

การเปรียบเทียบและตรวจสอบลายนิ้วมือ

Compare and check fingerprint



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ ปีการศึกษา 2545 เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2545

ภาควิชา อิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การเปรียบเทียบและตรวจสอบลายนิ้วมือ

ผู้จัดทำ

1. นาย พิชณนาท จิวลิตธิประไพ 43015268

2. นาย วสุ เฟื่องสวัสดิ์ 43015275



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเปรียบเทียบและตรวจสอบลายนิ้วมือ

นาย พิษณุ นาด จิวสิทธิประไพ
นาย วสุ เฟื่องสวัสดิ์
ดร.ยุทธนา คิดใจเดียว
อาจารย์ที่ปรึกษา
ปีการศึกษา 2545

บทคัดย่อ

การพิสูจน์ชี้ตัวบุคคลโดยใช้ลายนิ้วมือ เป็นสิ่งที่ต้องการในงานหลายๆ ด้าน เช่น งานทะเบียนราษฎรของกรมการปกครอง งานอาชญากรรมของกรมตำรวจ และงานด้านการรักษาความปลอดภัย แต่เนื่องจากในปัจจุบันประชากรมีจำนวนเพิ่มมากขึ้น การพิสูจน์แยกแยะโดยมนุษย์จะทำให้เกิดความล่าช้าและความยากลำบาก ด้วยเหตุนี้จึงได้มีการพัฒนาระบบจดจำลายนิ้วมืออัตโนมัติขึ้น ในบทความนี้ได้ประยุกต์ระบบทางด้าน Digital Image Processing กับภาพลายนิ้วมือ เพื่อพัฒนาอัลกอริทึมสำหรับตีเทคโครงสร้างลายเส้นสัน บนนิ้วมือที่ได้มีการหักเหอย่างอิสระจากการถ่ายภาพลายนิ้วมือจริง โดยใช้กล้องวิดีโอแล้วเก็บข้อมูลเป็นไฟล์ไว้ในฐานข้อมูล แล้วทำการเปรียบเทียบระหว่างลายนิ้วมือใหม่ลายนิ้วมือหนึ่งกับลายนิ้วมือเก่าลายหนึ่งว่าเป็นลายนิ้วมือเดียวกันหรือไม่ โดยตรวจสอบลักษณะลายนิ้วมือแล้วทำการตรวจสอบจุดสำคัญของแต่ละลายนิ้วมืออีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Compare and check fingerprint

Mr. Pissanunat Chiwsitthipapai

Mr. Vasu Fuengsawat

Advisor : Mr. Yuttana Kidchaidiaw

2003

ABSTRACT

Fingerprint identification has been used for many classes of works such as registration law enforcement and security. At this present since a number of citizen is large the manual fingerprint recognition system are difficult and slow to operate. So development of automatic fingerprint recognition system is proposed on this paper. In this paper we applied digital image processing system for developing of fingerprint structural detection. The structure of ridges in fingerprint are grasping by video camera storing in database. To recognize an unknown fingerprint it is compared with stored fingerprint in the database to verify their assimilation. The verification is performed by type recognition and focal points matching with each fingerprint.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หัวข้อ	หน้าที่
บทที่ 1 บทนำ	1
บทที่ 2 ความรู้เบื้องต้นและหลักการ	2
2.1 ลักษณะต่างๆบนลายนิ้วมือ	2
2.2 ชนิดและรูปแบบลายนิ้วมือ	4
2.3 ขั้นตอนการทำงาน	8
บทที่ 3 ทฤษฎีพื้นฐานที่เกี่ยวข้อง	10
3.1 ทฤษฎี k – mean	10
3.2 การหาทิศทาง	15
3.3 ฮิสโตแกรม อีควอไลเซชัน	16
บทที่ 4 การประมวลผลภาพ	20
4.1 Histogram Equalization	20
4.2 การหาภาพทิศทาง	20
4.3 การหาจุดศูนย์กลางของภาพ	25
4.3.1 การหาจุดที่คาดว่าจะเป็จุด core	26
4.3.2 การกำจัดจุดที่ไม่น่าจะใช้ทิ้ง	26
4.3.3 การตรวจสอบส่วนโค้ง	28
4.3.4 การรวมกลุ่ม	32
4.3.5 การหาแกนของพื้นที่	33
4.3.6 การหาบล็อกทิศทางที่แทนจุด core	35
4.4 เพิ่มเติมในขั้นตอนตรวจสอบส่วนโค้ง	39
4.5 การทำฐานข้อมูล	43
4.6 การเปรียบเทียบ	44
บทที่ 5 สรุปและวิจารณ์ผลการทดลอง	46
5.1 สรุปผลการทดลอง	46
5.2 สรุปและวิจารณ์ผลการทดลอง	57
5.2.1 ปัญหาที่เกิดขึ้น	57
5.2.2 ข้อเสนอแนะ	64

ภาคผนวก

เอกสารนี้เป็นลิขสิทธิ์ของกรมการศึกษานานาชาติ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีบรรณานุกรม ทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

	หน้าที่
รูปที่ 1 แสดงบริเวณที่มีลักษณะรูปแบบเป็นจุด Core	2
รูปที่ 2 แสดงบริเวณที่มีลักษณะเป็นจุด Core	3
รูปที่ 3 แสดงจุดแยกของลายนูน	3
รูปที่ 4 แสดงจุดปลายของลายนูน	4
รูปที่ 5 แสดงกลุ่มโค้งราบ (Arch)	4
รูปที่ 6 แสดงลักษณะของรูปแบบเส้นโค้งกระโจม (Tented Arch)	5
รูปที่ 7 แสดงลักษณะของกลุ่มมัดหวายเอียงขวา (Right Loop)	5
รูปที่ 8 แสดงลักษณะของกลุ่มมัดหวายเอียงซ้าย (Left Loop)	6
รูปที่ 9 แสดงลักษณะของกลุ่มมัดหวายคู่	7
รูปที่ 10 แสดงลักษณะของรูปแบบของกลุ่มกันหอย	7
รูปที่ 11 แสดงการหาจุดศูนย์กลางและการจัดกลุ่ม	11
รูปที่ 12 แสดงรูปแบบที่ถูกใช้ในตัวอย่าง	12
รูปที่ 13 แสดงรูปทิศทางแบบ 4 ทิศทาง	16
รูปที่ 14 แสดงการส่งผ่านฟังก์ชันของ หนึ่ง gray – level	17
รูปที่ 15 แสดงภาพก่อนข้างมืด	18
รูปที่ 16 แสดงภาพก่อนข้างสว่าง	18
รูปที่ 17 แสดงทิศทางที่ใช้ในการคำนวณ	22
รูปที่ 18 แสดงการตัดบล็อกทิศทางและจำนวนพิกเซลที่ใช้	22
รูปที่ 19 แสดงลักษณะพิกเซลที่ใช้ในการคำนวณ d (1-8)	23
รูปที่ 20 แสดงภาพลายนิ้วมือก่อนนำไปทำภาพทิศทาง	24
รูปที่ 21 แสดงทิศทางที่แปลงจากภาพลายนิ้วมือ	24
รูปที่ 22 แสดงตัวอย่างของบล็อกทิศทางที่เลือกเป็นจุด Core	26
รูปที่ 23 แสดงทิศทางที่อยู่ข้างเคียง	27
รูปที่ 24 แสดงภาพจุดที่คาดว่าน่าจะเป็นจุด Core	28
รูปที่ 25 แสดงการกำจัดจุดที่ไม่น่าจะใช่ทั้ง	28
รูปที่ 26 แสดงการแบ่งย่านของบล็อกที่ล้อมรอบจุด Core	29
รูปที่ 27 แสดงจุดที่คาดว่าน่าจะเป็นจุด Core ก่อนการตรวจสอบส่วนโค้ง	31
รูปที่ 28 แสดงจุดที่คาดว่าน่าจะเป็นจุด Core หลังการตรวจสอบส่วนโค้ง	31

รูปที่ 29 แสดงส่วนที่อยู่ข้างเคียงที่ใช้กำหนดแกนของพื้นที่	33
รูปที่ 30 แสดงรูปแบบพิกเซลที่นำมาใช้ในการคำนวณในการหมุม	35
รูปที่ 31 แสดงบล็อกทิศทางข้างเคียงของบล็อกทิศทางที่แทนจุดเฉพาะ	36
รูปที่ 32 แสดงการหมุมในทิศทางต่างๆ	39
รูปที่ 33 แสดงการแบ่งย่านทั้งสี่เมื่อหมุม ไปตามทิศทางที่ 2	40
รูปที่ 34 แสดงการแบ่งย่านทั้งสี่เมื่อหมุม ไปตามทิศทางที่ 3	40
รูปที่ 35 แสดงการแบ่งย่านทั้งสี่เมื่อหมุม ไปตามทิศทางที่ 4	41
รูปที่ 36 แสดงการแบ่งย่านทั้งสี่เมื่อหมุม ไปตามทิศทางที่ 5	41
รูปที่ 37 แสดงการแบ่งย่านทั้งสี่เมื่อหมุม ไปตามทิศทางที่ 6	42
รูปที่ 38 แสดงการแบ่งย่านทั้งสี่เมื่อหมุม ไปตามทิศทางที่ 7	42
รูปที่ 39 แสดงการแบ่งย่านทั้งสี่เมื่อหมุม ไปตามทิศทางที่ 8	43
รูปที่ 40 แสดงหน้าตาของ โปรแกรมการเปรียบเทียบและตรวจสอบลายนิ้วมือ	46
รูปที่ 41 แสดงภาพที่นำมาเปรียบเทียบ	47
รูปที่ 42 แสดงการทำ Histogram Equalization	48
รูปที่ 43 แสดงภาพทิศทาง	49
รูปที่ 44 แสดงการหาจุดที่คาดว่าจะจะเป็นจุด Core	50
รูปที่ 45 แสดงการตรวจสอบส่วนโค้ง	51
รูปที่ 46 แสดงการหาแกนของภาพ	52
รูปที่ 47 แสดงการหาจุดศูนย์กลางของภาพ	53
รูปที่ 48 แสดงการทำฐานข้อมูล	54
รูปที่ 49 แสดงการเปรียบเทียบและตรวจสอบลายนิ้วมือ	55
รูปที่ 50 แสดงการตรวจพบในฐานข้อมูล	56
รูปที่ 51 แสดงการทำ Threshold ปกติ	58
รูปที่ 52 แสดง Threshold แบบ 5x5	59
รูปที่ 53 แสดงการตกแต่งรูป	60
รูปที่ 54 แสดงการทำให้บาง	61
รูปที่ 55 แสดงการกำจัดเส้นกึ่ง	62
รูปที่ 56 แสดงการหาจุดเด่นของภาพ	63

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

ในปัจจุบันจำนวนประชากรของประเทศ มีจำนวนเพิ่มขึ้นทุกปี ส่งผลให้เกิดปัญหาต่างๆ มากมาย ซึ่งปัญหาต่างๆ ที่เกิดขึ้นมีผลกระทบต่อความมั่นคงของชาติ ปัญหาอย่างหนึ่ง คือ ปัญหาทางด้านอาชญากรรม ปัญหาดังกล่าวจะเกิดขึ้นตามจำนวนประชากร แต่ความสามารถของเจ้าหน้าที่ตำรวจยังมีเท่าเดิม ดังนั้น ปัญหาต่างๆ ที่เกิดขึ้น จึงไม่สามารถคลี่คลายได้ทัน วิธีหนึ่งที่ใช้ในการคลี่คลายปัญหาของเจ้าหน้าที่ตำรวจ จะใช้การตรวจสอบลายนิ้วมือของผู้ต้องสงสัย นำมาเปรียบเทียบกับรอยนิ้วมือที่เจ้าหน้าที่เก็บรวบรวมอยู่ และยังมีขีดความสามารถจำกัดในการใช้บุคคลทำทั้งด้านเวลาและความถูกต้องดังนั้นจึงนำคอมพิวเตอร์มาประยุกต์ใช้ โดยการการสร้างโปรแกรมจดจำลายนิ้วมือ เพื่อที่จะเก็บข้อมูลลายนิ้วมือในลักษณะที่โดดเด่นในแต่ละลาย สำหรับโปรแกรมในการจดจำลายนิ้วมือนี้นี้ยังสามารถนำไปใช้ประโยชน์ในงานด้านต่างๆ ที่เกี่ยวกับการตรวจสอบลายนิ้วมือได้ อาทิ เช่นงานทะเบียนประชาชน งานทางด้านธนาคาร เป็นต้น สิ่งสำคัญของโปรแกรมจดจำลายนิ้วมือก็คือ การพิสูจน์ว่าลายนิ้วมือนั้นเป็นของบุคคลใด ทำให้คอมพิวเตอร์มีประสิทธิภาพในการตรวจสอบสูงขึ้น ซึ่งเป็นข้อดีของการทำวิจัยชิ้นนี้ และงานวิจัยนี้ใช้โปรแกรมเดลไฟมาทำการเขียนโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

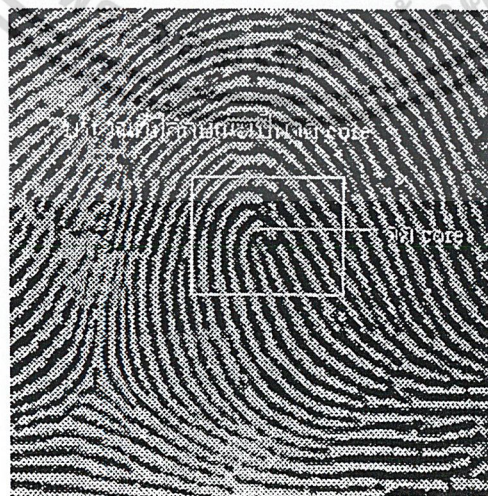
บทที่ 2

ความรู้เบื้องต้นและหลักการ

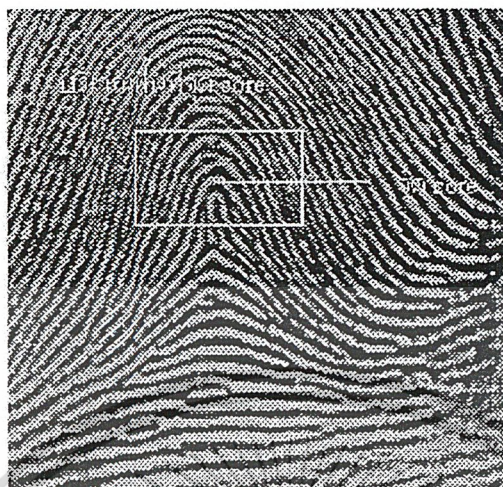
เมื่อเราหงายมือขึ้นและสังเกตไปที่นิ้วมือแต่ละนิ้ว จะพบว่านิ้วมือแต่ละนิ้วนั้นมีลักษณะเป็นลาย ๆ อยู่ เมื่อมองเข้าไปใกล้ ๆ จะเห็นว่าลายนิ้วมือจะประกอบไปด้วยสองส่วนใหญ่ คือ ส่วนที่เป็นลายนูน (Ridges) และร่อง (Valleys หรือ Furrows) วังวนเป็นลักษณะส่วนโค้งไปรอบ ๆ ช่วงปลายนิ้วมือสลับกันไประหว่างร่อง (Valleys หรือ Furrows) และ ลายนูน (Ridges) ทำให้เกิดเป็นลายขึ้นมา ลายดังกล่าวนี้จะมีส่วนของจุดที่เป็นจุดเฉพาะ (Singular point) คือ จุด core และ จุด delta รวมทั้งลักษณะเฉพาะ (Minutiae) (จุดปลายของลายนูน (Ridge Ending) และจุดแยกของลายนูน (Ridge Bifurcation) ประกอบอยู่บนลาย รูปแบบของลายนิ้วมือนี้จะมีอยู่หลายรูปแบบด้วยกันดังจะได้กล่าวในรายละเอียดต่อไป

2.1 ลักษณะของจุดต่าง ๆ บนลายนิ้วมือ

1. จุด core หมายถึง จุดบนเส้นโค้งของลายนูน (Ridges) ซึ่งลักษณะของเส้นโค้งลายนูน (Ridges) ต้องมีลักษณะเป็น โค้งขึ้นและเริ่มที่จะกลับตัวหรือเริ่มที่จะกลายเป็น โค้งลง แล้วจึงวิ่งสวนทางกลับกันกับในตอนแรกก่อนที่จะโค้งขึ้น เส้นโค้งลายนูน (Ridges) นี้ต้องเป็นเส้น โค้งลายนูน (Ridges) ที่อยู่ชั้นในที่สุดของบริเวณที่มีลักษณะรูปแบบเป็น core ลักษณะของบริเวณที่มีลักษณะรูปแบบเป็น core นั้นเป็นลักษณะของเส้น โค้งที่โค้งขึ้นมาซ้อน ๆ กันหลายเส้นในลักษณะของรูปมัดหวาย จาก โค้งวงกว้างลดลงมาเรื่อย ๆ จน โค้งเล็กสุด(เส้นที่อยู่ชั้นในที่สุดอาจไม่เป็นเส้น โค้งก็ได้) ลักษณะของรูปแบบเส้น โค้งที่อยู่ในบริเวณที่มีลักษณะรูปแบบเป็น core ยังมีอีกหลายลักษณะ ซึ่งจะกล่าวในหัวข้อต่อไป



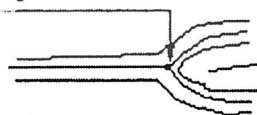
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกที่ 1 แสดงบริเวณที่มีลักษณะรูปแบบเป็นจุด Core เอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2 แสดงบริเวณที่มีลักษณะเป็นจุด Core

2. ลักษณะเฉพาะ (Minutiae) เป็นลักษณะเฉพาะในรูปลายนิ้วมือ ประกอบด้วยส่วนประกอบสำคัญ 2 ชนิด คือ จุดแยกของลายนิ้ว (Ridge Bifurcation) เป็นจุดที่อยู่บนลายนิ้ว (Ridges) ที่เกิดจากการแยกจากหนึ่งเส้นทางเป็นสองเส้นทาง ดังแสดงในรูปข้างล่าง และ จุดปลายของลายนิ้ว (Ridge Ending) เป็นจุดที่อยู่บนลายนิ้ว (Ridges) ในบริเวณปลายสุดของลายนิ้ว (Ridges) แสดงไว้ในรูปด้านล่าง

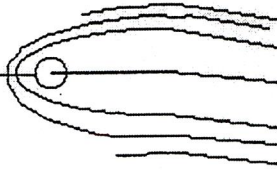
จุดแยกของลายนิ้ว (Ridge Bifurcations)



รูปที่ 3 แสดงจุดแยกของลายนิ้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จุดปลายของลายนิ้วมือ
(Ridge Endings)



รูปที่ 4 แสดงจุดปลายของลายนิ้วมือ

2.2 ชนิดและรูปแบบลายนิ้วมือ

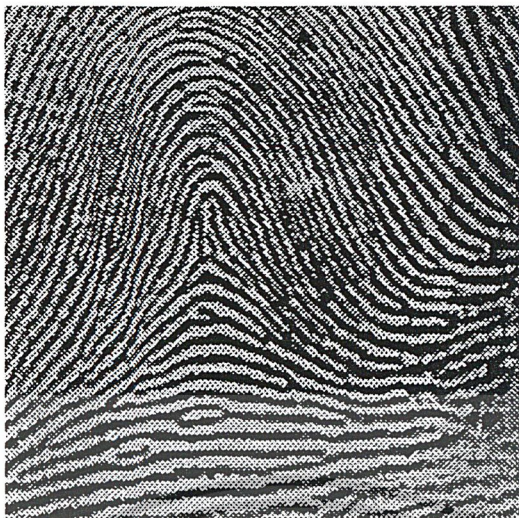
รูปแบบลายนิ้วมือ (Ridges) ในภาพลายนิ้วมือ ที่จับกลุ่มกันเป็นรูปแบบ สามารถแบ่งได้หลายชนิด ดังต่อไปนี้

1. กลุ่มเส้นโค้ง ลักษณะของรูปแบบในภาพลายนิ้วมือมีลักษณะการวิ่งของลายนิ้วมือ (Ridges) จากลักษณะที่ขนานกับพื้นราบแล้วพุ่ง โค้งขึ้นแล้วจึงมีการวิ่งในลักษณะขนานกับพื้นราบอีกครั้ง รูปแบบของกลุ่มเส้นโค้งนั้นมี 2 รูปแบบ คือ กลุ่มเส้นโค้งราบ (Arch) และ กลุ่มเส้นโค้งกระโจม (Tented Arch) โดยกลุ่มเส้นโค้งทั้งสองจะแตกต่างกันที่ลักษณะความสูงของส่วนโค้ง ดังตัวอย่างรูปด้านล่าง



รูปที่ 5 แสดงกลุ่มโค้งราบ (Arch)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

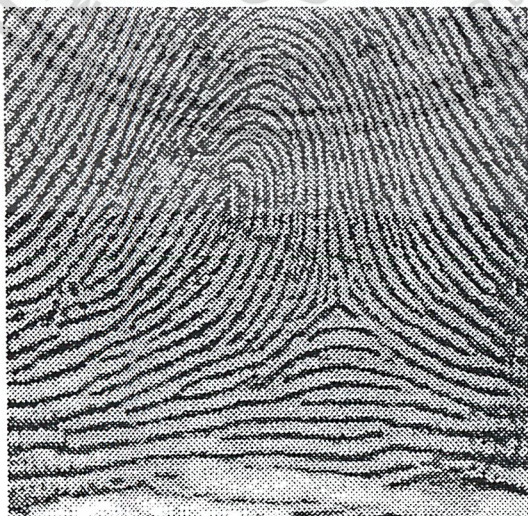


รูปที่ 6 แสดงลักษณะของรูปแบบเส้นโค้งกระโจม (Tented Arch)

2. กลุ่มมัดหวาย (Loop) ที่พบในลายนิ้วมือจะมีอยู่ 2 ประเภทด้วยกัน คือ

2.1 กลุ่มมัดหวายเดี่ยว (Single Loop) ในภาพลายนิ้วมือสามารถแบ่งได้ 2 ชนิด คือ

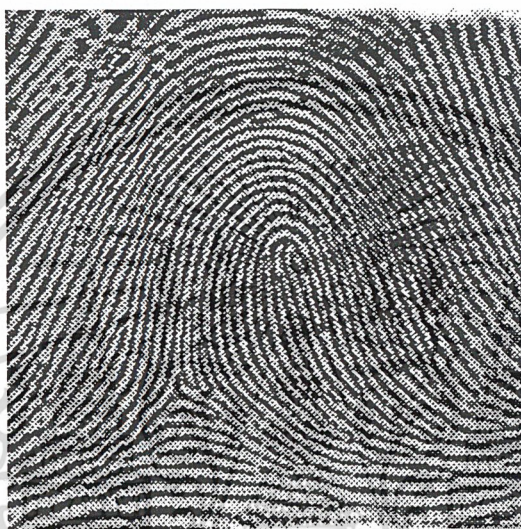
2.1.1 กลุ่มมัดหวายเอียงขวา (Right Loop) มีลักษณะของการโค้งของลายนิ้ว (Ridges) จากบริเวณด้านซ้ายมาที่บริเวณกลางนิ้วมือแล้วโค้งโค้งขึ้นไปด้านบนแล้วจึงโค้งลงและโค้งวนกลับไปตามทางเดิมที่ได้วิ่งมา เมื่อสังเกตด้วยตาเปล่าจะเห็นว่าเมื่อมีการจับกลุ่มเป็นรูปแบบคล้ายกับมัดหวายแล้วจะเอนเอียงไปทางด้านขวาจึงได้เรียกว่า “กลุ่มมัดหวายเอียงขวา”



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น คือห้ามมิให้คัดลอกหรือเผยแพร่ข้อมูลข้างต้นไปยังผู้อื่นของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 7 แสดงลักษณะของกลุ่มมัดหวายเอียงขวา (Right Loop)

2.1.2 กลุ่มมัดหวายเอียงซ้าย (Left Loop) รูปแบบจะคล้ายลักษณะของกลุ่มมัดหวายเอียงขวา คือ มีการวิ่งของลายนูน (Ridges) จากทางด้านขวามาที่บริเวณตรงกลางของนิ้วมือ จากนั้นจะวิ่งขึ้นไปด้านบนแล้วโค้งและวิ่งกลับไปตามทางเดิมที่วิ่งมา เมื่อทำการสังเกตจะพบว่ามีลักษณะรูปแบบคล้ายกับมัดหวาย และรูปแบบดังกล่าวจะมีการเอียงตัวไปทางด้านซ้าย เราจึงเรียกว่า “ กลุ่มมัดหวายเอียงซ้าย ” ปกติมักจะมี delta อยู่ทางด้านซ้ายของรูปแบบเช่นกัน



รูปที่ 8 แสดงลักษณะของกลุ่มมัดหวายเอียงซ้าย (Left Loop)

2.2 กลุ่มมัดหวายคู่ (Double Loop) เป็นกลุ่มของรูปแบบที่เกิดจากการรวมกันของกลุ่มมัดหวายเดี่ยวสองกลุ่มรวมตัวกันเป็นรูปแบบเดียว โดยลักษณะการวิ่งของลายนูน (Ridges) อาจมาจากทางด้านเดียวกันแล้วรวมตัวกันเป็นลักษณะของกลุ่มมัดหวายเดี่ยวสองกลุ่ม โดยที่กลุ่มหนึ่งจะมีรูปแบบเป็นลักษณะของโค้งขึ้นและกลุ่มมัดหวายอีกกลุ่มหนึ่งจะมีลักษณะโค้งลง เมื่อรวมกันเป็นกลุ่มมัดหวายคู่รูปแบบดังกล่าวอาจมีการเอียงซ้ายหรือเอียงขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 9 แสดงลักษณะรูปแบบของกลุ่มมัดหวนคู่ (Double Loop)

3. กลุ่มก้นหอย (Whorl) ลักษณะการวิ่งของลายนิ้วน (Ridges) มีรูปแบบเป็นลักษณะเส้นโค้งเป็นรูปวงกลมจากวงใหญ่ค่อย ๆ เล็กลงไปเรื่อย ๆ จนเหลือวงกลมวงเล็กที่สุด



รูปที่ 10 แสดงลักษณะรูปแบบของกลุ่มก้นหอย (Whorl)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ขั้นตอนการทำงาน

1. การทำ Histogram Equalization

เมื่อได้ภาพลายนิ้วมือจากกล้อง แล้วนำภาพมาทำการ Histogram Equalization เพื่อให้ภาพมีความสม่ำเสมอให้มีความโดดเด่นในค่าสีระดับใดระดับหนึ่ง และเป็นกระบวนการเตรียมภาพเบื้องต้น เพื่อไปทำในขั้นตอนต่อไป

2. การหาทิศทางของภาพ

นำภาพที่ผ่านการทำ Histogram Equalization มาหาทิศทางของภาพ เพื่อให้ทราบว่าลายนิ้วมือนั้นทิศทางไปในทิศทางใด

3. หาจุดที่คาดว่าจะจะเป็นจุด core

นำภาพที่ผ่านการหาทิศทางของภาพ มาหาจุดที่คาดว่าจะจะเป็นจุด core โดยนำค่าของทิศทางของภาพที่บล็อกกำลังพิจารณา ลบด้วย บล็อกก่อนหน้านั้น ต้องมากกว่า 2 หรือ มีค่าความแตกต่างของมุมมากกว่า 45 องศา

4. กำจัดจุดที่ไม่น่าจะใช้ทิ้ง

นำภาพที่คาดว่าจะจะเป็นจุด core นำมาทำการกำจัดจุดที่ไม่น่าจะใช้ทิ้ง โดยอาศัยบล็อกทิศทางข้างเคียงที่ล้อมรอบอยู่มาพิจารณาว่าควรจะเป็นจุด core ในภาพทิศทางนั้นหรือไม่ ผลที่ได้จากกระบวนการนี้จะกำจัดจุดที่ไม่น่าจะใช้จุด core ออกไป

5. การตรวจสอบส่วนโค้ง

นำภาพที่ได้จากการกำจัดจุดที่ไม่น่าจะใช้ทิ้ง มาทำการตรวจสอบส่วนโค้ง โดยนำบล็อกทิศทางที่คาดว่าจะจะเป็นจุด core มาทำการตรวจสอบกับบล็อกข้างเคียง แล้วหาค่าวาเรียนส์ที่มากที่สุด

6. การรวมกลุ่ม

นำภาพที่ผ่านการตรวจสอบส่วนโค้ง มาทำการรวมกลุ่ม โดยใช้ทฤษฎี K- Mean โดยจะกล่าวถึงทฤษฎีนี้ในบทต่อไป

7. การหาแกนของพื้นที่

เป็นการหาลักษณะการเอนเอียงของบริเวณที่เป็นจุด core เพื่อให้มีความถูกต้องของภาพลายนิ้วมือที่เข้ามา เมื่อมีลักษณะการเอนเอียงของภาพ

8. การหาบล็อกทิศทางที่แทนจุด core

เป็นขั้นตอนสุดท้ายของการหาบล็อกทิศทางที่แทนด้วยจุด core โดยการนำบล็อกทิศทางที่แทนด้วยจุด core ที่อยู่ในแต่ละกลุ่มมาพิจารณาว่าบล็อกทิศทางใดมีความน่าจะเป็นสูงที่สุด แล้วให้บล็อกนั้นเป็นบล็อกทิศทางของจุด core โดยจะได้ออกมาเพียงจุดเดียว

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของกรมการปกครอง กระทรวงมหาดไทย
 9. นำบล็อกทิศทางจุด core มาตรวจสอบว่าต้องการทำฐานข้อมูลหรือไม่ หากทำฐานข้อมูลก็ทำการเก็บข้อมูล หากไม่ทำฐานข้อมูลก็ไปทำขั้นตอนการเปรียบเทียบเอกสารทุกครั้งที่มีการนำไปใช้

10. การเปรียบเทียบ

นำบล็อกทิศทางของจุด CORE และข้อมูลภายในบล็อกทิศทางข้างเคียงมาทำการเปรียบเทียบ เพื่อยืนยันว่ามีอยู่ในฐานข้อมูลหรือไม่ หากพบในฐานข้อมูล จะแสดงบุคคลนั้นออกมา หากไม่มีในฐานข้อมูล จะแสดงข้อความว่าไม่พบในฐานข้อมูล



