สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

# การรู้จำตัวอักษรบนพื้นฐานลักษณะเด่น

# FEATURE BASED CHARACTER RECOGNITION

T110820

นางสางกาญจนา หน่อแก้ว
นางสาวขนิษฐา นาคทอง
นายจตุพร ภูดิศชินภัทร

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขหมู่.....
เลขทะเบียน 110820
วัน,เดือน,ปี 18 พ.ย. 2553

b.....
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัญฑิต
สาขาวิชาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2552

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# FEATURE BASED CHARACTER RECOGNITION

KANJANA NORKAEW

KANITTHA NAKTONG

JATUPORN PHUDITSHINNAPATRA

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUE OF TECHNOLOGY LARDKRABANG

ACADEMTIC YEAR 2009

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**สาขาวิชาวิศวกรรมการวัดคุมและควบคุม**

**คณะวิศวกรรมศาสตร์**

**สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง**

**ใบรับรองปริญญานิพนธ์**

.....

**หัวข้อปริญญานิพนธ์** การรู้จำตัวอักษรบนพื้นฐานลักษณะเด่น

FEATURE BASED CHARACTER RECOGNITION

| | | | | |
|---|---|---|---|---|
| **นักศึกษาผู้จัดทำ** | นางสาวกาญจนา | หน่อแก้ว | **รหัสนักศึกษา** | 49010043 |
| | นางสาวขนิษฐา | นาคทอง | **รหัสนักศึกษา** | 49010079 |
| | นายจตุพร | ภูดิศชินภัทร | **รหัสนักศึกษา** | 49010096 |

**ปริญญา** วิศวกรรมศาสตรบัณฑิต

**สาขาวิชา** วิศวกรรมการวัดคุม

**ปีการศึกษา** 2552

| **อาจารย์ผู้ควบคุมปริญญานิพนธ์** | **ลายมือชื่อ** |
|---|---|
| **รศ. ดร. เกษตร์ ศิริสันติสัมฤทธิ์** | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | | |
|---|---|---|---|---|
| **หัวข้อปริญญานิพนธ์** | การรู้จำตัวอักษรบนพื้นฐานลักษณะเด่น | | | |
| | FEATURE BASED CHARACTER RECOGNITION | | | |
| **นักศึกษาผู้จัดทำ** | นางสาวกาญจนา | หน่อแก้ว | รหัสนักศึกษา | 49010043 |
| | นางสาวขนิษฐา | นาคทอง | รหัสนักศึกษา | 49010079 |
| | นายจตุพร | ภูดิศชินภัทร | รหัสนักศึกษา | 49010096 |
| **อาจารย์ที่ปรึกษา** | รศ. ดร. เกษตร์ | ศิริสันติสัมฤทธิ์ | | |
| **ปีการศึกษา** | 2552 | | | |

## บทคัดย่อ

บทความนี้นำเสนอการรู้จำตัวอักษรบนพื้นฐานลักษณะเด่น และการปรับปรุงการรู้จำตัวอักษรด้วยวิธี TemplateMatching ที่รุ่นพี่ได้ศึกษาวิจัยไว้ในส่วนของการรู้จำตัวอักษรด้วยวิธี Template Matching ขั้นตอนของการคัดเลือกแม่แบบ และขั้นตอนบางขั้นตอนของการรู้จำตัวอักษรบนแผ่นป้ายทะเบียนรถยนต์ได้ถูกปรับปรุงแก้ไข เพื่อให้ได้ค่าความเหมือนของการจับคู่มีค่าสูงมากขึ้นในส่วนของการรู้จำตัวอักษรบนพื้นฐานลักษณะเด่น เนื่องจากตัวอักษรภาษาไทยบนแผ่นป้ายทะเบียนรถยนต์เป็นชนิดหัวทึบทำให้ยากต่อการรู้จำตัวอักษรในขั้นตอนถัดมา ดังนั้นการหาหัวของตัวอักษรจึงเป็นขั้นตอนที่สำคัญ ขั้นตอนต่อมาเป็นการหาเส้นโครงร่างของแต่ละตัวอักษรและหาลักษณะเด่นที่มีอยู่บนตัวอักษร ในผลการทดลองได้แสดงผลลัพธ์ของการหาแผ่นป้ายทะเบียนรถยนต์จากภาพถ่ายด้านหน้าของรถยนต์ การรู้จำตัวอักษรด้วยวิธี TemplateMatching ที่ได้ปรับปรุงและบางส่วนของขั้นตอนการรู้จำตัวอักษรบนพื้นฐานลักษณะเด่น เนื่องจากต้องเสียเวลากับค้นหาวิธีการหาหัวของตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | |
|---|---|---|
| **Thesis Title** | Feature Based Character Recognition | |
| **Authors** | Miss. Kanjana | Norkaew |
| | Miss. Kanittha | Naktong |
| | Mr. Jatuporn | Phuditshinnapatra |
| **Thesis Advisor** | Assoc. Prof. Dr. Kaset | Sirisantisamrid |
| **Year** | 2009 | |


## ABSTRACT

This paper presents the feature based character recognition and improvement of character recognition using Template matching that proposed by our seniors. In part of character recognition using Template matching, the procedure of selection of template images and some part in procedure of character recognition on the license plate are improved to increase the correlation coefficient of matching. In the feature based character recognition, due to the head of Thai characters on license plate are opaque type thereby it is difficult to recognize the Thai character in the next procedures. Therefore, the procedure of determination of head characters becomes an important step. The next procedures are extracting the skeletal line of each character and seeking the existing features on each character. In the experimental results, the results of detecting the license plate from the car images captured in front, improving the character recognition using Template matching and some results of feature based character recognition are shown because we taken so much time to find the determination method of head characters.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# กิตติกรรมประกาศ

ปริญญาที่นิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี เพราะได้รับคำแนะนำและให้ความช่วยเหลือผู้ทำวิจัยจาก รศ.ดร.เกษตร์ ศิริสันติสัมฤทธิ์ ซึ่งผู้ทำวิจัยรู้สึกเป็นเกียรติ์และทราบซึ้งเป็นอย่างยิ่ง จึงใช้โอกาสนี้กราบขอบพระคุณเป็นอย่างสูง

สุดท้ายนี้ขอกราบเท่าขอบพระคุณบิดา มารดา และผู้มีพระคุณของคณะผู้ทำวิจัยทุกท่าน ผู้สนับสนุนให้มีการศึกษาและอบรมเลี้ยงดูจนมีวันสัมฤทธิ์ผล ขออารธนาคุณพระศรีรัตนตรัย และสิ่งศักดิ์สิทธิ์ทั้งหลายที่อยู่ในสากลโลก จงปกป้องคุ้มครองและประทานพรอันประเสริฐให้มีแต่ความสุขความเจริญตลอดไปด้วยเทอญ

**คณะผู้จัดทำ**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

หน้า

**บทคัดย่อภาษาไทย.....I**

**บทคัดย่อภาษาอังกฤษ.....II**

**กิตติกรรมประกาศ.....III**

**สารบัญ.....IV**

**สารบัญตาราง.....VII**

**สารบัญรูป.....VIII**

**บทที่ 1 บทนำ.....1**

1.1 ความสำคัญของปริญญานิพนธ์.....1

1.2 วัตถุประสงค์ของปริญญานิพนธ์.....3

1.3 ขอบเขตของปริญญานิพนธ์.....3

1.4 สมมติฐาน.....3

1.5 ขั้นตอนการศึกษา.....4

1.6 ประโยชน์ที่คาดว่าจะได้รับ.....4

**บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....5**

2.1 การประมวลผลภาพดิจิตอล (Digital Image Processing).....5

2.1.1 การแทนภาพด้วยข้อมูลดิจิตอล.....5

2.1.2 ชนิดของข้อมูลภาพที่ใช้ในการประมวลผลภาพทางดิจิตอล.....7

2.1.2.1 RGB Image.....7

2.1.2.2 ภาพระดับเทา (Gray Scale Image).....8

2.1.2.3 ภาพสองระดับ(Binary Image).....9

2.2 การแบ่งส่วนภาพ (Image Segmentation).....12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ (ต่อ)

หน้า

2.2.1 การแบ่งส่วนภาพด้วยการพิจารณาความต่อเนื่องของข้อมูล.....12

2.2.2 การแยกภาพด้วยวิธี Region Labeling.....12

2.3 เทคนิค Template Matching.....14

2.4 ทฤษฎี The Skeleton of a Region.....14

**บทที่ 3 การรู้จำอักษรด้วยวิธี Template Matching.....19**

3.1 การประมวลผลภาพดิจิตอล.....19

3.2 การแยกส่วนที่เป็นตัวเลขและตัวพยัญชนะออกจากแผ่นป้ายทะเบียน.....28

3.3 การรู้จำตัวเลขและตัวพยัญชนะ.....30

3.3.1 วิธีการจัดทำตัวเลขและตัวพยัญชนะเพื่อนำมาเป็นต้นแบบ.....31

3.3.2 ขั้นตอนการรู้จำตัวเลขและตัวพยัญชนะ.....33

3.3.3 ผลการทดลอง.....33

**บทที่ 4 การรู้จำตัวอักษรพื้นฐานลักษณะเด่น.....42**

4.1 ทดลองใส่วงกลมสีขาวที่หัวของตัวพยัญชนะด้วยโปรแกรม Paint .....43

4.2 ตรวจหาหัวของตัวพยัญชนะด้วยแนวคิดการหาพื้นที่ที่มากที่สุดบนพยัญชนะ ด้วยโปรแกรมMATLAB .....45

4.3 ตรวจหาเพื่อใส่วงกลมสีขาวที่หัวของตัวพยัญชนะด้วยแนวคิดการหาความหนา ของตัวพยัญชนะ ด้วยโปรแกรม MATLAB .....51

**บทที่ 5 การทดลองและผลการทดลอง.....76**

5.1 การแยกตัวเลขและตัวพยัญชนะออกจากแผ่นป้ายทะเบียนรถยนต์.....76

5.2 ผลการทดลองการใส่หัวให้กับตัวพยัญชนะ และการหาเส้นโครง่ร่างของตัวพยัญชนะ.....99

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ (ต่อ)


หน้า

**บทที่ 6 สรุปและวิจารณ์ผลการทดลอง.....109**

6.1 สรุปผลการทดลอง.....109

6.2 ข้อเสนอแนะ.....112

6.3 วิจารณ์ผลการทดลอง.....112

**บรรณานุกรม.....114**

**ภาคผนวก.....115**


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญตาราง

**ตารางที่** **หน้า**

5.1 ผลการทดลองในการโปรแกรมการค้นหาแผ่นป้ายทะเบียนทั้งหมด 40 คัน.....94

5.2 การแยกตัวเลขที่อยู่บนแผ่นป้ายทะเบียนเปอร์เซ็นต์ความถูกต้องทั้งหมด.....96

5.3 การแยกตัวพยัญชนะที่อยู่บนแผ่นป้ายทะเบียนและเปอร์เซ็นต์ความถูกต้องทั้งหมด.....98

5.4 สรุปผลการทดลองการหาและใส่หัวให้กับตัวพยัญชนะ.....108

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป

**รูปที่** **หน้า**


1.1 ภาพการบันทึกแผ่นป้ายทะเบียนรถยนต์.....1

2.1 ข้อมูลรูปดิจิตอลแสดงในรูปเมตริกซ์.....6

2.2 การแทนข้อมูลรูปดิจิตอลด้วยฟังก์ชั่น.....6

2.3 โมเดลสี RGB.....7

2.4 ภาพสี RGB.....8

2.5 ภาพระดับเทา.....9

2.6 ภาพสองระดับ.....11

2.7 ภาพสี RGB ที่ถูกแปลงเป็นภาพสองระดับจากการเลือกค่าเทรชโฮลที่เหมาะสม.....11

2.8 ภาพสี RGB ที่ถูกแปลงเป็นสองระดับจากการเลือกค่าเทรชโฮลที่ไม่เหมาะสม.....12

2.9 การพิจารณาความต่อเนื่องกันแบบ 4 จุด.....13

2.10 การพิจารณาความต่อเนื่องกันแบบ 8 จุด.....13

2.11 หน้าต่างขนาด 3*3 ที่จะใช้ในการลบพิกเซลในตัวอักษร.....15

2.12 การลบขอบของตัวอักษรเพื่อหาเส้นโครงร่างของตัวอักษร.....16

2.13 ผลลัพธ์ที่ได้จากการลบพิกเซลทั้งหมดจนเหลือพิกเซลโครงร่างของตัวอักษร.....17

2.14 ภาพการวางตารางขนาด 3*3 เพื่อทำการลบพิกเซลตามเงื่อนไขที่กำหนด.....18

3.1 ภาพสี RGB บริเวณด้านหน้ารถยนต์ที่มีแผ่นป้ายทะเบียนรวมอยู่ด้วย.....20

3.2 ภาพระดับเทาที่แปลงมาจากภาพสี RGB.....21

3.3 ภาพสองระดับที่ได้จากการแปลงภาพระดับเทาด้วยคำสั่ง im2bw.....21

3.4 ภาพสองระดับภายหลังจากกำจัดกลุ่มพิกเซลสีขาวที่อยู่ติดกับขอบของรูป.....22

3.5 ภาพตัวอย่างปัญหาที่เกิดจากการใช้ค่า Threshold
ที่ไม่เหมาะสมและรูปที่ได้จากปรับระดับเทรชโฮลใหม่.....23

3.6 การกำหนดหมายเลขกลุ่มพิกเซลสีขาวและ
ให้กลุ่มพิกเซลสีขาวที่ใหญ่ที่สุดเป็นแผ่นป้ายทะเบียนรถยนต์.....25

3.7 ภาพสองระดับที่ถูกกลับจากพิกเซลสีขาวเป็นพิกเซลสีดำ
และพิกเซลสีดำเป็นพิกเซลสีขาว.....26



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป (ต่อ)

**รูปที่** **หน้า**

3.8 การกำหนดหมายเลขและกำจัดส่วนที่เป็นชื่อจังหวัดออกไป.....27
3.9 ภาพที่ได้จากการกำจัดกลุ่มพิกเซลสีขาวที่ติดขอบ
และสิ่งรบกวนต่างๆที่อยู่บนแผ่นป้ายทะเบียน.....27
3.10 ภาพแสดงแถวที่จะใช้เป็นแกนอ้างอิงในการตัดขอบเขตบนสุดและขอบเขตล่างสุด.....28
3.11 ภาพที่ได้จากการใช้แกนอ้างอิงและทำการตัดขอบเขตบนสุดและขอบเขตล่างสุด.....29
3.12 ภาพแสดงขอบเขตด้านซ้ายสุดและขอบเขตด้านขวาสุด.....29
3.13 ภาพผลที่ได้จากการนำ Column ขอบซ้ายและขอบขวามาแสดง.....30
3.14 ตัวเลขและตัวพยัญชนะที่ถูกแยกออกมาจากแผ่นป้ายทะเบียน.....30
3.15 ชุดแม่แบบหรือชุด template ทั้งหมดที่ใช้ในการทดลอง.....32
3.16 ภาพตัวพยัญชนะที่ต้องการพิจารณา.....33
3.17 หน้าต่าง Command window ที่เป็นผลมาจากการนำเอาข้อมูลรูปตัวเลขและตัวพยัญชนะ
ทุกตัวที่อยู่บนแผ่นป้ายทะเบียนรถยนต์มาเข้าสู่การรู้จำตัวเลขและตัวพยัญชนะ.....34
3.18 ภาพที่ได้จากการนำรูปแผ่นป้ายทะเบียน ธษ 8988
ไปประมวลผลในส่วนการรู้จำตัวเลขและตัวพยัญชนะ.....35
4.1 ภาพตัวพยัญชนะที่ได้มาจากแผ่นป้ายทะเบียนรถยนต์.....42
4.2 ภาพตัวพยัญชนะเมื่อหาเส้นโครงร่างเสร็จแล้ว.....43
4.3 ภาพการหาเส้นโครงร่างของตัวพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์.....43
4.4 ภาพเปรียบเทียบให้เห็นความแตกต่าง.....44
4.5 ภาพตัวพยัญชนะที่โหลดขึ้นมาจากฐานข้อมูลในคอมพิวเตอร์.....45
4.7 (ก) ภาพที่ยังไม่ได้ทำการกำจัดสิ่งรบกวนด้วยคำสั่งmedianfilter.....45
(ข) ภาพที่ได้ทำการกำจัดสิ่งรบกวนด้วยคำสั่ง medianfilter.....46
4.8 ภาพที่ได้ภายหลังการกลับภาพตัวพยัญชนะจากสีขาวเป็นสีดำ.....46
4.9 ภาพที่ได้ทำการแบ่งพยัญชนะออกเป็นสามส่วนในแนวนอน.....47
4.10 ภาพที่ได้ทำการแบ่งพยัญชนะออกเป็นสามส่วนในแนวตั้ง.....47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป (ต่อ)


**รูปที่** **หน้า**

4.11 ภาพที่ได้ทำการแบ่งตัวพยัญชนะออกเป็น 9 ช่อง……..……..48

4.12 ภาพได้ทำการแสดงหมายเลข label
ของพิกเซลสีขาวของแต่ละช่องที่ได้ทำการแบ่งไว้……48

4.13 ภาพที่ได้ทำการดึงส่วนที่มีกลุ่มพิกเซลสีขาวที่มากที่สุดออกมา……49

4.14 ภาพที่ได้ทำการสร้างพื้นที่สี่เหลี่ยมไปตรงกลางของกลุ่มพิกเซลสีขาวที่ดึงขึ้นมา……..49

4.16 ภาพผลการเปรียบเทียบระหว่างรูปที่ได้แก้ไขกับรูปที่ยังไม่ได้แก้ไข……..49

4.17 (ก) ภาพเส้นโครงร่างตัวพยัญชนะที่ใส่หัวแล้วแล้ว……....50
(ข) ภาพเส้นโครงร่างเปรียบเทียบระหว่างตัวพยัญชนะที่ไม่ได้ใส่หัวและใส่หัวแล้ว……50

4.18 ตัวพยัญชนะบ้างตัวกลุ่มพิกเซลสีขาวที่มากที่สุดก็ไม่ใช่ส่วนหัวเสมอไป……..……51

4.19 แบบแผนแสดงตำแหน่งส่วนหัวของตัวพยัญชนะ……..52

4.20 ไดอะแกรมแสดงรูปแบบโปรแกรมหาส่วนหัวของตัวพยัญชนะ……53

4.21 แสดงการหาเส้นสมมติ……...53

4.22 ภาพแสดงแถว (row) ที่มีความหนามากที่สุดของตัวพยัญชนะ……....55

4.23 ภาพแสดงขอบเขตที่ใช้ในการหาส่วนหัวของตัวพยัญชนะที่อยู่ช่วงบน……55

4.24 ภาพแสดงแถว (row) และหลัก (column) ที่มีพิกเซลสีขาวมากที่สุดแล้วถูกตีกรอบ……....56

4.25 ภาพแสดงการใช้พิกเซล 3*3 ในการหาความหนาของกลุ่มพิกเซลสีขาวตามแถว (row) …....56

4.26 ภาพแสดงขอบเขตความกว้างของกรอบที่ตีบริเวณส่วนหัวของตัวพยัญชนะ……..…57

4.27 ภาพแสดงการใช้พิกเซล 3*3 ในการหาความหนาของกลุ่มพิกเซลสีขาวตามหลัก (column)..57

4.28 ภาพแสดงขอบเขตความสูงของกรอบที่ตีบริเวณส่วนหัวของตัวพยัญชนะ……58

4.29 ภาพแสดงตัวอย่างตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงบน……58

4.30 ภาพแสดงสี่เหลี่ยม 3*3 ที่สร้างขึ้นโดยใช้กรอบที่ตีบริเวณส่วนหัว……59

4.31 ภาพแสดงลักษณะของตัวพยัญชนะที่มีหัวอยู่ช่วงบนเมื่อใส่หัวให้กับตัวพยัญชนะแล้ว…….59

4.32 ภาพแสดงหลัก (column) ที่อยู่ตรงกลางของตัวพยัญชนะ……..60

4.33 ภาพแสดงวิธีการหากลุ่มพิกเซลสีขาวในหลัก (column) ตรงกลางตัวพยัญชนะ……..……....60

4.34 ภาพขอบเขตการหาหัวของตัวพยัญชนะที่มีหัวอยู่ช่วงล่าง……..……....61



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป (ต่อ)


**รูปที่** **หน้า**

4.35 ภาพการหาส่วนหัวของตัวพยัญชนะที่มีหัวอยู่ช่วงล่าง.....62

4.36 ภาพตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงล่างแต่มีลักษณะพิเศษ.....63

4.38 ภาพแบ่งความสูงของตัวพยัญชนะออกเป็นสองส่วนเท่าๆกันได้แถวตรงกลาง.....63

4.39 ภาพการหากลุ่มพิกเซลสีขาวตามแถว (row) ที่อยู่ตรงกลาง.....64

4.40 ภาพขั้นตอนการหาส่วนหัวของตัวพยัญชนะที่มีหัวอยู่ช่วงล่าง.....64

4.41 ภาพขั้นตอนการหาส่วนหัวของตัวพยัญชนะที่มีหัวอยู่ช่วงกลาง.....65

4.42 ภาพการแบ่งความสูงของตัวพยัญชนะออกเป็นสองส่วน.....65

4.43 ภาพผลที่ได้จากการสำรวจจำนวนกลุ่มพิกเซลสีขาว.....66

4.44 ภาพแสดงการหาพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดในช่วงกลางและ
ช่วงล่างของตัวพยัญชนะโดยพื้นที่ในช่วงกลางมีค่ามากกว่าช่วงล่าง.....68

4.45 ภาพแสดงการหาพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดในช่วงกลางและ
ช่วงล่างของตัวพยัญชนะโดยพื้นที่ในช่วงกลางมีค่าน้อยกว่าช่วงล่าง.....69

4.46 ภาพแสดงขั้นตอนการหาส่วนหัวและใส่หัวให้กับตัวพยัญชนะ
และแสดงการหาขอบเขตของเงื่อนไขที่ 1 ประเภทที่ 1.....70

4.47 ภาพแสดงขั้นตอนการหาส่วนหัวและใส่หัวให้กับตัวพยัญชนะ
และแสดงขอบเขตของเงื่อนไขที่ 1 ประเภทที่ 2.....71

4.48 ภาพแสดงขั้นตอนการหาส่วนหัวและใส่หัวให้กับตัวพยัญชนะ
ที่มีส่วนหัวของตัวพยัญชนะอยู่ในช่วงล่าง.....71

4.49 ภาพการหาพื้นที่พิกเซลสีขาวที่มากที่สุดของช่วงกลางของตัวพยัญชนะ.....72

4.50 ภาพการหาพื้นที่พิกเซลสีขาวที่มากที่สุดของช่วงล่างของตัว.....73

4.51 ภาพการหาขอบเขต การหาส่วนหัวและใส่หัวให้กับตัวพยัญชนะ
ที่มีส่วนหัวของตัวพยัญชนะอยู่ในช่วงกึ่งกลาง.....74

4.52 การหาส่วนหัวและใส่หัวให้กับตัวพยัญชนะ ที่มีส่วนหัวของตัวพยัญชนะอยู่ในช่วงล่าง.....75



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป (ต่อ)

**รูปที่** **หน้า**


5.1 ภาพถ่ายด้านหน้ารถยนต์ที่เป็นภาพสี RGB.....76

5.2 ภาพผลลัพธ์ที่ได้จากการแปลงภาพสี RGB ให้เป็นภาพระดับเทา.....77

5.3 ภาพผลลัพธ์จากการแปลงภาพระดับเทาให้เป็นภาพสองระดับ.....77

5.4 ภาพผลลัพธ์ที่ทำการกำจัดกลุ่มพิกเซลสีขาวที่ติดกับขอบภาพออกไป.....78

5.5 ภาพที่ได้จากการนำคู่พิกัดของกลุ่มพิกเซลสีขาวที่มีพื้นที่มากที่สุด
ไปคัดลอกจากภาพสี RGB ที่รับเข้ามาในขั้นตอนแรก.....78

5.7 ภาพผลลัพธ์จากการกลับภาพสองระดับ.....79

5.8 ภาพผลลัพธ์ที่ได้จากการกำจัดส่วนที่เป็นชื่อจังหวัดออกไป.....79

5.9 ภาพผลลัพธ์ที่ได้จากการตัดขอบบนและขอบล่างให้กับกลุ่มตัวเลขและตัวพยัญชนะ.....79

5.10 ภาพค่าขอบเขตด้านซ้ายและขอบเขตด้านขวาที่เก็บอยู่ในตัวแปร T1 และ U1 ตามลำดับ.....80

5.11 ภาพผลลัพธ์หลังจากกลับภาพสองระดับ.....80

5.12 ภาพตัวเลขและตัวพยัญชนะที่ถูกแยกออกมาจากแผ่นป้ายทะเบียน.....80

5.13 ภาพผลการทดลองรถยนต์ทะเบียน ฌผ 8707.....81

5.14 ภาพผลการทดลองรถยนต์ทะเบียน กค 3065.....82

5.15 ภาพผลการทดลองรถยนต์ทะเบียน ชฟ 2468.....83

5.16 ภาพผลการทดลองรถยนต์ทะเบียน ชว 7565.....84

5.17 ภาพผลการทดลองรถยนต์ทะเบียน ฌช 968.....85

5.18 ภาพผลการทดลองรถยนต์ทะเบียน พพ 7654.....86

5.19 ภาพผลการทดลองรถยนต์ทะเบียน ยบ 4708.....87

5.20 ภาพผลการทดลองรถยนต์ทะเบียน สล 4677.....88

5.21 ภาพผลการทดลองรถยนต์ทะเบียน ฉษ 6469.....89

5.22 (ก) แปลงภาพสี RGB เป็นภาพสองระดับ.....90

(ข) ภาพสองระดับของภาพสี RGB ที่เกิดจากการปรับค่าเทรชโฮลอัตโนมัติ.....90

(ฃ) ภาพผลลัพธ์จากการโปรกแกรมซึ่งได้รูปไฟด้านหน้ารถยนต์.....90

(ค) แปลงภาพสี RGB เป็นภาพสองระดับ.....91



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป (ต่อ)

**รูปที่** **หน้า**

(ศ) ภาพสองระดับ ของภาพสีRGB ที่เกิดจากการปรับค่าเทรชโฮลเอง.....91
(ฆ) ภาพผลลัพธ์จากการโปรแกรมซึ่งได้ภาพที่เป็นกระจังหน้ารถยนต์มาแทน.....91
(ง) แปลงภาพสี RGB เป็นภาพสองระดับ.....92
(จ) ภาพสองระดับของภาพสี RGB ที่เกิดจากการปรับค่าเทรชโฮลเอง.....92
(ฉ) ภาพผลลัพธ์จที่เกิดจากการปรับค่าเทรชโดยอัตโนมัติ.....92
(ช) ภาพที่ทำการปรับค่าความสว่างด้วยโปรแกรม Photoscape.....93
(ซ) ภาพจากการแปลงภาพสี RGB เป็นภาพสองระดับ.....93
(ฌ) ภาพผลลัพธ์ที่เกิดจากการปรับค่าเทรชโฮลอัตโนมัติ.....93
(ญ) ภาพผลลัพธ์การค้นหาแผ่นป้ายทะเบียนรถยนต์และกลับภาพสองระดับ.....93
(ฎ) ภาพผลลัพธ์หลังจากทำการลบส่วนที่เป็นชื่อจังหวัดออกไป.....93
5.23 ภาพถ่ายด้านหน้ารถยนต์ทั้งหมดที่ใช้ในการทดลอง.....95
5.24 ภาพการเปรียบเทียบค่าสัมประสิทธิ์ความเหมือนในวิธีการ Template Matching .....99
5.25 ภาพแสดงตัวพยัญชนะทั้งหมดที่หาได้บนป้ายทะเบียนรถยนต์.....99
5.26 ภาพแสดงเส้นสมมติบนตัวพยัญชนะแต่ละตัวที่มีส่วนหัวอยู่ช่วงบน.....100
5.27 ภาพการหาและใส่หัวให้กับตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงบนและหาเส้นโครงร่าง.....101
5.28 ภาพตัวพยัญชนะห.หีบ ที่ได้ส่วนหัวด้านหลังออกมาแทน.....102
5.29 ภาพการหาและใส่หัวให้กับตัวพยัญชนะที่มีส่วนหัวอยู่กึ่งกลาง
และหาเส้นโครงร่างของตัวพยัญชนะ.....103
5.30 ภาพการหาและใส่หัวให้กับตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงกลางด้านหน้า
และหาเส้นโครงร่างของตัวพยัญชนะ.....104
5.31 ภาพการหาและใส่หัวให้กับตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงล่าง
และหาเส้นโครงร่างของตัวพยัญชนะ.....105
5.32 ภาพการหาและใส่หัวให้ตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงล่างแล้วหาเส้นโครงร่าง.....106
5.33 ภาพการหาและใส่หัวให้ตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงล่าง 2 หัว
และได้ด้านหลังออกมา.....106
5.34 ภาพตัวพยัญชนะที่เกิดปัญหาขึ้นจากสาเหตุที่หัวของตัวพยัญชนะไม่อยู่ในตำแหน่งพอดี...107

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1
# บทนำ

## 1.1 ที่มาและความสำคัญของปริญญานิพนธ์

ในการจอดรถตามสถานที่ต่างๆ โดยปรกติแล้วจะมีเจ้าหน้าที่คอยแจกบัตรที่จอดรถ คิดค่าจอดรถเวลาเข้า-ออก ซึ่งการทำงานของมนุษย์นั้นมักจะมีข้อผิดพลาดเกิดขึ้น จึงมีแนวคิดที่จะนำเทคโนโลยีเข้ามาใช้เพื่อลดข้อผิดพลาด และเพิ่มประสิทธิภาพในการทำงานด้วยโปรแกรมการตรวจสอบแผ่นป้ายทะเบียนรถยนต์โดยใช้วิธีในการตรวจจับภาพแผ่นป้ายทะเบียน และใช้โปรแกรมช่วยในการวิเคราะห์ภาพเพื่อให้คอมพิวเตอร์สามารถอ่านและเข้าใจได้ว่าทะเบียนรถนั้นเป็นหมายเลขใด

การนำโปรแกรมเข้ามาใช้นั้นจะช่วยแก้ปัญหาจากความผิดพลาดของมนุษย์ เนื่องจากโปรแกรมมีประสิทธิภาพในการทำงานที่ดีและรวดเร็วจึงช่วยลดเวลาในการเข้า-ออก หรือรับบัตรจอดรถตามสถานีจอดรถต่างๆ ในส่วนโปรแกรมนี้สามารถพัฒนาและนำไปประยุกต์ใช้กับกระบวนการอื่นๆ อาทิเช่น โปรแกรมตรวจจับรถยนต์ที่ขับฝ่าไฟแดง หรือใช้เป็นระบบรักษาความปลอดภัยของอาคารต่างๆ หรือใช้ตรวจจับแผ่นป้ายทะเบียนรถยนต์ของคนร้าย และใช้ในการตรวจสอบแผ่นป้ายทะเบียนเมื่อผ่านด่านเก็บเงินตามทางหลวง เป็นต้น

