

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบจำแนกชนิดยานพาหนะจากแผ่นป้ายทะเบียนโดยใช้โครงข่ายประสาทเทียม
TRANSPORTATION CLASSIFICATION FROM LICENSE-PLATE
USING ARTIFICIAL NEURAL NETWORK



๒๗.
๘489๖
2550

เลขหมู่.....**83300**
เลขทะเบียน.....
วัน,เดือน,ปี...**11 อ.ค. 2551**

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2550

b. 11๑๖๗๕23
i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบจำแนกชนิดยานพาหนะจากแผ่นป้ายทะเบียนโดยใช้โครงข่ายประสาทเทียม
TRANSPORTATION CLASSIFICATION FROM LICENSE-PLATE
USING ARTIFICIAL NEURAL NETWORK



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2550

11๙๒๕23

ปริญญาโทปีการศึกษา 2550

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบจำแนกชนิดยานพาหนะจากแผ่นป้ายทะเบียนโดยใช้โครงข่ายประสาทเทียม

TRANSPORTATION CLASSIFICATION FROM LICENSE-PLATE

USING ARTIFICIAL NEURAL NETWORK

ผู้จัดทำ

1. นายชาญณรงค์ รอดระหงษ์ 48015051

2. นายปริญญา เคนถาวร 48015063

3. นายสรณคมน์ นาคะเสถียร 48015083

อ.ดร.ดร.ดร.
(ผศ.อักรพล สุวีรัตน์)


(อ.สรวิวัฒน์ ชิวปรีชา)

อาจารย์ที่ปรึกษา

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบจำแนกชนิดยานพาหนะจากแผ่นป้ายทะเบียนโดยใช้โครงข่ายประสาทเทียม
TRANSPORTATION CLASSIFICATION FROM LICENSE-PLATE
USING ARTIFICIAL NEURAL NETWORK

โดย นายชาญณรงค์ รอดระหงษ์ 48015051
นายปริญญา เคนถาวร 48015063
นายสรณคมน์ นาคะเสถียร 48015083

อาจารย์ที่ปรึกษา ผศ.อักรพล ตีร์รัตน์
อ.ศรววัฒน์ ชิวปรีชา

บทคัดย่อ

โครงการนี้นำเสนอระบบจำแนกชนิดยานพาหนะ โดยอาศัยแผ่นป้ายทะเบียน ซึ่งในโครงการนี้มีองค์ประกอบหลักๆ อยู่ 3 ส่วน ส่วนแรกเป็นการประมวลผลภาพดิจิทัล ส่วนที่สองเป็นการหาคุณลักษณะพิเศษของภาพโดยใช้การแปลงคอสคริต โคไซน์แบบ 2 มิติ และส่วนที่สามคือการรู้จำป้ายทะเบียนโดยใช้โครงข่ายประสาทเทียมมาทำการวิเคราะห์แผ่นป้ายทะเบียนและอ้างอิงกับฐานข้อมูล

Abstract

This project presents the transportation classification system from license-plate. This project is divided into three main parts. In first part is digital image processing. The second part is feature extraction using 2-D discrete cosine transform. The third part is license plate recognition using artificial neural network to analyze the plate and refer to database.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้ สำเร็จได้ด้วยความกรุณาจากท่านอาจารย์ ซึ่งผู้จัดทำโครงการสมศกวรกราบ
ขอบพระคุณอย่างยิ่ง ได้แก่ ผู้ช่วยศาสตราจารย์อัครพล ตริรัตน์ และอาจารย์ศรววัฒน์ ชิวปรีชา ที่ปรึกษาปริญญา
นิพนธ์ ที่กรุณาให้คำปรึกษาแนะนำช่วยเหลือ ชี้แนะและปรับปรุงแก้ไข ด้วยความเอาใจใส่อย่างค้ำคูณตลอดมา

ผู้จัดทำโครงการขอขอบพระคุณ คุณบิดา คุณมารดา ที่เป็นกำลังใจและให้การสนับสนุนช่วยเหลือมา
โดยตลอด

ประโยชน์และคุณค่าที่พึงมีจากปริญญาานิพนธ์นี้ ผู้จัดทำโครงการขอมอบให้แก่ผู้มีส่วนร่วมทุกท่าน
ตลอดทั้งผู้ที่ต้องการศึกษาในการต่อไป



นายชาญณรงค์ รอดระหงษ์

นายปริญญา เคนถาวร

นายสรณคมน์ นาคะเสถียร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
สารบัญรูปภาพ	
สารบัญตาราง	
บทที่ 1 บทนำ	
1.1 วัตถุประสงค์	1
1.2 ขอบเขตของโครงการ	1
บทที่ 2 ทฤษฎีและหลักการ	
2.1 การประมวลผลภาพดิจิทัล (Digital Image Processing)	2
2.1.1 รูปร่างของภาพ (Image Pattern)	2
2.1.2 มาตรฐานของสี (Color Standard)	3
2.1.3 เทรชโฮลด์ (Threshold)	5
2.1.4 ทฤษฎีการหาขอบภาพ (Edge Detection Method)	6
2.1.5 การแยกภาพออกเป็นส่วนๆ (Image Segmentation)	8
2.2 หลักการพื้นฐานของการทรานสฟอร์ม	9
2.2.1 สมการของการทรานสฟอร์ม	9
2.2.2 การแปลงแบบคอสครีต โคซายน์ (Discrete Cosine Transform)	11
2.2.3 การแปลงคอสครีต โคซายน์ 2 มิติ (2 Dimensions Discrete Cosine Transform)	14
2.2.4 การแปลงฮัฟ (Hough Transform)	19
2.3 โครงข่ายประสาทเทียม (Neural Network)	21
2.3.1 ฟังก์ชันกระตุ้นความสนใจ (Activation Function)	23
2.3.2 กฎการเรียนรู้ของเพอร์เซพตรอน	24
2.3.3 กฎการเรียนรู้ของเซลล์ด้า	25
2.3.4 นิวรอลเน็ตเวิร์คแบบเลเยอร์เดียว (Neural Network singlelayer)	27
2.3.5 นิวรอลเน็ตเวิร์คแบบหลายเลเยอร์ (Neural Network Multilayer)	28
2.3.6 ปังจัยที่ทำให้ประสิทธิภาพการสอน โครงข่ายด้วยวิธี Back Propagation เพิ่มขึ้น	31
2.3.7 นิวรอลเน็ตเวิร์คกับพีชคณิตบูลีนแบบเอ็กซ์คลูซีฟ	34
2.4 ระบบฐานข้อมูล	38
2.4.1 ประเภทของคำสั่งของภาษา SQL	39
2.4.2 ชนิดของข้อมูลที่ใช้ในภาษา SQL	39
2.5 รูปแบบของแผ่นป้ายทะเบียน	41
บทที่ 3 การออกแบบและการสร้าง	
3.1 การประมวลผลภาพดิจิทัล (Digital Image Processing)	44

สารบัญ (ต่อ)

	หน้า
3.2 การแยกคุณลักษณะเฉพาะจากข้อมูลภาพ (Feature extraction)	54
3.3 การรู้จำป้ายทะเบียน (License-plate recognition)	55
3.3.1 การฝึกสอนโครงข่ายประสาทเทียม (Training)	55
3.3.2 การนำค่าถ่วงน้ำหนักไปใช้งาน (Working)	60
3.4 การอ้างอิงป้ายทะเบียนกับระบบฐานข้อมูล (Reference to Database)	61
บทที่ 4 ผลการทดลองและการวิเคราะห์	
4.1 การประมวลผลภาพดิจิทัล (Digital Image Processing)	64
4.2 การแยกคุณลักษณะเฉพาะจากข้อมูลภาพ (Feature extraction)	74
4.3 การรู้จำป้ายทะเบียน (License-plate recognition)	76
4.4 การอ้างอิงป้ายทะเบียนกับระบบฐานข้อมูล (Reference to Database)	96
4.5 ผลความผิดพลาดที่เกิดจากข้อจำกัดของระบบ	97
บทที่ 5 บทสรุปและวิจารณ์	
5.1 บทสรุป	101
5.2 บทวิจารณ์	101
5.3 ปัญหาและอุปสรรคที่พบ	101
ภาคผนวก ก.	
ภาคผนวก ข.	
กิตติกรรมประกาศ	
เอกสารอ้างอิง	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

	หน้า
รูปที่ 2.1 กลุ่มของพิกเซลที่เรียงกันเป็นตัวอักษร	3
รูปที่ 2.2 ปริภูมิสี RGB	4
รูปที่ 2.3 ระบบสี RGB รูปแบบเมตริกซ์	4
รูปที่ 2.4 ความเข้มแสง 256 ระดับ	5
รูปที่ 2.5 ภาพหลังทำการเทรซโฮลด์ค่าที่เหมาะสม	6
รูปที่ 2.6 ภาพหลังทำการเทรซโฮลด์ ที่ค่าเทรซโฮลด์สูงเกินไป	6
รูปที่ 2.7 ภาพหลังทำการเทรซโฮลด์ ที่ค่าเทรซโฮลด์ต่ำเกินไป	6
รูปที่ 2.8 หน้ากากที่ใช้คำนวณค่า G_x และ G_y	7
รูปที่ 2.9 ลักษณะของพิกเซล 8 Neighbors	9
รูปที่ 2.10 เมตริกซ์สัมประสิทธิ์การแปลงดิสครีตโคไซน์	12
รูปที่ 2.11 ลักษณะของการแปลงดิสครีตโคไซน์ของบล็อกข้อมูลขนาด 5×5 พิกเซล	14
รูปที่ 2.12 บล็อกโคอะแกรมของการแปลงดิสครีตโคไซน์แบบ 2 มิติ	15
รูปที่ 2.13 บล็อกโคอะแกรมการแปลงดิสครีตโคไซน์แบบ 1 มิติ ตามแนวนอน	15
รูปที่ 2.14 บล็อกโคอะแกรมการทรานสโพสบล็อกข้อมูล	15
รูปที่ 2.15 บล็อกโคอะแกรมการแปลงดิสครีตโคไซน์แบบ 1 มิติ ตามแนวตั้ง	16
รูปที่ 2.16 ภาพต้นแบบก่อนทำการแปลงดิสครีตโคไซน์	17
รูปที่ 2.17 ภาพของค่าสัมประสิทธิ์ดิสครีตโคไซน์หลังทำการแปลง	17
รูปที่ 2.18 บล็อกขนาด 5×5 ที่นำมาคูณกับรูปที่ 2.16	18
รูปที่ 2.19 ผลลัพธ์ของสัมประสิทธิ์จากการคูณด้วยบล็อกในรูปที่ 2.17	18
รูปที่ 2.20 ภาพหลังจากการแปลงกลับดิสครีตโคไซน์	18
รูปที่ 2.21 เส้นตรงในพิกัดเชิงขั้ว	20
รูปที่ 2.22 การทำ Hough Transform	21
รูปที่ 2.23 แบบจำลองหลักๆ ของ ANN	22
รูปที่ 2.24 Activation Function ชนิดต่างๆ	23
รูปที่ 2.25 โครงสร้างนิเวรอลเน็ตเวิร์คแบบเพอร์เซพตรอน	24
รูปที่ 2.26 โครงสร้างกฎการเรียนรู้ของเซลล์	25
รูปที่ 2.27 ความสัมพันธ์ระหว่างค่าผิดพลาดต่ำสุดและค่าถ่วงน้ำหนัก	27
รูปที่ 2.28 นิเวรอลเน็ตเวิร์คแบบเลเยอร์เดียว	27
รูปที่ 2.29 โครงสร้างของนิเวรอลเน็ตเวิร์คแบบหลายเลเยอร์	28
รูปที่ 2.30 การหาอนุพันธ์อันดับหนึ่งของฟังก์ชันแอคติเวชัน	32

สารบัญรูปภาพ (ต่อ)

	หน้า
รูปที่ 2.31 ทิศทางการเคลื่อนที่ของการปรับค่าดวงน้ำหนัก	33
รูปที่ 2.32 โครงสร้างนิเวรอลเน็ตเวิร์คสำหรับพีชคณิตบูลีนแบบเอ็กซ์คลูซีฟออร์	35
รูปที่ 2.33 กราฟการแยกจำพวกของกลุ่มแพทเทิร์นของพีชคณิตบูลีนแบบเอ็กซ์คลูซีฟออร์	36
รูปที่ 2.34 แผนภูมิการทำงานของการแพร่กระจายกลับอัลกอริทึม	37
รูปที่ 2.35 โครงสร้างของเพิ่มข้อมูล	38
รูปที่ 2.36 ป้ายทะเบียนแบบเก่า	41
รูปที่ 3.1 การทำงานของระบบ	43
รูปที่ 3.2 การรับข้อมูลภาพโดยการเชื่อมต่อกับอุปกรณ์ TV Tuner ด้วยโปรแกรม MATLAB	45
รูปที่ 3.3 แผนภูมิการทำงานของการแปลงภาพ RGB เป็นภาพขาว-ดำ	45
รูปที่ 3.4 แผนภูมิการทำงานของการแปลงภาพขาว-ดำ เป็นภาพไบนารี	46
รูปที่ 3.5 แผนภูมิการทำงานของการหาขอบภาพแบบ Sobel	47
รูปที่ 3.6 แผนภูมิการทำงานของการหาค่าแห่งป้ายทะเบียนด้วย Hough Transform	48
รูปที่ 3.7 แผนภูมิการทำงานของการทำ Image enhancement	49
รูปที่ 3.8 แผนภูมิการทำงานของการตัดส่วนที่เป็นจังหวัดออกจากป้ายทะเบียน	50
รูปที่ 3.9 ฮิสโตแกรมก่อนการตัดจังหวัดของป้ายทะเบียนออก โดยการสแกนภาพในแนวนอน	51
รูปที่ 3.10 ฮิสโตแกรมก่อนการตัดจังหวัดของป้ายทะเบียนออก โดยการสแกนภาพในแนวตั้ง	51
รูปที่ 3.11 แผนภูมิการทำงานของการแยกส่วนตัวอักษรออกจากกัน	52
รูปที่ 3.12 ฮิสโตแกรมหลังการตัดจังหวัดของป้ายทะเบียนออก โดยการสแกนภาพในแนวนอน	53
รูปที่ 3.13 ฮิสโตแกรมหลังการตัดจังหวัดของป้ายทะเบียนออก โดยการสแกนภาพในแนวตั้ง	53
รูปที่ 3.14 บล็อก 5×5 การเก็บค่าสัมประสิทธิ์จำนวน 10 ค่า	54
รูปที่ 3.15 บล็อก 10×10 การเก็บค่าสัมประสิทธิ์จำนวน 28 ค่า	55
รูปที่ 3.16 แผนภูมิการฝึกสอนโครงข่ายประสาทเทียม	59
รูปที่ 3.17 แผนภูมิการนำค่าดวงน้ำหนักไปใช้งาน ให้ประมวลผลรูปแบบตัวอักษรและตัวเลข	60
รูปที่ 3.18 การเชื่อมต่อระหว่างโปรแกรม MATLAB กับฐานข้อมูล	61
รูปที่ 3.19 โครงสร้างของฐานข้อมูลจำลองที่สร้างขึ้นเมื่อใช้คำสั่ง	62
รูปที่ 3.20 ซอฟต์แวร์ phpMyAdmin ที่ใช้ในการจัดการกับฐานข้อมูลที่บริการผ่านเว็บ	63
รูปที่ 3.21 ขั้นตอนการคิดอัตราค่าผ่านทาง โดยการพิจารณาจากจำนวนล้อของยานพาหนะ	63
รูปที่ 4.1 ภาพต้นฉบับของยานพาหนะพร้อมทะเบียน	64
รูปที่ 4.2 ภาพต้นฉบับที่ทำการแปลงเป็นภาพ Gray Scale	65
รูปที่ 4.3 ภาพ Gray Scale ที่ทำการแปลงเป็นภาพไบนารี	66

สารบัญรูปภาพ (ต่อ)

	หน้า
รูปที่ 4.4 ภาพหลังจากการหาขอบภาพด้วยวิธี Sobel edge detection	67
รูปที่ 4.5 ภาพหน้าฉากขนาด 4x4 ส่วนของภาพต้นฉบับ	68
รูปที่ 4.6 ภาพที่ได้จากการครอบด้วยหน้าฉาก	69
รูปที่ 4.7 เส้นแสดงตำแหน่งที่คาดว่าเป็นตำแหน่งของป้ายทะเบียน	70
รูปที่ 4.8 ภาพผลลัพธ์จากการหาตำแหน่งป้ายทะเบียน	71
รูปที่ 4.9 ภาพป้ายทะเบียนหลังจากปรับแต่งภาพ	72
รูปที่ 4.10 ภาพป้ายทะเบียนหลังจากที่ถูกตัดส่วนของจังหวัดออก	72
รูปที่ 4.11 ภาพหลังจากการแยกตัวอักษร	73
รูปที่ 4.12 ภาพหลังจากการปรับขนาดให้เท่ากันทุกตัวอักษร	73
รูปที่ 4.13 โครงสร้างระบบโครงข่ายประสาทเทียมที่ใช้ในการฝึกสอน	77
รูปที่ 4.14 กราฟความสัมพันธ์ระหว่างค่าผิดพลาดเฉลี่ยกำลังสองกับจำนวนรอบที่ทำการฝึกสอน	77
รูปที่ 4.15 ขอบของป้ายทะเบียนที่เอียงและมีสีกลมกลืนกับตัวยานพาหนะ	97
รูปที่ 4.16 ภาพหลังจากผ่านกระบวนการหาขอบภาพของยานพาหนะรูปที่ 4.15	97
รูปที่ 4.17 หลังจากผ่านการแปลงฮัฟจากรูปที่ 4.15 แล้วไม่สามารถระบุตำแหน่งได้	98
รูปที่ 4.18 ป้ายทะเบียนที่เอียงและถ่ายภาพในสภาพแวดล้อมที่มีแสงจ้า	98
รูปที่ 4.19 ภาพหลังจากผ่านกระบวนการหาขอบภาพของยานพาหนะรูปที่ 4.18	99
รูปที่ 4.20 ยานพาหนะที่ตัวอักษรบนป้ายทะเบียนจาง	99
รูปที่ 4.21 ป้ายทะเบียนที่ไม่สามารถแยกตัวอักษรได้หลังจากผ่านกระบวนการประมวลผลภาพ และหาตำแหน่งป้ายทะเบียนของรูปที่ 4.20 เนื่องจากตัวอักษรบนป้ายทะเบียนจาง	100
รูปที่ 4.22 ป้ายทะเบียนที่มีเนื้อด	100
รูปที่ 4.23 หลังจากผ่านกระบวนการแยกตัวอักษรจากรูปที่ 4.22 จะได้แพทเทิร์นอักษรที่เพี้ยน	100

สารบัญตาราง

	หน้า
ตารางที่ 2.1 คำสัมประสิทธิ์ของการทรานสฟอร์มในรูปแบบของฟังก์ชันตรีโกณมิติ	12
ตารางที่ 2.2 กลุ่มข้อมูลอินพุตแพทเทิร์นของพีชคณิตบูลีนแบบเอ็กซ์คลูซีฟออร์	34
ตารางที่ 2.3 รูปแบบป้ายทะเบียนแบบใหม่	42
ตารางที่ 3.1 ภาพตัวอักษรและตัวเลขที่นำมาฝึกสอนให้กับโครงข่ายประสาทเทียม	56
ตารางที่ 3.2 การกำหนดเป้าหมายในการฝึกสอนให้กับโครงข่ายประสาทเทียม	57
ตารางที่ 4.1 การเปรียบเทียบการแปลงดิสคริตโคไซน์ โดยใช้ DCT เมตริกซ์ ขนาด 5×5	75
ตารางที่ 4.2 การเปรียบเทียบการแปลงดิสคริตโคไซน์ โดยใช้ DCT เมตริกซ์ ขนาด 10×10	76
ตารางที่ 4.3 ค่าถ่วงน้ำหนักของชั้นฮิดเดน โหนด 1-8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 168 ค่า	78
ตารางที่ 4.4 ค่าถ่วงน้ำหนักของชั้นฮิดเดน โหนด 1-8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 240 ค่า	83
ตารางที่ 4.5 ค่าเอาต์พุตของการประมวลผลของโครงข่ายกับป้ายทะเบียน ฐน 7264	91
ตารางที่ 4.6 ค่าเอาต์พุตของการประมวลผลของโครงข่ายกับป้ายทะเบียน พม 6333	93
ตารางที่ 4.7 ผลของการจดจำป้ายทะเบียนจากการแปลงดิสคริตโคไซน์	94

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

ในปัจจุบันได้มีการพัฒนาระบบการบริหารงานบนทางด่วนที่ดี แต่ถ้าระบบที่ดีมีการใช้งานที่ยาวนานก็จะพบข้อบกพร่องขึ้น เกี่ยวกับการชำระหรือการเชื่อมต่อโทรมของอุปกรณ์ เช่น การชำระของอุปกรณ์ทางไฟฟ้า อุปกรณ์อิเล็กทรอนิกส์ต่างๆ ที่ใช้ในการนับจำนวนรถเข้าหรือออกของด่านเก็บเงิน ทำให้เป็นปัญหาในการตรวจสอบความถูกต้องในเรื่องจำนวนรถที่เข้าหรือออก ไม่ตรงกับค่าผ่านทางที่เก็บจากผู้ขับขี่รถยนต์ รวมถึงการจำแนกชนิดรถยังคงต้องใช้พนักงานที่เก็บค่าผ่านทางเป็นผู้วิเคราะห์ทำให้เกิดความล่าช้าขึ้นในขั้นตอนนี้ จึงทำให้เกิดเป็นแนวคิดในการประยุกต์และนำข้อดีที่เป็นจุดโหว่มาทำการศึกษา โดยการนำระบบที่เรียกว่า การรู้จำแผ่นป้ายทะเบียน (License Plate Recognition) มาใช้ในการประมวลผล ตลอดจนนำมาใช้ในการจำแนกชนิดของยานพาหนะด้วย

ในการวิเคราะห์ข้อมูลแผ่นป้ายทะเบียนที่ได้จากการถ่ายภาพของกล้อง CCD (CCD Camera) แล้วนำมาทำการประมวลผลทางคอมพิวเตอร์เพื่อให้ได้ผลลัพธ์เป็นตัวอักษร แล้วนำตัวอักษรที่ได้ จัดเก็บเป็นฐานข้อมูลใช้อ้างอิงกับฐานข้อมูลของทะเบียนรถเพื่อการจำแนกชนิดในการต่อไป

นอกจากนี้ระบบยังสามารถนำไปพัฒนาและประยุกต์ใช้งานด้านอื่นๆ ได้อีกด้วย อาทิเช่น ใช้ในระบบรักษาความปลอดภัยของสถาบันการศึกษา เพื่อตรวจการเข้า-ออกของรถและยังสามารถเก็บเป็นฐานข้อมูลหากมีการสูญหายในทรัพย์สินของสถาบันการศึกษา เป็นต้น

1.1 วัตถุประสงค์

พัฒนาโปรแกรมที่สามารถวิเคราะห์หมายเลขทะเบียนและจำแนกชนิดของยานพาหนะได้ว่าเป็นยานพาหนะประเภทไหน และยังสามารถแสดงข้อมูลบุคคลที่เป็นเจ้าของรถ ตลอดจนทราบถึงข้อมูลทางเทคนิคของยานพาหนะแต่ละประเภทด้วย โดยใช้การอ้างอิงกับฐานข้อมูล ทั้งหมดนี้ได้อาศัยการประมวลผลทางภาพ และความรู้เกี่ยวกับโครงข่ายประสาทเทียม

1.2 ขอบเขตของโครงการ

โปรแกรมที่พัฒนาขึ้นจะทำการวิเคราะห์และอ่านหมายเลขทะเบียนจากแผ่นป้ายทะเบียนที่เป็นแบบมาตรฐานได้อย่างถูกต้อง ทั้งที่เป็นตัวอักษรภาษาไทยและตัวเลข โดยวิเคราะห์จากภาพถ่ายป้ายทะเบียนและให้ผลเป็นข้อมูลตัวอักษรของป้ายทะเบียน หลังจากนั้นจะนำข้อมูลดังกล่าว ไปอ้างอิงกับระบบฐานข้อมูลที่จำลองขึ้นผ่านระบบอินเทอร์เน็ต ซึ่งทำการจำลองให้เป็นระบบฐานข้อมูลของกรมการขนส่งทางบก โดยในระบบฐานข้อมูลที่จำลองมานี้มีข้อมูลของป้ายทะเบียน จังหวัด ชื่อเจ้าของรถ จำนวนล้อและข้อมูลอื่นๆ ในการอ้างอิงนั้นจะทำการอ้างอิงจากป้ายทะเบียนที่ทำการวิเคราะห์จากภาพป้ายทะเบียน แล้วนำข้อมูลจากฐานข้อมูลของยานพาหนะที่กำลังอ้างอิงถึง มาใช้จำแนกชนิดยานพาหนะ

บทที่ 2 ทฤษฎีและหลักการ

ในบทนี้จะกล่าวถึงทฤษฎีต่างๆ ที่ใช้ในการกระบวนการของการประมวลผลภาพดิจิทัล การแยกคุณลักษณะเฉพาะของข้อมูลภาพด้วยการแปลงคิสคริตโคซายน์ การรู้จำแผ่นป้ายทะเบียนด้วยระบบโครงข่ายประสาทเทียม ระบบฐานข้อมูลที่จะนำมาจำลองใช้งานในการเก็บข้อมูลของยานพาหนะและข้อมูลของป้ายทะเบียนรถที่เกี่ยวข้องกับการนำไปใช้งานของโครงการนี้

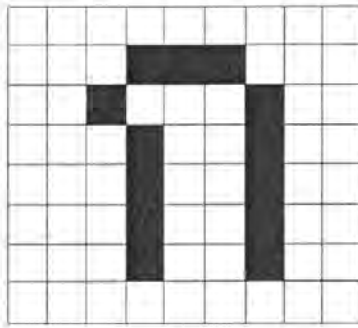
2.1 การประมวลผลภาพดิจิทัล (Digital Image Processing)

การประมวลผลภาพดิจิทัล (Digital Image Processing) คือ การประยุกต์ใช้งานการประมวลผลสัญญาณลักษณะรูปภาพ 2 มิติ จะเกี่ยวกับการแปลงข้อมูลภาพให้อยู่ในรูปแบบข้อมูลดิจิทัล (Digital format) ซึ่งสามารถที่จะนำเอาข้อมูลนี้จัดผ่านกระบวนการต่าง ๆ ด้วยดิจิทัลคอมพิวเตอร์ได้ ในระบบของดิจิทัล อินพุตและเอาต์พุตของระบบจะอยู่ในรูปแบบดิจิทัลเท่านั้น โดยที่ข้อมูลซึ่งเป็นรูปภาพจะถูกเปลี่ยนเป็นรูปแบบของตัวเลข ภาพเหล่านี้ได้จากการถ่ายของกล้องดิจิทัล สำหรับจุดประสงค์ของการประมวลผลภาพดิจิทัล คือ การแยกแยะข้อมูลที่สำคัญออกจากภาพ

การวิเคราะห์ภาพดิจิทัล (Digital Image Analysis) คือ การอธิบายถึงลักษณะที่มีความสำคัญ รวมถึงการจดจำข้อมูลภาพดิจิทัล ซึ่งอินพุตของระบบจะเป็นข้อมูลภาพดิจิทัลและเอาต์พุตจะเป็นเครื่องหมายที่ใช้แทนข้อมูลภาพดิจิทัล ในการวิเคราะห์ภาพนั้นมีอยู่หลายวิธีด้วยกัน เช่น การทำงานของตามนุษย์ (Human Vision) ซึ่งการมองเห็นของตามนุษย์นั้นเป็นกระบวนการที่ซับซ้อน ซึ่งจะให้ข้อมูลที่มีความจำเป็นสำหรับใช้ในางง่าย ๆ ตัวอย่างเช่น การจดจำวัตถุ แต่สำหรับงานที่มีความซับซ้อน ได้แก่ การพัฒนางานทางด้านความคิด กระบวนการจดจำ และการวิเคราะห์ จำเป็นต้องใช้กระบวนการทางคอมพิวเตอร์เข้ามาช่วย

2.1.1 รูปร่างของภาพ (Image Pattern)

โดยทั่วไปวัตถุที่อยู่ตามธรรมชาติและที่มนุษย์สร้างขึ้น มีรูปร่างที่ต่างกันออกไป ทั้งที่เป็นรูปทรงเรขาคณิตและไม่เป็นรูปทรงเรขาคณิต ในการประมวลผลภาพนั้น การกำหนดขอบเขตของภาพทุกภาพให้อยู่ในรูปสี่เหลี่ยม (Rectangular Image Model) ซึ่งเป็นวิธีที่ง่าย เนื่องจากทำให้การอ่านภาพ การจัดเก็บข้อมูลเป็นไปอย่างมีประสิทธิภาพ ซึ่งการจัดเก็บข้อมูลจะอยู่ในรูปของตัวแปรอาร์เรย์ (Array) โดยค่าในแต่ละช่องของอาร์เรย์แสดงถึงคุณสมบัติของจุดภาพ (Pixel: ย่อมาจาก Picture element บางทีเรียก Pel) และตำแหน่งของช่องอาร์เรย์จะเป็นตัวกำหนดตำแหน่งของจุดภาพ



รูปที่ 2.1 กลุ่มของพิกเซลที่เรียงกันเป็นตัวอักษร

2.1.2 มาตรฐานของสี (Color Standard)

มาตรฐานของสีที่ใช้อยู่ในปัจจุบันมีอยู่หลายระบบด้วยกัน ทั้งนี้ขึ้นอยู่กับนำไปใช้แต่โดยทั่วไปทุกมาตรฐานจะมีแนวคิดเดียวกัน คือ การแทนจุดสีด้วยจุดที่อยู่ในสเปซ (Space) 3 มิติ โดยจะมีแกนอ้างอิงสำหรับจุดสีนั้น ในสเปซซึ่งแต่ละแกนจะมีสระต่อกัน

2.1.2.1 ระบบสี RGB

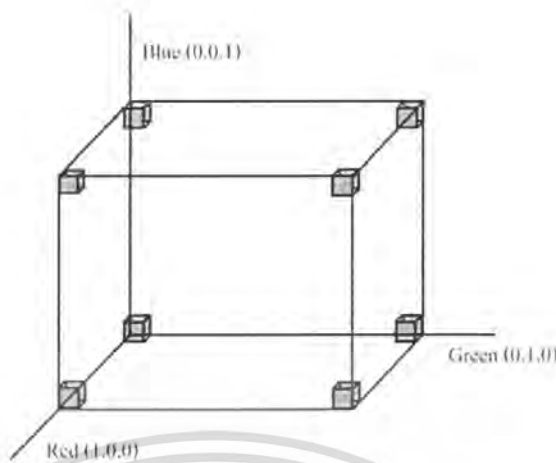
ระบบสี RGB เป็นระบบที่เกิดจากการรวมกันของแสงสีแดง เขียวและน้ำเงิน ในรูปภาพสีแต่ละพิกเซล $C(m, n) = \{R, G, B\}$ แทนด้วย 3 องค์ประกอบสี ถ้าแต่ละองค์ประกอบสีมี K ระดับและสามารถแทนด้วย

$$k = \log_2 K \text{ (บิต)} \quad (2.1)$$

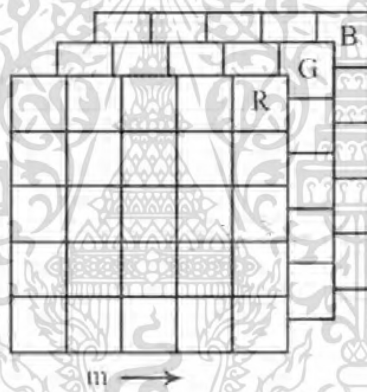
แต่ละพิกเซลจะแทนได้ด้วย $3k$ บิต และจำนวนบิตทั้งหมดของภาพนี้เท่ากับ

$$N = \text{จำนวนพิกเซล} \times \text{จำนวนบิต/พิกเซล} = 3 \text{ mnk (บิต)} \quad (2.2)$$

ดังนั้นหากค่า K เท่ากันรูปภาพดิจิทัลสีจะมีจำนวนบิตเป็นสามเท่าของรูปภาพดิจิทัลขาวดำ ตัวอย่างเช่น หากรูปภาพสีขนาด 20 แถวและ 20 หลัก ค่า $K = 256$ ระดับ แต่ละพิกเซลจะสามารถแทนได้ด้วย $k = 3 \times \log_2 256 = 24$ บิต รูปภาพนี้สามารถแทนได้ด้วย 9,600 บิต



รูปที่ 2.2 ปริภูมิสี RGB



รูปที่ 2.3 ระบบสี RGB รูปแบบเมตริกซ์

2.1.2.2 ระบบสี Grayscale

กระบวนการนี้จะพบว่า พิกเซลภาพแต่ละพิกเซลจะให้ค่าความเข้มของแสงออกมาเป็นตัวเลขที่มีค่าอยู่ระหว่าง 0 ถึง 255 เท่านั้น ช่วงข้อมูลดังกล่าวสามารถแทนด้วยข้อมูลขนาด 1 ไบท์ หรือ 8 บิต ที่จะทำให้ความละเอียด 2^8 หรือ 256 ระดับ ซึ่งทำให้ภาพที่เป็นโทนขาว-ดำ ถูกเรียกว่า Gray Scale Image ในรูปภาพขาว-ดำ แต่ละพิกเซล $C(m, n)$ แทนด้วยเลขสีขาวยุคจำนวน K ระดับและสามารถแทนด้วย

$$k = \log_2 K \text{ (บิต)} \tag{2.3}$$

และจำนวนบิตทั้งหมดของภาพนี้เท่ากับ

$$N = \text{จำนวนพิกเซล} \times \text{จำนวนบิต/พิกเซล} = mnk \text{ (บิต)} \tag{2.4}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างเช่น หากรูปภาพขาวดำขนาด 20 แถวและ 20 หลัก ค่า $K = 256$ ระดับ ตั้งแต่สีขาวไปจนถึงสีดำ แต่ละพิกเซลจะสามารถแทนได้ด้วย $k = \log_2 256 = 8$ บิต ดังนั้นรูปภาพนี้สามารถแทนได้ด้วย 3,200 บิต



รูปที่ 2.4 ความเข้มแสง 256 ระดับ

ฉะนั้นค่าสีขาว-ดำ (Gray Value) จะมีค่าตามสมการดังนี้

$$\text{Gray Scale} = (0.299 \times R) + (0.587 \times G) + (0.114 \times B) \quad (2.5)$$

2.1.3 เทรชโฮลด์ (Threshold)

ในงาน Image Processing การทำ Threshold ถือเป็นารลดขั้นตอนของการดำเนินงานให้ง่ายขึ้น และสามารถให้ประสิทธิภาพได้ดี โดยการแบ่งระดับสีจากภาพที่มีหลายระดับสี ซึ่งจะแปลงข้อมูลภาพขาว-ดำ (Gray Scale) ให้เป็นภาพไบนารี (Binary Image) มีกระบวนการแปลงภาพที่มีความเข้มหลายระดับ (Multilevel Image) เพื่อให้เกิดความแตกต่างบนภาพเพียง 2 ค่าเท่านั้นคือ 0 และ 1

Thresholding Technique คือ การพิจารณาจุดพิกเซล ว่าจุดใดควรจะเป็นจุดขาว หรือจุดใดควรจะเป็นจุดดำ โดยการเปรียบเทียบค่าแต่ละพิกเซล $G(i, j)$ กับค่าคงที่ ที่เรียกว่า Threshold Value เทคนิคนี้นิยมใช้กันมากในกรณีที่ความแตกต่างของวัตถุ (Object) และพื้นหลัง (Background) ซึ่งเป็นลักษณะเดียวกับแผ่นป้ายทะเบียนรถยนต์ ที่มีพื้นหลังเป็นสีขาว และตัวอักษรเป็นสีดำ แต่ในที่นี้เราจะกำหนดทำให้บิตข้อมูลที่เป็นบิต 1 (สีขาว)แทนตัวอักษร และบิต 0 (สีดำ)แทนพื้นหลังทะเบียน ตามสมการที่ (2.6) การกำหนดเช่นนี้ คือ การกำหนดค่าของพิกเซลว่าถ้าค่าใดค่าหนึ่งสูงกว่าหรือเท่ากับค่าเทรชโฮลด์ที่ตั้งไว้ให้ค่าเป็นค่าหนึ่ง แต่ถ้าต่ำกว่าค่าเทรชโฮลด์ ให้ค่าเป็นอีกค่าหนึ่ง เมื่อภาพที่ผ่านกระบวนการนี้แล้วจะได้ออกมาเป็นภาพไบนารี

$$B(i, j) = \begin{cases} 0, & G(i, j) \geq T \\ 1, & G(i, j) < T \end{cases} \quad (2.6)$$

ในการทำภาพไบนารีโดยการทำ Thresholding ให้ได้ภาพที่ดี และต้องเลือกค่าเทรชโฮลด์ ให้ถูกต้องและเหมาะสมดังรูปที่ 2.5 ถ้าเลือกค่าไม่เหมาะสม เช่นค่าเทรชโฮลด์ สูงเกินไปดังรูปที่ 2.6 หรือต่ำเกินไปดังรูปที่ 2.7 รายละเอียดของภาพที่ได้ก็จะขาดต่อการนำไปวิเคราะห์เนื่องจากภาพไม่คมชัด และถ้าภาพมืดเกินไปหรือสว่างเกินไป จะทำให้ภาพเกิดสิ่งรบกวน (Noise) ขึ้นบนภาพได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 ภาพหลังทำการเทรซโฮลด์ค่าที่เหมาะสม



รูปที่ 2.6 ภาพหลังทำการเทรซโฮลด์ ที่ค่าเทรซโฮลด์สูงเกินไป

รูปที่ 2.7 ภาพหลังทำการเทรซโฮลด์ ที่ค่าเทรซโฮลด์ต่ำเกินไป

2.1.4 ทฤษฎีการหาขอบภาพ (Edge Detection Method)

การหาขอบภาพ เป็นการหาขอบเขตของวัตถุภายในภาพ โดยขอบภาพเกิดจากความแตกต่างของความเข้มแสงจากจุดหนึ่งไปยังอีกจุดหนึ่ง ดังนั้นถ้าหากความเข้มแสงมีค่าความแตกต่างมากจะทำให้ขอบภาพมีความชัดเจน ในทางตรงข้ามถ้าหากมีค่าความแตกต่างน้อยขอบภาพจะไม่ชัดเจน ซึ่งการหาขอบภาพแบ่งออกเป็น 2 ประเภทคือ Gradient method และ Laplacian method โดยในแต่ละวิธีมีรายละเอียดดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4.1 Gradient Method

วิธีนี้จะหาขอบโดยการหาจุดต่ำสุดและจุดสูงสุดในรูปของอนุพันธ์อันดับหนึ่งของภาพ โดยจุดที่เป็นขอบจะอยู่ในส่วนที่เหนือค่า Threshold จึงอาจทำให้เส้นขอบที่ได้มีลักษณะหนา ตัวอย่างวิธีการหาขอบของกลุ่มนี้ เช่น Roberts, Prewitt, Sobel เป็นต้น แต่ในโครงการนี้จะนำวิธีการหาขอบภาพแบบ Sobel (Sobel Edge Detection) มาใช้

Sobel Edge Detection เป็นการหาขอบภาพที่ไม่เป็นเชิงเส้น สามารถเปลี่ยนแปลงค่าความไม่ต่อเนื่องได้ตามการปรับปรุขอบให้ A เป็นภาพต้นฉบับ และ A0, A1, ..., A7 แสดงถึงตำแหน่งของแต่ละ pixel ทั้ง 8 จุดที่ทวนเข็มนาฬิกา หน้ากากที่ใช้คำนวณมีขนาด 3×3

A3	A2	A1
A4	$G(i, j)$	A0
A5	A6	A7

รูปที่ 2.8 หน้ากากที่ใช้คำนวณค่า G_x และ G_y

สมการที่ (2.7) และ (2.8) คือ สมการหาขอบภาพในแนวนอนและแนวตั้งตามลำดับ

$$G_x(i, j) = \frac{1}{k+2} [(A5 + 2A6 + A7) - (A1 + 2A2 + A3)] \quad (2.7)$$

$$G_y(i, j) = \frac{1}{k+2} [(A7 + 2A0 + A1) - (A3 + 2A4 + A5)] \quad (2.8)$$

ดังนั้นเกรเดียนท์ของขอบภาพ Sobel เป็น M เมื่อค่า k ที่นำมาใช้มีค่าเท่ากับ 2

$$M(i, j) = (G_x^2(i, j) + G_y^2(i, j))^{1/2} \quad (2.9)$$

ทิศทางของเกรเดียนท์ของภาพ D คือ

$$D(i, j) = \tan^{-1} \left\{ \frac{G_x(i, j)}{G_y(i, j)} \right\} \quad (2.10)$$

2.1.4.2 Laplacian Method

วิธีนี้จะใช้หลักการพิจารณาการเปลี่ยนแปลงค่าเกรเดียนท์ ของค่าระดับสีเทาอีกต่อหนึ่งและจะกำหนดตำแหน่งที่มี Zero-Crossing เกิดขึ้นเป็นตำแหน่งขอบของวัตถุ ทั้งนี้การหาอนุพันธ์ย่อยอันดับสองของภาพ $f(x, y)$ คือการหาค่าลาปลาเซียนแบบ 2 มิติ เช่น Canny เป็นต้น

2.1.5 การแยกภาพออกเป็นส่วนๆ (Image Segmentation)

เป็นการแยกองค์ประกอบของภาพให้เป็นส่วนๆ เพื่อใช้ในการพิจารณาโดยเอาส่วนที่ต้องการออกจากกัน ในขั้นตอนนี้เป็นขั้นตอนที่สำคัญมาก เนื่องจากต้องแยกให้ได้ข้อมูลที่ถูกต้องที่สุด ถ้าขั้นตอนนี้ผิดพลาดจะทำให้ขั้นตอนในส่วนถัดไปเกิดผลเสียเป็นอย่างมาก

ประโยชน์ของการทำ Image Segmentation

1. ลดจำนวนข้อมูลรูปภาพที่ไม่จำเป็นในการวิเคราะห์
2. จัดระเบียบข้อมูลภาพให้ง่ายต่อการพิจารณาในขั้นตอนต่อไป
3. ให้ข้อมูลภาพมีความเป็นระเบียบ
4. การแสดงของข้อมูลภาพสามารถเข้าใจได้ง่าย

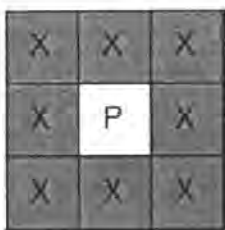
กระบวนการ Segmentation ถือเป็นขั้นตอนที่ยากที่สุดในกระบวนการวิเคราะห์ภาพ (Image Analysis) และกระบวนการ Segmentation นั้นจะแบ่งได้เป็น 2 ประเภท คือ

1. Similarity Properties ถูกใช้ในการรวมพิกเซลให้เป็นกลุ่มเดียวกัน
2. Discontinuity Of Pixel ถูกใช้ในการแยกพิกเซล ระหว่างพิกเซลที่เป็นวัตถุที่เราสนใจออกจาก Background

จาก Background

ในโครงการนี้จะใช้การ Segmentation ด้วยวิธีที่เรียกว่า Region Segmentation Method ซึ่งเป็นแบบ Discontinuity Of Pixel เป็นวิธีการแยกองค์ประกอบของภาพโดยดูจากตำแหน่งของพิกเซล และความเหมือนกันของคุณสมบัติของพิกเซลภายในพื้นที่ โดยถ้าพื้นที่ใดมีพิกเซลติดกันและมีคุณสมบัติเหมือนกันจะถูกจัดให้เข้ากลุ่มเดียวกันซึ่งการพิจารณาจะพิจารณาถึงความเข้มของพิกเซล

ในการขยายตัวของพิกเซล จำเป็นต้องมีกฎการตรวจสอบความเป็นอันหนึ่งอันเดียวกันของกลุ่มพิกเซลทุกๆ ระยะของการขยายตัว ในแต่ละระยะการขยายตัวจะมีการตรวจสอบว่ามีพิกเซลใดที่ยังไม่ได้จัดกลุ่ม คือ บริเวณพิกเซลทั้ง 8 (8 Neighbors) ที่เชื่อมต่อกับพิกเซลที่อยู่บริเวณขอบของกลุ่ม ถ้ามีค่าภายในขอบเขตที่กำหนดไว้ให้ถือว่าเป็นกลุ่มเดียวกัน แต่ถ้าไม่ใช่จัดว่าเป็น Noise ทันที



รูปที่ 2.9 ลักษณะของพิกเซล 8 Neighbors

2.2 หลักการพื้นฐานของการทรานสฟอร์ม

การลดข้อมูลด้วยวิธีการทรานสฟอร์มนั้น จะทำการแปลงข้อมูลอินพุตจากโดเมนหนึ่ง ให้อยู่ในรูปของโดเมนใหม่ โดยใช้วิธีการแปลงแบบต่าง ๆ เช่น การแปลงฟูรีเยร์ (Fourier Transform) ในการแปลงข้อมูลให้อยู่ในรูปสัมประสิทธิ์ของความถี่ จะทำให้ที่ความถี่ต่ำๆ มีพลังงานสูง และที่ความถี่สูงๆ พลังงานจะลดลงไป สำหรับข้อมูลภาพโดยทั่วไปนั้นจะประกอบด้วยส่วนที่เป็นความถี่สูง คือส่วนที่เป็นรายละเอียดหรือขอบภายในภาพ และส่วนที่เป็นความถี่ต่ำ คือข้อมูลส่วนที่เป็นพื้นของภาพ ดังนั้น เมื่อต้องการลดข้อมูลให้ได้มาก ค่าของพลังงานความถี่สูงจะถูกตัดทิ้งไป ทำให้รายละเอียดส่วนที่เป็นขอบในภาพขาดหายไป ภาพที่ได้จะขาดความคมชัด

การแปลงที่ใช้ในการลดข้อมูลภาพมีอยู่หลายวิธี เช่น การแปลงฟาสฟูรีเยร์ (Fast Fourier) การแปลงฮาร์ (Haar Transform) การแปลงคาร์ยูนเนนเกิฟ (Karchunen Loeve Transform) การแปลงวอลซ์-ฮาดามาร์ด (Walsh - Hadamard Transform) การแปลงคอสคริตโคซายน์ (Discrete Cosine Transform) การแปลงคอสคริตซายน์ (Discrete Sine Transform) เป็นต้น ซึ่งแต่ละแบบของการแปลงก็มีข้อดีข้อเสียต่างกัน แต่ที่นิยมใช้ในการลดขนาดของข้อมูลมากที่สุด คือ การแปลงคอสคริตโคซายน์ ตัวอย่างการใช้งานของการแปลงคอสคริตโคซายน์ คือ การลดขนาดของข้อมูลแบบ JPEG และ MPEG

2.2.1 สมการของการทรานสฟอร์ม

การทรานสฟอร์มทั้งหมดที่กล่าวถึงนี้จะกล่าวถึงการทรานสฟอร์มที่เป็นเชิงเส้นซึ่งสามารถเขียนสมการของการทรานสฟอร์มแบบ 1 มิติ (1D Forward Transform) ได้ดังนี้

$$\theta_n = \sum_{i=0}^{N-1} X_i a_{n,i} \quad (2.11)$$

- เมื่อ θ_n คือลำดับการทรานสฟอร์ม (Transform Sequence)
 X_i คือลำดับของอินพุต (Original Sequence)
 $a_{n,i}$ คือสัมประสิทธิ์การทรานสฟอร์ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถเขียนสมการของการอินเวอร์สทรานสฟอร์มแบบ 1 มิติ (1D Inverse Transform) ได้ดังนี้

$$X_n = \sum_{i=0}^{N-1} \theta_i b_{n,i} \quad (2.12)$$

จากสมการที่ (2.11) สามารถเขียนสมการการทรานสฟอร์ม 1 มิติ ในรูปเมตริกซ์ ได้ดังนี้

$$\theta = CX \quad (2.13)$$

จากสมการที่ (2.12) สามารถเขียนสมการการอินเวอร์สทรานสฟอร์ม 1 มิติ ในรูปเมตริกซ์ได้ดังนี้

$$X = D\theta \quad (2.14)$$

โดย เมตริกซ์ θ, X มีขนาด $N \times 1$
เมตริกซ์ C, D มีขนาด $N \times N$

จากข้อมูลแบบ 2 มิติ สามารถเขียนสมการของการทรานสฟอร์มแบบ 2 มิติ (2D Forward Transform) ได้ดังนี้

$$\Theta_{k,l} = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} X_{i,j} a_{k,i} a_{k,j} \quad (2.15)$$

