

การรู้จำรูปร่าง 2 มิติ โดยใช้วิธีการแทนด้วยพื้นที่สามเหลี่ยม

TRIANGLE AREA REPRESENTATION FOR 2D SHAPE RECOGNITION



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของงานวิจัยที่สนับสนุนโดยศูนย์วิจัยวิศวกรรมศาสตร์และเทคโนโลยี

ศูนย์วิจัยวิศวกรรมอิเล็กทรอนิกส์

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2551

KMITL-2008-EN-M-040-830

การรู้จำรูปร่าง 2 มิติ โดยใช้วิธีการแทนด้วยพื้นที่สามเหลี่ยม

TRIANGLE AREA REPRESENTATION FOR 2D SHAPE RECOGNITION



กฤษณา ยาวิชัย

KRITSANA YAWICHAI

เลขหมู่.....
เลขทะเบียน..... 82745
วัน,เดือน,ปี..... 22 ก.ค. 2551

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

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์
บัณฑิตวิทยาลัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และเผยแพร่ไปยังผู้อื่นถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
พ.ศ.2551

KMITL-2008-EN-M-040-330

TRIANGLE AREA REPRESENTATION FOR 2D SHAPE RECOGNITION



A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING IN ELECTRONICS ENGINEERING
SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่ King Mongkut's Institute of Technology Ladkrabang ขอสงวนสิทธิ์ในลิขสิทธิ์และสงวนไว้เพื่อใช้ในการศึกษาและการวิจัยเท่านั้น

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และเผยแพร่หรือแจกจ่ายแก่บุคคลอื่นโดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2008
KMITL-2008-EN-M-040-330



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

COPYRIGHT 2008

ไม่ว่ากรณีใดๆ ก็ตาม หากมีการแก้ไขหรือเปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ การรู้จำรูปร่าง 2 มิติ โดยใช้วิธีการแทนด้วยพื้นที่สามเหลี่ยม
Triangle area Representation for 2D Shape Recognition

นักศึกษา นางสาวกฤษณา ยาวีไชย

รหัสประจำตัว 49060420

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา วิศวกรรมอิเล็กทรอนิกส์

อาจารย์ที่ปรึกษาวิทยานิพนธ์ ผศ.ดร.ยุทธนา กิจใจเดียว

คณะกรรมการสอบวิทยานิพนธ์		ลายมือชื่อ
รศ.ดร.สุรพันธุ์	เอื้อไพฑูลย์	
รศ.ดร.ชูชาติ	ปิณฑวิรุจน์	
รศ.ดร.โกสินทร์	จ่านงไทย	
รศ.ดร.สมศักดิ์	ชุมช่วย	
ผศ.ดร.ยุทธนา	กิจใจเดียว	

วัน/เดือน/ปี ที่สอบ 12 พฤษภาคม 2551 เวลา 11.30-13.30 น.

สถานที่สอบ ณ ห้องประชุม 3 ชั้น 5 อาคาร A


บัณฑิตวิทยาลัยรับรองแล้ว
(รศ.ดร.รวีวรรณ ชินะตระกูล)
คณบดีบัณฑิตวิทยาลัย

วันที่ ๒๑ เดือน พฤษภาคม พ.ศ. ๒๕๕๑

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้า ไม่อนุญาตให้เผยแพร่หรือใช้ซ้ำโดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การรู้จำรูปร่าง 2 มิติ โดยใช้วิธีการแทนด้วยพื้นที่สามเหลี่ยม
นักศึกษา	นางสาวกฤษณา วิชาวิไชย
รหัสประจำตัว	49060420
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมอิเล็กทรอนิกส์
พ.ศ.	2551
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ผศ.ดร.บุษรนา คิดใจเดียว

บทคัดย่อ

การรู้จำรูปร่าง 2 มิติโดยใช้การประมวลผลภาพได้ถูกนำไปประยุกต์ใช้งานอย่างแพร่หลายซึ่งปัญหาส่วนใหญ่ที่เกิดขึ้นในการรู้จำคือ ข้อจำกัดในการรู้จำรูปร่างที่ได้จากมุมมองต่าง ๆ กันซึ่งจะมีลักษณะขนาด มุม ตำแหน่ง และการเอียงที่เปลี่ยนแปลงไปจากเดิม ด้วยเหตุนี้จะทำให้เกิดความผิดพลาดในการรู้จำได้ เพื่อลดผลกระทบจากกรณีดังกล่าว เราต้องการคุณลักษณะที่สำคัญจากรูปร่างเพื่อใช้เป็นตัวแทนของวัตถุ ซึ่งคุณลักษณะเฉพาะที่เป็นตัวแทนรูปร่างของวัตถุที่คั่นควรจะมีความทนต่อสัญญาณรบกวน และไม่เปลี่ยนแปลงต่อการแปลงแบบแอฟไฟน์ ในงานวิจัยนี้เราจึงทำการพัฒนาตัวอธิบายรูปร่างให้มีคุณสมบัติดังกล่าว เพื่อนำมาใช้ในการรู้จำรูปร่าง 2 มิติ ตัวอธิบายรูปร่างในงานวิจัยนี้ได้มาจากการนำเอาจุดบนเส้นรอบรูปมาคำนวณหาพื้นที่สามเหลี่ยมที่ค่าความยาวแขนต่าง ๆ กัน โดยอาศัยทฤษฎีทอริมิแนนท์ เราได้ทดสอบประสิทธิภาพในการรู้จำรูปร่างที่ถูกแปลงแบบแอฟไฟน์ในหลายๆรูปแบบ ด้วยวิธีนอร์มอลไลซ์ครอสคอร์รีเลชัน และโครงข่ายประสาทเทียม เพื่อพิสูจน์ว่าการแทนด้วยพื้นที่สามเหลี่ยมนั้นมีคุณสมบัติไม่ผันแปรต่อการแปลงแบบแอฟไฟน์ จากผลการทดลองแสดงให้เห็นว่าการแทนด้วยพื้นที่สามเหลี่ยมมีคุณสมบัติดังกล่าวอีกทั้งยังทนต่อสัญญาณรบกวนและการเปลี่ยนแปลงจุดเริ่มต้น ซึ่งส่งผลให้มีประสิทธิภาพในการรู้จำที่ดี นอกจากนี้เรายังนำเสนอการทนายหาคุณลักษณะที่ดีด้วยทฤษฎีพีชเชอร์ดีสคริมิแนนท์โดยไม่ต้องผ่านขบวนการรู้จำ ทำให้ลดเวลาในการหาคุณลักษณะที่ดีที่ใช้อธิบายรูปร่าง ซึ่งการทนายดังกล่าวสอดคล้องกับผลที่ได้จากขบวนการรู้จำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Triangle Area Representation for 2D Shape Recognition
Student	Miss Kritsana Yawichai
Student ID.	49060420
Degree	Master of Engineering
Program	Electronics Engineering
Year	2008
Thesis Advisor	Asst.Prof.Dr.Yuttana Kitjaidure

ABSTRACT

2D shape recognition through image processing has appeared in many applications. The main difficulty in recognition is the limitation of recognizing a 2D shape under arbitrary viewpoints. This shape can be in any kind of translation, scaling, rotation, and shear. As a result, the error may arise in the recognition process. To alleviate this, we have to extract the significant feature to represent the object. The feature that well represents the object should have property of noise tolerance and invariant to affine transform. In this work, we propose an affine invariant feature to use with 2D shape recognition. To find this shape representation, the coordinates of shape boundary are considered in computing the triangle area representation at different triangle side length according to the determinant theory. To prove the affine invariant property, we construct the experiment to measure recognition performance over a number of affine distorted shapes through 2 different recognition methods, Normalized Cross-correlation and Artificial Neural Networks. The experiment results show that the triangle area representation is effective and robustly tolerates translation, scaling, shear, orientation, starting point selection and noise. Moreover, we present the prediction of the optimum shape representation based on Fisher's discriminant analysis without considering through the whole recognition process. By this way, the large number of steps in finding the best shape representation is greatly reduced. The experiment results indicate that Fisher's discriminant analysis can be used to make accurate prediction corresponding with the recognition results.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์เล่มนี้สำเร็จได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา ผศ.ดร.ยุทธนา กิจใจเดียว ที่ให้ความช่วยเหลือ ให้คำชี้แนะ ช่วยแก้ปัญหา ตลอดจนให้ความรู้และประสบการณ์ที่ดีแก่ข้าพเจ้า ขอขอบพระคุณอาจารย์ทุกๆ ท่าน ที่ได้อบรมสั่งสอนและให้ความรู้ที่มีค่ายิ่งต่อการทำงาน ขอขอบคุณพี่ๆ เพื่อนๆ และน้องๆ ทุกคนที่คอยให้กำลังใจและช่วยเหลือในการทำงาน สุดท้ายต้องขอกราบขอบพระคุณบิดา มารดา พี่ น้อง และญาติๆ ของข้าพเจ้า ที่คอยเป็นกำลังใจที่ดีตลอดมา

สำหรับคุณงามความดีอันใดที่เกิดจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบให้กับบิดามารดา ซึ่งเป็นที่รักและเคารพยิ่ง ตลอดจนครูอาจารย์ที่เคารพทุกท่านที่ได้ถ่ายทอดความรู้และประสบการณ์ที่ดีให้แก่ข้าพเจ้า



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VIII
สารบัญรูป.....	IX
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	2
1.3 สมมติฐานของการศึกษา.....	2
1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย.....	2
1.5 ขอบเขตการวิจัย.....	3
1.6 ขั้นตอนของการศึกษา.....	3
บทที่ 2 เทคนิคที่ใช้อธิบายภาพของวัตถุ.....	5
2.1 บทนำ.....	5
2.2 การจำแนกประเภทของเทคนิคที่ใช้ในการอธิบายภาพของวัตถุ.....	6
2.3 เทคนิคทางเส้นรอบรูป.....	7
2.3.1 วิธีแบบโครงสร้างโดยรวม.....	7
2.3.1.1 ตัวอธิบายรูปร่างแบบง่าย.....	8
2.3.1.2 ตัวอธิบายรูปร่างแบบหลายสัญลักษณ์.....	8
2.3.1.3 ตัวอธิบายรูปร่างแบบ โมเมนต์เส้นรอบรูป.....	9
2.3.1.4 ตัวอธิบายรูปร่างแบบวิธีเคอร์ฟเวเจอร์สเกลสเปซ.....	10
2.3.1.5 ตัวอธิบายรูปร่างที่ได้มาจากการแปลงแบบสเปกตรอล.....	11
2.3.2 วิธีแบบโครงสร้างย่อย.....	12
2.3.2.1 ตัวอธิบายรูปร่างแบบรหัสห่วงโซ่.....	12
2.3.2.2 ตัวอธิบายรูปร่างตามลักษณะเหลี่ยมมุม.....	13
2.3.2.3 ตัวอธิบายรูปร่างที่อาศัยหลักการวิเคราะห์ทางไวยากรณ์ภาษา.....	13
2.4 เทคนิคทางพื้นที่.....	14

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้ในการเรียนการสอนเท่านั้น ไม่ให้นำไปใช้ประโยชน์ด้านธุรกิจ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังห้ามเผยแพร่ข้อมูลนี้ไปบนอินเทอร์เน็ตในลักษณะใดๆทั้งสิ้น

สารบัญ(ต่อ)

หน้า

2.4.1	วิธีแบบโครงสร้างโดยรวม.....	14
2.4.1.1	ตัวอย่างรูปร่างแบบโมเมนต์ที่มีคุณสมบัติไม่เปลี่ยนแปลงต่อการแปลงเชิงเรขาคณิต.....	14
2.4.1.2	ตัวอย่างรูปร่างแบบโมเมนต์ที่ไม่ผันแปรในเชิงพีชคณิต.....	15
2.4.1.3	ตัวอย่างรูปร่างแบบโอโรโทโกนอลโมเมนต์.....	15
2.4.1.4	ตัวอย่างรูปร่างในเชิงฟูเรียร์.....	16
2.4.1.5	ตัวอย่างรูปร่างที่อาศัยการสร้างกริด.....	16
2.4.1.6	ตัวอย่างรูปร่างในเชิงเมทริกซ์.....	17
2.4.2	วิธีแบบโครงสร้างย่อย.....	18
2.4.2.1	ตัวอย่างรูปร่างแบบคอนเวกซ์ฮาว.....	18
2.4.2.1	ตัวอย่างรูปร่างที่อาศัยหลักการของการหาแกนกลาง.....	18
บทที่ 3	ตัวแทนรูปร่างซึ่งไม่ผันแปรต่อการแปลงแบบแอฟไฟน์.....	20
3.1	บทนำ.....	20
3.2	การแปลงแบบแอฟไฟน์.....	20
3.3	ทฤษฎีดีเทอร์มิแนนต์.....	22
3.3.1	การหาดีเทอร์มิแนนต์ของเมทริกซ์ที่มีมิติ 2×2	23
3.3.2	การหาดีเทอร์มิแนนต์ของเมทริกซ์ที่มีมิติ 3×3	25
3.3.3	การหาดีเทอร์มิแนนต์ของเมทริกซ์ที่มีมิติ $n \times n$	25
3.3.4	คุณสมบัติของดีเทอร์มิแนนต์.....	26
3.4	การประยุกต์ใช้ทฤษฎีดีเทอร์มิแนนต์ในการหาลักษณะเฉพาะของรูปร่างที่ไม่ผันแปรต่อการแปลงแบบแอฟไฟน์.....	27
3.4.1	ขั้นตอนการหา 1-D triangle area representation.....	29
3.4.2	ชนิดของ 1-D triangle area representation ที่ใช้ในงานวิจัย.....	30
3.4.3	ตัวอย่างลักษณะของ 1-D triangle area representation ที่ไม่ผันแปรต่อการแปลงแบบแอฟไฟน์.....	31
บทที่ 4	วิธีการรู้จำภาพของวัตถุ.....	34
4.1	บทนำ.....	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ

สารบัญ(ต่อ)

หน้า

4.1.1 การรู้จำรูปแบบด้วยหลักการทางสถิติ.....	34
4.1.2 โครงข่ายประสาทเทียม.....	35
4.1.3 การรู้จำทางไวยากรณ์ภาษา.....	35
4.1.4 การรู้จำโดยการเปรียบเทียบกราฟ.....	36
4.1.5 ระบบพีซี.....	36
4.2 นอร์มอลไลซ์ครอสคอร์รีเลชัน.....	38
4.3 โครงข่ายประสาทเทียม.....	38
4.3.1 โครงข่ายประสาทชีวภาพ.....	38
4.3.2 โครงข่ายประสาทเทียม.....	40
4.3.2.1 โครงข่ายประสาทเทียมเซลล์เดียว.....	42
4.3.2.2 โครงข่ายประสาทเทียมแบบหลายอินพุท.....	43
4.3.2.3 โครงสร้างของโครงข่ายประสาทเทียม.....	43
4.3.2.3.1 โครงข่ายประสาทเทียมแบบชั้นเดียว.....	43
4.3.2.3.2 โครงข่ายประสาทเทียมแบบหลายชั้น.....	46
4.3.2.4 ฟังก์ชันถ่ายโอน.....	47
4.3.2.4.1 ฟังก์ชันถ่ายโอนแบบ Hard-limit.....	47
4.3.2.4.2 ฟังก์ชันถ่ายโอนแบบ Hard-limit สมมาตร.....	48
4.3.2.4.3 ฟังก์ชันถ่ายโอนแบบลอจิสึมอยด์.....	48
4.3.2.4.4 ฟังก์ชันถ่ายโอนแบบเชิงเส้นแบบบวก.....	49
4.3.2.4.5 ฟังก์ชันถ่ายโอนแบบเชิงเส้น.....	49
4.3.2.4.6 ฟังก์ชันถ่ายโอนแบบแทนซิกมอยด์.....	49
4.3.2.5 การฝึกสอนและการเรียนรู้ของโครงข่ายประสาทเทียม.....	50
4.3.2.5.1 การฝึกสอนแบบอิสระ.....	51
4.3.2.5.2 การฝึกสอนแบบควบคุม.....	51
บทที่ 5 การทดลองและผลการทดลอง.....	56
5.1 การเตรียมข้อมูล.....	56
5.2 การทดลองและผลการทดลองของประสิทธิภาพในการรู้จำ.....	58
5.2.1 ประสิทธิภาพของการรู้จำด้านความคงทนต่อการแปลงแบบแอฟฟายน์.....	59

สารบัญ(ต่อ)

	หน้า
5.2.1.1 การรู้จำด้วยวิธี Normalized cross-correlation.....	59
5.2.1.1.1 ความแม่นยำในการจับคู่ (matching).....	60
5.2.1.1.2 ประสิทธิภาพในการกู้คืนรูปร่าง.....	61
5.2.1.2 การรู้จำด้วยวิธี Artificial Neural Networks.....	65
5.2.1.3 การเปรียบเทียบประสิทธิภาพในการรู้จำระหว่างวิธี Normalized cross-correlation และวิธี Artificial Neural Networks.....	68
5.2.1.4 การวิเคราะห์ประสิทธิภาพในการแยกแยะข้อมูลของ 1-D triangle area representation ด้วย Fisher's discriminant.....	72
5.2.2 ประสิทธิภาพการรู้จำในด้านความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้น.....	76
5.2.3 ประสิทธิภาพการรู้จำในด้านความคงทนต่อสัญญาณรบกวน.....	80
5.2.4 การเปรียบเทียบกับงานวิจัยอื่นๆ.....	82
5.2.5 การทดสอบกับภาพถ่ายจริงที่ได้จากกล้องถ่ายภาพและตัวอย่างการประยุกต์ใช้งาน.....	83
บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ.....	88
เอกสารอ้างอิง.....	91
ภาคผนวก ก ผลงานวิจัยที่ได้รับการตีพิมพ์.....	95
ประวัติผู้เขียน.....	124

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
3.1 แสดงตัวอย่างภาพและเมทริกซ์ของการแปลงแบบแอฟไฟน์แบบต่างๆ.....	22
5.1 ตัวแปรทางการแปลงแบบแอฟไฟน์ของรูปร่างทั้ง 14 แบบ.....	57
5.2 ความแม่นยำในการ match ด้วยการ ใช้ ATA และ NATA ที่แขนของสามเหลี่ยม / ต่างๆ.....	61
5.3 ตัวอย่างผลของการรู้จากรูปร่าง.....	63
5.4 ความถูกต้องในการรู้จำด้วยวิธี NN ที่ถูกฝึกสอนด้วยขนาดและประเภทของข้อมูลต่างๆ.....	66
5.5 การเปรียบเทียบประสิทธิภาพในการรู้จำระหว่างวิธี NCC และวิธี NN ที่ใช้ชุดข้อมูล ขนาดเล็กและขนาดใหญ่ในการ match สำหรับวิธี NCC และการฝึกสอน NN โดยทดสอบกับ NATA ที่ความยาวแขนของสามเหลี่ยมต่างๆ.....	69
5.6 แสดงประสิทธิภาพของการรู้จำในด้านความทนต่อการเปลี่ยนแปลงจุดเริ่มต้นของวิธี NCC และวิธี NN ซึ่งทดสอบกับ NATA ที่หามาจากจุดเริ่มต้นที่แตกต่างกัน.....	79
5.7 ผลการทดลองของการเพิ่มประสิทธิภาพด้านความทนต่อการเปลี่ยนแปลงจุดเริ่มต้นของวิธี NN ด้วยการพิจารณาผลลัพธ์ของความยาวแขนของสามเหลี่ยมบางแขนว่าตอบclass ไหนมากที่สุด.....	80
5.8 แสดงการวัดประสิทธิภาพในการรู้จำในด้านความสามารถในการทนต่อสัญญาณรบกวน ของวิธี Normalized cross-correlation และ วิธี Artificial Neural Networks ที่วัดกับ NATA ที่ความยาวแขนของสามเหลี่ยมค่าต่างๆ.....	81
5.9 การเปรียบเทียบคะแนน Bullseye test ของ NATA ที่ใช้ในงานวิจัยนี้กับ Shape representation ที่ใช้ในงานวิจัยอื่น.....	83

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
1.1 ตัวอย่างของการนำเอาการรู้จำวัตถุไปประยุกต์ใช้งาน.....	1
1.2 การมองเห็นวัตถุในมุมมองต่างๆ.....	2
2.1 ลำดับขั้นในการแยกประเภทของเทคนิคที่ใช้ในการอธิบายภาพของวัตถุ.....	7
2.2 รูปร่างที่เกิดความผิดพลาดในการวัดความเบี้ยว (eccentricity) และความเป็นวงกลม (circularity).....	8
2.3 รูปร่างแอฟเปิดและ <i>Shape signature</i> ของรูปร่างที่หามาจากระยะทางจากจุดศูนย์กลาง.....	9
2.4 Curvature Scale Space (CSS).....	10
2.5 ตัวอย่างการนำเอาเส้นรอบรูปมาหา Shape representation ในทางเวฟเลท โดเมน.....	12
2.6 หลักการสร้าง chain code โดยเริ่มจากกำหนดจุดเริ่มต้นบนเส้นขอบภาพและทิศทางที่เป็นไปได้ในการพิจารณาหาจุดบนเส้นรอบรูปถัดไป.....	13
2.7 ตัวอย่างการหาเส้นรอบรูปที่หามาจากวิธี structural ในแบบ syntactic analysis.....	14
2.8 รูป Polar-raster.....	16
2.9 การสร้าง grid ไปบนรูปร่าง.....	17
2.10 การสร้างเส้นวงกลมและเส้นรัศมีวงกลมไปบนรูปร่าง.....	18
2.11 Convex hull.....	18
2.12 โครงสร้างของแกนเฉลี่ยของรูปร่างสี่เหลี่ยม.....	19
3.1 ส่วนประกอบของเมทริกซ์การแปลงแบบแอฟไฟน์ในระบบพิกัด โฮโมจีเนียส.....	21
3.2 รูปสี่เหลี่ยมด้านขนาน.....	24
3.3 รูปสี่เหลี่ยมด้านขนานที่จุดยอดมุมไม่ได้อยู่ที่จุด origin.....	24
3.4 ปริซึมหกด้านที่มีแต่ละด้านเป็นรูปสี่เหลี่ยมด้านขนาน.....	25
3.5 การแปลงสามเหลี่ยม S ให้เป็นสามเหลี่ยม $T(S)$ ด้วยสมการเชิงเส้น.....	27
3.6 การพิจารณาสามเหลี่ยมบนเส้นรอบรูปของรูปร่างค้นฉบับกับรูปร่างที่ถูกแปลงแบบแอฟไฟน์.....	28
3.7 ตัวอย่าง 1-D triangle area representation ของรูปร่างค้นฉบับ.....	29
3.8 จุดแหลมปลิง 3 จุดใดๆที่ TSL ต่างๆบนเส้นรอบรูปที่นำไปคำนวณพื้นที่สามเหลี่ยม.....	29
3.9 ตัวอย่าง 1-D triangle area representation คำนวณที่ $I=20$	30
3.10 แสดงถึงส่วนเว้าส่วนนูนของเส้นรอบรูปที่สัมพันธ์กับค่าดิเทอร์มิแนนต์หรือค่าพื้นที่.....	30
3.11 แสดงตัวอย่างของ ATA และ NATA ที่หามาจากแกนสามเหลี่ยม I ใดๆ.....	31

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.12 ตัวอย่างลักษณะของ 1-D triangle area representation ที่ไม่ผันแปรต่อการแปลงแบบแอฟฟายน์.....	31
3.13 รูปร่างที่ถูกแปลงแบบ Reflection หรือ Mirror.....	33
3.14 การสลับตำแหน่งคู่อันดับ (x,y)	33
4.1 General discrimination functions.....	35
4.2 โครงสร้างโคอะแกรมที่ใช้แทนรูปร่าง.....	36
4.3 ตรีกรแบบจริงเท็จ (บูลีนลอจิก) กับตรีกรแบบพีชชี (พีชชีลอจิก).....	37
4.4 ความไม่แน่นอน (uncertainty).....	37
4.5 กราฟ 2 กราฟ ที่ต้องการวัดความคล้ายคลึงกัน.....	38
4.6 นิเวรอนของสมองมนุษย์.....	39
4.7 นิเวรอนทางอิเล็กทรอนิกส์.....	40
4.8 โครงข่ายประสาทชีวภาพและโครงข่ายประสาทเทียม.....	41
4.9 โครงข่ายประสาทเทียมหนึ่งหน่วยแบบง่าย.....	42
4.10 โครงข่ายประสาทเทียมหนึ่งหน่วยแบบหลายอินพุท.....	43
4.11 โครงข่ายประสาทเทียมแบบชั้นเดียว.....	44
4.12 โครงข่ายประสาทเทียมแบบชั้นเดียวในรูปแบบย่อ.....	45
4.13 โครงข่ายประสาทเทียมแบบชั้นเดียวที่ใช้ใน MATLAB Toolbox.....	46
4.14 โครงข่ายประสาทเทียมสามชั้น.....	46
4.15 โครงข่ายประสาทเทียมสามชั้นในรูปแบบย่อ.....	47
4.16 ฟังก์ชันถ่ายโอนแบบ Hard-limit.....	48
4.17 ฟังก์ชันถ่ายโอนแบบ Hard-limit สมมาตร.....	48
4.18 ฟังก์ชันถ่ายโอนแบบลอคซิกมอยด์.....	48
4.19 ฟังก์ชันถ่ายโอนแบบเชิงเส้นแบบบวก.....	49
4.20 ฟังก์ชันถ่ายโอนแบบเชิงเส้น.....	49
4.21 ฟังก์ชันถ่ายโอนแบบแทนซิกมอยด์.....	50
4.22 โครงข่ายประสาทเทียมที่ใช้ในการเรียนรู้แบบแพร่ย้อนกลับ.....	52
5.1 รูปร่างที่ใช้ในงานวิจัยนี้.....	56
5.2 ตัวอย่างรูปร่างที่ถูกแปลงแบบแอฟฟายน์ไป 14 แบบ.....	57
5.3 รูปร่างที่ถูกแปลงแบบ Shear ไปมากเกินไปจนเสียลักษณะเค้าโครงของรูปร่างเดิม.....	57

สารบัญรูป(ต่อ)

รูปที่	หน้า
5.4 เส้นรอบรูปที่หาได้จากวิธี Contour follower.....	58
5.5 ตัวอย่างจุดแชมป์ถึง 100 จุดบนเส้นรอบรูป.....	58
5.6 ขั้นตอนการทดลองของการ match และ retrieval.....	60
5.7 ตัวอย่าง Precision-Recall curve.....	63
5.8 Precision Recall curve ที่ใช้วัดประสิทธิภาพในการรู้จำรูปร่าง โดยใช้ ATA และ NATA ที่ แกนของสามเหลี่ยมเท่ากับ 1, 20, และ 49.....	64
5.9 โครงสร้างของ Neural network ที่ใช้ในการรู้จำ.....	66
5.10 ประสิทธิภาพในการรู้จำรูปร่างด้วยวิธี Artificial Neural Networks ที่ถูกฝึกสอนด้วยขนาด ข้อมูลที่แตกต่างกัน ก) ฝึกสอนด้วยชุดข้อมูลขนาดเล็ก และข) ฝึกสอนด้วยชุดข้อมูลขนาด ใหญ่ นอกจากนั้นยังเป็นการเปรียบเทียบประสิทธิภาพในการใช้ประเภทของ Shape representation ที่แตกต่าง ATA กับ NATA ที่แกนของสามเหลี่ยม 1-49.....	67
5.11 การเปรียบเทียบประสิทธิภาพในการรู้จำระหว่างวิธี NCC และวิธี NN ที่ใช้ชุดข้อมูล ก) ชุด ข้อมูลขนาดเล็ก และข) ชุดข้อมูลขนาดใหญ่ โดยทดสอบกับ NATA ที่แกน ของสามเหลี่ยม 1-49.....	71
5.12 แสดงตัวอย่างการพิจารณาหา 1-D triangle area representation ที่แกนของสามเหลี่ยมค่าๆ ก) ที่แกนของสามเหลี่ยมเท่ากับ 1 และข) แสดง ATA และ NATA ที่คำนวณมาจากแกนของ สามเหลี่ยมเท่ากับ 1.....	73
5.13 แสดงตัวอย่างการพิจารณาหา 1-D triangle area representation ที่แกนของสามเหลี่ยมสูงๆ ก) ที่แกนของสามเหลี่ยมเท่ากับ 49 และข) แสดง ATA และ NATA ที่คำนวณมาจากแกนของ สามเหลี่ยมเท่ากับ 49.....	73
5.14 Fisher's discriminant.....	75
5.15 แสดงค่า Fisher's ratio ของ ก) ATA และข) NATA ที่แกนของสามเหลี่ยม เท่ากับ 1, 20, และ 49.....	75
5.16 เปรียบเทียบประสิทธิภาพในการแยกแยะข้อมูลด้วยค่าเฉลี่ย Fisher's ratio ของ ATA และ NATA ที่ความยาวแกนของสามเหลี่ยม 1-49.....	76
5.17 หลักการในการพัฒนาวิธีการให้มีความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้น โดยนำ 1-D triangle area representation ของรูปร่างที่นำมาทดสอบมาเลื่อนตำแหน่งไป n ครั้งตามจำนวน ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้ sampled point- n	77
5.18 การเพิ่มสัญญาณรบกวนไปยังเส้นรอบรูป.....	81

สารบัญรูป(ต่อ)

รูปที่	หน้า
5.19 รูปร่างที่ถูกแปลงแบบ Affine และ Perspective.....	84
5.20 ภาพของวัตถุแบนที่ถูกถ่ายจากมุมมองต่างๆ.....	85
5.21 ประสิทธิภาพในการรู้จำของวิธี Normalized cross-correlation และ วิธี Artificial Neural Networks ที่ทดสอบกับรูปร่างที่ถ่ายได้จากกล้องถ่ายรูป ด้วยการใช้ NATA ที่ค่าความยาวแกนของสามเหลี่ยม 1-49.....	86
5.2.2 การหารูปร่างที่มีความเหมือนกันระหว่างภาพ 2 ภาพที่ถูกถ่ายคนละมุมมองกล้องกัน.....	87



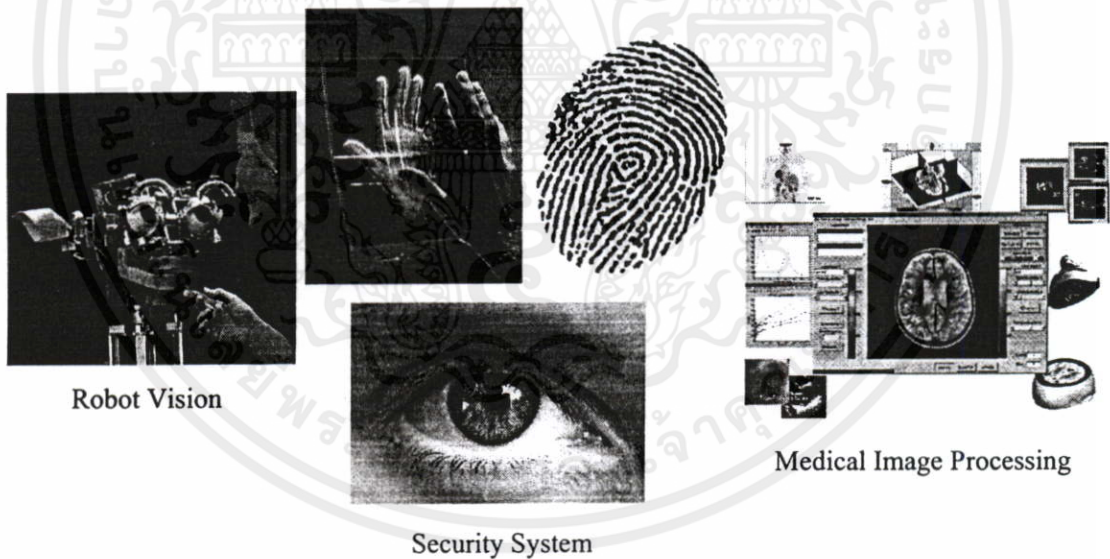
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การรู้จำวัตถุโดยใช้การประมวลผลภาพ (Image Processing) เป็นหัวข้อที่ได้รับความสนใจในการศึกษาและวิจัยมาเป็นเวลานาน ทั้งนี้เนื่องจากสามารถนำไปประยุกต์ใช้งานได้อย่างกว้างขวาง ไม่ว่าจะเป็นงานทางด้าน ระบบรักษาความปลอดภัย (Security System) การวิเคราะห์ภาพทางการแพทย์ (Medical Image Processing) และระบบการมองเห็นของหุ่นยนต์ (Robot Vision) เป็นต้น ดังแสดงในรูปที่ 1.1 ซึ่งงานด้านการประมวลผลภาพ ระบบการรู้จำวัตถุถือเป็นส่วนสำคัญที่ใช้ในการตัดสินใจและประมวลผลของหุ่นยนต์หรือคอมพิวเตอร์ ด้วยเหตุนี้การพัฒนาวิธีการในรู้จำภาพของวัตถุในมุมมองต่างๆให้มีประสิทธิภาพจึงเท่ากับเป็นการเพิ่มความฉลาดหรือความสามารถให้กับหุ่นยนต์ หรือคอมพิวเตอร์ หรืองานทางด้าน การประมวลผลภาพใดๆที่อาศัยหลักการรู้จำภาพวัตถุเป็นพื้นฐาน



รูปที่ 1.1 ตัวอย่างของการนำเอาการรู้จำวัตถุไปประยุกต์ใช้งาน

โดยปกติแล้วภาพของวัตถุที่หุ่นยนต์มองเห็นนั้นเปรียบเสมือนภาพที่ถ่ายได้จากมุมมองต่างๆ ซึ่งรูปร่างของวัตถุในภาพมักจะผิดรูปไปทั้ง ขนาด ตำแหน่ง รูปทรงที่หมุนและลักษณะที่ถูกเฉือน ดังแสดงในรูปที่ 1.2 ปัญหาก็คือเราจะทำอย่างไรให้หุ่นยนต์หรือคอมพิวเตอร์สามารถรู้จำวัตถุ ไม่ว่าจะมองวัตถุจากมุมมองใดๆก็ตาม



มุมมองต่างๆในการมองวัตถุ



ตัวอย่างของรูปร่างของวัตถุในภาพที่ผิดรูปไป

รูปที่ 1.2 การมองเห็นวัตถุในมุมมองต่างๆ

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

ดังที่ได้กล่าวไปแล้วข้างต้น หลักการรู้จำรูปร่างมีความสำคัญต่อการทำงานของหุ่นยนต์หรือคอมพิวเตอร์ ด้วยเหตุผลนี้ เราจึงทำการวิจัยและพัฒนาวิธีการในการรู้จำรูปร่างของวัตถุ โดยได้ทำการศึกษาถึงทฤษฎีที่จะนำมาใช้ในการหาคุณลักษณะที่จะใช้ในระบบการรู้จำให้ได้ประสิทธิภาพที่สามารถรู้จำรูปร่างได้ไม่ว่าวัตถุนั้นจะผิดรูปไปทั้ง ขนาด ตำแหน่ง รูปทรงที่หมุนและการเอียง

1.3 สมมติฐานของการศึกษา

จากลักษณะของรูปร่างของวัตถุที่เปลี่ยนแปลงไปตามมุมมองต่าง ๆ นั้น ทำให้เกิดปัญหาขึ้นว่าเราจะทำอย่างไรให้ระบบการรู้จำของเราสามารถรู้จำรูปร่างของวัตถุที่ผิดรูปเหล่านั้นได้ เราจึงตั้งสมมติฐานว่า ถ้าหากเราสามารถดึงเอาคุณลักษณะเฉพาะ (Feature Extraction) ของแต่ละรูปร่างที่มีคุณสมบัติเสถียร (ทนต่อผลกระทบที่เกิดจาก noise ไม่ว่าจะรูปร่างจะผิดรูปไปอย่างไร ก็ยังมีลักษณะเหมือนเดิมไม่เปลี่ยนแปลง) ออกมาเป็นตัวแทนของแต่ละรูปร่าง (Shape representation) ได้แล้วนั้น น่าจะสามารถช่วยลดความซับซ้อนในกระบวนการรู้จำ และระบบน่าจะมีประสิทธิภาพที่ดีหรือมีอัตราความผิดพลาดน้อย

1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย

จากสมมติฐานที่เราได้ตั้งไว้ งานวิจัยนี้จึงได้พยายามศึกษาค้นคว้าหาทฤษฎีที่จะนำมาใช้ในการหา Shape representation ที่ใช้เป็นเอกลักษณ์ของแต่ละรูปร่าง โดยเราพบว่าทฤษฎีพีชคณิตเชิงเส้น (Linear Algebra) สามารถนำมาประยุกต์ใช้เพื่อหาคุณลักษณะเฉพาะของรูปร่างที่ไม่เปลี่ยนแปลงต่อการเปลี่ยนขนาด (Scaling), การหมุน (Rotation), การเปลี่ยนตำแหน่ง (Translation),

และ การบิดเบือน (Shear) เราเรียกการเปลี่ยนแปลงรูปทรงเหล่านี้ว่า “การแปลงแบบแอฟไฟน์ (Affine transformation)” โดยเราได้นำทฤษฎีดีเทอร์มิแนนต์มาใช้กับเส้นรอบรูปของรูปร่างที่เราสนใจเพื่อให้ได้มาซึ่งคุณลักษณะเฉพาะของรูปร่างที่ไม่ผันแปรต่อการแปลงแบบแอฟไฟน์ (Affine Invariant Shape Representation) ซึ่งคุณลักษณะเฉพาะของ Shape Representation นี้คือ “ตัวแทนพื้นที่สามเหลี่ยมใน 1 มิติ (1-D triangle area representation)” ซึ่งเราได้ทำการแบ่งออกเป็นหลายระดับตามขนาดความยาวแขนของสามเหลี่ยม เพื่อสังเกตว่า 1-D triangle area representation ที่ค่าความยาวแขนของสามเหลี่ยมเท่าใด จะให้ประสิทธิภาพในการรู้จำที่ดีที่สุด ส่วนวิธีการในการรู้จำรูปร่างของระบบ ในตอนแรกเราได้ศึกษาวิธีการแบบนอร์มอไลซ์ครอสคอร์รีเลชัน (Normalized cross-correlation) โดยใช้วัดความเหมือนระหว่าง Shape representation ของรูปร่างที่นำมาทดสอบกับทุกๆรูปร่างในฐานข้อมูล เมื่อทำการทดลองเราก็พบว่า วิธีการรู้จำโดยใช้วิธีนอร์มอไลซ์ครอสคอร์รีเลชันให้ผลการทดลองที่ดี แต่ใช้เวลาในการคำนวณนานเพราะต้องทำการเปรียบเทียบกับทุกๆรูปร่างในฐานข้อมูล เราจึงศึกษาต่อว่าจะมีวิธีการใดหรือไม่ ที่มีคุณสมบัติในการรู้จำได้เร็วโดยไม่ทำให้ความแม่นยำลดลง เราพบว่าวิธีการรู้จำโดยใช้โครงข่ายประสาทเทียม (Artificial Neural Network) มีคุณสมบัติดังกล่าว นอกจากนั้นเรายังได้นำเสนอวิธีการหาขนาดความยาวแขนของสามเหลี่ยมที่ทำให้ 1-D triangle area representation มีประสิทธิภาพดีที่สุด โดยไม่ต้องทำการทดสอบการรู้จำ ซึ่งเราจะใช้การวิเคราะห์ความสามารถในการแยกแยะข้อมูลด้วยทฤษฎีฟิชเชอร์แทน (Fisher discriminant analysis) ซึ่งสามารถลดขั้นตอนและเวลาในการหาค่าความยาวแขนของสามเหลี่ยมที่ดีที่สุดลงได้

1.5 ขอบเขตการวิจัย

วิทยานิพนธ์ฉบับนี้นำเสนอการรู้จำรูปร่าง 2 มิติ ด้วยวิธี Normalized cross-correlation และวิธี Artificial Neural Network โดยใช้ 1-D triangle area representation เป็นตัวแทนแต่ละรูปร่างเพื่อใช้ในขั้นตอนการรู้จำ ซึ่งรายละเอียดของวิทยานิพนธ์มีดังนี้ บทที่ 2 กล่าวถึงเทคนิคที่ใช้อธิบายภาพของวัตถุ ส่วนบทที่ 3 กล่าวถึงตัวแทนรูปร่างที่ไม่ผันแปรต่อการแปลงแบบแอฟไฟน์ จากนั้นในบทที่ 4 จะกล่าวถึงวิธีการรู้จำภาพของวัตถุ ส่วนบทที่ 5 กล่าวถึงการทดลองและผลการทดลองและสุดท้ายบทที่ 6 เป็นสรุปผลการวิจัยและข้อเสนอแนะต่างๆ

1.6 ขั้นตอนของการศึกษา

เอกสารนี้เป็นเอกสารต้นฉบับของวิทยานิพนธ์ที่จัดทำขึ้นเพื่อใช้ในการศึกษาวิจัยเท่านั้น ไม่สามารถนำเอกสารนี้ไปเผยแพร่หรือใช้เพื่อวัตถุประสงค์อื่นใดได้โดยไม่ได้รับอนุญาตจากข้าพเจ้า

ขั้นตอนของการรู้จำรูปร่าง เริ่มด้วยการนำรูปร่างไปหา ตัวแทนพื้นที่สามเหลี่ยมใน 1 มิติ (1-D triangle area representation) ที่หลายๆขนาดความยาวแขนของสามเหลี่ยม (Multi-triangle side length) จากนั้นก็นำตัวแทนรูปร่าง (Shape representation) นี้ไปใช้ในขั้นตอนการรู้จำทั้งวิธี

Normalized cross-correlation และวิธี Artificial Neural Network ในการทดลองเราได้วัดประสิทธิภาพในการรู้จำรูปร่างที่ไม่ผันแปรต่อการแปลงแบบแอฟโฟมอีกทั้งยังได้พัฒนาวิธีการรู้จำของทั้งสองวิธีให้มีความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้น นอกจากนี้เราได้ทดลองใช้ทฤษฎี Fisher discriminant ทำการทำนายหา 1-D triangle area representation ที่ดีที่สุดแล้วเปรียบเทียบกับผลการหา 1-D triangle area representation จากขบวนการรู้จำโดยวิธี Normalized cross-correlation และวิธี Artificial Neural Network สุดท้ายเรายังทำการทดลองเพื่อทดสอบประสิทธิภาพในการทนต่อสัญญาณรบกวนและทำการเปรียบเทียบกับงานวิจัยอื่นๆ อีกทั้งยังนำวิธีการของเราไปทดสอบกับภาพถ่ายจริงเพื่อแสดงถึงการประยุกต์ใช้งานจริง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

เทคนิคที่ใช้อธิบายภาพของวัตถุ

(Shape representation and description techniques)

2.1 บทนำ

ในยุคแรกๆของการประยุกต์ใช้การประมวลผลภาพดิจิทัลเกิดขึ้นในปี 1960 ที่ห้องทดลองทางด้านการปล่อยเครื่องยนต์เจ็ทสู่อวกาศ (Jet Propulsion Laboratory-JPL) ขององค์การนาซา, สถาบันเทคโนโลยีแห่งรัฐแมสซาชูเซตส์ (Massachusetts Institute of Technology-MIT), ห้องแลปเบลล์ (Bell Labs) และ มหาวิทยาลัยแมริแลนด์ โดยได้นำการประมวลผลภาพดิจิทัลมาช่วยในการวิจัยอย่างจริงจังในด้านภาพถ่ายดาวเทียมและอวกาศ, ภาพทางการแพทย์, โทรศัพท์ที่สามารถเห็นคู่สนทนาได้ (videophone), การรู้จำตัวอักษร, และการแก้ไขปรับปรุงภาพ ด้วยข้อจำกัดทางเทคโนโลยีด้านคอมพิวเตอร์และอุปกรณ์ที่ใช้ในการประมวลผลซึ่งมีราคาค่อนข้างสูงในสมัยยุคแรกๆนั้น ทำให้เทคนิคที่ใช้ในการวิเคราะห์และประยุกต์ใช้ภาพดิจิทัลถูกจำกัดแค่หลักการง่ายๆไม่ซับซ้อน โดยส่วนใหญ่จะเป็นการนำเอาค่าสีหรือข้อมูลในแต่ละพิกเซลในภาพมาใช้โดยตรงไม่ผ่านการดึงเอาคุณลักษณะเฉพาะออกมา (Feature extraction) ใช้เป็นตัวแทนรูปร่างในภาพ (Shape representation) เหมือนในปัจจุบัน ในยุคนั้นวิวัฒนาการทางด้านการประมวลผลภาพดิจิทัลจึงเป็นไปอย่างช้าๆ ต่อมาในช่วงปี 1970 เทคโนโลยีทางคอมพิวเตอร์และเครื่องมืออุปกรณ์ทางระบบภาพดิจิทัลได้ถูกพัฒนาให้มีประสิทธิภาพมากขึ้นอีกทั้งราคาถูกลง ทำให้สามารถเข้าถึงเทคโนโลยีดังกล่าวได้ง่ายขึ้น การประมวลผลภาพดิจิทัลจึงได้รับความสนใจอย่างแพร่หลายจากนักวิจัยทั่วโลก ตั้งแต่นั้นมาทฤษฎีหรือเทคนิคทางการประมวลผลภาพดิจิทัลจำนวนมากก็ถูกพัฒนาขึ้นอย่างรวดเร็วจากนักวิจัยแต่ละด้าน โดยเฉพาะอย่างยิ่งเทคนิคทางการอธิบายภาพ (Shape representation technique) ที่เป็นส่วนสำคัญต่อการนำไปประยุกต์ใช้งานทางด้าน การมองเห็นของหุ่นยนต์ (Robot Vision) ระบบรักษาความปลอดภัย, การวิเคราะห์ภาพทางการแพทย์ เป็นต้น

ในปัจจุบันภาพดิจิทัลถูกนำไปใช้ในงานมากมายไม่ว่าจะเป็นงานด้านการรู้จำวัตถุ การค้นหาข้อมูลโดยใช้ภาพในระบบฐานข้อมูลและในระบบอินเทอร์เน็ต การรักษาความปลอดภัย และอีกมากมาย เพื่อให้งานเหล่านั้นมีประสิทธิภาพ เราต้องหาตัวอธิบายภาพของวัตถุที่มีลักษณะเฉพาะมาเป็นตัวแทนของแต่ละวัตถุ รูปร่าง (Shape) เป็นลักษณะที่สามารถมองเห็นซึ่งเป็นลักษณะพื้นฐานและสำคัญต่อการอธิบายภาพ (Shape representation) อย่างไรก็ตามการที่เราจะหาตัวอธิบายภาพของวัตถุที่ดีได้นั้นถือว่าเป็นสิ่งที่ยากทั้งนี้เนื่องจาก วัตถุนั้นอยู่ในรูป 3 มิติ แต่ภาพถ่ายของวัตถุนั้นอยู่ในรูป 2 มิติ หรือ ถูกโปรเจกต์ (projected) ให้อยู่ในระนาบ 2 มิติ ข้อมูลของวัตถุอีก

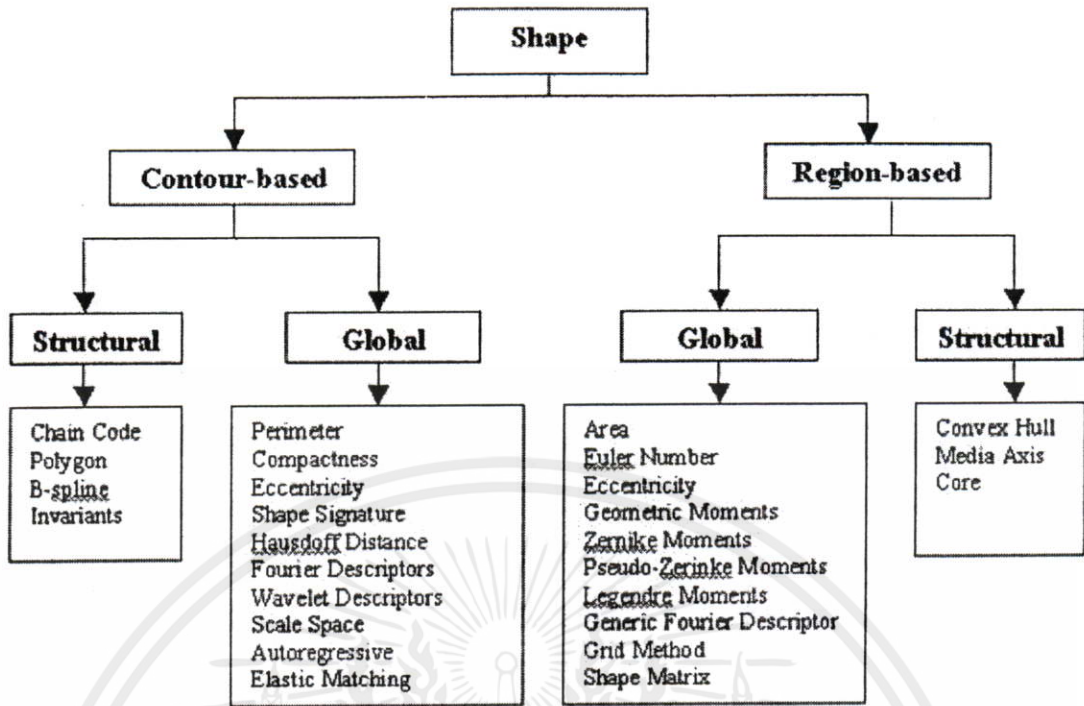
หนึ่งมิตินั้นหายไป ดังนั้นรูปร่างของวัตถุในภาพเป็นแค่ส่วนหนึ่งของวัตถุที่ถูกโปรเจกต์ (projected) ไม่เพียงแต่ปัญหาการสูญเสียข้อมูลของวัตถุไป 1 มิติ ที่ทำให้เกิดปัญหาในการหาตัวอธิบายวัตถุที่ดีแล้วนั้น ยังมีปัญหาอีกมากมายที่ทำให้ข้อมูลที่แท้จริงของภาพวัตถุผิดเพี้ยนไปไม่ว่าจะเป็นสัญญาณรบกวน (noise) รูปร่างของวัตถุที่ผิดรูปไปและการบดบังวัตถุทำให้ภาพบางส่วนของวัตถุหายไป

ตั้งแต่อดีตจนถึงปัจจุบันเทคนิคที่ใช้อธิบายภาพของวัตถุ (Shape representation and description techniques) ได้ถูกพัฒนาขึ้นมามากมาย ในบทนี้ได้ทำการรวบรวมเทคนิคดังกล่าวจากหลากหลายงานวิจัยและบทความ แล้วได้ทำการจำแนกประเภทของเทคนิคที่ใช้ในการอธิบายภาพของวัตถุ พบว่าสามารถแบ่งออกเป็น 2 เทคนิคใหญ่ๆคือ เทคนิคทางเส้นรอบรูป (Contour-based) และเทคนิคทางพื้นที่ (Region-based) โดยทั่วไปการหา Shape representation นั้นจะพยายามดึงเอาลักษณะเฉพาะที่สำคัญออกมาจากรูปร่าง (Shape) โดยอาจจะหามาจากเส้นรอบรูปหรือบริเวณพื้นที่ทั้งหมดของรูปร่างในภาพนั้นบรรดานักวิจัยหลายคนต่างได้สร้าง Shape representation ออกมาหลายแบบแตกต่างกันไป [2-32] อาทิเช่น *shape signature, signature histogram, shape invariants, moments, curvature, shape context, shape matrix, spectral features* เป็นต้น ซึ่งจะกล่าวโดยละเอียดต่อไป ซึ่งจะกล่าวถึง การจำแนกประเภทของเทคนิคที่ใช้ในการอธิบายภาพของวัตถุในหัวข้อที่ 2.2 เทคนิคทางเส้นรอบรูปในหัวข้อที่ 2.3 และเทคนิคทางพื้นที่ในหัวข้อที่ 2.4

2.2 การจำแนกประเภทของเทคนิคที่ใช้ในการอธิบายภาพของวัตถุ

ดังที่กล่าวไปแล้วข้างต้น โดยทั่วไปสามารถแบ่งประเภทของเทคนิคที่ใช้ในการอธิบายภาพของวัตถุออกเป็น 2 ประเภทคือ เทคนิคทางเส้นรอบรูป (Contour-based) และเทคนิคทางพื้นที่ (Region-based) การจำแนกประเภทนี้พิจารณาจาก Shape representation นั้นๆหามาจากเส้นรอบรูปหรือว่าบริเวณพื้นที่ทั้งหมดของรูปร่างในภาพ ในแต่ละเทคนิคนั้นยังแบ่งออกเป็น 2 ประเภทย่อยคือ แบบโครงสร้างย่อย (Structural) ใช้อธิบายรูปร่างออกเป็นส่วนย่อยๆ และ แบบโครงสร้างโดยรวม (Global) ใช้อธิบายรูปร่างโดยรวมทั้งหมด นอกจากนี้ยังสามารถแบ่งออกเป็น แบบโดเมนสเปซ (Space domain) ซึ่งเป็นการนำเอาข้อมูลของภาพไปทำการหา Shape representation โดยตรง และแบบโดเมนแปลง (Transform domain) ซึ่งเป็นการนำเอาข้อมูลของภาพไปทำการแปลงเช่น การแปลงแบบฟูเรียร์ (Fourier transform) หรือการแปลงแบบเวฟเลต (Wavelet transform) แล้วค่อยนำข้อมูลที่ทำการแปลงแล้วดังกล่าวไปหา Shape representation อีกที ลำดับขั้นในการแยกประเภท

เอกสารนี้ของเทคนิคที่ใช้ในการอธิบายภาพของวัตถุแสดงดังรูปที่ 2.1 ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 ลำดับชั้นในการแยกประเภทของเทคนิคที่ใช้ในการอธิบายภาพของวัตถุ

2.3 เทคนิคทางเส้นรอบรูป (Contour-based technique)

เทคนิคทางเส้นรอบรูปพิจารณาเฉพาะข้อมูลของเส้นรอบรูป โดยทั่วไปแล้วการสร้างตัวอธิบายวัตถุด้วยเส้นรอบรูป [1] มีอยู่ 2 วิธีที่แตกต่างกันคือ วิธีการแบบต่อเนื่องหรือโครงสร้างโดยรวม (Continuous หรือ Global) และ วิธีการแบบไม่ต่อเนื่องหรือโครงสร้างย่อย (Discrete หรือ Structural) ซึ่งวิธีการแบบต่อเนื่อง (Continuous) หมายถึงการนำเส้นรอบรูปทั้งหมดมาหา ลักษณะเฉพาะเพื่อใช้เป็นตัวแทนของรูปร่างนั้นๆ โดยไม่ได้ทำการแบ่งเป็นส่วนย่อย ดังนั้นจึงเรียกว่าวิธีการแบบต่อเนื่อง สำหรับการวัดความเหมือนของแต่ละรูปร่างก็วัดจากเมตริกซ์ระยะทางระหว่างลักษณะเฉพาะของรูปร่างที่ต้องการวัดความเหมือนกับลักษณะเฉพาะของรูปร่างที่นำมาเปรียบเทียบ ส่วนวิธีแบบไม่ต่อเนื่อง (Discrete) นั้นจะแบ่งเส้นรอบรูปออกเป็นส่วนย่อยๆ ก่อนแล้วค่อยหาลักษณะเฉพาะของแต่ละส่วนย่อยเหล่านั้น แล้วนำมาเป็นตัวแทนของรูปร่าง (Shape representation) ดังนั้นจึงเรียกว่าวิธีแบบไม่ต่อเนื่อง โดยปกติแล้วตัวแทนของรูปร่างที่ได้มาจากวิธีแบบไม่ต่อเนื่อง (Discrete) จะแสดงอยู่ในรูปการจัดเรียงเป็นแบบอนุกรม หรืออยู่ในรูปกราฟ หรืออยู่ในรูปโครงสร้างโคอะแกรม สำหรับการวัดความเหมือนนั้นทำได้โดยการจับคู่รูปร่าง (matching) ในหัวข้อถัดไปจะได้กล่าวถึง ทั้งสองวิธีโดยละเอียด

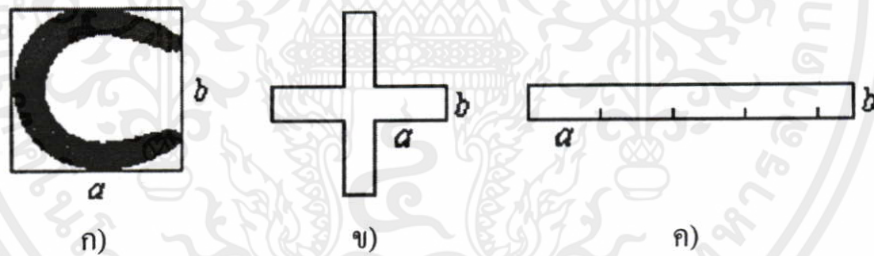
2.3.1 วิธีแบบโครงสร้างโดยรวม (Global)

