การรู้จำตัวอักษรคัดลายมือเขียนภาษาไทยโดยประยุกต์ใช้การทรานสฟอร์ม แบบคาร์ฮูเน็นเลฟ์เข้ากับโครงข่ายฟัซซีนิวรอล

HANDPRINTED THAI CHARACTER RECOGNITION USING K-L TRANSFORM WITH FUZZY NEURAL NETWORK

สกุล คำนวนชัย

SKUL KAMNUANCHAI

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าธนบุรี

พ.ศ. 2542

ISBN 974-622-616-9

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การรู้จำตัวอักษรคัดลายมือเขียนภาษาไทยโดยประยุกต์ใช้การทรานสฟอร์ม แบบคาร์ฮูเน็นเลฟเข้ากับโครงข่ายฟัซซี่นิวรอล

HANDPRINTED THAI CHARACTER RECOGNITION USING K-L TRANSFORM WITH FUZZY NEURAL NETWORK

สกุล คำนวนชัย

SKUL KAMNUANCHAI

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2542

ISBN 974-622-616-9

เลขหมู่ 34733
เลขทะเบียน
วัน, เดือน, ปี 29 S.A. 2542

HANDPRINTED THAI CHARACTER RECOGNITION USING K-L TRANSFORM WITH FUZZY NEURAL NETWORK

SKUL KAMNUANCHAI

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

1999

ISBN 974-622-616-9

COPYRIGHT 1999

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

หัวข้อวิทยานิพนธ์ การรู้จำตัวอักษรคัดลายมือเขียนภาษาไทยโดยประยุกต์ใช้การทรานสฟอร์มแบบคาร์ฮูเน็นเลิฟเข้ากับโครงข่ายฟัซซี่นิวรอล
HANDPRINTED THAI CHARACTER RECOGNITION USING K-L TRANSFORM WITH FUZZY NEURAL NETWORK

นักศึกษา นายสกุล คำนวนชัย

รหัสประจำตัว 35620058

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา วิศวกรรมไฟฟ้า

พ.ศ. 2542

อาจารย์ผู้ควบคุมวิทยานิพนธ์ รศ.ดร.ชม กิ้มปาน

บทคัดย่อ

ในวิทยานิพนธ์ฉบับนี้ ได้พัฒนาระบบการรู้จำตัวอักษรคัดลายมือเขียนภาษาไทย โดยได้ทำการทดสอบกับตัวอักษรพยัญชนะทั้งหมด 44 ตัว ซึ่งได้ประยุกต์การแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟเพื่อหาข้อมูลที่เป็นตัวแทนพยัญชนะแต่ละตัวโดยหาจากตัวอักษรแต่ละพยัญชนะที่มีรูปแบบต่างกัน 30รูปแบบได้จากผู้คัดลายมือ 10 คนโดยเขียนคนละ 3 ครั้ง จะได้ข้อมูลที่ลักษณะร่วมของพยัญชนะแต่ละตัว นอกจากนั้นได้ทำการปรับค่าของข้อมูลโดยใช้หลักการของฟัซซี่เซตให้เหลือค่าความเป็นสมาชิกของฟัซซี่คือ 0 , 0.2 , 0.4 , 0.6 , 0.8 และ 1 เป็นการลดจำนวนฟีเจอร์(feature) ของข้อมูล ตัวแทนพยัญชนะลง ข้อมูลที่ได้ก็คือ ข้อมูลที่ใช้เป็นตัวแทนพยัญชนะแต่ละตัวเข้าไปเป็นข้อมูล ให้กับขั้นตอนการเรียนรู้โดยใช้โครงข่ายนิวรอลแบบแบล็คพรอบพาเกชั่นซึ่งมีเลเยอร์(layer) ทั้งหมด 3 เลเยอร์คือ อินพุทเลเยอร์ , ฮิดเดนเลเยอร์ และเอ้าท์พุทเลเยอร์ โดยจำนวนโหนดของแต่ละเลเยอร์มีดังนี้ 512 โหนด, 100โหนด, 44 โหนด ตามลำดับโดยทำการจำลองโครงข่ายนิวรอล บนเครื่องคอมพิวเตอร์ส่วนบุคคล(Personal Computer)166 MHz CPU, 64 MB RAM, 8 MB Harddisk

การทดลองในงานวิทยานิพนธ์นี้ได้ทำการทดลองทั้งหมด 4 ลักษณะ โดยที่แต่ละลักษณะจะใช้วิธีที่แตกต่างกัน เพื่อหาตัวแทนของพยัญชนะแต่ละตัวนำข้อมูลที่ได้ไปทำการเรียนรู้โดยใช้โครงข่าย นิวรอล ใช้วิธีการนำข้อมูลมาหาค่าเฉลี่ยแบบธรรมดาได้ผลการรู้จำถูกต้องร้อยละ 80 , ใช้วิธีการแปลง ข้อมูลแบบคาร์ฮูเน็นเลิฟได้ผลการรู้จำถูกต้องร้อยละ80,ใช้วิธีการนำข้อมูลมาหาค่าเฉลี่ยแบบทำธรรมดาแล้วปรับข้อมูลให้อยู่ในระบบของฟัซซี่เซตได้ผลการรู้จำถูกต้องเป็นร้อยละ 85 ,ใช้วิธีการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟปรับข้อมูลให้อยู่ในระบบของฟัซซี่เซตได้ผลการรู้จำถูกต้องร้อยละ 85 จากการทดลองในหลายลักษณะทำให้ทราบว่า การหาตัวแทนของตัวอักษรแต่ละพยัญชนะโดยการ

แปลง ข้อมูลแบบคาร์ฮูเน็นเลิฟ กับค่าเฉลี่ยแบบทำธรรมดา จะทำให้ได้ผลการทดลองที่เหมือนกันแต่ผลลัพธ์ที่ได้จะให้ค่าที่มีความละเอียดมากกว่า สำหรับการนำฟัซซี่เข้ามาปรับลดจำนวนฟรีเจอร์ของข้อมูลจะทำให้ ผลการทดลองถูกต้องเพิ่มขึ้นในอัตราร้อยละ 5 นอกจากนั้นยังทำให้เวลาในการเรียนรู้ลดลง ดังนั้นจะเห็นได้ว่าการหาตัวแทนของข้อมูลโดยการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ และปรับข้อมูลให้อยู่ในรูปฟัซซี่เซ็ตจะให้ผลการทดลองถูกต้องมากที่สุด

จากงานวิจัยเรื่องการรู้จำตัวอักษรตัวพิมพ์ภาษาไทยโดยวิธีโครงข่ายนิวรอลแบบแบคพรอบพราเกชั่น (8) จะทำการรู้จำตัวอักษรทั้งหมด 87ตัวโดยใช้ข้อมูลบิตแม็บของตัวอักษรฟอร์นอังสานาขนาด 12 จุด นำเข้าไปเรียนรู้โดยใช้โครงข่ายนิวรอลใช้เวลาในการเรียนรู้ 13 วันให้ผลความถูกต้องร้อยละ 98 สำหรับในการทดลองให้วิทยานิพนธ์นี้จะใช้เวลาในการเรียนรู้ทั้งหมด 6 ชั่วโมงได้ผลความถูกต้องร้อยละ 85 ซึ่งจะเห็นได้ว่าวิธีการในวิทยานิพนธ์นี้จะทำให้ขั้นตอนในการเรียนรู้ใช้เวลาน้อยลงและผลของการทดลองก็อยู่ในเกณฑ์ดี

Thesis Title	Handprinted Thai Character Recognition Using K-L Transform With Fuzzy Neural Network
Student	Mr. Sakul Kamnaunchai
Student ID.	35620058
Degree	Master of Engineering
Programme	Electrical Engineering
Year	1999
Thesis Advisor	Assoc.Prof.Dr. Chom Kimpan

ABSTRACT

This thesis presents Hand-printed Thai Character Recognition. It was developed and experimented with forty-four Thai character sets of which each set contains thirty patterns of a character. Ten selected people who repeatedly wrote three same characters created the thirty patterns of each set. The K-L transformation was applied to create a group of forty-four Thai characters that represent each transformed character. The fuzzy set theory applied to the group to make all features of every element in the group to be on of the fuzzy values of 0, 0.2, 0.4, 0.6, 0.8 and 1. In the other hand, the features resolution was reduced. This new group was inputted to designed neural networks, which were back propagation type and which contained three layers. Their layers were composed of an input layer, a hidden layer, and an output layer. And they were composed of 512 nodes, 100 nodes, and 44 nodes, respectively. The neural network experiments were made on a personal computer of 166 MHz CPU, 64 MB RAM, 8 MB Harddisk.

Four experiments were made in this thesis. Each experiment was made differently in finding the represented characters group that was eventually the input of the neural networks. The experiments stated that the average character data method gave 80% of recognition success rate. The K-L transformation method also gave 80% of recognition success rate. For the group that made from the average character data method and modified to be in our designed fuzzy system could give 85% of recognition success rate. In addition to the group that made from the K-L transformation method and modified to be in our designed fuzzy system could give 85% of recognition success rate. The experiment also concluded that the average character data and the K-L transformation method gave the same rate of success. However, the fuzzy system could make

the experiment result more accuracy 5% and also reduce the learning time of the system. Obviously, it showed that hand-printed Thai character recognition by the K-L transformation with the fuzzy system give the best accuracy.

From the past research of Thai character recognition, using a back propagation neural network learning 87 characters of 12-pts Angsana font bitmap data, it took 13 days for the learning time and gave 98% accuracy. For our research, the learning time took 6 hours yet the given accuracy was 85%. It concluded that our research could shorten the learning time and our method accuracy was also fine.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้อย่างดี ด้วยคำแนะนำและคำปรึกษาเกี่ยวการรู้จำตัวอักษร รวมทั้งได้ตรวจสอบเนื้อหาตลอดจนห้องทดลองและอุปกรณ์เครื่องใช้ในการทำงานของวิทยานิพนธ์นี้จาก รศ.ดร. ชม กิ้มปาน ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์ ผู้วิจัยรู้สึกซาบซึ้งในความอนุเคราะห์จากท่านและขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณวิทยาลัยเทคนิคลพบุรี กรมอาชีวศึกษา กระทรวงศึกษาธิการ ที่ได้อนุญาตให้มาศึกษาต่อ

ขอขอบคุณ อ.พิศิษฎ์ โภคารัตน์กุล,นายเกรียงศักดิ์ เตมีย์และนายอุกฤษณ์ มารังค์และเพื่อนนักศึกษาทุกคนที่ช่วยเหลือแก้ไขและให้คำแนะนำทฤษฎีและอื่น ๆ ที่ผิดพลาด จนสำเร็จสมบูรณ์ยิ่งขึ้นและยังให้กำลังใจต่อผู้วิจัยอย่างใกล้ชิดตลอดมา

ขอขอบพระคุณ บิดา-มารดา ของข้าพเจ้าที่ได้อบรมสั่งสอนข้าพเจ้าด้วยดีตลอดมา

สุดท้ายขอขอบคุณบัณฑิตวิทยาลัย ที่ได้ให้การสนับสนุนในการทำวิทยานิพนธ์ครั้งนี้

คุณค่าและประโยชน์อันพึงมีจากวิทยานิพนธ์ฉบับนี้ ผู้วิจัยขอมอบแด่ผู้มีพระคุณทุกท่าน

สกุล คำนวนชัย

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....III

กิตติกรรมประกาศ.....V

สารบัญ.....VI

สารบัญตาราง.....IX

สารบัญภาพ.....X

บทที่ 1 บทนำ.....1

1.1 กล่าวนำ.....1

1.2 วัตถุประสงค์ของงานวิจัย.....3

1.3 ข้อตกลงเบื้องต้นในการทำวิทยานิพนธ์.....3

1.4 เนื้อหาของวิทยานิพนธ์.....4

บทที่ 2 ทฤษฎีของคาร์ฮูเม็นเลิฟ.....5

2.1 การวิเคราะห์รูปแบบตัวอักษรโดยใช้ K-L Transform.....5

บทที่ 3 ทฤษฎีฟัซซี่เซ็ต.....8

3.1 การแทนข้อมูลด้วยฟัซซี่เซ็ต.....8

3.2 ฟังค์ชั่นความเป็นสมาชิก.....11

3.2.1 เซ็ตเอกภพสัมพันธ์.....11

3.2.2 ซัพพอร์ตเซต.....12

3.2.3 การเทรชโฮลด์ฟัซซี่เซ็ต.....12

3.3 ตัวดำเนินการของฟัซซี่เซ็ต.....17

3.4 การเรียนรู้ของฟัซซี่เซ็ต.....18

3.5 การนำฟัซซี่เซ็ตมาใช้ในการตัดสินใจ.....21

บทที่ 4 ทฤษฎีนิวรอลเน็ตเวอร์ค.....22

4.1 ลักษณะโครงข่ายนิวรอล.....23

4.2 โครงข่ายของนิวรอลแบบต่าง ๆ.....28

สารบัญ (ต่อ)

หน้า

บทที่ 5 ขบวนการเรียนรู้.....37

5.1 ขั้นตอนการเตรียมข้อมูล.....37

5.1.1 ลักษณะของตัวอักขระที่ใช้ในการทดลอง.....37

5.1.2 ปรับข้อมูลให้อยู่ในรูปเมตริก.....38

5.1.3 หาลักษณะร่วมของข้อมูลโดยการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ.....39

5.1.4 แปลงข้อมูลให้อยู่ในรูปของฟัซซี่เซ็ต.....40

5.2 ขั้นตอนการฝึกสอน.....41

5.3 ขั้นตอนการรู้จำ.....41

บทที่ 6 การทดสอบและประเมินผล.....43

6.1 การทดสอบและประเมินผลความถูกต้องใน การจดจำตัวอักษรคัดลายมือเขียนภาษาไทยโดยประยุกต์ใช้การทรานส์ฟอร์มแบบคาร์ฮูเน็นเลิฟ เข้ากับโครงข่ายฟัซซี่นิวรอล.....43

6.2 เปรียบเทียบการทำงานของระบบนี้ในลักษณะต่างๆ กัน.....46

บทที่ 7 สรุปผลการทดลอง.....51

7.1 ระบบการรู้จำตัวอักษรคัดลายมือเขียนภาษาไทยโดยประยุกต์ใช้การทรานส์ฟอร์มแบบคาร์ฮูเน็นเลิฟเข้ากับโครงข่ายฟัซซี่นิวรอล.....51

7.2 ข้อสรุปและเปรียบเทียบของการรู้จำ.....51

7.3 ข้อจำกัดของระบบ.....52

7.4 ข้อเสนอแนะสำหรับแนวทางการพัฒนา.....53

7.5 ข้อวิเคราะห์เปรียบเทียบ.....54

เอกสารอ้างอิง.....55

สารบัญ (ต่อ)

หน้า

ภาคผนวก.....56
ภาคผนวก ก. ตัวอย่างตัวอักษรลัดลายมือภาษาไทย.....57
ภาคผนวก ข โปรแกรมที่พัฒนาขึ้นในการวิจัย.....62
ภาคผนวก ค ผลงานวิจัยที่ได้การตีพิมพ์.....79

ประวัติผู้เขียน.....87

สารบัญตาราง

ตารางที่ หน้า

6.1 ผลการทดสอบโครงข่ายนิวรอลโดยมีฮิดเดนโหนด 100, 200, 300 ,400 , 500.....44

6.2 ผลการทดสอบความถูกต้องของตัวอักขระชุดเดียวกับที่ใช้ในการเตรียมข้อมูลของระบบ.....44

6.3 ผลการทดสอบความถูกต้องของตัวอักขระจากผู้เขียนชุดใหม่.....45

6.4 ผลการทดสอบความถูกต้องของตัวอักขระชุดเดียวกับที่ใช้ในการเตรียมข้อมูลของระบบ.....47

6.5 ผลการทดสอบความถูกต้องของตัวอักขระจากผู้เขียนชุดใหม่.....48

6.6 สรุปความถูกต้องของการรู้จำของทั้งสองระบบ.....49

6.7 แสดงผลความถูกต้องจำนวนผู้คัดลายมือที่ใช้ในการเตรียมข้อมูลและฝึกสอนที่ต่างกัน.....49

6.8 แสดงผลความแตกต่างของลักษณะการคัดลายมือ.....50

7.1 แสดงผลการเปรียบเทียบการรู้จำงานวิจัยนี้กับระบบการรู้จำของนายทวี เปรมรัตนชัย.....52

สารบัญภาพ

ภาพที่ หน้า

1.1 แสดงขั้นตอนการทำงานของระบบการรู้จำตัวอักษรคัดลายมือคัดภาษาไทย.....2
2.1 การหาตัวแทนของตัวอักษรคัดลายมือ ว. ในภาษาไทยเพื่อนำไปใช้ในขบวนการการรู้จำ.....5
2.2 แสดงตำแหน่งของตัวอักษรหนึ่งตัว.....6
2.3 มิติของเมตริก $V = G.G^T$.....7
3.1 แสดงค่าความเป็นสมาชิกของเซ็ต A ในกรณีที่เป็นเซ็ตธรรมดา.....9
3.2 แสดงค่าความเป็นสมาชิกของเทอมเซต A ในกรณีที่เป็นฟัซซี่เซต.....10
3.3 แสดงกราฟฟังก์ชั่นการเป็นสมาชิกของตัวแปร "ความสูง" ที่ประกอบด้วย 3 เทอมเซ็ต.....11
3.4 แสดงภาพฟัซซี่รูปสามเหลี่ยม.....13
3.5 แสดงภาพฟัซซี่รูปสี่เหลี่ยมคางหมู.....14
3.6 แสดงเส้นโค้งรูปตัว S 15
3.7 แสดงภาพฟัซซี่เซ็ตรูประฆังคว่ำ.....15
3.8 แสดงภาพฟังก์ชั่น Gaussian.....16
3.9 แสดงบล็อกไดอะแกรมของแบบจำลองฟัซซี่ลอจิก.....18
3.10 แสดงการคำนวณค่าความเป็นสมาชิกของเทอมเซ็ต COOL และ NORMAL.....20
3.11 แสดงการกำหนดค่าของฟัซซี่เซต.....21
3.12 แสดงตัวแทนของตัวอักษร ว. ที่ผ่านระบบฟัซซี่เซต.....21
4.1 โครงสร้างและส่วนประกอบของเซลล์ประสาท.....24
4.2 แสดงไดอะแกรมของนิวรอลที่สร้างขึ้น (Artificail Neuron).....25
4.3 แสดงโมดุลนิวรอลที่สร้างขึ้นร่วมกับ Activation Function.....26
4.4 แสดงกราฟที่ได้จากสมการซิกมอยด์ลออจิสติกฟังก์ชั่น (Sigmoidal logistic function).....27
4.5 แสดง Hyperbolic Tangent Function.....28
4.6 แสดงลักษณะโครงข่ายประสาทเทียมแบบชั้นเดี่ยว (Sigle-Layer Neural Networks) 29
4.7 แสดงไดอะแกรมของ Backpropagation Neural Networks แบบสองชั้น.....30
4.8 แสดงกราฟในลักษณะ S-shape ของฟังก์ชั่นไบนารีซิกมอยด์.....36
5.1 แสดงลักษณะของพยัญชนะ ว.....37
5.2 ตัวอย่างของตัวอักขระที่ลดขนาดเหลือ 32 x 16 บิต.....38
5.3 แสดงขนาดของตัวอักขระ.....38

สารบัญภาพ (ต่อ)

ภาพที่ หน้า

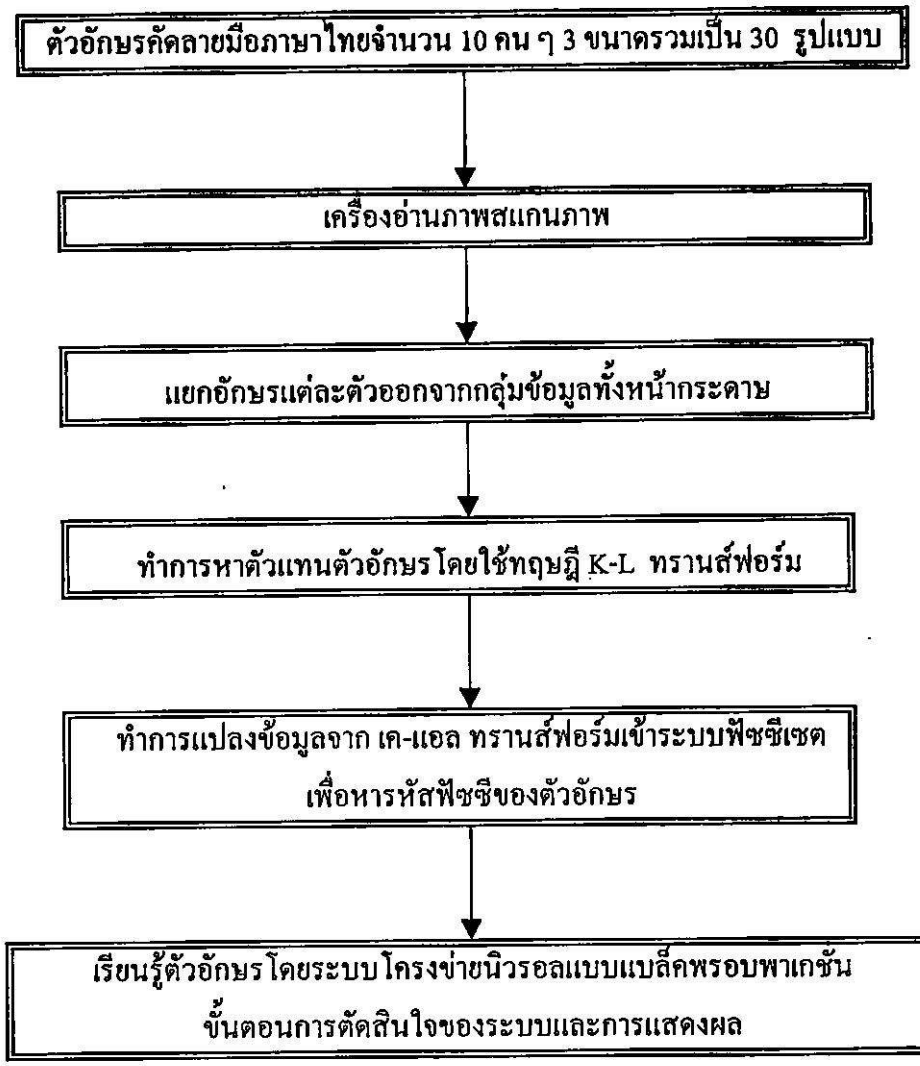
5.4 แสดงเมตริกก่อนทำการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ.....37
5.5 แสดงการแบ่งระดับฟัซซี่สำหรับปรับข้อมูล.....40
5.6 แสดงพยัญชนะ ว ที่ปรับข้อมูลตามระบบฟัซซี่.....40
5.7 แสดงโครงสร้างของโครงข่ายประสาทเทียม.....41
5.8 แสดงโครงสร้างของการทดสอบ.....42
6.1 แสดงภาพของพยัญชนะ ก-ฮ จากผู้คัดลายมือ 1 คน ครั้งที่ 1 43
7.1 แสดงข้อมูลที่ใช้ในการทดสอบ และตัวแทนที่ใช้ในการฝึกสอนของพยัญชนะ ซ.....53

บทที่ 1
บทนำ

1.1 กล่าวนำ

งานวิจัยด้านการเรียนรู้จำตัวอักษรทั้งภาษาอังกฤษ-ภาษาไทย มีทั้งแบบศึกษาวิจัยตัวอักษรตัวพิมพ์-ตัวเขียน ได้มีการศึกษาวิจัยมามากกว่า 15 ปีโดยตัวอักษรที่ป้อนเข้ามาในการเรียนรู้แปลงข้อมูลเป็นลักษณะเลขฐาน 2 ก่อน (binary) (1) จากนั้นก็นำเลขฐาน 2 มาวิเคราะห์ โดยลักษณะของการวิจัยทั่วไปพอจะแบ่งออกเป็น 2 ลักษณะกว้าง ๆ คือ การวิเคราะห์โครงสร้างของตัวอักษร(2) โดยหาลักษณะเด่นของตัวอักษร (feature extraction) (3)(4) และอีกลักษณะหนึ่งเป็นการเก็บข้อมูลเชิงสถิติ (statistical pattern recognition) ของข้อมูลตัวอักษรแล้วนำฟังก์ชั่นการตัดสินใจ (dicition function) มาใช้ในการพิจารณารู้จำตัวอักษร งานวิจัยที่มีการนำโครงข่ายนิวรอล(5) มาใช้ในการรู้จำมากขึ้นเพื่อความรวดเร็วในการตัดสินใจพิจารณาตัวอักษร พร้อมทั้งมีการงานวิจัย ที่ทำการลดขนาดของข้อมูล โดยใช้การแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ (karhunen loeve transform)(6) และการรู้จำตัวอักษรตัวพิมพ์ภาษาไทยโดยใช้วัดความเหมือนแบบฟัซซี(7) เป็นการรู้จำลักษณะตัวพิมพ์ซึ่งผลการทดลองสูงถึง 98 เปอร์เซ็นต์ แต่การนำโครงข่ายนิวรอลไปใช้ในการรู้จำตัวอักษรภาษาไทยนั้นมักจะใช้เวลาไปกับการฝึกสอนให้กับโครงข่ายนิลรอลรู้จำมากเช่น การรู้จำตัววอักษรตัวพิมพ์ภาษาไทยกับต้นแบบ 87 ตัว ใช้เวลาในการฝึกสอนถึง 13 วัน(8)

ดังนั้นงานวิจัยนี้ จึงคิดนำเอาข้อดีของแต่ละวิธีมาทำการวิจัย โดยนำข้อมูลตัวอักษรคัดลายมือภาษาไทยตัวละ 30 รูปแบบ (คนคัดลายมือภาษาไทย 10 คน ๆ ละ 3 ครั้ง) ผ่านเครื่องอ่านภาพ (scanner) จากนั้นนำตัวอักษรซึ่งมีลักษณะข้อมูลเป็นเลขฐาน 2 มาทำการลดขนาดของตัวอักษรโดยการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟเพื่อลดแบบตัวอักษร (font) ให้เหลือตัวแทนตัวอักษรเพียง 1 ตัวในการเรียนรู้ และตัวแทนตัวอักษรคัดลายมือภาษาไทยนั้นไปเข้าระบบฟัซซีเพื่อลดจำนวนฟีเจอร์ (Feature) ก่อนเข้าโครงข่ายนิวรอล แบบแบ็คพรอบพาเกชั่น เพื่อทำการเรียนรู้ตัวอักษรและเก็บคุณลักษณะไว้ใช้ในการทดสอบ ตัวอักษรตัวที่ไม่ทราบมาทำการรู้จำ ซึ่งมีขั้นตอนในการทำงานเป็นบล็อกไดอะแกรมกว้าง ๆ ดังภาพที่ 1.1



ภาพที่ 1.1 แสดงขั้นตอนการทำงานของระบบการรู้จำตัวอักษรคัดลายมือภาษาไทย

การทำงานของระบบการรู้จำตัววอักษรคัดลายมือภาษาไทยนี้ข้อมูลจากภายนอก ซึ่งสามารถรับข้อมูลลายมือเขียนได้โดยเครื่องอ่านสแกนภาพเข้ามาอยู่ในรูปของแฟ้มข้อมูลทางกราฟฟิค ซึ่งจะต้องการแปลงแฟ้มข้อมูลดังกล่าวมาเป็นแฟ้มข้อมูลชนิดเลขฐานสองทำการแยกแยกตัวอักษณจากกลุ่มข้อมูลลายมือเขียนที่รับเข้ามาทีละตัวอักษรแล้วนั้น ทำการเรียนรู้ (Learning) โดยจะให้โครงข่ายนิวรอลทำการเรียนรู้ลายมือเขียน ซึ่งก่อนที่จะทำการให้โครงข่ายนิวรอลเรียนรู้นี้ ในส่วนของวิทยานิพนธ์จะทำการศึกษาการหาค่าเฉลี่ยของตัวอักษรที่จะมาเป็นมาตรฐานไว้ 1 ตัวอักษรโดยการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟและนำผลมากำหนดค่าเป็นฟัซซี่เซตจากการป้อนตัวอักษรลายมือเขียนจาก 30 รูปแบบ แล้วนำมาเทรนโดยใช้โครงข่ายนิลรอล เพื่อเก็บไว้เป็นฐานความรู้ของการตัดสินใจในตัวอักษรอะไรของคอมพิวเตอร์

1.2 วัตถุประสงค์ของงานวิจัย

จากการที่ได้ศึกษาวิทยานิพนธ์ที่ผ่านมาจะต้องใช้เวลาการให้โครงข่ายนิวรอลฝึกสอนเรียนรู้แบบตัวอักษรหลายรูปแบบของตัวอักษรเดียวกันใช้เวลานานมากและหลายครั้งจึงจะสำเร็จเพื่อเป็นการลดเวลาของการทำให้โครงข่ายนิวรอลใช้เวลาในการเรียนรู้น้อยลง และถูกต้องมากขึ้นวิทยานิพนธ์นี้ จึงทำการวิจัย โดยมีวัตถุประสงค์ดังนี้

1. ศึกษาทฤษฎีของการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ มาใช้ในการลดขนาดของข้อมูลตัวอักษรคัดลายมือภาษาไทย ก่อนที่นำข้อมูลไปใช้งานเพื่อลดเวลาในการเรียนรู้ (Training)

2.ศึกษาทฤษฎีของฟัซซี่เซ็ต (Fuzzy set) และนำมาใช้กับข้อมูลตัวอักษรคัดลายมือแบบง่ายเพื่อเพิ่มความถูกต้องให้ระบบการเรียนรู้ให้มากขึ้น

1.3 ข้อตกลงเบื้องต้นในการทำวิทยานิพนธ์

1.3.1 งานวิจัยนี้มาเน้นศึกษาทฤษฎีของคาร์ฮูเน็นเลิฟมาช่วยในการลดขนาดของข้อมูล เพื่อลดเวลาในขั้นตอนการเรียนรู้ข้อมูลและศึกษาทฤษฎีของฟัซซีเซตมาช่วยในการเพิ่มความถูกต้องในการเรียนรู้ข้อมูลของการรู้จำตัวอักษรคัดลายมือภาษาไทย

1.3.2 ตัวอักษรที่ใช้ในงานวิจัยนี้จะต้องเป็นลักษณะคัดลายมือภาษาไทยตัวบรรจงและเขียนบนกระดาษขาวสำหรับเครื่องพิมพ์เลเซอร์โดยใช้ปากกาหมึกซึม

1.3.3 ผลการวิจัยได้จากอุปกรณ์ตามข้อกำหนดคุณลักษณะดังนี้

1.3.3.1 เครื่องคอมพิวเตอร์ Compaq Presario

CPU : Intel Pentium II MMX 166 MHz

RAM : 64 MB

Haddisk : 8 GByte

1.3.3.2 เครื่องอ่านภาพ (Scanner) Hewlett Packard

ScanJet 6100C และกำหนดคุณลักษณะในการอ่านภาพดังนี้

Type: Black and White Drawing

Brightness: 115

Contrast: 135

Scanling: 100%

1.4 เนื้อหาของวิทยานิพนธ์

บทที่ 1 กล่าวถึง บทนำ วัตถุประสงค์ของงานวิจัย และเนื้อหาโดยรวมของวิทยานิพนธ์

บทที่ 2 เป็นทฤษฎีของคาร์ฮูเน็นเลิฟ

บทที่ 3 เป็นทฤษฎีของฟัซซีซึ่งเป็นการอธิบายถึงทฤษฎีทั่วไปและการกำหนดฟัซซีเซตมาใช้เพิ่มความถูกต้องในการเรียนรู้ให้มากขึ้น

บทที่ 4 กล่าวนำทฤษฎีของการเรียนรู้โดยใช้โครงข่ายนิวรอลเน็ตเวิร์คแบบการแพร่ย้อนกลับ

บทที่ 5 กล่าวถึงขั้นตอนการทำงานของระบบ

บทที่ 6 เป็นการทดลอง ลักษณะต่าง ๆ เปรียบเทียบการทดลองหลายรูปแบบ

บทที่ 7 บทสรุป เป็นการทดลองสรุปผลการวิจัยทั้งหมดและข้อเสนอแนะในการทำวิจัย พร้อมข้อจำกัดต่าง ๆ ของงานวิจัย

ภาคผนวก

ภาคผนวก ก. ตัวอย่างของตัวอักษรคัดลายมือภาษาไทย

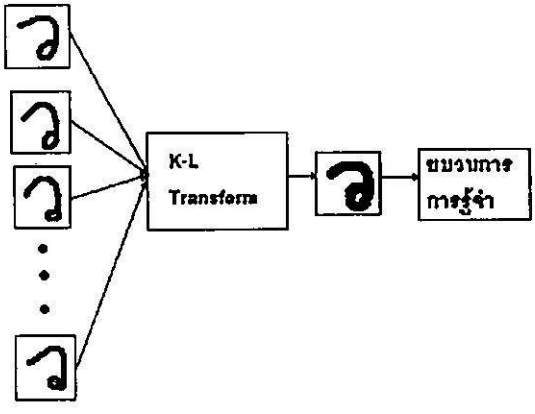
ภาคผนวก ข. ตัวอย่างโปรแกรมที่ใช้ในงานวิจัยนี้

