สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โปรแกรมจดจำวัตถุ

Object Recognition Program

โดย

นาย วีระชาติ อินต๊ะมี

นาย อรรถกร พิจยานนท์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขหมู่.....
เลขทะเบียน 64205
วัน,เดือน,ปี 7 ก.ย. 2549

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมจดจำวัตถุ

Object Recognition Program

โดย

นาย วีระชาติ อินต๊ะมี รหัส 45015253

นาย อรรถกร พิจยานนท์ รหัส 45015263

อาจารย์ที่ปรึกษา

ผศ.ดร. ชูชาติ ปิณฑวิรุจน์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2547

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โปรแกรมจดจำวัตถุ (Object Recognition Program)

ผู้จัดทำ

1. นาย วีระชาติ อินต๊ะมี รหัส 45015253
2. นาย อรรถกร พิจยานนท์ รหัส 45015263

.....

(ผศ.ดร. ชูชาติ ปิณฑวิรุจน์)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์เรื่อง โปรแกรมจดจำวัตถุ

Object Recognition Program

จัดทำโดย นาย วีระชาติ อินต๊ะมี รหัส 45015253

นาย อรรถกร พิจยานนท์ รหัส 45015263

อาจารย์ที่ปรึกษา ผศ.ดร. ชูชาติ ปิณฑวิรุจน์

ปริญญานิพนธ์นี้ได้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว

.....

(ผศ.ดร. ชูชาติ ปิณฑวิรุจน์)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมจดจำวัตถุ

นาย วีระชาติ อินต๊ะมี
นาย อรรถกร พิจยานนท์
ผศ.ดร. ชูชาติ ปิณฑวิรุจน์ (อาจารย์ที่ปรึกษา)
ปีการศึกษา 2547

บทคัดย่อ

ในรายงานฉบับนี้กล่าวถึงหลักการและทฤษฎีต่าง ๆ เกี่ยวกับการประมวลผลและการจดจำรูปแบบตัวอักษร ได้กล่าวถึงรายละเอียดของโครงงานที่ได้ทำ นั่นคือโปรแกรมจดจำวัตถุออกจากภาพ ซึ่งประกอบไปด้วยส่วนของการหาขอบเขตภาพ การกำจัดสิ่งรบกวนของวัตถุ การแปลงภาพสีเป็นขาวดำ การหาขอบเขตของวัตถุ การจดจำรูปแบบของวัตถุโดยวิธีโมเมนท์เกี่ยวกับระยะทางของภาพแบบดีสครีต รวมทั้งการทดลองและสรุปผลการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Object Recognition Program

Mr. Weerachart Intamee

Mr. Atthakorn Pitjayanont

Asst. Prof. Dr. Chucha Pintavirutne (Advisor)

Educational Year 2004

ABSTRACT

This report concerns about principles and theory of digital image processing and character recognition and including the detail of designing the " Object Recognition Program " which consist of edge detection , noise reduction , binarization , object segmentation and object recognition system using discrete image spatial moment . We also test the program and conclude the result of the test .

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้เป็นอย่างดีด้วยคำแนะนำและคำปรึกษาจากอาจารย์ชูชาติ ปิณฑวิรุจน์ ซึ่งเป็นอาจารย์ที่ปรึกษาในการทำงาน และ รุ่นพี่นักศึกษาปริญญาโทภาควิชาอิเล็กทรอนิกส์ ผู้จัดทำรู้สึกซาบซึ้งในความอนุเคราะห์จากท่านและกราบขอบพระคุณอย่างสูง

ขอบพระคุณเพื่อน ๆ ทุก ๆ คนที่ช่วยเหลือ ให้คำแนะนำต่าง ๆ และยังให้กำลังใจตลอดมา

คุณค่าและประโยชน์อันพึงมีจากปริญญานิพนธ์นี้ ผู้จัดทำขอมอบแด่ผู้มีพระคุณทุก ๆ ท่าน

วีระชาติ อินต๊ะมี

(นาย วีระชาติ อินต๊ะมี)

อรรถกร พิจยานนท์

(นาย อรรรถกร พิจยานนท์)

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
Abstract	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VII
สารบัญตาราง	IX
บทที่ 1 บทนำ	1
1.1 วัตถุประสงค์ของโครงงาน	1
1.2 ขอบเขตของโครงงาน	1
1.3 การประยุกต์ใช้งานของโปรแกรมจดจำวัตถุ	2
บทที่ 2 การประมวลผลภาพ	3
2.1 การประมวลผลภาพเชิงตัวเลข (Digital Image Processing)	3
2.1.1 การแทนภาพด้วยข้อมูลแบบดิจิตอล	3
2.1.2 ลักษณะการจัดเก็บข้อมูลภาพแบบดิจิตอล	4
2.2 การสร้างภาพไบนารี่	5
2.3 การแบ่งส่วนภาพ (Image Segmentation)	8
2.3.1 การตรวจรู้แนวเส้น (Line Detection)	8
2.3.2 การตรวจหาขอบ (Edge detection)	9
2.3.2.1 วิธีแกรเดียนท์ (Gradient Method)	10
2.3.2.2 วิธีลาปลาเซียน (Laplacian Method)	12
2.3.2.3 การหาขอบแบบแคนนี (Canny edge detection)	13
2.4 ทฤษฎีการแยกตัวอักขระออกจากภาพ	14
2.4.1 การหากรอบตัวอักขระโดย วิธี Line Crossing	14
2.4.2 เทคนิคการตามรอยขอบภาพ (Contour Following)	17
2.4.3 เทคนิคการแยกโดยวิธีการ Contour with matrix	19
2.4.4 เทคนิคการแยกตัวอักขระที่ติดกัน โดยHistogram	21
2.5 โมเมนต์เกี่ยวกับระยะทางของภาพแบบดีสครีต (Discrete Image Spatial Moment)	24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 คุณสมบัติพื้นฐานของ MCS-51 29
2.7 โครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51 29
2.8 โครงสร้างหน่วยความจำ 30
2.8.1 หน่วยความจำโปรแกรม 30
2.8.2 หน่วยความจำข้อมูล 31
2.9 วงจรนับ/จับเวลา (TIMMER/COUNTER) 32
2.10 ชุดคำสั่งในไมโครคอนโทรลเลอร์ MCS-51 33
2.11 โครงสร้างการอินเทอร์รัพต์ของ MCS-51 34
2.12 ฐานเวลาการทำงานของ CPU 35
2.13 รีจิสเตอร์หน้าที่พิเศษ SFR 35
2.14 พอร์ตอินพุต/เอาท์พุท 37
2.15 การใช้งานของพอร์ตอนุกรม 37
2.15.1 ความเร็วของการสื่อสารข้อมูลอนุกรม 38
2.15.2 รูปแบบการส่งข้อมูลอนุกรม 38
2.15.2.1 บิตเริ่มต้น (Start Bit) 38
2.15.2.2 บิตแสดงสภาวะความเป็นเลขคู่หรือเลขคี่ (Parity Bit) 38
2.15.2.3 บิตสุดท้าย (Stop Bit) 38
2.15.3 การส่งข้อมูลอนุกรมของ 8051 39
2.15.4 โหมดการทำงานของพอร์ตอนุกรมใน MCS – 51 40
2.15.4.1 พอร์ตอนุกรม (โหมด 0) 40
2.15.4.2 การรับส่งข้อมูลอนุกรม (โหมด 1) 40
2.15.4.3 การรับส่งข้อมูลอนุกรม (โหมด 2) 41
2.15.4.4 การรับส่งข้อมูลอนุกรม (โหมด 3) 42
2.15.5 การอินเทอร์รัพต์ของพอร์ตสื่อสารอนุกรม 42
2.15.5.1 กระบวนการรับส่งข้อมูลอนุกรมของ 8051 42
บทที่ 3 หลักการทำงานของโปรแกรม 44
3.1 โครงสร้างของโปรแกรมการจดจำวัตถุ 44
3.2 โปรแกรมการแปลงภาพสีเป็นภาพเกรย์สเกล 45
3.3 โปรแกรมการแปลงภาพเกรย์เป็นภาพขาวดำ 46
3.4 โปรแกรมการหาค่าโมเมนต์ของภาพ 47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 โปรแกรมตรวจจับภาพวัตถุและตรวจนับจำนวนของวัตถุ 48
3.6 โปรแกรมควบคุมการทำงานของมอเตอร์กระแสตรงโดย MCS – 51 49
บทที่ 4 ผลการทดลอง 51
4.1 การทดลองการแปลงภาพ True Color เป็นภาพแบบ GRAY SCALE 51
4.2 การทดลองการแปลงภาพ GRAY SCALE เป็นภาพแบบ BINARY 51
4.3 การทดลองหาค่า Moment ของภาพ 52
4.4 ทดลองหาค่าเฉลี่ยโมเมนท์ของวัตถุแต่ละอันดับ 55
4.5 ทดลองหาการตรวจจับภาพวัตถุและตรวจนับจำนวนของวัตถุ 57
4.5.1 การทดลองครั้งที่ 1 57
4.5.2 การทดลองครั้งที่ 2 57
4.5.3 การทดลองครั้งที่ 3 58
4.5.4 การทดลองครั้งที่ 4 59
4.5.5 การทดลองครั้งที่ 5 59
บทที่ 5 สรุปและวิจารณ์ผลการทดลอง 60
5.1 วิจารณ์ผลการทดลอง 60
5.2 สรุปผลการทดลอง 61
หนังสืออ้างอิง 62

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่ 1.1 ขั้นตอนการทำงานของโปรแกรม	2
รูปที่ 2.1 ภาพไบนารีและข้อมูลของแต่ละพิกเซล	6
รูปที่ 2.2 หน้ากากที่ใช้ในการหาแนวเส้น	9
รูปที่ 2.3 ตัวดำเนินการที่ใช้ในการคำนวณหาค่าแกรเดียนธ์	12
รูปที่ 2.4 หน้ากากที่ใช้ในตัวดำเนินการลาปลาเซียน	12
รูปที่ 2.5 แสดงการแยกตัวอักขระด้วยวิธี Line Crossing	15
รูปที่ 2.6 แสดง Flow Chart การแยกตัวอักขระด้วยวิธี Line Crossing	16
รูปที่ 2.7 แสดงการหาขอบด้วยวิธี Contour Following	17
รูปที่ 2.8 แสดง Flow Chart การหาขอบด้วยวิธี Contour Following	18
รูปที่ 2.9 เมตริกซ์ขนาด 3 X 3	19
รูปที่ 2.10 แสดงการหาขอบโดยวิธี Contour with matrix	19
รูปที่ 2.11 แสดง Flow Chart แสดงการหาขอบโดยวิธี Contour with matrix	20
รูปที่ 2.12 แสดงการกำหนดจุดตัดด้วย Histogram	22
รูปที่ 2.13 แสดง Flow Chart การแยกตัวอักขระที่ติดกันโดยการใช้ Histogram	23
รูปที่ 2.14 สัญญาลักษณ์ของไฟที่ถูกหมุนขยายและย่อ	26
รูปที่ 2.15 โครงสร้างภายในของ MCS-51	30
รูปที่ 2.16 แสดงการจัดพื้นที่หน่วยความจำโปรแกรมสำหรับ MCS-51	31
รูปที่ 2.17 แสดงการจัดพื้นที่หน่วยความจำสำหรับ MCS-51	32
รูปที่ 2.18 แสดงบิตแต่ละบิตของการใช้งานพอร์ตอนุกรม	39
รูปที่ 3.1 ขั้นการทำงานของโปรแกรม	44
รูปที่ 3.2 แสดงแผนผังโปรแกรมการแปลงภาพสีเป็นภาพเกรย์สเกล	45
รูปที่ 3.3 แสดงแผนผังการทำงานของโปรแกรม Binarization	46
รูปที่ 3.4 แสดงขั้นตอนการหาค่าโมเมนต์	47
รูปที่ 3.5 แสดงขั้นตอนตรวจจับภาพวัตถุและตรวจนับจำนวนของวัตถุ	48
รูปที่ 3.6 แสดงการควบคุมการทำงานของมอเตอร์กระแสตรง	49
รูปที่ 3.7 แสดงการเชื่อมต่อของวงจรควบคุมการทำงานของมอเตอร์กระแสตรง	50
รูปที่ 4.1 แสดงภาพที่ได้จากการแปลงภาพ GRAY SCALE	51
รูปที่ 4.2 แสดงภาพตัวอย่างที่ได้จากการแปลงเป็นภาพ BINARY	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.3 แสดงภาพตัวอย่างที่ใช้ในการค่า Moment 52
รูปที่ 4.4 แสดงการทดลองครั้งที่ 1 57
รูปที่ 4.5 แสดงการทดลองครั้งที่ 2 58
รูปที่ 4.6 แสดงการทดลองครั้งที่ 3 58
รูปที่ 4.7 แสดงการทดลองครั้งที่ 4 59
รูปที่ 4.8 แสดงการทดลองครั้งที่ 5 59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 แสดง Moment of Intertial Data of Test Images 25

ตารางที่ 2.2 แสดง Invariant Moment of Test Image 27

ตารางที่ 2.3 แสดง Invariant Moment of Test Image 28

ตารางที่ 2.4 ลักษณะสัญญาณที่เข้ามาอินเทอร์รัพต์ 34

ตารางที่ 2.5 ระดับความสำคัญและตำแหน่งแอดเดรสของ Interrupt Service Routine 34

ตารางที่ 2.6 แสดงบิตต่างๆ ภายในรีจิสเตอร์ PSW 35

ตารางที่ 2.7 แสดงการจัดพื้นที่ของหน่วยความจำรีจิสเตอร์หน้าที่พิเศษ 36

ตารางที่ 2.8 แสดงขาสัญญาควบคุมต่าง ๆ 37

ตารางที่ 2.9 รายละเอียดการเลือกโหมดการทำงาน 40

ตารางที่ 2.10 การเลือกอัตราบอดของวงจรพอร์ตอนุกรม 41

ตารางที่ 2.11การเซตค่าอัตราการส่งข้อมูล 43

ตารางที่ 4.1 แสดงการค่าScaled Moment (M(m,n)) ของภาพทดสอบ 51

ตารางที่ 4.2 แสดงการค่า Scaled Central Moment (U(m,n)) ของภาพทดสอบ 51

ตารางที่ 4.3 แสดงการค่า Normalization Scaled Central Moment (V(m,n)) ของภาพทดสอบ 52

ตารางที่ 4.4 แสดงการค่า Moment Invariant (h(i)) ของภาพทดสอบ 52

ตารางที่ 4.5 หาค่าเฉลี่ยโมเมนท์ของวัตถุโดยใช้โมเมนท์อันดับที่ 3 55

ตารางที่ 4.6 หาค่าเฉลี่ยโมเมนท์ของวัตถุโดยใช้โมเมนท์อันดับที่ 1 55

ตารางที่ 4.7 หาค่าเฉลี่ยโมเมนท์ของวัตถุโดยใช้โมเมนท์อันดับที่ 4 56

ตารางที่ 4.8 หาค่าเฉลี่ยโมเมนท์ของวัตถุโดยใช้โมเมนท์อันดับที่ 2 56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ในปัจจุบันเทคโนโลยีต่าง ๆ ได้พัฒนาขึ้นมาอย่างรวดเร็วมาก และได้คิดค้นทฤษฎีเกี่ยวกับการประมวลผลภาพเชิงตัวเลข (Digital Image Processing) ซึ่งได้นำไปประยุกต์ใช้ในงานด้านอุตสาหกรรมต่างๆ เกี่ยวกับทางด้านการแพทย์ และทางด้านระบบความปลอดภัย โดยใช้หลักการเปลี่ยนข้อมูลที่อยู่ในลักษณะรูปภาพ (Image Data) ให้อยู่ในรูปแบบการจัดการจัดเก็บข้อมูลในรูปตัวหนังสือ (Text Data) หรืออยู่ในรูปแบบการจัดเก็บข้อมูลในรูปร่างของวัตถุ (Object Recognition Data) เพื่อนำข้อมูลที่ได้ไปใช้กับระบบฐานข้อมูล (Database) หรือประยุกต์ใช้กับงานอื่น ๆ ได้ โดยอาศัยหลักการประมวลผลภาพ (Image Processing) และการจดจำรูปแบบ (Pattern Recognition) เป็นหลักสำคัญในการออกแบบ

1.1 วัตถุประสงค์ของโครงงาน

- เพื่อศึกษาหลักการประมวลผลภาพ
- เพื่อศึกษาหลักการจดจำรูปแบบ
- เพื่อศึกษาหลักการเขียนโปรแกรมด้วยภาษา เดลไฟ

1.2 ขอบเขตของโครงงาน

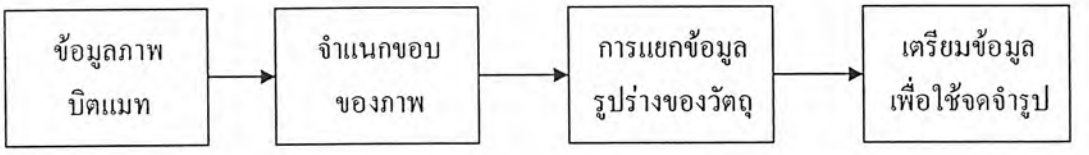
ระบบโปรแกรมจดจำวัตถุที่นำเสนอนี้ จะนำเอาข้อมูลภาพของวัตถุเป็นข้อมูลเพื่อใช้ในการหาตำแหน่งของรูปร่างของวัตถุโดยอาศัยขั้นตอนการประมวลผลภาพ เช่น การกำจัดสัญญาณรบกวน การทำไบนารี่ไรเซชัน (Image Binarization) การจำแนกภาพ (Image Segmentation) แล้วนำผลลัพธ์ทดเข้าสู่กระบวนการจดจำรูปแบบของวัตถุ เป็นการทำให้คอมพิวเตอร์สามารถรู้จักและเข้าใจภาพได้ (recognition)

ปริญญานิพนธ์ฉบับนี้ได้เสนอวิธีการแยกรูปร่างของวัตถุจากข้อมูลภาพเพื่อเตรียมข้อมูลเข้าสู่ขั้นตอนการจดจำรูปแบบโดยนำข้อมูลภาพของวัตถุที่มีการเก็บข้อมูลในการวิเคราะห์เพื่อแยกเฉพาะรูปร่างของวัตถุออกจากข้อมูลภาพทั้งหมด โดยมีขั้นตอนดังรูปที่ 1.1

หลังจากผ่านขั้นตอนของการแยกรูปร่างของวัตถุแล้ว ข้อมูลภาพรูปร่างของวัตถุจะนำเข้าสู่กระบวนการจดจำรูปแบบ เพื่อทำการประมวลผลข้อมูลและให้ผลลัพธ์ออกมาเป็นวัตถุที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรายงานฉบับนี้ได้เสนอวิธีการจดจำวัตถุโดยหลักการของ Hu Moment Invariant เพื่อที่จะนำไปเปรียบกับข้อมูลภาพที่เก็บไว้ใน Database ซึ่งรายละเอียดการทำงานจะอธิบาย ต่อไป



รูปที่ 1.1 ขั้นตอนการทำงานของโปรแกรม

1.3 การประยุกต์ใช้งานของโปรแกรมจดจำวัตถุ

- ระบบเซ็นเซอร์ภายในโรงงานอุตสาหกรรม

ในโรงงานอุตสาหกรรมทั่วไป จะเห็นได้ว่ามีวัสดุอุปกรณ์ต่าง ๆ มากมาย ซึ่งเราจำเป็นที่จะต้องตรวจเช็คจำนวนเชิงอุปกรณ์วัสดุเหล่านั้น ซึ่งการประยุกต์ใช้งานนี้จะช่วยทำให้เราสามารถลดต้นทุนใน ว่าจ้างพนักงาน และยังเพิ่มความแม่นยำและเที่ยงตรงในการตรวจจับและนับจำนวนของอุปกรณ์เหล่านั้น ทั้งนี้ทั้งนั้นยังช่วยในการประหยัดเวลาในการทำงาน

- ตรวจจับข้อผิดพลาดของแผ่นวงจร

ในอุตสาหกรรมที่ประกอบหรือติดตั้งอุปกรณ์ลงแผ่นวงจร ที่ใช้แรงงานคนในการลงอุปกรณ์ต่าง ๆ ในวงจร อาจเกิดข้อผิดพลาด เนื่องจากใส่อุปกรณ์ไม่ถูกต้องไม่ครบ ดังนั้นเราสามารถที่จะประยุกต์นำมาใช้ในการตรวจจับข้อผิดพลาดเหล่านั้น

- ระบบรักษาความปลอดภัย

ในปัจจุบันได้มีการติดตั้งกล่องวงจรปิดไว้ตามที่ต่าง ๆ เช่น สถานที่จอดรถ ร้อนทอง หรือธนาคาร เป็นต้น ซึ่งเราสามารถที่จะกำหนดให้บันทึกเฉพาะช่วงเวลาที่มีการเปลี่ยนแปลงเกิดขึ้นเท่านั้น เพื่อเป็นการประหยัดเทปในการบันทึก เช่นในช่วงเลากลางคืน ในธนาคารจะไม่มีการเข้าออก ดังนั้นเมื่อมีคนเข้าออกหรือมีขโมย ก็จะทำการบันทึกเทป พร้อมกับสั่งให้สัญญาณเตือนภัยดังขึ้น เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
การประมวลผลภาพ

2.1 การประมวลผลภาพเชิงตัวเลข (Digital Image Processing)

การประมวลผลภาพเชิงตัวเลข หมายถึง การนำภาพที่พบทั่วไปมาประมวลผลด้วยเครื่องคอมพิวเตอร์ โดยภาพที่นำมาประมวลผลด้วยเครื่องคอมพิวเตอร์นี้จะถูกแทนที่ให้อยู่ในรูปแบบของเมตริกซ์ แต่ภาพที่ได้โดยส่วนมากแล้วจะเป็นภาพที่ได้รับจากตัวรับสัญญาณ ซึ่งอยู่ในรูปของฟังก์ชัน f(x,y) ที่ต่อเนื่องในระนาบสองมิติ โดยจะเป็นสัดส่วนกับค่าความสว่างหรือความเข้มของภาพที่ตำแหน่ง (x,y) ซึ่งเรียกว่า ระดับสีเทา (Gray Level)

2.1.1 การแทนภาพด้วยข้อมูลแบบดิจิตอล

ภาพข้อมูลแบบดิจิตอล (Digital Image) เป็นภาพที่ถูกแปลงมาจากอนาลอก อยู่ในรูปของตัวเลขโดยภาพอนาลอกถูกแบ่งเป็นพื้นที่สี่เหลี่ยมเล็ก ๆ ที่เรียกว่า พิกเซล (Pixel) ในแต่ละพิกเซลจะถูกระบุตำแหน่งโดย (x,y) และค่าระดับสีเทาของพิกเซล โดยเราสามารถแปลงภาพเป็นข้อมูลแบบดิจิตอลได้มีขั้นตอนและวิธีการดังนี้

เมื่อเรานำสัญญาณอนาลอกที่ต้องการประมวลผลมาผ่านส่วนที่เรียกว่า ดิจิไตเซอร์ (Digitizer) ซึ่งจะมีหน้าที่ในการเปลี่ยนสัญญาณอนาลอกให้เป็นสัญญาณดิจิตอล อุปกรณ์ส่วนนี้ได้แก่ กล้องโทรทัศน์ดิจิไตเซอร์ จากนั้นทำการควอนไตซ์ (Quantizing) เพื่อที่จะประมวลผลสัญญาณด้วยระบบคอมพิวเตอร์ ฟังก์ชันของภาพ f(x,y) จะถูกทำให้เป็นสัญญาณที่ไม่ต่อเนื่องทั้งระนาบของภาพซึ่งเรียกว่า การสุ่มภาพ (Image Sampling) ของฟังก์ชันที่ได้เรียกว่า การควอนไตเซชัน (Gray Level Quantization) ก็จะได้ข้อมูลที่เป็นดิจิตอล

