สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การรู้จำวัตถุจากภาพเงาโครงร่าง

OBJECT RECOGNITION FROM SHADOW SHAPE

T042794

นายสุรศักดิ์ ทับเที่ยง
นายสุรสิทธิ์ ทับเที่ยง

เลขหมู่.....
เลขทะเบียน..... 42794
วัน, เดือน, ปี 10 มิ.ย. 2545

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรู้จำวัตถุจากภาพเงาโครงร่าง

OBJECT RECOGNITION FROM SHADOW SHAPE

โดย

นายสุรศักดิ์ ทับเที่ยง

นายสุรสิทธิ์ ทับเที่ยง

อาจารย์ที่ปรึกษา

รศ.ดร. ชม กิ้มปาน

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2543

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การรู้จำวัตถุจากภาพเงาโครงเงา

OBJECT RECOGNITION FROM SHADOW SHAPE

ผู้จัดทำ

1. นาย สุรศักดิ์ ทับเที่ยง รหัสประจำตัว 41013555
2. นาย สุรสิทธิ์ ทับเที่ยง รหัสประจำตัว 41013556

_____ อาจารย์ที่ปรึกษา

(รศ.ดร. ชม กิ้มปาน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรู้จำวัตถุจากภาพโครงร่างเงา

นายสุรศักดิ์ ทับเที่ยง 41013555
นายสุรสิทธิ์ ทับเที่ยง 41013556
รศ.ดร. ชม กิ้มปาน อาจารย์ที่ปรึกษา
ปีการศึกษา 2543

บทคัดย่อ

ปริญญานิพนธ์นี้ได้เสนอระบบการรู้จำวัตถุโดยใช้หลักการทางการประมวลผลภาพต่างๆ เช่นการหาขอบของวัตถุในภาพ การหาจุดศูนย์ถ่วงของวัตถุ เพื่อนำมาเป็นจุดอ้างอิงสำหรับการหาสภาวะปกติของแต่ละวัตถุ ทำให้ภาพที่นำมาเป็นอินพุทมีความยืดหยุ่นต่อขนาดของวัตถุในภาพ หรือการวางตำแหน่งของวัตถุที่ตำแหน่งใดๆและมุมที่แตกต่างกันได้ และทฤษฎีของโครงข่ายประสาทเทียมในการรู้จำวัตถุซึ่งในโครงงานนี้ใช้โครงข่ายแบบ An RCE Network (Restricted Coulomb Energy Network)

ระบบรู้จำจะประกอบด้วยสามส่วนคือ ส่วนของการรับภาพ ส่วนของการดึงลักษณะเด่นของวัตถุ และส่วนของการแยะแยะวัตถุ ภาพอินพุทที่ใช้จะเป็นภาพระดับสีเทา 256 ระดับสีเป็นภาพของวัตถุที่วางบนพื้นฉากสีขาว ในการหาลักษณะของวัตถุจะทำการแบ่งวัตถุออกเป็นส่วนๆ แล้วเก็บลักษณะเฉพาะต่างๆของแต่ละส่วนมาจัดเรียงเป็นเวคเตอร์คงที่ โดยเวคเตอร์นี้จะใช้เป็นอินพุทของโครงข่ายประสาทเทียมซึ่งเป็นส่วนการแยกแยะวัตถุ นอกจากนี้ยังได้ทำการทดสอบผลลัพธ์กับหลายๆวัตถุอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Object Recognition From Shadow Shape

Surasak Tubthiang

Surasit Tubthiang

Assoc. Prof. Dr. Chom Kimpan Advisor

ABSTRACT

This thesis to introduce Object Recognition System by use theories of Image Processing such as Edge Detection, finding Centroid Point, Which used to reference point for normalizing each object. Input image have flexible to scale of object in image or putting object on another positions and difference degrees of object in the input image and Use the theory of neuron network to recognition, In this thesis use An RCE Network (Restricted Coulomb Energy Network).

Recognition System can separate to three sections: Transducer section ,Feature Extraction section and Pattern Recognition section. The format of input images must are Gray Scale 256 levels and have a object on the white background. In feature extraction will separate object to 12 sectors then find features of each sector into ordered feature vector of fixed length. This feature vector is then used as an input to the classifier. Classifier can be done by neural network. Experimental results for recognition of several objects are also presented.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้คงไม่อาจเสร็จได้ด้วยดี หากไม่ได้รับความช่วยเหลือ และร่วมมือจากหลาย ๆ ฝ่าย ด้วยกัน บุคคลแรกที่ต้องกล่าวถึงเพราะเป็นส่วนสำคัญที่ทำให้วิทยานิพนธ์นี้เสร็จลงได้ก็คือ อาจารย์ ชม กิ้มปาน อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้ความเอาใจใส่ แนะนำ และช่วยเหลือเสมอมา ซึ่งต้องขอขอบพระคุณเป็นอย่างมาก

และต้องขอขอบพระคุณบุคคลสำคัญที่สุดที่ทำให้ข้าพเจ้ามีวันนี้ ก็คือ บิดา มารดา อันเป็นที่เคารพรักยิ่ง ซึ่งได้เลี้ยงดูผู้เขียนมาเป็นอย่างดี พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ และยังให้กำลังใจ เอาใจใส่เสมอมา ในทุก ๆ ด้านอันหาที่เปรียบมิได้ ข้าพเจ้าขอระลึกในพระคุณอันสุดประมาณ และขอกราบขอบพระคุณมา ณ ที่นี้

สุรศักดิ์ ทับเที่ยง

สุรสิทธิ์ ทับเที่ยง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูปภาพ	VI
สารบัญตาราง	IX
บทที่ 1 บทนำ	1
1.1 ความสำคัญและที่มา	1
1.2 วัตถุประสงค์ของโครงงาน	2
1.3 ขอบเขตของโครงงาน	2
1.4 วิธีดำเนินงาน	2
บทที่ 2 ทฤษฎีการประมวลผลภาพ	3
2.1 การเตรียมข้อมูลภาพ	3
2.1.1 การทำเทรสโฮล	3
2.1.2 การหาขอบของวัตถุ	3
2.1.3 การหาเส้นทางของขอบ	4
2.2 การประมวลผลภาพ	5
2.2.1 การหาจุดศูนย์ถ่วงของวัตถุ	5
2.2.2 การหาจุดอ้างอิงของขอบวัตถุ	5
2.2.3 การหามุมระหว่างจุดสองจุด	5
2.2.4 การทำฮิสโทรแกรม	6
2.3 การแสดงผลภาพ	7
2.3.1 การย้ายตำแหน่ง	7
2.3.2 การเปลี่ยนขนาด	7
2.3.3 การหมุนวัตถุ	8
บทที่ 3 การดึงลักษณะเด่น	9
3.1 ขั้นตอนการเตรียมข้อมูล	9
3.1.1 ลักษณะของภาพที่ใช้ในการทดลอง	9
3.1.2 แปลงภาพที่รับเข้ามาเป็นภาพขอบ	9
3.2 ขั้นตอนการดึงและทำนอร์มอลไลซ์ลักษณะเด่นของวัตถุ	10
3.2.1 ขั้นตอนการดึงลักษณะเด่น	10
3.2.2 การทำนอร์มอลไลซ์ลักษณะเด่นของวัตถุ	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4 ระบบรู้จำ 15
4.1 โหมดการเรียนรู้ 16
4.2 โหมดการแยกชนิดของวัตถุ 21
บทที่ 5 สรุปและผลการทดลอง 22
5.1 การทดลอง 22
5.2 การทดลองที่ 1 22
5.3 การทดลองที่ 2 26
5.4 การทดลองที่ 3 27
5.5 ปัญหาที่พบและการแก้ไข 29
บทที่ 6 ข้อเสนอแนะ 32
6.1 ระบบการรู้จำวัตถุจากภาพเงาโครงร่าง 32
6.2 ข้อสรุปและเปรียบเทียบของการรู้จำ 32
6.3 ข้อจำกัดของระบบ 32
6.4 แนวทางในการพัฒนาต่อ 32
6.5 การนำไปประยุกต์ใช้งาน 33
ภาคผนวก 35
ภาคผนวก ก ตัวแปรที่ใช้ในโปรแกรม 36
ภาคผนวก ข Flow Chart 37
ภาคผนวก ค การทำงานของโปรแกรมในส่วนการทำงานย่อย 40
ภาคผนวก ง การทำงานของโปรแกรมในส่วนการทำงานหลัก 48
ภาคผนวก จ การใช้งานโปรแกรม 61
บรรณานุกรม 72

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูปที่ 1-1	แสดงส่วนประกอบของระบบ	1
รูปที่ 2-1	แสดงหมายเลขทิศทางของจุดขอบ	4
รูปที่ 2-2	แสดงการหาจุดอ้างอิงของขอบ	5
รูปที่ 2-3	แสดงการทำฮิสโทรแกรมของจำนวนจุดขอบ	6
รูปที่ 2.4	แสดงการทำฮิสโทรแกรมของจำนวนจุดพื้นที่	6
รูปที่ 2-5	แสดงการหมุนจุดรอบจุดรอบจุดกำเนิด	8
รูปที่ 3-1	แสดงภาพโครงร่างวัตถุที่รับเข้ามาใช้งาน	9
รูปที่ 3-2	แสดงภาพสองระดับ	9
รูปที่ 3-3	แสดงภาพขอบของวัตถุ	10
รูปที่ 3-4	แสดงขั้นตอนการหาลักษณะเด่น	10
รูปที่ 3-5	แสดงตำแหน่งจุดศูนย์ถ่วงของประแจ	11
รูปที่ 3-6	แสดงตำแหน่งของจุดอ้างอิงของขอบ	11
รูปที่ 3-7	แสดงลักษณะทางตรรกะของวัตถุ	13
รูปที่ 3-8	แสดงลักษณะทางตรรกะของวัตถุ	13
รูปที่ 3-9	แสดงลักษณะทางตรรกะของวัตถุ	14
รูปที่ 3-10	แสดงลักษณะทางตรรกะของวัตถุ	14
รูปที่ 4-1	แสดงโครงร่างของ RCE Network	15
รูปที่ 4-2	แสดงการเพิ่มโหนดชั้นซ่อน และโหนดชั้นเอาท์พุท (Class) แรก	16
รูปที่ 4-3	แสดงการเพิ่มโหนดชั้นซ่อน และโหนดชั้นเอาท์พุท (Class) แรก	16
รูปที่ 4-4	แสดงการเพิ่มโหนดชั้นซ่อน และโหนดชั้นเอาท์พุท สำหรับแสดงถึงชนิดที่ 1	17
รูปที่ 4-5	แสดงการเพิ่มโหนดชั้นซ่อน และโหนดชั้นเอาท์พุท สำหรับแสดงถึงชนิดที่ 1	17
รูปที่ 4-6	แสดงการเพิ่มโหนดชั้นซ่อน และโหนดชั้นเอาท์พุท สำหรับแสดงถึงชนิดที่ 2	18
รูปที่ 4-7	แสดงการเพิ่มโหนดชั้นซ่อน และโหนดชั้นเอาท์พุท สำหรับแสดงถึงชนิดที่ 2	18
รูปที่ 4-8	แสดงการเพิ่มโหนดชั้นซ่อน และโหนดชั้นเอาท์พุท และการลดรัศมีสำหรับแสดงถึงชนิดที่ 2	19
รูปที่ 4-9	แสดงการเพิ่มโหนดชั้นซ่อน และโหนดชั้นเอาท์พุท และการลดรัศมีสำหรับแสดงถึงชนิดที่ 2	19
รูปที่ 4-10	แสดงการสร้างโหนดชั้นซ่อนขึ้นมาใหม่	20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4-11	แสดงการลดรัศมีของโหนดชั้นซ่อน	20
รูปที่ 5-1	แสดงตัวอย่างภาพวัตถุที่ใช้ในการสอนระบบ	23
รูปที่ 5-2	แสดงตัวอย่างภาพคำถามมุมต่างๆ	23
รูปที่ 5-3	แสดงผลการตั้งค่าความยืดหยุ่นของจุดอ้างอิง เท่ากับ 1.0	24
รูปที่ 5-4	แสดงผลการตั้งค่าความยืดหยุ่นของจุดอ้างอิง เท่ากับ 0.99	24
รูปที่ 5-5	แสดงผลการตั้งค่าความยืดหยุ่นของจุดอ้าง เท่ากับ 0.98 และ 0.97	25
รูปที่ 5-6	แสดงตัวอย่างภาพที่เพิ่มจากการทดลองที่ 1	26
รูปที่ 5-7	แสดงตัวอย่างวัตถุที่มีลักษณะคล้ายๆกัน	28
รูปที่ 5-8	แสดงภาพ Capsule ที่สอนให้ระบบ	30
รูปที่ 5-9	แสดงภาพคำถาม Capsule	30
รูปที่ 6-1	แสดงภาพที่มีอักษรมีลักษณะทั้งในแนวนอนและแนวตั้ง	33
รูปที่ 6-2	แสดงวัตถุซึ่งมีลวดลาย	33
รูปที่ 6-3	แสดง Left image และ Right image	34
รูปที่ 6-4	แสดงโครงร่างของวัตถุ	34
รูปที่ 1ข	แสดง Flow Chart ของการหาเวคเตอร์ลักษณะเด่นของวัตถุ	37
รูปที่ 2ข	แสดง Flow Chart ของการแยกชนิดวัตถุ	38
รูปที่ 3ข	แสดง Flow Chart ของการสอน	39
รูปที่ 1จ	แสดงเมนู Load Image	62
รูปที่ 2จ	แสดงทูลบาร์ Image	62
รูปที่ 3จ	แสดงไดอะล็อกการเปิดภาพ	62
รูปที่ 4จ	แสดงผลการประมวลผลภาพ	63
รูปที่ 5จ	แสดงเมนูTraining	63
รูปที่ 6จ	แสดงทูลบาร์ Train	63
รูปที่ 7จ	แสดงไดอะล็อกรับชนิดของวัตถุ	64
รูปที่ 8จ	แสดงข้อความเตือนหากป้อนอักขระเกิน 40 ตัว	64
รูปที่ 9จ	แสดงลักษณะเด่นที่ใช้สอน	64
รูปที่ 10จ	แสดงเมนู Classification	65
รูปที่ 11จ	แสดงทูลบาร์ Class	65
รูปที่ 12จ	แสดงผลของการแยกชนิดวัตถุ	65
รูปที่ 13จ	แสดงลักษณะที่ใช้ในการแยกชนิดวัตถุ	66
รูปที่ 14จ	แสดงเมนู Save as	66
รูปที่ 15จ	แสดงทูลบาร์ Sv As	66
รูปที่ 16จ	แสดงไดอะล็อกการบันทึกไฟล์ลักษณะของวัตถุ	67
รูปที่ 17จ	แสดงเมนู Load Data	67

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 18จ	แสดงทูลบาร์ Load	67
รูปที่ 19จ	แสดงไดอะล็อกการเปิดไฟล์ลักษณะของวัตถุ	68
รูปที่ 20จ	แสดงเมนู Save Data	68
รูปที่ 21จ	แสดงทูลบาร์ Save	68
รูปที่ 22จ	แสดงข้อความถามยืนยันการบันทึกการเปลี่ยนแปลง	69
รูปที่ 23จ	แสดงเมนู Show Network	69
รูปที่ 24จ	แสดงทูลบาร์ Show	69
รูปที่ 25จ	แสดงไดอะล็อกแสดงข้อมูลภายในโครงข่าย	70
รูปที่ 26จ	แสดงเมนู Clear Network	70
รูปที่ 27จ	แสดงทูลบาร์ Clear	70
รูปที่ 28จ	แสดงข้อความยืนยันการลบข้อมูลในโครงข่าย	71

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

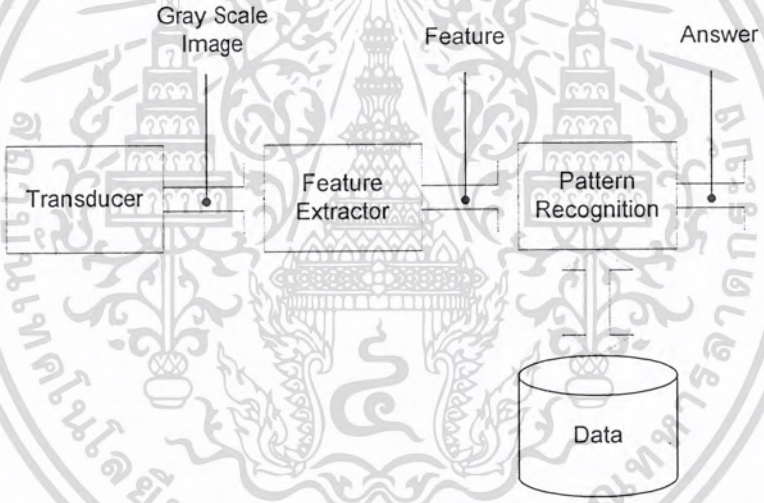
ตารางที่ 5-1	แสดงผลการทดลองที่ 1	22
ตารางที่ 5-2	แสดงผลการทดลองที่ 2	27
ตารางที่ 5-3	แสดงผลการทดลองที่ 3	27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มา

การใช้ทฤษฎีโครงข่ายประสาทเทียม (Neural Network) และทฤษฎีทางคณิตศาสตร์ต่างๆมาช่วยทำให้คอมพิวเตอร์สามารถแยกแยะความเหมือนและความแตกต่างของวัตถุต่างๆ ได้มีการพัฒนาวิจัยกันมาอย่างต่อเนื่อง ในโครงงานนี้ก็ได้นำเอาทฤษฎีต่างๆมาประยุกต์ใช้เพื่อให้คอมพิวเตอร์สามารถรู้จำวัตถุได้ โครงงานนี้ตัวระบบรู้จำจะรับภาพเข้ามา โดยภาพที่รับเข้ามานี้จะเป็นภาพระดับสีเทา (Gray Scale Image) 256 สี ที่มีรูปของวัตถุอยู่บนพื้นฉากสีขาว จากนั้นระบบจะนำภาพอินพุทนี้ไปประมวลผลแยกแยะลักษณะต่างๆที่ได้จากภาพแล้วสามารถจดจำหรือแยกแยะวัตถุได้โดยไม่ขึ้นกับความแปรผันจาก ขนาด (Scale) ตำแหน่ง (Translation) และการหมุน (Rotation) ของวัตถุในลักษณะของการวางที่ต่างกันได้



รูปที่ 1-1 แสดงส่วนประกอบของระบบ

ในโครงงานนี้แบ่งระบบการรู้จำวัตถุออกเป็นสามส่วน คือ ระบบการรับภาพ (Transducer) ระบบการดึงลักษณะเด่นของภาพออกมา (Feature Extractor) และ ระบบการแยกแยะวัตถุ (Classification) ดังรูปที่ 1-1

ส่วนที่หนึ่งคือระบบการรับภาพ โดยภาพที่จะนำมาประมวลผลจะเป็นภาพระดับสีเทา 256 สีที่มีรูปของวัตถุอยู่บนพื้นฉากสีขาว แล้วนำภาพนี้ส่งต่อไปยังส่วนถัดไป

ส่วนที่สองคือระบบการดึงลักษณะเด่นของภาพวัตถุ โดยในส่วนนี้จะแบ่งออกได้เป็นสองส่วนคือ การเตรียมข้อมูลภาพ และการประมวลผลภาพ ซึ่งในแต่ละส่วนก็จะมีฟังก์ชั่นการทำงานดังนี้ ส่วนของการเตรียมข้อมูลภาพ ประกอบด้วย การทำเทรสโฮล (Thresholding) การทำเซกเมนเทชั่น (Segmentation) การหาขอบของวัตถุ (Edge Detection) และการหาเส้นทางของขอบ (Chain Coding) ส่วนของการประมวลผลภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประกอบด้วย การหาจุดศูนย์ถ่วงของวัตถุ (Centroid) การหาจุดอ้างอิงของขอบวัตถุ (Base Point) และการหาพื้นที่ของแต่ละส่วนของวัตถุ

ส่วนที่สามเป็นส่วนการแยกแยะวัตถุ ซึ่งได้นำทฤษฎีโครงข่ายประสาทเทียมมาใช้ในการเรียนรู้เชื่อมโยงไปหาคำตอบ

1.2 วัตถุประสงค์ของโครงงาน

โครงงานนี้ทำขึ้นเพื่อศึกษาการประมวลผลภาพ และทฤษฎีโครงข่ายประสาทเทียมเพื่อประยุกต์ประกอบกันทำให้คอมพิวเตอร์สามารถรู้จำวัตถุได้จากลักษณะรูปร่างของวัตถุต่างๆ โดยให้มีความผิดพลาดน้อยและมีความยืดหยุ่นต่อระบบการมองเห็นของคอมพิวเตอร์ (Machine Vision) โดยมีวัตถุประสงค์ดังนี้

1. เพื่อศึกษาการประมวลผลภาพและทฤษฎีโครงข่ายประสาทเทียมพร้อมทั้งสร้างระบบรู้จำวัตถุได้
2. เพื่อศึกษาและพัฒนาระบบการมองเห็นของคอมพิวเตอร์ให้มีความยืดหยุ่นในการรับภาพเข้ามาโดยไม่ขึ้นกับขนาดของวัตถุในภาพอันเนื่องมาความแปรผันจากระยะของการมอง ตำแหน่งและ การหมุนของภาพวัตถุในองศาต่างๆ
3. เพื่อศึกษาข้อดีและข้อเสียของการรู้จำโดยใช้ภาพเงาโครงร่างของวัตถุและการนำมาประยุกต์ใช้กับโครงข่ายประสาทเทียม

1.3 ขอบเขตของโครงงาน

ผลที่ได้รับจากโครงงานจะเป็นระบบที่เกิดจากการนำเอาหลักการของการประมวลผลภาพ และระบบโครงข่ายประสาทเทียมมาประยุกต์สร้างเป็นระบบรู้จำวัตถุ โดยมีความถูกต้องไม่น้อยกว่า 90% พร้อมทั้งให้รู้ถึงสาเหตุข้อผิดพลาด การแก้ปัญหานั้นๆ ข้อจำกัดของระบบ แนวทางการพัฒนา และประโยชน์ของโครงงานตลอดจนการนำไปประยุกต์ใช้

1.4 วิธีดำเนินงาน

โครงงานนี้แบ่งการดำเนินงานออกเป็น 2 ช่วง คือเริ่มด้วยการศึกษาทฤษฎีพื้นฐานต่างๆเกี่ยวกับการประมวลผลภาพ เช่นการหาขอบของภาพ การหาจุดศูนย์ถ่วง เป็นต้น และศึกษาขั้นตอนการดึงข้อมูลต่างๆของลักษณะของวัตถุออกมาเพื่อใช้ในการเปรียบเทียบความเป็นวัตถุนั้น ทดลองและนำข้อมูลที่ต้องการออกมาให้ได้จริง จากนั้นส่วนที่สองจะเป็นส่วนที่เกี่ยวข้องกับการรู้จำ โดยในที่นี้ได้ศึกษาทฤษฎีโครงข่ายประสาทเทียมเพื่อนำมาใช้ในการเปรียบเทียบข้อมูลที่ได้จากส่วนการดำเนินงานของส่วนแรก โดยได้นำโครงข่ายประสาทเทียมแบบ An RCE Network (Restricted Coulomb Energy Network) มาใช้ โดยทดลองนำข้อมูลที่ได้มาจากการดึงลักษณะของวัตถุใส่ให้แก่โครงข่าย เพื่อให้ระบบทำการแยกชนิดของวัตถุ เพื่อดูความถูกต้องและวิเคราะห์หาสาเหตุของปัญหาที่เกิดขึ้น แล้วพัฒนาระบบเพื่อกำจัดข้อผิดพลาด พร้อมทั้งปรับปรุงให้ระบบใช้งานได้อย่างมีประสิทธิภาพมากขึ้น สุดท้ายเป็นการสรุปและประเมินผลที่ได้ รวมทั้งแสดงการประยุกต์ใช้งานและแนวทางในการทำการพัฒนาต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีการประมวลผลภาพ

2.1 การเตรียมข้อมูลภาพ (Image Preprocessing)

2.1.1 การทำเทรสโฮล

เป็นการลดจำนวนระดับความแตกต่าง สำหรับในโครงงานนี้จะทำการลดระดับสีจาก 256 สีให้เหลือเพียง 2 ระดับ คือสีที่ 0(สีดำ) แทนจุดภาพของวัตถุ และสีที่ 1(สีขาว) แทนส่วนอื่นของภาพ เพื่อจำกัดสัญญาณรบกวนต่างๆ เช่น ส่วนของเงาที่ไม่ต้องการและแสดงส่วนที่เป็นเนื้อวัตถุเท่านั้น ซึ่งทฤษฎีของการทำเทรสโฮลแบ่งได้เป็นสองแบบ ดังนี้

1. Strong Cut

$$\mu(x)=\begin{cases} f(x); ถ้า \mu(x) \geq \alpha \\ 0; ในกรณีอื่นๆ \end{cases}$$

2. Weak Cut

$$\mu(x)=\begin{cases} f(x); ถ้า \mu(x) > \alpha \\ 0; ในกรณีอื่นๆ \end{cases}$$

2.1.2 การหาขอบของวัตถุ

การหาขอบของวัตถุ เป็นการหารูปร่าง (Shape) ของวัตถุในภาพซึ่งเป็นลักษณะเฉพาะ (Feature) สำคัญที่จะนำมาใช้ในการรู้จำวัตถุ ในที่นี้เราจะใช้ตัวดำเนินการของลาปาส (Laplacian Operator) เพราะง่ายและมีความถูกต้องในการใช้งาน

ตัวดำเนินการของลาปาส

$$\nabla^2 f=\frac{\partial^2 f}{\partial x^2}+\frac{\partial^2 f}{\partial y^2}$$

$$\Delta_x f(x,y)=f(x,y)-f(x-1,y)$$

$$\Delta_y f(x,y)=f(x,y)-f(x,y-1)$$

ทำการดิฟเฟอร์เรนเชียล อันดับที่สอง จะได้

$$\begin{aligned}\Delta_x^2 f(x,y)&=\Delta_x f(x+1,y)-\Delta_x f(x,y)\\&=[f(x+1,y)-f(x,y)]-[f(x,y)-f(x-1,y)]\\&=f(x+1,y)+f(x,y-1)-2f(x,y)\end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในลักษณะเดียวกัน

$$\Delta_y^2 f(x,y) = f(x+1,y) + f(x,y-1) - 2f(x,y)$$

$$\therefore \nabla^2 f(x,y)$$
$$= \left[f(x,y) + f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1)\right] - 5f(x,y)$$
$$= f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y)$$