ภาคผนวก ค. ผลงานวิจัยของผู้เขียนที่ได้รับการตีพิมพ์

บทที่ 2

ทฤษฎีของคาร์ฮูเน็นเลิฟ (Karhunen-Loeve)

เนื่องจากลักษระการคัดลายมือของแต่ละบุคคลจะมีความคลาดเคลื่อนจากตัวอักษรคัดลายมือมาตรฐานมากน้อยแตกต่างกัน ทำให้ระบบการรู้จำตัวอักษรคัดลายมือมีจำนวนข้อมูลรูปแบบของตัวอักษรคัดลายมือจำนวนมาก ซึ่งจะส่งผลต่อระยะเวลาการประมวลผลข้อมูล และหน่วยความจำของเครื่องคอมพิวเตอร์ ดังนั้นในงานวิจัยนี้จึงมีวัตถุประสงค์ในการลดจำนวนข้อมูล ที่ใช้ในการประมวลผล โดยการนำข้อมูลตัวอักษรมาผ่านขบวนการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ เพื่อหาตัวแทนที่เก็บลักษณะร่วมของพยัญชนะแต่ละตัว ของตัวอักษระซึ่งรูปแบบต่างกันให้เหลือเพียงข้อมูลที่เป็นตัวแทนตัวเดียว มาผ่านขบวนการการรู้จำตัวอักษรดังแสดงในภาพที่ 2.1



ภาพที่ 2.1 การหาตัวแทนของตัวอักษรคัดลายมือ ว. ในภาษาไทยเพื่อนำไปใช้ในขบวนการการรู้จำ

2.1 การวิเคราะห์รูปแบบตัวอักษรโดยใช้ K-L Tranform

การกระจายแบบ K-L เป็นการกระจายของเวคเตอร์ไปตามขฉากรรมในไอเกนเวคเตอร์ของโควาเลียนเมตริกในกรณีของการจดจำรูปแบบสัมประสิทธิ์ของเมตริก $F_1, F_2, \ldots, F_n$ หมายถึง รูปร่างของตัวอักษรแต่ละตัวซึ่งเป็นเวคเตอร์แบบหนึ่ง (Column Vector) คือหนึ่งตัวอักษรจะแทนด้วย 1 Column แล้วโควาเลียนเมตริกจะคำนวณได้จาก

$$V = (\sum_i F_i F_i^T) / n \qquad (2.1)$$

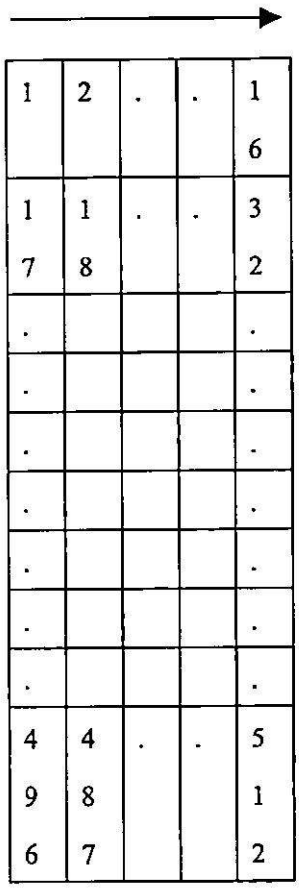
ค่าไอเกน (λ) และไอเกนเวคเตอร์ (P) ของเมตริกนี้คำนวณได้จากสมการต่อไปนี้

$$VP = \lambda P \qquad (2.2)$$

ไอเกนเวคเตอร์ที่มีค่าไอเกนมากที่สุดเป็นส่วนประกอบหลัก เทียบได้กับข้อมูลที่แสดงลักษณะร่วมของแต่ละพยัญชนะ ไอเกนเวคเตอร์นี้จะเก็บไว้เป็นอินพุทของระบบการรู้จำต่อไป

2.1.1 การคำนวณหาค่าไอเกนเวคเตอร์ และค่าไอเกน

ให้ n เป็นจำนวนจุดข้อมูลในแต่ละรูปแบบ F_i , m เป็นจำนวนรูปแบบของกลุ่ม i รูปแบบ F_i จะแทนเวคเตอร์ F_i หรือ G_i ดังภาพที่ 2.2



ภาพที่ 2.2 แสดงตำแหน่งของตัวอักษร 1 ตัว

$$F_I = (f_{i11}, f_{i12}, \ldots f_{i2l}, \ldots f_{i2n}, \ldots f_{imn})^t \qquad (2.3)$$

$$G_I = (g_{i1}, g_{i2}, \ldots . g_{in})^t \qquad (2.4)$$

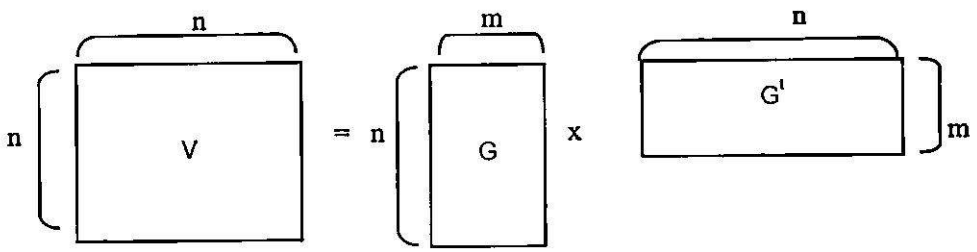
โดยที่ N= m x n เมตริก G หาได้จากสมการต่อไปนี้

$$G = \begin{bmatrix} g_{11}, g_{12}, g_{13}, \ldots\ldots g_{1M} \\ g_{21} \\ g_{31} \\ . \\ . \\ . \\ g_{N1}, g_{N2}, g_{N3}, \ldots\ldots g_{NM} \end{bmatrix} \ldots NxM \qquad (2.5)$$

เมตริก V ที่ใช้ในการคำนวณค่าไอเกน และ ไอเกนเวคเตอร์ ที่ใช้ในงานวิจัยนี้ หาได้จากสมการต่อไปนี้

$$V = G.G^T \qquad (2.6)$$

$$V_{ij} = \sum_{k=1}^{m} g_{ki} \cdot g_{kj} \qquad (2.7)$$



ภาพที่ 2.3 มิติของเมตริก $V = G.G^T$

ค่าไแกนที่ลำดับสูกกว่า m+1 ในเมตริกนี้จะเป็น 0 จึงสามารถเขียนสมการต่อไปนี้ได้

$$\lambda_1 \geq \lambda_2 \geq \ldots\ldots \lambda_m, \lambda_{m+1} = \lambda_{m+2} = \ldots \lambda_n j = 0 \qquad (2.8)$$

$$P_i = (P_{i1}, P_{i2}, \ldots, P_{in})^t \qquad (2.9)$$

บทที่ 3
ทฤษฎีฟัซซี่เซ็ต

ความสัมพันธ์ของสมาชิกภายในกลุ่มแต่ละตัวกับคำจำกัดความของเซตนั้นจะถูกแสดงในลักษณะของระดับความเป็นสมาชิกที่มีค่าอยู่ในช่วง [0, 1] ซึ่งจะแสดงว่าสมาชิกตัวใดเป็นสมาชิกหรือไม่เป็นสมาชิกของเซตที่แสดงถึงเป็นฟัซซี่เซต นั้นอย่างเซตธรรมดา (ซึ่งมีค่าเป็น { 0, 1} ถ้าเรากำหนดให้เป็นเซตเอกภพสัมพัทธ์และฟัซซี่เซต A มีสมาชิกของเซตเป็น $X_1,X_2,X_3,...X_n$ โดยที่ $X_i(i=1,2,3,...,n)$ เป็นค่าฟัซซี่เซต ดังนั้นฟัซซี่เซต A สามารถแสดงในรูปแบบของความสัมพันธ์ในฟัซซี่เซตได้โดย $\mu_A(Xi)$ ความสัมพันธ์เช่นนี้ในทางทฤษฎี ของฟัซซี่เซตจะเรียกว่า ฟังก์ชันความเป็นสมาชิก(Membership Function) และสามารถเขียนเป็นสัญญลักษณ์ทางคณิตศาสตร์ได้โดย

$$\mu_A(Xi) ; U \longrightarrow [0,1] \qquad (3.1)$$

โดยที่ $\mu_A(Xi)$ ก็คือฟังก์ชันความเป็นสมาชิกของเซต A

ฟังก์ชันความเป็นสมาชิกของเซตใดก็จะสอดคล้องกับคำจำกัดความของเซตนั้น และคำจำกัดความ เหล่านี้ก็คือตัวแปรต่างๆ ของฟัซซี่เซตที่อยู่ภายใต้ขอบเขตของเซตเอกภพสัมพัทธ์ หรือระบบใดๆ ที่เราสนใจ

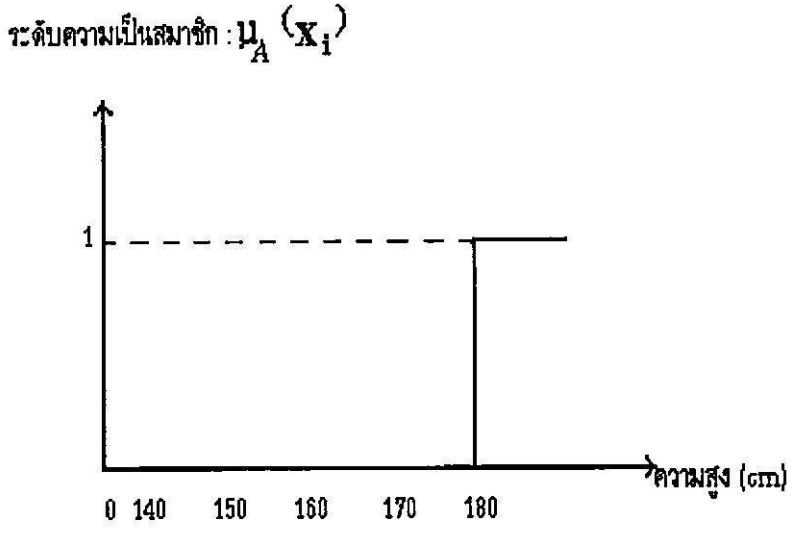
ตัวแปรฟัซซี่ (Fuzzy Variable) ซึ่งก็คือฟัซซี่เซตใดๆ ของระบบที่เราสนใจดังที่ได้กล่าวมาแล้วข้างต้น ตัวอย่างเช่น ถ้าเราพิจารณา "อุณหภูมิ" เป็นตัวแปรฟัซซี่ และค่าของตัวแปรของเซตนี้คือ {ต่ำ, ปานกลาง , สูง} จะเรียกเซตของค่าตัวแปรฟัซซี่นี้ว่าเทอมเซต(Term Set) หรือเซตย่อย นอกจากนี้ตัวแปรฟัซซี่แต่ละตัวอาจจะมีส่วนขยาย (Hedges หรือQualifiers) เพื่อปรับค่าตัวแปรให้มีความยืดหยุ่นมากยิ่งขึ้น เช่น จากค่าของ ตัวแปรอุณหภูมิ "ต่ำ $\longrightarrow$ ต่ำมาก" , "สูง $\longrightarrow$ ค่อนข้างสูง" เป็นต้น คำว่า "มาก" หรือ "ค่อนข้าง" ในที่นี้เป็นส่วนขยายของตัวแปรฟัซซี่ ซึ่งในทางปฏิบัติสามารถแทนด้วยฟังก์ชันทางคณิตศาสตร์

3.1 การแทนข้อมูลโดยฟัซซี่เซต

การแทนข้อมูลโดยฟัซซี่เซต ตัวอย่างของเหตุการณ์ที่เกิดขึ้นในชีวิตประจำวันคือเราจะพิจารณาความสูง โดย "ความสูง" ในที่นี้จะเป็นตัวแปรฟัซซี่เซต และเพื่อให้เข้าใจความหมายของฟัซซี่เซตได้เด่นชัดยิ่งขึ้นเราจะเปรียบเทียบการแทนข้อมูลของฟัซซี่เซตกับเซตธรรมดา การแปลงข้อมูลเพื่อคำนวณค่าความเป็นสมาชิกของเซตธรรมดาจะต้องมีการกำหนดค่าเทรชโฮลด์เพื่อจะนำมา

เป็นตัว ตัดสินว่าสมาชิกตัวใด ที่เป็นสมาชิกเซตที่กำลังพิจารณาอยู่บ้าง สำหรับในตัวอย่างของเซตความสูงของผู้ชาย สมมุติอยู่ในช่วง 140 – 180 cm (โดเมนของเซตเอกภพสัมพัทธ์) แต่ในที่นี้เราสนใจคนที่สูง ถ้าเรากำหนดให้เซต A เป็นเซตของคนสูง ("สูง" ในกรณีของฟัซซี่เซตจะเป็นเทอมเซต) ดังนั้นเซต A ในกรณีของเซตธรรมดาจะต้องมีค่าขีดเริ่มเปลี่ยนหรือค่าเทรชโฮลด์ (Thershold) ซึ่งในที่นี้จะ กำหนดให้เท่ากับ 180cm นั่นคือถ้า คนใดมีความสูงมากกว่าหรือเท่ากับ 180 cm จะถือว่าคนนั้นสูง (เป็นสมาชิกของเซต A) ซึ่งสามารถเขียนเป็นฟังก์ชันคุณลักษณะ (Characteristic Function) หรือฟังก์ชัน ความเป็นสมาชิกได้ดังนี้

$$\mu_A(X_I)\,;\ \{X_I \geq 180\} \qquad (3.2)$$



ภาพที่ 3.1 แสดงค่าความเป็นสมาชิกของเซต A ในกรณีที่เป็นเซตธรรมดา

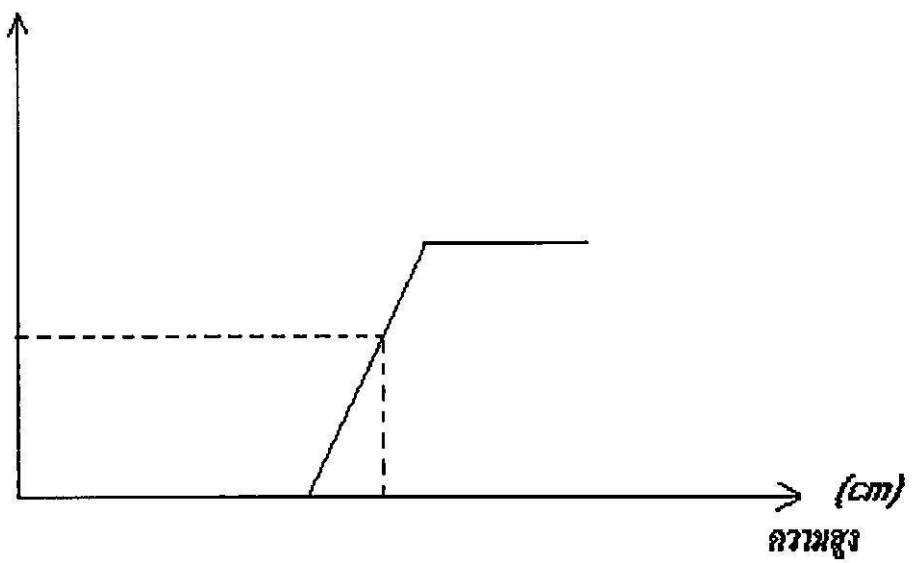
โดยที่ X_I คือความสูงของผู้ชาย คนที่ I

ในสมการที่ (3.2) สามารถนำมาเขียนเป็นกราฟของความเป็นสมาชิกได้ดังภาพที่ 7

ตามภาพที่ 3.1 บนแนวแกนตั้งแทนค่าระดับความเป็นสมาชิกที่มีค่าเป็น 0 (ไม่เป็นสมาชิกของเซต A)และ 1 (เป็นสมาชิกของเซต A) และจะเห็นว่าคนที่มีความสูงมากกว่าหรือเท่ากับ 180 cm ขึ้นไปเท่านั้นจึงจะถือว่าผู้ชายคนนั้นสูง นอกจากนั้นจะไม่ใช่สมาชิกของเซต A

ฟัซซี่เซตฟังก์ชันความเป็นสมาชิกจะมีความยืดหยุ่นกว่า และสอดคล้องกับความเป็นจริงมากกว่าเซตธรรมดา ดังในสมการที่ (3.3)

$$\mu_A(X_i) = \begin{cases} \dfrac{1}{(180-170)}(X_i - 170); & \text{ถ้า } 170 \leq 180 \\ 1; & \text{ถ้า } X_i > 170 \\ 0; & \text{ในกรณีอื่นๆ} \end{cases} \qquad (3.3)$$



ภาพที่ 3.2 แสดงค่าความเป็นสมาชิกของเทอมเซต A ในกรณีที่เป็นฟัซซี่เซต

จากภาพที่ 3.1 และภาพที่ 3.2 จะเห็นว่ากราฟแสดงความเป็นสมาชิกของฟัซซี่เซตจะมีความต่อเนื่องในค่าระดับความเป็นสมาชิกในช่วง [0,1] เช่น คนที่มีความสูง175 cm ก็จะถือว่าเป็นคนสูงเช่นกัน แต่เขาจะมี

ระดับความเป็นสมาชิกของเซตของคนสูงเท่ากับ 0.5 ถ้าเปรียบเทียบกับเซตธรรมดาคนที่มีความสูง 175 cm จะไม่เป็นสมาชิกของเซต A ทั้งที่เขามีความสูงแตกต่างจากคนสูงไม่กี่เซนติเมตร

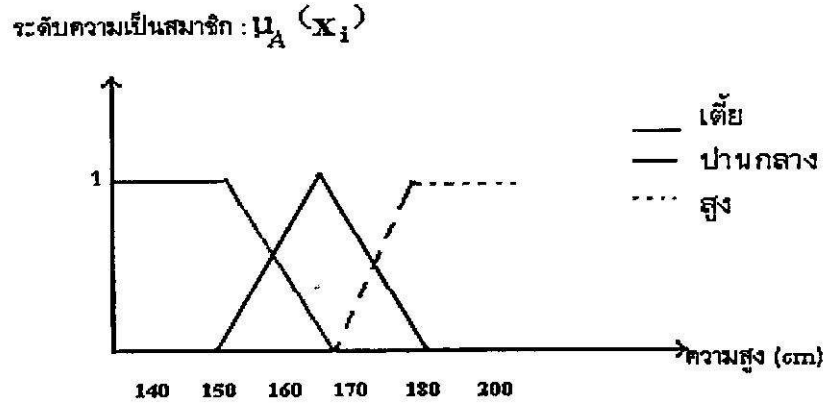
เท่านั้น หรือแม้แต่คนที่มีความสูง 179 cm ซึ่งมีความสูงต่างจากคนที่ สูงต่างจากคนที่ถือว่าสูงเพียง 1 cm เท่านั้น แต่ในเซตธรรมดาจะถือว่าเขาไม่ใช่คนสูงซึ่งจะไม่ตรงกับความเป็นจริง แต่สำหรับการแทนข้อมูลโดยฟัซซี่เซตจะถือว่าคนนี้เป็นเซตของคนสูงด้วยค่าระดับความเป็นสมาชิกท่ากับ 0.9 (คือเกือบจะเป็นคนสูงเลยทีเดียว)

3.2 ฟังก์ชันความเป็นสมาชิก

เรามาพิจรณา คุณสมบัติบางประการของฟัซซี่เซต ก่อนที่จะพิจารณาถึงฟังก์ชันความเป็นสมาชิกชนิดต่างๆ ฟังก์ชันความเป็นสมาชิกหรือฟังก์ชันคุณลักษณะในกรณีของฟัซซี่เซต เช่น

3.2.1 เซตเอกภพสัมพัทธ์

เมื่อพิจารณาถึงกราฟภาพที่ 3.1แกนนอนของกราฟจะแทนโดเมนของฟัซซี่เซต ส่วนแกนตั้งจะแทน ค่าระดับความเป็นสมาชิก แต่ในกราฟภาพที่ 8 มีเทอมเซตเพียงเทอมเดียว ซึ่งเป็นเทอมของคนสูงเท่านั้นเพื่อให้ครอบคลุมกลุ่มประชากรที่เราสนใจจึงเพิ่มเทอมเซตอีกสองเทอมคือ เทอมเซตของคน "เตี้ย" และ เทอมเซตของคนที่มีความสูง "ปานกลาง" ดังภาพที่ 9 ซึ่งในกรณีที่โดเมนของตัวแปรฟัซซี่เซตที่ครอบคลุมประชากรทั้งหมดที่ทำการศึกษา เราจะเรียกโดเมนของตัวแปรนี้ว่า เซตเอกภพสัมพัทธ์ และในแต่ละเทอม เซตก็จะมีโดเมนของตัวอย่างเช่น เทอมเซตของคนเตี้ยจะมีโดเมนอยู่ในช่วง140 -170 cm, เทอมเซตของคนที่มี ความสูงปานกลางจะมีโดเมนอยู่ในช่วง 150 - 180 cm, และเทอมเซตของคนที่สูงจะโดเมนอยู่ในช่วง 160 - 200 cm



ภาพที่ 3.3 แสดงกราฟฟังก์ชันความเป็นสมาชิกของตัวแปร " ความสูง " ที่ประกอบด้วย 3 เทอมเซต

3.2.2 ซัพพอร์ตเซต

ในกรณีที่ $X_i (\forall_i)$ ในบางเทอมเซตมีค่าน้อยกว่าโดเมนของมัน เช่นในเทอมของคนเตี้ยซึ่งมีโดเมนอยู่ในช่วง [140-170] เมื่อทำการวัดความสูงของประชากรกลุ่มนี้ปรากฏว่าความสูงที่วัดได้จริง ๆ ของพวกเขาอยู่ในช่วง [145 - 170] ซึ่งช่วงของข้อมูลที่วัดได้จริงๆ จะเรียกว่า ซัพพอร์ตเซต (Support Set) ของเทอมนั้นๆ สำหรับซัพพอร์ตเซตจะมีประโยชน์ในการปรับขนาดของเทอมเซตโดยอัตโนมัติ

3.2.3 การเทรชโฮลด์ฟัซซี่เซต

การเทรชโฮลด์หรืออัลฟาคัท (Alpha-Cut, หรือ $\propto$ - Cut) ของฟัซซี่เซตจะมีความสัมพันธ์กับซัพพอร์ตเซต คือค่าระดับของ $\propto$ ($0 \leq \propto \leq 1$)จะเป็นค่าเทรชโฮลด์ภายในขอบเขตโดเมนของเทอมเซตที่ค่า $\propto$ เป็นค่าขอบเขตล่างของซัพพอร์ตเซต หรืออาจจะเป็นค่าใดๆ ที่เราสนใจในช่วง [0,1] $\propto$ - Cut มีอยู่สองชนิดด้วยกันคือ

3.2.3.1. Strong $\propto$ - Cut ซึ่งกำหนดได้โดย

$$\mu_A(X_i) = \begin{cases} \mu_A(X_i) \text{ ; ถ้า } \mu_A(X_i) \geq \propto \\ 0 \quad \text{ ; ในกรณีอื่น ๆ} \end{cases} \tag{3.4}$$

3.2.3.2. Weak $\propto$ - Cut สามารถกำหนดได้โดย

$$\mu_A(X_i) = \begin{cases} \mu_A(X_i) \text{ ; ถ้า } \mu_A(X_i) > \propto \\ 0 \quad \text{ ; ในกรณีอื่น ๆ} \end{cases} \tag{3.5}$$

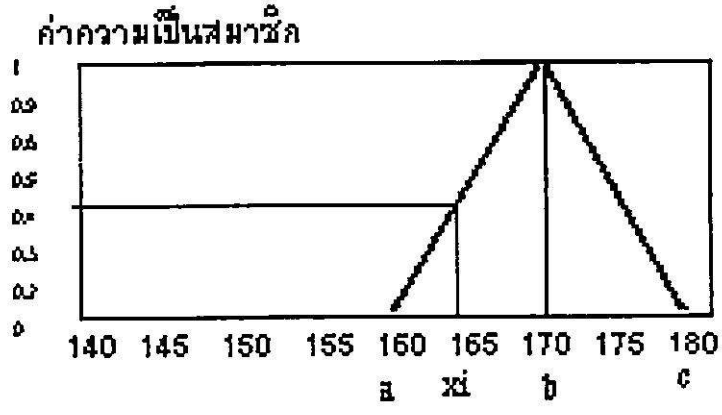
เมื่อ A เป็นเทอมเซตใดๆ ประโยชน์ของ $\propto$ - Cut มีอยู่สองประการด้วยกัน อย่างแรกคือ Strong $\propto$ - Cut ใช้ในการกำหนดซัพพอร์ต อย่างที่สองคือทำให้มีความยืดหยุ่นสำหรับระบบการควบคุมที่ใช้ฟัซซี่เซตในกรณีที่มีการอินเตอร์เซกชันกันหลายเทอมซึ่งจะได้อธิบายในหัวข้อของกฎฟัซซี่เซตต่อไป ตอนนี้เราจะพิจารณาถึงฟังก์ชันความเป็นสมาชิกแบบต่างๆ ที่จะใช้ในการประมาณค่าระดับความเป็นสมาชิกของแต่ละเทอมเซต โดยมีทั้งแบบที่ประมาณค่าเป็นเชิงเส้น เช่น รูปสามเหลี่ยม,รูปสี่เหลี่ยมคางหมู และแบบที่ประมาณค่าโดยฟังก์ชันต่อเนื่อง เช่น S,S^{-1} , ระฆังคว่ำ เป็นต้น ซึ่งมีรายละเอียดดังนี้

- สมการเชิงเส้น

ฟังก์ชันความเป็นสมาชิกแบบสมการเชิงเส้นที่ใช้ในแปลงค่าความสัมพันธ์ของโดเมนไปยังเรนจ์ สำหรับข้อมูลที่มีความสัมพันธ์เป็นเชิงเส้น ดังสมการที่ (3.3)

- รูปสามเหลี่ยม

การกำหนดฟังก์ชันความเป็นสมาชิกโดยใช้รูปสามเหลี่ยม นิยมใช้กันมากในระบบควบคุมที่ใช้ฟัซซี่เซต เทอมเซตที่กำหนดโดยฟังก์ชันนี้จะต้องมีค่าที่เหมาะสมที่สุดอยู่เพียงค่าเดียวที่ทำให้ค่าความเป็นสมาชิกเท่ากับหนึ่ง ส่วนค่าอื่นๆ จะมีความเป็นสมาชิกลดลงเรื่อยๆ เมื่อยิ่งห่างจากค่านี้มากขึ้น ในตัวอย่างของตัวแปรความสูง เทอมเซต "ปานกลาง" มีฟังก์ชันความเป็นสมาชิกเป็นรูปสามเหลี่ยม ถ้ากำหนดให้ $a \leq b \leq c$ เมื่อ a,b และ c เป็นเลขจำนวนจริงใดๆ ดังนั้นสมการของฟังก์ชันความเป็นสมาชิกของรูปสามเหลี่ยมสามารถกำหนดได้ดังนี้



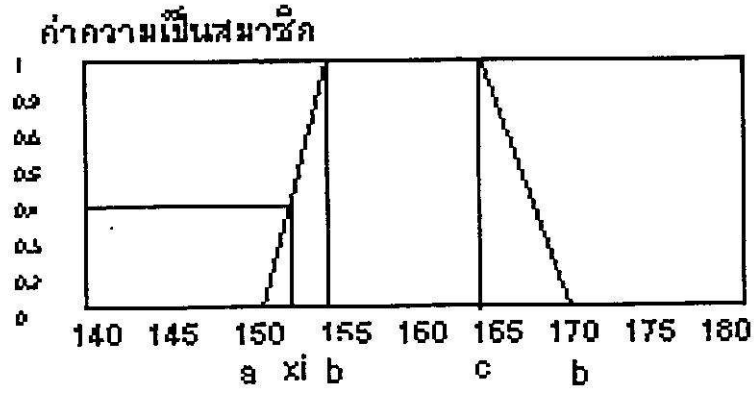
ภาพที่ 3.4 แสดงภาพฟัซซี่รูปสามเหลี่ยม

$$\mu_A(X_i) = \begin{cases} \frac{1}{b-a}(X_i-a) \text{ ; ถ้า } a \leq X_i \leq b \text{ และ } a \neq b \\ \frac{1}{c-b}(c-X_i) \text{; ถ้า } a \leq X_i \leq b \text{ และ } a \neq b \\ 0 \text{ ; ในกรณีอื่น ๆ} \end{cases} \quad (3.6)$$

- รูปสี่เหลี่ยมคางหมู

สำหรับฟังก์ชันความเป็นสมาชิกที่กำหนดด้วยสมการรูปสี่เหลี่ยมคางหมู นิยมใช้ในระบบควบคุมที่ใช้ฟัซซี่เซตเช่นกัน แต่จะพบน้อยกว่าฟังก์ชันความเป็นสมาชิกที่กำหนดด้วยสมการรูปสามเหลี่ยม ฟังก์ชันความเป็นสมาชิกที่จะกำหนดด้วยสมการรูปสี่เหลี่ยมคางหมูจะต้องมีช่วงของค่าที่เหมาะสมมากที่สุดอยู่กลุ่มหนึ่งที่ทำให้ค่าความเป็นสมาชิกของเทอมเซตนั้นๆ นอกจากนั้นจะมี

ค่าความเป็นสมาชิกน้อยลงเรื่อยๆ เมื่อยิ่งห่างจากข้อมูลกลุ่มนี้ ถ้ากำหนดให้ $a \leq b \leq c$ เมื่อ a,b,c และ d เป็นเลขจำนวนจริงใดๆดังนั้นสมการของฟังก์ชันความเป็นสมาชิกของรูปสี่เหลี่ยมคางหมู สามารถกำหนดได้ดังนี้

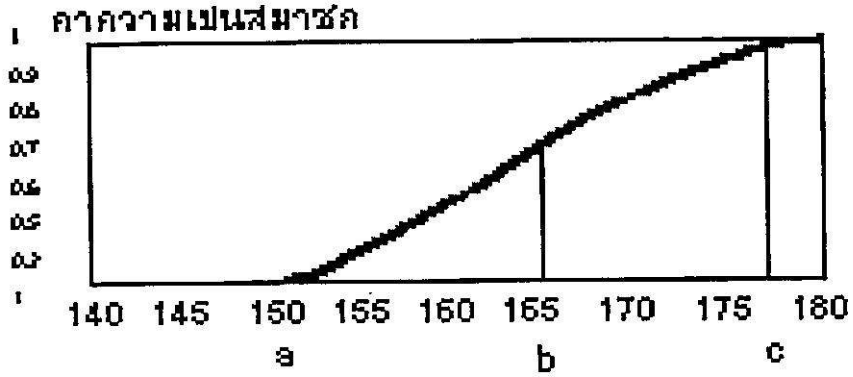


ภาพที่ 3.5 แสดงภาพฟัซซีรูปสี่เหลี่ยมคางหมู

$$\mu_A(X_i) = \begin{cases} \frac{1}{b-a}(X_i-a) & ; \text{ถ้า } a \leq X_i \leq b \text{ และ } a \neq b \\ 1 & ; \text{ถ้า } b \leq X_i \leq c \\ \frac{1}{d-c}(c-X_i) & ; \text{ถ้า } c \leq X_i \leq d \text{ และ } c \neq d \\ 0 & ; \text{ในกรณีอื่น ๆ} \end{cases} \quad (3.7)$$

- เส้นโค้งรูปตัว S

ฟังก์ชันความเป็นสมาชิกที่กำหนดด้วยเส้นโค้งรูปตัว S (S-Curve,Sigmoid หรือ Logistic) เป็นการกำหนดความสัมพันธ์ระหว่างโดเมนไปยังเรนจ์แบบไม่เป็นเชิงเส้น ตัวแปรที่เหมาะที่จะใช้ฟังก์ชันนี้คือพวกอายุการใช้งานของอุปกรณ์ต่างๆ, ตัวแปรส่วนใหญ่ที่ใช้ในทางการประมวลผลภาพ เป็นต้น ซึ่งตัวแปรพวกนี้มักจะมีความสัมพันธ์แบบไม่เป็นเชิงเส้นที่เป็นรูปตัว S หรือ S^{-1}ถ้ากำหนดให้ $a \leq b \leq c$ เมื่อ a,b และ c เป็นเลขจำนวนจริงใดๆดังนั้นสมการของฟังก์ชันความเป็นสมาชิกของเส้นโค้งรูปตัว S สามารถกำหนดได้ดังสมการที่ (3.8) และ S^{-1} สามารถคำนวณได้โดยหาค่าคอมพลีเมนต์ของสมการที่ (3.8) ($S^{-1} = 1 - \mu_A(X_i)$)โดยที่ $\mu_A(X_i)$ ได้จากสมการที่ (3.8)

