

การประยุกต์ใช้เมตริกซ์ค่าน้ำหนักร่วมกับแบบจำลองโครงข่ายประสาทเทียม
แบบลำดับชั้นสำหรับการรู้จำตัวอักษรลายมือเขียนภาษาไทย

APPLYING A WEIGHTING MATRIX TO A HIERARCHICAL NETWORK
MODEL FOR HANDWRITTEN THAI CHARACTER RECOGNITION

พัชรวิทย์ พูลสำราญ
PATCHARAWADEE POOLSAMRAN

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของกรณีศึกษาของหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2549

ISBN 974-98-0805-7

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การประยุกต์ใช้เมตริกซ์ค่านำหน้าร่วมกับแบบจำลองโครงข่ายประสาทเทียม
แบบลำดับชั้นสำหรับการรู้จำตัวอักษรลายมือเขียนภาษาไทย

APPLYING A WEIGHTING MATRIX TO A HIERARCHICAL NETWORK
MODEL FOR HANDWRITTEN THAI CHARACTER RECOGNITION



พัชรวดี พูลสำราญ

PATCHARAWADEE POOLSAMRAN

ฉพ.

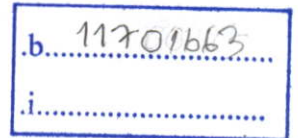
พ 516 ก

2549

เลขหมู่.....

เลขทะเบียน..... 69080

วัน,เดือน,ปี..... 7 ก.พ. 2550



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2549

ISBN 974-83-0805-7

**APPLYING A WEIGHTING MATRIX TO A HIERARCHICAL NETWORK
MODEL FOR HANDWRITTEN THAI CHARACTER RECOGNITION**

PATCHARAWADEE POOLSAMRAN

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF SCIENCE IN INFORMATION TECHNOLOGY
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2006

ISBN 974-83-0805-7

COPYRIGHT 2006

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

หัวข้อวิทยานิพนธ์	การประยุกต์ใช้เมตริกซ์ค่าน้ำหนักร่วมกับแบบจำลองโครงข่ายประสาทเทียมแบบลำดับชั้นสำหรับการรู้จำตัวอักษรลายมือเขียนภาษาไทย
นักศึกษา	นางสาวพัชรวดี พูลสำราญ
รหัสนักศึกษา	45061530
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
พ.ศ.	2549
อาจารย์ที่ปรึกษาวิทยานิพนธ์	รศ. ดร. อาริต ธรรมโน

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้นำเสนอแบบจำลองโครงข่ายประสาทเทียมเพื่อการรู้จำตัวอักษรลายมือเขียนภาษาไทยโดยไม่มีภารกิจคุณลักษณะเด่นของภาพ ในขั้นตอนการประมวลผลเบื้องต้นจะนำภาพตัวอักษรมาแบ่งเป็น 9 บล็อกขนาดเท่าๆ กันและนำค่าคุณภาพของแต่ละบล็อกมาสร้างเป็นเวกเตอร์อินพุตเพื่อนำไปใช้ในกระบวนการรู้จำ สำหรับสถาปัตยกรรมของโครงข่ายประสาทเทียมที่นำเสนอเป็นโครงข่ายประสาทเทียมแบบลำดับชั้นที่มีการประยุกต์ใช้เมตริกซ์ค่าน้ำหนักในการวัดค่าความเหมือนระหว่างภาพอินพุตที่เข้ามาเรียนรู้กับภาพโมเดลอ้างอิง โดยสถาปัตยกรรมดังกล่าวประกอบด้วย 4 ระดับชั้นคือชั้นอินพุต ชั้นซ่อน ชั้นคลัสเตอร์และชั้นเอาต์พุตตามลำดับ ส่วนข้อมูลที่ใช้ในงานวิจัยนี้เป็นข้อมูลลายมือเขียนแบบออฟไลน์จากฐานข้อมูลของศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติซึ่งข้อมูลดังกล่าวรวบรวมจากผู้เขียนจำนวน 63 คน โดยผู้เขียนแต่ละคนจะเขียนตัวอักษรพยัญชนะทั้งหมด 42 ตัวเขียนครั้งละ 1 ตัวอักษรจำนวน 3 ครั้ง

การทดลองในงานวิจัยนี้ประกอบด้วย 2 ชุดการทดลอง ชุดแรกเป็นการทดลองเพื่อทดสอบการรู้จำลายมือเขียนที่ขึ้นอยู่กับผู้เขียนและชุดที่ 2 การทดลองเพื่อทดสอบการรู้จำลายมือเขียนที่ไม่ขึ้นอยู่กับผู้เขียน ในการทดลองชุดที่ 1 จะแบ่งข้อมูลลายมือเขียนจำนวน 2 เซตของผู้เขียนทั้ง 63 คนเป็นข้อมูลสำหรับการเรียนรู้และข้อมูลที่เหลืออีก 1 เซตไว้สำหรับการทดสอบ ส่วนการทดลองชุดที่ 2 จะนำข้อมูลลายมือเขียนทั้ง 3 เซตของผู้เขียนจำนวน 42 คนเป็นข้อมูลที่ใช้สำหรับการเรียนรู้และลายมือเขียนของผู้เขียนที่เหลือจำนวน 21 คนจะนำไปใช้ในการทดสอบ

Thesis Title	Applying a Weighting Matrix to a Hierarchical Network Model for Handwritten Thai Character Recognition
Student	Miss Patcharawadee Poolsamran
Student ID.	45061530
Degree	Master of Science
Programme	Information Technology
Year	2006
Thesis Advisor	Assoc. Prof. Dr. Arit Thammano

ABSTRACT

This thesis proposes a new neural network approach to handwritten Thai character recognition that does not require feature extraction process. In preprocessing, the input is divided into 9 equal windows and input vector will be extracted from each individual window. This new neural network is a hierarchical neural network; it employs the concept of a weighting matrix in measuring the similarity between the incoming input pattern and the reference patterns. The proposed hierarchical network model has four layers which are input layer, hidden layer, cluster layer and output layer. The handwritten character images used in this study are the Off-line Thai Handwritten Character corpus developed by the National Electronic and Computer Technology Center (NECTEC). This benchmark data set is written by 63 people. Each person was asked to write 42 Thai characters 3 times.

Our experiments are performed in 2 modes, which are a writer dependent mode and a writer independent mode. The writer dependent mode, two examples of each writer are used for training and the remaining one example is used for testing. In the writer independent mode, data set written by 42 people are used for training and the remaining 21 data are used for testing.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้อย่างดี ด้วยการชี้แนะ การให้คำปรึกษาและการดูแลเอาใจใส่จาก รศ.ดร. อาริต ธรรมโน ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์ ข้าพเจ้ารู้สึกทราบซึ่งในความกรุณาจากท่านอาจารย์เป็นอย่างมากและขอขอบพระคุณเป็นอย่างสูง

ขอกราบขอบพระคุณคณาจารย์สาขาวิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า

ขอขอบคุณมหาวิทยาลัยบูรพาที่ให้โอกาสและการสนับสนุนการศึกษาในครั้งนี้

ขอขอบคุณเพื่อนๆ พี่ๆ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังและมหาวิทยาลัยบูรพาทุกคนที่ให้คำแนะนำและคอยให้กำลังใจเสมอมา

ขอขอบคุณเจ้าหน้าที่คณะเทคโนโลยีสารสนเทศ และบัณฑิตวิทยาลัย ที่ให้ความช่วยเหลือและคำแนะนำในการทำวิทยานิพนธ์

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณบิดา มารดา ที่คอยให้กำลังใจและปลอบโยนที่ทำให้ข้าพเจ้าเข้มแข็งและมีกำลังใจในการทำวิทยานิพนธ์ฉบับนี้ให้สำเร็จลุล่วงได้ด้วยดี และขอขอบใจน้องสาวและน้องชายของข้าพเจ้าด้วยเช่นกัน

คุณค่าและประโยชน์อันพึงมาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน

พัชรวิดี พูลสำราญ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	1
1.3 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย.....	2
1.4 ขอบเขตของการวิจัย.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.6 ขั้นตอนของการศึกษา.....	3
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง.....	4
2.1 บทนำ.....	4
2.2 ทฤษฎีที่เกี่ยวข้อง.....	4
2.2.1 โครงข่ายประสาทเทียมแบบซิมพลิไฟด์พีชชีอาร์ทแมพ.....	4
2.2.2 เทคนิคการหาโครงร่าง.....	7
2.3 บทความงานวิจัยที่เกี่ยวข้อง.....	11
2.3.1 Human Face Recognition using A Spatially Weighted Modified Hausdorff Distance.....	11
2.3.2 Spatially eigen-weighted Hausdorff distance for human face recognition....	14
2.3.3 A Novel Feature Extraction Technique for the Recognition of Segmented Handwritten Character.....	16
2.3.4 An Approach to Word Image Matching Based on Weighted Hausdorff distance.....	19
2.3.5 Off-line Handwritten Thai Characters from Word Script.....	21

สารบัญ (ต่อ)

หน้า

2.3.6 Distinctive Feature Analysis for Thai Handwritten Character Recognition based on modified Stroke Changing Sequence	24
2.3.7 Thai OCR : A Neural Network Application	26
บทที่ 3 การประยุกต์ใช้เมตริกซ์ค่าน้ำหนักร่วมกับแบบจำลองโครงข่ายประสาทเทียมแบบลำดับชั้น สำหรับการรู้จำตัวอักษรลายมือเขียนภาษาไทย.....	28
3.1 บทนำ.....	28
3.2 การเตรียมเวกเตอร์อินพุต	29
3.3 สถาปัตยกรรมโครงข่ายประสาทเทียมแบบลำดับชั้นที่นำเสนอ	29
3.4 กระบวนการเรียนรู้	31
3.5 วิธีการวัดระยะทางที่นำเสนอ.....	35
3.6 ตัวอย่างการคำนวณวิธีการวัดระยะทางที่นำเสนอ	36
บทที่ 4 การทดสอบการประยุกต์ใช้เมตริกซ์ค่าน้ำหนักร่วมกับแบบจำลองโครงข่ายประสาทเทียม แบบลำดับชั้นสำหรับการรู้จำตัวอักษรลายมือเขียนภาษาไทย.....	39
4.1 ข้อมูลทดสอบระบบ	39
4.2 การกำหนดพารามิเตอร์ของระบบ	40
4.3 ผลการทดลอง.....	40
4.3.1 การทดสอบการรู้จำตัวอักษรลายมือเขียนที่ขึ้นอยู่กับผู้เขียน	41
4.3.2 การทดสอบการรู้จำตัวอักษรลายมือเขียนที่ไม่ขึ้นอยู่กับผู้เขียน	42
4.3.3 การทดสอบจำนวนบล็อกของการแบ่งบล็อกภาพตัวอักษร	44
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ	47
5.1 สรุปผลการวิจัย.....	47
5.2 ข้อเสนอแนะ.....	49
บรรณานุกรม.....	50

สารบัญ (ต่อ)

	หน้า
ภาคผนวก	52
ภาคผนวก ก ตัวอย่างข้อมูลตัวอักษรที่ใช้ในการทดสอบโปรแกรม.....	53
ภาคผนวก ข. ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่.....	68
ประวัติผู้เขียน	75

สารบัญตาราง

ตารางที่	หน้า
2.1 การแบ่งกลุ่มของตัวอักษรในภาษาไทย.....	22
4.1 ตัวอย่างตัวอักษรพยัญชนะลายมือเขียนภาษาไทย.....	39
4.2 ผลการทดลองการรู้จำตัวอักษรลายมือเขียนที่ขึ้นอยู่กับผู้เขียน	41
4.3 ผลการทดลองการรู้จำตัวอักษรลายมือเขียนที่ไม่ขึ้นกับผู้เขียน	43
4.4 ผลการทดลองของการแบ่งบล็อกภาพ	46

สารบัญรูป

รูปที่	หน้า
2.1 สถาปัตยกรรมของโครงข่ายประสาทเทียมซิมพลิไฟด์พีซซีอาร์ทแมพ.....	5
2.2 ภาพตัวอักษรก่อนและหลังผ่านกระบวนการหาโครงร่าง	8
2.3 ตัวอย่างจุดภาพที่เป็นจุดเชื่อมและจุดสิ้นสุด.....	8
2.4 จุดภาพ p ที่จะพิจารณาลบภายในหน้าต่างขนาด 3×3	8
2.5 ตัวอย่างจุดภาพ p ที่ไม่สามารถลบได้ตามเงื่อนไข ($2 \leq B(p) \leq 6$)	10
2.6 ..ตัวอย่างจุดภาพ p ที่ไม่สามารถลบได้ตามเงื่อนไข ($X_H(p) = 1$)	10
2.7 ตัวอย่างจุดภาพ p ที่สามารถลบได้ตามเงื่อนไข $(x_1 \cdot x_3 \cdot x_5 = 0) \wedge (x_3 \cdot x_5 \cdot x_7 = 0)$	11
2.8 ตัวอย่างจุดภาพ p ที่สามารถลบได้ตามเงื่อนไข $(x_3 \cdot x_1 \cdot x_7 = 0) \wedge (x_1 \cdot x_7 \cdot x_5 = 0)$	11
2.9 กราฟสามมิติของฟังก์ชันน้ำหนัก $w(x)$	13
2.10 (a) เวกเตอร์ไอเคนอันดับหนึ่งและ (b) ค่าฟังก์ชันน้ำหนักไอเคน	15
2.11 (a) โครงร่างของเส้น, (b) ค่าพิกเซลของเส้น, (c) การระบุทิศทางของแต่ละจุด, (d) เส้นที่ผ่านการนอร์มอลไลซ์ทิศทาง.....	17
2.12 (a) โครงร่างที่ผ่านการนอร์มอลไลซ์ทิศทาง, (b) บล็อกของภาพ, (c) เวกเตอร์คุณลักษณะสำคัญของบล็อก	18
2.13 การตัดช่องว่างของภาพคำ.....	19
2.14 การแบ่งภาพคำภาษาอังกฤษเพื่อกำหนดค่าน้ำหนัก	20
2.15 การแบ่งภาพคำภาษาจีนเพื่อกำหนดค่าน้ำหนัก	20
2.16 การแยกตัวอักษรเดี่ยว	22
2.17 (a) การหาจุดปลาย (b) การหาตำแหน่งของลูป (c) Cp pattern.....	22
2.18 ข้อมูลคุณลักษณะ FC, LT, SR และ WH.....	23
2.19 การหาลำดับการเปลี่ยนแนวเส้นในแนวนอน	25
2.20 การหาลำดับการเปลี่ยนแนวเส้น ณ ตำแหน่งลูปของตัวอักษร	25
2.21 ภาพตัวอักษรที่มีปัญหาจำนวนลูปที่เกิดจากการเขียน	26
2.22 วิธีการย่อหรือขยายข้อมูลอินพุตลงในเมตริกซ์อินพุต	27
3.1 ขั้นตอนการประมวลผลภาพเบื้องต้น	28
3.2 การเตรียมเวกเตอร์อินพุต.....	29
3.3 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบลำดับชั้นที่นำเสนอ.....	30
3.4 แผนภาพขั้นตอนกระบวนการเรียนรู้	34

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.5 ตัวอย่างเซตข้อมูลอินพุทและเซตค่าน้ำหนัก	36
4.1 การแบ่งภาพตัวอักษรขนาด 4, 9 และ 16 บล็อก.....	45

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การรู้จำรูปแบบ (pattern recognition) เป็นแขนงวิชาหนึ่งทางด้านปัญญาประดิษฐ์ที่มีการนำคอมพิวเตอร์มาช่วยในการตัดสินใจเกี่ยวกับการจัดกลุ่มและการจำแนกกลุ่มข้อมูล เช่น การทำให้คอมพิวเตอร์จดจำและตัดสินใจได้ว่า ข้อมูลภาพที่กำลังพิจารณาเป็นข้อมูลภาพของตัวอักษรใด ข้อมูลเสียงที่เข้ามาเป็นเสียงพูดของตัวอักษรอะไร ซึ่งความสามารถในการแก้ไขปัญหาลักษณะนี้เป็นเรื่องง่ายสำหรับกระบวนการคิดของมนุษย์ เนื่องจากมนุษย์มีการสั่งสมความรู้และประสบการณ์ที่สามารถนำมาประยุกต์ใช้สถานการณ์ที่เปลี่ยนแปลงได้อย่างดีเยี่ยม ดังนั้นจึงได้มีความพยายามที่จะเลียนแบบกระบวนการแก้ปัญหาของมนุษย์เพื่อให้คอมพิวเตอร์สามารถประมวลผลและจดจำสิ่งต่างๆ ได้เสมือนการทำงานของเซลล์ประสาทของมนุษย์ ซึ่งแนวคิดนี้เรียกว่า โครงข่ายประสาทเทียม (Artificial Neural Networks) และปัจจุบันได้มีการประยุกต์ใช้โครงข่ายประสาทเทียมในการรู้จำตัวอักษรมากขึ้นและยังให้ประสิทธิภาพสูงด้วย [1]

ระบบการรู้จำประกอบด้วยขั้นตอนสำคัญดังนี้ 1. การประมวลผลข้อมูลเบื้องต้น เช่น การกำจัดสัญญาณรบกวน (noise removal) การหาโครงร่าง (skeleton) เป็นต้น 2. การดึงคุณลักษณะเด่น (feature extraction) ซึ่งเป็นขั้นตอนที่สำคัญมากเพราะต้องพิจารณาเลือกเทคนิคที่เหมาะสมที่ทำให้ได้ข้อมูลคุณลักษณะเด่นที่ยังคงความสามารถในการแบ่งแยกความแตกต่างของข้อมูลแต่ละกลุ่มออกจากกันได้ และ 3. การรู้จำ ซึ่งมีหลากหลายวิธีอาทิเช่น การจับคู่กับแม่แบบ (template matching) แบบจำลองฮิดเดนมาร์คอฟ (Hidden Markov Model) แบบจำลองโครงข่ายประสาทเทียม เป็นต้น

เมื่อพิจารณาขั้นตอนของระบบการรู้จำจะพบว่า กระบวนการดึงคุณลักษณะเด่นของภาพนอกจากเป็นปัจจัยสำคัญที่มีผลต่อประสิทธิภาพในการรู้จำแล้ว แต่ละเทคนิควิธีที่ถูกพัฒนาขึ้นยังยึดติดกับชุดข้อมูลที่เรียนรู้ด้วย ดังนั้นงานวิจัยนี้จึงมุ่งเน้นการพัฒนาเทคนิคใหม่ในการรู้จำโดยไม่มีการดึงคุณลักษณะเด่นของภาพมาประยุกต์ใช้ร่วมกับโครงข่ายประสาทเทียมแบบซิมพลิไฟด์ฟัซซีอาร์ทแมพ (Simplified Fuzzy ARTMAP) เพื่อให้ได้ระบบการรู้จำตัวอักษรที่มีประสิทธิภาพและมีความยืดหยุ่นกับข้อมูลแบบไบนารีด้วย

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

1. ศึกษาวิธีการรู้จำตัวอักษรลายมือเขียน โดยใช้โครงข่ายประสาทเทียมแบบซิมพลิไฟด์ฟัซซีอาร์ทแมพเป็นหลักในการจำแนกกลุ่ม

2. เพื่อพัฒนาเทคนิคใหม่ในการรู้จำตัวอักษรลายมือเขียน โดยไม่มีการดึงคุณลักษณะเด่นของภาพ

3. เพื่อเป็นแนวทางสำหรับการศึกษาและพัฒนาความรู้ทางด้านปัญญาประดิษฐ์และสาขาอื่นๆ ที่เกี่ยวข้อง

1.3 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย

มีผู้เสนองานวิจัยเกี่ยวกับการรู้จำตัวอักษรไว้มากมาย ซึ่งส่วนใหญ่เป็นงานวิจัยที่เน้นการนำเสนอเทคนิคและวิธีการในการดึงคุณลักษณะเด่นของตัวอักษรเพื่อนำไปใช้เป็นข้อมูลในขั้นตอนการรู้จำ ตัวอย่างเทคนิคที่นิยมใช้ เช่น การพิจารณาจากจำนวนรูปและจุดปลายของตัวอักษร การหาทิศทางของภาพโครงร่างตัวอักษร การหาลำดับการเปลี่ยนแนวเส้น (Stroke Changing Sequence) เป็นต้น ซึ่งแต่ละวิธีให้ผลความถูกต้องในการรู้จำที่สูง แต่ยังคงมีข้อจำกัดสำหรับการเขียนบางรูปแบบ เช่น การเขียนที่ไม่มีการขมวดหัวของตัวอักษร การเขียนที่เกิดรูปภายในตัวอักษรที่ไม่ใช่หัวหรือรูปพื้นฐานของตัวอักษร เป็นต้น ดังนั้นงานวิจัยนี้จึงมุ่งเน้นที่จะแก้ปัญหาข้อจำกัดดังกล่าวโดยการพัฒนาเทคนิคในการรู้จำตัวอักษรลายมือเขียนภาษาไทยโดยไม่มีการคุณลักษณะเด่นของตัวอักษร เพื่อให้ได้วิธีการใหม่ที่สามารถรองรับการเขียนที่หลากหลายรูปแบบมากขึ้นและมีความยืดหยุ่นในการทำงานกับชุดข้อมูลอื่นๆ ด้วยเช่น ข้อมูลภาพใบหน้าคน เป็นต้น

1.4 ขอบเขตของการวิจัย

1. งานวิจัยนี้นำเสนอการรู้จำตัวอักษรลายมือเขียนภาษาไทยโดยไม่มีการดึงคุณลักษณะเด่นของภาพ
2. งานวิจัยนี้เป็น การรู้จำตัวอักษรลายมือเขียนแบบออฟไลน์ (off-line)
3. งานวิจัยนี้รองรับข้อมูลภาพลายมือเขียนที่เป็นภาพแบบไบนารี

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. มีความรู้ความเข้าใจการทำงานของโครงข่ายประสาทเทียมแบบซิมพลิไฟด์พีชชีอาร์ทแมพ
2. ได้องค์ความรู้ใหม่สำหรับงานวิจัยด้านการรู้จำตัวอักษรภาษาไทย
3. เป็นพื้นฐานสำหรับการพัฒนาการรู้จำตัวอักษรภาษาไทยให้เหมาะกับการใช้งานจริง

1.6 ขั้นตอนของการศึกษา

วิทยานิพนธ์ฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

บทที่ 1 กล่าวถึงความเป็นมาและความสำคัญของปัญหา ความมุ่งหมายและวัตถุประสงค์ในการศึกษา ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย ขอบเขตของงานวิจัย ประโยชน์ที่คาดว่าจะได้รับ และขั้นตอนการศึกษางานวิจัยนี้

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในการวิจัย ซึ่งประกอบด้วยสถาปัตยกรรมและขั้นตอนการทำงานของโครงข่ายประสาทเทียมซิมพลิไฟด์พีซซีอาร์ทแมพ เทคนิคการหาโครงร่างของตัวอักษร และบทความงานวิจัยต่างๆ ที่เกี่ยวข้อง

บทที่ 3 กล่าวถึงการประยุกต์ใช้เมตริกซ์ค่าน้ำหนักร่วมกับแบบจำลองโครงข่ายประสาทเทียมแบบลำดับชั้นสำหรับการรู้จำตัวอักษรลายมือเขียนภาษาไทย โดยเริ่มต้นจากขั้นตอนการประมวลผลภาพเบื้องต้น การเตรียมเวกเตอร์อินพุต สถาปัตยกรรมโครงข่ายประสาทเทียมแบบลำดับชั้นที่นำเสนอ กระบวนการเรียนรู้และวิธีการวัดระยะทางวิธีใหม่ที่ใช้เมตริกซ์ค่าน้ำหนัก

บทที่ 4 กล่าวถึงการทดสอบวิธีการประยุกต์ใช้เมตริกซ์ค่าน้ำหนักร่วมกับแบบจำลองโครงข่ายประสาทเทียมแบบลำดับชั้นสำหรับการรู้จำตัวอักษรลายมือเขียนภาษาไทยกับชุดข้อมูลตัวอักษรลายมือเขียนของศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (National Electronics and Computer Technology Center : NECTEC) การออกแบบสถาปัตยกรรมโครงข่ายประสาทเทียมแบบลำดับชั้นที่ใช้ในการทดลอง การกำหนดพารามิเตอร์ที่ใช้ในการเรียนรู้ และผลการทดลอง

บทที่ 5 เป็นบทสรุปผลการวิจัยและข้อเสนอแนะ

บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

ในบทนี้จะแบ่งเนื้อหาออกเป็น 3 ส่วนคือ ส่วนแรกจะกล่าวถึงบทนำ ส่วนที่ 2 อธิบายถึงทฤษฎีพื้นฐานต่างๆ ที่เกี่ยวข้องกับงานวิจัยนี้คือ โครงข่ายประสาทเทียมแบบซิมพลิไฟด์พีชชีอาร์ทแมพและเทคนิคการหาโครงร่าง และส่วนที่ 3 เป็นรายละเอียดของบทความงานวิจัยต่างๆ ที่เกี่ยวข้อง

2.1 บทนำ

งานวิจัยทางการรู้จำตัวอักษรลายมือเขียนสามารถแบ่งออกเป็น 2 ประเภทใหญ่ๆ คือ การวิเคราะห์องค์ประกอบรวม (global analysis) และการวิเคราะห์ลักษณะโครงสร้าง (structural analysis) ลักษณะการวิเคราะห์องค์ประกอบรวมจะทำการประมวลผลโดยตรงกับข้อมูลภาพแบบจุดต่อจุด (pixel by pixel) เช่นถ้าเป็นภาพแบบไบนารีข้อมูลอินพุตก็คือค่า 0 และ 1 เป็นต้น จากนั้นจะนำค่าที่ได้การวิเคราะห์ทางหรือการหาค่าความเหมือนหรือค่าความต่างของภาพตัวอักษร 2 รูปใดๆ เป็นกลไกในกระบวนการเรียนรู้และการจำแนก ซึ่งงานวิจัยทางด้านนี้จะเน้นการนำเสนอวิธีการทางสถิติในการวิเคราะห์ทางระหว่างรูปภาพใดๆ