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

ทฤษฎีพื้นฐานที่เกี่ยวข้อง

3.1 ทฤษฎี K-Mean

วิธีนี้เป็นวิธีการรวมกลุ่ม และหาจุดศูนย์กลางของกลุ่ม โดยเราใช้วิธีนี้ในขั้นตอนการหาจุด core ซึ่งมีขั้นตอนรายละเอียดต่าง ๆ ดังนี้

ขั้นตอนที่ 1 เลือกจุดศูนย์กลางเริ่มต้นของกลุ่มจำนวน k จุด คือ $Z_1(1), Z_2(1), \dots, Z_k(1)$ ค่าเหล่านี้เลือกได้ตามความต้องการ เมื่อ $Z_k(n)$ เป็นจุดศูนย์กลางเริ่มต้น จำนวน k จุด จุดที่เลือกมาครั้งที่ n

ขั้นตอนที่ 2 จัดกลุ่มโดยหาสมาชิก $\{X\}$ ที่อยู่รอบ ๆ จุดศูนย์กลางตัวที่ $1, 2, \dots, k$ โดยใช้ความสัมพันธ์ดังต่อไปนี้

$$x \in S_j(k) \quad \text{ถ้า} \quad \|x - z_j(k)\| < \|x - z_i(k)\|$$

จากสมการเป็นการเปรียบเทียบว่าจุด x ควรเป็นสมาชิกของจุดศูนย์กลางใด โดยวัดจากระยะทางว่าจุด x อยู่ใกล้จุดศูนย์กลางใดมากกว่ากัน เมื่อทำการจัดกลุ่มเรียบร้อยแล้ว โดยจุดทุกจุดจะต้องเป็นสมาชิกของจุดศูนย์กลางใดจุดศูนย์กลางหนึ่ง ก็จะไปทำขั้นตอนที่ 3 ต่อไป เมื่อ $i = 1, 2, \dots, k$ ยกเว้น $i = j$ และ $j = 1, 2, \dots, k$ ยกเว้น $i = j$ โดยที่ $S_j(k)$ เป็นกลุ่มของจุดที่เป็นสมาชิกของจุดศูนย์กลางตัวที่ j , $Z_j(k)$ เราอาจตั้งเงื่อนไขเพิ่มเติมจากความสัมพันธ์ที่กำหนดไว้ข้างต้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 11 แสดงขั้นตอนในการหาจุดศูนย์กลางและการจัดกลุ่ม

ขั้นตอนที่ 3 การปรับปรุงจุดศูนย์กลางกลุ่ม จากผลที่ได้ในขั้นตอนที่ 2 จะนำจุดศูนย์กลางทุกจุดของแต่ละกลุ่มมาปรับปรุงคำนวณหาจุดศูนย์กลางใหม่ $Z_j(k+1)$ เมื่อ $j = 1, 2, \dots, k$ จุดศูนย์กลางใหม่ (จุดศูนย์กลางของกลุ่ม กลุ่มเดิมที่ปรับปรุงให้เลื่อนไปอยู่ตรงกลางของกลุ่มหลังการหาสมาชิกมาใหม่ในขั้นตอนที่ 2) กำหนดโดย

$$Z_j(k+1) = 1/N_j \sum_{x \in S_j(k)} X \quad j = 1, 2, \dots, k$$

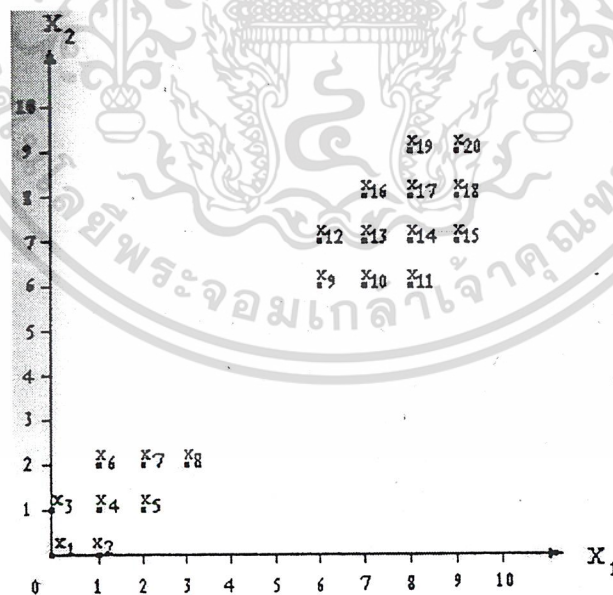
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ $N_j(k)$ คือ จำนวนของสมาชิกทั้งหมดใน $S_j(k)$ ชื่อ ตัวกลาง K (K-Mean) ตั้งขึ้นมาจากการที่จุดศูนย์กลางกลุ่มถูกปรับปรุงเรียงกันไปตามลำดับ

ขั้นตอนที่ 4 เปรียบเทียบจุดศูนย์กลาง ถ้าจุดศูนย์กลางที่หามาใหม่ (ได้มาจากการปรับปรุงจุดศูนย์กลางในขั้นตอนที่ 3) มีค่าเท่ากับจุดศูนย์กลางตัวเก่าก่อนที่จะนำมาทำการปรับปรุงในขั้นตอนที่ 3 ($Z_j(k+1) = Z_j(k)$ เมื่อ $j = 1, 2, \dots, k$) แสดงว่ากฎเกณฑ์ความจริง (algorithm) ได้ย้อนกลับมาดำเนินการจะสิ้นสุดลง แต่ถ้าจุดศูนย์กลางที่ได้รับการปรับปรุงในขั้นตอนที่ 3 มีค่าไม่เท่ากับค่าของจุดศูนย์กลางก่อนที่จะได้รับการปรับปรุงในขั้นตอนที่ 3 แล้วก็ให้ย้อนกลับมาทำในขั้นตอนที่ 2 ถึง 4 อีกครั้ง เมื่อทำมาถึงขั้นตอนที่ 4 ถ้าจุดศูนย์กลางที่ได้ปรับปรุงหรือหาได้ใหม่มีค่าเท่ากับจุดศูนย์กลางเดิมก่อนการปรับปรุงหรือหามาใหม่ก็เป็นอันสิ้นสุดการดำเนินการ แต่ถ้าไม่เท่ากันอีกก็จะวนไปขั้นตอนที่ 2 ถึง 4 อีกครั้งเช่นนี้ไปเรื่อยๆ จนกว่าจะได้ค่าเท่ากันในขั้นตอนที่ 4

ตัวอย่าง

เพื่อที่จะแสดงเกี่ยวกับกฎเกณฑ์ความจริงตัวกลาง K ขอให้พิจารณารูปแบบดังในรูปที่ 16 โดยการทำตามขั้นตอนวิธีการ ที่ได้กล่าวมาแล้วเราจะได้



รูปที่ 12 แสดงรูปแบบที่ถูกใช้ในตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 1 กำหนดจุดศูนย์กลางเริ่มต้น โดยกำหนดให้จำนวนจุดศูนย์กลางเริ่มต้นเท่ากับ 2 จุด ($k=2$) และเลือก $Z_1(1) = X_1 = (0,0)$, $Z_2(1) = X_2 = (1,0)$ เป็นจุดศูนย์กลางเริ่มต้น

ขั้นตอนที่ 2 ทำการจัดกลุ่ม โดยการหาสมาชิกจากการวัดระยะทางจากจุดใดมาที่จุดศูนย์กลางใด ๆ แล้วเปรียบเทียบระยะทาง ถ้าระยะทางจากจุดนั้นมีค่าน้อยที่สุดเมื่อเทียบกับจุดศูนย์กลางใด (แสดงว่าจุด ๆ นั้นใกล้กับจุดศูนย์กลางนั้น) ก็กำหนดให้จุดนั้นเป็นสมาชิกของจุดศูนย์กลางที่ใกล้ที่สุดตามสมการ

$$\begin{aligned} \text{เพราะว่า } & \|x_1 - z_1(1)\| < \|x_1 - z_i(1)\| \\ \text{และ } & \|x_3 - z_1(1)\| < \|x_3 - z_i(1)\| \end{aligned}$$

เมื่อ $i=2$ จะได้ $s_1(1) = \{x_1, x_3\} = \{(0,0), (0,1)\}$ ในทำนองเดียวกัน จุดอื่นๆ ที่ยังเหลืออยู่ที่อยู่ใกล้จุดศูนย์กลาง $Z_2(1)$ จะเป็นสมาชิกของจุดศูนย์กลางจุดนี้ ดังนั้น จะได้ $s_2(1) = \{x_2, x_4, x_5, \dots, x_{20}\} = \{(1,0), (1,1), (1,2), (2,1), (2,2), (3,2), (6,6), (6,7), (7,6), (7,7), (7,8), (8,6), (8,7), (8,8), (8,9), (9,7), (9,8), (9,9)\}$

ขั้นตอนที่ 3 ปรับปรุงจุดศูนย์กลางกลุ่ม โดยการหาจุดศูนย์กลางใหม่ของกลุ่มเดิม

$$\begin{aligned} z_1(2) &= (1/N_1) \sum_{x \in s_1(1)} x \\ &= (1/2) (x_1 + x_2) \\ &= (1/2) ((0,0), (0,1)) \\ &= (0,0.5) \end{aligned}$$

$$z_2(2) = (1/N_2) \sum_{x \in s_2(1)} x$$

$$\begin{aligned} &= (1/18) (x_2 + x_4 + x_5 + \dots + x_{20}) \\ &= (1/18) ((1,0) + (1,1) + (1,2) + (2,1) + \\ & (2,2) + (3,2) + (6,6) + (6,7) + (7,6) + (7,7) + (\\ & 7,8) + (8,6) + (8,7) + (8,8) + (8,9) + (9,7) + \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาวิจัยและการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้ง (9,8) + (9,9) นี้ขอหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$= (5.67, 5.33)$$

ขั้นตอนที่ 4 เปรียบเทียบจุดศูนย์กลาง เพราะว่า $z_j(2) \neq z_j(1)$, $j = 1, 2$ จึงกลับไปทำขั้นตอนที่ 2 ใหม่ (เนื่องจากจุดศูนย์กลางจุดใหม่ หลังการปรับปรุงจุดศูนย์กลาง ในขั้นตอนที่ 3 จุด $(0,0.5)$ ไม่เท่ากับ จุดศูนย์กลางเก่า ก่อนการปรับปรุงในขั้นตอนที่ 3 จุด $(0,0)$ และจุดศูนย์กลางใหม่อีกตัว จุด $(5.67,5.33)$ ไม่เท่ากับจุดศูนย์กลางเก่า ก่อนการปรับปรุงจุดศูนย์กลาง จุด $(1,0)$)

ขั้นตอนที่ 2 นำจุดศูนย์กลางกลุ่มจุดใหม่ ที่ได้มาหาสมาชิกใหม่อีกครั้ง โดยใช้ความสัมพันธ์ลักษณะเดิมคือ $\|x_L - z_1(2)\| < \|x_L - z_2(2)\|$ เมื่อ $L = 1, 2, \dots, 8$ และ $\|x_L - z_2(2)\| < \|x_L - z_1(2)\|$ เมื่อ $L = 9, 10, \dots, 20$ ดังนั้น กลุ่มของสมาชิกจะเปลี่ยนไป สมาชิกซึ่งเป็นจุดใหม่เพิ่มเข้ามาในกลุ่ม 1 แต่กลุ่ม 2 จะมีจำนวนลดลงไป จะได้

$$S_1(2) = \{x_1, x_2, \dots, x_8\} = \{(0,0), (0,1), (1,1), (1,2), (2,1), (2,2), (3,2)\} \text{ และ}$$

$$S_2(2) = \{x_9, x_{10}, \dots, x_{20}\} = \{(6,6), (6,7), (7,6), (7,7), (7,8), (8,6), (8,7), (8,8), (8,9), (9,7), (9,8), (9,9)\}$$

ขั้นตอนที่ 3 ปรับปรุงจุดศูนย์กลางของกลุ่ม หาจุดศูนย์กลางกลุ่มใหม่อีกครั้ง (เป็นจุดศูนย์กลางของกลุ่มเดิมแต่เลื่อนจุด ไปให้อยู่ตรงกลางของกลุ่ม) โดยนำสมาชิกใหม่ที่ได้มาใช้ในการปรับปรุงจุดศูนย์กลางใหม่

$$Z_1(3) = (1/N_1) \sum_{x \in S_1(2)} x$$

$$= (1/8) (x_1 + x_2 + \dots + x_8)$$

$$= (1/8) ((0,0), (0,1), (1,0), (1,1), (1,2), (2,1), (2,2), (3,2))$$

$$Z_2(3) = (1/N_2) \sum_{x \in S_2(2)} x$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{aligned}
&= (1/12) (x_9 + x_{10} + \dots + x_{20}) \\
&= (1/12) ((6,6),(6,7),(7,6),(7,7),(7,8),(8,6),(8,7), \\
&\quad (8,8),(8,9),(9,7),(9,8),(9,9)) \\
&= (7.67,7.33)
\end{aligned}$$

ขั้นตอนที่ 4 เพราะว่า $z_j(3) \neq z_j(2)$, $j = 1, 2$ เนื่องจากจุด (0,0.5) ไม่เท่ากับจุด (1.25,1.13) และจุด (5.67,5.33) ไม่เท่ากับจุด (7.67,7.33) จึงต้องกลับไปทำขั้นตอนที่ 2 อีกครั้ง

ขั้นตอนที่ 2 กระทำในลักษณะเดียวกับครั้งที่แล้ว จะได้กลุ่มสมาชิกเหมือนเดิมคือ

$$\begin{aligned}
S_1(4) &= S_1(3) = \{x_1, x_2, \dots, x_8\} \quad \text{และ} \\
S_2(4) &= S_2(3) = \{x_9, x_{10}, \dots, x_{20}\}
\end{aligned}$$

ขั้นตอนที่ 3 เมื่อกระทำแล้วจะได้เป็นจุดศูนย์กลางตัวเดิม เพราะสมาชิกเป็นตัวเดิมทั้งหมด

ขั้นตอนที่ 4 เปรียบเทียบจุดศูนย์กลาง เพราะว่า $z_j(4) = z_j(3)$ เมื่อ $j = 1, 2$ เนื่องจากจุดศูนย์กลางใหม่หลังการปรับปรุง (5.67,5.33) เท่ากับจุดศูนย์กลางเดิม (5.67,5.33) และจุดศูนย์กลางใหม่หลังการปรับปรุง (7.67,7.33) เท่ากับจุดศูนย์กลางเดิม (7.67,7.33) กฎเกณฑ์ความจริงจะเกิดการย้อนกลับ ดังนั้นจะได้จุดศูนย์กลางของกลุ่มเป็นดังนี้

$$\begin{aligned}
Z_1 &= (1.25,1.13) \quad \text{และ} \\
Z_2 &= (7.67,7.33)
\end{aligned}$$

3.2 การหาทิศทาง

ภาพทิศทาง (direction image) เป็นภาพที่ได้จากการแปลงลายนิ้วมือเป็นทิศทางต่างๆ จนเป็นภาพทิศทางก่อนที่จะเข้าสู่วิธีการหาจุด core และจุด delta ทั้ง 6 ขั้นตอน ภาพลายนิ้วมือที่ได้มาจะประกอบไปด้วย ลายนิ้ว (ridge) และร่อง (valley) โดยที่ทิศทางในแต่ละ pixel ของภาพแทนทิศทางการเดินของลายนิ้ว และร่อง ที่ pixel นั้น เรานำลายนิ้วมือมาทำ

การตัดเป็นบล็อกขนาด $g \times g$ pixel แล้วหาผลรวมของผลต่างของค่า gray value ตามทิศทาง d ค่าของผลรวมที่ได้น้อยที่สุดจะถูกใช้ในการกำหนดทิศทางซึ่งเป็นไปตามสูตรที่มีการนำไปใช้

$$S_d = \sum_{m=0}^{M-1} |G(i_m, j_m) - G(i, j)| \quad \text{สำหรับ } d = 1 \dots N$$

เมื่อ S_d เป็นผลรวมของผลต่าง ของค่าระดับเทา ในแต่ละพิกเซล ตามทิศทาง d และ M คือ จำนวนจุดตามทิศทาง d ที่ถูกเลือกใช้ในการคำนวณ ส่วน N เป็นทิศทางจำนวนทิศทางทั้งหมดที่ใช้ $G(i, j)$ เป็นค่าระดับเทาที่พิกัด i และ j g เป็นขนาดของบล็อกที่เลือกมาทำการคำนวณ โดยค่าของ d , g , M และ N เป็นขนาดของค่าคงที่ ที่ถูกเลือกใช้ในการคำนวณตามความเหมาะสม



รูปที่ 13 แสดงรูปทิศทางแบบ 4 ทิศทาง

3.3 ฮิสโตแกรม อีควอไลเซชัน

เนื่องจากภาพบางภาพที่ใช้ในการคำนวณ มีลักษณะมืดหรือเลอะเลือน ซึ่งเป็นผลมาจากวิธีการเก็บภาพหรือเหตุผลอื่นๆ ทำให้จุด **core** ที่เราหามาได้ไม่ตรงกับจุด **core** ที่แท้จริงในภาพลายนิ้วมือ วิธีการจัดการกับภาพคุณภาพต่ำเหล่านี้ โดยการใช้ วิธีของ **histogram equalization**

หลักการ

กำหนดให้ r เป็นระดับความเข้มตั้งแต่ ขาว ถึง ดำ โดยที่

เอกสารนี้เป็นเอกสารที่ $0 \leq r \leq 1$ ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$r = 0$ แสดงค่าสีดำสนิท

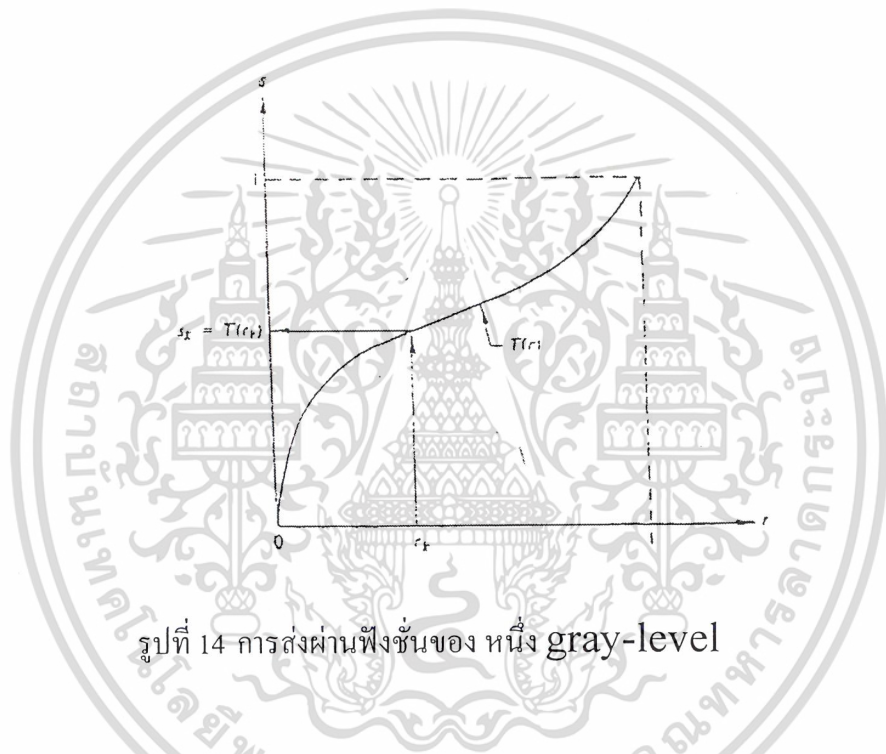
$r = 1$ แสดงค่าสีขาว

และ สำหรับ ค่า r ใดๆ เรากำหนดสมการ

$$S = T(r)$$

โดยที่ T เป็น Transformation function

S คือ ค่าความเข้มที่ปรับปรุงแล้ว โดยผ่าน Transformation function และ Inverse transformation จาก s ไป r คือ $r = T^{-1}(s)$



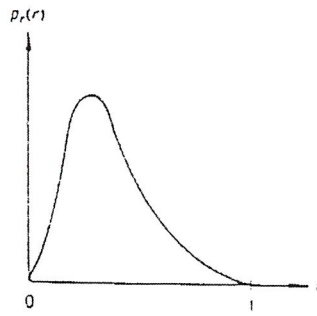
รูปที่ 14 การส่งผ่านฟังก์ชันของ หนึ่ง gray-level

ค่าของภาพดั้งเดิม และ ภาพที่ทำการแปลงแล้ว สามารถบอกคุณสมบัติได้โดย $P_r(r)$ และ $P_s(s)$ ซึ่งเป็นตัวบอกถึง ความหนาแน่นของระดับสีเทา เช่น ความสัมพันธ์ของ $P_r(r)$, $P_s(s)$ และ T_r คือ

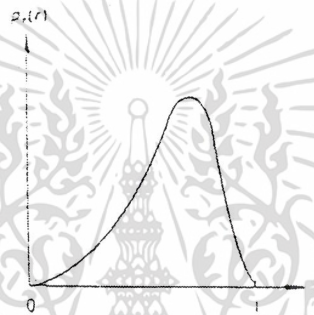
$$P_s(s) = P_r(r) \frac{dr}{ds} \quad \{ r = T^{-1}(s) \}$$

ดังนั้นวิธีการปรับปรุงภาพโดย Histogram นั้น อาศัยพื้นฐานของ การควบคุมความหนาแน่น ของ probability ของระดับสีเทา ก็โดยการใช้ transformation $T(r)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 15 ภาพก่อนข้างมืด



รูปที่ 16 ภาพก่อนข้างสว่าง

กำหนด function

$$S = T(r) = \int_0^r P_r(w) dw \quad \text{เมื่อ } 0 \leq r \leq 1$$

$\int_0^r P_r(w) dw$ เรียกว่า Cumulative distribution function (CDF)

ในการประมวลผลแบบไม่ต่อเนื่อง ค่า probability หาได้โดย

$$P_r(r_k) = n_k / n$$

$$\text{โดยที่ } 0 \leq r_k \leq 1$$

$$k = 0, 1, \dots, L-1$$

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของโรงเรียนโพธิสารวิทยาคาร ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น $P_r(r_k)$ เป็น probability ของระดับ k^{th} gray level

n_k เป็น จำนวนครั้งที่ระดับนั้นปรากฏในภาพ image

ค่า transformation function สามารถเขียนได้เป็น

$$\begin{aligned} S_k &= T(r_k) = n_k / n \\ &= P_r(r_k) \end{aligned}$$

โดยที่ $0 \leq r_k \leq 1$

$k = 0, 1, \dots, L-1$

และ inverse transformation คือ

$$r_k = T^{-1}(S_k) ; 0 \leq S_k \leq 1$$



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การประมวลผลภาพ

4.1 Histogram Equalization

เมื่อได้ภาพลายนิ้วมือจากกล้องมาแล้ว เรานำภาพที่ได้ไปทำการ histogram equalization เพราะภาพที่ได้มาจะมีค่าความถี่ ของ ระดับเทา มีค่าอยู่ในช่วงแคบ ๆ ต้องทำการกระจายความถี่ ให้มีค่าอยู่ในช่วง 0-255 การทำงานมีดังนี้

$$P_1(a) = (1/Area_1) H_1(a) \quad [1]$$

สมการที่ 1 แสดงความน่าจะเป็นที่จะเจอพิกเซล ที่ความเข้ม a ในภาพแทนด้วย $P_1(a)$ และ $Area_1$ เป็นจำนวนพิกเซลทั้งหมดบนภาพ และสุดท้าย $H_1(a)$ แสดงค่า histogram ของภาพ

$$P_1(a) = (1/Area_1) \sum_{i=0}^a H_1(a) \quad [2]$$

สมการที่ 2 แสดงความน่าจะเป็นสะสมของพิกเซล ที่ความเข้ม a ความน่าจะเป็นสะสมก็คือ ผลรวมของความน่าจะเป็นที่จะมีความเข้มสี ตั้งแต่น้อยกว่าจนถึงเท่ากับ a

$$f(a) = D_m (1/Area_1) \sum_{i=0}^a H_1(a) \quad [3]$$

สมการที่ 3 แสดงฟังก์ชัน Histogram Equalization $f(a)$ เมื่อ $H_1(i)$ แสดง histogram ของภาพเริ่มต้นที่ 1 และ D_m เป็นจำนวนสีที่ต้องการให้มีอยู่ในภาพที่ทำให้คมชัดแล้ว

4.2 การแปลงภาพลายนิ้วมือเป็นภาพทิศทาง

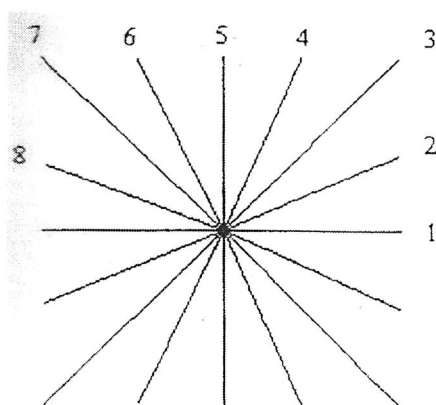
ในการแปลงภาพลายนิ้วมือที่ได้มาเป็นภาพทิศทาง (Directional Image) เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต
ภาพลายนิ้วมือที่ได้มาจะประกอบไปด้วย ลายนูน (ridge) และร่อง (valley) โดยที่

ทิศทางในแต่ละพิกเซลของภาพ จะแทนทิศทางของลายนูน และ ร่อง รวมกันเป็นภาพลายนิ้วมือออกมา ภาพลายนิ้วมือที่นำมาแปลงเป็นภาพทิศทางนั้น เราให้ทิศทาง ในภาพทิศทางแทนทิศทางการเดินของลายนูน โดยการตัดบล็อกขนาด $8*8$ pixel (ในที่นี้ภาพลายนิ้วมือที่ใช้จะเป็นภาพ gray image มีขนาด $256*256$) ลงบนภาพลายนิ้วมือจนทั่วทั้งภาพ จะได้ภาพบล็อกทิศทางขนาด $32*32$ บล็อก แล้วในแต่ละบล็อกนั้นเราจะหาทิศทาง การใช้บล็อกขนาด $8*8$ pixel เนื่องจากว่าเหมาะสม สามารถล้อมรอบลายนูน แต่ละลายได้ การตัดบล็อกขนาดใหญ่กว่านี้ก็สามารถทำได้ แต่ผลที่ได้นั้น คือ หนึ่งในบล็อกนั้นจะมีจำนวนลายนูน 2 ลาย หรือมากกว่า ซึ่งทำให้ทิศทางที่ได้ไม่ใช่ทิศทางที่แท้จริงของทิศทางลายนูน ส่วนถ้าบล็อกมีขนาดเล็กเกินไปแล้ว จะทำให้บล็อกบางบล็อกไม่มีลายนูนปรากฏอยู่ หรือ บล็อกไม่สามารถล้อมรอบลายนูนได้ ผลของทิศทางที่ได้จากการตัดบล็อกลักษณะนี้ ก็ไม่ได้ทิศทางที่แทนทิศทาง การเดินทางของลายนูนอีกเช่นกัน ดังนั้นเราจึงได้ใช้บล็อกขนาด $8*8$ ต่อมาในแต่ละบล็อกเราจะหาทิศทางแทนลายนูน ได้โดยการหาผลรวมของผลต่างของค่า gray level ตามทิศทาง d ที่มีค่าของผลรวมที่ได้น้อยที่สุด กำหนดทิศทางดังกล่าวแทนทิศทางการเดินทางของลายนูน ซึ่งเป็นไปตามสูตร

$$S_d = \sum_{m=0}^{M-1} |G(i_m, j_m) - G(I, j)| \quad \text{สำหรับ } d = 1 \dots N$$

M คือจำนวนจุดที่ถูกเลือกมาใช้ในการคำนวณ ในที่นี้กำหนดให้มีค่าเท่ากับ 8 เนื่องจากเราใช้บล็อกขนาด $8*8$ pixel และ d คือทิศทางที่เราใช้ในการคำนวณ ส่วน N เป็นจำนวนทิศทางทั้งหมดที่ใช้ กำหนดให้มีค่าเท่ากับ 8 (ทิศทาง d พิจารณาจากมุม $\Pi(d-1)/N$ เรเดียน) เพราะว่าถ้าเรากำหนดให้มีทิศทางน้อยเกินไป จุด core ที่ได้ในขั้นตอนสุดท้าย จะไม่ตรงกับจุดในภาพลายนิ้วมือจริง ส่วนการกำหนดให้มีทิศทางมากเกินไป จะทำให้ภาพทิศทางมีทิศทางที่ละเอียดจนเกินไป จุด core ที่ได้ จะไม่ตรง อีกเช่นกัน

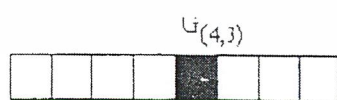
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