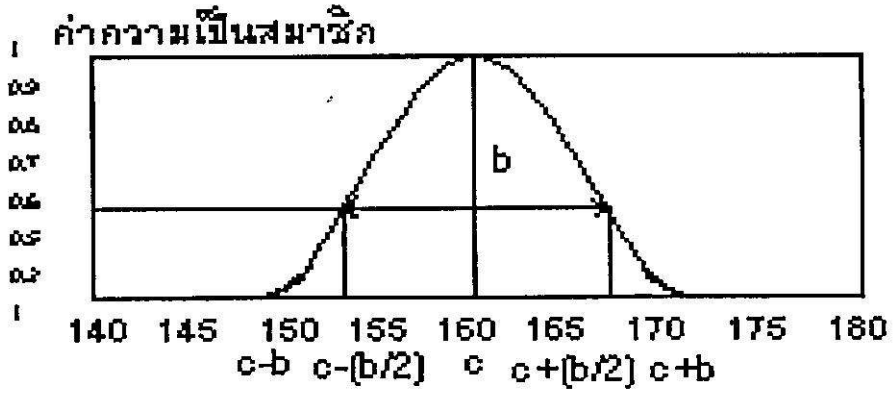


ภาพที่ 3.6 แสดงเส้นโค้งรูปตัว S

$$\mu_A(X_I) = \begin{cases} 0 & ; \text{ถ้า } X_I \le a \\ 2((X_I - a)/(c-a))^2 & ; \text{ถ้า } a < X_I \le b \\ 1-2((X_I - a)/(c-a))^2 & ; \text{ถ้า } b < X_I \le c \\ 1 & ; \text{ถ้า } X_I > c \end{cases} \qquad (3.8)$$

- รูประฆังคว่ำ

เทอมเซตที่ใช้ฟังก์ชันความเป็นสมาชิกแบบรูประฆังคว่ำ จะมีคุณสมบัติของข้อมูลคล้ายกับสมการรูปสามเหลี่ยม แต่ข้อมูลของโดเมนที่แปลงไปยังเรนจ์จะมีความสัมพันธ์ในลักษณะไม่เป็นเชิงเส้น สมการของรูประฆังคว่ำมีด้วยกันหลายสมการแต่ที่ใช้กันมากคือ ฟังก์ชัน π [1, 30], ฟังก์ชัน Gaussian และฟังก์ชัน Beta ซึ่งสามารถกำหนดได้ดังนี้



ภาพที่ 3.7 แสดงค่าฟัซซีเซตรูประฆังคว่ำ

-ฟังก์ชัน คำนวณได้จากสมการที่ (3.8)

$$S(X_I;a,b,c) = \begin{cases} 0 \ ; \text{ถ้า} & X_I \le a \\ 2((X_I - a)/(c-a))^2; \text{ถ้า} & a < X_I \le b \\ 1-2((X_I - a)/(c-a))^2; \text{ถ้า} & b < X_I \le c \\ 1 \ ; \text{ถ้า} & X_I > c \end{cases} \qquad (3.8)$$

โดยที่ $b = \frac{c-a}{2}$

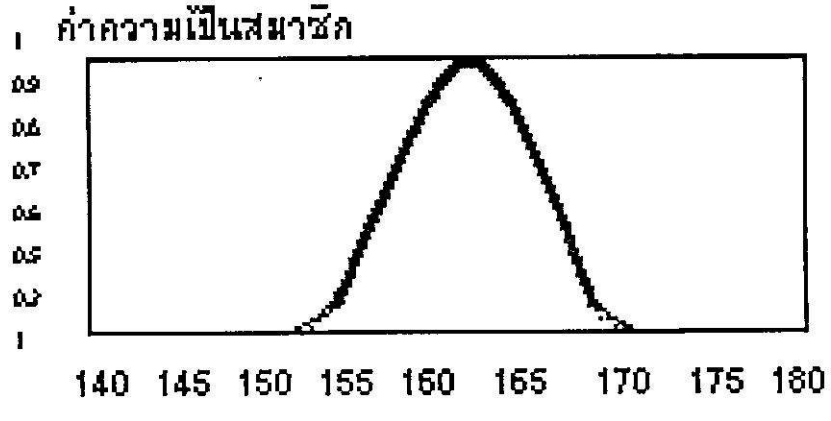
ดังนั้นฟังก์ชัน π สามารถกำหนดได้โดย

$$\mu_A(X_I) = \begin{cases} S(X_i ;c-b, c-b/2, c) \ ; & \text{ในกรณีที่ } X_I \le c \\ 1 - S(X_i ;c-b, c-b/2, c+b) \ ; & \text{ในกรณีที่ } X_I > c \end{cases} \qquad (3.9)$$

- ฟังก์ชัน Gaussian

$$\mu_A(X_I) = e^{K(x-xi)2} \qquad (3.10)$$

โดยที่ $K = 1/(2\sigma)^2$ ซึ่งเป็นโดเมนของฟัซซี่เซต A ที่มีค่าเฉลี่ยเป็น $\bar{x}$ และส่วนเบี่ยงเบนมาตรฐาน σ



ภาพที่ 3.8 แสดงค่าฟังก์ชั่น Gaussian

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

- ฟังก์ชัน Beta

ฟังก์ชันนี้จะมีคุณสมบัติคล้ายกับสมการรูปสี่เหลี่ยมคางหมู คือจะมีช่วงของข้อมูลที่เหมาะสมกับเทอมเซตนั้นๆ อยู่กลุ่มหนึ่งที่ทำให้ค่าความเป็นสมาชิกของเทอมเซตมีค่าเข้าใกล้ 1 ซึ่งข้อมูลกลุ่มนี้จะมีความกว้างของช่วงเท่ากับค่า β ซึ่งถ้าค่า $\sigma \rightarrow \infty$ จะทำให้ข้อมูลในช่วง β มีค่าเข้าใกล้ 1 มากยิ่งขึ้นโดยมีจุดศูนย์กลางอยู่ที่ $\overline{X}$ ดังแสดงในสมการที่ (3.11)

$$\mu_A(X_i) = 1/(1+(X_i - \overline{X})/\beta)^{\rho} \qquad (3.11)$$

3.3 ตัวดำเนินการของฟัซซี่เซต

ตัวดำเนินการของฟัซซี่เซตส่วนใหญ่แล้วจะมีตัวดำเนินการคล้ายกับเซตธรรมดา เช่น อินเตอร์เซกชัน ยูเนียน คอมพลีเมนต์ เป็นต้น ซึ่งตัวดำเนินการเหล่านี้จะใช้ในการอนุมานฟัซซี่เซตเพื่อประโยชน์ในการตัดสินใจโดยการประมาณค่าของตัวแปร ในที่นี้เราจะเสนอตัวดำเนินการของฟัซซี่เซตบางส่วนเฉพาะที่ในใช้แบบจำลองการตัดสินใจของระบบควบคุมโดยใช้ฟัซซี่เซตเท่านั้น ส่วนตัวดำเนินการอื่นของฟัซซี่เซต

กำหนดให้ U เป็นเซตเอกภพสัมพันธ์ , A , B เป็นเทอมเซตใดๆ ที่อยู่ใน U และตัวดำเนินการพื้นฐานต่างๆ สามารถกำหนดได้ดังนี้

- เซตย่อย

  A เป็นเซตย่อยของ B $(A \subseteq B)$ ก็ต่อเมื่อ

  $$\mu_A(X_i) \leq \mu_B(X_i) ; \forall X_i \in U \qquad (3.12)$$

- คอมพลีเมนต์

  A เป็นคอมพลีเมนต์ของ B $(A=B')$ ก็ต่อเมื่อ

  $$\mu_A(X_i) = \mu_{B'}(X_i) = 1 - \mu_B(X_i) ; \quad \forall X_i \in U \qquad (3.13)$$

- อินเตอร์เซกชันของ A และ B $(A \cap B)$ กำหนดได้โดย

  $$\mu_{A \cap B}(X_i) = Min\, \mu_A(X_i), \mu_B(X_i) ; \forall X_i \in U \qquad (3.14)$$

- ยูเนียนของ A กับ B $(A \cup B)$ กำหนดได้โดย

  $$\mu_{A \cup B}(X_i) = Max\, \mu_A(X_i), \mu_B(X_i) ; \forall X_i \in U \qquad (3.15)$$

- A ยกกำลังด้วยค่า $\propto$ $(A^{\propto})$ กำหนดได้โดย

  $$\mu A^{\propto}(X_i) = (\mu_A(X_i))^{\propto} ; \forall X_i \in U \qquad (3.16)$$

24733

- Concentration ของเซต A กำหนดได้โดย

$$\text{CON}(A) \Rightarrow \mu_{\text{CON}(A)}(X_i) = (\mu_A(X_i))^2 ; \forall X_i \in U \qquad (3.17)$$

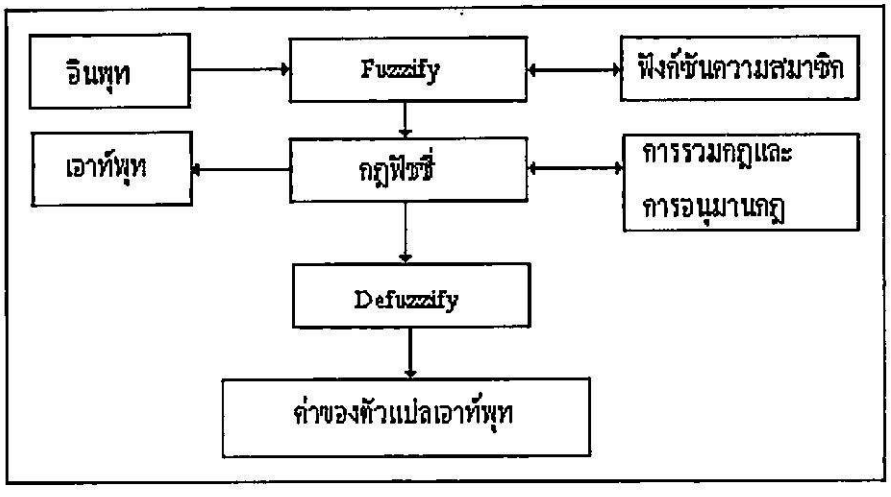
- Dilation ของเซต A กำหนดได้โดย

$$\text{DIL}(A) \Rightarrow \mu_{\text{DIL}(A)}(X_i) = (\mu_A(X_i))^2 ; \forall X_i \in U \qquad (3.18)$$

- Contrast Intensification ของเซต A กำหนดได้โดย

$$\text{INT}(A) \Rightarrow \mu_{INT(A)}(X_i) = \begin{cases} 2(\mu_A(X_i))^2 & ; \text{ถ้า } 0 \le (\mu_A(X_i) \le 0.5 \\ 1-2(1-\mu_A(X_i))^2 & ; \text{ถ้า } 0.5 \le (\mu_A(X_i) \le 1 \end{cases} \qquad (3.19)$$

3.4 การเรียนรู้ของฟัซซี่เซต



ภาพที่ 3.9 แสดงบล๊อกไดอะแกรมของแบบจำลองฟัซซี่ลอจิก

- อินพุทและเอาท์พุท

แบบจำลองฟัซซี่ลอจิกก็คือ ตัวแปรของฟัซซี่เซต ซึ่งตัวแปรแต่ละตัวอาจจะมีเทอมเซตได้หลายเทอม อินพุทซึ่งเป็นตัวแปรของระบบ (State Variables) อาจจะมีกี่ตัวก็ได้ซึ่งจะขึ้นอยู่กับระบบ สำหรับตัวแปรของเอาท์พุทหรือที่เรียกว่า Action Variables) ซึ่งเป็นตัวแปรที่ถูกประมาณค่าโดยแบบจำลองของฟัซซี่ลอจิกเพื่อนำไปใช้ในการออกแบบกฎฟัซซี่เพื่อใช้ในการควบคุมระบบต่อไป ปัญหาที่เกิดขึ้นในขั้นตอนนี้ก็คือ การกำหนดจำนวนเทอมเซตของแต่ละตัวแปรว่าควรจะมีสักกี่

เทอม จึงจะทำให้การอนุมานค่าของตัวแปรเอาท์พุทมีความใกล้เคียงค่าของตัวแปรที่แท้จริงมากที่สุด ซึ่งจุดนี้ผู้ออกแบบ แบบจำลองฟัซซี่จะต้องมีความรู้เกี่ยวกับระบบที่กำลังออกแบบอยู่พอสมควร

- Fuzzify

การแปลงค่าของตัวแปรของอินพุทไปเป็นค่าระดับความเป็นสมาชิกของแต่ละเทอมเซต โดยใช้ฟังก์ชันความเป็นสมาชิก การเลือกใช้ฟังก์ชันความเป็นสมาชิก เราต้องเลือกให้เหมาะสมกับคุณสมบัติของตัวแปร เช่นถ้าตัวแปรมีคุณสมบัติเป็นเชิงเส้น จะต้องเลือกฟังก์ชันความเป็นสมาชิกที่เป็นเชิงเส้นด้วย ซึ่งอาจจะเป็นสมการรูปสามเหลี่ยม และรูปสี่เหลี่ยมคางหมู ส่วนตัวแปรที่มีคุณสมบัติไม่เป็นเชิงเส้นก็ต้องให้ฟังก์ชันความเป็นสมาชิกที่ไม่เป็นเชิงเส้นด้วยเช่น ฟังก์ชันรูปตัว S, π, Beta, และ Guassian

- Fuzzy Rules

จะประกอบด้วยประพจน์ของฟัซซี่ โดยกฎฟัซซี่จะอยู่ในรูปของ IF-THEN เช่น

IF X is small THEN Y is positive

ที่ X และ Y เป็นตัวแปรฟัซซี่ ส่วน small และ positive เป็นเทอมเซตที่สอดคล้องกับตัวแปร X,Y และประพจน์ของฟัซซี่เซตคือ X is small และ Y is positive ประพจน์ที่ตามหลัง IF จะเป็นส่วนของเงื่อนไขที่เรียกว่า Antecedent ซึ่งก็คือส่วนของอินพุทของแบบจำลองฟัซซี่ลอจิกนั่นเอง และประพจน์ที่ตามหลัง THEN จะเรียกว่า Consequent หรือส่วนของเอาท์พุทของแบบจำลองฟัซซี่ลอจิก Antecedent อาจจะประกอบด้วยหลายประพจน์ ซึ่งจำนวนของประพจน์จะขึ้นอยู่กับจำนวนตัวแปรของอินพุทและจำนวนของเทอมเซตของตัวแปรแต่ละตัว เช่นถ้าอินพุทมีสองตัวแปรในส่วนของ Antecedent จะมีสองประพจน์เชื่อมต่อกันในกฎหนึ่งกฎ และในการเชื่อมต่อกันของประพจน์จะต้องมีตัวเชื่อมซึ่งในแบบจำลองฟัซซี่ลอจิกจะมีตัวเชื่อมอยู่สองตัวด้วยกันคือ ยูเนียน , และอินเตอร์เซกชัน เช่น ในกรณีที่เชื่อมด้วยยูเนียน

IF X is small OR T is cool THEN Y is positive

และในกรณีที่เชื่อมด้วยอินเตอร์เซกชัน

IF X is small AND T is cool THEN Y is positive

นอกจากนี้ในกฎฟัซซี่เทอมเซตอาจจะมี Hedge มาขยายเช่น very และ large

IF X is very small THEN Y is large positive

สำหรับจำนวนกฎในแบบจำลองฟัซซี่ลอจิกของระบบหนึ่งๆ จะขึ้นอยู่กับจำนวนของตัวแปรอินพุทและจำนวนเทอมของแต่ละตัวแปร

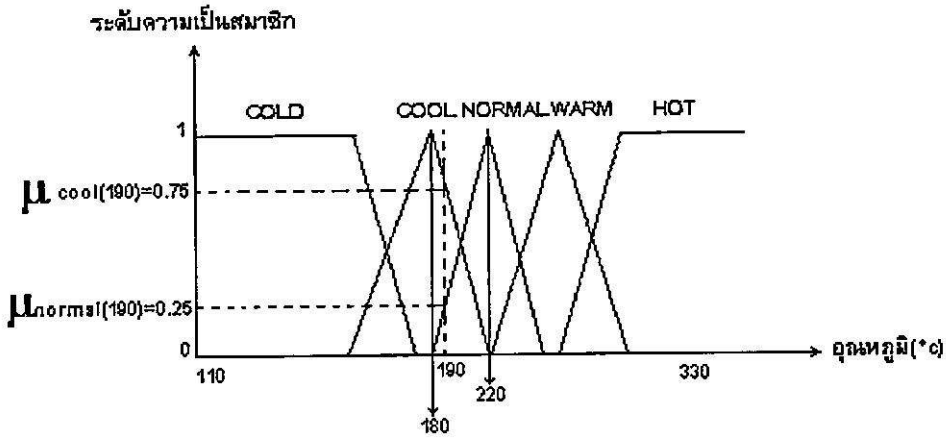
- Defuzzify

Defuzzification เป็นกระบวนการประมาณค่าของตัวแปรเอาท์พุท โดยการอนุมานจากค่าความจริงในส่วนของ Antecedent ของแบบจำลองฟัซซี่ลอจิก โดยค่าที่ได้จะเป็นเลขจำนวนจริงที่อยู่ในโดเมนของตัวแปรเอาท์พุท และค่านี้จะเป็นค่าคาดหวัง (Expected Value) ของตัวแปรฟัซซี่เซต ที่

จะนำไปใช้ในการควบคุมระบบต่อไป สำหรับการหาค่าคาดหวังของฟัซซี่เซตโดยการ Defuzzification นั้นมีอยู่หลายวิธีด้วยกันคือ

- กำหนดจากค่าความจริงที่มีค่าต่ำสุด(เฉพาะเทอมเซตที่เป็นสมาชิกเท่านั้น $\mu_A(X_i) > 0$ เมื่อ A เป็นเทอมเซตใดๆ) ของ Antecedent จากตัวอย่างแบบจำลองฟัซซี่ลอจิกที่ใช้กับระบบ Stream Turbine ในขั้นตอนของกฎฟัซซี่สามารถประมาณค่า I ที่เวลา t ใดๆ ได้ดังนี้

สมมุติอุณหภูมิที่วัดได้ที่เวลา t เท่ากับ 190° C และความดันที่เวลาเดียวกับที่วัดอุณหภูมิเท่ากับ 100 KgmM $^{-2}$ จากกฎฟัซซี่เทอมเซต COOL และ NORMAL ที่ถูกนำมาพิจารณา ดังนั้นเราสามารถประมาณค่าระดับการเปิดปิดวาวล์ ดังภาพที่ 14



ภาพที่ 3.10 แสดงการคำนวณค่าความเป็นสมาชิกของเทอมเซต COOL และ NORMAL

ที่อุณหภูมิ 190 C ไม่ได้เป็นสมาชิกของเทอมเซต COLD, WARM, และ HOT สำหรับความดันที่ 100 Kgm M – 2 นั้นจากกฎฟัซซี่จะมีเทอมเซต LOW และ OK ที่ถูกนำมาพิจารณา

- กำหนดจากค่าความจริงที่มีค่าสูงสุดของ Antecedent จากขั้นตอนการประมาณค่าตัวแปรเอาท์พุท จะมีความแตกต่างจากการ Defuzzification ขั้นตอนการคำนวณค่า $\mu_{INFERANCE}$= เท่านั้นซึ่งสามารถคำนวณได้ดังนี้

$$\mu_{INFERANCE} = \max(0.4, 0.6, 0.25, 0.25)$$

$$= 0.6$$

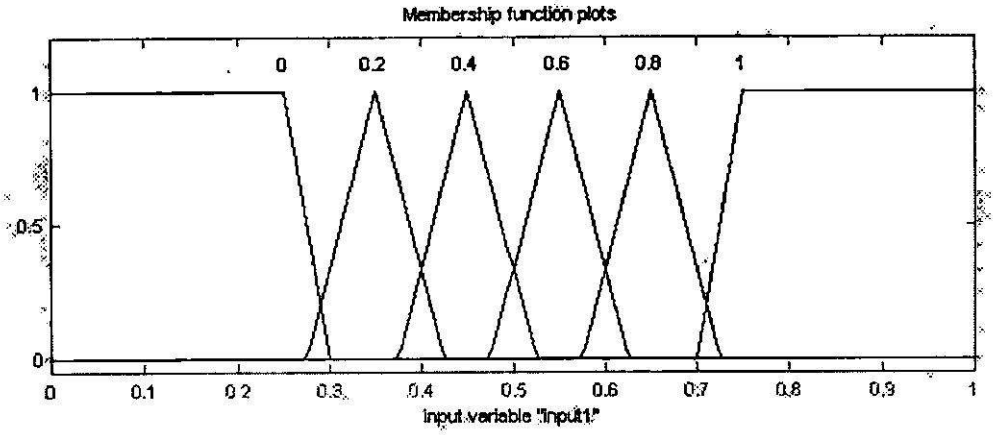
กฎที่ใช้ในการประมาณค่า I[t] ในกรณีการ Defuzzification โดยวิธีนี้คือ

IF T[t] is cool AND P [t] is ok THEN I [t] is approximately zero

เนื่องจากกฎข้อนี้ให้ค่าความจริงที่มีค่าสูงสุด $\mu_{INFERANCE} = 0.6$ ดังนั้นสามารถคำนวณค่าประมาณของ I[t]

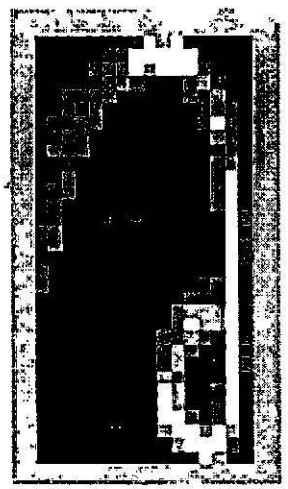
3.5 การนำฟัซซี่เซต มาใช้ในการตัดสินใจ

เมื่อได้ผลของการทรานฟอร์มของตัวอักษรคัดลายมือภาษาไทยจำนวน 30 ตัวให้เหลือตัวแทนของตัวอักษรตัวแทนเพียง 1 ตัว ซึ่งได้ค่าของน้ำหนักของแทนตัวอักษรออกมาหลากหลายมาก ดังนั้นจึงปรับค่าน้ำหนักของตัวแทนอักษรให้อยู่ในรูปฟัซซี่เซต เพื่อจะได้มีความถูกต้องมากเมื่อมีอยู่ขั้นตอนในการเรียนรู้ตัวอักษร โดยกำหนดค่าฟัซซี่เซตดังภาพที่ 3.11



ภาพที่ 3.11 แสดงการกำหนดค่าของฟัซซีเซต

ซึ่งผลของการนำตัวแทนของตัวอักษรที่ได้จากคาร์ฮูเน็นเลิฟมามาเข้าระบบฟัซซี่เซต เพื่อจะได้ผลของการเรียนรู้ตัวอักษรในขั้นตอนของการเทรนนิ่งในโครงข่ายนิวรอลเน็ตเวร์ค ซึ่งผลของการเข้าระบบฟัซซีเซตได้ดังภาพที่ 3.12



ภาพที่ 3.12 แสดงตัวแทนของตัวอักษร ว ที่ผ่านระบบฟัซซีเซต

บทที่ 4
ทฤษฎีนิวรอลเน็ตเวอร์ค

โครงข่ายประสาทเทียมได้เริ่มมีการคิดค้นมานานกันมาตั้งแต่ปี ค.ศ.1940 โดยในช่วงเวลาที่ก่อให้เกิดสถาปัตยกรรมของโครงข่ายประสาทเทียมชนิดต่างๆ ในปี ค.ศ. 1943 วอร์เรน แมคคูล ล๊อค (Warren McCulloch) และวอลเตอร์ พิทท์(Walter Ptiis) ได้ออกแบบนิวรอนที่ถือเป็นต้นกำเนิดและพื้นฐานของโครงข่ายประสาทเทียมในปัจจุบัน โครงข่ายของแมคคูลล็อคและพิทท์ ประกอบด้วยชั้นของโครงข่ายจำนวน 2 ชั้น คือ ชั้นอินพุทและชั้นเอาท์พุท โครงข่ายจะมีการ เชื่อมโยงจากชั้นอินพุทไปยังชั้นเอาต์พุทโดยผ่านน้ำหนัก (weight) ซึ่งค่าน้ำหนักจะถูกกำหนดค่า ไว้ตายตัว การทำงานเบื้องต้นของโครงข่ายคือนำไปใช้จำลองฟังก์ชั่นทางลอจิก พื้นฐานคือ ฟังก์ชั่น AND ฟังก์ชั่น OR ฟังก์ชั่น NOT และฟังก์ชั่น XOR โดยนำเสนอการกำหนดค่าน้ำหนักเพื่อให้โครงข่ายทำงานได้กับการจำลองตามฟังก์ชั่นดังกล่าว

ในปี ค.ศ. 1949 โดแนลด์ เฮบบ์ (Donald Hebb) ได้ออกแบบกฎการเรียนรู้สำหรับ โครงข่ายประสาทเทียมเป็นคนแรก กฎการเรียนรู้นี้ก็คือขั้นตอนของกระบวนการปรับแต่งค่า น้ำหนักเพื่อทำให้เกิดค่าเอาท์พุท

ในปี ค.ศ. 1957 แฟรงค์ โรเซนบลัทท์ (Frank Rosenblatt) ได้นำเสนอรูปแบบที่สำคัญ รูปแบบหนึ่งของโครงข่ายประสาทเทียมนั่นก็คือรูปแบบที่เรียกว่าเพอร์เซพตรอนเป็นสถาปัตยกรรมอันประกอบไปด้วยชั้นอินพุทต่อผ่านน้ำหนักไปยังชั้นนิวรอลที่ชั้นเอาท์พุท และใช้กฎการเรียนรู้ของเพอร์เซพตรอนในการปรับแต่งค่าน้ำหนักที่มีประสิทธิภาพดีกว่ากฎการเรียนรู้ของแฮบบ์

ในปี ค.ศ.1960 เบอร์นาร์ด วิดโรว์ (Bernard Widrow) และมาเซียน ฮอฟฟ์ (Marcian Hoff) ได้พัฒนากฎการเรียนรู้ซึ่งใกล้เคียงกับการเรียนรู้ของเพอร์เซตตรอนและนำกฎนี้ไปใช้กับระบบหรือรูปแบบที่ชื่อว่าอะดาไลน์ รูปแบบมาดาไลน์คือรูปแบบที่พัฒนาจากอะดาไลน์โดยวิดโรว์

ในปี ค.ศ. 1972 ทูโว คาโคเนน (Teuvo Kohonen) ได้เริ่มพัฒนาการเรียนรู้แบบจัดตัวเอง (self-organized) สถาปัตยกรรมของโครงข่ายคาโคเนนนี้ประกอบด้วยจำนวนชั้น 2 ชั้น คือชั้น อินพุทและชั้นเอาท์พุท การเชื่อมโยงที่ชั้นอินพุทไปยังนิวรอลที่ชั้นเอาท์พุทนั้นจะต่อผ่านน้ำหนัก ข้อแตกต่างจากโครงข่ายอื่นๆ คือกระบวนการเรียนรู้ที่สามารถจัดกลุ่มข้อมูลอินพุทได้ด้วยตัวของมันเองและเรียกการเรียนรู้นี้ว่าการเรียนรู้แบบแข่งขัน (competitive learning)

หลังจากปี ค.ศ. 1980 เป็นต้นมาได้มีพัฒนารูปแบบของโครงข่ายประสาทเทียมชนิดต่างๆ เช่น รูปแบบจัดตัวเองที่เรียกว่า ART คิดค้นโดย สตีเฟน กรอสเบิร์ก (Stephen Grossberg) และเกียล คาร์เพนเตอร์ (Gail Carpenter) รูปแบบโครงข่ายฮอปฟิลด์ คิดค้นโดย จอน ฮอปฟิลด์ (john Hopfield) นีโอคอกนิตรอน คิดค้นโดยคูนิฮิโกะ ฟูกูชิมา (Kunihiko Fukushima) รูปแบบโบลสแมนน์แมชชีน (Boltzmann machine) คิดค้นโดยนักวิจัยหลายคน และรูปแบบที่นิยมใช้ กัน

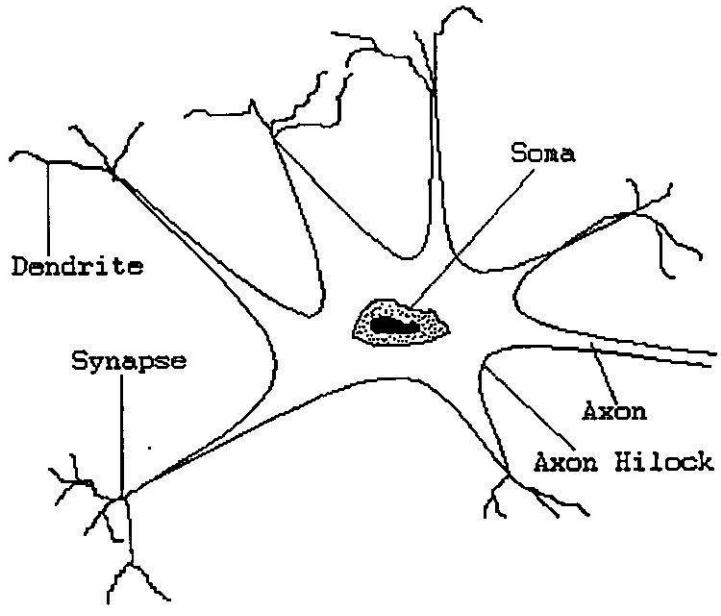
มากคือรูปแบบแพร่ย้อนกลับ (Back-Propagation) คิดค้นโดยเดวิด ปาร์คเคอร์ (David Parker) และเดวิด รูเมลฮาร์ท (David Rumelhard)

4.1 ลักษณะโครงข่ายนิวรอน

นิวรอลเน็ตเวอร์ค หมายถึงโครงข่ายใยประสาทที่เชื่อมต่อกันระหว่างเซลล์ประสาทจำนวนมากมาย มีความสามารถประมวลผลสูง ทำหน้าที่เป็นจุดศูนย์กลางการควบคุมกิจกรรมต่าง ๆ การวิจัยสร้างโครงข่ายประสาทเทียม (Artificial Neural Networks) มีแนวคิดเลียนแบบการทำงานของสมองชีวภาพ โดยเรียนรู้และศึกษาการทำงานของสมองชีวภาพเพื่อกำหนดแนวทางสำหรับการสร้างแบบจำลองขึ้นมา แล้วพยายามสมมติฐานลักษณะการทำงานโดยจำลองเป็นโมเดลคณิตศาสตร์ที่มีลักษณะเดียวกันแล้วคำเนินการคำนวณโดยใช้คอมพิวเตอร์

เซลล์ประสาทของสิ่งมีชีวิตมีหลายประเภทแล้วแต่หน้าที่ของมันเซลล์ประสาทในตัวของคนเราก็เช่นกันมีอยู่หลายประเภทตามตำแหน่งและหน้าที่ของมันเช่นเซลล์ประสาทของกล้ามเนื้อ เซลล์ประสาทในสมอง เซลล์ประสาทที่ลิ้นและจมูกเป็นต้น โครงสร้างและส่วนประกอบของเซลล์ประสาทโดยทั่วๆ ไป

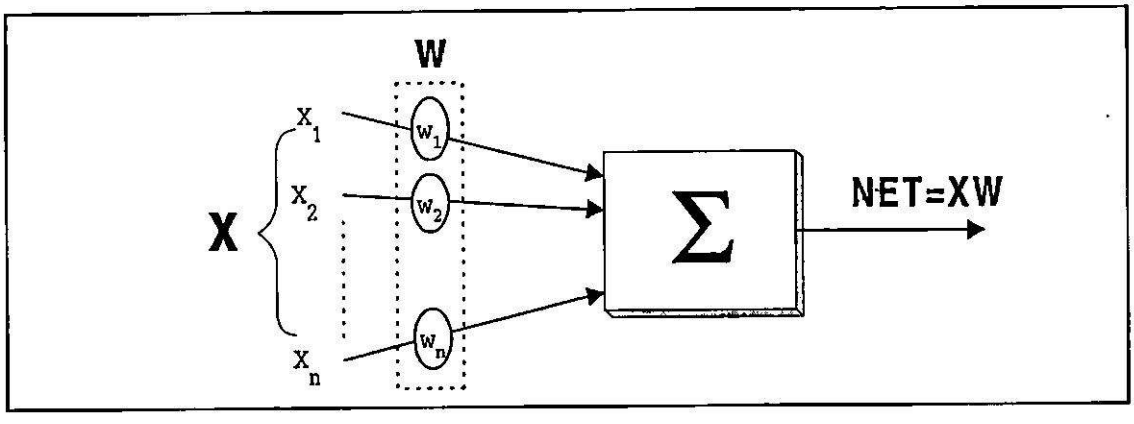
ระบบคิดคำนึงของมนุษย์ มีโครงสร้างพื้นฐานจากเซลล์สมองที่เรียกว่านิวรอล (Neurons) เรียงเป็นชั้นๆ อย่างซับซ้อนจำนวนมหาศาล ประมาณหมื่นล้าน(10^{11})นิวรอลและอาจมีจุดเชื่อมโยงส่งผ่านจุดเชื่อมโยงภายในถึงพันล้านล้าน(10^{15})จุด (PhilipD.Wasserman,1989:12)แต่ละนิวรอลจะมีคุณลักษณะแตกต่างกันไปโดยมีการทำงานคล้ายกันคือ รับเข้า, ประมวลผล, ส่งออกสัญญาณไฟฟ้าเคมีผ่านไปยังนิวรอล ซึ่งจะส่งสื่อสารไปตามระบบของสมอง



