สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การตรวจหาแผ่นป้ายทะเบียนรถยนต์

LICENSE PLATE DETECTION

นางสาวหทัยทิพย์ สิริพงศ์พันธ์
นายอธิศักดิ์ สาธุประดิษฐ์
นางสาวอัญมณี เกิดเพชร

เลขหมู่.....
เลขทะเบียน 104124
วัน,เดือน,ปี 30 ต.ค. 2552
.b 12102805
.i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมการวัดคุม
ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2551

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LICENSE PLATE DETECTION

HATHAITIP SIRIPONGPUN

ATHISAK SARTHUPRAKUL

UNYAMANEE KERDPHET

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING

DEPARTMENT OF INSTRUMENTATION ENGINEERING

FACULITY OF ENGINEERING

KING MONGKUT'S INSTITUE OF TECHNOLOGY LARDKRABANG

2008

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาควิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

.....

หัวข้อปริญญานิพนธ์ การตรวจหาแผ่นป้ายทะเบียนรถยนต์

LICENSE PLATE DETECTION

นักศึกษาผู้จัดทำ นางสาวหทัยทิพย์ สิริพงศ์พันธ์ **รหัสนักศึกษา** 48011034

นายอธิศักดิ์ สาธุประดิษฐ์ **รหัสนักศึกษา** 48011047

นางสาวอัญมณี เกิดเพชร **รหัสนักศึกษา** 48011106

ปริญญา วิศวกรรมศาสตรบัณฑิต

สาขาวิชา วิศวกรรมการวัดคุม

ปีการศึกษา 2551

อาจารย์ผู้ควบคุมปริญญานิพนธ์	ลายมือชื่อ
รศ. เกษตร์ ศิริสันติสัมฤทธิ์	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	การตรวจหาแผ่นป้ายทะเบียนรถยนต์			
	LICENSE PLATE DETECTION			
นักศึกษาผู้จัดทำ	นางสาวหทัยทิพย์	สิริพงศ์พันธ์	รหัสนักศึกษา	48011034
	นายอธิศักดิ์	สาธุประดิลภ์	รหัสนักศึกษา	48011047
	นางสาวอัญมณี	เกิดเพชร	รหัสนักศึกษา	48011106
อาจารย์ที่ปรึกษา	รศ. เกษตร	ศิริสันติสัมฤทธิ์		
ปีการศึกษา	2551			

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอวิธีการตรวจหาแผ่นป้ายทะเบียนรถยนต์ รวมถึงการรู้จำตัวเลขและพยัญชนะ โดยใช้โปรแกรม MATLAB โปรแกรมประกอบด้วย 3 ขั้นตอนหลัก คือ การค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์จากข้อมูลภาพ ที่ถ่ายจากด้านหน้าของรถยนต์ การแยกส่วนที่เป็นตัวเลขและพยัญชนะ ออกจากแผ่นป้ายทะเบียนรถยนต์ สุดท้ายคือการรู้จำตัวเลขและพยัญชนะทั้งหมดที่แยกออกมาได้ โดยใช้เทคนิค Template Matching ผลการทดลองกับภาพที่ถ่ายด้วยกล้อง Web Cam และโทรศัพท์มือถือ พบว่าสามารถแยกตัวเลขและพยัญชนะออกจากแผ่นป้ายทะเบียนได้ถูกต้อง 83% รู้จำตัวเลขและพยัญชนะได้ถูกต้อง 66%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	License Plate Detection	
Authors	Mr. Athisak	Sarthuprakul
	Miss Hathaitip	Siripongpun
	Miss Unyamanee	Kerdphet
Thesis Advisor	Assoc.Prof. Kaset	Sirisantisamrid
Year	2008	

ABSTRACT

This thesis presents a method for car license plate detection including numbers and characters recognition using MATLAB. The designed program mainly consists of three steps. First step, the license plate part is sought from input image taken from the front of car. Second step, the characters and numbers are separated from the license plate. Then, the separated numbers and characters are finally recognized by template matching technique. The experimental results with the images taken by web cam and mobile phone cameras found that the proposed method gives 83% and 66% correction of separating numbers and characters from license plate, and the numbers and characters recognition, respectively.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้เพราะได้รับความเมตตาจาก รศ. เกษตร์ ศิริสันติ-สัมฤทธิ์ ที่ได้ให้คำแนะนำแก่ผู้ทำวิจัยมาตลอด ซึ่งผู้วิจัยรู้สึกซาบซึ้งและขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณ อ.ธีรวัฒน์ เทพมณี ที่ให้คำแนะนำเกี่ยวกับการทำปริญญานิพนธ์ฉบับนี้ และที่ลืมเสียมิได้ก็คือ ขอกราบขอบพระคุณคุณพ่อ คุณแม่ อันเป็นที่รักยิ่ง ที่ให้การสนับสนุนและเป็นกำลังใจในการทำปริญญานิพนธ์ฉบับนี้

คุณค่าและประโยชน์อันพึงมีจากปริญญานิพนธ์ฉบับนี้ ผู้วิจัยขอมอบแด่ผู้มีพระคุณทุกท่าน

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญตาราง.....VI

สารบัญภาพ.....IX

บทที่ 1 บทนำ.....1

1.1 ที่มาและความสำคัญของปริญญานิพนธ์.....1

1.2 วัตถุประสงค์ของปริญญานิพนธ์.....2

1.3 ขอบเขตของปริญญานิพนธ์.....2

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....3

2.1 การประมวลผลภาพเชิงตัวเลข (Digital Image Processing).....3

2.2.1 การแทนภาพด้วยข้อมูลแบบดิจิตอล.....3

2.1.2 ชนิดของข้อมูลภาพที่ใช้ในการประมวลผลภาพทางดิจิตอล.....4

2.1.2.1 RGB Image.....4

2.1.2.2 ภาพระดับเทา (Gray Scale Image).....5

2.1.2.3 ภาพสองระดับ (Binary Image).....7

2.2 การแบ่งส่วนภาพ (Image Segmentation).....10

2.2.1 การแบ่งส่วนภาพด้วยการพิจารณาความต่อเนื่องของข้อมูล.....10

2.2.2 การแยกภาพด้วยวิธี Region Labeling.....10

2.3 เทคนิค Template Matching.....11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่ 3 การออกแบบโปรแกรมตรวจหาแผ่นป้ายทะเบียนรถยนต์.....13

3.1 การประมวลผลภาพดิจิตอล.....23

3.2 การแบ่งส่วนที่เป็นตัวเลขและพยัญชนะออกมาจาก
แผ่นป้ายทะเบียนรถยนต์.....44

3.3 การรู้จำตัวเลขและพยัญชนะ.....62

3.3.1 การจัดทำตัวเลขและพยัญชนะทั้งหมด
ของชุด Template.....62

3.3.2 ขั้นตอนการรู้จำตัวเลขและพยัญชนะ.....64

บทที่ 4 การทดลองและผลการทดลอง.....78

4.1 การแยกตัวเลขและพยัญชนะออกจากแผ่นป้ายทะเบียน.....78

4.2 การรู้จำตัวเลขและพยัญชนะ.....92

บทที่ 5 สรุปและวิจารณ์ผลการทดลอง.....105

5.1 สรุปผลการทดลอง.....105

5.2 ข้อเสนอแนะ.....106

5.3 วิจารณ์ผลการทดลอง.....106

บรรณานุกรม.....108

ภาคผนวก.....109

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ **หน้า**

4.1 สรุปผลการทดลองที่ได้จากตอนที่ 4.1.....91
4.2 สรุปผลการทดลองที่ได้จากตอนที่ 4.2.....104

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

ภาพที่ **หน้า**

1.1 ภาพการบันทึกภาพด้านหน้าของรถยนต์.....1

1.2 ภาพด้านหน้ารถยนต์พร้อมป้ายทะเบียนที่ได้.....2

2.1 การแสดงข้อมูลภาพดิจิตอลในรูปของเมตริกซ์.....4

2.2 การแทนข้อมูลภาพดิจิตอลด้วยฟังชันก์.....4

2.3 โมเดลสี RGB 5

2.4 ภาพสี RGB 6

2.5 ภาพ Gray Scale 7

2.6 ภาพ Binary Image 9

2.7 การทำ Binary Image ด้วยค่าเทรชโฮลที่เหมาะสม.....9

2.8 การทำ Binary Image ด้วยค่าเทรชโฮลที่ไม่เหมาะสม.....10

2.9 การพิจารณาความต่อเนื่องกันแบบ 4 จุด.....11

2.10 การพิจารณาความต่อเนื่องกันแบบ 8 จุด.....11

3.1 ภาพ RGB ส่วนด้านหน้ารถยนต์ที่รับเข้ามา.....15

3.2 ภาพ Gray Scale ที่แปลงมาจากภาพ RGB 16

3.3 ภาพ Binary ที่แปลงมาจากภาพ Gray Scale.....17

3.4 ภาพ Binary ที่กำจัดส่วนที่ติดกับขอบออกไปแล้ว.....18

3.5 ภาพที่กำจัดสิ่งรบกวนออกไปแล้ว.....19

3.6 การกำหนดหมายเลขหรือทำการ Label ให้กับกลุ่มพิกเซลสีขาว.....21

3.7 ภาพที่ได้จากการนำคู่พิกัดของกลุ่มพิกเซลสีขาวที่มีพื้นที่มากที่สุดจากภาพที่ 3.6 ไปคัดลอกจากภาพ RGB ที่รับเข้ามาในขั้นตอนแรก.....23

3.8 ตัวอย่างภาพที่เกิดจากการนำคู่พิกัดของกลุ่มพิกเซลสีขาวที่มีพื้นที่มากสุดไปคัดลอกมาจากภาพ RGB แล้วได้ส่วนที่ไม่ใช่แผ่นป้ายทะเบียนออกมา.....27

3.9 (ก) ภาพที่ได้จากการนำภาพ Binary ไปกำจัดกลุ่มพิกเซลสีขาวที่มีอัตราส่วนความกว้างต่อความยาวและความสูงที่คาดว่าไม่น่าจะเป็นส่วนของแผ่นป้ายทะเบียนรถยนต์ออกไป.....29

(ข) ภาพที่ได้จากการนำคู่พิกัดของกลุ่มพิกเซลสีขาวที่มีพื้นที่มากสุดจากภาพ (ก) ไปคัดลอกมาจากภาพ RGB.....29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

ภาพที่ **หน้า**

3.10 ตัวอย่างการแปลงภาพ RGB เป็นภาพ Binary โดยใช้ค่าเทรชโฮลอัตโนมัติ แล้วทำให้ส่วนที่เป็นแผ่นป้ายทะเบียนถูกลบออกไป 33

3.11 (ก) ภาพ Binary ที่เกิดจากการนำภาพ RGB จากภาพที่ 3.10 มาแปลงเป็นภาพ Binary โดยใช้ค่าเทรชโฮลที่เกิดจากการนำค่าความเข้มสีของทุกพิกเซลมาเฉลี่ยกันแล้วบวกด้วยค่าคงที่ค่าหนึ่ง 34

(ข) ภาพแผ่นป้ายทะเบียนที่ได้จากการนำคู่พิกัดของกลุ่มพิกเซลสีขาวที่มีพื้นที่มากสุด จากภาพ (ก) ไปคัดลอกมาจากภาพ RGB.....34

3.12 ภาพ Inverse Binary ของภาพที่ 3.7.....35

3.13 ภาพที่ได้จากการกำจัดส่วนที่เป็นชื่อจังหวัดออกไปจากภาพที่ 3.12.....36

3.14 ภาพที่ได้จากการกำจัดกลุ่มพิกเซลสีขาวที่ติดกับขอบภาพและสิ่งรบกวนต่าง ๆ ที่อยู่บนแผ่นป้ายออกไปจากภาพที่ 3.13.....38

3.15 ตัวอย่างการหมุนกลุ่มตัวเลขและตัวพยัญชนะที่มีความเอียงให้กลับมาอยู่ใน แนวราบมากขึ้น.....43

3.16 แสดงแถวที่เป็นแกนอ้างอิงในการตัดขอบบนและขอบล่างของกลุ่มตัวเลขและตัวพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์.....44

3.17 ภาพที่ได้จากการใช้แกนอ้างอิงแนวนอนที่หาได้ไปคัดลอกส่วนของภาพจากภาพที่ 3.16.....45

3.18 ขอบเขตด้านซ้าย (ดูเส้นประสีชมพู) และด้านขวา (ดูที่เส้นประสีม่วง) ของกลุ่มพิกเซลสีขาวที่ถูก Label ด้วยหมายเลขที่แตกต่างกัน 48

3.19 Inverse Binary ของภาพที่ 3.17.....56

3.20 ตัวเลขและพยัญชนะที่แยกออกมาจากแผ่นป้ายทะเบียน โดยเรียงลำดับให้เหมือนกับบนแผ่นป้ายทะเบียน.....62

3.21 ตัวเลขและพยัญชนะต้นแบบหรือชุด Template ทั้งหมดที่ใช้ในการทดลอง.....64

3.22 ภาพ Input ตัวแรกที่เป็นภาพ Binary 64

3.23 ผลที่ได้จากการนำภาพตัว ฐ ไปประมวลผลในส่วนการรู้จำตัวเลขและพยัญชนะ 75

3.24 หน้าต่าง Command Window ที่เป็นผลมาจากการนำข้อมูลภาพตัวเลขและพยัญชนะทุกตัวที่อยู่บนแผ่นป้ายทะเบียนรถยนต์มาเข้าสู่การรู้จำตัวเลขและพยัญชนะ 76

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

ภาพที่ **หน้า**

4.1 ภาพถ่ายส่วนด้านหน้ารถยนต์แบบ RGB 77

4.2 ภาพที่ได้จากการแปลงภาพ RGB ให้เป็นภาพ Gray Scale.....77

4.3 ภาพที่ได้จากการแปลงภาพ Gray Scale ให้เป็นภาพ Binary.....78

4.4 ภาพผลลัพธ์ที่ได้จากการกำจัดกลุ่มพิกเซลสีขาวที่ติดกับขอบออกไป.....78

4.5 ภาพผลลัพธ์ที่ได้จากการทำให้ภาพคมชัดขึ้นและกำจัดสิ่งรบกวนออกไป 78

4.6 ภาพ RGB ที่ได้จากการนำคู่พิกัดของกลุ่มพิกเซลสีขาวที่ใหญ่ที่จากภาพที่ 4.6
ไปคัดลอกส่วนของภาพจากภาพ RGB จากขั้นตอนที่ 1.....79

4.7 ภาพ Inverse Binary ของภาพที่ 4.8.....79

4.8 ภาพผลลัพธ์ที่ได้จากการนำภาพที่ 4.7 ไปกำจัดส่วนที่เป็นชื่อจังหวัด ลวดลายต่าง ๆ
ที่อยู่บนแผ่นป้ายทะเบียน และกลุ่มพิกเซลสีขาวที่ติดกับขอบออกไป.....79

4.9 ภาพผลลัพธ์ที่ได้จากการหมุนกลุ่มตัวเลขและพยัญชนะ
ให้กลับมาอยู่ในแนวราบมากขึ้น.....80

4.10 ภาพผลลัพธ์ที่ได้จากการนำภาพที่ 4.9 ไปตัดขอบบนและขอบล่าง
ให้กับกลุ่มตัวเลขและพยัญชนะ.....80

4.11 ผลที่ได้จากการนำ Column ที่เป็นขอบซ้ายและขอบขวา
ของตัวเลขและพยัญชนะ ไปเก็บในรูปของเมตริกซ์.....80

4.12 ผลจากโปรแกรมที่ได้จากการนำค่า Column ที่เป็นขอบซ้ายและขอบขวา
ของตัวเลขและพยัญชนะมาเรียงลำดับจากน้อยไปหามาก.....83

4.13 ภาพ Inverse Binary ของภาพที่ 4.9.....84

4.14 ภาพตัวเลขและพยัญชนะที่แยกออกมาได้จากภาพที่ 4.12 ในขั้นตอนที่ 12.....84

4.15 (ก -ช) ตัวอย่างภาพผลการทดลองที่โปรแกรมสามารถแยกตัวเลขและพยัญชนะ
ออกจากแผ่นป้ายได้ครบทุกตัว.....85

4.16 (ก–จ) ตัวอย่างภาพผลการทดลองที่โปรแกรมไม่สามารถแยกตัวเลขและพยัญชนะ
ออกจากแผ่นป้ายได้ครบทุกตัว 88

4.17 (ก-ญ) ตัวอย่างผลการทดลองที่โปรแกรมทั้งสองส่วนสามารถทำการรู้จำ
ตัวเลขและพยัญชนะที่อยู่บนแผ่นป้ายทะเบียนได้ถูกต้อง.....92

4.18 (ก-จ) ตัวอย่างภาพผลการทดลองที่โปรแกรมไม่สามารถทำการรู้จำตัวเลข
และพยัญชนะที่อยู่บนแผ่นป้ายทะเบียนได้ถูกต้องทุกตัว.....100

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

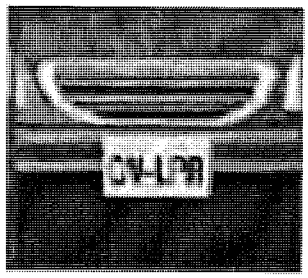
1.1 ที่มาและความสำคัญของปริญญานิพนธ์

ในการจอดรถตามสถานที่ต่าง ๆ โดยปกติแล้วจะมีเจ้าหน้าที่คอยแจกบัตรที่จอดรถ คิดค่าจอดรถ บันทึกเวลาที่รถเข้า-ออก ซึ่งการทำงานโดยมนุษย์อาจจะเกิดข้อผิดพลาดขึ้นได้ อันเป็นผลเนื่องจากความเหน็ดเหนื่อย เบื่อหน่ายเนื่องจากการทำงานติดต่อกันเป็นเวลาหลายชั่วโมง จึงทำให้เกิดแนวคิดที่จะแก้ปัญหาเหล่านี้ ด้วยการนำโปรแกรมการตรวจหาแผ่นป้ายทะเบียนรถยนต์มาใช้ โดยใช้วิธีการถ่ายภาพส่วนด้านหน้าของรถยนต์ และใช้โปรแกรมช่วยในการวิเคราะห์ภาพเพื่อค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ และสามารถรู้จำตัวเลขและพยัญชนะที่อยู่บนแผ่นป้ายทะเบียนรถยนต์ได้

นอกจากนี้ โปรแกรมยังสามารถนำไปพัฒนาต่อและประยุกต์ใช้งานในด้านอื่นๆได้อีกด้วย อาทิเช่น ใช้ในระบบรักษาความปลอดภัยของสถาบันการศึกษา เพื่อตรวจการเข้า-ออกของรถเก็บเป็นฐานข้อมูลหากมีการสูญหายในทรัพย์สินของสถาบันการศึกษา เป็นต้น จากภาพที่ 1.1 แสดงส่วนหนึ่งของระบบการรักษาความปลอดภัยในลานจอดรถยนต์ เพื่อตรวจสอบการเข้าออกของรถยนต์ เมื่อรถยนต์จอดที่ตำแหน่งที่กำหนดไว้ ซึ่งมีไม้กั้นขวางรถอยู่แล้ว กล้องจะเก็บภาพที่ด้านหน้าของรถยนต์ ดังแสดงในภาพที่ 1.2 และตรวจหาแผ่นป้ายทะเบียนพร้อมทั้งรู้จำตัวเลขและพยัญชนะที่อยู่บนแผ่นป้ายทะเบียน เมื่อตรวจพบว่าเป็นรถยนต์ที่อนุญาตให้ผ่านเข้ามาได้ ไม้กั้นจึงจะเปิดให้รถยนต์เข้ามา

ภาพที่ 1.1 ภาพการบันทึกภาพด้านหน้าของรถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 1.2 ภาพด้านหน้ารถยนต์พร้อมป้ายทะเบียนที่ได้

1.2 วัตถุประสงค์ของปริญญานิพนธ์

สร้างโปรแกรมที่สามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ จากภาพถ่ายส่วนด้านหน้าของรถยนต์ และนำตัวเลขและตัวอักษรทั้งหมดที่อยู่บนแผ่นป้ายทะเบียนไปประมวลผลเพื่อทำการรู้จำแผ่นป้ายทะเบียน

1.3 ขอบเขตของปริญญานิพนธ์

1. สามารถรู้จำเลขทะเบียนรถยนต์ได้ (ก – ฮ และ 0 – 9) ไม่รวมส่วนชื่อจังหวัด
2. ป้ายทะเบียนรถยนต์เป็นป้ายชนิดใหม่เท่านั้นและไม่รวมป้ายทะเบียนแบบพิเศษ เช่น ป้ายรถทหาร หรือป้ายที่มีพื้นหลังแบบต่างๆ เป็นต้น
3. ข้อมูลภาพที่รับมาจากกล้องเป็นภาพถ่ายหน้าตรง มีการตั้งกล้องทำมุมไม่เกิน 30 องศากับแนวตั้ง
4. ข้อมูลภาพมีขนาดความละเอียด 640 x 480 พิกเซล
5. ข้อมูลภาพถ่ายในสภาวะแสงปกติ และมีระยะการถ่ายภาพประมาณ 1 เมตร
6. แผ่นป้ายทะเบียนต้องไม่สกปรกหรือชำรุดเกินไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

ในบทนี้จะกล่าวถึงทฤษฎีต่าง ๆ ที่ใช้ในขั้นตอนการค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์จากข้อมูลภาพดิจิตอล การแยกตัวเลขและพยัญชนะออกมาจากแผ่นป้ายทะเบียนรถยนต์ ซึ่งทั้งสองขั้นตอนนี้ จะใช้ทฤษฎีของการประมวลผลภาพทางดิจิตอล (Digital Image Processing) สำหรับขั้นตอนการรู้จำตัวเลขและพยัญชนะจะใช้เทคนิค Template Matching

2.1 การประมวลผลภาพดิจิตอล (Digital Image Processing)

ภาพดิจิตอล (Digital Image) หมายถึงภาพที่เก็บอยู่ในรูปแบบดิจิตอล ซึ่งภาพที่เรามองเห็นด้วยสายตาทั่วไปนั้นเป็นภาพในลักษณะสามมิติ คือมีมิติของความกว้าง ความยาว และความลึก ส่วนภาพที่อยู่ในโทรทัศน์ หรือคอมพิวเตอร์จะเป็นการแปลงภาพจากสามมิติมาเป็นภาพสองมิติ โดยการแปลงสัญญาณไฟฟ้าในรูปแบบ Analog เช่น ในกล้องวิดีโอ เซนเซอร์ที่อยู่ในกล้องจะทำการสแกน วัดผลรวมความเข้มแสงที่จุดต่าง ๆ ไปตามแนวสแกนที่เรียกว่า Raster Scan ซึ่งการสแกนแบบนี้จะมีทิศทางจากบนลงล่างและจากซ้ายไปขวา เช่นเดียวกับเครื่องรับภาพวิดีโอ จะรับภาพที่ได้มาจากกล้องวิดีโอและแสดงผลเริ่มจากบนลงล่างและจากซ้ายไปขวาเช่นเดียวกัน

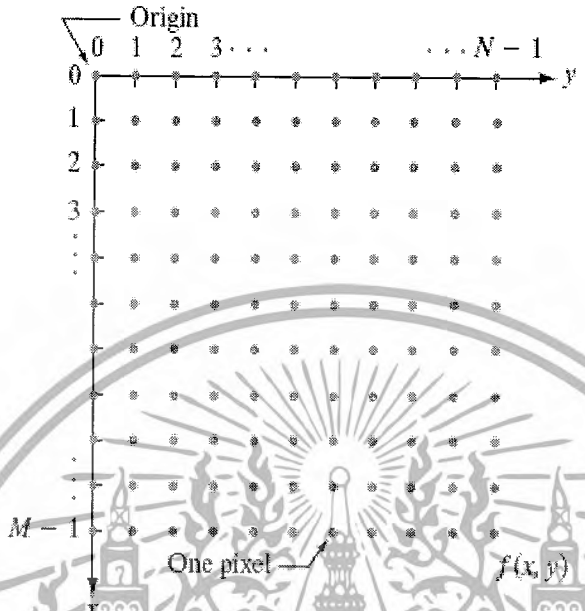
การประมวลผลภาพดิจิตอล (Digital Image Processing) หมายถึงการเรียกใช้ขั้นตอนหรือกรรมวิธีใด ๆ มากระทำกับภาพดิจิตอล โดยมีวัตถุประสงค์เพื่อปรับปรุงคุณภาพของภาพเพื่อให้ได้ภาพใหม่ที่มีคุณสมบัติตามที่ต้องการ เช่นมีความคมชัด ประหยัดพื้นที่ในการเก็บข้อมูล หรือใช้สำหรับการประมวลผลระดับสูง เช่นการจดจำรูปร่างลักษณะของสิ่งของ เป็นต้น

2.1.1 การแทนภาพด้วยข้อมูลแบบดิจิตอล

ข้อมูลภาพแบบดิจิตอลเป็นภาพที่ถูกดัดแปลงมาจากภาพแบบต่อเนื่อง หรือ Analog Image ให้อยู่ในรูปตัวเลข หรือ Digital Image ด้วยวิธีการ Digitization โดยภาพ Analog Image จะถูกแบ่งเป็นพื้นที่สี่เหลี่ยมเล็ก ๆ ที่เรียกว่า พิกเซล (Pixel) ซึ่งในแต่ละพิกเซลจะใช้(x,y) ในการระบุตำแหน่ง การแสดงข้อมูลภาพดิจิตอลสามารถอธิบายได้ในรูปของเมตริกซ์ (Matrix) ขนาด (M*N) ซึ่ง M และ N เป็นความสูงและความกว้างของภาพดิจิตอลตามลำดับ และให้จุดต่างๆที่อยู่ในเมตริกซ์เป็นจุดที่พิกัด(x,y) ใด ๆ เป็นส่วนประกอบของภาพ เมื่อเปรียบเทียบระหว่างภาพและ Pixel Matrix ดังภาพที่ 2.1 จะเห็นว่าจุดกำเนิดของภาพจะอยู่ที่มุมบนซ้าย ซึ่งเป็นลักษณะการประมวลผลภาพในกราฟฟิก (Graphic)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอมพิวเตอร์ที่กล่าวไปแล้วข้างต้น ส่วนภาพที่ 2.2 เป็นการแทนภาพดิจิตอลด้วยฟังก์ชั่น f(x,y) ซึ่ง M และ N คือความสูงและความกว้างของภาพเป็นพิกเซลตามลำดับ



ภาพที่ 2.1 การแสดงข้อมูลภาพดิจิตอลในรูปของเมตริกซ์

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & \cdots & f(1,N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1,N-1) \end{bmatrix}.$$

ภาพที่ 2.2 การแทนข้อมูลภาพดิจิตอลด้วยฟังชันก์

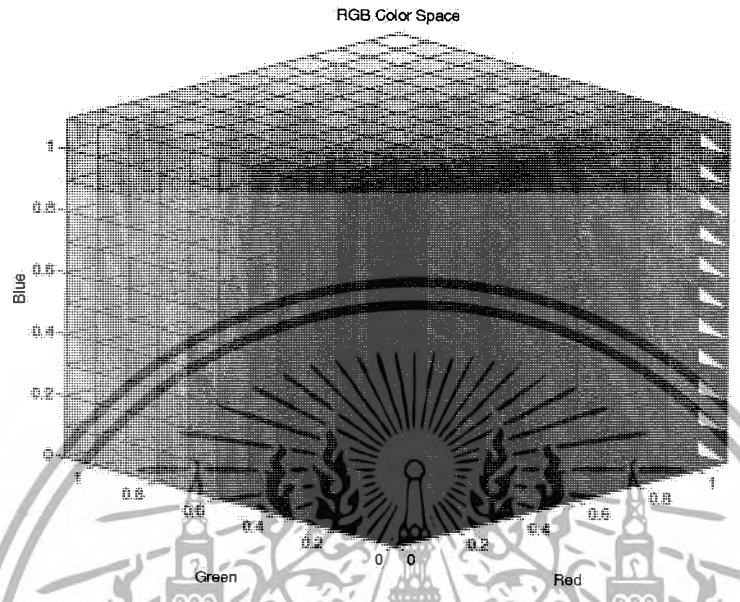
2.1.2 ชนิดของข้อมูลภาพที่ใช้ในการประมวลผลภาพทางดิจิตอล

2.1.2.1 RGB Image

RGB Image หรือภาพสี เป็นภาพที่ประกอบไปด้วย 3 แม่สีหลัก ได้แก่ สีแดง สีเขียว และสีน้ำเงิน ถ้านำแต่ละแม่สีมาพล็อตกราฟในระบบพิกัด Color Space ซึ่งเป็นระบบพิกัดแกน x, y และ z ดังแสดงในภาพที่ 2.3 โดยแต่ละสีมีค่าความเข้มแสงตั้งแต่ 0 ถึง 1 โดย 0 แสดงถึงค่าความมืด และ 1 แสดงถึงค่าความสว่าง จะทำให้ได้การผสมสีทางแสงหรือการบวกแม่สีเข้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

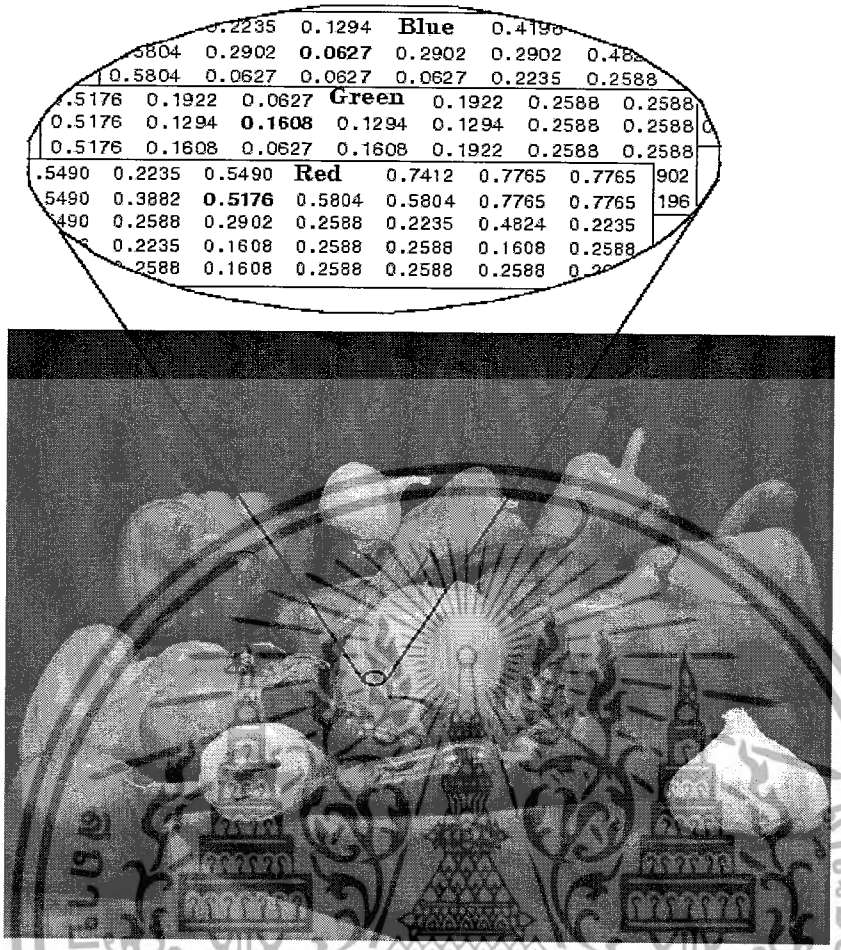
ด้วยกัน ซึ่งจากภาพนี้จะเห็นได้ว่าในแต่ละแกน เมื่อค่าความเข้มแสงมีค่าเข้าใกล้ 1 มากเท่าไหร่ แม่สีนั้นก็จะยิ่งมีความสว่างมากขึ้น



ภาพที่ 2.3 โมเดลสี RGB

จากภาพที่ 2.4 เป็นตัวอย่างของภาพ RGB ซึ่งวงกลมใหญ่ที่แสดงค่าความเข้มสีของแต่ละแม่สีนั้นขยายมาจากส่วนของภาพที่ถูกวงกลมไว้ในภาพ RGB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.4 ภาพสี RGB

2.1.2.2 ภาพระดับเทา (Gray Scale Image)

ภาพระดับเทา หมายถึงภาพที่แต่ละพิกเซล มีค่าความเข้มสีเป็นค่าเฉลี่ยของค่าแม่สีทั้ง 3 ในพิกเซลนั้น คือภาพสีในหนึ่งพิกเซลนั้นจะประกอบไปด้วย 3 แม่สีผสมกัน ได้แก่ สีแดง สีเขียว และสีน้ำเงิน ซึ่งมี 3 ค่าใน 1 พิกเซล เมื่อเป็นเช่นนี้จะทำให้ยากต่อการนำไปประมวลผล ดังนั้นในการนำไปประมวลผล จึงต้องทำให้ภาพมีความเข้มสีในระดับเดียวกันก่อนด้วยการทำให้เป็นภาพระดับเทา ซึ่ง Algorithm ที่นิยมใช้ในการทำภาพระดับเทานั้นก็คือการเฉลี่ยของค่าแม่สีทั้ง 3 ใน 1 พิกเซล เขียนเป็นสมการได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