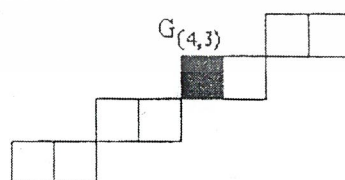
รูปที่ 17 แสดงทิศทางที่ใช้ในการคำนวณ



รูปที่ 18 แสดงการตัดบล็อกทิศทางและจำนวนพิกเซลที่ใช้ในการคำนวณ

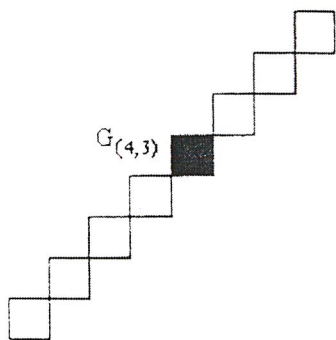


ทิศทางที่ 1

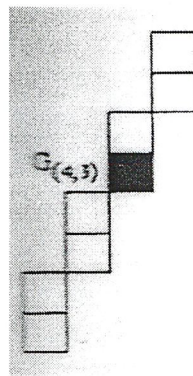


ทิศทางที่ 2

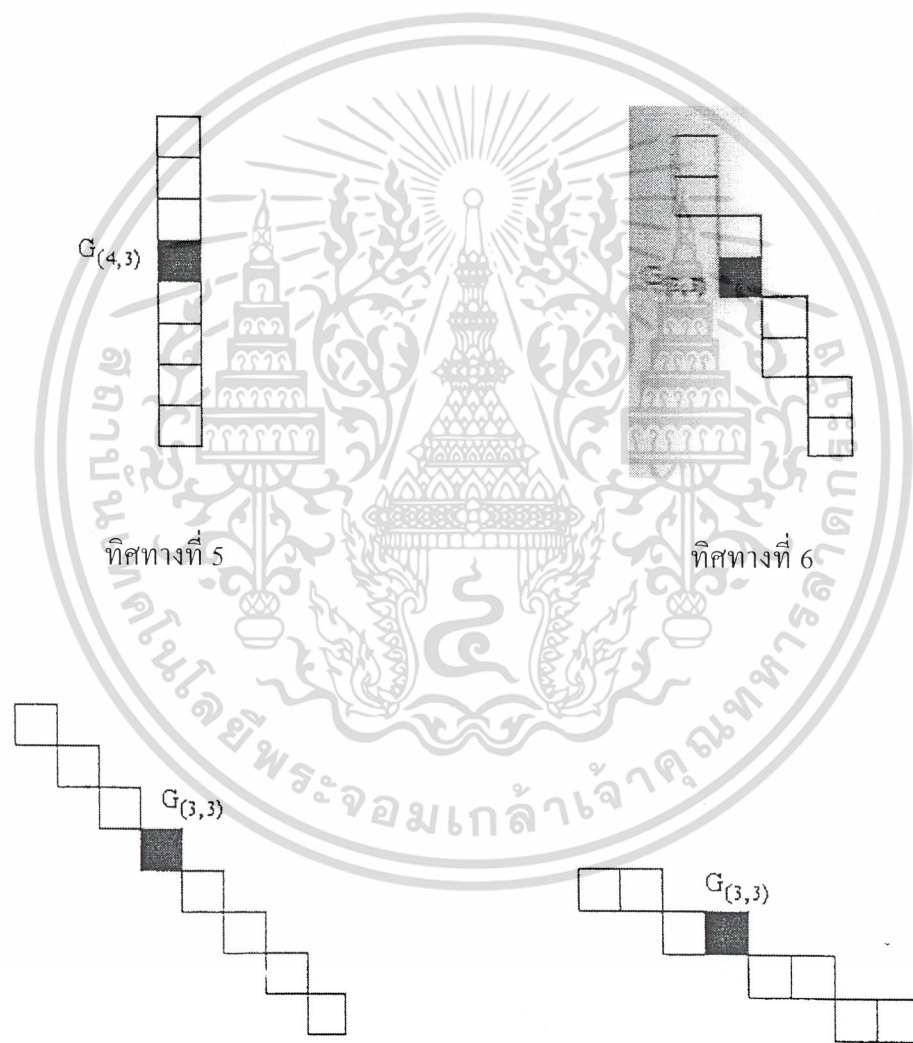
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ทิศทางที่ 3



ทิศทางที่ 4

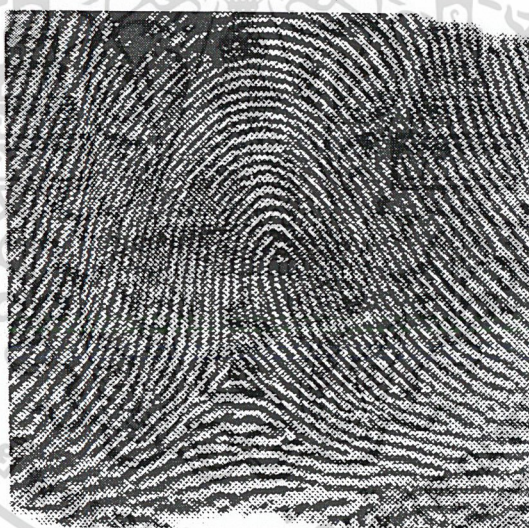


ทิศทางที่ 7

ทิศทางที่ 8

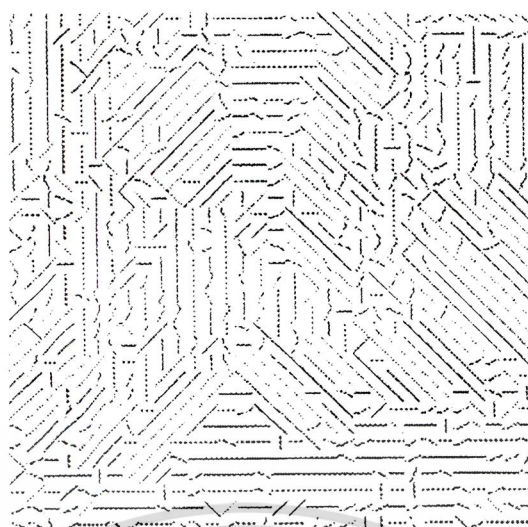
เอกสารนี้เป็นเอกสารที่รูปที่ 19 แสดงลักษณะพิกเซลที่ใช้ในการคำนวณตามทิศทาง d(1-8) โยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสูตรเมื่อ $G(i,j)$ และ $G(i_m,j_m)$ เป็นค่าระดับเทาที่ pixel (i,j) และ (i_m,j_m) ตามลำดับ pixel ที่พิจารณาไปยัง (i_m,j_m) คือ pixel อันดับที่ m ตามทิศทาง d จากจุด (i,j) ในการคำนวณเราพิจารณาให้จุด (i,j) เป็นจุด $(4,3)$ การหาผลรวมของ ทิศทางที่ 1 ถึง 5 เนื่องจาก ทิศทางที่ 1 ถึง 5 จะผ่าน pixel จุดที่ $(4,3)$ และกำหนดให้จุด (i,j) เป็นจุด $(3,3)$ ถ้าเราต้องการหาผลรวมของทิศทางที่ 6 ถึง 8 เพราะว่าทิศทางที่ 6 ถึง 8 จะผ่าน pixel จุดที่ $(3,3)$ การที่เราใช้จุด pixel จุด (i,j) ต่างกันจะไม่ส่งผลกระทบต่อใดๆ เนื่องจากว่าเราต้องการหาผลรวม ของผลต่างที่น้อยที่สุดในแต่ละทิศทางมากกว่า การที่มองจุดศูนย์กลางจุดใดจุดหนึ่ง เป็นจุดศูนย์กลางแล้วหมุนทิศทางไปตามทิศทางต่างๆ ทั้ง 8 ทิศทาง โดยมีจุดศูนย์กลางเดียวกัน มุมของแต่ละทิศทางจะมีขนาดต่างกันเท่ากับ 22.5 องศา ดังตัวอย่างภาพข้างล่างนี้



รูปที่ 20 แสดงภาพลายนิ้วมือที่ได้เก็บไว้ก่อนนำมาแปลงเป็นภาพทิศทาง

☺ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 21 แสดงภาพทิศทางที่แปลงมาจากภาพลายนิ้วมือ

เมื่อเราภาพทิศทางมาแล้ว ทิศทางทั้งหมด 8 ทิศทางเราจะแบ่งให้มีลักษณะอยู่ 3 ประเภท เพื่อใช้ประกอบในนิยามของบล็อกทิศทางแทนจุด core ในภาพทิศทาง คือ ทิศทางเฉียงซ้ายหมายถึง ทิศทางที่มีมุมน้อยกว่าหรือเท่ากับ 90 องศา ในที่นี้ก็คือทิศทางที่ 1 ถึง 5 ทิศทางเฉียงขวา หมายถึง ทิศทางที่มีมุมมากกว่า 90 องศา ซึ่งก็คือทิศทางที่ 5 ถึง 8 และทิศทางในแนวนอน หมายถึงทิศทางที่มีมุมน้อยกว่า 45 องศา หรือ มากกว่า 135 องศา นั่นก็คือทิศทางที่ 1,2,3,7 และ 8

จากบทที่ผ่านมา ในเรื่องของลักษณะจุดต่างๆ บนลายนิ้วมือ จุด core เป็นจุดบนเส้นลายนิ้ว (ridge) อยู่ส่วนบนที่สุดของเส้นโค้ง ของเส้นโค้งที่กำลังจะกลับตัว หรือกำลังจะโค้งลง เป็นเส้นโค้งลายนิ้วที่เกิดการกลับด้วยตัวมันเอง (เมื่อวิ่งขึ้นไปโค้งขึ้นแล้วจะโค้งลงกลับมาด้านล่าง) และเป็นเส้นโค้งในสุดของกลุ่มเส้นโค้งหรือกลุ่มรูปแบบอื่น ๆ ดังนั้น เมื่อมองในภาพทิศทางของบล็อกทิศทางที่แทนด้วยจุด core ในภาพทิศทางจะมีฮิสโตแกรมของทิศทางของส่วนที่อยู่ข้างเคียงทางด้านซ้าย หรือฮิสโตแกรมของทิศทางของส่วนที่อยู่ข้างเคียงขวา หรือฮิสโตแกรมของทิศทางที่ล้อมรอบอยู่ทางด้านขวาจะมีลักษณะทิศทางเฉียงขวา

4.3 การหาจุดศูนย์กลางของภาพ

การหาจุดศูนย์กลางของภาพ จุดประสงค์ของการหาจุดศูนย์กลางของภาพเพื่อเป็นมาตรฐานในกระบวนการขั้นตอนการตรวจสอบหรือทำฐานข้อมูล โดยการทำงานจะมีลักษณะตามขั้นตอนข้างล่างนี้

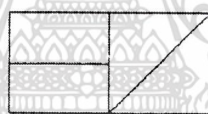
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.1 การหาจุดที่คาดว่าน่าจะเป็นจุด core

ในขั้นตอนนี้เป็นการเลือกบล็อกทิศทางที่ใช้แทนจุด core คาดว่าน่าจะเป็นจุด core ในภาพทิศทาง ซึ่งเราจะเลือกจุดที่คาดว่าน่าจะเป็นจุดดังกล่าวทุกจุดในภาพทิศทางโดยใช้หลักการดังนี้ จุด core บล็อกทิศทางที่เลือกเป็นบล็อกทิศทางของจุด core นั้นจะต้องมีมุมที่มีความแตกต่างจากบล็อกทิศทางทางซ้าย มากกว่าหรือเท่ากับ k_1 ในที่นี้เรากำหนดให้ k_1 มีค่าเท่ากับ 45 องศา เนื่องจากลักษณะของบล็อกทิศทางที่อยู่ล้อมรอบจุด core นั้นจะต้องมีบล็อกของส่วนประกอบที่อยู่ข้างเคียงด้านซ้ายมีลักษณะเอียงซ้าย และบล็อกของส่วนประกอบที่อยู่ทางด้านขวามีลักษณะเอียงขวา และมีความแตกต่างของมุมกับบล็อกข้างเคียงทางด้านซ้ายเท่ากับ 45 องศา ซึ่งเป็นไปตามสมการ

$$d \{ B(i,j) - B(i-1,j) \} \geq k_1$$

ให้ $B(i,j)$ เป็นบล็อกทิศทางของภาพที่แทนจุด core และคาดว่าน่าจะเป็นจุด core ในภาพทิศทางตัวอย่างของบล็อกทิศทางที่เลือกเป็นไปดังรูปที่ 22



$B(i-1, j)$ $B(i, j)$

รูปที่ 22 แสดงตัวอย่างของบล็อกทิศทางที่เลือกเป็นบล็อกทิศทางของจุด core

4.3.2 การกำจัดจุดที่ไม่น่าจะใช่ทิ้ง

ในขั้นตอนที่ผ่านมา เราได้ทำการเลือกบล็อกทิศทางที่คาดว่าน่าจะเป็นจุด core มาถึงขั้นตอนนี้จะเป็นการลบหรือกำจัดบล็อกที่ไม่น่าจะจะเป็นจุดเฉพาะดังกล่าวออก เพราะบล็อกที่เราเลือกมาในขั้นตอนแรกมีจำนวนมากมาย ดังนั้นเราจึงจำเป็นต้องกำจัดบล็อกที่ไม่น่าจะใช่บล็อกทิศทางของจุดเฉพาะเหล่านั้นออกเสียบ้าง เพื่อให้เหลือบล็อกทิศทางที่ใช้แทนจุด core ในภาพทิศทางมีน้อยลง เพื่อใช้ในการหาบล็อกทิศทางที่ใช้แทนจุดที่เป็นจุดเฉพาะในขั้นตอนต่อ ๆ ไป หลักในการลบจุดที่ไม่น่าจะใช่ทิ้ง เราจึงใช้เงื่อนไขดังต่อไปนี้

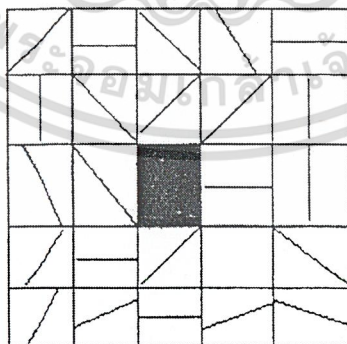
เราจะทำการอาศัยบล็อกทิศทางข้างเคียงที่อยู่ล้อมรอบมาร่วมพิจารณา วิธีการนี้จะทำการเอกลักษณะเป็นเอกสารที่ส่งมอบไว้สำหรับครูใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ตัดบล็อกล้อมรอบบล็อกของจุดที่คาดว่าน่าจะเป็นจุด core ในขั้นตอนแรกด้วยบล็อกขนาดไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5×5 แล้วพิจารณาว่าบล็อกทิศทางที่ล้อมรอบอยู่นั้นสนับสนุนให้บล็อกทิศทางที่เป็นจุด core ในภาพทิศทางนั้น ควรเป็นบล็อกทิศทางของจุด core ในภาพทิศทางนั้นหรือไม่ โดยใช้เงื่อนไขดังต่อไปนี้ในการพิจารณา

$$\text{เมื่อ} \quad \text{MAX}_{k=0}^{N-1} f_1 (H_4 (k)) < t_2$$

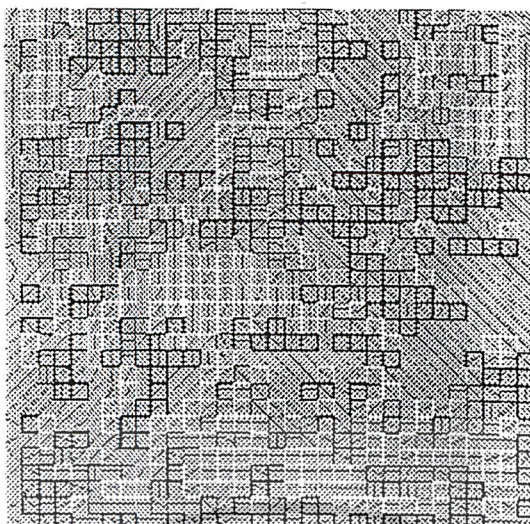
$$f_1 (H_4 (k)) = \sum_{j=(k-1) \bmod N}^{(k+1) \bmod N} H (j)$$

จากสมการข้างต้นฟังก์ชันของฮิสโตแกรม หรือความถี่ของทิศทางตั้งแต่ทิศทางที่ 0 ถึง 7 นั้นจะต้องมีค่าน้อยกว่า t_2 (ค่า threshold) โดยฟังก์ชันของความถี่ทิศทาง ที่ได้มาจากผลของความถี่ทิศทางข้างเคียงและทิศทางของฟังก์ชันนั่นเอง การ mod ด้วยค่า N นั้นเป็นวิธีการ mod ซึ่งเป็นลักษณะของการหาร แล้วเอาเฉพาะเศษเท่านั้นมาเป็นผลลัพธ์ โดยที่ผลลัพธ์นั้นจะมีค่าเท่ากับ 0 ถึง $N-1$ เท่านั้น ค่า t_2 นั้นเป็นค่าที่ได้มาจากการทดลอง เป็นตัวกำหนดว่าทิศทางที่อยู่ล้อมรอบจุด core ในภาพทิศทางควรมีทิศทางไปทางเดียวกันไม่เกินจำนวนเท่าใด การกำหนดให้มีย่านที่มีผลทำให้บล็อกทิศทางที่คาดว่าจะเป็นบล็อกทิศทางของจุด core ในภาพทิศทางที่ผ่านขั้นตอนนี้จะมีย่านที่น้อยลง และถ้ากำหนดให้มีย่านที่มากทำให้บล็อกทิศทางดังกล่าวผ่านขั้นตอนนี้ได้จำนวนมากขึ้น

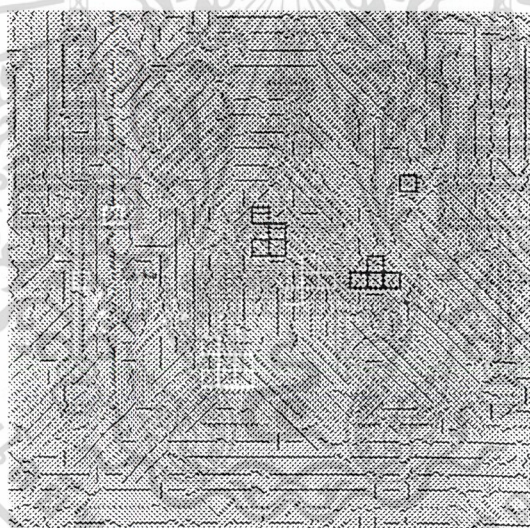


รูปที่ 23 แสดงทิศทางที่อยู่ข้างเคียงบล็อกทิศทางที่เราคาดว่าน่าจะเป็นจุด core

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 24 แสดงภาพของจุดที่เราคาดว่าจะจะเป็นจุด core ในภาพทิศทาง



รูปที่ 25 แสดงภาพของจุดที่คาดว่าน่าจะเป็นจุด core ในภาพทิศ
ทางหลังผ่านการกำจัดจุดที่ไม่น่าจะใช่จุดดังกล่าวทิ้ง

4.3.3 การตรวจสอบส่วนโค้ง

ในขั้นตอนนี้เป็นการนำบล็อกทิศทางที่คาดว่าน่าจะเป็นจุด core ในภาพทิศทางที่ผ่านขั้นตอนการกำจัดจุดที่ไม่น่าจะใช่ทิ้งมาแล้ว ซึ่งจะได้อบล็อกทิศทางของจุดที่คาดว่าน่าจะเป็นจุด core ในภาพทิศทางจำนวนหนึ่ง มาทำการตรวจสอบกับทิศทางที่อยู่ข้างเคียงว่าบล็อกทิศทางของจุดดังกล่าว นั้น อยู่ภายใต้ลักษณะส่วนโค้งที่มีลักษณะเป็นอย่างไรเป็นลักษณะของส่วน โค้งขึ้นหรือโค้งลง วิธีการของขั้นตอนนี้จะทำการตัดบล็อกล้อมรอบ ๆ จุด core ในภาพทิศทางและแบ่งบล็อกนี้เป็นการคำนวณว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ย่าน 4 ย่าน ในแต่ละย่านคือย่าน R_1, R_2, R_3 และ R_4 จะสร้างเป็นบล็อกที่มีขนาด 5×5 เพื่อใช้ในการพิจารณา

R_1	R_2
R_3	R_4

รูปที่ 26 แสดงการแบ่งย่านของบล็อกที่ล้อมรอบจุด core ในภาพทิศทาง

จากรูปที่ 28 เมื่อเราทำการแบ่งบล็อกที่ตัดเป็น 4 ย่านดังรูป ซึ่งจะประกอบไปด้วย ย่าน R_1, R_2, R_3 และ R_4 แล้ว นำความถี่ของทิศทาง H_1, H_2, H_3 และ H_4 ในย่าน R_1, R_2, R_3 และ R_4 ตามลำดับ มาพิจารณาหาลักษณะของส่วนโค้งในบล็อกที่เราตัดล้อมรอบจุด core โดยการพิจารณาการเอียงของทิศทางที่อยู่รอบ ๆ บล็อกทิศทางที่คาดว่าจะเป็นจุด core ในภาพทิศทางนั้น โดยใช้เงื่อนไขต่อไปนี้ในการพิจารณา

บล็อกทิศทางของจุดที่เป็นจุด core จะอยู่ในลักษณะของส่วนโค้งขึ้น

$$\text{ถ้า } f_{3+}(H_1, H_2) > f_{3-}(H_3, H_4)$$

$$\text{และ } f_{3+}(H_1, H_2) > t_3$$

บล็อกทิศทางของจุดที่เป็นจุด core จะอยู่ในลักษณะของส่วนโค้งลง

$$\text{ถ้า } f_{3+}(H_1, H_2) > f_{3-}(H_3, H_4)$$

$$\text{และ } f_{3-}(H_1, H_2) > t_3$$

และฟังก์ชัน f_3 กำหนดได้ดังนี้

$$f_{3+}(H_m, H_n) = \sum_{j=0}^{N-1} W_\alpha(j) H_m(j) + \sum_{j=0}^{N-1} W_\beta(j) H_n(j)$$

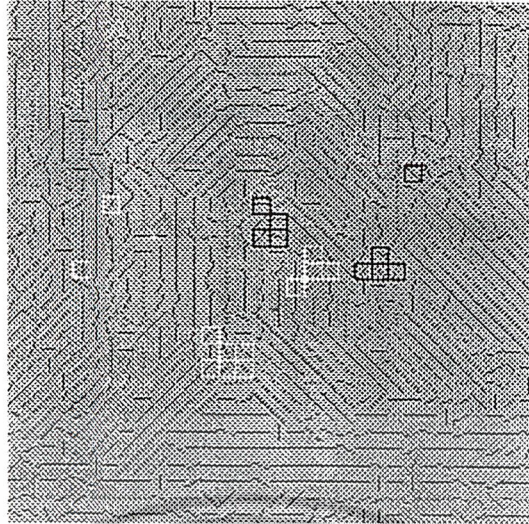
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ไปยังประชาชนด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$f_{3-}(H_m, H_n) = \sum_{j=0}^{N-1} W_\alpha(j) H_m(j) + \sum_{j=0}^{N-1} W_\beta(j) H_n(j)$$

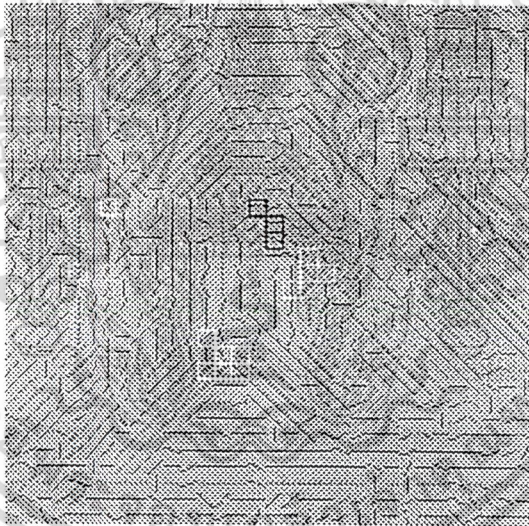
เมื่อฟังก์ชัน f_{3+} เป็นฟังก์ชันของฮิสโตแกรมทิศทางที่อยู่ในลักษณะของส่วนโค้งขึ้น ซึ่งเราจะพิจารณาในย่าน R_1 และ R_2 ส่วนฟังก์ชัน f_{3-} เป็นฟังก์ชันของฮิสโตแกรมทิศทางที่อยู่ในลักษณะของส่วนโค้งลงใช้พิจารณาในย่าน R_3 และ R_4

ค่าของ $w_\alpha(j)$ กับ $w_\beta(j)$ เป็นค่าคงที่ที่ได้กำหนดขึ้นมา โดยที่ค่าของ $w_\alpha(j)$ เมื่อ $\alpha=j$ จะมีค่ามากกว่าค่า $w_\alpha(j)$ เมื่อ $\alpha \neq j$ และค่า $w_\beta(j)$ เมื่อ $\beta=j$ จะมีค่ามากกว่าค่า $w_\beta(j)$ เมื่อ $\beta \neq j$ ค่าคงที่ทั้งสองตัวนี้ใช้คุณเข้าไปกับความถี่ของทิศทางเพื่อเป็นการสร้างความแตกต่างระหว่างทิศทางที่เราพิจารณา (เป็นทิศทางที่ทำให้เกิดลักษณะของส่วนโค้งขึ้นหรือโค้งลง) กับทิศทางอื่น ๆ และค่า threshold t_3 เป็นค่าที่ได้มาจากการทดลองใช้เมื่อพิจารณาฟังก์ชัน f_{3+} และฟังก์ชัน f_{3-} ว่าควรมีค่าเท่าใดจึงจะเป็นฟังก์ชันของลักษณะส่วนโค้งขึ้นหรือส่วนโค้งลง จากสมการข้างต้นถ้าทิศทางที่คิดว่าน่าจะเป็นจุด core ในภาพทิศทาง จุดใดอยู่ในลักษณะของส่วนโค้งขึ้นก็จะมีค่าของฟังก์ชัน f_{3+} ซึ่งอยู่ในย่าน ย่าน R_1 และ R_2 มากกว่าฟังก์ชัน f_{3-} ซึ่งอยู่ในย่านของ R_3 และ R_4 โดยที่ฟังก์ชัน f_{3+} ต้องมีค่ามากกว่า t_3 ด้วย ส่วนกรณีของลักษณะส่วนโค้งลง ฟังก์ชัน f_{3-} ที่อยู่ในย่าน R_3 และ R_4 ก็ต้องมากกว่าฟังก์ชัน f_{3+} ในย่าน R_1 และ R_2 รวมทั้งฟังก์ชัน f_{3-} จะต้องมากกว่าค่า t_3 ด้วยเช่นกัน ในการทดสอบเราใช้ค่า $w_\alpha(j)$ และค่า $w_\beta(j)$ มีค่าเท่ากับ 2 คุณเข้าไปในทิศทางที่เราพิจารณาในแต่ละย่านเท่านั้น ที่เราใช้ค่าเพียงแค่ 2 เพราะไม่ต้องการให้เกิดความแตกต่างระหว่างทิศทางที่เราต้องการพิจารณากับทิศทางอื่น ๆ มากจนเกินไป ซึ่งจากที่เราทำการตัดบล็อกทิศทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 27 แสดงจุดที่คาดว่าจะน่าจะเป็นจุด core ในภาพทิศทางก่อนการตรวจสอบส่วนโค้ง



รูปที่ 28 แสดงจุดที่คาดว่าจะน่าจะเป็นจุด core ในภาพทิศทางหลังการตรวจสอบส่วนโค้ง

ในแต่ละย่านให้มีขนาดใหญ่ก็ทำให้บล็อกทิศทางที่เราสร้างขึ้นสามารถครอบคลุมทิศทางที่ทำให้เกิดลักษณะส่วนโค้งที่เราต้องการแล้ว กรณีที่ทิศทางที่อยู่รอบข้างจุดที่เป็น core ในภาพทิศทางไม่ชัดเจน จุด core ในภาพทิศทางไม่สามารถผ่านการตรวจสอบส่วนโค้งไปได้เลย โปรแกรมที่เราเขียนขึ้นมาจะใช้ค่าที่ต่ำลงมาเอง โดยอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.4 การรวมกลุ่ม

ในขั้นตอนนี้เราจะนำบล็อกทิศทางที่ผ่านขั้นตอนที่แล้ว มาทำการจัดบล็อกทิศทางเหล่านั้น ให้เป็นกลุ่ม ๆ ซึ่งจากการทดลองหลังจากที่ผ่านขั้นตอนของการตรวจสอบของลักษณะส่วน โถ่ง จุดที่คาดว่าจะจะเป็นจุด **core** ในภาพทิศทาง จะวางตัวอยู่ในลักษณะเป็นกลุ่ม ๆ อยู่แล้ว แต่การจัดกลุ่มของบล็อกทิศทางในขั้นตอนนี้ ก็เพื่อเป็นการหาจุดศูนย์กลางและการรวมกลุ่มเหล่านั้น ซึ่งจะวางตัวอยู่ในบริเวณที่มีลักษณะรูปแบบเป็นบริเวณของจุด **core** และมักมีการเอนเอียงขึ้น เราจึงจำเป็นต้องหาลักษณะการเอียงของบริเวณที่มีลักษณะดังกล่าว ซึ่งจะกล่าวในขั้นตอนต่อไป