ภาพที่ 4.1 โครงสร้างและส่วนประกอบของเซลล์ประสาท

จากภาพที่ 4.1 ส่วนแขนงที่ขยายแยกออกไปจากตัวเซลล์ต่อไปยังเซลล์อื่น ๆ เพื่อรับสัญญาณเรียกว่า เดนไดรท์ (Dendrites) จุดรับสัญญาณจากเซลล์อื่นเข้ามายังตัวเซลล์จะผ่านมาทางจุดเชื่อมต่อที่เรียกว่า ซิน แนปส์ (Synapse) ซึ่งแอกซอน (Axon) จะเป็นตัวส่งสัญญาณเอาท์พุทออกไปยังนิวรอลอื่นจากผลการวิจัยพบว่า แต่ละนิวรอลจะเชื่อมต่อออกไปยังนิวรอลอื่นๆ ซึ่งแต่ละนิวรอลจะมีคุณสมบัติในการเพิ่มขยายหรือลดทอนความเข้มของสัญญาณบางสัญญาณที่เข้ามาทางเดนไดรท์ ของเซลล์ (ซึ่งมีแขนงมากมาย) อาจสามารถกระตุ้นตัวเซลล์ แต่บางสัญญาณก็อาจจะยับยั้งตัวเซลล์ เนื่องจากเซลล์ประสาทหนึ่งเซลล์มีเดนไดรท์มาก ฉนั้น สัญญาณ กระตุ้นจากเดนไดรท์ ที่รับเข้ามาจากเซลล์ประสาทอื่นๆ จะถูกนำมารวมกันที่ตัวเซลล์ประสาทที่เซลล์ประสาทจะมีค่าเทรสโฮล (Threshold) ค่าหนึ่งหากผลรวมของสัญญาณไฟฟ้าเคมี (Electrochemical) มีค่ามากกว่า เทรสโฮลเซลล์ เซลล์ประสาทก็จะส่งสัญญาณขนาดหนึ่งผ่านทางแอกซอนไปยังนิวรอลอื่นๆ การจัดเรียงชั้น (Layer) และลักษณะการเชื่อมโยงระหว่างนิวรอลในสมองนั้นมีการจัดเรียงที่ซับซ้อน สอดคล้องกับหน้าที่การทำงานเฉพาะส่วน มีการเจริญเติบโตสัมพันธ์กับสิ่งแวดล้อมและมีการเรียนรู้ตลอดเวลาซึ่งใช้เวลานานนับปีดังนั้นจึงยากที่จะสร้างโมดุลขึ้นมาเพื่อเลียนแบบให้มีคุณลักษณะคล้ายสมองชีวภาพได้ทั้งหมด ผลงานที่ได้จากการทำวิจัยในปัจจุบันเป็นเพียงการจำลอง การเลียนแบบและการทำงานเฉพาะบางส่วนของโครงข่ายประสาท มาใช้เฉพาะกับงานใดงานหนึ่ง ซึ่งมีการวิจัยลักษณะของโครงข่ายแบบต่างๆ ขึ้นมา โดยแต่ละแบบจะเหมาะกับงานประเภทหนึ่ง ๆ เท่านั้น

การออกแบบโครงสร้างประสาทเทียมนั้นมีสมมติฐานขั้นแรกจากคุณสมบัติของระบบประสาทชีวภาพ ดังที่กล่าวมา กล่าวคือ ชุดรับสัญญาณข้อมูล อินพุทของเซลล์ประสาทหนึ่งได้จากสัญญาณเอาท์พุทของเซลล์ประสาทอื่นๆ ผ่านทางซินแนปส์และเดนไดรท์ ข้อมูลแต่ละค่าที่รับมาจะถูกลดขนาดด้วย ซินแนปติกส์ ซึ่ง ภายในประกอบด้วยสารเคมีประเภท K^{+},Ca^{++},Na^{+},Cl^{-} ซึ่งจะมีลักษณะทางความนำ พัลส์ (Pulse) สัญญาณไฟฟ้าเคมีที่แตกต่างกัน (James A. Freeman and David M.Skapura,1991:8-9) ด้วยเหตุนี้ โมดุล ประสาทเทียมที่สร้างขึ้น จะต้องมีการถ่วงน้ำหนักให้กับโมดุลก่อนที่จะนำเข้าสู่โมดุลประสาทเทียม จุดนี้เรียกว่า ซินแนปติกส์เวจท์ ปริมาณของข้อมูลที่เข้าสู่นิวรอล จะถูกนำมารวมกัน และตัดสินใจด้วยระดับความสนใจของ นิวรอล (Activation level) แล้วจะส่งเป็นเอาท์พุทออกที่แอกซอนไปยังนิวรอลอื่นๆ



ภาพที่ 4.2 แสดงไดอะแกรมของนิวรอลที่สร้างขึ้น (Artificial Neuron)

จากภาพที่ 4.2 แสดงถึงโมดุลที่สร้างขึ้นโดยแนวความคิดจากเซลล์สมองชีวภาพ สัญญาณอินพุท คือ X_1, X_2, ..., X_n จะถูกป้อนเข้าไปยังนิวรอลที่สร้างขึ้น ซึ่งเปรียบเทียบได้กับสัญญาณที่ป้อนเข้ายังซินแนปส์ของ นิวรอลชีวภาพ สัญญาณอินพุทนี้จะนำไปคูณกับค่าซินแนปติกส์เวจท์ที่มีค่าตั้งแต่ 0.00-1 (Weight: ค่าที่ใช้ถ่วงน้ำหนัก) W_1, W_2, ..., W_n ก่อนที่จะเข้าสู่บล็อกซัมเมชั่น (Σ:Summation) ซึ่งค่าถ่วงน้ำหนักนี้จะสอดคล้องกับค่าสเตรงท์(Strength) ของจุดต่อซินแนปส์ชีวภาพแต่ละจุด(Single biological synaptic connection) บล็อกซัมเมชั่นนี้ก็จะทำหน้าที่สอดคล้องคล้ายกับตัวเซลล์สมองชีวภาพ ผลรวมทางคณิตศาสตร์ของอินพุตและเวจท์จะได้เป็นเอาท์พุท เราเรียกว่า เน็ต(NET) ซึ่งเราจะรวมกันในรูปของเวกเตอร์ได้ดังนี้

$$NET = X_1W_1 + X_2W_2 + \ldots\ldots\ldots + X_nW_n \quad \ldots\ldots(4.1)$$

จะได้

$$NET = XW \quad \ldots\ldots(4.2)$$

ฟังก์ชั่นกระตุ้นความสนใจ (Activation Function)

เมื่อได้สัญญาณ NET แล้วกระบวนการต่อมาที่นิวรอลต้องทำคือตัดสินใจเราจึงต้องกำหนดฟังก์ชั่นการ ตัดสินใจ เพื่อใช้เป็นระดับของการตัดสินใจให้กับนิวรอล เพื่อให้ได้สัญญาณเอาท์พุทของ นิวรอลออกมา ซึ่งเชื่อมต่อไปยังนิวรอลตัวอื่น ๆ เป็นโครงข่าย OUT ที่ได้อาจเป็น Simple linear function โดย

$$OUT = K[NET] \quad \text{.....(4.3)}$$

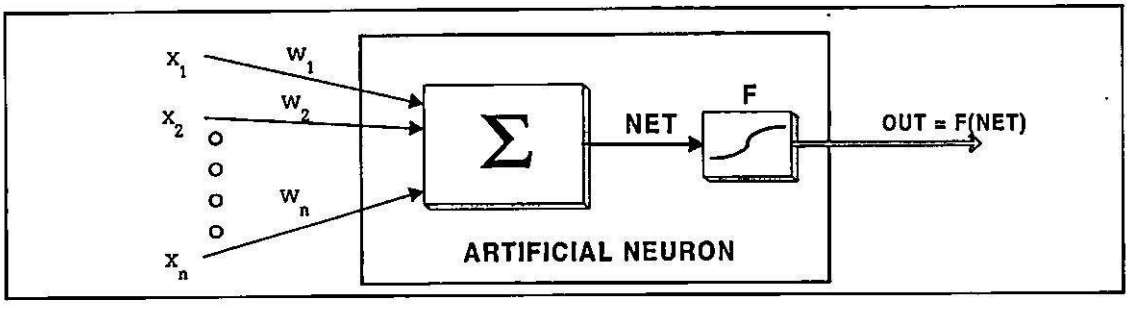
โดย K เป็นค่าคงที่ ที่เรียกว่า Threshold function

ตัวอย่างเช่น

$$OUT = 1 \text{ ถ้า } NET > T$$

$$OUT = 0 \text{ เมื่อเป็นกรณีอื่น} \quad \text{.....(4.4)}$$

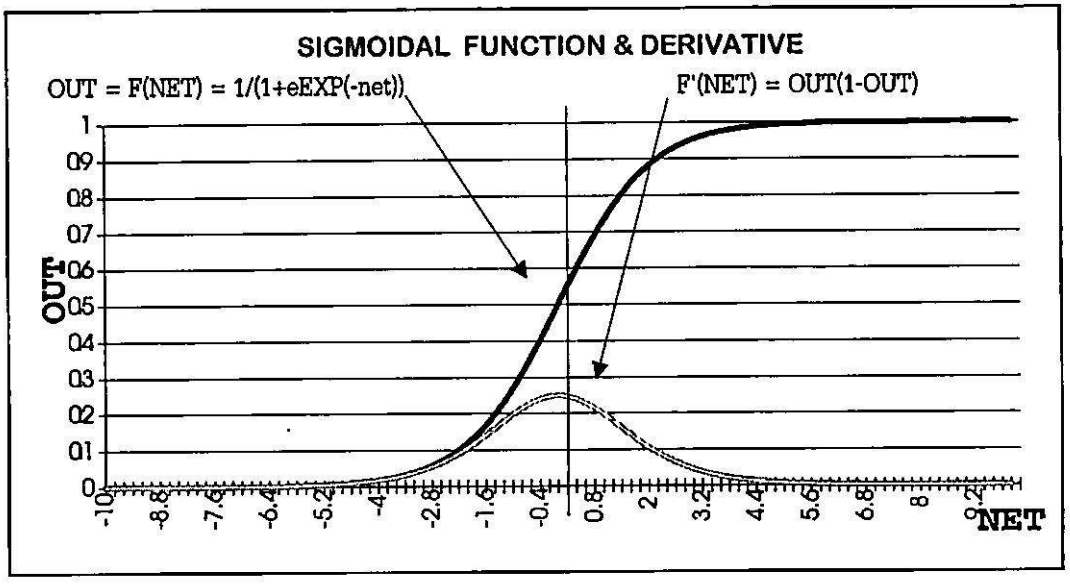
และ T เป็นค่าเทรสโฮลดคงที่ หรืออาจเป็น Function อื่น ๆ ที่เลียนแบบคุณสมบัติที่ไม่เป็นเชิงเส้นของเซลล์ประสาทชีวภาพได้อย่างใกล้เคียงกว่า และใช้เป็นฟังก์ชั่นให้กับโครงข่ายทั่วไปได้



ภาพที่ 4.3 แสดงโมดุลนิวรอลที่สร้างขึ้นร่วมกับ Activation Function

ในภาพที่ 4.3 บล็อก F จะรับผลที่ได้จาก NET มาสร้างเป็นสัญญาณเอาท์พุทที่ OUT โดยกระบวนการภายในบล็อก F จะบีบช่วงของ OUT ให้อยู่ในขอบเขตจำกัด ตามต้องการ ดังนั้น ค่า OUT จะมีค่าไม่ต่ำกว่าช่วงที่กำหนดโดยค่าของ NET เราเรียกบล็อก F นี้ว่าสแควชชิ่งฟังก์ชั่น (Squashing function) และโดยทั่วไป สแควชชิ่งฟังก์ชั่นที่ใช้เป็นแบบลอจิสติกฟังก์ชั่น หรือซิกมอยด์(Logistic function or "Sigmoid") ซึ่งมีรูปร่างคล้ายตัว S โดยเขียนเป็นสมการคณิตศาสตร์ได้ดังนี้คือ

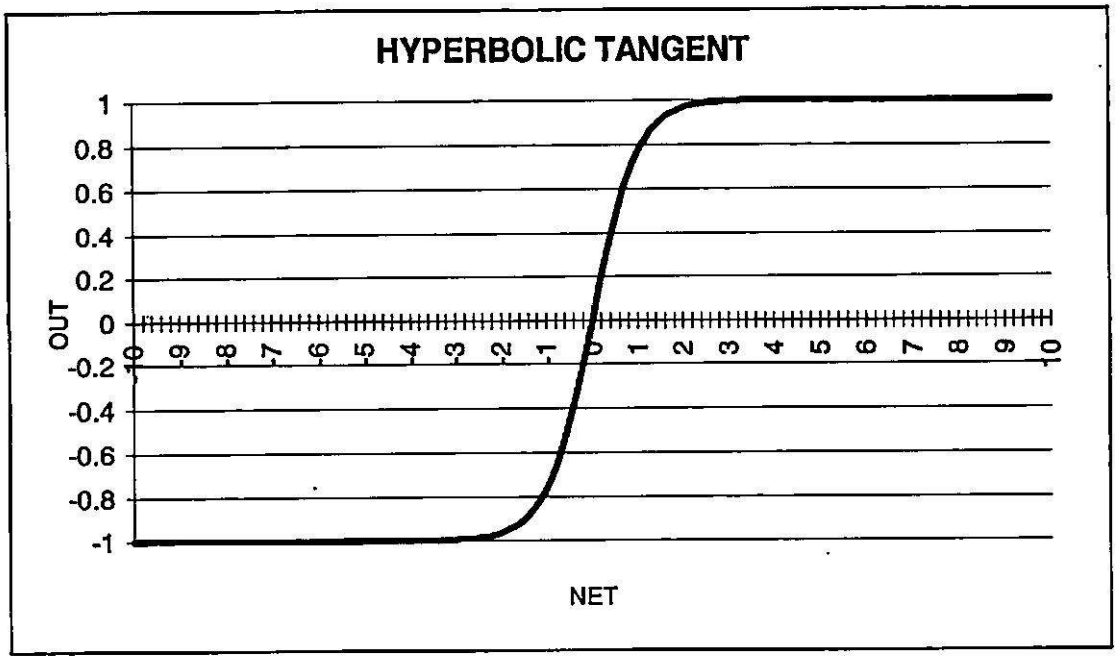
$$F(x) = \frac{1}{(1-e^{-x})} \quad \text{.....(4.5)}$$



ภาพที่ 4.4 แสดงกราฟที่ได้จากสมการซิกมอยด์ลอจิสติกฟังก์ชั่น(Sigmoidal logistic function)

ลักษณะของเทรสโฮลฟังก์ชั่นมีลักษณะเป็น Non-linear function เช่น S-Curve เราจะได้ค่าเอาท์พุท ที่มีความไวต่อสัญญาณอินพุทที่มีขนาดเล็กๆ และเฉลี่ยต่อสัญญาณแรงๆ ซึ่งสัญญาณอ่อนๆ ไปทางบวกเพียงเล็กน้อยก็จะให้ OUT ใกล้เคียง "1" กระตุ้นหรือสัญญาณอ่อนๆ ทางลบเพียงเล็กน้อยก็จะทำให้ Output ใกล้เคียง "0" (ยับยั้ง) ขณะที่สัญญาณแรงๆ ทางบวกก็ยังคงให้ Output ใกล้เคียง "1" และสัญญาณทางลบแรงๆ ก็คงให้ Output ใกล้เคียง "0" เช่นกัน คุณลักษณะแบบนี้เป็นแบบ NON-LINEAR GAIN ซึ่งคลอสส์เบอร์ก (Grossberg,1973) พบว่า คุณลักษณะที่เป็น Non-linear gain นี้สามารถแก้ปัญหา Noise-saturation dilemma ได้ และทำให้นิวรอลเทียมที่สร้างขึ้นสามารถทำงานกับขนาดของอินพุทได้กว้างมากขึ้น

ยังมีฟังก์ชั่นอื่นๆ อีกคือ ไฮเปอร์โบลิก แทนเจนท์ (Hyperbolic tangent) มันจะมีลักษณะคล้ายกับ Logistic function และนิยมใช้บ่อยๆ ในการสร้างโมดุลคณิตศาสตร์ การกระตุ้นเร้าความสนใจของเซลล์สมองเทียม ซึ่งมีคุณสมบัติคล้ายชีวภาพของเซลล์สมอง คือ OUT = Tanh(X)

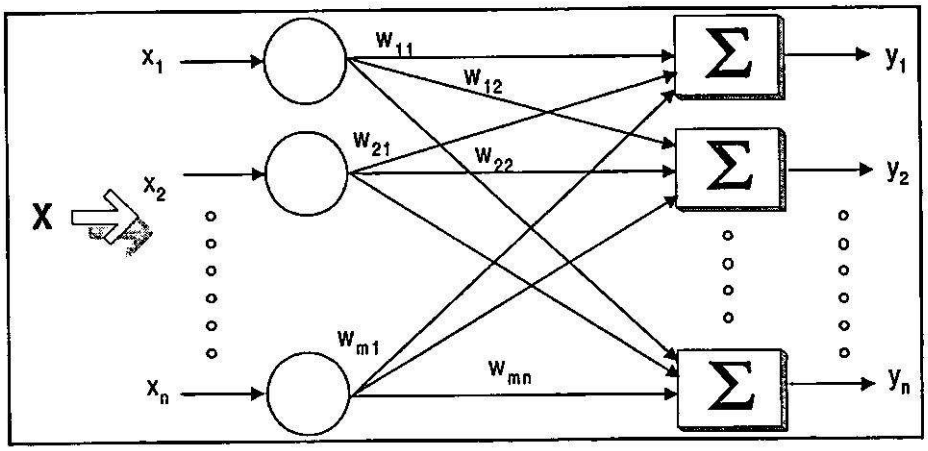


ภาพที่ 4.5 แสดง Hyperbolic Tangent Function

จากภาพที่ 4.5 ส่วนที่เหมือนกับ ซิกมอยด์ ลอจิสติก ฟังก์ชั่น คือมีลักษณะเป็น S แต่เนื่องจากมันจะมีความสมมาตรจึงให้ OUTPUT อยู่ระหว่าง "-1" ถึง "1" OUTPUT จะเป็น "0" เมื่อ NET เป็น "0" OUTPUT เข้าใกล้ "1" เมื่ออินพุทไปทางบวกและเข้าใกล้ "-1" เมื่อ อินพุทมีทิศทางไปทางลบ

4.2 โครงข่ายของนิวรอลแบบต่าง ๆ

ในการสร้างเซลล์ประสาทเทียมเพียงหนึ่งเซลล์ โดยใช้แนวความคิดจากเซลล์ประสาทชีวภาพ การจะนำเซลล์ประสาทเทียมมาใช้งานได้นั้น ต้องใช้เซลล์ประสาทเทียมที่มีคุณลักษณะต่างๆ กัน (ค่า Weight จะทำให้คุณสมบัติของเซลล์ประสาทเทียมแต่ละเซลล์มีคุณลักษณะแตกต่างกันไป) มาเชื่อมโยงเป็นโครงข่ายในลักษณะเดียวกับเซลล์สมองชีวภาพเสียก่อน ซึ่งลักษณะการเชื่อมโยงมีหลายชนิด แต่ละชนิดก็มีคุณลักษณะเด่นที่แตกต่างกันไป



ภาพที่ 4.6 แสดงลักษณะโครงข่ายประสาทเทียมแบบชั้นเดียว (Sigle-Layer Neural Networks)

จากภาพที่ 4.6 เป็นโครงข่ายประสาทเทียมแบบชั้นเดียว ที่ประกอบด้วยเซลล์ประสาทเทียมง่ายๆ หลายๆชุด ความสามารถในการคำนวณของโครงข่ายประสาทเทียมได้มาจากลักษณะการเชื่อมต่อเป็นโครงข่ายประสาทเทียมโครงข่ายง่ายๆ เป็นกลุ่มโมดุลประสาทเทียมที่เชื่อมต่อกัน เป็นชั้น ๆ (Layer) ในภาพที่ 4.6 เป็นโครงข่ายประสาทเทียม แบบชั้นเดียว (Single layer) ที่ประกอบด้วยเอาท์พุทเลเยอร์ (กลุ่มของบล็อกซัมเมชั่นที่อยู่ทางขวามือ) และอินพุทเลเยอร์(วงกลมทางซ้ายมือ)โดยไม่พิจารณาอินพุทเลเยอร์ว่าเป็นนิวรอลเลเยอร์ เนื่องจากอินพุทเลเยอร์จะทำหน้าที่เชื่อมต่ออินพุทที่รับมาและส่งออกไปให้ยังแต่ละอินพุทนิวรอลเลเยอร์ (ในที่นี้คือ Output layer) ในชั้นถัดไป โดยแต่ละอินพุทจะถูกคูณโดยค่าเวจท์เฉพาะแต่ละอินพุท โครงข่ายประสาทเทียมที่สร้างขึ้นในขั้นแรกไม่ซับซ้อน โดยแต่ละนิวรอล จะได้เอาท์พุทจาก

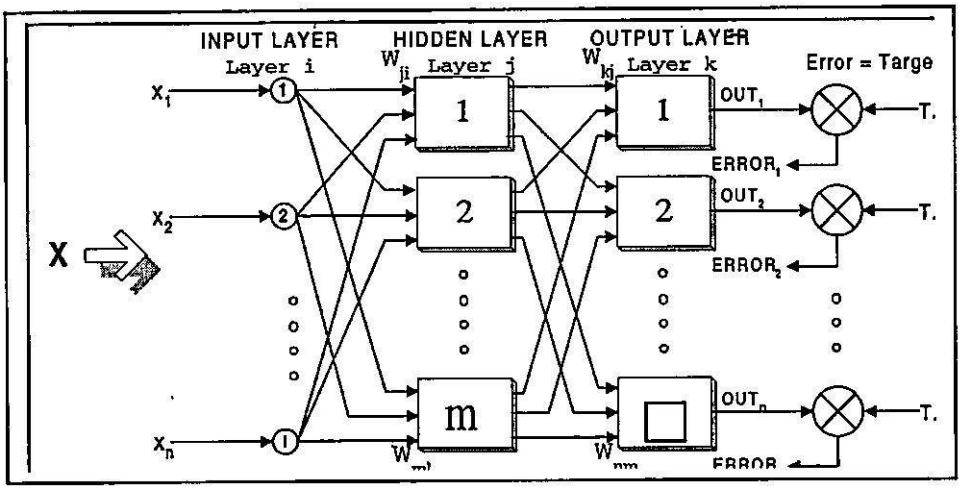
OUT = Logistic Function คูณ (ผลรวมของ Input X กับ Weight)

หรือ OUT = F(NET) .....(4.6)

อย่างไรก็ดีลักษณะการเชื่อมโยงระหว่างโครงข่ายไม่ได้มีแบบเดียว การเชื่อมโยงระหว่างเลเยอร์อาจมีการเชื่อมโยงย้อนกลับมาที่อินพุทเลเยอร์อีก ซึ่งโครงข่ายประสาทชีวภาพก็มีลักษณะดังกล่าวเช่นกัน สำหรับค่า Weight ในภาพที่ 4..6มีวิธีการพิจารณาในรูปของ เวจท์เมตริก (Weight matrix) ซึ่งหากโครงข่ายมีหลายชั้น จะช่วยให้ระบุค่าเวจท์ได้ง่ายขึ้น และเพื่อหลีกเลี่ยงความสับสนจะกำหนดเป็น ไดเมนชั่น (Dimensions) ของเมตริก โดยให้ m แทนจำนวนแถว หรือจำนวนของ อินพุทและ n แทนจำนวนของนิวรอล ที่สร้างขึ้น ตัวอย่างเช่น เวจท์ ที่เชื่อมระหว่างอินพุท ตัวที่ 4 กับนิวรอลตัวที่ 2 คือ $W_{4,2}$

โครงข่ายประสาทเทียมแบบหลายชั้น (Multilayer Artificial Neural Networks)

โครงข่ายที่ซับซ้อนจะมีความสามารถในการคำนวณที่ดีขึ้นมันจะเป็นโครงข่ายที่มีโครงสร้างเป็นจินตนาการที่น่าเป็นไปได้โดยการจัดการเชื่อมโยงนิวรอลมีโครงสร้างเป็นชั้นๆ คล้ายส่วนหนึ่งของสมอง และมีการพัฒนาอัลกอริทึมเกี่ยวกับการฝึกสอนให้โครงข่ายแบบหลายชั้นทำงานได้ตามความต้องการแล้วเมื่อไม่นานมานี้ โครงข่ายแบบหลายชั้น อาจจะสร้างจาก กลุ่มของโครงข่ายแบบชั้นเดียวเอาท์พุทของ Layer หนึ่ง จะใช้เป็นอินพุทของ Layer ถัดไป ในภาพที่ 43 แสดงเน็ตเวอร์คที่มีการเชื่อมต่อแบบสองชั้น



ภาพที่ 4.7 แสดงไดอะแกรมของ Backpropagation Neural Networks แบบสองชั้น

ภาพที่ 4.7 แสดงโครงข่ายประสาทเทียมแบบหลายชั้น ที่ต่อเชื่อมโยงแบบเต็มขั้น ในโครงข่ายแบบหลายชั้นมีการเรียกชื่อชั้นต่างๆ ดังนี้ คือ ชั้นที่ต่อโดยตรงกับอินพุท เรียกว่า อินพุทเลเยอร์ (Input layer)ชั้น นี้จะไม่มีการคำนวณ แต่จะทำหน้าที่ต่อเชื่อมข้อมูลไปยังชั้นถัดไป ชั้นที่อยู่ท้ายสุดทางขวามือเรียกว่า เอาท์พุทเลเยอร์ (Output layer) เป็นชั้นที่โครงข่ายจะให้ผลลัพธ์ ส่วนชั้นที่อยู่ระหว่างอินพุทเลเยอร์และเอาท์พุท เลเยอร์ จะมีกี่ชั้นก็ตามจะเรียกว่า ฮิดเดนเลเยอร์ (Hidden layer) หากฮิดเดนเลเยอร์ มีหลาย ๆ ชั้นก็จะมีการตั้งชื่อเฉพาะลงไปให้กับแต่ละชั้น

ฟังก์ชันกระตุ้นความสนใจแบบไม่เป็นเชิงเส้น (The Nonlinear Activation Function)

การนำเอาท์พุทของเลเยอร์หนึ่ง มาเชื่อมกับอินพุทของเลเยอร์ชั้นถัดไป โดยผ่านฟังก์ชันกระตุ้นความสนใจแบบไม่เป็นเชิงเส้น จะทำให้โครงข่ายมีความสามารถในการคำนวณเพิ่มขึ้น (หากไม่ผ่านฟังก์ชันดังกล่าวความสามารถการคำนวณจะไม่เพิ่มขึ้นและจะมีความสามารถไม่แตกต่างไปจาก Single layer networks)และสามารถกำหนดขอบเขตของเอาท์พุทให้อยู่ในช่วงที่ต้องการได้

การฝึกสอนให้กับโครงข่ายประสาทเทียม (Training of Artificial Neural Networks)

ค่าเวจท์ มีความสัมพันธ์กับอะไร ? เปลี่ยนแปลงอย่างไร ? นั้นก็เช่นเดียวกับเด็กที่คลอดออกมาก็มีสมองแล้วแต่สมองยังไม่เจริญเติบโตเพียงพอและยังไม่ได้รับการฝึกสอนและเรียนรู้ เด็กจึงไม่สามารถ ทำกิจกรรมใดๆ ด้วยตนเอง เว้นแต่กิจกรรมที่ธรรมชาติสร้างมาพร้อมกับการกำเนิดที่เรียกว่า "สัญชาติญาณ" ซึ่งธรรมชาติใส่คุณลักษณะบางอย่างให้เซลล์สมองบางส่วนตั้งแต่ทารกเจริญเติบโตอยู่ในครรภ์มารดา เช่น ระบบควบคุมการหายใจ, ความรู้สึก, การเรียกร้องเมื่อหิว, การตอบสนองต่อสิ่งเร้า ฯลฯ เด็กจะพัฒนาการเรียนรู้ไปตามขั้นตอน หลังจากนั้นสมองของเขาจะได้รับการฝึกสอน และเจริญเติบโตไปพร้อมๆ กัน เซลล์สมองจะได้รับการปรับคุณลักษณะสอดคล้องกับการฝึกสอน และจะเจริญเป็นโครงข่ายสอดคล้องกัน

โครงข่ายประสาทเทียมที่สร้างขึ้นมีลักษณะเช่นเดียวกัน คือ เมื่อสร้างเสร็จ แต่ละเซลล์ประสาทที่สร้างขึ้นมานั้น จะยังไม่มีคุณลักษณะใดเลย เนื่องจากยังไม่มีการกำหนดค่าซินแนปติกส์เวจท์ที่เหมาะสมกับงานที่ต้องการให้กับมัน จึงต้องมีการฝึกสอนเพื่อให้เน็ตเวอร์คที่สร้างขึ้นมี คุณลักษณะตามที่ต้องการ การ ฝึกสอนของโครงข่ายประสาทเทียม จะกระทำโดยการปรับเปลี่ยนค่าซินแนปติกส์เวจท์เพื่อให้โครงข่ายจดจำ แพตเทอร์นความสัมพันธ์ระหว่างอินพุทกับเอ้าท์พุทได้ โดยในขั้นแรกอาจกำหนดเป็นค่าสุ่มใดๆ (Random weight) ก่อนแล้วถึงปรับเปลี่ยนเวจท์ไปตาม อัลกอริทึมสมมติฐาน หลายๆ รอบจนกว่าจะได้เอาท์พุทของ เน็ตเวอร์ค เหมือนกับเอาท์พุทที่ต้องการในเงื่อนไขความผิดพลาดที่ยอมรับได้

วัตถุประสงค์ของการเทรนนิ่ง (Objective of Training)

เนื่องจากค่าเวจท์ที่ให้เป็นค่าสุ่มใดๆ โครงข่ายจึงไม่แสดงคุณลักษณะใดออกมา การฝึกสอน (Training) ให้โครงข่ายก็คือการปรับค่าเวจท์ทุก ๆ จุดให้สอดคล้องกับอินพุทหลายๆแบบ เพื่อให้ได้เอาท์พุทตามต้องการนั่นเอง การฝึกสอนโครงข่าย จะต้องบรรลุถึงกระบวนการเข้าใจพื้นฐานเสียก่อน คือ การเรียนรู้ในโครงข่ายประสาทเทียมนั้นมีขีดจำกัด ปัญหาต่างๆ ผู้ใช้คงต้องแก้ให้มันก่อน แล้วนำผลนั้นไปอ้างอิงสำหรับการปรับปรุงค่าเวจท์ หลังจากปรับเวจท์จนได้ค่าผิดพลาดที่เอาท์พุทเทียบกับเป้าหมายน้อยลงเป็นที่พอใจแล้ว โครงข่ายประสาทเทียมนั้นก็พร้อมที่จะวิเคราะห์อินพุทและให้เอาท์พุทตามลักษณะตัวอย่างที่มันเคยเรียนรู้มา การเรียนรู้จะมีการปรับเวจท์หลายๆ รอบ จนค่าเวจท์สอดคล้องกับตัวอย่างหลายๆ ตัวอย่าง และให้เอาท์พุทตามต้องการ พบว่าโครงข่ายได้ตัวอย่างสำหรับการเทรนนิ่งมากๆ โครงข่ายก็จะมีความแม่นยำสูงขึ้น แต่ก็จะใช้เวลาในการเทรนนิ่งเพิ่มขึ้นเช่นกัน หากพิจารณาต่อไปจะพบว่า โครงข่ายประสาทเทียมที่สร้างขึ้นจะมี พฤติกรรมคล้ายกับระบบการเรียนรู้ของมนุษย์มาก เป็นเพราะมีต้นแบบมาจากระบบประสาทชีวภาพนั่นเอง

การเทรนนิ่งแบบควบคุม (Supervised Training)

เทรนนิ่งอัลลกอลิทึมถูกจัดเป็น 2 ประเภท คือ แบบควบคุม (Supervised training) และแบบอิสระ (Unsupervised training) โดย การเทรนนิ่งแบบควบคุม จะต้องการคู่ของการเทรนนิ่งระหว่างอินพุทกับเป้าหมายที่ต้องการ ที่เรียกว่า เทรนนิ่งแพร์ (Training pairs) โครงข่ายจะถูกเทรนไปตามจำนวนของคู่ที่ เทรนนิ่ง (จำนวนคู่ของ Input กับ Output ที่ต้องการให้โครงข่ายรู้จัก) เอาท์พุทที่คำนวณได้จากโครงข่ายจะถูกเปรียบเทียบกับความสอดคล้องกับเป้าหมาย ค่าผิดพลาดที่เกิดขึ้นจะถูกป้อนกลับไปยังโครงข่ายและเปลี่ยนแปลงค่าเวจท์ให้สอดคล้องกับอัลกอริทึม ที่ทำให้แนวโน้มของค่าผิดพลาดที่เกิดขึ้นระหว่างเอาท์พุทกับเป้าหมายโดยเฉลี่ยมีค่าลดต่ำลง ตัวอย่างการเทนนิ่งแบบนี้ ได้แก่ การเทรนนิ่งแบบแพร่กลับ (Back propagation)

การเทรนนิ่งแบบอิสระ (Unsupervised Training)

