สำนักหอสมุดกลาง พระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ระบบการรู้จำตัวอักษรภาษาไทยโดยใช้โครงข่ายประสาทแบบปรับค่าย้อนกลับ

Thai Online Handwritten Character Recognition Using Backpropagation Neural Networks

T055734

โดย

นาย ภูมิพัฒน์ ลิขิตธรรมนิตย์

นาย ยุทธพงษ์ รัตนวงษ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เลขหมู่.....
เลขทะเบียน 55734
วัน,เดือน,ปี 25 พ.ค. 2548

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบ.....
เล่ม.....

Thai Online Handwritten Character Recognition Using Backpropagation Neural Networks

BY

MR. BHUMIPAT LIKITTHAMMANIT

MR. YUTTAPONG RATTANAVONG

A THESIS SUBMITED IN PARTIAL FULFILLMENT OF

THE REQUIREMENT FOR THE DEGREE OF

BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2003

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	**ระบบการรู้จำตัวอักษรภาษาไทยโดยใช้โครงข่ายประสาทแบบปรับค่าย้อนกลับ**
ชื่อนักศึกษา	**นาย ภูมิพัฒน์ ลิขิตธรรมนิตย์ รหัสประจำตัว 43010328** **นาย ยุทธพงษ์ รัตนวงษ์ รหัสประจำตัว 43010346**
อาจารย์ที่ปรึกษา	**ดร. สมเกียรติ อุดมหรรษากุล**
ระดับการศึกษา	**ปริญญาตรี วิศวกรรมศาสตร์บัณฑิต** **สาขาวิศวกรรมสารสนเทศ**
ภาควิชา	**วิศวกรรมสารสนเทศ**
ปีการศึกษา	**2546**

ปริญญานิพนธ์ฉบับนี้ได้รับความเห็นชอบจากอาจารย์ที่ปรึกษาเป็นที่เรียบร้อยแล้ว

.....

(ดร. สมเกียรติ อุดมหรรษากุล)

อาจารย์ผู้ควบคุมปริญญานิพนธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	**ระบบการรู้จำตัวอักษรภาษาไทยโดยใช้โครงข่ายประสาทแบบปรับค่าย้อนกลับ**
ชื่อนักศึกษา	**นาย ภูมิพัฒน์ ลิขิตธรรมนิตย์ รหัสประจำตัว 43010328**
	นาย ยุทธพงษ์ รัตนวงษ์ รหัสประจำตัว 43010346
อาจารย์ที่ปรึกษา	**ดร. สมเกียรติ อุดมหรรษากุล**
ระดับการศึกษา	**ปริญญาตรี วิศวกรรมศาสตร์บัณฑิต**
	สาขาวิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2546

บทคัดย่อ

ปริญญานิพนธ์ชิ้นนี้ได้จัดทำขึ้นเพื่อเสนอวิธีการรู้จำตัวอักษรภาษาไทย เนื่องจากตัวอักษรภาษาไทยมีลักษณะที่ค่อนข้างซับซ้อนและมีความใกล้เคียงกันมากระหว่างตัวอักษรบางกลุ่ม วิธีการรู้จำตัวอักษรภาษาไทยที่ถูกเลือกเพื่อนำมาประยุกต์ใช้กับโครงงานชิ้นนี้คือ โครงข่ายประสาทแบบหลายชั้น ซึ่งใช้วิธีการเรียนรู้แบบปรับค่าย้อนกลับ เนื่องจากวิธีการดังกล่าวสามารถแก้ปัญหาการรู้จำในรูปแบบต่าง ๆ ได้อย่างมีประสิทธิภาพและมีการประมวลผลที่รวดเร็ว อย่างไรก็ตามวิธีการนี้มีข้อจำกัดบางอย่างในการนำมาใช้กับการรู้จำตัวอักษร โครงงานนี้จึงได้นำเทคนิกวินโดวิงมาปรับใช้เพื่อแก้ไขปัญหาดังกล่าว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Thai Online Handwritten Character Recognition Using Backpropagation Neural Networks
Student	MR. Bhumipat Likitthammanit ID. 43010328
	MR. Yuttapong Rattanavong ID. 43010346
Advisor	Dr. Somkait Udomhunsakul
Graduate Level	Bachelor Degree of Information Engineering
Department	Information Engineering
Academic Year	2003

ABSTRACT

This thesis has created to present another method of Thai handwritten character recognition. As Thai characters have a complex structure and similarity between each other. The method of Thai handwritten character recognition that was chosen for this project is a well-known multi-layer perceptron (MLP) neural networks with backpropagation learning algorithm. The reasons of choosing this method are its fast processing time and its effective performance for pattern recognition problems. However this method has some limitation in applying with Thai handwritten character recognition, the windowing technique was applied to use along with MLP neural networks.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

รายงานฉบับนี้จะสำเร็จลงไม่ได้ ถ้าปราศจากความช่วยเหลือจาก ดร. สมเกียรติ อุดมหรรษากุล อาจารย์ที่ปรึกษาที่ได้ให้ความกรุณาช่วยเหลือและให้คำแนะนำในเรื่องต่าง ๆ จนสามารถทำผลงานได้สำเร็จตามเป้าหมาย ขอขอบพระคุณอาจารย์ทุกท่านที่ได้ประสิทธิประสาทวิชาให้ และขอขอบคุณเพื่อน ๆ ทุกคนที่ได้ให้ความช่วยเหลือ สุดท้ายนี้ขอขอบพระคุณบิดาและมารดาผู้ที่เล็งเห็นถึงความสำคัญของการศึกษา และได้ให้การสนับสนุนและส่งเสริมมาโดยตลอด ทางผู้จักทำต้องขอขอบพระคุณทุกท่านเป็นอย่างสูงมา ณ โอกาสนี้ด้วย

นาย ภูมิพัฒน์ ลิขิตธรรมนิตย์

นาย ยุทธพงษ์ รัตนวงษ์

ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้าที่
บทคัดย่อ	ก
Abstract	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูป	ช
สารบัญตาราง	ฌ
บทที่ 1 บทนำ	1
1.1 บทนำ	1
1.2 จุดประสงค์	1
1.3 ขอบเขตของโครงงาน	2
1.4 ผลที่คาดว่าจะได้รับ	2
1.5 ขั้นตอนการดำเนินงาน	2
1.6 ระบบการทำงาน	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	8
2.1 การสร้างจุดให้ต่อเนื่องกัน	8
2.1.1 Midpoint Line Algorithm	8
2.1.2 วิธีการคำนวณ	8
2.1.3 ตัวอย่างผลเมื่อผ่านขั้นตอนนี้	10
2.2 การกำจัดการตวัดของปลายปากกา	10
2.2.1 การทำงาน	11
2.2.2 ตัวอย่างผลเมื่อผ่านขั้นตอนนี้	11
2.3 การกำจัดจุดที่ไม่จำเป็นออก	11
2.3.1 หลักการ	11
2.3.2 การทำงาน	13
2.3.3 ตัวอย่างผลเมื่อผ่านขั้นตอนนี้	14
2.4 การดึงลักษณะเด่น	14
2.4.1 หลักการ	14
2.4.2 ลักษณะเด่นทั้ง 12 แบบ	14
2.4.2.1 ความยาวเฟรม (LengthOfLine)	15
2.4.2.2 ทิศทางของเฟรม (Direction)	15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2.3 ความน่าจะเป็นในการเป็นเฟรมเริ่มต้น (StartLine) 16
2.4.2.4 ความน่าจะเป็นในการเป็นเฟรมสุดท้าย (EndLine) 17
2.4.2.5 มุมเริ่มต้นของเฟรม (AngleStart) 17
2.4.2.6 มุมสุดท้ายของเฟรม (AngleEnd) 20
2.4.2.7 ลักษณะเด่นของตัวอักษร 22
2.4.3 ตัวอย่างผลเมื่อผ่านขั้นตอนนี้ 23
2.5 การใช้โครงข่ายประสาทแบบปรับค่าย้อนกลับในการจดจำตัวอักษร 24
2.5.1 โครงสร้างในเซลล์ประสาทของมนุษย์ 25
2.5.2 ทฤษฎีพื้นฐานของโครงข่ายประสาท 28
2.5.2.1 ลักษณะของโครงข่ายประสาท 28
2.5.2.2 โครงข่ายประสาทในลักษณะ 1 ชั้น (Single Layer Neural Network) 31
2.5.2.3 โครงข่ายประสาทในกรณีที่มีลักษณะหลาย ๆ ชั้น (Multilayer Neural Network) 32
2.5.3 การฝึกสอน (Training) ของโครงข่ายประสาท 33
2.5.3.1 เพอเซปตรอน (Perceptron) 33
2.5.3.2 โครงข่ายประสาทแบบปรับค่าย้อนกลับ 35
2.5.3.2.1 ลักษณะโครงสร้างของโครงข่าย 35
บทที่ 3 โปรแกรมที่เกี่ยวข้อง 41
3.1 โปรแกรมเก็บลักษณะเด่นของตัวอักษรไว้ในไฟล์ 41
3.2 โปรแกรมฝึกสอนระบบ 49
3.3 โปรแกรมแสดงการทำงานของระบบ 51
บทที่ 4 ผลการทดลอง 54
4.1 การทดสอบประสิทธิของโปรแกรม 54
4.1.1 การทดสอบประสิทธิภาพในกรณีตัวอักษรมีลักษณะแตกต่างกัน 54
4.1.1.1 การทดสอบ 54
4.1.1.2 ผลจากการทดสอบระบบ 54
4.1.2 การทดสอบประสิทธิภาพในกรณีตัวอักษรมีลักษณะคล้ายกัน 55
4.1.2.1 การทดสอบ 55
4.1.2.2 ผลจากการทดสอบระบบ 55
4.1.3 โปรแกรมสมบูรณ์ 55
4.1.3.1 การฝึกสอน 55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3.2 ผลจากการทดสอบระบบ 56
บทที่ 5 สรุป 58
5.1 สรุปการพัฒนาโครงงาน 58
5.2 ปัญหาที่เกิดขึ้น 58
5.3 ข้อจำกัดของโครงงาน 58
5.4 แนวทางในการพัฒนาต่อ 58
บรรณานุกรม 59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 1-1 แสดงขั้นตอนและระยะเวลาการดำเนินงานแต่ละขั้น 2
รูปที่ 1-2 แสดงระบบการทำงานช่วงฝึกสอนระบบ 3
รูปที่ 1-3 แสดงระบบการทำงานช่วงนำไปใช้งานจริง 4
รูปที่ 1-4 แสดงลักษณะของลิงค์ลิสท์ 5
รูปที่ 1-5 แสดงผลหลังผ่านขั้นตอนการสร้างจุดให้ต่อเนื่อง 5
รูปที่ 1-6 แสดงการตวัดของปลายปากกา 6
รูปที่ 1-7 แสดงผลหลังผ่านขั้นตอนการกำจัดจุดไม่จำเป็นออก 6
รูปที่ 2-1 แสดงการอธิบายการทำงานของ Midpoint Line Algorithm 8
รูปที่ 2-2 แสดงผลหลังผ่านขั้นตอนการสร้างจุดให้ต่อเนื่องกัน 10
รูปที่ 2-3 แสดงการตวัดของปลายปากกา 10
รูปที่ 2-4 แสดงผลหลังผ่านขั้นตอนการกำจัดการตวัดของปลายปากกา 11
รูปที่ 2-5 แสดงการเลือกจุดที่สำคัญ 12
รูปที่ 2-6 แสดงการหาระยะห่างระหว่างจุดกับเส้นตรง 12
รูปที่ 2-7 แสดงผลหลังผ่านขั้นตอนการกำจัดจุดที่ไม่จำเป็นออก 14
รูปที่ 2-8 แสดงลักษณะของเฟรม 14
รูปที่ 2-9 แสดงการหาทิศทางของเฟรม 15
รูปที่ 2.10 แสดงการหามุมของเฟรม 16
รูปที่ 2-11 แสดงมุมเริ่มต้นของเฟรม 17
รูปที่ 2-12 แสดงการวัดมุมของเฟรม 18
รูปที่ 2-13 แสดงการกำหนดค่าให้กับช่วงองศา 19
รูปที่ 2-14 แสดงการโค้งตามเข็มนาฬิกาและทวนเข็มนาฬิกาของเฟรม 19
รูปที่ 2-15 รูปแสดงมุมสุดท้ายของเฟรม 20
รูปที่ 2-16 แสดงการวัดมุมของเฟรม 21
รูปที่ 2-17 แสดงการกำหนดค่าให้กับช่วงองศา 21
รูปที่ 2-18 แสดงการโค้งตามเข็มนาฬิกาและทวนเข็มนาฬิกาของเฟรม 22
รูปที่ 2-19 แสดงการแบ่งพื้นที่ของตัวอักษร 23
รูปที่ 2-20 แสดงเซลประสาททางชีวภาพ 25
รูปที่ 2-21 แสดงการตอบสนองทางไฟฟ้าของเยื่อหุ้มเซล 26
รูปที่ 2-22 แสดงไซแนปส์หรือจุดที่เชื่อมต่อของเซลประสาท 27
รูปที่ 2-23 แสดงภาพขยายของไซแนปส์ 28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2-24 แสดงลักษณะของนิวรอนเซลของเซลประสาท 29
รูปที่ 2-25 แสดงนิวรอนของโครงข่ายประสาท 29
รูปที่ 2-26 แสดงสมการเชิงเส้นอย่างง่าย ๆ 30
รูปที่ 2-27 แสดงสมการระดับตัดสิน 30
รูปที่ 2-28 แสดงลักษณะของนิวรอน 1 หน่วย ที่ประกอบด้วยฟังก์ชันการกระตุ้น 31
รูปที่ 2-29 แสดงลักษณะโครงสร้างของโครงข่ายประสาทลักษณะ 1 ชั้น 31
รูปที่ 2-30 แสดงลักษณะโครงสร้างของโครงข่ายประสาทลักษณะ 2 ชั้น 32
รูปที่ 2-31 แสดงสมการของฟังก์ชันการกระตุ้น 33
รูปที่ 2-32 รูปแสดงลักษณะของเพอเซปตรอน 34
รูปที่ 2-33 แสดงเพอเซปตรอนที่ใช้แก้ปัญหา 34
รูปที่ 2-34 แสดงกราฟความสัมพันธ์ X-Y 35
รูปที่ 2-35 แสดงลักษณะนิวรอนของโครงข่ายแบบปรับค่าย้อนกลับ 36
รูปที่ 2-36 แสดงกราฟของฟังก์ชันการกระตุ้น 37
รูปที่ 2.37 แสดง โครงข่ายประสาทแบบ OCON 38
รูปที่ 2.38 แสดง โครงข่ายประสาทแบบ ACON 38
รูปที่ 3-1 แสดงส่วนติดต่อกับผู้ใช้ของโปรแกรมเก็บลักษณะเด่นของตัวอักษร 41
รูปที่ 3-2 แสดงส่วนติดต่อกับผู้ใช้ของโปรแกรมฝึกสอนระบบ 48
รูปที่ 3-3 แสดงผลลัพธ์จากการทำงานของโปรแกรมฝึกสอนระบบ 50
รูปที่ 3-4 แสดงส่วนติดต่อกับผู้ใช้ของโปรแกรมแสดงการทำงานของระบบ 52
รูปที่ 3-5 แสดงผลลัพธ์จากการทำงานของโปรแกรมแสดงการทำงานของระบบ 53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้าที่
ตารางที่ 2-1 ตารางแสดง XOR	35
ตารางที่ 2-2 ตารางแสดงการเปรียบเทียบประสิทธิภาพระหว่าง โครงข่ายประสาทแบบ ACON และOCON	39
ตารางที่ 4-1 ตารางแสดงเวลาที่ใช้ในการฝึกสอนระบบตัวอักษรแตกต่างกัน	54
ตารางที่ 4-2 ตารางแสดงประสิทธิภาพของระบบตัวอักษรแตกต่างกัน	54
ตารางที่ 4-3 ตารางแสดงเวลาที่ใช้ในการฝึกสอนระบบตัวอักษรคล้ายกัน	55
ตารางที่ 4-4 ตารางแสดงประสิทธิภาพของระบบตัวอักษรคล้ายกัน	55
ตารางที่ 4-5 ตารางแสดงเวลาที่ใช้ในการฝึกสอนระบบที่สมบูรณ์	56
ตารางที่ 4-6 ตารางแสดงประสิทธิภาพของระบบสมบูรณ์	57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 บทนำ

ในปัจจุบันการป้อนข้อมูลโดยใช้ปากกาอิเล็กทรอนิกส์ได้รับความสนใจเป็นอย่างมาก ดังจะเห็นได้จากการนำปากกาอิเล็กทรอนิกส์มาใช้ร่วมกับคอมพิวเตอร์พกพา (Personal Digital Assistance) และ พจนานุกรมอิเล็กทรอนิกส์ โดยความพยายามที่จะนำมาใช้แทนแป้นพิมพ์ (Keyboard) เพื่อจุดประสงค์ในการลดขนาดของอุปกรณ์ดังกล่าวและความสะดวกในการใช้งานของผู้ใช้

การเขียนตัวอักษรด้วยปากกาอิเล็กทรอนิกส์แทนการพิมพ์ตัวอักษรบนแป้นพิมพ์เพื่อป้อนข้อมูลให้กับระบบคอมพิวเตอร์เป็นจุดเด่นที่สำคัญอย่างหนึ่งของการเลือกใช้ปากกาอิเล็กทรอนิกส์ เนื่องจากคนทั่วไปคุ้นเคยกับการเขียนหนังสือด้วยปากกาและดินสอตั้งแต่เด็ก ๆ ทำให้เกิดงานวิจัยและวิธีการต่าง ๆ มากมายในการทำให้ระบบคอมพิวเตอร์สามารถเรียนรู้ได้ว่าตัวหนังสือที่ผู้ใช้เขียนลงไปคือตัวอักษรอะไร

อย่างไรก็ตามในปัจจุบันการที่ระบบคอมพิวเตอร์จะทำการแยกแยะตัวอักษรภาษาไทยว่าเป็นตัวอักษรอะไรให้ถูกต้องนั้นเป็นเรื่องที่ยาก เนื่องจากความคล้ายคลึงกันของตัวอักษรอย่างเช่น ก ถ ฤ และ ภ ข กับ ฃ ด กับ ต ผ กับ ฝ เป็นต้น ความซับซ้อนของตัวอักษรภาษาไทย และ วิธีในการเขียนตัวอักษรภาษาไทยที่แตกต่างกัน เช่น สระ เอ บางคนเริ่มเขียนจากส่วนหัวบางคนเริ่มเขียนจากส่วนท้ายของตัวอักษร

แม้ว่าจะมีวิธีและทฤษฎีเกี่ยวกับการรู้จำตัวอักษรภาษาไทยออกมามากมาย แต่ก็ยังไม่ได้ผลเป็นที่น่าพอใจมากนัก

โครงข่ายประสาทแบบหลายชั้นเป็นระบบการรู้จำที่เป็นที่นิยมอย่างมากแบบหนึ่ง เนื่องจากความสามารถในการประมวลผลที่รวดเร็วและประสิทธิผลในการทำงานที่น่าพอใจ อย่างไรก็ตามระบบนี้มีข้อจำกัดที่สำคัญคือ จำนวนสิ่งที่ป้อนให้กับระบบจะต้องมีจำนวนคงที่ ซึ่งไม่สอดคล้องกับความยาวของตัวอักษรที่มีขนาดไม่คงที่ โครงงานนี้จึงได้นำเทคนิควินโดวิงมาใช้เพื่อแก้ไขข้อจำกัดดังกล่าว

1.2 จุดประสงค์

1.2.1 เพื่อพัฒนาระบบการรู้จำตัวอักษรภาษาไทยให้มีประสิทธิภาพมากขึ้น

1.2.2 เพื่อเรียนรู้ระบบการทำงานของโครงข่ายประสาทแบบหลายชั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงงาน

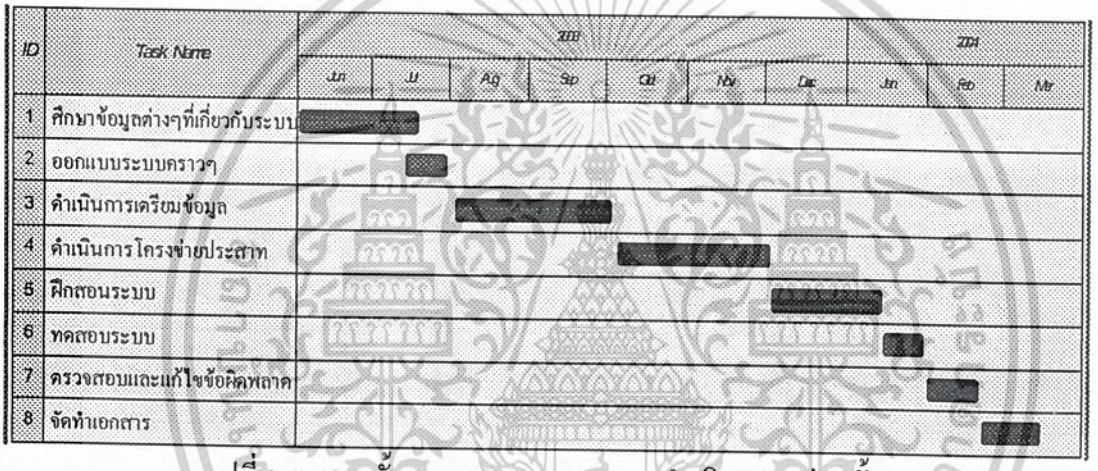
1.3.1 สร้างระบบการรู้จำตัวอักษรภาษาไทยที่สามารถแยกแยะตัวพยัญชนะบางตัวที่มีลักษณะคล้ายกัน เช่น ก กับ ภ ข กับ ฃ เป็นต้น ออกจากกันได้

1.4 ผลที่คาดว่าจะได้รับ

1.4.1 เข้าใจการทำงานของระบบการทำงานของโครงข่ายประสาทแบบหลายชั้น

1.4.2 โปรแกรมแสดงระบบรู้จำตัวอักษรภาษาไทยที่มีประสิทธิภาพ

1.5 ขั้นตอนการดำเนินงาน



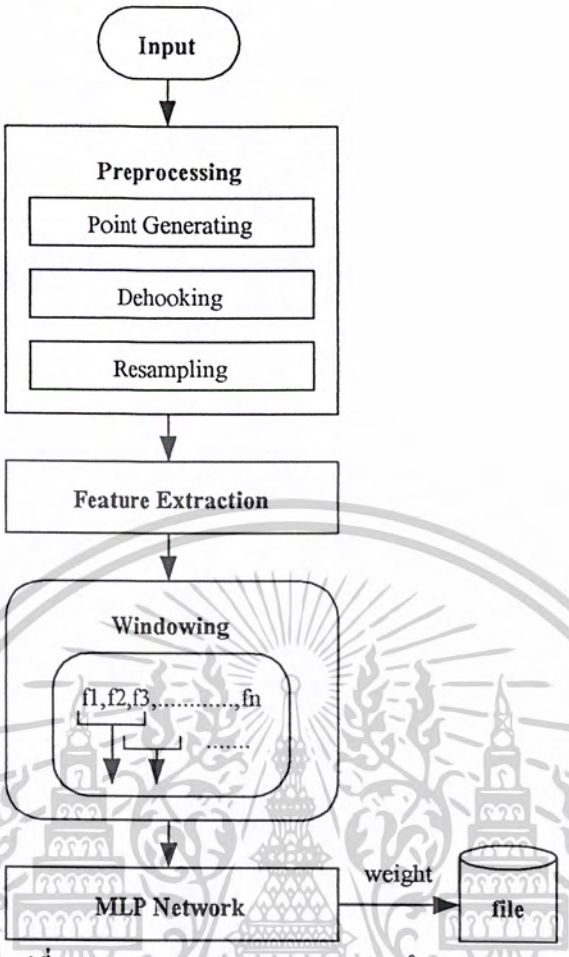
รูปที่ 1-1 แสดงขั้นตอนและระยะเวลาการดำเนินงานแต่ละขั้น