ซึ่งแปลงเป็นตารางกรองแล้วจะได้ดังนี้

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

หรือ

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

2.1.3 การหาเส้นทางของขอบ

รูปที่ 2-1 แสดงหมายเลขทิศทางของจุดขอบ

การหาเส้นทางของขอบเพื่อประโยชน์ในการเก็บ จะทำให้ลดหน่วยความจำในการเก็บลงได้ แทนที่จะเก็บบิทแมพ (Bitmap)ของวัตถุ ก็จะเก็บทิศทางของจุดขอบของวัตถุแทน ดังแสดงในรูปที่ 2-1 นอกจากนี้ข้อมูลของจุดขอบนี้ จะสามารถนำมาใช้งานได้สะดวกรวดเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 การประมวลผลภาพ (Image Processing)

2.2.1 การหาจุดศูนย์ถ่วงของวัตถุ

เป็นการหาตำแหน่งกึ่งกลางของวัตถุเพื่อใช้เป็นตำแหน่งอ้างอิง สามารถหาจุดศูนย์ถ่วง(Centroid) ได้โดยสมการ

$$x_c = \frac{1}{N_b} \sum_{i \in o_b} x_i$$

$$y_c = \frac{1}{N_b} \sum_{i \in o_b} y_i$$

N_b คือจำนวนจุดของวัตถุ

O_b ส่วนที่เป็นวัตถุ

2.2.2 การหาจุดอ้างอิงของขอบวัตถุ

จุดที่ใช้ในการอ้างอิงของจุดขอบวัตถุหาได้จาก คือจุดขอบซึ่งมีระยะห่างจากจุดศูนย์ถ่วงมากที่สุด ซึ่งสามารถหาทำได้ดังรูปที่ 2-2 และจากสมการของการหาระยะห่างระหว่างจุดสองจุดเป็นดังนี้

$$D = \sqrt{(x_c - x_i)^2 + (y_c - y_i)^2}$$

รูปที่ 2-2 แสดงการหาจุดอ้างอิงของขอบวัตถุ

2.2.3 การหามุมระหว่างจุดสองจุด

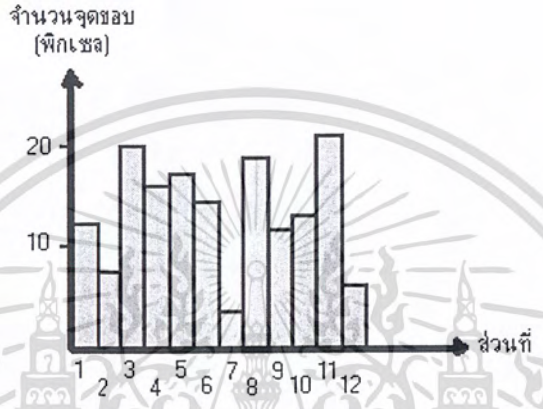
คือการหามุมระหว่างเวกเตอร์ของแกนอ้างอิงกับเวกเตอร์ของจุดนั้นๆโดยใช้จุดจุดศูนย์ถ่วงเป็นจุดหมุน เพื่อใช้ในการบอกทิศทางของจุดนั้นๆ ซึ่งหาได้จากสมการ ดังนี้

$$\cos\theta = \frac{A_1 \bullet A_2}{|A1||A2|}$$

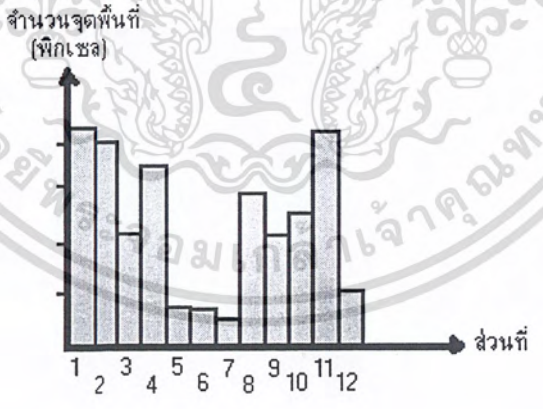
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 การทำฮิสโทรแกรม

จากการประมวลผลที่ผ่านมาเมื่อได้จุดและแกนอ้างอิงต่างๆแล้ว ก็เป็นส่วนของการดึงลักษณะเฉพาะของแต่ละวัตถุออกมาใช้ในการจดจำ โดยในโครงงานนี้ได้นำหลักการของฮิสโทรแกรมมาใช้ ฮิสโทรแกรมเป็นวิธีเก็บข้อมูลแบบสถิติซึ่งเป็นการเก็บข้อมูลด้านความถี่ในที่นี้ได้ใช้ฮิสโทแกรมเก็บข้อมูล 2 ลักษณะจากทั้งหมด 4 ลักษณะที่ใช้ในโครงงานนี้คือ จำนวนจุดขอบในแต่ละส่วน แสดงในรูปที่ 2-3 และจำนวนจุดพื้นที่ในแต่ละส่วน แสดงในรูปที่ 2-4



รูปที่ 2-3 แสดงการทำฮิสโทรแกรมของจำนวนจุดขอบ



รูปที่ 2-4 แสดงการทำฮิสโทแกรมของจำนวนจุดพื้นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การแสดงผลภาพ

2.3.1 การย้ายตำแหน่ง

เราสามารถย้ายจุดในระบบพิกัดฉาก (x,y) ไปยังตำแหน่งใหม่โดยการบวกระยะที่จะย้ายกับพิกัดของจุดเดิม สำหรับจุด $P(x,y)$ ที่ย้ายโดย d_x หน่วยจะเคลื่อนที่ขนานกับแกน x และเคลื่อนที่ขนานทางแกน y โดย d_y หน่วย ได้จุดใหม่เป็น $P'(x',y')$ สามารถเขียนเป็นสมการได้ดังนี้

$$x' = x + d_x$$
$$y' = y + d_y$$

เมื่อนิยามเป็น เวคเตอร์ในแนวตั้ง (Column Vectors)จะได้

$$P = \begin{bmatrix} x \\ y \end{bmatrix}$$
$$P' = \begin{bmatrix} x' \\ y' \end{bmatrix}$$
$$T = \begin{bmatrix} d_x \\ d_y \end{bmatrix}$$

สรุปเป็นสมการ ได้ดังนี้

$$P' = P + T$$

2.3.2 การเปลี่ยนขนาดวัตถุ

จุดต่างๆสามารถที่จะย่อหรือขยายได้โดย s_x เป็นการปรับขนาดทางแนวแกน x และโดย s_y เป็นการปรับนาดทางแกน y

$$x' = s_x \bullet x$$
$$y' = s_y \bullet y$$

เขียนในรูปแบบของเมทริกได้

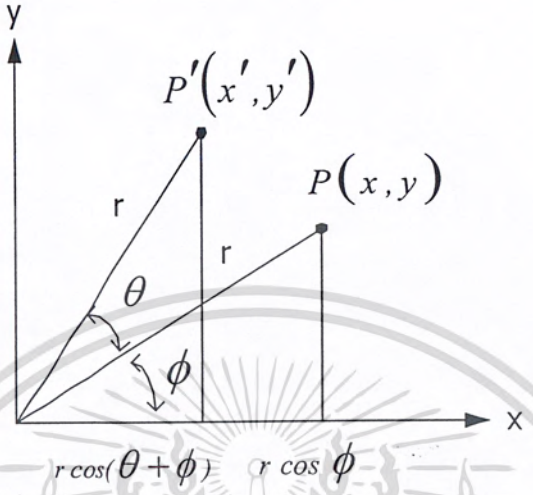
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \bullet \begin{bmatrix} x \\ y \end{bmatrix}$$

$$P' = S \bullet P$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 การหมุนวัตถุ

เราสามารถที่จะหมุนจุดต่างๆเทียบกับจุดกำเนิด (Origin Point) เป็นมุมเท่ากับ θ ด้วยสมการคณิตศาสตร์ดังแสดงในรูปที่ 2-5



รูปที่ 2-5 แสดงการหมุนจุดรอบจุดกำเนิด

$$x = r \bullet \cos\phi, y = r \bullet \sin\phi$$
$$x' = r \bullet \cos(\phi+\theta) = r \bullet \cos\phi \bullet \cos\theta - r \bullet \sin\phi \bullet \sin\theta$$
$$y' = r \bullet \sin(\phi+\theta) = r \bullet \cos\phi \bullet \sin\theta + r \bullet \sin\phi \bullet \cos\theta$$

ทำการลดรูปได้

$$x' = x \bullet \cos\theta - y \bullet \sin\theta$$
$$y' = x \bullet \sin\theta + y \bullet \cos\theta$$

เมื่อเขียนจัดรูปเป็นเมทริกและสรุปเป็นสมการได้ดังนี้

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \bullet \begin{bmatrix} x \\ y \end{bmatrix}$$

$$P' = R \bullet P$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การดึงลักษณะเด่น

3.1 ขั้นตอนการเตรียมข้อมูลภาพ

3.1.1 ลักษณะของภาพที่ใช้ในการทดลอง

ลักษณะของภาพที่ใช้ในการทดลอง เป็นภาพถ่ายของวัตถุที่อยู่บนพื้นฉากสีขาว แล้วแปลงเป็นภาพระดับสีขาวเทา 256 ระดับ โดยสีที่ 0 จะเป็นสีที่ดำสุดแล้วไล่เฉดสีความสว่างตามระดับไปจนถึงสีที่ระดับ 255 จะเป็นสีที่ขาวสุด และรูปที่ใช้กำหนดขนาดไว้ที่ 128*128 จุดภาพ (Pixel) ดังรูปที่ 3-1

รูปที่ 3-1 แสดงภาพเงาโครงร่างวัตถุที่รับเข้ามาใช้ในโครงงาน

3.1.2 แปลงภาพที่รับเข้ามาเป็นภาพขอบ

เริ่มจากนำภาพระดับสีขาวเทา 256 ระดับสีที่รับเข้ามาแปลงให้เป็นภาพ 2 ระดับสี คือสีดำเป็นสีที่ 0 แทนจุดของวัตถุ และสีขาวเป็นสีที่ 1 แทนฉาก จากนั้นจะเป็นการนำเอาภาพวัตถุที่เป็นภาพ 2 ระดับสีมาทำการหาขอบของวัตถุโดยใช้ ตารางกรองขอบตามทฤษฎีของลาปาส แสกนกรองจุดภาพทั้งหมด ตั้งแต่มุมบนซ้ายไปจนถึงมุมล่างขวา ซึ่งตารางกรองขอบจะมีลักษณะดังนี้

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

ซึ่งภาพสองระดับของวัตถุ แสดงในรูปที่ 3-2 และภาพขอบของวัตถุ แสดงในรูปที่ 3-3

รูปที่ 3-2 แสดงภาพสองระดับ

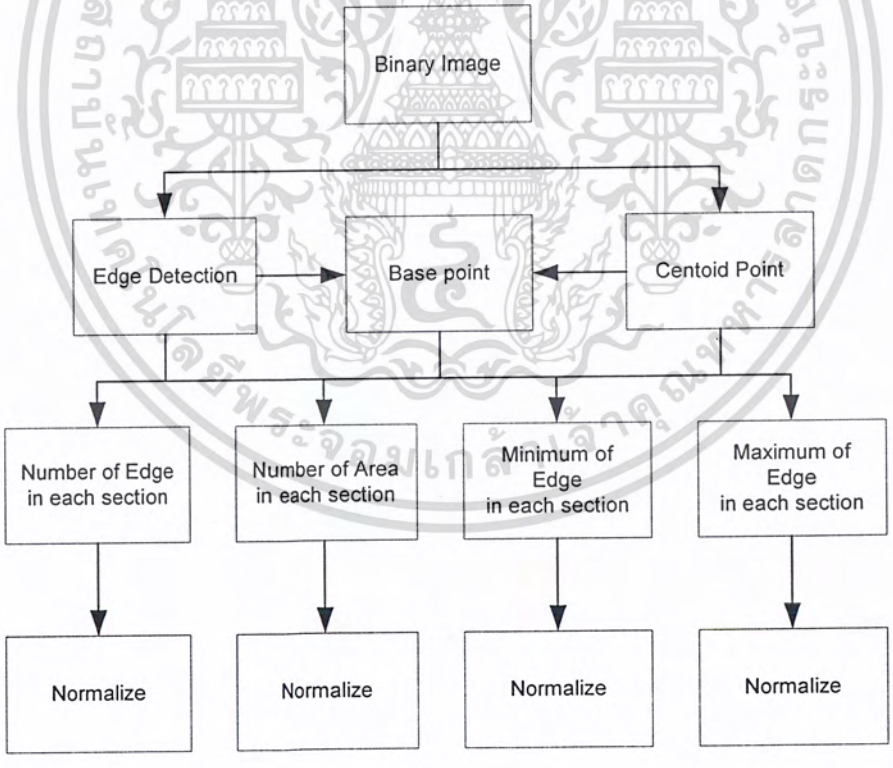
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3-3 แสดงภาพขอบของวัตถุ

3.2 ขั้นตอนการดึงและทำนอร์มอลไลซ์ลักษณะเด่นของวัตถุ

3.2.1 ขั้นตอนการดึงลักษณะเด่น

เมื่อได้ภาพสองระดับและภาพขอบของวัตถุแล้ว จะทำการหาลักษณะต่างๆของวัตถุที่จะใช้ในการกำหนดเป็นลักษณะเฉพาะของแต่ละวัตถุ ในโครงงานนี้จะใช้ลักษณะต่างๆ 4 ตัวคือ จำนวนของขอบในแต่ละส่วน จำนวนพื้นที่ในแต่ละส่วน ระยะขอบที่อยู่ใกล้และไกลที่สุดที่เทียบกับจุดศูนย์กลางของวัตถุในแต่ละส่วน โดยแต่ละส่วนนี้จะแบ่งตามช่วงมุมองศาช่วงละ 30 องศาได้เป็น 12 ส่วน ดังนั้นจะได้ข้อมูลลักษณะต่างๆเป็น 4*12 = 48 ตัว ที่จะใช้ในการกำหนดเป็นลักษณะของแต่ละวัตถุ ขั้นตอนการหาลักษณะต่างๆดังรูปที่ 3-4



รูปที่ 3-4 แสดงขั้นตอนการหาลักษณะเด่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับขั้นตอนการหาลักษณะต่างๆดังนี้คือ เมื่อได้ภาพสองระดับแล้วก็นำมาใช้ในการหาจุดศูนย์ถ่วงของวัตถุและหาภาพขอบของวัตถุ โดยจุดศูนย์กลางของวัตถุหาได้จากสมการ

$$x_c = \frac{1}{N_b}\sum_{i \in o_b} x_i$$

$$y_c = \frac{1}{N_b}\sum_{i \in o_b} y_i$$

ซึ่งอธิบายตัวแปรไปแล้วในส่วนของทฤษฎี จากสูตรเป็นการหาค่าเฉลี่ยสะสมของค่าตำแหน่งในแนวแกนนอนและแกนตั้ง ซึ่งก็จะได้เป็นตำแหน่งของจุดศูนย์ถ่วงของวัตถุ แสดงดังรูปที่ 3-5

รูปที่ 3-5 แสดงตำแหน่งจุดศูนย์ถ่วงของประแจ

ต่อไปเป็นการหาภาพขอบซึ่งได้อธิบายไปแล้วในหัวข้อการหาภาพขอบ ตอนนี้เมื่อได้จุดขอบทั้งหมดของภาพและจุดศูนย์ถ่วงของภาพแล้ว ก็จะนำข้อมูลที่ได้ทั้งสองนี้มาใช้ในการหาจุดอ้างอิงซึ่งเป็นจุดขอบของวัตถุที่มีระยะห่างจากจุดศูนย์ถ่วงมากที่สุด แสดงดังรูปที่ 3-6

รูปที่ 3-6 แสดงตำแหน่งของจุดอ้างอิงของขอบ

เมื่อได้ลักษณะและจุดอ้างอิงต่างๆครบแล้วต่อไปก็เป็นการหาลักษณะต่างๆ โดยแบ่งวัตถุออกเป็นส่วนตามช่วงขององศาส่วนละ 30 องศาให้จุดศูนย์ถ่วงเป็นจุดศูนย์กลางของพิกัด และให้จุดอ้างอิงเป็นจุดเริ่มที่ 0 องศาของขอบ จากนั้นก็เริ่มหาลักษณะต่างๆ 4 ตัวคือ จำนวนของขอบในแต่ละส่วน จำนวนพื้นที่ในแต่ละส่วน ระยะของขอบที่อยู่ใกล้และไกลที่สุดที่เทียบกับจุดศูนย์ถ่วงของวัตถุในแต่ละส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 การทำนอร์มอลไลซ์ลักษณะเด่นของวัตถุ (Normalization)

การหาค่าต่างๆที่เป็นจุดอ้างอิงแล้วนำมาเปรียบเทียบกับค่าใดๆแล้วทำให้ได้ค่ามาตรฐานเพื่อการประมวลผลที่ได้จะไม่ขึ้นกับการหมุน ตำแหน่งภาพของวัตถุ ขนาดของภาพวัตถุ และเป็นการกำหนดขอบเขตของค่าข้อมูลด้วย

จากขั้นตอนการหาจุดศูนย์ถ่วงเพื่อนำมาเป็นจุดศูนย์กลางก็เป็นการกำหนดตำแหน่งที่ใช้อ้างอิงสำหรับจุดต่างๆในภาพและจุดอ้างอิงของขอบเป็นตัวกำหนดจุดองศาของการแบ่งส่วน

ขั้นตอนนี้เป็นการนำข้อมูลที่ได้จากขั้นตอนการหาลักษณะต่างๆทั้งหมด 36 ตัวมาแปลงให้อยู่ในรูปมาตรฐานค่าที่ได้จะเป็นค่าระหว่าง 0-1 ก่อนที่จะนำไปใช้กำหนดค่าให้แต่ละวัตถุในนิวรอนเน็ทเวิร์ค โดย

1. การแปลงข้อมูลจำนวนขอบในแต่ละส่วนใช้สมการดังนี้

   (จำนวนของจุดขอบในส่วนนั้นๆ) / (จำนวนของจุดขอบในส่วนที่มีจำนวนจุดขอบมากที่สุด)

2. การแปลงข้อมูลจำนวนพื้นที่ในแต่ละส่วนใช้สมการดังนี้

   (จำนวนของพื้นที่ในส่วนนั้นๆ) / (จำนวนของพื้นที่ในส่วนที่มีจำนวนพื้นที่มากที่สุด)

3. การแปลงข้อมูลระยะของจุดขอบที่มากที่สุดในแต่ละส่วนใช้สมการดังนี้

   (ระยะของจุดขอบที่มากที่สุดในส่วนนั้นๆ) / (ระยะของจุดขอบที่มากที่สุดในส่วนที่มีระยะของจุดขอบที่มากที่สุดมากที่สุด)

4. การแปลงข้อมูลระยะของจุดขอบที่น้อยที่สุดในแต่ละส่วนใช้สมการดังนี้

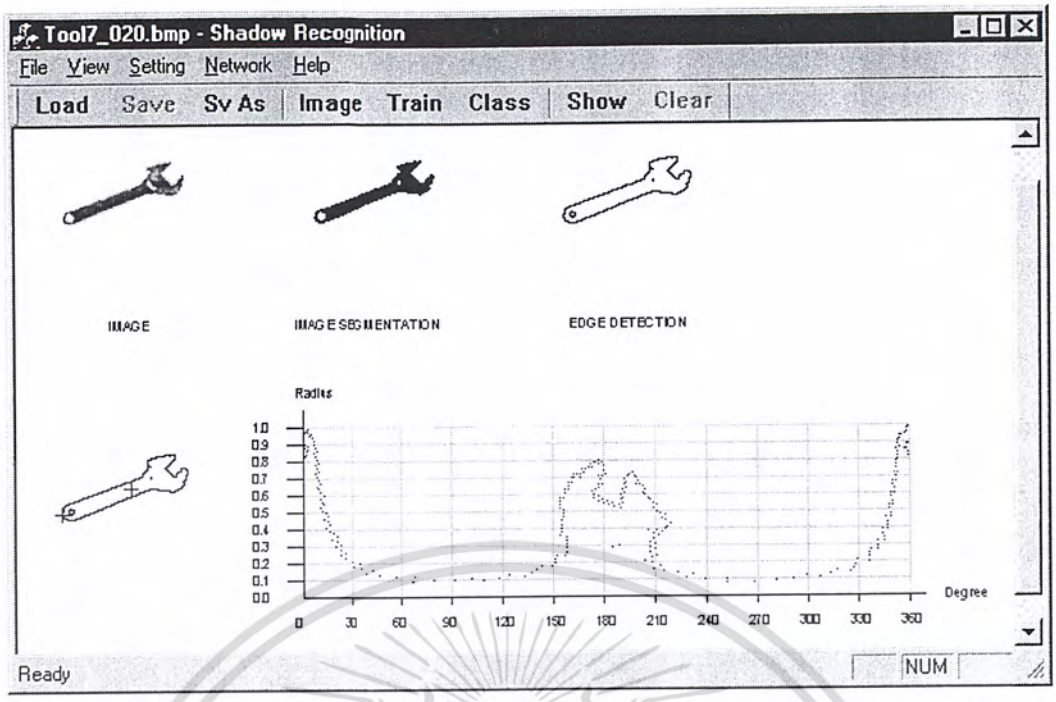
   (ระยะของจุดขอบที่น้อยที่สุดในส่วนนั้นๆ) / (ระยะของจุดขอบที่น้อยที่สุดในส่วนที่มีระยะของจุดขอบที่น้อยที่สุดมากที่สุด)

การดึงและการทำนอร์มอลไลซ์ลักษณะเด่นของวัตถุด้วยวิธีนี้ ทำให้สามารถแก้ปัญหาวัตถุที่มีความแปรผันของตำแหน่งการวาง ด้วยการใช้จุดศูนย์ถ่วงมาเป็นจุดกำเนิดของระบบพิกัดเชิงขั้ว และการแปรผันในเรื่องของขนาดของวัตถุนั้นมีการแก้ด้วยข้อมูลที่ทำนอร์มอลไลซ์ ส่วนการแปรผันจากการหมุนของวัตถุจะแก้โดยวิธีการเรียงข้อมูลสักษณะเด่นตามลำดับซึ่งใช้จุดอ้างอิงเป็นจุดเริ่มต้น

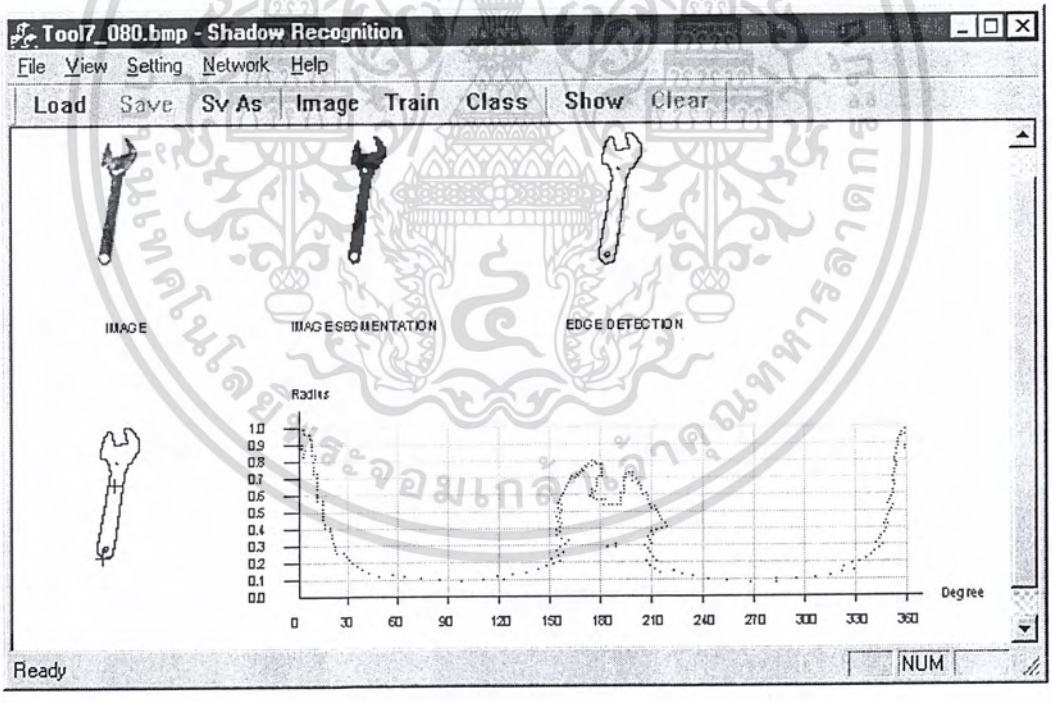
อย่างไรก็ตาม ในกรณีของความบิดเบือนของมุมมอง หรือกรณีที่วัตถุมีจุดที่มีรัศมีมากที่สุดอยู่หลายจุด ปัญหานี้แก้ด้วยการสอน (Train) ระบบรู้จำด้วยแต่ละจุดอ้างอิงที่เป็นไปได้

สังเกตว่าถึงแม้วัตถุจะมีความแปรผันจากการหมุน ขนาด หรือตำแน่ง ในการวางวัตถุแต่ลักษณะทางตรรกะ (Logical) ที่ได้จะยังมีความใกล้เคียงกัน แสดงดังรูปที่ 3-7 ถึงรูปที่ 3-10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