ถึงแม้ว่าอัลกอริทึมแบบควบคุม (Supervised training) สามารถจะประยุกต์ใช้เพื่อปรับคุณลักษณะของโครงข่ายได้สำเร็จ แต่ก็ยังมีข้อวิจารณ์อยู่ คือ มันเป็นไปอย่างแบบชีวภาพไม่ได้ และยากที่จะเชื่อได้ว่า กลไกการเทรนนิ่งของสมองจะต้องการ การเปรียบเทียบระหว่างค่าที่ต้องการกับเอาท์พุทจริง โดยกระบวนการป้อนกลับไปแก้ไขคุณลักษณะของโครงข่าย และถ้าสมมติว่า ถ้าสมองมีกลไกเช่นนี้ ต้องมีผู้หาเอาท์พุทที่ต้องการเพื่อนำมาเป็นเป้าหมายตลอดเวลา และจะเอามาจากที่ใด ? สรุปคือ ต้องมีผู้คิดเป้าหมายให้กับโครงข่ายก่อน โครงข่ายไม่สามารถคิดและปรับคุณลักษณะได้ก่อนด้วยตนเอง ในทางตรงกันข้ามหากพิจารณาทารกแรกเกิดสมองของเขาสามารถจัดระบบเองได้อย่างไร? การเทรนนิ่งแบบอิสระ (Unsupervise learning) ที่สร้างขึ้นคงยังห่างไกลความเป็นไปได้ ที่จะมีลักษณะการเทรนนิ่งแบบระบบของสมอง จนกระทั่งมีการพัฒนาการ เทรนนิ่งแบบอิสระนี้ขึ้นราวปี 1984 โดยโคโฮเนน (Kohonen) และบุคคลอื่นๆ โดยได้เสนอแนวคิดที่เป็นการเทรนนิ่งแบบไม่ต้องการเป้าหมาย ไม่มีการตัดสินใจด้วยเหตุผลในอุดมคติมาก่อน ชุดของการเทรนนิ่ง จะมีเพียงอินพุท เวกเตอร์เท่านั้น เทรนนิ่งอัลกอริทึมจะเปลี่ยนแปลงค่าเวจท์ของ โครงข่าย เพื่อสร้างเอาท์พุทที่มีความมั่นคง ยกตัวอย่าง เช่น หากให้โครงข่ายรู้จำภาพหน้าคนหนึ่ง หากภาพหน้าคนคนนั้น เปลี่ยนแปลงไปเล็กน้อย (Image อาจมี Noise ร่วมอยู่บ้าง) โครงข่ายนั้นก็ยังสามารถบอกได้ว่าคนคนนั้นเป็นคนเดิมเป็นต้น การเทรนนิ่งจะไม่มีการตัดสินใจมาก่อน ไม่มีการกำหนดแบบเอาท์พุทมาก่อน (อาจกล่าวได้ว่าแบบเอาท์พุทจะถูกกำหนดโดยอินพุทเวกเตอร์นั่นเอง) ดังนั้น เอาท์พุทของโครงข่ายก็เช่นกัน ส่วนใหญ่จะถูกแปรรูปซึ่งจะเข้าใจได้ภายหลังกระบวน การเทรนนิ่ง ดังนั้นจึงไม่สามารถแก้ปัญหาที่เคร่งครัดสำคัญได้ แต่มักนิยมใช้โครงข่ายแบบนี้กับงานง่ายๆ ประเภทการเปรียบเทียบเอกลักษณ์, รูปแบบที่สัมพันธ์กันระหว่างอินพุท-เอาท์พุท ที่ถูกกำหนดโดยโครงข่าย

วิธีการแก้ปัญหาการฝึกสอน (Traning Algorithm)

ส่วนใหญ่แล้วทุกวันนี้ การแก้ปัญหาฝึกสอนของโครงข่ายค่อยๆ พัฒนาก้าวหน้าขึ้นจากแนวความคิดของ ดี โอ เฮบบ์ (ปี 1961) เขาได้เสนอโมเดลของ การเทรนนิ่ง แบบอิสระ (Unsupervised training) ในแบบซินแนปติกส์สเตรงท์ หรือเวจท์ ซึ่งจะเพิ่มขึ้น ถ้าทั้งแหล่งกำเนิด (Input Source)และจุดหมายปลายทาง (Destination)ของนิวรอลได้รับการสนใจ กรณีนี้ถ้ามีการใช้งานทางเส้นนี้บ่อยๆก็จะทำให้ซินแนปติกส์สเตรงจ์(หรือเวจท์)แข็งแรงขึ้น (เซลล์สมองที่ใช้งานมากบ่อยๆก็จะทำให้ ซินแนปส์ ใหญ่ขึ้น การส่งผ่านข้อมูล พัลล์ไฟฟ้าทำได้ดีขึ้น ทำให้มีประสิทธิภาพดีขึ้น เช่นสามารถคิดหรือจดจำได้เร็วและดีขึ้น) โครงข่าย ประสาทเทียมนี้ ใช้การเรียนรู้แบบเฮบเบน (Hebbian learning) จะเพิ่มค่าเวจท์ของโครงข่ายอย่างสอดคล้องกับผลคูณของระดับความสนใจของแหล่งกำเนิดและจุดหมายของนิวรอล ตามสมการดังนี้

$$W_{ij}(n+1) = W_{ij}(n) + \alpha OUT_i OUT_j \quad \text{.....(4.7)}$$

โดย $W_{ij}(n)$ คือค่าเวจท์ จากนิวรอล i ไปยังนิวรอล j ก่อนปรับปรุงค่า

$W_{ij}(n+1)$ คือค่าเวจท์ จากนิวรอล i ไปยังนิวรอล j หลังปรับปรุงค่า

α คือค่าคงที่ของการเรียนรู้ (Learning rate coefficient)

OUT_i คือเอ้าท์พุทของนิวรอล i และเป็นอินพุทของนิวรอล j

OUT_j คือเอ้าท์พุทของนิวรอล j

โครงข่ายที่มีลักษณะการเทรนนิ่งแบบ เฮบเบน นั้น เป็นผลมาจากการพัฒนามาแล้วกว่า 20-30 ปี โดยเฉพาะงานของโรเซนเบลทท์ (Rosenblatt:1962), วิโดรว์(Widrow:1959), วิโดรว์และ ฮอฟฟ์ (Widrow&Hoff:1960) และอีกหลายๆ คนที่พยายามพัฒนาระบบการเทรนนิ่งแบบควบคุม ที่สร้างโครงข่ายที่สามารถเรียนรู้แบบของอินพุทได้อย่างกว้างขวางและมีอัตราการเรียนรู้สูง ที่ บรรลุผลได้จากหลักการพื้นฐานของการเรียนรู้หรือเทรนนิ่งให้กับโครงข่ายเช่น Perceptrons, Hopfieldnets, Backpropagation Networks และCounter propagation.

ทฤษฎีหลักการแพร่ย้อนกลับ

หลักการแพร่ย้อนกลับได้รับการพัฒนาโดยรูเมลฮาร์ทและถูกนำเสนอในช่วงปี ค.ศ.1986s] หลักการแพร่ย้อนกลับนี้สามารถนำไปใช้แก้ปัญหาในลักษณะเชิงเส้น (linear) และปัญหาที่ไม่เชิงเส้นได้ (nonlinear) ทำให้ประยุกต์เข้ากับโครงงานต่างๆ ได้อย่างหลากหลายและได้รับความนิยมมากที่สุดในปัจจุบัน หลักการแพร่ย้อนกลับมีพื้นฐานมาจากกฎเดลต้า (delta-rule) ที่พัฒนาขึ้นโดย วิดโรว์ และ ฮอฟฟ์ ในช่วงปี ค.ศ.1960-1962 ซึ่งกระบวนการของกฎเดลต้าจะลดค่าผิดพลาดที่เอาต์

พุทของโครงข่ายเปรียบเทียบกับค่าเป้าหมายโดยค่าผิดพลาดที่ลดลงนี้เกิดจากการปรับแต่งค่าน้ำหนักที่อินพุทของแต่ละนิวรอน

สถาปัตยกรรมของโครงข่ายบีพีเอ็น

โครงข่ายบีพีเอ็น (BPN,Back,Propagation Neural Network) เป็นโครงข่ายที่มีจำนวนชั้นมากกว่า 2 ชั้นคือมีได้ตั้งแต่ 3 ชั้นขึ้นไปคือ ชั้นอินพุท (input layer) ชั้นซ่อน (hidden layer) และชั้นเอาต์พุท (output layer) ลักษณะของโครงข่ายบีพีเอ็น ซึ่งเป็นฟีดฟอร์เวิร์ดเน็ทเวิร์ค (feedforword network) โดยมีการเชื่อมโยงในแต่ละชั้นแบบฟูลลี่ นั่นคือทุกๆ นิวรอนในชั้นอินพุทจะส่งสัญญาณไปยังทุกๆ นิวรอนในชั้นซ่อนชั้นแรก และทุกๆนิวรอนในชั้นซ่อนชั้นแรกจะส่งสัญญาณไปยังทุกๆ นิวรอนในชั้นถัดไปจนในที่สุดทุกๆ นิวรอนในชั้นซ่อนชั้นสุดท้ายจะส่งสัญญาณไปยังทุกๆ นิวรอนในชั้นเอาต์พุท จากภาพแสดงโครงข่ายที่มี 3 ชั้น คือชั้นอินพุท ชั้นซ่อน และชั้นเอาต์พุท ในทางปฏิบัติที่ชั้นซ่อนสามารถที่จะมีได้มากกว่า 1 ชั้น

โครงข่ายประสาทเทียมเรียกได้ว่าเป็นแมพปิ้งเน็ทเวิร์ค (mapping network) ถ้าหากโครงข่ายสามารถคำนวณฟังก์ชั่นในบางลักษณะที่กำเนิดความสัมพันธ์ระหว่างอินพุทและเอาต์พุทในโครงข่าย ตัวอย่างเช่น ถ้ามีอินพุทป้อนให้กับโครงข่ายซึ่งมีค่าเป็นมุม (angle) และเอาต์พุทคือค่าโคซายน์ (cosine) ของมุมที่อินพุท โครงข่ายจะมีการแมพปิ้งคือ $\theta \longrightarrow \cos(\theta)$ สิ่งสำคัญที่ทำให้การแมพปิ้งถูกต้องนั้นจำเป็นต้องมีการฝึกหัดโครงข่าย และการฝึกหัดโครงข่ายโดยอาศัยหลักการแพร่ย้อนกลับนั้น แบ่งออกได้เป็นสามขั้นตอนคือ

1. แพร่สัญญาณที่ชั้นอินพุทไปยังชั้นซ่อนแรก และเพร่สัญญาณไปยังชั้นซ่อนถัดไปจนถึงชั้นเอาต์พุทโดยผ่านฟังก์ชั่นกระตุ้น
2. คำนวณค่าผิดพลาดของค่าเอาต์พุทกับค่าเป้าหมายและแพร่ย้อนกลับค่าผิดพลาดเหล่านี้ไปยังชั้นซ่อน
3. ปรับแต่งค่าน้ำหนักที่ชั้นเอาต์พุทและชั้นซ่อนเพื่อลดค่าผิดพลาดโดยทำให้ค่าเอาต์พุทมีค่าใกล้เคียงหรือเท่ากับค่าเป้าหมาย

การฝึกหัดโครงข่ายด้วยวิธีการแพร่ย้อนกลับ

ซึ่งในวิทยานิพลธีนี้ใช้อินพุทเลเยอร์ 512 โนด,ฮิดเดนเลเยอร์ 100 โนดและเอาต์พุทเลเยอร์ 72 โนดเริ่มต้นด้วยการพิจารณาโครงข่ายขนาด 3 ชั้น ที่มีอินพุทเวกเตอร์ $X_p = (X_{P1}, X_{P2} \ldots\ldots, X_{Pn})t$ อินพุท X_p นี้จะถูกป้องให้กับชั้นอินพุทของโครงข่ายและที่ชั้นอินพุทนี้จะกระจายค่าไปยัง นิวรอลในชั้นซ่อน ที่ชั้นซ่อนนี้เองจะทำการคำนวณค่าเน็ท สมการหาค่าเน็ทที่นิวรอน j ในชั้นซ่อนคือ

$$net^{h}_{pj} = \sum_{i=1}^{n} W^{h}_{ji} X_{pi} + \theta_{hj} \qquad (4.8)$$

เมื่อ $W^{h}_{j\,I}$ คือน้ำหนักที่ชั้นซ่อนตรงจุดต่อจากโหนดอินพุท I ไปยังนิวรอน j และ θ_{hj} คือค่าเทอมไบแอส เมื่อได้ค่าเน็ทแล้วขั้นถัดไปคือคำนวณค่าเอาต์พุทของแต่ละนิวรอนในชั้นซ่อนโดยคำนวณได้ดังนี้

$$I_{pj} = f^{h}_{j}(net^{h}_{pj}) \qquad (4.9)$$

ใช้สมการทั้งสองคำนวณค่าในชั้นซ่อนทุกๆ ชั้น จากนั้นจึงทำการคำนวณค่าเน็ทและค่าเอาต์พุทที่ชั้นเอาต์พุทโดยใช้สมการที่คล้ายคลึงกันคือ

$$net^{o}_{pj} = \sum_{I=1}^{L} W^{o}_{KI}\, i_{pi} + \theta o_{K} \qquad (4.10)$$

$$0o_{K} = f^{o}_{K}(net^{o}_{pk}) \qquad (4.11)$$

การหาค่าผิดพลาดและการปรับน้ำหนักที่ชั้นเอาต์พุท

สมการปรับค่าน้ำหนักที่ชั้นเอาต์พุทแสดงได้ดังนี้

$$W^{o}_{kj}(t+1) = W^{o}_{kj}(t) + \Delta_{p} W^{o}_{kj}(t) \qquad (4.12)$$

$$\Delta W^{o}_{kj} = \eta (Y_{pk} - O_{pk}) f^{o}_{K}(net^{o}_{pk})\, i_{pj} \qquad (4.13)$$

$$I_{pj} = f^{h}_{j}(net^{h}_{pj}) \qquad (4.14)$$

องค์ประกอบ η คืออัตราการเรียนรู้ โดยปกติค่านี้จะมีค่าเป็นบวกและน้อยกว่า 1 ส่วน Y_{pk} คือค่าเป้าหมาย ลองกลับไปดูฟังก์ชั่นกระตุ้น f^{o}_{k} และ f^{h}_{j} ซึ่งมี 2 รูปแบบคือ

$$f^{o}_{k}(net^{o}_{jk}) = net^{o}_{jk} \qquad (4.15)$$

$$f^{o}_{k}(net^{o}_{jk}) = (1 + \exp(-net^{o}_{jk}))^{-1} \qquad (4.16)$$

ฟังก์ชั่นกระตุ้นรูปแบบแรกเป็นการกำหนดเอาต์พุทเป็นแบบลิเนียร์ ส่วนฟังก์ชั่นรูปแบบหลังเรียกว่าไบนารี่ซิกมอยด์ (binary sigmoid) ถ้าเลือกฟังก์ชั่นรูปแบบแรก $f^{O}_{k}' = 1$ ถ้าเลือกรูปแบบที่ 2 ฟังก์ชั่น $f^{O}_{k}' = f^{O}_{k}(1- f^{O}_{k}) = 0_{pk}(0_{pk})$ ดังนั้นการปรับน้ำหนักในกรณีของลิเนียร์จะกำหนดได้ดังนี้

$$W^{O}_{kj}(t+1) = W^{O}_{kj}(t) + \eta(Y_{pk} - O_{pk}) \quad I_{pj} \tag{4.17}$$

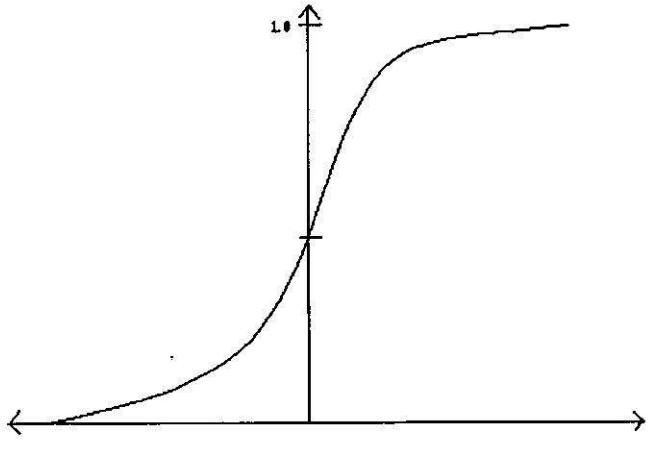
ส่วนฟังก์ชั่นไบนารีซิกมอยด์กำหนดได้ดังนี้

$$W^{O}_{kj}(t+1) = W^{O}_{kj}(t) + \eta(Y_{pk} - O_{pk}) O_{pk}(1- O_{pk}) I_{pj} \tag{4.18}$$

เขียนสมการให้ดูง่ายขึ้นได้ดังนี้

$$\sigma^{O}_{pk} = (Y_{pk} - O_{pk}) f^{O}_{k}'(net^{O}_{jk}) \tag{4.19}$$

$$W^{O}_{kj}(t+1) = W^{O}_{kj}(t) + \eta\, \sigma^{O}_{pk} I_{pj} \tag{4.20}$$



ภาพที่ 4.8 แสดงกราฟในลักษณะ S-shape ของฟังก์ชั่นไบนารีซิกมอยด์

บทที่ 5

ขบวนการเรียนรู้

เนื้อหาในบทนี้จะกล่าวถึงขั้นตอนต่างๆ ที่ทำในงานวิจัยนี้ โดยจะแบ่งขั้นตอนออกเป็น 3 ขั้นตอนดังนี้ คือ

ขั้นตอนการเตรียมข้อมูล (Preprocessing) เป็นขั้นตอนที่รับข้อมูลบิตแม็บของตัวอักษร แล้วนำไปคำนวณหาลักษณะเด่น (Feature) ของแต่ละตัวอักษร จากนั้นกำหนดโครงสร้างของตัวอักษรนั้นๆ เพื่อป้อนให้กับขั้นตอนฝึกสอนและรู้จำต่อไป

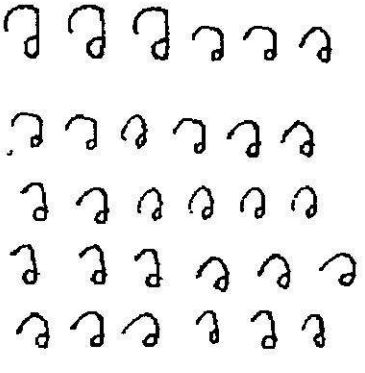
ขั้นตอนการฝึกสอน (Training) เป็นขั้นตอนที่นำเอาโครงสร้างข้อมูลที่ได้จากขั้นตอนแรกมาฝึกสอนให้คอมพิวเตอร์รู้ว่าตัวอักษรแต่ละตัวมีโครงสร้างอย่างไร โดยจะใช้โครงข่ายนิวรอล

ขั้นตอนการรู้จำ (Recognizing) เป็นขั้นตอนที่นำเอาโครงสร้างตัวอักษรที่ต้องการจะทดสอบมาเปรียบเทียบกับโครงสร้างของข้อมูลที่มีอยู่ในฐานข้อมูล

5.1 ขั้นตอนการเตรียมข้อมูล (Preprocessing)

5.1.1 ลักษณะของตัวอักษรที่ใช้ในการทดลอง

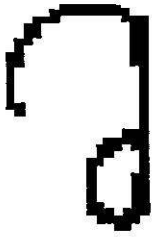
ในงานวิจัยนี้ เป็นการรู้จำตัวอักษรคัดลายมือเขียนภาษาไทย โดยจะทำการรู้จำเฉพาะตัวพยัญชนะทั้งหมด 44 ตัว โดยมีผู้เขียนทั้งหมด 10 คน เขียนคนละ 3 ครั้ง แสดงลักษณะของพยัญชนะ ว ได้ตามภาพที่ 5.1



ภาพที่ 5.1 แสดงลักษณะของของพยัญชนะ ว

5.1.2 ปรับข้อมูลให้อยู่ในรูปเมตริก

ทำการแยกตัวอักษรแต่ละตัวออกจากกัน แล้วทำการปรับขนาดข้อมูลให้มีขนาดเท่ากับ 32x16 บิต แสดงได้ดังภาพที่ 5.2



ภาพที่ 5.2 ตัวอย่างของตัวอักษรที่ลดขนาดเหลือ 32 x 16 บิต

กำหนดจุดของข้อมูล ซึ่งได้จากการกวาดตามแนวแถวแสดงไว้ในภาพที่ 5.3 ดังนั้นจำนวนข้อมูลของตัวอักษร 1 ตัวจะเท่ากับ 512 ข้อมูล

1	2	.	.	16
17	18	.	.	32
.				.
.				.
.				.
.				.
.				.
.				.
.				.
496	487	.	.	512

32x16

ภาพที่ 5.3 แสดงขนาดของตัวอักษร

นำตัวอักษรทุกตัวของแต่ละพยัญชนะ จัดทำเป็นเมตริก X กำหนดให้ χ_{ij} คือเวกเตอร์ของตัวอักษร โดยที่ i = 1,2,3...,N และ j = 1,2,3...,M ซึ่ง M คือจำนวนตัวอักษรทุกตัวในพยัญชนะนี้ และ N คือ จำนวนข้อมูลของตัวอักษร 1 ตัว ดังนั้นลักษณะเมตริก ที่จะนำไปทำในขั้นตอนต่อไป จะแสดงดังภาพที่ 5.4

$$X = \begin{bmatrix} x_{11}, x_{12}, x_{13}, \cdots\cdots x_{1M} \\ x_{21} \\ x_{31} \\ . \\ . \\ . \\ x_{N1}, x_{N2}, x_{N3}, \cdots\cdots x_{NM} \end{bmatrix} \ldots\ldots NxM$$

ภาพที่ 5.4 แสดงเมตริกก่อนทำการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ

5.1.3 หาลักษณะร่วมของข้อมูล โดยการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ

จากข้อมูลที่อยู่ในรูปของเมตริก X เราจะทำการหาลักษณะร่วมของข้อมูลโดยใช้การแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ จะมีขั้นตอนการทำดังนี้

ขั้นตอนที่ 1 นำเอาเมตริกซ์ X มาแปลงเป็นโคแวเรียนท์เมตริกซ์ ตามสมการที่ (1)

$$C = E\{(x_j - \overline{m}_j)(x_j - \overline{m}_j)^T\} \ldots\ldots\ldots (5.1)$$

เมื่อ $\overline{m} = E\{x\}$ คือเวกเตอร์เฉลี่ยในที่นี้จะกำหนดให้เป็น 0

ขั้นตอนที่ 2 คำนวณหาค่าไอเก็นเวกเตอร์ที่มีค่าไอเก็น (Eigenvalue) มากที่สุดของโคแวเรียนท์เมตริก C ตามสมการ ที่ 2

$$CV = \lambda_{max} V \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (5.2)$$

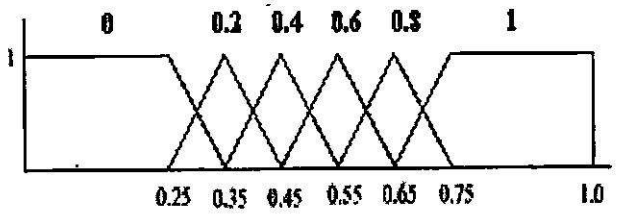
V คือ ไอเก็นเวกเตอร์ ซึ่งจะมีขนาดเท่ากับ Nx1 และ λ_{max} คือค่าไอเก็นที่มีค่ามากที่สุด

ข้อมูลไอเก็นเวกเตอร์ที่เราได้ ก็คือข้อมูลที่แสดงลักษณะร่วมของเวกเตอร์ต่างๆ ในเมตริก X นั้นก็หมายถึงลักษณะร่วมของ ตัวพยัญชนะแต่ละตัวที่มีรูปแบบต่างๆ กันนั้นเอง แต่ก่อนที่จะนำไปเป็นตัวแทนสำหรับเป็นอินพุตให้กับโครงข่ายนิวรอล ต้องทำการปรับข้อมูลให้อยู่ในช่วง 0-1 ก่อน ตามสมการที่ 5.3

$$V_i' = \frac{V_i - \min V}{\max V - \min V} \quad \text{.....(5.3)}$$

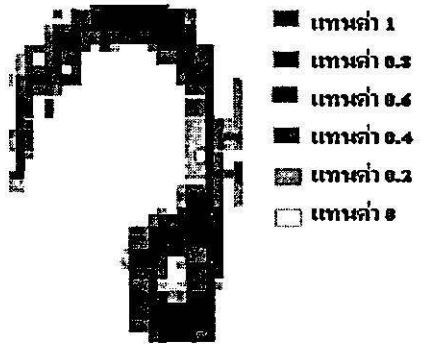
5.1.4 แปลงข้อมูลให้อยู่ในระบบฟัซซี่เซต

การแบ่งระดับค่าความเป็นสมาชิกในงานวิจัยนี้จะกำหนดช่วงของฟัซซี่ต่าง ๆ ตามภาพที่ 5.5



ภาพที่ 5.5 แสดงการแบ่งระดับฟัซซี่สำหรับปรับข้อมูล

นำข้อมูลที่ได้จากหัวข้อ 5.1.3 หาค่าความเป็นสมาชิก โดยใช้ค่า min จากข้อมูลของตัวพยัญชนะ ว เราจะได้ข้อมูลที่ถูกปรับให้อยู่ในระบบของฟัซซี่ แสดงข้อมูลนี้โดยจัดขนาดเมตริกให้มีขนาด 32x16 ได้ตามภาพที่ 5.6

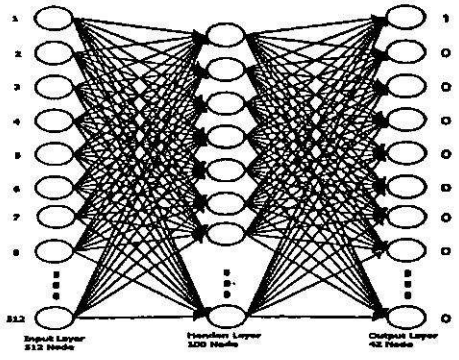


ภาพที่ 5.6 แสดงพยัญชนะ ว ที่ปรับข้อมูลตามระบบฟัซซี่

ข้อมูลที่เราได้ในขั้นตอนนี้จะนำไปเป็นตัวแทนของตัวอักษรทั้งหมดในพยัญชนะนี้ เข้าสู่ขั้นตอนการเรียนรู้ โดยจะมีขนาดจำนวนข้อมูลทั้งหมด 512 ข้อมูล ต่อพยัญชนะแต่ละตัว จะเห็นได้ว่าระบบในงานวิจัยนี้ จะไม่ใช้ข้อมูลของตัวอักษรทุกตัวไปทำการฝึกสอน จะใช้เพียงแค่ ข้อมูลที่เป็นตัวแทนเท่านั้น จะทำให้ลดขนาดข้อมูลที่ใช้ในการฝึกสอนลงได้มาก และทำให้เวลาที่ใช้น้อยลงมาก จะแสดงผลไว้ในส่วนของผลการทดลอง

5.2 ขั้นตอนการฝึกสอน (Training)

ในงานวิจัยนี้ได้เลือกวิธีโครงข่ายประสาทเทียมแบบแพร่กลับ(Back-Propagation Neural Network) ที่มี 3 ชั้น คือ อินพุทเลเยอร์ (Input Layer) ฮิดเดนเลเยอร์ (Hidden Layer) และ เอาท์พุทเลเยอร์ (Output Layer) มีลักษณะดังแสดงในภาพที่ 5.7



ภาพที่ 5.7 แสดงโครงสร้างของโครงข่ายประสาทเทียม

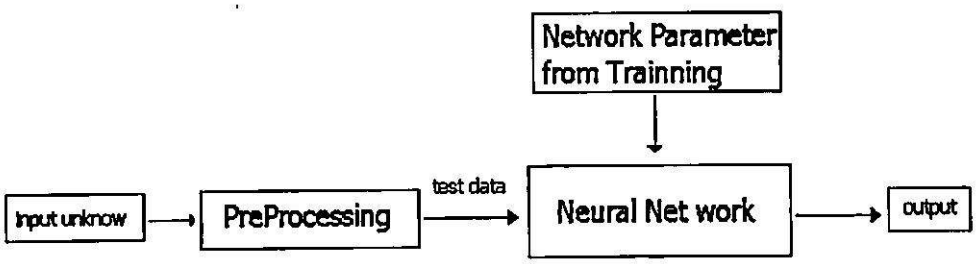
โดยที่ชั้นรับข้อมูลก็คือข้อมูลที่ได้จากขั้นตอนการเตรียมข้อมูล ดังนั้นจำนวนโหนดของอินพุทเลเยอร์ และ เอาท์พุทเลเยอร์ เท่ากับ 512 และ 44 ตามลำดับ สำหรับจำนวนโหนดของฮิดเดนเลเยอร์ (Hidden Layer Node) ได้ทดลองทั้งหมด 5 แบบคือ จำนวนโหนดของฮิดเดนเลเยอร์เท่ากับ 100,200,300,400 และ 500 ตามลำดับ ซึ่งการทดลองได้ กำหนดให้ ค่าถ่วงน้ำหนัก(weight)เริ่มต้นโดยการสุ่ม อัตราในการเรียนรู้ (Learning Rate) เท่ากับ 0.1 และ ค่าความผิดพลาด (Sum-Square Error) เท่ากับ 0.01

ผลลัพธ์จากการทดลองเพื่อหาจำนวนโหนดที่เหมาะสมของฮิดเดนเลเยอร์ ที่ให้ค่าความถูกต้องและเวลาในการเรียนรู้ (tranning) ที่เหมาะสมที่สุด มีค่าเท่ากับ 100 โหนด

นำค่าถ่วงน้ำหนักที่ได้จากการฝึกสอน (Final weight) เก็บไว้ในหน่วยความจำเพื่อเข้าสู่ขั้นตอนต่อไป และเพื่อทดสอบหาความถูกต้อง

5.3 ขั้นตอนการรู้จำ (Recognizing)

โครงสร้างการทำงานในขั้นตอนของการทดสอบแสดงไว้ในภาพที่ 5.8



ภาพที่ 5.8 แสดงโครงสร้างของการทดสอบ

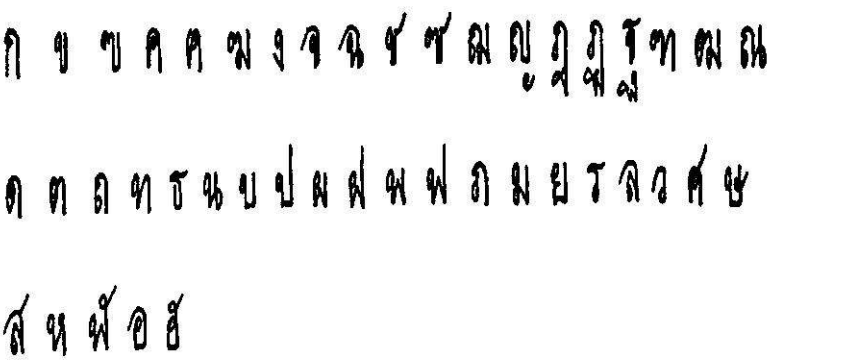
ข้อมูลที่ต้องการทดสอบ (Input Unknow) จะต้องผ่านขั้นตอนการเตรียมข้อมูลก่อน โดยการแยกตัวอักษรออกมาทดสอบที่ละตัวทำการปรับขนาดข้อมูลให้มีขนาดเท่ากับ 32x16 แล้วจัดเรียงข้อมูลใหม่ ซึ่งจะได้ข้อมูลที่มีขนาดเท่ากับ 512x1 เป็นข้อมูลสำหรับใช้ในการทดสอบ (test data) จากนั้นนำไปกระทำกับค่าถ่วงน้ำหนัก (weight) ที่ได้จากการเรียนรู้ (tranning) ก็สามารถที่จะวิเคราะห์ input unknow ได้

บทที่ 6

การทดสอบและประเมินผล

6.1 การทดสอบและประเมินผลความถูกต้องใน การจดจำตัวอักษรคัดลายมือเขียนภาษาไทยโดย ประยุกต์ใช้การทานส์ฟอร์มแบบคาร์ฮูเน็น เข้ากับโครงข่ายฟัซซี่นิวรอล

ในหัวข้อนี้จะกล่าวถึง การทดสอบ ความสามารถในการรู้จำตัวอักษรคัดลายมือเขียนภาษาไทย ตัวอย่างของภาพพยัญชนะ ก-ฮ ที่ใช้ในการทดลอง แสดงไว้ในภาพที่ 6.1



ภาพที่ 6.1 แสดงภาพของพยัญชนะ ก-ฮ จากผู้คัดลายมือ 1 คน ครั้งที่ 1

ตัวอักษรทุกตัวในแต่ละพยัญชนะจะนำไปหาลักษณะร่วมของข้อมูล โดยการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ และผ่านการปรับข้อมูลนี้ให้อยู่ในแบบของฟัซซี่เซต ผลลัพธ์ที่ได้ นี้คือ ตัวแทนข้อมูลของแต่ละพยัญชนะ ซึ่งจะใช้เป็นอินพุทในขั้นตอนการฝึกสอน โดยใช้โครงข่านนิวรอลแบคพรอบพาเกชั่น ซึ่งจะมีทั้งหมด 3 เลเยอร์ คือ อินพุตเลเยอร์ ฮิดเดนเลเยอร์ และ เอาท์พุทเลเยอร์ โดยกำหนดจำนวนโหนดในแต่ละเลเยอร์ คือ อินพุตเลเยอร์เท่ากับ 512 โหนด, เอาท์พุตเลอเยอร์ เท่ากับ 44 โหนด สำหรับการทดลองขั้นต้นของการนำโครงข่ายนิวรอลมาทำการรู้จำ ต้องทดลองหาจำนวนโหนดของ ฮิดเดนเลเยอร์ ที่เหมาะสมก่อน โดยกำหนดให้จำนวนฮิดเดนเลเยอร์ทั้งหมด 5 ค่า ดังนี้ 100, 200, 300 , 400 และ 500 จากการทดลองโดยการตั้งค่าอัตราการเรียนรู้ (Learning Rate) เท่ากับ 0.1 และค่าความผิดพลาด (Sum-Square Error) เท่ากับ 0.01 ได้ผลการทดสอบตามตารางที่ 1