เมื่อพิจารณาสมการที่ (2.15) สามารถเขียนสมการการทรานสฟอร์ม 2 มิติ ในรูปเมตริกซ์ ได้ดังนี้

$$\Theta = CXC^T \quad (2.16)$$

สามารถเขียนสมการการอินเวอร์สทรานสฟอร์ม 2 มิติ ในรูปเมตริกซ์ ได้ดังนี้

$$X = D\Theta D^T \quad (2.17)$$

โดย $D = C^{-1} = C^T$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือสามารถเขียนสมการการอินเวอร์สทรานสฟอร์ม 2 มิติ ในรูปเมตริกซ์ ได้ดังนี้

$$X = C^T \Theta C \quad (2.18)$$

- โดย Θ คือเอาต์พุตของการทรานสฟอร์ม เป็นเมตริกซ์ขนาด $N \times N$
 C, D คือสัมประสิทธิ์ของการทรานสฟอร์ม เป็นเมตริกซ์ขนาด $N \times N$
 X คืออินพุตของการทรานสฟอร์ม เป็นเมตริกซ์ขนาด $N \times N$

2.2.2 การแปลงแบบดิสครีตโคไซน์ (Discrete Cosine Transform)

การแปลงดิสครีตโคไซน์ คือ การแปลงข้อมูลที่อยู่ในสเปเชียล (Spatial Domain) ให้อยู่ในรูปแบบของโดเมนความถี่ เพื่อทำให้ค่าที่ได้ออกมาเป็นอิสระต่อกัน และลดขนาดของข้อมูลลงให้ค่าพลังงานส่วนใหญ่ของข้อมูลอยู่ในช่วงความถี่ต่ำ โดยการแปลงแบบดิสครีตโคไซน์ จะเป็นการแปลงแบบเชิงเส้นที่มีค่าสัมประสิทธิ์ในการแปลงคงที่ และค่าที่ได้ในโดเมนความถี่จะมีเฉพาะจำนวนจริงเท่านั้น ซึ่งข้อมูลส่วนใหญ่ที่จะใช้การแปลงแบบดิสครีตโคไซน์จะเป็นข้อมูลจากภาพนิ่ง หรือเป็นภาพเคลื่อนไหว อินพุตที่เข้ามาจะถูกแยกเป็นบล็อกเล็กๆ โดยส่วนมากแล้วขนาดของบล็อกจะมีค่าเป็นเลขยกกำลังสอง เช่น 4, 8, 16

ในกรณีของการแปลงดิสครีตโคไซน์ ค่าสัมประสิทธิ์ของการทรานสฟอร์มสามารถหาได้ดังนี้

$$[C]_{i,j} = \begin{cases} \sqrt{\frac{1}{N}} \cos \frac{(2j+1)i\pi}{2N} & i=0, j=0,1,\dots,N-1 \\ \sqrt{\frac{2}{N}} \cos \frac{(2j+1)i\pi}{2N} & i=1,2,\dots,N-1, j=0,1,\dots,N-1 \end{cases} \quad (2.19)$$

เมื่อ N คือ ขนาดของบล็อก ($N = 2^n, n = 2, 3, 4, \dots$)

จากสมการที่ (2.19) สามารถเขียนค่าสัมประสิทธิ์ของการแปลงดิสครีตโคไซน์ให้อยู่ในรูปของเมตริกซ์ ขนาด $N \times N$ เมื่อ $N = 5$ ได้ดังนี้

$$C = \begin{bmatrix} \frac{1}{\sqrt{5}} & \frac{1}{\sqrt{5}} & \dots & \frac{1}{\sqrt{5}} \\ \sqrt{\frac{2}{5}} \cos\left(\frac{\pi}{10}\right) & \sqrt{\frac{2}{5}} \cos\left(\frac{3\pi}{10}\right) & \dots & \sqrt{\frac{2}{5}} \cos\left(\frac{9\pi}{10}\right) \\ \dots & \dots & \dots & \dots \\ \sqrt{\frac{2}{5}} \cos\left(\frac{2\pi}{5}\right) & \sqrt{\frac{2}{5}} \cos\left(\frac{6\pi}{5}\right) & \dots & \sqrt{\frac{2}{5}} \cos\left(\frac{18\pi}{5}\right) \end{bmatrix} \quad (2.20)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากค่าสัมประสิทธิ์ของการทรานสฟอร์มในสมการที่ (2.20) สามารถเขียนแทนค่าฟังก์ชันตรีโกณมิติดังกล่าวด้วยตัวแปรได้ดังตารางที่ 2.1

ตารางที่ 2.1 ค่าสัมประสิทธิ์ของการทรานสฟอร์มในรูปแบบของฟังก์ชันตรีโกณมิติ

ค่าสัมประสิทธิ์	ฟังก์ชันโคไซน์ (cosine)
C_1	$\sqrt{\frac{2}{5}} \cos\left(\frac{\pi}{10}\right)$
C_2	$\sqrt{\frac{2}{5}} \cos\left(\frac{\pi}{5}\right)$
C_3	$\sqrt{\frac{2}{5}} \cos\left(\frac{3\pi}{10}\right)$
C_4	$\sqrt{\frac{2}{5}} \cos\left(\frac{2\pi}{5}\right)$

จากสมการที่ (2.13) เมื่อแทนค่าสัมประสิทธิ์การแปลงดิสครีตโคซายน์ ในสมการที่ (2.20) แล้วสามารถเขียนแสดงในรูปเมทริกซ์ได้ดังนี้

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{5}} & \frac{1}{\sqrt{5}} & \frac{1}{\sqrt{5}} & \frac{1}{\sqrt{5}} & \frac{1}{\sqrt{5}} \\ C_1 & C_3 & 0 & -C_3 & -C_1 \\ C_2 & -C_4 & -1 & -C_4 & C_2 \\ C_3 & -C_1 & 0 & C_1 & -C_3 \\ C_4 & -C_2 & 1 & -C_2 & C_4 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \end{bmatrix}$$

รูปที่ 2.10 เมทริกซ์สัมประสิทธิ์การแปลงดิสครีตโคซายน์

สมการที่ใช้สำหรับการแปลงดิสครีตโคซายน์ (Forward Discrete Cosine Transform) ยังสามารถเขียนสมการให้อยู่ในรูปของผลบวกของผลคูณของอินพุตกับค่าสัมประสิทธิ์ได้ดังนี้

1 มิติ

$$y(n) = e(n) \sum_{k=0}^{N-1} x(k) \cos\left[\frac{(2k+1)n\pi}{2N}\right] \quad ; \quad n = 0, 1, \dots, N-1 \quad (2.21)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2 มิติ

$$y(m,n) = \frac{4}{N^2} e(m)e(n) \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} x(k,l) \cos\left[\frac{(2k+1)n\pi}{2N}\right] \cos\left[\frac{(2l+1)m\pi}{2N}\right] \quad (2.22)$$

$$m = 0,1,\dots,N-1$$

$$n = 0,1,\dots,N-1$$

$$e(n) = \begin{cases} \frac{1}{\sqrt{2}} & ; n=0 \\ 1 & ; \text{Others} \end{cases}$$

$$e(m) = \begin{cases} \frac{1}{\sqrt{2}} & ; m=0 \\ 1 & ; \text{Others} \end{cases}$$

สมการการแปลงกลับของดิสครีตโคไซน์ (Inverse Discrete Cosine Transform) สามารถแสดง

ได้ดังนี้

1 มิติ

$$x(k) = \frac{2}{N} \sum_{n=0}^{N-1} e(n) y(n) \cos\left(\frac{(2k-1)n\pi}{2N}\right) ; k = 0,1,\dots,N-1 \quad (2.23)$$

2 มิติ

$$x(k,l) = \frac{2}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} e(m) e(n) y(m,n) \cos\left(\frac{(2k-1)m\pi}{2N}\right) \cos\left(\frac{(2l-1)n\pi}{2N}\right) \quad (2.24)$$

$$k = 0,1,\dots,N-1$$

$$l = 0,1,\dots,N-1$$

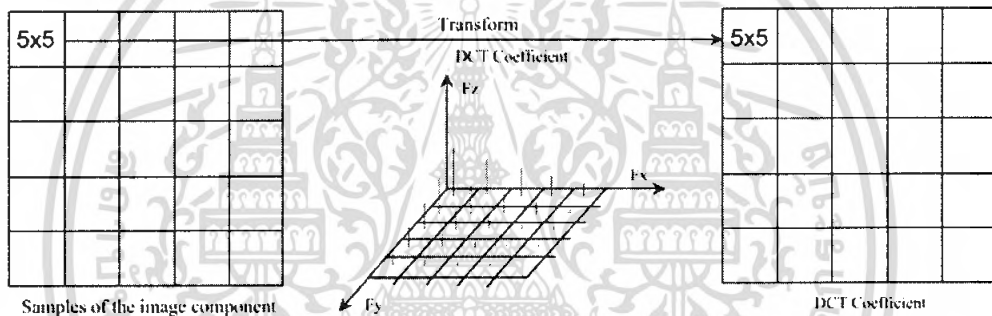
$$e(n) = \begin{cases} \frac{1}{\sqrt{2}} & ; n=0 \\ 1 & ; \text{Others} \end{cases}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$e(m) = \begin{cases} \frac{1}{\sqrt{2}} & ; m = 0 \\ 1 & ; \text{Others} \end{cases}$$

2.2.3 การแปลงดิสครีตโคซายน์ 2 มิติ (2 Dimensions Discrete Cosine Transform)

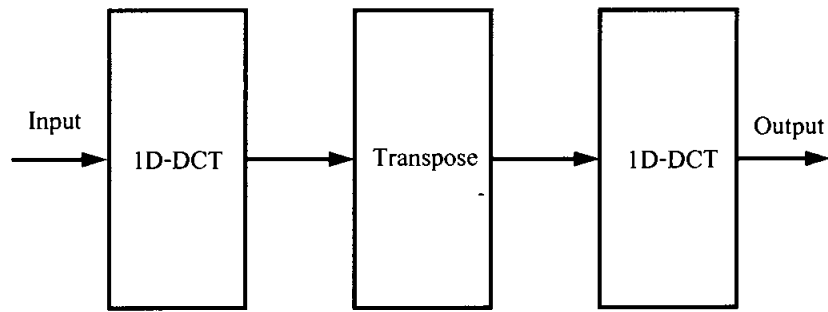
โครงการนี้จะทำการศึกษาข้อมูลภาพโดยยกตัวอย่างของข้อมูลภาพ gray-scale และไบนารี ซึ่งจะถูกแบ่งออกเป็นบล็อกย่อยๆ ขนาด $N \times N$ พิกเซล เพื่อที่จะทำการแปลงดิสครีตโคซายน์แบบ 2 มิติ (2D Discrete Cosine Transform) ทีละจุดจนครบทุกบล็อกทั้งหมดในข้อมูลภาพ ผลของการทรานสฟอร์มแต่ละบล็อกจะได้ข้อมูลที่มีความสำคัญสูง (เทอมความถี่ต่ำ) อยู่ในด้านมุมซ้ายบนโดยที่ตำแหน่ง (0, 0) เป็นองค์ประกอบของ DC ส่วนที่เหลือเป็นข้อมูลที่มีความสำคัญต่ำ (เทอมความถี่สูง) เป็นองค์ประกอบของ AC กระจายออกไปตามแกน x และ y ดังที่แสดงในรูปที่ 2.11



รูปที่ 2.11 ลักษณะของการแปลงดิสครีตโคซายน์ของบล็อกข้อมูลขนาด 5×5 พิกเซลจากข้อมูลทั้งหมด

จากผลการแปลงดิสครีตโคซายน์ของข้อมูลภาพแต่ละบล็อกที่แสดงในรูปที่ 2.11 จะเห็นว่าผลของข้อมูลที่มีความสำคัญสูง (มีค่ามาก) อยู่ในเทอมความถี่ต่ำ ส่วนเทอมความถี่สูง (มุมล่างด้านขวาของเมตริกซ์) จะมีค่าของข้อมูลน้อยมากเมื่อเทียบกับเทอมความถี่ต่ำที่มีความสำคัญมากกว่า แต่ก็ยังคงมีความสำคัญอยู่บ้าง ถ้านำข้อมูลที่ได้จากการแปลงดิสครีตโคซายน์ ไปผ่านการควอนไทซ์ ทำให้ข้อมูลบริเวณความถี่สูงมีค่าเข้าใกล้ศูนย์ ช่วยให้สามารถลดขนาดของข้อมูลลงได้

การแยกการแปลงดิสครีตโคซายน์แบบ 2 มิติ ให้เป็นการแปลงดิสครีตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง อธิบายโดยใช้บล็อกไดอะแกรมดังแสดงในรูปที่ 2.12

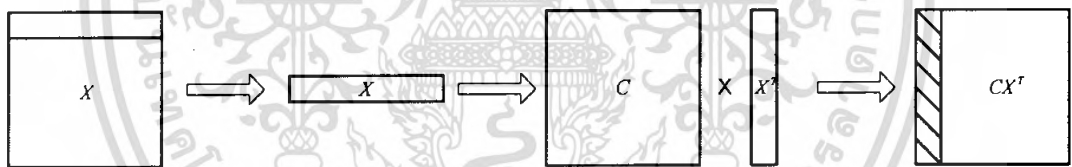


รูปที่ 2.12 บล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์แบบ 2 มิติ

การแยกการแปลงดิสครีตโคซายน์แบบ 2 มิติ ให้เป็นการแปลงดิสครีตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง สามารถอธิบายการทำงาน ได้ดังนี้

1. การแปลงดิสครีตโคซายน์แบบ 1 มิติ ตามแนวอนของเมตริกซ์

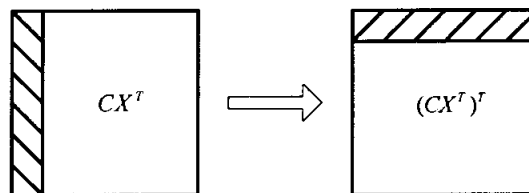
การนำข้อมูลตามแนวอน 1 มิติ ขนาด $1 \times N$ จากข้อมูลเริ่มต้น 2 มิติ ขนาด $N \times N$ มาทำการแปลงดิสครีตโคซายน์แบบ 1 มิติ ทีละแถว จนครบทั้งหมด N แถว สามารถแสดงบล็อกไดอะแกรมของการแปลงดิสครีตโคซายน์แบบ 1 มิติ ตามแนวอนในรูปของเมตริกซ์ได้ดังรูปที่ 2.13



รูปที่ 2.13 บล็อกไดอะแกรมการแปลงดิสครีตโคซายน์แบบ 1 มิติ ตามแนวอน

2. การทรานสโพสบล็อกข้อมูลที่ได้จากการแปลงดิสครีตโคซายน์แบบ 1 มิติ

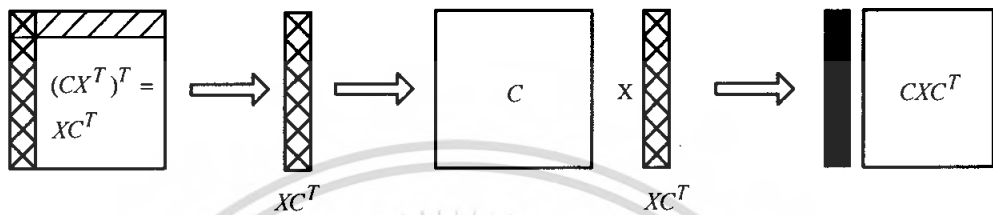
การนำข้อมูลตามแนวอนที่ได้จากการแปลงดิสครีตโคซายน์แบบ 1 มิติ มาทำการทรานสโพสเพื่อนำเข้าไปเป็นอินพุตของการแปลงดิสครีตโคซายน์แบบ 1 มิติตามแนวตั้ง สามารถแสดงบล็อกไดอะแกรมของการทรานสโพส ในรูปของเมตริกซ์ได้ดังรูปที่ 2.14



รูปที่ 2.14 บล็อกไดอะแกรมการทรานสโพสบล็อกข้อมูล

3. การแปลงดิสคริตโคซายน์แบบ 1 มิติ ตามแนวตั้งของเมตริกซ์

การนำข้อมูลตามแนวตั้ง 1 มิติ ขนาด $1 \times N$ จากข้อมูลที่ได้จากการแปลงดิสคริตโคซายน์แบบ 1 มิติ ในส่วนที่ 1 โดยเริ่มต้นที่ 2 มิติ ขนาด $N \times N$ มาทำการแปลงดิสคริตโคซายน์ 1 มิติทีละหลัก จนครบทั้งหมด N หลัก สามารถแสดงบล็อกไดอะแกรมของการแปลงดิสคริตโคซายน์แบบ 1 มิติ ตามแนวตั้งในรูปของเมตริกซ์ได้ดังรูปที่ 2.15



รูปที่ 2.15 บล็อกไดอะแกรมการแปลงดิสคริตโคซายน์แบบ 1 มิติ ตามแนวตั้ง

ผลลัพธ์จากการแปลงดิสคริตโคซายน์แบบ 2 มิติ โดยใช้การแปลงดิสคริตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง คือ

$$\begin{aligned}
 Y &= C(CX^T)^T \\
 Y &= C(X^T)^T C^T \\
 Y &= CXC^T
 \end{aligned}
 \tag{2.25}$$

จากการแปลงดิสคริตโคซายน์แบบ 2 มิติ โดยใช้การแปลงดิสคริตโคซายน์แบบ 1 มิติ จำนวน 2 ครั้ง ดังในสมการที่ (2.25) จะได้ผลลัพธ์เท่ากับการแปลงดิสคริตโคซายน์แบบ 2 มิติ ในสมการที่ (2.16)

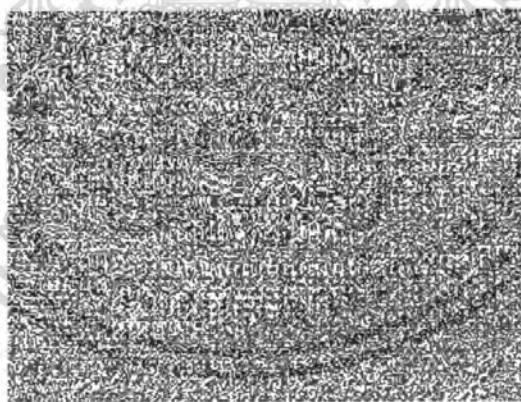
4. การนำการแปลงแบบดิสครีตโคซายน์ไปใช้งาน

ตัวอย่างการประมวลผลภาพด้วยการแปลงดิสครีตโคซายน์ จากภาพที่ 2.16 คือภาพต้นแบบที่จะนำมาทำการแปลงดิสครีตโคซายน์ ซึ่งมีขนาดภาพ 240×320 พิกเซล



รูปที่ 2.16 ภาพต้นแบบก่อนทำการแปลงดิสครีตโคซายน์

หลังจากการแปลงดิสครีตโคซายน์ด้วยบล็อกย่อยๆ ขนาด 5×5 พิกเซล ค่าสัมประสิทธิ์ของการแปลงดิสครีตโคซายน์ดังรูปที่ 2.17



รูปที่ 2.17 ภาพของค่าสัมประสิทธิ์ดิสครีตโคซายน์หลังทำการแปลง

จากรูปที่ 2.17 เก็บค่าของสัมประสิทธิ์ดิสครีตโคซายน์ในบางค่า เพื่อจำลองลดข้อมูลของภาพ โดยจะนำบล็อกย่อยๆ ขนาด 5×5 พิกเซล มาคูณกับรูปที่ 2.18 โดยทำการคูณในตำแหน่งที่เป็นพิกเซลต่อพิกเซล โดยบล็อกขนาด 5×5 พิกเซล มีข้อมูลดังรูปที่ 2.17

1	1	1	0	0
1	1	0	0	0
1	0	0	0	0
0	0	0	0	0
0	0	0	0	0

รูปที่ 2.18 บล็อกขนาด 5×5 ที่นำมาคูณกับรูปที่ 2.17

จากรูปที่ 2.18 การเก็บค่าสัมประสิทธิ์ดิสครีตโลชาวน์เพียงบางค่าที่สำคัญ โดยการเก็บค่าสัมประสิทธิ์ที่อยู่มุมขวาบน ไปเพียง 6 ค่าเท่านั้น หลังการคูณด้วยบล็อกรูปที่ 2.18 ได้ผลลัพธ์ดังรูปที่ 2.19



รูปที่ 2.19 ผลลัพธ์ของสัมประสิทธิ์จากการคูณด้วยบล็อกในรูปที่ 2.18

หลังจากนำค่าสัมประสิทธิ์ทั้งหมดของภาพดังรูปที่ 2.19 มาทำการแปลงกลับจะได้ผลลัพธ์ดังรูปที่ 2.20



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เฉพาะในกรณีฉุกเฉินเท่านั้น ไม่ควรนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากการแปลงกลับจะเห็นได้ว่าภาพจะเพี้ยนไป แต่ก็ยังคงสภาพภาพต้นฉบับไว้ และสิ่งที่สำคัญก็คือ ขนาดของข้อมูลภาพจะลดลงอย่างมากเมื่อเทียบกับภาพต้นแบบ ซึ่งในกรณีนี้ภาพต้นแบบมีขนาด 240×320 ต้องทำการเก็บข้อมูลภาพถึง 76,800 พิกเซล เมื่อผ่านการแปลงดิสครีตโคไซน์แล้ว การเก็บข้อมูลภาพทำการเก็บเพียง $\left(\frac{240}{5}\right) \times \left(\frac{320}{5}\right) \times 6 = 18,432$ พิกเซลเท่านั้น หรือคิดเป็นเปอร์เซ็นต์เท่ากับ $\left(\frac{18,432}{76,800}\right) \times 100 = 24\%$

2.2.4 การแปลงฮัฟ (Hough Transform)

Hough Transform เป็นวิธีการในการตรวจจับเส้นตรงที่อยู่ในภาพ โดยจะเป็นการตรวจหาเส้นตรงแต่ละเส้นที่แฝงอยู่ในภาพ โดยมีข้อกำหนดว่า ก่อนหน้าที่จะนำภาพเข้ากระบวนการ Hough Transform ใต้นั้น ภาพตั้งต้นจะต้องผ่านเทคนิคการหาขอบภาพมาก่อน อาจจะใช้เทคนิคของ Sobel Edge Detection หรือ Prewitt Edge Detection หรือเทคนิคอื่นๆ ก็ได้

แนวความคิดเบื้องต้นของ Hough Transform คือ ที่จุดใดๆ ของภาพ (x, y) จะสามารถมีเส้นตรงผ่านได้เป็นจำนวนไม่จำกัด ซึ่งเส้นตรงแต่ละเส้นจะมีสมการเส้นตรงของแต่ละเส้นเป็นของตัวเอง และเส้นตรงใดๆ จะเป็นเส้นตรงเดียวกันย่อมมีสมการเส้นตรงเป็นเส้นเดียวกัน จากสมการเส้นตรง

$$y = mx + b \quad (2.26)$$

โดย m คือความชันของเส้นตรงและ b คือจุดตัดของเส้นตรงกับแกน y

จุดใดๆ ที่อยู่บนเส้นตรงเดียวกัน จะมีค่า m และค่า b เท่ากันเสมอ นั่นคือ จากสมการที่ (2.26) สามารถเขียนได้เป็น

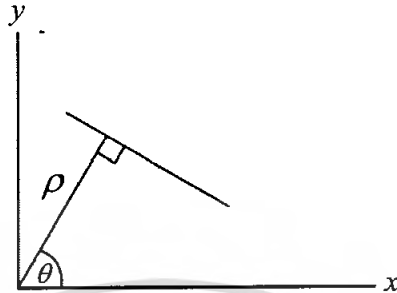
$$b = y - mx \quad (2.27)$$

หากทำการแปลงข้อมูลจากพิกัด (x, y) ให้อยู่ในพิกัด (m, b) แล้ว จะมีความสัมพันธ์ใหม่เกิดขึ้น นั่นคือหากจุดใดๆ อยู่บนเส้นตรงเดียวกัน ย่อมต้องมีค่า m และ b เท่ากัน และถ้าแปลงมาอยู่ในพิกัด (m, b) จะอยู่ในอาร์เรย์ช่องเดียวกัน หากใช้อาร์เรย์ (m, b) นี้เก็บค่าความถี่ของจุดในช่อง แต่ละช่องของ (m, b) จะเป็นจำนวนจุดที่อยู่บนเส้นตรงที่มีความชัน m และ b เท่ากัน ซึ่งอาร์เรย์พิกัด (m, b) นี้เรียกว่า Hough Space

ดังนั้น Hough Space คือ อาร์เรย์ที่ใช้เก็บค่าความถี่ของจุดที่อยู่บนเส้นตรงเดียวกันในภาพ โดยวิธีการค้นหาจะต้องตรวจสอบดูที่ทุกๆ จุดในภาพ ว่ามีเส้นตรงอยู่ที่ใดบ้าง แล้วทำการบวกเพิ่มในตารางความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการที่ (2.27) นี้สังเกตว่ายังใช้งานไม่ได้จริง เนื่องจากกรณีที่เส้นตรงทำมุมเป็น 90 องศา จะทำให้ค่าความชันมีค่าเป็นอนันต์ ดังนั้นต้องเปลี่ยนวิธีการในการเก็บค่าใหม่ โดยใช้การคำนวณบนพิกัดเชิงขั้วดังรูปที่ 2.21



รูปที่ 2.21 เส้นตรงในพิกัดเชิงขั้ว

สมการเส้นตรงสำหรับพิกัดเชิงขั้วมีลักษณะดังนี้

$$\rho = x \cos \theta + y \sin \theta \quad (2.28)$$

โดย ρ คือระยะทางที่ใกล้ที่สุดจากเส้นตรงถึงจุดกำเนิด หรือเป็นความยาวของเส้นที่ลากจากจุดกำเนิดไปตั้งฉากกับเส้นตรงนั่นเอง
 θ คือมุมระหว่างเส้นที่ลากจากจุดกำเนิดไปตั้งฉากกับเส้นตรงกับแกน x

Hough Space สำหรับพิกัดเชิงขั้วนี้คือ (ρ, θ) การเก็บข้อมูลในลักษณะเช่นนี้จะทำให้สามารถเก็บข้อมูลได้ทั้ง 360 องศา สำหรับวิธีการเก็บข้อมูลนั้น จะทำได้โดยการพิจารณาที่แต่ละพิกเซลในภาพตั้งต้นซึ่งผ่านการหาขอบภาพมาแล้ว ทำให้ภาพมีลักษณะเป็น Binary Image หากตำแหน่งพิกเซลใดมีค่าเป็น 1 แสดงว่าพิกเซลนั้นอาจเป็นส่วนประกอบของเส้นใดเส้นหนึ่งในภาพ ทำการแทนค่าสมการที่ (2.28) โดยแทนค่า x และ y ซึ่งเป็นตำแหน่งของพิกเซลนั้นๆ ในแนวแกน x และ y ตามลำดับ แล้วจึงแทนค่ามุม θ ที่ต้องการค้นหาทั้งหมด โดยมุม θ จะมีค่าระหว่าง 1 ถึง 360 องศา แบ่งช่วงละ 1 องศา แล้วทำการหาค่า ρ ออกมาตามสมการ

ถ้าเซตของจุดกลุ่มหนึ่งเป็นจุดที่อยู่บนเส้นตรงเดียวกัน จะทำให้การคำนวณค่าตามสมการที่ (2.28) นี้ได้ค่า ρ และ θ เท่ากัน นั่นคือเซตของจุดกลุ่มนี้จะตกในตำแหน่ง (ρ, θ) เดียวกันทั้งหมด ทำให้อาร์เรย์ (ρ, θ) ในตำแหน่งนี้มีความถี่สูง หากนำ Hough Space มาเขียนเป็นภาพจะเห็นว่าในตำแหน่งนั้นจะมีลักษณะเป็นจุดสว่างขึ้นมา ดังรูปที่ 2.22



(ก)



(ข)



(ค)

รูปที่ 2.22 การทำ Hough Transform

(ก) ภาพตั้งต้นในรูปแบบ Gray Scale

(ข) ภาพหลังจากทำ Edge Detection และทำการเทรซโฮสต์ให้เป็นภาพไบนารี

(ค) ภาพของ Hough Space โดยใช้ขอบเขตเท่ากับ $(0 \text{ to } \text{Max}(\rho), 0^\circ \text{ to } 90^\circ)$ ที่ได้จากการ Transform ของภาพ (ข)

สำหรับการกำหนดมุม และการแบ่งช่วงขึ้นอยู่กับผู้ใช้งานต้องการความละเอียดมากน้อยเพียงใด และขอบเขตเท่าใด เมื่อได้ค่า ρ แล้ว จะนำไปบวกค่าเพิ่มขึ้น 1 ใน Hough Space ในตำแหน่งที่ (ρ, θ) ที่คำนวณได้ ในโครงงานนี้จะกำหนดขอบเขต Hough Space ให้เท่ากับ $(0 \text{ to } \text{Max}(\rho), 0^\circ \text{ and } 90^\circ)$

2.3 โครงข่ายประสาทเทียม (Neural Network)

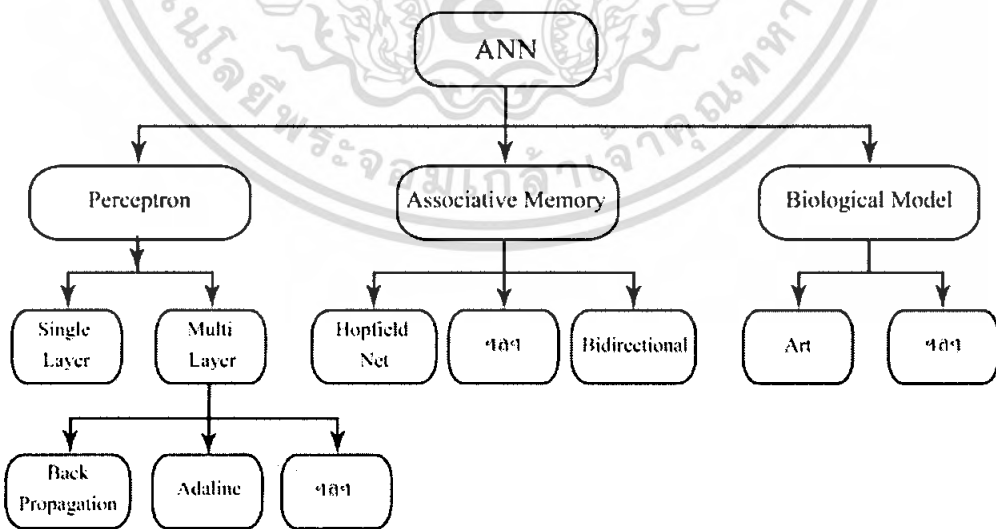
โครงข่ายประสาทเทียม หรือ Artificial Neural Network (ANN) เป็นชื่อแขนงหนึ่งในสาขาวิทยาการคอมพิวเตอร์ ในการศึกษาเลียนแบบการทำงานของสมองมนุษย์ เฉพาะงานที่คอมพิวเตอร์ยังทำไม่ได้ หรือทำได้ไม่ดี อาทิเช่น การอ่านหนังสือ การจัดหมวดหมู่สิ่งของ เป็นต้น ซึ่งในปัจจุบันนี้นักวิทยาศาสตร์ทางสาขาชีววิทยาและสาขาที่เกี่ยวข้องกับการศึกษารูปแบบการทำงานของสมองมนุษย์ก็ยังไม่สามารถระบุรายละเอียดทั้งหมดของกลไกการทำงานของสมองมนุษย์ได้ ส่งผลให้นักวิชาการทางวิทยาการคอมพิวเตอร์เสนอรูปแบบและวิธีการจำลองการทำงานของสมองมนุษย์ออกมามากมาย โดยมีเป้าหมายที่ความสามารถของแบบจำลองให้เหมือนการทำงานด้วยมนุษย์มากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิพนธ์ให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

องค์ประกอบของสมองมนุษย์นั้นประกอบด้วยเซลล์ประสาท หรือนิวรอล ประมาณ 10^{11} ตัว โดยที่เซลล์ประสาทหนึ่งๆ มีเส้นเชื่อมโยงต่อไปยังเซลล์ประสาทอื่นๆ ประมาณ 10^4 ตัว และเวลาในการเปลี่ยนข้อมูลระหว่างนิวรอล อยู่ในลำดับของ 10^{-3} วินาที นับว่าช้ามากเมื่อเทียบกับเวลาในการเปลี่ยนข้อมูลของคอมพิวเตอร์ เพียง 10^{-10} วินาทีเท่านั้น อย่างไรก็ตามสมองมนุษย์สามารถประมวลผลได้เร็วมาก เช่น คนเราสามารถจำหน้าแม่ของตนเองได้ภายในเวลา 10^{-4} วินาที ซึ่งน่าจะต้องผ่านการทำงานเป็นร้อยๆ ชั้นตอนกว่าที่จะได้คำตอบว่านี่คือหน้าของแม่เรา และเมื่อเทียบกับเวลาในการทำงานของแต่ละนิวรอลแล้ว จึงเกิดการตั้งสมมติฐานขึ้นว่า การทำงานของนิวรอลในสมองมนุษย์นั้น น่าจะใช้การประมวลผลแบบขนานขั้นสูง (Highly Parallel Processes) ดังนั้นการที่จะจำลองเซลล์ประสาทและการต่อเชื่อมโยงทั้งหมดของสมองมนุษย์คงจะต้องใช้คอมพิวเตอร์ขนาดใหญ่และสมรรถนะสูงมากและน่าจะต้องมีการนำการคำนวณแบบขนานเข้ามาร่วมด้วย

การนำ ANN มาประยุกต์ใช้งานนั้นแบ่งเป็น 2 ขั้นตอนหลัก คือขั้นตอนการฝึกหัด (Training) หรือเรียนรู้ (Learning) และขั้นตอนการทดสอบ (Testing) หรือใช้งานจริง ซึ่งในขั้นตอนการฝึกหัดนั้นมีการพัฒนาและเสนอรูปแบบจำลองการเรียนรู้ออกมามากมายซึ่งพอจะแบ่งออกได้เป็น 2 กลุ่มหลัก คือการเรียนรู้แบบมีผู้สอน (Supervise Learning Algorithm) และการเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning Algorithm)

ในปัจจุบันนี้มีแบบจำลองของ ANN มากทั้งที่เป็นแบบมีผู้สอน และไม่มีผู้สอน นอกจากนี้ยังมีบางรูปแบบที่ผสมผสานทั้งสองรูปแบบเข้ามามีใช้ด้วยกัน ซึ่งแบบจำลองที่ได้รับความนิยมค่อนข้างมากแบ่งออกได้เป็น 3 กลุ่มใหญ่ๆ คือ เพอร์เซพตรอน (Perceptron) การเชื่อมโยงหน่วยความจำ (Associative Memory) และแบบจำลองทางชีววิทยา (Biological Model) ดังรูปที่ 2.23

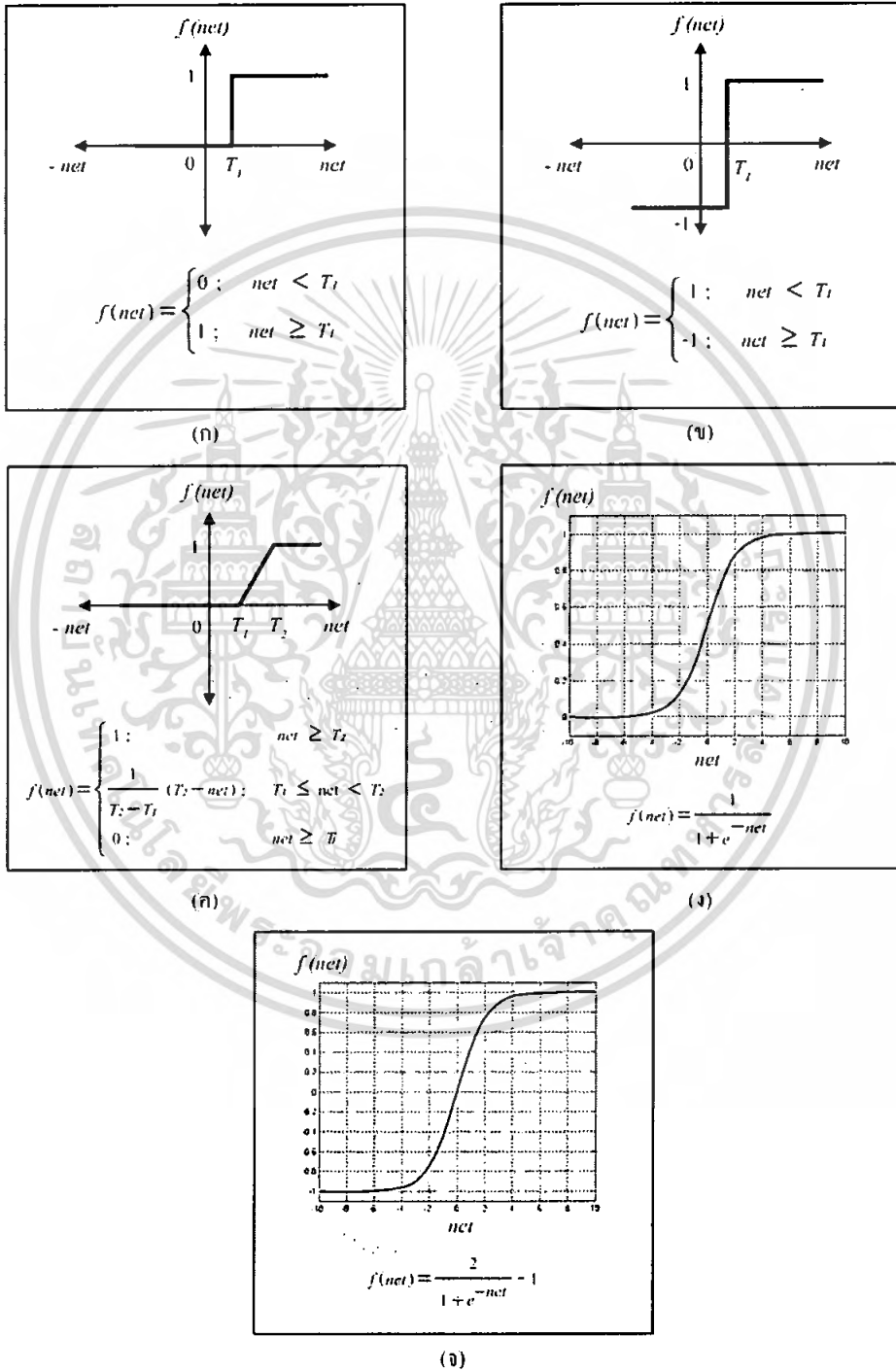


รูปที่ 2.23 แบบจำลองหลักๆ ของ ANN

2.3.1 ฟังก์ชันกระตุ้นความสนใจ (Activation Function)

Activation Function มีอยู่ด้วยกัน 2 ชนิดคือ ชนิดที่เป็นเชิงเส้นและไม่เป็นเชิงเส้น ดังรูปที่ 2.24

กำหนดให้ $net = w'x = \sum_{i=1}^n w_i x_i$ และ T_1, T_2 เป็นค่าเทรชโฮลด์



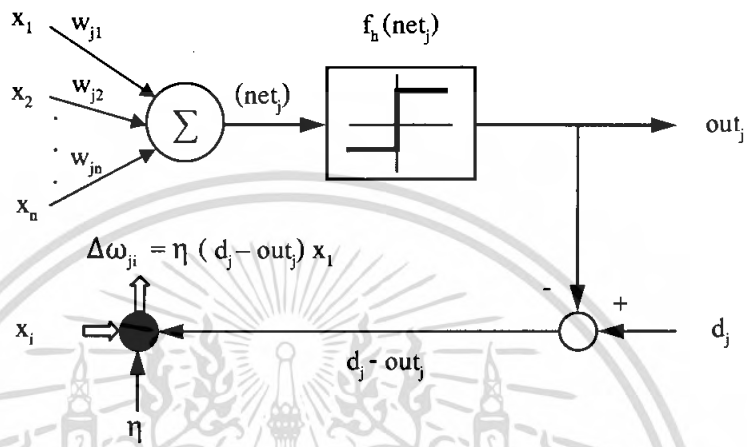
รูปที่ 2.24 Activation Function ชนิดต่างๆ (ก) Threshold Logic (ข) Bipolar Binary (ค) Linear Threshold

(ง) Sigmoid (จ) Bipolar Continuous หรือ Hyperbolic Tangent Function

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ผู้ใดเห็นไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 กฎการเรียนรู้ของเพอร์เซพตรอน

เน็ตเวิร์คแบบเลเยอร์เดียวจะมีโครงสร้างไม่ซับซ้อน สามารถเรียนรู้ที่จะจดจำและแยกแยะแพทเทิร์นอย่างง่ายได้ ทฤษฎีที่จะนำมาใช้กับเน็ตเวิร์คแบบเลเยอร์เดียวคือ กฎการเรียนรู้ของเพอร์เซพตรอนแบบไม่ต่อเนื่อง (Discrete Perceptron Learning Rule) ถูกค้นพบโดย Rosenblatt 1958 ดังในรูปที่ 2.25



รูปที่ 2.25 โครงสร้างนิวรอลเน็ตเวิร์คแบบเพอร์เซพตรอน

จากรูปที่ 2.24 ผลรวมของค่าอินพุตแพทเทิร์น (x_1, \dots, x_n) ในเลเยอร์ i คูณกับค่าถ่วงน้ำหนัก w_{ij} จะถูกส่งผ่านไปยังแอกติเวชันฟังก์ชันแบบฮาร์ดลิมิต ได้ค่าเอาต์พุตที่ได้จากการคำนวณ out_j ในเลเยอร์ j ซึ่งค่าเอาต์พุตจะมีค่าเป็น 1 หรือ 0 ค่าใดค่าหนึ่งเท่านั้น จากนั้นค่าเอาต์พุตจะถูกนำไปคำนวณเพื่อปรับค่าตัวเลขน้ำหนักอีกครั้งหนึ่ง

$$net_j = \sum_{i=1}^n x_i w_{ji} + \theta_j \quad (2.29)$$

$$out_j = f_h(net_j) \quad (2.30)$$

การปรับค่าตัวเลขน้ำหนักจะได้ว่า

$$\Delta w_{ji} = \eta (d_j - out_j) x_i \quad (2.31)$$

$$= \eta \left[d_j - f_h \left(\sum_i x_i w_{ji} + \theta_j \right) \right] \quad (2.32)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และกำหนดตัวแปรค่าผิดพลาดการเรียนรู้ err ดังนี้

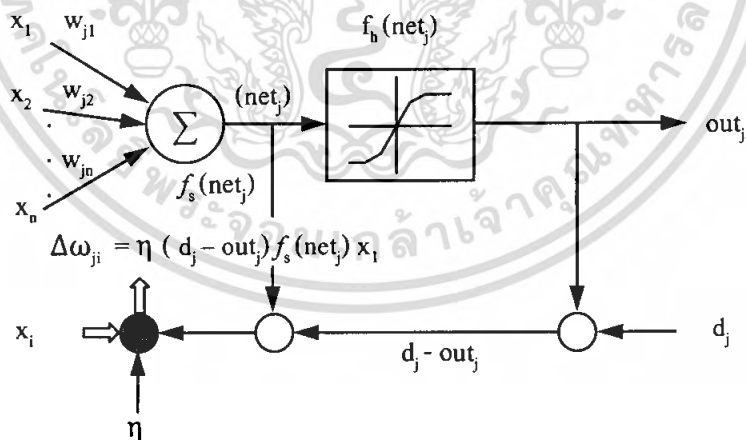
$$err = d_j - out_j \quad (2.33)$$

- โดย f_h คือแอกติเวชันฟังก์ชัน แบบฮาร์ดลิมิต
 θ_j คือค่าเทรชโฮลด์สำหรับเลเยอร์ j
 Δw_{ji} คือค่าตัวเลขน้ำหนักระหว่างเลเยอร์ i และ j
 η คือค่าคงที่การเรียนรู้
 d_j คือค่าเอาต์พุตจริงที่ต้องการในเลเยอร์ j

สังเกตค่า Δw_{ji} จะเท่ากับ 0 ก็ต่อเมื่อค่า out_j มีค่าเข้าใกล้ d_j

2.3.3 กฎการเรียนรู้ของเดลต้า

กฎการเรียนรู้ของเดลต้า (Delta Learning rule) ถูกค้นพบโดย McClelland และ Rumelhart (1986) โดยทฤษฎีนี้อาจเรียกอีกอย่างว่า กฎการเรียนรู้แบบต่อเนื่องของเพอร์เซพตรอน (Continues Perceptron Learning Rule) โดยจะใช้แอกติเวชันฟังก์ชันแบบต่อเนื่องของซิกมอยด์ ดังรูปที่ 2.24 (ง) ซึ่งอาศัยการเรียนรู้แบบมีครูสอนดังรูปที่ 2.26



รูปที่ 2.26 โครงสร้างกฎการเรียนรู้ของเดลต้า

ทฤษฎีการเรียนรู้นี้สามารถพิสูจน์ได้จากเงื่อนไขของค่าความผิดพลาดกำลังสองต่ำสุด E (Least Square Error) ระหว่างค่าเอาต์พุตจริงที่ต้องการ d_j และเอาต์พุตที่ได้จากการคำนวณ out_j สามารถแสดงเป็นสมการได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$E = \frac{1}{2} \sum_j (d_j - out_j)^2 \tag{2.34}$$

ซึ่งเท่ากับ

$$= \frac{1}{2} \sum_j \left[(d_j f'(net_j))^2 \right] \tag{2.35}$$

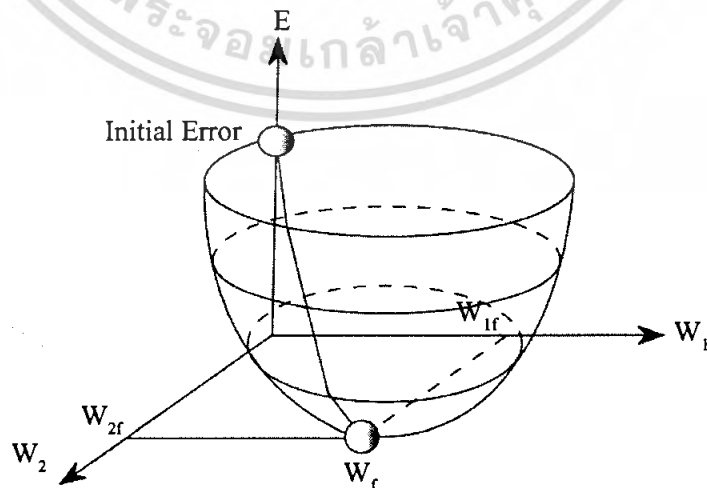
เพื่อให้ได้ค่าผิดพลาดต่ำสุดควรปรับเปลี่ยนตัวเลขน้ำหนักในทิศทางลบของ Gradient descent ได้ว่า

$$\Delta w_{ji}(t) = \eta (-\nabla E) \tag{2.36}$$

$$= \eta f'_s(net_j)(d_j - out_j)x_i \tag{2.37}$$

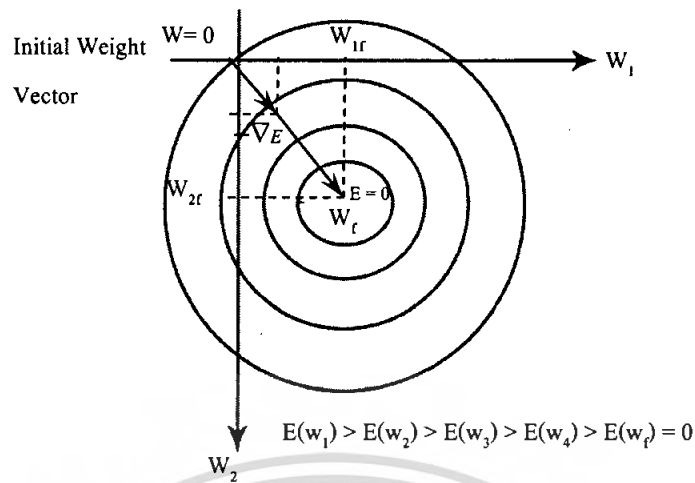
- โดย $\Delta w_{ji}(t)$ คือค่าตัวเลขน้ำหนักที่เปลี่ยนไปตามเวลา t ระหว่างเลขอร์ i และ j
- η คือค่าคงที่การเรียนรู้
- f_s คือแอกติเวชันฟังก์ชันแบบซิกมอยด์
- f'_s คือการหาอนุพันธ์อันดับหนึ่งของแอกติเวชันฟังก์ชันแบบซิกมอยด์
- ∇E คือ Gradient vector ของค่าผิดพลาด E