วิธีการวิเคราะห์ลักษณะ โครงสร้างจะเน้นการนำเสนอเทคนิคการดึงคุณลักษณะเด่นออกจากภาพตัวอักษรลายมือเขียน ตัวอย่างคุณลักษณะเด่นที่นิยมนำมาใช้เช่น จำนวนรูป จุดปลาย จุดตัด ทิศทางของแนวเส้น เป็นต้น ดังนั้นข้อมูลที่ใช้ในการเรียนรู้จึงเป็นเวกเตอร์อินพุตของคุณลักษณะเด่นของตัวอักษร ซึ่งแนวคิดนี้จะตั้งอยู่บนสมมติฐานที่ว่าตัวอักษรกลุ่มเดียวกันจะให้ค่าคุณลักษณะเด่นที่คล้ายคลึงกัน และตัวอักษรต่างกลุ่มจะมีคุณลักษณะเฉพาะที่แตกต่างออกไป จึงทำให้สามารถจำแนกกลุ่มข้อมูลได้ถูกต้อง

2.2 ทฤษฎีที่เกี่ยวข้อง

2.2.1 โครงข่ายประสาทเทียมแบบซิมพลิไฟด์พีชชีอาร์ทแมพ

โครงข่ายประสาทเทียมแบบซิมพลิไฟด์พีชชีอาร์ทแมพ [2-3] มีวิธีการวัดค่าความเหมือนโดยใช้ตัวดำเนินการพีชชีลอจิก AND ซึ่งตัวดำเนินการนี้จะต่างจากตัวดำเนินการลอจิก AND กล่าวคือตัวดำเนินการพีชชี AND จะเปรียบเทียบค่าที่น้อยกว่าระหว่างค่าข้อมูลสองค่าใดๆ

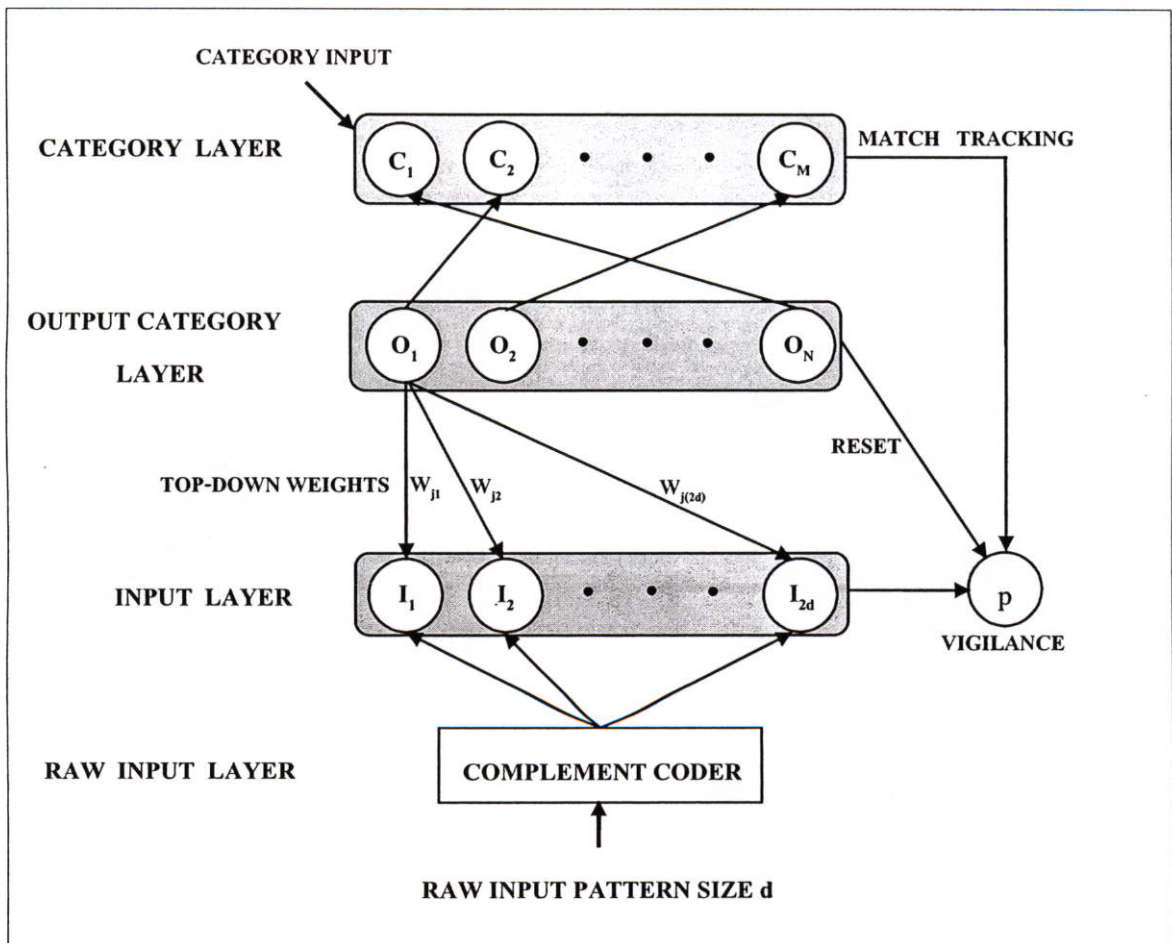
$$(C \wedge D)_i = \min(C_i, D_i) \quad (2.1)$$

และสิ่งที่น่าสนใจสำหรับโครงข่ายนี้คือสามารถสร้างโหนดเอาต์พุตได้มากกว่าหนึ่งโหนดสำหรับคลาสใดๆ หนึ่งคลาส ซึ่งจะมีความเหมาะสมกับการใช้จริงและมีความยืดหยุ่นมาก

ขึ้นถ้าเปรียบเทียบกับโครงข่ายประสาทเทียมแบบแบคพรอพพาเกชัน (Back-propagation Neural Network) ที่มีการสร้างโหนดเอาต์พุตได้เพียงโหนดเดียวสำหรับหนึ่งคลาสใดๆ

สถาปัตยกรรมของโครงข่ายนี้จะแบ่งเป็น 3 ระดับชั้นประกอบด้วยชั้นอินพุต ชั้นเอาต์พุต และชั้นคลาส โดยชั้นอินพุตและชั้นเอาต์พุตจะเชื่อมต่อกันด้วยค่าน้ำหนัก w_{ji} ซึ่งแทนค่าน้ำหนักที่เชื่อมจากโหนดที่ j ในชั้นเอาต์พุตมายังโหนดที่ i ของชั้นอินพุต ส่วนชั้นคลาสจะทำหน้าที่ในการรับเวกเตอร์เป้าหมายเข้าสู่โครงข่าย ดังรูปที่ 2.1 นอกจากนี้ยังมีกลไกที่มีบทบาทสำคัญที่ทำให้เกิดกระบวนการเรียนรู้คือ ค่าวิจิเลนซ์ (vigilance) แทนด้วย ρ ซึ่งเป็นค่าตัวเลขที่กำหนดถึงระดับความเหมือนของข้อมูลที่เรียนรู้ที่จะยอมรับให้จัดอยู่ในคลาสที่พิจารณาได้หรือไม่ ปกติจะกำหนดค่าให้อยู่ในช่วง 0 ถึง 1

ในการกำหนดค่าวิจิเลนซ์นั้นจะมีผลโดยตรงต่อจำนวนโหนดเอาต์พุต นั่นคือถ้ากำหนดให้ค่าวิจิเลนซ์มีค่าเข้าใกล้หนึ่งแล้ว แต่ละเวกเตอร์อินพุตที่เข้าสู่กระบวนการเรียนรู้จะถูกจัดให้อยู่ในกลุ่มใดก็ตามจะต้องมีความคล้ายคลึงกันมาก ไม่เช่นนั้นจะมีการสร้างโหนดเอาต์พุตโหนดใหม่ ซึ่งมีผลทำให้ชั้นเอาต์พุตมีจำนวนโหนดสูงกว่าการกำหนดค่าวิจิเลนซ์ที่เข้าใกล้ศูนย์ และถ้าค่าวิจิเลนซ์ที่กำหนดไว้เริ่มต้นนี้ไม่สามารถจำแนกกลุ่มได้ถูกต้องจะมีการปรับค่าวิจิเลนซ์เพิ่มขึ้นเพื่อให้เกิดการเรียนรู้ที่ถูกต้องต่อไป



รูปที่ 2.1 สถาปัตยกรรมของโครงข่ายประสาทเทียมซิมพลิไฟด์ฟัซซีอาร์ทแมพ

1. การเตรียมเวกเตอร์อินพุต

กระบวนการในการเตรียมข้อมูลของโครงข่ายนี้เรียกว่า Complement coding เป็นขั้นตอนในการนอร์มอลไลซ์ข้อมูลให้มีขนาดเป็น 2 เท่าของขนาดข้อมูลเดิม นั่นคือ ถ้าเวกเตอร์ข้อมูลภาพ $a = (a_1, a_2, \dots, a_d)$ ซึ่งมีขนาด d แล้วจะหาเวกเตอร์คอมพลีเมนต์ a' ได้จาก

$$a' = 1 - a \quad (2.2)$$

และสามารถหาเวกเตอร์อินพุต I ได้ตามสมการที่ 2.3 ซึ่งเวกเตอร์ที่ได้นี้จะมีความเป็นสองเท่าของเวกเตอร์ข้อมูลภาพ เช่น ถ้า $a = (0.3, 0.5, 0.7)$ แล้วจะได้ $a' = (0.7, 0.5, 0.3)$ ดังนั้น $I = (0.3, 0.5, 0.7, 0.7, 0.5, 0.3)$ เป็นต้น

$$I = (a, a') = (a_1, a_2, \dots, a_d, a'_1, a'_2, \dots, a'_d) \quad (2.3)$$

2. กระบวนการเรียนรู้

เริ่มต้นกระบวนการเรียนรู้โดยการนำเวกเตอร์อินพุตที่ได้จากขั้นตอนก่อนหน้านี้อะและเวกเตอร์เป้าหมายเข้าสู่โครงข่ายผ่านชั้นอินพุตและชั้นคลาสตามลำดับ แล้วนำมาประมวลผลตามขั้นตอนดังนี้

ทำการตรวจสอบว่าเคยมีการสร้าง โหนดเอาต์พุตที่มีคลาสเดียวกับเวกเตอร์อินพุตที่กำลังพิจารณาอยู่ก่อนหรือไม่ ถ้าไม่เคยมีคลาสนี้มาก่อน ให้สร้าง โหนดเอาต์พุตของคลาสนี้และทำการกำหนดค่าน้ำหนักที่เชื่อมจากโหนดเอาต์พุตใหม่นี้ไปยังทุกๆ โหนดในชั้นอินพุตด้วยค่าเวกเตอร์อินพุต

ถ้าเคยมีการสร้าง โหนดเอาต์พุตที่มีคลาสเดียวกับเวกเตอร์อินพุตที่กำลังพิจารณาอยู่ก่อนแล้ว ให้ทำการคำนวณค่าฟังก์ชันการกระตุ้น (activation function) เปรียบเทียบระหว่างเวกเตอร์อินพุต I กับเวกเตอร์ค่าน้ำหนักของ โหนดที่ j ของชั้นเอาต์พุต

$$T_j = \frac{|I \wedge w_j|}{\alpha + |w_j|} \quad (2.4)$$

เมื่อ α คือค่าคงที่ที่ค่าน้อยๆ ค่าหนึ่ง ปกติจะกำหนดไว้เท่ากับ 0.0000001 เมื่อคำนวณค่าฟังก์ชันการกระตุ้นของ โหนดเอาต์พุตครบทุกโหนดแล้ว ให้ทำการหาโหนดที่ชนะ (winning node) โดยพิจารณาจากโหนดที่มีค่าฟังก์ชันการกระตุ้นมากที่สุด แต่ถ้าโหนดเอาต์พุตที่ให้ค่ามากที่สุดมีมากกว่า 1 โหนดให้เลือกโหนดที่มีค่า j น้อยที่สุด

$$\text{Winner} = \max(T_j) \quad (2.5)$$

เมื่อได้โหนดที่ชนะแล้วให้ทำการตรวจสอบว่าเวกเตอร์อินพุตนี้มีความเหมือนมากพอที่จะจัดให้อยู่ในคลาสเดียวกับโหนดที่ชนะหรือไม่ โดยค่าที่ใช้ตรวจสอบนี้เรียกว่าค่าฟังก์ชันแมช (match function)

$$\frac{|I \wedge w_j|}{|I|} \geq p \quad (2.6)$$

ถ้าค่าฟังก์ชันแมชมากกว่าหรือเท่ากับค่าวิจิลแลนซ์แล้วจะทำให้สมการที่ 2.6 เป็นจริง จากนั้นให้พิจารณาต่อไปว่าคลาสของโหนดที่ชนะมีคลาสเดียวกันกับเวกเตอร์อินพุตหรือไม่ ถ้าคลาสทั้งสองตรงกันแสดงว่าเกิดเรโซแนนซ์ นั่นคือเวกเตอร์อินพุตมีความเหมือนอยู่ในระดับที่ยอมรับได้และยังสามารถให้จัดอยู่ในกลุ่มเดียวกับโหนดที่ชนะได้ด้วย จากนั้นจะปรับค่าน้ำหนักของโหนดที่ชนะ

$$w_j^{new} = \beta(I \wedge w_j^{old}) + (1 - \beta)w_j^{old} \quad (2.7)$$

เมื่อ β คืออัตราการเรียนรู้มีค่าระหว่าง 0 ถึง 1 แต่ถ้ากำหนดให้ β เท่ากับ 1 แล้วจะได้ฟังก์ชันการปรับค่าน้ำหนักดังนี้

$$w_j^{new} = (I \wedge w_j^{old}) \quad (2.8)$$

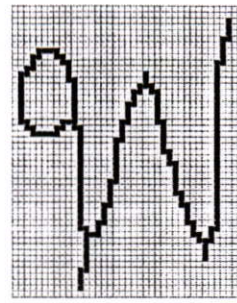
กรณีที่ค่าฟังก์ชันแมชมากกว่าหรือเท่ากับค่าวิจิลแลนซ์แต่คลาสของโหนดที่ชนะไม่ตรงกับคลาสเวกเตอร์อินพุต (category mismatch) จะต้องปรับค่าวิจิลแลนซ์ให้เท่ากับค่าฟังก์ชันแมชที่บวกเพิ่มด้วยค่าจำนวนบวกลบเล็กๆ ด้วย จากนั้นให้พิจารณาหาโหนดที่ชนะ โหนดถัดไปหรือกรณีที่ค่าฟังก์ชันแมชน้อยกว่าค่าวิจิลแลนซ์ ก็ให้พิจารณาหาโหนดที่ชนะ โหนดถัดไปเช่นกัน จนกว่าจะสามารถจำแนกเวกเตอร์อินพุตนี้เข้ากลุ่มโหนดเอาต์พุตใดๆ ได้ แต่ถ้าพิจารณาจนครบทุกโหนดเอาต์พุตแล้วแต่ไม่สามารถจำแนกเข้ากลุ่มใดได้เลย ให้ทำการสร้างโหนดเอาต์พุตโหนดใหม่และกำหนดค่าน้ำหนักที่เชื่อมจากโหนดเอาต์พุตใหม่นี้ยังทุกๆ โหนดในชั้นอินพุตด้วยค่าเวกเตอร์อินพุต จากนั้นให้รับเวกเตอร์อินพุตถัดไปเข้ามาเรียนรู้ในโครงข่ายต่อไป

2.2.2 เทคนิคการหาโครงร่าง

การหาโครงร่างของตัวอักษรคือการสกัดเอาจุดภาพที่เป็นขอบของตัวอักษรออกไป จนกระทั่งตัวอักษรมีความหนาเพียง 1 พิกเซลและเส้นที่ได้จะเป็นโครงกลางของตัวอักษร [4] ดังรูปที่ 2.2



(a) ภาพก่อนการประมวลผล



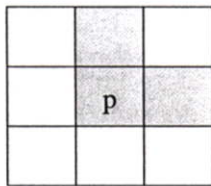
(b) ภาพโครงร่างตัวอักษร

รูปที่ 2.2 ภาพตัวอักษรก่อนและหลังผ่านกระบวนการหาโครงร่าง

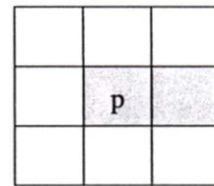
สิ่งสำคัญที่ต้องคำนึงถึงในการหาโครงร่างคือ

1) การรักษาความต่อเนื่องของเส้น ในการพิจารณาลบจุดใดจุดหนึ่งต้องคำนึงไว้ว่าจะต้องไม่ทำให้รูปร่างที่เหลือเกิดการขาดตอน เช่นจะไม่ลบจุดภาพที่เป็นจุดเชื่อม (connected pixel) เป็นต้น ตัวอย่างจุดภาพที่เป็นจุดเชื่อม ดังรูปที่ 2.3 (a)

2) ต้องไม่ทำให้รูปร่างของเส้นสั้นลง กล่าวคือต้องไม่ลบจุดภาพที่เป็นจุดสิ้นสุดของเส้น (end point) ตัวอย่างดังรูปที่ 2.3 (b)



(a) จุดภาพที่เป็นจุดเชื่อม



(b) จุดภาพที่เป็นจุดสิ้นสุด

รูปที่ 2.3 ตัวอย่างจุดภาพที่เป็นจุดเชื่อมและจุดสิ้นสุด

อัลกอริทึมของฮิลด์ทช์ (Hilditch's algorithm) [4-5] เป็นเทคนิคการหาโครงร่างที่งานวิจัยนี้เลือกใช้ ซึ่งขั้นตอนในการตัดสินใจจะลบจุดภาพค่าใดๆ หรือจะเก็บไว้เป็นส่วนหนึ่งของโครงร่างนั้นจะพิจารณาภายในหน้าต่างขนาด 3x3

x_8	x_1	x_2
x_7	p	x_3
x_6	x_5	x_4

รูปที่ 2.4 จุดภาพ p ที่จะพิจารณาลบภายในหน้าต่างขนาด 3x3

ภาพที่นำมาหาโครงร่างเป็นภาพแบบไบนารี ซึ่งมีค่าของจุดภาพที่เป็นไปได้มีเพียง 2 ค่า คือ 0 และ 1 ในที่นี้จะกำหนดให้ 1 แทนค่าจุดภาพดำและ 0 แทนค่าจุดภาพขาว

กฎ R_1 และ R_2 ที่ใช้ในการพิจารณาจุดภาพดำ ได้มีการนิยามฟังก์ชันไว้ 2 ฟังก์ชันคือ $B(p)$ เป็นฟังก์ชันการนับจำนวนจุดภาพดำของจุดภาพเพื่อนบ้านภายในหน้าต่างขนาด 3×3 จำนวนได้จาก

$$B(p) = \sum_{i=1}^8 x_i \quad (2.9)$$

และ $X_H(p)$ เป็นฟังก์ชันการนับจำนวนครั้งการเปลี่ยนจากจุดภาพขาวไปเป็นจุดภาพดำ เมื่อกำหนดให้ $x_0 = x_1$ แล้วจะได้ว่า

$$X_H(p) = \sum_{i=1}^8 b_i \quad (2.10)$$

โดยที่

$$b_i = \begin{cases} 1 & , (x_i = 0) \wedge (x_{i+1} = 1) \\ 0 & , \text{otherwise} \end{cases} \quad (2.11)$$

ตัวอย่างการคำนวณค่า $B(p)$ และ $X_H(p)$ เช่น

0	1	1
0	1	0
0	0	0

$$B(p) = 2$$

$$X_H(p) = 0 + 0 + 0 + 0 + 0 + 0 + 0 + 1$$

$$= 1$$

0	1	1
0	1	0
1	0	0

$$B(p) = 3$$

$$X_H(p) = 0 + 0 + 0 + 0 + 1 + 0 + 0 + 1$$

$$= 2$$

อัลกอริทึมนี้จะแบ่งการพิจารณาจุดภาพดำ เป็นรอบการทำงานย่อยๆ วนซ้ำพิจารณาตามกฎ R_1 และ R_2 ไปเรื่อยๆ จนกว่าจะมีรอบการทำงานที่ไม่มีการลบจุดภาพดำจึงจะหยุดการทำงาน

$$R_1 : (2 \leq B(p) \leq 6) \wedge (X_H(p) = 1) \wedge (x_1 \cdot x_3 \cdot x_5 = 0) \wedge (x_3 \cdot x_5 \cdot x_7 = 0) \quad (2.12)$$

$$R_2 : (2 \leq B(p) \leq 6) \wedge (X_H(p) = 1) \wedge (x_3 \cdot x_1 \cdot x_7 = 0) \wedge (x_1 \cdot x_7 \cdot x_5 = 0) \quad (2.13)$$

เงื่อนไขที่ 1 ($2 \leq B(p) \leq 6$)

เงื่อนไขนี้จะแบ่งเป็น 2 เงื่อนไขย่อยคือ $B(p) \geq 2$ เป็นการกำหนดจำนวนจุดภาพเพื่อนบ้านที่เป็นจุดภาพค่าต้องมีมากกว่า 1 จุดภาพเพื่อให้แน่ใจว่าเราจะไม่ลบจุดสิ้นสุดของโครงร่าง เพราะจะทำให้รูปทรงของตัวอักษรสั้นลงและจะไม่ลบจุดภาพเดี่ยว (isolated pixel) เพราะจุดภาพนี้เป็นจุดภาพรบกวน (noise pixel) ก็ควรถูกลบในขั้นตอนการกำจัดสัญญาณรบกวน

เงื่อนไขย่อยที่สองคือ $B(p) \leq 6$ เพื่อพิจารณาลบเฉพาะจุดภาพที่เป็นจุดขอบภาพ (boundary pixel) ของตัวอักษรเท่านั้น เนื่องจากการลบจุด p ที่มีจำนวนจุดภาพเพื่อนบ้านมากกว่า 6 จุดภาพจะทำให้เกิดการกัดเซาะรูปทรงตัวอักษร

0	0	0
0	1	0
0	0	0

$$B(p) = 0$$

0	1	0
0	1	0
0	0	0

$$B(p) = 1$$

1	1	1
1	1	0
1	1	1

$$B(p) = 7$$

รูปที่ 2.5 ตัวอย่างจุดภาพ p ที่ไม่สามารถลบได้ตามเงื่อนไข ($2 \leq B(p) \leq 6$)

เงื่อนไขที่ 2 ($X_H(p) = 1$)

เป็นเงื่อนไขในการตรวจสอบความต่อเนื่องของเส้นโครงร่าง เพื่อให้แน่ใจว่าจุดภาพ p ที่จะลบต้องไม่ทำให้รูปร่างที่เหลือขาดตอน

1	0	0
0	1	0
1	0	0

$$X_H(p) = 2$$

0	1	0
0	1	1
1	0	0

$$X_H(p) = 3$$

0	1	0
1	1	1
0	1	0

$$X_H(p) = 4$$

รูปที่ 2.6 ตัวอย่างจุดภาพ p ที่ไม่สามารถลบได้ตามเงื่อนไข ($X_H(p) = 1$)

เงื่อนไขที่ 3 ($(x_1 \cdot x_3 \cdot x_5 = 0) \wedge (x_3 \cdot x_5 \cdot x_7 = 0)$) ของกฎ R_1

เงื่อนไขนี้จะพิจารณาจุด p ที่เป็นจุดขอบด้านทิศตะวันออกเมื่อ x_3 เป็นจุดภาพขาวหรือจุด p ที่เป็นจุดขอบด้านทิศใต้เมื่อ x_5 เป็นจุดภาพขาวหรือจุด p ที่เป็นจุดมุมด้านทิศตะวันตกเฉียงเหนือเมื่อ x_1 และ x_7 เป็นจุดภาพขาว

1	1	0
1	X	0
1	1	0

(a) จุดขอบด้านทิศตะวันออก

1	1	1
1	X	1
0	0	0

(b) จุดขอบด้านทิศใต้

0	0	0
0	X	1
0	1	1

(c) จุดมุมด้านทิศตะวันตกเฉียงเหนือ

รูปที่ 2.7 ตัวอย่างจุดภาพ p ที่สามารถลบได้ตามเงื่อนไข $(x_1 \cdot x_3 \cdot x_5 = 0) \wedge (x_3 \cdot x_5 \cdot x_7 = 0)$

เงื่อนไขที่ 4 $(x_3 \cdot x_1 \cdot x_7 = 0) \wedge (x_1 \cdot x_7 \cdot x_5 = 0)$ ของกฎ R_2

เงื่อนไขนี้จะพิจารณาจุด p ที่เป็นจุดขอบด้านทิศตะวันตกเมื่อ x_7 เป็นจุดภาพขาวหรือจุด p ที่เป็นจุดขอบด้านทิศเหนือเมื่อ x_1 เป็นจุดภาพขาวหรือจุด p ที่เป็นจุดมุมด้านทิศตะวันออกเฉียงใต้เมื่อ x_3 และ x_5 เป็นจุดภาพขาว

0	1	1
0	X	1
0	1	1

(a) จุดขอบด้านทิศตะวันตก

0	0	0
1	X	1
1	1	1

(b) จุดขอบด้านทิศใต้

1	1	0
1	X	0
0	0	0

(c) จุดมุมด้านทิศตะวันออกเฉียงใต้

รูปที่ 2.8 ตัวอย่างจุดภาพ p ที่สามารถลบได้ตามเงื่อนไข $(x_3 \cdot x_1 \cdot x_7 = 0) \wedge (x_1 \cdot x_7 \cdot x_5 = 0)$

อัลกอริทึมนี้เป็นอัลกอริทึมที่ง่ายและมีประสิทธิภาพ แต่อย่างไรก็ตามยังมีจุดด้อยที่อาจทำงานได้ไม่สมบูรณ์มากนักกับบางรูปแบบเช่น ภาพเส้นทแยงมุมที่มีความกว้าง 2 พิกเซล ภาพสี่เหลี่ยมจัตุรัสที่มีขนาด 2×2 พิกเซล เป็นต้น แต่สามารถแก้ปัญหาได้ด้วยการแทนเงื่อนไขแรกของทั้งสองกฎด้วยเงื่อนไข $(3 \leq B(p) \leq 6)$

2.3 บทความงานวิจัยที่เกี่ยวข้อง

2.3.1 Human Face Recognition using A Spatially Weighted Modified Hausdorff Distance เขียนโดย Kwan-Ho, Baofeng Guo, Kin-Man Lam และ Wan-Chi Siu [6]

