

ระบบรู้จำใบหน้าผู้ก่อการร้ายด้วยการตรวจสอบสีผิวบุคคลและไอเก็นเฟส

TERRORISTS FACE RECOGNITION SYSTEM WITH COLOR SKIN DETECTION  
AND EIGEN FACE



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมบดงกันประเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2557

KMITL-2014-EN-M-011-206

ระบบรู้จำใบหน้าผู้ก่อการร้ายด้วยการตรวจสอบสีผิวบุคคลและไอเกินเฟส

TERRORISTS FACE RECOGNITION SYSTEM WITH COLOR SKIN DETECTION  
AND EIGEN FACE



T139432



พิศณุ คุมีชัย  
PISANU KUMEECHAI

จพ.  
พ 7445  
2557

เลขหมู่.....139432  
เลขทะเบียน.....  
วันเดือนปี.....3 11 2557

.b.....  
.i.....

12722984

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต  
สาขาวิชาวิศวกรรมป้องกันประเทศ  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
พ.ศ. 2557

KMITL-2014-EN-M-011-206

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TERRORISTS FACE RECOGNITION SYSTEM WITH COLOR SKIN DETECTION  
AND EIGEN FACE



A THESIS SUBMITTED IN PARTIAL FULFILLMENT  
OF THE REQUIREMENTS FOR THE DEGREE OF  
MASTER OF ENGINEERING IN DEFENCE ENGINEERING  
FACULTY OF ENGINEERING  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG  
2014  
KMUTL-2014-EN-M-011-206

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2014



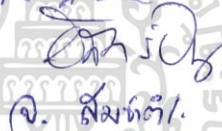

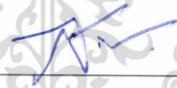
FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ ระบบรู้จำใบหน้าผู้ก่อการร้ายด้วยการตรวจสอบสีผิวบุคคลและไอเก็นเฟส  
Thesis Title Terrorists Face Recognition System with Color Skin Detection and Eigen Face  
นักศึกษา นาวาตรีพิศณุ คุ้มชัย  
รหัสประจำตัว 55614051  
ปริญญา วิศวกรรมศาสตรมหาบัณฑิต  
สาขาวิชา วิศวกรรมป้องกันประเทศ  
อาจารย์ที่ปรึกษาวิทยานิพนธ์ รศ.ดร.สุรพันธุ์ เอื้อไพบูลย์  
หมายเลขวิทยานิพนธ์ KMITL-2014-EN-M-011-206

คณะกรรมการสอบวิทยานิพนธ์		ลายมือชื่อ
รศ.ดร.จารุวัตร	เจริญสุข	
รศ.ประภาช	ไพรสวรรณา	
นาวาเอก ดร.อิสรา	จำไพกุล	
รศ.ดร.สมชาติ	จิรวีภากร	
รศ.ดร.สุรพันธุ์	เอื้อไพบูลย์	

วัน / เดือน / ปี ที่สอบ วันพุธที่ 9 ธันวาคม พ.ศ. 2557 เวลา 10.00-12.00 น.  
สถานที่สอบ ณ อาคาร 12 ชั้น ห้อง E12-304

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

คณะวิศวกรรมศาสตร์ รับรองแล้ว



(ผู้ช่วยศาสตราจารย์ ดร. คมสัน มาลีสี)

คณบดี คณะวิศวกรรมศาสตร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษา วันที่ 9 ธันวาคม พ.ศ. 2557  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	ระบบรู้จำหน้าใบหน้าผู้ก่อการร้ายด้วยการตรวจสอบสีผิวบุคคลและไอเก็นเฟส
นักศึกษา	นาวาตรีพิศณุ คุ้มชัย
รหัสประจำตัว	55614051
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมป้องกันประเทศ
พ.ศ.	2557
อาจารย์ที่ปรึกษาวิทยานิพนธ์	รศ. ดร. สุรพันธุ์ เอื้อไพบูลย์

### บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อนำเสนอวิธีการสำหรับการรู้จำหน้าใบหน้าผู้ก่อการร้ายโดยอิงจากหลักการ Eigenface ซึ่งเป็นวิธีการที่ใช้กันอย่างแพร่หลายในปัจจุบัน โดยงานวิจัยนี้ใช้สำหรับภาพใบหน้าผู้ก่อการร้ายระดับสีขนาดสองมิติ วิธีการคือในขั้นตอนแรกใช้วิธีการตรวจสอบค่าสีจากระบบสี HSV และวิธีการ Haar like feature ในการหาใบหน้าผู้ก่อการร้าย ต่อจากนั้นใช้วิธีการ Eigenface ในการหาคุณลักษณะเด่นบนใบหน้า เพื่อให้ได้ ไอเกนเวกเตอร์ หลังจากนั้นนำ ไอเกนเวกเตอร์ที่ได้จาก ภาพทดลองและภาพทดสอบทั้งหมด มาหาระยะทางแบบยุคลิด สุดท้ายนำผลการทดลองที่ได้จากวิธีการ ไอเกนเฟส มาเปรียบเทียบกับผลการทดลองที่ได้จากวิธี Linear discriminant analysis ซึ่งมีการเพิ่มสัญญาณรบกวน Salt and Pepper noise เข้าไปด้วยทั้งสองวิธีการ ผลการทดลองแสดงให้เห็นว่าผลลัพธ์ที่ได้สำหรับวิธีการรู้จำภาพใบหน้าผู้ก่อการร้ายให้ค่าความถูกต้องถึง 100% และเมื่อเพิ่มสัญญาณรบกวนให้ผลความถูกต้อง 98.80% สำหรับวิธีการ Linear discriminant analysis นั้น ให้ผลความถูกต้อง 95.80% และเมื่อเพิ่มสัญญาณรบกวนให้ผลความถูกต้อง 94.81% จากผลดังกล่าวแสดงให้เห็นว่าวิธีการที่ใช้ในงานวิจัยนี้ ให้ผลลัพธ์ที่มีค่าความถูกต้องสูงที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis	Terrorists Face Recognition System with Color Skin Detection and Eigen Face
Student	LCDr. Pisanu Kumeechai
Student ID.	55614051
Degree	Master of Engineering
Program	Defence Engineering
Year	2014
Thesis Advisor	Assoc. Prof. Dr. Surapan Airphaiboon

## ABSTRACT

This paper aims to introduce an alternative method for terrorists face recognition system based on Eigenface in which commonly used nowadays. The 2D RGB images were used in this research. First, HSV and Haar like feature were applied for face detecting occasionally. Second, Eigenface was used for feature extraction and retrieved Eigenvectors. Third, calculate the distances between eigenvectors retrieved from the test and training data using Euclidean method. Finally, the results from Eigenface method and Linear Discriminant analysis were then compared. The Salt and Pepper noise were added in all images used in this research. The results from the method used in this research based on Eigenface yielded 100 percent accuracy without noise and 98.80 percent accuracy when noise was added. The Linear Discriminant Analysis method gave 95.8 and 94.81 percent accuracy respectively. Hence, judging from the percent of accuracy, it can be concluded that the method used in this research provided the best results for the face recognition application.

## กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดี ข้าพเจ้าขอกราบขอบพระคุณ รศ.ดร.สุรพันธ์ เอื้อไพบูลย์ อาจารย์ที่ปรึกษาเป็นอย่างสูงที่ให้คำปรึกษา ให้คำแนะนำ ช่วยแก้ไขข้อบกพร่องต่างๆ ในการการทำวิทยานิพนธ์ ตลอดจนให้กำลังใจในการทำวิทยานิพนธ์ฉบับนี้จนสำเร็จได้ด้วยดี

ข้าพเจ้าขอกราบขอบพระคุณ รศ.ดร.จารุวัตร เจริญสุข, รศ.ประภาช ไพรสุวรรณ, น.อ.ดร.อิสรา รำไพกุล ร.น. และ รศ.ดร.สมชาติ จิรวิภากร ที่ได้กรุณาให้คำแนะนำ เสนอแนะข้อแก้ไขในเรื่องเอกสารในการทำวิทยานิพนธ์ รวมถึงเป็นกรรมการการสอบวิทยานิพนธ์ของข้าพเจ้า

ข้าพเจ้าขอกราบขอบพระคุณสถานศึกษา ร.ร.ศรีพฤตมา, รร.เตรียมทหาร, รร.นายเรือ และสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ได้ประสิทธิ์ประสาทความรู้แก่ข้าพเจ้า

ข้าพเจ้าขอขอบพระคุณผู้บังคับบัญชาทุกระดับชั้นในกองการฝึกกองเรือยุทธการ กองทัพเรือ ที่ได้เอื้ออำนวยข้าพเจ้าในทุกเรื่อง จนสามารถทำวิทยานิพนธ์ฉบับนี้ออกมาได้

ข้าพเจ้าขอขอบคุณ น.ต.ภานูวัฒน์ ธนะมูล ร.น. เพื่อน รร.นร.98 ที่ช่วยให้ความรู้ในการปฏิบัติงาน และปัญหาของเจ้าหน้าที่ในพื้นที่ก่อความไม่สงบ

ข้าพเจ้าขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ทุกคนในสาขาวิชาวิศวกรรมป้องกันประเทศรุ่นที่หนึ่ง รุ่นที่สอง และรุ่นที่สามที่เป็นกำลังใจให้ สุดท้ายขอกราบขอบพระคุณ บิดา มารดา ผู้ให้กำเนิดที่คอยให้การเลี้ยงดูและให้การศึกษาอย่างดี จนข้าพเจ้าสามารถทำงานออกมาได้อย่างที่ตั้งใจไว้

พิศณุ คุ้มชัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และทุกข้ออ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

	หน้า
บทคัดย่อ (ภาษาไทย).....	I
บทคัดย่อ (ภาษาอังกฤษ).....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญ.....	1
1.2 วัตถุประสงค์.....	1
1.3 ขอบเขตของวิทยานิพนธ์.....	1
1.4 ขั้นตอนการดำเนินงาน.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.6 รายละเอียดของวิทยานิพนธ์.....	2
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	3
2.1 งานวิจัยที่เกี่ยวข้อง.....	3
2.2 ระบบการมองเห็นภาพ (Vision System).....	4
2.2.1 การได้มาซึ่งภาพ (Image Acquisition).....	4
2.2.2 กระบวนการประมวลผลภาพ (Image Processing).....	4
2.2.3 ผลที่ได้หรือการแสดงผล (Output or Display).....	5
2.3 หลักการเบื้องต้นของการประมวลผลภาพ.....	5
2.3.1 พิกเซล (Pixel).....	5
2.3.2 ตำแหน่งของพิกเซล (Pixel Location).....	7
2.3.3 ระดับเกรย์ (Gray scale).....	7
2.3.4 พื้นฐานและระบบของสีโมเดล RGB (Red, Green, Blue).....	8
2.3.5 ความสว่าง (Brightness).....	8
2.4 เทคโนโลยีไปโอเมทริกซ์.....	10
2.4.1 ความหมายของไปโอเมทริกซ์.....	10
2.4.2 ลักษณะการทำงาน.....	10
2.4.3 ตัวอย่างการประยุกต์ใช้งานไปโอเมทริกซ์.....	10
2.4.4 ไปโอเมทริกซ์ประเภทต่างๆ.....	12
2.5 คำจำกัดความและขอบเขตของการรู้จำแบบ.....	12
2.5.1 การเทียบเทมเพลต (Template Matching).....	13
2.5.2 ลักษณะทางเรขาคณิต.....	13
2.5.3 การเทียบกราฟ (Graph Matching).....	14

## สารบัญ (ต่อ)

	หน้า
2.5.4 ไอเกนเฟซ (Eigenface) .....	14
2.5.5 โครงข่ายประสาทเทียม (Neural Network) .....	14
2.5.6 ภาพ 3 มิติ .....	14
2.6 การหาค่าประกอบสำคัญของใบหน้า .....	15
2.6.1 Appearance based method .....	15
2.6.2 Model based method .....	15
2.7 การเพิ่มสัญญาณรบกวน .....	15
2.8 การกำจัดสัญญาณรบกวน (Noise Reduction) .....	17
2.9 ระบบสี HSV (Hue, Saturation, Value) .....	18
2.10 การดึงวัตถุพื้นหน้าออกจากพื้นหลังโดยใช้ Threshold .....	19
2.11 การตรวจจับวัตถุโดยใช้ HSV .....	19
2.12 กระบวนการ Haar like feature .....	21
บทที่ 3 กระบวนการรู้จำ .....	24
3.1 ทฤษฎีและวิธีการไอเกนเฟซ .....	24
3.1.1 วิธีของไอเกนเฟซ .....	24
3.1.2 ทฤษฎีการวิเคราะห์องค์ประกอบหลัก .....	24
3.1.3 การแยกองค์ประกอบหลักชนิดสองมิติ (2DPCA) .....	27
3.1.4 หลักการสร้างภาพใบหน้าไอเกน (Eigenface) .....	28
3.2 Linear Discriminate Analysis (LDA) .....	31
3.3 ICA (Independent Components Analysis) .....	32
3.4 การแยกคุณลักษณะของใบหน้าทั้งใบหน้าและเฉพาะส่วนโดยใช้ แบบจำลองเชิงเรขาคณิต .....	33
บทที่ 4 การทดลองและผลการทดลอง .....	34
4.1 รายละเอียดของระบบที่ใช้ทดสอบ .....	35
4.1.1 ตัวอย่างภาพที่ใช้ในงานวิจัย .....	36
4.1.2 ขั้นตอนการเก็บและเตรียมภาพใบหน้า .....	37
4.2 ขั้นตอนการทำงาน .....	38
4.3 การทดลองระบุตัวบุคคลโดยใช้อัลกอริทึมไอเกนเฟซ .....	40
4.3.1 ทดสอบโดยใช้ใบหน้าผู้ทดสอบเป็นภาพต้นฉบับ .....	41
4.3.2 ทดสอบโดยใช้ใบหน้าผู้ทดสอบเป็นภาพใบหน้าที่มีสัญญาณรบกวน .....	43
4.3.3 ทดสอบระบบที่นำเสนอเปรียบเทียบกับวิธีอื่น .....	46
4.4 ขั้นตอนการทำงานในเครื่องคอมพิวเตอร์ .....	47
4.4.1 เปิดเพิ่มข้อมูลเลือก TrainDatabase .....	47
4.4.2 เปิดเพิ่มข้อมูลเลือก TestDatabase .....	48

เอกสารนี้เป็นเอกสารของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่สามารถนำออกจำหน่ายโดยไม่อนุญาตให้เผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
4.4.3 เลือกภาพใบหน้าที่จะทำการทดสอบ .....	48
4.4.4 ระบบทำการตรวจสอบ .....	49
4.4.5 ระยะเวลาในการประมวลผล .....	50
4.4.6 ค่าขนาดของข้อมูลที่แสดงในโปรแกรม .....	50
4.5 สรุปผลการทดลอง .....	51
บทที่ 5 สรุปผลการวิจัยและแนวทางในการพัฒนา .....	52
5.1 สรุปผลการวิจัย .....	52
5.1.1 สรุปผลประสิทธิภาพด้านซอฟต์แวร์ .....	52
5.2 ปัญหาที่เกิดขึ้น .....	52
5.2.1 ปัญหาโดยทั่วไป .....	52
5.2.2 การหาภาพที่นำมาใช้ .....	53
5.2.3 วิธีการแยกในส่วนของตาและปาก .....	53
5.2.4 อุปสรรคทางด้านซอฟต์แวร์ .....	53
5.2.5 ปัญหากระบวนการรู้จำเมื่อเพิ่มสัญญาณรบกวน .....	53
5.3 แนวทางในการพัฒนา .....	54
เอกสารอ้างอิง .....	55
ภาคผนวก .....	56
ภาคผนวก ก ผลงานวิจัยที่ได้รับการตีพิมพ์ .....	57
ภาคผนวก ข โปรแกรม .....	59
ประวัติผู้เขียน .....	93

# สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงค่า Gray Scale และ Gray Value Range .....	7
4.1 รายละเอียดของระบบที่ใช้ทดสอบ .....	35
4.2 แสดงผลการทดลองการระบุตัวบุคคลโดยใช้ภาพต้นฉบับ .....	41
4.3 แสดงผลการทดลองการระบุตัวบุคคลโดยใช้ภาพใบหน้าที่เพิ่มสัญญาณรบกวน Salt and pepper noise ของผู้ทดสอบ .....	44
4.4 แสดงผลการทดลองการระบุตัวบุคคลโดยใช้ภาพใบหน้าที่เพิ่มสัญญาณรบกวน Gaussian Noise ของผู้ทดสอบ .....	46
4.5 แสดงผลการทดลองการระบุตัวบุคคลโดยเปรียบเทียบกับวิธีอื่นๆ .....	47
4.6 แสดงผลการจับเวลาในการประมวลผลภาพในการทดสอบ .....	50
4.7 แสดงผลการวิเคราะห์ประสิทธิภาพของอัลกอริทึมไอเคนเฟซ .....	51



# สารบัญรูป

รูปที่	หน้า
2.1	ใบหน้าไอเกนเฟซจากงานวิจัยของ Turk and Pentland ..... 4
2.2	ลักษณะของภาพและตำแหน่งพิกเซล ..... 6
2.3	ดัชนีแสดงพิกเซลในเมทริกซ์ภาพๆ หนึ่ง ..... 6
2.4	ตัวอย่างของระดับเกรย์ (Gray Scales) ..... 8
2.5	แสดงโมเดลสี RGB ..... 9
2.6	ภาพแสดง Brightness Histogram ..... 10
2.7	มาตรฐานขนาดต่างๆ ที่ใช้ในการประมวลผลภาพบนสเปเชียลโดเมน ..... 16
2.8	ภาพที่แสดงบนสเปเชียลโดเมน ..... 16
2.9	ภาพที่แสดงบนโดเมนความถี่ ..... 16
2.10	ภาพตารางหมากรุก ..... 17
2.11	ภาพที่ประกอบด้วยสัญญาณรบกวนแบบพริกไทย ..... 17
2.12	มาตรฐานข้อมูลภาพที่พิจารณา ..... 17
2.13	ตัวกรองแบบเฉลี่ยข้อมูล ..... 17
2.14	ข้อมูลภาพที่พิจารณาด้วยมาตรฐานขนาด $3 \times 3$ และค่ามีฐานของข้อมูล ..... 18
2.15	โมเดลสี HSV ในรูปแบบโคน (Cone) ..... 18
2.16	รูปที่เก็บในแต่ละ channel ในระบบสี RGB และ HSV ..... 19
2.17	หลังจากดึงพื้นหน้าออกจากพื้นหลัง ..... 20
2.18	Bimodal image histogram ..... 20
2.19	แสดงตัวอย่างลักษณะเด่นของ Haar ที่ถูกใช้ใน OpenCV ..... 21
2.20	แสดงเทคนิคการรวมภาพ ..... 22
2.21	แสดงสายโซ่ของตัวกรอง ..... 22
2.22	แสดงตัวอย่างของตัวกรองสองตัวแรกใน Viola-Jones Cascade ..... 23
3.1	ภาพ 2 มิติขนาด $(N \times N)$ เมื่อทำเป็นรูปภาพแบบ 1 มิติ $N^2 \times 1$ (1-D Vector) ..... 25
3.2	การทำงานของ PCA ..... 25
3.3	แสดงหลักการ 2DPCA ..... 28
3.4	ใบหน้าก่อนหาค่า Eigenface และภาพใบหน้าไอเกน ..... 31
3.5	แสดงตำแหน่งพิกัดต่างๆ ที่ได้จากการคำนวณ ..... 33
4.1	กระบวนการรู้จำใบหน้าผู้ก่อการร้าย ..... 35
4.2	รูปตัวอย่างผู้ก่อการร้าย ..... 36
4.3	ใบหน้าผู้ก่อการร้ายรูปที่ 1 ..... 38
4.4	ใบหน้าผู้ก่อการร้ายรูปที่ 2 ..... 38
4.5	ใบหน้าผู้ก่อการร้ายรูปที่ 3 ..... 39
4.6	ใบหน้าผู้ก่อการร้ายรูปที่ 4 ..... 39
4.7	ใบหน้าผู้ก่อการร้ายรูปที่ 5 ..... 39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป

รูปที่	หน้า
4.8 ใบหน้าผู้ก่อการร้ายรูปที่ 6 .....	40
4.9 ใบหน้าผู้ก่อการร้ายรูปที่ 7 .....	40
4.10 ตัวอย่างการทดสอบผู้ก่อการร้ายคนที่ 19 ของโปรแกรม .....	42
4.11 ตัวอย่างการทดสอบผู้ก่อการร้ายคนที่ 22 ของโปรแกรม .....	42
4.12 ตัวอย่างการทดสอบผู้ก่อการร้ายคนที่ 38 ของโปรแกรม .....	43
4.13 รูปผู้ก่อการร้ายที่ใส่สัญญาณรบกวน Salt and pepper noise .....	44
4.14 รูปผู้ก่อการร้ายที่ไม่สามารถระบุตัวตนได้ถูกต้องจากการเพิ่มสัญญาณรบกวน แบบ Salt and pepper noise .....	45
4.15 รูปผู้ก่อการร้ายที่ใส่สัญญาณรบกวน Gaussian Noise .....	45
4.16 รูปผู้ก่อการร้ายที่ไม่สามารถระบุตัวตนได้ถูกต้องจากการเพิ่มสัญญาณรบกวน แบบ Gaussian Noise .....	46
4.17 การทำงานของระบบรูปที่ 1 .....	47
4.18 การทำงานของระบบรูปที่ 2 .....	48
4.19 การทำงานของระบบรูปที่ 3 .....	48
4.20 การทำงานของระบบรูปที่ 4 กรณีสามารถจดจำภาพได้ .....	49
4.21 การทำงานของระบบรูปที่ 5 กรณีไม่สามารถจดจำภาพได้ .....	49
4.22 การแสดงขนาดข้อมูลใน Workspace .....	51

# บทที่ 1

## บทนำ

ในบทนี้จะกล่าวถึงความเป็นมาและอธิบายถึงความสำคัญของการจัดทำวิทยานิพนธ์รวมถึงวัตถุประสงค์ ขอบเขต ขั้นตอนการดำเนินงานและประโยชน์ที่คาดว่าจะได้รับ

### 1.1 ความเป็นมาและความสำคัญ

ปัจจุบันเทคโนโลยีด้านอิมเมจโพรเซสซิ่ง เริ่มมีบทบาทมากขึ้นเนื่องจากสามารถนำไปประยุกต์ใช้งานด้านการประมวลผลสัญญาณบนสัญญาณ 2 มิติ เช่น ภาพนิ่ง (ภาพถ่าย) หรือภาพวีดิทัศน์ (วิดีโอ) และยังรวมถึงสัญญาณ 2 มิติอื่นๆ ที่ไม่ใช่ภาพด้วย

การรู้จำใบหน้าบุคคล เป็นหนึ่งในเทคนิคด้านการประมวลผลภาพดิจิทัลที่ยังไม่เป็นที่แพร่หลายในเหล่านักพัฒนาทางด้านวิศวกรรม สืบเนื่องจากเทคโนโลยีนี้ยังเป็นเทคโนโลยีใหม่ที่ยังมีการตีพิมพ์ไม่มากนัก ทำให้การพัฒนาระบบและการนำไปประยุกต์ใช้งานจริงยังไม่เป็นที่กว้างขวาง ด้วยประการนี้ จึงถือว่าเป็นโอกาสอันดีที่จะนำเทคโนโลยีนี้มาพัฒนาและประยุกต์ใช้งานในด้านการสืบค้นข้อมูลของทางราชการ

โดยที่วิทยานิพนธ์นี้ได้จัดทำการศึกษาการจำลองระบบตรวจสอบใบหน้าผู้ก่อการร้ายขึ้นมา โดยหลักการทำงานของระบบคือ เมื่อนำใบหน้าของผู้ต้องสงสัยมาเข้าตรวจสอบในระบบ จะสามารถตรวจสอบภาพใบหน้าของผู้ต้องสงสัยว่าเป็นผู้ก่อการร้ายหรือไม่ โดยนำไปทำการประมวลผลกับภาพใบหน้าผู้ก่อการร้ายที่อยู่ในฐานข้อมูล แล้วก็จะสามารถบ่งชี้ได้ว่าใบหน้าผู้ต้องสงสัยตรงกับใบหน้าผู้ก่อการร้ายที่บันทึกไว้ในฐานข้อมูลหรือไม่ และเมื่อได้ผลลัพธ์จากการประมวลผลแล้ว สามารถจะแสดงผลให้ผู้ใช้งานทราบได้ว่าใช่ใบหน้าของผู้ก่อการร้ายหรือไม่ หากใบหน้าผู้ต้องสงสัยที่นำมาตรวจสอบตรงกับผู้ก่อการร้ายแล้วก็จะสามารถดำเนินขั้นตอนทางกฎหมายได้อย่างรวดเร็วยิ่งขึ้น

### 1.2 วัตถุประสงค์

- 1.2.1 ศึกษาทฤษฎีและการใช้งานระบบการจดจำใบหน้า (face recognition)
- 1.2.2 ประยุกต์ใช้ระบบการจดจำใบหน้า เพื่อออกแบบระบบการตรวจสอบใบหน้าผู้ต้องสงสัยโดยใช้การจดจำใบหน้าบุคคล

### 1.3 ขอบเขตของวิทยานิพนธ์

- 1.3.1 ออกแบบและสร้างระบบการตรวจสอบใบหน้าผู้ก่อการร้ายโดยใช้การจดจำใบหน้า
- 1.3.2 สามารถจัดเก็บฐานข้อมูลใบหน้าผู้ก่อการร้ายได้ไม่น้อยกว่า 100 คน
- 1.3.3 สามารถแสดงผลและสภาวะการทำงานผ่านคอมพิวเตอร์

### 1.4 ขั้นตอนการดำเนินงาน

- 1.4.1 รวบรวมข้อมูลศึกษาทฤษฎีที่เกี่ยวข้อง จากแหล่งอ้างอิงต่างๆ
- 1.4.2 เสนอวิทยานิพนธ์
- 1.4.3 เริ่มการจัดทำวิทยานิพนธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 สามารถนำไปประยุกต์ใช้ประโยชน์ด้านการรักษาความปลอดภัยหรือระบบที่ต้องการใช้ประโยชน์จากการรู้จำใบหน้า และอื่นๆ

1.5.2 สามารถนำองค์ความรู้ในระบบไปพัฒนาต่อยอดเพื่อเพิ่มประสิทธิภาพและใช้งานตามจุดประสงค์ที่ต้องการได้

1.5.3 สามารถใช้เป็นแนวทางสำหรับการค้นคว้าวิจัยเพิ่มเติมเพื่อพัฒนาเทคโนโลยีใหม่ๆ ในอนาคต

## 1.6 รายละเอียดของวิทยานิพนธ์

เนื้อหาของวิทยานิพนธ์นี้ได้แยกออกเป็น 5 บท ได้แสดงวิธีการรู้จำภาพใบหน้าผู้ก่อการร้ายระดับสี่สองมิติ ซึ่งวิธีการและทฤษฎีต่างๆ ที่นำมาใช้ในวิทยานิพนธ์ รวมถึงผลการทดลองการสรุปผลและข้อเสนอแนะ มีรายละเอียดในบทต่างๆ ดังต่อไปนี้

### บทที่ 1 บทนำ

บทนี้กล่าวถึงความเป็นมาและอธิบายถึงความสำคัญของการจัดทำวิทยานิพนธ์รวมถึงวัตถุประสงค์ ขอบเขต ขั้นตอนการดำเนินงานและประโยชน์ที่คาดว่าจะได้รับ

### บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

บทนี้กล่าวถึงทฤษฎีและงานที่มีความเกี่ยวข้อง, ระบบการมองเห็นภาพ, หลักการเบื้องต้นของการประมวลผลภาพ, เทคโนโลยีไบโอเมทริกซ์, คำจำกัดความและขอบเขตของการรู้จำแบบ และการหาค่าประกอบสำคัญของใบหน้า

### บทที่ 3 กระบวนการรู้จำ

บทนี้กล่าวถึงกระบวนการรู้จำภาพบนใบหน้าด้วยวิธีการต่างๆที่ใช้ในงานวิจัยนี้ อันประกอบไปด้วย ทฤษฎีและวิธีการไอเคนเฟซ, Linear Discriminate Analysis (LDA), Independent Components Analysis (ICA) และการแยกคุณลักษณะของใบหน้าในส่วนองศา และปากด้วยแบบจำลองเชิงเลขาคณิตแล้วนำมาเข้าสู่กระบวนการรู้จำ

### บทที่ 4 การทดลองและผลการทดลอง

บทนี้กล่าวถึงแนวทางวิธีการทำการทดลองเพื่อทดสอบแนวคิดที่นำเสนอ

### บทที่ 5 สรุปผลการวิจัยและแนวทางในการพัฒนา

บทนี้กล่าวถึงการสรุปผลการวิจัยรวมถึงข้อเสนอแนะในการวิจัยและพัฒนาต่อไป

## บทที่ 2

# ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวความเป็นมาและความสำคัญของงานด้าน Image processing งานวิจัยต่างๆที่เกี่ยวข้อง รวมถึงทฤษฎีต่างๆ ที่นำมาประยุกต์ใช้กับโครงงานชิ้นนี้ ซึ่งมีทฤษฎีที่สำคัญ อันได้แก่ อัลกอริทึมไอเกนเฟซที่เป็นหัวใจสำคัญด้านซอฟต์แวร์

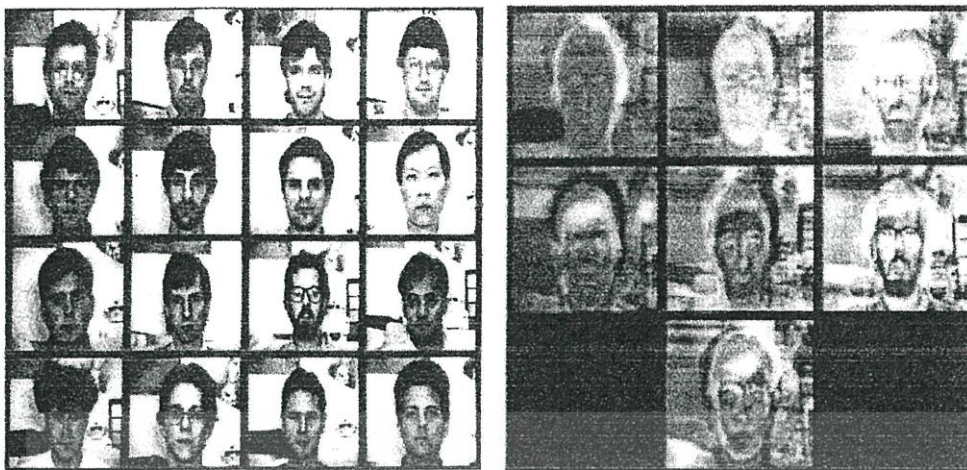
### 2.1 งานวิจัยที่เกี่ยวข้อง

ยุคแรกของการวิจัยและพัฒนาระบบรู้จำใบหน้า อัลกอริทึมการรู้จำใบหน้าจะทำได้โดยการค้นหาตำแหน่งส่วนที่เป็นคุณสมบัติเด่นบนใบหน้า อาทิ ตา จมูก และปาก แล้วทำการหาระยะทางระหว่างคุณสมบัติเหล่านี้ ตัวอย่างของงานวิจัยเหล่านี้ เช่น [Fischler, 1973], [Kanade, 1977] และ [Yuille, 1991] เป็นต้น

ในปัจจุบันนี้วิธีการที่ได้รับความนิยมคือวิธีการเปรียบเทียบค่าความสว่างของพิกเซล ในภาพใบหน้าอ้างอิงกับภาพใบหน้าที่ต้องการรู้จำ โดยการเปรียบเทียบจะทำแบบพิกเซลต่อพิกเซลในภาพทั้งสอง นอกจากนี้เพื่อทำให้ขนาดของข้อมูลที่ต้องประมวลผลมีขนาดเล็กลง ภาพใบหน้าจะถูกแปลงเป็นค่าสัมประสิทธิ์ ที่มีขนาดเล็กกว่าขนาดของภาพ (lower dimensional subspaces) งานวิจัยในกลุ่มเหล่านี้ ได้แก่ Eigen-face [Kirby, 1990], [Turk, 1991a] ซึ่งใช้หลักการของ Principal-Component Analysis (PCA), Fishers-faces [Moghaddam, 1997], Modular eigenspace [Moghaddam, 1997] สำหรับตัวอย่างของงานวิจัยที่พัฒนาต่อยอดจากแนวคิดข้างต้นมีดังต่อไปนี้