ตารางที่ 6.1 ผลการทดสอบโครงข่ายนิวรอลโดยมีฮิดเดนโหนด 100,200,300, 400, 500

จำนวน ฮิดเดนเลเยอร์	เวลาที่ใช้ในการฝึกสอน(ชม.)	ความถูกต้อง(%)
100	6	92.05
200	6.5	90.55
300	6.8	85.22
400	8	91.42
500	7	90.23

ดังนั้นในระบบนี้ จำนวนโหนดของฮิดเดนโหนด ที่ให้ค่าความถูกต้อง และเวลาในการเรียนรู้ ที่เหมาะสมที่สุดเท่ากับ 100 โหนด

การทดลองในลำดับต่อไป เราได้ทำการทดสอบความถูกต้องในการรู้จำ โดยจะทดสอบกับข้อมูลตัวอักษรเดียวกับที่ใช้ในการเตรียมข้อมูลของระบบ และจากผู้คัดลายมือชุดใหม่ ผู้คัดลายมือ 10 คน คนละ 3 ครั้ง แสดงผลการทดลองตามตารางที่ 6.2 และ 6.3 ตามลำดับ

ตารางที่ 6.2 ผลการทดสอบความถูกต้องของตัวอักษรชุดเดียวกับที่ใช้ในการเตรียมข้อมูลของระบบ

พยัญชนะ	จำนวน	รู้จำผิดพลาด	ความถูกต้องร้อยละ (%)	พยัญชนะ	จำนวน	รู้จำผิดพลาด	ความถูกต้องร้อยละ(%)
ก	30	0	100.00	ท	30	1	96.67
ข	30	6	80.00	ธ	30	8	73.33
ฃ	30	7	76.67	น	30	0	100.00
ค	30	7	76.67	บ	30	1	96.67
ฅ	30	6	80.00	ป	30	0	100.00
ฆ	30	8	73.33	ผ	30	2	93.33
ง	30	0	100.00	ฝ	30	0	100.00
จ	30	4	86.67	พ	30	0	100.00
ฉ	30	0	100.00	ฟ	30	1	96.67
ช	30	5	83.33	ภ	30	1	96.67
ซ	30	6	80.00	ม	30	0	100.00
ฌ	30	1	96.67	ย	30	0	100.00

ตารางที่ 6.2 (ต่อ)

พยัญชนะ	จำนวน	รู้จำผิดพลาด	ความถูกต้องร้อยละ (%)	พยัญชนะ	จำนวน	รู้จำผิดพลาด	ความถูกต้องร้อยละ(%)
ญ	30	1	96.67	ร	30	4	86.67
ฎ	30	6	80.00	ล	30	0	100.00
ฏ	30	1	96.67	ว	30	0	100.00
ฐ	30	2	93.33	ศ	30	5	83.33
ฑ	30	0	100.00	ษ	30	4	86.67
ฒ	30	3	90.00	ส	30	1	96.67
ณ	30	1	96.67	ห	30	2	93.33
ด	30	4	86.67	ฬ	30	0	100.00
ต	30	0	100.00	อ	30	3	90.00
ถ	30	4	86.67	ฮ	30	0	100.00
รวม	1320	105	92.05				

ตารางที่ 6.3 ผลการทดสอบความถูกต้องของตัวอักษร จากผู้เขียนชุดใหม่

พยัญชนะ	จำนวน	รู้จำผิดพลาด	ความถูกต้องร้อยละ (%)	พยัญชนะ	จำนวน	รู้จำผิดพลาด	ความถูกต้องร้อยละ(%)
ก	30	5	83.33	ท	30	0	100.00
ข	30	5	83.33	ธ	30	2	93.33
ฃ	30	7	76.67	น	30	4	86.67
ค	30	6	80.00	บ	30	0	100.00
ฅ	30	8	73.33	ป	30	0	100.00
ฆ	30	5	83.33	ผ	30	0	100.00
ง	30	0	100.00	ฝ	30	0	100.00
จ	30	7	76.67	พ	30	1	96.67
ฉ	30	1	96.67	ฟ	30	2	93.33
ช	30	8	73.33	ภ	30	0	100.00
ซ	30	9	70.00	ม	30	1	96.67

ตารางที่ 6.3 (ต่อ)

พยัญชนะ	จำนวน	รู้จำผิดพลาด	ความถูกต้องร้อยละ (%)	พยัญชนะ	จำนวน	รู้จำผิดพลาด	ความถูกต้องร้อยละ(%)
ฌ	30	1	90.00	ย	30	6	86.67
ญ	30	6	96.67	ร	30	6	86.67
ฎ	30	1	96.67	ว	30	0	100.00
ฐ	30	0	100.00	ศ	30	8	73.33
ฑ	30	0	100.00	ษ	30	4	86.67
ฒ	30	4	86.67	ส	30	4	86.67
ณ	30	1	96.67	ห	30	1	96.67
ค	30	6	80.00	ฬ	30	8	73.33
ต	30	6	80.00	อ	30	5	83.33
ถ	30	1	96.67	ฮ	30	1	96.67
รวม	1320	145	89.02				

จากตารางที่ 6.2 และ 6.3 จะได้อัตราเฉลี่ยนของ ความถูกต้องในการรู้จำเฉลี่ย เท่ากันร้อยละ 90.53 ถือว่ามีความถูกต้องในการรู้จำสูง เมื่อเทียบกับเวลาที่ใช้ในการฝึกสอน ในที่นี้จะใช้เวลาในการฝึกสอนทั้งหมดโดยประมาณ 6 ชั่วโมง

6.2 เปรียบเทียบการทำงานของระบบนี้ในลักษณะต่างๆ กัน

เนื่องจากผลลัพธ์ที่ได้จากการหาลักษณะร่วมของตัวอักษรทุกตัวในแต่ละพยัญชนะ โดยใช้การแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ ก็สามารถนำเอามาเป็นอินพุทให้กับขั้นตอนการฝึกสอนโดยที่เราไม่ต้องปรับข้อมูลที่ได้ให้อยู่ในระบบของฟัซซี่เซต จากการทดลองหาค่าความถูกต้องในส่วนที่ไม่ได้ใช้ระบบฟัซซี่เข้ามาช่วย แสดงได้ตามตารางที่ 6.4 และ 6.5

ตารางที่ 6.4 ผลการทดสอบความถูกต้องของตัวอักษรชุดเดียวกับที่ใช้ในการเตรียมข้อมูลของระบบ

พยัญชนะ	จำนวน	รู้จำผิดพลาด	ความถูกต้องร้อยละ (%)	พยัญชนะ	จำนวน	รู้จำผิดพลาด	ความถูกต้องร้อยละ(%)
ก	30	4	86.67	ท	30	4	86.67
ข	30	10	66.67	ธ	30	8	73.33
ฃ	30	7	76.67	น	30	0	100.00
ค	30	15	50.00	บ	30	1	96.67
ฅ	30	11	63.33	ป	30	0	100.00
ฆ	30	8	73.33	ผ	30	2	93.33
ง	30	0	100.00	ฝ	30	0	100.00
จ	30	7	76.67	พ	30	3	90.00
ฉ	30	1	96.67	ฟ	30	4	86.67
ช	30	10	66.67	ภ	30	1	96.67
ซ	30	6	80.00	ม	30	0	100.00
ฌ	30	1	96.67	ย	30	7	76.67
ญ	30	1	96.67	ร	30	4	86.67
ฎ	30	6	80.00	ล	30	0	100.00
ฏ	30	1	96.67	ว	30	0	100.00
ฐ	30	2	93.33	ศ	30	5	83.33
ฑ	30	0	100.00	ษ	30	4	86.67
ฒ	30	6	80.00	ส	30	1	96.67
ณ	30	4	86.67	ห	30	2	93.33
ด	30	9	70.00	ฬ	30	1	96.67
ต	30	7	76.67	อ	30	3	90.00
ถ	30	4	86.67	ฮ	30	1	96.67
รวม	1320	171	87.05				

ตารางที่ 6.5 ผลการทดสอบความถูกต้องของตัวอักษร จากผู้เขียนชุดใหม่

พยัญชนะ	จำนวน	รู้จำผิดพลาด	ความถูกต้องร้อยละ (%)	พยัญชนะ	จำนวน	รู้จำผิดพลาด	ความถูกต้องร้อยละ(%)
ก	30	5	83.33	ท	30	0	100.00
ข	30	8	73.33	ธ	30	2	93.33
ฃ	30	7	76.67	น	30	7	76.67
ค	30	10	66.67	บ	30	0	100.00
ฅ	30	8	73.33	ป	30	1	96.67
ฆ	30	5	83.33	ผ	30	0	100.00
ง	30	0	100.00	ฝ	30	1	96.67
จ	30	7	76.67	พ	30	5	83.33
ฉ	30	3	90.00	ฟ	30	6	80.00
ช	30	8	73.33	ภ	30	0	100.00
ซ	30	12	60.00	ม	30	1	96.67
ฌ	30	5	83.33	ย	30	4	86.67
ญ	30	3	90.00	ร	30	4	86.67
ฎ	30	9	70.00	ล	30	11	63.33
ฏ	30	4	86.67	ว	30	0	100.00
ฐ	30	2	93.33	ศ	30	10	66.67
ฑ	30	1	96.67	ษ	30	6	80.00
ฒ	30	8	73.33	ส	30	6	80.00
ณ	30	1	96.67	ห	30	3	90.00
ด	30	11	63.33	ฬ	30	10	66.67
ต	30	8	73.33	อ	30	7	76.67
ถ	30	1	96.67	ฮ	30	1	96.67
รวม	1320	211	84.02				

จากตารางที่ 6.4 และ 6.5 จะได้อัตราเฉลี่ยนของ ความถูกต้องในการรู้จำเฉลี่ย เท่ากับร้อยละ 85.53

จากผลการทดลองตามตารางนี้ 6.2 6.3 6.4 และ 6.5 นำมาสรุปใหม่ได้ตามตารางที่ 6.6

ตารางที่ 6.6 สรุปความถูกต้องการรู้จำทั้ง 2 ระบบ

ระบบ	จำนวนอักษรที่ทดสอบ	ความถูกต้องร้อยละ(%)
KL-T	1320	85.53
KL-T+Fuzzy	1320	90.53

จากตารางที่ 6.6 เราจะเห็นได้ชัดเจนว่า การนำเอาระบบฟัซซี่ เข้ามาใช้ในการปรับข้อมูลหลักจากขั้นตอนการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ จะทำให้มีค่าอัตราความถูกต้อง เพิ่มขึ้นถึง 5 เปอร์เซ็นต์ สาเหตุที่เป็นลักษณะนี้เพราะว่า ข้อมูลที่ได้จากขั้นตอนการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟนั้นจะมีค่าอยู่ในช่วงตั้งแต่ 0-1 แต่ถ้าเรานำมาปรับให้อยู่ในระบบฟัซซี่เซต เราจะเหลือจำนวนข้อมูลเพียงแค่ 6 ค่า คือ 0, 0.2, 0.4, 0.6, 0.8 และ 1 ก็เท่ากับว่าการนำฟัซซี่เข้ามาประยุกต์ใช้จะช่วยลดฟีเจอร์ (Feature) ของข้อมูลลงได้ แล้วยังให้ค่าความถูกต้องเพิ่มขึ้นมาอีก

ในงานวิจัยนี้ได้เปรียบเทียบจำนวนรูปแบบของอักษรที่ใช้ในขั้นตอนการเตรียมข้อมูล และการฝึกสอน โดยจะมีความแตกต่างรูปแบบดังนี้

1. ใช้ผู้คัดลายมือ 1 คน คัดตัวพยัญชนะตัวละ 30 ครั้ง
2. ใช้ผู้คัดลายมือ 5 คน คัดตัวพยัญชนะตัวละ 6 ครั้ง
3. ใช้ผู้คัดลายมือ 10 คน คัดตัวพยัญชนะตัวละ 3 ครั้ง

สำหรับข้อมูลที่ใช้ในการทดสอบ ใช้ผู้คัดลายมือชุดใหม่ 10 คน คัดตัวพยัญชนะทุกตัวคนละ 3 ครั้งได้ผลการทดลองตามตารางที่ 6.7 ดังนี้

ตารางที่ 6.7 แสดงผลความถูกต้องจำนวนผู้คัดลายมือที่ใช้ในการเตรียมข้อมูลและฝึกสอนที่ต่างกัน

จำนวนผู้คัดลายมือที่ใช้ในการเตรียมข้อมูลและฝึกสอน	จำนวนอักษรที่ใช้ทดสอบ	ความถูกต้องร้อยละ(%)
1 คน	1,320	85.15
5 คน	1,320	86.82
10 คน	1,320	89.02

จากผลการทดลองตามตารางที่ 6.7 แสดงให้เห็นว่าผลของความถูกต้องจะขึ้นอยู่กับรูปแบบที่มีความหลากหลายของตัวอักษรที่ใช้ในการเตรียมข้อมูล และฝึกสอน เพราะเหตุว่าการเตรียมข้อมูลจากตัวอักษรที่มีรูปแบบต่างกันน้อยข้อมูลที่เป็นตัวแทนแสดงลักษณะร่วมของตัวพยัญชนะแต่ละตัวก็จะไม่มีความยืดหยุ่นเพียงพอ แต่ถ้าพยัญชนะแต่ละตัวมีรูปแบบต่างกันมากขึ้นก็จะทำให้ลักษณะร่วมของข้อมูลที่ได้มีความครอบคลุมมากขึ้น เป็นผลให้อัตราการรู้จำสูงขึ้นตามลำดับ

ต่อไปจะทำการทดลองเปรียบเทียบระหว่าง ตัวอักษรจากการคัดลายมือแบบบรรจงกับไม่บรรจง ใช้สำหรับเตรียมข้อมูล และฝึกสอน สำหรับข้อมูลที่ใช้ในการทดสอบจะเป็นข้อมูลทั้งสองชุดรวมกัน ผลการทดลองแสดงไว้ตามตารางที่ 6.8 ดังนี้

ตารางที่ 6.8 แสดงผลความแตกต่างของลักษณะการคัดลายมือ

ลักษณะของการคัดลายมือ	จำนวนอักษรที่ใช้ทดสอบ	ความถูกต้อง (%)
คัดแบบบรรจง	2,640	85.40
คัดแบบไม่บรรจง	2,640	68.02

จากผลการทดลองตามตารางที่ 6.8 แสดงให้เห็นว่าข้อมูลที่ใช้ในการเตรียม และฝึกสอนควรจะเป็นข้อมูลที่ได้จากการคัดแบบบรรจง จะทำให้ได้ผลการทดลองถูกต้องมากขึ้น

บทที่ 7

สรุปผลการทดลอง

7.1 ระบบการรู้จำตัวอักษรคัดลายมือเขียนภาษาไทยโดยประยุกต์ใช้การทรานสฟอร์มแบบคาร์ฮูเน็นเลิฟเข้ากับโครงข่ายฟัซซี่นิวรอล

ในระบบนี้เราได้ใช้การแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ เพื่อทำการหาแปลงข้อมูลของตัวอักษรแต่ละพยัญชนะที่มีรูปแบบต่างๆ กัน จากผู้คัดลายมือ 10 คน คนละ 3 ครั้ง การใช้การแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ ก็คือเป็นการลดจำนวนข้อมูลบิตแม็บของตัวพยัญชนะแต่ละตัว โดยเอาแต่ข้อมูลที่แสดงลักษณะร่วมของข้อมูล นอกจากนั้นเรายังได้นำหลักการของฟัซซี่เซต เข้ามาประยุกต์ใช้ โดยข้อมูลที่ได้จากขั้นตอนการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟ เราจะทำการปรับข้อมูบให้อยู่ในระบบของฟัซซี่เซตเพื่อเป็นการลดจำนวนฟีเจอร์(Feature)ของข้อมูล นำข้อมูลที่ได้เข้าไปฝึกสอนในโครงข่ายนิวรอล

ข้อมูลที่ได้หลังการแปลงข้อมูลแบบคาฮูเน็นเลิฟ และปรับให้อยู่ในระบบของฟัซซี่เซต ก็คือ ข้อมูลที่ใช้เป็นตัวแทนพยัญชนะแต่ละตัว เข้าไปฝึกสอนโดยที่เราไม่จำเป็นต้องทำการฝึกสอนตัวอักษรทุกตัว ทำให้งานวิจัยนี้มีจุดเด่นตรงที่ ทำให้ขั้นตอนการฝึกสอนใช้เวลาน้อยลง

จากการทดลองโดยใช้โครงข่ายนิวรอล ซึ่งจะมีทั้งหมด 3 เลเยอร์ คือ อินพุตเลเยอร์ ฮิดเดนเลเยอร์ และ เอาท์พุทเลเยอร์ กำหนดจำนวนโหนดดั้งนี้ 512 100 และ 44 ตามลำดับ ใช้เวลาในการฝึกสอนประมาณ 6 ชั่วโมง

จากการทดสอบความถูกต้องของการรู้จำของตัวอักษรทั้งหมด 2,640 ความถูกต้องในการรู้จำเฉลี่ย เท่ากันร้อยละ 90.53

7.2 ข้อสรุปและเปรียบเทียบของการรู้จำ

ในการศึกษาทฤษฎีการรู้จำตัวอักษรนั้นสามารถรู้จำได้หลายวิธีเช่น การรู้จำอักษรตัวพิมพ์ภาษาไทยโดยวิธีโครงข่ายนิวรอลแบบแบคพรอบพาเกชั่น โดยการนำผลการวิจัยจากวิยานิพนธ์ของนายทวี เปรมรัตนชัย มาทำการเปรียบเทียบกับระบบการรู้จำในงานวิจัยฉบับนี้

ตาราง 7.1 แสดงผลเปรียบเทียบการรู้จำของงานวิจัยนี้กับระบบการรู้จำของนายทวี เปรมรัตนชัย

หัวข้อการเปรียบเทียบ	ระบบงานวิจัยนี้	ระบบงานวิจัยที่มาเปรียบเทียบ
1.หลักการทั่วไป	นำข้อมูลบิตแม็บของตัวอักษรแต่ละพยัญชนะ ที่มีรูปแบบต่างกันหามาตัวแทน เข้าสู่ขั้นตอนการเรียนรู้	นำข้อมูลบิตแม็บของตัวอักษรแต่ละพยัญชนะ ที่มีรูปแบบต่างกัน ทุกตัวเข้าสู่ขั้นตอนการเรียนรู้
2.ข้อมูลที่ใช้ในการทดลอง	ตัวอักษรคัดลายมือ	อักษรตัวพิมพ์
3.โครงสร้างของโครงข่ายนิวรอล	มี 3 เลเยอร์ อินพุตเลเยอร์ ฮิดเดนเลเยอร์ และ เอาท์พุทเลเยอร์ กำหนดจำนวนโหนดดั้งนี้ 512 100 และ 44 ตามลำดับ	มี 4 เลเยอร์ อินพุตเลเยอร์ ฮิดเดนเลเยอร์ ฮิดเดนเลเยอร์ และ เอาท์พุทเลเยอร์ กำหนดจำนวนโหนดดั้งนี้ 1,600 230 230 และ 87 ตามลำดับ
4.เครื่องมือ	PC รุ่น COMPAQ PRESARIO 4830	มินิคอมพิวเตอร์H.P. 9000/827
5.เวลาที่ใช้ในการฝึกสอน	6 ชั่วโมง	13 วัน
6.จำนวนตัวอักษรที่ใช้ในการฝึกสอน	1,320 ตัว	87 ตัว
7. ความถูกต้องในการรู้จำเฉลี่ย	ร้อยละ 90	ร้อยละ 98
8. ลักษณะของตัวอักษร(font)ที่ใช้ฝึกสอน	30 แบบลายมือเขียน	แบบอังสานายูพีซี 12 dpi

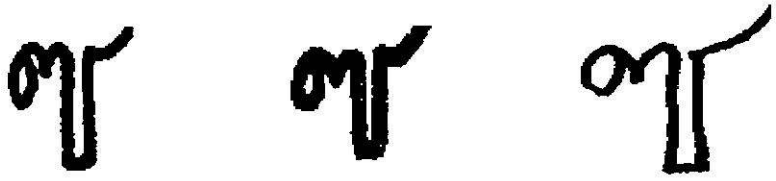
7.3 ข้อจำกัดของระบบ

ในการวิจัยนี้ได้ทำการทดลองในหลายลักษณะทำให้เราทราบข้อจำกัดของระบบในงานวิจัยดังนี้

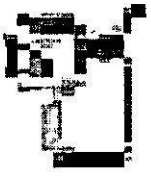
7.3.1 จากการทดลองเปรียบเทียบข้อมูลตัวอักษรที่ใช้ในการเตรียมข้อมูลและฝึกสอนระหว่างผู้คัดที่คัดลายมือแบบบรรจง กับแบบที่ไม่บรรจงทำให้ทราบว่าในระบบงานวิจัยนี้สามารถใช้ได้ดีกับตัวอักษรจากผู้ที่คัดลายมือแบบบรรจง

7.3.2 เนื่องจากตัวอักษรที่ได้จากการคัดลงบนกระดาษ A4 ผ่านเครื่องอ่านภาพ และเก็บข้อมูลอยู่ในรูปของ Bitmap หรือ .bmp ทำให้ขนาดของข้อมูลแต่ละตัวอักษรมีขนาดใหญ่ เช่น 1024 x 512 ในระบบของเราต้องทำการลดขนาดของตัวอักษรทุกตัวให้มีขนาด 32 x 16 บางครั้งทำให้ข้อมูลบางส่วนหายไปเป็นผลทำให้การวิเคราะห์ผิดพลาด ดังนั้นในระบบนี้ผู้ที่คัดลายมือจะต้องคัดตัวอักษรที่มีขนาดจำกัด

7.3.3 การผิดพลาดจากการหาลักษณะร่วมของข้อมูลโดยใช้วิธีการแบบคาร์ฮูเน็นเลิฟ จากผลการทดลองตามตารางที่ 6.2 และ 6.3 จะเห็นได้ว่ามีพยัญชนะบางตัวที่สามารถวิเคราะห์ตัวอักษรได้ถูกต้อง 100 เปอร์เซ็นต์ แต่ก็มีพยัญชนะบางตัวเกิดความผิดพลาดมากจากการวิเคราะห์โดยดูจากข้อมูลหลังจากทำการแปลงข้อมูลแบบคาร์ฮูเน็นเลิฟที่ใช้ในการฝึกสอนกับข้อมูลที่ใช้ในการทดสอบมีความแต่ต่างกันมาก แสดงให้เห็นว่าตัวอักษรที่ให้ผลความผิดพลาดมาก เกิดจากตัวอักษรตัวนั้นได้จากผู้คัดลายมือที่ไม่ดีพอ หรือลักษณะของพยัญชนะตัวนั้นมีโครงสร้างซับซ้อน ทำให้ข้อมูลที่แสดงลักษณะร่วมของข้อมูลพยัญชนะตัวนั้น ไม่สามารถเป็นตัวแทนที่ดีพอที่จะใช้ในขั้นตอนการฝึกสอน ภาพที่ 7.1 แสดงข้อมูลพยัญชนะ ซ ซึ่งมีความผิดพลาดของการรู้จำสูง เห็นได้ว่าข้อมูลที่เป็นตัวแทนสำหรับฝึกสอน กับข้อมูลที่ใช้ทดสอบมีความแตกต่างกันมาก



ตัวอย่างอักษรที่ใช้ทดสอบ



ตัวแทนที่ใช้ในการฝึกสอน

ภาพที่ 7.1 แสดงข้อมูลที่ใช้ในการทดสอบ และตัวแทนที่ใช้ในการฝึกสอนของพยัญชนะ ซ

7.4 ข้อเสนอแนะสำหรับแนวทางการพัฒนา

เนื่องจากในงานวิจัยฉบับนี้มีการทำงานที่ไม่ครอบคลุมการทำงานบางส่วนจึงได้มีข้อเสนอแนะเพื่อเป็นแล้วทางในการพัฒนาระบบนี้ให้ดีขึ้นต่อไป

1. เนื่องจากเราต้องกำหนดขนาดของตัวอักษรของผู้คัดลายมือเพราะ เนื่องจากระบบของเราต้องทำการปรับขนาดข้อมูลให้มีขนาด 32 x 16 ถ้าตัวอักษรที่มีขนาดใหญ่เกินไปจะทำให้มีการสูญเสียของข้อมูลเกิดขึ้น ดังนั้นหลังจากทำการปรับขนาดของข้อมูลแล้วควรจะทำการปรับปรุงข้อมูล (Enhancement) ให้ข้อมูลมีความสมบูรณ์เพิ่มขึ้น

2. ตัวพยัญชนะที่มีรูปร่างใกล้เคียงกันน่าจะนำมาจัดให้เป็นกลุ่มเดียวกันก่อนและทำการแยกตัวอักษรภายในกลุ่มนั้นอีกครั้งหนึ่ง คาดว่าจะทำให้เวลาในการฝึกสอนน้อยลงและได้ผลการทดลองดีขึ้น

3. ในการเลือกอักษรที่ใช้ในการเตรียมข้อมูลและฝึกสอนควรจะเลือกตัวอักษรที่เกิดจากการคัดลายมือแบบบรรจง มีความชัดเจนของข้อมูล และมีสิ่งรบกวนจากการอ่านจากเครื่องอ่านภาพให้น้อยที่สุด ซึ่งจะส่งผลให้การรู้จำมีประสิทธิภาพมากยิ่งขึ้น

4. ในระบบฟัซซี่ที่ทำในงานวิจัยนี้เรากำหนดให้มีจำนวนค่าความเป็นสมาชิกทั้งหมด 6 ระดับดังนี้ 0, 0.2, 0.4, 0.6, 0.8 และ 1 อาจจะเพิ่มจำนวนค่าความเป็นสมาชิกให้มากขึ้นเพราะจำนวนค่าความเป็นสมาชิกน้อยอาจจะทำให้ข้อมูลที่ใช้ในการฝึกสอนไม่เพียงพอต่อความต้องการ

7.5 ข้อวิเคราะห์เปรียบเทียบ

ผลการทดลองในตาราง 6.3 เป็นการทดสอบตัวอักษรใหม่กับการฝึกสอนโดยผ่านระบบคือนำตัวอักษรที่ต้องการฝึกสอนมาทำการหาลักษณะตัวแทนของตัวอักษรโดยใช้การ ทรานส์ฟอร์มแบบคาร์ฮูเน็นเลิฟ แล้วนำค่าฟีเจอร์ของตัวแทนตัวอักษรมาทำการลดค่าฟีเจอร์โดยปรับเข้าระบบฟัซซี่มีค่าความเป็นสมาชิก 6 ระดับจากนั้นนำค่าฟีเจอร์ใหม่ไปทำการฝึกสอนในระบบนิวรอลเน็ตเว็คและเมื่อนำผลมาเปรียบเทียบการฝึกสอนโดยไม่ต้องทำการปรับค่าฟีเจอร์ในฟัซซี่นั้น พบว่าการนำระบบฟัซซี่มาทำการปรับค่านั้นจะช่วยลดเวลาในการฝึกสอนแล้วยังเพิ่มความถูกต้องอีกประมาณ 5 เปอร์เซนต์ตามผลการทดลองในตาราง 6.6 และเมื่อเราเปลี่ยนวิธีการทรานส์ฟอร์มแบบคาร์ฮูเน็นเลิฟมาใช้วิธีการหาค่าเฉลี่ยตัวอักษรแทนพบว่าผลการทดสอบการเรียนรู้ไม่แตกต่างกัน เมื่อไปดูค่าฟีเจอร์ของข้อมูลตามผนวก ก การทรานส์แบบคาร์ฮูเน็นเลิฟจะมีรายละเอียดของค่าฟีเจอร์มากกว่าการนำตัวอักษรไปเฉลี่ย ซึ่งทำให้มีความเห็นว่าการทรานส์ฟอร์มแบบคาร์ฮูเน็นเลิฟจะเหมาะกับการใช้ลักษณะข้อมูลที่ต้องการความละเอียดของข้อมูลหลังจากทำการทรานส์ฟอร์ม เช่น การทรานส์ฟอร์มเสียงต่าง ๆ

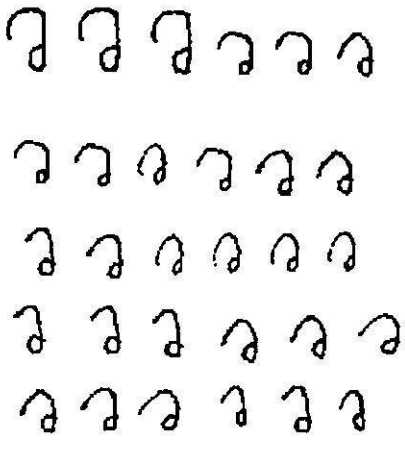
เอกสารอ้างอิง

1. Kimpan C., Ttoh A., and kawanishi K., "Recognition of Printed Thai Character using a matching method", IEE Proc., Vol. 130, PtE., No.6, Nov. 1983.
2. Airphaiboon S., and Kondo S., "Recognition of handprinted Thai Characters. Using Loop structures." IEICE Trans. INF.8 SYST., Vol.E79-D,NO.9, Sep.1996
3. Airpahiboon S., Sanyworasil M., Attachoo B., and Kimpan C., "Recognition of handwritten Thai Character considering the head of character." IEEE International Conference on Image Processig. Vol.2, Singapore, 1989.
4. Phokharatkul P. and Kimpan C., "Recoginition of handprinted Thai Character Using the cavity features of Character based on neural Network.", IEEE International Conference on 1998.
5. Robert J. Schalkoff, " Pattern Recognition," New youk : John Wiley &Son Inc., 1992
6. Kimpan C., Itoh A., And Kawashi K.," Fine Classification of printed Thai character recognition using the karhunen loeve expantion ," Proc.IEEE, Vol.134,Pt.E., No.5. Sept. 1987.
7. พิศิษฎ์ โภคารัตน์กุล และคณะ.2542. " การรู้จำอักษรตัวพิมพ์ภาษาไทยโดยวิธีวัดความเหมือนแบบฟัซซี่" หน้า 349-354 ใน การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 20. กรุงเทพฯ : คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย.
8. ทวี เปรมรัตนชัย. 2539. "การรู้จำตัวอักษรตัวพิมพ์ภาษาไทยโดยวิธีโครงข่ายนิวรอนแบบแบคพรอบพาเคชั่น." วิทยานิพนธ์วิทยาศาสตร์มหาบัณฑิตสาขาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ บัณฑิตวิทยาลัย, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

ภาคผนวก

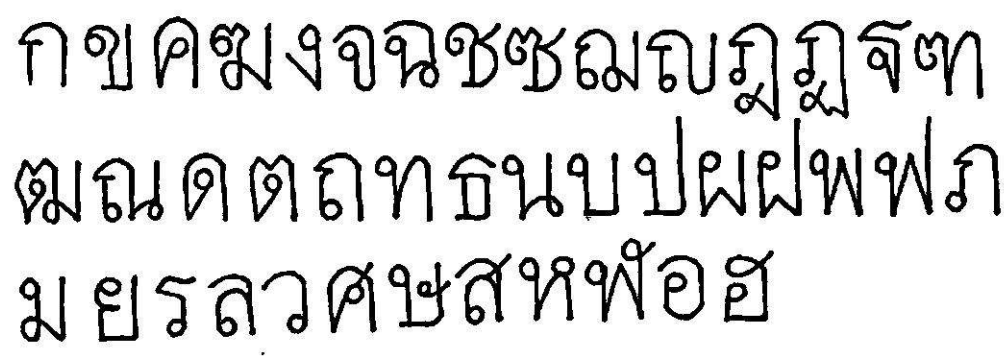
ภาคผนวก ก
ตัวอย่างลายมือคัดที่ใช้ในการทดลอง