สมมุติว่าสัญญาณภาพที่ต่อเนื่อง f(x,y) ถูกดิจิไตซ์ในระนาบ X และ Y เป็นช่วงเท่า ๆ กัน เราสามารถจัด f(x,y) ให้อยู่ในรูปของเมตริกซ์ ขนาด N × N ได้ดังสมาการที่ 2.1

$$f(x,y) = \begin{Bmatrix} f(0,0) & f(0,1) & f(0,2) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & f(1,2) & \cdots & f(1,N-1) \\ \vdots & \vdots & \vdots & & \vdots \\ f(N-1,0) & f(N-1,1) & f(N-1,2) & \cdots & f(N-1,N-1) \end{Bmatrix} \quad \ldots\ldots(2.1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งทางขวาของสมการเรียกว่า ภาพดิจิตตอล และทุก ๆ สมาชิกของเมตริกซ์จะเรียกว่า พิกเซล จากขบวนการสร้างภาพดิจิตตอลข้างต้น จะเห็นได้ว่าสามารถทราบขนาดของความละเอียดของภาพ N × N พิกเซล และจำนวนระดับของเกรย์สเกลในทางปฏิบัติการทำควอนไตเซชันในระบบภาพดิจิตตอล จะมีค่าดังสมการที่ 2.2

$$B = N \times N \times M \quad \text{บิต} \qquad .....(2.2)$$

เมื่อ B = ขนาดของข้อมูลภาพที่เป็นดิจิตตอล

G = จำนวนของเกรย์สเกลที่ต้องการใช้ในการเก็บข้อมูลภาพ

M = จำนวนบิตที่ใช้ในการแทนข้อมูลภาพ 1 พิกเซล

โดย M สามารถหาได้จาก $G = 2^M$

2.1.2 ลักษณะการจัดเก็บข้อมูลภาพแบบดิจิตตอล

โดยทั่วไปแล้วข้อมูลภาพจะมีค่าความเข้มตั้งแต่ 2 ระดับขึ้นไป แต่ที่ใช้กันมากจะใช้กันที่ค่าระดับความเข้มของจุดภาพเท่ากับ 256 ระดับ ซึ่งจะทำให้ค่าของจุดภาพอยู่ในช่วง 0 – 255 โดยใช้เนื้อที่การเก็บข้อมูลภาพขนาด 1 ไบต์ หรือ 8 บิต สำหรับข้อมูล 1 จุดภาพ ($2^8 = 256$) ในกรณีที่ต้องการภาพที่มีความละเอียดของระดับสูง ๆ อาจจะต้องการจำนวนบิตสำหรับการเก็บข้อมูลมากกว่า 8 บิต คือ อาจจะเป็น 16 บิต หรือ 24 บิต โดยค่าความเข้มของจุดภาพจะเท่ากับ 2^{16} และ 2^{24} โดยจะแยกให้เห็นชัดเจนดังนี้

1. ภาพ 2 ระดับ คือ มีเพียงแค่จุดขาวกับจุดดำเท่านั้น โดยแต่ละจุดภาพเป็นข้อมูลขนาด 1 บิต
2. ภาพ 16 ระดับ คือ ในแต่ละจุดภาพจะมีขนาดของข้อมูล 4 บิต ซึ่งทำให้สามารถแสดงได้ 16 ระดับสี หรือ 16 เกรย์สเกล ขึ้นอยู่กับภาพนั้นเป็นภาพสีหรือภาพขาวดำ
3. ภาพ 256 ระดับ คือ ในแต่ละจุดภาพจะมีขนาดข้อมูล 8 บิต ซึ่งทำให้สามารถแสดงภาพได้ 256 ระดับสี หรือ 256 เกรย์สเกล ขึ้นอยู่กับภาพนั้นเป็นภาพสีหรือภาพขาวดำ
4. ภาพทรูคัลเลอร์ (True Color) คือ ในแต่ละจุดภาพจะมีขนาดของข้อมูล 24 บิต ทำให้สามารถแสดงภาพได้เหมือนภาพจริงมากที่สุด เพราะสามารถแสดงสีได้ถึง 16,777,216 สีภาพทรูคัลเลอร์สามารถแสดงได้เฉพาะภาพสีเท่านั้น ไม่สามารถแสดงภาพขาวดำได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยทั่วไปวิธีการประมวลผลภาพเชิงตัวเลขที่ทำให้คอมพิวเตอร์สามรถรู้จักวัตถุภายในภาพได้นั้น พอจะแบ่งได้สองระดับด้วยกันคือ การประมวลผลภาพในระดับต่ำ (Low Level Image Processing) และการประมวลผลภาพในระดับสูง (High Level Image Processing) การประมวลผลภาพในระดับต่ำจะเป็นการประมวลผลในเชิงตัวเลขเกือบทั้งหมด เพื่อหาตัวแปรต่าง ๆ มาอธิบายข้อมูลภาพ โดยมีจุดประสงค์เพื่อนำตัวแปรเหล่านั้นไปประมวลผลในระดับสูงต่อไป โดยทั่วไปแล้วการประมวลผลภาพระดับต่ำจะประกอบด้วย การประมวลผลภาพก่อน (Preprocessing) การกำจัดสัญญาณรบกวนหรือการทำให้ภาพคมชัด การหาขอบภาพ เป็นต้น

การประมวลผลในระดับสูงเป็นการนำผลลัพธ์ หรือสัญลักษณ์ที่ได้จากการประมวลผลระดับต่ำมาตีความ หรือประมวลผลเพื่อให้คอมพิวเตอร์สามารถรู้จักและเข้าใจภาพได้สำหรับความแตกต่างของการประมวลผลระดับต่ำและระดับสูงนั้นคือ ข้อมูลที่นำมาใช้ในการประมวลผลภาพ โดยที่การประมวลผลภาพระดับต่ำจะใช้ค่าความสว่างของจุดโดยตรง ส่วนการประมวลผลภาพในระดับสูงนั้นข้อมูลของภาพที่นำมาประมวลผลจะถูกแสดงรูปของสัญลักษณ์ ซึ่งสัญลักษณ์เหล่านี้จะแสดงถึงสิ่งต่าง ๆ ที่มีอยู่ในภาพ เช่น ขนาดวัตถุ รูปร่าง และความสัมพันธ์กันระหว่างวัตถุภายในภาพ

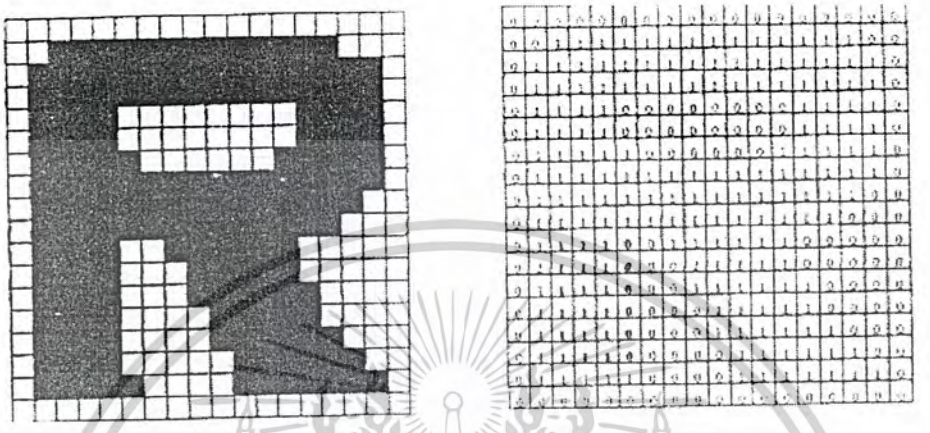
2.2 การสร้างภาพไบนารี่

อุปกรณ์ที่มีความสามารถในการแสดงผลได้แค่ 2 ระดับ หรือ 2 สี คือ สีขาวกับสีดำยังมีการใช้กันอย่างแพร่หลาย เช่น เครื่องพิมพ์(Printer) เครื่องโทรสาร(Fax) จอภาพแสดงผลแบบโมโนโครม (Monochrome Monitor) เป็นต้น เนื่องจากอุปกรนเหล่านี้เป็นอุปกรณ์ที่มีราคาถูก ดังนั้นการที่จะแสดงผลหรือพิมพ์รูปภาพที่มีระดับความเข้มของภาพหลายระดับซึ่งมีมากกว่าความสามารถในการแสดงผลของอุปกรณ์เหล่านี้ที่มีเพียงแค่ 2 ระดับเท่านั้น

จะเห็นได้ว่าการที่จะแก้ปัญหาการแสดงผลภาพที่มีความเข้มหลายระดับบนอุปกรณ์ที่สามารถแสดงผลได้ 2 ระดับนั้นจะต้องทำการแปลงข้อมูลภาพที่มีความเข้มหลายระดับ (Multi Level Image) ให้เป็นภาพที่มีระดับความเข้มเพียง 2 ระดับ นั่นคือ 1 จุดภาพมีได้ 2 ค่าเท่านั้นคือ 0 กับ 1 โดยจุดภาพที่แทนด้วย 0 จะหมายถึงจุดภาพที่มีสีดำ ส่วนจุดที่แทนด้วย 1 จะหมายถึงจุดภาพที่มีสีขาว เมื่อทำการแปลงเป็นภาพไบนารี่แล้วจึงนำภาพนั้นไปแสดงผลบนอุปกรณ์เหล่านั้น จะเห็นได้ว่าการแปลงข้อมูลภาพหลายระดับเป็นภาพไบนารี่จึงมีความจำเป็นและมีประโยชน์มากในการแสดงผลภาพที่มีระดับความเข้มของภาพหลายระดับบนอุปกรณ์ที่มีความสามารถในการแสดงผลได้ 2 ระดับ สำหรับประโยชน์อีกประการหนึ่งในการแปลงข้อมูลภาพนั้นเป็นภาพไบนารี่ คือการลดเนื้อที่การเก็บข้อมูลภาพจะใช้ในการเก็บ 8 บิต เมื่อสร้างเป็นภาพไบนารี่แล้วสามารถลดลงได้ถึง 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เท่า นั่นคือ 1 จุดภาพจะใช้เนื้อที่ในการเก็บ 1 บิต อีกทั้งยังสามารถนำไปประยุกต์ใช้งาได้อย่างแพร่หลายเช่น นำไปประยุกต์ใช้ในการวิเคราะห์เอกสารในขั้นตอนที่เรียกว่า การประมวลผลขั้นต้น (Preprocessing) เป็นต้น



รูปที่ 2.1 ภาพไบนารีและข้อมูลของแต่ละพิกเซล

ในการสร้างภาพไบนารี่สามารถทำได้โดยใช้เทคนิคการทำเทรชโฮล (Thresholding Technique) โดยพิจารณาว่าจุดภาพใดควรจะเป็นจุดขาวหรือจุดดำจะกระทำโดยการเปรียบเทียบระหว่างจุดภาพเริ่มต้นกับค่าคงที่ค่าหนึ่งที่เรียกว่า " ค่าเทรชโฮล " (Threshold Value) เทคนิคนี้ใช้กันมากในกรณีที่ข้อมูลภาพมีลักษณะแตกต่างกันระหว่างวัตถุ (Object) และพื้นหลัง (Background) โดยค่าของจุดภาพใด ๆ ที่มีค่าน้อยกว่าค่าเทรชโฮลจะถูกกำหนดให้เป็น 0 (จุดดำ) และถ้าค่ามากกว่าค่าเทรชโฮลจะถูกกำหนดให้เป็น 1 (จุดขาว)ซึ่งการทำงานสามารถแสดงได้ดังสมการที่ 2.3

$$b(x,y) = \begin{cases} 0 & ; g(x,y) < Thr \\ 1 & ; g(x,y) \geq Thr \end{cases} \qquad \text{.....(2.3)}$$

b(x,y) ข้อทูลภาพผลลัพธ์เป็นภาพไบนารี่

g(x,y) ข้อมูลภาพอินพุทที่มีระดับความเข้ม 0 ถึง 1 ระดับ

Thr ค่าเทรชโฮลเป็นค่าคงที่ที่อยู่ระหว่าง 0 ถึง L ระดับ

0 จุดดำ

1 จุดขาว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ L ระดับความเข้มของจุดภาพสูงสุด

ในการสร้างภาพไบนารี่โดยใช้เทคนิคเทรชโฮลเพื่อให้ได้ผลลัพธ์ที่ได้เหมาะสมและคมชัด สิ่งที่สำคัญที่สุดคือ ค่าเทรชโฮล ถ้าเลือกค่าเทรชโฮลไม่เหมาะสม (ค่าเทรชโฮลที่มีค่าน้อยเกินไป หรือมีค่ามากเกินไป) ภาพที่ได้อาจไม่เหมาะสมขาดความคมชัดและรายละเอียดบางส่วนขาดหายไป กล่าวคือภาพที่ได้อาจจะมืดเกินไป (จุดดำมากเกินไป) หรือสว่างเกินไป (จุดขาวมากเกินไป) หรือภาพที่ได้มีสิ่งรบกวนเกิดขึ้น (Noise) เกิดขึ้น อันเป็นผลให้ภาพผลลัพธ์ที่ได้ไม่สวยงามเท่าที่ควร ดังนั้นปัญหาของการสร้างภาพไบนารี่โดยวิธีเทรชโฮลนี้คือ ทำอย่างไรจึงจะสามารถคำนวณหาค่าเทรชโฮลที่เหมาะสมสำหรับแต่ละภาพที่จะนำมาทำการสร้างภาพไบนารี่ ซึ่งมีวิธีการคำนวณหาค่าเทรชโฮลหลายวิธี โดยแต่ละวิธีเหมาะสมสำหรับการทำงานที่ต่างไป เช่นการหาค่าเทรชโฮลโดยการกำหนดค่าล่วงหน้า (Preassigned Threshold Value) การหาค่าเทรชโฮลจากค่ากลาง (Mid-Rang Threshold Value) แต่ละวิธีอธิบายได้ดังนี้

การหาค่าเทรชโฮลโดยการกำหนดค่าล่วงหน้า (Preassigned Threshold Value) การหาค่าเทรชโฮลโดยวิธีการกำหนดล่วงหน้านี้เป็นวิธีที่ง่ายที่สุด เป็นการคำนวณค่าเทรชโฮลโดยการกำหนดเองจากผู้ใช้ ซึ่งการกำหนดนี้ขึ้นอยู่กับประสบการณ์ของผู้ใช้นั้น ๆ โดยการเลือกค่าคงที่ค่าหนึ่ง ซึ่งเรียกค่านั้นว่า เทรชโฮล โดยค่าที่เลือกมานี้จะเป็นค่าที่อยู่ระหว่างค่าต่ำสุดและค่าสูงสุดของระดับความเข้มของข้อมูลภาพอินพุท เช่น ภาพข้อมูลอินพุทมีเกรย์สเกล 256 ระดับ จะมีค่าเกรย์สเกลได้ตั้งแต่ 0 – 255 เมื่อเลือกค่าเทรชโฮลได้แล้วสามารถภาพไบนารี่ได้ดังสมการที่ 2.3

การหาค่าเทรชโฮลจากค่ากลาง (Mid – Rang Threshold Value) การหาค่าเทรชโฮลโดยพิจารณาจากค่ากลางเป็นการหาค่าเทรชโฮลที่แตกต่างจากการหาค่าเทรชโฮลวิธีแรก สำหรับวิธีนี้จะเป็นการคำนวณหาค่าโดยอัตโนมัติ โดยไม่ต้องให้ผู้ใช้เป็นผู้กำหนดการหาค่าเทรชโฮลวิธีนี้ได้อาศัยการคำนวณพื้นฐานทางสถิติในเรื่องของการหาค่ากลางหรือค่าเฉลี่ย (Mean) มาประยุกต์ใช้ค่าเทรชโฮลที่คำนวณได้จะเป็นค่าที่ได้จากค่ากึ่งกลางที่อยู่ระหว่างค่าระดับความเข้มสูงสุดและค่าความเข้มต่ำสุดของข้อมูลภาพอินพุท สำหรับการคำนวณหาค่ากึ่งกลางนี้สามารถคำนวณได้จากสมการที่ 2.4

$$Thr = \frac{Maximum\ (g(x,y)) + Minimum\ (g(x,y))}{2} \qquad .....(2.4)$$

โดยที่ Thr : ค่าเทรชโฮล

g(x,y) : ข้อมูลภาพอินพุทที่มีระดับความเข้ม 0 ถึง L ระดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Maximum(g(x,y)) : ค่าสูงสุดเกรย์สเกลของข้อมูลอินพุท

Minimum(g(x,y)) : ค่าต่ำสุดเกรย์สเกลของข้อมูลอินพุท

เมื่อทำการคำนวณค่าเทรชโฮลได้แล้วก็สามารถสร้างภาพไบนารี่ได้โดยนำค่าเทรชโฮลที่ได้มาแทนค่าในสมการ 2.3

การหาค่าเทรชโฮลจากค่าเฉลี่ยเลขคณิต หาได้จากสมการที่ 2.5

$$Thr = \frac{\sum_{t=0}^{NxN} gi(x,y)}{N \times N} \quad \text{.....(2.5)}$$

2.3 การแบ่งส่วนภาพ (Image Segmentation)

บางครั้งการวิเคราะห์ภาพจะเน้นที่วัตถุ (Object) บนภาพ ซึ่งวัตถุเหล่านั้นจะถูกแยกแยะออกได้ด้วยวิธีการที่หลากหลายตั้งแต่วิธีการง่ายๆ เช่นการหาส่วนที่เป็นแนวของวัตถุ หรือส่วนที่เป็นขอบของวัตถุ ไปจนกระทั่งวิธีการที่สลับซับซ้อนสำหรับวิธีการง่ายๆในการแบ่งส่วนภาพได้แก่

2.3.1 การตรวจรู้แนวเส้น (Line Detection)

สำหรับวิธีการนี้เป็นการเปรียบเทียบค่าระดับสีเทาระหว่างจุดภาพที่พิจารณากับจุดภาพข้างเคียงทั้งนี้จะเพิ่มน้ำหนักหรือให้ความสำคัญของจุดภาพข้างเคียงที่มีการเรียงตัวร่วมกับจุดภาพที่พิจารณาเป็นเส้นตรงรวม 4 แบบ คือ เส้นตรงในแนวตั้ง เส้นตรงในแนวนอน เส้นตรงในแนวทแยงจากล่างซ้ายขึ้นไปทางบนขวา หรือเรียกว่าเส้นทแยงมุม + 45° และเส้นตรงในแนวทแยงจากบนขวาลงไปทางล่างซ้าย หรือเรียกว่าเส้นทแยงมุม - 45° หมายความว่าแนวเส้นที่จะตรวจรู้เป็นแนวเส้นตั้ง แนวเส้นนอน และเส้นเอียงทำมุม + 45° หรือ - 45° เท่านั้น

กระบวนการตรวจรู้จะทำโดยนำหน้ากาก (mask) ขนาด 3 × 3 (อาจใช้ขนาดใหญ่กว่านี้ เช่น 5 × 5 หรือ 7 × 7 ก็ได้ แต่จะใช้เวลาในการประมวลผลเพิ่มมากขึ้น) ไปทาบกับจุดภาพที่ตำแหน่งที่จะทำการประมวลผล จากนั้นจะคำนวณหาค่าตามสมการที่ (2.6)

$$R = \sum_{t=0}^{9} W_K Z_K \quad \text{.....(2.6)}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ W_K เป็นค่าน้ำหนักของแต่ละจุดบนหน้ากากดังรูปที่ 2.2

Z_K เป็นค่าระดับสีเทาของจุดภาพในตำแหน่งที่ตรงกับจุดบนหน้ากาก กล่าวคือ

$Z_1 = f(x-1,y-1)$ $Z_2 = f(x,y-1)$ $Z_3 = f(x+1,y-1)$

$Z_4 = f(x-1,y)$ $Z_5 = f(x,y)$ $Z_6 = f(x+1,y)$

$Z_7 = f(x-1,y+1)$ $Z_8 = f(x,y+1)$ $Z_9 = f(x+1,y+1)$

W1	W2	W3
W4	W5	W6
W7	W8	W9

-1	-1	-1
2	2	2
-1	-1	-1

-1	2	-1
-1	2	-1
-1	2	-1

-1	-1	2
-1	2	-1
2	-1	-1

2	-1	-1
-1	2	-1
-1	-1	2

ก)หน้ากากทั่วไปข)หน้ากากแนวนอนค)หน้ากากแนวตั้ง ง)หน้ากากแนว+45° จ)หน้ากากแนว-45°

รูปที่ 2.2 หน้ากากที่ใช้ในการหาแนวเส้น

ในการตรวจหาแนวเส้นของภาพ จะนำหน้ากากทั้ง 4 แบบ ไปทาบบนจุดภาพต่าง ๆ จนครบทั้งภาพ แล้วคำนวณตามสมการที่ 2.6 จะได้ค่า R มา 1 ค่า คือ R_1 แล้วเปลี่ยนหน้ากากไปเรื่อย ๆ ตามรูปที่ 2.2 จนครบ 4 อัน จะได้ค่า R_2, R_3 และ R_4 ตามลำดับ จากนั้นทำการเปรียบเทียบค่า R ทั้ง 4 ค่า ถ้า $|R_i| > |R_j|$ ที่ทุกค่าของ $j \neq i$ จุดภาพนั้นจะถูกตัดสินว่าเป็นส่วนหนึ่งของแนวเส้นตามแบบของหน้ากากที่ i

บางครั้งมีความต้องการหาแนวเส้นในลักษณะใดลักษณะหนึ่ง ก็อาจทำได้โดยเลือกใช้หน้ากากแนวที่ต้องการมาทาบบนจุดภาพ และหาผลรวมตามสมการที่ 2.6 จากนั้นนำค่าที่ได้ไปเปรียบเทียบกับค่า เทรชโฮลด์ที่ได้กำหนดไว้แล้ว ถ้าค่าที่ได้สูงกว่าเทรชโฮลด์ จุดภาพนั้นจะถูกตัดสินว่าเป็นส่วนหนึ่งของแนวเส้น

2.3.2 การตรวจหาขอบ (Edge detection)

องค์ประกอบหรือวัตถุที่ถูกบันทึกภาพจะมีตำแหน่งการจัดวาง ที่ทำให้เกิดการสะท้อนแสงที่ต่างกัน ค่าความเข้มแสงหรือค่าระดับสีเทาที่ปรากฏบนภาพจะต้องมีขอบเขตองค์ประกอบนั้น ๆ แสดงให้เห็นชัดเจนหรือค่อนข้างชัดเจน เนื่องจากบริเวณดังกล่าวมีค่าระดับสีเทาแตกต่างกัน ดังนั้นการตรวจหาขอบของวัตถุบนภาพ จึงใช้หลักการพิจารณาการเปลี่ยนแปลงค่าระดับสีเทา ที่มี 2 แนวทาง คือ การพิจารณาอัตราหรือค่าความชันในการเปลี่ยนแปลงค่าระดับสีเทา ซึ่งเรียกว่าวิธีแกรเดียนธ์ (Gradient Method) และการพิจารณาอัตราการเปลี่ยนแปลงค่าความชันในการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปลี่ยนแปลงค่าระดับสีเทา ซึ่งเราเรียกว่า วิธีลาปลาเซียน (Laplacian Method)

2.3.2.1 วิธีแกรเดียนธ์ (Gradient Method *)*

จะใช้กับภาพที่มีองค์ประกอบส่วนใหญ่มีค่าระดับสีเทาไม่สม่ำเสมอตลอดทั้งภาพได้ แต่ความแตกต่างของค่าระดับสีเทาในวัตถุใด ๆ จะมีไม่มาก คือ มีการเปลี่ยนแปลงของระดับสีเทาอย่างช้า ๆ ในวัตถุใดวัตถุหนึ่ง การตัดสินใจว่าบริเวณใดเป็นขอบก็จะขึ้นอยู่กับค่าความชันในการเปลี่ยนแปลงค่าระดับสีเทา ซึ่งจะต้องมีค่ามากกว่าค่าเทรชโฮลด์ (Threshold) ซึ่งภาพที่ได้จะหาขอบได้ดีหรือไม่ก็ขึ้นกับการกำหนดค่าเทรชโฮลด์นั่นเอง