- 1) Independent Component Analysis (ICA) [Bartlett, 2002]
- 2) Eigenspace-based adaptive approach [Wechsler, 2000]
- 3) Multilinear (tensor) [Vasilescu, 2007]
- 4) Subspace learning [Cai, 2007]
- 5) Local distance functions [Frome, 2007]

งานวิจัยของ Turk and Pentland เรื่อง การค้นหาตำแหน่งภาพใบหน้าและ การรู้จำภาพใบหน้าโดยใช้ ภาพไอเกน ซึ่ง Turk and Pentland ได้ตั้งชื่อว่า “ภาพใบหน้าไอเกน” ภาพใบหน้าแต่ละภาพแทนด้วยน้ำหนัก ที่ได้จากการเรียนรู้จากภาพใบหน้าไอเกน เพื่อเก็บลงฐานข้อมูลภาพใบหน้า เมื่อต้องการทดสอบว่าภาพนั้นเป็นใครก็จะต้องนำภาพนั้นไปคำนวณให้เป็นค่าน้ำหนักและทำการค้นหาภาพใบหน้า ในฐานข้อมูลโดยใช้วิธีการวัดระยะทางแบบยูคลิเดียน ภาพใบหน้าในฐานข้อมูลที่ใกล้เคียงกับภาพใบหน้าที่นำมาทดสอบก็คือ ภาพนั้นมีระยะทางใกล้ กันมากที่สุด ฐานข้อมูลภาพใบหน้าประกอบไปด้วยภาพ ใบหน้า 2,500 ภาพ ภาพใบหน้าแต่ละคน มีลักษณะการเอียงของภาพ ใบหน้า 3 แบบมีขนาดของภาพใบหน้าที่ไม่เท่ากัน 3 ขนาด และอยู่ในสภาวะที่แสงเปลี่ยนแปลง 3 แบบ ผลการทดลองจากข้อมูลภาพใบหน้าที่มีการเปลี่ยนแปลงในรูปแบบต่างๆ เป็นดังนี้ ระบบจะให้ความถูกต้องสูงในกรณีที่ภาพมีการเปลี่ยนแปลงจากแสง แต่ให้ผลไม่ดีในกรณีของการเปลี่ยนขนาดภาพใบหน้า ในส่วนของการค้นหาภาพใบหน้า ได้ทำการค้นหาภาพที่ได้จากกล้องวีดีโอโดยให้ภาพใบหน้ามีการเคลื่อนไหว และใช้วิธีตัวกรองชั่วคราวสภาพโตค้นหาภาพใบหน้าที่มีการเคลื่อนไหว ในการกำหนดตำแหน่งของหัว ใช้กฎง่ายๆ และที่การกำหนดเองด้วยมือ เพื่อนำภาพที่ได้เข้าสู่ระบบรู้จำ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 ใบหน้าไอเคนเฟซจากงานวิจัยของ Turk and Pentland

## 2.2 ระบบการมองเห็นภาพ (Vision System)

กลไกระบบการมองเห็นภาพนั้น จะมีความหมายที่รวมไปถึงทุกสิ่งๆ ที่จำเป็นที่สามารถแปลงภาพนั้นๆ เป็นรหัสดิจิทัลเพื่อที่จะเอามาใช้กับระบบคอมพิวเตอร์ได้ การปรับปรุงข้อมูลและการนำเสนอภาพที่ได้มาหลังจากการปรับปรุงเปลี่ยนแปลงข้อมูลแล้ว ความยุ่งยากของระบบการมองเห็นภาพนี้จะขึ้นอยู่กับการใช้งาน ซึ่งสามารถแบ่งเป็น ขั้นตอนที่สำคัญ ได้ดังนี้

- 1) การได้มาซึ่งภาพ (Image Acquisition)
- 2) กระบวนการประมวลผลภาพ (Image Processing)
- 3) ผลที่ได้หรือการแสดงผล (Output or Display)

ในปัจจุบันนี้ได้มีการประยุกต์ใช้งานระบบการมองเห็นภาพอยู่มากมาย เช่น การใช้บาร์โค้ด การพิมพ์สิ่งพิมพ์ต่างๆ และการประยุกต์ใช้งานในโรงงาน เป็นต้น

### 2.2.1 การได้มาซึ่งภาพ (Image Acquisition)

การได้มาซึ่งภาพ หมายถึง การแปลงภาพในลักษณะทางกายภาพให้เป็นเขตของข้อมูลทางดิจิทัล ซึ่งเขตของข้อมูลนี้จะถูกส่งไปยังหน่วยประมวลผลต่อไป ฟังก์ชันของการได้มาซึ่งภาพนี้จะแบ่งออกเป็น 4 เฟส ได้แก่

- 1) การส่องสว่าง (Illumination)
- 2) รูปแบบของภาพ หรือ การทำให้ภาพชัดขึ้น (Image formation for -Focusing)
- 3) การตรวจจับภาพ หรือ การรับภาพ (Image detection or Sensing)
- 4) รูปแบบของผลสัญญาณที่ได้จากกล้อง (Formatting camera output signal)

การส่องสว่างเป็นตัวแปรสำคัญที่มีอิทธิพลต่อสัญญาณอินพุต (Input signal) ที่จะส่งต่อให้กับระบบการมองเห็นภาพ ดังนั้น เราจึงต้องออกแบบให้มีการส่องสว่างที่เหมาะสมกับการใช้งานที่แตกต่างกัน โดยที่ชนิดและวิธีการส่องสว่างของแหล่งกำเนิดแสง จะมีผลต่อกำลังงานของแสงที่ส่งออกมา ซึ่งจะมีผลต่อกระบวนการประมวลผลภาพและผลของสัญญาณที่ได้รับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2.2 กระบวนการประมวลผลภาพ (Image Processing)

การประมวลผลภาพ คือ การสร้างภาพใหม่โดยการแยกส่วนของข้อมูลที่เราส่งใจกับสิ่งรบกวน (Noise) ออกจากกันโดยการทำงานพื้นฐานของการประมวลผล คือ การกำจัดสิ่งรบกวนของภาพ (Noise Elimination) การปรับแต่งขอบภาพให้ดีขึ้น (Edge enhancement) การมองภาพ (Filtering) การปรับปรุงเปลี่ยนแปลงค่าระดับเกรย์ (Gray scale modification) โดยทั่วไปแล้วจะใช้ฮาร์ดแวร์ในการประมวลผลภาพ แต่ถ้ามีการประมวลผลที่ซับซ้อนขึ้นก็จะใช้ทั้งฮาร์ดแวร์และซอฟต์แวร์ ซึ่งความซับซ้อนของการประมวลผลจะขึ้นอยู่กับลักษณะของการใช้งาน และจะมีวิธีการประมวลผล 3 อย่าง คือ

1) ที่จุดเดียวกันในรูปภาพเดียวกัน (Point by point in one image) คือ การสร้างภาพใหม่โดยการเปลี่ยนค่าแบบจุดต่อจุด โดยจุดหนึ่ง คือ จุดในรูปภาพเดิม และอีกจุดหนึ่ง คือ จุดในรูปภาพใหม่ที่ได้รับการปรับปรุงเปลี่ยนแปลงจากรูปภาพเดิมแล้ว เช่น การแปลงภาพในระบบเลขฐานสอง จากจุดที่มีค่าเป็น 0 ถูกเปลี่ยนเป็น 1 และจากค่า 1 ถูกเปลี่ยนกลับเป็น 0

2) ที่จุดเดียวกันในรูปภาพที่แตกต่างกัน (Using corresponding point on different image) คือ การสร้างภาพใหม่โดยการจับคู่แต่ละจุดจากแหล่งภาพ 2 แหล่ง หรือมากกว่า จุดของรูปภาพที่แตกต่างกันนี้ จะนำมาลบบวกกันเพื่อหาค่าที่เปลี่ยนแปลงไป ข้อมูลของจุดในภาพใหม่จะเกิดจากการรวมกันของตัวเลขที่แตกต่างกันของภาพ

3) ที่บริเวณจุดนั้นในภาพนั้น (Using regional points in one image) คือ การสร้างภาพใหม่โดยการหาค่าเฉลี่ยรอบๆ จุดนั้นในภาพนั้น ค่าของจุดในภาพใหม่ คือ ค่าเฉลี่ยของจุดในภาพนั้นอันเดิม

## 2.2.3 ผลที่ได้หรือการแสดงผล (Output or Display)

รูปแบบของการแสดงผลนั้นขึ้นอยู่กับว่าจะนำข้อมูลที่ได้ไปใช้ต่อในลักษณะใด หรือแสดงผลอย่างไรจึงจะเหมาะสมที่สุด เช่น การแสดงผลโดยการพิมพ์ การแสดงผ่านหน้าจอ และการแสดงผลในรูปแบบของสัญญาณควบคุม เป็นต้น

## 2.3 หลักการเบื้องต้นของการประมวลผลภาพ (Image Processing Fundamentals)

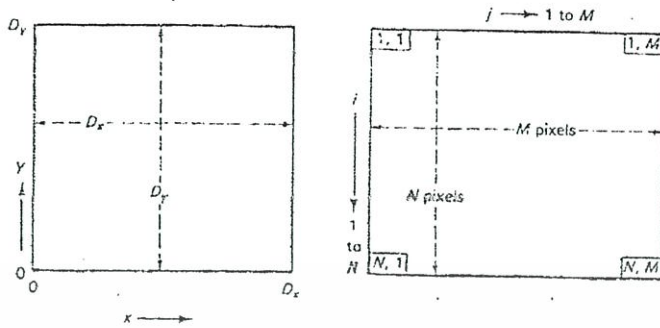
การประมวลผลภาพดิจิทัล เกี่ยวข้องกับการแปลงข้อมูลรูปที่เป็นสัญญาณแอนะล็อกให้อยู่ในรูปของสัญญาณดิจิทัล เพื่อใช้ในการประมวลผลผ่านทางคอมพิวเตอร์ได้ และ ยังนำมาใช้ในการลดปัญหาของภาพเช่น ลดสัญญาณรบกวนภายในภาพ เป็นต้น

ในการแปลงภาพให้เป็นสัญญาณดิจิทัลนั้น ระบบจะนำรูปที่รับเข้ามาไปคำนวณ โดยกระบวนการ Sampling และ Quantization และส่งข้อมูลออกมาในรูปแบบดิจิทัล คอมพิวเตอร์จะเก็บข้อมูลภาพลงหน่วยความจำ โดยการจองหน่วยความจำภายในเครื่องในรูปแบบของอาร์เรย์ โดยค่าในแต่ละช่องของ อาร์เรย์แสดงถึงคุณสมบัติต่างๆของรูปที่จุด พิกเซล นั้นๆ และตำแหน่งของช่องอาร์เรย์ก็เป็นตัวกำหนดตำแหน่งของจุด พิกเซล ภายในภาพด้วย

### 2.3.1 พิกเซล (Pixel)

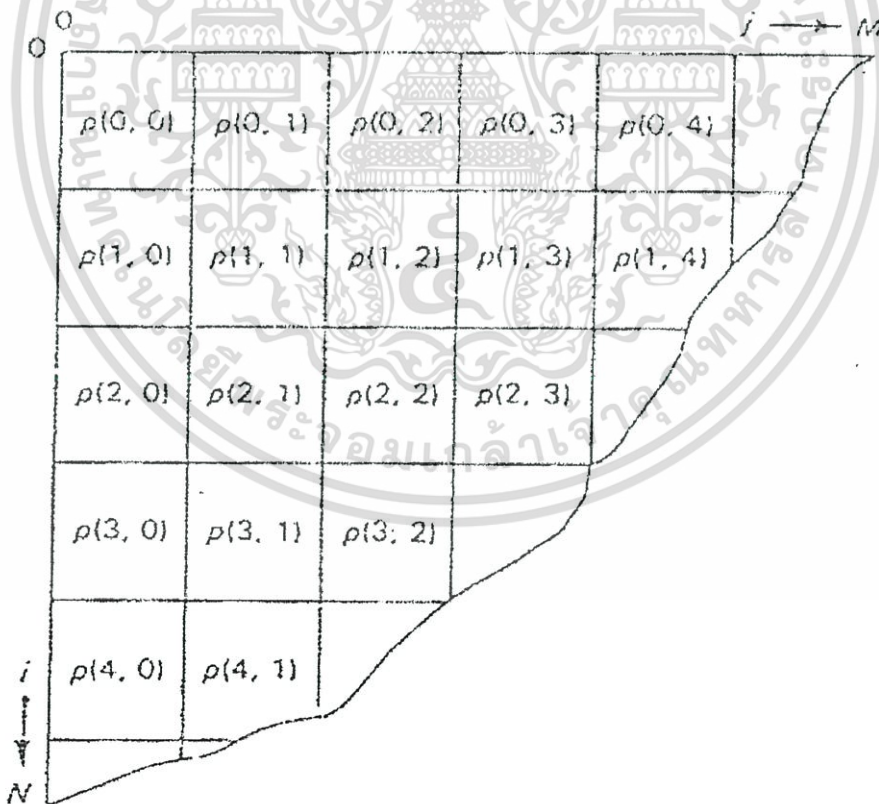
คือจุดสีเหลี่ยมเล็กๆ จำนวนมากที่เรียงต่อกันจนเกิดเป็นภาพ เมื่อทำการขยายภาพให้มีขนาดใหญ่มากขึ้น จะทำให้มองเห็นจำนวนพิกเซลเล็กๆ เหล่านั้น แต่ละพิกเซลจะระบุถึงตำแหน่งที่ไม่ว่างกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อยู่ของพิกเซลนั้นๆ บนภาพซึ่งอยู่ในรูปของแกน  $x, y$  และระดับความเข้มของสี โดย 1 พิกเซลจะแสดงได้เพียงสีเดียวเท่านั้น



รูปที่ 2.2 ลักษณะของภาพและตำแหน่งพิกเซล

ในภาพหนึ่งๆ เราสามารถอธิบายได้ด้วยเมทริกซ์ของจุดพิกเซลขนาด  $N \times M$  โดยใช้คู่ลำดับ  $p(i, j)$  แทนค่าของจุดแต่ละจุด โดย  $i$  และ  $j$  ต้องไม่เป็นจำนวนสเกลาร์ลบ และ  $p(i, j)$  นี้จะบ่งชี้ความเข้มของแสงที่จุดนั้นๆ



รูปที่ 2.3 ดัชนีแสดงพิกเซลในเมทริกซ์ภาพๆ หนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าที่กำกับแต่ละพิกเซลจะแสดงถึงค่าเฉลี่ยของความเข้มของแสงในภาพที่จุดพิกเซลนั้นแทนอยู่ โดยที่ค่าของพิกเซลนี้จะเขียนแทนด้วย  $P_{ij}$  ซึ่งมีค่าตั้งแต่ 0 ถึง 1

### 2.3.2 ตำแหน่งของพิกเซล (Pixel Location)

ในภาพหนึ่งๆ นั้น เราสามารถแทนอาเรย์ (Array)  $N \times M$  และค่าในแต่ละจุดพิกเซลจะหมายถึง ค่าเฉลี่ยของความเข้มของแสงที่ตกกระทบถึงภาพที่จุดพิกเซลนั้น ตัวอย่างเช่น ในรูปที่ 2.3 เป็นภาพขนาด  $10 \times 10$  นิ้ว ถ้าไม่มีแสงตกกระทบบริเวณด้านบนของภาพ แต่มีแสงตกกระทบมากที่บริเวณส่วนล่างของภาพ เราจะใช้ระบบเลขฐานสอง แทนค่าความเข้มของการส่องสว่าง คือ 0 แทนบริเวณที่ไม่ถูกแสงเลย และ 1 แทนบริเวณที่มีความเข้มของแสงมากที่สุด ในรูปที่ 2.3 นี้เป็นภาพที่ถูกเขียนแทนด้วยเมทริกซ์ขนาด  $5 \times 4$  แต่ละองค์ประกอบของภาพมีขนาด  $2.5 \times 2.0$  นิ้ว และค่าในแต่ละองค์ประกอบของภาพนี้จะขึ้นอยู่กับแสงที่ตกกระทบเฉลี่ยบนพื้นที่นั้น

บริเวณองค์ประกอบที่มีขนาด  $2.5 \times 2.0$  นิ้ว ตรงส่วนมุมบนซ้ายของภาพ จะถูกแทนด้วยตำแหน่ง (1,1) ซึ่งในเมทริกซ์  $5 \times 4$  มีค่าเท่ากับ 0 แสดงว่าไม่มีแสงมาตกกระทบเลยบริเวณองค์ประกอบที่มีขนาด  $2.5 \times 2.0$  นิ้ว ตรงส่วนมุมล่างขวาของภาพ จะถูกแทนด้วยตำแหน่ง (5,4) มีค่าเท่ากับ 1 แสดงว่ามีความเข้มของการส่องสว่างสูงสุด ถ้าใช้ระบบ 16 ระดับเกรย์ (16 Gray level system) แทนระบบเลขฐานสอง จะได้ว่าที่จุดพิกเซล (1,1) จะมีค่าเท่ากับ 0 และที่จุดพิกเซล (5,4) จะมีค่าเท่ากับ 15 ผู้ออกแบบจะต้องกำหนดค่าเทรชโฮลด์ (Threshold value) ของความเข้มของการส่องสว่างที่จะใช้เป็นเกณฑ์ในการเปลี่ยนระดับจาก 0 เป็น 1

### 2.3.3 ระดับเกรย์ (Gray scale)

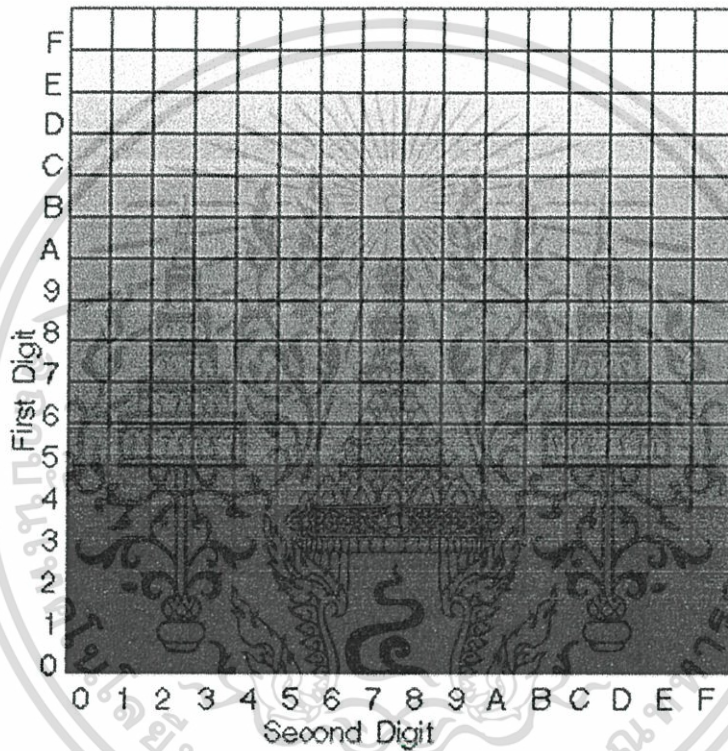
หากเราต้องการค่าของข้อมูลที่ละเอียดมากขึ้น จะต้องเพิ่มจำนวนบิตในการแสดงค่าของแต่ละพิกเซล เช่น ถ้าต้องการแบ่งความเข้มของการส่องสว่างให้มี 4 ระดับ ก็ต้องใช้เลขฐานสอง 2 บิต, 4 บิต สำหรับ 16 ระดับ และ 8 บิต สำหรับ 256 ระดับ ซึ่งจำนวนระดับที่ใช้ในระดับเกรย์นี้ มักจะเป็นเลขยกกำลังของ 2 ส่วนค่าต่ำสุด คือ 0 ถูกกำหนดให้เป็นสีดำ และ 1 หรือตัวเลขที่น้อยกว่าค่าสูงสุดของระดับเกรย์อยู่ 1 (เช่น 15 สำหรับระดับเกรย์ 16 ระดับ) ถูกกำหนดให้เป็นสีขาว ค่าที่กำหนดให้ในแต่ละพิกเซลมักเป็นจำนวนเต็ม

ตารางที่ 2.1 แสดงค่า Gray Scale และ Gray Value Range

Gray Scale	Gray Value Range
$2^1 = 2$ Value	0, 1
$2^3 = 8$ Value	0 to 7
$2^4 = 16$ Value	0 to 15
$2^8 = 256$ Value	0 to 255

ในยุคแรกๆ ของระบบการมองเห็นภาพ (Vision System) จะใช้ระบบเลขฐานสองแต่ในปัจจุบันเทคโนโลยีของไมโครโพรเซสเซอร์ (Microprocessors) เข้ามามีบทบาทมากขึ้น ซึ่งไมโครโพรเซสเซอร์จะมีหน่วยความจำอย่างน้อย 8 บิต ดังนั้น การแบ่งระดับเป็น 16, 64 หรือ 256 จึงเป็นเรื่องธรรมดา และในตอนนี้กำลังจะมีมากกว่า 256 ระดับ แต่ในการมองเห็นของมนุษย์สามารถ

แยกความแตกต่างได้เพียง 10 ถึง 15 ระดับเท่านั้น ดังนั้นการแบ่งโดยละเอียดเป็น 64 หรือ 256 ระดับ จะส่งผลทำให้มนุษย์ไม่สามารถแยกความแตกต่างได้ จึงนำไปประยุกต์ใช้กับงานประมวลผลภาพ หรือใช้ในขบวนการทางอุตสาหกรรมต่างๆ จึงอาจกล่าวได้ว่าจำนวนระดับเกรย์เป็นตัวจำกัดรายละเอียดของภาพ ยิ่งแบ่งระดับเกรย์เป็นหลายระดับก็จะยิ่งเพิ่มคุณภาพของภาพนั้นด้วย และการเพิ่มจำนวนพิกเซล เช่น จาก  $32 \times 32$  เป็น  $250 \times 250$  ก็จะเป็นการเพิ่มความละเอียดของภาพ (Resolution) และรายละเอียด (Detail) ของภาพ ซึ่งจะแตกต่างกับการขยายภาพ (Zoom) เพราะการขยายภาพเป็นการเพิ่มขยายของแต่ละพิกเซลให้ใหญ่ขึ้นไม่ได้เป็นการเพิ่มจำนวนพิกเซลในขณะที่การแบ่งระดับเกรย์เป็นการเพิ่มจำนวนของพิกเซล



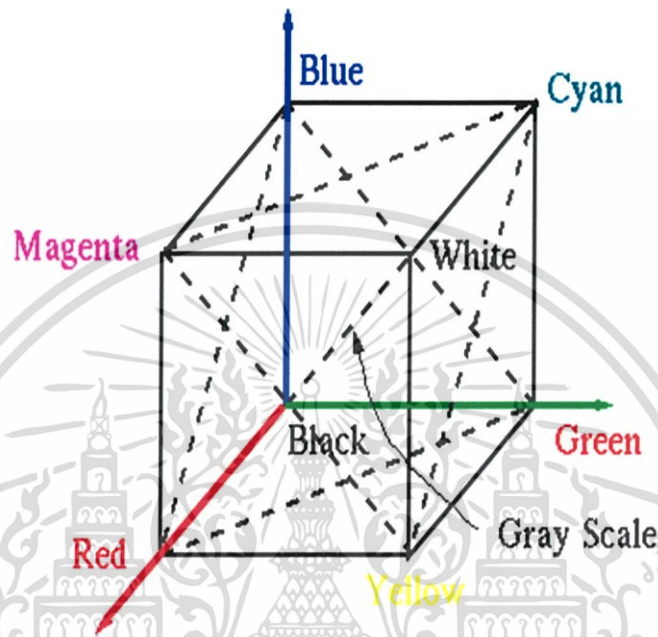
รูปที่ 2.4 ตัวอย่างของระดับเกรย์ (Gray Scales)

### 2.3.4 พื้นฐานและระบบของสีโมเดล RGB (Red, Green, Blue)

ในโมเดลนี้ สีแต่ละสีจะอยู่ในรูปของสี่พหุมุม (แดง, เขียว, น้ำเงิน) โมเดลนี้มีโครงสร้างอยู่ในระบบของพิกัดคาร์ทีเซียน (Cartesian Coordinate) มีลักษณะเป็นทรงลูกบาศก์ ดังรูปที่ 2.5 ค่าสีแดง, เขียว, น้ำเงิน จะอยู่ที่มุมทั้งสาม บนเส้นทแยงมุมตรงกันข้ามกัน และค่าสีคราม, มาเจน, เหลือง จะอยู่ที่มุมทั้งสามในลักษณะเดียวกัน ส่วนสีดำจะอยู่ที่จุดกำเนิด สีขาวอยู่ที่มุมที่มีระยะทางไกลที่สุดจากแหล่งกำเนิด ค่าของระดับสีเทา (Gray Scale) จะอยู่บนเส้นที่เชื่อมระหว่างสีดำกับสีขาว ค่าสี คือ จุดที่อยู่บนผิวหรือในลูกบาศก์ถูกกำหนดค่าโดยเวกเตอร์ที่ออกจากจุดกำเนิดเพื่อความสะดวกเราจะสมมติให้ค่าสีถูกนอร์มอลไลซ์ (Normalize) ทั้งสามสี โดยให้มีค่าอยู่ในช่วง 0 ถึง 1 ลูกบาศก์ที่แสดงในรูปที่ 2.5 นี้ จึงเป็นลูกบาศก์หนึ่งหน่วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

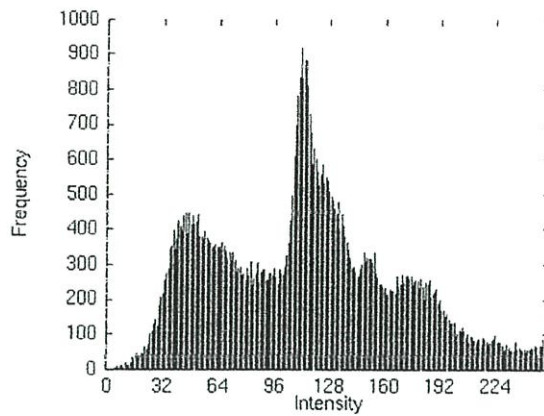
ภาพในโมเดล RGB ประกอบด้วยภาพสามระนาบที่เป็นอิสระจากกัน สำหรับแต่ละสีปฐมภูมิเมื่อป้อนเข้าไปในมอนิเตอร์ (Monitor) ที่เป็นแบบ RGB ภาพทั้งสามสีจะรวมตัวกันที่จอภาพกลายเป็นภาพสีผสม ดังนั้น การใช้โมเดล RGB ในการประมวลผลภาพนั้นจะสมเหตุสมผลเมื่อภาพถูกแยกออกโดยธรรมชาติให้อยู่ในเทอมของทั้งสามสี กล้องภาพสีส่วนใหญ่ที่ให้ภาพสีดิจิทัลจะอยู่ในรูปแบบของโมเดล RGB ดังนั้นโมเดลนี้จึงเป็นโมเดลที่สำคัญมากในการประมวลผลภาพ



รูปที่ 2.5 แสดงโมเดลสี RGB

### 2.3.5 ความสว่าง (Brightness)

ความสว่างเป็นคุณสมบัติของแสงที่มนุษย์เป็นผู้กำหนดให้ระหว่างความสว่างมากที่สุดกับความมืด ความสว่างเป็นสิ่งที่รับรู้ได้แต่ไม่สามารถสัมผัสได้ ความสว่างสามารถรับรู้ได้เมื่อมีแสงตกกระทบกับเรตินารูปแท่งและรูปกรวยภายในตาของเรา ซึ่งจะก่อให้เกิดการตอบสนองที่ไปเป็นเชิงเส้นและสลับซับซ้อน ความไวในการรับแสงของคนเราจะลดลงเมื่อแสงมีขนาดใหญ่มากขึ้น สำหรับความสว่างที่อยู่ในรูปภาพนั้น สามารถแสดงได้โดย Brightness Histogram ซึ่งเป็นกราฟการกระจายตัวของกลุ่มตัวเลขในกราฟนี้จะแสดงถึงระดับสีเทาของแต่ละพิกเซลที่อยู่ในรูปภาพ หรืออาจจะกล่าวได้ว่าเป็นการแสดงให้เห็นถึงจำนวนของพิกเซลในรูปภาพที่อยู่ในแต่ละระดับของสีเทามีอยู่เป็นจำนวนเท่าไร



รูปที่ 2.6 ภาพแสดง Brightness Histogram

## 2.4 เทคโนโลยีไบโอเมทริกซ์

### 2.4.1 ความหมายของไบโอเมทริกซ์

ไบโอเมทริกซ์ (Biometrics) เกิดจากการประสมกันของคำสองคำ คือ bio หมายถึง “ชีวิต” และ metrics ซึ่งหมายถึง “ที่สามารถวัดค่าหรือปริมาณได้” ได้มีการบันทึกไว้ว่าชาวอียิปต์โบราณมีการใช้ไบโอเมทริกซ์เป็นกลุ่มแรกเพื่อใช้ตรวจสอบ การใช้อุปกรณ์ไบโอเมทริกซ์สมัยใหม่มีเมื่อประมาณ 20 ปีมาแล้ว เป็นเครื่องวัดความยาวของนิ้วมือ เพื่อตรวจสอบเวลาทำงานแทนเครื่องตอกบัตร หลังจากนั้นจึงมีการพัฒนาอุปกรณ์สำหรับแยกแยะลักษณะมือเป็นจำนวนมากมาช่วยทำงานในระบบรักษาความปลอดภัย

### 2.4.2 ลักษณะการทำงาน

การทำงานจะเป็นการตรวจวัดคุณลักษณะทางกายภาพ (Physical Characteristics) และลักษณะทางพฤติกรรม (Behaviors) ที่เป็นลักษณะเฉพาะตัวของบุคคลนั้นๆ มาเปรียบเทียบกับคุณลักษณะที่มีการบันทึกไว้ในฐานข้อมูลก่อนหน้านี้ เพื่อใช้แยกแยะบุคคลนั้นจากบุคคลอื่นๆ ในงานด้านต่างๆ นอกจากนี้ยังสามารถใช้เพื่อตรวจสอบกลุ่มผู้ต้องสงสัย เพื่อหาตัวผู้กระทำความผิดได้อีกด้วย

### 2.4.3 ตัวอย่างการประยุกต์ใช้งานไบโอเมทริกซ์

1) งานควบคุมการเข้าออกสถานที่หรือการใช้ตรวจสอบเวลาทำงานการเข้าออกสถานที่ที่หวงห้ามในปัจจุบันมักใช้บัตรผ่าน หรือใช้รหัสผ่าน หรือแม้แต่การใช้ยามเฝ้า ซึ่งการป้องกันแบบนี้สามารถถูกหลอกได้ง่าย เช่น บัตรผ่าน หรือรหัสผ่าน อาจหาย ลืม หรือแม้แต่ให้คนอื่นยืมใช้ได้ ส่วนยามเฝ้าก็ขึ้นอยู่กับความเข้มงวดของยามแต่ละคน ความบกพร่องของระบบที่ใช้กันอยู่ในปัจจุบันจึงมีอยู่มาก การนำเอาไบโอเมทริกซ์มาช่วย เช่น การผ่านเข้าออกโดยใช้ลายนิ้วมือ, ใช้การตรวจสอบรูปหน้า, หรือแม้แต่การใช้การตรวจสอบลักษณะของเรตินาภายในดวงตา จึงเป็นทางออกที่ดีกว่าการใช้งานที่เป็นอยู่ในปัจจุบันระบบที่มีการใช้งานคล้ายกับการควบคุมการเข้าออกสถานที่ก็คือ การตรวจสอบเวลาทำงานของพนักงาน ซึ่งระบบที่ใช้กันอยู่ในปัจจุบันคือ การเซ็นชื่อ การใช้บัตรตอกลงเวลา และการใช้บัตรแถบแม่เหล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) ระบบใช้งานเครื่องคอมพิวเตอร์และภายในระบบเครือข่ายเครื่องคอมพิวเตอร์แบบ Notebook หลายรุ่น มีการนำเอาเทคโนโลยี Biometrics เช่น การใช้ลายนิ้วมือมาช่วยในการใช้งานเครื่องคอมพิวเตอร์ ซึ่งมีประโยชน์มากสำหรับบุคคลที่ต้องการความปลอดภัยในการรักษาข้อมูล เพราะถึงแม้ว่าเครื่องคอมพิวเตอร์จะถูกขโมยไปก็ไม่สามารถนำไปใช้งานได้ โดยการใช้ลายนิ้วมือมาช่วยมีอยู่หลักๆ สองประเภท คือ เครื่องคอมพิวเตอร์ Notebook ที่มีตัวตรวจจับลายนิ้วมืออยู่ในตัวเครื่องอยู่แล้ว และประเภทที่ใช้ PC Card ที่มีตัวตรวจจับลายนิ้วมืออยู่ใส่เข้าไปในช่อง PC Card ของเครื่องคอมพิวเตอร์ Notebook โดยที่ลายนิ้วมือจะเป็นการใช้ทดแทนการใส่รหัสผ่าน (Password)