บทความนี้นำเสนอเทคนิคการวัดค่าระยะทางแบบใหม่ 2 วิธีที่ใช้สำหรับการรู้จำภาพใบหน้า โดยมีพื้นฐานมาจากการวัดระยะทางแบบฮูสโดรอฟดิสแตรนซ์ แต่ได้ปรับปรุงเพิ่มเติมโดยให้มีการกำหนดค่าฟังก์ชันน้ำหนักขึ้นมา เนื่องจากเล็งเห็นว่าตำแหน่งพื้นที่ต่างๆ บนภาพใบหน้าที่มีความสำคัญในการจำแนกไม่เท่ากัน โดยเฉพาะตำแหน่งพื้นที่ของตาทั้งสองข้างและปากมี

ความสำคัญมากในการจำแนกกลุ่ม ดังนั้นจึงได้มีการกำหนดค่าฟังก์ชันน้ำหนักให้กับแต่ละพื้นที่บนภาพใบหน้า

เทคนิคแรกที่น่าเสนอคือการวัดระยะทางแบบ SWHD (spatially weighted hausdorff distance) เป็นวิธีการปรับปรุงการวัดระยะทางแบบ MHD (modified hausdorff distance) ให้มีการกำหนดค่าฟังก์ชันน้ำหนักตามตำแหน่งพื้นที่และเทคนิคที่สองคือการวัดระยะทางแบบ SW2HD (spatially weight 'doubly' hausdorff distance) เกิดจากการผสมผสานการวัดระยะทางแบบเทคนิคแรกที่น่าเสนอเข้ากับการระยะทางแบบ M2HD ('doubly' modified hausdorff distance)

การวัดระยะทางแบบ M2HD เป็นเทคนิคการวัดระยะทางที่พัฒนาต่อเนื่องมาจากการวัดระยะทางแบบอูสตรอฟดิสแทรนด์ กำหนดให้เซต $A = \{a_1, a_2, \dots, a_n\}$ และเซต $B = \{b_1, b_2, \dots, b_n\}$

$$H(A, B) = \max \{h(A, B), h(B, A)\} \quad (2.14)$$

โดยที่

$$h(A, B) = \frac{1}{N_a} \sum_{a \in A} d(a, B) \quad (2.15)$$

และ

$$d(a, B) = \max \left(I \cdot \min_{b \in N_B^a} \|a - b\|, (1 - I)P \right) \quad (2.16)$$

เมื่อ $\| \cdot \|$ คือการหาระยะทางจากจุดหนึ่งไปยังอีกจุดหนึ่ง, N_B^a คือจุดภาพเพื่อนบ้านของ a ในเซต B , I และ P คือค่าคงที่ค่าหนึ่ง ปกติค่า I จะกำหนดให้เท่ากับ 1 ถ้ามีจุดภาพค่า $b \in N_B^a$ และจะกำหนดให้เท่ากับ 0 ถ้าไม่มีจุดภาพค่า $b \in N_B^a$ และ N_a คือจำนวนจุดภาพค่าของเซต A

การวัดระยะทางแบบ SWHD เป็นเทคนิคการวัดระยะทางที่มีการกำหนดค่าน้ำหนักของแต่ละตำแหน่งบนภาพที่แตกต่างกัน เพื่อแสดงให้เห็นว่าแต่ละตำแหน่งบนใบหน้ามีความสำคัญไม่เท่ากัน การกำหนดค่าน้ำหนักจะเป็นไปตามฟังก์ชันน้ำหนัก กำหนดให้เซต $A = \{a_1, a_2, \dots, a_n\}$ และเซต $B = \{b_1, b_2, \dots, b_n\}$

$$H(A, B) = \max \{h_{sw}(A, B), h_{sw}(B, A)\} \quad (2.17)$$

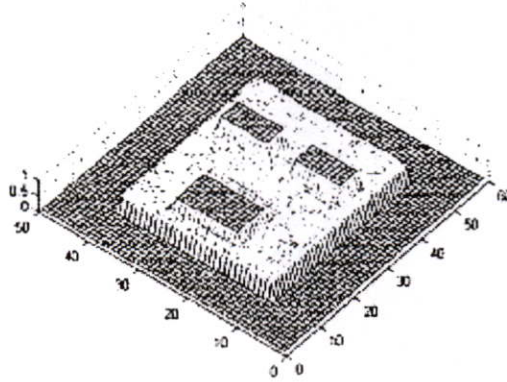
โดยที่

$$h_{sw}(A, B) = \frac{1}{N_a} \sum_{a \in A} w(a) \cdot \min_{b \in B} \|a - b\| \quad (2.18)$$

เมื่อฟังก์ชันน้ำหนัก $w(x)$ พิจารณาดังนี้

$$w(x) = \begin{cases} 1 & , x \in R_{ic} \\ w_u & , x \in R_u \\ w_{om} & , R_{im} \cap R_{gm} \\ w_{nm} & , (R_{im} \cap R_{gm}^c) \cup (R_{im}^c \cap R_{gm}) \\ 0 & , x \in R_b \end{cases} \quad (2.19)$$

- เมื่อ R_{ic} คือพื้นที่ตำแหน่งของตาทั้งสองข้างบนภาพใบหน้า
 R_{im} คือพื้นที่ตำแหน่งของปากบนภาพใบหน้าของอินพุท
 R_{gm} คือพื้นที่ตำแหน่งของปากบนภาพใบหน้าของโมเดล
 R_u คือพื้นที่บนภาพใบหน้าตำแหน่งอื่นๆที่ไม่ใช่ตำแหน่งของตาและปาก
 R_b คือพื้นที่ที่พื้นหลังของภาพใบหน้า
 R^c คือพื้นที่ตำแหน่งที่ไม่ใช่พื้นที่ของ R



รูปที่ 2.9 กราฟสามมิติของฟังก์ชันน้ำหนัก $w(x)$

การวิเคราะห์ทางเทคนิคนี้จะให้ความสำคัญแต่ละตำแหน่งบนใบหน้าแตกต่างกัน ถ้าเป็นพื้นที่ตำแหน่งของตาหรือพื้นที่ตำแหน่งของปากที่เกิดจากการซ้อนทับกันของภาพอินพุทกับภาพโมเดลจะให้ความสำคัญสูง เช่น กำหนดค่าเท่ากับ 1 เป็นต้น แต่ถ้าเป็นพื้นที่ที่ไม่ใช่พื้นที่ซ้อนทับจะให้ความสำคัญลดลง เช่น กำหนดค่าเท่ากับ 0.7 เป็นต้น ส่วนพื้นที่ตำแหน่งอื่นๆที่ไม่ใช่ตำแหน่งของปากและตาหรือเป็นพื้นที่พื้นหลัง จะกำหนดให้เท่ากับ 0.5 และ 0 ตามลำดับ ซึ่งค่าระดับความสำคัญจะกำหนดเป็นค่าคงที่ตายตัว

การวิเคราะห์ทางแบบ M2HD จะไม่มีการพิจารณาความสำคัญของแต่ละตำแหน่งบนภาพใบหน้า ในขณะที่การวิเคราะห์ทางแบบ SWHD จะมีการกำหนดความสำคัญของแต่ละตำแหน่งแตกต่างกันที่พอจะทำให้เห็นภาพใบหน้าแบบคร่าวๆได้ จึงได้มีการพัฒนาต่อเป็นเทคนิควิธีที่สองที่งานวิจัยนี้นำเสนอคือการวิเคราะห์ทางแบบ SW2HD โดยใช้ฟังก์ชันน้ำหนักเหมือนกับวิธีแรก

$$h(A, B) = \frac{1}{N_a} \sum_{a \in A} w(b) \cdot d(a, B) \quad (2.20)$$

โดยที่

$$d(a, B) = \max(I \cdot \min_{b \in N_B} \|a - b\|, (1 - I)P) \quad (2.21)$$

เทคนิคใหม่ที่น่าเสนอในบทความนี้ได้ทดสอบกับข้อมูลภาพของฐานข้อมูล ORL ที่มีภาพใบหน้าจำนวน 400 ภาพ แต่ละภาพถูกบันทึกในสภาวะที่แตกต่างกัน เช่น ช่วงเวลา ปริมาณแสงสว่าง การแสดงอารมณ์ ทิศทางการหันหน้า อัตราส่วนระยะไกลใกล้ เป็นต้น โดยภาพใบหน้า

จะถูกแบ่งเป็น 2 กลุ่มคือภาพที่ใช้สำหรับการเรียนรู้ 160 ภาพและภาพที่ใช้สำหรับการทดสอบ 240 ภาพ

การทดลองจะเปรียบเทียบประสิทธิภาพในการรู้จำกับการวัดระยะทางอีก 2 แบบคือ HD, และ M2HD ผลการทดลองพบว่า การวัดระยะทางแบบ SWHD และ SW2HD มีประสิทธิภาพในการรู้จำดีกว่าแบบอื่นๆ และการวัดระยะทางแบบ SW2HD ให้ผลการทดลองที่ดีที่สุดในทุกๆ ชุด การทดลองที่กำหนดค่าพารามิเตอร์ที่แตกต่างกันและค่าพารามิเตอร์ของ w_u , w_{nm} และ P ที่ทำให้ประสิทธิภาพในการรู้จำที่ดีที่สุดคือ 0.5 , 0.7 และ 4 ตามลำดับ

2.3.2 Spatially eigen-weighted Hausdorff distance for human face recognition เขียนโดย Kwan-Ho, Kin-Man Lam และ Wan-Chi Siu [7]

บทความนี้เป็นงานวิจัยด้านการรู้จำใบหน้าที่ได้นำเสนอเทคนิคใหม่ในการวัดค่าระยะทาง 2 วิธีด้วยกันคือการวัดระยะทางแบบ SEWHD (spatially eigen-weighted hausdorff distance) และการวัดระยะทางแบบ SEW2HD (spatially eigen-weighted doubly hausdorff distance) ซึ่งการวัดระยะทางทั้งสองแบบพัฒนาต่อเนื่องมาจากการวัดระยะทางแบบ SWHD และ SW2HD แต่ได้มีการปรับปรุงวิธีการกำหนดค่าฟังก์ชันน้ำหนักแบบใหม่โดยประยุกต์ใช้ไอเกนเฟซ (eigenface)

การวัดระยะทางแบบ SWHD จะมีการกำหนดค่าน้ำหนักต่างๆ กันตามระดับความสำคัญของแต่ละพื้นที่บนภาพใบหน้า โดยเริ่มจากกำหนดตำแหน่งพื้นที่ที่สัมผัสกับเส้นผิรอบบริเวณตาและปากแล้วให้ค่าน้ำหนักเป็น 1 ส่วนพื้นที่อื่นๆ จะมีค่าน้ำหนักลดลงมา การกำหนดค่าน้ำหนักนี้จะทำให้เห็นโครงร่างคร่าวๆ ของใบหน้า แต่ไม่ได้แสดงโครงร่างที่แท้จริงของใบหน้าได้

การวัดระยะทางแบบ SEWHD ได้มีการปรับปรุงวิธีการกำหนดค่าฟังก์ชันน้ำหนัก ดังนี้ กำหนดให้เซต $A = \{a_1, a_2, \dots, a_n\}$ และเซต $B = \{b_1, b_2, \dots, b_n\}$

$$H(A, B) = \max\{h_{sw}(A, B), h_{sw}(B, A)\} \quad (2.22)$$

โดยที่

$$h_{sw}(A, B) = \frac{1}{N_a} \sum_{N_a} w_c(b) \cdot \min_{b \in B} \|a - b\| \quad (2.23)$$

เมื่อ $\| \cdot \|$ คือการหาระยะทางจากจุดหนึ่งไปยังอีกจุดหนึ่ง, N_a คือจำนวนจุดภาพค่าของเซต A และ $w_c(x)$ คือค่าฟังก์ชันน้ำหนักไอเกน ที่เน้นการให้ความสำคัญแต่ละจุดภาพบนใบหน้าที่แตกต่างกัน ซึ่งค่าฟังก์ชันน้ำหนักไอเกนนี้ถูกสร้างขึ้นมาจากชุดข้อมูลสำหรับการเรียนรู้ ถ้ากำหนดให้ $\Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_k$ คือเซตของภาพใบหน้าที่เป็นข้อมูลสำหรับการเรียนรู้โดยแต่ละภาพจะมีขนาด $1 \times N^2$ มิติ แล้วจะได้เวกเตอร์ค่าเฉลี่ยใบหน้า (average face)

$$\Psi = \frac{1}{M} \sum_{n=1}^M \Gamma_n \quad (2.24)$$

และเวกเตอร์ค่าความแปรปรวน (difference face) ดังนี้

$$\Phi_i = \Gamma_i - \Psi \quad (2.25)$$

จากนั้นสามารถคำนวณหาเมตริกซ์ความสัมพันธ์ (correlation matrix) ได้จาก

$$M = \Phi \cdot \Phi^T \quad (2.26)$$

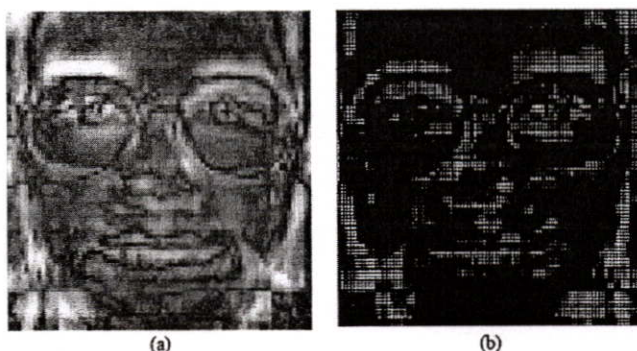
แล้วไอเจนเวกเตอร์ของเมตริกซ์ M นี้จะเป็นตัวแทนแสดงองค์ประกอบสำคัญของภาพใบหน้า นั่นคือตำแหน่งที่มีความสำคัญในการจำแนกกลุ่มสูง ก็จะมีค่าน้ำหนักสูงด้วย เช่น ตำแหน่งของตา ปาก คิ้ว เป็นต้น ถ้า $v(x, y)$ คือเวกเตอร์ไอเจนอันดับหนึ่งแล้ว

$$n_{\text{mask}}(x, y) = \left(\frac{255}{\max(v) - \min(v)} \right) v(x, y) \quad (2.27)$$

โดยที่

$$v(x, y) = |v(x, y)| \quad (2.28)$$

เมื่อ $n_{\text{mask}}(x, y)$ คือฟังก์ชันการนอร์มอลไลซ์เวกเตอร์ไอเจนอันดับหนึ่ง, $\max(v)$ และ $\min(v)$ คือค่าน้ำหนักที่มากที่สุดและน้อยที่สุดของเวกเตอร์ไอเจนอันดับหนึ่ง ตามลำดับ จากนั้นฟังก์ชันน้ำหนักไอเจนที่จะแทนค่าในสมการของวิธีการวัดระยะทางแบบ SEWHD และ SEW2HD ก็คือการนอร์มอลไลซ์ $n_{\text{mask}}(x, y)$ ให้มีค่าอยู่ระหว่าง 0 ถึง 1



รูปที่ 2.10 (a) เวกเตอร์ไอเจนอันดับหนึ่งและ (b) ค่าฟังก์ชันน้ำหนักไอเจน

ภาพใบหน้าที่ใช้ในการทดลองได้จากฐานข้อมูล ORL, Yale และ MIT ซึ่งต้องนำมาประมวลผลภาพเบื้องต้นก่อนโดยการหาขอบภาพระดับเทา จากนั้นให้แปลงเป็นภาพไบนารีโดยวิธีการปรับค่าเทรชโฮลด์ (adaptive thresholding method) เนื่องจากขอบภาพระดับเทาของแต่ละภาพอาจจะมีค่าในช่วงแตกต่างกัน การกำหนดค่าเทรชโฮลด์ตายตัวจะทำให้ได้ภาพไบนารีไม่สมบูรณ์นัก จึงใช้วิธีการปรับค่าเทรชโฮลด์ตามคุณลักษณะของภาพ โดยปกติแล้วคุณลักษณะ

สำคัญของภาพและขอบใบหน้าจะมีค่าระดับเทาต่ำกว่าส่วนอื่นๆ ดังนั้นการปรับค่าเทรสโซดัลคำนวณได้จาก

$$n(x, y) = \frac{E(x, y)}{f(x, y)} \quad (2.29)$$

เมื่อ $E(x, y)$ คือภาพที่ได้จากการหาขอบระดับเทาของภาพใบหน้า $f(x, y)$ จากนั้นให้นำค่าของ $n(x, y)$ มาจัดเรียงลำดับจากน้อยไปมาก นั่นคือการนำค่าระดับเทามาเรียงตามระดับความสำคัญจากมากไปหาระดับความสำคัญน้อย แล้วค่าระดับเทาที่เป็น 18% ของจุดที่มีระดับความสำคัญมากที่สุดของ $n(x, y)$ จะถูกกำหนดให้เป็นค่าเทรสโซดัลของภาพใบหน้านั้นๆ

บทความนี้ได้มีการทดลองเปรียบเทียบเทคนิคการวิเคราะห์ทางในการรู้จำใบหน้าแบบต่างๆ คือ PCA, HD, M2HD, SWHD และ SW2HD กับเทคนิคใหม่ที่น่าสนใจ โดยใช้ชุดข้อมูลทั้งหมด 4 ชุดคือ ORL, Yale, MIT และชุดข้อมูลสุดท้ายได้จากการรวมภาพใบหน้าจากฐานข้อมูลทั้งสาม ผลการทดลองพบว่าเทคนิคการวิเคราะห์ทาง SEWHD และ SEW2HD มีประสิทธิภาพในการรู้จำใบหน้าได้ดีกว่าวิธีอื่นๆ ที่เปรียบเทียบกับกันและการวิเคราะห์ทางแบบ SEW2HD มีประสิทธิภาพดีที่สุดในทุกๆ ชุดข้อมูลที่ทดลอง และเวกเตอร์ไอเกนอันดับหนึ่งให้ผลการรู้จำที่ดีที่สุดเมื่อเทียบกับเวกเตอร์ไอเกนอันดับอื่นๆ

2.3.3 A Novel Feature Extraction Technique for the Recognition of Segmented Handwritten Character เขียนโดย M. Blumenstein, B. Verma และ H. Basli [8]

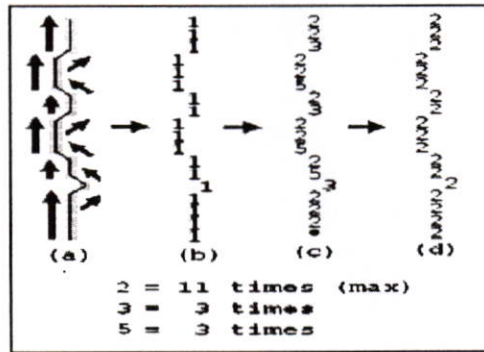
บทความนี้ได้แนะนำวิธีการดึงคุณลักษณะเด่นของภาพลายมือเขียนเพื่อใช้ในการรู้จำตัวอักษร ขั้นตอนแรกคือการประมวลผลภาพเบื้องต้น เริ่มต้นจากการหาโครงร่างแล้วนำผลลัพธ์ที่ได้ไปหาขอบของตัวอักษรเพื่อนำไปเป็นอินพุตสู่ขั้นตอนการดึงคุณลักษณะเด่นต่อไป ซึ่งวิธีการดึงคุณลักษณะเด่นที่น่าสนใจคือการดึงคุณลักษณะทิศทางของภาพโครงร่างตัวอักษร (direction feature)

ในการดึงคุณลักษณะทิศทางของภาพโครงร่างตัวอักษร ก่อนที่หาทิศทางของทั้งตัวอักษรได้จะต้องทำการแยกเส้นแต่ละเส้นที่เป็นองค์ประกอบออกมาก่อน จากนั้นก็หาทิศทางของแต่ละเส้นดังรูปที่ 2.11 แล้วจึงนำมารวมเป็น โครงร่างทิศทางของตัวอักษร สามารถสรุปเป็นขั้นตอนได้ดังนี้

1. หาจุดเริ่มต้นและตำแหน่งจุดตัด จุดภาพคำด้านล่างซ้ายสุดจะถือเป็นจุดเริ่มต้น เพราะการเขียนตัวอักษรภาษาอังกฤษส่วนใหญ่จะเริ่มจากด้านล่างซ้าย และการหาจุดตัดภายในโครงร่างตัวอักษรให้พิจารณาจากจำนวนจุดภาพคำที่เป็นจุดภาพเพื่อนบ้าน ถ้ามีมากกว่า 2 จุดแสดงว่าเป็นจุดตัดให้แทนค่าด้วย 9
2. แยกเส้นออกจากโครงร่าง พิจารณาแยกเส้นแต่ละเส้นจากข้อมูลจุดเริ่มต้น จุดตัดและการเปลี่ยนทิศทางของจุด

3. กำหนดหมายเลขทิศทาง เมื่อแยกแต่ละเส้นได้แล้วจะต้องมีการระบุทิศทางของจุดภาพคำทั้งหมดบนเส้น โดยมีกำหนดไว้ 4 ทิศทางคือทิศทางแนวตั้ง, ทิศทางเฉียงด้านซ้าย, ทิศทางเฉียงด้านขวาและทิศทางแนวนอนซึ่งแทนด้วยค่าคงที่คือ 2, 3, 4 และ 5 ตามลำดับ

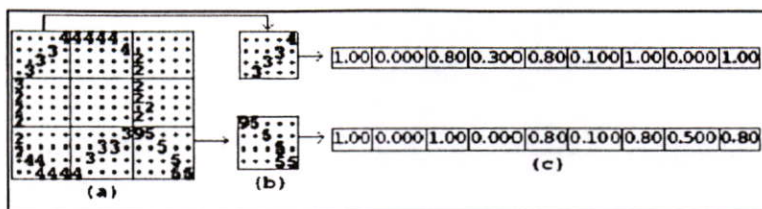
4. ทำการนอร์มอลไลซ์ทิศทางของแต่ละเส้น เนื่องจากเส้นที่แยกจากโครงร่างอาจจะประกอบด้วยหลายทิศทางภายในเส้นเดียวกัน ดังนั้นในการตัดสินใจว่าแต่ละเส้นมีทิศทางใดนั้นจะพิจารณาจากความถี่ของหมายเลขทิศทาง ถ้าหมายเลขทิศทางใดมีความถี่มากที่สุดก็จะถูกกำหนดให้เป็นทิศทางของเส้นนั้น แล้วทำการเปลี่ยนหมายเลขทิศทางของทั้งเส้นตามทิศทางที่ได้



รูปที่ 2.11 (a) โครงร่างของเส้น, (b) ค่าพิชิตของเส้น, (c) การระบุทิศทางของแต่ละจุด, (d) เส้นที่ผ่านการนอร์มอลไลซ์ทิศทาง

เมื่อได้ภาพ โครงร่างที่ผ่านการนอร์มอลไลซ์ทิศทางแล้ว ให้นำมาแบ่งเป็น 9 บล็อกขนาดเท่าๆ กัน ดังรูปที่ 2.12 (a) จากนั้นจะทำการดึงคุณลักษณะเด่นของแต่ละบล็อกโดยหนึ่งบล็อกจะมีคุณลักษณะเด่น 9 ค่าด้วยกันคือ

1. ค่าจำนวนของเส้นที่มีทิศทางแนวตั้ง
2. ค่าความยาวของเส้นที่มีทิศทางแนวตั้ง
3. ค่าจำนวนของเส้นที่มีทิศทางเฉียงด้านขวา
4. ค่าความยาวของเส้นที่มีทิศทางเฉียงด้านขวา
5. ค่าจำนวนของเส้นที่มีทิศทางเฉียงด้านซ้าย
6. ค่าความยาวของเส้นที่มีทิศทางเฉียงด้านซ้าย
7. ค่าจำนวนของเส้นที่มีทิศทางแนวนอน
8. ค่าความยาวของเส้นที่มีทิศทางแนวนอน
9. จำนวนจุดตัด



รูปที่ 2.12 (a) โครงร่างที่ผ่านการนอร์มอลไลซ์ทิศทาง, (b) บล็อกของภาพ,
(c) เวกเตอร์คุณลักษณะสำคัญของบล็อก

จากรูปที่ 2.12 (b) และ (c) แสดงตัวอย่างการคำนวณค่าของแต่ละคุณลักษณะเด่น เช่น การคำนวณจำนวนของเส้นของทุกๆ ทิศทางได้ดังสมการ

$$\text{ค่าจำนวนของเส้น} = 1 - ((\text{จำนวนของเส้น}/10) \times 2) \quad (2.30)$$

เช่นถ้าบล็อกที่กำลังพิจารณามีเส้นที่มีทิศทางแนวนอนจำนวน 1 เส้นจะได้ค่าจำนวนของเส้นแนวนอนเท่ากับ $1 - ((1/10) \times 2) = 0.8$ เป็นต้น

$$\text{ค่าความยาวของเส้น} = \frac{\text{จำนวนจุดที่ระบุทิศทางที่พิจารณา}}{\text{ความสูงหรือความกว้างของบล็อก}} \quad (2.31)$$

เช่นถ้าบล็อกที่มีทิศทางแนวดิ่งมีความยาว 7 พิกเซลและบล็อกมีขนาด 10x13 สามารถคำนวณค่าความยาวของเส้นได้เท่ากับ $7/(13 \times 2) = 0.269$ เป็นต้น ดังนั้นเมื่อดึงคุณลักษณะครบทุกบล็อกแล้วเราจะได้เวกเตอร์คุณลักษณะเด่นที่มีขนาด 1x81 มิติ

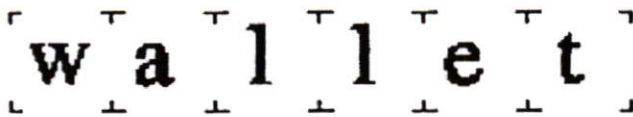
การดึงคุณลักษณะเด่นที่นำมาใช้ในการเปรียบเทียบกับเทคนิคที่นำเสนอคือการดึงคุณลักษณะจากการเปลี่ยนตำแหน่งจุดภาพขาวไปเป็นจุดภาพดำ (transition feature) [9] ทั้งในทิศทางแนวดิ่งและแนวนอน ซึ่งเทคนิคนี้สามารถดำเนินการกับภาพตัวอักษรที่ไม่ได้ผ่านการประมวลผลเบื้องต้นได้ และเวกเตอร์คุณลักษณะสำคัญจะมีขนาด 1x100 มิติ