เนื่องจากวิธีแกรเดียนธ์ใช้หลักการในการพิจารณาการเปลี่ยนแปลงของค่าระดับสีเทาของภาพ ถ้ามีการเปลี่ยนแปลงมาก ก็จะถูกกำหนดให้เป็นขอบของวัตถุ ทั้งนี้ค่าแกรเดียนธ์ของภาพ f(x,y) ณ ตำแหน่ง (x,y) จะเป็นค่าเวกเตอร์

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \qquad \text{.....(2.7)}$$

โดยที่ขนาดของค่าเวกเตอร์ ∇f เป็นปริมาณที่มีความสำคัญในการหาขอบของวัตถุ โดยทั่ว ๆ ไปจะเรียกว่าค่าแกรเดียนธ์ และใช้สัญญาลักษณ์ ∇f ที่มีค่า

$$\nabla f = \text{mag}(\nabla f) = (G_x^2 + G_y^2)^{1/2} \qquad \text{.....(2.8)}$$

ในทางปฏิบัติ เพื่อให้ง่ายต่อการคำนวณ จะทำการประมาณค่าแกรเดียนธ์ด้วยค่าสัมบูรณ์

$$\nabla f \approx |G_x| + |G_y| \qquad \text{.....(2.9)}$$

การหาค่าอนุพันธ์ย่อย G_x และ G_y ทำได้โดยการใช้หน้ากาก (mask) หรือหน้าต่าง (window) หรือตัวดำเนินการ (operator) ขนาด 3 X 3 ที่มีค่าสัมประสิทธิ์ดังแสดงรูปที่ 2.3 ก. เข้ามาช่วยปรับค่าระดับสีเทา ณ จุดภาพที่สอดคล้องกัน ซึ่งตัวดำเนินการมีหลายแบบดังนี้

2.3.2.1.1 ตัวดำเนินการแบบโรเบิร์ตครอสส์แกรเดียนธ์ (Robert – cross gradient operator)

ตัวดำเนินการแบบโรเบิร์ตครอสส์แกรเดียนต์ พิจารณาเป็นหน้ากากขนาด 3 X 3 ดังรูปที่ 2.3 ข ที่ให้ค่าอนุพันธ์ย่อยเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$G_x = |Z_5 - Z_9| \quad \text{.....(2.10ก)}$$

$$G_Y = |Z_6 - Z_8| \quad \text{.....(2.10ข)}$$

2.3.2.1.2 ตัวดำเนินการแบบโซเบล (Sobel Operator)

ตัวดำเนินการแบบโซเบล มีหน้ากากขนาด 3 X 3 ดังรูปที่ 2.3 ค. ทำให้ได้ค่าอนุพันย่อยเป็น

$$G_x = (Z_7 + 2Z_8 + Z_9) - (Z_1 + 2Z_2 + Z_3) \quad \text{.....(2.11ก)}$$

$$G_Y = (Z_3 + 2Z_6 + Z_9) - (Z_1 + 2Z_4 + Z_7) \quad \text{.....(2.11ข)}$$

2.3.2.1.3 ตัวดำเนินการแบบพรีวิตต์ (Prewitt Operator)

ตัวดำเนินการแบบพรีวิตต์ มีหน้ากากขนาด เช่นเดียวกับตัวดำเนินการโซเบลแต่มีค่าสัมประสิทธิ์แตกต่างกันดังรูปที่ 2.3 ง. ทำให้ได้ค่าอนุพันธ์ย่อยเป็น

$$G_x = (Z_7 + Z_8 + Z_9) - (Z_1 + Z_2 + Z_3) \quad \text{.....(2.12ก)}$$

$$G_Y = (Z_3 + Z_6 + Z_9) - (Z_1 + Z_4 + Z_7) \quad \text{.....(2.11ข)}$$

W_1	W_2	W_3
W_4	W_5	W_6
W_7	W_8	W_9

ก) หน้ากากขนาด 3 X 3 ที่จะใช้เป็นตัวดำเนินการ

0	0	0
0	1	0
0	0	-1

หน้ากากที่ใช้คำนวณค่า G_x

0	0	0
0	0	1
0	-1	0

หน้ากากที่ใช้คำนวณค่า G_Y

ข) ตัวดำเนินการแบบโรเบิร์ตครอสส์แกรเดียนธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-1	-2	-1
0	0	0
1	2	1

หน้ากากที่ใช้คำนวณค่า G_x

-1	0	1
-1	0	1
-1	0	1

หน้ากากที่ใช้คำนวณค่า G_y

ค) ตัวดำเนินการแบบโซเบล

-1	-1	-1
0	0	0
1	1	1

หน้ากากที่ใช้คำนวณค่า G_x

-1	0	1
-1	0	1
-1	0	1

หน้ากากที่ใช้คำนวณค่า G_y

ง) ตัวดำเนินการแบบพรีวิตต์

รูปที่ 2.3 ตัวดำเนินการที่ใช้ในการคำนวณหาค่าแกรเดียนธ์

2.3.2.2 วิธีลาปลาเซียน (Laplacian Method)

วิธีลาปลาเซียน ใช้หลักการพิจารณาการเปลี่ยนแปลงค่า Gradient ของค่าระดับสีเทาอีกต่อหนึ่ง และจะกำหนดตำแหน่งที่มี Zero – Crossing เกิดขึ้นเป็นตำแหน่งของขอบของวัตถุทั้งนี้ การหาอนุพันธ์ย่อยอันดับสองของภาพ f(x,y) คือการหาค่า Laplaciean แบบ 2 มิติ เป็นต้น

$$\nabla^2 f = \frac{\partial^2 f}{\partial^2 x} + \frac{\partial^2 f}{\partial^2 y} \qquad \text{.....(2.13ก)}$$

หน้ากากของวิธีลาปลาเซียนขนาด 3 X 3 จะมีค่าสัมประสิทธิ์แตกต่างกันหลายแบบดังรูปที่2.4 และตัวอย่างการใช้ตัวดำเนินการแบบ 4 – neighbor เพื่อหาค่าอนุพันธ์ย่อยอับดับสองของจุดภาพ f(x,y) คือ

$$\nabla^2 f = 4Z_5 - (Z_2 + Z_4 + Z_6 + Z_8) \qquad \text{.....(2.13ข)}$$

0	-1	0
-1	4	-1
0	-1	0

ก) แบบ 4 – neighborhood

-1	-1	-1
-1	8	-1
-1	-1	-1

ข) แบบ 8 – neighborhood

-2	1	-2
1	4	1
-2	1	-2

ค)แบบseperable4– neighborhood

รูปที่ 2.4 หน้ากากที่ใช้ในตัวดำเนินการลาปลาเซียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2.3 การหาขอบแบบแคนนี (Canny edge detection)

การหาขอบแบบแคนนีเป็นการดัดแปลงวิธีแกรเดียนธ์ได้เหมาะสมกับการตรวจตราหาขอบขององค์ประกอบในภาพที่มีสัญญาณรบกวนแต่อยู่บนเงื่อนไข3 ประการดังนี้

ก) ไม่เกิดความผิดพลาดในการหาขอบที่สำคัญ และไม่ควรมีการหาขอบที่ผิดเกิดขึ้น

ข) การบอกตำแหน่งของขอบ จะต้องมีระยะระหว่างตำแหน่งจริงและตำแหน่งที่ห่างขอบสั้นที่สุด

ค) ถ้ามีค่าหลายค่าในการหาขอบของขอบใดขอบหนึ่งค่าที่น้อยที่สุดจะเป็นค่าของขอบที่แท้จริง เพื่อแก้ปัญหาสัญญาณรบกวน ซึ่งจะทำให้ได้ขอบที่ไม่เรียบ

สำหรับขั้นตอนวิธรการหาขอบแบบแคนนี มี 4 ขั้นตอน คือ

ขั้นที่ 1 ใช้ตัวกรองไบโนเมียล (Binomial Filter) เพื่อลดทอนสัญญาณรบกวนดังสมการที่ 2.14

$$f(x,y) = \left(\frac{1}{16} \begin{pmatrix} 1 & 2 & 2 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix} \right)^{n} \qquad \text{.....(2.14)}$$

โดยที่ n = 2 เท่าของความแปรปรวนของการกระจายแบบเกาส์ (Gaussian distribution)

ขั้นที่ 2 ทำการคำนวณแกรเดียนธ์ตามสมการที่ (2.10) และใช้ตัวดำเนินการแบบพรีวิตต์ดังรูปที่ 2.3 ง. จะได้

$$\nabla f = \left|(Z_7+Z_8+Z_9) - (Z_1+Z_2+Z_3)\right| + \left|(Z_3+Z_6+Z_9) - (Z_1+Z_4+Z_7)\right| \qquad \text{....(2.15)}$$

ขั้นที่ 3 ทำการหาว่าจุดภาพมีค่าแกรเดียนธ์เฉพาะถิ่นสูงสุด (Maximum local gradient) หรือไม่ สมมติค่าแกรเดียนธ์ของแต่ละจุดภาพในกรอบขนาด 3 x 3 มีค่าเป็น

P1	P2	p3
P4	P5	P6
P7	P8	P9

ทำการพิจารณาเปรียบเทียบค่าแกรเดียนธ์ของจุดภาพกลางกับจุดภาพอื่น ๆ ในแนวเส้นตรงต่าง ๆ ประกอบด้วย แนวตั้ง แนวนอน แนวทแยง +45° และ - +45°

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้า P5 > Pk ทุกค่า k ≠ 5 และ P5 > ค่าเทรชโฮลด์ต่ำ (low threshold value) T_L แสดงว่าค่า P5 เป็นค่าแกรเดียนธ์เฉพาะถิ่นสูงสุดและจะนำค่าดังกล่าวไปคำนวณตามขั้นตอนที่ 4 ต่อ

ขั้นที่ 4 การทำเทรชโฮลด์แบบฮีสเทอรีซีส (Hysteresis Threshold)

ขั้นตอนสุดท้ายนี้ เป็นขั้นตอนการตัดสินใจว่าจุดภาพหนึ่ง ๆ เป็นขอบขององค์ประกอบหรือไม่ มีเงื่อนไข 3 ประการดังนี้

- จุดภาพที่มีขนาดของแกรเดียนธ์ใหญ่กว่าค่าเทรชโฮลด์สูง (High threshold value) T_H จุดภาพนั้นจะเป็นขอบขององค์ประกอบ
- ทุกจุดภาพที่จะประกอบกันเป็นแนวขอบแนวหนึ่ง จะต้องมีอย่างน้อย 1 จุดภาพในแนวนั้นที่มีค่าแกรเดียนธ์ > T_H
- จุดภาพที่มีค่าแกรเดียนธ์ < T_H และไม่สอดคล้องกับเงื่อนไขที่สองจะถูกตัดไป

2.4 ทฤษฎีการแยกตัวอักขระออกจากภาพ

2.4.1 การหากรอบตัวอักขระโดย วิธี Line Crossing

เทคนิคนี้จะมีหลักการง่าย ๆ คือ พยายามแบ่งตัวอักขระแต่ละตัวออกมาให้ได้มากที่สุด โดยเริ่มจากการสแกนเป็นแนวตรง ทั้งแนวนอนและแนวตั้ง ตัดไปตามช่องว่างทำให้เกิดเป็นตารางครอบตัวอักขระ ซึ่งตารางที่เล็กที่สุดที่ครอบตัวอักขระอยู่ถือว่าเป็นการ segment ได้หนึ่งตัวอักษรดังรูปที่ 2.5

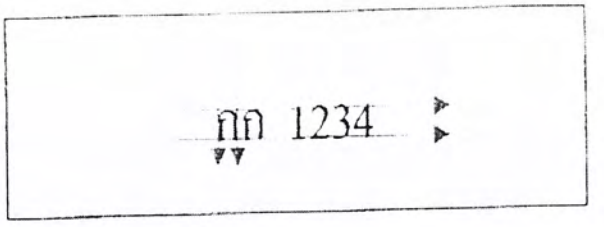
แนวคิดของวิธีการ Line Crossing ดังรูปที่ 2.5 มีขั้นตอนดังต่อไปนี้

1. สแกนเป็นเส้นตรงตามแนวนอน
2. สแกนเป็นเส้นตรงตามแนวตั้ง

(ขั้นตอนที่1 กับ 2 สามารถที่จะทำขั้นตอนใดก่อนก็ได้)

3. คัดลอกข้อมูลที่อยู่ในกรอบที่มีอักขระเก็บไว้ใน buffer เพื่อนำไปแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 แสดงการแยกตัวอักขระด้วยวิธี Line Crossing

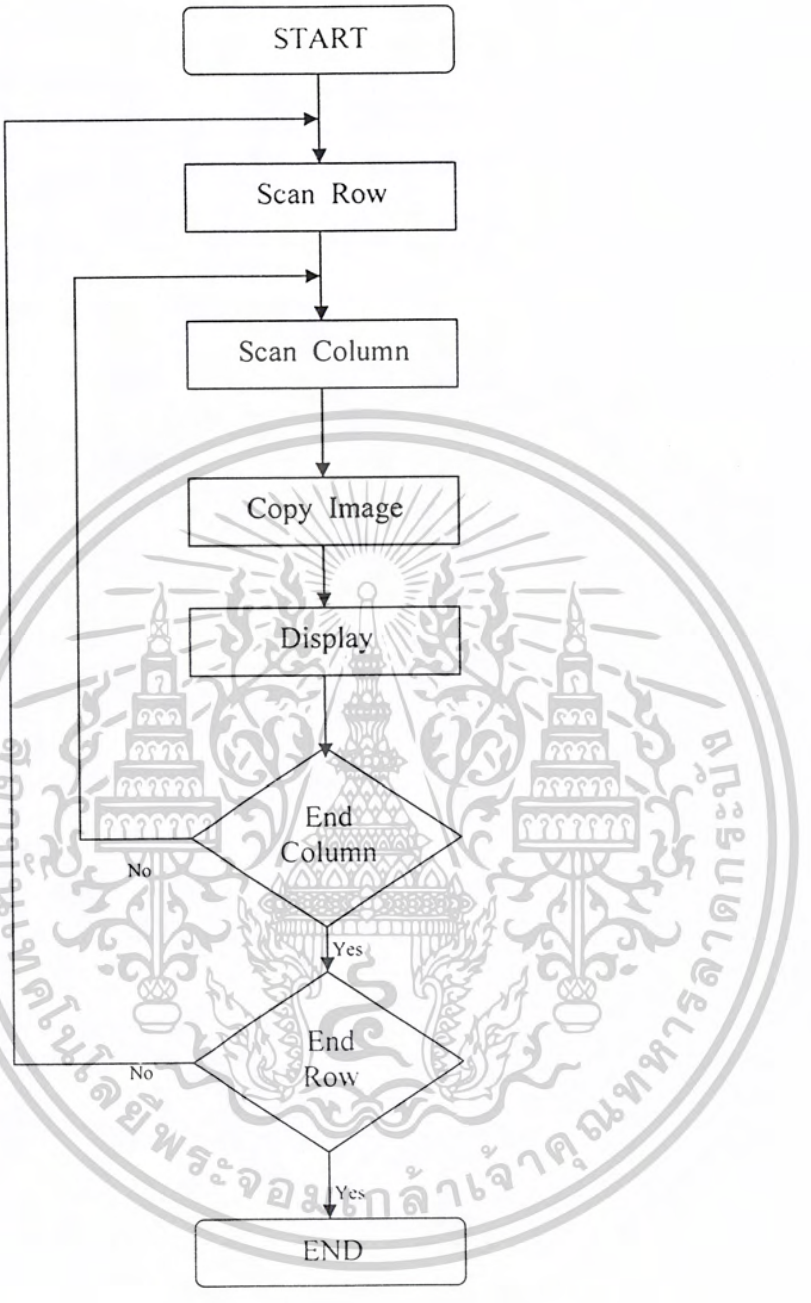
อธิบาย Flowchart การแยกตัวอักขระด้วยวิธี Line Crossing

1. Scan Row : หาช่องว่างในแนวนอน
2. Scan Column : หาช่องว่างในแนวตั้ง
3. Copy Image : คัดลอกข้อมูลส่วนที่ตัดได้
4. Display : แสดงรูปทำการตัดได้
5. End Column : การ Scan ข้อมูลในแนวตั้งครบหมด
   - Yes : End Row
   - No : Scan Column
6. End Row : การ Scan ข้อมูลในแนวนอนครบ
   - Yes : End Row
   - No : Scan Column

ข้อดีของการทำ Line Crossing ก็คือ ง่ายต่อการเขียนโปรแกรม และมีประสิทธิภาพในการแยกอักขระกับแบบอักขระที่มีข้อผิดพลาดน้อย ตัวอักขระไม่ติดกันมาก และถ้าเป็นการ Segment ตัวอักขระภาษาอังกฤษจะได้ผลลัพธ์ที่ดีกว่าอักขระภาษาไทย

ข้อเสียของวิธีการนี้ที่เห็นได้ชัดก็คือ ในการทำการแยกตัวอักขระที่มีการเหลื่อมล้ำกันจะไม่สามารถใช้เทคนิคในการแยกนี้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



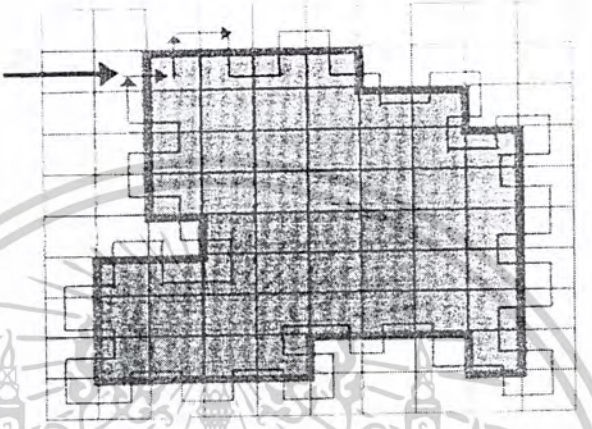
รูปที่ 2.6 แสดง Flow Chart การแยกตัวอักขระด้วยวิธี Line Crossing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.4.2 เทคนิคการตามรอยขอบภาพ (Contour Following)

เทคนิคการตามรอยขอบภาพจะช่วยให้เราทำการ segment ตัวอักษรได้ง่ายขึ้น โดยไม่ต้องคำนึงถึงรูปร่างของวัตถุที่จะทำการ Segment เมื่อพิจารณาขอบของวัตถุจะมีค่าสีที่มีความแตกต่างกันกับสีของพื้นหลัง เช่น ภาพตัวอักขระสีดำที่วางอยู่บนพื้นกระดาษสีขาว



รูปที่ 2.7 แสดงการหาขอบด้วยวิธี Contour Following

การทำงานของเทคนิคนี้ในการทำ Contour Following ตามรูปที่ 2.7 ทำได้ตามขั้นตอนต่อไปนี้

1. จุดภาพปัจจุบันมีข้อมูลเป็นดำ ให้เลี้ยวซ้ายจากทิศทางปัจจุบัน
2. จุดภาพปัจจุบันมีข้อมูลเป็นสีขาว ให้เลี้ยวขวาจากทิศทางปัจจุบัน
3. สิ้นสุดเมื่อจุดภาพปัจจุบันอยู่ตำแหน่งเดียวกับจุดเริ่มต้นพอดี

อธิบาย Flow Chart การแยกอักขระด้วยวิธีการ Contour Following จากรูปที่ 2.8

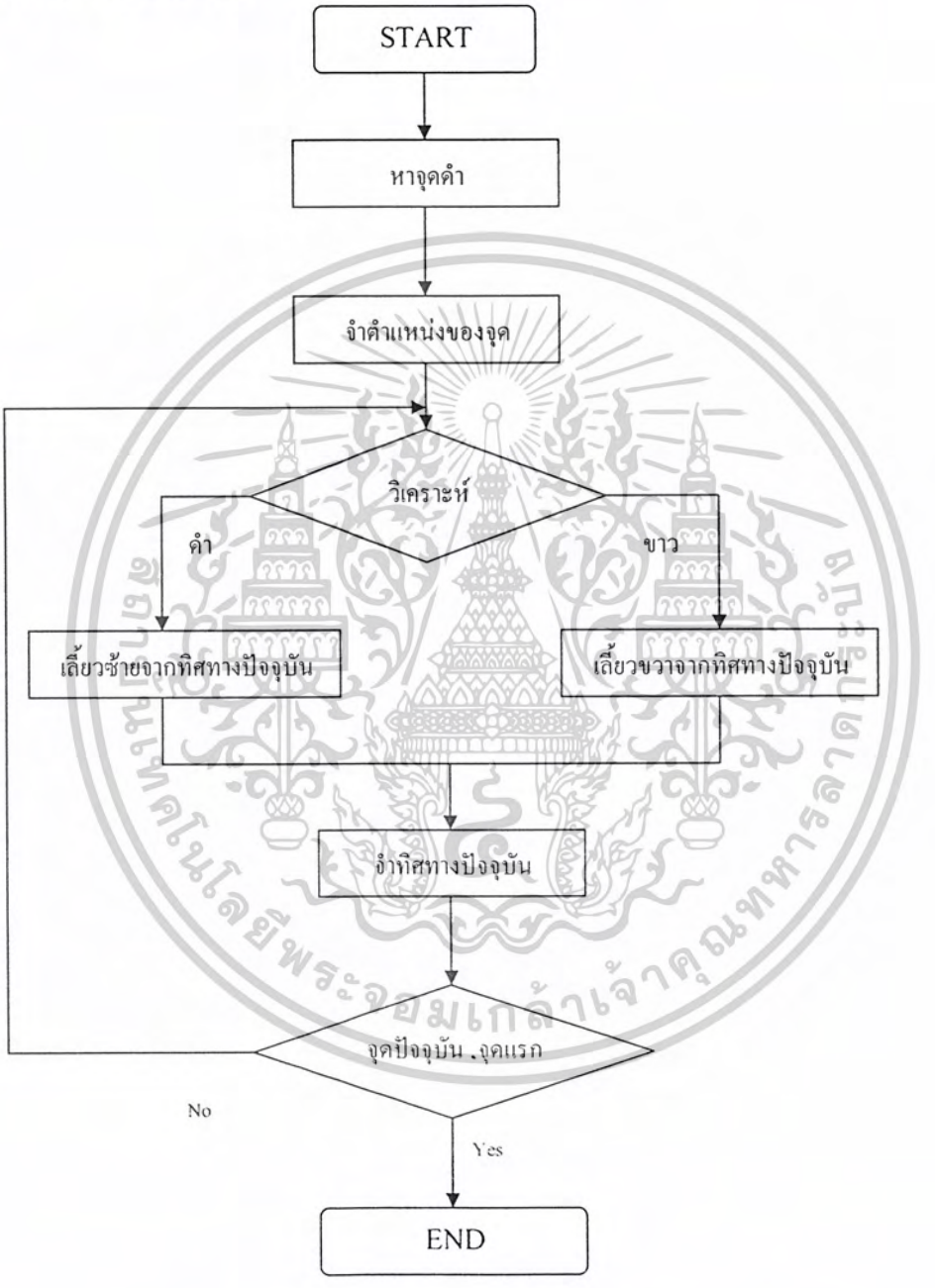
1. หาจุดดำ : เป็นการหาจุดดำจุดแรกที่พบ โดยการ scan ตามแนวนอน
2. จำตำแหน่งของจุด : เป็นการจำตำแหน่งของจุดดำที่พบ
3. วิเคราะห์จุด : มี 2 กรณี
   - ขวา : เลี้ยวขวาจากทิศทางปัจจุบัน
   - ดำ : เลี้ยวซ้ายจากทิศทางปัจจุบัน
4. จำทิศทางปัจจุบัน : จดจำทิศทางและตำแหน่งปัจจุบัน
5. จุดปัจจุบัน = จุดแรก : เป็นการเปรียบเทียบตำแหน่งปัจจุบันจุดเริ่มต้น
   - Yes : End
   - No : วิเคราะห์จุด