1.6 ระบบการทำงาน

ระบบการทำงานทั้งหมดสามารถถูกแบ่งออกได้เป็น 2 ช่วงการทำงาน ดังนี้

- ช่วงการฝึกสอน (Training)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

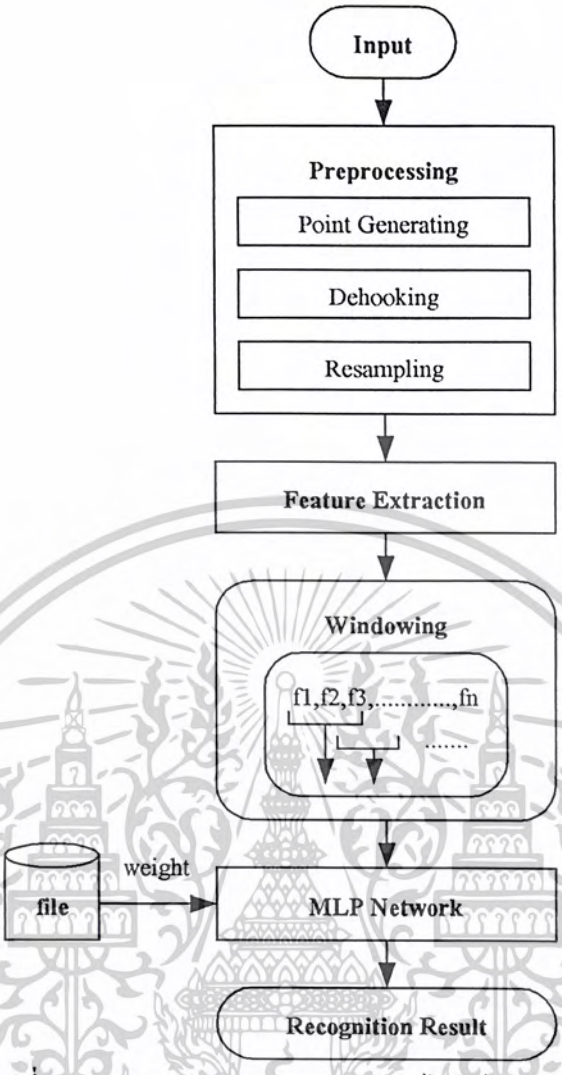


รูปที่ 1-2 แสดงระบบการทำงานช่วงฝึกสอนระบบ

ช่วงการทำงานนี้เป็นช่วงฝึกสอนระบบเพื่อให้ระบบสามารถรู้จำตัวอักษรว่า ตัวอักษรที่ป้อนไปด้วยการเขียนนั้นเป็นตัวอะไร โดยจะใช้ข้อมูลที่เกิดจากการเขียนตัวอักษรของคนหลาย ๆ คน คนละหลาย ๆ ครั้งเป็นข้อมูลในการฝึกสอนระบบ เมื่อทำการฝึกสอนจนได้ผลลัพธ์เป็นที่น่าพอใจแล้วจะทำการเก็บค่าน้ำหนักของโครงข่ายประสาทที่ให้ผลลัพธ์เป็นที่น่าพอใจนั้นไว้ เพื่อนำไปใช้ต่อไป

- ช่วงนำไปใช้งานจริง (Test)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1-3 แสดงระบบการทำงานช่วงนำไปใช้งานจริง

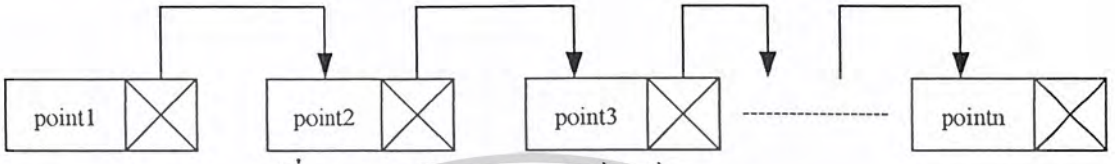
ช่วงการทำงานนี้เป็นช่วงในการนำไปใช้งานจริง โดยจะนำค่าน้ำหนักของโครงข่ายประสาทจากช่วงการทำงานช่วงที่แล้วมาใช้ โดยผลที่ได้จากระบบ (Recognition Result) จะถูกต้องเพียงใดนั้นขึ้นอยู่กับช่วงการฝึกสอนระบบว่ามีประสิทธิภาพเพียงใด

1.6.1 การรับข้อมูลอินพุต (Input)

ข้อมูลที่รับเข้ามาขณะเขียนตัวอักษรด้วยปากกาอิเล็กทรอนิกส์หรือด้วยการลากเมาส์จะเป็นคู่ลำดับ x, y ซึ่งมีลักษณะเป็นจุดที่ต่อเนื่องกันไปเรื่อย ๆ โดยจะนับการเขียนตัวอักษรตั้งแต่เริ่มลงปลายปากกาเขียนจนกระทั่งยกปลายปากกาขึ้นว่า 1 stroke เช่น ตัวอักษร ก มี 1 stroke ตัวอักษร ส มี 2 stroke เป็นต้น ดังนั้นจึงสรุปได้ว่าข้อมูลที่รับเข้ามาอาจมีได้หลาย stroke แต่ละ stroke จะประกอบไปด้วยลำดับ (Sequence) ของคู่ลำดับ x, y โดยจะเก็บเป็นอาร์เรย์ โดย 1 อาร์เรย์ เท่ากับ 1 stroke

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากอาร์เรย์ ในภาษาจาวาต้องมีการกำหนดขนาดของอาร์เรย์ที่แน่นอน (fixed size) ซึ่งไม่สอดคล้องกับความเป็นจริง คือ การเขียนของแต่ละคนหรือการเขียนในแต่ละครั้งไม่สามารถควบคุมความยาว (คู่ลำดับ) ของ stroke ได้ โครงงานนี้จึงใช้คลาส ArrayList ซึ่งมีลักษณะเหมือนลิงค์ลิสท์ เนื่องจากลิงค์ลิสท์สามารถเปลี่ยนขนาดได้ตลอดการทำงานของโปรแกรม อีกทั้งยังสามารถลบ เพิ่ม และ เปลี่ยนตำแหน่งได้ง่าย แสดงลักษณะของลิงค์ลิสท์ได้ดังนี้



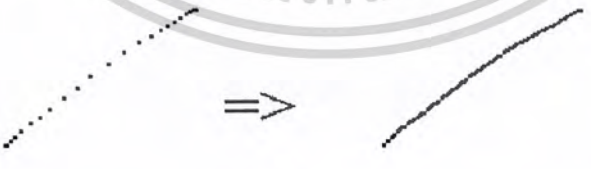
รูปที่ 1-4 แสดงลักษณะของลิงค์ลิสท์

1.6.2 การเตรียมข้อมูล (Preprocessing)

เป็นขั้นตอนในการจัดเตรียม เปลี่ยนแปลงข้อมูลให้เหมาะสมก่อนนำไปดำเนินการในขั้นตอนต่อไปซึ่งประกอบไปด้วย 3 ขั้นตอน ดังนี้

1.6.2.1 การสร้างจุดให้ต่อเนื่องกัน (Point Generating)

เนื่องจากในขณะเขียนตัวอักษรด้วยปากกาอิเล็กทรอนิกส์ หรือด้วยการลากเมาส์ของแต่ละคนในแต่ละครั้งมีความเร็วไม่เท่ากัน และเนื่องจากความเร็วนี้เองทำให้เครื่องคอมพิวเตอร์ไม่สามารถเก็บทุก ๆ จุดที่เราเขียนหรือลากเมาส์ผ่านได้ เช่น การเขียนหรือการลากเมาส์ผ่านจุด 10 จุด เครื่องคอมพิวเตอร์อาจเก็บจุดให้เราได้เพียง 6 จุด หรือถ้าเราเขียนหรือลากเมาส์ผ่านเร็วขึ้น เครื่องคอมพิวเตอร์อาจเก็บจุดให้เราได้เพียง 3 จุด ก็ได้ เป็นต้น ซึ่งจุดที่เก็บได้แทนที่จะห่างกัน 1 pixel กับกลายเป็นว่าจุด 2 จุดที่อยู่ใกล้กันที่เก็บได้ไม่ได้อยู่ใกล้กันเลย ขั้นตอนนี้จะทำการสร้างจุดที่ขาดหายไปให้ ดังภาพ



รูปที่ 1-5 แสดงผลหลังผ่านขั้นตอนการสร้างจุดให้ต่อเนื่อง

1.6.2.2 การกำจัดการตวัดของปลายปากกา (Dehooking)

เนื่องจากขณะที่เราเริ่มลงปลายปากกาเพื่อเขียนหรือยกปลายปากกาขึ้น ซึ่งมีลักษณะดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 1-6 แสดงการตวัดของปลายปากกา

ส่วนที่เกิดจากการตวัดปลายปากกาถ้าไม่มีการกำจัดออกอาจทำให้ระบบเข้าใจผิดได้ เช่น จากรูปด้านซ้ายมือ มีการตวัดปากกาที่ส่วนหัวของตัวอักษร ระบบอาจเข้าใจว่าเป็น ภ หรือ ถ ได้ แทนที่จะเป็น ก หรือ จากรูปด้านขวามือ มีการตวัดปลายปากกาที่ส่วนปลายของตัวอักษร ระบบอาจเข้าใจว่าเป็น ฮ แทนที่จะเป็น อ ก็ได้ การตวัดนี้จะทำการกำจัดการตวัดแบบนี้ทิ้ง

1.6.2.3 การกำจัดจุดที่ไม่จำเป็นออก (Resampling)

ขั้นตอนนี้เป็นขั้นตอนในการลดจุดคู่ลำดับที่ไม่จำเป็นออกไป เช่น เส้นตรงที่ประกอบด้วยจุดคู่ลำดับทั้งหมด 10 จุด แต่จริง ๆ แล้วจุดที่สำคัญมีเพียง 2 จุด คือ จุดเริ่มต้นและจุดสุดท้าย เพราะเพียง 2 จุดนี้ก็เพียงพอที่จะบอกถึงเส้นตรงได้แล้ว

รูปที่ 1-7 แสดงผลหลังผ่านขั้นตอนการกำจัดจุดไม่จำเป็นออก

1.6.2.4 การดึงลักษณะเด่น (Feature Extraction)

ขั้นตอนนี้เป็นขั้นตอนในการดึงลักษณะเด่น ซึ่งบอกความแตกต่างของตัวอักษรแต่ละตัวออกมา ลักษณะเด่นที่แตกต่างกันของตัวอักษรนี้ ทำให้ระบบที่สร้างขึ้นสามารถแยกความแตกต่างของตัวอักษรแต่ละตัวได้ ลักษณะเด่นที่นำมาใช้มีทั้งหมด 12 แบบ ซึ่งถูกแบ่งออกเป็น 2 กลุ่มใหญ่ ๆ คือ ลักษณะเด่นของเฟรม (เฟรม คือ การเชื่อมต่อระหว่างจุด 2 จุดที่อยู่ติดกันของข้อมูลที่ผ่านการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Resampling มาแล้วซึ่งใน 1 ตัวอักษรจะประกอบไปด้วยเฟรมหลาย ๆ เฟรม) และลักษณะเด่นของตัวอักษร (Global Feature)

โดยลักษณะเด่นของเฟรมจะประกอบไปด้วยลักษณะเด่นทั้งหมด 6 แบบ ได้แก่ ความยาว ทิศทาง ความน่าจะเป็นในการเป็นจุดเริ่มต้น ความน่าจะเป็นในการเป็นจุดสุดท้าย มุมเริ่มต้น และมุมปลาย ของแต่ละเฟรม

ลักษณะเด่นของตัวอักษรจะประกอบไปด้วยลักษณะเด่น 6 แบบ โดยหาได้จากการแบ่งพื้นที่ของตัวอักษรตามแนวนอน (horizontal) 3 ส่วน และ ตามแนวตั้ง (vertical) 3 ส่วน โดยดูว่าแต่ละส่วนมีจำนวนจุดที่ลากผ่านอยู่เท่าใด

1.6.3 เทคนิคการจัดกลุ่มของลักษณะเด่น (Windowing Technique)

เนื่องจากจำนวนของลักษณะเด่นที่ต้องป้อนให้กับโครงข่ายประสาท (Neural Network) จะต้องมีจำนวนคงที่ ซึ่งไม่สอดคล้องกับความเป็นจริง คือ เราไม่สามารถกำหนดตัวเลขที่ตายตัว (fixed number) ของเฟรมสำหรับตัวอักษรแต่ละตัวได้ แม้ว่าจะเป็นคน ๆ เดียวกันเขียนก็ตาม การที่จะแก้ปัญหาด้วยการลดจำนวนลักษณะเด่นบางตัวออกไป เพื่อให้มีจำนวนเท่ากับจำนวนที่ต้องป้อนให้กับโครงข่ายประสาทเป็นสิ่งที่ไม่ควร เนื่องจากจะทำให้ข้อมูลสำคัญบางอย่างขาดหายไปได้ เพื่อแก้ปัญหานี้เราจึงได้ใช้เทคนิคการจัดกลุ่มของลักษณะเด่นมาช่วย โดยจะจัดกลุ่มของลักษณะเด่นให้แต่ละกลุ่มมีจำนวนลักษณะเด่นเท่ากับจำนวนที่โครงข่ายประสาทต้องการ แล้วนำผลที่ได้จากแต่ละกลุ่มมารวมกันเพื่อตัดสินใจว่าเป็นตัวอักษรตัวใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

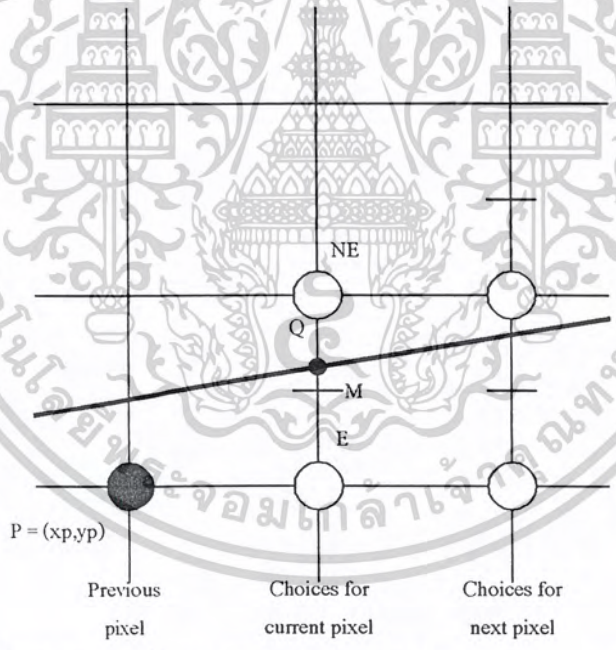
บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 การสร้างจุดให้ต่อเนื่องกัน

เนื่องจากในการเขียนตัวอักษร เครื่องคอมพิวเตอร์ไม่สามารถเก็บคู่ลำดับทุกคู่ลำดับที่เราลากผ่านได้ ยิ่งเราเขียนเร็วมากเท่าใดเครื่องคอมพิวเตอร์จะเก็บคู่ลำดับได้น้อยลงเท่านั้น ดังนั้นจึงจำเป็นต้องสร้างจุดที่ขาดหายไปขึ้นมา วิธีที่นำมาใช้ในการสร้างจุดที่ขาดหายไป คือ Midpoint Line Algorithm

2.1.1 Midpoint Line Algorithm

จะพิจารณาในช่วงที่ความชันมีค่าระหว่าง 0 และ 1 โดยกำหนดให้จุดเริ่มต้นซึ่งอยู่มุมล่างซ้ายแทนด้วยคู่ลำดับ (x_0,y_0) และมีจุดปลายซึ่งอยู่มุมบนขวาแทนด้วยคู่ลำดับ (x_1,y_1) ส่วนในช่วงอื่นที่ไม่ได้อยู่ในช่วงความชันระหว่าง 0 และ1 ให้นำวิธีการเดียวกันไปประยุกต์ใช้ได้ พิจารณารูปข้างล่างนี้



รูปที่ 2-1 ประกอบการอธิบายการทำงานของ Midpoint Line Algorithm

วงกลมสีดำหมายถึงจุดที่เลือกแล้วซึ่งในที่นี้แทนด้วยจุด $P(x_p,y_p)$ วงกลมสีขาวหมายถึงจุดที่จะตัดสินใจเลือกในขั้นตอนต่อไป ในที่นี้จุดที่จะเลือกครั้งต่อไปคือจุด E และจุด NE โดยการตัดสินใจในการเลือกจะพิจารณาจากจุดกึ่งกลางซึ่งแทนด้วย M ถ้าจุดกึ่งกลางอยู่เหนือเส้นตรงจะเลือกจุด E แต่ถ้าจุดกึ่งกลางอยู่ต่ำกว่าเส้นตรงจะเลือกจุด NE

2.1.2 วิธีการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เริ่มจากแสดงสมการเส้นตรงในรูป Implicit Function คือ $F(x,y) = ax+by+c = 0$ โดย a,b และ c เป็นสัมประสิทธิ์ ถ้า $dy = y_1 - y_0$ และ $dx = x_1 - x_0$ เราสามารถแสดงสมการเส้นตรงในรูปความชันได้ คือ

$$y = \frac{dy}{dx} * x + B$$

เมื่อนำไปแทนใน Implicit Function จะได้

$$F(x,y) = dy*x - dx*y + B*dx = 0$$

โดย $a = dy$, $b = -dx$ และ $c = B*dx$

ซึ่งถ้าค่า F(x,y) มีค่าเท่ากับ 0 หมายความว่าคู่ลำดับ (x,y) อยู่บนเส้นตรง ถ้ามีค่าเป็นบวกหมายความว่าคู่ลำดับ (x,y) อยู่ใต้เส้นตรง และ ถ้ามีค่าเป็นลบหมายความว่าคู่ลำดับ (x,y) อยู่เหนือเส้นตรง เนื่องจากเราต้องการหาความสัมพันธ์ของจุดกึ่งกลางกับเส้นตรงดังนั้นต้องแทนจุดกึ่งกลางไปในสมการเส้นตรง ได้ $F(M) = F(x_p+1, y+\frac{1}{2})$ โดยกำหนดให้ $d = F(x_p+1, y+\frac{1}{2})$ ถ้า $d > 0$ เราจะเลือกจุด NE แต่ถ้า $d \leq 0$ เราจะเลือกจุด E เป็นจุดต่อไป หลังจากเราเลือกจุดที่ต้องการได้แล้ว ขั้นตอนต่อไปเราสามารถเลือกจุดโดยทำการเปลี่ยนเฉพาะค่า d ซึ่งขึ้นอยู่กับว่าขั้นตอนที่แล้วเราเลือกจุด E หรือ จุด NE

ถ้าเราเลือกจุด E

$$d_{new} = F(x_p+2, y_p+\tfrac{1}{2}) = a(x_p+2) + b(y_p+\tfrac{1}{2}) + c$$

แต่

$$d_{old} = a(x_p+1) + b(y_p+\tfrac{1}{2}) + c$$

ดังนั้น

$$d_{new} = d_{old} + a$$

ซึ่ง a มีค่าเท่ากับ dy

แต่ถ้าเราเลือกจุด NE

$$d_{new} = F(x_p+2, y_p+\tfrac{3}{2}) = a(x_p+2) + b(y_p+\tfrac{3}{2}) + c$$

ดังนั้น

$$d_{new} = d_{old} + a + b$$

ซึ่งค่า a + b มีค่าเท่ากับ dy – dx

แต่เนื่องจากจุดกึ่งกลางจุดแรก คือ $(x_0+1, y_0+\frac{1}{2})$ ทำให้

$$\begin{aligned} F(x_0+1, y_0+\tfrac{1}{2}) &= a(x_0+1) + b(y_0+\tfrac{1}{2}) + c \\ &= a*x_0 + b*y_0 + c + a + \tfrac{b}{2} \\ &= F(x_0, y_0) + a + \tfrac{b}{2} \end{aligned}$$

แต่คู่ลำดับ (x_0, y_0) อยู่บนเส้นตรงทำให้ $F(x_0, y_0) = 0$ ดังนั้น $d_{strat} = a + \frac{b}{2} = dy - \frac{dx}{2}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่เนื่องจากเวลาที่เครื่องคอมพิวเตอร์ใช้ในการคำนวณการคูณหารจะใช้เวลามากกว่าการบวกลบจึง ต้องกำจัดตัวหาร 2 ออก โดยการคูณ 2 ตลอดสมการ ดังนั้นจึงสรุปได้ดังนี้

$$d_{start} = 2dy - 2dx$$

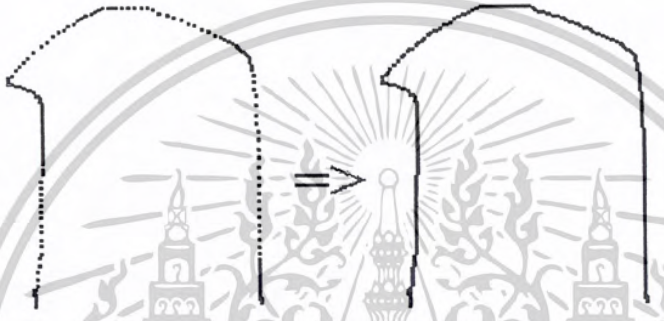
ถ้าเลือกจุด NE

$$d_{next} = 2dy - 2dx$$

แต่ถ้าเลือกจุด E

$$d_{next} = 2dy$$

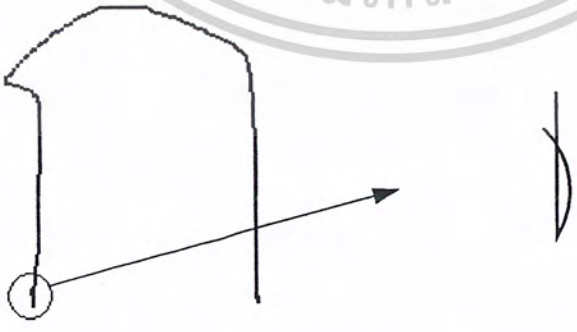
2.1.3 ตัวอย่างผลเมื่อผ่านขั้นตอนนี้



รูปที่ 2-2 แสดงผลหลังผ่านขั้นตอนการสร้างจุดให้ต่อเนื่องกัน

2.2 การกำจัดการตวัดของปลายปากกา

เป็นขั้นตอนในการตัดส่วนเกินที่เกิดจากการเขียน อาจจะเกิดจากการเขียนตวัดปลายปากกา และเพื่อป้องกันการทำงานผิดพลาด จึงได้มีการตัดส่วนเกินนั้นทิ้ง ซึ่งมักจะเกิดขึ้นที่จุดเริ่มต้นและจุดสุดท้ายของตัวอักษรลักษณะของการตวัดที่นำมาพิจารณาแสดงได้ดังภาพ



รูปที่ 2-3 แสดงการตวัดของปลายปากกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 การทำงาน