เทคนิคในการนำเส้นรอบรูปมาหาตัวแทนหรือตัวอธิบายรูปร่างด้วยวิธีแบบโครงสร้างโดยรวม (Global) นั้น โดยปกติแล้วจะนำข้อมูลของเส้นรอบรูปมาหาลักษณะเฉพาะโดยพยายาม

สร้างให้เป็นหลายๆระดับชั้น ต่อไปนี้เป็นตัวอย่างของ Shape representation ที่หามาจากเส้นรอบรูปด้วยวิธีแบบโครงสร้างโดยรวม (Global)

2.3.1.1 ตัวอธิบายรูปร่างแบบง่าย (Simple shape descriptor)

ในงานวิจัย [2] ได้นำเสนอตัวอธิบายรูปร่างที่ง่าย ๆ คือ พื้นที่ (area) ความเป็นวงกลม (circularity: $(2\pi r)^2 \times \text{area}$) ความเบี้ยว (eccentricity: อัตราส่วนระหว่างความยาวของแกนที่มีความยาวมากที่สุดกับความยาวของแกนที่มีความยาวสั้นที่สุด) มุมของแกนที่มีความยาวมากที่สุด และพลังงานการโค้งงอ (bending energy) ซึ่งจะมีค่าน้อยตามลักษณะมุมการโค้งงอของรูปร่าง โดยปกติตัวอธิบายรูปร่างเหล่านี้สามารถแยกแยะรูปร่างที่แตกต่างกันมาก ๆ เท่านั้น เพราะฉะนั้นจึงถูกนำไปใช้เป็นตัวแยกประเภทรูปร่าง โดยหยาบๆ หรือถูกนำไปพิจารณาร่วมกันเพื่อให้สามารถแยกแยะรูปร่างๆ ได้ดีขึ้น ซึ่งถ้าใช้ตัวอธิบายรูปร่างเพียงแบบใดแบบหนึ่งจะไม่เหมาะสม ตัวอย่างเช่น ในกรณีรูปร่างที่แสดงในรูปที่ 2.2 ก) มีค่าความเบี้ยวใกล้เคียง 1 ($a=b$) ซึ่งเป็นตัวอธิบายรูปร่างที่ไม่ถูกต้องเพราะเมื่อมองจากรูปแล้วจะเห็นว่ารูปร่างดังกล่าวเป็นรูปร่างที่มีลักษณะยาวเรียว ในกรณีนี้ความเป็นวงกลมน่าจะเป็นตัวอธิบายได้ดีกว่า รูปที่ 2.2 ข) และ ค) มีค่าความเป็นวงกลมเท่ากัน ($a=2b$) แต่จะเห็นว่าทั้งสองมีรูปร่างที่แตกต่างกันมาก ในกรณีนี้ควรใช้ความเบี้ยวมาเป็นตัวอธิบายรูปร่างจะดีกว่า



รูปที่ 2.2 รูปร่างที่เกิดความผิดพลาดในการวัดความเบี้ยว (eccentricity) และความเป็นวงกลม (circularity)

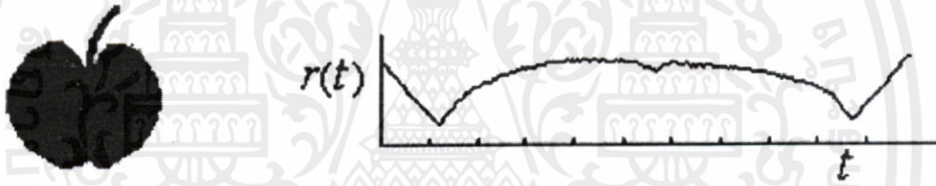
นอกจากนี้ Peura และ Iivarinen [3] ได้นำเสนอตัวแทนหรือตัวอธิบายรูปร่างที่หามาจากเส้นรอบรูปด้วยวิธีแบบโครงสร้างโดยรวม (Global) ซึ่งประกอบด้วย ลักษณะความโค้งเว้า (convexity) อัตราส่วนของแกนหลัก (ratio of principle axis) ความแปรปรวนของวงกลม (circular variance) และ ความแปรปรวนของวงรี (elliptic variance)

2.3.1.2 ตัวอธิบายรูปร่างแบบลายสัญลักษณ์ (Shape signature)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกสิ่งนี้ไป และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลายสัญลักษณ์ใช้เป็นตัวแทนหรืออธิบายรูปร่างในรูปแบบฟังก์ชัน 1 มิติซึ่งหามาจากจุดบนเส้นรอบรูป ลายสัญลักษณ์ ที่มีอยู่มีหลายประเภทเช่น ข้อมูลจุดศูนย์กลาง (centroidal profile) พิกัดเชิงซ้อน (complex coordinates) ระยะทางจากจุดศูนย์กลาง (centroid distance: ดังแสดงในรูปที่

2.3) มุมของเส้นสัมผัส (tangent angle) มุมสะสม (cumulative angle) ความโค้ง (curvature) พื้นที่ (area) และความยาวเส้นคอร์ด (chord-length) [4-6] โดยทั่วไป ลายสัญลักษณ์รูปร่าง จะถูกนอร์มอลไลซ์เพื่อให้มีลักษณะที่ไม่เปลี่ยนแปลงต่อการเลื่อนพิกัด (translation) และการเปลี่ยนขนาด (scale) ในการวัดความเหมือนจะทำโดยการจับคู่ (matching) แต่ในการจับคู่รูปร่าง ระหว่าง Shape signature ของสองรูปร่างนั้นบางทีเกิดปัญหาในลักษณะที่รูปร่างนั้นหมุนหรือเปลี่ยนมุมไปอาจจะทำให้เกิด error ในการวัดความเหมือนได้ ดังนั้นเพื่อลดปัญหานี้ ในการวัดความเหมือนระหว่างสองรูปร่างแต่ละครั้งจะต้องทำการ matching แบบเลื่อนทบรอบ คือเลื่อนจุดเริ่มต้นของ Shape signature แล้วทำการ matching ทำไปจนกว่าจะได้ค่าความเหมือนที่มากที่สุดเพื่อนำไปเป็นค่าความเหมือนระหว่างสองรูปร่างนั้น โดยส่วนใหญ่แล้ว Shape signature จะถูกนอร์มอลไลซ์เพื่อทำการ matching แบบเลื่อนใน 1 มิติ อย่างไรก็ตามบาง Shape signature ก็ทำการ matching ใน 2 มิติเช่น ข้อมูลจุดศูนย์กลาง (centroidal profile) [4] ทั้งสองกรณีนี้ต่างใช้เวลาไม่เหมาะสมสำหรับระบบการค้นหา และกู้คืนแบบออนไลน์



รูปที่ 2.3 รูปร่างแอปเปิ้ลและ Shape signature ของรูปร่างที่หามาจากระยะทางจากจุดศูนย์กลาง

2.3.1.3 ตัวอย่างรูปร่างแบบโมเมนต์เส้นรอบรูป (Boundary moments)

โมเมนต์เส้นรอบรูป (Boundary moments) สามารถนำไปใช้ในการลดมิติของตัวแทนรูปร่าง (Shape representation) ที่หามาจากเส้นรอบรูปได้ จาก [7] ได้ให้นิยามของ โมเมนต์เส้นรอบรูป (Boundary moments) ดังนี้ สมมติว่าเส้นรอบรูปแทนอยู่ในรูปสัญญาณ $(z(i)=(x(i),y(i)))$ โมเมนต์ (moment) ที่ r th คือ m_r , โมเมนต์จุดศูนย์กลาง (central moment) คือ μ_r แสดงอยู่ในรูปสมการได้ดังนี้

$$m_r = \frac{1}{N} \sum_{i=1}^N [z(i)]^r \quad \text{และ} \quad \mu_r = \frac{1}{N} \sum_{i=1}^N [z(i) - m_1]^r \quad (2.1)$$

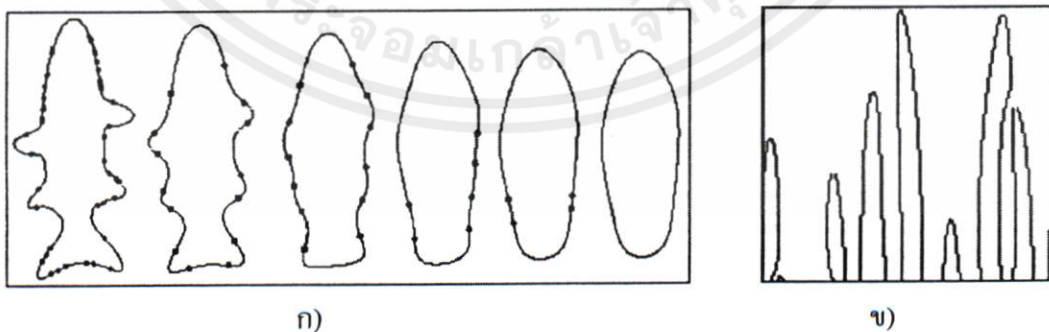
N คือจำนวนจุดบนเส้นรอบรูป สำหรับโมเมนต์ ที่ถูกนอร์มอลไลซ์ $\bar{m}_r = m_r / (\mu_2)^{r/2}$ และ $\bar{\mu}_r = \mu_r / (\mu_2)^{r/2}$ ต่างมีคุณสมบัติที่ไม่เปลี่ยนแปลงต่อการเลื่อนพิกัด การหมุนและการเปลี่ยนขนาด สำหรับตัวอย่างที่มีคุณสมบัติคงทนต่อสัญญาณรบกวน (noise) สามารถหาได้จาก

$$F_1 = (\mu_2)^{1/2} / m_1, \quad F_2 = \mu_3 / (\mu_2)^{3/2} \quad \text{และ} \quad F_3 = \mu_4 / (\mu_2)^2$$

ข้อดีของตัวอธิบายแบบโมเมนต์เส้นรอบรูป (*Boundary moments*) คือง่ายต่อการนำไปใช้ อย่างไรก็ตามโมเมนต์ ที่อันดับสูงๆนั้นยากต่อการนำไปใช้เพราะโดยลักษณะของโมเมนต์ อันดับสูงๆนั้นไม่สามารถบอกความหมายได้ว่าเป็นอะไร

2.3.1.4 ตัวอธิบายรูปร่างแบบวิธีเคอร์ฟเวเจอร์สเกลสเปซ (*Curvature scale space method*)

ปัญหาที่มักจะพบสำหรับ Shape representation ใน spatial domain คือความไม่คงทนต่อสัญญาณรบกวน (noise) และการเปลี่ยนแปลงของเส้นรอบรูป ด้วยเหตุนี้จึงทำให้เกิดการศึกษาวิเคราะห์ในหัวข้อเรื่อง Curvature Scale Space (CSS) ซึ่งนำเสนอโดย Mokhtarian และ Mackworth [8] การแทนหรืออธิบายรูปร่างด้วย Curvature Scale Space เริ่มด้วยการนำเอา Gaussian filter ไปคอนโวลูชันกับเส้นรอบรูปด้วย Gaussian ที่มีค่าความแปรปรวน (variance) ต่างๆ เมื่อใช้ Gaussian ที่มี variance มากขึ้นจะทำให้จุดที่ไม่มีความสำคัญในที่นี้คือมีความโค้งน้อยหายไปและทำให้เส้นรอบรูปเรียบขึ้นดังแสดงในรูปที่ 2.4 ก) จุดบริเวณส่วนโค้งที่เหลือนจะถูกนำไปใช้เป็นลักษณะเฉพาะของรูปร่างนั้นๆ ตัวอย่าง Shape representation ที่ได้มาจากการทำให้เส้นรอบรูปเรียบแล้วพิจารณาว่าแต่ละจุดยังเป็นจุดที่มีความโค้งอยู่ในเกณฑ์ที่กำหนดไว้หรือไม่ แสดงดังรูปที่ 2.4 ข) แกน x คือตำแหน่งจุดบนเส้นรอบรูป แกน y คือ variance ของ Gaussian จุดบนกราฟแสดงถึงความเป็นส่วนโค้ง ตัวอย่างวิธีการหา เมื่อพิจารณาที่จุดบนเส้นรอบรูป (พิจารณาที่แกน x) และพิจารณา variance ของ Gaussian ค่าหนึ่ง (พิจารณาแกน y) ที่นำไป คอนโวลูชันกับเส้นรอบรูป แล้วดูว่าที่จุดที่กำลังพิจารณาอยู่มีค่าความโค้งอยู่ในเกณฑ์ที่กำหนดหรือไม่ ถ้าอยู่ในเกณฑ์ที่กำหนดก็ทำการทำเครื่องหมาย (mark) จุดนั้น ถ้าไม่อยู่ในเกณฑ์ที่กำหนดก็ไม่ต้องทำการ mark อย่างไรก็ตาม Shape representation ชนิดนี้ก็ยากต่อการแปลความหมาย อีกทั้งใช้จำนวน variance ที่มากทำให้ค่อนข้างเสียเวลาในการคำนวณมาก



รูปที่ 2.4 Curvature Scale Space (CSS)

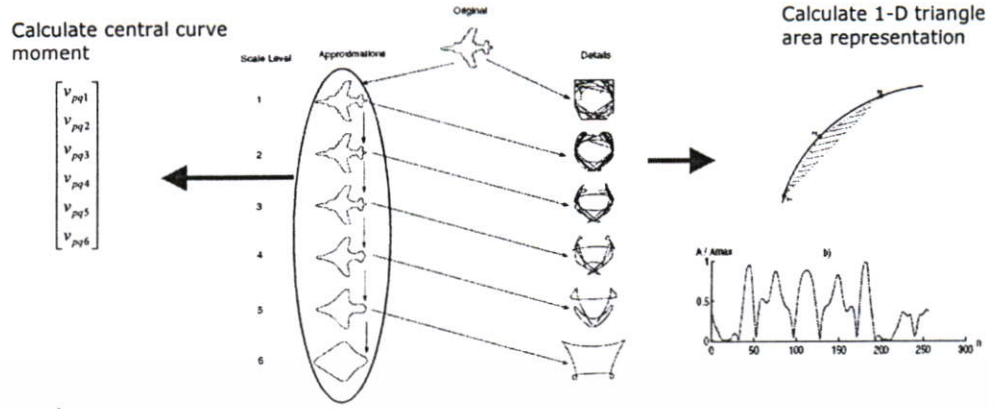
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ภายในเพื่อการศึกษาเท่านั้น มิได้อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1.5 ตัวอธิบายรูปร่างที่ได้มาจากการแปลงสเปกตรอล (Spectral transform)

Spectral transform ในที่นี้ก็คือการแปลงจาก domain ใดๆให้อยู่ในรูป spectral domain โดยปกติแล้วภาพอยู่ในโดเมน 2 มิติในที่นี้คืออยู่ใน spatial domain ในหัวข้อที่ผ่านมาเราได้รวบรวม Shape representation ที่หามาจาก spatial domain แต่สำหรับหัวข้อนี้เราจะนำเสนอ Shape representation ที่หามาจาก Spectral transform ซึ่งสามารถแก้ปัญหาของ noise และการเปลี่ยนแปลงเส้นรอบรูปได้ ตัวอธิบายทาง spectral ที่ใช้กันแพร่หลายในงานวิจัยคือ Fourier descriptor (FD) และ Wavelet descriptor (WD)

Fourier descriptor(FD) ถูกนำมาใช้เป็นตัวอธิบายภาพกันอย่างกว้างขวาง Granlund [9] ได้นำเสนอคุณสมบัติที่ไม่เปลี่ยนแปลงของ Fourier ซึ่งใช้อธิบายลักษณะที่สมมาตรต่อการหมุนของรูปร่าง ในตอนแรกๆนั้น Fourier descriptor (FD) ถูกนำไปหา Shape representation เฉพาะกับเส้นรอบรูปปิดเท่านั้น อย่างไรก็ตาม Lin et al.[10,11] ก็ได้พยายามพัฒนา Fourier descriptor(FD) ให้สามารถหา Shape representation กับเส้นรอบรูปเปิดหรือใช้อธิบายรูปร่างเฉพาะส่วนได้ต่อมา Arbter et al. [12,13] ได้ใช้ Fourier มาหาลักษณะที่ไม่เปลี่ยนแปลงต่อการแปลงแบบแอฟไฟน์ของแต่ละรูปร่าง เพื่อให้ได้ Shape representation ที่มีคุณสมบัติคงที่ไม่ว่ารูปร่างนั้นจะถูกเปลี่ยนขนาด เปลี่ยนตำแหน่ง หมุน หรือว่าบิดเบือน ไปก็ตาม

เมื่อไม่นานมานี้ นักวิจัยหลายคนได้กล่าวถึงการนำ Wavelet มาใช้อธิบายภาพ [14-16] ข้อดีของ Wavelet คือทนต่อสัญญาณรบกวนสามารถแยกองค์ประกอบออกเป็นหลายระดับ ทำให้สามารถอธิบายภาพได้ละเอียด แต่อย่างไรก็ตามที่ระดับ สูงๆนั้น จะมีข้อมูลอยู่น้อย นักวิจัยบางคน [15] จึงใช้ระดับต่างๆบางระดับ เพื่อนำมาอธิบายภาพหรือหา Shape representation ยิ่งไปกว่านั้น ขั้นตอนในการนำ Wavelet descriptor (WD) ไปใช้ในการจับคู่ (match) รูปร่างนั้นก็มีความซับซ้อนมากจึงไม่เหมาะสมกับการนำไปใช้งานทางด้านงานสืบค้นรูปร่างในฐานะข้อมูลแบบออนไลน์ ตัวอย่างงานวิจัยที่หา Shape representation ใน wavelet domain เช่นงานวิจัยของ Ibrahim El Rube', Maher Ahmed และ Mohamed Kamel [17] โดยมีหลักการแสดงดังรูปที่ 2.5 คือนำเอาคู่อันดับ (x,y) ของเส้นรอบรูปไปแปลงแบบเวฟเลทจะได้องค์ประกอบออกมา 2 ส่วนคือ สัมประสิทธิ์โดยประมาณ (Approximate coefficient) และ สัมประสิทธิ์โดยละเอียด (Detail coefficient) ซึ่งทั้งสององค์ประกอบนั้นแบ่งออกเป็น 6 ระดับย่อย จากนั้นก็นำองค์ประกอบใน wavelet domain ที่แยกออกมาได้เหล่านั้นไปหา shape representation โดยสัมประสิทธิ์โดยประมาณทั้ง 6 ระดับนั้นจะถูกนำไปคำนวณหาค่าโมเมนต์จุดศูนย์กลาง และสัมประสิทธิ์โดยละเอียดในบางระดับ จะถูกนำไปคำนวณ ตัวแทนพื้นที่สามเหลี่ยมใน 1 มิติ (1-D triangle area representation) ซึ่ง shape representation ทั้งสองชนิดที่คำนวณมาได้นั้นมีคุณสมบัติไม่เปลี่ยนแปลงต่อการแปลงแบบแอฟไฟน์



รูปที่ 2.5 ตัวอย่างการนำเอาเส้นรอบรูปมาหา Shape representation ในทางเวฟเลท โดเมน

2.3.2 วิธีแบบโครงสร้างย่อย (Structural)

เทคนิคในการนำเส้นรอบรูปมาหาตัวแทนหรือตัวอธิบายรูปร่างด้วยวิธีแบบ โครงสร้างย่อย (Structural) นั้น โดยปกติแล้วจะนำเส้นรอบรูปมาแบ่งออกเป็นส่วนย่อยๆ สำหรับวิธีแบบ โครงสร้างย่อย (Structural) นั้น มีรูปแบบที่แตกต่างกันออกไปทั้งในด้านวิธีการเลือกแบ่งเส้นรอบ รูปออกเป็นส่วนย่อยและการจัดเรียงหรือการจัดการกับส่วนย่อยนั้นๆ โดยปกติแล้ววิธีการแบ่งเส้น รอบรูปออกเป็นส่วนย่อยนั้นจะมีพื้นฐานมาจากการประมาณลักษณะหรือจำนวนด้านของเส้นรอบ รูปนั้นๆ หรืออาจจะแบ่งเส้นรอบรูปออกเป็นส่วนย่อยตามลักษณะความ โค้งของแต่ละจุดบนเส้น รอบรูป หรือนำเอาหลักการของการหาสมการแทนเส้น โค้ง Curve fitting [18] มาแบ่งเส้นรอบรูป ให้เป็นส่วนย่อย เมื่อเราได้ทำการแยกเส้นรอบรูปออกเป็นส่วนๆแล้ว ส่วนย่อยแต่ละส่วนนั้นจะถูก นำมาเรียงกันอยู่ในรูปของสตริง (string) ที่มีรูปแบบโดยทั่วไปดังสมการที่ 2.2

$$S = s_1, s_2, \dots, s_n \tag{2.2}$$

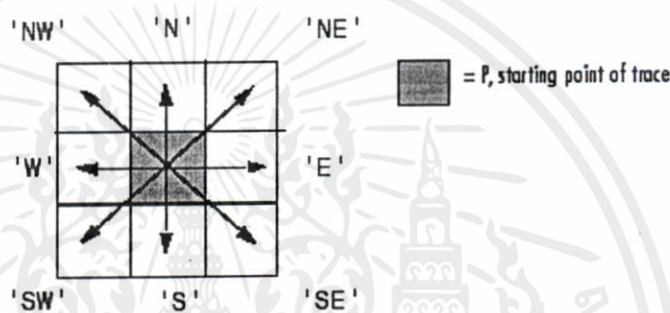
ซึ่ง s_i อาจจะเป็นองค์ประกอบย่อยของรหัสลูกโซ่ (chain code) หรือ ด้านของรูปหลายเหลี่ยม หรือ ส่วนโค้ง หรือ สไปน์ (spline) เป็นต้น s_i นั้นเป็นตัวอธิบายทางคณิตศาสตร์ที่อาจจะหาได้จาก ความ ยาวหรือ ความ โค้งเฉลี่ยหรือความ โค้งที่มากที่สุดหรือพลังงานการ โค้งงอ (bending energy) หรือ ค่า มุม เป็นต้น สตริง (string) เหล่านี้สามารถนำมาอธิบายและวิเคราะห์โครงสร้างของรูปร่างได้ ใน หัวข้อถัดไปเป็นการยกตัวอย่างวิธีการในการหาตัวอธิบายรูปร่างในรูปแบบของ string เพื่อนำมา เป็นตัวแทนแต่ละรูปร่าง

2.3.2.1 ตัวอธิบายรูปร่างแบบรหัสห่วงโซ่ (Chain code representation)

Chain code ใช้อธิบายรูปร่าง ในลักษณะการเรียงลำดับเส้น โค้งย่อยของเส้นรอบรูปตามมุม ของส่วนย่อยนั้น สำหรับวิธีการของ chain code นั้นถูกนำเสนอในปี 1961 โดย Freeman [19] ซึ่งได้ ทำการอธิบายถึงการเข้ารหัสรูปร่างลักษณะเชิงเรขาคณิตใดๆ ในวิธีนี้ได้ทำการแทนส่วนโค้งใดๆ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือมีเงื่อนไขการใช้งานอื่นใดที่ปรากฏในเอกสารฉบับนี้ ผู้ใช้ควรปฏิบัติตามเงื่อนไขการใช้งานที่ปรากฏในเอกสารฉบับนี้

ด้วยลำดับของกลุ่มเวกเตอร์หน่วยความยาวกลุ่มเล็กๆและกลุ่มของทิศทางที่เป็นไปได้ของแต่ละเวกเตอร์ เพราะฉะนั้นจึงเรียกวิธีนี้ว่า วิธี unit-vector ในด้านการใช้งานนั้น chain code นำมาประยุกต์ในการหารูปร่างในภาพ โดยทำการหาเส้นรอบรูปของรูปร่างที่ต้องการ ซึ่ง chain code นั้นถูกสร้างขึ้น เริ่มด้วยการกำหนดจุดเริ่มต้นบนขอบของรูปร่างก่อนจากนั้นก็พิจารณาจุดรอบๆจุดเริ่มต้นในลักษณะ 4 ทิศทางหรือ 8 ทิศทางเพื่อหาว่าจุดใดเป็นส่วนหนึ่งรูปร่าง หลังจากนั้นก็นำจุดบนเส้นรอบรูปใหม่ที่ได้นั้นมาพิจารณาแล้วทำเหมือนเดิมคือพิจารณาจุดรอบๆเพื่อหาจุดบนเส้นรอบรูปถัดไป ทำอย่างนี้ไปเรื่อยๆจนจุดสุดท้ายตรงกับจุดเริ่มต้นจะได้คู่อันดับ (x,y) ของเส้นรอบรูปทั้งหมด ดังแสดงดังรูปที่ 2.6



รูปที่ 2.6 หลักการสร้าง chain code โดยเริ่มจากกำหนดจุดเริ่มต้นบนเส้นขอบภาพและทิศทางที่เป็นไปได้ในการพิจารณาหาจุดบนเส้นรอบรูปถัดไป

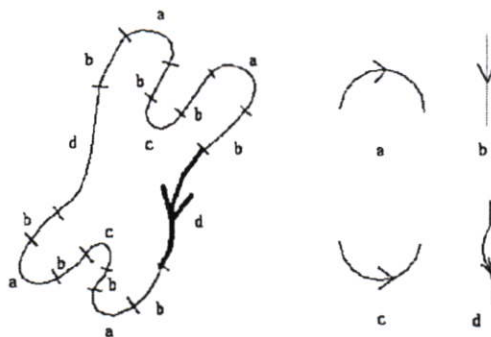
2.3.2.2 ตัวอย่างรูปร่างตามลักษณะเหลี่ยมมุม (Polygon decomposition)

ในงานวิจัย [20,21] ได้นำเสนอวิธี Polygon decomposition โดยนำเอาเส้นรอบรูปแยกออกเป็นส่วนโค้งย่อยตามการประมาณลักษณะด้านหรือเหลี่ยมมุมของรูปทรงนั้น จากนั้นก็นำส่วนโค้งย่อยแต่ละส่วนที่แยกออกมาได้มาหาคุณลักษณะเฉพาะที่ประกอบด้วยสตริง (string) ที่มีพารามิเตอร์ 4 แบบ คือ มุมภายใน ระยะทางระหว่างจุดยอดมุม พิกัด x และพิกัด y

2.3.2.3 ตัวอย่างรูปร่างที่อาศัยหลักการวิเคราะห์ทางไวยากรณ์ภาษา (Syntactic analysis)

การวิเคราะห์ทางไวยากรณ์ภาษา (Syntactic analysis) อาศัยหลักการของลักษณะโครงสร้างของภาษา [22] ที่ว่าด้วยลักษณะรูปแบบของประโยคซึ่งสร้างมาจากวลี วลีสร้างมาจากคำ คำสร้างมาจากตัวอักษร เป็นต้น สำหรับการหาตัวอย่างหรือตัวแทนรูปร่างตามการสร้างประโยคนั้นรูปร่างจะถูกแทนด้วยกลุ่มของลักษณะพื้นฐานที่ถูกกำหนดขึ้น (predefined primitives) หรือเรียกว่า codebook ส่วนแต่ละลักษณะพื้นฐาน (primitive) นั้นเรียกว่า codeword ตัวอย่างเช่น กำหนดให้ codeword มีลักษณะดังรูปทางขวาของรูปที่ 2.7 ซึ่งแสดงอยู่ในรูปไวยากรณ์แบบ string-S ดังสมการที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกร ใช้งานเพื่อการศึษาเท่านั้น ไม่อนุญาตให้ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทางสน ออกทงห้ามมิให้คิดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งท้มีการนำไปใช้



รูปที่ 2.7 ตัวอย่างทางเส้นรอบรูปที่หามาจากวิธี structural ในแบบ syntactic analysis

$$S=dbabcbabdbabcbab \quad (2.3)$$

2.4 เทคนิคทางพื้นที่ (Region-based technique)

การหาตัวอย่างรูปร่างในภาพด้วยเทคนิคทางพื้นที่นั้น จะนำทุกพิกเซลที่อยู่ในรูปร่างไปพิจารณาแทนที่จะพิจารณาเฉพาะเส้นรอบรูป ดังเช่นในเทคนิคทางเส้นรอบรูป โดยปกติแล้ว เทคนิคทางพื้นที่จะอธิบายรูปร่างด้วยโมเมนต์ (moment) สำหรับเทคนิคทางพื้นที่อื่น ๆ นั้น ประกอบด้วย วิธี grid, shape matrix, convex hull และ media axis เป็นต้น ส่วนการแบ่งประเภทของเทคนิคทางพื้นที่ที่มีหลักการเหมือนกับเทคนิคทางเส้นรอบรูปคือ แบ่งเป็นวิธีแบบโครงสร้างโดยรวม (global) และ วิธีแบบโครงสร้างย่อย (structural)

2.4.1 วิธีแบบโครงสร้างโดยรวม (Global)

ในส่วนของวิธีแบบโครงสร้างโดยรวม (Global) จะนำทุกพิกเซลที่อยู่ในรูปร่างทั้งหมดมาหาตัวอย่างพร้อมกันหมดโดยไม่มีการแยกพิกเซลภายในรูปร่างออกเป็นส่วนๆ เหมือนดังในวิธีแบบโครงสร้างย่อย (Structural) โดยตัวอย่างหรือตัวแทนรูปร่างที่หามาจากวิธีแบบโครงสร้างโดยรวม (Global) จะอยู่ในรูปเวกเตอร์ที่มีองค์ประกอบเชิงตัวเลข ซึ่งเวกเตอร์นี้จะแทนคุณลักษณะเฉพาะของแต่ละรูปร่าง เราสามารถวัดความเหมือนของแต่ละรูปร่างได้จากการนำเอาเวกเตอร์คุณลักษณะมาหาเมทริกซ์ระยะทาง (Distance matrix)

2.4.1.1 ตัวอย่างรูปร่างแบบโมเมนต์ที่มีคุณสมบัติไม่เปลี่ยนแปลงต่อการแปลงเชิงเรขาคณิต (Geometric moment invariants)

ในอดีตที่ผ่านมา Hu [23] ได้นำเสนอบทความที่มีความสำคัญบทความแรกเกี่ยวกับการนำเอาโมเมนต์ (moment) ซึ่งเป็นความรู้ทางสถิติมาหาคุณสมบัติที่ไม่เปลี่ยนแปลงของรูปภาพ 2 มิติ เพื่อนำไปประยุกต์ใช้กับรูปแบบการรู้จำ วิธีการของ Hu ได้อาศัยหลักการทางคณิตศาสตร์ของนักคณิตศาสตร์ในศตวรรษที่ 19 คือ Boole, Cayley และ Sylvester ซึ่งมีพื้นฐานอยู่บนทฤษฎีของรูปแบบฟังก์ชัน (algebraic) ดังแสดงในสมการที่ 2.4

$$m_{pq} = \sum_x \sum_y x^p y^q f(x, y) \quad p, q = 0, 1, 2, \dots \quad (2.4)$$

โมเมนต์ของภาพในสมการที่ 2.4 สามารถแบ่งออกเป็นหลายระดับ (level) โดยโมเมนต์ที่ระดับต่ำๆจะมีคุณสมบัติที่ไม่ผันแปรต่อการแปลงเชิงเรขาคณิต (geometric moment invariant) คือไม่เปลี่ยนแปลงต่อการเลื่อนพิกัด ขนาดและการหมุน สำหรับโมเมนต์ที่ระดับสูงๆ ไม่ค่อยมีความหมาย อีกทั้งไม่ถูกกล่าวถึงในการนำไปใช้สำหรับการรู้จำ ปัญหาหลักของ geometric moment คือการใช้แค่เพียงโมเมนต์ที่ระดับต่ำๆนั้นไม่เพียงพอสำหรับการอธิบายรูปร่างให้มีความถูกต้องสูง อีกทั้งโมเมนต์ที่ระดับสูง ก็ยากต่อการคำนวณ

2.4.1.2 ตัวอธิบายรูปร่างแบบโมเมนต์ที่ไม่ผันแปรในเชิงพีชคณิต (Algebraic moment invariants)

Taubin และ Cooper [24, 25] ได้นำเสนอตัวอธิบายรูปร่างที่ไม่ผันแปรเชิงพีชคณิต (Algebraic moment invariant) ซึ่งคำนวณมาจากโมเมนต์จุดศูนย์กลาง (central moment) m ระดับแรก ดังแสดงในสมการที่ 2.5

$$\text{Central moment} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (2.5)$$

จากนั้นนำสัมประสิทธิ์ของโมเมนต์จุดศูนย์กลางเหล่านี้ไปสร้างเป็นเมทริกซ์ $M_{(i,j)}$ เพื่อคำนวณหาค่าไอเกน (Eigen value) Algebraic moment invariant แตกต่างกับ โมเมนต์ของ Hu ตรงที่ Algebraic moment invariant นั้นไม่ว่าจะเป็นอันดับหรือระดับใด ต่างก็มีคุณสมบัติที่ไม่แปรผันต่อการแปลงแบบแอฟไฟน์

2.4.1.3 ตัวอธิบายรูปร่างแบบออร์โธโกนอลโมเมนต์ (Orthogonal moments)

Teague [26] ได้นำเสนอ orthogonal moment 2 แบบ คือ Legendre moments และ Zernike moments ซึ่ง orthogonal moment ทั้ง 2 แบบได้มาจากการแปลงสมการโมเมนต์ ในสมการที่ 2.4 โดยการแทนตัวแปร $x^p y^q$ ด้วย Legendre polynomial และ Zernike polynomial เพื่อให้ได้ Legendre moments และ Zernike moments ดังแสดงในสมการที่ 2.6 และสมการที่ 2.7 ตามลำดับ

- Legendre moments สามารถอธิบายอยู่ในรูปสมการดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้เฉพาะเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องส่งมอบคืนเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\lambda_{mn} = \frac{(2m+1)(2n+1)}{4} \sum_x \sum_y P_m(x) P_n(y) f(x, y) \quad (2.6)$$

$$\text{ซึ่ง} \quad P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} (x^2 - 1)^n$$

- Zernike moments สามารถอธิบายอยู่ในรูปสมการดังนี้

$$A_{nm} = \frac{n+1}{\pi} \sum_x \sum_y V_{nm}^*(x, y) f(x, y), \quad x^2 + y^2 \leq 1 \quad (2.7)$$

$$\text{ซึ่ง} \quad V_{nm}(x, y) = V_{nm}(\rho \cos \theta, \rho \sin \theta) = R_{nm}(\rho) \exp(jm\theta)$$

$$\text{และ} \quad R_{nm} = \sum_{s=0}^{(n-|m|)/2} (-1)^s \times \frac{(n-s)!}{s!((n+|m|)/2-s)!((n-|m|)/2-s)!} \rho^{n-2s}$$

ρ และ θ คือรัศมีและมุมของพิกเซล (x, y) เมื่อวัดจากศูนย์กลางมวลของรูปร่าง

2.4.1.4 ตัวอธิบายรูปร่างในเชิงฟูเรียร์ (Generic Fourier descriptor)

Generic Fourier descriptor (GFD) ได้ถูกนำเสนอโดย Zhang และ Lu [27] มีวิธีการคือนำเอาภาพ Polar-raster ของรูปร่าง (ตัวอย่าง รูป Polar-raster แสดงดังรูปที่ 2.8) มาแปลงเป็น Fourier ใน 2 มิติ ตามสมการที่ 2.8



รูปที่ 2.8 ก) รูปร่างเดิมก่อนจะถูกแปลงเป็นรูป polar-raster

ข) รูป polar-raster ของรูป ก) กราฟอยู่ในพิกัด Cartesian

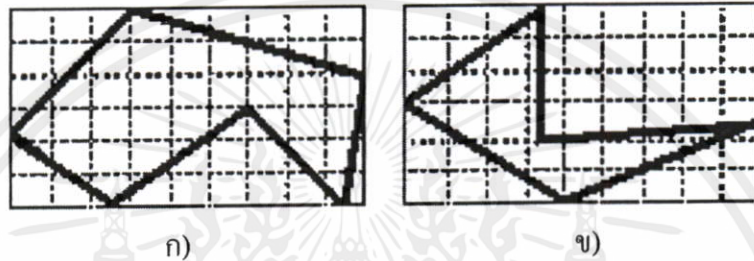
$$PF_2(\rho, \phi) = \sum_r \sum_i f(r, \theta_i) \exp \left[j2\pi \left(\frac{r}{R} \rho + \frac{2\pi i}{T} \phi \right) \right] \quad (2.8)$$

โดยที่ $0 \leq r \leq R$ และ $\theta_i = i(2\pi/T)$ ($0 \leq i < T$); $0 \leq \rho < R$, $0 \leq \phi < T$ ซึ่ง R และ T คือ radial frequency resolution และ angular frequency resolution ตามลำดับ

2.4.1.5 ตัวอธิบายรูปร่างที่อาศัยการสร้างกริด (Grid based method)

การอธิบายรูปร่างด้วยกริด นั้นถูกนำเสนอโดย Lu และ Sajjanhar [28] และถูกนำไปใช้ใน งานวิจัย [29-31] วิธีการนั้นเริ่มด้วยการตีเส้นกริด (grid) ไปบนรูปร่าง ทำให้เกิดเป็นเซลล์ (cell)

ย่อยๆ แล้วทำการสแกนเส้นกริด (grid) จากซ้ายไปขวาและบนลงล่าง เพื่อดูว่ามีเซลล์ (cell) ไหนบ้างที่ซ้อนทับ (overlap) กับรูปร่างอยู่ ถ้าหาก cell นั้นซ้อนทับกับรูปร่างอยู่ก็กำหนดค่าให้เป็น 1 ในทางกลับกันถ้าหาก cell นั้นไม่ได้ซ้อนทับกับรูปร่างก็กำหนดค่าให้เป็น 0 ในที่สุดเราก็จะได้เวกเตอร์ที่มีองค์ประกอบภายในเป็น 0 กับ 1 (binary vector) ซึ่งใช้เป็นคุณลักษณะเฉพาะที่แทนแต่ละรูปร่างสำหรับตัวอย่างตัวอักษรรูปร่างในรูปที่ 2.9 ซึ่งหามาจากวิธี grid คือ 001111000 011111111 111111111 111111111 111110011 001100011 สำหรับรูป ก) และ 001100000 011100000 111100000 111100000 011111100 000111000 สำหรับรูป ข)

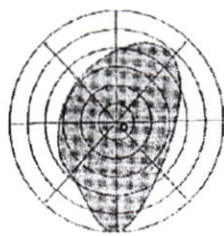


รูปที่ 2.9 การสร้าง grid ไปบนรูปร่าง

2.4.1.6 ตัวอักษรรูปร่างในเชิงเมทริกซ์ (Shape matrix)

จากหัวข้อที่ผ่านมาเป็นการหาตัวอักษรรูปร่างด้วยการใช้กริด (grid) หรือเซลล์สี่เหลี่ยมย่อย ซึ่งตัวอักษรรูปร่างที่ได้นั้น ไม่คงที่เมื่อมีการเลื่อนตำแหน่ง การหมุนและการเปลี่ยนขนาด Goshtasby [32] จึงได้นำเสนอตัวอักษรรูปร่างใหม่ที่เรียกว่า Shape matrix หลักการก็คล้ายๆกับการพิจารณาเซลล์สี่เหลี่ยมที่ซ้อนทับ (overlap) กับรูปร่างดังที่กล่าวไว้ในหัวข้อที่ผ่านมา แต่แทนที่จะใช้เส้นกริด (grid) ธรรมดาที่ใช้เส้นวงกลมที่มีจุดศูนย์กลางอยู่ที่จุดศูนย์กลางมวลของรูปร่างและเส้นรัศมีวงกลมดังแสดงในรูปที่ 2.10 แล้วจึงพิจารณาว่าบริเวณที่เส้นวงกลมและรัศมีวงกลมตัดกัน (intersect) กันนั้นเป็นส่วนหนึ่งของรูปร่างหรือไม่ โดยจะเริ่มพิจารณาจากเส้นรัศมีที่มีความยาวมากที่สุด ถ้าเป็นส่วนหนึ่งของรูปร่างก็กำหนดให้เป็น 1 ถ้าไม่เป็นก็กำหนดให้เป็น 0 จากนั้นจึงนำค่าไบนารี (binary) ดังกล่าวมาสร้างเมทริกซ์ โดยแต่ละคอลัมน์ (column) แทนวงกลมแต่ละเส้น และแต่ละแถว (row) แทนรัศมีของวงกลม ตัวอักษรรูปร่างในเชิงเมทริกซ์ (shape matrix) มีคุณสมบัติที่ไม่เปลี่ยนแปลงต่อการเลื่อนตำแหน่ง การหมุนและการเปลี่ยนขนาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



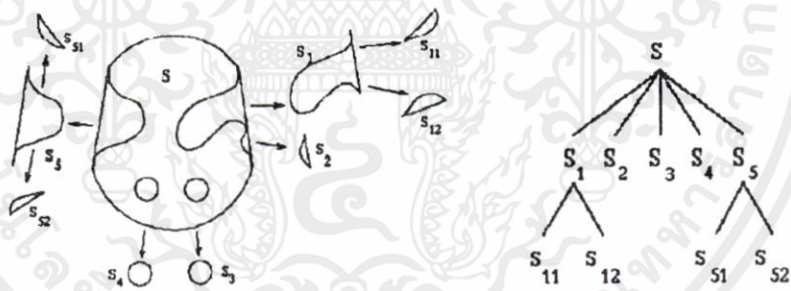
รูปที่ 2.10 การสร้างเส้นวงกลมและเส้นรัศมีวงกลมไปบนรูปร่าง

2.4.2 วิธีแบบโครงสร้างย่อย (Structural)

วิธีแบบ โครงสร้างย่อย (structural) นั้นพื้นที่ทั้งหมดของรูปร่างจะถูกแบ่งเป็นพื้นที่ส่วนย่อยๆก่อนที่จะนำไปหาตัวอธิบายหรือตัวแทนรูปร่าง ในหัวข้อต่อไปเป็นตัวอย่างของตัวอธิบายรูปร่างที่ได้มาจากวิธีแบบ โครงสร้างย่อย

2.4.2.1 ตัวอธิบายรูปร่างแบบคอนเวกซ์ฮิว (Convex hull)

เราสามารถแยก Convex hull ออกจากรูปร่างด้วยวิธีการตามรอยเส้นรอบรูป (boundary tracing) หรืออาจใช้วิธีมอโฟโลยี (morphology) ก็ได้ โดยพยายามแยกรูปร่างออกตามลักษณะโค้งมนในแต่ละส่วนดังแสดงในรูปที่ 2.11 ก) สำหรับการนำแต่ละส่วนย่อยมารวมกันเพื่อเป็นตัวแทนของรูปร่างนั้นอาจจะรวบรวมให้อยู่ในรูปสตริง (string) หรือโคอะแกรม ดังแสดงในรูปที่ 2.11 ข)



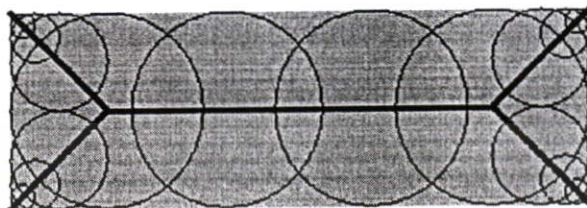
รูปที่ 2.11 ก) Convex hull และแต่ละส่วนย่อย

ข) การนำแต่ละส่วนย่อยของ Convex hull มารวมรวมกันให้อยู่ในรูปโคอะแกรม

2.4.2.2 ตัวอธิบายรูปร่างที่อาศัยหลักการของการหาแกนกลาง (Medial axis)

โครงร่างหรือเค้าโครง (Skeleton) ของรูปร่างสามารถนำมาใช้เป็นตัวอธิบายหรือตัวแทนรูปร่างได้ ซึ่งเราอาจนิยามโครงร่างของรูปร่างในความหมายทางเทคนิคการหาตัวแทนภาพได้ดังนี้ กลุ่มของแกนกลาง (Medial axis) ในแต่ละส่วนย่อยของรูปร่างซึ่งได้เชื่อมต่อกันเป็นโครงสร้างทั้งหมดของรูปร่าง แนวคิดของเทคนิคการแทนรูปร่างด้วยโครงร่างนั้น เกิดมาจากการต้องการลดจำนวนข้อมูลที่ใช้แทนรูปร่างให้เหลือเฉพาะโครงร่างเท่านั้น วิธีการแบบโครงร่างนั้นถูกนำเสนอ

โดย Blum [33] เรียก Blum's medial axis transform (MAT) แกนกลาง (medial axis) นั้นหมายถึง ตำแหน่งศูนย์กลางในแต่ละส่วนของรูปร่าง ดังแสดงตัวอย่างในรูปที่ 2.12 เส้นสีดำที่บดคือ skeleton ของรูปร่างสี่เหลี่ยมที่ถูกร่าง



รูปที่ 2.12 แกน โครงร่างของรูปร่างสี่เหลี่ยม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

ตัวแทนรูปร่างซึ่งไม่ผันแปรต่อการแปลงแบบแอฟไฟน์

(Affine Invariant Shape Representation)

3.1 บทนำ

โดยทั่วไปภาพของวัตถุหนึ่งที่ได้มาจากแต่ละมุมมองนั้นจะแตกต่างกันไปทั้งขนาด (Scaling) ตำแหน่ง (Translation) ลักษณะของวัตถุที่หมุน (Rotation) และเฉือน (Shear) ซึ่งตัวแปรที่เปลี่ยนไปดังกล่าวนี้คือ ตัวแปรของการแปลงแบบแอฟไฟน์ ในบทนี้เราจะอธิบายถึงทฤษฎีและขั้นตอนในการหาลักษณะเฉพาะของรูปร่างซึ่งไม่ผันแปรต่อการแปลงแบบแอฟไฟน์ เพื่อนำไปใช้กับขั้นตอนการรู้จำต่อไป โดยเนื้อหาในบทนี้เราจะอธิบายการแปลงแบบแอฟไฟน์ในเชิงคณิตศาสตร์ในหัวข้อที่ 3.2 ทฤษฎีทอรัมที่นำมาหาตัวแทนรูปร่าง (Shape representation) ในหัวข้อที่ 3.3 และ หลักการและการประยุกต์ใช้ทฤษฎีทอรัมในการหาลักษณะเฉพาะของรูปร่างซึ่งไม่ผันแปรต่อการแปลงแบบแอฟไฟน์ในหัวข้อที่ 3.4

3.2 การแปลงแบบแอฟไฟน์ (Affine Transformation)

โดยทั่วไปแล้วภาพดิจิทัลประกอบด้วยอาร์เรย์สองมิติที่เป็นข้อมูลแสดงสีหรือความเข้มของแสง เรียกองค์ประกอบของข้อมูลเหล่านี้ว่า “พิกเซล” (Pixel) โดยแต่ละพิกเซลจะมีข้อมูลที่สำคัญคือ ตำแหน่งหรือพิกัดของพิกเซลในภาพนั้นๆ และค่าของสีของพิกเซลนั้น สำหรับภาพของวัตถุที่ถูกแปลงแบบแอฟไฟน์เราสามารถอธิบายความสัมพันธ์ระหว่างภาพต้นฉบับกับภาพที่ถูกแปลงแอฟไฟน์ได้ด้วยสมการเมทริกซ์ (Matrix) [34] โดยพิจารณาค่าตำแหน่งพิกเซลของภาพทั้งสองแสดงดังสมการที่ (3.1)

การแปลงแบบแอฟไฟน์ใน 2 มิติ ($T_{affine}: R^2 \rightarrow R^2$) เขียนอยู่ในรูปสมการเมทริกซ์ได้ดังนี้

$$T\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} \quad (3.1)$$

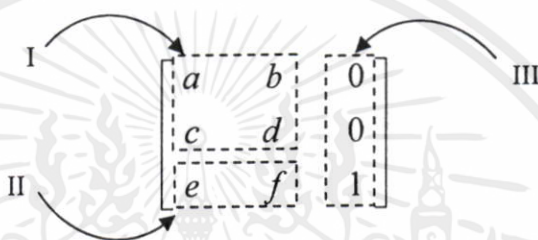
ซึ่ง $\begin{bmatrix} x \\ y \end{bmatrix}$ และ $T\left(\begin{bmatrix} x \\ y \end{bmatrix}\right)$ คือ พิกัดพิกเซลของภาพต้นฉบับกับภาพที่ถูกแปลงแอฟไฟน์ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
คือ ตัวดำเนินการของการแปลงทางขนาด (Scaling) การหมุน (Rotation) และ การ
ไม่ว่ากรณีใดๆ ทั้งนั้นมิให้คิดเปลี่ยนเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
เฉือน (Shear)

$\begin{bmatrix} e \\ f \end{bmatrix}$ คือ ตัวดำเนินการของการแปลงทางตำแหน่ง (Translation)

การพิจารณาการแปลงแบบแอฟไฟน์ใน 2 มิตินั้นจะพิจารณาได้ว่าเป็นการแปลงจุดใน 2 มิติ จากสมการที่ 3.1 นั้นจะเห็นว่าตัวดำเนินการทางการแปลงนั้นแยกกันอยู่ ($\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ และ $\begin{bmatrix} e \\ f \end{bmatrix}$) ซึ่งจะเกิดความซับซ้อนในการคำนวณทางเมทริกซ์ เพื่อให้เกิดความง่ายในการคำนวณ เราต้องทำการรวมตัวดำเนินการทางการแปลงให้อยู่ในเมทริกซ์เดียวเพื่อให้สามารถเขียนให้อยู่ในรูปสมการเชิงเส้นโดยนำระบบพิกัดโฮโมจีเนียส (Homogeneous Coordinate System) มาทำการรวมตัวดำเนินการดังกล่าวให้อยู่ในเมทริกซ์เดียว

ในระบบพิกัดโฮโมจีเนียส พิกัดพิกเซลในภาพ (x,y) จะถูกแทนด้วยระบบ 3 มิติ คือ (x,y,1) สำหรับเมทริกซ์ของการแปลงแบบแอฟไฟน์นั้นแสดงได้ดังรูปที่ 3.1







รูปที่ 3.1 ส่วนประกอบของเมทริกซ์การแปลงแบบแอฟไฟน์ในระบบพิกัดโฮโมจีเนียส

จากรูปที่ 3.1 เมทริกซ์ย่อย I เป็นส่วนของตัวดำเนินการของการแปลงทางขนาด(Scaling) การหมุน (Rotation) และ การเฉือน (Shear) ส่วนเมทริกซ์ย่อย II เป็นส่วนของตัวดำเนินการของการแปลงทางตำแหน่ง (Translation) และในส่วนสุดท้ายเมทริกซ์ย่อย III เป็นเมทริกซ์ที่ใช้รวมกันระหว่าง ตัวดำเนินการของการแปลงทางตำแหน่ง กับการแปลงทางขนาด การหมุนและการเฉือน ($\begin{bmatrix} e \\ f \end{bmatrix}$ และ $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$) ซึ่งสมการของการแปลงแบบแอฟไฟน์ในระบบพิกัดโฮโมจีเนียสสามารถแสดงดังสมการที่ 3.2 โดยรายละเอียดการแปลงแบบต่างๆแสดงในตารางที่ 3.1

$$T \left(\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \right) = \begin{bmatrix} a & b & 0 \\ c & d & 0 \\ e & f & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{3.2}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 แสดงตัวอย่างภาพและเมทริกซ์ของการแปลงแบบแอฟไฟน์แบบต่างๆ

Affine Transform	Example	Transformation Matrix	
Translation		$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ t_x & t_y & 1 \end{bmatrix}$	t_x specifies the displacement along the x axis t_y specifies the displacement along the y axis.
Scale		$\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	s_x specifies the scale factor along the x axis s_y specifies the scale factor along the y axis.
Shear		$\begin{bmatrix} 1 & sh_x & 0 \\ sh_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	sh_x specifies the shear factor along the x axis sh_y specifies the shear factor along the y axis.
Rotation		$\begin{bmatrix} \cos(q) & \sin(q) & 0 \\ -\sin(q) & \cos(q) & 0 \\ 0 & 0 & 1 \end{bmatrix}$	q specifies the angle of rotation.

