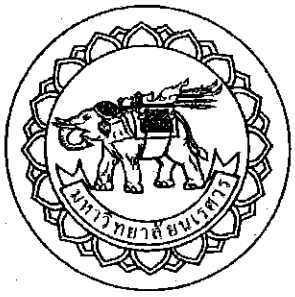


อัลกอริทึมในการตรวจวัดความผิดปกติของลูกตา

Algorithm in False Detection for Eye Image

นางสาวมุกดรินทร์ พิมพร รหัส 48361714

นางสาวสิริพร หว่างสิงห์ รหัส 48361967

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ...../...../.....
เลขทะเบียน 5200025
เลขเรียกหนังสือ ว.ป. ม.๒๖๔.ค
มหาวิทยาลัย 2551

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2551

ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน	อัลกอริทึมในการตรวจวัดความผิดปกติของลูกตา
ผู้ดำเนินโครงงาน	นางสาวมุกดรินทร์ พิมพร รหัส 48361714
	นางสาวสิริพร หว่างสิงห์ รหัส 48361967
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.สุชาติ แย้มเม่น
สาขาวิชา	วิศวกรรมไฟฟ้า
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2551

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า

คณะกรรมการสอบโครงงานวิศวกรรม

..... ประธานกรรมการ

(ผู้ช่วยศาสตราจารย์ ดร.สุชาติ แย้มเม่น)

..... กรรมการ

(ดร.นิพัทธ์ จันทรมินทร์)

..... กรรมการ

(อาจารย์ศิริพร เดชะศิลารักษ์)

หัวข้อโครงงาน	อัลกอริทึมในการตรวจวัดความผิดปกติของลูกตา
ผู้ดำเนินโครงงาน	นางสาวมุกดรินทร์ พิมพร รหัส 48361714
	นางสาวสิริพร หว่างสิงห์ รหัส 48361967
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.สุชาติ แย้มเม่น
สาขาวิชา	วิศวกรรมไฟฟ้า
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2551

บทคัดย่อ

โครงงานนี้เป็นการพัฒนาอัลกอริทึมในการตรวจวัดความผิดปกติลูกตาจากภาพถ่ายจอประสาทของมนุษย์โดยการใช้การประมวลผลภาพซึ่งมีการดำเนินงานอยู่ 3 ขั้นตอน คือ (1) แปลงภาพถ่ายจอประสาทแบบสีให้เป็นภาพระดับเทา (2) ทำการไดเลชัน (dilation) ภาพระดับเทาให้บริเวณส่วนผิดปกติเด่นชัดขึ้น และ (3) นำผลของภาพมาทำการหาขอบภาพเพื่อระบุบริเวณที่ผิดปกติของลูกตาในรูปแบบภาพขาวดำพร้อมกับนำภาพที่ได้รับไปคำนวณหาระดับความผิดปกติของลูกตาในรูปแบบ %Score

จากผลการทดลองและวิเคราะห์ผลการทดลอง อัลกอริทึมในการระบุความผิดพลาดของลูกตากับภาพลูกตา จำนวน 40 ภาพ ซึ่งแบ่งเป็นภาพลูกตาปกติ จำนวน 16 ภาพและภาพลูกตาผิดปกติ จำนวน 24 ภาพ สรุปได้ว่า ภาพลูกตาที่มีค่า %Score น้อยกว่า 50% จะจัดอยู่ในกลุ่มของภาพที่ปกติและภาพลูกตาที่มีค่า %Score มากกว่า 50% จะจัดอยู่ในกลุ่มของภาพที่ผิดปกติ โดยการใช้ภาพหน่วยโครงสร้างในกระบวนการไดเลชันที่มีขนาดเท่ากับ 15 พิกเซล ทำให้ได้ค่า %Score ในการทำนายผลความผิดปกติของลูกตาที่เหมาะสมเมื่อมีสิ่งรบกวนเล็กน้อยและพิกเซลสีขาวจะถูกขยายในระดับที่มีค่าความผิดพลาด (%Error) ในการทำนายอาการผิดปกติของลูกตาจริงเท่ากับ 10%

Project Title	Algorithm in False Detection for Eye Image
Name	M s. Mukdarin Pimporn ID. 48361714
	M s. Siriporn Wangsing ID. 48361967
Project Advisor	Assistant professor Suchart Yammen, Ph.D.
Major	Electrical Engineering
Department	Electrical and Computer Engineering
Academic Year	2008

ABSTRACT

This project is to develop an algorithm in false detection for eye image by using the three following image processing steps: (1) transform those RGB images to their gray-scale eye images. (2) apply the gray-sacle eye images to dilation operator for enhancing to indentify abnormal region of the images obviously and (3) use images obtained from the dilation process to Robert operator for indentifying abnormal region segmentation in term of the binary images as well as calculate the percentage of the abnormal eye image in term of %Score.

From the experimental result and analysis, the developed algorithm in false detection for eye image is tested with 40 eye images that consist of 16 normal-eye images and 24 abnormal-eye images. In conclusion, by using the size of the structuring element 15 pixel in the dilation process all eye images whose %Score are less than or equal to 50% will be arranged in the normal eye groups, and all eye images whose %Score are greater than 50% will be arranged in the abnormal eye groups. This leads to the optimum %Score value for 10% error in the false detection prediction of the eye images corrupted with a little additive noise.

กิตติกรรมประกาศ

การที่โครงงานนี้สำเร็จลุล่วงไปได้ด้วยดีทางผู้จัดทำใคร่ขอแสดงความขอบพระคุณเป็นอย่างสูงต่ออาจารย์ที่ปรึกษาโครงงาน ผู้ช่วยศาสตรจารย์ ดร.สุชาติ แย้มเม่น ดร.นิพัทธ์ จันทรมินทร์ และอาจารย์ศิริพร เดชะศิลารักษ์ อาจารย์ประจำภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร ที่ได้กรุณาให้แนวความคิด ช่วยชี้แนะแนวทางในการทำโครงงาน ตลอดจนกรุณาเอื้อเฟื้อเอกสารต่างๆที่เกี่ยวข้องกับโครงงานนี้ อีกทั้งยังช่วยแนะนำแหล่งข้อมูลในการค้นคว้าข้อมูลเพิ่มเติม ทำให้เป็นประโยชน์ต่อการทำโครงงานของผู้จัดทำ

นางสาวมุกดรินทร์ พิมพร

นางสาวสิริพร หว่างสิงห์

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ก

บทคัดย่อภาษาอังกฤษ ..... ข

กิตติกรรมประกาศ ..... ค

สารบัญ ..... ง

สารบัญ(ต่อ) ..... จ

สารบัญตาราง ..... ฉ

สารบัญรูป ..... ช

สารบัญรูป (ต่อ) ..... ซ

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญของโครงงาน 1

1.2 วัตถุประสงค์ของโครงงาน 1

1.3 ขอบเขตของโครงงาน 1

1.4 ขั้นตอนและแผนการดำเนินงาน 2

1.5 ประโยชน์ที่คาดว่าจะได้รับจากโครงงาน 2

1.6 งบประมาณ 3

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

2.1 ความรู้เบื้องต้นเกี่ยวกับโรคเบาหวาน 4

2.2 ภาพดิจิตอล (Digital Image) 7

2.3 มาตรฐานของสี (Color Standard) 9

2.4 ทฤษฎีการแปลงภาพระดับเทา (Gray Scale) 10

2.5 ทฤษฎีการขยายพิกเซลของภาพ (Dilation Theory) 10

2.6 ทฤษฎีการหาขอบ (Edge Detection Theory) 12

2.7 กระบวนการวัดทางสถิติ (Statistical Operations) 14

บทที่ 3 ขั้นตอนการดำเนินการ

3.1 การเตรียมข้อมูลภาพสำหรับกระบวนการการทดลอง 19

3.2 การอ่านไฟล์ภาพเข้ามาในโปรแกรม 20

สารบัญ (ต่อ)

หน้า

3.3 การปรับขนาดและการแปลงภาพสีให้เป็นภาพระดับเทา 20

3.4 การนำภาพอินพุตมาหาภาพ Mark 21

3.5 การหาค่า Score 23

3.6 การหาค่า %Score 24

บทที่ 4 ผลการทดลองและวิเคราะห์ผลการทดลอง

4.1 การทดลองของปรับค่า Structuring element ทั้ง 3 กรณี 25

4.2 วิเคราะห์ผลการทดลอง 37

บทที่ 5 สรุปผลการทดลอง

5.1 สรุปผลการทดลอง 39

5.2 ปัญหาและอุปสรรคของโครงงาน 39

5.3 ข้อเสนอแนะและวิธีการแก้ไข 40

บรรณานุกรม 41

ภาคผนวก

ภาคผนวก ก 44

ภาคผนวก ข 50

ประวัติผู้เขียนโครงงาน 56

สารบัญตาราง

ตารางที่	หน้า
1.1 ขั้นตอนการดำเนินงานและแผนการดำเนินงานตลอดโครงการวิจัย	2
2.1 ตารางแสดงค่าในภาพระดับเทา	15
2.2 ตารางแสดงค่าในภาพขาวดำหลังการทำเทร็ชโฮลด์ (Threshold)	15
4.1 สรุปค่า Score และค่า %Score ที่ Structuring element = 10×10 พิกเซล	28
4.1 (ต่อ) สรุปค่า Score และค่า %Score ที่ Structuring element = 10×10 พิกเซล	29
4.2 สรุปค่า Score และค่า %Score ที่ Structuring element = 15×15 พิกเซล	32
4.2 (ต่อ) สรุปค่า Score และค่า %Score ที่ Structuring element = 15×15 พิกเซล	33
4.3 สรุปค่า Score และค่า %Score ที่ Structuring element = 20×20 พิกเซล	36
4.3 (ต่อ) สรุปค่า Score และค่า %Score ที่ Structuring element = 20×20 พิกเซล	37

สารบัญรูป

รูปที่	หน้า
2.1 รูปตาที่ไม่ปกติที่เกิดจากโรคเบาหวานขึ้นจอประสาทตา	6
2.2 แสดงมาตรฐานสี	9
2.3 ตัวอย่างการไดเลชันระหว่างภาพ A กับภาพหน่วยโครงสร้าง B = {1,1,1}	11
2.4 ตัวอย่างการไดเลชันระหว่างภาพ A กับภาพหน่วยโครงสร้าง B = {1,3,2}	11
2.6 เทมเพลตของ s และ t	13
2.7 ตัวอย่างการหาขอบด้วยวิธีของโรเบิรตส์	13
3.1 ขั้นตอนการดำเนินงาน	18
3.2 รูปจอประสาทตาที่ผิดปกติ	19
3.3 รูปจอประสาทตาทีปกติ	20
3.4 ตัวอย่างภาพระดับเทา	21
3.5 ตัวอย่างภาพที่ผ่านการทำ Dilation	22
3.6 ตัวอย่างภาพต้นแบบและภาพ Mark	22
3.7 ภาพ Mark และ ภาพ Mask	25
4.1 ตัวอย่างภาพที่มีค่า Score = 0.003855, Percent Score = 23.744912	26
4.2 ตัวอย่างภาพที่มีค่า Score = 0.008832, Percent Score = 54.396639	26
4.3 ตัวอย่างภาพที่มีค่า Score = 0.012618 และ %Score = 77.721364	26
4.4 ตัวอย่างภาพที่มีค่า Score = 0.015047, Percent Score = 92.681753	26
4.5 ภาพผลการทดลองทั้งหมดที่ค่า Structuring element 10×10 พิกเซล	27
4.6 ตัวอย่างภาพที่มีค่า Score = 0.004232, Percent Score = 34.170970	29
4.7 ตัวอย่างภาพที่มีค่า Score = 0.006475, Percent Score = 52.277682	30
4.8 ตัวอย่างภาพที่มีค่า Score = 0.008714 และ %Score = 70.349971	30
4.9 ตัวอย่างภาพที่มีค่า Score = 0.010135, Percent Score = 81.830178	30
4.10 ภาพผลการทดลองทั้งหมดโดยที่ค่า Structuring element 15×15 พิกเซล	31
4.11 ตัวอย่างภาพที่มีค่า Score = 0.005499และ %Score = 37.785048	34
4.12 ตัวอย่างภาพที่มีค่า Score = 0.008510, Percent Score = 58.479418	34

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.13	ตัวอย่างภาพที่มีค่า Score = 0.009190, Percent Score = 63.147615	34
4.14	ตัวอย่างภาพที่มีค่า Score = 0.010708, Percent Score = 73.577811	34
4.12	ภาพผลการทดลองทั้งหมดโดยที่ค่า Structuring element 20×20 พิกเซล	35

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

เนื่องจากในปัจจุบัน มีผู้ที่ป่วยเป็นโรคเบาหวานขึ้นจอประสาทตา เพิ่มมากขึ้นอย่างรวดเร็ว ส่งผลให้แพทย์ต้องใช้เวลากับผู้ป่วยโรคเบาหวานมากขึ้น และค่าใช้จ่ายในการวินิจฉัยโรคก็เพิ่มมากขึ้นอีกด้วย เนื่องจากขั้นตอนในการวินิจฉัยค่อนข้างยุ่งยากและใช้เวลานาน ดังนั้นจึงได้จัดทำโครงการนี้ขึ้นเพื่อให้เป็นส่วนหนึ่งในการตรวจหาความผิดปกติที่แสดงออก ซึ่งเป็นส่วนหนึ่งอาการของโรคเบาหวานขึ้นจอประสาทตา เพื่อช่วยในการพัฒนางานวิจัยเกี่ยวกับโรคเบาหวานขึ้นจอประสาทตาในขั้นต่อไป

โครงการนี้ มุ่งเน้นเกี่ยวกับการนำภาพลูกตามาวิเคราะห์ หาระดับความผิดปกติของอาการที่แสดงออกผ่านทางลูกตาของผู้ป่วยโรคเบาหวานขึ้นจอประสาทตา และประมวลผลออกมาเป็นเปอร์เซ็นต์ความรุนแรงของอาการป่วย ซึ่งเป็นการวิเคราะห์ด้วยการเปรียบเทียบระดับอาการภายในฐานข้อมูล โดยการใช้วิธีการประมวลผลภาพ

1.2 วัตถุประสงค์ของโครงงาน

เพื่อศึกษา วิเคราะห์ ออกแบบและพัฒนาอัลกอริทึมในการหาระดับความผิดปกติของอาการที่แสดงออกผ่านทางลูกตาของผู้ป่วยโรคเบาหวานขึ้นจอประสาทตา พร้อมกับประมวลผลออกมาในรูปแบบเปอร์เซ็นต์ความรุนแรงของอาการป่วย (%score) เมื่อเปรียบเทียบระดับอาการผิดปกติของภาพลูกตาในฐานข้อมูลด้วยวิธีการประมวลผลภาพ

1.3 ขอบเขตของโครงงาน

การประมวลผลหาระดับความผิดปกติของอาการที่แสดงออกผ่านทางลูกตาของผู้ป่วยโรคเบาหวานขึ้นจอประสาทตา โดยการใช้วิธีการประมาวลผลภาพ โครงงานนี้ จะใช้ภาพลูกตาในการทดสอบ จำนวน 40 ภาพ ซึ่งแบ่งเป็นภาพลูกตาปกติ จำนวน 16 ภาพและภาพลูกตาผิดปกติ จำนวน 24 ภาพ

1.4 ขั้นตอนและแผนการดำเนินงาน

เพื่อในบรรลุวัตถุประสงค์ในการศึกษา การวิเคราะห์ การออกแบบและการพัฒนาอัลกอริทึมในการหาระดับความผิดปกติของอาการที่แสดงออกผ่านทางลูกตาของผู้ป่วยโรคเบาหวานขึ้นจอประสาทตาจากภาพถ่ายดิจิตัลพร้อมกับทดสอบประสิทธิภาพของอัลกอริทึมที่พัฒนาขึ้น จึงจำเป็นต้องมีการวางแผนและขั้นตอนการดำเนินงานต่างๆ ในโครงงาน จำนวน 5 ขั้น ดังสรุปรายละเอียดขั้นตอนและระยะเวลาดำเนินงานแต่ละขั้นตอนไว้ในตารางที่ 1.1

ตารางที่ 1.1 ขั้นตอนการดำเนินงานและแผนการดำเนินงานตลอดโครงการวิจัย

รายการ	ปี 2551							ปี 2552	
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.
1. ศึกษาข้อมูลที่เกี่ยวข้อง	←	→							
2. ศึกษาทฤษฎีการหาขอบภาพและการไดเลชัน			←			→			
3. วิเคราะห์หาความผิดปกติของภาพจอประสาทตา				←		→			
4. พัฒนาโปรแกรมหาความผิดปกติจากภาพลูกตา						←			→
5. สรุปผลและจัดทำรายงาน								←	→

1.5 ประโยชน์ที่คาดว่าจะได้รับจากโครงงาน

1. เข้าใจหลักการและทฤษฎีการประมวลผลภาพเพิ่มขึ้น
2. เข้าใจการวิเคราะห์ลักษณะอาการผิดปกติของโรคเบาหวานจากภาพจอประสาทตาเพิ่มขึ้น
3. สร้างความร่วมมือระหว่างสถาบันการศึกษากับผู้เชี่ยวชาญในการวินิจฉัยโรคเบาหวาน
4. ลดการนำเข้าเครื่องมือและอุปกรณ์ที่เกี่ยวกับการตรวจและวินิจฉัยโรคเบาหวาน
5. สร้างเครือข่ายและความร่วมมือในวงการแพทย์ไทยในด้านอาการผิดปกติทางลูกตา
6. พัฒนาศักยภาพบุคลากรในการตรวจและวินิฉัยอาการผิดปกติจากภาพจอประสาทตา

1.6 งบประมาณ

1. ถ่ายเอกสารและค่าเข้าเล่มโครงงานฉบับสมบูรณ์	600	บาท
2. ค่าหนังสือ	400	บาท
3. อื่นๆ	1,000	บาท
รวมเป็นเงิน (สองพันบาทถ้วน)	2,000	บาท

หมายเหตุ : ถัวเฉลี่ยทุกรายการ

บทที่ 2

หลักการและทฤษฎี

2.1 ความรู้เบื้องต้นเกี่ยวกับโรคเบาหวาน

โรคเบาหวานเป็นภาวะที่ร่างกายไม่สามารถที่จะใช้ หรือเก็บน้ำตาลได้ตามปกติ ทำให้ระดับน้ำตาลในเลือดสูง เป็นผลให้มีการเปลี่ยนแปลงทางเส้นเลือดของอวัยวะต่าง ๆ รวมทั้งที่จอตา ด้วยนอกจากนี้ยังพบว่าคนที่เป็นเบาหวานมักเป็นต้อกระจกเร็วกว่าคนปกติ ซึ่งจอตาหรือจอรับภาพ ซึ่งเป็นอวัยวะอยู่ภายใน มองจากภายนอกด้วยไฟฉายไม่เห็น ทำหน้าที่รับภาพที่เห็นส่งผ่านไปยังประสาทตาแล้วไปสมอง ในคนไข้เบาหวาน สาเหตุที่จะทำให้ตามัวมากที่สุด หรือถึงขั้นตาบอดคือจากเส้นเลือดที่จอตาเกิดเปลี่ยนแปลง เปราะบางมีเลือดหรือน้ำซึมออกมา มีหลอดเลือดงอกแตกแขนงผิดปกติ โอกาสที่จะเป็นแบบนี้ได้มีถึง 60% เมื่อป่วยเป็นเบาหวานมานานกว่า 15 ปี คนที่เป็นเบาหวานมีโอกาสตาบอดได้ถึง 25 เท่าเมื่อเทียบกับคนที่ไม่ได้เป็นเบาหวาน

