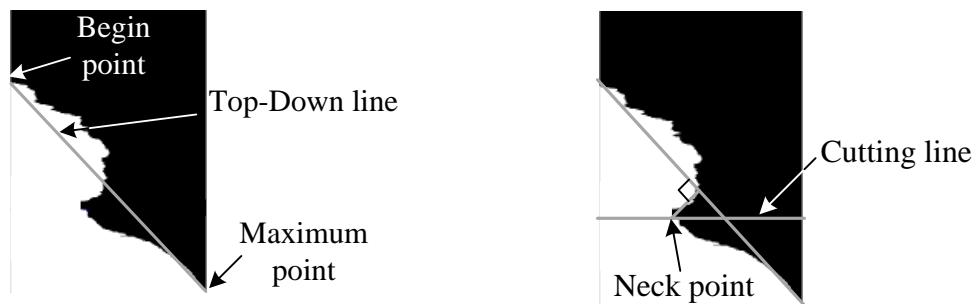
Fig. 3.3. An example showing two men wearing clothes with/without skin-like color.

First, we need to determine whether a skin region includes a shoulder part. In general, the shoulder of a person is wider than the head and neck. Based on this fact,

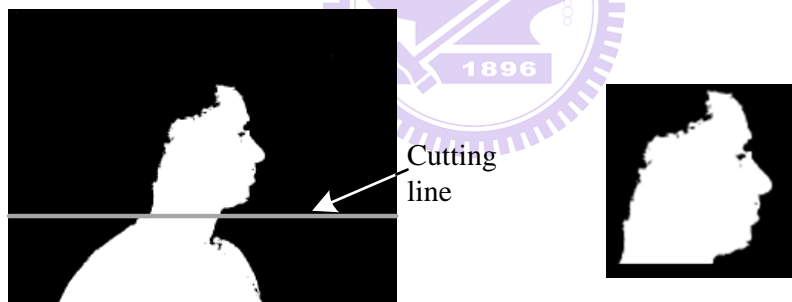
the horizontal projection of skin pixels is used to remove shoulder. Figs. 3.3(e) and 3.3(f) show the horizontal projections of skin pixels in Figs. 3.3(c) and 3.3(d), respectively. From these figures, we can see that if a shoulder exists, the number of the labeled skin pixels below the middle part of the whole skin region, L , should be larger than that of the upper one, U . Based on this fact, the inequation $\frac{L}{U} > \sigma_s$ is applied to judge whether a skin region includes a shoulder part. In this dissertation, the threshold σ_s is 1.4.

If a skin region is determined to have a shoulder, the neck location is considered as a good cutting point to remove the shoulder part. Since the neck is thinner than the other part of a human body, it will appear at the valley of the projection histogram (see Fig. 3.3(e)). Thus we take the valley point as the neck point. In order to locate the neck point, the begin and maximum points of the projection histogram are used to form a base line named as Top-Down line (see Fig. 3.4(a)). For each histogram point below the Top-Down line, we evaluate the distance from the histogram point to the Top-Down line. The point with maximum distance is considered as the neck point (see Fig. 3.4(b)). And the horizontal line passing the neck point is considered as the cutting line. Fig. 3.4(c) shows the detected cutting line on a skin region. After locating the cutting line, all skin pixels below the cutting line are removed. Fig. 3.4(d) shows the result of shoulder removing. Now, we will decide if the remaining skin is

a profile face. There are two kinds of profiles, one is right profile; the other is left profile. Since these two kinds of profiles have the same characteristics, in the following, we will only describe how to detect a right profile, as to left profiles, a similar way can be applied.



(a) The Top-Down line for a projection histogram. (b) The neck point and the cutting line located.



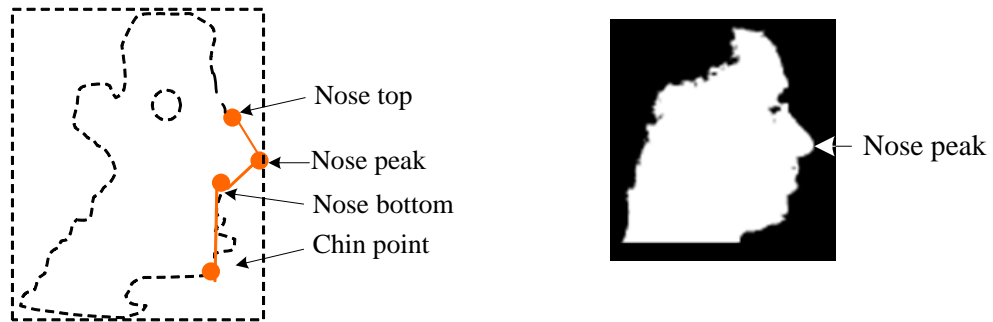
(c) The located cutting line on Fig. 3.3(c). (d) The result of removing shoulder from (c).

Fig. 3.4. An example for removing shoulder from a skin region.

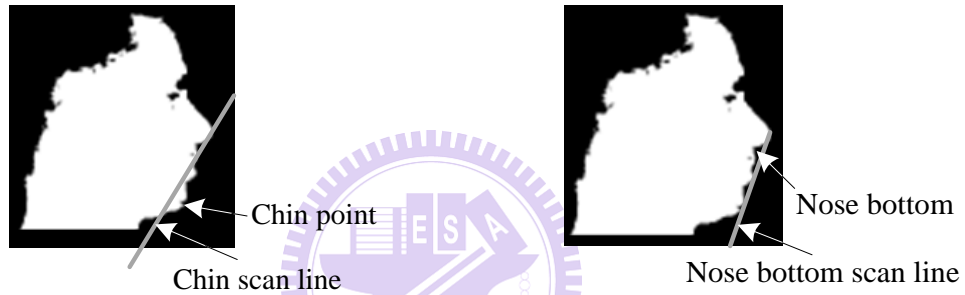
3.2.1.2 Profile Feature Extraction

There are some feature points (see Fig. 3.5(a)) on a profile. These are nose peak, nose bottom, nose top and chin point. Based on the geometric relations among these

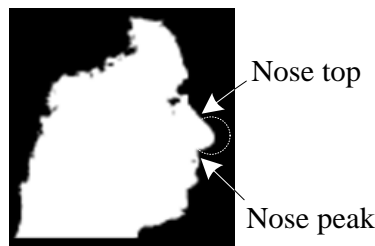
points, we can judge if a skin region is a profile. Here, we will provide a method to extract these points.



(a) Some feature points on a right profile. (b) The extracted nose peak.



(c) The extracted chin scan line and chin point. (d) The extracted nose bottom scan line and nose bottom.



(e) The extracted nose top.

Fig. 3.5. An example for illustrating the profile features.

First, the most right point of a skin region is considered as the nose peak (see Fig. 3.5(b)). Next, a vertical line V passing through the nose peak is taken. Line V is then

rotated clockwise w.r.t nose peak until the length of the intersection segment between the rotated V and the skin region is larger than a predefined threshold, the rotated V is then defined as the chin scan line. For each contour pixel of the skin region below the nose peak and at the right side of the chin scan line, the distance from the pixel to the chin scan line is evaluated. And the point with the maximum distance is considered as the chin point (see Fig. 3.5(c)).

Based on the extracted chin point, the line from the nose peak to the chin point is defined as nose bottom scan line, which will be used to locate the nose bottom. Following the contour points from the nose peak to the chin point, the contour point with the maximum distance to the nose bottom scan line is considered as the nose bottom (see Fig. 3.5(d)). Finally, in order to locate the nose top, we draw a circle using the nose peak as its center and the distance from the nose bottom to the center as its radius, the top point of the skin region on the circle is considered as the nose top (see Fig. 3.5(e)). Based on these extracted points, we will provide a procedure to check if a skin region is a profile.