**รูปที่ 1.1** ภาพการบันทึกแผ่นป้ายทะเบียนรถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างไรก็ตามจากการศึกษาค้นคว้าวิจัย เรื่องการตรวจสอบแผ่นป้ายทะเบียนที่รุ่นพี่ได้ทำไว้ ซึ่งขั้นตอนการทำงานเริ่มตั้งแต่การถ่ายภาพด้านหน้ารถยนต์ที่มีแผ่นป้ายทะเบียน จากนั้นทำการตรวจสอบหาบริเวณที่คาดว่าเป็นแผ่นป้ายทะเบียนด้วย Software และเมื่อทราบตำแหน่งหรือบริเวณที่เป็นแผ่นป้ายทะเบียนแล้วก็จะทำแยกตัวพยัญชนะและตัวเลขออกจากแผ่นป้ายทะเบียน จากนั้นทำการรู้จำตัวพยัญชนะและตัวเลขโดยวิธี Template Matching กล่าวคือนำตัวพยัญชนะและตัวเลขทีละตัวไปเปรียบเทียบกับตัวพยัญชนะและตัวเลขแม่แบบ ที่ถูกจัดเก็บไว้ในหน่วยความจำโดยการคำนวณหาค่าสัมประสิทธิ์ความเหมือน (Correlation Coefficient) ระหว่างตัวพยัญชนะที่ต้องการพิสูจน์กับตัวพยัญชนะแม่แบบ ถ้าค่าสัมประสิทธิ์ความเหมือนที่คำนวณได้มีค่ามากกว่าค่า Threshold ที่กำหนดไว้ก็จะพิจารณาว่าตัวพยัญชนะหรือตัวเลขนั้นก็คือตัวพยัญชนะหรือตัวเลขแม่แบบตัวนั้น อย่างไรก็ตามผลการทดลองที่ได้รับตามที่ได้รายงานไว้ในปริญญานิพจน์ไม่ค่อยดีมากนัก ด้วยเหตุนี้คณะผู้วิจัยจึงมีแนวความคิด 2 ประการคือ ประการแรกปรับปรุงแก้ไขขั้นตอนบางขั้นตอนของการรู้จำตัวพยัญชนะและตัวเลขด้วยวิธีการ Template Matching เพื่อให้ได้ค่าสัมประสิทธิ์ความเหมือนระหว่างตัวพยัญชนะที่ต้องการรู้จำกับตัวพยัญชนะแม่แบบมีค่าสูงขึ้น และเพื่อให้การรู้จำตัวพยัญชนะและตัวเลขถูกต้องเที่ยงตรงมากยิ่งขึ้น ประการถัดมาคณะผู้วิจัยมีแนวความคิดว่าควรจะหาวิธีการรู้จำตัวพยัญชนะใหม่ เพราะข้อเสียของการรู้ตัวพยัญชนะด้วยวิธี Template Matching ก็คือขนาดของตัวพยัญชนะที่ต้องรู้จำและของตัวพยัญชนะแม่แบบจะต้องมีขนาดเท่ากัน นอกจากนั้นตัวพยัญชนะที่ต้องการรู้จำจะต้องไม่เอียงหรือเมื่อตรวจพบว่า มีการเอียงก็จะต้องหมุนตัวพยัญชนะกลับมาให้ตรงก่อนที่จะทำการ Matching ได้ อย่างไรก็ตาม วิธีการเก็บภาพด้านหน้ารถยนต์ที่มีแผ่นป้ายทะเบียนที่ใช้ในงานวิจัยนี้และที่รุ่นพี่ทำไว้ ก็คือการเดินถ่ายภาพดังนั้นภาพถ่ายที่ได้มาอาจมีการเอียงไปบ้าง ขนาดแผ่นป้ายทะเบียนเล็กใหญ่แตกต่างกัน เป็นต้น สำหรับการรู้จำตัวอักษรวิธีใหม่ที่นำเสนอในปริญญานิพนธ์ฉบับนี้เป็นการรู้จำตัวพยัญชนะบนพื้นฐานของลักษณะเด่นที่มีอยู่ในแต่ละตัวพยัญชนะ ซึ่งขั้นตอนการรู้จำตัวพยัญชนะจะเริ่มตั้งแต่การหาส่วนหัวของตัวอักษรการหาโครงร่างของตัวพยัญชนะแต่ละตัว จากนั้นตรวจหาลักษณะเด่นที่มีอยู่บนตัวพยัญชนะแต่ละตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.2 วัตถุประสงค์ของปริญญานิพนธ์

1. ศึกษาความเป็นไปได้ของวิธีรู้จำตัวอักษรบนพื้นฐานลักษณะเด่นเพื่อจะนำไปใช้กับการรู้จำตัวอักษรบนแผ่นป้ายทะเบียนรถยนต์

2. ทำการแก้ไขและปรับปรุงวิธีการ Template Matching ของรุ่นพี่ที่ได้ทำไว้ก่อนหน้านี้

3. กรณีที่วิธีการรู้จำตัวอักษรบนพื้นฐานลักษณะเด่นใช้งานได้ ก็จะนำไปใช้งานเพื่อการรู้จำตัวอักษรและตัวเลขต่อไป และนำผลที่ได้รับไปเปรียบเทียบกับการรู้จำตัวอักษรด้วยวิธี Template Matching

## 1.3 ขอบเขตของปริญญานิพนธ์

1. สามารถรู้จำตัวพยัญชนะภาษาไทยและตัวเลขอารบิค ให้มากที่สุดเท่าที่จะมากได้

2. เปรียบเทียบผลที่ได้จากวิธีรู้จำตัวอักษรบนพื้นฐานลักษณะเด่นกับวิธี Template Matching เพื่อศึกษาว่าวิธีใดให้ผลลัพธ์ที่ดีกว่า

3. นำไปประยุกต์ใช้กับการรู้จำตัวอักษรบนแผ่นป้ายทะเบียนรถยนต์

## 1.4 สมมติฐาน

มนุษย์มีความสามารถที่จะทำการจดจำทะเบียนรถยนต์แต่ละคันได้ โดยวิธีการรู้จำด้วยตัวเองหรือจดบันทึกเอาไว้ แต่คอมพิวเตอร์ไม่สามารถรู้จำเองได้จะต้องมีการโปรแกรมเข้าไปเพื่อให้คอมพิวเตอร์รู้จำว่ารถยนต์แต่ละคันมีหมายเลขทะเบียนอะไร ซึ่งแนวคิดการรู้จำที่ศึกษามีอยู่ 2 วิธี ได้แก่ Template Matching กับการรู้จำตัวอักษรบนพื้นฐานลักษณะเด่น ด้วยวิธีการ Template Matching จะนำภาพด้านหน้าของรถยนต์มาหาส่วนที่เป็นแผ่นป้ายทะเบียนออกมา จากนั้นก็ทำการแยกตัวพยัญชนะและตัวเลขออกมาเป็นตัวๆ โดยเรียงลำดับให้เหมือนกับแผ่นป้ายทะเบียน แล้วจึงทำการวางทาบกับภาพแม่แบบที่ได้ทำการเก็บไว้ในฐานข้อมูลคอมพิวเตอร์ แล้วผลที่ได้ก็จะทำให้คอมพิวเตอร์สามารถรู้ได้ว่ารถยนต์คันนั้นมีหมายเลขทะเบียนอะไร แต่จะมีปัญหาอยู่ที่ว่าตัวพยัญชนะและตัวเลขของแผ่นป้ายที่จะนำมาวางทาบกับภาพแม่แบบที่เก็บไว้ ถ้าในขั้นตอนการเก็บภาพมีระยะการถ่ายภาพไม่เท่ากัน กล่าวคืออาจจะใกล้ไปหรือไกลไปหรือเอียงซ้ายเอียงขวาก็จะทำให้เมื่อวางทาบกับภาพแม่แบบแล้วผลลัพธ์ที่ได้ก็จะไม่สมบูรณ์หรือไม่ตรงกับตัวพยัญชนะหรือตัวเลขตัวนั้นๆ ดังนั้นจึงได้มีแนวคิดที่จะนำวิธีการรู้จำตัวอักษรบนพื้นฐานลักษณะเด่นมาใช้ เนื่องจากไม่ว่าขนาดของตัวพยัญชนะและตัวเลขบนแผ่นป้ายทะเบียนจะมีขนาดเล็กหรือใหญ่ การรู้จำตัวอักษรบนพื้นฐานลักษณะเด่นก็คาดว่าจะสามารถทำได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.5 ขั้นตอนการศึกษา

1. ศึกษาความเป็นมา ความสำคัญของปัญหา สมมติฐานของการศึกษา ทฤษฎีที่เกี่ยวข้องและแนวคิดในการทำปริญญานิพนธ์

2. ศึกษาการประมวลผลภาพทางดิจิตอล การแทนภาพด้วยข้อมูลแบบดิจิตอล ชนิดของข้อมูลภาพที่ใช้ในการประมวลผลภาพทางดิจิตอล

3. ศึกษาการแบ่งส่วนภาพด้วยวิธีการพิจารณาความต่อเนื่องของข้อมูล และการแยกภาพด้วยวิธี Region Labeling

4. ศึกษาเทคนิคการ Template Matching

5. ศึกษาเทคนิคการรู้จำตัวอักษรบนพื้นฐานลักษณะเด่น (Feature Based Character Recognition)

6. ออกแบบวิธีการหาหัวและใส่หัวให้กับตัวพยัญชนะไทย

7. แสดงผลการทดลองทั้งหมด

8. สรุปผลการทำงาน ปัญหาที่พบและข้อเสนอแนะ

## 1.6 ประโยชน์ที่คาดว่าจะได้รับ

1. นำไปประยุกต์ใช้เพื่อแบ่งเบาภาระมนุษย์ ในการตรวจสอบรถยนต์ที่ผ่านเข้าออกในสถานที่ต่างๆ

2. นำไปประยุกต์ใช้เพื่อช่วยในการคิดคำนวณเงินค่าจอดรถยนต์ตามสถานที่ให้บริการการจอดรถหรือห้างสรรพสินค้าต่างๆ

3. นำไปประยุกต์ใช้เพื่อช่วยตรวจจับรถยนต์ที่ฝ่าสัญญาณไฟจราจรตามสี่แยกต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 2
# ทฤษฎีและหลักการ

ในบทนี้จะกล่าวถึงทฤษฎีต่างๆ ที่ใช้ในขั้นตอนการค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์จากข้อมูลภาพดิจิตอล การแยกตัวเลขและพยัญชนะออกมาจากแผ่นป้ายทะเบียนรถยนต์ซึ่งทั้งสองขั้นตอนนี้ จะใช้ทฤษฎีของการประมวลผลภาพทางดิจิตอล (Digital Image Processing) สำหรับขั้นตอนการรู้จำตัวเลขและตัวพยัญชนะจะใช้เทคนิค Template Matching และการรู้จำตัวอักษรบนพื้นฐานลักษณะเด่น

## 2.1 การประมวลผลภาพดิจิตอล (Digital Image Processing)

ภาพดิจิตอล (Digital Image) หมายถึงภาพที่เก็บอยู่ในรูปแบบดิจิตอล ซึ่งภาพที่เรามองเห็นด้วยสายตาทั่วไปนั้นเป็นภาพในลักษณะสามมิติ คือมีมิติของความกว้าง ความยาว และความลึก ส่วนภาพที่อยู่ในโทรทัศน์หรือคอมพิวเตอร์จะเป็นการแปลงภาพจากสามมิติมาเป็นภาพสองมิติโดยการแปลงสัญญาณไฟฟ้ารูปแบบ Analog เช่นในกล้องวีดีโอ เซ็นเซอร์ที่อยู่ในกล้องจะทำการสแกนวัดผลรวมความเข้มแสงที่จุดต่างๆ ไปตามแนวสแกนที่เรียกว่า Raster Scan ซึ่งการสแกนแบบนี้จะมีทิศทางจากบนลงล่างจากซ้ายไปขวา เช่นเดียวกับเครื่องรับภาพวีดีโอจะรับภาพที่ได้มาจากกล้องวีดีโอและแสดงผลเริ่มต้นจากบนลงล่าง จากซ้ายไปขวาเช่นเดียวกัน

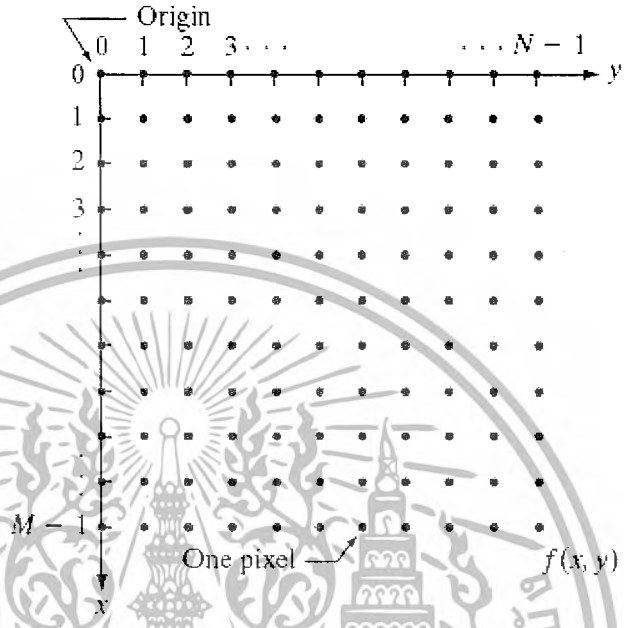
การประมวลผลภาพดิจิตอล (Digital Image Processing) หมายถึงการเรียกใช้ขั้นตอนหรือกรรมวิธีใดๆ มากระทำกับภาพดิจิตอล โดยมีวัตถุประสงค์เพื่อปรับปรุงคุณภาพของภาพเพื่อให้ได้ภาพใหม่ที่มีคุณสมบัติตามที่ต้องการ เช่นมีความคมชัด ประหยัดพื้นที่ในการเก็บข้อมูลหรือใช้สำหรับการประมวลผลระดับสูง เช่นการจดจำรูปร่างลักษณะของสิ่งของ เป็นต้น

### 2.1.1 การแทนภาพด้วยข้อมูลแบบดิจิตอล

ข้อมูลภาพแบบดิจิตอล เป็นภาพที่ถูกดัดแปลงมาจากภาพแบบต่อเนื่อง (Analog Image) ให้อยู่ในรูปตัวเลข (Digital Image) ด้วยวิธีการดิจิไตร์เซชั่น (Digitization) โดยภาพแบบต่อเนื่องจะถูกแบ่งเป็นพื้นที่สี่เหลี่ยมเล็กๆ ที่เรียกว่า พิกเซล (Pixel) ซึ่งในแต่ละพิกเซลจะใช้ (x,y) ซึ่ง x และ y คือ ระบบพิกัดระนาบ (Spatial Coordination) ในการระบุตำแหน่ง การแสดงข้อมูลภาพดิจิตอลสามารถอธิบายได้ในรูปของเมตริกซ์ (Matrix) ขนาด M แถวและ N หลัก เรียกว่าภาพดิจิตอลมีขนาด M*N ดังแสดงในรูปที่ 2.1 ส่วนรูปที่ 2.2 สามารถแทนภาพดิจิตอลด้วยฟังก์ชั่นสองมิติ (Two - Dimensional Function) f(x,y) เมื่อ x และ y คือระบบพิกัดระนาบ (SpatialCoordination) และแอปลิจูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Amplitude) ของ f คือความสว่าง (brightness) หรือความเข้ม (Intensity) หรืออาจถูกเรียกว่าค่าระดับเทา (Monochrome Image)




**รูปที่ 2.1** ข้อมูลภาพดิจิตอลแสดงในรูปของเมตริกซ์สองมิติ

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

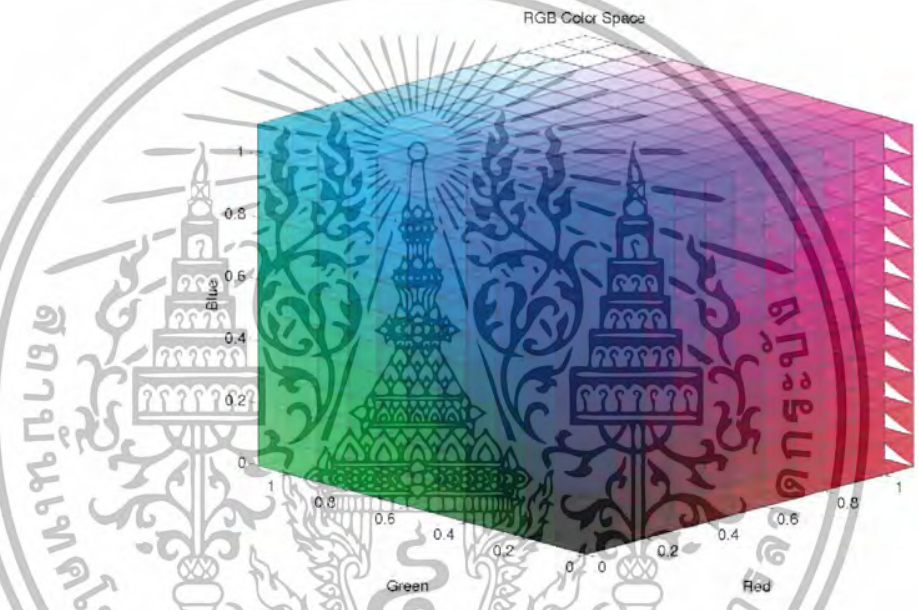
**รูปที่ 2.2** การแทนข้อมูลภาพดิจิตอลด้วยฟังก์ชั่นสองมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1.2 ชนิดของข้อมูลภาพที่ใช้ในการประมวลผลภาพดิจิตอล

#### 2.1.2.1 ภาพสี RGB

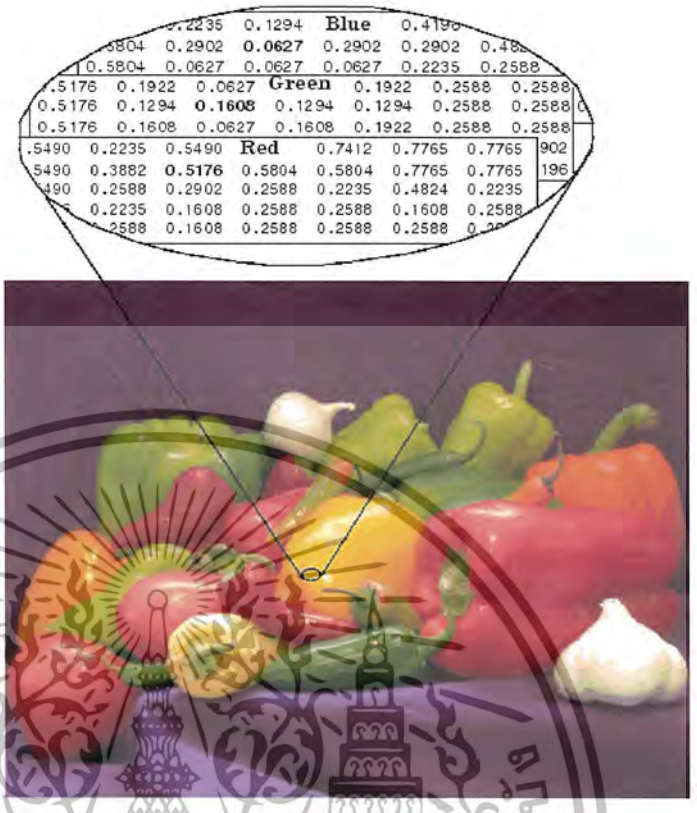
ภาพสี RGB เป็นภาพที่ประกอบไปด้วย 3 แม่สีหลัก ได้แก่ สีแดง สีเขียวและสีน้ำเงิน ถ้านำแต่ละสีมาพล็อตกราฟในระบบพิกัด Color Space ซึ่งเป็นระบบพิกัดแกน x, y และ z ดังแสดงในรูปที่ 2.3 โดยแต่ละสีมีค่าความเข้มแสงแสดงตั้งแต่ 0 ถึง 1 โดย 0 แสดงค่าความมืด และ 1 แสดงค่าความสว่าง ทำให้ได้การผสมสีทางแสงหรือการบวกแม่สีเข้าด้วยกัน ซึ่งจากภาพนี้จะเห็นได้ว่าในแต่ละแกน เมื่อค่าความเข้มแสงมีค่าเข้าใกล้ 1 มากเท่าไหร่ แม่สีนั้นก็จะยิ่งมีความสว่างมากขึ้น




**รูปที่ 2.3** โมเดลสี RGB

ตัวอย่างของภาพสี RGB ในรูปที่ 2.4 ส่วนของวงกลมเล็กๆ ในภาพสี RGB เมื่อถูกขยายออกและแสดงเป็นค่าความเข้มสีของสีแดง สีเขียว และสีน้ำเงิน จะได้ดังรูปวงกลมวงใหญ่ด้านบน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 2.4 ภาพสี RGB

**2.1.2.2 ภาพระดับเทา (Gray Scale Image)**

ภาพระดับเทา หมายถึงภาพที่แต่ละพิกเซล มีค่าความเข้มสีเป็นค่าเฉลี่ยของค่าแม่สีทั้ง 3 ในพิกเซลนั้น คือภาพสีในหนึ่งพิกเซลนั้นจะประกอบไปด้วย 3 แม่สีผสมกัน ได้แก่สีแดง สีเขียว และสีน้ำเงิน ซึ่งมี 3 ค่าใน 1 พิกเซล เมื่อเป็นเช่นนี้จะทำให้ยากต่อการนำไปประมวลผล ดังนั้นก่อนการนำภาพไปประมวลผลจึงต้องทำให้เป็นภาพระดับเทา ซึ่ง Algorithm ที่นิยมใช้ในการทำภาพระดับเทานั้นก็คือ การเฉลี่ยของค่าแม่สีทั้ง 3 ใน 1 พิกเซล เขียนเป็นสมการได้ดังนี้

$$\text{ค่าระดับเทา} = ((0.299 \times R_S) + (0.587 \times G_S) + (0.114 \times B_S)) \qquad (2.1)$$

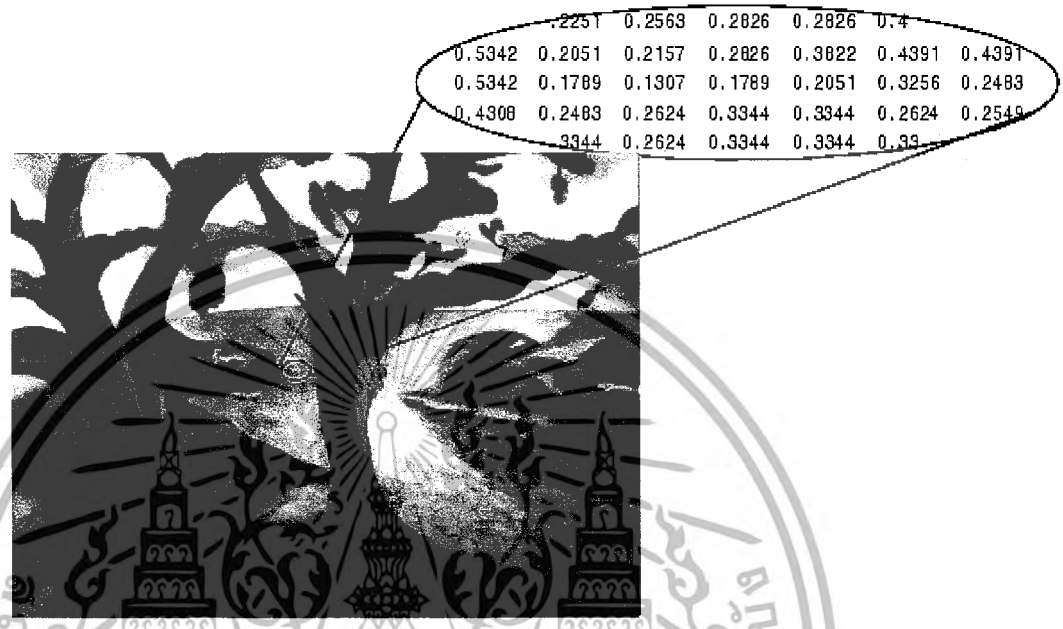
$R_S$ คือ ค่าอินพุตพิกเซลสีแดง

$G_S$ คือ ค่าอินพุตพิกเซลสีเขียว

$B_S$ คือ ค่าอินพุตพิกเซลสีน้ำเงิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการที่ 2.1 จะเห็นได้ว่าค่าระดับเทาสามารถมีค่าได้ตั้งแต่ 0 ถึง 1 ดังแสดงในรูปที่ 2.5 โดยที่ 0 หมายถึงพิกเซลสีดำ และ 1 หมายถึงพิกเซลสีขาว นั่นแสดงว่าถ้าพิกเซลไหนมีค่าระดับเทาเข้าใกล้ 1 มากเท่าไหร่ พิกเซลนั้นก็จะยิ่งมีความสว่างหรือมีสีที่ใกล้เคียงสีขาวมากเท่านั้น




**รูปที่ 2.5** ภาพระดับเทา

**2.1.2.3 ภาพสองระดับ (Binary Image)**

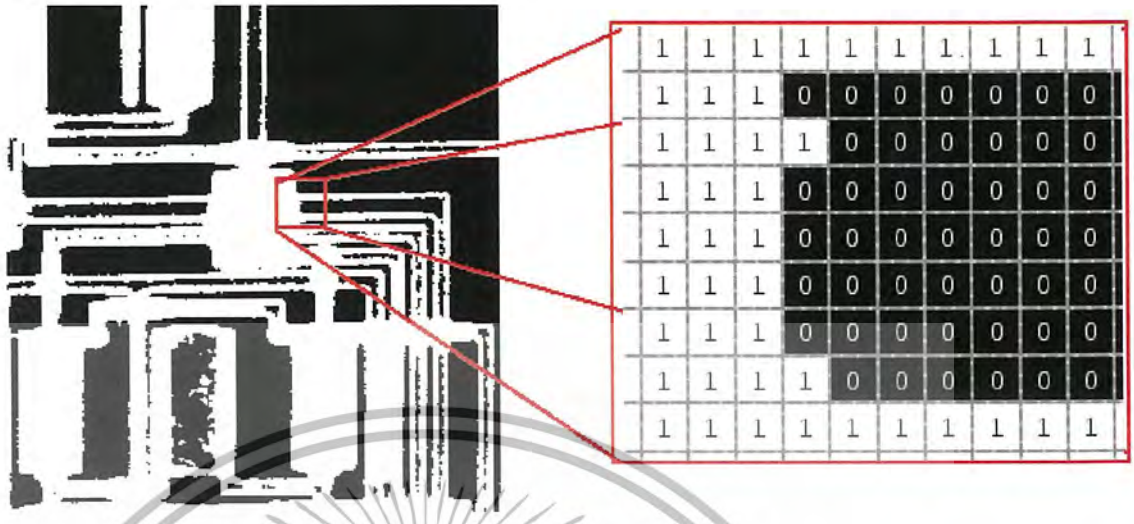
ภาพสองระดับ หมายถึงภาพที่มีระดับความเข้มของสีเพียง 2 ระดับ คือแต่ละพิกเซลสามารถมีค่าได้ 2 ค่าเท่านั้น คือ 1 และ 0 ซึ่ง 1 จะหมายถึงพิกเซลสีขาว ส่วน 0 จะหมายถึงพิกเซลสีดำ ดังแสดงในรูปที่ 2.6 ซึ่งในการประมวลผลภาพทางดิจิตอลโดยทั่วไปนั้น นิยมแปลงข้อมูลภาพหลายระดับ คือภาพระดับเทาและภาพสี RGB ให้เป็นภาพสองระดับ เพื่อเป็นการลดเนื้อที่เก็บข้อมูลภาพให้เหลือเพียง 1 บิต

การสร้างภาพสองระดับสามารถทำได้โดย การใช้เทคนิคการตัดเทรชโฮล (Threshold Techniques) ซึ่งในการพิจารณาว่าพิกเซลไหนของข้อมูลภาพ ควรจะเป็นพิกเซลสีขาวหรือสีดำนั้นจะกระทำโดยการเปรียบเทียบระหว่างพิกเซลนั้นๆ กับค่าคงที่ค่าหนึ่ง หรือค่าเทรชโฮลซึ่งสามารถกำหนดค่าได้ตั้งแต่ 0 ถึง 1 ถ้าหากพิกเซลไหนมีค่าน้อยกว่าค่าเทรชโฮล ก็ให้เปลี่ยนพิกเซลนั้นเป็นพิกเซลที่มีค่าเป็น 0 หรือเป็นพิกเซลสีดำ แต่ถ้าพิกเซลไหนมีค่ามากกว่าค่าเทรชโฮลก็ให้เปลี่ยนพิกเซลนั้นเป็นพิกเซลที่มีค่าเป็น 1 หรือเป็นสีขาว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเลือกค่าเทรชโฮลเพื่อให้ได้ภาพสองระดับนั้น จะต้องเลือกค่าเทรชโฮลให้เหมาะสม ถ้าเลือกค่าไม่เหมาะสม เช่น เลือกค่าเทรชโฮลที่สูงหรือต่ำจนเกินไป รายละเอียดของภาพที่ได้ก็จะยากต่อการนำไปวิเคราะห์ เนื่องจากภาพขาดความคมชัดและอาจจะเกิดสิ่งรบกวน (Noise) ขึ้นในภาพได้ ตัวอย่างเช่นในงานวิจัยนี้ต้องการค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ ดังนั้นในการแปลงภาพถ่ายส่วนด้านหน้ารถยนต์ที่โปรแกรมรับเข้ามาซึ่งเป็นภาพสี RGB ให้เป็นภาพสองระดับนั้น โปรแกรมจะสามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนได้ก็ต่อเมื่อ ส่วนที่เป็นแผ่นป้ายทะเบียนที่อยู่ในภาพสองระดับ ไม่ถูกลบออกไปหรือกลายเป็นพิกเซลสีดำ จากตัวอย่างรูปที่ 2.7 (ก) เมื่อแปลงภาพสี RGB นี้ให้เป็นภาพสองระดับ โดยใช้ค่าเทรชโฮลที่เหมาะสม ภาพผลลัพธ์ที่ได้จะเป็นรูปที่ 2.7 (ข) ซึ่งจะเห็นได้ว่าส่วนที่เป็นแผ่นป้ายทะเบียนไม่ถูกลบออกไปและมีขอบที่ชัดเจน เมื่อนำภาพสองระดับนี้ไปประมวลผลในการค้นหาส่วนที่เป็นแผ่นป้ายทะเบียน โปรแกรมก็จะสามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนจากภาพสองระดับนี้ได้ แต่ในทางตรงกันข้ามถ้านำภาพสี RGB รูปที่ 2.7 (ก) ไปแปลงเป็นภาพ สองระดับโดยใช้ค่าเทรชโฮลที่ไม่เหมาะสม กล่าวคือใช้ค่าเทรชโฮลที่ต่ำเกินไปหรือสูงเกินไปจะทำให้ได้ภาพ Binary ดังรูปที่ 2.8 (ก) และ (ข) ตามลำดับ ซึ่งจากภาพสองระดับทั้งสองนี้ จะเห็นได้ว่าส่วนที่เป็นแผ่นป้ายทะเบียนที่มีขอบเขตไม่ชัดเจนและมีบางส่วนของแผ่นป้ายทะเบียนถูกลบออกไป ซึ่งถ้านำภาพสองระดับทั้งสองนี้ไปประมวลผลในขั้นตอนการค้นหาส่วนที่เป็นแผ่นป้ายทะเบียน โปรแกรมก็อาจจะไม่สามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 2.6 ภาพสองระดับ