ในสหรัฐอเมริกา แคนาดา และประเทศในยุโรป เบาหวานขึ้นตาเป็นสาเหตุทำให้คนช่วงอายุ 40-60 ปี ตาบอดเป็นอันดับที่ 2 แต่ปัจจุบันวิวัฒนาการการรักษาจอประสาทตาด้วยเลเซอร์แพร่หลายขึ้น จึงทำให้สถิติตาบอดจากโรคนี้ลดลงเป็นอันดับที่ 4

ในประเทศไทย คาดว่ามีผู้ป่วยเบาหวานอีกจำนวนมากที่ไม่ได้รับการตรวจรักษาตาอย่างถูกต้อง และต้องสูญเสียสายตาไปเป็นจำนวนมาก

2.1.1 โรคเบาหวานขึ้นจอประสาทตา (Diabetic Retinopathy)

สาเหตุหลักของการสูญเสียการมองเห็นในผู้ป่วยโรคเบาหวาน คือ ภาวะเบาหวานขึ้นจอประสาทตา การที่ผู้ป่วยมีระดับน้ำตาลในเลือดสูง ทำให้เกิดความผิดปรกติของเส้นเลือดที่จอประสาทตา เกิดจอประสาทตาบวม เลือดออกในน้ำวุ้นตา จอประสาทตาหลุดลอก จนทำให้ตาบอดในที่สุด ดังนั้น ยิ่งผู้ป่วยเป็นโรคเบาหวานนานเท่าไหร่ โอกาสเกิดโรคเบาหวานขึ้นจอประสาทตาก็ยิ่งเพิ่มสูงขึ้นเท่านั้น

เมื่อผู้ป่วยเป็นเบาหวานนาน ๆ หลาย ๆ ปี เส้นเลือดฝอยทั่วร่างกายจะเกิดการเปลี่ยนแปลง ในส่วนของตาจะทำให้เกิดการเปลี่ยนแปลงที่ผนังของหลอดเลือดในจอประสาทตา ทำให้เกิดมีการผิดปกติ คือ มีเม็ดเลือด มีน้ำเหลือง และไขมัน ซึมออกมาในจอประสาทตา ทำให้เกิดจอประสาทตาบวม ขาดออกซิเจน และเมื่อเป็นเช่นนี้นานๆเข้า จะทำให้เกิด : เส้นเลือดงอกขึ้นใหม่ --> เกิด

เลือดออก --> ทำให้น้ำวุ้นภายในลูกตาขุ่นมัว --> จอประสาทตาลอก --> และท้ายสุดทำให้ตาบอด เราเรียกโรคแทรกซ้อนที่เกิดขึ้นนี้ว่า "เบาหวานขึ้นตา"

ปัจจัยที่ทำให้เกิดจอประสาทตาเสื่อมนั้น ขึ้นกับปัจจัยหลายประการ ประการแรกคือ ระยะเวลาที่เป็นโรคเบาหวาน ประการที่2 คือการควบคุมระดับน้ำตาลในเลือดได้ดีเพียงใด และประการที่ 3โรคอื่นๆที่มีส่วนทำให้เกิดการเสื่อมของจอประสาทตาเร็วขึ้น เช่น โรคไต ความดันโลหิตสูง หรือสตรีที่มีครรภ์

2.1.2 อาการของผู้ป่วยเป็นโรคเบาหวานขึ้นจอประสาทตา

ถึงแม้ว่าโรคเบาหวานจะทำให้สูญเสียการมองเห็นลงอย่างมาก หรือตาบอด แต่ผู้ป่วยจะไม่มีอาการเจ็บปวด ในความเป็นจริงผู้ป่วยที่มีเบาหวานขึ้นจอประสาทตา จะไม่รู้สึกถึงความผิดปรกติในการมองเห็น จนกระทั่งเป็นรุนแรงมากแล้ว ถ้าเบาหวานขึ้นจอประสาทตา และมีจุดกลางรับภาพจอประสาทตาบวม (Diabetic Macula edema) ร่วมด้วย ผู้ป่วยจะรู้สึกมองภาพไม่ชัด ตามัว หรือใช้สายตาระยะใกล้ เช่นอ่านหนังสือได้ลำบากถ้ามีเส้นเลือดงอกผิดปรกติ (Neovascularization) บนจอประสาทตาแตกจะมีเลือดออกในน้ำวุ้นตา (Vitreous hemorrhage) อาจมีอาการเหมือนมีเงาดำๆบัง ลอยไปมาจนถึงมืด และมองอะไรไม่เห็นได้ ดังนั้นผู้ป่วยโรคเบาหวาน จึงควรพบจักษุแพทย์ทันที ถ้ามีการมองเห็นผิดปรกติไป เช่น

- ตามัวลง 1-2 วัน และไม่ดีขึ้น
- อาการตามัวไม่สัมพันธ์กับระดับน้ำตาลในเลือดที่สูงขึ้น
- มีเงาดำลอยไปมา (floater)

อย่างไรก็ตาม ผู้ป่วยเบาหวานส่วนใหญ่ แม้มีเบาหวานขึ้นจอประสาทตามากแล้ว ก็ยังไม่แสดงการมองเห็นที่ผิดปรกติ จึงเป็นสิ่งสำคัญและจำเป็นอย่างที่สุด ที่ผู้ป่วยเบาหวานทุกราย จะต้องได้รับการขยายม่านตา ตรวจจอประสาทตาโดยจักษุแพทย์ อย่างน้อยปีละ 1 ครั้ง เพราะว่าโรคเบาหวานขึ้นจอประสาทตาสามารถรักษา และป้องกันไม่ให้ตาบอดได้ ถ้าได้รับการรักษาอย่างทันท่วงที

อาการเปลี่ยนแปลงของจอตาในผู้ป่วยเบาหวาน มี 2 ระยะ ได้แก่

ระยะที่ 1 มีการเปลี่ยนแปลงของหลอดเลือดที่จอรับภาพ บางเส้นมีขนาดเล็กลง บางเส้นใหญ่โป่งพองคล้าย ๆ ถุงน้ำ บางเส้นตีบแคบลงทำให้เกิดการอุดตันการไหลเวียนเลือด ทำให้มีเลือดหรือน้ำซึมออกมา ทำให้จอรับภาพบวม ระยะนี้ความผิดปกติของจอรับภาพยังไม่รุนแรง แต่ในบางรายที่มีน้ำซึมเข้าไปสะสมทำให้ตาพร่ามัวมักเป็นอาการเตือนที่จะนำไปสู่อาการขั้นรุนแรงต่อไปได้

ระยะที่ 2 จะมีอาการเกิดใหม่ของหลอดเลือดบนผิวของจอตา หลอดเลือดใหม่นี้จะเปราะแล้วแตกง่ายจึงมีเลือดออกในน้ำวุ้นของตา ทำให้แสงผ่านจากเลนส์ไปยังจอตาไม่ได้ดี นอกจากนี้ยังมีพังผืดหรือเนื้อเยื่อที่จะดึงรั้งทำให้จอรับภาพลอกหลุดมา ถึงขั้นนี้ผู้ป่วยอาจจะมองเห็นแค่มือไหว ๆ หรือเห็นเพียงลำแสงที่ส่องผ่านตา ระยะนี้การรักษาด้วยแสงเลเซอร์ไม่ได้ผล ทำผ่าตัดน้ำวุ้นตาก็อาจดีขึ้นบ้าง แต่เลือดอาจออกมาอีกจากหลอดเลือดที่งอกผิดปกติในจอตาได้

2.1.3 ลักษณะอาการที่แสดงออกของที่เป็นโรคเบาหวาน

จากรูปที่ 2.7 เป็นการแสดงลักษณะอาการของผู้ป่วยที่มีเบาหวานขึ้นจอประสาท ซึ่งประกอบด้วย 5 อาการหลักดังนี้ a) ภาพจอประสาทตาที่มีหลอดเลือดโป่งพอง b) ภาพจอประสาทตาที่มีเลือดออก c) ภาพไขมันขึ้นจอประสาทตา d) ภาพไขมันขึ้นจอประสาทตาเล็กน้อย e) ภาพเส้นเลือดงอกผิดปกติของจอประสาทตา ซึ่งในการทดลองนี้ได้วิเคราะห์ลักษณะแบบ c) เนื่องจากเป็นลักษณะของไขมันซึ่งเป็นพิกเซลที่สว่างจึงทำให้สามารถหา Dilation ได้

(a) **(b)** **(c)** **(d)** **(e)**

รูปที่ 2.1 รูปตาที่ไม่ปกติที่เกิดจากโรคเบาหวานขึ้นจอประสาทตา

โดยที่ **(a)** หลอดเลือดโป่งพอง (จุดที่ลูกศรชี้) **(b)** เลือดออกที่ประสาทตา **(c)** มีไขมันในชั้นประสาทตา **(d)** มีไขมันในชั้นประสาทตาเล็กน้อย (จุดที่ลูกศรชี้) **(e)** เส้นเลือดงอกผิดปรกติ

2.2 ภาพดิจิตอล (Digital Image)

ภาพดิจิตอล คือ ภาพที่ได้จากการถ่ายด้วยกล้องดิจิตอล การสแกนภาพด้วยเครื่องสแกน จะแตกต่างจากภาพอนาลอก ในหลายแง่มุมคือ ความละเอียด สี คุณภาพชิ้นงาน ขั้นตอนการทำงาน และการจัดการ ตลอดจนถึงค่าใช้จ่าย ในธรรมชาติ แสงสีเป็นพลังงานคลื่นแม่เหล็กไฟฟ้าที่มีสมบัติของคลื่น สมบัติเฉพาะทางไฟฟ้า และแม่เหล็กอยู่รวมกัน โดยที่มีความยาวคลื่นที่ตาของมนุษย์สามารถมองเห็นได้ในช่วง 380 – 780 นาโนเมตร ในช่วงความยาวคลื่นที่ตามองเห็นแตกต่างกันนี้ คือ แสงสีที่ต่างกันในสเปคตรัมของสี ดวงตามนุษย์สามารถมองเห็นสีที่แตกต่างกันได้จากที่ดวงตาของมนุษย์มีเซลล์รับแสงสีชนิดโคนที่ต่างกัน 3 ชนิดคือ โคนแดง โคนน้ำเงิน และโคนเขียว ที่ดูดแสงสีในช่วงคลื่น 570, 445, 535 นาโนเมตร การแปลผลสีอื่นๆ ที่ต่างไปจากนี้จะเกิดขึ้นได้จากการผสมแสงสีทั้ง 3 ในอัตราส่วนต่างๆกัน

2.2.1 รูปร่างของภาพ (Image Shape)

วัตถุที่มีอยู่ตามธรรมชาติและที่มนุษย์สร้างขึ้นมีรูปร่างที่แตกต่างกันไป ทั้งที่เป็นรูปทรงเรขาคณิตและไม่เป็นรูปทรงเรขาคณิตในศาสตร์ของการประมวลผลภาพนั้น การกำหนดขอบเขตของภาพทุกภาพให้อยู่ในรูปสี่เหลี่ยม (Rectangular image model) เป็นวิธีที่นิยมใช้กันมากที่สุด เนื่องจากทำให้การอ่านภาพ การจัดเก็บข้อมูลภาพในหน่วยความจำ และการแสดงภาพออกทางอุปกรณ์ต่างๆ เป็นไปได้อย่างมีประสิทธิภาพ

การเก็บข้อมูลภาพสามารถทำได้โดยการจองหน่วยความจำของเครื่องคอมพิวเตอร์ไว้ในรูปของตัวแปรอะเรย์ (Array) โดยค่าในแต่ละช่องของอะเรย์แสดงถึงคุณสมบัติของจุดภาพ (pixel) และตำแหน่งของช่องอะเรย์เป็นตัวกำหนดตำแหน่งของจุดภาพ (pixel)

2.2.2 พิกเซล (Pixel)

พิกเซล (Pixel) เป็นคำผสมของคำว่า Picture กับคำว่า Element หรือหน่วยพื้นฐานของภาพเทียบได้กับ “จุดภาพ” 1 จุด แต่ละพิกเซลเปรียบได้กับสี่เหลี่ยมเล็กๆที่บรรจุค่าสี โดยถูกกำหนดตำแหน่งไว้บนเส้นกริดของแนวแกน x และแกน y หรือในตารางเมตริกซ์สี่เหลี่ยม ภาพบิตแมปจะประกอบด้วยพิกเซลหลายๆพิกเซล (pixel)

จำนวนพิกเซลของภาพแต่ละภาพ จะเรียกว่า ความละเอียด หรือ Resolution โดยจะเทียบจำนวนพิกเซลกับความยาวต่อนิ้ว ดังนั้นจะมีหน่วยเป็นพิกเซลต่อนิ้ว (ppi : pixels per inch) หรือจุดต่อนิ้ว (dpi : dot per inch) ภาพขนาดเท่ากันแต่มีความละเอียดต่างกัน แสดงว่าจำนวนพิกเซลต่างกันและขนาดของจุดพิกเซลก็ต่างกันด้วย

2.2.3 ความละเอียดในการแสดงผล (Resolution)

คำนี้สามารถใช้ได้กับสถานการณ์ที่แตกต่างกัน เช่น ความละเอียดของการแสดงผลของเครื่องพิมพ์ หรือความละเอียดในการแสดงผลของจอภาพ ดังนั้นความละเอียดในการแสดงผลจึงหมายถึง จำนวนหน่วยต่อพื้นที่

ความละเอียดในการแสดงผล เป็นความสามารถในการปรับระยะการแสดงความละเอียดของภาพดิจิตอลระยะห่างของความถี่ในการแสดงภาพ (ความถี่ในการทำ sampling) ถูกระบุในรูปของความละเอียดในการแสดงผล ซึ่งหมายถึง จุดต่อนิ้ว (dpi) หรือ พิกเซลต่อนิ้ว (ppi) เป็นคำทั่วไปที่ใช้เรียกหรือบ่งบอกว่ามีการแสดงภาพอยู่ที่ระดับความละเอียดในการแสดงผลที่เท่าไร แต่อยู่ในขอบเขตจำกัด การเพิ่มความถี่ในการแซมปลิ้งเป็นการเพิ่มความละเอียดในการแสดงผลด้วยเช่นกัน

สมมติให้ภาพเป็นตัวแปรแบบอะเรย์ขนาด M x N (M แถว และ N คอลัมน์) ที่ใช้เก็บภาพขนาด M x N จุด (M จุดในแนวนอน และ N จุดในแนวตั้ง) ค่าสี (หรือความสว่าง ในกรณีที่เป็นภาพ grey level) ของจุดภาพในแถวที่ 5 คอลัมน์ที่ 4 จะตรงกับค่าของภาพ (5,4) จะเห็นว่าเราใช้ตำแหน่งของจุดภาพทั้งสองแกนเป็นตัวชี้ค่าข้อมูลในอะเรย์

การใช้หน่วยความจำเพื่อการเก็บภาพในลักษณะที่กล่าวมา เนื้อที่ที่ใช้ในการเก็บภาพสามารถคำนวณได้จาก M x N x g เมื่อ g เป็นจำนวนเต็มที่แทนจำนวนบิตของข้อมูลในแต่ละจุดภาพ ตัวอย่างถ้า g มีค่าเท่ากับ 8 บิต สามารถเก็บความแตกต่างของระดับสีที่เป็นไปสูงสุด 256 ระดับ ค่า M และ N จะเป็นตัวบอกถึงความละเอียดของภาพ สำหรับคอมพิวเตอร์ทั่วไปในระบบ VGA (Video Graphic Array) จะมีขนาด 640 x 480, 800 x 600 และ 1024x768 จุด เป็นต้น การกำหนดความละเอียดจะขึ้นอยู่กับงานที่จะใช้ ในงานบางอย่างใช้ความละเอียดแค่ 30 x 50 จุด ก็พอแล้วแต่ในงานบางชนิด ใช้ความละเอียดถึง 1000 x 1000 จุดก็ยังไม่พอ

2.2.4 บิต (BIT)

Bit ย่อมาจาก Binary Digit หมายถึง หน่วยความจำที่เล็กที่สุดของคอมพิวเตอร์ ประกอบด้วยตัวเลข 2 จำนวน คือ 0 หมายถึงปิด และ 1 หมายถึงเปิด หรือสีขาวและสีดำ

ความลึกของบิต (Bit Depth) หมายถึง จำนวนบิตที่ใช้ในแต่ละพิกเซลในกราฟิกแบบบิตแมปสีของพิกเซลถูกบันทึกโดยใช้บิต ถ้าใช้สีมากก็แสดงสีได้มากขึ้น ถ้ามีหน่วยความจำ 2 บิต ในการเก็บรวบรวมข้อมูล สามารถใช้สีได้ทั้งหมด 2 เท่ากับ 4 สี คือ สามารถกลับสีได้ 4 วิธี คือ 00, 01, 10 และ 11 ถ้ามี 2 บิต สามารถสร้างสีให้กับพิกเซลทั้งหมด 4 เฉดสี

จำนวนสีสูงสุดที่เป็นไปได้ของแต่ละจุดภาพขึ้นอยู่กับจำนวนบิตที่ใช้ เมื่อมีการกำหนดให้ขนาดของบิตต่อจุดมากขึ้นจะทำให้จำนวนของสีมากขึ้นด้วย ตัวอย่างเช่น 1 บิต = 2^1 = 2 สี, 2 บิต = 2^2 = 4 สี, 4 บิต = 2^4 = 16 สี, 8 บิต = 2^8 = 256 สี, 16 บิต = 2^{16} = 65536 สี

ภาพขาว-ดำอยู่ในรูปพิกเซล ที่แต่ละพิกเซลจะมี 1 บิต ซึ่งแสดงได้ 2 ระดับสี คือ ขาวและดำ โดยค่า 0 เป็นสีดำ และ 1 เป็นสีขาว หรือตรงกันข้าม

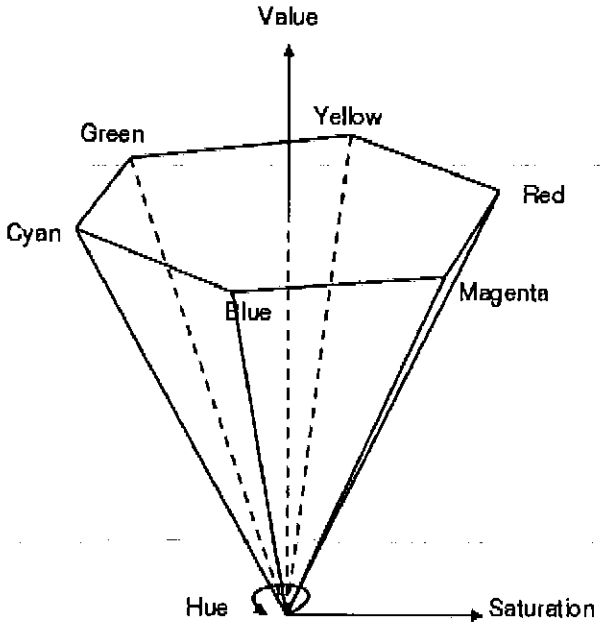
ภาพระดับเทา เป็นการเรียงของพิกเซลที่ใช้ข้อมูลแบบหลายบิต อยู่ในช่วงระหว่าง 2 – 8 บิต หรือมากกว่านั้น

ภาพสี แบบทั่วไปนั้นมีค่า ความลึกบิต อยู่ในช่วง 8 – 24 บิต หรือมากกว่า ภาพที่มี 24 บิต นั้นคือ บิตจะถูกแบ่งออกเป็น 3 กลุ่ม 8 สำหรับสีแดง 8 สำหรับสีเขียว 8 สำหรับสีน้ำเงิน สีทั้งหมด