หลักการในการรวมกลุ่มและการหาจุดศูนย์กลางของกลุ่มใช้ทฤษฎีของ **K – Mean** ร่วมกับค่าของ **threshold** t_1 ซึ่งจะเพิ่มเข้าไปเป็นเงื่อนไขในขั้นตอนการหาสมาชิก โดยจากเดิมในขั้นตอนที่ 2 ของ **K – Mean** ใช้การพิจารณาการหาสมาชิกจากจุดใด ๆ โดยกำหนดว่าจุดนั้นใกล้จุดศูนย์กลางจุดใดมากกว่ากัน ก็กำหนดให้จุดนั้นเป็นสมาชิกของจุดศูนย์กลางที่ใกล้ที่สุด แต่เงื่อนไขที่เพิ่มเข้าไปจะกำหนดด้วยว่าจุดที่จะเป็นสมาชิกของจุดศูนย์กลางใด ๆ ต้องมีระยะห่างจากจุดศูนย์กลางนั้นไม่เกินค่า t_1 และถ้าเกินค่า t_1 แล้ว จุดนั้นยังไม่เป็นสมาชิกของจุดศูนย์กลางใดเลย ก็จะกำหนดให้จุดนั้นเป็นจุดศูนย์กลางตัวใหม่ วิธีการทั้งหมดในขั้นตอนนี้คือ ในขั้นตอนแรกเลือกบล็อกใดก็ได้เป็นจุดศูนย์กลางกลุ่ม ซึ่งกำหนดให้เป็นจุดศูนย์กลางเริ่มต้น จากนั้นหาสมาชิกโดยวัดระยะทางของจุด X ใดๆ ถึงจุดศูนย์กลางกลุ่มต้องน้อยกว่า t_1 และถ้ามากกว่าแล้วจุด X จุดนั้นจะไม่เป็นสมาชิกของกลุ่ม หลังจากที่ทำกรหาสมาชิกของจุดศูนย์กลางที่จุดเรียบร้อยแล้ว ถ้าจุดใดยังไม่เป็นสมาชิกของกลุ่มใดเลย จะให้จุดนั้นเป็นจุดศูนย์กลางเริ่มต้นอีกตัว จากนั้นทำการปรับปรุงจุดศูนย์กลางของกลุ่ม โดยการนำสมาชิกทั้งหมดของกลุ่มมาใช้ในการคำนวณหาจุดศูนย์กลางใหม่ ขั้นตอนสุดท้ายถ้าจุดศูนย์กลางที่ได้มีตำแหน่งเปลี่ยนไปจากเดิม ก็ให้กลับไปทำการหาสมาชิกใหม่โดยใช้จุดศูนย์กลางของกลุ่มตำแหน่งใหม่มาหาสมาชิกใหม่ ซึ่งตอนนี้จะมีการเปลี่ยนแปลงของจำนวนสมาชิก ทำเช่นนี้ไปเรื่อย ๆ จนกว่าจะได้จุดศูนย์กลางของกลุ่มที่คำนวณมาใหม่มีตำแหน่งเดียวกับตำแหน่งเดิม จึงเป็นอันว่าเสร็จสิ้น จุดมุ่งหมายของขั้นตอนนี้ นอกจากจะหาจุดศูนย์กลางกลุ่ม แล้วยังต้องทำการจัดแบ่งจุดที่มีทั้งหมดเป็นกลุ่มซึ่งส่วนใหญ่จะสามารถแบ่งได้โดยสะดวก เพราะจากการทดลองจุดที่มีทั้งหมดที่เข้าในตอนนี้จะเกาะกลุ่มกันเองอยู่แล้ว แล้วจึงนำมาใช้ในการคำนวณขั้นตอนต่อไป ค่า t_1 กำหนดมีค่าเท่ากับ 16 เพราะจะการทดลองจุดที่เป็นจุด **core** ในภาพทิศทางที่ผ่านจากขั้นตอนที่ 3 มาแล้ว จะมีลักษณะการเกาะกลุ่มรวมกันอยู่ ค่า **threshold** ที่กำหนดสามารถควบคุมจุดทั้งหมดของกลุ่มได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.5 การหาแกนของพื้นที่

ในการหาแกนของพื้นที่ เป็นการหาลักษณะการเอนเอียงไปของบริเวณที่มีลักษณะเป็น core ซึ่งภาพหลายนิ้วมือที่แปลงเป็นภาพทิศทางมักจะมามีบริเวณ ที่มีลักษณะเป็น core เอนเอียงไป แต่การหาแกนในขั้นตอนนี้ก็ยังสามารถรวมทิศทางปกติของบริเวณที่มีลักษณะเป็น core ที่ไม่มีการเอนเอียงเอาไว้ด้วยเพื่อความถูกต้อง ในกรณีที่บริเวณดังกล่าวไม่มีการเอนเอียง การหาแกนหลักของกลุ่ม หลังจากที่เราได้ทำการจัดรวมกลุ่มและหาจุดศูนย์กลางของกลุ่ม ในขั้นตอนที่แล้ว ๆ วิธีการหาแกนหลักนี้แกนจะถูกหมุนไปตามทิศทางต่าง ๆ เป็นจำนวน 8 ทิศทาง (ใช้ทิศทางของการแปลงภาพหลายนิ้วมือไปเป็นภาพทิศทางในขั้นตอนแรก) จากนั้นเราจึงวิเคราะห์ส่วนประกอบข้างเคียง ที่อยู่ใกล้กับจุดศูนย์กลางในย่านที่เรากำหนด ขั้นตอนนี้คือ เราจะทำการตัดบล็อกกรอบบล็อกทิศทางขนาด 5×7 ล้อมรอบจุดศูนย์กลางของกลุ่มที่เป็นจุด core ที่หามาจากขั้นตอนที่แล้ว ต่อมาหมุนแกน A_k ไปเป็นมุมเท่ากับ μ/N เรเดียน นำทิศทางของบล็อกทิศทางข้างเคียงมาช่วยในการคำนวณ



รูปที่ 29 แสดงส่วนที่อยู่ข้างเคียงที่ใช้ในการกำหนดแกนของพื้นที่

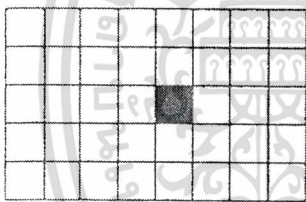
การหมุนของแกน A_k นั้น ได้มาจากการคำนวณโดยอาศัยความถี่ของทิศทาง H_μ และ $\mu = 0 \dots N-1$ ซึ่งเป็นความถี่ของทิศทางของบล็อกข้างเคียงที่ล้อมรอบจุดศูนย์กลางของกลุ่มที่อยู่ในวงปิดของบล็อกที่เราได้สร้างขึ้น และแกนที่หมุนไปตามค่าของ μ นั้นเป็นค่าของฟังก์ชัน f_5 ที่มีค่าสูงสุด กำหนดได้ดังนี้

$$f_5(H_\alpha) > f_{5\mu}(H_\mu) \quad \text{ที่ทุกๆ ค่าของ } \alpha \neq \mu, 0 \leq \alpha, \mu \leq N-1$$

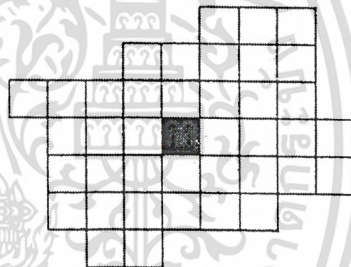
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$f_5(H) = \sum_{i=0}^{N-1} W_\mu(i) H(i)$$

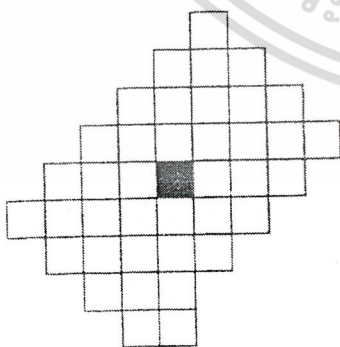
สรุปแล้วเมื่อเราได้ทำการหมุนแกน แล้วเราจะพิจารณาทิศทางของบล็อกทิศทางข้างเคียงที่อยู่ล้อมรอบจุดศูนย์กลาง ว่ามีค่ามากที่สุดเมื่อเปรียบเทียบกับการหมุนไปตามทิศทางอื่นหรือไม่ ถ้ามากที่สุดก็จะกำหนดให้แกนมีลักษณะการหมุนไปตามทิศทางนั้น แต่ถ้าไม่มากที่สุดก็จะหาแกนในทิศทางอื่นที่มีค่าของทิศทางที่อยู่ล้อมรอบมากที่สุดแทน $W_\mu(i)$ มากใช้เป็นตัวคูณเมื่อมุมของแกนถูกหมุนไป μ เพื่อเป็นการยกระดับความแตกต่างของทิศทางอื่นกับทิศทางที่แกนหมุนไป ดังนั้นการหาว่าแกนของพื้นที่ที่เป็นจุดศูนย์กลางจะหมุนไปที่องศาที่จะหาจากผลรวมของความถี่ของบล็อกทิศทางข้างเคียง ในบล็อกที่เรากำหนดขนาดขึ้นมา โดยที่เมื่อหมุนแกนไปเท่ากับ μ ค่าความถี่ของทิศทางที่ตรงกับค่าของมุมที่หมุนจะถูกนำมาคูณกับค่าคงที่ $W_\mu(i)$ ที่มีค่ามากที่สุด ลักษณะการหมุนบล็อกที่เราใช้ล้อมรอบจุดศูนย์กลางสามารถพิจารณาได้จากรูปด้านล่าง



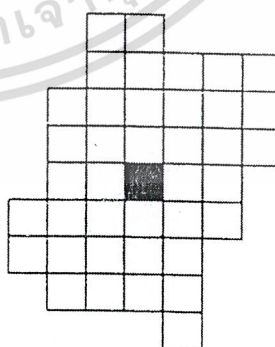
การหมุนทิศทางที่ 1



การหมุนทิศทางที่ 2

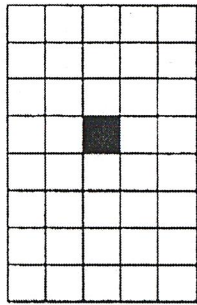


การหมุนทิศทางที่ 3

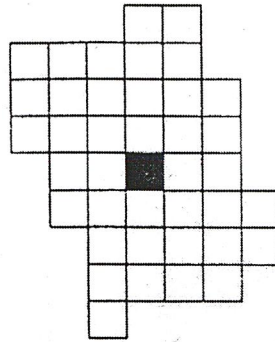


การหมุนทิศทางที่ 4

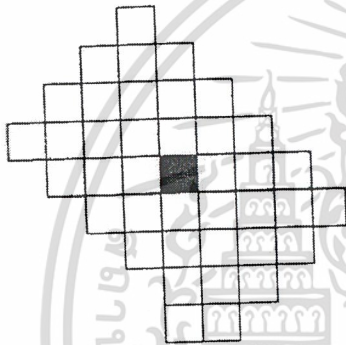
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



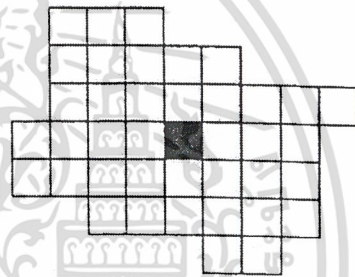
การหมุนทิศทางที่ 5



การหมุนทิศทางที่ 6



การหมุนทิศทางที่ 7



การหมุนทิศทางที่ 8

รูปที่ 30 แสดงรูปแบบพิกเซลที่นำมาใช้คำนวณในการหมุน

4.3.5 การหาบล็อกทิศทางที่แทนจุด core

ในขั้นตอนสุดท้ายของการหาบล็อกทิศทางที่แทนจุด core ในภาพทิศทาง โดยการนำบล็อกทิศทางที่แทนจุด core ที่อยู่ในแต่ละกลุ่มมาพิจารณาว่าบล็อกทิศทางใดมีความน่าจะเป็นสูงที่สุด ที่จะเป็นบล็อกทิศทางที่แทนจุด core ในภาพทิศทางได้

บล็อกทิศทางของจุด core จะต้องมีความน่าจะเป็น P มากที่สุด และความน่าจะเป็น P เป็นฟังก์ชันของความถี่ทิศทาง H_1, H_2, H_3 และ H_4 ในย่าน R_1, R_2, R_3 และ R_4 ตามลำดับ และย่านดังกล่าวนี้เราจะทำการแบ่งตามลักษณะรูปที่ได้แสดงไว้ด้านล่าง ซึ่งเป็นการหาทิศทางของโค้งขึ้นเท่านั้น เพราะเป็นไปตามค่านิยมของจุด core ดังนั้นความน่าจะเป็น P จะพิจารณาเฉพาะจุดที่โค้งขึ้นเท่านั้น โดยเราจะใช้หลักการดังต่อไปนี้

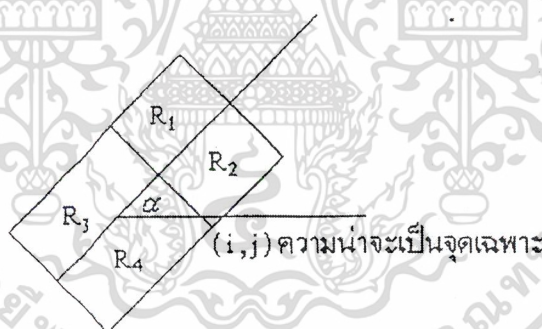
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$P = \sum_{j=1}^4 f_{7\alpha}(H_j)$$

ฟังก์ชัน $f_{7\alpha}$ กำหนดได้จาก

$$f_{7\alpha}(H) = \sum_{i=1}^{N-1} w_{\alpha}(i) H(i)$$

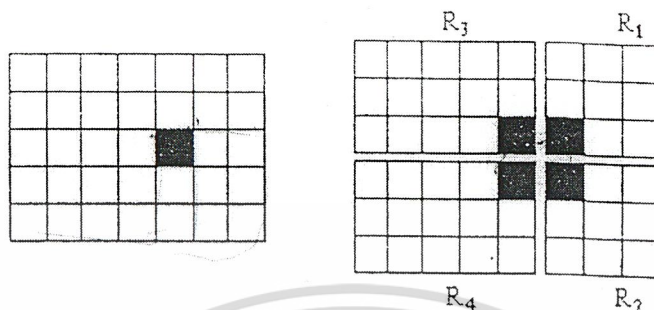
ในกรณีนี้เราจะไม่พิจารณาจุดที่เป็นโค้งลง เพราะจุดที่เป็นจุด CORE จากความหมายนั้น จะมีส่วนประกอบเป็นโค้งขึ้นเท่านั้น การใช้ค่า $w_{\alpha}(i)$ จะพิจารณาจากมุมที่แกน A_k ถูกหมุนไปจากขั้นตอนที่ผ่านมา โดยให้ค่า $w_{\alpha}(i)$ ที่ตรงกับมุมที่แกนหมุนไปมีค่ามากที่สุด



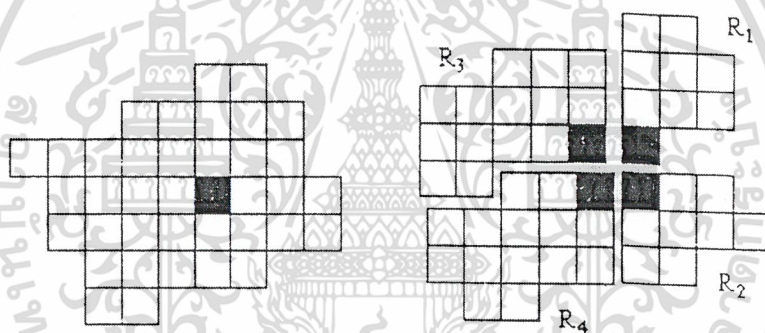
รูปที่ 31 แสดงบล็อกทิศทางข้างเคียงของบล็อกทิศทางที่แทนจุดเฉพาะ

จากรูปด้านล่าง การแบ่งย่านทั้ง 4 ย่านที่ใช้พิจารณาจุด CORE นั้น จะไม่อยู่ตรงกลางบล็อก เนื่องจากลักษณะเส้นโค้งที่มีจุด CORE เป็นเส้นโค้งสุดท้าย จึงมีส่วนที่อยู่ด้านล่างของจุด CORE เป็นทิศทางตามแกนการหมุน ส่วนที่อยู่เหนือจุด CORE เป็นทิศทางที่มีการเอียงเพียงเล็กน้อย ถ้าให้บล็อกทิศทางที่เป็นจุด CORE อยู่ตรงกลาง จะทำให้จุดที่ตรวจจับได้อยู่เหนือจุด CORE ขึ้นไป เนื่องจากการให้บล็อกทิศทางที่เป็นจุด CORE อยู่ตรงกลางส่วนที่ใช้ตรวจจับการเอกสารนี้เป็นเอกสารที่สแกนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

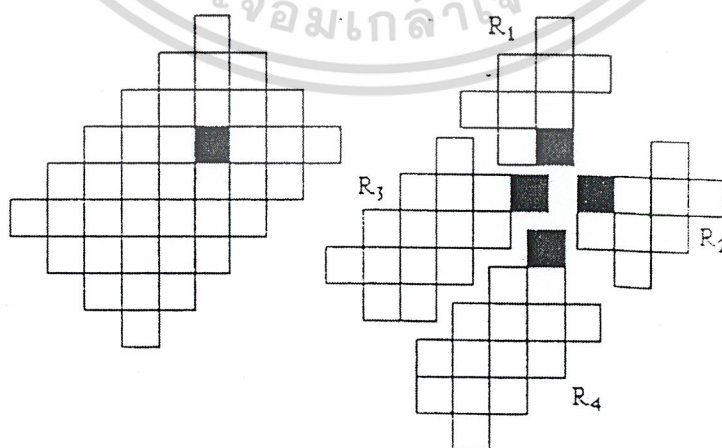
เอียงของจุด core ในย่าน R_1 และ R_2 ได้ค่ามากขึ้นไป ซึ่งในภาพทิศทางนี้การเอียงในย่าน R_1 และ R_2 ที่อยู่เหนือจุด core จะมีเพียงเล็กน้อยเท่านั้น



การหมุนทิศทางที่ 1

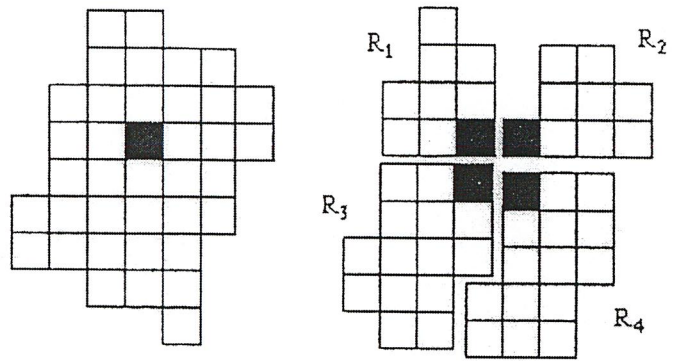


การหมุนทิศทางที่ 2

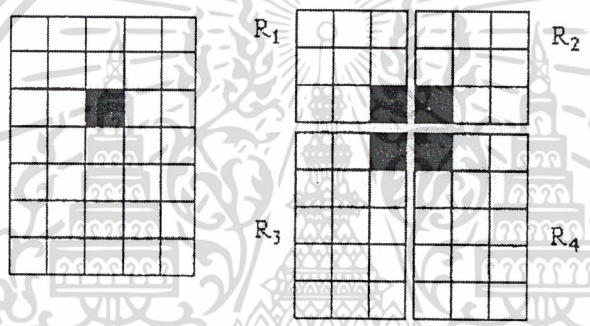
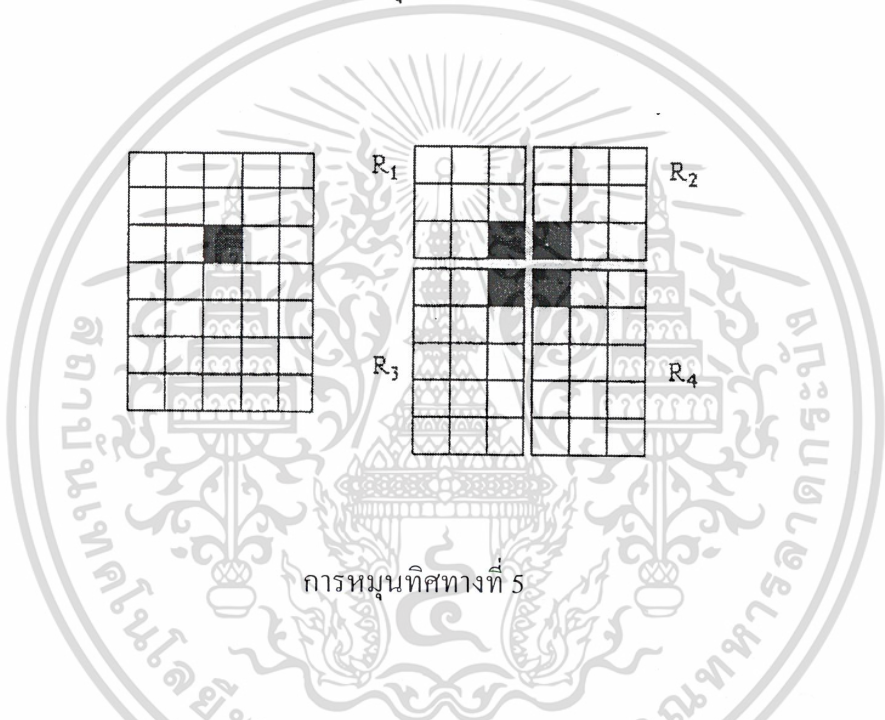


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแบบลงเน็ต และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

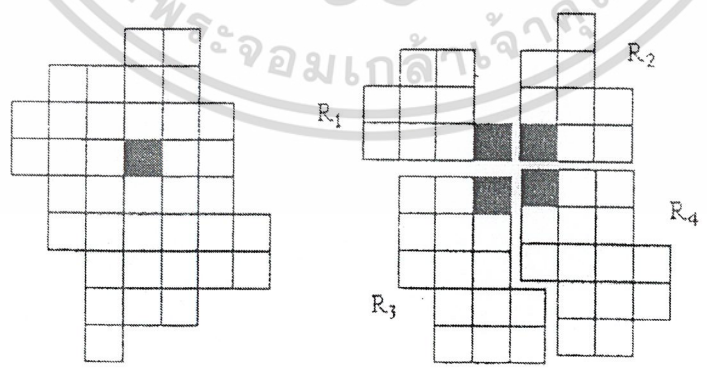
การหมุนทิศทางที่ 3



การหมุนทิศทางที่ 4

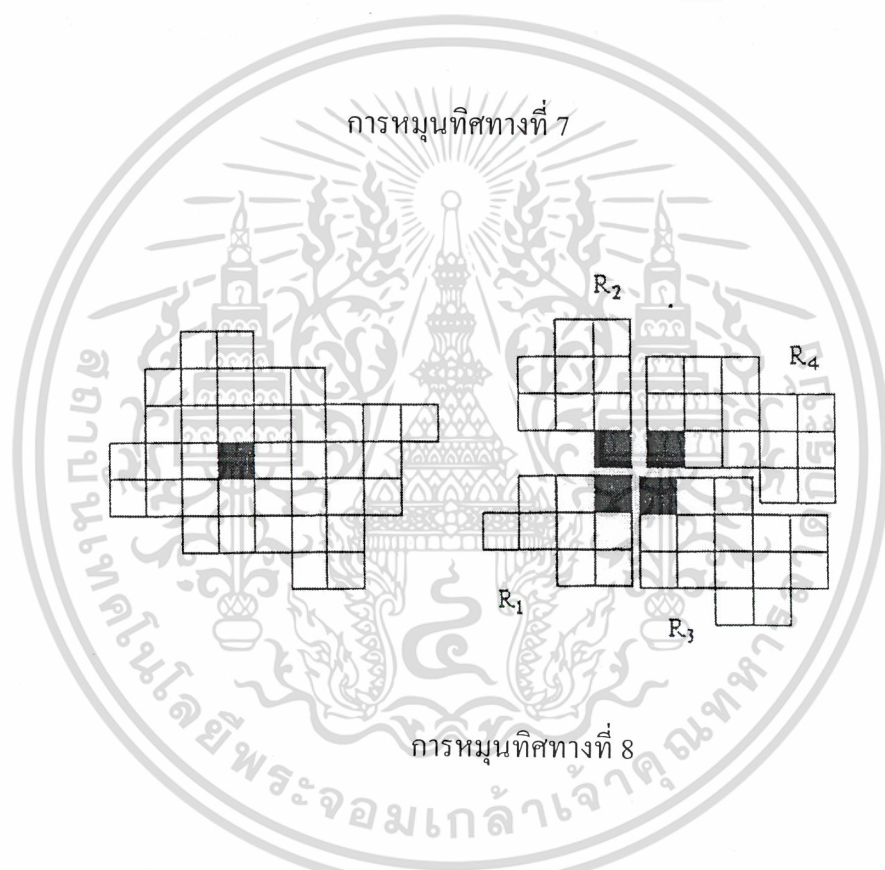
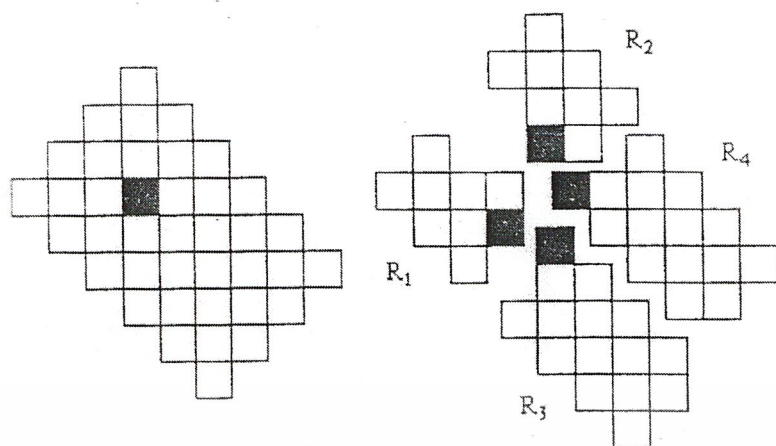


การหมุนทิศทางที่ 5



การหมุนทิศทางที่ 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



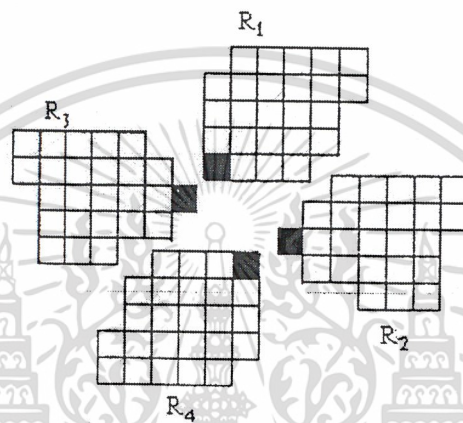
รูปที่ 32 แสดงการหมุนในทิศทางต่างๆและการแบ่งย่านในการหาจุด core

4.4 เพิ่มเติมในขั้นตอนการตรวจสอบส่วนโค้ง

ก่อนที่เราจะพิจารณาเงื่อนไขว่าจุด core นั้น อยู่ในลักษณะส่วนโค้งขึ้นหรือโค้งลง เรา จะทำการหมุนบล็อกที่เราตัดล้อมรอบจุด core ไปตามทิศทางต่างๆ (ทิศทางที่ 2 ถึง ทิศทางที่ 8) เนื่องจากกรณีที่ภาพลายนี้นี้มีมือที่เราเก็บมาได้มีการเอียงเกิดขึ้น (ถ้าไม่มีการเอียงบล็อกที่ล้อมรอบ จุด core) ขั้นตอนการตรวจสอบส่วนโค้งจะทำให้จุด core ที่ถูกต้องหรือจุดที่ใกล้เคียงจุด core ที่แท้จริงอาจถูกลบไปได้ เนื่องจากทิศทางข้างเคียงไม่สนับสนุนหรือทิศทางข้างเคียงที่เรา ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

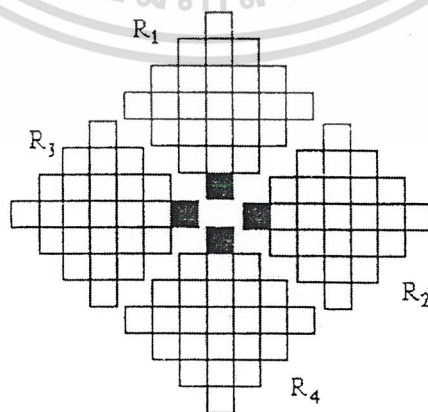
นำมาใช้ในการคำนวณมีการเอียงเปลี่ยนทิศทางไป เมื่อนำมาตรวจสอบส่วนโค้งจุดดังกล่าวจึงไม่สามารถผ่านการตรวจสอบส่วนโค้งไปได้

กรณีการหมุนบล็อกไปทิศทางที่ 2 เราจะพิจารณาทิศทางหลักๆ ในย่านทั้ง 4 ดังต่อไปนี้ ย่าน R_1 จะพิจารณาในทิศทางที่ 8 , ย่าน R_2 จะพิจารณาทิศทางที่ 4 , ย่าน R_3 จะพิจารณาทิศทางที่ 4 และ ย่าน R_4 จะพิจารณาทิศทางที่ 8 ดังรูปข้างล่างจะเป็นลักษณะการแบ่งย่านต่างๆ



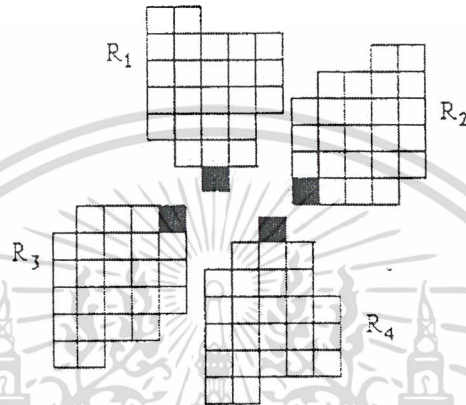
รูปที่ 33 แสดงการแบ่งย่านทั้งสี่เมื่อหมุนไปตามทิศทางที่ 2

กรณีการหมุนบล็อกไปทิศทางที่ 3 เราจะพิจารณาทิศทางหลักๆ ในย่านทั้ง 4 ดังต่อไปนี้ ย่าน R_1 จะพิจารณาในทิศทางที่ 1 , ย่าน R_2 จะพิจารณาทิศทางที่ 5 , ย่าน R_3 จะพิจารณาทิศทางที่ 5 และ ย่าน R_4 จะพิจารณาทิศทางที่ 1 ดังรูปข้างล่างจะเป็นลักษณะการแบ่งย่านต่างๆ



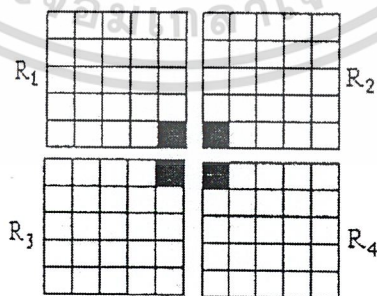
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 34 แสดงการแบ่งย่านทั้งสี่เมื่อหมุนไปตามทิศทางที่ 3
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีการหมุนบล็อกไปทิศทางที่ 4 เราจะพิจารณาทิศทางหลักๆ ในย่านทั้ง 4 ดังต่อไปนี้ ย่าน R_1 จะพิจารณาในทิศทางที่ 2 , ย่าน R_2 จะพิจารณาทิศทางที่ 6 , ย่าน R_3 จะพิจารณาทิศทางที่ 6 และ ย่าน R_4 จะพิจารณาทิศทางที่ 2 ดังรูปข้างล่างจะเป็นลักษณะการแบ่งย่านต่างๆ