Four rules are provided to verify the extracted profile features. At first, we define some profile variables. Let H be the height of the skin region, T be the vertical distance from the nose top to the chin point, C be the vertical distance from the nose peak to the chin point, D be the distance from the nose peak to the nose bottom, and K

be the vertical distance from the nose peak to the top of the skin region. Define A to be the angle formed by the three points: nose top, nose peak and nose bottom. In order to determine the front head part of a profile face, we define a vertical line $V2$ passing through the nose peak and then rotate counter clockwise w.r.t nose peak until the length of the intersection segment between rotated $V2$ and the skin region is larger than a predefined threshold. The rotated $V2$ is then defined as the front-head line and B to be the angle between the front-head line and the vertical line passing through the nose peak. Fig. 3.6(a) shows an example for the above defined variables. Based on

these variables, four rules for profile checking are given as follows:

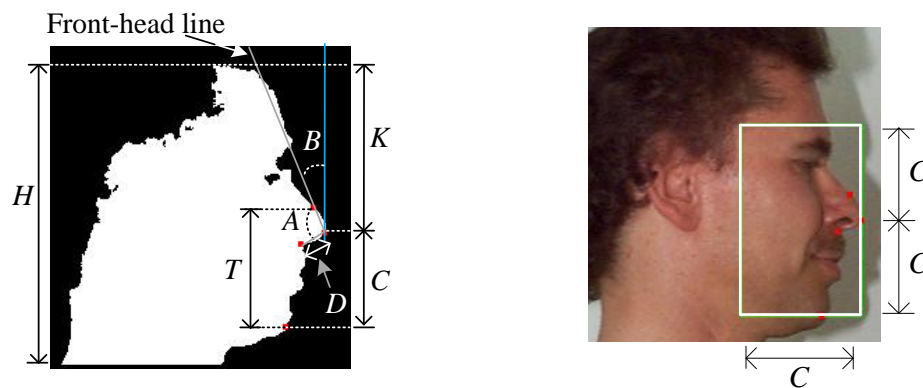
Rule 1: $T > H/4$. (3-1)

Rule 2: $D < 0.5 C$. (3-2)

Rule 3: $0.85 K < C < 1.2 K$. (3-3)

Rule 4: $45^\circ < A < 150^\circ$ and $15^\circ < B \leq 45^\circ$. (3-4)

If all rules are satisfied, the skin region is considered as a profile face.



(a) The defined variables for a skin region. (b) The defined profile face rectangle.

Fig. 3.6. An example to show the defined profile variables and the face rectangle.

If the skin region is not considered as a profile, the skin region is rotated according to the degree sequence $\{+5, -5, +10, -10, +15, -15\}$, and the profile detection procedure is applied again. Based on the nose peak and the distance C , a rectangle named as profile face rectangle is formed, we use the rectangle to identify the profile face location. Fig. 3.6(b) shows an example of the detected face rectangle for a profile face.

3.2.2 Non-Profile Face Detection

For a skin region not determined as a profile face, a non-profile face detection procedure is then applied. Jing and Chen [22] have proposed a method to locate the horizontal eye line and some eye-like rectangles on a skin image. Each eye-like rectangle has the same height, h . Based on the output of the Jing-Chen method, we will provide a method to determine if a skin region is a non-profile face. Fig. 3.7 gives an example, Fig. 3.7 (a) is a face image and Fig. 3.7(b) shows the located horizontal eye line (marked by green color) and some eye-like rectangles using Jing-Chen's method on Fig. 3.7(a). Based on the located horizontal eye line, we can define a checking bound, which is formed by a pair of lines with distance $1.5h$ from the horizontal eye line, to remove those false eye-like rectangles. In Fig. 3.7(b), we can see that only two eye-like rectangles are in the defined checking bound, and the

horizontal eye line passes through these two eye-like rectangles. We classify this layout as simple face model. For a simple face model, the only two eye-like rectangles within the checking bound are considered as the true eyes. Let de be the distance between the two eyes. A face rectangle is then defined as shown in Fig. 3.7(c). However, if a man wears glasses or a face is taken under a bias lighting source, then there will exist more than two eye-like rectangles in the checking bound (see Fig. 3.8). We classify these skin regions as complex face model. In the following, we will introduce a vertical symmetric line locator to solve the complex face model.

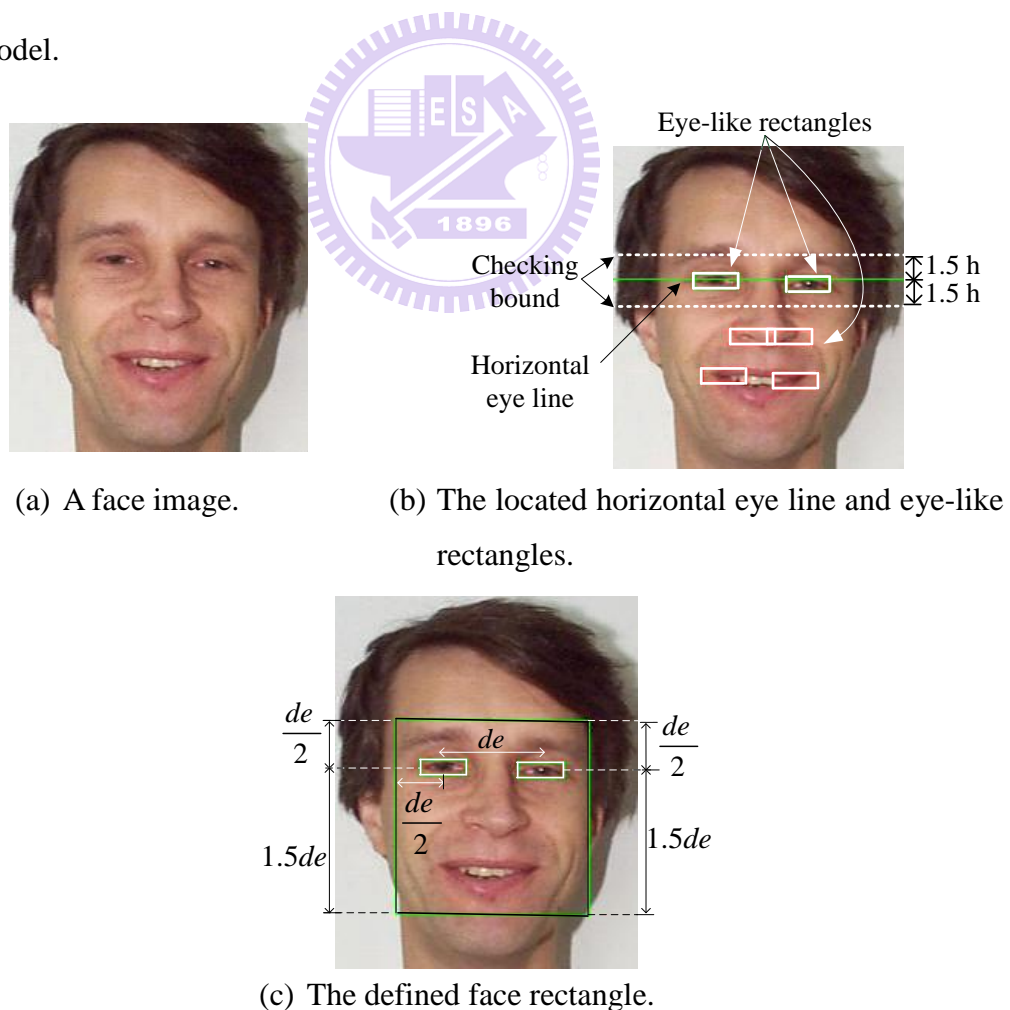


Fig. 3.7. An example for simple face model.



(a) A man with glasses.

(b) A man under bias lighting source.

Fig. 3.8. Two examples for the complexity face model.