กราฟที่เกิดขึ้นตามทฤษฎี Gradient descent ในทิศทางลบ แสดงในรูปที่ 2.27 โดยกำหนดค่าถ่วงน้ำหนักเริ่มต้น w_1 และ w_2 เป็น 0 พิจารณากราฟที่เกิดขึ้นระหว่างค่าผิดพลาด E และค่าถ่วงน้ำหนัก w_1 และ w_2 จะเห็นว่าเมื่อเวลาผ่านไป ค่าถ่วงน้ำหนักจะถูกปรับเปลี่ยนไปจนกระทั่งได้ค่าผิดพลาดต่ำสุด E เท่ากับ ศูนย์ เมื่อค่าถ่วงน้ำหนักเท่ากับ w_r



(ก) กราฟค่าผิดพลาดต่ำสุดขณะสอนนิเวรอลเน็ตเวิร์คเปลี่ยนไปตามเวลา t

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

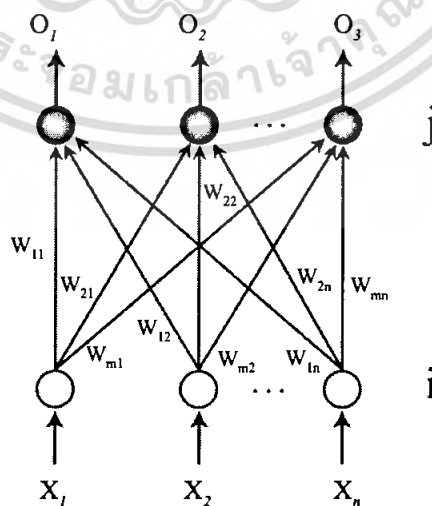


(จ) วงรอบแบบชั้นของค่าผิดพลาดต่ำสุดในทิศทาง gradient

รูปที่ 2.27 ความสัมพันธ์ระหว่างค่าผิดพลาดต่ำสุดและค่าถ่วงน้ำหนัก

2.3.4 นิวรอลเน็ตเวิร์คแบบเลเยอร์เดียว (Neural Network singlelayer)

วิธีการทำงานของนิวรอลเพียง โหนดเดียวซึ่งเทียบได้กับเซลล์เพียงเซลล์เดียว หากประกอบกันเป็นหลายเซลล์ จะต้องให้นิวรอลแต่ละตัวเชื่อมต่อกันทำให้เกิดเป็นเน็ตเวิร์คที่มีลักษณะเป็นชั้นๆ เรียกว่าเลเยอร์ (Layer) ในหนึ่งเลเยอร์สามารถมีจำนวนนิวรอลได้มากกว่า 1 นิวรอลดังแสดงในรูปที่ 2.28 ภายในเลเยอร์เดียวกันจะไม่มี การเชื่อมต่อกัน และนิวรอลที่มีเลเยอร์สูงกว่าจะรับค่ามาจากเอาต์พุตของเลเยอร์ที่อยู่ต่ำกว่าเพื่อเป็นอินพุตของตัวเอง จากลักษณะดังกล่าวจะเห็นว่า คุณสมบัติของนิวรอลทุกตัวที่อยู่ในเลเยอร์ที่ต่ำกว่าจะถูกถ่ายทอดไปยังนิวรอลในชั้นถัดไป



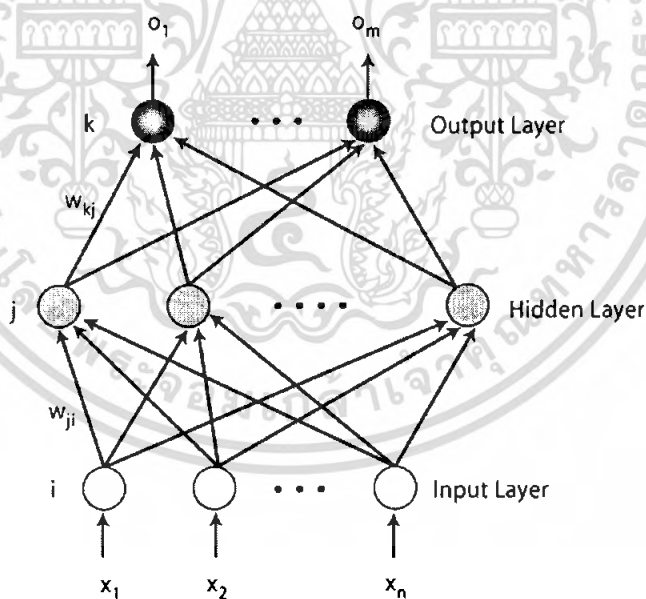
รูปที่ 2.28 นิวรอลเน็ตเวิร์คแบบเลเยอร์เดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.28 แสดงลักษณะของนิวรอลที่ประกอบกันเป็นนิวรอลเน็ตเวิร์คแบบเลเยอร์เดียว โดยในเลเยอร์แรกหรืออินพุตเลเยอร์โดยแท้จริงแล้วไม่ใช้นิวรอล เพราะไม่มีการคำนวณอยู่ภายในแต่จะแสดงลักษณะการกระจายของค่าอินพุตให้กับนิวรอลในชั้นถัดไปเท่านั้น

2.3.5 นิวรอลเน็ตเวิร์คแบบหลายเลเยอร์ (Neural Network Multilayer)

โครงสร้างของนิวรอลเน็ตเวิร์คแบบหลายเลเยอร์ ประกอบด้วยเลเยอร์ตั้งแต่ 2 เลเยอร์ขึ้นไป ในลักษณะของเน็ตเวิร์คที่มีขนาดใหญ่และซับซ้อนกว่าเน็ตเวิร์คแบบเลเยอร์เดียวมาก อาจมองว่าเน็ตเวิร์คแบบหลายเลเยอร์เกิดจากการรวมกันของเน็ตเวิร์คแบบเลเยอร์เดี่ยวก็ได้ เอาต์พุตแพทเทิร์นในเลเยอร์หนึ่งจะเป็นค่าอินพุตของเลเยอร์ถัดไป ซึ่งเลเยอร์ระหว่างอินพุตและเอาต์พุตถูกเรียกว่า ฮิดเดนเลเยอร์ (Hidden layer) หรือเลเยอร์ภายใน (Internal layer) โดยค่าเอาต์พุตในเลเยอร์สุดท้ายจะถูกเปรียบเทียบกับค่าเอาต์พุตจริงที่ต้องการ (Desired output) จากผู้สอน (Teacher) ผลลัพธ์ของค่าผิดพลาดในขณะฝึกสอนสามารถนำไปใช้ปรับค่าตัวเลขน้ำหนักของเน็ตเวิร์คให้เปลี่ยนแปลงในทิศทางที่จะใช้แยกกลุ่มของแพทเทิร์น และโครงสร้างของเน็ตเวิร์คให้เปลี่ยนแปลงในทิศทางที่จะใช้แยกกลุ่มของแพทเทิร์นและโครงสร้างของเน็ตเวิร์คในลักษณะนี้ถูกเรียกว่า โครงข่ายแบบป้อนไปหน้า (Feed-forward network) เนื่องจากสามารถสร้างเป็นเน็ตเวิร์คแบบหลายเลเยอร์ได้ ซึ่งจะตรงกันข้ามกับโครงข่ายแบบป้อนกลับ (Feedback network)



รูปที่ 2.29 โครงสร้างของนิวรอลเน็ตเวิร์คแบบหลายเลเยอร์

รูปที่ 2.29 แสดงสถาปัตยกรรมพื้นฐานของเน็ตเวิร์คแบบหลายเลเยอร์ มีเลเยอร์ภายในโครงข่ายเพียงเลเยอร์เดียว วงกลมในรูปแสดงโหนดหรือนิวรอลและลูกศรจะแสดงทิศทางของการเชื่อมต่อของการเคลื่อนที่ของสัญญาณระหว่างนิวรอลในแต่ละเลเยอร์ ซึ่งจะถูกแทนด้วยค่าถ่วงน้ำหนักหรือน้ำหนักระหว่างนิวรอลและโครงข่ายแบบหลายเลเยอร์นี้ จะถูกนำไปใช้กับอัลกอริทึมของการแพร่กระจายกลับ (Back Propagation) เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.5.1 การแพร่กระจายกลับ (Back Propagation)

หลักการของการแพร่กระจายกลับใช้สำหรับโครงสร้างเน็ตเวิร์คแบบหลายเลเยอร์ โดยการอาศัยประสบการณ์ที่ได้จากความรู้ของการแปลงสัญญาณระหว่างอินพุตและเอาต์พุตภายในเน็ตเวิร์คแบบหลายเลเยอร์ด้วยรูปแบบการเรียนรู้แบบมีครูสอน เหมือนกรณีกฎการเรียนรู้ของเซลล์ที่ศึกษามาก่อนหน้านี้ ขั้นตอนการดำเนินการของการแพร่กระจายกลับประกอบด้วย 2 ส่วนหลัก คือ ส่วนการป้อนไปหน้า อินพุตแพทเทิร์นจะถูกคำนวณผ่านเน็ตเวิร์คและสร้างเอาต์พุตที่แน่นอน ในส่วนการป้อนกลับนั้นค่าความแตกต่างระหว่างค่าเอาต์พุตจริงที่ต้องการ d_{pk} กับค่าเอาต์พุตที่ได้จากการคำนวณในเลเยอร์สุดท้าย o_{pk} จะสร้างสัญญาณค่าผิดพลาดของแพทเทิร์น E_p และจะถูกแพร่กระจาย (Propagate) แบบป้อนกลับผ่านเน็ตเวิร์คเดิมเพื่อปรับค่าถ่วงน้ำหนักของระบบ การทำงานจะเริ่มกระทำซ้ำในลักษณะเดิมอีกครั้ง จนกระทั่งค่าผิดพลาดของระบบโดยรวม E น้อยกว่าค่าผิดพลาดของระบบที่ยอมรับได้ E_a การเรียนรู้ของระบบจึงจะถือว่าสมบูรณ์

$$E_p = \frac{1}{2} \sum_k (d_{pk} - o_{pk})^2 \quad (2.38)$$

$$E = \frac{1}{P} \sum_p E_p \quad (2.39)$$

เพื่อให้ค่าผิดพลาดของระบบลดลงต่ำสุด E จะต้องให้ผลต่างกำลังสองระหว่างค่าเอาต์พุตที่ได้จากการคำนวณ o_{pk} และค่าเอาต์พุตจริงที่ต้องการมีค่าเข้าใกล้ศูนย์แต่เนื่องจากค่า d_{pk} ไม่มีการเปลี่ยนแปลง ดังนั้นต้องพยายามทำให้ค่า o_{pk} มีค่าเข้าใกล้ d_{pk} มากที่สุด โดยการหาอนุพันธ์อันดับหนึ่งของค่าผิดพลาดสำหรับแพทเทิร์นกับค่าถ่วงน้ำหนักที่เปลี่ยนไปในลักษณะเชิงเส้น

$$\Delta_p w_{kj} \propto -\frac{\partial E_p}{\partial w_{kj}} \quad (2.40)$$

เมื่อ

$$w_{kj}(t+1) = w_{kj}(t) + \Delta_p w_{kj} \quad (2.41)$$

โดย $w_{kj}(t+1)$ คือค่าตัวเลขน้ำหนักระหว่างเลเยอร์ k ไปเลเยอร์ j หลังจากปรับค่าแล้ว
 $w_{kj}(t)$ คือค่าตัวเลขน้ำหนักระหว่างเลเยอร์ k ไปเลเยอร์ ก่อนทำการปรับค่า
 $\Delta_p w_{kj}$ คือค่าตัวเลขน้ำหนักที่เปลี่ยนไปได้จากการคำนวณ
 t คือเวลาขณะใดๆ ($t+1$) เป็นเวลาถัดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการเมื่อความชันของค่าความผิดพลาด ($\partial E_p / \partial w_{kj}$) ลดลง จะมีผลทำให้ค่าถ่วงน้ำหนักที่เปลี่ยนไปลดลงด้วย วิธีการนี้จะกระทำซ้ำไปซ้ำมาจนได้ค่าที่ต้องการ จากสมการที่ (2.40) สามารถทำเป็นสมการเชิงเส้นได้ดังนี้

$$\Delta_p w_{kj} = -\eta \frac{\partial E_p}{\partial w_{kj}} \quad (2.42)$$

โดย η เป็นค่าของอัตราการเรียนรู้ (Learning Rate) ซึ่งมีค่าอยู่ระหว่าง 0.01 ถึง 1.00 และจากสมการที่ (2.42) สามารถแสดงการหาอนุพันธ์อันดับหนึ่งของ E_p เทียบกับ w_{kj} ดังนี้

$$\frac{\partial E_p}{\partial w_{kj}} = f'_k(\text{net}_{pk})(d_{pk} - o_{pk})o_{pj} \quad (2.43)$$

กำหนดให้

$$\delta_{pk} = f'_k(\text{net}_{pk})(d_{pk} - o_{pk}) \quad (2.44)$$

จะได้ว่า

$$\Delta_p w_{kj} = \eta \delta_{pk} o_{pj} \quad (2.45)$$

หากพิจารณาเพื่อหาค่าเปลี่ยนแปลงของตัวเลขน้ำหนัก $\Delta_p w_{kj}$ ระหว่างฮิดเคินเลเยอร์กับอินพุทเลเยอร์ในลักษณะเดียวกันจะได้ว่า

กำหนดให้

$$\delta_{pj} = f'_j(\text{net}_{pj}) \sum_k \delta_{pk} w_{kj} \quad (2.46)$$

จะได้ว่า

$$\Delta_p w_{ji} = \eta \delta_{pj} o_{pi} \quad (2.47)$$

จากสมการที่ (2.45) และ (2.47) จะเห็นว่าค่าถ่วงน้ำหนักจะเปลี่ยนไปตามสัญญาณค่าผิดพลาด (Error signal) ของ δ_{pk} และ δ_{pj} ตามลำดับ และเนื่องจาก $f'(\text{net})$ จะมีค่าเปลี่ยนแปลงไปตามสมการแอกติเวชัน หากเลือกใช้ซิกมอยด์ฟังก์ชัน โดยกำหนดให้ β เท่ากับ 1.0 จะได้ว่า

$$\text{net}_{pk} = \sum_j x_j w_{kj} + \theta_k \quad (2.48)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$o_{pk} = f_k(\text{net}_{pk}) = \frac{1}{1 + e^{-(\text{net}_{pk})}} \quad (2.49)$$

จะได้ว่า

$$f'_k(\text{net}_{pk}) = o_{pk}(1 - o_{pk}) \quad (2.50)$$

$$\text{net}_{pj} = \sum_i x_i w_{ji} + \theta_j \quad (2.51)$$

$$o_{pj} = f_j(\text{net}_{pj}) = \frac{1}{1 + e^{-(\text{net}_{pj})}} \quad (2.52)$$

จะได้ว่า

$$f'_j(\text{net}_{pj}) = o_{pj}(1 - o_{pj}) \quad (2.53)$$

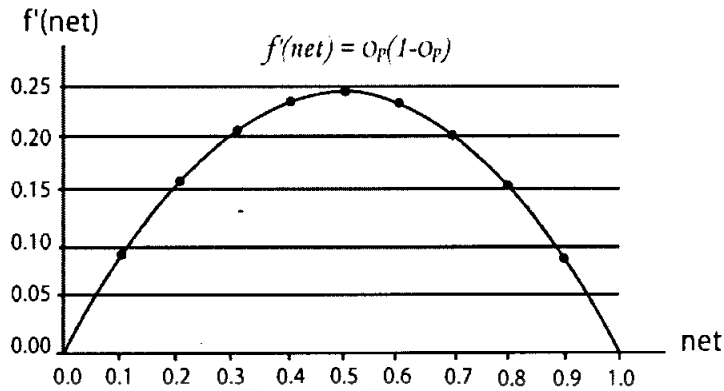
2.3.6 ปัจจัยที่ทำให้ประสิทธิภาพการสอนโครงข่ายด้วยวิธี Back Propagation เพิ่มขึ้นมีดังนี้

2.3.6.1 การกำหนดค่าเริ่มต้นให้กับเมตริกซ์ถ่วงน้ำหนัก

ก่อนจะทำการสอนโครงข่ายเพอร์เซพตรอนหลายเลเยอร์ (Multilayer Perceptron) โดยวิธี Error Back Propagation จำเป็นต้องกำหนดค่าเริ่มต้นให้กับเมตริกซ์ถ่วงน้ำหนักที่เชื่อมโยงระหว่างชั้นทุกชั้น โดยค่านี้จะเป็นเลขจำนวนจริงที่มีค่าน้อยๆ ที่ได้จากการสุ่ม ค่าเริ่มต้นของเมตริกซ์เชื่อมโยงจะมีผลต่อเวลาที่ใช้ในการสอนและอาจส่งผลถึงค่าถ่วงน้ำหนักหลังจากที่ได้สอนโครงข่ายไปแล้วว่ามีศักยภาพมากน้อยเพียงใดในการแปลงข้อมูลอินพุตไปสู่เอาต์พุต

2.3.6.2 ความเหมาะสมของการกำหนดค่าเอาต์พุตในทางปฏิบัติ

จากสมการที่ (2.45) และ (2.47) สังเกตว่าการปรับค่าถ่วงน้ำหนักจะสัมพันธ์กับค่า δ_{pk} และ δ_{pj} ตามลำดับ ซึ่งเกิดจากการหาอนุพันธ์ของฟังก์ชันแอกติเวชันแบบซิกมอยด์ในสมการที่ (2.50) และ (2.53) ดังนั้นจะเห็นว่า $o_p(1 - o_p)$ จะมีค่าสูงสุดเมื่อค่า o_p เท่ากับ 0.5 ค่าถ่วงน้ำหนักจะเปลี่ยนแปลงมากที่สุด และ $o_p(1 - o_p)$ จะมีค่าต่ำสุดเมื่อ o_p เท่ากับ 0 หรือ 1 (เมื่อ $0 < o_p < 1$) ดังแสดงในรูปที่ 2.30



รูปที่ 2.30 การหาอนุพันธ์อันดับหนึ่งของฟังก์ชันแอกติเวชัน

จากลักษณะของฟังก์ชันแอกติเวชันแบบซิกมอยด์จะเห็นว่าระบบไม่สามารถเข้าถึงค่าจริงสูงสุดและต่ำสุดของค่า 1 และ 0 ได้โดยปราศจากค่าถ่วงน้ำหนักที่ใหญ่ไม่จำกัดขอบเขต (Infinity) ดังนั้นในทางปฏิบัติหากกำหนดค่าผิดพลาดต่ำสุดที่ยอมรับได้ของระบบเป็นศูนย์ ค่าเอาต์พุตจริงที่ต้องการอาจใช้ตามแบบอย่างค่า 0.1 และ 0.9 แทนได้ แม้ว่างานที่เราทำจะใช้ค่าไบนารี 0, 1 ก็ตาม

2.3.6.3 อัตราการเรียนรู้ (Learning Rate)

อัลกอริทึมการเรียนรู้ของการแพร่กระจายกลับจะเปลี่ยนแปลงค่าถ่วงน้ำหนักตามสัดส่วนของ $\partial E_p / \partial w_p$ ในลักษณะ Gradient Descent ซึ่งทำงานแบบไม่จำกัดรอบในขณะที่เรียนรู้ ค่าคงที่ของอัตราการเรียนรู้ (η) หากมีค่ามากจะเปลี่ยนแปลงค่าถ่วงน้ำหนักได้เร็ว สำหรับในทางปฏิบัติเสนอให้เลือกอัตราการเรียนรู้ที่มากเท่าที่เป็นไปได้ โดยปราศจากการนำไปสู่การแกว่ง (Oscillate) บางครั้งอาจทดลองโดยเปลี่ยนแปลงไปเรื่อยๆ ไม่เช่นนั้นอาจจำเป็นต้องเพิ่มเทอมของโมเมนตัม (Momentum) ในสมการเคลด้าของการแพร่กระจายกลับเพื่อแก้ปัญหานี้

2.3.6.4 การใช้ค่าโมเมนตัม

วิธีสำหรับพัฒนาเวลาในการฝึกสอนของการแพร่กระจายกลับอัลกอริทึมที่เรียกว่า โมเมนตัม ด้วยวิธีการเพิ่มตัวแปรการปรับค่าตัวเลขน้ำหนักซึ่งสัมพันธ์กับค่าตัวเลขน้ำหนักที่ใช้ก่อนหน้า

$$\Delta_p w(t) = \eta \delta_p o_p + \alpha \Delta_p w(t-1) \quad (2.54)$$

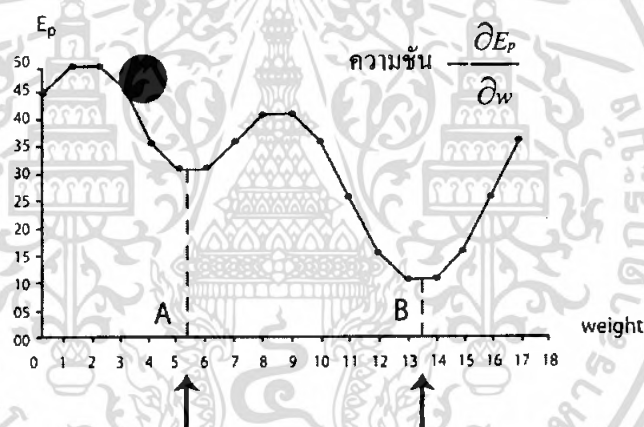
$$w(t+1) = w(t) + \Delta_p w(t) \quad (2.55)$$

ซึ่ง α เป็นสัมประสิทธิ์โมเมนตัม (Momentum coefficient) โดยทั่วไปมีค่าอยู่ประมาณ 0.9

การใช้วิธีของโมเมนต์จะทำให้เน็ตเวิร์คของการฝึกสอนเข้าใกล้จุดต่ำสุดของพื้นผิวค่าผิดพลาด (Error surface) เร็วยิ่งขึ้น วิธีนี้จะทำงานได้ดีกับบางกรณี เนื่องจากการหาค่าที่เหมาะสมของโมเมนต์จะขึ้นอยู่กับแต่ละกรณี

2.3.6.5 ตำแหน่งต่ำสุดแบบโลคอล (Local Minima) ของการแพร่กระจายกลับ

แม้ว่าการประยุกต์ใช้การแพร่กระจายกลับอัลกอริทึมจะประสบความสำเร็จมากมาย แต่ก็ยังไม่ใช่วิธีแก้ได้ทุกปัญหา สำหรับปัญหาที่ซับซ้อนบางอย่างอาจใช้เวลาเป็นวันหรือเป็นอาทิตย์เพื่อให้เน็ตเวิร์คเรียนรู้และการสอนนานๆ อาจจะมีหลุมเนื่องจากสาเหตุของตำแหน่ง Local Minima เพราะการแพร่กระจายกลับใช้หลักการของ Gradient Descent หมายความว่าความชันบนพื้นผิวค่าผิดพลาดจะเคลื่อนที่ในทิศทางที่ลดลง การปรับค่าถ่วงน้ำหนักจะเข้าหาจุดต่ำสุดแต่เนื่องจากพื้นผิวค่าผิดพลาดของเน็ตเวิร์คที่ซับซ้อนมีการบิดเบี้ยว มีเนินเขา หุบเขาในที่ว่างของหลายมิติ เมื่อมีหุบเขาต่างๆ ที่อยู่ใกล้กว่า เน็ตเวิร์คจึงมีโอกาสติดกับดักของตำแหน่งต่ำสุดท้องถิ่นดังแสดงในรูปที่ 2.31



รูปที่ 2.31 ทิศทางการเคลื่อนที่ของการปรับค่าถ่วงน้ำหนักกับปัญหาของ Local Minima จุด A และ B

วิธีทางสถิติจะช่วยหลีกเลี่ยงกับดักนี้ Wasserman (1988) ได้เสนอวิธีการทางสถิติของ Cauchy Machine กับ Gradient Descent ของการแพร่กระจายกลับ เพื่อให้ได้ระบบที่สามารถค้นหาจุดต่ำสุดทั่วไป (Global minimum) โดยใช้เวลาในการฝึกสอนมากกว่าปกติ

2.3.6.6 ความเสียหายจากค่าสมมาตร (Symmetry breaking)

ปัญหาอีกอย่างในอัลกอริทึมการเรียนรู้ของการแพร่กระจายกลับคือ เรื่องความเสียหายจากค่าสมมาตร หากมีการกำหนดให้ค่าถ่วงน้ำหนักทุกตัวมีค่าเท่ากัน ระบบจะไม่สามารถเรียนรู้ นั่นเป็นเพราะว่าทุกๆ ฮิดเดนยูนิตเชื่อมต่อโดยตรงไปยังอินพุตยูนิตในลักษณะแพร่ขยาย จะได้รับสัญญาณค่าผิดพลาดที่ได้จากการคำนวณเป็นค่าเดียวกัน ทำให้การปรับค่าถ่วงน้ำหนักเปลี่ยนแปลงไปในทิศทางเดียวกันเสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบจะเริ่มเข้าสู่ Local Minima ดังนั้นจึงควรหลีกเลี่ยงการแก้ปัญหาทำโดยการกำหนดค่าเริ่มต้นของค่าถ่วงน้ำหนักแบบสุ่มขนาดเล็กๆ ให้แก่ระบบ

ต่อไปเราจะศึกษาการทำงานทฤษฎีการแพร่กระจายกลับอัลกอริทึมกับปัญหาของพีชคณิตบูลีนแบบเอ็กซ์คลูซีฟออร์ เพื่อให้เห็นขั้นตอนในทางปฏิบัติและวิธีการนำไปใช้งานโดยทั่วไป

2.3.7 นิเวรอลเน็ตเวิร์กกับพีชคณิตบูลีนแบบเอ็กซ์คลูซีฟออร์

จากตัวอย่างการทำงานของเน็ตเวิร์กแบบเลเยอร์เดียว ซึ่งโหนดของอินพุตยูนิตเชื่อมต่อโดยตรงกับเอาต์พุตยูนิต (เน็ตเวิร์กไม่มีฮิดเดนยูนิตหรือเลเยอร์ภายใน) ระบบสามารถเรียนรู้ชุดของข้อมูลอินพุตจำนวนน้อยๆ จากตัวอย่างได้ง่ายๆ แต่ไม่ทุกกรณี เช่นตัวอย่างปัญหาของเอ็กซ์คลูซีฟออร์

ตารางที่ 2.2 กลุ่มข้อมูลอินพุตแพทเทิร์นของพีชคณิตบูลีนแบบเอ็กซ์คลูซีฟออร์

Input Pattern		Output Pattern	Input Pattern			Output Pattern
0	0	0	0	0	1	0
0	1	1	0	1	1	1
1	0	1	1	0	1	1
1	1	0	1	1	1	0

(ก) อินพุตแพทเทิร์นแบบที่ 1

(ข) อินพุตแพทเทิร์นแบบที่ 2

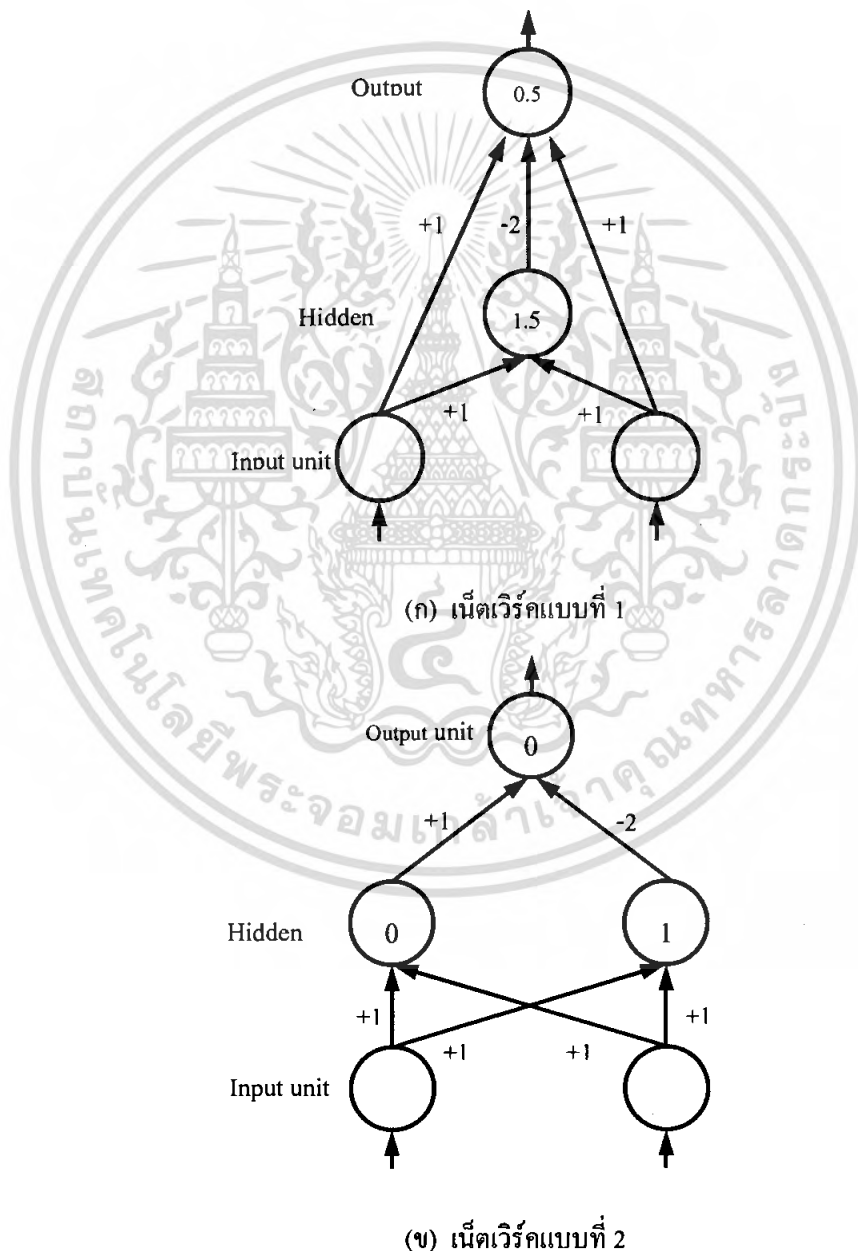
นอกจากการเพิ่มจำนวนเลเยอร์แล้ว การเพิ่มจำนวนโหนดให้กับอินพุตเลเยอร์ถือเป็นทางออกหนึ่งในการแก้ปัญหาของพีชคณิตบูลีนแบบเอ็กซ์คลูซีฟออร์ สำหรับนิเวรอลเน็ตเวิร์กแบบเลเยอร์เดียวได้ดังแสดงในตาราง (ข) แต่อาจใช้ไม่ได้กับกรณีอื่นๆ

Minsky และ Papert มีการวิเคราะห์ความสามารถในการแปลงสัญญาณภายใต้เงื่อนไขต่างๆ หากจำนวนชุดของอินพุตแพทเทิร์น (จำนวนเงื่อนไข) ที่ต้องการมีมากขึ้น โครงสร้างของเน็ตเวิร์กแบบเลเยอร์เดียวไม่สามารถแก้ปัญหาได้ และยังชี้ให้เห็นว่าการเพิ่มฮิดเดนยูนิตดังแสดงในรูปที่ 2.30 จะสามารถสนับสนุนความต้องการของการแปลงสัญญาณจากอินพุตไปยังเอาต์พุตได้ และสำหรับปัญหาของพีชคณิตบูลีนแบบเอ็กซ์คลูซีฟออร์นั้น การเพิ่มฮิดเดนยูนิตที่เพียงพอก็สามารถแก้ปัญหาของเอ็กซ์คลูซีฟออร์ได้อย่างถูกต้อง

รูปที่ 2.32 (ก) แสดงโครงสร้างนิเวรอลเน็ตเวิร์กสำหรับพีชคณิตบูลีนแบบเอ็กซ์คลูซีฟออร์ตัวเลขที่เขียนในวงกลมจะแสดงค่าเทรซโฮลด์ +1.5 สำหรับเทรซโฮลด์ของฮิดเดนยูนิตทำให้แน่ใจว่าจะให้ค่า On เท่านั้นเมื่ออินพุตยูนิตทั้งสองมีค่า On ส่วนค่า 0.5 สำหรับเอาต์พุตยูนิต ทำให้แน่ใจว่าจะให้ค่า On เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

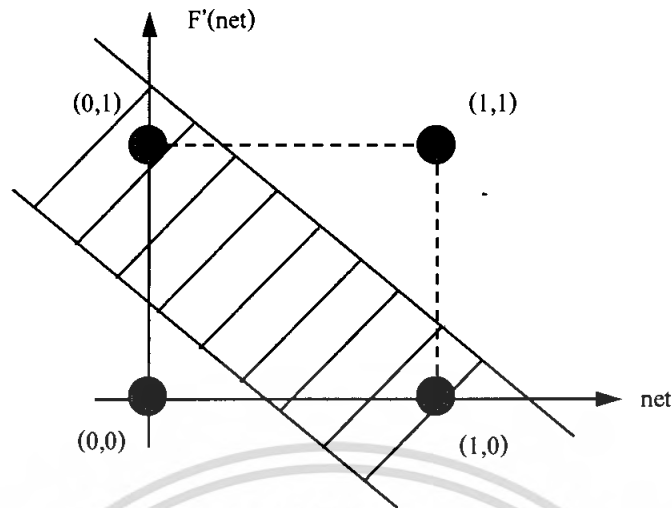
เท่านั้นเมื่อผลรวมสุทธิในโหนดมีค่ามากกว่า 0.5 ค่าถ่วงน้ำหนักของ -2.0 จากฮิดเดนยูนิตทำให้แน่ใจว่าเอาต์พุตยูนิตจะไม่กลายเป็น 0 เมื่อทั้งคู่ของอินพุตเป็น On

รูปที่ 2.32 (ข) แสดงโครงสร้างนิวรอลเน็ตเวิร์คโดยทั่วไป ซึ่งไม่มีการเชื่อมต่อโดยตรงระหว่างอินพุตและเอาต์พุตยูนิต และจะมีฮิดเดนยูนิตสองหน่วยทำหน้าที่ในการกำหนดขอบเขตการตัดสินใจ จากรูป 2.33 ฮิดเดนยูนิตแรกจะให้ค่า Off ต่อเมื่ออินพุตแพทเทิร์นเป็น (0, 0) ทำให้เอาต์พุตยูนิต เป็น Off และเมื่ออินพุตแพทเทิร์นเป็น (0, 1) และ (1, 0) ฮิดเดนยูนิตแรกจะให้ค่า On ทำให้เอาต์พุตยูนิตเป็น On ตามไปด้วย ถ้าอินพุตแพทเทิร์นเป็น (1, 1) ฮิดเดนยูนิตทั้งสองจะเป็น On แต่ว่าค่าถ่วงน้ำหนักที่มีค่ามากกว่าของฮิดเดนยูนิตที่สองมีผลให้เอาต์พุตยูนิตเป็น Off



รูปที่ 2.32 โครงสร้างนิวรอลเน็ตเวิร์คสำหรับพีชคณิตบูลีนแบบเอ็กซ์คลูซีฟออร์

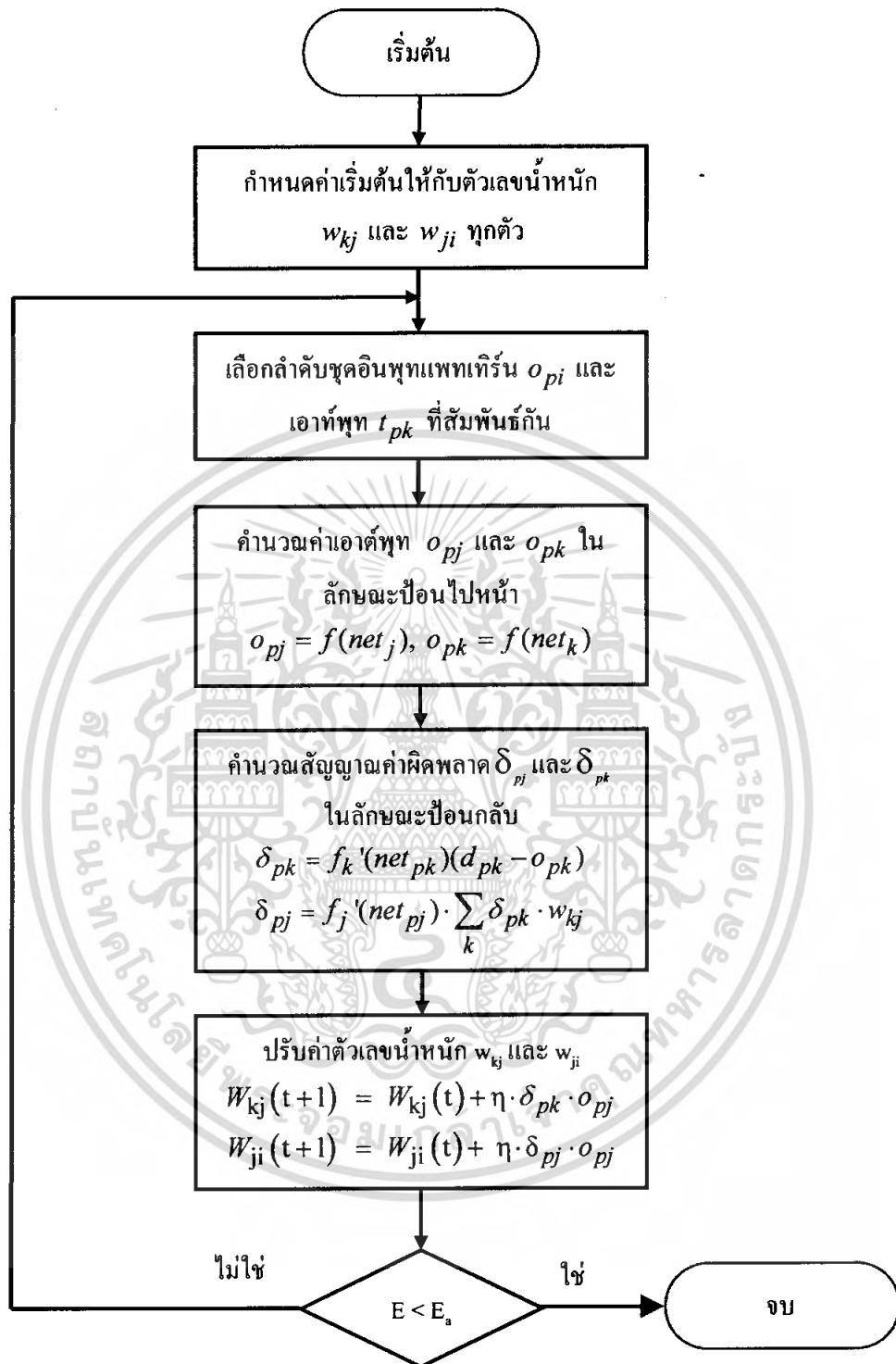
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.33 กราฟการแยกจำพวกของกลุ่มแพทเทิร์นของพีชคณิตบูลีนแบบเอ็กซ์คลูซีฟออร์
สำหรับสองฮิดเดนยูนิต

ในรูปที่ 2.33 กราฟการตัดสินใจสำหรับปัญหาของเอ็กซ์คลูซีฟออร์ เพื่อทำการแบ่งแยกจุดของแพทเทิร์นทั้งสี่จุด $(0, 0)$, $(0, 1)$, $(1, 0)$ และ $(1, 1)$ ออกเป็นสองคลาสหรือกลุ่ม จุดที่ 1 และ 4 อยู่ในกลุ่มแรกมีค่าเอาต์พุตเป็น 0 ($0 \text{ XOR } 0 = 0$ และ $1 \text{ XOR } 1 = 0$) จุดที่ 2 และ 3 อยู่ในกลุ่มสองมีค่าเอาต์พุตเป็น 1 ($0 \text{ XOR } 1 = 1$ และ $1 \text{ XOR } 0 = 1$) ฮิดเดนยูนิตหน่วยแรกจะสามารถสร้างเส้นตรงเพื่อแบ่งอาณาบริเวณ (Region) เป็นสองส่วน สำหรับทุกจุดบนข้างหนึ่งของเส้นจะให้ผลเป็น 1 สำหรับทุกจุดบนอีกข้างหนึ่งจะให้ผลเป็น 0 ตำแหน่งการหมุนของเส้นตรงถูกกำหนดโดยค่าถ่วงน้ำหนักบนอินพุตนิวรอลที่เชื่อมต่อสำหรับฮิดเดนยูนิตที่ 2 ก็ให้ผลเช่นกัน ทำให้เราสามารถแบ่งอินพุตแพทเทิร์นออกเป็นสองคลาสได้นั่นเอง

การทำงานของ การแพร่กระจายกลับอัลกอริทึมสำหรับโครงสร้างพื้นฐานของเน็ตเวิร์คแบบสองเลเยอร์ การเรียนรู้จะเริ่มการกำหนดค่าเริ่มต้นให้กับตัวแปรที่จำเป็นทุกตัวในชั้นคอนที่หนึ่ง จากนั้นเลือกลำดับอินพุต o_{pj} และเอาต์พุตแพทเทิร์น t_{pk} ที่สัมพันธ์กัน ใช้เป็นข้อมูลเริ่มต้นของระบบในชั้นคอนที่สอง คำนวณเอาต์พุตที่ได้จากการคำนวณ o_{pk} และ t_{pk} จะถูกนำมาคำนวณเพื่อหาสัญญาณค่าผิดพลาด δ_{pk} และ δ_{pj} ตามลำดับในลักษณะป้อนกลับในชั้นคอนที่สี่ แล้วนำมาคำนวณต่อไปเพื่อใช้ปรับค่าตัวเลขน้ำหนัก w_{kj} และ w_{ji} ในชั้นคอนที่ห้า หลังจากผ่านการอบรมของชุดข้อมูลอินพุตแพทเทิร์น (เมื่อ p คือลำดับของชุดแพทเทิร์น) การเรียนรู้จะหยุดต่อเมื่อค่าผิดพลาดสำหรับทุกแพทเทิร์น E มีค่าต่ำกว่าค่าผิดพลาดโดยรวมที่ยอมรับได้ E_a ในชั้นคอนที่หก ดังรูปที่ 2.34



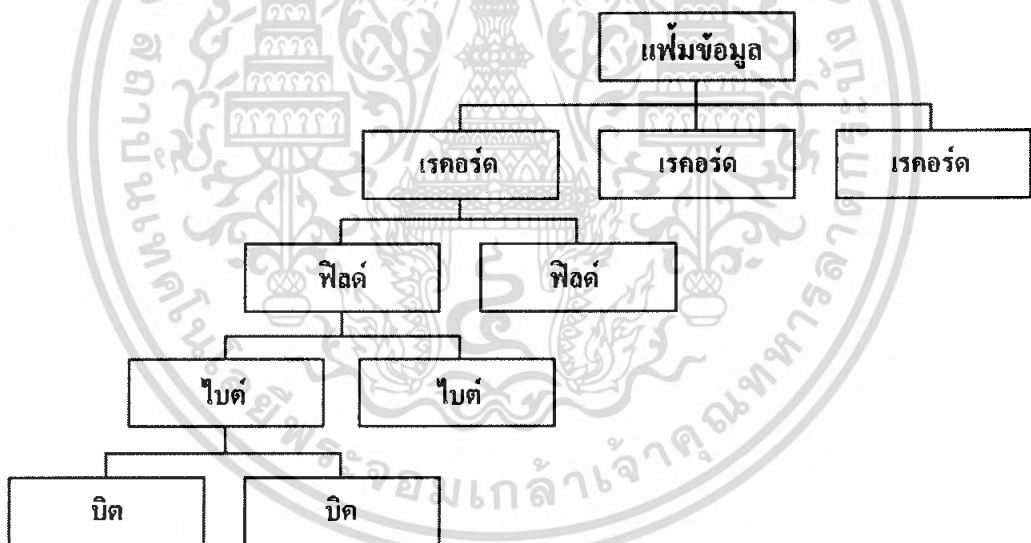
รูปที่ 2.34 แผนภูมิการทำงานของกระบวนการแพร่กระจายกลับอัลกอริทึม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ระบบฐานข้อมูล

การประมวลผลด้วยเครื่องคอมพิวเตอร์ ข้อมูลนับเป็นส่วนที่สำคัญยิ่งของการประมวลผลเพราะถ้าปราศจากข้อมูล การประมวลผลก็ไม่อาจทำได้ ข้อมูลที่ใช้ในการประมวลผลด้วยเครื่องคอมพิวเตอร์จะเป็นข้อมูลที่จัดเก็บเป็นแฟ้มข้อมูล (File) โดยแบ่งออกเป็นเรื่องตามชื่อแฟ้มข้อมูลนั้น เช่น แฟ้มข้อมูลเรื่องลูกค้า แฟ้มข้อมูลเรื่องสินค้า แฟ้มข้อมูลเรื่องการขาย แฟ้มข้อมูลเรื่องเช็คนาการ เป็นต้น ในการแบ่งเช่นนี้แต่ละแฟ้มข้อมูลก็จะประกอบด้วยข้อมูลในเรื่องเดียวกัน เช่น เมื่อหยิบแฟ้มข้อมูลลูกค้า จะมีรายละเอียดของลูกค้าทุกคน โดยทั่วไปกิจการจะมีการจัดข้อมูลในแง่ของการใช้ (File organization) โดยจัดเป็นโครงสร้างของแฟ้มข้อมูลที่ถูกรวบรวมบนอุปกรณ์เก็บข้อมูลของเครื่องคอมพิวเตอร์ เช่น การจัดเก็บข้อมูลแบบเรียงลำดับตัวอักษรชื่อ เป็นต้น เมื่อมีความต้องการรายละเอียดของลูกค้าคนใดก็จะนำแฟ้มข้อมูลลูกค้าออกมาเปิด และดึงเอารายละเอียดของลูกค้าคนนั้นออกมา ซึ่งรายละเอียดของลูกค้าแต่ละคนอาจประกอบด้วยข้อมูลเกี่ยวกับ ชื่อ ที่อยู่ เบอร์โทรศัพท์ เป็นต้น รายละเอียดของลูกค้าแต่ละคนนี้ เรียกว่า ระเบียบหรือเรคอร์ด แฟ้มข้อมูลหนึ่งๆ จะประกอบด้วยระเบียบหลายๆ ระเบียบ

ดังนั้นโครงสร้างของแฟ้มข้อมูลจะมีลักษณะดังนี้



รูปที่ 2.35 โครงสร้างของแฟ้มข้อมูล

ในระบบฐานข้อมูลจะมีการใช้ภาษา SQL เป็นภาษามาตรฐานในการจัดการฐานข้อมูล ซึ่ง SQL ย่อมาจาก Structured Query Language หรือภาษาในการสอบถามข้อมูล เป็นภาษาทางด้านฐานข้อมูล ที่สามารถสร้างและปฏิบัติการกับฐานข้อมูลแบบสัมพันธ์ (relational database) โดยเฉพาะ และเป็นภาษาที่มีลักษณะคล้ายกับภาษาอังกฤษ ภาษา SQL ถูกพัฒนาขึ้นจากแนวคิดของ relational calculus และ relational algebra เป็นหลัก ภาษา SQL เริ่มพัฒนาครั้งแรกโดย almaden research center ของบริษัท IBM โดยมีชื่อเริ่มแรกว่า “ซีเควล” (Sequel) ต่อมาได้เปลี่ยนชื่อเป็น “เอสคิวเอล” (SQL) หลังจากนั้นภาษา SQL ได้ถูกนำมาพัฒนาโดยผู้ผลิตซอฟต์แวร์ด้านระบบจัดการฐานข้อมูลเชิงสัมพันธ์ซึ่งมีเป็นจำนวนมาก อาทิเช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ใช้ได้เห็นเว็บไซต์ประกอบเนื้อหาการเรียนการสอนนี้แล้ว กรุณาอย่าลืมที่จะแจ้งให้เราทราบด้วย เพื่อที่เราจะได้ปรับปรุงเนื้อหาให้ดียิ่งขึ้นต่อไป

Oracle, Microsoft SQL Server, MySQL, PostgreSQL ซอฟต์แวร์เหล่านี้เป็นที่รู้จักและมีการใช้งานอย่างแพร่หลาย