(ก) (ข)

รูปที่ 2.7 ภาพสี RGB ที่ถูกแปลงเป็นภาพสองระดับ จากการเลือกค่าเทรชโฮลที่เหมาะสม
(ก) ภาพสี RGB
(ข) ภาพสองระดับ ที่ได้จากการเลือกค่าเทรชโฮลที่เหมาะสม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ก) (ข)

**รูปที่ 2.8** ภาพสี RGB ที่ถูกแปลงเป็นภาพสองระดับ จากการเลือกค่าเทรชโฮลที่ไม่เหมาะสม
(ก) ภาพสี RGB
(ข) ภาพสองระดับ ที่ได้จากการเลือกค่าเทรชโฮลที่ไม่เหมาะสม

## 2.2 การแบ่งส่วนภาพ (Image Segmentation)

กระบวนการสำคัญอีกขั้นตอนหนึ่งในการประมวลผลเบื้องต้น ก่อนจะนำไปสู่ขั้นตอนการรู้จำตัวอักษร ก็คือกระบวนการแยกวัตถุออกจากพื้นหลัง (Segmentation) ในปริญญานิพนธ์ฉบับนี้จะเป็นการแยกส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ออกจากภาพสองระดับและเมื่อสามารถแยกส่วนที่เป็นแผ่นป้ายทะเบียนออกจากภาพสองระดับได้แล้ว ขั้นตอนต่อมาก็จะเป็นการแยกส่วนที่เป็นตัวเลขและตัวพยัญชนะทั้งหมดที่อยู่บนแผ่นป้ายทะเบียนออกมา โดยแยกออกมาทีละตัวเพื่อนำไปสู่กระบวนการรู้จำตัวอักษรต่อไป

### 2.2.1 การแบ่งส่วนภาพด้วยการพิจารณาความต่อเนื่องของข้อมูล

หลักการคือหาพิกเซลที่เป็น 0 หรือพิกเซลสีดำที่ต่อเนื่องกัน ตลอดทั้งแนวตั้งและแนวนอน ซึ่งจะทำให้ได้ขนาดของกรอบ (Block) ของวัตถุที่มีอยู่ในข้อมูลภาพที่มีขนาดแตกต่างกัน จากนั้นจะพิจารณาเลือกขนาดของกรอบที่ต้องการ จากความแตกต่างของจำนวนพิกเซล ความสูง ความกว้าง และตำแหน่ง เป็นต้น อาทิเช่น เลือกกรอบที่มีจำนวนพิกเซลมากที่สุด หรือเลือกกรอบที่มีความสูงมากกว่าครึ่งภาพ เป็นต้น

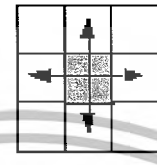
### 2.2.2 การแยกภาพด้วยวิธี Region Labeling

วิธีการแบ่งส่วนภาพแบบ Region Labeling นี้เป็นการกำหนดหมายเลขให้กับกลุ่มพิกเซลสีขาวหรือเรียกว่าทำการ Label ให้กับกลุ่มพิกเซลสีขาว ซึ่งถ้ามีพิกเซลสีขาวอยู่ติดๆ กันเป็นกลุ่มก้อนบริเวณเดียวกัน โดยที่ไม่ถูกแยกออกจากกันด้วยพิกเซลสีดำกลุ่มพิกเซลสีขาวนั้นก็จะมีหมายเลข Label ที่เดียวกัน ส่วนกลุ่มของพิกเซลสีขาวกลุ่มอื่นๆ ก็จะกระทำในลักษณะเดียวกันโดยจะกำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

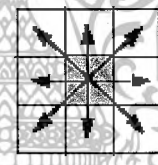
หมายเลข (Label) ที่แตกต่างออกไป ซึ่งการพิจารณาความต่อเนื่องของกลุ่มพิกเซลสีขาวมีหลักการพิจารณาดังต่อไปนี้

การต่อเนื่องกันแบบ 4 จุด จะพิจารณาเฉพาะพิกเซลที่อยู่ล้อมรอบพิเซลตรงกลางทั้งในแนวนอนและแนวตั้งเท่านั้น ดังแสดงในรูปที่ 2.9



**รูปที่ 2.9** การพิจารณาความต่อเนื่องกันแบบ 4 จุด

การต่อเนื่องกันแบบ 8 จุด จะพิจารณา 8 พิกเซลที่อยู่ล้อมรอบพิกเซลตรงกลาง ดังแสดงในรูปที่ 2.10



**รูปที่ 2.10** การพิจารณาความต่อเนื่องกันแบบ 8 จุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.3 เทคนิค Template Matching

Template Matching หรือการเข้าคู่กับภาพแม่แบบ เป็นเทคนิคการหาค่าสัมประสิทธิ์ความเหมือนระหว่างภาพสองภาพว่ามีโอกาสที่จะเป็นภาพเดียวกันหรือไม่ หลักการของ Template Matching จะนำภาพที่ต้องการพิจารณาไปวางทาบกับภาพแม่แบบ และทำการประมวลผลโดยที่ตำแหน่งพิกเซลของภาพที่ต้องการพิจารณา และของภาพแม่แบบจะต้องอยู่ในตำแหน่งเดียวกัน การเข้าคู่กัน (Matching) ระหว่างภาพที่ต้องการพิจารณาและภาพแม่แบบถูกแสดงด้วยค่าสัมประสิทธิ์ความเหมือน (Correlation Coefficient) ซึ่งสามารถเขียนเป็นสมการได้ดังนี้

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

เมื่อ $r$ คือ ค่าสัมประสิทธิ์ความเหมือน (Correlation Coefficient)

A และ B คือ ภาพสองภาพที่มีขนาดเท่ากัน ที่ต้องการตรวจสอบว่าเป็นภาพเดียวกันหรือไม่

$A_{mn}$ และ $B_{mn}$ คือ พิกเซลที่กำลังพิจารณาของแต่ละภาพ

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

ค่าสัมประสิทธิ์ความเหมือน ($r$) มีค่าตั้งแต่ -1 ถึง 1 โดยที่ 1 คือค่าที่แสดงการเท่ากันหรือมีความเหมือนกันที่สุดแต่ถ้าภาพสองภาพมีความแตกต่างกัน ค่าสัมประสิทธิ์ความเหมือนก็จะลดลง กรณีที่ค่าสัมประสิทธิ์ความเหมือนมีค่าเป็น 0 ก็แสดงว่าภาพทั้งสองภาพไม่มีความเหมือนกันเลย

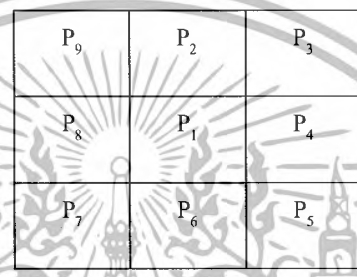
## 2.4 ทฤษฎี The Skeleton of a Region

เป็นการหาเส้นโครงร่างของวัตถุในภาพในที่นี้ ก็คือตัวพยัญชนะและตัวเลข โดยจะทำตัวพยัญชนะและตัวเลขให้บางลงจนเหลือแต่เส้นโครงร่างที่มีความหนาเพียงหนึ่งพิกเซล ดังนั้นจึงเป็นการลดขนาดของข้อมูลส่วนประกอบของตัวพยัญชนะและตัวเลขลงให้เหลือแต่ส่วนที่สำคัญเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในตัวพยัญชนะและตัวเลขนั้นจะประกอบขึ้นด้วยพิกเซลหลายๆ พิกเซลต่อรวมกัน จนเป็นตัวพยัญชนะและตัวเลขขึ้นมา และในแต่พิกเซลจะถูกกำหนดให้มีค่าเป็น 0 กับ 1 โดยพื้นที่ในตัวพยัญชนะและตัวเลขถูกกำหนดให้มีค่าเป็น 1 ส่วนพื้นที่อื่นๆ ที่ไม่ใช่ตัวอักษรจะถูกกำหนดให้มีค่าเท่ากับ 0

วิธีที่จะทำตัวพยัญชนะและตัวเลขให้บางลงนั้น จะทำการลบพิกเซลแต่ละพิกเซลในตัวพยัญชนะและตัวเลขออกไปทีละจุด โดยใช้หน้าต่างขนาด 3x3 วางทาบลงไปบนตัวพยัญชนะและตัวเลขโดยที่จุดกึ่งกลางของหน้าต่างกำหนดให้เป็นพิกเซลแรก ($p_1$) และพิกเซลอื่นๆเป็น $p_1$~$p_9$ ดังแสดงในรูปที่ 2.11 จากนั้นตรวจสอบดูว่าค่าทั้งหมดในหน้าต่างเป็นไปตามเงื่อนไขที่กำหนดไว้ทั้งหมดหรือไม่




**รูปที่ 2.11** หน้าต่างขนาด 3x3 ที่จะใช้ในการลบพิกเซลในตัวพยัญชนะและตัวเลข

สำหรับการลบพิกเซลออกจากตัวพยัญชนะและตัวเลขมี 2 ขั้นตอน

ขั้นตอนที่ 1 ลบด้านขวาและด้านล่างของตัวพยัญชนะและตัวเลข โดยมีเงื่อนไขดังต่อไปนี้

a) $2 \le N(p_1) \le 6$

b) $S(p_1) = 1$

c) $P_2 \cdot P_4 \cdot P_6 = 0$

d) $P_4 \cdot P_6 \cdot P_8 = 0$

ขั้นตอนที่ 2 ลบด้านซ้ายและด้านบนของตัวพยัญชนะและตัวเลข โดยมีเงื่อนไขดังต่อไปนี้

a) $2 \le N(p_1) \le 6$

b) $S(p_1) = 1$

c*) $P_2 \cdot P_4 \cdot P_8 = 0$

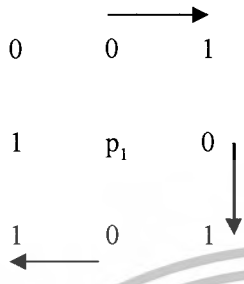
d*) $P_2 \cdot P_6 \cdot P_8 = 0$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เงื่อนไข a) เมื่อ $N(p_1)$ จะเป็นการรวมกันของค่าที่ไม่ใช่ 0 ที่อยู่รอบๆ $p_1$ เช่น $N(p_1) = p_2 + p_3 + \ldots\ldots + p_8 + p_9$

เงื่อนไข b) เมื่อ $S(p_1)$ จะเป็นการตรวจสอบจำนวนครั้งของการเปลี่ยนค่าจาก 0 ไป 1 ในตำแหน่ง $p_2$, $p_3$, ….. , $p_8$, $p_9$

ตัวอย่างเช่น



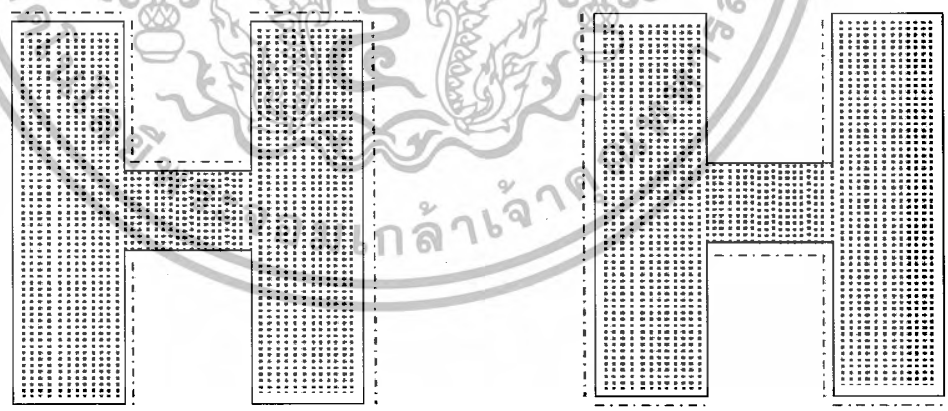

ดังนั้น $S(p_1)$ จึงมีค่าเท่ากับ 3

เงื่อนไข c), d) และ c*), d*)

c) $P_2 \cdot P_4 \cdot P_6 = 0$ c*) $P_2 \cdot P_4 \cdot P_8 = 0$

d) $P_4 \cdot P_6 \cdot P_8 = 0$ d*) $P_2 \cdot P_6 \cdot P_8 = 0$ เมื่อสัญลักษณ์ $\cdot$ แทนการ AND กันทางโลจิก

จุด $P_2$ $P_4$ $P_6$ และ $P_8$ ในทั้งสองเงื่อนไขเหล่านี้เป็นการ AND กันของจุดภาพสามจุดภาพ เพราะฉะนั้น เมื่อมีจุดภาพจุดใดจุดหนึ่งมีค่าเท่ากับ 0 เงื่อนไขทั้งสองข้อนี้ก็จะมีค่าเท่ากับ 0 ด้วย



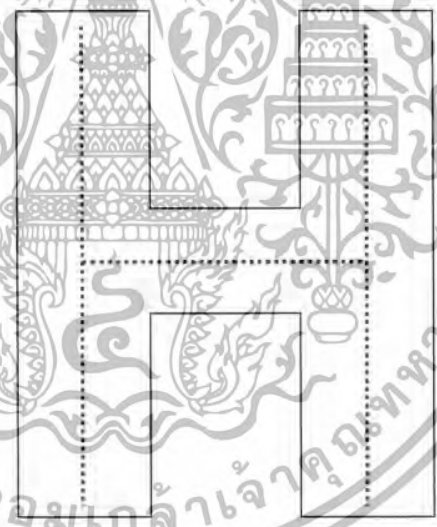
**(ก)** **(ข)**

**รูปที่ 2.12** การลบขอบของตัวพยัญชนะและตัวเลขเพื่อหาเส้นโครงร่างของตัวพยัญชนะและตัวเลข
(ก) การลบพิกเซลตามเงื่อนไขในขั้นตอนที่ 1ซึ่งจะลบพิกเซลในด้านขวามือและด้านบน
(ข) การลบพิกเซลตามเงื่อนไขในขั้นตอนที่ 2 ซึ่งจะลบพิกเซลในด้านซ้ายมือและด้านล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

วิธีที่การหาโครงร่างของตัวพยัญชนะและตัวเลข เป็นการลบพิกเซลออกไปทีละพิกเซลโดยจะใช้หน้าต่างขนาด 3x3 วางทาบลงไปบนตัวพยัญชนะและตัวเลข โดยจุดกึ่งกลางของหน้าต่างจะอยู่บนพิกเซลที่ 2x2 ของตัวพยัญชนะและตัวเลข (จุด $p_1$ จะอยู่บนพิกเซลที่ 2x2) แสดงดังรูปที่ 2.14 จากนั้นตรวจสอบค่าทั้งหมดในหน้าต่างว่าเป็นไปตามเงื่อนไขที่กำหนดไว้หรือไม่ ถ้าเป็นไปตามเงื่อนไขที่กำหนดไว้ทั้งหมด จุด $p_1$ ก็จะถูกลบออกไป แต่ถ้าไม่เข้าเงื่อนไขที่กำหมดไว้ทั้งหมด จุด $p_1$ ก็จะยังไม่ถูกลบออก ผลลัพธ์ที่ได้นั้นจะไม่เกิดขึ้นในรูปที่กำลังพิจารณาอยู่ในทันที แต่จะถูกแสดงอยู่ในรูปถัดไปแทน หลังจากนั้นหน้าต่างก็จะเลื่อนไปที่พิกเซลถัดไป (จุด $p_1$ ก็จะไปอยู่ที่พิกเซลถัดไป) โดยการลบพิกเซลจะทำการลบจากซ้ายไปขวา และจากบนลงล่างของรูป เมื่อทำเสร็จหนึ่งรอบก็จะเริ่มทำรอบใหม่ ในรูปถัดไป รูปที่ 2.12 แสดงการลบขอบของตัวอักษรตามเงื่อนไขในขั้นตอนที่ 1 และ 2 จากนั้นใช้รูปผลลัพธ์ที่ได้จากการทำการลบในรอบแรกจะถูกนำมาใช้เป็นรูปต้นแบบในรอบถัดไป การลบพิกเซลเพื่อหาเส้นโครงร่างนั้นก็จะทำแบบนี้ไปเรื่อยๆ จนเหลือแต่พิกเซลสุดท้ายที่เป็นเส้นโครงร่างของตัวพยัญชนะและตัวเลขเท่านั้นดังแสดงรูปที่ 2.13

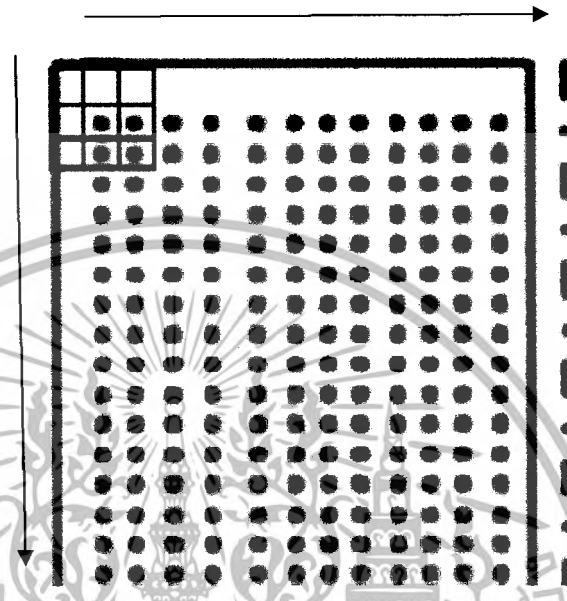


**รูปที่ 2.13** ผลลัพธ์ที่ได้จากการลบพิกเซลทั้งหมดจนเหลือพิกเซลโครงร่างของตัวพยัญชนะและตัวเลข

110820

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากตัวอักษร H ไม่ใช่ตัวพยัญชนะภาษาไทย แต่เป็นการตัวอย่างการหาเส้นโครงร่างของตัวเลขและตัวพยัญชนะเท่านั้น สำหรับตัวพยัญชนะและตัวเลขในปริญญานิพนธ์นี้การหาเส้นโครงร่างก็จะใช้หลักการเดียวกัน



**รูปที่ 2.14** ภาพการวางหน้าต่างขนาด 3X3 เพื่อทำการลบพิกเซลตามเงื่อนไขที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 3
# การรู้จำอักษรด้วยวิธี Template Matching

ในบทนี้จะอธิบายถึงวิธีการและขั้นตอนการตรวจหาแผ่นป้ายทะเบียนรถยนต์ ที่เขียนขึ้นด้วยโปรแกรม MATLAB ซึ่งจะประกอบด้วยขั้นตอนต่างๆ ที่เริ่มต้นจากการโหลดข้อมูลภาพถ่ายด้านหน้าของรถยนต์ที่มีแผ่นป้ายทะเบียนรวมอยู่ด้วยไปจนถึง การนำตัวพยัญชนะและตัวเลขที่อยู่บนแผ่นป้ายทะเบียนรถยนต์ไปประมวลผล เพื่อการรู้จำว่าในแผ่นป้ายทะเบียนนั้นมีตัวพยัญชนะและตัวเลขอะไรบ้าง วิธีการตรวจหาแผ่นป้ายทะเบียนมีโครงสร้างการทำงานสามขั้นตอนหลัก ดังนี้

1. การประมวลผลภาพดิจิตอลที่รวมการโหลดข้อมูลภาพ
2. การแยกส่วนที่เป็นตัวพยัญชนะและตัวเลขออกมาจากแผ่นป้ายทะเบียน
3. การรู้จำตัวพยัญชนะและตัวเลขที่อยู่บนแผ่นป้ายทะเบียนรถยนต์

เหตุผลที่งานวิจัยนี้เลือกใช้โปรแกรม MATLAB ก็เพราะว่าในโปรแกรม MATLAB มี Tool คำสั่งจำนวนมากที่สามารถนำมาใช้งานได้ ทำให้ลดเวลาในการเขียนโปรแกรมไม่เหมือนโปรแกรม Visual Basic (VB) หรือโปรแกรมภาษาซีที่จะต้องเขียนคำสั่งทุกคำสั่งขึ้นเอง ดังนั้นการใช้โปรแกรม MATLAB จะทำให้การเขียนคำสั่งเป็นไปได้อย่างสะดวกและรวดเร็ว และที่สำคัญยังมีคำสั่งที่ใช้เชื่อมต่อโปรแกรม MATLAB เข้ากับกล้องได้อีกด้วย

## 3.1 การประมวลผลภาพดิจิตอล

1. ถ่ายภาพด้านหน้ารถยนต์ด้วยกล้องดิจิตอล Panasonic ที่มีความละเอียด 10 ล้านพิกเซล เวลาที่ใช้ในการถ่ายภาพด้านหน้ารถยนต์แต่ละคันนั้นใช้ช่วงเวลาในการถ่ายภาพที่ไม่เท่ากัน เช่นเช้า กลางวัน หรือตอนเย็น ซึ่งการถ่ายภาพที่ด้านหน้าของรถยนต์นั้นได้มีการกำหนดระยะในการถ่ายภาพให้เท่ากันทุกๆ คัน โดยมีการกำหนดระยะการถ่ายไว้ประมาณ 1 เมตร จากด้านหน้ารถยนต์และได้ตั้งกล้องให้ตรงเพื่อให้ได้ภาพที่มีลักษณะเป็นภาพหน้าตรง อย่างไรก็ตามในทางปฏิบัติระยะในการถ่ายภาพและการตั้งกล้องแต่ละครั้งอาจจะเคลื่อนไปจากค่าที่กำหนดไว้ เพียงแต่ให้มีค่าใกล้เคียงกับที่กำหนดเท่านั้น การถ่ายภาพด้านหน้ารถยนต์จะเก็บภาพไว้ให้ได้จำนวนหนึ่ง จากนั้นจึงทำการเชื่อมต่อกล้องเข้ากับคอมพิวเตอร์ เพื่อที่จะทำการถ่ายโอนไฟล์ข้อมูลภาพทั้งหมดไว้ในฐานข้อมูลของคอมพิวเตอร์เพื่อการเรียกใช้ในขั้นตอนต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. โหลดข้อมูลภาพถ่ายด้านหน้ารถยนต์ที่เก็บไว้ในฐานข้อมูลเข้ามา ซึ่งภาพที่ได้นั้นจะเป็นภาพสี RGB โดยการใช้คำสั่ง imread ซึ่งเป็น Tool คำสั่งใน MATLAB ในการโหลดไฟล์ภาพขึ้นมา จะมีนามสกุลของไฟล์ภาพที่ใช้เป็น JPEG, TIF หรือนามสกุลอื่นๆ ก็ได้ แต่ในโปรแกรมเขียนไว้กำหนดให้นามสกุลของไฟล์ภาพเป็น JPEG รูปที่ 3.1 แสดงตัวอย่างภาพถ่ายด้านหน้ารถยนต์ที่มีแผ่นป้ายทะเบียนรวมอยู่ด้วย

**รูปที่ 3.1** ภาพสี RGB บริเวณด้านหน้ารถยนต์ที่มีแผ่นป้ายทะเบียนรวมอยู่ด้วย

3. แปลงภาพสี RGB ในขั้นตอนที่ 2 ให้เป็นภาพระดับเทา (Gray Scale Image) โดยใช้คำสั่ง rgb2gray ซึ่งเป็น Tool คำสั่งในโปรแกรม MATLAB ทำให้ภาพผลลัพธ์ที่ได้เป็นภาพระดับเทา ซึ่งในการทำภาพระดับเทานี้ ก็คือการเฉลี่ยกันของค่าแม่สีทั้ง 3 สีใน 1 พิกเซล ได้แก่ สีแดง สีเขียว และสีน้ำเงิน ซึ่งในการนำภาพไปประมวลจะต้องทำให้เป็นภาพระดับเทาก่อนจะทำให้ง่ายต่อการประมวลผลในขั้นตอนต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

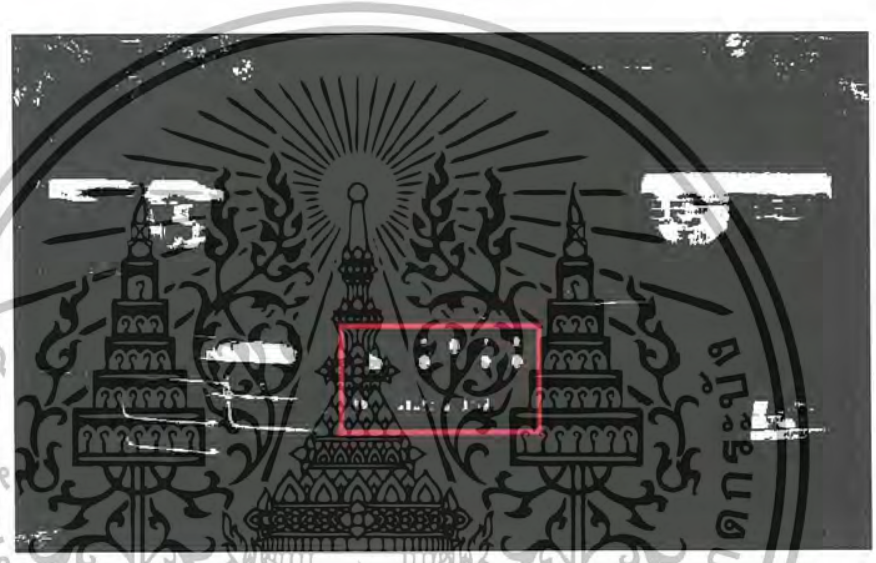
**รูปที่ 3.2** ภาพระดับเทาที่แปลงมาจากภาพสี RGB

4. แปลงภาพระดับเทาในขั้นตอนที่ 3 ให้เป็นภาพสองระดับ โดยใช้คำสั่ง im2bw ซึ่งเป็น Tool คำสั่งในโปรแกรม MATLAB ภาพที่ได้ออกมานั้นจะกลายเป็นภาพสองระดับ คือจะมีความเข้มของสีในภาพเพียง 2 ระดับเท่านั้นคือสีขาวกับสีดำ ซึ่งค่า 1 แทนพิกเซลสีขาวและค่า 0 แทนพิกเซลสีดำ การแปลงภาพระดับเทาให้เป็นภาพแบบสองระดับก็เพื่อช่วยลดเนื้อที่ในการเก็บข้อมูลภาพให้เหลือเพียง 1 บิตเท่านั้น และช่วยให้การประมวลผลเร็วขึ้น ดังรูปที่ 3.3

**รูปที่ 3.3** ภาพสองระดับที่ได้จากการแปลงภาพระดับเทาด้วยคำสั่ง im2bw

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ลบพื้นที่หรือกลุ่มพิกเซลสีขาวที่อยู่ติดกับขอบของภาพออกโดยใช้คำสั่ง imclearborder ซึ่งเป็น Tool คำสั่งในโปรแกรม MATLAB เพื่อตัดพื้นที่ส่วนที่คาดว่าไม่ใช่แผ่นป้ายทะเบียนออก และให้เหลือเฉพาะพื้นที่ส่วนที่คาดว่าเป็นแผ่นป้ายทะเบียนเท่านั้น จากการใช้คำสั่ง imclearborder กับภาพสองระดับในรูปที่ 3.3 จะได้ผลลัพธ์ดังรูปที่ 3.4 ซึ่งจะเห็นได้ว่าพื้นที่หรือกลุ่มพิกเซลสีขาวที่อยู่ติดกับขอบของภาพจะถูกกำจัดออกไป นอกจากนั้นยังพบว่ากลุ่มพิเซลสีขาวส่วนที่เป็นแผ่นป้ายทะเบียนถูกกำจัดออกไปด้วย เพราะว่ากลุ่มพิเซลสีขาวของแผ่นป้ายทะเบียนอยู่ติดกับขอบภาพ ในรูปที่ 3.4 ภาพของแผ่นป้ายทะเบียนที่ถูกกำจัดออกไปถูกตีกรอบด้วยเส้นสีแดง