3.3 ทฤษฎีดีเทอร์มิแนนต์ (Determinant Theory)

ดีเทอร์มิแนนต์เป็นทฤษฎีทางคณิตศาสตร์ที่ใช้ในการคำนวณเมทริกซ์ ซึ่งดีเทอร์มิแนนต์มีบทบาทสำคัญในการอธิบายและวิเคราะห์ข้อมูลของแต่ละเมทริกซ์ ความรู้ทางด้านดีเทอร์มิแนนต์สามารถนำไปประยุกต์ใช้กับการแก้ปัญหาในระบบสมการเชิงเส้นได้

สำหรับความเป็นมาของดีเทอร์มิแนนต์นั้นเริ่มขึ้นเมื่อปี ค.ศ.1683 โดยนักคณิตศาสตร์ชาวญี่ปุ่นชื่อ Seki Takakazu และ ในปี ค.ศ. 1693 โดยนักคณิตศาสตร์ชาวเยอรมันชื่อ Gottfried Leibniz โดยทั้งสองต่างระบุไว้ว่า ดีเทอร์มิแนนต์คือตัวเลขที่เรียงอยู่ในรูปสี่เหลี่ยมจัตุรัส หลังจากนั้นดีเทอร์มิแนนต์ก็ถูกนำไปใช้เป็นตัวอธิบายหลักของระบบสมการเชิงเส้นเป็นเวลาหลายปี จนในปีค.ศ. 1750 นักคณิตศาสตร์ชาวสวิสชื่อ Gabriel Cramer ได้เสนอบทความที่นำไปสู่แนวความคิดและหลักการที่ดีเทอร์มิแนนต์อาจจะสามารถเป็นประโยชน์ต่อการวิเคราะห์ทางเรขาคณิตได้ โดยในบทความนั้น Cramer ได้นำดีเทอร์มิแนนต์มาสร้างสมการเส้นโค้งในระนาบ x-y และได้สร้างกฎที่เป็นที่รู้จักกันดี (กฎของ Cramer ที่รู้จักกันดีในการศึกษาคณิตศาสตร์ทางเมทริกซ์) ซึ่งใช้แก้ระบบในมิติ nxn โดยวิธีทางดีเทอร์มิแนนต์ ต่อมาในปี ค.ศ. 1812 Augustin-Louis Cauchy ได้ตีพิมพ์บทความที่เสนอสูตรของดีเทอร์มิแนนต์ที่ใช้หาปริมาตรของรูปทรงที่มีหลายด้าน และในปัจจุบันดีเทอร์มิแนนต์ก็ยังมีบทบาทสำคัญต่องานวิจัยหลายๆด้าน อาทิเช่นงานด้านการประมวลผลภาพดิจิทัล ซึ่งดังที่กล่าวไปแล้วว่าภาพดิจิทัลประกอบด้วยอาร์เรย์สองมิติ หรืออาจเรียกได้ว่าภาพดิจิทัลภาพหนึ่งนั้นแสดงอยู่ในรูปของเมทริกซ์ เพราะฉะนั้นทฤษฎีดีเทอร์มิแนนต์จึงเป็นส่วนหนึ่งในการวิเคราะห์และประมวลผลภาพ ในปี 2004 Ibrahim El Rube', Maher Ahmed และ Mohamed Kamel [17] ได้เสนองานวิจัยด้านการจับคู่รูปร่างในภาพ โดยทำการประมวลผลภาพใน wavelet domain

เพื่อให้มีความสามารถทนต่อสัญญาณรบกวนและเพื่อสามารถอธิบายรูปร่างให้อยู่ในหลายระดับ โดยมีหลักการที่ว่าๆดังที่ได้อธิบายไปแล้วในหัวข้อที่ 2.3.1.5 ซึ่งดีเทอร์มิแนนท์ก็เป็นหนึ่งใน ทฤษฎีที่ Ibrahim El Rube' เลือกใช้กับสัมประสิทธิ์แบบละเอียด (detail coefficient) ในการดึงเอา ลักษณะเฉพาะของแต่ละรูปร่างออกมาเพื่อนำมาเป็นตัวแทนของแต่ละรูปร่างที่เรียกว่า “ตัวแทนทาง พื้นที่สามเหลี่ยมแบบ 1 มิติ (1-D triangle area representation)” ซึ่งมีคุณสมบัติที่ไม่เปลี่ยนแปลงต่อการแปลงแบบแอฟฟายน์ โดยคุณสมบัติที่ไม่เปลี่ยนแปลงนี้ได้มาจากทฤษฎีดีเทอร์มิแนนท์ จาก งานวิจัยของ Ibrahim El Rube' นี้ทำให้เกิดแนวความคิดที่จะพัฒนาและประยุกต์ใช้หลักการที่ใช้ ทฤษฎีดีเทอร์มิแนนท์ทาง wavelet domain มาใช้ในทาง Spatial domain แทนเพื่อให้ง่ายต่อการ คำนวณ ในงานวิจัยนี้จึงได้ทำการศึกษาการใช้คุณสมบัติของดีเทอร์มิแนนท์ที่ไม่เปลี่ยนแปลงต่อการแปลงแบบแอฟฟายน์ใน Spatial domain ซึ่งต่อไปจะได้กล่าวถึงทฤษฎีบทของดีเทอร์มิแนนท์ที่ นำมาใช้หาลักษณะเฉพาะของรูปร่างที่ไม่ผันแปรต่อการแปลงแบบแอฟฟายน์

David C. Lay ได้ตีพิมพ์หนังสือ “Linear Algebra and Its Applications” [35] โดยได้อธิบาย ถึงหลักการของคณิตศาสตร์ทางเมทริกซ์ รวมทั้งกล่าวถึงการนำเมทริกซ์ไปประยุกต์ใช้ ทฤษฎีดี เทอร์มิแนนท์ที่ David C. Lay ได้เรียบเรียงไว้มีทฤษฎีที่เกี่ยวข้องดังนี้

$$A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

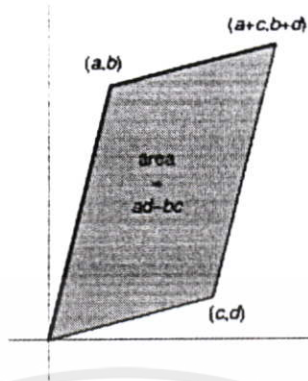
ดีเทอร์มิแนนท์ของเมทริกซ์ A เขียนแทนด้วย $|A|$ หรือ $\det(A)$ หรือเขียนเป็นสัญลักษณ์ ได้ดังนี้

$$|A| = \begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix}$$

3.3.1 การหาดีเทอร์มิแนนท์ของเมทริกซ์ที่มีมิติ 2x2

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

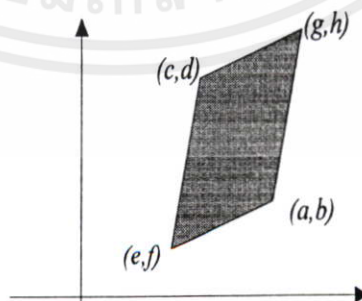
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น เอกสารนี้มีลิขสิทธิ์และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้ ของเมทริกซ์ในมิติ 2x2 นั้นความหมายในทางเรขาคณิตแล้วก็คือการหาพื้นที่ของสี่เหลี่ยมด้านขนานนั่นเอง ดังแสดงในตัวอย่าง



รูปที่ 3.2 รูปสี่เหลี่ยมด้านขนาน

สมมติว่ามีรูปสี่เหลี่ยมด้านขนานหนึ่งที่มีจุดยอดมุมทั้งสี่อยู่ที่จุด $(0,0)$, (a,b) , $(a+c, b+d)$, และ (c,d) ดังแสดงในรูปที่ 3.2 พื้นที่ของรูปสี่เหลี่ยมด้านขนานนี้หาได้จากดีเทอร์มิแนนต์ของเมทริกซ์ที่สร้างมาจากเวกเตอร์ที่แทนด้านของสี่เหลี่ยมด้านขนาน (เมทริกซ์ $A_{2 \times 2}$) ในกรณีที่จุดยอดมุมใดมุมหนึ่งของสี่เหลี่ยมด้านขนานไม่ได้อยู่ที่จุดกำเนิด (origin) ดังแสดงในรูปที่ 3.3 เราสามารถหาพื้นที่ของสี่เหลี่ยมด้านขนานได้โดยการหาดีเทอร์มิแนนต์ของเมทริกซ์ที่สร้างมาจากจุดยอดมุม 3 จุดใดๆของสี่เหลี่ยมด้านขนานดังแสดงในสมการที่ 3.3

$$area = \begin{vmatrix} a & b & 1 \\ c & d & 1 \\ e & f & 1 \end{vmatrix} \tag{3.3}$$



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

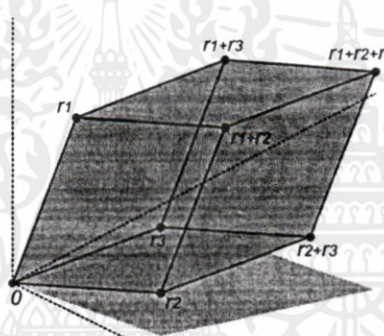
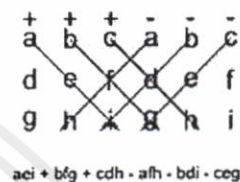
รูปที่ 3.3 รูปสี่เหลี่ยมด้านขนานที่จุดยอดมุมไม่ได้อยู่ที่จุด origin

3.3.2 การหาดีเทอร์มิแนนต์ของเมทริกซ์ที่มีมิติ 3x3

$$A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

เมทริกซ์ที่มีมิติ 3x3 A มีค่าดีเทอร์มิแนนต์คือ

$$\begin{aligned} \det(A) &= a \begin{vmatrix} e & f \\ h & i \end{vmatrix} - b \begin{vmatrix} d & f \\ g & i \end{vmatrix} + c \begin{vmatrix} d & e \\ g & h \end{vmatrix} \\ &= aei - afh - bdi + cdh + bfg - ceg \\ &= (aei + bfg + cdh) - (gec + hfa + idb) \end{aligned}$$



รูปที่ 3.4 ปริซึมหกด้านที่มีแต่ละด้านเป็นรูปสี่เหลี่ยมด้านขนาน

ซึ่งการหาดีเทอร์มิแนนต์ของเมทริกซ์ที่มีมิติ 3x3 ในที่นี้ก็คือการหาปริมาตรของปริซึมดังแสดงในรูปที่ 3.4 โดยปริมาตรของปริซึมนี้ได้มาจากการหาดีเทอร์มิแนนต์ของเมทริกซ์ที่สร้างมาจากเวกเตอร์ r_1, r_2 และ r_3 ซึ่งแต่ละเวกเตอร์มีพิกัดคือ $r_1=(a,b,c), r_2=(d,e,f)$ และ $r_3=(g,h,i)$

3.3.3 การหาดีเทอร์มิแนนต์ของเมทริกซ์ที่มีมิติ nxn

สูตรทั่วไปในการหาดีเทอร์มิแนนต์ของเมทริกซ์ที่มีมิติ nxn คือ

$$\det(A) = \sum_{j=1}^n A_{i,j} C_{i,j} = \sum_{j=1}^n A_{i,j} (-1)^{i+j} M_{i,j} \tag{3.4}$$

เอกสารนี้เป็นเอกสารที่... ซึ่ง $C_{i,j}$ คือ Cofactor มีค่าเท่ากับ $(-1)^{i+j} M_{i,j}$ คือ minor หาได้จากดีเทอร์มิแนนต์ของ... เมทริกซ์ A ที่ได้ทำการตัดแถวที่ i-th และหลักที่ j-th ทิ้ง ดังแสดงในตัวอย่างต่อไปนี้

$$A = \begin{bmatrix} -2 & 2 & -3 \\ -1 & 1 & 3 \\ 2 & 0 & -1 \end{bmatrix}$$

สมมติว่ามีเมทริกซ์ที่มีมิติ 3×3 A ให้หาดีเทอร์มิแนนต์ของเมทริกซ์นี้
ก่อนอื่นเราจะลองหาดีเทอร์มิแนนต์ตามวิธีที่ได้กล่าวไว้ในหัวข้อ 3.3.2 ได้ค่าดีเทอร์มิแนนต์ดังนี้

$$\begin{aligned} \det(A) &= (-2 \cdot 1 \cdot -1) + (-3 \cdot -1 \cdot 0) + (2 \cdot 3 \cdot 2) - (-3 \cdot 1 \cdot 2) - (-2 \cdot 3 \cdot 0) - (2 \cdot -1 \cdot -1) \\ &= 2 + 0 + 12 - (-6) - 0 - 2 = 18 \end{aligned}$$

และดีเทอร์มิแนนต์ที่ได้มาจากวิธีในหัวข้อที่ 3.3.3 คือ

$$\begin{aligned} \det(A) &= (-1)^{1+2} \cdot 2 \cdot \det \begin{bmatrix} -1 & 3 \\ 2 & -1 \end{bmatrix} + (-1)^{2+2} \cdot 1 \cdot \det \begin{bmatrix} -2 & -3 \\ 2 & -1 \end{bmatrix} \\ &= (-2) \cdot ((-1) \cdot (-1) - 2 \cdot 3) + 1 \cdot ((-2) \cdot (-1) - 2 \cdot (-3)) \\ &= (-2)(-5) + 8 = 18 \end{aligned}$$

3.3.4 คุณสมบัติของดีเทอร์มิแนนต์

- $\det(AB) = \det(A)\det(B)$ สำหรับเมทริกซ์ A และ B ที่มีมิติ $n \times n$
- $\det(A^{-1}) = \det(A)^{-1}$
- $\det(A^T) = \det(A)$
- การสลับหลักในเมทริกซ์ A ดีเทอร์มิแนนต์ของเมทริกซ์ที่ได้ทำการสลับหลักนี้มีค่าเท่ากับดีเทอร์มิแนนต์ของเมทริกซ์เดิมคูณด้วย -1
- การสลับแถวในเมทริกซ์ A ดีเทอร์มิแนนต์ของเมทริกซ์ที่ได้ทำการสลับแถวนี้มีค่าเท่ากับดีเทอร์มิแนนต์ของเมทริกซ์เดิมคูณด้วย -1
- การคูณแถวใดแถวหนึ่งของเมทริกซ์ A ด้วยจำนวนจริง r ดีเทอร์มิแนนต์ของเมทริกซ์ใหม่มีค่าเท่ากับ $r \cdot |A|$
- การคูณหลักใดหลักหนึ่งของเมทริกซ์ A ด้วยจำนวนจริง r ดีเทอร์มิแนนต์ของเมทริกซ์ใหม่มีค่าเท่ากับ $r \cdot |A|$
- การนำเอาแถวใดแถวหนึ่งในเมทริกซ์ที่ถูกคูณด้วยค่าคงที่ไปรวมกับอีกหนึ่งแถวอื่น ดีเทอร์มิแนนต์ของเมทริกซ์ใหม่มีค่าเท่ากับดีเทอร์มิแนนต์ของเมทริกซ์เดิม
- การนำเอาหลักใดหลักหนึ่งในเมทริกซ์ที่ถูกคูณด้วยค่าคงที่ไปรวมกับหลักอื่น ดีเทอร์มิแนนต์ของเมทริกซ์ใหม่มีค่าเท่ากับดีเทอร์มิแนนต์ของเมทริกซ์เดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับอาจารย์และบุคลากรในมหาวิทยาลัยเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix} = 1$$

- ถ้า A คือเมทริกซ์สามเหลี่ยม (Triangular matrix) หรือเมทริกซ์ที่มีองค์ประกอบคือ

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ 0 & a_{2,2} & \dots & a_{2,n} \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & 0 & a_{(n,n)} \end{bmatrix} \quad \text{หรือ} \quad \begin{bmatrix} a_{1,1} & 0 & 0 & 0 \\ a_{2,1} & a_{2,2} & 0 & 0 \\ \vdots & \vdots & \ddots & 0 \\ a_{n,1} & a_{n,2} & \dots & a_{(n,n)} \end{bmatrix}$$

$$\det(A) = a_{1,1} a_{2,2} \dots a_{(n,n)}$$

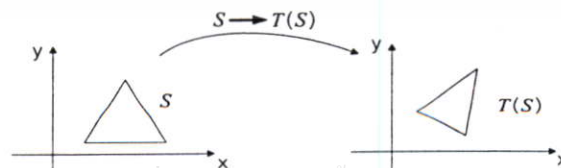
- ถ้า A คือเมทริกซ์เส้นทแยงมุม (Diagonal matrix) หรือเมทริกซ์ที่มีองค์ประกอบคือ

$$\begin{bmatrix} a_{1,1} & 0 & 0 & 0 \\ 0 & a_{2,2} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & a_{(n,n)} \end{bmatrix}$$

$$\det(A) = a_{1,1} a_{2,2} \dots a_{(n,n)}$$

3.4 การประยุกต์ใช้ทฤษฎีเทอร์มิแนนต์ในการหาลักษณะเฉพาะของรูปร่างที่ไม่ผันแปรต่อการแปลงแบบแอฟไฟน์

สมมติว่ารูปสามเหลี่ยม S รูปหนึ่งถูกแปลงเป็นรูปสามเหลี่ยม $T(S)$ ด้วยการแปลงแบบเชิงเส้น (หรือในที่นี้คือการแปลงแบบแอฟไฟน์) ดังแสดงในรูปที่ 3.5 ซึ่งมีเมทริกซ์ของการแปลงเชิงเส้นคือ A สมการของการแปลงรูปสามเหลี่ยมนี้แสดงดังสมการที่ 3.5



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งรูปที่ 3.5 การแปลงสามเหลี่ยม S ให้เป็นสามเหลี่ยม $T(S)$ ด้วยสมการเชิงเส้น

$$T(S) = AS \quad (3.5)$$

จากนั้นทำการหาดีเทอร์มิแนนต์

$$|T(S)| = |AS|$$

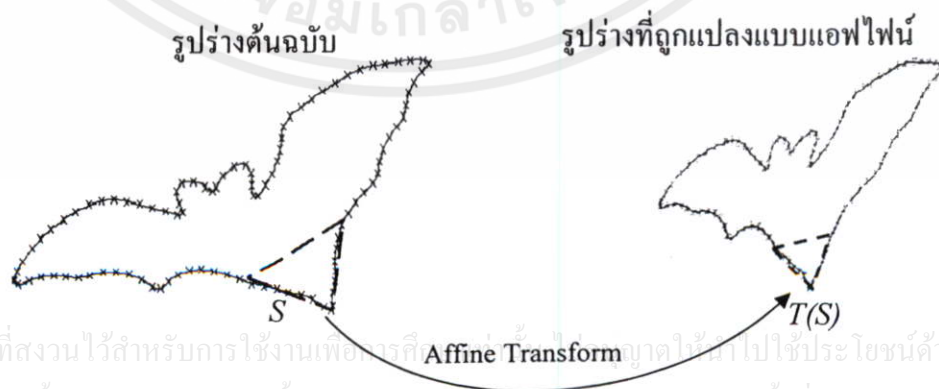
$$|T(S)| = |A| \cdot |S|$$

จากทฤษฎีดีเทอร์มิแนนต์ที่ได้กล่าวไว้ในหัวข้อที่ 3.3 การหาดีเทอร์มิแนนต์ในกรณีนี้คือการหาพื้นที่ เพราะฉะนั้นจากรูป พื้นที่ของสามเหลี่ยมที่ถูกแปลงนั้นจะเท่ากับพื้นที่ของสามเหลี่ยมเดิมคูณด้วยค่าคงที่ค่าหนึ่ง ซึ่งค่าคงที่นี้คือ ดีเทอร์มิแนนต์ของเมทริกซ์ของการแปลงเชิงเส้น A ความสัมพันธ์ของพื้นที่สามเหลี่ยมนี้แสดงดังสมการที่ 3.6

$$\{\text{area of } T(S)\} = |\det A| \cdot \{\text{area of } S\} \quad (3.6)$$

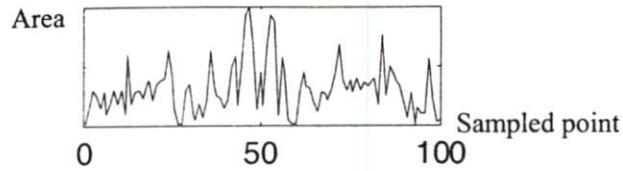
ความสัมพันธ์ของพื้นที่สามเหลี่ยมนี้เป็นหลักการพื้นฐานที่ใช้อธิบายแนวคิดในการหาคุณลักษณะเฉพาะของรูปร่างที่ถูกแปลงแบบแอฟฟายน์ โดยคุณลักษณะเฉพาะนี้มีคุณสมบัติไม่เปลี่ยนแปลงต่อการแปลงแบบแอฟฟายน์

ในงานวิจัยนี้พื้นที่สามเหลี่ยมก็คือ สามเหลี่ยมที่มีจุดยอดมุมทั้งสามอยู่บนเส้นรอบรูปของรูปร่างที่พิจารณา ดังแสดงในรูปที่ 3.6 จุดมุมของสามเหลี่ยมคือบรรดาจุดที่ได้ทำการสุ่มจากเส้นรอบรูปออกมา 100 จุด แล้วทำการหาพื้นที่สามเหลี่ยมดังกล่าวที่ทุกๆจุดแซมปลิง ก็จะได้คุณลักษณะเฉพาะของรูปร่างนั้นออกมา โดยนำพื้นที่สามเหลี่ยมมาวาดกราฟ เรียงตามจุดบนเส้นรอบรูป เรียกกราฟนี้ว่า “1-D triangle area representation” ดังแสดงตัวอย่างในรูปที่ 3.7 ซึ่งกราฟนี้จะแทนลักษณะเฉพาะของรูปร่างหนึ่งๆ สำหรับขั้นตอนในการหา 1-D triangle area representation โดยละเอียดนั้นจะกล่าวในหัวข้อที่ 3.4.1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.6 การพิจารณาสามเหลี่ยมบนเส้นรอบรูปของรูปร่างต้นฉบับกับรูปร่างที่ถูกแปลงแบบแอฟฟายน์



รูปที่ 3.7 ตัวอย่าง 1-D triangle area representation ของรูปร่างต้นฉบับ

3.4.1 ขั้นตอนการหา 1-D triangle area representation

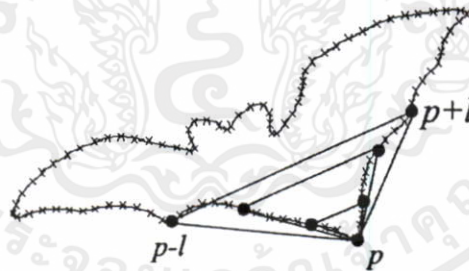
พิจารณาจุดแซมปลิงใดๆที่จุด p^h และจุดแซมปลิงสองจุดใดๆที่อยู่ห่างจากจุด p ไป l จุด เพื่อให้ได้จุดมุมของสามเหลี่ยมทั้งสามจุด ดังแสดงในรูปที่ 3.8 ตัวแปร l ในที่นี้แทนความยาวแขนของสามเหลี่ยม (Triangle side length-TSL) โดยทำการปรับค่าแขนของสามเหลี่ยม l จาก 1-49

หลังจากที่ได้จุดสามจุดที่ค่าความยาว l ใดๆ ก็ทำการคำนวณพื้นที่สามเหลี่ยมตามสมการที่

3.7

$$A_l(p) = \frac{1}{2} \begin{vmatrix} x_{k_{p-l}} & y_{k_{p-l}} & 1 \\ x_k & y_k & 1 \\ x_{k_{p+l}} & y_{k_{p+l}} & 1 \end{vmatrix} \quad (3.7)$$

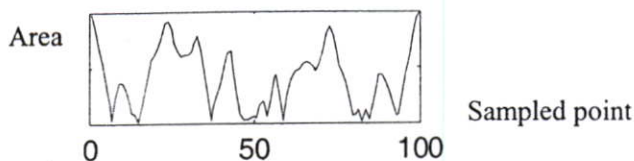
โดย x และ y คือพิกัดของจุดแซมปลิง



รูปที่ 3.8 จุดแซมปลิง 3 จุดใดๆที่ TSL ต่างๆบนเส้นรอบรูปที่นำไปคำนวณพื้นที่สามเหลี่ยม

เมื่อทำการคำนวณพื้นที่สามเหลี่ยมไปทุกจุดบนเส้นรอบรูปที่สุ่มมา แล้วทำการนอร์มอไลซ์จากนั้นนำค่าพื้นที่นั้นไปวาดกราฟ จะได้ 1-D triangle area representation ดังแสดงตัวอย่างในรูปที่ 3.9 โดยแกน x คือตำแหน่งของจุดสุ่มบนเส้นรอบรูป แกน y คือค่าพื้นที่สามเหลี่ยม

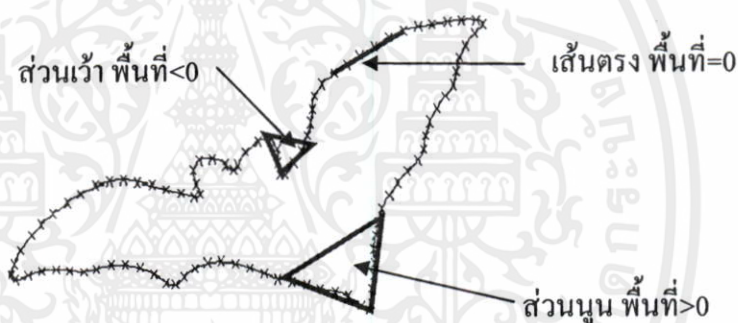
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 ตัวอย่าง 1-D triangle area representation จำนวนที่ $L=20$

3.4.2 ชนิดของ 1-D triangle area representation ที่ใช้ในงานวิจัย

จากสูตรการหาพื้นที่สามเหลี่ยมในสมการที่ 3.7 ในที่นี้ก็คือการหาดีเทอร์มิแนนต์ของเมทริกซ์ที่สร้างมาจากจุดแซมปลิงสามจุด ซึ่งโดยทั่วไปค่าดีเทอร์มิแนนต์ที่คำนวณได้จะมีทั้งค่าบวกและค่าลบ ซึ่งค่าดีเทอร์มิแนนต์หรือพื้นที่ที่มีค่าบวกหรือค่าลบนี้จะแสดงถึงจุดบนเส้นรอบรูปที่พิจารณาอยู่นั้นว่าเป็นส่วนเว้าหรือส่วนนูน โดยถ้าเราพิจารณาทิศวนเข็มนาฬิกา พื้นที่ที่มีค่าบวกแสดงว่าจุดบนเส้นรอบรูปที่พิจารณาอยู่นั้นเป็นส่วนนูน ในทางกลับกัน พื้นที่ที่มีค่าลบแสดงว่าจุดบนเส้นรอบรูปที่พิจารณาอยู่นั้นเป็นส่วนเว้า ดังแสดงในรูปที่ 3.10

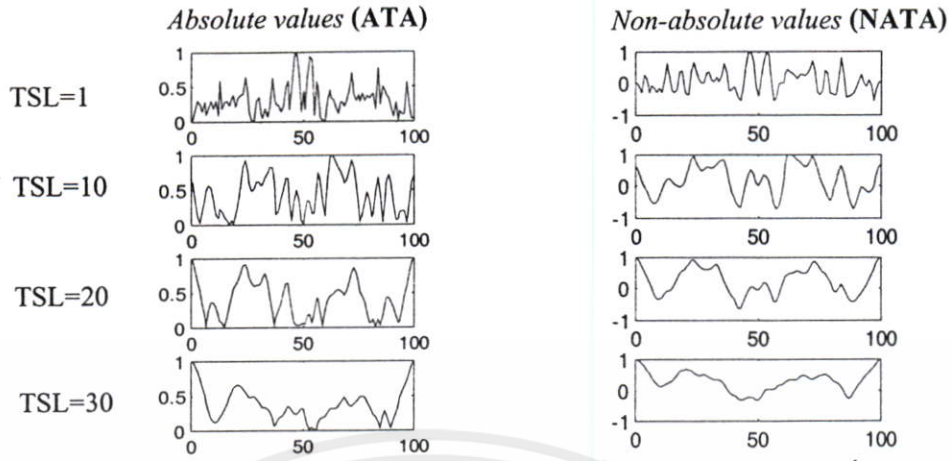


รูปที่ 3.10 แสดงถึงส่วนเว้าส่วนนูนของเส้นรอบรูปที่สัมพันธ์กับค่าดีเทอร์มิแนนต์หรือค่าพื้นที่

จากกรณีนี้เราได้ทำการแบ่งชนิดของ 1-D triangle area representation ออกเป็น 2 ชนิดคือ

1. Absolute 1-D triangle area representation (ATA) ซึ่งได้คำนวณมาจากสูตรที่ 3.7 แล้วทำการ absolute
2. Non- absolute 1-D triangle area representation (NATA) ซึ่งได้คำนวณมาจากสูตรที่ 3.7 โดยไม่ได้ทำการ absolute

ทั้งนี้ก็เพื่อศึกษาว่า 1-D triangle area representation แบบไหนและที่ความยาวแขนของสามเหลี่ยมที่ขนาด L เท่าไรจะมีประสิทธิภาพในการแยกแยะกลุ่มข้อมูลได้ดี ซึ่งประสิทธิภาพในการแยกแยะข้อมูลนั้นจะส่งผลต่อประสิทธิภาพในการรู้จำ โดยเราได้ทำการทดลองและเปรียบเทียบผลการทำงานที่ความยาวแขนสามเหลี่ยม L ใดๆ



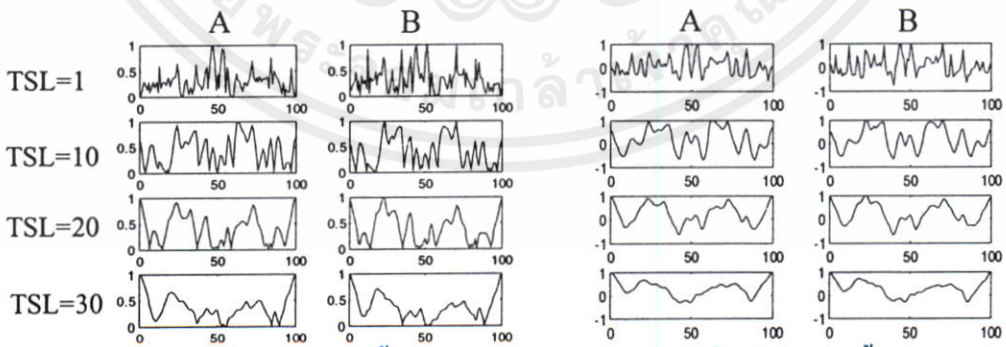
รูปที่ 3.11 แสดงตัวอย่างของ ATA และ NATA ที่หามาจากเซนสามเหลี่ยม / ใดๆ

3.4.3 ตัวอย่างลักษณะของ 1-D triangle area representation ที่ไม่ผันแปรต่อการแปลงแบบแอฟไฟน์

ในหัวข้อนี้จะแสดงตัวอย่างคุณลักษณะของ 1-D triangle area representation ที่ไม่ผันแปรต่อการแปลงแบบแอฟไฟน์ โดยเราจะแสดงตัวอย่าง 1-D triangle area representation ที่ความยาวเซน / ต่างๆของสามเหลี่ยม ของรูปร่าง 2 รูปร่างที่ถูกแปลงแบบแอฟไฟน์ ดังแสดงในรูปที่ 3.12 จะเห็นว่าถึงแม้ว่ารูปร่างทั้งสองจะได้มาจากคนละมุมมอง แต่ลักษณะของ 1-D triangle area representation ยังคงมีลักษณะคล้ายกัน



ก) รูปร่างที่ได้มาจากคนละมุมมอง



ข) ATA ที่ / ต่างๆของทั้งสองรูปร่าง

ค) NATA ที่ / ต่างๆของทั้งสองรูปร่าง

รูปที่ 3.12 ตัวอย่างลักษณะของ 1-D triangle area representation ที่ไม่ผันแปรต่อการแปลงแบบแอฟไฟน์

เอกสารนี้เป็นทรัพย์สินทางปัญญาของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่สามารถนำออกจำหน่ายหรือทำซ้ำโดยไม่ได้รับอนุญาตจากสถาบันฯ หากมีข้อสงสัยหรือต้องการข้อมูลเพิ่มเติม กรุณาติดต่อฝ่ายบริการลูกค้า โทร. 02-111-1111 หรือ e-mail: service@kmitl.ac.th

สำหรับในงานวิจัยนี้เราจะไม่นำรูปร่างที่ถูกแปลงในลักษณะสะท้อนกลับ (Reflection หรือ Mirror) มาพิจารณา ซึ่งต่อไปจะได้ทำการอธิบายถึงความหมายของรูปร่างที่ถูก reflect ในเชิงคณิตศาสตร์ก่อนที่จะกล่าวถึงสาเหตุที่ไม่นำมาพิจารณา

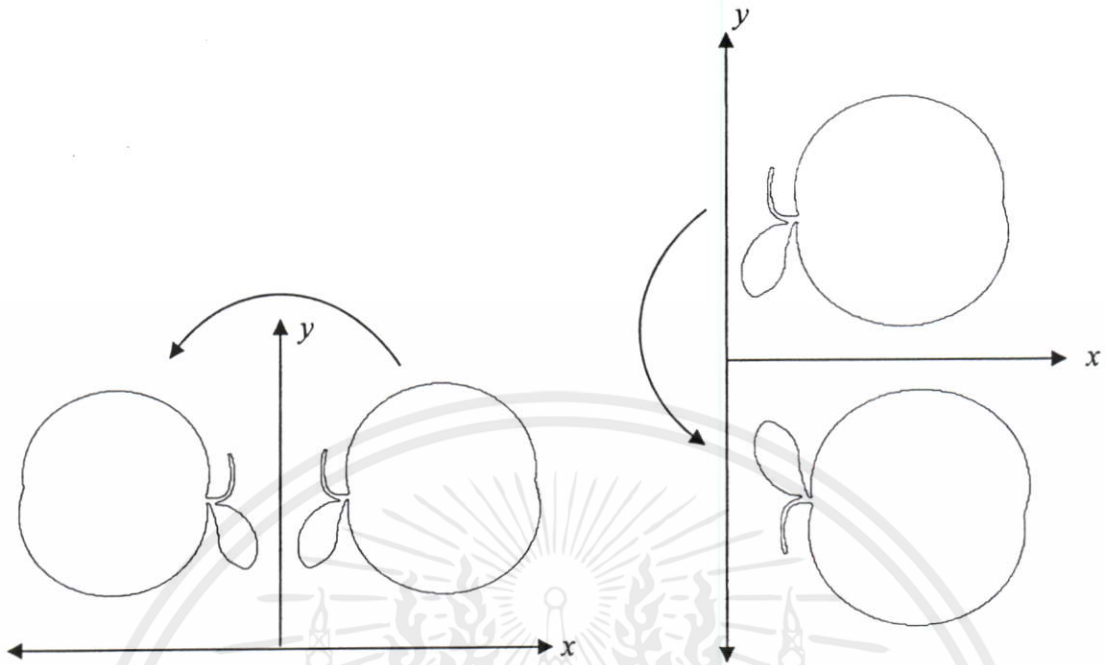
การแปลงแบบ Reflection หรือ mirror นั้นเป็นส่วนหนึ่งของการแปลงแบบแอฟไฟน์ในด้านการเปลี่ยนแปลงขนาด ดังแสดงในสมการที่ 3.8

$$T \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (3.8)$$

โดยที่ s_x และ s_y คือตัวแปรของการเปลี่ยนแปลงขนาด (Scaling) ถ้าค่าของ s_x และ s_y มากกว่าหนึ่งแสดงถึงการขยายขนาดในทิศทางที่สอดคล้องกัน ถ้ามีค่าอยู่ในช่วง 0 ถึง 1 แสดงถึงการย่อขนาดในทิศทางที่สอดคล้องกัน ในกรณีที่ค่าของ s_x และ s_y น้อยกว่าศูนย์ จะเกิดลักษณะที่เรียกว่า Reflection หรือ Mirror คือรูปกลับขวาเป็นซ้าย ซ้ายเป็นขวา ดังแสดงในรูปที่ 3.13

สำหรับสาเหตุที่ไม่นำรูปร่างที่ถูกแปลงแบบ Reflection มาพิจารณานั้น เนื่องจากจะเกิดปัญหาเกี่ยวกับการหา Shape representation แบบ NATA โดยจะทำให้ NATA มีลักษณะกลับค่าจากบวกลบ เป็นลบ จากลบเป็นบวก ซึ่งปัญหาดังกล่าวเกิดขึ้นในขั้นตอนการหาเส้นรอบรูปที่อาศัยหลักการ Contour tracking ไปตามขอบภาพในทิศทางที่กำหนด สำหรับในงานวิจัยนี้เราจะพิจารณาทิศทางทวนเข็มนาฬิกา ด้วยเหตุนี้จะส่งผลให้คู่อันดับ (x,y) ของรูปร่างเดิมกับรูปร่างที่ถูกแปลงแบบ Reflection สลับกลับด้านกันดังแสดงในรูป 3.14 เพราะฉะนั้นแล้วคู่อันดับ (x,y) ที่จะนำไปคำนวณพื้นที่สามเหลี่ยมตามสมการที่ 3.7 นั้นสลับตำแหน่งกันดังแสดงในสมการที่ 3.9 ซึ่งการสลับตำแหน่งแถวในเมทริกซ์นั้นจะทำให้ค่าดีเทอร์มิแนนท์ (พื้นที่สามเหลี่ยม) กลับค่าจากบวกลบ เป็นลบ หรือจากลบเป็นบวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



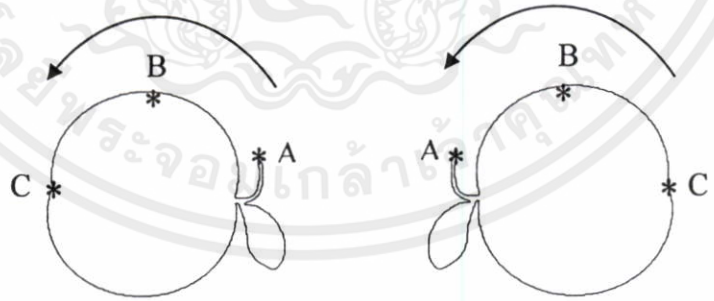
$$T \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{bmatrix} -s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

ก) $s_x < 0$

$$T \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & -s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

ข) $s_y < 0$

รูปที่ 3.13 รูปร่างที่ถูกแปลงแบบ Reflection หรือ Mirror



รูปที่ 3.14 การสลับตำแหน่งคู่อันดับ (x,y) จาก $A \rightarrow B \rightarrow C$ ของรูปร่างเดิม กลายเป็น $C \rightarrow B \rightarrow A$ ในรูปร่างที่ถูกแปลงแบบ Reflection

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์อื่นใด (3.9)
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกข้อมูลนี้ไปหา และต้องอ้างอิงที่มาของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

วิธีการรู้จำภาพของวัตถุ

การรู้จำภาพของวัตถุถือว่ามีความสำคัญต่อประสิทธิภาพหรือความฉลาดของหุ่นยนต์ หรืองานใดๆที่ต้องอาศัยระบบการประมวลผลภาพดิจิทัล ด้วยเหตุนี้การรู้จำวัตถุโดยใช้การประมวลผลภาพ (Image Processing) จึงเป็นหัวข้อที่ได้รับความนิยมในการศึกษาและวิจัยมาเป็นเวลานาน ทำให้เกิดวิธีการในการรู้จำภาพของวัตถุออกมามากมาย แต่ละวิธีก็อาศัยความรู้และทฤษฎีที่แตกต่างกันไป สำหรับในบทนี้ได้รวบรวมวิธีการรู้จำภาพของวัตถุที่นักวิจัยส่วนใหญ่ใช้กันและอธิบายไว้อย่างคร่าวๆในหัวข้อที่ 4.1 ส่วนวิธีการรู้จำภาพของวัตถุที่ได้ทำการศึกษาในงานวิจัยนี้คือ นอร์มอลไลซ์ครอสคอรีเลชัน (Normalized cross-correlation-NCC) และโครงข่ายประสาทเทียม (Artificial Neural Network-ANN) ซึ่งได้อธิบายไว้อย่างละเอียดในหัวข้อที่ 4.2 และ 4.3 ตามลำดับ

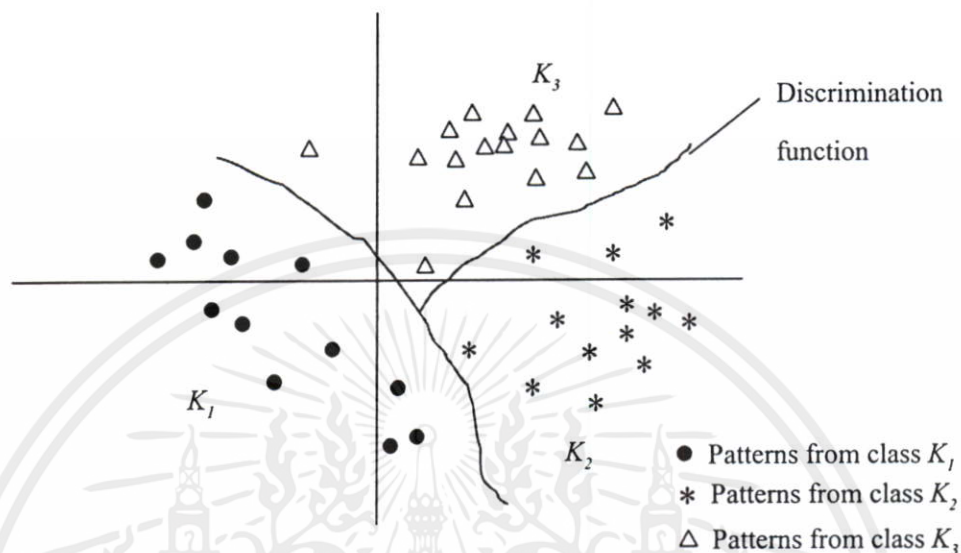
4.1 บทนำ

โดยทั่วไปแล้วการรู้จำภาพของวัตถุถือว่าเป็นส่วนหนึ่งของการรู้จำรูปแบบ (Pattern recognition) ซึ่งใช้ภาพของวัตถุเป็นปัจจัยหลักในการรู้จำ สิ่งที่น่าสนใจไม่ได้เลยสำหรับการรู้จำภาพของวัตถุในทุกงานวิจัยก็คือการดึงเอาลักษณะเฉพาะออกมาจากแต่ละรูปร่างแล้วนำเอาคุณลักษณะเฉพาะนั้นมาใช้ในขั้นตอนกระบวนการรู้จำ นักวิจัยแต่ละคนต่างก็มีวิธีการในการหาตัวอธิบายหรือตัวแทนรูปร่างเพื่อนำมาใช้เป็นคุณลักษณะเฉพาะของรูปร่างที่แตกต่างกันออกไปดังที่ได้ยกตัวอย่างไว้แล้วในบทที่ 2 ส่วนวิธีการในการรู้จำสามารถแบ่งอย่างคร่าวๆออกเป็น 5 วิธีใหญ่ๆ คือ Statistical pattern recognition, Neural network, Syntactic pattern recognition, Recognition as graph matching และ Fuzzy systems แต่ละวิธีมีหลักการ โดยรวมดังนี้

4.1.1 การรู้จำรูปแบบด้วยหลักการทางสถิติ (Statistical pattern recognition)

เกิดจากการนำเอากลุ่มข้อมูลหรือในที่นี้ก็คือตัวอธิบายรูปร่างมาแบ่งออกเป็นกลุ่มๆ ตามจำนวนชนิดของข้อมูลซึ่งเราได้กำหนดไว้แล้ว เพื่อที่จะสร้างกลุ่มข้อมูลที่ใช้ในการเรียนรู้ให้กับระบบ จากนั้นก็ทำการสอนระบบด้วยการสร้างรูปแบบทางสถิติ อาจจะเป็นการหาการกระจายของข้อมูล หรือตัวแปรทางสถิติอื่นไม่ว่าจะเป็นค่าเฉลี่ย ค่าความแปรปรวน เพื่อนำมาสร้างเป็น Discriminant function ดังแสดงตัวอย่างในรูปที่ 4.1 เมื่อต้องการทดสอบรูปร่างว่าอยู่ในกลุ่มไหนก็ทำการแทนค่าคุณลักษณะเฉพาะลงไปใน Discriminant function ถ้าอยู่ใกล้กับกลุ่มไหนมากที่สุด หรือระยะทางในระบบต่างๆ เช่น Euclidian distance ที่สั้นที่สุด ก็สามารถบอกได้ว่ารูปร่างที่นำมาทดสอบอยู่ในกลุ่มใด ที่กล่าวมาข้างต้นเป็นแค่หลักการของ Statistical pattern recognition ทั้งนี้

วิธีการรู้จำที่มีพื้นฐานอยู่บนความรู้ทางสถิติอาจมีความซับซ้อนหรือความแตกต่างกันออกไปบ้าง ตัวอย่างวิธีการรู้จำที่อาศัยความรู้ทางสถิติ เช่น PCA (principle component analysis), Bayesian classifier, Maximum likelihood, Posteriori probability เป็นต้น



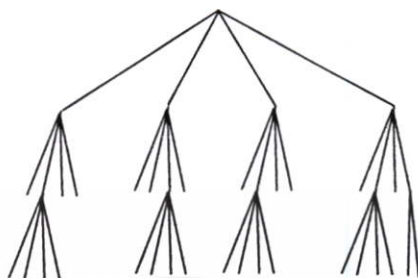
รูปที่ 4.1 General discrimination functions

4.1.2 โครงข่ายประสาทเทียม (Neural network)

โครงข่ายประสาทเทียมเป็นการจำลองการทำงานโครงข่ายประสาทของมนุษย์ ซึ่งประกอบด้วยส่วนของการประมวลผลที่เรียกว่านิวรอน (Neuron) ทุกๆ นิวรอนสามารถมีอินพุตได้หลายอินพุตแต่มีเอาต์พุตเพียงเอาต์พุตเดียว และทุกๆ เอาต์พุตจะต่อไปยังอินพุตของนิวรอนอื่นๆ ภายในโครงข่าย การติดต่อกันภายในระหว่างนิวรอนไม่ใช่ลักษณะการต่อแบบธรรมดา ทุกๆ อินพุตจะมีน้ำหนักเป็นตัวกำหนดกำลังของการติดต่อภายในและช่วยในการตัดสินใจ การทำงานของนิวรอนในบางโครงข่ายจะถูกกำหนดไว้ตายตัว แต่บางโครงข่ายสามารถที่จะปรับแต่งได้ซึ่งอาจจะเป็นการปรับแต่งจากภายนอกโครงข่ายหรือนิวรอนสามารถปรับได้ด้วยตัวเอง ซึ่งแสดงถึงความสามารถในการเรียนรู้และจดจำของโครงข่ายประสาทเทียม ซึ่งสามารถนำไปใช้งานอย่างกว้างขวาง รูปแบบการจดจำก็คือว่าเป็นการประยุกต์หนึ่งของโครงข่าย สำหรับทฤษฎีของ Artificial Neural network ที่นำมาใช้ในการจดจำรูปร่างจะกล่าวโดยละเอียดอีกทีในหัวข้อที่ 4.3

เอกสารนี้เป็นเอกสารที่ 4.1.3 การรู้จำทางไวยากรณ์ภาษา (Syntactic pattern recognition) นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น เป็นการรู้จำรูปภาพในรูปแบบคล้ายๆ กับโครงสร้างไวยากรณ์ของภาษา ซึ่งโครงสร้าง
ไวยากรณ์ในการจดจำรูปร่างเป็นรูปแบบที่เราสร้างขึ้นเอง โดยอาจจะแทนแต่ละรูปร่างด้วย
โครงสร้างไดอะแกรมดังแสดงในรูปที่ 4.2 ซึ่งรูปร่างแต่ละรูปจะเปรียบเสมือนกลุ่มคำส่วนย่อยของ

แต่ละรูปร่าง เปรียบเสมือนตัวอักษรดังได้อธิบายถึงวิธีการในการหาตัวอธิบายหรือตัวแทนรูปร่าง ด้วย syntactic analysis ในหัวข้อที่ 2.3.2.3



รูปที่ 4.2 โครงสร้างโคอะแกรมที่ใช้แทนรูปร่าง

4.1.4 การรู้จำโดยการเปรียบเทียบกราฟ (Recognition as graph matching)

วิธีการรู้จำด้วยการ matching graph คือการนำ graph มาเปรียบเทียบวัดความคล้ายคลึงกัน ซึ่ง graph ในที่นี้หมายถึงตัวอธิบายหรือตัวแทนรูปร่างที่อยู่ในลักษณะ graph หรือ signature ในส่วนของการวัดความคล้ายคลึงกันก็ทำการเปรียบเทียบ graph ที่ละคู่ แล้วใช้ฟังก์ชันมาวัดความเหมือน ทั้งนี้ก็แล้วแต่ว่านักวิจัยจะใช้ความรู้ทางคณิตศาสตร์ใดมาช่วยวัดความเหมือน อย่างเช่นในงานวิจัยนี้เราได้ใช้ นอร์มอลไลซ์ครอสคอร์รีเลชัน (Normalized cross-correlation-NCC) มาวัดความคล้ายคลึงกันของ shape representation ของแต่ละรูปร่าง ซึ่งจะได้อีกกล่าวโดยละเอียดอีกครั้งในหัวข้อที่ 4.2 สำหรับตัวอย่างฟังก์ชันหรือหลักการที่นำมาใช้ในการวัดความเหมือนในงานวิจัยอื่นๆ ก็มี อาทิเช่น ระยะทางยูคลิเดียน (Euclidian distance) ระยะทางแมนฮัตตัน (Manhattan distance) ระยะทางแบบมหาลาโนบิส (Mahalanobis distance) ไดนามิกโปรแกรม (Dynamic program) เป็นต้น

4.1.5 ระบบฟัซซี (Fuzzy systems)

ฟัซซีลอจิกคิดค้นโดย L. A. Zadeh ในปี ค.ศ. 1965 ซึ่งเป็นผลงานวิทยานิพนธ์ระดับปริญญาเอก ฟัซซีลอจิกเป็นตรรกะที่อยู่บนพื้นฐานความเป็นจริงที่ว่า ทุกสิ่งบนโลกแห่งความเป็นจริงไม่ใช่มีเฉพาะสิ่งที่มีความแน่นอนเท่านั้น แต่มีหลายสิ่งหลายเหตุการณ์ที่เกิดขึ้นอย่างไม่เที่ยงและไม่แน่นอน (uncertain) อาจเป็นสิ่งที่คลุมเครือ (fuzzy) ไม่ใช่ชัดเจน (exact) ฟัซซีลอจิกหรือตรรกะแบบคลุมเครือ เป็นคณิตศาสตร์แขนงใหม่ที่มีความสำคัญต่อเทคโนโลยีสารสนเทศเป็นอย่างมาก ฟัซซีลอจิกได้ถูกนำไปประยุกต์ใช้ในงานด้านต่าง ๆ ที่ต้องใช้คอมพิวเตอร์ในการประมวลผล

เอกสารนี้เป็นเอกสารที่สนับสนุนการตัดสินใจ การพยากรณ์ การคาดการณ์เหตุการณ์ หรือแม้แต่ในงานด้านราคา
ไม่ว่ากรณีใดก็ตามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรรกะแบบฟัซซี (fuzzy logic) เป็นเครื่องมือที่ช่วยในการตัดสินใจภายใต้ความไม่แน่นอนของข้อมูล โดยยอมให้มีความยืดหยุ่นได้ ใช้หลักเหตุผลที่คล้ายการเลียนแบบวิธีความคิดที่ซับซ้อนของมนุษย์ ฟัซซีลอจิกมีลักษณะที่พิเศษกว่าตรรกะแบบจริงเท็จ (Boolean logic) เป็นแนวคิดที่มีค่าความจริงบางส่วน (partial true) โดยค่าความจริงจะอยู่ในช่วงระหว่างจริง (completely true) กับเท็จ (completely false) ส่วนตรรกศาสตร์เดิมจะมีค่าเป็นจริงกับเท็จเท่านั้น แสดงดังรูปที่ 4.3



รูปที่ 4.3 ตรรกะแบบจริงเท็จ (บูลีนลอจิก) กับตรรกะแบบฟัซซี (ฟัซซีลอจิก)

รูปที่ 4.4 เป็นการแสดงให้เห็นว่าแนวทางในการตัดสินใจของปัญหาทั้งหมดมีเพียงส่วนน้อยที่เป็นสิ่งที่แน่นอน (certainty) ที่เหลือคือสิ่งที่ไม่แน่นอนซึ่งประกอบด้วยความไม่แน่นอนที่มีลักษณะแบบสุ่ม และความไม่แน่นอนที่มีลักษณะเป็นฟัซซี หรือคลุมเครือ ซึ่งมีมากกว่าร้อยละ 40 เพราะปัญหาส่วนมากเกี่ยวข้องกับการตัดสินใจของมนุษย์ซึ่งจะตัดสินใจตามพื้นฐานความคิดของตนเป็นหลัก

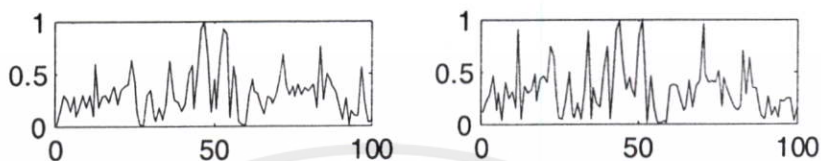


รูปที่ 4.4 ความไม่แน่นอน (uncertainty)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 นอร์มอไลซ์ครอสคอร์เรชัน (Normalized cross-correlation-NCC)

ในงานวิจัยนี้เราได้ใช้ Normalized cross-correlation ซึ่งถือว่าเป็นฟังก์ชันที่ใช้วัดความคล้ายคลึงกันระหว่าง feature vector 2 vectors ดังแสดงอยู่ในรูปแบบกราฟ ในรูปที่ 4.5 ฟังก์ชันของ Normalized cross-correlation แสดงดังสมการที่ 4.1



รูปที่ 4.5 กราฟ 2 กราฟ ที่ต้องการวัดความคล้ายคลึงกัน

$$R(Q, D) = \frac{\sum_{i=1}^n q_i d_i}{\sqrt{\sum_{i=1}^n q_i^2 \sum_{i=1}^n d_i^2}} \quad (4.1)$$

โดยที่ R คือค่า Normalized cross-correlation หรือค่าความเหมือน ค่ายิ่งมากแสดงว่าเหมือนมาก Q และ D คือ feature vector ทั้งสองที่ต้องการวัดความเหมือน ซึ่ง $Q = [q_1, q_2, \dots, q_n]$ และ $D = [d_1, d_2, \dots, d_n]$ โดยที่ q_i และ d_i คือค่าองค์ประกอบภายใน feature vector Q และ D ตามลำดับ ส่วน n คือจำนวนองค์ประกอบในแต่ละ feature vector

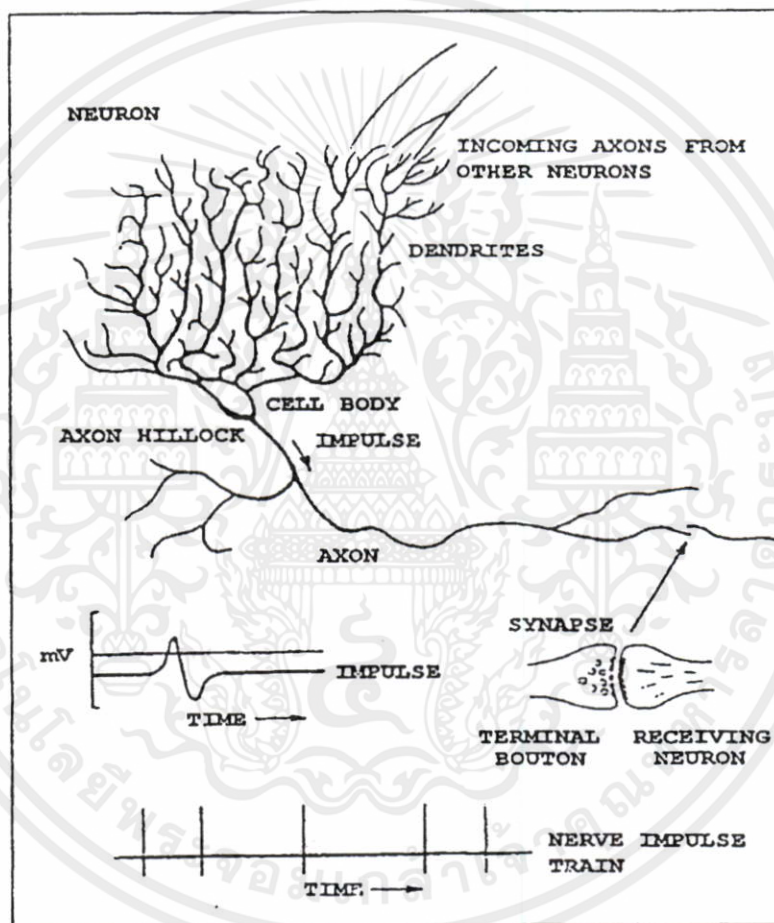
4.3 โครงข่ายประสาทเทียม (Artificial Neural Network-ANN)

โครงข่ายประสาทเทียมจำลองมาจากกระบวนการทางความคิดของมนุษย์ ซึ่งมีคุณลักษณะพิเศษหลายประการที่เป็นประโยชน์ต่อระบบการประมวลผล ดังนั้นจึงได้มีการพัฒนาและประยุกต์กระบวนการของโครงข่ายประสาทเทียมให้ใช้กับงานหลากหลายประเภท ตัวอย่างเช่น การประมวลผลสัญญาณ (Signal Processing) หรืองานเกี่ยวกับการจดจำรูปแบบต่างๆ (Pattern Recognition) เป็นต้น ซึ่งในหัวข้อนี้จะกล่าวถึงทฤษฎีและความเป็นมา รวมทั้งโครงสร้างและการเรียนรู้ของโครงข่ายประสาทชีวภาพ (Biological Neural Networks) และโครงข่ายประสาทเทียม (Artificial Neural Networks (ANNs))

4.3.1 โครงข่ายประสาทชีวภาพ (Biological Neural Networks)

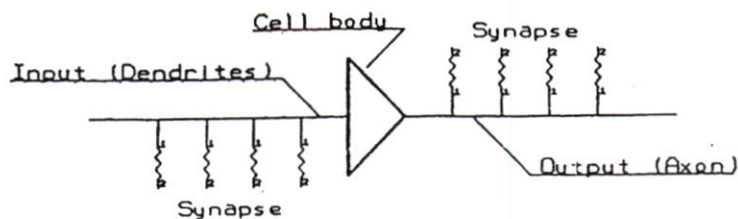
นิวรอน (Neuron) [36] มีลักษณะแบบเซลล์ชีวิตที่เป็นโครงข่ายร่างแห โดยโครงข่ายแต่ละชนิดมีความแตกต่างกันมากทั้งขนาดและรูปร่าง สมองของมนุษย์มีเซลล์นิวรอนเป็นโครงสร้างประมาณหนึ่งหมื่นล้านหรือหนึ่งแสนล้านเซลล์ แต่ละนิวรอนจะมีคุณลักษณะที่แตกต่างกันไป โดย

มีการทำงานคล้ายกันคือ รับเข้า ประมวลผลและส่งสัญญาณไฟฟ้าเคมีออกไปยังนิวรอนอื่น ซึ่งจะส่งสัญญาณไปตามระบบของสมอง โดยตัวเซลล์ร่างกาย (Cell Body) ประกอบด้วยนิวเคลียส (Nucleus) อยู่ตรงกลาง รอบๆ ตัวเซลล์นิวรอนจะมีสิ่งที่ยื่นออกไปเพื่อส่งสัญญาณไปยังเซลล์นิวรอนอื่นเรียกว่า เอ็กซอน (Axon) จุดรับสัญญาณจากเซลล์อื่นเข้ามายังตัวเซลล์จะผ่านจุดเชื่อมต่อเรียกว่า ไซแนปส์ (Synapse) และที่ปลายกิ่งจะแตกแขนงออกเป็นก้านย่อยๆ ต่อไปยังเซลล์อื่นๆ เพื่อรับสัญญาณเรียกว่า เดนไดรต์ (Dendrites) แสดงดังรูปที่ 4.6



รูปที่ 4.6 นิวรอนของสมองมนุษย์

โดยอาศัยวงจรอิเล็กทรอนิกส์ [37] ดังรูปที่ 4.7 เมื่อเปรียบเทียบความคล้ายคลึงของนิวรอน สมองมนุษย์กับนิวรอนอิเล็กทรอนิกส์ จะเห็นว่า เป็นวงจรขยายแบบอนาล็อกรับสัญญาณอินพุตผ่าน เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ตัวความต้านทาน แล้วส่งสัญญาณเอาท์พุตไปยังวงจรขยายตัวอื่นๆ ได้เช่นกัน ไม่ว่าจะกรณีใดๆ ทั้งสิ้น ออกกฎหมายให้คิดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 นิวรอนทางอิเล็กทรอนิกส์

นิวรอนแต่ละตัวจะมีคุณสมบัติในการเพิ่มขยายหรือลดทอนความเข้มของสัญญาณที่เข้ามาทางเดนไดรต์ของเซลล์ เนื่องจากเซลล์ประสาทหนึ่งเซลล์มีเดนไดรต์จำนวนมาก ดังนั้นสัญญาณจากเดนไดรต์ที่รับเข้ามาจากเซลล์ประสาทอื่นๆ จะถูกนำมารวมกันที่ตัวเซลล์ ซึ่งที่เซลล์ประสาทจะมีค่าขีดเริ่ม (Threshold) ค่าหนึ่ง โดยหากผลรวมของสัญญาณไฟฟ้าเคมี (Electrochemical) มีค่ามากกว่าค่าขีดเริ่ม เซลล์ประสาทก็จะส่งสัญญาณขนาดหนึ่งผ่านไปยังนิวรอนอื่นๆทางเอ็กซอน ส่วนการจัดเรียงชั้น (Layer) และลักษณะการเชื่อมโยงระหว่างนิวรอนในสมองของมนุษย์นั้นมีการจัดเรียงที่ซับซ้อน สอดคล้องกับหน้าที่การทำงานเฉพาะส่วน ซึ่งยากต่อการวิเคราะห์ออกมาได้ และในระบบนิวรอนยังมีฟังก์ชันการเรียนรู้ คือ กระบวนการของการปรับค่าน้ำหนัก (Weight) ที่อยู่ในระบบ เพื่อให้สอดคล้องกับกลุ่มตัวอย่างของชุดข้อมูลอินพุตและเอาต์พุตที่ป้อนเข้าสู่ระบบ ในขณะที่ทำการฝึกสอน กล่าวคือ น้ำหนักของจุดเชื่อมโยงจะต้องถูกปรับแต่งจนกระทั่งการทำงานของระบบ เป็นไปตามลักษณะพิเศษของกลุ่มตัวอย่างที่ทำการฝึกสอน

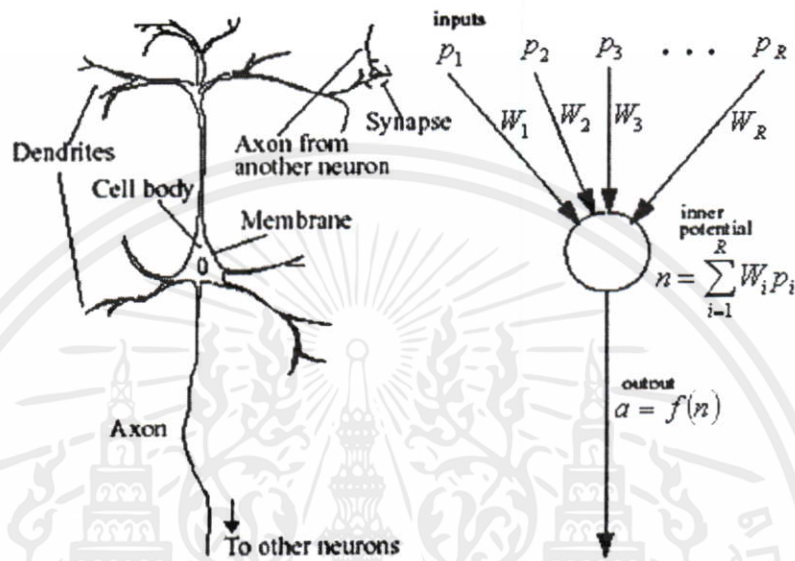
4.3.2 โครงข่ายประสาทเทียม (Artificial Neural Networks)

โครงข่ายประสาทเทียม [38] เป็นการจำลองการทำงานของสมองมนุษย์ ด้วยโปรแกรมคอมพิวเตอร์ เป็นแนวความคิดที่ต้องการให้คอมพิวเตอร์มีความชาญฉลาดในการเรียนรู้เหมือนที่มนุษย์มีการเรียนรู้ สามารถฝึกฝนได้ และสามารถนำความรู้และทักษะ ไปแก้ปัญหาต่างๆ มีนักวิจัยจำนวนมากได้คิดค้นรูปแบบโครงข่ายประสาทเทียมแบบต่างๆ เพื่อนำมาประยุกต์ใช้อย่างกว้างขวาง การประยุกต์ใช้งานโครงข่ายประสาทเทียมมีตั้งแต่การใช้เพื่อตัดสินใจง่ายๆ ไปจนถึงงานที่มีความยุ่งยากซับซ้อน ตัวอย่างการประยุกต์ใช้งานบางส่วน ได้แก่ งานด้านการควบคุม งานด้านการบิน ด้านยานยนต์ ด้านการบริหารจัดการ ด้านการธนาคาร ด้านการทหาร ด้านการบันเทิง และอื่นๆ อีกมากมาย

โครงข่ายประสาทเทียมมีประวัติความเป็นมาย้อนหลังไปประมาณ 60 กว่าปีก่อน ในปี ค.ศ.