64205

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อดีของการทำวิธี Contour Following ก็คือ เราไม่ต้องสนใจว่ารูปร่างของตัวอักขระจะมีลักษณะเป็นอย่างไร เมื่อเราวิ่งไปตามขอบของภาพ เราก็จะได้ขอบของตัวอักขระออกมา

ข้อเสียของการทำวิธีนี้ก็คือ ถ้าตัวอักขระมีส่วนที่ติดกันอยู่จะไม่สามารถใช้วิธีนี้ในการแยกตัวอักขระออกจากกัน



รูปที่ 2.8 แสดง Flow Chart การหาขอบด้วยวิธี Contour Following

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 เทคนิคการแยกโดยวิธีการ Contour with matrix

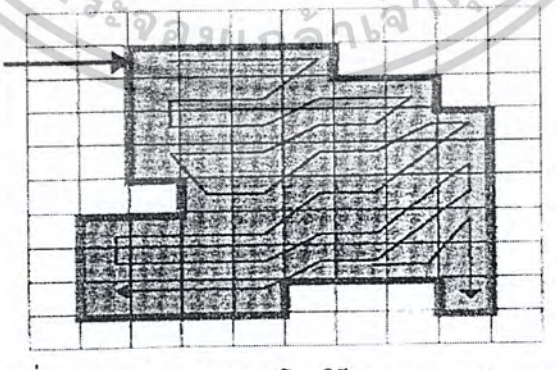
เทคนิคนี้คล้ายกับการทำ Contour Following แต่จะแตกต่างกันตรงที่เทคนิคนี้จะมีการดึงเอาข้อมูลภาพที่อยู่ในขอบเขตของการทำ Contour ไปใช้ด้วย ทำให้เราได้ตัวอักขระทั้งตัวไปใช้งาน และการใช้งานไม่เพียงแต่ดูว่าข้อมูลตรงจุดนั้นเป็น 0 หรือ 1 แต่ใช้ matrix ช่วยตรวจสอบแทน matrix ที่ใช้มีขนาด 3 X 3 ดังรูปที่ 2.9

P1	P2	P3
P4	P5	P6
P7	P8	P9

รูปที่ 2.9 เมตริกซ์ขนาด 3 X 3

แนวคิดของการทำ Contour with matrix ดังรูปที่ 2.10 มีขั้นตอนการทำงานดังนี้

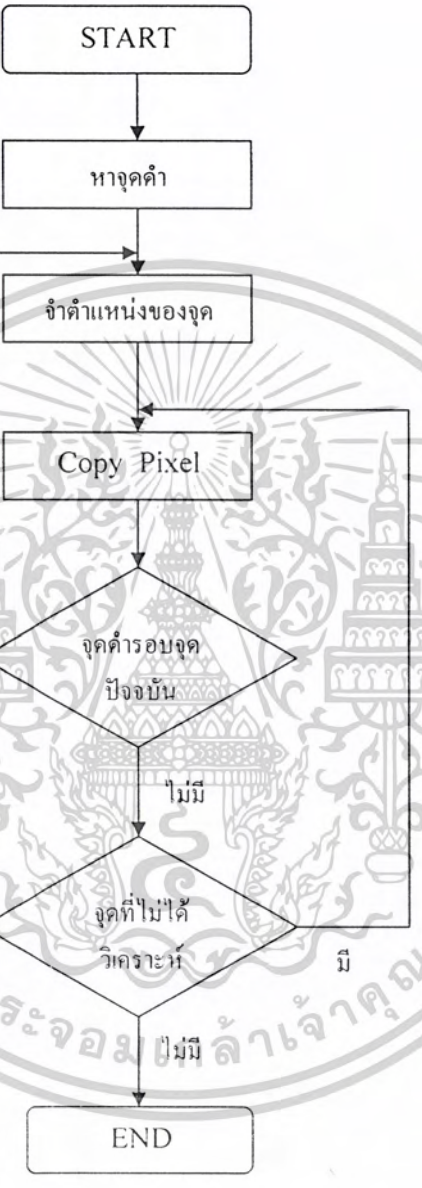
1. ให้กำหนดจุดที่ดำที่พบเป็นจุดกลางของ matrix (P5) คัดลอกจุดนี้ลงใน buffer แล้วลบจุดกลางนั้นออกจากรูปภาพ
2. หาจุดกลางถัดไป โดยตรวจสอบข้อมูลรอบ ๆ จุดกลางปัจจุบัน (P5) ซึ่งเป็นจุดสีดำ คือ ข้อมูลตัวอักขระ การตรวจสอบจะเริ่มจากด้านบนซ้ายก่อน คือ ตำแหน่ง P1 ,P2 ,P3 ,P4 ,P5 ,P6 ,P7 ,P8 และ P9 ตามลำดับ กำหนดจุดดำที่พบจุดแรกให้เป็นจุดกึ่งกลาง matrix
3. เมื่อรอบ ๆ จุดกลางนั้นไม่มีจุดดำแล้วจะทำการ return กลับไปยังจุดกลางตัวก่อน
4. การทำงานจะเป็นเช่นนี้ไปเรื่อย ๆ จนกระทั่งไม่มีจุดดำรอบจุดศูนย์กลางทุกตัว



รูปที่ 2.10 แสดงการหาขอบโดยวิธี Contour with matrix

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการทำงานด้วยวิธีนี้แสดงดังรูปที่ 2.11 เส้นทางการวิ่งจะเป็นไปตามลูกศรที่แสดงในรูป ซึ่งรูปตัวอย่างเป็นรูปเดียวกับตัวอย่างที่แสดงในวิธีการตามรอยขอบภา (Contour Following)



รูปที่ 2.11 แสดง Flow Chart แสดงการหาขอบโดยวิธี Contour with matrix

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อธิบาย Flow Chart การแยกอักขระด้วยวิธีการ Contour with matrix จากรูปที่ 2.11

1. หาจุดดำ : เป็นการหาจุดดำจุดแรกที่พบโดยการ Scan ตามแนวนอน
2. จำตำแหน่งของจุด : จำตำแหน่งจุดดำที่พบ
3. Copy Pixel : จะทำการคัดลอกจุดที่พบไว้ใน buffer ทำการลบจุดที่พบออกจากรูปภาพ
4. จุดดำรอบจุดปัจจุบัน : ตรวจว่ารอบจุดปัจจุบันมีจุดสีดำล้อมรอบหรือไม่
   - มี : จำตำแหน่งของจุด
   - ไม่มี : จุดที่ไม่ได้วิเคราะห์
5. จุดที่ไม่ได้วิเคราะห์ : ตรวจว่ามีจุดที่ยังไม่ได้ทำการวิเคราะห์
   - มี : Copy Pixels
   - ไม่มี : End

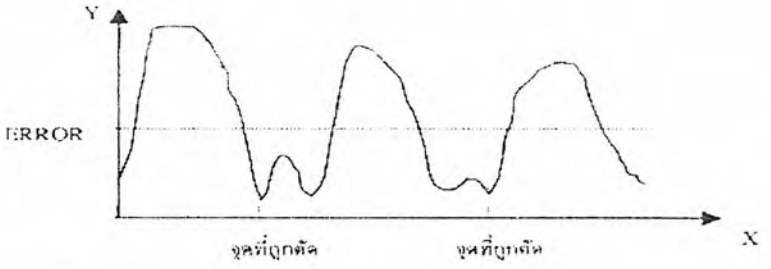
ข้อดีของวิธี Contour with matrix ก็คือ สามารถดึงเอาส่วนที่เป็นเนื้อตัวอักขระออกมาได้ทั้งหมด เป็นประโยชน์ต่อการนำไปใช้ในกระบวนการอื่น ๆ ที่เนื้อของข้อมูลเป็นสิ่งจำเป็น

ข้อเสียของวิธีนี้ก็คือ เมื่อเราต้องดึงเอาข้อมูลทุกส่วนของตัวอักขระออกมาก็จะทำให้เราต้องใช้เวลานานกับการคัดลอกและลบข้อมูล วิธีการนี้ทำงานแบบ Recursive ถ้ามีการ backtracking มากเกินไปอาจทำให้ stack overflow ได้

2.4.4 เทคนิคการแยกตัวอักขระที่ติดกันโดยHistogram

Histogram ใช้เพื่อวัดความหนาแน่นของข้อมูลภาพในช่วงที่กำหนด โดยค่าความหนาแน่นที่ใช้เป็นจำนวนจุดดำ เพื่อช่วยในการวิเคราะห์ภาพตัวอักขระ โดยเก็บเปอร์เซ็นต์ Error เพื่อนำไปใช้เปรียบเทียบกับตัวอักขระแต่ละตัว เพื่อหาตัวอักขระที่คาดว่าจะติดกันอยู่ และการใช้ histogram จะช่วยให้โปรแกรมสามารถตัดสินใจได้ว่าควรที่จะตัดส่วนที่ติดกันของอักขระที่ตรงส่วนไหน เพื่อให้สามารถแยกตัวอักขระที่ติดกัน ให้เป็นตัวอักขระเดี่ยวให้ได้มากที่สุด รูปที่ 2.12 แสดงการเลือกจุดที่จะทำการแยกตัวอักขระออกจากกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 แสดงการกำหนดจุดตัดด้วย Histrogram

การแยกส่วนที่ติดกันของตัวอักขระ โดยใช้ Histrogram มีวิธีการดังนี้

1. ทำการหาค่า histrogram ของตัวอักขระที่สามารถแยกออกมาจากรูปภาพได้
2. กำหนดค่าเปอร์เซ็นต์ Error เพื่อใช้ในการแยกตัวอักขระออกจากกัน
3. ทำการแยกตัวอักขระตรงส่วนที่มีค่าความหนาแน่นเท่ากับ หรือต่ำกว่าค่าเปอร์เซ็นต์ error ที่ได้กำหนดไว้ โดยการเลือกแยกตัวอักขระตรงส่วนที่มีค่าความหนาแน่นต่ำที่สุด
4. ถ้าค่าความหนาแน่นที่ต่ำที่สุดมีหลายตำแหน่ง จะใช้ตำแหน่งที่อยู่ตรงกลาง

อธิบายFlowchart การแยกตัวอักขระที่ติดกันด้วยการใช้ histrogram จากรูปที่ 2.13

1.กำหนดค่าError:เป็นการกำหนดค่าเปอร์เซ็นต์errorเพื่อใช้ในการแยกตัวอักขระ

2.หาค่า histrogram ในแนวนอน : หาค่าของ histrogram โดยหาค่าความหนาแน่นในแนวนอน

3.หาจุดตัดในแนวนอน:เลือกตำแหน่งในการแยกตัวอักขระออกจากกันในแนวนอน

4.หาค่า histrogram ในแนวตั้ง : หาค่าของ histrogram โดยหาค่าความหนาแน่นในแนวตั้ง

5.หาจุดตัดในแนวตั้ง : เลือกตำแหน่งในการแยกตัวอักขระออกจากกันในแนวตั้ง

6.Copy Image : คัดลอกข้อมูลของส่วนที่ตัดได้

7.Display : แสดงรูปที่ทำการตัดได้

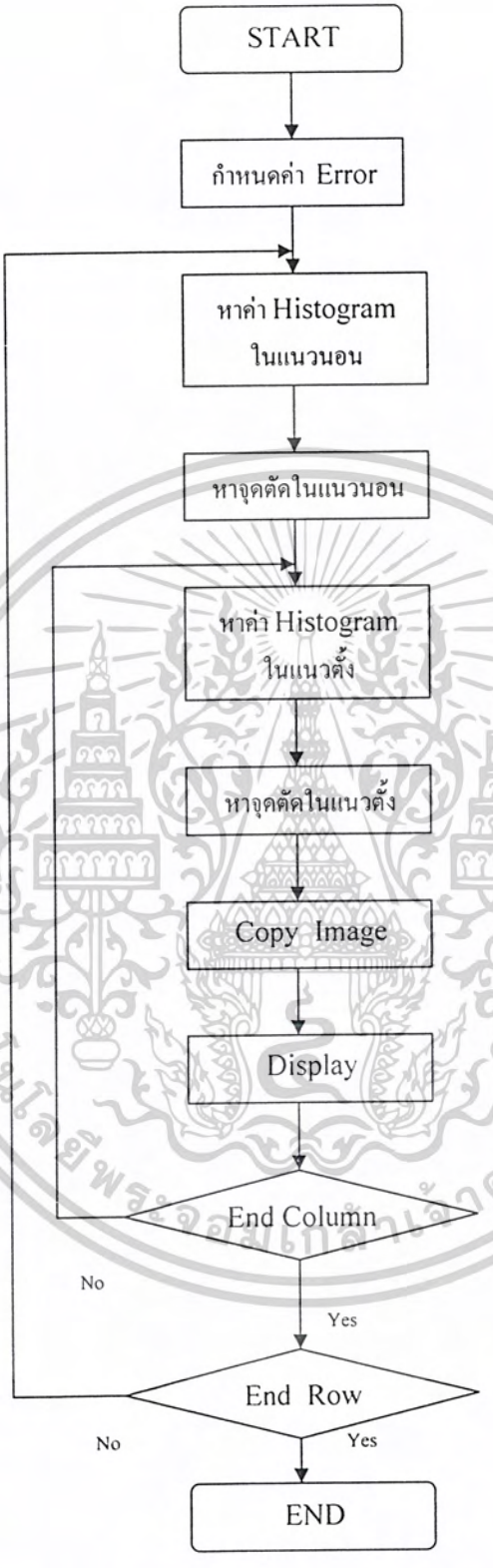
8.End Column : การ scan ข้อมูลในแนวตั้งครบหมด

- Yes : End Row
- No : หาค่า histrogram ในแนวตั้ง

9.Ehd Row : การ scan ข้อมูลในแนวนอนครบหมด

- Yes : End
- No : หาค่า histogram ในแนวนอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 แสดง Flow Chart การแยกตัวอักขระที่ติดกันโดยการใช้ Histogram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 โมเมนท์เกี่ยวกับระยะทางของภาพแบบดีสครีต (Discrete Image Spatial Moment)

แนวคิดของ Moment สามารถนำไปใช้กับภาพดีสครีต F(j,k) โดยการใช้เครื่องหมายการรวม(Summation) โมเมนท์หรือMoment ที่ $(m,n)^{th}$ ของภาพดีสครีต

$$M_U(m,n) = \sum_{j=1}^{J} \sum_{k=1}^{K} (x_k)^m (y_j)^n F(j,k) \qquad .....(2.18)$$

โดยที่เพื่อเป็นการเปลี่ยนให้จุดกำเนิดของโคออร์ดิเนตอยู่ที่มุมล่างซ้าย

$$x_k = k - 0.5 \qquad .....(2.18ก)$$

$$y_j = J + 0.5 - j \qquad .....(2.18ข)$$

เราเห็นว่า Moment มีค่าขึ้นอยู่กับขนาดอย่างมาก เพื่อเป็นการลดผลของขนาดภาพ เราได้นิยาม Scaled Moment ดังนี้

นั่นคือ

$$M(m,n) = \frac{M_U(m,n)}{J^n K^m} \qquad .....(2.19)$$

ถ้าพิจารณา Zero – order Moment

$$M(0,0) = \sum_{j=1}^{J} \sum_{k=1}^{K} F(j,k) \qquad .....(2.20)$$

เราได้ว่า Zero- order Moment คือผลรวมของค่าพิกเซ็ลของภาพ ถ้าภาพเป็นภาพขาวดำ (Binary) Zero - order Moment คือพื้นที่

ตารางที่ 2.1 แสดง Scaled Moment ของภาพทดสอบหลาย ๆ ภาพ ซึ่งเป็นตัวอักษรตามรูปที่ 2.14

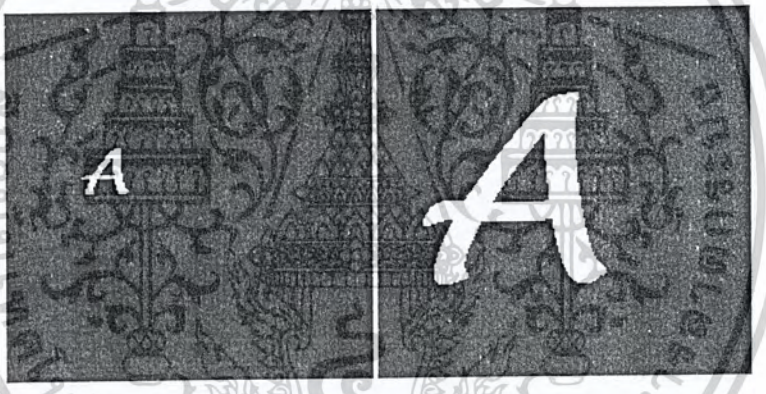
$$\overline{x_k} = \frac{M(1,0)}{M(0,0)} \qquad .....(2.21ก)$$

$$\overline{Y_j} = \frac{M(0,1)}{M(0,0)} \qquad .....(2.21ข)$$

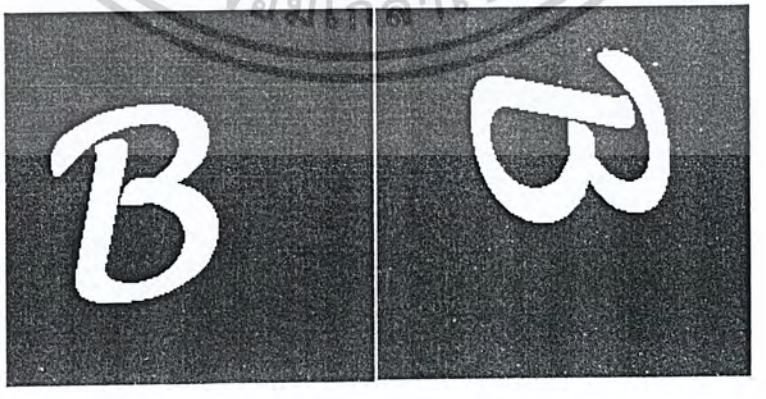
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Image	Mu[0,0] $*10^{-1}$	Mu[1,0] $*10^{-3}$	Mu[0,1] $*10^{-3}$	Mu[2,0] $*10^{-5}$	Mu[0,2] $*10^{-5}$	Mu[1,1] $*10^{-5}$
A	20.6	17.732	32.485	15.344	51.31	27.983
Scalling A	354.7	366.681	472.269	402.693	652.332	494.439
B	446.4	441.725	624.228	478.501	898.99	617.898
Rotate B	446.4	322.14	450.653	258.563	496.349	325.002
Rotate C 90°	148.2	224.18	102.901	343.96	79.752	154.412
Rotate C 180°	34.4	25.987	53.81	19.92	84.678	40.587

ตารางที่ 2. 1 แสดง Scaled Spatial Moment of Test Image

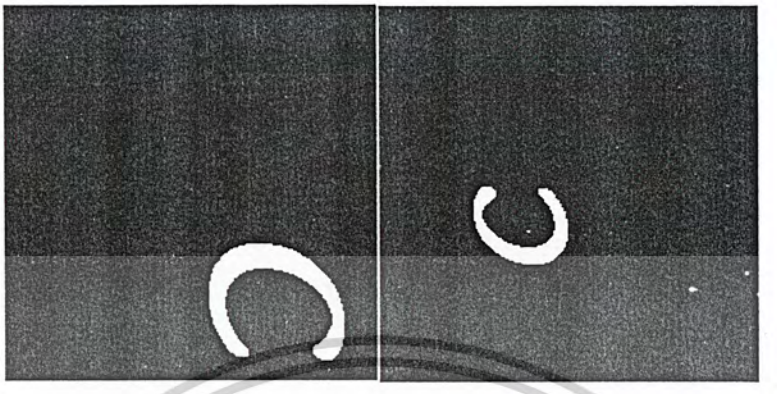


a) อักษร A b) Scaling A



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

c) อักษร B d) Rotate B



e) Rotate C 90° f) Rotate C 180°

รูปที่ 2.14 สัญญาลักษณ์ของตัวอักษรที่ถูกหมุนขยายและย่อ

เมื่อเรานิยาม Centroid แล้ว ต่อไปเราสามารถนิยาม Scaled Center Moment ของภาพได้ดังนี้

$$U_U(m,n) = \sum_{j=1}^{J} \sum_{k=1}^{K} (x_k - \bar{x}_k)^m (y_j - \bar{y}_j)^n F(j,k) \quad .....(2.22)$$

เพื่อเป็นการอ้างอิงในอนาคต เรานิยาม Unscaled Central Moment

$$\bar{x}_k = \frac{M_U(1,0)}{M_U(0,0)} \quad .....(2.23ก)$$

$$\bar{y}_j = \frac{M_U(0,1)}{M_U(0,0)} \quad .....(2.23ข)$$

เราได้ว่า

$$U(m,n) = \frac{M_U(m,n)}{J^n K^m} \quad .....(2.24)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Image	Uu[0,0]	Uu[1,0]	Uu[0,1]	Uu[2,0]	Uu[0,2]	Uu[1,1]
A	20.6	7.39	30.695	80.883	82.902	20.989
Scalling A	354.7	-1,786.02	-32,007.10	23,627.409	23,525.892	6,219.453
B	446.4	116.813	3,393.268	41,402.4	26,094.3	207.039
Rotate B	446.4	-2,805.791	-15,569.98	26,094.3	41,402.4	-207.039
Rotate C 90°	148.2	592.877	7,920.562	4,846.636	8,303.855	-1,244.777
Rotate C 180°	34.4	-45.191	-116.245	288.043	506	-63.349

ตารางที่ 2.2 แสดง Scaled Central Spatial Moment ของภาพทดสอบ ซึ่งเป็นตัวอักษร

Hu[ref] ได้เสนอการนอร์มอไลเซชัน (Normalization) Unscaled Central Moment ของสมการ 2.25 โดยใช้ความสัมพันธ์

$$V(m,n) = \frac{U_u(m,n)}{[M(0,0)]^{\alpha}} \qquad \text{.....(2.41ก)}$$

โดยที่ $$\alpha = \frac{m+n+1}{2} \qquad \text{.....(2.41ข)}$$

สำหรับ m + n = 2 ,3 ,.... Normalized Central Moment นี้ได้ถูกใช้โดย Hu เพื่อสร้าง Moment ในโดเมนภาพต่อเนื่องที่ไม่แปรผัน (ไม่เปลี่ยนแปลง) ต่อการเคลื่อนที่ (Translation) การหมุน (Rotate) และ ย่อ ขยาย (Scaling) ของภาพ เราเรียกว่า Hu Moment Invariant ซึ่งถูกนิยามดังนี้

$$h1 = V(2,0) + V(0,2) \qquad \text{.....(2.42ก)}$$

$$h2 = [V(2,0) + V(0,2)]^2 + 4[V(1,1)]^2 \qquad \text{.....(2.42ข)}$$

$$h3 = [V(3,0) - 3V(1,2)]^2 + [V(0,3) - 3V(1,1)]^2 \qquad \text{.....(2.42ค)}$$

$$h4 = [V(3,0) - 3V(1,2)]^2 + [V(0,3) - V(1,1)]^2 \qquad \text{.....(2.42ง)}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$h5 = [V(3,0) - 3V(1,2)][V(3,0) + V(1,2)][[V(3,0) + V(1,2)]^2 - 3[V(0,3) + V(2,1)]^2]$$
$$+ [3V(2,1) - V(0,3)][V(0,3) + V(2,1)][3[V(3,0) + V(1,2)]^2$$
$$- [V(0,3) + V(2,1)]^2] \quad \text{.....(2.42จ)}$$

$$h6 = [V(2,0) - V(0,2)][[V(0,3) + V(1,2)]^2 - [V(0,3) + V(2,1)]^2]$$
$$+ 4V(1,1)[V(3,0) + V(1,2)][V(0,3) + V(2,1)] \quad \text{.....(2.42ช)}$$

$$h7 = [3V(2,1) - V(0,3)][V(0,3) + V(1,2)][[V(0,3) + V(1,2)]^2 - 3[V(0,3) + V(2,1)]^2]$$
$$+ [3V(1,2) - V(3,0)][V(0,3) + V(2,1)][3[V(0,3) + V(1,2)]^2 - [V(0,3) + V(2,1)]^2] \text{ .....(2.42ซ)}$$