2.4.1 ประเภทของคำสั่งของภาษา SQL

ภาษา SQL เป็นภาษาที่ใช้งานได้ตั้งแต่ระดับเครื่องคอมพิวเตอร์ส่วนบุคคลพีซีไปจนถึงระดับเมนเฟรม ประเภทของคำสั่งในภาษา SQL (The subdivision of sql) แบ่งออกเป็น 3 ประเภท คือ

1. ภาษาสำหรับการนิยามข้อมูล (Data Definition Language: DDL) ประกอบด้วยคำสั่งที่ใช้ในการกำหนดโครงสร้างข้อมูลว่ามีคอลัมน์อะไร แต่ละคอลัมน์เก็บข้อมูลประเภทใด รวมถึงการเพิ่มคอลัมน์ การกำหนดคีย์ การกำหนดคิวหรือตารางเสมือนของผู้ใช้ เป็นต้น

2. ภาษาสำหรับการจัดการข้อมูล (Data Manipulation Language: DML) ประกอบด้วยคำสั่งที่ใช้ในการเรียกใช้ข้อมูล การเปลี่ยนแปลงข้อมูล การเพิ่มหรือลบข้อมูล เป็นต้น

3. ภาษาควบคุม (Data Control Language: DCL) ประกอบด้วยคำสั่งที่ใช้ในการควบคุม การเกิดภาวะพร้อมกัน หรือการป้องกันการเกิดเหตุการณ์ที่ผู้ใช้หลายคนเรียกใช้ข้อมูลพร้อมกัน และคำสั่งที่เกี่ยวข้องกับการควบคุมความปลอดภัยของข้อมูลด้วยการกำหนดสิทธิ์ของผู้ใช้ที่แตกต่างกัน

2.4.2 ชนิดของข้อมูลที่ใช้ในภาษา SQL

ในภาษา SQL การบรรจุข้อมูลลงในคอลัมน์ต่าง ๆ ของตารางจะต้องกำหนดชนิดของข้อมูล (data type) ให้แต่ละคอลัมน์ ชนิดของข้อมูลนี้จะแสดงชนิดของค่าที่อยู่ในคอลัมน์ ค่าทุกค่าในคอลัมน์ที่กำหนดจะต้องเป็นชนิดเดียวกัน เช่น ในตารางลูกค้าคอลัมน์ที่เป็นรายชื่อลูกค้า จะต้องเป็นตัวหนังสือ ในขณะที่คอลัมน์จำนวนเงินที่ลูกค้าซื้อสินค้าเป็นตัวเลข

ชนิดของข้อมูลของแต่ละคอลัมน์จะขึ้นกับลักษณะของข้อมูลแต่ละคอลัมน์ ซึ่งแบ่งได้ดังนี้ชนิดข้อมูลพื้นฐานในภาษา SQL ดังนี้

1. ตัวหนังสือ (character) ในภาษา SQL จะใช้

ตัวหนังสือแบบความยาวคงที่ (fixed-length character) จะใช้ char (n) หรือ character (n) แทนประเภทของข้อมูลที่เป็นตัวหนังสือใดๆ ที่มีความยาวของข้อมูลคงที่โดยมีความยาว n ตัวหนังสือประเภทนี้จะมีการจองเนื้อที่ตามความยาวที่คงที่ตามที่กำหนดไว้ ชนิดของข้อมูลประเภทนี้จะเก็บความยาวของข้อมูลได้มากที่สุดได้ 255 ตัวอักษร

ตัวหนังสือแบบความยาวไม่คงที่ (variable-length character) จะใช้ varchar (n) แทนประเภทของข้อมูลที่เป็นตัวหนังสือใดๆ ที่มีความยาวของข้อมูลไม่คงที่ โดยมีความยาว n ตัวหนังสือประเภทนี้จะมีการจองเนื้อที่ตามความยาวของข้อมูล ชนิดของข้อมูลประเภทนี้จะเก็บความยาวของข้อมูลได้มากที่สุดได้ 4000 ตัวอักษร

2. จำนวนเลข (numeric)

จำนวนเลขที่มีจุดทศนิยม (decimal) ในภาษา SQL จะใช้ dec (m,n) หรือ decimal(m,n) แทนประเภทข้อมูลที่เป็นจำนวนเลขที่มีจุดทศนิยมโดย m คือจำนวนตัวเลขทั้งหมด (รวมจุดทศนิยม) และ n คือจำนวนตัวเลขหลังจุดทศนิยม

จำนวนเลขที่ไม่มีจุดทศนิยมในภาษา SQL จะใช้ int หรือ integer เป็นเลขจำนวนเต็มบวกหรือลบ ขนาดปกติ เป็นตัวเลข 10 หลัก ที่มีค่าตั้งแต่ -2,147,483,648 ถึง 2,147,483,647 และในภาษา SQL จะใช้ smallint แทนประเภทข้อมูลที่เป็นเลขจำนวนเต็มบวกหรือลบขนาดเล็ก เป็นตัวเลข 5 หลัก ที่มีค่าตั้งแต่ -32,768 ถึง 32,767

เลขจำนวนจริง ในภาษา SQL อาจใช้ number (n) แทนจำนวนเลขที่ไม่มีจุดทศนิยมและจำนวนเลขที่มีจุดทศนิยม

3. ข้อมูลในลักษณะอื่นๆ

วันที่และเวลา (Date/Time) เป็นชนิดวันที่หรือเวลาในภาษา SQL จะใช้ date เป็นข้อมูลวันที่ซึ่งจะมีหลายรูปแบบให้เลือกใช้ เช่น yyyy-mm-dd (1999-10-31) เป็นต้น

ในโครงการนี้จะจำลองระบบฐานข้อมูลของยานพาหนะ โดยจะมีข้อมูลของป้ายทะเบียน จังหวัด ชื่อเจ้าของยานพาหนะและจำนวนล้อ เพื่อนำมาประกอบการตัดสินใจในการคิดค่าผ่านทางสำหรับทางด่วน ซึ่งระบบฐานข้อมูลที่จำลองขึ้นประกอบไปด้วย

1. ระบบปฏิบัติการ Linux

Linux เป็นระบบปฏิบัติการเกี่ยวกับระบบเครือข่าย (Network Operating System) ซึ่งมีลักษณะเด่นคือ เป็นระบบปฏิบัติการที่ไม่เสียค่าใช้จ่าย (Free software) และอยู่ภายใต้ลิขสิทธิ์ของ GNU (GNU's Not Unix) โดยระบบปฏิบัติการ Linux เป็นระบบปฏิบัติการแบบ open source คือ มีการเปิดเผย source code สามารถนำมาทำการพัฒนาต่อได้และต้องเปิดเผย source code ที่พัฒนาขึ้นให้กับผู้อื่นด้วย ทำให้ระบบปฏิบัติการ Linux มีการพัฒนาอย่างต่อเนื่องและมีเสถียรภาพในการทำงาน

2. ซอฟต์แวร์ระบบฐานข้อมูล MySQL

MySQL เป็นซอฟต์แวร์ระบบฐานข้อมูลแบบ open source ที่นิยมใช้กันมากเกี่ยวกับระบบสารสนเทศทางอินเทอร์เน็ต เพราะเป็นซอฟต์แวร์ระบบฐานข้อมูลที่ไม่เสียค่าใช้จ่าย ซึ่งเหมาะกับการนำมาใช้เพื่อการศึกษา

3. ซอฟต์แวร์ให้บริการเว็บ Apache HTTP Server

Apache เป็นซอฟต์แวร์ให้บริการเว็บ ไม่เสียค่าใช้จ่ายและเป็น open source โดยจะนำมาช่วยในการจัดการฐานข้อมูลผ่านเว็บ เนื่องจากในการจัดการกับฐานข้อมูลด้วยภาษา SQL จะมีความยุ่งยากสำหรับผู้ที่ไม่มีความรู้เกี่ยวกับภาษา SQL

4. PHP และ phpMyAdmin

PHP (Hypertext Preprocessor) เป็นภาษาโปรแกรมมิ่งชนิดหนึ่งที่ทำางร่วมกับ HTTP Server ซึ่ง PHP จะนำมาใช้ในการติดต่อกับฐานข้อมูล จัดการกับฐานข้อมูลและมีการประมวลผลที่ฝั่งผู้ให้บริการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Server-Side Processing) ส่วน phpMyAdmin ก็ซอฟต์แวร์พัฒนาด้วย PHP ที่สามารถทำงานและจัดการกับฐานข้อมูลผ่านเว็บได้ทันทีโดยจำเป็นต้องสร้างการติดต่อกับฐานข้อมูลด้วย PHP เอง

2.5 รูปแบบของแผ่นป้ายทะเบียน

ลักษณะของแผ่นป้ายทะเบียนที่ใช้ในปัจจุบัน จะมีป้ายทะเบียนแบบเก่า ดังรูปที่ 2.36 และป้ายทะเบียนแบบใหม่ ดังตารางที่ 2.3 โดยป้ายทะเบียนแบบใหม่มีรายละเอียดดังนี้

1. ขนาด 15×34 cm
2. มี 2 บรรทัด
3. บรรทัดแรกเป็นตัวอักษรประจำหมวดที่หนึ่ง ตัวอักษรประจำหมวดที่สองและหมายเลขทะเบียน
4. บรรทัดที่สองเป็นตัวอักษรบอกชื่อจังหวัด เว้นแต่อำเภอเบตง จังหวัดยะลา ให้ใช้คำว่า เบตง
5. ลักษณะของตัวอักษรประจำหมวด หมายเลขทะเบียน ตัวอักษรบอกชื่อจังหวัด และขอบแผ่นป้าย อัดเป็นรอยคูน
6. ตัวอักษรประจำหมวด และหมายเลขทะเบียนกว้างไม่น้อยกว่า 3.5 ซม. สูงไม่น้อยกว่า 6.5 ซม.
7. ชื่อจังหวัดใช้ตัวเต็ม กว้างไม่น้อยกว่า 1.5 ซม. สูงไม่น้อยกว่า 2 ซม.
8. เว้นแต่ตัวอักษรประจำหมวดและชื่อของจังหวัดที่ใช้อักษร ข ง ช ฌ ฉ ฎ ฏ ฐ ฒ ณ ฬ ฟ ศ ษ ส ฬ ส สระ วรรณยุกต์ และหมายเลข 1 อาจมีขนาดสูงหรือกว้างมากหรือน้อยกว่าที่กำหนดไว้ได้ตามความเหมาะสม
9. ในแผ่นป้ายมีเครื่องหมายเป็นตัวอักษร ขส อยู่ภายในวงกลม อัดเป็นรอยคูนที่มุมล่างขวา ของแผ่นป้าย สามารถมองเห็นด้วยตาเปล่าได้
10. ตัวอักษรประจำหมวด ใช้ตัวอักษร ไทยมาตรฐาน โดยตัวแรกเป็นตัวบอหมวดประเภทรถ และตัวที่ 2 เป็นตัววิ่ง ซึ่งมีจำนวน 39 ตัว
11. หมายเลขทะเบียนรถใช้ตัวเลขตั้งแต่ 1 ถึง 9999 ในแต่ละหมวด โดยสรุปจะมีรูปแบบเหมือนที่ใช้ในวันที่ 1 เมษายน พ.ศ. 2547 เพียงแต่ขนาดตัวอักษรและตัวเลข จะใหญ่กว่าแบบเดิม และเปลี่ยนตัวอักษรจากตัวกราฟิก มาใช้แบบมาตรฐานทำให้อ่านง่ายชัดเจนขึ้น
12. สีพื้นของแผ่นป้าย สีตัวอักษร ตัวเลข และสีขอบบนของแผ่นป้ายจะมีรูปแบบเหมือนเดิมที่ใช้ในวันที่ 1 เมษายน พ.ศ. 2547 แต่จะแตกต่างกันตามประเภทรถดังนี้



รูปที่ 2.36 ป้ายทะเบียนแบบเก่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 รูปแบบป้ายทะเบียนแบบใหม่

รูปแบบป้ายทะเบียน	ประเภทรถ	ขนาด (เซนติเมตร)	สีพื้น	สีอักษร
คน 9999 กรุงเทพมหานคร	รถยนต์รับจ้างระหว่างจังหวัด	15x34	เหลือง	แดง
คน 9999 กรุงเทพมหานคร	รถยนต์รับจ้างบรรทุกคน โดยสารไม่เกิน 7 คน	15x34	เหลือง	ดำ
คน 9999 กรุงเทพมหานคร	รถยนต์รับจ้างสามล้อ	15x34	เหลือง	เขียว
คน 9999 กรุงเทพมหานคร	รถยนต์ 4 ล้อเล็กรับจ้าง	15x34	เหลือง	น้ำเงิน
คน 9999 กรุงเทพมหานคร	รถยนต์บริการธุรกิจ รถยนต์บริการทัศนาจร และรถยนต์บริการให้เช่า	15x34	เขียว	ดำ
คน 9999 กรุงเทพมหานคร	รถยนต์นั่งส่วนบุคคลไม่เกิน 7 คน	15x34	ขาว	ฟ้า
คน 9999 กรุงเทพมหานคร	รถยนต์บรรทุกส่วนบุคคล	15x34	ขาว	เขียว
คน 9999 กรุงเทพมหานคร	รถยนต์ 4 ล้อส่วนบุคคล	15x34	ขาว	ดำ
คน 9999 กรุงเทพมหานคร	รถยนต์ 3 ล้อส่วนบุคคล	15x34	ขาว	แดง
คน 9999 กรุงเทพมหานคร	รถพ่วง รถบดถนน รถแทรกเตอร์ และรถใช้งานเกษตรกรรม	15x34	ส้ม	ดำ
US 0007	รถยนต์ของบุคคลใน คณะผู้แทนทางการทูต	11x38.7	ขาว	ดำ
US 0007	รถยนต์ของบุคคลในหน่วยงาน พิเศษของสถานทูต คณะผู้แทนของกงสุล และองค์การระหว่างประเทศ	11x38.7	ฟ้า	ขาว

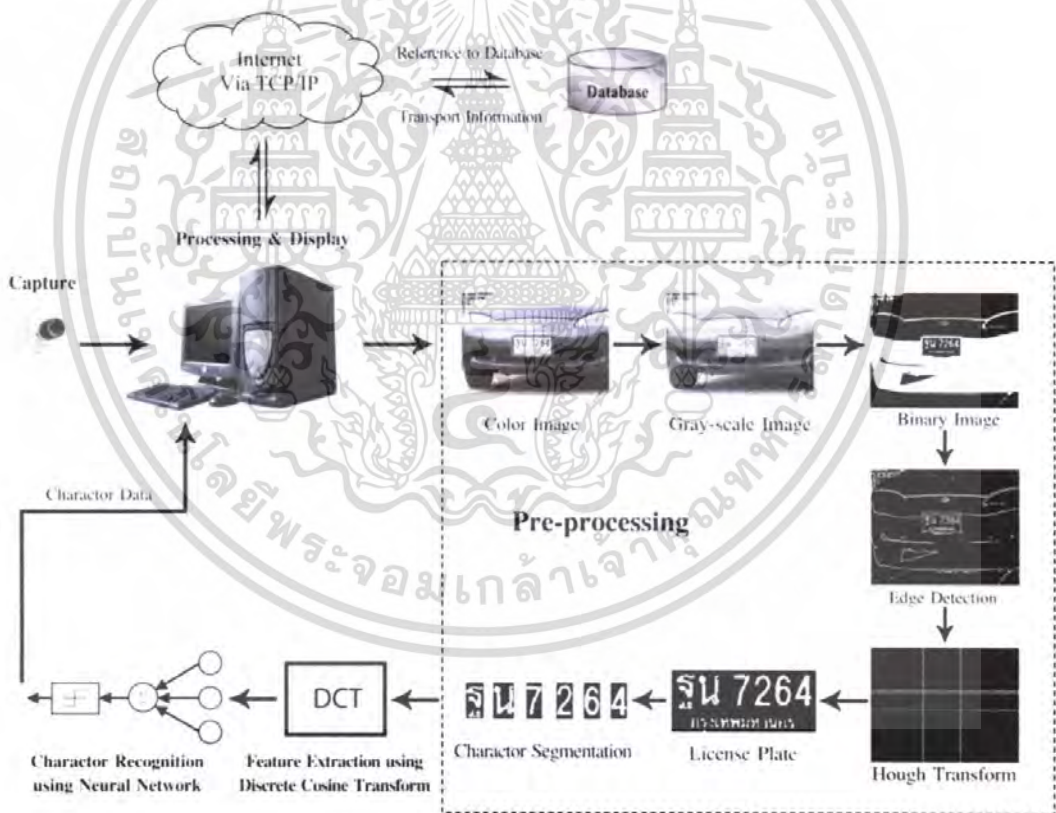
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการสร้าง

การดำเนินงานจะอธิบายกระบวนการต่างๆ ซึ่งเริ่มจากการรับข้อมูลภาพป้ายทะเบียนไปจนถึงการนำผลที่ได้จากการจดจำป้ายทะเบียนไปวิเคราะห์เพื่อหาราคาค่าผ่านทางสำหรับทางด่วน โดยมีโครงสร้างการทำงาน 4 ขั้นตอนหลักๆ ได้แก่

1. การประมวลผลภาพดิจิทัล (Digital Image Processing)
2. การแยกคุณลักษณะเฉพาะจากข้อมูลภาพ (Feature extraction)
3. การรู้จำป้ายทะเบียน (License-plate recognition)
4. การอ้างอิงป้ายทะเบียนกับระบบฐานข้อมูล (Reference to Database)



รูปที่ 3.1 การทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

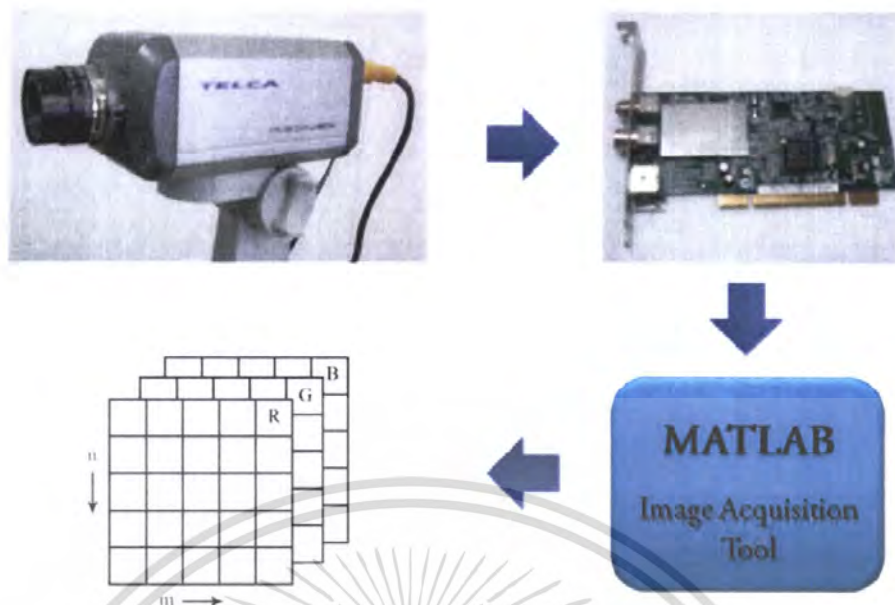
ขอบเขตของระบบในการทำงาน

1. รับข้อมูลภาพสีแบบดิจิทัลที่เป็นภาพหน้ารถพร้อมป้ายทะเบียนจากการติดต่อกับอุปกรณ์ TV Tuner และกล้อง CCD ด้วยโปรแกรม MATLAB
2. แปลงข้อมูลภาพสีเป็นภาพแบบ gray-scale 256 ระดับ
3. แปลงข้อมูลภาพแบบ gray-scale เป็นภาพแบบไบนารี 2 ระดับ (0, 1)
4. หาขอบภาพจากภาพไบนารีที่ได้และหาตำแหน่งของป้ายทะเบียนจากภาพหน้ารถทั้งหมดด้วยการใช้การแปลงฮัฟเข้ามามีช่วย แล้วคัดลอกเฉพาะภาพที่เป็นป้ายทะเบียนจากภาพหน้ารถที่เป็นภาพไบนารี
5. แยกภาพตัวอักษรทีละตัวออกจากภาพป้ายทะเบียนแล้วทำการ Normalize ข้อมูลภาพตัวอักษรให้มีขนาดเท่ากันหมดทุกภาพ
6. ทำการลดข้อมูลของภาพตัวอักษรจากการแยกคุณลักษณะเฉพาะของภาพ ด้วยการแปลงคิสคริตโคซายน์
7. นำคุณลักษณะเฉพาะของภาพตัวอักษรเข้าสู่การฝึกสอนการรู้จำป้ายทะเบียนด้วยโครงข่ายประสาทเทียม
8. นำระบบการรู้จำป้ายทะเบียนที่ทำการฝึกสอนแล้ว ไปใช้งานในการแปลงข้อมูลภาพเป็นข้อมูลตัวอักษร
9. อ้างอิงข้อมูลตัวอักษรที่ได้จากผลของการรู้จำป้ายทะเบียนกับระบบฐานข้อมูล เพื่อนำข้อมูลจำนวนล้อของยานพาหนะที่กำลังอ้างอิงอยู่มาคิดเป็นค่าผ่านทางสำหรับทางด่วน

3.1 การประมวลผลภาพดิจิทัล (Digital Image Processing)

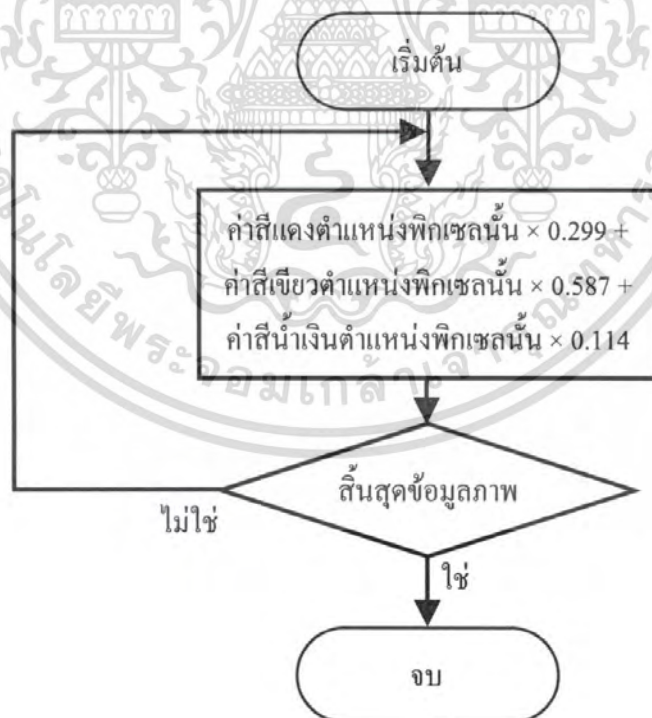
การประมวลผลภาพดิจิทัลเป็นการนำภาพดิจิทัลมาผ่านการประมวลผล เพื่อให้ได้ผลลัพธ์แบบใหม่ที่บ่งบอกถึงลักษณะและคุณสมบัติของภาพ โดยมีขั้นตอนดังนี้

1. ทำการ Capture ภาพหน้ารถพร้อมป้ายทะเบียนเป็นภาพสีดิจิทัล ด้วยการใช้โปรแกรม MATLAB ติดต่อกับอุปกรณ์ TV Tuner และกล้อง CCD ซึ่งข้อมูลภาพที่ Capture มานี้จะเป็นตัวแปรแบบอาร์เรย์ในโปรแกรม MATLAB ดังรูปที่ 3.2



รูปที่ 3.2 การรับข้อมูลภาพโดยการเชื่อมต่อกับอุปกรณ์ TV Tuner ด้วยโปรแกรม MATLAB

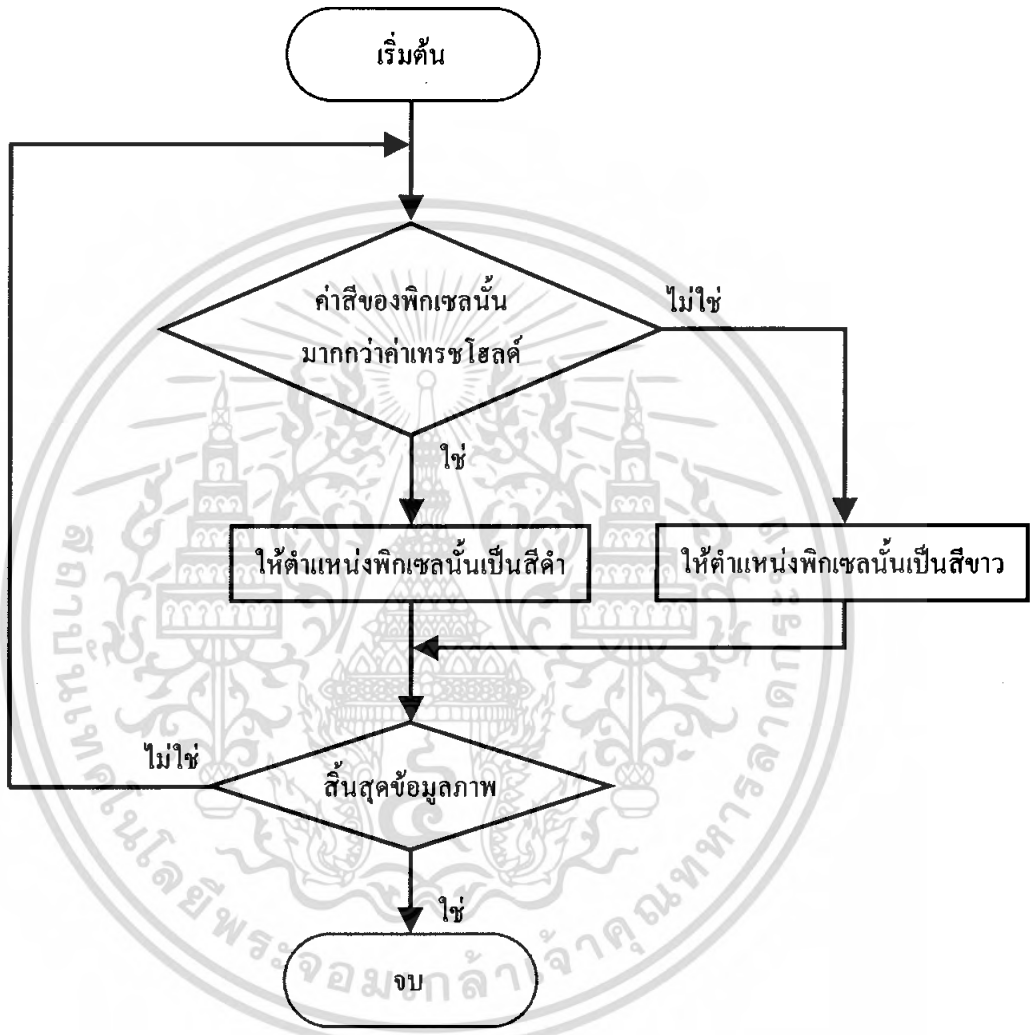
2. ทำการแปลงข้อมูลภาพสีให้เป็นภาพแบบขาว-ดำ (Gray Scale Image)



รูปที่ 3.3 แผนภูมิการทำงานของการแปลงภาพ RGB เป็นภาพขาว-ดำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

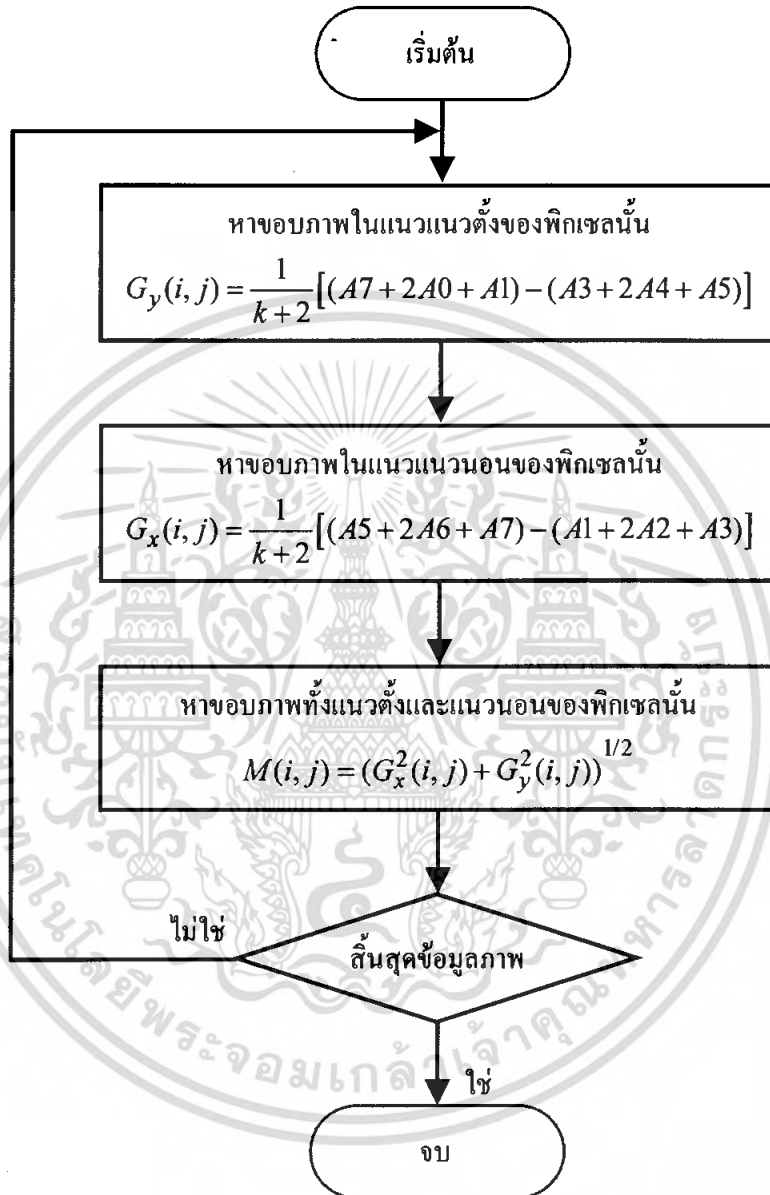
3. นำภาพแบบ Gray Scale มาทำการแปลงให้เป็นภาพแบบไบนารี (Binary Image) โดยใช้เทคนิคของการกำหนดค่าเทรชโฮลด์ เพื่อแยกระดับความเข้มของแสงออกจากกัน จากค่า 256 ระดับ ให้เหลือเพียง 2 ระดับคือ 0, 1



รูปที่ 3.4 แผนภูมิการทำงานของการแปลงภาพขาว-ดำ เป็นภาพไบนารี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

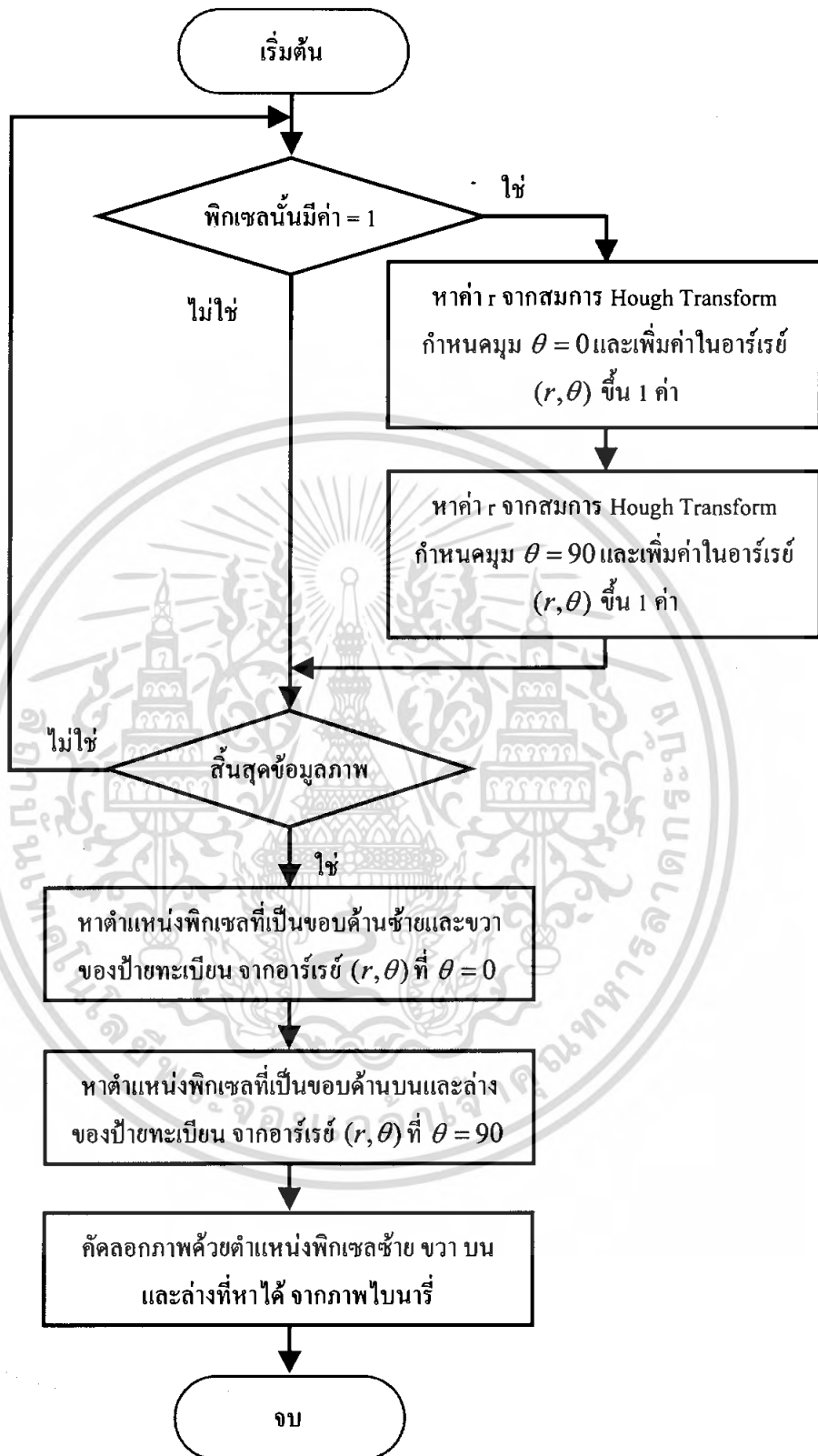
4. นำภาพแบบไบนารีมาทำการหาขอบภาพโดยใช้ Sobel Edge Detection โดยกำหนดให้สมการในการหาขอบภาพในแนวตั้งและแนวนอนมีค่า k เท่ากับ 2



รูปที่ 3.5 แผนภูมิการทำงานของการหาขอบภาพแบบ Sobel

5. เมื่อหาขอบภาพแล้ว จะนำภาพมาทำการ Threshold เพื่อทำให้ภาพมีเพียงบิต 0 และ 1 เท่านั้น จากนั้นจะทำการหาตำแหน่งของป้ายทะเบียน โดยใช้ทฤษฎีของ Hough Transform

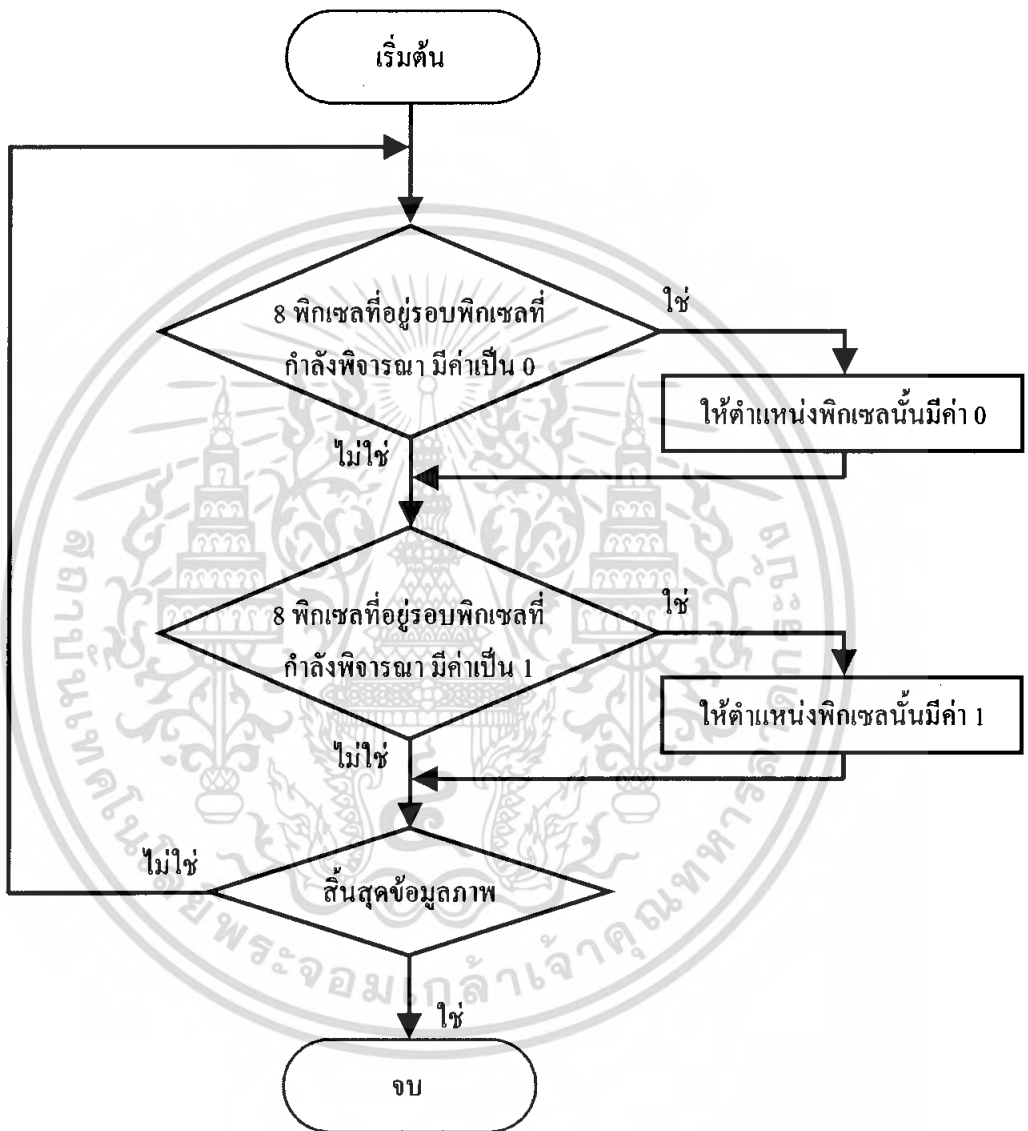
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 แผนภูมิการทำงานของการหาค่าตำแหน่งป้ายทะเบียนด้วย Hough Transform

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

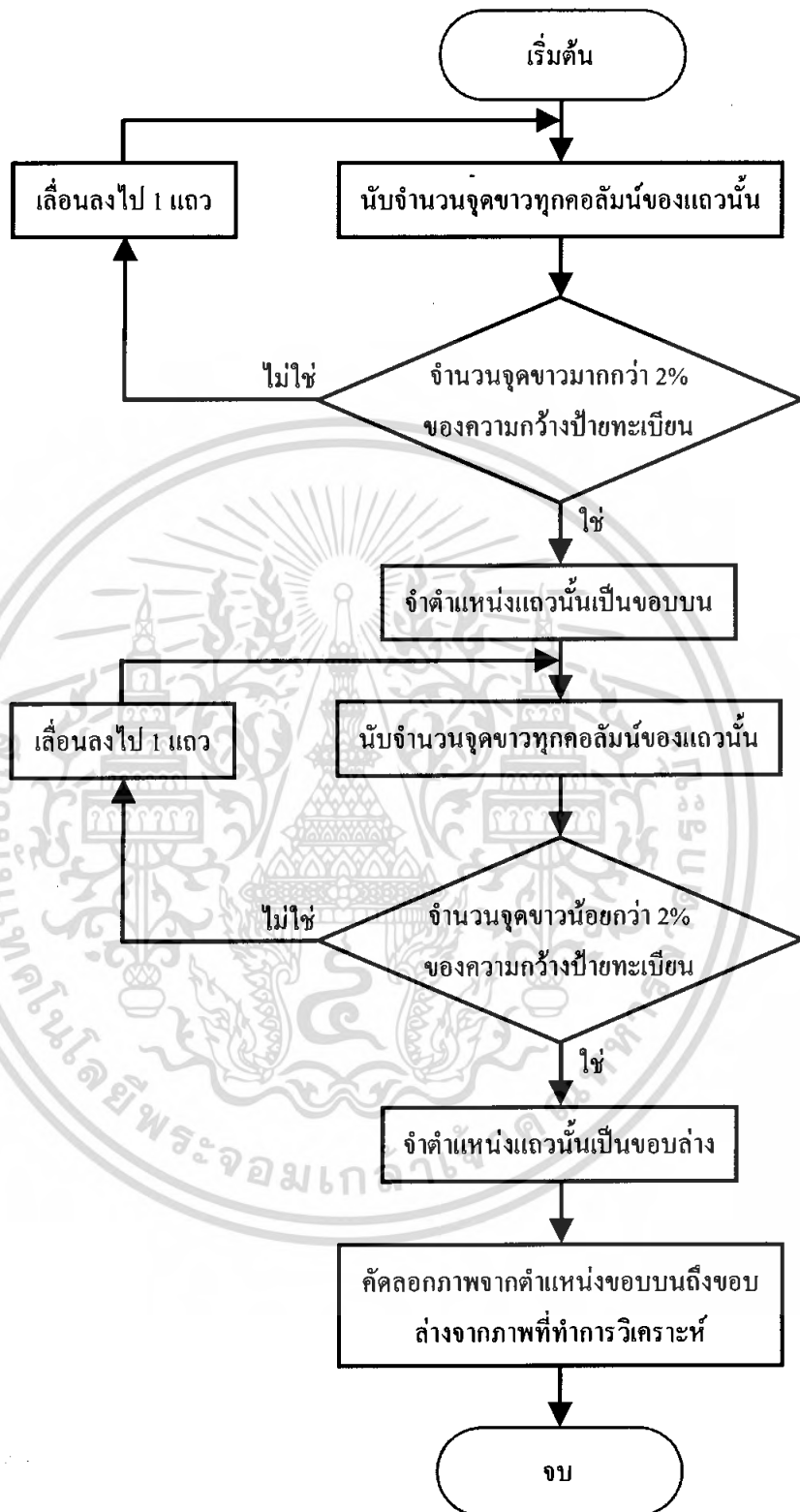
6. ทำการคัดลอกภาพเฉพาะตำแหน่งที่ได้จากขั้นตอนการหาตำแหน่งป้ายทะเบียน ซึ่งตำแหน่งที่ได้คือด้านซ้าย ขวา ล่างและบนของป้ายทะเบียน
7. นำภาพป้ายทะเบียนที่ได้ มาปรับแต่งภาพ (Image Enhancement) เพื่อให้ภาพเกิดความสมบูรณ์ขึ้น ซึ่งเป็นการลดสิ่งรบกวน (Noise) ของภาพลงด้วย



รูปที่ 3.7 แผนภูมิการทำงานของการทำ Image enhancement

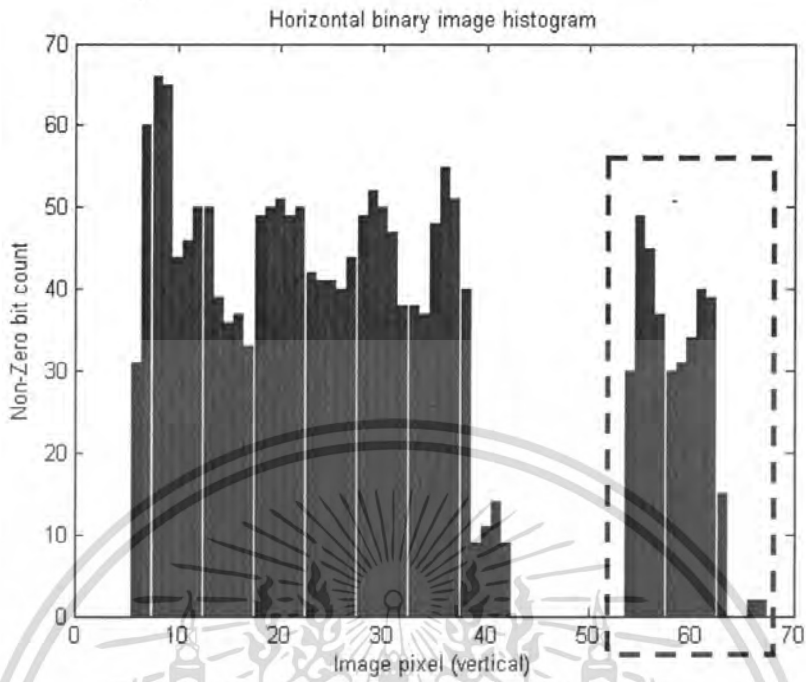
8. นำภาพที่ทำการปรับแต่ง มาตัดส่วนที่ไม่ได้ใช้ในวิเคราะห์ออก นั่นก็คือ ส่วนที่เป็นจังหวัด โดยใช้วิธีการวิเคราะห์ฮิสโตแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

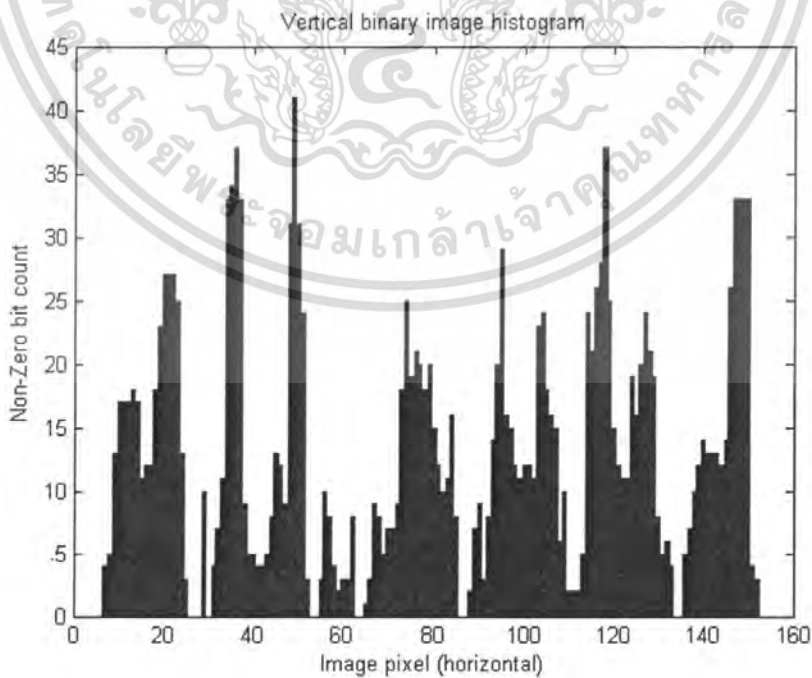


รูปที่ 3.8 แผนภูมิการทำงานของการตัดส่วนที่เป็นจังหวัดออกจากป้ายทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