บทความนี้ได้้นำเทคนิคการดึงคุณลักษณะทั้งสองวิธีมาใช้ร่วมโครงข่ายประสาทเทียมแบบแบคพรอพพาเกชันและโครงข่าย Radial Basis Function (RBF) เพื่อทดสอบประสิทธิภาพในการรู้จำกับข้อมูลลายมือเขียน 2 ชุด ชุดข้อมูลแรกคือ CEDAR CD-ROM Automatically Segmented (CAS) ซึ่งจะแบ่งการทดลองเป็น 2 กลุ่มคือตัวอักษรภาษาอังกฤษตัวพิมพ์เล็กและตัวพิมพ์ใหญ่ และชุดข้อมูลที่สอง Binary Alphanumeric Character (BAC) จากการทดลองสามารถสรุปได้ว่า ระบบการรู้จำตัวอักษรที่ใช้วิธีการดึงคุณลักษณะเด่นที่นำเสนอให้ความถูกต้องในการรู้จำสูงมากกว่า 80% เมื่อทดสอบกับฐานข้อมูล CEDAR

2.3.4 An Approach to Word Image Matching Based on Weighted Hausdorff distance เขียนโดย Yue Lu, Chew Lim Tan, Weihua Huang และ Liying Fan [10]

บทความนี้นำเสนอวิธีการจับคู่กับแม่แบบของคำ (word image matching) โดยการเปรียบเทียบความเหมือนจากเทคนิคการวัดระยะทางแบบ WHD (weighted hausdorff distance) ซึ่งจะมีประโยชน์อย่างมากในการค้นหาคำที่ต้องการในภาพเอกสาร

ขั้นตอนการประมวลผลภาพเบื้องต้นในการเปรียบเทียบจับคู่กับแม่แบบของคำ ประกอบด้วย 2 ขั้นตอนคือการตัดช่องว่างระหว่างตัวอักษรในคำ ปกติขนาดของตัวอักษรจะถูกกำหนดเป็นจำนวนพิกเซลต่อหนึ่งตัวอักษร เช่น ตัวอักษรขนาด 16 x16 พิกเซล ด้วยเหตุผลนี้จึงทำให้ช่องว่างระหว่างตัวอักษรไม่เท่ากัน จากรูปที่ 2.13 (a) จะเห็นว่าช่องว่างระหว่างตัวอักษร 1 ทั้งสองตัวมีมากกว่าช่องว่างระหว่างตัวอักษร w และ a ดังนั้นจึงต้องมีการตัดช่องว่างระหว่างตัวอักษรทั้งทั้งภาพคำที่เป็นแม่แบบ (template image) และภาพคำที่ดึงจากเอกสาร (word image) และขั้นตอนที่สองเป็นการปรับขนาดของภาพคำที่ดึงมาจากเอกสารให้มีขนาดเท่ากันขนาดของภาพคำที่เป็นแบบที่จะใช้ในการเปรียบเทียบ



(a) ภาพคำก่อนการประมวลผล



(b) ภาพคำที่ผ่านการตัดช่องว่าง

รูปที่ 2.13 การตัดช่องว่างของภาพคำ

การวัดระยะทางแบบ WHD เป็นเทคนิคที่พัฒนาต่อเนื่องมาจากการวัดระยะทางแบบ MHD (modified hausdorff distance) โดยเพิ่มการกำหนดค่าน้ำหนักให้กับพื้นที่ของคำตามระดับความสำคัญในการรู้จำ ถ้ากำหนดให้เซต $A = \{a_1, a_2, \dots, a_{N_a}\}$ และเซต $B = \{b_1, b_2, \dots, b_{N_b}\}$ แล้ว $d(a, B)$ คือระยะทางที่สั้นที่สุดจากจุด a ไปยังทุกๆ จุดของเซต B จะได้ว่า

$$H_{\text{WHD}} = \max(h_{\text{WHD}}(A, B), h_{\text{WHD}}(B, A)) \quad (2.32)$$

โดยที่

$$h_{\text{WHD}}(A, B) = \frac{1}{N_a} \sum_{a \in A} w(a) \cdot d(a, B) \quad (2.33)$$

และ

$$\sum_{a \in A} w(a) = N_a \quad (2.34)$$

บทความนี้ได้ทดสอบเทคนิคที่นำเสนอโดยการค้นหาคำที่ต้องการในภาพเอกสารภาษาอังกฤษและภาพเอกสารภาษาจีน โดยภาพเอกสารเหล่านี้ได้จากการสแกนภาพจากหนังสือต่างๆ และหนังสือพิมพ์ของหอสมุดกลางของมหาวิทยาลัยแห่งชาติของสิงคโปร์

การกำหนดค่าน้ำหนักสำหรับภาพเอกสารภาษาอังกฤษ จะทำการแบ่งภาพคำภาษาอังกฤษเป็น 3 ส่วนคือระดับบน ระดับกลางและระดับล่าง ดังรูปที่ 2.14 และค่าน้ำหนักของแต่ละระดับแทนด้วย $w(t)$, $w(m)$ และ $w(d)$ ตามลำดับ

proceeding ascender
mid zone
descender

รูปที่ 2.14 การแบ่งภาพคำภาษาอังกฤษเพื่อกำหนดค่าน้ำหนัก

ฟังก์ชันน้ำหนักสำหรับภาพคำภาษาอังกฤษ คำนวณได้จาก

$$w(t) = w(d) = 2 \times w(m) \quad (2.35)$$

การกำหนดค่าน้ำหนักสำหรับภาพเอกสารภาษาจีน ซึ่งจะมีความแตกต่างจากภาษาตะวันตกคือบริเวณมุมทั้ง 4 ด้านของคำภาษาจีนมีความสำคัญมากในการรู้จำของมนุษย์ ดังนั้นจึงได้แบ่งพื้นที่คำภาษาจีนเป็น 3 ส่วนคือพื้นที่มุม, พื้นที่ตรงกลางและพื้นที่รอบข้าง ดังรูปด้านล่าง

c	p	c
p	h	p
c	p	c

รูปที่ 2.15 การแบ่งภาพคำภาษาจีนเพื่อกำหนดค่าน้ำหนัก

ถ้ากำหนดให้ $w(c)$, $w(h)$ และ $w(p)$ คือค่าน้ำหนักของพื้นที่มุม, พื้นที่ตรงกลางและพื้นที่รอบข้าง ตามลำดับ จะได้ฟังก์ชันน้ำหนักสำหรับภาพคำภาษาจีน ดังนี้

$$w(c) = 2 \times w(h) = 3 \times w(p) \quad (2.36)$$

การกำหนดค่าฟังก์ชันน้ำหนักของภาพเอกสารทั้งสองจะสัมพันธ์กับจำนวนจุดภาพคำของแต่ละพื้นที่ จากผลการทดลองพบว่าการวัดระยะทางแบบ WHD มีผลทำให้ประสิทธิภาพในการรู้จำได้ดีกว่าการวัดระยะทางแบบ MHD สำหรับภาพเอกสารภาษาอังกฤษและภาพเอกสารภาษาจีน ถ้าต้องการนำไปประยุกต์ใช้กับภาพเอกสารภาษาอื่นๆ จะต้องมีการวิเคราะห์ว่าภาษานั้นๆ ให้ความสำคัญของแต่ละตำแหน่งอย่างไร เนื่องจากการกำหนดค่าฟังก์ชันน้ำหนักนี้พิจารณาจากระดับความสำคัญของพื้นที่เป็นหลัก

2.3.5 Off-line Handwritten Thai Characters from Word Script เขียนโดย Suraphun Airphaiboon, Manas Sangworasil และ Shozo Kondo [11]

บทความนี้นำเสนอวิธีการรู้จำตัวอักษรลายมือเขียนภาษาไทยแบบออฟไลน์ โดยการนำข้อมูลคุณลักษณะทางโครงสร้างของตัวอักษร (Structural Information) มาใช้ในการจำแนกกลุ่มตัวอักษร ประกอบด้วยขั้นตอนการทำงานดังนี้

ขั้นตอนที่ 1 การประมวลผลภาพเบื้องต้น จะแบ่งเป็น 3 ขั้นตอนย่อยๆ คือ

1. การกำหนดรูปแบบของข้อมูลอินพุต ซึ่งข้อมูลอินพุตที่ใช้ในที่นี้คือ รูปภาพคำไทยที่เป็นภาพตัวอักษรสีดำบนพื้นหลังสีขาวที่สแกนด้วยความละเอียด 100 จุดต่อนิ้วและมีขนาด 400x200 พิกเซล ซึ่งหลังจากการสแกนแล้วภาพที่ได้อาจจะมียูนิฟอร์มรบกวนจึงต้องนำไปผ่านกระบวนการกำจัดสัญญาณรบกวนก่อน

2. การปรับความเอียง การเขียนมีหลากหลายรูปแบบทั้งลักษณะการเขียนเอียงไปทางด้านหน้าหรือเอียงไปทางด้านหลัง เพื่อให้ระบบการรู้จำมีประสิทธิภาพมากขึ้น จึงได้เพิ่มขั้นตอนการปรับรูปภาพให้ตั้งตรง โดยเริ่มต้นจากการหามุมเอียงของภาพ ตามอัลกอริทึมนี้

- ทำการ โพรเจกชันแนวนอน แกวใดที่มีค่าความถี่น้อยกว่าค่าความถี่เฉลี่ยจะถูกตัดทิ้ง ไม่นำมาพิจารณาในขั้นต่อไป

- หาช่วงบรรทัดของตัวอักษรระดับกลางหรือช่วงบรรทัดของพยัญชนะ โดยพิจารณาจากจำนวนแถวที่มีค่าความถี่มากกว่าค่าความถี่เฉลี่ยที่เรียงติดต่อกันในแนวตั้งมากที่สุด

- ทำการ โพรเจกชันแนวตั้งเฉพาะพื้นที่บรรทัดของตัวอักษรระดับกลาง เพื่อให้ได้วินโดว์ของส่วนของตัวอักษรแต่ละตัว

- นำแต่ละวินโดว์ของส่วนของตัวอักษร มาแบ่งแนวนอนเป็นวินโดว์ย่อย 2 ส่วนขนาดเท่าๆ กันคือส่วนบนและส่วนล่าง จากนั้นหาจุดศูนย์กลางของส่วนของตัวอักษรทั้งสองส่วน แล้วลากเส้นตรงผ่านจุดศูนย์กลางทั้งสอง เพื่อคำนวณหามุมที่เส้นตรงนี้ทำมุมกับแกน x

- การปรับค่าความเอียงของรูปภาพคำ ทำได้โดยการย้ายจุดภาพคำจากจุด (x, y) ไปยังจุด (x', y') ตามสมการต่อไปนี้

$$x' = x - y \tan(\beta - \text{def}); \quad y' = y \quad (2.37)$$

เมื่อ β คือค่ามุมเอียงเฉลี่ยของวินโดว์ทั้งหมด

def คือค่าความเอียงปกติที่กำหนดขึ้น

ขั้นตอนที่ 2 การแยกตัวอักษรออกจากคำ

การทำฮิสโตแกรมจะช่วยแยกตัวอักษรแต่ละระดับชั้นออกจากกันได้ แต่ยังคงมีตัวอักษรบางตัวที่คาบเกี่ยวกันระหว่างระดับกลางและระดับบน เช่น โ ไ เป็นต้น หรือคาบเกี่ยวกันระหว่างระดับกลางกับระดับล่าง เช่น ฎ ฏ เป็นต้น สำหรับตัวอักษรเหล่านี้จะต้องทำการติดตามแนวเส้น

ศูนย์กลาง จากนั้นจุดปลาย EP จะตกอยู่ในแต่ละบล็อกที่แตกต่างกัน ซึ่งวิธีนี้ยังสามารถใช้ในการจำแนกตัวอักษรของกลุ่ม C ได้ด้วยและให้พิจารณาคำแหน่งของรูป LP ประกอบด้วย

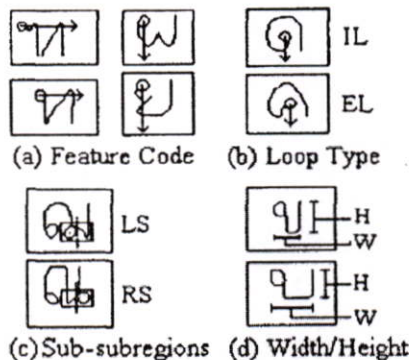
วิธีการจำแนกกลุ่มตัวอักษรในกลุ่ม B ทำได้โดยการแบ่ง Cp pattern เป็น 3 ส่วนเท่าๆ กัน ตามแนวนอน แล้วพิจารณาว่าตำแหน่งของรูป LP อยู่ในพื้นที่ส่วนบน UL ส่วนกลาง ML หรือส่วนล่าง LL ส่วนการพิจารณาคำแหน่งของจุดปลาย EP จะใช้วิธีเดียวกับกลุ่ม A แต่การพิจารณาเพียงตำแหน่งของจุดปลาย EP และตำแหน่งของรูป LP ยังไม่สามารถจำแนกกลุ่ม B และ C ได้ อย่างสมบูรณ์ จึงได้มีการใช้คุณลักษณะอื่นๆ ประกอบ คือ

Feature Code : FC จำนวนจุดภาพคำที่นับได้จากการลากเส้นผ่านตัวอักษรจากตำแหน่งของรูปทั้งในแนวนอนและแนวตั้ง

Loop type : LT พิจารณาประเภทของรูปว่าเป็นรูปภายใน IL หรือรูปภายนอก EL ถ้าเป็นรูปภายในตัวอักษร จะมีค่า $FC > 1$ และถ้าเป็นรูปภายนอกตัวอักษร จะมีค่า $FC = 1$

Sub-subregions : SR จากการที่แบ่งพื้นที่เป็น 4 บล็อกแล้วให้ทำการแบ่งพื้นที่ส่วนที่ 4 เป็นส่วนย่อยอีกตามแนวตั้ง เพื่อพิจารณาว่ารูปของตัวอักษรอยู่ในพื้นที่ย่อยทางด้านซ้าย LS หรือด้านขวา RS

Width/Height ratio : WH คืออัตราส่วนความกว้างต่อความยาวของตัวอักษร เพื่อใช้ในการแยกระหว่าง “ข” และ “บ”



รูปที่ 2.18 ข้อมูลคุณลักษณะ FC, LT, SR และ WH

การทดลองได้ใช้ทดสอบกับตัวพยัญชนะ 44 ตัว สระ 32 ตัวและวรรณยุกต์ 6 ตัว โดยข้อมูลอินพุตเป็นรูปภาพคำจำนวน 100 ชุดที่เขียนโดยผู้เขียนจำนวน 10 คน ผลการทดลองให้อัตราการรู้จำที่ถูกต้องถึง 99% แต่เทคนิควิธีที่นำเสนอนี้ใช้การจำแนกกลุ่มตัวอักษรจากจำนวนรูป ตำแหน่งของรูป และตำแหน่งของจุดปลาย ทำให้ยังมีข้อจำกัดสำหรับการเขียนบางรูปแบบ เช่น ลักษณะการเขียนตัวอักษรที่ไม่ได้ขมวดหัวเป็นวง การเขียนที่ตำแหน่งของจุดปลายอาจจะไม่ได้ลงตรงบล็อกปกติ การเขียนที่ทำให้เกิดรูปภายในตัวอักษร เช่น ข ข เป็นต้น ซึ่งสาเหตุเหล่านี้เกิดขึ้นได้บ่อยครั้งและมีผลทำให้ประสิทธิภาพในการรู้จำลดลง

2.3.6 Distinctive Feature Analysis for Thai Handwritten Character Recognition based on modified Stroke Changing Sequence เขียนโดย P. Charuengwiwat, S. Jitapankul, L. Wuttisittikullij และ P. Seehapan [12]

บทความนี้นำเสนอการรู้จำตัวอักษรลายมือเขียนภาษาไทย โดยแบ่งขั้นตอนการทำงานเป็น 2 ขั้นตอนหลักๆ ดังนี้ในขั้นตอนแรกเป็นการจัดกลุ่มของตัวอักษรจากคุณลักษณะเฉพาะของตัวอักษรและขั้นตอนที่สองการดึงคุณลักษณะเด่นของตัวอักษรโดยการปรับปรุงวิธีการหาลำดับการเปลี่ยนแนวเส้น (Stroke Changing Sequence) ของ Gwo-En และ Jhing-Fa [13]

การจัดกลุ่มของตัวอักษรจากคุณลักษณะเฉพาะของตัวอักษรได้นำวิธีการจัดกลุ่มของ Airphaiboon, S. และ Sangworasil, M. [14] มาประยุกต์ใช้ โดยการจัดกลุ่มวิธีนี้จะพิจารณาจากจำนวนหัวหรืออุปภายในตัวอักษรซึ่งสามารถแบ่งกลุ่มของตัวอักษรไทยได้ 4 กลุ่มคือ

กลุ่มที่ 1 ตัวอักษรไม่มีรูป มีจำนวน 2 ตัวคือ ก และ ฅ

กลุ่มที่ 2 ตัวอักษรที่มี 1 รูป มีจำนวนทั้งหมด 27 ตัวคือ ข, ฃ, ค, ฅ, ง, จ, ง, จ, ๓, ๔, ๕, ๖, ๗, ๘, ๙, ๐, ๑, ๒, ๓, ๔, ๕, ๖, ๗, ๘ และ ๙

กลุ่มที่ 3 ตัวอักษรที่มี 2 รูป มีจำนวนทั้งหมด 14 ตัวคือ ๐, ๑, ๒, ๓, ๔, ๕, ๖, ๗, ๘, ๙, ๐, ๑, ๒ และ ๓

กลุ่มที่ 4 ตัวอักษรที่มี 3 รูป มี 1 ตัวคือ ๐

Gwo-En และ Jhing-Fa ได้นำเสนอเทคนิควิธีการหาลำดับการเปลี่ยนแนวเส้นสำหรับการรู้จำตัวเลขอารบิก ซึ่งเทคนิคนี้มีจุดเด่นที่สามารถดำเนินการกับภาพที่ไม่ต้องผ่านกระบวนการหาโครงร่างของภาพได้ โดยจะเริ่มพิจารณาหาลำดับการเปลี่ยนแนวเส้นตามแนวนอนตั้งแต่แถวแรกจนถึงแถวสุดท้ายของภาพตัวเลข จากนั้นนำค่าลำดับที่ได้มาออร์มอลไลน์เป็นรหัสการเปลี่ยนแนวเส้นตามแนวนอน (horizontal Stroke Changing Sequence) ตัวอย่างการนำวิธีดังกล่าวมาใช้กับภาพตัวอักษรไทย ดังรูปที่ 2.19

เทคนิคการหาลำดับการเปลี่ยนแนวเส้นของ Gwo-En และ Jhing-Fa จะมีการหารหัสการเปลี่ยนแนวเส้นเฉพาะแนวนอนเท่านั้น เมื่อนำมาใช้กับตัวอักษรไทยจะมีตัวอักษรหลายตัวที่ให้รหัสการเปลี่ยนแนวเส้นตามแนวนอนซ้ำกันเช่น ป กับ ๓ เป็นต้น บทความนี้จึงได้เสนอวิธีการปรับปรุงการหาลำดับแนวเส้นใหม่โดยการหารหัสการเปลี่ยนแนวเส้นทั้งตามแนวนอนและแนวตั้งเฉพาะตำแหน่งของรูปของตัวอักษรเท่านั้น แต่ถ้าเป็นกลุ่มตัวอักษรที่ไม่มีรูปจะหารหัสการเปลี่ยนแนวเส้นจากตำแหน่งกึ่งกลางของตัวอักษรแทน



รูปที่ 2.21 ภาพตัวอักษรที่มีปัญหาจำนวนรูปที่เกิดจากการเขียน

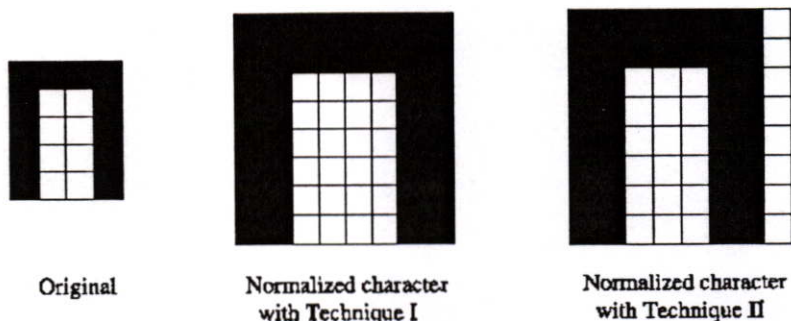
2.3.7 Thai OCR : A Neural Network Application เขียนโดย Chularat Tanprasert และ Thaweesak Koanantakool [15]

บทความนี้นำเสนอระบบการรู้จำตัวอักษรตัวพิมพ์ภาษาไทยด้วยโครงข่ายประสาทเทียม ปัจจุบันงานวิจัยทางด้านนี้มีอยู่มากมาย แต่การนำไปใช้งานจริงยังทำได้ไม่สมบูรณ์มากนัก เนื่องจากปัจจัยและข้อจำกัดหลายประการ เช่น ความไม่เที่ยงตรงในการสแกนภาพ ความไม่สมบูรณ์ของเอกสาร เอกสารมีสัญญาณรบกวน เป็นต้น ดังนั้นงานวิจัยนี้จึงพัฒนาระบบการรู้จำตัวอักษรตัวพิมพ์ภาษาไทยที่นอกจากจะเน้นขั้นตอนการรู้จำแล้วยังให้ความสนใจในขั้นตอนการประมวลภาพเบื้องต้น และขั้นตอนการประมวลผลหลังการรู้จำเพื่อเพิ่มประสิทธิภาพระบบการรู้จำให้ถูกต้องมากขึ้นด้วย

การประมวลภาพเบื้องต้น เป็นขั้นตอนในการปรับปรุงและเปลี่ยนแปลงข้อมูลอินพุทให้อยู่ในรูปแบบที่สามารถใช้งานร่วมกับโครงข่ายประสาทเทียมได้ โดยเริ่มต้นจากการนำเอกสารมาสแกนเก็บไว้ให้มีนามสกุลเป็น .tif จากนั้นจึงแปลงให้เป็นภาพแบบไบนารี ซึ่งภาพที่ได้อาจจะมีสัญญาณรบกวนหรือภาพเอกสารมีความเอียงจากขั้นตอนสแกน ดังนั้นจึงต้องมีการกำจัดสัญญาณรบกวน ปรับความเอียงของเอกสารให้เรียบร้อย หลังจากนั้นจึงจะทำการแยกบรรทัดของตัวอักษรจากภาพเอกสาร และแยกตัวอักษรเดี่ยวแต่ละตัวออกมา

สถาปัตยกรรมโครงข่ายประสาทเทียมจะรับเวกเตอร์อินพุทที่มีขนาดคงที่ จึงจำเป็นต้องทำการปรับขนาดเพื่อให้ตัวอักษรทุกตัวที่ใช้ในการเรียนรู้และการทดสอบมีขนาดเท่ากัน เนื่องด้วยตัวอักษรภาษาไทยมีความหลากหลาย บางตัวมีความกว้างกว่าปกติ เช่น ณ ญ ฒ เป็นต้น ตัวผอมบางกว่าปกติ เช่น ไ โ เป็นต้น จากการศึกษาเกี่ยวกับคุณลักษณะของตัวอักษรไทย พบว่าเมตริกซ์ขนาด 8x23 เหมาะสำหรับตัวอักษรภาษาไทย แต่ว่าพื้นที่บางส่วนจะถูกใช้สำหรับตัวอักษรบางตัวเท่านั้น งานวิจัยนี้จึงทดลองเปรียบเทียบขนาดรูปภาพที่แตกต่างกันเพิ่มเติมขึ้นคือขนาดเมตริกซ์ 8x8, 8x12 และ 8x16

บทความนี้ได้เสนอวิธีการย่อหรือขยายข้อมูลอินพุทลงในเมตริกซ์อินพุทขนาดต่างๆ กันไว้ 2 วิธีด้วยกันคือ เทคนิคที่ 1 การวางข้อมูลอินพุทโดยย่อหรือขยายให้พอดีทั้งความกว้างและความยาวของเมตริกซ์อินพุท และเทคนิคที่ 2 การขยายข้อมูลอินพุทในรูปอัตราส่วนให้พอดีในความกว้างหรือความยาวของเมตริกซ์อินพุทอย่างใดอย่างหนึ่งเท่านั้น



รูปที่ 2.22 วิธีการย่อหรือขยายข้อมูลอินพุตลงในเมตริกซ์อินพุต

ข้อมูลที่ใช้ในการทดลองเป็นตัวอักษรตัวพิมพ์ภาษาไทยจำนวน 10 ฟอรัมคือ AngsanaUPC, BrowalliaUPC, CordiaUPC, DilleniaUPC, EucrosiaUPC, FreesiaUPC, IrisUPC, JasmineUPC, SV Busaba และ TS Burrirum โดยทุกๆ ฟอรัมจะประกอบด้วยตัวอักษร 46 ตัว สระ 14 ตัว วรรณยุกต์ 4 ตัว ตัวเลขไทย 10 ตัว และสัญลักษณ์พิเศษ 4 ตัวรวมทั้งรวม 78 ตัวที่มีขนาด 8-22 จุดและมีรูปร่าง 4 แบบ คือ ตัวปกติ ตัวเอียง ตัวหนา และตัวหนาและเอียง ดังนั้นจำนวนตัวอักษรทั้งหมดของแต่ละฟอรัมจะมี 2,496 ตัว แล้วนำมาทดสอบกับโครงข่ายประสาทเทียมแบบแบคพรอพพาเกชันที่มีจำนวน โหนดชั้นอินพุตเท่ากับขนาดของเมตริกซ์ที่กำหนดไว้ จำนวน โหนดของชั้นซ่อนจะกำหนดตามความเหมาะสมของแต่ละโครงข่ายและจำนวน โหนดของชั้นเอาต์พุตมี 78 โหนด

ผลการทดลองสรุปได้ว่าขนาดของเมตริกซ์ที่เหมาะสมคือขนาด 8x8 ซึ่งจากผลการทดลองของเมตริกซ์ทุกขนาดให้ผลการทดลองที่ใกล้เคียง ดังนั้นจึงเลือกขนาดเมตริกซ์ที่เล็กที่สุดเพื่อประหยัดเวลาและพื้นที่ในการประมวลผล สำหรับเทคนิคการย่อและขยายข้อมูลอินพุตลงในเมตริกซ์อินพุตให้ผลไม่แตกต่างกันมาก จึงสรุปได้ว่าทั้งสองเทคนิคมีประสิทธิภาพใกล้เคียงกัน