1943 McCulloch และ Pitts แห่งมหาวิทยาลัยชิคาโก ประเทศสหรัฐอเมริกา ได้นำเสนอบทความวิชาการ "Boolean brain" ซึ่งได้กลายเป็นจุดกำเนิดของการจัดรูปแบบคณิตศาสตร์ของประสาทเทียม ต่อมาได้มีนักวิจัยได้คิดค้นรูปแบบโครงข่ายประสาทเทียมแบบต่างๆ มากมาย และทุกรูปแบบวิธีจะประกอบกับวิธีการสอนโครงข่ายด้วย ซึ่งวิธีการต่างๆ จะมีความซับซ้อนแตกต่างกันไป

โครงข่ายประสาทเทียมเป็นการจำลองการทำงานโครงข่ายประสาทของมนุษย์ ซึ่งประกอบด้วยส่วนของการประมวลผลที่เรียกว่านิวรอน (Neuron) ทุกๆ นิวรอนสามารถมีอินพุตได้หลายอินพุตแต่มีเอาต์พุตเพียงเอาต์พุตเดียว และทุกๆ เอาต์พุตจะต่อกับอินพุตของนิวรอนอื่นๆ ภายในโครงข่าย



รูปที่ 4.8 โครงข่ายประสาทชีวภาพและโครงข่ายประสาทเทียม

โครงสร้างของโครงข่ายประสาทเทียม จะมีลักษณะคล้ายคลึงกับโครงข่ายประสาทชีวภาพ ดังรูปที่ 4.8 โดยทั่วไปนั้นจะประกอบด้วยส่วนย่อยๆ ที่เรียกว่านิวรอน ซึ่งก็คือเซลล์ประสาทในทางชีวภาพ จะมีขาที่ใช้ในการรับส่งข้อมูลหลายขาขึ้นอยู่กับการออกแบบ ซึ่งก็คือส่วนของไซแนปส์ ในทางชีวภาพ ที่ทำหน้าที่เป็นทางผ่านให้ข้อมูลหรือข่าวสารสามารถส่งผ่านจากนิวรอนตัวหนึ่งไปยังนิวรอนตัวอื่น ซึ่งแต่ละไซแนปส์จะมีค่าน้ำหนักประจำตัวเรียกว่า ไซแนปติกสแวงท์ (Synaptic Weight (W)) เมื่อมีข้อมูลเข้ามา แต่ละข้อมูลจะต้องถูกคูณกับค่าน้ำหนักของไซแนปส์ที่รับข้อมูลนั้นๆ ตัวอย่างเช่น ค่าน้ำหนักอาจจะได้มาจากการเรียนรู้แบบ Delta Learning Rule คือ การเปรียบเทียบกันระหว่างคำตอบที่ได้จากโครงข่ายกับคำตอบเป้าหมาย แล้วนำค่าที่ผิดพลาด (Error) มาเข้าสมการเพื่อหาค่าน้ำหนักที่เหมาะสมต่อไป ลักษณะโครงสร้างของโครงข่ายประสาทเทียมมี 2 แบบคือแบบไปข้างหน้า (Feed Forward) และแบบป้อนกลับ (Feedback) ในบทนี้จะขอล่าวแต่เพียงแบบไปข้างหน้าอย่างเดียวซึ่งใช้ในวิทยานิพนธ์นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

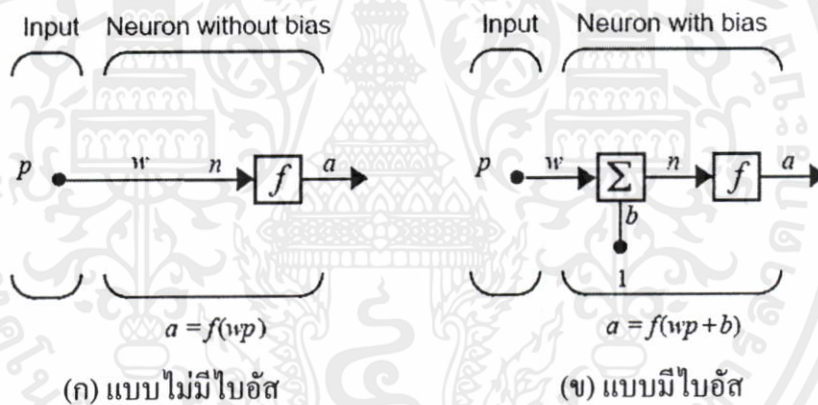
4.3.2.1 โครงข่ายประสาทเทียมเซลล์เดียว

จากลักษณะและการทำงานของเซลล์ประสาทหรือนิวรอน ดังที่กล่าวมาข้างต้น ได้ถูกนำมาสร้างทฤษฎีทางคณิตศาสตร์และจำลองการทำงานในรูปแบบพื้นฐานโดยใช้ชื่อว่าโครงข่ายประสาทเทียม

โครงข่ายประสาทเทียมแบบง่ายจะมีค่าอินพุตเป็นสเกลาร์หนึ่งอินพุต โดยไม่มีค่าไบอัส (bias) โดยค่าอินพุตสเกลาร์ p ถูกป้อนเข้าผ่านจุดต่อและคูณกับค่าถ่วงน้ำหนักสเกลาร์ (scalar weight (w)) และได้ผลคูณเป็นค่าสเกลาร์ wp กลายเป็นค่าอินพุตที่ถูกจัดน้ำหนัก (weighted input (n)) ถูกส่งต่อไปยังฟังก์ชันถ่ายโอน (transfer function (f)) ซึ่งเกิดเป็นค่าเอาต์พุตสเกลาร์ (scalar output (a)) ดังแสดงในรูปที่ 4.9 (ก) ซึ่งสามารถคำนวณค่าเอาต์พุต a ได้จากสมการที่ 4.3

$$n = wp \tag{4.2}$$

$$a = f(n) = f(wp) \tag{4.3}$$



รูปที่ 4.9 โครงข่ายประสาทเทียมหนึ่งหน่วยแบบง่าย

สำหรับรูปที่ 4.9 (ข) แสดงโครงข่ายประสาทเทียมแบบง่ายที่มีค่าอินพุตเป็นสเกลาร์หนึ่งอินพุต และมีค่าไบอัส b ซึ่งมีอินพุตเป็น 1 โดยค่าสเกลาร์เอาต์พุตสามารถคำนวณได้จากสมการที่ 4.5

$$n = wp + b \tag{4.4}$$

$$a = f(n) = f(wp + b) \tag{4.5}$$

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการเรียนการสอนของคณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี โดยไม่มีค่า
 ในที่นี้ f เป็นฟังก์ชันถ่ายโอน ตัวอย่างเช่น ฟังก์ชันซันบันไดและฟังก์ชันซิกมอยด์ เป็นต้น
 ซึ่งทำหน้าที่รับค่าอินพุต n ($n = wp$ หรือ $n = wp + b$) เพื่อเปลี่ยนเป็นค่าเอาต์พุต a ($a = f(n)$)

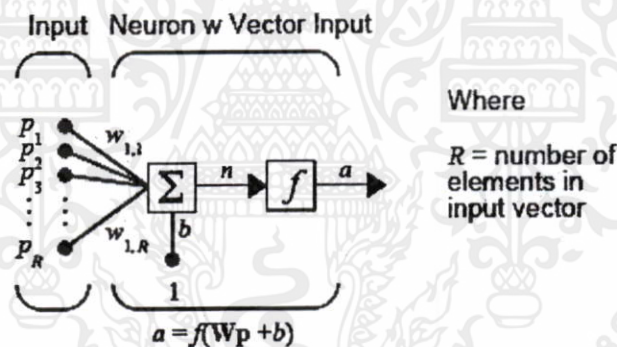
สำหรับค่าถ่วงน้ำหนัก w และค่าไบอัส b เป็นค่าพารามิเตอร์ของโครงข่ายประสาทเทียมที่สามารถปรับได้ จุดศูนย์กลางแนวคิดเกี่ยวกับโครงข่ายประสาทเทียมก็คือค่าพารามิเตอร์ w และ b สามารถปรับเปลี่ยนได้เพื่อให้โครงข่ายประสาทเทียมแสดงพฤติกรรมตามที่เรากำลังต้องการ ตัวอย่างเช่น การจดจำสิ่งที่เคยเรียนรู้มาก่อน ดังนั้นเราสามารถสอนโครงข่ายประสาทเทียมเพื่อให้งานที่เราต้องการ โดยการปรับแต่งค่าพารามิเตอร์น้ำหนักและไบอัสของโครงข่ายประสาทเทียม หรือบางครั้งโครงข่ายประสาทเทียมจะปรับพารามิเตอร์เพื่อให้ได้อย่างอัตโนมัติ

4.3.2.2 โครงข่ายประสาทเทียมแบบหลายอินพุท

โครงข่ายประสาทเทียมแบบหลายอินพุทในรูปเวกเตอร์ $\mathbf{p} = [p_1 \ p_2 \ p_3 \ \dots \ p_R]^T$ มีอินพุท R ค่า โดยอินพุทแต่ละตัวถูกคูณด้วยเวกเตอร์น้ำหนัก $\mathbf{W} = [w_{11} \ w_{12} \ w_{13} \ \dots \ w_{1R}]$ แล้วป้อนให้กับฟังก์ชันถ่ายโอน f เป็นเอาต์พุต a ซึ่งโครงสร้างแสดงในรูปที่ 4.10 และมีสมการดังนี้

$$n = \mathbf{W}\mathbf{p} + b = w_{11}p_1 + w_{12}p_2 + w_{13}p_3 + \dots + w_{1R}p_R + b \quad (4.6)$$

$$a = f(n) = f(\mathbf{W}\mathbf{p} + b) \quad (4.7)$$



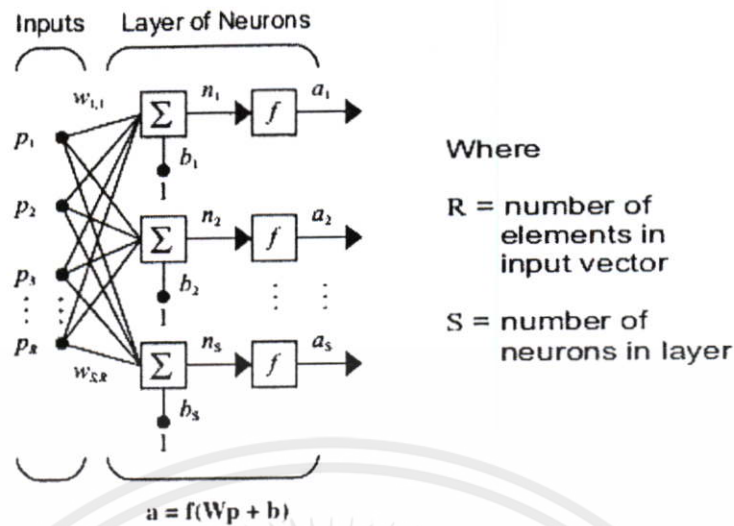
รูปที่ 4.10 โครงข่ายประสาทเทียมแบบหลายอินพุท

4.3.2.3 โครงสร้างของโครงข่ายประสาทเทียม

โครงข่ายประสาทเทียมที่มีเพียงเซลล์เดียวหลายอินพุท อาจจะไม่ใช่เพียงพอต่อการแก้ปัญหาที่ซับซ้อน ซึ่งสามารถแก้ไขได้โดยการนำนิเวรอนหลายๆ ตัวมาขนานกัน หรือเรียกว่า ชั้น (Layer) รูปแบบของการเชื่อมโยงโครงข่ายมีหลายชนิด แต่ละชนิดมีคุณลักษณะเด่นต่างกันไป ดังนี้

4.3.2.3.1 โครงข่ายประสาทเทียมแบบชั้นเดียว (Single Layer ANNs)

ในระบบงานปกติทั่วไปจะมีตัวแปรมากกว่าหนึ่งตัว โครงข่ายประสาทเทียมแบบนี้จะมีหลายสัญญาณเข้าและหลายสัญญาณออก ซึ่งทำให้สามารถนำไปประยุกต์ใช้กับระบบหลายตัวแปรก็ได้ โครงข่ายประสาทเทียมแบบหลายอินพุทเอาต์พุทมีชื่อทั่วไปว่า โครงข่ายประสาทเทียมแบบเป็นชั้น ซึ่งรูปที่ 4.11 แสดงโครงข่ายประสาทเทียมแบบชั้นเดียว



Where

R = number of elements in input vector

S = number of neurons in layer

รูปที่ 4.11 โครงข่ายประสาทเทียมแบบชั้นเดียว

จากรูปที่ 4.11 ค่าต่างๆ สามารถคำนวณได้จากสมการที่ 4.8 ถึง 4.11 ดังนี้

$$n_1 = w_{11}p_1 + w_{12}p_2 + w_{13}p_3 + \dots + w_{1R}p_R + b_1 \quad (4.8)$$

$$n_2 = w_{21}p_1 + w_{22}p_2 + w_{23}p_3 + \dots + w_{2R}p_R + b_2 \quad (4.9)$$

$$n_3 = w_{31}p_1 + w_{32}p_2 + w_{33}p_3 + \dots + w_{3R}p_R + b_3 \quad (4.10)$$

$$n_S = w_{S1}p_1 + w_{S2}p_2 + w_{S3}p_3 + \dots + w_{SR}p_R + b_S \quad (4.11)$$

และเอาที่พหุหาได้จากสมการที่ 4.12 ถึง 4.15

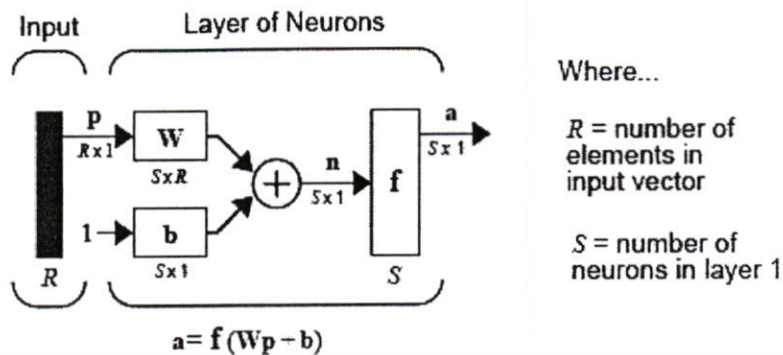
$$a_1 = f(n_1) = f(w_{11}p_1 + w_{12}p_2 + w_{13}p_3 + \dots + w_{1R}p_R + b_1) \quad (4.12)$$

$$a_2 = f(n_2) = f(w_{21}p_1 + w_{22}p_2 + w_{23}p_3 + \dots + w_{2R}p_R + b_2) \quad (4.13)$$

$$a_3 = f(n_3) = f(w_{31}p_1 + w_{32}p_2 + w_{33}p_3 + \dots + w_{3R}p_R + b_3) \quad (4.14)$$

$$a_S = f(n_S) = f(w_{S1}p_1 + w_{S2}p_2 + w_{S3}p_3 + \dots + w_{SR}p_R + b_S) \quad (4.15)$$

รูปที่ 4.12 แสดงโครงข่ายประสาทเทียมแบบหนึ่งชั้นที่มีหลายอินพุตหลายเอาต์พุตในรูปเวกเตอร์-เมทริกซ์ ทำให้ประหยัดการเขียน เมื่อ p เป็นอินพุตเวกเตอร์ขนาด $R \times 1$ W เป็นเมทริกซ์นำหนักขนาด $S \times R$ b เป็นไบอัสเวกเตอร์ขนาด $S \times 1$ n เป็นเวกเตอร์ผลบวกระหว่าง Wp และ b ส่วน f เป็นเวกเตอร์ฟังก์ชันถ่ายโอน และ a เป็นเวกเตอร์เอาต์พุต ส่วน R เป็นค่าสเกลาร์แสดงจำนวนอินพุต และ S เป็นค่าสเกลาร์แสดงจำนวนนินทรอน



Where...
 R = number of elements in input vector
 S = number of neurons in layer 1

รูปที่ 4.12 โครงข่ายประสาทเทียมแบบชั้นเดียวในรูปแบบย่อ

โดยกำหนดให้

$$p = [p_1 \ p_2 \ p_3 \ \dots \ p_R]^T \tag{4.16}$$

$$W = \begin{bmatrix} w_{11} & w_{12} & w_{13} & \dots & w_{1R} \\ w_{21} & w_{22} & w_{23} & \dots & w_{2R} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{S1} & w_{S2} & w_{S3} & \dots & w_{SR} \end{bmatrix} \tag{4.17}$$

$$b = [b_1 \ b_2 \ b_3 \ \dots \ b_S]^T \tag{4.18}$$

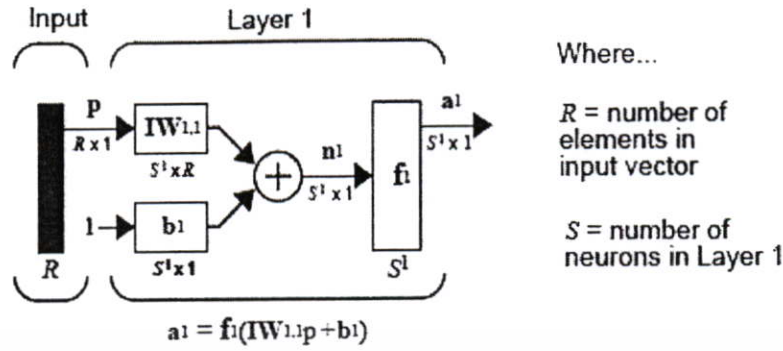
ดังนั้น จากรูปที่ 4.11 ค่าต่างๆ สามารถคำนวณได้ดังสมการต่อไปนี้

$$n = Wp + b \tag{4.19}$$

$$a = f(n) = f(Wp + b) \tag{4.20}$$

รูปที่ 4.13 แสดงโครงข่ายประสาทเทียมแบบหนึ่งชั้นที่ใช้ใน MATLAB Toolbox โดย $IW^{1,1}$ เป็นน้ำหนักอินพุต (Input Weight) เป็นการเชื่อมโยงจากต้นทางไปยังปลายทาง สำหรับพารามิเตอร์อื่นๆ ที่มีตัวยกเป็นเลข 1 เป็นการบอกว่าเป็นพารามิเตอร์ของชั้นที่ 1

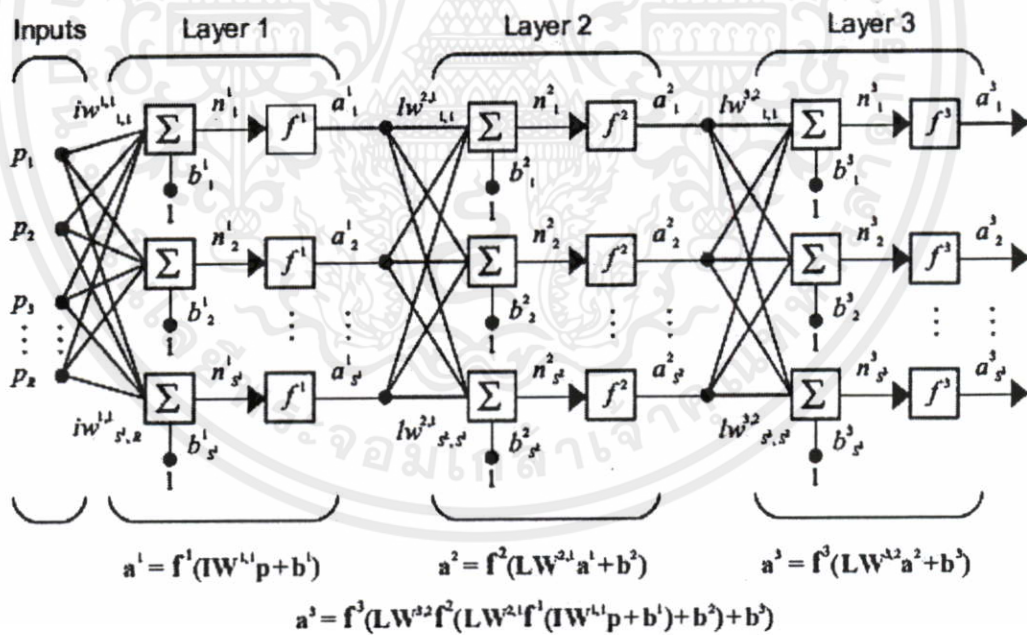
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 โครงข่ายประสาทเทียมแบบชั้นเดียวที่ใช้ใน MATLAB Toolbox

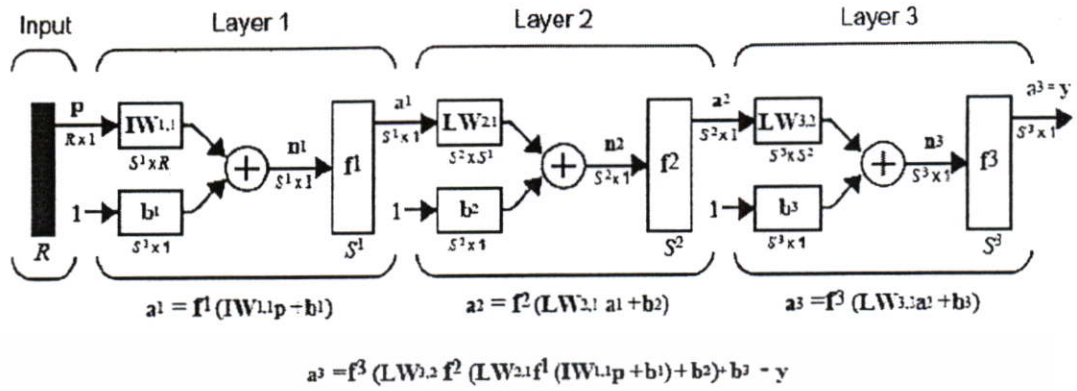
4.3.2.3.2 โครงข่ายประสาทเทียมแบบหลายชั้น (Multilayer ANNs)

โครงข่ายประสาทเทียมแบบหลายชั้น เป็นโครงข่ายประสาทเทียมที่นิยมใช้มากที่สุด ซึ่งสามารถทำงานที่มีความซับซ้อนมากๆ ได้ อาจกล่าวได้ว่าสามารถประยุกต์ใช้ได้กับงานเกือบทุกประเภท โดยมีข้อแม้ว่าต้องมีจำนวนชั้นและจำนวนนิวรอนที่เหมาะสม รูปที่ 4.14 แสดงโครงข่ายประสาทเทียมแบบสามชั้น และรูปที่ 4.15 แสดงโครงข่ายประสาทเทียมแบบสามชั้นในรูปแบบย่อ



รูปที่ 4.14 โครงข่ายประสาทเทียมสามชั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.15 โครงข่ายประสาทเทียมสามชั้นในรูปแบบย่อ

จากรูปที่ 4.14 เป็นตัวอย่างโครงข่ายประสาทเทียมแบบหลายชั้น โดยโครงข่ายประสาทเทียมแบบหลายชั้นนี้มีการเรียกชื่อชั้นต่างๆ ดังนี้ คือ ชั้นแรกที่อยู่ตรงโดยตรงกับอินพุต เรียกว่า ชั้นอินพุต (Input layer) ซึ่งในชั้นนี้ไม่มีการคำนวณ แต่จะทำหน้าที่เชื่อมต่อข้อมูลไปยังชั้นถัดไป ชั้นที่อยู่สุดท้ายทางขวามือ เรียกว่า ชั้นเอาต์พุต (Output layer) เป็นชั้นที่โครงข่ายให้ผลลัพธ์ ส่วนชั้นที่อยู่ระหว่างอินพุตและเอาต์พุต จะมีกี่ชั้นก็ตามจะเรียกว่า ชั้นซ่อน (Hidden layer)

การกำหนดจำนวนของชั้นซ่อนในโครงข่ายหนึ่งๆ ว่าควรมีกี่ชั้นและแต่ละชั้นประกอบด้วยกี่โหนดนั้น ไม่มีกฎเกณฑ์หรือทฤษฎีที่แน่นอน ดังนั้นในการกำหนดจำนวนชั้นและโหนดของแต่ละชั้นซ่อนสามารถได้มาจากการทดลอง

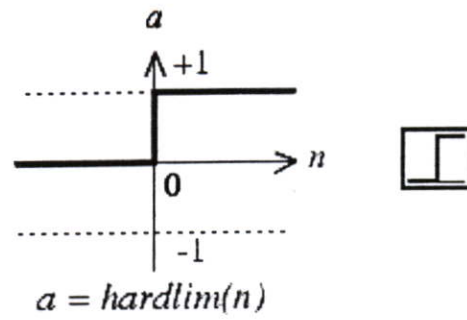
4.3.2.4 ฟังก์ชันถ่ายโอน (Transfer Function)

ฟังก์ชันถ่ายโอน [38] มีอยู่มากมายหลายชนิดด้วยกัน เช่น ฟังก์ชันถ่ายโอนแบบ Hard-limit ฟังก์ชันถ่ายโอนแบบ Symmetry Hard-limit สมมาตร ฟังก์ชันถ่ายโอนแบบเชิงเส้น (Linear) ฟังก์ชันถ่ายโอนแบบลอการิทึมซิกมอยด์ (Logarithmic sigmoid) ฟังก์ชันถ่ายโอนแบบไฮเพอร์บอลิกแทนเจนต์ซิกมอยด์ หรือแทนซิกมอยด์ (Hyperbolic tangent sigmoid) และอื่นๆ ซึ่งฟังก์ชันถ่ายโอนแต่ละชนิดมีรายละเอียดดังนี้

4.3.2.4.1 ฟังก์ชันถ่ายโอนแบบ Hard-limit

$$f(n) = \begin{cases} 0 & ; n < 0 \\ 1 & ; n \geq 0 \end{cases} \quad (4.21)$$

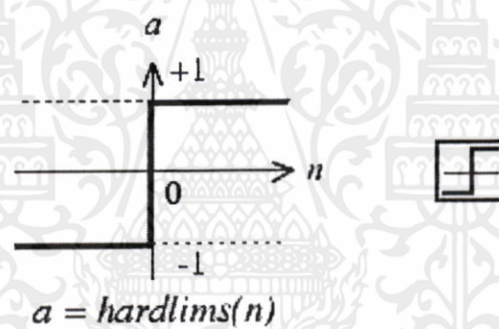
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 ฟังก์ชันถ่ายโอนแบบ Hard-limit

4.3.2.4.2 ฟังก์ชันถ่ายโอนแบบ Symmetry Hard-limit

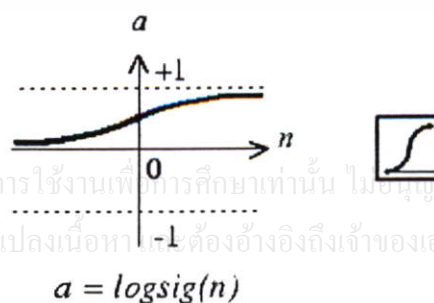
$$f(n) = \begin{cases} -1 & ; n < 0 \\ 1 & ; n \geq 0 \end{cases} \quad (4.22)$$



รูปที่ 4.17 ฟังก์ชันถ่ายโอนแบบ Symmetry Hard-limit

4.3.2.4.3 ฟังก์ชันถ่ายโอนแบบลอกซิกมอยด์

$$f(n) = \frac{1}{1 + \exp(-n)} \quad (4.23)$$

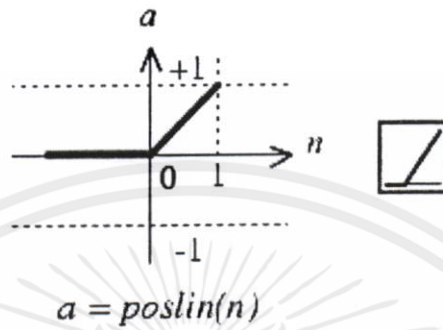


รูปที่ 4.18 ฟังก์ชันถ่ายโอนแบบลอกซิกมอยด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาใดๆ ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2.4.4 ฟังก์ชันถ่ายโอนแบบเชิงเส้นแบบบวก (Positive linear)

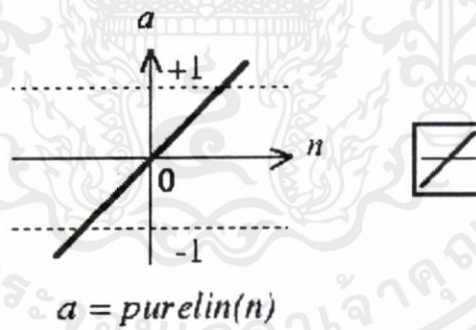
$$f(n) = \begin{cases} n & ; n \geq 0 \\ 0 & ; n < 0 \end{cases} \quad (4.24)$$



รูปที่ 4.19 ฟังก์ชันถ่ายโอนแบบเชิงเส้นแบบบวก

4.3.2.4.5 ฟังก์ชันถ่ายโอนแบบเชิงเส้น (Linear)

$$f(n) = n \quad (4.25)$$

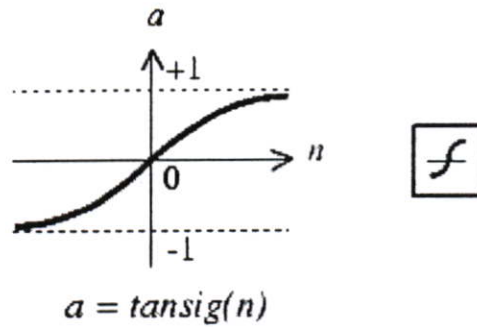


รูปที่ 4.20 ฟังก์ชันถ่ายโอนแบบเชิงเส้น

4.3.2.4.6 ฟังก์ชันถ่ายโอนแบบแทนซิกมอยด์

$$f(n) = \frac{\exp(n) - \exp(-n)}{\exp(n) + \exp(-n)} \quad (4.26)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาด้านนี้ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.21 ฟังก์ชันถ่ายโอนแบบแทนซิกมอยด์

ฟังก์ชันถ่ายโอนที่นิยมใช้ในการแก้ปัญหาที่ไม่เป็นเชิงเส้น (Non Linear) คือ ฟังก์ชันลอกซิกมอยด์ เนื่องจากเป็นฟังก์ชันที่มีความไวต่อสัญญาณอินพุตขนาดเล็กๆ คือ เมื่อสัญญาณมีค่าไปทางบวกเพียงเล็กน้อย เอาท์พุทที่ได้จะมีค่าใกล้เคียง “1” ในขณะที่สัญญาณมีค่าไปทางลบเพียงเล็กน้อย เอาท์พุทที่ได้จะมีค่าใกล้เคียง “0” และมีการเฉลี่ยสัญญาณที่มีขนาดใหญ่ คือ เมื่อสัญญาณมีค่าไปทางบวกมากๆ เอาท์พุทที่ได้ก็ยังมีค่าใกล้เคียง “1” ในขณะที่สัญญาณมีค่าไปทางลบมากๆ เอาท์พุทที่ได้ก็ยังมีค่าใกล้เคียง “0” ซึ่งเป็นคุณสมบัติของการขยายที่ไม่เป็นเชิงเส้น (Non Linear Gain)

4.3.2.5 การฝึกสอนและการเรียนรู้ของโครงข่ายประสาทเทียม

ค่าน้ำหนักเปลี่ยนแปลงได้อย่างไรและมีความสัมพันธ์กับอะไรบ้าง เนื่องจากโครงข่ายประสาทเทียมที่สร้างขึ้นมานั้น เปรียบเสมือนกับเด็กที่คลอดออกมาที่มีสมอง แต่สมองยังไม่เจริญเติบโตเพียงพอและยังไม่ได้รับการฝึกฝนและเรียนรู้ เด็กจึงไม่สามารถทำกิจกรรมใดๆ ได้ด้วยตนเอง เว้นแต่กิจกรรมที่ธรรมชาติสร้างมาให้พร้อมกับการกำเนิดที่เรียกว่า “สัญชาตญาณ” ซึ่งธรรมชาติใส่คุณลักษณะบางอย่างให้เซลล์สมองบางส่วนตั้งแต่ทารกเจริญเติบโตอยู่ในครรภ์ของมารดา อย่างเช่น ระบบควบคุมการหายใจ ความรู้สึก การเรียกร้องเมื่อหิว หรือการตอบสนองต่อสิ่งเร้า ฯลฯ เด็กจะพัฒนาการเรียนรู้ไปตามขั้นตอน หลังจากนั้นสมองของเขาจะได้รับการฝึกสอนและเจริญเติบโตไปพร้อมๆ กัน เซลล์สมองจะได้รับการปรับคุณลักษณะและเจริญเติบโตเป็นโครงข่ายที่สอดคล้องกับการฝึกสอน

โครงข่ายประสาทเทียมที่สร้างขึ้นก็มีลักษณะเช่นเดียวกันคือ เมื่อสร้างเสร็จ แต่ละเซลล์ประสาทที่สร้างขึ้นมานั้นจะยังไม่มีความรู้ใดๆ เลย เนื่องจากยังไม่มีการกำหนดค่าไซแนปติกส์

เอกสารนี้... ตามที่ต้องการ การฝึกสอนของโครงข่ายประสาทเทียม จะกระทำโดยการปรับเปลี่ยนค่า... ไม่ว่ากรณีใดไซแนปติกส์เวกซ์ ซึ่งก็คือการเรียนรู้ของโครงข่ายประสาทเทียมนั่นเอง

การปรับปรุงค่าถ่วงน้ำหนักอันเนื่องมาจากการฝึกสอนให้กับโครงข่ายประสาทเทียมสามารถแยกออกเป็น 2 ลักษณะ คือ

1. Incremental Mode คือ การปรับปรุงค่าถ่วงน้ำหนักหลังจากแต่ละอินพุตถูกป้อนเพื่อทำการฝึกสอนให้กับโครงข่ายประสาทเทียม

2. Batch Mode คือ การปรับปรุงค่าถ่วงน้ำหนักหลังจากอินพุตทั้งหมดถูกป้อนเพื่อทำการฝึกสอนให้กับโครงข่ายประสาทเทียม

โครงข่ายประสาทเทียมมีรูปแบบการฝึกสอนอยู่ด้วยกัน 2 แบบ ดังนี้

4.3.2.5.1 การฝึกสอนแบบอิสระ (Unsupervised Training)

การฝึกสอนแบบอิสระ เป็นการฝึกสอนที่ต้องการเฉพาะข้อมูลอินพุตป้อนเข้าสู่โครงข่าย และภายในโครงข่ายจะมีโหนดเอาต์พุตอยู่หลายโหนดด้วยกัน โดยแต่ละโหนดจะแทนกลุ่มของข้อมูลที่มีคุณสมบัติเหมือนกัน และเมื่อป้อนอินพุตเข้าสู่โครงข่าย โครงข่ายก็จะทำการคำนวณค่าความสัมพันธ์ที่มีอยู่ภายในเซตของอินพุต โดยอาศัยค่าถ่วงน้ำหนักเป็นตัวแยกความแตกต่างของอินพุตไปเก็บไว้ในโหนดเอาต์พุตของโครงข่าย ข้อดีของการฝึกสอนแบบนี้ คือ มีความใกล้เคียงกับโครงข่ายประสาทชีวภาพ แต่ไม่สามารถแก้ปัญหาที่ซับซ้อนมากๆ ได้

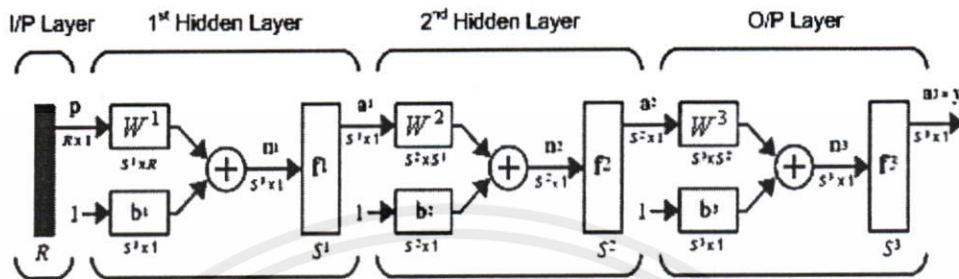
4.3.2.5.2 การฝึกสอนแบบควบคุม (Supervised Training)

การฝึกสอนโดยวิธีนี้จะกำหนดเซตของการสอนให้กับโครงข่าย ซึ่งภายในเซตประกอบด้วยข้อมูลอินพุตและเอาต์พุตที่ต้องการ เรียกว่า คู่ของการฝึกสอน (Training pairs) เมื่อทำการป้อนอินพุตให้กับโครงข่าย โครงข่ายก็จะทำการประมวลผลจนได้คำตอบ ซึ่งคำตอบที่ได้จากโครงข่ายนี้จะถูกนำมาคำนวณค่าความผิดพลาด (Error) ก็คือ ค่าความแตกต่างระหว่างคำตอบที่ได้จากโครงข่ายกับคำตอบเป้าหมาย (Target) ของอินพุตชุดเดียวกัน ว่ามีค่ามากน้อยเพียงใด ถ้ายังมีค่าความผิดพลาดสูงอยู่ ก็จะทำให้การฝึกสอนรอบใหม่ให้กับโครงข่าย เพื่อทำการปรับค่าไซแนปติกส์เวกซ์และทำการฝึกสอนต่อไปจนกระทั่งค่าความผิดพลาดระหว่างคำตอบที่ได้จากโครงข่ายกับคำตอบเป้าหมายของอินพุตชุดเดียวกันมีค่าน้อยพอที่จะยอมรับได้ จึงหยุดทำการฝึกสอนโครงข่ายต่อไปเป็นตัวอย่างของการฝึกสอนแบบควบคุมรูปแบบหนึ่ง ซึ่งในปัจจุบันนิยมนำมาประยุกต์ใช้กันมาก

การเรียนรู้แบบแพร่ย้อนกลับ (Backpropagation Learning)[39][40]

โครงข่ายประสาทเทียมแบบหลายชั้นสามารถแก้ปัญหาที่ซับซ้อนมากๆ ได้ดีกว่าโครงข่ายประสาทเทียมแบบชั้นเดียว แต่การฝึกสอนอาจจะยากกว่า ซึ่งโครงข่ายประสาทเทียมแบบหลายชั้นที่มีชั้นซ่อนหนึ่งหรือมากกว่านั้น การส่งข้อมูลระหว่างอินพุตและเอาต์พุต จะถูกส่งไปในทิศทางเดียว (Feed Forward) คือ จากอินพุตไปเอาต์พุต ไม่มีการส่งข้อมูลจากเอาต์พุตกลับไปยังอินพุต แต่อาจมีการส่งค่าผิดพลาดจากเอาต์พุตกลับไปยังอินพุตเพื่อแก้ไขความถูกต้องเรียกว่า การเรียนรู้แบบแพร่ย้อนกลับ เป็นการเรียนรู้แบบควบคุม คือ จะมีเซตของข้อมูลอินพุตและเอาต์พุตเป้าหมาย โดยทำการฝึกสอนให้กับโครงข่ายประสาทเทียมรอบแรก และเปรียบเทียบกับเอาต์พุตที่ได้จากระบบกับ

เอาต์พุตเป้าหมาย ค่าที่ได้นี้ คือ ค่าผิดพลาดของชั้นเอาต์พุต หลังจากนั้นส่งค่าผิดพลาดของชั้นเอาต์พุตนี้กลับไปเพื่อใช้ในการคำนวณค่าผิดพลาดของชั้นซ่อนและนำค่าผิดพลาดที่ได้จากการคำนวณทั้งหมดไปปรับปรุ้ค่าถ่วงน้ำหนัก (Update Weight) ในทุกๆ ชั้นของโครงข่าย



รูปที่ 4.22 โครงข่ายประสาทเทียมที่ใช้ในการเรียนรู้แบบแพร่ย้อนกลับ

รูปที่ 4.22 เป็นตัวอย่างของโครงข่ายประสาทเทียมที่ใช้ในการเรียนรู้แบบแพร่ย้อนกลับ ซึ่งขั้นตอนการเรียนรู้แบบแพร่ย้อนกลับสามารถอธิบายเป็นข้อๆ ได้ดังนี้

ขั้นตอนที่ 1 ออกแบบ โครงสร้างของระบบ

ทำการออกแบบระบบ โดยต้องกำหนดจำนวนชั้น ฟังก์ชันถ่ายโอน และเงื่อนไขการหยุดฝึกสอน ซึ่งในรูปที่ 4.22 เป็นตัวอย่างของโครงข่ายที่มี 3 ชั้น

ขั้นตอนที่ 2 กำหนดค่าถ่วงน้ำหนักและไบอัสเริ่มต้น

ทำการสุ่มค่าถ่วงน้ำหนัก (W) ในการเชื่อมต่อและค่าไบอัส (b) ทั้งหมดในโครงข่าย โดยกำหนดให้มีค่าน้อยๆ (-1 ถึง +1)

ขั้นตอนที่ 3 การฝึกสอน

เป็นขั้นตอนการฝึกสอนเพื่อให้โครงข่ายประสาทเทียมได้เรียนรู้ ซึ่งประกอบด้วย 3 ส่วนหลัก คือ ขั้นตอน Feedforward (ขั้นตอนที่ 4 ถึง 7) ขั้นตอน Backpropagation of Error (ขั้นตอนที่ 8 ถึง 10) และสุดท้ายเป็นขั้นตอนการปรับปรุ้ค่าถ่วงน้ำหนักและไบอัส (ขั้นตอนที่ 11)

ขั้นตอนที่ 4 ป้อนคู่ของอินพุตและเอาต์พุต

ป้อนคู่ของการฝึกสอน คือ อินพุต (p) และเอาต์พุตเป้าหมาย ($target(t)$) เข้าสู่โครงข่าย

เอกสารนี้เป็นประสาทเทียม วนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 5 รับสัญญาณอินพุต

เป็นการรับสัญญาณอินพุตเข้ามาในแต่ละโหนดของอินพุต และแพร่กระจายไปสู่ทุกๆ โหนดในชั้นถัดไป

ขั้นตอนที่ 6 คำนวณผลรวมค่าถ่วงน้ำหนักและอินพุตที่รับเข้ามาในทุกโหนดของโครงข่าย เป็นการคำนวณหาผลรวมของสัญญาณอินพุตที่ป้อนเข้ามาตามเส้นทางของค่าถ่วงน้ำหนักต่างๆ โดย n^1 คือ ค่าผลรวมของชั้นซ่อนที่ 1 มีสมการดังนี้

$$n^1 = \sum(pW^1 + b^1) \quad (4.27)$$

โดย p คือ สัญญาณอินพุต

W^1 คือ ค่าถ่วงน้ำหนักประจำตัวของแต่ละโหนดในชั้นซ่อนที่หนึ่ง

และ b^1 คือ ค่าไบอัสประจำตัวของแต่ละโหนดในชั้นซ่อนที่หนึ่ง

จากนั้นทำการปรับสัญญาณที่ออกจากชั้นนี้ด้วยฟังก์ชันถ่ายโอน ดังสมการที่ 4.28 แล้วทำการส่งต่อไปยังชั้นต่อไป

$$a^1 = f^1(pW^1 + b^1) \quad (4.28)$$

ทำการคำนวณตามขั้นตอนที่ 6 ให้กับแต่ละโหนดในทุกชั้นซ่อนของโครงข่ายประสาทเทียม

ขั้นตอนที่ 7 คำนวณผลรวมค่าถ่วงน้ำหนักและอินพุตที่รับเข้ามาในโหนดเอาต์พุต

เป็นการคำนวณหาผลรวมของสัญญาณอินพุตที่ป้อนเข้ามาตามเส้นทางของค่าถ่วงน้ำหนักต่างๆ โดย n^3 คือ ค่าผลรวมของชั้นเอาต์พุต สมการมีดังนี้

$$n^3 = \sum(a^2W^3 + b^3) \quad (4.29)$$

โดย a^2 คือ สัญญาณเอาต์พุตจากชั้นที่สอง ซึ่งเป็นสัญญาณอินพุตของชั้นเอาต์พุต

W^3 คือ ค่าถ่วงน้ำหนักประจำตัวของแต่ละโหนดในชั้นเอาต์พุต

และ b^3 คือ ค่าไบอัสประจำตัวของแต่ละโหนดในชั้นเอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรณีสืบค้นเพื่อการศึกษาค้นคว้าเท่านั้น ไม่สามารถนำออกจำหน่าย หรือทำซ้ำโดยไม่ได้รับอนุญาต

จากนั้นทำการปรับสัญญาณที่ออกจากชั้นนี้ด้วยฟังก์ชันถ่ายโอน ดังสมการที่ 4.30 ซึ่งเป็นเอาต์พุทของโครงข่าย

$$y = f^3(a^2 W^3 + b^3) \quad (4.30)$$

ขั้นตอนที่ 8 คำนวณค่าผิดพลาด

เป็นการหาค่าความคลาดเคลื่อน ระหว่างเอาต์พุทที่ได้จริงกับเอาต์พุทเป้าหมาย

$$E = t - y \quad (4.31)$$

ขั้นตอนที่ 9 หาอนุพันธ์ของฟังก์ชันในทุกชั้น

ขั้นตอนที่ 10 คำนวณค่าความไว

ทำการคำนวณค่าความไว (Sensitivity(s)) ของทุกชั้น โดยย้อนจากเอาต์พุทไปยังอินพุท

$$s^M = -2\dot{F}^M(n^M)(t - a) \quad (4.32)$$

$$s^m = \dot{F}^M(n^M)(W^{m+1})^T s^{m+1} \quad (4.33)$$

โดย $M =$ จำนวนชั้น

และ $m = M-1, \dots, 3, 2, 1$

ขั้นตอนที่ 11 ปรับปรุงค่าถ่วงน้ำหนักและค่าไบอัส

ทำการปรับปรุงค่าถ่วงน้ำหนักในทุกๆ ชั้นของโครงข่าย ดังสมการที่ 4.34

$$W^m(k+1) = W^m(k) - \alpha s^m (a^{m-1})^T \quad (4.34)$$

และปรับปรุงค่าไบอัสในทุกๆ ชั้น ดังสมการที่ 4.35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานที่ออกลิขสิทธิ์ฉบับนี้ ไม่อนุญาตให้นำไปใช้ประโยชน์อื่นใด
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$b^m(k+1) = b^m(k) - \alpha s^m \quad (4.35)$$

โดย α คือ อัตราการเรียนรู้

ขั้นตอนที่ 12 วนไปทำขั้นตอนที่ 4 ถึงขั้นตอนที่ 11 จนข้อมูลถูกฝึกสอนให้กับโครงข่ายประสาทเทียมครบทุกชุด

ขั้นตอนที่ 13 การหยุดฝึกสอนโครงข่ายประสาทเทียม ซึ่งมีเงื่อนไขดังนี้

1. ค่าผิดพลาด : โดยทำการเปรียบเทียบค่าผิดพลาด (E) ที่คำนวณได้จากข้อ 8 กับค่าผิดพลาดสูงสุด (E_k) ที่ได้กำหนดไว้ ถ้าค่าผิดพลาดที่ได้มากกว่าค่าผิดพลาดสูงสุด ($E > E_k$) จะต้องกลับไปทำข้อ 4 และเริ่มจากข้อมูลชุดแรกใหม่ จนกระทั่งค่าผิดพลาดที่ได้น้อยกว่าค่าผิดพลาดสูงสุด ($E < E_k$) จึงหยุดทำการฝึกสอนให้กับโครงข่าย

2. จำนวนรอบของการฝึกสอน : หยุดทำการฝึกสอนให้กับโครงข่าย เมื่อจำนวนรอบการฝึกสอนให้กับโครงข่ายถึงจำนวนรอบที่ตั้งไว้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การทดลองและผลการทดลอง

เนื้อหาในบทนี้แบ่งออกเป็น 2 ส่วนใหญ่ๆ ส่วนแรกในหัวข้อที่ 5.1 เราจะอธิบายถึงขั้นตอนในการเตรียมข้อมูล ส่วนที่สองหัวข้อที่ 5.2 จะกล่าวถึงการทดลองและผลการทดลองของประสิทธิภาพในการรู้จำ ซึ่งในส่วนนี้จะแบ่งออกเป็น 5 ส่วนย่อยคือ ประสิทธิภาพของการรู้จำในด้านความคงทนต่อการแปลงแบบแอฟไฟน์ในหัวข้อที่ 5.2.1 ประสิทธิภาพของการรู้จำในด้านความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้นในหัวข้อที่ 5.2.2 ประสิทธิภาพของการรู้จำในด้านความคงทนต่อสัญญาณรบกวนในหัวข้อที่ 5.2.3 การเปรียบเทียบประสิทธิภาพกับวิธีการในงานวิจัยอื่นๆ ในหัวข้อที่ 5.2.4 และการทดลองกับภาพที่ถ่ายได้จากกล้องถ่ายรูปในหัวข้อที่ 5.2.5

5.1 การเตรียมข้อมูล

ภาพของรูปร่างที่ในงานวิจัยนี้ได้มาจาก MPEG7 [41] ซึ่งเป็นชุดภาพมาตรฐานที่นักวิจัยทางด้านภาพดิจิทัลส่วนใหญ่ใช้กัน โดยเราได้ทำการเลือกรูปร่างใน ชุดภาพ MPEG7 มา 40 รูปร่าง ดังแสดงในรูปที่ 5.1 โดยแต่ละรูปร่างจะถูกแปลงแบบแอฟไฟน์ไป 14 แบบดังแสดงในรูปที่ 5.2 ด้วยตัวแปรทางการแปลงแบบแอฟไฟน์ดังแสดงในตารางที่ 5.1 ดังนั้นชุดข้อมูลที่ใช้ในงานวิจัยนี้ ประกอบด้วยจำนวนรูปร่างทั้งหมด 560 รูปร่าง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานที่วารสารวิชาการ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 5.1 รูปร่างที่ใช้ในงานวิจัยนี้

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 ตัวอย่างรูปร่างที่ถูกแปลงแบบแอฟไฟน์ไป 14 แบบ

ตารางที่ 5.1 ตัวแปรทางการแปลงแบบแอฟไฟน์ของรูปร่างทั้ง 14 แบบ

Affine Transform Parameter	ค่าของ Affine Transform Parameter ในแต่ละแบบ													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Scaling	1	1.2	1.4	1.6	0.6	0.7	0.8	1.2	1.4	1.6	0.6	0.7	0.8	1
Rotation(θ°)	0	15	30	45	60	90	105	0	15	30	45	60	90	105
Shear-x	0	0.4	0.8	1.2	1.5	0.4	0.8	0	0	0	0	0.4	0.8	0.4
Shear-y	0	0	0	0	0	0.4	0.8	0.4	0.8	1.2	1.5	0.4	0.8	0.8

หมายเหตุ สำหรับรูปร่างที่ถูกแปลงแบบ Shear ไปมากเกินไปจนเสียลักษณะเค้าโครงของรูปร่างเดิม ดังแสดงในรูปที่ 5.3 จะทำให้ไม่สามารถดึง Shape representation ที่มีคุณสมบัติที่ดีหรือที่มีลักษณะเฉพาะออกมาจากรูปร่างได้



ก) รูปร่างเดิม

ข) รูปร่างที่ถูกแปลงแบบ Shear ไปมากเกินไป

รูปที่ 5.3 รูปร่างที่ถูกแปลงแบบ Shear ไปมากเกินไปจนเสียลักษณะเค้าโครงของรูปร่างเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น สำหรับขั้นตอนในการเตรียมข้อมูลนั้นเริ่มแรกเราจะนำรูปร่างที่ได้ทำการแยกออกจากฉาก
หลัง (background) แล้วมาหาเส้นรอบรูปโดยใช้วิธี Contour follower [42] ซึ่งหลักการคือกำหนด