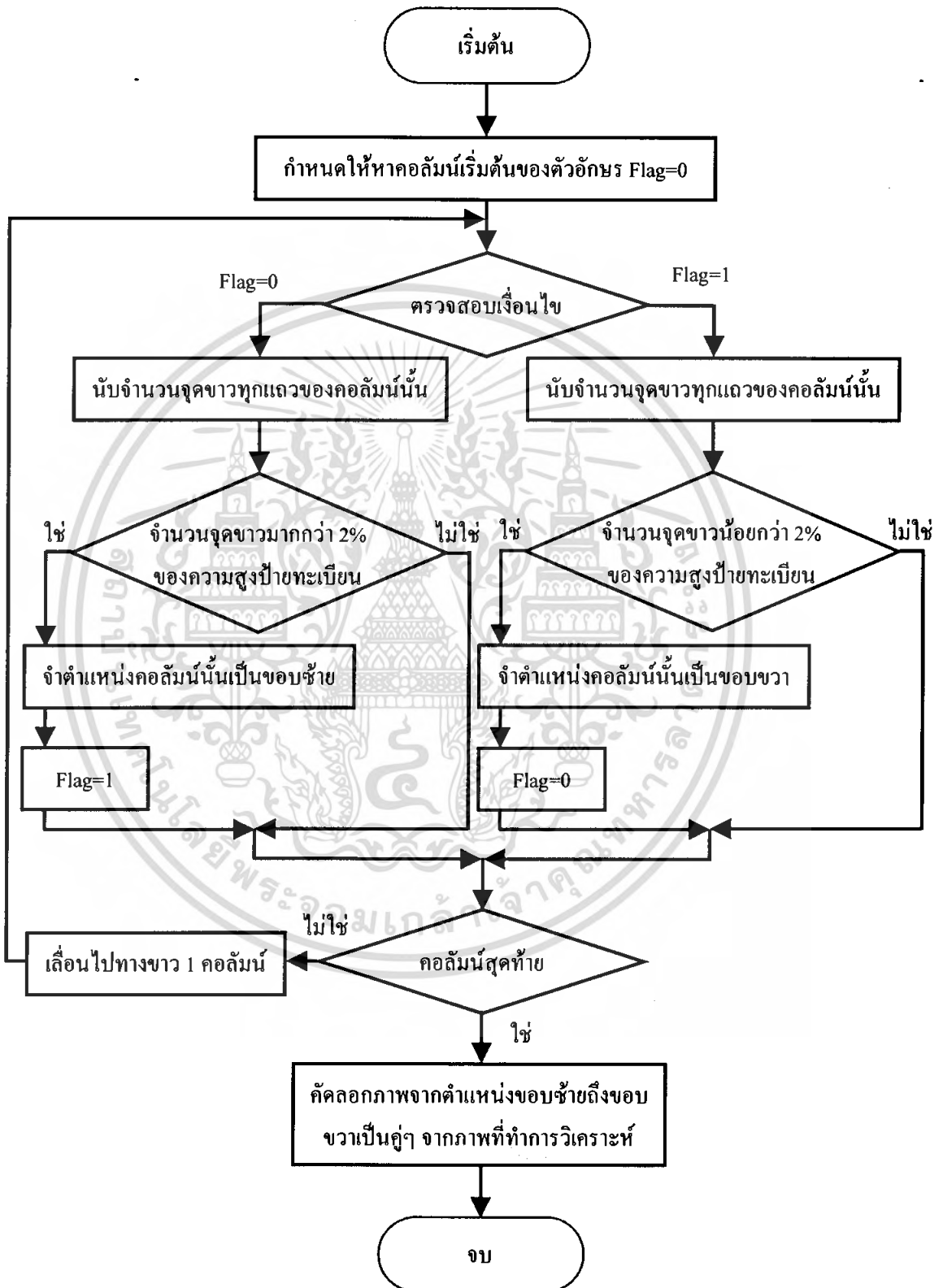


รูปที่ 3.9 ฮิสโตแกรมก่อนการตัดจังหวัดของป้ายทะเบียนออก โดยการสแกนภาพในแนวอน ซึ่งส่วนที่เป็นจังหวัดคือส่วนที่ล้อมรอบด้วยเส้นประ



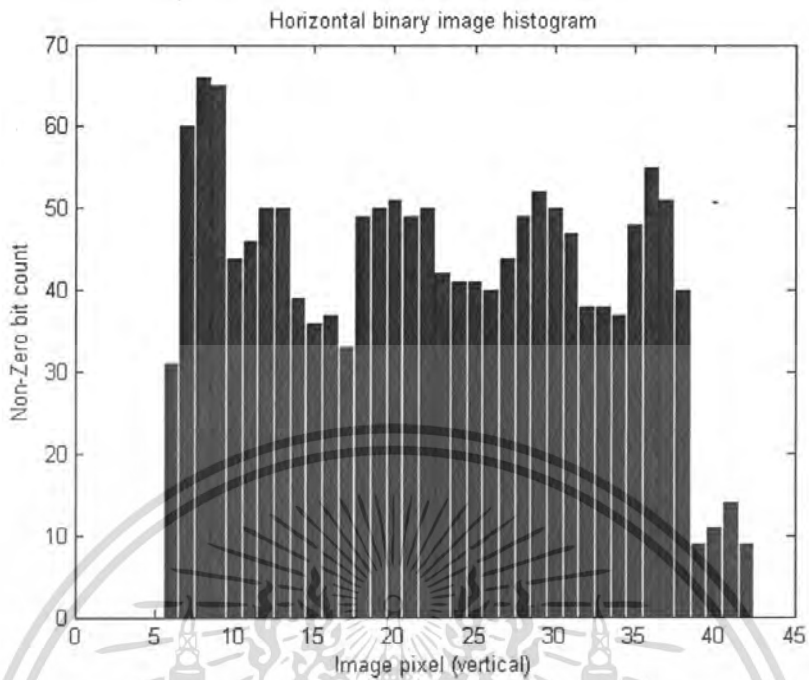
รูปที่ 3.10 ฮิสโตแกรมก่อนการตัดจังหวัดของป้ายทะเบียนออก โดยการสแกนภาพในแนวตั้ง เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปเผยแพร่โดยไม่หวังกำไรใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. นำภาพที่ตัดส่วนของจังหวัดแล้ว มาตัดตัวอักษรและตัวเลขแยกจากกัน ให้ได้เป็นตัวอักษรแต่ละตัว โดยใช้วิธีฮิสโตแกรม

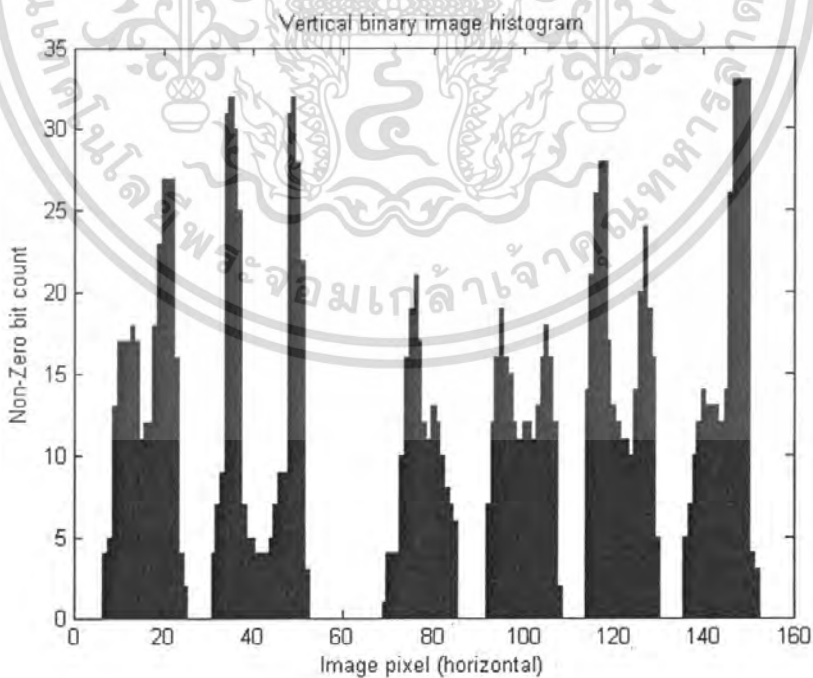


รูปที่ 3.11 แผนภูมิการทำงานของการแยกส่วนตัวอักษรออกจากกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 ฮิสโตแกรมหลังการตัดจังหวคของป้ายทะเบียนออก โดยการสแกนภาพในแนวนอน



รูปที่ 3.13 ฮิสโตแกรมหลังการตัดจังหวคของป้ายทะเบียนออก โดยการสแกนภาพในแนวตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. หลังจากตัดตัวอักษรและตัวเลขจากป้ายทะเบียนแล้ว ตัวอักษรและตัวเลขจะมีขนาดภาพที่ไม่เท่ากัน ดังนั้นจึงทำการปรับขนาดของภาพ (Image resizing) ให้มีขนาดเท่ากันทั้งหมด ในการปรับขนาดของภาพนั้น จะทำการปรับขนาดของภาพที่ได้ในอัตราส่วนความกว้างต่อความสูงเป็น 2 ต่อ 3 ในโครงการนี้จะทำการปรับขนาดภาพให้มีจำนวนแถวและหลักของภาพเป็น 30 พิกเซล และ 20 พิกเซล ตามลำดับ และทำการ threshold อีกครั้ง เพื่อให้ภาพมีเพียงบิต 0 และบิต 1

3.2 การแยกคุณลักษณะเฉพาะจากข้อมูลภาพ (Feature extraction)

การแยกคุณลักษณะเฉพาะของภาพ เป็นการลดขนาดข้อมูลของภาพโดยทำการเก็บเพียงค่าสัมประสิทธิ์ที่สำคัญของข้อมูลภาพ

เมื่อได้ภาพที่ผ่านการปรับขนาดให้เท่ากันหมดแล้ว จะนำภาพเข้าสู่กระบวนการแปลงดิคคริตโดซายน์แบบ 2 มิติ ซึ่งจะใช้ DCT เมตริกซ์ขนาด 5×5 และ 10×10 หลังจากนั้นนำภาพที่ผ่านการแปลง DCT แล้ว มาเก็บค่าสัมประสิทธิ์ที่สำคัญ โดยภาพที่ผ่านการแปลงด้วย DCT เมตริกซ์ขนาด 5×5 จะทำการเก็บค่าสัมประสิทธิ์จำนวน 10 ค่าต่อเมตริกซ์ จากนั้นนำค่าที่ได้หลังการแปลงไปคูณกับบล็อกดังรูปที่ 3.14 และสร้างเป็นเมตริกซ์ใหม่ขนาด 1×240 ซึ่งค่า 240 ได้มาจาก $\left(\frac{30}{5}\right) \times \left(\frac{20}{5}\right) \times 10 = 240$ ส่วนภาพที่ผ่านการแปลงด้วย DCT เมตริกซ์ขนาด 10×10 จะทำการเก็บค่าสัมประสิทธิ์จำนวน 28 ค่าต่อเมตริกซ์ จากนั้นนำค่าที่ได้หลังการแปลงไปคูณกับบล็อกดังรูปที่ 3.15 และสร้างเป็นเมตริกซ์ใหม่ขนาด 1×168 ซึ่งค่า 168 ได้มาจาก $\left(\frac{30}{10}\right) \times \left(\frac{20}{10}\right) \times 28 = 168$

1	1	1	1	0
1	1	1	0	0
1	1	0	0	0
1	0	0	0	0
0	0	0	0	0

รูปที่ 3.14 บล็อก 5×5 การเก็บค่าสัมประสิทธิ์จำนวน 10 ค่า

1	1	1	1	1	1	1	0	0	0
1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

รูปที่ 3.15 บล็อก 10×10 การเก็บค่าสัมประสิทธิ์จำนวน 28 ค่า

3.3 การรู้จำป้ายทะเบียน (License-plate recognition)

หลังจากที่ผ่านขั้นตอน การตัดตัวอักษรและตัวเลขออกจากกันแล้ว ก็จะเข้าถึงส่วนของการรู้จำป้ายทะเบียน ซึ่งในส่วนนี้จะใช้นาโครงข่ายประสาทเทียมมาใช้ในการรู้จำป้ายทะเบียน ซึ่งผู้พัฒนาเลือกใช้โครงข่ายประสาทเทียมแบบหลายเลเยอร์แพร่กระจายป้อนกลับ (Back-propagation Neural Network) ซึ่งการใช้โครงข่ายประสาทเทียมจะแยกออกเป็น 2 ส่วนใหญ่ ๆ คือ ส่วนการฝึกสอน (Training) และการนำไปใช้งาน (Working) สามารถอธิบายขั้นตอนการทำงานได้ดังนี้

3.3.1 การฝึกสอนโครงข่ายประสาทเทียม (Training)

เป็นขั้นตอนในการทำให้โครงข่ายประสาทเทียมรู้จักตัวอักษรและตัวเลขบนป้ายทะเบียน โดยมีแม่แบบตัวอักษรและตัวเลขเรียกว่า แพทเทิร์น (Pattern) และเป้าหมายในการฝึกสอน (Target) ในการฝึกสอนจะมีค่าถ่วงน้ำหนัก (Weight) เป็นตัวแปรสำคัญต่อการใช้งานโครงข่ายประสาทเทียม ในขั้นตอนการฝึกสอนคือการปรับค่าถ่วงน้ำหนักไปจนกว่าจะได้ค่าถ่วงน้ำหนักที่เหมาะสม ซึ่งแม่แบบตัวอักษรที่จะใช้ในการฝึกสอนโครงข่ายประสาทเทียมคือตัวอักษรภาษาไทย 35 ตัวและตัวเลข 10 ตัว

เมื่อนำข้อมูลการเก็บคุณลักษณะเฉพาะของภาพมาทำการแปลงกลับดิสครีตโคซายน์จะได้ตัวอย่างภาพที่ใช้ในการฝึกสอนดังตารางที่ 3.1

ตารางที่ 3.1 ภาพตัวอักษรและตัวเลขหลังจากการแปลงกลับสัมประสิทธิ์ที่จะนำไปฝึกสอนให้กับโครงข่ายประสาทเทียม

ก	ข	ค
ด	จ	ฉ
ช	ฎ	ฏ
ฐ	ฑ	ฒ
ณ	ด	ด
น	ต	ถ
ท	ถ	อ
ธ	ท	๑
ด	ด	๒
ค	ค	๓
ข	ข	๔
บ	บ	๕
อ	อ	๖
๑	๑	๗
๒	๒	๘
๓	๓	๙

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลในตารางที่ 3.1 จะถูกนำไปฝึกสอนให้กับโครงข่าย โดยโครงสร้างของโครงข่ายจะใช้ Input layer จำนวน 168 โหนด สำหรับข้อมูลที่ทำกรแปลงดิคคริตโคซายน์ด้วย DCT เมตริกซ์ขนาด 10×10 และ Input layer จำนวน 240 โหนด สำหรับข้อมูลที่ทำกรแปลงดิคคริตโคซายน์ด้วย DCT เมตริกซ์ขนาด 5×5 ในส่วนของ Hidden layer และ Output layer จะให้จำนวนโหนดเท่ากันคือ 30 และ 45 โหนด ตามลำดับ และมีการกำหนดเป้าหมายในการฝึกสอน (Target) ดังตารางต่อไปนี้

ตารางที่ 3.2 การกำหนดเป้าหมายในการฝึกสอนให้กับโครงข่ายประสาทเทียม

ตัวอักษร	ตำแหน่งของโหนดของ Output layer ที่ต้องการค่าเอาต์พุตเท่ากับ 1 (นอกนั้นเป็นศูนย์ทั้งหมด)
ก	1
ข	2
ค	3
ง	4
จ	5
ฉ	6
ช	7
ฉ	8
ง	9
ฉ	10
ฉ	11
ค	12
ถ	13
ท	14
ธ	15
น	16
บ	17
ป	18
ผ	19
ฝ	20
พ	21
ฟ	22

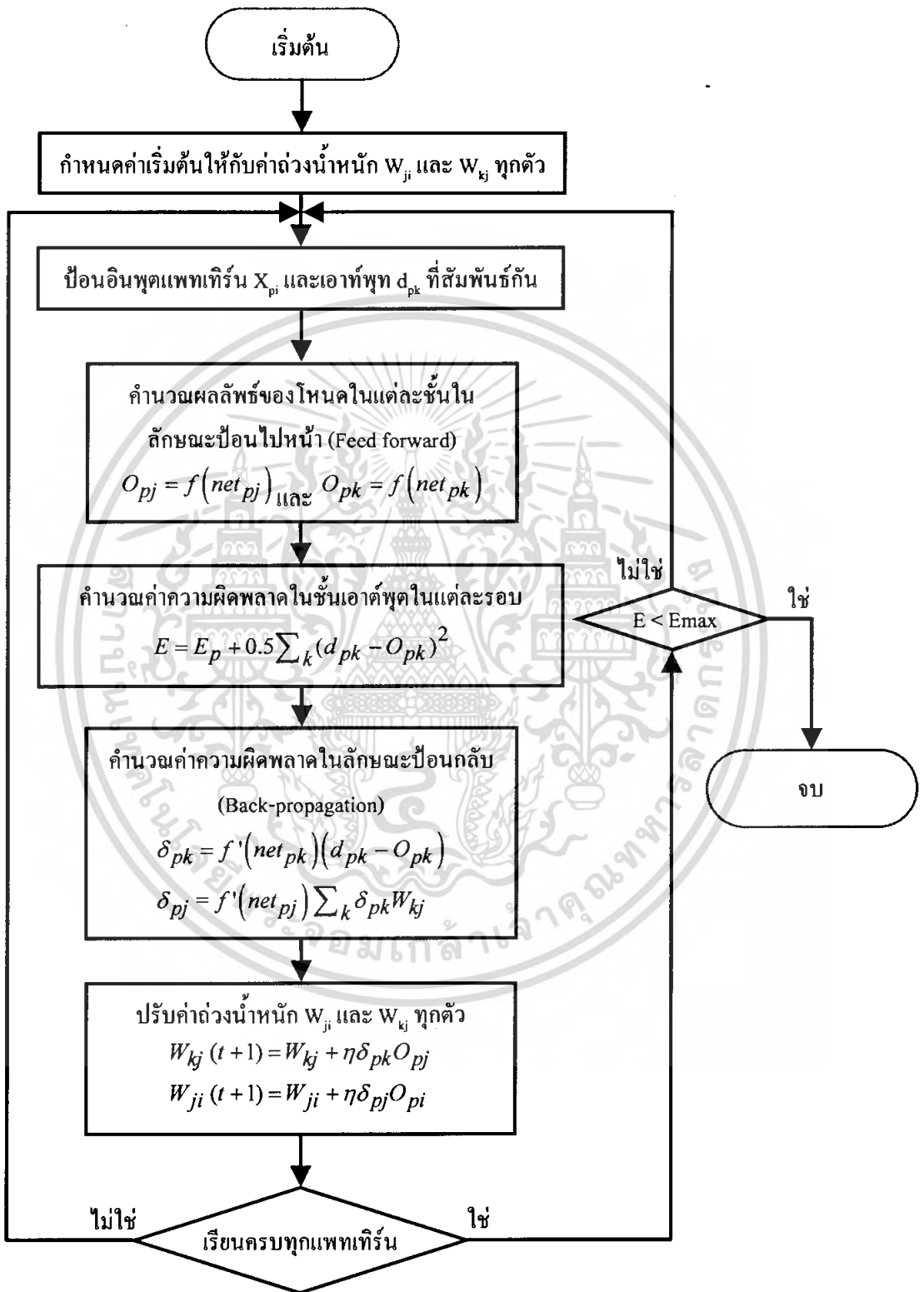
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 (ต่อ) การกำหนดเป้าหมายในการฝึกสอนให้กับโครงข่ายประสาทเทียม

ก	23
ข	24
ค	25
ด	26
ด	27
ว	28
ศ	29
ช	30
ส	31
ท	32
พ	33
อ	34
ฮ	35
0	36
1	37
2	38
3	39
4	40
5	41
6	42
7	43
8	44
9	45

หลังจากทำการกำหนดเป้าหมายในการฝึกสอนแล้ว ลำดับต่อไปคือการฝึกสอนโครงข่าย ซึ่งการฝึกสอนโครงข่ายมีกระบวนการ ดังที่แสดงในรูปที่ 3.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



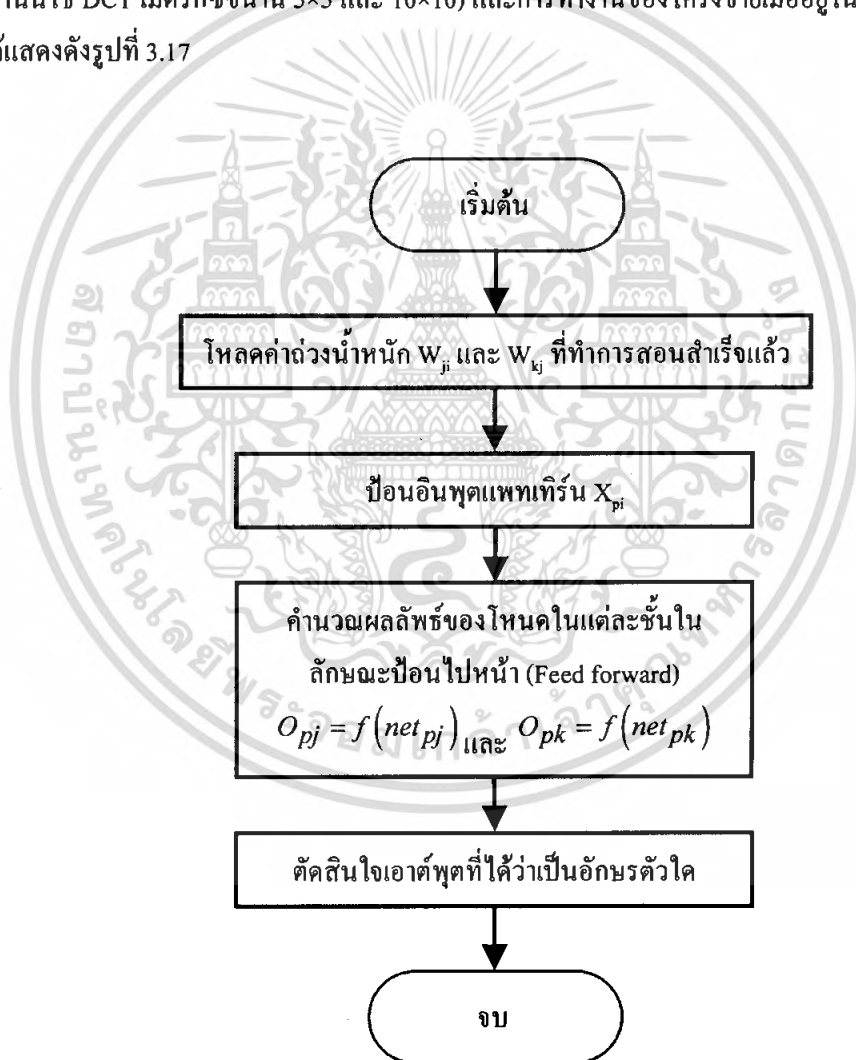
รูปที่ 3.16 แผนภูมิการฝึกสอนโครงข่ายประสาทเทียม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากทำการฝึกสอน โครงข่ายสำเร็จแล้วจึงทำการเก็บค่าถ่วงน้ำหนักพร้อมทั้งค่าตัวแปรต่างๆ ที่มีความสำคัญต่อระบบเป็นเพิ่มข้อมูลชื่อ NETWORK10.mat สำหรับข้อมูลลักษณะเฉพาะของภาพที่แปลงคิสคริตโคซายน์ด้วย DCT เมตริกซ์ขนาด 10×10 และเพิ่มข้อมูลชื่อ NETWORK5.mat สำหรับข้อมูลลักษณะเฉพาะของภาพที่แปลงคิสคริตโคซายน์ด้วย DCT เมตริกซ์ขนาด 5×5

3.3.2 การนำค่าถ่วงน้ำหนักไปใช้งาน (Working)

ในการใช้งานโครงข่ายประสาทเทียม จะนำค่าถ่วงน้ำหนักที่ได้หลังจากสิ้นสุดการฝึกสอนมาใช้งานโดยการโหลดเพิ่มข้อมูลชื่อ NETWORK10.mat หรือ NETWORK5.mat ซึ่งขึ้นอยู่กับอินพุตที่ป้อนให้กับโครงข่าย ว่าเป็นอินพุตที่ผ่านการแปลงคิสคริตโคซายน์ด้วย DCT เมตริกซ์ขนาดเท่าใด (ในโครงงานนี้ใช้ DCT เมตริกซ์ขนาด 5×5 และ 10×10) และการทำงานของโครงข่ายเมื่ออยู่ในขั้นตอนการใช้งานได้แสดงดังรูปที่ 3.17



รูปที่ 3.17 แผนภูมิการนำค่าถ่วงน้ำหนักไปใช้งาน ให้ประมวลผลรูปแบบตัวอักษรและตัวเลข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการตัดสินใจว่าเอาต์พุตที่ได้เป็นอักขระอะไร จะทำการพิจารณาจากค่าสูงสุดของ Output layer ที่โครงข่ายประมวลผลได้ว่าอยู่ที่โหนดใด และนำตำแหน่งของโหนดที่ได้ค่าสูงสุดนั้นไปอ้างอิงกับเป้าหมายในการฝึกสอนที่ได้ทำการกำหนดไว้ตั้งแต่ต้น

3.4 การอ้างอิงป้ายทะเบียนกับระบบฐานข้อมูล (Reference to Database)

การอ้างอิงกับฐานข้อมูล เป็นการนำผลที่ได้จากการแปลงข้อมูลคุณลักษณะเฉพาะของภาพเป็นข้อมูลตัวอักษรจากโครงข่ายประสาทเทียม ซึ่งในโครงการนี้ ได้ทำการจำลองระบบฐานข้อมูลที่มีข้อมูลของยานพาหนะคือ ข้อมูลป้ายทะเบียน จังหวัด ชื่อเจ้าของ จำนวนล้อ สีและยี่ห้อของยานพาหนะสำหรับเป็นฐานข้อมูลอ้างอิง และมีการจำลองฐานข้อมูลตรวจสอบให้เป็นตัวอย่างการใช้งานการจำแนกชนิดยานพาหนะสำหรับระบบทางด่วน โดยมีการเก็บข้อมูลเช่นเดียวกับฐานข้อมูลอ้างอิง แต่จะเพิ่มการบันทึกเวลาที่ทำการจำแนกยานพาหนะ (Time stamp) และข้อมูลที่สำคัญอื่นๆ ซึ่งมีโครงสร้างการเชื่อมต่อระหว่างโปรแกรม MATLAB กับฐานข้อมูล MySQL ดังรูปที่ 3.18



รูปที่ 3.18 การเชื่อมต่อระหว่าง โปรแกรม MATLAB กับฐานข้อมูล

จากรูปที่ 3.18 เป็นการเชื่อมต่อระหว่างโปรแกรม MATLAB ซึ่งใช้ในการประมวลผลภาพ ดิจิตอลรวมไปจนถึงการให้ผลลัพธ์ข้อมูลตัวอักษรของภาพป้ายทะเบียนกับฐานข้อมูลจำลองที่ใช้ MySQL โดยมีตัวกลางการเชื่อมต่อคือ ODBC (Open Database Connection) ที่ทำการเชื่อมต่อกับฐานข้อมูลผ่านโปรโตคอล TCP/IP

ในส่วนฐานข้อมูลจำลองจะใช้คำสั่ง SQL ในการสร้างฐานข้อมูล ตารางและฟิลด์ ดังต่อไปนี้

1. ทำการสร้างฐานข้อมูลชื่อ expressway และ event_log โดยให้ฐานข้อมูลมีการสนับสนุนภาษาไทยโดยใช้คำสั่ง

```
mysql> create database expressway
-> character set tis620 collate tis620_thai_ci;
-> create database event_log
-> character set tis620 collate tis620_thai_ci;
```

2. ใช้ฐานข้อมูลที่สร้างขึ้นและทำการสร้างตารางชื่อ data สร้างฟิลด์ชื่อ license, province, name, wheel, color, brand และให้ฟิลด์ชื่อ license เป็นดัชนีในการค้นหาสำหรับฐานข้อมูลอ้างอิงและทำการสร้างตารางชื่อ data เช่นเดียวกันกับกับฐานข้อมูลอ้างอิง สร้างฟิลด์ชื่อ license, province, name, wheel, color, brand, datetime, fee และให้ฟิลด์ชื่อ datetime เป็นดัชนีในการค้นหาสำหรับฐานข้อมูลตรวจสอบ โดยใช้คำสั่ง

```
mysql> use expressway;
-> create table data(
-> license varchar(6) not null primary key,
-> province varchar(15) not null,
-> name varchar(30) not null,
-> wheel tinyint(4) not null,
-> color varchar(30) not null,
-> brand varchar(15) not null);
-> create table data(
-> license varchar(6),
-> province varchar(15) not null,
-> name varchar(30) not null,
-> wheel tinyint(4) not null,
-> color varchar(30) not null,
-> brand varchar(15) not null,
-> datetime varchar(20) not null primary key,
-> fee tinyint(4) not null);
```

```
mysql> use expressway;
Database changed
mysql> show columns from data;
```

Field	Type	Null	Key	Default	Extra
license	varchar(6)	NO	PRI		
province	varchar(15)	NO			
name	varchar(30)	NO			
wheel	tinyint(4)	NO			
color	varchar(30)	NO			
brand	varchar(15)	NO			

```
6 rows in set (0.00 sec)

mysql> use event_log;
Database changed
mysql> show columns from data;
```

Field	Type	Null	Key	Default	Extra
license	varchar(6)	NO			
province	varchar(15)	NO			
name	varchar(30)	NO			
wheel	tinyint(4)	NO			
color	varchar(30)	NO			
brand	varchar(15)	NO			
datetime	varchar(20)	NO	PRI		
fee	tinyint(4)	NO			

```
8 rows in set (0.01 sec)

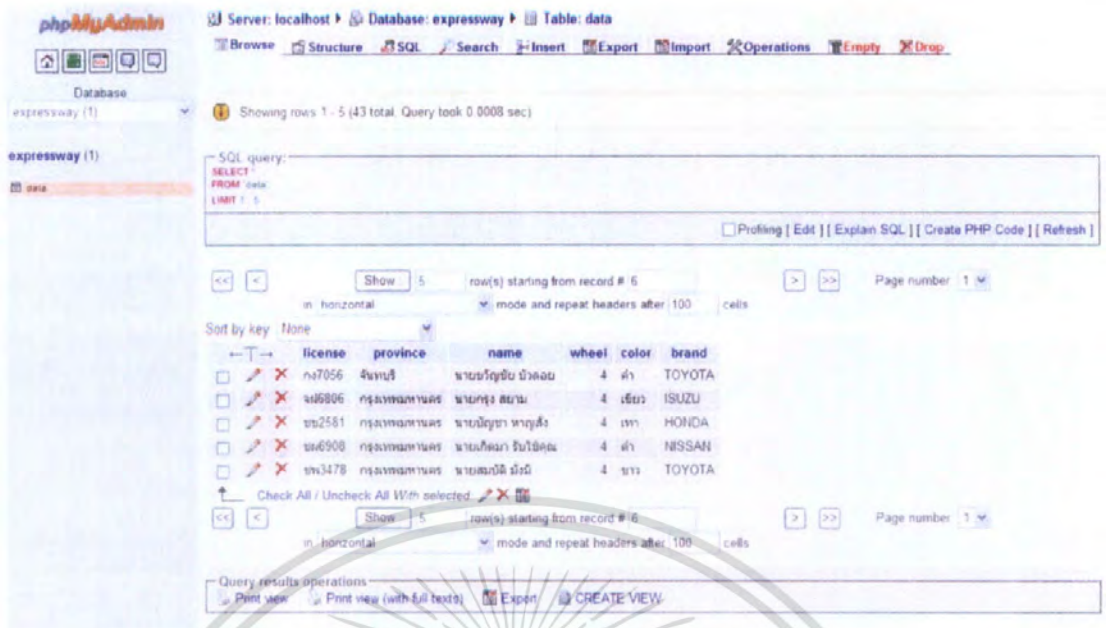
mysql>
```

รูปที่ 3.19 โครงสร้างของฐานข้อมูลจำลองที่สร้างขึ้นเมื่อใช้คำสั่ง

```
mysql> show columns from data;
```

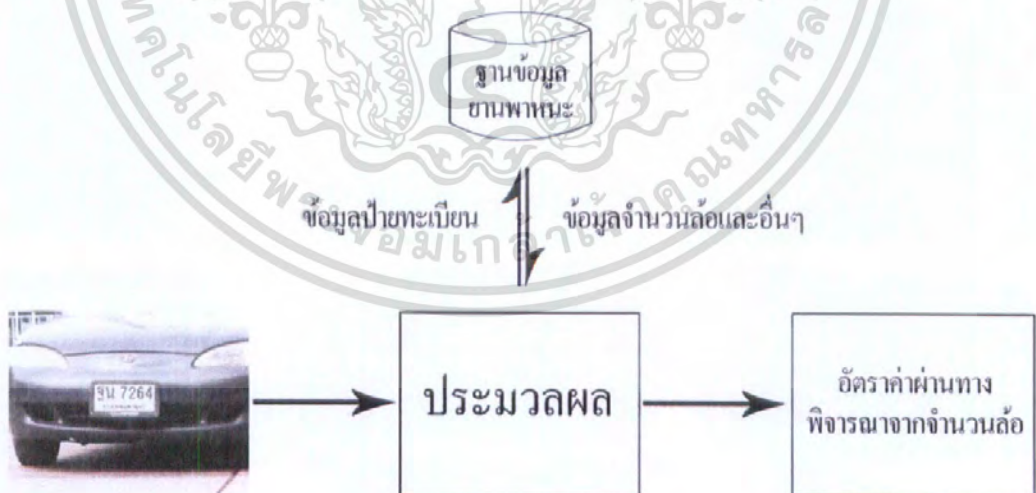
เมื่อระบบฐานข้อมูลถูกสร้างขึ้นแล้วจะทำการป้อนข้อมูลของยานพาหนะที่สมมติขึ้น โดยใช้เครื่องมือช่วยในการจัดการเกี่ยวกับฐานข้อมูลคือ phpMyAdmin ผ่านระบบบริการเว็บ (HTTP Service)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.20 ซอฟต์แวร์ phpMyAdmin ที่ใช้ในการจัดการกับฐานข้อมูลที่บริการผ่านเว็บ

ในการทำงานร่วมกับระบบฐานข้อมูล จะเป็นการอ้างอิงทะเบียนของยานพาหนะที่ได้จากผลของการรู้จำของโครงข่ายประสาทเทียมไปค้นหาข้อมูลอื่นๆ ของยานพาหนะ โดยหลักสำคัญคือ จะนำข้อมูลจำนวนล้อของยานพาหนะมาพิจารณาเป็นอัตราค่าผ่านทางสำหรับทางด่วน



รูปที่ 3.21 ขั้นตอนการคิดอัตราค่าผ่านทางโดยการพิจารณาจากจำนวนล้อของยานพาหนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลองและการวิเคราะห์

4.1 การประมวลผลภาพดิจิทัล (Digital Image Processing)

1. ภาพของด้านหน้าของยานพาหนะพร้อมป้ายทะเบียนที่จับภาพด้วยโปรแกรม



(ค)

รูปที่ 4.1 ภาพต้นฉบับของยานพาหนะพร้อมทะเบียน (ก) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีดำ

(ข) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีฟ้า (ค) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีเขียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การแปลงภาพต้นฉบับเป็นภาพขาว-ดำ (Gray scale) 256 ระดับ จะได้ผลลัพธ์ดังรูปที่ 4.2



(ก)



(ข)

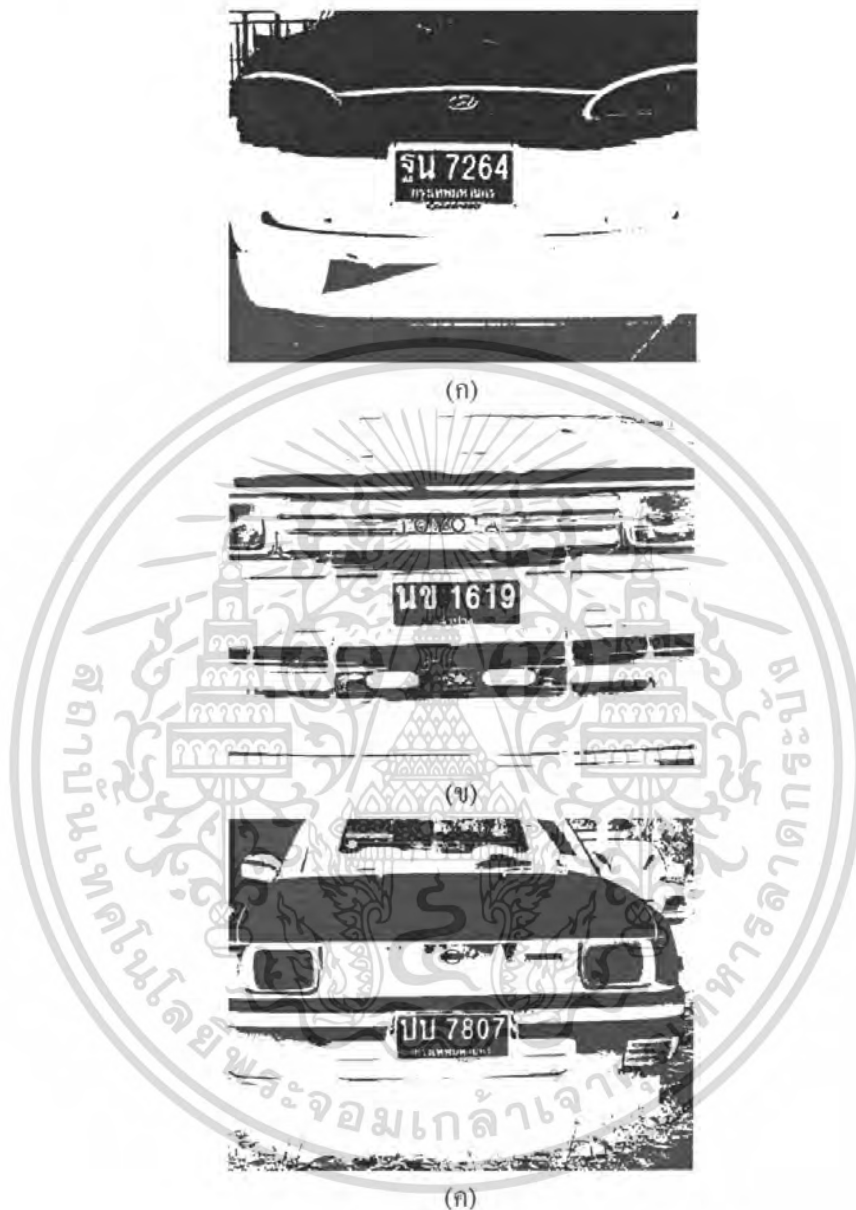


(ค)

รูปที่ 4.2 ภาพต้นฉบับที่ทำการแปลงเป็นภาพ Gray Scale (ก) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีดำ (ข) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีฟ้า (ค) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีเขียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

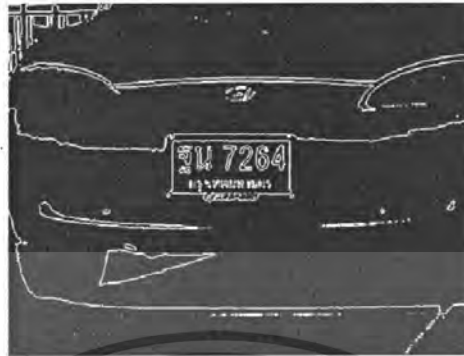
3. การแปลงภาพ Gray Scale เป็นภาพไบนารี จะได้ผลลัพธ์ดังรูปที่ 4.3



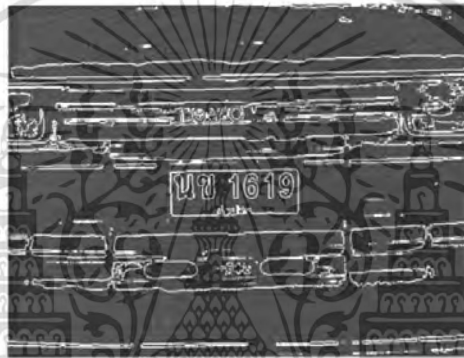
รูปที่ 4.3 ภาพ Gray Scale ที่ทำการแปลงเป็นภาพไบนารี (ก) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีดำ (ข) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีฟ้า (ค) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีเขียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. การหาขอบภาพด้วยวิธี Sobel edge detection จะได้ผลลัพธ์ดังรูปที่ 4.4



(ก)



(ข)

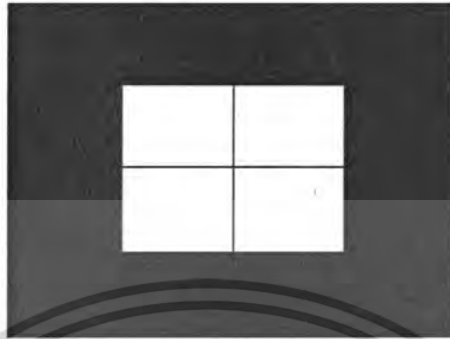


(ค)

รูปที่ 4.4 ภาพหลังจากการหาขอบภาพด้วยวิธี Sobel edge detection (ก) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีดำ (ข) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีฟ้า (ค) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีเขียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เมื่อได้ภาพที่ทำการหาขอบภาพแล้ว จะนำหน้าภาพมาครอบเพื่อเป็นการลดข้อมูลในการวิเคราะห์ลง โดยหน้าภาพที่นำมาครอบจะถูกแบ่งเป็นสี่ส่วน 4x4 ของภาพต้นฉบับ



รูปที่ 4.5 ภาพหน้าภาพขนาด 4x4 ส่วนของภาพต้นฉบับ

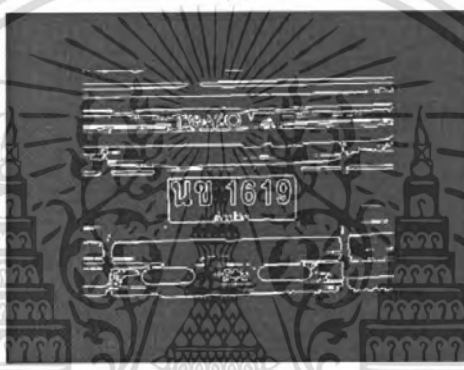


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. หลังจากทำการครอบด้วยหน้ากากในรูปที่ 4.5 จะได้ผลลัพธ์ดังรูปที่ 4.6



(ก)



(ข)

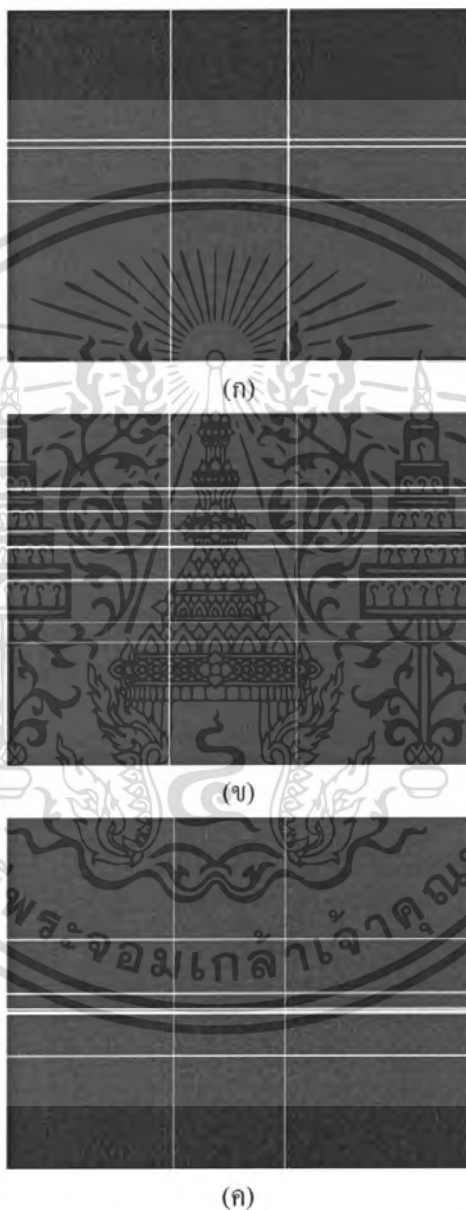


(ค)

รูปที่ 4.6 ภาพที่ได้จากการครอบด้วยหน้ากาก (ก) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีดำ (ข) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีฟ้า (ค) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีเขียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ตรวจสอบความเป็นไปได้ของตำแหน่งทะเบียนโดยนำค่าในอาร์เรย์ (r, θ) จากการแปลงฮัฟ ที่มุม $\theta = 0^\circ$ มาพิจารณาเพื่อหาตำแหน่งที่เป็นขอบด้านซ้ายและด้านขวาของป้ายทะเบียน จากนั้นนำค่าในอาร์เรย์ (r, θ) ที่มุม $\theta = 90^\circ$ มาพิจารณาหาตำแหน่งที่เป็นขอบด้านล่างและด้านบนของป้ายทะเบียน ซึ่งตำแหน่งที่หาได้แสดงในรูปที่ 4.7



รูปที่ 4.7 เส้นแสดงตำแหน่งที่คาดว่าเป็นตำแหน่งของป้ายทะเบียน (ก) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีดำ (ข) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีฟ้า (ค) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีเขียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากนั้นทำการคัดลอกข้อมูลภาพใบนารีในตำแหน่งที่หาได้จะได้ผลลัพธ์ดังรูปที่ 4.8

รฐน 7264
กรุงเทพมหานคร นคร

(ก)

นข 1619
ลำปาง

(ข)

บบ 7807
กรุงเทพมหานคร นคร

(ค)

รูปที่ 4.8 ภาพผลลัพธ์จากการหาดำแหน่งป้ายทะเบียน (ก) ภาพป้ายทะเบียนพื้นสีขาว
ตัวอักษรสีดำ (ข) ภาพป้าย ทะเบียนพื้นสีขาว ตัวอักษรสีฟ้า (ค) ภาพป้ายทะเบียนพื้นสี
ขาว ตัวอักษรสีเขียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ทำการปรับแต่งภาพ (Image Enhancement) ทำให้ภาพมีสัญญาณรบกวน (Noise) ลดลงจะได้ผลลัพธ์ดังรูปที่ 4.9

จน 7264
กรุงเทพมหานคร

(ก)

นข 1619
ฉะเชิงเทรา

(ข)

บบ 7807
กรุงเทพมหานคร

(ค)

- รูปที่ 4.9 ภาพป้ายทะเบียนหลังจากปรับแต่งภาพ (ก) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีดำ (ข) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีฟ้า (ค) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีเขียว

9. การตัดส่วนที่เป็นจังหวัดออก จะได้ผลลัพธ์ดังรูปที่ 4.10

จน 7264

(ก)

นข 1619

(ข)

บบ 7807

(ค)

- รูปที่ 4.10 ภาพป้ายทะเบียนหลังจากที่ถูกตัดส่วนของจังหวัดออก (ก) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีดำ (ข) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีฟ้า (ค) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีเขียว

10. การตัดแยกตัวอักษรและตัวเลขให้เป็นส่วนๆ จะได้ผลลัพธ์ดังรูปที่ 4.11

รูปที่ 4.11 (ก)

(ก)

รูปที่ 4.11 (ข)

(ข)

รูปที่ 4.11 (ค)

(ค)

รูปที่ 4.11 ภาพหลังจากการแยกตัวอักษร (ก) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีดำ
(ข) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีฟ้า (ค) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีเขียว

11. ทำการ Normalize โดยการปรับขนาดภาพให้เท่ากันคือ 30×20 พิกเซล จะได้ผลลัพธ์ดังรูปที่ 4.12

รูปที่ 4.12 (ก)

(ก)

รูปที่ 4.12 (ข)

(ข)

รูปที่ 4.12 (ค)

(ค)

รูปที่ 4.12 ภาพหลังจากการปรับขนาดให้เท่ากันทุกตัวอักษร (ก) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีดำ
(ข) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีฟ้า (ค) ภาพป้ายทะเบียนพื้นสีขาว ตัวอักษรสีเขียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การแยกคุณลักษณะเฉพาะจากข้อมูลภาพ (Feature extraction)

หลังจากภาพถูกปรับขนาดให้เท่ากันแล้ว จะเข้าสู่กระบวนการเก็บคุณลักษณะเฉพาะโดยการแปลงคีสคริตโคซายน์ ซึ่งจะใช้ DCT เมตริกซ์ ขนาด 5×5 และ 10×10 และมีการเปรียบเทียบประสิทธิภาพของการบีบอัด รวมถึงการเปรียบเทียบในด้านของความถูกต้องของผลลัพธ์ โดยจะใช้ค่า PSNR (Peak Signal to Noise Ratio) ซึ่งเป็นค่าที่ใช้แสดงความถูกต้องของผลลัพธ์ มีหน่วยเป็นเดซิเบล (dB) ซึ่งในการหาค่า PSNR สามารถทำได้จากสมการที่ (4.1) และ (4.2)













$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \|I(i, j) - K(i, j)\|^2 \quad (4.1)$$

$$PSNR = 20 \times \log_{10} \left(\frac{MAX_I}{\sqrt{MSE}} \right) \text{ (dB)} \quad (4.2)$$

โดย	$I(i, j)$	คือภาพต้นฉบับ
	$K(i, j)$	คือภาพที่ได้หลังการแปลงกลับคีสคริตโคซายน์
	MAX_I	คือค่าสูงสุดของประเภทของภาพต้นฉบับ เช่น ภาพ Gray scale MAX_I เท่ากับ $2^8 - 1$ ภาพไบนารี MAX_I เท่ากับ $2^1 - 1$