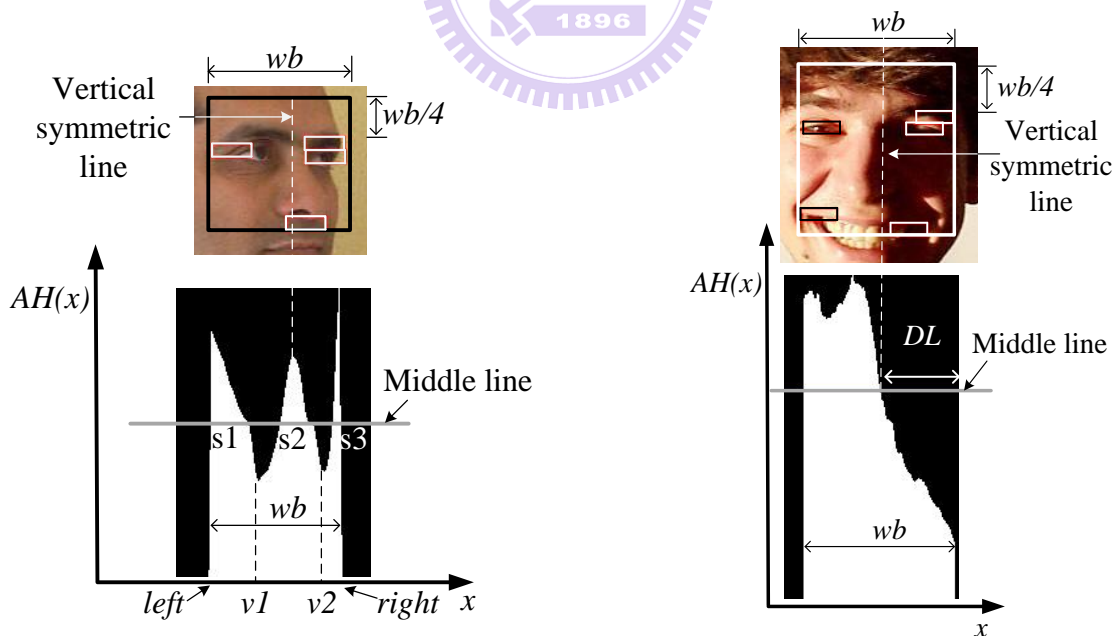
3.2.2.1 Vertical Symmetric Line Location

In general, face is a nearly symmetric pattern, a symmetric vertical line exists. There are some methods [23-25] based on this property to extract the symmetric vertical line. However, they may fail when a face is taken under biased lighting environment or is not frontal. A human face is a 3D object, different lighting source and poses will make the symmetric line location difficult. In order to overcome these factors, we will first classify the complex model into four patterns: half profile, bias lighting, near frontal and normal (see Fig. 3.9). From our observation, we found that the nose peak and forehead area always has higher lighting response than other facial regions. And the areas of eyebrows and eyes contain fewer skin color pixels and have lower gray values than other facial regions. Based on these facts, we define a vertical skin projection histogram $AH(x)$ as Eq. (3-5) to catch these characteristics. Note that the projection area, A , used in Eq.(3-5) is formed by extending BR $\frac{1}{4}wb$ height (see the black rectangle in Fig. 3.9(a)), where BR is the rectangle bounding

all eye-like rectangles, wb is the width of BR. Fig. 3.9 shows some projection results from some face images taken under different poses and lighting conditions.

$$AH(x) = \sum_{(x,y) \in A} (\text{gray}(x,y) + Cr(x,y)). \quad (3-5)$$

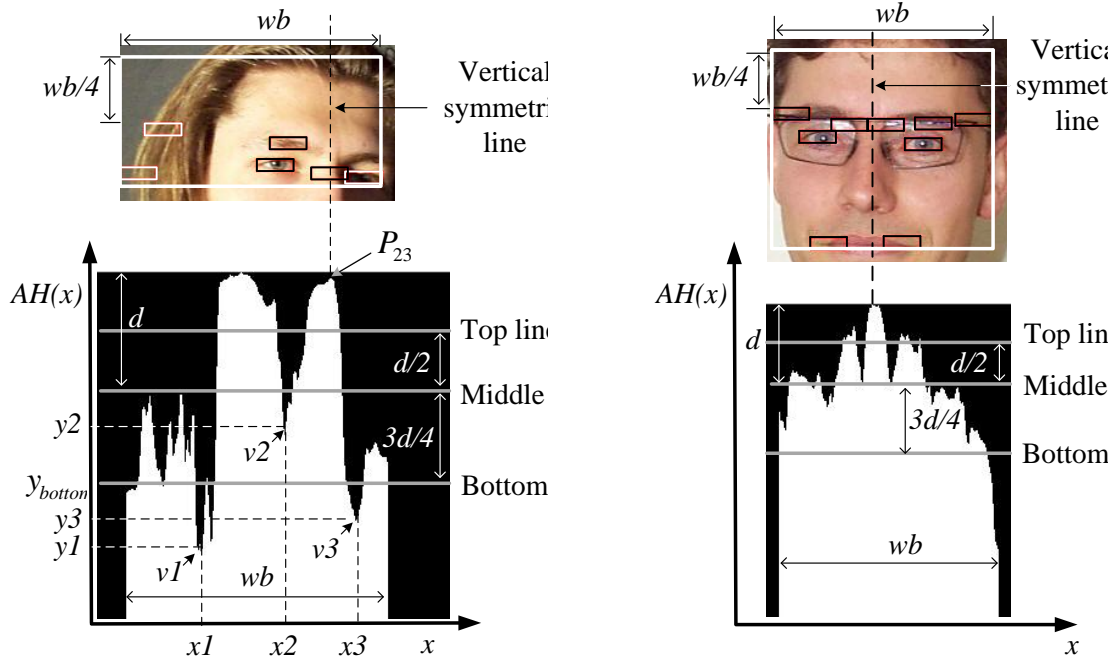
Based on the vertical projection histogram, we can do classification and then locate the vertical symmetric line. First, define $y_{mid} = \frac{1}{wb} \sum_{x=left}^{right} AH(x)$ to be the middle line of the histogram, where wb is the width of the projection area and *left* and *right* are the left and right bounds of the projection area. If there are only two valleys below the middle line (see Fig. 3.9(a)), and the intersection between the middle line and the histogram contains three segments, we say this is a half profile,



(a) A half profile pattern.

(b) A bias lighting pattern.

Fig. 3.9. The vertical histogram projection patterns and detected vertical symmetric lines (continued).



(c) A near frontal pattern with left-side pose.

(d) A normal pattern.

Fig. 3.9. The vertical histogram projection patterns and detected vertical symmetric lines.

and the vertical line passing the peak between two valleys is considered as the symmetric line (see Fig. 3.9(a)). If the skin region is not a half profile, then the bias

lighting testing is conducted. In order to judge the shadow ratio on a face, we collect

all dark segments on the middle line. Let DL be the longest dark segment (see Fig.

3.9(b)). If the length of DL is larger than $\left(\frac{wb}{2} - \sigma\right)$, we say this is a bias lighting

pattern, and the vertical line passing the end point of DL near the histogram middle

side is considered as the symmetric line (see Fig. 3.9(b)). In this dissertation, we set

the threshold σ as $\frac{wb}{8}$. If the skin region is neither a half profile nor a bias lighting

pattern, we apply the near frontal testing procedure. There are two kinds of near

frontal patterns, one is a face with left-side pose; the other with right-side pose. In the

following procedure, we will describe how to test a left-side near frontal pattern.