นอกจากนี้การใช้งานคอมพิวเตอร์ระบบเครือข่าย จะต้องให้ผู้ใช้ใส่รหัสผ่านก่อนการใช้งานทุกครั้ง แต่เนื่องจากรหัสผ่านสามารถถูกคาดเดา หรือขโมย หรือถูกเฝ้ามองไปใช้ได้ง่าย ดังนั้น การใช้ Biometrics มาเป็นตัวจัดการเริ่มเข้ามาใช้งานของผู้ใช้ระบบเครือข่าย จึงเป็นสิ่งที่สามารถยืนยันได้อย่างแท้จริงว่า ผู้ที่ใช้ระบบเครือข่ายอยู่คือผู้ที่มีสิทธิในการใช้งานได้จริง เทคโนโลยีที่ใช้ได้กับด้านนี้ นอกจากการใช้ลายนิ้วมือแล้ว วิธีที่เหมาะสมต่อการใช้งานมากอย่างหนึ่งคือ การใช้ Keystroke dynamics หรือ การตรวจสอบตัวบุคคลโดยใช้ลักษณะของการพิมพ์ ทั้งนี้เพราะการเข้าไปใช้งานระบบเครือข่าย ผู้ใช้ต้องทำการพิมพ์ชื่อและรหัสที่อยู่แล้ว การตรวจสอบตัวบุคคลโดยใช้ลักษณะของการพิมพ์จึงเป็นสิ่งที่ไม่ต้องให้ผู้ใช้ทำอะไรเพิ่มเติม เช่นไม่ต้องตรวจสอบลายนิ้วมือ อีกทั้งวิธีการตรวจสอบแบบนี้ยังไม่ต้องการอุปกรณ์เพิ่มเติมใดๆ

3) การใช้งานของสถาบันการเงิน การตรวจสอบตัวบุคคล เป็นสิ่งที่ เป็นพื้นฐานที่สำคัญของการทำธุรกิจของสถาบันการเงิน การใช้การตรวจสอบลายเซ็น ลายนิ้วมือ บัตรประจำตัว หรือแม้แต่รหัสผ่าน ล้วนแต่เป็นสิ่งที่ง่ายต่อการปลอมแปลง และเป็นปัญหาที่สถาบันการเงินประสบเรื่อยมาตั้งแต่เริ่มมีธุรกิจการเงิน เมื่อมีการนำเอาเทคโนโลยีทางด้านไบโอเมทริกซ์มาเป็นองค์ประกอบเพิ่มเติมในการตรวจสอบตัวบุคคล จึงเป็นสิ่งที่ได้รับความสนใจเป็นอย่างมากจากสถาบันทางการเงิน ทั้งทางด้าน การช่วยในการเบิกถอนเงินทั้งที่ผ่านทางเคาน์เตอร์และทั้งที่ผ่านทางเครื่อง ATM นอกจากนี้การใช้เทคโนโลยีไบโอเมทริกซ์ในการตรวจสอบผู้ใช้บัตรเครดิต ก็จะเป็นการช่วยลดการปลอมแปลง หรือการลักลอบใช้บัตรเครดิตของผู้อื่น รวมถึงช่วยลดอัตราการปฏิเสธความรับผิดชอบของผู้ใช้ เพราะมีหลักฐานที่แน่นอนในการระบุตัวบุคคลที่เชื่อถือได้

4) การใช้งานด้านการระบุตัวอาชญากร หนึ่งในแนวทางการสืบอาชญากรที่ทำมาในอดีต คือ การตรวจสอบลายนิ้วมือ ซึ่งในปัจจุบันสามารถนำเทคโนโลยีไบโอเมทริกซ์ มาช่วยในการค้นหาลายนิ้วมือโดยอัตโนมัติหรือเรียกว่า ระบบ AFIS (Automated Fingerprint Identification System) ซึ่งระบบ AFIS นี้ นอกจากจะช่วยตำรวจค้นหาและเปรียบเทียบลายนิ้วมือที่พบในที่เกิดเหตุกับหลักฐานข้อมูลอาชญากรแล้ว ระบบยังสามารถเป็นแหล่งข้อมูลให้กับองค์กรเอกชน สำหรับค้นหาประวัติการทำผิดกฎหมายของผู้สมัครงานหรือบุคคลากรภายในองค์กรได้อีกด้วย นอกจากการใช้ลายนิ้วมือเป็นเครื่องมือในการตรวจจับอาชญากรแล้ว ยังมีการใช้เทคโนโลยีในศาลในหลายแห่งในประเทศอเมริกา โดยใบหน้าของผู้ต้องสงสัย จะถูกนำไปเปรียบเทียบกับฐานข้อมูลผู้เคยกระทำผิดที่มีอยู่ ถ้ามีลักษณะของบุคคลที่เหมือนกัน ระบบจะทำการส่งรูปของผู้ต้องสงสัยไปให้ผู้เชี่ยวชาญตรวจสอบอีกทีหนึ่ง โดยขั้นตอนทั้งหมดนี้ใช้เวลาดำเนินการเพียงไม่กี่นาทีเท่านั้น

คุณลักษณะทางกายภาพของคนเรานั้นส่วนใหญ่จะกินเวลาในการเปลี่ยนแปลงจากสภาพเดิมมากกว่า คุณลักษณะทางพฤติกรรมที่สามารถเปลี่ยนแปลงได้รวดเร็วกว่า ความสามารถในเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การยืนยันตนจึงมีความน่าเชื่อถือที่น้อยกว่า ทำให้การพิสูจน์บุคคลโดยวิธีพิจารณาลักษณะทางกายภาพนั้น มีความน่าเชื่อถือขณะใช้งานมากกว่าลักษณะทางพฤติกรรม ตัวอย่างของคุณลักษณะทางกายภาพที่นิยมนำมาใช้ ได้แก่ ลายนิ้วมือ ม่านตา ช่องตาดำ ฝ่ามือ และรูปหน้า เป็นต้น ในส่วนของเสียงพูด การลงลายมือชื่อ การใช้แป้นพิมพ์ ซึ่งจัดเป็นคุณลักษณะทางพฤติกรรมของบุคคลที่สามารถเปลี่ยนแปลงได้ตามกาลเวลาและการเรียนรู้ของเจ้าของ แต่ข้อดีของการใช้ไบโอเมทริกซ์ประเภทนี้ คือ สะดวก เป็นที่ยอมรับของผู้ใช้ และมีอัตราเสี่ยงต่อการติดเชื่อต่ำ เนื่องจากไม่ต้องนำอวัยวะที่ไวต่อการติดเชื่อ (เช่น ดวงตา) ไปสัมผัสกับอุปกรณ์ที่ใช้ในการอ่านข้อมูล

#### 2.4.4 ไบโอเมทริกซ์ประเภทต่างๆ

##### 1) การรู้จำใบหน้า (Facial recognition)

ไบโอเมทริกซ์ที่ใช้ใบหน้า ได้รับการพิจารณาและยอมรับกันอย่างกว้างขวาง ว่าเป็นวิธีที่เป็นกลางในการพิสูจน์ตัวบุคคล เทคโนโลยีนี้มีความสามารถที่น่าสนใจอยู่ 2 อย่างคือ การรู้จำใบหน้า (Facial recognition) และสร้างหน้าใหม่ (Face reconstruction) สามารถใช้กับลักษณะที่ซับซ้อนได้ใช้ได้กับรูปร่างลักษณะหลายชนิดสามารถทำงานได้อย่างรวดเร็ว แม่นยำ และใช้ข้อมูลสำหรับเปรียบเทียบขนาดเล็กข้อมูลที่ใช้สามารถทำให้อยู่ในรูปข้อมูลทางสถิติได้

##### 2) การรู้จำม่านตา (Iris recognition)

การรู้จำม่านตาทำงานได้โดยไม่ต้องมีการสัมผัสกับเครื่องมือ และง่ายต่อการใช้งาน ในขณะที่สามารถทำงานได้อย่างมีประสิทธิภาพในการรักษาความปลอดภัยสูง กลายเป็นเทคโนโลยีที่ได้รับการยอมรับอย่างกว้างมีการทำงานที่มีประสิทธิภาพและความแม่นยำสูงรวดเร็ว มีระบบการตรวจสอบที่ง่าย และไม่ต้องมีการสัมผัสกับอุปกรณ์ม่านตาของมนุษย์ไม่มีการเปลี่ยนแปลงไปตามอายุขัย ความเหมือนกันของข้อมูลมีน้อย ทำให้มีความสับสนกันน้อย

##### 3) การรู้จำลายนิ้วมือ (Fingerprint recognition)

การรู้จำลายนิ้วมือมีการทดลอง, ทดสอบ และการทำงานด้านความปลอดภัยในการพิสูจน์บุคคล มีการใช้กับงานด้านต่างๆ กันในหลายจำพวก การรู้จำลายนิ้วมือมีข้อได้เปรียบดังต่อไปนี้ ทนทาน และสามารถเชื่อถือได้ใช้งานง่ายใช้กันอย่างกว้างขวางและยอมรับโดยทั่วไปได้รับการพิสูจน์แล้วว่าใช้งานได้จริงหลากหลายของการนำไปประยุกต์ใช้

##### 4) การรู้จำเสียง (Voice recognition)

เทคโนโลยีนี้ ทำงานได้อย่างมีประสิทธิภาพและความแม่นยำสูง เพื่อการยืนยันตัวบุคคลด้วยเสียงของตัวเอง เสียงพูดเป็นสิ่งที่มีติดตัวกันทุกคน และสะดวกในการใช้งานในสถานที่ต่างๆ โดยเฉพาะการใช้งานแบบไร้สัมผัสเป็นการทำงานที่มีประสิทธิภาพ และความถูกต้องมากยืนยันตัวตนเอกลักษณ์ของบุคคลได้จากเสียงของเขาทั้งหลายเองกระบวนการง่ายๆ สามารถฝังรวมกับระบบต่างๆ ได้หลากหลายเทคโนโลยีนำไปใช้กับระบบที่ต้องใช้บัตรหรือไม่ต้องใช้บัตรก็ได้มีความยืดหยุ่นสำหรับการนำไปใช้ในระบบจริง

### 2.5 คำจำกัดความและขอบเขตของการรู้จำแบบ

เนื่องจากมีปัจจัยหลายอย่างที่จำกัดการออกแบบ pattern recognition จึงทำให้การวิจัยในเรื่องนี้เป็นไปได้ยาก แต่ถ้าเราไม่เข้าใจพฤติกรรมของมนุษย์เราก็ไม่สามารถที่จะทำสำเร็จได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรู้จำรูปแบบ (Pattern recognition) ต้องอาศัยการรับรู้และการจำก่อน รวมทั้งประสบการณ์ในอดีตกับปัจจุบันสัมพันธ์กัน ตัวอย่างเช่น มีรูปสุนัขที่ไม่ชัดเจนและบิดเบือนอยู่รูปหนึ่ง ถ้ามีคนที่ไม่เคยได้รับรู้หรือมีประสบการณ์กับรูปนี้มาก่อนก็จะไม่สามารถตอบได้ว่ารูปนี้เป็นรูปอะไร แต่ถ้ามีคนเคยศึกษาหรือมีประสบการณ์มาก่อนก็จะสามารถตอบคำถามของรูปนี้ได้ถูกต้องถึงแม้ว่าในปัจจุบันเทคโนโลยีทางคอมพิวเตอร์และความก้าวหน้าใน pattern recognition และ artificial intelligence สูง การแยกภาพของมนุษย์ที่สมบูรณ์ยังคงเป็นหนทางที่ยาวไกล อย่างไรก็ตามก็ไม่สามารถขัดขวางความพยายามของนักวิจัยได้มีเทคนิคพื้นฐานจำนวนมากที่ใช้กับ pattern-recognition เกือบทั้งหมดไม่ว่าจะเป็นภาพ วัตถุ หรือ I-D time-vary signal เช่น คำพูด เทคนิคพื้นฐานได้แสดงไว้ดังนี้วิธีการในการรู้จำภาพใบหน้าตั้งแต่อดีตจนถึงปัจจุบัน ได้มีแนวคิดด้านการพิจารณามองเห็นใบหน้า และวิธีการในการตัดสินใจรู้จำใบหน้าที่แตกต่างกัน ซึ่งสามารถแบ่งได้เป็น 6 กลุ่มการวิจัย คือ

### 2.5.1 การเทียบเทมเพลต (Template Matching)

วิธีการทางเทียบเทมเพลต นี้ทำงานโดยการหาความสัมพันธ์ของภาพ 2 ภาพ โดยตรง ให้ผลอย่างมีประสิทธิภาพโดยมีอัตราในการรู้จำเป็น 100% เมื่อภาพมีขนาดเดียวกัน มีการวางอยู่ตรงกัน และมีการส่องสว่างของแสงเดียวกัน ถ้าภาพที่เข้ามาทดสอบมีการเปลี่ยนแปลงไปจากเดิม ต้องมีการประมวลผลก่อน (Preprocessing) เพื่อให้ภาพเข้าสู่ตำแหน่งหรือการส่องสว่างใกล้เคียงกัน

### 2.5.2 ลักษณะทางเรขาคณิต

มีนักวิจัยจำนวนมากที่เลือกรู้จำใบหน้าในลักษณะทางเรขาคณิต คือ อาศัยอัตราส่วนหรือสัดส่วนของระยะทางจากตาซ้ายตาขวา จากตาไปจมูก จากจมูกไปปาก รูปร่างของปาก รูปร่างของตา และจากรูปร่างของคาง ซึ่งเป็นลักษณะทางกายภาพของใบหน้า ดังมีรายละเอียดของการวิจัย เช่น ใช้อัตราของระยะทาง (Ratios of Distance) โดยมีผลการรู้จำอยู่ระหว่าง 45-75% ของฐานข้อมูลภาพ 20 คน ใช้การคำนวณเซตของลักษณะทางเรขาคณิต เช่น ความกว้างของจมูก ตำแหน่งของปากบน ใบหน้า และรูปร่างของคางใช้เทคนิคระยะทางผสม (Mixture Distance) ระบบการรู้จำใบหน้าแบบที่มีการพิจารณาทางลักษณะทางเรขาคณิตนี้ จะสามารถนำไปใช้เป็นระบบที่มีประโยชน์และเป็นไปได้ในทางปฏิบัติในการค้นหาในฐานข้อมูลขนาดใหญ่ได้ถ้ามีการวัดระยะทางจากจุดที่เป็นลักษณะเด่นได้อย่างถูกต้องเที่ยงตรง ทั้งนี้เพราะข้อมูลที่ได้จะมีจำนวนน้อยมากเมื่อเทียบกับการพิจารณาแบบอื่น จึงใช้เวลาน้อยกว่าแบบอื่นๆ ในการค้นหา ดังนั้น จึงมีการสร้างระบบฐานข้อมูลมาสนับสนุนวิธีการแบบนี้ ซึ่งฐานข้อมูลดังกล่าวเรียกว่าฐานข้อมูลภาพหลายด้าน (Mugshot Database) ทั้งนี้เพราะถ้าเรามีภาพด้านข้างด้วยจะเป็นการช่วยหาจุดที่เป็นลักษณะเด่นได้เที่ยงตรงแม่นยำมากขึ้น แต่ทั้งนี้ความเที่ยงตรงในการวัดยังขึ้นอยู่กับอัลกอริทึมที่ใช้ด้วย ดังนั้น ในปัจจุบันการทำงานอย่างอัตโนมัติก็ยังไม่ค่อยมีความเที่ยงตรงนัก จึงต้องรอกันต่อไปสำหรับวิธีการแบบนี้

### 2.5.3 การเทียบกราฟ (Graph Matching)

เป็นการมองใบหน้าในลักษณะที่เป็นเวกเตอร์ของกราฟซึ่งมีจุดและเส้นในการเชื่อมต่อเป็นโครงร่างที่ปิดไปบนใบหน้าบนลักษณะเด่นที่พิจารณา เช่น รูปร่างของกราฟของตา ปาก คางกับหู และคิ้ว เป็นต้น

### 2.5.4 ไอเกนเฟซ (Eigenface)

MIT (Massachusetts Institute of Technology) ได้ริเริ่มและเสนอวิธีการในการรู้จำใบหน้าโดยทำการฉาย (Projection) ภาพใบหน้าไปยังองค์ประกอบหลัก (Principal Components) โดยเรียกภาพใบหน้าที่ดังกล่าวนี้ว่าไอเกนเฟซ (Eigenface) โดยนำใบหน้าไอเกนดังกล่าวไปทำการค้นหาในฐานข้อมูลใบหน้าไอเกน โดยพวกเขาได้แสดงให้เห็นว่าโดยการทดสอบกับฐานข้อมูลใบหน้าไอเกน 16 คนที่มีการเลื่อน การวางตำแหน่งศีรษะหลายๆ ลักษณะ มีการย่อขยาย และการเปลี่ยนแปลงของแสง โดยการแทนด้วยใบหน้าไอเกนที่ใช้เทคนิคของพวกเขาแล้วจะมีการเปลี่ยนแปลงเล็กน้อย โดยระบบยังคงมีความสามารถในการรู้จำ เมื่อมีการเปลี่ยนแปลงของแสงมีความสามารถในการรู้จำ 96% มีการวางตำแหน่งศีรษะเปลี่ยนแปลงจะมีความสามารถในการรู้จำ 85% และมีการย่อขยายจะภาพมีความสามารถในการรู้จำ 64% โดยในการย่อขยายนั่นพวกเขาใช้อัลกอริทึมในการปรับขนาดศีรษะให้มีขนาดเดียวกับขนาดใบหน้าไอเกนโดยใช้การประมาณขนาดของศีรษะ และนำส่วนกลางของใบหน้ามาใช้เท่านั้นเพื่อลดผลจากการเปลี่ยนแปลงของทรงผม และฉากด้านหลังที่อาจเกิดการเปลี่ยนแปลงได้

### 2.5.5 โครงข่ายประสาทเทียม (Neural Network)

จากอดีตจนถึงปัจจุบันได้มีการเสนอบทความที่ใช้โครงข่ายประสาทเทียมในการรู้จำใบหน้ามากมาย แต่ส่วนใหญ่แล้วนำเสนอเสียด้วยวิธีการต่างๆ ที่เสนอนั้นมีการทดสอบกับฐานข้อมูลขนาดเล็กๆ ได้เท่านั้น (ต่ำกว่า 20 คน) จะมีที่ใช้ฐานข้อมูลขนาดใหญ่ไม่มากนัก

### 2.5.6 ภาพ 3 มิติ

การมองภาพใบหน้าเป็น 3 มิติ เป็นวิธีการขั้นสูงและเป็นวิธีที่ดีมากในปัจจุบันและมีการทำเป็นธุรกิจผลิตซอฟต์แวร์เพื่อเป็นการค้าเรียบร้อยแล้ว Facelt คือ ชื่อทางการค้าของบริษัท Visionics Corporation ในการรู้จำภาพในหน้ารูปแบบ 3 มิติแบบเวลาจริงที่ไม่ต้องการฮาร์ดแวร์พิเศษในโลกแห่งความเป็นจริง Facelt ใช้การมองภาพในรูปแบบสเตอริโอของภาพ 3 มิติที่ประกอบด้วยบล็อกความสูง โดยเรียกว่า หัวไอเกน (Eigen Head) ที่เป็นตัวแสดงภาพ 3 มิติของศีรษะที่ได้มาจากข้อมูลเฉดสี (Shading Information) ของภาพ 2 มิติ ซึ่งหัวไอเกนนี้จะไม่ขึ้นกับแสงของการส่องสว่างและท่าทางของการวางศีรษะ และมีการส่งภาพ 3 มิติที่มีการนอร์มอลไลต์แล้วแปลงไปเป็นรหัสด้วยการวิเคราะห์ลักษณะเด่นแบบท้องถิ่น (Local Feature Analysis) เช่น จมูก ปาก แก้ม กระจก และแนวขากรรไกร ซึ่งจะสร้างสิ่งที่เป็นเอกลักษณ์ของบุคคลคนเดียวกันที่เรียกว่า พิมพ์ของใบหน้า (Faceprint) พิมพ์ของใบหน้าสามารถนำไปค้นหาในฐานข้อมูลได้แบบ real time และมีความรวดเร็วในการประมวลผลสูงมากจนกระทั่งสามารถทำกับภาพที่เป็นภาพนิ่งและกับภาพจากวิดีโอก็ได้ ซึ่งจากความสามารถจริงของระบบอยู่ที่ความเร็ว และการแก้ไขปัญหาเรื่องการขมวดคิ้ว การยิ้ม การกระพริบตา การใส่แว่น ปัญหาของแสง การย่อขยาย การวางตำแหน่งศีรษะ และการเอกสาร์นี้เป็นเอกสาร์ที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้ไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอียงได้ทั้งหมด อีกทั้งยังเป็นแนวทางที่ใช้ลักษณะ 3 มิติที่เป็นลักษณะเด่นแบบท้องถิ่น คือ จมูก ปาก แก้ม กระตูด และแนวขากรรไกร ที่สามารถจะพัฒนาเป็นการประมาณใบหน้าเมื่ออายุเพิ่มมากขึ้น หรือลดลงได้อีกด้วย จึงเป็นสุดยอดแนวทางของการรู้จำใบหน้าอย่างแท้จริงที่สามารถใช้ในการค้นหาประวัติของเด็กหลงทางที่ไม่สามารถบอกบ้านหรือพ่อแม่ได้ หรือค้นหาคนที่หายไปหลายๆ ปีได้ว่าจะมีหน้าตาในปัจจุบันเป็นอย่างไร

## 2.6 การหาค่าประกอบสำคัญของใบหน้า

การหาค่าประกอบสำคัญของภาพใบหน้าสามารถแยกออกเป็น 2 ประเภทใหญ่ๆ ได้แก่

### 2.6.1 Appearance based method

คือ การมองแต่ละจุดในภาพเป็นตัวแปร และแยกแยะความแตกต่างของภาพจากค่าการกระจายตัวทางสถิติของค่าสีแต่ละพิกเซลทั้งหมดในภาพ ในวิธีนี้เราไม่จำเป็นต้องหาตำแหน่งของตา จมูก และปากภายในภาพ

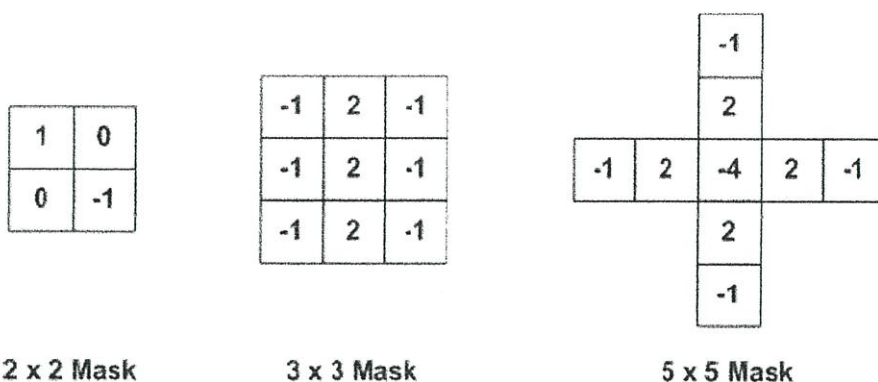
### 2.6.2 Model based method

คือ การมองภาพโดยรวม และแยกแยะใบหน้าของแต่ละคนจากตำแหน่งสำคัญๆ บนใบหน้า เช่น ตาม จมูก ปาก ให้ได้ก่อน แล้วจึงหาความสัมพันธ์ของตำแหน่งต่างๆ เหล่านี้เพื่อแยกแยะใบหน้าต่อไป

ในโครงการนี้ได้เลือกกระบวนการ Principle component analysis (PCA) ซึ่งเป็นหนึ่งในกระบวนการแบบ Appearance based method โดยทำการฉายภาพใบหน้าไปยังแกนของความแปรปรวนร่วม (covariance) ที่มากที่สุด เพื่อให้หาความแตกต่างของภาพแต่ละภาพได้แม่นยำขึ้น และวิธีนี้ยังนำหลักการไอเกนเวกเตอร์ (eigenvector) ซึ่งจะเรียกว่าไอเกนเฟซ (eigenface) มาใช้เพื่อลดการคำนวณลงและทำให้การทำงานมีประสิทธิภาพมากยิ่งขึ้น

## 2.7 การเพิ่มสัญญาณรบกวน

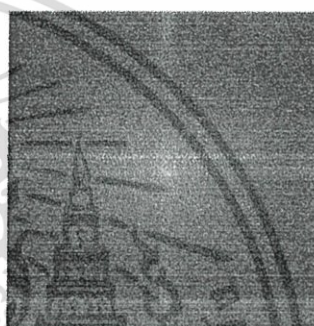
ในงานวิจัยนี้การประมวลผลต่างๆจะดำเนินการบนสเปเชียลโดเมน (Spatial Domain) ซึ่งเป็นรูปแบบปกติในการแสดงผลภาพเชิงเลข การคำนวณต่างๆสามารถกระทำที่จุดภาพได้โดยตรงในการประมวลผลจะต้องกำหนดขนาดของคอนโวลูชันมาสก์เสมอ (Convolution Mask) โดยมีขนาดเท่ากันทั้งแถวและหลัก ขนาดของมาสก์จะส่งผลต่อความไวต่อสัญญาณรบกวน (Noise's sensitivity) การกำหนดมาสก์ขนาดเล็กจะทำให้มีค่าความไวสูง ในทางตรงกันข้ามหากกำหนด มาสก์ขนาดใหญ่จะส่งผลให้ความไวต่อสัญญาณรบกวนมีค่าต่ำ เนื่องจากการคำนวณมีการถ่วงน้ำหนักจากจุดภาพโดยรอบจำนวนมาก ตัวอย่างการกำหนดขนาดของมาสก์แสดงได้ดังรูปที่ 2.6 แต่ในการประมวลผลภาพอาจใช้แบบโดเมนความถี่ก็ได้ (Frequency Domain) โดยการคำนวณจะอยู่บนพื้นฐานของการแปลงฟูเรียร์ (Fourier Transform) การประมวลผลภาพด้วยวิธีนี้จะพบบ่อย ในการบีบอัดข้อมูล (Data Compression) ตัวอย่างของการแปลงฟูเรียร์แสดงได้ดังภาพที่ 2.7



รูปที่ 2.7 มาสค์ขนาดต่างๆ ที่ใช้ในการประมวลผลภาพบนสเปเชียลโดเมน

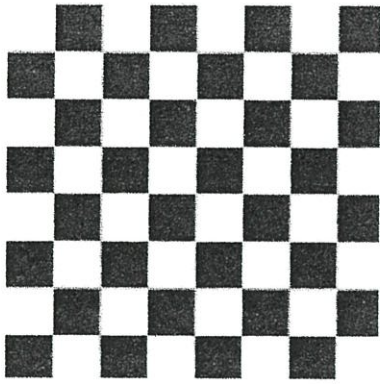


รูปที่ 2.8 ภาพที่แสดงบนสเปเชียลโดเมน

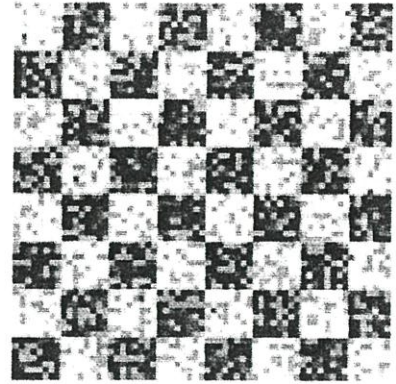


รูปที่ 2.9 ภาพที่แสดงบนโดเมนความถี่

สัญญาณภาพโดยทั่วไปจะประกอบด้วยสิ่งที่มีผิดไปจากสัญญาณในอุดมคติซึ่งอาจจะเกิดจากกระบวนการเก็บข้อมูลหรือจากกระบวนการประมวลผลสัญญาณเอง สิ่งแปลกปลอมเหล่านี้ถูกเรียกว่าสัญญาณรบกวน (Noise) บางครั้งลักษณะของสัญญาณรบกวนอาจเป็นแบบสุ่มเกิดขึ้นไม่แน่นอน และมีแหล่งที่มาหลายแบบยกตัวอย่างเช่น ความไม่คงที่ของตัวตรวจจับ ความเปลี่ยนแปลงของสภาพแวดล้อมโดยรอบ ความไม่ต่อเนื่องของสัญญาณ ความผิดพลาดในการประมาณค่า และอื่นๆ รูปที่ 2.9 แสดงลักษณะของภาพตั้งต้นและภาพที่มีสัญญาณรบกวนแบบพริกไทย (Pepper Noise) การปรับปรุงคุณภาพของภาพที่จะนำมาประมวลผลให้ดีขึ้น มีประโยชน์กับการวิเคราะห์องค์ประกอบภาพในหลายด้าน เช่น การแปลงภาพที่ไม่ชัดเจนเนื่องจากการถ่ายในขณะแสงน้อยหรือจ้าเกินไปให้ชัดเจนขึ้น, การกำจัดสัญญาณรบกวน, การหาขอบภาพ, การสร้างภาพไบนารี เป็นต้น



รูปที่ 2.10 ภาพตารางหมากรุก



รูปที่ 2.11 ภาพที่ประกอบด้วยสัญญาณรบกวนแบบพริกไทย

### 2.8 การกำจัดสัญญาณรบกวน (Noise Reduction)

การกำจัดสัญญาณรบกวนบนภาพใช้ในการกำจัดสิ่งไม่พึงประสงค์ที่มีขนาดเล็กออกจากภาพก่อนที่จะนำภาพนั้นไปประมวลผลในขั้นตอนอื่นต่อไป เมื่อพิจารณาภาพบนสเปเชียลโดเมนจุดภาพที่มีค่าระดับสีที่แตกต่างจากจุดรอบข้างมากๆ จุดเหล่านี้มักอยู่เดี่ยวๆไม่ค่อยมีความต่อเนื่องกับจุดโดยรอบซึ่งการกำจัดสัญญาณรบกวนบนภาพนั้นมีวิธีต่างๆ ดังนี้

1) การกำจัดสัญญาณรบกวนด้วยตัวกรองเชิงเส้น เป็นวิธีที่อาศัยการหาผลรวมของค่าที่ได้จากการถ่วงน้ำหนักระดับความเข้มแสงที่มีพิกัดสอดคล้องกับพิกัดบนมาส์คที่นำมาคอนโวลูชัน (Convolution) กับข้อมูลภาพ ตัวอย่างการกำจัดสัญญาณรบกวนด้วยวิธีนี้คือ ตัวกรองแบบเฉลี่ยข้อมูล (Mean Filter) โดยการหาค่าเฉลี่ย (Average) ของจุดภาพในตำแหน่งที่พิจารณาและจุดที่อยู่โดยรอบ ส่งผลให้จุดภาพที่พิจารณามีค่าความเข้มแสงแตกต่างจากจุดโดยรอบลดลง

23	57	110
60	87	114
62	77	112

รูปที่ 2.12 มาส์คข้อมูลภาพที่พิจารณา

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

รูปที่ 2.13 ตัวกรองแบบเฉลี่ยข้อมูล

2) การกำจัดสัญญาณรบกวนด้วยตัวกรองแบบไม่เป็นเชิงเส้น เป็นเทคนิคที่ช่วยลดสัญญาณรบกวนได้โดยการพิจารณาจากตำแหน่งของลำดับความเข้มแสง วิธีที่นิยมใช้คือตัวกรองแบบค่ามัธยฐาน (Median Filter) ใช้หลักการหาค่ามัธยฐานของข้อมูลในมาส์คที่พิจารณา วิธีนี้จะช่วยลดสัญญาณรบกวนในขณะที่ยังคงความเข้มแสงค่าเดิมของภาพตั้งต้นไว้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

123	125	126	130	140
122	124	126	127	135
118	120	150	125	134
119	115	119	123	133
111	116	110	120	130