จุดเริ่มต้นโดยจุดนี้ต้องเป็นจุดที่อยู่ในส่วนขอบนอกสุดของรูปร่างดังแสดงในรูปที่ 5.4 ก) จากนั้นก็หาจุดบนเส้นรอบรูปถัดไปโดยพิจารณาพิกเซล (pixel) รอบๆจุดเริ่มต้นนั้นเพื่อดูว่า pixel ไหนเป็นส่วนหนึ่งของรูปร่าง ทำอย่างนี้ไปเรื่อยๆจนกว่าจะได้จุดที่มีตำแหน่งเดียวกับจุดเริ่มต้น ในที่สุดเราก็จะได้คู่อันดับ (x,y) ของเส้นรอบของรูปร่างดังแสดงในรูปที่ 5.4 ข) สำหรับรูปร่างที่อยู่ในกลุ่มเดียวกันดังในรูปที่ 5.2 เราจะกำหนดให้มีจุดเริ่มต้นตำแหน่งเดียวกัน



ก) รูปร่างที่นำมาหาเส้นรอบรูปเครื่องหมาย * คือจุดเริ่มต้น

ข) เส้นรอบรูป

รูปที่ 5.4 เส้นรอบรูปที่หาได้จากวิธี Contour follower

โดยปกติแล้วเส้นรอบรูปของแต่ละรูปร่างจะมีจำนวนจุดบนเส้นรอบรูปไม่เท่ากัน เพื่อให้การพิจารณารูปร่างทุกรูปร่างอยู่ในมาตรฐานเดียวกัน เราจึงทำการสุ่มจุดในแต่ละเส้นรอบรูปออกมา 100 จุด ดังแสดงในรูปที่ 5.5 จากนั้นก็นำจุดสุ่มตัวอย่าง 100 จุดนั้น ไปหาคุณลักษณะเฉพาะที่ไม่เปลี่ยนแปลงต่อการแปลงแบบแอฟไฟน์ (1-D triangle area representation) ดังได้อธิบายไว้ในหัวข้อที่ 3.4.1 และ 3.4.2



รูปที่ 5.5 ตัวอย่างจุดสุ่มตัวอย่าง 100 จุดบนเส้นรอบรูป แสดงด้วยเครื่องหมาย *

5.2 การทดลองและผลการทดลองของประสิทธิภาพในการรู้จำ

ในการทดลองเราแบ่งเป็นห้าส่วนคือ ส่วนแรกการทดลองเพื่อดูประสิทธิภาพในการรู้จำด้าน Affine invariant อย่างเดียว โดยใช้ 1-D triangle area representation ที่ได้มาจากขั้นตอนในการหาโดยการกำหนดจุดเริ่มต้นในแต่ละรูปร่างดังได้อธิบายไว้ในหัวข้อก่อนหน้านั้น ทั้งนี้เพื่อลดผลกระทบที่เกิดจากการเปลี่ยนแปลงจุดเริ่มต้นซึ่งอาจจะส่งผลกระทบต่อประสิทธิภาพในการทนต่อการแปลงแบบแอฟไฟน์ ในส่วนที่สองนั้นเราได้ทำการพัฒนาวิธีการเพื่อให้การรู้จำสามารถทนต่อการ

เปลี่ยนแปลงจุดเริ่มต้น คือเราได้ทำการสุ่มจุดเริ่มต้น ดังนั้นในแต่ละรูปร่างจะมีจุดเริ่มต้นที่ต่างกัน จากนั้นจึงทำการทดลองเพื่อวัดประสิทธิภาพในการรู้จำของวิธีที่ทำการพัฒนาขึ้นดังกล่าว ในส่วนที่สามจะเป็นการทดลองเพื่อวัดความสามารถในการทนต่อสัญญาณรบกวน โดยเราจะทำการเพิ่มสัญญาณรบกวนไปยังเส้นรอบรูปแล้ววัดประสิทธิภาพในการรู้จำ ส่วนที่สี่เราจะทำการเปรียบเทียบประสิทธิภาพของวิธีการในงานวิจัยนี้กับงานวิจัยในบทความอื่นๆ และส่วนสุดท้ายส่วนที่ห้าเป็นการนำเอาวิธีการของเราไปทดลองวัดประสิทธิภาพในการรู้จำกับภาพถ่ายจริงที่ได้จากกล้องถ่ายรูป และแสดงตัวอย่างการนำไปประยุกต์ใช้งานจริง

5.2.1 ประสิทธิภาพของการรู้จำด้านความคงทนต่อการแปลงแบบแอฟไฟน์

ในงานวิจัยนี้เราได้ทำการศึกษาวิธีในการรู้จำ 2 วิธีคือ Normalized cross-correlation (NCC) และ Neural networks (NN) โดยในแต่ละวิธีเราจะทำการทดลองเพื่อวัดประสิทธิภาพในการรู้จำ แล้ววิเคราะห์ว่าทั้งสองวิธีมีข้อดีข้อเสียอะไรบ้าง นอกจากนั้นแล้วเรายังทำการสังเกตผลการทดลองของการรู้จำเพื่อศึกษาว่า 1-D triangle area representation แบบไหน (ATA หรือ NATA) และที่ความยาวแกนของสามเหลี่ยมที่เท่าไรจะมีประสิทธิภาพในการแยกแยะกลุ่มข้อมูลได้ดี โดยเราได้้นำการวิเคราะห์ทาง Fisher's discrimination มาเป็นทฤษฎีที่ใช้ทำนายผลเปรียบเทียบกับผลการทดลองหาความสามารถในการแยกแยะข้อมูลของ 1-D triangle area representation ที่ส่งผลต่อประสิทธิภาพของการรู้จำ สำหรับรายละเอียดดังกล่าวเราได้นำเสนอออกเป็น 4 หัวข้อย่อย คือ การรู้จำด้วยวิธี Normalized cross-correlation การรู้จำด้วยวิธี Neural networks การเปรียบเทียบข้อดีข้อเสียของทั้งสองวิธี และสุดท้ายกล่าวถึงการวัดประสิทธิภาพในการแยกแยะข้อมูลของ 1-D triangle area representation ด้วย Fisher's discrimination

5.2.1.1 การรู้จำด้วยวิธี Normalized cross-correlation

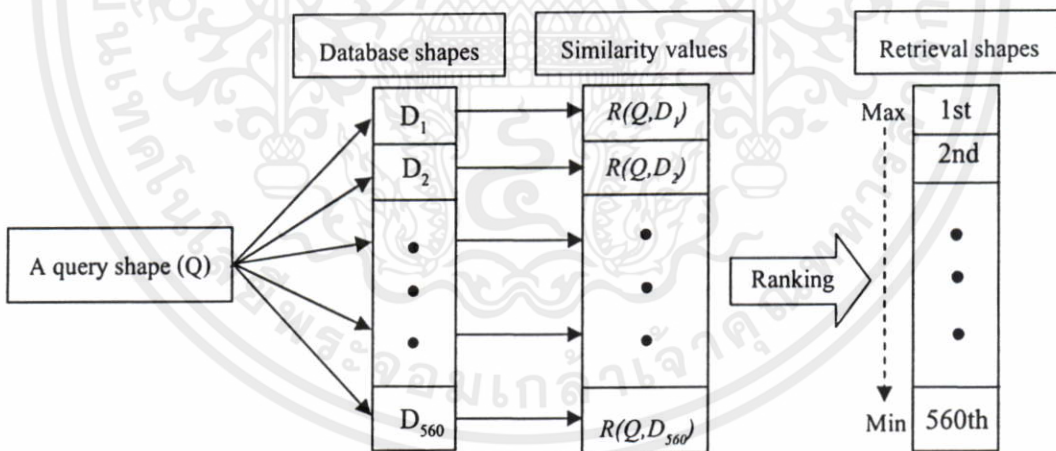
การรู้จำด้วยวิธี Normalized cross-correlation เป็นการระบุว่ารูปร่างที่นำมาทดสอบนั้นจัดอยู่ตรงกับกลุ่มใดในฐานข้อมูล ด้วยการนำ shape representation ที่อยู่ในรูปแบบกราฟหรือเวกเตอร์คุณลักษณะ (graph หรือ feature vector) ของรูปร่างดังกล่าวมาทำการเปรียบเทียบกับ shape representation ของทุกรูปร่างที่อยู่ในฐานข้อมูลแล้ววัดว่ารูปร่างในฐานข้อมูลใดที่ให้คะแนนความเหมือนมากที่สุด (โดยมีค่า Normalized cross-correlation ที่สูงที่สุด) รูปร่างที่นำมาทดสอบจะถูกจัดอยู่ในกลุ่มเดียวกับรูปร่างที่ให้คะแนนความเหมือนสูงสุดนั้น

ในส่วนของการทดลอง เราได้ทำการทดลองเพื่อวัดทั้งประสิทธิภาพในการรู้จำที่ได้จากการจับคู่รูปร่าง (matching) และการกู้คืนรูปร่างหรือ การสืบค้นข้อมูล (retrieval) ซึ่งหมายถึงการแสดงกลุ่มรูปร่างในฐานข้อมูลที่มีความคล้ายคลึงกับรูปร่างที่นำมาทดสอบ โดยพิจารณาตามค่าความเหมือน สำหรับในการทดลองฐานข้อมูลที่ใช้มีทั้งหมด 560 รูปร่าง ดังที่ได้กล่าวถึงไว้ในขั้นตอน

การเตรียมข้อมูล ซึ่งมีรูปร่างต่างกันอยู่ 40 ชนิด แต่ละชนิดถูกแปลงแบบแอฟฟاینไปชนิดละ 14 แบบ

ขั้นตอนในการทดลองแสดงดังบล็อกไดอะแกรมในรูปที่ 5.6 ซึ่งแสดงตัวอย่างของการนำรูปร่างหนึ่งไปทำการ match เริ่มด้วยการนำรูปร่างทดสอบ (Q) มาวัดความเหมือนกับทุกๆรูปร่างที่อยู่ในฐานข้อมูล (D) ด้วยฟังก์ชัน NCC ในสมการที่ 4.1 จากนั้นทำการเรียงคะแนน (Ranking) ความเหมือนของแต่ละรูปร่างในฐานข้อมูลเพื่อเลือกรูปร่างไหนมีความเหมือนมากที่สุด (best match) ในขณะที่เดียวกันการทำ Ranking จะได้ลำดับของรูปร่างที่มีความคล้ายคลึงกับรูปร่างที่นำมาทดสอบออกมาด้วย ซึ่งจะนำไปวัดประสิทธิภาพในการกู้คืนข้อมูล (retrieval) ต่อไป

ในที่นี้รูปร่างที่นำมาทดสอบนั้นคือรูปร่างที่อยู่ในฐานข้อมูล เพราะฉะนั้นเราจะไม่นำรูปร่างที่มีคะแนนความเหมือนมากที่สุด ($R(\max)=1$) มาพิจารณาแต่จะพิจารณารูปร่างที่มีคะแนนความเหมือนรองลงมาเป็นอันดับที่สอง โดยถ้าหากผลลัพธ์ที่ได้อยู่ในกลุ่มเดียวกันกับรูปร่างที่นำมาทดสอบจะถือว่าการ match นั้นถูกต้อง โดยรูปร่างทั้งหมดที่นำมาทดสอบมีจำนวน 560 รูป ในการหาความแม่นยำในการ match เราจะทำการนับจำนวนครั้งของการ match ที่ถูกต้อง รายละเอียดของการ match และ retrieve ด้วยพารามิเตอร์ ATA และ NATA ที่ความยาวแขนของสามเหลี่ยม I ต่างๆ แสดงดังนี้



รูปที่ 5.6 ขั้นตอนการทดลองของการ match และ retrieval

5.2.1.1.1 ความแม่นยำในการจับคู่รูปร่าง (matching)

เมื่อรูปร่างทดสอบทั้งหมด 560 รูปถูกนำไปวัดความแม่นยำในการ match ด้วยการใช้ 1-D triangle area representation ชนิดต่างๆ ความถูกต้องในการ match เป็นดังนี้

ตารางที่ 5.2 ความแม่นยำในการ match ด้วยการ ใช้ ATA และ NATA ที่ความยาวแขนของ สามเหลี่ยม / ต่างๆ

Triangle Side Length	The 1-D Triangle Area Representation Type					
	ATA			NATA		
	Number	Correct	%Accuracy	Number	Correct	%Accuracy
1	560	530	94.64%	560	549	98.04%
10	560	556	99.29%	560	558	99.82%
20	560	559	99.82%	560	559	99.82%
30	560	559	99.82%	560	559	99.82%
40	560	559	99.82%	560	559	99.82%
49	560	556	99.29%	560	559	99.82%

5.2.1.1.2 ประสิทธิภาพในการกู้คืนรูปร่าง (Retrieval performance)

โดยทั่วไปการวัดความถูกต้องหรือประสิทธิภาพของการกู้คืนข้อมูล (retrieve) สำหรับการสืบค้นข้อมูลในระบบฐานข้อมูลนั้น วัดได้โดย Precision-Recall curve นิยามของ Precision และ Recall มีดังนี้

- Precision คือ อัตราส่วนระหว่างจำนวนรูปร่างที่ Retrieve มาแล้วอยู่ในกลุ่มที่ถูกต้อง กับจำนวนรูปร่างทั้งหมด (m) ที่ Retrieve ดังแสดงในสมการที่ 5.1

$$\text{Precision (Q,m)} = \frac{\text{จำนวนสมาชิกของ set-}\{\text{Retrieve(Q,m)} \cap \text{Relevant(Q)}\}}{m} \quad (5.1)$$

- Recall คือ อัตราส่วน ระหว่างจำนวนรูปร่างที่ Retrieve มาแล้วอยู่ในกลุ่มที่ถูกต้อง กับจำนวนรูปร่างที่จัดอยู่ในกลุ่มเดียวกับรูปร่างที่นำมาทดสอบ-Q (Relevant(Q)) ดังแสดงในสมการที่ 5.2

$$\text{Recall (Q,m)} = \frac{\text{จำนวนสมาชิกของ set-}\{\text{Retrieve(Q,m)} \cap \text{Relevant(Q)}\}}{\text{Relevant(Q)}} \quad (5.2)$$

เมื่อได้คำนวณค่า Precision และ Recall แล้ว นำค่าทั้งหมดมาวาด Precision-Recall curve เพื่อวัดประสิทธิภาพในการ retrieve ของระบบต่อไป โดยแกน x คือค่า Recall แกน y คือค่า Precision โดยจุดที่ตำแหน่งมุมบนซ้ายของ curve แสดงค่า Precision-Recall ที่เหมือนมากที่สุด

เพื่อให้เกิดความเข้าใจยิ่งขึ้นในการหา Precision-Recall curve เราได้แสดงตัวอย่างในการคำนวณค่า Precision Recall และการ plot curve ไว้ดังนี้

ตัวอย่าง สมมติว่าเรานำรูปทดสอบ A และ B มาค้นหารูปร่างในฐานข้อมูลที่มีความเหมือนกับรูป A และ B ซึ่งแต่ละรูปร่าง มีจำนวน Relevant เท่ากับ 4 และในฐานข้อมูลนี้มีอยู่ทั้งหมด 10 รูป หลังจากที่เราได้ทำการวัดความเหมือนและเรียงคะแนนความเหมือนของรูปร่างในฐานข้อมูลแล้ว ความถูกต้องในการกู้คืนรูปร่างแสดงได้ดังนี้

A: ++-+-+-----

+ : correct

B: +-++-+-----

- : incorrect

จากนั้นก็ทำการคำนวณค่า Precision และ Recall ของ A และ B

P(A)	1/1	2/2	2/3	3/4	3/5	4/6	4/7	4/8	4/9	4/10
R(A)	1/4	2/4	2/4	3/4	3/4	4/4	4/4	4/4	4/4	4/4

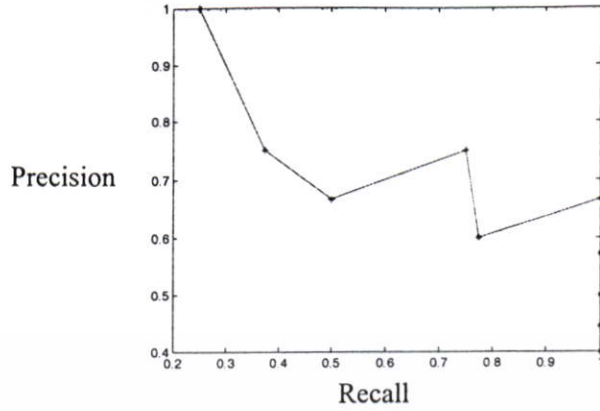
P(B)	1/1	1/2	2/3	3/4	3/5	4/6	4/7	4/8	4/9	4/10
R(B)	1/4	1/4	2/4	3/4	3/4	4/4	4/4	4/4	4/4	4/4

เพื่อดูประสิทธิภาพโดยรวมในการกู้คืนรูปร่างในฐานข้อมูล เราจะนำค่า Precision และ Recall ของ ทั้ง A และ B มาหาค่าเฉลี่ย แล้ว plot ค่าเฉลี่ยเพื่อให้ได้ Precision-Recall curve ดังแสดงในรูปที่ 5.7

$$\text{Average: } \begin{cases} P(AB) = \frac{P(A) + P(B)}{2} \\ R(AB) = \frac{R(A) + R(B)}{2} \end{cases}$$

P(AB)	1	3/4	2/3	3/4	3/5	4/6	4/7	4/8	4/9	4/10
R(AB)	1/4	3/8	2/4	3/4	3/4	1	1	1	1	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.7 ตัวอย่าง Precision-Recall curve

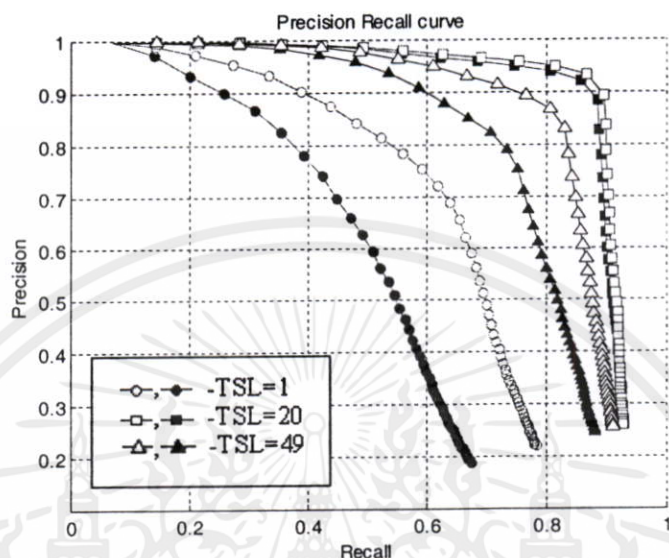
ตารางที่ 5.3 แสดงตัวอย่างผลของการกู้คืน โดยแสดงตัวอย่าง 2 รูปร่างที่ใช้ 1-D triangle area representation ชนิด ATA และ NATA ที่แกนของสามเหลี่ยมเท่ากับ 1 โดยแสดงรูปร่างที่กู้คืนมาได้ 14 อันดับแรกจากรูปร่างทั้งหมด 560 รูปที่มีค่าความเหมือน (NCC) เรียงลำดับจากมากไปน้อย รูปร่างแรกมีความเหมือนมากที่สุดเท่ากับ 1 เพราะคือตัวเอง รูปร่างที่มีเครื่องหมายกากบาทหมายถึงรูปร่างที่กู้คืนมาผิด

ตารางที่ 5.3 ตัวอย่างผลของการกู้คืนรูปร่าง

Queries		Retrieve shapes
	ATA(1)	
	NATA(1)	
	ATA(1)	
	NATA(1)	

สำหรับการวัดประสิทธิภาพของการกู้คืนรูปร่างนั้นทั้ง 560 รูปร่างที่ใช้ Shape representation ชนิด ATA และ NATA ที่ความยาวแกนของสามเหลี่ยมต่างๆ เราได้ทำการวัดด้วย Precision Recall curve ดังที่ได้อธิบายความหมายและวิธีการคำนวณไปแล้วก่อนหน้านี้ โดยในกรณีการทดลองของเราจะมีค่า Relevant เท่ากับ 14 ค่า Precision และ Recall ของทั้ง 560 รูปร่างจะเป็นค่าเฉลี่ยและนำมา plot ดังแสดงในรูปที่ 5.8 ซึ่งเป็น Precision Recall curve ของการใช้ ATA และ NATA ที่แกนของสามเหลี่ยมเท่ากับ 1, 20, และ 49 โดยแต่ละ curve แสดงประสิทธิภาพในการกู้คืนรูปร่าง 50 อันดับแรก (ทั้งนี้ที่เราทำการ plot เพียงบางค่าความยาวแกนของสามเหลี่ยม

(TSL) ที่มีลักษณะเด่นเพื่อสามารถดูแนวโน้มได้ เนื่องจากไม่สามารถแสดงทุกค่าความยาวแขนของสามเหลี่ยมลงไปในกราฟเดียวกันได้เพราะจะทำให้เกิดความยุ่งยากแก่การแยกแยะและวิเคราะห์)



รูปที่ 5.8 Precision Recall curve ที่ใช้วัดประสิทธิภาพในการกู้คืนรูปร่างโดยใช้ ATA (filled symbol) และ NATA (unfilled symbol) ที่แขนของสามเหลี่ยมเท่ากับ 1, 20, และ 49

จากผลการทดลองวัดประสิทธิภาพในการรู้จำรูปร่างที่ไม่ผันแปรต่อการแปลงแบบแอฟไฟน์ด้วยวิธีการ matching โดย Normalized cross-correlation จะเห็นว่ามีค่าความแม่นยำที่สูงมากกว่า 90% ไม่ว่าจะใช้คุณลักษณะแบบ ATA หรือ NATA ที่ความยาวแขนของสามเหลี่ยมใดก็ตาม ในการพิจารณาประสิทธิภาพในการกู้คืนรูปร่าง (Retrieve) จากรูปที่ 5.8 เราพบว่าการใช้ 1-D triangle area representation แบบ NATA ให้ผลที่ดีกว่าการใช้ ATA ที่ทุกแขนของสามเหลี่ยม และยังพบอีกว่าที่ความยาวแขนของสามเหลี่ยมประมาณ 20 ให้ประสิทธิภาพในการกู้คืนรูปร่างที่ดีที่สุด

อย่างไรก็ตามถึงแม้ว่าการรู้จำด้วยวิธี Normalized cross-correlation จะให้ความแม่นยำที่สูง แต่เมื่อพิจารณาถึงขั้นตอนในการรู้จำจะเห็นว่าต้องทำการ match รูปร่างที่นำมาทดสอบกับทุกรูปร่างในฐานข้อมูล ถ้าหากจำนวนรูปร่างในฐานข้อมูลมีจำนวนมากก็จะทำให้เวลาที่ใช้ในขั้นตอนการรู้จำมากตามไปด้วยเพราะต้องทำการเปรียบเทียบกับรูปร่างทั้งหมดในฐานข้อมูล ยิ่งไปกว่านั้นเปอร์เซ็นต์ความถูกต้องของวิธี Normalized cross-correlation ก็ขึ้นอยู่กับจำนวนรูปร่างของแต่ละกลุ่มที่เก็บไว้ในฐานข้อมูลด้วย ถ้าหากแต่ละกลุ่มมีรูปร่างตัวอย่างเก็บไว้มากความแม่นยำก็จะสูงด้วยเหตุนี้ทำให้เราต้องใช้หน่วยความจำในการเก็บข้อมูลเป็นจำนวนมากเพื่อให้ได้เปอร์เซ็นต์ความถูกต้องที่สูง จากปัญหาดังกล่าวทำให้เราพยายามที่จะวิจัยและพัฒนาวิธีการในการรู้จำรูปร่างที่มี

ประสิทธิภาพในการรู้จำที่สูงโดยไม่ใช้เวลาในการรู้จำที่มากและสามารถประหยัดหน่วยความจำด้วยการใช้วิธีโครงข่ายประสาทเทียม (Artificial Neural networks) ดังจะได้กล่าวในหัวข้อถัดไป

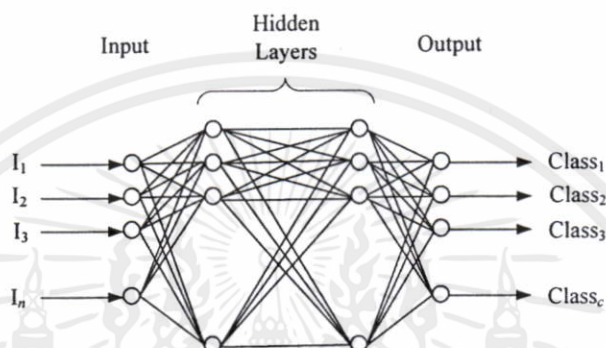
5.2.1.2 การรู้จำด้วยวิธี Artificial Neural networks

ดังที่กล่าวไปแล้วเราต้องการพัฒนาวิธีการในการรู้จำรูปร่างด้วยวิธี Artificial Neural networks ทั้งนี้ที่เราเลือก Neural networks (NN) มาประยุกต์ใช้ในการรู้จำเนื่องจากคุณสมบัติของ Neural networks นั้นสามารถที่จะทำให้ระบบรู้จำมีประสิทธิภาพถึงแม้ว่าจะเจอเงื่อนไขหรือสถานะการที่แตกต่างไปจากสิ่งที่ Neural networks (NN) ได้เรียนรู้แต่ network ก็พยายามประมวลผลข้อมูลให้ได้ผลที่ใกล้เคียงมากที่สุด ซึ่งเราสามารถทำการสอนให้ Neural network เรียนรู้จดจำรูปแบบของรูปร่างภายในสภาวะต่างๆกันได้มากมายเท่าที่เราต้องการได้ ด้วยการสร้างลักษณะโครงสร้างของ Neural network ที่เหมาะสมจะทำให้ Neural network สามารถรวบรวมรูปแบบของรูปร่างในแต่ละกลุ่มที่ได้ถูกฝึกสอนเอาไว้ภายในโครงสร้าง จึงทำให้ขั้นตอนการรู้จำเหลือเพียงขั้นตอนเดียวคือใส่ input เข้าสู่โครงสร้างที่เดียวก็ได้ output ออกมา ไม่จำเป็นต้องทำการเปรียบเทียบหลายครั้งเหมือนในวิธี Normalized cross-correlation ดังนั้น Neural network สามารถระบุได้ว่ารูปร่างอยู่ในกลุ่มไหนภายในขั้นตอนเดียวจึงสามารถลดปัญหาการสูญเสียเวลาในการรู้จำลงไปได้ และนอกนั้นยังช่วยประหยัดหน่วยความจำได้ดังที่กล่าวไว้แล้วว่า Neural network จดจำรูปแบบสิ่งที่เรียนรู้ไว้ในโครงสร้างของตัวเองด้วยการฝึกสอนที่เหมาะสมและเพียงพอเพราะฉะนั้นจึงไม่จำเป็นต้องเก็บรูปร่างไว้ในฐานข้อมูลเพื่อใช้สำหรับการเปรียบเทียบ

สำหรับขั้นตอนการทดลองเราได้สร้าง Neural networks ที่มีโครงสร้างดังแสดงในรูปที่ 5.9 เพื่อใช้วัดประสิทธิภาพในการรู้จำของ 1-D triangle area representation แบบ ATA และ NATA ที่ความยาวแขนของสามเหลี่ยมต่างๆ โดย 1-D triangle area representation จะถูกนำมาเป็น input ของ Neural network ซึ่งมีจำนวนนิวรอนใน input layer 100 node ตามจำนวนองค์ประกอบภายใน feature vector ของ 1-D triangle area representation สำหรับ Hidden layer มี 2 ชั้น ในแต่ละชั้น มีจำนวนนิวรอนชั้น ละ 70 node ส่วนใน output layer มีจำนวนนิวรอน 40 node ตามจำนวนกลุ่มของรูปร่าง (Class) ส่วนฟังก์ชันการตัดสินใจที่ใช้ในโครงสร้างทั้งหมดเป็นฟังก์ชันแทนซิกมอยด์ และโครงสร้างมีการฝึกสอนแบบ Scaled Conjugate Gradient เพื่อดูประสิทธิภาพของ Neural networks ที่ถูกฝึกสอนด้วยจำนวนข้อมูลที่แตกต่างกันคือ Neural networks ที่ถูกฝึกสอนด้วยข้อมูลจำนวนน้อยๆและ Neural networks ที่ถูกฝึกสอนด้วยข้อมูลจำนวนมากๆ เราได้เตรียมข้อมูลไว้ฝึกสอน Neural networks 2 ชุดคือ

1. กลุ่มข้อมูลขนาดเล็กที่ได้มาจาก Shape representation ของรูปร่างทั้งหมด 40 กลุ่ม กลุ่มละ 1 ตัวอย่าง (รวมมีข้อมูลที่ใช้ฝึกสอนทั้งหมด 40 ตัวอย่าง)
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามคัดลอกข้อมูลนี้ไปและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
2. กลุ่มข้อมูลขนาดใหญ่ที่ได้มาจาก Shape representation ของรูปร่างทั้งหมด 40 กลุ่ม กลุ่มละ 7 ตัวอย่าง (รวมมีข้อมูลที่ใช้ฝึกสอนทั้งหมด 280 ตัวอย่าง)

สำหรับ Shape representation ของรูปร่างที่เหลือนในแต่ละกลุ่มข้อมูลจะถูกนำมาทดสอบประสิทธิภาพการรู้จำ ผลการทดลองการรู้จำรูปร่างด้วยวิธี Neural networks ที่ถูกสอนด้วยจำนวนข้อมูลที่แตกต่างกันแสดงดังตารางที่ 5.4 ซึ่งประเภทของข้อมูลที่ใช้ฝึกสอนคือ 1-D triangle area representation แบบ ATA และ NATA ที่ความยาวแขนของสามเหลี่ยมต่างๆ โดยได้ทำการเปรียบเทียบประสิทธิภาพในการรู้จำของ Neural networks ที่ได้จากการฝึกสอนด้วยจำนวนข้อมูลที่แตกต่างกันดังกล่าวดังแสดงในตารางที่ 5.4 และรูปที่ 5.10

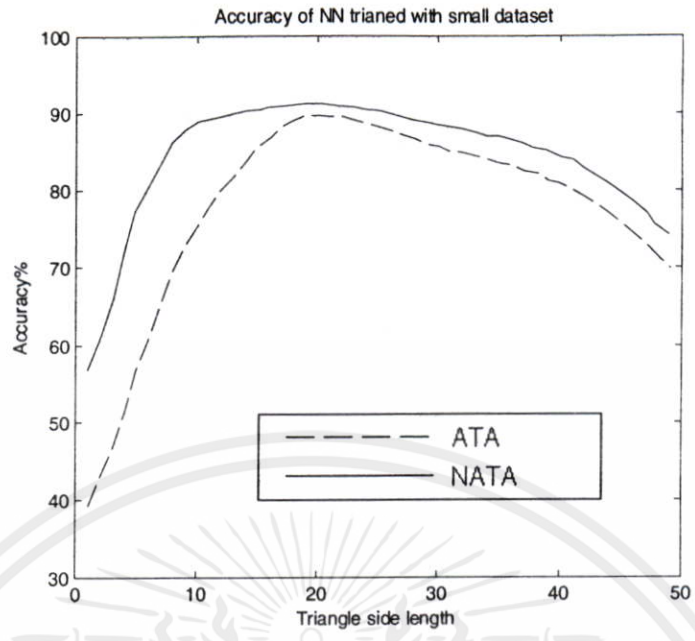


รูปที่ 5.9 โครงสร้างของ Artificial Neural network ที่ใช้ในการรู้จำ

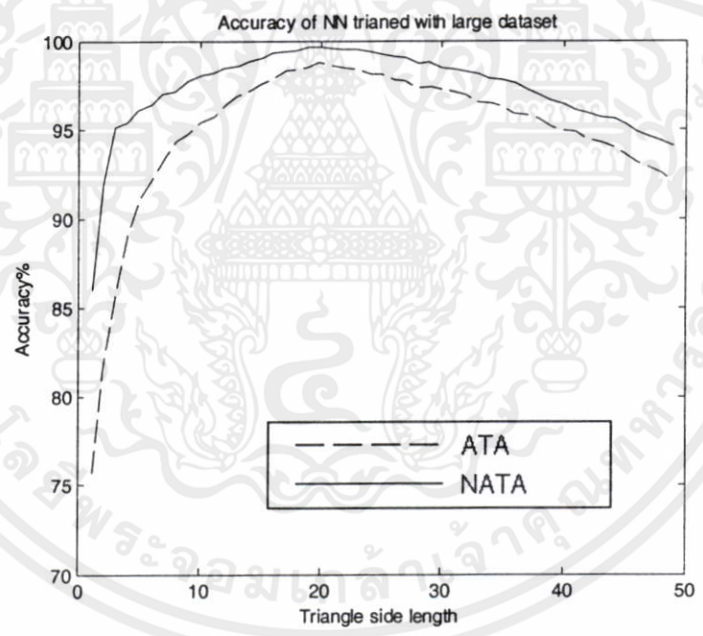
ตารางที่ 5.4 ความถูกต้องในการรู้จำด้วยวิธี NN ที่ถูกฝึกสอนด้วยขนาดและประเภทของข้อมูลต่างๆ

Triangle side length	Accuracy of NN%			
	Trained with small dataset		Trained with large dataset	
	ATA	NATA	ATA	NATA
1	39.42	56.73	75.71	86.07
3	46.54	66.12	86.07	95.36
5	56.73	77.12	91.07	97.14
10	75.38	88.85	95.36	98.57
20	89.81	91.15	98.57	99.64
30	85.58	88.46	97.14	98.57
40	80.77	84.42	95.71	96.43
49	69.81	74.04	92.14	94.29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ก)



ข)

รูปที่ 5.10 ประสิทธิภาพในการรู้จำรูปร่างด้วยวิธี Artificial Neural networks ที่ถูกฝึกสอนด้วยขนาดข้อมูลที่แตกต่างกัน ก) ฝึกสอนด้วยชุดข้อมูลขนาดเล็ก และข) ฝึกสอนด้วยชุดข้อมูลขนาดใหญ่ นอกจากนั้นยังเป็นการเปรียบเทียบประสิทธิภาพในการใช้ประเภทของ

Shape representation ที่แตกต่าง ATA กับ NATA ที่ความยาวแกนของสามเหลี่ยม 1-49 เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาติให้เผยแพร่เชิงพาณิชย์ การค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้ จากผลการทดลองในตารางที่ 5.4 และรูปที่ 5.10 แสดงให้เห็นว่า Neural networks ที่ถูกฝึกสอนด้วยสภาวะที่หลากหลายหรือจำนวนข้อมูลมากจะมีความสามารถในการรู้จำรูปร่างได้ดีกว่า

Neural networks ที่ถูกฝึกสอนด้วยสภาวะที่ถูกจำกัดแค่ตัวอย่างเดียวต่อรูปร่าง 1 ชนิด แต่ทั้งนี้เวลาในการฝึกสอนด้วยกลุ่มข้อมูลขนาดใหญ่ย่อมใช้เวลานานกว่าการฝึกสอนด้วยกลุ่มข้อมูลขนาดเล็ก สำหรับการพิจารณาว่า 1-D triangle area representation ประเภทใดให้ประสิทธิภาพที่ดี เมื่อสังเกตจากผลการทดลองของทั้ง Neural networks ที่ถูกฝึกสอนด้วยจำนวนข้อมูลที่มีขนาดเล็กและใหญ่ พบว่า NATA ให้ประสิทธิภาพในการรู้จำที่ดีกว่า ATA นอกจากนั้นแล้วยังพบว่าที่ความยาวแขนของสามเหลี่ยมประมาณ 20 ให้ประสิทธิภาพในการรู้จำที่ดีที่สุด

5.2.1.3 การเปรียบเทียบประสิทธิภาพในการรู้จำระหว่างวิธี Normalized cross-correlation และ วิธี Neural networks

ในหัวข้อนี้เป็นการทดลองเพื่อเปรียบเทียบประสิทธิภาพในการรู้จำระหว่างวิธี Normalized cross-correlation และวิธี Neural networks แล้วทำการวิเคราะห์ข้อดีข้อเสียของแต่ละวิธี เนื่องจากทั้งสองวิธีมีขั้นตอนในการรู้จำที่ต่างกันคือ วิธี Normalized cross-correlation นั้นเป็นการนำเอารูปร่างที่ต้องการทดสอบมาทำการเปรียบเทียบกับรูปร่างที่เก็บไว้ในฐานข้อมูล ส่วนวิธี Neural networks นั้นเปรียบเสมือนการนำเอารูปร่างที่ต้องการทดสอบมาคำนวณด้วยฟังก์ชันการรู้จำที่ถูกสร้างมาจากการฝึกสอนด้วยชุดข้อมูลจากรูปร่างต้นแบบจำนวนหนึ่ง เพราะฉะนั้นถ้าหากต้องการเปรียบเทียบประสิทธิภาพในการรู้จำระหว่าง 2 วิธีดังกล่าวให้อยู่ในมาตรฐานเดียวกัน จะต้องใช้ชุดข้อมูลเดียวกัน โดยมีจำนวนรูปร่างที่เก็บไว้ในฐานข้อมูลสำหรับวิธี Normalized cross-correlation และจำนวนรูปร่างต้นแบบที่ใช้ในการฝึกสอน Neural networks ที่เท่ากัน ในการทดลองรูปร่างที่นำมาเก็บไว้ในฐานข้อมูลและที่ใช้ฝึกสอน Neural networks เป็นข้อมูลชุดเดียวกันแบ่งเป็น 2 ประเภทคือกลุ่มข้อมูลชุดเล็กที่มีจำนวนรูปร่างต้นแบบอยู่ 40 รูป (ชนิดละ 1 ตัวอย่าง) และกลุ่มข้อมูลชุดใหญ่ที่มีจำนวนรูปร่างต้นแบบ 280 รูป (ชนิดละ 7 ตัวอย่าง) โดยชนิดของ Shape representation ที่ใช้ในการทดลองนี้คือ NATA ผลการทดลองแสดงดังตารางที่ 5.5 และรูปที่ 5.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.5 การเปรียบเทียบประสิทธิภาพในการรู้จำระหว่างวิธี NCC และวิธี NN ที่ใช้ชุดข้อมูลขนาดเล็กและขนาดใหญ่ในการ match สำหรับวิธี NCC และการฝึกสอน NN โดยทดสอบกับ NATA ที่ความยาวแขนของสามเหลี่ยมต่างๆ

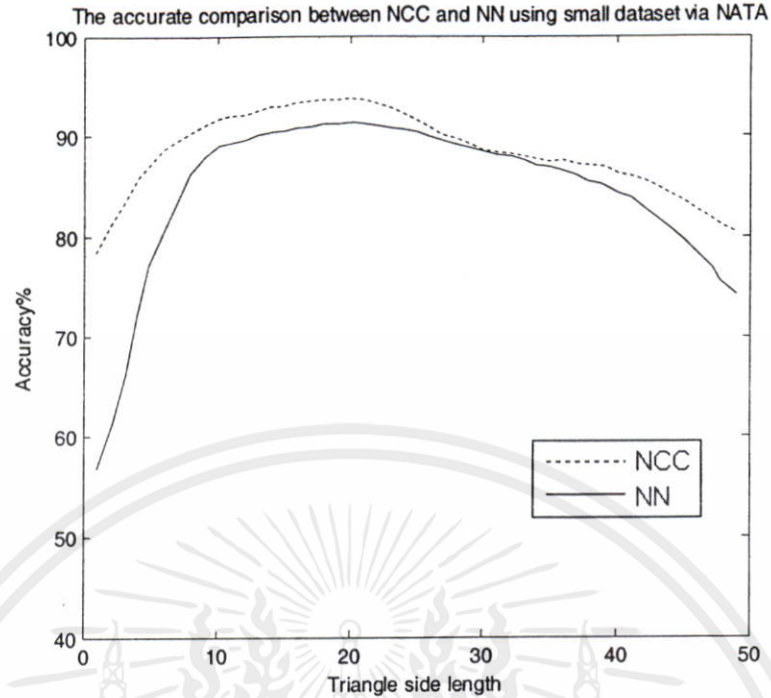
NATA(<i>l</i>)	Accuracy of NCC%		Accuracy of NN%	
	Matched with small dataset	Matched with large dataset	Trained with small dataset	Trained with small dataset
1	78.46	96.43	56.73	86.07
3	83.46	97.14	66.12	95.36
5	87.31	98.57	77.12	97.14
10	91.54	99.29	88.85	98.57
20	93.65	99.64	91.15	99.64
30	88.65	98.93	88.46	98.57
40	86.35	98.21	84.42	96.43
49	80.19	97.86	74.04	94.29

จากผลการทดลองในตารางที่ 5.5 และรูปที่ 5.11 จะเห็นว่าการรู้จำรูปร่างด้วยวิธี Normalized cross-correlation ให้ประสิทธิภาพที่ดีกว่าวิธี Neural networks ทั้งในส่วนของการใช้ชุดข้อมูลขนาดใหญ่และชุดข้อมูลขนาดเล็ก ทั้งนี้เนื่องจากวิธี Normalized cross-correlation เป็นวิธีที่ละเอียด ซึ่งในขั้นตอนการระบุรูปร่างแต่ละครั้ง รูปร่างนั้นจะถูกนำไปเปรียบเทียบกับรูปร่างที่อยู่ในฐานข้อมูลทีละรูปร่างเพื่อวัดความเหมือน ด้วยการเปรียบเทียบและคำนวณหลายขั้นตอนนี้จึงทำให้ความถูกต้องในการรู้จำสูง ในขณะที่วิธี Neural networks นั้นใช้แค่ขั้นตอนเดียวในการระบุรูปร่างคือ นำ Shape representation ของรูปร่างนั้นป้อนเข้าสู่ Neural networks ที่ผ่านกระบวนการสร้างและฝึกสอน อย่างไรก็ตามถ้าหากกล่าวถึงในด้านความเร็วในการระบุรูปร่างนั้นวิธี Neural networks ย่อมเร็วกว่าวิธี Normalized cross-correlation เพราะไม่จำเป็นต้องทำการเปรียบเทียบหลายครั้งเพื่อให้ได้คำตอบ โดยเราได้ทำการทดลองวัดเปรียบเทียบความเร็วในการระบุรูปร่างของทั้งสองวิธีด้วยโปรแกรม MATLAB บนเครื่องคอมพิวเตอร์ CPU AMD Athlon(tm)64X2 Dual Core 2 GHz และ RAM 1 GB พบว่าเวลาในการระบุรูปร่างหนึ่งรูปร่างของวิธี Neural networks ใช้ไป 83.93 μs ซึ่งน้อยกว่าวิธี Normalized cross-correlation ที่ใช้ไปถึง 564 μs สำหรับการ match กับชุดข้อมูลขนาดเล็ก (นำไปเปรียบเทียบกับรูปร่างในฐานข้อมูลที่มีจำนวน 40 รูปร่าง) และ 39.579x10² μs สำหรับการ match กับชุดข้อมูลขนาดใหญ่ (นำไปเปรียบเทียบกับรูปร่างใน

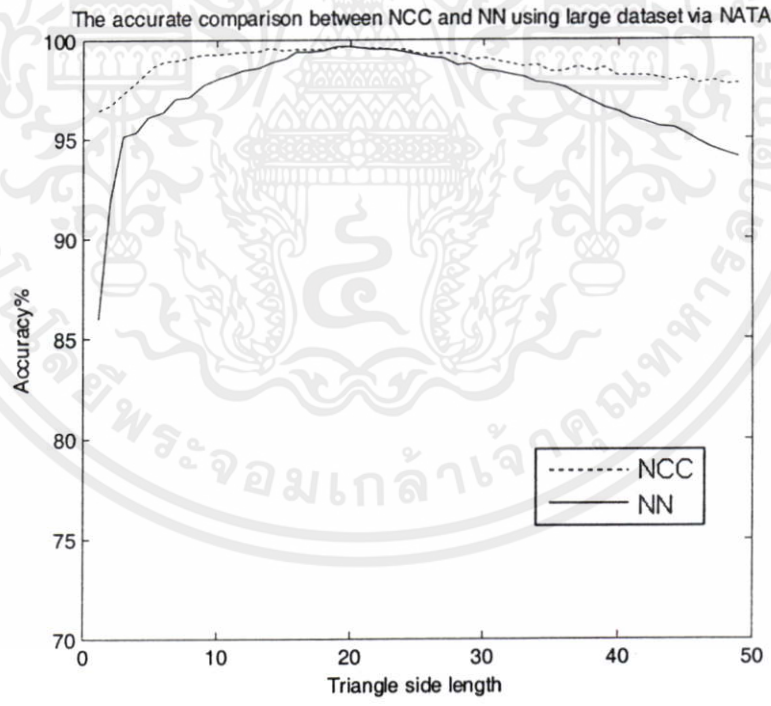
ฐานข้อมูลที่มีจำนวน 280 รูปร่าง) เพราะฉะนั้นถ้าต้องการสร้างระบบการรู้จำที่ใช้แยกแยะรูปร่างที่มีหลายชนิดหรือมีจำนวน class ที่มาก วิธี Normalized cross-correlation จึงไม่เหมาะสมเพราะต้องทำการเปรียบเทียบกับทุกรูปร่างในฐานข้อมูลซึ่งมีจำนวนมากทำให้ใช้เวลานานในการระบุรูปร่าง ซึ่งกล่าวได้ว่าวิธี Neural networks มีความเหมาะสมมากกว่า

สำหรับความแตกต่างของประสิทธิภาพในการรู้จำระหว่างการใช้ชุดข้อมูลขนาดเล็กและชุดข้อมูลขนาดใหญ่ด้วยทั้งสองวิธี จากผลการทดลองในตารางที่ 5.5 และรูปที่ 5.11 พบว่าการใช้ชุดข้อมูลขนาดใหญ่จะให้ประสิทธิภาพในการรู้จำที่ดีกว่าการใช้ชุดข้อมูลขนาดเล็ก ทั้งนี้เมื่อพิจารณาหลักการของวิธี Normalized cross-correlation การที่มีจำนวนข้อมูลในแต่ละ class น้อย ในกรณีนี้จะมีความเสี่ยงในระบุรูปร่างผิดสูง เช่นสมมติว่ารูปร่างหนึ่งมาเปรียบเทียบกับรูปร่างที่อยู่ใน class เดียวกันในฐานข้อมูลซึ่งมีแค่ 1 รูปร่างต่อหนึ่ง class ถ้าหากการเปรียบเทียบครั้งนี้มีค่าความเหมือนน้อยกว่า class อื่นก็ถือว่ากระบุรูปร่างครั้งนั้นผิดไปเลย ในขณะที่หากเพิ่มจำนวนรูปร่างต่อ class ให้มากขึ้น รูปร่างนี้อาจจะไปเหมือนกับรูปร่างอื่นๆที่อยู่ใน class เดียวกันและมีคะแนนความเหมือนมากที่สุดเมื่อเปรียบเทียบกับ class อื่นๆก็ได้ เพราะฉะนั้นการที่ฐานข้อมูลมีจำนวนรูปร่างในแต่ละ class มากจึงเท่ากับเป็นการเพิ่มโอกาสหรือความน่าจะเป็นที่จะทำให้การรู้จำรูปร่างมีความถูกต้องสูง ในส่วนหลักการของวิธี Neural networks นั้นการฝึกสอน networks ด้วยข้อมูลจำนวนมากเท่ากับเป็นการเพิ่มข้อมูลให้ Neural networks มีการเรียนรู้ที่หลากหลาย จึงทำให้สามารถประมวลผลได้อย่างถูกต้องไม่ว่าจะเจอเงื่อนไขที่แตกต่างไปจากสิ่งที่ networks เรียนรู้ก็ตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ก)



ข)

รูปที่ 5.11 การเปรียบเทียบประสิทธิภาพในการรู้จำระหว่างวิธี NCC และวิธี NN ที่ใช้ 2 ชุดข้อมูล เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานาน นโมอนุชาต ให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น ยกเว้นกรณีเกิดขบถแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มาไปใช้ สามเหลี่ยม 1-49

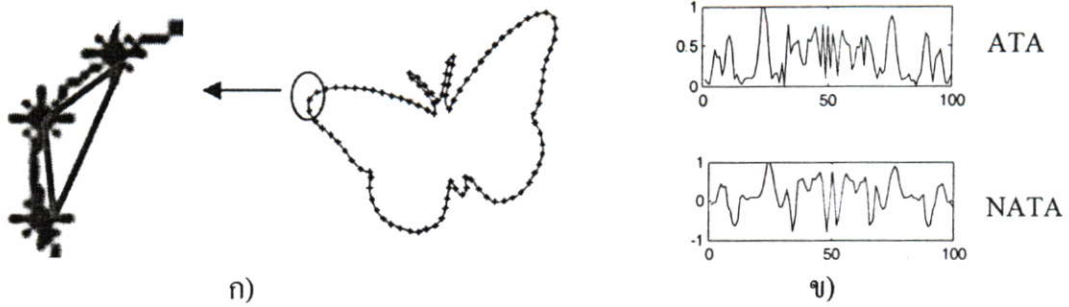
5.2.1.4 การวิเคราะห์ทำนายประสิทธิภาพในการแยกแยะข้อมูลของ 1-D triangle area representation ด้วย Fisher's discriminant

ในหัวข้อที่ผ่านมา ผลการทดลองการวัดประสิทธิภาพในการรู้จำที่ใช้ 1-D triangle area representation แบบ ATA และ NATA ที่ความยาวแขนของสามเหลี่ยมต่างๆ แสดงให้เห็นว่าการใช้ NATA ให้ประสิทธิภาพในการรู้จำที่ดีกว่า ATA และที่ความยาวแขนของสามเหลี่ยมต่างๆ ก็ให้ประสิทธิภาพในการรู้จำที่แตกต่างกันออกไป ในหัวข้อนี้เราจะทำการวิเคราะห์ถึงสาเหตุที่ทำให้ผลการทดลองออกมาเป็นแบบนั้น และทำการวิเคราะห์ทำนายประสิทธิภาพในการแยกแยะข้อมูลของ 1-D triangle area representation ด้วย Fisher's discriminant

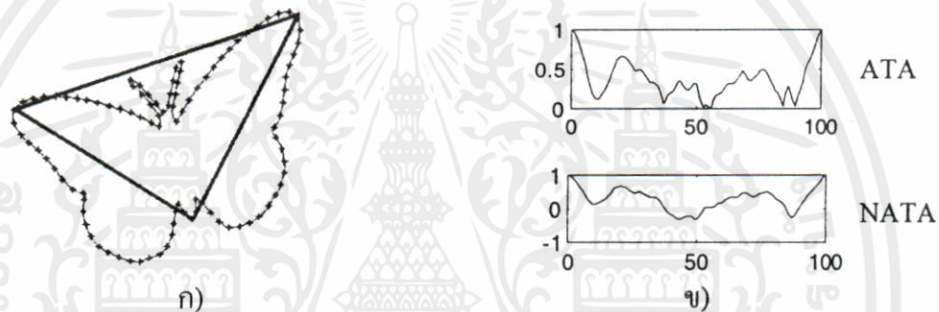
ปัจจัยที่ทำให้การรู้จำรูปร่างที่ใช้ NATA มีประสิทธิภาพที่ดีกว่า ATA นั้น เนื่องจาก NATA ให้รายละเอียดของลักษณะเฉพาะของแต่ละรูปร่างที่มากกว่า ATA โดยเมื่อพิจารณาถึงขั้นตอนการทำ ATA และ NATA ในหัวข้อที่ 3.4.1 และ 3.4.2 แล้วนั้น จะเห็นว่าค่าสัมประสิทธิ์ของ NATA ที่คำนวณมาจากคิเทอร์มีแนนในสมการที่ 3.7 มีค่าทั้งบวกและลบ ซึ่งค่าบวกและค่าลบของค่าคิเทอร์มีแนนนี้แสดงถึงความเป็นส่วนเว้าหรือส่วนนูนของแต่ละจุดบนเส้นรอบรูป ในขณะที่สัมประสิทธิ์ของ ATA มีเฉพาะค่าบวก การคำนวณค่า absolute นี้ทำให้ได้ข้อมูลที่มีเฉพาะค่าบวก ซึ่งเท่ากับเป็นการลดข้อมูลหรือรายละเอียดของลักษณะเฉพาะในแต่ละรูปร่างลง จึงทำให้การรู้จำที่ใช้ ATA มีประสิทธิภาพน้อยกว่า NATA ดังแสดงผลมาแล้วในรูปที่ 5.8 และ 5.10 และในตารางที่ 5.4

ในส่วนของการวิเคราะห์ผลของความยาวแขนของสามเหลี่ยมต่างๆ ที่ให้ประสิทธิภาพในการรู้จำที่แตกต่างกันและที่ความยาวแขนของสามเหลี่ยมเท่าไรให้ความถูกต้องในการรู้จำที่สูงที่สุด ก่อนอื่นเราจะทำการวิเคราะห์แนวโน้มเปอร์เซ็นต์ความถูกต้องในการรู้จำที่แขนของสามเหลี่ยมต่างๆ ของทั้ง ATA และ NATA จากรูปที่ 5.10 และ 5.11 จะเห็นว่าที่แขนของสามเหลี่ยมที่ต่ำเกินไปหรือสูงเกินไปจะให้ประสิทธิภาพในการรู้จำที่ต่ำกว่าที่แขนของสามเหลี่ยมประมาณกลางๆ หรือประมาณ 20 และยังพบอีกว่าที่ความยาวแขนของสามเหลี่ยมเท่ากับ 1 ให้ประสิทธิภาพในการรู้จำที่ต่ำที่สุด ทั้งนี้เนื่องจากที่ความยาวแขนของสามเหลี่ยมต่างๆ จะมีผลกระทบของ noise มากดังแสดงในรูปที่ 5.12 ในทางตรงกันข้ามที่ความยาวแขนของสามเหลี่ยมสูงๆ จะทำให้ขาดรายละเอียดของรูปร่างดังแสดงในรูปที่ 5.13 ด้วยเหตุนี้จึงทำให้ประสิทธิภาพในการรู้จำของทั้งสองกรณีต่ำ ความยาวแขนของสามเหลี่ยมที่ให้ประสิทธิภาพในการรู้จำที่ดีจะอยู่ที่ประมาณ 20 เพราะสามารถลดผลกระทบจาก noise และยังสามารถเก็บรายละเอียดของรูปร่างได้อย่างเหมาะสม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.12 แสดงตัวอย่างการพิจารณาหา 1-D triangle area representation ที่แขนของสามเหลี่ยมต่ำๆ (ก) ที่ความยาวแขนของสามเหลี่ยมเท่ากับ 1 และข) แสดง ATA และ NATA ที่คำนวณมาจากแขนของสามเหลี่ยมเท่ากับ 1



รูปที่ 5.13 แสดงตัวอย่างการพิจารณาหา 1-D triangle area representation ที่แขนของสามเหลี่ยมสูงๆ (ก) ที่ความยาวแขนของสามเหลี่ยมเท่ากับ 49 และข) แสดง ATA และ NATA ที่คำนวณมาจากแขนของสามเหลี่ยมเท่ากับ 49

จากการวิเคราะห์ผลการทดลอง นอกจากการรู้จำรูปร่างที่มีประสิทธิภาพจะได้มาจากการใช้ 1-D triangle area representation ที่สามารถเก็บรายละเอียดของแต่ละรูปร่างได้อย่างเหมาะสมแล้ว ที่สำคัญยิ่งกว่านั้นคือ 1-D triangle area representation ที่ดีนั้นควรจะมีลักษณะที่ทำให้สามารถแยกแยะรูปร่างแต่ละ class ออกจากกันได้อย่างง่ายหรือมีความสามารถแสดงความแตกต่างระหว่างลักษณะเฉพาะของแต่ละรูปร่างได้อย่างชัดเจน ที่ผ่านมาระบุได้ว่า 1-D triangle area representation แบบ ATA หรือ NATA และที่ความยาวแขนของสามเหลี่ยมไหน เป็น Shape representation ที่ดีจากการพิจารณาผลการทดลองซึ่งเป็นการบอกทางอ้อมว่า 1-D triangle area representation แบบใดมีความสามารถในการแยกแยะข้อมูล (high discriminant power) ได้ดีต่อไปนี้จะทำการวัดค่าของความสามารถในการแยกแยะข้อมูลออกมาเป็นตัวเลขโดยตรงด้วยทฤษฎี Fisher's discriminant เพื่อพิจารณาว่า 1-D triangle area representation แบบ ATA และ NATA และที่ความยาวแขนของสามเหลี่ยมต่างๆ มีความสามารถในการแยกแยะข้อมูล (Fisher's

discriminant value) มากน้อยเพียงใดและค่าดังกล่าวจะสอดคล้องกับผลการทดลองของประสิทธิภาพในการรู้จำหรือไม่