Using the similar way, we can test a right-side near frontal patterns. At first, define

two reference lines $y_{top} = y_{mid} + d/2$ as the top line and $y_{bottom} = y_{mid} - 3d/4$ as the

bottom line, where d is the distance from the global maximum peak of the histogram

to the middle line. If there exists three valleys $v1 = (x_1, y_1), v2 = (x_2, y_2)$ and

$v3 = (x_3, y_3)$ satisfying the left-side rule as follows:

The left-side rule:

$$\begin{aligned}
 & x_1 < x_2 < x_3, \\
 & t_1 |x_2 - x_3| < |x_2 - x_1|, \\
 & |x_1 - x_3| > \frac{wb}{2} \text{ and } y_1 < y_{bottom}.
 \end{aligned} \tag{3-6}$$

and the peak, $P_{2,3}$, between $v2$ and $v3$ is higher than y_{top} , then we consider the skin

region to be a left-side near frontal face pattern, and the vertical line passing $P_{2,3}$ as the

symmetric line. Fig. 3.9(c) shows an example of a man with left-side near frontal face

pattern. In this dissertation, t_1 is set as 1.2. Finally, for a skin region not belonging to

any above three patterns, the vertical line passing the maximum peak not near the

boundary of the projection area is considered as the symmetric face line (see Fig.

3.9(d)).

3.2.2.2 False Eye-Like Rectangle Removing

As we know, a pair of human eyes is symmetrically located on the left and right sides of the symmetric line of a face. That is, if we label one eye from a side of the symmetric line, then the other eye must locate at the other side and be symmetric to the labeled eye w.r.t the symmetric line. Using this property, those false eye-like rectangles can be eliminated. Based on the vertical symmetric line, a eye-like boundary is defined to remove those false eye-like rectangles. First, for each eye-like rectangle, evaluate the distance of the rectangle to the symmetric line. Second, two farrest eye-like rectangles at both sides of the symmetric line are taken, and let a be the distance of the near one. Finally, based on the horizontal eye line and the eye-like rectangle height (h), we can define the eye-like boundary as shown in Fig. 3.10. Any eye-like rectangle out of the eye-like boundary is considered as a false eye-like rectangle and is removed. Now, we will locate the true eyes from the remaining eye-like rectangles.

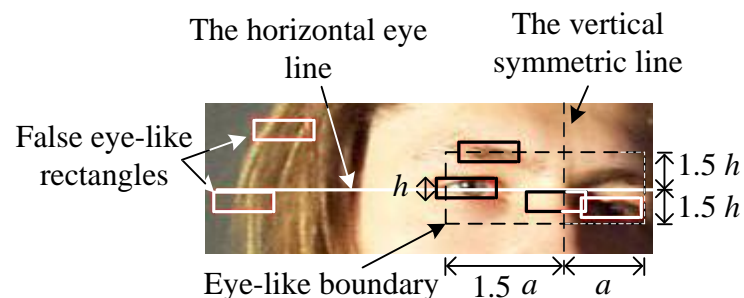


Fig. 3.10. An example to illustrate the eye-like boundary for a skin region.

3.2.3 True Eye Locating Procedure

Yow and Cipolla [26] use some geometric constraints to detect feature points including two eye points, however, when a face picture is taken under a biased lighting condition, these feature points cannot be detected correctly. Here, a two-step procedure is designed to locate two true eyes. In the first step, two rules are provided to identify the true eyes. With located eye centers (lx_e, ly_e) and (rx_e, ry_e) , a parallelogram with the top-left point $\left(lx_e - \frac{de}{2}, ly_e - \frac{de}{2}\right)$ and bottom-right point $\left(rx_e + \frac{de}{2}, ry_e + 1.5de\right)$ is then defined to be face area (see Figs. 3.11(b) and 3.11(d)), where de is the horizontal distance between the two eyes. In order to include the mouth area, if the distance de is shorter than a threshold $1.2w$, then the bottom-right point of the face area is refined as $\left(rx_e + \frac{de}{2}, ry_e + 2de\right)$.

Rule 1: If exact one eye-like rectangle locates at each side of the symmetric line, these two eye-like rectangles are considered as the true eyes and the face rectangle is located (see Figs. 3.11(a) and 3.11(b)).

Rule 2: If only one eye-like rectangle (ER) locates at one side of the symmetric line and more than one at the other side, consider the only rectangle ER as one true eye and identify the nearest rectangle of ER on the other side as the other true eye. Figs. 3.11(c) and 3.11(d) show an example of the located true eyes and the corresponding face location.

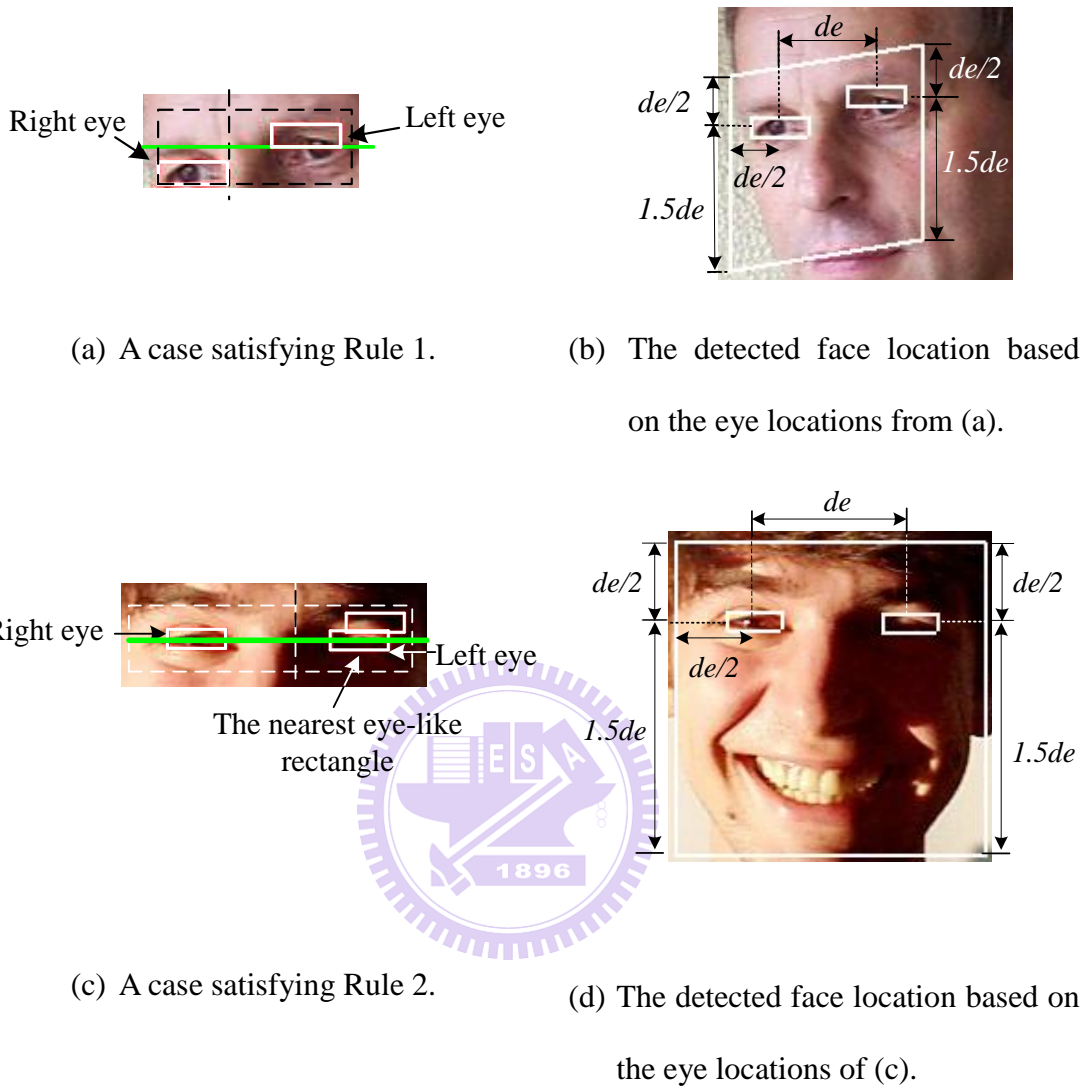


Fig. 3.11. Two examples for true eye and face location.