ตารางที่ 2.4 แสดง Invariant Moment ของวัตถุทดสอบ สังเกตว่า Moment ของวัตถุเดียวกันมีค่าใกล้เคียงกันโดยไม่ขึ้นกับการวางตัวของวัตถุ

Hu Invariant Moment เป็นคุณลักษณะที่ไม่แปรผันต่อการเคลื่อนที่ (Translation) การหมุน (Rotation) และ ย่อขยาย (Scaling) ของภาพ หรือที่เรียกว่า Similarity Transformation

Image	h[1] $*10^{-1}$	h[2] $*10^{-2}$	h[3] $*10^{-3}$	h[4] $*10^{-3}$
A	3.86	15.875	33.901	28.377
Scaling A	3.748	15.024	29.743	23.973
B	3.387	11.473	.663	.292
Rotate B	3.387	11.473	.663	.638
Rotate C 90°	5.988	37.135	30.655	31.301
Rotate C 180°	5.584	32.091	23.797	24.913

ตารางที่ 2.3 แสดง Invariant Moment of Test Image

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 คุณสมบัติพื้นฐานของ MCS-51

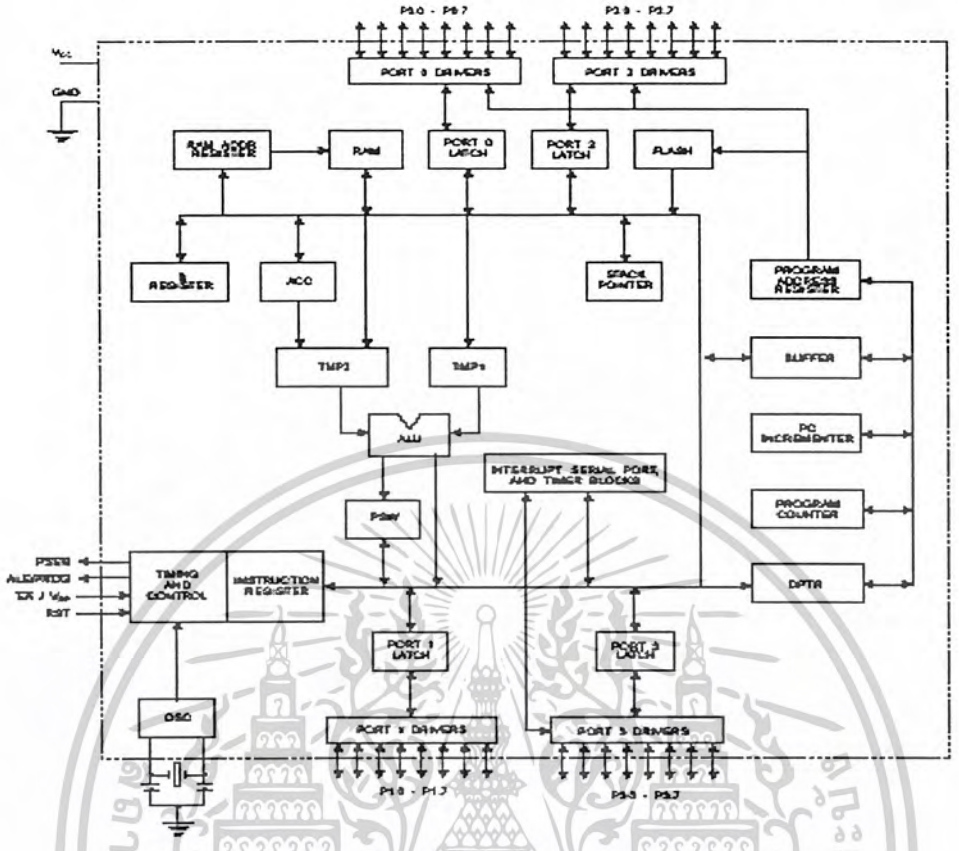
คุณสมบัติที่สำคัญของไมโครคอนโทรลเลอร์ตระกูล MCS-51 ประกอบด้วย

- หน่วยประมวลผลกลางขนาด 8 บิต
- สามารถประมวลผลข้อมูลได้ในระดับบิต
- สามารถอ้างตำแหน่งของหน่วยความจำโปรแกรมได้ 64 กิโลไบต์
- สามารถอ้างตำแหน่งของหน่วยความจำข้อมูลได้ 64 กิโลไบต์
- มีหน่วยความจำข้อมูลภายใน (Internal RAM) จำนวน 128 ไบต์
- พอร์ทอินพุท / เอาท์พุทแบบขนาน 32 เส้น สามรถแกการทำงานได้อิสระ
- มีวงจรนับ / จับเวลา ขนาด 16 บิต จำนวน 2 วงจร
- มีวงจรสื่อสารอนุกรมแบบ 2 ทิศทาง โดยรับส่งข้อมูลได้ในเวลาเดียวกัน (Full Duplex)
- มีวงจรควบคุมการอินเทอร์รัพต์จากแหล่งกำเนิด 6 แหล่ง
- มีวงจรออสซิลเลเตอร์ภายใน
- โดยทั่วไปแล้วไมโครคอนโทรเลอร์ตระกูลนี้จะเป็นไอซี LSI แบบ DIP 40 ขา

2.7 โครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51

โครงสร้างภายในของ MCS-51 แสดงดังรูปที่ 3.1 ซึ่งจะเห็นว่าแต่ละบล็อกที่เป็นวงจรควบคุมรีจีสเตอร์ (Register) หรือหน่วยความจำภายใน จะถูกเชื่อมต่อเข้าด้วยกันผ่านทางเส้นสัญญาณที่เรียกว่าบัสข้อมูลภายใน รีจีสเตอร์และหน่วยความจำต่อไปนี้จะถูกนำไปใช้ในระหว่างการประมลผลคำสั่ง หน้าที่ของโปรแกรมที่ผู้ใช้สร้างขึ้นมาก็จะเป็นการควบคุมการรับหรือส่งข้อมูลระหว่างรีจีสเตอร์เหล่านี้ ซึ่งอาจมีการดำเนินการร่วมกับหน่วยประมวลผลทางคณิตศาสตร์และลอจิก (ALU : Arithmetic and Logic Unit)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 โครงสร้างภายในของ MCS-51

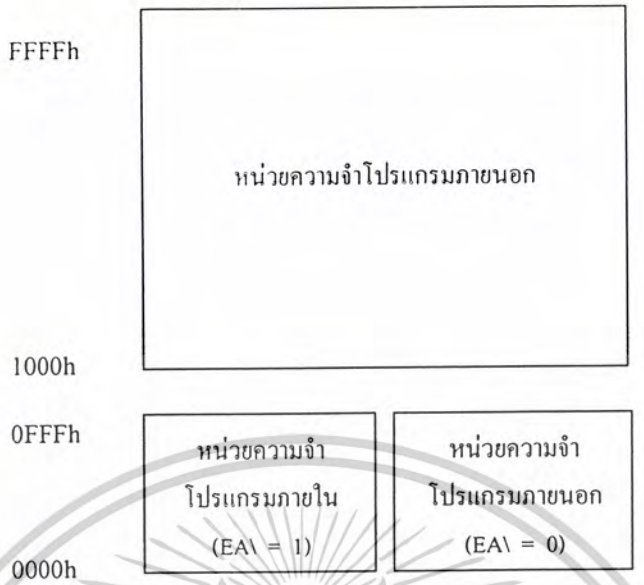
2.8 โครงสร้างหน่วยความจำ

ไมโครคอนโทรลเลอร์ MCS-51 แยกการจัดการหน่วยความจำออกเป็น 2 ส่วน คือ หน่วยความจำโปรแกรม และหน่วยความจำข้อมูล ซึ่งทั้ง 2 ส่วนนี้มีหน้าที่แตกต่างไปจากกัน และใช้วิธีการอ้างตำแหน่ง สัญญาณการติดต่อแยกจากกัน

2.8.1 หน่วยความจำโปรแกรม

เป็นหน่วยความจำประเภทที่เมื่อตัดการจ่ายไฟฟ้าออก ข้อมูลจะไม่หายไป จึงใช้สำหรับบรรจุคำสั่ง โปรแกรม หรือข้อมูลที่คงที่ต่าง ๆ มี 2 แบบ คือ หน่วยความจำภายในตัว MCS-51 ขนาดความจุจะขึ้นอยู่กับรุ่นที่ใช้ ซึ่งเหมาะสมสำหรับการผลิตจำนวนมาก หรือต้องการอุปกรณ์ที่กะทัดรัด อีกแบบหนึ่งเป็นการใช้ไอซีภายนอก ซึ่งจะเชื่อมต่อกับพอร์ต 0 และ 2 ในการอ้างอิงแอดเดรสและข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

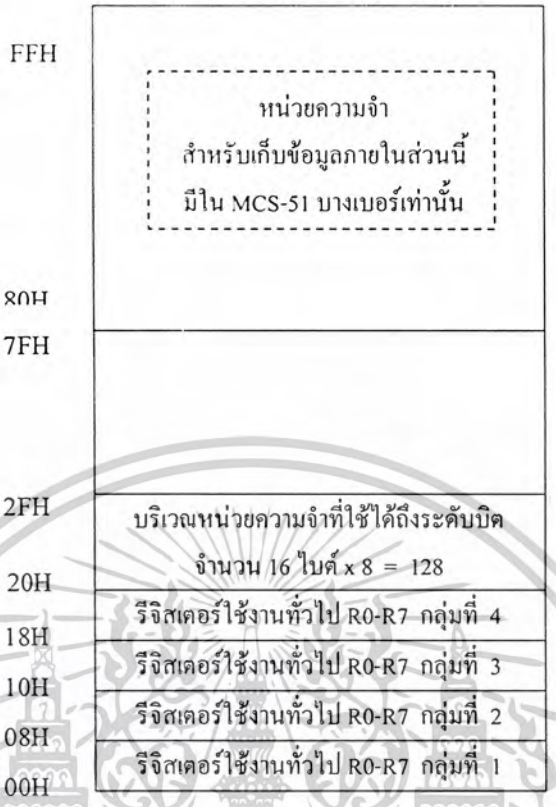


รูปที่ 2.16 แสดงการจัดพื้นที่หน่วยความจำโปรแกรมสำหรับ MCS-51

2.8.2 หน่วยความจำข้อมูล

เป็นหน่วยความจำที่ใช้สำหรับเก็บข้อมูลชั่วคราวในระหว่างการทำงานตามคำสั่ง RAM ภายในของ MCS-51 จะมีขนาด 256 ไบต์ โดย128 ไบต์สูง 80H-FFH จะเป็นพื้นที่ของรีจิสเตอร์หน้าที่พิเศษ (SFR) 20 ตำแหน่ง ส่วนใน 128 ไบต์ล่างผู้ใช้งานสามารถเข้าถึงได้โดยตรง แอดเดรส 00H-1FH จำนวน 32 ไบต์ จะถูกใช้งานในฐานะริสเตอร์ใช้งานทั่วไป จะเลือกใช้งานได้ครั้งละ 1 กลุ่ม (แบงก์) จำนวน 8 ไบต์ ปกติแล้วจะมีการใช้งานในแบงก์ 0 เท่านั้น แอดเดรส 20H-2FH จำนวน 16 ไบต์ บริเวณนี้สามารถอ้างอิงใช้งานได้ในระระดับบิต คือ บิต 00 (LSB ของ 20H) ถึง 7FH (MSB ของ 2FH) = 128 บิต แอดเดรส 30H-7FH จำนวน 80 ไบต์ สามารถใช้งานเป็นไบต์ข้อมูลได้อิสระ การจัดพื้นที่ของหน่วยความจำข้อมูลได้แสดงในรูปที่ 2.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 แสดงการจัดพื้นที่หน่วยความจำสำหรับ MCS-51

2.9 วงจรนับ/จับเวลา (TIMMER/COUNTER)

มีขนาด 16 บิต จำนวน 2 ตัว T0 และ T1 ทั้ง 2 ตัว ยังแบ่งได้เป็นรีจิสเตอร์ 8 บิต คือ TH0,TL0 และ TH1,TL1 ซึ่งสามารถนำไปใช้งานได้อย่างอิสระโดยควบคุมให้นับสัญญาณนาฬิกาภายใน หรือ นับสัญญาณพัลส์ จากภายนอกก็ได้ การควบคุมจะใช้รีจิสเตอร์ 2 ตัว คือ TMOD และ TCON การทำงานของตัวนับจะเพิ่มค่าขึ้นจากที่กำหนดเริ่มต้นไปเรื่อยๆ จนถึงค่าสูงสุดของรีจิสเตอร์แล้วถ้านับสัญญาณเพิ่มอีกจะเกิด Overflow รีจิสเตอร์จะกลับไปเป็น 0 และ Timer flag แสดงการอินเทอร์รัพต์ จะถูกเซต

ตัวนับ/จับเวลาสามารถโปรแกรมให้ทำงานได้ต่างกัน 4 โหมด โดยการตั้งค่าในรีจิสเตอร์ TMOD ซึ่งการทำงานแต่ละโหมดจะเป็นดังนี้

โหมด 0 โดยรีจิสเตอร์ตัวนับจะถูกกำหนดให้มี 13 บิต ประกอบด้วย TH1 8 บิต และ TL1 อีก 5 บิต

โหมด 1การทำงานจะเหมือนกับโหมด 0 แต่ตัวนับจะเป็น 16 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมด 2 จะใช้รีจิสเตอร์ TL1 เป็นตัวนับเพียงตัวเดียวและเมื่อ TL1 นับจนเป็น "ศูนย์" หมดทุกบิต ก็จะมีการโหลดค่าจากรีจิสเตอร์ TH1 เข้าไปใน TL1 โดยอัตโนมัติและทำการเซ็ต TF1 ค่าใน TH1 นี้สามารถตั้งค่าได้โดยซอฟแวร์

โหมด 3 เป็นการเพิ่มตัวจับเวลาขึ้นอีก 1 ตัวแต่จะเป็นขนาด 8 บิตทั้งคู่ ซึ่งลักษณะการทำงานอื่นๆ จะเหมือนกับโหมด 0

2.10 ชุดคำสั่งในไมโครคอนโทรลเลอร์ MCS-51

คำสั่งทั้งหมดสามารถแยกเป็นประเภทตามลักษณะการทำงานได้ดังนี้

กลุ่มคำสั่งทางคณิตศาสตร์ (Arithmatic Instructions) เป็นกลุ่มคำสั่งทางคณิตศาสตร์ทั้งหมด ซึ่งประกอบด้วยการ บวก ลบ คูณ หาร รวมทั้งคำสั่งในการเพิ่มค่าข้อมูลในหน่วยความจำสำหรับเก็บข้อมูลทั่วไปภายในชิพ

กลุ่มคำสั่งทางตรรกะ (Logical Instructions) ประกอบด้วยคำสั่ง เช่น AND, OR, Complement รวมทั้งคำสั่งสำหรับเลื่อนบิตข้อมูลไปทางซ้ายหรือขวา โดยผ่านบิต Carry Flag หรือไม่ก็ได้ นอกจากนี้ยังมีคำสั่งพิเศษที่ใช้ในการสลับที่ข้อมูลด้วย

กลุ่มคำสั่งเคลื่อนย้ายข้อมูล (Logical Instructions) เป็นคำสั่งที่ใช้สำหรับการเคลื่อนย้ายข้อมูลเพื่อใช้ในการนำข้อมูลที่ได้จากการประมวลผลไปเก็บยังหน่วยความจำ หรือเพื่อย้ายข้อมูลไปยังรีจิสเตอร์เพื่อประมวลผล เนื่องจากบางคำสั่งจำเป็นต้องทำที่รีจิสเตอร์เฉพาะเท่านั้น เช่นคำสั่งในการคูณหรือหารที่ต้องทำงานกับรีจิสเตอร์ A,B เท่านั้น คำสั่งนี้ยังแบ่งออกเป็นกลุ่มย่อย ๆ ดังนี้

- กลุ่มคำสั่งสำหรับเคลื่อนย้ายข้อมูลในหน่วยความจำสำหรับเก็บข้อมูลภายในชิพ
- กลุ่มคำสั่งสำหรับเคลื่อนย้ายข้อมูลในหน่วยความจำสำหรับเก็บข้อมูลภายนอกชิพ
- กลุ่มคำสั่งสำหรับเคลื่อนย้ายข้อมูลสำหรับโปรแกรมทั้งภายในและภายนอกชิพ

กลุ่มคำสั่งในการควบคุมลำดับการทำงานของโปรแกรม (Program Control Instructions) เป็นคำสั่งที่ใช้ในการควบคุมลำดับการทำงานของโปรแกรม ทั้งแบบมีและไม่มีเงื่อนไข

กลุ่มคำสั่งสำหรับประมวลผลแบบบูลีน (Boolean Instruction) การประมวลผลแบบนี้มีไว้สำหรับควบคุมโดยเฉพาะ จะเป็นการประมวลผลด้วยข้อมูลระดับบิต โดยมีหน่วยความจำขนาด 1 บิต สำหรับประมวลผลซึ่งสามารถอ้างตำแหน่งได้โดยตรง ซึ่งหน่วยความจำนี้จะอยู่บริเวณเดียวกับหน่วยความจำสำหรับเก็บหน่วยความจำทั่วไปภายในชิพและที่เป็นรีจิสเตอร์ใช้งานเฉพาะบางตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11 โครงสร้างการอินเทอร์รัพต์ของ MCS-51

ตามโครงสร้างด้านการอินเตอร์รัพต์ของ MCS-51 สามารถกำหนดให้มีการยอมหรือไม่ยอมให้มีการอินเทอร์รัพต์ของแต่ละสัญญาณได้ โดยการกำหนดค่าของบิตภายในรีจิสเตอร์ IE ซึ่งจะมีทั้งแบบระบุถึงอินเทอร์รัพต์โดยรวมทั้งหมด และอินเทอร์รัพต์แต่ละประเภทได้ ในกรณีที่มีการเซทบิตหมายถึงการยอมรับการอินเทอร์รัพต์

การกำหนดระดับความสำคัญให้กับสัญญาณอินเทอร์รัพต์แต่ละประเภทนั้น สามารถทำได้โดยการกำหนดข้อมูลให้กับบิตภายในรีจิสเตอร์ IP

สัญญาณที่เข้ามาทำการอินเทอร์รัพต์ CPU มี 5 ลักษณะ คือ

สัญญาณ	ความหมาย
INT 0	สัญญาณจากภายนอกจากขา P3.2
INT 1	สัญญาณจากภายนอกจากขา P3.3
TIMER 0	สัญญาณจากการโอเวอร์โฟลว์ของ T0
TMER 1	สัญญาณจากการโอเวอร์โฟลว์ของ T1
SERIAL PORT	การอินเทอร์รัพต์จากการรับ/ส่งข้อมูลอนุกรม

ตารางที่ 2.4 ลักษณะสัญญาณที่เข้ามาอินเทอร์รัพต์

ชื่อสัญญาณ	ความหมาย	แอดเดรสโปรแกรมย่อย(HEX)
IE0	อินเทอร์รัพต์ภายนอก 0	0003
TF0	Timer / Counter 0	000B
IE1	อินเทอร์รัพต์ภายนอก 1	0013
TF1	Timer / Counter 1	001B
TI หรือ RI	วงจรรับส่งข้อมูลอนุกรม	0023

ตารางที่ 2.5 ระดับความสำคัญและตำแหน่งแอดเดรสของ Interrupt Service Routine

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12 ฐานเวลาการทำงานของ CPU

วงจรออสซิลเลเตอร์ภายใน มีหน้าที่สร้างสัญญาณนาฬิกา ซึ่งใช้เป็นฐานเวลาในการกำหนดจังหวะการทำงานของหน่วยงานทั้งหมด ช่วงเวลาพื้นฐานหน่วยย่อยของการทำงานที่เรียกว่า STATE จะใช้ 2 คาบเวลา โดย 6 state จะเป็น 1 Machine cycle ทั่วไปแล้วจะสามารถหาเวลาของ 1 Machine cycle จากความถี่ของคริสตัลที่ใช้ต่อภายนอก = 12 / Crystal Frequency

2.13 ริจิสเตอร์หน้าที่พิเศษ SFR

- *Stack pointer มีขนาด 8 บิต* เมื่อ รีเซท จะมีค่าเป็น 07H จะทำงานโดยเพิ่มค่าแล้วเก็บข้อมูล
- *Data pointer มีขนาด 16 บิต* ใช้อ้างอิงแอดเดรสหน่วยความจำภายนอก Indirect addressing
- *Program status word* บอกถึงสภาวะการทำงานต่างๆ ของคำสั่ง รวมถึงการเลือกใช้งานแบงก์รีจิสเตอร์
- *Serial buffer 8 บิต* สำหรับการสื่อสารอนุกรมมี 2 ชุด รับและส่ง โดย CPU จะเลือกใช้งานให้พอร์ต 0 ,1 ,2 ,3 ขนาด 8 บิต ใช้เป็นอินพุท / เอาท์พุทข้อมูลระดับบิตได้

ชื่อบิต	ตำแหน่ง	ความหมาย
CY	PSW.7	CARRY FLAG
AC	PSW.6	AUXILIARY CARRY FLAG
F0	PSW.5	ORIGINAL FLAG
RS1	PSW.4	บิตสำหรับเลือกรีจิสเตอร์แบงก์
RS0	PSW.3	บิตสำหรับเลือกรีจิสเตอร์แบงก์
OV	PSW.2	OVERFLOW FLAG
-	PSW.1	-
P	PSW.0	PARITY FLAG

ตารางที่ 2.6 แสดงบิตต่างๆ ภายในรีจิสเตอร์ PSW

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับหน่วยความจำข้อมูลภายนอกของ MCS-51 สามารถเชื่อมต่อได้เช่นเดียวกับหน่วยความจำโปรแกรม ต่างกันตรงที่ต่อขาสัญญาณควบคุม โดยชุดคำสั่งของการติดต่อกับหน่วยความจำโปรแกรมและข้อมูล แบบภายในและ ภายนอกจะแยกออกต่างกันอย่างชัดเจน

ชื่อ	สัญลักษณ์	แอดเดรส	การใช้งานระดับบิต
พอร์ต O	PO	80H	ได้
STACK POINTER	SP	81H	ได้
DATA POINTER	DPL	82H	ได้
(DPTR)	DPH	83H	ได้
POWER CONTROL	PCON	87H	ไม่ได้
Timer/counter control	TCON	88H	ได้
Timer/counter mode	TMOD	89H	ไม่ได้
Timer/counter 0 (LOW)	TL0	8AH	ไม่ได้
Timer/counter 1 (LOW)	TL1	8BH	ไม่ได้
Timer/counter 0 (HIGH)	THO	8CH	ไม่ได้
Timer/counter 1 (HIGH)	TH1	8DH	ไม่ได้
พอร์ต 1	P1	90H	ได้
SERIAL CONTROL	SCON	98H	ไม่ได้
SERIAL DATA BUFFER	SBUF	99H	ไม่ได้
พอร์ต 2	P2	A0H	ได้
INTERRUPT ENABLE RECISTER	IE	A8H	ได้
พอร์ต 3	P3	B0H	ได้
INTERRUPT PRIORITY REGISTER	IP	B8H	ได้
PROGRAM STATUS WORD	PSW	DOH	ได้
ACCUMULATOR	ACC	E0H	ได้
B REGISER	B	F0H	ได้

ตารางที่ 2.7 แสดงการจัดพื้นที่ของหน่วยความจำรีจิสเตอร์หน้าที่พิเศษ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.14 พอร์ตอินพุต/เอาท์พุท

มีจำนวน 4 พอร์ตขนาน พอร์ตละ 8 บิต สามารถใช้งานในลักษณะสัญญาณเดี่ยวหรือกลุ่มของสัญญาณได้ นอกจากนี้บางพอร์ตยังสามารถนำไปใช้งานอื่น ได้แก่