1. ตัวอย่างข้อมูลที่ใช้ในการเตรียมข้อมูล และฝึกสอน โดยคัดแบบบรรจง

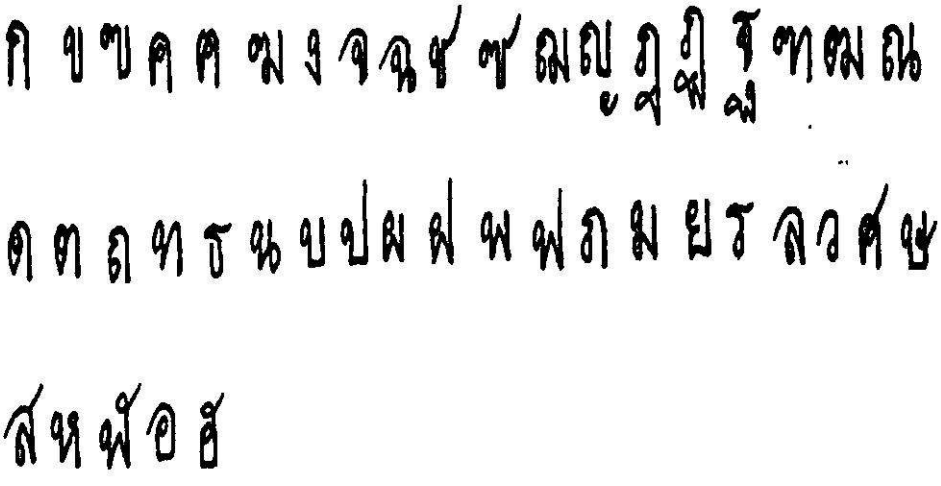


2. ตัวอย่างข้อมูลที่ใช้ในการเตรียมข้อมูล และฝึกสอน โดยคัดแบบไม่บรรจง

3. ตัวอย่างข้อมูลที่ใช้สำหรับการทดสอบ โดยผู้คัดชุดเดียวกับที่ใช้ในการเตรียมข้อมูล



4. ตัวอย่างข้อมูลที่ใช้สำหรับการทดสอบ โดยผู้คัดคนละชุดกับที่ใช้ในการเตรียมข้อมูล



5. ตัวอย่างของลายละเอียด(feature) ของตัวอักษรที่ผ่านการทรานส์ฟอร์มแบบคาร์ฮูเน็นเลิฟ

Furture of KL NN

ans =

Columns 1 through 7	Columns 8 through 14	Columns 15 through 16
0 0 0 0 0 0 0.3918	1.0000 1.0000 1.0000 0.3947 0 0 0	0 0
0 0 0 0 0.1902 1.0000 1.0000	1.0000 1.0000 1.0000 1.0000 0.8098 0 0	0 0
0 0 0 0.1902 1.0000 1.0000 1.0000	1.0000 1.0000 1.0000 1.0000 1.0000 0.7983 0	0 0
0 0 0.3946 1.0000 1.0000 1.0000 1.0000	1.0000 0.5849 0.1902 0.3918 1.0000 1.0000 0.6082	0 0
0 0.1902 0.7983 1.0000 1.0000 1.0000 0.8098	0 0 0 0 0 1.0000 1.0000	0.2134 0
0 0.7983 1.0000 1.0000 1.0000 0.2017 0	0 0 0 0 0 0.5962 1.0000	0.7983 0
0.3946 1.0000 1.0000 1.0000 0.6054 0 0	0 0 0 0 0 0 1.0000	1.0000 0.2134
0.7983 1.0000 1.0000 1.0000 0.3946 0.1902 0	0 0 0 0 0 0 1.0000	1.0000 0.4178
0.4038 1.0000 1.0000 0.8097 0.6080 0.6080 0.1902	0 0 0 0 0 0 1.0000	1.0000 0.2134
0.4151 1.0000 1.0000 1.0000 0.6080 0.6080 0.4036	0 0 0 0 0 0 1.0000	1.0000 0.2134
0.2017 0.2017 0.7956 1.0000 0.6080 0.6080 0.2134	0 0 0 0 0 0 1.0000	1.0000 0.2134
0.2017 0.2017 0.7956 1.0000 1.0000 0.3918 0	0 0 0 0 0 0 1.0000	1.0000 0.6054
0 0.2017 0.5822 1.0000 1.0000 0.2017 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 0.7956 1.0000 0.6195 0.2017 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 0.7956 1.0000 0.6195 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0.2017 0.7956 1.0000 0.4178 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0.2017 1.0000 1.0000 0.4178 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0.2017 1.0000 1.0000 0.4178 0 0	0 0 0 0 0 0 0.6195	1.0000 1.0000
0 0.2017 1.0000 1.0000 0.4178 0 0	0 0 0 0 0 0 0.6195	1.0000 1.0000
0 0.2017 1.0000 1.0000 0.4178 0 0	0 0 0 0 0 0 0.6195	1.0000 1.0000
0 0.2017 1.0000 1.0000 0.4178 0 0	0 0 0 0 0 0 0.6195	1.0000 1.0000
0 0.2017 1.0000 1.0000 0.4178 0 0	0 0 0 0 0 0 0.6195	1.0000 1.0000
0 0.2017 1.0000 1.0000 0.4178 0 0	0 0 0 0 0 0 0.6195	1.0000 1.0000
0 0.2017 1.0000 1.0000 0.4178 0 0	0 0 0 0 0 0 0.8098	1.0000 1.0000
0 0.2017 1.0000 1.0000 0.4178 0 0	0 0 0 0 0 0 0.8098	1.0000 1.0000
0 0.2017 1.0000 1.0000 0.4178 0 0	0 0 0 0 0 0 0.8098	1.0000 1.0000
0 0.2017 1.0000 1.0000 0.4178 0 0	0 0 0 0 0 0 0.8098	1.0000 1.0000
0 0.2017 1.0000 1.0000 0.2134 0 0	0 0 0 0 0 0 0.8098	1.0000 1.0000
0 0.2017 1.0000 1.0000 0.2134 0 0	0 0 0 0 0 0 0.8098	1.0000 1.0000
0 0.2017 0.4151 0.7956 0.2134 0 0	0 0 0 0 0 0 0.8098	1.0000 1.0000
0 0 0.2017 0.4151 0.2134 0 0	0 0 0 0 0 0 0	1.0000 0.8097

6. ตัวอย่างของลายละเอียด(feature) ของตัวอักษรที่ผ่านการทรานส์ฟอร์มแบบคาร์ฮูเน็นเลฟ และผ่านการปรับค่าโดยระบบฟัซซี

fariure KL FUZZY NN

Columns 1 through 7	Columns 8 through 14	Columns 15 through 16
0 0 0 0 0 0 0.4000	1.0000 1.0000 1.0000 0.4000 0 0 0	0 0
0 0 0 0 0 1.0000 1.0000	1.0000 1.0000 1.0000 1.0000 1.0000 0 0	0 0
0 0 0 0 1.0000 1.0000 1.0000	1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 0	0 0
0 0 0.4000 1.0000 1.0000 1.0000 1.0000	1.0000 0.8000 0 0.4000 1.0000 1.0000 0.6000	0 0
0 0 1.0000 1.0000 1.0000 1.0000 1.0000	0 0 0 0 0 1.0000 1.0000	0 0
0 1.0000 1.0000 1.0000 1.0000 0 0	0 0 0 0 0 0.8000 1.0000	1.0000 0
0.4000 1.0000 1.0000 1.0000 0.6000 0 0	0 0 0 0 0 0 1.0000	1.0000 0
1.0000 1.0000 1.0000 1.0000 0.4000 0 0	0 0 0 0 0 0 1.0000	1.0000 0.2000
0.2000 1.0000 1.0000 1.0000 0.6000 0.6000 0	0 0 0 0 0 0 1.0000	1.0000 0
0.2000 1.0000 1.0000 1.0000 0.6000 0.6000 0.2000	0 0 0 0 0 0 1.0000	1.0000 0
0 0 1.0000 1.0000 0.6000 0.6000 0	0 0 0 0 0 0 1.0000	1.0000 0
0 0 1.0000 1.0000 1.0000 0.4000 0	0 0 0 0 0 0 1.0000	1.0000 0.6000
0 0 0.8000 1.0000 1.0000 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 1.0000 1.0000 0.6000 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 1.0000 1.0000 0.6000 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 1.0000 1.0000 0.2000 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 1.0000 1.0000 0.2000 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 1.0000 1.0000 0.2000 0 0	0 0 0 0 0 0 0.6000	1.0000 1.0000
0 0 1.0000 1.0000 0.2000 0 0	0 0 0 0 0 0 0.6000	1.0000 1.0000
0 0 1.0000 1.0000 0.2000 0 0	0 0 0 0 0 0 0.6000	1.0000 1.0000
0 0 1.0000 1.0000 0.2000 0 0	0 0 0 0 0 0 0.6000	1.0000 1.0000
0 0 1.0000 1.0000 0.2000 0 0	0 0 0 0 0 0 0.6000	1.0000 1.0000
0 0 1.0000 1.0000 0.2000 0 0	0 0 0 0 0 0 0.6000	1.0000 1.0000
0 0 1.0000 1.0000 0.2000 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 1.0000 1.0000 0.2000 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 1.0000 1.0000 0.2000 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 1.0000 1.0000 0.2000 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 1.0000 1.0000 0 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 1.0000 1.0000 0 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 0.2000 1.0000 0 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 0 0.2000 0 0 0	0 0 0 0 0 0 0	1.0000 1.0000

7. ตัวอย่างของลายละเอียด(feature) ของตัวอักษรที่ทำการหาค่าเฉลี่ย

Feature MAEN NN

Columns 1 through 7	Columns 8 through 14	Columns 15 through 16
0 0 0 0 0 0 0.4000	1.0000 1.0000 1.0000 0.4000 0 0 0	0 0
0 0 0 0 0.2000 1.0000 1.0000	1.0000 1.0000 1.0000 1.0000 0.8000 0 0	0 0
0 0 0 0.2000 1.0000 1.0000 1.0000	1.0000 1.0000 1.0000 1.0000 1.0000 0.8000 0	0 0
0 0 0.4000 1.0000 1.0000 1.0000 1.0000	1.0000 0.6000 0.2000 0.4000 1.0000 1.0000 0.6000	0 0
0 0.2000 0.8000 1.0000 1.0000 1.0000 0.8000	0 0 0 0 0 1.0000 1.0000	0.2000 0
0 0.8000 1.0000 1.0000 1.0000 0.2000 0	0 0 0 0 0 0.6000 1.0000	0.8000 0
0.4000 1.0000 1.0000 1.0000 0.6000 0 0	0 0 0 0 0 0 1.0000	1.0000 0.2000
0.8000 1.0000 1.0000 1.0000 0.4000 0.2000 0	0 0 0 0 0 0 1.0000	1.0000 0.4000
0.4000 1.0000 1.0000 0.8000 0.6000 0.6000 0.2000	0 0 0 0 0 0 1.0000	1.0000 0.2000
0.4000 1.0000 1.0000 1.0000 0.6000 0.6000 0.4000	0 0 0 0 0 0 1.0000	1.0000 0.2000
0.2000 0.2000 0.8000 1.0000 0.6000 0.6000 0.2000	0 0 0 0 0 0 1.0000	1.0000 0.2000
0.2000 0.2000 0.8000 1.0000 1.0000 0.4000 0	0 0 0 0 0 0 1.0000	1.0000 0.6000
0 0.2000 0.6000 1.0000 1.0000 0.2000 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 0.8000 1.0000 0.6000 0.2000 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0 0.8000 1.0000 0.6000 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0.2000 0.8000 1.0000 0.4000 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0.2000 1.0000 1.0000 0.4000 0 0	0 0 0 0 0 0 1.0000	1.0000 1.0000
0 0.2000 1.0000 1.0000 0.4000 0 0	0 0 0 0 0 0 0.6000	1.0000 1.0000
0 0.2000 1.0000 1.0000 0.4000 0 0	0 0 0 0 0 0 0.6000	1.0000 1.0000
0 0.2000 1.0000 1.0000 0.4000 0 0	0 0 0 0 0 0 0.6000	1.0000 1.0000
0 0.2000 1.0000 1.0000 0.4000 0 0	0 0 0 0 0 0 0.6000	1.0000 1.0000
0 0.2000 1.0000 1.0000 0.4000 0 0	0 0 0 0 0 0 0.6000	1.0000 1.0000
0 0.2000 1.0000 1.0000 0.4000 0 0	0 0 0 0 0 0 0.6000	1.0000 1.0000
0 0.2000 1.0000 1.0000 0.4000 0 0	0 0 0 0 0 0 0.8000	1.0000 1.0000
0 0.2000 1.0000 1.0000 0.4000 0 0	0 0 0 0 0 0 0.8000	1.0000 1.0000
0 0.2000 1.0000 1.0000 0.4000 0 0	0 0 0 0 0 0 0.8000	1.0000 1.0000
0 0.2000 1.0000 1.0000 0.4000 0 0	0 0 0 0 0 0 0.8000	1.0000 1.0000
0 0.2000 1.0000 1.0000 0.2000 0 0	0 0 0 0 0 0 0.8000	1.0000 1.0000
0 0.2000 1.0000 1.0000 0.2000 0 0	0 0 0 0 0 0 0.8000	1.0000 1.0000
0 0.2000 0.4000 0.8000 0.2000 0 0	0 0 0 0 0 0 0.8000	1.0000 1.0000
0 0 0.2000 0.4000 0.2000 0 0	0 0 0 0 0 0 0	1.0000 0.8000

ภาคผนวก ข
โปรแกรมที่พัฒนาขึ้นในการวิจัย

1. โปรแกรมสำหรับการเตรียมข้อมูลก่อนเข้าโปรแกรมเรียนรู้

Preprocessing before Training

```
clear
echo on
clc
%  This is Program main preprocessing It is to do a step
%  Read and Show Data of file.bmp
%  ==========================================
echo off
CH={'ก','ข','ค','ฆ','ง','จ','ฉ','ช','ซ','ฌ','ญ','ฎ','ฏ','ฐ','ฑ','ฒ','ณ','ด','ต','ถ','ท','ธ','น','บ','ป','ผ','ฝ','พ','ฟ','ภ','ม','ย','ร','
ล','ว','ศ','ษ','ส','ห','ฬ','อ','ฮ'}
nfile=input('number of all Charector for clasifind is  ')
for nc=1:nfile
  T=strcat(CH(nc),'.bmp');
  T=char(T);
  pre ;
  in_vec(:,nc)=St1(:,:);
end;
```

Pre use for This is Program preprocessing It is to do a step Read and Show Data of file.bmp

```
echo on
clc
%  This is Program preprocessing It is to do a step
%  Read and Show Data of file.bmp
%  ==========================================
echo off
```

```
data=imread(T);
data=pattern(data,0);
figure;
imshow(data);
echo on
clc
% Strike any key to continue...
echo off
%pause
close
echo on
clc
% Segment and Normalization 32x16
% Output is Input for KLT name InKlt
echo off
mvec=1;
[Ximage Yimage]=size(data);
Check_S = 0;
count  = 0;
for X=1:Ximage;
   for Y=1:Yimage;
    if (data(X,Y)==0);
       [MinX MinY MaxX MaxY]=segment(data,X,Y);
    if ((MaxX-MinX)>=2)&((MaxY-MinY)>=2) ;
        Char=data(MinX:MaxX,MinY:MaxY);
        figure(1);imshow(Char);
        Char=pattern(Char,1);
        out=slrul(Char);
        [PC,CE]=Klt3(out,4);
        A=PC(:,1);
        m=std(A);
        PC1=convlogic(A,m);
        PC1=pattern(PC1,1);
```

```
        Stp1=PC1;
        PC1=reshape(PC1,16,32);
        PC1=PC1';
        figure(2)
        imshow(PC1)
        InKlt1(:,mvec)=Stp1;
        mvec=mvec+1
      end;
        data(MinX:MaxX,MinY:MaxY)=1;
      end;
    end;
 end;
close all
echo on
clc
% Find concentation of this Charector using KLT
% .....And Show.....
% Strike any key to continue...
echo off
%pause

[PC,CE]=Klt3(InKlt1,1);
A=PC(:,1);
[PC1]=MemberFuzzy(A);
St1=PC1;
PC1=reshape(PC1,16,32);
PC1=PC1';figure(5);
imshow(PC1)
```

Function Segmentation Return MinX,MaxX MinY MaxY is Position of Picture

Parameter data is data that you need to sent.

X,Y is start point

```
function [MinX,MinY,MaxX,MaxY]=segment(data,X,Y)
ix=X;   iy=Y;
jx=ix-1; jy=iy;
d=1;
bx=jx-ix; by=jy-iy;
round=1;
jx=ix;  jy=iy;
MinX=X; MaxX=X;
MinY=Y; MaxY=Y;
while d<=8
  dx=bx+by; dy=by-bx;
  if(dx~=0)  dx=dx/abs(dx);  end;
  if(dy~=0)  dy=dy/abs(dy);  end;
  ix=jx+dx; iy=jy+dy;
  if (data(ix,iy)==0)
    if (MinX>ix) MinX=ix;   end;
    if (MinY>iy) MinY=iy;   end;
    if (MaxX<ix) MaxX=ix;   end;
    if (MaxY<iy) MaxY=iy;   end;
    if (ix==X)&(iy==Y)&(round==2)
      break;
    else
      if (ix==X)&(iy==Y)&(round==1) round=2; end;
      bx=jx-ix; by=jy-iy;
      jx=ix;   jy=iy;
      d=1;
    end;
```

```
  else
    bx=dx; by=dy; d=d+1;
  end;
end;
```

Function Pattern use of convert Picture

```
function DataIn=pattern(DataInPut,Di)
% Di is '0' or '1' that you want is charactor
[SizeTR,SizeTC]= size(DataInPut);
  if (DataInPut(1,1)==Di)    |(DataInPut(1,SizeTC)==Di)...
      (DataInPut(SizeTR,1)==Di)|(DataInPut(SizeTR,SizeTC)==Di)
          for STC=1:SizeTC
        for STR=1:SizeTR
          if DataInPut(STR,STC)==Di
            DataIn(STR,STC)=1-Di;
          else
            DataIn(STR,STC)=Di;
          end;
        end;
      end;
  else
      DataIn=DataInPut;
  end;
```

Function Slrul use Shife Left, Right , Top and Botton

```
function out=slrul(inp)

sh=shife(inp,0,1);
sh=imresize(sh,[32 16],'nearest');
s=reshape(sh',1,512);
```

```
sh=shife(inp,1,1);
sh=imresize(sh,[32 16],'nearest');
sl1=reshape(sh',1,512);

sh=shife(inp,1,2);
sh=imresize(sh,[32 16],'nearest');
sl2=reshape(sh',1,512);

sh=shife(inp,2,1);
sh=imresize(sh,[32 16],'nearest');
sr1=reshape(sh',1,512);

sh=shife(inp,2,2);
sh=imresize(sh,[32 16],'nearest');
sr2=reshape(sh',1,512);

sh=shife(inp,3,1);
sh=imresize(sh,[32 16],'nearest');
su1=reshape(sh',1,512);

sh=shife(inp,3,2);
sh=imresize(sh,[32 16],'nearest');
su2=reshape(sh',1,512);

sh=shife(inp,4,1);
sh=imresize(sh,[32 16],'nearest');
sd1=reshape(sh',1,512);

sh=shife(inp,4,1);
sh=imresize(sh,[32 16],'nearest');
sd2=reshape(sh',1,512);
```

```
out=[s',sl1',sl2',sr1',sr2',su1',su2',sd1',sd2'];
```

Function Shife use for make enhancement Character

```
function sh=shife(inp,p,nu)
[m,n]=size(inp);
in=zeros(m,n);
in(:,:)=inp(:,:);
sh=zeros(m,n);

if p==0
  sh=in;
end

if p==1
  a=zeros(m,n+nu);
  for i=1:n
    a(:,i)=in(:,i);
  end
  for i=1:n
      sh(:,i)=a(:,i+nu);
  end
end

if p==2
  a=zeros(m,n+nu);
  for i=n:-1:1
    a(:,i+nu)=in(:,i);
  end
  for i=1:n
    sh(:,i)=a(:,i);
  end
end
```

```
if p==3
  a=zeros(m+nu,n);
  for i=1:m
    a(i,:)=in(i,:);
  end
  for i=1:m
    sh(i,:)=a(i+nu,:);
  end
end

if p==4
  a=zeros(m+nu,n);
  for i=m:-1:1
    a(i+nu,:)=in(i,:);
  end
  for i=1:m
    sh(i,:)=a(i,:);
  end
end
```

Function KL uses Transform by K-L Transform

```
function [PC,CE]=klt3(X,P)
%%% Karhunen-Lovev Tranfomation %%%
%    X is input
%    X must be a NxL matrix of N vectors in L dimensions.
%    P is number of principle component
% chk input dimention N x L (L must more than N)
```

```
[N,L]=size(X);
A=zeros(N,L);
A(:,:)=X(:,:);
X=A;
if L > N;
  X=X';
end;

% covariance matrix
[m,n] = size(X) ;% n = number of row ,m = number of collume

c = X' * X ;     % covariance matrix

%    eigenvalue&eigenvector of covriance matrix
[Vi, Si] = eig(c) ;
[l, idx] = sort(diag(Si)') ;
l = fliplr(l) ;
idx = fliplr(idx) ;
Vi=Vi(:,idx);%PC= V(:,1:P);
for i=1:n;
  V(:,i)=(1./sqrt(l(i))).*(X*Vi(:,i));
end;


%PC is principal components (P is number of principal components)
PC = V(:,1:P) ;

%    PC is principal components (P is number of principal components)
%for i=1:P
%  PC(:,i)=(1./sqrt(l(i))).*(X*PCC(:,i));
%end
```

```
%CE is coefficient of principal component
CE=PC'*X;
```

Function ConvertLogic use Convert Picture

```
function DaLogic=convlogic(DataInPut,Ind)
[SizeTR,SizeTC]= size(DataInPut);
    for STC=1:SizeTC
      for STR=1:SizeTR
        if DataInPut(STR,STC)<=Ind
          DaLogic(STR,STC)=0;
        else
          DaLogic(STR,STC)=1;
        end;
      end;
    end;
```

Function DMF use for fine Member Shift Function of Fuzzy Logic

```
function mf=dmf(IP)
% dmf is function for decide for Fuzzy memberzip
vmf(1,1) = evalmf(IP,[-1    0.25  0.3],'trimf');
vmf(2,1) = evalmf(IP,[0.275 0.35  0.425],'trimf');
vmf(3,1) = evalmf(IP,[0.375 0.45  0.525],'trimf');
vmf(4,1) = evalmf(IP,[0.475 0.55  0.625],'trimf');
vmf(5,1) = evalmf(IP,[0.575 0.65  0.725],'trimf');
vmf(6,1) = evalmf(IP,[0.7   0.75  2],'trimf');

I=find(vmf);
```

```
[ir,ic] = size(I);


switch ir
case 1
  switch I(1,1)
  case 1
    mf = 0;   break;
  case 2
    mf = 0.2; break;
  case 3
    mf = 0.4; break;
  case 4
    mf = 0.6; break;
  case 5
    mf = 0.8; break;
  case 6
    mf = 1;   break;
  otherwise
    mf = 2;   break;
  end;
  break;
case 2
  if (I(1,1)==1)&(I(2,1)==2)
    if (vmf(1,1)<=vmf(2,1))
      mf = 0;
    else
      mf = 0.2;
    end;
  elseif  (I(1,1)==2)&(I(2,1)==3)
    if (vmf(2,1)<=vmf(3,1))
      mf = 0.2;
    else
```

```
      mf = 0.4;
    end;
  elseif (I(1,1)==3)&(I(2,1)==4)
    if (vmf(3,1)<=vmf(4,1))
      mf = 0.4;
    else
      mf = 0.6;
    end;
  elseif (I(1,1)==4)&(I(2,1)==5)
    if (vmf(4,1)<=vmf(5,1))
      mf = 0.6;
    else
      mf = 0.8;
    end;
  elseif (I(1,1)==5)&(I(2,1)==6)
    if (vmf(5,1)<=vmf(6,1))
      mf = 0.8;
    else
      mf = 1;
    end;
  end;
otherwise
  mf = 2;
end;
```

2. โปรแกรมการเรียนรู้ตัวอักษรคัดลายมือภาษาไทย

Part Program use for Training Neural Network

```
function [w1,b1,w2,b2]=mtbpnn(in_vec)
echo on
clc
```

```
% This is Program main trining backpropegretion Neurul Netwoke It is to do a step
echo off
[InPN,Col]=size(in_vec);
HiDN=input('number of Hinden Layer Node is ')
OuPN=input('number of Output Layer Node is ')
out_vec=moutvec(OuPN)
w1 = RANDOM('t',5,HiDN,InPN);
w2 = RANDOM('t',5,OuPN,HiDN);
b1 = RANDOM('t',10,HiDN,1);
b2 = RANDOM('t',10,OuPN,1);
[w1,b1,w2,b2]=tbpnn(in_vec,out_vec,w1,w2,b1,b2);
save d:/users/tong/ocr/Thesis_SaKun/weight1 w1 w2 b1 b2
```

This Function use for Training Neural Network by Back propagation

---

```
function [w1,b1,w2,b2]=tbpnn(in_vec,out_vec,w1,w2,b1,b2)

df = 10;     % Frequency of progress displays (in epochs).
me = 100000;   % Maximum number of epochs to train.
eg = 0.1;    % Sum-squared error goal.
mc = 0.95;   % Momentum constant.
lr=0.1

tp = [df me eg lr NaN NaN mc];

%     Training begin...

flops(0), tic

[w1,b1,w2,b2,bp_ep] = trainbp(w1,b1,'logsig',w2,b2,'logsig',in_vec,out_vec,tp);

bp_flops = flops;
```

```
bp_time = toc;
```

This Function use for define OutPut Note

```
function out_vec=moutvec(nout)
out_vec=eye(nout);
```

3. โปรแกรมสำหรับการทดลองของตัวอักษรที่ต้องการทราบ

Part Program use for test recognition

```
clear
echo on
clc
%                This is main Program test
%   Thai Printed Character Recognition Using Karhunen-Loeve Tranformation
%                  and Neural Network
%
```

```
%   Print file name 'filename.bmp'
%   figure 1 show all charecter in file name
%   figure 2 show testing charecter

echo off
T=input('filename.bmp is ','s')
data=imread(T);
data=pattern(data,0);
show_data=data;
figure(1);imshow(show_data);

clc
echo on
%                This is main Program test
```

```
%   Thai Printed Character Recognition Using Karhunen-Loeve Tranformation
%                   and Neural Network
%
```

```
%   Print file name 'filename.bmp'
%   figure 1 show all charecter in file name
%   figure 2 show testing charecter

echo off

mvec=1;
[Ximage Yimage]=size(data);
Check_S = 0;
count  = 0;
for X=1:Ximage;
   for Y=1:Yimage;
     if (data(X,Y)==0);
       [MinX MinY MaxX MaxY]=segment(data,X,Y);
       if ((MaxX-MinX)>=2)&((MaxY-MinY)>=2) ;
        show_data(MinX:MaxX,MinY:MaxY)=1;
        figure(1);imshow(show_data);
        clc
        echo on
%                   This is main Program test
%   Thai Printed Character Recognition Using Karhunen-Loeve Tranformation
%                   and Neural Network
%
```

```
%   Print file name 'filename.bmp'
%   figure 1 show all charecter in file name
%   figure 2 show testing charecter
```

```
        echo off

        Char=data(MinX:MaxX,MinY:MaxY);
        figure(2);imshow(Char);
        Char=pattern(Char,1);
        Char=imresize(Char,[32 16],'nearest');
        Char=Char';
        in_vec(:,mvec)=reshape(Char,512,1);
        mvec=mvec+1

echo on
%                This is main Program test
%  Thai Printed Character Recognition Using Karhunen-Loeve Tranformation
%                  and Neural Network
%
=================================================================
%  Print file name 'filename.bmp'
%  figure 1 show all charecter in file name
%  figure 2 show testing charecter
%
=================================================================
%
%                  WHAT IS CHARECTER
%  Strike any key to continue...
echo off

         test_bpnn(in_vec);
         pause
      end;
       data(MinX:MaxX,MinY:MaxY)=1;
      end;
    end;
```

```
end;
```

This Function is subprogram for test recognition

```
function test_bpnn(in_vec)
CH={'ก','ข','ค','ฆ','ง','จ','ฉ','ช','ซ','ฌ','ญ','ฎ','ฏ','ฐ','ฑ','ฒ','ณ','ด','ต','ถ','ท','ธ','น','บ','ป','ผ','ฝ','พ','ฟ','
ภ','ม','ย','ร','ล','ว','ศ','ษ','ส','ห','ฬ','อ','ฮ'};
%CHS={'ก1','ข1','ค1','ฆ1','ง1','จ1','ฉ1','ช1','ซ1','ฌ1','ญ1','ฎ1','ฏ1','ฐ1','ฑ1','ฒ1','ณ1','ด1','ต1','ถ1','
ท1','ธ1','น1','บ1','ป1','ผ1','ฝ1','พ1','ฟ1','ภ1','ม1','ย1','ร1','ล1','ว1','ศ1','ษ1','ส1','ห1','ฬ1','อ1','ฮ1'};
load weight100;
[a1,a2] = simuff(in_vec,w1,b1,'tansig',w2,b2,'tansig');
[l, idx] = sort(a2) ;

[r,c]=size(a2);
idxch=idx(r,c);
THIS_CHARECTER_IS = CH(idxch)

clear
```

End of Program

ภาคผนวก ค
ผลงานวิจัยที่ได้การตีพิมพ์

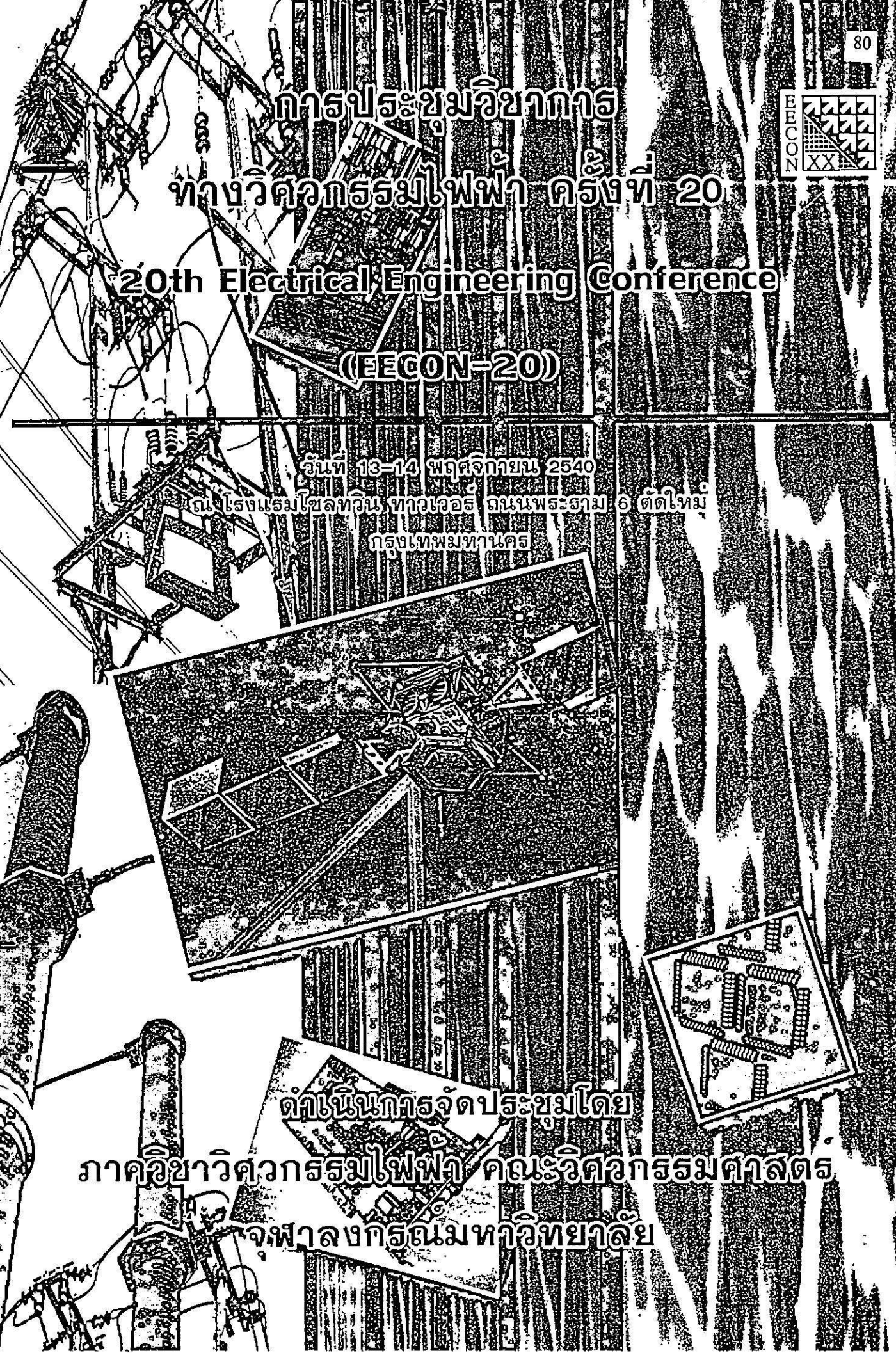
ผลงานวิจัยเรื่อง "การรู้จำตัวอักษรตัวพิมพ์ภาษาไทยโดยวิธีวัดความเหมือนแบบฟัซซี่" ได้นำเสนอและตีพิมพ์ใน งานการประชุมทางวิชาการ ทางวิศวกรรมไฟฟ้า ครั้งที่ 20 ซึ่งจัดประชุมโดย ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

การประชุมวิชาการ
ทางวิศวกรรมไฟฟ้า ครั้งที่ 20

20th Electrical Engineering Conference

(EECON-20)

วันที่ 13-14 พฤศจิกายน 2540
ณ โรงแรมโซลทวิน ทาวเวอร์ ถนนพระราม 6 ตัดใหม่
กรุงเทพมหานคร



ดำเนินการจัดประชุมโดย

ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์

จุฬาลงกรณ์มหาวิทยาลัย

การรู้จำอักษรตัวพิมพ์ภาษาไทยโดยวิธีวัดความเหมือนแบบฟัซซี่

Printed Thai Characters Recognition Using Fuzzy Similarity Measurement Method

พิศิษฎ์ โภคารัตน์กุล สกุล คำนวนชัย ชม กิ้มปาน

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ถนนฉลองกรุง ลาดกระบัง กรุงเทพฯ 10520 โทร. (02) 326-9969 โทรสาร (02) 326-9969 ต่อ 127

บทคัดย่อ

ในงานวิจัยนี้เป็นการนำเอาวิธีการวัดความเหมือนแบบฟัซซี่มาใช้ในการรู้จำอักษรตัวพิมพ์ภาษาไทย โดยการหาระยะทาง 8 ทิศทางจากกรอบสี่เหลี่ยมมายังตัวอักษร แล้ววัดหาความเหมือนแบบฟัซซี่ของระยะทางดังกล่าว ในการรู้จำจะใช้อักษรตัวพิมพ์ภาษาไทยมาตรฐานเป็นฐานข้อมูล เพื่อใช้เปรียบเทียบความเหมือนแบบฟัซซี่ระหว่างตัวอักษรมาตรฐานกับตัวอักษรที่ต้องการทราบ ซึ่งจะเป็นตัวอักษรที่มีค่าความเหมือนแบบฟัซซี่สูงสุด

Abstract

In this paper an attempt has been made to use the concept of fuzzy similarity measurement as a tool for classification of printed Thai characters. The features used for the method are the distances of the pattern form eight different points of the square frame. Using these features a fuzzy similarity function has found out. For the recognition of printed Thai characters ideal alphabets are generated using standard features. The fuzzy similarity value between the unknown pattern to be classified an all the standard Thai alphabets are then determined. The unknown pattern is recognized to be that Thai alphabet for which the fuzzy similarity valuc is maximum.