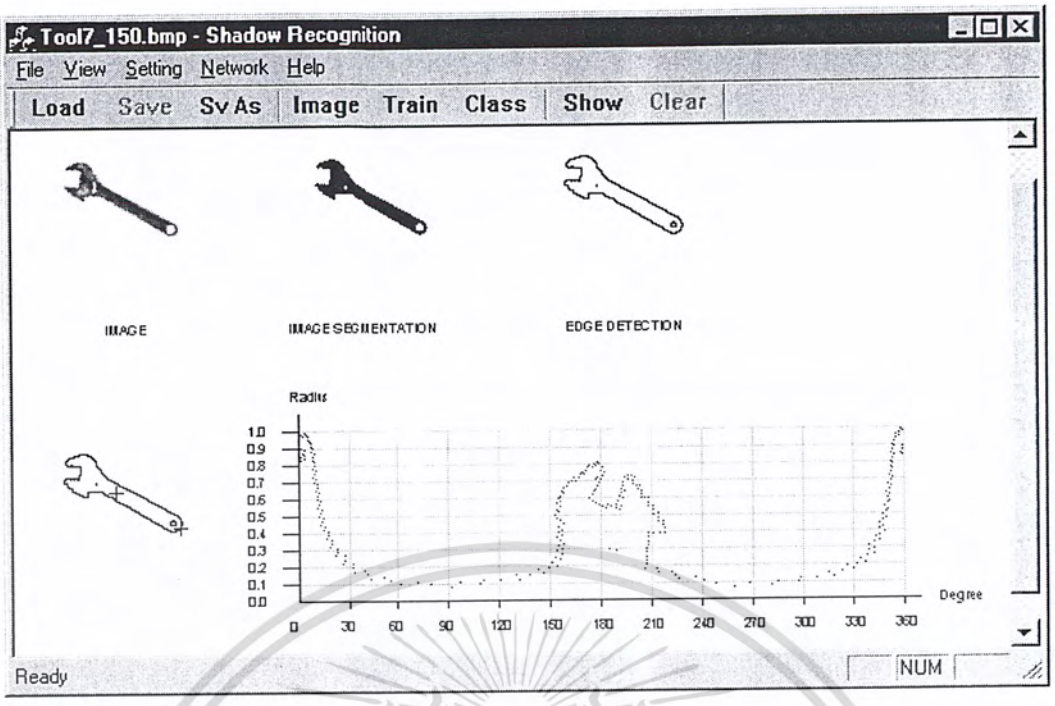


รูปที่ 3-7 แสดงลักษณะทางตรรกะของวัตถุ

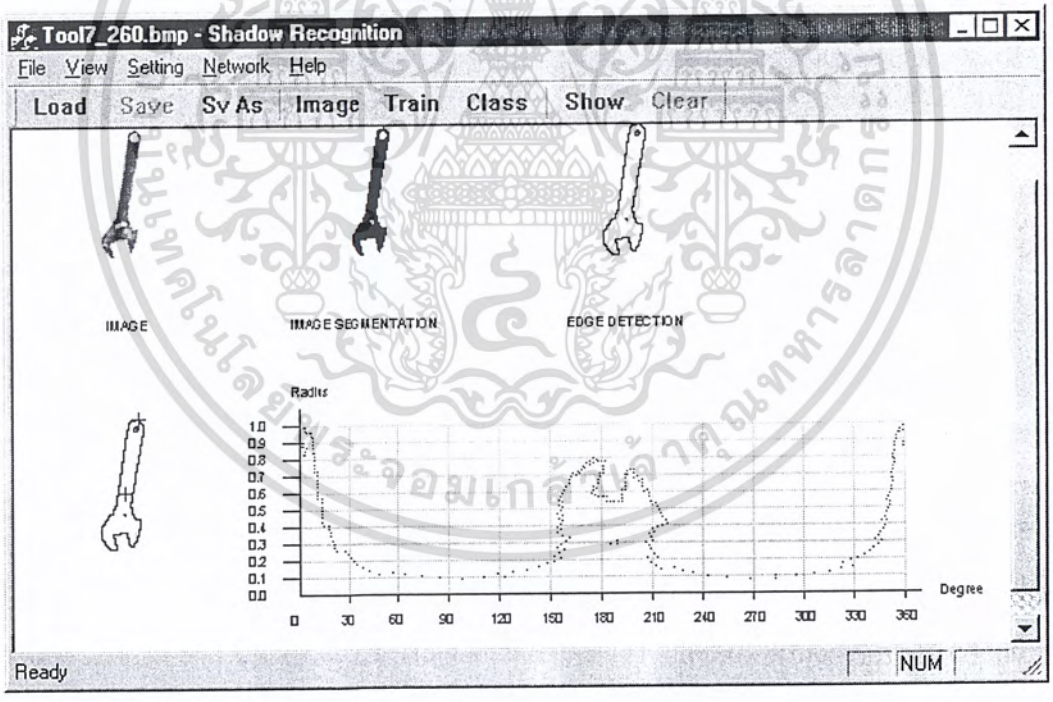


รูปที่ 3-8 แสดงลักษณะทางตรรกะของวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-9 แสดงลักษณะทางตรรกะของวัตถุ

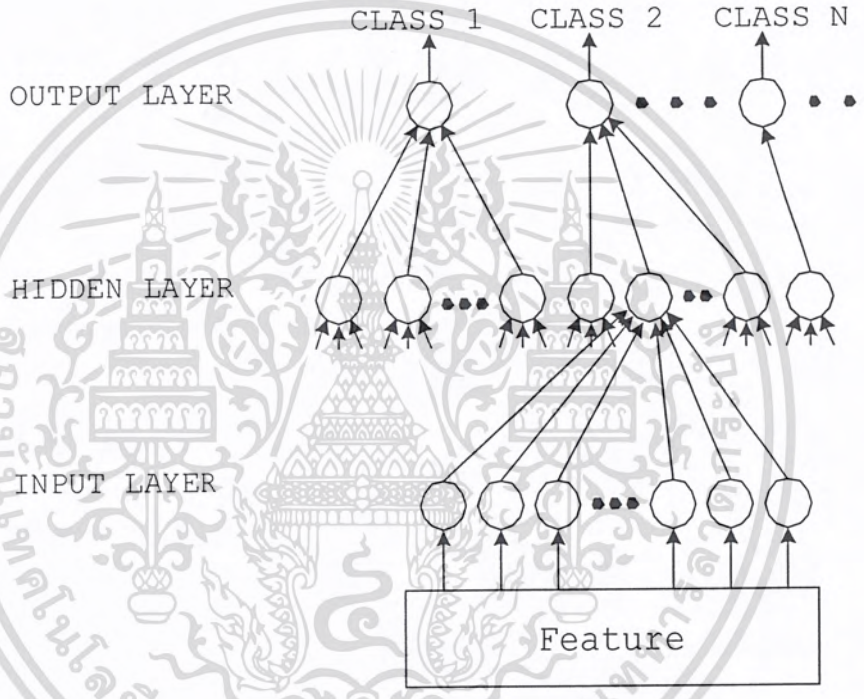


รูปที่ 3-10 แสดงลักษณะทางตรรกะของวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ระบบรู้จำ

หลักของการแยกชนิดของวัตถุ คือการเก็บลักษณะของวัตถุต่างๆไว้ ซึ่งในแต่ละลักษณะนั้นจะใช้บอกถึงชนิดของวัตถุ ในโครงการนี้จะใช้ An RCE Network (Restricted Coulomb Energy Network) ซึ่งประกอบด้วยสามชั้น คือ ชั้นอินพุท (Input Layer) ชั้นซ่อน (Hidden Layer) ชั้นเอาท์พุท (Output Layer) แต่ละชั้นมีการเชื่อมต่อกันเป็นแบบ Feed Forward ดังรูปที่ 4-1



รูปที่ 4-1 โครงสร้างของ An RCE Network

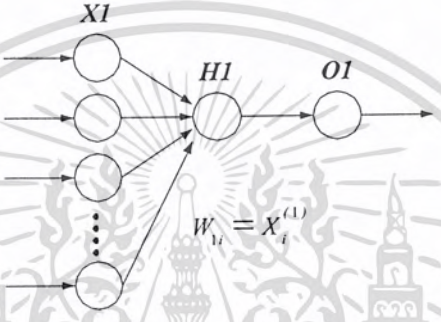
การดำเนินงานของโครงข่าย มีอยู่สองโหมด คือ โหมดการเรียนรู้ (Learning Mode) และโหมดการแยกแยะประเภท (Classification Mode) ซึ่งโหนดในแต่ละชั้นจะมีจำนวนต่างกันดังนี้ คือโหนด (Node) ในชั้นอินพุทมีจำนวนเท่ากับจำนวนของลักษณะที่ถูกดึงมาเป็นอินพุทเวคเตอร์ และจำนวนของโหนดในชั้นเอาท์พุทเท่ากับจำนวนชนิด (Class) ของวัตถุที่ทำการจดจำ ส่วนโหนดในชั้นซ่อนนั้นเป็นโหนดที่เก็บลักษณะของต้นแบบ (Prototype) แต่ละวัตถุไว้ ซึ่งวัตถุหนึ่งวัตถุก็อาจมีได้หลายต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

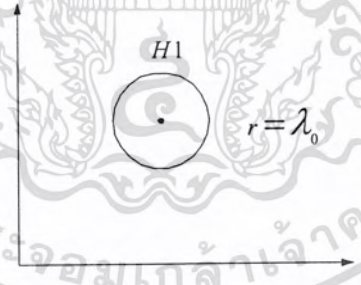
4.1 โหมดการเรียนรู้

ในโหมดการเรียนรู้นั้น จะเป็นการสอนความรู้ โดยป้อนลักษณะเฉพาะของวัตถุให้แก่โครงข่าย ซึ่งลักษณะที่ทำการหามาจากข้างต้นจะอยู่ในรูปของเวคเตอร์ แทนด้วย อินพุทเวคเตอร์ (Input Vector) $X^{(n)}$ สำหรับแต่ละค่าในเวคเตอร์ก็คือแต่ละค่าลักษณะเฉพาะของวัตถุ

เริ่มจากในครั้งแรกของการสอน เมื่อ อินพุทเวคเตอร์ $X^{(1)}$ ถูกใส่เข้าไปในโหนดชั้นอินพุท (Input Node) โครงข่ายจะทำการสร้างโหนดชั้นซ่อน (Hidden Node) H_1 และ โหนดชั้นเอาท์พุท (Output Node) O_1 โดยมีเส้นเชื่อมต่อระหว่าง โหนดชั้นอินพุทเหล่านี้กับ H_1 ดังรูปที่ 4-2 และรูปที่ 4-3



รูปที่ 4-2 แสดงการเพิ่มโหนดชั้นซ่อน และ โหนดชั้นเอาท์พุท สำหรับชนิด (Class) แรก



รูปที่ 4-3 แสดงการเพิ่มโหนดชั้นซ่อน และ โหนดชั้นเอาท์พุท สำหรับชนิด (Class) แรก

ซึ่งเส้นเชื่อมต่อเหล่านี้จะใส่ค่าน้ำหนักเป็น

$$W_{1i} = X_i^{(1)}$$

โดยทั่วไปเมื่อรับ อินพุทเวคเตอร์ $X^{(n)}$ เข้ามาโดยที่ $n \neq 1$ จะได้ เอาท์พุทของ H_1 เท่ากับ

1 ถ้า $\sum_i (W_{1i} - X_i^{(2)})^2 \leq \lambda_O$

0 ถ้า มากกว่า

ซึ่ง λ_O คือค่ารัศมีของ H_1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

O_1 จะแสดงถึงชนิดแรก จากการที่ต่อกันโดยตรงกับเอาท์พุทของ H_1 ซึ่งโดยทั่วไปแล้ว เอาท์พุทของ O_1 จะมีค่าเท่ากับ

1 ถ้า $\sum(Input \quad of \quad H_1) \geq 1$

0 ถ้า น้อยกว่า

ถ้าเอาท์พุทเป็น 1 หมายความว่าเวคเตอร์ ของลักษณะเฉพาะนั้นเป็น วัตถุชนิดที่ 1

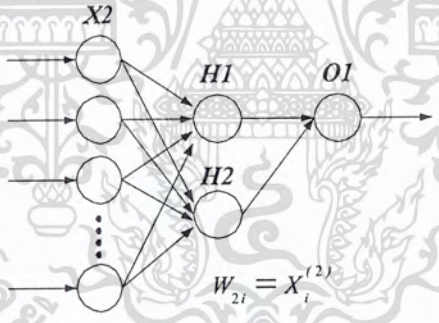
ต่อมาสำหรับการสอนครั้งต่อไปด้วยเวคเตอร์ที่สอง $X^{(2)}$

1) ถ้า $X^{(2)}$ เป็นเวคเตอร์ของชนิดที่ 1 และ เอาท์พุทของ O_1 เป็น 1 โครงข่ายจะไม่ต้องทำการปรับเปลี่ยน (Modify)

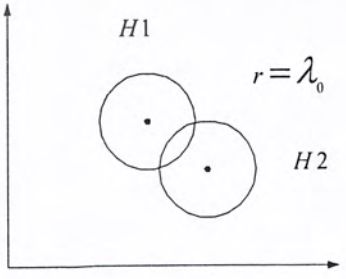
2) ถ้า $X^{(2)}$ เป็นเวคเตอร์ของชนิดที่ 1 แต่ เอาท์พุทของ O_1 เป็น 0 จะทำให้การแยกแยะชนิดผิด ดังนั้นจึงต้องทำการเพิ่มโหนดชั้นซ่อนใหม่ H_2 เข้าไปโดยที่มีเส้นเชื่อมต่อระหว่าง H_2 กับ โหนดชั้นอินพุท ที่มีค่าของน้ำหนักดังนี้

$$W_{2i} = X_i^{(2)}$$

และ λ_O เป็นรัศมีของ H_2 ด้วย ส่วนเอาท์พุทของ H_2 ถูกเชื่อมต่อกับ O_1 ดังรูปที่ 4-4 และรูปที่ 4-5



รูปที่ 4-4 แสดงการเพิ่มโหนดชั้นซ่อนสำหรับแสดงถึง ชนิดที่ 1



รูปที่ 4-5 แสดงการเพิ่มโหนดชั้นซ่อนสำหรับแสดงถึง ชนิดที่ 1

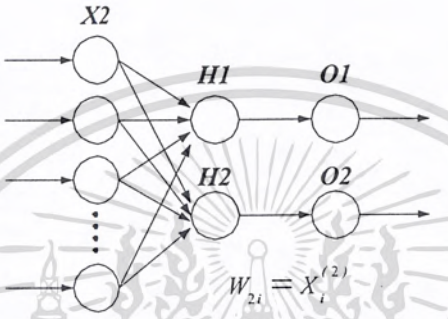
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

42794

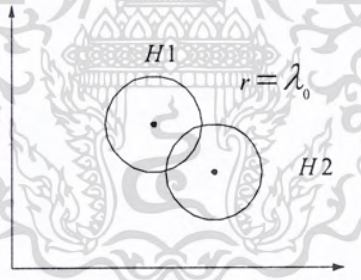
3) ถ้า $X^{(2)}$ เป็นเวคเตอร์ของชนิดที่ 2 และ เอาท์พุทของ O_1 เป็น 0 โครงข่ายจะต้องทำการเพิ่มโหนดใหม่ H_2 และ O_2 เข้าไป เพื่อใช้แสดงถึงชนิดที่ 2 และมีเส้นเชื่อมต่อจาก H_2 กับทุก โหนดชั้นอินพุท มีค่าของน้ำหนักเป็น

$$W_{2i} = X_i^{(2)}$$

รัศมีของ H_2 ยังคงใช้ค่า λ_O ดังรูปที่ 4-6 และรูปที่ 4-7



รูปที่ 4-6 แสดงการเพิ่มโหนดชั้นซ่อน และ โหนดชั้นเอาท์พุท สำหรับแสดงถึง ชนิดที่ 2

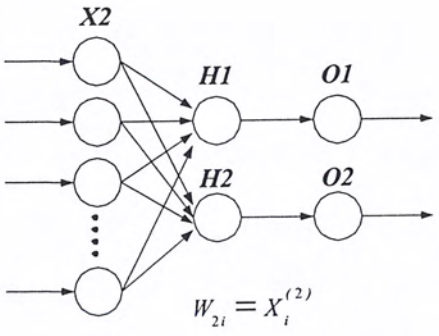


รูปที่ 4-7 แสดงการเพิ่มโหนดชั้นซ่อน และ โหนดชั้นเอาท์พุท สำหรับแสดงถึง ชนิดที่ 2

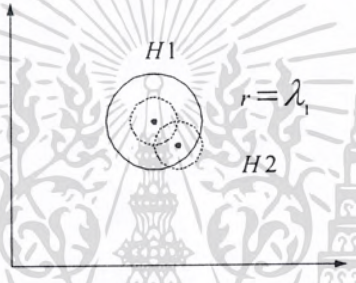
4) ถ้า $X^{(2)}$ เป็นเวคเตอร์ของชนิดที่ 2 แต่ เอาท์พุทของ O_1 เป็น 1 ซึ่งเป็นการแยกชนิดวัตถุที่ผิด ดังกรณีที่ผ่านมา H_2 และ O_2 ต้องทำการเพิ่มเข้าไปเพื่อให้เน็ทเวิร์ครับรู้ว่าเป็นลักษณะของชนิดที่ 2 และต้องมีการลดรัศมี λ_O ดังรูปที่ 4-8 และรูปที่ 4-9 เพื่อหลีกเลี่ยงการทำการแยกชนิดผิด ซึ่งรัศมีของทั้ง H_1 และ H_2 ที่เหมาะสมคือ

$$\lambda_1 \leq \sum_i (W_{1i} - X_i^{(2)})^2$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-8 แสดงการเพิ่มโหนดชั้นซ่อนและโหนดชั้นเอาท์พุท และ การลดรัศมี สำหรับแสดงถึง ชนิดที่ 2

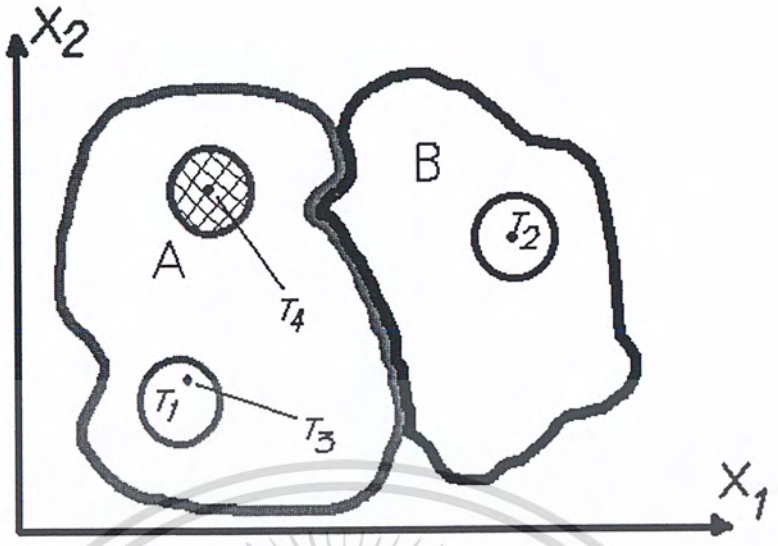


รูปที่ 4-9 แสดงการเพิ่มโหนดชั้นซ่อนและโหนดชั้นเอาท์พุท และ การลดรัศมี สำหรับแสดงถึง ชนิดที่ 2

ให้ทำการสอนให้โครงข่ายจนครบทุกภาพต้นแบบ แล้ววนทำการสอนให้กับโครงข่ายใหม่ตั้งแต่ภาพแรกจนถึงภาพสุดท้าย จนกระทั่งไม่มีการปรับเปลี่ยนใดๆในโครงข่าย คือไม่มีการสร้างนิวรอนโหนดใหม่ และไม่มีการลดรัศมีของนิวรอนโหนดใดอีก จึงหยุดทำการสอน เพราะในกรณีที่รอบแรกๆ หากอินพุทเวคเตอร์ $X^{(K)}$ เป็นชนิดเดียวกับ อินพุทเวคเตอร์ $X^{(j)}$ จะทำให้ $X^{(K)}$ ตกลงในรัศมีของ $X^{(j)}$ ดังนั้นจึงไม่มีการเพิ่มโหนดเข้าไป ต่อมาหากมีการลดรัศมีของ $X^{(j)}$ ทำให้ $X^{(K)}$ ไม่ตกลงในรัศมีของ $X^{(j)}$ ถ้ามีการถามเกี่ยวกับ $X^{(K)}$ จะทำให้ไม่สามารถแยกชนิดวัตถุได้

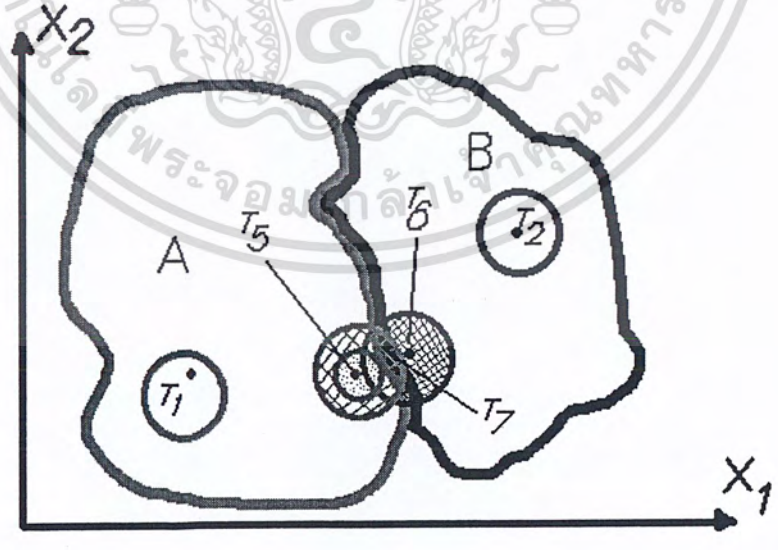
เพื่อแสดงการทำงานในโหมดการเรียนรู้ จะยกตัวอย่างดังรูปที่ 4-10 และรูปที่ 4-11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-10 แสดงการสร้างโหนดชั้นซ่อนขึ้นมาใหม่

โดยในรูปที่ 4-10 เมื่อทำการเรียนรู้ T1 และ T2โครงข่ายจะสร้างโหนดชั้นซ่อน T1 ขึ้นสำหรับแสดงถึงชนิด A และ T2 สำหรับแสดงถึงชนิด B ต่อมาเมื่อเรียนรู้ T3 ซึ่งเป็นชนิด A โครงข่ายคจะไม่ต้องทำการเปลี่ยนแปลงใดๆ เพราะ T3 ตกลงในรัศมีของ T1 ซึ่งเป็นพื้นที่ที่มีการเรียนรู้ของชนิด A อยู่แล้ว เมื่อเรียนรู้ T4 ซึ่งเป็นชนิด A จะมีการเพิ่มโหนดชั้นซ่อน T4 เข้าไป เพราะT4 ไม่ได้ตกลงในพื้นที่ที่มีการเรียนรู้สำหรับชนิด A จะเห็นว่าอาจมีได้หลายลักษณะต้นแบบในการใช้สำหรับแสดงชนิดของวัตถุหนึ่งชนิด



รูปที่ 4-11 แสดงการลดรัศมีของโหนดชั้นซ่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 4-11 มีอินพุทเวตเตอร์เข้ามาใหม่ 3 ตัว คือ T5 T6 T7 โดย T5 เป็นชนิด A และ T6 เป็นชนิด B และทั้งคู่มีรัศมีบางส่วนทับกันอยู่ เมื่อเรียนรู้ T7 ซึ่งเป็นชนิด B เข้ามา ทำให้ เอาท์พุทของ T5 และ T6 เกิดการแอคทีพขึ้นพร้อมกัน ทำให้ไม่สามารถแยกชนิดได้ถูก จึงต้องมีการลดรัศมีของ T5 ลง จนกระทั่ง T7 อยู่นอกรัศมีของ T5 ส่วน T7 ไม่ต้องเพิ่มเข้าไปเพราะตกลงในรัศมีของ T6 ซึ่งเป็นพื้นที่ที่มีการเรียนรู้ของ ชนิด B อยู่แล้ว

4.2 โหมดการแยกชนิดของวัตถุ

เมื่อต้องการทราบว่าวัตถุเป็นวัตถุอะไร จะทำการดึงลักษณะเฉพาะของวัตถุนั้นออกมา จัดเป็นอินพุทเวคเตอร์ เพื่อป้อนให้โครงข่าย แล้วผลลัพธ์ที่ได้จะเป็นอย่างใดอย่างหนึ่ง ดังนี้

1) มีโหนดชั้นเอาท์พุทเพียงโหนดเดียวที่แอคทีพ (Active) แสดงว่าวัตถุนั้นเป็นวัตถุชนิดซึ่งโหนดชั้นเอาท์พุท ที่แอคทีพอยู่
2) มีโหนดชั้นเอาท์พุท มากกว่าหนึ่งโหนดที่แอคทีพ แสดงว่าไม่สามารถทำการแยกชนิดของวัตถุได้
3) ไม่มีโหนดชั้นเอาท์พุทใดที่แอคทีพ แสดงว่าไม่สามรถจัดวัตถุนั้นให้ตรงกับชนิดของวัตถุใดที่เน็ทเวิร์ครู้จักอยู่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปและผลการทดลอง

5.1 การทดลอง

การทดลองเป็นการทดลองระบบเพื่อวัดประสิทธิภาพและความยืดหยุ่นต่างๆของการนำไปใช้งานจริง ซึ่งภาพที่ใช้ในการทดสอบนี้จะเป็นภาพที่ทำขึ้นจากโปรแกรมสร้างภาพสามมิติ โดยการสร้างวัตถุขึ้นมาเป็นโมเดลสามมิติ และจำลองแสงเงาตามสภาพแวดล้อมตามที่กำหนดคือ วัตถุจะวางอยู่หน้าฉากสีขาว และสร้างแหล่งกำเหนิดแสง 1 จุดแสงที่สร้างขึ้นเพื่อทำให้ภาพที่ได้มีเงาของวัตถุอยู่ด้วยเพราะเป็นการจำลองตามการนำระบบมาทดสอบในสภาพแวดล้อมจริงได้ เป็นการจำลองการนำวัตถุมาวางบนพื้นฉากสีขาวและเปิดไฟไว้ตรงกลางฉากและนำกล้องมาจับภาพ เงาที่เกิดตามขอบของภาพเงาวัตถุเกิดจากการกระเงาของแสงจะถูกจำกัดให้มีขนาดไม่มากจนเกินไป เพื่อวัดความยืดหยุ่นของระบบด้วย โปรแกรมที่นำมาสร้างภาพในการทดลองนี้คือโปรแกรม 3D MAX 3.1

การสอนระบบ จะสอนโดยภาพวัตถุจากภาพเดียวของวัตถุในมุมใดๆ ส่วนการถามจะถามจากภาพของวัตถุที่ทำการสอนไปแล้วในมุมต่างๆเพื่อดูการตอบคำถามจากผลของความผิดเพี้ยนต่างๆของภาพคำถามที่อาจเกิดได้เช่น ความเพี้ยนของมุมกล้อง ความเพี้ยนที่เกิดจากเงาของวัตถุที่ทำมุมต่างๆกับแสง หรือความผิดพลาดที่เกิดจากข้อจำกัดของระบบ

5.2 การทดลองที่ 1

การทดลองที่ 1 เป็นการทดลองการรู้จำวัตถุที่มีลักษณะโครงร่างต่างๆ และเงาของวัตถุที่เกิดขึ้นได้ตามสภาวะแวดล้อมจริงในห้องทดลอง รวมทั้งเป็นการทดลองเพื่อหาค่ายืดหยุ่นของจุดอ้างอิงที่จะใช้ในการตั้งค่าให้กับระบบ ตามตารางที่ 5-1เป็นผลการทดลองจากการตั้งค่าที่ 1, 0.99, 0.98 และ 0.97 ตามลำดับ