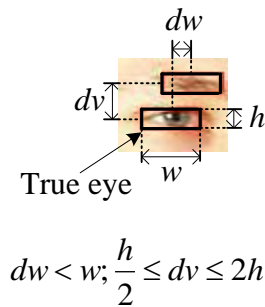
For those cases not satisfying *Rules* 1 or 2, the second step is conducted. Based on the relative locations among eyebrows, glasses and eyes, for the eye-like rectangle height (h) and width (w), two kinds of eye patterns (two-layered and three-layered) are defined to help locate true eyes.

Definition 1: A pair of two eye-like rectangles is called a two-layered pattern, if these two eye-like rectangles have horizontal distance dw with $dw < w$ and vertical distance dv with dv in $\left(\frac{h}{2}, 2h\right)$ (see Fig. 3.12(a)).

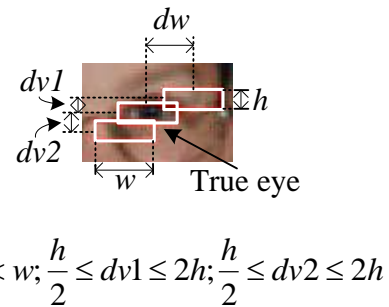
Definition 2: A group of three eye-like rectangles ($R1, R2, R3$) is called a three-layered pattern, if the horizontal distance between any two neighboring rectangles is less than w and the vertical distance in $\left(\frac{h}{2}, 2h\right)$. Fig. 3.12(b) gives an example of the three-layered pattern.

In the second step, if we find a pair of two-layered patterns which locate at each side of the symmetric line, respectively, then both bottom rectangles are identified as the true eyes (see Figs. 3.13(a) and 3.13(b)). If there is only one two-layered pattern detected, then choose the bottom eye-like rectangle as the true eye and remove all eye-like rectangles which intersect with the vertical symmetric line. The nearest eye-like rectangle on the other side of the symmetric line is considered as the other eye. Figs. 3.13(c) and 3.13(e) show two examples for this case. If there is no two-layered eye pattern at any side, then begin considering three-layered patterns. For a three-layered eye pattern, the middle rectangle is identified as the true eye. If a three-layered pattern is detected at one side, then the other eye is identified to be the closest eye-like rectangle at the other side. Fig. 3.13(g) shows an example of this case. If there are no any of the two kinds of patterns, consider the nearest rectangles to the

symmetric and horizontal lines as the true eyes (see Fig. 3.13(i)).

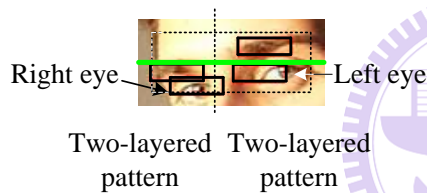


(a) The two-layered pattern.

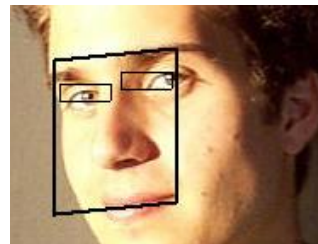


(b) The three-layered pattern.

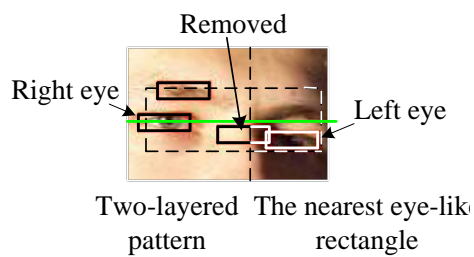
Fig. 3.12. Two examples for the two kinds of eye patterns.



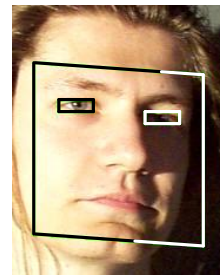
(a) Two-layered patterns at each side of the symmetric line.



(b) The detected face location based on the true eye locations on (a).

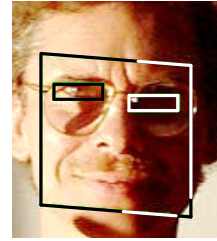
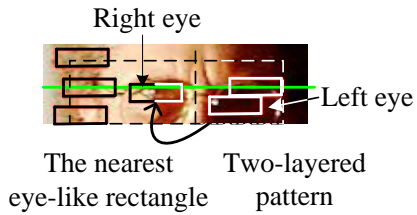


(c) One two-layered pattern detected.



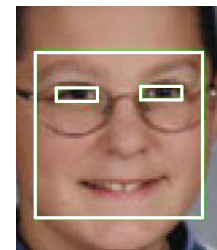
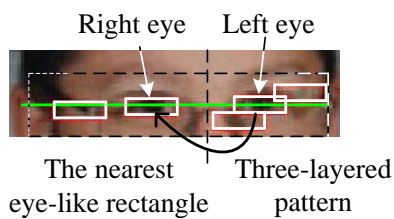
(d) The detected face location based on the eye locations on (c).

Fig. 3.13. Some examples for true eye and face rectangle location (*continued*).



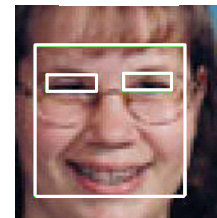
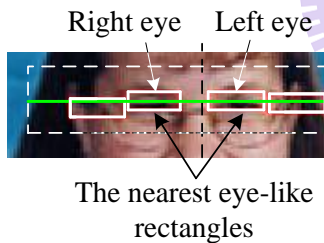
(e) Another example for only one two-layered pattern detected at one side.

(f) The detected face location based on the eye locations on (e).



(g) One three-layered pattern detected.

(h) The detected face location based on the eye locations on (g).



(i) An example of other patterns detected.

(j) The detected face location based on the eye locations on (i).

Fig. 3.13. Some examples for true eye and face rectangle location.

3.3 EXPERIMENTAL RESULTS

In order to show the effectiveness of the proposed method, we apply the method to HHI face database [19] of 206 images (see Fig. 3.14) and Champion face database [20] of 227 images (see Fig. 3.15). We also collect some images from our laboratory,

the Internet and MPEG7 video clips to evaluate the performance. These contain images of different racial persons under different kinds of lighting conditions (such as overhead, side and color lightings) and poses. The size of faces ranges from 252x229 to 79x79. Fig. 3.16 shows the successful results of applying our method to some faces with different skin colors. The successful results for faces with different poses and eye glasses are shown in Fig. 3.17. Even there is shadow on a face; the face can also be detected.