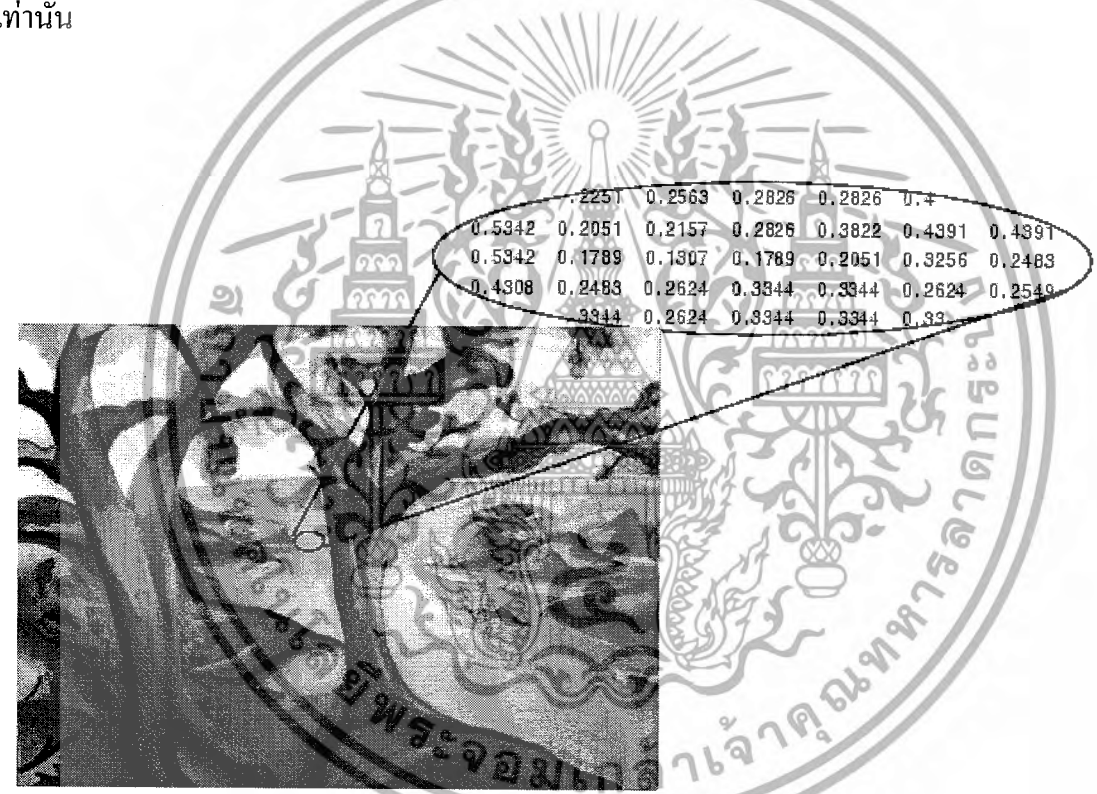
$$\text{Gray Scale} = ((0.299 \times R_S) + (0.587 \times G_S) + (0.114 \times B_S)) \quad (2.1)$$

โดย R_S คือ ค่าอินพุตพิกเซลสีแดง

G_S " ค่าอินพุตพิกเซลสีเขียว

B_S " ค่าอินพุตพิกเซลสีน้ำเงิน

จากสมการที่ 2.1 จะเห็นได้ว่าค่า Gray Scale สามารถมีค่าได้ตั้งแต่ 0 ถึง 1 ดังแสดงในภาพที่ 2.5 โดยที่ 0 หมายถึงพิกเซลสีดำ และ 1 หมายถึงพิกเซลสีขาว นั่นแสดงว่าถ้าพิกเซลไหนมีค่า Gray Scale เข้าใกล้ 1 มาก เท่าไหร่ พิกเซลนั้นก็จะยิ่งมีความสว่างหรือมีสีที่ใกล้เคียงสีขาวมากเท่านั้น



ภาพที่ 2.5 ภาพ Gray Scale

2.1.2.3 ภาพสองระดับ (Binary Image)

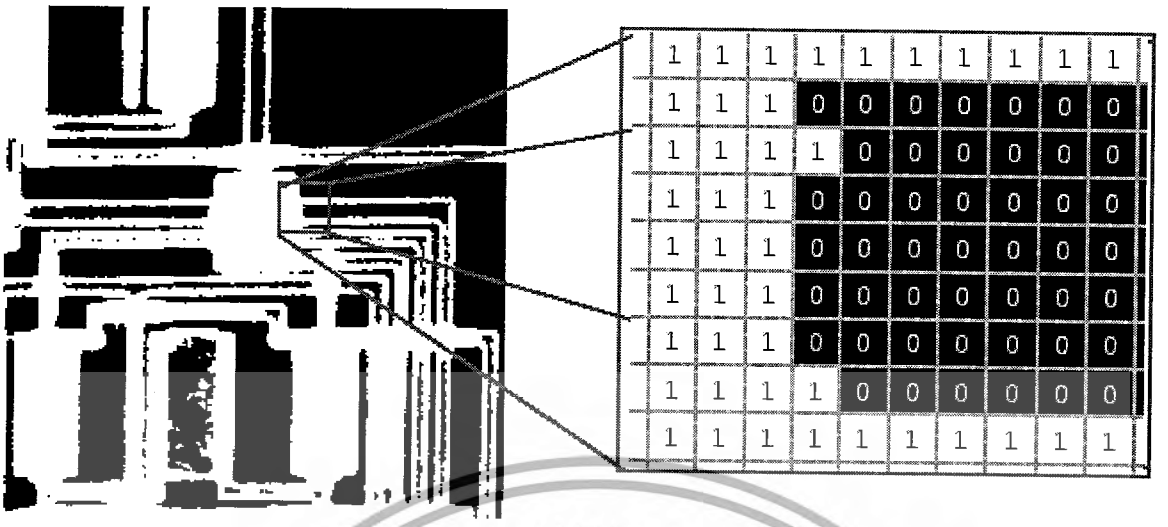
ภาพสองระดับ หมายถึงภาพที่มีระดับความเข้มของสีเพียง 2 ระดับ นั่นคือแต่ละพิกเซลสามารถมีค่าได้ 2 ค่าเท่านั้น คือ 1 และ 0 ซึ่ง 1 จะหมายถึงพิกเซลสีขาว ส่วน 0 จะหมายถึงพิกเซลสีดำ ดังแสดงในภาพที่ 2.6 ซึ่งในการประมวลผลภาพทางดิจิตอลโดยทั่วไปนั้น นิยมแปลงข้อมูลภาพหลายระดับ คือภาพแบบระดับเทาและ RGB ให้เป็นภาพแบบ Binary เพื่อเป็นการลดเนื้อที่เก็บข้อมูลภาพให้เหลือเพียง 1 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

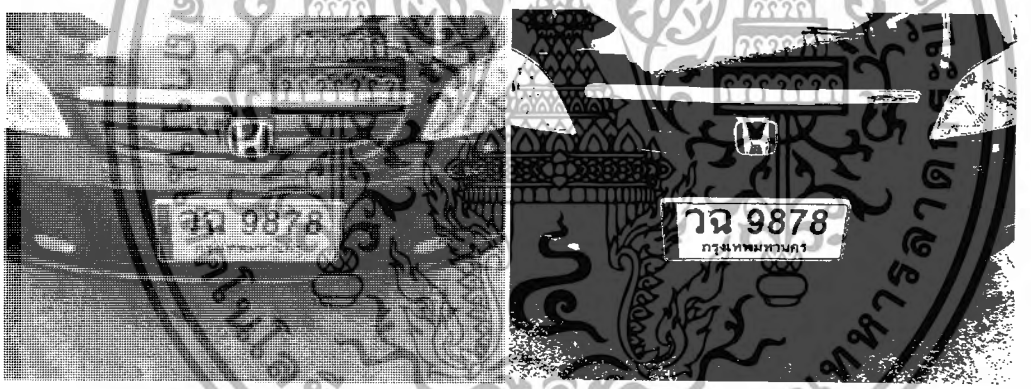
การสร้างภาพ Binary สามารถทำได้โดยการใช้เทคนิคการทำเทรชโฮล (Threshold Techniques) ซึ่งในการพิจารณาว่าพิกเซลไหนของข้อมูลภาพ ควรจะเป็นพิกเซลสีขาวหรือสีดำนั้น จะกระทำโดยการเปรียบเทียบระหว่างพิกเซลนั้น ๆ กับค่าคงที่ค่าหนึ่ง คือ ค่าเทรชโฮล ซึ่งจะมีค่าตั้งแต่ 0 ถึง 255 ถ้าหากพิกเซลไหนมีค่าน้อยกว่าค่าเทรชโฮล ก็ให้เปลี่ยนพิกเซลนั้นเป็นพิกเซลที่มีค่าเป็น 0 หรือเป็นพิกเซลสีดำ แต่ถ้าพิกเซลไหนมีค่ามากกว่าค่าเทรชโฮล ก็ให้เปลี่ยนพิกเซลนั้นเป็นพิกเซลที่มีค่าเป็น 1 หรือเป็นสีขาว

การเลือกค่าเทรชโฮลเพื่อให้ได้ภาพสองระดับที่ดีนั้น จะต้องเลือกค่าเทรชโฮลให้เหมาะสม ถ้าเลือกค่าไม่เหมาะสม เช่นเลือกค่าเทรชโฮลที่สูงหรือต่ำจนเกินไป รายละเอียดของภาพที่ได้ก็จะยากต่อการนำไปวิเคราะห์เนื่องจากภาพขาดความคมชัด และอาจจะเกิดสิ่งรบกวน (Noise) ขึ้นบนภาพได้ อย่างเช่นในงานวิจัยนี้ต้องการค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ ดังนั้นในการแปลงภาพถ่ายส่วนด้านหน้ารถยนต์ที่โปรแกรมรับเข้ามา ซึ่งเป็นภาพ RGB ให้เป็นภาพ Binary นั้น โปรแกรมจะสามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียน ได้ก็ต่อเมื่อส่วนที่เป็นแผ่นป้ายทะเบียนที่อยู่ในภาพ Binary ไม่ถูกลบออกไปหรือกลายเป็นพิกเซลสีดำ จากตัวอย่างภาพที่ 2.7 (ก) เมื่อแปลงภาพ RGB นี้ให้เป็นภาพ Binary โดยใช้ค่าเทรชโฮลที่เหมาะสม ภาพผลลัพธ์ที่ได้จะเป็นภาพ 2.7 (ข) ซึ่งจะเห็นได้ว่าส่วนที่เป็นแผ่นป้ายทะเบียนไม่ถูกลบออกไปและมีขอบที่ชัดเจน เมื่อนำภาพ Binary นี้ไปประมวลผลในการค้นหาส่วนที่เป็นแผ่นป้ายทะเบียน โปรแกรมก็จะสามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนจากภาพ Binary นี้ได้ แต่ในทางตรงกันข้าม ถ้านำภาพ RGB 2.7 (ก) ไปแปลงเป็นภาพ Binary โดยใช้ค่าเทรชโฮลที่ไม่เหมาะสม คือใช้ค่าเทรชโฮลที่ต่ำเกินไปหรือสูงเกินไป จะทำให้ได้ภาพ Binary ดังภาพที่ 2.8 (ก) และ (ข) ตามลำดับ ซึ่งจากภาพ Binary ทั้งสองนี้จะเห็นได้ว่าส่วนที่เป็นแผ่นป้ายทะเบียนมีขอบไม่ชัดเจนและส่วนที่เป็นแผ่นป้ายทะเบียนถูกลบออกไป ซึ่งถ้านำภาพ Binary ทั้งสองนี้ไปประมวลผลในการค้นหาส่วนที่เป็นแผ่นป้ายทะเบียน โปรแกรมก็อาจจะไม่สามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



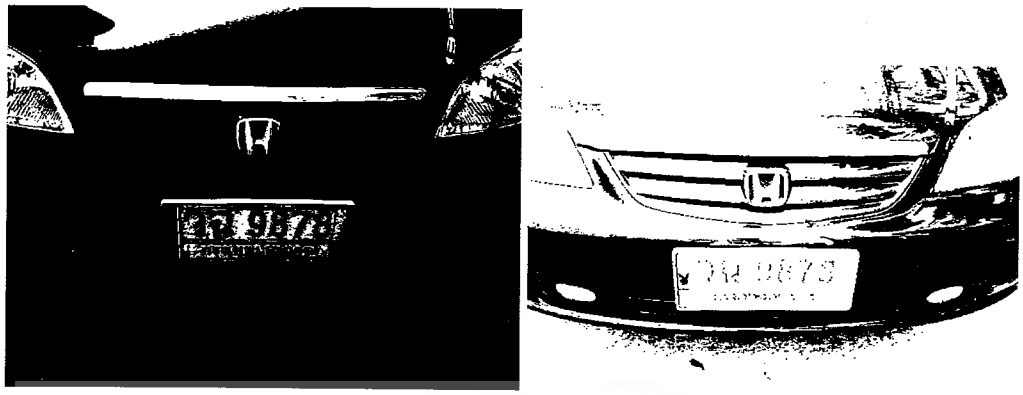
ภาพที่ 2.6 ภาพ Binary Image



(ก) (ข)

ภาพที่ 2.7 การทำ Binary Image ด้วยค่าเทรชโฮลที่เหมาะสม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) (ข)

ภาพที่ 2.8 การทำ Binary Image ด้วยค่าเทรชโฮลที่ไม่เหมาะสม

2.2 การแบ่งส่วนภาพ (Image Segmentation)

กระบวนการสำคัญอีกขั้นตอนหนึ่งในการประมวลผลเบื้องต้น ก่อนจะนำไปสู่ขั้นตอนการรู้จำคือกระบวนการแยกวัตถุออกจากพื้นหลัง (Segmentation) ในปริญญานิพนธ์นี้ จะเป็นการแยกส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ออกจากภาพ Binary และเมื่อสามารถแยกส่วนที่เป็นแผ่นป้ายทะเบียนออกจากภาพ Binary ได้แล้ว ขั้นตอนต่อมาก็จะเป็นการแยกส่วนที่เป็นตัวเลขและพยัญชนะทั้งหมดที่อยู่บนแผ่นป้ายทะเบียนออกมา โดยแยกออกมาทีละตัวเพื่อนำไปสู่กระบวนการรู้จำ ซึ่งสามารถประมวลผลได้ทีละหนึ่งตัวเท่านั้น

2.2.1 การแบ่งส่วนภาพด้วยการพิจารณาความต่อเนื่องของข้อมูล

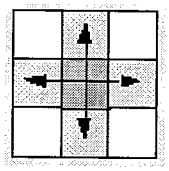
หลักการ คือหาพิกเซลที่เป็น 0 หรือพิกเซลสีดำที่ต่อเนื่องกันตลอดทั้งแนวตั้งและแนวนอน ซึ่งจะทำให้ได้ขนาดของกรอบ (Block) ของวัตถุที่มีอยู่ในข้อมูลภาพที่มีขนาดแตกต่างกัน จากนั้นจะพิจารณาเลือกขนาดของกรอบที่ต้องการ จากความแตกต่างของจำนวนพิกเซล ความสูง ความกว้าง และตำแหน่ง เป็นต้น ซึ่งจะได้กรอบของภาพที่ต้องการพิจารณา

2.2.2 การแยกภาพด้วยวิธี Region Labeling

วิธีการแบ่งส่วนภาพแบบ Region Labeling นี้เป็นการกำหนดหมายเลขให้กับกลุ่มพิกเซลสีขาวหรือเรียกว่า ทำการ Label ให้กับกลุ่มพิกเซลสีขาว ซึ่งถ้ามีพิกเซลสีขาวอยู่ติด ๆ กันเป็นกลุ่มก้อนบริเวณเดียวกัน โดยที่ไม่ถูกแยกออกจากกันด้วยพิกเซลสีดำ กลุ่มพิกเซลสีขาวนั้นก็จะมีหมายเลข Label ที่เหมือนกัน ส่วนกลุ่มของพิกเซลสีขาวกลุ่มอื่น ๆ ก็จะกระทำในลักษณะเดียวกัน โดยจะกำหนดหมายเลข Label ที่แตกต่างออกไป ซึ่งการพิจารณาความต่อเนื่องของกลุ่มพิกเซลสีขาวมีหลักการพิจารณาดังต่อไปนี้

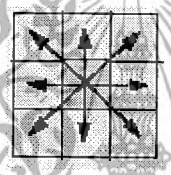
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การต่อเนื่องกันแบบ 4 จุด จะพิจารณาเฉพาะ 4 พิกเซลที่กำลังพิจารณา ทั้งทางแนวนอนและแนวตั้ง ดังแสดงในภาพที่ 2.9



ภาพที่ 2.9 การพิจารณาความต่อเนื่องกันแบบ 4 จุด

การต่อเนื่องกันแบบ 8 จุด จะพิจารณาทั้ง 8 พิกเซลที่อยู่ล้อมรอบพิกเซล จุดที่กำลังพิจารณา ดังแสดงในภาพที่ 2.10



ภาพที่ 2.10 การพิจารณาความต่อเนื่องกันแบบ 8 จุด

2.3 เทคนิค Template Matching

Template Matching หรือการเข้าคู่กับแม่แบบ เป็นเทคนิคการหาค่าความเหมือนระหว่างภาพ 2 ภาพ ว่ามีโอกาสที่จะเป็นภาพเดียวกันหรือไม่ ในการพิจารณาจะนำแม่แบบไปวางทาบกับภาพ ที่ต้องการพิจารณาและทำการประมวลผล โดยที่ตำแหน่งของพิกเซลของภาพ และของแม่แบบจะต้องอยู่ในตำแหน่งเดียวกัน ค่าความเหมือนระหว่างภาพ 2 ภาพ สามารถบอกได้โดยค่า Correlation Coefficient หรือ ค่าสัมประสิทธิ์ความเหมือน ซึ่งเป็นค่าแสดงความสัมพันธ์ระหว่างภาพ 2 ภาพ

เขียนเป็นสมการได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{(\sum_m \sum_n (A_{mn} - \bar{A})^2)}\sqrt{(\sum_m \sum_n (B_{mn} - \bar{B})^2)}} \qquad (2.2)$$

เมื่อ r คือ ค่าสัมประสิทธิ์ความเหมือน หรือค่า Correlation Coefficient

A และ B " ภาพ 2 ภาพที่มีขนาดเท่ากัน ที่ต้องการตรวจสอบว่าเป็นภาพเดียวกันหรือไม่

A_{mn} และ B_{mn} " พิกเซลที่กำลังพิจารณาของแต่ละภาพ

$\bar{A}$ และ $\bar{B}$ " ค่าความเข้มสีเฉลี่ยของทุกพิกเซลของภาพ A และ B ตามลำดับ

ค่าสัมประสิทธิ์ความเหมือน (r) มีค่าตั้งแต่ -1 ถึง 1 โดย 1 คือค่าที่แสดงการเท่ากันหรือมีความเหมือนกันที่สุด แต่ถ้าภาพ 2 ภาพมีความแตกต่างกัน ค่าสัมประสิทธิ์ความเหมือนก็จะลดลง กรณีที่ค่าสัมประสิทธิ์ความเหมือนมีค่าเป็น 0 ก็แสดงว่าภาพทั้ง 2 ภาพไม่มีความเหมือนกันเลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบโปรแกรมตรวจหาแผ่นป้ายทะเบียนรถยนต์

ในบทนี้จะอธิบายถึงการออกแบบโปรแกรมตรวจหาแผ่นป้ายทะเบียนรถยนต์ ซึ่งจะประกอบด้วยกระบวนการต่าง ๆ เริ่มจากการรับข้อมูลภาพถ่ายส่วนด้านหน้าของรถยนต์ ไปจนถึงการนำตัวเลขและพยัญชนะที่อยู่บนแผ่นป้ายทะเบียนรถยนต์ไปประมวลผล วิธีการตรวจหาแผ่นป้ายทะเบียนรถยนต์มีโครงสร้างการทำงาน 3 ขั้นตอนหลัก ได้แก่

1. การประมวลผลภาพดิจิตอล
2. การแยกส่วนที่เป็นตัวเลขและพยัญชนะออกมาจากแผ่นป้ายทะเบียน
3. การรู้จำตัวเลขและพยัญชนะ

3.1 การประมวลผลภาพดิจิตอล

1. รับภาพจากกล้อง Webcam เข้ามาโดยใช้คำสั่ง videoinput จากนั้นทำการกำหนดค่าการต่าง ๆ ของกล้องและเก็บข้อมูลภาพที่ได้ไว้ในไฟล์ข้อมูลเพื่อการเรียกใช้

- การกระตุ้น (triggerconfig) เป็นการจับภาพที่เราต้องการจากวีดีโอ มีการตั้งค่าสองแบบ คือ อัตโนมัติ (Auto) และกำหนดค่าเอง (Manual) เลือกการจับภาพแบบ Manual เพื่อรับเฉพาะภาพในเวลาที่ต้องการเท่านั้น ง่ายต่อการประมวลผล การกำหนดค่า การจับภาพสามารถทำการกำหนดได้ว่า ในหนึ่งรูปที่ได้นั้น จะมาจากการจับภาพกี่ครั้ง

- FramesPerTrigger เป็นการเลือกจำนวนการจับภาพต่อหนึ่งรูป เลือกการจับภาพหนึ่งครั้งต่อหนึ่งภาพรูป

- TriggerRepeat เป็นการกำหนดค่าการจับภาพ สามารถกำหนดได้หลายรูปแบบ เช่น 0 จะทำงานหนึ่งครั้งตามเงื่อนไขการจับภาพแรกที่เจอ หากเป็นตัวเลขค่าบวก ตัวเลขนั้นจะมาเป็นค่าเวลา ที่ใช้เริ่มต้นการทำงาน เมื่อเจอเงื่อนไขการจับภาพครั้งแรกที่เจอ และถ้าเป็น Inf การจับภาพจะทำงานตลอดเวลาตามเงื่อนไขการจับภาพที่กำหนด และหยุดเมื่อพบคำสั่งหยุด เลือกใช้ Inf

- การกำหนดรูปแบบสีของภาพ ReturnedColorSpace ซึ่งต้องการภาพเช่นเดียวกับภาพที่มนุษย์มองเห็น จึงเลือกเป็นภาพสี RGB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานเริ่มจากการกำหนดเงื่อนไขการจับภาพ นับ 1 ถึง 10 แล้วทำการจับภาพ เนื่องจากเมื่อเริ่มต้นนั้นการรับภาพเข้ามาจะใช้เวลาในการติดต่อ หากทำการจับภาพโดยทันที จะได้ภาพที่ไม่สมบูรณ์ คือมีความสว่างแต่ละช่วงไม่เท่ากัน จึงต้องกำหนดเงื่อนไขเว้นระยะเวลาก่อนการถ่ายภาพ จากนั้นแสดงผลรับข้อมูลภาพที่ได้ หยุดการรับภาพ และล้างภาพวีดีโอเดิมออก เพื่อเตรียมรับภาพใหม่ที่จะเข้ามา เก็บภาพที่ได้เป็นไฟล์ข้อมูลเพื่อการนำไปเรียกใช้ในขั้นตอนต่อไป คำสั่งทั้งหมดเขียนได้ดังนี้

```
vid=videoinput('winvideo',1);
% รับภาพวีดีโอจากกล้องเข้ามา
triggerconfig(vid,'manual');
%กำหนดการจับภาพวีดีโอเป็นแบบปรับด้วยการกำหนด
set(vid,'FramesPerTrigger',1);
%กำหนดค่ารูปจากการจับภาพจากวีดีโอเป็นรูปที่ได้มาจากการจับภาพหนึ่งครั้ง
set(vid,'TriggerRepeat',Inf);
 %กำหนดให้การจับภาพจะเกิดขึ้นตาม เงื่อนไขที่เรากำหนด
set(vid,'ReturnedColorSpace','rgb');
%กำหนดรูปแบบภาพที่ได้เป็นแบบ RGB
start(vid); % เริ่ม
for i=1:10
%เริ่มนับ i นับจาก 1 ไปถึง 10
trigger(vid);
% เมื่อค่า i =10 ทำการจับภาพ
im=getdata(vid,1);
%รับข้อมูลจากภาพเข้ามาเก็บในตัวแปร im
imshow(im);  % แสดงภาพ
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
stop(vid),clear vid;
%หยุดการรับภาพ ล้างข้อมูลเดิมออก
ime=imresize(im,[288 352]);
%ปรับขนาดภาพเพื่อสะดวกต่อการประมวลผล
Imwrite ( ime , 'D:\car\1.jpg');
%เก็บข้อมูลภาพไว้ในไฟล์ข้อมูลเพื่อการเรียกใช้
 imwrite(ime,'C:\'D:\car\1.jpg');
```

จากนั้นเรียกข้อมูลภาพส่วนด้านหน้าของรถยนต์เข้ามา ซึ่งเป็นภาพแบบ RGB โดยใช้คำสั่ง imread เขียนคำสั่งได้ดังนี้

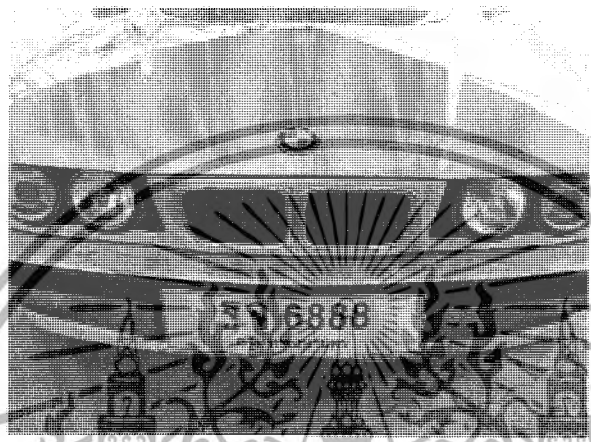
```
%----เรียกภาพชื่อ 1 นามสกุล JPEG อยู่ในแฟ้มที่ชื่อว่า car ในไดร์ฟ (Drive) D แล้วนำไปเก็บไว้ที่
%----ตัวแปร I
                I=imread('D:\car\1.jpg');
```

ภาพที่ 3.1 ภาพ RGB ส่วนด้านหน้ารถยนต์ที่รับเข้ามา

2. แปลงภาพจาก RGB เป็น Gray Scale โดยใช้คำสั่ง rgb2gray เขียนคำสั่ง ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----นำภาพ RGB ที่เก็บไว้ในตัวแปร I ไปแปลงเป็นภาพ Gray Scale แล้วเก็บไว้ในตัวแปร J
J=rgb2gray(I);
```



ภาพที่ 3.2 ภาพ Gray Scale ที่แปลงมาจากภาพ RGB

3. แปลงภาพ Gray Scale เป็นภาพ Binary โดยใช้วิธีการตัดระดับเทรชโฮลอัตโนมัติ ซึ่งมีอยู่ในโปรแกรม MATLAB โดยใช้คำสั่ง im2bw เขียนคำสั่งได้ดังนี้

```
%----หาค่าเทรชโฮลจากภาพ Gray Scale ที่เก็บไว้ในตัวแปร J แล้วเก็บไว้ที่ตัวแปร level
level = graythresh(J);
%----แปลงภาพ Gray Scale ที่เก็บไว้ในตัวแปร J ให้เป็นภาพ Binary โดยใช้ค่าเทรชโฮลที่ได้จาก
%----ตัวแปร level แล้วเก็บไว้ในตัวแปร bw
bw = im2bw(J,level);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ภาพที่ 3.3 ภาพ Binary ที่แปลงมาจากภาพ Gray Scale

4. ลบพื้นที่ภาพที่ติดกับส่วนขอบออกโดยใช้คำสั่ง imclearborder ซึ่งคำสั่งนี้จะลบกลุ่มพิกเซลสีขาวที่อยู่ติดกับขอบภาพโดยใช้จุดข้างเคียง 8 พิกเซลของหน้าต่างขนาด 3x3 จากภาพที่ 3.3 เมื่อเปรียบเทียบกับภาพที่ 3.4 จะเห็นได้ว่ากลุ่มพิกเซลสีขาวที่อยู่ติดกับขอบ จะถูกกำจัดออกไป ซึ่งสามารถเขียนคำสั่งได้ดังนี้

```
%----ลบกลุ่มพิกเซลสีขาวที่ติดกับขอบออกไปจากภาพ Binary ที่เก็บ ไว้ในตัวแปร bw โดยพิจารณา
%----จากจุดข้างเคียง 8 พิกเซล แล้วเก็บไว้ในตัวแปร BWnobord
                BWnobord = imclearborder(bw, 8);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
104124

ภาพที่ 3.4 ภาพ Binary ที่กำจัดส่วนที่ติดกับขอบออกไปแล้ว

5. กำจัดสิ่งรบกวนที่เป็นกลุ่มพิกเซลสีขาวจุดเล็ก ๆ ออกโดยใช้คำสั่ง imerode เพราะว่ากลุ่มพิกเซลสีขาวเหล่านี้ อาจจะทำให้โปรแกรมไม่สามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนได้ ซึ่งก่อนที่จะใช้คำสั่งนี้ จะต้องมีการเลือกใช้โครงสร้างของรูปที่จะนำไปทำการกัด (erode) ก่อน เช่น disk, diamond, ball, pair เป็นต้น โดยใช้คำสั่ง strel เป็นตัวกำหนดโครงสร้างของรูป หลังจากเลือกโครงสร้างที่จะนำไปทำการกัดแล้ว จะต้องทำการกำหนดค่ารัศมีหรือความกว้างของโครงสร้างซึ่งขึ้นอยู่กับชนิดที่เลือก อย่างไรก็ตามในงานวิจัยนี้ เราเลือกโครงสร้างชนิดสี่เหลี่ยมจัตุรัสด้วยความกว้าง 4 พิกเซล นั่นคือ se = strel('square',4) ซึ่งภาพผลลัพธ์ที่ได้หลังการกำจัดสิ่งรบกวนจะได้ดังภาพที่ 3.5 ซึ่งสามารถเขียนคำสั่งได้ดังนี้

```
%----กำหนดโครงสร้างที่จะนำไปกัดภาพเป็นโครงสร้างแบบ square ขนาด 4×4 พิกเซล แล้วเก็บไว้
%----ที่ตัวแปร se
se = strel('square',4);
%----ทำการกัดภาพ Binary ที่เก็บไว้ในตัวแปร BWnobord ด้วยโครงสร้างที่เก็บไว้ในตัวแปร se
%----แล้วนำภาพผลลัพทธ์ที่ได้จากการกัดไปเก็บไว้ในตัวแปร erodedBW
erodedBW = imerode(BWnobord,se);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 3.5 ภาพที่กำจัดสิ่งรบกวนออกไปแล้ว

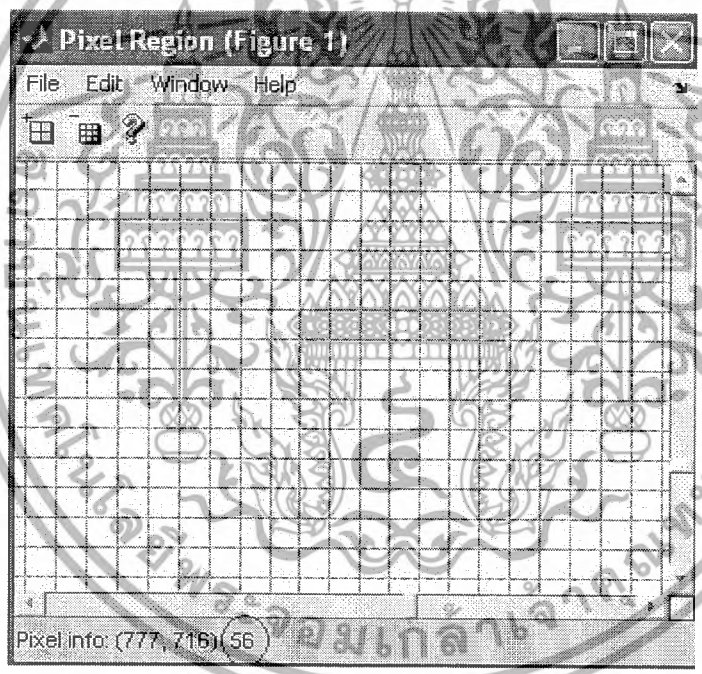
6. เมื่อกำจัดสิ่งรบกวนออกไปแล้ว ก็สามารถหาตำแหน่งของป้ายทะเบียนรถยนต์ได้ โดยทำการกำหนดหมายเลขให้กับกลุ่มพิกเซลสีขาว หรือเรียกว่าทำการ Label ให้กับกลุ่มพิกเซลสีขาว ซึ่งถ้ามีพิกเซลสีขาวอยู่ติดกันเป็นกลุ่มก้อนบริเวณเดียวกัน โดยที่ไม่ถูกแยกออกจากกันด้วยพิกเซลสีดำ กลุ่มพิกเซลสีขาวนั้นก็จะมีหมายเลข Label ที่เหมือนกัน จากภาพที่ 3.6 จะเป็นการ Label ให้กับภาพที่ 3.5 จากขั้นตอนที่ 5 ให้สังเกตที่ภาพ 3.6 (ก) และ (ค) ที่กลุ่มพิกเซลสีขาวที่มีสี่เหลี่ยมสีฟ้าวางอยู่ ซึ่งจะเห็นได้ว่ากลุ่มพิกเซลสีขาวของแต่ละภาพเป็นกลุ่มพิกเซลสีขาวบริเวณดียวกัน คือไม่ถูกแยกออกจากกันโดยพิกเซลสีดำ และภาพที่ 3.6 (ข) และ (ง) จะเป็นภาพขยายจากบริเวณที่มีสี่เหลี่ยมสีฟ้าวางอยู่ของภาพ (ก) และ (ค) ตามลำดับ ซึ่งหมายเลข 56 ที่ถูกวงกลมไว้ก็คือหมายเลขประจำกลุ่มพิกเซลสีขาวนั่นเอง ส่วนกลุ่มของพิกเซลสีขาวกลุ่มอื่น ๆ ก็จะกระทำในลักษณะเดียวกันโดยจะกำหนดหมายเลข Label ที่แตกต่างออกไป เมื่อทำการ Label ภาพ Binary แล้วให้ทำการสำรวจดูทุกกลุ่มพิกเซลสีขาวว่ากลุ่มไหนมีพื้นที่มากที่สุด ซึ่งพิกเซลสีขาวกลุ่มนี้ก็คือส่วนที่คาดว่าน่าจะเป็นส่วนของแผ่นป้ายทะเบียนรถยนต์ จากนั้นให้ใช้คู่พิกัด (Coordinate) ของกลุ่มพิกเซลสีขาวกลุ่มนี้เป็นจุดอ้างอิงในการคัดลอกภาพจากภาพ RGB ที่รับเข้ามาในขั้นตอนแรก จะได้ภาพผลลัพธ์ดังแสดงในภาพที่ 3.7 เพื่อที่ว่าจะแปลงภาพ RGB นี้ให้เป็นภาพ Binary โดยใช้ค่าเทรชโฮลอัตโนมัติที่พิจารณาเฉพาะส่วนที่เป็นแผ่นป้ายทะเบียนเท่านั้น ซึ่งจะทำให้ตัวเลขและพยัญชนะทั้งหมดที่อยู่บนแผ่นป้ายมีความชัดเจน เขียนเป็นคำสั่งได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----ทำการ Label ให้กับภาพที่เก็บไว้ในตัวแปร erodedBW ส่วนตัวแปร numObjects จะเป็นตัว
%----แปรที่เก็บจำนวนกลุ่มพิกเซลสีขาวที่มีอยู่ภาพ Binary นั้นไว้
[labeled,numObjects] = bwlabel(erodedBW,4);
squaredata11 = regionprops(labeled,'basic');
%----ตัวแปร allsquares11 จะเก็บค่าจำนวนพิกเซลของทุกกลุ่มพิกเซลสีขาวไว้
allsquares11=[squaredata11.Area];
%----ตัวแปร maxArea11 จะเก็บค่าจำนวนพิกเซลของกลุ่มพิกเซลสีขาวที่มีพื้นที่มากที่สุดจาก
%----ตัวแปร allsquares
maxArea11=max(allsquares11);
%----ตัวแปร biggestGrains11 จะหากลุ่มพิกเซลสีขาวในภาพ Binary ที่มีพื้นที่พื้นที่เท่ากับ
%----จำนวนพิกเซลที่เก็บไว้ในตัวแปร maxArea11
biggestGrain11 = find([squaredata11.Area]==maxArea11);
%----ตัวแปร r11 และ c11 จะเก็บคู่พิกัดเป็น rows และ column ที่มีกลุ่มพิกเซลสีขาวนี้อยู่
[r11,c11] = find(labeled==biggestGrain11);
i11max=max(r11); %----เก็บค่า rows ที่มากที่สุดไว้ที่ตัวแปร i11max
i11min=min(r11);  %----เก็บค่า rows ที่น้อยที่สุดไว้ที่ตัวแปร i11min
j11max=max(c11); %---- เก็บค่า column ที่มากที่สุดไว้ที่ตัวแปร i11max
j11min=min(c11); %---- เก็บค่า column ที่น้อยที่สุดไว้ที่ตัวแปร i11min
%----ใช้คู่พิกัดของกลุ่มพิกเซลสีขาวกลุ่มนี้เป็นแกนอ้างอิงในการคัดลอกภาพจากภาพ RGB ที่
%----รับเข้ามาในขั้นตอนแรก ซึ่งเก็บไว้ในตัวแปร I แล้วเก็บภาพที่ได้จากการคัดลอกไว้ในตัว
%----แปร A11
A11=I(i11min:i11max,j11min:j11max);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