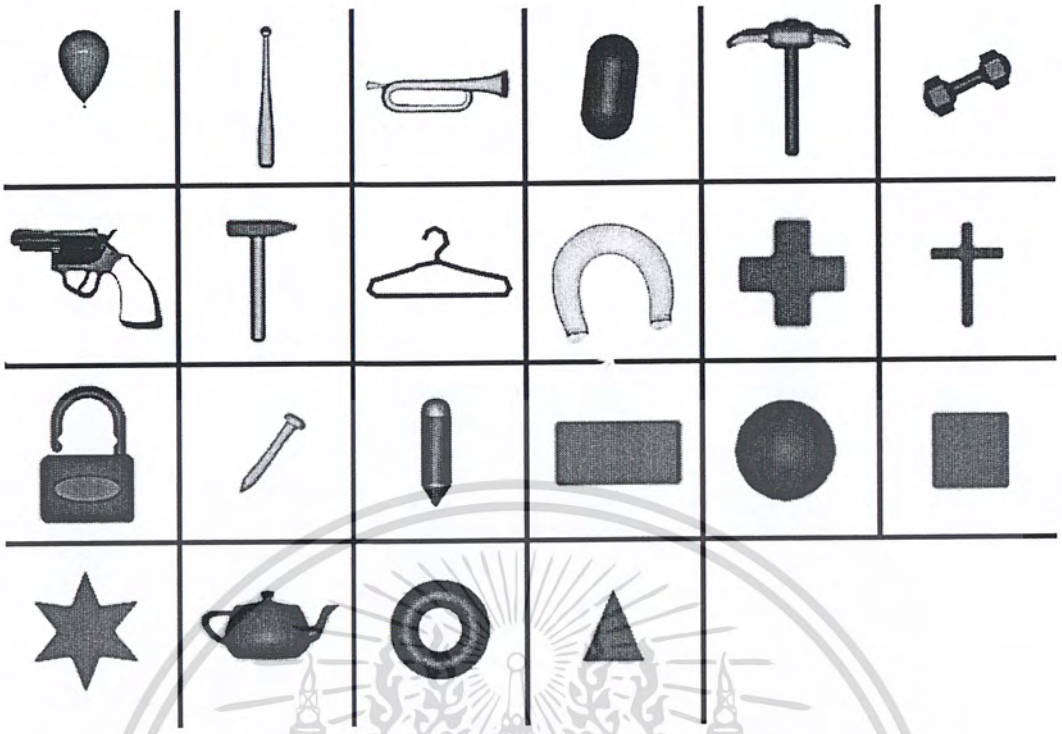
วัตถุนำมาสอนระบบ จำนวน 22 วัตถุ

ภาพที่ใช้เป็นคำถาม จำนวน 47 ภาพ

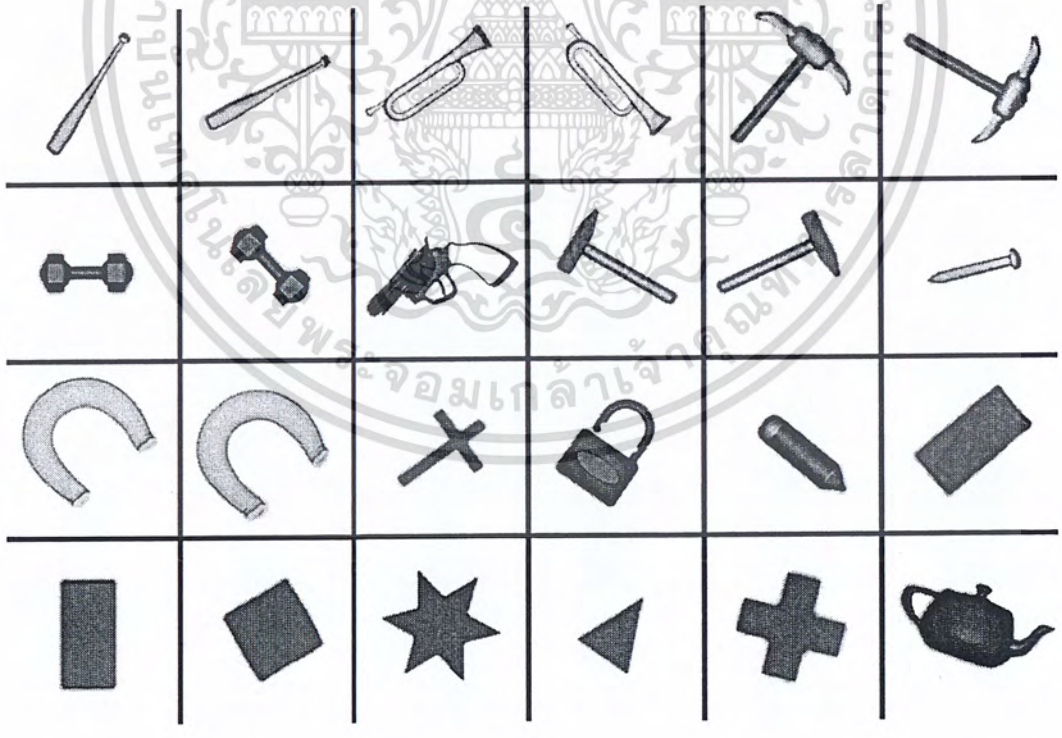
ค่าตั้งจุดอ้างอิง	1	0.99	0.98	0.97
ไม่มีคำตอบ	Dumbell 1 Rectangle 2	Dumbell 1	-	-
	รวม 3 ภาพ	รวม 1 ภาพ	-	-
มีคำตอบ	Karngken 1	Capsule 1	Capsule 1	Capsule 1
มากกว่า 1	รวม 1 ภาพ	รวม 1 ภาพ	รวม 1 ภาพ	รวม 1 ภาพ
ค่าผิดพลาด%	8.51 %	4.25 %	2.12 %	2.12 %
ค่าความถูกต้อง%	91.48 %	95.74 %	97.87 %	97.87 %

ตารางที่ 5-1 แสดงผลการทดลองที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



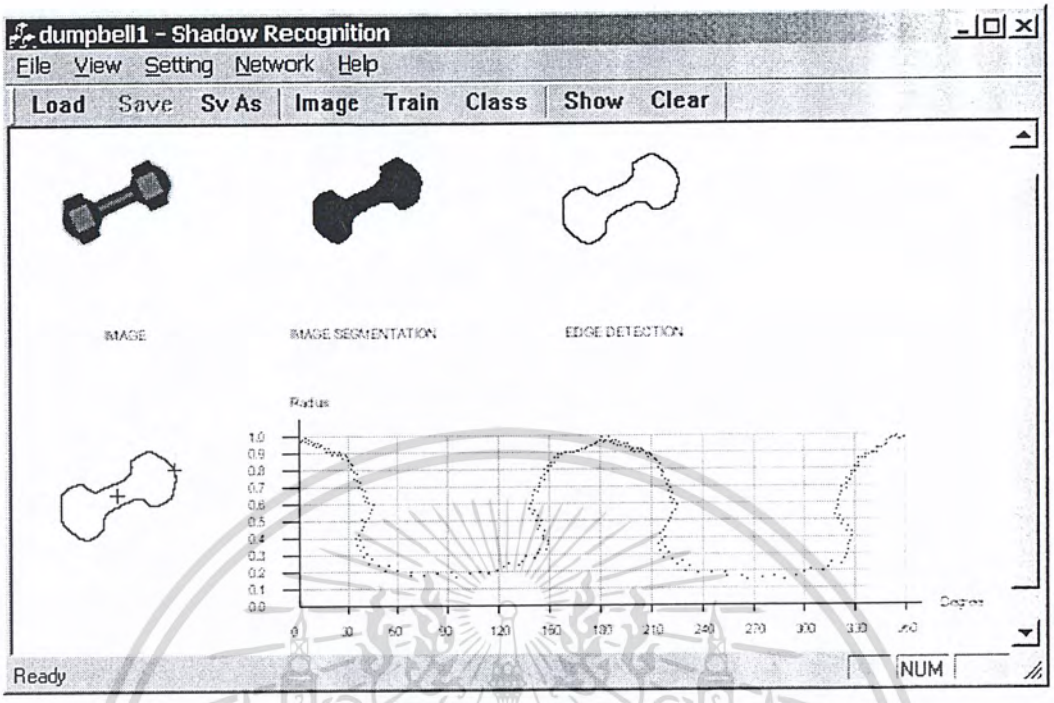
รูปที่ 5-1 แสดงตัวอย่างภาพวัตถุที่ใช้ในการสอนระบบ



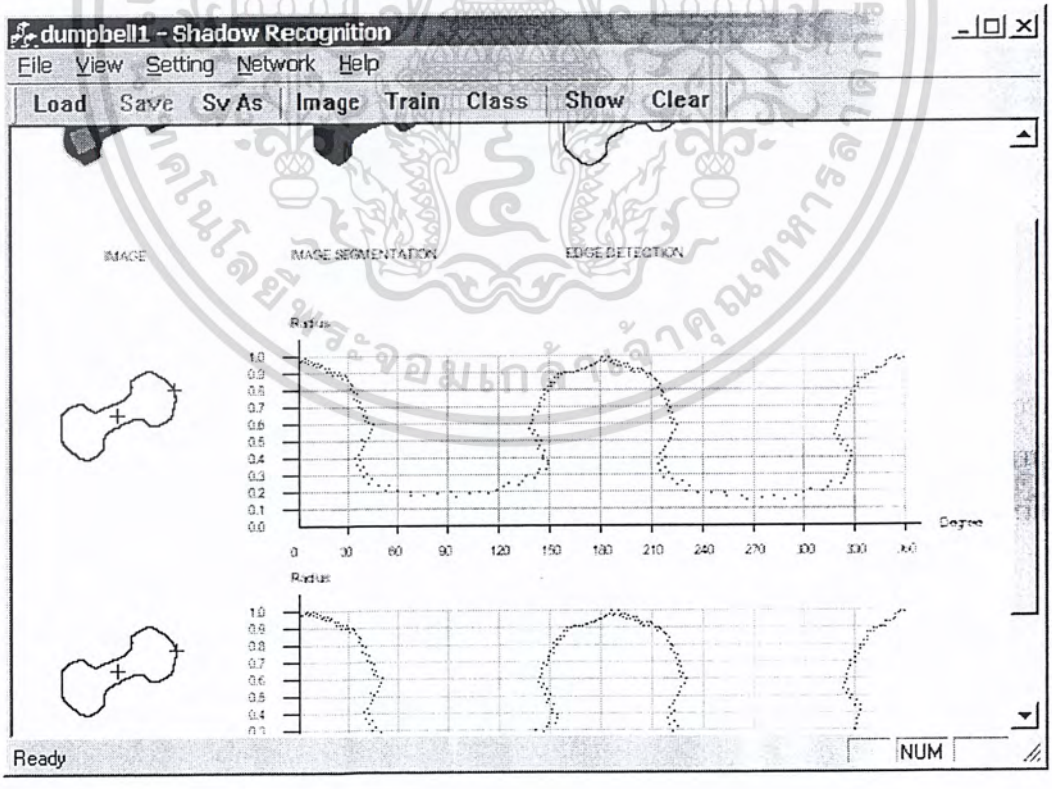
รูปที่ 5-2 แสดงตัวอย่างภาพคำถามมุมต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างข้อมูลที่ประมวลผลโดยระบบจากการตั้งค่าความยืดหยุ่นที่ค่าต่างๆ

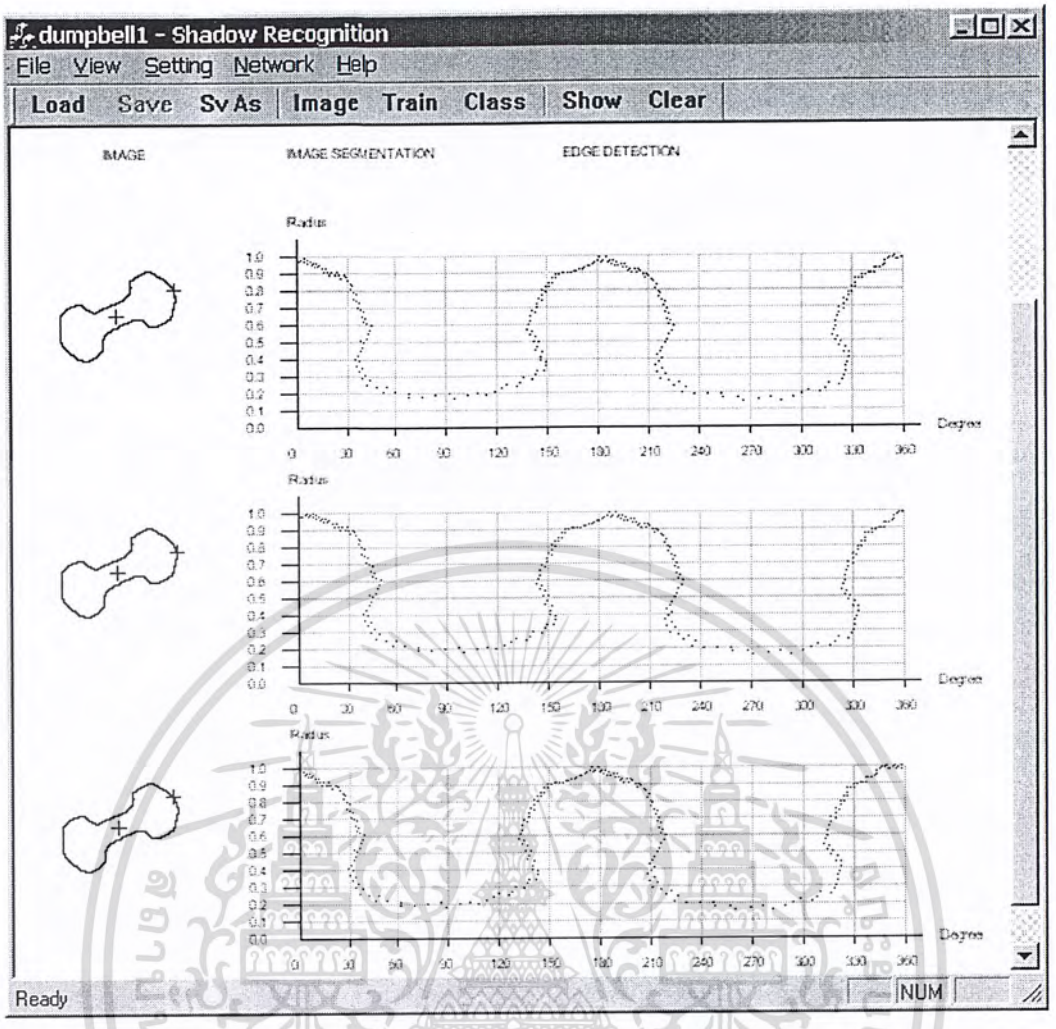


รูปที่ 5-3 แสดงผลการตั้งค่าความยืดหยุ่นของจุดอ้างอิงเท่ากับ 1.0



รูปที่ 5-4 แสดงผลการตั้งค่าความยืดหยุ่นของจุดอ้างอิงเท่ากับ 0.99

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5-5 แสดงผลการตั้งค่าความยืดหยุ่นของจุดอ้างอิงเท่ากับ 0.98 และ 0.97

ค่าความยืดหยุ่นของค่าจุดอ้างอิง เป็นค่าที่ยอมให้จุดขอบใดๆที่มีค่าระยะห่างจากจุดศูนย์ถ่วงของวัตถุน้อยกว่าจุดที่มีระยะห่างมากที่สุดได้ตามค่าอัตราส่วนตั้งไว้ มีประโยชน์เพื่อช่วยในการสอนระบบแทนที่จะมาสอนระบบด้วยภาพวัตถุหลายๆภาพ ระบบจะหาจุดอ้างอิงได้หลายค่าที่คิดว่าน่าจะเป็นไปได้ในกรณีที่ภาพมีมุมมองเปลี่ยนไปแล้วเกิดการเปลี่ยนจุดอ้างอิงเปลี่ยนไปจากตำแหน่งเดิม จึงมีการกำหนดให้ระบบสามารถเลือกค่าที่ใกล้เคียงและมีโอกาสเป็นจุดอ้างอิงในภายหลังได้ด้วย มาทำการสอนระบบในครั้งเดียวโดยถ้าค่าใดๆที่ได้นี้ไม่ต้องสอนทั้งหมดถ้าค่าที่ได้นั้นมีผลลัพธ์เดียวกับค่าที่ได้เหมือนกับจุดอ้างอิงที่ได้สอนไปก่อนแล้ว จากรูปที่ 5-3 เป็นค่าที่ตั้งไว้ที่ 1 จะทำให้มีการจำของระบบเพียง 1 รูปแบบ รูปที่ 5-4 เป็นการตั้งค่าที่ 0.99 จะมีความยืดหยุ่นคือจะมีการจำวัตถุได้ 2 รูปแบบ และสุดท้ายรูปที่ 5-5 ค่าที่ตั้งเป็น 0.98 และ 0.97 ซึ่งมีค่าความยืดหยุ่นเพิ่มขึ้นและได้รูปแบบของวัตถุที่จำถึง 3 รูปแบบ

การตั้งค่าความยืดหยุ่นที่ค่าต่ำจะทำให้การสอนโดยภาพวัตถุเพียงภาพเดียวมีค่าความถูกต้องมากกว่าจากการทดลองทดลองที่ 1 นี้ค่า 0.98 และ 0.97 มีค่าความถูกต้องมากที่สุดแต่ การเลือกตั้งค่าที่ 0.98 จะทำให้โปรแกรมไม่ต้องเสียเวลาในการเก็บค่าที่ไม่จำเป็นมาตรวจสอบมากเกินไป และถ้าตั้งค่านี้ต่ำเกินไปก็อาจจะทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้เกิดข้อผิดพลาดได้มากขึ้น เนื่องจากเป็นการยอมให้นำค่าที่ไม่ใช่ค่ามากสุดมาทำการการนอร์มอลไลซ์ตรงนี้เป็นการยอมรับให้เกิดเพราะโอกาสเกิดน้อย

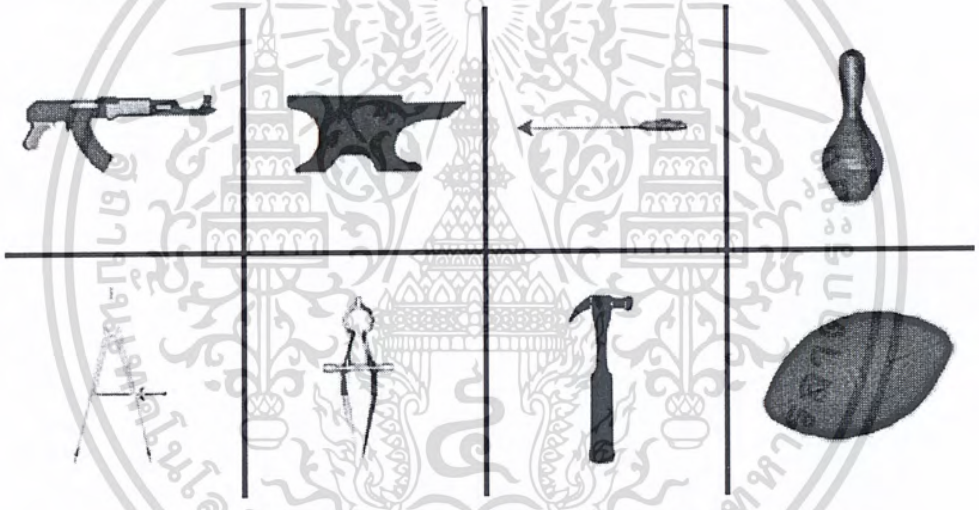
และอีกตัวอย่างข้อผิดพลาดเกิดจากภาพที่เป็นคำถามที่จุดอ้างอิงเปลี่ยนไป และไม่ได้เป็นภาพที่นำมาสอนคือนำมาเปลี่ยนแปลงน้ำหนักข้อมูลในนิวรอน ซึ่งอาจทำให้ตัววัตถุในภาพนั้นตกอยู่ในช่วงที่มากกว่า 1 คำตอบ เช่นกรณีของภาพ Capsule

5.3 การทดลองที่2

การทดลองที่ 2 เป็นการเพิ่มจำนวนวัตถุที่ใช้ในการทดลองเพื่อวิเคราะห์สภาพและผลลัพธ์ต่างๆ และดูแนวโน้มเปอร์เซ็นความถูกต้องของการตอบคำถามจากปริมาณจำนวนวัตถุที่เพิ่มขึ้นตามรูปที่ 5-6

วัตถุนำมาสอนระบบ จำนวน 30 วัตถุ

ภาพที่ใช้เป็นคำถาม จำนวน 72 ภาพ



รูปที่ 5-6 แสดงตัวอย่างภาพที่เพิ่มจากการทดลองที่ 1

ค่าตั้งจุดอ้างอิง	1	0.99	0.98	0.97
ไม่มีคำตอบ	Dumbell 1 Rectangle 2 Campass 1	Dumbell 1	-	-
	รวม 4 ภาพ	รวม 1 ภาพ	-	-
มีคำตอบ มากกว่า 1	Karngken 1 Capsule 1 Rugby 1	Capsule 1	Capsule 1	Capsule 1
	รวม 1 ภาพ	รวม 1 ภาพ	รวม 1 ภาพ	รวม 1 ภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าผิดพลาด%	9.72 %	2.77 %	1.38 %	1.38 %
ค่าความถูกต้อง%	90.28 %	97.23 %	98.62 %	98.62 %

ตารางที่ 5-2 แสดง ตารางผลการทดลอง

จากผลการทดลองจากตารางที่ 5-2 จะเห็นว่าจำนวนที่เพิ่มขึ้นจะได้เปอร์เซ็นต์ความถูกต้องเพิ่มขึ้น สำหรับกรณีที่มีการตั้งค่าที่ต่ำลงมาจนถึง 0.98 เนื่องจากภาพที่เพิ่มเข้าไปในการทดลองนี้ส่วนใหญ่จะเป็นภาพที่มีลักษณะที่มีโอกาสเปลี่ยนจุดอ้างอิงได้ แต่ยังเป็นลักษณะที่ยังไม่ทำให้เกิดการผิดพลาดจากการใช้ค่าความยืดหยุ่นที่ 0.98 นี้

5.4 การทดลองที่ 3

การทดลองที่ 3 เป็นการเพิ่มวัตถุที่มีลักษณะคล้ายกันมากๆ และเพิ่มจำนวนวัตถุที่ใช้ในการทดลอง เพื่อดูผลลัพธ์การรู้จำของระบบต่อวัตถุที่มีลักษณะคล้ายๆกันและมีจำนวนวัตถุในการสอนที่มากขึ้น เพื่อหาโอกาสการเกิดข้อผิดพลาดหรือข้อจำกัดของระบบ จากตารางที่ 5-3 เป็นผลการทดลองที่ได้จากการเพิ่มวัตถุดังต่อไปนี้

	เดิม	เพิ่ม
ค้อน	2	1
ปืน	1	4
ดาบ	0	7
กาน้ำ	1	1

วัตถุนำมาสอนระบบ จำนวน 43 วัตถุ

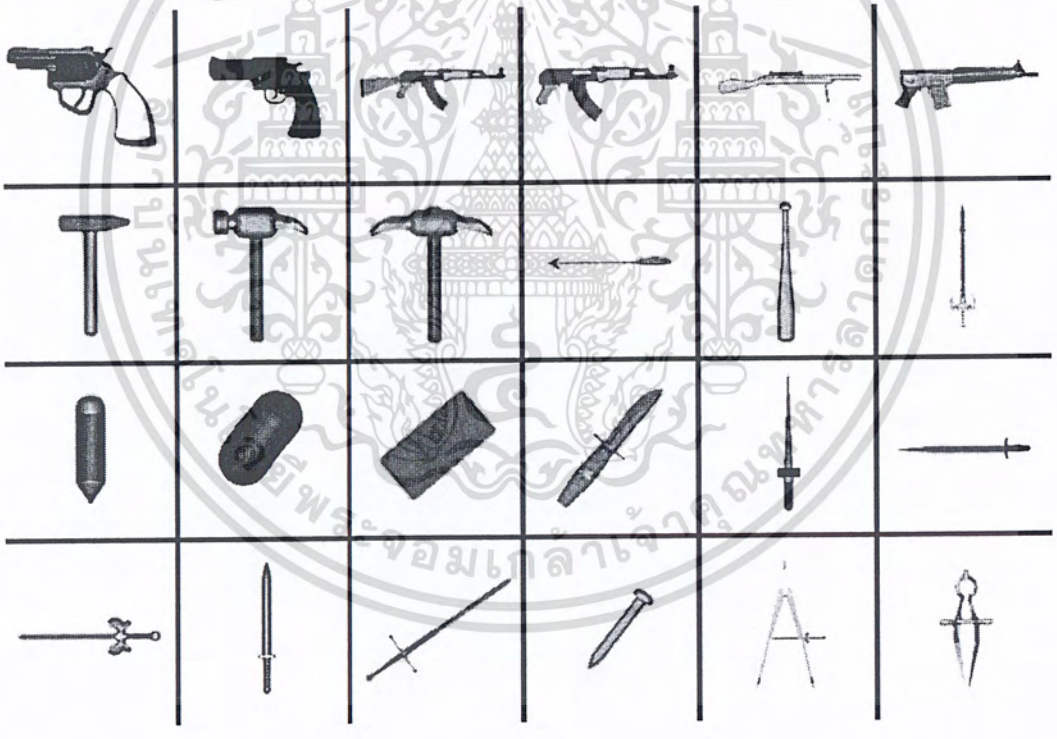
ภาพที่ใช้เป็นคำถาม จำนวน 106 ภาพ

ค่าตั้งจุดอ้างอิง	1	0.99	0.98	0.97
ไม่มีคำตอบ	Dumbell 1 Rectangle 2 Campass 1 Sniper 2	Dumbell 1	-	-
	รวม 6 ภาพ	รวม 1 ภาพ	-	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีคำตอบ มากกว่า 1	Bassball 1 Capsule 1 Rugby 1 Knife 1 Dig 2 Shrtsword 2 Sword1 2 Sword3 1	Capsule 1 Dig 2 Sniper 2 Knife 1 Shrtsword 2 Sword1 2	Capsule 1 Dig 2 Knife 1 Shrtsword 2 Sword1 2	Capsule 1 Dig 2 Knife 1 Shrtsword 2 Sword1 2
	รวม 11 ภาพ	รวม 10 ภาพ	รวม 8 ภาพ	รวม 8 ภาพ
ค่าผิดพลาด%	16.03 %	10.37 %	7.54 %	7.54 %
ค่าความถูกต้อง%	83.97 %	89.63 %	92.46 %	92.46 %

ตารางที่ 5-3 แสดงตารางผลการทดลอง



รูปที่ 5-7 แสดงตัวอย่างวัตถุที่มีลักษณะคล้ายๆกัน

สำหรับภาพในการทดลองที่ 3 นี้ ปัญหาส่วนใหญ่จะเกิดจากภาพที่มีลักษณะคล้ายกันและในการทดลองนี้ใช้ภาพที่มีขนาดเล็กไปเมื่อเทียบกับข้อมูลในกลุ่มนี้ เช่นประเภทของดาบที่จะทำให้มีข้อมูลในการเปรียบเทียบน้อยและอัตราส่วนของความเพี้ยนจากเงาสูงซึ่งทำให้เกิดผิดพลาดขึ้นได้ในการนำภาพมุมมองอื่นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