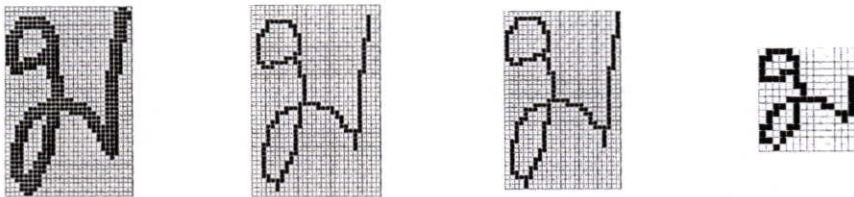
บทที่ 3

การประยุกต์ใช้เมตริกซ์ค่าน้ำหนักร่วมกับแบบจำลองโครงข่ายประสาทเทียมแบบลำดับชั้นสำหรับการรู้จำตัวอักษรลายมือเขียนภาษาไทย

3.1 บทนำ

งานวิจัยนี้นำเสนอระบบการรู้จำตัวอักษรลายมือเขียนภาษาไทยที่มีการวัดระยะทางจากเมตริกซ์ค่าน้ำหนักมาปรับใช้ร่วมกับแบบจำลองโครงข่ายประสาทเทียมซิมพลิไฟด์พีซซีอาร์ทแมพ ซึ่งเทคนิคใหม่นี้จะไม่มีการดึงคุณลักษณะเด่นของตัวอักษร ดังนั้นกระบวนการทำงานจึงประกอบด้วย 2 ขั้นตอนหลักๆ คือ ขั้นตอนการประมวลผลภาพเบื้องต้นและขั้นตอนการรู้จำ

ขั้นตอนการประมวลผลภาพเบื้องต้น เป็นขั้นตอนการเตรียมข้อมูลภาพเพื่อใช้ในการเรียนรู้และการทดสอบ ข้อมูลภาพที่ใช้ในงานวิจัยนี้เป็นภาพของตัวอักษรลายมือเขียนอักษรเดี่ยว (isolated character) และเป็นภาพแบบไบนารีที่มีตัวอักษรสีดำอยู่บนพื้นหลังสีขาว เริ่มต้นจากการนำภาพตัวอักษรไปหาโครงร่างของภาพด้วยอัลกอริทึมของฮิลคิทซ์ตามที่กล่าวไว้แล้วในบทที่ 2 จากนั้นนำภาพโครงร่างที่ได้มาผ่านกระบวนการหาขอบภาพและขั้นตอนสุดท้ายทำการปรับขนาดของภาพตัวอักษรตามต้องการ ในงานวิจัยนี้ได้ใช้ภาพตัวอักษรขนาด 15x15 พิกเซล



(a) ภาพก่อนการประมวลผล (b) โครงร่างของภาพ (c) การหาขอบภาพ (d) ภาพขนาด 15x15 พิกเซล

รูปที่ 3.1 ขั้นตอนการประมวลผลภาพเบื้องต้น

ขั้นตอนการรู้จำสามารถแบ่งการทำงานได้เป็น 2 ส่วน ส่วนแรก ขั้นตอนการเรียนรู้เป็นขั้นตอนที่สอนให้คอมพิวเตอร์เรียนรู้และจดจำภาพตัวอักษรและส่วนที่สอง ขั้นตอนการทดสอบจะใช้สำหรับวัดประสิทธิภาพของอัลกอริทึมที่ใช้ในการเรียนรู้ การทำงานเริ่มต้นจากนำภาพตัวอักษรที่ผ่านการประมวลผลภาพเบื้องต้นแล้วมาแบ่งเป็นบล็อกขนาดเท่าๆ กันและนำค่าจุดภาพของแต่ละบล็อกมาสร้างเป็นเวกเตอร์อินพุตและคลาสของภาพตัวอักษรจะถูกนำมาสร้างเป็นเวกเตอร์เป้าหมายเพื่อนำไปใช้ในขั้นตอนการเรียนรู้ตามอัลกอริทึมของโครงข่ายประสาทเทียม

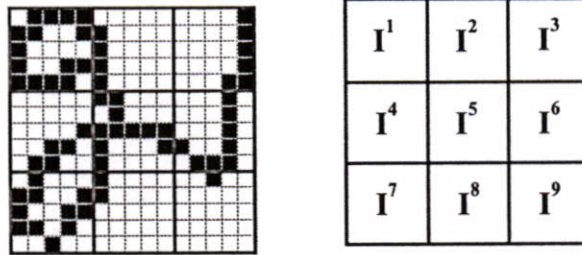
แบบซิมพลิไฟด์พีซซีอาร์ทแมพที่ได้ปรับเปลี่ยนการวัดระยะทางด้วยเทคนิคใหม่ที่น่าเสนอคือการวัดระยะทางจากเมตริกซ์ค่าน้ำหนัก และออกแบบสถาปัตยกรรมของโครงข่ายประสาทเทียมให้เป็นแบบลำดับชั้นซึ่งจะกล่าวรายละเอียดต่อไปในหัวข้อที่ 3.3

3.2 การเตรียมเวกเตอร์อินพุท

เวกเตอร์อินพุทเป็นเซตของข้อมูลที่จะนำไปใช้ในกระบวนการเรียนรู้ ขั้นตอนการสร้างเวกเตอร์อินพุทเริ่มต้นจากการนำภาพที่ผ่านกระบวนการประมวลผลภาพเบื้องต้น ตัวอย่างดังรูปที่ 3.1 (d) จากนั้นจะนำมาแบ่งเป็นบล็อกภาพขนาดเท่าๆ กันในที่นี้จะแสดงตัวอย่างการแบ่งบล็อกภาพเป็น 9 บล็อก ตั้งแต่บล็อกที่ 1 ถึงบล็อกที่ 9 โดยเรียงลำดับบล็อกจากด้านซ้ายไปขวาและจากด้านบนลงล่างดังรูปที่ 3.2 จะได้เวกเตอร์อินพุท I ดังนี้

$$I = \{I^1, I^2, \dots, I^s, \dots, I^9\} \quad (3.1)$$

$$= \{\{I_1^1, \dots, I_N^1\}, \{I_1^2, \dots, I_N^2\}, \dots, \{I_1^s, \dots, I_N^s\}, \dots, \{I_1^9, \dots, I_N^9\}\}$$



รูปที่ 3.2 การเตรียมเวกเตอร์อินพุท

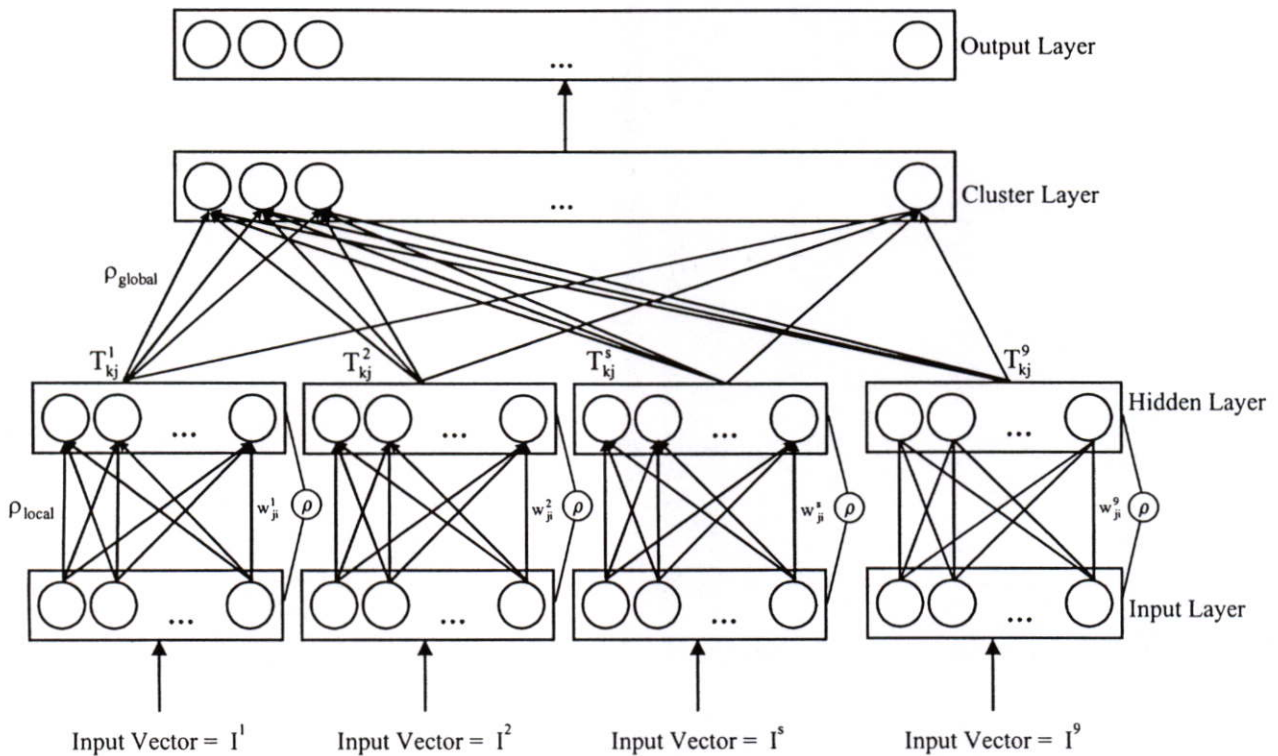
เช่น $I^1 = \{(1, 0, 0, 0, 0), (0, 0, 1, 1, 0), (0, 1, 1, 1, 1), (0, 1, 1, 0, 0), (0, 0, 0, 0, 1)\}$ ซึ่งเป็นเวกเตอร์อินพุทของบล็อกที่ 1 เป็นต้น เมื่อกำหนดให้ค่าจุดภาพดำและจุดภาพขาวแทนด้วย 0 และ 1 ตามลำดับ

3.3 สถาปัตยกรรมโครงข่ายประสาทเทียมแบบลำดับชั้นที่น่าเสนอ

สถาปัตยกรรมโครงข่ายประสาทเทียมแบบลำดับชั้นที่น่าเสนอประกอบด้วย 4 ระดับชั้นคือชั้นอินพุทเป็นชั้นแรก (input layer) ชั้นสองคือชั้นซ่อน (hidden layer) ชั้นถัดไปคือชั้นคลัสเตอร์ (cluster layer) และชั้นสุดท้ายคือชั้นเอาต์พุท (output layer) ดังรูปที่ 3.3

ชั้นอินพุทเป็นชั้นที่รับเวกเตอร์อินพุทเข้าสู่โครงข่าย ถ้าภาพที่ใช้ในการเรียนรู้มีการแบ่งบล็อกภาพเป็น 9 บล็อกแล้วชั้นนี้จะถูกแบ่งเป็น 9 เซกเมนต์เช่นเดียวกัน เริ่มตั้งแต่เซกเมนต์ที่ 1, 2, ..., s, ..., 9 โดยแต่ละเซกเมนต์จะมีจำนวน N โหนดเท่ากับจำนวนข้อมูลอินพุทของแต่ละบล็อก

เช่น ภาพที่ใช้ในการเรียนรู้มีขนาด 15x15 พิกเซล หลังจากทำการแบ่งบล็อกภาพแล้วแต่ละบล็อกภาพจะมีขนาด 5x5 พิกเซล ดังนั้นแต่ละเซกเมนต์จึงมีโหนดจำนวน 25 โหนด เป็นต้น



รูปที่ 3.3 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบลำดับชั้นที่น่าเสนอ

ชั้นที่สองคือชั้นซ่อน จะถูกแบ่งเป็น 9 เซกเมนต์เช่นเดียวกับชั้นอินพุต โดยโหนดของแต่ละเซกเมนต์ในชั้นนี้จะถูกสร้างระหว่างขั้นตอนการเรียนรู้ และทุกๆ โหนดที่สร้างขึ้นจะเชื่อมต่อแบบทั่วถึงไปยังทุกๆ โหนดในเซกเมนต์เดียวกันของชั้นอินพุตผ่านค่าน้ำหนัก w^s_{ji} ซึ่งเป็นค่าน้ำหนักที่เชื่อมจากโหนดที่ i ในเซกเมนต์ที่ s ของชั้นอินพุตไปยังโหนดที่ j ในเซกเมนต์เดียวกันของชั้นซ่อน

ชั้นคลัสเตอร์ แต่ละโหนดในชั้นนี้ถูกสร้างขึ้นระหว่างการเรียนรู้เช่นเดียวกับชั้นก่อนหน้า นี้แต่ทุกๆ โหนดจะเชื่อมไปยังโหนดใดโหนดหนึ่งในแต่ละเซกเมนต์ของชั้นซ่อนผ่านค่าเอาต์พุต T^s_{kj} ซึ่งเป็นค่าเอาต์พุตที่เชื่อมจากโหนดที่ j ในเซกเมนต์ที่ s ของชั้นซ่อน ไปยังโหนดที่ k ของชั้นคลัสเตอร์

ชั้นเอาต์พุตเป็นชั้นที่ทำหน้าที่รับเวกเตอร์เป้าหมาย (target vector) เข้ามายังโครงข่าย ซึ่งเวกเตอร์เป้าหมายนี้ก็คือคลาสของเวกเตอร์อินพุต ดังนั้นเมื่อสิ้นสุดกระบวนการเรียนรู้แล้วจำนวนโหนดของชั้นนี้จะเท่ากับจำนวนคลาสของชุดข้อมูลที่เรียนรู้ทั้งหมด ในงานวิจัยนี้ทดสอบการรู้จำตัวพยัญชนะจำนวน 42 ตัว ดังนั้นชั้นเอาต์พุตจึงมีจำนวนโหนด 42 โหนด

3.4 กระบวนการเรียนรู้

ภาพตัวอักษรทั้งหมดจะถูกแบ่งออกเป็น 2 กลุ่มโดยกลุ่มแรกจะนำไปใช้สำหรับสอนให้คอมพิวเตอร์เรียนรู้และกลุ่มที่สองจะนำไปใช้ในขั้นตอนการทดสอบเพื่อวัดประสิทธิภาพในการรู้จำ เวกเตอร์อินพุตกลุ่มแรกจะถูกนำเข้าสู่โครงข่ายตามลำดับพร้อมด้วยเวกเตอร์เป้าหมายของอินพุตนั้นๆ ตามอัลกอริทึมของกระบวนการเรียนรู้ดังนี้

1. ถ้าเป็นเวกเตอร์อินพุตแรกที่เข้ามาในโครงข่าย จะทำการสร้างโหนดที่หนึ่งของแต่ละเซกเมนต์ในชั้นซ่อนและเชื่อมต่อแบบทั่วถึงกับทุกๆ โหนดในเซกเมนต์เดียวกันของชั้นอินพุตผ่านค่าฟังก์ชันน้ำหนัก (weight function)

$$w_{ii}^s = f(I_i^s) \quad , s = 1, 2, \dots, NB \quad (3.2)$$

โดยที่

$$f(x) = \begin{cases} 0 & , x \notin R_p \\ C & , x \in R_p \end{cases} \quad (3.3)$$

เมื่อ R_p คือเซตพื้นที่จุดภาพค่าของภาพตัวอักษร

C คือค่าน้ำหนักเริ่มต้น

NB คือจำนวนการแบ่งบล็อกภาพ

จากฟังก์ชันน้ำหนักถ้าจุดภาพ i ที่กำลังพิจารณาเป็นจุดภาพค่าจะกำหนดให้มีค่าน้ำหนักเท่ากับ C แต่ถ้าเป็นจุดภาพขาวจะกำหนดให้ค่าน้ำหนักเท่ากับ 0 จากนั้นจะสร้างโหนดแรกของชั้นคลัสเตอร์และทำการเชื่อมต่อกับโหนดที่หนึ่งของทุกๆ เซกเมนต์ในชั้นซ่อน

2. ถ้าเป็นเวกเตอร์อินพุตลำดับที่ 2 เป็นต้นไป จะต้องนำเวกเตอร์อินพุตของบล็อกที่ s มาเปรียบเทียบกับความต่างกับค่าน้ำหนักของทุกๆ โหนดที่ถูกสร้างขึ้นในเซกเมนต์ที่ s ($D(I^s, w_j^s)$) ซึ่งวิธีการวัดระยะทางหาความต่างที่นำเสนอนี้จะกล่าวรายละเอียดในหัวข้อที่ 3.5 เมื่อได้ค่าระยะทางของทุกๆ โหนดครบทั้ง 9 เซกเมนต์แล้ว จะทำการตรวจสอบว่าคลาสของอินพุตที่กำลังพิจารณานี้ได้มีการสร้างมาก่อนหรือไม่

2.1 กรณีที่หนึ่ง ถ้าเคยมีการสร้างคลาสของอินพุตอยู่ก่อนแล้ว ให้คำนวณค่าเอาต์พุตของทุกๆ โหนดในชั้นคลัสเตอร์

$$O_k = \frac{1}{NB} \sum_{s=1}^{NB} T_{kj}^s \quad (3.4)$$

โดยที่

$$T_{kj}^s = D(I^s, w_j^s) \quad (3.5)$$

เมื่อ O_k คือค่าเอาต์พุตของโหนดที่ k ในชั้นคลัสเตอร์

T_{kj}^s คือค่าฟังก์ชันการกระตุ้นที่เชื่อมจากโหนดที่ j ในเซกเมนต์ที่ s ของชั้นช่อนไปยังโหนดที่ k ของชั้นคลัสเตอร์

จากนั้นให้ทำการจัดเรียงค่าเอาต์พุตทั้งหมดโดยเรียงลำดับจากค่าน้อยไปยังค่ามากเพื่อนำไปหาโหนดที่ชนะ (winning node) ตามสมการที่ (3.6) แต่ถ้ามีโหนดที่ให้ค่าเอาต์พุตน้อยที่สุดมากกว่าหนึ่งโหนด ให้เลือกโหนดที่มีค่า k น้อยที่สุด

$$K = \arg \min_k \{O_k\} \quad (3.6)$$

ทำการพิจารณาเปรียบเทียบค่าเอาต์พุตของโหนดที่ชนะกับค่าวิเจเลนซ์รวม ρ_{global} (global vigilance) ถ้าค่าเอาต์พุตดังกล่าวน้อยกว่าหรือเท่ากับค่า ρ_{global} ซึ่งทำให้สมการที่ 3.7 เป็นจริง และให้พิจารณาเงื่อนไขต่อไปคือ คลาสของโหนดที่ชนะตรงกับคลาสของอินพุตหรือไม่ ถ้าคลาสของโหนดที่ชนะตรงกับคลาสของอินพุต แสดงว่าเกิดเรโซแนนซ์

$$O_k \leq \rho_{\text{global}} \quad (3.7)$$

การเกิดเรโซแนนซ์แสดงให้เห็นว่าเวกเตอร์อินพุตของทั้ง 9 บล็อกมีความเหมือนมากพอกับค่าน้ำหนักของโหนดที่ชนะทั้ง 9 เซกเมนต์ ดังนั้นจึงสามารถจัดเวกเตอร์อินพุตนี้ให้อยู่ในกลุ่มของโหนดที่ชนะได้ ซึ่งถือว่าจบการทำงานของอินพุตตัวนี้และให้ไปดำเนินการต่อในขั้นตอนที่ 3

ถ้าค่าเอาต์พุตของโหนดที่ชนะ K น้อยกว่าหรือเท่ากับค่า ρ_{global} แต่ว่าคลาสของโหนดที่ชนะ K ไม่ตรงกับคลาสของอินพุตหรือค่าเอาต์พุตของโหนดที่ชนะ K มากกว่าค่า ρ_{global} ให้ไปดำเนินการในขั้นตอนที่ 2.2 ต่อไป

2.2 กรณีที่สอง ถ้าโครงข่ายไม่เคยมีการสร้างโหนดที่มีคลาสเดียวกับคลาสของอินพุตมาก่อนหรือค่าเอาต์พุตของโหนดที่ชนะ K มากกว่าค่า ρ_{global} หรือโหนดที่ชนะ K มีคลาสไม่ตรงกันกับคลาสของอินพุต ในขั้นตอนนี้จะหาโหนดที่ชนะในแต่ละเซกเมนต์ของชั้นช่อน โดยพิจารณาจากโหนดที่ให้ค่า T_j^s ที่น้อยที่สุดตามสมการที่ 3.8 แต่ถ้ามีโหนดที่ให้ค่า T_j^s น้อยที่สุดมากกว่าหนึ่งโหนด ให้เลือกโหนดที่มีค่า j น้อยที่สุด

$$J = \arg \min_j \{T_j^s(I^s)\} \quad (3.8)$$

จากนั้นให้ทำการเปรียบเทียบค่า T_j^s ของโหนดที่ชนะ J ของเซกเมนต์ที่ s กับค่าโลคอลวิเจเลนซ์ ρ_{local} (local vigilance) ถ้าค่า T_j^s ของโหนดที่ชนะ J น้อยกว่าหรือเท่ากับค่า ρ_{local} จะต้องปรับค่าน้ำหนักที่เชื่อมระหว่างโหนดที่ชนะ J ในเซกเมนต์ที่ s ของชั้นช่อนกับทุกๆ โหนดในเซกเมนต์เดียวกันของชั้นอินพุตตามฟังก์ชันการปรับค่าน้ำหนักต่อไปนี้

$$w_{ji}^s(t+1) = \begin{cases} C & , R_p \cap \bar{R}_w \\ w_{ji}^s(t) + c_1 & , R_p \cap R_w \\ w_{ji}^s(t) - c_2 & , \bar{R}_p \cap R_w \\ 0 & , \bar{R}_p \cap \bar{R}_w \end{cases} \quad (3.9)$$

เมื่อ R_p คือเซตพื้นที่จุดภาพคำของภาพตัวอักษร

\bar{R}_p คือเซตของพื้นที่พื้นหลังของตัวอักษร

R_w คือเซตพื้นที่ของค่าน้ำหนัก

\bar{R}_w คือเซตพื้นที่ที่ไม่มีค่าน้ำหนัก

C คือค่าน้ำหนักเริ่มต้น

c_1, c_2 คือค่าคงที่เพื่อเพิ่มและลดระดับความสำคัญค่าน้ำหนักตามลำดับ

จากสมการที่ (3.9) สามารถอธิบายได้เป็น 4 กรณีดังนี้

กรณีที่ 1 $R_p \cap \bar{R}_w$ ถ้าตำแหน่งจุดภาพที่กำลังพิจารณาอยู่บนพื้นที่ตัวอักษรแต่ไม่ได้อยู่บนพื้นที่ค่าน้ำหนัก แสดงว่าจุดภาพนี้เป็นจุดภาพคำแต่ยังไม่เคยมีการกำหนดค่าน้ำหนักมาก่อน ฟังก์ชันการปรับค่าน้ำหนักจะกำหนดค่าน้ำหนักเริ่มต้นให้เท่ากับ C

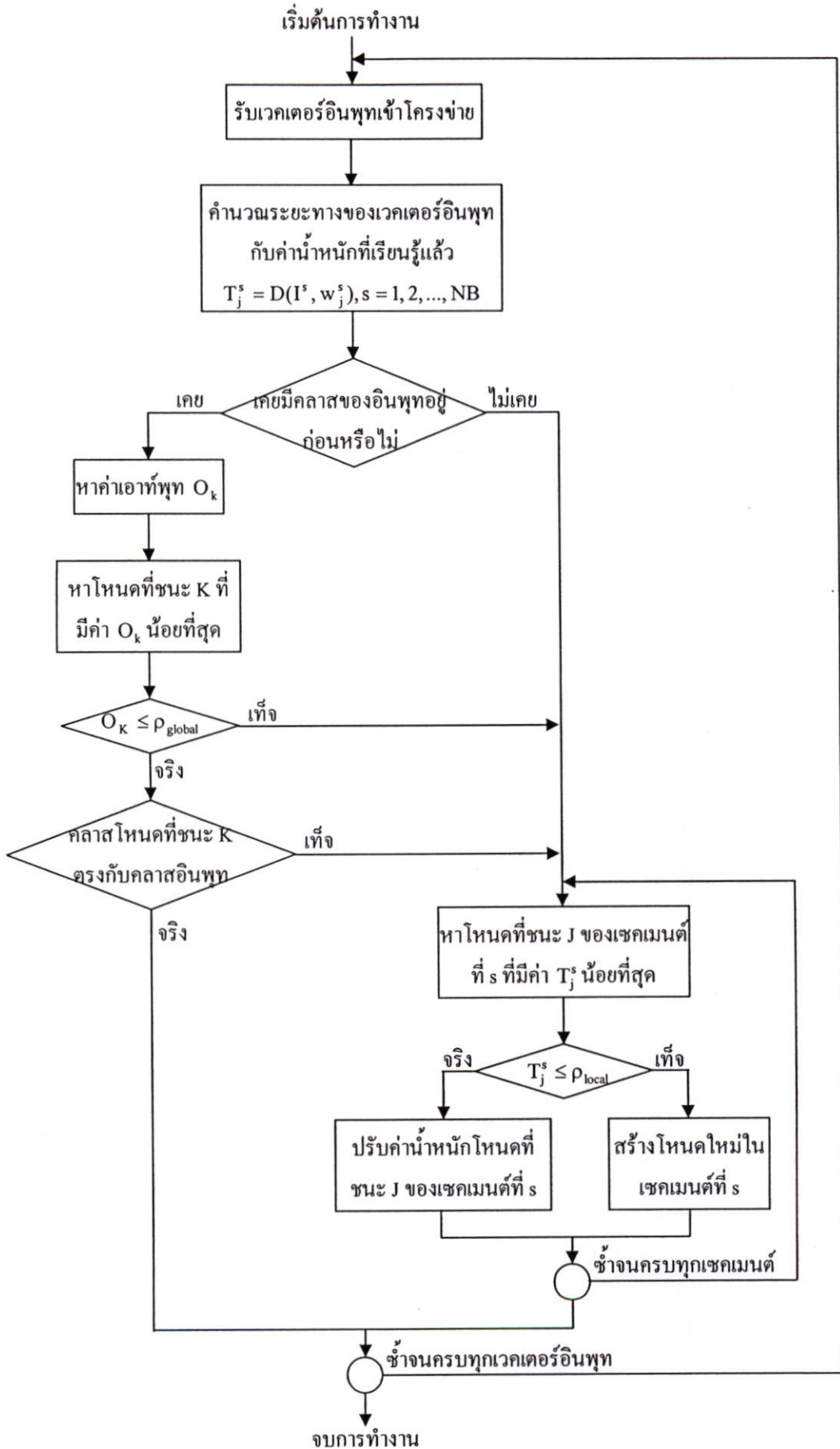
กรณีที่ 2 $R_p \cap R_w$ ถ้านำบล็อกของภาพตัวอักษรวางซ้อนทับบล็อกของค่าน้ำหนักแล้วตำแหน่งของจุดภาพที่กำลังพิจารณาอยู่บนพื้นที่ซ้อนทับระหว่างพื้นที่ทั้งสอง แสดงว่าตำแหน่งนั้นเป็นตำแหน่งของจุดภาพคำที่ได้มีการกำหนดค่าน้ำหนักมาก่อนแล้ว ฟังก์ชันการปรับค่าน้ำหนักจะกำหนดระดับความสำคัญของตำแหน่งดังกล่าวให้สูงขึ้นด้วยการเพิ่มค่าน้ำหนักเดิมด้วยค่า c_1