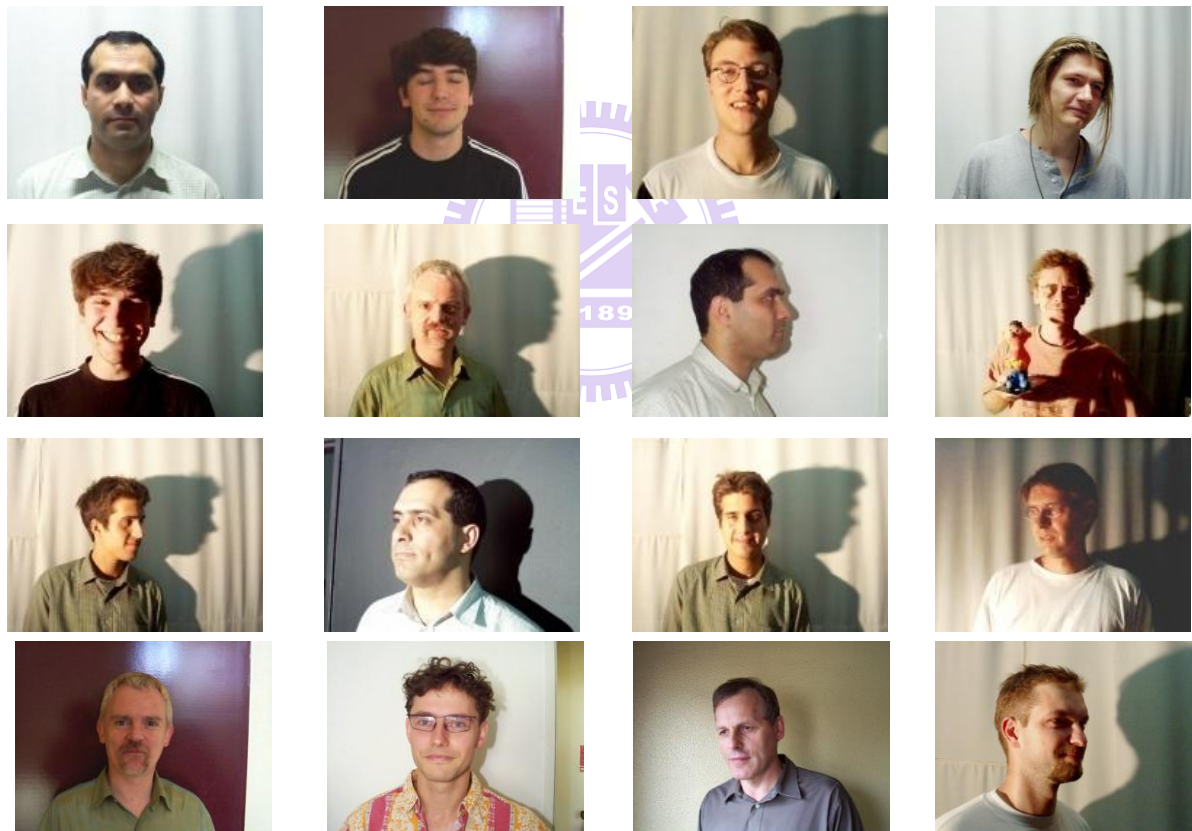


Fig. 3.14. A part of images from HHI face database.



Fig. 3.15. A part of images from Champion face database.



Fig. 3.16. The detection results for persons with different skin colors.

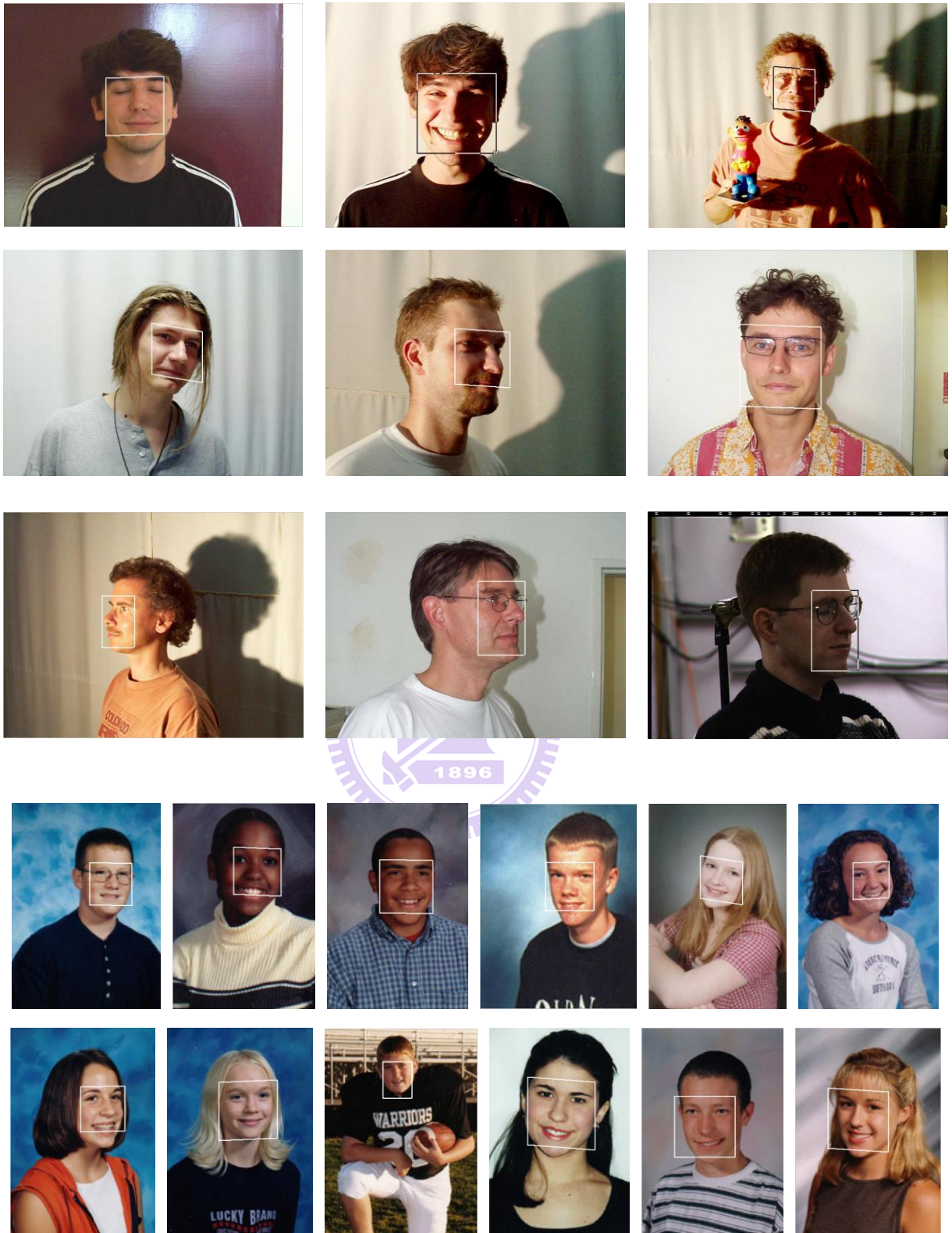


Fig. 3.17. The detection results for persons with different poses.

In the dissertation, if the located face rectangle bounds a face, consider it as a correct detection. If a face rectangle is found for a non-face region, we call it a false positive detection. For HHI database, the correct detection rate is 93.68%, and the correct detection rate on Champion database is 95.15%. Tables 3.1 and 3.2 show the detail detection results for both databases. Table 3.3 shows the detection result of the non-profile faces in HHI. In order to show the effectiveness of the proposed method, the original results of Hsu [18] using both databases and Chow [7] for HHI database are also given in Tables 3.1-3.3 to do comparisons.

Table 3.1 Detection results on HHI database (image size 640 x 480).

Head pose		Frontal	Near Frontal	Half Profile	Profile	Total
No. of images		66	54	75	11	206
Face Detection						
No. of false positive	Proposed method	5	1	6	0	12
	Hsu [18]	4	6	14	3	17
No. of miss	Proposed method	5	1	6	1	13
	Hue[18]	7	5	19	9	40
Correct detection rate (%)	Proposed method	92.4	98.1	92	90.9	93.68
	Hsu [18]	89.4	90.74	74.67	18.18	80.58
Average execution time per image (sec)	Proposed method	7.5 (on the mobile processor 1.4 G Hz CPU)				
	Hsu [18]	22.97 (on the 1.7 G Hz CPU)				

Table 3.2 Detection results on Champion database (image size ~ 150 x 220).

	Proposed method	Hsu [18]
No. of images	227	
No. of false positive	7	14
No. of miss	11	19
Correct detection rate (%)	95.15	91.63
Average execution time per image (sec)	4 (on the mobile processor 1.4 G Hz CPU)	5.78 (on the 1.7 G Hz CPU)

Table 3.3 Detection results on non-profile faces of HHI database.