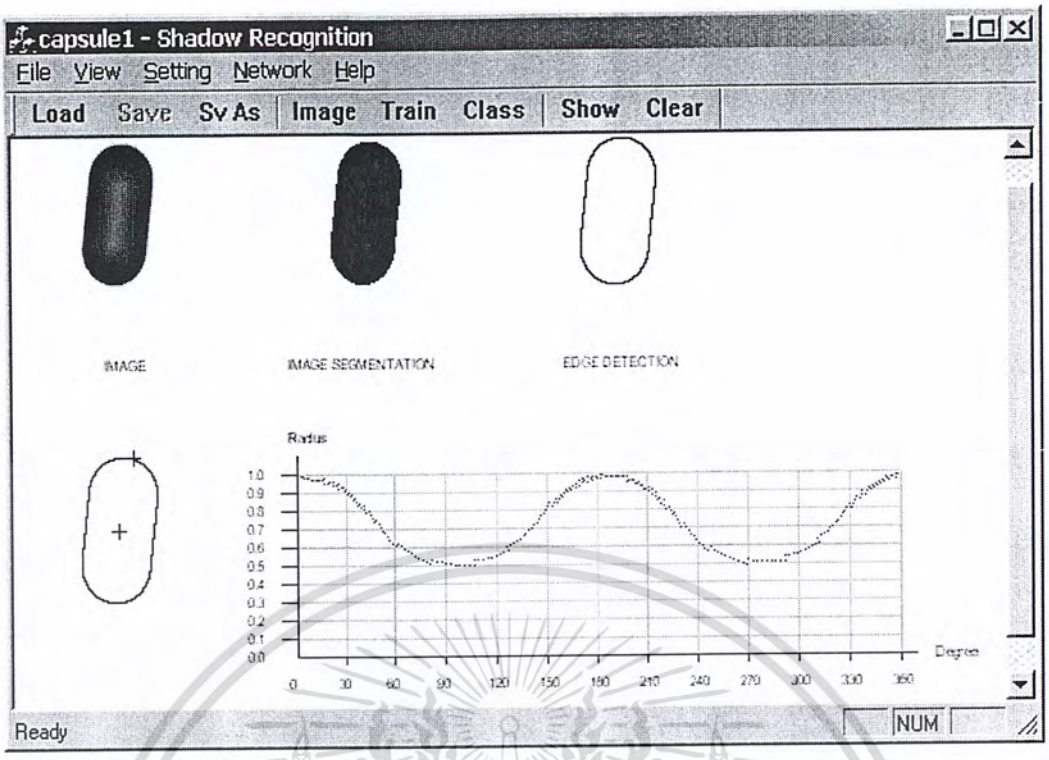
เกิดเงาแล้วทำให้ข้อมูลลักษณะตกอยู่ในช่วงคำตอบที่มากกว่า 1 ดังรูปที่ 5-7 เป็นตัวอย่างของวัตถุที่มีลักษณะคล้ายๆกัน

5.5 ปัญหาที่พบและการแก้ไข

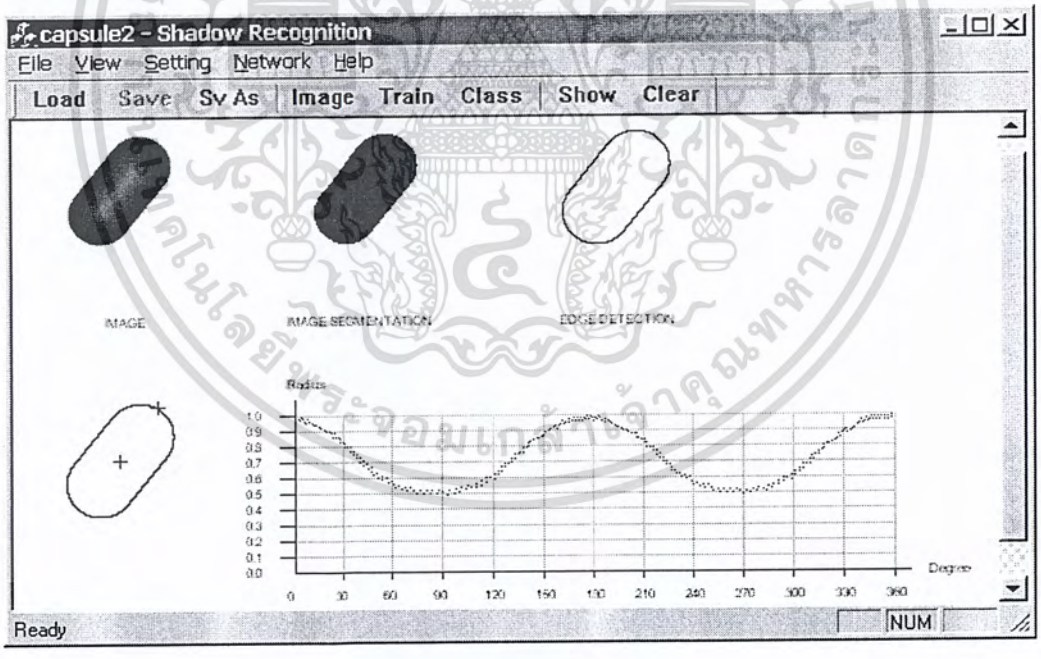
1. การที่ภาพคำถามมีความเพี้ยนเนื่องจากการใช้งานจริงเราไม่สามารถที่จะวางตำแหน่งและองค์ประกอบต่างๆให้เหมือนเดิมทุกครั้งได้ดังนั้นภาพที่ได้อาจจะเพี้ยนจากการวางวัตถุในองศาต่างๆ ตำแหน่งหรือระยะ ทำให้ภาพอาจมีเงาที่เปลี่ยนไปซึ่งทำให้ได้ข้อมูลที่ไม่แน่นอน ดังนั้นจึงมีการออกแบบระบบเพื่อให้รองรับจุดนี้ได้ โดยการหาจุดอ้างอิงต่างๆแล้วแบ่งวัตถุเป็นส่วนแยกประมวลผลความแตกต่างแต่ละส่วนซึ่งสามารถเกิดความเคลื่อนได้ตามค่าที่กำหนด
2. ปัญหาวัตถุที่เมื่อเปลี่ยนมุมจะมีจุดอ้างอิงที่เปลี่ยนไปได้ เช่นสี่เหลี่ยมผืนผ้า ต้องทำการตรวจสอบให้ครบทุกจุดที่มีโอกาสเป็นไปได้
3. ปัญหาการที่ภาพวัตถุคำถามตกในหลายคำตอบ เพราะเนื่องจากความเพี้ยนของภาพในมุมนั้นๆซึ่งเพี้ยนไปจากภาพที่ทำการสอนให้ระบบและไปตกในช่วงของวัตถุอื่นด้วย ทำการแก้ไขได้โดยทำการสอนภาพนั้นเข้าไป เพื่อเปลี่ยนแปลงน้ำหนักในนิวรอนใหม่
4. การที่วัตถุมีลักษณะคล้ายคลึงกันและลักษณะค่อนข้างมีอัตราส่วนของส่วนที่เพี้ยนเทียบกับข้อมูลมากทำให้ไม่สามารถวิเคราะห์ได้ สามารถแก้ได้โดยเพิ่มขนาดของภาพที่ใช้เช่นจากการทดลองจะเห็นว่าค้อนกับที่ขุดหรือปืนสั้นทั้งสองถึงแม้จะมีลักษณะที่คล้ายกันแต่ก็ยังมีข้อมูลพอให้วิเคราะห์ได้ แต่ลักษณะของดาบที่เหมือนกันแล้วยังมีตัวโครงร่างน้อยจะไม่สามารถวิเคราะห์ข้อมูลได้ ซึ่งทางแก้โดยอาจจะเพิ่มขนาดของภาพที่ใช้ และลดเงากระเจิงออกมานอกขอบวัตถุมากเกินไปเพื่อจำกัดอัตราส่วนความเพี้ยนต่อข้อมูลโครงร่างจริง

ตัวอย่างภาพที่มีปัญหาต่างๆจากภาพที่ 5-8 เป็นภาพวัตถุที่ใช้ในการสอน และภาพที่ 5-9 เป็นภาพของวัตถุเดิมแต่เมื่อมีมุมและแสงเงาที่เปลี่ยนไป ซึ่งนำมาเป็นภาพที่ใช้ถามระบบแล้วคำตอบที่ได้ไปตกอยู่ในช่วงของคำตอบอื่นได้ แต่ก็สามารถนำภาพนี้มาสอนให้วัตถุเพื่อเปลี่ยนแปลงข้อมูลในนิวรอน เพื่อทำให้ระบบสามารถตอบคำถามได้ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5-8 แสดงภาพCapsuleที่สอนให้ระบบ



รูปที่ 5-9 แสดงภาพคำถามCapsule

5.6 สรุปผลการทดลอง

จากการทดลองจะเห็นถึงปัญหาและข้อจำกัดต่างๆของระบบ จากหลักการขั้นต้นของการเลือกจุดอ้างอิงเพียงจุดเดียวจะทำให้ได้ความถูกต้องมากเมื่อเทียบเป็นจำนวนรอบในการเรียนรู้ของตัวโปรแกรมกับการที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลือกให้มีค่ายืดหยุ่นของจุดอ้างอิงซึ่งจะทำการประมวลผลมากกว่าและก็อาจมีข้อผิดพลาดได้จากขั้นตอนการนอร์มอลไลซ์ การเลือกจุดอ้างอิงจุดเดียวคือตั้งค่าความยืดหยุ่นเป็น 1 จะต้องมีการสอนระบบในหลายมุมให้แก่ระบบแต่แบบนี้จะไม่เกิดปัญหาความผิดเพี้ยนจากการเลือกค่าที่ไม่ใช่ค่าที่มากที่สุดมาเป็นจุดอ้างอิงซึ่งเป็นความเพี้ยนในช่วงการนอร์มอลไลซ์ข้อมูล แต่ในที่นี้เราทำการทดลองเพื่อเลือกค่าความยืดหยุ่นที่รับได้ในเรื่องความถูกต้องและเวลาในการประมวลผลอย่างคุ้มค่าที่สุด

สรุปจากการทดลองค่าความยืดหยุ่นที่เหมาะสมที่น่าจะนำมาใช้คือ 0.98 เพราะเป็นค่าที่ได้ความถูกต้องสูงและไม่เสียเวลาในการสอนมากเกินไป ซึ่งระบบสามารถรู้จำวัตถุที่มีลักษณะต่างๆได้ดีและมีความยืดหยุ่นต่อสภาพแสงเงาที่เกิดขึ้นซึ่งทำให้ข้อมูลภาพโครงร่างของวัตถุที่เปลี่ยนไปได้ ในแต่ละภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
ข้อเสนอแนะ

6.1 ระบบการรู้จำวัตถุจากภาพเงาโครงร่าง

ในโครงงานนี้ได้เสนอหลักการพื้นฐานของการออกแบบระบบการรู้จำเอาไว้ 4 ข้อหลักๆคือ

1. การเตรียมภาพอินพุทเพื่อกำจัดสัญญาณรบกวนต่างๆและปรับความชัดเจนของภาพ
2. หาขอบเขตของส่วนที่เป็นวัตถุ เพื่อแยกวัตถุกับพื้นฉาก
3. หาสภาวะปกติสำหรับแต่ละวัตถุ เพื่อการรู้จำที่ไม่ขึ้นกับตำแหน่ง ขนาด องศาของวัตถุ
4. การแยกวัตถุออกเป็นส่วนๆ เพื่อเก็บลักษณะและแยกแยะความแตกต่างของลักษณะของแต่ละส่วน

6.2 ข้อสรุปและเปรียบเทียบของการรู้จำ

ข้อดีของการระบบรู้จำแบบนี้จะมีความยืดหยุ่นต่อขนาด ตำแหน่ง และมุมในการวางวัตถุ นอกจากนี้ยังใช้หน่วยความจำในการเก็บข้อมูลน้อย และมีความถูกต้องในการทำงานค่อนข้างสูง

แต่ก็มีข้อเสียอยู่บ้าง คือเวลาการค้นหาคำตอบจะเพิ่มขึ้นตามจำนวนของวัตถุที่เก็บในโครงข่ายประสาทเทียม เพราะต้องเปรียบเทียบกับทุกนิวรอนโหนด

6.3 ข้อจำกัดของระบบ

ข้อจำกัดของระบบ คือ ยังไม่ได้รับภาพวัตถุจากกล้องวีดีโอโดยตรง ทำให้ต้องบันทึกภาพเป็นไฟล์ก่อนแล้วจึงนำมาใช้งาน ซึ่งถ้าจะเพิ่มส่วนนี้ก็ต้องคำนึงถึงเรื่องของสิ่งรบกวนจากสภาพแวดล้อม เช่น แสงและคุณภาพของกล้อง

6.4 แนวทางการพัฒนาต่อ

แนวทางในการพัฒนาต่อเพื่อให้ระบบมีความสมบูรณ์มากขึ้น คือเพิ่มในส่วนของการรับภาพและระบบการเตรียมข้อมูลขึ้นมา เพื่อให้สามารถทำงานติดต่อกับผู้ใช้ได้สะดวกขึ้นและจะทำให้การพัฒนาระบบเป็นไปได้อย่างรวดเร็วขึ้น นอกจากนี้ยังสามารถนำหลักการพื้นฐานทั้ง 4 ข้อที่ได้เสนอไปมาใช้โดยปรับเปลี่ยนหาฟังก์ชั่นในการทำงานของส่วนต่างๆ เช่น การหาวิธีที่มีประสิทธิภาพที่สามารถแยกเงาที่ไม่ต้องออกจากภาพวัตถุหรืออาจจะประยุกต์ใช้โครงข่ายประสาทเทียมชนิดอื่นที่มีประสิทธิภาพมากขึ้น หรือเพิ่มขนาดของภาพซึ่งจะได้ความละเอียดและความถูกต้องเพิ่มขึ้น การเพิ่มลักษณะต่างๆในการจำที่จะช่วยให้ระบบมีความถูกต้องมากขึ้น และสุดท้ายก็คือวิธีในการหาคำตอบภาพโดยที่เวลาการค้นหาคำตอบไม่ขึ้นกับจำนวนข้อมูลวัตถุในฐานความรู้ที่เพิ่มขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.5 การนำไปประยุกต์ใช้งาน

ในโครงงานนี้มีการพัฒนาระบบการรู้จำ เป็นการรู้จำวัตถุจากภาพเงาโครงร่าง ซึ่งก็สามารถนำไปประยุกต์ใช้ในงานด้านอื่นๆ เช่น

1. การรู้จำตัวอักษรและลายนิ้วมือ

โดยการรู้จำประเภทนี้ก็สามารถนำการรู้จำภาพโครงร่างมาใช้ในการรู้จำได้ ในที่นี้ถ้านำไปประยุกต์ใช้ในเครื่องสแกนข้อความ หรือลายนิ้วมือจะช่วยให้สามารถแสกนวัตถุได้ในหลายแนวไม่จำกัดว่าจะต้องอยู่ในแนวนอนเพียงอย่างเดียว ดังแสดงในรูปที่ 6-1

รูปที่ 6-1 แสดงภาพที่มีอักษรมีลักษณะทั้งในแนวนอนและแนวตั้ง

2. การรู้จำวัตถุโดยเพิ่มส่วนของผิวหนัง (Texture)

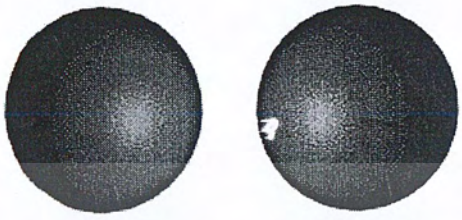
เช่นวัตถุที่มีลักษณะคล้ายๆกันจึงต้องนำเรื่องของผิวมาเปรียบเทียบด้วย เช่น ลูกฟุตบอลกับลูกบาสเก็ตบอล หรือกล่องมีลวดลายต่างๆ ดังรูปที่ 6-2 แสดงวัตถุที่มีลักษณะของขอบที่เหมือนกัน ซึ่งระบบยังไม่อาจสามารถแยกแยะได้ จึงต้องเพิ่มลายละเอียดการเปรียบเทียบกันในส่วนของสีในตำแหน่งต่างๆของวัตถุ

รูปที่ 6-2 แสดงวัตถุซึ่งมีลวดลาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การรู้จำวัตถุในลักษณะ 3 มิติ

การจับภาพรอบวัตถุในมุมต่างๆ แล้วสร้างโครงร่างในมุมต่างๆขึ้นมาเชื่อมโยงกันเป็นวัตถุอาจใช้หลักการของสเตอริโอวิชั่นเข้ามาช่วยในการสร้างโครงร่าง เพื่อสามารถรู้จำวัตถุได้หลายมุมมอง โดยแบ่งวัตถุออกเป็นส่วนๆตามพิกัดของ 3 มิติ แล้วเรียนรู้ด้วยโครงข่ายประสาทเทียม



รูปที่ 6-3 แสดง Left Image และ Right Image

รูปที่ 6-4 แสดงโครงร่างของวัตถุ

4. การประยุกต์ใช้ในหุ่นยนต์

การนำหุ่นยนต์ไปใช้ในสภาวะสิ่งแวดล้อมที่หลากหลายและมีการเปลี่ยนแปลงได้ ทั้งการใช้หุ่นยนต์ทำงานแทนมนุษย์ในสถานที่ที่อันตราย สถานที่ที่มนุษย์ไม่สามารถเข้าไปทำงานได้หรือทำงานได้ไม่สะดวก เช่น ในสถานที่ที่มีสารพิษหรือในท่อเล็กๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
ตัวแปรที่ใช้ในโปรแกรม

ตัวแปรที่สำคัญที่ใช้ในโปรแกรม

1. ตัวแปรที่ใช้เก็บข้อมูลของภาพวัตถุ คือ

```
CDIBitmap* m_pDIB;
```

2. ตัวแปรที่ใช้เป็นเก็บจุดภาพของภาพสองระดับ คือ

```
BYTE SegmentTbl[PICSIZE];
```

3. ตัวแปรที่ใช้เป็นเก็บจุดขอบของภาพ คือ

```
BYTE EdgeTbl[PICSIZE];
```

4. ตัวแปรที่ใช้เป็นเก็บจุดขอบของภาพในลักษณะเป็นแถว คือ

```
struct EdgePt {
double X, Y, Rad, RadNrml, Deg;
} ChainCode[PICSIZE];
```

5. ตัวแปรที่ใช้เป็นเก็บข้อมูลนอร์มอลไลซ์ในแต่ละส่วน ซึ่งแบ่งส่วนละ 30 องศาจำนวน 12 ส่วน คือ

```
struct structPiece {
        double Area, Bound, MaxRad, MinRad;
} Piece[12];
```

6. ตัวแปรที่ใช้เป็นชั้นอินพุท

```
struct structInputLayer {
        double Data[48];
} InputLayer;
```

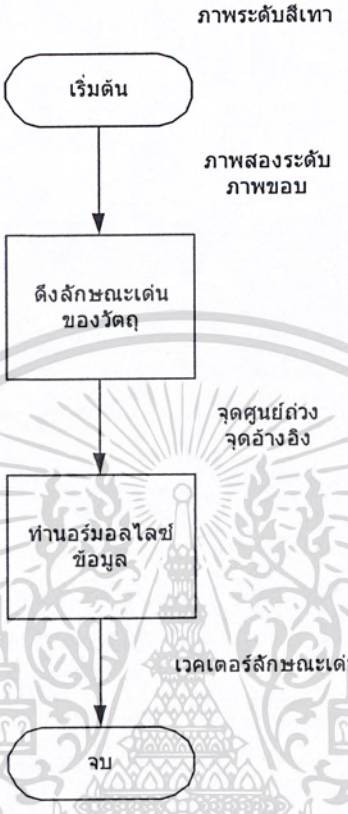
7. ตัวแปรที่ใช้เป็นโครงข่าย

```
struct structNetwork {
        โครงข่ายชั้นเอาท์พุท
        CString Name;
        โครงข่ายชั้นซ่อน
        double Rad;
        double Data[48];
} Network[MAXPROTOTYPE];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

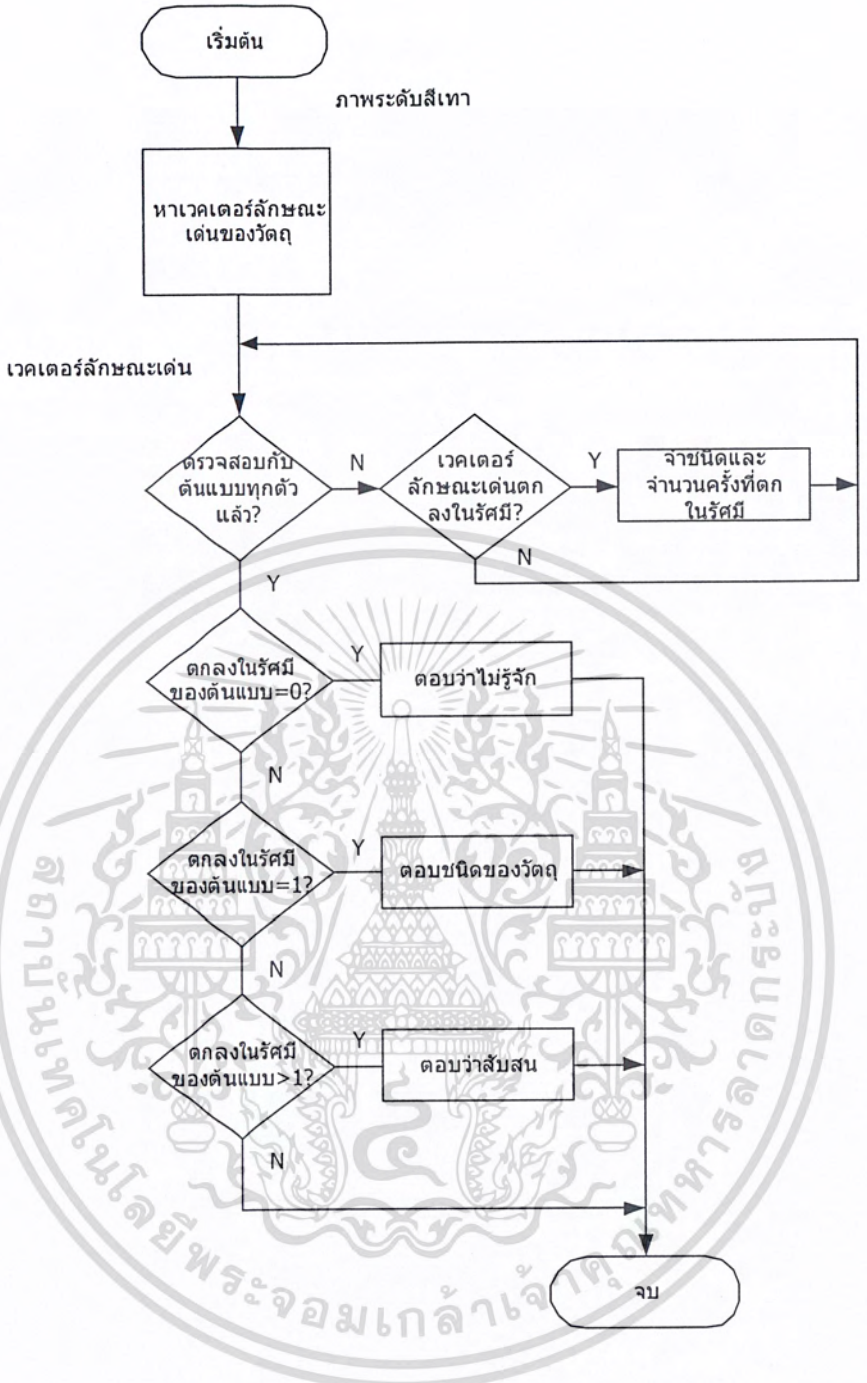
ภาคผนวก ข

Flow Chart



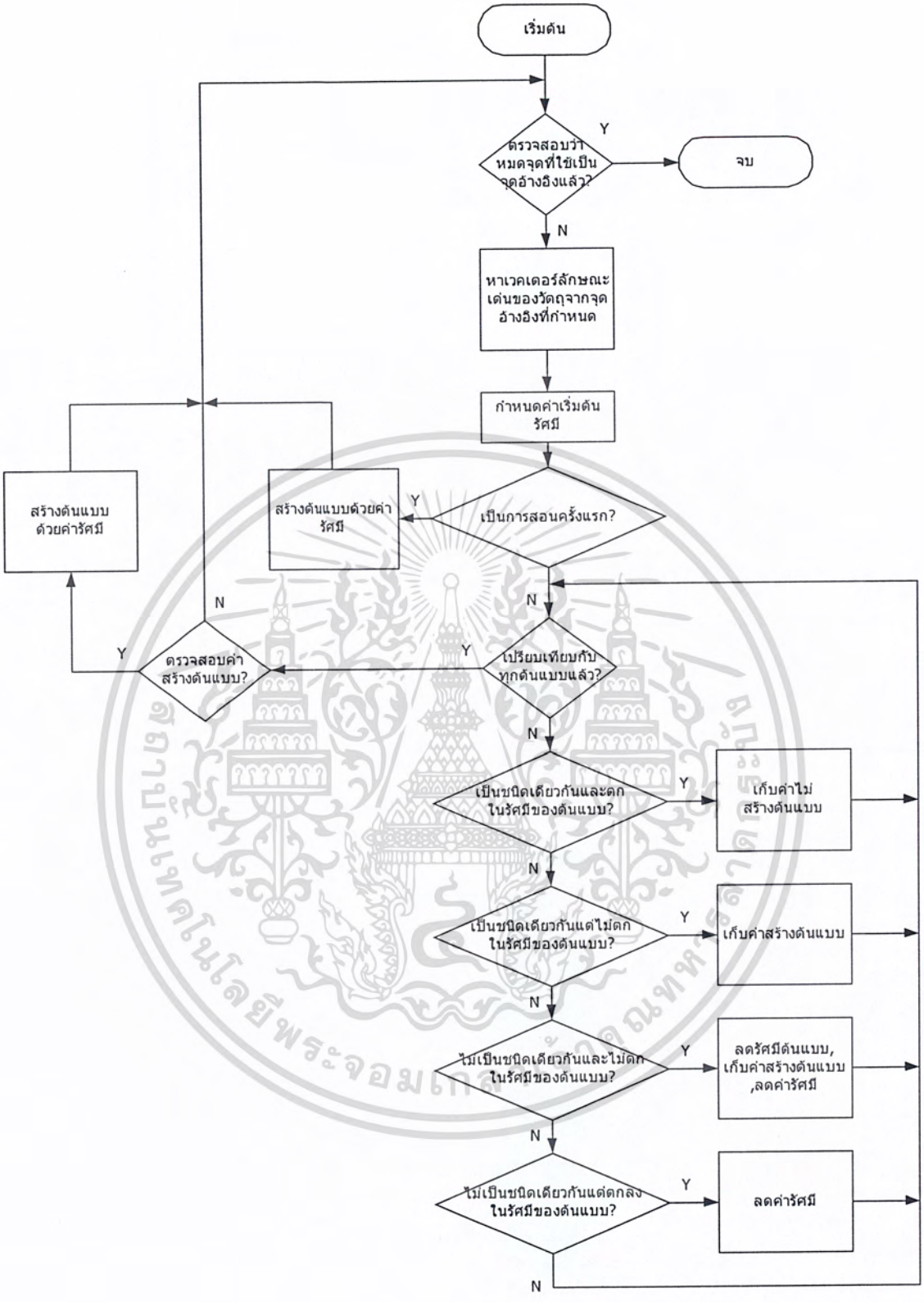
รูปที่ 1ค แสดง Flow Chart ของการหาเวคเตอร์ลักษณะเด่นของวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2ค แสดงFlow Chart ของการแยกชนิดวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3ค แสดงโฟลว์ชาร์ตของการสอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค
การทำงานของโปรแกรมในส่วนของการทำงานย่อย

ฟังก์ชันการทำงานย่อยที่สำคัญ

1. ฟังก์ชันการทำภาพสองระดับ
2. ฟังก์ชันการหาจุดศูนย์ถ่วงของวัตถุ
3. ฟังก์ชันการหาขอบของวัตถุ
4. ฟังก์ชันการเก็บจุดขอบของวัตถุ
5. ฟังก์ชันการคำนวณค่าต่างๆให้แก่จุดขอบ
6. ฟังก์ชันการหาระยะห่างจากจุดศูนย์ถ่วงถึงจุดขอบ
7. ฟังก์ชันการหาจุดที่มีระยะห่างจากจุดศูนย์ถ่วงถึงจุดขอบมากที่สุด
8. ฟังก์ชันการหามุมของจุดขอบซึ่งเทียบกับจุดศูนย์องศา
9. ฟังก์ชันการทำนอร์มอลไลซ์ค่าของระยะห่างจากจุดศูนย์ถ่วงถึงจุดขอบ
10. ฟังก์ชันการหาข้อมูลนอร์มอลไลซ์ในแต่ละชิ้นส่วนของวัตถุ จากการแบ่งวัตถุเป็น 12 ชิ้นชิ้นละ 30 องศา
11. ฟังก์ชันการหาค่านอร์มอลไลซ์ของพื้นที่ในแต่ละชิ้นส่วนของวัตถุ
12. ฟังก์ชันการหาค่านอร์มอลไลซ์ของจำนวนจุดขอบในแต่ละชิ้นส่วนของวัตถุ
13. ฟังก์ชันการหาค่านอร์มอลไลซ์ของค่ารัศมีที่มากที่สุดของจุดขอบในแต่ละชิ้นส่วนของวัตถุ
14. ฟังก์ชันการหาค่านอร์มอลไลซ์ของค่ารัศมีที่น้อยที่สุดของจุดขอบในแต่ละชิ้นส่วนของวัตถุ
15. ฟังก์ชันการโหลดค่าต่างๆของจุดขอบที่เก็บไว้ใน ChainCode[] มาเก็บใน TChainCode[]
16. ฟังก์ชันการหมุน TchainCode[] ให้จุดขอบที่เป็นจุดอ้างอิงมาอยู่ที่ 0 องศา

รายละเอียดและการทำงานของฟังก์ชันย่อย

1. ฟังก์ชันการทำภาพสองระดับ

เป็นการแปลงวัตถุจากภาพ Gray Scale 256 Level ให้เป็นภาพ Binary Image คือเปลี่ยนจุดของวัตถุเป็นสีดำ และจุดของฉากเป็นสีขาว ซึ่งผลลัพธ์จะอยู่ในตัวแปร SegmentTbl ดังนี้