เริ่มจากพิจารณาที่คู่ลำดับแรกของ stroke ว่าอยู่ที่ตำแหน่งที่เท่าไร จากนั้นพิจารณาคู่ลำดับต่อไปว่ามีตำแหน่งเดียวกับคู่ลำดับแรกหรือมีระยะห่างจากคู่ลำดับแรกเท่ากับ 1 พิกเซลหรือไม่โดยมีเงื่อนไขว่าคู่ลำดับต่อไปต้องไม่ใช้คู่ลำดับคู่ที่สองและทำไปจนถึงคู่ลำดับ 1 ใน 4 ของคู่ลำดับทั้งหมดเนื่องจากการตวัดมักเกิดที่ช่วงแรกและช่วงสุดท้ายของ stroke ถ้าใช่จะนำคู่ลำดับแรกจนถึงคู่ลำดับที่มีตำแหน่งเดียวกับคู่ลำดับแรกหรือมีระยะห่างจากคู่ลำดับแรกเท่ากับ 1 พิกเซลมาพิจารณาหาพื้นที่ และนำไปเปรียบเทียบกับพื้นที่ทั้งหมดของตัวอักษรนั้น โดยพื้นที่จะหาจากสมการดังนี้

ความกว้าง = ค่า x ที่มากที่สุดในคู่ลำดับทั้งหมด – ค่า x ที่น้อยที่สุดในคู่ลำดับทั้งหมด + 1

ความยาว = ค่า y ที่มากที่สุดในคู่ลำดับทั้งหมด – ค่า y ที่น้อยที่สุดในคู่ลำดับทั้งหมด + 1

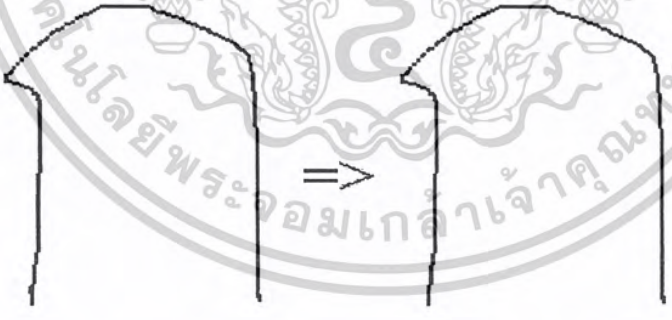
พื้นที่ = ความกว้าง * ความยาว

จากนั้นนำพื้นที่ทั้งสองมาทำการเปรียบเทียบ ถ้าพื้นที่ที่เกิดจากการเขียนตวัดมีค่าน้อยกว่า 1/10 ของพื้นที่ตัวอักษรก็จะทำการตัดทิ้ง โดยการตัดจะพิจารณาจากจำนวนคู่ลำดับที่อยู่ในกรอบสี่เหลี่ยมของส่วนเกินว่ามีจำนวนคู่ลำดับเท่าใด แล้วนำมาหารสอง ได้เท่าใดก็จะนำไปตัดออกโดยเริ่มจากจุดเริ่มต้น

หมายเหตุ ถ้าพิจารณาการตวัดที่ปลายของตัวอักษรจะทำแบบเดียวกันแต่ทำย้อนกลับ

ในกรณีที่ตัวอักษรหนึ่งตัวมีการเขียนสองครั้ง (2 stroke) เช่น ษ ศ ส ญ ฐ ก็จะทำการพิจารณาแยกทีละ stroke

2.2.2 ตัวอย่างผลเมื่อผ่านขั้นตอนนี้



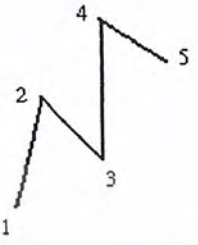
รูปที่ 2-4 แสดงผลหลังผ่านขั้นตอนการกำจัดการตวัดของปลายปากกา

2.3 การกำจัดจุดที่ไม่จำเป็นออก

2.3.1 หลักการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนนี้จะทำการเก็บเฉพาะจุดที่ความสำคัญต่อการนำไปทำขั้นตอนต่อไปเท่านั้นเช่น เส้นตรงเราจะเก็บเพียงจุด 2 จุด คือ จุดเริ่มต้นและจุดสุดท้ายของเส้นตรงนั้นหรือถ้ารูปมีลักษณะดังนี้



รูปที่ 2-5 แสดงการเลือกจุดที่สำคัญ

เราจะทำการเก็บเพียง 5 จุด คือ จุดที่1 2 3 4 และ 5 ตามลำดับซึ่งจุดเพียง 5 จุดนี้ เราสามารถนำมาสร้างเป็นรูปเดิมได้ โดยมีลักษณะไม่ต่างไปจากรูปเดิม
จากรูปจะสังเกตเห็นว่าจุดเริ่มต้นและจุดสุดท้ายจะต้องมีไม่ว่าจะเป็นรูปใด ๆ ก็ตาม

การระบุว่าจุดใดมีความสำคัญมากพอที่จะเลือกเก็บไว้นั้นจะใช้หลักการที่ว่า จุดที่อยู่ระหว่างจุดเริ่มต้นและจุดสุดท้ายที่นำมาสร้างเป็นเส้นตรง มีระยะห่างจากจุดนั้นลากมาตั้งฉากกับเส้นตรงที่สร้างขึ้นมีค่าเท่าใด โดยถ้ามีระยะห่างมากกว่าที่กำหนดไว้ แสดงว่าจุดนั้นเป็นจุดสำคัญ แต่ถ้ามีระยะห่างน้อยกว่าที่กำหนดไว้แสดงว่าจุดนั้นเป็นจุดไม่สำคัญซึ่งไม่ควรเก็บไว้

โดยสูตรที่ใช้ในการหาระยะห่างระหว่างจุดกับเส้นตรงแสดงได้ดังนี้

รูปที่ 2-6 แสดงการหาระยะห่างระหว่างจุดกับเส้นตรง

สมมุติให้จุดที่อยู่ระหว่างจุดที่ 1 และ 5 มีจุด 3 จุด คือจุดที่ 2 3 และ 4 นำจุดที่ 1 และ 5 มาสร้างสมการเส้นตรง

$$Ax + By + C = 0$$

โดยค่า A, B และ C นำมาจากสมการ $y = mx + b \Rightarrow mx - y + b = 0$

เพราะฉะนั้นได้ $A = m = dy/dx$ โดย dx จะมีค่าเป็น 0 ไม่ได้

$B = -1$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

C = b = -mx+y โดย x และ y แทนด้วยจุดใด ๆ ที่อยู่บนเส้นตรงนั้น

หลังจากได้สมการเส้นตรงมาแล้ว ต่อไปจะเป็นการหาระยะห่างจากจุดใด ๆ ที่อยู่ระหว่างจุด 2 จุดที่นำมาสร้างเส้นตรงกับเส้นตรงที่สร้างจากจุด 2 จุดนั้น โดยมีสูตรดังนี้

$$d = \frac{|Ax + By + C|}{\sqrt{A^2 + B^2}}$$

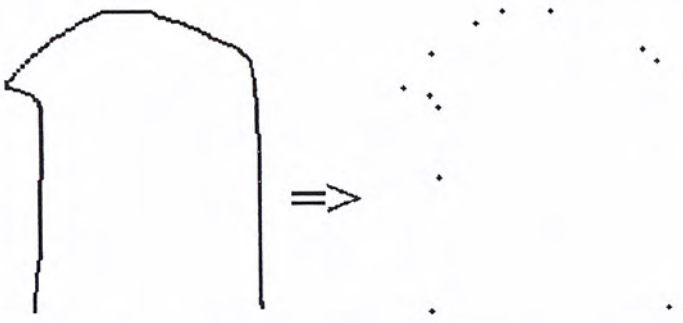
โดยค่า A, B และ C ได้มาจากสมการเส้นตรงที่ผ่านมา ค่า d เป็นระยะห่างจากจุดใด ๆ ที่อยู่ระหว่างจุด 2 จุดที่นำมาสร้างเส้นตรงกับเส้นตรงที่สร้างจากจุด 2 จุดนั้น ส่วนค่า x และ y แทนด้วยจุดใดๆที่ต้องการนำมาวัดระยะห่าง โดยจะเริ่มวัดจากจุดที่ 1 ถึง จุดที่ 5 จุดใดมีระยะห่างมากกว่าที่กำหนดจะถูกเก็บไว้

2.3.2 การทำงาน

เริ่มต้นเราจะคิดที่ละ stroke (ตั้งแต่เริ่มลงปลายปากกาเขียนจนยกปลายปากกาขึ้นนับเป็น 1 stroke) โดยจะนำจุดแรกของ stroke มาเป็นจุดเริ่มต้นเพื่อนำไปสร้างสมการเส้นตรงจากนั้นจะเริ่มหาจุดต่อไปที่จะนำมาเป็นจุดสุดท้ายของสมการเส้นตรง โดยมีเงื่อนไขว่าจุดที่นำมาเป็นจุดสุดท้ายของสมการเส้นตรง โดยมีเงื่อนไขว่าจุดที่จะนำมาเป็นจุดสุดท้ายนั้นต้องมีค่า x ไม่เท่ากับค่า x ของจุดเริ่มต้น เพราะจะทำให้ค่า dx = 0 ส่งผลให้ไม่สามารถหาค่า m ได้ และ ต้องมีค่า y ไม่เท่ากับค่า y ของจุดเริ่มต้นเพื่อช่วยลดเวลาในการคำนวณหาระยะห่างในกรณีที่จุดต่อไปอยู่ในแนวนอนเส้นเดียวกัน เมื่อเราได้จุดเริ่มต้นและจุดสุดท้ายมาแล้ว จากนั้นนำไปหาค่า A, B และ C จากนั้นนำจุดทั้งหมดที่อยู่ระหว่างจุดเริ่มต้นและจุดสุดท้ายมาทำการหาระยะห่างจากจุดนั้นกับเส้นตรงจากสมการข้างต้นโดยเก็บเฉพาะจุดที่มีระยะห่างมากที่สุดไว้เพียงจุดเดียว แล้วนำระยะห่างนี้ไปเปรียบเทียบกับเกณฑ์ที่กำหนดไว้ ถ้ามากกว่าเกณฑ์ที่กำหนดไว้ให้ทำการเก็บจุดนั้นไว้แล้วเริ่มการคำนวณใหม่โดยให้จุดเริ่มต้นของสมการเส้นตรงเส้นต่อไปคือจุดที่เพิ่งได้เก็บไป แต่ถ้ามีค่าน้อยกว่าหรือเท่ากับเกณฑ์ที่กำหนดไว้ให้เปลี่ยนจุดสุดท้ายของสมการเส้นตรงไปเป็นจุดถัดไป แล้วเริ่มต้นคำนวณใหม่ ในกรณีที่จุดถัดไปมีค่า x เท่ากับค่า x ของจุดเริ่มต้น ทำให้ไม่สามารถหาสมการเส้นตรงได้ (dx = 0 ไม่สามารถหาค่า m ได้) การหาระยะห่างทำโดยหาค่าความแตกต่างระหว่างค่า x ของจุดที่อยู่ระหว่างจุดเริ่มต้นและจุดสุดท้ายที่ละจุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 ตัวอย่างผลเมื่อผ่านขั้นตอนนี้



รูปที่ 2-7 แสดงผลหลังผ่านขั้นตอนการกำจัดจุดที่ไม่จำเป็นออก

2.4 การดึงลักษณะเด่น

2.4.1 หลักการ

หลังจากผ่านขั้นตอน Resampling แล้ว ข้อมูลที่ได้จะเหลือแต่คู่ลำดับที่สำคัญ ๆ เท่านั้น โดยเราจะกำหนดให้ช่วงระหว่างจุดคู่ลำดับ 2 จุด คือ 1 เฟรม (frame) ดังรูป

รูปที่ 2-8 แสดงลักษณะของเฟรม

จากรูปจะเห็นได้ว่าจำนวนเฟรมจะมีค่าน้อยกว่าจำนวนจุดคู่ลำดับอยู่ 1 เสมอ

โดยจะนำเฟรมแต่ละเฟรมมาหาลักษณะเด่นที่จะนำไปใช้เป็นข้อมูลป้อนให้กับโครงข่ายประสาท ซึ่งในแต่ละเฟรมจะประกอบด้วยลักษณะเด่นทั้งหมด 12 แบบ แบ่งออกเป็น 2 กลุ่มใหญ่ ๆ คือ ลักษณะเด่นของเฟรม ซึ่งประกอบไปด้วยลักษณะเด่น 6 แบบ คือ ความยาวเฟรม ทิศทางของเฟรม ความน่าจะเป็นในการเป็นเฟรมเริ่มต้น ความน่าจะเป็นในการเป็นเฟรมสุดท้าย มุมเริ่มต้นของเฟรม และมุมปลายของเฟรม และ ลักษณะเด่นของตัวอักษร ซึ่งประกอบไปด้วยลักษณะเด่น 6 แบบ โดยจะแบ่งพื้นที่ของตัวอักษรตามแนวนอนออกเป็น 3 ส่วน ตามแนวตั้งออกเป็น 3 ส่วน แล้วดูว่าแต่ละส่วนมีจุดที่ลากผ่านอยู่เท่าไร ซึ่งลักษณะเด่นทั้ง 12 แบบ จะมีค่าอยู่ระหว่าง –1 และ 1 ซึ่งเป็นเงื่อนไขของโครงข่ายประสาทซึ่งจะกล่าวถึงในเรื่องโครงข่ายประสาทเรื่องต่อไป

2.4.2 ลักษณะเด่นทั้ง 12 แบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2.1 ความยาวเฟรม (LengthOfLine) : นำเฟรมแต่ละเฟรมซึ่งประกอบด้วยจุด 2 จุด มาทำการวัดความยาวของเฟรมโดยใช้สูตร

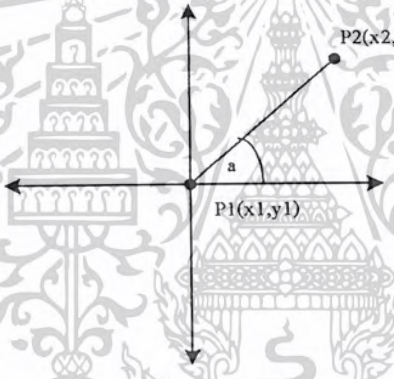
$$\text{ความยาวเฟรม} = \sqrt{(x1 - x2)^2 + (y1 - y1)^2}$$

เมื่อ x1 y1 เป็นคู่ลำดับของจุดที่ 1 และ x2 y2 เป็นคู่ลำดับของจุดที่ 2 เนื่องจากค่าที่นำมาใช้ต้องมีค่าไม่เกิน 1 เพราะฉะนั้นความยาวของแต่ละเฟรมที่นำไปใช้จะมีค่าเท่ากับ

$$\text{ความยาวเฟรมที่ i} = \frac{\text{ความยาวเฟรมที่ i}}{\text{ความยาวของทุกเฟรมรวมกัน}}$$

หมายเหตุ ถ้าเฟรมที่กำลังสนใจเป็นเฟรมที่อยู่ระหว่าง stroke ที่หนึ่งและสอง กำหนดให้ความยาวเฟรมเท่ากับ –1

2.4.2.2 ทิศทางของเฟรม (Direction) : การหาทิศทางของเฟรมจะหาจากมุมที่เฟรมทำกับระนาบแกน x โดยจะวัดมุมในทิศทางทวนเข็มนาฬิกา ดังภาพ



รูปที่ 2-9 แสดงการหาทิศทางของเฟรม

เมื่อ a คือมุมที่ต้องการ

โดยขั้นตอนในการทำจะเริ่มจาก นำจุด 2 จุดของแต่ละเฟรมมาทำการหาทิศทาง โดยกำหนดให้ จุดที่ 1 แทนด้วย P1 และ จุดที่ 2 แทนด้วย P2 จากนั้นจะแบ่งขั้นตอนออกเป็น 5 กรณีดังนี้

1. P1.x = P2.x และ P1.y > P2.y a จะมีค่าเท่ากับ 90 องศา
2. P1.x = P2.x และ P1.y < P2.y a จะมีค่าเท่ากับ 270 องศา
3. P1.y = P2.y และ P1.x > P2.x a จะมีค่าเท่ากับ 180 องศา
4. P1.y = P2.y และ P1.x ≤ P2.x a จะมีค่าเท่ากับ 0 องศา
5. นอกเหนือจาก 4 กรณีนี้จะทำการหาความชันของเฟรมโดยใช้สูตร

$$\text{ความชัน} = \frac{(P2.y - P1.y)}{(P2.x - P1.x)}$$

จากนั้นนำค่าความชันที่ได้ไปหามุมโดยใช้สูตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มุมระหว่างเฟรมกับระนาบแกน x = $\left|\frac{\tan^{-1}(slop)*180}{\pi}\right|$

เมื่อ slop มีค่าเท่ากับความชันที่หาได้

โดยมุมที่ได้ออกมาจะมีค่าอยู่ระหว่าง 0 องศา ถึง 90 องศา เท่านั้นซึ่งไม่ตรงกับความเป็นจริงดังนั้น ต้องมีขั้นตอนเพิ่มเติมโดยแบ่งออกเป็นอีก 4 กรณี คือ

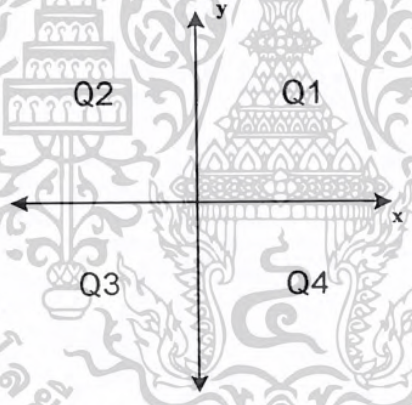
5.1 P1.x > P2.x และ P1.y > P2.y a จะมีค่าเท่ากับ (90-มุมที่หาได้)+90 ซึ่งอยู่ใน Q2

5.2 P1.x > P2.x และ P1.y < P2.y a จะมีค่าเท่ากับ มุมที่หาได้+180 ซึ่งอยู่ใน Q3

5.3 P1.x < P2.x และ P1.y < P2.y a จะมีค่าเท่ากับ (90-มุมที่หาได้)+270 ซึ่งอยู่ใน Q4

5.4 นอกเหนือจาก 3 a จะมีค่าเท่ากับ มุมที่หาได้ ซึ่งอยู่ใน Q1

โดย Q1, Q2, Q3 และ Q4 แสดงได้ดังรูปข้างล่างนี้



รูปที่ 2.10 แสดงการหามุมของเฟรม

แต่เนื่องจากค่าที่นำมาใช้ต้องมีค่าไม่เกิน 1 เพราะฉะนั้นมุมของแต่ละเฟรมที่นำไปใช้จะมีค่าเท่ากับ

มุมที่นำไปใช้ = $\frac{a}{360}$ องศา

2.4.2.3 ความน่าจะเป็นในการเป็นเฟรมเริ่มต้น (StartLine) : โดยกำหนดให้จุดเริ่มต้นมีค่ามากที่สุดซึ่งมีค่าเท่ากับจำนวนจุดหรือจำนวนคู่ลำดับทั้งหมดหลังการ Resampling ลบ 1 จุดที่เกิดตามมาที่หลังจะมีค่าน้อยลงเรื่อย ๆ ที่ละ 1 จนถึงจุดสุดท้ายมีค่าน้อยที่สุดซึ่งมีค่าเท่ากับ 1 โดยความน่าจะเป็นในการเป็นเฟรมเริ่มต้นหาได้โดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\text{ความน่าจะเป็นในการเป็นเฟรมเริ่มต้นของเฟรมที่ i} = \frac{\text{ค่าของจุดเริ่มต้นของเฟรมที่ i ที่กำหนดไว้}}{\text{จำนวนจุดทั้งหมด - 1}}$$

หมายเหตุ ถ้าเฟรมที่กำลังสนใจเป็นเฟรมที่อยู่ระหว่าง stroke ที่หนึ่งและสองกำหนดให้ความน่าจะเป็นในการเป็นเฟรมเริ่มต้นมีค่าเท่ากับ –1

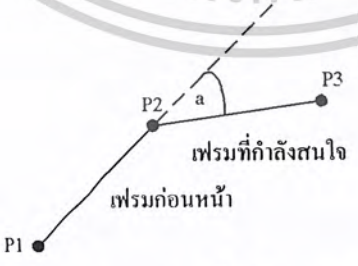
2.4.2.4 ความน่าจะเป็นในการเป็นเฟรมสุดท้าย (EndLine) : โดยกำหนดให้จุดเริ่มต้นมีค่าน้อยที่สุดซึ่งมีค่าเท่ากับ 1 จุดที่เกิดตามมาที่หลังจะมีค่ามากขึ้นเรื่อย ๆ ที่ละ 1 จนถึงจุดสุดท้ายมีค่ามากที่สุดซึ่งมีค่าเท่ากับจำนวนจุดหรือจำนวนคู่ลำดับทั้งหมดหลังการ Resampling ลบ 1 โดยความน่าจะเป็นในการเป็นเฟรมสุดท้ายหาได้โดย

$$\text{ความน่าจะเป็นในการเป็นเฟรมสุดท้ายของเฟรมที่ i} = \frac{\text{ค่าของจุดสุดท้ายของเฟรมที่ i ที่กำหนดไว้}}{\text{จำนวนจุดทั้งหมด - 1}}$$

หมายเหตุ ถ้าเฟรมที่กำลังสนใจเป็นเฟรมที่อยู่ระหว่าง stroke ที่หนึ่งและสองกำหนดให้ความน่าจะเป็นในการเป็นเฟรมสุดท้ายมีค่าเท่ากับ –1

2.4.2.5 มุมเริ่มต้นของเฟรม (AngleStart) : คือมุมเริ่มต้นของเฟรมปัจจุบันหรือเฟรมที่กำลังสนใจอยู่ทำกับเฟรมก่อนหน้า โดยถ้าเป็นเฟรมแรกของ stroke จะกำหนดให้มีค่าเท่ากับ –1 ถ้าไม่ใช้มุมเริ่มต้นของเฟรมแรกสามารถหาได้ดังนี้

เริ่มต้นโดยนำจุด 3 จุดมาคำนวณ ซึ่งจุดแรกคือ จุดเริ่มต้นของเฟรมที่กำลังสนใจอยู่ จุดที่สองคือ จุดสุดท้ายของเฟรมที่กำลังสนใจอยู่ และ จุดที่สามคือ จุดเริ่มต้นของเฟรมก่อนหน้า ซึ่งกำหนดให้แทนด้วย P2, P3, และ P1 ตามลำดับ สามารถแสดงได้ดังรูป



รูปที่ 2-11 แสดงมุมเริ่มต้นของเฟรม

ซึ่งมุม a หาได้จากสูตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

55734

$$a = \cos^{-1}\left[\frac{\overrightarrow{P1P2} \bullet \overrightarrow{P2P3}}{\left|\overrightarrow{P1P2}\right| * \left|\overrightarrow{P2P3}\right|}\right]$$

โดย $\overrightarrow{P1P2}$ คือ เวกเตอร์มีค่าเท่ากับ x1i+y1j มีขนาดเท่ากับ $\left|\overrightarrow{P1P2}\right|$ และ $\overrightarrow{P2P3}$ คือเวกเตอร์มีค่าเท่ากับ x2i+y2j มีขนาดเท่ากับ $\left|\overrightarrow{P2P3}\right|$ นำมาแทนในสูตรข้างต้นได้สูตรดังนี้

$$a = \cos^{-1}\left[\frac{(x1x2 + y1y2)}{\sqrt{x1^2 + y1^2} * \sqrt{x2^2 + y2^2}}\right]$$