รูปที่ 35 แสดงการแบ่งย่านทั้งสี่เมื่อหมุนไปตามทิศทางที่ 4

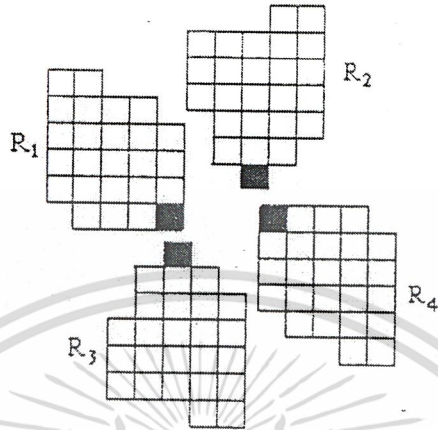
กรณีการหมุนบล็อกไปทิศทางที่ 5 เราจะพิจารณาทิศทางหลักๆ ในย่านทั้ง 4 ดังต่อไปนี้ ย่าน R_1 จะพิจารณาในทิศทางที่ 3 , ย่าน R_2 จะพิจารณาทิศทางที่ 7 , ย่าน R_3 จะพิจารณาทิศทางที่ 7 และ ย่าน R_4 จะพิจารณาทิศทางที่ 3 ดังรูปข้างล่างจะเป็นลักษณะการแบ่งย่านต่างๆ



รูปที่ 36 แสดงการแบ่งย่านทั้งสี่เมื่อหมุนไปตามทิศทางที่ 5

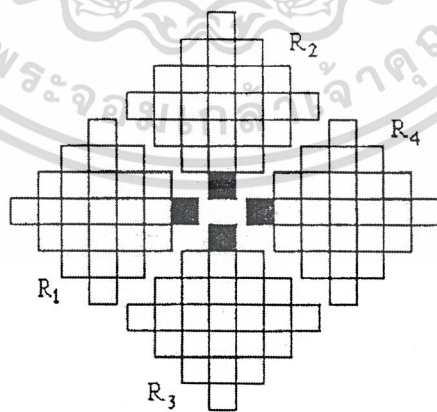
เอกสารนี้เป็นกรณีการหมุนบล็อกไปทิศทางที่ 6 เราจะพิจารณาทิศทางหลักๆ ในย่านทั้ง 4 ดังต่อไปนี้ การค้า ย่าน R_1 จะพิจารณาในทิศทางที่ 4 , ย่าน R_2 จะพิจารณาทิศทางที่ 8 , ย่าน R_3 จะไปใช้

พิจารณาทิศทางที่ 8 และ ย่าน R_4 จะพิจารณาทิศทางที่ 4 ดังรูปข้างล่างจะเป็นลักษณะการแบ่งย่านต่าง ๆ



รูปที่ 37 แสดงการแบ่งย่านทั้งสี่เมื่อหมุนไปตามทิศทางที่ 6

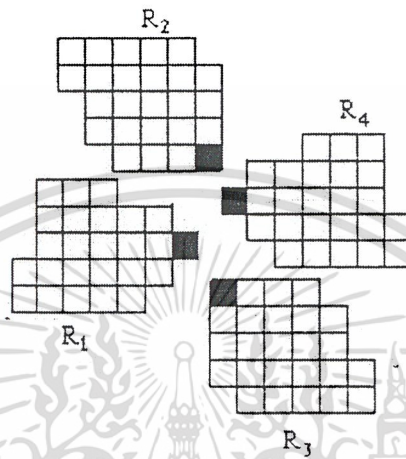
กรณีการหมุนบล็อกไปทิศทางที่ 7 เราจะพิจารณาทิศทางหลัก ๆ ในย่านทั้ง 4 ดังต่อไปนี้ ย่าน R_1 จะพิจารณาในทิศทางที่ 5 , ย่าน R_2 จะพิจารณาทิศทางที่ 1 , ย่าน R_3 จะพิจารณาทิศทางที่ 1 และ ย่าน R_4 จะพิจารณาทิศทางที่ 5 ดังรูปข้างล่างจะเป็นลักษณะการแบ่งย่านต่าง ๆ



รูปที่ 38 แสดงการแบ่งย่านทั้งสี่เมื่อหมุนไปตามทิศทางที่ 7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีการหมุนบล็อกไปทิศทางที่ 8 เราจะพิจารณาทิศทางหลัก ๆ ในย่านทั้ง 4 ดังต่อไปนี้ ย่าน R_1 จะพิจารณาในทิศทางที่ 6 , ย่าน R_2 จะพิจารณาทิศทางที่ 2 , ย่าน R_3 จะพิจารณาทิศทางที่ 2 และ ย่าน R_4 จะพิจารณาทิศทางที่ 6 ดังรูปข้างล่างจะเป็นลักษณะการแบ่งย่านต่าง ๆ



รูปที่ 39 แสดงการแบ่งย่านทั้งสี่เมื่อหมุนไปตามทิศทางที่ 8

4.5 การทำฐานข้อมูล

เมื่อเราได้ภาพทิศทางที่หาบล็อกของจุด core ได้แล้ว เราจะมาทำฐานข้อมูล โดยทำการตัดบล็อกขนาด 7×7 ล้อมรอบจุด core (ภาพที่ผ่านการหาทิศทางของภาพแล้ว จะมีค่าคงที่ประจำอยู่ในแต่ละบล็อก เช่น ทิศทางที่ 1 ให้มีค่าประจำบล็อกเท่ากับ 1 ทิศทางที่ 2 ให้มีค่าประจำบล็อกเท่ากับ 2 จนถึง ทิศทางที่ 8 ให้มีค่าประจำบล็อกเท่ากับ 8) และเราจะใช้ทฤษฎีของ นิเวรอนเน็ตเวิร์คแบบคิจิตอล คือ เราจะให้มีสมาชิกนิเวรอน 50 ตัว โดยนิเวรอนแต่ละตัวมี 8 cell (8 ขา หรือ 8 ตำแหน่ง) โดยแต่ละ cell จะมีตำแหน่ง xy ระบุไว้ที่แต่ละ cell จากนั้นเราจะทำการเก็บค่าที่อยู่ใน cell นั้นๆ มาเก็บในฐานข้อมูล โดยนิเวรอน 1 ตัวจะเก็บค่าไว้จำนวน 8 ค่า

สมาชิกนิเวรอนตัวที่ 1 คือ G1 มี 8 cell (8 ขา หรือ 8 ตำแหน่ง)

G2 มี 8 cell (8 ขา หรือ 8 ตำแหน่ง)

⋮

G50 มี 8 cell (8 ขา หรือ 8 ตำแหน่ง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ญาติเห็นหรือใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 การเปรียบเทียบ

การเปรียบเทียบภาพนั้นเราจะนำภาพที่นำมาเปรียบเทียบ มาทำตามขั้นต่าง ๆ ดังที่กล่าวมาแล้วข้างต้น คือ ขั้นแรก ทำการ **histogram equalization** ต่อมาทำการหาทิศทางของภาพ จากนั้นทำขบวนการหาบล็อกของจุด **core** เมื่อเราได้ภาพที่ผ่านการหาทิศทางของจุด **core** ได้แล้ว ก็ทำการตัดบล็อกขนาด $7*7$ ที่ล้อมรอบ บล็อกทิศทางที่เป็นจุด **core** จากนั้นเราให้นิวรอนตัวแรก (ที่ระบุค่าของตำแหน่ง **xy** เอาไว้แล้ว) ไปจับตำแหน่งของภาพที่จะนำมาเปรียบเทียบ แล้วทำการนำค่าที่อยู่ในตำแหน่งนั้น ๆ มาทำการเปรียบเทียบกับข้อมูลที่มีอยู่ในฐานข้อมูล โดยนิวรอน 1 ตัว จะมีการจับค่าไว้ได้ 8 ค่า แล้วจะนำค่าที่ได้ไปเปรียบเทียบในฐานข้อมูล โดยเปรียบเทียบกันแบบบิตต่อบิต เราจะให้มีการผิดพลาด 20 % แล้วทำนิวรอนให้ครบทั้ง 50 ตัว ในขั้นตอนสุดท้ายเราจะนำผลของนิวรอนแต่ละตัวมารวมกันว่าจะให้ผลเป็นของบุคคลใด

ตัวอย่าง

G1 มีค่าในฐานข้อมูลคือ 68336263

G2 ” 28132472

.

.

G50 ” 81451276

เรานำภาพที่จะมาเปรียบเทียบ แล้วนำนิวรอนไปจับที่ตำแหน่งนั้น ๆ (ตำแหน่งที่ระบุไว้ในนิวรอนที่ **cell** ต่าง ๆ) เพื่อนำค่ามาเปรียบเทียบ เช่น ค่าที่ได้คือ

G1 ค่าที่เก็บมาได้ คือ 68346163

G2 ” 18142462

.

.

.

G50 ” 51451216

เปรียบเทียบกันแบบบิตต่อบิต คือ

$$\begin{aligned} G1 &= |(6-6)|+|(8-8)|+|(3-3)|+|(3-4)|+|(6-6)|+|(2-1)|+|(6-6)|+|(3-3)| \\ &= 2 \end{aligned}$$

สมมุติว่าค่าที่เปรียบเทียบแล้วให้ผลออกมาว่าเป็นบุคคล **A** มากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

G2 = ให้ผลออกมาเป็น บุคคล B มากที่สุด

.

.

.

G50 = ให้ผลออกมาเป็น ไม่พบในฐานข้อมูล

สุดท้าย นำนิเวรอนแต่ละมารวมกันว่าให้ผลออกมาเป็นใครมากที่สุด

เช่น บุคคล A = 30 ตัว

บุคคล B = 15 ตัว

ไม่พบในฐานข้อมูล = 5 ตัว

สรุปลายนิ้วมือที่นำมาเปรียบเทียบ เป็นบุคคล A ที่พบในฐานข้อมูล



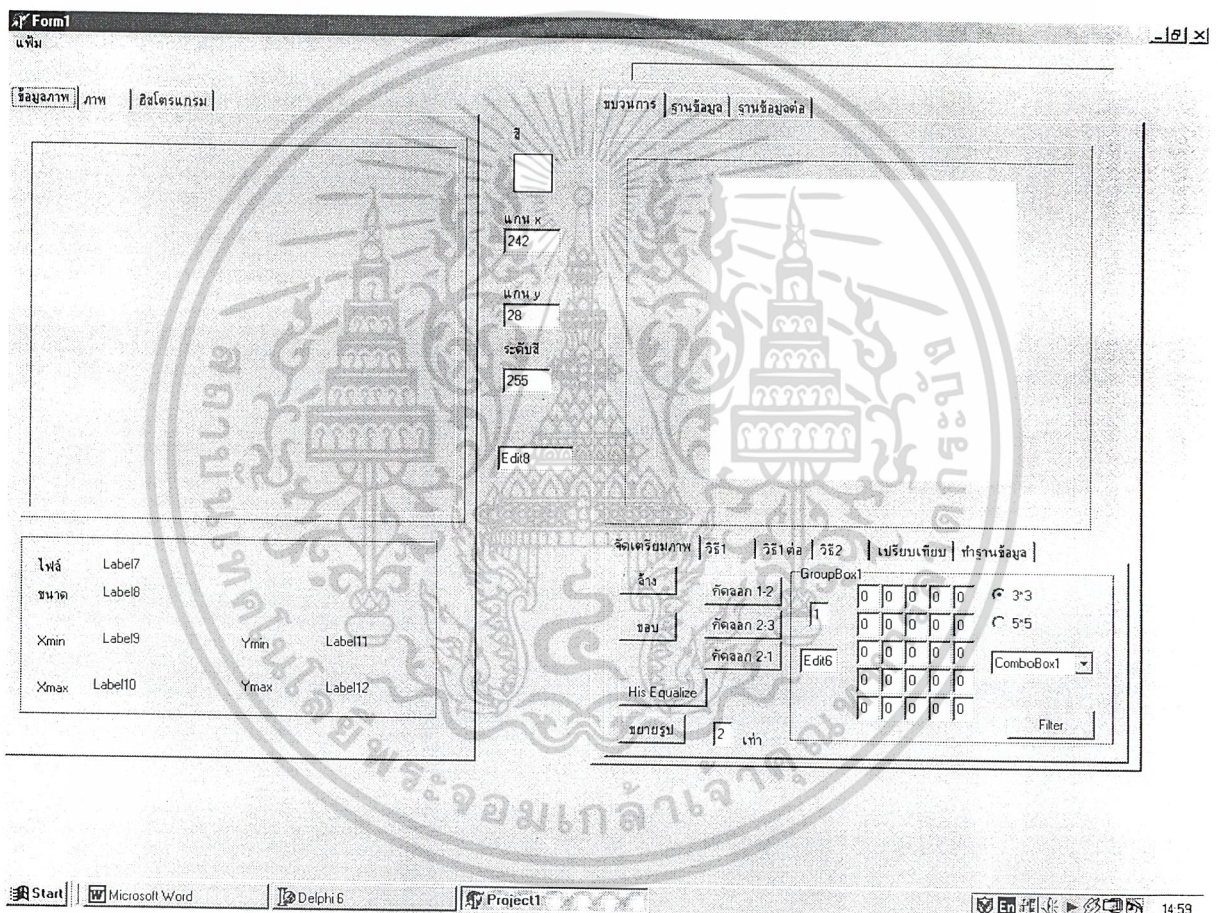
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปและวิจารณ์ผลการทดลอง

5.1 ผลการทดลอง

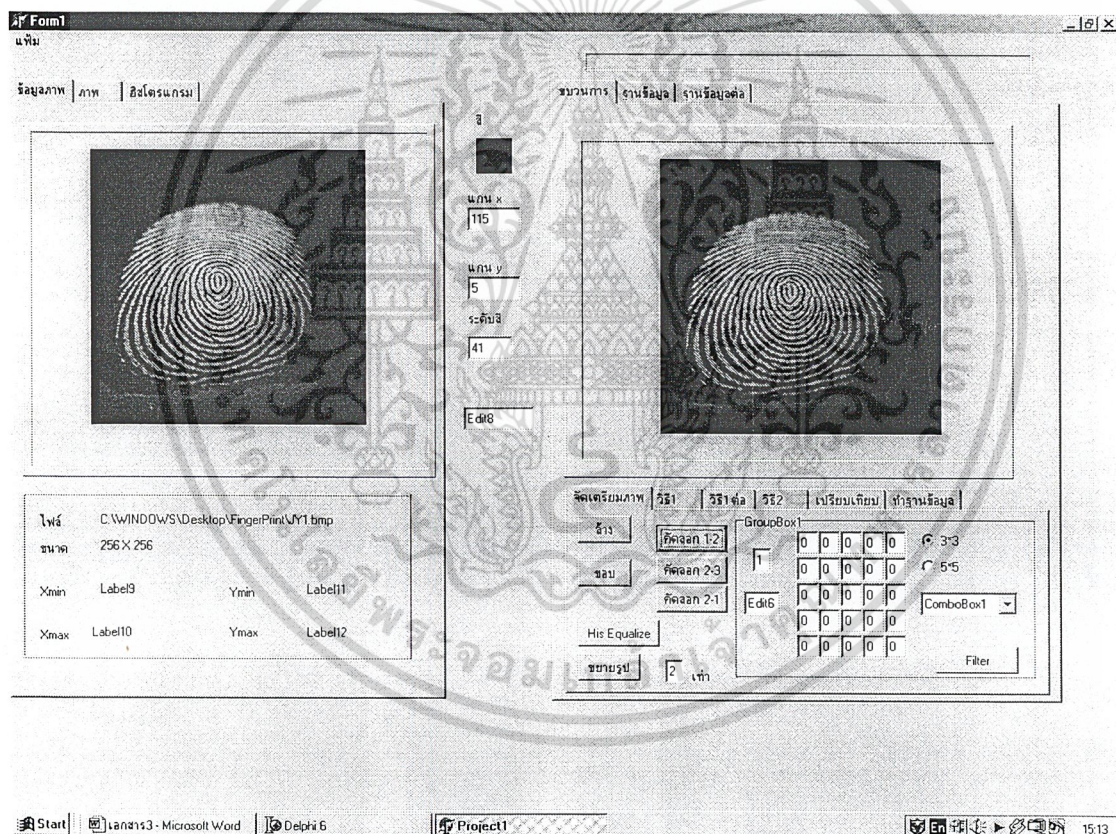
จากการที่ได้เขียน โปรแกรมตามขั้นตอนวิธีต่าง ๆ ที่ได้เรียนมาตามทฤษฎีนั้น ผลที่ได้จากการตรวจสอบและเปรียบเทียบของบุคคล A จะมีลำดับการตรวจสอบ ดังนี้



รูปที่ 40 แสดงหน้าต่างของ โปรแกรมการตรวจสอบและเปรียบเทียบลายนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

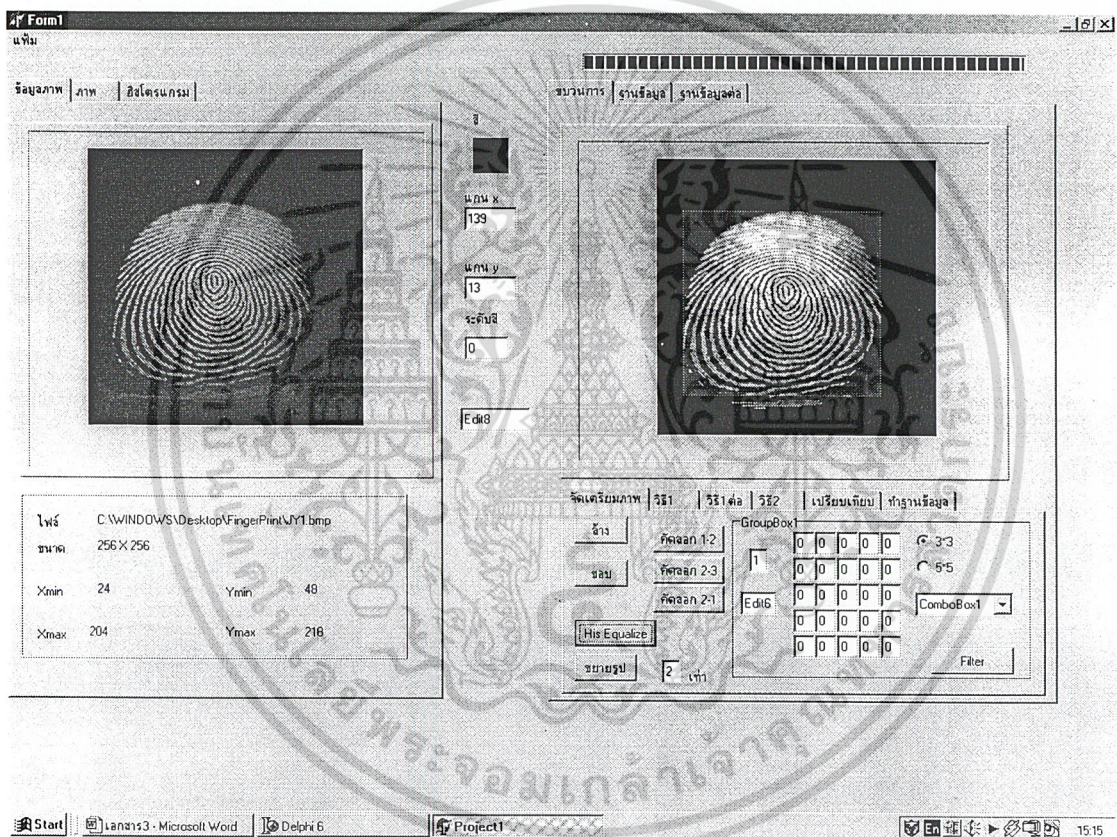
ในภาพที่ 41 นี้ เป็นการเปิดเพิ่มเพื่อนำภาพลายนิ้วมือเข้ามา (จะเห็นว่า ภาพลายนิ้วมือที่นำเข้ามามีลักษณะเป็นแบบก้นหอย) ภาพที่ปรากฏจะอยู่ทางซ้ายมือ (Image 1) จากนั้น ไปคลิกปุ่มคัดลอก 1 – 2 ซึ่งภาพที่ปรากฏจะไปปรากฏทางขวามือ (Image 2) เราจะกระทำภาพตามขั้นตอนต่าง ๆ ตามทฤษฎีในภาพทางขวามือ เพื่อเป็นการเปรียบเทียบกับภาพที่นำเข้ามาที่ปรากฏทางซ้ายมือ ที่ช่องของแกน x และ แกน y จะบอกถึงค่าของแกน x และ แกน y เมื่อเราทำการเลื่อนเมาท์ไปตามส่วนต่าง ๆ ของภาพ



รูปที่ 41 แสดงภาพที่นำมาเปรียบเทียบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

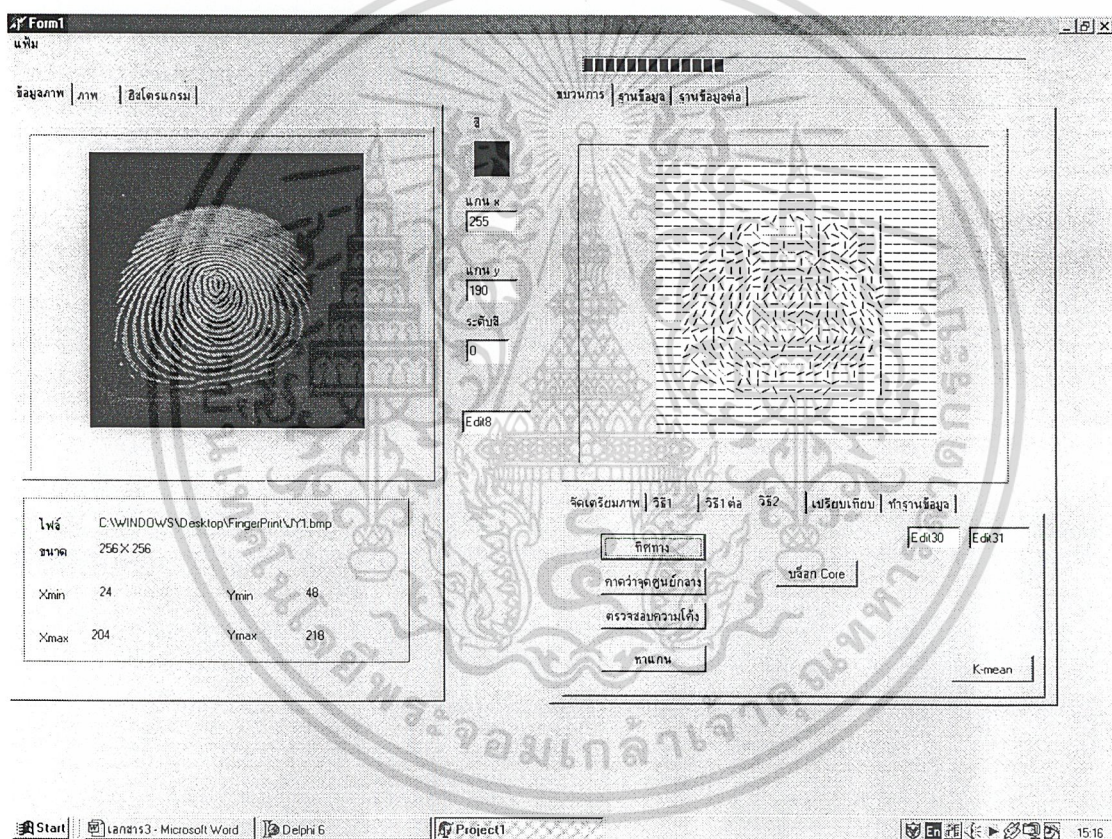
ในภาพที่ 42 นี้ เป็นขั้นตอนการทำ Histogram Equalization ของภาพลายนิ้วมือที่นำเข้ามา เพื่อให้ค่าระดับเทามีการกระจายของภาพให้มีค่าตั้งแต่ 0 – 255 ระดับประโยชน์ของการทำ Histogram Equalization นี้ คือ ภาพลายนิ้วมือที่นำเข้ามาที่มีลักษณะค่อนข้างมืดจะทำให้ภาพมีลักษณะสว่างขึ้น หรือ ภาพที่มีลักษณะค่อนข้างสว่างจะทำให้ภาพลดความสว่างลงมีความนุ่มนวลเพิ่มขึ้น



รูปที่ 42 การทำ Histogram Equalization

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

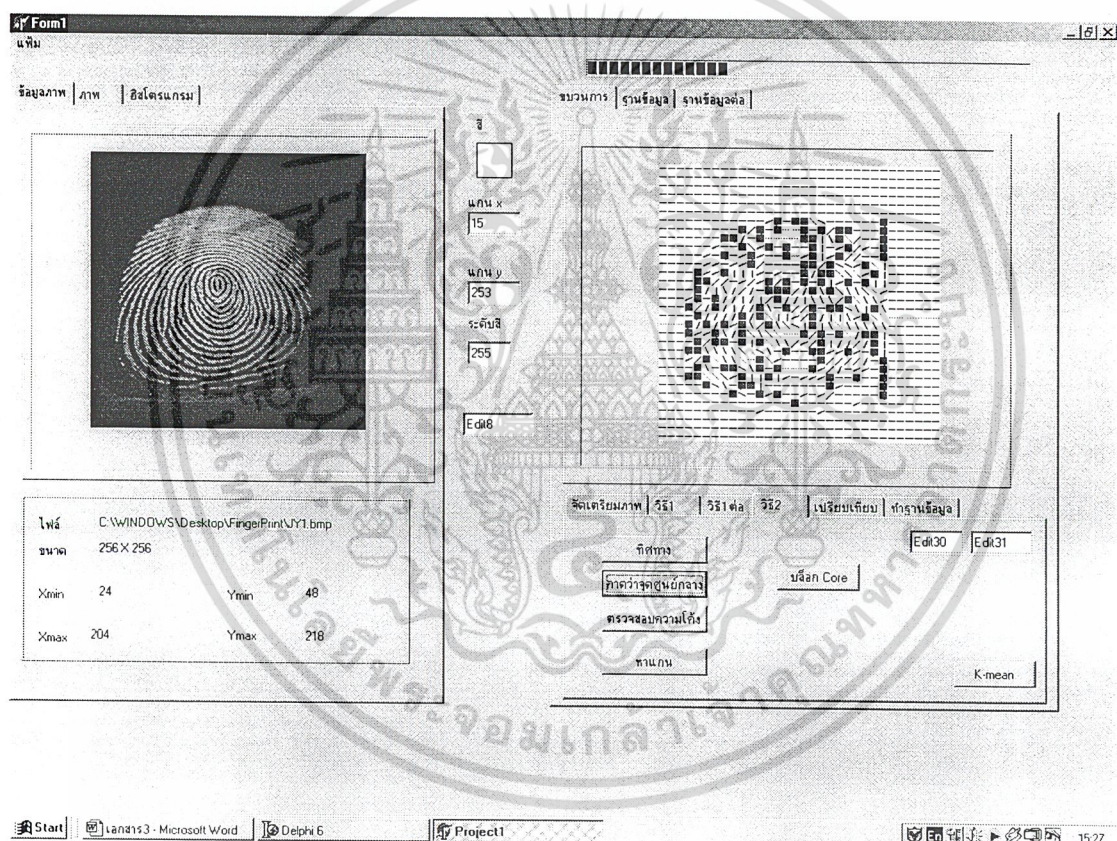
ในภาพที่ 43 นี้ เป็นการหาลักษณะทิศทางของภาพ เมื่อภาพที่ผ่านการทำ Histogram Equalization แล้ว มาคลิกปุ่ม “ทิศทาง” ซึ่งอยู่ในหน้าต่างของวิธี 2 ในโปรแกรม โดยเราจะแบ่งทิศทางของภาพเป็น 8 ทิศทาง ซึ่งเป็นค่าที่เหมาะสม หากเราทำการแบ่งทิศทางของภาพให้มีค่ามาก หรือ น้อยเกินไปจะทำให้ผลที่ได้คือออกมาเกิดการผิดพลาดเกิดขึ้น (ภาพทิศทางที่ได้ไม่ตรงกับลายนิ้วมือ) และมีการประมวลผลที่นานมากขึ้น



รูปที่ 43 แสดงทิศทางของภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

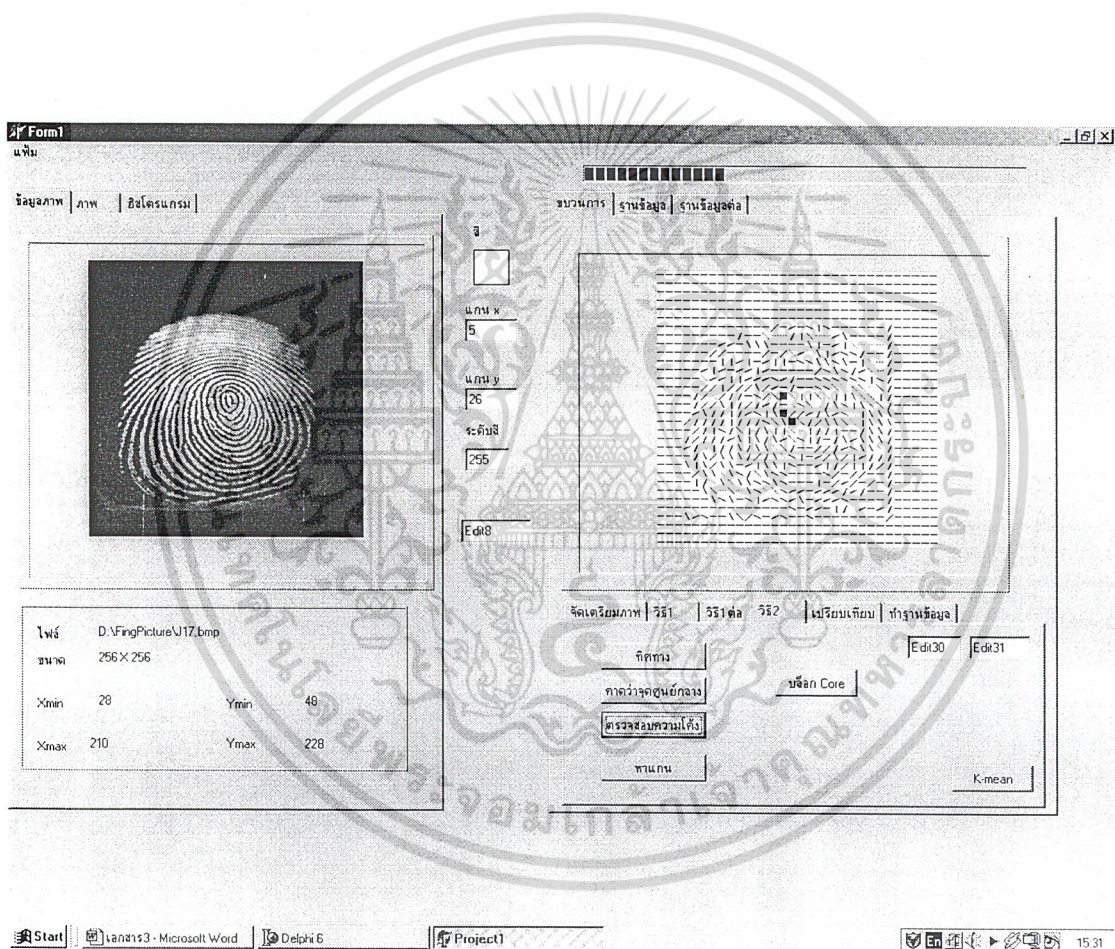
ในรูปที่ 44 นี้ เป็นการหาจุดที่คาดว่าจะจะเป็นจุด core โดยนำภาพที่ผ่านการหาทิศทางของภาพมาทำการหาจุดที่คาดว่าจะจะเป็นจุด core วิธีนี้เราทำการแสดกนไปที่ละบล็อกจากซ้ายไปขวา โดยนำค่าของทิศทางที่อยู่ในบล็อกที่เราพิจารณา มาลบกับค่าทิศทางของบล็อกที่อยู่ทางด้านซ้าย (บล็อกที่ผ่านมา) ให้มีค่ามากกว่าหรือเท่ากับ 2 ผลที่ได้ตามรูปข้างล่างนี้ (คลิกปุ่ม “ คาดว่าจุดศูนย์กลาง ”)