```
//void CProject1Doc::Segmentation()
        for(i=0;i<PICSIZE;i++)
                SegmentTbl[i] = (SegmentTbl[i] < 200 )? BLACK:WHITE;
```

2. ฟังก์ชันการหาจุดศูนย์ถ่วงของวัตถุ

เป็นฟังก์ชันที่ใช้หาจุดศูนย์ถ่วงของวัตถุด้วยสมการ ซึ่งผลลัพธ์จะเก็บใส่ตัวแปร CentroidX , CentroidY ดังนี้ //void CProject1Doc::CalcCentroid()

```
        xi = xj = xb = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for (i=0;i<128;i++)
        for (j=0;j<128;j++)
                if (SegmentTbl[i*128+j] == BLACK)
                {
                        xi = i + xi;
                        xj = j + xj;
                        xb = 1 + xb;
                }
if (xb!=0)
{
        CentroidX = (xj/xb);
        CentroidY = (xi/xb);
}
```

และต้องมีการป้องกัน หาก xb เท่ากับ 0 โปรแกรมจะเกิด Run Time Error จากการหารด้วยค่าศูนย์

```
else
{
        CentroidX = CentroidY = 0;
}
```

3. ฟังก์ชันการหาขอบของวัตถุ

เป็นฟังก์ชันที่ทำหน้าที่หาขอบของวัตถุจากภาพสองระดับ โดยแสกนจากบนลงล่าง และจากซ้ายไปขวาเพื่อคำนวณด้วยมาส์ก ซึ่งผลลัพธ์จะเก็บใส่ตัวแปร EdgeTbl ดังนี้

```
//void CProject1Doc::EdgeDetection()
        for(i=1;i<127;i++)
                for(j=1;j<127;j++)
                        EdgeTbl[i*128+j] = - SegmentTbl[(i+1)*128 + j]
                                           - SegmentTbl[(i-1)*128 + j]
                                           - SegmentTbl[i*128 + (j+1)]
                                           - SegmentTbl[i*128 + (j-1)]
                                           - SegmentTbl[(i+1)*128 + (j+1)]
                                           - SegmentTbl[(i-1)*128 + (j+1)]
                                           - SegmentTbl[(i+1)*128 + (j-1)]
                                           - SegmentTbl[(i-1)*128 + (j-1)]
                                           + 8*SegmentTbl[i*128 + j];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นทำการแปลงขอบให้เป็นสีดำและฉากเป็นสีขาว

```
for(i=0;i<PICSIZE;i++)
        EdgeTbl[i] = (EdgeTbl[i] > 200 )? BLACK:WHITE;
```

4. ฟังก์ชันการเก็บจุดขอบของวัตถุ

เป็นฟังก์ชันการเก็บจุดขอบใส่ตัวแปร ChainCode และเก็บจำนวนของจุดขอบของวัตถุใส่ในตัวแปร NumChainPt ดังนี้

```
//void CProject1Doc::ChainCoding()
for(i=0;i<127;i++)
    for(j=0;j<127;j++)
        if(EdgeTbl[j*128+i]==BLACK)
        {
            ChainCode[NumChainPt].X = i;
            ChainCode[NumChainPt].Y = j;
            NumChainPt++;
        }
```

5. ฟังก์ชันการหาค่าต่างๆให้แก่จุดขอบ

เป็นฟังก์ชันที่เรียกฟังก์ชันย่อยอื่นขึ้นมาคำนวณค่าให้แก่จุดขอบ โดยผลลัพธ์จะเก็บไว้ในตัวแปร ChainCode[].Rad, MaxRadPt , ChainCode[].Deg, ChainCode[].RadNrml ตามลำดับ ดังนี้

```
//void CProject1Doc::CalcChain()
```

คำนวณรัศมีให้แต่ละจุดขอบ

```
CalcChainRadius();
```

นับจำนวนจุดขอบของวัตถุ

```
CalcChainMaxRadiusPt();
```

ทำการนอร์มอลไลซ์รัศมีแต่ละจุดขอบ

```
CalcChainRadiusNormalize();
```

คำนวณมุมของแต่ละจุดขอบ

```
CalcChainDegree();
```

6. ฟังก์ชันการหาระยะห่างจากจุดศูนย์ถ่วงถึงจุดขอบ

เป็นฟังก์ชันซึ่งหาระยะห่างจากจุดศูนย์ถ่วงถึงจุดขอบแต่ละจุด โดยผลลัพธ์ที่ได้เก็บไว้ในแปร ChainCode[].Rad ดังนี้

```
//void CProject1Doc::CalcChainRadius()
for(i=0;i<NumChainPt;i++)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ChainCode[i].Rad = (double)sqrt(pow((CentroidX-ChainCode[i].X),2)+ pow
((CentroidY-ChainCode[i].Y),2));
```

7. ฟังก์ชันการหาจุดที่มีระยะห่างจากจุดศูนย์ถ่วงถึงจุดขอบมากที่สุด

คำนวณหาจุดที่มีระยะห่างจากจุดศูนย์ถ่วงถึงจุดขอบมากที่สุด และเก็บไว้ในตัวแปร MaxRadPt โดยการตั้งค่า MaxRadPt เท่ากับศูนย์ แล้วไล่หาจุดที่มีรัศมีมากกว่ามาเก็บไว้ สุดท้ายจะได้จุดที่มีรัศมีมากที่สุด ดังนี้

```
//void CProject1Doc::CalcChainMaxRadiusPt()
    MaxRadPt = 0;
    for(i=0;i<NumChainPt;i++)
        if(ChainCode[MaxRadPt].Rad<ChainCode[i].Rad)
            MaxRadPt = i;
```

8. ฟังก์ชันการหามุมของจุดขอบซึ่งเทียบกับจุดศูนย์องศา

ทำการหามุมด้วยสร้างเวคเตอร์ขึ้นมา 2 เวคเตอร์แล้วทำการหามุมระหว่างเวคเตอร์นั้น โดยเวคเตอร์แรกคือเวคเตอร์ที่มีทิศทางจากจุดศูนย์ถ่วงไปตามแกน X และเวคเตอร์ที่สองคือเวคเตอร์ที่มีทิศทางจากจุดศูนย์ถ่วงไปยังจุดขอบ โดยผลลัพธ์จะเก็บในตัวแปร ChainCode[].Deg ดังนี้

```
//void CProject1Doc::CalcChainDegree()
    double x1,y1,x2,y2;
    int i;
    // U = (a,b), V = (c,d)
    double a,b,c,d,UV,Ul,Vl,zeta;

    x1 = CentroidX;
    y1 = CentroidY;
    x2 = CentroidX+50;
    y2 = CentroidY;
    a = x2-x1;
    b = y2-y1;

    for(i=0;i<NumChainPt;i++)
    {
        c = ChainCode[i].X-x1;
        d = ChainCode[i].Y-y1;
        UV = a*c+b*d;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Ul = sqrt(a*a+b*b);
Vl = sqrt(c*c+d*d);
```

เปลี่ยนมุมให้อยู่ในรูปของดีกรี

```
zeta = acos(UV/(Ul*Vl));
zeta = zeta*180/3.142857;
//if(c>0 && d>0) ;
//if(c<0 && d>0) ;
if(c<0 && d<0) zeta = 360-zeta;
if(c>0 && d<0) zeta = 360-zeta;
ChainCode[i].Deg = (int) zeta;
}
```

9. ฟังก์ชันการทำนอร์มอลไลซ์ค่าของระยะห่างจากจุดศูนย์ถ่วงถึงจุดขอบ

ทำการหารรัศมีของแต่ละจุดขอบด้วยรัศมีที่มากที่สุด แล้วเก็บผลลัพธ์ลงใน ChainCode[].RadNrml ดังนี้

```
//void CProject1Doc::CalcChainRadiusNormalize()
for(i=0;i<NumChainPt;i++)
ChainCode[i].RadNrml = (double)(ChainCode[i].Rad/ChainCode
[MaxRadPt].Rad);
```

10. ฟังก์ชันการหาข้อมูลนอร์มอลไลซ์ต่างๆในแต่ละชิ้นส่วนของวัตถุ จากการแบ่งวัตถุเป็น 12 ชิ้นชิ้นละ 30 องศาโดยมีฟังก์ชันของการทำงานย่อยดังนี้

ทำการเรียกฟังก์ชันย่อยอื่นๆ มาทำการหาข้อมูลนอร์มอลไลซ์ในแต่ละ 30 องศาของวัตถุ ของจุดอ้างอิงที่รับเข้ามา โดยผลลัพธ์จะเก็บ ไว้ใน Piece[].Area, Piece[].Bound, Piece[].MaxRad, Piece[].MinRad ดังนี้

```
//void CProject1Doc::CalcEachPiece(int base)
```

หาค่านอร์มอลไลซ์ของพื้นที่ในแต่ละชิ้นส่วนของวัตถุ สำหรับจุดอ้างอิงที่รับเข้ามา

```
CalcEachPieceArea(base);
```

ทำการโหลดค่าของจุดขอบของวัตถุที่คำนวณเก็บไว้ใน ChainCode[] มาใช้

```
GetTChainCode();
```

ทำการเลื่อนลำดับของจุดขอบใน TChainCode[] ให้จุดแรกเป็นจุดอ้างอิงซึ่งมีมุมที่ 0 องศา

```
CalcTChainRotate(TChainCode[base].Deg);
```

หาค่านอร์มอลไลซ์ของจำนวนจุดขอบในแต่ละชิ้นส่วนของวัตถุ สำหรับจุดอ้างอิงที่รับเข้ามา

```
CalcEachPieceBound(base);
```

หาค่านอร์มอลไลซ์ของค่ารัศมีที่มากที่สุดของจุดขอบในแต่ละชิ้นส่วนของวัตถุ สำหรับจุดอ้างอิงที่รับเข้ามา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CalcEachPieceMaxRad(base);
```

หาค่านอร์มอลไลซ์ของค่ารัศมีที่น้อยที่สุดของจุดขอบในแต่ละชิ้นส่วนของวัตถุ สำหรับจุดอ้างอิงที่รับเข้ามา

```
CalcEachPieceMinRad(base);
```

11. ฟังก์ชันการหาค่านอร์มอลไลซ์ของพื้นที่ในแต่ละชิ้นส่วนของวัตถุ

เป็นการหาค่านอร์มอลไลซ์ของพื้นที่ในแต่ละชิ้นส่วนของวัตถุ สำหรับจุดอ้างอิงที่รับเข้ามา โดยคำนวณว่าจุดนั้นทำมุมเป็นเท่าไรกับ จุดอ้างอิง โดยใช้วิธีคำนวณมุมระหว่างเวคเตอร์ แล้วทำการเพิ่มค่าให้กับตัวแปรที่ใช้เก็บจำนวนพื้นที่ของแต่ละชิ้นส่วน ซึ่งตัวแปรที่ใช้เก็บผลลัพธ์คือ Piece[].Area ดังนี้

```
//void CProject1Doc::CalcEachPieceArea(int base)
    if(0<=zeta && zeta<30) Piece[0].Area++;
    else if(30<=zeta && zeta<60) Piece[1].Area++;
    else if(60<=zeta && zeta<90) Piece[2].Area++;
    else if(90<=zeta && zeta<120) Piece[3].Area++;
    else if(120<=zeta && zeta<150) Piece[4].Area++;
    else if(150<=zeta && zeta<180) Piece[5].Area++;
    else if(180<=zeta && zeta<210) Piece[6].Area++;
    else if(210<=zeta && zeta<240) Piece[7].Area++;
    else if(240<=zeta && zeta<270) Piece[8].Area++;
    else if(270<=zeta && zeta<300) Piece[9].Area++;
    else if(300<=zeta && zeta<330) Piece[10].Area++;
    else if(330<=zeta && zeta<360) Piece[11].Area++;
```

12. ฟังก์ชันการหาค่านอร์มอลไลซ์ของจำนวนจุดขอบในแต่ละชิ้นส่วนของวัตถุ

หาค่านอร์มอลไลซ์ของจำนวนจุดขอบในแต่ละชิ้นส่วนของวัตถุ สำหรับจุดอ้างอิงที่รับเข้ามา โดยเปรียบเทียบว่าจุดขอบนั้นมามุมกี่องศากับจุดอ้างอิง แล้วทำการเพิ่มค่าของจำนวนจุดขอบในส่วนนั้น จากนั้นทำการนำจำนวนจุดขอบในแต่ละส่วนมาหารด้วยจำนวนจุดขอบทั้งหมดแล้วเก็บไว้ ซึ่งตัวแปรที่ใช้เก็บผลลัพธ์คือ Piece[].Bound ดังนี้

```
//void CProject1Doc::CalcEachPieceBound(int base)
    for(i=0;i<12;i++)
    {
        Piece[i].Bound = 0;
        for(j=0;j<NumTChainPt;j++)
        {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    if(TChainCode[j].Deg >= i*30  && TChainCode[j].Deg < (i+1)*30)
                        Piece[i].Bound++;
                }
ทำการหารจำนวนจุดขอบในแต่ละส่วนด้วยจำนวนจุดขอบทั้งหมด
                Piece[i].Bound=Piece[i].Bound/NumTChainPt;
        }
```

13. ฟังก์ชันการหาค่านอร์มอลไลซ์ของค่ารัศมีที่มากที่สุดของจุดขอบในแต่ละชิ้นส่วนของวัตถุ

หาค่านอร์มอลไลซ์ของค่ารัศมีที่มากที่สุดของจุดขอบในแต่ละชิ้นส่วนของวัตถุ สำหรับจุดอ้างอิงที่รับเข้ามา โดยเก็บผลลัพธ์ไว้ในตัวแปร Piece[].MaxRad ดังนี้

```
//void CProject1Doc::CalcEachPieceMaxRad(int base)
for(i=0;i<12;i++)
{
        if(Piece[i].Bound == 0) Piece[i].MaxRad = 0;
        else
        {
            Piece[i].MaxRad = 0;
            for(j=0;j<NumTChainPt;j++)
                if(Piece[i].MaxRad < TChainCode[j].RadNrml && TChainCode[j].Deg
                >= i*30  && TChainCode[j].Deg < (i+1)*30 )
                Piece[i].MaxRad = TChainCode[j].RadNrml;
        }
}
```

14. ฟังก์ชันหาค่านอร์มอลไลซ์ของค่ารัศมีที่น้อยที่สุดของจุดขอบในแต่ละชิ้นส่วนของวัตถุ

หาค่านอร์มอลไลซ์ของค่ารัศมีที่น้อยที่สุดของจุดขอบในแต่ละชิ้นส่วนของวัตถุ สำหรับจุดอ้างอิงที่รับเข้ามา โดยเก็บผลลัพธ์ไว้ในตัวแปร Piece[].MinRad ดังนี้

```
//void CProject1Doc::CalcEachPieceMinRad(int base)
for(i=0;i<12;i++)
{
        if(Piece[i].Bound == 0) Piece[i].MinRad = 0;
        else
        {
            Piece[i].MinRad = 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                for(j=0;j<NumTChainPt;j++)
                        if(Piece[i].MinRad > TChainCode[j].RadNrml && TChainCode[j].Deg
                        >= i*30 && TChainCode[j].Deg < (i+1)*30 )
                Piece[i].MinRad = TChainCode[j].RadNrml;
        }
}
```

15. ฟังก์ชันการโหลดค่าต่างๆของจุดขอบที่เก็บไว้ใน ChainCode[] มาเก็บใน TChainCode[]

ทำหน้าที่โหลดค่าต่างๆของจุดขอบเช่น Rad, Deg, RadNrml ของ ChainCode[] มาใส่ใน TchainCode [] และโหลดค่า NumChainPt มาใส่ใน TNumChainPt เพื่อใช้ในการคำนวณค่าต่างๆให้แต่ละจุดที่ใช้เป็นจุดอ้างอิง ดังนี้

```
//void CProject1Doc::GetTChainCode()
NumTChainPt = NumChainPt;
TMaxRadPt = MaxRadPt;
for(i=0;i<NumTChainPt;i++)
{
        TChainCode[i].Deg = ChainCode[i].Deg ;
        TChainCode[i].Rad = ChainCode[i].Rad;
        TChainCode[i].RadNrml = ChainCode[i].RadNrml;
        TChainCode[i].X = ChainCode[i].X;
        TChainCode[i].Y = ChainCode[i].Y;
}
```

16. ฟังก์ชันการหมุน TchainCode[] ให้จุดขอบที่เป็นจุดอ้างอิงมาอยู่ที่ 0 องศา

ทำหน้าที่หมุนจุดขอบที่อยู่ใน TchainCode[] ให้จุดขอบที่ใช้เป็นจุดอ้างอิงมาอยู่ที่ 0 องศาโดยการหมุนทุกจุดขอบด้วยมุมของจุดที่ใช้อ้างอิง ดังนี้

```
//void CProject1Doc::CalcTChainRotate(double deg)
for(i=0;i<NumChainPt;i++)
{
        TChainCode[i].Deg = TChainCode[i].Deg - deg;
        if(TChainCode[i].Deg<0)
                TChainCode[i].Deg += 360;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง
การทำงานของโปรแกรมในส่วนของการหลัก

ฟังก์ชันการทำงานหลักที่สำคัญ

1. ฟังก์ชันการโหลดวัตถุ
2. ฟังก์ชันการสอน
3. ฟังก์ชันการแยกชนิดของวัตถุ
4. ฟังก์ชันการแสดงโครงข่าย
5. ฟังก์ชันการลบโครงข่าย
6. ฟังก์ชันการเปิดไฟล์ข้อมูลการสอน
7. ฟังก์ชันการบันทึกไฟล์ข้อมูลการสอนเดิม
8. ฟังก์ชันการบันทึกไฟล์ข้อมูลการสอนลงไฟล์
9. ฟังก์ชันการแสดงผลลงบนหน้าจอวินโดวส์

รายละเอียดและการทำงานของฟังก์ชันหลัก

1. ฟังก์ชันการโหลดวัตถุ

ทำการโหลดวัตถุตามชื่อไฟล์ที่ทำการเลือก จากนั้นทำการประมวลผลภาพ เพื่อเก็บข้อมูลขอบของวัตถุ ดังนี้

```
//void CProject1Doc::Serialize(CArchive& ar)
CFile* fp = ar.GetFile();
ASSERT(fp);
ar.Flush();
if(m_pDIB != NULL)
        delete m_pDIB;
m_pDIB = new CDIBitmap;
ถ้าโหลดภาพไม่สำเร็จ จะทำการแจ้ง แล้วหยุดการทำงานของฟังก์ชันนี้
if(!m_pDIB->LoadDIB(fp))
{
        AfxMessageBox("Error loading DIB file.");
        return;
}
ถ้าโหลดภาพสำเร็จ จะทำการประมวลผลภาพต่อไป
Segmentation();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CalcCentroid();
EdgeDetection();
ChainCoding();
CalcChain();
```

2. **ฟังก์ชันการสอน**

ทำการสอนลักษณะของวัตถุให้แก่โครงข่ายโดยเริ่มจากแสดงไดอะล็อกรับชนิดของวัตถุที่ทำการสอน

```
//void CProject1Doc::OnButtonTrain()
CDlgEnterClass Dlg;
if(Dlg.DoModal()!=IDOK) return;
```

หลังจากป้อนชนิดแล้ว ถ้ากดตกลง จะทำการเก็บชื่อไว้ใน InputName

```
InputName = Dlg.m_ClassName;
```

แล้วทำการดึงลักษณะของวัตถุออกมา และเปลี่ยนเป็นอินพุทเวคเตอร์ป้อนให้โครงข่าย ซึ่งส่วนของการปรับเปลี่ยนโครงข่ายในการ การเพิ่ม และลดรัศมี อธิบายไว้ในหัวข้อของการสอน

```
if(NumPrototype==0)
{
        rad = 0.5;
        //Insert = true;
}
else
{
        for(i=0;i<NumPrototype;i++)
        {
                result = 0;
                for(j=0;j<48;j++)
                {
                        result += (InputLayer.Data[j]-Network[i].Data[j])
                                *(InputLayer.Data[j]-Network[i].Data[j]);
                }
                if(result<=0.000000 ) // case : rename image's class
                {
                        Insert=false;
                        Network[i].Name = InputName;
                }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                else if(result<=Network[i].Rad && Network[i].Name==InputName)
                {
                        Insert = false;
                }
                else if(result>Network[i].Rad && Network[i].Name==InputName)
                {
                }
                else if(result<=Network[i].Rad && Network[i].Name!=InputName)
                {
                        result = result*9/10;
                        Network[i].Rad = result;
                        if(result<rad) rad = result;
                }
                else if(result>Network[i].Rad && Network[i].Name!=InputName)
                {
                        if(result<rad)
                        rad = result*9/10;
                }
        } // for
} // if else
```

3. ฟังก์ชันการแยกชนิดของวัตถุ

ทำการแยกชนิดของวัตถุ โดยเริ่มจากดึงลักษณะของวัตถุออกมาแล้ว เปลี่ยนลักษณะนั้นให้เป็นอินพุทเวคเตอร์ป้อนให้แก่โครงข่าย

```
//void CProject1View::OnButtonClass()
CProject1Doc* pDoc = GetDocument();
ASSERT_VALID(pDoc);
pDoc->GetTChainCode();
pDoc->CalcEachPiece(pDoc->TMaxRadPt);

for(i=0;i<12;i++)
{
        pDoc->InputLayer.Data[4*i] = pDoc->Piece[i].Area;
        pDoc->InputLayer.Data[4*i+1] = pDoc->Piece[i].Bound;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            pDoc->InputLayer.Data[4*i+2] = pDoc->Piece[i].MaxRad;
            pDoc->InputLayer.Data[4*i+3] = pDoc->Piece[i].MinRad;
    }
```

ส่วนของการแยกชนิดของวัตถุคือจะดูว่าอินพุทเวคเตอร์ที่รับเข้ามาตกลงในรัศมีของเวคเตอร์ของชนิดของวัตถุใด

```
    for(i=0;i<pDoc->NumPrototype;i++)
    {
            result = 0;
            for(j=0;j<48;j++)
            {
                    result += (pDoc->InputLayer.Data[j]-pDoc->Network[i].Data[j])
                                    *(pDoc->InputLayer.Data[j]-pDoc->Network[i].Data[j]);
            }
```

หากอินพุทเวคเตอร์ที่รับเข้ามาตกลงในรัศมีของเวคเตอร์ต้นแบบใด จะทำการเก็บชนิดไว้

```
            if(result<pDoc->Network[i].Rad)
            {
                    if(ImageName != pDoc->Network[i].Name)
                    {
                            ImageName = pDoc->Network[i].Name;
                            ทำการเก็บจำนวนโหนดเอาท์พุทที่แอคทีฟ
                            n++;
                    }
            }
    }
```

หลังจากทำการแยกชนิดวัตถุแล้ว โครงข่ายจะให้แสดงไดอะล็อกคำตอบออกมาอย่างใดอย่างหนึ่ง คือ ไม่รู้จัก หรือ บอกชนิดของวัตถุ หรือ บอกว่ามีโหนดเอาท์พุทที่แอคทีฟมากกว่าหนึ่งโหนด

```
    if(n==0)
            Str= "Don't know.";
    else if(n==1)
            Str.Format(_T("This image 's class is %s."),ImageName);
    else if(n>1)
            Str = "More than one class are active.";
    MessageBox(Str,"Classification");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ฟังก์ชันการแสดงโครงข่าย

ทำการแสดงค่าของนิวรอนโหนดในโครงข่าย ซึ่งใช้ลิสต์คอนโทรลที่หนึ่งแสดงค่าของอินพุทโหนด และใช้ลิสต์คอนโทรลที่สองแสดงค่าของโหนดชั้นซ่อน และระยะห่างระหว่างอินพุทเวคเตอร์กับเวคเตอร์ในโหนดชั้นซ่อน ดังนี้