จะถูกรวมกันเพื่อแสดงสีอื่นๆภาพ 24 บิต สามารถแสดงค่าสีได้ถึง 16.7 ล้านสี (2^{24}) สำหรับเครื่องสแกนนั้นได้เพิ่มจำนวนบิตในการจับภาพเอกสารเป็น 10 บิต หรือมากกว่านั้นเพราะบ่อยครั้งหลังจากจับภาพเอกสารที่ 8 บิต จะมีสัญญาณรบกวน รวมเข้าไปด้วย และเพื่อให้ภาพที่ออกนั้นมีความเหมือนใกล้เคียงกับที่มนุษย์ต้องการ อาจจะมีการใช้สีถึง 36, 48, 64 บิต แต่ในคอมพิวเตอร์ทั่วๆ ไปมักจะใช้สีไม่เกิน 32 บิต

2.3 มาตรฐานของสี (Color Standard)

มาตรฐานของสีที่ใช้อยู่ในปัจจุบันมีหลายระบบด้วยกัน แต่โดยทั่วไปแล้วทุกมาตรฐานจะมีแนวคิดเดียวกันก็คือ การแทนจุดสีด้วยจุดที่อยู่ภายในสเปซ 3 มิติ โดยจะมีแกนอ้างอิงสำหรับจุดสีนั้นในสเปซซึ่งแต่ละแกนจะมีอิสระต่อกัน ตัวอย่างเช่นในระบบ RGB จะมีแกนสีคือ แกนสีแดง สีเขียว และสีน้ำเงิน ในระบบ HLS จะมีแกนเป็นค่าสี (Hue) ความสว่าง (Lighness) และความบริสุทธิ์ของสี (Saturation)



รูปที่ 2.2 แสดงมาตรฐานสี

2.4 การแปลงภาพเป็นระดับเทา (Gray Scale)

การแปลงภาพสีเป็นภาพระดับเทาได้นั้นคือการ แปลงค่า RGB ให้เป็นค่าเฉลี่ย โดยการนำค่าสี RG และ B ในแต่ละพิกเซลมาบวกกันและหาด้วย 3จากนั้นแทนลงไปในพิกเซลนั้นๆซึ่งจะได้จำนวนบิตที่ใช้แทนระดับความเข้มของภาพเท่ากับ 24 บิตเหมือนเดิม โดยทั่วไปมักแสดงดังสมการที่ (2.1)

วิธีการหาค่าระดับเทา (Gray Level)

$$\text{Gray Level} = \frac{R+G+B}{3} \tag{2.1}$$

วิธีการหาค่าระดับเทาที่กล่าวมาข้างต้นเป็นการเฉลี่ยค่าของแม่สีทั้งสาม ซึ่งเป็นวิธีการที่ง่ายที่สุด แต่ก็อาจจะมีความเพี้ยนของสีได้ จึงมีวิธีอีกอย่างหนึ่งซึ่งจะคิดตามความสว่างของแต่ละแม่สี โดยมีรูปแบบดังสมการที่ (2.2), (2.3) และ (2.4)

$$R_R = \frac{(R_S + G_S + B_S)}{3} \tag{2.2}$$

$$G_R = \frac{(R_S + G_S + B_S)}{3} \tag{2.3}$$

$$B_R = \frac{(R_S + G_S + B_S)}{3} \tag{2.4}$$

โดยที่ R_R หมายถึง ค่าเอาต์พุตพิกเซลสีแดง G_R หมายถึง ค่าเอาต์พุตพิกเซลสีเขียว

B_R หมายถึง ค่าเอาต์พุตพิกเซลสีน้ำเงิน R_R หมายถึง ค่าอินพุตพิกเซลสีแดง

G_S หมายถึง ค่าอินพุตพิกเซลสีเขียว B_S หมายถึง ค่าอินพุตพิกเซลสีน้ำเงิน

2.5 ทฤษฎีการขยายพิกเซลของภาพ (Dilation Theory)

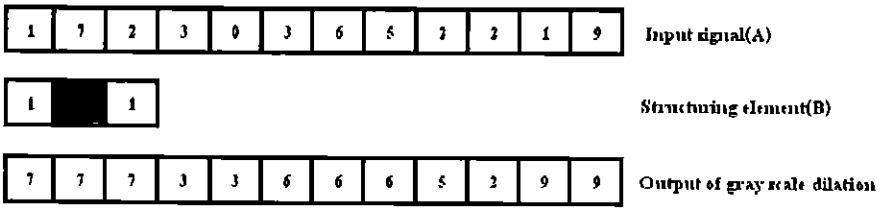
การขยายพิกเซลของภาพ (Dilation Methods) คือ การขยายพิกเซลที่มีค่าสูงกว่าพิกเซลรอบๆให้มีขนาดกว้างขึ้น ซึ่งโครงงานนี้จะใช้การขยายพิกเซลของภาพระดับเทา

การขยายพิกเซลของภาพระดับเทา (Gray Scale Dilation)

จากสมการที่ (2.5) กำหนดให้ A คือ ภาพอินพุตมีค่าอยู่ระหว่าง 0 ถึง 255 B คือ Structuring element ที่มีค่าแต่ละพิกเซลเป็น 1 และกำหนดให้พิกเซลตรงกลางของ Structuring element คือจุด Origin ซึ่งจุด Origin เป็นตำแหน่งอ้างอิงที่ใช้ในการคำนวณหาภาพเอาต์พุต การหาภาพเอาต์พุตสามารถทำได้โดยเริ่มจาก นำ Structuring element ที่ตำแหน่ง Origin ไปเทียบกับพิกเซลที่ 1 ของ

ภาพ A และพิจารณาค่าพิกเซลของภาพ A ภายใน Structuring element นำค่าที่มากที่สุดของภาพ A ใส่ลงในค่าของพิกเซลที่ 1 ของภาพเอาต์พุต จากนั้นเลื่อนไปที่ละพิกเซลจนครบหมดทุกพิกเซลในภาพ A โดยที่ค่าที่มากที่สุดของภาพ A จะถูกใส่ในทุกๆพิกเซลที่ตำแหน่ง Origin เทียบอยู่ ซึ่งได้ผลดังรูปที่ 2.3

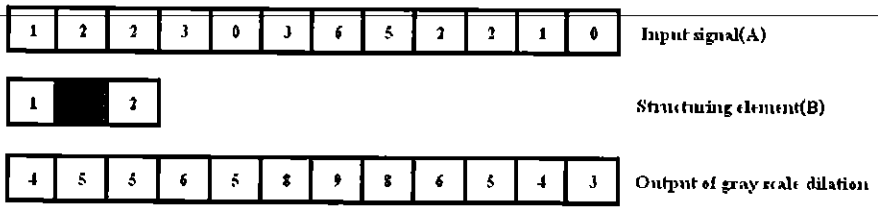
$$A \oplus B = \max_{i \in B, \forall x} (A_{x+i}) \qquad (2.5)$$



รูปที่ 2.3 ตัวอย่างการไดเลชันระหว่างภาพ A กับภาพหน่วยโครงสร้าง B = {1,1,1}

จากสมการที่ (2.6) กำหนดให้ A คือ ภาพอินพุตมีค่าอยู่ระหว่าง 0 ถึง 255 B คือ Structuring element มีค่าแต่ละพิกเซลเป็นจำนวนเต็ม 0 ถึง 255 ขั้นตอนการหาภาพเอาต์พุต สามารถทำได้เหมือนจากสมการ (2.5) แต่ในส่วนนี้ค่าของเอาตุตที่ได้มาจากการนำภาพอินพุต A บวกกับค่าแต่ละพิกเซลของ Structuring element พิจารณาผลบวกที่มากที่สุดที่ได้ เป็นค่าของตำแหน่งใน Origin ของรูปเอาต์พุต

$$A \oplus B = \max_{i \in B, \forall x} (A_{x+i} + B_i) \qquad (2.6)$$



รูปที่ 2.4 ตัวอย่างการไดเลชันระหว่างภาพ A กับภาพหน่วยโครงสร้าง B = {1,3,2}

2.6 ทฤษฎีการหาขอบ (Edge Detection Theory)

การหาขอบภาพ (Edge Detection) เป็นการหาเส้นรอบวัตถุที่อยู่ในภาพ เมื่อทราบเส้นรอบวัตถุ เราจะสามารถคำนวณหาพื้นที่ (ขนาด) หรือรู้จักชนิดของวัตถุนั้นได้ อย่างไรก็ตามการหาขอบภาพที่ถูกต้องสมบูรณ์ไม่ใช่เป็นเรื่องที่ง่าย โดยเฉพาะอย่างยิ่งการหาขอบของภาพที่มีคุณภาพต่ำ มีความแตกต่างระหว่างพื้นหน้ากับพื้นหลังน้อย หรือมีความสว่างไม่สม่ำเสมอทั่วภาพ ขอบภาพเกิดจากความแตกต่างของความเข้มแสงจากจุดหนึ่งไปยังอีกจุดหนึ่ง หากต่างนี้มีค่ามาก ขอบภาพก็จะเห็นได้ชัด ถ้าความแตกต่างมีค่าน้อย ขอบภาพก็จะไม่ชัดเจน ถ้าต้องการหาขอบภาพในแนวนอนอย่างง่าย วิธีการก็คือหาผลต่างระหว่างจุดหนึ่งกับจุดที่อยู่ข้างล่าง (หรือข้างบน) ของจุดนั้น ดังนี้

$$Y_{diff}(x,y) = I(x,y) - I(x,y+1) \tag{2.7}$$

โดยที่ Y_{diff} คือค่าความแตกต่างในแนวแกนตั้ง และ $I(x,y)$ คือค่าความเข้มแสงของจุดภาพที่ตำแหน่ง (x,y) การหาขอบภาพในแนวตั้งก็สามารถหาได้เช่นเดียวกันคือ

$$X_{diff}(x,y) = I(x,y) - I(x-1,y) \tag{2.8}$$

โดยที่ X_{diff} คือค่าความแตกต่างในแนวนอน

บางครั้งเราต้องการรวมผลต่างของค่าความแตกต่างในแนวแกนนอน และแกนตั้งเข้าด้วยกัน เพื่อที่จะได้มีตัววัดความแรงของขอบภาพ (Gradient Magnitude) เพียงตัวเดียว เนื่องจากค่าความแตกต่างอาจมีค่าเป็นบวกหรือลบ ดังนั้น การบวกค่าความแตกต่างของทั้งสองแกนอาจทำให้ขอบภาพเกิดการหักล้างกันเอง ในทางปฏิบัติ เราจะต้องนำค่าสัมบูรณ์ (Absolute Value) หรือค่ากำลังสอง (Squared Value) ของค่าความแตกต่างของทั้งสองแกนมาบวกกันแทน นอกจากหาความแรงของขอบภาพแล้ว การหาทิศทางของขอบภาพ (Gradient Direction) ก็มีประโยชน์เช่นกัน การหาทิศทางของขอบภาพสามสามารถทำได้โดยการใช้สมการต่อไปนี้

$$GD(x,y) = \tan^{-1}\left(\frac{Ydiff(x,y)}{Xdiff(x,y)}\right) \tag{2.9}$$

โดยที่ GD (x,y) คือ ทิศทางของขอบภาพที่ตำแหน่ง (x,y) และ Ydiff(x,y) , Xdiff(x,y) มาจากสมการที่ (2.7) , (2.8) ตามลำดับ โดยทั่วไปแล้ว เป็นเรื่องยากที่จะบอกว่าการหาขอบแบบใดเหมาะสมที่จะใช้กับภาพแต่ละลักษณะ ในส่วนนี้ได้กล่าวถึงการหาขอบภาพที่ไม่ต่อเนื่องโดยใช้เทคนิค Roberts

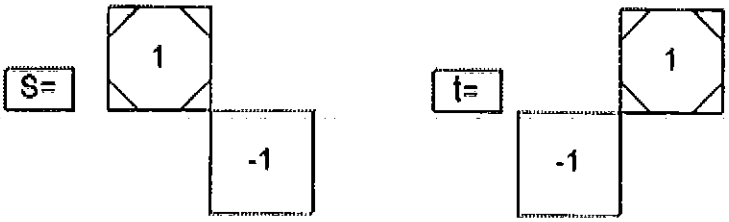
2.6.1 Roberts Edge Detector

การหาขอบด้วยวิธีนี้ก็เป็นคือการใช้เทคนิคการปรับปรุงขอบที่ไม่ต่อเนื่องโดยกำหนดให้ $a \in R^{M \times N}$ เป็นภาพต้นแบบ และขอบที่ได้เป็น $b \in R^{M \times N}$ ซึ่ง $b \in R^{M \times N}$ ที่ได้ เกิดจากผลรวมของขอบในแนวตั้ง แนวนอน และแนวทแยง ผลที่ได้คือสามารถหาขอบภาพได้ทุกทิศทาง ทำให้ได้ขอบของภาพที่สมบูรณ์มากกว่าการหาเพียงทิสทางใดทิสทาหนึ่ง โดยแสดงในสมการที่ (2.10)

$$b(i,j) = ((a(i,j) - a(i+1,j+1))^2 + (a(i,j+1) - a(i+1,j))^2)^{\frac{1}{2}} \qquad (2.10)$$

โดยที่ a คือค่าเมตริกซ์ของรูปต้นแบบ b คือค่าเมตริกซ์ของขอบภาพ $R^{M \times N}$ คือจำนวนจริงภายในเมตริกซ์ภาพขนาด $M \times N$

จากรูปที่ 2.5 เป็นการแสดงลักษณะของเทมเพลตที่นำมาใช้ในการคำนวณหาขอบภาพ ของ Roberts มีค่าเป็นบวกและลบในแนวทแยง แบบออกเป็นสองลักษณะคือ s และ t ซึ่ง s เป็นการหาขอบในแนวทแยงซ้ายโดยให้ค่าเทมเพลตเป็นเมตริกซ์ [1 0 0 -1] ส่วน t เป็นการหาขอบในแนวทแยงขวามีค่าเทมเพลตเป็นเมตริกซ์ [0 1 -1 0]



รูปที่ 2.5 เทมเพลตของ s และ t

จากรูปที่ 2.6 แสดงตัวอย่างการหาขอบด้วยเทคนิคโรเบิรตส์โดยที่ (a) คือรูปต้นแบบก่อนทำการหาขอบและ (b) คือรูปที่ผ่านการหาขอบด้วยเทคนิคโรเบิรตส์แล้ว

(a) (b)

รูปที่ 2.6 ตัวอย่างการหาขอบด้วยวิธีของโรเบิรตส์

2.7 กระบวนการวัดทางสถิติ (Statistical Operations)

ที่ได้กล่าวมาแล้วเกี่ยวกับการรับภาพเข้าสู่คอมพิวเตอร์และรูปแบบของการนำภาพที่อยู่ในหน่วยความจำไปแสดงผลด้วยอุปกรณ์ต่างๆจะเห็นได้ว่ามีกระบวนการทำได้หลายๆวิธี เพื่อที่จะให้ได้ผลตามที่ต้องการ

ในการแบ่งกระบวนการประมวลผลเกี่ยวกับภาพแบ่งได้หลักๆ3 อย่างด้วยกันคือ ระดับต่ำ ระดับปานกลาง และระดับสูง โดยขึ้นอยู่กับการกระทำเกี่ยวกับบิตของภาพ ซึ่งอาจต้องผ่านการประมวลระดับต่ำ ระดับปานกลาง และระดับสูง ตามลำดับเพื่อให้ได้ผลลัพธ์ออกมาดีที่สุด

การประมวลระดับต่ำ จะจัดการกับภาพแบบขาว-ดำ โดยปกติจะทำการสร้างภาพที่สองขึ้นมาโดยให้มีเฉพาะข้อมูลที่ต้องการ ส่วนไหนที่ไม่ต้องการจะทำการตัดออกไป

การประมวลระดับปานกลาง เป็นการประมวลเกี่ยวกับการบ่งบอกว่าภาพมีลักษณะรูปร่าง พื้นที่ หรือจุดของภาพจากการประมวลผลแบบระดับต่ำ

การประมวลระดับสูง เป็นการรู้ลักษณะทั่วๆไปที่จำเป็นของภาพ เช่น มีการเชื่อมต่อกันของรูปร่างอะไรบ้าง เพื่อที่จะทำให้รู้รูปร่างที่แท้จริงของวัตถุ ผลของการประมวลผลในระดับนี้จะนำไปสู่การวิเคราะห์ภาพด้วย

ในส่วนนี้จะกล่าวถึงการประมวลผลภาพในระดับอย่างต่ำซึ่งจะขึ้นอยู่กับตำแหน่งต่างของพิกเซล

2.7.1 เทร็ชโฮลด์ (Threshold)

การทำเทร็ชโฮลด์ (Threshold) ใช้เพื่อปรับระดับสีของภาพให้เป็นกลุ่มเพื่อแยกให้เห็นความแตกต่างชัดเจนขึ้น กระบวนการทำเทร็ชโฮลด์ (Threshold) เริ่มด้วยการนำค่าระดับเทาของภาพ $I(x,y)$ แต่ละพิกเซลมาเปรียบเทียบกับค่าเทร็ชโฮลด์ (Threshold) ถ้าค่าระดับเทาของภาพมีค่ามากกว่าหรือเท่ากับค่าเทร็ชโฮลด์ (Threshold) ที่กำหนด ในตำแหน่งพิกเซลนั้นจะเปลี่ยนเป็น 1 แต่ถ้าค่าระดับเทาน้อยกว่าค่าเทร็ชโฮลด์ (Threshold) จะเปลี่ยนเป็น 0 ดังสมการที่ (2.11)

$$B(x,y)=\begin{cases}1, I(x,y)\geq 128\\ 0, I(x,y)<128\end{cases} \qquad (2.11)$$

โดยที่ $I(x,y)$ คือ ค่าระดับเทาแต่ละพิกเซลของภาพระดับก่อนทำเทร็ชโฮลด์ (Threshold) และ $B(x,y)$ คือ ค่าระดับเทาแต่ละพิกเซลที่ผ่านการทำเทร็ชโฮลด์ (Threshold) แล้ว

จากตารางที่ 2.1 กำหนดให้แนวแกนตั้งเป็น y และแนวแกนนอนเป็น x เมื่อนำ I(x,y) ไปเปรียบเทียบกับค่าเทร็ชโฮลด์ (Threshold) ซึ่งเท่ากับ 128 จะได้ค่า B(x,y) ดังแสดงในตาราง 2.2 ถ้าพิกเซลใดมีค่ามากกว่า 128 ค่าที่ผ่านการเทร็ชโฮลด์ (Threshold) แล้วจะเป็น 1 และพิกเซลใดน้อยกว่า 128 จะได้ค่าที่ผ่านการเทร็ชโฮลด์ (Threshold) แล้ว จะเป็น 0 แสดงดังตารางที่ 2.2

ตารางที่ 2.1 ตารางแสดงค่าในภาพระดับเทา

I(x,y) =

47	230	170	237	71	219	124	30	53	113
66	111	40	170	74	145	132	44	78	187
199	190	141	224	160	103	170	241	198	196
162	195	192	15	111	160	2	97	224	162
88	152	210	175	229	62	155	157	22	54
216	223	117	196	27	6	250	74	1	83
190	231	88	99	39	28	195	197	145	7
118	206	95	252	114	52	54	136	215	13
1	145	88	183	97	72	152	91	92	117
228	4	28	253	245	207	28	173	210	39

ตารางที่ 2.2 ตารางแสดงค่าในภาพขาวดำหลังการทำเทร็ชโฮลด์ (Threshold)

B(x,y) =

0	1	1	1	0	1	0	0	0	0
0	0	0	1	0	1	1	0	0	1
1	1	1	1	1	0	1	1	1	1
1	1	1	0	0	1	0	0	1	1
0	1	1	1	1	0	1	1	0	0
1	1	0	1	0	0	1	0	0	0
1	1	0	0	0	0	1	1	1	0
0	1	0	1	0	0	0	1	1	0
0	1	0	1	0	0	1	0	0	0
1	0	0	1	1	1	0	1	1	0