1. บทนำ

ในปัจจุบันได้มีการพัฒนาระบบการรู้จำรูปแบบตัวอักษรภาษาไทยโดยใช้วิธีการต่างๆเพื่อทำให้คอมพิวเตอร์สามารถเปลี่ยนแปลงข้อมูลของรูปภาพ ให้ไปอยู่ในรูปของ text file ที่สามารถนำไปใช้กับ word processor หรือประยุกต์ใช้กับงานอื่น ๆ ได้อย่างอัตโนมัตินั้น ต้องอาศัยวิธีการต่าง ๆ มาใช้กับการรู้จำตัวอักษรภาษาไทย วิธีการฟัซซี่ลอจิกเป็นวิธีการหนึ่งที่ได้ถูกนำมาใช้ในการพัฒนาระบบการรู้จำตัวอักษรภาษาไทยเช่น เดชา รัตนาธาร และสมชาย จิตะพันธ์กุล ได้ใช้เทคนิคฟัซซี่ลอจิกวัดความเหมือนโครงสร้างของตัวอักษรร่วมกับวิธีซินแทกติก [1] ตรีทิพย์ เดวิทย์ และเอื้อน ปิ่นเงิน ใช้วิธีหาความสัมพันธ์เสมือนแบบฟัซซี่ของรูปแบบมาตรฐานของตัวอักษรโดยอาศัยการวิเคราะห์แบบซินแท็กติก [2]

ในงานวิจัยนี้เป็นการพัฒนาระบบการรู้จำอักษรตัวพิมพ์ภาษาไทยโดยใช้วิธีวัดความเหมือนแบบฟัซซี่ของระยะทางทั้ง 8 ทิศทางจากกรอบภาพ

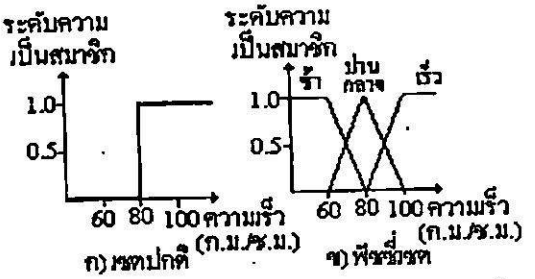
1. ทฤษฎีฟัซซี่เซต (Fuzzy Set Theory)

ฟัซซี่เซตเป็นทฤษฎีทางคณิตศาสตร์ที่ใช้วิเคราะห์ข้อมูลที่มีความคลุมเครือเพื่อการตัดสินใจของมนุษย์ และได้ถูกนำไปประยุกต์ใช้กับงานต่าง ๆ เช่น การรู้จำตัวอักษร (characters recognition) [1,2] การประมวลผลภาพ (image processing) [3] การควบคุมระบบ (control system) [4] จากคุณสมบัติของฟัซซี่เซตจะมีลักษณะของการพิจารณาปัญหาแตกต่างจากเซตปกติ (crisp set) ตัวอย่างเช่น การพิจารณาความเร็วในเซตปกติอาจกำหนดว่าความเร็วมากกว่า 80 กิโลเมตรต่อชั่วโมงถือว่าเร็ว ความเร็วน้อยกว่า 80 กิโลเมตรต่อชั่วโมงถือว่าช้า แต่สำ-หรับในฟัซซี่เซตจะแปลงความเร็วในลักษณะตัวแปรภาษา

มนุษย์ (linguistic) ในเทอมของเซต T(ความเร็ว) ดังสมการที่ 1

$$T(\text{ความเร็ว}) = \{\text{ช้า, ปานกลาง, เร็ว}\} \quad (1)$$

ฟัซซี่เซตจะแปลงความเร็วออกมาในรูปของค่าความเป็นสมาชิก (membership function : $0 \le \mu_{\text{ความเร็ว}} \le 1$) ดังแสดงในรูปที่ 1



รูปที่ 1 แสดงการแปลงความเร็วของเซตปกติกับฟัซซี่เซต

2.1 นิยามต่าง ๆ ของฟัซซี่เซต

2.1.1 นิยามที่ 1 ฟัซซี่เซต

ถ้า X เป็นเซตเอกภพ (universal set) แล้ว ฟัซซี่เซต A ใน X จะนิยามได้ว่า

$$A = \{(x, \mu_A(x)), | x \varepsilon X\} \quad (2)$$

เมื่อ μ_A คือค่าความเป็นสมาชิกของ x ในฟัซซี่เซต A

ระดับความเป็นสมาชิกของ x ในฟัซซี่เซต A อยู่ในช่วงระหว่าง 0 ถึง 1 เมื่อประยุกต์เข้ากับฟัซซี่ลอจิกค่า μ_A จะเป็นค่าความจริงที่แสดงค่าระดับของความเป็นสมาชิกของ x ซึ่งการแสดงค่าความเป็นสมาชิกของฟัซซี่เซตสามารถกระทำได้ 2 วิธีคือ วิธี numerical definition และวิธี functional definition

2.1.2 นิยามที่ 2 การยูเนี่ยน (union) ของฟัซซี่เซต

ยูเนี่ยนของฟัซซี่เซต A และ B ซึ่งมีฟังก์ชันระดับความเป็นสมาชิก $\mu_A(x)$ และ $\mu_B(x)$ ตามลำดับ และแทนด้วยฟัซซี่เซต $C = A \cup B$ สามารถนิยามได้ว่า

$$\mu_C(x) = \max(\mu_A(x), \mu_B(x)), x \varepsilon X \quad (3)$$

2.1.3 นิยามที่ 3 อินเตอร์เซคชั่น (intersection) ของฟัซซี่เซต

ฟัซซี่เซต A และ B ซึ่งมีฟังก์ชั่นระดับความเป็นสมาชิก $\mu_A(x)$ และ $\mu_B(x)$ ตามลำดับ และแทนด้วยฟัซซี่เซต $D = A \cap B$ สามารถนิยามได้ว่า

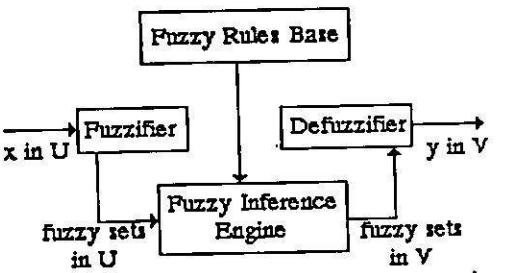
$$\mu_D(x) = \min(\mu_A(x), \mu_B(x)), x \varepsilon X \quad (4)$$

2.1.4 ถ้าฟัซซี่เซต A และ B ซึ่งมีฟังก์ชั่นระดับความเป็นสมาชิก $\mu_A(x)$ และ $\mu_B(y)$ ตามลำดับ และแทนด้วยฟัซซี่เซต $S = A + B$ สามารถนิยามได้ว่า

$$\mu_S(z) = \sup_{z=x+y} \min(\mu_A(x), \mu_B(y)) \quad (5)$$

2.2 การวิเคราะห์แบบฟัซซี่

จากรูปที่ 2 แสดงแผนผังการทำงานในระบบฟัซซี่จะประกอบด้วยส่วนสำคัญ 3 ส่วนคือ



รูปที่ 2 แสดงแผนผังการทำงานของระบบฟัซซี่

2.2.1 Fuzzifier เป็นการแปลงข้อมูลเข้าให้อยู่ในรูปความเป็นสมาชิกในระบบฟัซซี่

2.2.2 Fuzzy Inference Engine เป็นการนำค่าความเป็นสมาชิก (μ) ที่รับเข้ามาไปประมวลผลตามกฎ (rules base) ที่ระบบได้เรียนรู้หรือออกแบบไว้เพื่อหาคำตอบ

2.2.3 Defuzzifier เป็นการแปลงผลจากค่าความเป็นสมาชิกที่ได้จากขั้นตอนที่ 2 ให้อยู่ในรูปของค่า output เพื่อนำไปใช้งานในระบบการรู้จำต่อไป

2.3 ความสัมพันธ์แบบฟัซซี่ (Fuzzy Relation)

กำหนดให้ X และ Y เป็นเซต 2 เซต การหาความสัมพันธ์แบบฟัซซี่ (Fuzzy Relation) นิยามได้จากเซต R ดังนี้

$$R = \{(x, y) | x \varepsilon X, y \varepsilon Y, \text{ y ใกล้เคียงกับ x}\} \quad (6)$$

คำว่า "ใกล้เคียง" เป็นคำที่คลุมเครือ ในกรณีนี้ถ้าใช้ความเป็นสมาชิกมาช่วยระบุความใกล้เคียง ก็จะทำให้เข้าใจได้ดีขึ้น ซึ่งจะสังเกตได้ว่าการกำหนดฟังก์ชั่นความเป็น

สมาชิกแม้จะแตกต่างกันไปในแต่ละกรณี แต่ละปัญหาแต่จะต้องมีความสมเหตุสมผลอยู่ด้วย ดังนั้นจึงสามารถเขียนแจกแจงเซต R ได้ดังนี้

$$R = \int_{(x,y)} [\mu(x,y) \mid (x,y)] \qquad (7)$$

และเรียกฟัซซี่เซต R ที่มีสมาชิกเป็นคู่ อย่างเช่นกรณีนี้ว่า ความสัมพันธ์แบบฟัซซี่

2.4 ฟังก์ชั่นความเหมือนแบบฟัซซี่ (Fuzzy Similarity Function)

การแบ่งกลุ่มตัวอักษรสามารถกระทำได้โดยใช้ฟังก์ชั่นความเหมือนแบบฟัซซี่วัดลักษณะเด่นของตัวอักษร ซึ่งจะสอดคล้องกับคุณสมบัติ reflexive symmetric และ transitive [5]

2.4.1 คุณสมบัติ reflexive

$$\mu_s(x,y) = 1 \qquad (8)$$

2.4.2 คุณสมบัติ symmetric

$$\mu_s(x,y) = \mu_s(y,x) \qquad (9)$$

2.4.3 คุณสมบัติ transitive

$$\mu_s(x,y) \geq \beta_s(x,y) \qquad (10)$$

ที่ทุก ๆ (x,y) โดย $\beta_s(x,y)$ คือ

$$\beta_s(x,y) = \max [\min(\mu_s(x,z), \mu_s(z,y)] \qquad (11)$$

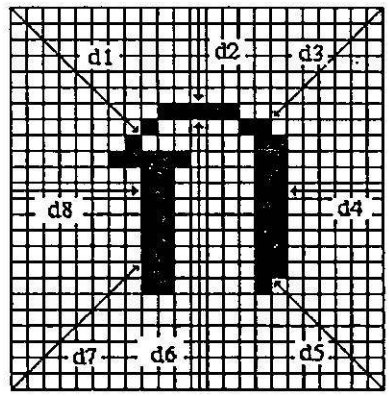
โดยที่ min (k_1,k_2) หมายถึงค่าที่น้อยกว่าระหว่าง k_1กับ k_2 กล่าวคือ ฟังก์ชั่นความเหมือนแบบฟัซซี่จะมีคุณสมบัติที่เรียกว่า transitive ก็ต่อเมื่อฟังก์ชั่นความเป็นสมาชิก $\mu_s(x,y)$ มีคุณสมบัติดังสมการที่ 10

3. การหาระยะทางทั้ง 8 ทิศทางของตัวอักษร.

การตรวจรู้โดยวิธีการนี้ เป็นการตรวจรู้จากข้อมูลทางสถิติ คือจะใช้ข้อมูลที่เป็นตัวเลขในการวิเคราะห์ความคล้ายของตัวอักษร ลักษณะเด่นที่ใช้ในการแยกแยะอักษรแต่ละตัวเพื่อใช้ในการหาค่าความสัมพันธ์ระหว่างตัวอักษรนั้นเรียกว่า fuzzy similarity relationship ซึ่งก็คือค่าระยะทางทั้ง 8 ทิศทางของตัวอักษรที่วัดภายในกรอบขนาดมาตรฐาน ซึ่งในงานวิจัยนี้ใช้กรอบรูปแบบข้อมูลภาพ 64x64

ค่าระยะทางทั้ง 8 ทิศทางที่วัดได้จากตำแหน่งต่าง ๆ ของกรอบรูปแบบข้อมูลภาพจะประกอบด้วย

- มุมบนซ้ายสุด ของกรอบ 64x64
- ตำแหน่งตรงกลางของกรอบ ทางส่วนบนของกรอบ
- มุมบนขวาสุด ของกรอบ 64x64
- ตำแหน่งตรงกลางของกรอบ ทางด้านขวาของกรอบ
- มุมล่างขวาสุด ของกรอบ 64x64
- ตำแหน่งตรงกลางของกรอบ ทางด้านล่างของกรอบ
- มุมล่างซ้ายสุดของกรอบ 64x64 และ
- ตำแหน่งตรงกลางของกรอบ ทางด้านซ้ายของกรอบ



รูปที่ 3 ทิศทางทั้ง 8 ทิศของการหาระยะทาง

การหาลักษณะเด่นนั้นเป็นขบวนการที่สำคัญในระบบการรู้จำ และลักษณะเด่นที่ใช้ในขั้นตอนนี้ก็คือ ค่าระยะทางทั้ง 8 ทิศทางของรูปแบบที่วัดจากตำแหน่งจุดต่าง ๆ ของกรอบดังรูปที่ 3 โดย ■ จะแทนข้อมูลภาพของตัวอักษร ค่าระยะทางทั้ง 8 ทิศทางนี้ทำให้สามารถหาฟังก์ชั่นที่แสดงถึงความสัมพันธ์ระหว่างตัวอักษรได้ โดยค่าความสัมพันธ์นี้จะเป็นค่าความสัมพันธ์ระหว่างตัวอักษรที่ต้องการตรวจรู้กับตัวอักษรต้นแบบทุกตัว ซึ่งหากค่าความสัมพันธ์ระหว่างตัวอักษรที่ต้องการตรวจรู้กับตัวอักษรต้นแบบตัวใดมีค่ามากที่สุด จะได้ว่าตัวอักษรที่ต้องการตรวจรู้นั้นก็คืออักษรตัวนั้นนั่นเอง

4. การเก็บข้อมูลลักษณะเด่นของตัวอักษรต้นแบบ

ในการหาค่าความสัมพันธ์ระหว่างตัวอักษร จำเป็นจะต้องหาค่าระยะทางทั้ง 8 ทิศทางของตัวอักษรต้นแบบทุกตัวเพื่อใช้เป็นฐานข้อมูลในการเปรียบเทียบกับตัวอักษรที่ต้องการตรวจรู้ มีขั้นตอนดังต่อไปนี้

4.1 นำตัวอักษรต้นแบบที่จะใช้เป็นฐานข้อมูลมาผ่านขบวนการการจัดการล่วงหน้า โดยการกำจัดสัญญาณรบกวนใช้วิธีการของ [6] ทำขอบภาพตัวอักษรให้เรียบใช้วิธีการของ [7] ทำการดึงข้อมูลของแต่ละตัวอักษรภาษาไทยออกจากข้อมูลของรูปประโยคในแต่ละบรรทัดใช้วิธีการของ [8] และทำตัวอักษรให้มีขนาดมาตรฐานโดยการหาค่า D_{max} ระหว่างความกว้าง (w) กับความสูง (h) ของตัวอักษร ถ้า h > w h จะเป็นค่า D_{max} จะได้อัตราส่วนขยายภาพความสูงของตัวอักษรเป็น 40 / D_{max} และอัตราส่วนขยายภาพความกว้างของตัวอักษรเป็น w / h ในทำนองเดียวกัน ถ้า w > h w จะเป็นค่า D_{max} จะได้อัตราส่วนขยายภาพความกว้างของตัวอักษรเป็น 40 / D_{max} และอัตราส่วนขยายภาพความสูงของตัวอักษรเป็น h / w จากวิธีการดังกล่าวข้างต้นจะได้ตัวอักษรขนาด 40 X 40 จุดภาพ จัดตัวอักษรที่ได้ให้อยู่ในสมมาตรภายในกรอบ 64 X 64 จุดภาพ เพื่อหาระยะทางทั้ง 8 ทิศทาง แล้วเข้าสู่ขบวนการการวัดความเหมือนแบบฟัซซี่

4.2 ภายในกรอบรูปแบบข้อมูลภาพขนาด 64x64 จุดภาพ ทำการหาค่าระยะทางทั้ง 8 ทิศทางที่วัดมาถึงโครงร่างของตัวอักษรอันประกอบด้วยระยะทาง $d_1,d_2,d_3,d_4,d_5,d_6,d_7$ และ d_8 ตามลำดับ

4.3 นำค่าระยะทางทั้ง 8 มาจัดเข้ารูปสมการเรียกว่า feature vector

$$FV=(d_1,d_2,d_3,d_4,d_5,d_6,d_7,d_8) \qquad (12)$$

4.4 ทำการนอร์มอลไลซ์ (normalized) ระยะทางทั้ง 8 นั้นโดยการนำเอาค่าระยะทางที่มากที่สุดที่หาได้ มาหารค่าระยะทางทั้ง 8 ทิศทางนั้น ดังสมการที่ 13

$$a_i = d_i / \max(d_i) \quad \text{โดย } i=1,2,3,...,8 \qquad (13)$$

แล้วจัดเข้ารูปสมการเป็น normalized feature vector ดังนี้

$$FVN=(a_1,a_2,a_3,a_4,a_5,a_6,a_7,a_8) \qquad (14)$$

นำค่าที่ได้เก็บไว้เป็นฐานข้อมูลของตัวอักษรต้นแบบโดยทำในลักษณะเดียวกันนี้กับตัวอักษรต้นแบบทุกตัว

5.ขั้นตอนการหาค่าความสัมพันธ์ระหว่างตัวอักษร (Fuzzy Similarity Relationship)

ในการหาค่าความสัมพันธ์ระหว่างตัวอักษร ในขั้นแรกนั้น จะนำตัวอักษรที่ต้องการตรวจรู้ที่ผ่านขบวนการการจัดการล่วงหน้า มาแยกตัวอักษรออกจากประโยคโดยใช้วิธีการของ [8] ทำตัวอักษรให้มีขนาดมาตรฐานตามวิธีที่กล่าวมาแล้วในข้อ 4 ทำการหาระยะทางทั้ง 8 ทิศทาง และทำการนอร์มอลไลซ์ระยะทางทั้ง 8 ทิศทางดังวิธีการที่กล่าวไปแล้วในขบวนการการเก็บข้อมูลลักษณะเด่นของตัวอักษรต้นแบบในหัวข้อที่ 4 โดยจะเป็นไปในลักษณะเดียวกัน

ขั้นตอนการตรวจรู้ตัวอักษรจะกระทำได้โดยอาศัยฟังก์ชันที่ใช้ในการหาค่าความสัมพันธ์ระหว่างตัวอักษรที่ต้องการตรวจรู้กับตัวอักษรต้นแบบ โดยอาศัยค่าระยะทางทั้ง 8 ทิศทางของตัวอักษรที่ต้องการตรวจรู้กับระยะทางทั้ง 8 ทิศทางของตัวอักษรต้นแบบทุกตัวในฐานข้อมูล มาวัดความเหมือนแบบฟัซซี่ ถ้า normalized feature vector ของตัวอักษรถูกแทนด้วย x จะได้ว่า

$$\{x_i\} \in X \text{ หรือ } x_i \in X \text{ เมื่อ } i=1,2,...,N \qquad (15)$$

เมื่อ N คือจำนวนตัวอักษร ซึ่งจะหาค่าความเหมือนแบบฟัซซี่ได้ตามสมการดังนี้

$$\mu_s(x_i,x_j) = 1 - \left[\sum_{k=1}^{8} (a_i^k - a_j^k)^2\right]^{1/2} \qquad (16)$$

เมื่อ $x_i,x_j \in X$

a_i^k คือระยะทางของตัวอักษรที่ต้องการตรวจรู้ที่ผ่านการหารด้วยค่าสูงสุด (สมการ 13)

a_j^k คือระยะทางของตัวอักษรต้นแบบที่ผ่านการหารด้วยค่าสูงสุด (สมการ 13)

โดยสมการที่ (16) จะมีคุณสมบัติของฟังก์ชั่นความเหมือนแบบฟัซซี่คือ reflexive symmetric และ max-min transitive ดังนั้นถ้า

$$\mu_s(x_i,x_j) = 1-[\sum_{k=1}^{s}(a_i^k - a_j^k)^2]^{1/2} = 1 \qquad (17)$$

จะกล่าวว่า $\mu_s(x_i,x_j)$ มีคุณสมบัติ reflexive ถ้า

$$\mu_s(x_j,x_i) = 1-[\sum_{k=1}^{s}(a_j^k - a_i^k)^2]^{1/2}$$

$$= 1-[\sum_{k=1}^{s}(a_i^k - a_j^k)^2]^{1/2}$$

$$= \mu_s(x_i,x_j) \qquad (18)$$

เรียกว่า $\mu_s(x_i,x_j)$ มีคุณสมบัติ symmetric แต่ถ้า

$$\mu_s(x_i,x_j) \geq \vee[\mu_s(x_i,x_\lambda) \wedge \mu_s(x_\lambda,x_j)] \qquad (19)$$

เมื่อ $\vee$ แทนค่า maximum และ $\wedge$ แทนค่า minimum เรียกว่า $\mu_s(x_i,x_j)$ มีคุณสมบัติ transitive

จากคุณสมบัติดังกล่าวข้างต้นของสมการที่ 16 จะนำมาใช้ในการตรวจรู้อักษรตัวพิมพ์ภาษาไทยในงานวิจัยนี้ โดยวัดค่าความเหมือนของระยะทางทั้ง 8 ทิศทางของตัว ร

$$\mu_s(\varnothing, X_i) = 1-[\sum_{k=1}^{s}(a_\varnothing^k - a_i^k)^2]^{1/2} \qquad (20)$$

โดย $\varnothing$ คือ ตัวอักษรที่ต้องการตรวจรู้

X_i คือ กลุ่มตัวอักษรต้นแบบทั้งหมดภายในฐานข้อมูล

$a_\varnothing^k$ คือ ระยะทางของอักษรที่ต้องการตรวจรู้ที่ผ่านการหารด้วยค่าสูงสุด

a_i^k คือ ระยะทางของตัวอักษรต้นแบบทั้งหมดในฐานข้อมูล

ตัวอักษรที่ต้องการตรวจรู้ $\varnothing$ จะถูกตรวจรู้ให้เป็นตัวอักษรต้นแบบตัวนั้น ๆ ก็ต่อเมื่อค่าความสัมพันธ์ที่คำนวณได้ (μ_s) มีค่ามากที่สุดจากค่าความสัมพันธ์ทั้งหมดที่คำนวณได้จากตัวอักษรต้นแบบทั้งหมดที่มีอยู่ในฐานข้อมูล เช่นมันจะถูกตรวจรู้ให้เป็นอักษรตัวที่ m ถ้า

$$\mu_s(\varnothing,X_m) > \mu_s(\varnothing,X_i) \qquad (21)$$

สำหรับ X_iทุกตัว โดยที่ $X_m \neq X_i$

$\mu_s(\varnothing,X_m)$ ที่หาได้จากสมการที่ 20 จะมีค่ามากที่สุดถ้าเป็นตัวอักษรเดียวกันเช่น $\mu_s(\text{ก}, X_m)$ (ภาพ ก เป็นภาพที่ต้องการทราบว่าเป็นอักษรตัวใด) จะมีค่ามากที่สุดคือประมาณ 1 เมื่อนำไปเปรียบเทียบความเหมือนแบบฟัซซี่กับ X_m ที่เป็นตัวอักษร ก ในฐานข้อมูลเท่านั้น ถ้านำไปเปรียบเทียบกับ X_i ที่เป็นตัวอักษรอื่น ๆ ในฐานข้อมูลจะมีค่าน้อยกว่า ในกรณีการตรวจรู้อักษรตัวพิมพ์ภาษาไทยให้ผลลัพธ์ที่อาจเป็นไปได้มากกว่า 1 ตัวอักษร จะต้องหาความเหมือนแบบฟัซซี่ของทิศทางทั้ง 8 ทิศทางในแต่ละควอทแดรนท์เพิ่มขึ้น หรืออาจกระทำได้โดยการเพิ่มจำนวน feature vector (ระยะทางจากกรอบภาพในทิศทางต่าง ๆ) ให้มากกว่า 8 ทิศทาง จึงจะสามารถทราบได้ว่าตัวอักษรนั้นเป็นอักษรตัวใด ตัวอักษรที่คล้ายคลึงกัน และต้องใช้วิธีการดังกล่าวข้างต้นได้แก่ กลุ่มตัวอักษร ก ค กลุ่มตัวอักษร ฎ ฏ

6. การทดสอบการรู้จำ

ในการทดสอบการรู้จำ ใช้ตัวอักษร 2 รูปแบบ คือ Cordia UPC และ Angsana UPC ประกอบด้วย พยัญชนะ สระ วรรณยุกต์ ตัวเลขไทย และตัวเลขอารบิค รวมทั้งหมด 890 ตัว ขนาดตั้งแต่ 20-28 point จากเอกสารที่พิมพ์ด้วยเครื่องพิมพ์เลเซอร์ และ ink jet นำไปสแกนด้วยเครื่องตรวจกวาดภาพขนาดความละเอียด 300 dpi ประมวลผลด้วยเครื่องไมโครคอมพิวเตอร์ PC รุ่น Pentium-60 ความเร็ว 60 เมกกะเฮิรตซ์ มีอัตราการรู้จำเฉลี่ย 98.09% จากจำนวนตัวอักษรทั้งหมด 890 ตัวอักษร โดยมีการรู้จำผิด 17 ตัวอักษร หรือผิดพลาดไป 1.91 %

7. สรุป และวิจารณ์

การวัดความเหมือนแบบฟัซซี่ของระยะทางทั้ง 8 ทิศทางของตัวอักษร จะต้องเก็บสถิติค่านอร์มอลไลซ์

ตารางที่ 1 ผลการทดสอบการรู้จำ

ชื่อรูปแบบตัวอักษร	ขนาด (point)	จำนวน	รู้จำได้	รู้จำผิด	ความถูกต้อง (%)
Angsana UPC	20	89	86	3	96.63
	22	89	87	2	97.75
	24	89	88	1	98.88
	26	89	86	3	96.63
	28	89	89	0	100
Cordia UPC	20	89	87	2	97.75
	22	89	89	3	96.63
	24	89	87	2	97.75
	26	89	88	1	98.88
	28	89	86	0	100
	รวม	890	873	17	98.09

ของระยะทางทั้ง 8 ทิศทางของตัวอักษรต้นแบบ เพื่อใช้เป็นฐานข้อมูลในการวัดความเหมือนแบบฟัซซี่ ซึ่งจากการทดลองจะพบว่าข้อมูลที่นำมาทดสอบที่ได้จากเครื่องตรวจกวาดภาพขนาดความละเอียด 300 dpi จะมีผลต่อตัวอักษรที่มีขนาดเล็กกว่า 20 point คือตัวอักษรจะเบลอ และผิดพลาดสูง หรือติดกันกับตัวอักษรข้างเคียงจนไม่สามารถแยกออกจากกันได้ นอกจากนั้นการเอียงของตัวอักษรจะต้องไม่มากกว่า 10 องศา เพราะจะทำให้ระยะทาง 8 ทิศทางผิดพลาด ซึ่งจะส่งผลกระทบต่อการรู้จำของตัวอักษร

การวิจัยนี้เป็นการนำเอาเทคนิคการวัดความเหมือนแบบฟัซซี่ของระยะทาง 8 ทิศทางของตัวอักษรมาใช้ในการรู้จำตัวอักษร โดยไม่ต้องใช้ร่วมกับวิธีชินแทกติกหรือวิธีอื่น ๆ และสามารถกระทำได้โดยง่าย ไม่จำเป็นต้องทำตัวอักษรให้บาง ซึ่งลดขั้นตอนในการประมวลผลลง ผลการทดสอบกับตัวอักษร 2 รูปแบบที่มีขนาดต่าง ๆ กัน แบบละ 5 ขนาดตัวอักษร พบว่าอัตราการรู้จำประมาณ 98 %

8. เอกสารอ้างอิง

[1] เดชา รัตนาธาร และสมชาย จิตะพันธ์กุล "การรู้จำตัวอักษรพิมพ์ภาษาไทยโดยใช้เทคนิคแบบฟัซซี่โลจิกและวิธีชินแทคติก" ประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 18 โรงแรมแอมบาสซาเดอร์ ระยอง 22-24 พฤศจิกายน 2537 หน้า 971-976.

[2] ศรีทิพย์ เตวิทย์ และเอื้อน ปิ่นเงิน "การรู้จำรูปแบบโดยใช้ความสัมพันธ์เสมือนแบบฟัซซี่ " วิศวสารลาดกระบัง ปีที่ 13 ฉบับที่ 2 เมษายน 2540 หน้า 10-19.

[3] กิตติ ไพฑูรย์วัฒนกิจ และสาธิต อินทรจักร "การปรับปรุงภาพอย่างอัตโนมัติโดยใช้ฟัซซี่" วิศวกรรมสาร วารสารทางวิชาการวิศวกรรมสถานแห่งประเทศไทยฯ ปีที่ 46 ,ฉบับที่ 6 ,หน้า 86-89 ,2536.

[4] ภากร ศันสนียวานิช, สุวรรณ อรุณวิวัฒน์, เสริมศักดิ์ เลิศมโนรัตน์ และพิศิษฏ์ โภคารัตน์กุล "การควบคุมการผลิตพลังงานไฟฟ้าโดยใช้ฟัซซี่ลอจิกที่เรียนรู้ระบบได้ด้วยตัวเอง" ประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 19 โรงแรมเจริญธานีปริ๊นเซส ขอนแก่น 7-8 พฤศจิกายน 2539.

[5] Abraham Kandel, Fuzzy Techniques in Pattern Recognition. John Wiley & Sons,1982.

[6] สุรพันธ์ เอื้อไพบูลย์ การจดจำตัวอักษรลายมือเขียนภาษาไทยโดยการพิจารณาหัวของตัวอักษร วิทยานิพนธ์มหาบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง ปีการศึกษา 2531.

[7] พิศิษฏ์ โภคารัตน์กุล และชม กิ้มปาน "การรู้จำอักษรตัวพิมพ์ภาษาไทยโดยวิธีหารหัสดวอไซโทโปโลยี่" การประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 19 โรงแรมเจริญธานีปริ๊นเซส ขอนแก่น 7-8 พฤศจิกายน 2539.

[8] Chom Kimpan, Printed Thai Characters Recognition Dissertation Doctor of Engineering , Faculty of Engineering, King Mongkut's Institue of Technology chaokhun Taharn Lardkrabang, Academic year 1986.

ประวัติผู้เขียน

นายสกุล คำนวนชัย เกิดเมื่อวันที่ 20 สิงหาคม 2508 ที่จังหวัดเชียงใหม่ สำเร็จการศึกษาครุศาสตร์อุตสาหกรรมบัณฑิต(วิศวกรรมโทรคมนาคม) จากสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปีการศึกษา 2531

ปี พ.ศ. 2531 เข้ารับราชการในตำแหน่ง อาจารย์ 1 ระดับ 3 สังกัดวิทยาลัยเทคนิคนครปฐม กรมอาชีวศึกษา กระทรวงศึกษาธิการและปัจจุบันดำรงตำแหน่ง อาจารย์ 1 ระดับ 5 สังกัดวิทยาลัยเทคนิคลพบุรี กรมอาชีวศึกษา กระทรวงศึกษาธิการ