ระบบประมวลผลภาพสำหรับหุ่นยนต์สำรวจ

Image Processing System for Observation Robot

โดย

นาย นิสิทธิ์ สมิทธ์สมบูรณ์

นาย ชาตรี พันธุ์อัจฉริยกุล

นาย ชวาล วรุณเทพรักษา

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2536

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

033166

ปริญญานิพนธ์ปีการศึกษา 2536
ภาควิชา วิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เรื่อง ระบบประมวลผลภาพสำหรับหุ่นยนต์สำรวจ

ผู้จัดทำ

1. นายนิสิทธิ์ สมิทธ์สมบูรณ์
2. นายชาตรี พันธุ์อัจฉริยกุล
3. นายชวาล วรุณเทพรักษา

..... อาจารย์ที่ปรึกษา

( )

อาจารย์ที่ปรึกษา

( )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบประมวลผลภาพสำหรับหุ่นยนต์สำรวจ

Image Processing System for Observation Robot

โดย

นาย ชวาล วรุณเทพรักษา 33100076

นาย ชาตรี พันธุ์อัจฉริยกุล 33100090

นาย นิสิทธิ์ สมิทธ์สมบูรณ์ 33100180

อาจารย์ที่ปรึกษา รศ. ครรชิต ไมตรี

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของโครงงานวิจัยสร้างต้นแบบระบบประมวลผลสำหรับหุ่นยนต์สำรวจ ซึ่งเป็นหนึ่งใน 4 ส่วนของโครงงานสร้างต้นแบบระบบหุ่นยนต์สำรวจ ซึ่งดำเนินงานโดย 4 ภาควิชา คือ ภาควิชาวิศวกรรมคอมพิวเตอร์,วิศวกรรมระบบควบคุม,วิศวกรรมการวัดคุมฯ และวิศวกรรมโทรคมนาคม โดยโครงงานในส่วนของภาควิชาวิศวกรรมคอมพิวเตอร์นี้มีจุดมุ่งหมายเพื่อให้หุ่นยนต์สามารถรับรู้และเข้าใจความหมายของภาพที่เห็นได้ เพื่อนำไปสู่การตัดสินใจต่าง ๆ ตามความเหมาะสมของเหตุการณ์

ปริญญานิพนธ์นี้จะกล่าวถึง วิธีการรับและแปลงสัญญาณภาพเป็นภาพดิจิตอล (Digital Image),การแสดงผลภาพ,การประมวลผลภาพวิธีต่าง ๆ,การวิเคราะห์หาองค์ประกอบที่สำคัญของภาพ และการใช้ระบบโครงข่ายประสาทเทียม (Neural Network) ซึ่งเป็นเทคโนโลยีใหม่ในการวิเคราะห์หาความหมายของภาพ

Abstract

This thesis is a part of the Image Processing for Observation Robot Prototype Research,which is one of four topics of Observation Robotic System Project operated by four engineering departments:Computer Department,Control Department,Instrument Department and Telecomunication Department. In terms of Computer Department, the objective of the project is to study Robot's image conception and understanding for making a suitable decision.

In this thesis would contain the topics of detecting and modulating analog to digital image,studying the output display and certain types of Image Processing and analyzing of image's components and Neural Network for image's meaning analysis.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

content

บทที่ หน้า

1.บทนำ

1.1 ส่วนวิเคราะห์ข้อมูลจากภาพ 1

1.1.1 ส่วนรับภาพจากการ์ด 1

1.1.2 ส่วนแสดงผล 1

1.1.3 ส่วนวิเคราะห์แยกองค์ประกอบของภาพ 2

1.1.4 ส่วนวิเคราะห์องค์ประกอบของภาพ(บล็อบ(Blob)) 2

1.1.5 ส่วนวิเคราะห์ความหมายของข้อมูลภาพ 2

1.2 ส่วนติดต่อกับตัวหุ่นยนต์ 3

2.ทฤษฎีและหลักการ

2.1 ส่วนรับภาพจากการ์ด 4

2.1.1 แผงวงจร DigiImage 5

2.2 ส่วนแสดงผลภาพ 18

2.2.1 การติดต่อกับระบบฮาร์ดแวร์ของการ์ดวีจีเอ (VGA Card Interfacing) 18

2.3 ส่วนวิเคราะห์แยกองค์ประกอบของภาพ (Image Segmentation) 27

2.3.1 Image Enhancement 27

2.3.2 วิธีการของ Image Enhancement 27

2.3.3 ทำไมจึงต้องทำอิมเมจเซกเมนเตชัน 28

2.3.4 หลักการของอิมเมจเซกเมนเตชัน 28

2.4 ส่วนวิเคราะห์องค์ประกอบของภาพ(บล็อบ(Blob)) 41

2.4.1 การวิเคราะห์ความต่อเนื่องของจุด 41

2.4.2 บล็อบ (Blob) 41

2.4.3 ปัญหาการต่อของจุด (connectivity) 43

2.4.4 การวิเคราะห์ความต่อเนื่องของจุด (connectivity analysis) 45

2.5 ส่วนวิเคราะห์ความหมายของข้อมูลภาพ 49

2.5.1 โครงข่ายประสาทเทียม(Artificial Intelligence Network). 49

2.5.2 ส่วนประกอบของเซลล์ประสาท 49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.3 ข้อแตกต่างระหว่างโครงข่ายประสาทเทียมและ
ปัญญาประดิษฐ์ 50
2.5.4 แบบจำลองทางคณิตศาสตร์ของนิวรอลเน็ตเวร์ค 50
2.5.5 Back Propagation Neural Network 52
2.6 ส่วนเชื่อมต่อกับตัวหุ่นยนต์ 55

3.การคำนวณและการสร้าง
3.1 โปรแกรมรับภาพ 56
3.2 โปรแกรมแสดงผลภาพ 56
3.3 ส่วนวิเคราะห์แยกองค์ประกอบของภาพ 58
3.4 ส่วนวิเคราะห์บล็อบภาพ 61
3.5 ส่วนส่วนวิเคราะห์ความหมายของข้อมูลภาพ 62
3.6 ส่วนการติอต่อกับตัวหุ่นยนต์ 63
3.6.1 ส่วนรับสัญญาณภาพจากกล้องวีดีโอ 63
3.6.2 ส่วนสื่อสารคอมมานด์และรับข้อมูลจากเซนเซอร์ 64

4. การทดลองและผลการทดลอง
4.1การทดลองเกี่ยวกับการรับภาพ 66
4.2 การทดลองปรับปรุงคุณภาพของภาพ 66
4.3 การทดลองหาวิธีปรับค่าอัตราการเรียนรู้ของโครงข่าย
สมองเทียมโดยอัตโนมัติ 71

5. บทวิจารณ์และสรุป
5.1 ผลการวิเคราะคุณสมบัติของการประมวลผลภาพด้วยวิธีการต่าง ๆ .. 73
5.1.1 Sum Square Delta 73
5.1.2 Zero Crossing 73
5.1.3 Shortest Spanning Tree(SST) 73
5.1.4 Low Pass filter 74
5.1.5 Histogram Equalization 74
5.2 แนวทางในการพัตนาต่อเพื่อเพิ่มประสิทธิภาพในการหาขอบภาพ ... 74
5.2.1 การเพิ่มประสิทธิภาพของวิธี Sum Square Delta 75
5.2.2 Zero Crossing 75
5.2.3 Shortest Spanning Tree(SST) 75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.4 Threshold 75
5.3 สรุปผลการทดลองหาวิธีปรับค่าอัตราการเรียนรู้ของโครงข่าย
สมองเทียมโดยอัตโนมัติ 76
5.4 สรุประบบการทำงานในส่วนของการประมวลผลภาพ 76
5.5 สรุปผลโครงงาน 76
เอกสารอ้างอิง
กิตติกรรมประกาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปภาพ **หน้า**

รูปที่ 2.2.1 โครงสร้างหน่วยความจำแบบเชิงเส้นในโหมด 13 H 20
รูปที่ 2.2.2 การจัดแบ่งชุดของรีจิสเตอร์สี 22
รูปที่ 2.2.3 การตั้งค่าชุดของรีจิสเตอร์สี 23
รูปที่ 2.3.1 รูปกราฟระหว่างค่าความสว่างกับจำนวนจุด โดยใช้หลักการ ฮีสโตแกรม 30
รูปที่ 2.3.2 รูปกราฟตัวอย่างที่เกิดจากทฤษฎีกราฟ ที่ใช้ในการทำเซกเมนเตชันภาพ 32
รูปที่ 2.3.3 รูปกราฟย่อยที่เป็นช็อตเตสท์สแพนนิ่งทรี 33
รูปที่ 2.3.4 รูปแสดงการแปลงข้อมูลภาพไปเป็นกราฟ 34
รูปที่ 2.3.5 โฟลว์ชาร์ตของการหาช็อตเตสท์สแพนนิ่งทรี 35
รูปที่ 2.3.6 รูปแสดง SST ของกราฟและการได้มาของภาพเซกเมนเตชัน 37
รูปที่ 2.3.7 ภาพเซกเม้นต์ที่ได้จากข้อมูลในรูปที่ 2.3.6 38
รูปที่ 2.4.1 ตัวอย่างภาพที่แสดงความสัมพันธ์ตามลำดับชั้นระหว่างบล็อบ 42
รูปที่ 2.4.2 รูปแบบการต่อของจุดแบบต่าง ๆ 42
รูปที่ 2.4.3 ตัวอย่างภาพแสดงปัญหาการของต่อของจุด 43
รูปที่ 2.4.4 รูปแสดงฟิลด์ต่าง ๆ ในเรคอร์ดของบล็อบ 44
รูปที่ 2.4.5 ตัวอย่างการเก็บข้อมูลของภาพในลักษณะเรคอร์ด ของ run-length 46
รูปที่ 2.4.6 รูปแสดงค่าของวินโดว์ในสถานะต่าง ๆ 16 สถานะ 47
รูปที่ 2.4.7 ตัวอย่างภาพที่ใช้อธิบายวิธีการทำงาน 48
รูปที่ 2.5.1 โครงสร้างและส่วนประกอบของเซลล์ประสาททั่วไป 49
รูปที่ 2.5.2 แบบจำลองเชิงคณิตศาสตร์ของเซลประสาทซึ่งเสนอโดย McCulloch และ Pits 51
รูปที่ 2.5.3 ตัวอย่างการจับกลุ่มของเซลล์ประสาท 2.5.3 ชั้น ของโครงข่าย ประสาทเทียมที่มีการเรียนรู้แบบแพร่กลับ และสัญญาณเข้าออกของแต่ละเซลล์ 52
รูปที่ 2.5.4 ความหมายของ ω ji 52
รูปที่ 3.3.1 ภาพแสดงเมตริกที่ใช้ในการประมวลผล 58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.6.1 ภาพแสดงวิธีการติดต่อส่งผ่านข้อมูลภาพระหว่างหุ่นยนต์กับเครื่องคอมพิวเตอร์ควบคุม 63
รูปที่ 3.6.2 ภาพแสดงวิธีการส่งข้อมูลที่ใช้ควบคุมหุ่นยนต์โดยเครื่องคอมพิวเตอร์ 65
รูปที่ 4.1 67
รูปที่ 4.2 68
รูปที่ 4.3 69
รูปที่ 4.4 70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง หน้า

ตารางที่ 2.5.1 ข้อแตกต่างระหว่างโครงข่ายประสาทเทียม และปัญญาประดิษฐ์ 50

ตารางที่ 3.6.1 รูปแบบของคอมมานด์ในการติดต่อกับหุ่นยนต์ 64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

โครงงานนี้เป็นโครงงานร่วมระหว่าง 4 ภาควิชา โดยมีจุดประสงค์ที่จะสร้างระบบจำลอง ของหุ่นยนต์สำรวจ (Observation Robot) ซึ่งในส่วนของภาควิชาคอมพิวเตอร์นี้ รับผิดชอบในส่วนของการสร้างระบบวิเคราะห์ภาพที่รับมาจากหุ่นยนต์ โดยรับผ่านทางกล้องวีดีโอ นอกจากนี้ยังรับผิดชอบในการส่วนของการติดต่อระหว่างผู้ใช้กับตัวหุ่นยนต์ ซึ่งงานในส่วนที่รับผิดชอบนี้จะเป็นส่วนของโปรแกรม(Software) ทั้งสิ้นซึ่ง โดยแบ่งฟังก์ชันการทำงาน ออกได้เป็น 2 ส่วนใหญ่ ๆคือ ส่วนวิเคราะห์ข้อมูลภาพ และส่วนการเชื่อมต่อ (Interface) กับหุ่นยนต์

1.1 ส่วนวิเคราะห์ข้อมูลจากภาพ

โดยในโปรแกรมระบบนี้จะแบ่งงานออกเป็น 5 ส่วน คือ

- ส่วนรับภาพ
- ส่วนแสดงผล
- ส่วนหาองค์ประกอบสำคัญของภาพ (Image Segmentation)
- ส่วนคำนวนหาข้อมูลบล็อบ(Blob) ภาพ

ส่วนวิเคราะห์ความหมายของภาพโดยใช้ โครงข่ายประสาทเทียม(Neural Network)

โดย 4 ส่วนแรกเป็นส่วนของการเตรียมข้อมูลสำหรับการประมวลผล(Pre - processing) ส่วนโครงข่ายประสาทเทียม เป็นส่วนของการวิเคราะห์ข้อมูล

1.1.1 ส่วนรับภาพจากการ์ด

จะทำหน้าที่รับภาพจากสัญญาณวิดิโอ ซึ่งมาจากกล้องถ่ายภาพวิดิโอเข้าสู่โปรแกรมโดยจะรับข้อมูลจากการ์ด DigiImage ซึ่งเป็นการ์ดสำหรับแปลงสัญญาณวิดิโอให้เป็น Digital Image ซึ่งมีคุณสมบัติรับภาพได้ความระเอียดสูงสุด 256*256 จุด และจำนวนสีมากที่ทั้งหมด 64(ระดับสีแดง)*64(ระดับสีเขียว)*64(ระดับสีน้ำเงิน) ซึ่งเราจะนำภาพที่ได้นี้ไปแปลงเป็นภาพขนาด 256*256 จุดและระดับสี 256 ระดับ(grey level) เพื่อส่งให้ส่วนต่อ ๆไปนำไปประมวลผล

1.1.2 ส่วนแสดงผล

จะทำหน้าที่แสดงผลของการวิเคราะห์ข้อมูลต่าง ๆ โดยแสดงออกทางจอภาพ ซึ่งข้อมูลที่ถูกแสดงจะมีหลายอย่างเช่น ภาพที่รับมาจากกล้องวีดีโอ ผลจากการวิเคราะห์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพ สถานะการทำงานของตัวหุ่นยนต์ และข้อมูลจากเซนเซอร์ต่าง ๆ ซึ่งติดตั้งอยู่ที่ตัวหุ่นยนต์

1.1.3 ส่วนวิเคราะห์แยกองค์ประกอบของภาพ

ทำหน้าที่วิเคราะห์แยกหรือกลั่นกรองส่วนประกอบต่าง ๆของภาพออกมาให้เหลือเฉพาะรอยต่อระหว่างส่วนต่าง ๆของภาพ แล้วจึงส่งภาพที่ได้นั้นไปยังส่วนแยกองค์ที่ทำหน้าที่วิ-เคราะห์ความหมายของภาพที่รับเข้ามา ในขั้นตอนนี้นั้นจะนำภาพที่ได้จากขั้นตอนแรกมาทำการประมวลผลซึ่งจะได้ผลรับเป็นภาพไบนารี(Binary imgae)

1.1.4 ส่วนวิเคราะห์องค์ประกอบของภาพ(บล็อบ(Blob))

จะนำ ภาพไบนารี ที่ได้จากขั้นตอนที่ 1.1.3 มาทำการวิเคราะห์หาบล็อบของภาพ บล็อบภาพ คือองค์ประกอบต่าง ๆในภาพ ซึ่งประกอบไปด้วยจุดหลาย ๆจุดต่อเนื่องกันรวมกันเป็นองค์ประกอบต่าง ๆ เราจะทำการตรวจหาบล็อบภาพว่าในภาพมีบล็อบรูปร่างอย่างไรบ้าง โดยจะเก็บข้อมูลองค์ประกอบของแต่ละบล็อบ บล็อบละ 17 อย่าง ยกตัวอย่างเช่น พิกัด สี พื้นที่ จำนวนบล็อบลูก และค่าทางสถิติต่าง ๆ ของแต่ละบล็อบ โดยในแต่ละรูปจะเลือกประมวลผลเฉพาะ 50 บล็อบแรกซึ่งมีขนาดพื้นที่สูงสุด จากนั้นจะส่งข้อมูลบล็อบ ขนาด 50*17 ตัวนี้ไปให้แก่ส่วนที่ 1.1.5 ซึ่งเป็นส่วนวิเคราะห์ความหมายของภาพ โดยรายละเอียดจะนำเสนอในส่วนต่อไป

1.1.5 ส่วนวิเคราะห์ความหมายของข้อมูลภาพ

ส่วนนี้จะทำหน้าที่วิเคราะห์ข้อมูลองค์ประกอบของบล็อบต่าง ๆที่ได้จากส่วนที่ 3 โดยในการวิเคราะห์หาความหมายของภาพนั้น เราต้องการให้มีความยืดหยุ่นคือสามารถใช้หาความหมายของภาพได้ในหลาย ๆอย่างหลาย ๆแบบ การวิเคราะห์ภาพโดยการใช้โปร-แกรมธรรมดาจึงเป็นการยุ่งยากเนื่องจากหากต้องการวิเคราะห์ความหมายใหม่ ๆก็จำเป็นที่จะต้องเขียนโปรแกรมในส่วนนี้ใหม่

ฉะนั้นการทำงานในส่วนนี้จึงใช้โครงข่ายประสาทเทียมหรือนิวรอลเน็ตเวอร์ก (Neural Network) โดยที่เราสามารถที่จะนำนิวรอลเน็ตเวอร์กนี้ไปวิเคราะห์ความหมายของภาพในแง่ใดก็ได้ที่ไม่ซับซ้อนจนเกินไป แต่จะต้องทำการสอนตัวนิวรอลเน็ตเวอร์กสำหรับความหมายนั้น ๆเสียก่อน โดยในการทำโครงงานครั้งนี้จะทดลองวิเคราะห์ความหมายของภาพว่า เป็นภาพของหน้าคนหรือไม่ โดยจะจำกัดขอบเขตเบื้องต้นว่าจะต้องเป็นภาพหน้าตรงขนาด 1/2 ของจอภาพขึ้นไปและพื้นหลังของภาพจะต้องเรียบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เหตุที่ทดลองวิเคราะห์ภาพหน้าคนนี้ เนื่องจากภาพหน้าคนเป็นสิ่งที่สามารถทำความเข้าใจได้ยากกว่าวัตถุอื่นทั่ว ๆไป ฉะนั้นจึงกล่าวได้ว่าหากสามารถทำการตรวจจับหน้าคนได้ ก็จะหมายความว่าเราจะนำระบบนี้ไปใช้กับงานอื่น ๆที่ง่ายกว่าได้ด้วย เช่นการให้หุ่นยนต์เดินตามลูกศร หรือ เส้นตรง การสั่งงานในใช้ระบบภาพ การตรวจสอบเครื่อง ๆจักรในสภานที่อันตราย และนำไปสู่การสร้างระบบหุ่นยนต์สำรวจที่สามารถวิเคราะห์ภาพของสิ่งที่มีความซับซ้อนยิ่งขึ้นไปอีกได้

1.2 ส่วนติดต่อกับตัวหุ่นยนต์

ส่วนนี้ทำหน้าที่ติดต่อกับตัวหุ่นยนต์ซึ่งสร้างโดยนักศึกษาในส่วนของภาควิชาวิศวกรรมระบบควบคุม ซึ่งในการติดต่อนั้นมีการสั่งการและการขอรับข้อมูลต่าง ๆ โดยในขณะนี้สามารถสั่งงานได้โดยการสั่งเดินหน้า,ถอยหลัง,เลี้ยวซ้าย-ขวา และในส่วนของการรับข้อมูลนั้นสามารถรับระยะห่างจากวัตถุใน 3 ทิศทาง(ด้านหน้า และด้านซ้าย-ขวา)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

จากที่ได้กล่าวไว้เมื่อข้างต้นแล้วว่า โปรแกรมในระบบนี้นั้น จะประกอบด้วยส่วนการทำงานต่างหลายส่วน คือ ส่วนรับภาพ ส่วนแสดงผล ส่วนวิเคราะห์แยกองค์ประกอบของภาพ ส่วนวิเคราะห์องค์ประกอบของภาพ (บล็อบ(Blob)) ส่วนวิเคราะห์ความหมายของข้อมูลภาพ และส่วนติดต่อกับตัวหุ่นยนต์ ซึ่งในบทนี้จะได้นำเสนอรายละเอียดเพิ่มเติมเกี่ยวกับการทำงานของแต่ละส่วน

2.1 ส่วนรับภาพจากการ์ด

ส่วนนี้ทำการรับข้อมูลภาพจากการ์ด DigiImage โดยการติดต่อกับการ์ด Digi Image จะติดต่อสั่งงานโดยใช้การรับส่งข้อมูลจากพอร์ต(IN / OUT PORT) โดยการ์ด DigiImage จะ map address ของหน่วยความจำของการ์ดไว้ 16 KByte โดยหลังจากถ่ายภาพแล้ว เราสามารถที่จะทำการอ่านข้อมูลรูปภาพออกจากการ์ดได้โดยการอ่านภาพครั้งละ 16 KByte ซึ่งจะต้องทำงานทั้งหมด 4 ครั้งสำหรับ 1 สี(64 KByte) ดังนั้นในการอ่านข้อมูลภาพสีจึงต้องอ่านทั้งหมดเท่ากับ 3 color * 4 page รวมทั้งหมดอ่านทั้งหมด 12 ครั้ง หรือสำหรับภาพขาวดำ 64 ระดับเทา (gray level) จะต้องอ่านจากทั้งหมด 4 page สำหรับ 1 ภาพ

ในโปรแกรมของเรานี้จะใช้การอ่านภาพสี เพื่อนำมาประมวลผล และการอ่านภาพขาวดำซึ่งมี 64 ระดับเทา ไว้สำหรับการแสดงผลบนหน้าจอให้ผู้ที่ใช้สามารถเห็นภาพบนจอภาพได้

ในส่วนการรับภาพจากการ์ดนี้ จะมี 2 ส่วนคือ ส่วนหนึ่งรับภาพขาวดำ 64 ระดับเพื่อใช้ในแสดงสู่จอภาพ อีกส่วนหนึ่งใช้ในการรับข้อมูลภาพสีจริงแล้วแปลงเป็นภาพขาวดำ 256 ระดับ เพื่อใช้ในการวิเคราห์ภาพในขบวนการอื่นๆ โดยการเก็บภาพเพื่อทดสอบนี้ เราจะเก็บภาพโดยอ่านโหมดของการถ่ายภาพสี ซึ่งจะได้สีแดง 64 ระดับ สีเขียว 64 ระดับและสีน้ำเงิน 64 ระดับ จากนั้นเราจะทำการแปลงเป็น 256 ระดับเทา โดยการ mark บิทที่ 0 และบิทที่ 1 ของสีทั้ง 3 (แดง,เขียว,น้ำเงิน) ซึ่งไม่ได้ใช้งานให้มีค่าเป็น 0 หลังจากนั้นจึงนำมาผ่านฟังก์ชั่นแปลงดังนี้

$$Value=(แดง)*0.30 + (เขียว) * 0.59 + (น้ำเงิน)*0.11$$

จึงสิ้นสุดการทำงานของส่วนเก็บภาพเพื่อใช้ในการวิเคราะห์ ภาพที่ผ่านขบวนนี้จะถูกนำไปเก็บลงสู่ไฟล์ในรูปของบิทแมพ(Bitmap) หรือถูกส่งต่อไปยังขบวนการต่างๆเพื่อวิเคราะห์ผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากแผนวงจร DigiImage เป็นส่วนประกอบสำคัญในการรับภาพ ดังนั้นจึงได้เสนอ คุณสมบัติต่าง ๆ หลักการทำงาน การติดตั้ง การติดตั้งต่อ รวมทั้งการควบคุมสั่งงานแผนวงจร DigiImage ไว้เพิ่มเติมด้วย

2.1.1 แผงวงจร DigiImage

1. คุณสมบัติของแผงวงจร DigiImage

1.1 การถ่ายภาพ

1.บนแผงวงจรมีหน่วยความจำสำหรับภาพขนาด 192K ไบต์ ซึ่งไมโครคอมพิวเตอร์สามารถ ติดต่อกับหน่วยความจำนี้ได้โดยตรง

2.วงจรถ่ายภาพเป็นแบบ real time สำหรับภาพขาวดำ และเกือบ real rime สำหรับภาพสี

3.สามารถดูภาพ LIFE TIME จากจอ VGA ได้ขณะถ่าย (digitize)

4.การปรับแต่งภาพสามารถควบคุมความสว่าง (brightness) ,ความชัดเจน (contrast) และความเข้มของสี (saturation) ได้จากซอฟท์แวร์

5.รับสัญญาณอินพุท แบบ COMPOSITE ขาวดำ หรือสี (ระบบ PAL) ก็ได้ เช่นสัญญาณจากกล้องถ่ายวีดีโอ

1.2 ความละเอียดของภาพ

1. ความละเอียดของภาพสีได้ถึง 256 * 256 จุด 64*64*64 สี
2. ความละเอียดของภาพขาวดำได้ถึง 256*256 จุด 64 ระดับเทา

2. หลักการทำงานของแผงวงจร DigiImage

แผงวงจร DigiImage ปรกอบด้วยส่วนต่าง ๆที่สำคัญ ดังนี้ คือ

1. หน่วยความจำสำหรับเก็บภาพอิมเมจ(RAM) 3 Bank Bank ละ 64K ไบต์
2. ส่วนแปลงสัญญาณวีดีโอจากอนาล็อกให้เป็นข้อมูลดิจิตอล (A/D)
3. ส่วนแยกสัญญาณสี
4. ส่วนปรับ contrast,brightness,saturation
5. ส่วนควบคุมการติดต่อระหว่างส่วนต่าง ๆ จากคอมพิวเตอร์

เมื่อถ่ายภาพสีสัญญาณสีจากกล้องวีดีโอจะเข้ามาทางอินพุทของวงจรไปเข้าส่วนแยกสัญญาณสีออกมาเป็น3สีคือสีแดง(R),เขียว(G),น้ำเงิน(B)แล้วจึงส่งออกไปยังส่วนแปลงสัญญาณ อนาล็อกเป็นดิจิตอล(A/D)เพื่อนำสัญญาณภาพ ไปเก็บในหน่วยความจำ(RAM)

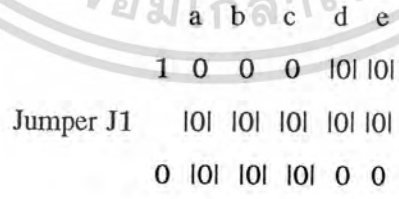
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วน ของวงจรแยกสีนี้เราสามารถควบคุมระดับ ความเข้มสีจากไม่มีสีไปจนถึงมีสีเข้มมากได้ เรียกว่าการปรับ SATURATION ซึ่งควบคุมได้ด้วยซอฟท์แวร์ ส่วนในวงจร A/D เราสามารถควบคุมความชัดเจน ของภาพได้ด้วยการปรับ Reference+(Ref+)และReference-(Ref-) ซึ่งโดยทั่วไป Ref+ จะอยู่สูง กว่าสัญญาณภาพ

ลักษณะของการนำสัญญาณภาพที่ถูกแปลงเป็นดิจิตอลแล้วไปเก็บลงในหน่วยความจำนั้น จะทำการเก็บทีละสีลงในหน่วยความจำสีละ BANK จนครบ 3 BANK ตามลำดับ เมื่อถ่ายภาพขาวดำ(สั่งจากซอฟท์แวร์)สัญญาณภาพจะถูกนำไปเข้าส่วนวงจรแปลงอนาล็อกเป็นดิจิตอลเลย ดังนั้นการเก็บข้อมูลภาพจึงอาจเก็บไว้ในหน่วยความจำเพียง BANK เดียวเท่านั้น ไมโครคอมพิวเตอร์สามารถติดต่อกับหน่วยความจำ (RAM) บน card ได้ โดยการสั่งผ่าน พอร์ทควบคุม ปกติหน่วยความจำภาพนี้จะแยกออกจากระบบของไมโครคอมพิวเตอร์ ในการถ่าย ภาพผู้ใช้สามารถเลือกติดต่อกับหน่วยความจำBANKใดก็ได้ เพื่อทำการเก็บ ข้อมูลภาพที่ถ่ายนั้น ในทำนองเดียวกันผู้ใช้ก็สามารถเลือกติดต่อกับหน่วยความจำ BANK ใดก็ได้ เพื่อทำการอ่านข้อมูล ที่เก็บไว้แล้วมาใช้แต่จะต้องสั่งให้ วงจรหยุดทำการถ่ายภาพเสียก่อน (freeze) จึงจะทำการอ่าน ข้อมูลได้ และเมื่ออ่านข้อมูลจากหน่วยความจำเรียบร้อยแล้ว จึงสามารถสั่งให้หน่วยความจำนั้น กลับ ไปอยู่โหมดการถ่ายภาพ เพื่อถ่ายภาพใหม่ได้อีก

3. การติดตั้งระบบฮาร์ดแวร์

แผงวงจร DigiImage สามารถกำหนดตำแหน่งของ พอร์ทและหน่วยความจำได้โดยผู้ใช้ ด้วยการตั้งตำแหน่งของ jumper J1 ตำแหน่งแอดเดรสที่เลือกได้จะมีดังนี้