```
//void CMainFrame::OnButtonShow()
แสดงไดอะล็อก
        CDlgShow Dlg;
        Dlg.DoModal();
```

ซึ่งในส่วนของการสร้างไดอะล็อก เป็นดังนี้

```
//BOOL CDlgShow::OnInitDialog()
ทำการสร้างคอร์ลัมให้แก่ลิสต์คอนโทรลที่หนึ่ง
for(i=0;i<48;i++)
{
        Str.Format("%d",i+1);
        m_List1.InsertColumn(i,Str,LVCFMT_LEFT,70);
}
ทำการสร้างคอร์ลัมให้แก่ลิสต์คอนโทรลที่สอง
m_List2.InsertColumn(0,"Dif",LVCFMT_LEFT,70);
m_List2.InsertColumn(1,"Rad",LVCFMT_LEFT,70);
m_List2.InsertColumn(2,"Name",LVCFMT_LEFT,70);
for(i=0;i<48;i++)
{
        Str.Format("%d",i+1);
        m_List2.InsertColumn(3+i,Str,LVCFMT_LEFT,70);
}

นำค่าของอินพุทโหนดซึ่งเก็บอยู่ใน Document มาแสดงในลิสต์คอนโทรลที่หนึ่ง
pDoc->GetTChainCode();
pDoc->CalcEachPiece(pDoc->TMaxRadPt);
for(i=0;i<12;i++)
{
        pDoc->InputLayer.Data[4*i] = pDoc->Piece[i].Area;
        pDoc->InputLayer.Data[4*i+1] = pDoc->Piece[i].Bound;
        pDoc->InputLayer.Data[4*i+2] = pDoc->Piece[i].MaxRad;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        pDoc->InputLayer.Data[4*i+3] = pDoc->Piece[i].MinRad;
    }
    Str.Format(_T("%lf"),pDoc->InputLayer.Data[0]);
    m_List1.InsertItem(0,Str);
    for(i=1;i<48;i++)
    {
        Str.Format(_T("%lf"),pDoc->InputLayer.Data[i]);
        m_List1.SetItemText(0,i,Str);
    }
```

นำค่าของโหนดชั้นซ่อนซึ่งเก็บอยู่ใน Document และคำนวณระยะห่างระหว่างอินพุทเวคเตอร์กับเวคเตอร์ในโหนดซ่อน มาแสดงในลิสต์คอนโทรลที่สอง

```
    for(j=0;j<48;j++)
    {
        คำนวณระยะห่างระหว่างอินพุทเวคเตอร์กับเวคเตอร์ในโหนดซ่อน
        temp += (pDoc->InputLayer.Data[j]-pDoc->Network[i].Data[j])
                *(pDoc->InputLayer.Data[j]-pDoc->Network[i].Data[j]);
    }
    Str.Format("%lf",temp);
    m_List2.InsertItem(0,Str);
    Str.Format(_T("%lf"),pDoc->Network[i].Rad);
    m_List2.SetItemText(0,1,Str);
    m_List2.SetItemText(0,2,pDoc->Network[i].Name);
    for(j=0;j<48;j++)
    {
        Str.Format(_T("%lf"),pDoc->Network[i].Data[j]);
        m_List2.SetItemText(0,3+j,Str);
    }
```

5. ฟังก์ชันการลบโครงข่าย

ทำการลบค่าในนิวรอนโหนดของโครงข่ายทั้งหมด ทั้งในชั้นอินพุท และในชั้นซ่อน ซึ่งถูกเก็บอยู่ใน Document

```
    //void CProject1View::OnButtonClear()
    pDoc->InputName ="";
    for(i=0;i<48;i++)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            pDoc->InputLayer.Data[i] = 0;
        for(i=0;i<pDoc->NumPrototype;i++)
        {
            pDoc->Network[i].Name = "";
            pDoc->Network[i].Rad = 0;
            for(j=0;j<48;j++)
                pDoc->Network[i].Data[j] = 0;
        }
        pDoc->NumPrototype = 0;
```

6. ฟังก์ชันการโหลดไฟล์ข้อมูลการสอน

มีทำหน้าที่โหลดไฟล์ข้อมูลการสอนมาใส่ในโครงข่าย ซึ่งเป็นไฟล์ที่เคยทำการเก็บของลักษณะต้นแบบของการสอนเดิมเอาไว้ เพื่อที่สามารถนำมาใช้แยกชนิดของวัตถุในครั้งต่อไปได้

```
//void CProject1Doc::OnButtonLoad()
```

ทำการแสดงไดอะล็อกเปิดไฟล์โดยแสดงเฉาะไฟล์ที่มีนามสกุล RCE เท่านั้น

```
CFile file;
CFileDialog dlg(true,"rce","*.rce",OFN_HIDEREADONLY,"Rce File (*.rce)|*.rce||");
if(dlg.DoModal()!=IDOK)
        return;
```

หากมีการรับชื่อไฟล์และกดตกลงจะเก็บชื่อไฟล์ไว้ในตัวแปร FileName

```
FileName = dlg.GetPathName();
```

ทำการเปิดไฟล์ที่ต้องการเพื่อทำการเขียนข้อมูลลงไฟล์

```
if(!file.Open(FileName,CFile::modeRead|CFile::typeBinary))
        return;
```

ทำการอ่านค่าจากไฟล์มาเก็บในโครงข่าย พร้อมกับทำการนับจำนวนของลักษณะต้นแบบด้วย

```
NumPrototype=0;
while(file.Read(&var, sizeof(var)))
{
        Network[NumPrototype].Name = (CString)var.Name;
        Network[NumPrototype].Rad=var.rad;
        for(i=0;i<48;i++)
            Network[NumPrototype].Data[i]=var.Data[i];
        NumPrototype++;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการปิดไฟล์

```
file.Close();
```

7. ฟังก์ชันการบันทึกไฟล์ข้อมูลการสอนเดิม

หลังจากโหลดไฟล์ใดขึ้นมาแล้วมีการปรับเปลี่ยนโครงข่าย จะสามารถที่จะบันทึกลักษณะต้นแบบในโครงข่ายกลับไปยังไฟล์ข้อมูลการสอนเดิมได้

```
//void CProject1View::OnButtonSave()
```

แสดงไดอะล็อกเพื่อถามว่าต้องการบันทึกทับไฟล์เดิมหรือไม่ เพื่อป้องกันกรณีที่กดปุ่มบันทึกโดยไม่ตั้งใจ ซึ่งถ้าไม่ต้องการก็สามารถกดปุ่มยกเลิกได้

```
if( MessageBox(" Do you want to save changes? ", NULL, MB_YESNO |
        MB_ICONEXCLAMATION ) != IDYES ) return;
```

ทำการเปิดไฟล์เพื่อทำการเขียนข้อมูลลงไฟล์

```
CFile file;
if(!file.Open(pDoc->FileName,CFile::modeCreate|CFile::modeWrite|CFile::typeBinary))
        return;
int i,j;
for(i=0;i<pDoc->NumPrototype;i++)
{
        for (j=0;j<pDoc->Network[i].Name.GetLength();j++)
                var.Name[j] = (char) pDoc->Network[i].Name.GetAt(j);
        var.Name[j]='\0';
        var.rad=pDoc->Network[i].Rad;
        for(j=0;j<48;j++)
                var.Data[j]=pDoc->Network[i].Data[j];
        file.Write(&var, sizeof(var));
}
```

ทำการปิดไฟล์

```
file.Close();
```

8. ฟังก์ชันการบันทึกไฟล์ข้อมูลการสอนลงไฟล์

ทำหน้าที่บันทึกลักษณะต้นแบบที่ทำการสอนไว้ลงในไฟล์ เริ่มจากแสดงไดอะล็อกการบันทึกไฟล์ โดยจะสามารถบันทึกเป็นไฟล์ได้เฉพาะนามสกุล RCE

```
//void CProject1Doc::OnButtonSaveAs()
CFile file;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CFileDialog dlg(false,"rce","*.rce",OFN_HIDEREADONLY,"Rec File (*.rce)|*.rce||");
if(dlg.DoModal()!=IDOK)
        return;
```

ทำการเปิดไฟล์เพื่อทำการบันทึก

```
if(!file.Open(dlg.GetPathName(),CFile::modeCreate|CFile::modeWrite|CFile::typeBinary))
        return;
```

ทำการบันทึกแต่ละลักษณะต้นแบบที่ทำการสอนไว้ในโครงข่าย

```
int i,j;
for(i=0;i<NumPrototype;i++)
{
        for (j=0;j<Network[i].Name.GetLength();j++)
                var.Name[j] = (char) Network[i].Name.GetAt(j);
        var.Name[j]='\0';

        var.rad=Network[i].Rad;
        for(j=0;j<48;j++)
                var.Data[j]=Network[i].Data[j];
        file.Write(&var, sizeof(var));
}
```

ทำการปิดไฟล์

```
file.Close();
```

9. ฟังก์ชันการแสดงผลลงบนหน้าจอวินโดวส์

ทำหน้าที่แสดงภาพ ของวัตถุที่โหลดมา และผลลัพธ์จากการทำงานต่างๆเช่น การทำภาพสองระดับ การหาขอบของวัตถุ การดึงและเข้ารหัสลักษณะของวัตถุ ซึ่งทำการแสดงโดยใช้ทั้งการวาด และตัวหนังสือ

```
//void CProject1View::OnDraw(CDC* pDC)
```

ทำการตั้งค่ารูปแบบอักษรที่ใช้แสดงผลบน View

```
CFont fontText;
fontText.CreateFont(10, 0, 0, 0, 400, false, false, 0,
        ANSI_CHARSET, OUT_DEFAULT_PRECIS, CLIP_DEFAULT_PRECIS,
        DEFAULT_QUALITY, DEFAULT_PITCH | FF_SWISS, " Arial " );
pDC     ->SelectObject(&fontText);
```

ทุกครั้งที่มีการประมวลผลจะมีแสดงภาพวัตถุ ผลการทำภาพสองระดับ การหาขอบ

```
pDC->TextOut(55,135,"IMAGE");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
pDC->TextOut(165,135,"IMAGE SEGMENTATION");
pDC->TextOut(330,135,"EDGE DETECTION");
แสดงภาพวัตถุ
::StretchDIBits(pDC->GetSafeHdc(),
        0,
        0,
        pDoc->m_pDIB->m_pBMI->bmiHeader.biWidth,
        pDoc->m_pDIB->m_pBMI->bmiHeader.biHeight,
        0,
        0,
        pDoc->m_pDIB->m_pBMI->bmiHeader.biWidth,
        pDoc->m_pDIB->m_pBMI->bmiHeader.biHeight,
        pDoc->m_pDIB->GetBMBitsPtr(),
        pDoc->m_pDIB->GetBMHdrPtr(),
        DIB_RGB_COLORS,
        SRCCOPY);
แสดงภาพสองระดับของวัตถุด้วย สีของระดับสีเทา
::StretchDIBits(pDC->GetSafeHdc(),
        150,
        0,
        pDoc->m_pDIB->m_pBMI->bmiHeader.biWidth,
        pDoc->m_pDIB->m_pBMI->bmiHeader.biHeight,
        0,
        0,
        pDoc->m_pDIB->m_pBMI->bmiHeader.biWidth,
        pDoc->m_pDIB->m_pBMI->bmiHeader.biHeight,
        pDoc->SegmentTbl,
        pDoc->m_pDIB->GetBMHdrPtr(),
        DIB_RGB_COLORS,
        SRCCOPY);
แสดงภาพขอบของวัตถุ ด้วยสีของระดับสีเทา
::StretchDIBits(pDC->GetSafeHdc(),
        300,
        0,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
pDoc->m_pDIB->m_pBMI->bmiHeader.biWidth,
pDoc->m_pDIB->m_pBMI->bmiHeader.biHeight,
0,
0,
pDoc->m_pDIB->m_pBMI->bmiHeader.biWidth,
pDoc->m_pDIB->m_pBMI->bmiHeader.biHeight,
pDoc->EdgeTbl,
pDoc->m_pDIB->GetBMHdrPtr(),
DIB_RGB_COLORS,
SRCCOPY);
```

และผลลัพธ์ของการทำงานอื่นๆแล้วแต่กรณีดังนี้

หลังจากทำการสอนจะมีการแสดงการจุดศูนย์ถ่วง จุดอ้างอิงของวัตถุ และการแสดงกราฟลักษณะของวัตถุที่ทำการสอนเข้าไป โดยแกนตั้งคือรัศมี แกนนอนคือมุม

ทำการวนลูปเพื่อแสดงค่าต่างๆของทุกจุดอ้างอิงที่ใช้สอนของวัตถุ

```
for(i=0;i<pDoc->NumBasePt;i++)
{
```

หาจุดอ้างอิงแต่ละจุดของวัตถุ

```
Base = pDoc->BasePt[i];
GetVChainCode();
CalcVChainRotate(VChainCode[Base].Deg);
memcpy(Tbl,pDoc->EdgeTbl,16384);
```

ทำการวาดจุดอ้างอิง

```
for(l=0;l<5;l++)
{
    Tbl[(int)VChainCode[Base].Y*128+(int)VChainCode[Base].X+l] = 0;
    Tbl[(int)VChainCode[Base].Y*128+(int)VChainCode[Base].X-l] = 0;
    Tbl[(int)(VChainCode[Base].Y-l)*128+(int)VChainCode[Base].X] = 0;
    Tbl[(int)(VChainCode[Base].Y+l)*128+(int)VChainCode[Base].X] = 0;
}
```

ทำการวาดจุดศูนย์ถ่วง

```
for(l=0;l<5;l++)
{
    Tbl[(int)pDoc->CentroidY*128+(int)pDoc->CentroidX+l] = 0;
    Tbl[(int)pDoc->CentroidY*128+(int)pDoc->CentroidX-l] = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                Tbl[(int)(pDoc->CentroidY-l)*128+(int)pDoc->CentroidX] = 0;
                Tbl[(int)(pDoc->CentroidY+l)*128+(int)pDoc->CentroidX] = 0;
        }
```

แสดงภาพขอบของวัถุ

```
        ::StretchDIBits(pDC->GetSafeHdc(),
                0,
                sy-128+i*150,
                pDoc->m_pDIB->m_pBMI->bmiHeader.biWidth,
                pDoc->m_pDIB->m_pBMI->bmiHeader.biHeight,
                0,
                0,
                pDoc->m_pDIB->m_pBMI->bmiHeader.biWidth,
                pDoc->m_pDIB->m_pBMI->bmiHeader.biHeight,
                Tbl,
                pDoc->m_pDIB->GetBMHdrPtr(),
                DIB_RGB_COLORS,
                SRCCOPY);
```

ทำการวาดตารางกราฟ

```
        for(k=0;k<360;k++)
                for(j=0;j<110;j+=10)
                        pDC->SetPixel(sx+k,sy-j+i*150,RGB(0,255,255));

        for(k=0;k<390;k+=30)
                for(j=0;j<100;j++)
                        pDC->SetPixel(sx+k,sy-j+i*150,RGB(0,255,255));
        pDC->MoveTo(sx,sy-110+i*150);
        pDC->LineTo(sx,sy+i*150);
        pDC->LineTo(sx+370,sy+i*150);

        for(k=0;k<10;k++)
                for(j=10;j<110;j+=10)
                        pDC->SetPixel(sx-k,sy-j+i*150,RGB(0,0,0));

        for(k=30;k<390;k+=30)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                for(j=0;j<5;j++)
                        pDC->SetPixel(sx+k,sy+j+i*150,RGB(0,0,0));
```

ทำการเขียนข้อความอธิบายกราฟ

```
        pDC->TextOut(sx-5,sy-125+i*150,"Radius");
        pDC->TextOut(sx+380,sy-5+i*150,"Degree");
        int n;
        for(n=0;n<11;n++)
        {
                s.Format("%1.1lf",(double)n/10);
                pDC->TextOut(sx-30,sy-n*10-5+i*150,s);
        }
        for(n=0;n<390;n+=30)
        {
                s.Format("%d",n);
                pDC->TextOut(sx+n-5,sy+10+i*150,s);
        }
```

ทำการวาดข้อมูลของจุดขอบในรูปของพิกัดเชิงขั้วลงบนตารางกราฟ

```
        for(j=0;j<VNumChainPt;j++)
        {
                pDC->SetPixel(sx + (int) VChainCode[j].Deg,
                                sy - (int) (VChainCode[j].RadNrml*100-i*150),
                                RGB(0,0,0));
        }
}
```

สำหรับกรณีที่ทำการแยกชนิดของวัตถุ จะแสดงข้อมูลต่างๆเหมือนกับการสอน ต่างกันตรงที่จะแสดงเฉพาะข้อมูลที่เกิดจากจุดอ้างอิงที่มีรัศมีเท่ากับ 1.0 เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก จ
การใช้งานโปรแกรม

1. ความต้องการของโปรแกรม

ระบบปฏิบัติการ Window 95/ 98/ NT

2. การติดตั้ง

ทำการสำเนาไฟล์ Project1.exe ลงในเครื่องคอมพิวเตอร์ เท่านี้ก็สามารถใช้งานโปรแกรมได้

3. การคอมไพล์โปรแกรม

ทำการสำเนาไฟล์ทั้งหมดลงในโฟลเดอร์เดียวกัน จากนั้นเปิดโปรแกรมด้วย Microsoft Visual C++ 6.0 โดยเลือกเมนู File\ Open Work Space... จะปรากฏไฟล์ไดอะล็อก ให้เลือกไฟล์ Porject1.dws หลังจากโปรแกรมถูกโหลดเสร็จ เมื่อต้องการคอมไพล์ให้เลือก Build\Compile

4. การใช้งานโปรแกรม

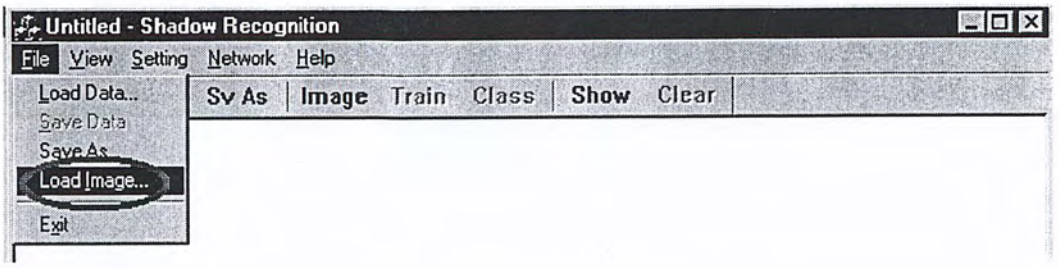
การใช้งานโปรแกรม ผู้ใช้ควรศึกษาวิธีการที่โปรแกรมใช้ในการเรียนรู้และการแยกชนิดของวัตถุจากปริญญานิพนธ์มาก่อน เพื่อจะได้มีความเข้าใจในที่จะใช้งานโปรแกรมได้ง่ายขึ้น

เมื่อต้องการใช้โปรแกรมทำการรู้จำวัตถุนั้น เริ่มจากให้โปรแกรมเรียนรู้ลักษณะเด่นของวัตถุก่อน โดยเปิดไฟล์ภาพโครงร่างเงาวัตถุซึ่งเป็นไฟล์ภาพระดับสีเทา 256 ระดับสีของวัตถุที่วางอยู่บนฉากขาว โปรแกรมจะดึงลักษณะเด่นของวัตถุมาเก็บไว้ หลังจากนั้นทำการสอนเข้าไปในโครงข่าย โดยปริมาณของภาพวัตถุที่นำมาสอนจะขึ้นกับคล้ายกันของวัตถุ และจำนวนชนิดที่ต้องการให้โปรแกรมทำการแยกชนิดของวัตถุ คือถ้าวัตถุที่ความคล้ายกันน้อย หรือมีวัตถุไม่กี่ชนิด ความจำเป็นในการใช้ภาพวัตถุก็จะน้อย แต่ถ้าวัตถุมีความคล้ายกันมาก หรือมต้องการแยกวัตถุหลายชนิด ก็ต้องใช้ภาพวัตถุจำนวนมากขึ้น เมื่อทำการสอนเสร็จแล้ว ก็สามารถแยกชนิดของวัตถุที่เรียนรู้เข้าไปได้

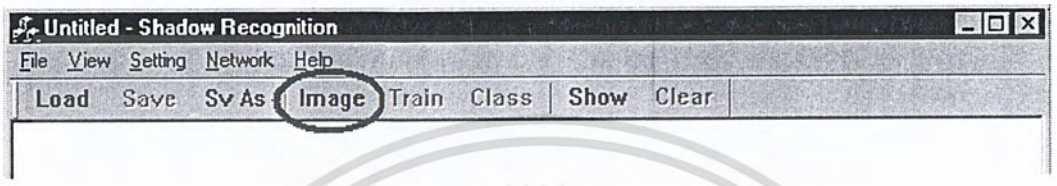
5. การโหลดภาพวัตถุ

โดยการเลือกเมนู File \ Load Image ดังรูปที่ 1จ หรือเลือกทูลบาร์ Image หรือกด i ดังรูปที่ 2จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

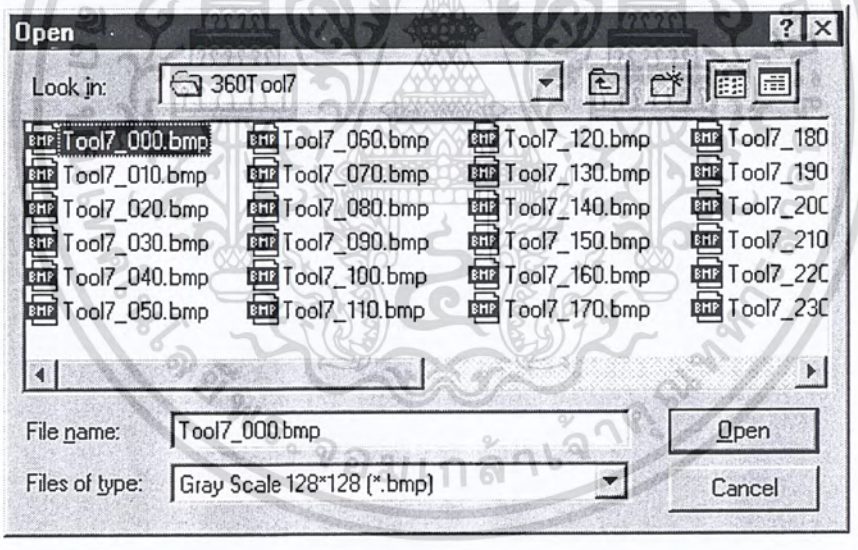


รูปที่ 1จ แสดงเมนู Load Image



รูปที่ 2จ แสดงทูลบาร์ Image

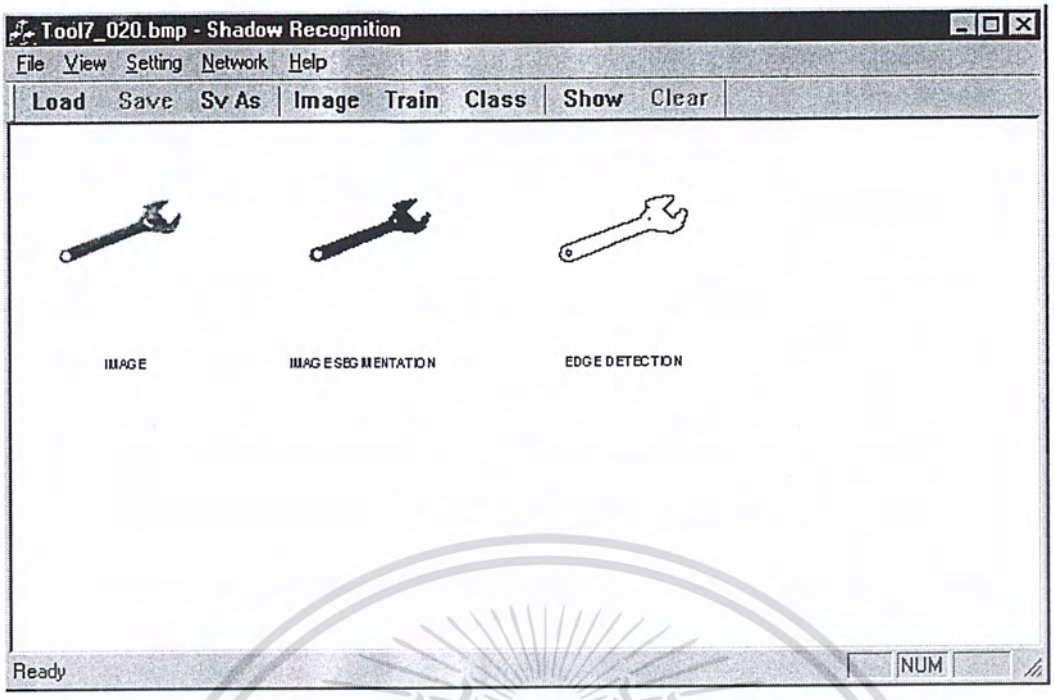
โปรแกรมแสดงไดอะล็อกการเปิดไฟล์ขึ้นมา ซึ่งจะแสดงเฉพาะไฟล์นามสกุล *.bmp ดังรูปที่ 3จ



รูปที่ 3จ แสดงไดอะล็อกการเปิดภาพ

ทำการเลือกไฟล์วัตถุที่ต้องการ โปรแกรมจะทำการโหลดและประมวลผลภาพต่างๆและให้ผลลัพธ์ ดังรูปที่ 4จ

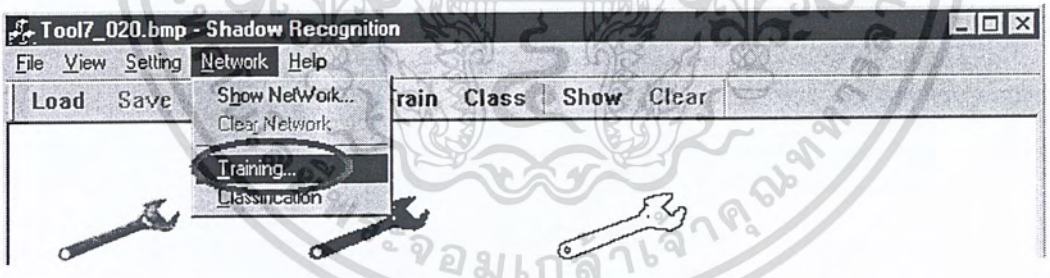
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



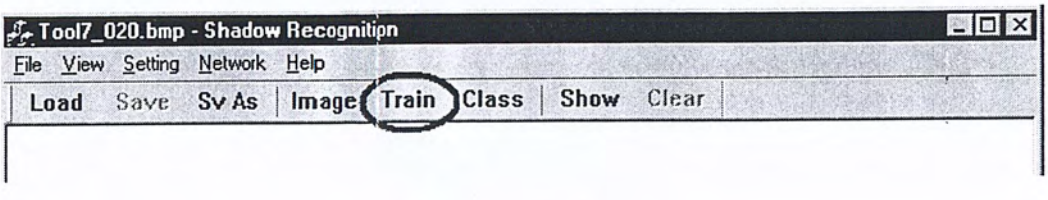
รูปที่ 4จ แสดงผลการประมวลภาพ

6. การสอนลักษณะของวัตถุให้โครงข่าย

โดยการเลือกเมนู Network \ Training ดังรูปที่ 5จ หรือเลือกทูลบาร์ Train หรือกด t ซึ่งจะสามารถสอนได้หลังจากมีการโหลดวัตถุแล้วเท่านั้น ดังรูปที่ 6จ