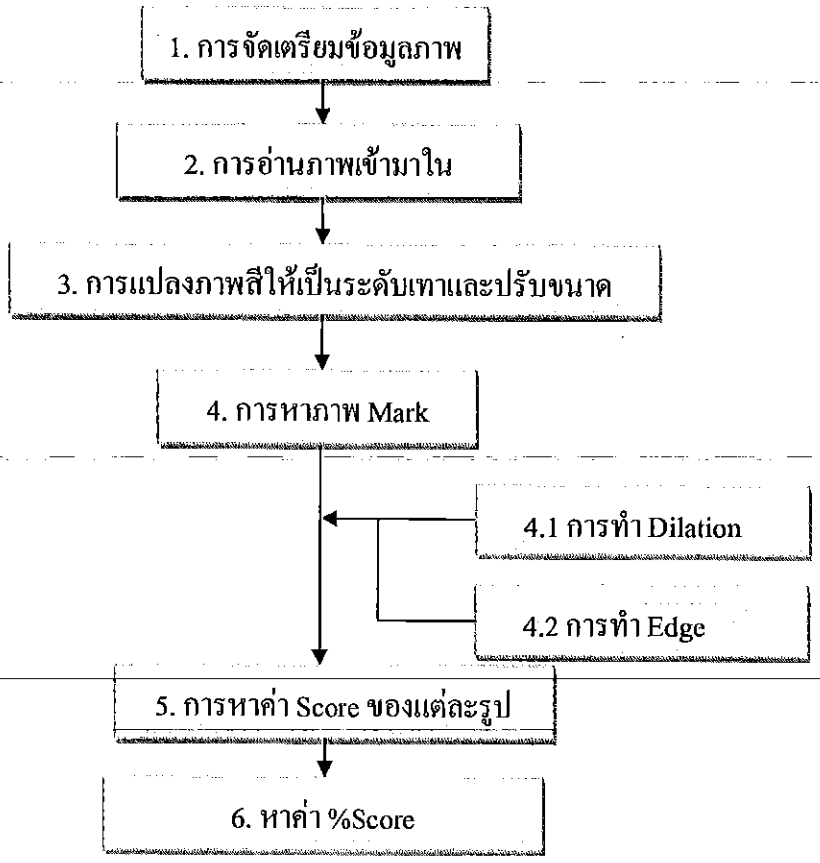
การปรับระดับสีของภาพให้เป็นกลุ่มเพื่อง่ายต่อการพิจารณารูป ซึ่งเป็นส่วนสำคัญในการนำค่าแต่ละพิกเซลไปคำนวณหาความผิดปกติในจอประสาทตาในส่วนถัดไป

บทที่ 3
ขั้นตอนการดำเนินงาน

ในบทนี้กล่าวถึงขั้นตอนในการดำเนินงานทั้งหมดของโครงงาน เริ่มจากการจัดหาข้อมูลภาพ โดยแบ่งกลุ่มภาพจอประสาทตาที่ปกติและผิดปกติ แล้วจึงทำการแปลงภาพให้อยู่ในระดับเทาและปรับขนาด จนกระทั่งถึงขั้นตอนในการปรับปรุงภาพหรือการหาภาพ Mark ซึ่งจากนั้นจึงเข้าสู่กระบวนการทำ Dilation และ Edge Detection เพื่อนำผลที่ได้ (ค่า Score) ไปวิเคราะห์หาค่าเปอร์เซ็นต์ความผิดปกติของภาพจอประสาทตา (ค่า %Score)

ขั้นตอนหลักในการดำเนินงาน

จากรูปที่ 3.1 เป็นการแสดงขั้นตอนทั้งหมดในการดำเนินงานของโครงงานนี้ เพื่อช่วยให้เข้าใจหลักการและขั้นตอนในการทำงานเพิ่มมากขึ้น ซึ่งประกอบด้วย 6 ขั้นตอนหลักดังนี้

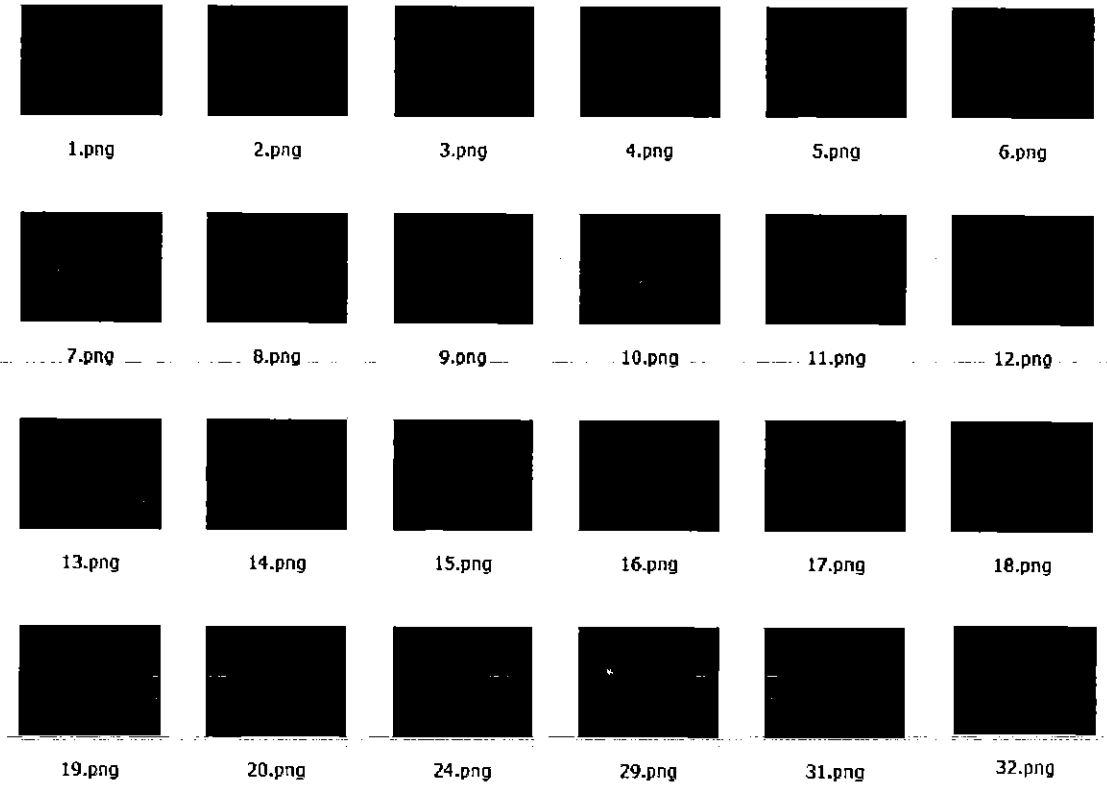


รูปที่ 3.1 ขั้นตอนการดำเนินงาน

3.1 การเตรียมข้อมูลภาพสำหรับกระบวนการการทดลอง

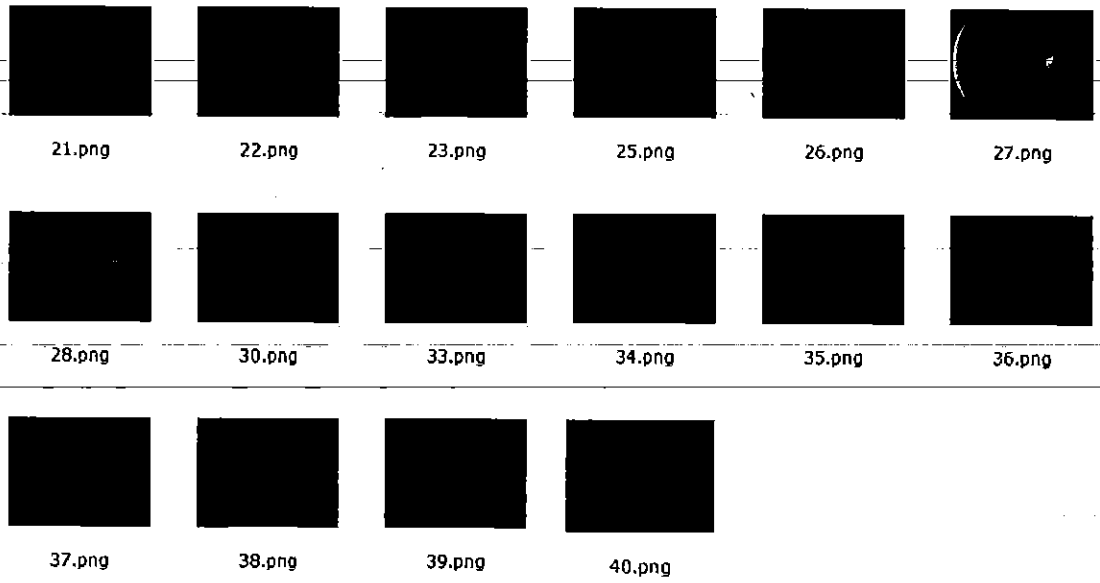
ทำการจัดหาภาพจอประสาทตาที่จะใช้ในการทดลอง ซึ่งการทดลองนี้ใช้ภาพจอประสาทตาทั้งหมด 40 ภาพ และจะทำการแบ่งกลุ่มภาพออกเป็น 2 กลุ่ม โดยการสังเกตด้วยตาเปล่า คือกลุ่มที่ผิดปกติและปกติ โดยกลุ่มที่ผิดปกติแสดงไว้ดังรูปที่ 3.2 ซึ่งมีทั้งหมด 24 ภาพ คือภาพที่ 1ถึง20 และ24, 29, 31, 32 และกลุ่มที่ปกติแสดงไว้ดังรูปที่ 3.3 ซึ่งมีทั้งหมด 16 ภาพ คือภาพที่ 21, 22, 23, 25 ถึง 28, 30, 33 ถึง 40

กลุ่มที่ 1 กลุ่มของตาที่ผิดปกติ



รูปที่ 3.2 รูปจอประสาทตาที่ผิดปกติ

กลุ่มที่ 2 กลุ่มของตาที่ปกติ



รูปที่ 3.3 รูปจอประสาทตาที่ปกติ

3.2 การอ่านไฟล์ภาพเข้ามาในโปรแกรม

ในการทดลองนี้ต้องมีไฟล์ภาพเก็บไว้เป็นฐานข้อมูลและอ่านไฟล์ภาพเข้ามาในโปรแกรมด้วยคำสั่ง "Imread" ซึ่งไฟล์ภาพที่อ่านเข้ามาจะใช้เป็นไฟล์ภาพ '.PNG' เนื่องจากไฟล์ภาพ '.PNG' นั้นมีความละเอียดน้อยกว่าไฟล์ภาพชนิดอื่นๆ

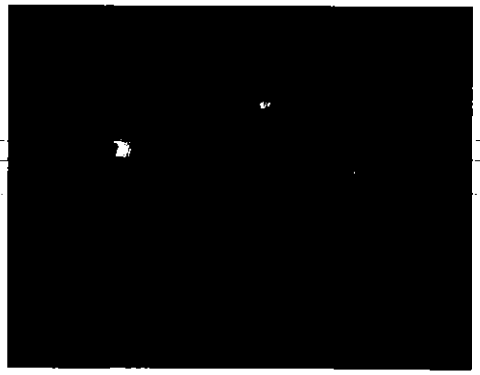
3.3 การปรับขนาดและการแปลงภาพสีให้เป็นภาพระดับเทา

1). การปรับขนาด

ทำการปรับขนาดของภาพอินพุตที่อ่านเข้ามาทุกภาพเพื่อให้ภาพที่จะนำมาพิจารณามีขนาดเท่ากันหมดโดยใช้คำสั่ง 'IMSIZE'

2). การแปลงภาพสีให้เป็นภาพระดับเทา

การแปลงภาพจากภาพสีให้เป็นภาพขาวดำโดยใช้คำสั่ง 'RGB2GRAY' (RGB to Gray) และผลที่ได้จะเป็นดังแสดงในรูปที่ 3.4



รูปที่ 3.4 ตัวอย่างภาพระดับเทา

3.4 การนำภาพอินพุตมาหาภาพ Mark

ภาพ Mark คือภาพที่แยกส่วนผิดปกติออกมาแสดงเพื่อนำไปวิเคราะห์หาเปอร์เซ็นต์ความผิดปกติต่อไป ซึ่งเป็นกระบวนการนำภาพอินพุตมาผ่านขั้นตอนการปรับปรุงภาพเพื่อให้ได้ภาพตามที่เราต้องการ ซึ่งประกอบด้วย 2 ขั้นตอนหลักดังนี้

1). การทำ Dilation

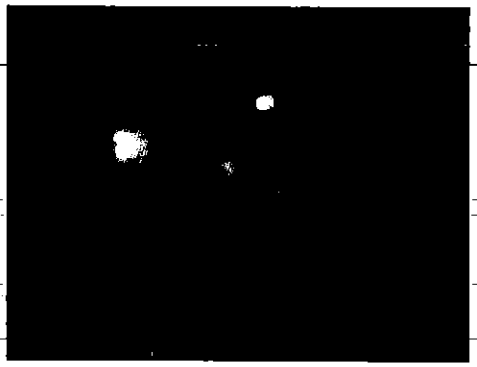
การทำ Dilation คือการขยายพิกเซลที่สว่างให้มีขนาดกว้างขึ้น จะทำให้ได้ลักษณะผิดปกติที่เป็นไขมันหรือบริเวณกลุ่มสีขาวออกมา และในทางกลับกับกัน การทำ Dilation ก็จะลดพิกเซลสว่างน้อย ให้เล็กลงด้วยเช่นกันจึงทำให้เส้นเลือดมีขนาดเล็กลง เนื่องจากเส้นเลือดจะมีลักษณะสีเป็นสีแดงทึบซึ่งมีความสว่างน้อยกว่าบริเวณที่มีไขมันเกาะอยู่

ขั้นตอนการทำ Dilation นั้นสิ่งที่ควรคำนึงถึงคือ การเลือกใช้ Structuring element ให้เหมาะสม ในการทดลองนี้เลือกใช้ Structuring element ชนิด Ball ซึ่งเหมาะกับลักษณะภาพจอประสาทตามากที่สุด นอกจากการเลือกใช้ชนิดของ Structuring element แล้ว ขนาดของ Structuring element ก็มีผลเช่นกัน เนื่องจากขนาดของ Structuring element ที่แตกต่างกันนั้น จะมีผลต่อความละเอียดของภาพ Mark ที่ได้ ในการทดลองนี้ได้ทำการปรับค่า Structuring element ทั้งหมด 3 ครั้ง คือ 10 ,15 และ20 เพื่อหาขนาดที่เหมาะสมที่สุดในการปรับปรุงภาพ

การทำ Dilation ของภาพใน Matlab สามารถทำได้โดยการใช้ฟังก์ชัน “imdilate” และเทคนิคที่ใช้คือ ball ซึ่งมีรูปแบบคำสั่งในโปรแกรมดังนี้

```
I=imread('image.png'); % Input Image
se=strel('ball',R,N);     % R และ N คือตัวแปรขนาดในการปรับค่า Structuring element
I2=imdilate(I,se);        % การ Dilation แล้วแสดงผล
```

ซึ่งผลของการใช้ฟังก์ชัน "Imdilate" แสดงไว้ดังรูปที่ 3.5

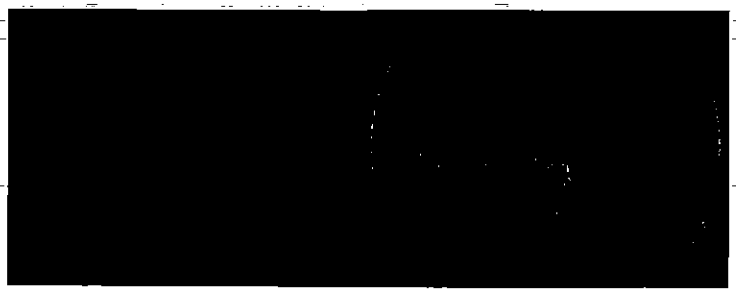


รูปที่ 3.5 ตัวอย่างภาพที่ผ่านการทำ Dilation

2). การทำ Edge Detection

การทำ Edge Detection คือการหาขอบของภาพ ในการทดลองนี้ ขอบที่ต้องการพิจารณาคือขอบของบริเวณที่ผิดปกติ ดังนั้นการนำภาพที่ผ่านการทำ Dilation แล้วมาทำการหาขอบ จึงทำให้ได้ขอบของอาการที่ผิดปกติค่อนข้างชัดเจน แทบไม่มีส่วนที่เป็นเส้นเลือดแสดงให้เห็นเลย ซึ่งเป็นผลดีต่อการนำค่าของภาพที่ได้ไปวิเคราะห์ในขั้นตอนต่อไป

ในการเลือกใช้เทคนิคการหาขอบของภาพเพื่อนำมาวิเคราะห์นั้นสำคัญมากเพราะแต่ละเทคนิคจะใช้กับการหาขอบที่มีลักษณะต่างกัน การตรวจหาจุดก็จะต่างกัน ดังนั้นในการทดลองนี้จึงเลือกใช้เทคนิค Roberts Edge ซึ่งเป็นการหาขอบโดยใช้เทคนิคการปรับปรุงขอบที่ไม่ต่อเนื่อง จึงเหมาะสำหรับการนำมาใช้กับภาพจอประสาทตา เนื่องจากภาพจอประสาทตานี้ มีลักษณะของอาการผิดปกติที่แตกต่างกัน ดังนั้นจากการทดลองใช้เทคนิคการหาขอบชนิด Roberts Edge นั้น ภาพ Mark ที่ได้มีตำแหน่งของความผิดปกติที่เราต้องการค่อนข้างชัดเจน ดังตัวอย่างที่แสดงไว้ในรูปที่ 3.6



ภาพต้นแบบ ภาพ Mark

รูปที่ 3.6 ตัวอย่างภาพต้นแบบและภาพ Mark

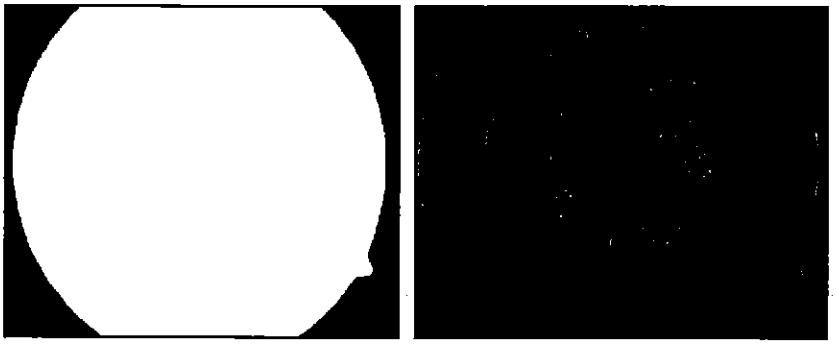
3.5 การหาค่า Score

ค่า Score คือค่าที่บอกถึงผลรวมของพิกเซลเฉพาะส่วนที่ผิดปกติ ซึ่งหาค่าได้จากการหารระหว่างผลรวมของพิกเซลในภาพ Mark กับ ภาพ Mask เพื่อให้ได้พิกเซลสีขาวออกมาจากภาพ Mark ซึ่งมีขั้นตอนในการหา 3 ขั้นตอนดังนี้

1). ขั้นตอนการหาข้อมูลของภาพ Mask และภาพที่ผ่านการหา Mark

เมื่อเราได้ภาพ Mask และภาพ Mark ดังรูปที่ 3.7 แล้วจะนำภาพที่ได้มาทำการแปลงจากข้อมูลภาพให้เป็นข้อมูลเมตริกซ์ตัวเลข (ข้อมูลชนิด Double) เพื่อที่จะสามารถนำไปใช้ในการคำนวณหาผลรวมของพิกเซลภาพได้ โดยการแปลงข้อมูลภาพให้เป็นข้อมูลชนิด Double สามารถทำได้โดยใช้รูปแบบฟังก์ชันในโปรแกรมดังนี้

```
I1 = imread('image.png'); % Input Image
I2 = double(I1);     % เปลี่ยนเป็นข้อมูลชนิดdouble
```