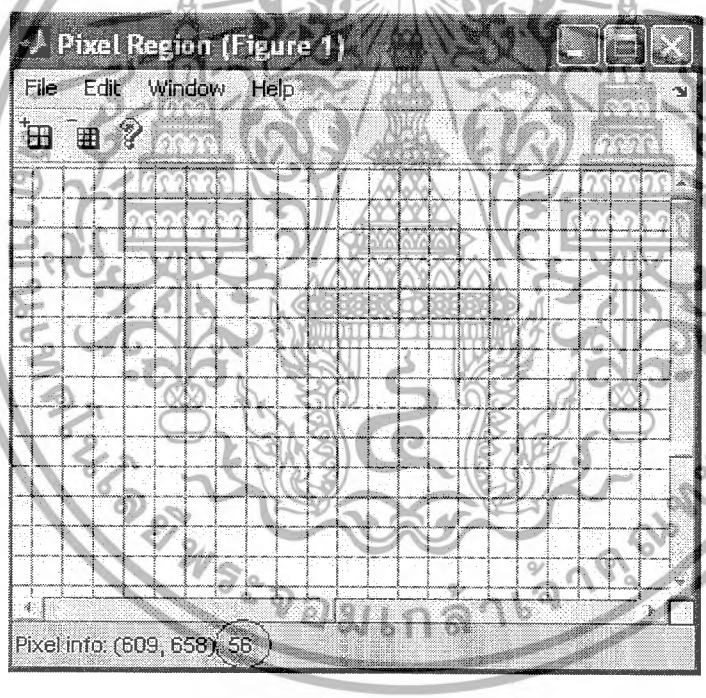
(ก)



(ข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ค)



(ง)

ภาพที่ 3.6 การกำหนดหมายเลขหรือทำการ Label ให้กับกลุ่มพิกเซลสีขาว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฐจ 6888
กรุงเทพมหานคร

ภาพที่ 3.7 ภาพที่ได้จากการนำคู่พิกัดของกลุ่มพิกเซลสีขาวที่มีพื้นที่มากที่สุดจากภาพที่ 3.6 ไปคัดลอกจากภาพ RGB ที่รับเข้ามาในขั้นตอนแรก

7. ตรวจสอบภาพที่ได้จากขั้นตอนที่ 6 ว่าเป็นส่วนของแผ่นป้ายทะเบียนรถยนต์หรือไม่ เพราะบางทีกลุ่มพิกเซลสีขาวที่มีพื้นที่มากสุด ซึ่งคาดว่าน่าจะเป็นแผ่นป้ายทะเบียนรถยนต์แต่จริง ๆ แล้วอาจจะไม่ใช่ ซึ่งอาจจะเป็นส่วนอื่น ๆ ของด้านหน้ารถยนต์ เช่น ไฟ กันชน ก็ได้ วิธีการตรวจสอบคือให้แปลภาพ RGB เป็นภาพ Inverse Binary หมายถึงถ้าพิกเซลไหนที่มีความเข้มสีสูงกว่าค่าเทรชโฮล ให้เปลี่ยนพิกเซลนั้น เป็นพิกเซล 0 หรือพิกเซลสีดำ ส่วนพิกเซลที่มีความเข้มสีต่ำกว่าค่าเทรชโฮล ให้เปลี่ยนพิกเซลนั้นเป็นพิกเซล 1 หรือสีขาว ซึ่งจะทำให้ได้ภาพ Binary ที่มีส่วนของตัวเลขและพยัญชนะเป็นพิกเซลสีขาว และส่วนพื้นของแผ่นป้ายทะเบียนรถยนต์จะเป็นพิกเซลสีดำ แล้วทำการ Label ลบกลุ่มพิกเซลสีขาวที่มีความสูงต่ำกว่าค่าที่กำหนดไว้ (ในโปรแกรมกำหนดให้เป็น 0.25 เท่าของความสูงภาพ Binary) เพื่อที่จะลบกลุ่มพิกเซลสีขาวที่มีความสูงที่คาดว่าไม่น่าจะเป็นตัวเลขและพยัญชนะที่อยู่บนแผ่นป้ายทะเบียนออกไป แล้วกำจัดกลุ่มพิกเซลสีขาวที่ติดกับขอบโดยคำสั่ง imclearborder ออกไป จากนั้นให้นับกลุ่มพิกเซลสีขาวที่มีเหลืออยู่ภายในข้อมูลภาพ ถ้ามีกลุ่มพิกเซลสีขาวเหลืออยู่มากกว่าค่าที่กำหนดไว้ แสดงว่าส่วนนี้คือส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ ให้ภาพนี้ไว้ใช้ในขั้นตอนต่อไป แต่ถ้ามีกลุ่มพิกเซลสีขาวเหลืออยู่ในภาพ Inverse Binary น้อยกว่าค่าที่กำหนดไว้ แสดงว่าส่วนที่คาดว่าเป็นส่วนของแผ่นป้ายทะเบียนรถยนต์นี้ อันที่จริงแล้วไม่ใช่ส่วนของแผ่นป้ายทะเบียนรถยนต์ ถ้าเข้าเงื่อนไขนี้ให้ทำการ Label ภาพ Binary จากขั้นตอนที่ 6 อีกครั้งเพื่อกำจัดกลุ่มพิกเซลสีขาวที่ใหญ่ที่สุด ซึ่งจากการตรวจสอบแล้วพบว่าไม่ใช่ส่วนที่เป็นแผ่นป้ายทะเบียน และกลุ่มพิกเซลสีขาวที่มีอัตราส่วนด้านกว้างต่อด้านยาว ความสูงและความกว้างที่คาดว่าไม่น่าจะใช่ส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ออกไป ซึ่งในโปรแกรมได้กำหนดอัตราส่วนด้านกว้างต่อด้านยาวที่นอกเหนือจาก $0.25 \leq$ อัตราส่วนด้านกว้างต่อด้านยาว ≤ 0.6 ถ้ากลุ่มพิกเซลสีขาวกลุ่มไหนเข้าเงื่อนไขตามที่กำหนดไว้ ให้กำจัดกลุ่มพิกเซลสีขาวนั้นออกไปหรือเปลี่ยนกลุ่มพิกเซลสีขาวนั้นให้เป็นพิกเซลสีดำทั้งหมดนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตัวอย่างภาพที่ 3.8 จะเห็นได้ว่ากลุ่มพิกเซลสีขาวที่ใหญ่ที่สุดใหญ่ที่สุดในภาพ Binary 3.8 (ก) ไม่ใช่ส่วนที่เป็นแผ่นป้ายทะเบียน แต่กลับเป็นส่วนของกระจังรถแทน เมื่อใช้คู่พิกัดของกลุ่มพิกเซลสีขาวนี้ ไปคัดลอกส่วนของภาพมาจากภาพ RGB ก็จะได้ภาพกระจังรถแบบ RGB ดังแสดงในภาพที่ 3.8 (ข) เมื่อนำภาพ RGB นี้ไปแปลงเป็นภาพ Inverse Binary ก็จะได้ภาพผลลัพธ์เป็นภาพ 3.8 (ค) เมื่อนำภาพ 3.8 (ค) ไปกำจัดกลุ่มพิกเซลสีขาวที่มีความสูงต่ำกว่า 0.25 เท่าของภาพออกและกำจัดกลุ่มพิกเซลสีขาวที่ติดกับขอบออกไป ภาพผลลัพธ์ที่ได้คือภาพ 3.8 (ง) ซึ่งแสดงให้เห็นว่าภาพ 3.8 (ข) ไม่ใช่ส่วนที่เป็นแผ่นป้ายทะเบียน เมื่อยังไม่ได้ส่วนที่เป็นแผ่นป้ายทะเบียน ให้นำภาพ 3.8 (ก) ไปกำจัดกลุ่มพิกเซลสีขาวที่ใหญ่ที่สุด ซึ่งผ่านการตรวจสอบแล้วว่าไม่ใช่ส่วนที่เป็นแผ่นป้ายทะเบียน กลุ่มพิกเซลสีขาวที่มีอัตราส่วนด้านกว้างต่อด้านยาวที่นอกเหนือจาก $0.25 \leq$ อัตราส่วนด้านกว้างต่อด้านยาว ≤ 0.6 จะทำให้ได้ภาพผลลัพธ์ดังภาพที่ 3.9 (ก) สุดท้ายจึงใช้คู่พิกัดกลุ่มพิกเซลสีขาวที่ใหญ่ที่สุดจากภาพนี้ไปคัดลอกส่วนของภาพมาจากภาพ RGB ก็จะทำให้ได้ส่วนที่เป็นแผ่นป้ายทะเบียนออกมา ดังแสดงในภาพที่ 3.9 (ข)

การทำงานทั้งหมดในขั้นตอนนี้เขียนเป็นคำสั่งได้ดังนี้

```
level11 = graythresh(A11); %----หาค่าเทรชโฮลของภาพ RGB ที่เก็บไว้ในตัวแปร A11
bw11 = im2bw(A11,level11); %----แปลงภาพ RGB ที่เก็บไว้ในตัวแปร A11 ให้เป็นภาพ Binary
%----หาขนาดของภาพที่เก็บ ไว้ในตัวแปร bw11 โดยตัวแปร m11 จะเก็บค่าความสูงของภาพ bw11
%----ในหน่วยพิกเซล ส่วนตัวแปร n11 จะเก็บค่าความกว้างในหน่วยพิกเซลเช่นกัน
[m11,n11]=size(bw11);
for i11=1:m11
   for j11=1:n11
%---- ถ้าพิกเซลที่กำลังพิจารณามีค่าความสว่างมากกว่าค่าเทรชโฮล ให้เปลี่ยนพิกเซลนั้นมีค่าความ
%----สว่างเป็น 0 หรือเป็นพิกเซลสีดำ
      if bw11(i11,j11) > level11;
         bw11(i11,j11)=0;
%---- ถ้าพิกเซลที่กำลังพิจารณามีค่าความสว่างน้อยกว่าค่าเทรชโฮล ให้เปลี่ยนพิกเซลนั้นมีค่าความ
%----สว่างเป็น 1 หรือเป็นพิกเซลสีขาว
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else bw11(i11,j11)<= level11;
bw11(i11,j11)=255;
end
end
end
[x113,y113]=size(bw11); %----หาขนาดของภาพ Binary ที่เก็บไว้ในตัวแปร bw11 แล้วทำการ
%----Label ให้กับภาพ Binary
[labeled113,numObjects113] = bwlabel(bw11,4);
for u113=1:numObjects113 %----สำรวจทุกกลุ่มพิกเซลสีขาวที่มีอยู่ในภาพ Binary
[rr113,cc113] = find(labeled113==u113);
ii113max=max(rr113); %----เก็บขอบล่างของแต่ละกลุ่มพิกเซลสีขาวไว้ในตัวแปร ii113max
ii113min=min(rr113); %----เก็บขอบบนของแต่ละกลุ่มพิกเซลสีขาวไว้ในตัวแปร ii113min
jj113max=max(cc113); %----เก็บขอบขวาของแต่ละกลุ่มพิกเซลสีขาวไว้ในตัวแปร jj113max
jj113min=min(cc113); %----เก็บขอบซ้ายของแต่ละกลุ่มพิกเซลสีขาวไว้ในตัวแปร jj113max
%----นำค่าขอบล่างลบออกด้วยค่าขอบบนจะทำให้ได้ความสูงของกลุ่มพิกเซลสีขาวนั้น แล้วเก็บ
%----ค่าความสูงนี้ไว้ในตัวแปร high113
high113=ii113max-ii113min;
%----ถ้าค่าความสูงของกลุ่มพิกเซลสีขาวในตัวแปร high113 มีค่าน้อยว่า 0.25 เท่าของภาพ Binary
%----ที่เก็บไว้ในตัวแปร bw11 ให้เปลี่ยนกลุ่มพิกเซลสีขาวนั้นเป็นพิกเซลสีดำทั้งหมด
if high113 < (0.25*x113)
bw11(labeled113==u113)=0;
end
end %----ทำการสำรวจทุกกลุ่มพิกเซลสีขาวครบทุกกลุ่มแล้ว
BWnobord11 = imclearborder(bw11, 4); %----กำจัดกลุ่มพิกเซลสีขาวออกไปจากภาพ Binary ที่
%----เก็บไว้ในตัวแปร bw11 แล้วเก็บภาพผลลัพธ์ไว้ในตัวแปร BWnobord11
[x112,y112]=size(BWnobord11);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

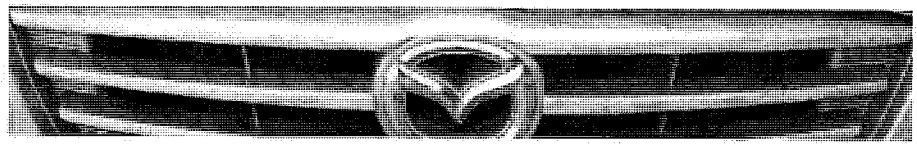
```
%----ทำการ Label เพื่อนับจำนวนกลุ่มพิกเซลสีขาวที่มีเหลืออยู่ในภาพ Binary ที่เก็บไว้ในตัวแปร
%----BWnobord11 แล้วเก็บภาพผลลัพธ์จากการ Label ไว้ในตัวแปร labeled112 ซึ่งจะเป็นการ
%----ตรวจสอบว่าถ้าภาพนี้คือส่วนที่เป็นแผ่นป้ายทะเบียน ก็จะต้องมีจำนวนกลุ่มพิกเซลสีขาว ซึ่ง
%----หมายถึงส่วนที่เป็นตัวเลขและพยัญชนะเหลืออยู่ในภาพนี้
[labeled112,numObjects112] = bwlabel(BWnobord11,4);
for u112=1:numObjects112
[rr112,cc112] = find(labeled112==u112);
ii112max=max(rr112);
ii112min=min(rr112);
jj112max=max(cc112);
jj112min=min(cc112);
high112=ii112max-ii112min;
%----ถ้าความสูงของกลุ่มพิกเซลสีขาวนั้นมากกว่า 0.4 เท่าของความสูงของภาพ Binary ที่เก็บไว้
ใน
%----ตัวแปร BWnobord11 ให้ตัวแปร count1 นับ 1
if high112 > (0.4*x112)
   count1=count1+1;
end
end
%----ถ้าตัวแปร count1 มีค่าน้อยกว่าหรือเท่ากับ 3 หรือมีจำนวนกลุ่มพิกเซลสีขาวเหลืออยู่น้อย
%----กว่า 3 กลุ่ม ก็แสดงว่าภาพ Binary ที่เก็บไว้ในตัวแปร BWnobord11 ไม่ใช่ส่วนที่เป็นแผ่น
%----ป้ายทะเบียนรถยนต์ ดังนั้นให้ทำการกำจัดกลุ่มพิกเซลสีขาวที่คาดว่าไม่ใช่ส่วนที่เป็นแผ่น
%----ป้ายทะเบียนออกไป โดยพิจารณาจากอัตราส่วนด้านกว้างต่อด้านยาว ความสูง
if numObjects112<3 ||count1<=3
%---กำจัดกลุ่มพิกเซลสีขาวที่ผ่านการตรวจสอบแล้วพบว่าไม่ใช่ส่วนที่เป็นแผ่นป้ายทะเบียน
%----ออกไป
biggestGrain11 = find([squaredata11.Area]==maxArea11);
[r11,c11] = find(labeled==biggestGrain11);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
labeled(r11,c11)=0;
[JJ,KK]=size(labeled);
for u=1:numObjects
   [r,c] = find(labeled==u);
imax=max(r);
imin=min(r);
jmax=max(c);
jmin=min(c);
%----ถ้ากลุ่มพิกเซลสีขาวกลุ่มนั้นมีอัตราส่วนความกว้างต่อความยาวน้อยกว่า 0.25 และมากกว่า
%----หรือเท่ากับ 0.6 ให้เปลี่ยนกลุ่มพิกเซลสีขาวนั้นเป็นพิกเซลสีดำทั้งหมด
 if  (imax-imin)/(jmax-jmin)<0.25 |(imax-imin)/(jmax-jmin)>=0.6
    labeled(r,c)=0;
end
end
end
```

(ก)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ข)

(ค)

(ง)

ภาพที่ 3.8 ตัวอย่างภาพที่เกิดจากการนำคู่พิกัดของกลุ่มพิกเซลสีขาวที่มีพื้นที่มากสุด ไปคัดลอกมา จาก ภาพ RGB แล้วได้ส่วนที่ไม่ใช่แผ่นป้ายทะเบียนออกมา

(ก) ภาพ Binary

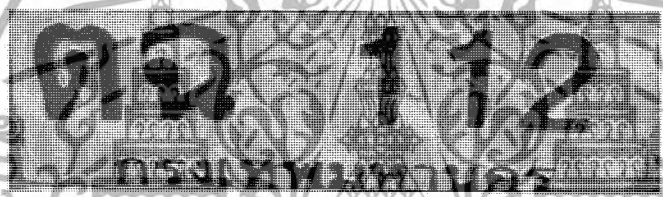
(ข) ภาพที่ได้จากการนำ Coordinate ของกลุ่มพิกเซลสีขาวที่มีพื้นที่มากที่สุด จากภาพ (ก) ไปคัดลอกจากภาพ RGB

(ค) ภาพที่ได้จากการนำภาพ (ข) ไปทำ Inverse Binary Image

(ง) ภาพที่ได้จากการนำภาพ (ค) ไปกำจัดกลุ่มพิกเซลสีขาวที่มีความสูงน้อยกว่าค่าที่กำหนดไว้และมีพื้นที่ที่ติดกับขอบภาพออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ก)



(ข)

ภาพที่ 3.9 (ก) ภาพที่ได้จากการนำภาพ Binary ไปกำจัดกลุ่มพิกเซลสีขาวที่มีอัตราส่วนความกว้างต่อความยาวและความสูงที่คาดว่าไม่น่าจะเป็นส่วนของแผ่นป้ายทะเบียนรถยนต์ออกไป
(ข) ภาพที่ได้จากการนำคู่พิกัดของกลุ่มพิกเซลสีขาวที่มีพื้นที่มากสุดจากภาพ (ก) ไปคัดลอกมาจากภาพ RGB

8. ให้ตรวจสอบดูว่ากลุ่มพิกเซลสีขาวที่ใหญ่ที่สุด หรือมีพื้นที่มากสุดที่เหลืออยู่ มีพื้นที่มากกว่าค่าที่กำหนดไว้หรือไม่ (ในโปรแกรมกำหนดไว้ที่ 50 พิกเซล) ซึ่งถ้ากลุ่มพิกเซลสีขาวที่มีพื้นที่มากสุดที่มีเหลืออยู่ในภาพ Binary มีพื้นที่น้อยกว่าค่าที่กำหนดไว้ แสดงว่าส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ถูกลบออกไป ทั้งนี้เป็นเพราะว่าภาพ RGB ที่รับเข้ามาในขั้นตอนแรกไม่เหมาะสม ที่จะใช้ค่าเทรชโฮลอัตโนมัติในการแปลงให้เป็น Binary Image ทำให้ภาพ Binary ที่ได้อาจจะมีความสว่าง หรือมีพิกเซลสีขาวมากเกินไป เมื่อเป็นเช่นนี้ขอบเขตของส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ก็จะมีความไม่ชัดเจนพอ จึงทำให้ค้นหาส่วนที่เป็นแผ่นป้ายทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถยนต์ไม่ได้ เนื่องจากคำสั่ง imclearborder ในขั้นตอนที่ 4 จะลบส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์นี้ออกไป เพราะว่าส่วนของแผ่นป้ายทะเบียนรถยนต์มีพิกเซลสีขาวบางส่วนติดกับขอบภาพ เมื่อนำข้อมูลภาพนี้ไปประมวลผลต่อไปขั้นตอนที่ 5 และ 6 จะทำให้ได้ภาพ Binary ที่เป็นพิกเซลสีดำเกือบทั้งหมด ซึ่งทำให้ไม่สามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ได้ (ดูภาพที่ 3.10) ดังนั้นจึงแก้ปัญหาโดยการนำภาพ RGB ที่รับเข้ามาในขั้นตอนแรกแปลงให้เป็นภาพ Binary โดยใช้ค่าเทรชโฮลที่เกิดจากการนำค่าความเข้มสีของทุกพิกเซลใน RGB มาเฉลี่ยกันแล้วบวกด้วยค่าคงที่ค่าหนึ่ง (ในโปรแกรมกำหนดไว้ที่ 50) จากนั้นให้นำภาพ Binary ที่ได้จากขั้นตอนนี้ไปประมวลผลในขั้นตอนต่อไป เหมือนข้อ 5 ไปจนถึงข้อ 11 ต่างกันที่ข้อขั้นตอนที่ 6 คือถ้ากลุ่มพิกเซลสีขาวที่มีพื้นที่มากที่สุดที่มีเหลืออยู่ในภาพ Binary หลังจากที่ผ่านการกำจัดสิ่งรบกวนต่าง ๆเรียบร้อยแล้วมีพื้นที่น้อยกว่า 50 พิกเซล ให้หยุดทำการประมวลผลในขั้นตอนต่อไปทันที ข้อมูลภาพที่เป็นผลลัพธ์ที่ได้จากการนำภาพ Binary ในส่วนนี้ไปประมวลผลจะถูกเรียกใช้ถ้าหากภาพ Binary ที่เกิดจากการแปลงโดยใช้ค่าเทรชโฮลอัตโนมัติที่มีอยู่ในโปรแกรม MATLAB ไม่สามารถนำไปประมวลผลได้ในบางขั้นตอน ตัวอย่างเช่น การค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์มีความผิดพลาด หรือว่าไม่สามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ได้ เพราะว่าค่าเทรชโฮลที่เกิดจากการนำค่าความเข้มสีทุกพิกเซลของภาพ RGB มาเฉลี่ยกันแล้วบวกเพิ่มด้วยค่าคงที่จะทำให้ส่วนของแผ่นป้ายทะเบียนรถยนต์ในภาพ Binary ที่ได้มีขอบเขตเป็นของตัวเองที่ชัดเจนมากขึ้น เมื่อนำไปประมวลผลในขั้นตอนต่อไป ส่วนนี้จะไม่ถูกลบออกไป ดังแสดงในรูปที่ 3.11 (ข) และเมื่อใช้คู่พิกัดของกลุ่มพิกเซลสีขาวที่ใหญ่ที่สุดจากภาพนี้ ไปคัดลอกมาจากภาพ RGB ทำให้สามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนได้ ดังแสดงในภาพที่ 3.11 (ข) ซึ่งเขียนเป็นคำสั่งได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

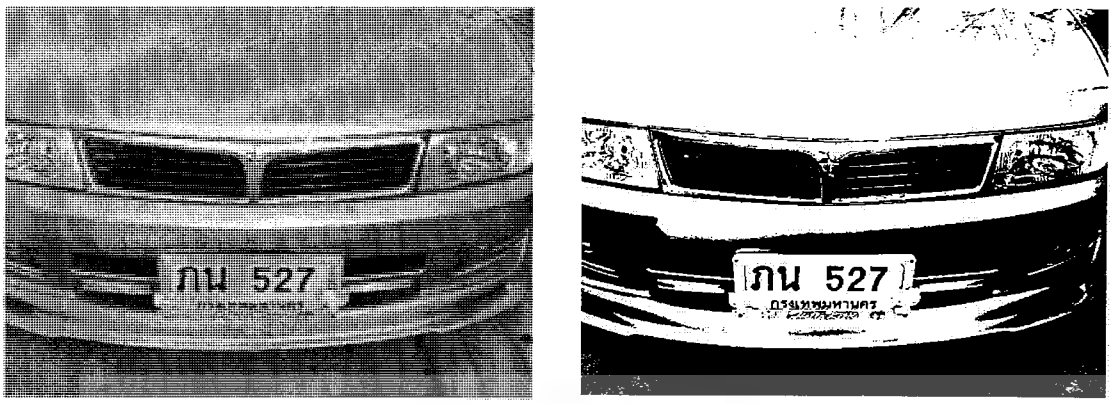
```
[labeled_1,numObjects_1] = bwlabel(labeled,4);
[row,columm]=size(labeled_1);
squaredata = regionprops(labeled_1,'basic');
allsquares=[squaredata.Area];
maxArea=max(allsquares);
[m,n]=size(labeled_1);
squaredata = regionprops(labeled_1,'basic');
%----ถ้ากลุ่มพิกเซลสีขาวที่มีพื้นที่ใหญ่ที่สุดมีพื้นที่มากกว่า 50 พิกเซล ให้หาคู่พิกัดของกลุ่ม
%----พิกเซลสีขาวนี้
if maxArea>50
biggestGrain = find([squaredata.Area]==maxArea);
 [r_1,c_1] = find(labeled_1==biggestGrain);
 i_1max=max(r_1);
 i_1min=min(r_1);
 j_1max=max(c_1);
 j_1min=min(c_1);
%----ใช้คู่พิกัดของกลุ่มพิกเซลสีขาวนี้เป็นจุดอ้างอิงในการคัดลอกส่วนนี้มาจากภาพ RGB
%----ที่เก็บไว้ในตัวแปร I แล้วเก็บภาพผลลัพธ์ที่ได้ไว้ในตัวแปร A
A=I(i_1min:i_1max,j_1min:j_1max);
 End
%----หาค่าเทรชโฮลอัตโนมัติจากภาพ RGB ที่เก็บไว้ในตัวแปร A แล้วเก็บค่าเทรชโฮลที่ได้
%----ไว้ในตัวแปร level2
level2 = graythresh(A);
%----แปลงภาพ RGB ที่เก็บไว้ในตัวแปร A ให้เป็นภาพ Binary โดยใช้ค่าเทรชโฮลที่เก็บไว้ใน
%----ตัวแปร level2 แล้วเก็บภาพ Binary ที่ได้ไว้ในตัวแปร bw2
bw2 = im2bw(A,level2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

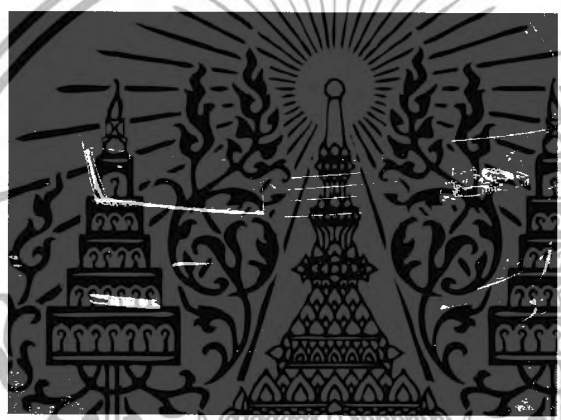
```
[m,n]=size(bw2);
for i=1:m
   for j=1:n
%----ถ้าพิกเซลที่กำลังพิจารณาเป็นสีขาว ให้เปลี่ยนพิกเซลนั้นเป็นสีดำ
      if bw2(i,j) ==1;
          bw2(I,j)=0;
%----ถ้าพิกเซลที่กำลังพิจารณาเป็นสีดำ ให้เปลี่ยนพิกเซลนั้นเป็นสีขาว
     else bw2(i,j)=0;
         bw2(i,j)=1;
      end
   end
end
PICTURE=bw2;
%---แต่ถ้ากลุ่มพิกเซลสีขาวที่มีขนาดใหญ่ที่สุดมีพื้นที่น้อยกว่า 50 พิกเซล ให้ดึงส่วนนี้มา
%---จากการภาพ Binary Image ที่เกิดจากค่าเทรชโฮลที่คำนวณได้ไปประมวลผล
else PICTURE=bw2_2nd;
```

จากภาพที่ 3.10 (ก) เป็นตัวอย่างภาพ RGB ที่ไม่เหมาะสมในการแปลงให้เป็นภาพ Binary โดยใช้ค่าเทรชโฮลอัตโนมัติ เพราะภาพ Binary ที่ได้ ดังแสดงในภาพที่ 3.10 (ข) จะเห็นได้ว่าส่วนที่เป็นแผ่นป้ายทะเบียนมีกลุ่มพิกเซลสีขาวที่ติดกับขอบภาพ เมื่อภาพนี้ถูกประมวลผลผ่านขั้นตอนที่ 4 จะทำให้กลุ่มพิกเซลสีขาวนี้ถูกกำจัดออกไปด้วยคำสั่ง imclearborder ดังแสดงในภาพที่ 3.10 (ค) ซึ่งถ้านำภาพนี้ไปประมวลผลในขั้นตอนต่อไป โปรแกรมก็จะไม่สามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนได้ แต่ถ้านำภาพตัวอย่าง RGB นี้ไปแปลงเป็นภาพ Binary โดยใช้ค่าเทรชโฮลที่เกิดจากการนำค่าความเข้มสีทุกพิกเซลของภาพ RGB มาเฉลี่ยกันแล้วบวกเพิ่มด้วย 50 จะทำให้ได้ภาพผลลัพธ์ดังแสดงในภาพที่ 3.11 (ก) ซึ่งจะเห็นได้ว่าส่วนที่เป็นแผ่นป้ายทะเบียนมีขอบเขตที่ชัดเจนหรือไม่มีกลุ่มพิกเซลสีขาวที่ติดกับขอบภาพ โปรแกรมก็จะสามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนจากภาพ Binary นี้ได้ ดังแสดง ในภาพที่ 3.11 (ข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) (ข)

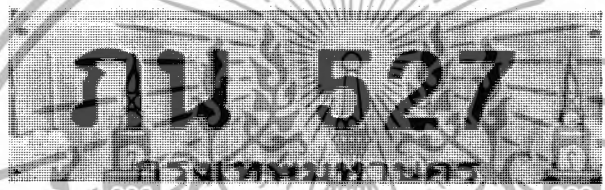


(ค)

ภาพที่ 3.10 ตัวอย่างการแปลงภาพ RGB เป็นภาพ Binary โดยใช้ค่าเทรชโฮลอัตโนมัติแล้ว ทำให้ส่วนที่เป็นแผ่นป้ายทะเบียนถูกลบออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ก)



(ข)

ภาพที่ 3.11 (ก) ภาพ จากภาพที่ 3.10

9. เมื่อค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ได้แล้ว ให้นำภาพแผ่นป้ายทะเบียนรถยนต์ที่ได้มาจากขั้นตอนที่ 8 มาตัดระดับเทรชโฮลอัตโนมัติจากโปรแกรม MATLAB อีกครั้ง ที่ทำเช่นนี้เพราะว่าส่วนที่เป็นตัวเลขและตัวพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ จะได้มีความคมชัดมากขึ้น และจากนั้นให้ทำการ Inverse ข้อมูลภาพ Binary Image ผลที่ได้คือส่วนที่เป็นตัวเลขและพยัญชนะจะเป็นพิกเซลสีขาว และส่วนพื้นของแผ่นป้ายทะเบียนรถยนต์จะเป็นสีดำ ดังภาพที่ 3.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฐจ 6888
กรุงเทพมหานคร