รูปที่ 44 แสดงการหาจุดที่คาดว่าจะจะเป็นจุด core

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

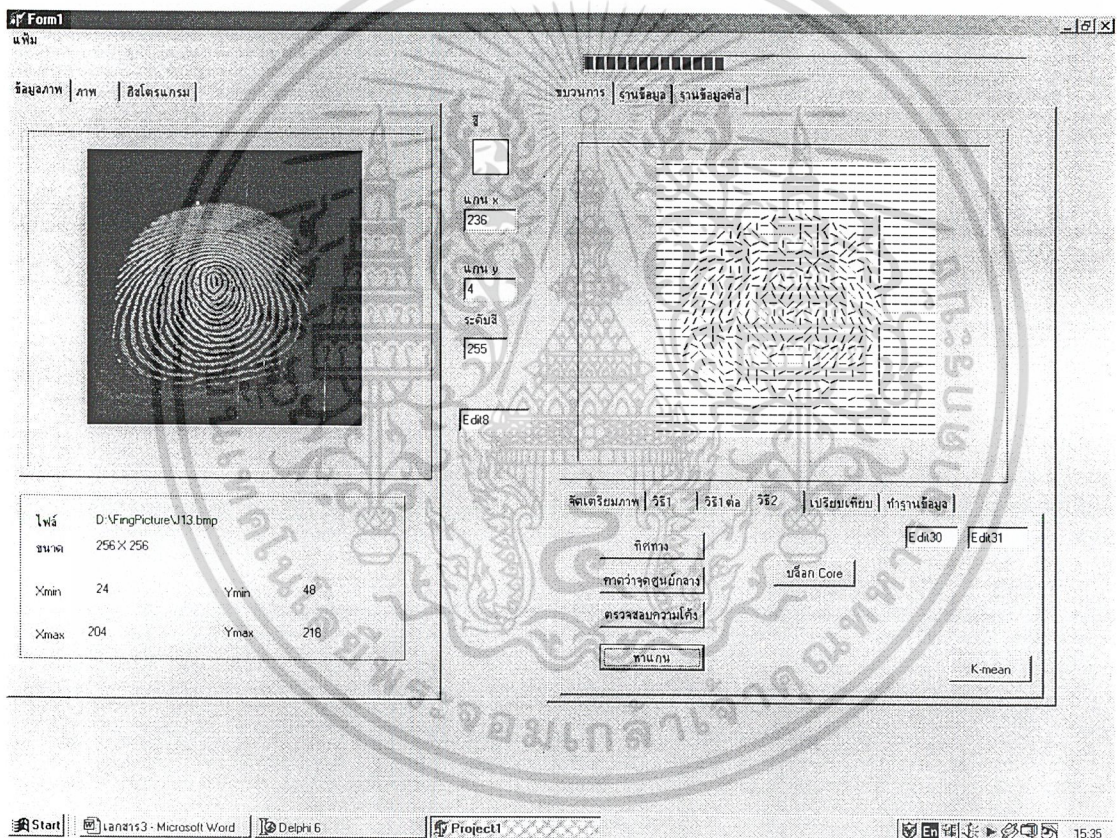
รูปที่ 45 นี้ เราจะนำภาพที่ผ่านการหาจุดที่คาดว่าจะจะเป็นจุด CORE มาตรวจสอบส่วนโค้ง ซึ่งเราจะให้ส่วนโค้งที่มีการโค้งขึ้นผ่านได้เท่านั้น และส่วนโค้งที่มีการโค้งลงจะไม่ให้ผ่าน เช่น ในภาพลายนิ้วมือข้างล่างนี้จะมีลักษณะเป็นแบบก้นหอย จะมีทั้งส่วนโค้งที่มีการโค้งขึ้นและโค้งลง โดยเราจะให้ส่วนโค้งที่มีการโค้งขึ้นผ่านไปได้ ส่วนโค้งที่มีลักษณะโค้งลงเราจะไม่นำมาพิจารณา เพราะจากการที่เราดูรูปลายนิ้วมือ ส่วนโค้งที่มีลักษณะโค้งลงไม่ใช่จุดศูนย์กลางของลายนิ้วมือ (คลิกปุ่ม “ตรวจสอบส่วนโค้ง”)



รูปที่ 45 แสดงการตรวจสอบส่วนโค้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

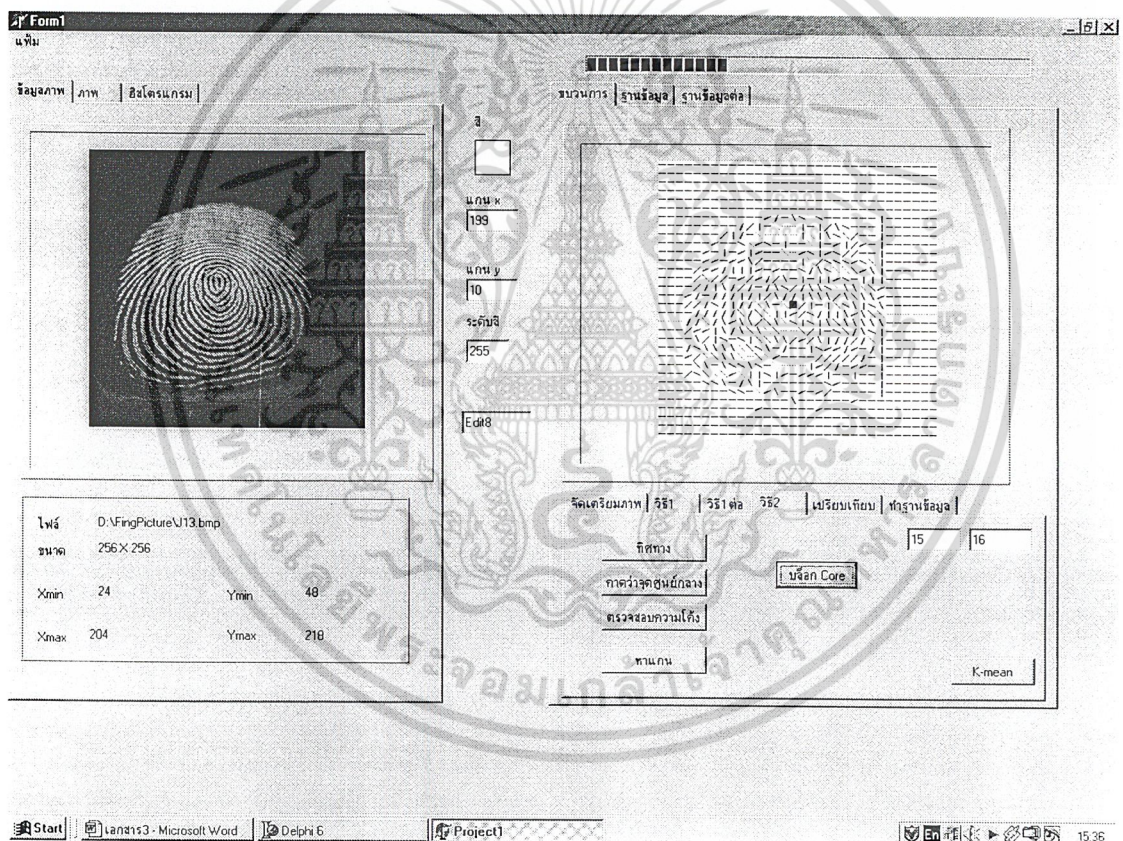
ในรูปที่ 46 นี้ เป็นการหาแกนของภาพ โดยนำภาพที่ผ่านการตรวจสอบส่วนโค้งมาหาแกนของภาพ จุดประสงค์ของขั้นตอนนี้ก็คือ การหาจุดศูนย์กลางของภาพให้เป็นจุดศูนย์กลางเดิม เมื่อภาพลายนิ้วมือที่นำเข้ามามีลักษณะเอนเอียงไป เพราะเมื่อภาพมีลักษณะเอนเอียงจะทำให้จุดศูนย์กลางของภาพมีการเอนเอียงไปด้วย หากไม่มีขั้นตอนนี้จะทำให้การหาจุดศูนย์กลางของภาพที่ได้มีการผิดพลาด (ไม่ใช่จุดศูนย์กลางของภาพที่แท้จริง)



รูปที่ 46 แสดงการหาแกนของภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

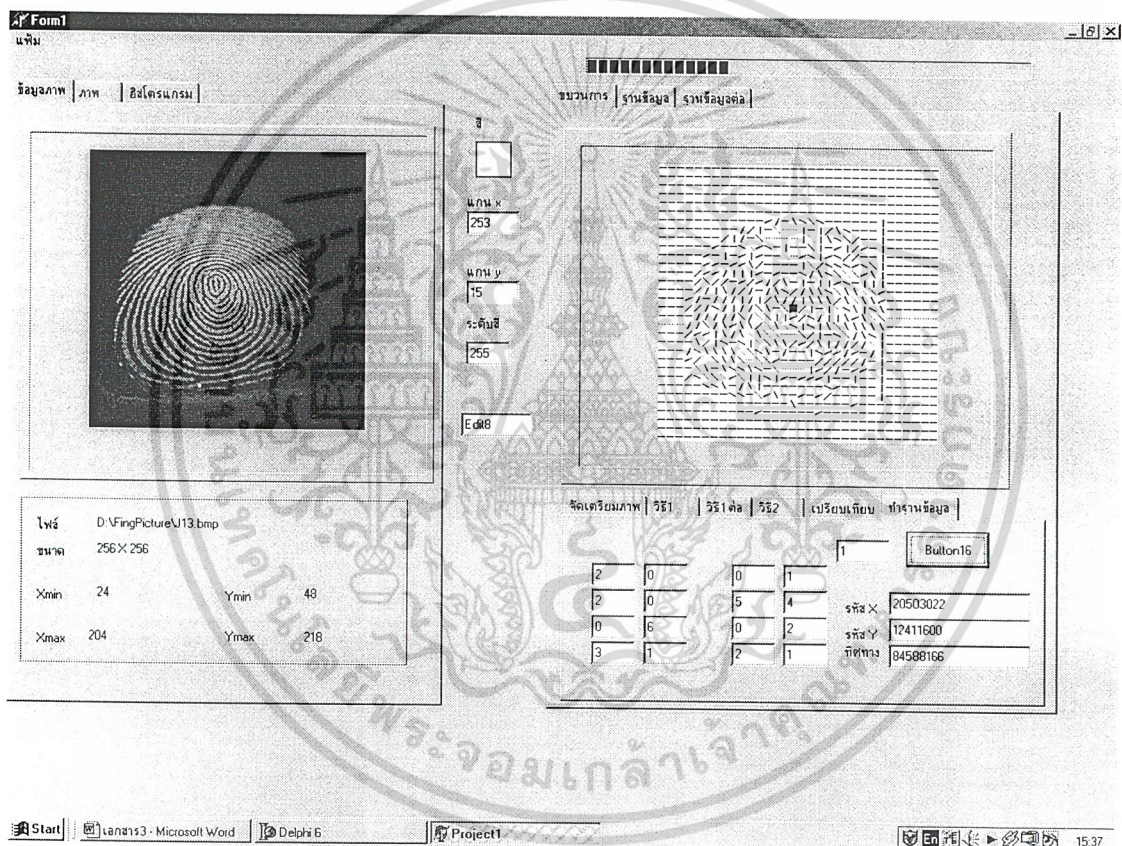
ในรูปที่ 47 นี้ เป็นการหาจุดศูนย์กลางของภาพ โดยนำภาพที่ผ่านการหาแกนของภาพมาทำการจัดกลุ่มโดยใช้ทฤษฎี $K - Mean$ ซึ่งภาพที่ผ่านการหาแกนของภาพจะมีลักษณะเป็นกลุ่ม ๆ เราจะจัดบล็อกของภาพเหล่านั้นให้เป็นกลุ่ม ๆ ในระยะที่กำหนด หากพ้นระยะที่กำหนดให้ไปอยู่ในกลุ่มอื่น ถ้าไม่มีกลุ่มให้อยู่ก็ให้เป็นจุดศูนย์กลางของกลุ่มตัวใหม่ เมื่อเราจัดบล็อกเหล่านั้นให้เป็นกลุ่ม ๆ ได้แล้วก็ทำการหาจุดศูนย์กลางของกลุ่ม สุดท้ายเมื่อเราได้จุดศูนย์กลางของกลุ่มเราก็หาความน่าจะเป็นที่มากที่สุดของแต่ละจุดศูนย์กลางนั้น ให้เป็นบล็อกของจุด CORE (จุดศูนย์กลาง)



รูปที่ 47 การหาจุดศูนย์กลางของภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

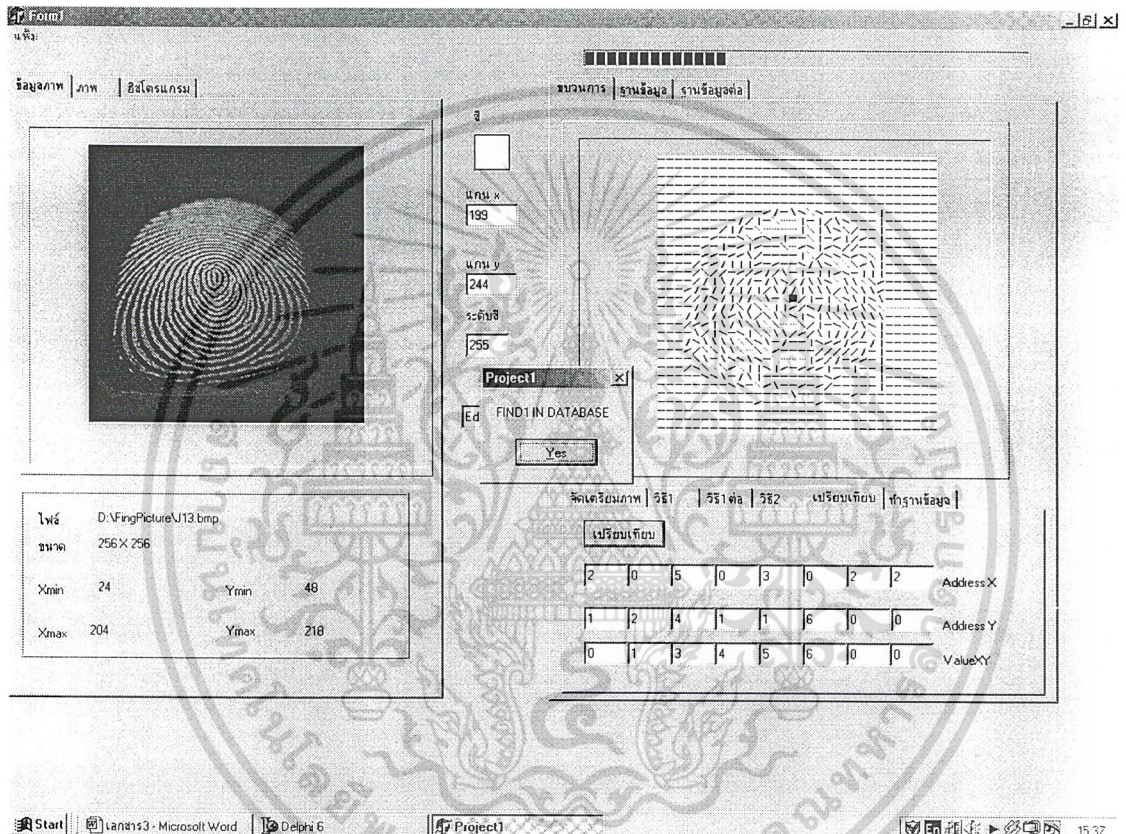
รูปที่ 48 นี้ เป็นการทำฐานข้อมูล เมื่อเราได้จุดศูนย์กลางของภาพแล้ว เราก็จะนำภาพนั้นไปทำฐานข้อมูล โดยเราจะใช้ทฤษฎีของนิเวรอนมาทำการหาค่าของบล็อกทิศทาง เราจะทำการตัดบล็อกขนาด 7×7 ที่ล้อมรอบจุดศูนย์กลาง แล้วให้นิเวรอนแต่ละตัว (มีทั้งหมด 50 ตัว) โดยแต่ละตัวจะมี 8 เซลล์ (8 ขา หรือ 8 ตำแหน่ง) ไปทำการจับค่าที่อยู่ในบล็อกนั้น แล้วนำค่านั้นไปเก็บในตารางที่อยู่ในฐานข้อมูล (คลิกปุ่มทำฐานข้อมูล)



รูปที่ 48 แสดงการทำฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 49 นี้ แสดงการเปรียบเทียบและตรวจสอบลายนิ้วมือ เมื่อเรานำลายนิ้วมือที่เข้ามาทำการเปรียบเทียบผ่านขั้นตอนต่าง ๆ แล้ว เราก็จะทำการเปรียบเทียบลายนิ้วมือนี้ว่ามีอยู่ในฐานข้อมูลหรือไม่ หากพบว่ามียู่ในฐานข้อมูลก็จะขึ้นข้อความว่า FIND IN DATABASE หากไม่พบในฐานข้อมูลก็จะขึ้นข้อความว่า NO IN DATABASE



รูปที่ 49 แสดงการเปรียบเทียบและตรวจสอบลายนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 50 นี้ เป็นการตรวจสอบในฐานข้อมูลว่าลายนิ้วมือที่นำมาเปรียบเทียบเป็นของบุคคลใด หากพบว่ามียู่ในฐานข้อมูลก็จะแสดงข้อมูลของบุคคลนั้นขึ้นมา



รูปที่ 50 แสดงการตรวจพบในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 สรุปและวิจารณ์ผลการทดลอง

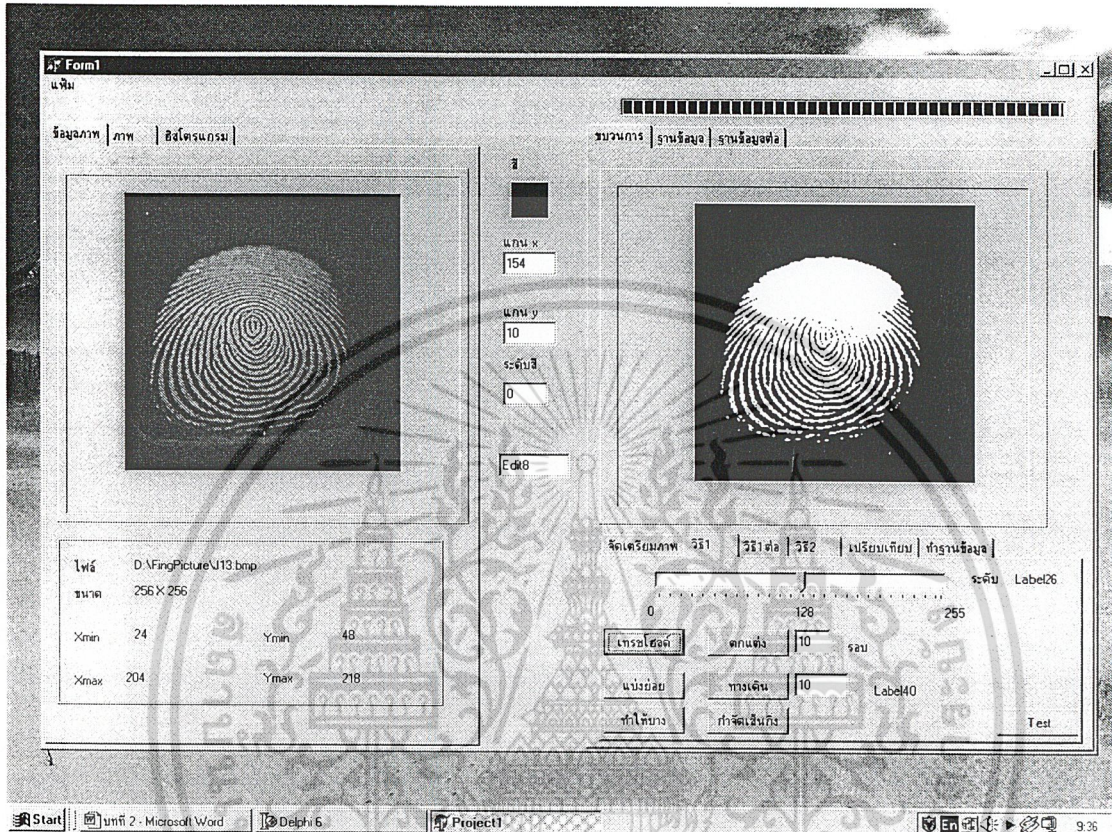
ระบบของการเปรียบเทียบและตรวจสอบลายนิ้วมือนี้ สามารถนำไปประยุกต์ใช้ในงานต่างๆ ได้มากมาย เช่น ระบบรักษาความปลอดภัย , ใช้พิสูจน์ลายนิ้วมือ เป็นต้น ในการทำมีขั้นตอนต่างๆ ดังนี้ ขั้นแรก นำภาพลายนิ้วมือจากกล้องมาทำฮิสโตแกรมฮิวอิสเชียน จากนั้นนำมาหาทิศทางของภาพ , ตรวจสอบส่วนโค้ง , หาแกนของภาพ , การจัดกลุ่มโดยใช้ทฤษฎีของ k-mean และสุดท้าย การหาจุดศูนย์กลางของภาพ ขั้นตอนเหล่านี้ที่ผ่านมาเป็น การเตรียมภาพเพื่อที่จะนำไปสู่ขั้นตอนต่อไป จากนั้นเราก็มาเลือกที่จะนำภาพลายนิ้วมือที่ได้มา ไปทำฐานข้อมูลหรือว่าจะนำไปเปรียบเทียบกับภาพลายนิ้วมือที่มีอยู่ในฐานข้อมูล

5.2.1 ปัญหาที่เกิดขึ้น

- อย่างแรกก็คือ ผู้จัดทำไม่มีประสบการณ์ในการเขียน โปรแกรมของเดสก์ทอปมาก่อน ทำให้ต้องเสียเวลามากในการทำ ความเข้าใจและศึกษาค้นคว้า
- ไม่มีความรู้ทางด้าน Image Processing มาก่อนทำให้เสียเวลาการศึกษา ค้นคว้า อยู่พอสมควร และข้อมูลที่ได้มาเป็นภาษาอังกฤษทำให้ใช้เวลานานขึ้นไปอีก
- ผู้จัดทำได้ทำการของขั้นตอนต่างๆ ทาง Image Processing มาพอสมควรแต่ก็ถึงทางตัน ของการหาจุดเด่น ซึ่งอุปสรรคตามขั้นตอนต่างๆ วิธีนี้มีดังนี้

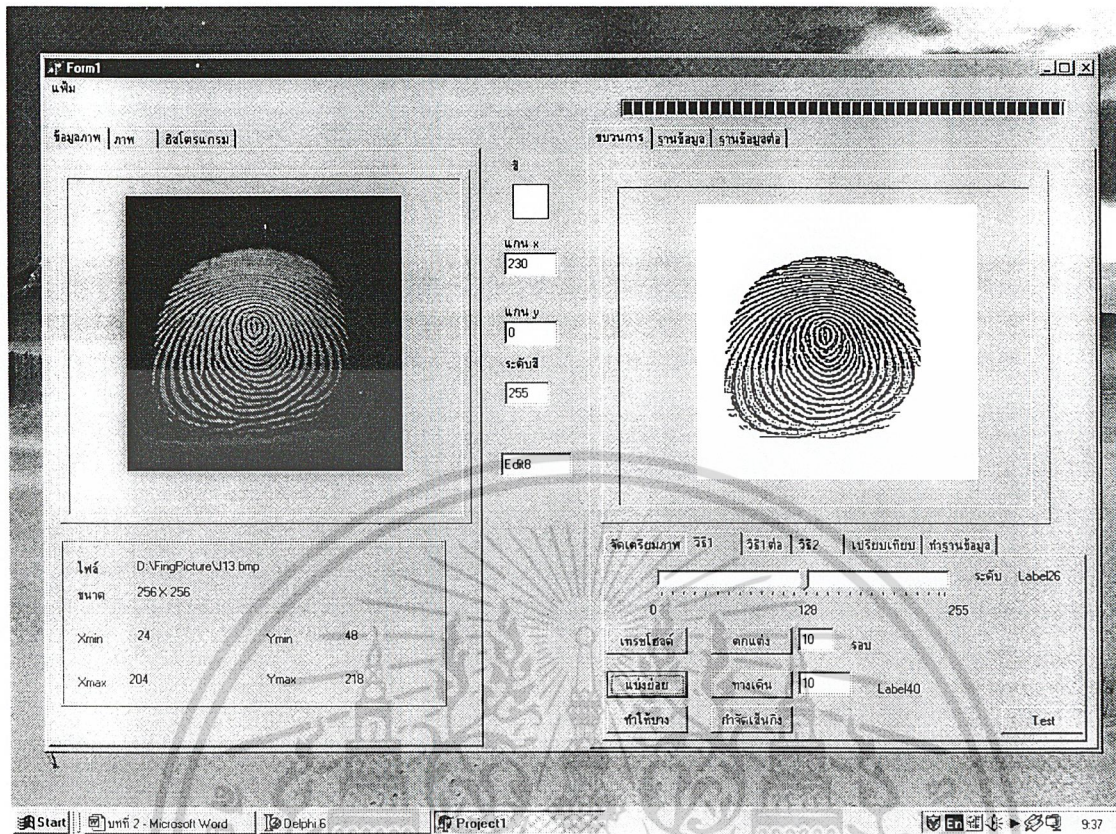
(ก) นำภาพลายนิ้วมือที่ได้มาทำการ smooth และทำ Threshold ของภาพลายนิ้วมือ ซึ่งการหาค่าของเทรชโฮลด์นั้นใช้ค่า 128 เลยไม่ได้ (อยู่ตรงกลางระหว่าง 0 – 255) เพราะจะทำให้ภาพที่ผ่านการทำเทรชโฮลด์ด้วยวิธีนี้ลายนิ้วมือบางส่วนจะขาดหายไป (ซึ่งเป็นส่วนที่สำคัญ) เราจะใช้วิธีการทำบล็อกรูปขนาด 5×5 ล้อมรอบจุดพิกเซลที่เราพิจารณาอยู่ แล้วค่าเทรชโฮลด์ที่อยู่ในบล็อกรูปนั้น ทำให้ทั่วทั้งภาพ เราก็จะได้ภาพที่ผ่านการทำเทรชโฮลด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 51 แสดงการทำ Threshold แบบปกติ

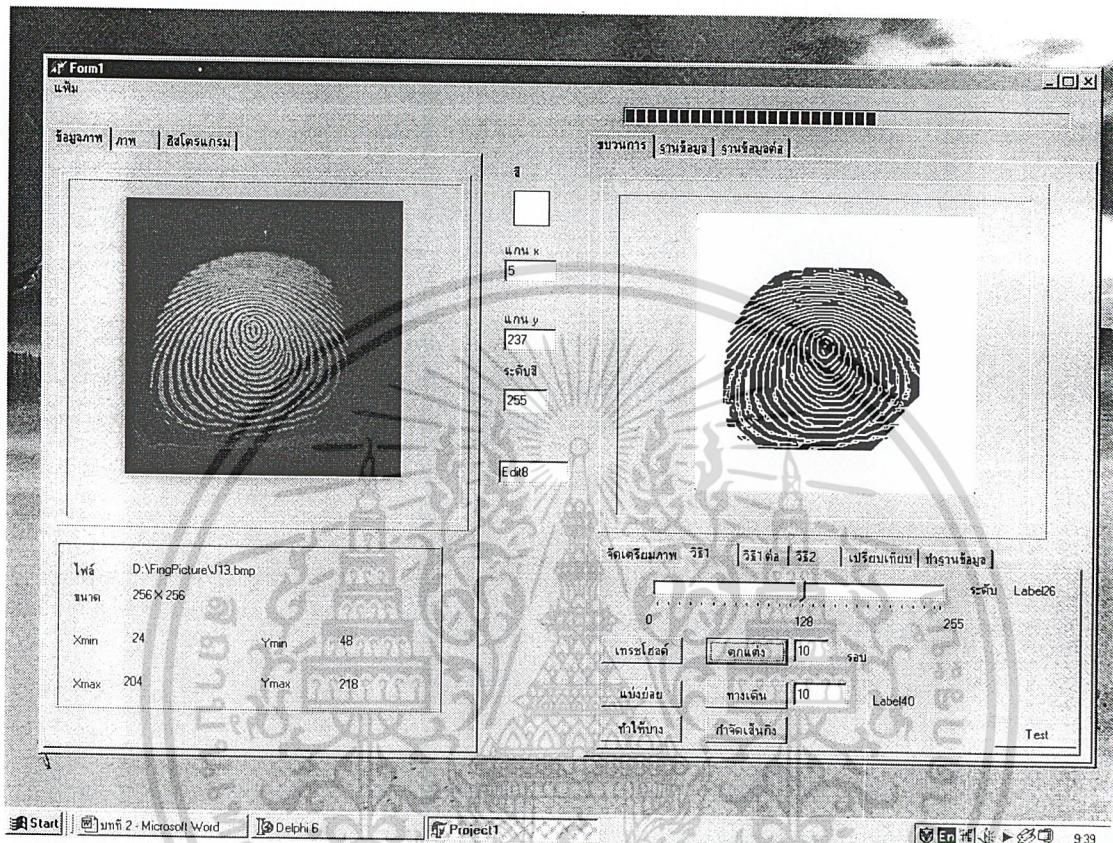
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 52 แสดงการทำ Threshold แบบ บล็อก 5×5