**Neighbourhood values:**

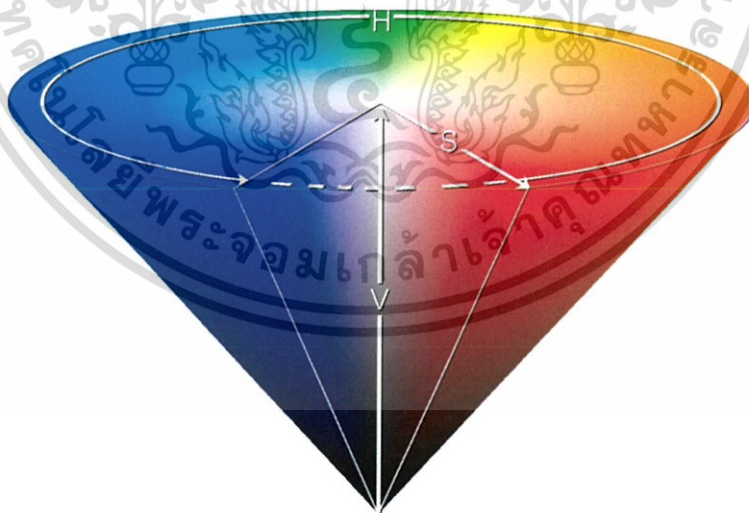
**115, 119, 120, 123, 124,  
125, 126, 127, 150**

**Median value: 124**

รูปที่ 2.14 ข้อมูลภาพที่พิจารณาด้วยมาสค์ขนาด  $3 \times 3$  และค่ามัธยฐานของข้อมูล

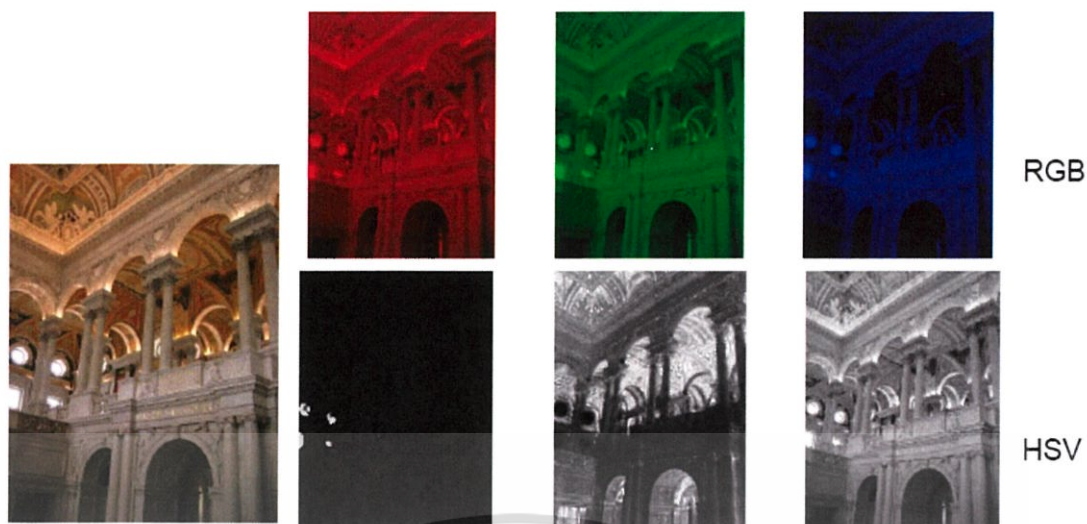
## 2.9 ระบบสี HSV (Hue, Saturation, Value)

หรือ HSB (Hue, Saturation, Brightness) เป็นระบบสีที่นิยมใช้กันในหมู่นักแต่งภาพ เนื่องจากเป็นระบบสีที่ใกล้เคียงกับความคิดของมนุษย์ได้ดีกว่าระบบสี RGB โดย Hue คือสีของภาพ, Saturation คือ ปริมาณความอิ่มตัวของสี ยิ่งมีค่ามาก ภาพจะมีสีสดยิ่งมีน้อย ภาพจะยังมีสีน้อยลง จนในที่สุดจะกลายเป็นรูปที่ลักษณะแบบ Grayscale และ Value หรือ Brightness เป็นค่าที่แสดงถึง ปริมาณความสว่างของภาพ หากมีค่ามากภาพจะยิ่งมีความสว่างมาก



รูปที่ 2.15 โมเดลสี HSV ในรูปแบบโคน (Cone)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.16 รูปที่เก็บในแต่ละ channel ในระบบสี RGB และ HSV

## 2.10 การดึงวัตถุพื้นหน้าออกจากพื้นหลังโดยใช้ Threshold

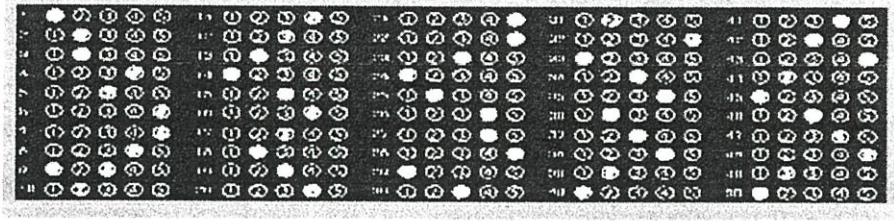
กระบวนการ Segmentation เป็นกระบวนการแยกวัตถุ หรือองค์ประกอบต่างๆ ออกจากภาพอินพุท วิธีการ Intensity thresholding เป็นการแยกแยะวัตถุจากฉากหลังโดยดูจาก Intensity ของพิกเซลเป็นหลัก ข้อดีของวิธีการนี้คือมีขั้นตอนในการทำงานที่ง่ายไม่ซับซ้อนทำงานได้รวดเร็ว แต่ข้อเสียคือไม่สามารถใช้กับภาพที่มีสัญญาณรบกวนมาก หรือภาพที่มีความสว่างไม่สม่ำเสมอ เทคนิคการทำขีดแบ่ง (Thresholding Techniques) เป็นการพิจารณาว่าจุดภาพใดควรเป็นจุดขาวหรือจุดดำ ซึ่งทำได้โดยการเปรียบเทียบระหว่างจุดภาพเริ่มต้นกับค่าคงที่ค่าหนึ่ง เรียกว่าค่าขีดแบ่ง ซึ่งเป็นค่าความเข้มแสงค่าหนึ่งที่ใช้แยกแยะประเภทของจุดภาพ เทคนิคนี้ใช้กันมากในกรณีที่มีข้อมูลภาพมีลักษณะแตกต่างกันระหว่างวัตถุและพื้นหลัง ในขั้นตอนนี้ทำการตัดพื้นหลังออกจากพื้นหน้าโดยใช้วิธี Threshold ที่มีระดับความเข้มอยู่ระหว่างกลุ่มทั้งสองของ Histogram ซึ่งค่า Threshold ที่ได้จะอยู่ระหว่าง 0-255 เท่านั้น Threshold จะถูกนำไปเพื่อเปรียบเทียบค่าของแต่ละพิกเซล หากค่า  $f(x, y)$  น้อยกว่า Threshold จุดพิกเซลนั้นจะถูกปรับให้เป็นสีดำหรือส่วนของวัตถุ และหากค่า  $f(x, y)$  มากกว่าหรือเท่ากับ Threshold จุดพิกเซลนั้นจะถูกปรับให้เป็นสีขาวหรือส่วนของพื้นหลัง สามารถเขียนแทนด้วยสมการดังนี้

$$f_{thr}(x, y) = \begin{cases} 1, & f_f(x, y) < Threshold \\ 0, & f_f(x, y) \geq Threshold \end{cases} \quad (2.1)$$

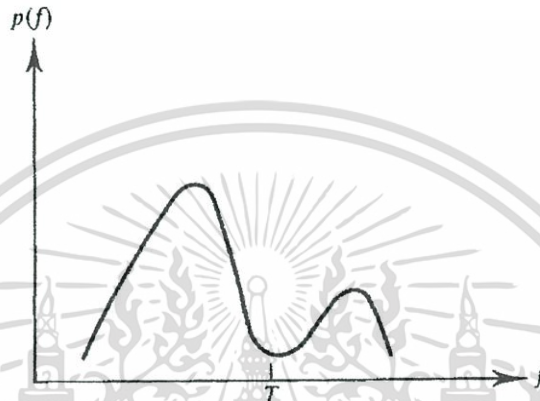
โดยกำหนดให้

- 1 คือ สีดำ ซึ่งเป็นส่วนของวัตถุ
- 0 คือ สีขาว ซึ่งเป็นส่วนของพื้นหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 หลังจากตั้งพื้นหน้าออกจากพื้นหลัง



รูปที่ 2.18 Bimodal image histogram

### 2.11 การตรวจจับวัตถุโดยใช้ HSV

ระบบสี HSV (Hue, Saturation, Value) หรือ HSB (Hue, Saturation, Brightness) เป็นระบบสีที่นิยมใช้กันในหมู่นักวิจัย เนื่องจากเป็นระบบสีที่ใกล้เคียงกับความคิดของมนุษย์ได้ดีกว่าระบบสี RGB โดย Hue คือสีของภาพ, Saturation คือ ปริมาณความอิ่มตัวของสี ยิ่งมีค่านี้มาก ภาพจะมีสีสดยิ่งมีน้อย ภาพจะยิ่งมีสีนี้ลดลง จนในที่สุดจะกลายเป็นรูปที่ลักษณะแบบ Grayscale และ Value หรือ Brightness เป็นค่าที่แสดงถึงปริมาณความสว่างของภาพ หากมีค่ามากภาพจะยิ่งมีความสว่างมาก ในการตัดพื้นหลังออกจากภาพใช้วิธีการแปลงค่าสีจาก RGB เป็น HSV และทำการหาช่วงของสีของพื้นหลัง ที่ซึ่ง HSV สามารถแปลงจาก RGB ได้ดังนี้

$$v = \max, \quad (2.2)$$

$$s = \begin{cases} \frac{(\max - \min)}{\max} & \text{if } \max \neq 0 \\ 0 & \text{if } \max = 0 \end{cases}$$

(2.3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

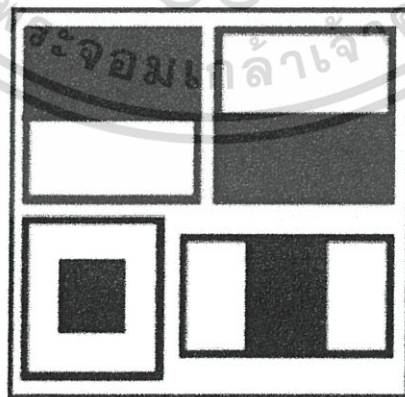
$$H = \begin{cases} -60, \text{if}(S = 0) \\ \frac{(G-B)}{(\max-\min)} \times 60, \text{if}(R = \max) \\ 2 + \frac{(B-R)}{\max-\min} \times 60, \text{if}(G = \max) \\ 4 + \frac{(R-G)}{(\max-\min)} \times 60, \text{if}(B = \max) \end{cases} \quad (2.4)$$

เมื่อ  $\max = \sup (R, \sup (G, B))$  คือค่าสูงสุดของหนึ่งจุดสี RGB และ  $\min = \inf (R, \inf (G, B))$  คือค่าต่ำสุดของหนึ่งจุดสี RGB ตามลำดับ

## 2.12 กระบวนการ Haar like feature

หลังจากได้บริเวณคร่าวๆของใบหน้ามาแล้วนั้น ในขั้นตอนนี้ทำการตรวจจับใบหน้าโดยใช้ Haar like-Feature ตามวิธีของ Viola และ Jones [7] ซึ่งเป็นวิธีการตรวจจับและตีความวัตถุภายในภาพ ด้วยการสร้าง Feature ที่แสดงถึงผลต่างระหว่างพื้นที่ส่วนสีขาว และส่วนที่เป็นสีดำ ซึ่ง Feature สามารถเปลี่ยนแปลงขนาด และตำแหน่งได้ ใช้สำหรับการตรวจจับลักษณะบนภาพแบบต่าง เช่น เส้นตรง, วงกลม เป็นต้น โดยใช้การตรวจจับวัตถุในภาพพร้อมกับแนวคิดหลัก 4 แนวคิดดังนี้

- ใช้ Haar features
  - ใช้การรวมภาพ(Integral Image) เพื่อตรวจจับลักษณะเด่นอย่างรวดเร็ว (Rapid Feature Detection)
  - ใช้ The Adaboost machine-learning method
  - ใช้ Cascaded classifier เพื่อรวมลักษณะเด่นต่างๆ เข้าด้วยกันอย่างมีประสิทธิภาพ
- การมีอยู่ของลักษณะเด่นของ Haar ถูกกำหนดโดยการลบค่าเฉลี่ยบริเวณจุดภาพมืดออกจากค่าเฉลี่ยบริเวณจุดภาพสว่าง ถ้าผลลัพธ์ที่ได้มีค่ามากกว่าค่าขีดแบ่ง (ถูกตั้งค่าในระหว่างการเรียนรู้) แสดงว่ามีลักษณะเด่นของ Haar อยู่

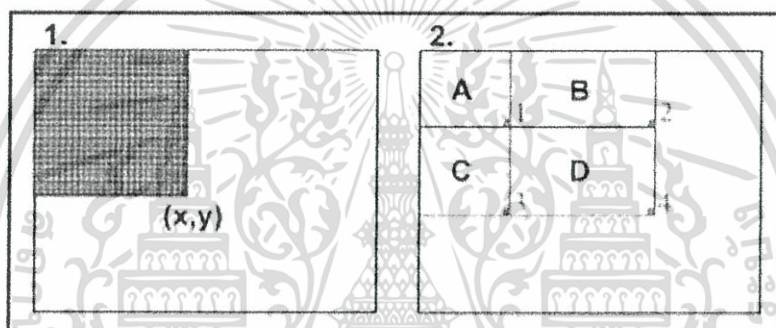


รูปที่ 2.19 แสดงตัวอย่างลักษณะเด่นของ Haar ที่ถูกใช้ใน OpenCV

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

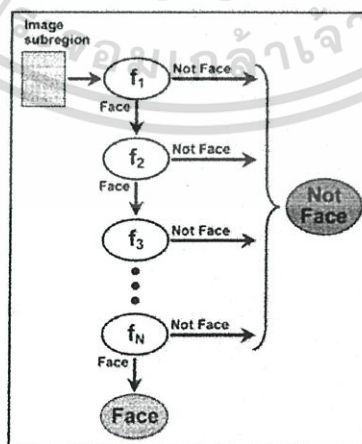
การตรวจสอบหาลักษณะเด่นของ Haar ในแต่ละส่วนของภาพอย่างมีประสิทธิภาพนั้น Viola และ Jones ใช้เทคนิคที่เรียกว่าการรวมภาพ (Integral Image) โดยการรวมจุดภาพเล็กๆเข้าด้วยกัน จากรูปที่ 2.20 การจะหาค่าของรูปสี่เหลี่ยม D คือการนำ  $A+B+C+D$  (ตำแหน่งที่ 4) ลบออกด้วย  $A+B$  (ตำแหน่งที่ 2) และ  $A+C$  (ตำแหน่งที่ 3) และบวกเพิ่มเข้าไปด้วย  $A$  (ตำแหน่งที่ 1) นั่นคือ  $D=(A+B+C+D)-(A+B)-(A+C)+(A)$  ซึ่งมีค่าเท่ากับ  $(x_4,y_4)-(x_2,y_2)-(x_3,y_3)+(x_1,y_1)$

ในการเลือกลักษณะเด่นของ Haar และการตั้งระดับค่าขีดแบ่ง Viola และ Jones ใช้ machine-learning method ที่เรียกว่า “Adaboost” ซึ่งรวม weak classifier (ได้รับคำตอบที่ถูกต้องมากกว่าการเดาสุ่มเพียงเล็กน้อย) จำนวนมากเข้าด้วยกันเพื่อสร้าง strong classifier โดยใช้หลักที่ว่า weak classifier แต่ละตัวค่อยๆ ผลักดันคำตอบสุดท้ายไปในทิศทางที่ต้องการ Adaboost จะทำการเลือกชุดของ weak classifier ในการรวมและกำหนดน้ำหนักของแต่ละตัว การรวมกันของน้ำหนักก็คือ strong classifier นั่นเอง Viola และ Jones [7] รวมชุดของ Adaboost classifiers เป็นเหมือนโซ่กรองดังแสดงในรูปที่ 2.21 ซึ่งมีประสิทธิภาพสูงในการแบ่งพื้นที่ภาพ



รูปที่ 2.20 แสดงเทคนิคการรวมภาพ

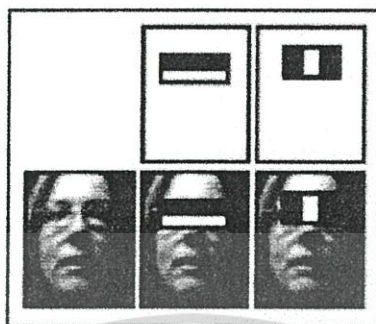
หลังจากที่รวมภาพแล้ว จุดภาพที่ตำแหน่ง  $(x,y)$  จะรวมค่าของทุกจุดภาพในสี่เหลี่ยมสีดำ ผลรวมค่าของทุกจุดภาพในสี่เหลี่ยม D คือ  $(x_4,y_4)-(x_2,y_2)-(x_3,y_3)+(x_1,y_1)$



รูปที่ 2.21 แสดงสายโซ่ของตัวกรอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บริเวณเล็กๆของภาพซึ่งสามารถผ่านตัวกรองทั้งหมดจะถูกจัดว่าเป็นใบหน้าส่วนที่เหลือถูกจัดว่าไม่ใช่ใบหน้า ตำแหน่งของตัวกรองในลำดับอยู่บนพื้นฐานของค่าน้ำหนักความสำคัญที่ Adaboost กำหนดไว้ ตัวกรองที่มีค่าน้ำหนักสูงสุดจะอยู่ในลำดับแรกเพื่อกำจัดส่วนของภาพที่ไม่ใช่ใบหน้าออกไป



รูปที่ 2.22 แสดงตัวอย่างของตัวกรองสองตัวแรกใน Viola-Jones Cascade

จากรูปที่ 2.22 จะพบว่าตัวกรองแรกใช้ความจริงที่ว่าบริเวณแก้มสว่างกว่าบริเวณดวงตา และตัวกรองตัวที่สองใช้ความจริงที่ว่าบริเวณสันจมูกจะสว่างกว่าบริเวณดวงตา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

# กระบวนการรู้จำ

บทนี้กล่าวถึงกระบวนการรู้จำภาพใบหน้าด้วยวิธีการต่างๆที่ใช้ในงานวิจัยนี้ อันประกอบไปด้วย ทฤษฎีและวิธีการโอเคนเฟซ, Linear Discriminate Analysis (LDA), Independent Components Analysis (ICA) และการแยกคุณลักษณะของใบหน้าทั้งใบหน้าและเฉพาะส่วนโดยใช้แบบจำลองเชิงเรขาคณิตโดยทำการเก็บฐานข้อมูลในส่วนของตาและปากของภาพใบหน้าแต่ละภาพไว้และนำมาเข้ากระบวนการ PCA ซึ่งในแต่ละวิธีจะทำให้ได้ค่าโอเคนเวกเตอร์ของภาพใบหน้าและส่วนประกอบของใบหน้าในแต่ละส่วนเพื่อนำมาสู่กระบวนการรู้จำภาพใบหน้าต่อไป

### 3.1 ทฤษฎีและวิธีการโอเคนเฟซ

#### 3.1.1 วิธีของโอเคนเฟซ

จากการศึกษาก่อนหน้านี้ของระบบจดจำใบหน้า โดยส่วนมากแล้วการทำงานของระบบนี้จะจดจำใบหน้าอย่างซีเฉพาะเจาะจง โดยการวัดค่าจากตัวกำหนดที่สัมพันธ์กันและเหมาะสมจากข้างต้นนี้ได้ให้ข้อมูลในการบ่อนรหัสและถอดรหัสรูปใบหน้านั้น จะเน้นที่จุดหลักๆ คือ ลักษณะหน้าตาโดยทั่วไปแล้วบางทีลักษณะหน้าตาอาจจะไม่ตรงกับลักษณะใบหน้าที่แท้จริงในบางส่วน เช่น ตา จมูก ริมฝีปาก และผมในทฤษฎีของ Language of information โดยต้องการที่จะคัดเลือกข้อมูลที่สัมพันธ์กันกับรูปหน้าตาเพื่อตั้งให้เป็นรหัสออกมาให้เหมือนมากที่สุดเท่าที่จะเป็นไปได้ และเปรียบเทียบกับรูปหน้าที่ตั้งรหัสโดยใช้ฐานข้อมูลของรูปพรรณสัณฐาน ขั้นตอนแรกที่จะคัดเลือกข้อมูลนี้ต้องประกอบด้วยรูปของใบหน้าที่ได้จากกลุ่มใบหน้าที่ได้ถ่ายมาโดยมีหลายรูปหน้า หน้าตาหลายๆ แบบ และใช้รูปภาพเหล่านี้มาตั้งใส่รหัสหลังจากนั้นมาเปรียบเทียบกับหน้าเป็นรายบุคคลไป

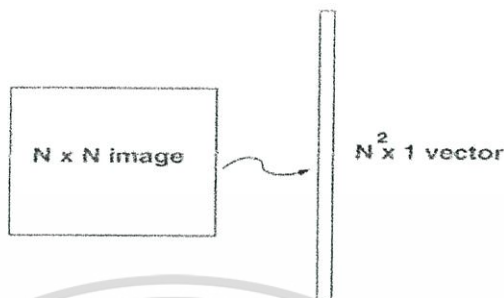
ในระบบทางคณิตศาสตร์นั้น ต้องการที่จะหองค์ประกอบที่สำคัญของส่วนประกอบใบหน้า หรือโอเคนเวกเตอร์ของโควาเรียนเมทริกซ์ ของรูปแบบใบหน้า โอเคนเวกเตอร์สามารถคิดแยกรูปแบบลักษณะรูปหน้าที่มีลักษณะคล้ายคลึงกันออกเป็นชุดๆ ได้ ซึ่งในแต่ละรูปจะมีโอเคนเวกเตอร์ที่แตกต่างกันออกไป ดังนั้นเราสามารถแสดงผลของโอเคนเวกเตอร์ออกมาอย่างง่ายๆ หยิบๆ ได้ ที่เราเรียกว่า โอเคนเฟซในแต่ละรูปหน้านั้นสามารถแสดงผลออกมาในรูปแบบของการรวมตัวของเส้นตรงต่างๆ ของโอเคนเฟซ จำนวนหมายเลขที่โอเคนเฟซจะเท่ากันจำนวนรูปของใบหน้าในชุดที่มี อย่างไรก็ตาม ใบหน้าเหล่านี้ก็สามารถประมาณหาโอเคนเฟซที่ดีที่สุดได้ ซึ่งนั่นจะมีค่าโอเคนเวกเตอร์มากที่สุด ดังนั้น จะมีความสัมพันธ์ก็จะมากในรูปชุดนี้ด้วย เหตุผลหลักๆที่ใช้โอเคนเฟซน้อยๆ เพื่อการคำนวณที่มีประสิทธิภาพมากยิ่งขึ้น

#### 3.1.2 ทฤษฎีการวิเคราะห์องค์ประกอบหลัก (Principal Component Analysis)

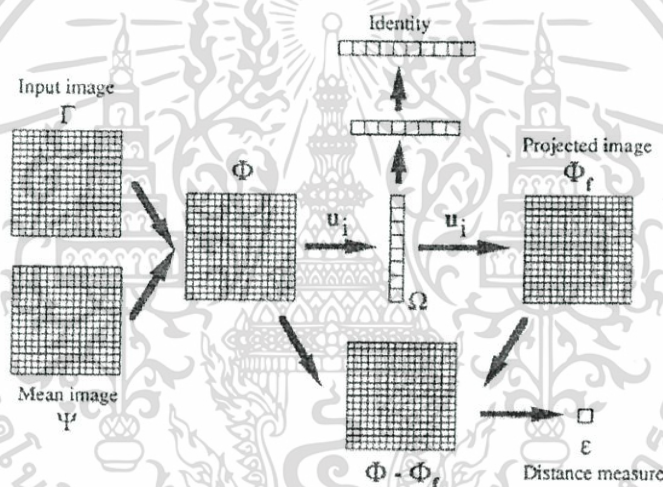
วิธีการวิเคราะห์องค์ประกอบหลัก (Principal Component Analysis-PCA) เป็นวิธีการทางสถิติ ซึ่งถูกนำไปประยุกต์ใช้ในงานต่างๆ เช่น การบีบอัดข้อมูล และการสร้างภาพใบหน้าโอเคนเฟซเพื่อใช้ในกระบวนการรู้จำ เป็นต้น วิธีนี้ถูกนำมาสร้างเป็นภาพใบหน้าโอเคน (Eigen Faces) ครั้งแรกโดย Sirovich and Kirby ในปี 1987 และต่อมา Turk and Pentland ได้สร้างระบบรู้จำใบหน้าโดย

เอกลักษณะเด่นของเอกสารนี้คือการนำเอาแนวคิดของ Eigen Faces มาใช้ในการรู้จำใบหน้าโดยการคำนวณค่า Eigen Faces จากชุดของภาพใบหน้าหลายๆ ภาพ และใช้ค่า Eigen Faces นี้ในการรู้จำใบหน้า โดยวิธีการนี้สามารถรู้จำใบหน้าได้แม้ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การนำภาพใบหน้าไอเจนในการรู้จำ โดยกำหนดให้  $I(x,y)$  คือภาพที่ใช้เป็น (2-D Image) 2 มิติ มีขนาด  $80 \times 80$  Pixel ขนาด  $(N \times N)$  ชนิด สีเทา เมื่อทำเป็นรูปภาพแบบ 1 มิติ  $N^2 \times 1$  (1-D Vector) มีความเท่ากับ 6,400 จุด ต่อ 1 ภาพ เพื่อทำเป็นชุด Training Set Image จากภาพใบหน้าต้นฉบับ



รูปที่ 3.1 ภาพ 2 มิติขนาด  $(N \times N)$  เมื่อทำเป็นรูปภาพแบบ 1 มิติ  $N^2 \times 1$  (1-D Vector)



รูปที่ 3.2 การทำงานของ PCA

โดยกำหนดให้  $T_1, T_2, T_3, \dots, T_M$  เป็นกลุ่มเรียนรู้ ของภาพใบหน้า การสร้างกลุ่มเรียนรู้ จะคำนวณค่าเฉลี่ย ค่าโควาเรียนซ์เมทริกและคำนวณค่าไอเจนเวกเตอร์ตามลำดับ มีขั้นตอนดังต่อไปนี้

ขั้นตอนที่ 1 คำนวณหาค่าเฉลี่ย ( $\Psi = \text{Average Face}$ ) จากภาพทุกใบหน้า ดังสมการที่ 3.1

$$\Psi = \frac{1}{M} \sum_{i=1}^M T_i \quad (3.1)$$

เมื่อ  $\Psi$  คือ ใบหน้าเฉลี่ย  
 $M$  คือ จำนวนใบหน้า  
 $T$  คือ กลุ่มเรียนรู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 2 คำนวณหาผลต่างระหว่างกลุ่มเรียนรู้และค่าเฉลี่ย หรือเรียกว่าค่าเบี่ยงเบนมาตรฐานของภาพใบหน้า ดังสมการที่ 3.2

$$\Phi = T_1 - \Psi \quad (3.2)$$

เมื่อ  $\Psi$  คือ ใบหน้าเฉลี่ย  
 $\Phi$  คือ ใบหน้าผลต่าง  
 $T$  คือ กลุ่มเรียนรู้

ขั้นตอนที่ 3 คำนวณหาค่าเมทริกความแปรปรวนร่วม โดยนำค่าเบี่ยงเบนมาตรฐานจากขั้นตอนที่ 2 แล้ว คูณกับค่าเบี่ยงเบนมาตรฐานที่ทำการทรานสโพสดังสมการที่ 3.3

$$C = \frac{1}{M} \sum_1^M \Phi_i \Phi_i^T \quad (3.3)$$

$L = AA^T$   
 ซึ่ง  $A = [\Phi_1, \Phi_2, \dots, \Phi_M]$  เป็นชุดข้อมูลค่าเบี่ยงเบนมาตรฐาน  
 เมื่อ  $C$  คือ เมทริกความแปรปรวนร่วม  
 $A$  คือ กลุ่มของเมทริกความแปรปรวนร่วม

ขั้นตอนที่ 4 คำนวณหา ไอเกนเวกเตอร์ ( $v$ ) และค่าไอเกนแวลู ( $\mu$ ) ของเมทริกความแปรปรวนร่วมเนื่องจากเมทริก  $C$  มีขนาด  $N^2 \times N^2$  มิติ นำมาสร้างเป็นไอเกนเวกเตอร์ ขนาด  $N^2$  และค่าไอเกน ขนาด  $N^2$  ด้วยรูปภาพ ขนาด  $80 \times 80$  พิกเซล จะไม่สามารถกระทำได้ซึ่ง  $N^2$  จะได้เมทริกที่มีขนาดใหญ่มาก ซึ่งวิธี PCA เป็นการลดมิติข้อมูลลง จึงทำให้ได้ผลลัพธ์มีขนาดมากเกินไป และอาจจะมียไอเกนเวกเตอร์บางค่าที่สัมพันธ์กับค่าไอเกนที่มีค่าเป็นศูนย์ จึงไม่นำมาเป็นข้อมูลในการพิจารณาเพื่อแก้ปัญหานี้ Turk and Pentland ได้ทำการปรับปรุงขั้นตอนวิธีโดยการเสนอวิธีใหม่โดยการสร้างเมทริก  $A^T A$  แทนที่จะใช้  $AA^T$  ดังนั้นค่าไอเกนเวกเตอร์ สามารถทำได้ ดังสมการที่ 3.4

$$A^T A v_i = \mu_i v_i \quad (3.4)$$

เมื่อ  $A$  คือ กลุ่มของเมทริกความแปรปรวนร่วม  
 $\mu_i$  คือ ค่า ไอเกน Eigen value  
 $v_i$  คือ ค่า Eigen vector

ขนาดของเมทริกความแปรปรวนร่วม จะมีขนาด  $M \times M$  ซึ่งเป็นเมทริกขนาดใหญ่ จึงแทนค่าไอเกนเป็นขนาด  $N^2$  ซึ่งจะได้สมการใหม่ ดังสมการที่ 3.5

$$AA^T A v_i = \mu_i A v_i \quad (3.5)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในขั้นตอนนี้จะใช้วิเคราะห์องค์ประกอบหลักโดยใช้หน้าไอเกนที่มีนัยสำคัญที่สุด ดังนั้นเพื่อลดพื้นที่ในการเก็บข้อมูลโดยการเลือกเอาเฉพาะใบหน้าไอเกนแรกๆ มาเก็บไว้เท่านั้น

ขั้นตอนที่ 5 วิธีการรู้จำ เริ่มจากการนำภาพที่ต้องการทดสอบการรู้จำ  $T$  จึงนำภาพใบหน้าไอเกนไปทำการหาค่าน้ำหนัก โดยนำภาพใบหน้าไอเกนที่เลือกคูณกับภาพใบหน้าซึ่งเป็นการฉายภาพ (Projection) ภาพใบหน้ากับภาพใบหน้าไอเกนใน Face space ดังสมการที่ 3.6

$$W_k = \mu_k^T (T - \psi) \quad (3.6)$$

จากสมการที่ 3.6 ค่า  $k$  คือจำนวนใบหน้าทีเลือก ซึ่งเป็นนำภาพใบหน้ามาคำนวณเพื่อหาค่าน้ำหนักที่จะนำไปเป็นตัวแทนจำแนกหรือรู้จำบุคคลต่อไป ซึ่งจะได้  $\Omega^T = [w_1, w_2, w_3, \dots, w_M]$

ขั้นตอนที่ 6 คำนวณหาค่าระยะห่างระหว่างภาพใบหน้าทีนำมาทดสอบกับภาพใบหน้าไอเกน (Euclidean distance) ดังสมการที่ 3.7

$$\varepsilon_k = \|\Omega - \Omega_k\| \quad (3.7)$$

เมื่อ  $\varepsilon_k$  คือ ค่าระยะห่าง  
 $\Omega$  คือ ใบหน้าไอเกน  
 $\Omega_k$  คือ ใบหน้าทดสอบ

จากสมการ 7 ค่า  $\Omega_k$  คือระยะห่างของใบหน้าลำดับที่  $k$  ที่สามารถค้นพบ หลังจากมีการคำนวณภาพระยะห่างภาพใบหน้ากลุ่มเรียนรู้ กับภาพใบหน้าทีนำมาทดสอบแล้ว นำค่าระยะห่างมาวิเคราะห์ โดยพิจารณาว่าข้อมูลของภาพทีนำมาทดสอบมีค่าใกล้เคียงกับภาพใบหน้าในกลุ่มเรียนรู้ภาพใด มีระยะห่างน้อยที่สุด สรุปผลออกมาได้ว่า ภาพทีนำมาทดสอบนั้นอยู่กลุ่มเดียวกับภาพในกลุ่มเรียนรู้ภาพนั้น