แอดเดรสของพอร์ทเลือกที่ jumper a,b แอดเดรสของหน่วยความจำเลือกที่ jumper c,d,e

I/O Address	a	b
210-21Fh	0	0
2A0-2AFh	0	1
310-31Fh	1	0

Memory Address	c	d	e
C400-C3FFh	0	0	0
C800-CBFFh	0	0	1
CC00-CFFFh	0	1	0

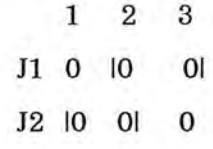
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3A0-3AFh	1	1		D000-D3FFh	0	1	1
				D800-DBFFh	1	0	0
				DC00-DFFFh	1	0	1
				E000-E3FFh	1	1	0
				Not select	1	1	1

โดยปกติแผงวงจร DigiImage จะถูกติดตั้งแอดเดรสของพอร์ทไว้ที่ 210H และแอดเดรสของหน่วยความจำไว้ที่ D000H หากต้องการเลือกแอดเดรสของพอร์ท ไว้ที่ตำแหน่งอื่น ให้เลือกตำแหน่งของ jumper J1 ตรง a,b หากต้องการเลือกแอดเดรสของหน่วยความจำใหม่ก็ให้เลือกตำแหน่งของ jumper J1 ตรง c.d.e ผู้ใช้สามารถทำการติดตั้งระบบฮาร์ดแวร์ด้วยตนเองได้โดยการเสียบแผงวงจร DigiImage ลงในช่องขยาย (Expansiom Slot) ช่องใดช่องหนึ่งของเครื่องไมโครคอมพิวเตอร์ สำหรับเครื่องคอมพิวเตอร์ทั่วไปไม่ว่าจะเป็นระบบ PC,XT,AT หรือ AT386 ที่ประกอบด้วยแผงจอภาพ (ภาษาไทย 25 บรรทัด,Mono,HGC,CGA,EGA, VGA) แผงควบคุมดิสค์ไดรฟ์,ฮาร์ดดิสค์และแผงวงจร Multi I/O (หรือ Multi Function) จะสามารถติดตั้งแผงวงจร DigiImage ลงไปได้ทันที

หากเครื่องคอมพิวเตอร์มีแผงวงจรอื่นนอกจากที่กล่าวมาแล้วติดตั้งอยู่ เช่น แผง วงจร LAN, แผงวงจรสแกนเนอร์,แผงวงจร EMS และวงจรพิเศษอื่นๆรวมทั้งวงจรพิเศษที่มีการติดตั้งอยู่บนเมนบอร์ด เช่นวงจร EMS (สำหรับเครื่อง AT บางรุ่น) หากแอดเดรสของพอร์ทหรือหน่วยความจำของวงจรพิเศษเหล่านี้ ถูกเลือกไว้ตรงกับของแผงวงจร DigiImage ก็จะต้องมีการย้ายตำแหน่งแอดเดรสของแผงวงจรอันใดอันหนึ่งใหม่ เพื่อไม่ให้ตำแหน่ง Address ของแผงวงจร DigiImage ไปชนกับ address ของ แผงวงจรอื่น

ส่วนตำแหน่ง jumper ที่ J2 และ J3 ใช้สำหรับการเลือกสัญญาณจาก input เพื่อให้เหมาะสมกับ แผงวงจร DigiImage ความหมายและตำแหน่งที่เลือกมีดังนี้



J2 เชื่อมระหว่าง

1,2 = เลือกใช้กับกล้องที่ให้สัญญาณออกมาเป็น ขาว/ดำ

2,3 = เลือกใช้กับกล้องที่ให้สัญญาณออกมาเป็น สี

J3 เชื่อมระหว่าง

1,2 = สำหรับสัญญาณวีดีโอที่ต้องการ INPUT IMPEDANCE สูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2,3 = สำหรับสัญญาณวีดีโอที่ต้องการ INPUT IMPEDANCE 75 โอห์ม

โดยปกติตำแหน่งของ jumper J2 จะตั้งไว้ 2,3 และ J3 จะตั้งไว้ที่ 1,2 เมื่อทำการติดตั้งแผงวงจร DigiImage ลงไปในเครื่องเรียบร้อยแล้ว นำสัญญาณ VIDEO ป้อนเข้าที่ แจ็ค VDO-IN จากนั้นให้เรียกโปรแกรม DZ2DI.COM เลือก menu ไปที่ Digitize now เพื่อนำสัญญาณภาพ video ไปแสดงบนจอ VGA ตามขั้นตอนต่อไป

4. การติดต่อกับ card DigiImage

บนแผงวงจร DigiImage มีพอร์ท Output Port อยู่ 4 Port , Input Port อยู่ 2 Port ซึ่งมีหน้าที่ต่าง ๆดังนี้

Output Port 4 Port มีดังนี้

CtrlPort:เป็นพอร์ทคำสั่งในการถ่ายภาพ รวมทั้งการเลือก BANK,PAGE ของหน่วยความจำ และสีที่ ต้องการถ่าย ตำแหน่งของพอร์ตนี้ อยู่ที่ 210h,2a0h,310h,3a0h ตามการตั้งตำแหน่งของ jumper ความหมายของพอร์ทนี้ มีดังนี้

d7 เป็นคำสั่งในการถ่ายภาพหรือหยุดภาพ

0 = สั่งให้หยุด digitize

1 = สั่งให้ digitize

d6 สัญญาณนี้ต่อออกไปที่ช่อง J-out สามารถนำไปควบคุมอะไรก็ได้ ระดับของสัญญาณ จะเหมือนกับสัญญาณที่ออกมาจาก IC TTL

d5,d4 เลือกให้สัญญาณที่ต้องการถ่าย ไปเข้าวงจร A/D

00 = เลือกถ่ายสี Y (สัญญาณขาวดำ)

01 = เลือกถ่ายสี R (แดง)

10 = เลือกถ่ายสี G (เขียว)

11 = เลือกถ่ายสี B (น้ำเงิน)

d3,d2 เลือกหน่วยความจำ BANK ที่ต้องการเก็บภาพที่ถูกแปลงเป็นดิจิตอล หรือ เลือกหน่วย ความจำ BANK ที่ต้องการอ่านหรือเขียนข้อมูล ซึ่งมี 3 BANK

00 = เลือก RAM BANK 0

01 = เลือก RAM BANK 1

10 = เลือก RAM BANK 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

d1,d0 เลือก PAGE ของหน่วยความจำที่ต้องการอ่านหรือเขียน โดยในหน่วยความจำแต่ละ BANK จะติดต่อกับ CPU เป็น PAGE ซึ่งมี BANK ละ 4 PAGE แต่ละ PAGE มีหน่วย ความจำอยู่ 16K byte

00 = เลือก RAM PAGE 0

01 = เลือก RAM PAGE 1

10 = เลือก RAM PAGE 2

11 = เลือก RAM PAGE 3

RefhPort:มีตำแหน่งอยู่ที่ CtrlPort+1 เป็นค่าของ reference+ สำหรับควบคุมการทำงานของวงจร แปลงอนาล็อกเป็นดิจิตอล

d5-d0 = ค่าตั้งแต่ 0-63

(ReflPort:มีตำแหน่งอยู่ที่ CtrlPort+2 เป็นค่าของ reference- สำหรับควบคุมการทำงานของวงจร แปลงอนาล็อกเป็นดิจิตอล

d5-d0 = ค่าตั้งแต่ 0-63

SatPort:มีตำแหน่งอยู่ที่ CtrlPort+3 เป็นค่าของ Saturation สำหรับควบคุมการทำงานของวงจรแยกสี

d5-d0 = ค่าตั้งแต่ 0-63

Input Port 2 Port มีดังนี้

StatPort:มีตำแหน่งอยู่ที่ CtrlPort เป็นพอร์ทอ่านศัญญาณที่ใช้ในการตรวจสอบเมื่อจะถ่ายภาพ รวมทั้งอ่านข้อมูลของ CtrlPort บางอย่างกลับมาด้วย

d7 อ่านมาจาก D7 ของ CtrlPort

d6ตรวจสอบสัญญาณจากช่อง J-IN เพื่อนำไปใช้ในโปรแกรมต่าง ๆผู้ใช้สามารถ ต่อไปใช้งานได้ สัญญาณที่ต้องการจะต้องมีระดับแรงดันเท่ากับอินพุทของ IC TTL ต้องการ ปกติจะถูกดึงขึ้นไฟบวก 5 Volt ด้วย ค่าความต้านทาน 4.7K ohm

d5 ตรวจสอบสัญญาณการถ่ายภาพ digi_stat (เป็น PULSE)

d4 ตรวจสอบสัญญาณ Hor. sync (เป็น PULSE)

d3-d0 อ่านมาจาก d3-d0 ของ CtrlPort

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
033169

SWPort:มีตำแหน่งของพอร์ทอยู่ที่ CtrlPort+1 เป็นพอร์ทที่ใช้อ่านตำแหน่งของ jumper J1 ตัว c,d,e ที่ได้ตั้งไว้พอร์ทนี้มีคุณสมบัติพิเศษคือ จะต้องอ่านอย่างน้อย 2 ครั้ง โดยครั้งแรกจะได้ ข้อมูลที่ผิดแต่ครั้งต่อ ๆไปจะถูก

d3,d2,d1 อ่านจากตำแหน่งของ jumper J1 ตรง c,d,e ตามลำดับ

d0 มีค่าเป็น 0 (สำหรับตรวจสอบว่าเป็นแผงวงจรสี)

ตัวอย่าง การติดต่อกับ DigiImage

สั่งให้ DigiImage ถ่ายภาพ (digitize)

```
out CtrlPort+2,(63-brightness) ; Ref-
out CtrlPort+1,(reference) + ((63-reference-)*(63-contrast))/63 ; Ref+
out CtrlPort+3,Saturation
out CtrlPort,80h    ; ถ่ายสี Y ลง RAM BANK 0
out CtrlPort,90h    ; ถ่ายสี R ลง RAM BANK 0
out CtrlPort,0a4h   ; ถ่ายสี G ลง RAM BANK 1
out CtrlPort,0b8h   ; ถ่ายสี B ลง RAM BANK 2
out CtrlPort,0      ; หยุดการถ่าย (freeze)
```

สั่งให้ติดหน่วยความจำ BANK 0 ของ DigiImage

```
out CtrlPort,00     ; PAGE 0
out CtrlPort,01     ; PAGE 1
out CtrlPort,02     ; PAGE 2
out CtrlPort,03     ; PAGE 3
```

สั่งให้ติดหน่วยความจำ BANK 1 ของ

```
out CtrlPort,04     ; PAGE 4
out CtrlPort,05     ; PAGE 5
out CtrlPort,06     ; PAGE 6
out CtrlPort,07     ; PAGE 7
```

สั่งให้ติดหน่วยความจำ BANK 2 ของ DigiImage

```
out CtrlPort,08     ; PAGE 8
out CtrlPort,09     ; PAGE 9
out CtrlPort,0ah    ; PAGE 10
out CtrlPort,0bh    ; PAGE 11
```

5. ขั้นตอนการควบคุม DigiImage

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ก่อนที่เราจะสั่งถ่ายภาพ เราจะต้องส่งคำสั่งออกไปทาง RefhPort, ReflPort และ SatPort ก่อน เพื่อกำหนดสภาวะของการทำงานของวงจรถ่ายภาพโดย RefhPort และ ReflPort คือคำนวณมาจาก Contrast และ Brightness ที่กำหนดไว้ก่อน ส่วน SatPort คือค่า Saturation ของภาพ ปกติควรให้ค่าของ Contrast = 32, Brightness = 50 และ Saturation = 32
2. ตรวจสัญญาณ Hor-Sync (ที่บิท d4 ของ StatPort) รอจนกว่าจะเป็นของขาขึ้น (จาก 0 ไป 1) สั่ง digitize โดยการส่งคำสั่งออก CtrlPort ซึ่งประกอบด้วยคำสั่งในการถ่าย,เลือกสัญญาณที่ต้องการ ถ่ายไปเข้าวงจรอนาล็อกเป็นดิจิตอล และ เลือกหน่วยความจำ BANK ที่ต้องการ
3. ตรวจสัญญาณ digi_stat (ที่บิท d5 ของ SatPort) จะต้องมี PULSE (1) ภายในไม่เกิน 20ms แล้วรอสัญญาณนี้จนตกเป็น 0 หมายถึงถ่ายภาพครบ 1 frame
4. หากต้องการส่งให้หยุด digitize (freeze) ให้ส่งคำสั่งหยุดภาพออก CtrlPort (บิท d7 เป็น 0)
5. สำหรับการถ่ายภาพในโหมด color RGB จะต้องสั่ง digitize ภาพสี R,G,B ลงหน่วยความจำ BANK 0,1,2 ตามลำดับ ดังนั้นจึงต้องสั่ง digitize ทั้งหมด 3 ครั้ง และในแต่ละครั้งจะต้องรอ สัญญาณ digi_stat ตกจาก PULSE 1 เป็น 0 เสมอ
6. เมื่อต้องการอ่านข้อมูลที่อยู่บนหน่วยความจำของการ์ดDigiImage มาลงยังหน่วยความจำของ คอมพิวเตอร์ เราสามารถอ่านได้ครั้งละ 16K byte หรือเท่ากับ 1 PAGE ฉะนั้นการอ่านภาพสี 64 K byte 3 BANK จะต้องอ่านถึง12เที่ยว ซึ่งในการอ่านแต่ละเที่ยวจะต้องส่งคำสั่งออก CtrlPort เพื่อทำการเลือก BANK และ PAGE ที่ต้องการอ่าน

ขั้นตอนต่าง ๆ สามารถเขียนเป็นกระบวนความได้ดังนี้

1.การตรวจสัญญาณของคาร์ด จะสแกนหา Address ของ CtrlPort เป็นการตรวจหาพอร์ททั้ง 4 ตำแหน่ง คือ 210h,2a0h,310h และ 3a0h โดยการตรวจดูสัญญาณที่บ่งบอกว่าเป็นคาร์ด DigiImage สัญญาณที่ตรวจจะอยู่ ที่พอร์ทห่างออกไป 8001h เช่น 210h จะตรวจที่ 8211h เป็นต้น

```
PROCEDURE Chk_ID_Card
    PortBase [ARRAY 1 TO 4] = 210h,2a0h,310h,3a0h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Index = 1
  FOR Count = 1 TO 4 DO
      CtrlPort = PortBase[Index]
      SWPort  = CtrlPort + 1
      ChkPort  = CtrlPort OR 8001h
      IN PORT SWPort TO Chk1 (Skip first byte)
      IN PORT SWPort TO Chk1
      Chk1   = ((Chk1 XOR 0fh) AND 0fh)
      Chk2   = Chk1

      FOR Time = 0 TO 200 DO
       IN PORT ChkPort TO Chk1
       Chk1  = ((chk1 XOR 0fh)  AND 0fh)
       IF ( Chk1 <> Chk2 )
         GOTO NextBase
       END IF
       Chk2  = ((Chk2 XOR 0fh) AND 0fh)
      END FOR
      EXIT (with CtrlPort)

      NextBase:
      Index  =  Index + 1
  END FOR
 PRINT "Card DigiImage Not Found 1 1"
END PROCEDURE
```

2. การตรวจสัญญาณ Digi_Stat (ที่บิท d5 ของ StatPort) จะต้องตรวจหลังจากส่งคำสั่งถ่ายภาพออกไปแล้ว ซึ่งเมื่อออกจาก PROCEDURE นี้แล้วจะหมายถึงถ่ายภาพได้ครบ 1 เฟรม แต่ถ้าสัญญาณ Digi_Stat ไม่เคยเป็น 1 เลย ภายในช่วงเวลาที่กำหนด แสดงว่าไม่มีสัญญาณ Video ส่งมาที่คาร์ด

```
PROCEDURE Chk_DIgi_Stat
   FOR Pulse = 1 DOWN TO 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Time = 0
    Time_Out = 4000
        WHILE digi_stat <> Pulse
        Time = Time + 1
          IF(Time>Time_Out)
            PRINT "VIDEO Signal Not Found."
            EXIT
          END IF
        END WHILE
    END FOR
END PROCEDURE
```

3. การสั่งถ่ายภาพเราสามารถเลือกถ่ายได้ทั้ง โหมดสีและขาว-ดำ สำหรับ PROCEDURE Digitize_RGB จะเป็นการสั่งถ่ายภาพสี R,G,B ขนาด 256*256 จุด ลงหน่วยความจำ BANK 0,1,2 ตามลำดับ และ PROCEDURE Digitize_B&W เป็นการสั่งถ่ายภาพขาว-ดำ ขนาด 256*256 จุด ลงหน่วยความจำ BANK0 เพียง BANK เดียว จะเห็นว่าขั้นตอนในการ สั่งถ่ายแต่ละครั้ง จะต้องทำการตรวจสัญญาณ Digi_Stat เสมอ เพื่อเป็นการบอกให้รู้ว่า ภาพที่กำลังถ่ายอยู่นั้นถ่ายครบเฟรมหรือยัง เมื่อกระบวนการถ่ายเสร็จสิ้นไปแล้ว เราจึงสั่งให้หยุดถ่ายเป็นการถ่ายครบ 1 ภาพ และเมื่อต้องการถ่ายภาพต่อไปก็ให้ทำตามขั้นตอนของกระบวนความนี้

การสั่งถ่ายภาพสี

```
PROCEDURE Digitize_RGB
    DigiCom_R = 90h (10010000 b)
    DigiCom_G = A4h (10100100 b)
    DigiCom_B = B8h (10111000 b)
    DigiComStop = 0   (00000000 b)
          OUT CtrlPort,DigiCom_R
             CALL Chk_Digi_Stat
          OUT CtrlPort,DigiCom_G
             CALL Chk_Digi_Stat
          OUT CtrlPort,DigiCom_B
             CALL Chk_Digi_Stat
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        OUT CtrlPort,DigiComStop
END PROCEDURE
```

การสั่งถ่ายภาพขาวดำ

```
PROCEDURE Digitize_B&W
    DigiCom_Y = 80h (10000000 b)
    DigiComStop = 0 (00000000 b)
        OUT CtrlPort,DigiCom_y
            CALL Chk_Digi_Stat
        OUT CtrlPort,DigiComStop
END PROCEDURE
```

4. การปรับค่าของ Contrast,Brightness และ Saturation สามารถเปลี่ยนค่านี้ได้ตั้งแต่ 0-63 ระดับ และค่าที่ได้จากการเปลี่ยนแปลงเราจะนำมาโปรแกรมตามขั้นตอนของกระบวน ความนี้

```
PROCEDURE Con_Bri_Sat_Level
    Contrast    = 32 (Vary from 0 to 63 )
    Brightness  = 50 (Vary from 0 to 63 )
    Saturation  = 32(Vary from 0 to 63 )
    Reference_  = 63-Brightness
    RefhPort    = CtrlPort+1
    ReflPort    = CtrlPort+2
    SatPort     = CtrlPort+3
        OUT ReflPort,Reference_
        OUT ReflPort,Reference_+[[63-Reference-]*[63-Contrast]]/63
        OUT SatPort,Saturation
END PROCEDURE
```

5.การอ่านตำแหน่ง Address ของหน่วยความนำบน Card DigiImage ทำได้โดยการอ่านค่าของ J1 จาก SWPort แล้วจึงเปิดตารางหาตำแหน่งของAddress

```
PROCEDURE   Read_Address
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SWPort  = CtrlPort + 1
IN SWPort TO ChkSW (LATCH DATA)
IN SWPort TO ChkSW (GET DATA)
ChkSW  = ChkSW  AND 0eh (00001110 b)
    CASE ChkSW DO
        0 : CardSeg  =  C400h
        2 : CardSeg  =  C800h
        4 : CardSeg  =  CC00h
        6 : CardSeg  =  D000h
        8 : CardSeg  =  D800h
        10 : CardSeg  =  DC00h
       12 : CardSeg  =  E000h
       14 : CardSeg  =  NULL
    END CASE
END PROCEDURE
```

6. การติดต่อกับหน่วยความจำของ DigiImageเราจะแยกเป็น PROCEDURE Read_RGB สำหรับอ่านข้อมูลจากหน่วยความจำ BANK 0,1,2 ของ Card DigiIMage ซึ่งเป็นหน่วยความจำที่เก็บข้อมูลของภาพสี R,G,B ตามลำดับ ลงหน่วยความจำ ของระบบที่จองไว้ 64 K*3 อันละ PROCEDURE Read_Y จะเป็นการอ่านดาต้าจากหน่วยความจำ BANK 0 ที่เก็บ ข้อมูลของภาพขาว-ดำไว้เพียง BANK เดียว เพื่อนำลงหน่วยความจำของระบบที่จองไว้ 64 K byte ส่วน PROCEDURE Read_RAM_Bank จะเป็นโปรแกรมย่อยเพื่อให้เรียกใช้โดย จะต้องส่งพารามิเตอร์ RamBANK และ BufSeg ไปให้

การติดต่อกับหน่วยความจำสำหรับภาพสี

```
PROCEDURE Read_RGB
    Bank0_R  = 10h (00010000 b)
    Bank1_G  = 24h (00100100 b)
    Bank2_G  = 38h (00111000 b)
    BufSeg_R = 64 Kbyte ALLOCATE MEMORY
    BufSeg_G = 64 Kbyte ALLOCATE MEMORY
    BufSeg_B = 64 Kbyte ALLOCATE MEMORY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CALL Read_RAM_Bank WITH Bank0_R,BufSeg_R
        CALL Read_RAM_Bank WITH Bank1_G,BufSeg_G
        CALL Read_RAM_Bank WITH Bank2_B,BufSeg_B
END PROCEDURE
```

การติดต่อกับหน่วยความจำสำหรับภาพขาวดำ

```
PROCEDURE Read_Y
    Bank0_Y  = 0 (00000000 b)
    BufSeg_Y = 64 Kbyte ALLOCATE MEMORYIS
      CALL Read_RAM_Bank WITH Bank0_Y,BufSeg_Y
END PROCEDURE
```

โปรแกรมย่อยสำหรับอ่านหน่วยความจำจากคาร์ดลงหน่วยความจำของระบบ

```
PROCEDURE Read_RAM_Bank <RamBank,BufSeg>
(CardSeg from PROCEDURE Read_Address)
Card0ffs  =  0
Buf0ffs   =  0
    FOR Count  = 1 TO 4
      OUT CtrlPort,RamBank
      Move DATA FROM CaedSeg:Card0ffs TO BufSeg:Buf0ffs LENGTH 16K
byte
      RamBank  = RamBank + 1 (to next page)
      Buf0ffs     = Buf0ffs + 16K
    END FOR
END PROCEDURE
```

6. ลักษณะของข้อมูลภาพ

ภาพขาวดำที่เก็บอยู่ในหน่วยความจำนั้นจะมี ขนาด 256*256 จุด โดยใช้จำนวนบิททั้งหมด 6 บิทบนคือตั้งแต่บิท 2 ถึง บิท 7 แบ่งระดับเทาเป็น 0-63 ระดับ และแต่ละจุด จะต้องใช้เนื้อที่ขนาด 1 ไบต์เรียงกันไปในหน่วยความจำขนาด 64K ไบต์ ของ BANK 0 โดยข้อมูลภาพจุดแรกจะอยู่ใน หน่วยความจำแอดเดรสแรกคือ 0 ข้อมูลภาพจุดถัดมา (ตามแนวนอน) ก็จะอยู่ที่แอดเดรส 1 เรียงกันไปจนจบ ข้อมูลภาพจะมีแอดเดรสเป็น 65535 หรือ 0FFFFH สำหรับภาพสี จะใช้หน่วยความจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3 BANK แบ่งเป็นสี R,G,B สีละ BANK แต่ละ BANK สามารถบรรจุได้ทั้งหมด 256*256 จุด แต่ละจุดจะมีข้อมูลภาพ อยู่ 6 บิทบน คือตั้งแต่บิท 2 ถึง บิท 7 ดังนั้นบิท 0 กับ บิท 1 จะเป็นข้อมูลที่ไม่ได้ใช้ และการเก็บข้อมูลในแต่ละ BANK จะเหมือนกับการเก็บข้อมูล ของภาพขาวดำ

โดยปกติข้อมูลบิท 0 และ 1 ของทุก ๆไบต์ที่เก็บอยู่ในหน่วยความจำจะเป็นตัวบอกว่าข้อมูลนี้เป็นสีอะไร คือ

D1	D0		
0	0	=	ข้อมูลจากสัญญาณ Y
0	1	=	ข้อมูลจากสัญญาณ R
1	0	=	ข้อมูลจากสัญญาณ G
1	1	=	ข้อมูลจากสัญญาณ B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ส่วนแสดงผลภาพ

ส่วนนี้เป็นส่วนของการแสดงสถานะและผลลัพธ์ต่าง ๆของระบบโดยจะแสดงผลออกทางจอภาพ ซึ่งสิ่งที่แสดงก็จะประกอบไปด้วยข้อมูลภาพที่รับเข้ามา ผลของการประมวลผลภาพ สถานะการทำงาน ของตัวหุ่นยนต์(เช่น ความเร็ว และ ทิศทางการเคลื่อนที่) และข้อมูลต่าง ๆ จากเซนเซอร์ที่ติดกับตัวหุ่นยนต์ นอกจากนี้ ยังทำหน้าที่รายงานความผิดพลาดต่าง ๆ แก่ผู้ควบคุม ในกรณีที่มีความผิดพลาดเกิดขึ้น ทั้งความ ผิดพลาดจากการ ทำงานของตัวเครื่อง และจากตัวหุ่นยนต์ โดยในการแสดงผลของจอภาพนั้น เนื่องจากเราต้อง การแสดงทั้งส่วนของภาพและข้อความ เราจึงต้องทำการแสดงผลในรูปแบบกราฟฟิก ซึ่งเราได้เลือกใช้การแสดงผลบนจอวีจีเอ(VGA) ซึ่งในการแสดงผลภาพนั้นเพื่อให้สามารถแสดงภาพได้เหมือนจริงที่สุด เราจึง เลือกแสดงภาพในระดับสี 256 สี ซึ่งเป็นระดับสีสูงสุดซึ่งสามารถใช้งานได้บนการ์ดวีจีเอ แต่เนื่อง จากต้องการ แสดงผลด้วยความรวดเร็วและสีของภาพ ไม่จำเป็นต่อการใช้งานมากนัก จึงทำการ แสดงผลด้วยระบบสีแบบ 64 ระดับเทา(gray level) ส่วนความละเอียดของการแสดงผล ในโหมดนี้นั้น(โหมด 13h) จะมีความละเอียด ของจอภาพเพียง 320x200 จุด ซึ่งภาพที่เรารับ มาได้นั้นเป็นขนาด 256x256 จุด จึงทำการลดขนาดของภาพที่จะแสดงเป็นภาพขนาด 128x128 จุด 64 ระดับเทา เพื่อให้สามารถแสดงผลได้เป็นภาพ และมีพื้นที่จอภาพเหลือเพียงพอที่จะใช้ในการแสดงผลสำหรับข้อมูลอื่น ๆ พร้อม ๆกันไป

2.2.1 การติดต่อกับระบบฮาร์ดแวร์ของการ์ดวีจีเอ (VGA Card Interfacing)

ในโครงงานนี้นั้นเลือกใช้การจัดการทางกราฟิกในโหมด 13 ฐานสิบหกของการ์ดวีจีเอ ซึ่งสามารถแสดงได้ 256สี โดยใช้หน่วยความจำขนาด 256กิโลไบท์ ที่แบ่งออกเป็น 4 ระนาบบิท (bit-plane) ซึ่งต่อกันอย่างลูกโซ่ (chained memory) หลักการทำงานในโหมดนี้จะแตกต่างจากโหมดอื่นในเรื่องของการอ่าน-เขียนหน่วยความจำภาพโดยตรง (direct video memory) เพราะในโหมดอื่นจะเป็นการจัดการโครงสร้างหน่วยความจำแบบระนาบ (planar configuration) ส่วนในโหมด 13h จะเป็นการจัดการโครงสร้างหน่วยความจำแบบลูกโซ่ต่อกัน (chain configuration) และยังแตกต่างจากโหมดอื่นในเรื่องของการติดต่อกับชุดของรีจิสเตอร์สี (color registers) เพราะ โหมด 13h แสดงได้ถึง 256สี ในขณะที่โหมดอื่นจะแสดงได้ 2 ระดับ โมโนโครม, 2สี, 4สี หรือ 16สี แต่ในด้านความละเอียดของภาพแล้วโหมด 13 ฐานสิบหก จะไม่ละเอียดนัก (มีความละเอียด 320x200) เมื่อเทียบกับโหมดอื่น

รายละเอียดที่ควรรู้ก่อนการโปรแกรมเพื่อใช้งานทางกราฟิกของการ์ดวีจีเอใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมด 13 ฐานสิบหก ตามลำดับ ได้แก่

-กราฟิกแอทริบิวท์ (Graphics Attributes)

-การจัดการหน่วยความจำภาพ (Display Memory)

-โหมดการแสดงผลทางกราฟิก (Graphics Display Modes)

- กราฟิกแอทริบิวท์ (Graphics Attributes)

ในโหมดกราฟิกแอทริบิวท์ของตัวอักษร (character attribute) จะถูกจัดการต่างไปจากในโหมดตัวเลขตัวอักษร (alphanumeric mode) ในโหมดกราฟิกนั้น รูปแบบของจุดที่ประกอบ เป็นตัวอักษรจะถูกเก็บลงในหน่วยความจำภาพเลย โดยจัดการแอทริบิวท์ตามโหมดกราฟิกที่ใช้แสดงอยู่ โดยจุดต่าง ๆที่เป็นฉากหน้า (foreground) ของตัวอักษรจะถูกทำการโหลดลงเท่านั้น ซึ่งค่าในไบท์ แอทริบิวท์ (attribute byte) ส่วนฉากหลัง (background)จะไม่ถูกสนใจ