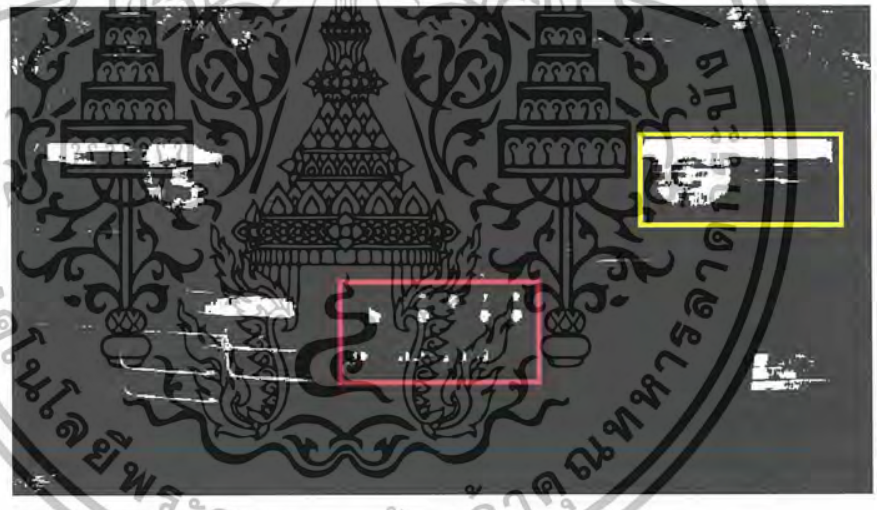


**รูปที่ 3.4** ภาพสองระดับภายหลังจากกำจัดกลุ่มพิกเซลสีขาวที่อยู่ติดกับขอบของภาพ

จากรูปที่ 3.4 เป็นตัวอย่างภาพสองระดับที่ได้จากการตัดค่า Threshold ไม่เหมาะสม เนื่องจากภาพที่ถ่ายอาจจะเกิดแสงตกกระทบลงบนวัตถุมากหรือน้อยจนเกินไป ดังนั้นเมื่อทำการแปลงภาพจากระดับเทาเป็นภาพสองระดับ โดยการตัดระดับเทรชโฮลแบบอัตโนมัติ ด้วย Tool คำสั่งในโปรแกรม MATLAB ทำให้ส่วนที่เป็นแผ่นป้ายทะเบียนเป็นกลุ่มพิกเซลสีขาวที่ติดอยู่กับขอบภาพไปด้วยและเมื่อทำการกำจัดกลุ่มพิกเซลสีขาวที่อยู่ติดกับขอบของภาพออกไป ด้วยคำสั่ง imclearborder จึงทำให้กลุ่มพิกเซลสีขาวที่ติดอยู่กับขอบภาพถูกกำจัดออกไปและถูกแทนที่ด้วยพิกเซลสีดำ ซึ่งถ้าจะนำภาพนี้ไปประมวลผลในขั้นตอนต่อไป จะทำให้ไม่สามารถทำการค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ได้ แต่จะได้ส่วนที่เป็นไฟด้านหน้าของรถยนต์มาแทนดังรูปที่ 3.5 (ก)-(ข) สาเหตุที่ได้ภาพส่วนที่เป็นไฟหน้ารถยนต์ เนื่องจากโปรแกรมการตัดค่าเทรชโฮลอัตโนมัติทำให้ได้ค่าเทรชโฮลที่มีค่าไม่เหมาะสมและเมื่อทำการ imclearborder เพื่อลบกลุ่มพิกเซลสีขาวที่ติดอยู่กับขอบภาพออกไป จะเห็นได้ว่าจำนวนกลุ่มพิกเซลสีขาวที่มากที่สุดไม่ใช่ส่วนที่เป็นแผ่นป้ายทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถยนต์ (กรอบสีขาว) แต่จะได้จำนวนกลุ่มพิกเซลสีขาวที่มากที่สุดอยู่ในส่วนของไฟด้านหน้ารถยนต์ (กรอบสีเหลือง) สำหรับวิธีการแก้ไขสามารถทำได้ด้วยการปรับระดับความสว่างของภาพเอง (ปรับระดับเทรชโฮลโดย manual) โดยปรกติจะทำการโปรแกรมโดยการปรับค่าเทรชโฮลอัตโนมัติ ซึ่งในการใช้วิธีการปรับค่าเทรชโฮลอัตโนมัติ ไม่สามารถทำการโปรแกรมเพื่อค้นหาส่วนที่เป็นภาพแผ่นป้ายทะเบียนได้ครบทุกภาพ ดังนั้นในส่วนของภาพที่ไม่สามารถโปรแกรมการค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ได้ จะต้องทำการแก้ไขด้วยวิธีปรับระดับค่าความสว่างของภาพเอง (ปรับระดับเทรชโฮลแบบ manual) ด้วยคำสั่ง level = level + 0.25 เมื่อ level คือค่าเทรชโฮลที่ได้จาก Tool คำสั่งในโปรแกรม MATLAB และค่า 0.25 เป็นค่าที่กำหนดขึ้นเอง เพื่อให้ภาพผลลัพธ์มีความสว่างมากขึ้นหรือน้อยลงตามที่ต้องการ และสามารถกำหนดค่าเป็นบวกหรือลบก็ได้แต่จะต้องไม่เกินบวกหรือลบ 1 ซึ่งเมื่อทำการปรับค่าเทรชโฮลจนเหมาะสมแล้ว ผลลัทธ์ที่ได้จะเป็นดังรูปที่ 3.5 (ค) จะเห็นได้ว่าในภาพมีส่วนที่เป็นแผ่นป้ายทะเบียนหรือกลุ่มพิเซลสีขาวรวมอยู่ด้วยไม่ถูกกำจัดออกไป



(ก) ภาพตัวอย่างที่ใช้คำสั่ง imclearborder เพื่อที่จะกำจัดกลุ่มพิกเซลสีขาวที่ติดกับขอบภาพออกไป และเมื่อผ่านขั้นตอนนี้ไปแล้วโปรแกรมได้พิจารณาว่าไฟด้านหน้ารถยนต์คือบริเวณที่เป็นแผ่นป้ายทะเบียน และตีกรอบด้วยเส้นสีเหลือง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

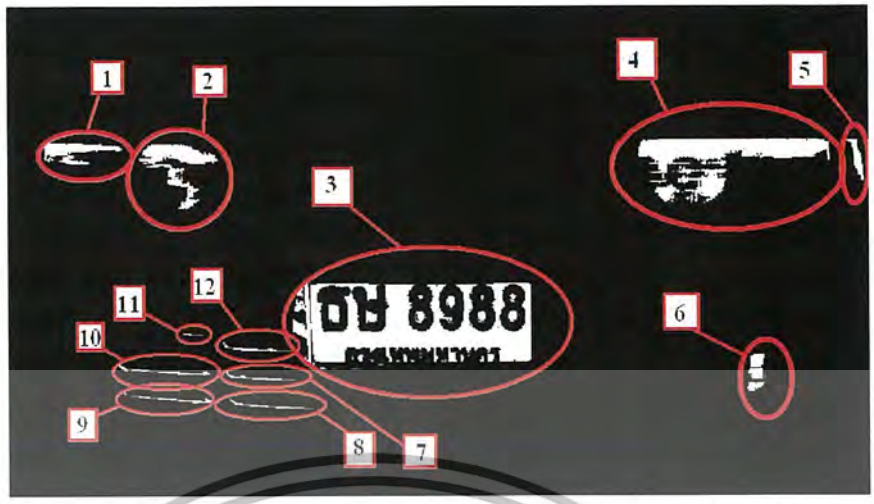
**(ข)** ภาพไฟหน้ารถยนต์ที่โปรแกรมพิจารณาว่าเป็นแผ่นป้ายทะเบียน

**(ค)** ภาพผลลัพธ์เมื่อทำการปรับระดับเทรชโฮลใหม่ ทำให้โปรแกรมตรวจหาแผ่นป้ายทะเบียนได้

**รูปที่ 3.5** ภาพตัวอย่างปัญหาที่เกิดจากการใช้ค่า Threshold ที่ไม่เหมาะสม และแก้ไขโดยการปรับระดับเทรชโฮลด้วยมือ

6. เมื่อปรับระดับเทรชโฮลที่เหมาะสม ส่วนหนึ่งทำให้สามารถกำจัดส่วนที่คาดว่าไม่ใช่แผ่นป้ายทะเบียนออกไปได้และได้กลุ่มพิกเซลสีขาวที่คาดว่าน่าจะเป็นแผ่นป้ายทะเบียน ดังแสดงในรูปที่ 3.5 (ค) จากนั้นกำหนดหมายเลขให้กับกลุ่มพิกเซลสีขาวแต่ละกลุ่มหรือเรียกว่าการทำ Label ให้กับกลุ่มพิกเซลสีขาว ซึ่งถ้าพิกเซลสีขาวอยู่ติดกันเป็นกลุ่มก้อนบริเวณเดียวกันโดยไม่ถูกแยกออกจากกันด้วยพิกเซลสีดำ กลุ่มพิกเซลสีขาวนั้นก็จะมีหมายเลข (Label) เดียวกัน ดังรูปที่ 3.6 (ก) จากรูปจะสังเกตุเห็นได้ว่า มีกลุ่มพิกเซลสีขาวอยู่ทั้งหมด 12 กลุ่มและกลุ่มพิกเซลสีขาวที่ใหญ่ที่สุดคือกลุ่มที่ 3 ซึ่งเป็นส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ และเมื่อโปรแกรมเลือกพื้นที่หรือกลุ่มพิกเซลสีขาวที่ใหญ่ที่สุดและกำหนดให้เป็นแผ่นป้ายทะเบียนรถยนต์ก็จะได้ผลลัพธ์ ดังรูปที่ 3.6 (ข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




(ก) การกำหนดหมายเลขกลุ่มให้กับกลุ่มพิกเซลสีขาวแต่ละกลุ่ม

(ข) ภาพผลลัพธ์ที่โปรแกรมเลือกพื้นที่หรือกลุ่มพิกเซลสีขาวที่ใหญ่ที่สุดเป็นแผ่นป้ายทะเบียนรถยนต์

**รูปที่ 3.6** การกำหนดหมายเลขกลุ่มพิกเซลสีขาวและให้กลุ่มพิกเซลสีขาว
ที่ใหญ่ที่สุดเป็นแผ่นป้ายทะเบียนรถยนต์

7. เมื่อค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ได้แล้ว ให้นำภาพแผ่นป้ายทะเบียนรถยนต์ที่ได้มาจากขั้นตอนที่ 6 มาปรับระดับเทรชโฮลอัตโนมัติ ที่ต้องทำเช่นนี้ก็เพื่อที่จะช่วยให้ตัวเลขและตัวพยัญชนะบนแผ่นป้ายทะเบียนมีความคมชัดมากขึ้น และจากนั้นให้ทำการกลับ (Inverse) ข้อมูลภาพสองระดับ ผลที่ได้คือส่วนที่เป็นตัวเลขและตัวพยัญชนะจะกลายเป็นพิกเซลสีขาว ส่วนพื้นหลังของแผ่นป้ายทะเบียนจะกลายเป็นสีดำแทน ดังรูปที่ 3.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**รูปที่ 3.7** ภาพสองระดับที่ถูกกลับจากพิกเซลสีขาวเป็นพิกเซลสีดำ และพิกเซลสีดำเป็นพิกเซลสีขาว ของภาพที่ 3.6 (ข)

8. ทำการกำหนดหมายเลข (Label) ข้อมูลภาพที่ได้จากขั้นตอนที่ 7 เพื่อสำรวจกลุ่มพิกเซลสีขาวทุกกลุ่มที่มีอยู่บนภาพสองระดับ และหาขอบเขตด้านบนสุด ด้านล่างสุด ซ้ายสุด และขวาสุดของแต่ละกลุ่มพิกเซลสีขาว จากนั้นนำค่าขอบเขตด้านบนสุดลบด้วยขอบเขตด้านล่างสุดจะทำให้ทราบความสูงของกลุ่มพิกเซลสีขาว ในทำนองเดียวกันนำค่าขอบเขตซ้ายสุดลบด้วยขอบเขตขวาสุดจะทำให้ทราบความกว้างของกลุ่มพิกเซลสีขาว อย่างไรก็ตามถ้ากลุ่มพิกเซลสีขาวที่มีความสูงต่ำกว่า 0.25 เท่าของความสูงของภาพสองระดับที่เป็นแผ่นป้ายทะเบียนจะถูกกำจัดออกไป ซึ่งก็คือส่วนที่เป็นชื่อจังหวัดเพราะชื่อจังหวัดมีตัวอักษรขนาดเล็กยากต่อการรู้จำ โดยการเปลี่ยนกลุ่มพิกเซลสีขาวที่เข้าเงื่อนไขให้เป็นพิกเซลสีดำทั้งหมด

**(ก)** ส่วนสูงในส่วนที่เป็นชื่อจังหวัดของแผ่นป้ายทะเบียนรถยนต์ซึ่งมีความสูงน้อยกว่า 0.25 เท่าของความสูงของภาพสองระดับที่เป็นแผ่นป้ายทะเบียนรถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**(ข)** การกำจัดส่วนที่เป็นชื่อจังหวัดออกไปจากรูป (ก)

**รูปที่ 3.8** การกำหนดหมายเลขและกำจัดส่วนที่เป็นชื่อจังหวัดออกไป

9. กำจัดกลุ่มพิกเซลสีขาวที่คาดว่าน่าจะเป็นสิ่งรบกวนออก ซึ่งในบางกรณีอาจจะมีสิ่งรบกวนติดอยู่ในภาพของแผ่นป้ายทะเบียน ดังนั้นจึงควรกำจัดกลุ่มพิกเซลสีขาวที่เป็นสิ่งรบกวนออก เช่น ลวดลายบนแผ่นป้าย สิ่งสกปรกบนแผ่นป้ายทะเบียน และกลุ่มพิกเซลสีขาวที่ติดกับขอบภาพด้วย โดยใช้คำสั่ง medianfilter ซึ่งเป็น Tool ในโปรแกรม MATLAB ขนาดหน้าต่างที่ใช้มีขนาด 3x3 จนถึง9x9 ขึ้นอยู่กับว่าภาพที่ใช้มีสิ่งรบกวนมากน้อยเพียงใด เนื่องจากในการเก็บภาพสี RGB ด้านหน้ารถยนต์นั้น การตั้งกล้องอาจจะเอียงหรือบนแผ่นป้ายทะเบียนมีสิ่งสกปรกติดอยู่จึงทำให้ภาพที่ได้เมื่อแปลงเป็นภาพสองระดับแล้ว จะทำให้ได้พิกเซลสีขาวจำนวนมากจนทำให้เห็นตัวเลขหรือพยัญชนะบนแผ่นป้ายทะเบียนไม่ชัดเจน ซึ่งถ้ากำจัดสิ่งรบกวนนี้ออกไปได้ก็จะทำให้ตัวเลขและตัวพยัญชนะมีความแตกต่างจากพื้นหลังของแผ่นป้ายมากยิ่งขึ้น ดังรูปที่ 3.9

**รูปที่ 3.9** กำจัดกลุ่มพิกเซลสีขาวที่ติดขอบและสิ่งรบกวนต่างๆ ที่อยู่บนแผ่นป้ายทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2 การแยกส่วนที่เป็นตัวเลขและตัวพยัญชนะออกมาจากแผ่นป้ายทะเบียนรถยนต์

เมื่อได้ส่วนที่เป็นแผ่นป้ายทะเบียนออกมาแล้ว ในส่วนของรุ่นพี่ต้องทำการหมุนภาพของแผ่นป้ายทะเบียนให้มีทิศทางที่อยู่ในแนวระนาบ ซึ่งแผ่นป้ายทะเบียนอาจจะมีบางส่วนที่เอียงเป็นผลอันเนื่องมาจากการตั้งกล้องถ่ายภาพ ซึ่งในการทำผลการทดลองนี้ได้ทำการปรับปรุงให้มีการจำกัดระยะการถ่ายภาพไว้คงที่และการตั้งกล้องถ่ายภาพไว้ให้ตรง ส่วนขั้นตอนต่อไปก็จะเป็นการแยกตัวเลขและตัวพยัญชนะทั้งหมดที่อยู่บนแผ่นป้ายออกมาทีละตัว โดยทำการเรียงลำดับให้เหมือนกับบนแผ่นป้ายทะเบียน มีขั้นตอนดังต่อไปนี้

**ขั้นตอนที่ 1** หาแถว (Row) อ้างอิงเพื่อที่จะทำการตัดขอบเขตด้านบนสุด และขอบเขตด้านล่างสุดของตัวเลขและตัวพยัญชนะทั้งหมดออกมา โดยมีหลักการที่ว่าถ้าแถวที่กำลังพิจารณาอยู่นั้นมีจำนวนพิกเซลทุกพิกเซลเป็นสีดำ (พิกเซลที่มีค่าเป็น 0 ทั้งหมด) และแถวถัดไปมีพิกเซลที่เป็นพิกเซลสีขาวทั้งหมด ถ้าเข้าเงื่อนไขนี้ให้บันทึกแถวที่มีพิกเซลสีดำทั้งหมดที่ทำการสำรวจไว้เป็นขอบเขตด้านบนสุด หรือแกนอ้างอิงในการตัดขอบเขตด้านบนสุดของกลุ่มตัวเลขและตัวพยัญชนะบนแผ่นป้ายทะเบียน ส่วนขอบเขตด้านล่างจะพิจารณาไปทางตรงกันข้ามกับขอบเขตด้านบน กล่าวคือ ถ้าแถวที่กำลังพิจารณาอยู่มีจำนวนพิกเซลทุกพิกเซลเป็นสีขาวทั้งหมด และแถวถัดไปมีจำนวนพิกเซลทุกพิกเซลเป็นสีดำทั้งหมด ถ้าเข้าเงื่อนไขนี้ก็ให้บันทึกแถวที่มีพิกเซลสีดำทั้งหมดที่ทำการสำรวจไว้เป็นขอบเขตด้านล่างสุด หรือแกนอ้างอิงในการตัดขอบเขตด้านล่างสุดของตัวเลขและตัวพยัญชนะบนแผ่นป้ายทะเบียน ดังรูปที่ 3.10 (เส้นทึบสีขาวแสดงแกนอ้างอิงในการตัดขอบเขตด้านบนและขอบเขตด้านล่าง) เมื่อได้แกนอ้างอิงในการตัดขอบเขตด้านบนสุดและขอบเขตด้านล่างสุด ของกลุ่มตัวเลขและตัวพยัญชนะบนแผ่นป้ายทะเบียนแล้วก็นำแกนอ้างอิงทั้งสองนี้ไปตัดข้อมูลภาพที่ได้จากรูปที่ 3.10 จะได้ผลลัพธ์ดังรูปที่ 3.11

**รูปที่ 3.10** แถว (row) ที่จะใช้เป็นแกนอ้างอิงในการตัดขอบเขตด้านบนสุด และขอบเขตด้านล่างสุด (แสดงด้วยเส้นทึบสีขาว) ของตัวเลขและตัวพยัญชนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**รูปที่ 3.11** ภาพที่ได้หลังจากการตัดขอบเขตด้านบนสุดและขอบเขตด้านล่างสุดแล้ว

**ขั้นตอนที่ 2** เมื่อทำการหาแกนอ้างอิงในการตัดขอบเขตด้านบนสุด และขอบเขตด้านล่างสุดจากขั้นตอนที่ 1 จากนั้นหาขอบเขตด้านบนซ้ายสุดและขวาสุดของแต่ละกลุ่มพิกเซลสีขาวที่ได้กำหนดหมายเลขไว้แล้ว เพื่อที่จะแยกตัวเลขและตัวพยัญชนะที่อยู่บนแผ่นป้ายทะเบียนออกมาทีละตัว โดยเรียงลำดับให้เหมือนแผ่นป้ายทะเบียน ดังแสดงในรูปที่ 3.12 ซึ่งจะเห็นขอบเขตด้านซ้ายสุดและขอบเขตด้านขวาสุด (แสดงด้วยเส้นทึบสีขาว) ของตัวเลขและตัวพยัญชนะแต่ละตัว ซึ่งตัวแปร T1 และตัวแปร U1 จะเก็บขอบเขตด้านซ้ายสุดและด้านขวาสุดของแต่ละกลุ่มพิกเซลสีขาว ตามลำดับแสดงดังรูปที่ 3.13

**รูปที่ 3.12** ขอบเขตด้านซ้ายสุดและขอบเขตด้านขวาสุด (แสดงด้วยเส้นทึบสีขาว) ของตัวเลขและตัวพยัญชนะของกลุ่มพิกเซลสีขาวที่มีหมายเลขที่ต่างกัน

```
T1  =

     9    52   121   159   198   237

U1  =

    45    91   150   188   227   266
```

**รูปที่ 3.13** ผลที่ได้จากการนำ Column ที่เป็นขอบซ้ายและชอบขวาของตัวเลขและตัวพยัญชนะไปเก็บไว้ในรูปของตัวแปร T1 และ U1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ขั้นตอนที่ 3** เมื่อได้หาขอบเขตด้านซ้ายสุดและด้านขวาสุดแล้ว ทำให้สามารถจากนั้นแยกตัวเลขและตัวพยัญชนะออกมาทีละตัวโดยเรียงลำดับให้เหมือนบนแผ่นป้ายทะเบียน จากนั้นทำการกลับให้เป็นภาพสองระดับเพื่อให้ส่วนที่เป็นตัวเลข และตัวพยัญชนะเป็นสีดำและมีพื้นหลังที่เป็นสีขาว ดังรูปที่ 3.14

Letter 1 letter 2 letter 3 letter 4 letter 5 letter 6

**รูปที่ 3.14** ตัวเลขและตัวพยัญชนะที่ถูกแยกออกมาจากแผ่นป้ายทะเบียน โดยทำการเรียงลำดับให้เหมือนกับแผ่นป้ายทะเบียนรถยนต์

### 3.3 การรู้จำตัวเลขและตัวพยัญชนะ

เมื่อแยกภาพส่วนที่เป็นตัวเลข และตัวพยัญชนะออกมาจากภาพแผ่นป้ายทะเบียนรถยนต์ได้ครบทุกตัวโดยเรียงลำดับให้เหมือนกับแผ่นป้ายทะเบียนแล้ว จากนั้นให้นำภาพของตัวเลขและตัวพยัญชนะทั้งหมดมาเข้าสู่การรู้จำตัวเลขและตัวพยัญชนะ โดยการใช้เทคนิคที่เรียกว่า Template Matching ซึ่งก็คือ การนำตัวเลขและตัวพยัญชนะมาทำการหาค่าสัมประสิทธิ์ความเหมือน (Correlation Coefficient) กับตัวเลขและตัวพยัญชนะทุกตัวที่มีอยู่ในชุดภาพแม่แบบ ถ้าค่าสัมประสิทธิ์ความเหมือนที่เกิดจากการเปรียบเทียบกับตัวเลขและตัวพยัญชนะกับภาพชุดแม่แบบตัวใดมีค่ามากที่สุดหรือมีค่าใกล้เคียง 1 หรือ 100% ให้แสดงตัวเลขหรือตัวพยัญชนะนั้นออกมาและผลที่ได้ออกมาให้เรียงลำดับตามแผ่นป้ายทะเบียนรถยนต์

ในการจัดทำชุดแม่แบบ (Template) ทั้งตัวเลขและตัวพยัญชนะได้ทำการปรับปรุงและแก้ไขชุดภาพแม่แบบขึ้นมาใหม่ โดยการจัดทำภาพชุดแม่แบบของรุ่นพี่ได้กำหนดค่าความสูงของภาพตัวเลขและตัวพยัญชนะไว้ที่ 100 พิกเซล และกำหนดค่าความกว้างของภาพตัวเลขและตัวพยัญชนะไว้หลายขนาด เช่น ภาพตัวพยัญชนะ ก ไก่ มีค่าขนาดความกว้างทั้งหมด 3 ขนาด คือ มีค่าความกว้างที่ 78 ,86 , 92 พิกเซล ดังนั้นจึงได้ทำการปรับปรุงและแก้ไขภาพชุดแม่แบบขึ้นมาใหม่ โดยทำการกำหนดความสูงของภาพตัวเลขและตัวพยัญชนะไว้ที่ 65 พิกเซล แต่ค่าความกว้างของภาพตัวเลขและตัวพยัญชนะแต่ละตัวจะกำหนดค่าความกว้างให้ทุกตัวมีค่าเท่ากันไม่ได้ เช่น ตัวพยัญชนะ ก ไก่ มีค่าความกว้างของตัวพยัญชนะเท่ากับ 28 พิกเซลและตัวพยัญชนะตัว ณ เณร มีค่าความกว้างเท่ากับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

36 พิกเซล ดังนั้นในการกำหนดค่าความกว้างของภาพตัวเลขและตัวพยัญชนะของแต่ละตัว ขึ้นอยู่กับลักษณะของตัวเลขและตัวพยัญชนะตัวนั้นๆ ภาพตัวเลขและตัวพยัญชนะทั้งหมดที่ใช้เป็นแม่แบบมีทั้งหมด 45 ภาพ โดยมีภาพตัวเลขมีทั้งหมด 10 ภาพ และภาพตัวพยัญชนะมีทั้งหมด 35 ภาพ มีภาพตัวพยัญชนะที่ขาดหายไปทั้งหมด 5 ตัว คือตัว ซ ฒ ญ ฆ ฏ ฆ ฃ ฅ ซึ่งไม่สามารถหาได้จากแผ่นป้ายทะเบียน

### 3.3.1 วิธีการจัดทำตัวพยัญชนะและตัวเลขเป็นแม่แบบมีขั้นตอนดังนี้

1. ทำการจัดหาแผ่นป้ายทะเบียนที่มีความสว่างปรกติ หน้าตรง ตัวเลขและตัวพยัญชนะไม่เอียง และมีความสมบูรณ์ของตัวเลขและตัวพยัญชนะที่ชัดเจน ไม่ขาดช่วงกัน ไม่มีส่วนที่เป็นน็อตหรือตะปูติดอยู่ การถ่ายภาพถ่ายในระยะประมาณ 1 เมตร การหาแผ่นป้ายทะเบียนในขั้นตอนการจัดทำแม่แบบ ผู้จัดทำได้ทำการค้นหาภาพแผ่นป้ายทะเบียนรถยนต์บางส่วนมาจาก Internet เนื่องจากว่าภาพแผ่นป้ายทะเบียนที่คณะผู้ทำวิจัยได้ทำการถ่ายภาพมานั้น มีตัวเลขและตัวพยัญชนะบางส่วนที่ไม่ชัดเจน ไม่มีความสมบูรณ์และตัวพยัญชนะและตัวเลขบางตัวไม่สามารถหาได้ ส่วนภาพแผ่นป้ายทะเบียนที่หามาได้จาก Internet นั้น จะมีความสมบูรณ์ของตัวเลขและตัวพยัญชนะมากกว่า ดังเช่นรูปที่ 3.15 (ก) เป็นภาพตัวอย่างของแผ่นป้ายทะเบียนรถยนต์ที่ได้ทำการค้นหามาจาก www. talkystory.com

**รูปที่ 3.15 (ก)** ภาพแผ่นป้ายทะเบียนรถยนต์ที่ทำการค้นหาจาก Internet

2. แปลงข้อมูลในขั้นตอนที่ 1 จากภาพสี RGB ให้เป็นภาพระดับเทาและภาพสองระดับตามลำดับ

3. ทำการตัดขอบเขตด้านบนสุดและขอบเขตด้านล่างสุด ของตัวเลขและตัวพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ จากนั้นหาขอบเขตด้านซ้ายและขอบเขตด้านขวาของตัวเลขและตัวพยัญชนะทุกๆ ตัวที่อยู่บนแผ่นป้ายทะเบียนรถยนต์ เพื่อที่จะใช้เป็นแกนอ้างอิงในการแบ่งภาพตัวเลขและตัวพยัญชนะออกมาทีละตัวจากแผ่นป้ายทะเบียนรถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ทำการปรับขนาดของตัวเลขและตัวพยัญชนะด้วยโปรแกรม Adobe photoshop CS เพื่อที่จะทำการกำหนดค่าความกว้างและความสูงของตัวเลขและตัวพยัญชนะที่จะนำมาทำเป็นแม่แบบ โดยมีการกำหนดให้ขนาดความสูงของตัวเลขและตัวพยัญชนะทุกตัว ที่จะนำมาทำเป็นแม่แบบมีค่าความสูงที่ 65 พิกเซล และในการกำหนดค่าความกว้างของตัวเลขและตัวพยัญชนะแต่ละตัว จะกำหนดค่าความกว้างที่ไม่เท่ากัน เนื่องจากค่าของความกว้างขึ้นอยู่กับลักษณะของตัวเลขและตัวพยัญชนะตัวนั้นๆ ดังนั้นจึงต้องทำการหาค่าความกว้างของภาพตัวเลขและตัวพยัญชนะแต่ละตัวที่จะนำมาทำเป็นภาพชุดแม่แบบ เมื่อได้ค่าความกว้างของภาพตัวเลขและตัวพยัญชนะทุกตัว ที่จะมาทำเป็นภาพชุดแม่แบบแล้วก็จะใช้โปรแกรม Adobe photoshop CS ในการปรับขนาดของค่าความกว้างและความสูง โดยใช้คำสั่ง Image และเลือก Image Size เพื่อที่จะปรับขนาดของภาพตัวเลขและตัวพยัญชนะ

5. เมื่อได้ภาพตัวเลขและตัวพยัญชนะ ที่ทำการปรับขนาดความกว้างและความสูงครบทุกตัวแล้ว จากนั้นใช้โปรแกรม Paint เพื่อที่จะนำรูปของตัวเลขและตัวพยัญชนะทั้งหมดมาเรียงให้เป็นแถวเดียวกัน ดังรูปที่ 3.15

กขคงจฉชฌญฐฑฒถทธน
บปผฝพฟภมยรล
วศษสหฬอฮ
0123456789

**รูปที่ 3.15 (ข)** ชุดแม่แบบทั้งหมดที่ใช้ในการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.2 ขั้นตอนการรู้จำตัวเลขและตัวพยัญชนะ

1. ทำการโหลดภาพชุดแม่แบบขึ้นมาและหาขนาดของภาพชุดแม่แบบ จากนั้นหาหลัก (column) ที่เป็นขอบเขตด้านซ้ายสุดและขอบเขตด้านขวาสุดของตัวเลขและตัวพยัญชนะทั้งหมดในชุดแม่แบบมาเก็บอยู่ตัวแปรหนึ่ง และจึงทำการหาขอบเขตด้านบนและขอบเขตด้านล่างของตัวเลขและตัวพยัญชนะเก็บไว้ในอีกตัวแปรหนึ่ง

2. โหลดภาพตัวเลขหรือตัวพยัญชนะซึ่งเป็นภาพสองระดับที่ต้องการพิจารณา จากนั้นทำการหาขนาดของภาพ ที่ต้องการพิจารณาทั้งความสูงและความกว้างและทำการย่อขนาดของภาพที่ต้องการพิจารณาให้มีความสูงและความกว้างเท่ากับชุดแม่แบบที่เตรียมไว้ในหัวข้อที่ 3.3.1