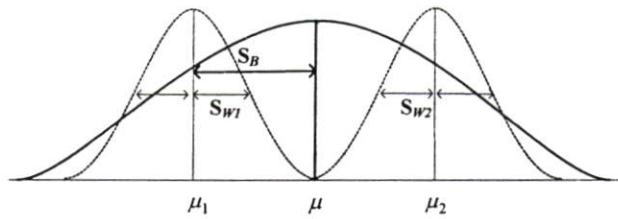
Fisher's Linear Discriminant (FLD) เป็นวิธีการที่ใช้กันอย่างแพร่หลายในการแยกลักษณะเด่นและลคมิติของข้อมูลในงานทางด้านความรู้จำรูปแบบ แนวคิดของ Fisher คือการหาทิศทางของการโปรเจกต์ (หรือ Basis) W ที่ให้ความแตกต่างของค่าเฉลี่ยของกลุ่มได้ดี (เมื่อโปรเจกต์ข้อมูลลงบนทิศทาง W) ในขณะที่ให้ค่าความแปรปรวน (variance) รอบค่าเฉลี่ยของกลุ่มเหล่านั้นมีค่าน้อยๆ เมื่อกล่าวในทางคณิตศาสตร์ FLD จะทำการเลือกทิศทางการโปรเจกต์ W จนกระทั่งทำให้อัตราส่วนของเมตริกซ์การกระจายระหว่างกลุ่ม (between-class scatter matrix) และเมตริกซ์การกระจายภายในกลุ่ม (within-class scatter matrix) มีค่ามากที่สุด [43]

สำหรับในงานวิจัยนี้เรานำ FLD มาประยุกต์ใช้ในการวัดความสามารถในการแยกแยะข้อมูลของ 1-D triangle area representation โดยไม่ต้องทำการหาทิศทางของการโปรเจกต์ตามขั้นตอนทั้งหมดของวิธี Fisher's discriminant ทั่วไป [43] ซึ่งเราใช้แค่อัตราส่วนของ fisher ในที่นี้ก็คืออัตราส่วนระหว่างความแปรปรวนระหว่างกลุ่มและความแปรปรวนภายในกลุ่ม ดังแสดงในสมการที่ 5.3

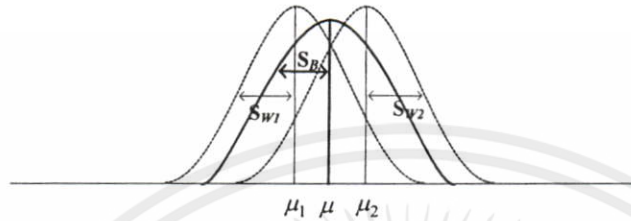
$$\text{Fisher's ratio} = \frac{S_B}{S_W} = \frac{\sum_i^c (\mu_i - \mu)^2}{\sum_i^c \sum_{x_k \in X_i} (x_k - \mu_i)^2} \quad (5.3)$$

เมื่อ c คือจำนวนกลุ่มของข้อมูล μ_i คือ ข้อมูลเฉลี่ยของกลุ่ม X_i , μ คือ ข้อมูลเฉลี่ยของทุกกลุ่ม โดยถ้าอัตราส่วนนี้มีค่ามากแสดงว่าข้อมูลหรือ shape representation แบบนั้นสามารถแยกแยะข้อมูลได้ดี เพื่อความเข้าใจมากยิ่งขึ้นว่าทำไมค่าอัตราส่วนที่มากแสดงถึงความมีประสิทธิภาพในการแยกแยะข้อมูล เราได้แสดงภาพของความแปรปรวนระหว่างกลุ่ม S_B และความแปรปรวนภายในกลุ่ม S_W ในรูปที่ 5.14 อัตราส่วนของ fisher ที่มากเกิดขึ้นเมื่อ S_B มีค่ามากกว่า S_W ยิ่ง Fisher's ratio มากแสดงว่าข้อมูลระหว่างกลุ่มมีความแตกต่างกันมากเพราะฉะนั้นจึงสามารถแยกแยะข้อมูลระหว่างกลุ่มได้ง่าย เมื่อ S_W มีค่าน้อยแสดงให้เห็นว่าข้อมูลภายในกลุ่มมีความแปรปรวนที่น้อยซึ่งเป็นหนึ่งในคุณสมบัติของ shape representation ที่ดี

เราได้นำ 1-D triangle area representation แบบ ATA และ NATA และที่แกนของเอกสารนี้สามเหลี่ยมต่างๆมาวัดค่า Fisher's ratio ตามสมการที่ 5.3 แล้วนำมา plot กราฟโดยแกน x คือการคำนวณว่ากรณี sampled points และแกน y คือค่า Fisher's ratio ดังแสดงในรูปที่ 5.15 เอกสารทุกครั้งที่มีการนำไปใช้

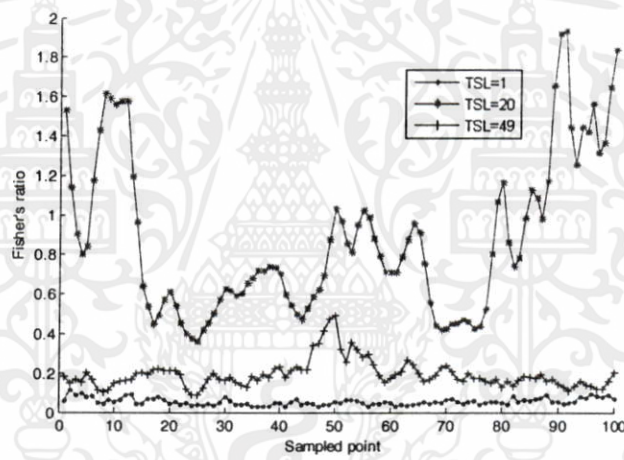


ก) Good discriminant

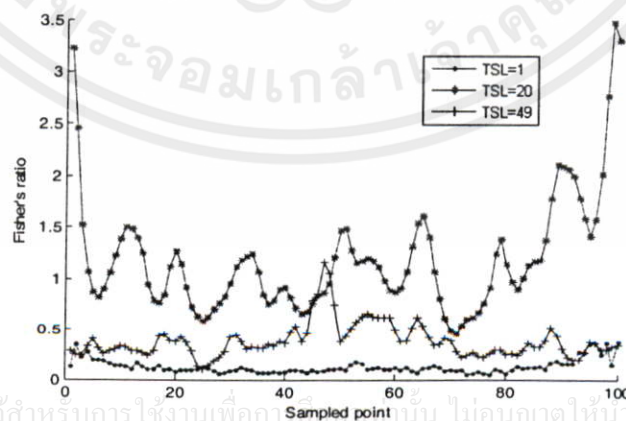


ข) Bad discriminant

รูปที่ 5.14 Fisher's discriminant



ก) ATA



ข) NATA

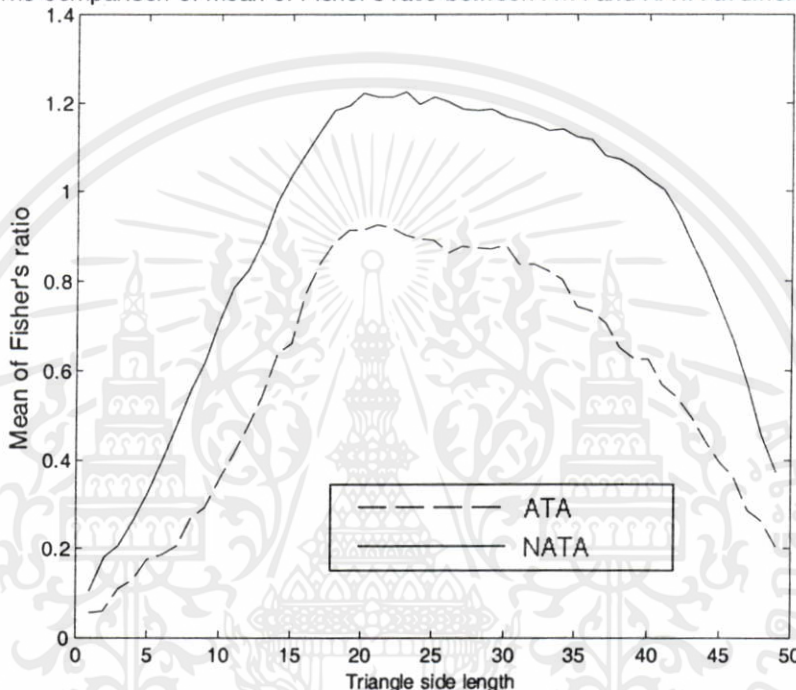
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และเผยแพร่ไปยังเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.15 แสดงค่า Fisher's ratio ของ ก) ATA และข) NATA ที่แกนของสามเหลี่ยม 1, 20, และ 49

เพื่อให้ง่ายต่อการพิจารณาว่า 1-D triangle area representation แบบไหนและที่ความยาวแขนของสามเหลี่ยมใดจะมีประสิทธิภาพในการแยกแยะข้อมูลได้ดี เราได้ทำการหาค่าเฉลี่ยของค่า Fisher's ratio จากทุก sampled point แล้วนำมา plot กราฟโดยแกน x คือความยาวแขนของสามเหลี่ยมใดๆ แกน y คือค่าเฉลี่ยของค่า Fisher's ratio ดังแสดงในรูปที่ 5.16

The comparison of mean of Fisher's ratio between ATA and NATA at different TSL



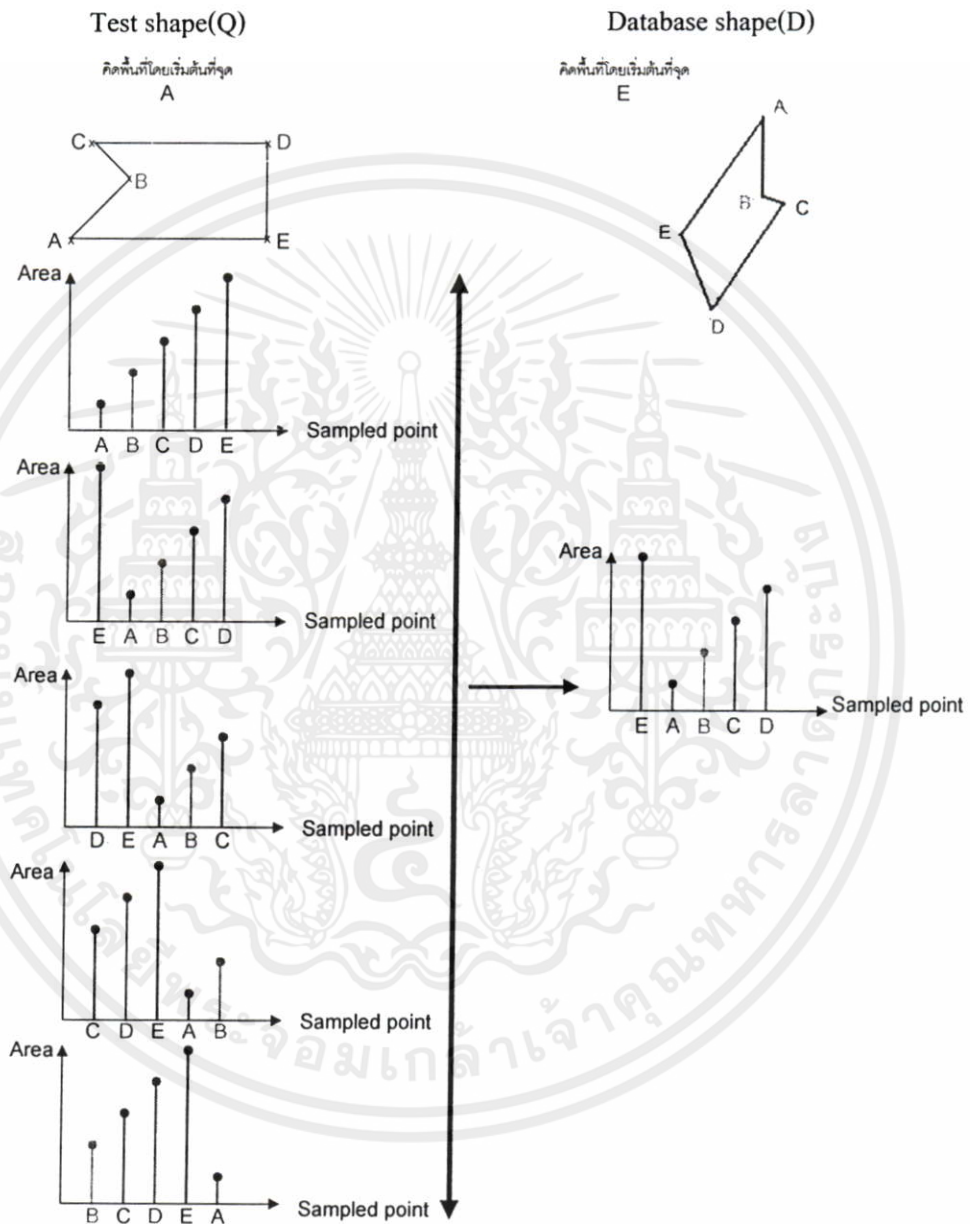
รูปที่ 5.16 เปรียบเทียบประสิทธิภาพในการแยกแยะข้อมูลด้วยค่าเฉลี่ย Fisher's ratio ของ ATA และ NATA ที่ความยาวแขนของสามเหลี่ยม 1-49

จากรูปที่ 5.15 และ 5.16 แสดงให้เห็นว่า NATA มีประสิทธิภาพในการแยกแยะข้อมูลได้ดีกว่า ATA และที่ความยาวแขนของสามเหลี่ยมที่ต่ำเกินไปหรือสูงเกินไปจะทำให้ประสิทธิภาพในการแยกแยะต่ำลง ที่ความยาวแขนของสามเหลี่ยมประมาณ 20 ให้ประสิทธิภาพในการแยกแยะที่ดีที่สุด ซึ่งสอดคล้องกับผลการทดลองวัดประสิทธิภาพในการรู้จำที่แสดงในรูปที่ 5.10 และ 5.11

5.2.2 ประสิทธิภาพการรู้จำในด้านความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้น

จากหัวข้อที่ผ่านมาเป็นการวัดประสิทธิภาพในการรู้จำในด้านความคงทนต่อการแปลงแบบแอฟเฟนโดยกำหนดจุดเริ่มต้นที่แน่นอน ในหัวข้อนี้เราจะทำการพัฒนาวิธีการรู้จำทั้งวิธี Normalized cross-correlation และวิธี Neural networks ให้สามารถรู้จำรูปร่างได้โดยสามารถทนการเปลี่ยนแปลงจุดเริ่มต้น

เริ่มด้วยการพิจารณาลักษณะของ 1-D triangle area representation ที่หาจากการกำหนดจุดเริ่มต้นที่ต่างกันดังแสดงรูปที่ 5.17 ด้านซ้าย จะเห็นว่าพื้นที่สามเหลี่ยมของแต่ละ sampled points หลังจากการทำ normalized แล้วมีค่าเท่าเดิมเพราะฉะนั้นลักษณะ 1-D triangle area representation ของรูปร่างหนึ่งที่มาจากจุดเริ่มต้นต่างกันจึงมีลักษณะที่เลื่อนตำแหน่งไปเท่านั้น



รูปที่ 5.17 หลักการในการพัฒนาวิธีการให้มีความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้นโดยนำ 1-D triangle area representation ของรูปร่างที่นำมาทดสอบมาเลื่อนตำแหน่งไป n ครั้งตาม

เอกสารนี้เป็นเอกสารจำนวน n จำนวน sampled point- n งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากหลักการในรูปที่ 5.17 นำไปสู่การพัฒนาวิธีการรู้จำให้คงทนต่อการเปลี่ยนแปลงจุดเริ่มต้นดังนี้

1. นำรูปร่างที่ต้องการทดสอบมาหา 1-D triangle area representation
2. นำ 1-D triangle area representation ของรูปร่างที่นำมาทดสอบมา normalized เลื่อนตำแหน่งไป n ครั้งตามจำนวน sampled point- n
3. นำ 1-D triangle area representation ที่ได้จากข้อ 2 มาทำการทดสอบประสิทธิภาพในการรู้จำกับวิธี Normalized cross-correlation และวิธี Neural networks ซึ่งแต่ละวิธีมีหลักการดังนี้

3.1 วิธี Normalized cross-correlation

3.1.1 จากรูปที่ 5.17 ในการเปรียบเทียบวัดความเหมือนระหว่างรูปร่างที่นำมาทดสอบกับรูปร่างในฐานข้อมูลรูปร่างหนึ่ง ค่าความเหมือนระหว่างสองรูปร่างนั้นพิจารณาจากค่า NCC ที่มากที่สุด ระหว่าง 1-D triangle area representation ที่ได้ทำการเลื่อนกับ 1-D triangle area representation ของรูปร่างในฐานข้อมูลดังแสดงในสมการที่ 5.4 ค่า NCC ที่มากที่สุดนี้จะแทนค่าความเหมือนระหว่าง 2 รูปร่างดังกล่าว

$$R(Q, D) = \max \frac{\sum_{i=1}^n q_{i+j} d_i}{\sqrt{\sum_{i=1}^n q_i^2 \sum_{i=1}^n d_i^2}} ; \text{ for } j = 1, 2, 3, \dots, n \quad (5.4)$$

3.1.2 เพื่อทำการระบุรูปร่างที่นำมาทดสอบนี้ตรงกับรูปร่างในฐานข้อมูล ก็ทำการคำนวณค่าความเหมือนของรูปร่างที่นำมาทดสอบกับรูปร่างอื่นๆในฐานข้อมูลตามหลักการในข้อที่ 3.1.1 รูปร่างที่นำมาทดสอบก็จะถูกระบุว่าอยู่ในกลุ่มเดียวกับรูปร่างในฐานข้อมูลที่ให้ค่าความเหมือนมากที่สุด

3.2 วิธี Artificial Neural networks

ในการที่จะระบุรูปร่างที่นำมาทดสอบด้วยวิธี Neural networks ให้นำ 1-D triangle area representation ที่ได้ทำการเลื่อนไป n ครั้งทั้งหมดไปเป็น input ให้กับ Neural networks แล้วพิจารณาค่า output ของ Neural networks ที่มากที่สุดว่าเป็นของ class ไหน รูปร่างที่นำมาทดสอบก็จะถูกระบุว่าอยู่ใน class นั้น

เอกสารนี้เป็นเอกสาร สำหรับรูปร่างที่นำมาทดสอบคุณสมบัติความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้นมีจำนวน 560 รูป โดยใช้ 1-D triangle area representation แบบ NATA ของแต่ละรูปร่างที่ทำมาจากจุดเริ่มต้นคนละจุดกัน (Different starting points) ซึ่งจะถูกรับนำมาทดสอบคุณสมบัติความคงทนต่อการ

เปลี่ยนแปลงจุดเริ่มต้นตามวิธีที่ได้อธิบายไว้ก่อนหน้านี้ โดยจะใช้ NATA จากจุดเริ่มต้นเดียวกันในแต่ละ class มาเป็นตัวแทนรูปร่างที่เก็บไว้ในฐานข้อมูลเพื่อใช้ในการเปรียบเทียบกับตัวทดสอบ (สำหรับวิธี Normalized cross-correlation) และใช้ในการฝึกสอน Neural networks ซึ่งจำนวนรูปร่างที่ใช้เก็บในฐานข้อมูลและใช้ฝึกสอน Neural networks มีจำนวน 560 รูป ผลการทดลองแสดงดังในตารางที่ 5.6

ตารางที่ 5.6 แสดงประสิทธิภาพของการรู้จำในด้านความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้นของวิธี NCC และวิธี NN ซึ่งทดสอบกับ NATA ที่หามาจากจุดเริ่มต้นที่แตกต่างกัน

NATA(<i>l</i>)	Accuracy of NCC%	Accuracy of NN%
1	92.14	15.0
3	94.64	30.89
5	97.5	35.89
10	98.21	54.29
20	98.93	62.86
30	98.04	56.79
40	96.79	48.93
49	95.36	18.21

จากผลการทดลองในตารางที่ 5.6 จะเห็นว่าประสิทธิภาพของความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้นของวิธี Normalized cross-correlation ให้ความแม่นยำสูงมากกว่า 90% ในขณะที่วิธี Neural networks ให้ประสิทธิภาพที่ต่ำ ด้วยเหตุนี้เราจึงพยายามพัฒนาวิธี Neural networks ให้มีประสิทธิภาพเพิ่มขึ้น โดยพิจารณาผลลัพธ์ของความยาวแขนของสามเหลี่ยมร่วมกัน โดยในการระบุรูปร่างแต่ละครั้ง เราจะพิจารณาผลลัพธ์จากความยาวแขนของสามเหลี่ยมที่ 1-49 ว่าตอบ class ไหนมากที่สุด รูปร่างที่นำมาทดสอบนั้นก็จะต้องระบุให้อยู่ใน class นั้น เมื่อทำการทดลองตามแนวคิดดังกล่าวพบว่าประสิทธิภาพด้านความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้นของวิธี Neural networks เพิ่มขึ้นเป็น 80.54%

จากผลการทดลองวัดประสิทธิภาพในการแยกแยะข้อมูลหรือค่า Fisher's ratio ของแต่ละความยาวแขนของสามเหลี่ยมที่แสดงในรูปที่ 5.16 แสดงให้เห็นว่าแต่ละความยาวแขนของสามเหลี่ยมมีความสามารถแยกแยะข้อมูลไม่เท่ากัน เราจึงตั้งสมมติฐานขึ้นมาว่าถ้าหากนำผลลัพธ์ของความยาวแขนของสามเหลี่ยมบางแขนที่มีค่า Fisher's ratio สูงมาพิจารณา จะสามารถเพิ่มความสามารถในด้านความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้นของวิธี Neural networks ให้ดียิ่งขึ้นได้ เราได้ทดลอง

พิสูจน์สมมติฐานดังกล่าวเพื่อพัฒนาวิธี Neural networks ให้มีประสิทธิภาพคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้นโดยพิจารณาผลลัพธ์ของความยาวแขนของสามเหลี่ยมบางแขน ผลการทดลองแสดงดังตารางที่ 5.7

ตารางที่ 5.7 ผลการทดลองของการเพิ่มประสิทธิภาพด้านความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้นของวิธี NN ด้วยการพิจารณาผลลัพธ์ของความยาวแขนของสามเหลี่ยมบางแขนว่าครอบคลุม class ไหนมากที่สุด

Number of TSL	Accuracy of NN%
49 [1-49]	80.54
43 [3-45]	80.71
41 [5-45]	80.54
38 [3-40]	80.54
36 [5-40]	79.46
30 [6-35]	77.14
20[11-30]	71.07
10[21-30]	63.04

จากผลการทดลองจะเห็นว่าการศึกษาผลลัพธ์ของความยาวแขนของสามเหลี่ยมบางแขนสามารถเพิ่มประสิทธิภาพด้านความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้นได้ ที่สำคัญจะสังเกตเห็นว่าเมื่อใช้ความยาวแขนของสามเหลี่ยมที่ 3-45 นั้นจะให้เปอร์เซ็นต์ความถูกต้องสูงกว่าการพิจารณาความยาวแขนของสามเหลี่ยมทั้งหมด ที่ 1-49 แสดงให้เห็นว่าการเลือกใช้ NATA เฉพาะความยาวแขนของสามเหลี่ยมที่มีค่า Fisher's ratio สูงและใช้จำนวนแขนที่เหมาะสม ไม่เพียงแต่จะสามารถลดพารามิเตอร์ในการพิจารณาแล้วยังสามารถเพิ่มประสิทธิภาพในด้านความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้นได้อีกด้วย

5.2.3 ประสิทธิภาพการรู้จำในด้านความคงทนต่อสัญญาณรบกวน

ในหัวข้อนี้เราต้องการวัดประสิทธิภาพในการทนต่อสัญญาณรบกวน โดยเราได้ทำการเพิ่มสัญญาณรบกวนไปยังเส้นรอบรูปด้วยค่า SNR=35db และ 45db ดังแสดงในรูปที่ 5.18 แล้วทำการวัดประสิทธิภาพในการรู้จำด้วยวิธี Normalized cross-correlation และ วิธี Neural networks ซึ่งผลการทดลองแสดงดังตารางที่ 5.8



ก) SNR=35db



ข) SNR=45db

รูปที่ 5.18 การเพิ่มสัญญาณรบกวนไปยังเส้นรอบรูป

ตารางที่ 5.8 แสดงการวัดประสิทธิภาพในการรู้จำในด้านความสามารถในการทนต่อสัญญาณรบกวน ของวิธี Normalized cross-correlation และ วิธี Neural networks ที่วัดกับ NATA ที่ความยาวแขนของสามเหลี่ยมค่าต่างๆ

NATA(<i>l</i>)	Accuracy of NCC%		Accuracy of NN%	
	SNR 35db	SNR 45db	SNR 35db	SNR 45db
1	19.29	76.25	21.79	81.43
3	69.29	97.14	71.61	98.93
5	91.43	98.75	86.79	99.11
10	96.96	99.29	91.61	99.82
20	99.64	99.82	99.11	99.82
30	99.11	99.82	96.61	99.82
40	97.5	99.64	95.36	99.82
49	90.89	99.64	93.75	99.64

จากตารางที่ 5.8 จะเห็นว่า ประสิทธิภาพของการรู้จำของทั้งวิธี Normalized cross-correlation และ วิธี Neural networks มีความสามารถในการทนต่อสัญญาณรบกวนได้ดี โดยเฉพาะที่ความยาวแขนของสามเหลี่ยมสูงๆ แต่ที่ความยาวแขนของสามเหลี่ยมต่ำๆ ประสิทธิภาพในการทนต่อสัญญาณรบกวนของทั้งสองวิธีต่างลดลง โดยเฉพาะเมื่อได้รับสัญญาณรบกวนมากขึ้น

(SNR=35db)
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.4 การเปรียบเทียบกับงานวิจัยอื่นๆ

สำหรับในหัวข้อนี้เราจะทำการเปรียบเทียบประสิทธิภาพของวิธีการในงานวิจัยนี้กับงานวิจัยในบทความอื่นที่ใช้ Shape representation และ วิธีการรู้จำที่แตกต่างกันออกไปในแต่ละบทความ โดยใช้ชุดข้อมูลในการทดลองชุดเดียวกัน คือ MPEG-7 ซึ่งประกอบไปด้วยภาพของรูปร่างทั้งหมด 1400 ภาพ แบ่งเป็น 70 class แต่ละ class มี 20 ภาพ สำหรับการเปรียบเทียบประสิทธิภาพจะใช้มาตรฐานการวัดแบบ “Bullseye test” ซึ่งถูกใช้เป็นเกณฑ์การเปรียบเทียบกันอย่างแพร่หลายในงานวิจัยแต่ละงาน

Bullseye test เป็นการวัดประสิทธิภาพในการกู้คืนรูปร่าง โดยจะนำรูปร่างแต่ละรูปที่อยู่ในฐานข้อมูล (MPEG-7) มาเป็นรูปร่างทดสอบ จากนั้นก็นำรูปร่างทดสอบดังกล่าวไปทำการ match กับรูปร่างทั้งหมดที่อยู่ในฐานข้อมูลเพื่อวัดความเหมือน (ซึ่งในแต่ละงานวิจัยก็จะใช้ Shape representation และวิธีวัดความเหมือนที่แตกต่างกันออกไป) จากนั้นก็จะพิจารณารูปร่างที่กู้คืนมาได้ 40 อันดับแรกที่มีความเหมือนมากที่สุด แล้วนับจำนวนรูปร่างที่ตรงกับ class ของรูปร่างที่นำมาทดสอบ ซึ่งในการทดสอบการกู้คืนรูปร่างของรูปร่างหนึ่งนั้นความเป็นไปได้มากที่สุดที่บรรดารูปร่างคำตอบจะตรงกับ class ของรูปร่างที่นำมาทดสอบคือ 20 โดยคะแนน Bullseye test คือ อัตราส่วนระหว่างรูปร่างที่กู้คืนมาได้แล้วถูกต้องของรูปร่างที่นำมาทดสอบทั้งหมด 1400 รูป กับความเป็นไปได้มากที่สุดที่รูปร่างซึ่งกู้คืนมาได้จะถูกต้อง (20×1400)

สำหรับการวัดประสิทธิภาพของวิธีในงานวิจัยนี้ด้วยมาตรฐานการวัดแบบ Bullseye test เราจะใช้ NATA เป็น Shape representation และใช้วิธี Normalized cross-correlation เป็นตัววัดความเหมือน ซึ่งผลคะแนน Bullseye test ของงานวิจัยนี้แสดงดังตารางที่ 5.9 โดยทำการเปรียบเทียบกับคะแนนของการใช้ Shape representation ต่างๆกันออกไปดังที่ได้รายงานไว้ในแต่ละบทความ

จากตารางที่ 5.9 จะเห็นว่า NATA ที่ความยาวแขนของสามเหลี่ยมเท่ากับ 1 และ 49 ให้คะแนนที่ค่อนข้างต่ำกว่างานวิจัยอื่น แม้กระทั่ง NATA ที่ความยาวแขนของสามเหลี่ยมเท่ากับ 20 (ซึ่งถือว่าเป็นมีประสิทธิภาพที่ดีเมื่อดูจากผลการทดลองในหัวข้อที่ผ่านมา) ก็ยังให้คะแนนที่ค่อนข้างต่ำ 66.76% เราจึงพยายามเพิ่มประสิทธิภาพ โดยอาศัยหลักการเหมือนกับการเพิ่มเปอร์เซ็นต์ความถูกต้องในด้านความสามารถทนต่อการเปลี่ยนแปลงจุดเริ่มต้นให้กับ NN ดังที่ได้อธิบายไว้ในหัวข้อที่ 5.2.2 คือพิจารณาผลลัพธ์จากความยาวแขนของสามเหลี่ยมที่ 1-49 ว่าตอบ class ไหนมากที่สุด โดยในที่นี้เราจะทำการพิจารณารูปร่างที่กู้คืนมาได้แต่ละอันดับ (1-40) ว่าในอันดับนั้นๆ บรรดาคำตอบของความยาวแขนของสามเหลี่ยม 1-49 ตอบ class ไหนมากที่สุดก็จะนำ class นั้นมาเป็นคำตอบ และเมื่อทำการพิจารณาตามหลักการดังกล่าวแล้ว ปรากฏว่าคะแนน Bullseye test เพิ่มขึ้นเป็น 77.98% ซึ่งถือว่ามีค่าค่อนข้างสูงเมื่อเทียบกับงานวิจัยอื่นๆ โดยมีเพียงงานวิจัยที่ใช้ Shape

representation แบบ Curve edit distance เท่านั้นที่มีคะแนนสูงกว่า แต่ก็ถือว่าสูงกว่างานวิจัยของเราไปไม่มากนัก

ตารางที่ 5.9 การเปรียบเทียบคะแนน Bullseye test ของ NATA ที่ใช้ในงานวิจัยนี้กับ Shape representation ที่ใช้ในงานวิจัยอื่น

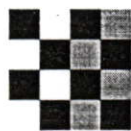
Method	The score of Bullseye test
Curve edit distance [44]	78.17%
NATA(TSL=1)	46.45%
NATA(TSL=20)	66.76%
NATA(TSL=49)	60.11%
NATA(all TSL [TSL=1-49])	77.98%
Shape context [45]	76.51%
Curvature Scale Space (CSS) [46]	75.44%
Zernike moment [47]	70.22%
Wavelet descriptor [48]	58.5%
Fourier descriptor [49]	55.2%

5.2.5 การทดสอบกับภาพถ่ายจริงที่ได้จากกล้องถ่ายรูปและตัวอย่างการประยุกต์ใช้งาน

โดยปกติแล้วเมื่อทำการถ่ายภาพวัตถุที่มีลักษณะแบนหรืออยู่ในรูป 2D จากมุมมองต่างๆ ภาพของวัตถุที่ได้นั้นจะเปรียบเสมือนถูกแปลงแบบ Perspective ซึ่งเป็นการแปลงแบบไม่เป็นเชิงเส้น ต่างจากการแปลงแบบ Affine ที่เป็นการแปลงแบบเชิงเส้น อย่างไรก็ตามลักษณะของรูปร่างที่ถูกแปลงแบบ Perspective กับรูปร่างที่ถูกแปลงแบบ Affine นั้นจะมีคุณสมบัติที่เหมือนกันคือเส้นตรงก่อนและหลังการแปลงจะยังคงเป็นเส้นตรงอยู่ ต่างกันตรงที่เส้นตรงที่มีลักษณะขนานกันอยู่เมื่อถูกแปลงแบบ Affine จะยังคงขนานกันอยู่ แต่สำหรับการแปลงแบบ Perspective เส้นตรงนั้นจะไม่ขนานกัน ดังแสดงรูปที่ 5.19

D. W. Thompson [50] และ R. Hartley [51] ได้กล่าวว่าภาพของวัตถุแบน (Planar object) ที่ถูกแปลงแบบ Perspective อันเนื่องมาจากการถ่ายรูปจากมุมมองต่างๆ นั้น สามารถแทนหรือเปรียบได้กับการแปลงแบบ Affine ได้ ถ้าหากจุดศูนย์กลางกล้องอยู่ไกลพอกับระนาบของวัตถุแบน หรือ ถ้าหากภาพของวัตถุนั้นถูกแปลงแบบ Perspective ไปไม่มาก (Weak perspective) ด้วยเหตุนี้เราจึงทำการทดลองวิธีการของเราซึ่งมีคุณสมบัติไม่ผันแปรต่อการแปลงแบบ Affine กับภาพถ่าย

ของวัตถุแบนที่ได้จากมุมมองต่างๆ 14 มุม โดยใช้วัตถุแบน 40 แบบที่แสดงในรูปที่ 5.1 ตัวอย่างภาพถ่ายจากมุมมองต่างๆแสดงดังรูปที่ 5.20



ก) รูปร่างเดิม



ข) รูปร่างที่ถูกแปลงแบบ Affine

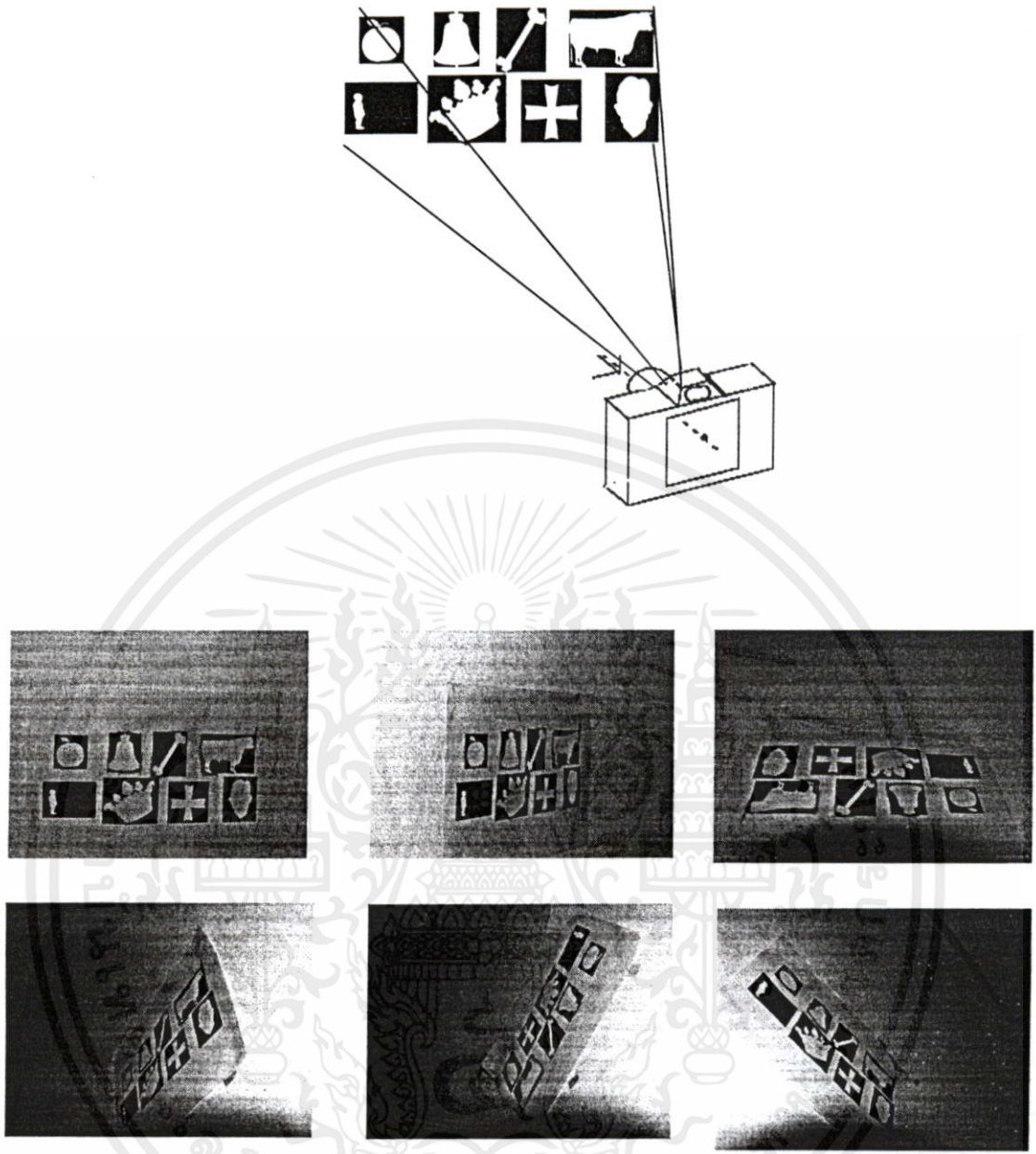


ค) รูปร่างที่ถูกแปลงแบบ Perspective

รูปที่ 5.19 รูปร่างที่ถูกแปลงแบบ Affine และ Perspective

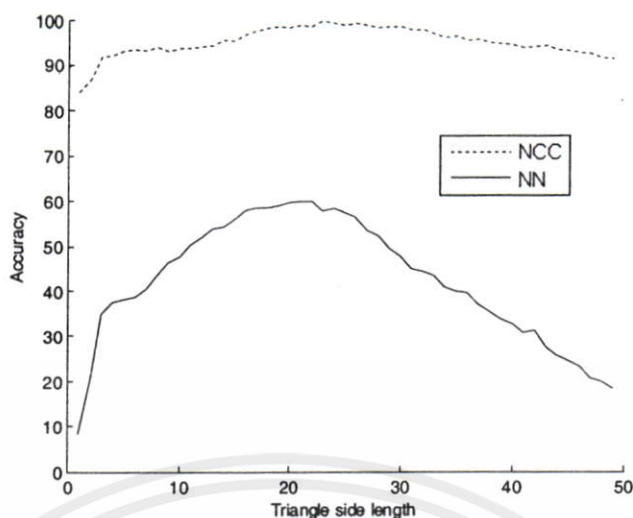
ในการทดลองเราได้วัดประสิทธิภาพในการรู้จำด้วยวิธี Normalized cross-correlation และวิธี Neural networks โดยใช้ NATA ของรูปร่างที่ถูกแปลงแบบ Affine จากหัวข้อที่ 5.1 ซึ่งจะนำรูปร่างทั้ง 14 แบบในแต่ละclass มาเป็นรูปร่างตัวอย่างที่ใช้เก็บไว้ในฐานข้อมูล (สำหรับวิธี Normalized cross-correlation) และใช้ฝึกสอน Neural networks สำหรับรูปร่างที่ใช้ทดสอบนั้นคือรูปร่างที่ถ่ายมาได้ทั้ง 14 แบบ เพราะฉะนั้นจะมีรูปร่างที่ใช้ทดสอบทั้งหมด 560 รูป ซึ่งผลการทดลองแสดงดังรูปที่ 5.21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.20 ภาพของวัตถุแบนที่ถูกถ่ายจากมุมกล้องต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

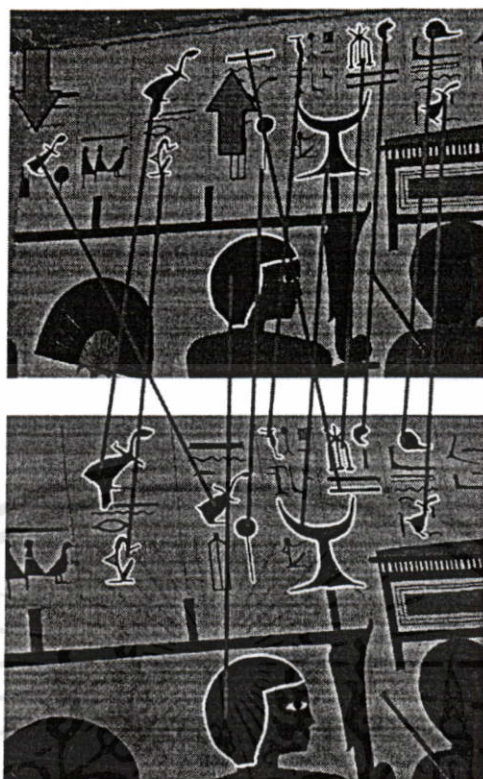


รูปที่ 5.21 ประสิทธิภาพในการรู้จำของวิธี Normalized cross-correlation และ วิธี Neural networks ที่ทดสอบกับรูปร่างที่ถ่ายได้จากกล้องถ่ายภาพ ด้วยการใช้ NATA ที่ค่าความยาวแกนของสามเหลี่ยม 1-49

จากกราฟในรูปที่ 5.2.1 ประสิทธิภาพในการรู้จำของวิธี Normalized cross-correlation ให้ความถูกต้องค่อนข้างสูง กล่าวได้ว่าสามารถนำไปประยุกต์ใช้งานกับภาพถ่ายจริงได้ สำหรับวิธี Neural networks นั้นให้ผลที่ไม่ค่อยสูงนัก โดยที่ค่าความยาวแกนที่ดีที่สุดให้ความถูกต้องอยู่ที่ประมาณ 60% ด้วยเหตุนี้เราจึงทำการเพิ่มประสิทธิภาพของวิธี Neural networks ด้วยการพิจารณาคำตอบร่วมทั้ง 49 ความยาวแกนของสามเหลี่ยมตามหลักการที่ได้กล่าวไว้ก่อนหน้านี้ ผลปรากฏว่าประสิทธิภาพของ Neural networks เพิ่มขึ้นเป็น 76.25% ซึ่งถือว่ายอมรับได้

จากการทดลองได้แสดงให้เห็นแล้วว่า วิธีการของเราสามารถใช้กับภาพถ่ายจริง โดยเราได้นำไปประยุกต์หารูปร่างที่มีความเหมือนกันระหว่างภาพ 2 ภาพที่ถ่ายจากคนละมุมมองกล้องกัน ดังแสดงในรูปที่ 5.22 โดยรูปร่างที่เราสนใจจะแสดงเส้นขอบไว้ เส้นตรงที่เชื่อมระหว่าง 2 รูปร่างในแต่ละภาพแสดงถึงรูปร่างที่มีความเหมือน ส่วนรูปร่างที่มีลูกศรปรากฏแสดงถึงการรู้จำที่ผิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.22 การหารูปร่างที่มีความเหมือนกันระหว่างภาพ 2 ภาพที่ถูกถ่ายคนละมุมมองด้วยกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

ในงานวิจัยนี้ได้นำเสนอวิธีการรู้จำรูปร่างด้วยการนำภาพดิจิทัลของรูปร่างมาทำการพิจารณา โดยเราได้ทำการวิจัยและพัฒนาวิธีการให้มีความสามารถในการระบุรูปร่างแม้ว่ารูปร่างในภาพดังกล่าวจะมีการเปลี่ยนแปลงของขนาด ตำแหน่ง มุมและการเอียง ซึ่งการเปลี่ยนแปลงดังกล่าวเรียกว่าการแปลงแบบแอฟไฟน์ (Affine transform) หลักการที่ทำให้วิธีการรู้จำมีความคงทนต่อการเปลี่ยนแปลงดังกล่าวคือ การดึงเอาคุณลักษณะเฉพาะที่มีคุณสมบัติไม่ผันแปรต่อการแปลงแบบแอฟไฟน์ออกมาจากรูปร่างแต่ละชนิด ในที่นี้เราได้นำทฤษฎีคิเทอร์มินแนมาหาคุณลักษณะเฉพาะดังกล่าวซึ่งเรียกว่า 1-D triangle area representation ซึ่งเป็นคุณลักษณะที่หาได้ง่ายไม่ซับซ้อนที่เป็นตัวแทนของแต่ละรูปร่างในโดเมนสเปซ เพื่อจะนำไปใช้ในกระบวนการรู้จำต่อไป

สำหรับวิธีการรู้จำที่ได้ทำการศึกษาในงานวิจัยนี้มี 2 วิธีคือ Normalized cross-correlation (NCC) และ Neural networks (NN) ซึ่งแต่ละวิธีก็มีหลักการในการรู้จำที่แตกต่างกันไปโดยวิธี Normalized cross-correlation นั้นเป็นการนำเอารูปร่างที่ต้องการทดสอบมาทำการเปรียบเทียบกับรูปร่างที่เก็บไว้ในฐานข้อมูล ส่วนวิธี Neural networks นั้นเปรียบเสมือนการนำเอารูปร่างที่ต้องการทดสอบมาคำนวณด้วยฟังก์ชันการรู้จำที่ถูกสร้างมาจากการฝึกสอนด้วยชุดข้อมูลจากรูปร่างต้นแบบจำนวนหนึ่ง

ในการทดลองเราได้ทำการวัดประสิทธิภาพในการรู้จำของทั้งสองวิธีด้วยการใช้ 1-D triangle area representation แบบ ATA และ NATA ที่ความยาวแขนของสามเหลี่ยมต่างๆ จากผลการทดลองพบว่าวิธีการรู้จำด้วยวิธี Normalized cross-correlation (NCC) ให้ประสิทธิภาพที่ดีกว่า Neural networks (NN) อย่างไรก็ตามถ้าหากกล่าวถึงในด้านความเร็วในการระบุรูปร่างนั้นวิธี Neural networks มีความเร็วกว่าวิธี Normalized cross-correlation ส่วนการพิจารณาว่า 1-D triangle area representation แบบ ไหนและที่ความยาวแขนของสามเหลี่ยมใดมีประสิทธิภาพในการแยกแยะข้อมูลได้สูง จากผลการทดลองของการรู้จำและการทำนายความสามารถในการแยกแยะข้อมูลด้วย Fisher's ratio ของ 1-D triangle area representation แต่ละชนิด พบว่าวิธีการรู้จำที่ใช้ 1-D triangle area representation แบบ NATA ที่ความยาวแขนของสามเหลี่ยมประมาณ 20 ให้ประสิทธิภาพในการรู้จำและ Fisher's ratio ที่สูง ซึ่งแสดงว่ามีความสามารถในการแยกแยะข้อมูลสูง

นอกจากนี้เรายังได้พัฒนาวิธีการรู้จำให้มีความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้น โดยนำ 1-D triangle area representation มาทำการเลื่อนตำแหน่งไปตาม sampled point แล้วนำเข้าสู่กระบวนการรู้จำ จากผลการทดลองพบว่าวิธีการรู้จำในด้านความคงทนต่อการเปลี่ยนแปลงจุดเริ่มต้นมีประสิทธิภาพที่ดี

สำหรับการวัดประสิทธิภาพในการทนต่อสัญญาณรบกวนนั้น เราได้ทำการเพิ่มสัญญาณรบกวนให้กับเส้นรอบรูปในระดับหนึ่ง แล้วทำการวัดประสิทธิภาพในการรู้จำ ผลการทดลองแสดงให้เห็นว่าวิธีการของเราสามารถทนต่อสัญญาณรบกวนได้ในระดับหนึ่ง

เพื่อพิจารณาว่าวิธีการของเรามีประสิทธิภาพมากน้อยเพียงใดเมื่อเทียบกับงานวิจัยอื่นๆ เราได้ทำการทดลองเปรียบเทียบประสิทธิภาพของวิธีเรากับงานวิจัยอื่นที่ใช้ชุดข้อมูลเดียวกันคือ MPEG-7 และใช้มาตรฐานการวัดประสิทธิภาพในการกู้คืนรูปร่างที่เรียกว่า Bullseye test ผลการทดลองแสดงให้เห็นว่า NATA ที่ได้นำเสนอในงานวิจัยของเรานั้น มีประสิทธิภาพที่ดีกว่า Shape context [45], Curvature Scale Space (CSS) [46], Zernike moment [47], Wavelet descriptor [48] และ Fourier descriptor [49] อย่างไรก็ตาม NATA มีประสิทธิภาพน้อยกว่า Curve edit distance [44] แต่ก็ถือว่าต่ำกว่าเล็กน้อย

ในส่วนของการทดลองกับภาพถ่ายจริง เราได้ทำการทดลองวิธีการที่นำเสนอเกี่ยวกับภาพถ่ายของวัตถุแบน 2 มิติ ซึ่งโดยปกติแล้วภาพถ่ายของวัตถุแบนจากมุมมองกล้องต่างๆ นั้นจะเปรียบเทียบเหมือนการแปลงแบบ Perspective แต่เราสามารถประมาณการแปลงแบบ Perspective นั้นได้ด้วยการแปลงแบบ Affine ในกรณีที่จุดศูนย์กลางกล้องอยู่ไกลพอจากระนาบของวัตถุแบน หรือภาพที่ได้ถูกแปลงแบบ Perspective ไปไม่มาก (Weak perspective) ผลการทดลองแสดงให้เห็นว่าวิธีการที่นำเสนอสามารถรู้จำรูปร่างที่มาจากภาพถ่ายจริงได้ นอกจากนั้นเรายังได้แสดงตัวอย่างการประยุกต์ใช้งานกับการหารูปร่างที่มีความเหมือนกับ 2 รูปร่างที่ถ่ายมาจากคนละมุมกล้องกัน อย่างไรก็ตามวิธีการของเราจะสามารถใช้งานได้ก็ต่อเมื่อเราสามารถแยกรูปร่างออกจากฉากหลังได้ดีโดยมีเส้นรอบรูปที่สมบูรณ์ไม่ขาดหายหรือถูกบดบังไป

ข้อจำกัดของงานวิจัยนี้คือไม่สามารถใช้กับรูปร่างที่ถูกบดบังหรือเส้นรอบรูปเปิดได้ นอกจากนี้ยังไม่สามารถแยกแยะระหว่างรูปร่างกลมกับรูปร่างรีได้เนื่องจากรูปร่างเป็นเหมือนรูปร่างกลมที่ถูกแปลงแบบแอฟไฟน์ รวมทั้งรูปร่างที่ถูกแปลงแบบ Reflection ถือเป็นข้อจำกัดของวิธีการของเรา เนื่องจากจะทำให้ลักษณะของ NATA ผิดจากลักษณะเดิมไปคือกลับค่าจากบวกเป็นลบ ซึ่งจะส่งผลกระทบต่อความถูกต้องในการรู้จำได้ ส่วนรูปร่างที่ถูกแปลงแบบเฉือนไปมากจนเกินไปจนเสียเค้าโครงของรูปร่างเดิมอาจจะทำให้ลักษณะของ Shape representation ของเราไม่สามารถเก็บลักษณะเฉพาะไว้ได้ ด้วยเหตุนี้อาจจะทำให้เกิดข้อผิดพลาดในการรู้จำได้ เนื่องจากวิธีการที่นำเสนอสามารถเปรียบเทียบได้เฉพาะขอบภาพ จึงไม่สามารถเก็บรายละเอียดภายในรูปร่างได้ สำหรับกรณีรูปร่างที่มีลักษณะโดยรวมคล้ายกันและถูกจัดให้อยู่ในประเภทเดียวกัน แต่เมื่อคุณลักษณะโดยละเอียดในแต่ละส่วนแล้วกลับต่างกัน กรณีนี้จะทำให้เส้นรอบรูปของรูปร่างดังกล่าวแตกต่างกันซึ่งจะส่งผลให้ Shape representation แตกต่างกันไปด้วย อาจจะทำให้การระบุรูปร่างผิดพลาดได้ เช่นเดียวกันกับกรณีของรูปร่างที่ถูกจัดให้อยู่คนละประเภทแต่มีลักษณะที่คล้ายกัน ในส่วนนี้อาจจะมีปัญหาในการรู้จำขึ้นได้

จากการศึกษาและทำการทดลองพบว่าในบางขั้นตอนของเราก็ยังมีข้อบกพร่องอยู่เช่น ในขั้นตอนการพัฒนาวิธีการในการรู้จำให้สามารถทนต่อการเปลี่ยนแปลงจุดเริ่มต้น โดยทำการเลื่อนตำแหน่งของ Shape representation ของรูปร่างหนึ่งไปจำนวน 100 ครั้งตามจำนวนจุด Sampling ซึ่งถือว่าใช้เวลาในขั้นตอนการรู้จำมาก เช่นในกรณีของวิธี Normalized cross-correlation ในการระบุรูปร่างที่ต้องการทดสอบรูปร่างหนึ่งนั้นเราต้องทำการเปรียบเทียบกับทุกๆรูปร่างที่อยู่ในฐานข้อมูล ที่มี N รูปร่าง และในการเปรียบเทียบรูปร่างแต่ละคู่กันนั้นต้องนำ Shape representation ของรูปร่างที่นำมาทดสอบ ทำการเลื่อนตำแหน่งและวัดความเหมือนกับรูปร่างที่นำมาเปรียบเทียบไป 100 ครั้ง เพราะฉะนั้นเมื่อนับจำนวนครั้งของการวัดความเหมือนของรูปร่างหนึ่งกับรูปร่างทั้งหมดที่อยู่ในฐานข้อมูลแล้วจะมีจำนวน $N \times 100$ ครั้ง ซึ่งถือว่ามาก เราจึงมีข้อเสนอแนะคือ ทำการเลื่อนตำแหน่ง Shape representation ให้น้อยลงกว่า 100 จุด ซึ่งอาจจะเลื่อนตำแหน่งไปจำนวน c ครั้งตามจำนวนจุดวิกฤต (Critical) ซึ่งจุด Critical นี้อาจจะเป็นส่วนเว้าส่วนนูนบนเส้นรอบรูป หรืออาจจะเป็นจุดที่มีลักษณะเด่นบนเส้นรอบรูปซึ่งก็แล้วแต่ว่าจะหาโดยวิธีใด นอกจากนั้นเราอาจจะสามารถพัฒนา Shape representation ของเราที่จากเดิมที่มีลักษณะเป็น vector เดียว อาจจะนำมารวมกันให้อยู่ในรูปของเมทริกซ์แล้วเลือกใช้เฉพาะองค์ประกอบภายในเมทริกซ์นั้นที่มีค่า Fisher's ratio สูงๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] D. Zhang and G. Lu, "Review of shape representation and description techniques," *Patt. Recog.*, vol. 37, no. 1, pp. 1-19, 2004.
- [2] I. Yong, J. Walker, J. Bowie, "An analysis technique for biological shape", *Comput. Graphic Image Process.* 25(1974) 357-370.
- [3] M. Peura, J. Iivarinen, "Efficiency of simple shape descriptor", *Proceedings of the Third International Workshop on Visual Form, Capri, Italy, May, 1997*, pp. 443-451.
- [4] E.R. Davies, *Machine Vision: "Theory, Algorithms, Practicalities"*, Academic Press, New York, 1997, pp. 171-191.
- [5] P.J. van Otterloo, "A Contour-Oriented Approach to Shape Analysis", Prentice-Hall International (UK) Ltd, Englewood Cliffs, NJ, 1991, pp. 90-108.
- [6] D.S. Zhang, G. Lu, "A comparative study of Fourier descriptors for shape representation and retrieval", in: *Proceedings of the Fifth Asian Conference on Computer Vision (ACCV02)*, Melbourne, Australia, January 22-25, 2002, pp. 646-651.
- [7] M. Sonka, V. Hlavac, R. Boyle, "Image Processing, Analysis and Machine Vision", Chapman & Hall, London, UK, NJ, 1993, pp. 193-242.
- [8] Farzin Mokhtarian and Alan K. Mackworth, "A Theory of Multiscale, Curvature-Based Shape Representation for Planar Curves," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 8, pp. 789-805, Aug. 1992.
- [9] G. Granlund, "Fourier preprocessing for hand print character recognition", *IEEE Trans. Comput.* 21 (1972) 195-201.
- [10] C.C. Lin, R. Chellappa, "Classification of partial 2D shapes using Fourier descriptors", *IEEE Trans. Pattern Anal. Mach. Intell.* 9 (5) (1987) 686-690.
- [11] O.R. Mitchell, T.A. Grogan, "Global and partial shape discrimination for computer vision", *Opt. Eng.* 23 (5) (1984), pp. 484-491.
- [12] K. Arbter, "Affine-invariant Fourier descriptors", in: J.C. Simon (Ed.), *From Pixels to Features*, Elsevier Science Publishers B.V. (North-Holland), Amsterdam, 1989, pp. 153-164.
- [13] K. Arbter, W.E. Snyder, H. Burkhardt, G. Hirzinger, "Application of affine-invariant Fourier descriptors to recognition of 3-D objects", *IEEE Trans. Pattern Anal.*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mach. Intell. 12 (7) (1990) 640–647.