ผลลัพธ์ของการแปลงคีสคริตโคซายน์พร้อมทั้งค่า PSNR แสดงในตารางที่ 4.1 และ 4.2

ตารางที่ 4.1 การเปรียบเทียบการแปลงดิคริตโคซายน์ โดยใช้ DCT เมตริกซ์ ขนาด 5×5 กับภาพต้นฉบับ

ภาพต้นฉบับ	ภาพหลังการแปลง DCT กลับ แบบ 5×5 เก็บสัมประสิทธิ์ 10 ค่า	PSNR (dB)
		14.62
		14.72
		18.30
		16.36
		14.99
		16.88

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

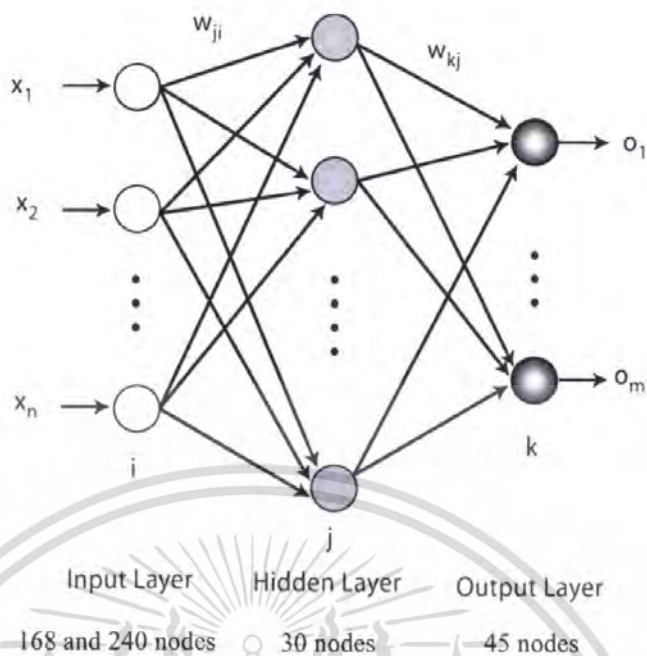
ตารางที่ 4.2 การเปรียบเทียบการแปลงดิสครีตโคซายน์ โดยใช้ DCT เมตริกซ์ ขนาด 10×10 กับภาพ
ต้นฉบับ

ภาพต้นฉบับ	ภาพหลังการแปลง DCT กลับ แบบ 10×10 เก็บสัมประสิทธิ์ 28 ค่า	PSNR (dB)
		13.63
		13.63
		16.99
		15.41
		13.82
		16.04

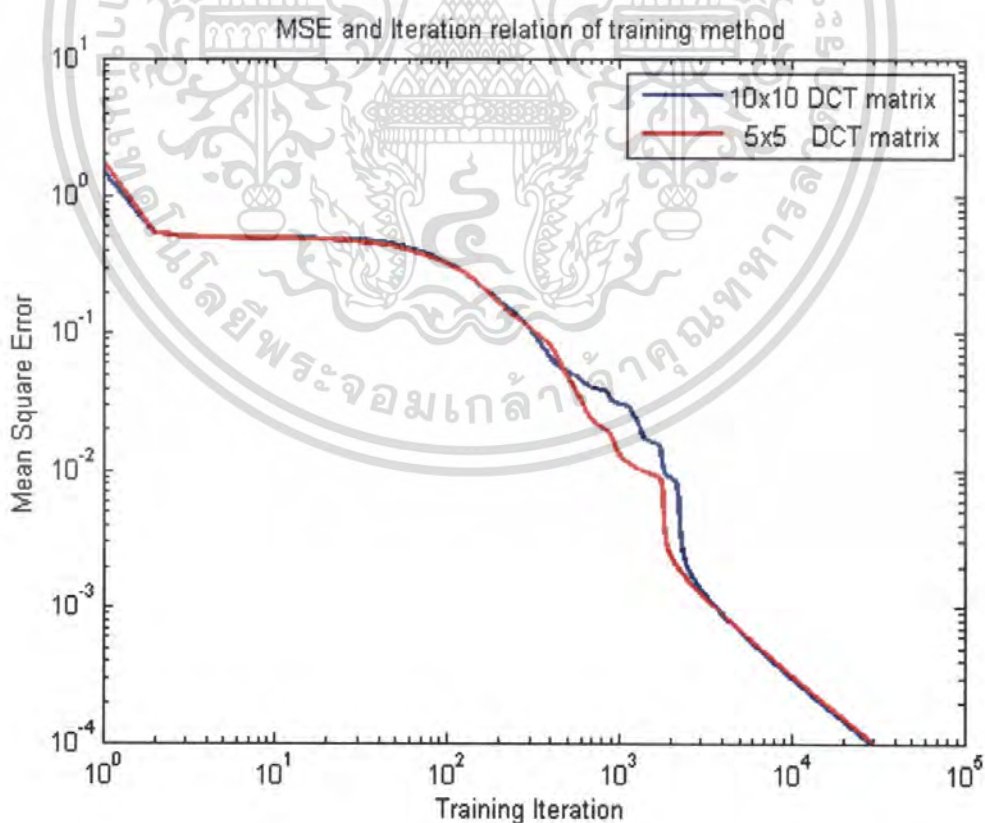
4.3 การรู้จำป้ายทะเบียน (License-plate recognition)

1. ในการฝึกสอนโครงข่ายประสาทเทียม โครงข่ายประสาทเทียมแบบแพร่กระจายกลับที่ใช้ใน
โครงข่ายนี้จะมีโครงข่าย 2 โครงข่ายคือ โครงข่ายแบบอินพุตโหนดจำนวน 168 โหนด และ
โครงข่ายแบบอินพุตโหนด 240 โหนด ทั้งสองโครงข่ายมีจำนวนโหนดในชั้นฮิดเดนและ
เอาต์พุตเท่ากันคือ 30 โหนด และ 45 โหนดตามลำดับ ดังแสดงในรูปที่ 4.13

ตัวแปรที่จะทำให้ทราบว่าโครงข่ายสามารถเรียนรู้ได้สำเร็จคือ ค่าผิดพลาดเฉลี่ยกำลังสอง (MSE: Mean Square Error) เมื่อโครงข่ายทำการเรียนรู้ข้อมูลได้มากขึ้น ค่า MSE ของระบบจะลดลงเรื่อยๆ จนเข้าสู่ศูนย์ ซึ่งในการฝึกสอนโครงข่ายประสาทเทียมให้รู้จักอินพุตตัวอักษรต่างๆ จนสำเร็จ สามารถแสดงเป็นความสัมพันธ์ระหว่างค่า MSE กับจำนวนรอบ (Iteration) ที่ทำการฝึกสอน (หนึ่งรอบของการฝึกสอนคือการให้โครงข่ายเรียนรู้ตัวอักษรครบทุกตัว) โดยกำหนดค่าผิดพลาดเฉลี่ยกำลังสองต่ำสุดไว้ที่ 10^{-4} ดังรูปที่ 4.14 และค่าถ่วงน้ำหนักบางส่วนที่ได้หลังจากการฝึกสอนสำเร็จแสดงในตารางที่ 4.3 และตารางที่ 4.4



รูปที่ 4.13 โครงสร้างระบบโครงข่ายประสาทเทียมที่ใช้ในการฝึกสอน



รูปที่ 4.14 กราฟความสัมพันธ์ระหว่างค่าผิดพลาดเฉลี่ยกำลังสองกับจำนวนรอบที่ทำการฝึกสอนเมื่อ

กำหนดค่าผิดพลาดเฉลี่ยกำลังสองต่ำสุดไว้ที่ 10^{-4} นั้น ไม่นิยามให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 ค่าถ่วงน้ำหนักของชั้นฮิดเดนโหนด 1 ถึง 8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 168 ค่า

โหนด 1	โหนด 2	โหนด 3	โหนด 4	โหนด 5	โหนด 6	โหนด 7	โหนด 8
-0.0264	0.7029	-0.2437	0.3119	-0.0938	-0.4065	0.9284	0.2966
0.0847	-0.5244	-0.5230	-0.7662	-0.2541	0.3655	0.0840	0.0823
0.1368	-0.2045	0.0911	0.3555	0.5328	0.2741	-0.7916	-0.6710
-0.6472	-0.4896	-0.0506	-0.0470	0.0605	-0.6852	0.5094	0.0905
-0.2082	0.4672	-0.4991	0.4614	0.6427	0.2619	-0.7522	-0.1426
0.0445	-0.3915	0.5108	0.4911	0.1330	-0.5217	-0.1265	0.3769
0.0326	-0.2441	0.1888	-0.5988	0.1398	-0.1869	-0.1225	0.1153
-0.0218	0.0221	-0.4373	-0.0913	0.7299	-0.0627	-0.3674	-0.5947
-0.7412	-0.0889	0.5351	-0.7599	-0.0028	-0.0213	-0.1474	-0.6056
-0.1690	-0.0509	-0.5769	0.4854	-0.2647	1.2020	0.5334	-0.3232
-0.2350	0.2299	-0.8736	-0.0446	0.4522	0.1147	0.1873	0.4203
-0.2058	-0.1013	-0.0760	0.6216	-0.2187	-0.4577	-0.3544	-0.2127
-0.5838	0.1625	0.6783	0.2031	-0.5654	-0.4924	-0.2251	0.5776
0.5138	0.3625	0.2201	-0.2431	0.0390	-0.2858	0.0172	-0.0887
0.1589	0.3046	-1.3512	-0.1636	-0.1192	-1.3176	-0.8510	-0.1094
1.0369	-0.2266	-0.5336	-0.1467	0.3046	0.1290	-0.5805	0.3014
0.4334	-0.0211	0.0718	0.0366	0.0592	0.7233	0.0536	-0.2601
0.3221	0.1729	0.1816	0.4473	-0.0540	0.7250	-0.2345	-0.4238
-0.2069	-0.4394	0.1710	-0.4500	-0.4959	0.6128	0.0566	0.0006
0.2966	0.3661	-0.1098	0.1146	-0.0870	0.7502	0.2456	-0.4068
0.0512	0.0544	0.4815	0.2222	0.5295	0.0925	0.3726	0.5250
0.2478	-0.1241	0.5796	-0.1453	-1.0922	-0.5640	-0.2860	-0.0870
-0.7795	0.0641	0.6340	0.4965	0.3942	-0.3014	-0.2122	0.1801
-0.3565	-0.4360	0.6491	-0.3075	-0.0609	-0.4814	0.2458	0.3407
-0.2295	-0.1353	-0.5026	-0.1353	-0.3180	-0.1749	-0.2415	0.3145
0.0483	-0.4197	0.3492	-0.1756	-0.0078	-0.4539	-0.1019	-0.5381
0.5017	0.0079	-0.6680	-0.3140	0.6592	-0.0468	-0.4430	-0.1468
0.4923	-0.2113	0.8484	0.2055	0.3551	-1.0423	0.3984	-0.2602
-0.3331	0.1098	0.4645	0.4571	0.6894	-0.0979	0.4055	-0.0345
-0.8589	-0.5523	-0.1920	0.6369	0.6523	0.1937	-0.7369	-0.1091
0.3555	-0.3080	-0.1216	-0.0131	-0.7513	-0.0489	0.3328	0.4328

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของโรงเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ไปยังผู้อื่นโดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 (ต่อ) ค่าถ่วงน้ำหนักของชั้นฮิดเดน โหนด 1 ถึง 8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 168 ค่า

-0.2438	-0.5272	0.2398	-0.3509	0.8621	0.0054	0.4490	0.1690
0.6742	0.1859	-0.1756	-0.0941	-0.5905	0.0679	0.2331	-0.0672
-0.4358	0.2758	0.2604	0.6218	0.2008	0.2180	0.5191	-0.4290
-0.4517	-0.1380	-0.4062	0.0317	-0.1204	-0.4629	0.3600	-0.1194
0.5218	-0.4296	0.3451	-0.4104	-0.0265	0.1803	0.1120	0.3123
-0.3102	0.1210	0.5115	0.3874	0.4295	-0.0615	0.1133	-0.2922
-0.4795	-0.1811	0.1802	0.2886	-0.0505	-0.0500	-0.3008	0.4640
-0.1738	-0.1874	0.5272	0.0548	0.2987	0.1344	-0.6425	0.0918
-0.1175	0.0420	0.3038	-0.4639	0.3956	0.4572	0.3318	-0.3468
0.1953	0.0084	-0.0843	-0.0576	-0.3516	0.4207	-0.6350	0.4180
0.5652	-0.0840	-0.3414	-0.1969	0.6647	0.2000	0.2782	0.0452
-0.1179	0.1065	0.0138	-0.4007	0.0648	-0.4375	0.4115	0.3069
0.4276	0.3522	-0.0729	0.0897	0.1914	0.5205	0.2828	-0.4150
-0.0844	-0.3731	0.4685	0.3473	0.3185	-0.1028	0.0721	0.0950
-0.1799	-0.2904	-0.1196	-0.3565	-0.1165	-0.1499	0.0474	0.1396
0.5652	0.0795	0.1770	-0.1861	-0.4165	0.0761	-0.4044	0.3854
0.1067	-0.1533	-0.0268	-0.0646	-0.5079	-0.0335	0.1543	-0.0320
0.0347	0.0187	-0.1683	0.0117	-0.3772	0.1121	0.2877	-0.2786
-0.0816	0.1009	0.3367	-0.4284	0.0957	-0.1343	-0.3305	0.5672
-0.5052	-0.2448	-0.2817	0.0594	0.0994	-0.2387	-0.2300	-0.1759
-0.2643	-0.0661	-0.3398	-0.6609	0.5333	-0.3047	0.9466	0.1212
-0.1822	0.1971	-0.2549	-0.0061	-0.2466	-0.1827	-0.1655	0.0832
0.0816	0.0700	0.2799	0.2327	-0.3213	0.1640	-0.3783	-0.3675
0.1608	-0.1631	-0.0013	-0.3572	-0.1524	0.5329	-0.4375	0.1900
0.1281	-0.2772	0.4656	0.4468	0.5213	-0.0099	-0.3923	0.5195
0.1371	0.4546	0.5001	0.0865	-0.9632	-0.0766	0.1562	-1.3432
-0.5731	-0.0488	0.0839	-0.7932	-0.4519	0.1119	-0.0797	-0.3417
0.3361	0.6458	-0.2161	0.8912	-0.5773	0.2578	1.1607	-0.0922
0.7372	-0.1216	-0.3571	0.1915	0.5963	0.2724	0.7526	0.4146
-0.2752	-0.2749	-0.3451	0.5199	0.0822	-0.3951	-0.3927	0.5714
-0.4474	0.0412	-0.5735	0.5981	0.2012	-0.9567	-0.0264	-0.4094

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 (ต่อ) ค่าถ่วงน้ำหนักของชั้นฮิดเดนโหนด 1 ถึง 8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 168 ค่า

-0.0506	0.0036	0.7571	-0.5927	0.5146	-0.0092	-0.0966	0.4563
0.8580	-0.4019	0.5015	0.7257	-0.2987	-0.4988	-0.2664	-0.0838
0.8516	0.3078	0.1519	0.5189	-0.8971	0.5303	-0.2247	-1.0051
0.4314	-0.2256	-0.0328	0.0553	-1.1840	-0.3226	0.7280	-0.1725
-0.1338	0.1645	-0.5120	0.1443	0.1978	0.8889	0.8640	0.2623
-0.5492	0.3221	-0.5321	0.3771	0.7418	-0.5149	-0.5169	-0.6599
-0.3511	-0.0699	0.4859	0.1018	0.5620	-0.3380	0.1659	0.3875
0.8284	-0.4599	-0.3208	0.1566	-0.3326	0.7555	0.6137	-0.0722
0.6153	0.0398	0.5376	-0.0430	-0.2144	-0.1317	0.0264	-0.3156
0.8519	-0.3720	-0.5141	-0.4314	-0.4748	-0.5233	0.5817	-0.3424
-0.3374	0.0627	0.4060	0.2610	0.2180	0.3894	0.2808	-0.1106
0.0325	-0.4347	-0.1283	-0.3414	0.1880	-0.3827	0.0981	0.0181
0.2841	0.5055	0.3363	-0.2024	-0.0223	-0.0001	0.0176	-0.2424
-0.2446	0.0517	-0.3232	0.3668	0.0506	0.4200	0.0222	0.1414
-0.0996	0.4364	-0.0256	-0.0626	0.8656	-0.6631	-0.2640	-0.2870
-0.2655	0.5954	0.4251	-0.3552	0.0898	0.3229	0.2126	0.3412
0.2964	-0.2021	-0.3030	-0.2493	0.2857	0.3002	-0.1168	-0.0297
0.0224	0.3593	-0.2780	0.3976	-0.2601	0.5119	-0.0318	0.3985
0.2899	-0.3458	-0.0462	0.2016	0.2326	-0.7395	0.1223	0.1696
-0.4978	-0.1056	0.1149	0.3642	-0.2354	-0.3184	-0.1491	0.4058
-0.0098	-0.3814	0.0005	-0.2685	-0.3318	-0.1153	0.1262	0.1346
0.3960	0.4435	-0.2580	-0.2665	0.6904	0.5504	-0.2483	0.2395
0.5413	-0.2305	-0.1212	0.7067	-0.8965	-0.3286	0.4315	0.2804
0.2824	-0.2564	0.1705	0.0760	0.5211	0.1245	-0.2526	-0.2943
0.2559	-0.3610	0.4608	-0.4734	0.1368	0.2122	-0.0046	0.0094
0.0381	-0.0232	0.5387	-0.1323	-0.0574	-0.2236	0.2297	0.5441
-0.1079	-0.1389	-0.2805	-0.6938	-0.6772	-0.8050	0.5266	-0.2134
0.3810	-0.1519	0.2324	0.0162	-0.2821	-0.0395	0.0438	-0.6776
0.1073	0.0373	-0.4149	0.3236	0.2689	-0.5973	-0.0656	-0.4225
-0.3169	0.4323	-0.0305	-0.1226	0.3298	-0.2976	-0.2290	0.1657
0.2023	-0.0868	-0.0786	-0.1856	0.0954	0.0331	-0.2778	-0.6903
-0.0448	-0.3369	-0.4745	0.1741	-0.2701	0.0332	0.4326	0.0837

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 (ต่อ) ค่าถ่วงน้ำหนักของชั้นฮิดเดนโหนด 1 ถึง 8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 168 ค่า

-0.2347	0.3081	-0.2439	-0.3572	0.4664	0.2292	0.4717	0.3826
0.2884	-0.0256	-0.2694	-0.1607	-0.2415	0.4437	0.0888	-0.1431
0.3433	0.4329	-0.3359	-0.1669	0.5493	-0.3209	0.3824	0.3739
-0.5543	-0.0467	-0.0818	-0.0998	0.2242	-0.3714	0.5501	0.1438
0.0066	-0.4375	0.4532	-0.3991	-0.0326	-0.4675	0.1683	0.1851
-0.1545	0.2241	-0.0523	0.2820	-0.0629	0.2582	-0.4647	0.1612
-0.1731	-0.1855	0.4485	0.1617	0.1261	-0.2337	0.2573	0.2392
-0.2021	-0.1085	-0.4599	0.3769	0.0995	-0.4818	0.1411	-0.5197
0.1278	-0.0548	0.2135	-0.2102	-0.3336	0.1139	0.1428	-0.2903
-0.0153	-0.2042	0.5366	-0.2504	0.5189	-0.2749	-0.3371	0.2769
0.2584	-0.2682	-0.0512	0.4896	0.2342	-0.1143	0.2111	0.4999
-0.1867	-0.4588	0.3399	0.4098	-0.2869	-0.1605	0.2395	0.0810
-0.5694	0.2506	-0.1292	0.0200	0.2611	0.2654	0.2508	-0.4106
0.5784	0.0016	0.4593	-0.0605	0.4132	-0.1171	0.3010	0.5060
-0.2431	-0.0039	0.3167	0.0204	-0.1294	-0.0409	-0.0287	-0.0820
-0.2290	0.1878	-0.3802	-0.0304	0.2064	0.3699	0.3337	0.2707
0.1477	-0.1338	-0.3688	0.1924	-0.1005	0.4959	0.1895	0.3641
-0.2990	-0.1080	-0.1373	0.3006	0.4694	0.1835	0.2924	0.5361
0.4542	-0.0556	0.0115	0.1331	0.3210	0.5523	-0.3485	0.2556
-0.6082	0.4302	-0.5947	0.5470	0.0085	0.2124	0.7182	-0.8865
-0.2227	0.3851	-0.2652	-0.3174	-0.3439	-0.3699	0.5286	-0.0228
-0.1146	-0.0218	-0.1691	-0.2199	0.4682	-0.3069	1.0891	0.1543
-0.2659	0.0922	-0.1734	-0.5859	-0.0932	-0.2200	-0.0046	-0.1036
0.2915	0.1623	0.1430	-0.0071	0.1507	-0.1848	-0.3712	-0.5324
0.2122	0.2115	0.0510	0.5056	0.8766	-0.0298	-0.6367	0.5564
0.0993	0.5539	-0.2823	0.2457	0.2619	0.0869	-0.2830	0.0597
0.1121	0.1053	0.4890	0.7366	0.0309	-0.9165	-0.3584	-0.1067
0.5265	0.3816	-0.2677	-0.3506	-0.3879	0.3715	-0.0207	0.1941
-0.2663	-0.0383	0.0879	-0.7713	0.4139	0.4444	0.5788	0.5788
0.3424	0.1303	-0.2154	0.1655	-0.3114	0.4338	-0.1410	-0.1963
0.3764	0.1508	-0.3086	0.3566	0.4840	0.3472	-0.2488	0.1941
-0.2664	-0.1227	0.0981	-0.4177	-0.2760	0.6847	0.3380	-0.2793

เอกสารนี้เป็นเอกสารทสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตเห็นาเบไซประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 (ต่อ) ค่าถ่วงน้ำหนักของชั้นฮิดเดน โหนด 1 ถึง 8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 168 ค่า

-0.0281	0.5346	-0.3148	0.2783	-0.0630	0.8919	-1.0372	0.4204
-0.0966	0.0312	0.6879	-0.3329	0.0002	0.4585	-0.3316	0.1670
-0.4211	0.3288	-0.8859	-0.0171	-0.2394	0.3334	0.1254	-0.4462
0.4713	0.3660	0.0630	-0.3593	0.0849	-0.2783	0.4427	-0.3333
0.1510	0.3616	-0.1019	0.0247	0.7133	0.3740	-0.3296	-0.5197
-0.0841	0.3951	-0.1121	-0.0137	0.2834	0.3194	-0.2407	-0.3525
0.1429	-0.0444	-0.6382	0.4453	0.2013	0.7143	-0.2022	-0.1721
0.4391	0.6216	-0.4142	-0.4382	-0.6716	-0.4808	-0.8167	-0.3120
-0.3340	-0.0854	-0.0184	0.3476	-0.3111	0.4828	0.1275	0.3057
0.0693	0.0740	0.0480	0.1843	0.1452	-0.3587	-0.0752	0.0274
-0.1854	-0.0151	0.0048	0.0586	-0.0624	0.5712	0.6971	0.0654
-0.2248	0.1203	0.4121	0.1586	-0.3563	0.0041	-0.3734	-0.1578
-0.1264	0.3736	-0.0730	0.1341	-0.3359	-0.4736	0.1933	0.0596
0.0208	0.0033	-0.4156	-0.6660	0.6251	-0.1086	-0.2047	0.1269
-0.2664	-0.1752	0.3770	0.2037	-0.0164	-0.3231	-0.1918	0.0645
-0.3331	-0.4666	0.8879	-0.0823	-0.2633	-0.2723	-0.4748	-0.4111
-0.0357	-0.1066	-0.1494	-0.4497	0.1665	0.3275	-0.0284	-0.1225
-0.4960	-0.2996	0.8611	0.6470	-0.2378	0.6120	0.1357	0.2341
-0.1901	-0.5095	0.3350	-0.2879	-0.0047	0.3440	-0.5261	-0.0021
0.4124	-0.1471	0.2744	-0.3510	-0.0971	0.5345	0.1753	-0.0674
-0.3020	-0.4412	-0.5349	-0.5469	0.3271	-0.5831	-0.1288	-0.5495
-0.5737	0.0949	-0.2740	0.0696	-0.0117	-0.3729	0.4031	-0.4527
-0.2592	-0.4300	0.5147	-0.0349	-0.1612	0.1103	0.5088	-0.0845
0.3538	0.1838	0.4295	-0.5727	-0.0127	0.2234	-0.3344	-0.5355
0.2874	0.4278	0.2692	0.2793	-0.2066	0.0008	-0.2600	0.1538
-0.3273	-0.1056	-0.5329	0.3828	-0.5323	0.5053	0.2324	-0.0169
-0.2609	-0.0870	-0.0500	0.0785	-0.0084	0.0351	0.1485	-0.0337
-0.5050	-0.1012	-0.4423	0.3456	0.3125	-0.6985	0.0519	0.3161
-0.1964	0.2358	-0.5685	0.1781	-0.1621	0.4840	-0.3574	0.0129
0.0506	-0.1296	0.2642	0.5734	0.1715	0.0757	0.2061	0.2489
-0.0761	0.1365	-0.3320	0.0030	0.3277	0.0038	0.2623	-0.2174
0.1477	0.0901	0.6805	-0.1656	-0.6443	0.3357	-0.4478	-0.1728

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่หรือใช้
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 (ต่อ) ค่าถ่วงน้ำหนักของชั้นฮิดเดน โหนด 1 ถึง 8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 168 ค่า

0.1937	0.1894	-0.1783	-0.1369	-0.2736	0.3221	-0.1520	0.2997
0.0104	0.3244	-0.2929	0.1434	0.4029	0.5583	-0.0356	0.4089
-0.0247	0.2105	-0.4833	0.0300	0.0824	-0.3012	0.0359	0.2353
-0.1502	0.0588	-0.2303	0.3069	-0.5124	0.1018	0.3055	0.2626
0.1913	-0.0174	0.3301	-0.5848	0.5094	-0.1544	0.4542	-0.1486
0.0258	-0.1095	-0.2536	0.2163	-0.0755	0.0699	-0.4414	-0.2033
-0.2150	0.1331	0.1380	-0.3534	0.3831	0.4135	0.4818	-0.5650
0.2128	0.2997	0.4569	-0.4210	0.3801	0.3900	-0.1377	-0.3457
-0.3280	0.2568	-0.1244	0.1433	-0.2007	-0.4881	-0.3269	0.3544
-0.3090	-0.4266	-0.2893	0.3643	-0.2334	0.0048	0.0359	0.4602

ตารางที่ 4.4 ค่าถ่วงน้ำหนักของชั้นฮิดเดน โหนด 1 ถึง 8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 240 ค่า

โหนด 1	โหนด 2	โหนด 3	โหนด 4	โหนด 5	โหนด 6	โหนด 7	โหนด 8
-0.0811	0.6990	0.1929	0.1480	0.5045	-0.2206	-0.0998	-0.5365
-0.2668	0.1468	0.2217	-0.3002	-0.2987	0.3937	-0.2312	-0.0855
-0.4338	-0.2283	0.0372	0.0625	-0.0785	0.3708	-0.3594	0.6330
0.3427	0.3496	-0.0061	-0.0915	-0.0242	0.3390	0.4036	-0.1620
-0.0900	-0.1364	0.6685	0.3396	0.0762	-0.1809	0.0332	0.1271
-0.3764	0.4869	-0.2507	0.1311	0.0354	-0.4173	-0.1883	-0.3379
0.1463	0.5229	0.1628	-0.1510	-0.3766	-0.4246	0.0141	0.4800
0.2279	0.3328	-0.2716	-0.4600	0.3290	0.2558	0.0096	0.1689
-0.1710	0.0108	0.1842	0.0604	0.0485	-0.0800	0.5744	-1.1234
-0.2604	-0.5211	0.7069	-0.2176	-0.4343	0.1364	-0.0720	0.5898
-0.1076	-0.2900	0.2241	0.4332	0.4093	-0.0353	-0.3046	0.1814
-0.3606	-0.4803	0.2463	0.4913	-0.3378	0.1664	-0.3544	0.1899
-0.1327	1.3446	-0.2658	0.9097	0.7373	-0.1554	-0.1917	-0.3232
0.7341	-0.4770	-0.2590	-0.2582	-0.0068	-0.0946	-0.0392	-0.2684
0.2882	0.0235	0.1629	0.5576	-0.2855	-0.4414	-0.2482	0.4431
0.0564	-0.5502	-0.4010	-0.2239	0.0222	0.3228	0.1328	-0.4965
-0.5877	0.7508	1.0258	0.1859	0.4679	0.3652	0.4650	-0.3471
0.3119	-0.0142	-0.4866	-0.4065	0.1790	0.3315	0.1680	0.0075

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนไว้ใช้เฉพาะในวงจำกัดเท่านั้น ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาตจากเจ้าของลิขสิทธิ์

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 (ต่อ) ค่าถ่วงน้ำหนักของชั้นฮิดเดนโหนด 1 ถึง 8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 240 ค่า

-0.4141	-0.4887	-0.4302	-0.0830	-0.3456	-0.3027	0.0029	-0.3152
-0.0163	-0.0685	-0.5429	-0.6770	-0.6354	-0.3506	0.1316	0.1766
-0.3632	0.0261	0.3114	0.0720	0.3739	0.0990	-0.4224	-0.1892
-0.4545	-0.2548	-0.3546	-0.3891	-0.3817	0.4913	0.2283	0.0085
-0.4794	-0.3736	-0.2028	0.5122	-0.0003	-0.5631	0.3415	-0.0437
0.0255	-0.1330	0.0725	-0.4227	0.1362	-0.5292	0.2117	-0.3015
-0.2424	0.4993	0.3933	0.3184	0.2090	0.1987	-0.3247	-0.3234
-0.5112	0.5269	-0.1110	0.5363	0.5227	-0.3822	-0.1761	0.5811
-0.2521	0.3885	-0.2806	0.2015	0.5829	-0.3776	-0.0274	-0.8234
0.5839	-0.7261	-0.0646	-0.0141	-0.3189	0.1092	-0.2585	0.2895
0.0978	-0.1863	-0.1347	0.3265	-0.0161	0.3945	-0.3179	-0.1151
0.1766	-0.5322	0.0240	0.0936	-0.1279	-0.2299	-0.4248	0.5200
-0.5277	-0.4785	0.2385	-0.0355	0.4685	-0.4610	-0.3514	0.3423
0.3472	0.1989	0.0155	-0.0831	-0.1201	-0.1482	0.4884	-0.0894
0.3549	0.2912	0.0372	-0.1202	0.1197	0.2890	0.1713	0.5764
0.1904	0.0313	0.0416	0.4608	-0.3907	0.1807	0.2476	0.2533
0.0190	-0.7685	0.1349	-0.3893	0.3417	-0.0984	0.0389	-0.2540
-0.6859	0.6299	-0.4829	-0.0123	0.1134	0.1312	-0.1213	0.1006
-0.0717	-0.0930	0.1786	0.3841	-0.1518	-0.3634	-0.2365	0.2497
-0.2568	0.1800	-0.4840	0.1024	0.4046	-0.1575	0.0634	0.6843
-0.2004	0.3757	-0.0309	0.4426	-0.3905	0.4272	0.4249	-0.2493
-0.1658	0.2633	-0.2347	-0.1624	-0.4542	-0.3124	0.3137	-0.1478
0.1307	0.1622	0.5836	-0.0206	0.8563	-0.6409	0.5553	-0.1862
0.0527	-0.3106	-0.4625	0.0474	-0.4997	0.6995	0.0130	-0.0370
-0.2574	0.0009	-0.0149	-0.2766	0.2902	0.4115	-0.0115	-0.2273
0.1736	-0.1041	0.2989	0.2485	-0.3487	0.3000	0.0077	-0.0442
-0.7656	0.4280	0.1165	0.6689	-0.1903	-0.4606	0.2063	-0.1888
-0.3927	-0.3382	-0.6992	-0.2135	0.3749	-0.4064	0.2428	-0.2774
0.2676	0.0671	0.4812	-0.4859	0.5565	-0.1714	0.3082	0.0556
0.4307	0.3538	0.8127	0.1063	-0.4320	-0.2937	0.4599	0.1902
0.4852	0.2630	-0.2987	0.6182	0.9894	0.0384	-0.0419	-0.3481
-0.0598	-0.0183	0.8545	-0.7217	-0.0492	-0.0547	-0.4695	0.5508

ตารางที่ 4.4 (ต่อ) ค่าถ่วงน้ำหนักของชั้นฮิดเดนโหนด 1 ถึง 8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 240 ค่า

-0.2004	0.3640	-0.2323	-0.2236	-0.4247	-0.2925	-0.2334	0.1906
0.1273	-0.4904	0.4360	0.3582	-0.5629	0.1987	-0.0197	-0.5171
0.5890	-0.1244	0.3490	-0.7282	-0.4330	-0.4107	0.5565	-0.7218
0.0849	0.0611	0.0314	0.6819	0.2206	0.2266	0.2755	0.0411
-0.0137	0.4700	-0.1080	-0.1968	-0.4602	-0.4431	0.4000	-0.2974
0.1063	-0.3869	-0.1732	0.5584	0.1174	0.2620	0.1811	0.2083
-0.2558	-0.4943	0.2087	0.3769	-0.3844	0.3358	-0.4387	0.3459
-0.2050	0.2705	-0.3990	0.1330	-0.0020	-0.1578	-0.3260	0.2346
-0.0005	-0.3003	-0.1422	-0.3135	0.5283	-0.2820	0.0336	-0.2528
0.4763	-0.4594	0.3807	0.0107	0.8677	0.5095	-0.4784	-0.2239
-0.4033	-0.1442	0.7384	-0.5196	-0.6254	0.0024	-0.1571	-0.5329
0.1588	0.2430	-0.4094	-0.4264	-0.5274	0.3664	0.1261	0.4414
-0.1524	-0.7186	0.5068	-0.1822	0.4201	0.2551	0.4331	0.1209
-0.2185	0.4896	0.1446	0.1608	0.1354	0.5568	0.4719	0.2924
-0.4196	-0.5297	0.3532	-0.3268	-0.2631	-0.6016	0.2322	-0.0099
0.8912	-0.2384	0.5552	-0.2786	-0.2666	0.0281	0.0770	0.2191
-0.2139	0.2730	-0.3750	0.0371	0.4962	0.0805	0.4438	-0.0469
0.3393	-0.3547	-0.0595	0.4439	-0.2988	0.2671	-0.4070	0.2467
0.3849	0.1134	-0.1430	-0.2206	-0.5788	0.4678	-0.2318	0.1461
0.2899	-0.1566	0.1653	-0.0731	0.0025	-0.3726	-0.4675	0.0704
-0.0797	0.1738	0.0835	0.4168	-0.2978	0.5153	-0.3493	0.1221
-0.4382	0.0894	-0.0522	-0.3195	0.0149	0.1970	0.4124	0.1699
-0.3538	-0.5726	0.2591	-0.4977	0.1846	0.1020	-0.4241	-0.3111
-0.1813	0.2227	-0.2065	-0.1997	0.1783	0.1811	-0.4291	-0.1256
-0.2558	-0.1286	-0.3078	-0.1601	-0.1367	-0.1401	0.1289	0.3846
0.2546	-0.3712	-0.6580	0.3044	0.0849	-0.0754	0.4075	-0.0494
-0.1884	-0.2108	-0.1898	-0.5098	0.3987	0.1888	0.2198	0.2117
-0.3793	0.4241	-0.0862	0.1145	-0.1672	-0.3386	0.2465	-0.3048
-0.3972	-0.3260	0.0568	0.3607	0.4153	0.0106	0.2766	-0.2162
-0.5241	0.4550	-0.4272	-0.0965	0.0051	0.5323	0.0678	-0.2272
0.4165	0.1414	0.8304	0.1171	-0.0463	-0.3726	0.0424	0.1736
0.5742	0.3135	-0.3473	-0.3078	0.0556	0.1155	-0.4068	0.0457

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 (ต่อ) ค่าถ่วงน้ำหนักของชั้นฮิดเดนโหนด 1 ถึง 8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 240 ค่า

0.2193	0.3634	0.3642	0.6587	0.0869	-0.3713	0.0108	0.2627
0.0417	-0.0233	-0.1149	0.0377	-0.2781	0.4181	-0.0701	-0.1230
-0.0670	-0.3858	0.5764	0.6905	-0.8524	0.5264	0.0096	0.3402
-0.5701	-0.0586	0.4301	-0.0555	-0.5260	-0.2602	0.0132	-0.3257
0.3578	0.4057	0.0394	-0.4568	0.4764	0.3613	0.1529	-0.4623
0.5295	-0.4326	0.5170	-0.3847	-0.1078	-0.0157	-0.2675	0.1959
0.0523	-0.2241	0.5199	-0.1849	-0.1804	-0.2971	-0.0426	-0.2635
0.1834	-0.0906	0.5827	-0.0993	-0.5804	0.2021	-0.3713	-0.2181
0.5897	-0.3817	-0.5766	0.3434	0.0166	-0.3371	-0.4638	-0.0070
-0.3178	0.3565	-0.2351	-0.4738	0.2232	-0.3429	-0.2988	0.2796
0.1291	0.4306	-0.6757	-0.3200	-0.2199	-0.0481	0.4142	-0.4908
-0.3768	0.4377	-0.3132	-0.0108	-0.3038	0.0536	0.1433	0.2414
-0.1607	0.2399	-0.4698	0.3338	-0.2905	0.3637	-0.3097	0.0481
-0.0109	-0.6282	0.4985	0.3513	-0.4989	0.0710	0.1553	-0.1270
0.2157	-0.4044	0.6174	0.3771	-0.5291	-0.2868	-0.0803	0.1443
0.2199	-0.0166	-0.1397	-0.0869	0.3749	0.2124	0.1090	0.3424
0.4195	-0.3103	0.0490	-0.2115	-0.2402	0.3019	0.2949	-0.0197
0.4671	0.3252	0.3333	0.8202	-0.3664	0.5287	0.0377	0.0746
-0.4888	-0.7725	0.1175	0.6571	0.3964	-0.4458	0.1836	0.4965
0.0298	-0.1849	0.0133	-0.1661	0.4441	-0.4273	0.0155	-0.1942
-0.0317	0.5522	-0.0802	0.4494	0.2351	-0.2750	0.2600	0.1160
0.0215	-0.1978	-0.3421	-0.0922	-0.4064	-0.2283	0.0444	0.3054
-0.3271	0.1723	-0.0401	0.0351	0.0392	0.4159	-0.3707	-0.3162
0.0775	-0.4069	-0.0883	-0.3019	-0.0743	0.3235	-0.1033	0.0502
-0.2511	-0.0368	-0.4737	-0.0212	-0.1097	-0.1711	0.4209	-0.1266
-0.0310	-0.4336	-0.0000	0.1554	0.1004	0.3176	0.1068	0.4675
-0.2089	0.2497	0.2553	0.2949	0.1846	0.4643	0.1331	0.2041
0.1754	-0.5467	0.2352	0.4195	0.5284	-0.2849	-0.1952	0.3121
0.0978	0.4447	-0.5812	-0.6429	0.0366	-0.6904	-0.3037	-0.4753
0.0759	-0.4779	-0.3008	-0.2710	0.4423	0.1824	-0.3767	-0.3223
-0.5628	-0.1744	-0.0695	0.3149	0.2695	0.0174	-0.4963	0.4216
-0.1756	0.3292	0.2164	-0.1551	0.1811	0.3232	0.1229	0.3967

ตารางที่ 4.4 (ต่อ) ค่าถ่วงน้ำหนักของชั้นฮิดเดน โหนด 1 ถึง 8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 240 ค่า

0.0679	0.1350	0.5447	-0.3029	0.3737	-0.4422	0.3162	-0.4729
-0.1958	-0.1816	-0.3278	-0.2289	-0.1188	-0.3054	0.0729	0.1216
-0.3251	0.0523	-0.2431	-0.4761	0.3728	-0.3169	0.1865	-0.3379
-0.4494	-0.4651	-0.0589	-0.3653	0.4781	0.2255	-0.0141	-0.4890
-0.4361	0.4165	-0.5599	-0.3417	-0.1302	-0.4923	-0.3570	0.2518
0.4874	0.2543	0.1460	0.2510	0.2490	0.0947	0.2760	-0.0879
0.3389	-0.6525	0.1280	-0.4837	-0.0446	0.3164	0.4553	0.5248
0.2797	0.0238	-0.7598	0.3685	-0.3716	0.5928	-0.2866	-0.2813
-0.1325	0.0387	-0.0867	0.0101	0.0561	-0.3812	0.3912	-0.4509
-0.1561	-0.3309	-0.4437	-0.2731	-0.0036	-0.1380	0.2032	-0.0650
0.2792	0.6014	-0.0314	-0.5224	-0.0681	0.3210	0.2755	-0.3153
0.1021	0.3639	-0.3822	-0.2027	-0.1788	0.1187	-0.4033	0.2003
0.5068	-0.1189	0.0630	-0.0731	-0.1777	-0.1875	-0.2389	-0.0304
-0.0221	-0.1127	0.5058	-0.2933	0.3672	0.0154	0.3554	0.2622
0.0869	-0.2621	0.3028	0.1144	0.1184	0.3325	0.1735	-0.3562
0.4746	-0.0934	0.2144	0.1475	-0.1580	0.5477	-0.2229	-0.7879
0.4358	-0.5015	-0.7008	-0.1386	0.2741	0.2667	-0.2645	0.4193
-0.3411	0.3790	0.0423	-0.2134	-0.1598	0.2578	0.0329	-0.0972
0.3013	0.0502	-0.4693	-0.1259	0.1143	0.0516	0.3983	0.5600
-0.7004	0.2550	-0.6509	-0.2049	-0.1984	-0.1379	0.2326	0.6436
0.2648	0.4741	0.3087	-0.3544	-0.4199	-0.5495	0.1814	0.0467
0.1363	-0.2618	0.6016	-0.2734	-0.4359	0.0339	0.2669	0.6011
0.1762	-0.0070	0.3429	0.6556	0.0436	-0.2465	-0.0116	-0.0172
0.4372	0.1067	-0.4080	0.1262	0.3654	0.3294	-0.1215	-0.1606
0.1082	0.0493	-0.5420	0.7228	0.4301	0.2474	-0.1882	-0.1510
-0.6969	0.2612	0.3881	0.1957	-0.2609	0.5330	-0.0804	0.6735
0.0441	0.4492	0.1266	0.2363	-0.3016	0.0279	0.1582	-0.3388
0.4680	-0.2377	0.2914	0.4633	-0.2469	0.1441	-0.2222	0.0314
-0.3759	-0.6410	0.3632	-0.1246	0.0657	0.4709	0.2348	0.1023
0.2902	-0.0318	0.0803	-0.1975	-0.4829	0.4216	0.0206	0.3990
0.4036	-0.4015	0.2272	0.1927	0.3146	0.3942	-0.2438	-0.4720
0.0196	-0.1733	-0.2649	0.2985	0.4103	0.0555	0.3876	-0.5070

ตารางที่ 4.4 (ต่อ) ค่าถ่วงน้ำหนักของชั้นฮิดเดนโหนด 1 ถึง 8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 240 ค่า

-0.3229	-0.4396	-0.3414	-0.1798	0.3930	0.1624	-0.3107	0.3760
0.0835	0.3389	0.3461	-0.2126	0.4325	-0.0897	-0.4415	-0.4544
0.1424	-0.5209	0.0262	0.1737	-0.1851	0.1609	-0.2125	0.5030
-0.3941	-0.2836	0.3207	0.0324	0.1586	0.1926	-0.2966	0.2864
-0.2338	0.1172	-0.1886	-0.0674	0.2154	-0.4562	-0.4345	0.4982
0.1676	0.1172	-0.0481	-0.2382	-0.2685	-0.0371	-0.4841	0.1950
0.2999	-0.0593	-0.1826	0.0831	0.0313	-0.0444	-0.1051	0.1420
-0.5910	-0.1308	-0.6094	-0.0691	-0.5777	0.2612	-0.1599	0.1537
-0.1203	-0.4415	-0.1844	0.1832	0.0980	-0.0119	0.4363	-0.0928
0.4909	-0.3020	-0.0163	-0.3131	-0.3964	-0.1057	-0.2206	-0.3323
0.0850	-0.4300	0.0782	-0.0305	0.0501	-0.0285	-0.3830	-0.0546
0.3561	-0.1343	-0.0873	-0.1631	0.3742	-0.2073	0.0263	-0.0939
-0.4500	-0.4041	0.6052	-0.1459	-0.0426	0.3338	-0.1589	0.2025
-0.0376	-0.4770	0.1671	0.0648	0.1962	-0.4393	0.2880	-0.2924
-0.0368	-0.1642	-0.2520	0.1203	-0.3260	0.6236	-0.3483	0.4527
-0.1164	-0.4505	-0.6445	0.3761	-0.2693	0.2017	-0.2070	-0.0965
0.3997	-0.3479	-0.4230	0.1256	-0.2109	-0.6344	-0.1273	-0.1671
-0.2703	-0.0698	-0.4013	-0.1396	0.1025	-0.2431	-0.3356	0.2561
1.0333	0.6457	0.2929	0.2473	-0.2454	-0.2543	-0.0738	0.3394
-0.2435	-0.3941	-0.2399	0.2181	0.3388	-0.3479	-0.3537	-0.3077
-0.0193	-0.7938	-0.1436	-0.2713	0.4420	-0.2290	0.4216	0.5842
0.0250	-0.2415	0.5304	0.3921	0.5517	-0.4284	0.2738	0.3042
0.2176	-0.4450	-0.5574	1.0826	0.1109	0.6375	-0.0852	-0.0403
0.1368	-0.1978	-0.1110	-0.3795	-0.2026	-0.5465	-0.3036	-0.3370
0.4280	0.1070	-0.2451	-0.1469	-0.2688	0.2532	-0.1940	0.3466
0.4175	0.1049	0.4343	-0.4009	-0.2272	0.6096	-0.4859	-0.3449
0.1679	-0.3433	-0.1905	-0.6488	-0.0290	0.0494	-0.1656	0.9980
0.2295	0.3908	-0.0869	0.5355	-0.2937	-0.5002	-0.3879	0.0208
-0.3089	-0.3759	-0.6492	0.1917	0.0263	-0.2823	0.4495	-0.4287
0.1392	0.1861	0.8523	0.5713	-0.2928	-0.1615	0.3039	0.7335
-0.1756	-0.8266	-0.5963	-0.3911	-0.0238	0.1220	0.0662	0.3198
0.2326	-0.1645	-0.1434	-0.3172	-0.3113	0.1372	0.2372	-0.4824

ตารางที่ 4.4 (ต่อ) ค่าถ่วงน้ำหนักของชั้นฮิดเดน โหนด 1 ถึง 8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 240 ค่า

0.0489	-0.0511	0.2447	-0.3478	-0.2568	-0.2551	-0.4045	-0.3707
0.4158	0.0755	-0.3008	0.3216	0.5025	-0.5094	-0.0479	-0.4413
0.4393	0.1902	-0.0177	-0.3383	-0.5719	-0.1958	-0.2595	-0.4935
0.5481	-0.1037	0.2729	0.1670	0.2332	-0.0719	0.3374	0.0036
-0.3654	-0.3579	0.3461	-0.2283	0.5044	-0.1691	0.4542	0.3851
-0.1916	-0.2600	0.7059	0.4848	-0.1486	-0.0679	-0.2910	-0.2960
-0.2436	0.2435	-0.2528	0.3364	-0.2232	-0.0467	-0.4437	-0.0736
-0.2909	-0.0678	0.4145	-0.0806	-0.0701	-0.1548	-0.0095	0.3034
-0.1797	0.1323	0.2620	0.0597	0.1786	-0.0047	-0.0874	0.5179
-0.4462	-0.3049	0.2414	-0.2129	-0.1062	0.2718	0.4773	0.4638
-0.0252	-0.2335	0.3988	0.1327	0.1836	0.2879	-0.2875	-0.4679
0.3431	0.2197	0.4476	0.2160	-0.3713	0.4867	-0.1504	0.0599
-0.1182	-0.0435	0.0348	-0.0149	0.5622	0.3883	-0.2913	-0.0699
0.1608	-0.2918	0.3621	-0.5912	0.0466	-0.4590	0.4793	0.1673
0.3825	0.0085	0.0794	0.4682	-0.2236	-0.2184	0.3715	0.1821
0.0830	0.1892	-0.1788	-0.0960	-0.2329	0.4246	-0.2057	0.2932
0.2836	-0.5128	0.1652	0.4320	-0.1428	-0.4006	0.3546	-0.4740
-0.4335	0.3844	-0.0839	0.1892	-0.2175	0.1367	-0.4212	0.1359
0.2183	0.2987	-0.2060	0.0944	0.2748	-0.0462	-0.0630	0.0395
0.3081	-0.0621	-0.3759	-0.2754	0.4768	0.4787	0.4942	-0.1200
-0.3959	0.0999	0.3119	0.1466	0.3070	0.2840	0.2585	-0.1926
-0.0985	-0.0920	0.5038	-0.1998	-0.3867	0.2258	0.4227	0.0372
-0.3565	0.1076	0.4252	-0.2501	-0.4092	-0.1272	0.4139	-0.4165
0.6278	-0.3842	-0.0187	0.7475	0.0238	0.1650	0.3177	-0.7190
-0.1268	0.2087	0.2390	0.1463	-0.0622	0.2884	-0.3988	-0.3174
0.4280	-0.5161	0.0396	0.2231	0.2828	0.4194	0.0237	-0.0481
-0.3368	0.6047	-0.8161	-0.2951	0.1013	-0.6090	-0.1041	0.5325
-0.0734	-0.0909	0.2821	-0.1869	0.1219	-0.4815	0.1532	-0.2722
-0.4134	0.2836	0.5558	0.4051	0.0755	-0.2670	-0.4247	-0.2117
0.0146	-0.1510	0.5198	0.4003	-0.3145	0.5159	0.2315	0.3758
-1.0083	-0.8906	-0.3434	-0.2992	0.8394	0.1852	-0.2507	-0.0362
0.1047	-0.5146	0.4379	-0.5599	0.7061	-0.3256	0.4587	0.1708