ภาพที่ 3.12 ภาพ Inverse Binary ของภาพที่ 3.7

10. ทำการ Label ข้อมูลภาพที่ได้จากขั้นตอนที่ 9 (ภาพที่ 3.12) เพื่อกำจัดกลุ่มพิกเซลสีขาวที่มีความสูงต่ำกว่า 0.25 เท่าของความสูงของภาพ Binary เพื่อที่จะกำจัดส่วนที่เป็นชื่อจังหวัดออกไปจากภาพป้ายทะเบียนนี้ เหตุผลเนื่องจากว่าในงานวิจัยนี้ต้องการตรวจหาและรู้จำเฉพาะตัวเลขและพยัญชนะที่อยู่แถวบนของแผ่นป้ายทะเบียนเท่านั้น เขียนคำสั่งได้ดังนี้

```
%----ทำการ Label ให้กับภาพ Binary ของส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ที่โปรแกรมค้นหาได้
%----ซึ่งเก็บไว้ในตัวแปร PICTURE แล้วเก็บภาพผลลัพธ์ที่ได้จากการ Label ไว้ที่ตัวแปร
%----labeled16
[labeled16,numObjects16] = bwlabel(PICTURE,4);
squaredata16 = regionprops(labeled16,'basic');
%----ทำการสำรวจทุกกลุ่มพิกเซลสีขาวที่มีอยู่ภาพ Binary ที่เก็บไว้ในตัวแปร labeled16
for u16=1:numObjects16
   [rr16,cc16] = find(labeled16==u16);
%----หาขอบเขตของแต่ละกลุ่มพิกเซลสีขาว
ii16max=max(rr16); %----เก็บค่าขอบล่างไว้ในตัวแปร ii16max
ii16min=min(rr16); %----เก็บค่าขอบบนไว้ในตัวแปร ii16min
jj16max=max(cc16); %----เก็บค่าขอบขวาไว้ในตัวแปร jj16max
jj16min=min(cc16); %----เก็บค่าขอบซ้ายไว้ในตัวแปร jj16min
%----นำค่าขอบล่างลบด้วยขอบบน จะทำให้ได้ความสูงของกลุ่มพิกเซลสีขาว แล้วเก็บไว้ในตัวแปร
%----high
high=ii16max-ii16min;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----นำค่าขอบขวาลบด้วยค่าขอบซ้าย จะทำให้ได้ความกว้างของกลุ่มพิกเซลสีขาว แล้วเก็บไว้
%----ในตัวแปร width
width=jj16max-jj16min;
%----ถ้ากลุ่มพิกเซลสีขาวนั้นมีความสูงน้อยกว่า 0.25 เท่าของภาพ Binary ที่เก็บไว้ในตัวแปร
%----PICTURE ให้เปลี่ยนกลุ่มพิกเซลสีขาวนั้นให้เป็นพิกเซลสีดำทั้งหมด
if high<(0.25*m)
   PICTURE(labeled16==u16)=0;
end
end
```

ภาพที่ 3.13 ภาพที่ได้จากการกำจัดส่วนที่เป็นชื่อจังหวัดออกไปจากภาพที่ 3.12

11. กำจัดกลุ่มพิกเซลสีขาวที่คาดว่าเป็นสิ่งรบกวน เช่นลวดลายของแผ่นป้ายทะเบียนและกรอบ และกำจัดกลุ่มพิกเซลสีขาวที่ติดกับขอบของภาพที่ 3.13 จากขั้นตอนที่ 10 ออกไปโดยใช้คำสั่ง imclearborder เพราะว่าในการเก็บภาพ RGB ส่วนของด้านหน้ารถยนต์นั้น ถ้าตั้งกล้องเอียง หรือเนื่องมาจากลักษณะของกรอบป้ายทะเบียนรถยนต์ จะทำให้รูปส่วนของแผ่นป้ายทะเบียนรถยนต์ที่เป็นภาพ Binary มีขอบของภาพเป็นพิกเซลสีขาว ดังภาพที่ 3.13 ซึ่งถ้ากำจัดส่วนนี้ออกไปจะทำให้ตัวเลขและตัวอักษร แตกต่างจากพื้นหลังของแผ่นป้ายทะเบียนรถยนต์มีความชัดเจนมากยิ่งขึ้น เขียนคำสั่งได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
[J,K]=size(PICTURE);
%----ทำให้แถวที่ 1 ถึง 3 ของภาพที่เก็บไว้ในตัวแปร PICTURE เป็นสีดำทั้งหมด เพื่อป้องกันไม่ให้
%----ส่วนที่เป็นตัวเลขและพยัญชนะที่อยู่บนแผ่นป้ายถูกลบออกไปด้วยคำสั่ง imclearborder
PICTURE(1:3,1:K)= 0;
[j,k]=size(PICTURE);
%----ถ้าแถวไหนมีพิกเซลสีขาวติดกันต่อเนื่องกันเป็นจำนวน  (0.3 *ของความกว้างภาพ)พิกเซล
%----ให้เปลี่ยนพิกเซลในแถวนั้นเป็นพิกเซลสีดำทั้งแถว
K=int32(0.3*k);
for jj=1:j
   for kk=1:k-K
      if PICTURE(jj,kk)==1 & PICTURE(jj,(kk:(kk+K)))==1
         PICTURE(jj,1:k)=0;
      end
   end
end
%----กำจัดกลุ่มพิกเซลสีขาวที่ติดกับขอบออกไปจากภาพ Binary ที่เก็บไว้ในตัวแปร PICTURE
%----แล้วเก็บภาพผลลัพธ์ที่ได้ไว้ในตัวแปร BW2
BW2 = imclearborder(PICTURE, 4);
%----ทำการ Label ให้กับภาพที่เก็บไว้ในตัวแปร BW2 แล้วเก็บภาพผลลัพธ์ที่ได้จากการทำ Label
%----ไว้ในตัวแปร labeled17
[labeled17,numObjects17] = bwlabel(BW2,4);
[j1,k1]=size(labeled17);
%----ทำการสำรวจทุกกลุ่มพิกเซลสีขาว
for u17=1:numObjects17
   [rr17,cc17] = find(labeled17==u17);
%----หาขอบเขตของแต่ละกลุ่มพิกเซลสีขาว
ii17max=max(rr17); %----เก็บค่าขอบล่างของกลุ่มพิกเซลสีขาวไว้ในตัวแปร ii17max
ii17min=min(rr17); %----เก็บค่าขอบบนของกลุ่มพิกเซลสีขาวไว้ในตัวแปร ii17min
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
jj17max=max(cc17); %---- เก็บค่าขอบขวาของกลุ่มพิกเซลสีขาวไว้ในตัวแปร jj17max
jj17min=min(cc17); %----เก็บค่าขอบซ้ายของกลุ่มพิกเซลสีขาวไว้ในตัวแปร jj17min
%----หาความสูงของกลุ่มพิกเซลสีขาว ซึ่งหาได้จากนำค่าขอบล่างลบด้วยค่าขอบบน แล้วเก็บไว้
%----ในตัวแปร high
high=ii17max-ii17min;
 %----หาความกว้างของกลุ่มพิกเซลสีขาว ซึ่งหาได้จากนำค่าขอบขวาลบด้วยค่าขอบซ้าย แล้วเก็บ
%----ไว้ในตัวแปร width
width=jj17max-jj17min;
%----ถ้ากลุ่มพิกเซลสีขาวนั้นมีความสูงน้อยกว่า 0.3 หรือมากกว่า 0.75 เท่าของความสูงของภาพ
%----Binary หรือ มีความกว้างน้อยกว่า 0.025 เท่าของความกว้างของภาพ Binary หรือมีอัตราส่วน
%----ความสูงต่อความกว้างน้อยกว่าหรือเท่ากับ 0.15 ให้เปลี่ยนกลุ่มพิกเซลสีขาวนั้นเป็นพิกเซล
%----สีดำทั้งหมด
if high<(0.3*j1)||high>(0.75*j1)||width<(0.025*k1)|| width/high <=0.15
   BW2(labeled17==u17)=0;
end
end
```

ฉจ 6888

ภาพที่ 3.14 ภาพที่ได้จากการกำจัดกลุ่มพิกเซลสีขาวที่ติดกับขอบภาพและสิ่งรบกวนต่าง ๆ ที่อยู่บนแผ่นป้ายออกไปจากภาพที่ 3.13

12. เมื่อได้ภาพ Binary ที่เหลือเฉพาะส่วนที่เป็นตัวเลขและพยัญชนะแถวบนที่อยู่บนแผ่นป้ายทะเบียนแล้ว บางทีส่วนนี้อาจจะมีความเอียง ซึ่งเป็นผลเนื่องมาจากการตั้งกล้องถ่ายภาพ ให้ทำการหมุนส่วนนี้ให้กลับมาอยู่ในแนวนอนมากขึ้น โดยใช้คำสั่ง imrotate ซึ่งคำสั่งนี้ ถ้ากำหนดค่ามุมเป็นบวก ภาพจะหมุนไปในทิศทางทวนเข็มนาฬิกา แต่ถ้ากำหนดค่ามุมเป็นลบ ภาพจะหมุนไปใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทิศทางตามเข็มนาฬิกา ดังนั้นก่อนที่จะหมุนภาพ Binary Image ที่ได้ในขั้นตอนนี้ ก็จะต้องหาค่ามุมที่กลุ่มตัวเลขและพยัญชนะทำกับแนวนอนก่อนซึ่งมีขั้นตอนดังต่อไปนี้

1. เลือก Column ที่เป็นขอบขวามือสุดของตัวเลขหรือพยัญชนะตัวใดตัวหนึ่งออกมาหนึ่งค่า

2. หา Column ที่เป็นขอบขวามือสุดของตัวเลขหรือพยัญชนะตัวที่อยู่ก่อนหน้าตัวเลขหรือพยัญชนะ ที่ได้ทำการเลือกไว้จากขั้นตอนที่ 1

3. นำค่า Column ที่เป็นขอบขวาที่ได้จากขั้นตอนที่ 1 ลบด้วยค่าที่ได้จากขั้นตอนที่ 2 จะทำให้ได้ความกว้างฐานของรูปสามเหลี่ยม (ดูภาพที่ 3.15 (ค) และ (ข))

4. หาขอบล่างของตัวเลขหรือพยัญชนะที่ได้ทำการเลือกไว้จากขั้นตอนที่ 1

5. หาขอบล่างของตัวเลขหรือพยัญชนะที่ได้ทำการเลือกไว้จากขั้นตอนที่ 2

6. นำค่าที่ได้จากขั้นตอนที่ 4 ลบค่าที่ได้จากขั้นตอนที่ 5 ก็จะทำให้ได้ความสูงของรูปสามเหลี่ยม (ดูภาพที่ 3.15 (ค) และ (ข))

7. หาด้านตรงข้ามมุมฉาก ของรูปสามเหลี่ยม

8. ในการกำหนดทิศทางการหมุน ให้ตรวจสอบทิศทางการเอียงของกลุ่มตัวเลขและพยัญชนะในภาพแผ่นป้ายทะเบียน เพื่อที่จะกำหนดทิศทางการหมุนให้กลุ่มตัวเลขและพยัญชนะกลับมาอยู่ในแนวราบมากขึ้น โดยทำการ Label เพื่อหากลุ่มพิกเซลสีขาวที่อยู่ติดกับขอบล่างของภาพ Binary มากที่สุด เมื่อทราบว่าเป็นกลุ่มพิกเซลสีขาวไหนแล้ว ให้หาขอบด้านซ้ายของกลุ่มพิกเซลสีขาวนั้น ซึ่งเป็น Column ของเมตริกซ์ เมื่อได้ค่า Column นั้นแล้ว ถ้าค่า Column มีค่ามากกว่า 0.5 ของความกว้างของภาพ แสดงว่าจุดต่ำสุดของกลุ่มพยัญชนะและตัวเลขอยู่ทางด้านขวามือ เมื่อไล่ไปทางด้านซ้ายมือจะพบว่าพยัญชนะและตัวเลขตัวถัดอยู่ห่างจากขอบภาพของภาพ Binary มากกว่าตัวเลขตัวนี้ ดังนั้นจะต้องหมุนข้อมูลภาพไปในทิศทางทวนเข็มนาฬิกา โดยกำหนดค่ามุมที่หาได้ให้เป็นค่าบวก เพื่อให้กลุ่มตัวเลขและพยัญชนะอยู่ในแนวราบมากขึ้น แต่ในทางตรงกันข้าม ถ้าค่าขอบซ้ายซึ่งเป็น Column ของกลุ่มพิกเซลสีขาวที่อยู่ติดกับขอบล่างของภาพ Binary มีค่าน้อยกว่า 0.5 เท่าของความกว้างของภาพ Binary แสดงว่าจุดต่ำสุดของกลุ่มตัวเลขและพยัญชนะอยู่ทางด้ายซ้ายมือ เมื่อไล่ไปทางด้านขวามือ จะพบว่าพยัญชนะและตัวเลขตัวถัดมาอยู่ห่างจากขอบล่างของภาพ Binary มากกว่า ดังนั้น จะต้องหมุนข้อมูลภาพไปในทิศทางตามเข็มนาฬิกา โดยการกำหนดค่ามุมที่หาได้ให้เป็นค่าลบ

จากภาพที่ 3.15 เป็นตัวอย่างภาพที่มีกลุ่มตัวเลขและพยัญชนะทำมุมเอียงกับแนวราบ ในภาพที่ 3.15 (ก) จะแสดงการหาขอบเขตของกลุ่มพิกเซลสีขาวที่ใช้เป็นแกนอ้างอิงในการหาความกว้างฐานและความสูงของรูปสามเหลี่ยม เพื่อที่จะหามุมที่กลุ่มตัวเลขและพยัญชนะเอียงทำมุมกับแนวราบ ซึ่งรูปสามเหลี่ยมที่ได้จะเป็นสามเหลี่ยมเส้นสีม่วงในภาพนี้ ในภาพที่ 3.15 (ข) จะเป็นภาพสามเหลี่ยมที่ขยายมาจากสามเหลี่ยมเส้นสีม่วงในภาพ 3.15 (ก) ซึ่งในภาพนี้จะแสดงการหามุมที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่มตัวเลขและพยัญชนะทำมุมกับแนวราบ จากภาพ 3.15 (ก) จะเห็นได้ว่าขอบซ้ายของกลุ่มพิกเซลสีขาวที่อยู่ติดกับขอบล่างของภาพ Binary มากที่สุด (ตัว ช) มีค่าขอบซ้ายที่เป็น Column น้อยกว่า 0.5 เท่าของของความกว้างของภาพ Binary ดังนั้นในการกำหนดทิศทางการหมุนให้กับโปรแกรมจะต้องกำหนดให้มุมเป็นค่าลบ เพื่อที่จะหมุนให้กลุ่มตัวเลขและพยัญชนะในภาพ Binary นี้กลับมาอยู่ในแนวราบมากขึ้นในทิศทางตามเข็มนาฬิกา ซึ่งภาพผลลัพธ์ที่ได้จากการหมุนจะแสดงในภาพที่ 3.15 (ค)

การทำงานทั้งหมดในขั้นตอนนี้เขียนเป็นคำสั่งได้ดังนี้

```
[labeled19,numObjects19] = bwlabel(PIX,4); %----ทำการ Label ภาพที่เก็บไว้ในตัวแปร PIX
for u19=1:numObjects19 %----ทำการตรวจสอบทุกกลุ่มพิกเซลสีขาว
[rr19,cc19] = find(labeled19==u19);
%----เก็บค่าขอบล่างของแต่ละกลุ่มพิกเซลสีขาวไว้ในตัวแปร ii19max
ii19max=max(rr19);
%----นับจำนวนขอบล่างของกลุ่มพิกเซลสีขาวทั้งหมด แล้วเก็บไว้ในตัวแปร count31
count31=count31+1;
%----นำค่าขอบล่างของกลุ่มพิกเซลสีขาวไปเก็บในรูปแบบของเมตริกซ์ในตัวแปร LLINE
LLINE(count31)=ii19max;
%----หาขอบบนของกลุ่มพิกเซลสีขาวแต่ละกลุ่ม แล้วเก็บไว้ในตัวแปร ii19min
ii19min=min(rr19);
%----หาขอบขวาของกลุ่มพิกเซลสีขาวแต่ละกลุ่ม แล้วเก็บไว้ในตัวแปร jj19max
jj19max=max(cc19);
%----นับจำนวนขอบขวาของกลุ่มพิกเซลสีขาวทั้งหมด แล้วเก็บไว้ในตัวแปร count32
count32=count32+1;
%----นำค่าขอบขวาของกลุ่มพิกเซลสีขาวไปเก็บในรูปแบบเมตริกซ์ในตัวแปร RIMM2
RIMM2(count32)=jj19max;
%----หาค่าขอบซ้ายของกลุ่มพิกเซลสีขาวแต่ละกลุ่ม แล้วเก็บไว้ในตัวแปร jj19min
jj19min=min(cc19);
```

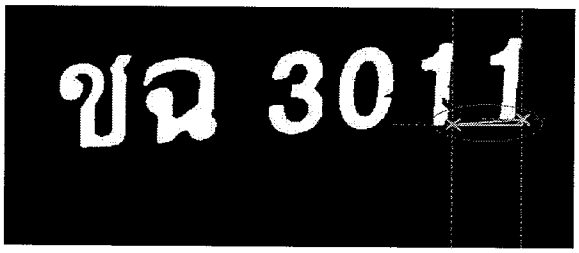
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----นับจำนวนขอบซ้ายของกลุ่มพิกเซลสีขาวทั้งหมด แล้วเก็บไว้ในตัวแปร count33
count33=count33+1;
%----นำค่าขอบซ้ายของกลุ่มพิกเซลสีขาวไปเก็บในรูปแบบเมตริกซ์ในตัวแปร RIMM1
RIMM1(count33)=jj19min;
end
%----ทำการ Label ภาพที่เก็บไว้ในตัวแปร PIXX
[labeled20,numObjects20] = bwlabel(PIX,4);
%----ตรวจสอบทุกกลุ่มพิกเซลสีขาว
for u20=1:numObjects20
   [rr20,cc20] = find(labeled20==u20);
%----หาขอบล่างของแต่ละกลุ่มพิกเซลสีขาว แล้วเก็บไว้ในตัวแปร ii20max
   ii20max=max(rr20);
%----ถ้ากลุ่มพิกเซลสีขาวกลุ่มไหนมีขอบล่างเท่ากับค่าที่มากที่สุดที่เก็บไว้ในเมริกซ์ LLINE
%----ให้เก็บหมายเลข Label ของกลุ่มพิกเซลสีขาวนั้นไว้ในตัวแปร D
   if ii20max==max(LLINE);
      D=u20;
%----หาขอบซ้ายของกลุ่มพิกเซลสีขาวที่มีหมายเลข Label ตรงกับค่าที่เก็บไว้ในตัวแปร D แล้ว
%----เก็บค่าขอบซ้ายที่หาได้ไว้ในตัวแปร jj20min
      [rr21,cc21] = find(labeled20==D);
      jj20min=min(cc20);
   end
end
[p,o]=size(PIX);
%----หาความยาวฐานของสามเหลี่ยม โดยนำค่าที่อยู่ Column สุดท้ายจากเมตริกซ์ RIMM2 ลบ
%----ออกด้วยค่าที่อยู่ Column รองสุดท้ายจากเมตริกซ์ RIMM1 แล้วเก็บค่าไว้ในตัวแปร RRIM
RRIM=RIMM2(1,count32)-RIMM1(1,count33-1);
```

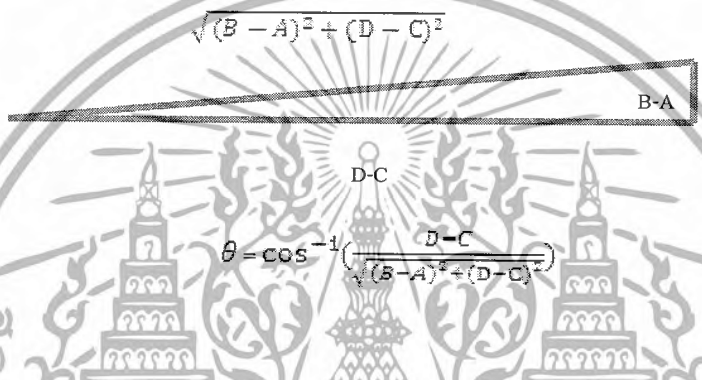
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----หาความสูงของสามเหลี่ยม โดยนำค่าที่อยู่ Column สุดท้ายจากเมตริกซ์ LLINE ลบออกด้วย
%---ค่าที่อยู่ Column รองสุดท้ายจากเมตริกซ์เดียวกัน แล้วเก็บค่าไว้ในตัวแปร LLINEE
LLINEE=LLINE(1,count31)-LLINE(1,count31-1);
%----หาความยาวของด้านตรงข้ามมุมฉาก แล้วเก็บไว้ในตัวแปร TOTALLINE
TOTALLINE=sqrt((LLINEE*LLINEE)+(RRIM*RRIM));
%----ถ้าค่าสัมบูรณ์ของค่าที่เก็บไว้ในตัวแปร LLINEE มากกว่า 0 ก็ไม่ต้องทำการหมุนกลุ่มตัว
%---- เลขและพยัญชนะให้กลับมาอยู่ในแนวราบ เพราะว่าขอบล่างของตัวเลขหรือพยัญนะ
%----ที่เลือกมา 2 ตัวมีค่าเท่ากัน นั่นคือตัวเลขและพยัญชนะทั้งสองที่เลือกมาอยู่บน Row เดียวกัน
%----ในภาพ
if abs(LLINEE)>0;
%----หามุมที่กลุ่มตัวเลขและตัวอักษรทำมุมกับแนวราบ แล้วเก็บค่ามุมในหน่วยองศาไว้ในตัว
%----แปร ANGLE
ANGLE=acosd(RRIM/TOTALLINE);
elseif LLINEE==0;
   ANGLE=0;
End
%----ถ้าค่าขอบซ้ายที่เก็บไว้ในตัวแปร jj20min มีค่ามากกว่า 0.5 เท่าของความกว้างของภาพที่เก็บ
%----ไว้ในตัวแปร PIX ให้หมุนภาพนี้ไปในทิศทางทวนเข็มนาฬิกา แต่ถ้าค่าที่เก็บไว้ในตัวแปร
%----jjmin20 มีค่าน้อยกว่า 0.5 เท่าของความกว้างของภาพที่เก็บไว้ในตัวแปร PIX ให้หมุนภาพ
%----นี้ไปในทิศทางตามเข็มนาฬิกา
if jj20min>(0.5*o)
   AANGLE=ANGLE;
else AANGLE=-1*ANGLE;
end
%----เก็บภาพที่ได้ทำการหมุนแล้วไว้ในตัวแปร PIXX
PIXX=imrotate(PIX,AANGLE);
```

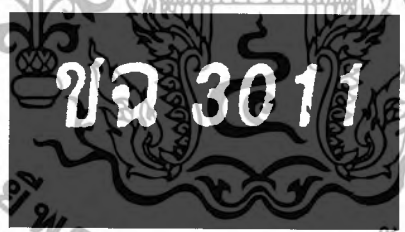
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก)



(ข)



(ค)

ภาพที่ 3.15 ตัวอย่างการหมุนกลุ่มตัวเลขและพยัญชนะที่มีความเอียง ให้กลับมาอยู่ในแนวราบมากขึ้น

(ก) ตัวอย่างภาพ Binary ที่มีกลุ่มตัวเลขและพยัญชนะเอียง

(ข) ขยายมุมที่กลุ่มตัวเลขและพยัญชนะทำกับแนวราบจากภาพ (ก)

(ค) ภาพที่ได้จากการหมุนกลุ่มตัวเลขและพยัญชนะให้กลับมาอยู่ในแนวราบ โดยใช้ค่ามุมที่หาได้จากภาพ (ข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การแบ่งส่วนที่เป็นตัวเลขและพยัญชนะออกมาจากแผ่นป้ายทะเบียนรถยนต์

เมื่อค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ได้แล้ว ขั้นตอนต่อไปก็จะเป็นการแยกตัวเลขและพยัญชนะทั้งหมดที่อยู่บนแผ่นป้ายออกมาทีละตัวโดยให้เรียงลำดับเหมือนบนแผ่นป้าย ซึ่งมีขั้นตอนทั้งหมดดังต่อไปนี้

1. หาแถว(Row) อ้างอิงที่จะตัดขอบบนและขอบล่างของกลุ่มตัวเลขและพยัญชนะทั้งหมดออกมาก่อน โดยให้ใช้เงื่อนไขที่ว่าถ้าแถวที่กำลังทำการสำรวจอยู่นั้นมีพิกเซลทุกพิกเซลเป็นสีดำหรือพิกเซลที่มีค่าเป็น 0 ทั้งหมด แต่แถวถัดไปอีกหนึ่งแถวมีพิกเซลที่ไม่ใช่พิกเซลสีดำทั้งหมด ถ้าเข้าเงื่อนไขนี้ ให้บันทึกค่าแถวอีกหนึ่งแถวถัดจากแถวที่เรากำลังทำการสำรวจอยู่ไว้ เพราะถ้าแถวที่มีคุณสมบัติตรงตามเงื่อนไขนี้ แถวถัดไปอีกหนึ่งแถวนั้นจะเป็นแกนอ้างอิงแนวนอนในการตัดขอบบนของกลุ่มตัวเลขและพยัญชนะออกมาจากแผ่นป้ายทะเบียนรถยนต์ และในการหาแถวที่เป็นแกนอ้างอิงในการตัดขอบล่างของกลุ่มตัวเลขและพยัญชนะ ให้ใช้เงื่อนไขที่ว่าถ้าแถวที่เรากำลังทำการสำรวจอยู่มีพิกเซลที่ไม่ใช่พิกเซลสีดำทุกพิกเซล แต่แถวถัดไปอีกหนึ่งแถวจากแถวที่เรากำลังทำการสำรวจมีพิกเซลทุกพิกเซลเป็นสีดำ ให้บันทึกค่าแถวที่เข้าเงื่อนไขนี้ไว้ ซึ่งแถวที่มีคุณสมบัติตรงตามเงื่อนไขในกรณีนี้ ก็คือแถวที่เป็นแกนอ้างอิงในการตัดขอบล่างของกลุ่มตัวเลขและพยัญชนะที่อยู่ในภาพแผ่นป้ายทะเบียนรถยนต์ (ดูที่เส้นประสีชมพู ภาพที่ 3.16) เมื่อได้แกนอ้างอิงในการตัดขอบบนและขอบล่างของกลุ่มตัวเลขและพยัญชนะแล้วก็นำไปแกนอ้างอิงทั้งสองนี้ ไปตัดข้อมูลภาพ Binary Image ที่ได้จากขั้นตอนที่ 12 ในหัวข้อ 3.1 ซึ่งภาพผลลัพธ์ที่ได้จากการนำแกนอ้าง 2 แกนนี้ไปคัดลอกส่วนของภาพมาจากภาพที่
3.16 ก็คือภาพที่ 3.17

ภาพที่ 3.16 แสดงแถวที่เป็นแกนอ้างอิงในการตัดขอบบนและขอบล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 3.17 ภาพที่ได้จากการใช้แกนอ้างอิงแนวนอนที่หาได้ไปคัดลอกส่วนของภาพจากภาพที่ 3.16

```
%----หาขนาดของภาพที่เก็บไว้ในตัวแปร BW2 แล้วเก็บค่าความกว้างไว้ในตัวแปร x และเก็บค่า
%----ความยาวไว้ในตัวแปร y
[x,y]=size(BW2);
for e =1:x-1
%----ถ้าแถวที่เรากำลังทำการสำรวจอยู่นี้เป็นพิกเซลสีขาวทั้งแถว และแถวถัดไปอีกหนึ่งแถวนั้น
%----ไม่ใช่พิกเซลสีขาวทั้งหมด ให้เก็บค่าแถวถัดไปอีกหนึ่งแถวนั้นไว้ในตัวแปร TOP ซึ่งใน
%----กรณีนี้ก็คือแถวนั้นเป็นขอบบนของกลุ่มตัวเลขและพยัญชนะ
   if sum(BW2(e,1:y))==0 && sum(BW2(e+1,1:y))~=0
      TOP=e+1;
%----อีกกรณีหนึ่ง ถ้าแถวที่เรากำลังทำการทำการสำรวจอยู่นี้ไม่เป็นพิกเซลสีขาวหมดทั้งแถว แต่
%----แถวถัดไปอีกหนึ่งแถวเป็นพิกเซลสีขาวหมดทั้งแถว ให้เก็บค่าแถวนั้นไว้ในตัวแปร LOW
%----ซึ่งถ้าเข้ากรณีนี้ก็คือแถวนั้นเป็นขอบล่างของกลุ่มตัวเลขและพยัญชนะ
elseif sum(BW2(e,1:y))~=0 && sum(BW2(e+1,1:y))==0
      LOW=e;
end
end
%----ใช้ค่าที่เก็บไว้ในตัวแปร TOP และ LOW ไปคัดลอกส่วนของภาพมาจากภาพที่เก็บไว้ใน
%----ตัวแปร BW2 แล้วเก็บภาพผลลัพทธ์ที่ได้จากการคัดลอกไว้ในตัวแปร another
another=BW2(TOP:LOW,1:y);
%----หาขนาดของภาพที่เก็บไว้ในตัวแปร another
[ll,kk]=size(another);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