- [14] J.R. Ohm, F.B. Bunjamin, W. Liebsch, B. Makai, K. Muller, A. Somlic, D. Zier, “A set of visual feature descriptors and their combination in a low-level description scheme”, *Signal Process. Image Commun.* 16 (2000) 157–179.
- [15] Q.M. Tieng, W.W. Boles, “Recognition of 2D object contours using the wavelet transform zero-crossing representation”, *IEEE Trans. Pattern Anal. Mach. Intell.* 19 (8) (1997) 910–916.
- [16] H.S. Yang, S.U. Lee, K.M. Lee, “Recognition of 2D object contours using starting-point-independent wavelet coefficient matching”, *J. Visual Commun. Image Represent.* 9 (2) (1998) 171–181.
- [17] Ibrahim. El Rube', M. Ahmed, and M. Kamel. “Affine Invariant Multiscale Wavelet-based Shape Matching Algorithm”. *Proceeding of the First Canadian Conference on Computer and Robot Vision*, pp. 217-224, May 2004.
- [18] T. Pavlidis, “Algorithms for Graphics and Image Processing”, Computer Science Press, Rockville, MD, 1982, p. 143.
- [19] H. Freeman, “On the encoding of arbitrary geometric configurations”, *IRE Trans. Electron. Comput.* EC-10 (1961) 260–268.
- [20] W.I. Groskey, R. Mehrotra, “Index-based object recognition in pictorial data management”, *Comput. Vision Graphics Image Process.* 52 (1990) 416–436.
- [21] W.I. Groskey, P. Neo, R. Mehrotra, “A pictorial index mechanism for model-based matching”, *Data Knowledge Eng.* 8 (1992) 309–327.
- [22] K.S. Fu, “Syntactic Methods in Pattern Recognition”, Academic Press, New York, 1974.
- [23] M.K. Hu, “Visual pattern recognition by moment invariants”, *IRE Trans. Inf. Theory* IT-8 (1962) 179–187.
- [24] G. Taubin, D.B. Cooper, “Recognition and positioning of rigid objects using algebraic moment invariants”, *SPIE Conference on Geometric Methods in Computer Vision*, Vol. 1570, University of Florida, Florida, USA 1991, pp. 175–186.
- [25] G. Taubin, D.B. Cooper, “Object recognition based on moment” (or Algebraic, in: J. Mundy, A. Zisserman (Eds.), *Geometric Invariance in Computer Vision*, MIT Press, Cambridge, MA, 1992, pp. 375–397.
- [26] M.R. Teague, “Image analysis via the general theory of moments”, *J. Opt. Soc. Am.* 70 (8) (1980) 920–930.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [27] D.S. Zhang, G. Lu, "Generic Fourier descriptor for shape-based image retrieval", in: Proceedings of IEEE International Conference on Multimedia and Expo (ICME2002), Vol. 1, Lausanne, Switzerland, August 26–29, 2002, pp. 425–428.
- [28] G.J. Lu, A. Sajjanhar, "Region-based shape representation and similarity measure suitable for content-based image retrieval", *Multimedia Syst.* 7 (2) (1999) 165–174.
- [29] K. Chakrabarti, M.O. Binderberger, K. Porkaew, S. Mehrotra, "Similar shape retrieval in MARS", in: Proceedings of IEEE International Conference on Multimedia and Expo, Vol. 2, New York, USA, 2000, pp. 709–712.
- [30] M. Safar, C. Shahabi, X. Sun, "Image retrieval by shape: a comparative study", in: Proceedings of IEEE International Conference on Multimedia and Expo, Vol. 1, New York, USA, 2000, pp. 141–144.
- [31] T.S. Huang, S. Mehrotra, K. Ramachandran, "Multimedia analysis and retrieval system (MARS) project, in: Proceedings of 33rd Annual Clinic on Library Application of Data Processing—Digital Image Access and Retrieval, University of Illinois at Urbana-Champaign, Illinois, USA 1996, pp. 101–117.
- [32] A. Goshtasby, "Description and discrimination of planar shapes using shape matrices", *IEEE Trans. Pattern Anal. Mach. Intell.* 7 (1985) 738–743.
- [33] H. Blum, "A transformation for extracting new descriptors of shape", in: W. Whaten-Dunn (Ed.), *Models for the Perception of Speech and Visual Forms*, MIT Press, Cambridge, MA, 1967, pp. 362–380.
- [34] H. Anton and C. Rorres, "Elementary Linear Algebra," Wiley and Sons, Inc, ninth edition, 2005, pp. 458-459.
- [35] David C. Lay, "Linear Algebra and Its Applications". Pearson Education, Inc, third edition, 2006
- [36] L. Fausett, "Fundamentals of Neural Networks", Prentice-Hall International, Inc, 1994.
- [37] นิสิตท์ สมิตท์สมบุรณ์, ชาตรี พันธุ์อัจฉริยกุล และ ชวาล วรณเทพรักษา, "ระบบประมวลผลภาพสำหรับหุ่นยนต์สำรวจ", *ปริชญานิพนธ์ วิศวกรรมศาสตรบัณฑิต กรุงเทพฯ: คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง* 2536, หน้า (49-54).
- [38] Martin T. Hagan, Howard B. Demuth, M. Beale, "Neural Network Design", An International Thomson Publishing Company, 1996.
- [39] Simon Haykin, "Neural Networks, A Comprehensive Foundation", Macmillan, 1994.

- [40] Dariusz Z. Lejtman, Susan E. George, "On-line Handwritten Signature Verification using Wavelets and Back-Propagation Neural Networks", *IEEE Document Analysis and Recognition*, 10-13 Sept. 2001, pp.992-996.
- [41] MPEG-7 dataset. Citing Internet sources URL <http://www.cis.temple.edu/~latecki/>.
- [42] W.K.Pratt. 1991. "Digital Image Processing": John Wiley and sons, Inc, second edition.
- [43] Peter N. Belhumeur, Joao P. Hespanha, and David J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection", *IEEE Transactions on pattern analysis and machine intelligenc*, Vol. 19, No. 7, July 1997
- [44] T. B. Sebastian, P. N. Klein and B. B. Kimia. "On Aligning Curve", *IEEE Trans. Pattern Anal. Mach. Itell.*, 25(1):116-125, 2003.
- [45] S. Belongie, J. Malik and J. Puzicha. "Shape Matching and Object Recognition Using Shape Context," *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(24):509-522, 2002
- [46] F. Mokhtarian, S. Abbasi and J. Kittler. "Efficient and Robust Retrieval by Shape Content through Curvature Scale Space," in A. W. M. Smeulders and R. Jain, editors, *Image Databases and Multi-Media Search*, 51-58, World Scientific, 1997.
- [47] A. Khotanzan and Y. H. Hong. "Invariant image recognition by zernike moments," *IEEE Trans. PAMI*, 12:489-497, 1990
- [48] G. C.-H. Chuang and C.-C.J. Kuo, "Wavelet descriptor for planar curves:Theory and applications," *IEEE Trans. Image Process.*, vol. 5, no. 1, pp. 56-70, Jan. 1996.
- [49] C. T. Zahn and R. Z. Roskies, "Fourier descriptors for plane closed curves," *IEEE Trans. Comput.*, vol.C-21,no. 3, pp. 269-281, Mar.1972
- [50] D. W. Thompson and J. L. Mundy, "Three dimensional model matching from an unconstrained viewpoint," in Proc. Conf. Robotics and Automation, Raleigh, NC, pp. 208-220, 1987.
- [51] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000.

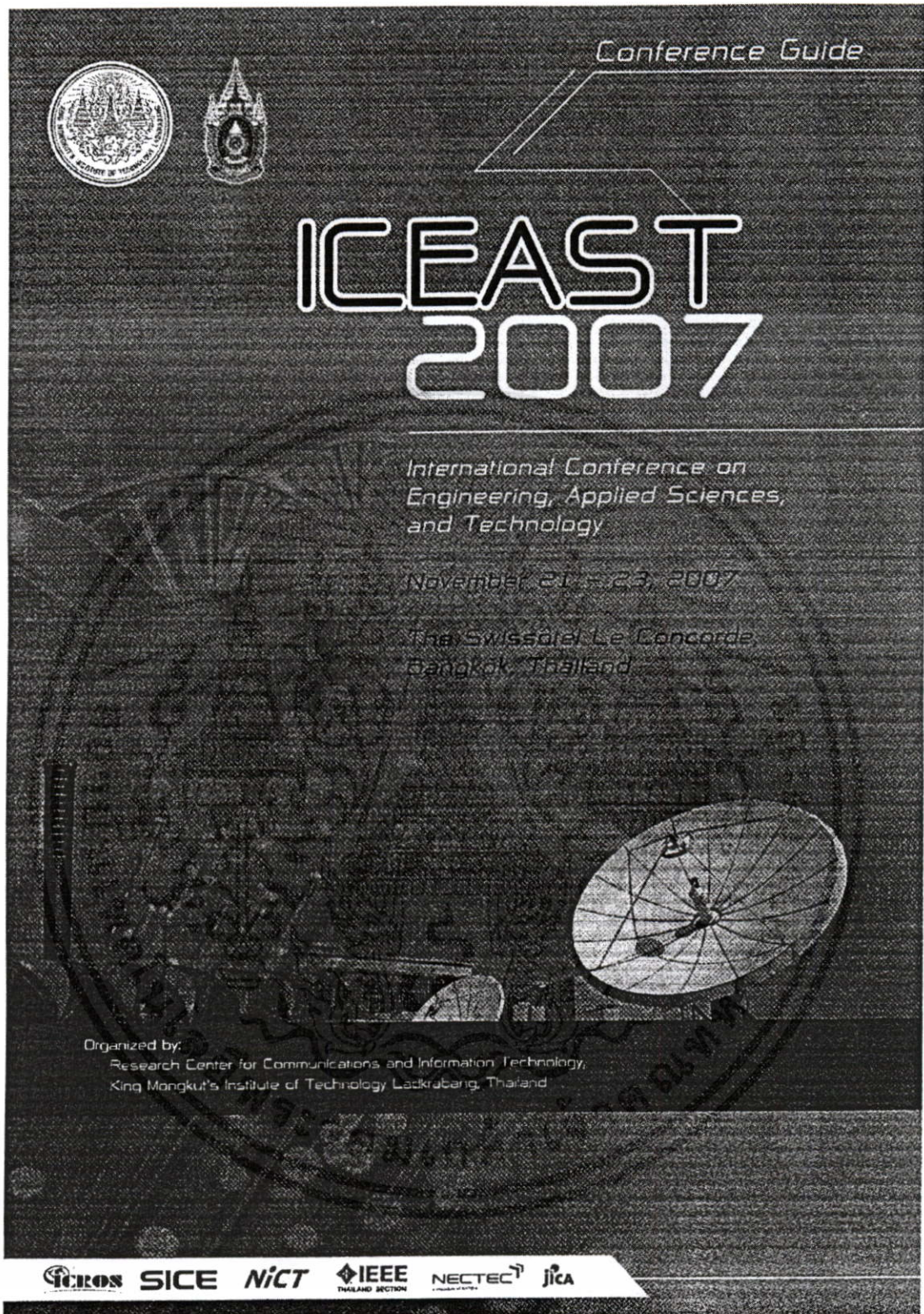
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

ผลงานวิจัยที่ได้รับการตีพิมพ์

1. Kritsana Yawichai, Yuttana Kitjaidure, "A Simple and Efficient Algorithm for Affine Invariant Shape Matching," International Conference on Engineering, Applied Sciences, and Technology (ICEAST), Bangkok, Thailand, pp.246-249, 2007.
2. Kritsana Yawichai, Yuttana Kitjaidure, "Multi-view Shape Matching and Retrieval," International Conference on Robotics, Vision, Information, and Signal Processing (ROVISP), Penang, Malaysia, pp.454-458, 2007.
3. Kritsana Yawichai, Yuttana Kitjaidure, "Fast Classification of Multi-view Shape using Neural network," 1st Joint International Conference on Information and Communication Technology (JICT), Vientiane, Lao PDR, pp.88-92, 2007.
4. Thourn Kosorl, Kritsana Yawichai, Yuttana Kitjaidure, "Affine Invariant Shape Recognition Based on Neural Networks", International Conference on Embedded Systems and Intelligent Technology (ICESIT), Bangkok, Thailand, pp.130-134, 2008.
5. Thourn Kosorl, Kritsana Yawichai, Yuttana Kitjaidure, "Optimum Shape Representation Based on Fisher's Discriminant Analysis", The Fifth International Conference in Electrical Engineering/Electronics, Computer, Telecommunications, and Information Technology (ECTI-CON), Krabi, Thailand, pp. 509-512, 2008.
6. Kritsana Yawichai, Yuttana Kitjaidure, "Multi-view Invariant Shape Recognition Based on Neural Networks", The 3rd IEEE Conference on Industrial Electronics and Applications (ICIEA), Singapore, 2008, in press.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Conference Guide

**ICEAST
2007**

International Conference on
Engineering, Applied Sciences,
and Technology

November 21-23, 2007

King Swissôtel Le Concorde
Bangkok, Thailand

Organized by:
Research Center for Communications and Information Technology,
King Mongkut's Institute of Technology Ladkrabang, Thailand

ICROS SICE NiCT IEEE THAILAND SECTION NECTEC JICA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A Simple and Efficient Algorithm for Affine Invariant Shape Matching

Kritsana Yawichai and Yuttana Kitjaidure

Department of Electronics

Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Thailand

Abstract— Shape matching has been a subject of extensive study for a long time. It appears in many applications including pattern recognition, shape discrimination and analysis. In real world situation when we capture the planar object in multiple viewpoints, we normally obtain each shape distorted in different kinds of affine transform. Developing suitable affine invariant shape feature is still a challenging problem in shape matching and retrieval. This work presents an approach to solve the problem. The shape boundary is considered and the determinant theory is used to compute the 1-D triangle area representation that is invariant to the affine transformation. On matching stage, similarity is measured by calculating the normalized cross correlation of the 1-D triangle area representation of the shape contour. The proposed algorithm can be used to match and retrieve two-dimensional shapes without regard to their orientation, location, size, and shear.

Keywords— Affine transformation, shape matching and retrieval

I. INTRODUCTION

In human vision, shape is a significant visual feature and is used in recognition system. In robot and computer vision, shape is also used in pattern recognition but one more problem occurs, how we can create an algorithm to increase recognition performance of robot and computer visions as well as humans do. Our research presents an approach to solve this problem. As mentioned before, shape is an important visual feature so we use shape of the object in matching and retrieval process. Shape description techniques can be divided into contour-based and region-based. In [1], contour-based representation uses less data points and is usually less sensitive to noise than the region-based techniques that why we decide to choose the contour-based description technique in our work. Several contour features have been introduced in the literature [2], [3].

The appropriate descriptor of shape feature should not change under perspective distortion which typically arises if a planar object is observed under arbitrary viewpoints with respect to the plane. However, the affine transformation may be used as approximation to the perspective transformation rather than the use of similarity transformation [4]. Miyatake, Matsuyama, and Nagao [5] proposed an optimizing procedure to produce invariance under affine transformations. Ibrahim El Rube', Maher Ahmed and Mohamed Kamel [6] had investigated affine invariant shape matching based on the dyadic wavelet transform (DWT) which was used to transform contour of shape from spatial domain into

frequency domain. The frequency components are composed of approximation coefficients and detail coefficients. Then, the curve moment invariant and the 1-D triangle area representation are used with those two types of coefficients to measure dissimilarity and similarity of each shape's contour respectively. It can be seen that most of the previous works with the shape classification are very computationally expensive. The simple algorithm developed in this paper has adopted the 1-D triangle area representation [6] in spatial domain instead of frequency domain to be an affine invariant shape descriptor. Our algorithm is not only uncomplicated but also works reasonably well in the presence of moderate amount of noise.

The rest of this paper is organized as follow: Section II introduces the proposed algorithm with detailed description to all stages. In section III the experiment results are obtained. Finally, Section IV gives conclusion and future work.

II. PROPOSED METHODOLOGY

In this section, we present our methodology to achieve shape matching and retrieval. The overall process is shown in Fig. 1. First, the shape contours of both query and database shapes are extracted. Then, they are smoothed by a Gaussian filter and sampled with the same number of positions, n . Next, the sampled points of each contour are calculated to obtain the 1-D triangle area representation, each of which is an affine invariant descriptor. Finally, shape matching and retrieval are achieved by computing the normalized cross-correlation function between the 1-D triangle area representations of the query shape and database shapes. The detailed descriptions of all stages are discussed in the next section.

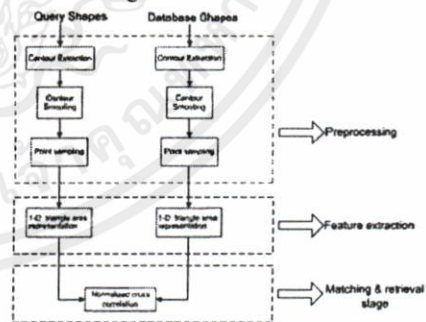


Fig. 1. The proposed process

A. Preprocessing

1. Contour extraction

The shape is segmented from the image in this section. Then, the contour following technique [7] is applied to the shape to obtain the contour as a sequence of points $(x(u), y(u))$ which is the coordinates of each point on the contour. Fig. 2 shows the result of the extracted contour.

2. Contour smoothing

To avoid effect of noises caused by digitization noise and segmentation errors, the contour sequences $(x(u), y(u))$ are smoothed by a Gaussian filter with σ set to 1.5% of N , where N is the total number of coordinate points on the considered contour. Fig. 3 illustrates the result of smoothing the contour sequences $(x(u), y(u))$.

3. Point sampling

Every contour (query and database contours) is sampled at the equal number of points, n . The sampled points will be denoted by k_1, k_2, \dots, k_n , where $k_i = (x_i, y_i)$. The number of sampled points should be large enough to capture significant structure of the interest shape. The appropriate number of sampled points, n found experimentally for this work is 100, which is sufficient for all examples we have tested. Fig. 4 shows the sampled points on the contour.

B. Affine invariant feature extraction

The aim of this section is to extract feature of the shape's contour that is invariant to affine distortion caused by changing viewpoint. Usually, a planar object's shape observed by a camera undergo arbitrary position is the class of perspective transformed shape. From [8], it is shown that a perspective transformation can be approximated by an affine transformation as long as the imaged object is planar and the camera optical center is far enough from such plane.



Fig. 2. The extracted contour



Fig. 3. The digitized contour smoothed by Gaussian filter a) The original contour b) The smoothed contour



Fig. 4. The contour's sampled points

The affine transform included translation, scaling, rotation, and shear can be mathematically described in matrix as explained in [9] as follows.

If $T: R^2 \rightarrow R^2$ is the matrix operator whose standard matrix is

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad (1)$$

then
$$T\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} \quad (2)$$

where $T\left(\begin{bmatrix} x \\ y \end{bmatrix}\right)$ is the affine-transformed version of $\begin{bmatrix} x \\ y \end{bmatrix}$.

In our case $\begin{bmatrix} x \\ y \end{bmatrix}$ or $T\left(\begin{bmatrix} x \\ y \end{bmatrix}\right)$ is the coordinate vector of points on the contour. a, b, c and d are the coefficients of scaling, rotation, and shear. e and f are the coefficients of translation.

As mentioned in the introduction, we apply the 1-D triangle area representation [6] as the affine invariant descriptor in spatial domain. The concept of the 1-D triangle area representation is derived from the theory of determinant in [10]

$$\{\text{area of } T(S)\} = \left| \det \begin{bmatrix} a & b \\ c & d \end{bmatrix} \right| \cdot \{\text{area of } S\} \quad (3)$$

where $T: R^2 \rightarrow R^2$ is the linear transformation determined by a 2×2 matrix in (1). S is a parallelogram in R^2 defined by the points X_1, X_2, X_3 , and X_4 corresponding to coordinates $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ and (x_4, y_4) respectively as shown in Fig. 5.

From (2), translating a parallelogram (S) by $\begin{bmatrix} e \\ f \end{bmatrix}$ does not effect its area.

Considering the sampled points of a contour, k_1, k_2, \dots, k_n obtained from the previous section, the triangle area is calculated instead of the parallelogram. Let any three adjacent sampled points, k_1, k_2, k_3 be the vertices of a triangle as shown in Fig. 6. The triangle area can be calculated by:

$$A = \frac{1}{2} \begin{vmatrix} x_{k_1} & y_{k_1} & 1 \\ x_{k_2} & y_{k_2} & 1 \\ x_{k_3} & y_{k_3} & 1 \end{vmatrix} \quad (4)$$

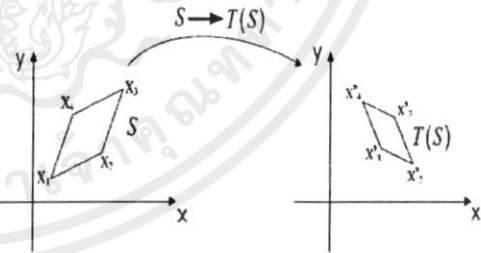


Fig. 5. A Parallelogram (S) is transformed to $T(S)$.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

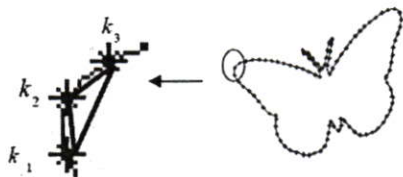


Fig. 6. Three adjacent sampled points on the contour used to calculate the triangle area.

To obtain the 1-D triangle area representation, we calculate the triangle areas of all sampled points. Let $AREA = \{a_1, a_2, \dots, a_n\}$ be the set of areas, where n denotes the number of the sampled points ($n=100$). The area values are plotted in Fig. 7c-d (the vertical and horizontal axis represent the area values and the sampled points on contour respectively). It is noticed that the characteristic of 1-D triangle area representations plotted from the same shapes captured by a camera in different viewpoints (Fig. 7a-b) are similar. Then, the 1-D triangle area representation of each contour is normalized as shown in Fig. 7e-f.

C. Matching stage

To match the query shape to the database, the normalized cross-correlation between the 1-D triangle area representation of the query shape (denoted by $Q = \{q_1, q_2, \dots, q_n\}$) and each 1-D representation of the database shapes (denoted by $D = \{d_1, d_2, \dots, d_n\}$) is taken into account. The normalized cross-correlation function is defined by:

$$R(Q, D) = \max_j \frac{\sum_{i=1}^n q_i d_{i+j}}{\sqrt{\sum_{i=1}^n q_i^2 \sum_{i=1}^n d_i^2}}; \text{ for } j = 1, 2, 3, \dots, n \quad (5)$$

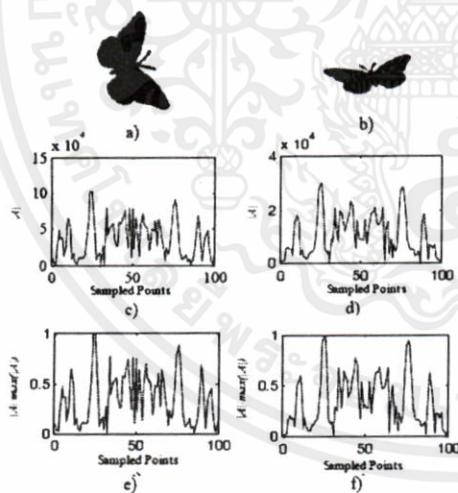


Fig. 7. (a)-(b) the same shapes captured in different viewpoints. (c)-(d) the 1-D triangle area representation of (a) and (b). (e)-(f) the 1-D triangle area representations are normalized.

To be invariant to the starting point, the value of $R(Q, D)$ at each matching pair (the query shape and one of the database shapes) is found from the maximum value calculated at j^{th} time-step. Then, the query shape will be assigned to one of the database shapes corresponding to the maximum values of the normalized cross correlation.

D. Retrieval stage

When $R(Q, D)$ has been computed for all shapes in the database. The system returns a list of the database shapes in decreasing order of the R values. Table I shows the retrieval results obtained from 3 random queries and each retrieval shape displays with the normalized cross-correlation value.

III. EXPERIMENTAL RESULTS

The MPEG-7 dataset from <http://www.cis.temple.edu/~latecki> has been used in many researches. Unfortunately, this dataset does not contain shapes in the presence of perspective distortion, thus it is not suitable for testing the performance of our affine invariant algorithm. So, we construct the multi-view shape dataset which consists of 40 classes of shapes chosen from the MPEG-7 dataset. Each class has 14 different perspective distorted shapes. To create this dataset, we have printed those 40 shapes on white papers then used a digital camera to capture those ones in the 14 different viewpoints. Fig. 8 shows the samples from the class "bird". This multi-view shape dataset has a total of 560 shapes.

A. Best-match accuracy

We compared the matching results between unsmoothed and smoothed contours of database shapes to show how much this algorithm can reduce the effect of digitization noise and segmentation errors. The best-match accuracy of the system was evaluated on the multi-view shape dataset. Each shape in the database was used as a test query, and the match to itself was excluded from the evaluation. The number of queries for the closest match identified the same class as the query was counted. For the 560-shapes database with the smoothed contours, the closest matches were recalled at

TABLE I

The retrieval results of the queries. The crosses indicate the wrong retrievals.

Query	Retrievals						
	1	0.9919	0.9403	0.9316	0.9205	0.9188	0.9115
	0.0059	0.8940	0.8924	0.8782	0.8446	0.8446	0.8371
	1	0.8770	0.8617	0.8322	0.8173	0.8181	0.8017
	0.7952	0.7311	0.7252	0.7215	0.7107	0.6928	0.6973
	1	0.9845	0.9549	0.9469	0.9440	0.9438	0.9277
	0.8949	0.8814	0.8639	0.8495	0.8331	0.8172	0.8151

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Fig. 8. Samples for the class "bird".

548 times (97.86%) achieved 3.22 % higher than the unsmoothed contours at 530 times (94.64%).

B. Retrieval performance

We performed the retrieval results using the precision-recall curve calculated over the multi-view shape dataset. Each shape in the database was used in turn as the query. To evaluate the effectiveness of retrieval for the unsmoothed and smoothed contour set of the multi-view shape dataset, we computed:

- 1) Precision is the percentage of qualifying (similar) shapes retrieved with respect to the total number of retrieved shapes.
- 2) Recall is the percentage of qualifying shapes retrieved with respect to the total number of similar shapes in the database.

In Fig. 9, we presented a precision-recall plot to compare the retrieval performance of the unsmoothed and smoothed contour set. The precision-recall curve was plotted by averaging precision and recall over all contours of the database. The top-left point of a precision-recall curve corresponded to the precision-recall values for the best answer or best match (The first retrieval is always the query itself). It can be noticed that the smoothed contours outperforms the unsmoothed contours in the retrieval graph.

C. Real world applications

Our method can be used to apply with many applications, for instance, applying our method to find similar shapes between images taken from different viewpoints which can adapt for image registration. Fig. 10 shows the best 10 matches and 2 mismatches between the set of shapes (each interested shape is surrounded by yellow border) extracted from the pair of "Egypt" images, each of which is capture from different viewpoints. Notice that the mismatched shapes are very similar.

IV. CONCLUSION AND FUTURE WORK

This paper proposed a method for matching and retrieving shapes in the presence of affine distortion

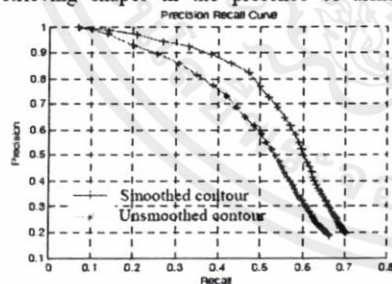


Fig. 9. Precision-Recall graph

utilizing the 1-D triangle area representation. The experiments on the proposed method were performed with the multi-view shape database. The matching accuracy showed 97.86% accuracy and robustness to noise. The retrieval performance was visualized as a precision-recall curve. Moreover, we demonstrated the application of our method in finding similar shapes between the images taken from different viewpoints, which led to image registration technique. Research on this promising approach to shape matching and retrieval is on going, focusing on improving an algorithm that is robust to occlusion.

REFERENCES

- [1] P. Otterloo. *A Contour-Oriented Approach to Shape Analysis*. Prentice Hall Int., 1991. pp. 5-8.
- [2] S. Loncaric, "A survey of shape analysis techniques," *Patt. Recog.*, vol. 31, no. 8. pp. 983-1001, 1998.
- [3] D. Zhang and G. Lu, "Review of shape representation and description techniques," *Patt. Recog.*, vol. 37, no. 1. pp. 1-19, 2004.
- [4] D. W. Thompson and J. L. Mundy, "Three dimensional model matching from an constrained viewpoint," in *Proc. Conf. Robotics and Automation, Raleigh, NC*, pp. 208-220, 1987.
- [5] T. Miyatake, T. Matsuyama, and M. Nagao, "Affine transform invariant curve recognition using Fourier descriptors (original in Japanese)," *Trans. Inform. Processing Soc. Japan*, vol. 24, no. 1. pp. 64-71, 1983.
- [6] I. El Rubei, M. Ahuned, and M. Kamel, "Affine Invariant Multiscale Wavelet-based Shape Matching Algorithm", *Proceeding of the First Canadian Conference on Computer and Robot Vision*, pp. 217-224, May 2004.
- [7] W.K.Pratt. "Digital Image Processing". John Wiley and sons, Inc, third edition, 2001. pp. 580-583.
- [8] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000.
- [9] H. Anton and C. Rorres "Elementary Linear Algebra". Wiley and Sons, Inc, ninth edition, 2005. pp. 458-459.
- [10] David C. Lay "Linear Algebra and Its Applications". Pearson Education, Inc, third edition, 2006. pp. 205-207.

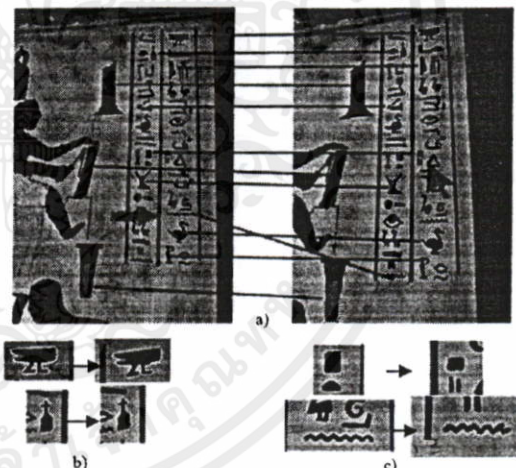


Fig. 10. a) Views of the "Egypt" scenes. The lines connect the corresponding shapes and the arrows indicate the shape mismatches. b) Zoomed versions of the correct matches. c) Zoomed versions of the two mismatches.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Multi-view Shape Matching and Retrieval

Kritsana Yawichai and Yuttana Kitjaidure

Department of Electronics

Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang
3 Moo. 2 Chalongkrung Rd., Ladkrabang Bangkok, 10520 Thailand

Tel: +662 3264222-122, Fax: +662 7392398, E-mail: astronaut_aor@hotmail.com and kyuttan@kmitl.ac.th

Abstract

In real world situation when we capture the planar object in multiple viewpoints, we normally obtain each shape distorted in various affine transform. This situation inspires us to develop affine invariant shape feature to be efficiently used in shape matching and retrieval. This work presents the affine invariant shape matching and retrieval via the 1-D triangle area representation at different triangle side length (TSL). Our method has been evaluated over a number of affine distorted shapes. The experimental results show that our relatively simple scheme is effective and robustly tolerates translation, scaling, shear, orientation, starting point selection and noise.

Keywords:

Affine transformation, Shape matching and retrieval

1. Introduction

Shapes in images play an important role in object identification methods used in robot vision and image processing systems. The increasing recognition performance of robot vision as well as human do is still a challenging problem for many researchers. This research presents an approach to solve the problem. As mentioned before, shape is an important visual feature so we use shape of the object in matching and retrieval process. Shape description techniques can be divided into contour-based and region-based. In [1], contour-based representation uses less data points and is usually less sensitive to noise than the region-based techniques. For this reason, we decide to choose the contour-based description technique in our work. Several contour features have been introduced in the literature [2], [3].

The appropriate descriptor of shape feature should not change under perspective distortion which typically arises if a planar object is observed under arbitrary viewpoints with respect to the plane. However, the affine transformation may be used as approximation to the perspective transformation rather than the use of similarity transformation [4]. Miyatake, Matsuyama, and Nagao [5] proposed an optimizing procedure to produce invariance under affine transformations. Ibrahim El Rubei, Maher Ahmed and Mohamed Kamel [6] had investigated affine invariant shape matching based on the dyadic wavelet transform (DWT) which was used to transform contour of shape from spatial domain into frequency domain. The frequency components

are composed of approximation coefficients and detail coefficients. Then, the curve moment invariant and the absolute 1-D triangle area representation are used with those two types of coefficients to measure dissimilarity and similarity of each shape's contour respectively. It can be seen that most of the previous works with the shape classification are very computationally expensive. The simple algorithm developed in this paper has adopted the affine invariant shape descriptor with the absolute 1-D triangle area representation in [6] to be the non-absolute one and used it in spatial domain instead of frequency domain. Moreover, we detect fine to coarse feature of shape by computing this shape descriptor at different triangle side length (TSL). Our algorithm is not only uncomplicated but also works reasonably well in the presence of moderate amount of noise.

The rest of this paper is organized as follows. Section 2 introduces the proposed algorithm with detailed description to all stages. In section 3, the experimental results are obtained. Finally, Section 4 gives conclusions and future work.

2. Approach and Methods

In this section, we present our methodology to achieve shape matching and retrieval. The overall process is shown in Figure 1. First, the shape contours of both query and database shapes are extracted. Then, they are smoothed by a Gaussian filter and sampled with the same number of points, n . Next, the sampled points of each contour are calculated to obtain the 1-D triangle area representation at various TSLs, each of which is an affine invariant descriptor. Finally, shape matching and retrieval are achieved by computing the normalized cross-correlation of the 1-D triangle area representations at the same TSL of the query shape and database shapes. The detailed descriptions of all stages are discussed in the next section.

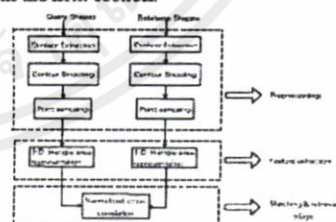


Figure 1- The proposed process

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1. Preprocessing

Contour Extraction

The shape is segmented from the image. Then, the contour following technique [7] is applied to the shape to obtain the contour as a sequence of points $(x(u), y(u))$, which is the coordinates of each point on the contour. Figure 2 shows the result of the extracted contour.

Contour Smoothing

To avoid effect of noises caused by digitization noise and segmentation errors, the contour sequences $(x(u), y(u))$ are smoothed by a Gaussian filter with σ set to 1.5% of N, where N is the total number of coordinate points on the considered contour. Figure 3 illustrates the result of smoothing the contour sequences $(x(u), y(u))$.

Point Sampling

Every contour (query and database contours) is sampled at the same number of points, n . The sampled points will be denoted by k_1, k_2, \dots, k_n , where $k_i = (x_i, y_i)$. The number of sampled points should be large enough to capture significant structure of the interest shape. The appropriate number of sampled points, n found experimentally for this work is 100, which is sufficient for all examples we have tested. Figure 4 shows the sampled points on the contour.



Figure 2-The extracted contour



Figure 3-The digitized contour smoothed by Gaussian filter
a) The original contour b) The smoothed contour



Figure 4- The contour's sampled points

2.2. Affine Invariant Feature Extraction

The aim of this section is to extract feature of the shape's contour that is invariant to affine distortion caused by changing viewpoint. Usually, a planar object's shape observed by a camera undergo arbitrary position is the class of perspective transformed shape. From [8], it is shown that a perspective transformation can be approximated by an affine transformation as long as the imaged object is planar

and the camera optical center is far enough from such plane. The affine transform included translation, scaling, rotation, and shear can be mathematically described in matrix as explained in [9] as follows.

If $T: R^2 \rightarrow R^2$ is the matrix operator whose standard matrix is

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad (1)$$

then $T\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix}$ (2)

where $r\left(\begin{bmatrix} x \\ y \end{bmatrix}\right)$ is the affine-transformed version of $\begin{bmatrix} x \\ y \end{bmatrix}$.

In our case $\begin{bmatrix} x \\ y \end{bmatrix}$ or $r\left(\begin{bmatrix} x \\ y \end{bmatrix}\right)$ is the coordinate vector of points on the contour. a, b, c and d are the coefficients of scaling, rotation, and shear. e and f are the coefficients of translation.

As mentioned in the introduction, we apply the 1-D triangle area representation [6] as the affine invariant descriptor in spatial domain. The concept of the 1-D triangle area representation is derived from the theory of determinant in [10]

$$\{\text{area of } T(S)\} = \left| \det \begin{bmatrix} a & b \\ c & d \end{bmatrix} \right| \cdot \{\text{area of } S\} \quad (3)$$

where $T: R^2 \rightarrow R^2$ is the linear transformation determined by a 2×2 matrix in Equation (1). S is a parallelogram in R^2 defined by points X_1, X_2, X_3 , and X_4 corresponding to coordinates $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ and (x_4, y_4) respectively as shown in Figure 5.

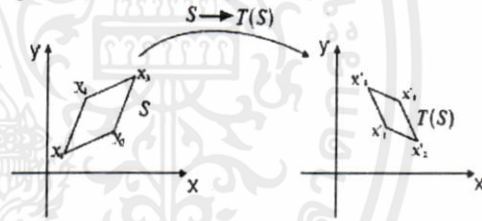


Figure 5-A Parallelogram (S) is transformed to T(S)

From Equation (2), translating a parallelogram (S) by $\begin{bmatrix} e \\ f \end{bmatrix}$ does not effect its area.

Considering the sampled points of a contour, k_1, k_2, \dots, k_n obtained from the previous section, the triangle area is calculated instead of the parallelogram. Let any three sampled points, k_{p-1}, k_p, k_{p+1} be the vertices of a triangle as shown in figure 6. The triangle area can be calculated by:

$$A_t(p) = \frac{1}{2} \begin{vmatrix} x_{k_{p-1}} & y_{k_{p-1}} & 1 \\ x_{k_p} & y_{k_p} & 1 \\ x_{k_{p+1}} & y_{k_{p+1}} & 1 \end{vmatrix} \quad (4)$$

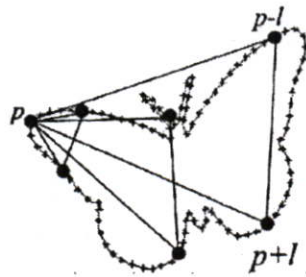


Figure 6- Any three sampled points on the contour (at p^{th} sampled point) used to calculate the area at any TSL.

where p is the sampled points, $p=1,2,\dots,100$. l is the triangle side length (TSL). We have used $l=1, 10, 20, 30, 40$ and 49 in this paper. Each TSL is obtained by considering l sampled points further from p . The reason for varying TSL is to find the optimum length that best represent feature of shape. Then, we obtain the 2 types of 1-D triangle area representation which are the absolute and non-absolute value of Equation (4) respectively. Both representations were plotted at various TSLs as shown in Figure 7c-d (the vertical and horizontal axis represent the normalized area values and the sampled points on the contour respectively). It is noticed that the characteristic of 1-D triangle area representations plotted from the same shapes captured by a camera in different viewpoints (Figure 7a-b) are similar.

We will compare the effectiveness of the absolute and non-absolute 1-D triangle area representation at different TSL in section 3.

2.3. Matching stage

To match the query shape to the database, the normalized cross-correlation values of the 1-D triangle area representation at the same TSL between the query shape (denoted by $Q = \{q_1, q_2, \dots, q_n\}$) and each of the database shapes (denoted by $D = \{d_1, d_2, \dots, d_n\}$) are computed.

$$R(Q, D) = \max_j \frac{\sum_{i=1}^n q_i d_{i,j}}{\sqrt{\sum_{i=1}^n q_i^2 \sum_{i=1}^n d_i^2}}; \text{ for } j=1,2,3,\dots,n \quad (5)$$

To be invariant to the starting point, the value of $R(Q, D)$ at each matching pair (the query shape and one of the database shapes) is found from the maximum value calculated at j^{th} time-step. Then, the query shape will be assigned to one of the database shapes corresponding to the maximum values of the normalized cross correlation.

2.4. Retrieval stage

When $R(Q, D)$ has been computed for all shapes in the database, the system returns a list of the database shapes in

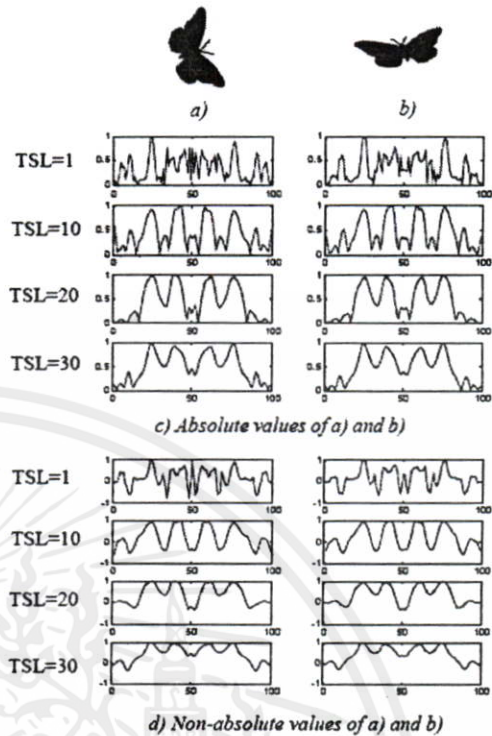


Figure 7-(a)-(b) the same shapes captured in different viewpoints. (c) the absolute (d) non-absolute 1-D triangle area representation of (a) and (b) respectively.

decreasing order of the R values. Table 1 shows the retrieval results (which is the result of the absolute 1-D triangle area representation at TSL=1) obtained from 4 random queries. Each retrieval shape displays with the normalized cross-correlation values.

Table 1- The retrieval results of the queries. The crosses indicate the wrong retrievals.

Query	Retrievals

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Experimental Results and Discussion

The MPEG-7 dataset [11] has been used in many researches. Unfortunately, this dataset does not contain shapes in the presence of perspective distortion, thus it is not suitable for testing the performance of our affine invariant algorithm. So, we construct the multi-view shape dataset which consists of 40 classes of shapes chosen from the MPEG-7 dataset. Each class has 14 different perspective distorted shapes. To create this dataset, we have printed those 40 shapes on white papers then used a digital camera to capture those ones in the 14 different viewpoints. Figure 8 shows the samples from the class "butterfly". This multi-view shape dataset has a total of 560 shapes.



Figure 8- The class "butterfly".

3.1. Best-match Accuracy

The best match accuracy of the system was evaluated on the multi-view shape dataset. Each shape in the database was used as a test query, and the match to itself was excluded from the evaluation. The number of queries for the closest match identified the same class as the query was counted. We compared the matching accuracy between unsmoothed and smoothed contours evaluated on the absolute 1-D triangle area representation at triangle side length = 1. The closest matches for the database with smoothed contours were recalled correctly at 97.86% better than the unsmoothed contours, 94.64%. This result demonstrated that the smoothing contour can decrease the effect of noise. As a result, the 1-D triangle area representation of the smoothed contour was used in this work. The matching performance of both absolute and non-absolute 1-D triangle area representation at different side length were measured and listed in Table 2. We observed that the 1-D triangle area representations at all TSL gave pretty high matching performance. The non-absolute 1-D triangle area representation performed slightly better than the absolute 1-D triangle area representation.

Table 2- The matching results.

Triangle Side Length	The 1-D Triangle Area Representation Type	
	Absolute	Non-absolute
1	97.86%	99.29%
10	99.11%	99.82%
20	99.82%	99.82%
30	99.82%	99.82%
40	99.82%	99.82%
49	99.82%	99.82%

3.2. Retrieval Performance

We performed the retrieval results using the precision-recall

curve calculated over the multi-view shape dataset. Each shape in the database was used in turn as the query Q . Let define the retrieval results obtained from the arrangement in decreasing order of the R values as 'retrieval(Q, m)', m is the order of retrieval results (1-560). Let 'relevant(Q)' denote the set of shapes that are grouped in the same class of Q (14 shapes). The precision and the recall are defined by:

$$precision(Q, m) = \frac{\text{the cardinality of } PR}{m} \quad (6)$$

$$recall(Q, m) = \frac{\text{the cardinality of } PR}{14} \quad (7)$$

where PR is the set, $\{retrieval(Q, m) \cap relevant(Q)\}$. We plotted the precision recall curve by averaging precision and recall over all Q as shown in Figure 9. To see the differences of precision recall curve obviously, only curve of TSL=1, 20 and 49 were depicted.

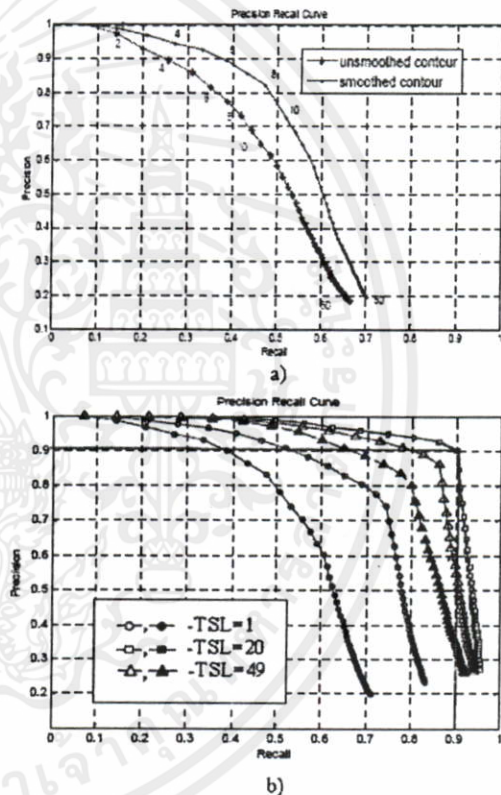


Figure 9-Precision recall curves plotted with the top-50 retrievals ($m=1-50$) and labeled by m . a) The retrieval performance of smoothed and unsmoothed contour which were evaluated on the absolute 1-D triangle area representation at triangle side length = 1. b) The retrieval performance at different triangle side length of the non-absolute and the absolute 1-D triangle area representation indicated with unfilled and filled symbols respectively.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The smoothed contour outperformed the unsmoothed contour in retrieval performance shown in Figure 9a. Figure 9b showed that at each triangle side length, the non-absolute 1-D triangle area representation achieved higher accuracy than the absolute one at all recall values. The examples of retrieval result of 4 random query and their top 14 retrieval shapes were shown in Table 1.

From the retrieval experiment, it is observed that when TSL is small, we will obtain the local feature of shape which is quite sensitive to boundary noise and trivial details. When TSL is too large the retrieval performance is also declined due to the fact that only rough structure of shape is obtained. We have found that the triangle side length at 20 gives satisfactory results, since it can detect optimal global feature of shape which makes our algorithm insensitive to noise. At this triangle side length, when $m=14$ the precision is higher than 90.37% and the recall higher than 90.37%. This means that the possibility of the top 14 retrieval shapes which are from the same class of query is 90.37%.

3.3. Real World Applications

Our method can be used to apply with many applications, for instance, applying our method to find similar shapes between images taken from different viewpoints which can adapt for image registration. Figure 10 shows the best 10 matches and 2 mismatches between the set of shapes (each interested shape is surrounded by yellow border) extracted from the pair of "Egypt" images, each of which is captured from different viewpoints. Notice that the mismatched shapes are very similar.

4. Conclusion and Future Work

This paper proposed a method for matching and retrieving shapes in the presence of affine distortion utilizing the 1-D triangle area representation. The experiments on the proposed method were performed with the multi-view shape database. The experimental results illustrated that our approach is well suited to shape matching and retrieval on shape with moderate amounts of noise and distortion. The performance of our approach also achieved high precision and recall. Moreover, we demonstrated the application of our method in finding similar shapes between the images taken from various viewpoints, which led to image registration technique. Research on this promising approach to shape matching and retrieval is on going, focusing on improving an algorithm that will further enhance its performance and system's speed.

References

- [1] P. Otterloo. 1991. *A Contour-Oriented Approach to Shape Analysis*: Prentice Hall Int.
- [2] S. Loncaric. 1998. A survey of shape analysis techniques. *Patt. Recog.*, 31(8):983-1001.
- [3] D. Zhang and G. Lu. 2004. Review of shape representation and description techniques. *Patt. Recog.*,

37(1):1-19.

- [4] D. W. Thompson and J. L. Mundy. 1987. Three dimensional model matching from a constrained viewpoint, in Proc. Conf. Robotics and Automation, Raleigh, NC.
- [5] T. Miyatake, T. Matsuyama, and M. Nagao. 1983. Affine transform invariant curve recognition using Fourier descriptors (original in Japanese), *Trans. Inform. Processing Soc. Japan*, vol. 24, no. 1, pp. 64-71.
- [6] I. El Rube', M. Ahmed, and M. Kamel. 2004. Affine Invariant Multiscale Wavelet-based Shape Matching Algorithm. Proceeding of the First Canadian Conference on Computer and Robot Vision, 217-224.
- [7] W.K.Pratt. 1991. *Digital Image Processing*: John Wiley and sons, Inc, second edition.
- [8] R. Hartley and A. Zisserman. 2000. *Multiple View Geometry in Computer Vision*: Cambridge University Press.
- [9] H. Anton and C. Corres. 2005. *Elementary Linear Algebra*: Wiley and Sons, Inc, ninth edition.
- [10] David C. Lay. 2006. *Linear Algebra and Its Applications*: Pearson Education, Inc, third edition.
- [11] MPEG-7 dataset. Citing Internet sources URL <http://www.cis.temple.edu/~latecki/>.

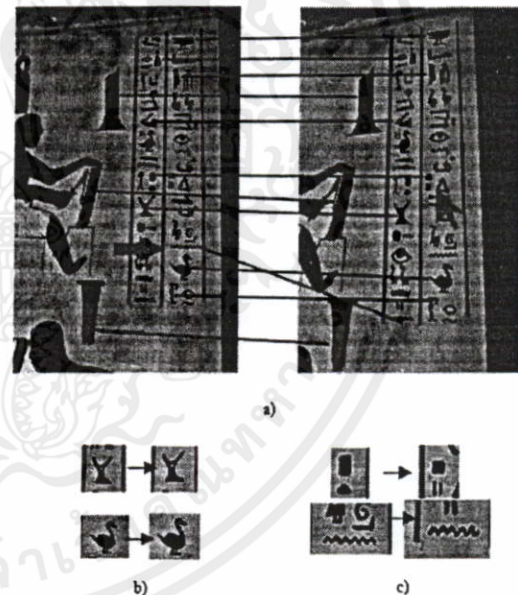


Figure 10.-a) Views of the "Egypt" scenes. The lines connect the corresponding shapes and the arrows indicate the shape mismatches. b) Zoomed versions of the correct matches. c) Zoomed versions of the two mismatches.

**JICT
2007**

Proceedings
Joint International Conference
on Information Communication Technology



JICA



**TOKAI
UNIVERSITY**



**MEIJI
UNIVERSITY**

**ECTI
Association**

日本シミュレーション学会
JSST
 Japan Society for Simulation Technology

19-22 December 2007
Donchan Palace Hotel, Vientiane, Lao PDR



CT JICT JICT JICT JICT JICT JICT
 007 2007 2007 2007 2007 2007 2007

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FAST CLASSIFICATION OF MULTI-VIEW SHAPE USING NEURAL NETWORK

Kritsana Yawichai and Yuttana Kitjaidure

Department of Electronics

Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang

3 Moo. 2 Chalongkrung Rd., Ladkrabang Bangkok, 10520 Thailand

E-mail: astronaut_aor@hotmail.com and kkyuttan@kmitl.ac.th

ABSTRACT

Several recently published shape classification systems employ various method have achieved high accuracy. However, some of these systems based on pairwise shape matching technique have faced a time consuming problem when evaluated on a large database. So this drawback makes the system impractical for real-time applications. Motivated by this obstacle, we have investigated a novel and robust neural network solution to achieve high speed of shape classification without sacrificing accuracy via various side lengths of 1-D triangle area representations (TA). Our method has been evaluated over a number of affine distorted shapes. The experimental results demonstrate that a shape classification system using neural networks can achieve high accuracy and high speed comparable with the prior system.

1. INTRODUCTION

In the field of machine vision and image processing systems, shape plays a significant role in object characterization which is commonly used in object classification, image registration and so on. In addition, the shape classification is also an important part of machine intelligence that is useful for both decision-making and data processing. More importantly, developing suitable shape feature is also a central problem in shape classification. The appropriate descriptor of shape feature should not change under perspective distortion which typically arises if a planar object is observed under arbitrary viewpoints with respect to the plane. However, the affine transformation may be used as an approximation to the perspective transformation rather than the use of similarity transformation [1]. Shape description techniques are mainly divided into contour-based and region-based. In [2], the contour-based representation uses less data points and is usually less sensitive to noise than the region-based techniques. For this reason, we decide to choose the contour-based description technique in our work. Several contour features have been introduced in the literature [3], [4]. Farzin Mokhtarian and Alan K. Mackworth [5] have presented a multiscale, curvature-based shape descriptor technique for classification planar curves. This algorithm is computationally intensive due to the huge number of scales it requires. Haibin Ling and

Dawid W. Jacobs [6] have used the inner-distance to build shape descriptors for shape matching through dynamic programming. It can be seen that the previous proposed works have employed complex shape descriptor and matcher to achieve high accuracy. To alleviate this problem, the work presented in [7] has proposed a relatively simple shape descriptor, the absolute 1-D triangle area representation (ATA) to match and retrieve shape using normalized cross-correlation function. Although ATA offers a good result, there are rooms for improvement. Since this ATA has low discrimination capability and faces major difficulties in discriminating among those indistinct shape descriptors. Furthermore, similarity measure using normalized cross-correlation function is the pairwise matching between query shapes and database shapes, which takes the identification procedure much time consumption when evaluated on a large database.

Our aim in this research is to improve ATA by varying triangle side length (TSL) to find optimum TSL, which gives good discriminatory power. Moreover, we increase the speed of shape classification without sacrificing accuracy by using neural networks. This technique is simple, computationally cost effective and can be easily implemented. Our method has been evaluated over a number of affine distorted shapes captured from an arbitrary viewpoint. We conduct the experiment to compare method presented in this paper with the previous work [7]. The algorithm presented here has high accuracy and is significantly faster than the previous work.

In this paper, the approach and method with detailed description to all processes are given in section 2. In section 3, the experimental results are obtained. Finally, section 4 gives conclusions and future work.

2. APPROACH AND METHOD

2.1. Preprocessing

First, the shape is segmented from the image. Then, contour following technique [8] is applied to the shape to obtain the contour as a sequence of points $(x(u), y(u))$, which is the coordinates of each point on the contour. To avoid effect of noise caused by digitization noise and segmentation errors, the contour sequences $(x(u), y(u))$ are smoothed by a Gaussian filter with σ set to 1.5% of the total number of coordinate points on the considered

contour, N . Finally, all contours (query and database contours) are sampled at the same number of points, n . The sampled points will be denoted by k_1, k_2, \dots, k_n , where $k_i = (x_i, y_i)$. The number of sampled points should be large enough to capture significant structure of the interest shape. The appropriate number of sampled points, n found experimentally for this work is 100, which is sufficient for all examples we have tested. The preprocessing steps are shown in Figure 1.

2.2. Affine Invariant Feature Extraction

The aim of this section is to extract feature of the shape's contour that is invariant to affine distortion caused by changing viewpoint. Usually, a planar object's shape observed by a camera undergo arbitrary position is the class of perspective transformed shape. From [9], it is shown that a perspective transformation can be approximated by an affine transformation as long as the imaged object is planar and the camera optical center is far enough from such plane. The affine transform included translation, scaling, rotation, and shear can be mathematically described in matrix model explained in [10] as follows.

If $T: R^2 \rightarrow R^2$ is the matrix operator whose standard matrix is

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad (1)$$

then $T\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix}$ (2)

where $r\left(\begin{bmatrix} x \\ y \end{bmatrix}\right)$ is the affine-transformed version of $\begin{bmatrix} x \\ y \end{bmatrix}$.

In our case $\begin{bmatrix} x \\ y \end{bmatrix}$ or $r\left(\begin{bmatrix} x \\ y \end{bmatrix}\right)$ is the coordinate vector of points on the contour. a, b, c and d are the coefficients of scaling, rotation, and shear. e and f are the coefficients of translation.