สังเกตได้ว่าขนาดของเวกเตอร์ต้องมีขนาดไม่เท่ากับศูนย์ และมุมที่ได้มีหน่วยเป็นเรเดียน ดังนั้นจึงต้องแปลงเป็นหน่วยของศาโดยใช้สูตร

$$a\ (\text{องศา}) = \frac{a_{radian} * 180}{\pi}$$

เนื่องจากมุมที่นำไปใช้จะวัดในลักษณะทวนเข็มนาฬิกาดังรูปซึ่งมีสองลักษณะ

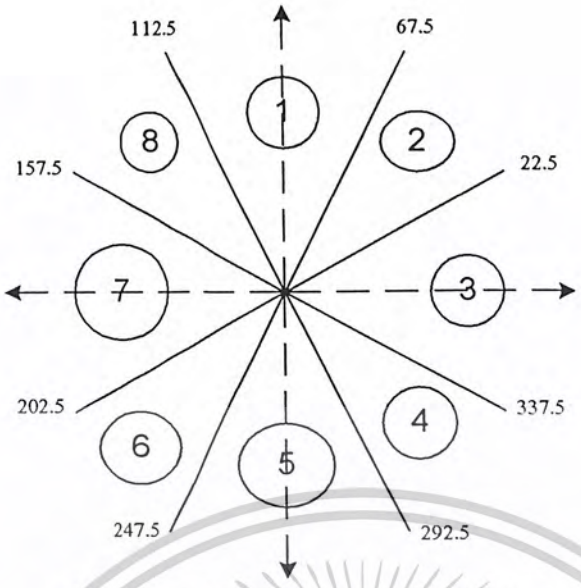
รูปที่ 2-12 แสดงการวัดมุมของเฟรม

เพราะฉะนั้นต้องตรวจสอบลักษณะของเฟรมว่ากำลังโค้งตามเข็มนาฬิกาหรือกำลังโค้งทวนเข็มนาฬิกา

- ถ้ามีลักษณะกำลังโค้งตามเข็มนาฬิกาตามรูปด้านซ้ายมือ ต้องเอามุมที่ได้ลบออกจาก 360
- ถ้ามีลักษณะกำลังโค้งทวนเข็มนาฬิกาตามรูปด้านขวามือ สามารถนำมุมที่หามาได้ไปใช้ได้เลย

การตรวจสอบว่าเฟรมกำลังจะโค้งไปทางใดนั้นจะใช้วิธี Chain Code ช่วยโดยจะแทนเฟรมด้วยเลขจำนวนเต็มตั้งแต่ 1 ถึง 8 โดยจะแทนด้วยค่าอะไรนั้นขึ้นอยู่กับมุมในหน่วยองศาระหว่างเฟรมกับระนาบแกน x ซึ่งวัดทวนเข็มนาฬิกา โดยแบ่งเป็นลักษณะดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-13 แสดงการกำหนดค่าให้กับช่วงองศา

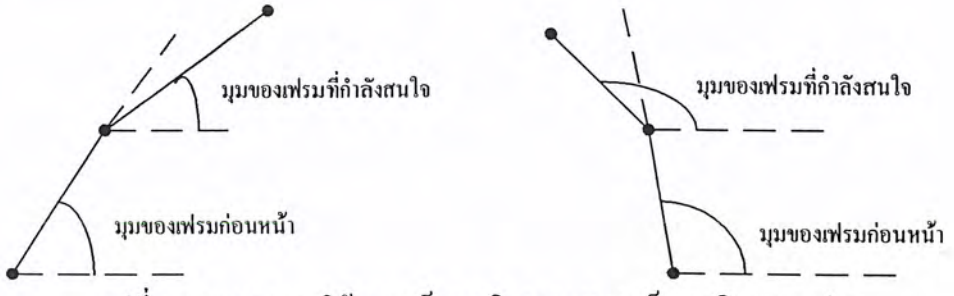
โดยจะแบ่งออกเป็น 8 ส่วน ส่วนละ 45 องศา โดยดูว่าแต่ละเฟรมมีมุมในหน่วยของศาระหว่างเฟรมกับระนาบแกน x เท่าไร เช่น ถ้ามุมในหน่วยของศาระหว่างเฟรมกับระนาบแกน x มีค่าเท่ากับ 270 องศา ก็แทนเฟรมนั้นด้วย 5 ถ้ามุมในหน่วยของศาระหว่างเฟรมกับระนาบแกน x มีค่าเท่ากับ 20 ก็แทนเฟรมนั้นด้วย 3 เป็นต้น

จากนั้นนำค่าจำนวนเต็มที่แทนลงไปในแต่ละเฟรมมาคำนวณ โดยใช้สูตร

Decision = (ค่าจำนวนเต็มของเฟรมก่อนหน้า + 8 – ค่าจำนวนเต็มของเฟรมที่กำลังสนใจ) % 8

ซึ่ง % คือการมอดดูเลชัน (modulation)

- ถ้า Decision มีค่ามากกว่า 4 หมายความว่าเฟรมกำลังโค้งตามเข็มนาฬิกา
- ถ้า Decision มีค่าน้อยกว่าหรือเท่ากับ 4 หมายความว่าเฟรมกำลังโค้งทวนเข็มนาฬิกา
- ถ้า Decision มีค่าเท่ากับ 0 จะดูจากมุมของเฟรม โดยถ้ามุมของเฟรมก่อนหน้ามีค่ามากกว่ามุมของเฟรมที่กำลังสนใจจะพิจารณาว่าเฟรมกำลังโค้งตามเข็มนาฬิกา ตามรูปด้านล่างซ้าย แต่ถ้ามุมของเฟรมก่อนหน้ามีค่าน้อยกว่าหรือเท่ากับมุมของเฟรมที่กำลังสนใจจะพิจารณาว่าเฟรมกำลังโค้งตามเข็มนาฬิกา ตามรูปด้านล่างขวา



รูปที่ 2-14 แสดงการโค้งตามเข็มนาฬิกาและทวนเข็มนาฬิกาของเฟรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่เนื่องจากค่าที่นำมาใช้ต้องมีค่าไม่เกิน 1 เพราะฉะนั้นมุมเริ่มต้นของแต่ละเฟรมที่นำไปใช้จะมีค่าเท่ากับ

$$\text{มุมเริ่มต้นที่นำไปใช้} = \frac{a}{360} \text{ องศา}$$

หมายเหตุ ถ้าเฟรมที่กำลังสนใจเป็นเฟรมที่อยู่ระหว่าง stroke ที่หนึ่งและสองกำหนดให้มุมเริ่มต้นของเฟรมมีค่าเท่ากับ –1

2.4.2.6 มุมสุดท้ายของเฟรม (AngleEnd) : คือมุมสุดท้ายของเฟรมปัจจุบันหรือเฟรมที่กำลังสนใจอยู่ทำกับเฟรมต่อไป โดยถ้าเป็นเฟรมสุดท้ายของ stroke จะกำหนดให้มีค่าเท่ากับ –1 ถ้าไม่ใช้มุมสุดท้ายเของเฟรมสุดท้ายสามารถหาได้ดังนี้ เริ่มต้นโดยนำจุด 3 จุดมาคำนวณ ซึ่งจุดแรกคือจุดเริ่มต้นของเฟรมที่กำลังสนใจอยู่ จุดที่สองคือ จุดสุดท้ายของเฟรมที่กำลังสนใจอยู่ และ จุดที่สามคือ จุดสุดท้ายของเฟรมต่อไป ซึ่งกำหนดให้แทนด้วย P1, P2, และ P3 ตามลำดับ สามารถแสดงได้ดังรูป

รูปที่ 2-15 แสดงมุมสุดท้ายของเฟรม

ซึ่งมุม a หาได้จากสูตร

$$a = \cos^{-1}\left[\frac{\overrightarrow{P1P2} \bullet \overrightarrow{P2P3}}{\left|\overrightarrow{P1P2}\right| * \left|\overrightarrow{P2P3}\right|}\right]$$

โดย $\overrightarrow{P1P2}$ คือ เวกเตอร์มีค่าเท่ากับ $x_1 i+y_1 j$ มีขนาดเท่ากับ $\left|\overrightarrow{P1P2}\right|$ และ $\overrightarrow{P2P3}$ คือเวกเตอร์มีค่าเท่ากับ x2i+y2j มีขนาดเท่ากับ $\left|\overrightarrow{P2P3}\right|$ นำมาแทนในสูตรข้างต้นได้สูตรดังนี้

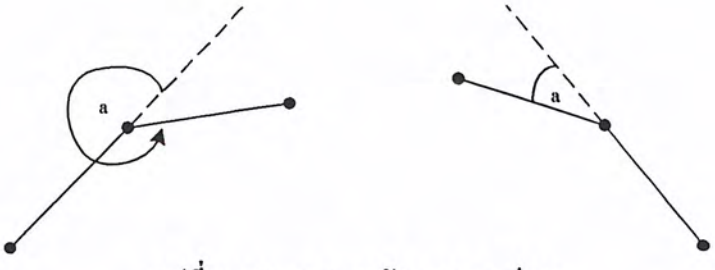
$$a = \cos^{-1}\left[\frac{(x1x2 + y1y2)}{\sqrt{x1^2 + y1^2} * \sqrt{x2^2 + y2^2}}\right]$$

สังเกตได้ว่าขนาดของเวกเตอร์ต้องมีขนาดไม่เท่ากับศูนย์ และมุมที่ได้มีหน่วยเป็นเรเดียน ดังนั้นจึงต้องแปลงเป็นหน่วยองศาโดยใช้สูตร

$$a \text{ (องศา)} = \frac{a_{radian} * 180}{\pi}$$

เนื่องจากมุมที่นำไปใช้จะวัดในลักษณะทวนเข็มนาฬิกาดังรูปซึ่งมีสองลักษณะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

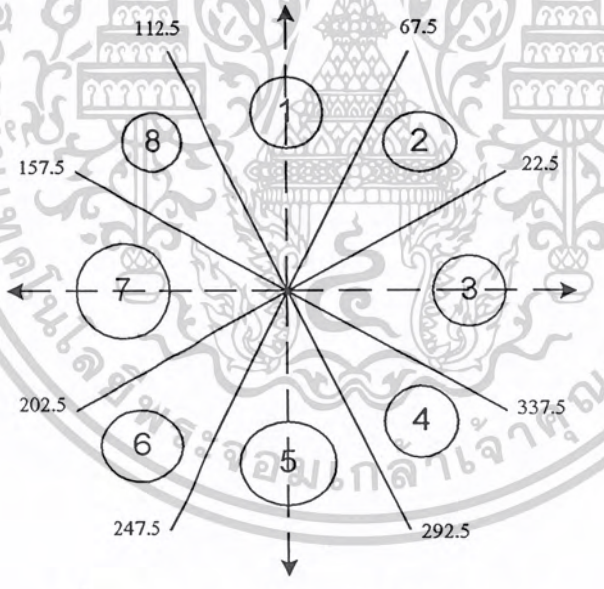


รูปที่ 2-16 แสดงการวัดมุมของเฟรม

เพราะฉะนั้นต้องตรวจสอบลักษณะของเฟรมว่ากำลังโค้งตามเข็มนาฬิกาหรือกำลังโค้งทวนเข็มนาฬิกา

- ถ้ามีลักษณะกำลังโค้งตามเข็มนาฬิกาตามรูปด้านซ้ายมือ ต้องเอามุมที่ได้ลบออกจาก 360
- ถ้ามีลักษณะกำลังโค้งทวนเข็มนาฬิกาตามรูปด้านขวามือ สามารถนำมุมที่หามาได้ไปใช้ได้เลย

การตรวจสอบว่าเฟรมกำลังจะโค้งไปทางใดนั้นจะใช้วิธี Chain Code ช่วยโดยจะแทนเฟรมด้วยเลขจำนวนเต็มตั้งแต่ 1 ถึง 8 โดยจะแทนด้วยค่าอะไรนั้นขึ้นอยู่กับมุมในหน่วยองศาระหว่างเฟรมกับระนาบแกน x ซึ่งวัดทวนเข็มนาฬิกา โดยแบ่งเป็นลักษณะดังนี้



รูปที่ 2-17 แสดงการกำหนดค่าให้กับช่วงองศา

โดยจะแบ่งออกเป็น 8 ส่วน ส่วนละ 45 องศา โดยดูว่าแต่ละเฟรมมีมุมในหน่วยองศาระหว่างเฟรมกับระนาบแกน x เท่าไร เช่น ถ้ามุมในหน่วยองศาระหว่างเฟรมกับระนาบแกน x มีค่าเท่ากับ 270 องศา ก็แทนเฟรมนั้นด้วย 5 ถ้ามุมในหน่วยองศาระหว่างเฟรมกับระนาบแกน x มีค่าเท่ากับ 20 ก็แทนเฟรมนั้นด้วย 3 เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นนำค่าจำนวนเต็มที่แทนลงไปในแต่ละเฟรมมาคำนวณโดยใช้สูตร

Decision = (ค่าจำนวนเต็มของเฟรมที่กำลังสนใจ + 8 – ค่าจำนวนเต็มของเฟรมต่อมา) % 8

ซึ่ง % คือการมอดดูเลชัน (modulation)

- ถ้า Decision มีค่ามากกว่า 4 หมายความว่าเฟรมกำลังโค้งตามเข็มนาฬิกา
- ถ้า Decision มีค่าน้อยกว่าหรือเท่ากับ 4 หมายความว่าเฟรมกำลังโค้งทวนเข็มนาฬิกา
- ถ้า Decision มีค่าเท่ากับ 0 จะดูจากมุมของเฟรม โดยถ้ามุมของเฟรมก่อนหน้ามีค่ามากกว่ามุมของเฟรมที่กำลังสนใจจะพิจารณาว่าเฟรมกำลังโค้งตามเข็มนาฬิกา ตามรูปด้านล่างซ้าย แต่ถ้ามุมของเฟรมก่อนหน้ามีค่าน้อยกว่าหรือเท่ากับมุมของเฟรมที่กำลังสนใจจะพิจารณาว่าเฟรมกำลังโค้งตามเข็มนาฬิกา ตามรูปด้านล่างขวา

รูปที่ 2-18 แสดงการโค้งตามเข็มนาฬิกาและทวนเข็มนาฬิกาของเฟรม

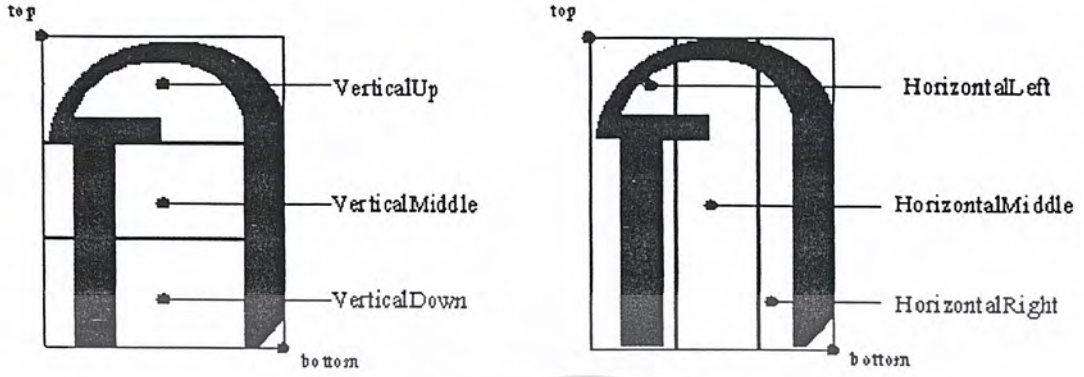
แต่เนื่องจากค่าที่นำมาใช้ต้องมีค่าไม่เกิน 1 เพราะฉะนั้นมุมเริ่มต้นของแต่ละเฟรมที่นำไปใช้จะมีค่าเท่ากับ

$$\text{มุมเริ่มต้นที่นำไปใช้} = \frac{a}{360} \text{ องศา}$$

หมายเหตุ ถ้าเฟรมที่กำลังสนใจเป็นเฟรมที่อยู่ระหว่าง stroke ที่หนึ่งและสองกำหนดให้มุมสุดท้ายของเฟรมมีค่าเท่ากับ –1

2.4.2.7 ลักษณะเด่นของตัวอักษร : แบ่งออกเป็น 6 ลักษณะเด่น ซึ่งเกิดจากการแบ่งพื้นที่ตัวอักษรออกเป็น 6 ส่วน ประกอบด้วย ตามแนวขวาง 3 ส่วน ได้แก่ ขวางบน (VerticalUp) ขวางกลาง (VerticalMiddle) ขวางล่าง (VerticalDown) และ ตามแนวดิ่ง 3 ส่วน ได้แก่ ดิ่งซ้าย (HorizontalLeft) ดิ่งกลาง (HorizontalMiddle) ดิ่งขวา (HorizontalRight) ซึ่งแสดงได้ดังภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-19 แสดงการแบ่งพื้นที่ของตัวอักษร

โดยคู่ลำดับ top ค่า x มีค่าเท่ากับค่า x ที่น้อยที่สุดของคู่ลำดับต่าง ๆ ในตัวอักษร
ค่า y มีค่าเท่ากับค่า y ที่น้อยที่สุดของคู่ลำดับต่าง ๆ ในตัวอักษร

คู่ลำดับ bottom ค่า x มีค่าเท่ากับค่า x ที่มากที่สุดของคู่ลำดับต่าง ๆ ในตัวอักษร
ค่า y มีค่าเท่ากับค่า y ที่มากที่สุดของคู่ลำดับต่าง ๆ ในตัวอักษร

โดย พื้นที่ต่าง ๆ ที่เกิดขึ้นในแนวขวางเกิดจากค่า y ของ bottom ลบออกจาก ค่า y ของ top หารด้วย 3

พื้นที่ต่าง ๆ ที่เกิดขึ้นในแนวดิ่งเกิดจากค่า x ของ bottom ลบออกจาก ค่า x ของ top หารด้วย 3

โดยค่าต่าง ๆ ที่นำไปใช้ ซึ่งได้แก่ ขวางบน ขวางกลาง ขวางล่าง ดิ่งซ้าย ดิ่งกลาง และดิ่งขวา มีค่าเท่ากับจำนวนคู่ลำดับในพื้นที่นั้นหารด้วยจำนวนจุดทั้งหมดซึ่งจะมีค่าไม่เกินหนึ่ง

2.4.3 ตัวอย่างผลเมื่อผ่านขั้นตอนนี้

Frame : 1

LengthOfLine : 0.26278278	Direction : 0.24860393	StartLine : 1.0
EndLine : 0.06666667	HorizontalLeft : 0.5043478	HorizontalMiddle : 0.0
HorizontalRight : 0.0	VerticalUp : 0.0	VerticalMiddle : 0.22608696
VerticalDown : 0.27826086	AngleStart : -1.0	AngleEnd : 0.058496196

Frame : 2

LengthOfLine : 0.019694127	Direction : 0.30710012	StartLine : 0.93333334
EndLine : 0.13333334	HorizontalLeft : 0.5555556	HorizontalMiddle : 0.0
HorizontalRight : 0.0	VerticalUp : 0.0	VerticalMiddle : 0.5555556
VerticalDown : 0.0	AngleStart : 0.058496196	AngleEnd : 0.10688942

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 การใช้โครงข่ายประสาทแบบปรับค่าย้อนกลับในการจดจำตัวอักษร

โครงข่ายประสาท (Neural Network) เป็นความพยายามอีกอย่างของมนุษย์ในการที่จะลอกเลียนแบบธรรมชาติ โดยทั่วไปแล้ว โครงข่ายประสาทจะถูกประกอบขึ้นจากหน่วยย่อยที่มีหน้าที่การทำงานคล้ายคลึงกับเซลล์สมองของมนุษย์ จากนั้นหน่วยย่อยเหล่านี้จะถูกจัดเชื่อมต่อกันเป็นโครงข่ายในลักษณะที่ สัมพันธ์กับกายวิภาคทางสมองหรือบางครั้งอาจจะไม่ก็ได้ และโดยการเชื่อมต่อกันในโครงข่ายนี้ทำให้โครงข่ายประสาทสามารถแสดงพฤติกรรมมากมายที่มีการตอบสนองคล้ายคลึงกับระบบประสาทในสิ่งมีชีวิตยกตัวอย่างเช่นความสามารถในการที่จะเรียนรู้จากประสบการณ์ การจัดกลุ่มของข่าวสารจากตัวอย่างที่ให้ หรือจับลักษณะเด่นของข้อมูลป้อนเข้าที่ประกอบจากข้อมูลที่ไม่สมบูรณ์ เป็นต้น

การศึกษาและสิ่งประดิษฐ์ทางด้านโครงข่ายประสาทได้ถูกนำเสนอออกมาทั้งในรูปของฮาร์ดแวร์และซอฟแวร์ โดยผลิตภัณฑ์ที่เป็นฮาร์ดแวร์มักเรียกว่า Neuralcomputer ซึ่งจะทำการจำลองการทำงานของเซลล์ประสาทลงบนวงจรทางไฟฟ้าหรือทางแสง จากนั้นอาจถูกนำมาใช้ร่วมกันกับคอมพิวเตอร์ธรรมดาได้ในลักษณะของ Co – processor ซึ่งสามารถเรียกใช้เหมือนกับเป็นโปรแกรมย่อยอันหนึ่งในระบบได้ หรือในอีกแนวทางหนึ่งจะอยู่ในรูปของซอฟท์แวร์โดยสร้างโปรแกรมเพื่อจำลองเซลล์ประสาทและจุดเชื่อมต่อของเซลล์ประสาทเหล่านั้นลงบนคอมพิวเตอร์แบบธรรมดา เพื่อทำการศึกษาพฤติกรรมของมัน ตัวอย่างของ Neurocomputer เช่น Neoro-07 ของบริษัท NEC ที่จะประกอบด้วยไมโครคอมพิวเตอร์ Neuro-Engine Board และ Network Learning Software และสำหรับทางด้านซอฟแวร์มีสถิติว่าโปรแกรมทางด้านนิวรอลเน็ตเวิร์คได้ถูกจำหน่ายไปถึงประมาณ 10,000 ชุดในอเมริกาในปี ค.ศ. 1988 เพียงปีเดียว

จุดอ่อนของโครงข่ายประสาทมีอยู่ เช่นเดียวกับสมองของมนุษย์นั่นคือขีดจำกัดในการคำนวณตัวเลขในขณะที่นิวรอลเน็ตเวิร์คมีความชำนาญพิเศษในการประมวลผลข้อมูลที่ไม่สมบูรณ์ การเรียนรู้จากประสบการณ์ ฯลฯ แต่ในข้อมูลปรกติทั่วไปการใช้คอมพิวเตอร์ธรรมดาจะเหมาะสมกว่าดังนั้นการจะนำโครงข่ายประสาท หรือ Neurocomputer ไปทำการคำนวณบัญชีเงินเดือน บัญชีลูกหนี้ ฯลฯ จึงไม่อยู่ในวิสัยที่ควรจะทำหรือกล่าวอีกอย่างหนึ่งก็คือ แม้วิชาการทางโครงข่ายประสาทจะถูกพัฒนาก้าวหน้าไปไกลเท่าใดก็ตามก็ไม่สามารถเข้ามาแทนที่คอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปแล้วได้ หากแต่จะเข้ามาช่วยเหลือในงานส่วนที่คอมพิวเตอร์ไม่สะดวกเท่านั้น และในทำนองเดียวกันถ้าหากเราพิจารณาความสามารถพิเศษของโครงข่ายประสาท ดังเช่น ความสามารถในการเรียนรู้จากประสบการณ์หรืออีกหลาย ๆ อย่าง จะเห็นว่าความสามารถเหล่านั้นสามารถทำได้จากวิชาการทางด้านปัญญาประดิษฐ์ (AI) เช่นกัน โดยเฉพาะทางด้านระบบผู้เชี่ยวชาญ (Expert System) ซึ่งประสบความสำเร็จอย่างมากในหลายที่ แต่ก็ประสบกับปัญหาความยุ่งยากในการสร้างความสามารถสำหรับการเรียนรู้จากประสบการณ์ไม่น้อย จนหลายคนอาจมองว่าโครงข่ายประสาท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