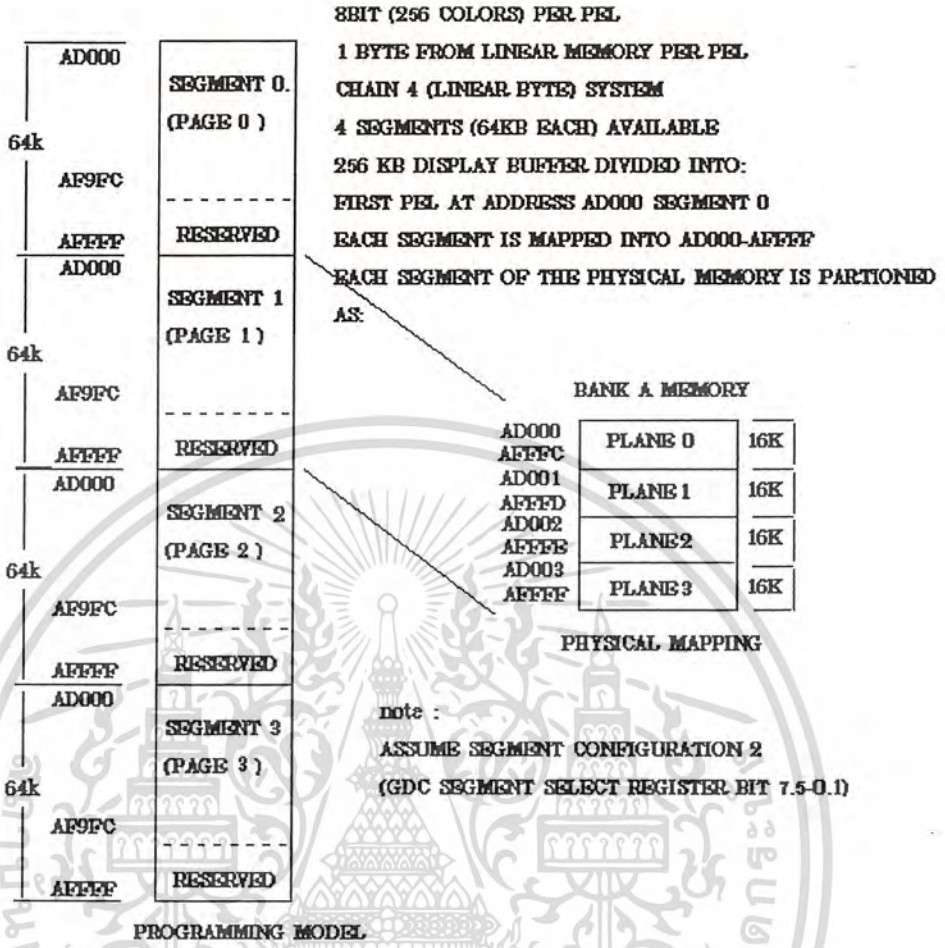
ในโหมด 13 ฐานสิบหก ค่าในรีจิสเตอร์พาเลทท์ (register pallete) จะใช้ในการ กำหนดตำแหน่งแอดเดรสของรีจิสเตอร์สี (color register) ที่เก็บสีที่ต้องการแสดงผลซึ่งในโหมดนี้เท่านั้น ถ้าต้องการเปลี่ยนสีในชุดสีที่มีจะต้องตั้งค่าในรีจิสเตอร์สีใหม่ตามความต้องการ

- การจัดการหน่วยความจำในการแสดงภาพ (Video Memory Organifzation)

หน่วยความจำในการแสดงภาพจะประกอบด้วยหน่วยความจำขนาด 256 กิโลไบท์ (แบงก์ A) และที่เพิ่มเติมมาอีก 256 กิโลไบท์ (แบงก์ B) รวมเป็น 512 กิโลไบท์ หน่วยความจำส่วนนี้จะแมพได้เพียง 128 กิโลไบท์ในระบบของ IBM โดยโปรแกรม GDC Miscellaneous Register (GDC Index Register 6) ดังนี้

GDC Misc. Reg<3,2>	Address space	Sie	Mode
00	A0000-BFFFF	128KB	Hi-resolution
01	A0000-B0000	64KB	
10	B0000-B8000	32KB	
11	B8000-BFFFF	32KB	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2.1 โครงสร้างหน่วยความจำแบบเชิงเส้นในโหมด 13 H

- โหมด 256 สี (VGA)

ทุก ๆจุดถูกทำการแทนโดยข้อมูลขนาด 8 บิท ซึ่งแบ่งเป็น 2 ส่วน ๆละ 4 บิท ส่วน แรก คือ 4 บิทล่าง (บิท 0-3) แอดเดรสรีจิสเตอร์พาเลท , เอาท์พุท 4 บิทล่างของรีจิสเตอร์พาเลท จะใช้ในการแอดเดรส (บิท 0-3) ของรีจิสเตอร์สี ส่วนที่สอง คือ 4 บิทบน (บิท 4-7) รีจิสเตอร์สี ในรีจิสเตอร์สี จะเก็บรหัสของสีที่เอาท์พุทผ่านตัวแปลง สัญญาณจากดิจิตอลไปอนาลอก (DAC)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากหลักการการแอดเดรสรีจิสเตอร์สี (Color Register) ในวีจีเอจะทำได้ใน 2 ลักษณะ คือ

1. แบบที่ใช้เอาท์พุทจาก รีจิสเตอร์พาเลท 6 บิท (บิท0-5) ร่วมกับอีก 2 บิท จาก Color Select Register (แอดเดรส 3C0 ฐานสิบหก อินเดกซ์ 14 ฐานสิบ หก, C7/C6) และจะทำงานในลักษณะนี้โดย เซทบิท PSS (Palette Size Select) ใน Mode Control Register (แอดเดรส 3C0 ฐานสิบหกอินเดกซ์ 10 ฐานสิบหก) เป็น 1 เสมือนกับเป็นการแบ่งออกเป็น 4 แบงค์ๆละรีจิสเตอร์สี 64 ตัว

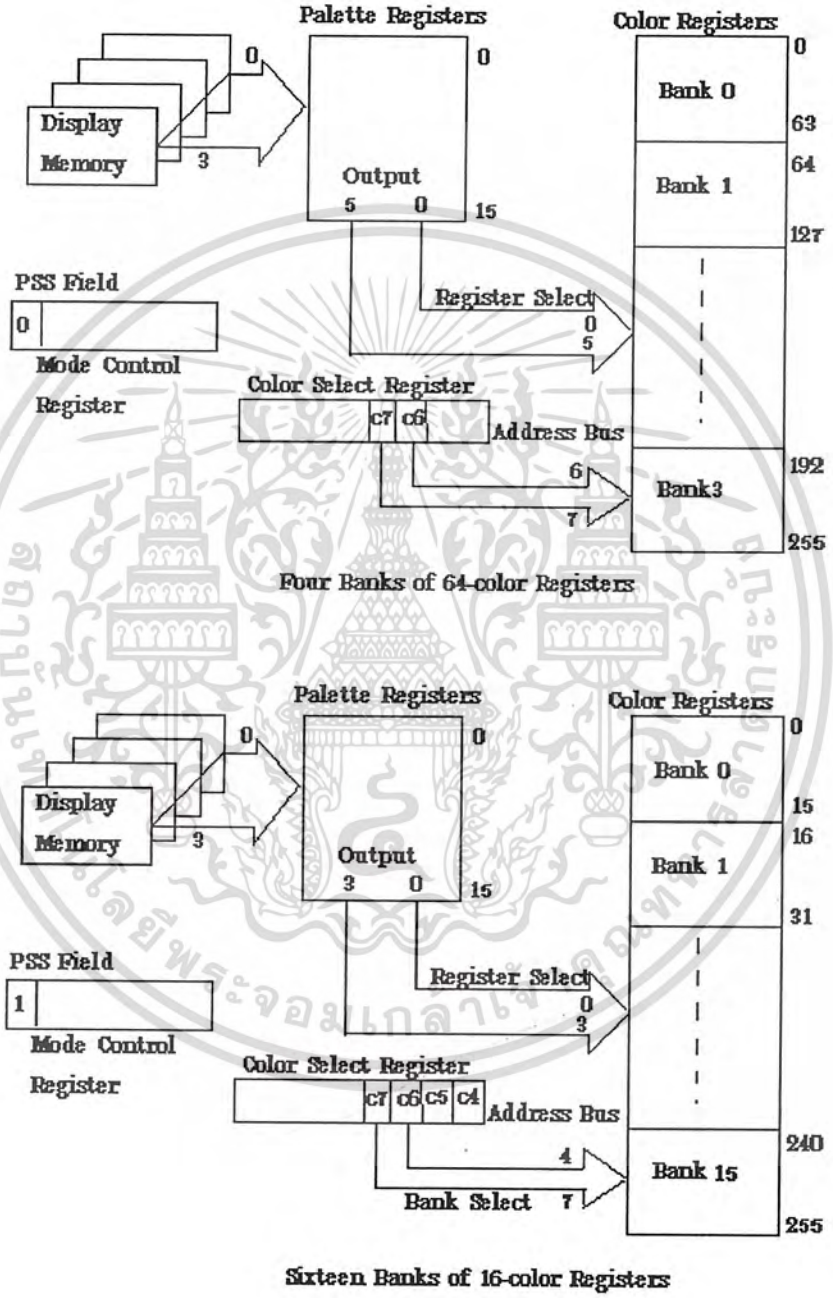
(หมายเหตุ ในวีจีเอโหมด 13 ฐานสิบหก บิท 6 ของ Mode Control Register จะถูกเซทเป็น 1)

2. แบบที่ใช้เอาท์พุทจาก palette registers 4 บิท(บิท 0-3) ร่วมกับอีก 4 บิทจาก Color Select Register (แอดเดรส 3C0 ฐานสิบหก ,C7/C6/C5/C4) และจะทำงานในลักษณะนี้โดยเซทบิท PSS ใน Mode Control Register (แอดเดรส 3C0 ฐานสิบหก) เป็น 0 เสมือนกับเป็นการแบ่งออกเป็น 16 แบงค์(bank) ๆละ 16 รีจิสเตอร์สี

การทำการแก้ไขรีจิสเตอร์สีโดยตรงโดยอาศัยชุดของรีจิสเตอร์ Video Digital-to-Analog Converter 4 ตัว ดังนี้

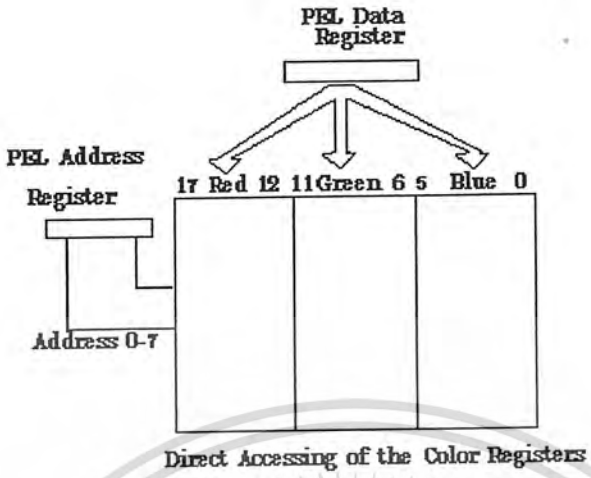
แอดเดรส	อ่าน/เขียน	หน้าที่
1. 3C8 ฐานสิบหก	อ่าน/เขียน	กำหนดแอดเดรสของ PEL ระหว่างเขียน
2. 3C7 ฐานสิบหก	เขียนเท่านั้น	กำหนดแอดเดรสของ PEL ระหว่างอ่าน
3. 3C7 ฐานสิบหก	อ่านเท่านั้น	บอกสถานะของ DAC(DAC state)
4. 3C9 ฐานสิบหก	อ่าน/เขียน	รีจิสเตอร์ข้อมูลของ PEL (PEL Data Register)
5. 3C6 ฐานสิบหก	อ่าน/เขียน	ใช้สำหรับมาล์กรีจิสเตอร์ PEL(PEL mask)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2.2 การจัดแบ่งชุดของรีจิสเตอร์สี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2.3 การตั้งค่าชุดของรีจิสเตอร์สี

PEL Address Register

มีขนาด 8 บิทและใช้ในการแอดเดรสรีจิสเตอร์ตัวใดตัวหนึ่งในรีจิสเตอร์สี 256 ตัว รีจิสเตอร์แอดเดรสของ PEL จะมีอยู่ 2 แอดเดรสในระบบของพอร์ทรับ-ส่งข้อมูลของคอมพิวเตอร์แม่ข่าย (host port I/O space) ซึ่งจะใช้แอดเดรสรีจิสเตอร์สีชุดที่จะให้ค่าหรืออ่านค่า โดยการเขียนพอร์ทแอดเดรส 3C8 ฐานสิบหก ใช้บอกว่าต้องการทำการเขียนรีจิสเตอร์สีตัวที่กำหนด

แอเดรสไว้,ส่วนการเขียนพอร์ทแอดเดรส 3C7 ฐานสิบหก ใช้บอกว่าต้องการทำการอ่านรีจิสเตอร์สีตัวที่กำหนดแอิดเดรสไว้

PEL Data Register

จะเก็บฟิลด์ขนาด 6 บิทที่จะถูกใช้โดยคอมพิวเตอร์ตัวแม่ในการอ่านข้อมูลจากหรือเขียนข้อมูลลงรีจิสเตอร์สี โดยต้องทำการอ่าน/เขียน PEL Data Registers 3 ครั้งติดต่อกันจึงจะครบ 1 ชุดของรีจิสเจอร์สีคือ แดง (Red), เขียว (green),และน้ำเงิน (blue) ทีละตัวในแต่ละชุดแล้วแอดเดรสจะเพิ่มโดยอัตโนมัติ ซึ่งจะเริ่มการอ่าน/ เขียนที่ฟิลด์ของสีแดง ในรีจิสเจอร์สีแล้วตามด้วยเขียวและน้ำเงินตามลำดับการเพิ่มแอดเดรสโดยอัตโนมัติ เพื่อที่ สามารถจะ ติดต่อกับรีจิสเตอร์สีต่างๆ โดยไม่ต้องทำการตั้งค่าแอดเดรสใน PEL Address Register ทุกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องมาจากข้อจำกัดของฮาร์ดแวร์ทำให้มีดีเลย์เล็กน้อยระหว่างที่ได้ทำการอ่าน /เขียนกับ PEL Data Register เสร็จแล้ว

ขณะทำการแก้ไขรีจิสเตอร์สีต่าง ๆอาจจะเหิดสโนว์ (snow) ได้ เพื่อที่จะแก้ปัญหาการเกิดสโนว์ควรจะไม่ทำการแก้ไขในระหว่างช่วงแอดทีฟรีเฟรช (active refresh period) ซึ่งมีทางเป็นไปได้สองทาง คือ ทำการแก้ไขเฉพาะในช่วงรีเทรช (retrace period) เท่านั้น หรืออาจจะทำการหยุดการรีเฟรชไว้โดยการใช้ฟิลด์ Screen Off ใน Clocking Mode Register DAC State Register

สภาพของการติดต่อโดยคอมพิวเตอร์ตัวแม่กับรีจิสเตอร์สีสามารถตรวจดูได้จาก DAC State Register ที่ต้องมีไว้ตรวจสอบได้ถูกเขียนลง PEL Address Register อาจ เกิดการขัดข้องของข้อมูลขึ้นได้ ถ้ามีการตั้งค่าของ PEL Address Register ก่อนจะเสร็จสิ้นลำดับของการอ่าน/เขียนกับ PEL Data Register บิทที่บอกสภาพอยู่ที่บิท 0 และ 1 ของ DAC State Register โดยจะบอกสถานภาพ ดังนี้

บิทที่ 0,1	ความหมาย
00	ขณะนี้ DAC อยู่ในโหมดของการอ่าน
11	ขณะนี้ DAC อยู่ในโหมดของการเขียน

ชุดคำสั่งเบื้องต้นที่ใช้ในการแสดงผลในการแสดงผลในโหมดกราฟิก โหมด 13 ฐานสิบหก

(เรียงตามลำดับขั้นตอนการทำงานในโปรแกรม)

1. set_mode

ใช้ในการเปลี่ยนจากโหมดตัวอักษรไปเข้าสู่โหมดกราฟิก ในที่นี้จะอาศัยการเรียกใช้ฟังก์ชันของไบออส ฟังก์ชัน 10H และฟังก์ชันย่อย 00H (โดยจะไปทำการเซทรีจิสเตอร์ต่าง ๆบนการ์ดอย่างมีลำดับ)

ทำได้โดยการตั้งค่ารีจิสเตอร์ดังนี้

AH = 00H

AL = ค่าของโหมดที่จะตั้งไว้

2. get_curmode

ใช้ในการตรวจสอบว่าสามารถเข้าสู่โหมดที่ต้องการหรือไม่ โดยการที่ใช้ฟังก์ชั่นนี้ ไปเอาค่าของโหมดปัจจุบันมาตรวจสอบ จะอาศัยการเรียกใช้ฟังก์ชั่นของ ไบออส ฟังก์ชัน

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10H และฟังก์ชั่นย่อย 0FH โดยโหมดของการแสดงผลจะส่งค่ากลับ มาทางรีจิสเตอร์ AL

3. plot13vga

ใช้ในการแสดงผลบนจอทีละจุดโดยจะทำการเขียนลงหน่วยความจำภาพโดยตรงโดยหน่วย

ความจำภาพจะเริ่มที่ตำแหน่งแอดเดรส A0000H แล้วแต่ละจุดบนหน่วยความจำเรียงต่อกันแบบเชิงเส้น

โดยใช้สูตรคำนวณตำแหน่งแอดเดรสดังนี้

แอดเดรส = y * 320 + x

ตัวอย่างโปรแกรม

```
addr = y * 320 + x;
pokeb(0xA000,addr,color);
```

โดยที่

addr = ค่าออฟเซตภายในเซกเมนต์ของหน่วยความจำภาพ A0000H

color = ค่าของสีที่จะเขียน(แสดง) บนจอ

4. pad_image13h

ใช้แสดงภาพบนจอโดยจะแสดงบนจอทีละเส้น

5. writeadac

ใช้เขียนข้อมูลของรีจิสเตอร์สีแดง,เขียว และน้ำเงิน ภายในแต่ละชุดของรีจิสเตอร์สี(color regiister)

โครงสร้างของข้อมูลของรีจิสเตอร์สีแต่ละชุด

```
struct DAC_TAB{
    char red;
    char green;
    char blue;
}
```

โดยจะมีทั้งสิ้น 256 ชุด (ทำให้มีได้ทั้งสิ้น 256 สี)

ตัวอย่างโปรแกรมมีลำดับดังนี้

```
writeadac(slot,number)
struct DAC_TAB *slot;
unsigned int number;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
outportb(0x3C8,number);
outportb(0x3C9,slot->red);
outportb(0x3C9,slot->green);
outportb(0x3C9,slot->blue);
}
```

ในการตั้งค่ารีจิสเตอร์สีหลาย ๆ ชุดจะต้องการเซทไม่ให้ทำการแสดงผล (Screen-off) ไว้ก่อนแล้วจึงค่อยตั้งค่าให้ครบทั้ง 256 ตัว โดยเซทบิท Screen- off ของรีจิสเตอร์ซีเควนเซอร์ (sequencer register) อินเดกซ์ไปยังรีจิสเตอร์ โหมดจังหวะนาฬิกา (Clocking Mode register)ให้เป็น 1