### 3.1.3 การแยกองค์ประกอบหลักชนิดสองมิติ (2DPCA)

แนวคิดของวิธี 2DPCA เป็นเทคนิคที่สามารถนำเอาภาพสองมิติมาคำนวณได้โดยตรง ซึ่งมีข้อดี คือ การคำนวณหาค่าโควาเรียนซ์เมทริก (Covariance Matrix) สามารถทำได้โดยตรง ส่งผลให้เวลาในการคำนวณหาค่าไอเกนเวกเตอร์ (Eigenvector) มีค่าน้อย หลักการคำนวณคือเริ่มจากการคำนวณหาโควาเรียนซ์เมทริก (Covariance Matrix) จากภาพฝึกสอนทั้งหมด แล้วมาทำการหาแกนการฉายที่เหมาะสม แล้วจึงมาฉายเป็นลักษณะเด่นของภาพใบหน้า เพื่อนำไปใช้ในการหาการรู้จำการหาค่าโควาเรียนซ์ (Covariance Value) จากภาพฝึกสอนทั้งหมด กำหนดให้ชุดข้อมูลภาพฝึกสอนทั้งหมด  $M$  ภาพ เมทริก  $A_j (j= 1,2, \dots, M)$  และภาพเฉลี่ยของภาพฝึกสอนทั้งหมดเขียนแทนด้วย  $\bar{A}$  ดังนั้นสามารถหา  $G_t$  ได้จากสมการ

$$G_t = \frac{1}{M} \sum (A_j - \bar{A})^T (A_j - \bar{A}) \quad (3.8)$$

และการหาแกนการฉายที่เหมาะสมสามารถหาได้โดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$J(X) = X^T G_t X \quad (3.9)$$

การสกัดลักษณะเด่น (Feature Extraction) สมการคือ

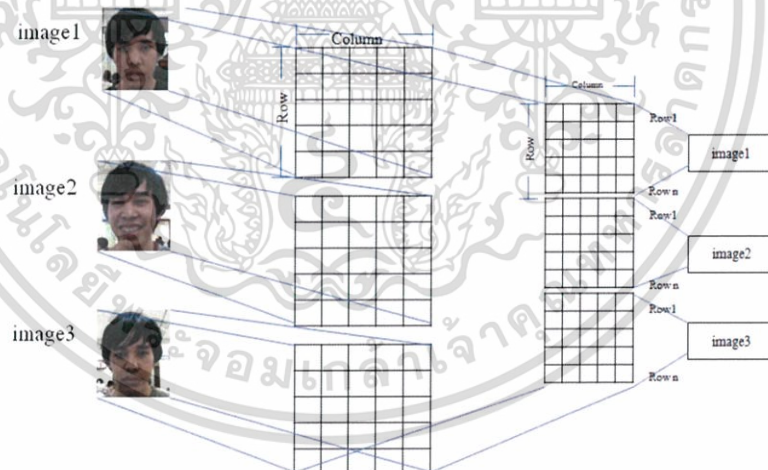
$$Y = AX \quad (3.10)$$

โดย  $Y$  คือเวกเตอร์จากการฉายภาพ  $A$  ลงบนแกนการฉายที่เหมาะสม  $X$  ขนาด  $m$  มิติ ซึ่งเรียกเวกเตอร์  $Y$  เป็นเวกเตอร์ลักษณะเด่น (Feature Vector) ของภาพ  $A$

การหาการรู้จำ สามารถทำได้โดยนำ เอลักษณะเด่นมาคำนวณหาระยะทางยูคลิเดียน (Euclidean Distance) ระหว่างระยะห่างของสองเมทริกลักษณะเด่น (Feature Matrix)  $Y_i = [y_1^{(i)}, y_2^{(i)}, \dots, y_d^{(i)}]$  กับ  $Y_j = [y_1^{(j)}, y_2^{(j)}, \dots, y_d^{(j)}]$  สามารถหาได้จาก

$$d(Y_i, Y_j) = \sum_{p=1}^d \|y_p^{(i)} - y_p^{(j)}\|_2 \quad (3.11)$$

โดยที่  $\|y_p^{(i)} - y_p^{(j)}\|_2$  ระยะทางยูคลิเดียน (Euclidean Distance) ระหว่างสองเมทริกลักษณะเด่น (Feature Matrix) โดยภาพทดสอบใดได้ระยะทางเมื่อวัดกับทุกภาพฝึกสอนแล้วมีค่าน้อยที่สุดถือว่าเป็นภาพของกลุ่มภาพนั้น



รูปที่ 3.3 แสดงหลักการ 2DPCA

### 3.1.4 หลักการสร้างภาพใบหน้าไอเกน (Eigenface)

Eigenface เป็นอัลกอริธึมหนึ่งที่ใช้ในการรู้จำใบหน้าและการระบุตัวบุคคล ซึ่งมีพื้นฐานอยู่บน หลักการของการวิเคราะห์องค์ประกอบหลัก (Principal Component Analysis: PCA) โดยจะแทนภาพ ใบหน้าด้วยสมการเชิงเส้นของเวกเตอร์ที่ตั้งฉากกัน โดยการหา Eigen Vector ของเมทริกความแปรปรวนร่วม (covariance matrix) จากรูปภาพในฐานข้อมูลทั้งหมด โดยนำภาพ

ในฐานข้อมูลแต่ละภาพมาเก็บข้อมูลแบบเวกเตอร์ แล้วจึงนำมาหาเวกเตอร์ค่าเฉลี่ยของภาพ ซึ่งภาพ  
เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น เมื่อผู้ผู้เห็นแปะชื่อหรือชื่อในการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบหน้าที่ต้องการทำเป็นแบบจำลองต้องเป็นระดับเทา (Gray Level) เนื่องจากใช้หน่วยความจำในการเก็บข้อมูลน้อยกว่าภาพสี ทำให้ประหยัดหน่วยความจำเมื่อเทียบกับการเก็บข้อมูลของภาพสี ทำให้ภาพระดับ Gray Level ใช้เวลาในการประมวลผลที่เร็วกว่าภาพสี จึงต้องทำการแปลงภาพที่ใช้เป็นภาพรูปแบบสีเทา

ภาพใบหน้าที่ใช้ในการสร้างแบบจำลองเป็นภาพระดับเทา (Gray Level) มีขนาดกว้าง 64 จุดภาพ สูง 64 จุดภาพ แต่ละภาพจัดให้อยู่ในรูปแบบของเวกเตอร์ขนาด  $64 \times 64 = 4,096$  มิติ นำภาพทุกภาพมาจัดให้อยู่ในรูปแบบของเมทริกจะได้เมทริก  $A_{i,j}$  ซึ่งหมายถึงข้อมูลภาพที่  $i$  มิติที่  $j$

ขั้นตอนการหา Eigenface แสดงได้ ดังนี้

1) แปลงภาพแต่ละภาพให้จัดอยู่ในรูปแบบของเวกเตอร์ นำภาพทุกภาพมาจัดให้อยู่ในรูปแบบเมทริก จะได้เมทริก  $A_{i,j}$  ซึ่งหมายถึงข้อมูลภาพที่  $i$  มิติที่  $j$

$$\begin{bmatrix} A_{11} & A_{12} & \dots & A_{1m} \\ A_{21} & A_{22} & \dots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \dots & A_{nm} \end{bmatrix} \quad \begin{matrix} n = 20 \\ m = 4096 \end{matrix}$$

2) นำเมทริก  $A_{i,j}$  มาทำการหาค่าเฉลี่ยของแต่ละคอลัมน์ ซึ่งก็คือหาค่าเฉลี่ยของทุกภาพ ใบหน้าในแต่ละมิติ

$$mean_j = \frac{1}{n} \sum_{i=1}^n A_{i,j} \quad ; \quad 1 \leq j \leq m \quad (3.12)$$

3) หาค่าเบี่ยงเบนมาตรฐาน (Standard Deviation) ของข้อมูลภาพทุกๆ ภาพในแต่ละมิติ

$$C_{i,j} = A_{i,j} - mean_j \quad ; \quad 1 \leq i \leq n; 1 \leq j \leq m \quad (3.13)$$

4) นำค่าเบี่ยงเบนมาตรฐานมาสร้างเมทริกของความแปรปรวนร่วม (Covariance Matrix) เพื่อใช้ในการคำนวณหาค่าไอเกนและเวกเตอร์ไอเกนต่อไป

$$S_{j_2, j_1} = S_{j_1, j_2} = \sum_{i=1}^n C_{i, j_1} C_{i, j_2} \quad ; \quad 1 \leq j_1 \leq m; j_1 \leq j_2 \leq m \quad (3.14)$$

Covariance matrix  $s = c \times c^T$

5) นำเมทริกของความแปรปรวนร่วมที่ได้มาคำนวณค่าไอเกนและเวกเตอร์ไอเกนโดยค่าไอเกน (Eigen Values)

$$\lambda = ss^T \quad (3.15)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเวกเตอร์ไอเกน (Eigen Vector)

$$e = \frac{CS}{\sqrt{\lambda}} \quad (3.16)$$

6) ผลลัพธ์จากการคำนวณจะได้ค่าไอเกน ( $\lambda$ ) และเวกเตอร์ไอเกน ( $e$ ) ซึ่งข้อมูลทั้งสองจะมีความสมนัย (Correspondence) ซึ่งกันและกัน โดย  $V = [v_1 v_2 v_3 \dots v_n]$  เป็นเมทริกซ์ที่แต่ละหลักเป็น Eigenvector ของ  $S$  ซึ่ง Eigenvector  $v_i$  เป็นฐานหลัก (basis) ของ row space แต่สิ่งที่ต้องการคือ basis ของ column space เนื่องจากภาพที่เรานำมาสร้างเป็นเมทริกซ์  $A$  แต่ละภาพ ถูกวางอยู่ในแนว หลัก (column) จึงทำการหาฐานหลักของ column space ของเมทริกซ์  $A$  จากสมการ

$$u_j = \frac{1}{\lambda_j} A v_j \quad (3.17)$$

7) เมื่อ  $\lambda_j$  คือ Eigenvalue ที่สัมพันธ์กับ  $v_j$  Eigenvector โดยจะเรียกเวกเตอร์  $u_j$  ว่า Eigenface เนื่องจากหลัก (column) ของเมทริกซ์  $A$  ประกอบด้วยเวกเตอร์ของรูปใบหน้า เพราะฉะนั้น  $u_1, u_2, \dots, u_n$  เป็น orthonormal basis ของ column space ของเมทริกซ์  $A$  หรือ ปริภูมิภาพใบหน้า (Face space) ดังนั้น จะได้ว่า

$$A_{i,j} - \text{mean } j_1 = w_{(i,1)} u_1 + w_{(i,2)} u_2 + \dots + w_{(i,n)} u_n \quad (3.18)$$

โดยที่

$$w_{(i,j)} = \langle A_{i,j} - \text{mean } j, u_j \rangle$$

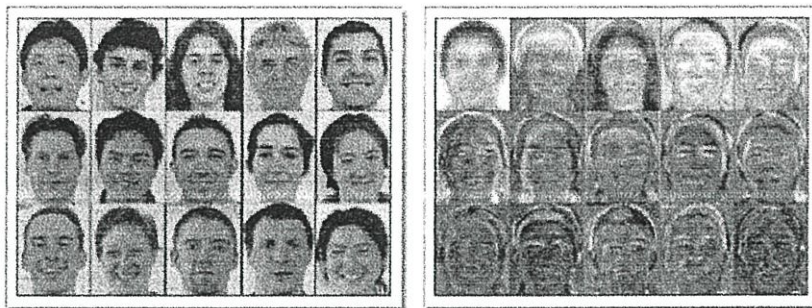
8) จะได้  $w_{(i,j)}$  เป็นตัวเลขที่เกิดจากภาพในฐานข้อมูลหมายเลขที่  $i$  กับ Eigenface ที่  $j$  ทำให้ได้ว่า  $A_{i,j}$  สามารถเขียนแทนด้วยค่าน้ำหนัก

$$\Omega_i = \begin{bmatrix} w_{i,1} \\ w_{i,2} \\ \vdots \\ w_{i,n} \end{bmatrix} \quad (3.19)$$

9) การรู้จำใบหน้าทำได้โดยนำค่าน้ำหนักมาเปรียบเทียบกับค่า Euclidean norm ซึ่ง Euclidean เป็นค่าที่ใช้ในการเปรียบเทียบระยะทางระหว่างจุดดังกล่าวกับจุดอื่นๆ ที่มีอยู่ในฐานข้อมูล และหาค่าที่น้อยที่สุดถึงจะระบุว่าเป็นบุคคลนั้น

$$\Sigma = \min (\|\Omega_i - \Omega_{test}\|) \quad (3.20)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 ใบหน้าก่อนหาค่า Eigenface และภาพใบหน้าไอเกน

### 3.2 Linear Discriminate Analysis (LDA)

Linear Discriminate Analysis (LDA) การวิเคราะห์จำแนกประเภทเชิงเส้น [10] เป็นเทคนิคที่ใช้สำหรับการเรียนรู้ แบบมีผู้สอนใช้เทคนิคการหากลุ่มของปริมาณ (Base vector) โดย  $W_k$  เป็นเวกเตอร์ของกลุ่มตัวอย่าง สามารถแสดงสมการ  $W_k$  ได้ดังนี้

$$W_{opt} = \frac{\arg \max_w \frac{|w' s_c w|}{|w' s_v w|}}{w} \quad (3.21)$$

$$= [w_1, w_2, \dots, w_L]$$

โดยที่  $\{W_x | 1 \leq K \leq L\}$  ให้  $L$  เป็นมิติของชุดข้อมูล  $s_c$  และ  $s_v$  เป็นเมตริกที่อยู่ใน class สามารถคำนวณหาเมตริก  $s_c$  และ  $s_v$  ได้จากสมการ 3.22 และสมการ 3.23

$$S_c = \sum_{k=1}^a M_x (U_k - U)(U_k - U)^T \quad (3.22)$$

$$S_v = \sum_{k=1}^a \sum_{x_u=x_k} M_x (U_k - U)(U_k - U)^T \quad (3.23)$$

เมื่อ  $a$  แทนจำนวน class และ  $X \in R^n$  เป็นข้อมูล

$X_k$  แทน ข้อมูลทดสอบของ class  $k$

$\mu_k$  แทนข้อมูลทั้งหมดของ class  $k$

$M_k$  คือ จำนวนข้อมูลที่ใช้ทดสอบใน class  $k$  ถ้า  $s_v$  เป็น non-singular ที่ใช้ค้นหา  $w_k$  จากสมการ 3.12 ซึ่ง  $L$  แรกจะมีขนาดใหญ่กว่าไอเกน  $\{W_x | 1 \leq K \leq L\}$

### 3.3 ICA (Independent Components Analysis)

วิธีแยกองค์ประกอบอิสระที่เรียกว่าอินดิเพนเดนทคอมโพเนนท์อานาลิซิส (Independent Components Analysis, ICA) โดยเทคนิคนี้ใช้การเปรียบเทียบภาพใบหน้า และขวาหรือจากจุดที่แตกต่างกันด้วยวิธีการทางคณิตศาสตร์สถิติ เพื่อแยกภาพใบหน้าเหล่านี้ออกมา เป็นหลายๆ องค์ประกอบ (Component) โดยแต่ละองค์ประกอบที่แยกได้จะเป็นสัญญาณที่มีความเป็นอิสระต่อกัน (Independent)

แนวคิดของการแยกองค์ประกอบของภาพใบหน้าที่รวมกันอยู่สมมุติว่าในห้องๆ หนึ่งมีคนสองคน และมีกล้องสองตัวที่ตั้งอยู่คนละที่กัน และกล้องทั้งคู่จะผลิตภาพใบหน้า ที่เกิดจากภาพใบหน้าของคนทั้งสองคน ถ้าแทนภาพใบหน้าทีกล้องตัวแรกผลิตขึ้นมาด้วย  $X_1(t)$  และแทนภาพใบหน้าของกล้องตัวที่สองด้วย  $X_2(t)$  โดยที่  $X_1$  และ  $X_2$  คือขนาดของภาพใบหน้า และ  $t$  คือเวลา และแทนภาพใบหน้าของทั้งสองคน ณ ตำแหน่งแต่ละคนยืนอยู่ด้วย  $S_1(t)$  และ  $S_2(t)$  แล้ว สามารถแสดงความสัมพันธ์ของภาพใบหน้า และภาพใบหน้าจากกล้องได้ด้วย สมการดังต่อไปนี้

$$X_1(t) = a_{11}S_1 + a_{12}S_2 \quad (3.24)$$

$$X_2(t) = a_{21}S_1 + a_{22}S_2 \quad (3.25)$$

โดยที่  $a_{11}$   $a_{12}$   $a_{21}$  และ  $a_{22}$  คือพารามิเตอร์ที่มีค่าขึ้นอยู่กับระยะห่างระหว่างกล้องและภาพใบหน้า จะเป็นประโยชน์อย่างมากถ้าหากสามารถสร้าง (Reconstruction) ภาพใบหน้าของคนทั้งสอง ( $S_1$  และ  $S_2$ ) กลับคืนมาจากภาพใบหน้า ที่ได้จากกล้องทั้งสองตัว ( $X_1$  และ  $X_2$ ) แต่ปัญหาคือภาพใบหน้า ที่ได้จากกล้องเกิดจากการผสมกันของภาพใบหน้าสองแหล่งที่มีระยะห่างระหว่างต้นกำเนิดและจุดรับต่างกัน ซึ่งปัญหาแบบนี้ถูกเรียกว่า “Cocktail Party Problem”

จากสมการ (3.24) และ (3.25) ถ้าตัดเรื่องเป็นเวลา ( $t$ ) ในสมการออกไป และสมมุติว่ามีจำนวนขององค์ประกอบที่อิสระจากกัน  $n$  องค์ประกอบ และให้  $s_k$  เป็นตัวแปรสุ่ม (Random Variable) จะได้สมการใหม่ดังนี้

$$X_j = a_{j1}S_1 + a_{j2}S_2 + \dots + a_{jn}S_n, \text{ for all } j \quad (3.26)$$

และถ้าเขียนสมการ (3.26) ในรูปของเวกเตอร์เมตริก (Vector-Matrix) จะได้ดังนี้

$$X = AS \quad (3.27)$$

โดยที่  $A$  เป็นเมตริกของสมาชิก  $a_{ij}$  และ  $X$  เป็นแรนดอมเวกเตอร์ (random vector) ที่มีสมาชิกเป็นสัญญาณผสม  $X_1, \dots, X_n$  และ  $S$  เป็นเวกเตอร์ที่มีสมาชิกเป็น  $S_1, \dots, S_n$  โดยทั้งหมดจะเป็นคอลัมน์เวกเตอร์ในแนวตั้ง (Column Vector) สมการ (3.27) จะถูกเรียกว่าโมเดลไอซีเอทฤษฎีของไอซีเอคือการสังเกต  $X$  ภายใต้สมมุติฐานที่เป็นไปได้เพื่อประมาณค่าของ  $A$  และ  $S$  โดยสมมุติฐานแรกคือสมมุติว่าแต่ละองค์ประกอบ  $S_j$  มีความเป็นอิสระต่อกันในทางสถิติ (Statically

Independent) ซึ่งจะอธิบายในหัวข้อต่อไปว่าจากสมมุติฐานนี้สามารถจะประมาณค่า A ได้อย่างไร และถ้าให้ส่วนกลับ (Inverse) ของ A เท่ากับ W ก็จะได้สมการนี้

$$S = WX \quad (3.28)$$

### 3.4 การแยกคุณลักษณะของใบหน้าทั้งใบหน้าและเฉพาะส่วนโดยใช้แบบจำลองเชิงเรขาคณิต

แบบจำลองใบหน้าเชิงเรขาคณิตมาประยุกต์ใช้เพื่อคำนวณหาตำแหน่งของจมูกและปาก โดยสร้างความสัมพันธ์กับค่าที่ได้จากตำแหน่งพิกัดของตาทั้งสองข้างตามขั้นตอนดังนี้

- 1) หาจุดกึ่งกลางระหว่างตำแหน่งตาทั้งสองข้าง
- 2) สร้างวงกลมโดยมีรัศมีเป็นครึ่งหนึ่งของระยะห่างระหว่างตาทั้งสองข้าง จะได้จุดตัดบนแกน Y ของวงกลม กำหนดให้เป็นตำแหน่งของจมูกซึ่งมีโอกาสเป็นได้ทั้งจุดบนและจุดล่าง แต่ในกรณีทดลองนี้ใช้เป็นภาพแบบหน้าตรงจึงใช้จุดด้านล่างเพียงจุดเดียวมาพิจารณา
- 3) สร้างวงรีโดยให้แกนเอกมีความยาวเท่ากับระยะห่างระหว่างตาทั้งสองข้างและแกนโทมีความยาวเป็นครึ่งหนึ่งของระยะห่างของตาทั้งสองข้างหารด้วย 0.6 จะได้จุดตัดบนแกน Y ของวงรี กำหนดให้เป็นตำแหน่งของปาก ซึ่งมีโอกาสเป็นได้ทั้งจุดบนและจุดล่างเช่นเดียวกับจมูก
- 4) ใช้ตำแหน่งพิกัดต่างๆ ที่คำนวณได้เป็นจุดศูนย์กลางในการแยกส่วนประกอบของภาพใบหน้า ดังแสดงในรูปที่ 3.5



รูปที่ 3.5 แสดงตำแหน่งพิกัดต่างๆ ที่ได้จากการคำนวณ

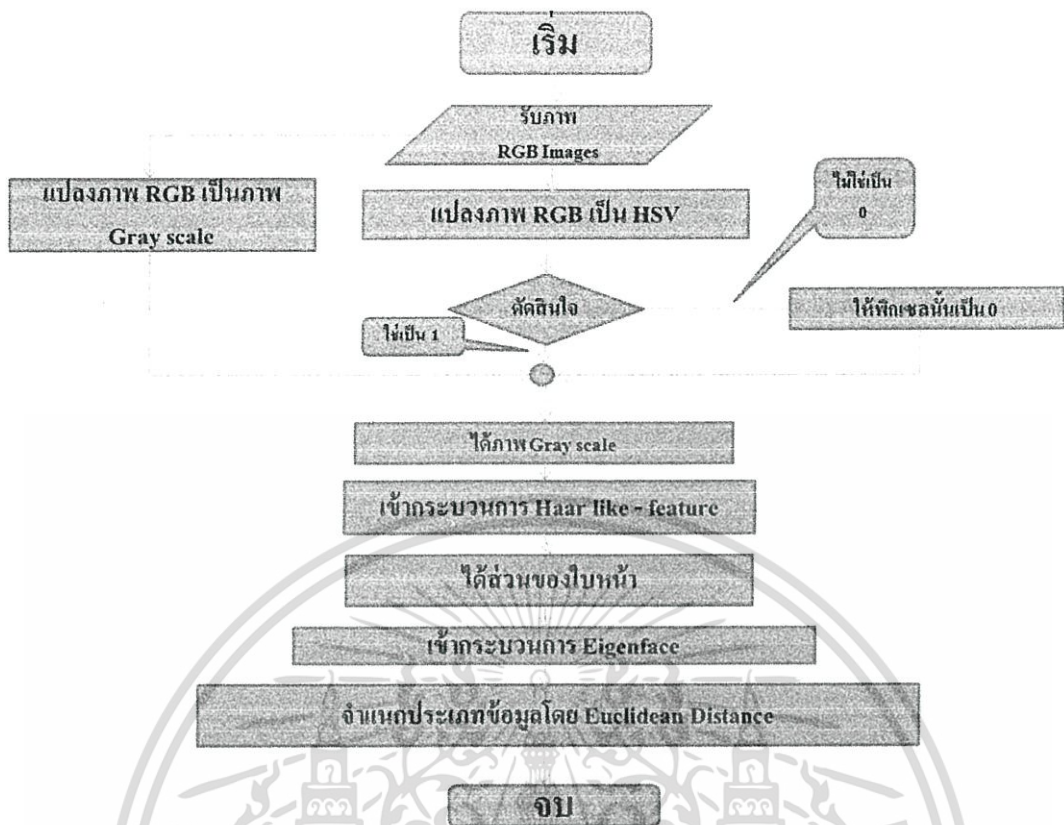
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### การทดลองและผลการทดลอง

จากการอธิบายในส่วนของทฤษฎีและงานวิจัยที่เกี่ยวข้องรวมถึงการดึงคุณลักษณะเด่นของใบหน้า ที่กล่าวในบทที่แล้ว ประกอบเป็นงานวิจัยฉบับนี้โดยจะใช้ภาพใบหน้าผู้ก่อการร้ายในสามจังหวัดชายแดนภาคใต้ที่ได้ก่อเหตุความรุนแรงในพื้นที่แล้วมีการนำรูปมาลงประกาศจับแล้ว นำมาใช้เป็นฐานข้อมูลและเป็นใบหน้าสี่สองมิติ โดยจะทำการตรวจสอบค่าสีจากระบบสี HSV และวิธีการ Haar like feature เพื่อให้ได้ใบหน้าผู้ก่อการร้าย ต่อจากนั้นใช้วิธีการ Eigenface ในการหาคุณลักษณะเด่นบนใบหน้าเพื่อให้ได้ไอเก้นเวกเตอร์ หลังจากได้ไอเก้นเวกเตอร์ของทั้งภาพทดลองและภาพทดสอบทั้งหมดมาหาระยะทางแบบยุคลิด สุดท้ายนำผลการทดสอบมาเปรียบเทียบกับวิธีการอื่นๆ ซึ่งการทดลองต่างๆในบทนี้จะเริ่มจากการทดสอบด้านซอร์ฟแวร์ โดยทดสอบอัลกอริทึมไอเกนเฟสกับรูปแบบใบหน้าในลักษณะที่แตกต่างกันเพื่อจะหาเปอร์เซ็นต์ความถูกต้องในการหาค่าเฉลี่ยไอเกนเวกเตอร์ รวมไปถึงการทดสอบฮาร์ดแวร์ว่าทำงานตรงตามเงื่อนไขที่กำหนดไว้หรือไม่

โดยขั้นตอนของกระบวนการจำแนกภาพใบหน้าผู้ก่อการร้าย จะเริ่มตั้งแต่การรับภาพใบหน้าผู้ก่อการร้ายเป็นภาพ RGB Image ภาพมีขนาดประมาณ  $96 * 96$  Pixel ขั้นตอนต่อไปคือนำภาพมาเข้าการตรวจจับวัตถุโดยการแปลงภาพ RGB เป็น HSV โดยเมื่อทำการตัดสินใจสีผิว ได้ในส่วนของใบหน้าในส่วนที่ต้องการแล้วจะนำมาคูณกับภาพ Gray scale แล้วปรับขนาดภาพเป็นขนาด  $196 * 196$  Pixel ในขั้นตอนนี้จะได้ภาพใบหน้าอย่างคร่าวๆ เป็นภาพในลักษณะ Gray scale หลังจากนั้นจะนำภาพใบหน้าในลักษณะ Gray scale มาเข้ากระบวนการตรวจจับใบหน้าผู้ก่อการร้ายโดยการใช้ Haar like-Features จากนั้นจะได้ภาพในส่วนของใบหน้าในลักษณะ Gray scale ที่ดีขึ้นกว่ากระบวนการที่ผ่านมา แล้วนำมาเข้ากระบวนการในการจดจำใบหน้าโดยวิธีการ Eigenface ทั้งนี้ภาพที่ใช้ในขั้นตอนการเรียนรู้และภาพที่ใช้ในขั้นตอนการทดสอบจะนำมาผ่านกระบวนการข้างต้นทั้งสองส่วนเพื่อจะนำในส่วนของค่า Eigenvector มาเข้าสู่กระบวนการจำแนกประเภทข้อมูลโดยวิธีการ Euclidean Distance โดยมีกระบวนการทั้งหมดในการรู้จำใบหน้าได้แสดงด้วยผังงานรูปที่ 4.1



รูปที่ 4.1 กระบวนการรู้จำใบหน้าผู้ก่อการร้าย

#### 4.1 รายละเอียดของระบบที่ใช้ทดสอบ

ในการทดสอบได้แบ่งรายละเอียดของการทดสอบออกเป็น 2 ส่วนคือ รายละเอียดทางฮาร์ดแวร์และ รายละเอียดทางซอฟต์แวร์ ดังตารางที่ 4.1

ตารางที่ 4.1 รายละเอียดของระบบที่ใช้ทดสอบ

Hardware	Micro-Computer	Processor	Intel Centrino Duo 1.6 GHz
		Memory	1.5 GB of RAM
	Database Input	ภาพ	100 ตัวอย่าง
Software	Operating System	Windows	Version 2007
	Program Simulation	Matlab	Version 10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

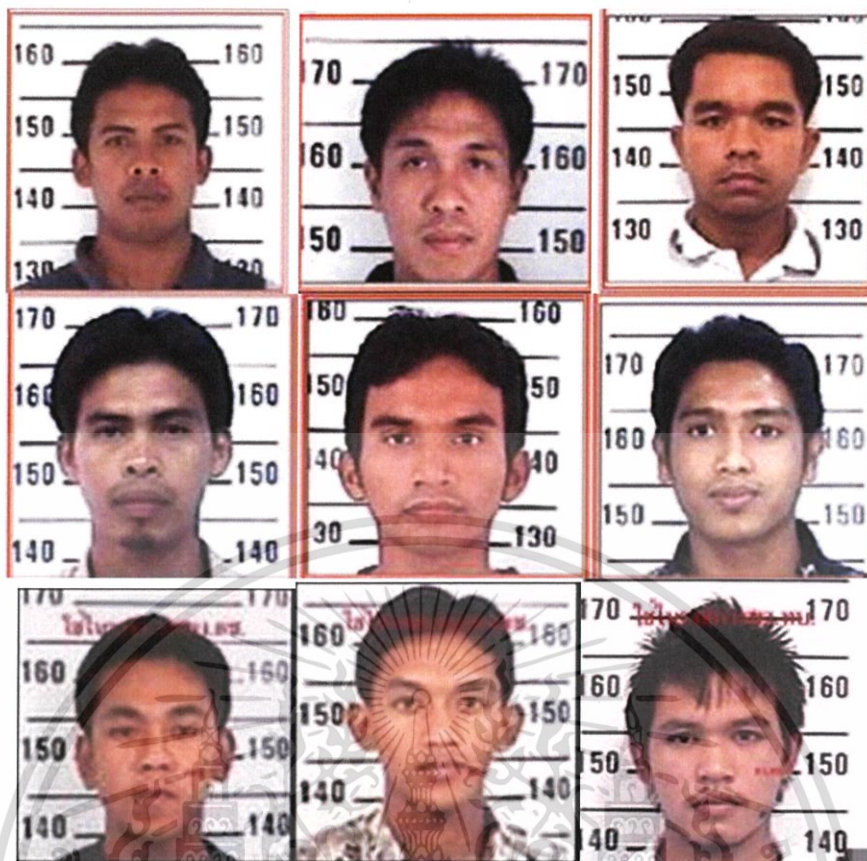
#### 4.1.1 ตัวอย่างภาพที่ใช้ในงานวิจัย

ตัวอย่างภาพที่ใช้ในงานวิจัยทั้งหมดเป็นภาพที่มีการเผยแพร่แล้วจากทางราชการ โดยเป็นภาพแบบบิตแมปประเภทภาพสี RGB Image รูปแบบของไฟล์ภาพเป็นแบบนามสกุล JPEG ขนาดไฟล์ภาพในแต่ละรูปบุคคลมีขนาดประมาณ 2.7 ถึง 2.8 KB เป็นภาพที่มีขนาดระหว่าง 96 \* 96 Pixel ก่อนเข้าทำการประมวลผลในโปรแกรม



รูปที่ 4.2 รูปตัวอย่างผู้ก่อการร้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 (ต่อ)

#### 4.1.2 ขั้นตอนการเก็บและเตรียมภาพใบหน้า

ขั้นตอนในการเก็บและเตรียมภาพนั้นจะแยกออกเป็น 2 ส่วนคือ ภาพที่ใช้ในขั้นตอนการเรียนรู้และในส่วนของภาพที่ใช้ในการทดสอบ โดยภาพที่ได้มาจะเป็นภาพใบหน้าผู้ก่อการร้าย ใบหน้าเดี่ยวอันมีรายละเอียดดังต่อไปนี้