ภาพ Mask ภาพ Mark

รูปที่ 3.7 ภาพ Mask และภาพ Mark

2). ขั้นตอนการรวมค่าของพิกเซลในภาพ Mark และ Mask

คือ การนำค่าของภาพในแต่ละพิกเซลมาทำการรวมกันเพื่อที่จะได้ผลรวมทั้งหมดของทุกพิกเซลในภาพ โดยที่พิกเซลที่เป็นสีดำ จะมีค่าเป็น 0 และพิกเซลที่เป็นสีขาวจะมีค่าเป็น 1 ซึ่งผลรวมนี้จะทำให้เราได้ค่าของพิกเซลสีขาวทั้งหมดซึ่งเป็นการแสดงออกถึงความผิดปกติของภาพ

3). ขั้นตอนการหาผลหารระหว่างผลรวมของพิกเซลของภาพ Mark และ ภาพ Mask

คือ กระบวนการในการนำผลรวมของพิกเซลที่ได้จากทั้งสองรูปมาหารกันเพื่อให้เหลือเฉพาะส่วนที่เราต้องการจะพิจารณานั้นก็คือ ในส่วนของพิกเซลสีขาวหรือส่วนที่ผิดปกติ ซึ่งในขั้นตอนนี้เป็นขั้นตอนสุดท้ายที่ทำให้ได้ค่า Score ของภาพออกมา

3.6 การหาค่า %Score

ค่า %Score คือ ค่าที่บอกถึงระดับเปอร์เซ็นต์ความผิดปกติของลูกตา ซึ่งหาได้จากการนำค่า Score ที่ได้มาคำนวณเทียบกับค่า Score ของภาพทุกภาพที่เรามีเก็บไว้เป็นฐานข้อมูลของการวิเคราะห์ โดยเราจะทำการหาข้อมูลค่า Score ที่เป็นค่าสูงสุดในฐานข้อมูล และเมื่อได้ค่า Score สูงสุดนั้นแล้ว เราจะกำหนดให้ค่า Score ของภาพนั้นเป็นค่า Score Maximum และเราก็จะนำค่า Score ของภาพที่เราต้องการหาเปอร์เซ็นต์นั้น มาเทียบกับค่า Score Maximum และได้ผลสุดท้ายออกมาเป็น %Score ซึ่งจะสามารถสรุปเป็นสูตรได้ดังนี้

$$\%\text{Score} = \frac{\text{ค่า Score ของภาพ}}{\text{ค่า Score Maximum}} \times 100$$

ห้องสมุดคณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร
ปร. 5200025
ม614อ
2551

บทที่ 4

ผลการทดลองและวิเคราะห์ผลการทดลอง

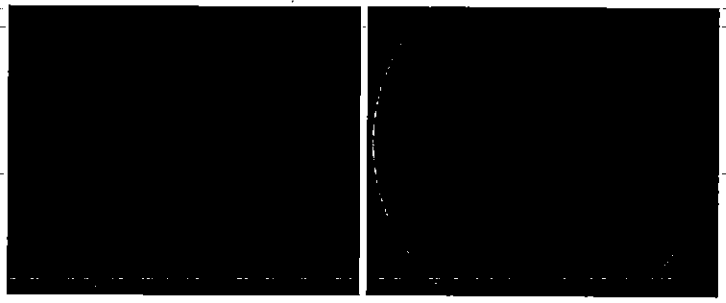
ในบทนี้กล่าวถึงผลการทดลองและวิเคราะห์ผลการทดลองในการตรวจวัดความผิดปกติภาพลูกตาโดยใช้ Dilation Theory และ Edge Detection โดยภาพที่ใช้ในการทดลองคือภาพจอประสาทตาที่ปกติและผิดปกติ โดยแบ่งการทดลองออกเป็น 3 กรณีคือ กรณีที่ 1 การทำ Dilation โดยใช้ค่า Structuring element เท่ากับ 10 พิกเซล กรณีที่ 2 การทำ Dilation โดยใช้ค่า Structuring element เท่ากับ 15 พิกเซล และกรณีสุดท้ายใช้ค่า Structuring element เท่ากับ 20 พิกเซล จากการทดลองทั้ง 3 กรณีเป็นการทดลองเพื่อหาค่า Structuring element ที่ทำให้ได้ค่าเปอร์เซ็นต์ความผิดพลาดเหมาะสมที่สุด

4.1 ผลการทดลองปรับค่า Structuring element ทั้ง 3 กรณี

<u>กรณีที่ 1</u> ผลการทดลองของค่า Structuring element = 10×10 พิกเซล

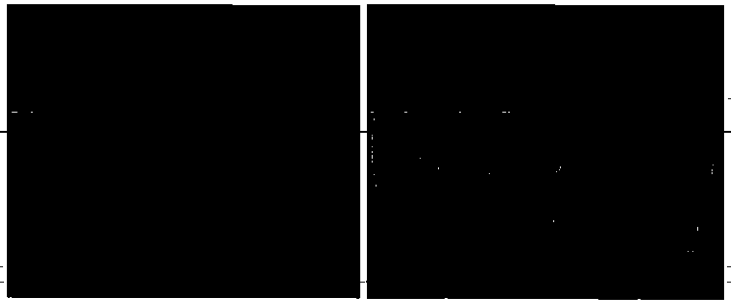
1). ตัวอย่างผลการทดลองที่ค่า Structuring element = 10×10พิกเซล

จากรูปที่ 4.1, 4.2,4.3 และ 4.4 เป็นภาพการเปรียบเทียบระหว่าง (1) ภาพต้นแบบกับ (2) ภาพผลการทดลอง (ภาพ Mark) ซึ่งทำ Dilation โดยใช้ค่า Structuring element เท่ากับ 10×10 พิกเซล แล้วจึงนำไปหา Edge Detection ซึ่งภาพที่ได้ออกมานั้นจะค่อนข้างมีสิ่งที่ไม่ต้องการมากจนเกินไป เช่น ส่วนของเส้นเลือด ที่ไม่ต้องการ เพราะทำให้ภาพที่ได้นั้นไปมีผลกับจำนวนของพิกเซลสีขาวซึ่งนำไปคำนวณเป็นค่า Score และค่า %Score ที่ต้องการวิเคราะห์



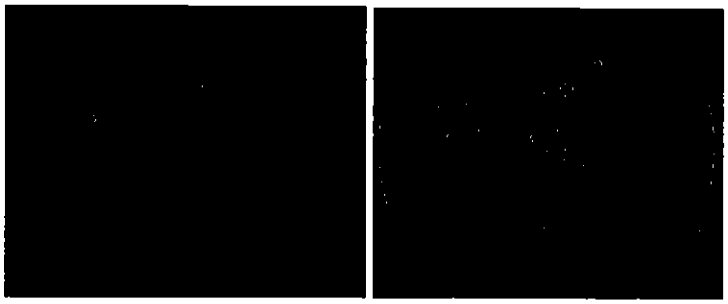
(1) (2)

รูปที่ 4.1 ตัวอย่างภาพที่มีค่า Score = 0.003855 และ %Score = 23.744912



(1) (2)

รูปที่ 4.2 ตัวอย่างภาพที่มีค่า Score = 0.008832 และ %Score = 54.396639



(1) (2)

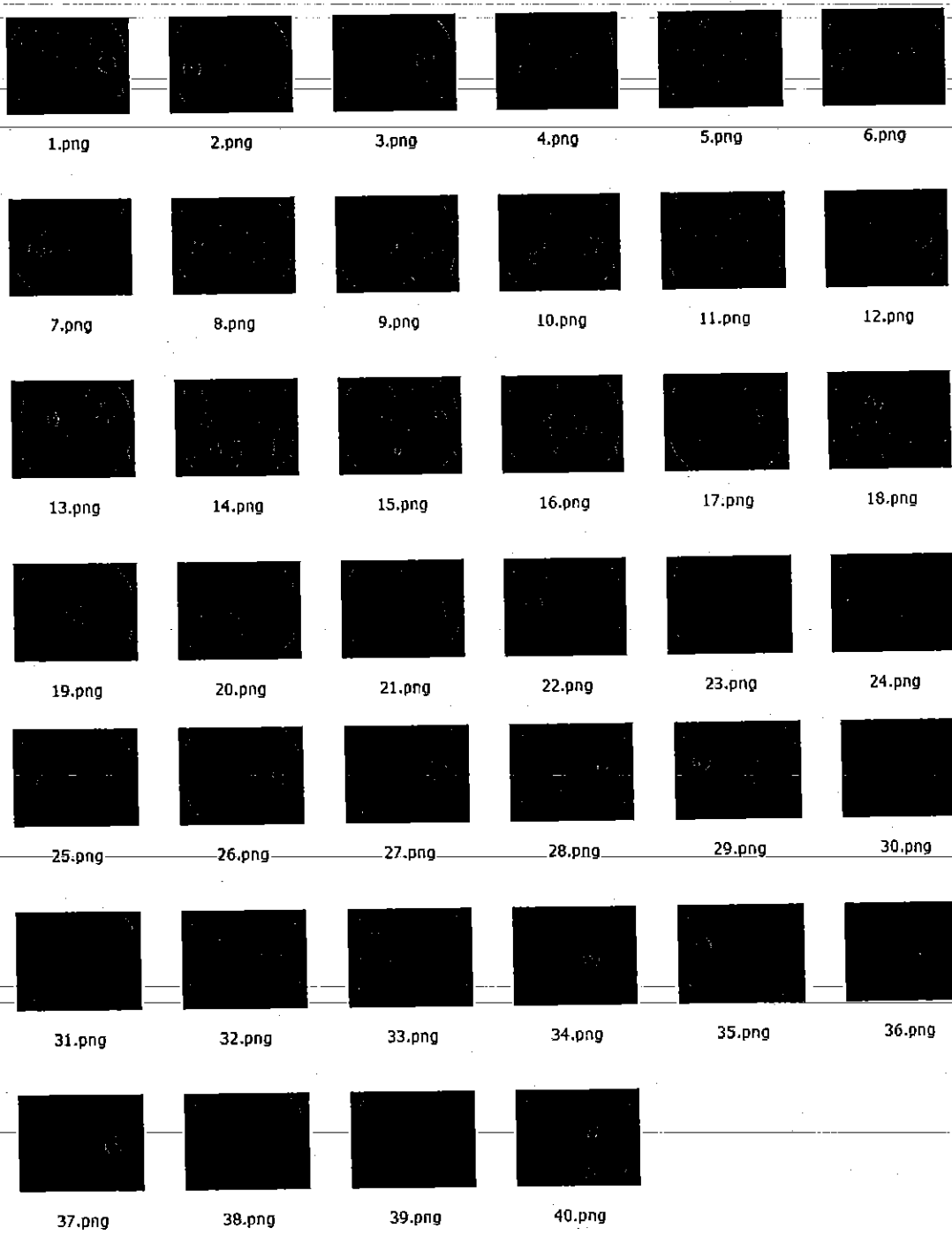
รูปที่ 4.3 ตัวอย่างภาพที่มีค่า Score = 0.012618 และ %Score = 77.721364

(1) (2)

รูปที่ 4.4 ตัวอย่างภาพที่มีค่า Score = 0.015047 และ %Score = 92.681753

2). ภาพผลการทดลอง Structuring element = 10×10 พิกเซล ทั้งหมด

จากรูปที่ 4.5 เป็นการแสดงภาพผลการทดลองทั้งหมดซึ่งได้จากการใช้ค่า Structuring element เท่ากับ 10×10 พิกเซล โดยทำการทดลองทั้ง 40 ภาพ เพื่อที่จะนำภาพที่ได้ทุกภาพไปทำการวิเคราะห์หาค่า Score และค่า %Score แล้วนำค่าที่ได้มาทำตารางเปรียบเทียบค่า เพื่อดูผลความแตกต่างของ%จากการใช้ค่า Structuring element แต่ละค่า



รูปที่ 4.5 ภาพผลการทดลองทั้งหมดโดยที่ค่า Structuring element 10×10 พิกเซล

3). ตารางผลการทดลองค่า Structuring element = 10×10พิกเซล

จากการทำ Dilation ซึ่งทดลองใช้ค่า Structuring element เท่ากับ10×10 พิกเซล จะเห็นได้ว่าค่า Score ที่ได้นั้นจะมีค่าค่อนข้างมาก เนื่องจาก ค่า Structuring element = 10×10 พิกเซล นั้น จะมีการกรองที่ไม่ค่อยมาก ทำให้รายละเอียดหรือ พิกเซลสีขาวของภาพจอประสาทตาผิดปกติ ที่ได้ออกมานั้นค่อนข้างเยอะจึงทำให้ค่า Score สูงขึ้นและเมื่อค่า Score สูงขึ้น ก็จะส่งผลให้ค่า %Score สูงขึ้นตามไปด้วย ดังแสดงไว้ในตารางที่ 4.1

หมายเหตุ : ค่า Score และค่า % Score ที่เป็นสีดำในตารางนั้น คือค่าน้อยกว่า 50% ซึ่งในการทดลองนี้จะให้ถือว่าเป็นภาพจอประสาทตาที่ปกติ

ค่า Score และค่า %Score ที่เป็นสีแดง คือค่ามากกว่า 50% ซึ่งในการทดลองนี้จะให้ถือว่าเป็นภาพจอประสาทตาที่ผิดปกติ

ค่า %Score ที่มีสัญลักษณ์ * นั้นในการทดลองนี้จะให้ถือว่าภาพนั้นเป็นภาพที่ไม่เข้ากลุ่ม หรือ ภาพที่ผิดพลาด

ตารางที่4.1 สรุปค่า Score และค่า % Score ที่ Structuring element = 10×10 พิกเซล

ลำดับ Image	Score	% Score
1	0.010698	65.890489
2	0.009261	57.044689
3	0.008572	52.799055
4	0.008832	54.396639
5	0.015047	92.681753
6	0.015134	93.215739
7	0.011811	72.749157
8	0.009082	55.941699
9	0.016235	100
10	0.012647	77.896442
11	0.012618	77.721364
12	0.012196	75.12146
13	0.012142	74.784436
14	0.014748	90.83906
15	0.009284	57.184751
16	0.013155	81.025955
17	0.009128	56.221823
18	0.014025	86.38771
19	0.008832	54.396639
20	0.008832	54.396639
21	0.006557	40.386046

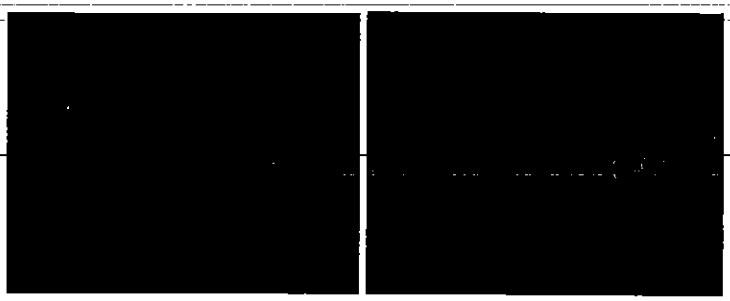
ตารางที่4.1 (ต่อ) สรุปค่า Score และค่า % Score ที่ Structuring element = 10×10 พิกเซล

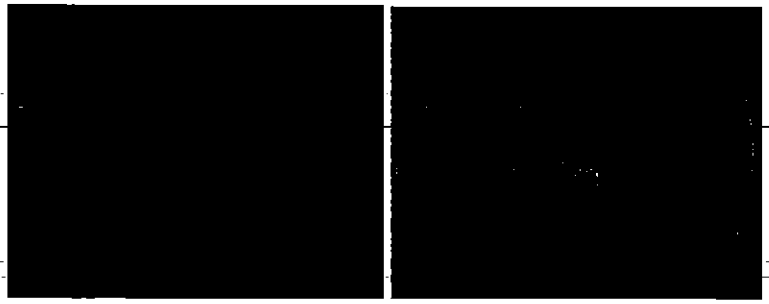
22	0.008054	49.608264
23	0.003855	23.744912
24	0.007105	43.760669*
25	0.005491	33.820633
26	0.006705	41.29645
27	0.005056	31.141944
28	0.008994	55.398958*
29	0.013146	80.969055
30	0.004755	29.286121
31	0.006523	40.18033*
32	0.007054	43.449906*
33	0.007743	47.691163
34	0.005553	34.205804
35	0.006428	39.59382
36	0.006625	40.806233
37	0.007258	44.706088
38	0.004303	26.506762
39	0.005675	34.954261
40	0.012949	79.756642

<u>กรณีที่ 2</u> ผลการทดลองของค่า Structuring element = 15×15 พิกเซล

1). ตัวอย่างผลการทดลองที่ค่า Structuring element = 15×15 พิกเซล

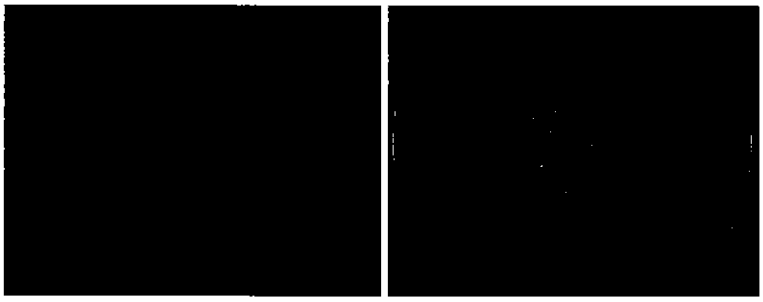
จากรูปที่ 4.6, 4.7,4.8 และ 4.9 เป็นภาพการเปรียบเทียบระหว่าง (1) ภาพต้นแบบ กับ (2) ภาพผลการทดลอง (ภาพMark) ซึ่งทำ Dilation โดยใช้ค่า Structuring element เท่ากับ 15×15 พิกเซล แล้วจึงนำไปหา Edge Detection ซึ่งภาพที่ได้ออกมา ค่อนข้างชัดเจนไม่มีส่วนที่เป็นเส้นเลือดติดมา และจะกรองเฉพาะบริเวณที่ผิดปกติหรือพิกเซลสีขาว ซึ่งค่า Structuring element เท่ากับ 15×15 พิกเซล เป็นค่าที่จะนำมาวิเคราะห์ใช้จริงในการทดลอง เนื่องจากเป็นค่าที่ทำให้ได้ภาพ Mark ที่สมบูรณ์ที่สุด





(1) (2)

รูปที่ 4.7 ตัวอย่างภาพที่มีค่า Score = 0.006475 และ %Score = 52.277682



(1) (2)