ตัวอย่างโปรแกรม

```
outportb(0x3C4,0x01) /*รีจิสเตอร์อินเดกซ์ของชุดซีเควนเซอร์*/
/* โดยต้องการจะโปรแกรม Clocking Mode register */
clockmode = inportb(0x3C5);
/* เซทให้บิท Screen-off เป็น 1 ไม่มีการสแกนจอแสดงผลไว้ชั่วคราว
   เพื่อหลีกเลี่ยงการเกิดสโนว์ */
newclockmode = clockmode | 0x20;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ส่วนวิเคราะห์แยกองค์ประกอบของภาพ (Image Segmentation)

2.3.1 Image Enhancement

Image Enhancement คือการปรับปรุงคุณภาพของภาพที่จะนำมาประมวลผลให้ดีขึ้น เช่นแปลงภาพที่ไม่ชัดเจนจากการถ่ายในขณะแสงน้อยหรือจ้าเกินไปให้ชัดเจนขึ้น แปลงภาพที่มีสีหยาบไม่สม่ำเสมอให้ดูเรียบยิ่งขึ้น ฯลฯ เราจะใช้ในการปรับปรุงคุณภาพของภาพที่รับมาจากส่วนรับภาพจากการ์ด DigiImage ให้เหมาะสมกับการทำอิมเมจเซกเมนเตชัน หรือใช้ในการปรับปรุงภาพจากการทำอินเมจเซกเมนเตชันบางวิธี

2.3.2 วิธีการของ Image Enhancement

ในการศึกษาถึงระบบการทำ Image Enhancement นั้นได้ทำการศึกษาและทดลองใช้งานอยู่หลายวิธี ซึ่งจากการทดลองนั้นได้เลือกใช้วิธีที่เหมาะสมและเป็นประโยชน์ในการทำอิมเมจเซกเมนเตชันอยู่ 2 วิธีดังนี้

Image Smoothing

เป็นการปรับปรุงสีของภาพให้มีความนุ่มนวล และเรียบขึ้น เพื่อลดส่วนผิดพลาด ที่อาจมีอยู่ในภาพ วิธีนี้จะเป็นประโยชน์อย่างมากที่จะใช้กำจัดจุดที่มีค่าระดับสีแตกต่างจากจุดรอบข้างมาก ๆเนื่องจากจุดเหล่านี้จะทำให้เกิดความผิดพลาดในขบวนการอิมเมจเซกเมนเตชั่นอย่างยิ่งทั้งวิธี Edge Detection และ Region Clustering เนื่องจากในขบวนการ Edge Detection นั้นการที่จุดดังกล่าวนี้มีความแตกต่างจากจุดรอบ ๆข้างมาก ทำให้มันถูกมองว่าเป็นส่วนหนึ่งของขอบภาพ และในวิธี Region Clustering มันจะมีค่าแตกต่างจาก Region ที่อยู่รอบ ๆ มาก มันจึงไม่ถูกรวมเข้าไปใน Region ใด ๆ ทำให้ Region ซึ่งมีความแตกต่างกันน้อยจำเป็นต้องถูกรวมเข้าเป็น Region เดียวกัน เพื่อให้เหลือจำนวน Region เท่าที่ต้องการ ดังแสดงอยู่ในบทที่ 4 "การทดลองและผลการทดลอง

สำหรับการการอิมพลีเมนท์ Image Smoothing นี้ จะทำโดยใช้ lowpass filter ซึ่งจะทำหน้าที่กรองข้อมูลที่มีส่วนที่การเปลี่ยนแปลงของระดับสีสูง (high frequency) ให้มีการเปลี่ยนแปลงน้อยลง และคงสภาพส่วนที่มีการเปลี่ยนแปลงต่ำให้คงอยู่ ซึ่งวิธีการทำงานของ lowpass filter นี้จะทำโดยการเปลี่ยนค่าของจุดที่พิจารณาให้เป็นค่าเฉลี่ยระหว่างตัวมันเองกับจุดรอบ ๆข้าง

Histogram Equalizing

เป็นวิธีการปรับปรุงภาพซึ่งมีความแตกต่างระหว่างสีน้อยอันอาจเกิดจากการถ่ายภาพในขณะแสงน้อยหรือจ้ามากเกินไปให้มีความแตกต่างระหว่าง โดยวิธีการนี้จะทำให้ความแตกต่างระหว่างสีต่าง ๆกระจายได้ดีขึ้น โดยสีที่มีปริมาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริมาณการถูกใช้มากก็จะถูกเปลี่ยนให้เป็นสีซึ่งแตกต่างจากสีอื่น ๆมากยิ่งขึ้น ทำให้สามารถพิจารณาองค์ประกอบต่าง ๆของภาพได้ชัดเจนยิ่งขึ้น

กระบวนการนี้เหมาะสมที่จะนำไปใช้ในการปรับปรุงภาพซึ่งมีความคมชัดน้อยและระดับของสีรวมอยู่ในช่วงใดช่วงหนึ่ง เช่นการรวมอยู่ในช่วงระดับสีต่ำภาพถ่ายขณะแสงน้อย และการรวมอยู่ในช่วงระดับสีเข้มของภาพถ่ายขณะแสงจ้าดังภาพที่แสดงผลอยู่ในบทที่ 4

2.3.3 ทำไมจึงต้องทำอิมเมจเซกเมนเตชัน

ลักษณะการมองของมนุษย์นั้น จะรับหรือกลั่นกรองเอาส่วนที่สำคัญของภาพที่รับเข้ามาทางสายตาแล้วส่งส่วนที่สำคัญเหล่านั้นไปยังสมองเพื่อที่จะแปรความหมายของภาพที่รับเข้ามา จุดประสงค์ของการทำอิมเมจเซกเมนเตชันก็เช่นเดียวกัน กล่าวคือ การแยกหรือกลั่นกรองส่วนที่สำคัญของภาพออกมาให้เห็นเด่นชัดขึ้น และแยกส่วนที่มีค่าระดับความเข้มที่ต่างกันมากออกไปเป็นอีกส่วนหนึ่ง ซึ่งค่าระดับความแตกต่างที่มีค่าน้อยนั้นบางครั้งสายตามนุษย์ไม่สามารถแยกแยะได้ แต่เมื่อนำหลักของการประมวลผลสัญญานภาพเข้ามาใช้ร่าม ความแตกต่างนั้นจะเห็นได้ชัดขึ้น

2.3.4 หลักการของอิมเมจเซกเมนเตชัน

ดังที่ได้กล่าวมาแล้วว่า การทำอิมเมจเซกเมนเตชัน เป็นการแยกหรือกลั่นกรองส่วนที่สำคัญของภาพออกมาให้เด่นชัดขึ้น โดยจะทำการแยกส่วนพื้นที่ต่าง ๆในภาพ (region) หรือขอบรอยต่อของภาพ(edge) ออกเป็นส่วน ๆ ทั้งนี้เนื่องจากเนื้อของวัตถุที่เป็นบริเวณเดียวกัน จะมีลักษณะของจุดภาพที่มีคุณสมบัติพื้นฐานที่เหมือนกัน อย่างเช่นสี ความสว่างของภาพหรือโครงร่างของภาพจากนั้นจึงทำการแทนค่าของจุดภาพในภาพด้วยค่าต่าง ๆ ซึ่งบ่งบอกถึงคุณสมบัติของจุดหรือกลุ่มจุดนั้น เช่นค่าแสดงความเป็นขอบภาพหรือไม่ ค่าแสดงหมายเลขกลุ่มของจุด (Segment) การทำเซกเมนเตชันนั้นเราสามารถใช้หลักการต่าง ๆ ดังต่อไปนี้ได้

- ***การตรวจหาขอบภาพ (Edge Detection)*** เป็นวิธีการหาขอบต่าง ๆภายในภาพ โดยอาศัยคุณสมบัติที่ว่า ขอบรอยต่อของภาพต่าง ๆซึ่งจะมีระดับของสีแตกต่างกันมากว่าพื้นภาพของวัถถุ ซึ่งวิธีการตรวจสอบว่าจุดนั้น ๆเป็นรอยต่อของภาพหรือไม่สามารถทำได้หลายวิธีดังจะกล่าวต่อไป ซึ่งขอบภาพนี้สามารถที่จะนำมาใช้ในขบวนการของการวิเคราะห์แยกองค์ประกอบของภาพได้ เพราะว่าลักษณะของขอบวัตถุหรือลักษณะขอบขององค์ประกอบต่าง ๆของวัตถุสามารถที่จะไช้ตรวจสอบวัตถุนั้น ๆได้ได้

- ***Region Clustering*** เป็นวิธีการทำเซกเมนเตชันที่อาศัยคุณสมบัติที่เหมาะสมของจุดภาพในการสร้างส่วนของภาพ โดยที่ภายในบริเวณเดียวกันจะต่อถึงกันและมีค่าระดับสีเทาใกล้เคียงกัน ในปัจจุบันนี้มีวิธีการที่ใช้หลักการอันนี้หลายวิธีด้วยกัน เช่น split and merge วิธีการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รนี้จะเริ่มต้นด้วยการเลือกจุดภาพขึ้นมาก่อนหนึ่งจุด ต่อจากนั้นก็จะทำการพิจารณาจุดภาพที่อยู่ใกล้เคียง (neighbours) เพื่อที่จะรวมเข้าเป็นส่วนเดียวกัน จุดภาพที่อยู่ในบริเวณใกล้เคียงทุกจุดที่อยู่ภายใต้*เกณฑ์การรวม*จะกลายเป็นส่วนของภาพที่เกิดขึ้นในภาพนั้น แต่ถ้าจุดภาพที่อยู่ใกล้เคียงนั้นตรวจสอบแล้วไม่อยู่ภายใต้ของเกณฑ์การรวม จุดภาพนั้นจะไม่ถูกรวมเข้าไปในส่วนนั้นของภาพแต่จะถูกเลือกให้เป็นจุดเริ่มต้นของส่วนอื่น ๆ ต่อไป

การตรวจหาขอบภาพ (Edge Detection)

วิธีการตรวจหาขอบต่าง ๆภายในภาพนั้นมีอยู่ด้วยกันหลายวิธีแบ่งออกเป็นกลุ่มใหญ่ ๆได้ 2 กลุ่ม คือกลุ่มที่ใช้การประมวลผลแบบ frequency domain ซึ่งผลของของจุด ๆหนึ่งจะขึ้นอยู่กับทุก ๆจุดในภาพและวิธี Special Domain ซึ่งผลของจุดหนึ่ง ๆจะขึ้นอยู่กับจุดซึ่งอยู่บริเวณรอบ ๆจุดนั้นเท่านั้น ซึ่งวิธีที่กลุ่มของข้าพเจ้าทดลองใช้งานแล้วได้ผลเป็นที่น่าพอใจก็มีอยู่ด้วยกัน 3 วิธี ซึ่งใช้การประมวลผลแบบ Special Domain ทั้งหมด

Sum Square Delta

เป็นวิธีวิเคราะห์ความแตกต่างของกลุ่มข้อมูลรอบ ๆจุดที่พิจารณาโดยการนำค่าระดับสีของจุดที่พิจารณาและจุดรอบ ๆข้างมาหาค่าเฉลี่ย แล้วหาค่าผลรวมของ ค่าความแตกต่างยกกำลังสองของทุก ๆจุดรอบ ๆกับค่าเฉลี่ยนั้น จากนั้นนำค่าที่ได้ไปทดสอบกับค่า Threshold ทีกำหนดไว้ หากมีค่าสูงกว่าที่กำหนดจะให้ความหมายว่าจุดนั้นเป็นส่วนหนึ่งของขอบภาพ หากมีค่าน้อยกว่า แสดงว่าจุดดังกล่าวไม่ใช่ส่วนของขอบภาพ

Zero-Crossing (Gradient)

เป็นขบวนการวิเคราะห์ความแตกต่างของกลุ่มข้อมูลรอบ ๆจุดที่พิจารณาเพื่อหาขอบภาพเช่นเดียวกับวิธีคำนวณแบบ Sum Square Delta โดยการ Differential เพื่อหาค่าอัตราการเปลี่ยนแปลงรอบจุด ๆนั้นในแนวแกน X และ Y โดยอาจเลือกพิจารณาในแกนใดแกนหนึ่งหรือทั้งสองแกนก็ได้

High-Pass Filter

เป็นขบวนการวิเคราะห์ความแตกต่างของกลุ่มข้อมูลรอบ ๆจุดที่พิจารณา

เช่นเดียวกับสองวิธีแรก โดย Filter นี้จะทำหน้าที่กรองข้อมูลที่มีส่วนที่การเปลี่ยนแปลงของระดับสีต่ำ (low frequency) ให้ถูกลดทอนหายไป และคงสภาพส่วนที่มีการเปลี่ยนแปลงสูงให้คงอยู่ โดยใช้วิธี high-pass filter นี้จะใช้วิธี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การปรับค่าจุดที่อยู่ตรงกลางให้มีค่าเท่ากับค่าผลรวมของผลต่างระหว่างตัวมันกับจุดรอบ ๆ

Thresholding

วิธีการของ Thresholding นี้ อาศัยพื้นฐานของฮีสโตแกรม (histogram) ของระดับสีเทาที่กระจายอยู่ในภาพเป็นตัวแบ่งส่วนของภาพ โดยการกำหนดว่าถ้าฮีสโตแกรมแยกออกเป็นหลาย ๆ กลุ่มที่เด่นชัด กลุ่มเหล่านี้จะเป็นตัวแสดงคุณสมบัติที่แตกต่างของภาพนั้น ๆ ค่าเธร็สโฮลด์ (threshold) จะถูกเลือกจากตำแหน่งรอยต่อระหว่างกลุ่มสี(mode) ของกลุ่มข้อมูลในฮีสโตแกรม แล้วจึงนำมาใช้ในการแบ่งส่วนของภาพนั้น ซึ่งวิธีนี้ส่วนของภาพที่ได้จะขึ้นอยู่กับการกระจายของการใช้สีต่าง ๆในฮีสโตแกรมของภาพ เราจะใช้วิธีการนี้ ในการแปลงภาพ grey level หลาย ๆระดับ ให้มีค่าระดับสีลดลงเหลือเท่ากับจำนวนระดับ Threshold + 1 ซึ่งโดยทั่วไปและในโครงงานนี้ การใช้ Thresohold จะหมายถึงการใช้ Threshold ซึ่งมีค่าระดับเดียว เป็นผลทำให้ระดับสีที่ได้เหลือเพียง 2 ระดับ (ภาพไบนารี่)

เราจะใช้ขบวนการนี้ในการแปลงภาพซึ่งได้จากกรรมวิธีหาขอบภาพทั้ง 3 วิธี ดังที่กล่าวไว้ในข้างต้นให้เป็นภาพไบนารี่ ซึ่งมีค่าจุดที่เป็นองค์ประกอบของขอบภาพเป็น 1 และจุดพื้นภาพทั่ว ๆไปเป็น 0 โดยในการทำ Threshold นี้ เราจะต้องเลือกค่า Threshold ให้เหมาะสม หากมีค่าสูงเกินไปจะทำให้ขอบภาพบางส่วนขาดหายไป หากต่ำเกินไปพื้นภาพซึ่งมีระดับสีไม่เรียบก็จะถูกระบุให้เป็นส่วนหนึ่งของขอบภาพ การพิจารณาค่า threshold ที่เหมาะสมสามารถแสดงได้ดังรูปที่ 2.3.1

รูปที่ 2.3.1 กราฟระหว่างค่าความสว่างกับจำนวนจุด จากกราฟจุด T คือค่าระดับกึ่งกลางที่ดีที่สุด

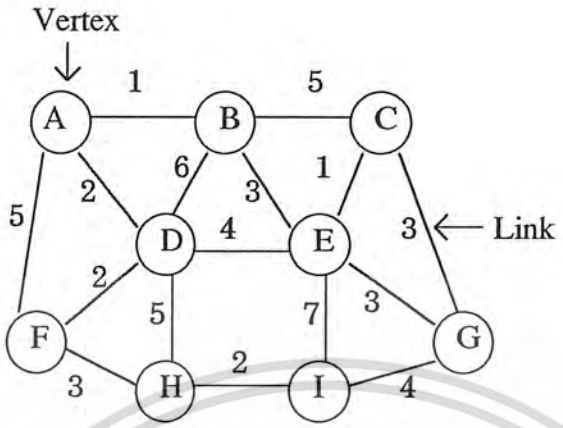
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Region Clustering

เป็นวิธีการทำเซกเมนเตชันโดยการรวมจุดต่าง ๆของภาพซึ่งมีระดับสีใกล้เคียงกันและอยู่ติดกัน ให้เป็นกลุ่มเดียวกัน วิธีการนี้จะเริ่มต้นด้วยการเลือกจุดภาพขึ้นมาก่อนหนึ่งจุด ต่อจากนั้นก็จะทำการพิจารณาจุดรอบ ๆที่อยู่ใกล้เคียง เพื่อที่จะรวมเข้าเป็นส่วนเดียวกัน จุดภาพที่อยู่ในบริเวณใกล้เคียงทุกจุดที่อยู่ภายใต้*เกณฑ์การรวม*จะกลายเป็นส่วนของภาพที่เกิดขึ้นในภาพนั้น แต่ถ้าจุดภาพที่อยู่ใกล้เคียงนั้นตรวจสอบแล้วไม่อยู่ภายใต้ของ*เกณฑ์การรวม* จุดภาพนั้นจะไม่ถูกรวมเข้าไปในส่วนนั้นของภาพแต่จะถูกเลือกให้เป็นจุดเริ่มต้นของส่วนอื่น ๆ ต่อไป และหลังจากที่จุดภาพทุกจุดได้รวมตัวกันเป็นกลุ่ม(region)เรียบร้อยแล้วการที่จะทำให้ได้ส่วนของภาพสมบูรณ์ขึ้น ก็สามารถที่จะเริ่มใหม่โดยการแยกแต่ละส่วนออกแล้วใช้หลักการรวมกันของจุดภาพที่กล่าวมาแล้วอีกครั้งหนึ่ง วิธีการของ Region Clusterin จะพยายามหาจุดภาพเริ่มต้นในการสร้าง region เพื่อให้ได้ผลที่ดีที่สุดในการรวบรวมจุดภาพ วิธีการทำเซกเมนเตชันที่ใช้หลักการอันนี้อีกวิธีหนึ่งคือ การทำอิมเมจเซกเมนเตชันด้วยการแบ่งแยก และรวมโดยตรง (Directed Spilt-and-Merge)[7] วิธีการนี้อาศัยโครงสร้าง Quartic Picture Tree ของข้อมูล แต่จะมีข้อเสียในขั้นตอนของการรวมจุดภาพเข้าด้วยกัน เนื่องจากในขั้นตอนดังกล่าวจะขาดความอิสระในการรวมจุดภาพ ดังนั้นจึงได้มีผู้เสนอวิธีการใหม่คือ วิธีการแบ่งและรวบรวมที่มีการปรับปรุง(Modification of Split- and-Merge)[8] ถึงแม้ว่าวิธีการนี้จะให้ผลดีกว่าสองวิธีการที่กล่าวมาแล้วข้างต้นก็จริง แต่อย่างไรก็ตามการทำเซกเมนเตชันด้วยหลักการของ region clustering ก็ยังมีข้อจำกัดอยู่ คือไม่สามารถที่จะสร้างส่วนของภาพที่เป็นเส้นที่มีขนาดเล็กในภาพได้ดีพอ

ในหัวข้อนี้จะเป็นการกล่าวนำถึงความหมายของคำศัพท์ และรายละเอียดของทฤษฎีกราฟที่จำเป็นที่ต้องนำมาใช้ในการทำเซกเมนเตชันภาพ ทฤษฎีกราฟเป็นการศึกษาเกี่ยวกับกราฟและการนำมาประยุกต์ใช้งานของกราฟ โดยให้ G = (V,E) เป็นกราฟที่ประกอบด้วยจุดยอด (vertices) ที่ต่อเชื่อมกับจุดยอดอื่น ๆ โดยตัวเชื่อม (Links) E_{ij} จะเชื่อมต่อจุดยอด V_i และ V_j เข้าด้วยกัน ในขณะที่น้ำหนัก (Weight) จุดยอดของกราฟ V_i คือ v_i และค่าน้ำหนักของตัวเชื่อม (Link Weight) $E_{i,j}$ คือ $e_{i,j}$ จุดยอดแต่ละจุดของกราฟไม่จำเป็นที่จะต้องเชื่อมต่อกับจุดยอดอื่น ๆทุกจุด แต่ถ้ามีการเชื่อมทุก ๆ จุดก็จะเป็นกราฟที่สมบูรณ์ รูปที่ 2.3.2 เป็นตัวอย่างของกราฟที่ประกอบด้วยจุดยอด 9 จุดคือ A , B , C , D , E , F , G , H และ I ตามลำดับ พร้อมกันนี้ได้แสดงค่าน้ำหนักของตัว เชื่อมระหว่างจุดยอดแต่ละจุดด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3.2 แสดงกราฟตัวอย่าง

สำหรับคำศัพท์ที่สำคัญ ๆ ในทฤษฎีกราฟที่จำเป็นต้องนำมาใช้ในการทำเซกเมนเตชันของภาพ ดังมีรายละเอียดดังต่อไปนี้

กราฟย่อย (Patial Graph) เป็นกราฟที่ประกอบด้วยจำนวนจุดยอดเท่ากับกราฟต้นแบบ(Original Graph) แต่จะมีตัวเชื่อมเป็นซับเซ็ต(subset) ของกราฟต้นแบบ ดังตัวอย่างในรูปที่ 2.3.3

ลูกโซ่ (Chain) เป็นจำนวนของจุดยอดที่ต่อเนื่องกัน โดยจุดยอดแต่ละจุดจะเชื่อมต่อกับจุดยอดถัด ๆ ไปด้วยตัวเชื่อมภายในกราฟ

ไซเคิล(Cycle) เป็นทางเดินลูกโซ่ที่มีจุดยอดเริ่มต้นและจุดยอดสุดท้ายเป็นจุดเดียวกันดังตัวอย่างของจุดยอด A , B , E และ D ในรูปที่ 2.3.2

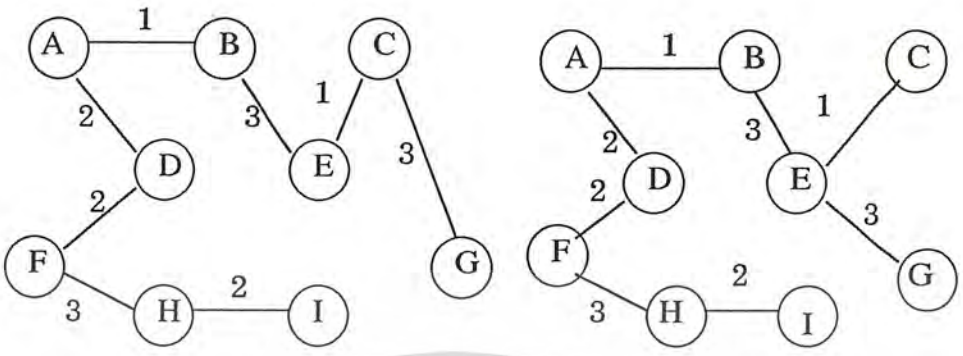
ทรี (Tree) เป็นเซ็ตของลูกโซ่ที่ต่อถึงกันในกราฟ แต่จะไม่ครบวงจรหรือไม่เป็นไซเคิลดังนั้นทรีที่เชื่อมต่อด้วยจุดยอด N จุด จะมีตัวเชื่อมทั้งหมดเท่ากับ N-1

สแพนนิ่งทรี (Spanning Tree) คือทรีที่เป็นกราฟย่อย

ช็อตเตสท์สแพนนิ่งทรี (Shortest Spanning Tree) หรือ SST เป็นสแพนนิ่งทรีที่ให้ผลรวมของน้ำหนักตัวเชื่อมมีค่าต่ำสุดเท่าที่จะเป็นไปได้ ดังนั้นช็อตเตสท์สแพนนิ่งทรีจึงไม่จำเป็นที่จะต้องมีเพียงกรณีเดียวดังตัวอย่างในรูปที่ 2.3.3 แสดงลักษณะ ของกราฟย่อยที่เป็น SST ด้วย

ฟอเรสท์(Forest) เป็นเซ็ตของทรีที่อยู่ในกราฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3.3 แสดงกราฟย่อยที่เป็นช็อตเตสท์สแพนนิ่งทรี

จากคำศัพท์ต่าง ๆที่กล่าวมา จะถูกนำมาใช้ในขั้นตอนของการแปลงข้อมูลภาพให้อยู่ในรูปของกราฟ สำหรับการทำเซกเมนเตชัน การแปลงข้อมูลภาพไปเป็นกราฟ

การใช้ทฤษฎีกราฟในการวิเคราะห์ข้อมูลภาพหรือการทำเซกเมนเตชันนั้น ข้อมูลของภาพ (original image) จะถูกเปลี่ยนให้อยู่ในรูปของกราฟเสียก่อน โดยการแปลงจุดภาพแต่ละจุดให้เป็นจุดยอดของกราฟขึ้นอยู่กับระดับความเข้มของจุดภาพ

ดังนั้นถ้าความเข้มหรือระดับสีเทาของจุดภาพที่ตำแหน่ง (x,y) คือ f(x,y) แล้วน้ำหนักจุดยอดของกราฟก็คือ

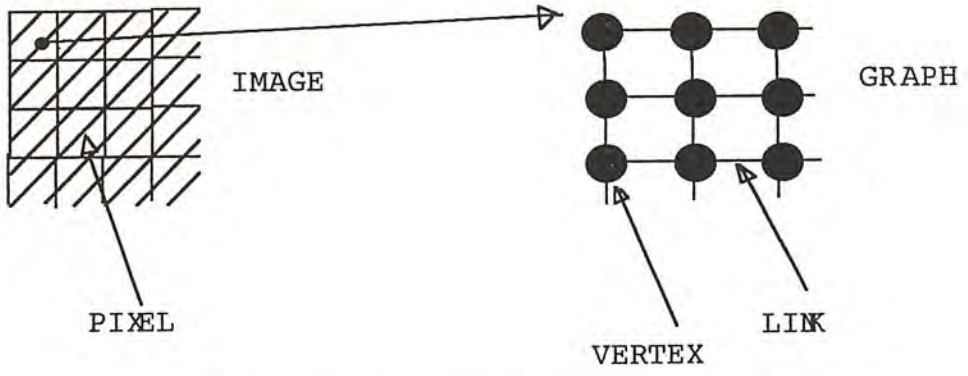
$$V_i = f(x,y) \qquad (2.3.1)$$

โดยที่ x,y ถูกแปลง(map) ไปเป็น i ในลักษณะ one-to-one mapping หลังจากนั้นทำการกำหนดค่าน้ำหนักตัวเชื่อมของจุดยอดต่าง ๆ ซึ่งสามารถที่จะกำหนดได้หลายแบบแต่ในที่นี้จะใช้เป็นค่าสมบูรณ์ (absolute value) ของค่าความแตกต่างระหว่างจุดยอดของกราฟที่อยู่ข้างเคียง ซึ่งเป็นการวัดความเหมือนหรือความใกล้เคียง (similarity) ของระดับความเข้มระหว่างจุดยอดของกราฟที่อยู่ข้างเคียงซึ่งเป็นการวัด ความเหมือนหรือความใกล้เคียง (similarity) ของระดับความเข้มระหว่างจุดภาพที่อยู่ใกล้ ๆ กันนั้นเอง จะได้ว่า

$$e_{i,j} = |V_i - V_j| \qquad (2.3.2)$$

จุดยอดของกราฟแต่ละจุดสามารถที่จะเชื่อมกับจุดยอดอื่น ๆ ภายในกราฟ แต่ที่มีประโยชน์และเพื่อลดความยุ่งยาก จะใช้การเชื่อมต่อของจุดยอดเฉพาะจุดยอดที่อยู่ใกล้กันที่สุดเท่านั้น โดยเซกเมนเตชันเราจะใช้การเชื่อมต่อแบบ 4 ทิศทาง และ 8 ทิศทาง แต่ในที่นี้การทำเซกเมนเตชันเราจะใช้การเชื่อมต่อแบบ 4 ทิศทาง เพื่อความสะดวกและเป็นการลดขนาดของข้อมูลในตอนประมวล(process) จากรูปที่ 2.3.4 เป็นตัวอย่างการแปลงข้อมูลภาพเดิมให้เป็นกราฟสำหรับนำไปดำเนินการในขั้นต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3.4 แสดงการแปลงข้อมูลภาพไปเป็นกราฟ

การหาซ็อตเตสท์สแพนนิ่งทรีของกราฟ

วิธีการในการคำนวณหาซ็อตเตสท์สแพนนิ่งทรีของกราฟมีรูปแบบการคำนวนอยู่หลายวิธี เมื่อภาพถูกแปลงให้อยู่ในรูปของกราฟที่มีความสัมพันธ์แบบ 4 ทิศทาง วิธีการ Kruskal's Algorithm [10] ถูกนำมาใช้ ในการคำนวณของวิธีนี้จะเป็นสัดส่วนกับจำนวนของตัวเชื่อมคูณด้วยค่าล็อก (log) ของจุดยอดหรือเท่ากับ E.log V เมื่อ E เป็นจำนวนของตัวเชื่อม และ V เป็นจำนวนจุดยอดในกราฟ วิธีการนี้จะสร้างซ็อตเตสท์สแพนนิ่งทรีโดยการเชื่อมทรีของฟอเรสท์เข้าด้วยกันด้วยตัวเชื่อมที่มีค่าน้ำหนักต่ำที่สุด เริ่มต้นจากจุดยอดทุก ๆจุด ทำการหาค่าตัวเชื่อมที่มีค่าต่ำสุด แล้วจึงทำการสร้างทรีขึ้นมาจากจุดยอดเหล่านี้ โดยค่อย ๆเชื่อมจุดยอดของกราฟที่มีค่าน้ำหนักของตัวเชื่อมน้อยที่สุดขึ้นเป็นทรีก่อน แล้วจึงเชื่อมทรีเหล่านี้เข้าด้วยกันเป็นซ็อตเตสท์สแพนนิ่งทรี ด้วยตัวเชื่อมที่มีค่าน้ำหนักต่ำสุดเช่นกัน จนกระทั่งจุดยอดทุกจุดบนกราฟกลายเป็นสแพนนิ่งทรีของกราฟ