	Proposed method	Chow [7]
No. of images (non-profile faces)	195 (206-11)	151 (selected)
No. of false positive	12	4
Correct detection rate (%)	93.8	92.7

The ROC curves for both databases are also given in Fig. 3.18. Fig. 3.19 shows the successful detection results for a set of images from our laboratory, the Internet and MPEG7 video clips. Fig. 3.20 shows multiple faces detection results.

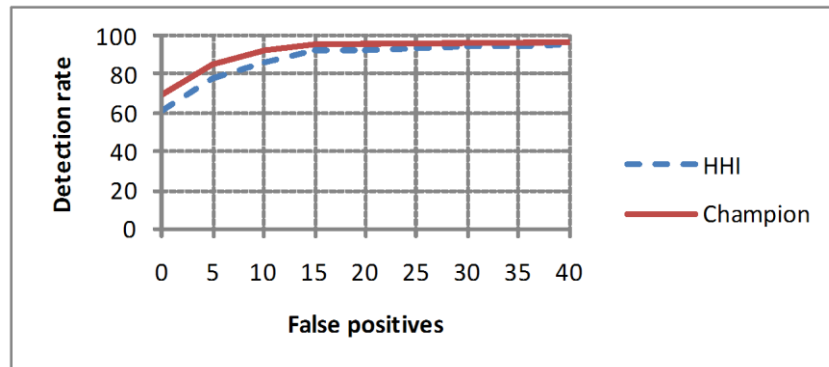


Fig. 3.18. The ROC curves for our face detector on HHI and Champion databases.

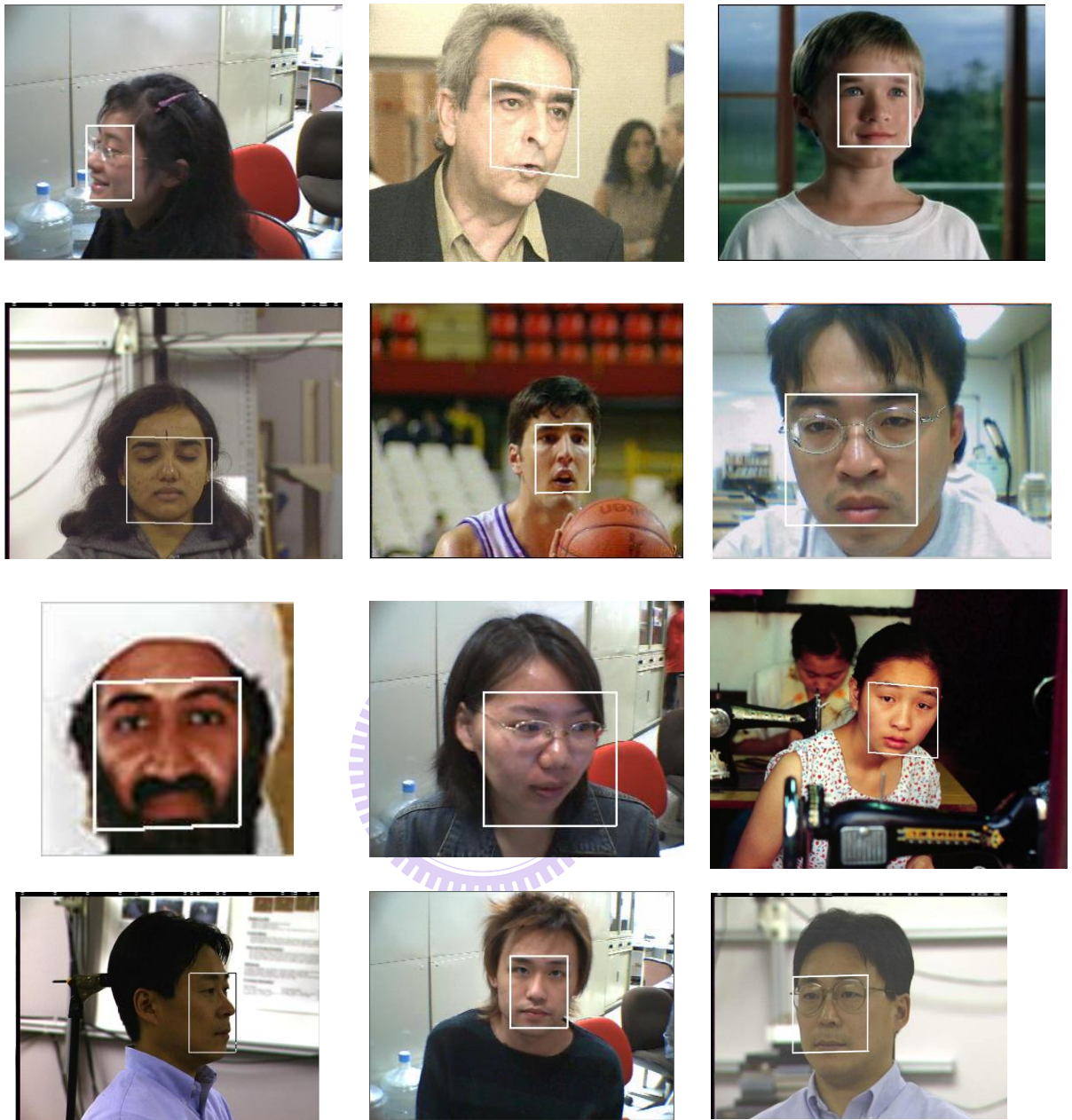


Fig. 3.19. The detection results of face images collected from our laboratory, the Internet, and MPEG7 video clips.

We had developed a face detector which can detect a face with different poses (left/right profile face and non-profile face) under various environments. However, some faces such as up-side or down-side face may fail to detect. We use a fixed ratio

threshold to detect the horizontal eye line for a given face candidate. This may cause the horizontal eye line detection failure for such faces. Another false detection case is that a profile face is taken from a skin-like background. Because we only use skin shape information to detect the profile face features. If we can not segment the profile face skin region, we cannot detect a profile face.

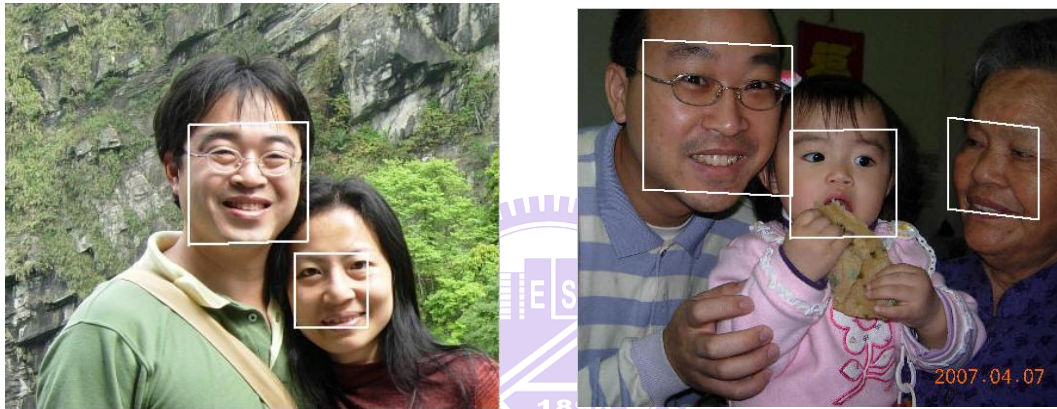


Fig. 3.20. The detection results for multiple faces.