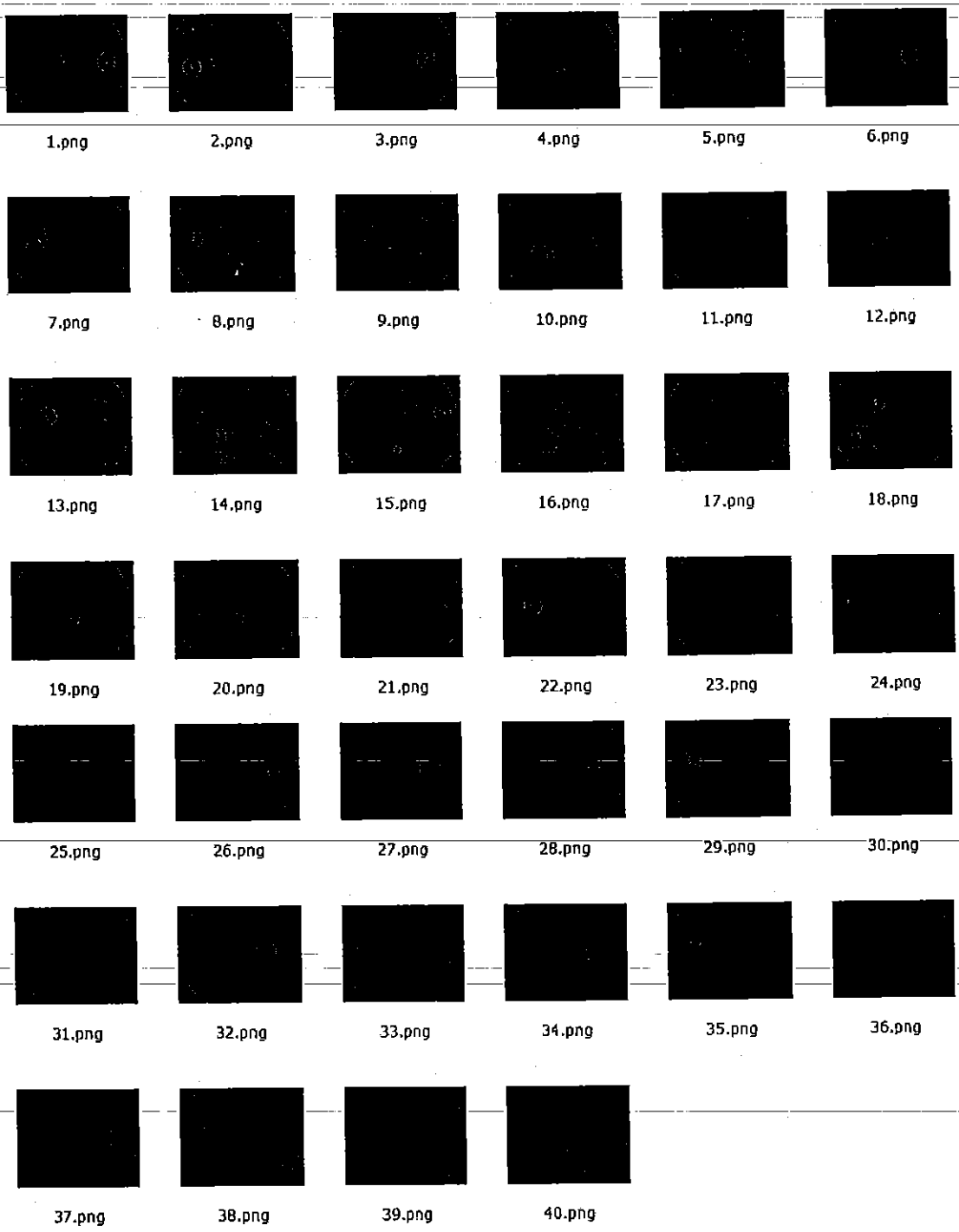
รูปที่ 4.8 ตัวอย่างภาพที่มีค่า Score = 0.008714 และ %Score = 70.349971

(1) (2)

รูปที่ 4.9 ตัวอย่างภาพที่มีค่า Score = 0.010135 และ %Score = 81.830178

2). ภาพผลการทดลอง Structuring element = 15×15 พิกเซล ทั้งหมด

จากรูปที่ 4.10 เป็นการแสดงภาพผลการทดลองทั้งหมดซึ่งได้จากการใช้ค่า Structuring element เท่ากับ 15×15 พิกเซล โดยทำการทดลองทั้ง 40 ภาพ เพื่อที่จะนำภาพที่ได้ทุกภาพไปทำการวิเคราะห์หาค่า Score และค่า %Score แล้วนำค่าที่ได้มาทำตารางเปรียบเทียบค่า เพื่อดูผลความแตกต่างของเปอร์เซ็นต์จากการใช้ค่า Structuring element แต่ละค่า



รูปที่ 4.10 ภาพผลการทดลองทั้งหมดโดยที่ค่า Structuring element 15×15 พิกเซล

3). ตารางผลการทดลองค่า Structuring element = 15×15พิกเซล

จากการทำ Dilation ซึ่งทดลองใช้ค่า Structuring element เท่ากับ15×15 พิกเซล จะเห็นได้ว่าค่า Score ที่ได้นั้นจะมีค่าน้อยกว่าการใช้ค่า Structuring element เท่ากับ10×10 พิกเซล เนื่องจากค่า Structuring element = 15×15 พิกเซล นั้น จะมีการกรองที่ดีกว่า ทำให้รายละเอียดหรือ พิกเซลสีขาวของภาพจอประสาทตาผิดปกติ ที่ได้ออกมานั้นค่อนข้างชัดเจนและไม่มีลักษณะของเส้นเลือดติดมาด้วย จึงทำให้ค่า Score ที่ได้ลดลงจากเดิมแต่ก็ยังอยู่ในระดับที่เหมาะสม และเมื่อค่า Score ลดลง ก็จะส่งผลให้ค่า %Score ลดลงตามไปด้วย แต่ทั้งนี้ค่า Score และค่า %Score ที่ได้นั้น ยังคงเป็นค่าที่เหมาะสมและใกล้เคียงกับความเป็นจริงมากที่สุด ดังแสดงไว้ในตารางที่ 4.2

หมายเหตุ : ค่า Score และค่า %Score ที่เป็นสีดำในตารางนั้น คือค่าน้อยกว่า 50% ซึ่งในการทดลองนี้จะให้ถือว่าเป็นภาพจอประสาทตาที่ปกติ

ค่า Score และค่า %Score ที่เป็นสีแดง คือค่ามากกว่า 50% ซึ่งในการทดลองนี้จะให้ถือว่าเป็นภาพจอประสาทตาที่ผิดปกติ

ค่า %Score ที่มีสัญลักษณ์ * นั้นในการทดลองนี้จะให้ถือว่าภาพนั้นเป็นภาพที่ไม่เข้ากลุ่ม หรือ ภาพที่ผิดพลาด

ตารางที่ 4.2 สรุปค่า Score และค่า %Score ที่ Structuring element = 15×15 พิกเซล

ลำดับ Image	Score	% Score
1	0.008056	65.037292
2	0.007084	57.194492
3	0.007177	57.94607
4	0.006475	52.277682
5	0.010135	81.830178
6	0.011479	92.679289
7	0.007828	63.201377
8	0.006604	53.321859
9	0.012386	100
10	0.009959	80.401606
11	0.009137	73.769363
12	0.009169	74.027539
13	0.010085	81.422834
14	0.009907	79.988526
15	0.007913	63.884108
16	0.008714	70.349971
17	0.006444	52.025244
18	0.009632	77.762478

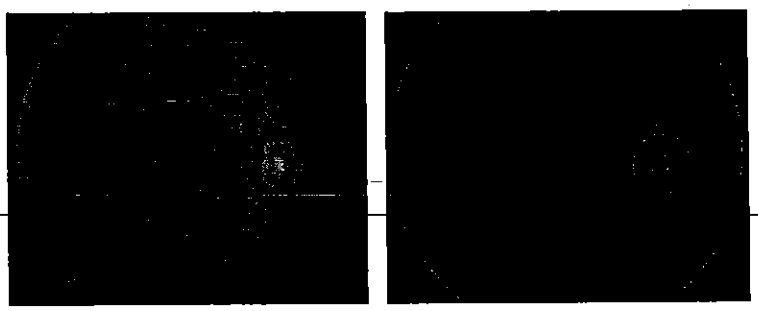
ตารางที่ 4.2(ต่อ) สรุปค่า Score และค่า %Score ที่ Structuring element = 15×15 พิกเซล

19	**0.006475**	**52.277682**
20	**0.006475**	**52.277682**
21	0.004934	39.83362
22	0.0054	43.597246
23	0.002388	19.282846
24	0.003753	30.304073*
25	0.003845	31.044177
26	0.003968	32.036718
27	0.003935	31.772806
28	0.005293	42.730924
29	**0.009322**	**75.261044**
30	0.004353	35.140562
31	0.005123	41.359725*
32	0.00506	40.854848*
33	0.004666	37.670683
34	0.005492	44.337349
35	0.004974	40.160643
36	0.00483	38.995984
37	0.004232	34.17097
38	0.003149	25.427424
39	0.004191	33.83821
40	**0.007699**	**62.162937***

<u>กรณีที่ 3</u> ผลการทดลองของค่า Structuring element = 20×20พิกเซล

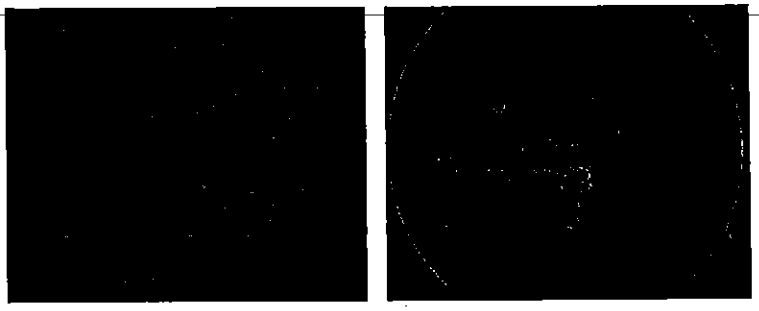
1). ตัวอย่างผลการทดลองที่ค่า Structuring element = 20×20พิกเซล

จากรูปที่ 4.11, 4.12,4.13 และ 4.14 เป็นภาพการเปรียบเทียบระหว่าง (1) ภาพต้นแบบ กับ (2) ภาพผลการทดลอง (ภาพ Mark) ซึ่งได้จากการทำ Dilation โดยใช้ค่า Structuring element เท่ากับ 20×20 พิกเซล แล้วนำไปหา Edge Detection ซึ่งภาพที่ได้ ไม่ค่อยชัดเจนเนื่องจากรายละเอียดของภาพจะถูกกรองมากจนเกินไป ทำให้บริเวณที่ผิดปกติบางส่วนหายไป ซึ่งค่า Structuring element เท่ากับ 20×20 พิกเซล นี้เป็นค่าที่กรองภาพมากเกินไปจาก Structuring element ทั้งสามค่าดัง จึงไม่ใช้ค่า Structuring element เท่ากับ 20×20 พิกเซล ในการทดลองจริง



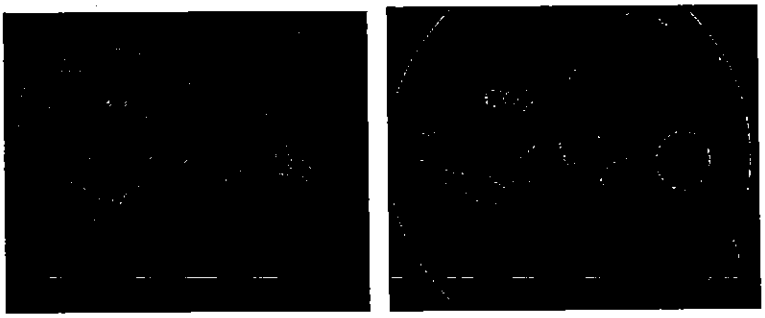
(1) (2)

รูปที่ 4.11 ตัวอย่างภาพที่มีค่า Score = 0.005499และ %Score = 37.785048



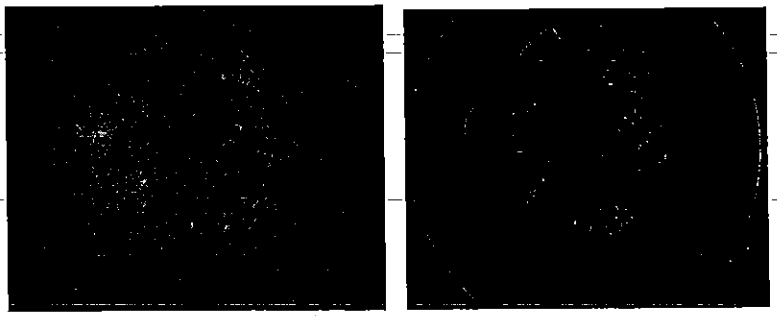
(1) (2)

รูปที่ 4.12 ตัวอย่างภาพที่มีค่า Score = 0.008510 และ %Score = 58.479418



(1) (2)

รูปที่ 4.13 ตัวอย่างภาพที่มีค่า Score = 0.009190 และ %Score = 63.147615

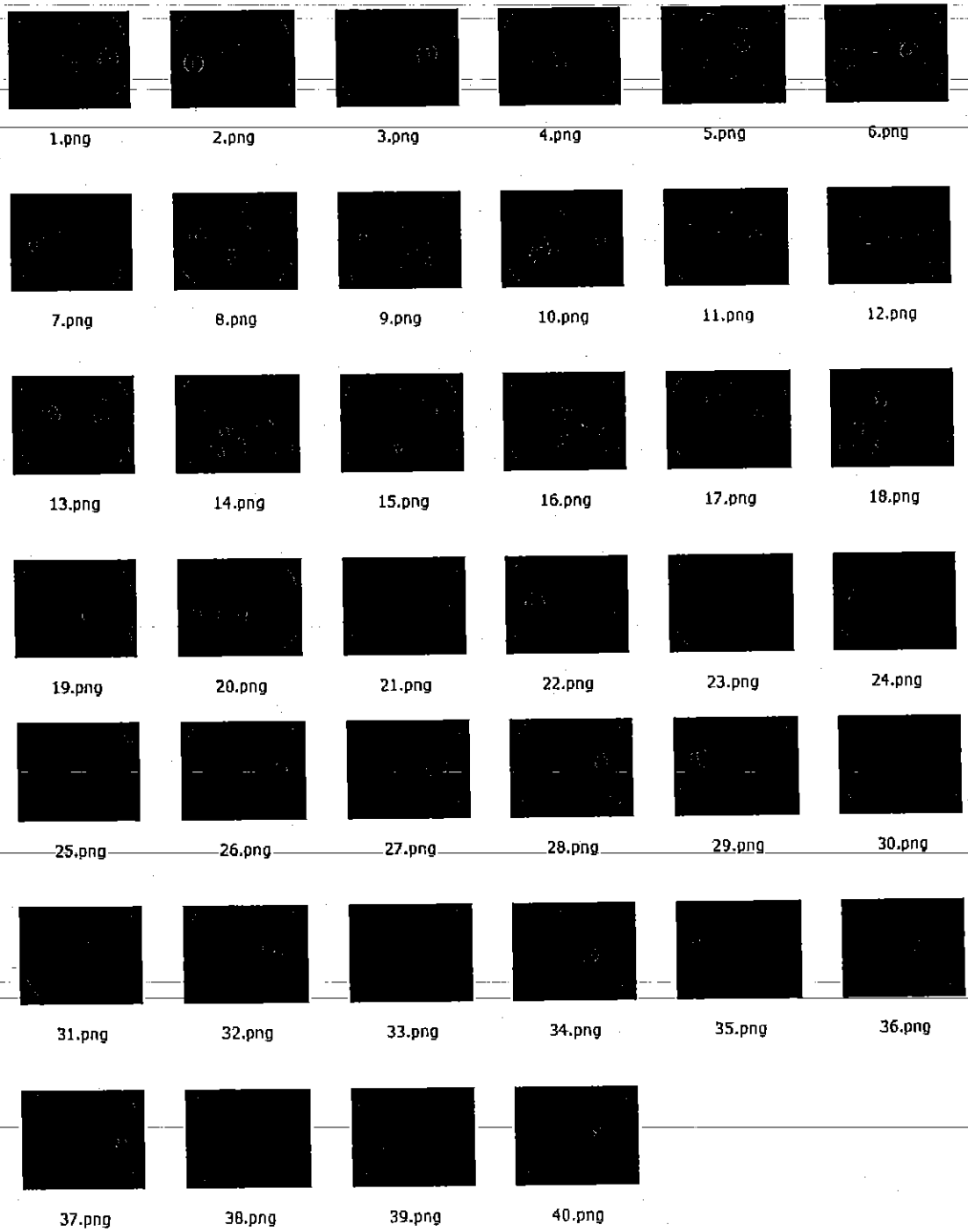


(1) (2)

รูปที่ 4.14 ตัวอย่างภาพที่มีค่า Score = 0.010708 และ %Score = 73.577811

2). ภาพผลการทดลอง Structuring element = 20×20 พิกเซล ทั้งหมด

จากรูปที่ 4.12 เป็นการแสดงภาพผลการทดลองทั้งหมดซึ่งได้จากการใช้ค่า Structuring element เท่ากับ 20×20 พิกเซล โดยทำการทดลองทั้ง 40 ภาพ เพื่อที่จะนำภาพที่ได้ทุกภาพไปทำการวิเคราะห์หาค่า Score และค่า %Score แล้วนำค่าที่ได้มาทำตารางเปรียบเทียบค่า เพื่อดูผลความแตกต่างของเปอร์เซ็นต์จากการใช้ค่า Structuring element แต่ละค่า



รูปที่ 4.12 ภาพผลการทดลองทั้งหมดโดยที่ค่า Structuring element 20×20 พิกเซล

3). ตารางผลการทดลองค่า Structuring element = 20×20พิกเซล

จากการทำ Dilation ซึ่งทดลองใช้ค่า Structuring element เท่ากับ 20×20 พิกเซล จะเห็นได้ว่าค่า Score ที่ได้นั้นจะมีค่าน้อยที่สุดจากค่า Structuring element ทั้งสามค่า เนื่องจากค่า Structuring element เท่ากับ 20×20 พิกเซล นั้นจะมีการกรองมากจนเกินไป ทำให้รายละเอียดหรือพิกเซลสีขาวของภาพจอประสาทตาผิดปกติ ที่ได้ออกมานั้นค่อนข้างที่จะขาดหายไป จึงทำให้ค่า Score ที่ได้มีค่าน้อยที่สุดและเมื่อค่า Score น้อย ก็จะส่งผลให้ค่า % Score น้อยตามไปด้วย ดังนั้นเราจึงไม่ใช้ค่า Structuring element เท่ากับ 20×20 พิกเซล ในการทดลองจริง ซึ่งค่า Score และค่า %Score ที่ได้จะแสดงไว้ดังตารางที่ 4.3

หมายเหตุ : ค่า Score และค่า %Score ที่เป็นสีดำในตารางนั้น คือค่าน้อยกว่า 50% ซึ่งในการทดลองนี้จะให้ถือว่าเป็นภาพจอประสาทตาที่ปกติ

ค่า Score และค่า %Score ที่เป็นสีแดง คือค่ามากกว่า 50% ซึ่งในการทดลองนี้จะให้ถือว่าเป็นภาพจอประสาทตาที่ผิดปกติ

ค่า %Score ที่มีสัญลักษณ์ * นั้นในการทดลองนี้จะให้ถือว่าภาพนั้นเป็นภาพที่ไม่เข้ากลุ่ม หรือ ภาพที่ผิดพลาด

ตารางที่4.3 สรุปค่า Score และค่า %Score ที่ Structuring element = 20×20 พิกเซล

ลำดับ Image	Score	%Score
1	0.007923	54.441135
2	0.009032	62.063577
3	0.006967	47.873431*
4	0.00851	58.479418
5	0.010708	73.577811
6	0.014553	100
7	0.009265	63.665218
8	0.008304	57.05845
9	0.014019	96.332829
10	0.01193	81.981542
11	0.009885	67.923238
12	0.00919	63.147615
13	0.009914	68.128327
14	0.010379	71.321842
15	0.00741	50.915572
16	0.009625	66.140925

ตารางที่4.3 สรุปค่า Score และค่า %Score ที่ Structuring element = 20×20 พิกเซล

17	0.009114	62.630011
18	0.012827	88.143952
19	0.00851	58.479418
20	0.00851	58.479418
21	0.004512	31.00249
22	0.00636	43.703306
23	0.003311	22.755017
24	0.00519	35.660921*
25	0.003774	25.933883
26	0.004318	29.669417
27	0.003915	26.900728
28	0.006958	47.809952
29	0.011304	77.679574
30	0.004277	29.391084
31	0.006172	42.409297*
32	0.004953	34.034865*
33	0.004305	29.581523
34	0.005112	35.128668
35	0.004654	31.979101
36	0.004981	34.230187
37	0.005499	37.785048
38	0.003085	21.197324
39	0.005369	36.89145
40	0.006749	46.374335