ตารางที่ 4.4 (ต่อ) ค่าถ่วงน้ำหนักของชั้นฮิดเดนโหนด 1 ถึง 8 โดยแต่ละโหนดมีค่าถ่วงน้ำหนัก 240 ค่า

0.2650	-0.1657	0.2181	-0.1411	-0.6484	-0.3464	0.4664	0.1493
-0.4609	0.2860	0.0957	0.4635	0.2604	0.0985	0.0131	0.0655
-0.1517	-0.4459	0.3368	-0.4839	0.0485	-0.0670	0.4247	0.1594
-0.6387	-0.1265	-0.4332	-0.1894	0.5340	0.1509	0.1696	-0.0927
0.3802	0.2748	-0.4486	0.1355	0.0505	-0.1704	0.0271	-0.3621
0.0036	0.2499	-0.1911	0.4348	0.0718	-0.1295	0.0373	0.4772
-0.0146	-0.2578	-0.1131	-0.0989	0.5890	-0.2904	0.2381	0.0721
0.2875	0.3556	0.2465	-0.2471	0.0070	0.3056	-0.2041	-0.3610
-0.2999	-0.1193	-0.0697	0.1030	0.0417	-0.0837	0.2829	0.0735
0.5691	-0.2551	-0.4170	0.1947	-0.4077	0.2282	-0.0894	0.4830
0.1919	0.0063	0.2886	0.4633	-0.3035	0.4531	0.0665	-0.2719
-0.1246	0.1078	0.1140	0.1676	-0.4373	0.2846	-0.1254	0.0972
0.5458	-0.6191	-0.2037	-0.1094	-0.3140	-0.0631	-0.0955	-0.2444
0.1909	0.3866	-0.2824	-0.5326	-0.0388	0.1147	-0.3800	0.0958
-0.3042	0.0097	-0.2343	0.2017	0.6000	-0.2457	0.3655	0.1511
-0.4000	-0.5857	-0.2859	-0.2789	0.3620	0.3623	0.5175	0.1065
-0.4811	0.1777	-0.2044	0.3564	-0.1874	-0.0609	0.0858	-0.3979
0.2813	0.1706	0.1572	-0.1491	-0.4712	0.3225	-0.1796	0.1647
0.1350	0.3613	0.5852	0.4601	-0.2256	-0.2850	-0.1919	-0.0196
-0.4125	-0.2786	0.1453	-0.1084	0.2372	-0.2931	-0.0522	0.1702
0.2357	-0.2873	0.1782	-0.2165	0.0748	-0.4501	-0.1109	0.1293
0.3554	0.1914	-0.1146	-0.4686	-0.0378	-0.3455	-0.0601	-0.1229
-0.3419	-0.4070	0.3390	0.3066	-0.3547	0.1346	-0.3594	-0.0953
0.1666	-0.2780	-0.3492	0.3321	-0.3393	0.2383	-0.1622	-0.0374
-0.4540	-0.0996	-0.1217	0.4320	-0.2336	0.0034	-0.2422	0.0282
0.5031	0.1605	-0.0808	0.2258	-0.5667	0.2475	-0.5003	-0.0768
0.3039	-0.2360	-0.3061	-0.3215	-0.3772	0.1775	-0.2087	-0.5495
0.3898	-0.2302	0.3783	0.3523	0.0807	0.4097	-0.1916	0.0871
-0.2409	-0.0828	-0.1068	0.1580	0.3429	0.4502	0.2221	0.1218
0.2208	-0.4139	0.0708	0.0778	0.1567	-0.4681	-0.2266	-0.2824

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. หลังจากการฝึกสอนโครงข่ายประสาทเทียมให้รู้จักตัวอักษรได้สำเร็จ จะเข้าสู่การใช้งานในการแปลงข้อมูลคุณลักษณะเฉพาะของภาพให้เป็นข้อมูลตัวอักษร ในตารางที่ 4.5 จะแสดงค่าเอาต์พุตที่โหนดต่างๆ ของ Output layer ที่ได้จากโครงข่าย เมื่อทำการแปลงข้อมูลป้ายทะเบียน รุ่น 7264 โดยอินพุตเป็นข้อมูลภาพที่ทำการแปลงดิสคริตโคไซน์ด้วย DCT เมตริกซ์ขนาด 10×10 ในตารางที่ 4.4 จะแสดงค่าเอาต์พุตที่โหนดต่างๆ ของ Output layer ที่ได้จากโครงข่าย เมื่อทำการแปลงข้อมูลป้ายทะเบียน พม 6333 โดยอินพุตเป็นข้อมูลภาพที่ทำการแปลงดิสคริตโคไซน์ด้วย DCT เมตริกซ์ขนาด 5×5 และในตารางที่ 4.6 แสดงผลของการรู้จำป้ายทะเบียนต่างๆ

ตารางที่ 4.5 ค่าเอาต์พุตของการประมวลผลของโครงข่ายกับป้ายทะเบียน รุ่น 7264

ทะเบียน โหนด	ฐ	น	7	2	6	4
1	0.0000	0.0003	0.0000	0.0010	0.0006	0.0000
2	0.0000	0.0002	0.0022	0.0001	0.0000	0.0001
3	0.0001	0.0000	0.0000	0.0000	0.0006	0.0009
4	0.0003	0.0020	0.0009	0.0007	0.0000	0.0013
5	0.0001	0.0001	0.0009	0.0009	0.0000	0.0000
6	0.0146	0.0415	0.0033	0.0000	0.0000	0.0014
7	0.0001	0.0002	0.0000	0.0012	0.0000	0.0089
8	0.0131	0.0005	0.0000	0.0000	0.0010	0.0000
9	0.3533	0.0000	0.0010	0.0002	0.0000	0.0008
10	0.0001	0.0000	0.0015	0.0000	0.0004	0.0005
11	0.0000	0.0004	0.0005	0.0000	0.0000	0.0004
12	0.0000	0.0000	0.0002	0.0010	0.0001	0.0003
13	0.0001	0.0000	0.0000	0.0005	0.0002	0.0004
14	0.0001	0.1674	0.0000	0.0000	0.0010	0.0000
15	0.0024	0.0001	0.0046	0.0001	0.0003	0.0032
16	0.0013	0.7498	0.0001	0.0002	0.0000	0.0001
17	0.0011	0.0001	0.0012	0.0001	0.0000	0.0000
18	0.0001	0.0011	0.0000	0.0002	0.0000	0.0000
19	0.0000	0.0024	0.0021	0.0167	0.0000	0.0005
20	0.0000	0.0035	0.0015	0.0013	0.0005	0.0001
21	0.0025	0.0936	0.0000	0.0000	0.0003	0.0002

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.5 (ต่อ) ค่าเอาต์พุตของการประมวลผลของโครงข่ายกับป้ายทะเบียน รุ่น 7264

22	0.0002	0.0091	0.0000	0.0000	0.0004	0.0000
23	0.0001	0.0000	0.0040	0.0016	0.0000	0.0018
24	0.0000	0.0495	0.0000	0.0000	0.0000	0.0005
25	0.0003	0.0004	0.0012	0.0021	0.0002	0.0000
26	0.0127	0.0000	0.0006	0.0000	0.0004	0.0035
27	0.0000	0.0003	0.0000	0.0058	0.0000	0.0021
28	0.0001	0.0008	0.0001	0.0000	0.0000	0.0000
29	0.0000	0.0000	0.0001	0.0004	0.0024	0.0000
30	0.0029	0.1225	0.0000	0.0027	0.0000	0.0004
31	0.0000	0.0002	0.0001	0.0007	0.0003	0.0078
32	0.0011	0.0000	0.0000	0.0000	0.0000	0.0042
33	0.0000	0.0998	0.0000	0.0018	0.0002	0.0029
34	0.0002	0.0000	0.0017	0.0002	0.0004	0.0001
35	0.0001	0.0001	0.0062	0.0029	0.0000	0.0040
36	0.0001	0.0005	0.0000	0.0003	0.0054	0.0005
37	0.0015	0.0002	0.0029	0.0034	0.0011	0.0106
38	0.0000	0.0000	0.0018	0.9962	0.0000	0.0012
39	0.0000	0.0002	0.0008	0.0166	0.0017	0.0001
40	0.0001	0.0001	0.0020	0.0028	0.0002	0.9841
41	0.0000	0.0000	0.0020	0.0043	0.0026	0.0011
42	0.0000	0.0000	0.0000	0.0000	0.9507	0.0000
43	0.0001	0.0000	0.9871	0.0028	0.0003	0.0002
44	0.0004	0.0000	0.0015	0.0015	0.0029	0.0005
45	0.0000	0.0000	0.0000	0.0060	0.0009	0.0000

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.6 ค่าเอาต์พุตของการประมวลผลของโครงข่ายกับป้ายทะเบียน พม 6333

ทะเบียน โทนคร	พ	ม	6	3	3	3
1	0.0000	0.0000	0.0017	0.0008	0.0009	0.0014
2	0.0008	0.0022	0.0000	0.0000	0.0000	0.0000
3	0.0000	0.0011	0.0028	0.0001	0.0001	0.0001
4	0.0010	0.0000	0.0000	0.0017	0.0014	0.0007
5	0.0034	0.0010	0.0000	0.0001	0.0001	0.0001
6	0.0000	0.0004	0.0000	0.0002	0.0002	0.0009
7	0.0006	0.0001	0.0001	0.0001	0.0001	0.0002
8	0.0001	0.0002	0.0004	0.0002	0.0002	0.0001
9	0.0000	0.0003	0.0009	0.0106	0.0091	0.0012
10	0.0000	0.0008	0.0010	0.0003	0.0002	0.0000
11	0.0000	0.0014	0.0005	0.0000	0.0000	0.0001
12	0.0003	0.0000	0.0054	0.0001	0.0001	0.0000
13	0.0001	0.0002	0.0000	0.0000	0.0000	0.0000
14	0.0075	0.0000	0.0001	0.0002	0.0002	0.0000
15	0.0000	0.0000	0.0018	0.0088	0.0089	0.0044
16	0.0006	0.0024	0.0000	0.0001	0.0001	0.0005
17	0.0050	0.0003	0.0001	0.0000	0.0000	0.0001
18	0.0000	0.0015	0.0000	0.0003	0.0003	0.0004
19	0.0014	0.0038	0.0000	0.0002	0.0002	0.0002
20	0.0000	0.0007	0.0016	0.0000	0.0000	0.0000
21	0.2182	0.0059	0.0000	0.0000	0.0000	0.0000
22	0.2101	0.0001	0.0000	0.0000	0.0000	0.0001
23	0.0005	0.0022	0.0000	0.0000	0.0000	0.0000
24	0.0007	0.9876	0.0000	0.0000	0.0000	0.0000
25	0.0018	0.0059	0.0000	0.0002	0.0002	0.0001
26	0.0000	0.0003	0.0000	0.0003	0.0003	0.0003
27	0.0003	0.0001	0.0027	0.0000	0.0000	0.0000
28	0.0000	0.0000	0.0000	0.0002	0.0002	0.0001
29	0.0001	0.0001	0.0003	0.0015	0.0014	0.0016

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.6 (ต่อ) ค่าเอาต์พุตของการประมวลผลของโครงข่ายกับป้ายทะเบียน พม 6333

30	0.0006	0.0007	0.0002	0.0002	0.0002	0.0002
31	0.0016	0.0053	0.0001	0.0001	0.0001	0.0002
32	0.0002	0.0014	0.0001	0.0000	0.0000	0.0000
33	0.0004	0.0045	0.0000	0.0000	0.0000	0.0000
34	0.0000	0.0017	0.0001	0.0008	0.0008	0.0013
35	0.0000	0.0043	0.0000	0.0008	0.0008	0.0012
36	0.0000	0.0000	0.0047	0.0001	0.0001	0.0005
37	0.0081	0.0015	0.0001	0.0001	0.0001	0.0000
38	0.0001	0.0000	0.0002	0.0381	0.0347	0.0052
39	0.0001	0.0000	0.0004	0.9518	0.9540	0.9821
40	0.0001	0.0000	0.0002	0.0003	0.0003	0.0007
41	0.0002	0.0052	0.0027	0.0007	0.0007	0.0015
42	0.0000	0.0000	0.9940	0.0014	0.0014	0.0006
43	0.0001	0.0003	0.0000	0.0001	0.0001	0.0000
44	0.0001	0.0000	0.0049	0.0180	0.0182	0.0094
45	0.0001	0.0000	0.0007	0.0003	0.0003	0.0001

ตารางที่ 4.7 ผลของการจดจำป้ายทะเบียนจากการแปลงคิสตรีคโตชานน์

ป้ายทะเบียน	ผลของการจดจำของประสาทเทียม จากการแปลงด้วย DCT เมตริกซ์ ขนาด 5×5 (ไม่เว้นวรรค)	ผลของการจดจำของประสาทเทียม จากการแปลงด้วย DCT เมตริกซ์ ขนาด 10×10 (ไม่เว้นวรรค)
กค 8405	กค8405	กค8405
กง 7056	กง7056	กง7056
จฝ 6806	จฝ6806	จฝ6806
ชข 2581	ชข2581	ชข2581
ชผ 6908	ชผ6908	ชผ6908
ชพ 3478	ชพ3473	ชพ3478
ณย 9199	ณย9199	ณย9199
ตช 4213	ตช4213	ตช4213
ถอ 9826	ถอ9826	ถอ9826

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.7 (ต่อ) ผลของการจดจำป้ายทะเบียนจากการแปลงคิศจริตโคชาชน

นง 3027	นง3027	นง3027
บท 5732	บท5732	บท5732
ปร 6018	ปร6018	ปร6018
พม 6333	พม6333	พม6333
ฟศ 9337	ฟศ9337	ฟศ9337
ลก 9644	ลก9644	ลก9644
วต 8442	วต8442	วต8442
วธ 8888	วธ8888	วธ8888
วท 2109	วท2109	วท2109
ศศ 9837	ศศ9837	ศศ9837
ขฐ 805	ขฐ805	ขฐ805
ศฉ 8948	ศฉ8948	ศฉ8948
ศฎ 9432	ศฎ9432	ศฎ9432
สพ 5630	สพ5630	สพ5630
สอ 9026	สอ9026	สอ9026
2ฐ - 6069	2ฐ6069	2ฐ6069
2ท - 0035	2ท0035	2ท0035
3ธ - 3307	3ธ3307	3ธ3307
3พ - 6284	3พ6284	3พ6284
4พ - 9371	4พ9371	4พ9371
5ศ - 3607	5ศ3607	5ศ3607
6ฎ - 4715	6ฎ4715	6ฎ4715
8พ - 5335	8พ5335	8พ5335
8ธ - 2029	8ธ2029	8ธ2029
9ข - 6965	9ข6965	<u>9ข6985</u>
9ฒ - 8665	9ฒ8665	<u>3ฒ8665</u>
ก - 4209	<u>ก14209</u>	<u>ก14209</u>
ข - 4401	<u>ข14401</u>	<u>ข14401</u>
น - 6742	<u>น16742</u>	<u>น46742</u>
41 - 4361	414361	414361
77 - 6255	776255	776255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางที่ 4.7 เป็นผลของความสามารถในการจดจำรูปแบบตัวอักษรป้ายทะเบียนของโครงข่ายประสาทเทียม โดยทำการทดสอบกับป้ายทะเบียนจำนวน 40 ป้ายทะเบียน ผลของการรู้จำป้ายทะเบียนเมื่อทำการแปลงคิสกริตโคซายน์ด้วย DCT เมตริกซ์ขนาด 5×5 มีความถูกต้อง 23 ป้ายทะเบียน ซึ่งคิดเป็นเปอร์เซ็นต์ความถูกต้อง $\frac{37}{40} \times 100 = 92.50$ เปอร์เซ็นต์ ส่วนผลของการรู้จำป้ายทะเบียนเมื่อทำการแปลงคิสกริตโคซายน์ด้วย DCT เมตริกซ์ขนาด 10×10 มีความถูกต้อง 28 ป้ายทะเบียน ซึ่งคิดเป็นเปอร์เซ็นต์ความถูกต้อง $\frac{35}{40} \times 100 = 87.50$ เปอร์เซ็นต์

4.4 การอ้างอิงป้ายทะเบียนกับระบบฐานข้อมูล (Reference to Database)

หลังจากผ่านกระบวนการรู้จำด้วยโครงข่ายประสาทเทียม และได้ข้อมูลป้ายทะเบียนที่เป็นข้อมูลตัวอักษร จะนำข้อมูลดังกล่าวไปอ้างอิงกับฐานข้อมูล โดยนำไปค้นหาข้อมูลอื่นๆ ของยานพาหนะ ผลของการรันโปรแกรมที่ทำงานตั้งแต่การรับข้อมูลที่เป็นภาพยานพาหนะพร้อมทะเบียนและมีเอาต์พุตออกมาเป็นอัตราค่าผ่านทางเป็นดังนี้

Transport Classification and Management System.
Enter transport image filename : Sample01.jpg

License-plate : รุน7264
Province : กรุงเทพมหานคร
Owner : นายอุดม วิทยา
Wheel : 4
Color : ดำ
Brand : FORD
Expressway FEE : 40

จากผลข้างต้นเพิ่มข้อมูล Sample01.jpg คือ ข้อมูลภาพดังรูปที่ 4.1(ก) ซึ่งมีการจำลองข้อมูลต่างๆ ขึ้น โดยในตัวโปรแกรมเมื่อผ่านกระบวนการรู้จำป้ายทะเบียนแล้ว จะนำข้อมูลที่ได้ไปทำการค้นหาข้อมูลอื่นๆ ในฐานข้อมูล ซึ่งใช้คำสั่ง SQL ดังนี้

```
mysql> select * from data where license="รุน7264";
```

ซึ่งจะทำให้ได้ข้อมูลทั้งหมดของยานพาหนะทะเบียน รุน 7264 และในส่วนของ Expressway FEE เป็นการกำหนดอัตราค่าผ่านทางขึ้นมา ซึ่งในที่นี้จะให้ยานพาหนะที่มีจำนวนล้อ 4 ล้อ ต้องเสียค่าผ่านทางเป็นเงิน 40 บาท

4.5 ผลความผิดพลาดที่เกิดจากข้อจำกัดของระบบ

ความผิดพลาดของระบบในโครงการนี้สามารถแบ่งออกเป็น 2 ส่วนใหญ่ คือ ความผิดพลาดในการหาตำแหน่งป้ายทะเบียนและความผิดพลาดในการแยกตัวอักษรก่อนที่จะนำเข้าสู่การรู้จำด้วยโครงข่ายประสาทเทียม ซึ่งเกิดจากข้อจำกัดของระบบ

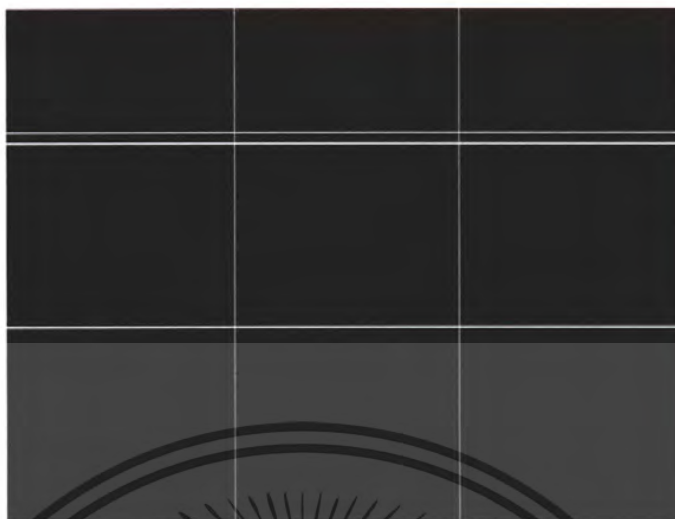
1. ความผิดพลาดในการหาตำแหน่งป้ายทะเบียน เกิดจากป้ายทะเบียนเอียง ขอบของป้ายทะเบียนมีสีกลมกลืนกับตัวยานพาหนะ หรือเกิดจากการถ่ายภาพในสภาพแวดล้อมที่มีแสงจ้ามากเกินไป หลังจากผ่านการหาขอบภาพแล้วจะไม่สามารถหาตำแหน่งของป้ายทะเบียนได้ เนื่องจากไม่สามารถหาขอบของป้ายทะเบียนได้หรืออาจได้ตำแหน่งป้ายทะเบียนที่ผิด หลังจากผ่านกระบวนการแปลงฮัฟ เป็นผลทำให้ไม่สามารถประมวลผลในขั้นตอนต่อไปได้



รูปที่ 4.15 ขอบของป้ายทะเบียนที่เอียงและมีสีกลมกลืนกับตัวยานพาหนะ



เอกสารนี้เป็นเอกสารที่รูปที่ 4.16 ภาพหลังจากผ่านกระบวนการหาขอบภาพของยานพาหนะรูปที่ 4.15
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

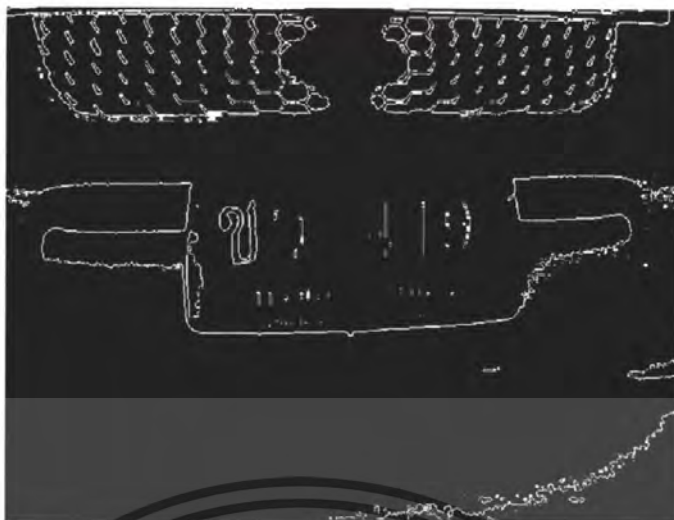


รูปที่ 4.17 หลังจากผ่านการแปลงอีฟจากรูปที่ 4.15 แล้ว ไม่สามารถระบุตำแหน่งขอบบนและขอบล่างของป้ายทะเบียนได้



รูปที่ 4.18 ป้ายทะเบียนที่เอียงและถ่ายภาพในสภาพแวดล้อมที่มีแสงจ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.19 ภาพหลังจากผ่านกระบวนการหาขอบภาพของยานพาหนะรูปที่ 4.18 และไม่สามารถหาตำแหน่งป้ายทะเบียนได้

2. ความผิดพลาดในการแยกตัวอักษร จะต่อเนื่องจากความผิดพลาดในการหาตำแหน่งป้ายทะเบียนและเกิดจากตัวอักษรบนป้ายทะเบียนจาง หลังจากผ่านการประมวลผลภาพและทำการหาตำแหน่งป้ายทะเบียนแล้วจะทำให้ไม่สามารถแยกตัวอักษรออกจากป้ายทะเบียนได้ และป้ายทะเบียนที่มีน็อตหรือมีคราบโคลนติดอยู่ จะทำให้ได้ตัวอักษรมากกว่าหนึ่งตัวในแพทเทิร์นตัวอักษรตัวหลังจากทำการแยกแล้ว เนื่องจากน็อตหรือคราบโคลนที่ติดระหว่างตัวอักษรจะทำให้ระบบตัดสินใจว่าเป็นตัวอักษรตัวเดียวกันหรือทำให้ตัวอักษรเพี้ยน เป็นผลให้การรู้จำป้ายทะเบียนเกิดความผิดพลาดด้วย



รูปที่ 4.20 ยานพาหนะที่ตัวอักษรบนป้ายทะเบียนจาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.21 ป้ายทะเบียนที่ไม่สามารถแยกตัวอักษรได้ หลังจากผ่านกระบวนการประมวลผลภาพและหาตำแหน่งป้ายทะเบียนของรูปที่ 4.20 เนื่องจากตัวอักษรบนป้ายทะเบียนจาง



รูปที่ 4.22 ป้ายทะเบียนที่มีน็อด

รูปที่ 4.23 หลังจากผ่านกระบวนการแยกตัวอักษรจากรูปที่ 4.22 จะได้แพทเทิร์นตัวอักษรที่เพี้ยน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลองและวิจารณ์

5.1 บทสรุป

- ซอฟต์แวร์ที่พัฒนาขึ้นในส่วนการประมวลผลภาพดิจิทัล สามารถทำการค้นหาตำแหน่งป้ายทะเบียนของยานพาหนะและสามารถแยกตัวอักษรของป้ายทะเบียนได้
- ซอฟต์แวร์ที่พัฒนาขึ้นในส่วนการแยกคุณลักษณะเฉพาะของภาพโดยใช้การแปลงดิสครีตโคไซน์ สามารถทำการลดข้อมูลของภาพตัวอักษรแต่ละตัวหลังจากผ่านการปรับขนาดให้เท่ากัน (Normalization by Resizing) ทำให้ลดความซับซ้อนในการออกแบบโครงข่ายประสาทเทียมได้
- ซอฟต์แวร์ที่พัฒนาขึ้นในส่วนการรู้จำป้ายทะเบียนยานพาหนะ โดยการใช้โครงข่ายประสาทเทียม สามารถรู้จำตัวอักษรต่างๆ บนป้ายทะเบียนได้
- ซอฟต์แวร์ที่พัฒนาขึ้นในการติดต่อกับระบบฐานข้อมูล สามารถติดต่อกับฐานข้อมูลได้
- การรวมการทำงานของซอฟต์แวร์ในแต่ละส่วนเข้าด้วยกันสามารถทำงานร่วมกันได้

5.2 บทวิจารณ์

โครงการนี้ ทำให้ทราบถึงทฤษฎีและขั้นตอนการทำงานต่างๆ ของการประมวลผลภาพดิจิทัล การแยกคุณลักษณะเฉพาะของภาพ ตลอดจนกระบวนการทำงานของโครงข่ายประสาทเทียม การทำงานร่วมกันได้ของซอฟต์แวร์แต่ละส่วน ทำให้บรรลุตามวัตถุประสงค์ที่ได้ตั้งไว้ในขอบเขตของโครงการและสามารถนำไปพัฒนาต่อ ให้มีประสิทธิภาพมากยิ่งขึ้นได้

5.3 ปัญหาและอุปสรรคที่พบ

ปัญหาที่พบบวกในโครงการ จะอยู่ในส่วนของการค้นหาตำแหน่งป้ายทะเบียนและการแยกตัวอักษรจากป้ายทะเบียนเป็นส่วนใหญ่

ปัญหาที่ทำให้ค้นหาตำแหน่งป้ายทะเบียนไม่ได้ ซึ่งเกิดจาก

1. ขนาดของไฟหน้า หรือส่วนอื่นๆ บริเวณด้านหน้าของยานพาหนะ มีลักษณะคล้ายกับลักษณะของป้ายทะเบียน ทำให้การตัดสินใจในการหาตำแหน่งป้ายทะเบียนเกิดความผิดพลาด
2. หลังจากการจับภาพยานพาหนะแล้ว ภาพในส่วนที่เป็นป้ายทะเบียนมีความเอียงมากเกินไป กล่าวคือ ไม่มีลักษณะความเป็นสี่เหลี่ยมผืนผ้า ซึ่งเป็นรูปร่างของป้ายทะเบียน
3. ป้ายทะเบียนที่ถูกบังจากการปรับแต่งโครงสร้างของยานพาหนะ เช่น การเปลี่ยนกันชน เป็นต้น

ปัญหาที่เกิดขึ้นดังที่กล่าวไว้ข้างต้น อาจแก้ไขโดยการกำหนดเงื่อนไขในการติดตั้งกล้องจับภาพ เพื่อให้ได้ภาพยานพาหนะที่มีตำแหน่งของป้ายทะเบียนชัดเจน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัญหาในการแยกตัวอักษรจากป้ายทะเบียนเป็นส่วนๆ ในบางครั้งมีความผิดพลาด ซึ่งเกิดจาก

1. ป้ายทะเบียนชำรุด ตัวหนังสือจาง มีคราบฝุ่น โคลนติด มีการฉีกขาด พื้นทะเบียนสีเข้ม (ป้ายแดง)
2. ป้ายทะเบียนที่มีเนื้อติดปนกับตัวอักษรบนป้ายทะเบียน ซึ่งเกิดจากการติดแผ่นป้าย

ปัญหาดังกล่าว ไม่สามารถทำการแก้ไขได้ในส่วนของซอฟต์แวร์ที่พัฒนาขึ้นในโครงการนี้ได้ ซึ่งยังเป็นข้อจำกัดของระบบอยู่



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] กิตติ ไพฑูรย์วัฒนกิจ. การประมวลผลภาพดิจิทัล. พิมพ์ครั้งที่ 1. กรุงเทพมหานคร : สจล, 2549
- [2] นวภัค เอื้ออนันต์. Artificial Neural Network วงจรข่ายนิวรอลเทียม. ปรินูญานิพนธ์วิศวกรรมศาสตรบัณฑิต. มหาวิทยาลัยขอนแก่น, 2547
- [3] พุทธิชัย สัมมา. การต่อภาพด้วยเทคนิคฮัฟและเฟสคอรี่เลขชั้นๆ. วิทยานิพนธ์วิศวกรรมศาสตรมหาบัณฑิต. สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2548
- [4] ไพศาล โมลิสกุลมงคล. พัฒนา Web Database ด้วย PHP. พิมพ์ครั้งที่ 1. กรุงเทพมหานคร : ไทยเจริญการพิมพ์, 2538
- [5] มนต์ สัจวรศิลป์, วรรรัตน์ ภัทรอมรกุล. คู่มือการใช้ Matlab ฉบับสมบูรณ์. พิมพ์ครั้งที่ 1 นนทบุรี : อินโฟเพรส, 2543
- [6] ศิวกร รัฐสุขชัยกุล. การประมวลผลภาพด้วยโปรแกรมแมทแลบ. ปรินูญานิพนธ์วิศวกรรมศาสตรบัณฑิต. สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2545
- [7] เศรษฐพล ลินปรัชญา. การจำตัวอักษรแบบตัวพิมพ์โดยนิวรอลเน็ตเวิร์ค. วิทยานิพนธ์วิศวกรรมศาสตรมหาบัณฑิต. สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2540
- [8] อานันตยา ทองเพ็ญ, พรเทพ สงวนถ้อย. ระบบขอบภาพด้วยอุปกรณ์ฮาร์ดแวร์. ปรินูญานิพนธ์วิศวกรรมศาสตรบัณฑิต. สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2545
- [9] Z.Mohd-Yusof, I.Suleiman, Z.Asper, "Implementation of Two Dimensional Forward DCT And Inverse DCT Using FPGA, IEEE, 2001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

การคำนวณ Neural Network ขั้นพื้นฐานในลักษณะต่างๆ

1. การใช้ perceptron แก้ปัญหาตรรกะ “และ” (AND) แบบ 1 กับ -1

ตารางที่ 1 ตารางความจริงของ AND

x1	x2	t
1	1	1
1	-1	-1
-1	1	-1
-1	-1	-1

ค่าระดับเริ่มต้นเป็น

$$w = [0 \ 0 \ 0]$$

มีค่าอัตราการเรียนรู้

$$\eta = 1$$

สามารถเขียนค่าน้ำหนัก (weight) และอินพุต ในรูปของเวกเตอร์ ดังนี้

$$w = [w_1, w_2, \dots, w_n]^T$$

$$x = [x_1, x_2, \dots, x_n, 1]^T$$

จะหาค่า u ได้จาก

$$u = w^T x$$

และหาค่า y ได้จาก

$$y = \begin{cases} 1, & u \geq 0 \\ -1, & u < 0 \end{cases}$$

ตารางที่ 2 ตารางการคำนวณ Perceptron ของ AND

	x1	x2	1	u	y	t	w1	w2	θ
	-	-	-	-	-	-	0	0	0
รอบที่ 1	1	1	1	0.3	0	1	0	0	0
	1	-1	1	0	1	-1	-1	1	-1
	-1	1	1	1	1	-1	0	0	-2
	-1	-1	1	-2	-1	-1	0	0	-2
รอบที่ 2	1	1	1	-2	-1	1	1	1	-1
	1	-1	1	-1	-1	-1	1	1	-1
	-1	1	1	-1	-1	-1	1	1	-1
	-1	-1	1	-3	-1	-1	1	1	-1
รอบที่ 3	1	1	1	1	1	1	1	-1	

2. การใช้ perceptron แก้ปัญหาตรรกะ “หรือ” (OR) แบบ 1 กับ -1

ตารางที่ 3 ตารางความจริงของ OR

x1	x2	t
1	1	1
1	-1	1
-1	1	1
-1	-1	-1

ค่าระดับเริ่มต้นเป็น $w = [0\ 0\ 0]$

มีค่าอัตราการเรียนรู้ $\eta = 1$

สามารถเขียนค่าน้ำหนัก (weight) และอินพุท ในรูปของเวกเตอร์ ดังนี้

$$w = [w_1, w_2, \dots, w_n]^T$$

$$x = [x_1, x_2, \dots, x_n, 1]^T$$

จะหาค่า u ได้จาก $u = w^T x$

และหาค่า y ได้จาก $y = \begin{cases} 1, & u \geq 0 \\ -1, & u < 0 \end{cases}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4 ตารางการคำนวณ Perceptron ของ OR

	x1	x2	1	u	y	t	w1	w2	θ
	-	-	-	-	-	-	0	0	0
รอบที่ 1	1	1	1	0	1	1	0	0	0
	1	-1	1	0	1	1	0	0	0
	-1	1	1	0	1	1	0	0	0
	-1	-1	1	0	1	-1	0	0	-2
รอบที่ 2	1	1	1	-2	-1	1	2	2	0
	1	-1	1	2	1	1	2	2	0
	-1	1	1	2	1	1	2	2	0
	-1	-1	1	0	1	-1	2	2	-2
รอบที่ 3	1	1	1	2	-1	1	2	2	-2
	1	-1	1	0	1	1	2	2	-2
	-1	1	1	0	1	1	2	2	-2
	-1	-1	1	-2	-1	-1	2	2	-2

3. การใช้ perceptron แก้ปัญหาตรรก “และ” (AND) แบบ 1 กับ 0

ตารางที่ 5 ตารางความจริงของ AND

x1	x2	t
1	1	1
1	0	0
0	1	0
0	0	0

ค่าระดับเริ่มต้นเป็น $w = [0 \ 0 \ 0]$

มีค่าอัตราการเรียนรู้ $\eta = 1$

สามารถเขียนค่าน้ำหนัก (weight) และอินพุท ในรูปของเวกเตอร์ ดังนี้

$$w = [w_1, w_2, \dots, w_n]^T$$

$$x = [x_1, x_2, \dots, x_n, 1]^T$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะหาค่า u ได้จาก $u = w^T x$

และหาค่า y ได้จาก $y = \begin{cases} 1, & u > 1 \\ 0, & u \leq 1 \end{cases}$

ตารางที่ 6 ตารางการคำนวณ Perceptron ของ AND

	x1	x2	1	u	y	t	w1	w2	θ
	-	-	-	-	-	-	0.1	0.1	0.1
รอบที่ 1	1	1	1	0.3	0	1	1.1	1.1	1.1
	1	0	1	2.2	1	0	0.1	1.1	0.1
	0	1	1	1.2	1	0	0.1	0.1	-0.9
	0	0	1	-0.9	0	0	0.1	0.1	-0.9
รอบที่ 2	1	1	1	-0.7	0	1	1.1	1.1	0.1
	1	0	1	1.2	1	0	0.1	1.1	-0.9
	0	1	1	0.2	0	0	0.1	1.1	-0.9
	0	0	1	-0.9	0	0	0.1	1.1	-0.9
รอบที่ 3	1	1	1	0.3	0	1	1.1	2.1	0.1
	1	0	1	1.2	1	0	0.1	2.1	-0.9
	0	1	1	1.2	1	0	0.1	1.1	-1.9
	0	0	1	-1.9	0	0	0.1	1.1	-1.9
รอบที่ 4	1	1	1	-0.7	0	1	1.1	2.1	-0.9
	1	0	1	0.2	0	0	1.1	2.1	-0.9
	0	1	1	1.2	1	0	1.1	1.1	-1.9
	0	0	1	-1.9	0	0	1.1	1.1	-1.9
รอบที่ 5	1	1	1	0.3	0	1	2.1	2.1	-0.9
	1	0	1	1.2	1	0	1.1	2.1	-1.9
	0	1	1	0.2	0	0	1.1	2.1	-1.9
	0	0	1	-1.9	0	0	1.1	2.1	-1.9
รอบที่ 6	1	1	1	1.3	1	1	1.1	2.1	-1.9
	1	0	1	-0.8	0	0	1.1	2.1	-1.9
	0	1	1	0.2	0	0	1.1	2.1	-1.9
	0	0	1	-1.9	0	0	1.1	2.1	-1.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. การใช้ perceptron แก้ปัญหาตรรก “หรือ” (OR) แบบ 1 กับ 0

ตารางที่ 7 ตารางความจริงของ OR

x1	x2	t
1	1	1
1	0	1
0	1	1
0	0	0

ค่าระดับเริ่มต้นเป็น $w = [0\ 0\ 0]$

มีค่าอัตราการเรียนรู้ $\eta = 1$

สามารถเขียนค่าน้ำหนัก (weight) และอินพุท ในรูปของเวกเตอร์ ดังนี้

$$w = [w_1, w_2, \dots, w_n]^T$$

$$x = [x_1, x_2, \dots, x_n, 1]^T$$

จะหาค่า u ได้จาก $u = w^T x$

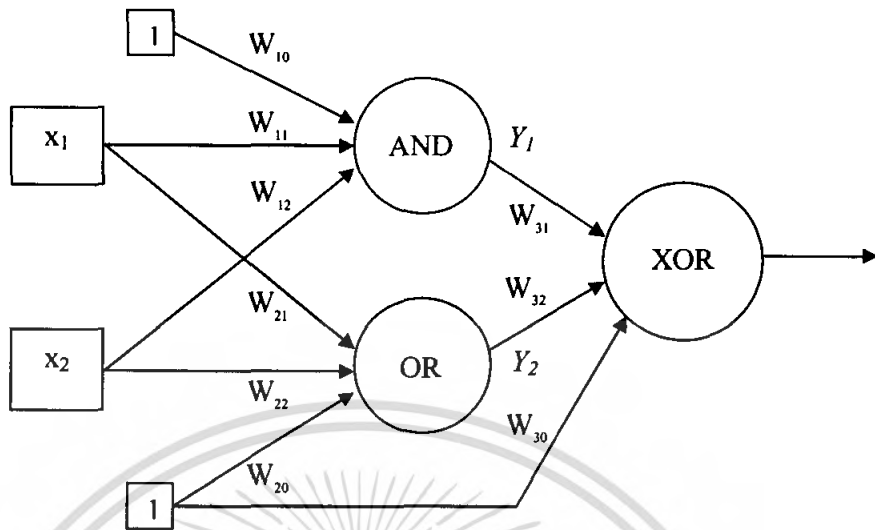
และหาค่า y ได้จาก $y = \begin{cases} 1, & u > 1 \\ 0, & u \leq 1 \end{cases}$

ตารางที่ 8 ตารางการคำนวณ Perceptron ของ OR

	x1	x2	1	u	y	t	w1	w2	θ
	-	-	-	-	-	-	0.1	0.1	0.1
รอบที่ 1	1	1	1	0.3	0	1	1.1	1.1	1.1
	1	0	1	2.2	1	1	1.1	1.1	1.1
	0	1	1	2.2	1	1	1.1	1.1	1.1
	0	0	1	1.1	1	0	1.1	1.1	0.1
รอบที่ 2	1	1	1	2.3	1	1	1.1	1.1	0.1
	1	0	1	1.2	1	1	1.1	1.1	0.1
	0	1	1	1.2	1	1	1.1	1.1	0.1
	0	0	1	0.1	0	0	1.1	1.1	0.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. การใช้ perceptron แก้ปัญหาตรรก “หรือ” (XOR)



รูปที่ 1 การออกแบบโครงข่าย

ตารางที่ 9 ตารางความจริงของ XOR

x1	x2	y1	y2	t
1	1	1	-1	-1
1	-1	-1	1	1
-1	1	-1	1	1
-1	-1	-1	1	-1

ค่าระดับเริ่มต้นเป็น $w = [0\ 0\ 0]$

มีค่าอัตราการเรียนรู้ $\eta = 1$

สามารถเขียนค่าน้ำหนัก (weight) และอินพุต ในรูปของเวกเตอร์ ดังนี้

$$w = [w_1, w_2, \dots, w_n]^T$$

$$x = [x_1, x_2, \dots, x_n, 1]^T$$

จะหาค่า u ได้จาก $u = w^T x$

และหาค่า y ได้จาก $y = \begin{cases} 1, & u \geq 0 \\ 0, & u < 0 \end{cases}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 10 ตารางการคำนวณ Perceptron ของ XOR

	x1	x2	1	u	y	t	w	w	θ	
	-	-	-	-	-	-	0	0	0	
รอบที่ 1	1	1	1	0	1	1	0	0	0	
	1	-1	1	0	1	-1	-1	1	-1	
	-1	1	1	1	1	-1	0	0	-2	
	-1	-1	1	-2	-1	-1	0	0	-2	
	1	1	1	0	1	1	0	0	0	
	1	-1	1	0	1	1	0	0	0	
	-1	1	1	0	1	1	0	0	0	
	-1	-1	1	0	1	-1	1	1	-1	
	1	1	1	0	1	-1	-1	-1	-1	
	1	-1	1	-3	-1	1	0	0	0	
	-1	1	1	0	1	1	0	0	0	
	-1	-1	1	0	1	-1	1	-1	-1	
	รอบที่ 2	1	1	1	-2	-2	1	1	1	-1
		1	-1	1	-1	-1	-1	1	1	-1
		-1	1	1	-1	-1	-1	1	1	-1
		-1	-1	1	-3	-1	-1	1	1	-1
1		1	1	1	1	1	1	1	-1	
1		-1	1	-1	-1	1	2	0	0	
-1		1	1	-2	-1	1	1	1	1	
-1		-1	1	-1	-1	-1	1	1	1	
1		1	1	-3	-1	-1	1	-1	-1	
1		-1	1	-1	-1	1	0	-2	0	
-1		1	1	2	1	1	0	-2	0	
-1		-1	1	2	1	-1	1	-1	-1	
รอบที่ 3	1	1	1	1	1	1	1	1	-1	
	1	-1	1	-1	-1	-1	1	1	-1	
	-1	1	1	-1	-1	-1	1	1	-1	
	-1	-1	1	-3	-1	-1	1	1	-1	
	1	1	1	3	1	1	1	1	1	
	1	-1	1	1	1	1	1	1	1	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 10 (ต่อ) ตารางการคำนวณ Perceptron ของ XOR

	-1	1	1	1	1	1	1	1	1
	-1	-1	1	-1	-1	-1	1	1	1
	1	1	1	-1	-1	-1	1	-1	-1
	1	-1	1	-3	-1	1	0	0	0
	-1	1	1	0	1	1	0	0	0
	-1	-1	1	0	1	-1	1	1	-1
รอบที่ 4	1	1	1	1	1	1	1	1	-1
	1	-1	1	-1	-1	-1	1	1	-1
	-1	1	1	-1	-1	-1	1	1	-1
	-1	-1	1	-3	-1	-1	1	1	-1
	1	1	1	3	1	1	1	1	1
	1	-1	1	1	1	1	1	1	1
	-1	1	1	1	1	1	1	1	1
	-1	-1	1	-1	-1	-1	1	1	1
	1	1	1	1	1	-1	0	0	-2
	1	-1	1	-2	-1	1	-1	1	-1
	-1	1	1	1	1	1	-1	1	-1
	-1	-1	1	-1	-1	-1	-1	1	-1
รอบที่ 5	1	1	1	1	1	1	1	1	-1
	1	-1	1	-1	-1	-1	1	1	-1
	-1	1	1	-1	-1	-1	1	1	-1
	-1	-1	1	-3	-1	-1	1	1	-1
	1	1	1	3	1	1	1	1	1
	1	-1	1	1	1	1	1	1	1
	-1	1	1	1	1	1	1	1	1
	-1	-1	1	-1	-1	-1	1	1	1
	1	1	1	-1	-1	-1	-1	1	-1
	1	-1	1	1	1	1	-1	1	-1
	-1	1	1	1	1	1	-1	1	-1
	-1	-1	1	-1	-1	-1	-1	1	-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

Program source codes.