As mentioned in the introduction, we apply the 1-D triangle area representation (TA) derived from the theory of determinant in [11]

$$\{\text{area of } T(S)\} = \left| \det \begin{bmatrix} a & b \\ c & d \end{bmatrix} \right| \cdot \{\text{area of } S\} \quad (3)$$

where $T: R^2 \rightarrow R^2$ is the linear transformation determined by a 2×2 matrix in Equation (1). S is a parallelogram in R^2 defined by points X_1, X_2, X_3 , and X_4 corresponding to coordinates $(x_1, y_1), (x_2, y_2), (x_3, y_3)$, and (x_4, y_4) respectively as shown in Figure 2.

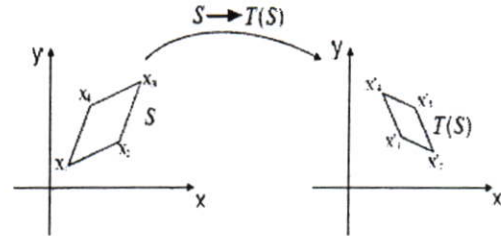


Figure 2. A Parallelogram (S) is transformed to $T(S)$

From Equation (2), translating a parallelogram (S) by $\begin{bmatrix} e \\ f \end{bmatrix}$ does not affect its area.

Considering the sampled points of a contour, k_1, k_2, \dots, k_n obtained from the previous section, the triangle area is calculated instead of the parallelogram. Let any three sampled points, k_{p-l}, k_p, k_{p+l} be the vertices of a triangle as shown in Figure 3. The triangle area can be calculated by:

$$A_l(p) = \frac{1}{2} \begin{vmatrix} x_{k_{p-l}} & y_{k_{p-l}} & 1 \\ x_{k_p} & y_{k_p} & 1 \\ x_{k_{p+l}} & y_{k_{p+l}} & 1 \end{vmatrix} \quad (4)$$

where p is the sampled point, $p=1,2,\dots,100$. l is the triangle side length (TSL) considering l sampled points further from p . Then, this triangle area is turn to be an absolute value to obtain the absolute 1-D triangle area representation (ATA). ATA at triangle side length= l (ATA(l)) is used in [7]. In our work, we evaluated on ATA at various TSLs, ATA(l). These ATA are plotted at TSL=1, 10, 20 and 30 in Figure 4c-d (the vertical and horizontal axis represent the normalized absolute area values and the sampled points on the contour respectively). It is noticed that the characteristic of 1-D triangle area representations plotted from the same shapes captured by a camera in different viewpoints (Figure 4a-b) are similar.

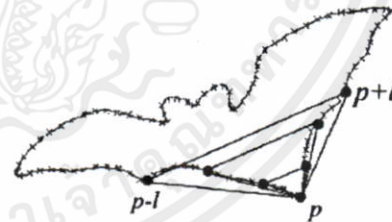
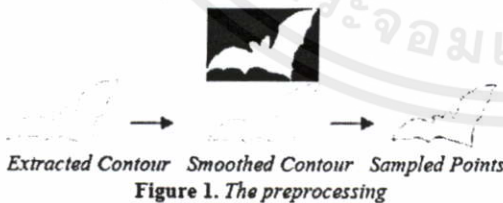


Figure 3. Any three sampled points on the contour (at p^{th} sampled point) are used to calculate the area at any TSL.



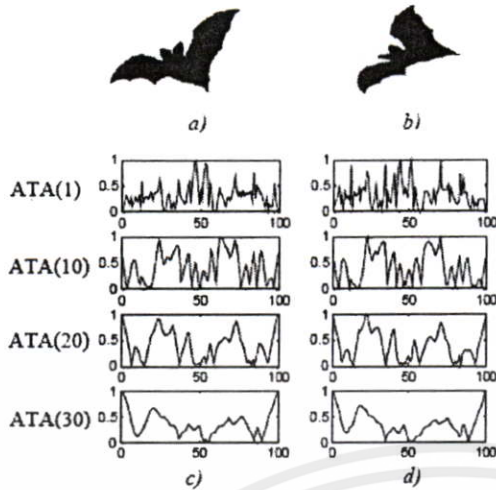


Figure 4. (a)-(b) the same shapes captured in different viewpoints. (c)-(d) are the absolute 1-D triangle area representations of (a) and (b) respectively.

2.3. Classification

In this section, we explain classification methods including normalized cross-correlation used and neural network.

2.3.1 Similarity Measure Using Normalized Cross-correlation

To match the query shape to the database, the normalized cross-correlation values of the TA between the query shape (denoted by $Q = [q_1, q_2, \dots, q_n]$) and each of the database shapes (denoted by $D = [d_1, d_2, \dots, d_n]$) are computed.

$$R(Q, D) = \frac{\sum_{i=1}^n q_i d_i}{\sqrt{\sum_{i=1}^n q_i^2 \sum_{i=1}^n d_i^2}} \quad (5)$$

Then, the query shape will be assigned to one class of the database shapes corresponding to the maximum values of the normalized cross correlation.

2.3.2 Similarity Measure Using Neural Networks

To consider discrimination capability of ATA at different TSL, we create neural networks [12] for each feature (ATA(l)). Then, every shape descriptor and its own primary shape class are used as an input and an output of the neural networks respectively. The network attempts to implement a mapping between an input pattern $I = [I_1, I_2, \dots, I_n]^T$ and a desired output pattern $O = [O_1, O_2, \dots, O_c]^T$. The feed forward back-propagation neural network with two hidden layers is used as shown in

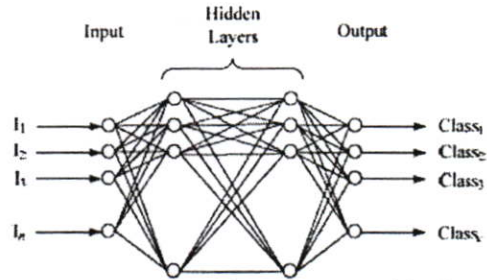


Figure 5. Neural network architecture used for shape classification.

Figure 5, with 100 input neurons, 70 neurons in each hidden layer, and 40 neurons in the output layer. Tangsigmod activation functions are used to all network layers except the non-active input layer.

In the training phase, the shape descriptors I^b ($c, b=1, \dots, M$ of M curve prototypes (c denotes the c^{th} class in the database) are used to train the supervised neural network. While the desired output $O(c)$ is the shape prototype class. The weight ω_j of the j^{th} neuron of each layer is updated in the k -th training cycle according to the following equation.

$$\Delta \omega_j(k+1) = \gamma \Delta \omega_j(k) + (1-\gamma) \alpha p_{k+1} \quad (6)$$

The parameters α and γ are the learning rate and the momentum factor respectively. The Scaled-Conjugated Gradient method is used for training.

$$r_{k+1} = -g_{k+1} \quad (7)$$

$$\beta_k = \frac{|r_{k+1}|^2 - r_{k+1} r_k}{p_k^T r_k} \quad (8)$$

$$p_{k+1} = r_{k+1} + \beta_k p_k \quad (9)$$

where p and g are the search direction and gradient respectively.

In the testing phase, the shape descriptor of the queries is used as an input to the neural network. The value of the network output is the similarity values. The classification results are obtained from the input classified to the shape class corresponding to the maximum network output.

3. EXPERIMENTAL RESULTS AND DISCUSSION

The MPEG-7 dataset [13] has been used in many researches. Unfortunately, this dataset does not contain shapes in the presence of perspective distortion, thus it is not suitable for testing the performance of our affine invariant algorithm. So, we construct the multi-view shape dataset which consists of 40 classes of shapes chosen from the MPEG-7 dataset. Each class has 14 different perspective distorted shapes. To create this dataset, we have printed those 40 shapes on white papers then used a digital camera to capture those ones in the 14 different viewpoints. Figure 6 shows the samples from the class



Figure 5. The class "bat".

"bat". This multi-view shape dataset has a total of 560 shapes. 7 views per class are actually used for training and the rest are the testing dataset. Our system is implemented in MATLAB version 7.3 running on a computer AMD Athlon 64x2 Dual core 2 GHz and RAM 1 GB.

3.1. Classification Accuracy

To demonstrate the benefit of using the neural network (NN), we compare the classification results with the pairwise matching utilizing normalized cross-correlation (NCC). They are evaluated on the same shape dataset (the multi-view shape dataset). The answered shape corresponding to the maximum network output or the maximum value of the normalized cross-correlation, which is identified to the same class as query, is counted. The comparison results for the evaluation on $ATA(l)$ are shown in Table 1 (to see the trend of accuracy, only the explicitly results of some TSL are depicted).

From the results, it is observed that the classification performance of ATAs are decreased when TSL is small. At the small TSL, we obtain only local features of shape, which are quite sensitive to boundary noises and trivial details. In the same way when TSL is too large, the classification performance are also declined due to the fact that only rough structure of the shape is obtained. We have found that the satisfactory results is obtained from the triangle side length at 20 since it can detect optimal global feature of the shape, which makes our algorithm insensitive to noise and more stable to handle distorted shape variation. As shown in Table 1, classification performance of the normalized cross-correlation is slightly better than the neural network. However, for the speed comparison, we have considered the computation time per 1 query of both method and found that the neural network takes only $83.93 \mu s$, which is less than the normalized cross-correlation ($39.5 \times 10^2 \mu s$). This can be concluded that the neural network increases the speed of shape classification without sacrificing accuracy and is more suitable to use with a large dataset.

3.2. Real World Applications

Our method can be used to apply with many applications, for instance, applying our method to find similar shapes between images taken from different viewpoints which can adapt for image registration. Figure 7 shows the best 12 matches and 2 mismatches between the set of shapes (each interested shape is surrounded by lighted border) extracted from the pair of "Egypt" images, each of which is captured from different viewpoints. Notice that the mismatched shapes are very similar.

Table 1. The comparison of the classification of ATA at various TSL through the normalized cross-correlation and the neural network.

ATA(<i>l</i>)	Accuracy of testing set %	
	NN	NCC
1	85.36	95.36
3	92.5	97.14
5	93.93	97.5
10	97.86	98.21
20	99.64	99.64
30	98.93	98.57
40	98.21	98.21
49	97.86	98.21

4. CONCLUSION AND FUTURE WORK

This paper has proposed the neural network for classifying shapes in the presence of affine distortion utilizing the 1-D triangle area representation. The experiments on the proposed method are performed with the multi-view shape database. The experimental results have illustrated that our approach achieves high speed and high accuracy even under a moderate range of visual transformation. Moreover, we have demonstrated the application of our method in finding similar shapes between the images taken from various viewpoints, which lead to image registration technique. Research on this promising approach to shape classification is on going focusing on reducing parameter of shape descriptor without sacrificing important feature to decrease the complication of the neural network's structure.

REFERENCES

- [1] D. W. Thompson and J. L. Mundy, "Three dimensional model matching from an constrained viewpoint," in *Proc. Conf. Robotics and Automation*, Raleigh, NC, pp. 208-220, 1987.
- [2] P. Otterloo. *A Contour-Oriented Approach to Shape Analysis*. Prentice Hall Int., 1991, pp. 5-8.
- [3] S. Loncaric, "A survey of shape analysis techniques," *Pat. Recog.*, vol. 31, no. 8, pp. 983-1001, 1998.
- [4] D. Zhang and G. Lu. "Review of shape representation and description techniques," *Pat. Recog.*, vol. 37, no. 1, pp. 1-19, 2004.
- [5] Farzin Mokhtarian and Alan K. Mackworth. "A Theory of Multiscale, Curvature-Based Shape Representation for Planar Curves," *IEEE Trans. Pattern Analysis and Machine Intelligent*, vol. 14, no. 8, pp. 789-805, Aug.1992.
- [6] Haibin Ling and Dawid W. Jacobs. "Shape Classification Using the Inner-Distance," *IEEE Trans. Pattern Analysis and Machine Intelligent*, vol. 29, no. 2, pp. 286-299, Feb.2007.

- [7] Kritsana Yawichai and Yuttana Kitjaidure, "A Simple and Efficient Algorithm for Affine Invariant Shape Matching," *International Conference on Engineering, Applied sciences, and Technology*, Bangkok, Thailand, 2007, in press
- [8] W.K.Pratt., "Digital Image Processing". John Wiley and sons, Inc, third edition, 2001, pp. 580-583.
- [9] R. Hartley and A. Zisserman, "Multiple View Geometry in Computer Vision," Cambridge University Press, 2000.
- [10] H. Anton and C. Rorres, "Elementary Linear Algebra," Wiley and Sons, Inc, ninth edition, 2005, pp. 458-459.
- [11] David C. Lay, "Linear Algebra and Its Applications". Pearson Education, Inc, third edition, 2006, pp. 205-207.
- [12] Martin T. Hagan and Howard B. Demuth, "Neural Network Design," International Thomson Publishing Company, 1995.
- [13] MPEG-7 dataset. Citing Internet sources URL <http://www.cis.temple.edu/~latecki/>.

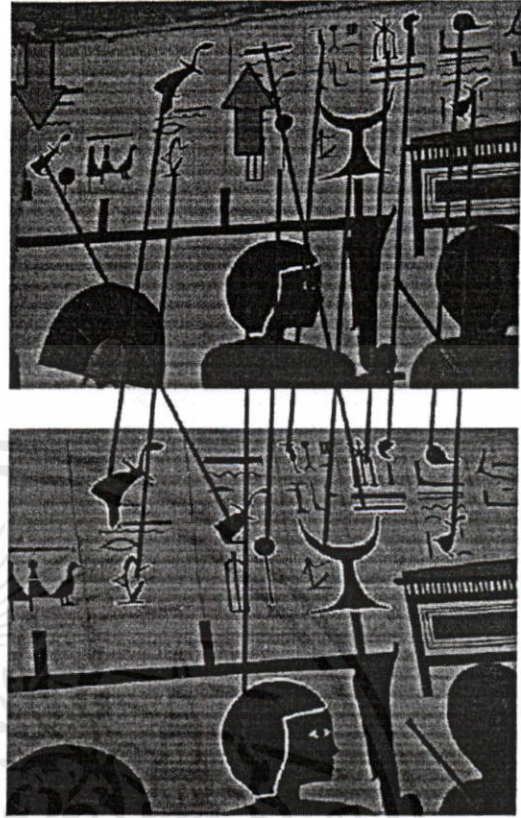
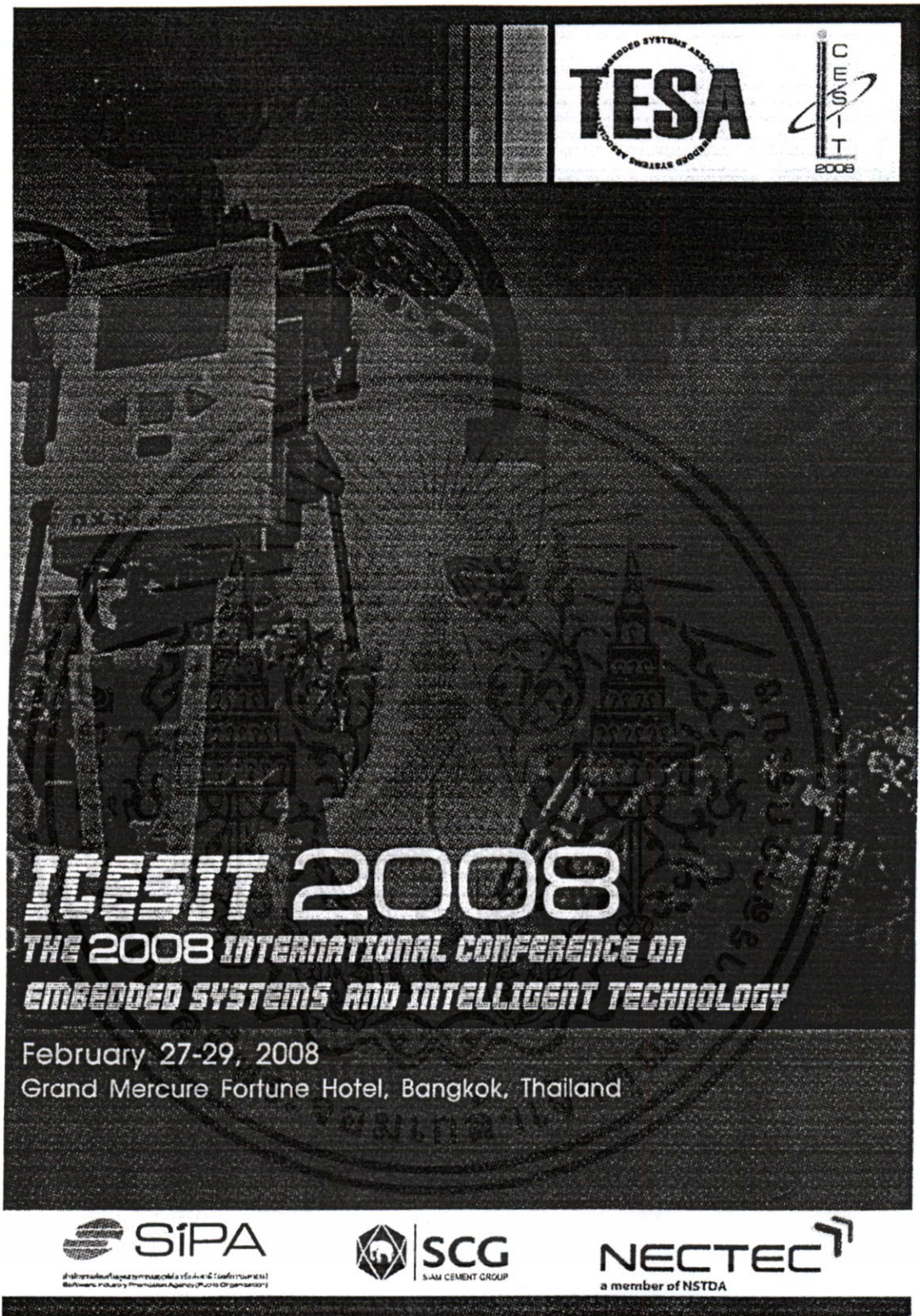


Figure 6. Views of the "Egypt" scenes. The lines connect the corresponding shapes and the arrows indicate the shape mismatches.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





EMBEDDED SYSTEMS ASSOCIATION
TESA
TECHNOLOGY ENHANCING SOCIETY

CEST
2008

IEST 2008
THE 2008 INTERNATIONAL CONFERENCE ON
EMBEDDED SYSTEMS AND INTELLIGENT TECHNOLOGY

February 27-29, 2008
Grand Mercure Fortune Hotel, Bangkok, Thailand

 **SiPA**
สำนักงานส่งเสริมการค้าในต่างประเทศ (ส.พ.อ.) (Software Policy Promotion Agency) (ส.พ.อ.)

 **SCG**
SIAM CEMENT GROUP

NECTEC
a member of NSTDA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Affine Invariant Shape Recognition Based on Neural Networks

Thoum Kosorl, Kritsana Yawichai and Yuttana Kitjaidure

Department of Electronics

Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang

3 Moo. 2 Chalongkrung Rd., Ladkrabang Bangkok, 10520 Thailand

E-mail: kosal_th@yahoo.com astronaut_aor@hotmail.com and kkyuttan@kmit.ac.th

Abstract. Several shape recognition systems based on pairwise shape matching technique have achieved high accuracy but they face a time consuming problem when evaluated on a large database. So this drawback makes the system impractical for real-time applications. Motivated by this obstacle, we have investigated a novel and robust neural network solution to achieve high speed of shape recognition without sacrificing accuracy via the non-absolute 1-D triangle area representation (NATA). Our method has been evaluated over a number of affine distorted shapes. The experimental results demonstrate that a shape recognition system using off-line training neural networks can achieve high speed comparable with the prior system.

I. INTRODUCTION

The ultimate goal of computer vision field is to make machine able to identify and distinguish shapes observed at an arbitrary viewpoint as capable as human do. Shape descriptor is a powerful and versatile tool for recognition of machines. The appropriate descriptor of shape feature should not change under perspective distortion which typically arises if a planar object is observed under arbitrary viewpoints with respect to the plane. However, the affine transformation may be used as an approximation to the perspective transformation rather than the use of similarity transformation [1]. Shape description techniques can be divided into contour-based and region-based. In [2], the contour-based representation uses less data points and is usually less sensitive to noise than the region-based techniques. For this reason, we decide to choose the contour-based description technique in our work. Several contour features have been introduced in the literature [3], [4]. Farzin Mokhtarian and Alan K. Mackworth [5] have presented a multiscale, curvature-based shape descriptor technique for classification planar curves. This algorithm is computationally intensive due to the huge number of scales it requires. Haibin Ling and Dawid W. Jacobs [6] have used the inner-distance to build shape descriptors for shape matching through dynamic programming. It can be seen that the previous proposed works have employed complex shape descriptor and matcher to achieve high accuracy. To alleviate this problem, our work presented in [7] has proposed a relatively simple shape descriptor, the absolute 1-D triangle area representation (ATA) to match and retrieve shape using normalized cross-correlation function. However this ATA has low discrimination capability and faces major difficulties in discriminating among those indistinct shape

descriptors. So in [8], we have proposed the non-absolute 1-D triangle area representation (NATA) to deal with this problem by varying triangle side length (TSL) to find the optimum TSL that gives a good discriminatory power. It is found that similarity measure using normalized cross-correlation function through NATA offers better results than ATA.

From the previous works, [7] and [8], similarity measure using normalized cross-correlation function is the pairwise matching between a query shape and database shapes. However, several real-life systems usually have a large database. If the normalized cross-correlation function is used in real-life system, it would take the identification procedure much time consumption. Our aim in this research is to increase the speed of shape recognition without sacrificing accuracy via NATA (used in [8]) by using neural networks. This technique is simple, computationally cost effective and can be easily implemented. Our method has been evaluated over a number of affine distorted shapes captured from arbitrary viewpoints. We conduct the experiment to compare with the previous work in [8]. The algorithm presented here is significantly faster than the previous work.

In this paper, the approach and method with detailed description to all processes are given in section 2. In section 3, the experimental results are obtained. Finally, section 4 gives conclusions and future work.

II. APPROACH AND METHOD

A. Preprocessing

First, the shape is segmented from the image. Then, contour following technique [9] is applied to the shape to obtain the contour as a sequence of points $(x(u), y(u))$, which is the coordinates of each point on the contour. To avoid effect of noise caused by digitization noise and segmentation errors, the contour sequences $(x(u), y(u))$ are smoothed by a Gaussian filter with σ set to 1.5% of the total number of coordinate points on the considered contour, N . Finally, all contours (query and database contours) are sampled at the same number of points, n . The sampled points will be denoted by k_1, k_2, \dots, k_n , where $k_i = (x_i, y_i)$. The number of sampled points should be large enough to capture significant structure of the interest shape. The appropriate number of sampled points, n found experimentally for this work is 100, which is sufficient for all examples we have tested. The preprocessing steps are shown in Fig. 1.

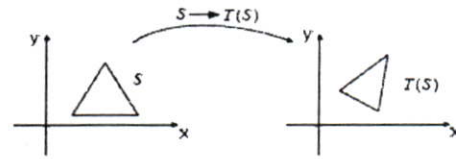
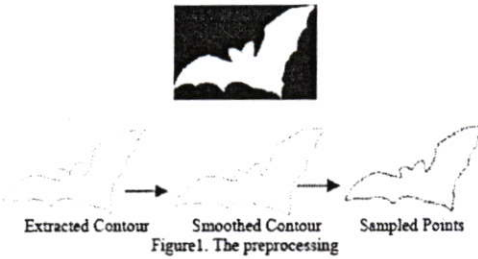


Figure 2. A triangle (S) is transformed to T(S)

$$\{\text{area of } T(S)\} = \left| \det \begin{bmatrix} a & b \\ c & d \end{bmatrix} \right| \cdot \{\text{area of } S\} \quad (3)$$

This area relation is used to derive the affine invariant property of NATA.

Considering the sampled points of a contour, k_1, k_2, \dots, k_n obtained from the previous section. Let any three sampled points, k_{p-1}, k_p, k_{p+l} be the vertices of a triangle as shown in Fig. 3. The triangle area can be calculated by:

$$A_i(p) = \frac{1}{2} \begin{vmatrix} x_{k_{p-1}} & y_{k_{p-1}} & 1 \\ x_{k_p} & y_{k_p} & 1 \\ x_{k_{p+l}} & y_{k_{p+l}} & 1 \end{vmatrix} \quad (4)$$

where p is the sampled points, $p=1,2,\dots,100$. l is the triangle side length (TSL) considering l sampled points further from p . Then, we obtain the non-absolute 1-D triangle area representation at various triangle side length NATA(l). These representations are plotted in Fig. 4c-d (the vertical and horizontal axis represent the normalized area's values and the sampled points on the contour respectively). It is noticed that the characteristic of non-absolute 1-D triangle area representations plotted from the same shapes captured by a camera in different viewpoints (Fig. 4a-b) are similar.

C. Recognition Stage

In this section, we explain the recognition methods including normalized cross-correlation and neural networks, which are used in the previous works and the current work respectively



Figure 3. Three sampled points on the contour (at p^{th} sampled point) are used to calculate the area at any TSL.

B. Affine Invariant Feature Extraction

In this subsection, shape descriptors are extracted in order to prepare for feeding into neural networks. A shape descriptor is a set of numeric values that describes the shape in a way that makes it distinguishable from other shapes. Below are some criteria for shape descriptor for reliable shape recognition.

- *Uniqueness*: A descriptor must uniquely specify a shape.
- *Robustness*: A descriptor must be resistant to moderate amounts of distortion and noise. Because it is not possible to guarantee that a shape descriptor is not affected by such factors.
- *Invariance*: A descriptor must be invariant to viewing condition, that is, it must be invariant to translation, scale, rotation, viewing angle changes.

This work, we use the shape descriptor, NATA which is invariant to affine transform. The affine transform includes translation, scaling, rotation, and shear. The invariant property of NATA is derived from the theory of determinant in [10]. Suppose a triangle (S) is transformed to be a distorted triangle ($T(S)$) by affine transform as shown in Fig 2. The affine transform can be mathematically described in matrix explained in [11] as follows.

If $T: R^2 \rightarrow R^2$ is the matrix operator whose standard matrix is

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad (1)$$

then

$$T \left(\begin{bmatrix} x \\ y \end{bmatrix} \right) = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} \quad (2)$$

where $T \begin{bmatrix} x \\ y \end{bmatrix}$ is the affine-transformed version of $\begin{bmatrix} x \\ y \end{bmatrix}$.

In our case $T \begin{bmatrix} x \\ y \end{bmatrix}$ or $\begin{bmatrix} x \\ y \end{bmatrix}$ is the coordinate vector of points on the contour. a, b, c and d are the coefficients of scaling, rotation, and shear. e and f are the coefficients of translation.

From the determinant theory, the area of S and $T(S)$ are related by (3). (From (2), translating a triangle (S) by $\begin{bmatrix} e \\ f \end{bmatrix}$ does not affect its area.)

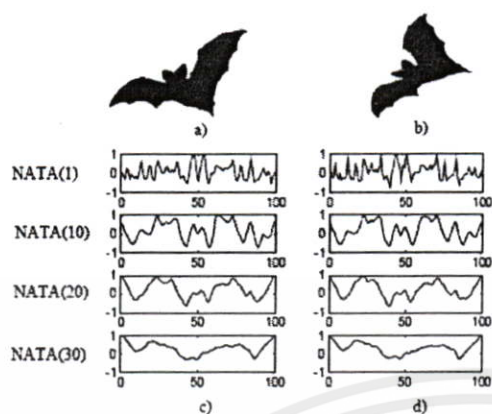


Figure 4. (a)-(b) the same shapes captured in different viewpoints. (c)-(d) the non-absolute 1-D triangle area representations of (a) and (b) respectively.

1. Similarity Measure Using the Normalized Cross-correlation

To match a query shape to the database, the normalized cross-correlation values of the NATA between the query shape (denoted by $Q = [q_1, q_2, \dots, q_n]$) and each of the database shapes (denoted by $D = [d_1, d_2, \dots, d_n]$) are computed.

$$R(Q, D) = \frac{\sum_{i=1}^n q_i d_i}{\sqrt{\sum_{i=1}^n q_i^2 \sum_{i=1}^n d_i^2}} \quad (5)$$

Then, the query shape will be assigned to one of the database shapes corresponding to the maximum values of the normalized cross correlation.

2. Similarity Measure Using the Neural Network

This work, we have chosen neural networks because of their favorable properties that make them an excellent choice for shape recognition. The most important of these properties are:

- **Generalization:** Small distortions can be handled easily, a necessity for shape recognition.
- **Representing multiple samples:** A class of shapes can easily be represented by multiple samples under multiple conditions. Because a neural network incorporates in its structure what it learns, the recognition of a shape becomes a single step. In this way there is no need for multiple comparisons as in pairwise matching method. The network determines in single step to what class the shape belongs.
- **Memory saving:** An advantage of the neural

network from the previously mentioned characteristic is that there is no need to store all the standard shapes to be used for comparison. Once the network is trained properly, it contains the necessary information and the data of shapes become expendable and can be removed from memory.

To consider discrimination capability of NATA at different TSLs, we create neural networks [12] for each feature (NATA(i)). Then, every shape descriptor and its own primary shape class are used as an input and an output of the neural networks respectively. The network attempts to implement mapping between the input pattern $I = [I_1, I_2, \dots, I_n]^T$ and the desired output pattern $O = [O_1, O_2, \dots, O_c]^T$. The feed-forward back-propagation neural network with two hidden layers is used as shown in Fig. 5, with 100 input neurons, 70 neurons in each hidden layer, and 40 neurons in the output layer. Tan-sigmoid activation functions are used to all layers except the non-active input layer.

In the training phase, the shape descriptors $I^b(c)$, $b=1, \dots, M$ of M curve prototypes (c denotes the c^{th} class in the database) are used to train the supervised neural network. While, the desired output $O(c)$ is the shape prototype class. The weight ω_j of the j^{th} neuron of each layer is updated in the k -th training cycle according to the following equation.

$$\Delta \omega_j(k+1) = \gamma \Delta \omega_j(k) + (1-\gamma) \alpha p_{k+1} \quad (6)$$

The parameters α and γ are the learning rate and the momentum factor respectively. The Scaled-Conjugated Gradient method is used for training.

$$r_{k+1} = -g_{k+1} \quad (7)$$

$$\beta_k = \frac{|r_{k+1}|^2 - r_{k+1} r_k}{p_k^T r_k} \quad (8)$$

$$p_{k+1} = r_{k+1} + \beta_k p_k \quad (9)$$

where p and g are the search direction and gradient respectively.

In the testing phase, the shape descriptor of the queries is used as an input to the neural network. The value of the network output is the similarity values. The recognition results are obtained from the input classified to the shape class corresponding to the maximum network output.

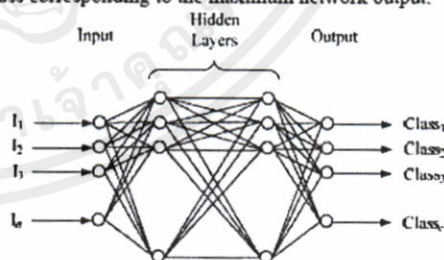


Figure 5. Neural network architecture used for shape recognition.

III. EXPERIMENTAL RESULTS AND DISCUSSION

The MPEG-7 dataset[13] has been used in many researches. Unfortunately, this dataset does not contain shapes in the presence of affine distortion, thus it is not suitable for testing the performance of our affine invariant algorithm. So, we construct the multi-view shape dataset which consists of 40 classes of shapes chosen from the MPEG-7 dataset. Each class has 14 different affine distorted shapes. Fig. 6 shows the samples from the class "bat". This multi-view shape dataset has a total of 560 shapes. 1 view per class is actually used for training and the set of 13 unknown various views are used for testing. Our system is implemented in MATLAB version 7.3 running on a computer AMD Athlon 64x2 Dual core 2 GHz and RAM 1 GB.

A. Recognition Accuracy

To demonstrate the benefit of using the neural network (NN), we compare the recognition results with the pairwise matching utilizing normalized cross-correlation (NCC). They are evaluated on the same shape dataset (the multi-view shape dataset). The answered shape corresponding to the maximum network output or the maximum value of the normalized cross-correlation, which is identified to the same class as query, is counted. The comparison results for the evaluation on NATA(l) are shown in Table 1 (only the explicitly results of some TSL are depicted to see the trend of accuracy).

From the results, it is observed that the recognition performance of NATAs are decreased when TSL is small. At the small TSL, we obtain only local features of shape, which



Figure 6. The class "bat".

TABLE I
The percent accuracy for recognition of the neural network and the normalized cross-correlation utilizing NATA at various TSL.

NATA(l)	Accuracy of testing set %	
	NN	NCC
1	58.65	76.92
3	66.54	84.42
5	76.54	87.12
10	89.23	92.69
20	91.15	93.65
30	88.65	88.65
40	85.77	87.69
49	78.46	83.27

are quite sensitive to boundary noises and trivial details. In the same way when TSL is too large, the recognition performance are also declined due to the fact that only rough structure of the shape is obtained. We have found that the satisfactory results is obtained from the triangle side length at 20 since it can detect optimal global feature of the shape, which makes our algorithm insensitive to noise and more stable to handle distorted shape variation. As shown in Table 1, recognition performance of the normalized cross-correlation is slightly better than the neural network. However, for the speed comparison, we have considered the computation time per 1 query of both methods and found that the neural network takes only 83.93 μ s, which is less than the normalized cross-correlation (564 μ s). This can be concluded that the neural network increases the speed of shape recognition without sacrificing accuracy and is more suitable to use with a large dataset.

B. Real world applications

Our method can be used to apply with many applications, for instance, applying our method to find similar shapes between images taken from different viewpoints which can adapt for image registration. Fig. 7 shows the best 14 matches and 2 mismatches between the set of shapes (each interested shape is surrounded by yellow border) extracted from the pair of "Egypt" images, each of which is captured from different viewpoints. Notice that the mismatched shapes are very similar.

IV. CONCLUSION AND FUTURE WORK

This paper has proposed the neural network for shape recognition in the presence of affine distortion utilizing the 1-D triangle area representation. The experiments on the proposed method are performed with the multi-view shape database. The experimental results have illustrated that our approach achieves high speed and high accuracy even under a moderate rage of visual transformation. Moreover, we have demonstrated the applications of our method in finding similar shapes between the images taken from various viewpoints, which leads to image registration technique. Research on this promising approach to shape recognition is on going, focusing on improving an algorithm further enhancing its performance.

REFERENCES

- [1] D. W. Thompson and J. L. Mundy, "Three dimensional model matching from an constrained viewpoint." in *Proc. Conf. Robotics and Automation, Raleigh, NC*, pp. 208-220, 1987.
- [2] P. Otterloo. "A Contour-Oriented Approach to Shape Analysis." Prentice Hall Int., 1991. pp. 5-8.
- [3] S. Loncaric, "A survey of shape analysis techniques." *Patt. Recog.*, vol. 31, no. 8, pp. 983-1001, 1998.
- [4] D. Zhang and G. Lu, "Review of shape representation and description techniques," *Patt. Recog.*, vol. 37, no. 1, pp. 1-19, 2004.
- [5] Farzin Mokhtarian and Alan K. Mackworth, "A Theory of Multiscale, Curvature-Based Shape Representation for Planar

- Curves." *IEEE Trans. Pattern Analysis and Machine Intelligent*, vol. 14, no. 8, pp. 789-805, Aug., 1992.
- [6] Haibin Ling and Dawid W. Jacobs, "Shape Classification Using the Inner-Distance." *IEEE Trans. Pattern Analysis and Machine Intelligent*, vol. 29, no. 2, pp. 286-299, Feb.2007.
- [7] Kritsana Yawichai and Yuttana Kitjaidure, "A Simple and Efficient Algorithm for Affine Invariant Shape Matching." *International Conference on Engineering, Applied sciences, and Technology*, Bangkok, Thailand, pp. 19, Nov 2007.
- [8] Kritsana Yawichai and Yuttana Kitjaidure, "Multi-view Shape Matching and Retrieval." *International Conference on Robotics, Vision, Information, and Signal Processing*, Penang, Malaysia, Nov 2007.
- [9] W.K.Pratt., "Digital Image Processing". John Wiley and sons, Inc, third edition, 2001, pp. 580-583.
- [10] H. Anton and C. Rorres, "Elementary Linear Algebra," Wiley and Sons, Inc, ninth edition, 2005.
- [11] David C. Lay, "Linear Algebra and Its Applications". Pearson Education, Inc, third edition, 2006, pp. 205-207.
- [12] Martin T. Hagan and Howard B. Demuth, "Neural Network Design," International Thomson Publishing Company, 1995.
- [13] MPEG-7 dataset. Citing Internet sources URL <http://www.cis.temple.edu/~latecki/>.

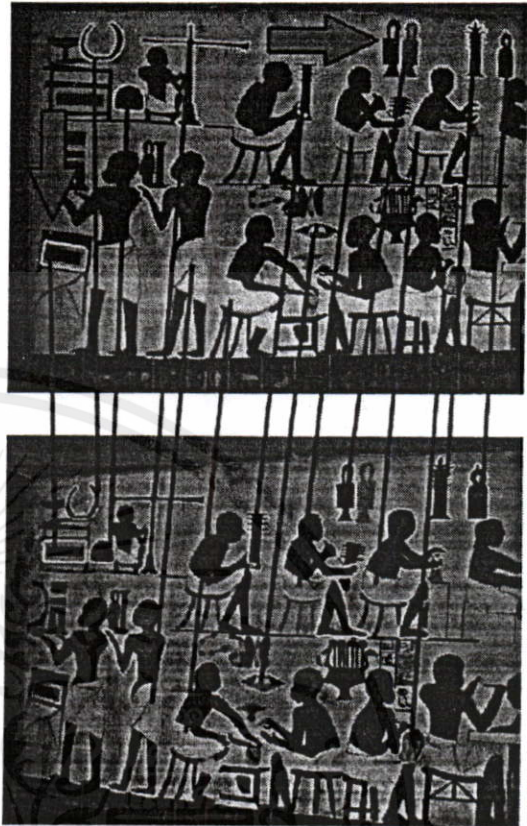


Figure 7. Views of the "Egypt" scenes. The lines connect the corresponding shapes and the arrows indicate the shape mismatch.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ECTI-CON 2008
THE 2008 ECTI INTERNATIONAL CONFERENCE

Volume 1

Proceedings of the 2008 Electrical Engineering/ Electronics, Computer,
Telecommunications and Information Technology (ECTI)
International Conference

May 14-17, 2008
Maritime Park and Spa Resort, Krabi, THAILAND

ECTI Association
TRIDI
Telecommunications Research and Industrial Development Institute
a member of NSTDA
NECTEC
Seagate
TNGC
THAILAND SECTION
IEEE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Optimum Shape Representation Based on Fisher's Discriminant Analysis

Thourn Kosorl, Kritsana Yawichai and Yuttana Kitjaidure
Department of Electronics

Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang
3 Moo. 2 Chalokkrung Rd., Ladkrabang Bangkok, 10520 Thailand

E-mail: kosal_th@yahoo.com, astronaut_aor@hotmail.com and kkyuttan@kmitl.ac.th

Abstract—Shape recognition is an important part of machine intelligence in both decision making and data processing. A good shape representation in shape recognition should describe the shape in the way that makes it distinguishable from other shapes and be invariant to transform of position, size, angle and skew. More importantly, developing and finding appropriate shape representation are still a challenging problem. In our previous works, the 1-D area representation at various triangle side lengths has been proposed as an affine invariant shape representation. Finding the optimum triangle side length, the best 1-D area representation, needs to conduct through recognition systems experimentally. In this work, Fisher's discriminant analysis is applied to predict the optimum triangle side length instead of obtaining it from the experiment. This method has been evaluated over a number of affine distorted shapes. The predicted optimum triangle side length is compared with the optimum results obtained from simulation through the recognition systems (neural networks and normalized cross-correlation). The results demonstrate that the performance of recognition depends on the discriminant power of shape representation. The higher Fisher's ratio gives the better recognition performance. The best performance is achieved by using shape representation that has the maximum Fisher's ratio.

I. INTRODUCTION

Human makes an object recognition task trivial. We can easily identify objects no matter what kind of the circumstance. Even objects appearing in many different forms can easily be generalized by our brain to one kind of objects. Many different algorithms have been created to develop robot or machine to perform object recognition as well as human do. Most of them have employed a shape-based approach to recognize objects. The shape representations are extracted in order to prepare them for the recognition process. Shape representation techniques are mainly divided into contour-based and region-based [1].

The appropriate descriptor of shape representation should not change under perspective distortion which typically arises if a planar object is observed under arbitrary viewpoints with respect to the plane. However, the affine transformation may be used as approximation to the perspective transformation as long as the imaged object is planar and the camera optical center is far enough from such plane [2], [3]. Farzin Mokhtarian and Alan K. Mackworth [4] have presented a multiscale, curvature-based shape descriptor technique for

classification planar curves. This algorithm is computationally intensive due to the huge number of scales it requires. Haibin Ling and Dawid W. Jacobs [5] have used the inner-distance to build shape descriptors for shape matching through dynamic programming. It can be seen that the previous proposed works have employed complex shape descriptor. To alleviate this problem, our research has attempted to develop shape representation for recognition systems. In our previous work, we have applied the determinant theory to calculate the absolute 1-D triangle area representation (ATA) [6], [7], which is not only simple to extract but also invariant to affine transformation. Furthermore, we have developed the non-absolute 1-D triangle area representation (NATA) by varying the triangle side length (TSL) [8] to find the appropriate side length that can be used efficiently for shape recognition. However, the optimum TSL is determined only from the experiment via recognition systems. From this, we have observed that a good shape representation could obviously distinguish among the different shapes. In addition, the performance of shape recognition depends on the discriminative ability of shape representation. This inspires us an idea that if we could find any tool that can be measured the discriminative property of shape representation, so we can predict the optimum shape representation without considering through the whole recognition process as it has been done in our previous works. By this way, the large step for judgment of the best 1-D area representation is greatly reduced. Moreover, finding the optimum shape representation based on the discriminative power can form a good basis for further development to increase the certainty of decision making of robot and other applications.

We have found that the theory of Fisher's discriminant can be applied for the discriminative measure. The concept of Fisher's discriminant can be described by considering the ratio of the between-class scatter and the within-class scatter. The optimal 1-D area representation can be obtained from the maximum of this ratio. Thus, we apply Fisher's discriminant analysis to predict the optimum affine invariant shape recognition and prove it via neural networks and normalized cross-correlation. In this work, the quality of discriminant based on the Fisher's method is measured on the NATA sets in order to find the optimum triangle side length that gives the highest discriminative power. The experimental results

indicate that the maximum Fisher's ratio can be used to make accurate prediction of the optimum feature of shape representation.

II. AFFINE INVARIANT SHAPE REPRESENTATION

A shape representation is a set of numeric values that describes the shape in a way that makes it distinguishable from other shapes. Some criterias of shape representation for reliable shape recognition are shown below.

- *Uniqueness*: A descriptor must uniquely specify a shape.
- *Robustness*: A descriptor must be resistant to moderate amounts of distortion and noise. Because it is not possible to guarantee that a shape descriptor is not affected by such factors.
- *Invariance*: A descriptor must be invariant to viewing condition, that is, it must be invariant to translation, scale, rotation, viewing angle changes.

In this work, we use NATA as a shape descriptor, which is invariant to affine transform. The invariant property of NATA is derived from the theory of determinant in [9]. Suppose a triangle (S) is transformed to be a distorted triangle (T(S)) by affine transform as shown in Fig. 1. The affine transform can be mathematically described in matrix as follows.

$$T \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} \quad (1)$$

where $T \begin{bmatrix} x \\ y \end{bmatrix}$ is the affine-transformed version of $\begin{bmatrix} x \\ y \end{bmatrix}$.

In this case $T \begin{bmatrix} x \\ y \end{bmatrix}$ or $\begin{bmatrix} x \\ y \end{bmatrix}$ is the coordinate vector of points on the contour. a, b, c and d are the coefficients of scaling, rotation, and shear. e and f are the coefficients of translation.

From the determinant theory, the area of S and T(S) are related by (2). (From (1), translating a triangle (S) by $\begin{bmatrix} e \\ f \end{bmatrix}$ does not affect its area.)

$$\{\text{area of } T(S)\} = \left| \det \begin{bmatrix} a & b \\ c & d \end{bmatrix} \right| \cdot \{\text{area of } S\} \quad (2)$$

This relation is used to derive the affine invariant property of NATA. NATA is the contour-based shape representation which is calculated from 100 sampled points of the shape's contour. These sampled points are represented in coordinate (x,y). Let any three sampled points, $p-l, p, p+l$ be the vertices of a triangle as shown in Fig. 2. The triangle area can be calculated by:

$$A(p) = \frac{1}{2} \begin{vmatrix} x_{p-l} & y_{p-l} & 1 \\ x_p & y_p & 1 \\ x_{p+l} & y_{p+l} & 1 \end{vmatrix} \quad (3)$$

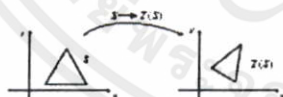


Figure 1. A triangle (S) is transformed to T(S).



Figure 2. Three sampled points on the contour (at p^{th} sampled point) are used to calculate the area at any TSL.

where p is the index of the sampled points, $p=1, 2, \dots, 100$. l is the triangle side length (TSL) considering l sampled points further from p . Then, we obtain the non-absolute 1-D triangle area representation at various triangle side lengths, NATA(l). These representations are exemplified in Fig. 3c-d (the vertical and horizontal axis represent the normalized values of 1-D area representation and the sampled points on the contour respectively). It is noticed that the characteristic of non-absolute 1-D triangle area representations plotted from the same shapes captured by a camera in different viewpoints (Fig. 3a-b) are similar.

III. FISHER'S DISCRIMINANT

It is important that shapes we consider to be alike have the same representations while at the same time, they should also make it possible to adequately distinguish among different shapes. The main motivation for this research lies on the discriminative property analyzing of NATA at each TSL based on Fisher's discriminant.

Fisher's Linear Discriminant (FLD) [10] is a widely used method for feature extraction and dimensionality reduction in pattern recognition. The Fisher's idea is based on seeking projection direction W that separates the class means well (onto the projected direction W) while achieving a small variance around these means. Mathematically, FLD selects the projection W in such a way that the ratio of the between-class scatter and the within-class scatter (as in (4)) is maximized. From this concept, the ratios are computed from the data set of NATA at each TSL without projecting onto the new basis.

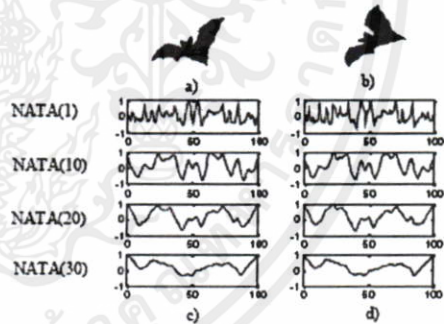


Figure 3. (a)-(b) the same shapes captured in different viewpoints. (c)-(d) the non-absolute 1-D triangle area representations of (a) and (b) respectively.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\text{Fisher's ratio} = \frac{S_B}{S_W} = \frac{\sum_i^c (\mu_i - \mu)^2}{\sum_i^c \sum_{x_i \in X_i} (x_i - \mu_i)^2} \quad (4)$$

where c is the number of class, μ_i is the sample mean of class X_i , μ is the sample mean of all classes.

The interpretation of this ratio can be explained by considering S_B and S_W in Fig. 4. The maximum ratio occurs when S_B is large comparing with S_W . The larger S_B indicates the information between the class is so much different hence we can distinguish each class easily. The smaller S_W indicates the information within the class has small variance as the criteria of a good shape representation should have.

IV. RECOGNITION STAGE

In order to compare with the Fisher's concept described above, the recognition systems including normalized cross-correlation and neural networks are simulated.

A. Normalized Cross-correlation

To match the query shape to the database, the normalized cross-correlation (NCC) values of the triangle area vector between the query shape (denoted by $Q = [q_1, q_2, \dots, q_n]$) and each of the database shapes (denoted by $D = [d_1, d_2, \dots, d_n]$) are computed.

$$R(Q, D) = \frac{\sum_{i=1}^n q_i d_i}{\sqrt{\sum_{i=1}^n q_i^2 \sum_{i=1}^n d_i^2}} \quad (5)$$

Then, the query shape will be assigned to one of the database shapes corresponding to the maximum values of the normalized cross correlation.

B. Neural Networks

To consider discrimination capability of NATA among different TSLs, we create the neural networks (NN) [11] for each of NATA(l) feature. Then, every shape descriptor and its own primary shape class are used as an input and an output of the neural networks respectively. The network attempts to implement mapping between the input patterns $I = [I_1, I_2, \dots, I_n]^T$ and the desired output patterns $O = [O_1, O_2, \dots, O_n]^T$. The feed-

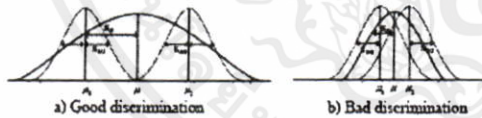


Figure 4. The Fisher's discriminant.

forward back-propagation neural network with two hidden layers is used as shown in Fig. 5, with 100 input neurons, 70 neurons in each hidden layer, and 40 neurons in the output layer. Tan-sigmoid activation functions are used to all layers except the non-active input layer.

In the training phase, the shape descriptors $I^b(c)$, $b = 1, \dots, M$ of M curve prototypes (c denotes the c^{th} class in the database) are used to train the supervised neural network. While, the desired output $O(c)$ is the shape prototype class. The random weight initialize method is used. The weight ω_j of the j^{th} neuron of each layer is updated in the k -th training cycle according to the following equation.

$$\Delta \omega_j(k+1) = \gamma \Delta \omega_j(k) + (1 - \gamma) \alpha p_{k+1} \quad (6)$$

The parameters α and γ are the learning rate and the momentum factor respectively. The Scaled-Conjugated Gradient method is used for training.

$$r_{k+1} = -g_{k+1} \quad (7)$$

$$\beta_k = \frac{|r_{k+1}| - r_{k+1} r_k}{p_k^T r_k} \quad (8)$$

$$p_{k+1} = r_{k+1} + \beta_k p_k \quad (9)$$

where p and g are the search direction and gradient respectively.

In the testing phase, the shape descriptor of the queries is used as an input to the neural network. The value of the network output is the similarity measure. The recognition results are obtained from the input classified to the shape class corresponding to the maximum network output.

V. EXPERIMENTAL RESULTS AND DISCUSSION

The MPEG-7 dataset [12] has been used in many researches. Unfortunately, this dataset does not contain shapes in the presence of affine distortion, thus it is not suitable for testing the performance of our affine invariant algorithm. So, we construct the multi-view shape dataset which consists of 40 classes of shapes chosen from the MPEG-7 dataset. Each class has 14 different affine distorted shapes. Fig.6 shows the samples from the class "bat". This multi-view shape dataset has a total of 560 shapes. One view per class is actually used for training and the set of 13 unknown various views are used for testing. Our system is implemented in MATLAB version 7.3 running on a computer AMD Athlon 64x2 Dual core 2 GHz and RAM 1 GB.

We measure the Fisher's ratio comparing with the accuracy of NATA feature obtained from the recognition systems at each TSL. The results are shown in Fig. 7, Fig. 8 and Table I (only the explicitly results of some TSL are depicted to see the trend of Fisher's ratio and the system's accuracy).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

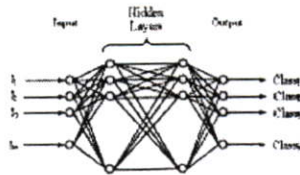


Figure 5. The neural network architecture used for shape recognition.



Figure 6. The class "bat".

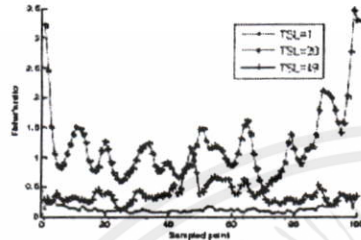
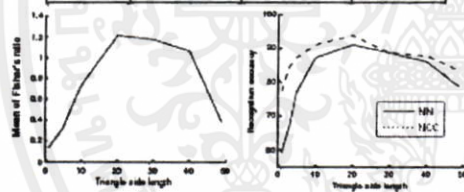


Figure 7. The Fisher's ratio of NATA at some TSLs.

TABLE I

The means of Fisher's ratio are relatively compared with the percent accuracy of shape recognition obtained from the neural network and the normalized cross-correlation utilizing NATA at various TSLs.

NATA(<i>l</i>)	Mean of Fisher's ratio	The simulation performance(%accuracy)	
		NN	NCC
1	0.129	58.65	76.92
3	0.2146	66.54	84.42
5	0.31	76.54	87.12
10	0.7137	89.23	92.69
20	1.2103	91.15	93.65
30	1.1697	88.65	88.65
40	1.0585	85.77	87.69
49	0.3812	78.46	83.27



a) means of the Fisher's ratio b) the recognition system
Figure 8. Trend of the Fisher's mean and the recognition accuracy.

From Fig. 7 and Table I, it is observed that when TSL is too small or too large the Fisher's ratio is decline. This indicates that the NATAs of both cases can not distinguish between different shapes. At the small TSL, we obtain only local features of shape, which are quite sensitive to boundary noises and trivial details. In the same way when TSL is too large the only rough structure of the shape is obtained. We have found

that the highest Fisher's discriminant value is obtained from the triangle side length at around 20 since it can detect optimal global features of the shape, which makes NATA insensitive to noise and more stable to handle distorted shape variation. From Fig.8, it can be seen that the trend of Fisher's discriminant means corresponds with the trend of recognition accuracy of both methods (the neural network and normalized cross-correlation). So, we can conclude that Fisher's discriminant values can be used to predict the optimum 1-D area representations.

VI. CONCLUSION AND FUTURE WORK

This paper has proposed the Fisher's discriminant analysis for finding the optimum feature for shape representation. The proposed method are performed with affine invariant shape representation, NATA, at various TSLs. The results show that the optimum triangle side length has the highest Fisher's ratio, which gives the highest recognition accuracy. It can be concluded that Fisher's discriminant analysis can be used to find the optimum shape representation of our works. However, there are some other parameters left to be observed in the future work, for example, the variance of the ratios. In addition, the Fisher's value of each sample point can inform the significant of point locations. With this advantage, we can apply to the estimation of robot or computer vision for further performance enhancing.

REFERENCES

- [1] D. Zhang and G. Lu, "Review of shape representation and description techniques," *Pat. Recog.*, vol. 37, no. 1, pp. 1-19, 2004.
- [2] D. W. Thompson and J. L. Mundy, "Three dimensional model matching from an constrained viewpoint," in *Proc. Conf. Robotics and Automation*, Raleigh, NC, pp. 208-220, 1987.
- [3] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000.
- [4] Farzin Mokhtarian and Alan K. Mackworth, "A Theory of Multiscale, Curvature-Based Shape Representation for Planar Curves," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 8, pp. 789-805, Aug., 1992.
- [5] Haibin Ling and David W. Jacobs, "Shape Classification Using the Inner-Distance," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, no. 2, pp. 286-299, Feb 2007.
- [6] Kritsana Yawichai and Yuttana Kijaidura, "A Simple and Efficient Algorithm for Affine Invariant Shape Matching," *International Conference on Engineering, Applied sciences, and Technology*, Bangkok, Thailand, pp. 19, Nov 2007.
- [7] Kritsana Yawichai and Yuttana Kijaidura, "Fast Classification of Multi-view Shape using Neural network," *1st Joint International Conference on Information and Communication Technology*, Vientiane, Lao PDR, Dec 2007.
- [8] Kritsana Yawichai and Yuttana Kijaidura, "Multi-view Shape Matching and Retrieval," *International Conference on Robotics, Vision, Information, and Signal Processing*, Penang, Malaysia, Nov 2007.
- [9] H. Anton and C. Roosre, "Elementary Linear Algebra," Wiley and Sons, Inc, ninth edition, 2005.
- [10] Peter N. Belhumeur, Joao P. Hespanha, and David J. Kriegman, "Eigenface vs. Fisherface: Recognition Using Class Specific Linear Projection", *IEEE Trans on Pattern Analysis and machine Intelligence*, Vol. 19, No. 7, July 1997.
- [11] Martin T. Hagan and Howard B. Demuth, "Neural Network Design," International Thomson Publishing Company, 1995.
- [12] MPEG-7 dataset. Citing Internet sources URL <http://www.cis.temple.edu/~latecki/>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อ นางสาวกฤษณา ขาววิไชย
 เกิดวันที่ 12 พฤศจิกายน 2526
 การศึกษา ปีการศึกษา 2539-2544
 ระดับมัธยมศึกษา โรงเรียนท่าวังพาพิทยาคม

ปีการศึกษา 2545-2548
 ระดับปริญญาตรีวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมอิเล็กทรอนิกส์
 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้