**รูปที่ 3.16** ภาพตัวพยัญชนะที่ต้องการพิจารณา

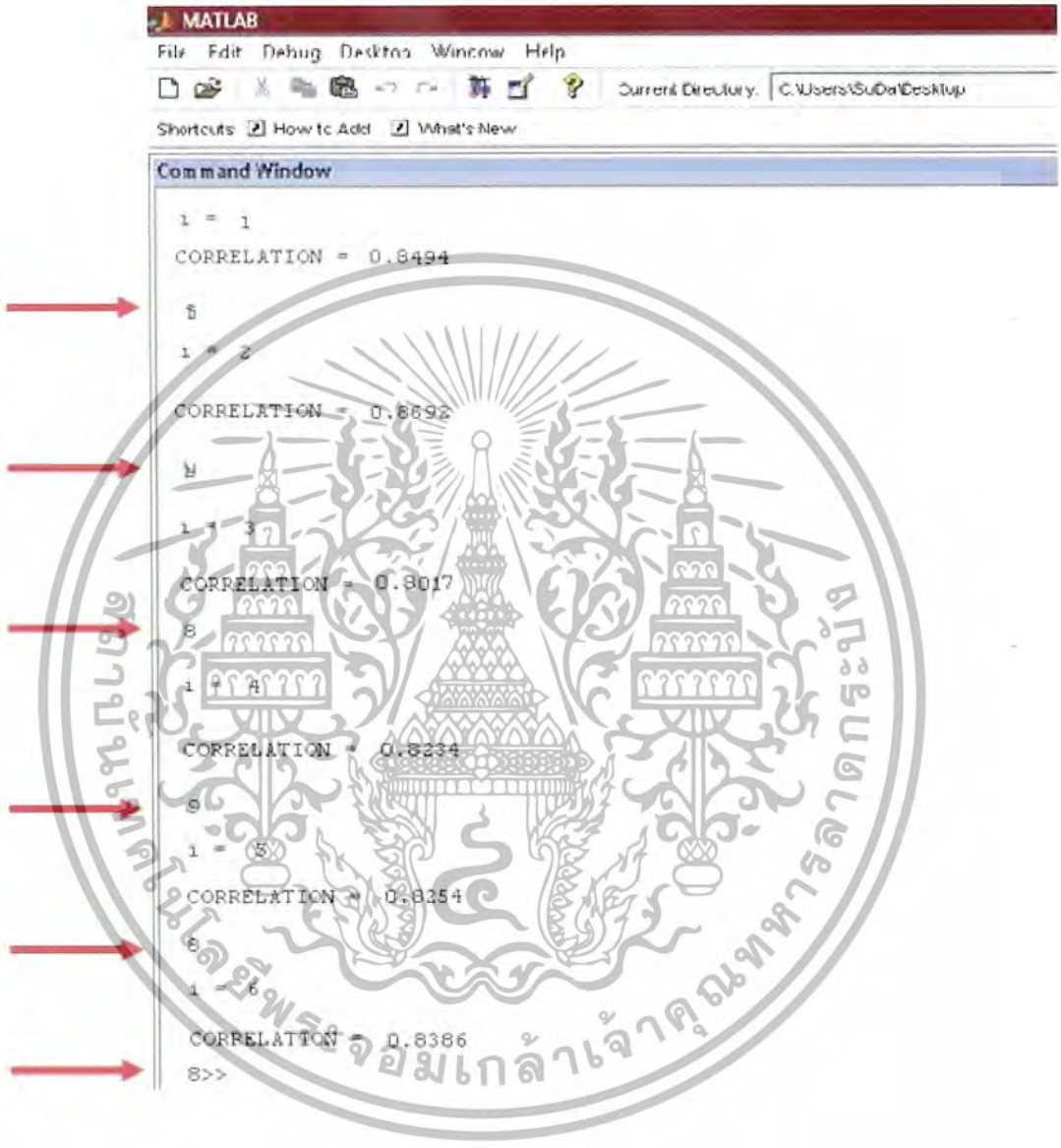
3. เมื่อย่อขนาดของภาพที่ต้องการพิจารณาให้มีขนาดทั้งความสูง และความกว้างเท่ากับภาพชุดแม่แบบที่ได้เตรียมไว้แล้ว ก็นำภาพตัวเลขหรือตัวพยัญชนะที่ต้องการพิจารณานั้นไปวางทาบกับภาพชุดแม่แบบทีละตัว เพื่อที่จะหาค่าสัมประสิทธิ์ความเหมือนกับภาพชุดแม่แบบแต่ละตัวว่ามีค่าสัมประสิทธิ์ความเหมือนกับภาพชุดแม่แบบแต่ละตัวเท่าใด โดยใช้สมการ 2.2 ในการหาค่าสัมประสิทธิ์ความเหมือนในหัวข้อที่ 2.3 และค่าผลลัพธ์ที่ได้มานั้นก็จะโชว์บน Command window โดยมีการกำหนดค่าตัดระดับเทสโฮล (threshould) ไว้ที่ 0.600 ถ้าค่าสัมประสิทธิ์ความเหมือนที่ได้จากการวางทาบกับแม่แบบทุกตัว ค่าที่ได้ค่าใดมีค่าน้อยกว่า 0.600 ก็จะไม่แสดงผล แต่ถ้าค่าใดมากกว่า 0.600 ก็จะแสดงผลตัวเลขหรือตัวพยัญชนะตัวนั้นขึ้นมา และโปรแกรมก็จะทำซ้ำวิธีเดิมกับทุกตัวพยัญชนะและตัวเลขบนแผ่นป้ายทะเบียนทั้งหมด

**ผลการทดลอง**

ผลที่ได้จากการนำภาพแผ่นป้ายทะเบียน ธษ 8988 ไปประมวลผลในส่วนการรู้จำตัวเลขและตัวพยัญชนะ เมื่อนำภาพตัว ธ.ธง ไปวางทาบกับภาพแม่แบบแล้วผลที่ได้ คือมีค่าสัมประสิทธิ์ความเหมือนที่มากกว่าค่าตัดระดับเทสโฮล 0.600 ที่ตั้งไว้เพียงค่าเดียวคือ 0.8494 ซึ่งเป็นค่าของภาพชุดแม่แบบตัวที่ 15 แสดงว่าข้อมูลภาพที่ต้องการพิจารณานั้นมีความเหมือนกับภาพชุดแม่แบบตัวที่ 15 มากที่สุด ซึ่งเมื่อย้อนกลับไปดูที่ภาพชุดแม่แบบภาพชุดแม่แบบตัวที่ 15 ก็คือตัวพยัญชนะ ธ.ธง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลที่ได้มานั้นจะถูกแสดงออกมาบน Command window ของโปรแกรม MATLAB ส่วนตัวพยัญชนะหรือตัวเลขอื่นๆ ก็จะทำในลักษณะเช่นนี้เหมือนกัน ผลลัพธ์ที่ได้แสดงดังรูปที่ 3.17

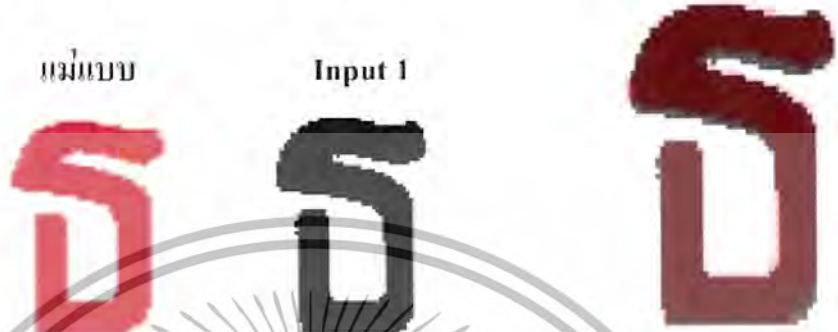



**รูปที่ 3.17** หน้าต่าง Command window ที่เป็นผลมาจากการนำเอาข้อมูลภาพตัวเลขและตัวพยัญชนะทุกตัวที่อยู่บนแผ่นป้ายทะเบียนมาหาค่าสัมประสิทธิ์ความเหมือนกับภาพชุดแม่แบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตัวอย่าง** ผลที่ได้จากการนำภาพแผ่นป้ายทะเบียน ธษ 8988 ไปประมวลผลในการหาค่าสัมประสิทธิ์ความเหมือนกับภาพชุดแม่แบบ

**Input >> Letter 1 ( ค่า Correlation = 0.8494 หรือ 84.94% )**



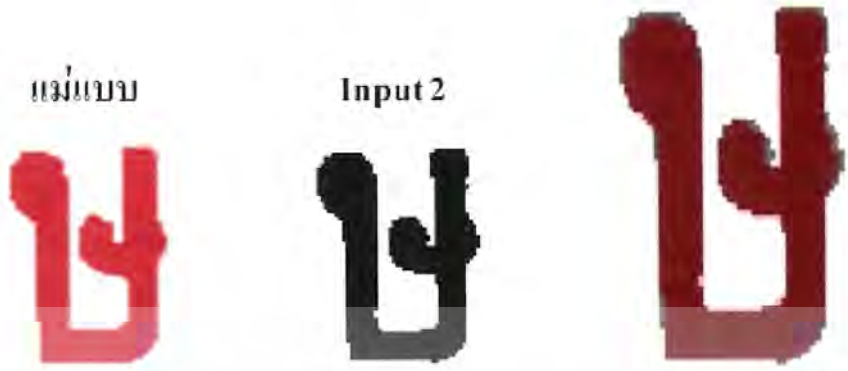

**(ก)**

```
i = 1
CORR =
  Columns 1 through 15
    0.2201    0.1182    0.1254   -0.0377    0.0139
    0.2628    0.1690    0.0887    0.3014    0.4698
    0.1010    0.1399    0.2438    0.2220    0.8494
  Columns 16 through 30
    0.3039    0.2988    0.3482   -0.0594   -0.1640
    0.1840    0.0643    0.2221    0.0232    0.2722
    0.5153    0.0864    0.2364    0.2404    0.4044
  Columns 31 through 45
    0.0587    0.2353    0.2398    0.3070    0.4897
    0.3277    0.1223    0.2112    0.3694    0.0868
    0.3982    0.3647    0.0680    0.3607    0.2717
CORRELATION =  0.8494
```

จากรูปที่ 3.18 (ก) จะเห็นว่าค่าสัมประสิทธิ์ความเหมือนที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้ที่ 0.600 มีเพียงค่าเดียวคือ 0.8494 ซึ่งเมื่อย้อนกลับไปดูที่ภาพชุดแม่แบบจะเป็นตัวพยัญชนะ ธ.ธง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Input >> Letter 2 (ค่า Correlation = 0.8692 หรือ 86.92%)**



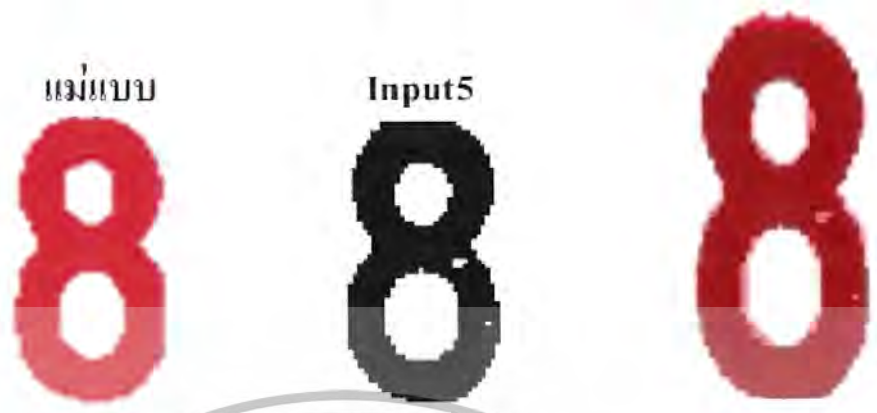

(ข)

```
i = 2
CORR =
  Columns 1 through 15
    0.1437    0.1614    0.1373    0.0322    0.1742
    0.2477    0.2907   -0.0105    0.3188    0.1454
    0.1057    0.1337    0.2010    0.4042    0.4523
  Columns 16 through 30
    0.5187    0.5125    0.5039    0.0390   -0.0357
    0.4303    0.3085    0.2279    0.2268    0.3569
    0.1720    0.1895    0.1880    0.2106    0.8692
  Columns 31 through 45
    0.1824    0.4437    0.2408    0.2705    0.2293
    0.3033    0.0716    0.1628    0.3929    0.0130
    0.3100    0.3280    0.0776    0.4276    0.2375
CORRELATION =  0.8692
 ษ
```

จากรูปที่ 3.18 (ข) จะเห็นว่า ค่าสัมประสิทธิ์ความเหมือนที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้ที่ 0.600 มีเพียงค่าเดียวคือ 0.8692 ซึ่งเมื่อย้อนกลับไปดูที่ภาพชุดแม่แบบจะเป็นตัวพยัญชนะ ษ.ฤาษี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Input >> Letter 3 (ค่า Correlation = 0.8017 หรือ 80.17%)**



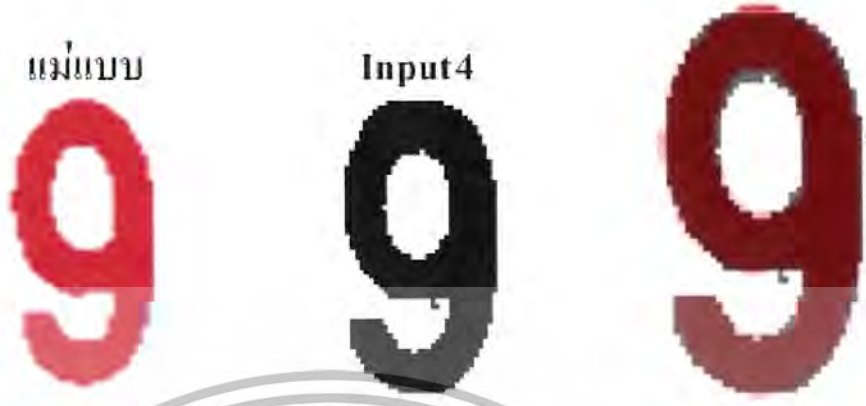

(ค)

```
i = 3
CORR =
  Columns 1 through 15
    0.3076    0.0058    0.2838   -0.0411    0.2329
    0.2249    0.0918    0.1490    0.3559    0.0728
    0.1563    0.2189    0.3221    0.1400    0.3631
  Columns 16 through 30
    0.1751    0.1677    0.2074   -0.0192   -0.0121
    0.0786    0.0198    0.2114    0.0559    0.2645
    0.2106    0.3237    0.2570    0.2600    0.3871
  Columns 31 through 45
    0.2334    0.2025    0.0642    0.4214    0.3669
    0.3781   -0.0322    0.1341    0.6203    0.1081
    0.5321    0.5739    0.0088    0.8017    0.3104
  CORRELATION =  0.8017
  8
```

จากรูปที่ 3.18 (ค) จะเห็นว่า ค่าสัมประสิทธิ์ความเหมือนที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้ที่ 0.600 มีเพียงค่าเดียวคือ 0.8017 ซึ่งเมื่อย้อนกลับไปดูที่ภาพชุดแม่แบบจะเป็นตัวเลข 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Input >> Letter 4 (ค่า Correlation = 0.8234 หรือ 82.34%)**



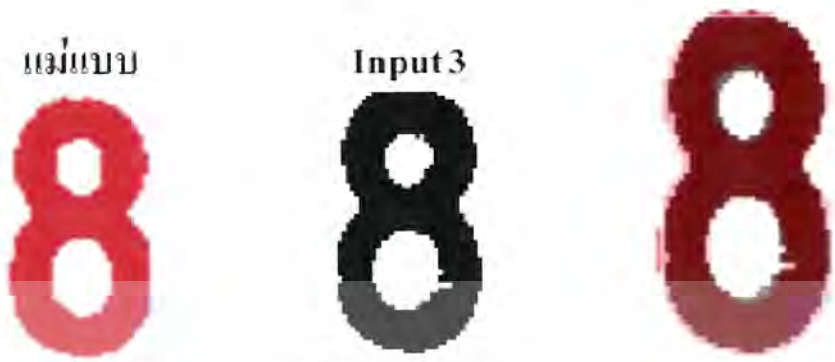

(ง)

```
i = 4
CORR =
  Columns 1 through 15
    0.3219    0.0986    0.1348    0.2356    0.3032
    0.3271    0.1231    0.1941    0.2792    0.1666
    0.1453    0.1979    0.3400    0.0723    0.2965
  Columns 16 through 30
    0.1886    0.1820    0.2181    0.0919   -0.0111
    0.0423    0.0503    0.2022    0.0747    0.2529
    0.1839    0.2250    0.4071    0.1617    0.2468
  Columns 31 through 45
    0.1867    0.0891    0.2009    0.3973    0.2737
    0.4954    0.0956    0.3696    0.3757   -0.0985
    0.3868    0.3521   -0.0108    0.4197    0.8234
  CORRELATION = 0.8234
   9
```

จากรูปที่ 3.18 (ง) จะเห็นว่า ค่าสัมประสิทธิ์ความเหมือนที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้ที่ 0.600 มีเพียงค่าเดียวคือ 0.8234 ซึ่งเมื่อย้อนกลับไปดูที่ภาพชุดแม่แบบจะเป็นตัวเลข 9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Input >> Letter 5 (ค่า Correlation = 0.8254 หรือ 82.54%)**



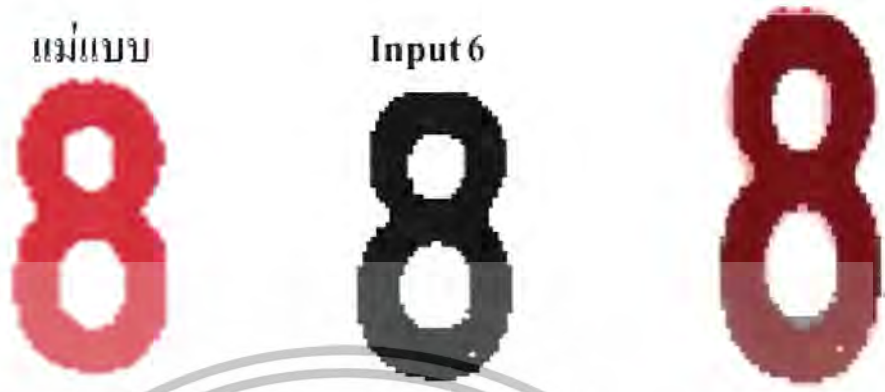

(จ)

```
i = 5
CORR =
  Columns 1 through 15
    0.3102   -0.0104    0.2798   -0.0855    0.2260
    0.2342    0.0781    0.1669    0.3426    0.0833
    0.1452    0.2032    0.3374    0.1339    0.3439
  Columns 16 through 30
    0.1915    0.1455    0.1838   -0.0517   -0.0318
    0.0950    0.0096    0.2270    0.0020    0.1912
    0.2396    0.3258    0.2621    0.2597    0.3551
  Columns 31 through 45
    0.2543    0.1669    0.0914    0.4079    0.3422
    0.4557    0.0147    0.0990    0.5988    0.1625
    0.5205    0.6169    0.0003    0.8254    0.3485
  CORRELATION =  0.8254
   8
```

จากรูปที่ 3.18 (จ) จะเห็นว่า ค่าสัมประสิทธิ์ความเหมือนที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้ที่ 0.600 มีเพียงค่าเดียวคือ 0.8254 ซึ่งเมื่อย้อนกลับไปดูที่ภาพชุดแม่แบบจะเป็นตัวเลข 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Input >> Letter 6 (ค่า Correlation = 0.8386 หรือ 83.86%)**




(ฉ)

```
i =  6

CORR =

 Columns 1 through 15

   0.3014   -0.0118    0.2831   -0.0298    0.1757
   0.2215    0.0819    0.1231    0.4206    0.0714
   0.1113    0.2018    0.3357    0.1559    0.3621

 Columns 16 through 30

   0.2319    0.2035    0.2466   -0.0235    0.0205
   0.0839    0.0230    0.2875    0.0564    0.2087
   0.2284    0.3240    0.2646    0.2639    0.3907

 Columns 31 through 45

   0.2366    0.1768    0.0883    0.4673    0.3230
   0.4220   -0.0377    0.1107    0.6407    0.1389
   0.4927    0.6110   -0.0107    0.8386    0.3020

CORRELATION =  0.8386

 8>>
```

จากรูปที่ 3.18 (ฉ) จะเห็นว่า ค่าสัมประสิทธิ์ความเหมือนที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้ที่ 0.600 มีเพียงค่าเดียวคือ 0.8386 ซึ่งเมื่อย้อนกลับไปดูที่ภาพชุดแม่แบบจะเป็นตัวเลข 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลการทดลองจะเห็นได้ว่าจากรูปที่ 3.18 (ก) - (ฉ) ภาพที่ต้องการพิจารณาทั้งหมด 6 ตัว (ตัวพยัญชนะและตัวเลขสีดำ) มาวางทาบกับภาพชุดแม่แบบ (ตัวพยัญชนะและตัวเลขสีแดง) จะพบว่าค่าสัมประสิทธิ์ความเหมือนที่ได้จะมีค่าไม่เท่ากับ 1 หรือ 100 % เนื่องจากภาพที่ต้องการพิจารณาที่ได้จากแผ่นป้ายทะเบียนมีความไม่สมบูรณ์ของตัวเลขและตัวพยัญชนะ เมื่อนำภาพที่ต้องการพิจารณาไปวางทาบกับภาพชุดแม่แบบที่ได้เตรียมไว้ ค่าสัมประสิทธิ์ความเหมือนที่ได้ก็จะมีค่าน้อยลง ส่วนตัวพยัญชนะหรือตัวเลขตัวเดียวกัน อาจจะมีค่าสัมประสิทธิ์ความเหมือนที่ไม่เท่ากัน เนื่องจากตัวเลขหรือตัวพยัญชนะบนแผ่นป้ายทะเบียนแผ่นเดียวกัน อาจจะมีความสมบูรณ์หรือความชัดเจนไม่เท่ากันทุกตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

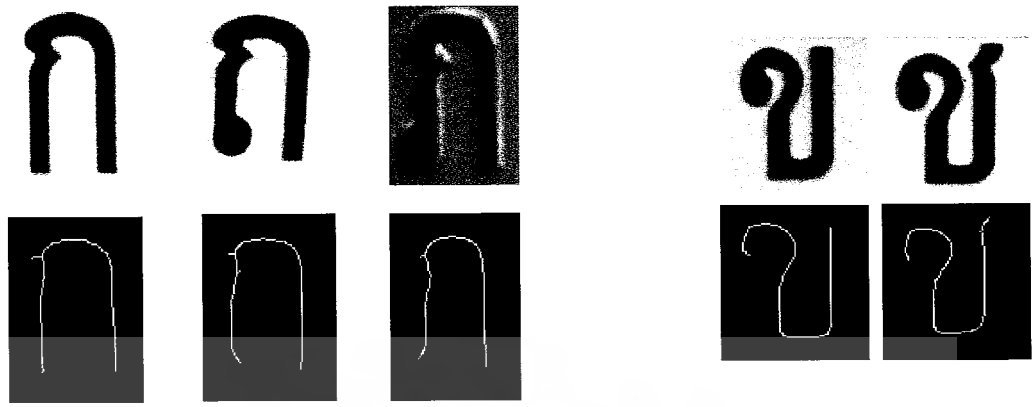
# บทที่ 4
# การรู้จำตัวอักษรบนพื้นฐานลักษณะเด่น

ในบทนี้จะอธิบายถึงขั้นตอนการตรวจหาลักษณะเด่นบนตัวพยัญชนะและตัวเลขแต่ละตัวที่อยู่บนแผ่นป้ายทะเบียนรถยนต์ อย่างไรก็ตามเนื่องจากหัวของพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์มีลักษณะที่เป็นหัวทึบทั้งหมด (ดังรูปที่ 4.1) ดังนั้นเมื่อนำตัวพยัญชนะแต่ละตัวมาทำการหาเส้นโครงร่างจึงทำให้เกิดปัญหาความคล้ายคลึงกันของตัวพยัญชนะ (ดังรูปที่ 4.2) ตัวอย่างเช่น ตัวพยัญชนะ ก.ไก่ กับตัวพยัญชนะ ถ.ถุง และตัวพยัญชนะ ภ.สำเภา หรือตัวพยัญชนะ ด.เด็ก กับตัวพยัญชนะ ค.ควาย และอื่นๆ เป็นต้น ดังที่ได้กล่าวไว้แล้วข้างต้นเมื่อทำการหาเส้นโครงร่างของตัวพยัญชนะแต่ละตัวจะไม่สามารถแยกแยะได้ว่าตัวพยัญชนะนั้นเป็นตัวพยัญชนะใด จึงได้มีแนวคิดที่จะสร้างความแตกต่างให้กับตัวพยัญชนะแต่ละตัวเพื่อให้ง่ายต่อการแยกแยะตัวพยัญชนะออกจากกันก่อน ซึ่งขั้นตอนการตรวจหาลักษณะเด่นบนตัวพยัญชนะมีดังต่อไปนี้

1. ทดลองใส่วงกลมสีขาวที่หัวของตัวพยัญชนะ ด้วยโปรแกรม Paint เพื่อสังเกตความแตกต่างที่เกิดขึ้น
2. ตรวจหาหัวของตัวพยัญชนะด้วยแนวคิดการหาพื้นที่ที่มากที่สุดบนตัวพยัญชนะ ด้วยโปรแกรม MATLAB เพื่อใส่วงกลมสีขาวที่หัวพยัญชนะ
3. ตรวจหาหัวของตัวพยัญชนะด้วยแนวคิดการหาความหนาของตัวพยัญชนะ ด้วยโปรแกรม MATLAB

**รูปที่ 4.1** ตัวพยัญชนะที่ได้มาจากแผ่นป้ายทะเบียนรถยนต์ซึ่งมีลักษณะหัวทึบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 4.2** ตัวพยัญชนะภายหลังหาเส้นโครงร่างแล้วซึ่งตัวพยัญชนะเหล่านี้มีลักษณะที่คล้ายคลึงกันมากยากที่จะแยกแยะได้ว่าเป็นตัวพยัญชนะตัวใด

**4.1 ทดลองใส่วงกลมสีขาวที่หัวของตัวพยัญชนะด้วยโปรแกรม Paint เพื่อสังเกตความแตกต่างที่ เกิดขึ้น**

เมื่อนำตัวพยัญชนะและตัวเลขบนแผ่นป้ายทะเบียนรถยนต์ ซึ่งส่วนหัวของพยัญชนะมีลักษณะหัวทึบมาทำการหาเส้นโครงร่างด้วยทฤษฎีการหาเส้นโครงร่าง ที่ได้อธิบายไว้ในหัวข้อ 2.4 ผลลัพธ์จะได้ดังรูปที่ 4.3

**รูปที่ 4.3** การหาเส้นโครงร่างของตัวพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์

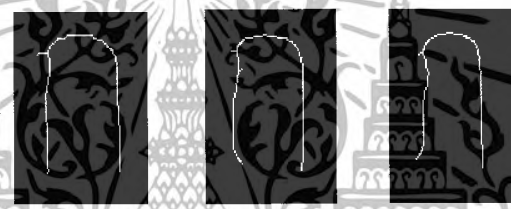
จากรูปที่ 4.3 เป็นภาพที่ได้จากการหาเส้นโครงร่างด้วยโปรแกรม MATLAB ตามที่ได้อธิบายไว้ในหัวข้อที่ 2.4 ผลลัพธ์ที่ได้ปรากฏว่าตัวพยัญชนะบางตัวมีลักษณะคล้ายคลึงกันมาก เช่น ตัวพยัญชนะ ก.ไก่ กับตัวพยัญชนะ ถ.ถุง และตัวพยัญชนะ ภ.สัมเภา และตัวพยัญชนะ ข.ไข่ กับตัวพยัญชนะ ช.ช้าง และตัวพยัญชนะ ค.ควาย กับตัวพยัญชนะ ด.เด็ก และตัวพยัญชนะ ต.เต่า เป็นต้น ทำให้ยากที่จะให้โปรแกรมสามารถแยกแยะได้ว่าเส้นโครงร่างที่ได้รับเป็นตัวพยัญชนะตัวใด ปัญหาดังกล่าวจึงมีผลต่อขั้นตอนการรู้จำตัวอักษร ด้วยเหตุนี้จึงได้มีแนวคิดที่จะแก้ไขปัญหาด้วยการใส่วงกลมสีขาวที่หัวของตัวพยัญชนะ โดยการสร้างรูปวงรีไว้ที่ตรงกลางหัวของตัวพยัญชนะแต่ละตัวด้วยโปรแกรม Paint เพื่อทดสอบดูว่ามีความแตกต่างเกิดขึ้นหรือไม่ ผลการเปรียบเทียบการใส่วงกลมที่หัวของตัวพยัญชนะมีดังรูปที่ 4.4 (ก) และ (ข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

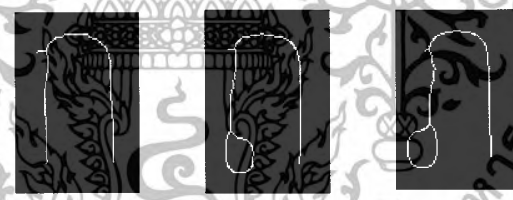
(ก)

(ข)

**รูปที่ 4.4 (ก)-(ข)** ความแตกต่างที่เกิดขึ้นอย่างชัดเจนตรงส่วนหัวของตัวพยัญชนะระหว่างตัวพยัญชนะที่ยังไม่ได้ทำการแก้ไข (ก) กับพยัญชนะที่ทำการแก้ไขแล้ว (ข)



(ก)



(ข)

**รูปที่ 4.4 (ค)–(ง)** เส้นโครงร่างของตัวพยัญชนะที่ยังไม่ได้แก้ไขและแก้ไขแล้ว

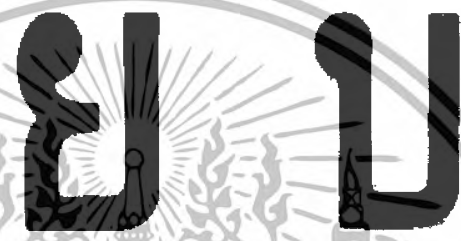
ผลลัพธ์เปรียบเทียบที่ได้ จากการหาเส้นโครงร่างของตัวพยัญชนะทั้งที่ยังไม่ได้แก้ไขและแก้ไขแล้วแสดงดังรูปที่ 4.4 (ค)-(ง) แสดงให้เห็นว่าแนวคิดดังกล่าวใช้ได้ผลจึงได้นำแนวคิดนี้ไปเริ่มขั้นตอนต่อไป คือขั้นตอนการออกแบบโปรแกรมการใส่หัวให้กับพยัญชนะแต่ละตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**4.2 ตรวจหาหัวของตัวพยัญชนะด้วยแนวคิดการหาพื้นที่ที่มากที่สุดบนพยัญชนะด้วยโปรแกรม MATLAB**

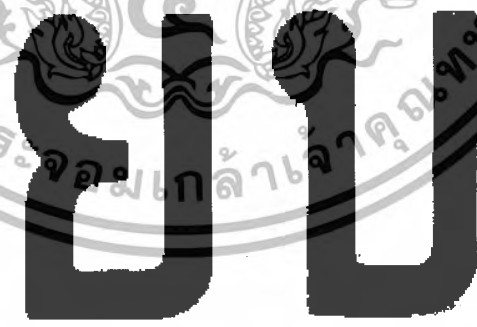
จากแนวคิดที่ว่าหัวของตัวพยัญชนะไทยส่วนใหญ่จะมีพื้นที่ที่มากที่สุด ดังนั้นจึงใช้หน้าต่างขนาด 3*3 วางทับลงไปบนตัวพยัญชนะ ทำให้พื้นที่ตัวพยัญชนะถูกแบ่งออกเป็น 9 ช่องเท่าๆ กัน จากนั้นก็จะใช้หลักการหาพื้นที่ที่มากที่สุดในแต่ละช่อง เพื่อจะดึงส่วนที่เป็นหัวพยัญชนะออกมา ซึ่งมีขั้นตอนดังต่อไปนี้