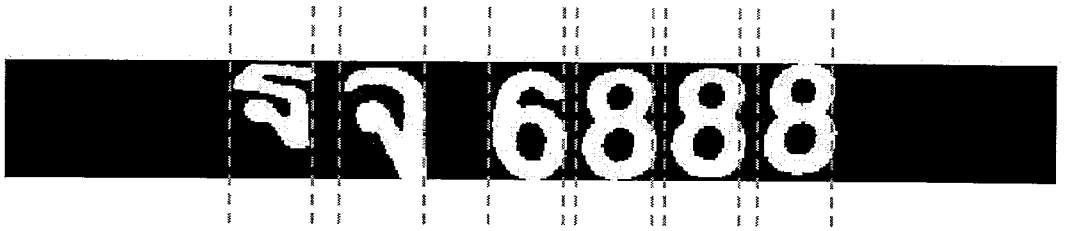
2. ทำการ Label ข้อมูลภาพ Binary Image ที่ได้จากขั้นตอนที่ 1 เพื่อแบ่งตัวเลขและพยัญชนะที่อยู่บนแผ่นป้ายทะเบียนรถยนต์ออกมาทีละตัว โดยทำการสำรวจทุกกลุ่มพิกเซลสีขาว แล้วเก็บค่าหลัก ที่น้อยสุดและมากที่สุดที่มีกลุ่มพิกเซลสีขาวกลุ่มที่กำลังทำการสำรวจอยู่ไว้ในรูปของเมตริกซ์ โดยเก็บค่าหลัก (Column) น้อยที่สุดไว้ที่เมตริกซ์หนึ่ง และเก็บหลักที่มากที่สุดไว้อีกเมตริกซ์หนึ่ง เขียนเป็นคำสั่งได้ดังนี้ จากภาพที่ 3.18 จะแสดงให้เห็นขอบซ้ายและขอบขวาของแต่ละกลุ่มพิกเซลสีขาว ซึ่งขอบซ้ายของแต่ละกลุ่มสีขาวจะแทนด้วยเส้นประสีชมพู และขอบขวาแทนด้วยเส้นประสีม่วง

```
%----ทำการ Label ภาพที่เก็บไว้ในตัวแปร another เพื่อหาขอบซ้ายและขอบขวาของตัวเลขและ
%----พยัญชนะบนป้ายทะเบียนรถ แล้วเก็บภาพผลลัพธ์ที่ได้จากการ Label ไว้ในตัวแปร labeled3
[labeled3,numObjects3] = bwlabel(another,4);
%----พิจารณาทุกกลุ่มพิกเซลสีขาว
for u3=1:numObjects3
[r3,c3] = find(labeled3==u3);
%----เก็บค่าขอบซ้ายของตัวเลขและพยัญชนะแต่ละตัวไว้ในตัวแปร j3min
j3min=min(c3);
%----นำค่าที่เก็บไว้ในตัวแปร j3min มาจัดเรียงให้อยู่ในรูปของเมตริกซ์แถวเดียว มีจำนวนหลัก
%----ตามจำนวนกลุ่มพิกเซลสีขาวหรือตัวเลขและพยัญชนะทั้งหมดที่อยู่บนแผ่นป้ายทะเบียน
%----(ซึ่งถูกเก็บไว้ในตัวแปร u3) ไว้ในตัวแปร left
left(u3)=j3min;
%----เก็บค่าขอบขวาของตัวเลขและตัวหนังสือแต่ละตัวไว้ในตัวแปร j3max
j3max=max(c3);
%----นำค่าที่เก็บไว้ในตัวแปร j3min มาจัดเรียงให้อยู่ในรูปของเมตริกซ์แถวเดียว มีจำนวนหลัก
%----ตามจำนวนกลุ่มพิกเซลสีขาวหรือตัวเลขและพยัญชนะทั้งหมดที่อยู่บนแผ่นป้ายทะเบียน
right(u3)=j3max;
end
%----ถ้ามีกลุ่มพิกเซลสีขาวเหลืออยู่ในภาพที่เก็บไว้ในตัวแปร labeled3 เป็นจำนวน 3 กลุ่ม
%----แสดงว่ามีตัวเลขและพยัญชนะบนแผ่นป้ายทะเบียนทั้งหมด 3 ตัว ให้นำค่าขอบซ้ายและ
%----ขอบขวาของทุกกลุ่มพิกเซลสีขาวที่เก็บไว้ในรูปของเมตริกซ์ไว้ในตัวแปร left และ right
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----มาจัดเรียงใหม่แบบเมตริกซ์ไว้ในตัวแปร T1 และ U1 ตามลำดับ
if u3==3
   T1=[left(1,1) left(1,2) left(1,3)];
   U1=[right(1,1) right(1,2) right(1,3)];
%----ถ้ามีกลุ่มพิกเซลสีขาวเหลืออยู่ในภาพที่เก็บไว้ในตัวแปร labeled3 เป็นจำนวน 4 กลุ่ม
%----แสดงว่ามีตัวเลขและพยัญชนะบนแผ่นป้ายทะเบียนทั้งหมด 4 ตัว
elseif u3==4
   T1=[left(1,1) left(1,2) left(1,3) left(1,4)];
   U1=[right(1,1) right(1,2) right(1,3) right(1,4)];
%----ถ้ามีกลุ่มพิกเซลสีขาวเหลืออยู่ในภาพที่เก็บไว้ในตัวแปร labeled3 เป็นจำนวน 5 กลุ่ม
%----แสดงว่ามีตัวเลขและพยัญชนะบนแผ่นป้ายทะเบียนทั้งหมด 5 ตัว
elseif u3==5
   T1=[left(1,1) left(1,2) left(1,3) left(1,4) left(1,5)];
   U1=[right(1,1) right(1,2) right(1,3) right(1,4) right(1,5)];
%----ถ้ามีกลุ่มพิกเซลสีขาวเหลืออยู่ในภาพที่เก็บไว้ในตัวแปร labeled3 เป็นจำนวน 6 กลุ่ม
%----แสดงว่ามีตัวเลขและพยัญชนะบนแผ่นป้ายทะเบียนทั้งหมด 6 ตัว
elseif u3==6
   T1=[left(1,1) left(1,2) left(1,3) left(1,4) left(1,5) left(1,6)];
   U1=[right(1,1) right(1,2) right(1,3) right(1,4) right(1,5) right(1,6)];
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.18 ขอบเขตด้านซ้าย (ดูเส้นประสีชมพู) และด้านขวา (ดูที่เส้นประสีม่วง) ของกลุ่มพิกเซล สีขาวที่ถูก Label ด้วยหมายเลขที่แตกต่างกัน

3. ค่า Column ที่น้อยที่สุดและมากที่สุดที่เป็นที่อยู่ของกลุ่มพิกเซลสีขาวทุกกลุ่ม ที่ถูกเก็บไว้ในรูปของเมตริกซ์อาจจะไม่เรียงลำดับจากน้อยไปหามาก แต่ในการแยกตัวเลขและพยัญชนะออกมาทีละตัวจากแผ่นป้ายทะเบียนนั้น จะต้องแยกออกมาโดยให้เรียงลำดับเหมือนบนแผ่นป้ายทะเบียน เพื่อนำไปสู่ขั้นตอนการรู้จำตัวเลขและพยัญชนะต่อไป ดังนั้นจะต้องเรียงลำดับจำนวนที่เป็นสมาชิกของทั้ง สองเมตริกซ์จากน้อยไปหามากก่อน และนำค่าที่น้อยที่สุดของทั้งสองเมตริกซ์ออกมาทีละคู่ เพื่อใช้เป็นขอบเขตในการแยกตัวเลขและพยัญชนะออกมาทีละตัวโดยให้เรียงลำดับเหมือนบนแผ่นป้าย คือให้นำค่าที่น้อยที่สุดออกมาจากเมตริกซ์ที่เก็บค่าหลักที่น้อยที่สุด ซึ่งที่เป็นที่อยู่ของกลุ่มพิกเซลสีขาวแต่ละกลุ่มไว้ จากนั้นให้นำค่าที่น้อยที่สุดออกมาจากเมตริกซ์ที่เก็บค่าหลักที่มากที่สุด ซึ่งเป็นที่อยู่ของกลุ่มพิกเซลสีขาวแต่ละกลุ่มไว้ และนำค่าที่น้อยที่สุดคู่นี้มาเป็นขอบเขตในการตัดลอกภาพจากขั้นตอนที่ 2 ก็จะทำให้ได้ตัวเลขหรือพยัญชนะตัวแรกที่อยู่บนแผ่นป้ายทะเบียนรถยนต์ออกมา เมื่อนำค่าที่น้อยที่สุดคู่นี้ออกมาใช้แล้วให้ลบค่าทั้ง 2 ค่านี้ออกจากแต่ละเมตริกซ์ เพื่อที่เราจะหาค่าที่น้อยที่สุดจากเมตริกซ์ทั้ง 2 คู่ต่อไป แล้วใช้เป็นขอบเขตในการตัดตัวเลขหรือพยัญชนะตัวต่อมาออกจากภาพในขั้นตอน 2 ให้ทำเช่นนี้ซ้ำต่อไปเรื่อย ๆ จนกว่าจะแยกตัวเลขและพยัญชนะที่มีอยู่บนแผ่นป้ายทะเบียนรถได้ออกมาทีละตัวจนครบทั้งหมด เขียนคำสั่งได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----หาค่าขอบซ้ายที่น้อยที่สุดที่เก็บไว้ในเมตริกซ์ T1 แล้วเก็บไว้ในตัวแปร cutpoint1 ซึ่งค่าขอบ
%----ซ้ายที่น้อยที่สุดนี้ก็คือขอบซ้ายของพยัญชนะตัวแรกที่อยู่บนแผ่นป้ายทะเบียน
cutpoint1=min(T1);
[row1,columm1]=size(T1);
for num1=1:columm1
%----ตรวจสอบหาหลักที่เป็นที่อยู่ของตัวแปร cutpoint1 ในเมตริกซ์ T1
        if T1(1,num1)==cutpoint1
%----เก็บค่าหลักที่เป็นที่อยู่ของตัวแปร coupoint1 ไว้ในตัวแปร point1
                point1=num1;
        end
end
%----หาค่าขอบขวาที่น้อยที่สุดที่เก็บไว้ในเมตริกซ์ U1 แล้วเก็บไว้ในตัวแปร cutpoint2 ซึ่งค่าขอบ
%----ขวาที่น้อยที่สุดนี้ก็คือขอบขวาของพยัญชนะตัวแรกที่อยู่บนแผ่นป้ายทะเบียน
cutpoint2=min(U1);
[row11,columm11]=size(U1);
for num11=1:columm11
%----ตรวจสอบหาหลักที่เป็นที่อยู่ของตัวแปร cutpoint2 ในเมตริกซ์ U1
        if U1(1,num11)==cutpoint2
%----เก็บค่าหลักที่เป็นที่อยู่ของตัวแปร cutpoint2 ไว้ในตัวแปร point11
                point11=num11;
        end
end
%----ลบค่าหลักที่เก็บไว้ในตัวแปร point 1 ออกไปจากเมตริกซ์ T1 เพราะในการที่จะหาขอบซ้าย
%----ของพยัญชนะตัวที่ 2 ที่อยู่บนแผ่นป้ายทะเบียน ก็คือหาค่าที่น้อยที่สุดที่ถัดจากค่าที่เก็บไว้ในตัว
%----แปร cutpoint1  ดังนั้นจะต้องลบค่าที่เก็บไว้ในตัวแปรนี้ออกจากเมตริกซ์ T1  ก่อนจึงจะหาได้
%---- ในการหาขอบขวาที่มีค่าน้อยที่สุดตัวถัดไปก็ทำแบบนี้เช่นกัน
T1(:,point1)=[];
%----นำเมตริกซ์ที่ลบค่าที่น้อยที่สุดตัวแรกออกไปแล้ว ไปเก็บไว้ในตัวแปร T2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
T2=T1;
%----หาค่าขอบซ้ายที่น้อยที่สุดในเมตริกซ์ T2 ซึ่งก็คือขอบซ้ายของพยัญชนะตัวที่ 2 ที่อยู่บนแผ่น
%---- ป้ายทะเบียน แล้วเก็บค่าขอบซ้ายนี้ไว้ในตัวแปร cutpoint3
cutpoint3=min(T2);
[row2,columm2]=size(T2);
for num2=1:columm2
%----ตรวจสอบหาหลักที่เป็นที่อยู่ของตัวแปร cutpoint3 ในเมตริกซ์ T2
        if T2(1,num2)==cutpoint3
%----เก็บค่าหลักที่เป็นที่อยู่ของตัวแปร cutpoint3 ไว้ในตัวแปร point2
                point2=num2;
        end
end
%----ลบค่าหลักที่เก็บไว้ในตัวแปร point 11 ออกไปจากเมตริกซ์ U1 เพื่อที่จะหาขอบขวา
%----ของตัวเลขตัวที่ 2 ที่อยู่บนแผ่นป้ายทะเบียน แล้วนำเมตริกซ์ที่ได้ไปเก็บไว้ในตัวแปร U2
U1(:,point11)=[];
U2=U1;
cutpoint4=min(U2);%----ขอบขวาของตัวเลขหรือพยัญชนะตัวที่สองที่อยู่บนแผ่นป้ายทะเบียน
[row22,columm22]=size(U2);
for num22=1:columm22
%----ตรวจสอบดูว่าตัวแปร cutpoint4 อยู่หลักใดของเมตริซ์ U2
        if U2(1,num22)==cutpoint4
%----เก็บค่าหลักที่เป็นที่อยู่ของตัวแปร cutpoint4 ไว้ในตัวแปร point22
                point22=num22;
        end
end
%----ลบค่าหลักที่เก็บไว้ในตัวแปร point 2 ออกไปจากเมตริกซ์ T3 เพื่อที่จะหาขอบซ้าย
%----ของตัวเลขตัวที่ 3 ที่อยู่บนแผ่นป้ายทะเบียน
 T2(:,point2)=[];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----นำเมตริกซ์ T2 ที่ลบค่าที่น้อยที่สุดตัวแรกออกไปแล้ว ไปเก็บไว้ในตัวแปร T3
T3=T2;
%----หาค่าขอบซ้ายที่น้อยที่สุดในเมตริกซ์ T3 ซึ่งก็คือขอบซ้ายของตัวเลขตัวที่ 3 ที่อยู่บนแผ่น
%---- ป้ายทะเบียน แล้วเก็บค่าขอบซ้ายนี้ไว้ในตัวแปร cutpoint5
cutpoint5=min(T3);
[row3,columm3]=size(T3);
for num3=1:columm3
%----ตรวจสอบหาหลักที่เป็นที่อยู่ของตัวแปร cutpoint5 ในเมตริกซ์ T3
        if T3(1,num3)==cutpoint5
                point3=num3; %----เก็บค่าหลักที่ได้ไว้ในตัวแปร point3
        end
end
%----ลบค่าหลักที่เก็บไว้ในตัวแปร point 22 ออกไปจากเมตริกซ์ U2 เพื่อที่จะหาขอบขวา
%----ของตัวเลขตัวที่ 3 ที่อยู่บนแผ่นป้ายทะเบียน แล้วนำเมตริกซ์ที่ได้ไปเก็บไว้ในตัวแปร U3
U2(:,point22)=[];
U3=U2;
%----หาค่าขอบขวาที่น้อยที่สุดในเมตริกซ์ U3 ซึ่งก็คือขอบขวาของตัวเลขตัวที่ 3 ที่อยู่บนแผ่น
%---- ป้ายทะเบียน แล้วเก็บค่าขอบขวานี้ไว้ในตัวแปร cutpoint6
cutpoint6=min(U3);
[row33,columm33]=size(U3);
for num33=1:columm33
%----ตรวจสอบหาหลักที่เป็นที่อยู่ของตัวแปร cutpoint6 ในเมตริกซ์ U3
        if U3(1,num33)==cutpoint6
                point33=num33;
        end
end
%----ถ้ามีตัวเลขและพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ทั้งหมด 3 ตัวให้หยุดไว้ที่ขั้นตอนนี้
if u3>3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----ลบค่าหลักที่เก็บไว้ในตัวแปร point 3 ออกไปจากเมตริกซ์ T3 เพื่อที่จะหาขอบซ้าย
%----ของตัวเลขตัวที่ 4 ที่อยู่บนแผ่นป้ายทะเบียน
T3(:,point3)=[];
%----นำเมตริกซ์ T3 ที่ลบค่าที่น้อยที่สุดตัวแรกออกไปแล้ว ไปเก็บไว้ในตัวแปร T4
T4=T3;
%----หาค่าขอบซ้ายที่น้อยที่สุดในเมตริกซ์ T4 ซึ่งก็คือขอบซ้ายของตัวเลขตัวที่ 4 ที่อยู่บนแผ่น
%---- ป้ายทะเบียน แล้วเก็บค่าขอบซ้ายนี้ไว้ในตัวแปร cutpoint7
cutpoint7=min(T4);
[row4,columm4]=size(T4);
for num4=1:columm4
%----ตรวจสอบหาหลักที่เป็นที่อยู่ของตัวแปร cutpoint7 ในเมตริกซ์ T4
        if T4(1,num4)==cutpoint7
                point4=num4; %----เก็บค่าหลักที่ได้ไว้ในตัวแปร point4
        end
end
%----ลบค่าหลักที่เก็บไว้ในตัวแปร point 33 ออกไปจากเมตริกซ์ U3 เพื่อที่จะหาขอบขวา
%----ของตัวเลขตัวที่ 4 ที่อยู่บนแผ่นป้ายทะเบียน แล้วนำเมตริกซ์ที่ได้ไปเก็บไว้ในตัวแปร U4
U3(:,point33)=[];
U4=U3;
%----หาค่าขอบขวาที่น้อยที่สุดในเมตริกซ์ U4 ซึ่งก็คือขอบขวาของตัวเลขตัวที่ 4 ที่อยู่บนแผ่น
%---- ป้ายทะเบียน แล้วเก็บค่าขอบขวานี้ไว้ในตัวแปร cutpoint8
cutpoint8=min(U4);
[row44,columm44]=size(U4);
for num44=1:columm44
%----ตรวจสอบหาหลักที่เป็นที่อยู่ของตัวแปร cutpoint8 ในเมตริกซ์ U4
        if U4(1,num44)==cutpoint8
                point44=num44; %----เก็บค่าหลักที่ได้ไว้ในตัวแปร point44
        end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end
%----ถ้ามีตัวเลขและพยัญชนะบนแผ่นป้ายทะเบียนทั้งหมด 4 ตัวให้หยุดไว้ที่ขั้นตอนนี้
if u3>4
%----ลบค่าหลักที่เก็บไว้ในตัวแปร point 4 ออกไปจากเมตริกซ์ T4 เพื่อที่จะหาขอบซ้าย
%----ของตัวเลขตัวที่ 5 ที่อยู่บนแผ่นป้ายทะเบียน
T4(:,point4)=[];
%----นำเมตริกซ์ T4 ที่ลบค่าที่น้อยที่สุดตัวแรกออกไปแล้ว ไปเก็บไว้ในตัวแปร T5
T5=T4;
%----หาค่าขอบซ้ายที่น้อยที่สุดในเมตริกซ์ T5 ซึ่งก็คือขอบซ้ายของตัวเลขตัวที่ 5 ที่อยู่บนแผ่น
%---- ป้ายทะเบียน แล้วเก็บค่าขอบซ้ายนี้ไว้ในตัวแปร cutpoint9
cutpoint9=min(T5);
[row5,columm5]=size(T5);
for num5=1:columm5
%----ตรวจสอบหาหลักที่เป็นที่อยู่ของตัวแปร cutpoint9 ในเมตริกซ์ T5
        if T5(1,num5)==cutpoint9
                point5=num5;
        end
end
%----ลบค่าหลักที่เก็บไว้ในตัวแปร point 44 ออกไปจากเมตริกซ์ U4 เพื่อที่จะหาขอบขวา
%----ของตัวเลขตัวที่ 5 ที่อยู่บนแผ่นป้ายทะเบียน แล้วนำเมตริกซ์ที่ได้ไปเก็บไว้ในตัวแปร U5
U4(:,point44)=[];
U5=U4;
%----หาค่าขอบขวาที่น้อยที่สุดในเมตริกซ์ U5 ซึ่งก็คือขอบขวาของตัวเลขตัวที่ 5 ที่อยู่บนแผ่น
%---- ป้ายทะเบียน แล้วเก็บค่าขอบขวานี้ไว้ในตัวแปร cutpoint10
cutpoint10=min(U5);
[row55,columm55]=size(U5);
for num55=1:columm55
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----ตรวจสอบดูว่าตัวแปร cutpoint10 อยู่หลักใดของเมตริกซ์ U5
        if U5(1,num55)==cutpoint10
                point55=num55; %----เก็บค่าหลักที่ได้ไว้ในตัวแปร point55
        end
end
end
%----ถ้ามีตัวเลขและพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ทั้งหมด 5 ตัวให้หยุดไว้ที่ขั้นตอนนี้
if u3>5
%----ลบค่าหลักที่เก็บไว้ในตัวแปร point 5 ออกไปจากเมตริกซ์ T5 เพื่อที่จะหาขอบซ้าย
%----ของตัวเลขตัวที่ 6 ที่อยู่บนแผ่นป้ายทะเบียน
T5(:,point5)=[];
%----นำเมตริกซ์ T5 ที่ลบค่าที่น้อยที่สุดตัวแรกออกไปแล้ว ไปเก็บไว้ในตัวแปร T6
T6=T5;
%----หาค่าขอบซ้ายที่น้อยที่สุดในเมตริกซ์ T6 ซึ่งก็คือขอบซ้ายของตัวเลขตัวที่ 6 ที่อยู่บนแผ่น
%---- ป้ายทะเบียน แล้วเก็บค่าขอบซ้ายนี้ไว้ในตัวแปร cutpoint11
cutpoint11=min(T6);
[row6,columm6]=size(T6);
for num6=1:columm6
%----ตรวจสอบหาหลักที่เป็นที่อยู่ของตัวแปร cutpoint11 ในเมตริกซ์ T6
        if T6(1,num6)==cutpoint11
                point6=num6; %----เก็บค่าหลักที่ได้ไว้ในตัวแปร point6
        end
end
%----ลบค่าหลักที่เก็บไว้ในตัวแปร point 55 ออกไปจากเมตริกซ์ U5 เพื่อที่จะหาขอบขวา
%----ของตัวเลขตัวที่ 6 ที่อยู่บนแผ่นป้ายทะเบียน แล้วนำเมตริกซ์ที่ได้ไปเก็บไว้ในตัวแปร U6
U5(:,point55)=[];
U6=U5;
%----หาค่าขอบขวาที่น้อยที่สุดในเมตริกซ์ U6 ซึ่งก็คือขอบขวาของตัวเลขตัวที่ 6 ที่อยู่บนแผ่น
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%---- ป้ายทะเบียน แล้วเก็บค่าขอบขวานี้ไว้ในตัวแปร cutpoint12
cutpoint12=min(U6);
[row66,columm66]=size(U6);
for num66=1:columm66
%----ตรวจสอบหาหลักที่เป็นที่อยู่ของตัวแปร cutpoint12 ในเมตริกซ์ U6
        if U6(1,num66)==cutpoint12
                point66=num66; %----เก็บค่าหลักที่ได้ไว้ในตัวแปร point66
        end
end
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ทำ Inverse Binary ให้กับภาพที่ 3.17 จะทำให้ได้ภาพ Binary ที่มีส่วนตัวเลขและพยัญชนะเป็นสีดำและพื้นหลังเป็นสีขาว ดังแสดงในภาพที่ 3.19

ภาพที่ 3.19 Inverse Binary ของภาพที่ 3.17

5. ใช้ค่า Column ที่เป็นขอบซ้ายและขอบขวาทั้งหมดที่ผ่านการเรียงลำดับจากน้อยไปหาน้อยไปหามากแล้วเป็นแกนอ้างอิง ในการคัดลอกพยัญชนะและตัวเลขทั้งหมดออกมาจากภาพที่ 3.19 โดยนำออกมาใช้เป็นคู่ ๆ จะทำให้สามารถคัดลอกภาพพยัญชนะและตัวเลขทั้งหมดออกมาได้ทีละภาพ โดยเรียงลำดับเหมือนบนแผ่นป้าย พร้อมกับตัดขอบบนและขอบล่างอีกครั้ง เขียนเป็นคำสั่งได้ดังนี้

```
newone=(~another); %----ทำ Inverse Binary ของภาพที่เก็บไว้ในตัวแปร another แล้ว
%----เก็บภาพผลลัพธ์ไว้ในตัวแปร newone
%----นำค่าที่น้อยที่สุดตัวแรกจากจากเมตริกซ์ที่เก็บค่า Column ที่เป็นขอบซ้ายและขอบขวาของ
%----ตัวเลขและพยัญชนะไว้ ซึ่งเก็บไว้ในตัวแปร cutpoint1 และ cutpoint2 มาเป็นขอบเขตในการ
%----แยกพยัญชนะตัวแรกที่อยู่บนแผ่นป้ายทะเบียนออกมา
letter1=newone(1:ll,cutpoint1:cutpoint2); %---- ll เป็นตัวแปรที่เก็บค่าความสูงของภาพที่เก็บไว้
%----ในตัวแปร newone
[x1,y1]=size(letter1);
%----ตัดขอบบนและขอบล่างของพยัญชนะที่แยกออกมาได้จากแผ่นป้ายทะเบียนอีกครั้ง
for xx1=1:x1-1
        if sum(letter1(xx1,1:y1)) == y1 && sum(letter1(xx1+1,1:y1)) ~= y1
                %----เก็บค่า Row ที่เป็นขอบบนไว้ในตัวแปร X1
%----ถ้าตัวเลขหรือพยัญชนะนั้นอยู่ชิดขอบบนของภาพพอดี แถวอ้างอิงในการตัดขอบบนก็คือ 1
    elseif sum(letter1(1,1:y1)) ~= y1
                X1=xx1+1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        X1=1;
      end
if sum(letter1(xx1,1:y1)) ~= y1 && sum(letter1(xx1+1,1:y1)) == y1
%----เก็บค่า Row ที่เป็นขอบล่างไว้ในตัวแปร X2
 X2=xx1;
%----ถ้าตัวเลขหรือพยัญชนะนั้นอยู่ชิดขอบล่างของภาพพอดี ค่า Row ที่ใช่เป็นขอบเขตในการตัด
%----ขอบล่างก็คือ Row สุดท้ายของภาพนั้น
 elseif sum(letter1(x1,1:y1)) ~= y1
        X2=x1;
     end
   end
%----ใช้ Row ที่เป็นขอบบนและขอบล่างของตัวเลขหรือพยัญชนะนั้นคัดลอก
%----ส่วนของภาพมาจาก
%-----ภาพที่เก็บไว้ในตัวแปร letter1 แล้วเก็บภาพผลลัพธ์ที่ได้จากการคัดลอกไว้ในตัวแปร
%----letter1neww
letter1neww=letter1(X1:X2,1:y1);
%----แสดงภาพที่ได้ที่เก็บไว้ในตัวแปร letter1neww โดยมี Title ของภาพว่า letter1
 figure(10),imshow(letter1neww);
 title('letter1');
%----นำค่าขอบซ้ายและขอบขวาที่เก็บไว้ในตัวแปร cutpoint3 และ cutpoint4 ตามลำดับ เป็น
%---Column อ้างอิงในการคัดลอกภาพพยัญชนะตัวที่ 2 ออกมาจากภาพแผ่นป้ายทะเบียน
letter2=newone(1:ll,cutpoint3:cutpoint4);
  [x2,y2]=size(letter2);
%----ตัดขอบบนและขอบล่างให้กับภาพพยัญชนะ
   for xx2=1:x2-1
     if sum(letter2(xx2,1:y2)) == y2 && sum(letter2(xx2+1,1:y2)) ~= y2
     X3=xx2+1;
 elseif sum(letter2(1,1:y2)) ~= y2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        X3=1;
    end
    if sum(letter2(xx2,1:y2)) ~= y2 && sum(letter2(xx2+1,1:y2)) == y2
        X4=xx2;
elseif sum(letter2(x2,1:y2)) ~= y2
        X4=x2;
      end
  end
%----ได้ภาพพยัญชนะตัวที่สองที่อยู่บนแผ่นป้ายทะเบียนออกมา พร้อมกับการตัดขอบบนและ
%----ขอบล่างเรียบร้อยแล้ว เก็บภาพนี้ไว้ในตัวแปร letter2neww
letter2neww=letter2(X3:X4,1:y2);
%----แสดงภาพที่ได้ที่เก็บไว้ในตัวแปร letter2neww โดยมี Title ของภาพว่า letter2
figure(11),imshow(letter2neww);
title('letter2');
%----นำค่าขอบซ้ายและขอบขวาที่เก็บไว้ในตัวแปร cutpoint5 และ cutpoint6 ตามลำดับ เป็น
%---Column อ้างอิงในการคัดลอกภาพตัวเลขตัวที่ 3 ออกมาจากภาพแผ่นป้ายทะเบียน
letter3=newone(1:ll,cutpoint5:cutpoint6);
[x3,y3]=size(letter3);
  for xx3=1:x3-1
    if sum(letter3(xx3,1:y3)) == y3&&&sum(letter3(xx3+1,1:y3))~= y3
      X5=xx3+1;
    elseif sum(letter3(1,1:y3)) ~= y3
      X5=1;
    end
if sum(letter3(xx3,1:y3))~= y3&& sum(letter3(xx3+1,1:y3))== y3
        X6=xx3;
    elseif sum(letter3(x3,1:y3))~= y3
X6=x3;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        end
    end
%----ได้ภาพตัวเลขตัวที่ 3 ที่อยู่บนแผ่นป้ายทะเบียนออกมา พร้อมกับการตัดขอบบนและ
%----ขอบล่างเรียบร้อยแล้ว เก็บภาพนี้ไว้ในตัวแปร letter3neww
letter3neww=letter3(X5:X6,1:y3);
%----แสดงภาพที่ได้ที่เก็บไว้ในตัวแปร letter3neww โดยมี Title ของภาพว่า letter3
figure(12),imshow(letter3neww);
title('letter3');
%----ถ้ามีตัวเลขและพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ทั้งหมด 3 ตัวให้หยุดไว้เพียงแค่นี้
if u3>3
%----นำค่าขอบซ้ายและขอบขวาที่เก็บ ไว้ในตัวแปร cutpoint7 และ cutpoint8 ตามลำดับ เป็น
%---Column อ้างอิงในการคัดลอกภาพตัวเลขตัวที่ 4 ออกมาจากภาพแผ่นป้ายทะเบียน
letter4=newone(1:ll,cutpoint7:cutpoint8);
[x4,y4]=size(letter4);
   for xx4=1:x4-1
%----ตัดขอบบนและขอบล่างให้กับภาพตัวเลข
    if sum(letter4(xx4,1:y4)) == y4&&sum(letter4(xx4+1,1:y4))~= y4
        X7=xx4+1;
    elseif sum(letter4(1,1:y4)) ~= y4
       X7=1;
end
    if sum(letter4(xx4,1:y4))~= y4&& sum(letter4(xx4+1,1:y4))== y4
         X8=xx4;
    elseif sum(letter4(x4,1:y4))~= y4
       X8=x4;
    end
   end
%----ได้ภาพตัวเลขตัวที่ 4 ที่อยู่บนแผ่นป้ายทะเบียนออกมา พร้อมกับการตัดขอบบนและ
%----ขอบล่างเรียบร้อยแล้ว เก็บภาพนี้ไว้ในตัวแปร letter4neww
```

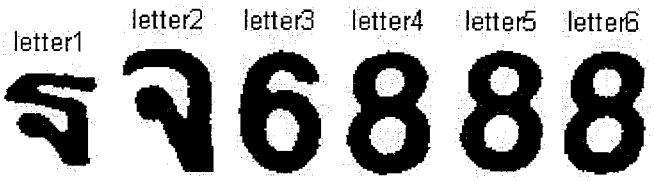
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
letter4neww=letter4(X7:X8,1:y4);
%----แสดงภาพที่ได้ที่เก็บไว้ในตัวแปร letter4neww โดยมี Title ของภาพว่า letter4
figure(13),imshow(letter4neww);
title('letter4');
end
%----ถ้ามีตัวเลขและพยัญชนะบนแผ่นป้ายทะเบียนทั้งหมด 4 ตัวให้หยุดไว้เพียงแค่นี้
if u3>4
%----นำค่าขอบซ้ายและขอบขวาที่เก็บไว้ในตัวแปร cutpoint9 และ cutpoint10 ตามลำดับ เป็น
%---Column อ้างอิงในการคัดลอกภาพตัวเลขตัวที่ 5 ออกมาจากภาพแผ่นป้ายทะเบียน
letter5=newone(1:ll,cutpoint9:cutpoint10);
[x5,y5]=size(letter5);
%----ตัดขอบบนและขอบล่างให้กับภาพตัวเลข
   for xx5=1:x5-1
    if sum(letter5(xx5,1:y5)) == y5&&sum(letter5(xx5+1,1:y5))~= y5
X9=xx5+1;
elseif sum(letter5(1,1:y5)) ~= y5
         X9=1;
 end
    if sum(letter5(xx5,1:y5))~= y5&& sum(letter5(xx5+1,1:y5))== y5
         X10=xx5;
 elseif sum(letter5(x5,1:y5))~= y5
       X10=x5;
end
end
%----ได้ภาพตัวเลขตัวที่ 5 ที่อยู่บนแผ่นป้ายทะเบียนออกมา พร้อมกับการตัดขอบบนและ
%----ขอบล่างเรียบร้อยแล้ว เก็บภาพนี้ไว้ในตัวแปร letter5neww
letter5neww=letter5(X9:X10,1:y5);
%----แสดงภาพที่ได้ที่เก็บไว้ในตัวแปร letter5neww โดยมี Title ของภาพว่า letter5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
figure(14),imshow(letter5neww);
title('letter5');
end
%----ถ้ามีตัวเลขและพยัญชนะบนแผ่นป้ายทะเบียน 5 ตัวให้หยุดไว้ที่ขั้นตอนนี้
if u3>5
%----นำค่าขอบซ้ายและขอบขวาที่เก็บไว้ในตัวแปร cutpoint11 และ cutpoint12 ตามลำดับ เป็น
%---Column อ้างอิงในการคัดลอกภาพตัวเลขตัวที่ 6 ออกมาจากภาพแผ่นป้ายทะเบียน
letter6=newone(1:ll,cutpoint11:cutpoint12);
  [x6,y6]=size(letter6);
%----ตัดขอบบนและขอบล่างให้กับภาพตัวเลข
   for xx6=1:x6-1
      if sum(letter6(xx6,1:y6)) == y6&&sum(letter6(xx6+1,1:y6))~= y6
         X11=xx6+1;
elseif sum(letter6(1,1:y6)) ~= y6
         X11=1;
      end
if sum(letter6(xx6,1:y6))~= y6&& sum(letter6(xx6+1,1:y6))== y6
         X12=xx6;
elseif sum(letter6(x6,1:y6))~= y6
         X12=x6;
      end
   end
%----ได้ภาพตัวเลขตัวที่ 6 ที่อยู่บนแผ่นป้ายทะเบียนออกมา พร้อมกับการตัดขอบบนและ
%----ขอบล่างเรียบร้อยแล้ว เก็บภาพนี้ไว้ในตัวแปร letter6neww
letter6neww=letter6(X11:X12,1:y6);
%----แสดงภาพที่ได้ที่เก็บไว้ในตัวแปร letter6neww โดยมี Title ของภาพว่า letter6
figure(15),imshow(letter6neww);
title('letter6');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.20 ตัวเลขและพยัญชนะที่แยกออกมาจากแผ่นป้ายทะเบียน โดยเรียงลำดับให้เหมือนกับบนแผ่นป้ายทะเบียน

3.3 การรู้จำตัวเลขและพยัญชนะ

เมื่อแยกภาพตัวเลข และพยัญชนะออกมาจากภาพแผ่นป้ายทะเบียนรถยนต์ได้ครบทุกตัว โดยเรียงลำดับให้เหมือนบนแผ่นป้ายทะเบียนแล้ว ให้นำข้อภาพตัวเลขและพยัญชนะทั้งหมดนี้มาเข้าสู่การรู้จำตัวเลและพยัญชนะโดยใช้เทคนิคที่เรียกว่า Template Matching ซึ่งทำได้โดยการนำตัวเลขหรือพยัญชนะนั้นมาทำการหาค่าสัมประสิทธิ์ความเหมือน หรือที่เรียกว่าค่า Correlation Coefficient กับตัวเลขและพยัญชนะทุกตัวที่มีอยู่ในข้อมูลภาพ ถ้าค่าสัมประสิทธิ์ความเหมือนที่เกิดจากการเปรียบเทียบกับตัวเลขและพยัญชนะกับ Template ตัวใดมีค่ามากที่สุด ให้พิมพ์ตัวเลขหรือพยัญชนะนั้นออกมาซึ่งมีขั้นตอนการทำงานของโปรแกรมดังนี้

3.3.1 การจัดทำตัวเลขและพยัญชนะทั้งหมดของชุด Template

ในตอนนี้เป็นการจัดทำตัวเลขและพยัญชนะทั้งหมด ที่ใช้เป็นชุดต้นแบบหรือชุด Template เพื่อนำไปวางทาบแล้วเปรียบเทียบหาค่าสัมประสิทธิ์ความเหมือนหรือค่า Correlation Coefficient กับตัวเลขและพยัญชนะที่แยกออกมาได้จากแผ่นป้ายทะเบียนรถยนต์ เพื่อประมวลผลตัวเลขและพยัญชนะที่แยกออกมาได้ให้เป็น Text File

ภาพตัวเลขและพยัญชนะทั้งหมดที่ใช้เป็นภาพต้นแบบในงานวิจัยนี้ ได้มาจากการเก็บภาพตัวอย่างจากแผ่นป้ายทะเบียนรถยนต์ โดยเลือกจากภาพถ่ายที่มีส่วนของกลุ่มตัวเลขและพยัญชนะที่ชัดและไม่เอียง เนื่องจากว่าผู้วิจัยได้ทำการติดต่อไปยังกรมขนส่งทางบกและโรงงานทำแผ่นป้ายทะเบียนแล้ว แต่ก็ไม่สามารถทราบ Font หรือรูปแบบ ที่ใช้ในการจัดทำตัวเลขและพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ได้ ภาพตัวเลขและพยัญชนะทั้งหมด ที่ใช้เป็นต้นแบบในงานวิจัยนี้มีทั้งหมด 97 ภาพ โดยมีภาพพยัญชนะ 74 ภาพ และภาพตัวเลข 23 ภาพ มีพยัญชนะที่ขาดไป 5 ตัว คือ ฏ ฑ ฒ ฝ และ ฆ เนื่องจากผู้วิจัยไม่สามารถจัดหาภาพแผ่นป้ายทะเบียนที่มีพยัญชนะดังกล่าวบนแผ่นป้ายได้ การจัดทำภาพตัวเลขและพยัญชนะต้นแบบมีขั้นตอนทั้งหมดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

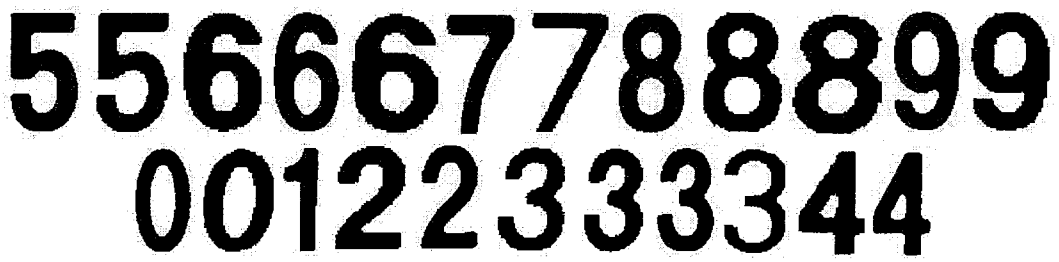
1. จัดหาภาพป้ายทะเบียนรถยนต์ที่ความสว่างปกติ หน้าตรง ตัวเลขและพยัญชนะที่อยู่บนแผ่นป้ายทะเบียนสมบูรณ์ คือตัวเลขและพยัญชนะมีความชัดเจน ไม่ขาดช่วงกัน ไม่มีส่วนที่เป็นน็อตหรือตะปูติดอยู่ ถ่ายภาพในระยะใกล้ คือประมาณ 0.5 เมตร

2. แปลงข้อมูลภาพในขั้นตอนที่ 1 จากภาพ RGB ให้เป็นภาพ Gray Scale และภาพ Binary ตามลำดับ

3. คัดลอกตัวเลขและพยัญชนะที่มีอยู่ในข้อมูลภาพ Binary Image แล้วเก็บไว้ใน File นามสกุล JPEG ภาพละตัว

4. นำตัวเลขและพยัญชนะที่เก็บไว้ทั้งหมดมาทำการตัดขอบบน ขอบล่าง ขอบซ้าย และขอบขวา แล้วทำการปรับขนาดให้ทุก ๆ ภาพมีความสูงเป็น 100 พิกเซล เพราะว่าในการทดลองนั้นจะเก็บภาพส่วนด้านหน้ารถยนต์ ที่ระยะประมาณ 1 เมตร เมื่อโปรแกรมสามารถค้นหาแผ่นป้ายทะเบียนและแยกตัวเลขและพยัญชนะทั้งหมด ออกมาจากแผ่นป้ายทะเบียนได้แล้วนั้น ภาพตัวเลขและพยัญชนะจะมีความสูงอยู่ที่ประมาณ 100 พิกเซล จากนั้นเก็บไว้เป็นภาพที่ให้ตัวเลขและพยัญชนะตัวอักษรเรียงกันเป็นแถวเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.21 ตัวเลขและพยัญชนะต้นแบบหรือชุด Template ทั้งหมดที่ใช้ในการทดลอง

3.3.2 ขั้นตอนการรู้จำตัวเลขและพยัญชนะ

1. รับภาพ Binary Image ที่เป็นตัวเลขหรือพยัญชนะตัวแรก จากขั้นตอนส่วนแรกที่อยู่บนแผ่นป้ายทะเบียนเข้ามา

ภาพที่ 3.22 ภาพ Input ตัวแรกที่เป็นภาพ Binary

2. แยกภาพพยัญชนะและตัวเลขที่อยู่ในภาพชุด Template ออกมาทีละภาพ เริ่มจากพยัญชนะตัวแรกไปจนถึงตัวเลขตัวสุดท้าย พร้อมกับตัดขอบบนและขอบล่างให้เรียบร้อย แล้วนำมาวางทาบกับตัวเลขหรือพยัญชนะนั้น เพื่อหาค่าสัมประสิทธิ์ความเหมือน แต่ในการที่จะหาค่าสัมประสิทธิ์ความเหมือนได้นั้น ขนาดของข้อมูลภาพทั้ง 2 จะต้องมีขนาดเท่ากัน ทั้งความสูงและความกว้าง ดังนั้น จะต้องทำการปรับขนาดความสูงของข้อมูลภาพตัวเลขหรือพยัญชนะที่ตัดได้มาจากแผ่นป้ายทะเบียนก่อน โดยจะปรับให้มีความสูงเท่ากับพยัญชนะที่ตัดได้ จากข้อมูลภาพชุด Template เมื่อภาพ Binary ทั้งสองมีความสูงที่เท่ากันแล้ว อาจจะมีความกว้างที่ไม่เท่ากันอีกก็ได้ ดังนั้นในการหาค่าสัมประสิทธิ์ความเหมือน ถ้าภาพอันไหนมีด้านกว้างมากกว่า เราจะพิจาณาถึงเฉพาะพิกเซลหลักที่เท่ากับความกว้างของภาพที่มีความกว้างน้อยกว่าเท่านั้น อย่างเช่น ภาพ A มีความกว้างของภาพเป็น 15 พิกเซล ส่วนภาพ B มีความกว้างของภาพเป็น 13 พิกเซล ดังนั้น ในการหาค่าสัมประสิทธิ์ความเหมือนระหว่างภาพสองภาพนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในภาพ A เราจะพิจารณาถึงเฉพาะด้านกว้างสิ้นสุดที่พิกเซลหลักที่ 13 เท่านั้น จากนั้นก็นำภาพตัวเลขหรือพยัญชนะตัวแรกที่คัดลอกได้มาจากภาพแผ่นป้ายตัวนั้น ไปหาค่าสัมประสิทธิ์ความเหมือนกับพยัญชนะตัวต่อไปบนข้อมูลภาพชุด Template จนถึงตัวเลขตัวสุดท้าย ทั้งหมด 97 ตัว เขียนคำสั่งได้ดังนี้