4.2 วิเคราะห์ผลการทดลอง

1). ค่า %Score

จากผลการทดลองสามารถสรุปได้ว่าภาพที่มี %Score ตั้งแต่ 50% ขึ้นไปนั้นจะถือว่าเป็นภาพของจอประสาทตาที่มีอาการผิดปกติ โดยเป็นการอ้างอิงเทียบจากเปอร์เซ็นต์ของภาพทั้งหมดในฐานข้อมูลภาพที่มี ซึ่งจะสังเกตได้ว่าภาพที่หาค่า %Score ออกมาแล้วได้น้อยกว่า 50% จะอยู่ในกลุ่มของภาพที่ปกติทั้งหมด และ ภาพที่หาค่า %Score ออกมาแล้วได้ค่ามากกว่า 50% ขึ้นไป จะอยู่ในกลุ่มของภาพที่ผิดปกติทั้งหมด ซึ่งภาพจอประสาทตาที่ผิดปกติและปกตินี้ได้ถูกแยกไว้แล้วในบทที่ 3 รูปที่ 3.2 และ รูปที่ 3.3

และนอกจากนี้จากการคำนวณหาค่า Score และ %Score ในบทที่ 3 ยังสามารถบอกได้ว่า

- ค่า %Score ที่ได้จะเปลี่ยนแปลงไปตามกับค่า Score

- ค่า Score และ %Score ที่ได้ สามารถแบ่งระดับความแตกต่างของความผิดปกติได้พอสมควร

2). ค่า Structuring element

จากผลการทดลองจะทำให้ทราบว่า ค่า Structuring element ที่ใช้ในการปรับหารูปภาพที่ต่างกัน จะทำให้ได้ผลของภาพที่ได้ออกมาต่างกัน ทั้งในด้านการขยาย ความละเอียดและค่าของสิ่งรบกวน

- ค่า structuring element 10 พิกเซล เป็นค่าที่ต่ำที่สุดที่ใช้ในการทดลองซึ่งผลที่ได้ออกมาก็คือ ภาพที่ได้จะมีสิ่งรบกวนหรือสิ่งที่เราไม่ต้องการค่อนข้างมากเกินไป จึงทำให้ภาพที่ได้ออกมานั้น ยังไม่ค่อยเหมาะสมสำหรับการนำไปวิเคราะห์หาค่า Score

- ค่า structuring element 15 พิกเซล เป็นค่าระดับกลางที่ใช้ทดลองซึ่งผลที่ได้ก็คือ ภาพค่อนข้างมีความสมบรูณ์ไม่มีสิ่งรบกวนมากจนเกินไปและพิกเซลสีขาวถูกขยายในระดับที่เหมาะกับการนำไปวิเคราะห์หาค่า Score

- ค่า structuring element 20 พิกเซล เป็นค่าที่สูงที่สุดที่ใช้ในการทดลองซึ่งภาพที่ได้นั้น พิกเซลสีขาวจะถูกขยายมากเกินไป ทำให้สิ่งที่ต้องการนั้นขยายใหญ่มากซึ่งจะทำให้ไม่ค่อยเห็นรายละเอียดของความผิดปกติ และภาพที่ได้ก็จะมีความผิดเพี้ยนไปจากความจริงมากเกินไป

3). ค่าความผิดพลาด (%Error)

- การคำนวณหาค่าความผิดพลาด นั้น จะสามารถหาได้จากการนับจำนวนภาพของค่า %Score ที่ไม่เข้าพวกกับกลุ่มภาพที่แยกไว้จาก รูปที่ 3.2 และ รูปที่ 3.3 ในบทที่ 3 ตัวอย่างเช่น ภาพที่มีลักษณะความผิดปกติน้อย มีบริเวณที่เป็นพิกเซลสีขาวน้อยและอยู่ในกลุ่มของรูปที่ 3.3 ซึ่งเป็นภาพจอประสาทตาที่ปกติ แต่กลับมีค่า %Score สูง ภาพนั้นจะถูกกำหนดให้เป็นภาพความผิดพลาด แล้วนำจำนวนของภาพความผิดพลาดที่หาได้มาหารด้วยจำนวนภาพทั้งหมด คือ 40 ภาพ ก็จะทำให้ได้ ค่าความผิดพลาด (%Error) ออกมา

- ค่าความผิดพลาดจากการใช้ Structuring element ทั้งสามค่า 10, 15 และ 20 พิกเซล นั้นจะมีจำนวนภาพความผิดพลาดเท่ากันคือ 4 ภาพ ซึ่งทำให้ผลของค่าความผิดพลาด (%Error) ออกมาเท่ากันซึ่งมีค่าเท่ากับ 10% แต่ภาพที่เกิดความผิดพลาด ในแต่ละค่านั้นอาจจะเป็นภาพที่แตกต่างกัน

- ภาพที่เกิดความผิดพลาดนั้นอาจเกิดขึ้นจากสาเหตุระดับความสว่างของภาพมีน้อย ภาพนำมาวิเคราะห์เป็นภาพที่ถ่ายมาแบบไม่ชัดเจน

บทที่ 5

บทสรุป

5.1 สรุปผลการทดลอง

ความผิดปกติของลูกตามนุษย์ สามารถหาวิเคราะห์และวินิจฉัยได้จากภาพถ่ายจอประสาทที่ได้รับจากกล้องถ่ายภาพด้วยการใช้เทคนิคการประมวลผลภาพซึ่งประกอบด้วย 3 ขั้นตอน คือ (1) แปลงภาพถ่ายจอประสาทแบบสีให้เป็นภาพระดับเทา (2) ทำการไดเลชัน (dilation) ภาพระดับเทาให้บริเวณส่วนผิดปกติเด่นชัดขึ้น และ (3) นำผลของภาพมาทำการหาขอบภาพเพื่อระบุบริเวณที่ผิดปกติของลูกตาในรูปแบบภาพขาวดำ

จากผลการทดลองและวิเคราะห์ผลการทดลองอัลกอริทึมในการบอกความผิดพลาดของลูกตากับภาพลูกตา จำนวน 40 ภาพ ซึ่งแบ่งเป็นภาพลูกตาปกติ จำนวน 16 ภาพและภาพลูกตาผิดปกติ จำนวน 24 ภาพ สรุปได้ว่า ภาพลูกตาที่มีค่า %Score น้อยกว่า 50% จะจัดอยู่ในกลุ่มของภาพที่ปกติ และภาพลูกตาที่มีค่า %Score มากกว่า 50% จะจัดอยู่ในกลุ่มของภาพที่ผิดปกติ โดยการใช้ภาพหน่วยโครงสร้างในกระบวนการไดเลชันที่มีขนาดเท่ากับ 15 พิกเซล ทำให้ได้ค่า %Score ในการทำนายผลความผิดปกติของลูกตาค่อนข้างมีความสมบรูณ์ไม่มีสิ่งรบกวนมากจนเกินไปและพิกเซลสีขาวจะถูกขยายในระดับที่เหมาะกับค่าความผิดพลาด (%Error) ในการทำนายอาการผิดปกติของลูกตาจริงอยู่ในระดับ 10%

5.2 ปัญหาและอุปสรรคของโครงงาน

ในการดำเนินโครงงานนี้ คณะผู้จัดทำโครงงานได้พบปัญหาและอุปสรรคในเรื่องเกี่ยวกับการหาผู้เชี่ยวชาญในการตรวจสอบผลของภาพลูกตาว่าสิ่งใดผิดปกติและไม่ผิดปกติตลอดจนมีความยากลำบากในการหาภาพลูกตามาทำการทดลองและวิเคราะห์ผลการทดสอบโดยรวมของปัญหาและอุปสรรค พอสรุปได้ดังนี้

- อัลกอริทึมที่พัฒนาขึ้นนี้ จะตรวจสอบได้เฉพาะอาการไขมันขึ้นที่จอประสาทตา
- สีของภาพลูกตาที่ผิดปกติกับไม่ผิดปกติมีค่าใกล้เคียงกันทำให้มีผลต่อการหาขอบภาพ
- ขาดความรู้ในการประมวลผลภาพและความชำนาญในการใช้โปรแกรมแม็ทแล็บ
- ขาดความรู้และแหล่งอ้างอิงในการวินิจฉัยภาพจอประสาทตาที่ผิดปกติและปกติไม่เพียงพอ
- ทักษะการเขียนโปรแกรมแม็ทแล็บมีน้อยทำให้การรันโปรแกรมนานมาก

5.3 ข้อเสนอแนะและวิธีการแก้ไข

จากปัญหาและอปสรรคในหัวข้อย่อยที่ 5.3 ทางคณะได้มีแนวทางในการดำเนินการแก้ไขตลอดจนข้อเสนอแนะ ดังนี้

- หากต้องการพัฒนาอัลกอริทึมที่พัฒนาในการตรวจสอบลักษณะอาการได้มากกว่าหนึ่งอาการ ควรปรับปรุงและหาวิธีการขอบภาพที่ดีกว่าเดิม
- ในปัญหาเรื่องสีของภาพลูกตาที่ผิดปกติกับไม่ผิดปกติมีค่าใกล้เคียงกัน ควรทำการปรับปรุงภาพก่อนเข้าขั้นตอนการตรวจสอบทุกภาพที่ได้รับก่อน
- ในปัญหาการขาดความรู้ในการประมวลผลภาพและความชำนาญในการใช้โปรแกรมพร้อมกับในปัญหาเรื่องการขาดความรู้และแหล่งอ้างอิงในการวินิจฉัยภาพจอประสาทตาที่ผิดปกติและปกติไม่เพียงพอ และทักษะการเขียนโปรแกรมแม็ทแล็บมีน้อยทำให้การรันโปรแกรมนานมาก คณะผู้จัดทำได้ศึกษาหาความรู้เพิ่มเติมจากตำราและผู้เชี่ยวชาญตลอดจนอาจารย์ที่ปรึกษา

บรรณานุกรม

[1] A. Hadid A, M. Pietik¨ainen, and B. Martinkauppi B. **Color-based face detection using skin locus model and hierarchical filtering.** In *Proc.* 16th International Conference on Pattern Recognition, pages 196–200.

[2] E. Bailliere, S. Bengio, F. Bimbot, M. Hamouz, J. Kittler, J. Mariethoz, J. Matas, K. Messer, V. Popovici, F. Poree, B. Ruiz, and J.P. Thiran. **The BANCA database and evaluation protocol.** In Proc. of the Int. Conf. on Audio- and Video-Based Biometric Person Authentication, pages 625–638, 2003.

[3] Alan D. Fleming, Sam Philip, Keith A. Goatman, John A. Olson, and Peter F. Sharp. **Automated microaneurysm detection using local contrast normalization and local vessel detection.** IEEE Transactions in Medical Imaging, 25(9):1223–1232, September 2006.

[4] Harihar Narasimha-Iyer, Ali Can, Bandrinath Roysam, Charles V. Stewart, Howard L.Tanenbau, Anna Majerovics, and Hanumant Singh. **Robust detection and classfication of longitudinal changes in color retinalfundus images for monitoring diabetic retinopathy.** IEEE Transactions on Biomedical Engineering, 53(6):1084–1098, June 2006.

[5] M. Niemeijer, M. D Abr`amoff, and B. van Ginneken. **Automatic detection of the presence of bright lesions in color fundus photgraphs.** In Proceedings of IFMBE the 3rd European Medical and Biological Engineering Confrence, volume 11 of 1, pages 1823–2839, Prague and Czech Republic, November 2005.

[6] M. Niemeijer, B. van Ginneken, J. Staal, M. S. A. Suttorp-Schulten, and N. D. Abramoff. **Automatic detection of red lesion in digital color fundus photographs.** IEEE Transactions on Medical Imaging, 24(5):584–592, May 2005.

[7] Alireza Osareh, Majid Mirmehdi, Barry Thomas, and Richard Markham. **Classification and localization of diabetic-related eye disease.** In Proc. of 7th European Conference on Computer Vision (ECCV), pages 502–516, 2002.

[8] C. I. S`anchez, R. Hornero, M. I .L`opez, and J. Poza. **Retinal image analysis to detect and quantify lesions associated with diabetic retinopathy.** In Proceedings of the 26th Annual International Confrence of the IEEE Engineering in

Medicine and Biology Society (EMBS), pages 1624–1627, San Francisco, CA, USA, September 2004.

[9] C. Sinthayothin, J. F. Boyce, T. H. Williamson, E. Mensah, S.Lal, and D. Usher. **Automated detection of diabetic retinopathy on digital fundus images.** Diabetic Medicine, 19:105–112, 2002.

[10] T. Walter, J.-C. Klein, P. Massin, and A. Erginay. **A contribution of image processing to the diagnosis of diabetic retinopathy - detection of exudates in color fundus images of the human retina.** IEEE Transactions on Medical Imaging, 21:1236–1243, October 2002.

[11] Xiaohoui Zhang and Opas Chutape. **A SVM approach for detection of hemorrhages in background diabetic retinopathy.** In Proceedings of International Joint Confrence on Neural Networks, pages 2435–2440, Montreal and Canada, July 2005. July 2005.

[12] Xiaohui Zhang and O. Chutatape. **Detection and classification of bright lesions in colour fundus images. In Proceedings of IEEE Internationl Confrence on Image Processing (ICIP)**, volume 1, pages 139–142, October 2004.

[13] มหาวิทยาลัยนเรศวร.2543. **คู่มือการเขียนโครงงานนิสิต.**ครั้งที่ 2.พิษณุโลก :ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

[14] กนกวรรณ ชำนาญจุ้ย,วาสนา วงศ์ษา.**"การตรวจหาลายภาพ ."** วิทยานิพนธ์ วศ.ม. พิษณุโลก : มหาวิทยาลัยนเรศวร, 2549.

ภาคผนวก

ภาคผนวก ก

โค้ดโปรแกรม

1. โค้ดหลัก (Main Code)

```
clear all % เคลียร์ค่าตัวแปรทั้งหมดในโปรแกรม
clc % ลบ command window ทั้งหมดในโปรแกรม
close all % ปิดหน้าต่างทั้งหมดในโปรแกรม
% path ที่เก็บข้อมูลของรูปภาพที่จะนำมาวิเคราะห์
TrainDatabasePath = uigetdir('E:\MukDaRin\โปรเจค
1\db1\diaretdb1_v_1_1\resources\ddb1_fundusimages', 'Select training database path' );
% path ที่เก็บข้อมูลของรูปภาพที่ผ่านการวิเคราะห์ด้วย dilation และ edge แล้ว
EvalhistDatabasePath = uigetdir('E:\MukDaRin\โปรเจค
1\db1\diaretdb1_v_1_1\resources\evalresults_hist', 'Select test database path');
%%%%%%%%%%%%Delete file in EvalhistDatabasePath path %%%%%%%%%%
imgfile = '\*.png'; % ไฟล์รูปภาพทั้งหมดใน EvalhistDatabasePath ที่ต้องการลบก่อนรันโปรแกรม
txtfile = '\*.txt'; % ไฟล์ text ทั้งหมดใน EvalhistDatabasePath ที่ต้องการลบก่อนรันโปรแกรม
EvalImage = strcat(EvalhistDatabasePath,imgfile); % เก็บค่าของ path และรูปภาพปลายทาง
EvalFile = strcat(EvalhistDatabasePath,txtfile); % เก็บค่าของ path และ text ปลายทาง
delete(EvalImage); % ลบข้อมูลของรูปภาพ
delete(EvalFile); % ลบข้อมูลของ text file
%%%%%%%%%%%% Create Database for Image Colour Histogram %%%%%%%%%%
% ทำการรัน function ในการทำ dilation และ edge detection แล้วเก็บเป็น database
Target = createdatabase_evaldiaretdb1_edge(TrainDatabasePath, EvalhistDatabasePath);
%%%%%%%%% Evaluate Image Colour Histogram for Diabetic Retinopathy %%%%%%%%%
% ทำการรันในส่วนของการคำนวณหาค่าเปอร์เซนต์ของความผิดปกติของภาพลูกตา
evaldiaretdb1_testimage_test_edge;
```

.....

2. โค้ดการทำ Dilationและ Edge Detection

```
function T = createdatabase_evaldiaretdb1_edge(TrainDatabasePath, EvalhistDatabasePath)
%%%%%%%%%%%%%%%%%%%% File management %%%%%%%%%%%%%%%%%%%%
TrainFiles = dir(TrainDatabasePath); % อ่านค่า path ปลายทางมาเก็บที่ตัวแปร TrainFiles
Train_Number = 0; % กำหนดค่าตัวแปร Train_Number
% ทำการวนลูปหาค่าของจำนวนของไฟล์ทั้งหมดที่มีอยู่ใน path
for i = 1:size(TrainFiles,1)
   if
not(strcmp(TrainFiles(i).name,'.')|strcmp(TrainFiles(i).name,'..')|strcmp(TrainFiles(i).name,'Thum
bs.db'))
      Train_Number = Train_Number + 1; % Number of all images in the training database
   end
end
%%%%%% Mark image Dilation and Edge Detection to Evaluate Database %%%%%%%%%
T = []; % กำหนดค่าเมตริกซ์ว่างให้กับตัวแปร T
for i = 1 : Train_Number
   % แสดงค่าของรูปภาพว่ากำลังทำอยู่ที่ภาพใด
   fprintf('\nDilation and Edge Detection Image Number: %d',i);
   str1 = int2str(i); % แปลงค่าจาก integer เป็น charecter
%%%%%%%%% Write text file to Evaluate Path in testdatafile.txt %%%%%%%%%
   txtfile = 'testdatafile.txt'; % เก็บค่าตัวแปร text file
   txtfilestr = strcat('\',txtfile); % เก็บค่าตัวแปร \text file โดยเก็บเป็น charecter
   txtfileopen = strcat(EvalhistDatabasePath,txtfilestr); % เก็บค่าตัวแปรและ path ปลายทาง
   strfile = strcat(str1,'.png'); % เก็บค่าของ text เช่น 1.png
   fid = fopen(txtfileopen,'a'); % ทำการเปิดการเขียนไฟล์ text ไปยัง path ปลายทาง
   fprintf(fid,'%s ',strfile); % ทำการเขียนไฟล์ text ไปยัง path ปลายทางโดยเขียนต่อในไฟล์เดียว
   fclose(fid); % ทำการปิดการเขียนไฟล์ text ไปยัง path ปลายทาง
%%%%%%%%%%%%%%%%% Write image to Evaluate Path %%%%%%%%%%%%%%%%%%
```

```
str = strcat('\',str1,'.png'); % เก็บค่าตัวแปรรูปภาพ เช่น 1.png
streval = strcat('\',str); % เก็บค่าตัวแปรรูปภาพ เช่น \1.png
str = strcat(TrainDatabasePath,str); % เก็บค่า path ของรูปภาพต้นทาง
strhist = strcat(EvalhistDatabasePath,streval); % เก็บค่า path ของรูปภาพปลายทาง
%%%%%%%%%%%%%%%% Perform Dilation and Edge Detection %%%%%%%%%%
img = imread(str); % อ่านรูปภาพเข้ามาในโปรแกรม
se = strel('ball',15,15); % ใช้วิธีการ ball ในการทำ dilation รูปภาพ
img2 = imdilate(img,se); % ทำการ dilation รูปภาพต้นทาง
img3 = rgb2gray(img2); % ทำการแปลงรูปภาพที่ผ่านการ dilation แล้วเป็นรูป gray level
result = edge(img3,'roberts'); % นำภาพที่ผ่านการ dilation แล้วมาทำ edge detection
imwrite(result,strhist); % ทำการเขียนรูปภาพผลลัพธ์ที่ได้ลงใน path ปลายทาง
end
```

.....