1. โหลดภาพตัวพยัญชนะแต่ละตัวที่ได้จากขั้นตอนที่ 3 ของหัวข้อที่ 3.2 ของการแยกส่วนตัวเลขและตัวพยัญชนะออกจากแผ่นป้ายทะเบียนจากฐานข้อมูลในคอมพิวเตอร์



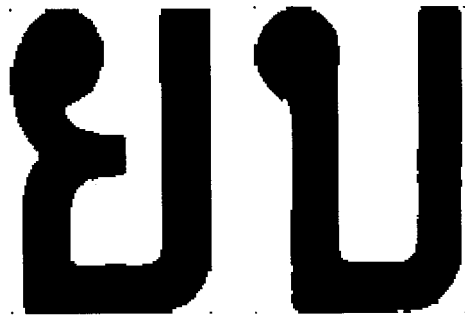
**รูปที่ 4.5** ตัวอย่างภาพตัวพยัญชนะที่โหลดจากฐานข้อมูลในคอมพิวเตอร์

2. กรณีมีสิ่งรบกวนอยู่บนรูปตัวพยัญชนะ ทำการกำจัดสิ่งรบกวนนั้นด้วยคำสั่ง medianfilter หน้าต่างที่ใช้มีขนาด 3x3 หรือมากกว่าขึ้นอยู่กับขนาดสิ่งรบกวนนั้นมีมากน้อยแค่ไหน ถ้ามีสิ่งรบกวนมากก็ต้องเพิ่มขนาดเข้าไปเป็น 5x5 หรือ 7*7 หรือ 9x9 เป็นต้น



**(ก)**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**(ข)**

**รูปที่ 4.7 (ก)** ภาพที่ยังไม่ได้ทำการกำจัดสิ่งรบกวน

**(ข)** ภาพที่ได้ทำการกำจัดสิ่งรบกวนแล้ว

เนื่องจากภาพตัวพยัญชนะที่นำมาเป็นตัวอย่างนี้มีสิ่งรบกวนอยู่ไม่มาก ดังนั้นเมื่อการกำจัดสิ่งรบกวนแล้ว ภาพผลลัพธ์ที่ได้จึงไม่แตกต่างกัน อย่างไรก็ตามถ้าภาพใช้มีสิ่งรวบกวนเป็นจำนวนมากเมื่อทำการกำจัดสิ่งรบกวนแล้ว ภาพผลลัพธ์ที่ได้จะมีความแตกต่างอย่างชัดเจน

3. กลับภาพตัวพยัญชนะที่มีสีดำให้เป็นสีขาวและจากสีขาวเป็นสีดำ ภาพผลลัพธ์ที่ได้จะแสดงดังรูปที่ 4.8

**รูปที่ 4.8** ภาพที่ได้ภายหลังการกลับภาพตัวพยัญชนะจากสีขาวเป็นสีดำ

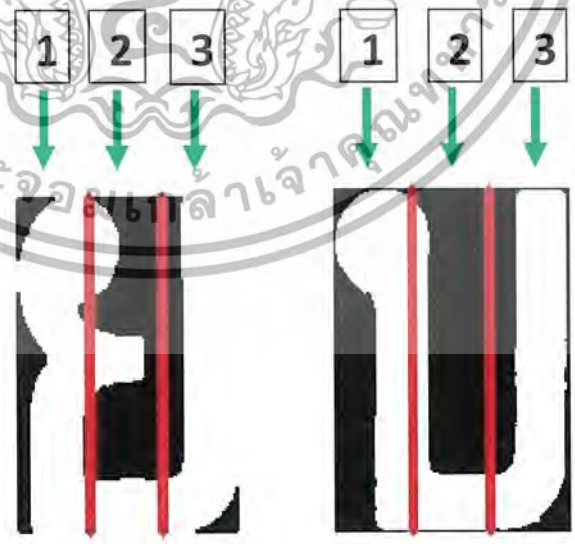
4. ทำการแบ่งพื้นที่ภาพตัวพยัญชนะออกเป็น 9 ช่องเท่าๆ กัน โดยทำการแบ่งความกว้างและความสูงของตัวพยัญชนะออกเป็น 3 ส่วนเท่าๆ กัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**กรณีที่ 1** แบ่งความสูงของตัวพยัญชนะออกเป็น 3 ส่วนเท่าๆ กัน เมื่อทราบความสูงของตัวพยัญชนะว่ามีค่าเท่าใด จากนั้นนำค่าความสูงของตัวพยัญชนะนั้นหารด้วยสามก็จะได้พื้นที่ตัวพยัญชนะออกเป็นสามส่วนเท่าๆ กัน โดยจะใช้เส้นสีแดงในการแบ่งพื้นที่ทั้งสามส่วนออกจากกันดังรูปที่ 4.9

**รูปที่ 4.9** ภาพที่ได้จากการแบ่งตัวพยัญชนะออกเป็นสามส่วนในแนวนอน (แบ่งแถว) โดยเส้นสีแดงคือเส้นแบ่งพื้นที่

**กรณีที่ 2** แบ่งความกว้างของตัวพยัญชนะออกเป็น 3 ส่วนเท่าๆ กัน เมื่อทราบความกว้างของตัวพยัญชนะว่ามีค่าเท่าใด จากนั้นก็นำค่าความกว้างของตัวพยัญชนะนั้นหารด้วยสามก็จะได้พื้นที่ตัวพยัญชนะออกเป็นสามส่วนเท่าๆ กัน โดยจะใช้เส้นสีแดงในการแบ่งพื้นที่ทั้งสามส่วนออกจากกันดังรูปที่ 4.10




**รูปที่ 4.10** ภาพที่ได้จากการแบ่งตัวพยัญชนะออกเป็นสามส่วนในแนวตั้ง (แบ่งหลัก) โดยเส้นสีแดงคือเส้นแบ่งพื้นที่

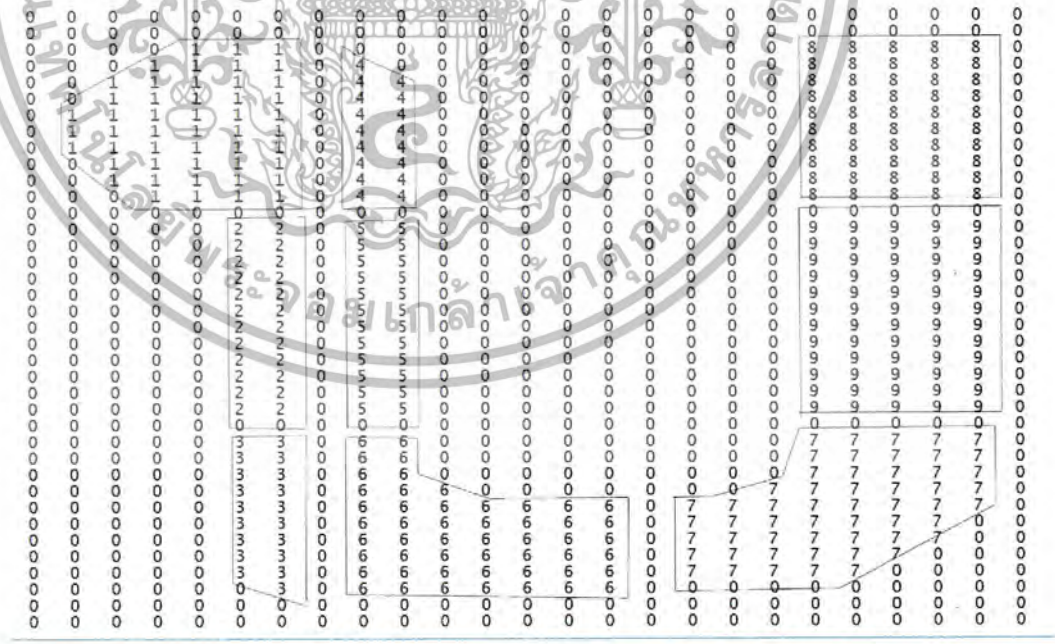
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อแบ่งตัวพยัญชนะออกเป็นสามส่วนเท่าๆกันทั้ง 2 กรณีแล้ว ผลลัทธ์ที่ได้จะทำให้ตัวพยัญชนะถูกแบ่งออกเป็น 9 ช่องเท่าๆ กัน ดังรูปที่ 4.11

**รูปที่ 4.11** ภาพที่ได้จากการแบ่งตัวพยัญชนะออกเป็น 9 ช่องเท่าๆ กัน

จากรูปที่ 4.11 จะเห็นว่าเมื่อทำการแบ่งตัวพยัญชนะออกเป็น 9 ช่องเท่าๆ กันแล้ว ทำให้สอดคล้องกับแนวความคิดที่ว่า ตัวพยัญชนะไทยมีส่วนหัวของตัวพยัญชนะเป็นพื้นที่สีขาวที่มากที่สุด

5. กำหนดหมายเลข (Label) ให้กับกลุ่มพิกเซลทั้งหมด 9 ช่อง เพื่อให้กลุ่มพิกเซลสีขาวในแต่ละช่องเป็นกลุ่มพิกเซลเดียวกัน ดังแสดงในรูปที่ 4.12



**รูปที่ 4.12** ภาพแสดงหมายเลขของพิกเซลสีขาวของแต่ละช่องที่ได้ทำการแบ่งไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ดึงกลุ่มพิกเซลสีขาวที่มากที่สุดออกมา เพื่อจะดูว่าส่วนที่มากที่สุดที่ถูกดึงออกมานั้นจะเป็นส่วนหัวของตัวพยัญชนะดังที่คิดไว้หรือไม่ ดังรูปที่ 4.13

**รูปที่ 4.13** ภาพที่ได้หลังจากการดึงส่วนที่มีกลุ่มพิกเซลสีขาวที่มากที่สุดออกมา

จากรูปที่ 4.13 จะเห็นว่ากลุ่มพิกเซลสีขาวที่มากที่สุดของตัวพยัญชนะ ย.ยักษ์และบ.ใบไม้ ที่ถูกดึงขึ้นมาส่วนที่ได้จะเป็นส่วนหัวของตัวพยัญชนะตามที่ต้องการ

7. สร้างพื้นที่สี่เหลี่ยมลงไปในหัวของตัวพยัญชนะที่ดึงขึ้นมา เพื่อทำให้ตัวพยัญชนะที่มีหัวทึบ กลายเป็นตัวพยัญชนะหัวโปร่งแทนแสดงดังรูปที่ 4.14

**รูปที่ 4.14** ภาพการสร้างพื้นที่สี่เหลี่ยมลงไปตรงกลางของกลุ่มพิกเซลสีขาวที่มากที่สุด เพื่อให้ตัวพยัญชนะจากที่มีหัวทึบกลายเป็นตัวพยัญชนะที่มีหัวโปร่งแทน

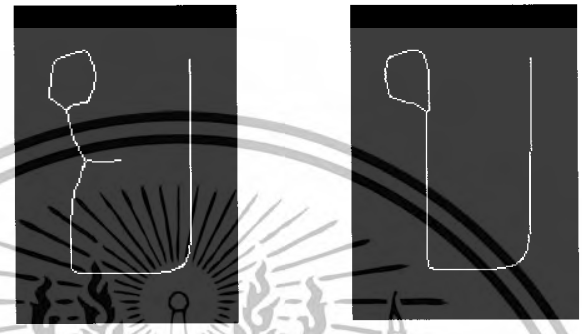
หลังจากที่ได้ทำการวางรูปที่ได้สร้างสี่เหลี่ยมลงไปภาพผลลัพธ์ที่ได้จะเป็นดังรูปที่ 4.16

**รูปที่ 4.16** ภาพการเปรียบเทียบตัวพยัญชนะที่แก้ไขหัวของตัวพยัญชนะและยังไม่ได้แก้ไข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.16 จะเห็นได้ชัดว่า เมื่อทำการแก้ไขหัวของตัวพยัญชนะที่มีหัวทึบให้กลายเป็นตัวพยัญชนะที่มีหัวโปร่ง จะเห็นว่าตัวพยัญชนะที่ได้ที่มีหัวโปร่งจะดูง่ายกว่าตัวพยัญชนะที่มีหัวทึบ

เมื่อแก้ไขหัวของตัวพยัญชนะให้มีลักษณะหัวโปร่งแล้ว จากนั้นทำการหาเส้นโครงร่างของตัวพยัญชนะ ผลลัพธ์ที่ได้เป็นดังรูปที่ 4.17 (ก)



**รูปที่ 4.17 (ก)** ภาพเส้นโครงร่างตัวพยัญชนะที่ใส่หัวแล้ว

จากการูปที่ 4.16 จะเห็นว่าเมื่อทำการหาเส้นโครงร่างของตัวพยัญชนะทำให้สามารถแยกแยะได้ว่าเป็นตัวพยัญชนะใด รูปที่ 4.17 (ก) แสดงภาพเส้นโครงร่างเปรียบเทียบระหว่างตัวพยัญชนะที่ไม่ได้ใส่หัว และใส่หัวแล้ว

**รูปที่ 4.17(ข)** ภาพเส้นโครงร่างเปรียบเทียบระหว่างตัวพยัญชนะที่ไม่ได้ใส่หัวและใส่หัวแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างไรก็ตามวิธีที่อธิบายมาข้างต้น สามารถแก้ไขปัญหาตัวพยัญชนะที่มีลักษณะคล้ายกันได้แต่ไม่สามารถใช้วิธีนี้ได้กับตัวพยัญชนะทั้งหมด เนื่องจากการหาพื้นที่ที่มากที่สุดในตัวพยัญชนะนั้น ตัวพยัญชนะบางตัวไม่ได้มีตัวของหัวพยัญชนะเป็นพื้นที่ที่มากที่สุด ทำให้ได้ส่วนอื่นของตัวพยัญชนะขึ้นมาแทน ดังรูปที่ 4.18

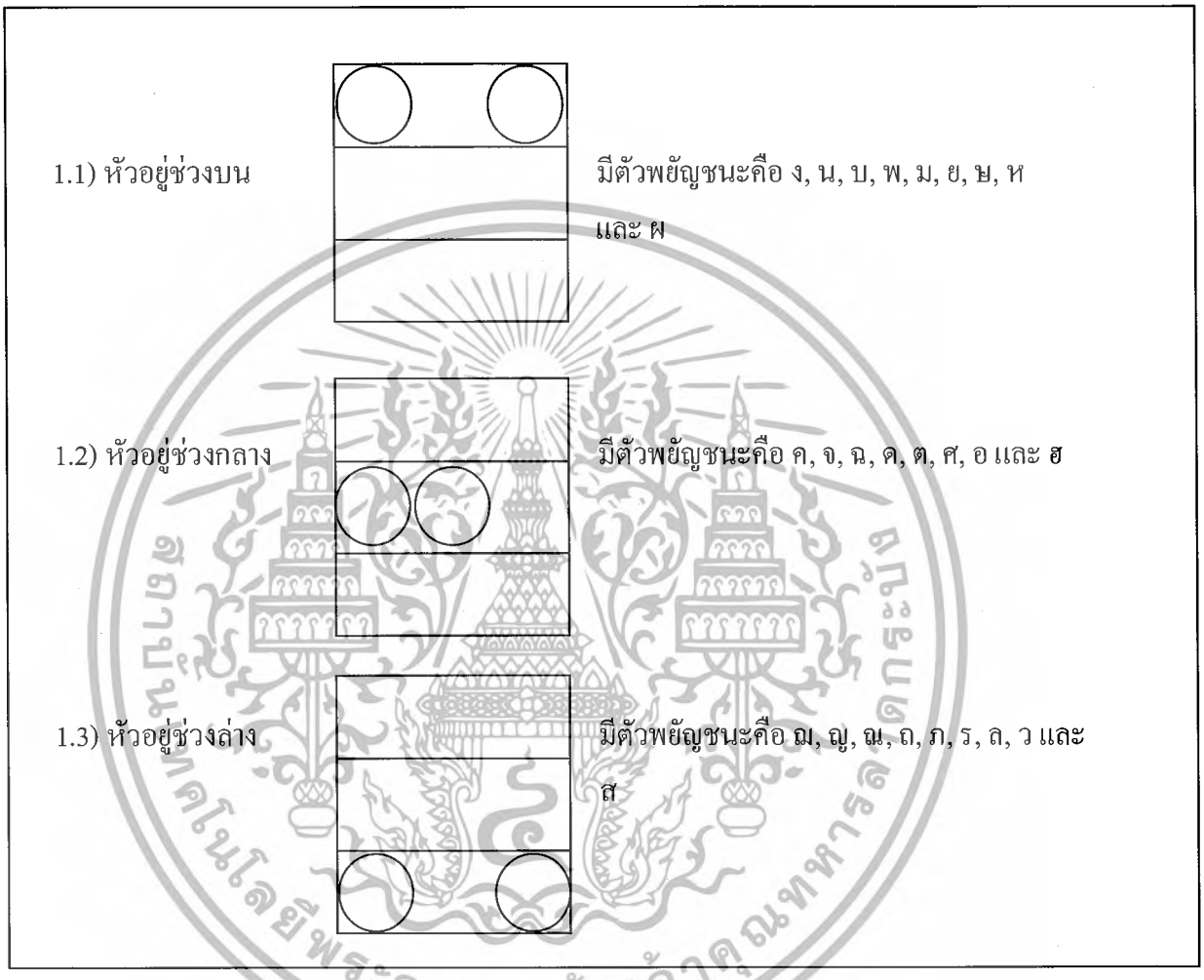
รูปที่ 4.18 ตัวพยัญชนะบางตัวที่กลุ่มพิกเซลสีขาวที่มากที่สุดไม่ใช่ส่วนหัว ทำให้ได้ส่วนอื่นออกมาแทน

**4.3 ตรวจหาเพื่อใส่วงกลมสีขาวที่หัวของตัวพยัญชนะด้วยแนวคิดการหาความหนาของตัวพยัญชนะด้วยโปรแกรม MATLAB**

จากการหาเส้นโครงร่างของตัวพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ ที่มีลักษณะเป็นหัวทึบเกิดปัญหาความคล้ายคลึงกันของตัวพยัญชนะบางตัว อาทิเช่น ตัวพยัญชนะ ก.ไก่ คล้ายกับตัวพยัญชนะ ถ.ถุง และตัวพยัญชนะ ภ.สำเภา ตัวพยัญชนะ ด.เด็ก คล้ายกับตัวพยัญชนะ ต.เต่า เป็นต้น จากปัญหานี้ทำให้เกิดความยุ่งยากขึ้น ในขั้นตอนการรู้จำเนื่องจากตัวพยัญชนะมีความแตกต่างกันน้อยเกินไปทำให้มีเงื่อนไขไม่มากพอในการแยกตัวพยัญชนะออกจากกัน จึงเกิดแนวความคิดในการหาความแตกต่างจากลักษณะเฉพาะของพยัญชนะแต่ละตัว คือการทำให้ส่วนหัวของตัวพยัญชนะบนแผ่นป้ายทะเบียนที่มีลักษณะทึบให้เป็นหัวโปร่ง จึงออกแบบโปรแกรมเพื่อหาส่วนหัวของตัวพยัญชนะให้ได้แล้วจึงใส่หัวให้กับตัวพยัญชนะนั้น โดยโปรแกรมหาส่วนหัวนั้นมีรูปแบบดังนี้

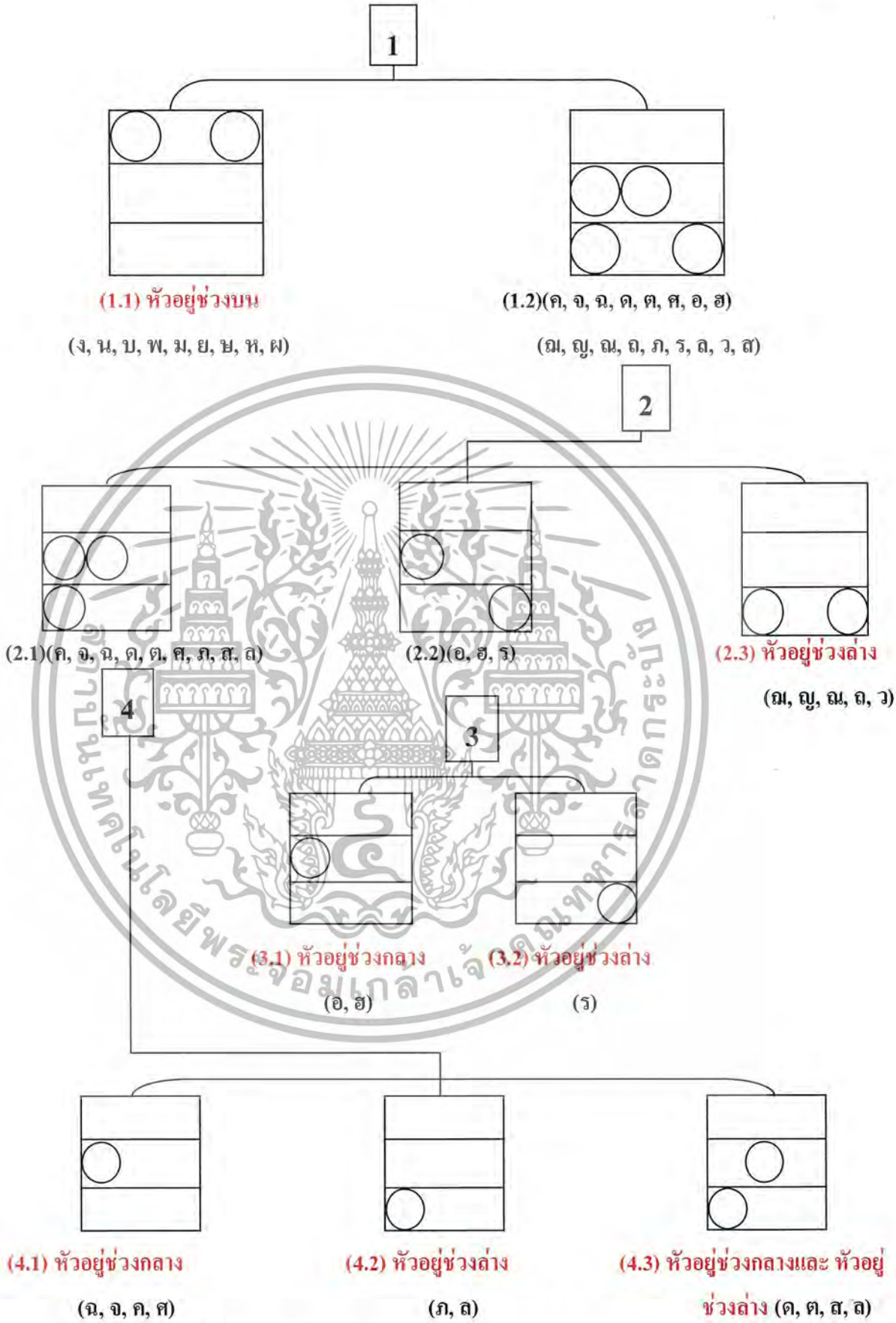
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบแผนแสดงตำแหน่งต่างๆของส่วนหัวของตัวพยัญชนะ เพื่อนำไปสร้างเงื่อนไขในการหาส่วนหัวของตัวพยัญชนะโดยสามารถแบ่งได้เป็น 3 ประเภทดังรูปด้านล่าง แต่มีตัวพยัญชนะที่ไม่อยู่ใน 3 ประเภทนี้คือตัวพยัญชนะที่ไม่มีหัวและตัวพยัญชนะที่มีหัวอยู่ระหว่างช่วงบนและช่วงกลาง ซึ่งตัวพยัญชนะเหล่านี้จะเกิดปัญหาขึ้นในขั้นตอนการหาส่วนหัว




**รูปที่ 4.19** แบบแผนแสดงตำแหน่งส่วนหัวของตัวพยัญชนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 4.20** ไดอะแกรมแสดงรูปแบบโปรแกรมหาส่วนหัวของตัวพยัญชนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.20 แสดงแนวคิดในการหาส่วนหัวของตัวพยัญชนะอย่างเป็นขั้นตอน ขั้นตอนที่ 1 การแยกตัวพยัญชนะที่มีหัวอยู่ช่วงบนออกมา แล้วเข้าเงื่อนไขหาส่วนหัวโดยใช้ขอบเขตช่วงบน ส่วนอีกกลุ่มเป็นตัวพยัญชนะที่มีหัวอยู่ช่วงกลางและช่วงล่าง ต้องนำตัวพยัญชนะในกลุ่มนี้เข้าขั้นตอนที่ 2 สามารถแยกได้เป็น 3 กลุ่มคือกลุ่มที่ 2.1, 2.2 และ 2.3 ซึ่งกลุ่มที่ 2.3 เป็นตัวพยัญชนะที่มีหัวอยู่ช่วงล่างจึงเข้าโปรแกรมหาส่วนหัวโดยใช้ขอบเขตช่วงล่างส่วนอีก 2 กลุ่มที่เหลือคือ 2.1และ 2.2 เป็นกลุ่มของตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงกลางและช่วงล่าง จึงต้องแยกทั้งสองช่วงนี้ออกจากกัน โดยกลุ่มที่ 2.2 สามารถแยกได้เป็น 3.1 เป็นตัวพยัญชนะที่มีหัวอยู่ช่วงกลางจึงเข้าเงื่อนไขหาส่วนหัวโดยใช้ขอบเขตช่วงกลาง และ 3.2 เป็นตัวพยัญชนะที่มีหัวอยู่ช่วงล่างจึงเข้าเงื่อนไขหาส่วนหัวโดยใช้ขอบเขตช่วงล่าง สำหรับกลุ่มที่ 2.1 สามารถแบ่งได้ 3 กลุ่ม คือ 4.1 หัวอยู่ช่วงกลาง, 4.2 หัวอยู่ช่วงล่างและ4.3 หัวอยู่ช่วงกลางและล่าง โดยกลุ่มที่ 4.1 เป็นตัวพยัญชนะที่มีหัวอยู่ช่วงกลางจึงเข้าเงื่อนไขหาส่วนหัวโดยใช้ขอบเขตช่วงกลาง กลุ่มที่ 4.2 เป็นตัวพยัญชนะที่มีหัวอยู่ช่วงล่างจึงเข้าเงื่อนไขหาส่วนหัวโดยใช้ขอบเขตช่วงล่าง และ 4.3 เป็นตัวพยัญชนะที่มีหัวอยู่ช่วงกลางและช่วงล่างจึงต้องแยกตัวพยัญชนะทั้งสองช่วงออกจากกันก่อน ซึ่งขั้นตอนต่างๆ ตามแผนภาพไดอะแกรมนี้ สามารถอธิบายรายละเอียดได้ดังนี้

1. แยกตัวพยัญชนะที่มีหัวอยู่ช่วงบน ออกจากตัวพยัญชนะที่มีหัวอยู่ช่วงกลางและช่วงล่างด้วยวิธีการหาความหนาของตัวพยัญชนะ โดยมีหลักการคือแบ่งความสูงของตัวพยัญชนะออกเป็น 3 ส่วนเท่าๆ กัน ให้เส้นแรกนับจากด้านบนเป็นเส้นสมมติ ดังรูปที่ 4.21

**รูปที่ 4.21** แสดงการหาเส้นสมมติ

เมื่อได้เส้นสมมติแล้ว จากนั้นหาความหนาที่มากที่สุดของตัวพยัญชนะทีละแถว แล้วเก็บแถว(row) ที่มีความหนามากที่สุดไว้ในตัวแปรตัวหนึ่ง ดังรูปที่ 4.22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**รูปที่ 4.22** แสดงแถว (row) ที่มีความหนามากที่สุดของตัวพยัญชนะ

เมื่อได้เส้นสมมติและแถวที่มีความหนามากที่สุดแล้ว จากนั้นนำมาเปรียบเทียบกันโดย ถ้าแถวที่มีความหนามากที่สุดอยู่ต่ำกว่าเส้นสมมติ จะเข้าโปรแกรมหาส่วนหัวของตัวพยัญชนะในขอบเขตช่วงบน ในทางตรงกันข้ามถ้าแถวที่มีความหนามากที่สุดอยู่ต่ำกว่าเส้นสมมติจะเข้าเงื่อนไขหัวอยู่ช่วงกลางและหัวอยู่ช่วงล่าง จากนั้นต้องแยกพยัญชนะที่มีส่วนหัวอยู่ช่วงกลางและช่วงล่างออกจากกัน

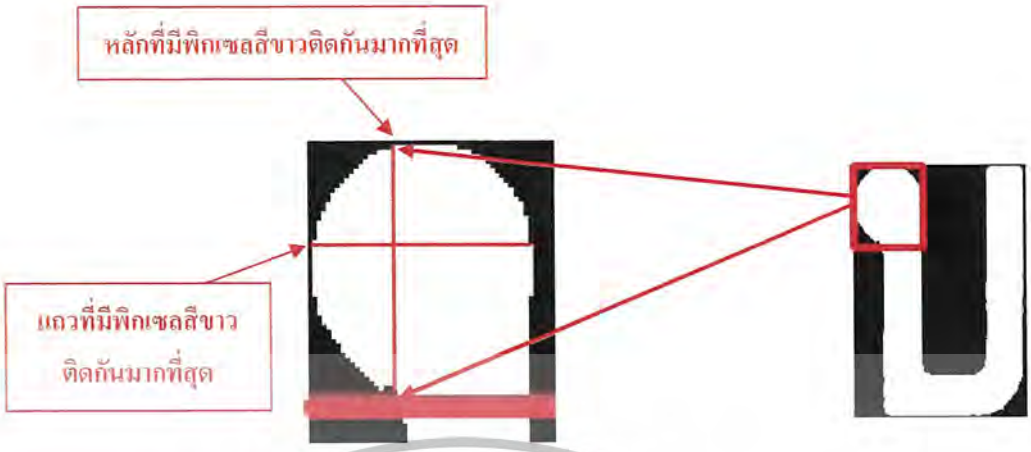
ดังนั้นถ้าแถวที่มีความหนามากที่สุดอยู่ต่ำกว่าเส้นสมมติ จะเข้าโปรแกรมหาส่วนหัวของตัวพยัญชนะในขอบเขตช่วงบน การหาส่วนหัวของตัวพยัญชนะที่มีหัวอยู่ช่วงบนมีขั้นตอนดังนี้

1.1) แบ่งความสูงของตัวพยัญชนะออกเป็น 3 ส่วนเท่าๆ กัน แล้วใช้ขอบเขตช่วงบน เพราะส่วนหัวของตัวพยัญชนะที่อยู่ช่วงบนจะอยู่ในขอบเขตนี้ด้วย แสดงดังรูปที่ 4.23