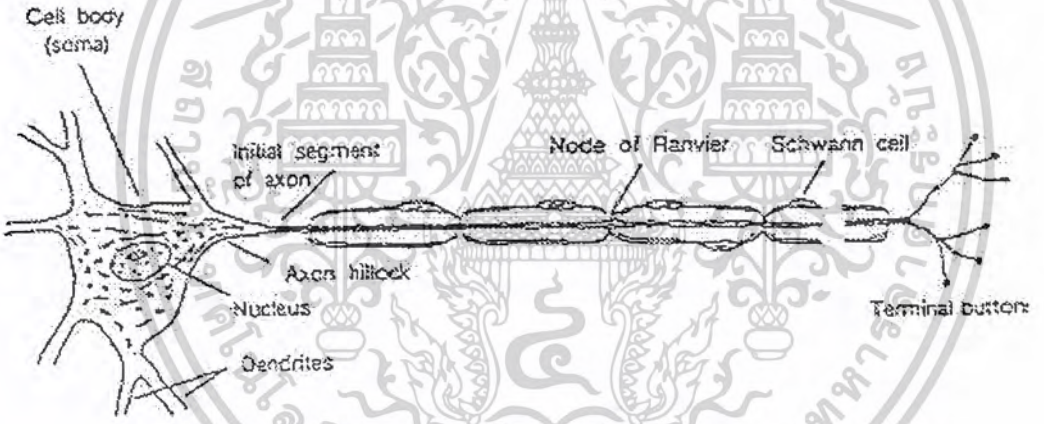
อาจจะเข้ามาแทนที่ AI แต่ที่จริงแล้วมีสิ่งบอกเหตุหลายอย่างที่ชี้ว่าเทคนิคทั้งสองแบบน่าจะรวมกันได้เป็นระบบโดยที่แต่ละตัวทำงานในส่วนที่ตัวเองถนัดมากกว่า

ความแตกต่างระหว่างคอมพิวเตอร์ธรรมดากับโครงข่ายประสาทอีกอย่างหนึ่งคือ ในคอมพิวเตอร์ธรรมดาเมื่อต้องการให้ทำงานอย่างใดสามารถทำได้โดยการสร้างโปรแกรมแต่ในทางตรงข้ามโครงข่ายประสาทเมื่อต้องการให้ทำงานอย่างใดอย่างหนึ่งสามารถทำได้โดยการสอน (Training) โดยการฝึกสอนนี้จะทำให้โครงข่ายประสาทนำบทเรียนที่ได้จากการฝึกเข้าไปเก็บในโครงร่างประสาท และจากนั้นจะถูกเรียกกลับขึ้นมาใช้เมื่ออยู่ระหว่างการประมวลผลข้อมูล

เนื่องจากโครงข่ายประสาทเป็นความพยายามที่จะเลียนแบบการทำงานของธรรมชาติดังได้กล่าวมาแล้วนั้น จึงจะขอพูดถึงเซลล์ประสาทในธรรมชาติไว้ในที่นี้พอสังเขป

2.5.1 โครงสร้างในเซลล์ประสาทของมนุษย์

ระบบประสาทของมนุษย์ประกอบด้วยเซลล์ประสาทประมาณ 10 พันล้านตัว เซลล์ประสาทแต่ละตัวประกอบด้วย เซลล์ประสาท (Cell Body) และแขนงที่แยกออกจากตัวเซลล์ประสาท ซึ่งได้แก่ แอกซอน (Axon) และเดนไดรท์ (Dendrite) ดังแสดงในรูปข้างล่าง



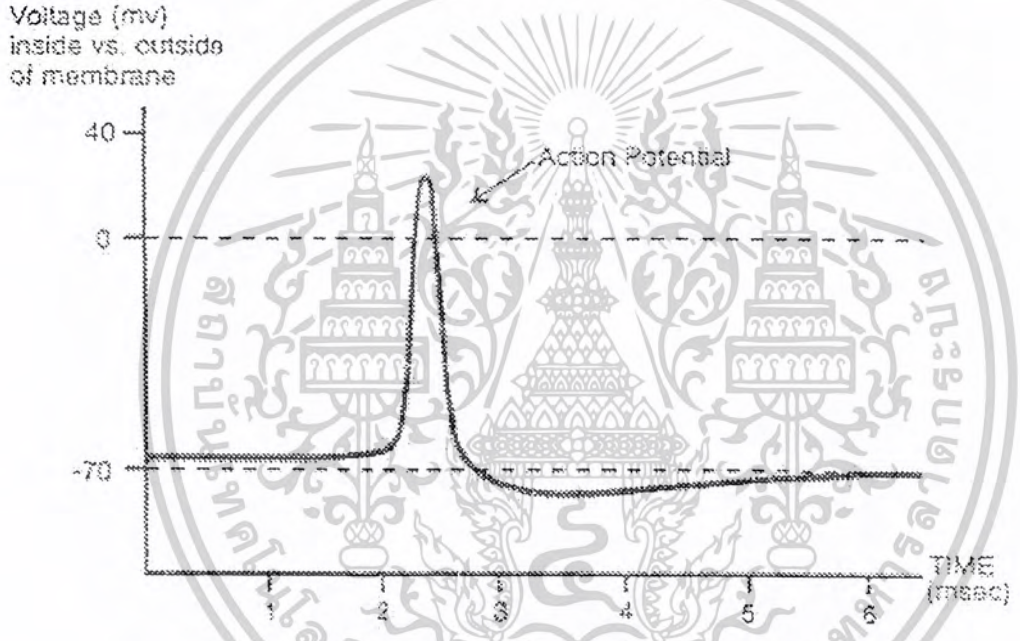
รูปที่ 2-20 แสดงเซลประสาททางชีวภาพ

ตัวเซลล์ประสาทจะประกอบด้วยนิวเคลียสและซัยโตปลาสซึม ซึ่งมีออร์แกเนลมากมาย เช่น ไมโตคอนเดรีย กอลจิคอมเพล็กซ์ ร่างแหเอนโดพลาสมิค เออร์แกสโตปลาสซึมและไรโบโซม สำหรับ แอกซอน (Axon) นั้น มีจุดที่เรียกว่า Axon Hilloc ซึ่งเป็นส่วนที่หนาที่สุดของเซลล์ประสาท โดยเซลล์ประสาทหนึ่งตัวจะมีแอกซอนเพียงหนึ่งอันที่จะทอดตัวยาวออกไปยังอวัยวะแสดงผลเช่นกล้ามเนื้อหรือไปยังเซลล์ประสาทตัวอื่น ๆ ที่ทำงานร่วมกัน ทางด้านของเดนไดรท์ (Dendrite) จะเป็นแขนงสั้น ๆ ออกจากเซลล์ประสาทก็จะแยกแขนงออกไปทันที ถ้าดูตามหน้าที่เดนไดรท์และตัวเซลล์ประสาทจะมีบริเวณของตัวรับ (Receptor Zone) ซึ่งจะรับข้อมูลที่ส่งมาจากปลายแอกซอนของเซลล์ประสาทตัวอื่น ข้อมูลเหล่านี้ซึ่งอยู่ในรูปแบบของการเปลี่ยนแปลงทางไฟฟ้าเคมีจะกระตุ้นเซลล์ประสาทเพื่อให้ส่งสัญญาณออกจากเซลล์ไปสู่เซลล์ประสาทตัวอื่น หรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อวัยวะสำแดงผล หรือกล่าวโดยสรุปก็คือ ในเซลล์ประสาทนั้นหน้าที่ของแอกซอนคือ ผลลัพธ์ที่จะนำสัญญาณหรือกระแสประสาทออกจากเซลล์ประสาทตัวอื่นหรืออวัยวะสำแดงผลและเดนไดรท์เป็นตัวป้อนข้อมูลหรือข่าวสารเข้าสู่เซลล์นั่นเอง

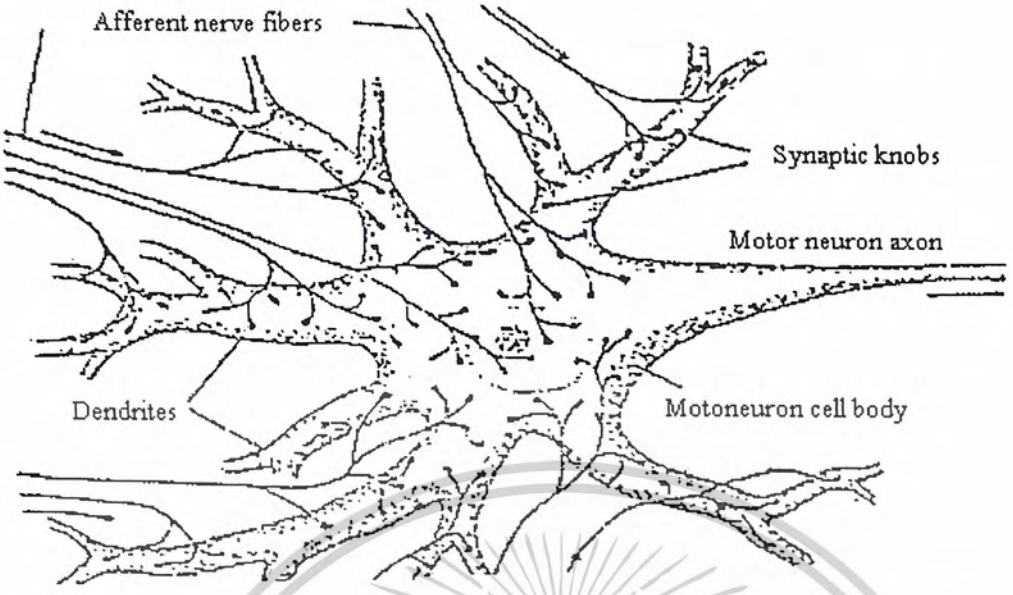
เซลล์ประสาททั่วไปจะทำงานโดยการถูกกระตุ้น การกระตุ้นนี้จะเกิดขึ้นเมื่อการเปลี่ยนแปลงสมดุลย์ของอิออนที่บริเวณตัวรับ (Receptor Zone) มากจนถึงระดับตัดสิน (Threshold) ความต่างศักย์ของเยื่อหุ้มเซลล์ขณะปรกติจะมีค่าประมาณ –60 ถึง -90 มิลลิโวลต์ จากนั้นเมื่อมีการเปลี่ยนแปลงความต่างศักย์ของเยื่อหุ้มเซลล์จนถึงระดับตัดสิน ตัวเซลล์ประสาทก็จะส่งกระแสประสาทออกไปทางแอกซอน (Axon) โดยลักษณะการเปลี่ยนแปลงความต่างศักย์ของเยื่อหุ้มเซลล์ได้แสดงไว้ในรูปข้างล่าง



รูปที่ 2-21 แสดงการตอบสนองทางไฟฟ้าของเยื่อหุ้มเซล

โครงสร้างที่น่าสนใจอีกส่วนหนึ่งของเซลล์ประสาทก็คือส่วนที่เรียกว่า ไซแนปส์ (Synapse) ซึ่งไซแนปส์นี้คือจุดระหว่างของแอกซอนของเซลล์ประสาทหนึ่ง กับเดนไดรท์ของเซลล์ประสาทอีกตัวโดยผ่านทางรอยต่อที่เรียกว่า ไซแนปส์ (Sunapse) ภาพของจุดที่เชื่อมต่อระหว่างเซลล์ประสาทและภาพขยายของไซแนปส์ได้ถูกแสดงไว้ในรูปข้างล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

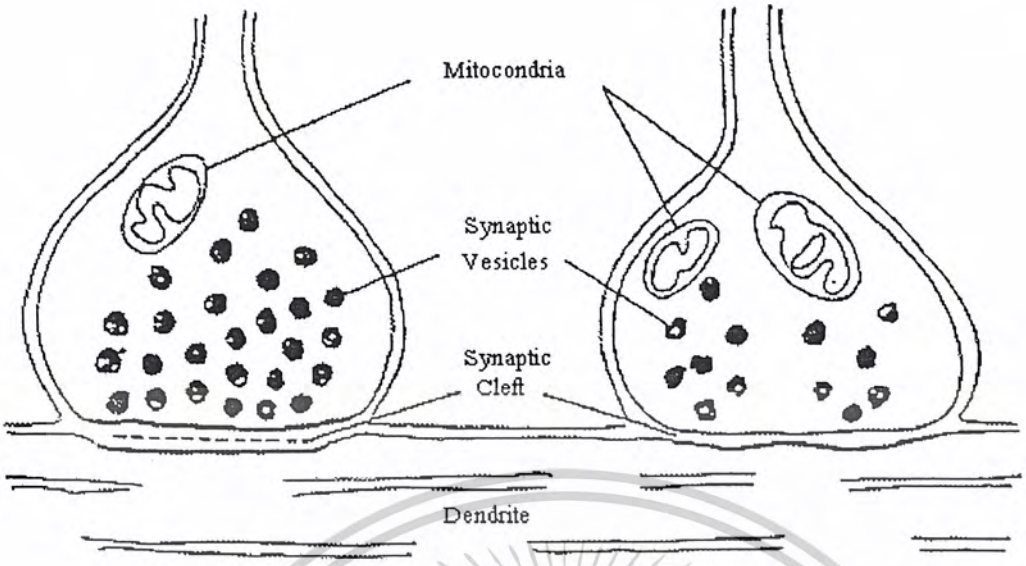


รูปที่ 2-22 แสดงไซแนปส์หรือจุดที่เชื่อมต่อของเซลประสาท

จากรูป จะเห็นว่ามีโครงสร้างที่เรียกว่า Synapse Vesicles ซึ่งภายใน Synapse Vesicles นี้ ประกอบด้วยสารเคมีที่ใช้ในการติดต่อระหว่างเซลล์หรือที่เรียกว่าสารสื่อประสาท (Chemical Transmitter) โดยสารนี้จะถูกหลั่งออกสู่ Synapse Cleft เมื่อมีศักย์แพร่เข้ามาที่ไซแนปส์ ลักษณะของสารสื่อประสาทแบ่งตามการทำงานได้ 2 ประเภทคือ

1. Excitstory Transmitters เป็นสารสื่อประสาทที่เมื่อหลั่งออกมาจะไปกระตุ้นให้อีกเซลล์หนึ่งทำงาน
2. Inhibitory Transmitters เป็นสารสื่อประสาทที่เมื่อหลั่งออกมาแล้วจะไปยับยั้งการทำงานของอีกเซลล์หนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-23 แสดงภาพขยายของไซแนปส์

การทำงานโดยใช้สารสื่อประสาทนี้ทำให้ไซแนปส์สามารถทำหน้าที่สำคัญให้กับเซลล์ประสาทได้ดังนี้คือ

1. ควบคุมให้ข้อมูลหรือกระแสประสาทเดินไปในทิศทางเดียวเพื่อไม่ให้เกิดความสับสนของกระแสประสาทเกิดขึ้น
2. รวบรวมสัญญาณกระแสประสาทจากในหลาย ๆ ไซแนปส์ในลักษณะที่เป็นกลไกของ Summation ตามลักษณะของสารสื่อประสาทที่เป็น Excitory หรือ Inhibitory
3. ขยายสัญญาณของข้อมูลกระแสประสาทเพื่อให้สามารถส่งไปได้กว้างขวางและแรงขึ้น

2.5.2 ทฤษฎีพื้นฐานของโครงข่ายประสาท

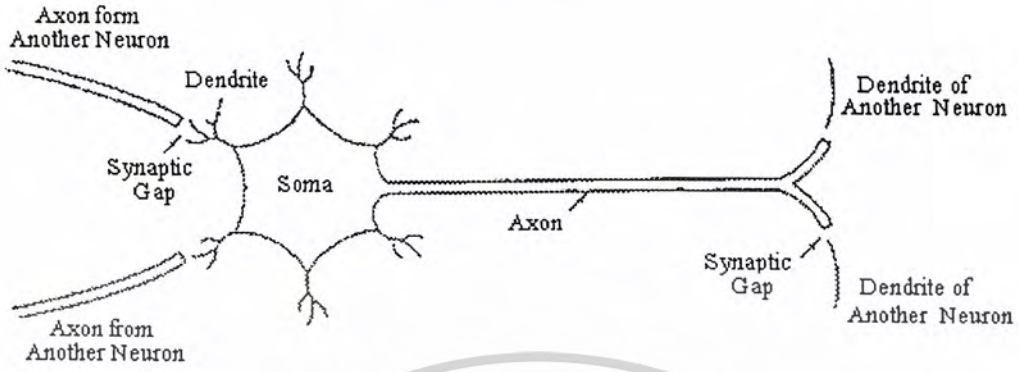
โครงข่ายประสาทเป็นแนวความคิดหนึ่งทาง AI (Artificial Intellingence) ในการพยายามสร้างเครื่องจักร ที่สามารถเรียนรู้และจดจำ ซึ่งมีกระบวนการคล้ายคลึงกับกระบวนการทางสมองของมนุษย์

2.5.2.1 ลักษณะของโครงข่ายประสาท

โครงข่ายประสาทมีลักษณะเป็นการจำลองข้อมูลทางชีววิทยา ทางโครงสร้างสมองของมนุษย์ โดยการจำลองเอามาบางส่วนที่จำเป็นมาใช้ผสมกับรูปแบบการจัดโครงสร้าง ที่ถูกคิดขึ้นเพื่อทำให้เกิดฟังก์ชันที่ต้องการ โดยทั่วไปแล้วลักษณะโครงสร้างของโครงข่ายประสาทมีอยู่หลายรูปแบบ แต่ลักษณะโครงสร้างพื้นฐานจะเหมือนกัน กล่าวคือ ลักษณะทางโครงสร้างพื้นฐานจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

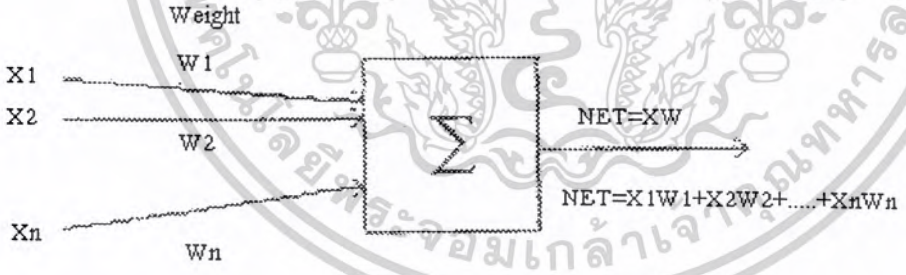
ประกอบไปด้วยนิวรอน (Neuron) ซึ่งเป็นเซลล์ทางสมองของมนุษย์ ซึ่งโครงสร้างของนิวรอนจำลองแบบมาจากลักษณะโดยทั่วไปของเซลล์ประสาท (Neural Cell) มีลักษณะดังรูป



รูปที่ 2-24 แสดงลักษณะของนิวรอนเซลของเซลประสาท

กล่าวคือจะประกอบด้วยส่วนของเดน ไดรท์ (Dendrites) ซึ่งยื่นออกมาจากเซลล์ร่างกาย ไปเชื่อมต่อยังนิวรอนเซลล์อื่น ๆ โดยที่จุดเชื่อมต่อเหล่านี้ จะมีส่วนของไซแนปส์ (Synapse) ซึ่งสามารถปรับตัวได้เมื่อ input ได้ถูกส่งผ่านไซแนปส์มาแล้วไปรวมกับเซลล์ร่างกาย และนิวรอนเซลล์จะส่ง output ออกทางเซลล์แอกซอน (axon) ตามลักษณะการรวมตัวกันของ input ที่เซลล์ร่างกาย

สำหรับลักษณะของนิวรอนของโครงข่ายประสาทจะมีลักษณะดังรูป



รูปที่ 2-25 แสดงนิวรอนของโครงข่ายประสาท

จากรูปจะมีลักษณะคล้ายกับเซลล์ประสาท กล่าวคือ ส่วนของข้อมูลป้อนเข้า ที่เข้ามาจะมีค่าน้ำหนัก (Weight) ซึ่งเปรียบเสมือนส่วนของไซแนปส์ ซึ่งสามารถปรับค่าได้ ผ่านเข้ามาในส่วนที่ทำหน้าที่รวมข้อมูลป้อนเข้าที่เข้ามาและออกเป็นค่าผลลัพธ์ ค่าหนึ่งสำหรับค่าผลลัพธ์ที่ออกมาสามารถคำนวณได้ดังนี้

$$NET = X1W1 + X2W2 + \ldots\ldots\ldots + X3W3$$

หรืออาจเขียนเป็นรูปแบบของเวคเตอร์ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

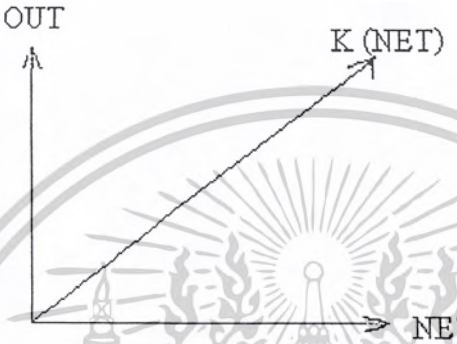
NET = XW

โดย X = เวคเตอร์ของข้อมูลป้อนเข้า

W = เวคเตอร์ค่าน้ำหนัก

NET = เวคเตอร์ผลลัพธ์

สำหรับโดยทั่วไปแล้ว ส่วนผลลัพธ์ที่ออกมาจะถูกกระทำต่อไป โดยส่วนที่เรียกว่า ฟังก์ชันการกระตุ้น (Activation function) ซึ่งมีอยู่หลายรูปแบบซึ่งอาจจะมีลักษณะเป็นสมการเชิงเส้นอย่างง่าย ๆ

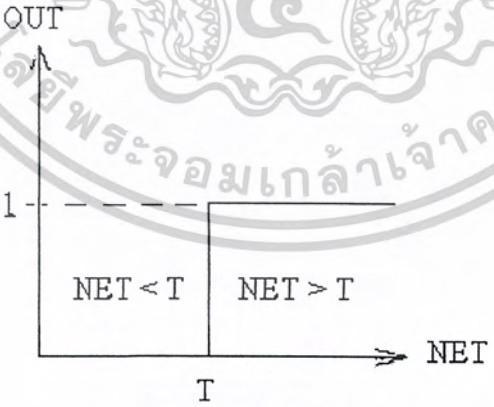


OUT = K(NET)