1. Main program.

```
% Main program.
fprintf('Transportation Classification System.\n');
IN_01=IM_CAPTURE(vid);

% Pre-processing.
OUT_01=RGB_GRAY(IN_01);
OUT_02=IM_BIN_8NEIGHBORS(IM_THRESHOLD(OUT_01,'I'));
[OUT_03,OUT_04,OUT_05]=FIND_PLATE(OUT_02);

% Character segmentation.
[T,C1,C2,C3,C4,C5,C6]=CHAR_SEGMENT(OUT_03);
if C1==0
    fprintf('\nCannot segment character on license-plate.\n');
    return
end
CHAR_DISP(C1,C2,C3,C4,C5,C6);
C1=IM_THRESHOLD(imresize(C1,[30 20]),'N');
C2=IM_THRESHOLD(imresize(C2,[30 20]),'N');
C3=IM_THRESHOLD(imresize(C3,[30 20]),'N');
C4=IM_THRESHOLD(imresize(C4,[30 20]),'N');
C5=IM_THRESHOLD(imresize(C5,[30 20]),'N');
C6=IM_THRESHOLD(imresize(C6,[30 20]),'N');
CHAR_DISP(C1,C2,C3,C4,C5,C6);

% Feature Extraction using Discrete Cosine Transform.
D1=MTX_TRANSFORM5(DCT_TRANSFORM(C1,'N5'),'N');
D2=MTX_TRANSFORM5(DCT_TRANSFORM(C2,'N5'),'N');
D3=MTX_TRANSFORM5(DCT_TRANSFORM(C3,'N5'),'N');
D4=MTX_TRANSFORM5(DCT_TRANSFORM(C4,'N5'),'N');
D5=MTX_TRANSFORM5(DCT_TRANSFORM(C5,'N5'),'N');
D6=MTX_TRANSFORM5(DCT_TRANSFORM(C6,'N5'),'N');

% License-plate result.
if T==4
    [Char1,M1]=CHAR_CLASSIFY(D1);
    [Char2,M2]=CHAR_CLASSIFY(D2);
    [Char3,M3]=CHAR_CLASSIFY(D3);
    [Char4,M4]=CHAR_CLASSIFY(D4);
    Plate=[Char1 Char2 Char3 Char4];
elseif T==5
    [Char1,M1]=CHAR_CLASSIFY(D1);
    [Char2,M2]=CHAR_CLASSIFY(D2);
    [Char3,M3]=CHAR_CLASSIFY(D3);
    [Char4,M4]=CHAR_CLASSIFY(D4);
    [Char5,M5]=CHAR_CLASSIFY(D5);
    Plate=[Char1 Char2 Char3 Char4 Char5];
elseif T==6
    [Char1,M1]=CHAR_CLASSIFY(D1);
    [Char2,M2]=CHAR_CLASSIFY(D2);
    [Char3,M3]=CHAR_CLASSIFY(D3);
    [Char4,M4]=CHAR_CLASSIFY(D4);
    [Char5,M5]=CHAR_CLASSIFY(D5);
    [Char6,M6]=CHAR_CLASSIFY(D6);
```

```

    Plate=[Char1 Char2 Char3 Char4 Char5 Char6];
elseif T==7
    [Char1,M1]=CHAR_CLASSIFY(D1);
    [Char2,M2]=CHAR_CLASSIFY(D2);
    [Char3,M3]=CHAR_CLASSIFY(D3);
    [Char4,M4]=CHAR_CLASSIFY(D4);
    [Char5,M5]=CHAR_CLASSIFY(D5);
    [Char6,M6]=CHAR_CLASSIFY(D6);
    Plate=[Char1 Char2 Char3 Char4 Char5 Char6];
end

% Database connection checking.
DBCHK1=whos;
[STRUCT_ROW,STRUCT_COL]=size(DBCHK1);
DBCHK2=0;
for i=1:1:STRUCT_ROW
    if
strcmp(DBCHK1(i).name,'Connection_01')==true||strcmp(DBCHK1(i).name,'
Connection_02')==true
        DBCHK2=DBCHK2+1;
    end
end
if DBCHK2~=2
    Connection_01=database('REF_DB','root','BallNaoNao');
    exec(Connection_01,'SET NAMES TIS620');
    Connection_02=database('LOG_DB','root','BallNaoNao');
    exec(Connection_02,'SET NAMES TIS620');
end

% Compare to database and display all result.
SQL=['SELECT * FROM data WHERE license="' Plate '"'];
Cursor=exec(Connection_01,SQL);
Cursor=fetch(Cursor);
data=Cursor.Data;
CHK=data{1};
fprintf('\nLicense-plate : %s\n',Plate);
if strcmp(CHK,'No Data')==true
    fprintf('No data for this license-plate.\n\n');
else
    if data{4}==4
        FEE=40; % Adjustable
    elseif data{4}==6
        FEE=60; % Adjustable
    elseif data{4}>6
        FEE=80; % Adjustable
    end
    fprintf('Province : %s\n',data{2});
    fprintf('Owner : %s\n',data{3});
    fprintf('Wheel : %.0f\n',data{4});
    fprintf('Color : %s\n',data{5});
    fprintf('Brand : %s\n',data{6});
    fprintf('Expressway FEE : %.0f\n\n',FEE);
end

% Record to database. (Data logger)
if strcmp(CHK,'No Data')==true
    clear data
    data=cell(1,6);
    data{2}='Unknown';
    data{3}='Unknown';
    data{5}='Unknown';

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

data{6}='Unknown';
data{4}=input('Enter wheel of transport : ');
FEE=input('Enter expressway fee      : ');
fprintf('\n');
end
DATETIME=datestr(clock);
WHEEL=num2str(data{4});
FEE=num2str(FEE);
SQL=['INSERT INTO data
(license,province,name,wheel,color,brand,datetime,fee) values ('
Plate "'," data{2} ',' data{3} ',' WHEEL ',' data{5} ',' data{6}
',' DATETIME ',' FEE ')'];
exec(Connection_02,SQL);
fprintf('Data saved to database.\n\n');

clear T C1 C2 C3 C4 C5 C6 D1 D2 D3 D4 D5 D6
clear Char1 Char2 Char3 Char4 Char5 Char6 Plate

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Digital image pre-processing part.

2.1 RGB image to grayscale image.

```
function [OUT]=RGB_GRAY(IN)

%RGB_GRAY Convert RGB image to grayscale.
%
% OUT = RGB_GRAY(IN)
% Converts RGB image to grayscale image.

[H,W,L]=size(IN);
OUT=zeros(H,W);
for i=1:1:H
    for j=1:1:W
        OUT(i,j)=((0.587*IN(i,j,L-2))+(0.114*IN(i,j,L-1))+(0.299*IN(i,j,L)));
    end
end
OUT=uint8(OUT);
```

2.2 Grayscale image to binary image.

```
function [OUT]=IM_THRESHOLD(IN,args)

%IM_THRESHOLD Convert intensity image to binary image.
%
% OUT = IM_THRESHOLD(IN,'N')
% More than threshold value set to bit '1'.
% Less than threshold value set to bit '0'.
%
% OUT = IM_THRESHOLD(IN,'I')
% More than threshold value set to bit '0'.
% Less than threshold value set to bit '1'.

[H,W]=size(IN);OUT=zeros(H,W);
THRESHOLD=(max(max(IN))-min(min(IN)))/2;
if strcmp(args,'N')==true
    for i=1:1:H
        for j=1:1:W
            if IN(i,j)>=THRESHOLD
                OUT(i,j)=1;
            else
                OUT(i,j)=0;
            end
        end
    end
elseif strcmp(args,'I')==true
    for i=1:1:H
        for j=1:1:W
            if IN(i,j)>=THRESHOLD
                OUT(i,j)=0;
            else
                OUT(i,j)=1;
            end
        end
    end
end
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 Sobel edge detector.

```
function [OUT]=SOBEL_EDGE(IN)

%SOBEL_EDGE Sobel edge detector.
%
% [OUT] = SOBEL_EDGE(IN)
% Return edge of binary image.

[H,W]=size(IN);
IN=double(IN);
OUT=zeros(H,W);
for i=1:1:(H-2)
    for j=1:1:(W-2)
        SH=((IN(i+2,j)+(2*IN(i+2,j+1))+IN(i+2,j+2))-
(IN(i,j+2)+(2*IN(i,j+1))+IN(i,j)))/4;
        SV=((IN(i+2,j+2)+(2*IN(i+1,j+2))+IN(i,j+2))-
(IN(i,j)+(2*IN(i+1,j))+IN(i+2,j)))/4;
        OUT(i,j)=sqrt((SH^2)+(SV^2));
    end
end
```

2.4 Hough transform.

```
function [OUT]=HOUGH_TRANSFORM(IN,args)

%HOUGH_TRANSFORM Hough transform, line detection.
%
% OUT = HOUGH_TRANSFORM(IN,'V')
% Return theta = 0 hough space, vertical straight line.
%
% OUT = HOUGH_TRANSFORM(IN,'H')
% Return theta = 90 hough space, horizontal straight line.

[H,W]=size(IN);
if strcmp(args,'V')==true
    OUT=zeros(1,W);
    for i=H:-1:1
        for j=1:1:W
            if IN(i,j)==1
                RHO=j; % Absolute rho. (|r|)
                OUT(RHO)=OUT(RHO)+1;
            end
        end
    end
elseif strcmp(args,'H')==true
    OUT=zeros(1,H);
    for i=H:-1:1
        for j=1:1:W
            if IN(i,j)==1
                RHO=H-i+1; % Absolute rho. (|r|)
                OUT(RHO)=OUT(RHO)+1;
            end
        end
    end
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 License-plate detection.

```
function [OUT1,OUT2,OUT3]=FIND_PLATE(IN)

%FIND_PLATE Extracts license-plate from image.
%
% [OUT1,OUT2,OUT3] = FIND_PLATE(IN)
% OUT1 return binary license-plate image.
% OUT2 return straight line of input image.
% OUT3 return image which cropped by mask.

[H,W]=size(IN);
IM=SOBEL_EDGE(IN);
IM=IM_BIN_8NEIGHBORS(IM_THRESHOLD(IM,'N'));

MASK1=round(H/4);
MASK2=round(W/4);
IM_EDGE=zeros(H,W);
for i=(MASK1+1):1:(H-MASK1)
    for j=(MASK2+1):1:(W-MASK2)
        IM_EDGE(i,j)=IM(i,j);
    end
end

OUT3=zeros(H,W);
for i=(MASK1+1):1:(H-MASK1)
    for j=(MASK2+1):1:(W-MASK2)
        OUT3(i,j)=IM_EDGE(i,j);
    end
end

HT_V=HOUGH_TRANSFORM(IM_EDGE,'V');

[M,E]=max(HT_V);
VER1=E;
HT_V(E)=0;
[M,E]=max(HT_V);
VER2=E;
HT_V(E)=0;
[M,E]=max(HT_V);
VER3=E;
HT_V(E)=0;
[M,E]=max(HT_V);
VER4=E;

CHECK=0;
while CHECK~=1
    if (VER4-VER3>0) && (VER3-VER2>0) && (VER2-VER1>0)
        CHECK=1;
    elseif (VER4-VER3)<0
        FLAG=VER3;
        VER3=VER4;
        VER4=FLAG;
    elseif (VER3-VER2)<0
        FLAG=VER2;
        VER2=VER3;
        VER3=FLAG;
    elseif (VER2-VER1)<0
        FLAG=VER1;
        VER1=VER2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        VER2=FLAG;
    end
end

OUT2=zeros(H,W);
OUT2(:,VER1)=1;
OUT2(:,VER2)=1;
OUT2(:,VER3)=1;
OUT2(:,VER4)=1;

if (VER2-VER1)>100
    SIDE1=VER1;
    SIDE2=VER2;
elseif (VER3-VER2)>100
    SIDE1=VER2;
    SIDE2=VER3;
elseif (VER4-VER3)>100
    SIDE1=VER3;
    SIDE2=VER4;
elseif (VER3-VER1)>100
    SIDE1=VER1;
    SIDE2=VER3;
elseif (VER4-VER1)>100
    SIDE1=VER1;
    SIDE2=VER4;
elseif (VER4-VER2)>100
    SIDE1=VER2;
    SIDE2=VER4;
end

WIDTH=SIDE2-SIDE1+1;
NCPLATE=zeros(H,WIDTH);
for i=1:1:H
    for j=1:1:WIDTH
        NCPLATE(i,j)=IM_EDGE(i,(SIDE1-1+j));
    end
end

HT_H=HOUGH_TRANSFORM(NCPLATE,'H');

for i=H:-1:1
    if HT_H(H-i+1)>=round(WIDTH*50/100)
        OUT2(i,:)=1;
    end
end

for i=H:-1:round(WIDTH/2.10)
    if sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-round(WIDTH/2.80),:))>=WIDTH
        TOP=i-round(WIDTH/2.80);
        BOT=i;
        break
    elseif sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-
round(WIDTH/2.75),:))>=WIDTH
        TOP=i-round(WIDTH/2.75);
        BOT=i;
        break
    elseif sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-
round(WIDTH/2.70),:))>=WIDTH
        TOP=i-round(WIDTH/2.70);
        BOT=i;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        break
    elseif sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-
round(WIDTH/2.65),:))>=WIDTH
        TOP=i-round(WIDTH/2.65);
        BOT=i;
        break
    elseif sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-
round(WIDTH/2.60),:))>=WIDTH
        TOP=i-round(WIDTH/2.60);
        BOT=i;
        break
    elseif sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-
round(WIDTH/2.55),:))>=WIDTH
        TOP=i-round(WIDTH/2.55);
        BOT=i;
        break
    elseif sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-
round(WIDTH/2.50),:))>=WIDTH
        TOP=i-round(WIDTH/2.50);
        BOT=i;
        break
    elseif sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-
round(WIDTH/2.45),:))>=WIDTH
        TOP=i-round(WIDTH/2.45);
        BOT=i;
        break
    elseif sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-
round(WIDTH/2.40),:))>=WIDTH
        TOP=i-round(WIDTH/2.40);
        BOT=i;
        break
    elseif sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-
round(WIDTH/2.35),:))>=WIDTH
        TOP=i-round(WIDTH/2.35);
        BOT=i;
        break
    elseif sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-
round(WIDTH/2.30),:))>=WIDTH
        TOP=i-round(WIDTH/2.30);
        BOT=i;
        break
    elseif sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-
round(WIDTH/2.25),:))>=WIDTH
        TOP=i-round(WIDTH/2.25);
        BOT=i;
        break
    elseif sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-
round(WIDTH/2.20),:))>=WIDTH
        TOP=i-round(WIDTH/2.20);
        BOT=i;
        break
    elseif sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-
round(WIDTH/2.15),:))>=WIDTH
        TOP=i-round(WIDTH/2.15);
        BOT=i;
        break
    elseif sum(OUT2(i,:))>=WIDTH&&sum(OUT2(i-
round(WIDTH/2.10),:))>=WIDTH
        TOP=i-round(WIDTH/2.10);
        BOT=i;
        break

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

else
    TOP=0;
    BOT=0;
end
end
end

HIGH=BOT-TOP+1;
OUT1=zeros(HIGH,WIDTH);
for i=1:1:HIGH
    for j=1:1:WIDTH
        OUT1(i,j)=IN(TOP-1+i,SIDE1-1+j);
    end
end
end

```

2.6 Character segmentation.

```

function [TYPE,C1,C2,C3,C4,C5,C6]=CHAR_SEGMENT(IN)

%CHAR_SEGMENT Extract character from license-plate binary image.
%
% [TYPE,C1,C2,C3,C4,C5,C6] = CHAR_SEGMENT(IN)
% Return character pattern variables and type of license-plate.

[H,W]=size(IN);
for i=1:1:H
    for j=1:1:W
        if i<=2||i>=(H-1)
            IN(i,j)=0;
        end
        if j<=2||j>=(W-1)
            IN(i,j)=0;
        end
    end
end
end

NEW_PLATE=FIT_CHAR(IN);

[HV,HH]=IM_BIN_HISTOGRAM(NEW_PLATE,'N');
[H,W]=size(NEW_PLATE);
FLAG=1;
TYPE=0;
SEC=zeros(1,14);
N=1;
for i=1:1:W
    if FLAG==1
        if HV(i)>ceil(H*2/100)
            TYPE=TYPE+1;
            SEC(1,N)=i;
            N=N+1;
            FLAG=0;
        end
    elseif FLAG==0
        if HV(i)<ceil(H*2/100)
            TYPE=TYPE+1;
            SEC(1,N)=i-1;
            N=N+1;
            FLAG=1;
        end
    end
end
end
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

TYPE=TYPE/2;
if (ceil(TYPE)-TYPE)~=0 || TYPE<4 || TYPE>7
    C1=0;C2=0;C3=0;C4=0;C5=0;C6=0;
    return
end

if TYPE==7
    C1=zeros(H, (SEC(2)-SEC(1)+1));
    C2=zeros(H, (SEC(4)-SEC(3)+1));
    C3=zeros(H, (SEC(8)-SEC(7)+1));
    C4=zeros(H, (SEC(10)-SEC(9)+1));
    C5=zeros(H, (SEC(12)-SEC(11)+1));
    C6=zeros(H, (SEC(14)-SEC(13)+1));
    for i=1:1:H
        for j=1:1:(SEC(2)-SEC(1)+1)
            C1(i,j)=NEW_PLATE(i, (SEC(1)-1+j));
        end
    end
    C1=FIT_CHAR(C1);
    for i=1:1:H
        for j=1:1:(SEC(4)-SEC(3)+1)
            C2(i,j)=NEW_PLATE(i, (SEC(3)-1+j));
        end
    end
    C2=FIT_CHAR(C2);
    for i=1:1:H
        for j=1:1:(SEC(8)-SEC(7)+1)
            C3(i,j)=NEW_PLATE(i, (SEC(7)-1+j));
        end
    end
    C3=FIT_CHAR(C3);
    for i=1:1:H
        for j=1:1:(SEC(10)-SEC(9)+1)
            C4(i,j)=NEW_PLATE(i, (SEC(9)-1+j));
        end
    end
    C4=FIT_CHAR(C4);
    for i=1:1:H
        for j=1:1:(SEC(12)-SEC(11)+1)
            C5(i,j)=NEW_PLATE(i, (SEC(11)-1+j));
        end
    end
    C5=FIT_CHAR(C5);
    for i=1:1:H
        for j=1:1:(SEC(14)-SEC(13)+1)
            C6(i,j)=NEW_PLATE(i, (SEC(13)-1+j));
        end
    end
    C6=FIT_CHAR(C6);
elseif TYPE==6
    C1=zeros(H, (SEC(2)-SEC(1)+1));
    C2=zeros(H, (SEC(4)-SEC(3)+1));
    C3=zeros(H, (SEC(6)-SEC(5)+1));
    C4=zeros(H, (SEC(8)-SEC(7)+1));
    C5=zeros(H, (SEC(10)-SEC(9)+1));
    C6=zeros(H, (SEC(12)-SEC(11)+1));
    for i=1:1:H
        for j=1:1:(SEC(2)-SEC(1)+1)
            C1(i,j)=NEW_PLATE(i, (SEC(1)-1+j));
        end
    end
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

C1=FIT_CHAR(C1);
for i=1:1:H
    for j=1:1:(SEC(4)-SEC(3)+1)
        C2(i,j)=NEW_PLATE(i,(SEC(3)-1+j));
    end
end
C2=FIT_CHAR(C2);
for i=1:1:H
    for j=1:1:(SEC(6)-SEC(5)+1)
        C3(i,j)=NEW_PLATE(i,(SEC(5)-1+j));
    end
end
C3=FIT_CHAR(C3);
for i=1:1:H
    for j=1:1:(SEC(8)-SEC(7)+1)
        C4(i,j)=NEW_PLATE(i,(SEC(7)-1+j));
    end
end
C4=FIT_CHAR(C4);
for i=1:1:H
    for j=1:1:(SEC(10)-SEC(9)+1)
        C5(i,j)=NEW_PLATE(i,(SEC(9)-1+j));
    end
end
C5=FIT_CHAR(C5);
for i=1:1:H
    for j=1:1:(SEC(12)-SEC(11)+1)
        C6(i,j)=NEW_PLATE(i,(SEC(11)-1+j));
    end
end
C6=FIT_CHAR(C6);
elseif TYPE==5
C1=zeros(H,(SEC(2)-SEC(1)+1));
C2=zeros(H,(SEC(4)-SEC(3)+1));
C3=zeros(H,(SEC(6)-SEC(5)+1));
C4=zeros(H,(SEC(8)-SEC(7)+1));
C5=zeros(H,(SEC(10)-SEC(9)+1));
C6=0;
for i=1:1:H
    for j=1:1:(SEC(2)-SEC(1)+1)
        C1(i,j)=NEW_PLATE(i,(SEC(1)-1+j));
    end
end
C1=FIT_CHAR(C1);
for i=1:1:H
    for j=1:1:(SEC(4)-SEC(3)+1)
        C2(i,j)=NEW_PLATE(i,(SEC(3)-1+j));
    end
end
C2=FIT_CHAR(C2);
for i=1:1:H
    for j=1:1:(SEC(6)-SEC(5)+1)
        C3(i,j)=NEW_PLATE(i,(SEC(5)-1+j));
    end
end
C3=FIT_CHAR(C3);
for i=1:1:H
    for j=1:1:(SEC(8)-SEC(7)+1)
        C4(i,j)=NEW_PLATE(i,(SEC(7)-1+j));
    end
end
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

C4=FIT_CHAR(C4);
for i=1:1:H
    for j=1:1:(SEC(10)-SEC(9)+1)
        C5(i,j)=NEW_PLATE(i,(SEC(9)-1+j));
    end
end
C5=FIT_CHAR(C5);
elseif TYPE==4
C1=zeros(H,(SEC(2)-SEC(1)+1));
C2=zeros(H,(SEC(4)-SEC(3)+1));
C3=zeros(H,(SEC(6)-SEC(5)+1));
C4=zeros(H,(SEC(8)-SEC(7)+1));
C5=0;C6=0;
for i=1:1:H
    for j=1:1:(SEC(2)-SEC(1)+1)
        C1(i,j)=NEW_PLATE(i,(SEC(1)-1+j));
    end
end
C1=FIT_CHAR(C1);
for i=1:1:H
    for j=1:1:(SEC(4)-SEC(3)+1)
        C2(i,j)=NEW_PLATE(i,(SEC(3)-1+j));
    end
end
C2=FIT_CHAR(C2);
for i=1:1:H
    for j=1:1:(SEC(6)-SEC(5)+1)
        C3(i,j)=NEW_PLATE(i,(SEC(5)-1+j));
    end
end
C3=FIT_CHAR(C3);
for i=1:1:H
    for j=1:1:(SEC(8)-SEC(7)+1)
        C4(i,j)=NEW_PLATE(i,(SEC(7)-1+j));
    end
end
C4=FIT_CHAR(C4);
end

```

2.7 Image enhancement. (Noise filtering)

```
function [OUT]=IM_BIN_8NEIGHBORS(IN)
```

```
%IM_BIN_8NEIGHBORS Binary image enhancement.
```

```
%
```

```
% OUT = IM_BIN_8NEIGHBORS(IN)
```

```
% Return enhanced binary image.
```

```
[H,W]=size(IN);
```

```
for i=1:1:(H-2)
```

```
    for j=1:1:(W-2)
```

```
        if
```

```
(IN(i,j)+IN(i,j+1)+IN(i,j+2)+IN(i+1,j)+IN(i+1,j+2)+IN(i+2,j)+IN(i+2,j+1)+IN(i+2,j+2))==0
```

```
            IN(i+1,j+1)=0;
```

```
        end
```

```
        if
```

```
(IN(i,j)+IN(i,j+1)+IN(i,j+2)+IN(i+1,j)+IN(i+1,j+2)+IN(i+2,j)+IN(i+2,j+1)+IN(i+2,j+2))==8
```

```
            IN(i+1,j+1)=1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        end
    end
end
OUT=IN;

```

2.8 Image enhancement. (Best shape character image)

```

function [OUT]=FIT_CHAR(IN)

%FIT_CHAR Character pattern size fitting.
%
%   OUT = FIT_CHAR(IN)
%   Return fitted character pattern variable.

[H,W]=size(IN);
[HV,HH]=IM_BIN_HISTOGRAM(IN,'N');
for i=1:1:H
    if HH(i)>=ceil(W*2/100)
        TOPROW=i;
        break
    end
end
for i=(TOPROW+1):1:H
    if HH(i)<ceil(W*2/100)
        BOTROW=i-1;
        break
    elseif i==H
        BOTROW=H;
    end
end
OUT=zeros(BOTROW-TOPROW+1,W);
for i=1:1:(BOTROW-TOPROW+1)
    for j=1:1:W
        OUT(i,j)=IN(i+TOPROW-1,j);
    end
end
end

```

2.9 Histogram computation.

```

function [HV,HH]=IM_BIN_HISTOGRAM(IN,args)

%IM_BIN_HISTOGRAM Binary image histogram computation.
%
%   [HV,HN] = IM_BIN_HISTOGRAM(IN,'N')
%   Compute binary image histogram in background process.
%
%   [HV,HN] = IM_BIN_HISTOGRAM(IN,'S')
%   Compute binary image histogram and display histogram.

[H,W]=size(IN);
PIXEL_HOR=1:1:W;
HV=zeros(1,W);
for ROW_SCAN=1:1:H
    for COL_SCAN=1:1:W
        if IN(ROW_SCAN,COL_SCAN)==1
            HV(COL_SCAN)=HV(COL_SCAN)+1;
        end
    end
end
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

PIXEL_VER=1:1:H;
HH=zeros(1,H);
for ROW_SCAN=1:1:H
    for COL_SCAN=1:1:W
        if IN(ROW_SCAN,COL_SCAN)==1
            HH(ROW_SCAN)=HH(ROW_SCAN)+1;
        end
    end
end
end
if strcmp(args,'N')==true
elseif strcmp(args,'S')==true
    figure(),bar(PIXEL_HOR,HV);
    xlabel('Horizontal image pixel');
    ylabel('Non-zero bit counted');
    title('Binary image vertical histogram');
    figure(),bar(PIXEL_VER,HH);
    xlabel('Vertical image pixel');
    ylabel('Non-zero bit counted');
    title('Binary image horizontal histogram');
end
end

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3 Feature extraction.

3.1 Discrete Cosine Transform. (DCT)

```
function [OUT]=DCT_TRANSFORM(IN,args)
```

```
%DCT_TRANSFORM Discrete Cosine Transform.
%
%   OUT = DCT_TRANSFORM(IN,'N10')
%   Discrete Cosine Transform with 10x10 DCT matrix.
%
%   OUT = DCT_TRANSFORM(IN,'N5')
%   Discrete Cosine Transform with 5x5 DCT matrix.
%
%   OUT = DCT_TRANSFORM(IN,'I10')
%   Inverse Discrete Cosine Transform with 10x10 DCT matrix.
%
%   OUT = DCT_TRANSFORM(IN,'I5')
%   Inverse Discrete Cosine Transform with 5x5 DCT matrix.
%
%   OUT = DCT_TRANSFORM(IN,'M10')
%   Transforms and multiply by specified mask with 10x10 DCT matrix.
%
%   OUT = DCT_TRANSFORM(IN,'M5')
%   Transforms and multiply by specified mask with 5x5 DCT matrix.

IN=double(IN);
if strcmp(args,'N10')==true
    DCT_MTX=dctmtx(10);
    OUT=blkproc(IN,[10 10],'P1*x*P2',DCT_MTX,DCT_MTX');
elseif strcmp(args,'N5')==true
    DCT_MTX=dctmtx(5);
    OUT=blkproc(IN,[5 5],'P1*x*P2',DCT_MTX,DCT_MTX');
elseif strcmp(args,'I10')==true
    DCT_MTX=dctmtx(10);
    OUT=blkproc(IN,[10 10],'P1*x*P2',DCT_MTX',DCT_MTX);
elseif strcmp(args,'I5')==true
    DCT_MTX=dctmtx(5);
    OUT=blkproc(IN,[5 5],'P1*x*P2',DCT_MTX',DCT_MTX);
elseif strcmp(args,'M10')==true
    DCT_MTX=dctmtx(10);
    OUT=blkproc(IN,[10 10],'P1*x*P2',DCT_MTX,DCT_MTX');
    MASK=[1 1 1 1 1 1 0 0 0
          1 1 1 1 1 1 0 0 0
          1 1 1 1 1 0 0 0 0
          1 1 1 1 0 0 0 0 0
          1 1 1 0 0 0 0 0 0
          1 1 0 0 0 0 0 0 0
          1 0 0 0 0 0 0 0 0
          0 0 0 0 0 0 0 0 0
          0 0 0 0 0 0 0 0 0
          0 0 0 0 0 0 0 0 0];
    OUT=blkproc(OUT,[10 10],'P1.*x',MASK);
elseif strcmp(args,'M5')==true
    DCT_MTX=dctmtx(5);
    OUT=blkproc(IN,[5 5],'P1*x*P2',DCT_MTX,DCT_MTX');
    MASK=[1 1 1 1 0
          1 1 1 0 0
          1 1 0 0 0
          1 0 0 0 0
          0 0 0 0 0];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    OUT=blkproc(OUT,[5 5],'P1.*x',MASK);
end

```

3.2 Matrix transformation for 5X5 DCT matrix.

```
function [OUT]=MTX_TRANSFORM5(IN,args)
```

```
%MTX_TRANSFORM5 Matrix transformation.
```

```

%
%   OUT = MTX_TRANSFORM(IN,'N')
%   Transforms 30x20 to 1x240 matrix.
%
%   OUT = MTX_TRANSFORM(IN,'I')
%   Transforms 1x240 to 30x20 matrix.

```

```

if strcmp(args,'N')==true
    [H,W]=size(IN);
    OUT=zeros(1,240);
    K=0;
    if H==30&&W==20
        for i=1:1:30
            for j=1:1:20
                if i==1||i==6||i==11||i==16||i==21||i==26
                    if j>=1&&j<=4
                        K=K+1;
                        OUT(K)=IN(i,j);
                    elseif j>=6&&j<=9
                        K=K+1;
                        OUT(K)=IN(i,j);
                    elseif j>=11&&j<=14
                        K=K+1;
                        OUT(K)=IN(i,j);
                    elseif j>=16&&j<=19
                        K=K+1;
                        OUT(K)=IN(i,j);
                    end
                elseif i==2||i==7||i==12||i==17||i==22||i==27
                    if j>=1&&j<=3
                        K=K+1;
                        OUT(K)=IN(i,j);
                    elseif j>=6&&j<=8
                        K=K+1;
                        OUT(K)=IN(i,j);
                    elseif j>=11&&j<=13
                        K=K+1;
                        OUT(K)=IN(i,j);
                    elseif j>=16&&j<=18
                        K=K+1;
                        OUT(K)=IN(i,j);
                    end
                elseif i==3||i==8||i==13||i==18||i==23||i==28
                    if j>=1&&j<=2
                        K=K+1;
                        OUT(K)=IN(i,j);
                    elseif j>=6&&j<=7
                        K=K+1;
                        OUT(K)=IN(i,j);
                    elseif j>=11&&j<=12
                        K=K+1;
                        OUT(K)=IN(i,j);
                    end
                end
            end
        end
    end
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

elseif j>=16&&j<=17
    K=K+1;
    OUT(K)=IN(i,j);
end
elseif i==4||i==9||i==14||i==19||i==24||i==29
    if j==1
        K=K+1;
        OUT(K)=IN(i,j);
    elseif j==6
        K=K+1;
        OUT(K)=IN(i,j);
    elseif j==11
        K=K+1;
        OUT(K)=IN(i,j);
    elseif j==16
        K=K+1;
        OUT(K)=IN(i,j);
    end
end
end
end
end
elseif strcmp(args,'I')==true
    [H,W]=size(IN);
    OUT=zeros(30,20);
    K=0;
    if H==1&&W==240
        for i=1:1:30
            for j=1:1:20
                if i==1||i==6||i==11||i==16||i==21||i==26
                    if j>=1&&j<=4
                        K=K+1;
                        OUT(i,j)=IN(K);
                    elseif j>=6&&j<=9
                        K=K+1;
                        OUT(i,j)=IN(K);
                    elseif j>=11&&j<=14
                        K=K+1;
                        OUT(i,j)=IN(K);
                    elseif j>=16&&j<=19
                        K=K+1;
                        OUT(i,j)=IN(K);
                    end
                elseif i==2||i==7||i==12||i==17||i==22||i==27
                    if j>=1&&j<=3
                        K=K+1;
                        OUT(i,j)=IN(K);
                    elseif j>=6&&j<=8
                        K=K+1;
                        OUT(i,j)=IN(K);
                    elseif j>=11&&j<=13
                        K=K+1;
                        OUT(i,j)=IN(K);
                    elseif j>=16&&j<=18
                        K=K+1;
                        OUT(i,j)=IN(K);
                    end
                elseif i==3||i==8||i==13||i==18||i==23||i==28
                    if j>=1&&j<=2
                        K=K+1;
                        OUT(i,j)=IN(K);
                    end
                end
            end
        end
    end
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

elseif j>=6&&j<=7
    K=K+1;
    OUT(i,j)=IN(K);
elseif j>=11&&j<=12
    K=K+1;
    OUT(i,j)=IN(K);
elseif j>=16&&j<=17
    K=K+1;
    OUT(i,j)=IN(K);
end
elseif i==4||i==9||i==14||i==19||i==24||i==29
if j==1
    K=K+1;
    OUT(i,j)=IN(K);
elseif j==6
    K=K+1;
    OUT(i,j)=IN(K);
elseif j==11
    K=K+1;
    OUT(i,j)=IN(K);
elseif j==16
    K=K+1;
    OUT(i,j)=IN(K);
end
end
end
end
end
end
end
end

```

3.3 Matrix transformation for 10×10 DCT matrix.

```

function [OUT]=MTX_TRANSFORM10(IN,args)

%MTX_TRANSFORM Matrix transformation.
%
% OUT = MTX_TRANSFORM(IN,'N')
% Transforms 30x20 to 1x168 matrix.
%
% OUT = MTX_TRANSFORM(IN,'I')
% Transforms 1x168 to 30x20 matrix.

if strcmp(args,'N')==true
    [H,W]=size(IN);
    OUT=zeros(1,168);
    K=0;
    if H==30&&W==20
        for i=1:1:30
            for j=1:1:20
                if i==1||i==11||i==21
                    if j>=1&&j<=7
                        K=K+1;
                        OUT(K)=IN(i,j);
                    elseif j>=11&&j<=17
                        K=K+1;
                        OUT(K)=IN(i,j);
                    end
                elseif i==2||i==12||i==22
                    if j>=1&&j<=6
                        K=K+1;
                    end
                end
            end
        end
    end
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        OUT(K)=IN(i,j);
    elseif j>=11&&j<=16
        K=K+1;
        OUT(K)=IN(i,j);
    end
elseif i==3||i==13||i==23
    if j>=1&&j<=5
        K=K+1;
        OUT(K)=IN(i,j);
    elseif j>=11&&j<=15
        K=K+1;
        OUT(K)=IN(i,j);
    end
elseif i==4||i==14||i==24
    if j>=1&&j<=4
        K=K+1;
        OUT(K)=IN(i,j);
    elseif j>=11&&j<=14
        K=K+1;
        OUT(K)=IN(i,j);
    end
elseif i==5||i==15||i==25
    if j>=1&&j<=3
        K=K+1;
        OUT(K)=IN(i,j);
    elseif j>=11&&j<=13
        K=K+1;
        OUT(K)=IN(i,j);
    end
elseif i==6||i==16||i==26
    if j>=1&&j<=2
        K=K+1;
        OUT(K)=IN(i,j);
    elseif j>=11&&j<=12
        K=K+1;
        OUT(K)=IN(i,j);
    end
elseif i==7||i==17||i==27
    if j==1
        K=K+1;
        OUT(K)=IN(i,j);
    elseif j==11
        K=K+1;
        OUT(K)=IN(i,j);
    end
end
end
end
end
elseif strcmp(args,'I')==true
    [H,W]=size(IN);
    OUT=zeros(30,20);
    K=0;
    if H==1&&W==168
        for i=1:1:30
            for j=1:1:20
                if i==1||i==11||i==21
                    if j>=1&&j<=7
                        K=K+1;
                        OUT(i,j)=IN(K);
                    elseif j>=11&&j<=17

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 Peak Signal-to-Noise Ratio. (PSNR)

```
function [OUT]=IM_PSNR(IN1,IN2)

%IM_BIN_PSNR Peak Signal-to-Noise Ratio.
%
% [OUT] = IM_BIN_PSNR(IN1,IN2)
% Return PSNR(dB) of master image and noise image.

[H1,W1]=size(IN1);
[H2,W2]=size(IN2);
SUM=0;
if (H1==H2)&&(W1==W2)
    for i=1:1:H1
        for j=1:1:W1
            SUM=SUM+(IN1(i,j)-IN2(i,j))^2;
        end
    end
    MSE=SUM/(H1*W1);
    OUT=20*log10(max(max(IN1))/sqrt(MSE));
else
    OUT=0;
end
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4 Neural Network. (Back-propagation)

4.1 Network training.

```
%TRAIN_CHAR Back-Propagation Neural Network.
% Character training program.
% Network structure.
% - 168 and 240 input layer nodes.
% - 30 hidden layer nodes.
% - 45 output layer nodes.
clear all

% Network variables.
% =====
TYPE=10; % Type if DCT matrix (5)=5x5 (10)=10x10.
if TYPE==10
    IN=168; % Input layer node.
    HN=30; % Hidden layer node. (Adjustable)
    load PTTN10.mat % DCT coefficient of all characters.
elseif TYPE==5
    IN=240;
    HN=30;
    load PTTN5.mat
end
ON=45; % Output layer node.
PT=76; % Number of pattern.
WH=rand(HN,IN)-0.5; % Hidden layer weights.
BH=rand(1,HN); % Hidden layer biases.
WO=rand(ON,HN)-0.5; % Output layer weights.
BO=rand(1,ON); % Output layer biases.
NETH=zeros(1,HN); % Hidden layer NET variables.
YH=zeros(1,HN); % Hidden layer output.
FH=zeros(1,HN); % Hidden layer Error factors.
NETO=zeros(1,ON); % Output layer NET variables.
YO=zeros(1,ON); % Output layer output.
FO=zeros(1,ON); % Output layer Error factors.
N=0.2; % Learning rate. (Adjustable)
Emax=0.0001; % Maximum error. (Adjustable)
Esum=1; % All patterns error variable.
Ep=0; % Pattern error variable.
L=0; % Loop count variable.

% Specify pattern array variables.
% =====
P=zeros(PT,IN);
P(1,:)=D01;P(46,:)=D46; % ก
P(2,:)=D02;P(47,:)=D47; % ข
P(3,:)=D03; % ค
P(4,:)=D04; % ง
P(5,:)=D05; % จ
P(6,:)=D06;P(48,:)=D48; % ฉ
P(7,:)=D07;P(49,:)=D49; % ช
P(8,:)=D08;P(50,:)=D50; % ฌ
P(9,:)=D09;P(51,:)=D51; % ฎ
P(10,:)=D10; % ฏ
P(11,:)=D11;P(52,:)=D52; % ฐ
P(12,:)=D12; % ฑ
P(13,:)=D13; % ฒ
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

P(14,:) = D14; P(53,:) = D53;      ๑  ๓
P(15,:) = D15; P(54,:) = D54;      ๑  ๓
P(16,:) = D16; P(55,:) = D55;      ๑  ๓
P(17,:) = D17;                      ๑  ๓
P(18,:) = D18;                      ๑  ๓
P(19,:) = D19;                      ๑  ๓
P(20,:) = D20;                      ๑  ๓
P(21,:) = D21; P(56,:) = D56;      ๑  ๓
P(22,:) = D22;                      ๑  ๓
P(23,:) = D23;                      ๑  ๓
P(24,:) = D24;                      ๑  ๓
P(25,:) = D25;                      ๑  ๓
P(26,:) = D26;                      ๑  ๓
P(27,:) = D27;                      ๑  ๓
P(28,:) = D28;                      ๑  ๓
P(29,:) = D29; P(57,:) = D57;      ๑  ๓
P(30,:) = D30;                      ๑  ๓
P(31,:) = D31;                      ๑  ๓
P(32,:) = D32;                      ๑  ๓
P(33,:) = D33; P(58,:) = D58;      ๑  ๓
P(34,:) = D34;                      ๑  ๓
P(35,:) = D35; P(59,:) = D59;      ๑  ๓
P(36,:) = D36;                      ๑  ๓
P(37,:) = D37; P(60,:) = D60; P(67,:) = D67; ๑  ๓ ๐
P(38,:) = D38; P(61,:) = D61; P(68,:) = D68; ๑  ๓ ๑
P(39,:) = D39; P(62,:) = D62; P(69,:) = D69; ๑  ๓ ๒
P(40,:) = D40; P(63,:) = D63; P(70,:) = D70; ๑  ๓ ๓
P(41,:) = D41; P(64,:) = D64; P(71,:) = D71; ๑  ๓ ๔
P(42,:) = D42; P(65,:) = D65; P(72,:) = D72; ๑  ๓ ๕
P(43,:) = D43; P(66,:) = D66; P(73,:) = D73; ๑  ๓ ๖
P(44,:) = D44;                      ๑  ๓
P(45,:) = D45;                      ๑  ๓
P(46,:) = D46;                      ๑  ๓
P(47,:) = D47;                      ๑  ๓
P(48,:) = D48;                      ๑  ๓
P(49,:) = D49;                      ๑  ๓
P(50,:) = D50;                      ๑  ๓
P(51,:) = D51;                      ๑  ๓
P(52,:) = D52;                      ๑  ๓
P(53,:) = D53;                      ๑  ๓
P(54,:) = D54;                      ๑  ๓
P(55,:) = D55;                      ๑  ๓
P(56,:) = D56;                      ๑  ๓
P(57,:) = D57;                      ๑  ๓
P(58,:) = D58;                      ๑  ๓
P(59,:) = D59;                      ๑  ๓
P(60,:) = D60;                      ๑  ๓
P(61,:) = D61;                      ๑  ๓
P(62,:) = D62;                      ๑  ๓
P(63,:) = D63;                      ๑  ๓
P(64,:) = D64;                      ๑  ๓
P(65,:) = D65;                      ๑  ๓
P(66,:) = D66;                      ๑  ๓
P(67,:) = D67;                      ๑  ๓
P(68,:) = D68;                      ๑  ๓
P(69,:) = D69;                      ๑  ๓
P(70,:) = D70;                      ๑  ๓
P(71,:) = D71;                      ๑  ๓
P(72,:) = D72;                      ๑  ๓
P(73,:) = D73;                      ๑  ๓
P(74,:) = D74;                      ๑  ๓
P(75,:) = D75;                      ๑  ๓
P(76,:) = D76;                      ๑  ๓

```

```

* Specify pattern targets.

```

```

* =====

```

```

T=zeros(PT,ON);
for i=1:1:ON
    T(i,i)=1;
end
T(46,:)=0;T(46,1)=1;
T(47,:)=0;T(47,2)=1;
T(48,:)=0;T(48,6)=1;
T(49,:)=0;T(49,7)=1;
T(50,:)=0;T(50,8)=1;
T(51,:)=0;T(51,9)=1;
T(52,:)=0;T(52,11)=1;
T(53,:)=0;T(53,14)=1;
T(54,:)=0;T(54,15)=1;
T(55,:)=0;T(55,16)=1;
T(56,:)=0;T(56,21)=1;
T(57,:)=0;T(57,29)=1;
T(58,:)=0;T(58,33)=1;
T(59,:)=0;T(59,35)=1;
T(60,:)=0;T(60,37)=1;
T(61,:)=0;T(61,38)=1;
T(62,:)=0;T(62,39)=1;
T(63,:)=0;T(63,40)=1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

T(64,:) = 0; T(64,41) = 1;
T(65,:) = 0; T(65,42) = 1;
T(66,:) = 0; T(66,43) = 1;
T(67,:) = 0; T(67,36) = 1;
T(68,:) = 0; T(68,37) = 1;
T(69,:) = 0; T(69,38) = 1;
T(70,:) = 0; T(70,39) = 1;
T(71,:) = 0; T(71,40) = 1;
T(72,:) = 0; T(72,41) = 1;
T(73,:) = 0; T(73,42) = 1;
T(74,:) = 0; T(74,43) = 1;
T(75,:) = 0; T(75,44) = 1;
T(76,:) = 0; T(76,45) = 1;

% Training method.
% =====
while Esum > Emax
    L=L+1;
    Esum=0;
    for p=1:1:PT % PT patterns training loop.
        Ep=0; % Clear pattern error value.
        for i=1:1:HN % Hidden layer computation per pattern.
            NETH(i)=(P(p,:)*WH(i,:))'+BH(i);
            YH(i)=1/(1+exp(-NETH(i)));
        end
        for i=1:1:ON % Output layer computation per pattern.
            NETO(i)=(YH*WO(i,:))'+BO(i);
            YO(i)=1/(1+exp(-NETO(i)));
        end
        for i=1:1:ON % Pattern error, output layer error factor.
            Ep=Ep+(0.5*(T(p,i)-YO(i))^2);
            FO(i)=YO(i)*(1-YO(i))*(T(p,i)-YO(i));
        end
        for i=1:1:HN % Hidden layer error factor computation.
            RESULT=0;
            for j=1:1:ON
                RESULT=RESULT+(FO(j)*WO(j,i));
            end
            FH(i)=YH(i)*(1-YH(i))*RESULT;
        end
        for i=1:1:ON % Output layer weights and biases update.
            WO(i,:)=WO(i,:)+(N*FO(i)*YH);
            BO(i)=BO(i)+(N*FO(i));
        end
        for i=1:1:HN % Hidden layer weights and biases update.
            WH(i,:)=WH(i,:)+(N*FH(i)*P(p,:));
            BH(i)=BH(i)+(N*FH(i));
        end
        Esum=Esum+Ep;
    end
    Esum=Esum/PT;
    fprintf('\nLoop : %.0f, overall error = %.6f',L,Esum);
    fprintf('\n');
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 Character classification (Feed forward Neural Network).

```
function [OUT,M]=CHAR_CLASSIFY(D_IN)

%CHAR_CLASSIFY Character classification function.
% Using Feed-Forward Neural Network.
% Network Structure.
% - 168 and 240 input layer nodes.
% - 30 hidden layer nodes.
% - 45 output layer nodes.

% Network variables.
% =====
TYPE=5; % Type if DCT matrix (5)=5x5 (10)=10x10.
if TYPE==10
    IN=168; % Input layer node.
    HN=30; % Hidden layer node. (Adjustable)
    ON=45; % Output layer node.
    WH=zeros(HN,IN); % Hidden layer weights.
    WO=zeros(ON,HN); % Output layer weights.
    BH=zeros(1,HN); % Hidden layer biases.
    BO=zeros(1,ON); % Output layer biases.
    load NETWORK10.mat % Load network weights and biases.
elseif TYPE==5
    IN=240;
    HN=30;
    ON=45;
    WH=zeros(HN,IN);
    WO=zeros(ON,HN);
    BH=zeros(1,HN);
    BO=zeros(1,ON);
    load NETWORK5.mat
end
ON=45; % Output layer node.
NETH=zeros(1,HN); % Hidden layer NET variables.
YH=zeros(1,HN); % Hidden layer output.
NETO=zeros(1,ON); % Output layer NET variables.
YO=zeros(1,ON); % Output layer output.
P=D_IN;

for i=1:1:HN % Hidden layer computation per pattern.
    NETH(i)=(P*WH(i,:)')+BH(i);
    YH(i)=1/(1+exp(-NETH(i)));
end
for i=1:1:ON % Output layer computation per pattern.
    NETO(i)=(YH*WO(i,:)')+BO(i);
    YO(i)=1/(1+exp(-NETO(i)));
end

% Character classification.
% =====
[M,E]=max(YO);
if E==1
    OUT='n';
elseif E==2
    OUT='u';
elseif E==3
    OUT='k';
elseif E==4
    OUT='v';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
elseif E==5
    OUT='ง';
elseif E==6
    OUT='ข';
elseif E==7
    OUT='ค';
elseif E==8
    OUT='ด';
elseif E==9
    OUT='ย';
elseif E==10
    OUT='ฉ';
elseif E==11
    OUT='ช';
elseif E==12
    OUT='ต';
elseif E==13
    OUT='ถ';
elseif E==14
    OUT='ฑ';
elseif E==15
    OUT='ท';
elseif E==16
    OUT='ธ';
elseif E==17
    OUT='ด';
elseif E==18
    OUT='น';
elseif E==19
    OUT='น';
elseif E==20
    OUT='น';
elseif E==21
    OUT='น';
elseif E==22
    OUT='น';
elseif E==23
    OUT='น';
elseif E==24
    OUT='น';
elseif E==25
    OUT='น';
elseif E==26
    OUT='น';
elseif E==27
    OUT='น';
elseif E==28
    OUT='น';
elseif E==29
    OUT='น';
elseif E==30
    OUT='น';
elseif E==31
    OUT='น';
elseif E==32
    OUT='น';
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
elseif E==33
    OUT='พ';
elseif E==34
    OUT='อ';
elseif E==35
    OUT='ค';
elseif E==36
    OUT='0';
elseif E==37
    OUT='1';
elseif E==38
    OUT='2';
elseif E==39
    OUT='3';
elseif E==40
    OUT='4';
elseif E==41
    OUT='5';
elseif E==42
    OUT='6';
elseif E==43
    OUT='7';
elseif E==44
    OUT='8';
elseif E==45
    OUT='9';
end
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5 Image Acquisition.

5.1 Load camera or tv tuner device driver.

```
function [OUT]=INIT_IMAQ

%INIT_IMAQ Start Image Acquisition tool.
%
% [OUT] = INIT_IMAQ
% Start Image Acquisition tool.

fprintf('Image Acquisition.\n\n');
dev_info=imaqhwinfo('winvideo',1);
disp(struct(dev_info));
[R,C]=size(dev_info.SupportedFormats);
disp('Device supported format(s).');
for i=1:1:C
    fprintf('    %s\n',dev_info.SupportedFormats{i});
end
OUT=videoinput('winvideo',1);
triggerconfig(OUT,'manual');
set(OUT,'FramesPerTrigger',1);
set(OUT,'TriggerRepeat',Inf);
start(OUT);
```

5.2 Capture image from camera or tv tuner device.

```
function [OUT]=IM_CAPTURE(args)

%IM_CAPTURE Capture image from camera or tv tuner device.
%
% [OUT] = IM_CAPTURE(args)
% Capture and return image to workspace.
% By args is variable name (Case-sensitive) of INIT_IMAQ.

trigger(args);
OUT=getdata(args,1);
```

6 Database connection.

6.1 Connect to database.

```
function [OUT]=DB_CONNECT
```

```
%DB_CONNECT Connect to database.
```

```
%
```

```
% [OUT] = DB_CONNECT
```

```
% By using data source ODBC. (Open Database Connection)
```

```
% Connect via TCP/IP.
```

```
DS='MySQL'; % ODBC data source name.
```

```
USER='root'; % MySQL user.
```

```
PASS='rootPassword'; % User password for MySQL.
```

```
OUT=database(DS,USER,PASS);
```

```
exec(OUT,'SET NAMES TIS620');
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้