**รูปที่ 4.23** ขอบเขตที่ใช้ในการหาส่วนหัวของตัวพยัญชนะที่อยู่ช่วงบน

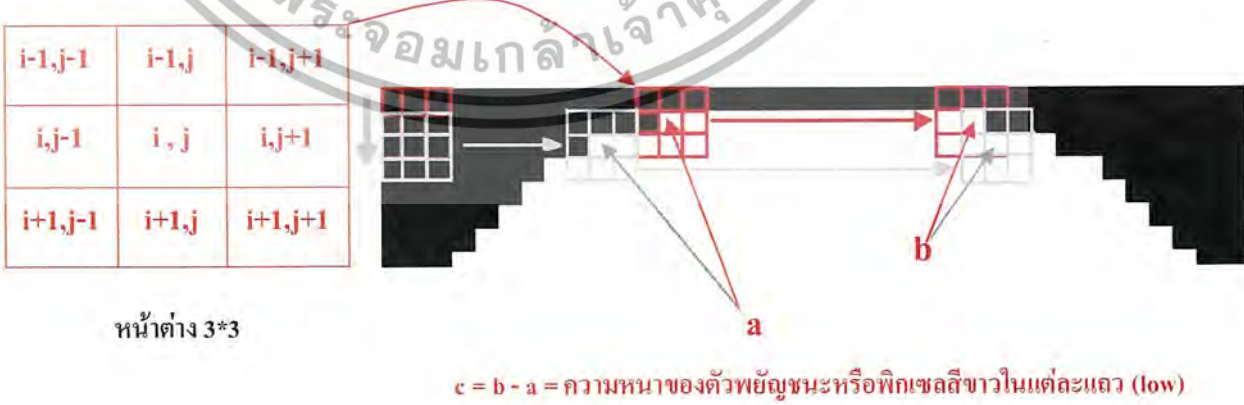
1.2) หาความหนาของตัวพยัญชนะหรือความหนาของพิกเซลสีขาวในขอบเขตช่วงบนทั้งในแนวตั้งและแนวนอนหรือหาความหนาทั้งแถว (row) และหลัก (column) โดยแถวและหลักที่มีความหนาของพิกเซลสีขาวมากที่สุดจะถูกตีกรอบไว้ แสดงดังรูปที่ 4.24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 4.24** ตีกรอบที่แถว (row) และหลัก (column) ที่มีพิกเซลสีขาวมากที่สุด

หลักการหาความหนาของตัวพยัญชนะหรือพิกเซลสีขาว ทั้งแนวตั้งและแนวนอนตามแถว (row) และหลัก (column) เราใช้หน้าต่าง 3*3 วางที่ตำแหน่งมุมเริ่มต้นของภาพ ตำแหน่ง [(i,j) หรือ (ตำแหน่ง2*2)] ของหน้าต่าง 3*3 จึงเริ่มตำแหน่งแรกที่แถวและหลักที่ 2 ของภาพ การหาความหนาของตัวพยัญชนะตามแนวนอนหรือแถว (row) นั้น ให้หน้าต่าง 3*3 เคลื่อนที่ทีละตำแหน่งจากซ้ายไปขวา การหาความหนาหรือหาจำนวนพิกเซลสีขาวที่อยู่ติดกันทำได้โดย เมื่อตำแหน่ง (i,j) เป็น 1 หรือเป็นสีขาว และ (i,j-1) เป็น 0 หรือสีดำให้เก็บค่าหลัก (column) นั้นไว้ในตัวแปร a จากนั้นหน้าต่าง 3x3 จะเคลื่อนที่จากซ้ายไปขวาจนถึงตำแหน่งที่ (i,j) เป็น 1 หรือสีขาว และ (i,j+1) เป็น 0 หรือสีดำให้เก็บค่าหลัก (column) นั้นไว้ในอีกตัวแปร b เมื่อได้ค่าในตัวแปร b แล้วนำ b – a = c จะได้ค่าความหนาของพิกเซลสีขาวที่อยู่ติดกันตามแนวนอนหรือแถว (row) นั้น ไว้ในตัวแปร c เมื่อหาแถวแรกแล้ว หน้าต่าง 3x3 จะเคลื่อนลงไปหาในแถวต่อไปจนครบ แสดงดังรูปที่ 4.25



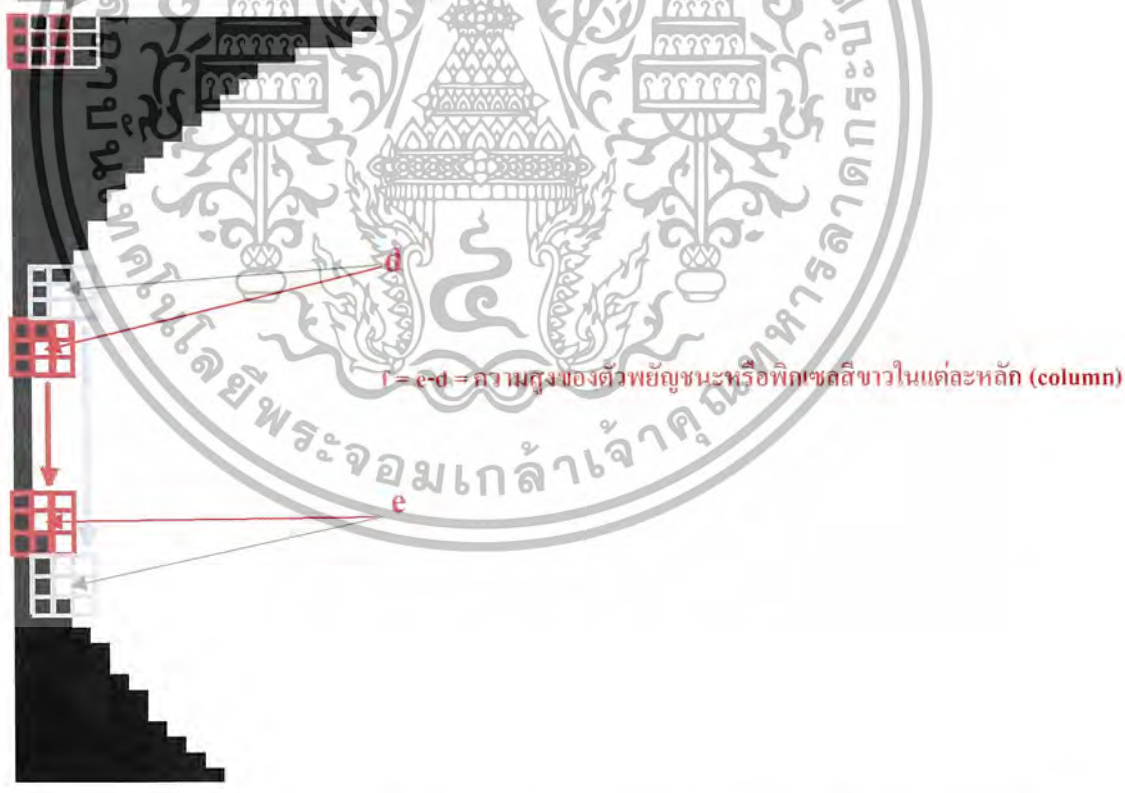

**รูปที่ 4.25** การใช้หน้าต่าง 3x3 ในการหาความหนาของกลุ่มพิกเซลสีขาวตามแถว (row)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อได้ความหนาหรือพิกัดหัวท้าย ของกลุ่มพิกเซลสีขาวที่อยู่ติดกันตามแถว (row) แล้ว เลือกแถวที่มีความหนามากที่สุดเป็นขอบซ้ายและขวาหรือความกว้างของกรอบนั่นเอง แสดงดังรูปที่ 4.26

**รูปที่ 4.26** ขอบเขตความกว้างของกรอบบริเวณส่วนหัวของตัวพยัญชนะ

สำหรับแนวตั้งหรือหลัก (column) ก็คิดในทำนองเดียวกันแต่เปลี่ยนเป็นการวนจากบนลงล่างเมื่อได้ความหนาหรือพิกัดหัวท้ายของกลุ่มพิกเซลสีขาวที่อยู่ติดกันตามหลัก (column) จึงเลื่อนไปหลักต่อไปจนครบทุกหลัก แสดงดังรูปที่ 4.27




**รูปที่ 4.27** การใช้หน้าต่าง 3x3 ในการหาความหนาของกลุ่มพิกเซลสีขาวตามหลัก (column)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อได้ความหนาหรือพิกัดหัวท้ายของกลุ่มพิกเซลสีขาวที่อยู่ติดกันตามหลัก (column) แล้ว เลือกหลัก (column) ที่มีความหนามากที่สุดเป็นขอบบนและล่างหรือความสูงของกรอบนั่นเอง ดังรูปที่ 4.28

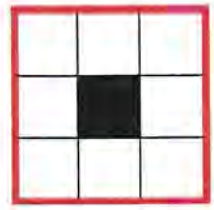
**รูปที่ 4.28** ขอบเขตความสูงของกรอบบริเวณส่วนหัวของตัวพยัญชนะ

1.3) นำความหนาตามแถว (row) และความสูงตามหลัก (column) มารวมกันเป็นกรอบที่ครอบส่วนหัวของตัวพยัญชนะไว้ ซึ่งเป็นตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงบน แสดงดังรูปที่ 4.29

**รูปที่ 4.29** ตัวอย่างตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงบน

1.4) แบ่งความกว้างและความสูงของกรอบสีเหลี่ยมเป็นสามส่วน ได้เป็นสี่เหลี่ยม 3x3 เปลี่ยนสี่เหลี่ยมตรงกลางเป็นพิกเซลสีดำ เกิดเป็นหัวของตัวพยัญชนะ ดังรูปที่ 4.30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 4.30** สี่เหลี่ยมขนาด 3x3 ที่สร้างขึ้นจากกรอบที่ตีบริเวณส่วนหัวของตัวพยัญชนะ มีสี่เหลี่ยมกึ่งกลางเป็นสีดำหรือเกิดเป็นหัวโปร่งของตัวพยัญชนะ

สำหรับเงื่อนไขแรกนี้เป็นพยัญชนะที่มีหัวอยู่ช่วงบนทั้งหมด คือ ง, น, บ, พ, ม, ย, ษ, ห และ ผ เมื่อหาส่วนหัวของตัวพยัญชนะแล้วใส่สี่เหลี่ยมสีดำลงไปที่ส่วนหัวของตัวพยัญชนะ ทำให้ตัวพยัญชนะมีลักษณะดังรูปที่ 4.31

**รูปที่ 4.31** ลักษณะของตัวพยัญชนะที่มีหัวอยู่ช่วงบนเมื่อใส่หัวให้กับตัวพยัญชนะแล้ว

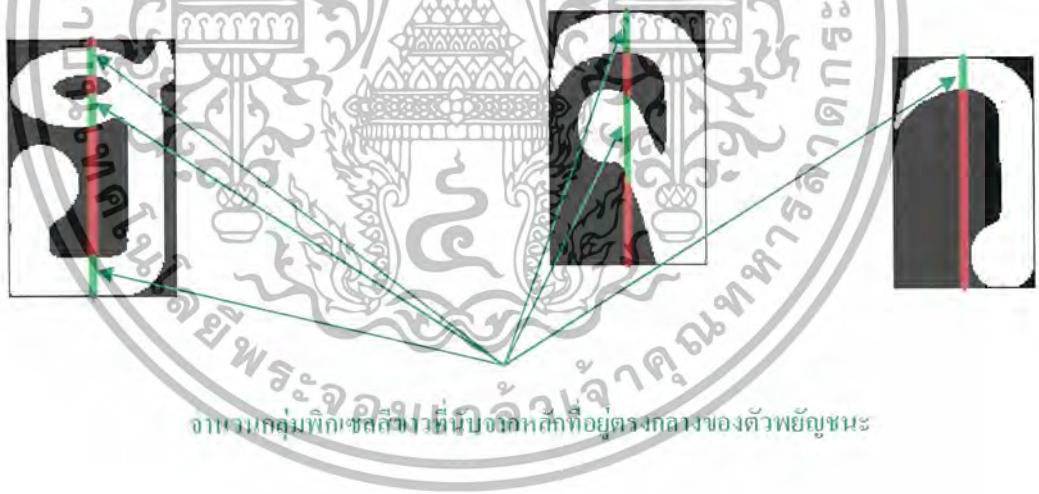
2. การหาส่วนหัวของพยัญชนะที่มีส่วนหัวอยู่ช่วงกลางและช่วงล่างของตัวพยัญชนะ ซึ่งมีตัวพยัญชนะคือ ค, จ, ฉ, ด, ต, ศ, อ, ฮ, ฌ, ญ, ฒ, ถ, ภ, ร, ล, ว, ส โดยมีขั้นตอนการหาส่วนหัวของตัวพยัญชนะเหล่านี้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1) เมื่อรับภาพของตัวพยัญชนะเข้ามา แบ่งความกว้างหรือจำนวนหลัก (column) ของตัวพยัญชนะออกเป็นสองส่วนเท่าๆ กัน จะได้หลักที่อยู่ตรงกลางของตัวพยัญชนะ ดังรูปที่ 4.32

**รูปที่ 4.32** แสดงหลัก (column) ที่อยู่ตรงกลางของตัวพยัญชนะ

2.2) สำรวจจำนวนกลุ่มพิกเซลสีขาวบนหลัก (column) ที่อยู่ตรงกลางของตัวพยัญชนะ โดยสำรวจจากบนลงล่าง ซึ่งพิกเซลสีขาวที่อยู่ติดกันจะนับเป็นกลุ่มเดียวกันจากวิธีนี้เราสามารถแบ่งตัวพยัญชนะออกได้เป็น 3 ชนิด คือ จำนวนพิกเซลสีขาว 1 กลุ่ม 2 กลุ่ม และ 3 กลุ่ม ดังรูปที่ 4.33




**รูปที่ 4.33** แสดงวิธีการหากลุ่มพิกเซลสีขาวในหลัก (column) ตรงกลางตัวพยัญชนะ

1. มีกลุ่มพิกเซลสีขาว 1 กลุ่ม คือ ถ, ว, ณ
2. มีกลุ่มพิกเซลสีขาว 2 กลุ่ม คือ ฉ, จ, ค, ด, ต, ศ, ภ, ส, ล
3. มีกลุ่มพิกเซลสีขาว 3 กลุ่ม คือ ร, อ, ฮ

แสดงตัวอย่างตัวพยัญชนะทั้ง 3 กลุ่ม ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. มีกลุ่มพิกเซลสีขาว 1 กลุ่ม คือ ถ, ว, ณ, ฌ

2. มีกลุ่มพิกเซลสีขาว 2 กลุ่ม คือ ฉ, จ, ค, ด, ต, ศ, ภ, ส, ล

3. มีกลุ่มพิกเซลสีขาว 3 กลุ่ม คือ ร, อ, ฮ

2.3) การสร้างเงื่อนในการหาส่วนหัวของตัวพยัญชนะ คือเมื่อสำรวจจำนวนกลุ่มพิกเซลสีขาวตามขั้นตอนที่ 2 แล้ว ถ้ามีกลุ่มพิกเซลสีขาวอยู่ 1 กลุ่ม ให้เข้าโปรแกรมหาส่วนหัวของตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงล่าง โดยมีวิธีหาส่วนหัวคล้ายกลับการหาส่วนหัวของตัวพยัญชนะที่มีหัวอยู่ช่วงบนของตัวพยัญชนะ แต่เปลี่ยนจากขอบเขตช่วงบนเป็นขอบเขตช่วงล่าง แสดงดังรูปที่ 4.34

**รูปที่ 4.34** ขอบเขตการหาหัวของตัวพยัญชนะที่มีหัวอยู่ช่วงล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหาส่วนหัวของตัวพยัญชนะในขอบเขตช่วงล่างมีขั้นตอนตามรูปที่ 4.35 ด้านล่างนี้

**รูปที่ 4.35** การหาส่วนหัวของตัวพยัญชนะที่มีหัวอยู่ช่วงล่าง

ขั้นตอนที่ 1 ใช้ช่วงล่างเป็นขอบเขตในการหาส่วนหัวของตัวพยัญชนะ

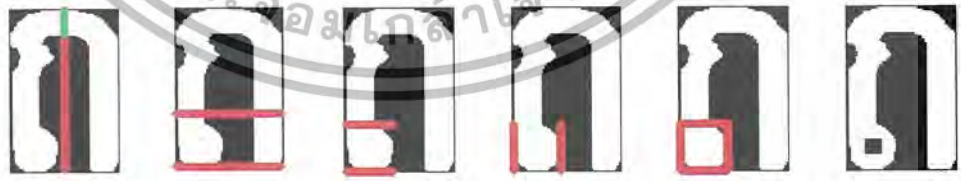
ขั้นตอนที่ 2 หาความหนาหรือพิกัดหัวท้ายของกลุ่มพิกเซลสีขาวที่อยู่ติดกันมากที่สุดตามหลัก (column) ได้ขอบบนและล่างหรือความสูงของกรอบ

ขั้นตอนที่ 3 หาความหนาหรือพิกัดหัวท้ายของกลุ่มพิกเซลสีขาวที่อยู่ติดกันมากที่สุดตามแถว (row) ได้ขอบซ้ายและขวาหรือความกว้างของกรอบ

ขั้นตอนที่ 4 นำความหนาตามแถว (row) และความสูงตามหลัก (column) มารวมกันเป็นกรอบครอบส่วนหัวของตัวพยัญชนะ ซึ่งเป็นส่วนหัวที่อยู่ช่วงล่างของตัวพยัญชนะ

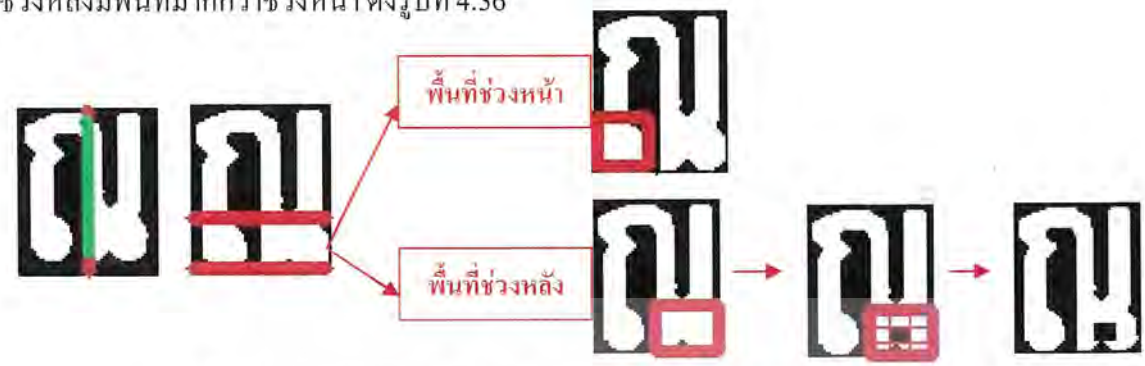
ขั้นตอนที่ 5 สร้างสี่เหลี่ยมขนาด 3x3 จากกรอบที่ตีบริเวณส่วนหัวของตัวพยัญชนะ เปลี่ยนสี่เหลี่ยมกึ่งกลางเป็นสีดำจึงเกิดเป็นหัวของตัวพยัญชนะ

โดยตัวพยัญชนะที่เข้าเงื่อนไขนี้ ได้แก่ ตัวพยัญชนะ ถ, ว, ฌ แสดงดังรูปด้านล่าง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับตัวพยัญชนะ ณ.เณร จะได้ช่วงหลังแทนช่วงหน้า เนื่องจากเราเลือกพื้นที่ที่มากที่สุด ซึ่งช่วงหลังมีพื้นที่มากกว่าช่วงหน้า ดังรูปที่ 4.36




**รูปที่ 4.36** ตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงล่างแต่มีลักษณะพิเศษ

2.4 การสร้างเงื่อนไขในกรณีที่มีกลุ่มพิกเซลสีขาวจำนวน 3 กลุ่ม ซึ่งทำให้ได้พยัญชนะที่มีหัวอยู่ช่วงกลางและหัวอยู่ช่วงล่าง ด้วยเหตุนี้เราจึงต้องแยกตัวพยัญชนะที่มีหัวอยู่ช่วงกลางกับตัวพยัญชนะที่มีหัวอยู่ช่วงล่างออกจากกัน โดยการแบ่งความสูงของตัวพยัญชนะออกเป็นสองส่วนเท่าๆ กัน จะได้แถว (row) ที่อยู่ตรงกลางของตัวพยัญชนะนั้น แสดงดังรูปที่ 4.37

**รูปที่ 4.38** แบ่งความสูงของตัวพยัญชนะออกเป็นสองส่วนเท่าๆกันได้แถวตรงกลาง

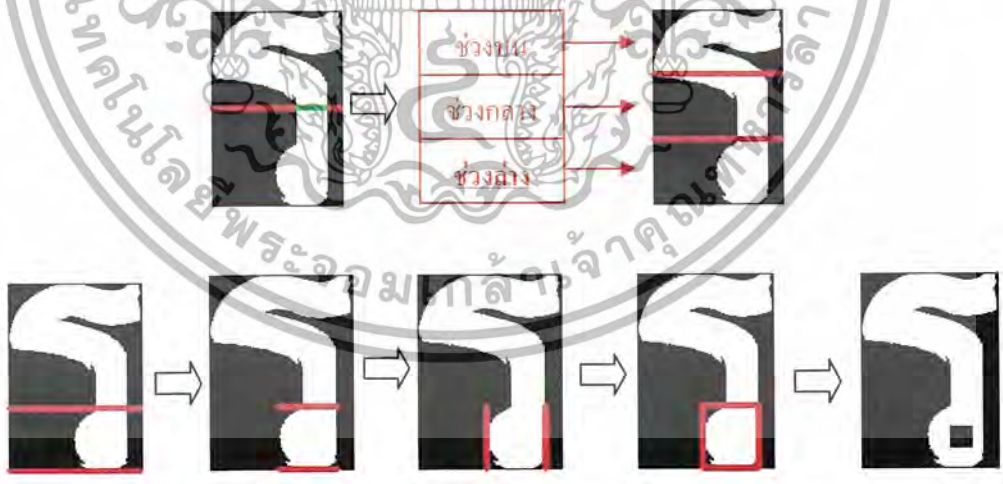
จากนั้นทำการสำรวจหากลุ่มพิกเซลสีขาวตามแถว (row) ที่แบ่งครึ่งตัวพยัญชนะโดยสำรวจทุกหลัก (column) จากซ้ายไปขวา ทำให้ได้กลุ่มพิกเซลสีขาวสองประเภทคือ มีกลุ่มพิกเซลสีขาว 1 กลุ่ม และมีกลุ่มพิกเซลสีขาว 2 กลุ่ม จากผลดังกล่าวทำให้สามารถแยกตัวพยัญชนะที่มีตำแหน่งของหัวอยู่ช่วงกลางและช่วงล่างออกจากกันได้ คือ กลุ่มพิกเซลสีขาว 1 กลุ่ม คือตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงล่าง ได้แก่ตัวพยัญชนะ ร.เรือ และกลุ่มพิกเซลสีขาว 2 กลุ่ม คือตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงกลาง ได้แก่ตัวพยัญชนะ อ.อ่าง และตัวพยัญชนะ ฮ.นกฮูก ดังรูปที่ 4.39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**รูปที่ 4.39** การหากลุ่มพิกเซลสีขาวตามแถว (row) ที่แบ่งครึ่งตัวพยัญชนะ

จากวิธีนี้ได้ 2 กรณี คือ กรณีที่ 1 คือตัวพยัญชนะมีกลุ่มพิกเซลสีขาว 1 กลุ่ม จะหาส่วนหัวของตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงล่าง และกรณีที่ 2 คือตัวพยัญชนะมีกลุ่มพิกเซลสีขาว 2 กลุ่ม จะหาส่วนหัวของตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงกลาง การหาส่วนหัวของตัวพยัญชนะและการใส่หัวของทั้งสองกรณีมีดังนี้

3.1) กรณีที่ 1 ตัวพยัญชนะมีกลุ่มพิกเซลสีขาว 1 กลุ่ม หาส่วนหัวของตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงล่าง โดยใช้ขอบเขตช่วงล่างหาความหนาหรือคู่พิกัดหัวท้ายของกลุ่มพิกเซลสีขาวที่อยู่ติดกันมากที่สุดตามแถว (row) และหลัก (column) เมื่อรวมขอบเขตบน-ล่างและซ้าย-ขวา เข้าด้วยกัน ได้พื้นที่ที่มากที่สุดเป็นส่วนหัวของตัวพยัญชนะและสร้างสี่เหลี่ยมขนาด 3x3 จากกรอบที่ตีบริเวณส่วนหัวของตัวพยัญชนะ เปลี่ยนสี่เหลี่ยมกึ่งกลางเป็นสีดำจึงเกิดเป็นหัวของตัวพยัญชนะ ดังรูปที่ 4.40

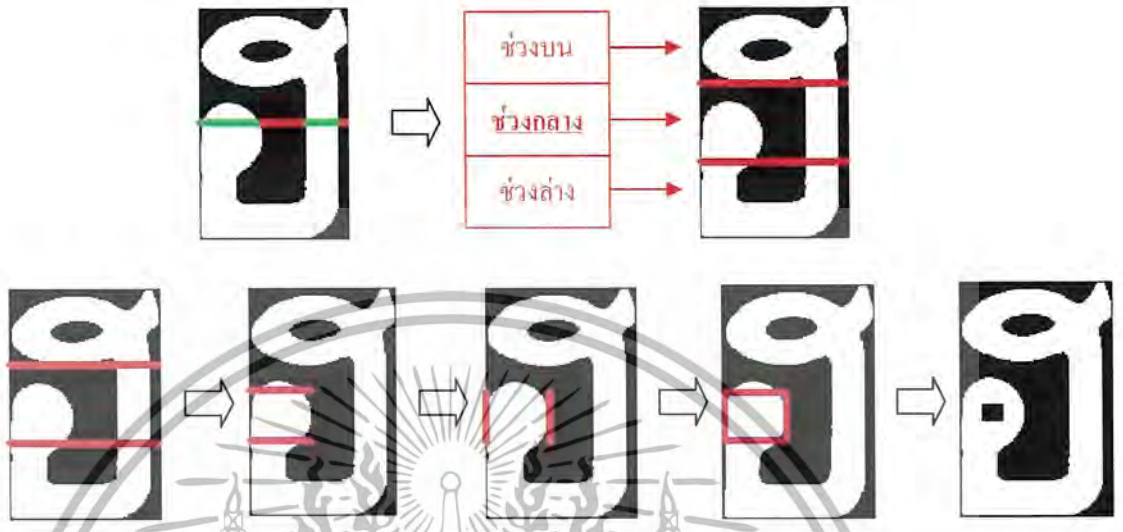



**รูปที่ 4.40** ขั้นตอนการหาส่วนหัวของตัวพยัญชนะที่มีหัวอยู่ช่วงล่าง

3.2) กรณีที่ 2 ตัวพยัญชนะมีกลุ่มพิกเซลสีขาว 2 กลุ่ม หาส่วนหัวของตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงกลาง โดยใช้ขอบเขตช่วงกลางหาความหนาหรือคู่พิกัดหัวท้ายของกลุ่มพิกเซลสีขาวที่อยู่ติดกันมากที่สุดตามแถว (row) และหลัก (column) เมื่อรวมขอบเขตบน-ล่างและซ้าย-ขวา ได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พื้นที่ที่มากที่สุดเป็นส่วนหัวของตัวพยัญชนะและสร้างสี่เหลี่ยมขนาด 3x3จากกรอบที่ตีบริเวณส่วนหัวของตัวพยัญชนะ เปลี่ยนสี่เหลี่ยมกึ่งกลางเป็นสีดำจึงเกิดเป็นหัวของตัวพยัญชนะดังรูปที่ 4.41



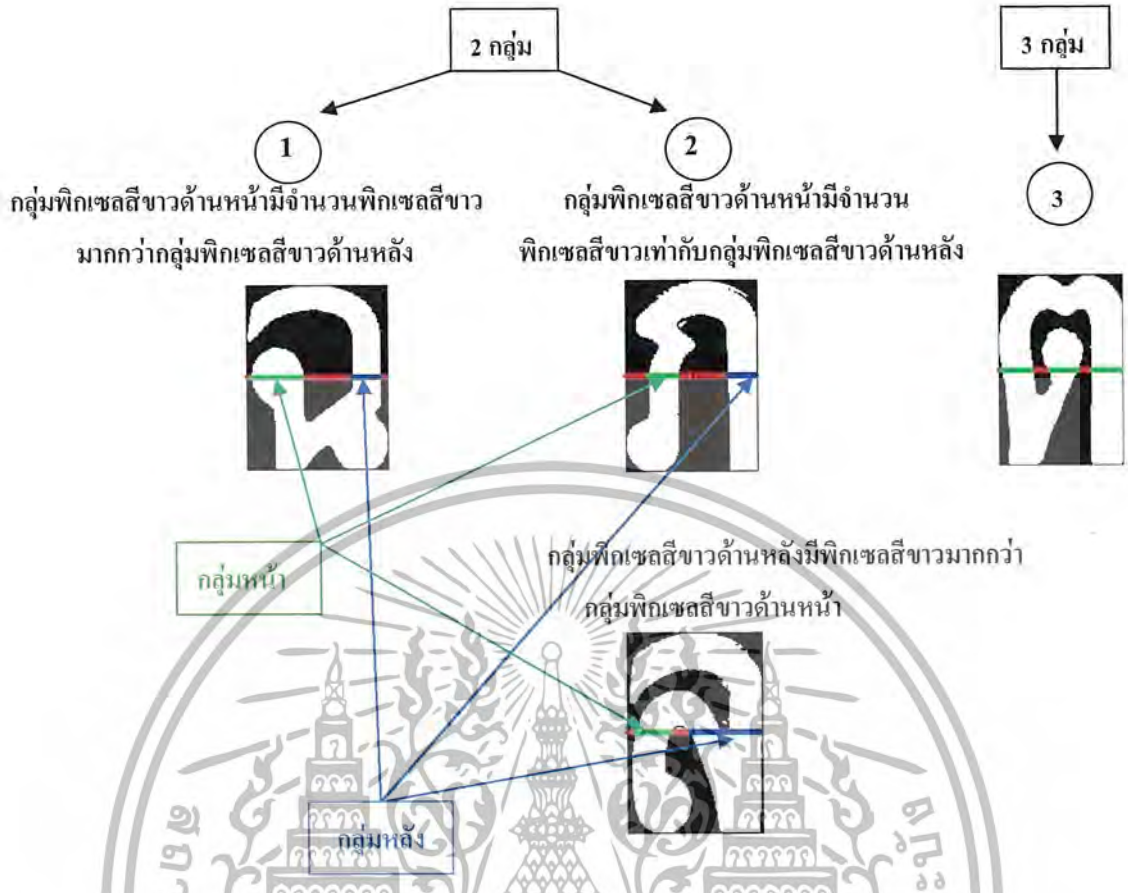

**รูปที่ 4.41** ขั้นตอนการหาส่วนหัวของตัวพยัญชนะที่มีหัวอยู่ช่วงกลาง

2.5 สร้างเงื่อนไขในการหาส่วนหัวของตัวพยัญชนะ กรณีที่มีกลุ่มพิกเซลสีขาวอยู่ 2 กลุ่ม ซึ่งมีตัวพยัญชนะที่มีหัวอยู่ช่วงกลางและหัวอยู่ช่วงล่าง จึงต้องแยกตัวพยัญชนะที่มีหัวอยู่ช่วงกลางออกจากพยัญชนะที่มีหัวอยู่ช่วงล่าง ตัวพยัญชนะที่เข้าเงื่อนไขนี้มี ฉ, จ, ค, ด, ต, ศ, ภ, ส และ ล โดยแบ่งความสูงของตัวพยัญชนะออกเป็นสองส่วนเท่าๆ กัน จะได้แถว (row) ที่แบ่งครึ่งความสูงของตัวพยัญชนะนั้นๆ ดังรูปที่ 4.42