(ข) การตกแต่งรูป เนื่องจากภาพที่ได้เมื่อนำไปทำขั้นตอนการทำ Thinning เลยนั้น จะมี noise เข้าไปรบกวนการทำให้บางได้ เพราะว่าการทำให้บางจะทำให้ภาพลายนิ้วมือที่มีขนาดใหญ่กว่า 1 พิกเซล ให้เหลือเพียง 1 พิกเซล การตกแต่งรูปก็คือ การกำจัด noise นั้นเอง

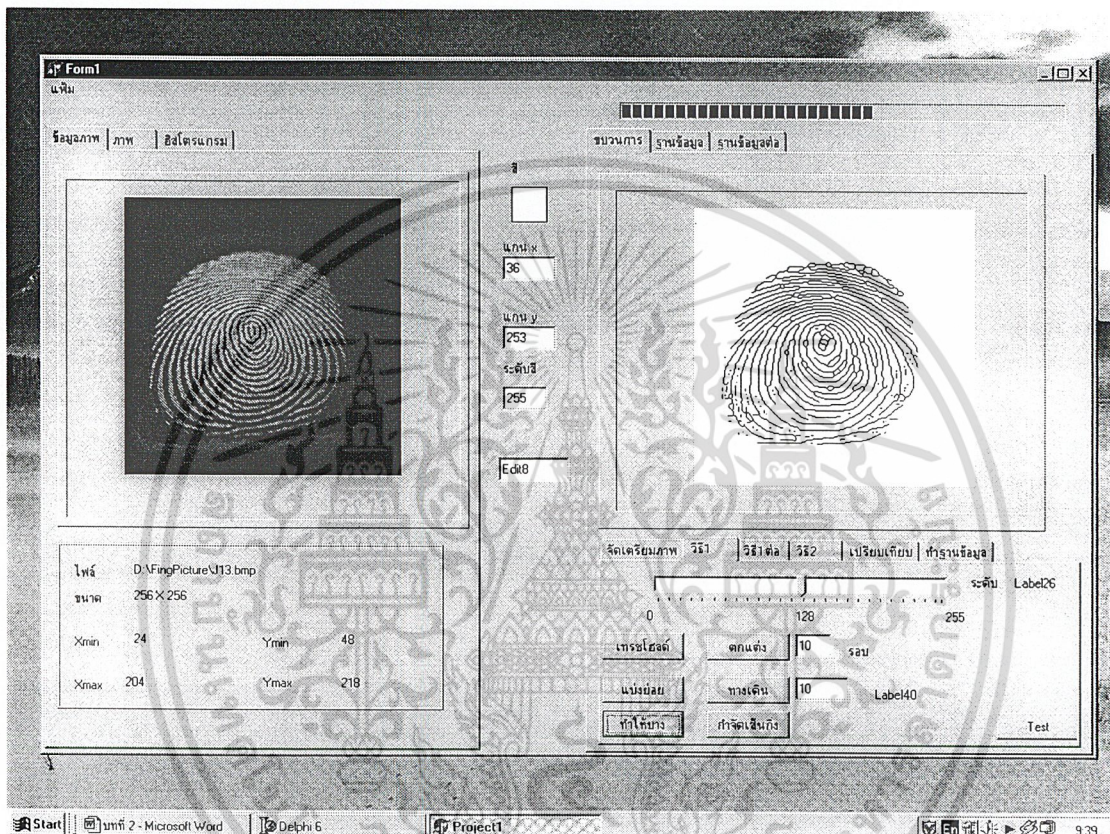
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 53 แสดงการตกแต่งรูป

(ค) การทำให้บาง (Thinning) จะทำให้ภาพลายนิ้วมือที่มีขนาดใหญ่กว่า 1 พิกเซล ให้เหลือเพียง 1 พิกเซล เพราะง่ายต่อการนำไปประมวลในขั้นตอนต่างๆ หรือการเก็บข้อมูล เพราะว่าจะมีข้อมูลเพียงเล็กน้อยเท่านั้น ซึ่งวิธีที่ใช้นั้นจะใช้วิธีของ ขาง และ ซวาล เป็นวิธีที่นิยมกันมากที่สุด

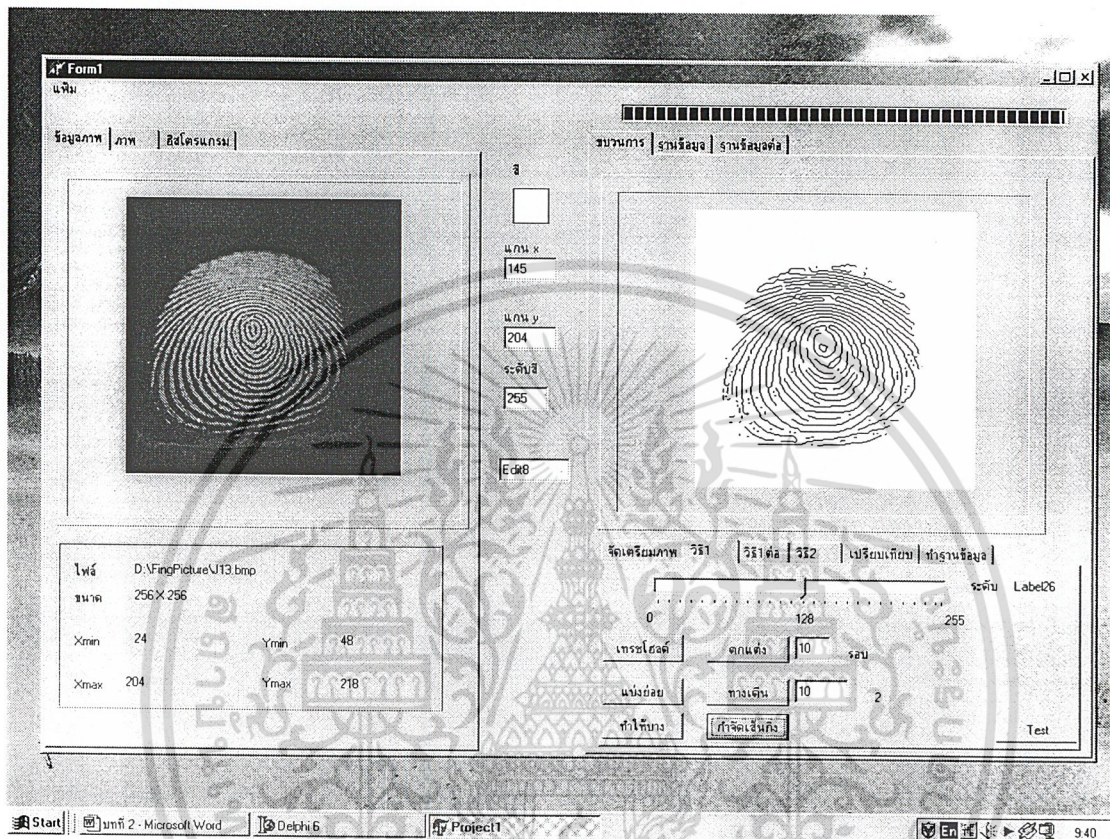
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 54 แสดงการทำให้บาง

(ง) การกำจัดเส้นกึ่ง ภาพที่ผ่านการทำให้บางจะสังเกตเห็นว่าผลที่ได้ บางเส้นจะมี loop เกิดขึ้น หรือ บางเส้นจะมีเส้นกึ่งอยู่ (มีลักษณะแบบ 3 แยก) จากการที่เราทำการกำจัดเส้นกึ่งนั้น ผลที่ได้จะกำจัดเส้นที่สำคัญๆ ไปด้วย ทำให้มีความผิดพลาดเกิดขึ้น ซึ่งผลที่ได้เป็นเช่นนั้น อาจเกิดจากการผิดพลาดของตัวทฤษฎี หรือ อาจมาจากการเขียนโปรแกรม

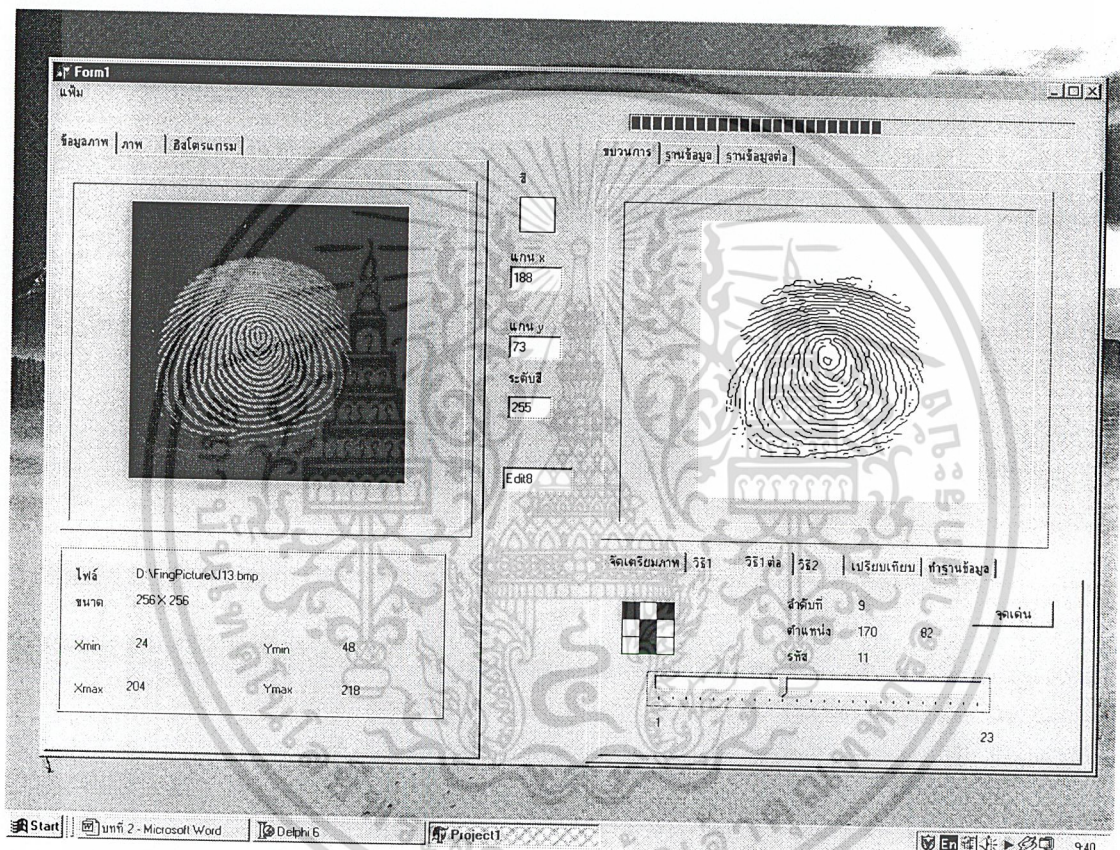
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 55 แสดงการกำจัดเส้นกึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ง) การหาจุดเด่น เป็นการหาจุดเด่นของภาพซึ่งก็คือทางแยกนั่นเอง ผลของการหาจุดเด่น จะได้ค่าที่ไม่เท่ากันเลยสักครั้ง เลยทำให้ต้องยกเลิกวิธีนี้



รูปที่ 56 แสดงการหาจุดเด่นของภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ในการทำวิธีใหม่นี้จะไม่ค่อยพบปัญหา แต่จะพบปัญหาตรงที่การเขียนโปรแกรมจะเป็นส่วนใหญ่

5.2.2 ข้อเสนอแนะ

- ผู้ที่สนใจในการทำ เรื่องนี้ ควรมีความรู้พื้นฐานทางด้าน **Image Processing** มาบ้างพอสมควร

- ควรมีความสามารถทางด้านเขียนโปรแกรมมาพอสมควร จะได้ไม่ต้องเสียเวลาการเรียนรู้ และเสียเวลาฝึกเขียน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

unit Unit1;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, Menus, ComCtrls, ExtCtrls, StdCtrls, ExtDlgs, TeEngine, Series,
TeeProcs, Chart, DB, DBTables, DBCtrls, Mask, Grids, DBGrids;

type

TForm1 = class(TForm)

PageControl1: TPageControl;

TabSheet1: TTabSheet;

TabSheet2: TTabSheet;

TabSheet3: TTabSheet;

Panel1: TPanel;

Panel2: TPanel;

Panel3: TPanel;

Panel4: TPanel;

PageControl2: TPageControl;

TabSheet4: TTabSheet;

TabSheet5: TTabSheet;

Panel5: TPanel;

Panel6: TPanel;

ProgressBar1: TProgressBar;

MainMenu1: TMainMenu;

N1: TMenuItem;

Exit: TMenuItem;

Panel7: TPanel;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label5: TLabel;

Label4: TLabel;

Label6: TLabel;

Label7: TLabel;

Label8: TLabel;

Label9: TLabel;

Label10: TLabel;

Label11: TLabel;

Label12: TLabel;

safe: TMenuItem;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่อาจถือลิขสิทธิ์อื่น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Edit30: TEdit;
Edit31: TEdit;
Core: TButton;
FindKan: TButton;
CheckCurve: TButton;
Button5: TButton;
S_a: TShape;
S_b: TShape;
S_c: TShape;
S_d: TShape;
Shape6: TShape;
S_e: TShape;
S_f: TShape;
S_g: TShape;
S_h: TShape;
Label41: TLabel;
Label42: TLabel;
Label43: TLabel;
Label44: TLabel;
Label45: TLabel;
Label46: TLabel;
Label47: TLabel;
Label48: TLabel;
Label49: TLabel;
Button17: TButton;
TrackBar2: TTrackBar;
procedure OpenClick(Sender: TObject);
```

```
procedure ExitClick(Sender: TObject);
procedure safeClick(Sender: TObject);
procedure CleanColorClick(Sender: TObject);
procedure Image2OnMouseMove(Sender: TObject; Shift: TShiftState; X,
Y: Integer);
procedure Button2Click(Sender: TObject);
```

```
procedure Button6Click(Sender: TObject);
procedure ZoomClick(Sender: TObject);
procedure HistrogramClick(Sender: TObject);
procedure ComboBox1Change(Sender: TObject);
procedure FilterClick(Sender: TObject);
```

```
procedure Button7Click(Sender: TObject);
procedure HistrogramEqualizationClick(Sender: TObject);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ก่อกรรมใดๆ ให้เสียชื่อเสียง หรืออำนาจหน้าที่แต่แปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure CoreClick(Sender: TObject);
procedure Button14Click(Sender: TObject);
procedure CheckCurveClick(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure TrackBar1Change(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure TokTangClick(Sender: TObject);
procedure ThinningClick(Sender: TObject);
procedure Button9Click(Sender: TObject);
procedure Button4Click(Sender: TObject);
procedure FindKanClick(Sender: TObject);
procedure Button5Click(Sender: TObject);
procedure Button8Click(Sender: TObject);
procedure DBImage1Click(Sender: TObject);
procedure Button15Click(Sender: TObject);
procedure Button16Click(Sender: TObject);
procedure DBImage2Click(Sender: TObject);
procedure Button10Click(Sender: TObject);
procedure Button12Click(Sender: TObject);
procedure Button17Click(Sender: TObject);
procedure TrackBar2Change(Sender: TObject);

```

```

private
  { Private declarations }
public
  { Public declarations }
end;

```

```
var
```

```

Form1: TForm1;
xy : array[0..1,0..1] of integer;
nGrayValue,pixels,Th,yakarray,bidarray,dudarray,elarray : array[0..256,0..256] of integer;
KeepDir : array[0..32,0..32] of byte;
xmi,ymi,xma,yma,cx,cy : byte;
memx : array[0..32,0..32] of byte;
lineF : array[0..32] of byte;

```

```
New1C,New2C,New3C,New4C,New5C,New6C : array[0..31,0..31] of byte;
```

```
New7C,New8C,New9C,New10C,New11C,New12C : array[0..31,0..31] of byte;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FreqM1,FreqM2,FreqM3,FreqM4,FreqM5,FreqM6,FreqM7,FreqM8,FreqM9,FreqM10,FreqM11,FreqM12 : array[0..8] of byte;

Z1,Z2,Z3,Z4,Z5,Z6,Z7,Z8,Z9,Z10,Z11,Z12 : array[0..8]of byte;

////////////////////////////////////

////////// random test 8 point

PointX : array[0..7]of Integer; //// x

PointY : array[0..7]of Integer; //// y

////////// box center of 0-31,0-31 big box

CenterX,CenterY : byte;

////////// box 7x7 of image process from centerbox

SmallBoxXY: array[0..6,0..6]of Integer;

////////// branch,cross 200

BC_X:array[0..1000]of integer; // x

BC_Y:array[0..1000]of integer; // y

BC_Code:array[0..1000]of integer; // code

code,count_BC:integer;

function checkBC(x,y:integer):Boolean;

implementation

uses unit2,unit3,unit4;

{SR *.dfm}

procedure TForm1.OpenClick(Sender: TObject);

var current:shortstring;

begin

if openpictorialog1.Execute then

begin

current:=openpictorialog1.FileName;

Label7.Caption:=current; // show filename

image1.picture.loadfromfile(current);

Label8.Caption:=IntToStr(image1.Width)+' X '+IntToStr(image1.Height);

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่าจะสื่อในรูปแบบใดก็ตาม อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

edit1.Text:='Idle';

```

end;
end;
////////////////////////////////////////////////////////////////
procedure TForm1.ExitClick(Sender: TObject);
begin
    close;
end;

```

```

////////////////////////////////////////////////////////////////
procedure TForm1.safeClick(Sender: TObject);
begin
    if SavepictureDialog1.Execute then
        begin
            image2.Picture.SaveToFile(savepicturedialog1.FileName);
        end;
end;

```

```

////////////////////////////////////////////////////////////////
procedure TForm1.CleanColorClick(Sender: TObject);
var x,y,nPixelColor,nRed,nGreen,nBlue,nGray : integer;
    new,line1,line2 :integer; // count line
    pf:array[0..300] of integer; // address,first
    ps:array[0..300] of integer; // address secound
    p:array[0..300] of integer; // address status line 0=black ,1 =have white pixel
begin
    new:=0; // count line at white pixel

```

```

//////////////////////////////////////////////////////////////// mem point error
for y := 0 to image2.Height-1 do
begin
    pf[y]:=0;
    ps[y]:=ps[y];
    new:=0;
    for x := 0 to image2.Width-1 do

begin
        nPixelColor := image2.Canvas.Pixels[x,y];
        nRed := GetRValue(nPixelcolor);
        nGreen := GetGValue(nPixelcolor);
        nBlue := GetBValue(nPixelcolor);

```

```

nGray := Trunc((nRed*0.299)+(nGreen*0.587)+(nBlue*0.114));

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

line2:=0;

if nGray>=80 then
begin
  line1:=y;
  ps[y]:=x; // second point
  // image2.Canvas.Pixels[x,y] := 0+(255 shl 8)+(0 shl 16);

  if new=0 then
  begin
    p[y]:=1;// have white pixel
    pt[y]:=x;// mem second point
    // image2.Canvas.Pixels[x,y] := 255+(0 shl 8)+(0 shl 16);
    end;

    new:=new+1;

  end;

  if (nGray <100) and (new=0) then
  begin
    p[y]:=0;// mem err point
    end;

  end; // for x
end; // for y

////////// clean point ////////////

for y := 0 to image2.Height-1 do
begin
  for x := 0 to image2.Width-1 do

  begin
    nPixelColor := image2.Canvas.Pixels[x,y];
    nRed := GetRValue(nPixelcolor);
    nGreen := GetGValue(nPixelcolor);
    nBlue := GetBValue(nPixelcolor);

    nGray := Trunc((nRed*0.299)+(nGreen*0.587)+(nBlue*0.114));

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if p[y]=1 then // have white in line
begin

if nGray<80 then // black
begin
if (x>pt[y]) and (x<ps[y]) then // in picture
begin
image2.Canvas.Pixels[x,y] := nGray+(nGray shl 8)+(nGray shl 16);
end
else // out picture
begin
image2.Canvas.Pixels[x,y] := 0+(0 shl 8)+(0 shl 16);
end;
end // end black point
end // end p[y]=1
else
begin
image2.Canvas.Pixels[x,y] := 0+(0 shl 8)+(0 shl 16);
end;

end; // for x
ProgressBar1.Position:=y; // show work

end; // for y

end;
////////////////////
procedure TForm1.Image2OnMouseMove(Sender: TObject; Shift: TShiftState; X,
Y: Integer);

```

```

Var Co,R,G,B,Gray: Integer;

```

```

begin

```

```

Edit1.Text:=IntToStr(X);

```

```

Edit2.Text:=IntToStr(Y);

```

```

Co := image2.Canvas.Pixels[X,Y];

```

```

R:= GetRValue(Co);

```

```

G:= GetGValue(Co);

```

```

B:= GetBValue(Co);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Shape1.Brush.Color:=co;

Gray := Trunc((R*0.299)+(G*0.587)+(B*0.114));
Edit3.Text:=IntToStr(Gray);
// end;

end;
////////////////////
procedure TForm1.Button2Click(Sender: TObject);
var i,x,y,nPixelColor,nRed,nGreen,nBlue,nGray : integer;

begin

i:=0;

xy[0,0]:=0; // reset mem
xy[0,1]:=0;
xy[1,0]:=0;
xy[1,1]:=0;

for y := 0 to image2.Height-1 do
begin
for x := 0 to image2.Width-1 do

begin
nPixelColor := image2.Canvas.Pixels[x,y];
nRed := GetRValue(nPixelcolor);
nGreen := GetGValue(nPixelcolor);
nBlue := GetBValue(nPixelcolor);
nGray := Trunc((nRed*0.299)+(nGreen*0.587)+(nBlue*0.114));

////////////////////
if nGray<=100 then // white =255
begin
// image2.Canvas.Pixels[x,y] := 255+(255 shl 8)+(255 shl 16);

end // if

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 else
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

begin           // back =0
//image2.Canvas.Pixels[x,y] := 0+(0 shl 8)+(0 shl 16);

////////// find x,y max min
i:=i+1; // count pixel

if i=1 then  /////// first point
begin
  xy[0,0]:=x; // x min
  xy[0,1]:=y; // y min
  xy[1,0]:=x; // x max
  xy[1,1]:=y; // y max
end;
////////// next point

if x <=xy[0,0] then xy[0,0]:=x;
if x >=xy[1,0] then xy[1,0]:=x;
if y >=xy[1,1] then xy[1,1]:=y;

////////// end find
end; // else

////////// show gray ///////
image2.Canvas.Pixels[x,y] := nGray+(nGray shl 8)+(nGray shl 16);

nGrayValue[x,y] :=nGray; // recrod

end; // for x
// showMessage(inttostr(nGrayValue[100,80]));
ProgressBar1.Position:=y; // show work

end; // for y

xmi:=xy[0,0]; // x min
ymi:=xy[0,1]; // y min
xma:=xy[1,0]; // x max
yma:=xy[1,1]; // y max
Label9.Caption:=IntToStr(xmi); // x min

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่า Label9.Caption:=IntToStr(xmi); // x min ก็คือตัดแปงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Label11.Caption:=IntToStr(ymin); // y min
Label10.Caption:=IntToStr(xmax); // x max
Label12.Caption:=IntToStr(ymax); // y max

```

```
end;
```

```
////////////////////////////////////
```

```
procedure TForm1.Button6Click(Sender: TObject);
```

```
begin
```

```
    image2.Picture:=image1.Picture;
```

```
end;
```

```
////////////////////////////////////
```

```
procedure TForm1.ZoomClick(Sender: TObject);
```

```
var zoom,x,y,co,i,j,r :Integer; // r=ratio of zoom
```

```
    nPixelColor,nRed,nGreen,nBlue,nGray : integer;
```

```
begin
```

```
    for y:= Ymin to Ymax do //59 to 179 do
```

```
        begin
```

```
            for x:= Xmin to Xmax do //69 to 229 do
```

```
                begin
```

```
                    zoom:=StrToInt(Edit4.Text);
```

```
                if zoom > 1 then
```

```
                    begin
```

```
                        r:=zoom;
```

```
                    end
```

```
                else
```

```
                    begin
```

```
                        Edit4.Text:='1';
```

```
                        r:=1;
```

```
                    end;
```

```
                end;
```

```
            end;
```

```
        /////////////////////////////////// zoom from center of image
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่
 $C_x := \text{trunc}(((xy[1,0] - xy[0,0]) / 2) + xy[0,0]); // (\text{max} - \text{min}) / 2 + \text{min} = \text{center } x$
อ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Cy:=trunc(((xy[1,1]-xy[0,1])/2)+xy[0,1]); // (max-min)/2 +min = center y
```

```
Xmi:=Cx- trunc(256/(2*r)); // xmin :320 is width of image
```

```
Xma:=Cx+ trunc(256/(2*r)); // xmax :320 is width of image
```

```
Ymi:=Cy- trunc(256/(2*r)); // ymin :240 is heigh of image
```

```
Yma:=Cy+ trunc(256/(2*r)); // ymax :240 is heigh of image
```

```
for y := 0 to image2.Height-1 do
```

```
begin
```

```
ProgressBar1.Position:=y mod 100;
```

```
for x := 0 to image2.Width-1 do
```

```
// show work
```

```
begin
```

```
Pixels[x,y] := image2.Canvas.Pixels[x,y];
```

```
end; //x
```

```
end; // y
```

```
////////////////////////////////////
```

```
for y:= Ymi to Yma do //59 to 179 do
```

```
begin
```

```
for x:= Xmi to Xma do //69 to 229 do
```

```
begin
```

```
for i:=0 to r do // 2 =zoom ratio
```

```
begin
```

```
for j:=0 to r do
```

```
begin
```

```
image2.Canvas.Pixels[(x*r)+i-((Xmi)*r),(y*r)+j-((Ymi)*r)]:=Pixels[x,y];
```

```
end; //j
```

```
end; //i
```

```
end; // x
```

```
ProgressBar1.Position:=y mod 100; // show work
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
end; // y
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

end;

////////////////////////////////////

procedure TForm1.HistogramClick(Sender: TObject);

var x,y,Co1,R,Gr,B1,Gray:Integer;

his:array[0..255] of Integer;

begin

Series1.Clear;

for x:=0 to 255 do his[x]:=0; // reset his

////////// read pixel and memory to his

for y:= 0 to image2.Height-1 do //59 to 179 do

begin

for x:= 0 to image2.Width-1 do //69 to 229 do

begin

Co1:= image2.Canvas.Pixels[x,y];

R := GetRValue(Co1);

Gr:= GetGValue(Co1);

B1:= GetBValue(Co1);

Gray := Trunc((R*0.299)+(Gr*0.587)+(B1*0.114));

his[Gray]:=his[Gray]+1;

end;

end;

////////// draw

for x:=0 to 255 do

begin

Series1.AddXY(x,his[x],clTeeColor);

end;

Chart1.LeftAxis.Maximum:=Series1.MaxYValue;

////////// show max

Label19.Caption:=CurrToStr(series1.MaxYValue);

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ for x:=0 to 255 do new draw อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
begin
    Series1.AddXY(x,his[x],"cTeeColor);
```

```
if his[x]=Series1.MaxYValue then
    Label21.Caption:=intToStr(x);
```

```
end;
```

```
end;
```

```
////////////////////////////////////
```

```
procedure TForm1.ComboBox1Change(Sender: TObject);
```

```
begin
```

```
    case ComboBox1.ItemIndex of
```

```
        0 : // lowpass
```

```
            begin
```

```
                E11.Text:=InttoStr(1);
```

```
                E12.Text:=InttoStr(1);
```

```
                E13.Text:=InttoStr(1);
```

```
                E21.Text:=InttoStr(1);
```

```
                E22.Text:=InttoStr(1);
```

```
                E23.Text:=InttoStr(1);
```

```
                E31.Text:=InttoStr(1);
```

```
                E32.Text:=InttoStr(1);
```

```
                E33.Text:=InttoStr(1);
```

```
            Edit6.Text:=InttoStr(9);
```

```
            end;
```

```
        1 : // lowpass
```

```
            begin
```

```
                E11.Text:=InttoStr(1);
```

```
                E12.Text:=InttoStr(1);
```

```
                E13.Text:=InttoStr(1);
```

```
                E21.Text:=InttoStr(1);
```

```
                E22.Text:=InttoStr(2);
```

```
                E23.Text:=InttoStr(1);
```

```
                E31.Text:=InttoStr(1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

E32.Text:=InttoStr(1);

E33.Text:=InttoStr(1);

Edit6.Text:=InttoStr(10);

end;

2 : // lowpass

begin

E11.Text:=InttoStr(1);

E12.Text:=InttoStr(2);

E13.Text:=InttoStr(1);

E21.Text:=InttoStr(2);

E22.Text:=InttoStr(4);

E23.Text:=InttoStr(2);

E31.Text:=InttoStr(1);

E32.Text:=InttoStr(2);

E33.Text:=InttoStr(1);

Edit6.Text:=InttoStr(16);

end;

3 : // lowpass 5*5

begin

E00.Text:=InttoStr(1);

E01.Text:=InttoStr(1);

E02.Text:=InttoStr(1);

E03.Text:=InttoStr(1);

E04.Text:=InttoStr(1);

E10.Text:=InttoStr(1);

E11.Text:=InttoStr(4);

E12.Text:=InttoStr(4);

E13.Text:=InttoStr(4);

E14.Text:=InttoStr(1);

E20.Text:=InttoStr(1);

E21.Text:=InttoStr(4);

E22.Text:=InttoStr(12);

E23.Text:=InttoStr(4);

E24.Text:=InttoStr(1);

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่าจะในรูปแบบใดก็ตาม หากมีให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

E30.Text:=InttoStr(1);
E31.Text:=InttoStr(4);
E32.Text:=InttoStr(4);
E33.Text:=InttoStr(4);
E34.Text:=InttoStr(1);

E40.Text:=InttoStr(1);
E41.Text:=InttoStr(1);
E42.Text:=InttoStr(1);
E43.Text:=InttoStr(1);
E44.Text:=InttoStr(1);

Edit6.Text:=InttoStr(70);

end;

else

end;

end;

////////// Filter ////////////

procedure TForm1.FilterClick(Sender: TObject);
var i,x,y,nPixelColor,nRed,nGreen,nBlue,nGray,han,sum : integer;
    LevelnGray : array[0..256,0..256] of integer;
    mask : array[0..4,0..4] of byte;
    a,b : short integer;

begin
    for y := 0 to image2.Height-1 do
        begin
            for x := 0 to image2.Width-1 do

begin
    เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
    nPixelColor := image2.Canvas.Pixels[x,y];
    ไม่ว่าจะคิดอย่างไรก็ตามก็จำเป็นต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
    nRed := GetR Value(nPixelcolor);