กรณีที่ 3 $\bar{R}_p \cap R_w$ ถ้าตำแหน่งจุดภาพที่กำลังพิจารณาเป็นพื้นหลังของตัวอักษรแต่ตำแหน่งนั้นเคยมีการกำหนดค่าน้ำหนักมาก่อนแล้ว กล่าวคือจุดภาพนี้เป็นจุดภาพขาวที่เคยมีการกำหนดค่าน้ำหนักไว้แล้ว กรณีนี้ฟังก์ชันการปรับค่าน้ำหนักจะทำการลดระดับความสำคัญของตำแหน่งดังกล่าวลงโดยการลบค่าน้ำหนักเดิมด้วยค่า c_2

กรณีที่ 4 $\bar{R}_p \cap \bar{R}_w$ ถ้าตำแหน่งจุดภาพที่กำลังพิจารณาอยู่บนพื้นหลังของตัวอักษรและตำแหน่งนั้นไม่เคยมีการกำหนดค่าน้ำหนัก ฟังก์ชันการปรับค่าน้ำหนักจะกำหนดค่าน้ำหนักเท่ากับ 0 เช่นเดิม

แต่ถ้าค่า T_j ของโหนดที่ชนะ J มากกว่าค่า p_{local} แสดงว่าโหนดที่ชนะมีความเหมือนกับเวกเตอร์อินพุตของบล็อกที่ s มากที่สุดแต่อย่างไรก็ตามเหมือนไม่มากพอที่จะจัดให้บล็อกที่ s ของอินพุตนี้อยู่รวมกลุ่มกับโหนดที่ชนะได้ ดังนั้นจะต้องสร้างโหนดใหม่ในเซกเมนต์ที่ s ของชั้นซ่อนแล้วให้เชื่อมต่อแบบทั่วถึงกับทุกๆ โหนดในเซกเมนต์เดียวกันของชั้นอินพุตผ่านค่าน้ำหนักที่คำนวณได้โดยการนำเวกเตอร์อินพุตบล็อกที่ s ไปหาค่าฟังก์ชันน้ำหนัก ตามสมการที่ (3.2)

3. ทำซ้ำขั้นตอนที่ 2 จนกระทั่งครบทุกอินพุต



รูปที่ 3.4 แผนภาพขั้นตอนกระบวนการเรียนรู้

3.5 วิธีการวัดระยะทางที่นำเสนอ

กำหนดให้เซต $A = \{a_{11}, a_{12}, \dots, a_{ij}, \dots, a_{mn}\}$ แทนเซตข้อมูลอินพุทของบล็อกที่กำลังพิจารณาและเซต $B = \{b_{11}, b_{12}, \dots, b_{ij}, \dots, b_{mn}\}$ แทนเซตค่านำหนักของโหนดในเซกเมนต์ที่ตรงกับบล็อกที่จะเปรียบเทียบกับ จะได้ว่า

$$D(A, B) = \left(\frac{1}{N_A} \sum_{a \in A} d(a, b) \right) + P_r + P_c \quad (3.10)$$

เมื่อ N_A คือจำนวนจุดภาพคำของเซต A และ $d(a, b)$ คือค่าระยะทางจากจุดภาพคำ a ไปยังค่านำหนัก b ที่สอดคล้องกัน จากสมการดังกล่าวจะเห็นว่าค่าระยะทางจากเซต A ไปยังเซต B ประกอบด้วย 3 เทอมดังนี้ เทอมแรกเป็นค่าเฉลี่ยของระยะทางจากจุดภาพคำทั้งหมดในเซต A ไปยังเซต B เทอมที่สอง P_r เป็นค่าความต่างจำนวนแถว (penalty of row) ของบล็อกข้อมูลอินพุทและบล็อกค่านำหนักและเทอมสุดท้ายคือ P_c เป็นค่าความต่างจำนวนคอลัมน์ (penalty of column) ของบล็อกข้อมูลอินพุทและบล็อกค่านำหนัก ซึ่งสองเทอมสุดท้ายนี้สามารถคำนวณได้จาก

$$P_r = r_1 \times |N_r^A - N_r^B| \quad (3.11)$$

$$P_c = r_2 \times |N_c^A - N_c^B| \quad (3.12)$$

เมื่อ N_r^A, N_r^B คือจำนวนแถวที่มีจุดภาพคำของเซต A และเซต B ตามลำดับ

N_c^A, N_c^B คือจำนวนคอลัมน์ที่มีจุดภาพคำของเซต A และเซต B ตามลำดับ

r_1, r_2 คือค่าสัมประสิทธิ์ของเทอม P_r และ P_c ตามลำดับ

การวัดระยะทางจากจุดภาพคำใดๆ ในเซต A ไปยังเซต B คือการเปรียบเทียบระหว่างเมตริกซ์ของจุดภาพเพื่อนบ้าน (neighborhood point) ของ a ที่มีขนาด $n \times n$ แทนด้วย $M_A^n(a)$ เทียบกับเมตริกซ์ของจุดภาพเพื่อนบ้านของ a ในเซต B ที่มีขนาดเท่ากันแทนด้วย $M_B^n(a)$

ขั้นตอนแรกจะต้องมีการกำหนดขนาดของเมตริกซ์ที่จะใช้ในการเปรียบเทียบระหว่างเซต A และเซต B ก่อน ตัวอย่างเช่น ถ้ากำหนดให้เมตริกซ์ที่ใช้ในการเปรียบเทียบมีขนาด 3×3 แล้วเมตริกซ์ของจุดภาพเพื่อนบ้านของ a_{ij} และเมตริกซ์ของจุดภาพเพื่อนบ้านของ a_{ij} ในเซต B คือ

$$M_A^3(a_{ij}) = \begin{bmatrix} a_{i-1,j-1} & a_{i-1,j} & a_{i-1,j+1} \\ a_{i,j-1} & a_{i,j} & a_{i,j+1} \\ a_{i+1,j-1} & a_{i+1,j} & a_{i+1,j+1} \end{bmatrix}, \quad M_B^3(a_{ij}) = \begin{bmatrix} b_{i-1,j-1} & b_{i-1,j} & b_{i-1,j+1} \\ b_{i,j-1} & b_{i,j} & b_{i,j+1} \\ b_{i+1,j-1} & b_{i+1,j} & b_{i+1,j+1} \end{bmatrix}$$

ดังนั้น $d(a, b)$ คำนวณได้จาก

$$d(a, b) = \begin{cases} \frac{\|M_A^n(a) \cdot M_B^n(a) \cdot E^n\| (1 - b_{ij})}{\|M_B^n(a)\|} & , \|M_B^n(a)\| > 0 \\ C & , \text{otherwise} \end{cases} \quad (3.13)$$

เมื่อ $A \cdot B$ คือ Hadamard product หรือ Entrywise product เป็นการคูณเมตริกซ์อีลิเมนต์ต่ออีลิเมนต์ของเมตริกซ์ A และ B [16]

$\|A\|$ คือการรวมค่าของอีลิเมนต์ทั้งหมดของเมตริกซ์ A

E^n คือยูคลิเดียนเมตริกซ์ (Euclidean distance matrix) จากจุดศูนย์กลางขนาด $n \times n$

C คือค่าคงที่ค่าหนึ่ง ปกติจะให้เท่ากับ $\sqrt{n^2 + n^2}$

ยูคลิเดียนเมตริกซ์ คือเมตริกซ์ของการวัดระยะทางแบบยูคลิเดียนจากจุดศูนย์กลางเช่น ถ้ากำหนดให้ยูคลิเดียนเมตริกซ์มีขนาด 3×3 จะได้

$$E^3 = \begin{bmatrix} \sqrt{2} & 1 & \sqrt{2} \\ 1 & 0 & 1 \\ \sqrt{2} & 1 & \sqrt{2} \end{bmatrix}$$

3.6 ตัวอย่างการคำนวณวิธีการวัดระยะทางที่นำเสนอ

กำหนดให้ A แทนเซตข้อมูลอินพุตบล็อกที่ s (I^s), B แทนเซตค่าน้ำหนักของโหนดหนึ่งในเซกเมนต์ที่ s (w_j^s) และขนาดของเมตริกซ์ที่ใช้ในการเปรียบเทียบมีขนาด 3×3 พิกเซล

ดังนั้นสามารถคำนวณค่า $D(A, B) = \left(\frac{1}{N_A} \sum_{a \in A} d(a, b) \right) + P_r + P_c$ ได้ดังนี้

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

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0.6 \\ 0 & 0 & 0 & 0.3 & 0.6 \\ 0 & 0 & 0 & 0.5 & 0.2 \\ 0 & 0 & 0 & 0.6 & 0 \\ 0 & 0 & 0 & 0.3 & 0 \end{bmatrix}$$

รูปที่ 3.5 ตัวอย่างเซตข้อมูลอินพุตและเซตค่าน้ำหนัก

ขั้นตอนที่ 1 กำหนดขนาดของเมตริกซ์ที่ใช้ในการคำนวณ ในที่นี้กำหนดให้มีขนาด 3×3 ดังนั้นจะได้ยูคลิเดียนเมตริกซ์ดังนี้

$$E^3 = \begin{bmatrix} \sqrt{2} & 1 & \sqrt{2} \\ 1 & 0 & 1 \\ \sqrt{2} & 1 & \sqrt{2} \end{bmatrix}$$

ขั้นตอนที่ 2 คำนวณค่าระยะทางจากจุดภาพค่า a_{43} , a_{34} และ a_{25} ของเซต A ไปยังเซต B

2.1 คำนวณค่า $d(a_{43}, b_{43})$ ได้ดังนี้

$$M_A^3(a_{43}) = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad M_B^3(a_{43}) = \begin{bmatrix} 0 & 0 & 0.5 \\ 0 & 0 & 0.6 \\ 0 & 0 & 0.3 \end{bmatrix}$$

ดังนั้น

$$\begin{aligned} d(a_{43}, b_{43}) &= \frac{((1 \times 0 \times \sqrt{2}) + (1 \times 0 \times 1) + (0 \times 0.5 \times \sqrt{2}) + (1 \times 0 \times 1) + 0 + (1 \times 0.6 \times 1) + (1 \times 0 \times \sqrt{2}) + (1 \times 0 \times 1) + (1 \times 0.3 \times \sqrt{2})) \times (1 - 0)}{0.5 + 0.6 + 0.3} \\ &= \frac{(0 + 0 + 0 + 0 + 0 + 0.6 + 0 + 0 + 0.4243) \times 1}{1.4} \\ &= \frac{1.0243}{1.4} \\ &= 0.7316 \end{aligned}$$

2.2 คำนวณค่า $d(a_{34}, b_{34})$ ได้ดังนี้

$$M_A^3(a_{34}) = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \quad M_B^3(a_{34}) = \begin{bmatrix} 0 & 0.3 & 0.6 \\ 0 & 0.5 & 0.2 \\ 0 & 0.6 & 0 \end{bmatrix}$$

ดังนั้น

$$\begin{aligned} d(a_{34}, b_{34}) &= \frac{((1 \times 0 \times \sqrt{2}) + (1 \times 0.3 \times 1) + (0 \times 0.6 \times \sqrt{2}) + (1 \times 0 \times 1) + 0 + (1 \times 0.2 \times 1) + (0 \times 0 \times \sqrt{2}) + (1 \times 0.6 \times 1) + (1 \times 0 \times \sqrt{2})) \times (1 - 0.5)}{0.3 + 0.5 + 0.6 + 0.6 + 0.2} \\ &= \frac{(0 + 0.3 + 0 + 0 + 0 + 0.2 + 0 + 0.6 + 0) \times 0.5}{2.2} \\ &= \frac{1.1 \times 0.5}{2.2} \\ &= 0.25 \end{aligned}$$

2.3 คำนวณค่า $d(a_{25}, b_{25})$ ได้ดังนี้

$$M_A^3(a_{25}) = \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 0 & 1 \\ \hline 0 & 1 & 1 \\ \hline \end{array}$$

$$M_B^3(a_{25}) = \begin{array}{|c|c|c|} \hline 0 & 0.6 & 0 \\ \hline 0.3 & 0.6 & 0 \\ \hline 0.5 & 0.2 & 0 \\ \hline \end{array}$$

ดังนั้น

$$\begin{aligned} d(a_{25}, b_{25}) &= \frac{((1 \times 0 \times \sqrt{2}) + (1 \times 0.6 \times 1) + (1 \times 0 \times \sqrt{2}) + (1 \times 0.3 \times 1) + 0 + (1 \times 0 \times 1) + (0 \times 0.5 \times \sqrt{2}) + (1 \times 0.2 \times 1) + (1 \times 0 \times \sqrt{2})) \times (1 - 0.6)}{0.6 + 0.3 + 0.6 + 0.5 + 0.2} \\ &= \frac{(0 + 0.6 + 0 + 0.3 + 0 + 0 + 0 + 0.2 + 0) \times 0.4}{2.2} \\ &= \frac{0.44}{2.2} \\ &= 0.2 \end{aligned}$$

ขั้นตอนที่ 3 คำนวณค่า P_r และ P_c ถ้ากำหนดให้ $r_1 = 0.5$ และ $r_2 = 0.5$

$$\begin{aligned} P_r &= r_1 \times |N_r^A - N_r^B| \\ &= 0.5 \times |3 - 5| \\ &= 1 \end{aligned}$$

$$\begin{aligned} P_c &= r_2 \times |N_c^A - N_c^B| \\ &= 0.5 \times |3 - 2| \\ &= 0.5 \end{aligned}$$

ขั้นตอนที่ 4 คำนวณค่า $D(A, B)$

$$\begin{aligned} D(A, B) &= \left(\frac{1}{N_A} \sum_{a \in A} d(a, b) \right) + P_r + P_c \\ &= \left(\frac{0.7316 + 0.25 + 0.2}{3} \right) + 1 + 0.5 \\ &= \left(\frac{1.1816}{3} \right) + 1.5 \\ &= 1.8939 \end{aligned}$$

ดังนั้นค่าระยะทางหรือค่าความเหมือนระหว่างเซต A ข้อมูลอินพุตบล็อกที่ s และเซต B
ค่าน้ำหนักของโหนดหนึ่งในเซกเมนต์เดียวกันมีค่า $D(A, B) = 1.8939$

บทที่ 4

การทดสอบการประยุกต์ใช้เมตริกซ์ค่าน้ำหนักร่วมกับ แบบจำลองโครงข่ายประสาทเทียมแบบลำดับชั้นสำหรับการรู้จำ ตัวอักษรลายมือเขียนภาษาไทย

4.1 ข้อมูลทดสอบระบบ

ตัวอักษรพยัญชนะไทยประกอบด้วย 44 ตัวคือตั้งแต่ ก ถึง ฮ แต่ในปัจจุบันนี้ไม่มีการใช้พยัญชนะตัวที่ 3 และตัวที่ 5 คือ “จ” และ “ค” ตามลำดับ ดังนั้นจึงเหลือพยัญชนะที่ใช้อยู่จริงเพียง 42 ตัว ด้วยเหตุนี้การทดสอบระบบการรู้จำตัวอักษรลายมือเขียนของงานวิจัยนี้จึงทดลองกับตัวอักษรพยัญชนะจำนวน 42 ตัวเท่านั้น

ข้อมูลตัวอักษรลายมือเขียนภาษาไทยที่ใช้ในการทดสอบประสิทธิภาพของการประยุกต์ใช้เมตริกซ์ค่าน้ำหนักร่วมกับแบบจำลองโครงข่ายประสาทเทียมแบบลำดับชั้นเป็นข้อมูลลายมือเขียนของฐานข้อมูลศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ ซึ่งข้อมูลดังกล่าวรวบรวมจากผู้เขียนจำนวน 63 คนเป็นเพศชาย 28 คนและเพศหญิง 35 คน อายุเฉลี่ยของกลุ่มผู้เขียนอยู่ระหว่าง 20-29 ปี โดยผู้เขียนแต่ละคนจะเขียนตัวอักษรพยัญชนะครั้งละ 1 ตัวอักษรจำนวน 3 ครั้ง ดังนั้นผู้เขียนแต่ละคนจะมีข้อมูลลายมือเขียนคนละ 3 เซต รวมมีข้อมูลที่ใช้ในการทดลองทั้งหมดจำนวน $63 \times 42 \times 3 = 7,938$ ตัว

ตารางที่ 4.1 ตัวอย่างตัวอักษรพยัญชนะลายมือเขียนภาษาไทย

ก	ข	ค	ฅ	ง	จ	ฉ
ช	ซ	ฌ	ญ	ฉ	ฆ	ฐ
ฑ	ฒ	ณ	ด	ต	ถ	ฒ
ฐ	ฑ	ฒ	ด	ต	ถ	ฒ
ณ	ด	ต	ถ	ฒ	ฑ	ฐ
ด	ต	ถ	ฒ	ฑ	ฐ	ณ

4.2 การกำหนดพารามิเตอร์ของระบบ

ค่าพารามิเตอร์ที่ต้องกำหนดในการทดลองมีดังต่อไปนี้

1. ค่า ρ_{global} เป็นค่าระยะทางหรือค่าความต่างที่ยอมรับได้ระหว่างภาพอินพุตที่เข้ามาเรียนรู้กับภาพโมเดลอ้างอิง จากที่ได้ทำการทดลองควรกำหนดค่านี้ให้มีค่าอยู่ระหว่าง 0 ถึง 1
2. ค่า ρ_{local} เป็นค่าระยะทางหรือค่าความต่างที่ยอมรับได้ระหว่างบล็อกภาพอินพุตที่เข้ามาเรียนรู้กับบล็อกภาพโมเดลอ้างอิง จากที่ได้ทำการทดลองควรกำหนดค่านี้ให้มีค่าอยู่ระหว่าง 0 ถึง 0.1
3. ค่าพารามิเตอร์ในฟังก์ชันการปรับค่าน้ำหนัก

$$w_{ji}^s(t+1) = \begin{cases} C & , R_p \cap \bar{R}_w \\ w_{ji}^s(t) + c_1 & , R_p \cap R_w \\ w_{ji}^s(t) - c_2 & , \bar{R}_p \cap R_w \\ 0 & , \bar{R}_p \cap \bar{R}_w \end{cases} \quad (4.1)$$

ค่า C คือค่าน้ำหนักเริ่มต้นจะกำหนดให้มีค่าอยู่ระหว่างค่า 0 ถึง 1

ค่า c_1, c_2 คือค่าคงที่เพื่อเพิ่มและลดระดับความสำคัญค่าน้ำหนักตามลำดับ จะกำหนดให้ให้มีค่าอยู่ระหว่าง 0 ถึง 1

4. กำหนดขนาดของยูคลิเดียนเมตริกซ์ที่ใช้การการวัดค่าความเหมือนระหว่างภาพอินพุตที่เข้ามาเรียนรู้กับภาพโมเดลอ้างอิง ในที่นี้จะกำหนดให้ยูคลิเดียนเมตริกซ์มีขนาด 3×3
5. ค่าพารามิเตอร์ของสมการหาค่าความต่างจำนวนแถว (penalty of row)

$$P_r = r_1 \times |N_r^A - N_r^B| \quad (4.2)$$

ค่า r_1 คือค่าสัมประสิทธิ์ของค่าความต่างจำนวนแถว กำหนดให้มีค่าเท่ากับ 0.5

6. ค่าพารามิเตอร์ของสมการหาค่าความต่างจำนวนคอลัมน์ (penalty of column)

$$P_c = r_2 \times |N_c^A - N_c^B| \quad (4.3)$$

ค่า r_2 คือค่าสัมประสิทธิ์ของค่าความต่างจำนวนคอลัมน์ กำหนดให้มีค่าเท่ากับ 0.5

4.3 ผลการทดลอง

การทดลองในงานวิจัยนี้ประกอบด้วย 3 ชุดการทดลอง คือ

การทดลองที่ 1 การทดลองเพื่อทดสอบการรู้จำลายมือเขียนที่ขึ้นอยู่กับผู้เขียน โดยในการทดสอบจะทำการเปรียบเทียบประสิทธิภาพในการทำงานของอัลกอริธึมที่นำเสนอกับงานวิจัยของ Arit Thammano และ Songpol Ruensuk [17] ที่ได้มีการนำการวัดระยะทางแบบฮาฮูสโครฟดิสแทนซ์มาใช้ร่วมกับพีชชีอาร์ทแมพ โดยพัฒนาให้มีการหาตัวแทนของแต่ละคลาสก่อนขั้นตอนการเรียนรู้ ตามตารางผลการทดลองที่ 4.2 จะใช้สัญลักษณ์ WM และ MHD แทนเทคนิควิธีการวัดระยะทางจากเมตริกซ์ค่าน้ำหนักที่งานวิจัยนี้เสนอและงานวิจัยที่นำมาเปรียบเทียบตามลำดับ

การทดลองที่ 2 เป็นการทดลองเพื่อทดสอบการรู้จำลายมือเขียนที่ไม่ขึ้นอยู่กับผู้เขียน โดยการทดสอบจะทำการเปรียบเทียบประสิทธิภาพในการทำงานของอัลกอริทึมที่นำเสนอกับงานวิจัยของ Arit Thammano และ Songpol Ruensuk [17] ตามตารางผลการทดลองที่ 4.3 จะใช้สัญลักษณ์ WM และ MHD แทนเทคนิควิธีการวัดระยะทางจากเมตริกซ์ค่าน้ำหนักที่งานวิจัยนี้นำเสนอและงานวิจัยที่นำมาเปรียบเทียบตามลำดับ

การทดลองที่ 3 เป็นการทดลองเพื่อใช้ทดสอบสมมติฐานที่ว่าจำนวนการแบ่งบล็อกภาพที่แตกต่างกันของภาพตัวอักษรที่นำมาใช้ในการทดลองมีผลต่อประสิทธิภาพในการรู้จำตัวอักษรหรือไม่ ในการทดสอบสมมติฐานนี้ได้มีการกำหนดจำนวนการแบ่งบล็อกภาพเป็น 4, 9 และ 16 บล็อก

4.3.1 การทดลองที่ 1 : การทดสอบการรู้จำตัวอักษรลายมือเขียนที่ขึ้นอยู่กับผู้เขียน

ข้อมูลลายมือเขียนแบบออฟไลน์ของฐานข้อมูลศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติได้เก็บรวบรวมจากผู้เขียนจำนวน 63 คน ซึ่งแต่ละคนจะมีข้อมูลลายมือเขียนคนละ 3 เซต การเตรียมการทดลองในชุดนี้จะแบ่งข้อมูลลายมือเขียนจำนวน 2 เซตของผู้เขียนทั้งหมด 63 คนเป็นข้อมูลสำหรับการเรียนรู้และข้อมูลที่เหลืออีก 1 เซตไว้สำหรับขั้นตอนการทดสอบ ดังนั้นจึงมีข้อมูลที่ใช้ในการเรียนรู้จำนวน 5,292 ตัวและมีข้อมูลที่ใช้ในการทดสอบจำนวน 2,646 ตัว

การเตรียมชุดข้อมูลสำหรับการทดลองเพื่อทดสอบการรู้จำตัวอักษรลายมือเขียนที่ขึ้นอยู่กับผู้เขียน ได้จัดชุดข้อมูลไว้ 3 ชุดดังนี้

ข้อมูลชุดที่ 123 จะแบ่งข้อมูลลายมือเขียนเซตที่ 1 และเซตที่ 2 เป็นข้อมูลสำหรับการเรียนรู้และข้อมูลลายมือเขียนเซตที่ 3 เป็นข้อมูลสำหรับการทดสอบ

ข้อมูลชุดที่ 132 จะแบ่งข้อมูลลายมือเขียนเซตที่ 1 และเซตที่ 3 เป็นข้อมูลสำหรับการเรียนรู้และข้อมูลลายมือเขียนเซตที่ 2 เป็นข้อมูลสำหรับการทดสอบ

ข้อมูลชุดที่ 231 จะแบ่งข้อมูลลายมือเขียนเซตที่ 2 และเซตที่ 3 เป็นข้อมูลสำหรับการเรียนรู้และข้อมูลลายมือเขียนเซตที่ 1 เป็นข้อมูลสำหรับการทดสอบ

ตารางที่ 4.2 ผลการทดลองการรู้จำตัวอักษรลายมือเขียนที่ขึ้นอยู่กับผู้เขียน

	% ความถูกต้องในการรู้จำ		
	MHD	WM (ไม่มีการแบ่งบล็อกภาพ)	WM (แบ่ง 9 บล็อกภาพ)
ข้อมูลชุดที่ 123	62.59%	73.96%	77.55%
ข้อมูลชุดที่ 132	61.68%	75.32%	78.08%
ข้อมูลชุดที่ 231	59.26%	73.39%	77.10%

ข้อมูลชุดที่ 123 จากการทดลองสามารถสรุปได้ว่าเทคนิคใหม่ที่น่าเสนอโดยไม่มีการแบ่งบล็อกภาพให้ความถูกต้องในการรู้จำ 73.96% และใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 1,189,350 พิกเซล ($15 \times 15 \times 5,286$) โดยมีการกำหนดค่าพารามิเตอร์ ρ_{global} , C , c_1 และ c_2 เท่ากับ 0.01, 0.5, 0.02 และ 0.02 ตามลำดับและการทดสอบเทคนิคใหม่ที่น่าเสนอโดยการแบ่งภาพเป็น 9 บล็อกให้ความถูกต้องในการรู้จำ 77.55% และใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 423,025 พิกเซล ($5 \times 5 \times 16,921$) โดยมีการกำหนดค่าพารามิเตอร์ ρ_{local} , ρ_{global} , C , c_1 และ c_2 เท่ากับ 0.01, 0.1, 0.5, 0.02 และ 0.02 ตามลำดับ ส่วนงานวิจัยที่นำมาเปรียบเทียบให้ความถูกต้องในการรู้จำ 62.59% และใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 1,188,900 พิกเซล ($15 \times 15 \times 5,284$)