```
%----หาขนาดของภาพชุด Template ที่เก็บไว้ในตัวแปร template แล้วนำไปเก็บไว้ที่ตัวแปร xx
%----และ yy
 [xx,yy]=size(template);
for jj=1:yy-1
%----หา column ที่เป็นขอบซ้ายของตัวเลขและพยัญชนะแต่ละตัวที่อยู่บนภาพชุด Template
        if  sum(template(1:xx,jj))==xx && sum(template(1:xx,jj+1))~=xx
%----นับจำนวน Column ที่เป็นขอบซ้ายของตัวเลขและพยัญชนะทั้งหมดแล้วเก็บไว้ในตัวแปร
%----count2
                count2=count2+1;
%----นำ Column ที่เป็นขอบซ้ายของตัวเลขและพยัญชนะทั้งหมดมาเก็บไว้เป็นเมตริกซ์หนึ่งแถว มี
%----จำนวนหลักตามที่ตัวแปร count2 นับได้ในตัวแปร XX
                XX(count2)=jj+1;
%----หา column ที่เป็นขอบขวาของตัวเลขและพยัญชนะแต่ละตัวที่อยู่บนภาพชุด Template
        elseif sum(template(1:xx,jj))~=xx && sum(template(1:xx,jj+1))==xx
%----นับจำนวน Column ที่เป็นขอบขวาของตัวเลขและพยัญชนะทั้งหมดแล้วเก็บไว้ในตัวแปร
%----count4
                count4=count4+1;
%----นำ Column ที่เป็นขอบขวาของตัวเลขและพยัญชนะทั้งหมดมาเก็บไว้เป็นเมตริกซ์หนึ่งแถว มี
%----จำนวนหลักตามที่ตัวแปร count4 นับได้ในตัวแปร YY
                YY(count4)=jj;
        end
end
[x3,y3]=size(XX);
[x4,y4]=size(YY);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for j3=1:y3

%----ทีละตัวโดยใช้ค่าขอบซ้ายและขอบขวาที่เก็บไว้ในเมตริกซ์ XX และ YY ตามลำดับ โดยนำ
%----ออกมาใช้ทีละคู่ โดยเก็บภาพตัวเลขและพยัญชนะที่คัดลอกออกมาได้ไว้ในตัวแปร AA
        AA= template(1:xx,XX(1,j3):YY(1,j3));
        [m,n]=size(AA);
%----ตัดขอบบนและขอบล่างให้กับตัวเลขและพยัญชนะที่คัดลอกมาได้ ที่เก็บไว้ในตัวแปร AA
                for i1=1:m-1
                        if sum(AA(i1,1:n))==n  && sum(AA(i1+1,1:n))~= n
%----เก็บค่า row ที่เป็นขอบบนของตัวเลขและพยัญชนะที่คัดลอกมาได้ไว้ในตัวแปร TOP
                                TOP=i1+1;
                        elseif  sum(AA(i1,1:n))~=n  && sum(AA(i1+1,1:n))==n
%----เก็บค่า row ที่เป็นขอบล่างของตัวเลขและพยัญชนะที่คัดลอกมาได้ไว้ในตัวแปร LOW
                                LOW=i1;
                        end
                end
%----ใช้ค่า row ที่เป็นขอบบนและขอบล่างที่เก็บไว้ในตัวแปร TOP และ LOW ไปคัดลอกส่วนของ
%----ภาพมาจากภาพ AA ก็จะได้ภาพตัวเลขและพยัญชนะที่ผ่านการตัดขอบบนและขอบล่างแล้ว
%----เก็บภาพนี้ไว้ในตัวแปร B
B=AA(TOP:LOW,1:n);
%----รับภาพตัวเลขและพยัญชนะตัวแรกที่อยู่บนแผ่นป้ายทะเบียน ซึ่งเก็บไว้ในตัวแปร
%---- letter1neww เข้ามาแล้วเก็บไว้ในตัวแปร Input
Input=(letter1neww);
%----เก็บขนาดของภาพในตัวแปร Input เป็นความสูงและความกว้างไว้ในตัวแปร m และ n
%----ตามลำดับ
[m,n]=size(Input);
%----เก็บขนาดของภาพในตัวแปร B เป็นความสูงและความกว้างไว้ในตัวแปร x และ y
%---ตามลำดับ
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 [x,y]=size(B);
%----ทำการปรับขนาดภาพในตัวแปร Input ให้มีความสูงเท่ากับค่าที่เก็บไว้ในตัวแปร x แล้วเก็บ
%----ภาพผลลัพธ์ที่ปรับความสูงแล้วไว้ในตัวแปร new_Input
 new_Input=imresize(Input,[x NaN]);
%----หาขนาดของภาพในตัวแปร new_Input เป็นความสูงและความกว้าง โดยเก็บไว้ในตัวแปร
%----newm และ newn ตามลำดับ
[newm,newn]=size(new_Input);
%----นำภาพที่เก็บไว้ในตัวแปร new_Input มาวางทาบกับภาพตัวเลขและพยัญชนะทั้งหมดที่ตัด
%----จากภาพชุด Template ออกมาทีละตัวที่เก็บไว้ในตัวแปร B เพื่อหาค่า Correlation Coefficient
%----ถ้าความกว้างของภาพที่เก็บไว้ในตัวแปร new_Input มีค่าน้อยกว่าภาพที่เก็บไว้ในตัวแปร B
%----ในการนำภาพทั้งสองมาวางทาบกันเพื่อหาค่าสัมประสิทธิ์ความเหมือน สำหรับภาพที่เก็บไว้
%----ในตัวแปร B จะพิจารณาด้านกว้างถึงเฉพาะ Column ตามค่าที่เก็บไว้ในตัวแปร newn
if newn<y
        X=corr2(new_Input,B(1:x,1:newn)); %---เก็บค่าที่ได้ไว้ในตัวแปร X
%----ถ้าความกว้างของภาพที่เก็บไว้ในตัวแปร B มีค่าน้อยกว่าภาพที่เก็บไว้ในตัวแปร new_Input
%----ในการนำภาพทั้งสองมาวางทาบกันเพื่อหาค่าสัมประสิทธิ์ความเหมือน สำหรับภาพที่เก็บไว้
%----ในตัวแปร new_Input จะพิจารณาด้านกว้างถึงเฉพาะ Column ตามค่าที่เก็บไว้ในตัวแปร
%----newn
elseif y<newn
        X=corr2(B,new_Input(1:newm,1:y));
end
%----ถ้าภาพที่เก็บไว้ในตัวแปร Input และ B มีขนาดเท่ากันพอดี ในการหาค่าสัมประสิทธิ์ความ
%----เหมือนก็สามารถนำภาพทั้งสองไปวางทาบกันได้เลย
if m==x && n==y
        X=corr2(Input,B);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%---ถ้าภาพที่เก็บไว้ในตัวแปร B และ  Input มีความสูงเท่ากัน แต่ถ้าภาพที่เก็บไว้ในตัวแปร B
%----มีค่าความกว้างมากกว่าภาพที่เก็บไว้ในตัวแปร Input ในการหาค่าสัมประสิทธิ์ความเหมือน
%----สำหรับภาพที่เก็บไว้ในตัวแปร B จะพิจารณาถึงเฉพาะด้านกว้างตามค่า Column ถึงค่าที่
%----เก็บไว้ในตัวแปร n เท่านั้น
elseif m==x && n<y
        X=corr2(Input,B(1:x,1:n));
%---ถ้าภาพที่เก็บไว้ในตัวแปร B และ  Input มีความสูงเท่ากัน แต่ถ้าภาพที่เก็บไว้ในตัวแปร Input
%----มีค่าความกว้างมากกว่าภาพที่เก็บไว้ในตัวแปร Bในการหาค่าสัมประสิทธิ์ความเหมือน
%----สำหรับภาพที่เก็บไว้ในตัวแปร Input จะพิจารณาถึงเฉพาะด้านกว้างตามค่า Column ถึงค่าที่
%----เก็บไว้ในตัวแปร y เท่านั้น
elseif m==x && y<n
        X=corr2(B,Input(1:m,1:y));
end
%----นำค่าสัมประสิทธิ์ความเหมือนที่ได้จากการนำภาพพยัญชนะตัวแรกที่อยู่บนแผ่นป้าย
%----ทะเบียนไปวางทาบกับตัวเลขและพยัญชนะจากภาพชุด Template ทุกตัวมาเก็บไว้ใน
%----จำนวนเมตริกซ์ corr ซึ่งเป็นเมตริกซ์หนึ่งแถว มีจำนวนหลักตามค่าที่เก็บไว้ในตัวแปร j3
%----ซึ่งเป็น Column ที่เป็นขอบซ้ายของตัวเลขและพยัญชนะทั้งหมดที่อยู่ในภาพชุด Template
corr(j3)=(X);
%----หาค่าที่มากที่สุดจากเมตริกซ์ corr แล้วเก็บไว้ในตัวแปร correlation
correlation=max(corr);
end
%----นำเมตริกซ์ corr ไปเก็บไว้ในตัวแปร CORR
CORR=corr;
%----ถ้าอัตราส่วนด้านกว้างต่อด้านยาวของภาพที่เก็บไว้ในตัวแปร B (ภาพพยัญชนะตัวแรกที่อยู่
%----บนแผ่นป้ายทะเบียน) มีค่าน้อยกว่าหรือเท่ากับ 0.25 ให้ค่าสัมประสิทธิ์ความเหมือนเมื่อนำ
%----ภาพนี้ไปวางทาบกับภาพตัวเลข 1 ซึ่งอยู่ตำแหน่งที่ 77 ของภาพชุด Template มีค่าเป็น  1
if n/m<=0.25
   CORR(1,77)=1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----ถ้าอัตราส่วนด้านกว้างต่อด้านยาวของภาพที่เก็บไว้ในตัวแปร B (ภาพพยัญชนะตัวแรกที่อยู่
%----บนแผ่นป้ายทะเบียน) มีค่ามากกว่า 0.25 ให้ค่าสัมประสิทธิ์ความเหมือนเมื่อนำภาพนี้ไปวาง
%----ทาบกับภาพตัวเลข 1 ซึ่งอยู่ตำแหน่งที่ 77 ของภาพชุด Template มีค่าเป็น  0
elseif n/m > 0.25
   CORR(1,77)=0;
 end
%----ตรวจสอบดูว่าค่าสัมประสิทธิ์ความเหมือนที่มากสุดที่เก็บไว้ในตัวแปร CORRELATION
%----อยู่ตรง Columnไหนของเมตริกซ์ CORR ซึ่งค่า Column นี้ สามารถบอกได้ว่าตัวเลขและ
%----พยัญชนะที่ตัดออกมาจากแผ่นป้ายทะเบียนนั้นมีค่าความเหมือนกับตัวเลขและพยัญชนะ
%----ของ Template ตัวไหนมากที่สุด แล้วเก็บค่า Column ที่ได้ไว้ในตัวแปร W1
CORRELATION=max(CORR)
[q,w]=size(CORR);
for ww=1:w
   if CORR(1,ww)==CORRELATION;
      W1=ww;
   end
end
%----ถ้าค่าสัมประสิทธิ์ความเหมือนที่เก็บ ไว้ในตัวแปร CORRELATION มีค่ามากกว่า 0.6
%----โปรแกรมจะทำงานในส่วนนี้ คือจะแสดงผลการรู้จำภาพพยัญชนะตัวนี้ทางหน้าต่าง
 %----Command Window
if CORRELATION>=0.6
%----ถ้าค่าสัมประสิทธิ์ความเหมือนที่มีค่ามากที่สุดอยู่ที่ Column 1 หรือ 2 หรือ  3 ของเมตริกซ์
%----CORR แสดงว่าภาพพยัญชนะตัวนี้มีค่าความเหมือนมากที่สุดเมื่อนำไปวางทาบหาค่า
%----สัมประสิทธิ์ความเหมือนกันภาพพยัญชนะ ก ของภาพชุด Template ให้แสดงผลการรู้
%----จำเป็นตัวพยัญชนะ ก ทางหน้าต่าง Command Window
if W1==1 || W1==2 || W1==3
   fprintf('ก')
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
elseif W1==4 || W1==5||W1==6
    fprintf('ข')
elseif W1==7 || W1==8
    fprintf('ค')
elseif W1==9||W1==10||W1==11||W1==12
    fprintf('ช')
elseif W1==13||W1==14
    fprintf('ฌ')
elseif W1==15||W1==16
    fprintf('ณ')
elseif W1==17||W1==18||W1==19
    fprintf('บ')
elseif W1==20||W1==21
    fprintf('ย')
elseif W1==22||W1==23
    fprintf('ศ')
elseif W1==24||W1==25||W1==26
    fprintf('อ')
elseif W1==27||W1==28||W1==29
    fprintf('ท')
elseif W1==30||W1==31
    fprintf('ย')
elseif W1==32||W1==33
    fprintf('ห')
elseif W1==34||W1==35
    fprintf('ส')
elseif W1==36||W1==37
    fprintf('น')
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
elseif W1==38||W1==39
    fprintf('ร')
elseif W1==40
    fprintf('ล')
elseif W1==41||W1==42
    fprintf('ว')
elseif W1==43||W1==44
    fprintf('ศ')
elseif W1==45||W1==46
    fprintf('ต')
elseif W1==47||W1==48
    fprintf('ย')
elseif W1==49||W1==50
    fprintf('ว')
elseif W1==51
    fprintf('ห')
elseif W1==52||W1==53
    fprintf('พ')
elseif W1==54||W1==55
    fprintf('ผ')
elseif W1==56
    fprintf('ป')
elseif W1==57
    fprintf('ง')
elseif W1==58||W1==59
    fprintf('จ')
elseif W1==60||W1==61
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
elseif W1==62
    fprintf('ฎ')
elseif W1==63||W1==64
    fprintf('ฉ')
elseif W1==65
    fprintf('ม')
elseif W1==66||W1==67
    fprintf('ธ')
elseif W1==68
    fprintf('ฌ')
elseif W1==69
    fprintf('ล')
elseif W1==70
    fprintf('ป')
elseif W1==71
    fprintf('ห')
elseif W1==72
    fprintf('ฟ')
elseif W1==73
    fprintf('ศ')
elseif W1==74
    fprintf('ต')
elseif W1==75||W1==76
    fprintf('0')
elseif W1==77
    fprintf('1')
elseif W1==78||W1==79
    fprintf('2')
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
elseif W1==80||W1==81||W1==82||W1==83
    fprintf('3')
elseif W1==84||W1==85
    fprintf('4')
elseif W1==86||W1==87
    fprintf('5')
elseif W1==88||W1==89||W1==90
    fprintf('6')
elseif W1==91||W1==91
    fprintf('7')
elseif W1==93||W1==94||W1==95
    fprintf('8')
elseif W1==96||W1==97
    fprintf('9')
end
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CORR =

Columns 1 through 12

0.0505 0.2099 0.1788 0.1634 0.2155 0.1901 0.0904 0.2072 0.0678 0.1042 0.1130 0.1187

Columns 13 through 24

-0.0153 0.0114 0.1143 0.0448 -0.0175 0.1745 0.0007 0.0903 -0.0469 0.0692 0.0254 -0.0256

Columns 25 through 36

0.0846 -0.0315 0.1554 0.3106 0.2016 0.0622 -0.0410 0.1789 0.2350 0.0024 -0.0035 0.0995

Columns 37 through 48

0.2390 0.3735 0.2323 0.0192 0.0916 -0.0038 0.1619 0.2632 0.0351 0.0821 0.0599 0.0223

Columns 49 through 60

0.3557 0.2094 0.0519 0.2288 0.2806 -0.1675 -0.1597 0.0281 0.0270 0.2945 0.4609 0.5090

Columns 61 through 72

0.8269 -0.0061 -0.0266 0.1738 0.0198 0.1129 0.2787 -0.0742 0.0764 0.1204 0.2990 0.2306

Columns 73 through 84

0.0135 0.1215 -0.0578 -0.1180 0 -0.0065 0.1043 0.1427 0.2070 0.2024 0.2053 0.0061

Columns 85 through 96

0.0285 -0.0598 -0.0470 -0.0960 -0.0728 -0.1824 0.2095 0.1589 0.0960 0.0285 -0.0130 0.1491

Column 97

0.1245

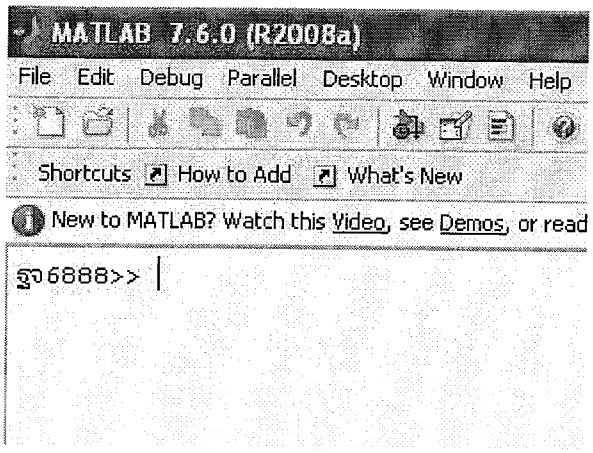
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CORRELATION =
0.8269
W1 =
61
```

ภาพที่ 3.23 ผลที่ได้จากการนำภาพตัว ฐ ไปประมวลผลในส่วนการรู้จำตัวเลขและพยัญชนะ จากภาพตัวแปร CORR จะเป็นเมตริกซ์ที่เก็บค่าสัมประสิทธิ์ความเหมือนที่ได้จากการนำภาพตัว ฐ ไปวางทางกับภาพตัวเลขและพยัญชนะทุกตัว ซึ่งผลที่ได้จะเห็นได้ว่าค่าสัมประสิทธิ์ความเหมือนที่มากสุดอยู่ที่หลัก (ซึ่งถูกเก็บไว้ในตัวแปร W1) ที่ 61 ของเมตริกซ์ที่เก็บค่าสัมประสิทธิ์ความเหมือนไว้ นั่นแสดงว่าข้อมูลภาพ Input ที่รับเข้ามามีความเหมือน กับตัวเลขหรือตัวอักษรที่ 61 ของข้อมูลภาพชุด Template มากที่สุด ซึ่งในโปรแกรมได้กำหนดไว้ว่าถ้า W1 มีค่าเท่ากับ 61 ให้แสดงผลการรู้จำเป็น ฐ ในหน้าต่าง Command

3. เมื่อนำภาพตัวเลขหรือพยัญชนะตัวแรก ที่อยู่บนแผ่นป้ายทะเบียนมาเข้าสู่การรู้จำแล้ว ให้นำภาพตัวเลขและพยัญชนะตัวต่อมาที่อยู่บนแผ่นป้ายทะเบียน มาเข้าสู่การรู้จำอีกต่อไปจนครบทุกตัว โดยมีขั้นตอนเหมือนกับขั้นตอนที่ 1-3 เพียงแต่ถ้าถึงตัวที่ 3 ของแผ่นป้ายทะเบียนรถยนต์แล้ว ให้ใช้ภาพชุด Template ภาพใหม่ที่มีแต่ภาพตัวเลขเท่านั้น เพราะตัวที่ 3 เป็นต้นไปในแผ่นป้ายทะเบียนรถยนต์จะเป็นตัวเลขเท่านั้น ทั้งนี้ก็เพื่อให้การประมวลผลมีความรวดเร็วยิ่งขึ้น ซึ่งผลที่ได้จากการรู้จำตัวเลขและพยัญชนะทั้งหมดจะแสดงในหน้าต่าง Command Window ของโปรแกรม MATLAB ดังแสดงในภาพที่ 3.24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.24 หน้าต่าง Command Window ที่เป็นผลมาจากการนำข้อมูลภาพตัวเลขและพยัญชนะทุกตัวที่อยู่บนแผ่นป้ายทะเบียนรถยนต์มาเข้าสู่การรู้จำตัวเลขและพยัญชนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

การทดลองแบ่งเป็น 2 ส่วน คือส่วนแยกตัวเลขและพยัญชนะออกจากแผ่นป้ายทะเบียนและการรู้จำแผ่นป้ายทะเบียนรถยนต์ โดยในการทดลองโปรแกรมจะรับภาพถ่ายส่วนด้านหน้ารถยนต์จากกล้องมือถือ ซึ่งเป็นภาพแบบ RGB สภาพแวดล้อมขณะทำการถ่ายภาพเป็นสภาพแวดล้อมในที่ร่ม ไม่มีแสงแดดจ้าจนเกินไป ระยะถ่ายภาพอยู่ที่ประมาณ 1 เมตร

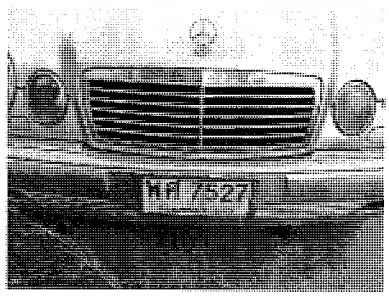
4.1 การแยกตัวเลขและพยัญชนะออกจากแผ่นป้ายทะเบียน

การทดลองในตอนนี้ เมื่อโปรแกรมสามารถค้นหาแผ่นป้ายทะเบียนได้แล้ว โปรแกรมจะทำการแยกตัวเลขและพยัญชนะออกจากแผ่นป้ายทะเบียนทีละตัวจนครบ ซึ่งขั้นตอนในการทำการทดลองมีดังนี้

1. รับภาพถ่ายส่วนด้านหน้าของรถยนต์เข้ามา ซึ่งเป็นภาพแบบ RGB จะได้ผลดังภาพที่ 4.1

ภาพที่ 4.1 ภาพถ่ายส่วนด้านหน้ารถยนต์แบบ RGB

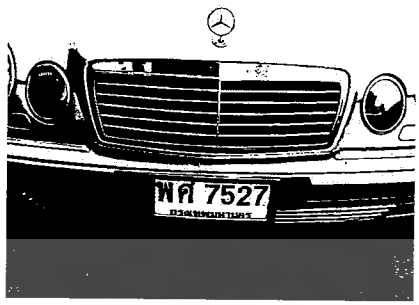
2. แปลงภาพ RGB ให้เป็นภาพ Gray Scale จะได้ผลดังภาพที่ 4.2



ภาพที่ 4.2 ภาพที่ได้จากการแปลงภาพ RGB ให้เป็นภาพ Gray Scale

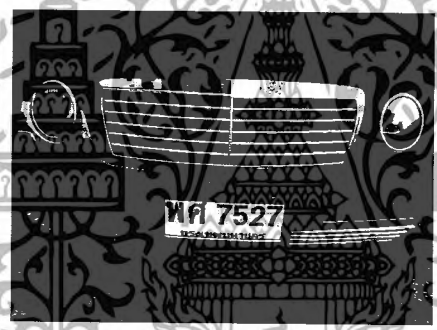
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. แปลงภาพจาก Gray Scale ให้เป็นภาพ Binary จะได้ผลดังภาพที่ 4.3



ภาพที่ 4.3 ภาพที่ได้จากการแปลงภาพ Gray Scale ให้เป็นภาพ Binary

4. กำจัดกลุ่มพิกเซลสีขาวที่ติดกับขอบภาพออกไป จะได้ผลดังภาพที่ 4.4



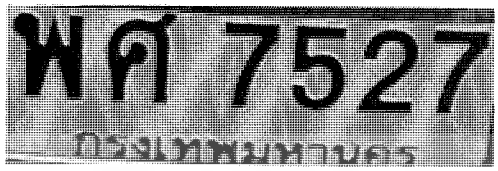
ภาพที่ 4.4 ภาพผลลัพธ์ที่ได้จากการกำจัดกลุ่มพิกเซลสีขาวที่ติดกับขอบออกไป

5. ทำให้ภาพคมชัดขึ้น และกำจัดสิ่งรบกวนออกไป จะได้ผลดังภาพที่ 4.5

ภาพที่ 4.5 ภาพผลลัพธ์ที่ได้จากการทำให้ภาพคมชัดขึ้นและกำจัดสิ่งรบกวนออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. หากลุ่มพิกเซลสีขาวที่ใหญ่ที่สุดในภาพที่ 4.5 จากขั้นตอนที่ 5 แล้วนำคู่พิกัดของกลุ่มพิกเซลสีขาวนี้ไปคัดลอกส่วนของภาพมาจากภาพ RGB ที่รับเข้ามาในขั้นตอนที่ 1 จะได้ผลดังภาพที่ 4.6



ภาพที่ 4.6 ภาพ RGB ที่ได้จากการนำคู่พิกัดของกลุ่มพิกเซลสีขาวที่ใหญ่ที่จากภาพที่ 4.6 ไปคัดลอกส่วนของภาพจากภาพ RGB จากขั้นตอนที่ 1

7. ทำ Inverse Binary ให้กับภาพที่ 4.6 จากขั้นตอนที่ 6 เพื่อที่จะตรวจสอบว่าภาพ RGB นี้คือส่วนที่เป็นแผ่นป้ายทะเบียนหรือไม่ จะได้ผลดังภาพที่ 4.7 เมื่อตรวจสอบแล้วพบว่าส่วนของภาพที่คัดลอกมานี้คือส่วนที่เป็นแผ่นป้ายทะเบียน ให้นำภาพ Inverse Binary นี้กำจัดส่วนที่เป็นชื่อจังหวัด และส่วนอื่น ๆ ที่ไม่ใช่ตัวเลขและพยัญชนะแถวบน ลวดลายต่าง ๆ ของแผ่นป้ายทะเบียน และกำจัดกลุ่มพิกเซลสีขาวที่ติดกับขอบออกไป จะได้ผลดังภาพที่ 4.8

ภาพที่ 4.7 ภาพ Inverse Binary ของภาพที่ 4.8

ภาพที่ 4.8 ภาพผลลัพธ์ที่ได้จากการนำภาพที่ 4.7 ไปกำจัดส่วนที่เป็นชื่อจังหวัด ลวดลายต่าง ๆ ที่อยู่บนแผ่นป้ายทะเบียน และกลุ่มพิกเซลสีขาวที่ติดกับขอบออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. เมื่อได้ภาพ Binary ที่เหลือเฉพาะส่วนที่เป็นตัวเลขและพยัญชนะแถวบนของแผ่นป้ายทะเบียนแล้ว ให้หมุนกลุ่มตัวเลขและพยัญชนะนี้ให้กลับมาอยู่ในแนวนอนมากขึ้น จะได้ผลดังภาพที่ 4.9

ภาพที่ 4.9 ภาพผลลัพธ์ที่ได้จากการหมุนกลุ่มตัวเลขและพยัญชนะให้กลับมาอยู่ในแนวราบมากขึ้น

9. ตัดขอบบนและขอบล่างให้กับกลุ่มตัวเลขและพยัญชนะจากภาพที่ 4.9 ขั้นตอนที่ 8 จะได้ผลดังภาพที่ 4.10

ภาพที่ 4.10 ภาพผลลัพธ์ที่ได้จากการนำภาพที่ 4.9 ไปตัดขอบบนและขอบล่างให้กับกลุ่มตัวเลขและพยัญชนะ

10. ทำการ Label เพื่อหา Column ที่เป็นขอบซ้ายและขอบขวาของตัวเลขและพยัญชนะแต่ละตัวจากภาพที่ 4.10 ในขั้นตอนที่ 9 โดยเก็บค่า Column ที่เป็นขอบซ้ายไว้ในเมตริกซ์ T1 และเก็บค่า Column ที่เป็นขอบขวาไว้ในเมตริกซ์ U1 ซึ่งผลจากโปรแกรมที่ได้จากการนำ Column ที่เป็นขอบซ้ายและขอบขวาไปเก็บไว้ในแต่ละเมตริกซ์จะได้ผลดังภาพที่ 4.11

```
T1 =
    5   74  159  209  257  313
U1 =
   56  123  199  248  300  353
```

ภาพที่ 4.11 ผลที่ได้จากการนำ Column ที่เป็นขอบซ้ายและขอบขวาของตัวเลขและพยัญชนะไปเก็บในรูปของเมตริกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. นำค่า Column ที่เก็บไว้ในเมตริกซ์ T1 และ U1 มาเรียงลำดับจากน้อยไปหามาก ซึ่งผลที่ได้จากโปรแกรมดูได้จากภาพที่ 4.12