พอร์ต 0 ใช้เป็นมัลติเพล็กซ์ระหว่าง บัสแอดเดรสไบต์ต่ำ และบัสข้อมูลสำหรับหน่วยความจำภายนอก

พอร์ต 2 ใช้เป็นบัสแอดเดรสไบต์สูงร่วมกับพอร์ต 0

พอร์ต 3 สามารถนำไปเป็นขาสัญญาควบคุมต่าง ๆ ได้ดังนี้

I/O ปกติ	Alternate function	หน้าที่
P3.0	RxD	ขารับข้อมูลของพอร์ตอนุกรม
P3.1	TxD	ขาส่งข้อมูลของพอร์ตอนุกรม
P3.2	INT0\	ขาอินพุทสัญญาณอินเทอร์รัพต์จากภายนอก
P3.3	INT1\	ขาอินพุทสัญญาณอินเทอร์รัพต์จากภายนอก
P3.4	T0	สัญญาณพัลส์ภายนอกสำหรับ Timer 0
P3.5	T1	สัญญาณพัลส์ภายนอกสำหรับ Timer 1
P3.6	WR\	สัญญาณเพื่อเขียนข้อมูลไปหน่วยความจำภายนอก
P3.7	RD\	สัญญาณเพื่ออ่านข้อมูลหน่วยความจำภายนอก

ตารางที่ 2.8 แสดงขาสัญญาณควบคุมต่าง ๆ

2.15 การใช้งานของพอร์ตอนุกรม

การสื่อสารข้อมูลอนุกรมเป็นการรับหรือส่งข้อมูลในลักษณะกลุ่มของบิตคราวละ 1 บิต เรียงลำดับเรื่อยไปจนสิ้นสุด การสื่อสารแบบนี้จะมีข้อดีแตกต่างจากการสื่อสารแบบขนานเป็นอย่างมากเนื่องจากข้อมูลมีการโอนย้ายมากพร้อมกัน จึงมีความจำเป็นต้องใช้จำนวนเส้นสัญญาณมากขึ้นตามจำนวนบิตของข้อมูลด้วย ในขณะที่การสื่อสารข้อมูลแบบอนุกรมนั้นต้องการเส้นสัญญาณเพียง 2-3 เส้นเท่านั้น ดังนั้นการสื่อสารแบบขนานจึงไม่เหมาะสมในการสื่อสารกับอุปกรณ์ภายนอกที่อยู่ห่างไกลกัน เพราะจะทำให้สิ้นเปลืองค่าใช้จ่ายมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.15.1 ความเร็วของการสื่อสารข้อมูลอนุกรม

เนื่องจากการสื่อสารแบบอนุกรมเป็นการรับ / ส่งข้อมูลในลักษณะกลุ่มของบิตข้อมูล (Bit Stream) ดังนั้นจึงต้องให้ความสนใจในการพิจารณาถึงเรื่องอัตราความเร็วในการรับ / ส่งบิตเหล่านี้เป็นลำดับแรก โดยทั่วไปมักจะระบุกันในหน่วยของจำนวนบิตข้อมูลภายในเวลา 1 วินาที เรียกว่าอัตราบอด ตามค่ามาตรฐานเหล่านี้ ได้แก่ 110,150,300,1200,2400,4800,9600,19200 บอด ข้อมูลทั้ง 8 บิต นี้หากว่าถูกส่งออกมาด้วยอัตรา 2400 บอด จะใช้เวลาในการส่งข้อมูล 1 บิต มีค่าเท่ากับ 1/2400 หรือ 416 us และเวลาในกากรส่งข้อมูลทั้ง 8 บิตมีค่าเท่ากับ (8*416)

2.15.2 รูปแบบการส่งข้อมูลอนุกรม

การสื่อสารอนุกรมแบบ อะซิงโครนัส จะให้การแปลงข้อมูลให้เป็นอนุกรมแล้วเพิ่มเติมบิตบางอย่างร่วมไปกับการส่งข้อมูลจริง ได้แก่

2.15.2.1 บิตเริ่มต้น (Start Bit)

บิตเริ่มต้นมีหน้าที่สำหรับการบ่งบอกให้ทราบถึงตำแหน่งจุดเริ่มต้นก่อนบิตข้อมูล ตามปรกติแล้วค่าของบิตเริ่มต้นจะเป็นระดับลอจิกต่ำ

2.15.2.2 บิตแสดงสภาวะความเป็นเลขคู่หรือเลขคี่ (Parity Bit)

บิตนี้มีหน้าที่เพื่อตรวจสอบความถูกต้องของข้อมูล โดยทั่วไปมักเรียนกว่า บิตพาริตี้ และจะนำไปต่อท้ายบิตข้อมูล ค่าของบิตนี้ขึ้นอยู่กับจำนวนค่าของบิตที่เป็น 1 ซึ่งจะเป็นได้ 2 ลักษณะ คือ พาริตี้คู่ (Event Parity) หรือพาริตี้คี่ (Odd Parity) ตัวอย่างเชน ระบบที่ติดต่อกัน โดยระบุว่าจะใช้พาริตี้คี่ ทางด้านส่งจะนำค่าของข้อมูลที่จะส่งมาพิจารณาหา จำนวนของบิตที่มีค่าเป็น 1 ถ้าเป็นจำนวนคู่อยู่แล้วพาริตี้บิตจะมีค่าเป็น 0 แต่หากว่าจำนวนของบิตทีมีค่าเป็น 1 เป็นเลขคี่ ค่าของพาริตี้บิตก็จะมีค่าเป็น 1 การพิจารณาด้านรับเป็นการตรวจสอบจำนวนบิตที่มีค่าเป็น 1 ของข้อมูลที่ได้รับมาทั้งหมดรวมทั้งบิตพาริตี้ ถ้ามีค่าเป็นเลขจำนวนคู่ แสดงว่าข้อมูลที่ได้รับเข้ามานี้ถูกต้อง แต่หากเป็นเลขจำนวนคี่ แสดงว่าเกิดการผิดพลาดของข้อมูลขึ้น เป็นต้น

2.15.2.3 บิตสุดท้าย (Stop Bit)

บิตสุดท้ายเป็นบิตที่เพิ่มขึ้นเพื่อระบุถึงขอบเขตการสิ้นสุดของกลุ่มบิดข้อมูล บิตสุดท้ายนี้สามารถโปรแกรมบิตได้คือ 1 บิต ½ บิต และ 2 บิต ดังนั้นในกรณีการส่งข้อมูล 8 บิต หากข้อมูลถูกส่งออกไปด้วยอัตราเร็ว 2400 บอด เวลาโดยรวมในการส่งข้อมูล 1 ไบต์ จะมีค่าเป็น (12*416) us

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.15.3 การส่งข้อมูลอนุกรมของ 8051

พอร์ตอนุกรมของ 8051 มีโครงสร้างการทำงานในแบบที่เรียกว่า ฟูลดูเพล็กซ์ในการรับและส่งข้อมูลอนุกรมได้ในเวลาเดียวกัน โดยทางด้านวงจรของตัวส่ง ประกอบด้วยข้อมูลออกไปยังพอร์ตอนุกรม ทาสงขาสัญญา TxD ส่วนวงจรทางด้านรับ ประกอบด้วย SBUF เช่นเดียวกับสัญญาณข้อมูลอนุกรมที่รับเข้ามาทางขาสัญญาณ RxD

พอร์ตอนุกรมของ 8051 สามารถโปรแกรมการทำงานได้หลายโหมดด้วยกัน โดยเลือกที่บิต SMO และ SMI ซึ่งอยู่ในรีจิสเตอร์ควบคุม SCON การทำงานทั้ง 4 โหมดของพอร์ตอนุกรมมีดังนี้

โหมด 0: ใช้รับส่งข้อมูล 8 บิต โดยการส่งจะเลื่อนออกที่ละบิต โดยส่งบิต D 0 ออกไปก่อนทางขา RxD และไม่มีการส่งบิตเริ่มต้น แต่จะส่ง Shift Clock ทางขา TxD ความเร็ว 1/12 เท่าของ CPU CLOCK

โหมด 1: ใช้สำหรับการเชื่อมต่ออนุกรมแบบ UART (Universal Asynchronous Receiver / Transmitter) โดยส่งแบบ 10 บิต ข้อมูล 8 บิต และบิตเริ่มต้น กับบิตสุดท้ายอีกอย่างละ 1 บิตและสามารถเปลี่ยนแปลงอัตราความเร็วในการส่งข้อมูลได้ โดยขึ้นกับบิต SMOD ใน PXON และอัตรา Overflow ของ Timer 1

โหมด 2: ใช้สำหรับการเชื่อมต่อนุกรมแบบ UART โดยการใช้กลุ่มข้อมูลแบบ 11 บิต และกำหนดอัตราความเร็วในการส่งข้อมูลเท่ากับ 1 / 32 และ 1 / 64 ของ CPU CLOCK และอัตราโอเวอร์โฟลว์ของ Timer 1 นอกนจากนี้โหมด 2 และ โหมด 3 ยังมีการดำเนินการอีกแบบหนึ่ง โดยสามารถนำมาใช้ประโยชน์ในการสื่อสารข้อมูลแบบที่มีไมโครโปรเซสเซอร์ หลายตัวทำงานร่วมกันได้ ซึ่งมีชื่อเรียกว่า Multi Processor Mode

Multi Processor Mode: ในโหมดนี้เราจะใช้ไมโครโปรเซสเซอร์ 1 ตัวสำหรับเป็น Master และอีก 0-256 ตัว เป็น Slave รีจิสเตอร์ที่ใช้ควบคุมการรับส่งข้อมูลอนุกรมดังรายละเอียดดังรูปที่ 2.18

SMO	SM1	SM2	REN	TB8	RB8	TI	RI

รูปที่ 2.18 แสดงบิตแต่ละบิตของการใช้งานพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SMO	SM1	โหมด 0	การทำงาน
0	0	0	ทำงานเป็นรีจิสเตอร์อัตราเร็วในการรับ / ส่งข้อมูลเท่ากัน 1 / 12 ของความถี่ออสซิลเลเตอร์
0	1	1	8 Bit UART อัตราเร็วในการรับ / ส่ง ข้อมูลกำหนดเองได้
1	0	2	9 bit UART อัตราเร็วในการรับ / ส่งข้อมูล =1 / 13 หรือ 1 /1ข64 ของความถี่ออสซิลเลเตอร์ ขึ้นกับบิต SMOD ใน PCON
1	1	3	9 bit UART อัตราเร็วในการรับ / ส่งข้อมูล กำหนดเองได้

ตารางที่ 2.9 รายละเอียดการเลือกโหมดการทำงาน

2.15.4 โหมดการทำงานของพอร์ตอนุกรมใน MCS - 51

2.15.4.1 พอร์ตอนุกรม (โหมด 0)

การทำงานของพอร์ตอนุกรม (โหมด 0) เป็นการรับและส่งข้อมูลอนุกรมจำนวน 8 โดยใช้เพียงขาสัญญาณ RxD เท่านั้น (ขานี้ใช้งาน 2 หน้าที่ ใช้รับและส่งข้อมูล) ส่วนขาสัญญาณ TxD จะนำไปใช้เพื่อเป็นขาสัญญาณนาฬิกาในการให้จังหวะ การเลื่อนข้อมูลกับวงจร เลื่อนบิตภายนอก สำหรับอัตราความเร็วจะถูกกำหนดไว้คงที่ที่ค่า 1 / 12 ของค่าความถี่ออสซิลเลเตอร์ เนื่องจากโหมดนี้ไม่มีการส่ง START BIT และ STOP BIT ดังนั้น จึงจำเป็นที่จะต้องส่งสัญญาณ SHIFT CLOCK ออกไป เพื่อใช้ Synchronize ระหว่างฝ่ายส่งและฝ่ายรับ โดยจะใช้ขา TxD ส่วนการรับข้อมูลจะรับข้อมูลเข้าทาง RxD และรับ Shift Clocks เข้าทางขา TxD ถ้าความถี่ออสซิลเลเตอร์มีค่าเท่ากับ 12 MHz ก็จะส่งได้ถึง 1 ล้าน ซึ่ง (โหมด 0) เป็นโหมดที่ส่งข้อมูลได้เร็วที่สุด

2.15.4.2 การรับส่งข้อมูลอนุกรม (โหมด 1)

การทำงานในโหมด 1 เป็นการสื่อสารข้อมูลอนุกรมจำนวน 10 บิต ประกอบด้วยบิตเริ่มต้น 1 บิต ข้อมูลจำนวน 8 บิต และบิตสุดท้ายอีก 1 บิต โดยข้อมูลจะถูกส่งออกทาง TxD และรับเข้ามาทางขาสัญญาณ RxD ในส่วนของข้อมูล 8 บิต ที่ได้รับหรือทำการส่งออก จะเป็นบิตนัยสำคัญต่ำเป็นลำดับแรกส่งทางฝ่ายรับค่าของ Stop Bit จะส่งเข้ามาจัดเก็บไว้ในบิต RB8 ภายในรีจิสเตอร์ SCON สำหรับอัตราความเร็วในการส่งข้อมูลของโหมด 1 นั้น สามารถกำหนดเลือกได้จาก Timer 1

ดังได้กล่าวแล้วว่าการส่งข้อมูลอนุกรม (โหมด 1) สามารถเปลี่ยนแปลงความเร็วได้ โดยใช้ Timer 1 ทำหน้าที่เป็นตัวกำเนิดอัตราการส่งข้อมูล และใช้แฟล็กที่เกิดขึ้นจากการโอเวอร์โฟลว์ของ Timer 1 โดย โปรแกรม Timer 1 ทำงานใน (โหมด 2) 8 - Bit Automatic Reload

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความถี่อัตราบอดขึ้นกับ $= (2^{SMOD} * \text{อัตราโอเวอร์โฟลว์ของ Timer 1}) /32$

โดย SMOD เป็นค่าของบิตภายในรีจิสเตอร์ PCON (มีค่าเป็น 1 หรือ 0) ค่าภายในรีจิสเตอร์ THI ซึ่งเป็นค่าสำหรับโหลดซ้ำ

Baud Rate	Fosc	SMOD	TIMER	MODE	RELOAD VALUE
(Mode0) Max:1MHz	12 MHz	X	X	X	X
(Mode 1) Max:375MHz	12 MHz	1	X	X	X

ตารางที่ 2.10 การเลือกอัตราบอดของวงจรพอร์ตอนุกรม

SM2 บิตเลือกการทำงานแบบ

1. เลือก Multiprocessor Mode ใช้กับโหมด 2, 3
2. เลือก Single Processor Mode ใช้ได้กับทุกโหมด

เมื่อเลือกการทำงานรับข้อมูลแบบ Multiprocessor Mode แล้วเมื่อ ข้อมูลบิตที่ 9 ที่รับได้มีค่าเป็น 1 RI จะเซ็ตทันที รายละเอียดจะได้กล่าวถึงต่อไป

REN บิตควบคุมให้รับหรือไม่รับข้อมูล

1 : ให้รับข้อมูลได้

2 : ห้ามรับข้อมูล

TB8 (Transmit bit D8) ข้อมูลบิตที่ 9 ที่จะส่งออกไปในโหมด 2, 3 ให้ใส่ในบิต TB8

RB8 (Receive bit D8) ข้อมูลบิตที่ 9 ที่รับเข้ามาจะเก็บในบิตนี้

TI บิต TI จะเป็น 1 เมื่อสิ้นสุดการส่งข้อมูล 1 ไบต์

RI บิต RI จะเป็น 1 เมื่อสิ้นสุดการรับข้อมูล 1 ไบต์

2.15.4.3 การรับส่งข้อมูลอนุกรม (โหมด 2)

โหมดนี้ใช้ทั้งหมด 11 บิต โดยแบ่งเป็น Start Bit, 9 Data, I Stop Bit โดยบิตที่ 9 ผู้ใช้สามารถกำหนดค่าเองได้ว่าจะส่งค่าอะไรออกไป โดยจะต้องนำไปใส่ไว้ในบิต TB8 ในรีจิสเตอร์ Scon ส่วนมากผู้ใช้มักจะนำบิตนี้มาใช้เป็น Parity bit โดยโหลดค่ามาจาก Parity Flag ใน PSW ส่วนทางด้านรับบิตที่ 9 จะถูกนำมาเก็บไว้ใน RB8 อัตราความเร็วในการส่ง / รับข้อมูลขึ้นกับความถี่ออสซิเลเตอร์ของ CPU และค่า SMOD ซึ่งอยู่ในบิตที่ 7 ของ PCON

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MODE 2 บอด RATES = (2^{SMOD} * Fosc) /64

ถ้า CPU RUN ที่ 12 MHz และ SMOD มีค่า 0 และ 1

- เมื่อSMOD=0 จะได้ = 20(12) (106)/64 187500 บอด
- เมื่อSMOD=1 จะได้ = 21(12) (106)/64 357500 บอด

2.15.4.4 การรับส่งข้อมูลอนุกรม (โหมด 3)

เหมือนกับโหมด 2 ทุกประการ ยกเว้นความเร็วในการรับส่งข้อมูลจะขึ้นกับอัตรา โอเวอร์โฟลว์ของ Timer 1 หรือ 2 การเปลี่ยนแปลงความเร็วสามารถคำนวณได้จากสูตร ดังนี้

Mode 1, 3 = (2 * fosc) / [(32) (12) (256-TH1)}

หรือ = (K Fosc) / {(32) (12) (256-TH1)}

ถ้า SMOD = 0 จะได้ K = 1

SMOD = 1 จะได้ K = 2

เมื่อต้องการหาค่า THI จะได้

THI = 256 - {(K * Fosc) / (384*Boudrate)}

2.15.5 การอินเทอร์รัพต์ของพอร์ตสื่อสารอนุกรม

เนื่องจากการส่งหรือรับข้อมูลอนุกรมจะส่งทีละไบต์ 8051 จึงได้กำหนดให้บิตหรือแฟล็กสถานะที่จัดรวมอยู่ในรีจิสเตอร์ SCON เช่น แฟล็ก TI จะมีค่าเป็น 1 เมื่อข้อมูลได้ทำการส่งออกไปยังภายนอกเสร็จสิ้นแล้ว และแฟล็ก RI จะมีค่าเป็น 1 เพื่อให้รู้ว่าข้อมูลผ่านเข้ามาทางพอร์ตอนุกรมเรียบร้อยแล้ว เสร็จแล้วเมื่อแฟล็ก RI, TI นี้มีค่าเป็น 1 จะมีผลทำให้เกิดการอินเตอร์รัพต์ขึ้น ดังนั้นภายในโปรแกรมรับหรือส่งจะต้องทำการตรวจสอบจากสภาวะของแฟล็กเหล่านี้เองว่าเป็นการรับหรือการส่งข้อมูล

2.15.5.1 กระบวนการรับส่งข้อมูลอนุกรมของ 8051

การส่งข้อมูลออกทางพอร์ตอนุกรมของ 8051 จะเริ่มต้นขึ้นภายหลังเมื่อมีการเขียนข้อมูลลงใน SBUF ข้อมูลนี้จะถูกเลื่อนที่ละบิต และส่งสัญญาณออกไปภายนอกโดยอัตโนมัติ เมื่อข้อมูลลงใน SBUF ข้อมูลนี้ถูกเลื่อนทีละบิต และส่งสัญญาณออกไปภายนอกโดยอัตโนมัติ เมื่อข้อมูลเหล่านี้ได้ส่งออกครบถ้วนแล้วจะทำให้ค่าของแฟล็ก TI เป็น 1 เพื่อแจ้งให้ทราบว่าขณะที่ SBUF ว่างและพร้อมที่จะส่งข้อมูลไบต์ต่อไปแล้วในกรณีที่ใช้เขียนข้อมูลใหม่ลงในรีจิสเตอร์ SBUF โดยไม่รอให้แฟล็ก TI มีค่าเป็น 1 ก่อน จะมีผลทำให้ข้อมูลที่ส่งไปผิดพลาดได้

การรับข้อมูลจากพอร์ตอนุกรมจะต้องเริ่มต้น โดยการกำหนดเซตค่าดังนี้ REN (Receiver Enable) ให้มีค่าเป็น 1 ก่อน หลังจากนั้นเมื่อมีข้อมูลภายนอกถูกส่งเข้ามายัง 8051 ที่ละบิตจนครบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเมื่อบิตสุดท้ายเลื่อนเข้ามาเรียบร้อยแล้ว ข้อมูลนั้นจะถูกย้ายมาเก็บไว้ยังริจิสเตอร์ SBUF และแฟล็ก RI ก็จะมีค่าเป็น 1 หลังจากนั้นก็จะเกิดการอินเตอร์รัพต์

(Mode2)Min:187.5MHz	12MHz	0	X	X	X
(Mode1,3):62.5 MHz	12MHz	1	0	2	FFH
19.2K	11.059MHz	1	0	2	FDH
9.6K	11.059MHz	0	0	2	FDH
4.8K	11.059MHz	0	0	2	FAH
2.4K	11.059MHz	0	0	2	F4H
1.2K	11.059MHz	0	0	2	F8H
137.5	11.059MHz	0	0	2	IDH
110	6MHz	0	0	2	72H
110	12MHz	0	0	1	FEEBH

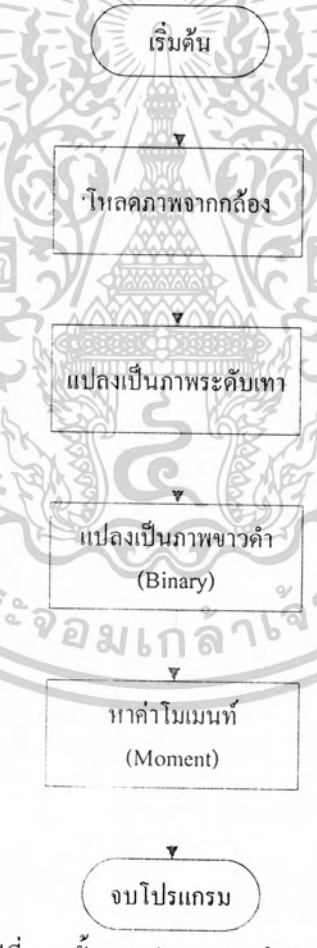
ตารางที่ 2.11การเซตค่าอัตราการส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
หลักการทำงานของโปรแกรม

3.1 โครงสร้างของโปรแกรมการจดจำวัตถุ

โปรแกรมจดจำวัตถุมีขั้นตอนต่อไปนี้คือ ในขั้นแรกทำการรับภาพจากกล้องวีดีโอซึ่งเป็นภาพสี เมื่อรับภาพมาได้ก็ทำการแปลงภาพให้เป็นภาพเกรย์สเกล 256 ระดับ จากนั้นจะทำการหาค่าเทรชโฮลเพื่อนำไปใช้ในการแปลงภาพให้เป็นไบนารี่หรือก็คือภาพขาวดำนั้นเอง ซึ่งเราจะใช้ภาพไบนารี่ที่ได้มาทำการหาค่าของโมเมนท์



รูปที่ 3.1 ขั้นการทำงานของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 โปรแกรมการแปลงภาพสีเป็นภาพเกรย์สเกล

ก่อนที่เราจะนำภาพไปหาค่าโมเมนท์เราจะต้องทำการแปลงภาพสีแบบ True Co lour ที่ได้มาจากกล้องให้กลายเป็นภาพเกร์สเกล 256ระดับก่อน โดยช้าค่าเฉลี่ยของค่าสีแดง สีเขียว และสีน้ำเงิน โดยมีขั้นตอนคือ เริ่มแรกเมื่อรับภาพแบบ True Colour จากกล้องก็ทำการหาค่าสีแดง สีเขียว และสีน้ำเงิน ซึ่งค่าสีทั้งมาจะนำไปคำนวณหาค่า เกร์ยสเกล โดยเราจะทำการหาทีละพิกเซลไปเรื่อยๆ จนครบทุกจุดของภาพ