ข้อมูลชุดที่ 132 จากการทดลองสามารถสรุปได้ว่าเทคนิคใหม่ที่น่าเสนอโดยไม่มีการแบ่งบล็อกภาพให้ความถูกต้องในการรู้จำ 75.32% และใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 1,190,700 พิกเซล ($15 \times 15 \times 5,292$) โดยมีการกำหนดค่าพารามิเตอร์ ρ_{global} , C , c_1 และ c_2 เท่ากับ 0.01, 0.5, 0.02 และ 0.02 ตามลำดับและการทดสอบเทคนิคใหม่ที่น่าเสนอโดยการแบ่งภาพเป็น 9 บล็อกให้ความถูกต้องในการรู้จำ 78.08% และใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 422,800 พิกเซล ($5 \times 5 \times 16,912$) โดยมีการกำหนดค่าพารามิเตอร์ ρ_{local} , ρ_{global} , C , c_1 และ c_2 เท่ากับ 0.01, 0.1, 0.5, 0.02 และ 0.02 ตามลำดับ ส่วนงานวิจัยที่นำมาเปรียบเทียบให้ความถูกต้องในการรู้จำ 61.68% และใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 1,189,800 พิกเซล ($15 \times 15 \times 5,288$)

ข้อมูลชุดที่ 231 จากการทดลองสามารถสรุปได้ว่าเทคนิคใหม่ที่น่าเสนอโดยไม่มีการแบ่งบล็อกภาพให้ความถูกต้องในการรู้จำ 73.39% และใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 1,190,475 พิกเซล ($15 \times 15 \times 5,291$) โดยมีการกำหนดค่าพารามิเตอร์ ρ_{global} , C , c_1 และ c_2 เท่ากับ 0.01, 0.5, 0.02 และ 0.02 ตามลำดับและการทดสอบเทคนิคใหม่ที่น่าเสนอโดยการแบ่งภาพเป็น 9 บล็อกให้ความถูกต้องในการรู้จำ 77.10% และใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 426,325 พิกเซล ($5 \times 5 \times 17,053$) โดยมีการกำหนดค่าพารามิเตอร์ ρ_{local} , ρ_{global} , C , c_1 และ c_2 เท่ากับ 0.01, 0.1, 0.5, 0.02 และ 0.02 ตามลำดับ ส่วนงานวิจัยที่นำมาเปรียบเทียบให้ความถูกต้องในการรู้จำ 59.26% และใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 1,190,250 พิกเซล ($15 \times 15 \times 5,290$)

4.3.2 การทดลองที่ 2 : การทดสอบการรู้จำตัวอักษรลายมือเขียนที่ไม่ขึ้นอยู่กับผู้เขียน

การทดลองเพื่อทดสอบการรู้จำตัวอักษรลายมือเขียนโดยไม่ขึ้นอยู่กับผู้เขียนนี้ได้แบ่งข้อมูลลายมือเขียนของผู้เขียนจำนวน 63 คนเป็น 2 กลุ่มดังนี้ ชุดข้อมูลที่ใช้ในการเรียนรู้ประกอบด้วยข้อมูลลายมือเขียนทั้ง 3 เซตของผู้เขียนจำนวน 42 คน ในขณะที่ข้อมูลที่ใช้ในการทดสอบเป็นข้อมูลลายมือเขียนของผู้เขียนที่เหลืออีก 21 คนครบทั้ง 3 เซตเช่นกัน ดังนั้นจะมีข้อมูลที่ใช้ในการเรียนรู้จำนวน 5,292 ตัวและข้อมูลที่ใช้ในการทดสอบจำนวน 2,646 ตัว

การเตรียมชุดข้อมูลสำหรับการทดสอบการรู้จำตัวอักษรลายมือเขียนที่ไม่ขึ้นอยู่กับผู้เขียน แต่ละชุดจะมีการสุ่มลำดับของผู้เขียนทั้งข้อมูลที่ใช้ในการเรียนรู้และการทดสอบ โดยแต่ละชุดข้อมูลมีรายละเอียดข้อมูลชุดดังนี้

ข้อมูลชุดที่ 1

ข้อมูลสำหรับการเรียนรู้ประกอบด้วยลายมือเขียนของผู้เขียนลำดับที่ 45, 15, 31, 9, 32, 49, 21, 57, 41, 52, 27, 3, 33, 28, 11, 38, 13, 16, 51, 34, 7, 50, 18, 60, 55, 35, 39, 17, 5, 30, 10, 19, 44, 37, 40, 2, 47, 20, 61, 29, 63 และ 26

ข้อมูลสำหรับการทดสอบประกอบด้วยลายมือเขียนของผู้เขียนลำดับที่ 14, 48, 6, 23, 59, 24, 46, 25, 42, 53, 54, 12, 58, 4, 22, 36, 1, 62, 43, 56 และ 8

ข้อมูลชุดที่ 2

ข้อมูลสำหรับการเรียนรู้ประกอบด้วยลายมือเขียนของผู้เขียนลำดับที่ 60, 39, 10, 31, 48, 45, 29, 22, 59, 44, 35, 33, 58, 16, 7, 30, 20, 51, 5, 46, 14, 49, 13, 61, 42, 53, 25, 9, 28, 26, 37, 6, 21, 52, 2, 34, 24, 32, 40, 57, 54 และ 18

ข้อมูลสำหรับการทดสอบประกอบด้วยลายมือเขียนของผู้เขียนลำดับที่ 1, 4, 11, 56, 36, 62, 38, 19, 23, 27, 43, 12, 15, 3, 47, 17, 41, 55, 8, 63 และ 50

ข้อมูลชุดที่ 3

ข้อมูลสำหรับการเรียนรู้ประกอบด้วยลายมือเขียนของผู้เขียนลำดับที่ 30, 22, 1, 63, 29, 37, 28, 49, 53, 52, 16, 14, 38, 62, 10, 31, 5, 7, 60, 40, 25, 23, 42, 39, 61, 33, 45, 36, 32, 17, 4, 35, 11, 12, 3, 18, 15, 43, 56, 13, 9 และ 58

ข้อมูลสำหรับการทดสอบประกอบด้วยลายมือเขียนของผู้เขียนลำดับที่ 20, 27, 19, 34, 55, 8, 57, 54, 41, 24, 50, 2, 6, 47, 21, 59, 51, 48, 26, 46 และ 44

ตารางที่ 4.3 ผลการทดลองการรู้จำตัวอักษรลายมือเขียน โดยไม่ขึ้นอยู่กับผู้เขียน

	% ความถูกต้องในการรู้จำ		
	MHD	WM (ไม่มีการแบ่งบล็อกภาพ)	WM (แบ่ง 9 บล็อกภาพ)
ข้อมูลชุดที่ 1	56.69%	69.95%	73.28%
ข้อมูลชุดที่ 2	59.30%	69.58%	73.70%
ข้อมูลชุดที่ 3	58.01%	70.48%	73.43%

ข้อมูลชุดที่ 1 จากผลการทดลองพบว่า เทคนิคใหม่ที่น่าเสนอโดยไม่มีการแบ่งบล็อกภาพให้ความถูกต้องในการรู้จำ 69.95% และได้ใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 1,189,575 พิกเซล (15 x 15 x 5,287) โดยมีการกำหนดค่าพารามิเตอร์ ρ_{global} , C, c_1 และ c_2 เท่ากับ 0.01, 0.5, 0.02

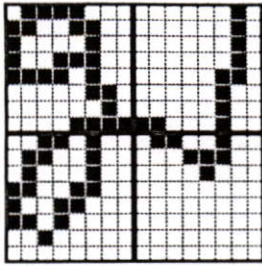
และ 0.02 ตามลำดับและการทดสอบเทคนิคใหม่ที่น่าเสนอโดยการแบ่งภาพเป็น 9 บล็อกให้ความถูกต้องในการรู้จำ 73.28% และได้ใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 417,925 พิกเซล ($5 \times 5 \times 16,717$) โดยมีการกำหนดค่าพารามิเตอร์ ρ_{local} , ρ_{global} , C , c_1 และ c_2 เท่ากับ 0.01, 0.1, 0.5, 0.02 และ 0.02 ตามลำดับ ส่วนงานวิจัยที่นำมาเปรียบเทียบให้ความถูกต้องในการรู้จำ 56.69% และใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 1,188,900 พิกเซล ($15 \times 15 \times 5,284$)

ข้อมูลสรุปชุดที่ 2 จากผลการทดลองพบว่า เทคนิคใหม่ที่น่าเสนอโดยไม่มี การแบ่งบล็อกภาพให้ความถูกต้องในการรู้จำ 69.58% และได้ใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 1,189,350 พิกเซล ($15 \times 15 \times 5,286$) โดยมีการกำหนดค่าพารามิเตอร์ ρ_{global} , C , c_1 และ c_2 เท่ากับ 0.01, 0.5, 0.02 และ 0.02 ตามลำดับและการทดสอบเทคนิคใหม่ที่น่าเสนอโดยการแบ่งภาพเป็น 9 บล็อกให้ความถูกต้องในการรู้จำ 73.70% และได้ใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 420,700 พิกเซล ($5 \times 5 \times 16,828$) โดยมีการกำหนดค่าพารามิเตอร์ ρ_{local} , ρ_{global} , C , c_1 และ c_2 เท่ากับ 0.01, 0.1, 0.5, 0.02 และ 0.02 ตามลำดับ ส่วนงานวิจัยที่นำมาเปรียบเทียบให้ความถูกต้องในการรู้จำ 59.30% และใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 1,188,675 พิกเซล ($15 \times 15 \times 5,283$)

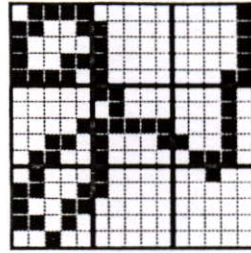
ข้อมูลสรุปชุดที่ 3 จากผลการทดลองพบว่า เทคนิคใหม่ที่น่าเสนอโดยไม่มี การแบ่งบล็อกภาพให้ความถูกต้องในการรู้จำ 70.48% และได้ใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 1,189,125 พิกเซล ($15 \times 15 \times 5,285$) โดยมีการกำหนดค่าพารามิเตอร์ ρ_{global} , C , c_1 และ c_2 เท่ากับ 0.01, 0.5, 0.02 และ 0.02 ตามลำดับและการทดสอบเทคนิคใหม่ที่น่าเสนอโดยการแบ่งภาพเป็น 9 บล็อกให้ความถูกต้องในการรู้จำ 73.43% และได้ใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 421,600 พิกเซล ($5 \times 5 \times 16,864$) โดยมีการกำหนดค่าพารามิเตอร์ ρ_{local} , ρ_{global} , C , c_1 และ c_2 เท่ากับ 0.01, 0.1, 0.5, 0.02 และ 0.02 ตามลำดับ ส่วนงานวิจัยที่นำมาเปรียบเทียบให้ความถูกต้องในการรู้จำ 58.01% และใช้พื้นที่ในการจัดเก็บข้อมูลจำนวน 1,188,675 พิกเซล ($15 \times 15 \times 5,283$)

4.3.3 การทดลองที่ 3 : การทดสอบจำนวนบล็อกของการแบ่งบล็อกภาพตัวอักษร

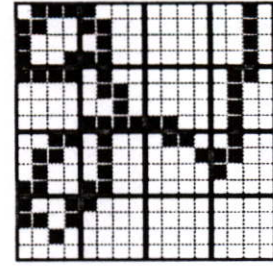
สถาปัตยกรรมโครงข่ายประสาทเทียมแบบลำดับชั้นที่น่าเสนอประกอบด้วย 4 ชั้นคือชั้นอินพุท ชั้นซ่อน ชั้นคลัสเตอร์และชั้นเอาต์พุทตามลำดับ โดยชั้นอินพุทและชั้นซ่อนจะมีการแบ่งเป็นเซกเมนต์ โดยจำนวนของเซกเมนต์ของทั้งสองชั้นนี้จะสอดคล้องกับจำนวนบล็อกของการแบ่งบล็อกภาพตัวอักษร งานวิจัยนี้ต้องการทดสอบสมมติฐานว่าจำนวนบล็อกของการแบ่งบล็อกภาพที่แตกต่างกันของภาพตัวอักษรที่นำมาใช้ในการทดลองมีผลต่อประสิทธิภาพในการรู้จำตัวอักษรหรือไม่ ในการทดสอบสมมติฐานนี้ได้มีการกำหนดจำนวนการแบ่งบล็อกภาพเป็น 4, 9 และ 16 บล็อก ดังรูปที่ 4.1



(a) การแบ่งภาพ 4 บล็อก



(b) การแบ่งภาพ 9 บล็อก



(c) การแบ่งภาพ 16 บล็อก

รูปที่ 4.1 การแบ่งภาพตัวอักษรขนาด 4, 9 และ 16 บล็อก

การแบ่งภาพตัวอักษรขนาด 15×15 พิกเซลเป็น 4 บล็อกภาพจะได้ขนาดบล็อกไม่เท่ากัน จึงแก้ไขโดยการเพิ่มแถวและคอลัมน์ของจุดภาพขาวจำนวนอย่างละ 1 แถว ดังนั้นบล็อกภาพที่ได้ จึงมีขนาด 8×8 พิกเซลและสถาปัตยกรรมของโครงข่ายสำหรับการเรียนรู้ตัวอักษรที่ถูกแบ่งเป็น 4 บล็อกจะประกอบด้วยชั้นอินพุตและชั้นซ่อนที่มี 4 เซกเมนต์ โดยแต่ละเซกเมนต์ในชั้นอินพุตจะมีจำนวนโหนดคงที่คือเซกเมนต์ละ 64 โหนด ส่วนชั้นซ่อนถึงแม้จะมีการแบ่งเซกเมนต์เช่นเดียวกับชั้นอินพุตแต่จำนวนโหนดของชั้นนี้ไม่คงที่ขึ้นอยู่กับข้อกำหนดค่าพารามิเตอร์ที่ใช้ในการเรียนรู้รวมทั้งโหนดของชั้นคลัสเตอร์ด้วย และชั้นเอาต์พุตจะมีจำนวนโหนด 42 โหนด

การแบ่งภาพตัวอักษรเป็น 9 บล็อกที่ขนาดเท่าๆ กันจะได้บล็อกภาพที่มีขนาด 5×5 พิกเซล และสถาปัตยกรรมของโครงข่ายที่ใช้ในการเรียนรู้จะประกอบด้วยชั้นอินพุตและชั้นซ่อนที่ถูกแบ่งเป็น 9 เซกเมนต์ โดยแต่ละเซกเมนต์ของชั้นอินพุตจะมี 25 โหนด ส่วนชั้นซ่อน ชั้นคลัสเตอร์ และชั้นเอาต์พุตจะเหมือนกับสถาปัตยกรรมของการแบ่งบล็อกภาพขนาด 4 บล็อก

การแบ่งภาพตัวอักษรเป็น 16 บล็อกต้องมีการดำเนินการเพิ่มแถวและคอลัมน์ของจุดภาพขาวเช่นเดียวกับการแบ่งภาพเป็น 4 บล็อก ดังนั้นแต่ละบล็อกภาพที่ได้จึงมีขนาด 4×4 พิกเซล และสถาปัตยกรรมของโครงข่ายจะมีชั้นอินพุตและชั้นซ่อนที่ประกอบด้วย 16 เซกเมนต์โดยแต่ละเซกเมนต์มี 16 โหนด

การเตรียมข้อมูลสำหรับการทดสอบสมมติฐานนี้ประกอบด้วยลายมือเขียนของผู้เขียนจำนวน 15 คนที่มีลำดับผู้เขียนดังนี้ 3, 5, 10, 15, 18, 26, 35, 36, 40, 43, 44, 45, 48, 50 และ 55 โดยแบ่งข้อมูลลายมือเขียนเซตที่ 1 และเซตที่ 3 เป็นข้อมูลสำหรับการเรียนรู้และข้อมูลลายมือเขียนเซตที่ 2 เป็นข้อมูลสำหรับการทดสอบ ดังนั้นจึงมีข้อมูลที่ใช้ในการเรียนรู้จำนวน 1,260 ตัวและมีข้อมูลที่ใช้ในการทดสอบจำนวน 630 ตัว

ตารางที่ 4.4 ผลการทดลองของการแบ่งบล็อกภาพ

จำนวนการแบ่งบล็อกภาพ	4 บล็อก	9 บล็อก	16 บล็อก
จำนวนโหนดชั้นซ่อน	3,851	5,344	4,498
จำนวนชั้นคลัสเตอร์	1,252	1,253	1,142
เวลาเฉลี่ยในการเรียนรู้ต่อข้อมูล (วินาที)	2.40	3.06	2.77
เวลาเฉลี่ยในการทดสอบต่อข้อมูล (วินาที)	4.63	5.55	4.57
ความถูกต้องในการรู้จำ	78.41%	78.10%	79.68%

จากผลการทดลองพบว่า การแบ่งบล็อกภาพจำนวน 4 บล็อกให้ความถูกต้องในการรู้จำ 78.41% โดยได้มีการกำหนดค่า ρ_{local} , ρ_{global} , C , c_1 และ c_2 เท่ากับ 0.03, 0.1, 0.6, 0.02 และ 0.02 ตามลำดับ

จากผลการทดลองพบว่า การแบ่งบล็อกภาพจำนวน 9 บล็อกให้ความถูกต้องในการรู้จำ 78.10% โดยได้มีการกำหนดค่า ρ_{local} , ρ_{global} , C , c_1 และ c_2 เท่ากับ 0.01, 0.1, 0.7, 0.06 และ 0.06 ตามลำดับ

จากผลการทดลองพบว่า การแบ่งบล็อกภาพจำนวน 16 บล็อกให้ความถูกต้องในการรู้จำ 78.41% โดยได้มีการกำหนดค่า ρ_{local} , ρ_{global} , C , c_1 และ c_2 เท่ากับ 0.01, 0.3, 0.5, 0.02 และ 0.02 ตามลำดับ

ในการทดลองที่ 3 นี้ ได้ทดสอบการแบ่งบล็อกภาพตัวอักษรเป็น 4, 9 และ 16 บล็อกจากการทดลองสามารถสรุปได้ว่าจำนวนการแบ่งบล็อกภาพที่แตกต่างกันให้ประสิทธิภาพในการรู้จำตัวอักษรลายมือเขียนได้ใกล้เคียงกัน โดยการแบ่งบล็อกภาพจำนวน 16 บล็อกจะให้ประสิทธิภาพในการรู้จำดีที่สุด ส่วนเวลาโดยเฉลี่ยที่ใช้ในการเรียนรู้และการทดสอบต่อจำนวนข้อมูลของการแบ่งบล็อกภาพทั้ง 3 แบบอยู่ในช่วงที่ใกล้เคียงกันมาก

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

โดยทั่วไปแล้วตัวอักษรไทยแต่ละตัวจะมีเอกลักษณ์เฉพาะที่แตกต่างกัน อย่างไรก็ตาม ยังคงมีตัวอักษรอีกหลายคู่ที่มีความคล้ายคลึงกันมาก เช่น ช ช ฎ ฏ ท ฑ เป็นต้น ซึ่งตัวอักษรเหล่านี้จะส่งผลให้ประสิทธิภาพในการรู้จำตัวอักษรมีความถูกต้องน้อยลง และยิ่งถ้าเป็นการรู้จำตัวอักษรลายมือเขียนแล้วจะเป็นอุปสรรคอย่างมากเนื่องจากรูปแบบการเขียนของแต่ละคนจะมีลักษณะที่แตกต่างกัน เช่น การเขียนเอียงไปด้านหน้าหรือด้านหลัง การเขียนที่ไม่มีการขมวดหัวเป็นวง การเขียนหางตัวอักษรสั้นหรือยาวแตกต่างกัน เป็นต้น ดังนั้นการนำเสนอเทคนิคการดึงคุณลักษณะเด่นของตัวอักษรเพื่อนำไปใช้ในการจำแนกกลุ่มตัวอักษรถึงแม้จะมีความถูกต้องในการรู้จำสูง แต่ยังคงมีข้อจำกัดสำหรับรูปแบบการเขียนที่หลากหลายดังที่กล่าวมาแล้วข้างต้น ซึ่งอาจจะไม่สามารถนำไปพัฒนาเพื่อนำไปใช้งานจริงได้เต็มที่นัก

ดังนั้นงานวิจัยนี้จึงมุ่งเน้นที่จะแก้ปัญหาข้อจำกัดดังกล่าวโดยการพัฒนาเทคนิคในการรู้จำตัวอักษรลายมือเขียนภาษาไทยโดยไม่มีการคุณลักษณะเด่นของตัวอักษร เพื่อให้ได้วิธีการใหม่ที่สามารถรองรับการเขียนที่หลากหลายรูปแบบมากขึ้นและมีความยืดหยุ่นในการใช้งานกับชุดข้อมูลอื่นๆ ที่เป็นข้อมูลแบบไบนารีเช่น ข้อมูลภาพใบหน้าคน เป็นต้น

การรู้จำตัวอักษรลายมือเขียนภาษาไทยโดยไม่มีการดึงคุณลักษณะเด่นของภาพที่นำเสนอ มีสถาปัตยกรรมของโครงข่ายประสาทเทียมเป็นโครงข่ายประสาทเทียมแบบลำดับชั้นที่มีการประยุกต์ใช้เมตริกซ์ค่าน้ำหนักในการวัดค่าความเหมือนระหว่างภาพอินพุตที่เข้ามาเรียนรู้กับภาพโมเดลอ้างอิง ซึ่งข้อมูลที่ใช้ในการทดสอบระบบเป็นข้อมูลลายมือเขียนแบบออฟไลน์จากฐานข้อมูลของศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติจำนวน 7,938 ตัว โดยในแต่ละการทดลองจะทำการแบ่งข้อมูลออกเป็น 2 ชุดคือชุดข้อมูลที่ใช้ในการเรียนรู้จำนวน 5,292 ตัวและชุดข้อมูลที่ใช้ในการทดสอบจำนวน 2,646 ตัว

การทดลองที่ 1 เป็นการทดลองเพื่อทดสอบการรู้จำลายมือเขียนที่ขึ้นอยู่กับผู้เขียน จากผลการทดลองสรุปได้ว่าการทดสอบเทคนิคใหม่ที่นำเสนอโดยไม่มีการแบ่งบล็อกภาพกับชุดข้อมูลจำนวน 3 ชุดให้ประสิทธิภาพในการรู้จำตัวอักษรลายมือเขียนได้ดีกว่างานวิจัยที่นำมาเปรียบเทียบ โดยให้ความถูกต้องดีขึ้น 18.18%, 22.12% และ 23.85% ตามลำดับ แต่ได้ใช้พื้นที่ในการจัดเก็บข้อมูลมากกว่า 0.04%, 0.08% และ 0.02% ตามลำดับ ส่วนการทดสอบเทคนิคใหม่ที่นำเสนอโดยให้มีการแบ่งภาพเป็น 9 บล็อกกับชุดดังกล่าวยังทำให้ประสิทธิภาพในการรู้จำตัวอักษรลายมือเขียนดีขึ้นจากงานวิจัยที่นำมาเปรียบเทียบถึง 23.91%, 26.59% และ 30.10% ตามลำดับ

และยังช่วยประหยัดพื้นที่ในการจัดเก็บข้อมูลลดลงจากงานวิจัยที่นำมาเปรียบเทียบ 64.42%, 64.47% และ 64.18% ตามลำดับอีกด้วย

การทดลองที่ 2 เป็นการทดลองเพื่อทดสอบการรู้จำลายมือเขียนที่ไม่ขึ้นอยู่กับผู้เขียน จากผลการทดลองสรุปได้ว่า การทดสอบเทคนิคใหม่ที่น่าเสนอโดยไม่มีการแบ่งบล็อกภาพกับข้อมูลกลุ่มชุดที่ 1, ข้อมูลกลุ่มชุดที่ 2 และข้อมูลกลุ่มชุดที่ 3 ให้ประสิทธิภาพในการรู้จำตัวอักษรลายมือเขียนได้ดีกว่างานวิจัยที่นำมาเปรียบเทียบ โดยให้ความถูกต้องดีขึ้น 23.40%, 17.34% และ 21.50% ตามลำดับแต่ใช้พื้นที่ในการจัดเก็บข้อมูลมากกว่า 0.06%, 0.06% และ 0.04% ตามลำดับและเมื่อทำการทดสอบเทคนิคใหม่ที่น่าเสนอโดยไม่มีการแบ่งภาพเป็น 9 บล็อกกับชุดข้อมูลดังกล่าว ยิ่งทำให้ประสิทธิภาพในการรู้จำตัวอักษรลายมือเขียนดีเพิ่มขึ้นจากงานวิจัยที่นำมาเปรียบเทียบ 29.27%, 24.28% และ 26.58% ตามลำดับและยังช่วยประหยัดพื้นที่ในการจัดเก็บข้อมูลลดลงจากงานวิจัยที่นำมาเปรียบเทียบ 64.85%, 64.61% และ 64.53% ตามลำดับอีกด้วย

การทดลองที่ 3 เป็นการทดลองเพื่อทดสอบสมมติฐานที่ว่าจำนวนการแบ่งบล็อกภาพที่แตกต่างกันของภาพตัวอักษรมีผลต่อประสิทธิภาพในการรู้จำตัวอักษรหรือไม่ ในการทดสอบสมมติฐานนี้ได้มีการกำหนดจำนวนการแบ่งบล็อกภาพเป็น 4, 9 และ 16 บล็อกภาพ หลังจากการทดลองแล้วพบว่าจำนวนการแบ่งบล็อกภาพที่แตกต่างกันให้ประสิทธิภาพในการรู้จำตัวอักษรลายมือเขียนได้ใกล้เคียงกันคือ 78.41%, 78.10% และ 79.68% สำหรับการแบ่งบล็อกภาพเป็น 4, 9 และ 16 บล็อกตามลำดับ โดยการแบ่งบล็อกภาพตัวอักษรจำนวน 16 บล็อกให้ประสิทธิภาพในการรู้จำดีที่สุด

จากการศึกษาการประยุกต์ใช้เมตริกซ์ค่าน้ำหนักร่วมกับแบบจำลองโครงข่ายประสาทเทียมแบบลำดับชั้นสำหรับการรู้จำตัวอักษรลายมือเขียนภาษาไทย สามารถสรุปได้ดังต่อไปนี้