CHAPTER 4

CONCLUSIONS AND FUTURE WORKS

In this dissertation, we have presented method to address the problems of face detection with poses under various environments. The main contribution of the work is developing a face detector which can detect a face with various poses (left/right profile face and non-profile face), different races, and robust over a wide range of lighting conditions (such as overhead, side and color lightings). Even if there is a closed eye on a face and there is a shadow on a face, the face can also be detected. The experimental results show that the proposed method is efficient and robust. The proposed face detector has a higher correct detection rate than those of Hsu *et al.* and Chow *et al.* In addition, we also present a novel efficiency algorithm to extract the head region of a person who wears a skin-like color clothes. On the basis of the output, we can easily separate the head and shoulder part of the body.

How to locate the eyes of a driver under extreme lighting condition is an important issue for a successful intelligent transportation system. In future, we will develop an efficiency eye blinking detection system based on the results of the proposed method to handle this problem.

REFERENCES

- [1] T. Hayami, K. Matsunaga, K. Shidoji, and Y. Matsuki, "Detecting drowsiness while driving by measuring eye movement – a pilot study," IEEE Proc. 5th Int'l Conf. Intelligent Transportation Systems, pp. 156-161, 2002.
- [2] E. Hjelmås and B. K. Low, "Face Detection: A Survey," Computer Vision and Image Understanding, vol. 83, no. 3, pp. 236-274, Sep. 2001.
- [3] M. H. Yang, D. J. Kriegman, and N. Ahuja, "Detecting faces in images: a survey," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 24, no. 1, pp. 34-58, Jan. 2002.
- [4] M. Turk and A. Pentland, "Eigenfaces for recognition," J. Cognitive Neuroscience, vol. 3, no. 1, pp. 71-86, 1991.
- [5] H. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 20, no. 1, pp. 23-38, Jan. 1998.
- [6] C. A. Waring and X. Liu, "Face detection using spectral histograms and SVMs," IEEE Trans. Systems, Man and Cybernetics — Part B: Cybernetics, vol. 35, No. 3, pp. 467-476, June 2005.
- [7] T. Y. Chow, K. M. Lam, and K. W. Wong, "Efficient color face detection algorithm under different lighting conditions," J. Electronic Imaging, vol. 15, issue 1, pp. 013015(1)-013015(10), May 2006.
- [8] F. Y. Shih and C. F. Chuang, "Automatic extraction of head and face boundaries and facial features," Information Sciences, vol. 158, pp. 117-130, Jan. 2004.
- [9] J. Wu and Z. Zhou, "Efficient face candidates selector for face detection," Pattern Recognition, vol. 36, issue 5, pp. 1175-1186, May 2003.
- [10] J. Miao, B. Yin, K. Wang, L. Shen, and X. Chen, "A hierarchical multiscale and

- multiangle system for human face detection in a complex background using gravity-center template,” *Pattern Recognition*, vol. 32, issue 7, pp. 1237-1248, July 1999.
- [11] J. Song, Z. Chi, and J. Liu, “A robust eye detection method using combined binary edge and intensity information,” *Pattern Recognition*, vol. 39, issue 6, pp. 1110-1125, June 2006.
- [12] V. Perlibakas, “Automatic detection of face features and exact face contour,” *Pattern Recognition Letters*, vol. 24, issue 16, pp. 2977-2985, Dec. 2003.
- [13] H. Wang, P. Li, and T. Zang, “Boosted Gaussian Classifier with Integral Histogram for Face Detection,” *Int. J. Pattern Recognition and Artificial Intelligence*, vol. 21, issue 7, pp. 1127-1139, 2007.
- [14] F. Y. Shih, S. Cheng, C. F. Chuang, and Patrick S. P. Wang, “Extracting Faces and Facial Features from color Images,” *Int. J. Pattern Recognition and Artificial Intelligence*, vol. 22, issue 3, pp. 515-534, 2008.
- [15] P. Kakumanu, S. Makrogiannis, and N. Bourbakis, “A survey of skin-color modeling and detection methods,” *Pattern Recognition*, vol. 40, issue 3, pp. 1106-1122, Mar. 2007.
- [16] S. Satoh, Y. Nakamura, and T. Kanade, “Name-it: naming and detecting faces in news videos,” *IEEE Multimedia*, vol. 6, no. 1, pp. 22-35, 1999.
- [17] D. Saxe and R. Foulds, “Toward robust skin identification in video images,” *Proc. Second Int’l Conf. Automatic Face and Gesture Recognition*, pp. 379-384, 1996.
- [18] R. L. Hsu, M. A. Mottaleb, and A. K. Jain, “Face detection in color images,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 696-706, May. 2002.
- [19] MPEG7 Content Set from Heinrich Hertz Institute, <http://www.darmstadt.gmd>.

- de/mobile/hm/projects/MPEG7/Documents/N2466.html. Oct. 1998.
- [20] The Champion Database, http://www.libfind.unl.edu/alumni/events/breakfastt_for_champions.htm. Mar. 2001.
- [21] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression database," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1615-1618, Dec. 2003.
- [22] M. Q. Jing and L. H. Chen, "A novel method for horizontal eye line detection under various environments," accepted by the *International Journal of Pattern Recognition and Artificial Intelligence*.
- [23] N. Nakao, W. Ohyama, T. Wakabayashi, and F. Kimura, "Automatic Detection of Facial Midline And Its Contributions To Facial Feature Extraction," *Electronic Letters on Computer Vision and Image Analysis*, vol. 6, no. 3, pp. 55-65, 2008.
- [24] X. Chen, P. J. Rynn, and K. W. Bowyer, "Fully automated facial symmetry axis detection in frontal color images," *Fourth IEEE Workshop on Automatic Identification Advanced Technologies*, pp. 106-111, 2005.
- [25] H. J. So, M. H. Kim, Y. S. Chung, and N. C. Kim, "Face Detection Using Sketch Operators and Vertical Symmetry," *Lecture Notes in Computer Science*, vol. 4027, pp.541-551, 2006.
- [26] K. C. Yow and R. Cipolla, "Feature-Based Human Face Detection," CUED /F-INFENG/TR 249, University of Cambridge, Department of Engineering, England, Aug. 1996.

PUBLICATION LIST

We summarize the publication status of the proposed methods and our research status in the following.

- (1) M. Q. Jing and L. H. Chen, “A novel method for horizontal eye line detection under various environments,” accepted by the International Journal of Pattern Recognition and Artificial Intelligence.
- (2) M. Q. Jing and L. H. Chen, “A novel face detection method under various environments,” accepted by *Opt. Eng.*
- (3) M. Q. Jing, C. H. Yu, H. L. Lee, and L. H. Chen, “Solving Japanese Puzzles With Logical Rules And Depth First Search Algorithm,” accepted by the International conference on Machine Learning and Cybernetics, 12-15 July, 2009.
- (4) M. Q. Jing, W. C. Yang, and L. H. Chen, “A New Steganography Method Via Various Animation Timing Effects In PowerPoint Files,” accepted by the International conference on Machine Learning and Cybernetics, 12-15 July, 2009.
- (5) M. Q. Jing, W. J. Ho, and L. H. Chen, “A Novel Method For Shoeprints Recognition And Classification,” accepted by the International conference on Machine Learning and Cybernetics, 12-15 July, 2009.
- (6) M. Q. Jing, C. C. Wang, and L. H. Chen, “A Real-Time Unusual Voice Detector Based On Nursing At Home,” accepted by the International conference on Machine Learning and Cybernetics, 12-15 July, 2009.
- (7) M. Q. Jing, C. R. Weng, C. H. Lee, and L. H. Chen, “An Algorithm for Eye Blink Detection,” submitted to *Electronics Letters*.

VITA

Min-Quan Jing was born in Ilan, Taiwan, Republic of China on December 15, 1973. He received the B.S. degree in Computer Science engineering from Chung Hua University, Hsinchu, Taiwan in 1997, and M.S degree in Computer & Information Science from National Chiao Tung University in 1999. He is now a Ph.D candidate of the Department of Computer Science at the National Chiao Tung University. His current research interests include image processing, pattern recognition and face detection.