รูปที่ 3.2 แสดงแผนผังโปรแกรมการแปลงภาพสีเป็นภาพเกรย์สเกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 โปรแกรมการแปลงภาพเกรย์เป็นภาพขาวดำ

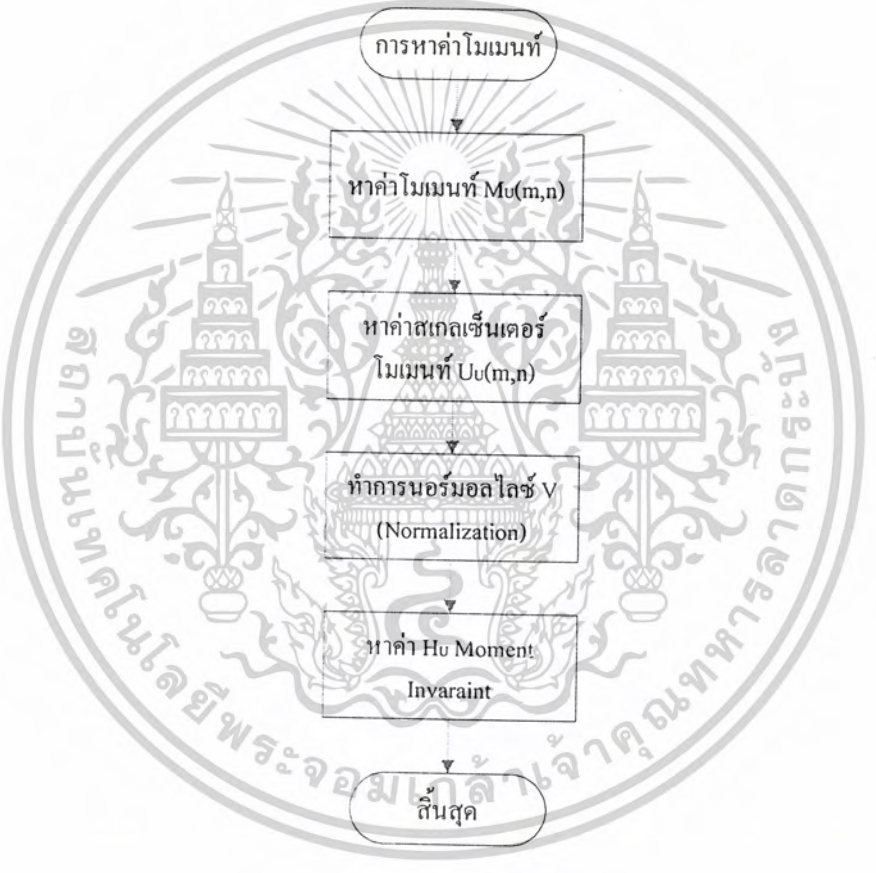
เมื่อเราได้ภาพที่เป็นเกรย์สเกลแล้วเราจะต้องทำการไบนารี่เซชั่น Binarization ซึ่งเราจะนำค่าสีของภาพเกรย์สเกลมาเปรียบเทียบกับค่าเทรชโฮล โดยการหาค่าเทรชโฮลนั้นเราจะหามาจากค่าเฉลี่ยของภาพเกรย์สเกล ถ้าค่าสีของพิกเซลนั้น มีค่าน้อยกว่าหรือเท่ากับ ค่าเทรชโฮล ให้พิกเซลนั้นเป็นสีขาว และถ้าค่าสีที่พิกเซลนั้นมีค่ามากกว่าค่าของเทรชโฮล ให้พิกเซลนั้นเป็นสีดำ ซึ่งเราจะทำการเปรียบเทียบอย่างนี้ไปจบครบทุกพิกเซลของภาพ

รูปที่ 3.3 แสดงแผนผังการทำงานของโปรแกรม Binarization

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 โปรแกรมการหาค่าโมเมนท์ของภาพ

ในการจดจำวัตถุนั้นเราจะต้องทำการหาค่าของภาพ เพื่อนำไปเปรียบเทียบกับค่าโมเมนท์ในดาต้าเบส ซึ่งมีขั้นตอนการหาคือเราจะทำการหาค่าของโมเมนท์ $M_U(m,n)$ แล้วหาค่าสเกลเซ็นเตอร์โมเมนท์ $U_U(m,n)$ เพื่อนำไปหาค่าที่ทำการนอร์มอลไลซ์ V (Normalization) ซึ่งสุดท้ายเราจะได้หาค่า H_U Moment Invariant ซึ่งเป็นค่าที่เรานำไปเปรียบในดาต้าเบส



รูปที่ 3.4 แสดงขั้นตอนการหาค่าโมเมนท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

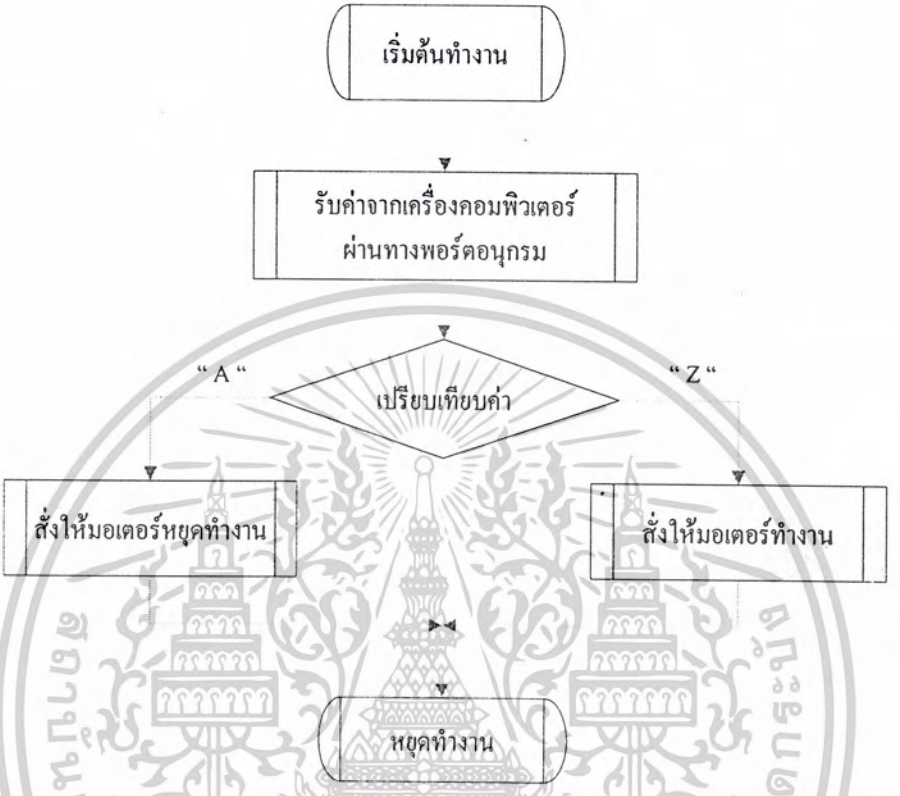
3.5 โปรแกรมตรวจจับภาพวัตถุและตรวจนับจำนวนของวัตถุ

รูปที่ 3.5 แสดงขั้นตอนตรวจจับภาพวัตถุและตรวจนับจำนวนของวัตถุ

ในการตรวจจับภาพนั้นเราจะใช้กล้องในการตรวจจับภาพของวัตถุเข้าก่อนจากนั้นทำการแปลงภาพวัตถุที่ได้เป็นภาพระดับสีเทาจากนั้นทำการแปลงภาพให้เป็นภาพไบนารี่แล้วทำการหาโมเมนท์ของภาพจากนั้นนำค่าโมเมนท์ที่ได้ไปทำการเปรียบเทียบกับค่าที่เก็บไว้ใน Database ถ้าหากว่าค่านั้นตรงกับค่าที่เก็บไว้ใน Database ตัวโปรแกรมก็จะทำการแยกแยะวัตถุและตรวจนับจำนวนของวัตถุจากนั้นจะทำการบันทึกค่าได้ไว้ใน Database อีกตัวหนึ่ง แล้วกลับไปทำขบวนการตอนต้นต่อไปเรื่อย ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

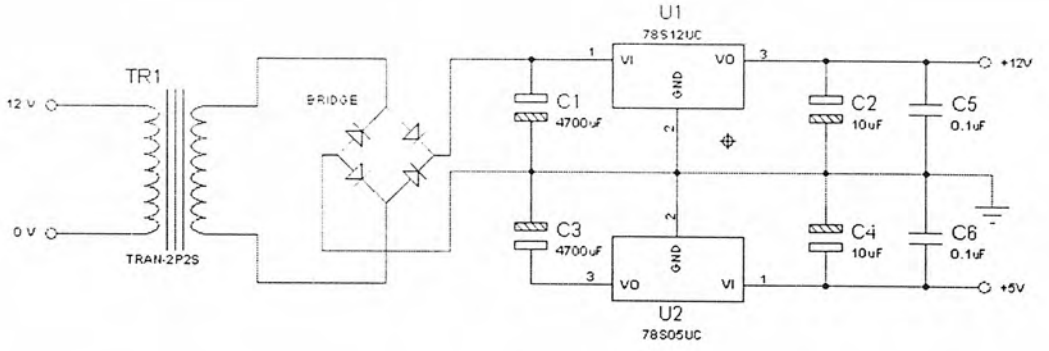
3.6 โปรแกรมควบคุมการทำงานของมอเตอร์กระแสตรงโดย MCS - 51



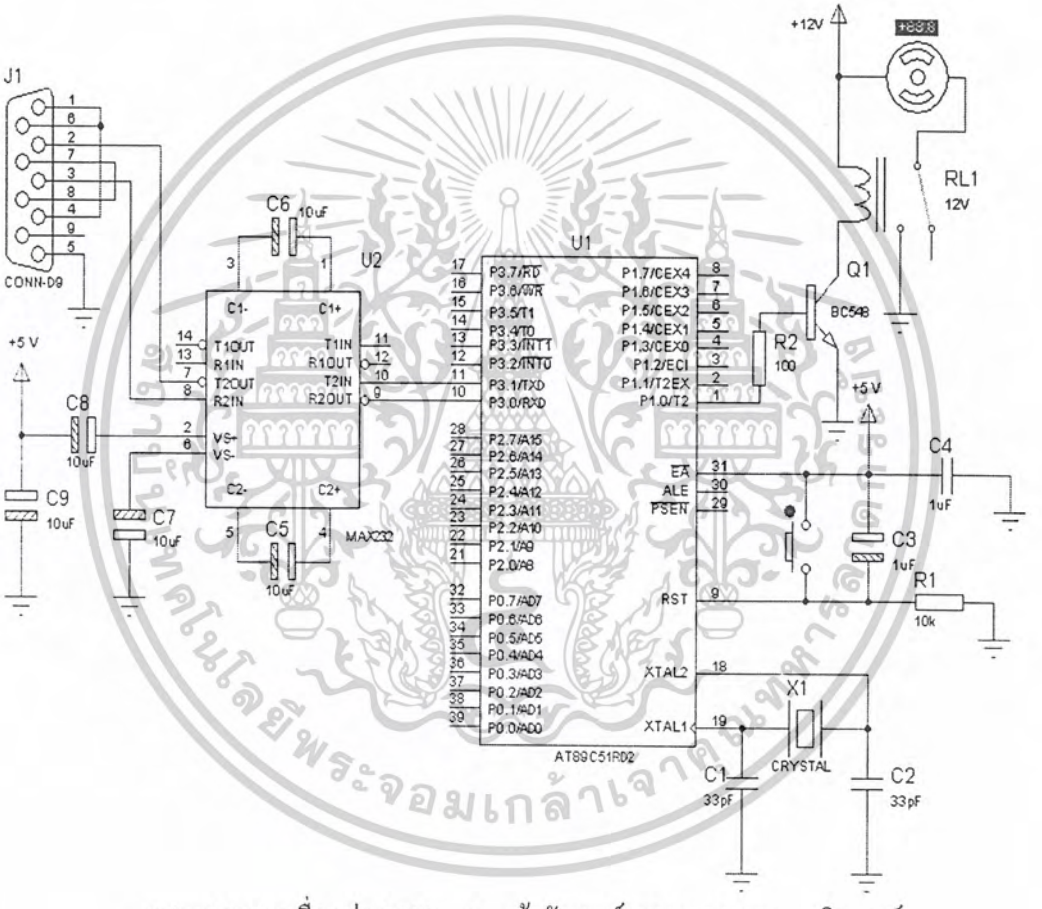
รูปที่ 3.6 แสดงการควบคุมการทำงานของมอเตอร์กระแสตรง

การทำงานเริ่มจากการรับค่าที่จะนำไปควบคุมการทำงานของมอเตอร์กระแสตรงผ่านทางพอร์ตอนุกรมของคอมพิวเตอร์แล้วจากนั้นนำค่าที่ได้ไปเปรียบเทียบกัน โดยถ้าหากค่าที่รับเข้ามามีค่าเป็น " A " ตัวโปรแกรมก็จะสั่งงานให้มอเตอร์หยุดทำงาน และถ้าหากค่าที่รับเข้ามามีค่าเป็น " Z " ตัวโปรแกรมก็จะสั่งงานให้มอเตอร์ทำงาน ส่วนวงจรที่ใช้ในการควบคุมการทำงานของมอเตอร์แสดงดังรูปที่ 3.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ก) แสดงวงจรแหล่งจ่ายไฟ



ข) แสดงการเชื่อมต่อ MCS – 51 เข้ากับพอร์ตอนุกรมของคอมพิวเตอร์

รูปที่ 3.7 แสดงการเชื่อมต่อของวงจรควบคุมการทำงานของมอเตอร์กระแสตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ผลการทดลอง

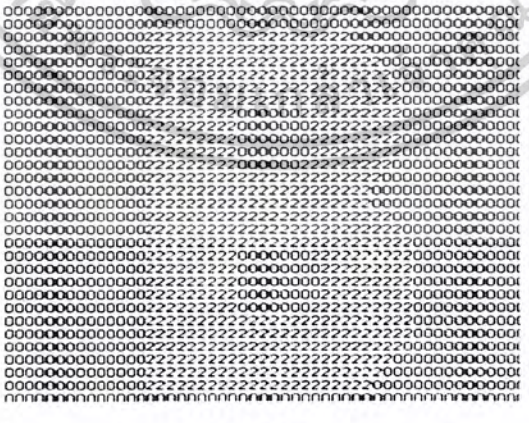
4.1 การทดลองการแปลงภาพ True Color เป็นภาพแบบ GRAY SCALE

ทำการนำภาพมีลักษณะ True Color เป็นภาพแบบ GRAY SCALE โดยทำการค่าของแต่ละสีเพื่อนำมาแปลงเป็นภาพแบบ GRAY SCALE โดยหาได้จาก 0.3R + 0.59G + 0.11B

รูปที่ 4.1 แสดงภาพที่ได้จากการแปลงภาพ GRAY SCALE

4.2 การทดลองการแปลงภาพ GRAY SCALE เป็นภาพแบบ BINARY

ทำการหาค่าเทรชโฮลจากภาพ GRAY SCALE ถ้าค่าระดับสีเทาของภาพ GRAY SCALE ต่ำกว่าเทรชโฮลให้เป็นสีขาว หรือมีค่าเป็น 1 ถ้ามากกว่าระดับเทรชโฮลเป็นสีดำ หรือมีค่าเป็น 0



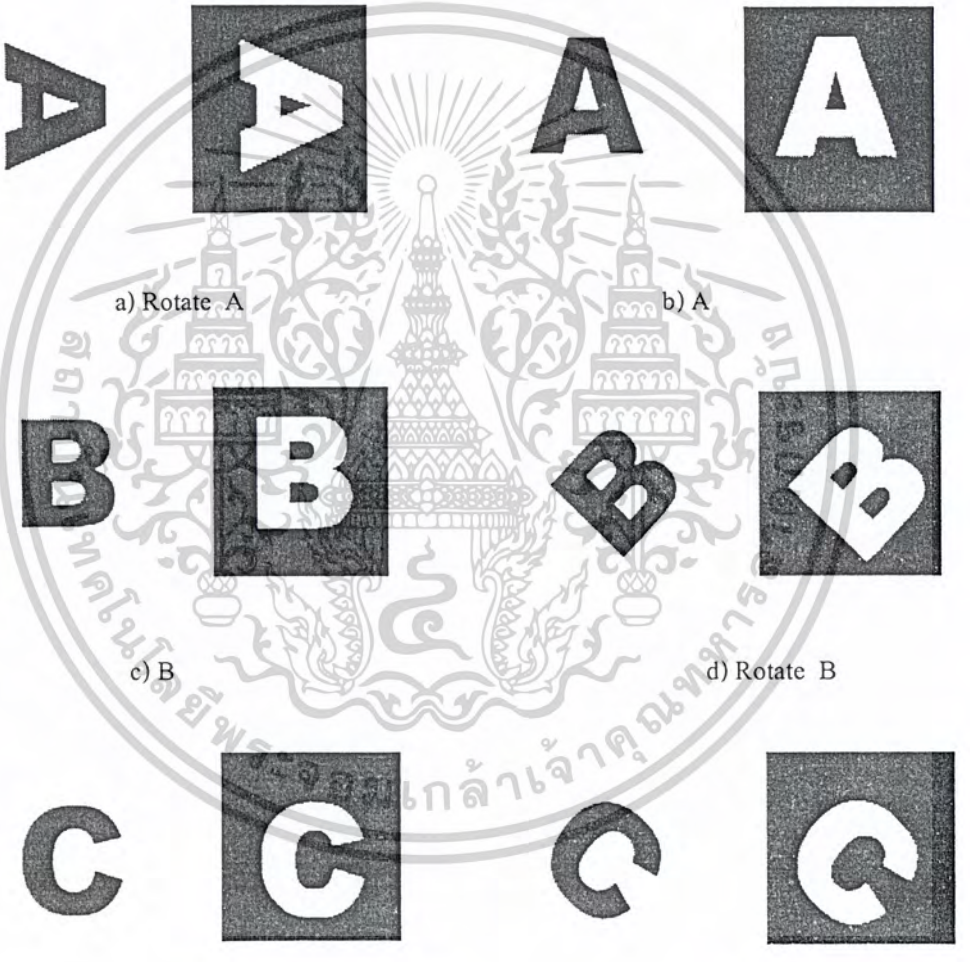
รูปที่ 4.2 แสดงภาพตัวอย่างที่ได้จากการแปลงเป็นภาพ BINARY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดลองหาค่า Moment ของภาพ

สำหรับการทดลองนั้น เราจะศึกษาถึงปัจจัยต่าง ๆ ที่มีผลต่อการทำงาของโปรแกรมเพื่อเป็นแนวทางในการออกแบบและปรับปรุงแก้ไขต่อไปโดยการทดลองจากภาพตัวอย่างที่ทำการถ่ายภาพออกมาในรูปแบบของตัวหนังสือและทำการจัดเก็บภาพแบบ True Color

ในการทดลองโปรแกรมนั้นผลของการทดลองจะหาออกมาอยู่ในรูปของค่า Scaled Moment (M(m,n)) ค่า Scaled Central Moment (U(m,n)) ค่า Normalization Scaled Central Moment (V(m,n)) และค่า Moment Invariant (h(i)) ตามลำดับ



a) Rotate A b) A

c) B d) Rotate B

e) C f) Rotate C

รูปที่ 4.3 แสดงภาพตัวอย่างที่ใช้ในการค่า Moment

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Image	A	Rotated A	B	Rotated B	C	Rotated C
M(0,0)	$8.06*10^{3}$	$8.06*10^{3}$	$9.75*10^{3}$	$9.76*10^{3}$	$7.60*10^{3}$	$7.60*10^{3}$
M(1,0)	$0.743*10^{6}$	$0.77*10^{6}$	$0.87*10^{6}$	$1.03*10^{6}$	$0.86*10^{6}$	$0.81*10^{6}$
M(0,1)	$0.837*10^{6}$	$0.76*10^{6}$	$0.99*10^{6}$	$0.89*10^{6}$	$0.79*10^{6}$	$0.85*10^{6}$
M(2,0)	$76.02*10^{6}$	$80.70*10^{6}$	$89.17*10^{6}$	$117.69*10^{6}$	$0.11*10^{9}$	$0.09*10^{9}$
M(0,2)	$93.02*10^{6}$	$79.39*10^{6}$	$109.83*10^{6}$	$91.89*10^{6}$	$88.94*10^{6}$	$102.1*10^{6}$
M(1,1)	$77.10*10^{6}$	$73.00*10^{6}$	$88.90*10^{6}$	$93.2*10^{6}$	$89.12*10^{6}$	$92.16*10^{6}$
M(3,0)	$8.37*10^{9}$	$8.97*10^{9}$	$9.91*10^{9}$	$14.38*10^{9}$	$14.06*10^{9}$	$12.16*10^{9}$
M(0,3)	$11.00*10^{9}$	$8.87*10^{9}$	$12.92*10^{9}$	$10.24*10^{9}$	$10.59*10^{9}$	$13.00*10^{9}$
M(2,1)	$7.91*10^{9}$	$7.72*10^{9}$	$9.06*10^{9}$	$10.56*10^{9}$	$10.95*10^{9}$	$10.97*10^{9}$
M(1,2)	$8.70*10^{9}$	$7.62*10^{9}$	$9.81*10^{9}$	$9.47*10^{9}$	$10.00*10^{9}$	$11.23*10^{9}$

ตารางที่ 4.1 แสดงการค่าScaled Moment (M(m,n)) ของภาพทดสอบ

Image	A	Rotated A	B	Rotated B	C	Rotated C
U(0,0)	$8.07*10^{3}$	$8.07*10^{3}$	$9.75*10^{3}$	$9.75*10^{3}$	$7.61*10^{3}$	$7.60*10^{3}$
U(1,0)	$-3.19*10^{-9}$	$-13.5*10^{-9}$	$-0.43*10^{-9}$	$-2.83*10^{-9}$	$-0.57*10^{-9}$	$0.03*10^{-9}$
U(0,1)	$16.40*10^{-9}$	$5.30*10^{-9}$	$26.82*10^{-9}$	$-51.2*10^{-9}$	$-24.57*10^{-9}$	$-17.2*10^{-9}$
U(2,0)	$7.67*10^{6}$	$6.39*10^{6}$	$10.68*10^{6}$	$9.31*10^{6}$	$9.28*10^{6}$	$8.32*10^{6}$
U(0,2)	$6.37*10^{6}$	$7.70*10^{6}$	$8.55*10^{6}$	$9.97*10^{6}$	$6.62*10^{6}$	$7.61*10^{6}$
U(1,1)	$-13.00*10^{3}$	$10.38*10^{3}$	$-254.0*10^{3}$	$-1.02*10^{6}$	$-0.17*10^{6}$	$1.29*10^{6}$
U(3,0)	$-68.00*10^{6}$	$-1.24*10^{6}$	$-9.87*10^{6}$	$19.20*10^{6}$	$-8.80*10^{6}$	$71.27*10^{6}$
U(0,3)	$1.34*10^{6}$	$-6.81*10^{6}$	$-16.75*10^{6}$	$2.27*10^{6}$	$-36.81*10^{6}$	$-68.17*10^{6}$
U(2,1)	$-0.31*10^{6}$	$-111.0*10^{6}$	$16.28*10^{6}$	$-5.54*10^{6}$	$-54.99*10^{6}$	$-5.01*10^{6}$
U(1,2)	$0.11*10^{9}$	$0.0001*10^{9}$	$10.52*10^{6}$	$-17.91*10^{6}$	$3.49*10^{6}$	$-15.40*10^{6}$