ซึ่ง K = ค่าคงที่

รูปที่ 2-26 แสดงสมการเชิงเส้นอย่างง่าย ๆ

หรือในรูปของฟังก์ชันระดับตัดสิน



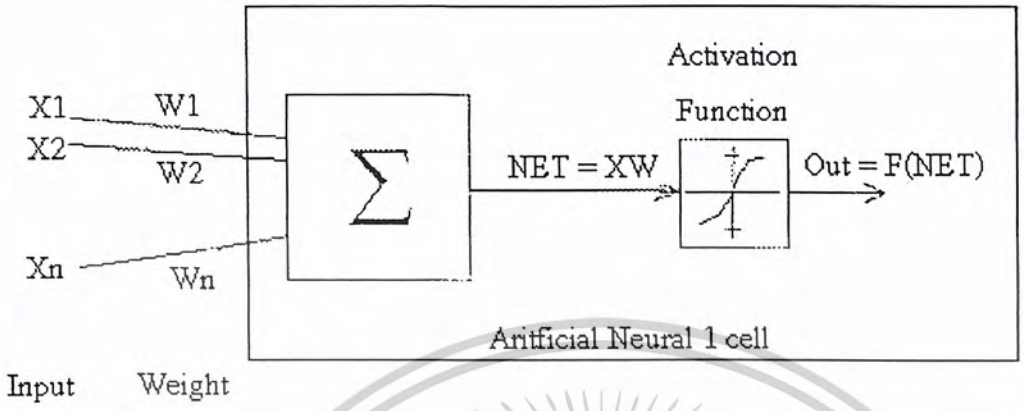
OUT = 1 ถ้า NET > T

OUT = 0 กรณีอื่น

รูปที่ 2-27 แสดงสมการระดับตัดสิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรืออาจอยู่ในรูปฟังก์ชันที่มีลักษณะที่ไม่เป็นเชิงเส้น (Nonlinear function) ตามรูปข้างบน ที่ได้แสดงลักษณะของนิวรอนที่ได้รวมเอาฟังก์ชันการกระตุ้น ตามรูป

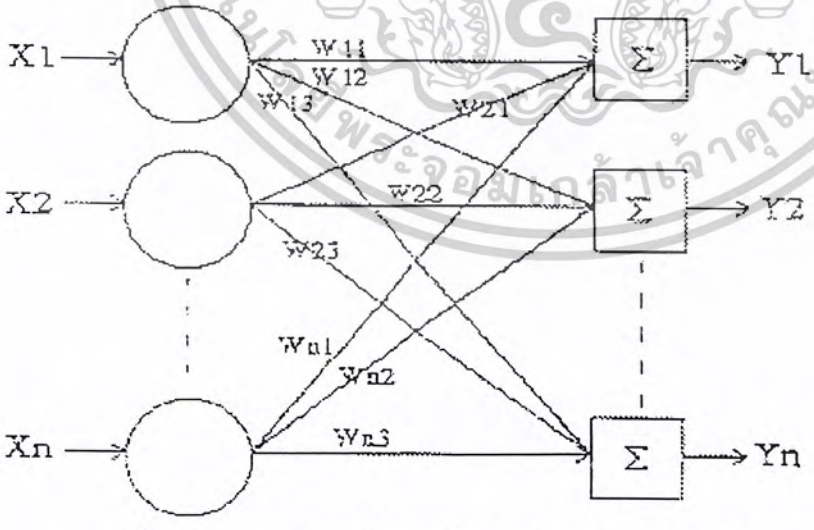


รูปที่ 2-28 แสดงลักษณะของนิวรอน 1 หน่วย ที่ประกอบด้วยฟังก์ชันการกระตุ้น

2.5.2.2 โครงข่ายประสาทในลักษณะ 1 ชั้น (Single Layer Neural Network)

ถึงแม้ว่าลักษณะของนิวรอนเพียง 1 นิวรอนจะสามารถทำฟังก์ชันบางอย่างได้ แต่ประสิทธิภาพที่เพิ่มขึ้นจากการทำงานมาจากการรวมนิวรอนหลาย ๆ นิวรอนเข้าด้วยกัน

ลักษณะรูปแบบอย่างง่ายที่สุดของโครงข่ายประสาท จะจัดกลุ่มของนิวรอนมีลักษณะ 1 ชั้น ดังที่แสดงดังรูป



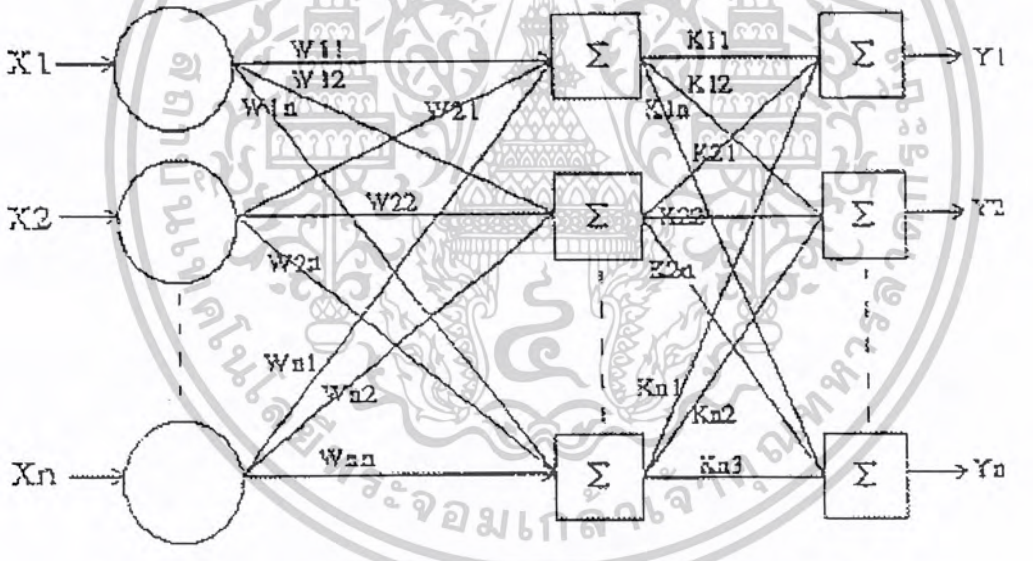
รูปที่ 2-29 แสดงลักษณะโครงสร้างของโครงข่ายประสาทลักษณะ 1 ชั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับในส่วนของวงกลมจะเป็นส่วนที่แบ่งกระจายข้อมูลป้อนเข้า ไปยังนิวรอนต่าง ๆ สำหรับลักษณะโครงสร้างในรูป อาจจะมีการเชื่อมโยงในบางส่วนที่ถูกตัดทิ้งไปหรืออาจจะมีการป้อนกลับจากผลลัพธ์ ไปยังข้อมูลป้อนเข้าได้ เรียกว่ามีลักษณะย้อนกลับ (Backward Learning) เพื่อส่งสัญญาณผิดพลาดกลับไปปรับแต่งค่าน้ำหนักให้ดีขึ้น เพื่อลดการผิดพลาดของของสัญญาณที่เกิดขึ้นในการเรียนรู้กลุ่มข้อมูลป้อนเข้าชุดหนึ่ง ๆ โดยโครงสร้างอย่างง่ายนี้ สามารถฝึกสอนกลุ่มข้อมูลป้อนเข้าที่มีลักษณะอย่างง่าย ๆ ได้ แต่ถ้าต้องการนำไปประยุกต์ใช้กับการป้อนเข้าที่มีรูปร่างยาก ๆ หรือคล้าย ๆ กัน ต้องเพิ่มชั้นของโครงข่ายประสาทอีกเพื่อขยายค่าน้ำหนักให้มากขึ้นในการเรียนรู้

2.5.2.3 โครงข่ายประสาทในกรณีที่มีลักษณะหลาย ๆ ชั้น (Multi Layer Neural Network)

ลักษณะของโครงข่ายประสาทในลักษณะเพียง 1 ชั้น (Single Layer Neural Network) ถึงแม้ว่าจะทำงานได้ดีขึ้นจากนิวรอน เพียง 1 นิวรอน แต่ประสิทธิภาพในการทำงานจะเพิ่มมากขึ้น ถ้าเราจัดรูปแบบโครงสร้างในลักษณะหลาย ๆ ชั้น ในรูปข้างล่าง แสดงลักษณะโครงข่ายประสาทในลักษณะ 2 ชั้น (Two Layer Neural Network)



รูปที่ 2-30 แสดงลักษณะโครงสร้างของโครงข่ายประสาทลักษณะ 2 ชั้น

สำหรับประสิทธิภาพในการทำงานของโครงข่ายประสาทในลักษณะหลาย ๆ ชั้นนี้ จะไม่เพิ่มขึ้นหรือไม่แตกต่างจากโครงข่ายประสาทในลักษณะ 1 ชั้นเลย ถ้าในส่วนของฟังก์ชัน การกระตุ้นมีลักษณะของสมการเชิงเส้น ซึ่งสามารถแสดงได้ดังนี้

ถ้า X เป็น input vector

W1 เป็นค่าน้ำหนักในรูปเมตริกซ์ในชั้นที่ 1

W2 เป็นค่าน้ำหนักในรูปเมตริกซ์ในชั้นที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพราะฉะนั้นค่าผลลัพธ์ที่ออกมา อยู่ในรูป

Output = (XW1)W2

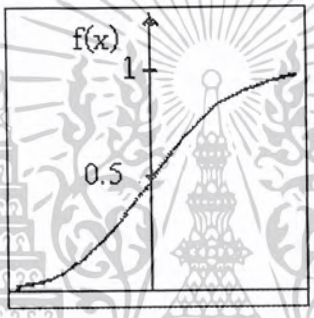
ซึ่งจะจัดได้ใหม่จากรูปแบบการคูณ ดังนี้

Output = X(W1W2)

ซึ่งแสดงว่าลักษณะของโครงข่ายประสาทในลักษณะ 2 ชั้น จะเหมือนกับลักษณะ 1 ชั้น คือมีค่าเมตริกซ์ ของค่าน้ำหนัก เท่ากับผลคูณของเมตริกซ์ของค่าน้ำหนักในชั้นที่ 1 คูณกับชั้นที่ 2

ดังนั้นในการเพิ่มประสิทธิภาพในการทำงานของโครงข่ายนิวรอนในลักษณะหลาย ๆ ชั้น เพิ่มจากโครงข่ายประสาทในลักษณะ 1 ชั้น จึงควรกำหนดฟังก์ชันการกระตุ้นในลักษณะไม่เชิงเส้น โดยมีสมการของฟังก์ชันการกระตุ้นดังนี้

$$f(t) = 1/(1+\exp(-x))$$



รูปที่ 2-31 แสดงสมการของฟังก์ชันการกระตุ้น

2.5.3 การฝึกสอน (Training) ของโครงข่ายประสาท

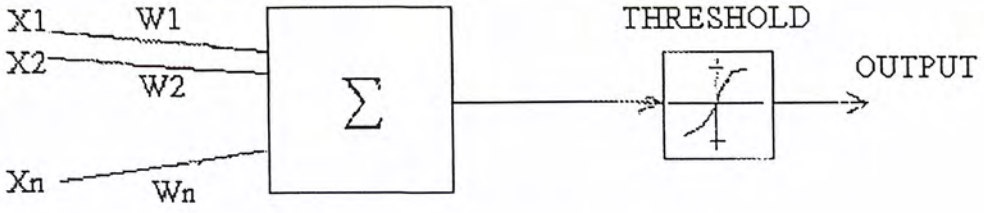
การฝึกสอนเป็นลักษณะที่จะพยายามสอนให้โครงข่ายนิวรอนมีความรู้ตามที่เราต้องการ โดยในลักษณะ การฝึกสอนของโครงข่ายประสาทจะเป็นการป้อนชุดข้อมูลให้แก่โครงข่ายประสาท และพยายามปรับค่าของน้ำหนักไปเรื่อย ๆ เพื่อให้มีค่าลู่เข้าสู่ค่าของชุดผลลัพธ์ที่เราต้องการให้มีค่าออกมา

โดยทั่วไปแล้วเราจะเรียกวิธีการสอนเพื่อให้โครงข่ายสามารถเรียนรู้ เพื่อให้ได้ค่าผลลัพธ์ที่เราต้องการว่ากระบวนการในการฝึกสอน (Training Algorithm)

2.5.3.1 เพอเซปตรอน (Perceptron)

ลักษณะของนิวรอนในยุคแรก ๆ ของโครงข่ายประสาท มีชื่อเรียกว่า เพอเซปตรอน ซึ่งมีลักษณะดังรูปข้างล่าง

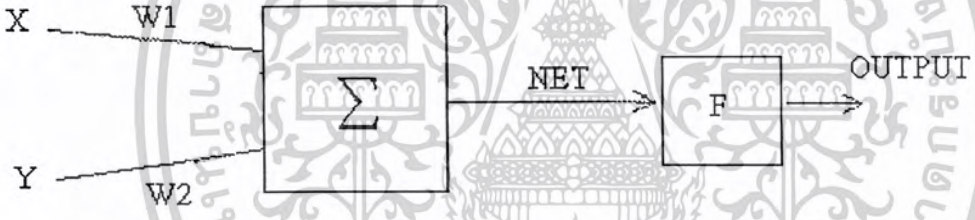
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-32 แสดงลักษณะของเพอเซปตรอน

ลักษณะโดยทั่วไปจะเหมือนกับนิวรอนที่ได้กล่าวมาแล้ว แต่มีลักษณะเฉพาะคือ ในส่วนของฟังก์ชันการกระตุ้น จะเป็นฟังก์ชันระดับตัดสินซึ่งให้ค่า 1 ออกมา ถ้ารวมค่าที่เข้ามามีค่ามากเกินกว่าค่าที่กำหนด จะให้ค่า 0 ออกมา ถ้าผลรวมมีค่าน้อยกว่าค่าที่กำหนด

สำหรับการฝึกสอน จะกระทำในลักษณะเพียง 1 ชั้น ซึ่งประสบผลสำเร็จมากในช่วงแรก ๆ แต่ต่อมาได้ประสบปัญหาบางอย่างซึ่งลักษณะของเพอเซปตรอนเพียง 1 ชั้น ไม่สามารถแก้ปัญหาได้ เช่นปัญหาเกี่ยวกับ เอกคลูซีฟออร์



รูปที่ 2-33 แสดงเพอเซปตรอนที่ใช้แก้ปัญหา

ถ้า X และ Y เป็นข้อมูลป้อนเข้าที่จะทำการ XOR เพราะฉะนั้นจะมีค่า NET ที่ออกมาเท่ากับ

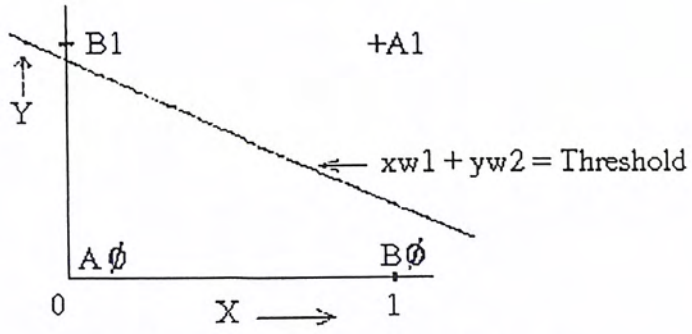
$$NET = XW1 + YW2$$

ถ้าเรากำหนดค่าระดับตัดสินของฟังก์ชัน F ให้มีค่า ๆ หนึ่ง และกำหนดค่า W1 และ W2 เป็นค่า ๆ หนึ่งแล้วจะได้สมการเป็น

$$Threshold = XW1 + YW2$$

ถ้าเราทำการพล็อตกราฟในแกนพิกัด X-Y จะได้กราฟเป็นเส้นตรงดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-34 แสดงกราฟความสัมพันธ์ X-Y

พิจารณาตาราง XOR ในรูปจะได้

Point	Xvalue	Yvalue	Desired Output
A0	0	0	0
B0	1	0	1
B1	0	1	1
A1	1	1	0

ตารางที่ 2-1 ตารางแสดง XOR

2.5.3.2 โครงข่ายประสาทแบบปรับค่าย้อนกลับ

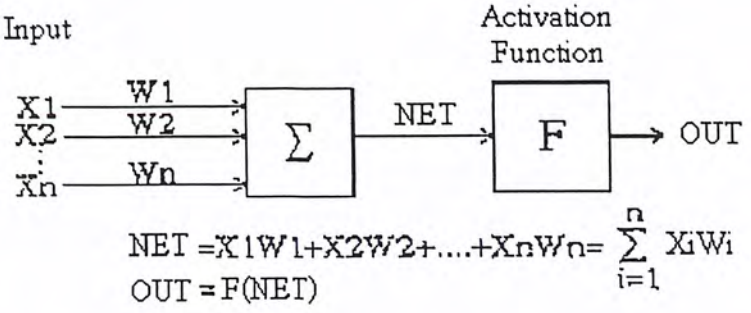
หลังจากประสบปัญหาเกี่ยวกับการฝึกสอนให้โครงข่ายประสาทแบบหลาย ๆ ชั้นมาเป็นเวลานาน ต่อมาได้มีการคิดกระบวนการขึ้นมาเพื่อทำการฝึกสอนให้แก่โครงข่ายประสาทในลักษณะหลายๆชั้น โดยเรียกวิธีการสอนนี้ว่า โครงข่ายประสาทแบบปรับค่าย้อนกลับ

2.5.3.2.1 ลักษณะโครงสร้างของโครงข่าย

1. นิวรอน

ลักษณะโครงสร้างของโครงข่ายแบบปรับค่าย้อนกลับ มีลักษณะดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-35 แสดงลักษณะนิวรอนของโครงข่ายแบบปรับค่าย้อนกลับ

และลักษณะทั่ว ๆ ไป จะคล้ายกับลักษณะทั่ว ๆ ไปของนิวรอนที่ได้กล่าวมาแล้ว แต่มีส่วนที่ต่างกันคือ จะมีการเพิ่มฟังก์ชันการกระตุ้นที่มีลักษณะไม่เป็นเชิงเส้น ซึ่งสมการที่นิยมใช้กับฟังก์ชันการกระตุ้นคือ

$$OUT = F(NET) = 1/(1 + \exp(-NET))$$

จากสมการอธิบายได้ว่า ถ้า NET = 0 แล้ว OUT = 0.5

NET >> 0 แล้ว OUT = 1

NET << 0 แล้ว OUT = 0

ซึ่งสามารถแสดงกราฟได้ดังรูป

ถ้าพิจารณาสมการที่ให้ค่า NET ออกมาใหม่อีกทีจะเห็นว่าค่า NET ที่มีมากกว่าระดับตัดสินจะมีค่าอยู่ทางซีกใดซีกหนึ่งของเส้นตรง ส่วนค่า NET ที่มีค่าน้อยกว่าจะอยู่อีกซีกหนึ่ง

ปัญหาของ XOR นี้จะแก้ได้โดยการใช้ลักษณะของการใช้เพอเซปตรอนหลาย ๆ ชั้น แต่วิธีการที่ให้ได้มาซึ่งค่าของน้ำหนักของโครงข่ายเป็นลักษณะของการกำหนดค่าโดยใช้ความคิดขึ้นมาเอง ไม่ใช่ลักษณะของการฝึกสอน ดังนั้นปัญหาจึงเกิดจากการขาดการฝึกสอนตัวโครงข่ายหลาย ๆ ชั้น

กฎของเดลตา

กฎของเดลตานี้เป็นกฎสำคัญของการฝึกสอนเพอเซปตรอน แต่ต่อมาขยายการฝึกให้กับโครงข่ายประสาทที่มีลักษณะหลายชั้น โดยกฎนี้จะถูกใช้ในการปรับน้ำหนักของเพอเซปตรอนได้อย่าสมเหตุผล โดยมีลักษณะดังนี้

$$\delta = (T - A)$$

โดยที่ T เป็นค่า output ที่ต้องการจากเพอเซปตรอน

A เป็นค่า output ที่ได้ออกมาจริง ตามการป้อน

δ เป็นค่าผลต่างของ 2 ค่าบน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากได้ค่าของเดลตาแล้ว ค่านี้จะถูกนำไปคูณกับค่าข้อมูลป้อนเข้าแต่ละค่าข้อมูล เพื่อนำไปใช้ในการปรับค่าน้ำหนักให้เหมาะสม นอกจากนั้นยังต้องถูกนำไปคูณกับค่าอัตราการเรียนรู้เพื่อเพิ่มประสิทธิภาพในการฝึกสอนให้เร็วขึ้นหรือช้าลง ซึ่งเขียนเป็นสูตรได้ดังนี้

$$\Delta i = n\delta Xi$$

ซึ่งค่า Δi หมายถึงค่าที่ต้องถูกนำมาใช้ในการปรับน้ำหนักของแต่ละค่าน้ำหนัก

Xi คือค่าเวกเตอร์ของข้อมูลป้อนเข้า

n คือค่าอัตราการเรียนรู้

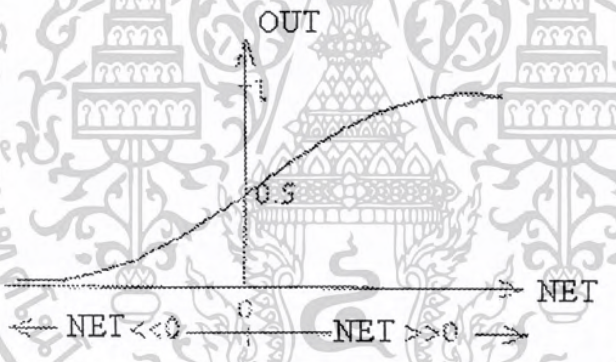
ดังนั้นสูตรการปรับค่าน้ำหนักสามารถเขียนได้ดังนี้

$$wi(n+1) = wi(n) + \Delta i$$

โดย Δi เป็นค่าน้ำหนักที่ต้องปรับของแต่ละข้อมูลป้อนเข้า

wi(n+1) เป็นค่าน้ำหนักหลังการปรับ

wi(n) เป็นค่าน้ำหนักก่อนการปรับ



รูปที่ 2-36 แสดงกราฟของฟังก์ชันการกระตุ้น

ซึ่งฟังก์ชันนี้เรียกว่า Sigmoid Function เนื่องมาจากเส้นโค้งของกราฟมีลักษณะคล้ายรูปตัว S ซึ่งค่า Derivative ของฟังก์ชันมีค่าดังนี้

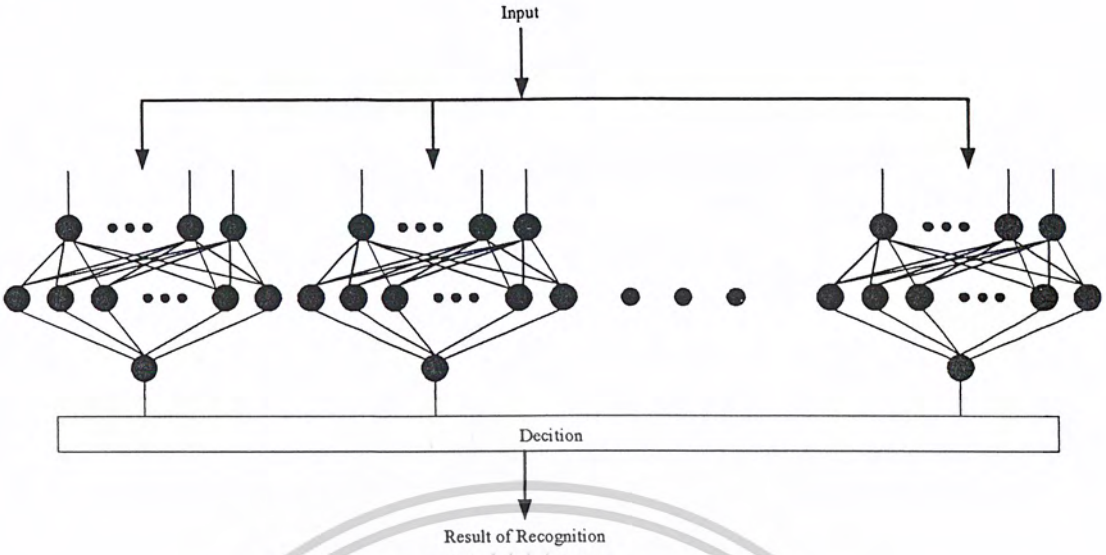
$$DOUT/dNET = OUT(1-OUT)$$

ซึ่งค่านี้จะใช้ในการปรับแต่งน้ำหนักโดยจะมีผลในการทำให้เกิดการลู่เข้าของค่าผลลัพธ์ได้เป็นอย่างดี