- ภาพสำหรับใช้ในขั้นตอนการเรียนรู้ เป็นภาพแบบบิตแมปประเภทภาพสี RGB Image รูปแบบของไฟล์ภาพเป็นแบบนามสกุล JPEG ขนาดไฟล์ภาพในแต่ละรูปบุคคลมีขนาดประมาณ 2.7 ถึง 2.8 KB เป็นภาพที่มีขนาดระหว่าง  $96 * 96$  Pixel จำนวน 100 ภาพ หลังผ่านขั้นตอนการแปลงภาพ RGB เป็นภาพ HSV แล้วทำกลับมาเป็นภาพแบบ Gray scale จะนำมาปรับเป็นภาพขนาด  $196 * 196$  Pixel ขนาดประมาณ 3.14 KB ซึ่งเป็นขนาดภาพที่มีความพอดีในการประมวลผล จากการประมาณค่าด้วยสายตาของผู้ทำวิจัย โดยทำการสุ่มภาพมาทำการขยายภาพ หดภาพ หาขนาดภาพที่มีความเหมาะสม ได้ภาพที่มีความชัดเจนสามารถมองเห็นลักษณะรายละเอียดของภาพได้

- ภาพที่ใช้ในขั้นตอนการทดสอบ เป็นภาพแบบบิตแมปประเภทภาพสี RGB Image รูปแบบของไฟล์ภาพเป็นแบบนามสกุล JPEG โดยแบ่งออกเป็น 2 ส่วนคือ ภาพแบบปกติที่สุ่มออกมาจากภาพของฐานข้อมูลจำนวน 50 ภาพจาก 100 ภาพ และภาพที่สุ่มออกมาจากฐานข้อมูลจำนวน 50 ภาพจาก 100 ภาพ แล้วใส่สัญญาณรบกวนจำนวน 2 ชนิด ชนิดแรกคือใส่สัญญาณรบกวนแบบ Salt and pepper noise จำนวน 20% ในแต่ละภาพจำนวน 50 ภาพ และชนิดที่สองใส่สัญญาณรบกวน

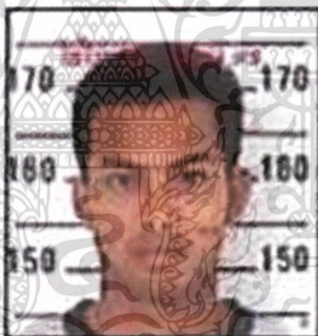
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รบกวนแบบ Gaussian Noise โดยจะใส่สัญญาณรบกวน 0.2 และค่าความเบี่ยงเบน 0.01 ในแต่ละภาพจำนวน 50 ภาพ รวมเป็น 2 ชุดข้อมูลในการทดสอบ โดยรูปแบบของไฟล์ภาพจะเป็นนามสกุล JPEG ขนาดไฟล์ภาพในแต่ละรูปบุคคลก่อนใส่สัญญาณรบกวนมีขนาดประมาณ 2.7 ถึง 2.8 KB เป็นภาพที่มีขนาดระหว่าง  $96 * 96$  Pixel จำนวนอย่างละ 50 ภาพ หลังผ่านขั้นตอนการแปลงภาพ RGB เป็นภาพ HSV แล้วทำกลับมาเป็นภาพแบบ Gray scale จะนำมาปรับเป็นภาพขนาด  $196 * 196$  Pixel ขนาดประมาณ 3.14 KB โดยจะใส่สัญญาณรบกวนในขั้นตอนการทำงานหลังจากที่ผ่านกระบวนการ Haar like-Features แล้วได้ภาพแบบ Gray Scale รวมเป็นภาพที่ใช้ในการทดสอบแบบปกติ 50 ภาพ และภาพที่ใช้ทดสอบแบบเพิ่มสัญญาณรบกวนแบบ Salt and pepper noise จำนวน 50 ภาพ, ภาพที่ใช้ทดสอบแบบเพิ่มสัญญาณรบกวนแบบ Gaussian Noise จำนวน 50 ภาพ

## 4.2 ขั้นตอนการทำงาน

ขั้นตอนนี้คือกระบวนการจำแนกภาพใบหน้าผู้ก่อการร้าย โดยกระบวนการจะเริ่มตั้งแต่การรับภาพใบหน้าผู้ก่อการร้ายเป็นภาพ RGB Image แล้วดำเนินการในแต่ละขั้นตอน จนสุดท้ายจะได้ในส่วน of ใบหน้าแล้วเข้าสู่กระบวนการจดจำใบหน้า ในส่วนของวิธีการ Eigenface และกระบวนการจำแนกประเภทข้อมูลโดยวิธีการ Euclidean Distance ต่อไป โดยมีกระบวนการดังที่กล่าวไปดังนี้

- 1) รับภาพ RGB Images โดยภาพมีขนาดประมาณ  $96 * 96$  Pixel



รูปที่ 4.3 ใบหน้าผู้ก่อการร้ายรูปที่ 1

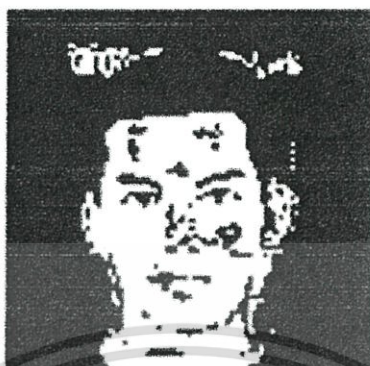
- 2) ดึงส่วนของใบหน้าที่ต้องการโดยวิธีการตรวจจับวัตถุโดยการใช้ระบบสี HSV โดยทำการแปลงสี RGB เป็นสีในระบบ HSV ดังรูปที่ 4.4



รูปที่ 4.4 ใบหน้าผู้ก่อการร้ายรูปที่ 2

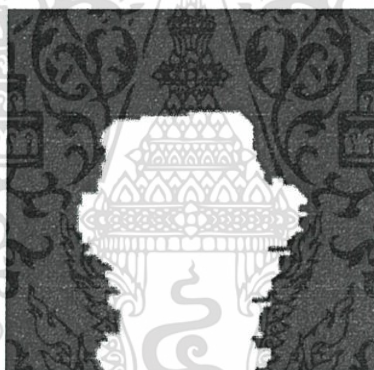
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับวงจำกัดเท่านั้นเพื่อการศึกษาเท่านั้นเมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ทำการดึงใบหน้าอย่างคร่าวๆ โดยการใช่วิธี Threshold โดยที่ค่าสีของบริเวณใบหน้า ผู้ก่อการร้ายอยู่ในช่วง 0.00 – 0.10 ค่าปริมาณความเข้มตัวของสี ของบริเวณใบหน้าผู้ก่อการร้ายอยู่ในช่วง 0.10 – 0.50 ค่าปริมาณความสว่างของสีของบริเวณใบหน้าผู้ก่อการร้ายอยู่ในช่วง 0.50 – 0.85



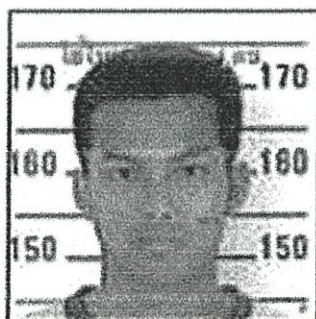
รูปที่ 4.5 ใบหน้าผู้ก่อการร้ายรูปที่ 3

4) ทำการปรับปรุงภาพโดยใช่วิธี Morphology ในการปรับปรุงส่วนที่ขาดหายไปของบริเวณ ใบหน้า



รูปที่ 4.6 ใบหน้าผู้ก่อการร้ายรูปที่ 4

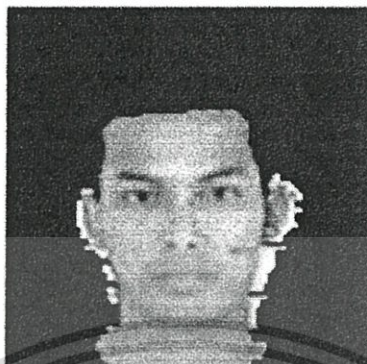
5) นำภาพ Gray Scale มาเตรียมไว้เพื่อกระบวนการในขั้นตอนต่อไป



รูปที่ 4.7 ใบหน้าผู้ก่อการร้ายรูปที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6) ทำการหาภาพจริง โดยนำภาพจากขั้นตอนที่สี่คูณเข้ากับภาพ Gray Scale ในขั้นตอนที่ห้า ทำการปรับขนาดภาพเป็นขนาด  $196 * 196$  Pixel



รูปที่ 4.8 ใบหน้าผู้ก่อการร้ายรูปที่ 6

7) หลังจากได้บริเวณคร่าวๆของใบหน้า ผู้ก่อการร้ายมาแล้วนั้น ในขั้นตอนนี้ทำการตรวจจับใบหน้าผู้ก่อการร้ายโดยการใช้ Haar like-Features ซึ่งเป็นวิธีการตรวจจับและตีความวัตถุภายในภาพ



รูปที่ 4.9 ใบหน้าผู้ก่อการร้ายรูปที่ 7

8) นำภาพใบหน้าที่หาได้นำไปสู่กระบวนการในการจดจำใบหน้า ในส่วนของวิธีการ Eigenface และกระบวนการจำแนกประเภทข้อมูลโดยวิธีการ Euclidean Distance โดยวัดระยะที่สั้นที่สุดในขั้นตอนต่อไป

#### 4.3 การทดลองระบุตัวบุคคลโดยใช้อัลกอริทึมไอเกนเฟซ

ขั้นตอนนี้จะเป็นขั้นตอนที่มีกระบวนการต่อเนื่องมาจากขั้นตอนข้างต้น คือเมื่อได้รับภาพที่ผ่านกระบวนการดึงคุณลักษณะเด่นในส่วนของใบหน้ามาจากขั้นตอน Haar like-Features จะได้ในส่วนของใบหน้าเป็นภาพแบบ Gray Scale มาเข้ากระบวนการรู้จำภาพใบหน้าด้วยวิธีการ Eigenface ในขั้นตอนส่วนนี้จะประกอบไปด้วยข้อมูลสองส่วนคือ ภาพสำหรับใช้ในขั้นตอนการเรียนรู้จำนวน 100 ภาพ และภาพที่ใช้ในขั้นตอนการทดสอบจำนวน 50 ภาพ โดยนำข้อมูลทั้งสองส่วนมาผ่านขั้นตอนทั้งหมด และเข้าสู่กระบวนการ Eigenface ทำการเก็บค่าไอเกนเวกเตอร์ของข้อมูลภาพสำหรับใช้ในขั้นตอนการเรียนรู้และภาพที่ใช้ในขั้นตอนการทดสอบเอาไว้ สุดท้ายนำค่าไอเกนเวกเตอร์ของทั้งสอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลมาทำการจำแนกประเภทข้อมูลโดยวิธีการ Euclidean Distance โดยวัดระยะที่น้อยที่สุด ในส่วนของการหาอัตราความถูกต้องหาได้จากสมการดังต่อไปนี้

$$\text{Accuracy}(\%) = \left( \frac{\text{In-Out}}{\text{Total}} \right) \times 100 \quad (4.1)$$

โดย In คือจำนวนภาพที่นำเข้ากระบวนการทดสอบ, Out คือจำนวนภาพที่ไม่สามารถรู้จำได้, Total คือภาพทั้งหมดที่ใช้ในกระบวนการรู้จำภาพใบหน้า

#### 4.3.1 ทดสอบโดยใช้ใบหน้าผู้ทดสอบเป็นภาพต้นฉบับ

ในการทดลองนี้จะนำภาพหน้าตรงของบุคคลในฐานข้อมูลมาทำการทดลองทั้งหมด คือจำนวน 100 ภาพ ข้อมูลทดสอบมีภาพหน้าใบหน้า 50 ภาพ โดยภาพที่ใช้ในการทดลองมีขอบเขต ดังนี้

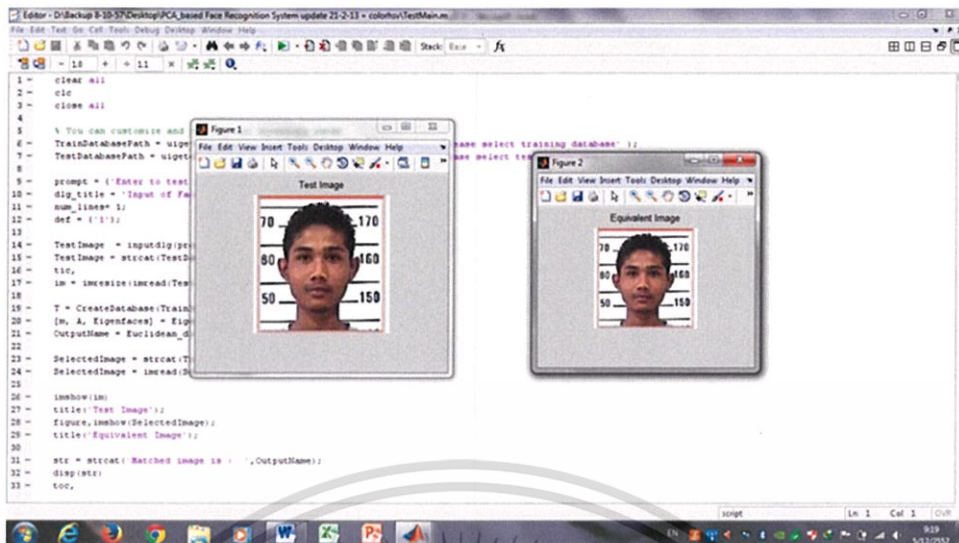
- ภาพที่ใช้ในการทดลองเป็นภาพนิ่งและเป็นภาพสี
- ภาพที่ใช้ในการทดลองเป็นภาพที่มีพื้นหลังไม่ซับซ้อน
- ภาพที่ใช้เป็นภาพหน้าตรง และไม่มีการแสดงอารมณ์บนใบหน้า เช่น ยิ้ม หัวเราะ พุด เป็นต้น
- ภาพที่ใช้เป็นภาพหน้าถ่ายหน้าตรง

ทำการทดลองเปรียบเทียบกับใบหน้าผู้ทดสอบในลักษณะการถ่ายภาพที่เป็นใบหน้าตรงและทำการทดลองทั้งหมด 5 ครั้ง โดยในแต่ละครั้งจะทำการสุ่มภาพใบหน้าจากภาพต้นฉบับจำนวน 50 ภาพ มาทำการรู้จำในโปรแกรม

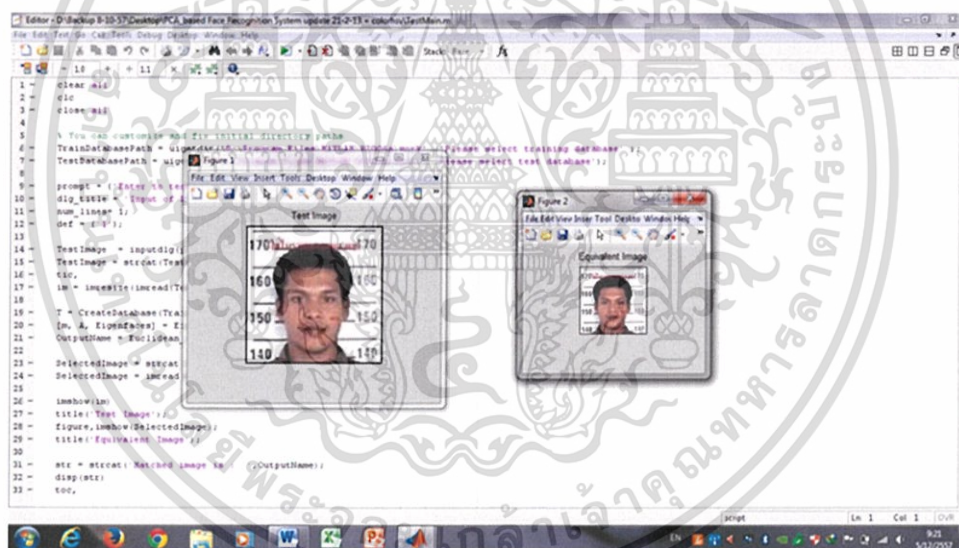
ตารางที่ 4.2 แสดงผลการทดลองการระบุตัวบุคคลโดยใช้ภาพต้นฉบับ

ครั้งที่	จำนวนภาพที่ทดลอง	จำนวนภาพที่ระบุตัวบุคคลได้ถูกต้อง	% ความถูกต้อง	% ความถูกต้องเฉลี่ย
1	100	50	100	100
2	100	50	100	
3	100	50	100	
4	100	50	100	
5	100	50	100	

จากการทดลอง จะเห็นได้ว่าเมื่อนำภาพใบหน้าจากฐานข้อมูลมาทดสอบกับใบหน้าผู้ทดสอบในลักษณะใบหน้าตรงนั้นสามารถระบุตัวบุคคลได้ 100%

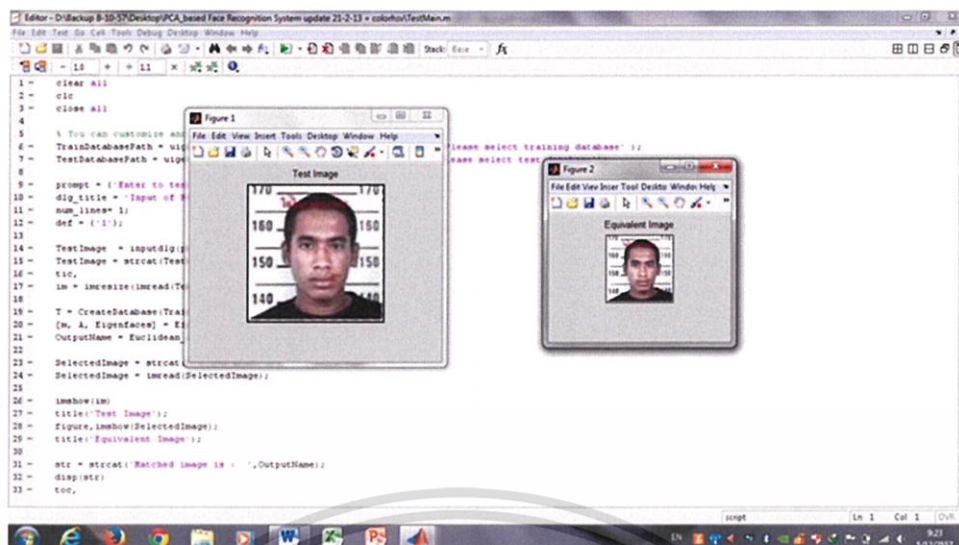


รูปที่ 4.10 ตัวอย่างการทดสอบผู้ก่อการร้ายคนที่ 19 ของโปรแกรม



รูปที่ 4.11 ตัวอย่างการทดสอบผู้ก่อการร้ายคนที่ 22 ของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 ตัวอย่างการทดสอบผู้ก่อการร้ายคนที่ 38 ของโปรแกรม

#### 4.3.2 ทดสอบโดยใช้ใบหน้าผู้ทดสอบเป็นภาพใบหน้าที่มีสัญญาณรบกวน

ในขั้นตอนนี้จะเป็นการเตรียมข้อมูลภาพที่ใช้ในการทดสอบ โดยจะใส่สัญญาณรบกวนลงในภาพเมื่อมาถึงขั้นตอนในส่วนที่ผ่านกระบวนการแปลงภาพ RGB เป็น HSV แล้วได้ภาพใบหน้าคร่าวๆในลักษณะ Gray scale แล้วจะนำภาพส่วนที่ได้มาเข้าวิธีการตรวจจับและตีความวัตถุภายในภาพด้วยวิธี Haar-like-Features หลังจากนั้นจะได้ในส่วนของใบหน้าเพียงอย่างเดียวในลักษณะ Gray scale แล้วจึงนำมาเพิ่มสัญญาณรบกวนภายในภาพก่อนจะเข้าสู่กระบวนการจดจำภาพใบหน้าด้วยวิธีการ Eigenface และ Euclidean Distance ต่อไป

##### 4.3.2.1 สัญญาณรบกวนใช้แบบ Salt and pepper noise

สัญญาณรบกวนใช้แบบ Salt and pepper noise ใส่เป็น 20% ของภาพทั้งหมดที่นำมาทำการทดสอบจำนวน 50 ภาพ ในการทดลองนี้จะนำภาพหน้าตรงของบุคคลในฐานข้อมูลมาทำการทดลองทั้งหมด คือจำนวน 100 ภาพ นำภาพที่ใส่สัญญาณรบกวน Salt and pepper noise จำนวน 50 ภาพ เข้าทำการทดสอบกับภาพในฐานข้อมูล เพื่อเปรียบเทียบกับใบหน้าผู้ทดสอบและทำการทดลองทั้งหมด 5 ครั้ง

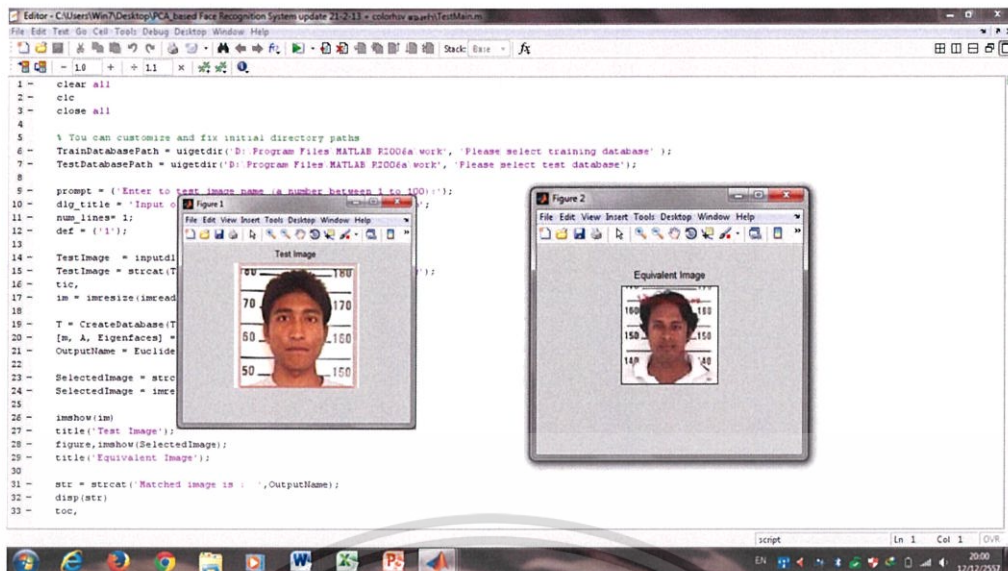


รูปที่ 4.13 รูปผู้ก่อการร้ายที่ใส่สัญญาณรบกวน Salt and pepper noise

ตารางที่ 4.3 แสดงผลการทดลองการระบุตัวบุคคลโดยใช้ภาพใบหน้าที่มีสัญญาณรบกวน Salt and pepper noise ของผู้ทดสอบ

ครั้งที่	จำนวนภาพที่ทดลอง	จำนวนภาพที่ระบุตัวบุคคลได้ถูกต้อง	%ความถูกต้อง	%ความถูกต้องเฉลี่ย
1	100	50	100	98.80
2	100	49	98	
3	100	49	98	
4	100	50	100	
5	100	49	98	

จากการทดลอง จะเห็นได้ว่าเมื่อนำภาพใบหน้าจากฐานข้อมูลมาทดสอบกับใบหน้าผู้ทดสอบ ในลักษณะภาพใบหน้าที่มีสัญญาณรบกวนนั้นสามารถระบุตัวบุคคลได้ดี แต่ก็พบความผิดพลาดอยู่บ้างดังรูปที่ 4.14



รูปที่ 4.14 รูปผู้ก่อการร้ายที่ไม่สามารถระบุตัวตนได้ถูกต้องจากการเพิ่มสัญญาณรบกวนแบบ Salt and pepper noise

### 4.3.2.1 สัญญาณรบกวนใช้แบบ Gaussian Noise

ซึ่งสัญญาณรบกวนใช้แบบ Gaussian Noise ซึ่งจะเป็นสัญญาณรบกวนที่เป็นอิสระในแต่ละพิกเซลของภาพ และเป็นอิสระจากความเข้มของสัญญาณภาพ โดยจะใส่สัญญาณรบกวน 0.2 และค่าความเบี่ยงเบน 0.01 ของภาพทั้งหมดที่นำมาใช้ในการทดสอบจำนวน 50 ภาพ ในการทดลองนี้จะนำภาพหน้าตรงของบุคคลในฐานข้อมูลมาทำการทดลองทั้งหมด คือจำนวน 100 ภาพ เพื่อเปรียบเทียบกับใบหน้าผู้ทดสอบและทำการทดลองทั้งหมด 5 ครั้ง



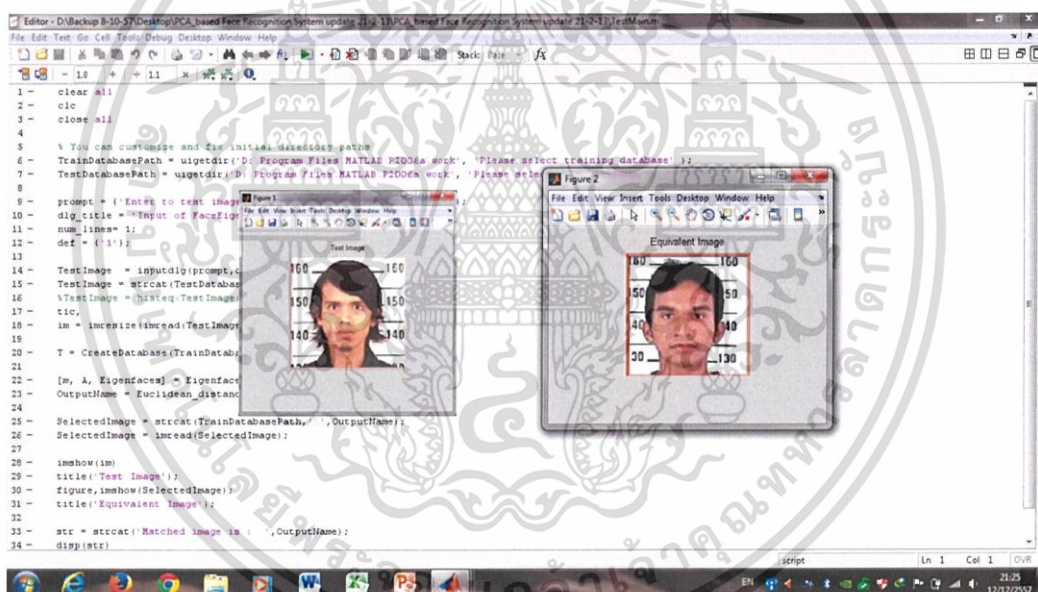
รูปที่ 4.15 รูปผู้ก่อการร้ายที่ใส่สัญญาณรบกวน Gaussian Noise

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 แสดงผลการทดลองการระบุตัวบุคคลโดยใช้ภาพใบหน้าที่มีเพิ่มสัญญาณรบกวน Gaussian Noise ของผู้ทดสอบ

ครั้งที่	จำนวนภาพที่ทดลอง	จำนวนภาพที่ระบุตัวบุคคลได้ถูกต้อง	%ความถูกต้อง	%ความถูกต้องเฉลี่ย
1	100	3	6	8.8
2	100	6	12	
3	100	4	8	
4	100	6	12	
5	100	3	6	

จากการทดลอง จะเห็นได้ว่าเมื่อนำภาพใบหน้าจากฐานข้อมูลมาทดสอบกับใบหน้าผู้ทดสอบ ในลักษณะภาพใบหน้าที่มีสัญญาณรบกวนแบบ Gaussian Noise จะพบว่ามีความผิดพลาดสูงมาก ดังแสดงดังรูปที่ 4.15



รูปที่ 4.16 รูปผู้ก่อการร้ายที่ไม่สามารถระบุตัวตนได้ถูกต้องจากการเพิ่มสัญญาณรบกวนแบบ Gaussian Noise

#### 4.3.3 ทดสอบระบบที่นำเสนอเปรียบเทียบกับวิธีอื่น

ในการทดลองนี้จะนำภาพหน้าตรงของบุคคลในฐานข้อมูลมาทำการทดลองทั้งหมด คือจำนวน 100 ภาพ เพื่อเปรียบเทียบกับใบหน้าผู้ทดสอบและทำการทดลองทั้งหมด 5 ครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.5 แสดงผลการทดลองการระบุตัวบุคคลโดยเปรียบเทียบกับวิธีอื่นๆ

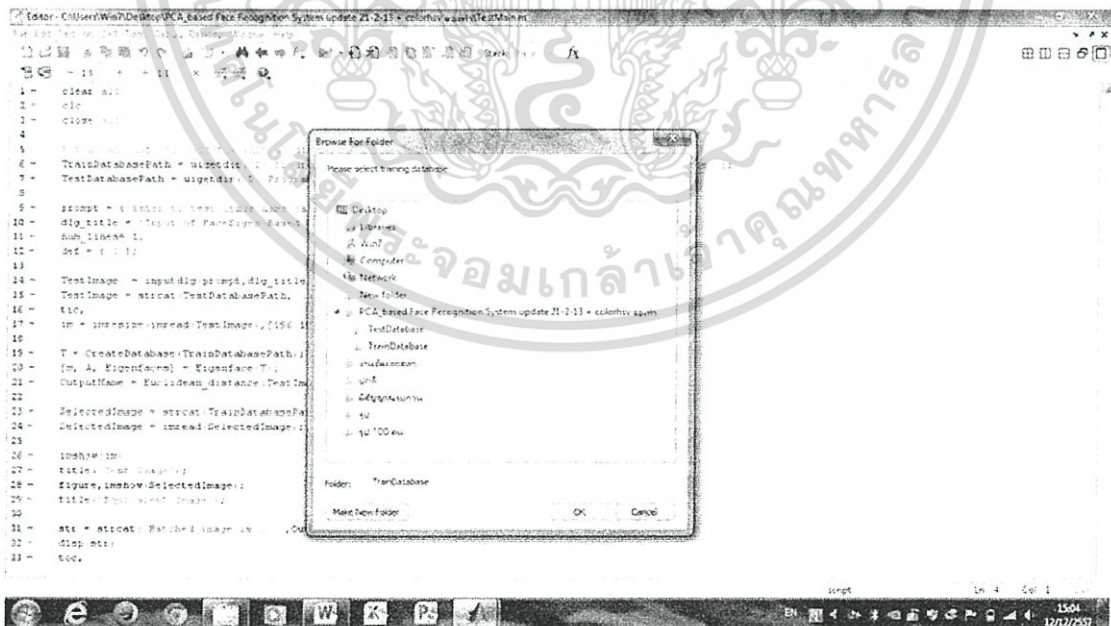
วิธีการ ภาพ	Proposed อัตราความถูกต้อง	LDA อัตราความถูกต้อง	ICA อัตราความถูกต้อง	Geometry-based methods อัตราความถูกต้อง
ภาพปกติ	100%	95.80%	94.66%	89.41%
เพิ่ม Salt and pepper	98.80%	94.81%	91.50%	82.73%
เพิ่ม Gaussian Noise	8.8%	ไม่ได้ทำการทดลอง	ไม่ได้ทำการทดลอง	ไม่ได้ทำการทดลอง

#### 4.4 ขั้นตอนการทำงานในเครื่องคอมพิวเตอร์

ขั้นตอนในส่วนนี้จะขั้นตอนการทำงานของเครื่องคอมพิวเตอร์เมื่อทำการเก็บข้อมูลของทั้งภาพที่ใช้ในการเรียนรู้และภาพที่ใช้ในขั้นตอนการทดสอบเรียบร้อยแล้ว จะทำการตรวจสอบการรู้จำของโปรแกรมโดยทำการเลือกภาพใบหน้าผู้ก่อการร้ายที่เก็บภาพไว้ในส่วนของภาพที่ใช้ในขั้นตอนการทดสอบ แล้วทำการเลือกภาพตัวอย่างมาทำการทดสอบในโปรแกรมดังตัวอย่างด้านล่างดังนี้