```
cutpoint1 =
     5
point1 =
     1
cutpoint2 =
    56
point11 =
     1
T1 =
    74   159   209   257   313
T2 =
    74   159   209   257   313
cutpoint3 =
    74
point2 =
     1
U1 =
   123   199   248   300   353
U2 =
   123   199   248   300   353
cutpoint4 =
   123
point22 =
     1
T2 =
   159   209   257   313
T3 =
   159   209   257   313
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
cutpoint5 =
   159
point3 =
    1
U2 =
   199   248   300   353
U3 =
   199   248   300   353
cutpoint6 =
   199
point33 =
    1
T3 =
   209   257   313
T4 =
   209   257   313
cutpoint7 =
   209
point4 =
    1
U3 =
   248   300   353
U4 =
   248   300   353
cutpoint8 =
   248
point44 =
    1
   T4 =
   257   313
T5 =
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   257   313
cutpoint9 =
   257
point5 =
     1
U4 =
   300   353
U5 =
   300   353
point55 =
     1
T5 =
   313
T6 =
   313
cutpoint11 =
   313
point6 =
     1
U5 =
   353
U6 =
   353
cutpoint12 =
   353
```

ภาพที่ 4.12 ผลจากโปรแกรมที่ได้จากการนำค่า Column ที่เป็นขอบซ้ายและขอบขวาของตัวเลขและพยัญชนะมาเรียงลำดับจากน้อยไปหามาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

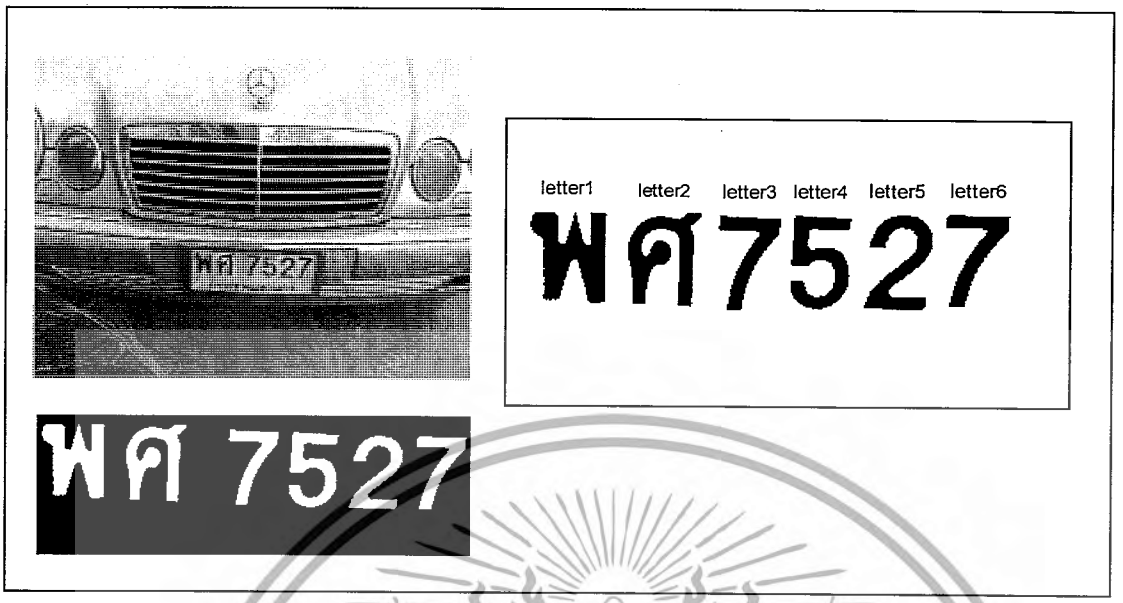
12. นำค่า Column ที่เป็นขอบซ้ายและขอบขวาของตัวเลขและพยัญชนะ แต่ละตัวที่เรียงดับแล้วเป็นแกนอ้างอิงในการคัดลอกภาพตัวเลขและพยัญชนะออกมาจากภาพ Inverse Binary ของภาพที่ 4.10 จากขั้นตอนที่ 9 ดังแสดงในภาพที่ 4.12 จะได้ผลดังภาพที่ 4.13

ภาพที่ 4.13 ภาพ Inverse Binary ของภาพที่ 4.9

ภาพที่ 4.14 ภาพตัวเลขและพยัญชนะที่แยกออกมาได้จากภาพที่ 4.12 ในขั้นตอนที่ 12

ตัวอย่างผลการทดลองที่ได้ในหัวข้อที่ 4.1 นี้ แต่ละภาพจะแสดงภาพผลการทดลองที่ได้จากขั้นตอนที่ 7 และ 12 จากตัวอย่างผลการทดลองภาพที่ 4.14 จะเห็นได้ว่าภาพที่โปรแกรมสามารถทำการ Label แยกตัวเลขและพยัญชนะออกจากแผ่นป้ายทะเบียนได้ครบทุกตัวหลังจากที่โปรแกรมสามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนได้แล้วนั้น ส่วนต่าง ๆ ที่อยู่บนแผ่นป้ายทะเบียนที่ไม่ใช่ตัวเลขและพยัญชนะแถวบนของแผ่นป้ายทะเบียน อย่างเช่นส่วนที่เป็นชื่อจังหวัดลวดลายต่าง ๆ ที่อยู่บนแผ่นป้ายทะเบียน จะต้องถูกกำจัดออกไปให้หมด และระยะห่างระหว่างตัวเลขและพยัญชนะแต่ละตัวที่อยู่บนแผ่นป้ายทะเบียนจะต้องไม่มีน็อตหรือตะปูแทรกอยู่ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก)

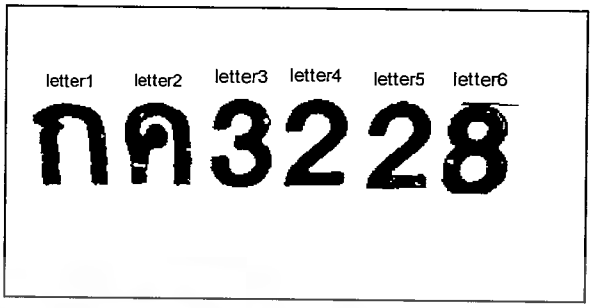
(ข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

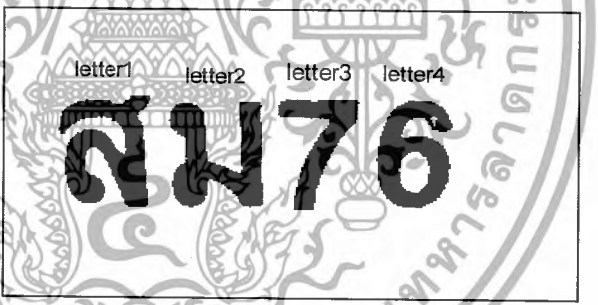
(ค)

(ง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

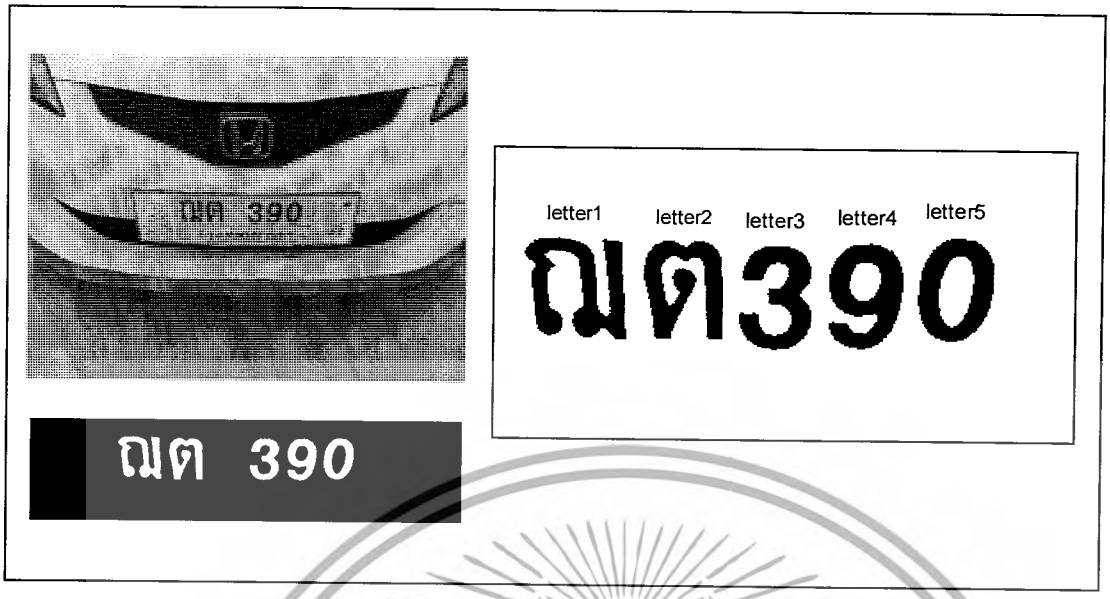


(จ)



(ฉ)

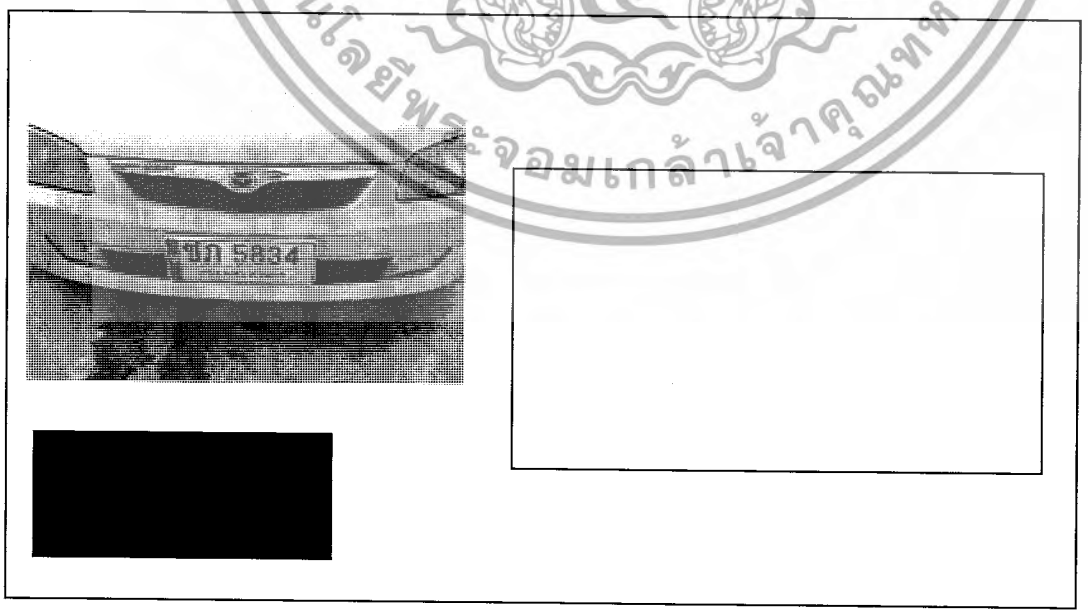
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ช)

ภาพที่ 4.15 (ก -ช) ตัวอย่างภาพผลการทดลองที่โปรแกรมสามารถแยกตัวเลขและพยัญชนะออกจากแผ่นป้ายได้ครบทุกตัว

การทดลองในหัวข้อ 4.1 นี้ จากผลการทดลองพบว่ามีบางภาพที่โปรแกรมไม่สามารถทำการแยกตัวเลขและพยัญชนะออกมาจากภาพแผ่นป้ายทะเบียนได้ ดังแสดงในภาพที่ 4.15



(ก) โปรแกรมไม่สามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนได้

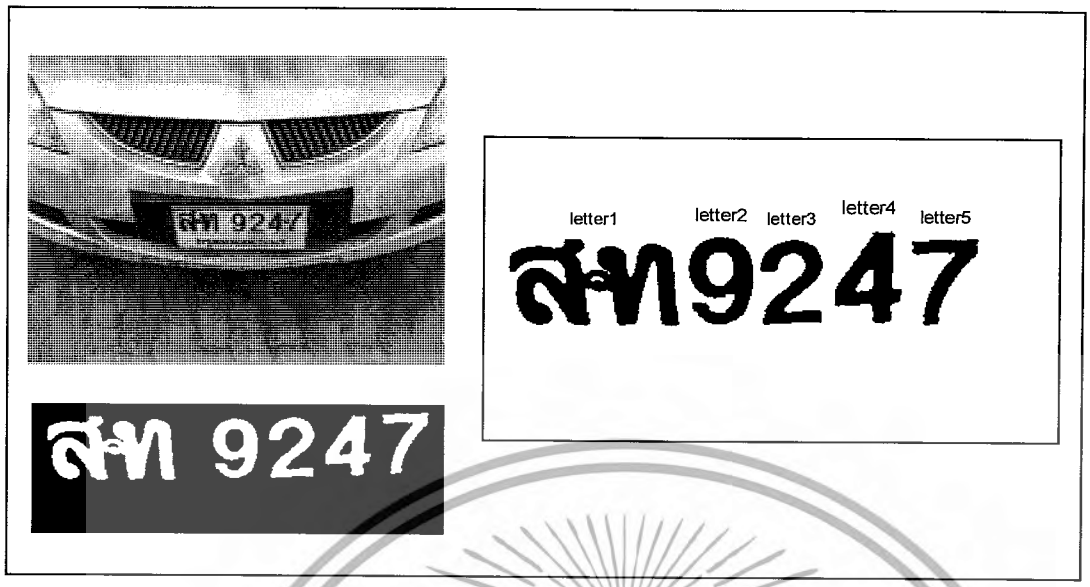
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 4.15 (ก) โปรแกรมไม่สามารถทำการแยกตัวเลขและพยัญชนะออกมาจากภาพแผ่นป้ายทะเบียนได้ เพราะว่าโปรแกรมไม่สามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนได้ เป็นผลเนื่องมาจากในการแปลงภาพ RGB ให้เป็นภาพ Binary โดยใช้ทั้งค่าเทรชโฮลอัตโนมัติในโปรแกรม MATLAB และทั้งค่าเทรชโฮลที่เกิดจากการนำค่าความเข้มสีของทุกพิกเซลภาพ RGB มาเฉลี่ยกัน แล้วบวกด้วยค่าคงที่ค่าหนึ่งแล้วทำให้ส่วนที่เป็นแผ่นป้ายทะเบียนถูกลบออกไป ดังแสดงในภาพที่ 4.15 (ข-ค)

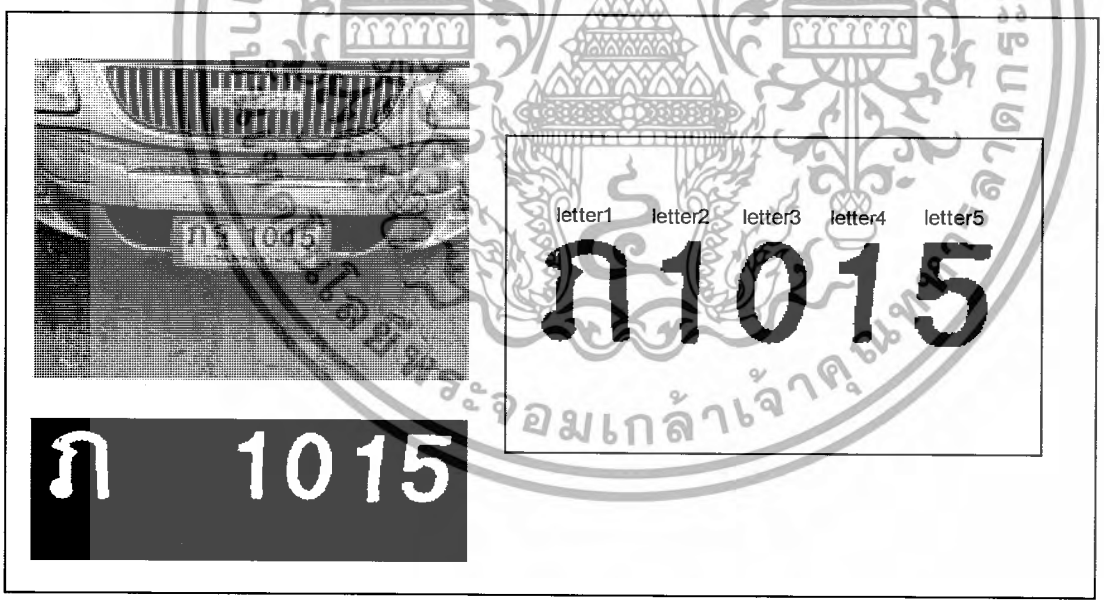
(ข) (ค)

ภาพที่ 4.16 ภาพ Binary ที่โปรแกรมไม่สามารถค้นหาแผ่นป้ายทะเบียนได้ (ข) ภาพ Binary ที่เกิดจากค่าเทรชโฮลที่คำนวณจากค่าเฉลี่ยของความเข้มสีทุกพิกเซลและบวกด้วยค่าคงที่ค่าหนึ่งของภาพ RGB จากภาพที่ 4.15 (ก) (ค) ภาพ Binary ของภาพ RGB จากภาพที่ 4.15 (ก) ที่เกิดจากค่าเทรชโฮลอัตโนมัติในโปรแกรม MATLAB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ง) ระยะห่างระหว่างตัวเลขและตัวหนังสือบนแผ่นป้ายทะเบียน มีน็อตแทรกอยู่ตรงกลางและเชื่อมตัวเลขหรือพยัญชนะ 2 ตัวให้ติดกันซึ่งเป็นผลทำให้โปรแกรมประมวลผลเป็นตัวเลขหรือพยัญชนะตัวเดียว



(จ) พยัญชนะบนแผ่นป้ายทะเบียน (ตัว ฐ) ขาดออกจากกัน ทำให้แต่ละส่วนที่ขาดช่วงไปถูกโปรแกรมประมวลผลให้เป็นวัตถุละกลุ่มก้อนกัน

ภาพที่ 4.16 (ก – จ) ตัวอย่างภาพผลการทดลองที่โปรแกรมไม่สามารถแยกตัวเลขและพยัญชนะออกจากแผ่นป้ายได้ครบทุกตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพตัวอย่างผลการทดลองที่ได้ในตอนที่ 4.1 นี้ จะเห็นได้ว่าภาพที่โปรแกรมไม่สามารถแยกตัวเลขและพยัญชนะออกจากแผ่นป้ายทะเบียนได้ครบทุกตัว ซึ่งมีสาเหตุมาจาก

1. โปรแกรมไม่สามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนได้ เพราะว่าเมื่อแปลงภาพจาก RGB ให้เป็นภาพ Binary โดยใช้ทั้งค่าเทรชโฮลอัตโนมัติในโปรแกรม MATLAB และทั้งค่าเทรช-โฮลที่เกิดจากการนำค่าความเข้มสีของทุกพิกเซลภาพ RGB มาเฉลี่ยกันแล้วบวกด้วยค่าคงที่ค่าหนึ่ง แล้วทำให้ส่วนที่เป็นแผ่นป้ายทะเบียนถูกลบออกไป ซึ่งในกรณีนี้มักจะเกิดขึ้นกับรถสีขาวและขอบป้ายทะเบียนไม่มีความชัดเจนพอ

2. ระยะห่างระหว่างตัวเลขและพยัญชนะที่อยู่บนภาพแผ่นป้ายทะเบียนมีน็อตแทรกอยู่ตรงกลางและเชื่อมตัวเลขหรือพยัญชนะ 2 ตัวให้ติดกันซึ่งเป็นผลทำให้โปรแกรมประมวลผลเป็นตัวเลขหรือพยัญชนะตัวเดียว

3. ภาพตัวเลขหรือพยัญชนะในภาพแผ่นป้ายทะเบียนขาดช่วงกัน ซึ่งในกรณีนี้จะทำให้แต่ละส่วนที่ขาดช่วงไป ถูกโปรแกรมประมวลผลให้เป็นวัตถุที่ไม่ใช่กลุ่มก้อนเดียวกัน (ในขั้นตอนของการ Label เพื่อกำจัดกลุ่มพิกเซลสีขาวที่มีความสูงนอกเหนือจากค่าที่กำหนด) ซึ่งเมื่อกลายเป็นวัตถุละกลุ่มก้อนกัน ก็มักจะมีความสูงไม่มากพอที่โปรแกรมจะประมวลผลให้เป็นภาพตัวเลขและพยัญชนะแถวบนได้ แต่กลับประมวลผลเป็นสิ่งรบกวนอื่น ๆ ที่อยู่บนแผ่นป้ายแทน ซึ่งทำให้ส่วนของตัวเลขหรือพยัญชนะที่ขาดช่วงกันไปนี้ถูกลบออกไป

ผลการทดลองที่ได้ในตอนนี้สามารถสรุปได้ดังตารางที่ 4.1

ตารางที่ 4.1 สรุปผลการทดลองที่ได้จากตอนที่ 4.1

จำนวนภาพที่ใช้ทำการทดลอง	จำนวนภาพที่โปรแกรมสามารถแยกตัวเลขและพยัญชนะออกมาจากภาพแผ่นป้ายทะเบียนได้ครบทุกตัว	เปอร์เซนต์ความถูกต้อง
60	50	83%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การรู้จำตัวเลขและพยัญชนะ

การทดลองในส่วนนี้จะนำภาพทั้งหมด 50 ภาพที่โปรแกรมสามารถทำการแยกตัวเลขและพยัญชนะออกมาจากแผ่นป้ายได้ครบทุกตัวจากการทดลองในหัวข้อ 4.1 เมื่อโปรแกรมสามารถแยกตัวเลขและพยัญชนะออกมาจากภาพแผ่นป้ายทะเบียนได้แล้ว ให้นำตัวเลขและพยัญชนะเหล่านี้มาทำการรู้จำ โดยการนำไปวางทาบกับภาพตัวเลขและพยัญชนะต้นแบบเพื่อหาค่าสัมประสิทธิ์ความเหมือน แล้วให้โปรแกรมแสดงผลการรู้จำทาง Command Window ของโปรแกรม MATLAB ซึ่งมีตัวอย่างภาพที่ได้จากการผลการทดลองดังแสดงในภาพที่ 4.16 โดยในแต่ละภาพจะแสดงภาพที่ได้จากการทดลองในขั้นตอนที่ 7 ของการทดลองตอนที่ 4.1 และผลจากการนำตัวเลขและพยัญชนะทั้งหมดที่สามารถแยกออกมาจากแผ่นป้ายทะเบียนได้ไปประมวลผลในการรู้จำ

(ก)

ภาพที่ 4.17 (ก) ตัวอย่างภาพผลการทดลองที่โปรแกรมสามารถแยกตัวเลขและพยัญชนะออกจากแผ่นป้ายได้ครบทุกตัว

จากภาพที่ 4.16 (ก) เมื่อนำภาพตัวเลขและพยัญชนะทั้งหมดที่สามารถแยกออกมาได้จากภาพแผ่นป้ายทะเบียนไปเข้าสู่ขั้นตอนการรู้จำทีละภาพ โดยให้เรียงลำดับเหมือนบนภาพแผ่นป้ายทะเบียนแล้ว จะได้ค่าสัมประสิทธิ์ความเหมือนที่ได้จากการนำภาพตัวเลขและพยัญชนะเหล่านี้ไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วางทาบกับภาพตัวเลขและพยัญชนะบนภาพชุด Template ทั้งหมด 97 ภาพ จะได้ผลจากการรู้จำดังแสดงในภาพที่ 4.16 (ข-ฉ)

CORR =

Columns 1 through 12

0.3786 0.3878 0.5624 0.2385 0.1445 0.2257 0.2416 0.2167 0.0966 -0.0019

0.0015 0.2246

Columns 13 through 24

0.3016 0.6188 0.2249 0.2363 0.5370 0.1856 0.2241 0.0782 0.3653 -0.1397

0.2675 0.3774

Columns 25 through 36

0.2511 0.4853 0.4184 0.1633 0.4980 0.0984 0.1998 0.4012 0.4075 -0.0658

0.3753 0.5201

Columns 37 through 48

0.1763 0.3095 -0.0210 0.2797 0.6065 0.5149 0.5209 0.1939 0.1366 0.3301

0.3574 0.1417

Columns 49 through 60

0.4050 0.1166 0.1909 0.2974 0.1575 0.1605 0.1974 0.4928 -0.0338

Columns 61 through 72

0.0215 0.5252 0.5783 0.3760 0.3590 0.3064 0.3863 **0.6534** 0.2335 0.1921

0.1222 0.3281

Columns 73 through 84

0.2897 0.0989 -0.0183 0.2883 0 0.1023 0.0568 0.0707 -0.0455 -0.0864

0.0100 -0.2059

Columns 85 through 96

-0.0520 0.0675 0.0407 0.2333 0.0720 0.3270 0.1771 0.0166 0.0722 0.1826

0.3430 -0.0435

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Column 97

0.1498

CORRELATION =

0.6534 %----ค่าสัมประสิทธิ์ความเหมือนที่มีค่ามากที่สุดที่ได้จากเมตริกซ์ CORR

W1 = %----ค่าสัมประสิทธิ์ความเหมือนที่มากที่สุดอยู่ที่ Column ที่ 68 ของเมตริกซ์ CORR ที่

%----เก็บค่าสัมประสิทธิ์ความเหมือนไว้

68

ฌ %----ผลที่ได้จากการรู้จำภาพพยัญชนะ ฌ ก็คือตัว ฌ

ภาพที่ 4.17 (ข) ค่าสัมประสิทธิ์ความเหมือนที่ได้จากการนำภาพพยัญชนะ ฌ ที่แยกมาได้จากแผ่นป้ายทะเบียนไปวางทาบกับภาพตัวเลขและพยัญชนะทุกตัวจากภาพต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CORR2 =

Columns 1 through 12

0.0651 0.5147 0.4096 0.0082 0.2074 0.1759 0.1034 0.6129 -0.0663 0.0340 0.0217 0.0744

Columns 13 through 24

0.0532 0.2786 0.3057 0.2420 0.0520 0.1419 -0.0485 0.1118 0.0621 0.0993 0.3352 0.0325

Columns 25 through 36

0.3908 0.0730 0.2125 0.4002 0.3038 0.2020 -0.0657 0.1998 0.2874 0.0660 0.1536 0.0156

Columns 37 through 48

0.0524 -0.0250 -0.0441 0.1330 0.4845 0.1835 0.1036 0.3952 0.1324 0.5393 0.0473 -0.1279

Columns 49 through 60

0.2271 -0.1788 -0.0032 0.1818 0.1969 -0.0438 -0.0072 0.0604 -0.0641 -0.1753 0.2068 -0.0646

Columns 61 through 72

0.1390 0.0071 0.0584 0.2847 0.0128 0.0342 0.1888 0.0886 0.4363 0.2237 0.3528 0.1875

Columns 73 through 84

0.2015 **0.8084** 0.2241 0.0095 0 0.1207 0.2545 -0.0325 0.0300 0.0448 -0.0739 -0.0975

Columns 85 through 96

-0.0069 0.1868 0.2051 0.1473 0.2113 0.1114 0.2433 0.2794 0.1589 0.0517 0.0330 0.2594

Column 97

0.1479

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CORRELATION2 =
0.8084 %----ค่าสัมประสิทธิ์ความเหมือนที่มีค่ามากที่สุดที่ได้จากเมตริกซ์ CORR2
W =
 74 %----ค่าสัมประสิทธิ์ความเหมือนที่มากที่สุดอยู่ที่ Column ที่ 74 ของเมตริกซ์ CORR2 ที่
%----เก็บค่าสัมประสิทธิ์ความเหมือนไว้
ต %----ผลที่ได้จากการรู้จำภาพพยัญชนะ ต ก็คือตัว ต
```

ภาพที่ 4.17 (ค) ค่าสัมประสิทธิ์ความเหมือนที่ได้จากการนำภาพพยัญชนะ ต ที่แยกมาได้จากแผ่นป้ายทะเบียนไปวางทาบกับภาพตัวเลขและพยัญชนะทุกตัวจากภาพต้นแบบ

```
CORR3 =
Columns 1 through 12
0.2906   0.4373   0.2108   0.4241   0.4142   0.7653   0.5694   0.4719   0.5088   -0.0937
-0.1756   0.2237
Columns 13 through 23
0.2590   0.4052   0.2780   0.4510   0.0350   0.0944   0.2869   0.5759   0.5464   0.1230
0.3685
CORRELATION3 =
0.7653 %----ค่าสัมประสิทธิ์ความเหมือนที่มีค่ามากที่สุดที่ได้จากเมตริกซ์ CORR3
W =
6 %----ค่าสัมประสิทธิ์ความเหมือนที่มากที่สุดอยู่ที่ Column ที่ 6 ของเมตริกซ์ CORR3 ที่
%----เก็บค่าสัมประสิทธิ์ความเหมือนไว้
3 %----ผลที่ได้จากการรู้จำภาพเลข 3
```