รายละเอียดของวิธีการหาซ็อตเตสท์สแพนนิ่งทรีมีดังต่อไปนี้

ขั้นตอนที่ 1 จัดเรียงลำดับค่าน้ำหนักตัวเชื่อมของกราฟจากค่าต่ำไปหาค่าสูง

ขั้นตอนที่ 2 เริ่มจากฟอเรสท์ของกราฟที่ยังไม่มีตัวเชื่อม

ขั้นตอนที่ 3 กระทำซ้ำดังนี้

- หาตัวเชื่อมที่มีค่าน้ำหนักต่ำสุดตัวต่อไป
- ถ้าตัวเชื่อม ๆต่อกันแล้วไม่ทำให้เกิดเป็นไซเคิล

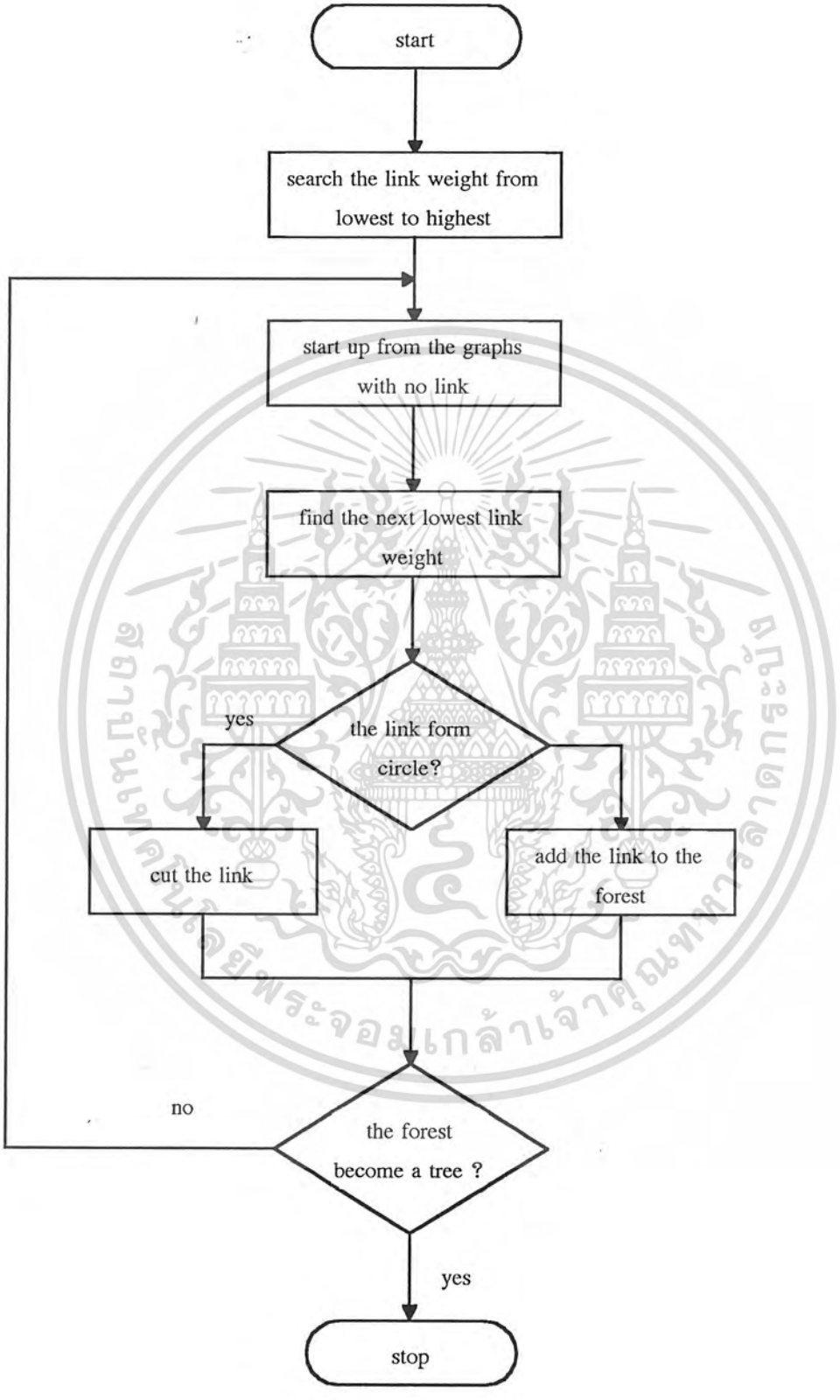
  [รวมตัวเชื่อมตัวนี้เข้าไปในฟอเรสท์]
- ถ้าทำให้เกิดไซเคิล

  [ตัดตัวเชื่อมนี้ทิ้งไป]

ขั้นตอนที่ 4 จนกระทั่งฟอเรสท์กลายเป็นสแพนนิ่งทรีของกราฟ

จากลำดับหัวข้อทั้ง 9 สามารถที่จะเขียนเป็นโฟลว์ชาร์ตของการหาซ็อตเตสท์สแพนนิ่ง ทรีของกราฟได้ดังรูปที่ 2.3.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3.5 แสดงโฟลว์ชาร์ต การหา SST ของกราฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแบ่งส่วนภาพจากสแพนนิ่งทรีของกราฟ (Segmentation from Spanning Tree)

เมื่อการทำอิมเมจเซกเมนเตชันถึงการแบ่งภาพออกเป็น region ย่อยๆ ที่ไม่ซ้อนทับกันและทุกๆ region ยังคงเป็นส่วนประกอบของภาพอยู่เช่นเดิม ดังนั้นสแพนนิ่งทรีซึ่งเป็นส่วนของกราฟที่ได้มาจากการแปลงข้อมูลภาพ จึงสามารถที่จะนำมาใช้ในการแบ่งส่วนของภาพได้ โดยที่เซกเม้นต์แต่ละเซกเมนต์จะได้มาจากการตัดตัวเชื่อมของสแพนนิ่งทรี ฟอเรสท์ของสแพนนิ่งทรีที่โดนตัดจะแทนแต่ละส่วนของภาพ โดยถ้า T เป็นทรีในฟอเรสท์ ดังนั้น P(T) 1 สามารถกำหนดได้ดังนี้

$$P(T)_i = \begin{cases} 1 \text{ ถ้า } V_i \text{ เป็นสมาชิกของ T} \\ 0 \text{ เมื่อเป็นอย่างอื่น} \end{cases} \qquad (2.3.3)$$

ถ้ากำหนดให้ M x,y เป็นการแปลง (mapping) จาก (x,y) ไปเป็น T ดังนั้น $P(T)_i$ ของกราฟสามารถแปลงกลับไปเป็นภาพได้ โดยที่ทุกๆตัวของ $P(T_i)$ จะเป็นตัวกำหนดขอบเขตและจำนวนของจุดภาพในแต่ละเซกเม้นต์ของภาพ และแต่ละเซกเม้นต์จะมีค่าระดับของความเข้มเป็นค่า ๆ หนึ่ง เมื่อกำหนดให้ p(T) เป็นค่าความเข้มของเซกเม้นต์นั้น ๆจะได้ว่า

$$p(T) = \frac{\sum_i P(T)_i \cdot V_i}{\sum_i P(T)_i} \quad \text{สำหรับทุกค่าของ i} \qquad (2.3.4)$$

ในที่นี้ p(T) จึงหมายถึงค่าเฉลี่ยน้ำหนักของจุดยอดของทรี ที่ใช้เป็นค่าความเข้มของเซกเม้นต์ในภาพ ส่วนการแปลงจากภาพเดิมไปเป็นภาพเซกเมนเตชัน $S_{x,y}$ นั้นสามารถที่จะเขียนได้ดังนี้

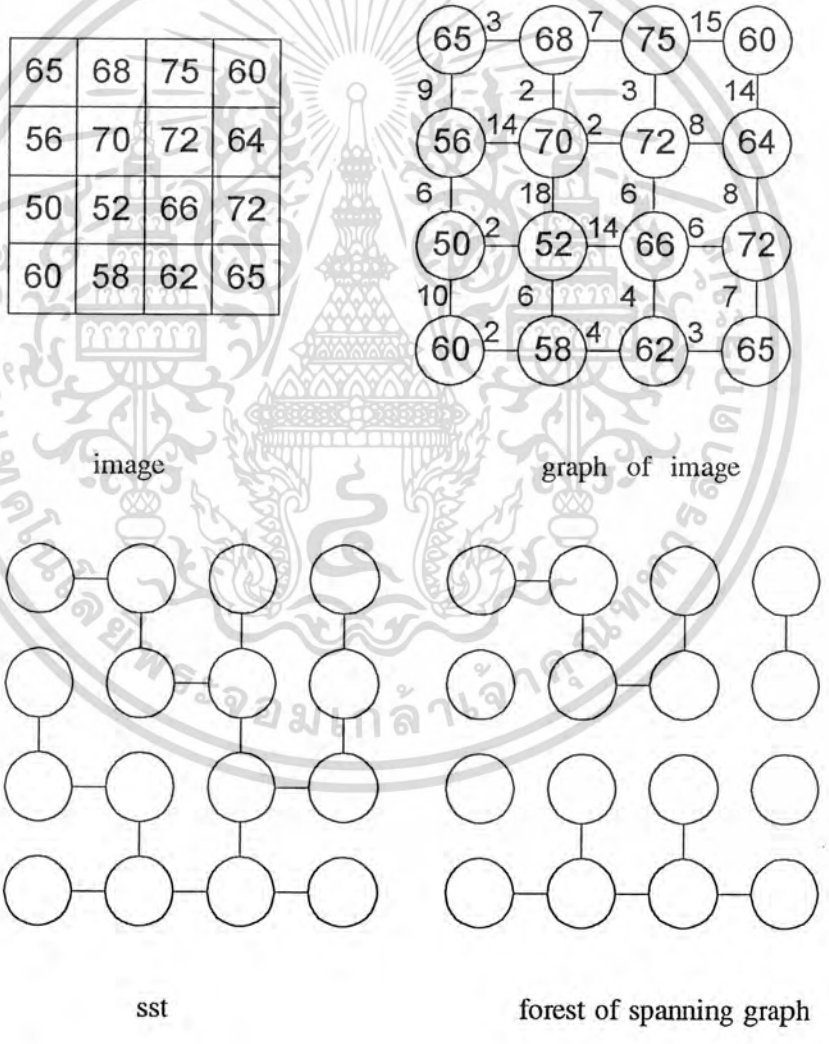
$$S_{x,y} = p(M_{x,y}) \qquad (2.3.5)$$

จากกราฟที่เราได้ทำการคำนวณหา SST ไว้แล้ว สามารถนำมาใช้ในการทำเซกเมนเตชันโดยการตัด SST ของกราฟตรงตำแหน่งที่มีค่าของตัวเชื่อมที่มีค่าสูงสุด ทั้งนี้เพื่อให้แน่ใจว่าแต่ละเซกเม้นต์ของภาพที่แยกกันนั้นมีค่าความแตกต่างจากบริเวณจุดภาพที่ใกล้เคียงมากที่สุด และยังเป็นตัวยืนยันได้ว่าจุดภาพทุกๆจุด ที่อยู่ในเซกเม้นต์เดียวกัน จะมีคุณสมบัติใกล้เคียงกับจุดภาพที่อยู่ใกล้ ๆ กันในของแต่ละเซกเม้นต์มากกว่าจุดภาพอื่น ๆ ที่อยู่ต่างเซกเม้นต์ออกไป และเซกเม้นต์ของภาพที่แยกกันนั้นมีค่าความแตกต่างจากบริเวณจุดภาพที่ใกล้เคียงมากที่สุด และยังเป็นตัวยืนยันได้ว่าจุดภาพทุกๆ จุด ที่อยู่ในเซกเม้นต์เดียวกัน จะมีคุณสมบัติใกล้เคียงกับจุดภาพที่อยู่ใกล้ๆกันในของแต่ละเซกเม้นต์มากกว่าจุดภาพอื่น ๆที่อยู่ต่างเซกเม้นต์ออกไป และเซกเม้นต์ต่อๆ ไปสามารถหาได้โดยการตัดตัวเชื่อมของ SST ตัวต่อไป วิธีการในการทำเซกเม้นเตชั่นจาก SST สามารถสรุปได้ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ทำการแปลงข้อมูลภาพให้อยู่ในรูปของกราฟ
2. ทำการหา SST ของกราฟ
3. ทำการตัด SST ของกราฟเป็นจำนวน N-1ครั้ง จะทำให้ได้ภาพที่ประกอบด้วย N เซกเม้นต์
4. ทำการแปลงกราฟที่ถูกตัดแล้วซึ่งเรียกว่า ฟอเรสท์ กลับเป็นภาพเซกเม้นต์

รูปที่ 2.3.6 เป็นตัวอย่างข้อมูลภาพขนาด 4 x 4 จุดภาพนำมาแปลงเป็นกราฟ แล้วหา SST ของกราฟ ต่อจากนั้นจึงนำมาทำเซกเมนเตชันดังรูปที่ 2.3.7 ตามขั้นตอนที่กล่าวมาแล้วข้างต้น



รูปที่ 2.3.6 แสดง SST ของกราฟและการได้มาของภาพเซกเมนเตชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

70	70	70	62
56	70	70	62
60.6	60.6	60.6	60.6
60.6	60.6	60.6	60.6

segment image

รูปที่ 2.3.7 แสดงภาพเซกเม้นต์ที่ได้จากข้อมูลในรูปที่ 2.3.6

การแบ่งส่วนภาพจากรีเคอร์ซีฟช็อตเตสท์สแพนนิ่งทรี (RSST)

หมายเหตุ กรรมวิธี RSST นี้เป็นวิธีที่ถูกปรับปรุงขึ้นเพื่อใช้ในการแก้ปัญหาของ SST ซึ่งในส่วนของ RSST นี้ กลุ่มของข้าพเจ้ามีเวลาไม่เพียงพอในการทดลองอิมพลิเมนต์ แต่เนื่องจากเมื่อพิจารณาในแง่ทฤษฎีและหลักการแล้วพบว่ามีความน่าเชื่อ ถือมากพอสมควร กลุ่มของข้าพเจ้าจึงนำทฤษฎีในส่วนนี้มารวมไว้ เพื่อเป็นประโยชน์ต่อผู้ที่สนใจในภายภาคหน้า ฉะนั้นภาพและผลการทดลองต่าง ๆของกระบวนการนี้ จึงมาจากการอ้างอิงจากวิทยานิพนต์ที่อ้างอิงไว้ทั้งสิ้น

เนื่องจากขั้นตอนในการสร้าง SST ของกราฟนั้น ได้จากการจัดเรียงค่าลิ้งเวทจากค่าต่ำสุดในการจัดเรียงค่าลิ้งเวทแต่ละครั้งหมายถึงการรวมเอา region ทั้งสองที่ถูกเชื่อมต่อด้วยลิ้งเวทนั้นเข้าด้วยกัน ภายหลังการรวม region แล้วลิ้งเวทอื่น ๆ ที่ เชื่อมต่อกับ region นั้นมิได้มีการปรับเปลี่ยนค่าให้สอดคล้องกับค่าเฉลี่ยของ region ขบวนการดังกล่าวก่อให้เกิดข้อเสียสองประการ กล่าวคือ ประการแรก ถ้าหากเกิดสัญญาณรบกวนในภาพก่อนทำการเซกเมนเตชันจะทำให้จุดภาพที่เป็นสัญญาณรบกวนถูกแยกออกมาเป็นเซกเม้นต์เล็ก ๆ ซึ่งจะมีผลเสียโดยตรงต่อการลดข้อมูลภาพ ประการสุดท้ายจะเกิดปัญหาเนื่องจากบางครั้งจุดภาพขาวกับจุดภาพดำที่แตกต่างกันมาก อาจถูกนำมารวมอยู่ในเซกเม้นต์เดียวกัน ถ้าหากจุดภาพทั้งสองดังกล่าวถูกเชื่อมต่อด้วยกลุ่มจุดภาพที่มีค่าระดับความเข้มสว่างเรียงไล่กันไป ดังนั้นค่าความแตกต่างของระดับความเข้มสว่างของจุดภาพที่ใกล้กันจะมีค่าแตกต่างกันน้อย ทำให้ค่าลิ้งเวทที่ได้จากกลุ่มจุดภาพดังกล่าวมี ค่าต่ำ ๆ จึงมีผลทำให้ต้องรวมจุดภาพขาวกับจุดภาพดำอยู่ใน region เดียวกัน

ปัญหาต่าง ๆของวิธี SST นั้นเกิดจากค่าเวทที่ใช้ กล่าวคือไม่ได้มีการปรับเปลี่ยนค่าเวทและค่าลิ้งเวทให้สอดคล้องกับค่าเฉลี่ยของ region ซึ่งเป็น global information ดังนั้นในการแก้ปัญหาของ SST สามารถทำได้โดยการใช้วิธีของ Recursive Shortest Spanning Tree (RSST) ซึ่งใน RSST นี้จะได้ทำการปรับเปลี่ยนค่าลิ้งเวทต่าง ๆ ที่เชื่อมต่อกับ region ผลลัพธ์ให้สอดคล้องกับค่าเฉลี่ยของ region (ซึ่งถือเป็นเวทของ region นั้นเอง) ที่ได้จากการกำจัดค่าลิ้งเวทต่ำสุด อันเป็นการรวมจุดภาพเข้ามาเป็นส่วนหนึ่งของ region นั้นเอง ดังนั้นในการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น รวมทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รวมจุดภาพแต่ละครั้งจะมีการปรับปรุงค่าลิ้งเวทตามค่าเฉลี่ยของ region ใหม่ตลอดเวลา วิธีการแบบนี้จึงถูก เรียกว่า รีเคอร์ซีฟช็อตเตสท์สแพนนิ่งทรี (Recursive Shortest Spanning Tree)

เนื่องจากทุก ๆ จุดภาพในแต่ละ region จะมีค่าระดับสีเทาเท่ากัน ซึ่งอาจกล่าวได้ว่าจุดยอดของกราฟจุดหนึ่งสามารถใช้แสดงแทนตลอดทั้ง region ถ้าหากตัด SST ออกเป็นฟอเรสท์ (เพื่อแบ่งเป็น region ต่าง ๆ) แต่ละทรีจะกลายเป็นจุดยอดจุดหนึ่ง ดังนั้นในการแปลงจากกราฟกลับไปเป็นภาพจะเป็นการแปลงแบบหนึ่งจุดไปยังหลายจุด และแต่ละpartition $P(i)_{x,y}$ ที่ถูกกำหนดโดยจุดยอด V_i ของกราฟจะเป็นไป ตามสมการ (2.3.6)

$$P(i)_{x,y} = \begin{cases} 1 \text{ ถ้า } V_i \text{ map ไปเป็น } (X,Y) \\ 0 \text{ เมื่อเป็นอย่างอื่น} \end{cases} \qquad (2.3.6)$$

และค่าเวทของจุดยอดของกราฟสามารถกำหนดได้จาก

$$V_i = \frac{\sum_{x,y} P(i)_{x,y}.f(x,y)}{\sum_{x,y} P(i)_{x,y}} \quad \text{สำหรับทุก ๆค่าของ x และ y} \qquad (2.3.7)$$

f(x,y) คือค่าระดับสีเทาของจุดภาพ ณ ตำแหน่ง x,y

การแปลงข้อมูลภาพไปเป็นกราฟในตอนเริ่มแรกนั้นเหมือนกับวิธีแรกที่กล่าวมาแล้ว แต่ในขั้นตอนต่อ ๆไป โครงสร้างของกราฟจะถูกทำให้เปลี่ยนแปลงตลอดเมื่อมีการคำนวณซ้ำ ๆ (recursion) ในตอนเริ่มต้นจุดภาพแต่ละจุดจะถูกพิจารณาเป็นส่วนของภาพที่แยกกันเป็นคนละ region จากนั้นจะทำการรวมเอาสอง region เข้าด้วยกัน การรวมจุดยอดจะเริ่มจากจุดยอด V_i และ V_j ที่มีค่าความแตกต่างน้อยที่สุด จะถูกรวมเข้าด้วยกันได้เป็นจุดยอดใหม่ V_k ซึ่งเป็นตัวกำหนดค่าเฉลี่ยของ region ใหม่นั้นเองโดยที่

$$V_k = \frac{\sum_{x,y} (P(i)_{x,y}.f(x,y)+P(j)_{x,y}.f(x,y))}{\sum_{x,y} (P(i)_{x,y}+P(j)_{x,y})} \quad ; \text{สำหรับทุกค่าของ X,Y} \qquad (2.3.8)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ region ใหม่ที่ได้คือ $P(k)_{x,y}$ ซึ่งเป็นผลจากการรวม region ทั้งสองเข้า ด้วยกันนั้น คือ

$$P(k)_{x,y} = P(i)_{x,y} + P(j)_{x,y} \qquad (2.3.9)$$

ลิ้งที่เชื่อมต่อจุดยอด V_i และ V_j จะถูกตัดออกและเก็บเอาไว้เพื่อนำมาใช้ในขั้นตอนการแปลงกราฟกลับเป็นภาพ ลิ้งอื่นทุก ๆตัวที่มีจุดยอดใกล้เคียงกับจุดยอด V_i และ V_j จะถูกคำนวนค่าของลิ้งเวทใหม่

ขั้นตอนและวิธีการของการทำเซกเมนเตชันโดยใช้รีเคอร์ซีฟช็อตเตสท์สแพนนิ่งทรีของกราฟมีดังต่อไปนี้

ขั้นตอนที่ 1 แปลงข้อมูลภาพไปอยู่ในรูปจุดยอดของกราฟ

ขั้นตอนที่ 2 ในขณะที่มีจุดยอดของกราฟมากกว่าหนึ่งจุด

- หาค่าเวทของลิ้งที่มีค่าต่ำสุดตัวต่อไป
- เก็บลิ้งของตำแหน่งนั้นไว้
- รวมจุดยอดทั้งสองที่ต่อด้วยลิ้งตัวนี้
- คำนวณค่าเวทของจุดยอดใหม่และของลิ้งต่าง ๆที่เชื่อมต่อกับจุดยอดอันนี้

ขั้นตอนที่ 3 สร้างสแพนนิ่งทรีจากลิ้งที่ได้เก็บเอาไว้แล้ว

ขั้นตอนที่ 4 ทำการตัดสแพนนิ่งทรีของกราฟเป็นจำนวนเท่ากับ regionของภาพที่ต้องการลบหนึ่ง

ขั้นตอนที่ 5 แปลงกราฟกลับเป็นภาพเซกเม้นต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ส่วนวิเคราะห์องค์ประกอบของภาพ(บล็อบ(Blob))

2.4.1 การวิเคราะห์ความต่อเนื่องของจุด

ในการแปลความหมายจากภาพเป็นสิ่งที่สำคัญมากสำหรับหุ่นยนต์ระดับชาญฉลาด(intelligent robot) ถึงแม้ปัจจุบันนี้ได้มีการวิจัยเกี่ยวกับการเข้าใจภาพ 3 มิติ ที่จำเป็นต้องใช้ภาพเชิงเลข (gray scale pictures) ที่มีการแปลงเป็นค่าของระดับความสว่างของแต่ละจุด (pixel) บนภาพ แต่ในการใช้งานส่วนใหญ่ก็ยังคงใช้ภาพเลขฐานสอง (binary image) ซึ่งจะสามาถแปลความหมายของภาพได้ในลักษณะของ 2 มิติเท่านั้น เหตุผลหนึ่งที่ยังคงใช้ภาพเลขฐานสอง เพราะต้องการหน่วยความจำในการจัดเก็บภาพน้อยกว่า รวมถึงโปรแกรมและความเร็วในการแปลความหมายของภาพก็สามารถทำได้ง่ายและเร็วกว่าด้วย

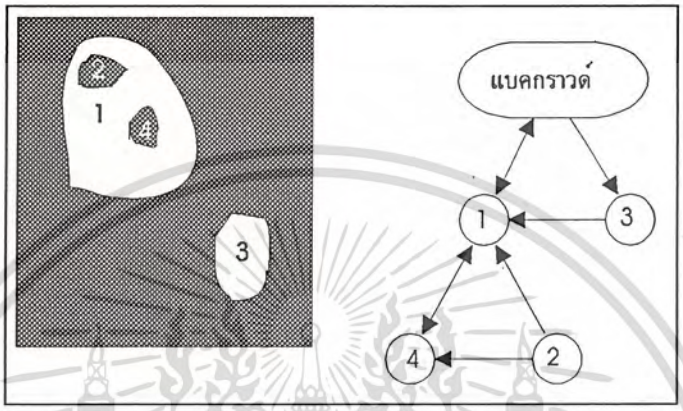
2.4.2 บล็อบ (Blob)

การแปลความหมายของภาพก็คือ การหาวิธีการในการบรรยายลักษณะของภาพให้ได้ ก่อนอื่นจะต้องแบ่งภาพออกเป็นกลุ่มๆ ซึ่งแต่ละกลุ่มของจุดจะมีสีเดียวกัน (ขาวหรือดำ) เราเรียกกลุ่ม เหล่านี้ว่า บล็อบ (blob) ซึ่งบล็อบในภาพอาจจะเป็นได้ทั้งวัตถุ (objects) , ช่องหรือโพรง (holes) หรือพื้นภาพก็ได้ ดังนั้นในหนึ่งภาพจะประกอบขึ้นด้วยบล็อบจำนวนหนึ่ง ซึ่งแน่นอนต้องมีบล็อบอันหนึ่งแทนพื้นภาพเสมอ ถ้าสังเกตให้ดีการบรรยายลักษณะของภาพก็คือ การบรรยายลักษณะของบล็อบนั่นเอง เราเรียกตัวบรรยายนี้ว่าบล็อบดีสคริปเตอร์ (blob descriptor) ซึ่งภายในจะมีข้อมูลทุกๆสิ่งที่ต้องการจะรู้เกี่ยวกับบล็อบ เช่น พื้นที่ (area) จุดศูนย์กลาง (centroid) จำนวนช่องหรือโพรง สี และอื่นๆ ที่เกี่ยวข้องกับรูปร่างของบล็อบ นอกจากนี้ยังมีข้อมูลเกี่ยวกับความสัมพันธ์ของบล็อบนั้นกับบล็อบอื่นๆ ในภาพ โดยบอกเป็นลักษณะของ parent , child และ sibling

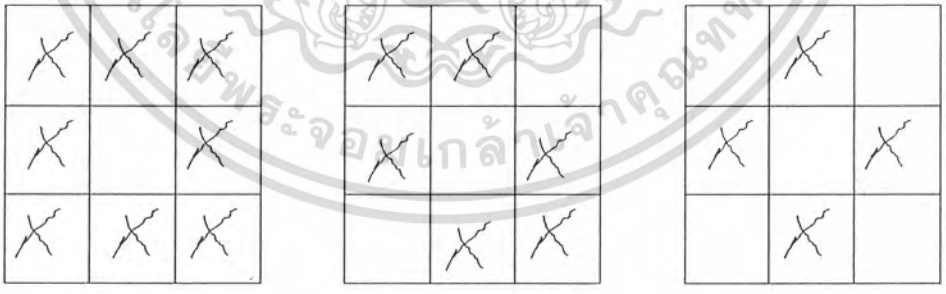
ดูตัวอย่างในรูปที่ 2.4.1 บล็อบ 1 มี 2 child คือ บล็อบ 2 และ 4 หรืออาจกล่าวว่าบล็อบ 2 และ 4 มี parent คือ บล็อบ 1 โดยบล็อบ 2 และ 4 เป็น sibling กัน และจะเห็นอีกว่าบล็อบ 1 กับ 3 เป็น sibling กัน โดยมี parent คือ บล็อบพื้นภาพ

ความสัมพันธ์ระหว่าง child และ parent จะเป็นความสัมพันธ์ในลักษณะของบล็อบหนึ่ง บรรจุอยู่ในอีกบล็อบหนึ่งที่อยู่ชิดกัน ดังนั้นจะไม่ถือว่าบล็อบ 2 และ 4 เป็น child ของ บล็อบพื้นภาพ ถ้าสังเกตสักนิดจากรูปจะเห็นว่าบล็อบที่เป็น child และ parent กันจะมีสีต่างกัน ทำนองตรงกันข้ามถ้าเป็น sibling กันจะมีสีเหมือนกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4.1 ตัวอย่างภาพที่แสดงความสัมพันธ์ตามลำดับชั้นระหว่างบล็อบ



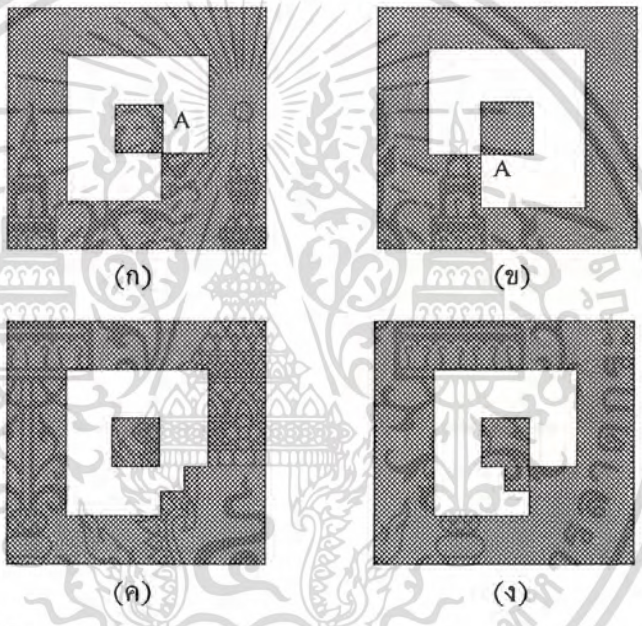
(ก) (ข) (ค)

รูปที่ 2.4.2 (ก) รูปแบบการต่อของจุดแบบ 8 ทาง (ข) รูปแบบการต่อของจุดแบบ 6 ทาง (ค) รูปแบบการต่อของจุดแบบ 4 ทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 ปัญหาการต่อของจุด (connectivity)

ก่อนที่จะเข้าสู่การหาและการสร้างบล็อบ ปัญหาหนึ่งที่ต้องพิจารณาคือปัญหาการต่อของจุด จุดสองจุดที่อยู่ชิดกันจะต่อกันได้ต้องมีสีเดียวกัน และเป็นไปตามรูปแบบของการต่อ ซึ่งตามธรรมดาจะมีอยู่ 2 แบบคือ แบบ 4 ทาง (4-connectivity) และแบบ 8 ทาง (8-connectivity) ดังในรูปที่ 2.4.2 จะเห็นว่าแบบ 4 ทางจุดที่อยู่กึ่งกลางจะต่อกับจุดที่อยู่ชิดกันเพียง 4 ทิศทางเท่านั้นคือ บน , ล่าง , ซ้าย และ ขวา สำหรับ 8 ทางจะเพิ่มจากแบบ 4 ทางมาอีก 4 ทิศ คือ ตามแนวเส้นทะแยงมุม



รูปที่ 2.4.3

(ก) ตัวอย่างภาพที่มีการพบกันของบล็อบที่จุดเดียวกันคือจุด A

(ข) ภาพที่เกิดจากการหมุนภาพในรูป (ก) 90 องศา

(ค), (ง) เป็นภาพที่จะเกิดขึ้นจริงในทางปฏิบัติของภาพ ก

ปัญหาที่จะเกิดขึ้นเนื่องจากการต่อของจุด ยกตัวอย่างเช่นภาพในรูปที่ 2.4.3(ก) ที่มีการพบกันของบล็อบที่จุดเดียวกันคือจุด A ถ้าทุกบล็อบใช้การต่อของจุดแบบ 8 ทาง จากบล็อบวัตถุจะมองสี่เหลี่ยมดำที่อยู่กึ่งกลางไม่ต่อกับพื้นภาพนั่นคือ เป็นช่องหรือโพรงของ วัตถุนั่นเอง แต่ถ้ามองจากบล็อบพื้นภาพสี่เหลี่ยมดำนั้นจุดต่อกับพื้นภาพนั่นคือ ไม่ใช่เป็นช่องหรือโพรงของวัตถุ แต่ถ้าทุกบล็อบใช้การต่อของจุดแบบ 4 ทาง บล็อบวัตถุจะมองสี่เหลี่ยมดำเป็นส่วนเดียวกันกับพื้นภาพซึ่งจะกลับกับการมองจากบล็อบพื้นภาพจะเห็นสี่เหลี่ยมดำเป็นช่องหรือโพรงของวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตัวอย่างนี้เห็นว่า เราไม่สามารถที่จะบรรยายลักษณะของบล็อบในภาพนี้ได้ ทางหนึ่งที่จะแก้ไขปัญหานี้ได้ก็คือ การนำรูปแบบการต่อของจุดทั้งสองรูปแบบมาใช้ในภาพเดียวกันนั่นคือ ให้รูปแบบหนึ่งใช้กับบล็อบที่มีสีๆหนึ่ง และอีกรูปแบบหนึ่งใช้กับบล็อบที่มีสีต่างกันออกไป สำหรับการแก้ปัญหาทางอื่นอาจจะใช้รูปแบบการต่อของจุดแบบ 6 ทางแทน แต่การแก้ปัญหาโดยวิธีนี้ไม่สมบูรณ์เสียทีเดียว เพราะถ้าใช้กับรูปเดิม 4(ก) สี่เหลี่ยมดำจะเป็นส่วนเดียวกับพื้นภาพ แต่ถ้าหมุนภาพนี้ไป 90 องศาตามเข็มนาฬิกาเป็นรูป 4 (ข) จะได้สี่เหลี่ยมสีดำไปเป็นส่วนเดียวกับพื้นภาพแทน ดังนั้นจะเห็นว่าการบรรยายลักษณะภาพอาจเปลี่ยนความหมายจากภาพหนึ่งเป็นอีกภาพหนึ่งได้ แต่ในทางปฏิบัติจริงย่อมจะเกิดคว่าผิดพลาดขึ้นในภาพบ้างดังนั้นโอกาสที่จะเป็นภาพในรูป 2.4.3(ก) และ 2.4.3 (ข) เป็นไปไม่ได้ น่าจะเป็นรูป 2.4.3(ค) หรือ 2.4.3 (ง) รูปใดรูปหนึ่งมากกว่า จึงสามารถตัดปัญหาที่เกิดจากการใช้รูปแบบการต่อจุดแบบ 6 ทางได้ ในรายงานฉบับนี้จะใช้รูปแบบนี้เป็นหลักในการพิจารณาภาพ

Fields in Blob Record

FLINK	sequential	usedby
RLINK	links	connectivity only
PARENT		
CHILD	hierarchical	
SIBLING	links	
AREA		
Sum of I		
Sum of J		
Sum of J2		
Sum of J2		used by
Sum of IJ	location	
PERIMETER	shape,	both
IMIN	and size	
JMIN		algorithms
IMAX		
JMAX		
NHOLES	number of holes	
COLOR		
NUMBER		

รูปที่ 2.4.4 แสดงถึงฟิลด์ต่าง ๆ ในเรคอร์ดของบล็อบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.4 การวิเคราะห์ความต่อเนื่องของจุด (connectivity analysis)

วิธีหนึ่งที่จะหาบล็อบดีสคริปเตอร์ก็คือ ใช้การวิเคราะห์ความต่อเนื่องของจุด ซึ่งก็มีโปรแกรมที่เขียนด้วยภาษาซี ก่อนอื่นต้องมาดูก่อนว่าภายในเรคอร์ดของ บล็อบดีสคริปเตอร์ (ในโปรแกรมคือ อาร์เรย์ B) แสดงในรูปที่ 2.4.4 ประกอบด้วยฟิลด์อะไรบ้าง และจะบอกอะไรกับเราได้บาง จากรูปฟิลด์ FLINK และ RLINK เป็นการเชื่อมโยงระหว่างเรคอร์ดของบล็อบในลักษณะเป็นวงกลมเพื่อสะดวกในการค้นหาเพิ่มหรือลบเรคอร์ดของบล็อบในโปรแกรม แต่ฟิลด์ที่ทำให้ทราบถึงการเชื่อมโยงตามลำดับขั้นก็คือ ฟิลด์ แต่ฟิลด์ที่ทำให้ทราบถึงการเชื่อมโยงตามลำดับชั้นก็คือ ฟิลด์ PARENT , CHILD และ SIBLING สำหรับฟิลด์ที่เหลือจะเป็นฟิลด์ที่บอกเกี่ยวกับรูปร่างลักษณะของบล็อบ เริ่มจากฟิลด์ AREA จะบอกถึงจำนวนจุดภายในบล็อบนั้น ฟิลด์ color เป็นการบอกสีของบล็อบ ฟิลด์ PERIM จะบอกความยาวรอบรูป บล็อบ ฟิลด์ IMIN และ JMIN บอกตำแหน่งทางแนวตั้งและแนวนอนที่น้อยที่สุดของบล็อบตามลำดับ รวมถึงฟิลด์ IMAX และ JMAX ก็จะบอกตำแหน่งทางแนวตั้งและแนวนอนที่มากที่สุดของบล็อบตามลำดับเช่นกัน ฟิลด์ NHOLES บอกถึงจำนวนช่องหรือโพรงภายในบล็อบ ฟิลด์ NUMBER จะบอกว่าเรคอร์ดของบล็อบนั้นมีหมายเลขของบล็อบเป็นอะไร ซึ่งจะเป็นเลขเฉพาะของแต่ละบล็อบ