ตารางที่ 4.2 แสดงการค่า Scaled Central Moment (U(m,n)) ของภาพทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Image	A	Rotated A	B	Rotated B	C	Rotated C
V(0,0)	1	1	1	1	1	1
V(1,0)	$-4.42*10^{-15}$	$-0.19*10^{-15}$	$-0.45*10^{-15}$	$2.94*10^{-15}$	$-0.87*10^{-15}$	$0.05*10^{-15}$
V(0,1)	$22.60*10^{-15}$	$7.30*10^{-15}$	$27.86*10^{-15}$	$-53.0*10^{-15}$	$-37.0*10^{-15}$	$-25.9*10^{-15}$
V(2,0)	0.12	0.09	0.11	0.10	0.16	0.14
V(0,2)	0.10	0.12	0.09	0.10	0.11	0.13
V(1,1)	$-0.20*10^{-3}$	$0.15*10^{-3}$	$-2.67*10^{-3}$	$-10.72*10^{-3}$	$-2.88*10^{-3}$	$22.3*10^{-3}$
V(3,0)	$-11.67*10^{-3}$	$-0.21*10^{-3}$	$-1.05*10^{-3}$	$2.03*10^{-3}$	$-1.74*10^{-3}$	$14.09*10^{-3}$
V(0,3)	$0.23*10^{-3}$	$-11.6*10^{-3}$	$-1.79*10^{-3}$	$0.24*10^{-3}$	$-7.29*10^{-3}$	$-13.47*10^{-3}$
V(2,1)	$-53.30*10^{-6}$	$19.00*10^{-3}$	$1.73*10^{-3}$	$-0.59*10^{-3}$	$-10.89*10^{-3}$	$-0.99*10^{-3}$
V(1,2)	$19.00*10^{-3}$	$2.49*10^{-5}$	$1.12*10^{-3}$	$-1.90*10^{-3}$	$-0.69*10^{-3}$	$-3.04*10^{-3}$

ตารางที่ 4.3 แสดงการค่า Normalization Scaled Central Moment (V(m,n)) ของภาพทดสอบ

Image	A	Rotated A	B	Rotated B	C	Rotated C
h1	0.22	0.22	0.20	0.20	0.28	0.27
h2	0.05	0.05	0.04	0.04	0.08	0.08
h3	0.10	0.13	0.07	0.1	0.12	0.14
h4	0.01	0	0	$0.18*10^{-3}$	$0.03*10^{-3}$	$1.80*10^{-3}$
h5	$-26.97*10^{-9}$	$-27.46*10^{-9}$	0	0	0	0
h6	$-6.34*10^{-6}$	$2.70*10^{-6}$	$-0.01*10^{-6}$	$0.02*10^{-6}$	$-2.17*10^{-6}$	$-16.10*10^{-6}$
h7	$0.55*10^{-9}$	$0.34*10^{-9}$	0	0	$-24.59*10^{-9}$	$-197.5*10^{-9}$

ตารางที่ 4.4 แสดงการค่า Moment Invariant (h(i)) ของภาพทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ทดลองหาค่าเฉลี่ยโมเมนท์ของวัตถุแต่ละอันดับ

วัตถุ	A	B	C	D	E	F	G	H
1	572.76	8.12	88.04	35.84	14.25	401.49	27.75	0.27
2	553.69	5.90	92.51	35.95	15.99	356.98	27.75	0.38
3	541.06	6.80	113.01	38.39	14.84	343.84	27.90	0.22
4	565.31	5.79	88.66	40.10	22.75	370.49	26.89	0.16
5	553.37	6.424	86.98	32.86	12.80	355.26	29.32	0.87
6	549.17	6.17	96.44	32.61	20.92	366.39	28.70	0.77
7	567.33	6.23	89.38	43.73	17.50	378.19	21.48	1.36
8	565.59	5.04	109.62	40.57	18.44	367.16	21.14	2.77
9	566.79	7.31	68.90	43.00	24.18	395.83	22.23	0.41
10	566.31	7.45	82.51	43.48	20.00	397.15	26.28	0.52
ค่าเฉลี่ย	560.14	6.52	91.60	38.65	18.17	373.28	25.94	0.77

ตารางที่ 4.5 หาค่าเฉลี่ยโมเมนท์ของวัตถุโดยใช้โมเมนท์อันดับที่ 3

วัตถุ	A	B	C	D	E	F	G	H
1	226.26	216.68	298.86	232.88	245.41	246.83	253.51	260.03
2	228.23	213.21	293.53	235.07	245.45	247.79	254.97	260.91
3	230.15	210.65	298.38	233.99	248.53	248.05	260.75	261.38
4	225.99	212.98	288.05	234.2	245.86	250.98	254.6	252.24
5	224.64	212.6	299.21	232.49	244.05	254.46	257.83	251.79
6	223.98	211.85	291.5	233.35	256.22	271.1	254.6	254.65
7	224.63	212.77	293.37	232.5	252.81	263.14	256.13	255.11
8	222.14	214.9	293.92	236.57	259.45	253.12	249.39	250.57
9	223.18	213.78	294.6	238.97	247.7	264.42	249.25	252.28
10	223.42	216.09	295.3	233.88	252.21	260.55	253.27	256.33
ค่าเฉลี่ย	225.26	213.55	294.67	234.39	249.77	256.04	254.43	255.53

ตารางที่ 4.6 หาค่าเฉลี่ยโมเมนท์ของวัตถุโดยใช้โมเมนท์อันดับที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัตถุ	A	B	C	D	E	F	G	H
1	250.2	0.29	213.5	19.38	55.94	484.93	83.81	5.47
2	262.61	0.34	255.4	17.22	48.75	486.81	87.53	5.61
3	295.29	0.37	603.48	29.99	35.19	429.3	82.23	4.82
4	170.14	0.14	363.21	31.79	47.16	488.23	96.88	5.36
5	231.1	8.21	417.62	22.3	41.01	53.01	88.55	0.21
6	192.94	8.93	259.18	18.68	241.15	93.48	61.87	2.33
7	352.91	7.59	507.75	18.37	238.61	475.84	68.79	3.49
8	246.23	7.34	177.05	17.73	59.07	509.92	22.57	2.52
9	360.42	0.99	246.79	14.96	223.09	556.66	12.66	0.87
10	247.91	3.95	433.48	48.59	33.21	397.47	88.92	3.93
ค่าเฉลี่ย	260.97	3.82	347.74	23.9	102.32	397.56	69.38	3.46

ตารางที่ 4.7 หาค่าเฉลี่ยโมเมนท์ของวัตถุโดยใช้โมเมนท์อันดับที่ 4

วัตถุ	A	B	C	D	E	F	G	H
1	24.57	7.61	544.66	103.02	2027.14	9409.86	669.98	4249.67
2	7.89	11.18	419.42	77.87	1661.23	11290.78	676.56	4712.1
3	34.09	7.72	608.02	75.75	1939.12	10468.48	468.48	4908.09
4	10.45	14.48	446.25	52.56	2142.26	10263.45	582.21	4530.05
5	1.98	26.75	617.74	89.99	2142.26	10351.39	321.66	4635.34
6	48.17	10.17	574.27	47.35	2079.33	10390.34	266.12	4008.78
7	30.18	43.84	586.22	70.46	2165.08	9584.51	364.19	3974.63
8	3.5	1.035	727.51	46.51	2292.39	11745.17	435.36	4639.32
9	5.54	43.65	456.2	37.93	1920.27	9857.7	487.67	4390.82
10	81.04	19.05	444.28	105.4	2036.75	11078.2	424.36	4078.13
ค่าเฉลี่ย	24.74	18.55	542.46	70.69	2040.58	10443.99	469.66	4412.69

ตารางที่ 4.8 หาค่าเฉลี่ยโมเมนท์ของวัตถุโดยใช้โมเมนท์อันดับที่ 2

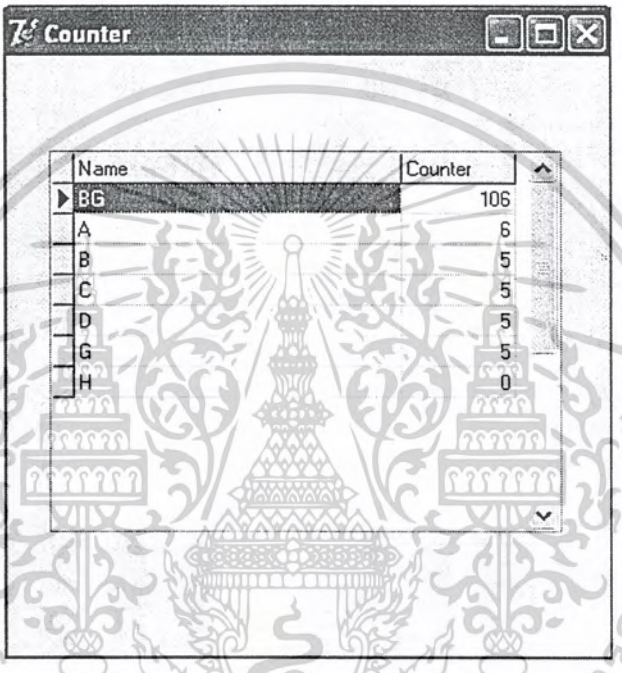
จากการทดลองหาค่าเฉลี่ยของโมเมนท์โดยเราทำการทดลองโมเมนท์อันดับที่1 , 2 , 3 , 4 โดยใช้วัตถุแต่ละชนิดหาค่าโมเมนท์จำนวน 10 ครั้งและหาค่าเฉลี่ยออกมานั้นจะเห็นว่าโมเมนท์อันดับที่ 3 นั้นมีค่าโมเมนท์ที่น่าจะนำไปใช้มาก ดังนั้นจึงเลือกใช้โมเมนท์อันดับที่ 3 .ในการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 ทดลองหาการตรวจจับภาพวัตถุและตรวจนับจำนวนของวัตถุ

โดยนำวัตถุแต่ละชนิดมาทำการวางรางเลื่อนโดยใช้กล้องตรวจจับวัตถุเพื่อใช้แยกชนิดของวัตถุโดยแต่ละชนิดวิ่งบนรางจำนวน 5 รอบ และเราใช้วัตถุจำนวน 5 ชนิด รวมแล้วทดลอง 25รอบ

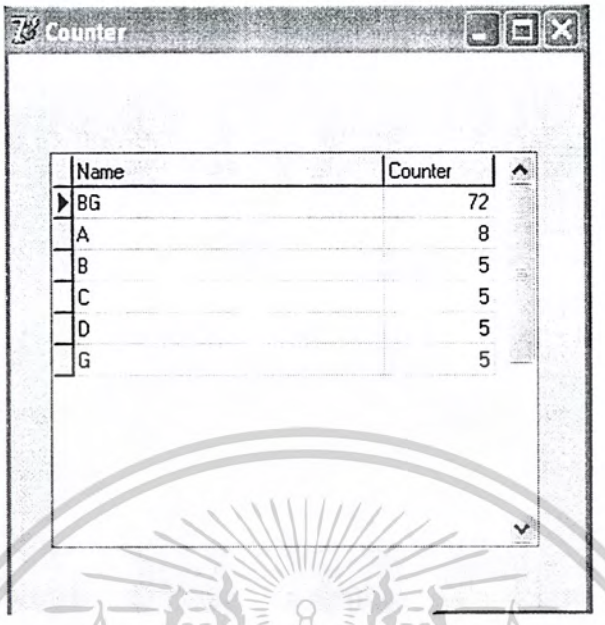
4.5.1 การทดลองครั้งที่ 1 เห็นว่าทำการทดลอง 25 รอบสามารถตรวจจับวัตถุได้ 26 ตัว เนื่องจากว่าวัตถุติดขอบกล้อง 1 ครั้ง ทำให้เกิดข้อผิดพลาดขึ้นทำให้วัตถุ A เกิดนับผิด 1 ตัว ดังแสดงดังรูปที่ 4.4



รูปที่ 4.4 แสดงการทดลองครั้งที่ 1

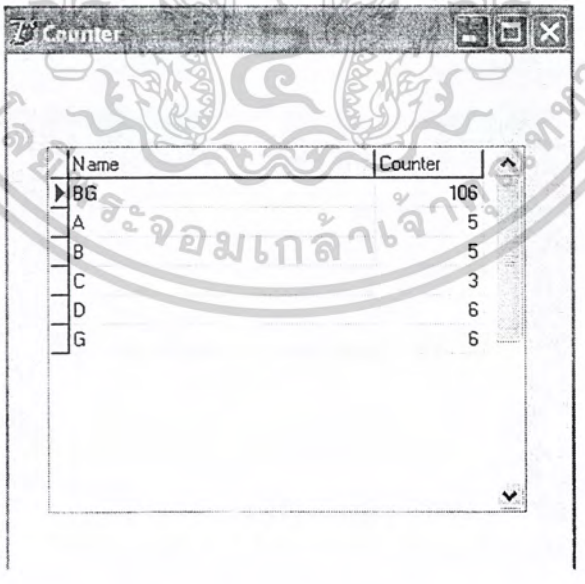
4.5.2 การทดลองครั้งที่ 2 เห็นว่าทำการทดลอง 25 รอบสามารถตรวจจับวัตถุได้ 28 ตัว เนื่องจากว่าวัตถุติดขอบกล้อง 3 ครั้ง ทำให้เกิดข้อผิดพลาดขึ้นทำให้วัตถุ A เกิดนับผิด 3 ตัว ดังแสดงดังรูปที่ 4.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แสดงการทดลองครั้งที่ 2

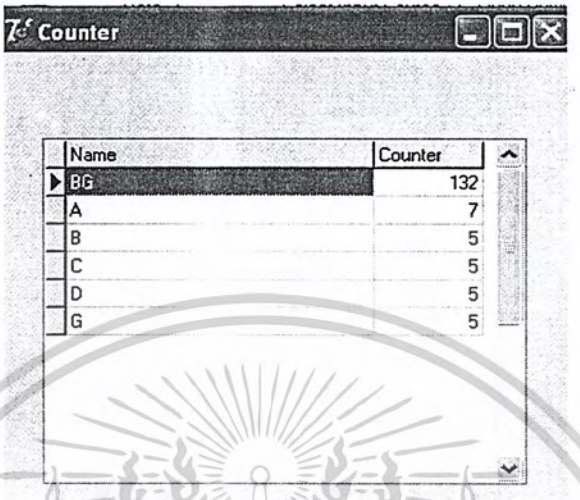
4.5.3 การทดลองครั้งที่ 3 เห็นว่าทำการทดลอง 25 รอบสามารถตรวจจับวัตถุได้ 25 ตัว ทำให้เกิดข้อผิดพลาดขึ้นทำให้วัตถุ C เกิดนับผิดออกวัตถุ D 2 ตัวและวัตถุ D เกิดนับผิดออกวัตถุ G 1 ตัว ดังแสดงดังรูปที่ 4.6



รูปที่ 4.6 แสดงการทดลองครั้งที่ 3

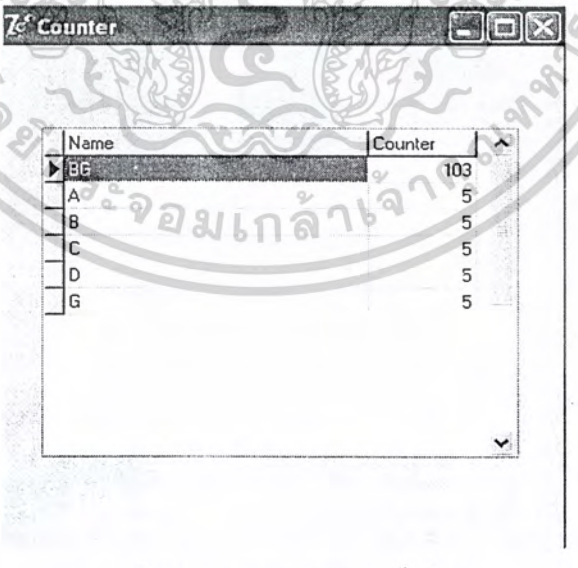
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.4 การทดลองครั้งที่ 4 เห็นว่าทำการทดลอง 25 รอบสามารถตรวจจับวัตถุได้ 27 ตัว เนื่องจากว่าวัตถุติดขอบกล้อง 2 ครั้ง ทำให้เกิดข้อผิดพลาดขึ้นทำให้วัตถุ A เกิดนับผิด 2 ตัว ดังแสดงดังรูปที่ 4.7



รูปที่ 4.7 แสดงการทดลองครั้งที่ 4

4.5.5 การทดลองครั้งที่ 5 เห็นว่าทำการทดลอง 25 รอบสามารถตรวจจับวัตถุได้ 25 ตัว แสดงว่าสามารถตรวจจับวัตถุได้ทุกตัว ดังแสดงดังรูปที่ 4.8



รูปที่ 4.8 แสดงการทดลองครั้งที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปและวิจารณ์ผลการทดลอง

5.1 วิจารณ์ผลการทดลอง

ในการเตรียมภาพนี้จะนำไปสู่การประมวลผลเพื่อจำแนก ว่าวัตถุที่ผ่านเข้ามาเป็นอะไรนั้น เนื่องจากภาพที่ได้นั้นอาจไม่สมบูรณ์ จนไม่สามารถจำแนกได้ว่าวัตถุที่เข้ามาคืออะไร ทั้งนี้ทั้งนั้นเนื่องมาจากสาเหตุหลายประการ อันได้แก่

- ขนาดของภาพที่ได้ไม่เหมาะสม ซึ่งเราสามารถแก้ไขได้โดยการทำการปรับขนาดภาพก่อนการประมวลผล
- ภาพที่ได้ขาดหายไปบางส่วนทำให้ค่าโมเมนท์ที่ได้ไม่ถูกต้องทำให้ไม่สามารถจำแนกได้ภาพนั้นได้
- ความสว่างของภาพที่ได้ไม่เหมาะสมทำให้การค่าที่ได้ไม่ถูกต้อง
- ความสามารถของกล้องที่นำมาถ่ายเนื่องจากกล้องไม่สามารถปรับโฟกัสในระยะใกล้ๆได้ จึงทำให้ภาพที่ได้นั้นมัวไม่ชัดเจน
- ไม่สามารถจำได้เนื่องจากไม่มีข้อมูลในDatabase ซึ่งสามารถแก้ปัญหาได้โดยการเพิ่มข้อมูลภาพนั้นๆ ไป

ดั้งนั้นสาเหตุทั้งหมดเป็นข้อจำกัดในทางปฏิบัติ ซึ่งเป็นสิ่งสำคัญในการออกแบบระบบให้สามารถใช้งานได้จริง โดยเฉพาะในส่วนของการรับข้อมูลภาพและการประมวลผลขั้นต้น ซึ่งต้องออกแบบให้ครอบคลุมถึงข้อจำกัดเหล่านั้น เพื่อที่จะทำระบบสามารถทำงานได้อย่างถูกต้องแม่นยำและมีประสิทธิภาพ ซึ่งจะต้องทำการปรับปรุงโปรแกรมที่มีอยู่ให้สมบูรณ์ยิ่งขึ้น ซึ่งจะเป็นแนวทางที่ต้องพัฒนาต่อไปในอนาคต

จากการนำไปประยุกต์ใช้งาน ผลการทดลองปรากฏว่าโปรแกรมสามารถนับจำนวนวัตถุได้อย่างถูกต้องถ้าวัตถุนั้นไม่อยู่ที่ขอบของกล้องขณะประมวลผล ซึ่งในการทำงานเราจะต้องปรับความสว่าง แล้วเลือกค่าเทรชโฮลที่เหมาะสม และเลือกค่าโมเมนท์ที่จะใช้ในการประมวลผลของวัตถุแต่ละชุด ซึ่งการเลือกค่าโมเมนท์หรือก็คือการเลือกอันดับ(Order) ของโมเมนต์นั่นเอง โดยสามารถหาได้จากการที่เราทำการหาค่าเฉลี่ยของโมเมนท์ในแต่อันดับ(Order) ของวัตถุในแต่ละชิ้น ซึ่งค่าเฉลี่ยนี้จะเป็นข้อมูลในฐานข้อมูล(Database) ที่จะนำไปใช้ในการเปรียบเทียบหรือจำแนกวัตถุ

จากผลการทดลองก่อนการใส่ชุดไมโครคอนโทลเลอร์ควบคุมมอเตอร์ผลปรากกฏว่าส่วนใหญ่วัตถุที่ต้องการนับและจำแนก จะอยู่ที่ขอบของภาพทำให้ได้ค่าโมเมนท์ที่ผิดพลาดไป เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใหญ่วัตถุที่ต้องการนับและจำแนก จะอยู่ที่ขอบของภาพทำให้ได้ค่าโมเมนท์ที่ผิดพลาดไป เป็นสาเหตุทำให้เกิดการนับและจำแนกวัตถุผิดพลาดไป แต่เมื่อเราทำการใส่ชุดควบคุมมอเตอร์จะทำให้ค่าผิดพลาดนี้มีค่าน้อยลงหรือคือวัตถุจะอยู่ที่ขอบของภาพน้อยลง แต่ก็ยังมีบางครั้งที่วัตถุอยู่ที่ขอบของภาพ ซึ่งในส่วนของค่าผิดพลาดที่ยังมีอยู่นี้สามารถแก้ไขได้โดยการวางวัตถุในแต่ละครั้งจะต้องสัมพันธ์กับการทำงานของโปรแกรม หรือเราอาจจะแก้ไขโปรแกรมให้ทำการตรวจเช็ควัตถุก่อนการประมวลผลหาค่าโมเมนท์ว่าวัตถุนั้นอยู่ที่ขอบของภาพหรือไม่ ซึ่งการทำเช่นนี้จะต้องใช้ทฤษฎีที่สูงเข้ามาช่วยในการจำแนก ซึ่งในการทดลองของโปรแกรมนี้ไม่มีส่วนของการตรวจเช็คภาพที่ขอบก่อนการประมวลผล แต่เราจะใช้โปรแกรมในการควบคุมชุดไมโครคอนโทลเลอร์ผ่านทางพอร์ตอนุกรมให้หยุดวัตถุให้อยู่นิ่งแล้วค่อยทำการประมวลผลหาค่าโมเมนท์

จากผลการทดลองตรวจนับวัตถุที่เป็น สามมิติยังคงมีค่าผิดพลาดอยู่มากทั้งนี้เนื่องจากหลักการและทฤษฎีของการใช้ค่าโมเมนท์ในการจำแนกวัตถุมีเงื่อนไขที่ว่าวัตถุที่จะใช้ในการตรวจและจำแนกนั้นจะต้องมีรูปร่างที่แบน ซึ่งนี้เป็นข้อจำกัดที่สำคัญมากเนื่องจากวัตถุโดยทั่วไปแล้วไม่เป็นวัตถุแบน

5.2 สรุปผลการทดลอง

จากผลการทดลองโปรแกรมการจดจำวัตถุนั้น จึงสรุปได้ว่าการใช้โมเมนท์นั้นสามารถจำแนกวัตถุต่างๆได้อย่างมีประสิทธิภาพ โดยการจำแนกนั้นจะไม่ขึ้นอยู่กับ การเคลื่อนที่ (Translation), การหมุน (Rotation), และการขยายของภาพ (Scaling)

ซึ่งโดยรวมแล้วจะการจดจำและจำแนกภาพโดยวิธีนี้ มีประสิทธิภาพมาก สามารถนำไปประยุกต์ใช้งานได้จริง

จากการนำไปประยุกต์ใช้งานในการนับและจำแนกวัตถุโดยใช้ทฤษฎีของการหาค่าโมเมนท์ในการจำแนกวัตถุมีเงื่อนไขคือ วัตถุที่จะใช้ในการจำแนกจะต้องมีลักษณะที่แบน และวัตถุนั้นจะต้องไม่อยู่ที่ขอบของภาพขณะทำการหาค่าโมเมนท์ ถ้าทำได้ตามเงื่อนไขดังกล่าวของการทดลองตรวจจับและจำแนกวัตถุจะมีความถูกต้องและมีประสิทธิภาพสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

1. ชูชาติ ปิณฑวิรุจน์, เอกสารประกอบการสอนวิชา "Digital Image Processing
2. อรพินท์ ฐิตรัตนพลและ เอกวัฒน์ เบญจพรกุลนิจ, ปริญญานิพนธ์เรื่อง "ระบบจดจำทะเบียนรถยนต์"
3. Gonzalez, and R.E. Wood, **Digital Image Processing**, Addison-Wesley, 1992
4. Wayne Niblack, **An Introduction to Digital Image Processing**, Prentice Hall, 1986

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้