##### 4.4.1 เปิดเพิ่มข้อมูลเลือก TrainDatabase

ส่วนนี้คือการเลือกในส่วนที่เก็บข้อมูลของภาพผู้ก่อการร้ายทั้ง 100 คนเอาไว้

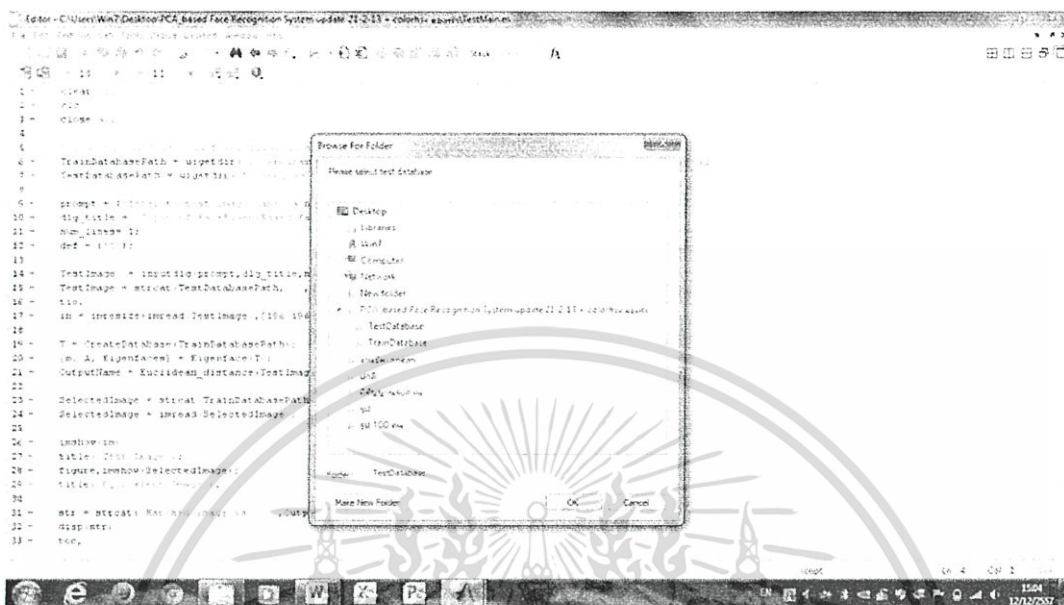


รูปที่ 4.17 การทำงานของระบบรูปที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.4.2 เปิดเพิ่มข้อมูลเลือก TestDatabase

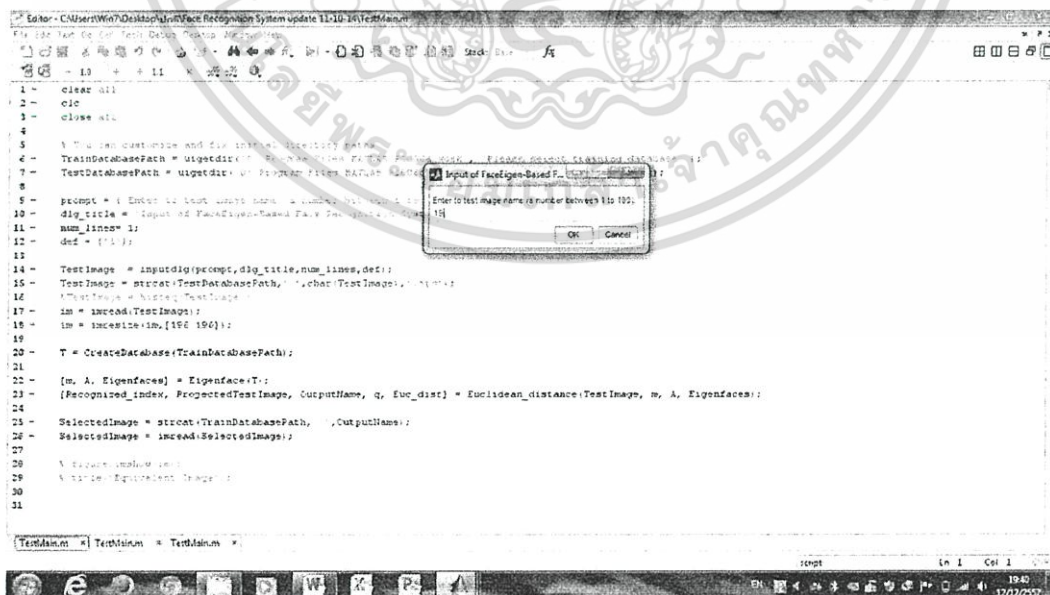
ส่วนนี้คือการเลือกในส่วนของข้อมูลที่เก็บข้อมูลของภาพผู้ก่อการร้ายที่จะนำมาทดสอบการรู้จำในโปรแกรม



รูปที่ 4.18 การทำงานของระบบรูปที่ 2

### 4.4.3 เลือกภาพใบหน้าที่จะทำการทดสอบ

เลือกภาพใบหน้าที่ต้องการทำการทดสอบรู้จำในโปรแกรม โดยในตัวอย่างทำการเลือกภาพใบหน้าผู้ก่อการร้ายคนที่ 19

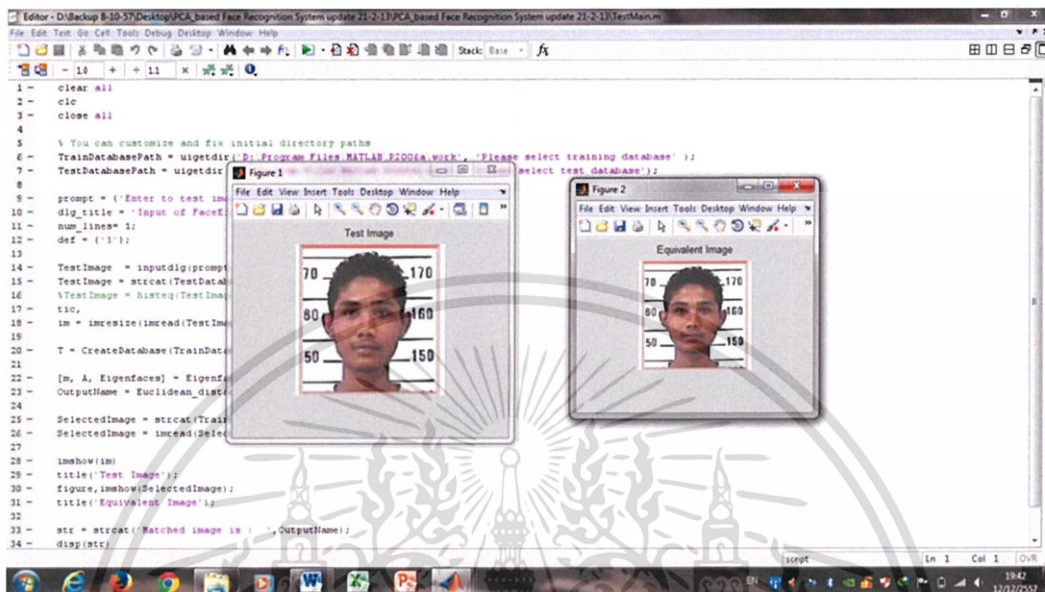


รูปที่ 4.19 การทำงานของระบบรูปที่ 3

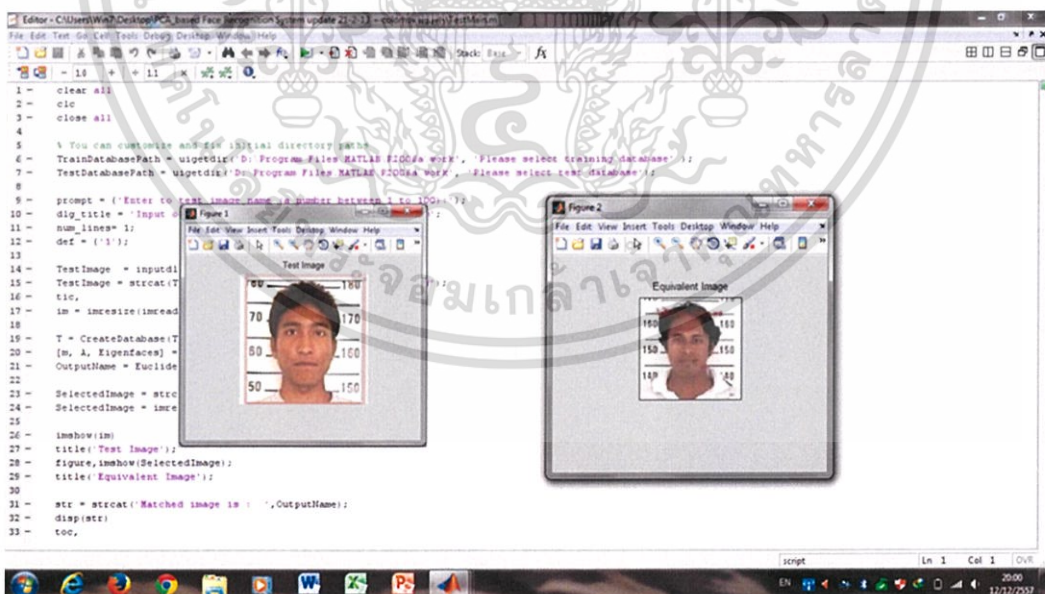
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4.4 ระบบทำการตรวจสอบ

ระบบจะทำการประมวลผลตามขั้นตอนการทำงานทั้งหมดตั้งที่ได้กล่าวไว้ในตอนต้นจนจบกระบวนการ ผลการรู้จำจะออกมาในสองกรณีคือ สามารถจดจำภาพใบหน้าได้และไม่สามารถจดจำภาพใบหน้าได้ ดังที่แสดงในรูปภาพตัวอย่าง



รูปที่ 4.20 การทำงานของระบบรูปที่ 4 กรณีสามารถจดจำภาพได้



รูปที่ 4.21 การทำงานของระบบรูปที่ 5 กรณีไม่สามารถจดจำภาพได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4.5 ระยะเวลาในการประมวลผล

ในส่วนนี้จะเป็นการจับเวลาในการประมวลผลภาพในขั้นตอนของโปรแกรม ตั้งแต่การเริ่มนำภาพเข้ามาทดสอบในกระบวนการจนจบที่การแสดงผลของการจดจำภาพใบหน้าในแต่ละภาพ โดยจะทำการจับเวลาเฉลี่ยของภาพทั้งสามแบบ คือ ภาพแบบปกติ ภาพที่มีสัญญาณรบกวนแบบ Salt and pepper noise และภาพที่มีสัญญาณรบกวนแบบ Gaussian Noise อย่างละ 50 ภาพ แล้วทำการจับเวลาในการประมวลผลทั้งสามแบบขณะทำการทดลอง สุดท้ายทำการหาค่าเฉลี่ยของเวลาในการประมวลผล ทั้งนี้ในส่วนของการจับเวลาในการประมวลผลจะไม่ได้คิดในเรื่องการจดจำภาพใบหน้าที่มีความถูกต้องด้วย แต่จะจับเพียงเวลาในการประมวลผลเท่านั้น

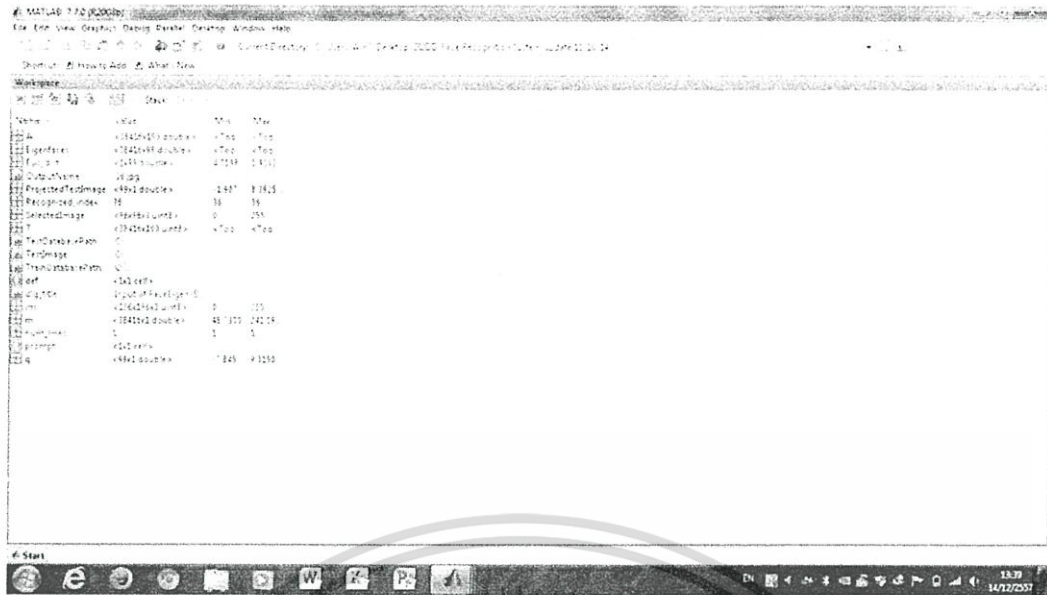
ตารางที่ 4.6 แสดงผลการจับเวลาในการประมวลผลภาพในการทดสอบ

ลักษณะใบหน้า	เวลาเฉลี่ยเป็นวินาที
ภาพปกติ	5.066700
ภาพที่มีสัญญาณรบกวนแบบ Salt and pepper noise	5.152600
ภาพที่มีสัญญาณรบกวนแบบ Gaussian Noise	5.134400

จากเวลาเฉลี่ยจะเห็นได้ว่าภาพในลักษณะปกติที่ปราศจากสัญญาณรบกวนจะใช้เวลาในการประมวลผลการรู้จำที่เร็วกว่าภาพที่มีสัญญาณรบกวนทั้งสองแบบ

#### 4.4.6 ค่าขนาดของข้อมูลที่แสดงในโปรแกรม

จากการทดลองจะเห็นได้ว่าขนาดของข้อมูลมีการลดขนาดโดเมนชั้นของข้อมูลลงจากวิธีการ PCA ช่วยให้การประมวลผลของโปรแกรมสามารถทำได้เร็วขึ้น จากปกติภาพ 1 ภาพที่เป็นภาพแบบ 2 มิติขนาด ( $N \times N$ ) จะมีขนาดโดเมนชั้นคือ  $196 * 196$  หากมีการใช้ 100 ภาพจะได้ขนาด  $19,600 * 19,600$  เมื่อมาทำกระบวนการ Eigenface ใน 1 ภาพ จะได้ขนาดภาพ  $38,416 * 1$  ถ้าทำ 99 ภาพจะได้  $38,416 * 99$  จะเห็นได้ว่าขนาดโดเมนชั้นภาพ 2 มิติ เมื่อทำเป็นรูปภาพแบบ 1 มิติ  $N^2 \times 1$  จะลดลงจาก 196 เป็น 1 (กรณีคิดที่ 1 ภาพ)



รูปที่ 4.22 การแสดงขนาดข้อมูลใน Workspace

#### 4.5 สรุปผลการทดลอง

การวิเคราะห์ประสิทธิภาพของอัลกอริทึมไอเกนเฟซหลังจากที่ได้ทำการทดลองระบุใบหน้าทดสอบอ้างอิงกับฐานข้อมูลดังที่กล่าวมาในข้างต้นแล้ว ในที่นี้จะกล่าวถึงการวิเคราะห์โดยรวมของประสิทธิภาพการระบุบุคคลของโปรแกรมโดยจะแสดงดังตารางต่อไปนี้

ตารางที่ 4.7 แสดงผลการวิเคราะห์ประสิทธิภาพของอัลกอริทึมไอเกนเฟซ

ลักษณะของใบหน้าทดสอบ	ต้นแบบ	เพิ่มสัญญาณรบกวน
%ความถูกต้อง	100	98.8

จากผลการวิเคราะห์แสดงให้เห็นว่า ประสิทธิภาพของอัลกอริทึมโดยรวมอยู่ในเกณฑ์ดีเป็นที่น่าพอใจ แต่การทดลองจำเป็นจะต้องอยู่ในขอบเขตที่กำหนด

## สรุปผลการวิจัยและแนวทางในการพัฒนา

### 5.1 สรุปผลการวิจัย

ในกระบวนการรู้จำภาพใบหน้าผู้ก่อการร้ายโดยใช้วิธีการหลักในงานวิจัยนี้คือวิธีการโอเคนเฟช ซึ่งใช้หลักการของ PCA ทำให้สามารถแยกองค์ประกอบของชุดข้อมูลให้ออกมาในรูปแบบที่อธิบายได้ง่าย และไม่มี ความซับซ้อนของข้อมูลและทำให้สามารถรู้จำภาพใบหน้าผู้ก่อการร้ายที่นำมาประกอบ งานวิจัยได้ถึง 100 % และเมื่อทำการเปรียบเทียบกับวิธีการอื่น อาทิ เช่น ICA และ Geometry-based methods จะเห็นได้ว่าได้ผลที่ดีกว่า ICA ซึ่งเป็นวิธีการค้นหาองค์ประกอบความเปลี่ยนแปลง ซึ่งก่อให้เกิดสัญญาณรบกวนต่อองค์ประกอบของต้นกำเนิดแล้วกลายเป็นชุดข้อมูลนั้น ส่วนวิธีการ Geometry-based methods ทำการหาในส่วนของตา และปากเพื่อทำการประมวลผลก็ยังได้ผลการ ทดลองในการรู้จำที่ได้ผลการรู้จำน้อยที่สุด อันน่าจะเนื่องมาจากการใช้ข้อมูลในการประมวลผลน้อย เพียงแค่ในส่วนตา และปากเพียงเท่านั้น

การทดลองประสิทธิภาพการทำงานของระบบ คือส่วนของซอฟต์แวร์ ทำให้สามารถสรุป ผลได้ดังนี้

#### 5.1.1 สรุปผลประสิทธิภาพด้านซอฟต์แวร์

จากการทดสอบการระบุใบหน้าบุคคลโดยใช้อัลกอริทึมโอเคนเฟช ทดสอบลักษณะ ของภาพในหลายๆ รูปแบบทำให้สามารถสรุปผลได้ดังนี้

- 1) โปรแกรมจะสามารถระบุใบหน้าบุคคลได้อย่างมีประสิทธิภาพมาก โดยที่รูปบุคคล ทดสอบนั้นต้องอยู่ในลักษณะการถ่ายและสภาพแวดล้อมที่ใกล้เคียงกับรูปบุคคลที่อยู่ในฐานข้อมูล
- 2) ผลกระทบที่ส่งผลกับประสิทธิภาพของการทำงานส่วนมากจะมาจากประสิทธิภาพ ของคอมพิวเตอร์ค่อนข้างมาก เพราะฉะนั้นในบางกรณีอาจมีอาการแฮงค์ของโปรแกรมเกิดขึ้นได้

### 5.2 ปัญหาที่เกิดขึ้น

#### 5.2.1 ปัญหาโดยทั่วไป

ที่เกิดขึ้นจะเป็นในขั้นตอนของการดึงภาพของใบหน้าด้วยวิธีการ RGB TO HSV เนื่องจากภาพที่ใช้บางภาพจะมีความสว่างของภาพมากเกินไปและบางภาพมีโทนสีเดียวกัน ทำให้ ขณะที่ทำการเลือกช่วงค่าสี HSV ทำได้โดยยาก และเมื่อทำแล้วก็ได้ส่วนใบหน้าที่ไม่สมบูรณ์เท่าที่ควร แต่ก็สามารถทำการแก้ไขได้โดยทำการปรับปรุงภาพด้วยวิธีการต่างๆ เช่น การยืดฮีสโตแกรม การบวก ด้วยค่าคงที่ การหารด้วยค่าคงที่ เป็นต้น อีกส่วนหนึ่งคือในเรื่องของข้อจำกัดเกี่ยวกับเรื่องฐาน ข้อมูลภาพใบหน้าผู้ก่อการร้ายที่นำมาใช้ ไม่สามารถหาภาพใบหน้าที่น่ามาทำการทดลองได้ในหลาย รูปแบบและหลายลักษณะ เนื่องจากเป็นภาพที่มีฐานข้อมูลในชั้นความลับของทางราชการ จึงใช้ได้ เพียงภาพใบหน้าผู้ก่อการร้ายที่มีการประกาศจับจากทางส่วนราชการแล้ว อีกทั้งภาพที่นำมาใช้ก็มี จำนวนน้อยเพียง 100 ภาพ ซึ่งปกติควรจะมีข้อมูลที่ใช้ในการทดลองโดยทั่วไปที่ 500 ภาพขึ้นไป จึง นับเป็นอีกปัญหาในการทดลอง

### 5.2.2 การหาภาพที่นำมาใช้

เนื่องจากภาพที่นำมาใช้เป็นภาพของผู้ก่อการร้ายในสามจังหวัดชายแดนภาคใต้ที่มีการประกาศจับแล้วเป็นภาพในลักษณะหน้าตรง ทำให้สามารถใช้งานได้จริงในงานการค้นหภาพใบหน้าบุคคลผู้ต้องสงสัยให้มีความรวดเร็วยิ่งขึ้นได้ แต่ก็ไม่สามารถใช้ในงานของการค้นหภาพใบหน้าบุคคลในลักษณะอื่นได้ เช่นภาพด้านข้างหรือได้มาในมุมอื่นๆ ที่ไม่ใช่ใบหน้าตรง

### 5.2.3 วิธีการแยกในส่วนของตาและปาก

ส่วนหนึ่งที่เลือกวิธีการนี้เพราะต้องการจะทำการทดสอบว่าหากได้เพียงส่วนตาหรือปากของผู้ต้องสงสัยมาแล้ว จะสามารถทำการทดสอบกับโปรแกรมได้ดีขนาดไหน แต่ก็ได้ผลที่น่าพอใจคือสามารถรู้จำได้ถึง 80 % ขึ้นไป แต่ในความเป็นจริงหากยังไม่สามารถทำการตรวจสอบได้ผลถูกต้องมากกว่านี้ก็ยังไม่ควรนำมาใช้เพราะเป็นเรื่องของสิทธิมนุษยชนหากจับผู้ต้องสงสัยผิดตัวมาก็จะมีความผิดทางกฎหมาย

### 5.2.4 อุปสรรคทางด้านซอฟต์แวร์

เนื่องด้วยความรู้ที่ไม่มีประสบการณ์ในการเขียนโปรแกรม Matlab ทำให้ใช้เวลาค่อนข้างนานในการศึกษา เพื่อที่จะทำการออกแบบโปรแกรมทั้งหมด และด้วยเหตุนี้อาจจะยังทำให้โปรแกรมไม่มีประสิทธิภาพสูงสุด ในส่วนนี้สามารถแก้ไขได้โดยการเข้าพบผู้ที่มีความเชี่ยวชาญเฉพาะด้านโดยตรงก็จะสามารถใช้โปรแกรมที่มีความเหมาะสมกับงาน และทำให้โปรแกรมการทำงานมีประสิทธิภาพสูงขึ้น

### 5.2.5 ปัญหากระบวนการรู้จำเมื่อเพิ่มสัญญาณรบกวน

จากการทดลองเมื่อทำการใส่ในส่วนของสัญญาณรบกวน Salt and pepper noise จำนวน 20% และ สัญญาณรบกวน Gaussian Noise โดยใส่สัญญาณรบกวน 0.2 และค่าความเบี่ยงเบน 0.01 ของภาพที่นำมาเข้ากระบวนการรู้จำ โดยสัญญาณรบกวนทั้งสองแบบเป็นสัญญาณรบกวนชนิดที่มักจะทำให้เกิดขึ้นกับภาพในลักษณะภาพถ่ายดิจิทัล โดยการเลือกขนาดของสัญญาณรบกวนในการทดสอบนั้น เลือกโดยการพิจารณาถึงปริมาณของสัญญาณโดยทั่วไปที่อาจเกิดขึ้นจากภาพถ่ายชนิดที่ได้รับมาเป็นฐานข้อมูล โดยเมื่อใส่สัญญาณรบกวนลงในภาพทดสอบแล้ว จะพบว่า โดยเฉพาะสัญญาณรบกวนแบบ Gaussian Noise ระบบการทำงานจะไม่สามารถจดจำภาพใบหน้าได้ถูกต้องเป็นจำนวนมาก แนวทางการแก้ไขจะเริ่มตั้งแต่ขั้นการเก็บภาพมาในขั้นตอนแรกจะทำการปรับปรุงคุณภาพด้วยวิธีการต่างๆตามลักษณะสัญญาณรบกวนที่เกิดขึ้นภายในภาพ เช่น การกรองเชิงพื้นที่ การกรองด้วยสถิติเชิงอันดับ และการกรองแบบปรับตัวได้ โดยในโปรแกรม Matlab จะมีฟังก์ชันการแก้ไขภาพพรมามั่วอยู่ด้วยกันสี่ฟังก์ชันที่สามารถนำมาใช้ได้ดังนี้คือ การแก้ไขภาพพรมามั่วด้วยตัวกรองแบบเวียเนอร์ การแก้ไขภาพพรมามั่วด้วยตัวกรองแบบปกติ การแก้ไขภาพพรมามั่วด้วยตัวกรองแบบลูซี-ริชาร์ดสัน และการแก้ไขภาพพรมามั่วด้วยตัวกรองแบบไบลนตีคอนโวลูชัน

### 5.3 แนวทางในการพัฒนา

ระบบการรู้จำใบหน้าผู้ก่อการร้ายที่นำมาใช้ในงานวิจัยนี้ถึงจะสามารถระบุตัวตนของผู้ก่อการร้ายได้ถึง 100 % แต่ก็ยังเป็นเพียงใบหน้าในลักษณะหน้าตรงเพียงอย่างเดียว ประกอบกับมีฐานข้อมูลของภาพเพียงลักษณะเดียวคือหนึ่งคนต่อหนึ่งภาพในการทดสอบ จึงทำให้มีประสิทธิภาพในการจดจำภาพใบหน้าที่สูง ซึ่งแนวทางในการวิจัยพัฒนาต่อจากนี้คือการใช้การประมวลผลภาพในลักษณะใบหน้าในทุกลักษณะ โดยอาจจะทำการเสนอเรื่องให้ทางหน่วยงานที่ทำการเก็บฐานข้อมูลของบุคคลให้ทำการถ่ายด้านข้างทั้งสองข้าง และนำมาพัฒนาโปรแกรมร่วมกับอัลกอริทึมที่เหมาะสมต่อไป

การจัดทำระบบการรู้จำภาพใบหน้าผู้ก่อการร้าย ทำให้ผู้จัดทำได้เข้าใจถึงความเป็นมาของงานทางด้าน Image processing ทราบถึงงานวิจัยและทฤษฎีที่เกี่ยวข้อง รู้ขั้นตอนและวิธีการคิดคำนวณอัลกอริทึมโอเคนเฟช ระบบจะสามารถทำงานได้รวดเร็ว หากได้รับการประมวลผลด้วยคอมพิวเตอร์ที่มีประสิทธิภาพสูง

1) ถ้าสามารถนำทฤษฎีเกี่ยวกับ Image processing อื่นๆที่เกี่ยวข้องเข้ามาช่วยในการดึงคุณลักษณะเด่นบนใบหน้าเพื่อจะทำให้ผลที่ได้ถูกต้องแม่นยำมากขึ้น อาทิเช่น การแยกคุณลักษณะเด่นของใบหน้าและเฉพาะส่วนโดยใช้แบบจำลองเชิงเรขาคณิตที่สามารถเพิ่มประสิทธิภาพในการจดจำใบหน้าได้

2) สามารถนำอัลกอริทึมไปประยุกต์ใช้กับภาษาคอมพิวเตอร์อื่นๆได้ เช่น Fortran, Borland C/C++, Microsoft Visual C++ เป็นต้น

3) สามารถนำภาพสองมิติที่มีไปทำการพัฒนาสร้างแบบจำลองใบหน้าในลักษณะสามมิติด้วยวิธีการ เช่น Eigenface model, ASM (Active Shape model) และ AAM (Active Appearance model)

## เอกสารอ้างอิง

- [1] กษิรพันธ์ มาสกุล และ สมชัย ตั้งสถิตยวงกูร. “Biometrics.” [online]. Available: <http://www.spu.ac.th/~bmetric/index.htm>
- [2] Bagherian, E. and Rahmat, R.W.O.K. “Facial feature extraction for face recognition: a review.” *International Symposium on 2: 1-9*, 2008;
- [3] Kanade, T. “Computer Recognition of Human faces”. Basel and Stuttgart: Birkhauser, 1997.
- [4] Yuille, A., Cohen, D. and Hallinan, P. “Facial feature extraction from faces using deformable templates.” *Proc. IEEE Computer Soc. Conf. On Computer Vision and Pattern Recognition*, pp. 104-109, 1989.
- [5] Chang, T.C., Huang, T.S. and Novak, C. “Facial feature extraction from colour images.”, *Proceedings of the 12<sup>th</sup> IAPR International Conference on Pattern Recognition*, 2: 39-43, 1994.
- [6] Tian, Y., Kanade, T. and Cohn, J.F. “Evaluation of Gaborwavelet- based facial action unit recognition in image sequences of increasing complexity.”, *Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 218 –223, May 2002.
- [7] Viola, P. and Jones, M.J. “Rapid Object Detection using a Boosted Cascade of Simple Features.” *IEEE CVPR*, 2001.
- [8] Turk, M. and Pentland, A. “Eigenfaces for recognition.” *Journal of Cognitive Neuroscience*, 3(1): 71-86, 1991.
- [9] Xu, D., Yan, S., Shang, L., Li, M., Ma, W. and Liu, Z. “Parallel Image Matrix Compression for Face Recognition.” *IEEE 2005 International Multimedia Modeling Conference (MMM'05)*, 12 – 14 January 2005, pp. 1-7, 2005.
- [10] Huttenlocher, D.P., Klanderman, G.A. and Rucklidge, W.J. “Comparing Images Using the Hausdorff Distance.” *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(9): 850-863, 1993.
- [11] Jain, A.K. “Fundamentals of Digital Image Processing.” New York: Prentice-Hall International, Inc. 1989.
- [12] Jain, A.K. “Advances in Mathematical Models for Image Processing”. *Proceedings IEEE 69(5): 502-528*, 1981.



ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ผลงานวิจัยที่ได้รับการตีพิมพ์

พิศณุ คุมีชัย. “ระบบรู้จำใบหน้าผู้ก่อการร้ายด้วยการตรวจสอบสีผิวบุคคลและไอเก็นเฟส.”