ก่อนที่จะกล่าวถึงฟิลด์ที่เหลืออีก 5 ฟิลด์ ต้องมาทำความเข้าใจก่อนว่าการสแกนภาพจะเริ่มจากซ้ายไปขวา และบนลงล่าง โดยจะให้มุมบนซ้ายสุดของภาพเป็นตำแหน่ง (0,0) เสมอ ในการสแกนแต่ละครั้งจะพบจุดที่มีสีเดียวกันติดต่อกันและก็มีการสลับสีกันบ้างภายในเส้นเดียวกัน ดังนั้นจึงสามารถเปลี่ยนข้อมูลของภาพนี้ให้อยู่ในรูปของเรคอร์ด run-length (ในโปรแกรมคือ อาร์เรย์ R) ซึ่งประกอบด้วย ฟิลด์ 3 ฟิลด์ คือ ตำแหน่งทางแนวตั้งที่เริ่มต้น ความยาวและหมายเลขบล็อบที่จุดเหล่านั้นอยู่ เพื่อความเข้าใจดียิ่งขึ้นดูรูปที่ 2.4.5 ประกอบ จะเห็นว่าตำแหน่งที่เริ่มต้นของแต่ละเส้นก็คือ ตำแหน่งของ run-length ที่ถูกชี้โดยอาร์เรย์ทางแนวนอน (ในโปรแกรมคือ อาร์เรย์ RP) ทีนี้ย้อนกลับมาดูฟิลด์ในเรคอร์ดของบล็อบที่เหลือคือฟิลด์ SUMI , SUMJ , SUMI2 , SUMJ2 และ SUMIJ ซึ่งใน 1 run-length จะมีการคำนวณตามสมการดังนี้

$$SUMI = (I0+0)+(I1+1)+...+(I0+(L-1))$$

$$= L-1\sum K=0(I0+K)=\sum L-1K=0\ I0+\sum L-1K=0K$$

เนื่องจาก $L-1\sum K=0\ K=L(L-1)/2$

$$\therefore SUMI=LI0+L(L2-1)/2=L(I0+(L-1)/2) \qquad .....(1)$$

$$\therefore SUMI2 = (I0+0)2+(I1+1)2+...+(I0+(L-1)2$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$=L-1\sum K=0(I0+K)2=\sum L-1K=0(I02+2I0K+K2)$

$=L-1\sum K=0I02+2I0\sum L-1K=0K+\sum L-1K=0K2$

เนื่องจาก $L-1\sum K=0K2=L(L-1)(2L-1)/6$

$\therefore SUMI2 = LI02+2I0L(L-1)/2+L(L-1)(2L-1)/6$

$=L(I02+I0(L-1)+(L-1)(2L-1)/6)$ .....(2)

$SUMJ = LJ$ .....(3)

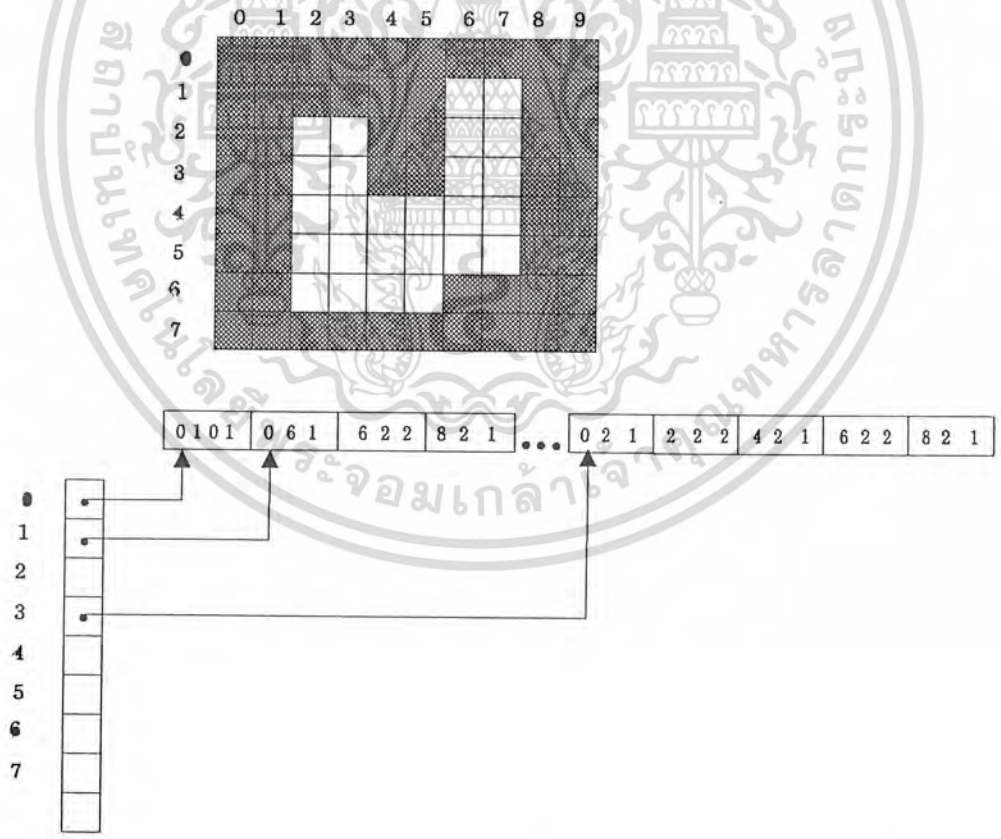
$SUMJ2 = LJ2$ .....(4)

$SUMJ = LJ(I0+(L+1)/2)$ .....(5)

I0 คือ ตำแหน่งจุดเริ่มต้นทางแนวตั้งของ run-length

L คือ ความยาวหรือจำนวนจุดใน run-length

J คือ ตำแหน่งทางแนวนอนของ run-length

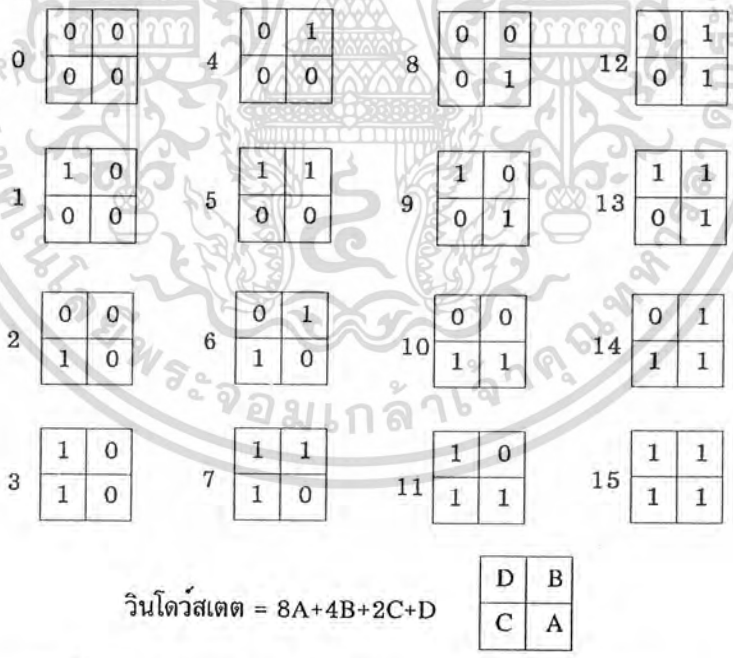


รูปที่ 2.4.5 ตัวอย่างการเก็บข้อมูลของภาพในลักษณะเรคอร์ด ของ run - length

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงาน

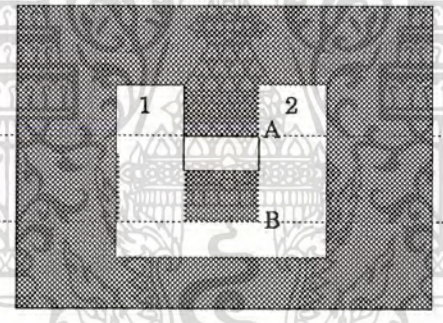
ทีนี้ก็จะมาดูการทำงานของการวิเคราะห์ความต่อเนื่องของจุดกันบ้าง ในที่นี้จะกล่าวแต่พอสังเขปเท่านั้น จะไม่เข้าไปกล่าวถึงตัวแปรในโปรแกรมมากนัก เริ่มจากจะสมมติก่อนว่าในภาพใดๆ จะมีจุดของบล็อบพื้นภาพล้อมรอบภาพอยู่อย่างน้อยมีความหนาเท่ากับ 1 จุดเสมอ เพื่อความสะดวกในการเขียนโปรแกรม ดังนั้นโปรแกรมจะเริ่มทำงานที่เส้นที่ 2 ของภาพเสมอ การทำงานของโปรแกรมจะทำทีละจุดตามการสแกนภาพ แต่ต้องอาศัยข้อมูลของจุดก่อนในเส้นเดียวกันและอีก 2 จุด ในตำแหน่งแนวตั้งเดียวกัน แต่เป็นของเส้นก่อน เพื่อใช้ในการพิจารณาการเปลี่ยนแปลงของสีจากดำเป็นขาวหรือขาวเป็นดำในแนวนอนทั้งสองเส้น ดังนั้นเราจะได้เป็นวินโดว์ (window) ขนาด 2*2 ซึ่งสามารถแบ่งออกเป็นได้ 16 สถานะ ตามรูปที่ 2.4.6 จุดที่อยู่มุมล่างขวาเป็นจุดที่กำลังพิจารณา ค่าของวินโดว์ในสถานะต่างๆ จะเป็นตามสมการในรูปจาก 16 สถานะ ถ้าพิจารณาว่าการเปลี่ยนแปลงของสีจากขาวเป็นดำ เป็นกรณีเดียวกับดำเป็นขาวก็จะเหลือเพียง 8 กรณี แต่ที่กรณี 0,15 และ 5,10 ไม่มีการเปลี่ยนแปลงของสีในทางแนวนอนจึงสามารถตัดออกไปได้เหลือเพียง 6 กรณีเท่านั้น ในแต่ละกรณีจะมีความเป็นไปได้ของภาพต่างๆ กัน ทำให้การทำงานในโปรแกรมต่างกันออกไป เช่น กรณี 7,8 จะเป็นจุดเริ่มต้นของบล็อบใหม่ จึงต้องมีการตั้งค่าตัวแปรบางตัวใหม่เพื่อเริ่มบันทึกบล็อบใหม่ขึ้น



รูปที่ 2.4.6 แสดงค่าของวินโดว์ในสถานะต่างๆ 16 สถานะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อการสแกนมีไปถึงจุดสิ้นสุดของสีหนึ่งๆ ก็คือจุดสิ้นสุดของ run-length หนึ่งๆ เช่นกัน ข้อมูลในเรคอร์ดของ run-length ก็จะถูกปรับปรุงให้ถูกต้องรวมทั้งข้อมูลเกี่ยวกับรูปร่างของบล็อบ เช่น AREA ก็จะถูกสะสมเป็นสถิติที่ถูกต้องขึ้นในบล็อบที่ฟิลด์ NUMBER ของ run-length ชี้อยู่ สำหรับการพิจารณาว่า run-length นี้เป็นส่วนหนึ่งของบล็อบใดก็ต้องเป็นไปตามรูปแบบการต่อของจุดแบบ 6 ทาง โดยใช้ข้อมูลของจุดในเส้นก่อนเป็นตัวร่วมพิจารณาด้วย ในการพิจารณาภาพบางภาพ เช่น ภาพในรูปที่ 2.4.7 ถ้าการสแกนยังไม่ถึงจุด A ในบล็อบ 1 และ 2 ของภาพก็ยังคงเป็นคนละส่วนกัน จนกระทั่งการสแกนมาถึงจุด A ข้อมูลทางสถิติเกี่ยวกับรูปร่างของบล็อบ 2 ก็จะถูกนำมารวมกันกับบล็อบ 1 และลบบล็อบ 2 ทิ้งเมื่อมีการสแกนมาถึงจุด B บล็อบของสี่เหลี่ยมดำตรงกลางก็จะสมบูรณ์ ซึ่งดูได้จากเส้นรวมบล็อบต่อกันครบรอบพอดี ข้อมูลเกี่ยวกับการเชื่อมโยงลำดับชั้นของบล็อบก็คือฟิลด์ PARENT,CHILD และ SIBLING ก็จะถูกบันทึกขึ้นที่จุด B นี้



รูปที่ 2.4.7 ตัวอย่างภาพที่ใช้อธิบายวิธีการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 ส่วนวิเคราะห์ความหมายของข้อมูลภาพ

โปรแกรมในส่วนนี้จะทำหน้าที่วิเคราะห์ข้อมูลองค์ประกอบของบล็อบต่าง ๆที่ได้จาก ส่วนที่ 2.4 โดยในการวิเคราะห์หาความหมายของภาพนั้น ดังที่ได้กล่าวแล้วในตอนต้นว่าเราต้องการ ให้โปรแกรมของเราสามารถยืดหยุ่นต่อการแก้ปัญหาในหลาย ๆรูปแบบ ซึ่งไม่ซับซ้อนจนเกินไปนัก

ฉะนั้นการทำงานในส่วนนี้เราจึงเลือกใช้โครงข่ายประสาทเทียมหรือนิวรอลเน็ตเวอร์ก (Neural Network) ซึ่งเป็นโปรแกรมในรูปแบบหนึ่งซึ่งสามารถที่จะทำการนำไปประยุกค์ใช้ในการจดจำและรับรู้ความหมายในแง่ต่าง ๆของภาพได้โดยไม่ต้องทำการแก้ไขตัวโปรแกรมแต่อย่างใด

2.5.1 โครงข่ายประสาทเทียม(ARTIFICIAL INTELLIGENCE NETWORK)

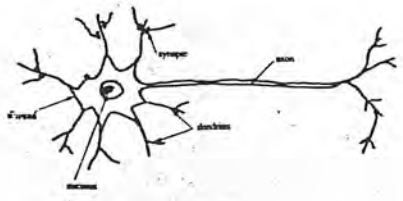
ความพยายามที่จะสร้างเครื่องคำนวณที่สมบูรณ์แบบที่มีความสามารถเหมือนสมองของคนได้มีมานานนับศตวรรษแล้ว เครื่องคำนวณที่สมบูรณ์แบบจะต้องสามารถคิดเองได้และโปรแกรมได้ด้วย

เครื่องคำนวณที่เรารู้จักกันดีก็คือเครื่องคอมพิวเตอร์ หลักการทำงานของเครื่องคอมพิวเตอร์ที่ใช้กันในปัจจุบันตั้งอยู่บนพื้นฐานของคณิตศาสตร์แบบบูลลีน ซึ่งไม่ได้มีส่วนเกี่ยวข้องกับหลักการทำงานของเซลล์ประสาท แต่มันก็สามารถทำให้คอมพิวเตอร์มีประสิทธิภาพในด้านการทำงานต่าง ๆที่ต้องการ

นิวรอลเน็ตเวอร์กนั้น เริ่มต้นมาจากการศึกษาระบบเซลประสาทของมนุษย์เพื่อที่จะเข้าใจถึงการทำงานของกลุ่มเซลล์ประสาทในสมอง และเพื่อที่จะ เอาผลการศึกษาการทำงานของกลุ่มเซลล์ประสาทมาอธิบายสภาวะของโรคทางสมองเช่น ความจำเสื่อมได้ จากนั้นจึงได้ถูกพัฒนาขึ้นเป็นแบบจำลองของเซลล์ประสาทในเชิงคณิตศาสตร์ และถูกนำไปประยุกต์ใช้งานต่อมา

2.5.2 ส่วนประกอบของเซลล์ประสาท

ในธรรมชาติเซลล์ประสาทของสิ่งมีชีวิตมีหลายประเภทแล้วแต่หน้าที่ของมัน เซลล์ประสาทในตัวของคนเราก็เช่นกันมีอยู่หลายประเภทตามแต่ตำแหน่งและหน้าที่ของมัน เช่น เซลล์ลิ้นและจมูก เป็นต้น โครงสร้างและส่วนประกอบของเซลล์ประสาทโดยทั่วไปสามารถแสดงได้ดังในรูปที่ 2.5.1



รูปที่ 2.5.1 โครงสร้างและส่วนประกอบของเซลล์ประสาททั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เซลล์ประสาทประกอบด้วยส่วนใหญ่ 4 ส่วนคือ ตัวเซลล์ประสาท ซึ่งมีนิวเคลียร์อยู่ตรงกลาง รอบๆตัว เซลล์ประสาท มีสิ่งที่ยื่นออกไปเพื่อส่งและรับ สัญญาณจากเซลล์ประสาทอื่นสิ่งดังกล่าวเรียกว่า axon ที่ปลายกิ่งจะแตกออกเป็นก้าน ย่อยๆ เรียกว่า dendrite รอยต่อ ระหว่างก้านของเซลล์ประสาทที่ต่างกันเรียกว่า synapse ซึ่ง สามารถเปลี่ยนค่าความต้านทานได้ตามสัญญาณที่ส่งระหว่างกันของเซลประสาท การส่งสัญญาณระหว่างเซลล์ประสาท ทำโดยการถ่ายเทสารประกอบของโซเดียมและโพแทสเซียม

Hodgkin และ Huxley ซึ่งได้รับรางวัลโนเบลทางชีววิทยาได้ค้นพบว่าการไหลของสารประกอบโซเดียมและต่างศักย์ระหว่างสารที่อยู่ใน axon และสารที่อยู่นอก axon ค่าความต่างศักย์จะอยู่ระหว่าง 50 ถึง 70 มิลลิโวลต์ จากผลการศึกษาดังกล่าวทำให้เราสามารถจำลองการทำงานเซลล์ประสาทโดยอาศัยวงจรอิเล็กทรอนิกส์ได้

2.5.3 ข้อแตกต่างระหว่างโครงข่ายประสาทเทียมและปัญญาประดิษฐ์

ทั้งโครงข่ายประสาทเทียมและปัญญาประดิษฐ์(Artificial Intelligence) ถูกคิดค้นขึ้นมาเพื่อเลียนแบบความฉลาดและการทำงานของสมองคนหรือสมองของสิ่งมีชีวิต ข้อแตกต่างๆที่เห็นได้ชัดระหว่างสิ่งประดิษฐ์ทั้ง 2 ก็คือ วิธีการแก้ปัญหาและลักษณะการทำงาน โดยสามารถสรุปข้อแตกต่างที่เห็นได้ชัดๆ ดังต่อไปนี้

โครงข่ายประสาทเทียม	ปัญญาประดิษฐ์
1.แก้ปัญหาโดยอาศัยสมการคณิตศาสตร์มีการ คำนวณอย่างมาก	1.แก้ปัญหาโดยการจัดการกับกลุ่มของตัวอักษรไม่มีการคำนวณมาก
2. มีการเรียนรู้ได้จากตัวอย่างข้อมูลใหม่	2.ไม่มีการเรียนรู้ อาศัยขั้นตอนที่แน่นอนในเชิงalgorithmic ในการแก้ปัญหา
3.สามารถปรับใช้ได้กับหลายปัญหาโดยไม่ต้องปรับโครงสร้างของโครงข่าย	3. ปรับไม่ได้ แต่ละปัญหาต้องใช้คนละวิธีการ
4.วิธีแก้ปัญหาไม่เป็นแบบ NP	4. วิธีแก้ปัญหาเป็นแบบ Hueristic เพราะปัญหาส่วนใหญ่เป็น NP
5.สามารถสร้างวงจรไฟฟ้าให้ทำงานแบบขนาน ได้	5. สร้างเป็นวงจรจริงยาก

ตารางที่ 2.5.1 ข้อแตกต่างระหว่างโครงข่ายประสาทเทียมและปัญญาประดิษฐ์

2.5.4 แบบจำลองทางคณิตศาสตร์ของนิวรอลเน็ตเวร์ค

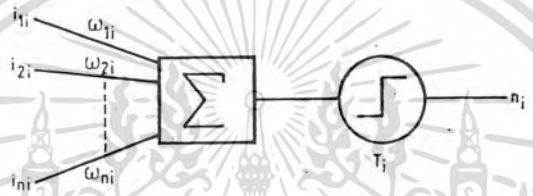
จากโครงสร้างแบบง่ายของเซลล์ประสาทที่แสดงในรูปที่ 2.5.1 ในปี ค.ศ.1943 McCulloch และ Pitts ได้เสนอแบบจำลองเชิงคณิตศาสตร์ของเซลล์ประสาทดังกล่าว โครงสร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของแบบจำลองนี้แสดงในรูปที่ 2.5.2

สัญญาณที่ส่งออกมาจากเซลล์ประสาทในรูปที่ 2.5.2 คือ ni สามารถเขียนสมการคณิตศาสตร์ของ n ที่เวลา t+1 ได้ดังนี้

$$ni(t+1) = f(\sum_{j} \omega_{ji} n_j(t) - T_i)$$



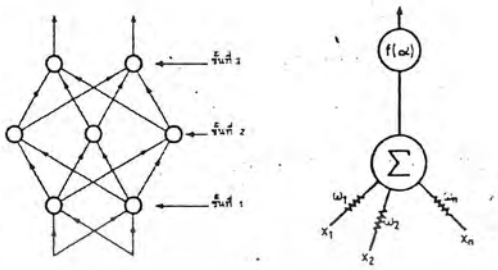
รูปที่ 2.5.2แบบจำลองเชิงคณิตศาสตร์ของเซลประสาท ซึ่งเสนอโดย McCulloch และ Pits

เมื่อ ω_{ji} คือ ค่าความต้านทานของ synapse ที่ j ของเซลล์ประสาทตัวที่ i , ค่า n_j (t) คือ ค่าเปรียบเทียบ ซึ่งเรียกว่า threshold ส่วน f()คือฟังก์ชันแบบขั้นกระได ฟังก์ชัน f ถูกนิยามดังต่อไปนี้

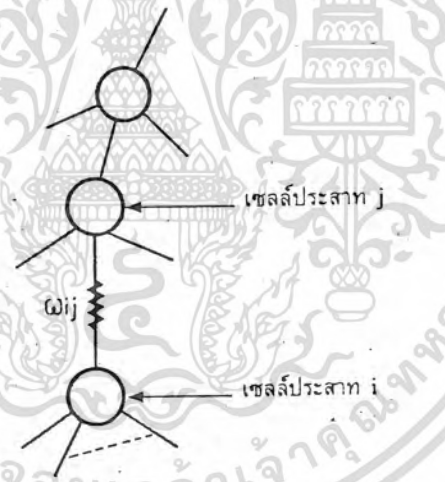
$$f(x) = \begin{cases} 1 & \text{ถ้า } x >= 0 \\ 0 & \text{ถ้า } x < 0 \end{cases}$$

ค่าของ X ในที่นี้ก็คือ $(\sum_{j} \omega_{ji} n_j(t) - T_i)$ นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5.3 ตัวอย่างการจับกลุ่มของเซลล์ประสาท 2.5.3 ชั้น ของโครงข่ายประสาทเทียมที่มีการเรียนรู้แบบแพร่กลับ และสัญญาณเข้าออกของแต่ละเซลล์



รูปที่ 2.5.4 ความหมายของ ω ji

เซลล์ประสาทเรียนรู้โดยการเปลี่ยนค่าของ ω ji ดังนั้นสิ่งที่สำคัญมากที่สุดของการสร้างแบบจำลองของเซลล์ประสาทนอกจากแบบจำลองเชิงคณิตศาสตร์แล้ว เราจำเป็นต้องกำหนดวิธีการที่จะเปลี่ยนค่าของ ωji ตามปัญหาต่าง ๆได้ด้วยเพื่อให้เซลล์ประสาทส่งสัญญาณออกมาตามต้องการจากสมการข้างต้น

2.5.5 Back Propagation Neural Network

เทคนิคของการเรียนรู้ของโครงข่ายประสาทเทียมมีอยู่หลายวิธีด้วยกัน วิธีที่นิยม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และแบบหลากหลายมากที่สุดก็คือ วิธีการเรียนรู้หรือการเปลี่ยนค่าของ ω ji โดยใช้วิธีที่เรียกว่า การแพร่กลับ (Back Propagation) หลักการเรียนรู้โดยวิธีการแพร่กลับพัฒนามาจากสมการพื้นฐานของการเรียนรู้แบบ McCullock และ Pitts ข้างต้น โดยการพิจารณา

ฟังก์ชันของความคลาดเคลื่อนระหว่างค่าของเป้าหมายที่ต้องการ และค่าของสัญญาณที่ส่งออกมาจากตัว neural เอง นอกจากนั้นวิธีการเรียนรู้โดยวิธีแพร่กลับยังพิจารณาค่าสัญญาณที่ส่งออกมาจากตัว neural ในขอบข่ายของค่าจริงด้วย (ไม่ใช่แค่ 0 กับ 1 หรือ +1 กับ -1 เท่านั้น)

โครงข่ายประสาทเทียมที่มีการเรียนรู้โดยวิธีแพร่กลับนี้จะประกอบไปด้วยกลุ่มของเซลล์ประสาทจัดตัวกันเป็นชั้น ๆดังตัวอย่างที่แสดงในรูปที่ 2.5.3 ในรูปนี้ประกอบไปด้วยเซลล์ประสาทจับกลุ่มกันทั้งหมด 3 ชั้น ชั้นที่ 1 ประกอบด้วยเซลล์ประสาท 2 เซลล์ ชั้นที่ 2 ประกอบด้วยเซลล์ประสาท 3 เซลล์ และชั้นที่ 3 ว่า ชั้นนอกสุด ส่วนชั้นอื่นที่ไม่ใช่ชั้นนอกสุดจะเรียกว่า ชั้นที่ซ่อนอยู่

วิธีการเรียนแบบแพร่กลับเขียนเป็นขั้นตอนได้ดังต่อไปนี้ สมมติแต่ละเซลล์ประสาทจะมีขาให้สัญญาณเข้า Xi ทั้งหมด n ขา (ในแต่ละชั้นค่าของ n ไม่เท่ากัน) แต่ละขาของสัญญาณเข้าจะมีค่าของความต้านทาน ωi ค่าของสัญญาณออกของแต่ละเซลล์คำนวณได้จากสมการต่อไปนี้

$$f(\alpha) = \frac{1}{1 + e^{-(\alpha-\theta)}}$$

เมื่อ e คือ ค่าฐานของ log ธรรมชาติ ซึ่งมีค่าประมาณ 2.7182818

α คือ $\Sigma\omega_i X_i$

θ คือ ค่าคงที่

1. กำหนดค่าความต้านทาน ω_i และ θ ของแต่ละเซลล์อย่างสุ่ม

2. ป้อนเวคเตอร์ (X1,X2,...,Xn) และสัญญาณที่ต้องการให้ปรากฏที่ชั้นนอกสุด(d1,d2,...Xn)

3. จากเวคเตอร์ (X1,X2,...,Xn) คำนวณหาค่าสัญญาณที่ส่งออกจากเซลล์แต่ละเซลล์ที่ชั้นนอกสุด โดยไล่ไปทีละชั้น จากชั้นที่ 1 การคำนวณทำโดยใช้สูตร f(α) ข้างต้น ให้ (Y1,Y2,...,Yn) เป็นค่าของสัญญาณที่ได้จากการคำนวณเวคเตอร์ (d1,d2,...,dm) จะแตกต่างจากเวคเตอร์ (Y1,Y2,...,Yn)

4. ปรับค่าของความต้านทานของแต่ละเซลล์ โดยเริ่มจากชั้นนอกสุดไล่ไปจนถึงชั้นที่ 1 ค่าความต้านทานใหม่ปรับได้โดยสมการต่อไปนี้

$$\omega_{ji}(t+1) = \omega_{ji}(t) + \eta\delta_j x'_j$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ ω_{ji} คือ ความต้านทานระหว่างเซลล์ประสาท i และ j ดังแสดงในรูปที่ 2.5.4

η คือ ค่าคงที่ที่เรากำหนดเอง ค่าของ η เรียกว่าอัตราการเรียนรู้ โดยทั่วไป η จะมีค่าน้อยว่า 1

j หาได้โดย

ก.) ถ้าเซลล์ j เป็นเซลล์ชั้นนอกสุด ค่าของ j จะเท่ากับ

$$\delta_j = y_j(1-y_j)(d_j-y_j)$$

ข.) ถ้าเซลล์ j เป็นเซลล์ในชั้นที่ซ่อนอยู่ ค่าของ δ_j จะเท่ากับ

$$\delta_j = x'_j(1-x'_j)\Sigma\delta_k\omega_{jk}$$

เมื่อ x'_j คือ ค่าสัญญาณขาเข้าของเซลล์

jk คือ จำนวนเซลล์ทั้งหมดบนชั้นที่อยู่ถัดจากชั้นของเซลล์ j ขึ้นไป

ทำซ้ำซ้อนตอนที่ 2 ถึง 5 จนกว่าค่าความแตกต่าง E มีค่าน้อยกว่า ΔE กำหนดดังนี้

$$E = \frac{1}{2}\Sigma(di - yi)^2_i$$

หลังจากเราสอนโครงข่ายประสาทให้เรียนรู้แล้ว ต่อไปจะนำมันมาใช้งานโดยการป้อนเวคเตอร์เข้าไป แล้วคำนวณหาค่าสัญญาณที่ออกจากแต่ละเซลล์ประสาทโดยใช้สมการข้างต้นคือ

$$f(\alpha) = \frac{1}{1 + e^{-(\alpha-\theta)}}$$

โดยไล่ไปทีละเซลล์ในแต่ละชั้นของโครงข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 ส่วนเชื่อมต่อกับตัวหุ่นยนต์

การทำงานในส่วนนี้นั้นเป็นการส่งผ่านข้อมูลธรรมดาระหว่างเครื่องควบคุมกับตัวหุ่นยนต์ โดยสัญญาณที่ส่งผ่านนี้จะมีทั้งสัญญาณควบคุมและสัญญาณข้อมูลต่าง ๆ รวมทั้งสัญญานภาพ ซึ่งข้อมูลที่สื่อสารกันนี้จะถูกส่งผ่านเป็นคลื่นสัญญาณความถี่ เพื่อให้ตัวหุ่นยนต์สามารถเคลื่อนที่ได้โดยปราศจากสายสัญญาณ ซึ่งจะมีส่วนรับ/ส่งข้อมูลด้วยคลื่นนี้ติดอยู่ที่ทั้ง 2 ฝ่ายคือที่เครื่องคอมพิวเตอร์ควบคุมและที่ตัวหุ่นยนต์ เพื่อคอยทำหน้าที่แปลงสัญญาณจากคลื่นเป็นไฟฟ้าและสัญญานไฟฟ้าเป็นคลื่นสำหรับใช้ในการสื่อสาร ซึ่งนอกจากการรับข้อมูลภาพมาประมวลผลแล้ว ยังต้องทำหน้าที่เป็นตัวกลางในการสื่อสารระหว่างผู้ควบคุมกับตัวหุ่นยนต์ ทั้งในการสั่งงานและการรับข้อมูลจากเซอร์ต่าง ๆซึ่งติดตั้งอยู่บนตัวหุ่นยนต์ โดยสามารถทำการสั่งงานให้หุ่นยนต์เดินหน้า, ถอยหลัง, เลี้ยวซ้าย,เลี้ยว-ขวา ได้ ส่วนในด้างของการรับข้อมูลนั้น สามารถรับระยะห่างจากวัตถุใน 3 ทิศทาง(ด้านหน้า และด้านซ้าย-ขวา)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การคำนวณและการสร้าง

สำหรับโครงงานนี้ ผลงานที่ได้มีจะมีลักษณะเป็น โปรแกรมระบบ (Software) ทั้งสิ้น โดยในการทำงานของโปรแกรมระบบนี้นั้นจะประกอบไปด้วยส่วนต่าง ๆ ดังที่ได้เสนอไปแล้วในส่วนของ ทฤษฎีและหลักการ ดังนั้นในส่วนของบทนี้จะได้เสนอถึงการออกแบบและการสร้างโปรแกรมที่ทำหน้าที่ การรับภาพและแสดงผล ซึ่งจะส่งข้อมูลที่เกิดขึ้นไปยังส่วนถัดไปคือ ส่วนวิเคราะห์แยกองค์ประกอบของภาพ ส่วนวิเคราะห์บล็อบภาพ และส่วนวิเคราะห์ความหมายของข้อมูลภาพตามลำดับ

3.1 โปรแกรมรับภาพ

โปรแกรมนำส่วนนี้สร้างขึ้นเป็นรูทีนย์ย่อยของโปรแกรมสำหรับการติดต่อสั่งงาน โดยเฉพาะซึ่งประกอบไปด้วย ส่วยสำคัญหลัก ๆคือ

- ส่วน Initial Card
- ส่วนการสั่งงานถ่ายภาพสี 64*64*64 สี
- ส่วนการสั่งงานถ่ายภาพขาวดำ 64 ระดับเทา
- ส่วนอ่านโอนถ่ายข้อมูลภาพจากการ์ด