```



```

nGreen := GetGValue(nPixelcolor);
nBlue := GetBValue(nPixelcolor);
nGray := Trunc((nRed*0.299)+(nGreen*0.587)+(nBlue*0.114));
LevelnGray[x,y] := nGray;

end; // end x

end; // end y

```

```

////////// mask //////////

```

```

mask[0,0]:=StrToInt(E00.Text);
mask[0,1]:=StrToInt(E01.Text);
mask[0,2]:=StrToInt(E02.Text);
mask[0,3]:=StrToInt(E03.Text);
mask[0,4]:=StrToInt(E04.Text);

mask[1,0]:=StrToInt(E10.Text);
mask[1,1]:=StrToInt(E11.Text);
mask[1,2]:=StrToInt(E12.Text);
mask[1,3]:=StrToInt(E13.Text);
mask[1,4]:=StrToInt(E14.Text);

mask[2,0]:=StrToInt(E20.Text);
mask[2,1]:=StrToInt(E21.Text);
mask[2,2]:=StrToInt(E22.Text);
mask[2,3]:=StrToInt(E23.Text);
mask[2,4]:=StrToInt(E24.Text);

mask[3,0]:=StrToInt(E30.Text);
mask[3,1]:=StrToInt(E31.Text);
mask[3,2]:=StrToInt(E32.Text);
mask[3,3]:=StrToInt(E33.Text);
mask[3,4]:=StrToInt(E34.Text);

mask[4,0]:=StrToInt(E40.Text);
mask[4,1]:=StrToInt(E41.Text);
mask[4,2]:=StrToInt(E42.Text);
mask[4,3]:=StrToInt(E43.Text);
mask[4,4]:=StrToInt(E44.Text);

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 han := StrToInt(Edit6.Text);
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

////////////////////////////////////
// mask 5X5
// |00|01|02|03|04|
// |10|11|12|13|14|
// |20|21|22|23|24|
// |30|31|32|33|34|
// |40|41|42|43|44|
//
////////////////////////////////////
//
// mask 3X3
//
// |11|12|13|
// |21|22|23|
// |31|32|33|

for y:=0 to image2.Height-1 do
begin
for x:= 0 to image2.Width-1 do
begin
if RadioButton1.Checked=True then // 3*3
begin
sum:=0;

for a:= -1 to 1 do
begin
for b:= -1 to 1 do
begin
sum :=(LevelnGray[x+b,y+a]*mask[2+b,2+a])+sum;

end; // end j
end; // end i

sum:= Trunc(sum/han) mod 256;

end // end if
else
begin

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 begin
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

sum := 0;
for a:= -2 to 2 do
begin
for b:= -2 to 2 do
begin
sum :=(LevelnGray[x+b,y+a]*mask[2+b,2+a])+sum;

end; // end j
end; // end i
sum:= Trunc(sum/han) mod 256;

end; // end else
image2.Canvas.Pixels[x,y] := sum+(sum shl 8)+(sum shl 16);

end; //end x

ProgressBar1.Position:= y;
end; // end y

end;

procedure TForm1.Button7Click(Sender: TObject);
begin
image1.Picture:=image2.Picture;
end;

////////// Histogram Equalization //////////
procedure TForm1.HistogramEqualization(Sender: TObject);
var i,j,x,y,nPixelColor,nRed,nGreen,nBlue,nGray : integer;
his,sumh,f : array[0..256]of integer;
sum : integer;

begin

for x:=0 to 256 do // reset his sumh
begin
his[x]:=0;
sumh[x]:=0;
sum:=0;
end;

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//////////
```

```
for y := (ymi) to (yma) do
```

```
begin
```

```
for x := (xmi) to (xma) do
```

```
begin
```

```
nPixelColor := image2.Canvas.Pixels[x,y];
```

```
nRed := GetRValue(nPixelcolor);
```

```
nGreen := GetGValue(nPixelcolor);
```

```
nBlue := GetBValue(nPixelcolor);
```

```
nGray := Trunc((nRed*0.299)+(nGreen*0.587)+(nBlue*0.114));
```

```
// LevelnGray[x,y] := nGray;
```

```
his[nGray]:=his[nGray]+1; // Gray Level
```

```
end;
```

```
end;
```

```
//////////
```

```
for x:=0 to 256 do // sasom his
```

```
begin
```

```
sum := sum+his[x];
```

```
sumh[x] := sum
```

```
end;
```

```
for y:=(ymi) to (yma) do
```

```
begin
```

```
for x:=(xmi) to (xma) do
```

```
begin
```

```
nPixelColor := image2.Canvas.Pixels[x,y];
```

```
nRed := GetRValue(nPixelcolor);
```

```
nGreen := GetGValue(nPixelcolor);
```

```
nBlue := GetBValue(nPixelcolor);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ใช้สิทธิอันชอบด้วยกฎหมายที่คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

nGray := Trunc((nRed*0.299)+(nGreen*0.587)+(nBlue*0.114));
// LevelnGray[x,y] := nGray;

f[nGray]:=trunc(sumh[nGray]*(255/((xma-xmi)*(y-ma-yi))));

if f[nGray]>255 then f[nGray]:=255;

image2.Canvas.Pixels[x,y]:=f[nGray]+(f[nGray] shl 8)+(f[nGray] shl 16);

Pixels[x,y]:=f[nGray];

end;

end;

```



end;

//////////////////// Direction //////////////////////

procedure TForm1.DirectionClick(Sender: TObject);

var i,j,x,y,nPixelColor,nRed,nGreen,nBlue,nGray,a : integer;

LevelnGray,small :array[0..255,0..255] of integer;

Sd,HisDir : array[1..8]of integer;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

begin
for y := 0 to image2.Height-1 do
begin
for x := 0 to image2.Width-1 do

begin
nPixelColor := image2.Canvas.Pixels[x,y];
nRed := GetRValue(nPixelcolor);
nGreen := GetGValue(nPixelcolor);
nBlue := GetBValue(nPixelcolor);
nGray := Trunc((nRed*0.299)+(nGreen*0.587)+(nBlue*0.114));
LevelnGray[x,y] := nGray;

```

```

image2.Canvas.Pixels[x,y]:=255+(255 shl 8)+(255 shl 16);

```

```

end;

```

```

end;

```

```

for x:= 1 to 7 do HisDir[x]:=0;

```

```

////////////////////////////////////

```

```

for y:=0 to 31 do // 8*32 = 256

```

```

begin

```

```

for x:= 0 to 31 do // 8*32 = 256

```

```

begin

```

```

small[x,y]:=10000;

```

```

for i:= 0 to 7 do // Block

```

```

begin

```

```

for j:= 0 to 7 do

```

```

begin

```

```

if (j=4) and (i=3) then // 1-5

```

```

begin

```

```

Sd[1]:=(LevelnGray[(x*8)+j-4,(y*8)+i] +

```

```

LevelnGray[(x*8)+j-3,(y*8)+i] +

```

```

LevelnGray[(x*8)+j-2,(y*8)+i] +

```

```

LevelnGray[(x*8)+j-1,(y*8)+i] +

```

```

LevelnGray[(x*8)+j+1,(y*8)+i] +

```

```

LevelnGray[(x*8)+j+2,(y*8)+i] +

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น คือลิขสิทธิ์ของเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LevelnGray[(x*8)+j+3,(y*8)+i] -
(7*LevelnGray[(x*8)+j,(y*8)+i]));

// 2

Sd[2]:=(LevelnGray[(x*8)+j-4,(y*8)+i+2] +
LevelnGray[(x*8)+j-3,(y*8)+i+2] +
LevelnGray[(x*8)+j-2,(y*8)+i+1] +
LevelnGray[(x*8)+j-1,(y*8)+i+1] +
LevelnGray[(x*8)+j+1,(y*8)+i] +
LevelnGray[(x*8)+j+2,(y*8)+i-1] +
LevelnGray[(x*8)+j+3,(y*8)+i-1] -
(7*LevelnGray[(x*8)+j,(y*8)+i]));

// 3

Sd[3]:=(LevelnGray[(x*8)+j-4,(y*8)+i+4] +
LevelnGray[(x*8)+j-3,(y*8)+i+3] +
LevelnGray[(x*8)+j-2,(y*8)+i+2] +
LevelnGray[(x*8)+j-1,(y*8)+i+1] +
LevelnGray[(x*8)+j+1,(y*8)+i-1] +
LevelnGray[(x*8)+j+2,(y*8)+i-2] +
LevelnGray[(x*8)+j+3,(y*8)+i-3] -
(7*LevelnGray[(x*8)+j,(y*8)+i]));

// 4

Sd[4]:=(LevelnGray[(x*8)+j-2,(y*8)+i+4] +
LevelnGray[(x*8)+j-2,(y*8)+i+3] +
LevelnGray[(x*8)+j-1,(y*8)+i+2] +
LevelnGray[(x*8)+j-1,(y*8)+i+1] +
LevelnGray[(x*8)+j,(y*8)+i-1] +
LevelnGray[(x*8)+j+1,(y*8)+i-2] +
LevelnGray[(x*8)+j+1,(y*8)+i-3] -
(7*LevelnGray[(x*8)+j,(y*8)+i]));

// 5

Sd[5]:=(LevelnGray[(x*8)+j,(y*8)+i+4] +
LevelnGray[(x*8)+j,(y*8)+i+3] +
LevelnGray[(x*8)+j,(y*8)+i+2] +
LevelnGray[(x*8)+j,(y*8)+i+1] +
LevelnGray[(x*8)+j,(y*8)+i-1] +
LevelnGray[(x*8)+j,(y*8)+i-2] +
LevelnGray[(x*8)+j,(y*8)+i-3] -
(7*LevelnGray[(x*8)+j,(y*8)+i]));

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น ลิขสิทธิ์ของเอกสารนี้สงวนไว้แก่เจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

end;

if (j=3) and (i=3) then // 6-8

begin

// 6

Sd[6]:= (LevelnGray[(x*8)+j-1,(y*8)+i-3] +
LevelnGray[(x*8)+j-1,(y*8)+i-2] +
LevelnGray[(x*8)+j,(y*8)+i-1] +
LevelnGray[(x*8)+j+1,(y*8)+i+1] +
LevelnGray[(x*8)+j+1,(y*8)+i+2] +
LevelnGray[(x*8)+j+2,(y*8)+i+3] +
LevelnGray[(x*8)+j+2,(y*8)+i+4] -
(7*LevelnGray[(x*8)+j,(y*8)+i]));

// 7

Sd[7]:= (LevelnGray[(x*8)+j-3,(y*8)+i-3] +
LevelnGray[(x*8)+j-2,(y*8)+i-2] +
LevelnGray[(x*8)+j-1,(y*8)+i-1] +
LevelnGray[(x*8)+j+1,(y*8)+i+1] +
LevelnGray[(x*8)+j+2,(y*8)+i+2] +
LevelnGray[(x*8)+j+3,(y*8)+i+3] +
LevelnGray[(x*8)+j+4,(y*8)+i+4] -
(7*LevelnGray[(x*8)+j,(y*8)+i]));

// 8

Sd[8]:= (LevelnGray[(x*8)+j-3,(y*8)+i-1] +
LevelnGray[(x*8)+j-2,(y*8)+i-1] +
LevelnGray[(x*8)+j-1,(y*8)+i] +
LevelnGray[(x*8)+j+1,(y*8)+i+1] +
LevelnGray[(x*8)+j+2,(y*8)+i+1] +
LevelnGray[(x*8)+j+3,(y*8)+i+2] +
LevelnGray[(x*8)+j+4,(y*8)+i+2] -
(7*LevelnGray[(x*8)+j,(y*8)+i]));

end;

end; // end j

end; // end i

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for a:= 1 to 8 do      // make Sd +
begin
  if Sd[a]<=0 then
    begin
      Sd[a]:=(-1)*(Sd[a]);
    end;
  end;
end;

```

```

for a:=1 to 8 do      // find small Sd
begin
  if small[x,y]>Sd[a] then
    begin
      small[x,y] := Sd[a];
      KeepDir[x,y] := a;
    end;
  end; // end for a
end; // end x
end; // enn y

```

```

////////// output //////////

```

```

for y:= 0 to 31 do  // 8*32 = 256
begin
  for x:= 0 to 31  do /// 8*32 = 256
    begin
      if KeepDir[x,y]=1 then      // 1
        begin
          HisDir[1]:=HisDir[1]+1;

```

```

// (0,3)-(4,3)-(7,3)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 image2.Canvas.Brush.Color:=clRed;
 image2.Canvas.MoveTo((x*8)+4,(y*8)+3);

ไม่ว่ากรณีใดๆก็ตาม กรุณาแจ้งผู้ดูแลระบบของเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

image2.Canvas.LineTo((x*8)+0,(y*8)+3);
image2.Canvas.LineTo((x*8)+7,(y*8)+3);

end

else if KeepDir[x,y]=2 then // 2
begin
    HisDir[2]:=HisDir[2]+1;
    // (0,5)-(4,3)-(7,2)
    image2.Canvas.Pen.Color:=clGreen;
    image2.Canvas.MoveTo((x*8)+4,(y*8)+3);
    image2.Canvas.LineTo((x*8)+0,(y*8)+5);
    image2.Canvas.LineTo((x*8)+7,(y*8)+2);

end

else if KeepDir[x,y]=3 then // 3
begin
    HisDir[3]:=HisDir[3]+1;
    // (0,7)-(4,3)-(7,0)
    image2.Canvas.Pen.Color:=clBlue;
    image2.Canvas.MoveTo((x*8)+4,(y*8)+3);
    image2.Canvas.LineTo((x*8)+0,(y*8)+7);
    image2.Canvas.LineTo((x*8)+7,(y*8)+0);

end

else if KeepDir[x,y]=4 then // 4
begin
    HisDir[4]:=HisDir[4]+1;
    // (2,7)-(4,3)-(5,0)
    image2.Canvas.Pen.Color:=clLime; // red
    image2.Canvas.MoveTo((x*8)+4,(y*8)+3);
    image2.Canvas.LineTo((x*8)+2,(y*8)+7);
    image2.Canvas.LineTo((x*8)+5,(y*8)+0);

end

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 else if KeepDir[x,y]=5 then //5
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        HisDir[5]:=HisDir[5]+1;
        // (4,7)-(4,3)-(4,0)
        image2.Canvas.Pen.Color:=clMaroon;
        image2.Canvas.MoveTo((x*8)+4,(y*8)+3);
        image2.Canvas.LineTo((x*8)+4,(y*8)+7);
        image2.Canvas.LineTo((x*8)+4,(y*8)+0);

    end

else if KeepDir[x,y]=6 then //6
    begin
        HisDir[6]:=HisDir[6]+1;
        // (2,0)-(3,3)-(5,7)
        image2.Canvas.Pen.Color:=clNavy; // red
        image2.Canvas.MoveTo((x*8)+3,(y*8)+3);
        image2.Canvas.LineTo((x*8)+2,(y*8)+0);
        image2.Canvas.LineTo((x*8)+5,(y*8)+7);

    end

else if KeepDir[x,y]=7 then //7
    begin
        HisDir[7]:=HisDir[7]+1;
        // (0,0)-(3,3)-(7,7)
        image2.Canvas.Pen.Color:=clYellow; // red
        image2.Canvas.MoveTo((x*8)+3,(y*8)+3);
        image2.Canvas.LineTo((x*8)+0,(y*8)+0);
        image2.Canvas.LineTo((x*8)+7,(y*8)+7);

    end

else if KeepDir[x,y]=8 then //8
    begin
        HisDir[8]:=HisDir[8]+1;
        // (0,2)-(3,3)-(7,5)
        image2.Canvas.Pen.Color:=clPurple; // red
        image2.Canvas.MoveTo((x*8)+3,(y*8)+3);
        image2.Canvas.LineTo((x*8)+0,(y*8)+2);
        image2.Canvas.LineTo((x*8)+7,(y*8)+5);
    end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะวิธีใด ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end; // end x
```

```
ProgressBar1.Position:=y;
```

```
end; // enn y
```

```
end;
```

```
////////////////////////////////////
```

```
procedure TForm1.CoreClick(Sender: TObject);
```

```
var a,b,c,d,line,k,m,n,x,y,nPixelColor,nRed,nGreen,nBlue,nGray ,n1,n2 : integer;
```

```
FreqD : array[1..25]of Shortint;
```

```
FreqMAX,i,j : Short int;
```

```
Hr11,Hr12,Hr13,Hr14,Hr21,Hr22,Hr23,Hr24,Hr31,Hr32,Hr33,Hr34,Hr41,Hr42,Hr43,Hr44 : array[1..8] of Shortint;
```

```
Hr51,Hr52,Hr53,Hr54,Hr61,Hr62,Hr63,Hr64,Hr71,Hr72,Hr73,Hr74,Hr81,Hr82,Hr83,Hr84 : array[1..8] of Shortint;
```

```
sumHr11,sumHr13,sumHr21,sumHr23,sumHr31,sumHr33,sumHr41,sumHr43 : integer;
```

```
sumHr51,sumHr53,sumHr61,sumHr63,sumHr71,sumHr73,sumHr81,sumHr83 : integer;
```

```
sumFHr11,sumFHr12,sumFHr21,sumFHr22,sumFHr31,sumFHr32,sumFHr41,sumFHr42 : integer;
```

```
sumFHr51,sumFHr52,sumFHr61,sumFHr62,sumFHr71,sumFHr72,sumFHr81,sumFHr82 : integer;
```

```
F1,F2,F3,F4,F5,F6,F7,F8,G : integer;
```

```
sum11,sum12,sum13,sum14,sum15,sum16,sum17,sum18 : real;
```

```
D11,D12,D21,D22,D31,D32,D41,D42,D51,D52,D61,D62,D71,D72,D81,D82 : array[1..8] of byte;
```

```
F11,F12,F21,F22,F31,F32,F41,F42,F51,F52,F61,F62,F71,F72,F81,F82 : integer;
```

```
Xbar1,Xbar2 : real;
```

```
begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
k:=0;
```

ไม่อนุยัติให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for y:=0 to image2.Height-1 do
```

```

begin
  for x:=0 to image2.Width-1 do
    begin
      nPixelColor := image2.Canvas.Pixels[x,y];
      nRed      := GetRValue(nPixelcolor);
      nGreen    := GetGValue(nPixelcolor);
      nBlue     := GetBValue(nPixelcolor);
      nGray := Trunc((nRed*0.299)+(nGreen*0.587)+(nBlue*0.114));

      end; //end x

    end; // end y
  //////////////////////////////////////

  m:=0; k:=0;
  n:=0;
  G:= 10;
  line:=0;
  for i:=0 to 32 do lineF[i]:=0;
  for i:=0 to 32 do
    begin
      for j:=0 to 32 do
        begin
          memx[j,i]:=0;

          end;
        end;
      //////////////////////////////////////

      for y:=0 to 32 do
        begin
          for x:= 0 to 32 do
            begin

              // check guese core >= 2
              if ((KeepDir[x,y]-KeepDir[x-1,y])>=2) or ((KeepDir[x,y]-KeepDir[x-1,y])<=2) then // if big
                begin

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 // image2.Canvas.Rectangle((x*8)+0,(y*8)+0,(x*8)+7,(y*8)+7);
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

FreqMax:=0;
for i:=1 to 25 do FreqD[i]:=0;

for i:=1 to 8 do          // Reset Hr
begin

    Hr11[i]:=0; Hr12[i]:=0; Hr13[i]:=0; Hr14[i]:=0;
    Hr21[i]:=0; Hr22[i]:=0; Hr23[i]:=0; Hr24[i]:=0;
    Hr31[i]:=0; Hr32[i]:=0; Hr33[i]:=0; Hr34[i]:=0;
    Hr41[i]:=0; Hr42[i]:=0; Hr43[i]:=0; Hr44[i]:=0;
    Hr51[i]:=0; Hr52[i]:=0; Hr53[i]:=0; Hr54[i]:=0;
    Hr61[i]:=0; Hr62[i]:=0; Hr63[i]:=0; Hr64[i]:=0;
    Hr71[i]:=0; Hr72[i]:=0; Hr73[i]:=0; Hr74[i]:=0;
    Hr81[i]:=0; Hr82[i]:=0; Hr83[i]:=0; Hr84[i]:=0;

    D11[i]:=0; D12[i]:=0; D21[i]:=0; D22[i]:=0;
    D31[i]:=0; D32[i]:=0; D41[i]:=0; D42[i]:=0;
    D51[i]:=0; D52[i]:=0; D61[i]:=0; D62[i]:=0;
    D71[i]:=0; D72[i]:=0; D81[i]:=0; D82[i]:=0;

end;

////////////////////////////////////
// mask 5*5
for i:= -5 to 5 do
begin
for j:=-5 to 5 do
begin

case KeepDir[x+j,y+i] of

    1 : FreqD[1]:=FreqD[1]+1;
    2 : FreqD[2]:=FreqD[2]+1;
    3 : FreqD[3]:=FreqD[3]+1;
    4 : FreqD[4]:=FreqD[4]+1;
    5 : FreqD[5]:=FreqD[5]+1;
    6 : FreqD[6]:=FreqD[6]+1;
    7 : FreqD[7]:=FreqD[7]+1;
    8 : FreqD[8]:=FreqD[8]+1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ที่ **else** อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

end; // end case

end; // end j

end; // end i

for a:=1 to 8 do // find fmax
begin
if FreqD[a]>FreqMax then
begin
FreqMax:=FreqD[a];
end;

end; // end for a
////////////////////

// check cure for direct other
if (FreqMax<=23) then

begin

k:=k+1;

// image2.Canvas.Rectangle((x*8)+0,(y*8)+0,(x*8)+7,(y*8)+7);

// Direc 1//

// 1 R1

// if k=1 then begin // test
// image2.Canvas.Rectangle((x*8)+0,(y*8)+0,(x*8)+7,(y*8)+7);
a:=0; b:=0; c:=0; d:=0;
for i:=-4 to 0 do
begin
for j:=0 to 4 do
begin
case keepdir[x+j,y+i] of
1 : Hr11[1]:=Hr11[1]+1;
2 : Hr11[2]:=Hr11[2]+1;
3 : Hr11[3]:=Hr11[3]+1;
4 : Hr11[4]:=Hr11[4]+1;
5 : Hr11[5]:=Hr11[5]+1;
6 : Hr11[6]:=Hr11[6]+1;
7 : Hr11[7]:=Hr11[7]+1;

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

8 : Hr11[8]:=Hr11[8]+1;
else
end; // end case

a:= a + keepdir[x+j,y+i]; // sum find xbar

end; // end j
end; // end i

// showmessage(currtostr(Hr11[1]));

// 1 R2 //
for i:=0 to 4 do
begin
for j:=0 to 4 do
begin
case keepdir[x+j,y+i] of
1 : Hr12[1]:=Hr12[1]+1;
2 : Hr12[2]:=Hr12[2]+1;
3 : Hr12[3]:=Hr12[3]+1;
4 : Hr12[4]:=Hr12[4]+1;
5 : Hr12[5]:=Hr12[5]+1;
6 : Hr12[6]:=Hr12[6]+1;
7 : Hr12[7]:=Hr12[7]+1;
8 : Hr12[8]:=Hr12[8]+1;
else
end; // end case

b:= b + keepdir[x+j,y+i]; // sum find xbar

end; // end j
end; // end i

for i:=1 to 8 do
begin
D11[i]:=Hr11[i]+Hr12[i];

end;

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะในรูปแบบใดก็ตาม อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Xbar1:=0;

Xbar1 := ((a+b)/50) ;

// showMessage(currtostr(xbar1));

sum11:=0;sum12:=0;sum13:=0;sum14:=0;sum15:=0;sum16:=0;sum17:=0;sum18:=0;

for i:=1 to 8 do // x-x bar
begin
case i of
1: sum11:=D11[i]*((1-Xbar1)*(1-Xbar1));
2: sum12:=D11[i]*((2-Xbar1)*(2-Xbar1));
3: sum13:=D11[i]*((3-Xbar1)*(3-Xbar1));
4: sum14:=D11[i]*((4-Xbar1)*(4-Xbar1));
5: sum15:=D11[i]*((5-Xbar1)*(5-Xbar1));
6: sum16:=D11[i]*((6-Xbar1)*(6-Xbar1));
7: sum17:=D11[i]*((7-Xbar1)*(7-Xbar1));
8: sum18:=D11[i]*((8-Xbar1)*(8-Xbar1));

else // else case
end; // end case

end; // end for
F11:=0;
F11:=Trunc((sum11+sum12+sum13+sum14+sum15+sum16+sum17+sum18)/49);

// showMessage(currtostr(f11));

// 1 R3 //
for i:=-4 to 0 do
begin
for j:=-4 to 0 do
begin
case keepdir[x+j,y+i] of
1 : Hr13[1]:=Hr13[1]+1;
2 : Hr13[2]:=Hr13[2]+1;
3 : Hr13[3]:=Hr13[3]+1;
4 : Hr13[4]:=Hr13[4]+1;
5 : Hr13[5]:=Hr13[5]+1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใด ๆ ก็ให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

6 : Hr13[6]:=Hr13[6]+1;
7 : Hr13[7]:=Hr13[7]+1;
8 : Hr13[8]:=Hr13[8]+1;
else
end; // end case

c:=c+keepdir[x+j,y+i];
end; // end j
end; // end i

7: sum17:=D82[i]*((7-Xbar2)*(7-Xbar2));
8: sum18:=D82[i]*((8-Xbar2)*(8-Xbar2));

else // else case
end; // end case

end; // end for
F82:=0;
F82:=Trunc((sum11+sum12+sum13+sum14+sum15+sum16+sum17+sum18)/(49));
// showmessage(currtostr(f81));

if (F81>F82) then
begin
F8:=F81;
end
else F8:=0;

////////////////////////////////////

```

```

if (F11)and(F21)and(F31)and(F41)and(F51)and(F61)and(F71)and(F81) > 0

```

```

then

```

```

begin // 1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะฉีกหรือดัดแปลงขึ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if line=y then // if line mem point x ,y
begin
lineF[y]:=y;
memx[x,y]:=x;

// showmessage(inttostr(memx[x,y]));

end; // end line

// k:=k+1;

// image2.Canvas.Rectangle((x*8)+0,(y*8)+0,(x*8)+7,(y*8)+7);

end; // end 1

F11:=0;F12:=0;F21:=0;F22:=0;F31:=0;F32:=0;F41:=0;F42:=0;
F51:=0;F52:=0;F61:=0;F62:=0;F71:=0;F72:=0;F81:=0;F82:=0;

// image2.Canvas.Rectangle((x*8)+0,(y*8)+0,(x*8)+7,(y*8)+7);
// end; // end test

end; // end if freqmax

end; // end if big

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 end; // end x
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ProGressBar1.Position:=y;  
end; // end y  
  
// showmessage(inttostr(k));  
  
end;
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี ด้วยความอนุเคราะห์ทั้งทางด้านความรู้ คำปรึกษา คำแนะนำ และอุปกรณ์ ของ ดร. ยุทธนา คีตใจเดียว ซึ่งต้องขอขอบพระคุณไว้ ณ ที่นี้ด้วย

ขอบพระคุณเพื่อน ๆ และพี่ ๆ ที่ให้คำปรึกษาเกี่ยวกับการเขียน โปรแกรม และคำแนะนำต่าง ๆ จนสำเร็จสมบูรณ์ยิ่งขึ้น

สุดท้ายนี้ขอขอบคุณ เตี่ยและแม่ที่คอยให้กำลังใจลูกคนนี้เสมอ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. ครรชิต ไมตรี, มนัส สังวรศิลป์ และ สุรพันธ์ เอื้อไพฑูรย์, “ระบบเปรียบเทียบลายนิ้วมืออัตโนมัติ” , การประชุมวิชาการครั้งที่ 4 ของ NECTEC , ณ โรงแรมอโนมาสวิทโฮเต็ล , 21 – 22 ตุลาคม 2535 , หน้า (17 – 27)
2. Rafael C.Gonzalez , Richard E. Wood , “ Digital Image Processing ” , 1992
3. C.V. Kameshwara Roo , B. Prasad and K.R. Shama , “ Automatic fingerprint a syntactic approach ” , Proc.3rd Int. Conf. on Pattern Recognition , San Diego , November 1976 , pp. (778 – 782)
4. Robert J. Schalkoff , “ Digital image processing and computer version ” , John Wiley & Sons , Singapore , 1989



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้