3. โค้ดเก็บไฟล์ปลายทาง

```
% path ที่ใช้เก็บภาพที่ใช้ในการทำ mask
diaretdb1.fmaskpath = 'E:\MukDaRin\โปรเจค
1\db1\diaretdb1_v_1_1\resources\images\ddb1_fundusmask\';
% file ที่ใช้เก็บภาพปลายทางที่จะนำมาวิเคราะห์หาเปอร์เซนต์ความผิดปกติ
diaretdb1.evalfiles = 'E:\MukDaRin\โปรเจค
1\db1\diaretdb1_v_1_1\resources\evalresults_hist\testdatafile.txt';
% path ที่ใช้เก็บภาพปลายทางที่จะนำมาวิเคราะห์หาเปอร์เซนต์ความผิดปกติ
diaretdb1.evalfilepath = {
'E:\MukDaRin\โปรเจค 1\db1\diaretdb1_v_1_1\resources\evalresults_hist\'; ...
};
```

.....

4. โค้ดหาค่า Score และ % Score

```
clear all % เคลียร์ค่าตัวแปรทั้งหมดในโปรแกรม
clc % ลบ command window ทั้งหมดในโปรแกรม
close all % ปิดหน้าต่างทั้งหมดในโปรแกรม
evaldiaretdb1_configure_test_edge; % อ่านค่าจากไฟล์ configure
% อ่านค่าจากไฟล์ text ที่เก็บภาพมาเก็บไว้ที่ตัวแปร imfiles
imfiles     = textread(diaretdb1.evalfiles,'%s','delimiter',' ');
% เก็บจำนวนของภาพทั้งหมดใน path ปลายทางที่ต้องการนำมาวิเคราะห์
numfindings  = size(diaretdb1.evalfilepath,1);
% เก็บจำนวนภาพที่ต้องการใน path ปลายทางที่ต้องการนำมาวิเคราะห์
numfiles     = size(imfiles,1);
% แสดงจำนวนของภาพที่ต้องการนำมาวิเคราะห์
fprintf('Number files: %d',numfiles);
fprintf('\n');
try
 display(['Reading fundus mask file... ']);
 % อ่านภาพ mask เข้ามาเก็บที่ตัวแปร fmaskim
 fmaskim  = imread([diaretdb1.fmaskpath 'fmask.tif']);
catch
 % อ่านภาพ mask เข้ามาเก็บที่ตัวแปร fmaskim ถ้าไม่เจอภาพ mask จะแสดงค่า error
 display(['Error occured when reading file... ']);
end
for j = 1:numfindings
  evaldata(j).numfindings  =  numfindings; % เก็บค่า numfindings ในตัวแปรอาร์เรย์
  evaldata(j).numfiles     =  numfiles; % เก็บค่า numfiles ในตัวแปรอาร์เรย์
  evaldata(j).numfmaskpix  =  sum(sum(fmaskim>0)); % เก็บค่า pixel ของภาพ mask
  evaldata(j).numevalpix = 0; % เก็บค่า 0 ในตัวแปร numevalpix
end
```

```
for j = 1:numfindings
  for i = 1:numfiles
    evalfile = deblank(imfiles{i}); % ลบ whitespace และ null character
    try
    % อ่านภาพเข้ามาเก็บไว้ที่ตัวแปร evalim
    evalim = imread([diaretdb1.evalfilepath{j} evalfile]);
    evalim = double(evalim); % แปลงค่ารูปภาพเป็นชนิด double แล้วไปเก็บในตัวแปร evalim
    catch
    % ถ้าการอ่านภาพเข้ามาในโปรแกรมแล้ว error แล้วจะแสดงข้อความว่าเกิด error
    display(['Error occured when reading files... ']);
    end
    % หาผลรวมของ pixel ในแต่ละภาพ โดยเลือกเฉพาะ pixel ที่มีค่าเป็น 1
    evaldata(j).numevalpix(i) = sum(sum(evalim >0));
    % หาค่า score ของภาพโดยนำผลรวม pixel ของแต่ละภาพหารด้วยผลรวม pixel
    % ของภาพ mask
    evaldata(j).numevalscore(i) = sum(sum(evalim >0))./evaldata(j).numfmaskpix;
    % หาค่า score ของภาพโดยนำผลรวม pixel ของแต่ละภาพหารด้วยผลรวม pixel
    % ของภาพ mask
    result = evalim > 0;
    evaldata(j).score(i,:) = sum(result(:))./evaldata(j).numfmaskpix;
  end
%%%%%% Evaluation for Persent of image may be diabetic retinopathy %%%%%%%
  score = evaldata(j).score; % เก็บค่า score ในตัวแปร score
  scorethres = unique(evaldata(j).score); % เก็บค่า unique score ในตัวแปร scorethres
  max_score = max(score); %หาค่า max score แล้วเก็บในตัวแปร max_score
  for k = 1:size(score(:),1);
  percent_score = score(k)*100/max_score; % คำนวณหาเปอร์เซนต์ความผิดปกติในแต่ละภาพ
  % แสดงค่า score และผลลัพธ์ของเปอร์เซนต์ความผิดปกติในแต่ละภาพ
```

```
fprintf('\rEvaluating image number: %d   ',k);
fprintf('\nDiabetic Retinopathy Score = %f',score(k));
fprintf('\nDiabetic Retinopathy Percent Score = %f ',percent_score);
fprintf('\n');
end
% แสดงค่า unique score ในแต่ละภาพ
scorethres = [0; scorethres];
fprintf('\nScorethres Value = %f',scorethres);
fprintf('\n');
end
```

ภาคผนวก ข

คำสั่ง Matlab ที่ใช้ในการเขียนโปรแกรม

1. ตัวดำเนินการ เครื่องหมายพิเศษ และตัวดำเนินการเปรียบเทียบตรรกะ

โปรแกรม Matlab ได้กำหนดตัวดำเนินการ เครื่องหมายเปรียบเทียบ และตัวดำเนินการเปรียบเทียบทางตรรกะมาใช้ในการคำนวณเพื่ออำนวยความสะดวกในการเขียนโปรแกรม ดังนี้

ตัวดำเนินการเมตริกซ์

ลักษณะดำเนินการ	ตัวดำเนินการ	รูปแบบของ Matlab
การบวก	+	A+B
การลบ	-	A-B
การคูณ	*	A*B
การคูณเชิงสมาชิก	.*	A.*B
การหารทางขวา	/	A/B
การหารทางซ้าย	\	A\B
การหารเชิงสมาชิก	./	A./B
การยกกำลัง	Ab	A^B
การยกกำลังสมาชิก	A.b	A.^B

2. ฟังก์ชันทั่วไป

Matlab มี function ทางคณิตศาสตร์พื้นฐาน เพื่อสะดวกแก่การใช้งานซึ่งได้แก่

abs(x) หาค่า absolute ของ x

sqrt(x) หาค่า รากที่ 2 ของ x

round(x) ทำให้ x เป็นจำนวนเต็ม โดยปัดค่าให้เป็นจำนวนเต็มที่ใกล้ x ที่สุด

fix(x) ทำให้ x เป็นจำนวนเต็ม โดยปัดค่า x ให้เป็นจำนวนเต็มที่ใกล้ศูนย์ที่สุด

floor(x) ทำให้ x เป็นจำนวนเต็ม โดยปัดค่า x ให้เป็นจำนวนเต็มที่ใกล้ x ไปทาง - infinity

ceil(x) ทำให้ x เป็นจำนวนเต็ม โดยปัดค่า x ให้เป็นจำนวนเต็มที่ใกล้ x ไปทาง + infinity

sign(x) บอกเครื่องหมายของ x โดยจะเป็น -1 ถ้า x < 0, เป็น 1 ถ้า x > 0 และเป็น 0 ถ้า x = 0

rem(x,y) หาเศษที่ได้จากการหาร x ด้วย y หรือ เศษของ x/y

exp(x) หาค่า e^x (เอกโพเนนเชียล)

log(x) หาค่า ln(x) หรือ natural logarithm ของ x

log10(x) หาค่า log10 x หรือ logarithm ฐาน 10 ของ x

3. ฟังก์ชันทางสถิติ

สำหรับฟังก์ชันทางสถิติที่ใช้ใน MATLABจะประกอบด้วยฟังก์ชันหลัก ๆ ดังต่อไปนี้

3.1 mean(x) คือ การหาค่าเฉลี่ยของ A ค่าเฉลี่ยจะมีค่าเท่ากับผลรวมของข้อมูลทั้งหมดแล้วหารด้วยจำนวนข้อมูลซึ่งสามารถอธิบายได้ด้วยสมการดังนี้

$$\overline{X} = \frac{\sum X_i}{n}$$

เมื่อ $\overline{X}$ คือค่าเฉลี่ยของกลุ่มตัวอย่าง

$\overline{X}$ คือค่าข้อมูลแต่ละตัว

n คือจำนวนข้อมูลทั้งหมด

ถ้า A เป็นเวกเตอร์ค่าผลลัพธ์ที่ได้จากการคำนวณจะเป็นค่าสเกลาร์ ถ้า A เป็นเมตริกซ์ค่าผลลัพธ์ที่ได้จากการคำนวณจะเป็นเวกเตอร์แถวที่ได้จากกค่า เฉลี่ยในแต่ละหลักของเมตริกซ์ A ตัวอย่างเช่น

```
>> x=[1 9 8;2 12 4;4 7 6]
>> mean(x)
>> ans =
2.3333 9.3333 6.0000
```

3.2 sum(x) ถ้า x เป็นเวกเตอร์ จะได้ผลเป็นการรวมค่าทั้งหมดของ Element ใน x แต่ถ้า x เป็นเมตริกซ์จะได้ผลเป็นเวกเตอร์แถวที่บรรจุผลบวกแต่ละหลักของเมตริกซ์ x ไว้

3.3 max(x) โปรแกรม Matlabจะให้ค่าที่มากที่สุดของ x ถ้า x เป็นเมตริกซ์จะได้ผลเป็นเวกเตอร์แถวที่บรรจุค่าสูงสุดในแต่ละหลัก ของ x ถ้า x เป็นจำนวนเชิงซ้อนจะได้ผลเป็นค่าที่มีขนาดสูงสุด

3.3 min(x) โปรแกรม Matlabจะให้ค่าที่น้อยที่สุดของ x ถ้า x เป็นเมตริกซ์จะได้ผลเป็นเวกเตอร์แถวที่บรรจุค่าต่ำสุดในแต่ละหลัก ของ x ถ้า x เป็นจำนวนเชิงซ้อนจะได้ผลเป็นค่าที่มีขนาดต่ำสุด

```
>> x=[4;1;6]
>> max(x)
>> ans =
6
>> min(x)
>> ans =
1
```

4. คำสั่งที่ควบคุมการทำงานของ M-file

เช่นเดียวกับการเขียนโปรแกรม ทั่วไป Matlabมีคำสั่งที่ใช้ในการควบคุมขั้นตอนการทำงาน แต่ข้อแตกต่างจากการใช้ภาษาพื้นฐานเหล่านั้น คือ

- ใน Matlabไม่มี line number ดังนั้นการเขียนโปรแกรมหากต้องการข้ามชุดคำสั่งใดอาจต้องใช้ข้อความทางตรรกเข้าช่วยหรือใช้วิธีการใช้ subroutine หรือ function เข้าช่วย
- ใน Matlabไม่ต้องมีการบอกว่าตัวแปรใดเป็น string, integer หรือถ้าเป็นเลขทศนิยมก็ไม่จำเป็นต้องบอกว่าจะมี precision แบบใดยกเว้นว่าต้องการกำหนดเอง
- ไม่จำเป็นต้องจอง array หรือจอง dimension ของตัวแปร
- สามารถเรียกใช้ function ต่าง ๆ ที่มีอยู่ใน Matlabหรือที่สร้างขึ้นเองได้ตลอดเวลา หรือไม่จำเป็นต้องเขียน subroutine ขึ้นมาใหม่
- Matlabทำงานได้โดยไม่ต้อง compile ก่อน ดังนั้น โดยทั่วไป Matlabจะหยุดทำงานทันทีเมื่อพบ error

คำสั่งที่ใช้ควบคุมการทำงานหลัก ๆ มีดังนี้

4.1 คำสั่ง For loop

สำหรับ For loop นั้นก็จะเหมือนกับ For loop (หรือ Do loop) ของภาษาอื่น ๆ โดยมีโครงสร้างเป็น

for m = a: b

ชุดคำสั่งหรือการคำนวณ

end

โดยคำสั่งนี้จะกำหนดให้ Matlabทำเป็นวงรอบ (loop) โดยเริ่มจากค่า m = a แล้วทำบรรทัดต่อมา คือ คำนวณชุดคำสั่งไม่ว่าจะมีชุดคำสั่งเท่าใดก็ตาม จากนั้นเมื่อพบคำสั่ง end แล้ว Matlabก็จะกลับไปเริ่มต้นที่คำสั่ง for อีก แล้วทำงานต่อโดยใช้ค่า m = a + 1 ไปเรื่อย และ จะเลิกทำงานเมื่อคำนวณค่าสุดท้ายที่ m = b ซึ่ง ในกรณีการใช้คำสั่งนี้ a และ b จะเป็นจำนวนเต็ม หรือหากใช้ for **m = a:b:c** จะเป็นการทำค่า l เริ่มจาก a ถึง c โดยเพิ่มขึ้นครั้งละ b

4.2 คำสั่ง If Statement

สำหรับการใช้ คำสั่ง If มีรูปแบบดังนี้

if เงื่อนไข

ชุดคำสั่ง

end

โดยถ้าเงื่อนไขเป็นจริง Matlabจะคำนวณชุดคำสั่ง แต่ถ้าไม่เป็นจริง Matlabจะข้ามไปทำคำสั่งในบรรทัดที่ต่อจาก end ต่อไป ในเงื่อนไขหรือ condition จะต้องมีค่าเป็นจริงหรือเท็จ เท่านั้น เครื่องหมายที่ใช้เปรียบเทียบ ส่วนใหญ่เป็นดังนี้

ตารางแสดง เครื่องหมายที่ใช้เปรียบเทียบ

ความหมาย	สัญลักษณ์คณิตศาสตร์	Matlab
เท่ากับ	=	==
ไม่เท่ากับ	≠	~=
มากกว่า	>	>
น้อยกว่า	<	<
มากกว่าหรือเท่ากับ	≥	=>
น้อยกว่าหรือเท่ากับ	≤	=<
และ	AND	&
หรือ	OR	|

การใช้คำสั่ง If-Elseif-Else การใช้ else และ elseif นั้นมีรูปแบบดังนี้

if เงื่อนไขที่ 1

ชุดคำสั่งที่ 1

elseif เงื่อนไขที่ 2

ชุดคำสั่งที่ 2

elseif เงื่อนไขที่ 3

ชุดคำสั่งที่ 4

.....

else เงื่อนไขที่ n

ชุดคำสั่งที่ n

end

ในกรณีนี้ถ้าเงื่อนไขที่ 1 เป็นจริง จะทำชุดคำสั่งที่ 1 แล้วมาที่ end แต่ถ้าไม่เป็นจริงจะพิจารณาเงื่อนไขที่ 2 ถ้าเป็นจริง จะทำชุดคำสั่งที่ 2 แล้วมาที่ end แต่ถ้ายังไม่เป็นจริงอีกจะพิจารณาเงื่อนไขที่ 3 ถ้าเป็นจริงจะทำชุดคำสั่งที่ 3 แล้วไปที่ end ทำเช่นนี้ ไปเรื่อยๆ จนกระทั่งถึงเงื่อนไขที่ n ถ้าไม่มีเงื่อนไขใดเลยที่เป็นจริง ก็จะไม่ทำชุดคำสั่งใดแล้วมาที่ end เลย

4.3 คำสั่ง While Loop

สำหรับ while loop นั้นก็จะคล้าย ๆ กับ for loop จะต่างกันที่ while loop นี้จะไม่กำหนดจำนวนรอบเหมือนกับ for loop แต่จะเป็นการวน loop ไปเรื่อย ๆ ตราบเท่าที่เงื่อนไขที่ให้ยังเป็นจริง อยู่รูปแบบของ while loop คือ

while เงื่อนไข

ชุดคำสั่ง

end

โดย loop นี้จะดำเนินไปเรื่อย ๆ จนกว่าเงื่อนไขนี้จะเป็นเท็จ เมื่อเงื่อนไขเป็นเท็จจะมาทำที่คำสั่ง end แล้วไปทำคำสั่งบรรทัดต่อไป

5. เมตริกซ์

โปรแกรม Matlabจะมีฟังก์ชันต่างๆ สำหรับสร้างอาร์เรย์มาตรฐานเพื่อความรวดเร็วและสะดวกต่อการใช้งาน อาร์เรย์ที่เป็นเมตริกซ์พิเศษเหล่านี้ ประกอบด้วยเมตริกซ์ที่เป็นศูนย์ทั้ง หมด เมตริกซ์ที่เป็นหนึ่งทั้งหมด เมตริกซ์เอกลักษณ์ เมตริกซ์ของการสุ่มจำนวน เมตริกซ์แนวทแยงมุม และเมตริกซ์ที่ให้ค่าคงที่เฉพาะ ตัวอย่างเช่น

zeros(n) คำสั่งนี้ยะใช้สร้างเมตริกซ์ศูนย์ซึ่งเป็นเมตริกซ์จัตุรัสที่มีขนาด [n x n]

zeros(m,n) เป็นการสร้าง matrix ขนาด [m x n] ที่มีทุก element เท่ากับศูนย์

ones(m,n) เป็นการสร้าง matrix ขนาด [m x n] ที่มีทุก element มีค่าเท่ากับ 1

eye(m,n) เป็นการสร้าง matrix ขนาด [m x n] ที่มี element ในแนว diagonal เท่ากับ 1 นอกนั้นเป็นศูนย์ ซึ่ง matrix นี้ ไม่จำเป็นต้องเป็น square matrix

magic(N) เป็นการสร้าง square matrix ขนาด [N x N] จากจำนวนเต็ม 1 จนถึง N2 โดยที่ผลรวมของ element ในแต่ละ row, column และใน diagonal จะมีค่าเท่ากัน สำหรับ N ที่มีค่าเท่ากับ 1, 3, 4, 5,...

size(A) คำสั่งนี้ใช้หาขนาดของเมตริกซ์ A ใดๆ

cell array ใช้ในการสร้าง cell array ที่ว่างเปล่าสามารถทำได้โดยใช้คำสั่ง cell สร้างจะได้ในแต่ละ element ของ matrix เป็น matrix

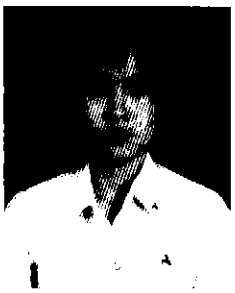
6. การเขียนฟังก์ชัน

Function คือ คำสั่งในการสร้างฟังก์ชันใน M-file ตัวอย่างเช่น

```
function [mean,stdev] = stat(x)
n = length(x);
mean = sum(x)/n;
stdev = sqrt(sum((x-mean).^2/n));
```

ห้องสมุดคณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ประวัติผู้เขียนโครงงาน



ชื่อ นางสาวมุดรินทร์ พิมพร

ภูมิลำเนา 9 หมู่ 11 ต.หนองกุง อ.โนนคูณ จ.ศรีสะเกษ

ประวัติการศึกษา

- จบมัธยมศึกษาจากโรงเรียนพระสุธรรมยานเถระวิทยา จังหวัดอุทัยธานี
- ปัจจุบันกำลังศึกษาอยู่ระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: **pimporn_yah@hotmail.com**

ชื่อ นางสาวสิริพร หว่างสิงห์

ภูมิลำเนา 174 หมู่ 15 ถนนสระบุรี-หล่มสัก ต.หนองไผ่ อ.หนองไผ่ จ.เพชรบูรณ์ 67140

ประวัติการศึกษา

- จบมัธยมศึกษาจากโรงเรียนพระสุธรรมยานเถระวิทยา จังหวัดอุทัยธานี
- ปัจจุบันกำลังศึกษาอยู่ระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: **mint_siriporn9@hotmail.com**