- รูทีนย่อยสำหรับถ่ายข้อมูลภาพจากการ์ดสู่จอแสดงผล

ฯลฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 โปรแกรมแสดงผลภาพ

โปรแกรมส่วนนี้ออกเป็นให้ทำงานแบบ Object Oriented โดยตัวออปเจคนี้จะทำหน้าที่เป็นตัววัตถุคือการ์ด โปรแกรมนี้จะทำการเรียกใช้รูทีนย่อยจากโปรแกรมอีก 2 ส่วนคือ โปรแกรมรับภาพ และโปรแกรมติดต่อกับ VGA card โดยออปเจคตัวนี้เราสามารถที่จะสั่งให้มันทำงานต่าง ๆเกี่ยวกับระบบรับภาพได้ นอกจากนี้ออปเจคตัวนี้ยังทำการ hook Timer Interrupt เข้าสู่ตัวมันทำให้มันสามารถทำการสแกนข้อมูลจากการ์ด มาสู่จอภาพได้โดยอัตโนมัติ ซึ่งมีต่าง ๆที่สำคัญดังนี้

- สั่งเริ่มสแกนภาพจากการ์ดสู่จอภาพ
- สั่งหยุดสแกน
- สั่งเก็บภาพลงหน่วยความจำที่ระบุ
- ส่งตรวจสอบสถานะของการจัดเก็บภาพว่าเรียบร้อยหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ส่วนวิเคราะห์แยกองค์ประกอบของภาพ

โปรแกรมส่วนนี้จะประกอบไปด้วยฟังก์ชันการคำนวนทาง image processing หลาย ๆชนิด โดยโปรแกรมซึ่งทำหน้าที่นี้จะมีอยู่ด้วยกัน 2 ชุด ชุดหนึ่งใช้การสังงานแบบ manual เพื่อใช้ในการทดลองประมวลผลภาพด้วยวิธีการต่าง ๆ โปรแกรมชุดนี้จะมีวิธีการประมวลผลหลาย ๆ แบบให้เลือกใช้ ส่วนโปรแกรมอีกชุดหนึ่งจะถูกสร้างเป็นออปเจคที่ทำหน้าที่จะมวลผลภาพซึ่งจะทำงานโดยอัตโนมัติตามลำดับที่ระบุ โดยลำดับของการประมวลผลนั้นจะได้มาจากการใช้โปรแกรมในชุดแรกทดลองปรับภาพด้วยวิธีต่าง ๆ

รูทีนส่วนต่าง ๆที่สำคัญของโปรแกรมส่วนนี้มีดังนี้

- ส่วนอ่านข้อมูลภาพจากไฟล์เข้ามาประมวลผล
- ส่วนแสดงภาพขณะถูกประมวลผลด้วยกรรมวิธีต่าง ๆ
- ส่วนทำ LowPass filter ภาพ
- ส่วนทำ Hight pass Filter
- ส่วนทำกระบวนการ Sum Square Delta
- ส่วนทำกระบวนการ ZeroCrossing
- ส่วนโชว์ Histogram
- ส่วนทำ Histogram Equalization
- ส่วนการทำ threshold
- ส่วนของการทำ Region clustering โดยใช้วิธี SST
- ส่วนของการ Clean noise

ในส่วนของการหาขอบภาพนี้ทางกลุ่มของข้าพเจ้าได้เลือกใช้การประมวลแบบ special domain ที่อาศัยหลักการของ Matrix ในการประมวลผล

การหาขอบภาพโดยใช้ Matrix นั้นเราทำการวิเคราะห์ค่าของจุดใด ๆโดยพิจารณาจากจุดรอบ ๆมัน ซึ่งในที่นี้เราจะใช้เมตริกซ์แบบ 3*3 ในการประมวลผล คือเมื่อจะประมวลผลจุดตรงกลางก็จะนำค่า 9 ค่า คือ 8 จุดรอบ ๆ และตัวในเองดังแสดงในรูป 3.3.1 มาคำนวนเพื่อหาค่าผลลัพดังนี้

8	7	8
7	7	7
7	4	5

รูปที่ 3.3.1 ภาพแสดงเมตริกที่ใช้ในการประมวลผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการทำการ filter ต่างๆ นั้นเราจะนำเมตริกของของ filter นั้นๆ คูณเข้าไปกับเมตริกของข้อมูล โดยการคูณค่าที่อยู่ในตำแหน่งเดียวกันของเมตริกเข้าด้วยกัน (เช่น ตำแหน่ง [1,1] กับ [1,1] ตำแหน่ง [1,2] กับ [1,2] . . . . . .ตำแหน่ง [3,3] กับ [3,3]) นำค่าผลคูณที่ได้ทั้งหมดมาบวกัน แล้วหาค่าสัมบูรณ์ของผลลัพธ์ จากนั้นหารด้วย 9 ซึ่งเป็นจำนวนข้อมูล ก็จะได้ค่าใหม่ของจุดที่อยู่ในตำแหน่งกลางมา จะต้องจำอย่างๆนี้กับจุดทุกๆจุดบนภาพ ก็จะได้จำนวนจุดใหม่เท่ากับจำนวนจุดภาพเดิม ซึ่งก็คือภาพที่ผ่าน filter แล้วนั่นเอง

โดยสำหรับการทำ low pass filter นั้นเมตริกที่จะนำมาคูณจะเป็นเมตริก

1	1	1
1	1	1
1	1	1

และเมตริก สำหรับ High pass filter

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

เมตริกของ Gradient operator in X direction

–1	-2	-1
0	0	0
1	2	1

เมตริกของ Gradient operator in Y direction

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และสำหรับการทำ Gradient Operator in XY direction ก็นำค่าที่ได้จากการคูณด้วยเมตริก X และค่าจากการคูณเมตริก Y มา บวกกัน

Sum Square Delta

ฟังกชัน Sum Square Delta นี้ จะมีการทำงานเป็นแบบ special domain ขนาด 3x3 เช่นกัน แต่จะมีฟังก์ชันการคำนวนที่แตกต่างออกไป ซึ่งขัดตอนการทำงานจะเป็นรูปแบบดังนี้

ขั้นตอนที่ 1 แยกพิจารณา Matrix ย่อย ๆนาด 3 x 3

ขั้นตอนที่ 2 นำ Matrix ขนาด 3 x 3 มาผ่านขั้นตอนดังต่อไปนี้จนครบทั้งภาพ

2.1 หาค่าเฉลี่ยของ Matrix ย่อยนั้น ๆ (ค่าเฉลี่ยของจุดทั้ง 9 จุด)

2.2 นำค่าเฉลี่ยที่ได้นี้มาลบออกจากจุดทั้ง 9 จุดบน Matrix ย่อย

2.3 นำค่าตัวเลขที่เกิดขึ้นนี้มายกกำลังสอง แล้วนำไปบวกกันกับจุดอื่น ๆทุก ๆจุดบน Matrix นั้น

ขั้นตอนที่ 3 หากค่า sum square delta รอบ ๆจุดใดมีค่ามากกว่าค่า threshold ที่ตั้งไว้ จะมีค่าเป็น 255 และหากมีค่าน้อยกว่าจุดนั้นมีค่าเป็น 0

8	7	8
7	7	7
7	4	5

ค่า เฉลี่ย = (8+7+8+7+7+7+7+4+5) / 9 = 6 (ปัดเศษทิ้ง)

นำค่า 6 ที่ไดนั้นมาลบออกจากตัวเลขแต่ละช่อง หลังจากนั้นนำค่าที่ได้มายกกำลังสอง แล้วบวกกัน จะได้ผลเท่ากับ =

$$(8-6)^2+(7-6)^2+(8-6)^2+$$
$$(7-6)^2+(7-6)^2+(7-6)^2+$$
$$(7-6)^2+(4-6)^2+(5-6)^2+$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

histrogram equalize

เป็นขบวนการการกระจายความเข้มสีของจุด จากที่มีความเข้มสีใกล้ ๆกันให้ห่างออกจากกัน มากยิ่งขึ้น ซึ่งจะทำให้ภาพที่มีแสงน้อยสามารถมองเห็นได้ชัดเจนยิ่งขึ้น โดยความห่างระหว่างสีแต่ละสีนั้นจะขึ้นอยู่กับปริมาณของจุดที่มีสีนั้น โดยความห่างระหว่างสีแต่ละสีนั้น ขึ้นกับปริมาณของจุดที่มีสีนั้น หากค่าสีใดมีปริมาณจุดของค่า สีนั้นมาก ค่าสีนั้นจะถูกปรับให้แยกออกจากค่าสีอื่นมากขึ้นเพื่อให้เห็นความแตกต่างได้เด่นชัด ขึ้น และค่าสีที่ไม่คอยได้ใช้จะถูกตัดออกไประดับของสีจะกระจายตั้งแต่ค่า 0 ถึง 225 โดยค่า

Clean noise

ส่วนของการ clean noise นี้เราใช้วิธีฐานนิยมรอบ ๆจุด โดยใน 9 จุดที่เราพิจารณานั้น หากมีจำนวนจุดที่มีค่า 255 มากกว่าจำนวนจุดที่มีค่าเป็น 0 แล้วเราจะให้จุดกลาง ของบริเวณจุดที่พิจารณานั้นมีค่าเป็น 255 แต่หากมีจำนวนจุดที่มีค่าเป็น 0 มากกว่าจำนวนจุดที่มีค่าเป็น 255 ให้จุด ๆนั้นมีค่าเป็น 0

3.4 ส่วนวิเคราะห์บล็อบภาพ

ในส่วนของการวิเคราห์บล็อบภาพนี้ ก็จะมีโปรแกรมเป็น 2 ชุดเช่นเดียวกับส่วนที่แล้ว คือมีส่วนหนึ่งซึ่งเป็นออปเจคซึ่งจะใช้ในการประมวลผลแบบออนไลน์ และอีกชุดเป็นภาษาซีธรรมดาซึ่งใช้สำหรับการประมวลผลข้อมูลแบบ Batch เช่นเดียวกับส่วนของการวิเคราะห์แยกองค์ประกอบของภาพ ทฤษฎีและวิธีการในส่วนของการหาบล็อบนั้นได้อธิบายไว้แล้วในส่วนของทฤษฎีและหลักการ จึงจะไม่ได้กล่าวถึงอีก แต่ในการสร้างโปรแกรมนั้น เรายังได้เพิ่มฟังก์ชั่นเพื่อช่วยให้การประมวลผลภาพให้สมบูรณ์ยิ่งขึ้นโดยกำจัดค่า error ส่วนใหญ่ที่อาจเกิดขึ้นได้ในระหว่างประมวลผลเช่น จำนวนบล็อบที่มากเกินไป โดยจะมีฟังก์ชันในการคัดเลือกบล็อบที่มีขนาดพื้นที่น้อยที่สุดลบทิ้งไปเพื่อเหลือที่สำหรับ สร้างบล็อบใหม่ และจากนั้นยังมีฟังก์ชันในการจัดเรียงลำดับของส่วนของบล็อบต่าง ๆโดยตรวจดูความเป็น parent หรือ child blob ของบบล็อบนั้น ๆ เพื่อให้บล็อบมีรูปร่างของข้อมูลซึ่งง่ายต่อการประมวลผลยิ่งขึ้น บล็อบที่ได้ จากในขั้นตอนนี้จะถูกส่งไปยังขั้นตอนต่อไป คืออาจถูกจัดเก็บเป็นไฟล์สำหรับส่งให้ neural network ในการเก็บตัวอย่างเพื่อใช้ในการ train network หรือ ถูกส่งไปให้กับ neural network เพื่อทดสอบความหมายของข้อมูลภาพต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 ส่วนวิเคราะห์ความหมายของข้อมูลภาพ

สำหรับโครงข่ายซึ่งเราใช้ในการทดลองครั้งนี้นั้น เราใช้เป็นโครงข่าย 2 คือแบบแรกป็น Neural Network แบบชั้นเดียว 2 ตัว และแบบที่ 2 เป็น Neural Network แบบ 2 ชั้นอีก 2 ตัว ซึ่งมีขนาดอินพุทต์ตัวละเท่ากับจำตัวของมูลบล้อบคือ 500*17 ตัว และ Output 2 ตัว โดยกำหนดผลลัพย์ว่า ถ้าเป็นภาพหน้าคนจะต้องให้ผลลัพธ์ของ NetWork ตัวแรกมากกว่า 0.5 และเน็ตเวอร์คตัวที่หลังนั้นน้อยกว่า 0.5 ทั้งกรณี Network 1 ชั้นและ 2 ชั้น

โดยในการฝึก(train) นิวรอลเน็ตเวอร์คนั้น เราจะนำข้อมูลบล็อบซึ่งได้จากขั้นตอนการวิเคราะห์บล็อบภาพ มาเก็บไว้มากๆ ทั้งภาพที่เป็นหน้าคนและไม่ใช่หน้าคน จากนั้นนำข้อมูลเหล่านี้ไปสอนให้แก่ โครงข่ายประสาทเทียม ทั้ง 2 คู่ โดยทำการสอนเนต์เวอร์คตัวที่ต้องการให้ Active(มีค่ามากกว่า 0.5)ขณะทดสอบภาพคน ด้วยค่า 1 และขณะแทนภาพอื่นๆที่ไม่ใช่หน้าคนด้วย 0 และฝึกเนตเวร์คอีกตัวหนึ่งด้วยค่ากลับกัน

โดยในช่วงของการ Train Network นั้น เพื่อให้ได้ค่าผลลัพธ์(ค่า weight) ของนิวรอลเนตเวอร์ค รวจเร็วยิ่งขึ้น จึงได้ทำการ port โปรแกรมไป run บนเครื่อง work station เพื่อลดเวลาในการ train network หลังจากนั้นจะนำค่า weight ที่ได้ไปทดลองใช้และเลือกเอาเฉพาะ pattern ที่ให้ผลลัพธ์ผิดพลาดมา รวมเข้ากับ input pattern เดิม แล้วนำไปทำการ train ใหม่อีกจนกว่าค่าความผิดพลาดจะลดลงถึงจุดที่พอใจ ก็จะนำค่า weight ที่ได้นั้นไปใช้ในโปรแกรมเพื่อใช้ในการวิเคราะห์ภาพที่ต้องการตรวจสอบต่อไป

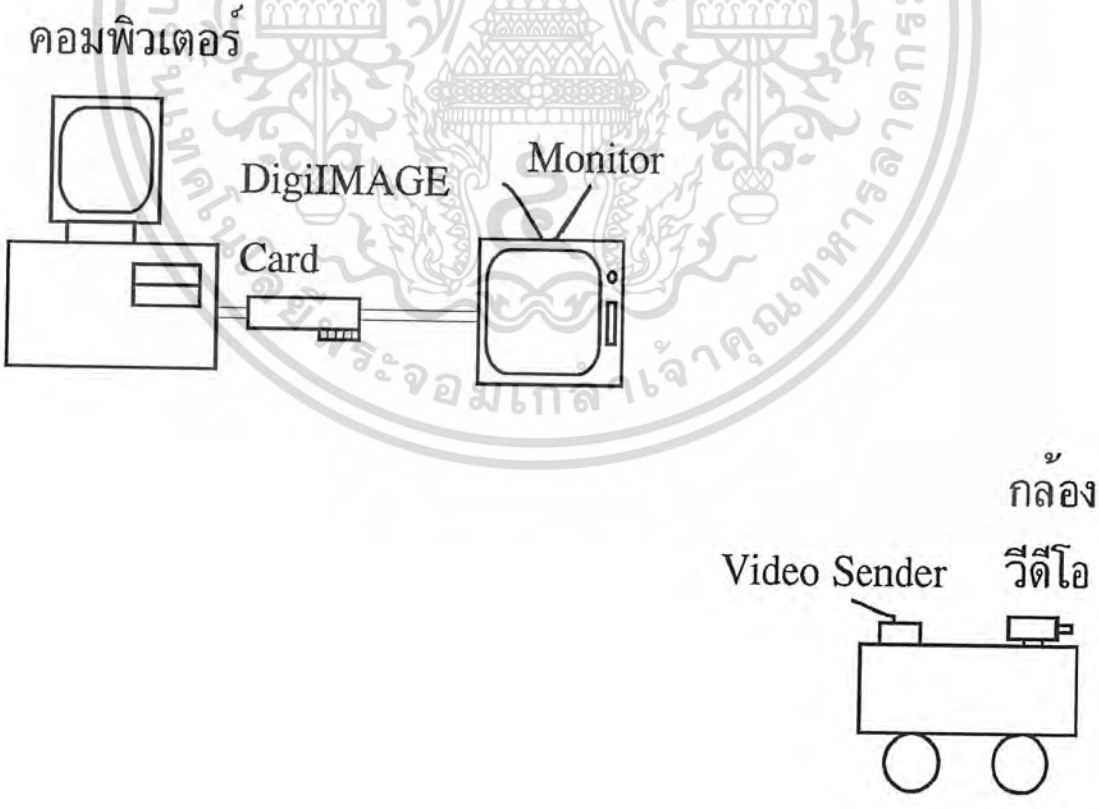
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 ส่วนการติดต่อกับตัวหุ่นยนต์

ในการทำการติดต่อรับสั่งข้อมูลระหว่างคอมพิวเตอร์ควบคุมและตัวหุ่นยนต์นั้น เราสามารถแบ่งออกเป็น 2 ส่วนใหญ่ ๆ คือ

3.6.1 ส่วนรับสัญญาณภาพจากกล้องวีดีโอ

ในส่วนนี้นั้นในขณะทดลองนี้เราจะทำการส่งสัญญานจากกล้องวีดีโอซึ่งอยู่บนตัวหุ่นยนต์ เป็นคลื่นโทรทัศน์ช่อง 23 โดยใช้เครื่อง V.D.O. Sender ซึ่งเป็นอุปกรณ์ที่สามารถแปลงสัญญาณวีดีโอ ส่งเป็นคลื่นโทรทัศน์ จากนั้นเราจะใช้เครื่องรับโทรทัศน์ซึ่งทำหน้าที่เป็นจอมอนิเตอร์ทำการรับสัญญาณช่อง 23 นี้เข้ามา แล้วแปลงคลื่นโทรทัศน์นี้เป็นสัญญาณวีดีโอส่งออกมาทางช่อง V.D.O. OUT ของเครื่องซึ่งต่ออยู่กับการ์ด DigiImage ซึ่งการ์ดนี้ก็จะทำการแปลงสัญญาน V.D.O. เป็นภาพเชิงเลข (digital image) อีกทีหนึ่ง รูปที่ 3.6.1 เป็นภาพแสดงวิธีการติดต่อส่งผ่านข้อมูลภาพระหว่างหุ่นยนต์กับเครื่องคอมพิวเตอร์ควบคุม



รูปที่ 3.6.1 ภาพแสดงวิธีการติดต่อส่งผ่านข้อมูลภาพระหว่างหุ่นยนต์กับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องคอมพิวเตอร์ควบคุม

3.6.2 ส่วนสื่อสารคอมมานด์และรับข้อมูลจากเซนเซอร์

ส่วนนี้จะทำหน้าที่รับคำสั่งต่าง ๆจากผู้ควบคุมส่งไปยังตัวหุ่นยนต์ รวมทั้งขอรับข้อมูลต่าง ๆจากเซนเซอร์ซึ่งติดตั้งอยู่บนตัวหุ่นยนต์ ซึ่งการสื่อสารข้อมูลในส่วนนี้จะติดต่อผ่านทางพอร์ตอนุกรม โดยกำหนดลักษณะสัญญาณไว้ดังนี

- ลักษณะสัญญาน มาตรฐาน RS232
- รูปแบบ Asyncronous
- อัตราบิท(bitrate) 2400 b/s
- จำนวนบิทใน 1 byte 8 bit
- Stop bit 1 bit

โดยสัญญาณดังกล่าวจะไม่ได้เชื่อมต่อกันระหว่างตัวหุ่นยนต์กับคอมพิวเตอร์ควบคุมโดยตรง แต่จะเชื่อมต่อผ่านวงจรรับส่งคลื่นวิทยุย่าน UHF ดังแสดงในรูปที่ 3.6.2 ซึ่งวงจรในส่วนแปลงสัญญาณรับสั่งนี้เป็นคลื่นวิทยุนั้นดำเนินการโดยภาควิชาวิศวกรรมโทรมนาคม โดยรูปแบบคำสั่งที่ใช้ในการสังงานหุ่นยนต์แสดงดังตารางที่ 3.6.1

คำสั่งที่ต้องการ	รูปแบบคอมมานด์	รูปแบบสัญญานตอบรับ
ความเร็วในการ เดินหน้า-ถอยหลัง	'R' <ระดับความเร็ว>	[ACK],[NAK]
ระบบบังคับเลี้ยว ซ้าย-ขวา	'S' < มุมเลี้ยว >	[ACK],[NAK]
อ่านเซนเซอร์ ด้านหน้า	'D' 'H'	[ACK]<ระยะห่าง>,[NAK]
อ่านเซนเซอร์ ด้านขวา	'D' 'R'	[ACK]<ระยะห่าง>,[NAK]
อ่านเซนเซอร์ ด้านซ้าย	'D' 'L'	[ACK]<ระยะห่าง>,[NAK]

ตารางที่ 3.6.1 รูปแบบของคอมมานด์ในการติดต่อกับหุ่นยนต์

โดย

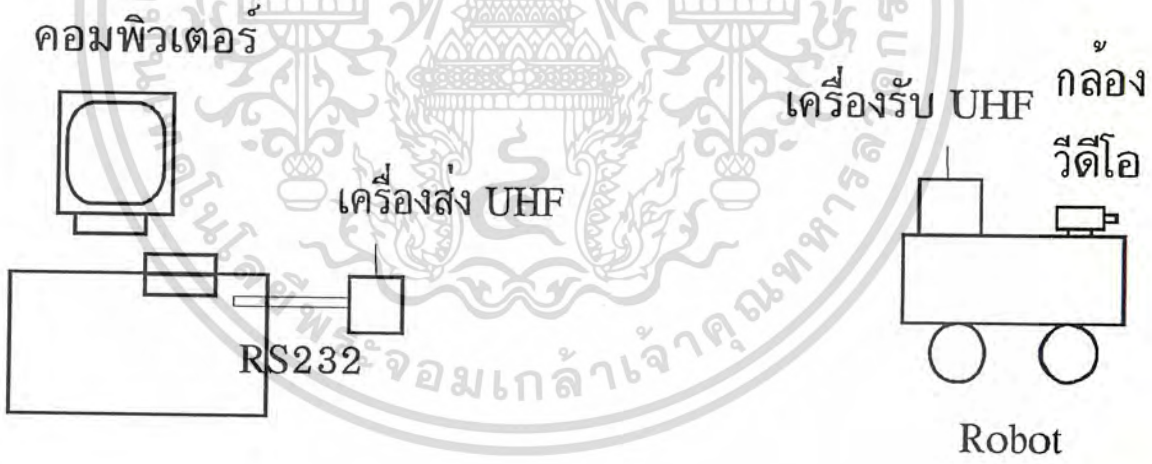
<ระดับความเร็ว> เป็น 1 - 7 โดยค่าน้อยกว่า 4 ทั้ง 3 ค่า เป็นการสั่งถอยหลังด้วยความเร็ว 3 ระดับ ค่า 4 เป็นการสั่งหยุด และค่ามากกว่า 4 เป็นการสั่งเดินหน้าด้วยความเร็ว 3 ระดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<มุมเลี้ยว> เป็น 1 - 7 โดยค่าน้อยกว่า 4 ทั้ง 3 ค่า เป็นการสั่งเลี้ยวซ้ายด้วยมุมเลี้ยว 3 ระดับ ค่า 4 เป็นการสั่งให้ตรงไป และค่ามากกว่า 4 เป็นการสั่งเลี้วงขวาด้วยมุมเลี้ยว 3 ระดับ

<ระยะห่าง> มีค่า 256 ระดับ(ข้อมูล 1 ไบต์) โดยค่าระยะห่างนี้เป็นระยะห่างระหว่างตัวหุ่นยนต์กับสิ่งกีดขวางในทิศทางต่าง ๆที่กำหนด โดยค่านี้จะแทนระยะห่างหน่วยเป็นเซนติเมตร แต่จะมีค่าความแม่นยำสูงในระยะไม่เกิน 100 เมตร

สำหรับรูปแบบของสัญญาณตอบรับจากหุ่นยนต์นั้น โดยปรกติจะต้องตอบรับโดยสัญญาน Acknowlage [ACK] (รหัส ASCII(6)) แล้วตามด้วยค่าระยะห่าง 1 byte หากเป็นการถามระยะทาง แต่ในบางกรณีที่ตัวหุ่นยนต์ไม่สามารถทำงานตอบสนองต่อคำสั่งนั้น ๆได้เช่น อุปกรณ์เสียหาย หรือ รูปแบบของคำสั่งผิดพลาด ก็จะทำการตอบกลับโดยใช้สัญญาน Negative Acknowlage [NAK] (รหัส ASCII(21))



รูปที่ 3.6.2 ภาพแสดงวิธีการส่งข้อมูลที่ใช้ควบคุมหุ่นยนต์โดยเครื่องคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

4.1การทดลองเกี่ยวกับการรับภาพ

ในการทดลองเกี่ยวกับการรับภาพนั้น เรารับภาพมาด้วยกัน 2 แบบ คือ 64 ระดับเทา และ 256 ระดับเทา ซึ่งแปลงมาจากภาพสี ขนาด 64 x 64 x 64 สี นำข้อมูลภาพจากทั้งสองวิธีนี้เข้าไปประมวลผลเพื่อหาขอบภาพ

ผลการทดลองปรากฏว่า ภาพ 64 ระดับเทาสามารถประมวลผลได้เร็วกว่า ภาพ 256 ระดับเทาเพียงเล็กน้อย ส่วนสำหรับผลของการหาขอบภาพนั้น ภาพ 64 ระดับเทาจำเป็นจะต้องมีความเข้มแสงมากจึงจะสามารถแยกขอบภาพต่าง ๆออกมาได้ ซึ่งสำหรับภาพ 256 ระดับเทานั้น สามารถหาขอบภาพได้ดีกว่ามาก

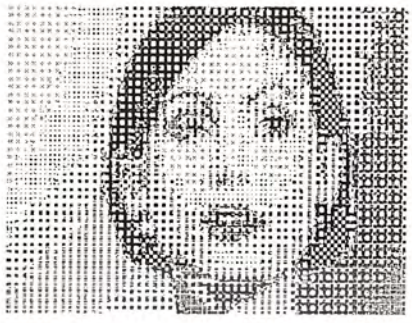
4.2 การทดลองปรับปรุงคุณภาพของภาพ

ในการทดลองครั้งนี้มีการใช้วิธีการประมวลผลภาพต่าง ๆเพื่อปรับปรุงคุณภาพของภาพให้เหมาะสมในการนำไปตรวจหาขอบภาพ ซึ่งภาพแต่ละภาพก็ต้องการกรรมวิธีในการปรับปรุงภาพที่แตกต่างกันไป ซึ่งในที่นี้เราได้ทำการทดลองแสดงให้เห็นถึงคุณประโยชน์ของวิธีการต่าง ๆในการใช้ปรับปรุงภาพสีแต่ละแบบ โดยวิธีการต่าง ๆที่ใช้ในการประมวลผลนี้จะมีดังนี้

- ***LowPass filter (LP)***
- ***Sum Square Delta (SQ)***
- ***ZeroCrossing (ZC)***
- ***Histogram Equalization (HQ)***
- ***Shortest Spanning Tree(SST)***

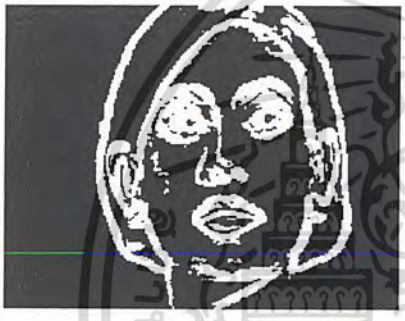
โดยตัวอย่างผลของการประมวลผลด้วยวิธีต่าง ๆนั้น แสดงดังรูปที่ 4.1 - 4.4 โดยแต่ละภาพนั้นจะผ่านวิธีการประมวลผลในลำดับต่าง ๆกันไปตามลำดับตัวย่อของวิธีการที่ระบุอยู่ใต้ภาพ เช่น HQ-LP-SQ หมายถึงเป็นภาพที่ผ่านกระบวนการ histogram equalization ตามด้วย lowpass filter และการบวนการหาค่า ZeroCrossing ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) ภาพต้นแบบ

(ข) ZC



(ค) SQ

(ง) LP-ZC

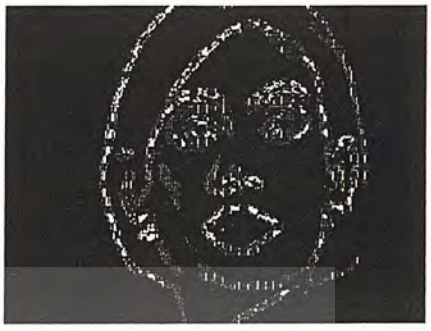
(จ) LP-SQ

(ฉ) LP-ZC-HQ

รูปที่ 4.1

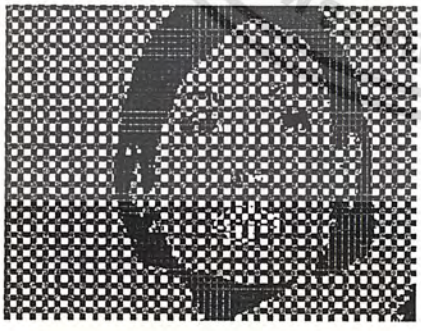
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ก) HQ-SQ



(ข) HQ-LP-ZC

(ค) HQ-LP-SQ

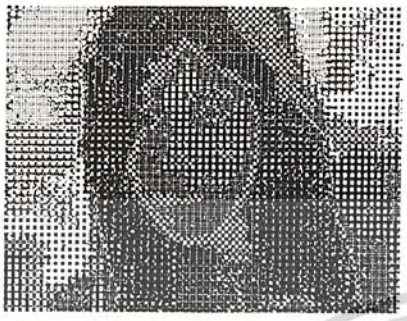


(ง) SST

(จ) LP-SST

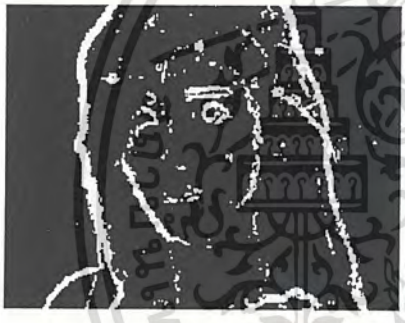
รูปที่ 4.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) ภาพต้นแบบ

(ข) HQ



(ค) SQ

(ง) LP-SQ

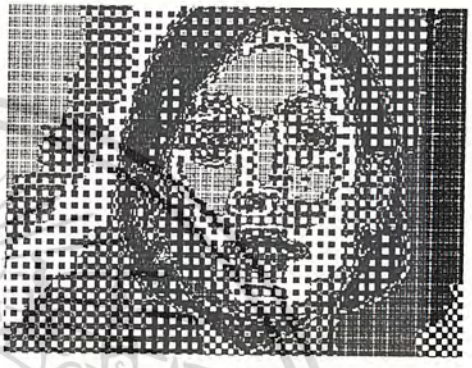
(จ) HQ-SQ

(ฉ) HQ-LP-SQ

รูปที่ 4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ก) LP-HQ-ZC-SQ



(ข) LP

(ค) LP-ZC-SQ

(ง) ZC-SQ

รูปที่ 4.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเหตุ

ในผลการทดลองนี้การทำ lowpass filter หมายถึงการทำ *lowpass 2 ครั้งติดกัน* และ Sum Square Delta นั้นจะต้องใช้ค่า ***threshold เท่ากับ 256*** เสมอ ซึ่งค่านี้เป็นผลจากการทดลองหลาย ๆครั้งจนพบว่าเป็นค่าที่ให้ประสิทธิภาพโดยเฉลี่ยของการตรวจหาขอบภาพสูงสุด และสำหรับกระบวนการ ZeroCrossing นั้นภาพที่ได้จะถูกแสดงในแบบ 256 ระดับเทาเนื่องจากค่า threshold ที่เหมาะสมสำหรับแต่ละภาพมีค่าไม่แน่นอน และสำหรับวิธี SST นั้นจะเป็นการทำ *SST แบบ 200 ส่วน(region)* และค่าสีของแต่ละส่วนจะไม่มีความเกี่ยวข้องกัน ค่าสีที่ต่างกันหมายถึงว่าหลังจากการใช้ SST ตรวจจับแล้วพบว่าเป็น region คนละส่วนกัน ระดับความเข้มสีต่าง ๆไม่ได้มีความหมายใด ๆทั้งสิ้น

4.3 การทดลองหาวิธีปรับค่าอัตราการเรียนรู้ของโครงข่ายสมองเทียมโดยอัตโนมัติ

ในการทำการทดลองนี้ก็เพื่อที่จะหาวิธีการลดเวลาที่โครงข่ายสมองเทียมใช้ในการเรียนรู้ ซึ่งเวลานี้จะขึ้นอยู่กับค่าของ อัตราการเรียนรู้ ซึ่งค่านี้จะอยู่ใน ช่วง 0 ถึง 1 ซึ่งหามีค่าเป็น 0 ก็หมายความว่าเน็ตเวอร์คจะไม่สามารถเรียนรู้ได้เลย ยิ่งค่านี้มีค่าสูงเท่าไหร่ก็จะเรียนรู้ได้เร็วเท่านั้น แต่ในทางปฏิบัตินั้นโดยปรกติค่านี้จะถูกตั้งไว้คงที่และให้มีค่าต่ำคือน้อยกว่า 0.3 ทำให้การเรียนรู้เป็นไปได้ช้า

เหตุที่มักตั้งค่าอัตราการเรียนรู้กันไว้ต่ำก็เนื่องจากหากตั้งค่านี้สูงเกินไป เมื่อเนตเวอร์เรียนรู้ไปถึงจุดหนึ่งก็จะเกิดการออสซิเลตขึ้นโดยค่าความผิดผลาด(Error) จะแกว่งขึ้น ๆลง ๆ ไม่เข้าสู่เป้าหมาย