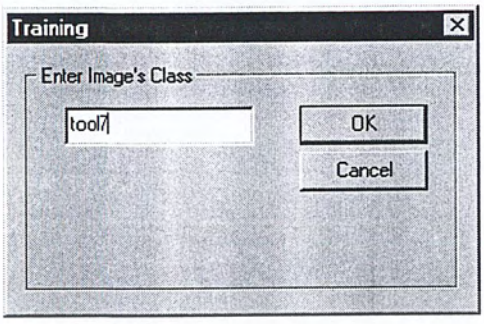
รูปที่ 5จ แสดงเมนู Training



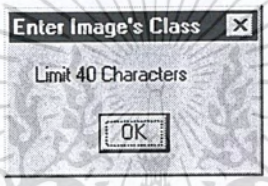
รูปที่ 6จ แสดงทูลบาร์ Train

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะปรากฏไดอะล็อกรับชนิดของวัตถุ ดังรูปที่ 7จ ให้ทำการป้อนชนิดวัตถุลงไปโดยความยาวไม่เกิน 40 อักขระหากป้อนเกิน 40 อักขระจะแสดงข้อความเตือน ดังรูปที่ 8จ

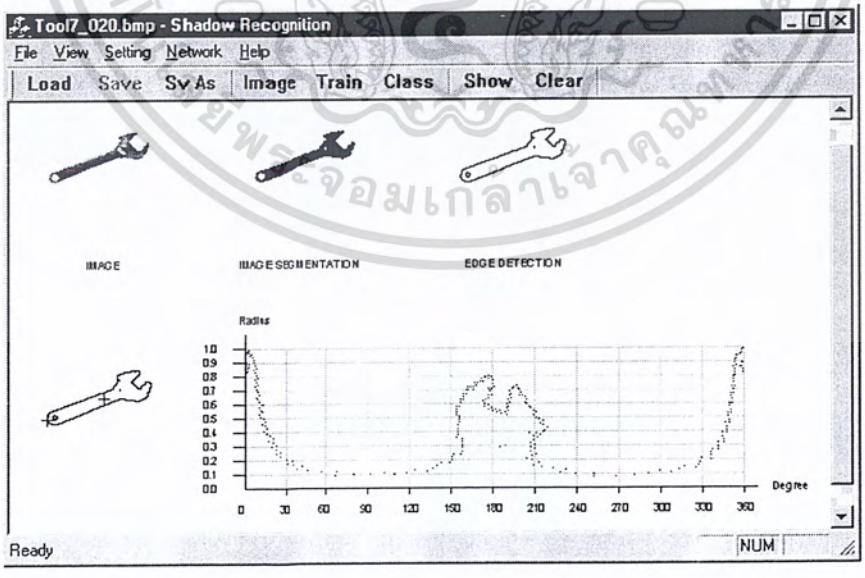


รูปที่ 7จ แสดงไดอะล็อกรับชนิดของวัตถุ



รูปที่ 8จ แสดงข้อความเตือนหากป้อนอักขระเกิน 40 ตัว

หลังจากป้อนชื่อแล้วกดตกลง โปรแกรมจะแสดงลักษณะเด่นของวัตถุที่ดึงมาสอนให้โครงข่าย ดังรูปที่ 9จ

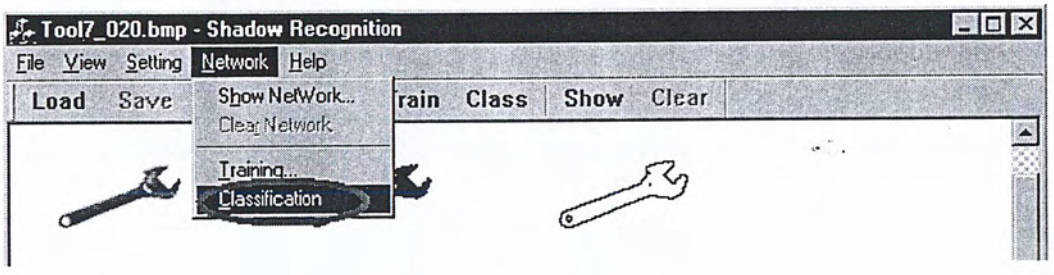


รูปที่ 9จ แสดงลักษณะเด่นที่ใช้สอน

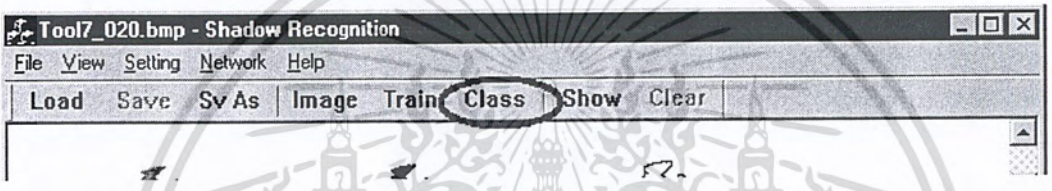
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. การแยกชนิดวัตถุ

โดยการเลือกเมนู Network \ Classification ดังรูปที่ 10จ หรือเลือกทูลบาร์ Class หรือกด c ซึ่งจะสามารถแยกชนิดของวัตถุได้หลังจากมีการโหลดวัตถุแล้วเท่านั้น ดังรูปที่ 11จ



รูปที่ 10จ แสดงเมนู Classification



รูปที่ 11จ แสดงทูลบาร์ Class

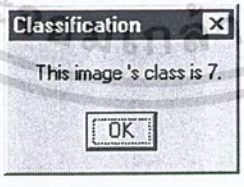
โปรแกรมจะแสดงข้อความต่างๆ ตัวอย่างในการแยกชนิดของวัตถุ ดังรูปที่ 12จ

หากชนิดของวัตถุแอคทีพเพียงชนิดเดียวจะแสดงข้อความ "This Image 's Class is Tool6."

หากไม่มีชนิดของวัตถุใดแอคทีพ จะแสดงข้อความ "Don't Know."

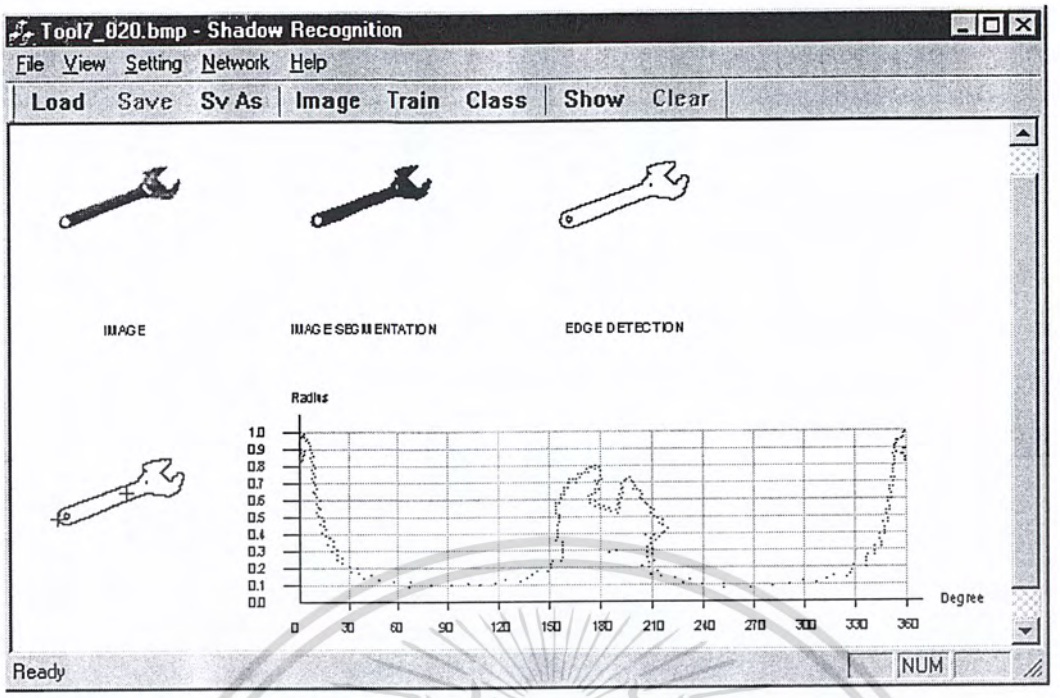
หากชนิดของวัตถุแอคทีพมากกว่าชนิดเดียวจะแสดงข้อความ "More Tham One Output Node Are Active."

และแสดงลักษณะเด่นของวัตถุที่ดึงมาใช้แยกชนิดวัตถุ ดังรูปที่ 13จ



รูปที่ 12จ แสดงผลของการแยกชนิดวัตถุ

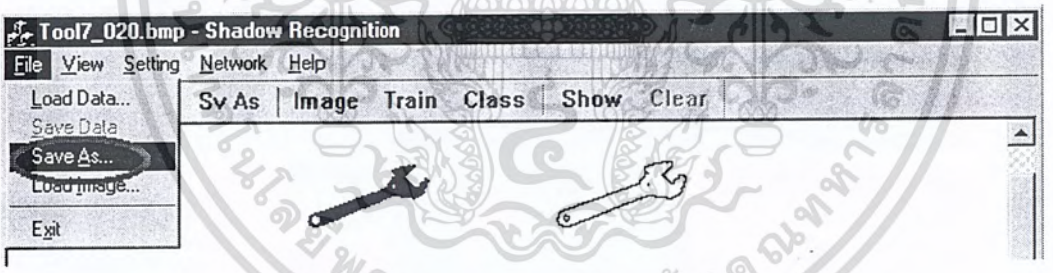
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



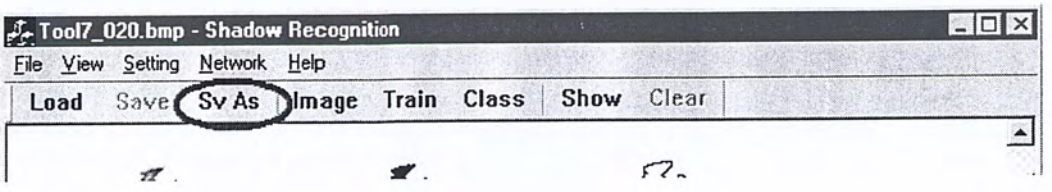
รูปที่ 13จ แสดงลักษณะที่ใช้ในการแยกชนิดวัตถุ

8. การบันทึกลักษณะที่ทำการเรียนรู้ลงไฟล์

โดยการเลือกเมนู File \ Save As ดังรูปที่ 14จ หรือกดทูลบาร์ Sv As หรือกด a ดังรูปที่ 15จ



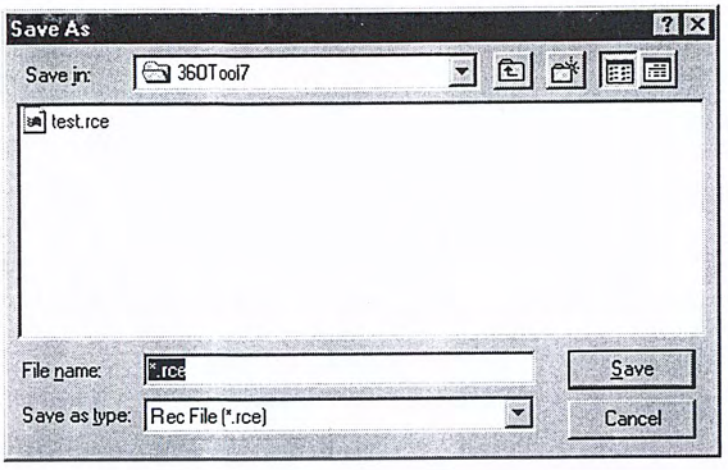
รูปที่ 14จ แสดงเมนู Save as



รูปที่ 15จ แสดงทูลบาร์ Sv As

หลังจากนั้นโปรแกรมจะแสดงไดอะล็อกการบันทึกไฟล์เป็นไฟล์นามสกุล RCE ดังแสดงในรูปที่ 16จ

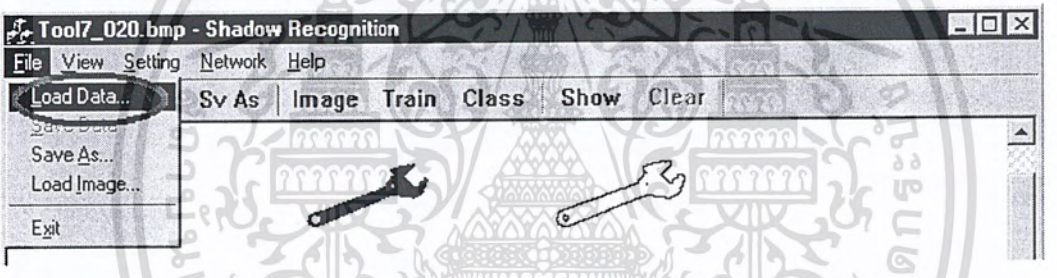
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



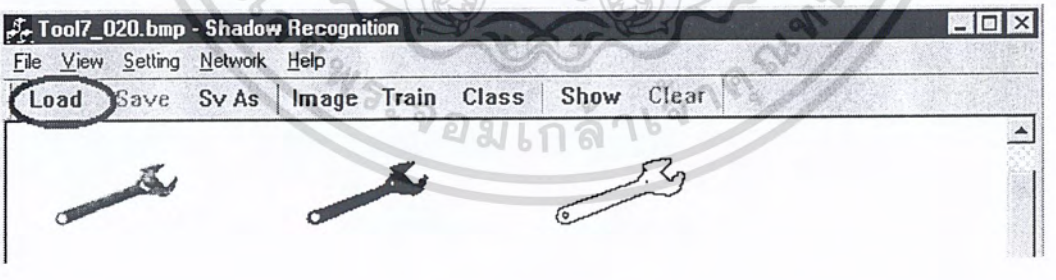
รูปที่ 16จ แสดงไดอะล็อกการบันทึกไฟล์ลักษณะของวัตถุ

9. การโหลดความรู้ลักษณะของวัตถุจากไฟล์

โดยการเลือกเมนู File \ Load Data ดังรูปที่ 17จ หรือกดทูลบาร์ Load หรือกด 1 ดังรูปที่ 18จ



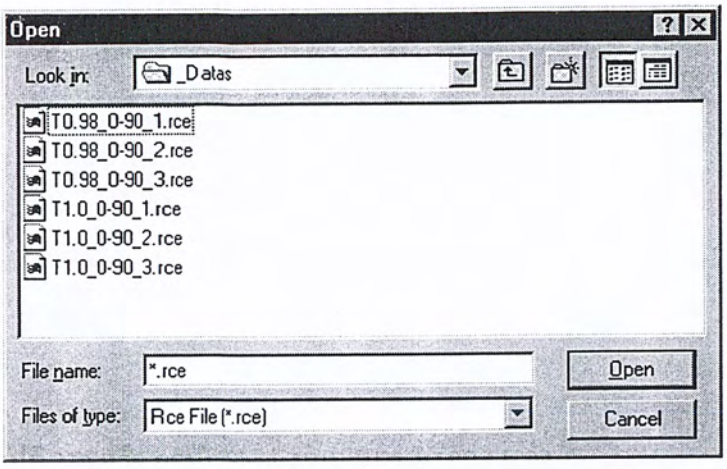
รูปที่ 17จ แสดงเมนู Load Data



รูปที่ 18จ แสดงทูลบาร์ Load

โปรแกรมจะแสดงไดอะล็อกการเปิดไฟล์ข้อมูล เพื่อให้ทำการเลือกไฟล์ลักษณะข้อมูลที่ต้องการ ดังรูปที่ 19จ

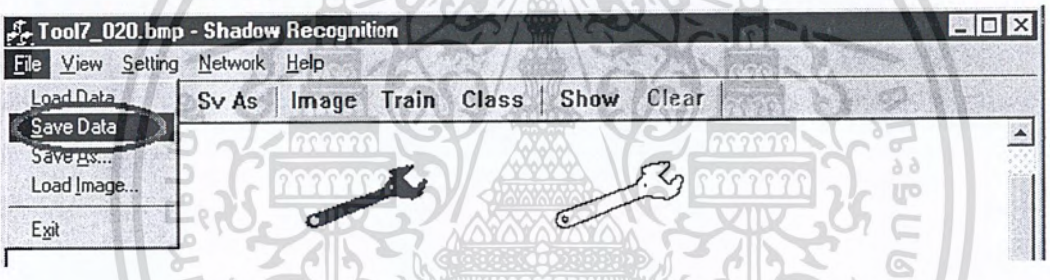
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



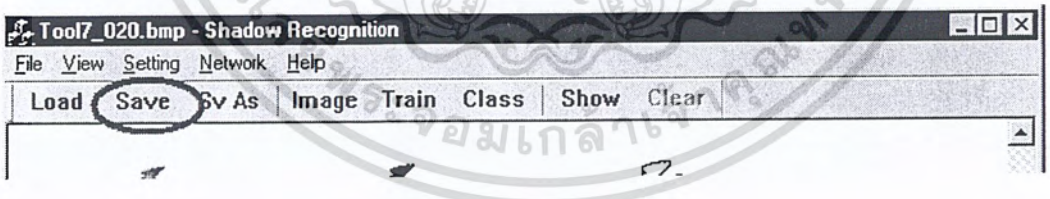
รูปที่ 19จ แสดงไดอะล็อกการเปิดไฟล์ลักษณะของวัตถุ

10. การบันทึกลักษณะที่ทำการเรียนรู้ลงไฟล์เดิม

โดยการเลือกเมนู File \ Save Data ดังรูปที่ 20จ หรือกดทูลบาร์ Save หรือกด s ดังรูปที่ 21จ



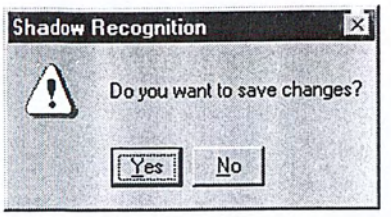
รูปที่ 20จ แสดงเมนู Save Data



รูปที่ 21จ แสดงทูลบาร์ Save

ซึ่งโปรแกรมจะแสดงข้อมูลขึ้นยันความต้องการบันทึกการเปลี่ยนแปลง ดังรูปที่ 22จ หากเรากดตกลงก็จะทำการบันทึกทับ

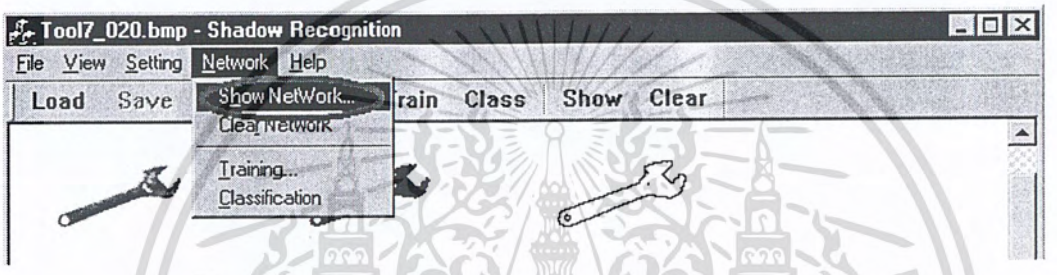
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



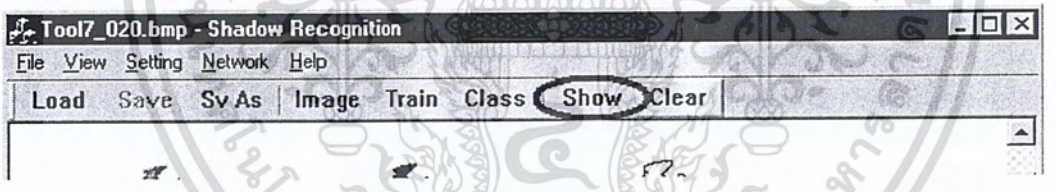
รูปที่ 22จ แสดงข้อความถามยืนยันการบันทึกการเปลี่ยนแปลง

11. การแสดงโครงข่าย

โดยการเลือกเมนู Network / Show Network ดังรูปที่ 23จ หรือ กดปุ่ม Show บนทูลบาร์ หรือกด h ดังรูปที่ 24จ



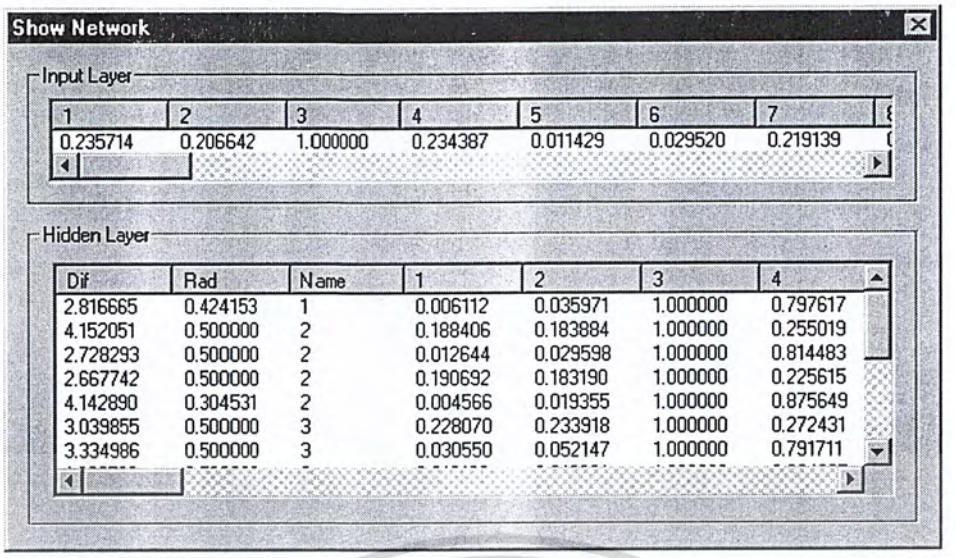
รูปที่ 23จ แสดงเมนู Show Network



รูปที่ 24จ แสดงทูลบาร์ Show

ซึ่งโปรแกรมจะแสดงไดอะล็อกแสดงข้อมูลภายในโครงข่ายออกมา ดังรูปที่ 25จ

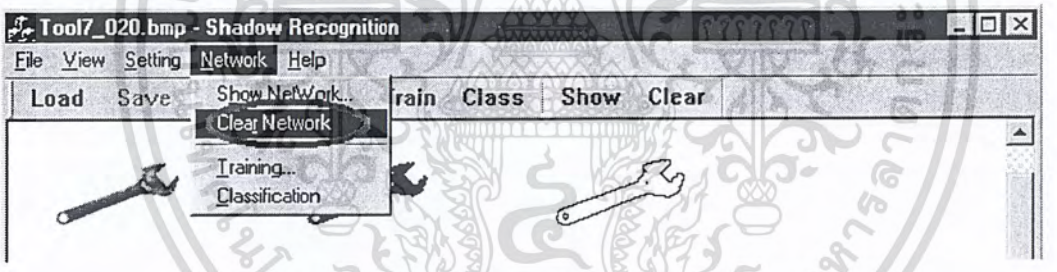
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



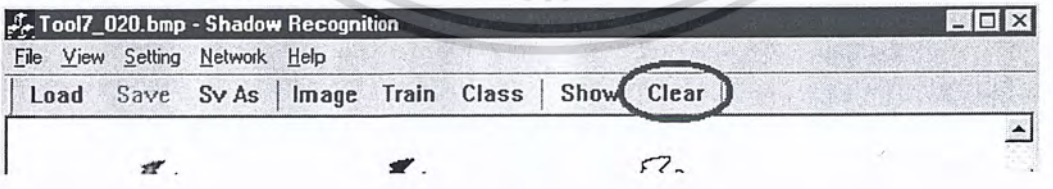
รูปที่ 25จ แสดงไดอะล็อกแสดงข้อมูลภายในโครงข่าย

12. การลบข้อมูลในโครงข่าย

โดยการเลือกเมนู Network / Clear Network ดังรูปที่ 26จ หรือกดปุ่ม Clear บนทูลบาร์ หรือกด r ดังรูปที่ 27จ



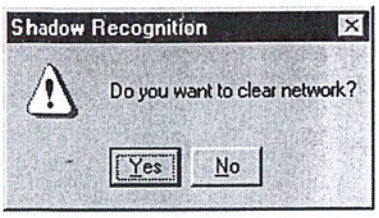
รูปที่ 26จ แสดงเมนู Clear Network



รูปที่ 27จ แสดงทูลบาร์ Clear

โปรแกรมจะแสดงไดอะล็อกยืนยันการลบข้อมูลในโครงข่าย ดังรูปที่ 28จ หากกดตกลงโปรแกรมก็จะทำการลบข้อมูลในโครงข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 28จ แสดงข้อความถามยืนยันการลบข้อมูลในโครงข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] Martin T.Hagan , Howard B. demuth and Mark Beal. :"*Neural Network Design*", PWS Publishing Company, 1996

[2] Ahlberg, J.H., E, N.Nilson, J.L Walsh, "*The Theory of Splines and Their Applications.*" New York, Academic Press, 1967

[3] Ballard, D.H. and C.M. Brown, "*Computer Vision*", Englewood Cliff, New Jersey, Prentic-hall, Inc., 1982

[4] D. Marr, Vision." *A Computational Investigation into the Human Representation &Processing of Vision Information*" , San Francisco, CA : Freeman, 1982

[5] D. Marr and E. Hildreth, "*Theory of edge detection*", in Proc. Roy. Soc. London, vol.-B207, pp. 187-217, 1980

[6] L. S . Davis, "*Understanding shape: Angles and sides* ", IEEE Trans Comput. Vol C-26, no 23, pp. 236-242, 1977

[7] H Samet., "*The Design and Analysis of Spatial Data Structures.*", Addison-Wisley, Reading, New York, 1982

[8] T Pavliidis. "*Algorithms for Graphics and Image Processing*, "Computer Science Press, New York, 1982

[9] W. k. Pratt, "*Digital Image Processing*", Wiley, New York, 1978

[10] Robert J. Schalkoff, "*Pattern Recognition*", New York: John Wiley &Son Inc., 1992

[11] Wei Li and Nasser M. Nasrabadi, "*Invariant Object Recognition Based On A Neural Network of Cascaded RCE Nets*", IEEE 171-184

[12] Y-H. Pao, "*Adaptive Pattern Recognition and Neural Networks*", Addison-Wesley, New York, 1989

[13] D. L. Reilly, L. N. Cooper and C. Elbaum, "*A neural model for category learning*",Biol. Cybern, Vol 45, pp 35-41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้