ภาพที่ 4.17 (ง) ค่าสัมประสิทธิ์ความเหมือนที่ได้จากการนำภาพเลข 3 ที่แยกมาได้จากแผ่นป้ายทะเบียนไปวางทาบกับภาพตัวเลขและพยัญชนะทุกตัวจากภาพต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CORR4 =
Columns 1 through 12
0.3484  0.5885  0.0902  0.3595  0.2415  0.4048  0.3163  0.2233  0.3389  -0.1035
-0.1735  0.2435
Columns 13 through 23
0.2792  0.3839  0.3375  0.2993  -0.1221  -0.0858  0.2747  0.3577  0.2238  0.5219
0.8156
CORRELATION4 =
0.8156 %----ค่าสัมประสิทธิ์ความเหมือนที่มีค่ามากที่สุดที่ได้จากเมตริกซ์ CORR4
W =
23 %----ค่าสัมประสิทธิ์ความเหมือนที่มากที่สุดอยู่ที่ Column ที่ 23 ของเมตริกซ์ CORR4 ที่
%----เก็บค่าสัมประสิทธิ์ความเหมือนไว้
9 %----ผลที่ได้จากการรู้จำภาพเลข 9
```

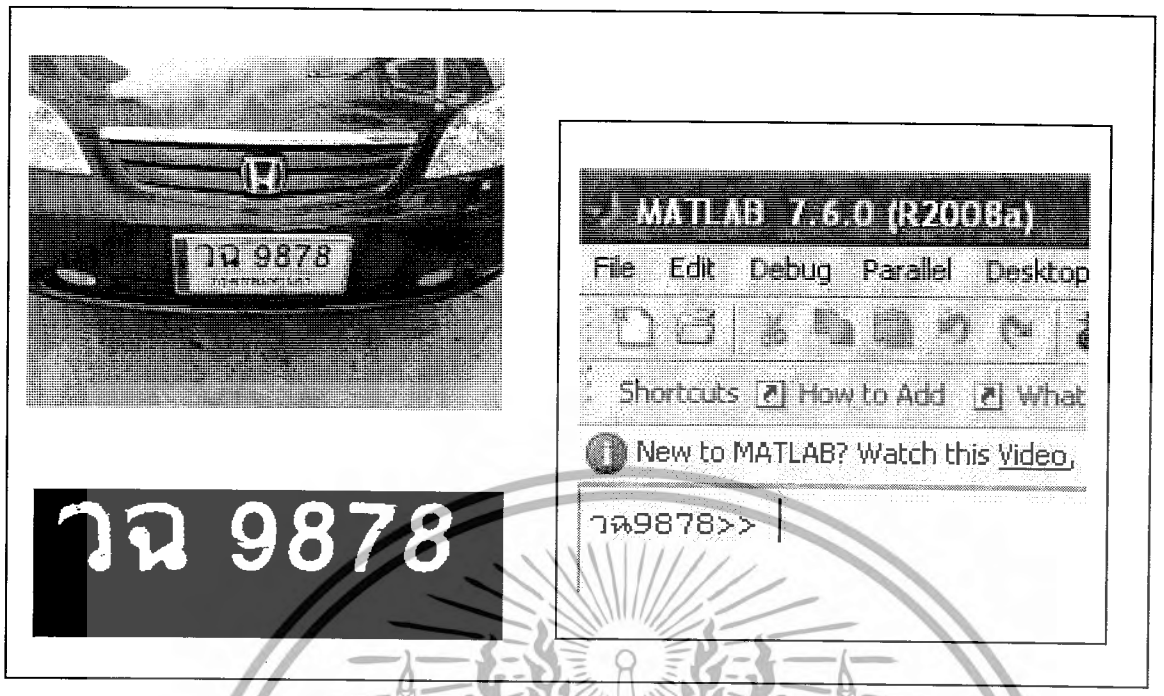
ภาพที่ 4.17 (จ) ค่าสัมประสิทธิ์ความเหมือนที่ได้จากการนำภาพเลข 9 ที่แยกมาได้จากแผ่นป้ายทะเบียนไปวางทาบกับภาพตัวเลขและพยัญชนะทุกตัวจากภาพต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

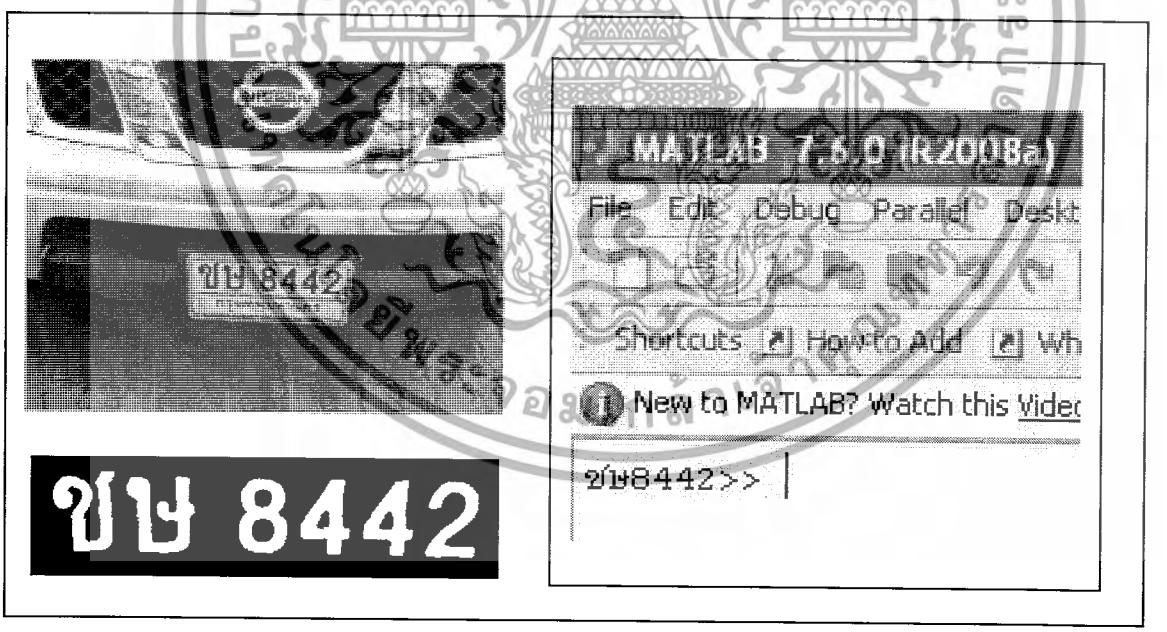
```
CORR5 =
Columns 1 through 12
0.2510  0.8154  0.0636  0.1539  0.0146  0.3632  0.1365  0.0495  0.0749  -0.0368
0.0588  0.1645
Columns 13 through 23
0.2067  0.5349  0.3584  0.5349  -0.1440  -0.1871  0.2139  0.3912  0.4656  0.0176
0.4022
CORRELATION5 =
0.8154 %----ค่าสัมประสิทธิ์ความเหมือนที่มีค่ามากที่สุดที่ได้จากเมตริกซ์ CORR5
W =
2 %----ค่าสัมประสิทธิ์ความเหมือนที่มากที่สุดอยู่ที่ Column ที่ 2 ของเมตริกซ์ CORR5 ที่
%----เก็บค่าสัมประสิทธิ์ความเหมือนไว้
0 %----ผลที่ได้จากการรู้จำภาพตัวเลข 0
```

ภาพที่ 4.17 (ฉ) ค่าสัมประสิทธิ์ความเหมือนที่ได้จากการนำภาพเลข 0 ที่แยกมาได้จากแผ่นป้ายทะเบียน ไปวางทาบกับภาพตัวเลขและพยัญชนะทุกตัวจากภาพต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ช)

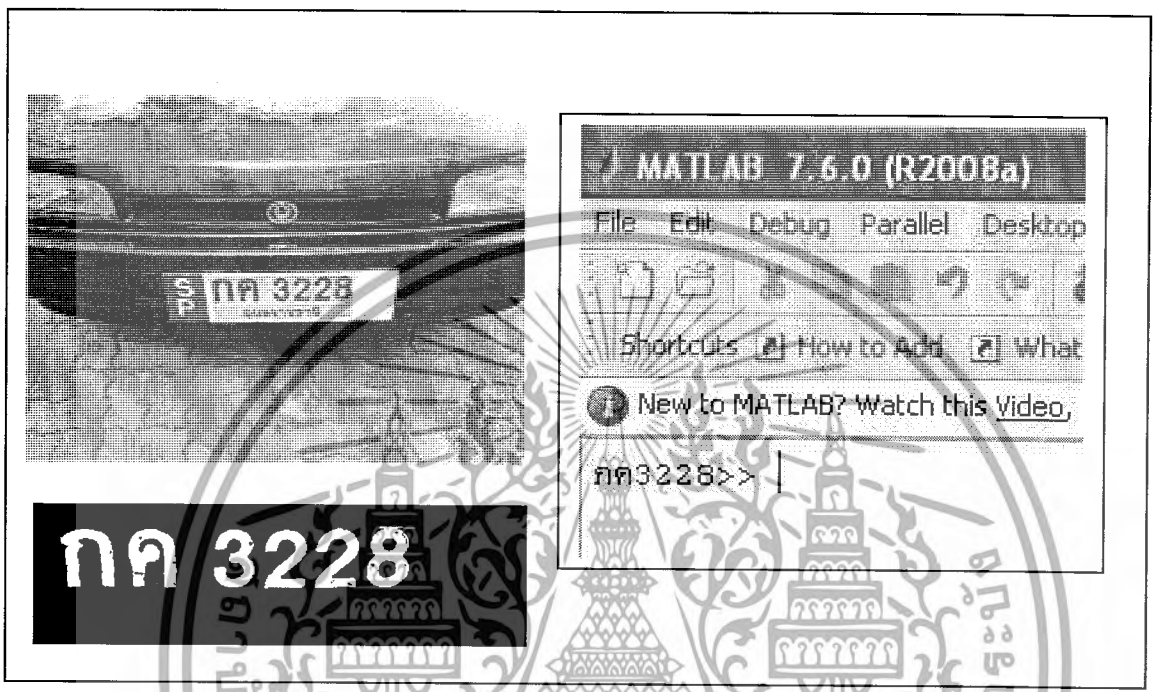


(ญ)

ภาพที่ 4.17 (ก-ญ) ตัวอย่างผลการทดลองที่โปรแกรมทั้งสองส่วนสามารถทำการรู้จำตัวเลขและพยัญชนะที่อยู่บนแผ่นป้ายทะเบียนได้ถูกต้องทุกตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองในตอนนี้ จากผลการทดลองพบว่าบางแผ่นป้ายทะเบียน โปรแกรมไม่สามารถทำการรู้จำตัวเลขและพยัญชนะที่อยู่บนแผ่นป้ายทะเบียนได้ถูกต้องครบถูกตัว ดังแสดงในตัวอย่างภาพที่ 4.18



(ก)

จากภาพที่ 4.18 (ก) จะเห็นได้ว่าภาพพยัญชนะ ก ที่อยู่บนแผ่นป้ายทะเบียนเอียง ทำให้โปรแกรมประมวลผลในส่วนของการรู้จำตัวเลขและพยัญชนะผิดพลาดได้ ซึ่งจากภาพที่ 4.18 (ข) แสดงให้เห็นค่าสัมประสิทธิ์ความเหมือนที่ได้จากการนำภาพพยัญชนะ ก นี้ไปวางทาบกับตัวเลขและพยัญชนะแม่แบบทุกตัว ค่าสัมประสิทธิ์ความเหมือนที่มากที่สุดอยู่ที่ Column ที่ 44 ของเมตริกซ์ CORR ทำให้โปรแกรมประมวลผลตัว ก เป็นตัว ภ แทนที่ 4.18 (ก) ไปวางทาบกับตัวเลขและตัวอักษรแม่แบบทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CORR =

Columns 1 through 12

0.0571 0.5887 0.3888 0.1476 0.0632 0.0875 0.0813 0.4136 0.1139 0.2214
0.1974 -0.0073

Columns 13 through 24

0.1884 0.3325 0.0819 -0.0568 0.2870 0.4459 0.2404 0.3416 0.1138 0.0264
0.0915 0.1995

Columns 25 through 36

0.5507 0.2275 0.3214 0.3980 0.3419 0.3894 0.0548 0.2572 0.3210 0.0081
0.0246 0.2802

Columns 37 through 48

0.3433 -0.0012 0.0283 0.0387 0.5253 0.1521 0.3937 **0.7114** 0.0832 0.1983
0.2097 0.1304

Columns 49 through 60

0.1891 -0.2118 0.0545 0.2081 0.2671 -0.1430 -0.0801 0.2733 -0.1884 -0.1110
0.1636 -0.0094

Columns 61 through 72

0.2313 0.2162 0.1719 0.3745 0.1498 0.2585 0.2789 0.1597 0.4688 0.4724
0.4288 0.1772

Columns 73 through 84

0.0047 0.3722 -0.1138 0.1077 0 0.1789 0.0879 -0.0482 -0.0674 -0.0926 -
0.1045 -0.1158

Columns 85 through 96

-0.0156 0.0384 0.0802 0.2024 0.0383 0.1863 0.3644 0.1860 0.0008 0.0916
0.1919 -0.0681

Column 97

0.0789

CORRELATION =

0.7114

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
W1 =
 44
ภ
```

(ข) ค่าสัมประสิทธิ์ความเหมือนที่ได้จากการนำภาพพยัญชนะ ก ที่แยกออกมาได้จากแผ่นป้ายทะเบียนไปวางทาบกับภาพตัวเลขและพยัญชนะทุกตัวของภาพต้นแบบ

(ค)

จากภาพที่ 4.18 (ค) จะเห็นได้ว่าโปรแกรมไม่แสดงผลที่ได้จากการรู้จำของภาพเลข 9 ที่อยู่บนแผ่นป้ายทะเบียน เพราะว่าภาพเลข 9 เอียงไปทางด้านหน้าหรือด้านหลังมากเกินไป ทำให้ค่าสัมประสิทธิ์ความเหมือนที่ได้จากการทำ Template Matching มีค่าต่ำกว่า 0.6 ซึ่งถ้าค่าสัมประสิทธิ์ความเหมือนต่ำกว่า 0.6 โปรแกรมจะไม่ทำการแสดงผลที่ได้จากการรู้จำภาพเลข 9 ทาง Command Window ในโปรแกรม ถึงแม้ว่าค่าสัมประสิทธิ์ความเหมือนที่มากที่สุดจะเกิดจากการนำเลข 9 ไปวางทาบกับตัวเลข 9 ที่เป็นแม่แบบก็ตาม ดังแสดงในภาพที่ 4.18 (ง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CORR6 =

Columns 1 through 12

0.0831 0.4804 0.2840 0.2420 0.0876 0.3423 0.1563 0.1283 0.1754 -0.1263

-0.0586 0.2018

Columns 13 through 23

0.2363 0.2996 0.1942 0.3247 -0.0803 -0.1245 0.1939 0.2908 0.4273 0.1971

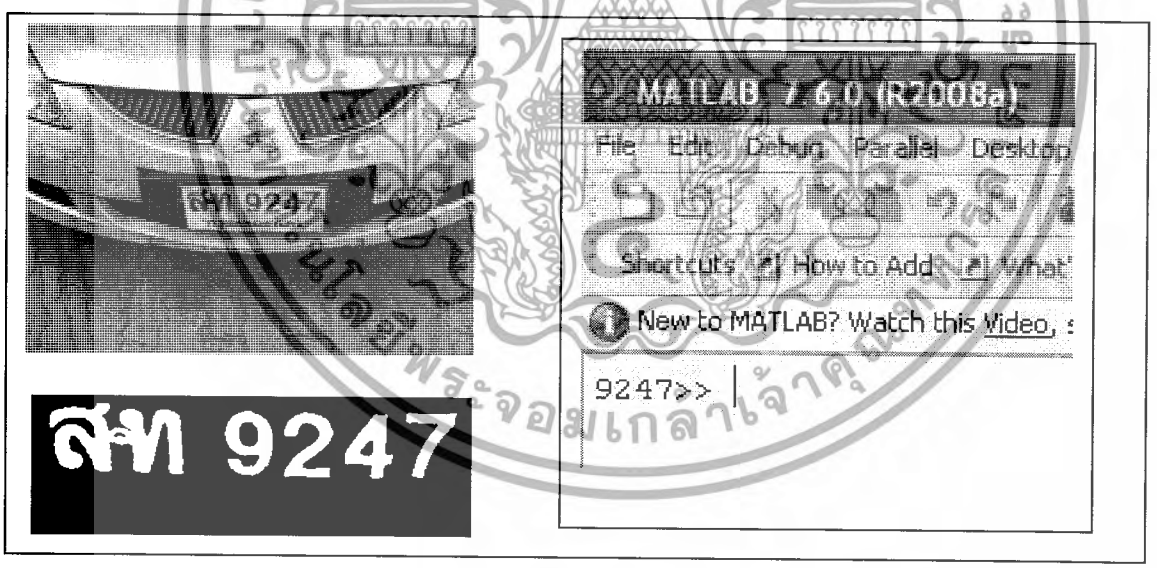
0.5058

CORRELATION6 =

0.5058

(ง)

ค่าสัมประสิทธิ์ความเหมือนที่ได้จากการนำภาพเลข 9 ที่อยู่บนแผ่นป้ายทะเบียน จาก ภาพที่ 4.18 (ค) ไปวางทาบกับตัวเลขแม่แบบทั้งหมด



(จ)

ภาพที่ 4.18 (ก-จ) ตัวอย่างภาพผลการทดลองที่โปรแกรมไม่สามารถทำการรู้จำตัวเลขและพยัญชนะที่อยู่บนแผ่นป้ายทะเบียนได้ถูกต้องทุกตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 4.18 (จ) จะเห็นได้ว่าภาพตัวพยัญชนะ ส และ ท กลายเป็นพื้นที่เดียวกันเพราะตะปูที่ติดอยู่ ทำให้โปรแกรมประมวลผลในขั้นตอนการ Label แยกภาพพยัญชนะและตัวเลขออกมาจากภาพแผ่นป้ายทะเบียนเป็นภาพพยัญชนะตัวเดียวกัน ซึ่งถ้านำภาพนี้ไปวางทาบกับภาพตัวเลขและพยัญชนะที่เป็นแม่แบบเพื่อหาค่าสัมประสิทธิ์ความเหมือน ก็จะทำให้ได้ค่าสัมประสิทธิ์ความเหมือนที่ต่ำกว่า 0.6 โปรแกรมจึงไม่แสดงผลการรู้จำภาพพยัญชนะทาง Command Window

จากการทดลองในหัวข้อที่ 4.2 พบว่าบางแผ่นป้ายทะเบียน โปรแกรมไม่สามารถทำการรู้จำตัวเลขและตัวพยัญชนะได้ถูกต้องครบทุกตัว ซึ่งมีสาเหตุมาจาก

1. โปรแกรมที่ออกแบบ ไม่สามารถแยกภาพตัวเลขและพยัญชนะออกมาจากภาพแผ่นป้ายทะเบียนได้ครบทุกตัว ซึ่งเป็นผลมาจากการทดลองในหัวข้อที่ 4.1

2. ถ้าภาพตัวเลขและพยัญชนะที่แยกออกมาจากภาพแผ่นป้ายทะเบียนเอียงไปทางด้านหน้าหรือด้านหลังมากเกินไป ทำให้ค่าสัมประสิทธิ์ความเหมือนที่ได้จากการทำ Template Matching มีค่าต่ำกว่า 0.6 ซึ่งถ้าค่าสัมประสิทธิ์ความเหมือนต่ำกว่า 0.6 โปรแกรมจะไม่ทำการแสดงผลตัวเลขหรือพยัญชนะนั้นทาง Command Window ในโปรแกรม MATLAB

3. พยัญชนะบางตัวมีลักษณะใกล้เคียงกัน อย่างเช่น ก กับ ภ และ ถ, บ กับ ป, พ กับ ฟ เป็นต้น ซึ่งอาจจะทำให้โปรแกรมประมวลผลในส่วนของการรู้จำตัวเลขและพยัญชนะผิดพลาดได้

ผลการทดลองที่ได้ในตอนนี้สามารถสรุปได้ดังตารางที่ 4.2

ตารางที่ 4.2 สรุปผลการทดลองที่ได้จากตอนที่ 4.2

จำนวนภาพที่ใช้ทำการทดลอง	จำนวนภาพที่โปรแกรมสามารถทำการรู้จำแผ่นป้ายทะเบียนได้ถูกต้อง	เปอร์เซนต์ความถูกต้อง
50	33	66%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปและวิจารณ์ผลการทดลอง

5.1 สรุปผลการทดลอง

จากผลการทดลองพบว่า โปรแกรมที่ออกแบบสามารถค้นหาที่เป็นแผ่นป้ายทะเบียนจากภาพถ่ายส่วนด้านหน้ารถยนต์ ซึ่งเป็นภาพแบบ RGB และแยกตัวเลขและพยัญชนะทั้งหมด ที่อยู่บนแผ่นป้ายออกมาเพื่อทำการรู้จำโดยใช้เทคนิค Template Matching ได้ ซึ่งในการทำการทดลองได้ใช้ภาพตัวอย่างทั้งหมด 60 ภาพ โปรแกรมที่ออกแบบสามารถค้นหาแผ่นป้ายทะเบียนและแยกตัวเลขและพยัญชนะทั้งหมดที่อยู่บนแผ่นป้ายได้เป็นจำนวน 50 ภาพ คิดเป็นเปอร์เซนต์ความถูกต้อง 50% และเมื่อนำภาพที่ถูกต้องจากส่วนที่ 1 ทั้ง 50 ภาพนี้ ไปทดลองในส่วนการรู้จำ โปรแกรมสามารถทำการรู้จำตัวเลขและพยัญชนะที่อยู่บนแผ่นป้ายได้ถูกต้องครบทุกตัวเป็นจำนวน 33 ภาพ คิดเป็นเปอร์เซนต์ความถูกต้องได้ 66%

การที่โปรแกรมจะสามารถค้นหาแผ่นทะเบียนจากภาพ RGB ได้นั้น ในการแปลงภาพ RGB ให้เป็นภาพ Binary ส่วนที่เป็นแผ่นป้ายทะเบียนในภาพ Binary จะต้องมีขอบที่ชัดเจน อย่างเช่นส่วนของแผ่นป้ายทะเบียนจะต้องไม่มีพิกเซลสีขาวติดกับขอบภาพ เป็นต้น เพราะว่าจะทำให้ส่วนที่เป็นแผ่นป้ายทะเบียนถูกกำจัดออกไปด้วยคำสั่ง imclearborder ซึ่งทำให้โปรแกรมไม่สามารถค้นหาแผ่นป้ายทะเบียนได้

เมื่อโปรแกรมสามารถค้นหาแผ่นป้ายทะเบียนรถยนต์ได้แล้ว โปรแกรมก็จะทำการแยกตัวเลขและพยัญชนะทั้งหมดที่อยู่บนแผ่นป้ายออกมาทีละตัว โดยเรียงลำดับให้เหมือนบนแผ่นป้าย เพื่อนำไปประมวลผลในส่วนของการรู้จำแผ่นป้ายทะเบียน ในการที่โปรแกรมจะสามารถแยกตัวเลขและพยัญชนะออกมาจากแผ่นป้ายได้ครบทุกตัวนั้น ส่วนอื่น ๆ ที่อยู่บนแผ่นป้ายทะเบียนที่ไม่ใช่ตัวเลขและพยัญชนะแถวบนของแผ่นป้าย อย่างเช่นลวดลายของแผ่นป้าย จะต้องถูกกำจัดออกไปให้หมด และระยะห่างระหว่างตัวเลขและพยัญชนะ จะต้องไม่มีน็อตหรือตะปูคั่นตรงกลาง เพราะน็อตหรือตะปูนี้จะเชื่อมตัวเลขหรือพยัญชนะให้มีส่วนติดกัน ซึ่งในการ Label เพื่อแยกตัวเลขและพยัญชนะออกมาจากแผ่นป้ายทะเบียน โปรแกรมก็จะประมวลผลให้ตัวเลขหรือพยัญชนะที่ติดกัน 2 ตัวนั้นเป็นตัวเลขหรือพยัญชนะเพียงตัวเดียว

ในการรู้จำแผ่นป้ายทะเบียน จากการทดลองพบว่าแผ่นป้ายทะเบียนที่โปรแกรมจะสามารถทำการรู้จำตัวเลขและพยัญชนะทั้งหมดที่อยู่บนแผ่นป้ายได้ถูกต้องทุกตัวนั้น ถ้าโปรแกรมสามารถแยกตัวเลขและพยัญชนะออกมาจากแผ่นป้ายทะเบียนได้ครบทุกตัว(จากการทดลองในหัวข้อ 4.1) โปรแกรมก็จะมาสารถนำตัวเลขและพยัญชนะเหล่านี้ ไปเข้าสู่ขั้นตอนการรู้จำแผ่นป้ายทะเบียนได้ครบทุกตัวเช่นกัน แต่ตัวเลขและพยัญชนะที่อยู่บนแผ่นป้ายทะเบียน จะต้องไม่เอียงจนเกินไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และมีความชัดเจนพอ เพราะเมื่อตัวเลขและพยัญชนะเหล่านั้นไปวางทาบหาค่าความเหมือนกับตัวเลขและพยัญชนะต้นแบบจะทำให้ได้ค่าสัมประสิทธิ์ความเหมือนที่ต่ำกว่า 0.6 ทำให้โปรแกรมไม่แสดงผลการรู้จำตัวเลขหรือพยัญชนะนั้นทาง Command Window หรืออาจจะทำให้การประมวลผลผิดพลาด อย่างเช่น ตัว ก ที่แยกออกมาจากแผ่นป้ายทะเบียนได้นั้นเอียงไปทางด้านหน้า เมื่อนำไปวางทาบกับตัวเลขและพยัญชนะทุกตัวที่เป็นแม่แบบ อาจจะมีค่าสัมประสิทธิ์ความเหมือนเมื่อนำไปวางทาบกับตัว ถ มากที่สุด ซึ่งโปรแกรมก็จะประมวลผลพยัญชนะ ก นั้นให้เป็นพยัญชนะ ถ แทน

5.2 ข้อเสนอแนะ

ความผิดพลาดโดยส่วนมากเกิดขึ้นกับการทดลองในส่วนของการรู้จำตัวเลขและพยัญชนะโดยใช้เทคนิค Template Matching ซึ่งจุดอ่อนนี้มีวิธีการแก้ไขดังนี้

1. จากโปรแกรมที่ออกแบบ ในการนำภาพทั้งสองมาวางทาบกันเพื่อหาค่าสัมประสิทธิ์ความเหมือนนั้น โปรแกรมสามารถปรับขนาดภาพตัวเลข และพยัญชนะที่แยกออกมาได้จากแผ่นป้ายทะเบียนได้เฉพาะความสูงของภาพเท่านั้น คือ ปรับให้มีความสูงเป็น 100 พิกเซล เท่ากับความสูงของภาพตัวเลขและพยัญชนะต้นแบบ แต่ไม่สามารถปรับความกว้างได้ ทำให้ภาพทั้งสองที่จะนำมาวางทาบกันเพื่อหาค่าสัมประสิทธิ์ความเหมือนอาจจะมีความกว้างไม่เท่ากัน ดังนั้นการทำงานของโปรแกรมที่ออกแบบในส่วนนี้ สำหรับภาพที่มีความกว้างมากกว่า โปรแกรมจะพิจาณาหาค่าสัมประสิทธิ์ความเหมือนถึงเฉพาะส่วนที่วางทับกับภาพ ที่มีความกว้างน้อยกว่าเท่านั้น ซึ่งอาจจะทำให้ได้ค่าสัมประสิทธิ์ความเหมือนที่มีค่าน้อย ทั้ง ๆ ที่ภาพทั้งสองเป็นภาพตัวเลขหรือพยัญชนะตัวเดียวกัน ดังนั้นแนวทางในการพัฒนาโปรแกรมในส่วนการรู้จำตัวเลขและพยัญชนะ คือ ออกแบบโปรแกรมให้สามารถปรับขนาดของภาพตัวเลขและพยัญชนะที่แยกออกมาได้จากแผ่นป้ายทะเบียนให้มีขนาดเท่ากับภาพตัวเลขและพยัญชนะต้นแบบได้ทั้งความสูงและความกว้าง ซึ่งจะทำให้ได้ค่าสัมประสิทธิ์ความเหมือนเมื่อนำภาพทั้งสองไปวางทาบกันที่สูงขึ้น

2. ใช้วิธีการรู้จำตัวเลขและพยัญชนะ ที่มีความซับซ้อนมากกว่าการใช้เทคนิค Template Matching อย่างเช่น การรู้จำตัวเลขและพยัญชนะโดยใช้โครงข่ายประสาทเทียม (Artificial Neural Network) เป็นต้น

5.3 วิจารณ์ผลการทดลอง

จากผลการทดลองพบว่า ความผิดพลาดจากการทดลองโดยส่วนมากจะเกิดขึ้นกับการรู้จำแผ่นป้ายทะเบียนรถยนต์ ซึ่งมีสาเหตุมาจากข้อจำกัดของการรู้จำตัวเลขและพยัญชนะโดยใช้เทคนิค Template Matching เพราะในการนำภาพตัวเลขและพยัญชนะที่แยกออกมาได้จากแผ่นป้ายทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไปวางทาบกับตัวเลขและพยัญชนะต้นแบบเพื่อที่จะหาค่าสัมประสิทธิ์ความเหมือน การที่จะได้ค่าสัมประสิทธิ์ความเหมือน มาก ๆ หรือมีค่าเกือบ 1 นั้น ภาพทั้ง 2 ภาพจะต้องวาบทาบกันได้พอดี แต่ว่าภาพตัวเลขและพยัญชนะแม่แบบทั้งหมดที่ใช้ทำการทดลอง เป็นภาพตัวเลขและพยัญชนะแบบตรง ส่วนภาพตัวเลขและพยัญชนะที่แยกออกมาจากภาพแผ่นป้ายทะเบียนบางภาพอาจจะมีความเอียง หรือมีส่วนที่เป็นหาง อย่างเช่นตัวอักษร ฟ, ส และ ช ไม่ชัดเจน ซึ่งถ้านำภาพตัวเลขและพยัญชนะที่มีลักษณะแบบนี้ไปวางทาบกับตัวเลขและพยัญชนะต้นแบบ อาจจะทำให้โปรแกรมประมวลผลผิดพลาดได้ อย่างเช่นภาพพยัญชนะ ส ที่แยกออกมาได้จากแผ่นป้ายทะเบียนมีส่วนหางไม่ชัดเจน ทำให้เมื่อนำภาพพยัญชนะ ส นี้ ไปวางทาบกับภาพพยัญชนะ ล ของภาพแม่แบบมีค่าสัมประสิทธิ์ความเหมือนมากกว่าเมื่อนำไปวางทาบกับภาพพยัญชนะ ส ของภาพแม่แบบ เมื่อเป็นเช่นนี้โปรแกรมก็จะประมวลผลภาพพยัญชนะ ส นี้เป็นพยัญชนะ ล แทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. Digital Image Processing Using MATLAB

Rafael C. Gonzalez University of Tennessee

Richard E. Woods MedData Interactive

Steven L. Eddins

2. Digital Signal and Image Processing using MATLAB

Gérard Blanchet, Maurice Charbit

3. การใช้ MATLAB สำหรับงานทางวิศวกรรม

โศรฎา แข็งการ และ กนต์ธร ชำนิประศาสน์

4. An Introduction to Digital Image Processing with Matlab Notes for SCM2511 Image Processing 1

Semester 1, 2004 Alasdair McAndrew

School of Computer Science and Mathematics

Victoria University of Technology

5. Digital Image Processing *Second Edition*

Instructorzs Manual

Rafael C. Gonzalez,Richard E. Woods

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในโครงงานนี้ ผู้จัดทำโครงงานใช้คำสั่งต่าง ๆ ที่มีอยู่ในโปรแกรม MATLAB เพื่อช่วยในการทำงานของโปรแกรม ซึ่งมีทั้งหมดดังนี้

1. คำสั่ง videoinput

เป็นคำสั่งรับภาพจากกล้องวีดีโอเข้ามา

2. คำสั่ง triggerconfig

เป็นการจับภาพที่เราต้องการจากวีดีโอ มีการตั้งค่าสองแบบ คือ อัตโนมัติ (Auto) และกำหนดค่า (Manual) เลือกการจับภาพแบบ Manual เพื่อรับเฉพาะภาพในเวลาที่ต้องการเท่านั้น ทำให้ง่ายต่อการประมวลผล การกำหนดค่าในการจับภาพสามารถทำการกำหนดได้ว่า ในหนึ่งรูปที่ได้นั้นจะมาจากการจับภาพกี่ครั้ง

3. คำสั่ง FramesPerTrigger

เป็นการเลือกจำนวนการจับภาพต่อหนึ่งรูป เลือกการจับภาพหนึ่งครั้งต่อหนึ่งภาพรูป

4. คำสั่ง TriggerRepeat

เป็นการกำหนดค่าการจับภาพ สามารถกำหนดได้หลายรูปแบบ เช่น 0 จะทำงานหนึ่งครั้งตามเงื่อนไขการจับภาพแรกที่เจอ หากเป็นตัวเลขค่าบวก ตัวเลขนั้นจะมาเป็นค่าเวลา ที่ใช้เริ่มต้นการทำงาน เมื่อเจอเงื่อนไขการจับภาพครั้งแรกที่เจอ และถ้าเป็น Inf การจับภาพจะทำงานตลอดเวลาตามเงื่อนไขการจับภาพที่กำหนด และหยุดเมื่อพบคำสั่งหยุด เลือกใช้ Inf

5. คำสั่ง ReturnedColorSpace

การกำหนดรูปแบบสีของภาพ ต้องการภาพเช่นเดียวกับภาพที่มนุษย์มองเห็น เลือกเป็นภาพสี RGB

6. คำสั่ง getdata

เป็นการรับข้อมูลภาพเพื่อนำมาเก็บไว้ในค่าตัวแปรที่กำหนด

7. คำสั่ง stop(vid), clear vid;

เป็นคำสั่งหยุดและล้างวีดีโอที่กำลังรับเข้ามา

8. คำสั่ง imresize

เป็นคำสั่งปรับขนาดภาพตามที่ต้องการ ทั้งการเพิ่มและลดขนาด โดยการกำหนดความกว้างและความยาวของภาพใหม่ตามที่ต้องการ

9. คำสั่ง Imwrite

เป็นการเก็บข้อมูลภาพไว้ในไฟล์ที่ต้องการ

10. คำสั่ง imclearborder

คำสั่งนี้จะล้างกลุ่มพิกเซลสีขาวที่มีบริเวณอยู่ติดกับขอบภาพออกไป โดยใช้จุดข้างเคียง 4 หรือ 8 พิกเซล ของหน้าต่างขนาด 3x3 เป็นจุดพิจารณาในการลบพิกเซลสีขาวนั้น เขียนเป็นคำสั่งได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
IM2 = imclearborder(IM,conn)
```

IM คือ ข้อมูลภาพ Input ที่เป็น Grayscale Image หรือ Binary Image

Conn คือ จำนวนจุดข้างเคียงที่ใช้พิจารณาในการลบพิกเซลสีขาวมี 2 ค่า คือ 4 และ 8

IM2 คือ ข้อมูลภาพ Output ที่ได้ ซึ่งข้อมูลภาพที่ได้จะเป็นชนิดเดียวกับ ข้อมูลภาพ Input ที่ป้อนเข้าไป

2. คำสั่ง imerode

คำสั่งนี้ทำให้ภาพคมชัดขึ้นและลบสิ่งรบกวนจุดเล็ก ๆ ออกไป ซึ่งก่อนที่จะใช้คำสั่งนี้จะต้องมีการเลือกใช้โครงสร้างของรูปที่จะนำไปทำการกัด (erode) ก่อน เช่น disk, diamond, pair เป็นต้น โดยใช้คำสั่ง strel เป็นตัวกำหนดโครงสร้างของรูป หลังจากเลือกโครงสร้างที่จะนำไปทำการกัดแล้ว จะต้องกำหนดค่ารัศมีหรือความกว้างของโครงสร้างซึ่งขึ้นอยู่กับชนิดที่เลือก อย่างเช่น เลือกโครงสร้างชนิดสี่เหลี่ยมจัตุรัสด้วยความกว้าง 4 พิกเซล เขียนเป็นคำสั่งได้ se = strel('square',4) แล้วจึงตามด้วยคำสั่ง imerode เขียนเป็นคำสั่งได้ดังนี้

```
originalBW = imread('circles.png');
se = strel('square',4);
erodedBW = imerode(originalBW,se);
```

originalBW คือ ข้อมูลภาพ Input ที่รับเข้ามา เป็น Grayscale Image หรือ Binary Image

3. คำสั่ง imresize

เป็นคำสั่งที่ใช้ปรับขนาดภาพให้มีขนาดตามที่ต้องการทั้งการเพิ่มและลดขนาดภาพ สำหรับภาพที่มีขนาดใหญ่ จะทำให้การประมวลช้า จึงต้องมีการลดขนาดภาพ หรือถ้าภาพมีขนาดเล็กก็อาจต้องมีการขยายภาพเพื่อดูรายละเอียด เป็นต้น สามารถเขียนเป็นคำสั่งได้ดังนี้

```
B = imresize(A, scale)
```

หรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
B = imresize(A, [mrows ncols])
```

เมื่อ A คือ ข้อมูลภาพ Input ที่เราต้องการปรับขนาด สามารถเป็นได้ทั้งข้อมูลภาพแบบ RGB Image, Grayscale Image และ Binary Image

คำสั่งแรก เป็นการปรับขนาดภาพให้มีอัตราส่วนตามที่ต้องการ โดยผู้ใช้กำหนดค่าสเกล (Scale) ของภาพเข้าไปในคำสั่งเอง อย่างเช่นถ้าผู้ใช้กำหนดให้ตัวแปร scale เป็น 2 โปรแกรมจะปรับขนาดของภาพให้มีความสูงและความกว้างเป็น 2 เท่าของความกว้างและความสูงเดิม ส่วนคำสั่งต่อมาเป็นการปรับขนาดภาพให้มีความกว้างและความสูงตามที่ผู้ใช้ต้องการ โดยกำหนดค่า mrows ซึ่งหมายถึงความสูงของภาพในหน่วยพิกเซล และ ncols ซึ่งหมายถึงความกว้างของภาพในหน่วยพิกเซลเข้าไปในคำสั่ง แต่ถ้าจะไม่กำหนดค่าใดค่าหนึ่งเพื่อใส่ไปในคำสั่ง ก็สามารถทำได้โดยแทนค่า mrows หรือ ncols ด้วยตัวแปร NaN (Not a number) ซึ่งโปรแกรมจะปรับขนาดของข้อมูลภาพให้สมดุลกับค่าความสูงหรือความกว้างที่เรากำหนดไว้โดยอัตโนมัติ อย่างเช่น

```
B = imresize(A, [NaN ncols])
```

หรือ

```
B = imresize(A, [mrows NaN])
```

เป็นต้น

4. คำสั่ง imrotate

เป็นคำสั่งที่ใช้ในการหมุนภาพให้อยู่ในตำแหน่งที่ต้องการ ซึ่งสามารถหมุนได้ทั้งแบบตามและทวนเข็มนาฬิกา ถ้ากำหนดค่ามุมเป็นบวก ภาพจะหมุนไปในทิศทางทวนเข็มนาฬิกา แต่ถ้ากำหนดค่ามุมเป็นลบ ภาพก็จะหมุนไปในทิศทางตามเข็มนาฬิกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
B = imrotate(A,angle)
```

เมื่อ A คือ ข้อมูลภาพ Input ที่ต้องการหมุน และ angle คือมุมที่ต้องการให้ภาพหมุนในหน่วยองศา

5. คำสั่ง corr2

เป็นคำสั่งที่ใช้ในการหาค่าความเหมือน หรือค่า Correlation Coefficient ระหว่างภาพ 2 ภาพหรือเมตริกซ์ 2 เมตริกซ์ ใช้สำหรับภาพและเมตริกซ์ 2 มิติ (ซึ่งถ้าเป็นภาพก็หมายถึงภาพแบบ Gray Scale และ Binary) โดยภาพหรือเมตริกซ์ทั้ง 2 นี้จะต้องมีขนาด (row และ column) เท่ากัน สามารถเขียนคำสั่งได้ดังนี้

```
r = corr2(A,B)
```

โดย A และ B คือ ภาพหรือเมตริกซ์ A และ B ตามลำดับ

r คือ ค่า Correlation Coefficient ซึ่งมีค่าตั้งแต่ -1 ถึง 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้