ในการทดลองครั้งนี้ได้ทดลองทำการทดลองหลายแบบ โดยใช้เน็ตเวอร์คแบบ 1 ชั้นและ 2 ชั้นในการเรียนรู้ โดยมีวิธีในการปรับค่าดังนี้

วิธีที่ 1. ตั้งค่าไว้สูงสูง(0.9) หากค่าความผิดพลาดในขณะใดเพิ่มขึ้น ก็จะทำการลดอัตราการเรียนรู้ลงเหลือ 90% จากเดิม ทุก ๆครั้งที่เพิ่มขึ้น

วิธีที่ 2 ตั้งค่าไว้สูงสูง(0.9) หากค่าความผิดพลาดในขณะใดเพิ่มขึ้น ก็จะทำการลดอัตราการเรียนรู้ลงเหลือ 90% จากเดิม แต่หากค่าความผิดพลาดลดลง 5 ครั้งติดกัน ก็จะทำการเพิ่มอัตราการเรียนรู้ขึ้นอีก 10% จากเดิมทุก ๆรอบของการทดสอบจนกว่าจะมีช่วงที่ค่าความผิดพลาดเพิ่มขึ้นหรืออัตราการเรียนรู้สูงจนถึง 0.95

วิธีที่ 3 ตั้งค่าไว้สูงสูง(0.9) หากค่าความผิดพลาดในขณะใดเพิ่มขึ้น ก็จะทำการลดอัตราการเรียนรู้ลงโดย

อัตราการเรียนรู้ = **อัตราการเรียนรู้เดิม** x ค่าความผิดพลาดเดิม/ค่าความผิดพลาดใหม่

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่หากค่าความผิดพลาดลดลงก็จะทำการเพิ่มอัตราการเรียนรู้ขึ้นอีก 10% ทุกครั้งที่ค่าความผิดพลาดลดลง 5 ครั้งติดกัน จนกว่าค่าอัตราการเรียนรู้สูงจนถึง 0.95

ผลการทดลอง

จากการสอนเน็ตเวร์คทั้ง 2 ชนิด(1ชั้น กับ 2 ชั้น)ด้วยการปรับอัตราการเรียนรู้ 3 วิธีที่ได้กล่าวไปแล้ว ได้ผลของแต่ละวิธีดังนี้

วิธีที่ 1 เน็ตเวอ์คทั้ง 2 แบบสามารถเรียนรู้ได้ โดยช่วงแรกค่าอัตราการเลียนรู้จะลดลงเรื่อย ๆจนถึงจุดหนึ่งก็จะเริ่มคงที่

วิธีที่ 2 สำหรับเน็ตเวอร์ค 1 ชั้นในกรณีที่ข้อมูลอินพุทน้อย ๆ(100 ชุด) จะสามารถเรียนรู้รวดเร็วมากที่สุด โดยช่วงแรกค่าอัตราการเรียนรู้จะลดลงจากนั้นจะค่อย ๆ เพิ่มขึ้นมาเรื่อย ๆจนถึง 0.95 แล้วก็จงคงที่จนเสร็จการเรียนรู้ และในกรณีที่ข้อมูลอินพุทมีมากประมาณ(300 ชุด) จะให้ผลเช่นเดียวกันเน็ดเวอร์ค 2 ชั้น คือส่วนใหญ่จะไม่สามารถเรียนรู้ได้ โดยค่าความผิดพลาดและอัตราการเรียนรู้จะวิ่งขึ้น ๆลง ๆตามกันไป

วิธีที่ 3 เน็ตเวร์ก 2 ชั้นนั้นไม่สามารถทำการเรียนรู้ได้เช่นเดียวกับวิธีที่ 2 แต่สำหรับเน็ตเวอร์คชั้นเดียวนั้นสามารถที่จะทำการเรียนรู้ได้แม้ว่าจะเพิ่มข้อมูลอินพุทจนถึง 300 ชุดก็ตาม โดยในช่วงแรกค่าความผิดพลาดจะเพิ่มขึ้น และค่าอัตราการเรียนรู้จะลดลง แล้วก็จะเพิ่มขึ้นแล้วก็จะลดลงในบางครั้งแล้วก็เพิ่มขึ้นอีกจนกระทั่งคงที่ ที่ 0.95

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และสรุป

5.1 ผลการวิเคราะคุณสมบัติของการประมวลผลภาพด้วยวิธีการต่าง ๆ

จากการทดลองต่าง ๆดังที่ได้แสดงในบทที่แล้ว เราสามารถที่จะทำการสรุปคุณสมบัติ ข้อดีข้อเสีย ของแต่ละวิธีการ และ สามารถพิจารณาหาลำดับขั้นตอนที่ดีที่สุดสำหรับภาพทั่ว ๆ ไปได้ดังนี้

5.1.1 Sum Square Delta

จากการทดลองสรุปได้ว่าวิธีนี้เป็นวิธีที่ได้ผลดีที่สุดสำหรับภาพทั่ว ๆไป เมื่อเทียบกับอีก 3 วิธีคือ High Pass filter, ZeroCrossing และ SST และยังเหมาะสมสำหรับนำไปอิมพลิเมนต์ปัญหาที่ใช้ เนื่องจากในวิธีนี้ค่าระดับ threshold ที่เหมาะสมนั้นจะมีค่าค่อนข้างคงที่ โดยทุก ๆภาพที่ทดลองนั้น เราตั้งค่า threshold ไว้คงที่ที่ 256 แต่การที่จะให้ผลของการตรวจจับขอบภาพดีเพียงดีนั้น ก็จะยังขึ้นอยู่กับวิธีการประมวลผลเพื่อปรับข้อมูลอินพุทที่เลือกใช้ให้มีความเหมาะสม ก็จะช่วยเพิ่มประสิทธิภาพการตรวจหาขอบภาพได้เป็นอย่างดี ดังจะกล่าวต่อไป

5.1.2 ZeroCrossing

จากการทดลองพบว่า วิธีนี้นั้นมีประสิทธิภาพในการหาจุดเปลี่ยนแปลงได้ดีแต่ระดับของความเปลี่ยนแปลงที่ได้มีค่าน้อยและจะมีค่าใกล้เคียงกัน ดังแสดงในรูปที่ 4.1(ข) และ (ง) ภาพนี้สามารถทำให้เห็นชันเจนขึ้นโดยการผ่าน histogram equalize ก็จะทำให้เห็นจุดที่เกิดการเปลี่ยนแปลงระดับสีที่ถูกตรวจจับโดยวิธีนี้ชัดเจนขึ้นดังแสดงในรูปที่ 4.1(ฉ) ซึ่งจะเห็นว่าวิธีนี้มีความอ่อนไหวต่อการเปลี่ยนแปลงของระดับสีมากเกินไป ซึ่งการที่จะแปลงภาพนี้ให้เป็นภาพไบนารี 2 สีนั้นจะทำได้ยาก เนื่องจากต้องใช้วิธีการ thresholding ซึ่งเป็นการยากที่จะหาค่า threshold ที่เหมาะสม เนื่องจากภาพที่ได้จากวิธีนี้มีค่า threshold ที่เหมาะสมแตกต่างกัน

5.1.3 Shortest Spanning Tree(SST)

จากการทดลองนั้นจะพบว่าภาพที่ออกมานั้นได้ผลไม่ดีเท่าที่ควรดังแสดงในภาพที่ 4.2 (จ) ซึ่งมีการแบ่งเป็น region ทั้งหมด 200 region ซึ่งถ้าเราทำการลดจำนวน region จนเหลือ 50 region (หมายถึง 50 บล็อบเท่ากับที่ใช้ในการประมวลผล) ก็จะยิ่งทำให้ region ต่าง ๆนั้นถูกรวมทดหายไปอีก เหตุผลก็เนื่องจากภาพที่รับเข้ามาโดยปรกตินั้นจะมีจุดซึ่งมีค่าระดับสีแตกต่างจากจุดรอบข้างอยู่มาก ซึ่งจุดเหล่านี้จะมีผลต่อการที่จะทำให้ region ที่ถูกต้องเหมาะสม 2 region ต้องถูกยุบรวมกัน เพื่อให้เหลือจำนวน region เท่าที่ต้องการ เนื่องจากจุดสีเดี่ยว ๆที่มีอยู่ในภาพนั้นจะกินจะนวนของ region ไว้มาก ทำให้เหลือจำนวน region สำหรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เดี่ยว ๆที่มีอยู่ในภาพนั้นจะทำจำนวนของ region ไว้มาก ทำให้เหลือจำนวน region สำหรับ region จริง ๆ น้อยเกินไป

นอกจากนี้ข้อเสียอีกอย่างของวิธีนี้ก็คือสำหรับกลุ่มภาพ 2 กลุ่มซึ่งมีสีต่างกันเมื่อมีการไล่ระดับสีเทาเข้ามาใกล้กันเรื่อย ๆหากมีระดับสีที่จุดใด ๆเท่ากัน ภาพทั้ง 2 ส่วนนั้นก็จะถูกกลืนรวมเป็นส่วนเดียวกันทั้งหมดไม่เหลือแม้กระทั่งรอยต่อซึ่งสามารถที่จะตรวจจับได้ในกระบวนการหาขอบภาพซึ่งเป็นจุดที่ด้อยกว่าดังแสดงในรูป 4.2(ง) และ 4.2(จ)

5.1.4 LowPass filter

วิธีนี้มีประสิทธิภาพในการช่วยกำจัดจุดซึ่งไม่เป็นระเบียบในพื้นภาพได้เป็นอย่างดี เมื่อเราใช้ในกรรมวิธีนี้ก่อนที่จะนำไปคำนวนหาขอบภาพ จะทำให้ลดจุดขยะในภาพนี้ได้อย่างมาก ดังแสดงในรูปที่ 4.2(ก) กับ 4.2(ข) และรูปที่ 4.3(จ) กับ 4.3 (ฉ) แต่หากนำไปใช้กับภาพที่มีลักษณะเบลอ ซึ่งขอบภาพเดิมก็มีน้อยอยู่แล้ว เมื่อผ่าน lowpass filter ก็จะทำให้ขอบภาพขาส่วนถูกกรองหดหายไป ดังแสดงในรูปที่ 4.3(ค) กับ 4.3 (ง)

5.1.5 Histogram Equalization

วิธีนี้มีประสิทธิภาพดีในการช่วยให้สีต่าง ๆในภาพกระจายกันอย่างทั่วถึง ทำให้ภาพที่ได้มีความชัดเจนยิ่งขึ้นดังแสดงในรูป 4.3(ก) และ 4.3(ข) แต่กระบวนการนี้จะทำให้พื้นผิวของภาพที่ได้ไม่สม่ำเสมอ ไม่เหมาะสมที่จะนำไปตรวจหาขอบภาพ เพราะจำทำให้เกินจุดต่าง ๆ กระจายอยู่ทั่วภาพ ดังแสดงในรูปที่ 4.2(ก) และ 4.3(จ) ซึ่งภาพที่ได้นี้จะมีคุณภาพแย่กว่าภาพที่ได้จากการตัดขอบภาพโดยไม่ผ่านกระบวนการนี้เสียอีก(ดูรูป 4.1(ค) และ 4.3(ค) เทียบ) แต่ อย่างไรก็ตามเมื่อนำ lowpass filter เข้ามาช่วยแล้วก็จะทำให้การตรวจหาขอบภาพมีประสิทธิภาพสูงขึ้นอย่างมากทีเดียว ดังแสดงในรูปที่ 4.2 (ค) และ 4.3(ฉ)

จากผลการทดลองจะเห็นว่าชุดของการประมวลผล HQ-LP-SQ จะมีประสิทธิภาพโดยทั่วไปของการหาขอบภาพดีที่สุด มีเพียงบางครั้งเท่านั้นที่จะให้จำนวนของขอบภาพเยอะเกินไป ซึ่งก็ถือว่าการประมวลผลเรียงตามลำดับนี้นั้นเหมาะสมที่สุด เว้นแต่ว่าจะใช้อัลกอริธึมที่สามาถเลือกพิจารณาใช้กระบวนการต่าง ๆในการประมวลผลใช้เหมาะสมกับภาพได้โดยไม่ใช้เวลาในการเลือกวิธีประมวลผลมากเกินไป

5.2 แนวทางในการพัฒนาต่อสำหรับเพิ่มประสิทธิภาพในการหาของภาพ

จากผลการทดลองต่าง ๆที่ได้กล่าวไปแล้วนั้นสามารถวิเคราะห์วิธีการพัฒนาต่อเพื่อเพิ่มประสิทธิภาพในการหาขอบภาพได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.1 การเพิ่มประสิทธิภาพของวิธี Sum Square Delta

แม้ว่าวิธีการนี้นั้นจะให้ขอบภาพที่ค่อนข้างดีกว่าวิธีอื่น แต่จะเห็นว่าขอบภาพที่ได้นั้นไม่ต่อเนื่องเท่าที่ควร ซึ่งการขาดของขอบภาพนี้จะมีผลทำให้ขอบภาพที่ได้ไม่เป็นวงปิด ซึ่งจะทำให้ไม่ได้รูปบล็อบที่สมบูรณ์ ฉะนั้นนอกจากการประมวลผลโดยใช้ลำดับต่าง ๆเพื่อเพิ่มความสามารถในการตรวจหาขอบภาพดังที่กล่าวไปแล้วนั้น หากเราสามารถ*เพิ่มวิธีการของ Edge Linked* เพื่อเชื่อมขอบภาพให้ต่อกันสมบูรณ์เป็นวงรอบได้ ก็จะเป็นการเพิ่มความเชื่อถือได้ของระบบ และให้สามารถได้บล็อบภาพที่มีความสมบูรณ์สูงขึ้น

5.2.2 ZeroCrossing

แม้ว่าวิธีนี้จะไม่สามารถแบ่งแยกขอบภาพได้อย่างชัดเจนเหมือนวิธี Sum Square Delta ก็ตาม แต่ข้อดีของวิธีนี้นั้นก็คือได้เส้นขอบต่าง ๆค่อนข้างครบ(แต่เกิน) ฉะนั้นหากสามารถหาอัลกอริทึมที่สามารถแยกเฉพาะเส้นทางที่สมบูรณ์ออกมาได้ ก็จะทำให้ประสิทธิภาพของวิธีนี้สูงกว่าวิธี Sum square delta เสียอีก

5.2.3 Shortest Spanning Tree(SST)

แม้ว่าผลภาพที่ได้จากวิธีนี้นั้นยังอยู่ในระดับที่ไม่น่าพอใจ แต่จุดเด่นของวิธีการนี้นั้นอยู่ที่การวิเคราะห์ภาพโดยนำเอาข้อมูลส่วนใหญ่ของภาพ (global data) มาใช้เป็นเกณฑ์ในการตัดสินการแยกและการรวมของ region ด้วย ดังนั้นการทำเซกเมนเตชันโดยใช้วิธีของทฤษฎีกราฟจึงน่าจะสามารถให้ภาพเซกเม้นต์ที่ได้ดีกว่าการพิจารณาเฉพาะข้อมูลรอบ ๆจุดที่ใช้ในวิธีการหาขอบภาพ ซึ่งหากเราทำการ หาวิธีแก้ไปญญหาทั้ง 2 อย่างที่ได้ก็อาจจะให้ประสิทธิภาพที่ดีกว่าวิธีเดิมที่ใช้อยู่ก็ได้ ซึ่งวิธีการที่จะแก้ปัญหาเหล่านี้วิธีหนึ่งนั้น ก็ทำได้โดยการใช้วิธี RSST (Recursive SST) แทนวิธี SST ซึ่งใช้อยู่ในปัจจุบัน

5.2.4 Thresholding

แม้ว่าในการทดลองประมวลผลภาพที่ผ่านมานั้น เราจะไม่ได้ทำการทดลองเกี่ยวกับการทำ thresholding นัก เนื่องจากการทำงานในส่วนนี้ไม่ได้ซับซ้อนนัก แต่หากมีกรรมวิธีในการทำ threshold ที่ดี ก็จะสามารถเพิ่มประสิทธิภาพของวิธี Sum square delta และวิธี Zero Cossing ได้ โดยกรรมวิธีที่ว่านี้ก็ได้แก่การเพิ่มกระบวนการตรวจสอบและพิจารณาค่า threshold ที่ เหมาะสมสำหรับแต่ละภาพโดยอัตโนมัติ หรือถ้าสามารถพิจารณาแยกค่า threshold ที่เหมาะสมสำหรับแต่ละส่วนของภาพได้ ก็จะยิ่งเป็นการดี ซึ่งวิธีการที่กล่าวมานี้ จะทำให้ประสิทธิภาพในการหาขอบภาพของวิธีทั้ง 2 ที่กล่าวมาแล้วดียิ่งขึ้นไปอีก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 สรุปผลการทดลองหาวิธีปรับค่าอัตราการเรียนรู้ของโครงข่ายสมองเทียมโดยอัตโนมัติ

จากรผลการทดลองที่ได้นั้น แสดงให้เห็นว่าการแปลงแปลงค่าอัตราการเรียนรู้ขณะโปรแกรมกำลังทำการเรียนรู้อยู่นั้นจะต้องใช้วิธีที่มีความซับซ้อนและละเอียดมากกว่าวิธีที่ใช้อยู่ ซึ่งจะเห็นว่าไม่สามารถทำการเรียนรู้บนเน็ตเวอร์ค 2 ชั้นได้เลย และในเน็ตเวอร์กแบบชั้นเดียวนั้น เมื่อมีข้อมูลอินพุทเพิ่มขึ้นก็จะทำให้วิธีการปรับอัตราการการเรียนรู้ในวิธีที่ 2 ไม่สามารถทำได้ แต่สำหรับวิธีที่ 3 ซึ่งยังสามารถเรียนอยู่ได้นั้นเมื่อพิจารณาจาก 2 วิธีแรก ก็ทำให้ไม่สามารถยืนยันได้ว่าจะสามารถใช้กับข้อมูลอินพุทที่มากกว่านี้ได้หรือไร ฉะนั้นในโปรแกรมส่วนนี้จึงอาจต้องใช้การปรับเปลี่ยนอัลกอริธึมต่าง ๆตามสภาพของค่าความผิดพลาดที่เกิดขึ้น เพื่อให้สามารถเรียนรู้ได้โดยใช้ระยะเวลาไม่นานนัก

5.4 สรุประบบการทำงานในส่วนของการประมวลผลภาพ

หลังจากทำการทดลองในขั้นตอนต่าง ๆดังที่กล่าวมาแล้วในบทที่ 4 และจากการสรุปผลการทำงานของวิธีการต่าง ๆ ในบทนี้ เราจะสรุปรายละเอียดของระบบงานที่เลือกใช้ได้ดังนี้

ระบบจะเริ่มต้นโดยการรับสัญญานภาพจากการ์ด DigiImage โดยนำภาพขาวดำ 64 ระดับเทาความละเอียด 128*128 จุด ไปทำการแสดงบนจอภาพวีจีเอโหมด 13h เมื่อต้องการทดสอบภาพใด ๆ ก็จะทำการอ่านข้อมูลภาพขนาด 256*256 จุด 64*64*64 สี มาแปลงเป็นแบบ 256 ระดับเทา จากนั้นก็นำไปทำการประมวลผลโดยผ่านกระบวนการ histogram equalization จากนั้นนำภาพที่ได้มาผ่านการทำ lowpass filter อีก 2 ครั้งแล้วนำภาพที่ได้ไปหาขอบภาพโดยกระบวนการ Sum Square Delta ซึ่งก็จะได้ภาพไบนารี จากนั้นก็นำไปทำการกำจัดจุดบางส่วนออก แล้วจึงทำการวิเคราะห์หาบล็อบภาพส่งบล็อบภาพที่ได้ 50 บล็อบไปยัง นิวรอลเน็ตเวอร์คแบบ 1 ชั้นขนาด 50 x 17 อินพุท 2 เอาท์พุท ทำการคำนวนผลลัพธ์ แล้วนำผลที่ได้มาพิจารณาว่าเป็นภาพหน้าคนหรือไม่ แล้วแสดงผลออกทางจอภาพ ก็เป็นอันเสร็จสิ้นกระบวนการทดสอบ

5.5 สรุปผลโครงงาน

โครงงานการประมวลผลภาพของหุ่นยนต์สำรวจนี้นั้น เป็นโครงงานซึ่งจะเป็นประโยชน์ในการพัฒนาระบบหุ่นยนต์สำรวจต่อ ๆไป ผลจากโครงงานชิ้นนี้สามารถที่จะนำไปพัฒนาระบบหุ่นยนต์ซึ่งใช้ภาพในการประมวลผลได้ สำหรับผลของการทดลองใช้กับภาพหน้าคนนั้นสามารถทำงานได้เป็นที่น่าพอใจอย่างยิ่ง นอกจากนี้ในการประมวลผลนั้น ยัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีการใช้เทคโนโยลีของโครงข่ายสมองเทียม ทำให้สามารถนำไปใช้ในการวิเคราะห์หาความหมายของภาพได้โดยไม่ต้องเปลี่ยนโปรแกรมหรืออัลกอริธึม เป็นการสร้างระบบประมวลผลภาพซึ่งมีความยืดหยุ่นสูงพอสมควร โครงงานนี้จึงเป็นสิ่งซึ่งน่าสนใจเป็นอย่างมากสำหรับผู้สนใจสร้างระบบหุ่นยนต์ที่มีความฉลาดในการประมวลผลด้วยตนเอง

สำหรับอุปสรรคของการทำงานในครั้งนี้นั้น ก็มีบ้างพอสมควร ซึ่งส่วนใหญ่จะเกิดจากการศึกษาระบบและ เทคโนโลยีใหม่ๆ ซึ่งไม่ได้มีความรู้ความชำนาญมาก่ก่อน เช่น เรื่องของการประมวลผลภาพ เรื่องของโครงข่ายสมองเทียม วิธีการย้ายโปรแกรมไปทำงานขนเครื่องคอมพิวเตอร์ Work Station ซึ่งใช้ระบบ UNIX ฯลฯ แต่ปัญญาต่างๆเหล่านี้นั้น กลับเป็นแรงใจให้กลุ่มของข้าพเจ้าทำการศึกษาหาความรู้เพิ่มเติม จนสามารถที่จะทำโครงงานชิ้นนี้ให้ประสพความสำเร็จได้ ในที่สุด

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

วารสาร

[1] ไชยันต์ สุวรรณชีวะศิริ," เจาะกึ๋นหุ่นยนต์ ตอน การวิเคราะห์ความต่อเนื่องของจุด ",วารสารเซมิคอนดักเตอร์ อิเลคทรอนิคส์,ฉบับที่ 122, พ.ย. 35,หน้า 105-111.

[2] ดร.ชิดชนก เหลือสินทรัพย์," โครงข่ายประสาทเทียม (Artificial Neural Network) ", วารสารเซมิคอนดักเตอร์ อิเลคทรอนิคส์,ฉบับที่ 121, ต.ค. 35,หน้า 91-97.

ปริญญานิพนธ์

[1] สุชาติ ฉิมประดิษฐ์,ร.ศ.ดร.ฟูศักดิ์ ชีวสุวิทย์," การลดข้อมูลภาพโดยการเข้ารหัสพื้นที่ของภาพ ",วิทยานิพนธ์สำหรับปริญญาวิศวกรรมศาสตร์มหาบัณฑิต,สาขาวิศวกรรมไฟฟ้า,คณะวิศวกรรมศาสตร์,บัณฑิตวิทยาลัย,สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง,ปีการศึกษา 2534,หน้า 28-43 และ 78-92.

[2] เชิดพงษ์ รัตนอำนวยชัย,สยม งามเจริญ,สุกิจ เมฆจำเริญ,ผศ.ดร.บุญวัฒน์อัตชู," การประมวลผลภาพ ",ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตร์บัณฑิต,สาขาวิศวกรรมคอมพิวเตอร์ ,คณะวิศวกรรมศาสตร์,สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง,ปีการศึกษา 2533,หน้า 9-30 และ 56-67.

[3] กฤษณ์ โกสวัสดิ์,กฤษฎ์ชัย สมสมาน,ธีรพงษ์ โศภิษฐิกุล,รศ.ดร.โยธิน เปรมปราณีรัชต์, " ROBOT VISION ",ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตร์บัณฑิต,ภาควิชาระบบควบคุม,คณะวิศวกรรมศาสตร์,สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง,ปีการศึกษา 2533.

หนังสือ

[1] Don Pearson," Image Processing ",Mc Graw-Hill Book Company,1991

[2] Wahl,F.M.(Friedrich M.)," Digital Image Signal Processing ",Artech House,INC.,1987.

[3] Rafael C.Gonzalez,Paul Wintz," Digital Image Processing ",Second Edition,Addison-Wesley Publishing Company,1987.

[4] Stephen P.Banks," Signal Processing,Image Processing and Pattern Recognition ", Prentice Hall,1990.

[5] Aeriel Rosenfeid and Avirasho.kak," Computer Science and Applied Mathematics Digital Picture Processing ",Volume 2,Second Edition,Academic Press,INC.,1982.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[6] Edward R. Dougherty,Chareles R. Giardinia,″ Matrix Structured Image Processing ″,Prentice-Hall,INC.,1987.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ในการทำโครงงานชิ้นนี้ จะไม่สามารถสำเร็จลุล่วงไปได้เลย ถ้าขาดการสนับสนุนและคำแนะนำอันมีค่าจากคุณพ่อคุณแม่,คณาจารย์ และเพื่อน ๆ ดังนั้นจึงอยากจะแสดงความขอบคุณแด่

- พ่อและแม่ที่ให้ความเข้าใจและให้ความสนับสนุนในการศึกษา และในการทำโครงงานชิ้นนี้
- อ.ครรชิต อาจารย์ที่ปรึกษาที่ให้โอกาสทำโครงงานชิ้นนี้
- อ.วัชระ ที่ให้คำแนะนำทางด้าน Neural Network
- อ.บุญธีร์ ที่เอื้อเฟื้อให้ใช้เครื่องในห้องของอ.บางโอกาสที่อ.ไม่อยู่
- ม่อน ที่ให้คำปรึกษาเกี่ยวกับ image processing
- ไก่ ที่ให้คำปรึกษาเกี่ยวกับการทำ implement Neural Network
- เพ้ง ที่ให้คำแนะนำเกี่ยวกับการใช้งานบนระบบ UNIX และการ port program Neural Network ลงสู่ UNIX
- เปิ้ล และหนุย ที่ช่วยพิมพ์รายงานบางส่วนในฉบับนี้
- เพื่อน ๆ ทุก ๆคนที่เอื้อเฟื้อให้ถ่ายภาพเพื่อนำมาใช้ในโครงงาน

และผู้มีอุปการะคุณคนอื่น ๆ ที่ไม่ได้กล่าวมาในณ.ที่นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้