2. ลักษณะโครงข่ายประสาทหลาย ๆ ชั้น

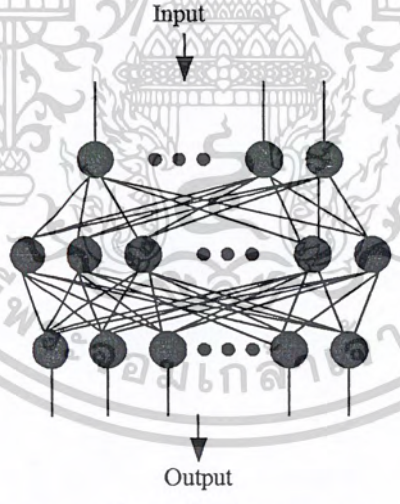
ลักษณะโครงข่ายที่เลือกใช้ คือ OCON (One Class One Network) ซึ่งจะประกอบไปด้วยหลาย ๆ โครงข่ายแต่ละโครงข่ายจะให้ผลลัพธ์เพียง 1 ผลลัพธ์ เท่านั้น แสดงได้ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.37 แสดงโครงข่ายประสาทแบบ OCON

เหตุที่เลือกลักษณะโครงข่ายแบบนี้เนื่องจากลักษณะแบบนี้จะทำให้ การรู้จำเร็วขึ้นและง่ายขึ้น โดยสามารถเปรียบเทียบกับลักษณะโครงข่ายแบบ ACON (All Class One Network) ซึ่งจะใช้เพียงโครงข่ายเดียวที่มีหลาย ๆ ผลลัพธ์ ซึ่งแสดงได้ดังรูป



รูปที่ 2.38 แสดงโครงข่ายประสาทแบบ ACON

ประสิทธิภาพระหว่างลักษณะโครงข่าย 2 แบบนี้แสดงได้จากตารางข้างล่างนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	ความถูกต้องในการฝึกสอน	ความถูกต้องโดยเฉลี่ย	เวลาที่ใช้ในการฝึกสอน
ACON (BP)	405/432 = 94%	324/396 = 82%	Normalized = 1.00
OCON (BP)	430/432 = 99.5%	344/396 = 87%	About = 0.25

ตารางที่ 2-2 ตารางแสดงการเปรียบเทียบประสิทธิภาพระหว่าง
โครงข่ายประสาทแบบ ACON และOCON

โดยแต่ละตัวอักษรจะมีเพียง 1 โครงข่ายซึ่งในที่นี้ประกอบไปด้วย ชั้นข้อมูลป้อนเข้า (Input Layer) 36 โหนด ชั้นกลาง (Hidden Layer) 30 โหนด และ ชั้นผลลัพธ์ (Output Layer) 1 โหนด โดยจะให้ผลลัพธ์ออกมาว่าใช้ตัวอักษรนั้นหรือไม่ จากนั้นจะนำผลลัพธ์ของทุกโครงข่ายมาพิจารณาว่าควรเป็นตัวอะไร

<u>กำหนดให้</u>

Input [i] = ข้อมูลป้อนเข้าที่ใช้ในการฝึกสอนตัวที่ i

TAR [k] = เป้าหมายที่ต้องการ

Deltao [k] = ตำแหน่งที่ใช้ในการปรับค่าความผิดพลาดของค่าน้ำหนักระหว่างชั้นกลางกับ ชั้นผลลัพธ์ให้ถูกต้อง

Deltah [h] = สัดส่วนที่ใช้ในการปรับค่าความผิดพลาดให้ถูกต้องสำหรับค่าน้ำหนักระหว่างชั้นข้อมูลป้อนเข้ากับชั้นกลาง

n = อัตราการเรียนรู้มีค่าระหว่าง 0.1 ถึง 0.9

alpha = โมเมนตัม (Momentum) ของการปรับค่าน้ำหนัก มีค่าระหว่าง 0.1 ถึง 0.9

WH = ค่าน้ำหนักระหว่างชั้นข้อมูลป้อนเข้ากับชั้นกลาง

WO = ค่าน้ำหนักระหว่างชั้นกลางกับ ชั้นผลลัพธ์

BH [h] = ค่าไบแอส (Bias) บนชั้นกลางตัวที่ h

BO [k] = ค่าไบแอสบนชั้นผลลัพธ์ตัวที่ k

Outputh [h] = ผลลัพธ์บนชั้นกลางตัวที่ h

Output [k] = ผลลัพธ์บนชั้นผลลัพธ์ตัวที่ k

<u>กระบวนการฝึกสอน</u>

<u>ขั้นตอนที่ 0</u> เริ่มกำหนดค่าน้ำหนักทั้งหมดเป็นเลขระหว่าง –0.001 ถึง +0.001

<u>ขั้นตอนที่ 1</u> ทำขั้นตอนที่ 2 ถึง 8 ซ้ำไปจนกว่าจะครบ 10000 รอบ หรือ ความผิดพลาดครั้งปัจจุบันกับครั้งก่อนหน้าต่างกันน้อยกว่า 0.0002

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>กระบวนการดำเนินการไปข้างหน้า</u>

<u>ขั้นตอนที่ 2</u> รับค่าลักษณะเด่นมากำหนดเป็นค่าข้อมูลป้อนเข้าทีละ 36 ค่า

<u>ขั้นตอนที่ 3</u> คำนวณหาค่าผลลัพธ์บนชั้นกลาง โดยมีสมการคือ

Outputh [h] = 1/(1+exp(-temp [h])) โดยที่

$$\text{tempo [h]} = \text{BH [h]} + \sum_{i}^{36} Input[i] * WH[i][h]$$

<u>ขั้นตอนที่ 4</u> คำนวณค่าผลลัพธ์บนชั้นผลลัพธ์ โดยที่สมการคือ

Output [k] = 1/(1+exp(-tempo [k])) โดยที่

$$\text{tempo [k]} = \text{BO [k]} + \sum_{i}^{30} Input[i] * WO[i][k]$$

<u>ขั้นตอนที่ 5</u> errorptemp = TAR [k] – Output [k] จากนั้นนำค่าผิดพลาดรวมกัน ดังนี้

$$\text{errorp} = \sum_{j}^{\text{totalframe}} (\text{errorptemp [j]}\text{^}2)/2$$

<u>กระบวนการย้อนกลับเพื่อปรับค่าน้ำหนักและค่าไบแอส</u>

<u>ขั้นตอนที่ 6</u> คำนวณหา Deltao [k] โดยใช้สมการคือ

Deltao [k] = (TAR [k]-Output [k])*(1-Output [k])*Output [k]

ทำการปรับค่าน้ำหนักระหว่างค่าน้ำหนักระหว่างชั้นกลางกับชั้นผลลัพธ์ โดยมีสมการดังนี้

WO [i][k] = WO [i][k]+n*Deltao[k]*Outputh [i]+alpha*(WO [i][k]- WO_{old} [i][k])

<u>ขั้นตอนที่ 7</u> คำนวณหา Deltah [h] โดยใช้สมการคือ

$$\text{Deltah [h]} = \text{Outputh [h]}*(1- \text{Outputh [h]})* \sum (\text{Deltao [k]} * \text{WO})$$

ทำการปรับค่าน้ำหนักระหว่างค่าน้ำหนักค่าน้ำหนักระหว่างชั้นข้อมูลป้อนเข้ากับชั้นกลาง โดยมีสมการดังนี้

WH [i][h] = WH [i][h]+n*Deltah[h]*Input [i]+alpha*(WH [i][h]- WH_{old} [i][h])

<u>ขั้นตอนที่ 8</u> ทำการคำนวณค่าไบแอสที่ชั้นผลลัพธ์

BO [k] = BO [k]+n*Deltao [k]+alpha*(BO [k]- BO_{old} [k])

<u>ขั้นตอนที่ 9</u> ทำการคำนวณค่าไบแอสที่ชั้นกลาง

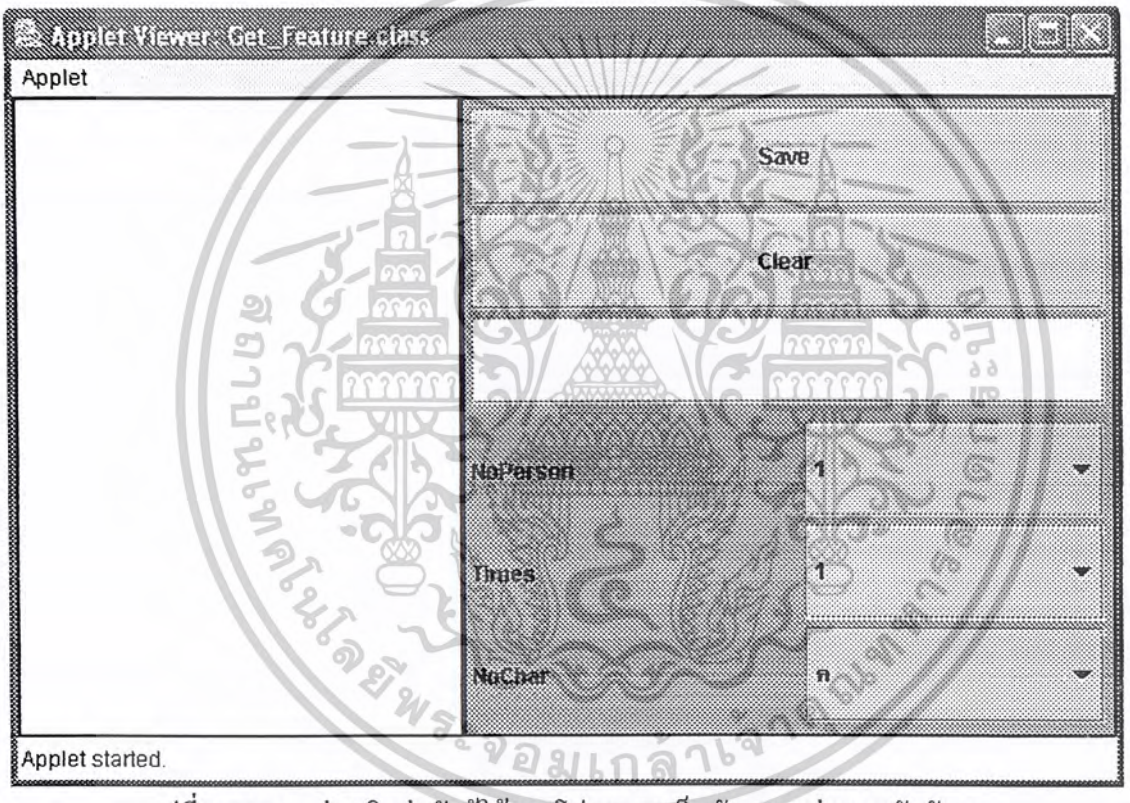
BH [h] = BH [h] +n*Deltah [h]+alpha*(BH [h]- BH_{old} [h])

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
โปรแกรมที่เกี่ยวข้อง

3.1 โปรแกรมเก็บลักษณะเด่นของตัวอักษรไว้ในไฟล์

โปรแกรมนี้ทำหน้าที่ในการดึงลักษณะเด่นของตัวอักษรที่ได้จากการเขียนเก็บไว้ในไฟล์ ซึ่งสามารถแสดงส่วนติดต่อกับผู้ใช้ได้ดังนี้



รูปที่ 3-1 แสดงส่วนติดต่อกับผู้ใช้ของโปรแกรมเก็บลักษณะเด่นของตัวอักษร

โปรแกรมนี้ประกอบไปด้วยคลาสของภาษาจาวาทั้งหมด 7 คลาส ดังนี้

3.1.1 Clear_Button ทำหน้าที่ในการตั้งค่าเริ่มต้นให้กับโปรแกรม

3.1.2 FeatureExtraction ทำหน้าที่ในการดึงลักษณะเด่นของตัวอักษรออกมา

3.1.3 Get_Feature ทำหน้าที่รวมคลาสต่าง ๆ เข้ามาทำงานร่วมกันและแสดงเป็นส่วนติดต่อผู้ใช้

3.1.4 GetFeature_Button ทำหน้าที่ในการเก็บลักษณะเด่นของตัวอักษรไว้ในไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.5 Input_Panel ทำหน้าที่ในการรับข้อมูลจากการเขียน

3.1.6 TFeature เก็บตัวแปรลักษณะเด่นของตัวอักษร

3.1.7 TInfor เก็บข้อมูลอื่น ๆ เกี่ยวกับตัวอักษร

การทำงานของโปรแกรมนี้จะเริ่มจากรับค่าด้วยการเขียนตัวอักษร กำหนดชื่อผู้เขียน กำหนดลำดับผู้เขียน กำหนดจำนวนครั้งที่เขียน และกำหนดตัวอักษรที่จะเขียน จากนั้นโปรแกรมจะทำการดึงลักษณะเด่นของตัวอักษรออกมาเก็บไว้ไฟล์หลังจากคลิกปุ่ม save โดยชื่อไฟล์จะสอดคล้องกับข้อมูลที่ป้อนเข้ามาซึ่งจะมีชื่อว่า "ลำดับผู้เขียน-กำหนดจำนวนครั้งที่เขียน-ลำดับตัวอักษร" ซึ่งก่อนป้อนข้อมูลใหม่ลงไปทุกครั้งต้องคลิกปุ่ม clear เสมอ โดยสามารถแสดงตัวอย่างรายละเอียดข้อมูลในไฟล์ได้ดังนี้

Name : Rujjana Utaiwan

TotalFrame : 15

Frame : 1

LengthOfLine : 0.26278278

Direction : 0.24860393

StartLine : 1.0

EndLine : 0.06666667

HorizontalLeft : 0.5043478

HorizontalMiddle : 0.0

HorizontalRight : 0.0

VerticalUp : 0.0

VerticalMiddle : 0.22608696

VerticalDown : 0.27826086

AngleStart : -1.0

AngleEnd : 0.058496196

Frame : 2

LengthOfLine : 0.019694127

Direction : 0.30710012

StartLine : 0.93333334

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

EndLine : 0.13333334

HorizontalLeft : 0.5555556

HorizontalMiddle : 0.0

HorizontalRight : 0.0

VerticalUp : 0.0

VerticalMiddle : 0.5555556

VerticalDown : 0.0

AngleStart : 0.058496196

AngleEnd : 0.10688942

Frame : 3

LengthOfLine : 0.0134404795

Direction : 0.41398954

StartLine : 0.8666667

EndLine : 0.2

HorizontalLeft : 0.5

HorizontalMiddle : 0.0

HorizontalRight : 0.0

VerticalUp : 0.0

VerticalMiddle : 0.5

VerticalDown : 0.0

AngleStart : 0.10688942

AngleEnd : 0.14060439

Frame : 4

LengthOfLine : 0.034266636

Direction : 0.554594

StartLine : 0.8

EndLine : 0.26666668

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HorizontalLeft : 0.53333336

HorizontalMiddle : 0.0

HorizontalRight : 0.0

VerticalUp : 0.0

VerticalMiddle : 0.53333336

VerticalDown : 0.0

AngleStart : 0.14060439

AngleEnd : 0.9191218

Frame : 5

LengthOfLine : 0.0140209105

Direction : 0.47371575

StartLine : 0.73333335

EndLine : 0.33333334

HorizontalLeft : 0.5714286

HorizontalMiddle : 0.0

HorizontalRight : 0.0

VerticalUp : 0.0

VerticalMiddle : 0.5714286

VerticalDown : 0.0

AngleStart : 0.9191218

AngleEnd : 0.7938959

Frame : 6

LengthOfLine : 0.020872876

Direction : 0.26761165

StartLine : 0.6666667

EndLine : 0.4

HorizontalLeft : 0.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HorizontalMiddle : 0.0

HorizontalRight : 0.0

VerticalUp : 0.2

VerticalMiddle : 0.3

VerticalDown : 0.0

AngleStart : 0.7938959

AngleEnd : 0.92182887

Frame : 7

LengthOfLine : 0.049651723

Direction : 0.18944052

StartLine : 0.6

EndLine : 0.46666667

HorizontalLeft : 0.52380955

HorizontalMiddle : 0.0

HorizontalRight : 0.0

VerticalUp : 0.52380955

VerticalMiddle : 0.0

VerticalDown : 0.0

AngleStart : 0.92182887

AngleEnd : 0.96531814

Frame : 8

LengthOfLine : 0.061375983

Direction : 0.15475868

StartLine : 0.53333336

EndLine : 0.53333336

HorizontalLeft : 0.5217391

HorizontalMiddle : 0.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HorizontalRight : 0.0

VerticalUp : 0.5217391

VerticalMiddle : 0.0

VerticalDown : 0.0

AngleStart : 0.96531814

AngleEnd : 0.9526297

Frame : 9

LengthOfLine : 0.044278055

Direction : 0.10738836

StartLine : 0.46666667

EndLine : 0.6

HorizontalLeft : 0.5

HorizontalMiddle : 0.0

HorizontalRight : 0.0

VerticalUp : 0.5

VerticalMiddle : 0.0

VerticalDown : 0.0

AngleStart : 0.9526297

AngleEnd : 0.95317113

Frame : 10

LengthOfLine : 0.024825862

Direction : 0.06055947

StartLine : 0.4

EndLine : 0.6666667

HorizontalLeft : 0.0

HorizontalMiddle : 0.54545456

HorizontalRight : 0.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VerticalUp : 0.54545456

VerticalMiddle : 0.0

VerticalDown : 0.0

AngleStart : 0.95317113

AngleEnd : 0.07754697

Frame : 11

LengthOfLine : 0.06491005

Direction : 0.9830125

StartLine : 0.33333334

EndLine : 0.73333335

HorizontalLeft : 0.0

HorizontalMiddle : 0.4827586

HorizontalRight : 0.03448276

VerticalUp : 0.51724136

VerticalMiddle : 0.0

VerticalDown : 0.0

AngleStart : 0.07754697

AngleEnd : 0.93837434

Frame : 12

LengthOfLine : 0.03403326

Direction : 0.9213868

StartLine : 0.26666668

EndLine : 0.8

HorizontalLeft : 0.0

HorizontalMiddle : 0.0

HorizontalRight : 0.5

VerticalUp : 0.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VerticalMiddle : 0.0

VerticalDown : 0.0

AngleStart : 0.93837434

AngleEnd : 0.9460401

Frame : 13

LengthOfLine : 0.034266636

Direction : 0.8674269

StartLine : 0.2

EndLine : 0.8666667

HorizontalLeft : 0.0

HorizontalMiddle : 0.0

HorizontalRight : 0.5

VerticalUp : 0.5

VerticalMiddle : 0.0

VerticalDown : 0.0

AngleStart : 0.9460401

AngleEnd : 0.93378127

Frame : 14

LengthOfLine : 0.058312993

Direction : 0.80120814

StartLine : 0.13333334

EndLine : 0.93333334

HorizontalLeft : 0.0

HorizontalMiddle : 0.0

HorizontalRight : 0.52

VerticalUp : 0.52

VerticalMiddle : 0.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VerticalDown : 0.0

AngleStart : 0.93378127

AngleEnd : 0.95855224

Frame : 15

LengthOfLine : 0.26326758

Direction : 0.75976044

StartLine : 0.06666667

EndLine : 1.0

HorizontalLeft : 0.0

HorizontalMiddle : 0.0

HorizontalRight : 0.5043478

VerticalUp : 0.07826087

VerticalMiddle : 0.27826086

VerticalDown : 0.14782609

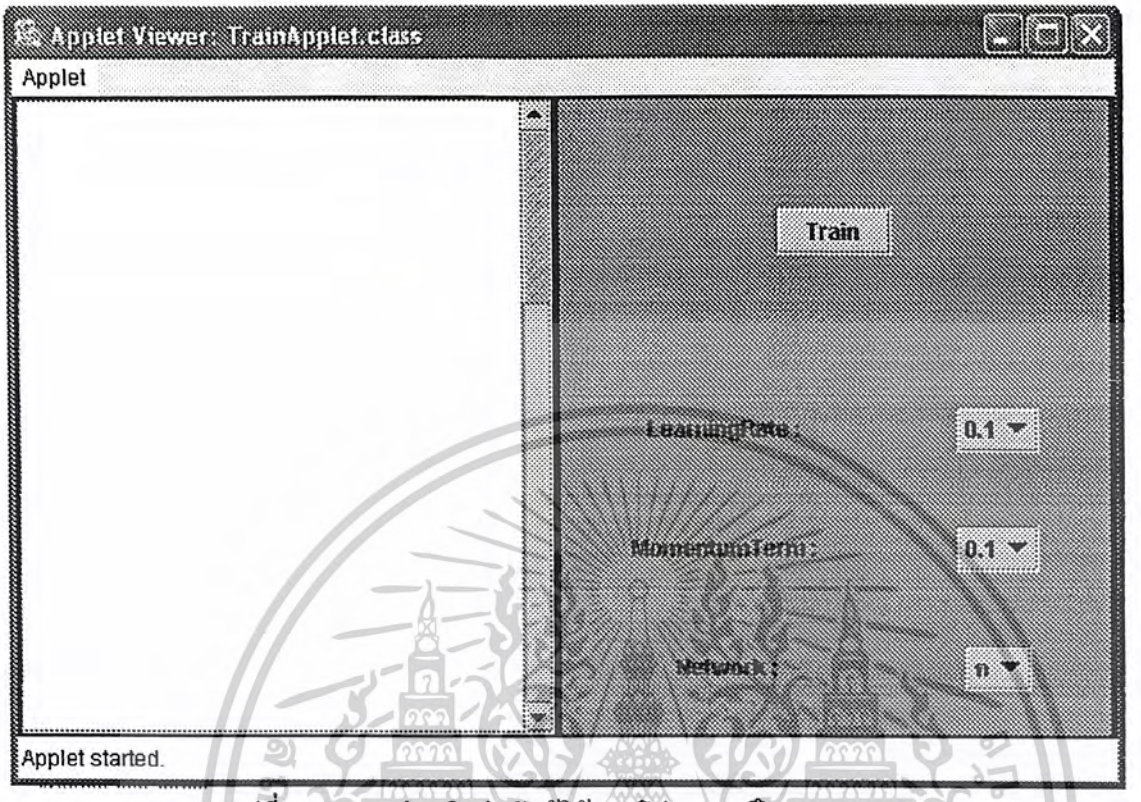
AngleStart : 0.95855224

AngleEnd : -1.0

3.2 โปรแกรมฝึกสอนระบบ

โปรแกรมนี้ทำหน้าที่ฝึกสอนระบบเพื่อให้ระบบเรียนรู้ความแตกต่างของตัวอักษรแต่ละตัวซึ่งสามารถแสดงส่วนติดต่อกับผู้ใช้ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-2 แสดงส่วนติดต่อกับผู้ใช้ของโปรแกรมฝึกสอนระบบ