Ladkrabang Engineering Journal, Bangkok 2 June 2014.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรม

โปรแกรมที่ใช้เป็น Matlab Version 10 ซึ่งประกอบไปด้วยโปรแกรมใน M-file ที่มา

Face detection

<http://www.mathworks.com/matlabcentral/fileexchange/11073-face-detection-system>

Face recognition

<http://www.mathworks.com/matlabcentral/fileexchange/16760-face-recognition>

- 1) Cent\_moment
- 2) ConvertHaarcascadeXMLOpenCV
- 3) CreateDatabase
- 4) Eigenface
- 5) Euclidean distance
- 6) ObjectDetection
- 7) Radialpoly
- 8) Testfull
- 9) Testhaar
- 10) Testhaaraddcolor
- 11) TestMain
- 12) Untitled
- 13) Zernike\_main



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* Cent_moment*/
function n_pq=cent_moment(p,q,A)

[m n]=size(A);
moo=sum(sum(A));

m1o=0;
mo1=0;
for x=0:m-1
for y=0:n-1
m1o=m1o+(x)*A(x+1,y+1);
mo1=mo1+(y)*A(x+1,y+1);
end
end
xx=m1o/moo;
yy=mo1/moo;

mu_oo=moo;

mu_pq=0;
for ii=0:m-1
for jj=0:n-1
y=jj-yy;
mu_pq=mu_pq+(x)^p*(y)^q*A(ii+1,jj+1);
end
end

gamma=0.5*(p+q)+1;
n_pq=mu_pq/moo^(gamma);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/* ConvertHaarcasadeXMLOpenCV */
```

```
function ConvertHaarcasadeXMLOpenCV(filename)
```

```
% This function ConvertHaarcasadeXMLOpenCV, converts a openCV .xml file  
% into a matlab data file and script with the same structure inside.
```

```
%
```

```
% Example,
```

```
% filename='haarcascade_eye';
```

```
% ConvertHaarcasadeXMLOpenCV(filename);
```

```
%
```

```
% Example,
```

```
% f=dir('*.*xml');
```

```
% for i=1:length(f),
```

```
%   filename=f(i).name;
```

```
%   ConvertHaarcasadeXMLOpenCV(filename(1:end-4));
```

```
% end
```

```
%
```

```
% Function is written by D.Kroon University of Twente (November 2010)
```

```
j=find(filename==' '); if(~isempty(j)), filename=filename(1:j-1); end
```

```
fid = fopen([filename '.xml'], 'r');
```

```
c = fread(fid, inf, 'char=>char');
```

```
fclose(fid);
```

```
c(c==13)=[];
```

```
c(c==10)=[];
```

```
fl = regexp(c, '<', 'split');
```

```
fid = fopen([filename '.m'], 'w');
```

```
h=0; nw=0;
```

```
infoname=cell(1,10);
```

```
infocount=zeros(1,10);
```

```
for i=2:length(fl)
```

```
  str=fl{i};
```

```
  if(length(str)>1), st=str(1); else st=""; end
```

```
  switch(st)
```

```
    case {'!', ',', '?'}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

continue
case '/'
t=find(str=='>',1,'first');
name=str(2:t-1);
infocount(h)=0;
h=h-1;
continue
otherwise
t1=find(str=='>',1,'first');
t2=find(str(1:t1-1)==' ');
if(~isempty(t2))
t=t2;
else
t=t1;
end
name=str(1:t-1);

data=str(t+1:end);
h=h+1;
if(name=='_')
infocount(h-1)=infocount(h-1)+1;
name='value';
end
infoname{h}=name;
while(~isempty(data)&&(data(1)==' ')); data=data(2:end); end
while(~isempty(data)&&(data(end)==' ')); data=data(1:end-1); end
end
if(~isempty(data))
in="";
for j=1:h
if(infocount(j)>0)
count=['(' num2str(infocount(j)) ')'];
else
count="";
end

if(j==1)
in=[infoname{j} count];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

else
in=[in '.' infoname{j} count];
end
end
in=strrep(in, '.value.', ':');
num=str2num(data);
if(~isempty(num))
if(length(num)==1)
str=[in ' = ' data ''];
else
str=[in ' = [' data ''];];
end
else
str=[in ' = "" data ""];];
end
nw=nw+1;
if(nw==1), str=['%' str]; end
fprintf(fid,'%s\n',str);
if(nw>1)
eval(str)
end
end
end
fclose(fid);
save([filename '.mat'],'opencv_storage');

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* CreateDatabase */
function T = CreateDatabase(TrainDatabasePath)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% File management
TrainFiles = dir(TrainDatabasePath);
Train_Number = 0;

for i = 1:size(TrainFiles,1)
    if
not(strcmp(TrainFiles(i).name, '.')|strcmp(TrainFiles(i).name, '..')|strcmp(TrainFiles(i).name, '
Thumbs.db'))
Train_Number = Train_Number + 1; % Number of all images in the training database
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Construction of 2D matrix from 1D
image vectors
T = [];
for i = 1 : Train_Number

% I have chosen the name of each image in databases as a corresponding
% number. However, it is not mandatory!
str = int2str(i);
str = strcat('\',str, '.jpg');
str = strcat(TrainDatabasePath, str);

img = imresize(imread(str), [196 196]);
img = rgb2gray(img);

[irow icol] = size(img);

temp = reshape(img', irow*icol, 1); % Reshaping 2D images into 1D image vectors
T = [T temp]; % 'T' grows after each turn
End

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* Eigenface */
function [m, A, Eigenfaces] = Eigenface(T)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Calculating the mean image
m = mean(T,2); % Computing the average face image m = (1/P)*sum(Tj's) (j = 1 : P)
Train_Number = size(T,2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Calculating the deviation of each
image from mean image
A = [];
for i = 1 : Train_Number
    temp = double(T(:,i)) - m; % Computing the difference image for each image in the
training set Ai = Ti - m
    A = [A temp]; % Merging all centered images
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Snapshot method of Eigenface
methos

L = A'*A; % L is the surrogate of covariance matrix C=A*A'.
[V D] = eig(L); % Diagonal elements of D are the eigenvalues for both L=A*A and
C=A*A'.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Sorting and eliminating eigenvalues

L_eig_vec = [];
for i = 1 : size(V,2)
    if( D(i,i)>1 )
        L_eig_vec = [L_eig_vec V(:,i)];
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Calculating the eigenvectors of
covariance matrix 'C'
Eigenfaces = A * L_eig_vec; % A: centered image vectors
%%

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/* Euclidean distance */
```

```
function [Recognized_index, ProjectedTestImage, OutputName, q, Euc_dist] =  
Euclidean_distance(TestImage, m, A, Eigenfaces)
```

```
% Description: This function compares two faces by projecting the images into  
facespace and
```

```
% measuring the Euclidean distance between them.
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Projecting centered image vectors into  
facespace
```

```
ProjectedImages = [];
```

```
Train_Number = size(Eigenfaces,2);
```

```
for i = 1 : Train_Number
```

```
temp = Eigenfaces*A(:,i); % Projection of centered images into facespace
```

```
ProjectedImages = [ProjectedImages temp];
```

```
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Extracting the PCA features from test  
image
```

```
InputImage = imresize(imread(TestImage),[196 196]);
```

```
%InputImage = histeq(rgb2gray(InputImage));
```

```
temp = InputImage(:, :, 1);
```

```
%%
```

```
I = temp;
```

```
ConvertHaarcasadeXMLOpenCV('HaarCascades/haarcascade_frontalface_alt.xml');
```

```
I1 = I;
```

```
FilenameHaarcasade = 'HaarCascades/haarcascade_frontalface_alt.mat';
```

```
Objects=ObjectDetection(I1,FilenameHaarcasade);
```

```
ShowDetectionResult(I1,Objects);
```

```
[x1, x2, y1, y2] = ShowDetectionResult(I1,Objects);
```

```
I_roi = I1(y1:y2,x1:x2);
```

```
figure, imshow(I_roi)
```

```
I_roi = imresize(I_roi, [196 196]);
```

```
temp = I_roi;
```

```
%%
```

```
[irow icol] = size(temp);
```

```
InImage = reshape(temp',irow*icol,1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

InImage = histeq(InImage);
Difference = double(InImage)-m; % Centered test image
ProjectedTestImage = Eigenfaces*Difference; % Test image feature vector

%%%%%%%%%%%%%% Calculating Euclidean distances

Euc_dist = [];
for i = 1 : Train_Number
q = ProjectedImages(:,i);
temp = ( norm( ProjectedTestImage - q ) )^2;
Euc_dist = [Euc_dist temp];
end

[Euc_dist_min , Recognized_index] = min(Euc_dist)
OutputName = strcat(int2str(Recognized_index),'.jpg');

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* ObjectDetection */
function Objects = ObjectDetection(Picture,FilenameHaarcasade,Options)
% This function ObjectDetection is an implementation of the Detection in
% the Viola-Jones framework. In this framework Haar-like features are used for
% rapid object detection. It supports the trained classifiers in the
% XML files of OpenCV which can be download as part of the OPENCV software
% on opencv.willowgarage.com
%
% Objects=ObjectDetection(I,FilenameHaarcasade,Options)
%
% inputs,
% I : 2D image, or Filename of an image
% FilenameHaarcasade : The filename of a Matlab file with a Haarcasade
% which is created from an OpenCV xml file, using the
% function ConvertHaarcasadeXMLOpenCV.
% Options : A struct with options
% Options.ScaleUpdate : The scale update, default 1/1.2
% Options.Resize : If boolean is true (default), the function will
% resize the image to maximum size 384 for less cpu-time
% Options.Verbose : Display process information
%
% outputs,
% Objects : An array n x 4 with [x y width height] of the detected
% objects
%
% Literature :
% - Rainer Linehart and Jochend Maydt, "An Extended Set of Haar-like
% Features for Rapid Object Detection"
% - Paul viola and Michael J. Jones, "Rapid Object Detection using
% a Boosted Cascade of Simple Features", 2001
%
% Example 1,
% % Convert an OpenCV classifier XML file to a Matlab file
% ConvertHaarcasadeXMLOpenCV('HaarCascades/haarcascade_frontalface_alt.xml');
% % Detect Faces
% Options.Resize=false;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

%
ObjectDetection('Images/1.jpg','HaarCascades/haarcascade_frontalface_alt.mat',Options);
%
% Example 2,
% % Convert an OpenCV classifier XML file to a Matlab file
% ConvertHaarcascadeXMLOpenCV('HaarCascades/haarcascade_frontalface_alt.xml');
% I = imread('Images/2.jpg');
% FilenameHaarcascade = 'HaarCascades/haarcascade_frontalface_alt.mat';
% Objects=ObjectDetection(I,FilenameHaarcascade);
% ShowDetectionResult(I,Objects);
%
% Function is written by D.Kroon University of Twente (November 2010)

% The default Options
defaultoptions=struct('ScaleUpdate',1/1.2,'Resize',true,'Verbose',true);

% Add subfunction path to Matlab Search Path
functionname='ObjectDetection.m';
functiondir=which(functionname);
functiondir=functiondir(1:end-length(functionname));
addpath([functiondir '/SubFunctions'])

% Check inputs
%if(ischar(Picture))
% if(~exist(Picture,'file'))
%   error('face_detect:inputs','Image not Found');
% end
%end
%if(~exist(FilenameHaarcascade,'file'))
%   error('face_detect:inputs','Haarcascade not Found');
%end

% Process input options
if(~exist('Options','var')), Options=defaultoptions;
else
tags = fieldnames(defaultoptions);
for j=1:length(tags),

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(~isfield(Options,tags{i})), Options.(tags{i})=defaultoptions.(tags{i}); end
end
if(length(tags)~=length(fieldnames(Options))),
warning('image_registration:unknownoption','unknown options found');
end
end

% Read the Picture from file if Picture is a filename
%if(ischar(Picture))
% Picture = imread(Picture);
%end

% Get the HaarCasade for the object detection
HaarCasade=GetHaarCasade(FileNameHaarcasade);

% Get the integral images
IntergralImages= GetIntergralImages(Picture,Options);

Objects = HaarCasadeObjectDetection(IntergralImages,HaarCasade,Options);

% Show the finale results
if(nargout==0)
ShowDetectionResult(Picture,Objects);
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* Radialpoly */
function rad = radialpoly(r,n,m)
% -----
% Copyright C 2012 Amir Tahmasbi
% The University of Texas at Dallas
% a.tahmasbi@utdallas.edu
% http://www.utdallas.edu/~a.tahmasbi/research.html
%
% License Agreement: You are free to use this code in your scientific
% research but you should cite the following papers:
%
% [1] - A. Tahmasbi, F. Saki, S. B. Shokouhi,
% Classification of Benign and Malignant Masses Based on Zernike Moments,
% J. Computers in Biology and Medicine, vol. 41, no. 8, pp. 726-735, 2011.
%
% [2] - A. Tahmasbi, F. Saki, H. Aghapanah, S. B. Shokouhi,
% A Novel Breast Mass Diagnosis System based on Zernike Moments as Shape and
% Density Descriptors,
% in Proc. IEEE, 18th Iranian Conf. on Biomedical Engineering (ICBME'2011),
% Tehran, Iran, 2011, pp. 100-104.
% -----
% Function to compute Zernike Polynomials:
%
% f = radialpoly(r,n,m)
% where
% r = radius
% n = the order of Zernike polynomial
% m = the repetition of Zernike moment

rad = zeros(size(r));          % Initilization
for s = 0:(n-abs(m))/2
c = (-1)^s*factorial(n-s)/(factorial(s)*factorial((n+abs(m))/2-s)*factorial((n-abs(m))/2-s));
rad = rad + c*r.^(n-2*s);
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* Testfull */
function varargout = testfull(varargin)
% TESTFULL MATLAB code for testfull.fig
%   TESTFULL, by itself, creates a new TESTFULL or raises the existing
%   singleton*.
%
%   H = TESTFULL returns the handle to a new TESTFULL or the handle to
%   the existing singleton*.
%
%   TESTFULL('CALLBACK',hObject,eventData,handles,...) calls the local
%   function named CALLBACK in TESTFULL.M with the given input arguments.
%
%   TESTFULL('Property','Value',...) creates a new TESTFULL or raises the
%   existing singleton*. Starting from the left, property value pairs are
%   applied to the GUI before testfull_OpeningFcn gets called. An
%   unrecognized property name or invalid value makes property application
%   stop. All inputs are passed to testfull_OpeningFcn via varargin.
%
%   *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
%   instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help testfull

% Last Modified by GUIDE v2.5 02-Jul-2013 18:25:13
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @testfull_OpeningFcn, ...
                  'gui_OutputFcn',  @testfull_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before testfull is made visible.
function testfull_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to testfull (see VARARGIN)
% Choose default command line output for testfull
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes testfull wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = testfull_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

%%
[FileName,PathName] = uigetfile({'*.avi','Movie'},'Open Movie');
%%
if FileName
    %movieout = aviread([PathName,FileName]);
xyloObj = mmreader([PathName,FileName]);
nFrames = xyloObj.NumberOfFrames;
vidHeight = xyloObj.Height;
vidWidth = xyloObj.Width;
%%
% Preallocate movie structure.
mov(1:nFrames) = ...
struct('cdata', zeros(vidHeight, vidWidth, 3, 'uint8'),...
    'colormap', []);
%%
% Read one frame at a time.

for k = 1 : nFrames-2
    mov(k).cdata = read(xyloObj, k);
    c1 = uint8(mov(k).cdata);
    I = imresize(c1,[256 256]);
    axes(handles.axes1); % คำสั่งให้รูป ไปอยู่ในกรอบที่สร้างไว้
    imshow(I); % คำสั่งให้รูป ไปอยู่ในกรอบที่สร้างไว้
    handles.I = I; % คำสั่งให้รูป ไปอยู่ในกรอบที่สร้างไว้
    %
    ConvertHaarcasadeXMLOpenCV('HaarCascades/haarcascade_frontalface_alt.xml');
    I1 = I;
    FilenameHaarcasade = 'HaarCascades/haarcascade_frontalface_alt.mat';
    Objects=ObjectDetection(I1,FilenameHaarcasade)
    %ShowDetectionResult(I1,Objects);
    %[x1, x2, y1, y2] = ShowDetectionResult(I1,Objects);
    size(Objects,1)
    for n=1:size(Objects,1)
        x1=Objects(n,1)
        y1=Objects(n,2)
        x2=x1+Objects(n,3)
        y2=y1+Objects(n,4)

        %plot([x1 x1 x2 x2 x1],[y1 y2 y2 y1 y1]);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

end  
end  
end



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* Testhaar */
clc,

img_orig = imread('D:/2.jpg');
height = size(img_orig,1);
width = size(img_orig,2);

%[height width] = size(img_orig);

%Initialize the output images
out = img_orig;
bin = zeros(height,width);

%Apply Grayworld Algorithm for illumination compensation
img = grayworld(img_orig);
figure, imshow(img), impixelinfo

img_ycbcr = rgb2ycbcr(img);

Cb = img_ycbcr(:,:,2);
figure, imshow(Cb), impixelinfo
Cr = img_ycbcr(:,:,3);
figure, imshow(Cr), impixelinfo

[r,c,v] = find(Cb>=139 & Cb<=145 & Cr>=110 & Cr<=120);
numind = size(r,1);

%Mark Skin Pixels
for i=1:numind
    out(r(i),c(i),:) = [0 0 255];
    bin(r(i),c(i)) = 1;
end
imshow(img_orig);
figure; imshow(out);
figure; imshow(bin);

ConvertHaarcasadeXMLOpenCV('HaarCascades/haarcascade_frontalface_alt.xml');

```

```
I = bin;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FilenameHaarcasade = 'HaarCascades/haarcascade_frontalface_alt.mat';  
Objects=ObjectDetection(I,FilenameHaarcasade);  
ShowDetectionResult(I,Objects);
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* Testhaaraddcolor */
clc,
%I = imread('D:/images.JPG');
%I = rgb2gray(im);
    %path = 'D:\8. Ph.D KU\code\face\New folder
(3)\ViolaJones_version\testvideo\test';
%subname = '.jpg';
%for iimage = 1 : 298

% iimage = num2str(iimage);
% imagename = strcat(path,iimage,subname);
I = imresize(imread('Images/9.jpg'), [512 512]);
ConvertHaarcascadeXMLOpenCV('HaarCascades/haarcascade_frontalface_alt.xml');
I1 = I;
FilenameHaarcascade = 'HaarCascades/haarcascade_frontalface_alt.mat';
Objects=ObjectDetection(I1,FilenameHaarcascade);
ShowDetectionResult(I1,Objects);
[x1, x2, y1, y2] = ShowDetectionResult(I1,Objects);
%
I_roi = I1(y1:y2,x1:x2);
figure, imshow(I_roi)
%
% I1(y1:y2,x1:x2,:) = 0;
% figure, imshow(I1)

%imwrite(I1, strcat('IMG_9006', '.jpg'));
%end
%%
%function [M]=feature_vec(A)
A = I_roi;

% First Moment
n20=cent_moment(2,0,A);
n02=cent_moment(0,2,A);
M1=n20+n02;

% Second Moment

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
n20=cent_moment(2,0,A);
n02=cent_moment(0,2,A);
n11=cent_moment(1,1,A);
M2=(n20-n02)^2+4*n11^2;
```

```
% Third Moment
```

```
n30=cent_moment(3,0,A);
n12=cent_moment(1,2,A);
n21=cent_moment(2,1,A);
n03=cent_moment(0,3,A);
M3=(n30-3*n12)^2+(3*n21-n03)^2;
```

```
% Fourth Moment
```

```
n30=cent_moment(3,0,A);
n12=cent_moment(1,2,A);
n21=cent_moment(2,1,A);
n03=cent_moment(0,3,A);
M4=(n30+n12)^2+(n21+n03)^2;
```

```
% Fifth Moment
```

```
n30=cent_moment(3,0,A);
n12=cent_moment(1,2,A);
n21=cent_moment(2,1,A);
n03=cent_moment(0,3,A);
M5=(n30-3*n21)*(n30+n12)*[(n30+n12)^2-3*(n21+n03)^2]+(3*n21-
n03)*(n21+n03)*[3*(n30+n12)^2-(n21+n03)^2];
```

```
% Sixth Moment
```

```
n20=cent_moment(2,0,A);
n02=cent_moment(0,2,A);
n30=cent_moment(3,0,A);
n12=cent_moment(1,2,A);
n21=cent_moment(2,1,A);
n03=cent_moment(0,3,A);
n11=cent_moment(1,1,A);
M6=(n20-n02)*[(n30+n12)^2-(n21+n03)^2]+4*n11*(n30+n12)*(n21+n03);
```

```
% Seventh Moment
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

n30=cent_moment(3,0,A);
n12=cent_moment(1,2,A);
n21=cent_moment(2,1,A);
n03=cent_moment(0,3,A);
M7=(3*n21-n03)*(n30+n12)*[(n30+n12)^2-3*(n21+n03)^2]-
(n30+3*n12)*(n21+n03)*[3*(n30+n12)^2-(n21+n03)^2];

```

```

% The vector M is a column vector containing M1,M2,...M7

```

```

MM=[M1 M2 M3 M4 M5 M6 M7]';

```

```

%and this is the Feature vector

```

```

%% gskewness

```

```

x = I_roi;
x = imresize(x, [196 196]);
mc = x(:,1:98); %class mark
f = x(:,99:196); %absolut frequency
s = sum(f.*mc);
m1 = s/sum(f);
m2 = sum(f.*(mc - m1).^2)/sum(f);
m3 = sum(f.*(mc - m1).^3)/sum(f);
g1 = m3/m2^1.5;
n=1;
if n == 0;
    y = sqrt(sum(f)*(sum(f) - 1)/(sum(f) - 2) * g1; %skewness adjusted for
        %bias
else n = 1; %default
    y = g1; %skewness not adjusted for bias
end
%%
p = sum(mean(I_roi));
n = 7;
m = 1;

```

```

% Example:

```

```

% 1- calculate the Zernike moment (n,m) for an oval shape,

```

```

% 2- rotate the oval shape around its centroid,

```

```

% 3- calculate the Zernike moment (n,m) again,

```

```

% 4- the amplitude of the moment (A) should be the same for both images

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่หรือใช้  
 ใ้ในทางอื่นใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

% 5- the phase (Phi) should be equal to the angle of rotation

```

N = size(p,1);
x = 1:N; y = x;
[X,Y] = meshgrid(x,y);
R = sqrt((2.*X-N-1).^2+(2.*Y-N-1).^2)/N;
Theta = atan2((N-1-2.*Y+2),(2.*X-N+1-2));
R = (R<=1).*R;
Rad = radialpoly(R,n,m); % get the radial polynomial

Product = p(x,y).*Rad.*exp(-1i*m*Theta);
Z = sum(Product(:)); % calculate the moments

cnt = nnz(R)+1; % count the number of pixels inside the unit circle
Z = (n+1)*Z/cnt; % normalize the amplitude of moments
A = abs(Z); % calculate the amplitude of the moment
Phi = angle(Z)*180/pi; % calculate the phase of the mement (in degrees)
%% Correlation coefficients
Cor_test = var(sum(l_roi));
%%
std_test = std2(l_roi);
%%
x = sum(l_roi);
filterSize = 400;
termIter = 30;
T = 100;
M = 7;
plotMode = 1;

```

% Force x into a column vector

```
x = x(:);
```

```
L = filterSize;
```

```
OriginalLength = length(x);
```

% Perform a resampling of x to an integer period if required

```
if( abs(round(T) - T) > 0.01 )
```

```
    % We need to resample x to an integer period
```

```
    T_new = ceil(T);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

% The rate transformation factor
Factor = 20;

% Calculate the resample factor
P = round(T_new * Factor);
Q = round(T * Factor);
Common = gcd(P,Q);
P = P / Common;
Q = Q / Common;

% Resample the input
x = resample(x, P, Q);
T = T_new;
else
    T = round(T);
end

N = length(x);

% Calculate XmT
XmT = zeros(L,N,M+1);
for( m = 0:M)
for( l = 1:L )
if( l == 1 )
XmT(l,(m*T+1):end,m+1) = x(1:N-m*T);
else
XmT(l, 2:end,m+1) = XmT(l-1, 1:end-1,m+1);
end
end
end

% Calculate the matrix inverse section
Xinv = inv(XmT(:,1)*XmT(:,1)');

% Assume initial filter as a delayed impulse
f = zeros(L,1);
f(round(L/2)) = 1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

f(round(L/2)+1) = -1;
f_best = f;
ck_best = 0;
iter_best = 0;

% Initialize iteration matrices
y = zeros(N,1);
b = zeros(L,1);
ckIter = [];

% Iteratively adjust the filter to minimize entropy
n = 1;
delta = 0;
while n == 1 || ( n <= termIter )
    % Compute output signal
    y = (f*XmT(:,1));

    % Generate yt
    yt = zeros(N,M);
    for( m = 0:M )
        if( m == 0 )
            yt(:,m+1) = y;
        else
            yt(T+1:end,m+1) = yt(1:end-T,m);
        end
    end

    % Calculate alpha
    alpha = zeros(N,M+1);
    for( m = 0:M )
        alpha(:,m+1) = (prod(yt(:,[1:m (m+2):size(yt,2)]),2).^2).*yt(:,m+1);
    end

    % Calculate beta
    beta = prod(yt,2);

    % Calculate the sum Xalpha term
    Xalpha = zeros(L,1);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for( m = 0:M )
    Xalpha = Xalpha + XmT(:,m+1)*alpha(:,m+1);
end

% Calculate the new filter coefficients
f = sum(y.^2) / (2*sum(beta.^2)) * Xinvs * Xalpha;

f = f/sqrt(sum(f.^2));

% Calculate the ck value fo this iteration
cklter(n) = sum(prod(yt,2).^2)/( sum(y.^2)^(M+1) );

% Update the best match filter if required
if( cklter(n) > ck_best )
    ck_best = cklter(n);
    f_best = f;
    iter_best = n;
end

n = n + 1;

end

% Update the final result
f_final = f_best;
y_final = filter(f_final,1,x);

% Resample the final result
if( length(y_final) ~= OriginalLength )
    y_final = y_final(1:OriginalLength);
end

aa = sum(mean(y_final));
%%
test_data = [MM' Z A Phi Cor_test std_test aa];
%%
data = [0 0 0 0 0 0 0 0 0 0 0 255 82 0
         0 0 0 0 0 0 0 0 0 0 0 255 67 231

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

0 0 0 0 0 0 0 0 0 0 0 255 82 0
0 0 0 0 0 0 0 0 0 0 0 255 83 0
0 0 0 0 0 0 0 0 0 0 0 255 83 0
0 0 0 0 0 0 0 0 0 0 0 255 64 185
0 0 0 0 0 0 0 0 0 0 0 255 86 0
0 0 0 0 0 0 0 0 0 0 0 255 61 0
0 0 0 0 0 0 0 0 0 0 0 255 63 0];

p = data;
t = [0 0 0 0 1 1 1 1 1];
% [pn,ps] = mapminmax(p);
% [tn,ts] = mapminmax(t);
% %%
% net=newff(pn,tn,[13 15 5],{'tansig' 'tansig' 'purelin'},'trainrp');
%
% %%
% net.trainParam.show = 10;
% net.trainParam.epochs = 300;
% net.trainParam.goal = 1e-5;
% %% training
% %net = newp(p,t);
% net=train(net,pn,tn);
%
% %% simulation
% y1=sim(net,test_data);
% y = mapminmax('reverse',y1,ts);
%% K-NN
k=1;
dataMatrix = test_data; % เอา target ของ NN มาใส่
queryMatrix = t'; % เอา o/p ของ NN มาใส่

neighborIds = zeros(size(queryMatrix,1),k);
neighborDistances = neighborIds;
numDataVectors = size(dataMatrix,1);
numQueryVectors = size(queryMatrix,1);

for i=1:numQueryVectors,
    dist = sum((repmat(queryMatrix(i,:),numDataVectors,1)-dataMatrix).^2,2);
    [sortval sortpos] = sort(dist,'ascend');

```

```

neighborIds(i,:) = sortpos(1:k);
neighborDistances(i,:) = sqrt(sortval(1:k));
end
%%
[Euc_dist_min, Recognized_index] = min(neighborDistances);
b = Recognized_index;

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* TestMain */
clear all
clc
close all

% You can customize and fix initial directory paths
TrainDatabasePath = uigetdir('D:\Program Files\MATLAB\R2006a\work', 'Please select
training database' );
TestDatabasePath = uigetdir('D:\Program Files\MATLAB\R2006a\work', 'Please select
test database');

prompt = {'Enter to test image name (a number between 1 to 100):'};
dlg_title = 'Input of FaceEigen-Based Face Recognition System';
num_lines= 1;
def = {'1'};

TestImage = inputdlg(prompt,dlg_title,num_lines,def);
TestImage = strcat(TestDatabasePath,'\char(TestImage),'.jpg');
%TestImage = histeq(TestImage);
im = imread(TestImage);
im = imresize(im,[196 196]);

T = CreateDatabase(TrainDatabasePath);

[m, A, Eigenfaces] = Eigenface(T);
[Recognized_index, ProjectedTestImage, OutputName, q, Euc_dist] =
Euclidean_distance(TestImage, m, A, Eigenfaces);

SelectedImage = strcat(TrainDatabasePath,'\OutputName);
SelectedImage = imread(SelectedImage);

% figure,imshow(im);
% title('Equivalent Image');
/* Untitled */
clc,
close all,
clear all,
%%

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

[FileName,PathName] = uigetfile({'*.avi','Movie'},'Open Movie');
%%
if FileName
    %movieout = aviread([PathName,FileName]);
xyloObj = mmreader([PathName,FileName]);
nFrames = xyloObj.NumberOfFrames;
vidHeight = xyloObj.Height;
vidWidth = xyloObj.Width;
%%
% Preallocate movie structure.
mov(1:nFrames) = ...
    struct('cdata', zeros(vidHeight, vidWidth, 3, 'uint8'),...
        'colormap', []);
%%
% Read one frame at a time.

for k = 1 : nFrames-2
    kstr = num2str(k);
    mov(k).cdata = read(xyloObj, k);
    c1 = uint8(mov(k).cdata);
    l = imresize(c1,[700 700]);
    imwrite(l, strcat('test',kstr, '.jpg'));
    %plot([x1 x1 x2 x2 x1],[y1 y2 y2 y1 y1]);
end
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* Zernike_main */
% -----
% Copyright C 2012 Amir Tahmasbi
% The University of Texas at Dallas
% a.tahmasbi@utdallas.edu
% http://www.utdallas.edu/~a.tahmasbi/research.html
%
% License Agreement: You are free to use this code in your scientific
%         research but you should cite the following papers:
%
% [1] - A. Tahmasbi, F. Saki, S. B. Shokouhi,
%       Classification of Benign and Malignant Masses Based on Zernike Moments,
%       J. Computers in Biology and Medicine, vol. 41, no. 8, pp. 726-735, 2011.
%
% [2] - A. Tahmasbi, F. Saki, H. Aghapanah, S. B. Shokouhi,
%       A Novel Breast Mass Diagnosis System based on Zernike Moments as Shape and
%       Density Descriptors,
%       in Proc. IEEE, 18th Iranian Conf. on Biomedical Engineering (ICBME'2011),
%       Tehran, Iran, 2011, pp. 100-104.
% -----
% A demo of how to use the Zernike moment function.
%
% Example:
% 1- calculate the Zernike moment (n,m) for an oval shape,
% 2- rotate the oval shape around its centeroid,
% 3- calculate the Zernike moment (n,m) again,
% 4- the amplitude of the moment (A) should be the same for both images
% 5- the phase (Phi) should be equal to the angle of rotation

clc; clear all; close all;

n = 4; m = 2;          % Define the order and the repetition of the moment

disp('-----');
disp(['Calculating Zernike moments ..., n = ' num2str(n) ', m = ' num2str(m)]);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

% row 1
p = rgb2gray(imread('Oval_H.png'));
figure(1);subplot(2,3,1);imshow(p);
title('Horizontal oval');
p = logical(not(p));
tic
[~, AOH, PhiOH] = Zernikmoment(p,n,m);    % Call Zernikemoment fuction
Elapsed_time = toc;
xlabel({'A = ' num2str(AOH)}; ['\phi = ' num2str(PhiOH)}]);

p = rgb2gray(imread('Oval_45.png'));
figure(1);subplot(2,3,2);imshow(p);
title('-45 degree oval');
p = logical(not(p));
[~, AOH, PhiOH] = Zernikmoment(p,n,m);    % Call Zernikemoment fuction
xlabel({'A = ' num2str(AOH)}; ['\phi = ' num2str(PhiOH)}]);

p = rgb2gray(imread('Oval_V.png'));
figure(1);subplot(2,3,3);imshow(p);
title('Vertical oval');
p = logical(not(p));
[~, AOH, PhiOH] = Zernikmoment(p,n,m);    % Call Zernikemoment fuction
xlabel({'A = ' num2str(AOH)}; ['\phi = ' num2str(PhiOH)}]);

%-----
% row 2
p = rgb2gray(imread('shape_0.png'));
figure(1);subplot(2,3,4);imshow(p);
title('Horizontal shape');
p = logical(not(p));
[~, AOH, PhiOH] = Zernikmoment(p,n,m);    % Call Zernikemoment fuction
xlabel({'A = ' num2str(AOH)}; ['\phi = ' num2str(PhiOH)}]);

p = rgb2gray(imread('shape_90.png'));
figure(1);subplot(2,3,5);imshow(p);
title('Vertical shape');
p = logical(not(p));
[~, AOV, PhiOV] = Zernikmoment(p,n,m);    % Call Zernikemoment fuction

```

```

xlabel(['A = ' num2str(AOV)]; ['\phi = ' num2str(PhiOV)]);

p = rgb2gray(imread('Rectangular_H.png'));
figure(1);subplot(2,3,6);imshow(p);
title('Horizontal Rectangle');
p = logical(not(p));
[~, AOH, PhiOH] = Zernikmoment(p,n,m);    % Call Zernikemoment fuction
xlabel(['A = ' num2str(AOH)]; ['\phi = ' num2str(PhiOH)]);

%-----
% show the elapsed time
disp('Calculation is complete. ');
disp(['The elapsed time per image is ' num2str(Elapsed_time) ' seconds']);

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ประวัติผู้เขียน

ชื่อ	น.ต.พิศณุ คุมีชัย ร.น.
เกิดวันที่	10 ธ.ค. 2523 ที่จังหวัดกรุงเทพมหานครฯ
การศึกษาก่อนเข้ารับราชการ	ปี 2542 คุณวุฒิ ม. 6 ของ รร.เตรียมทหาร ปี 2547 คุณวุฒิปริญญาตรี ของ รร.นายเรือ
รายการแสดงการรับราชการ	ปี 2548 นายทหารฝึกโรงเรียนชุมพลทหารเรือ ปี 2549 นายทหารฝึกโรงเรียนชุมพลทหารเรือ ปี 2550 นายทหารสื่อสารเรือหลวงนาคา ปี 2551 นายทหารประจำกองเรือยุทธการ ปี 2555 ครู มว.ตอร์ปิโด แผนกการอาวุธใต้น้ำ กฝอ.กฝร. ปี 2557 หัวหน้า มว.ตอร์ปิโด แผนกการอาวุธใต้น้ำ กฝอ.กฝร.
ปัจจุบัน	หัวหน้า มว.ตอร์ปิโด แผนกการอาวุธใต้น้ำ กฝอ.กฝร.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้