1. เป็นพื้นฐานงานวิจัยทางด้านการรู้จำตัวอักษรลายมือเขียนภาษาไทยที่มีประสิทธิภาพในการรู้จำสูงและมีความยืดหยุ่นในการรองรับข้อมูลอื่นๆ ที่เป็นข้อมูลแบบไบนารีได้ด้วย
2. ตัวอักษรไทยแต่ละตัวจะมีบางส่วนของตัวอักษรที่มีลักษณะเด่นเฉพาะที่แตกต่างกัน เช่น ส่วนหัวหรือรูป ส่วนหาง เป็นต้น ในขณะที่เดียวกันก็จะมีบางส่วนของตัวอักษรที่มีความคล้ายคลึงกันมาก ดังนั้นการแบ่งบล็อกภาพตัวอักษรในงานวิจัยนี้จะทำให้บางส่วนของตัวอักษรที่มีคุณลักษณะเหมือนกันหรือคล้ายกันถึงแม้จะอยู่คลาสเดียวกันหรือต่างคลาสนั้นก็สามารถใช้เมตริกซ์ค่าน้ำหนักร่วมกันได้ และบางส่วนของตัวอักษรที่เป็นเอกลักษณ์เฉพาะของตัวอักษรคลาสหนึ่งคลาสนั้นก็จะใช้เมตริกซ์ค่าน้ำหนักร่วมกันได้เฉพาะคลาสนั้นเช่นกัน
3. เมื่อทำการทดลองเปรียบเทียบกับงานวิจัยของ Arit Thammano และ Songpol Ruensuk [17] แล้ว เทคนิคใหม่ที่น่าเสนอช่วยประหยัดพื้นที่ในการจัดเก็บข้อมูลได้อย่างมาก

5.2 ข้อเสนอแนะ

1. การทดลองในงานวิจัยนี้ได้ทดสอบกับข้อมูลตัวอักษรพยัญชนะเท่านั้น เพื่อให้สามารถนำไปพัฒนาซอฟต์แวร์ทางด้านระบบการรู้จำตัวอักษรภาษาไทยเพื่อใช้งานจริงได้จึงควรขยายชุดข้อมูลให้ครอบคลุมมากขึ้น เช่น ข้อมูลตัวเลขไทย ตัวเลขอารบิก สระ วรรณยุกต์และอักขระหรือสัญลักษณ์พิเศษ เป็นต้น
2. สามารถนำไปพัฒนาหรือปรับปรุงใช้กับข้อมูลใบนารีอื่นๆ เช่น ภาพใบหน้าคน ภาพเส้นลายมือ เป็นต้น
3. ควรมีการพัฒนาระบบให้มีประสิทธิภาพในการรู้จำสูงมากยิ่งขึ้นและลดเวลาในการประมวลผลทั้งในขั้นตอนการเรียนรู้และการทดสอบด้วย

บรรณานุกรม

- [1] V. Sornlertlamvanich, T. Potipiti, C. Wutiwiwatchai and P. Mittrapiyanuruk, "**The State of the Art in Thai Language Processing**," Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL2000), Hong Kong, pp. 597-598, Oct 2000.
- [2] T. Kasuba, "**Simplified fuzzy ARTMAP**," AI Expert, pp. 18-25, Nov 1993.
- [3] S. Rajasekaran and G. A. V. Pai, "**Image recognition using Simplified fuzzy ARTMAP augmented with a Moment Based Feature Extractor**," International Journal of Pattern Recognition and Artificial Intelligence, Vol. 14, No. 8, pp. 1081-1095, 2000.
- [4] L. Lam, S. W. Lee and C. Y. Lee, "**Thinning Methodologies - A Comprehensive Survey**," IEEE Transaction Pattern Analysis and Machine Intelligence, Vol. 14, pp. 869-885, 1992.
- [5] T. Y. Zhang and C. Y. Suen, "**A Fast Parallel Algorithm for Thinning Digital Patterns**," Communications of the ACM, Vol. 27, No. 3, pp. 236-239, Mar 1984.
- [6] K. H. Lin, B. Gao, K. M. Lam and W. C. Siu, "**Human face recognition using a spatially weighted modified Hausdorff Distance**," Proceedings of 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing, pp. 477-480, May 2001.
- [7] K. H. Lin, K. M. Lam and W. C. Siu, "**Spatially eigen-weighted Hausdorff Distances for human face recognition**," Pattern Recognition, Vol. 36, No. 8, pp. 1827-1834, 2003.
- [8] M. Blumenstein, B. Verma and H. Basli, "**A Novel Feature Extraction Technique for the Recognition of Segmented Handwritten Characters**," Proceedings of the 7th International Conference on Document Analysis and Recognition, pp. 137-141, 2003.
- [9] P. D. Gader, M. Mohamed and J. H. Chiang, "**Handwritten Word Recognition with character and inter-character Neural Networks**," IEEE Transactions on Systems, Man, Cybernetics, Vol. 27, pp. 158-164, 1997.
- [10] Y. Lu, C. L. Tan, W. Huang, and L. Fan, "**An Approach to Word Image Matching Based on Weighted Hausdorff Distance**," Proceeding of the 6th International Conference on Document Analysis and Recognition, pp. 921-925, 2001.

- [11] S. Airphaboon , M. Sangworasil and S. Kondo, “**Off-line Handwritten Thai Character from Word Script,**” Proceedings of the 12th IAPR International, Vol. 2, pp. 445-449, 1994.
- [12] P. Choruegwiat, J. Jitapankul, L. Wuttisittikulij and P. Seehapen, “**Distinctive Feature Analysis for Thai Handwritten Character Recognition based on modified Stroke Changing Sequence,**” Proceedings of the 1998 IEEE Asia-Pacific Conference on Circuits and Systems (APCCAS '98), pp. 543-546, 1998.
- [13] Gwo-En and Jhing-Fa Wang, “**A New Approach for Recognition of unconstrained Handwritten Numberal,**” IEEE TENCON, 1993.
- [14] S. Airphaboon and M. Sangworasil, “**Recognition of Handwritten Thai Character Considering The Head of Character,**” IEEE International Conference on Image Processing, Singapore, 1989.
- [15] C. Tanprasert and T. Koanantakool, “**Thai OCR : A Neural Network Application,**” IEEE TENCON. Digital Signal Processing Applications, Vol. 1, pp. 90-95, Nov 1996.
- [16] [online] available : <http://encyclopedia.thefreedictionary.com/Matrix+product>
- [17] A. Thammano and S. Ruensuk , “**Human face Recognition Using Modified Huasdoroff ARTMAP,**” International Conference on Intelligent Computing (ICIC 2005), China, Aug 2005.

ภาคผนวก

ภาคผนวก ก.

ตัวอย่างข้อมูลตัวอักษรที่ใช้ในการทดสอบโปรแกรม

ภาคผนวก ข.

ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่

1. A. Thammano and P. Poolsamran, “**Applying A Weighting Matrix to A Hierarchical Network Model for Handwritten Thai Character Recognition,**” the proceedings of the International Conference on Computational Intelligence for Modeling, Control and Automation (CIMCA 2006), Sydney, Australia, 2006.

**International Conference on Computational Intelligence for Modelling,
Control and Automation - CIMCA06**

Jointly with

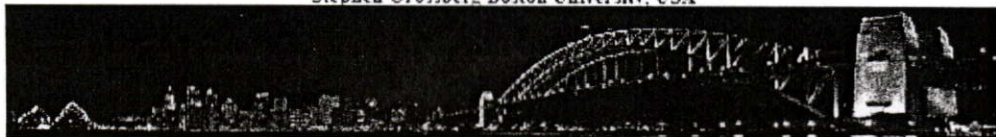
**2006 International Conference on Intelligent Agents, Web Technologies
and Internet Commerce - IAWTIC06**

29 November to 1 December 2006

Sydney, Australia

Honorary Chairs: Lotfi A. Zadeh University of California, USA

Stephen Grossberg Boston University, USA



[Home](#) | [Call for papers](#) | [Important Dates](#) | [Paper Template](#) | [IEEE Computer Society](#)

[Home](#)

[Important Dates](#)

[Preliminary Program
of CIMCA'2006](#)

[Programme
Committee](#)

[Organising Committee](#)

[Registration form](#)

[Keynote speakers](#)

[Tutorials](#)

[Collaborators](#)

[Sponsors](#)

[Accommodation](#)

[Contact Us](#)

[Local Attractions](#)

Call for papers-

Conference Proceedings will be published as books by IEEE (The Institute of Electrical and Electronic Engineering) in USA and it will be indexed by EI and it will also appear in IEEE explore. All papers will be peer reviewed by at least two reviewers.

Papers will be selected based on their originality, significance, correctness, and clarity of presentation. Papers (6 pages maximum) should be submitted to the following e-mail address:

For e-mail submission of your paper/s, send your paper/s to: cimca@canberra.edu.au

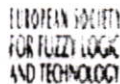
For postal submission of your paper, send your paper/s to:

CIMCA'2006 Secretariat
School of Information Science & Engineering
University of Canberra
ACT, Canberra, 2614, Australia

Papers should present original work, which has not been published or being reviewed for other conferences. Papers should be written in English. The name and affiliation of authors should be omitted on the paper. A separate page must be included with each paper, containing, the names, affiliations, postal address and e-mail addresses of authors as well as the address of contact author. E-mail submission of papers are encouraged.

Important Dates

- 7 August 2006 Notification of acceptance
- 28 August 2006 Deadline for camera-ready copies of accepted papers
- 28-30 November 2006 Conference sessions



Applying a Weighting Matrix to the Hierarchical Neural Network Model for Handwritten Thai Character Recognition

Arit Thammano and Patcharawadee Poolsamran
 Computational Intelligence Laboratory
 Faculty of Information Technology
 King Mongkut's Institute of Technology Ladkrabang
 Bangkok, 10520 Thailand

Phone: (662)737-2551 to 4 ext. 529, Fax: (662)326-4332

E-mail: arit@it.kmitl.ac.th and patcharp@bua.ac.th

Abstract

This paper proposes a new neural network approach to the off-line handwritten Thai character recognition. This new neural network is a hierarchical neural network; it employs the concept of a weighting matrix in measuring the similarity between the incoming input pattern and the reference patterns. The experiments have been conducted to recognize both slipshod and proper handwritten characters. The results demonstrate a very promising performance of the proposed approach.

1. Introduction

Handwritten character recognition problem has long been identified as the problem that computer can hardly solve; however, human-beings take this for granted. There has been substantial research undertaken to solve this problem. The techniques which are widely used for the handwritten character recognition are statistical approaches [1, 2, 3, 4], structural matching [5], neural networks [6, 7, 8, 9, 10], and the hybrid approaches [11, 12].

This research focuses on the off-line Thai handwritten character recognition. Even though many researches have been conducted on the recognition of Thai characters since the past decade, the success in recognizing Thai handwritten characters is still limited, compared to English characters. This is due to the following reasons: (a) there are a large number of characters in Thai language, each of which has a complicated structure; (b) many Thai characters look very alike, e.g. ก-ก-ก, ข-ข-ข, ฃ-ฃ, ฅ-ฅ, ๖-๖, ๗-๗, ๘-๘; (c) there is a great deal of style variation among the handwritten characters.

Typically, any capable pattern recognition system requires an efficient combination of a feature extraction process and a classifier [1]. However, the intention of this research is to obtain the system that can recognize the handwritten characters straight from the character images without any feature extraction process. The off-line handwritten Thai character recognition system proposed in this paper employs the concept of the hierarchical neural network and the weighting matrix. The performance of the proposed network is evaluated against the modified Hausdorff ARTMAP neural network [13]. The data set used in the evaluation process is from the National Electronic and Computer Technology Center (NECTEC). This data set contains both the slipshod and proper handwriting.

Following this introduction, section 2 describes the architecture of the proposed model and its learning algorithm. The preprocessing of the character images is described in section 3. In section 4, the experimental results are demonstrated and discussed. Finally, section 5 is the conclusion.

2. The proposed model

The proposed model is a four-layer feedforward neural network as shown in Figure 1. The first layer is the input layer, which consists of 9 segments. Each segment contains N input nodes. The number of input nodes in a segment is associated with the number of pixels in that segment of the character image. For example, if the size of the character image is 15×15 pixels, each segment will consist of 25 input nodes (5×5 pixels). The second layer is the hidden layer. Each node in the s^{th} segment of the second layer is fully connected to the input nodes of the same segment

via the connections w_{ji}^s . The weight vector w_j^s of dimension N represents the reference pattern of the j^{th} node in the s^{th} segment of the hidden layer. The third layer is the cluster layer. The nodes in the second and the third layers are constructed during the training process. The fourth layer is the output layer. Each node in the output layer represents a class of character. During the supervised learning, the input vector is presented to the model, together with its respective target output vector. The input vector is denoted by $I = \{I^1, \dots, I^s, \dots, I^p\} = \{(I_1^1, \dots, I_N^1), \dots, (I_1^p, \dots, I_N^p)\}$, where s is the s^{th} segment in the input layer. Once the model receives the input and its associated target output (I, O) , the output of the j^{th} node in the s^{th} segment of the hidden layer, which is the distance between the input vector of the s^{th} segment (I^s) and the weight vector of the same segment (w_j^s) , is computed.

$$T_j^s(I^s) = D(I^s, w_j^s) \quad (1)$$

$$D(A, B) = \left[\frac{1}{\|A\|} \sum_{a \in A} d(a, B) \right] + P_r + P_c \quad (2)$$

$$d(a, B) = \begin{cases} \frac{\|M_B^a(a) \cdot M_A^a(a) \cdot E^a\| (1 - b_{ij})}{\|M_B^a(a)\|} \cdot \|M_B^a(a)\| > 0 \\ C. \quad \text{Otherwise} \end{cases} \quad (3)$$

$$P_r = r_1 \times |N_A^r - N_B^r| \quad (4)$$

$$P_c = r_2 \times |N_A^c - N_B^c| \quad (5)$$

where $\|A\|$ is the summation of all elements in matrix A.

$A \cdot B$ is the Hadamard product between matrix A and matrix B.

$M_A^a(a)$ is the $n \times n$ neighborhood of point a in set A.

A.

$M_B^a(a)$ is the $n \times n$ neighborhood of point a in set B.

B.

E^a is the $n \times n$ matrix that contains the Euclidean distances between point a and its neighbors. For example,

$$E^a = \begin{bmatrix} \sqrt{2} & 1 & \sqrt{2} \\ 1 & 0 & 1 \\ \sqrt{2} & 1 & \sqrt{2} \end{bmatrix}$$

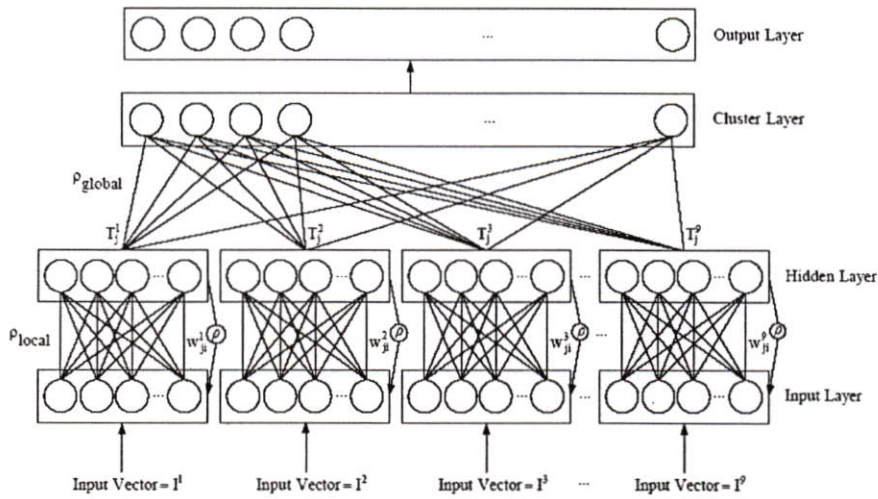


Figure 1. Architecture of the hierarchical neural network

b_j is the weight value of point a in set B .
 P_r and P_c are the row- and column-penalty function respectively.

N_A^r is the number of rows in A that contain at least one black pixel.

N_B^r is the number of rows in B that contain at least one black pixel.

N_A^c is the number of columns in A that contain at least one black pixel.

N_B^c is the number of columns in B that contain at least one black pixel.

r_1 and r_2 are the weight coefficients of the row- and column-penalty function respectively.

Next, the output of each k^{th} node in the cluster layer is calculated as follows:

$$O_k = \frac{1}{9} \sum_{i=1}^9 T_j^i w_{kj}^i \quad (6)$$

where w_{kj}^i is the weight of the connection between the cluster node k and the hidden node j . It is either 0 or 1.

The system then makes a cluster choice by selecting the winning node K with minimum choice function value, among all the nodes k in the cluster layer.

$$K = \arg \min_k \{O_k\} \quad (7)$$

Next, the output value of the winning node (O_K) is compared to the global vigilance (ρ_{global}). ρ_{global} has a value between 0 and 1. If O_K is less than or equal to ρ_{global} and the winning cluster node K belongs to the correct class defined by the target output vector, the remaining steps in the training process are skipped and the training is continued with the next input pattern. However, if O_K is greater than ρ_{global} or the winning cluster node K does not belong to the correct class defined by the target output vector, the system will look back on each segment of the hidden layer. For each segment, the hidden node with the lowest output value, $T_j^i(I^i)$, is selected as the winning node. In case of a tie, the node with the smallest index is chosen. Next, the local vigilance criterion (equation 9) is evaluated to check whether the degree of matching between the input pattern and the reference pattern of the chosen node is within an acceptable level.

$$J = \arg \min_j \{T_j^i(I^i)\} \quad (8)$$

$$T_j^i(I^i) \leq \rho_{\text{local}} \quad (9)$$

where ρ_{local} is the local vigilance parameter at the hidden layer. It has a value between 0 and 1. If the winning node meets the above criterion, the weight vector of the winning node (w_j^i) will be updated according to the following equation:

$$w_j^i(t+1) = \begin{cases} C, & R_i \cap \bar{R}_w \\ w_j^i(t) + c_1, & R_i \cap R_w \\ w_j^i(t) - c_2, & \bar{R}_i \cap R_w \\ 0, & \bar{R}_i \cap \bar{R}_w \end{cases} \quad (10)$$

where C is the initial weight value, which is between 0 and 1.

c_1 and c_2 are the predefined constants.

R_i represents the regions in the input pattern that contain black pixels.

R_w represents the regions in the reference pattern that contain weights.

\bar{R}_i and \bar{R}_w represent the background regions of the input pattern and the reference pattern respectively.

However, if the condition in (9) is not satisfied, a new hidden node is recruited to code the input pattern. The weight of this new node is initialized according to the following equation.

$$w_j^i = f(I_j^i) \quad (11)$$

$$f(x) = \begin{cases} 0, & x \in R_i \\ C, & x \in \bar{R}_i \end{cases} \quad (12)$$

After the winning hidden nodes are picked or created, a new cluster node K is recruited. Then the connection between the new cluster node and each winning hidden node (w_{Kj}^i), and the connection between the new cluster node and the target output are created.

During testing, each testing vector is applied in turn and its class is predicted. The class whose cluster node returns the minimum output value is the result of the prediction.

3. Preprocessing

The handwritten character images used in this study have various sizes as shown in Figure 2. Therefore, they must be normalized before being used. In this research, the original images of the handwritten characters are normalized to the size of 15×15 pixels. Then the thinning algorithm is applied to each character image to reduce the thickness of the character image to its skeleton. Next, each character image is divided into 9 equal segments, each of which is a 5×5

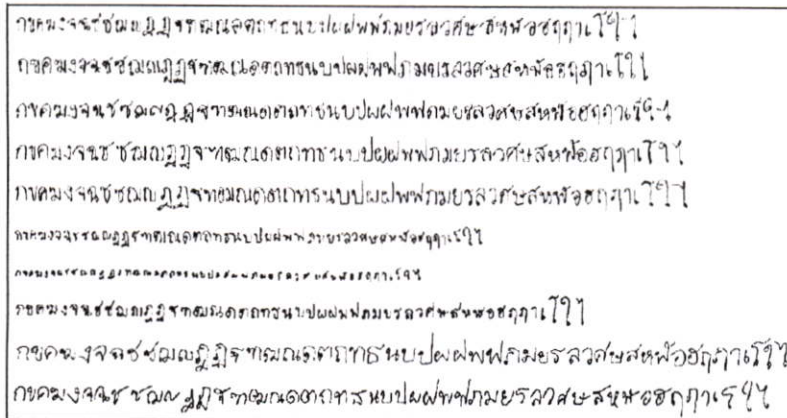


Figure 2. Samples of the tested handwritten character data set



Figure 3. Segmentation of the input image

bitmap image (Figure 3). These 5x5 bitmap images are then sent directly to the hierarchical neural network.

4. Experimental results

The handwritten character images used in this study are parts of the Off-line Thai Handwritten Character corpus [14] developed by the National Electronic and Computer Technology Center (NECTEC). This benchmark data set is written by 30 people. Each person was asked to write 42 Thai characters 3 times; each time is referred later in this paper as an example. Therefore, the total number of character images is 3780 (90 examples).

The experiments are performed in a writer dependent mode. Two examples of each writer are used for training and the remaining one example is used for testing. Since each example consists of 42 characters, the total of 2520 input patterns are used to train the network and the remaining 1260 input patterns are used as the testing data.

The performance of the proposed model is evaluated against the modified Hausdorff ARTMAP

neural network. For the proposed model, the recognition rate is 78.49%. It is obtained when the following parameters are set: $C = 0.5$, c_1 and $c_2 = 0.02$, $\rho_{local} = 0.01$, $\rho_{global} = 0.1$. On the other hand, the best recognition rate of the modified Hausdorff ARTMAP is 64.37%. The proposed model outperforms the modified Hausdorff ARTMAP by a wide margin.

From the detailed analysis, the errors are caused by two main reasons: (a) many of the handwritings are slipshod; and (b) the writer tends to write very small characters, therefore reducing the distinctness of the unique characteristics of each character.

5. Conclusion

The experiments show that the proposed approach can be used successfully to recognize the handwritten Thai characters. With the additional weighting matrix and the model's ability to separately train each segment of the network, the proposed hierarchical neural network can achieve much higher performance on the experimental data in comparison to the modified Hausdorff ARTMAP neural network.

6. References

- [1] B. Gosselin and A. Paggiaro, "A Study of Hidden Markov Models for Off-line Recognition of Handwritten Characters," *Proceedings of the Program for Research on Integrated Systems and Circuits (PRORISC 99)*, November 1999, pp. 177-184.
- [2] J. Doménech et al., "An Off-line HTK-based OCR System for Isolated Handwritten Lowercase Letters,"

Proceedings of the IX Spanish Symposium on Pattern Recognition and Image Analysis, Vol. 2, May 2001, pp. 49-54.

[3] R. Nopsuwanchai and D. Povey, "Discriminative Training for HMM-based Offline Handwritten Character Recognition," *Proceedings of the Seventh International Conference on Document Analysis and Recognition*, Vol. 1, August 2003, pp. 114-118.

[4] T. Theeramunkong and C. Wongtapan, "Off-line Isolated Handwritten Thai OCR Using Island-based Projection with N-gram Model and Hidden Markov Models," *Information Processing and Management*, Vol. 41, Issue 1, January 2005, pp. 139-160.

[5] A. Malaviya and L. Peters, "Handwriting Recognition with Fuzzy Linguistic Rules," *Proceedings of the Third European Congress on Intelligent Techniques and Soft Computing (EUFIT 95)*, 1995, pp. 1430-1434.

[6] Y. H. Tay and M. Khalid, "Comparison of Fuzzy ARTMAP and MLP Neural Networks for Hand-written Character Recognition," *Proceedings of International Federation of Automatic Control (IFAC) Symposium on Artificial Intelligence in Real-Time Control*, September 1997, pp. 363-371.

[7] Y. H. Tay et al., "Hand-written Postcode Recognition by Fuzzy ARTMAP Neural Network," *COSTAM National Science Congress*, 1997.

[8] A. Missiourev, "Hand-printed Character Recognition by Neural Networks," *Proceedings of the 5th German-Russian Workshop on Pattern Recognition and Image Understanding*, September 1998.

[9] C. Koutsougeras and A. Jameel, "Experiments with Various Neural Architectures for Handwritten Character Recognition," *Proceedings of the 37th IEEE Midwest Symposium on Circuits and Systems*, 1995, pp. 573-576.

[10] P. Phokharatkul and C. Kimpan, "Recognition of Handprinted Thai Characters Using the Cavity Features of Character Based on Neural Network," *Proceedings of the 1998 IEEE Asia-Pacific Conference on Circuits and Systems*, 1998, pp. 149-152.

[11] P. Phokharatkul and C. Kimpan, "Handwritten Thai Character Recognition Using Fourier Descriptors and Genetic Neural Networks," *Proceedings of the Fourth Symposium on Natural Language Processing*, May 2000, pp. 108-123.

[12] A. Pornchaikajornsak and A. Thammano, "Handwritten Thai Character Recognition Using Fuzzy Membership Function and Fuzzy ARTMAP," *Proceedings of the 2003 IEEE International Symposium on Computational Intelligence in Robotics and Automation*, July 2003, pp. 40-44.

[13] A. Thammano and S. Ruensuk, "Human Face Recognition Using Modified Hausdorff ARTMAP," *Advances in Intelligent Computing (LNCS 3645)*, August 2005, pp. 248-256.

[14] Off-line Thai Handwritten Character Corpus, Retrieved January 27, 2003, from <http://arthat.lnks.nectec.or.th/twrt/onhwcp.htm>

ประวัติผู้เขียน

นางสาวพัชรวดี พูลสำราญ เกิดเมื่อวันที่ 28 สิงหาคม พ.ศ.2522 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาตรีวิทยาศาสตร์บัณฑิต สาขาวิทยาการคอมพิวเตอร์ จากภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยบูรพา ในปีการศึกษา 2543 และในปีเดียวกันได้สอบเข้าบรรจุเป็นพนักงานมหาวิทยาลัย ตำแหน่งอาจารย์ประจำสาขาคอมพิวเตอร์ธุรกิจ มหาวิทยาลัยบูรพา วิทยาเขตสารสนเทศสระแก้ว และได้ขออนุมัติลาศึกษาต่อในระดับปริญญาโท ในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ในปีการศึกษา 2545 ภาคเรียนที่ 2