โปรแกรมนี้ประกอบไปด้วยคลาสของภาษาจาวาทั้งหมด 3 คลาส ดังนี้

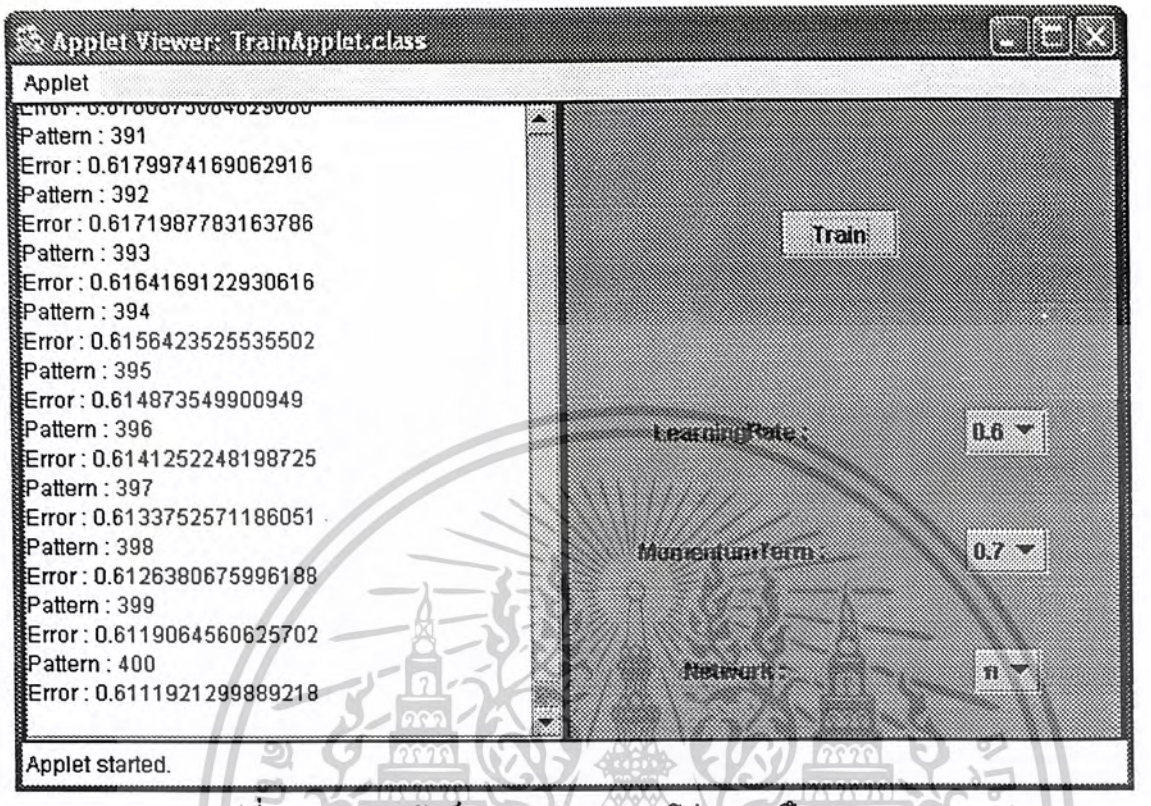
3.1.1 TrainApplet ทำหน้าที่รวมคลาสต่าง ๆ เข้ามาทำงานร่วนกันและแสดงเป็นส่วนติดต่อผู้ใช้

3.1.2 TrainNeural ทำหน้าที่ในการนำค่าต่าง ๆ เช่น ค่าลักษณะเด่นของตัวอักษรที่ได้จากโปรแกรมแรก เป็นต้น มาคำนวณเพื่อให้ได้ค่าน้ำหนักของโครงข่ายประสาท

3.1.3 Train_Button ทำหน้าที่สั่งเริ่มการทำงานของโปรแกรมและเก็บค่าน้ำหนักของโครงข่ายประสาทไว้ในไฟล์

การทำงานของโปรแกรมนี้จะเริ่มจากรับค่าด้วยการกำหนดว่าต้องการจะฝึกสอนให้กับระบบของตัวอักษรใด กำหนดค่าอัตราการเรียนรู้ และกำหนดค่าโมเมนตัม(MomentumTerm) จากนั้นโปรแกรมจะนำค่าทั้งหมดรวมทั้งค่าลักษณะเด่นของตัวอักษรที่ได้จากโปรแกรมแรกมาทำการคำนวณจนได้ค่าน้ำหนักที่น่าพอใจ จากนั้นจะเก็บค่าน้ำหนักไว้ในไฟล์ โดยการทำงานต่าง ๆ จะเริ่มขึ้นหลังจากคลิกที่ปุ่ม Train โดยสามารถแสดงผลการทำงานของโปรแกรมได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

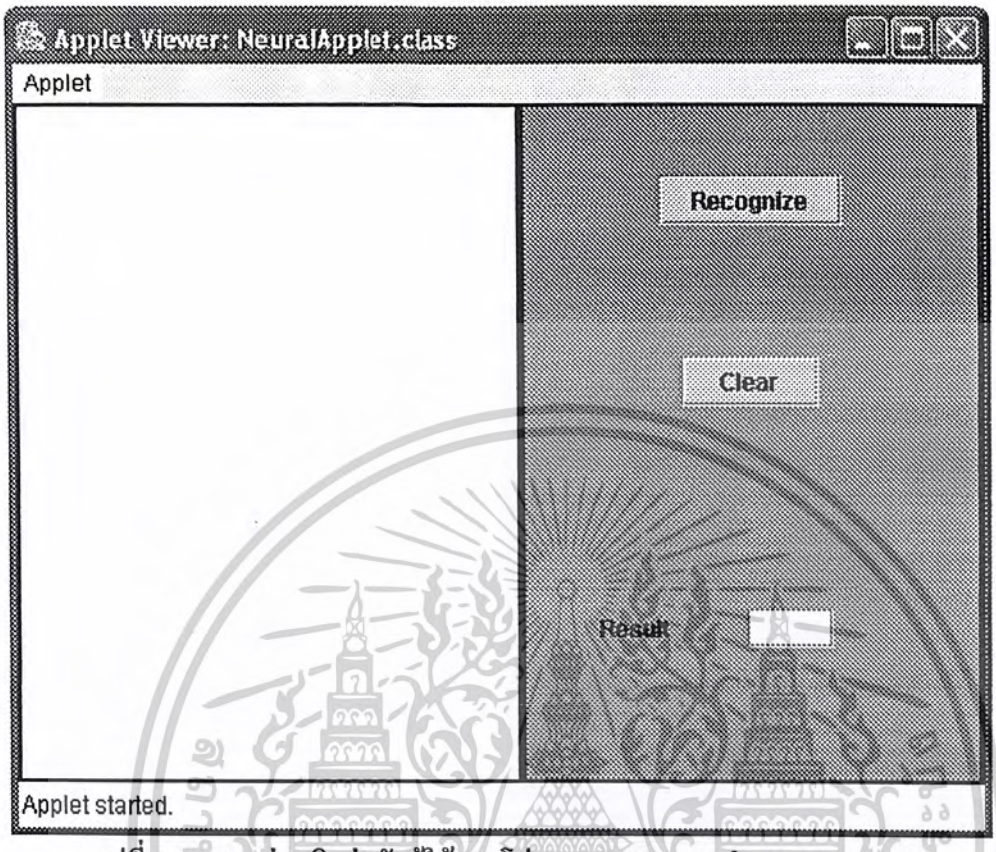


รูปที่ 3-3 แสดงผลลัพธ์จากการทำงานของโปรแกรมฝึกสอนระบบ

3.3 โปรแกรมแสดงการทำงานของระบบ

โปรแกรมนี้ทำหน้าที่แสดงผลการทำงานของระบบหลังจากได้มีการฝึกสอนระบบแล้ว หรือก็คือโปรแกรมที่นำไปใช้งานจริง ๆ ซึ่งสามารถแสดงส่วนติดต่อกับผู้ใช้ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-4 แสดงส่วนติดต่อกับผู้ใช้ของโปรแกรมแสดงการทำงานของระบบ

โปรแกรมนี้ประกอบไปด้วยคลาสของภาษาจาวาทั้งหมด 9 คลาส ดังนี้

3.3.1 Clear_Button ทำหน้าที่ในการตั้งค่าเริ่มต้นให้กับ โปรแกรม

3.3.2 FeatureExtraction ทำหน้าที่ในการดึงลักษณะเด่นของตัวอักษรออกมา

3.3.3 Input_Panel ทำหน้าที่ในการรับข้อมูลจากการเขียน

3.3.4 NeuralApplet ทำหน้าที่รวมคลาสต่างๆเข้ามาทำงานร่วมกันและแสดงเป็นส่วนติดต่อผู้ใช้

3.3.5 Recognize_Button ทำหน้าที่เริ่มการทำงานของระบบ

3.3.6 TestNeural ทำหน้าที่นำค่าต่าง ๆ มาคำนวณ และวิเคราะห์ตัวอักษร

3.3.7 TFeature เก็บค่าลักษณะเด่นของตัวอักษร

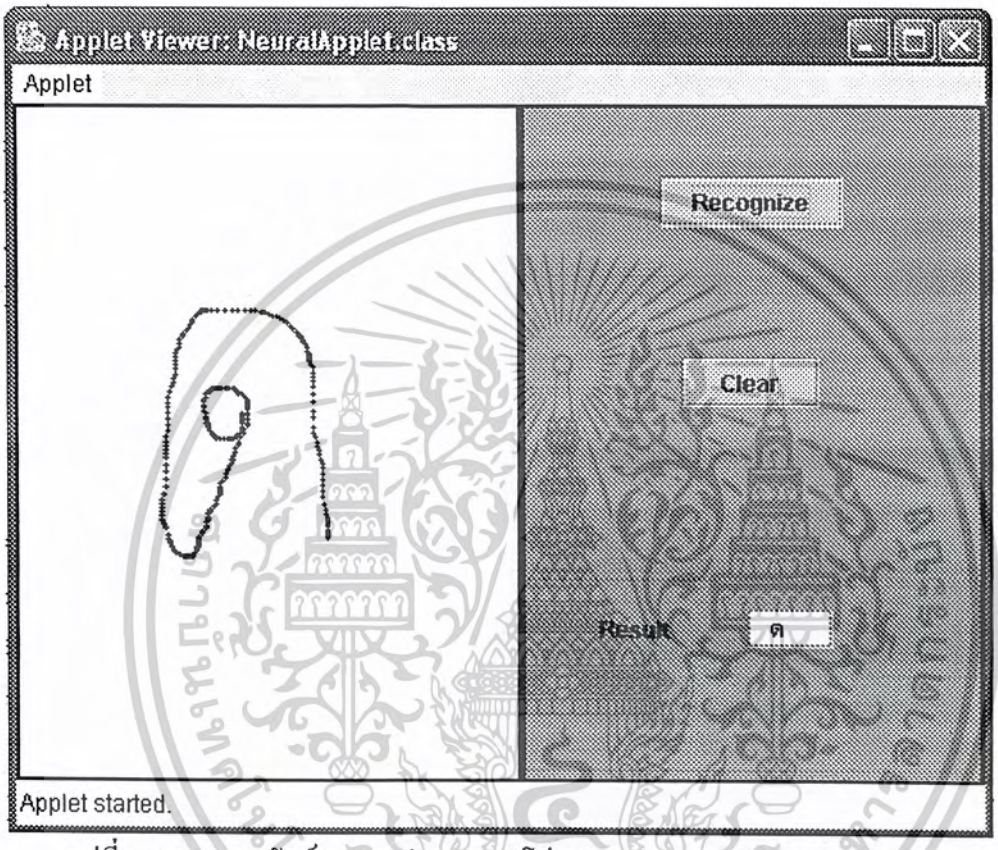
3.3.8 TInfor เก็บข้อมูลอื่น ๆ เกี่ยวกับตัวอักษร

3.3.9 TProb เก็บค่าความน่าจะเป็นของตัวอักษร

การทำงานของโปรแกรมนี้จะเริ่มจากรับค่าด้วยการเขียนตัวอักษร จากนั้นโปรแกรมจะทำการวิเคราะห์ว่าเป็นตัวอักษรตัวอะไรหลังจากคลิกปุ่ม Recognize โดยผลลัพธ์ที่ได้จะแสดงออกมาในช่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Result เมื่อต้องการป้อนข้อมูลใหม่ลงไปทุกครั้งต้องคลิกปุ่ม clear เสมอ โดยสามารถแสดงตัวอย่างรายละเอียดข้อมูลในไฟล์ได้ดังนี้



รูปที่ 3-5 แสดงผลลัพธ์จากการทำงานของโปรแกรมแสดงการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

4.1 การทดสอบประสิทธิของโปรแกรม

การทดสอบประสิทธิของโปรแกรมจะแบ่งชุดตัวอักษรที่ใช้ในการทดสอบออกเป็น 2 ชุด ดังนี้

4.1.1 การทดสอบประสิทธิภาพในกรณีตัวอักษรมีลักษณะแตกต่างกัน

การทดสอบประสิทธิภาพของตัวอักษรในชุดนี้จัดทำขึ้นมาก่อนการทดสอบประสิทธิภาพในกรณีตัวอักษรมีลักษณะคล้ายกัน ซึ่งเป็นเป้าหมายของโครงงานในครั้งนี้ โดยมาจากแนวคิดที่ว่า "ก่อนที่ระบบจะสามารถแยกความแตกต่างของตัวอักษรที่มีลักษณะคล้ายกันได้ ระบบควรจะแยกความแตกต่างของตัวอักษรที่มีลักษณะแตกต่างกันได้ก่อน" ตัวอักษรในชุดนี้ประกอบไปด้วย ก ข พ ล และ อ

4.1.1.1 การฝึกสอน

ในการทดลองชุดนี้จะใช้คนในการฝึกสอนระบบทั้งหมด 3 คน โดยแต่ละคนจะเขียนตัวอักษรที่ละตัว คือ ก ข พ ล และ อ ทั้งหมด 5 ครั้ง ลงในโปรแกรมรับค่า จากนั้นจะทำการฝึกสอนระบบโดยจะเรียกโปรแกรมฝึกสอนระบบ โดยจะทำการฝึกสอนทีละตัวอักษร ซึ่งเวลาที่ใช้ในการฝึกสอนระบบสามารถแสดงได้ดังนี้

	ก	ข	พ	ล	อ
เวลาที่ใช้ในการฝึกสอนระบบ(นาที)	9.47	13.22	12.53	11.12	9.49

ตารางที่ 4-1 ตารางแสดงเวลาที่ใช้ในการฝึกสอนระบบตัวอักษรแตกต่างกัน

4.1.1.2 ผลจากการทดสอบระบบ

ในการทดสอบระบบชุดนี้ จะใช้คนในการทดสอบระบบทั้งหมด 10 คน โดยแต่ละคน จะเขียนตัวอักษรทีละตัว คือ ก ข พ ล และ อ ทั้งหมด 3 ครั้ง ลง ในโปรแกรมทดสอบระบบ ซึ่งผลที่ได้สามารถแสดงได้ดังนี้

	ก	ข	พ	ล	อ
จำนวนครั้งที่ให้ผลถูกต้อง	30	30	30	30	30
เปอร์เซ็นต์ความถูกต้อง	100	100	100	100	100

ตารางที่ 4-2 ตารางแสดงประสิทธิภาพของระบบตัวอักษรแตกต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 การทดสอบประสิทธิภาพในกรณีตัวอักษรมีลักษณะคล้ายกัน

การทดสอบประสิทธิภาพของตัวอักษรในชุดนี้เป็นเป้าหมายของการจัดทำโครงงานในครั้งนี้ โดยมาจากแนวคิดที่ว่า "ระบบที่สามารถแยกความแตกต่างของตัวอักษรที่มีลักษณะแตกต่างกันได้ ระบบก็น่าจะแยกความแตกต่างของตัวอักษรอื่น ๆ ได้เหมือนกัน" ตัวอักษรในชุดนี้ประกอบไปด้วย ก ถ ภ ด และ ต

4.1.2.1 การฝึกสอน

ในการทดลองชุดนี้จะใช้คนในการฝึกสอนระบบทั้งหมด 6 คน โดยแต่ละคนจะเขียนตัวอักษรที่ละตัว คือ ก ถ ภ ด และ ต ทั้งหมด 5 ครั้ง ลงในโปรแกรมรับค่า จากนั้นจะทำการฝึกสอนระบบโดยจะเรียกโปรแกรมฝึกสอนระบบ โดยจะทำการฝึกสอนทีละตัวอักษร ซึ่งเวลาที่ใช้ในการฝึกสอนระบบสามารถแสดงได้ดังนี้

	ก	ถ	ภ	ด	ต
เวลาที่ใช้ในการฝึกสอนระบบ(นาที)	17.42	156.10	213.03	173.51	147.42

ตารางที่ 4-3 ตารางแสดงเวลาที่ใช้ในการฝึกสอนระบบตัวอักษรคล้ายกัน

4.1.2.2 ผลจากการทดสอบระบบ

ในการทดสอบระบบชุดนี้ จะใช้คนในการทดสอบระบบทั้งหมด 10 คน โดยแต่ละคน จะเขียนตัวอักษรทีละตัว คือ ก ถ ภ ด และ ต ทั้งหมด 3 ครั้ง ลง ในโปรแกรมทดสอบระบบ ซึ่งผลที่ได้สามารถแสดงได้ดังนี้

	ก	ถ	ภ	ด	ต
จำนวนครั้งที่ให้ผลถูกต้อง	30	28	30	27	30
เปอร์เซ็นต์ความถูกต้อง	100	93.33	100	90	100

ตารางที่ 4-4 ตารางแสดงประสิทธิภาพของระบบตัวอักษรคล้ายกัน

โดยความผิดพลาดที่เกิดขึ้นมีดังนี้

- เมื่อเขียน ถ บางครั้งให้ผลเป็น ภ

- เมื่อเขียน ด บางครั้งให้ผลเป็น ต

4.1.3 โปรแกรมสมบูรณ์

โปรแกรมนี้พยายามจะให้สามารถแยกแยะตัวอักษรให้มากที่สุดเพื่อความสมบูรณ์ที่สุดของโปรแกรม ตัวอักษรในชุดนี้ประกอบไปด้วย ก ข ค ฆ ง จ ฉ ช ซ ฌ ญ ฐ ฒ ณ ด ต ถ ท ธ และ น

4.1.3.1 การฝึกสอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการทดลองชุดนี้จะใช้คนในการฝึกสอนระบบทั้งหมด 16 คน โดยแต่ละคนจะเขียนตัวอักษรที่ละตัว คือ ก ข ค ฆ ง จ ฉ ช ซ ฌ ญ ฐ ฒ ณ ด ต ถ ท ธ และ น ทั้งหมด 2 ครั้ง ลงในโปรแกรมรับค่า จากนั้นจะทำการฝึกสอนระบบโดยจะเรียกโปรแกรมฝึกสอนระบบ โดยจะทำการฝึกสอนทีละตัวอักษร ซึ่งเวลาที่ใช้ในการฝึกสอนระบบสามารถแสดงได้ดังนี้

ตัวอักษร	เวลาที่ใช้ในการฝึกสอนระบบ(นาที)
ก	34.33
ข	229.44
ค	371.06
ฆ	622.25
ง	269.59
จ	840.13
ฉ	710.15
ช	138.55
ซ	61.27
ฌ	867.21
ญ	1433.10
ฐ	574.25
ฒ	30.00
ณ	23.05
ด	734.24
ต	154.24
ถ	316.52
ท	244.05
ธ	435.14
น	30.58

ตารางที่ 4-5 ตารางแสดงเวลาที่ใช้ในการฝึกสอนระบบที่สมบูรณ์

4.1.3.2 ผลจากการทดสอบระบบ

ในการทดสอบระบบชุดนี้ จะใช้คนในการทดสอบระบบทั้งหมด 10 คน โดยแต่ละคน จะเขียนตัวอักษรทีละตัว คือ ก ข ค ฆ ง จ ฉ ช ซ ฌ ญ ฐ ฒ ณ ด ต ถ ท ธ และ น ทั้งหมด 3 ครั้ง ลง ในโปรแกรมทดสอบระบบ ซึ่งผลที่ได้สามารถแสดงได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอักษร	จำนวนครั้งที่ให้ผลถูกต้อง	เปอร์เซ็นต์ความถูกต้อง
ก	30	100
ข	30	100
ค	29	96.67
ฆ	28	93.33
ง	30	100
จ	29	96.77
ฉ	29	96.77
ช	27	90
ซ	30	100
ฌ	20	66.67
ญ	21	70
ฐ	29	96.67
ฒ	28	93.33
ณ	30	100
ด	24	80
ต	30	100
ถ	30	100
ท	27	90
ธ	30	100
น	30	100

ตารางที่ 4-6 ตารางแสดงประสิทธิภาพของระบบสมบูรณ์

หมายเหตุ เหตุผลที่ไม่สามารถทำให้ครบ 44 ตัวอักษรนั้น คือ การฝึกสอนระบบต้องใช้เวลานานมากเมื่อจำนวนตัวอักษรเพิ่มขึ้น และ ไม่สามารถหาคนมาฝึกสอนระบบในจำนวนมากๆได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุป

5.1 สรุปการพัฒนาโครงงาน

ประสิทธิภาพที่ได้นั้นเป็นที่น่าพอใจซึ่งมีเปอร์เซ็นต์ความถูกต้องถึง 90 เปอร์เซ็นต์ อย่างไรก็ตาม การทดลองนี้เป็นการทดลองตัวอักษรที่มีจำนวนไม่มากนัก

5.2 ปัญหาที่เกิดขึ้น

5.2.1 ไม่สามารถหาผู้ที่จะฝึกสอนระบบได้มากตามต้องการ เนื่องจากการเพิ่มจำนวนผู้ฝึกสอนจะเป็นการเพิ่มเวลาที่ใช้ในการฝึกสอน

5.2.2 แป้นเขียนปากกาอิเล็กทรอนิกส์มีขนาดเล็กเกินไป ทำให้เขียนไม่ถนัด

5.3 ข้อจำกัดของโครงงาน

5.3.1 งบประมาณที่น้อยเกินไปทำให้ไม่สามารถซื้อปากกาอิเล็กทรอนิกส์ที่มีคุณภาพดีได้ ทำให้ผู้ฝึกสอนระบบและผู้ทดสอบระบบเขียนไม่เหมือนกับที่เคยเขียนโดยใช้ปากกาปรกติ

5.3.2 เนื่องจากการฝึกสอนระบบต้องใช้เวลามากทำให้ไม่สามารถเพิ่มจำนวนตัวอักษรของระบบได้มากนัก

5.4 แนวทางในการพัฒนาต่อ

โปรแกรมนี้ได้ออกแบบจนสมบูรณ์แล้วผู้ที่ต้องการนำไปพัฒนาต่อสามารถนำไปใช้ได้เลยโดยไม่ต้องแก้ไขโปรแกรม ซึ่งการพัฒนาต่ออาจทำได้โดย เพิ่มจำนวนตัวอักษรจนครบทุกตัว หรืออาจพัฒนาไปจนถึงสามารถวิเคราะห์คำหรือประโยคได้ แต่ในกรณีหลังต้องมีการแก้โปรแกรมบ้างเนื่องจากโปรแกรมนี้ออกแบบมาเพื่อตัวอักษรเพียงตัวเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. Sutat Sae-Tang and Ithipan Methaste , "Thai Online Handwritten Character Recognition Using Backpropagation Neural Networks" Information Research and Development Division, National Electronics and Computer Technology Center, National Science and Technology Development Agency, Bangkok

2. Abhijit S. Pandya and Robert B. Macy, "Pattern Recognition with Neural Networks in C++", 1996

3. Jame A. FreeMan, "Simulating Neural Networks with Mathematica"

4. นาย นราธิป สันติทฤษฎีกร และ นางสาว วนิชา สังวรเวชภัณฑ์, "การจดจำรูปแบบตัวอักษร", ปริญญานิพนธ์, คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า วิทยาเขตเจ้าคุณทหาร ลาดกระบัง กรุงเทพฯ, ปีการศึกษา 2540

5. Aleksander, I. And Morton, H., "An Introduction to Neural Computing" Chapman & Hall, London, 1990

6. ดร. วีระศักดิ์ ซึงถาวร, "Java programming volume 1", กรุงเทพฯ, 2543

7. ดร. วีระศักดิ์ ซึงถาวร, "Java programming volume 2, กรุงเทพฯ, 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้