**รูปที่ 4.42** การแบ่งความสูงของตัวพยัญชนะออกเป็นสองส่วน

เมื่อได้แถว (row) ที่แบ่งครึ่งความสูงของตัวพยัญชนะแล้ว จากนั้นทำการสำรวจจำนวนกลุ่มพิกเซลสีขาวที่อยู่ในแถว (row) กลางนี้ทั้งหมด ผลที่ได้คือมีกลุ่มพิกเซลสีขาว 2 กลุ่ม และ 3 กลุ่ม โดยกลุ่มพิกเซลสีขาว 2 กลุ่มสามารถแบ่งได้อีกเป็น (1) กลุ่มพิกเซลสีขาวด้านหน้ามากกว่ากลุ่มพิกเซลสีขาวด้านหลัง และ (2) กลุ่มพิกเซลสีขาวด้านหลังมากกว่าหรือเท่ากับกลุ่มพิกเซลสีขาวด้านหน้า ดังรูปที่ 4.43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 4.43** ผลที่ได้จากการสำรวจจำนวนกลุ่มพิกเซลสีขาว

จากรูปที่ 4.43 สามารถแบ่งเงื่อนไขการหาส่วนหัวของตัวพยัญชนะออกเป็น 3 กลุ่มด้วยกัน คือ

1. พบกลุ่มพิกเซลสีขาว 2 กลุ่ม โดยกลุ่มพิกเซลสีขาวด้านหน้ามีจำนวนพิกเซลสีขาวมากกว่ากลุ่มพิกเซลสีขาวด้านหลัง ซึ่งตัวพยัญชนะที่เข้าเงื่อนไขนี้มีตัวพยัญชนะ ฉ, จ, ค, ศ

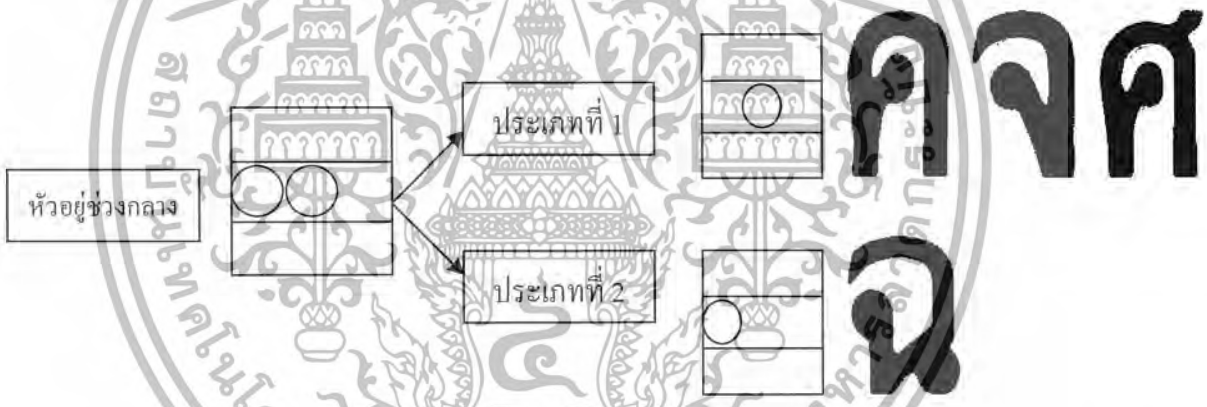
2. พบกลุ่มพิกเซลสีขาว 2 กลุ่ม โดยกลุ่มพิกเซลสีขาวด้านหน้ามีจำนวนพิกเซลสีขาวน้อยกว่าหรือเท่ากับกลุ่มพิกเซลสีขาวด้านหลัง ซึ่งตัวพยัญชนะที่เข้าเงื่อนไขนี้มีตัวพยัญชนะ ภ, ล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. พบกลุ่มพิกเซลสีขาว 3 กลุ่ม โดยตัวพยัญชนะที่เข้าเงื่อนไขนี้มีตัวพยัญชนะ ค, ต และ ส

จากเงื่อนไขทั้ง 3 ข้อสามารถอธิบายวิธีการหาส่วนหัวของตัวพยัญชนะได้ดังนี้

2.5.1 กรณีมีกลุ่มพิกเซลสีขาว 2 กลุ่ม โดยกลุ่มหน้ามีจำนวนพิกเซลสีขาวมากกว่ากลุ่มหลัง ซึ่งตัวพยัญชนะทั้งหมดที่เข้าเงื่อนไขนี้เป็นตัวพยัญชนะที่มีหัวอยู่ช่วงกลาง การหาส่วนหัวและใส่หัวให้กับตัวพยัญชนะนั้น ต้องแยกออกเป็นสองประเภทเนื่องจากตำแหน่งส่วนหัวของตัวพยัญชนะที่ต่างกัน คือประเภทที่ 1 หัวอยู่กึ่งกลางของตัวพยัญชนะคือตัวพยัญชนะ ค, จ, ศ และประเภทที่ 2. หัวอยู่ช่วงกลางด้านหน้าของตัวพยัญชนะคือตัวพยัญชนะ ฉ

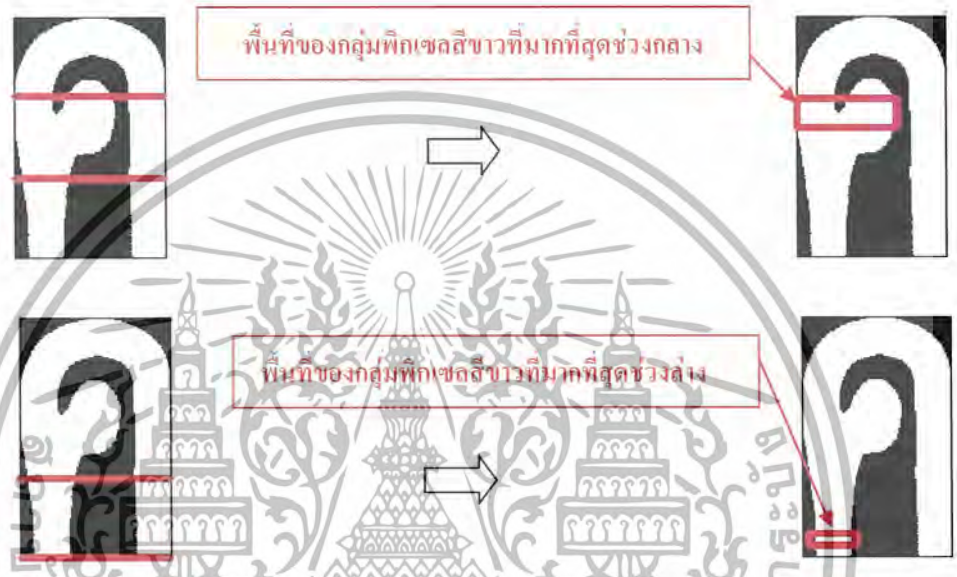



เนื่องจากตัวพยัญชนะทั้งสองประเภทนี้ ใช้ขอบเขตในการหาส่วนหัวของตัวพยัญชนะที่ต่างกัน จึงต้องแยกตัวพยัญชนะทั้งสองประเภทนี้ออกจากกันก่อน ด้วยวิธีการหาพื้นที่ของพิกเซลสีขาวที่มากที่สุดระหว่างช่วงกลางกับช่วงล่างของตัวพยัญชนะ โดยมีเงื่อนไขว่าถ้าพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดของช่องกลางมีค่ามากกว่ากลุ่มพิกเซลสีขาวที่มากที่สุดของช่วงล่าง จะใช้ขอบเขตประเภทที่ 1 หาส่วนหัวของตัวพยัญชนะ แต่ถ้าพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดของช่องกลางมีค่าน้อยกว่ากลุ่มพิกเซลสีขาวที่มากที่สุดของช่วงล่าง จะใช้ขอบเขตประเภทที่ 2 หาส่วนหัวของตัวพยัญชนะ โดยเงื่อนไขนี้เกิดจากการทดลองหาพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดในช่วงกลางและช่วงล่างพบว่า ตัวพยัญชนะที่อยู่ในกลุ่มนี้ส่วนใหญ่มีพื้นที่ของกลุ่มพิกเซลสีขาวช่วงกลางมากกว่าช่วงล่าง เนื่องจากส่วนหัวของตัวพยัญชนะอยู่ช่วงกลาง ยกเว้น ตัวพยัญชนะ ฉ.ฉิ่ง ที่มีพื้นที่ของกลุ่มพิกเซลสีขาวช่วงล่างมากกว่าช่วงกลางเพราะว่าตัวพยัญชนะ ฉ.ฉิ่ง มีสองหัวอยู่ทั้งช่วงกลางและช่วงล่างและพบว่า ฉ.ฉิ่ง มีตำแหน่งของหัวที่แต่งต่างจากพยัญชนะตัวอื่น คือมีส่วนหัวอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในตำแหน่งช่วงกลางด้านหน้า จากความแตกต่างนี้เราจึงนำมาสร้างเงื่อนไขการแยกตัวพยัญชนะทั้งสองประเภทนี้ออกจากกัน โดยขั้นตอนการหาพื้นที่ของกลุ่มพิกเซลสีขาวในช่วงกลางและช่วงล่างของตัวพยัญชนะทั้งสองประเภทดูจากรูปที่ 4.44 และ 4.45 ด้านล่าง

**ประเภทที่ 1.ส่วนหัวอยู่กึ่งกลางของตัวพยัญชนะ**



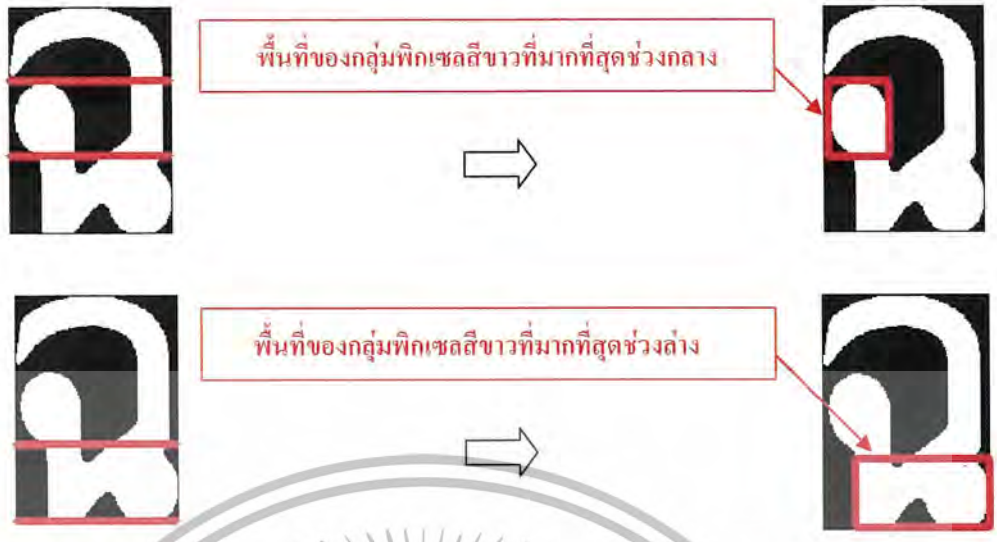

**รูปที่ 4.44** การหาพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดในช่วงกลางและช่วงล่างของตัวพยัญชนะโดยพื้นที่ในช่วงกลางมีค่ามากกว่าช่วงล่าง

จากรูปที่ 4.44 พบว่าพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดช่วงกลางมีค่ามากกว่าช่วงล่าง จึงเข้าเงื่อนไขหาส่วนหัวประเภทที่ 1 คือส่วนหัวอยู่กึ่งกลางของตัวพยัญชนะ โดยตัวพยัญชนะที่เข้าเงื่อนไขนี้มีตัวพยัญชนะ ค, จ และ ศ

**ประเภทที่ 2.ส่วนหัวอยู่ช่วงกลางด้านหน้าของตัวพยัญชนะ**

ถ้าพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดช่วงล่าง มีค่ามากกว่าพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดช่วงกลาง จะเข้าเงื่อนไขหาส่วนหัวประเภทที่ 2 คือส่วนหัวอยู่ช่วงกลางด้านหน้าของตัวพยัญชนะ โดยตัวพยัญชนะที่เข้าเงื่อนไขนี้มีตัวพยัญชนะ ฉ แสดงดังรูปที่ 4.45

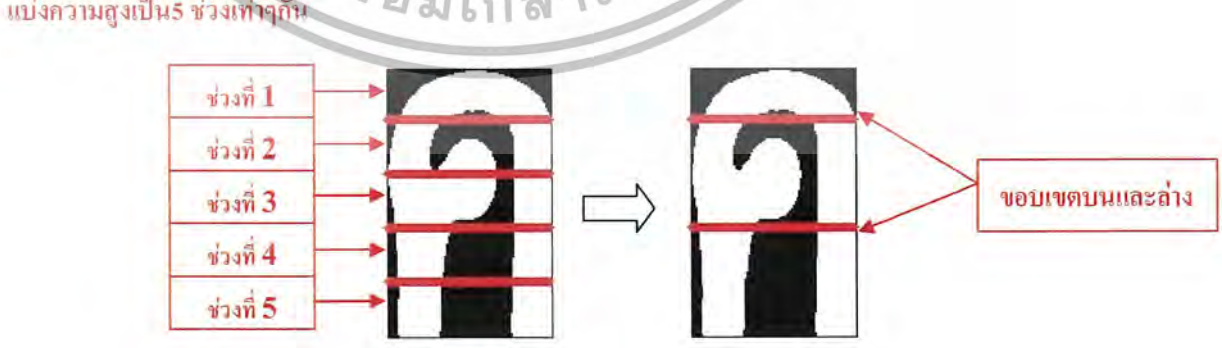
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



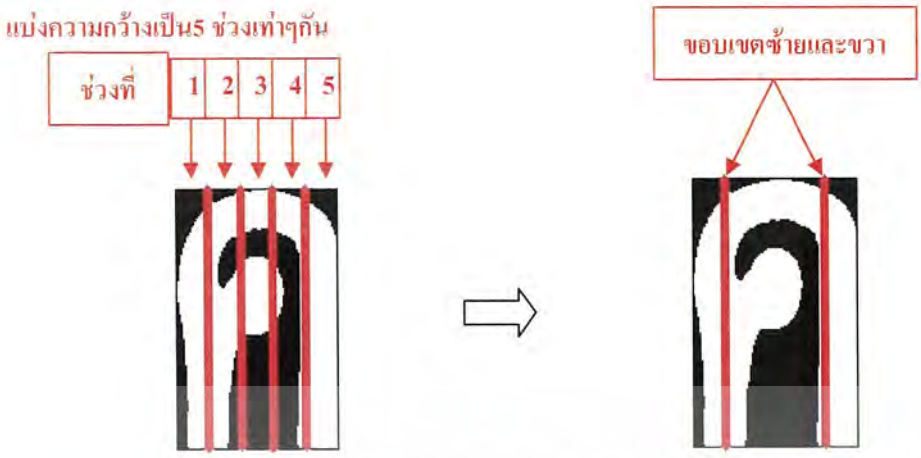

**รูปที่ 4.45** การหาพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดในช่วงกลางและช่วงล่างของตัวพยัญชนะโดยพื้นที่ในช่วงกลางมีค่าน้อยกว่าช่วงล่าง

เมื่อแยกพยัญชนะทั้งสองประเภทได้แล้ว ต้องหาส่วนหัวของตัวพยัญชนะทั้งสองประเภทนี้ โดยใช้ขอบเขตที่แต่งต่างกัน อธิบายได้ดังนี้

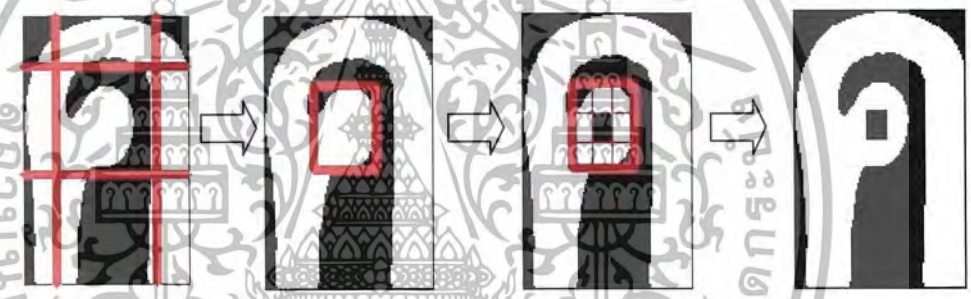
**ประเภทที่ 1** หัวอยู่กึ่งกลางของตัวพยัญชนะขอบเขตที่ใช้ ได้จากการนำความสูงและความกว้างของตัวพยัญชนะแบ่งเป็นห้าช่วงเท่าๆ กัน จากนั้นขอบเขตบนและล่างใช้ช่วงที่สองถึงสาม และขอบเขตซ้ายและขวาใช้ช่วงที่สองถึงสี่ จากนั้นหาส่วนหัวของตัวพยัญชนะโดยการหาพิกัดหัวท้ายของแถว (row) และหลัก (column) ที่มีความกว้างและความสูงของกลุ่มพิกเซลสีขาวที่มากที่สุดในขอบเขตนี้ มีขั้นตอนดังรูปที่ 4.46




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

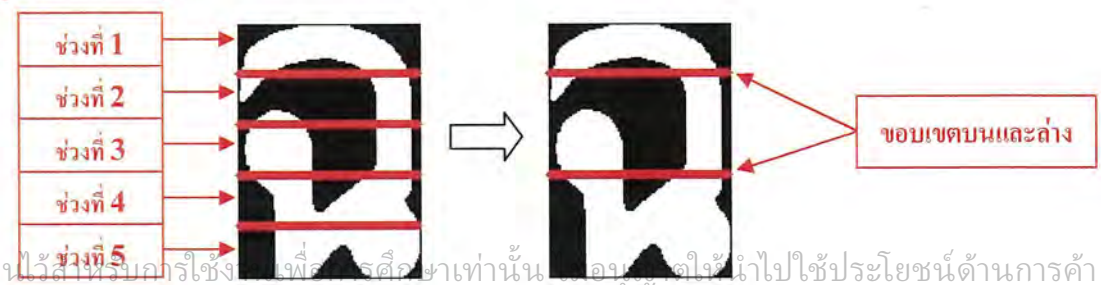



รวมขอบเขตบน-ล่างและซ้าย-ขวา หาพิกัดหัวท้ายของแถว (row) และหลัก (column) ที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตนี้ แล้วตีกรอบได้ส่วนหัวของตัวพยัญชนะ จากนั้นสร้างสี่เหลี่ยม 3x3 ในกรอบนี้แล้วเปลี่ยนสี่เหลี่ยมตรงกึ่งกลางเป็นสีดำ จึงเกิดเป็นหัวของตัวพยัญชนะที่หัวอยู่กึ่งกลาง



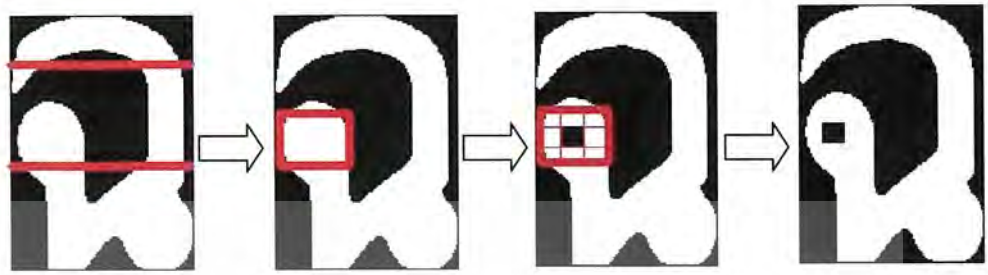
**รูปที่ 4.46** การหาส่วนหัวและใส่หัวให้กับตัวพยัญชนะและแสดงการหาขอบเขตของเงื่อนไขที่ 1 ประเภทที่ 1

**ประเภทที่ 2** หัวอยู่ช่วงกลางด้านหน้าของตัวพยัญชนะ ขอบเขตที่ใช้ ได้จากการนำความสูงของตัวพยัญชนะแบ่งเป็นห้าช่วงเท่าๆ กัน จากนั้นขอบเขตบนและล่างใช้ช่วงที่สองถึงสาม และขอบเขตซ้ายและขวาใช้ทั้งภาพ จากนั้นหาส่วนหัวของตัวพยัญชนะด้วยวิธีหาคู่พิกัดบนแถว (row) และหลัก (column) ที่มีความกว้างและความสูงของกลุ่มพิกเซลสีขาวที่อยู่ติดกันมากที่สุดในขอบเขตนี้ มีขั้นตอนดังรูปที่ 4.47



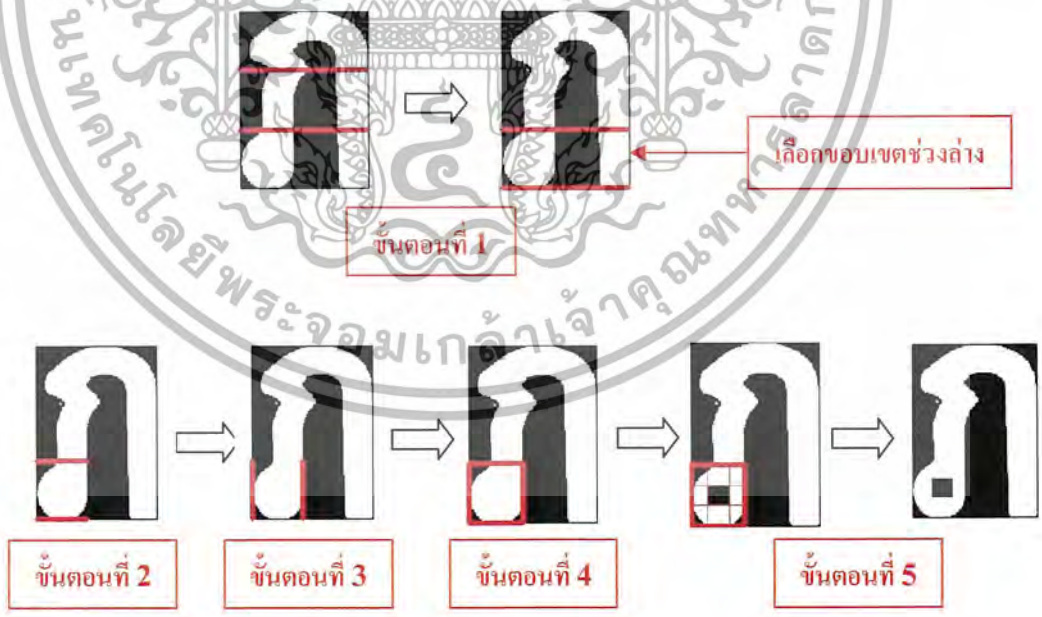

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หาพิกัดของแถว (row) และหลัก (column) ที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตนี้ แล้วตีกรอบได้ส่วนหัวของตัวพยัญชนะ จากนั้นสร้างสี่เหลี่ยม 3x3 ในกรอบนี้แล้วเปลี่ยนสี่เหลี่ยมตรงกึ่งกลางเป็นสีดำ จึงเกิดเป็นหัวของตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงกลางด้านหน้า



**รูปที่ 4.47** การหาส่วนหัวและใส่หัวให้กับตัวพยัญชนะและแสดงขอบเขตของเงื่อนไขที่ 1 ประเภทที่ 2

2.5.2 กรณีมีกลุ่มพิกเซลสีขาว 2 กลุ่ม โดยกลุ่มหน้ามีจำนวนพิกเซลสีขาวน้อยกว่าหรือเท่ากับกลุ่มหลัง โดยตัวพยัญชนะที่เข้าเงื่อนไขนี้มีส่วนหัวอยู่ในช่วงล่างของตัวพยัญชนะ จึงหาส่วนหัวของตัวพยัญชนะ โดยใช้ขอบเขตช่วงล่างในการหาพิกัดหัวท้ายของแถว (row) และหลัก (column) ที่มีพิกเซลสีขาวอยู่ติดกันมากที่สุด ซึ่งตัวพยัญชนะที่เข้าเงื่อนไขนี้มีตัวพยัญชนะ ภ และ ฦ มีขั้นตอนดังรูปที่ 4.48




**รูปที่ 4.48** การหาส่วนหัวและใส่หัวให้กับตัวพยัญชนะ ที่มีส่วนหัวของตัวพยัญชนะอยู่ในช่วงล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 1 เลือกช่วงล่างเป็นขอบเขตในการหาส่วนหัวของตัวพยัญชนะ

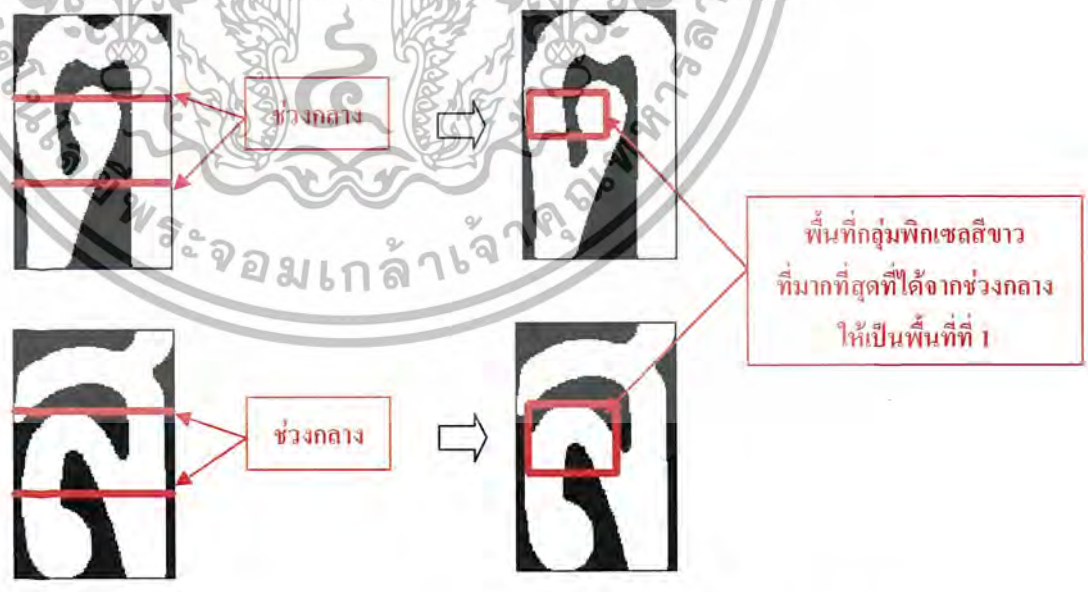
ขั้นตอนที่ 2 หาความหนาหรือพิกัดหัวท้ายตามหลัก (column) ที่มีพิกเซลสีขาวอยู่ติดกันมากที่สุด ได้ขอบบนและล่างหรือความสูงของกรอบ

ขั้นตอนที่ 3 หาความหนาหรือพิกัดหัวท้ายตามแถว (row) ที่มีพิกเซลสีขาวอยู่ติดกันมากที่สุด ได้ขอบซ้ายและขวาหรือความกว้างของกรอบ

ขั้นตอนที่ 4 รวมขอบบน-ล่างและซ้าย-ขวาได้เป็นกรอบที่ครอบส่วนหัวของตัวพยัญชนะ

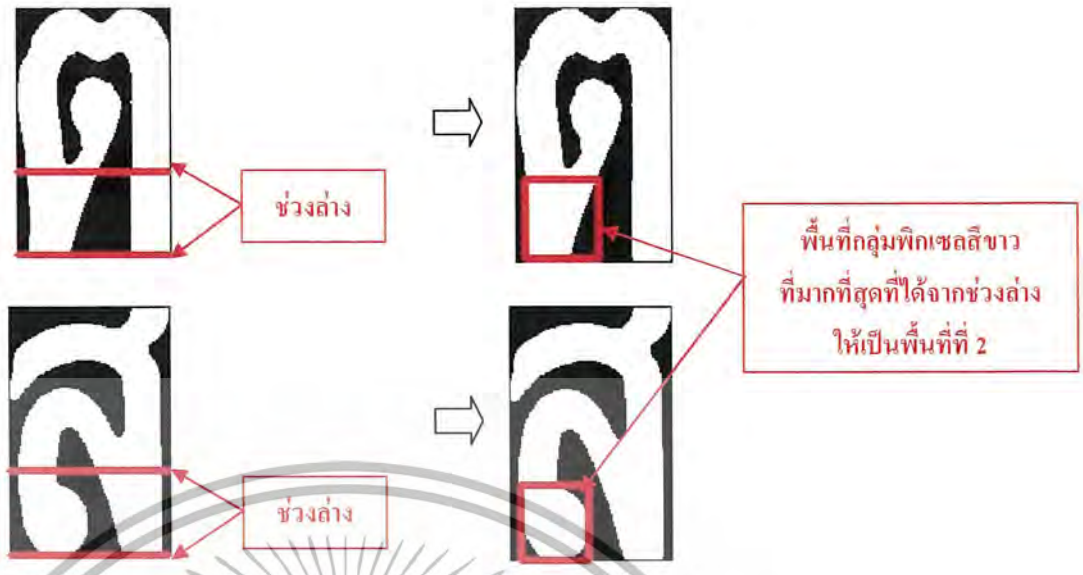
ขั้นตอนที่ 5 สร้างสี่เหลี่ยมขนาด 3x3 จากกรอบที่ตีบริเวณส่วนหัวของตัวพยัญชนะ เปลี่ยนสี่เหลี่ยมกึ่งกลางเป็นสีดำ จึงเกิดเป็นหัวของตัวพยัญชนะ

2.5.3 กรณีมีกลุ่มพิกเซลสีขาว 3 กลุ่ม สามารถแยกประเภทของตัวพยัญชนะตามตำแหน่งของส่วนหัวของตัวพยัญชนะเหมือนกับในข้อที่ 2.5.1 คือหาพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดในช่วงกลางและช่วงล่างของตัวพยัญชนะ โดยให้พื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดในช่วงกลางเป็นพื้นที่ที่ 1 และพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดในช่วงล่างเป็นพื้นที่ที่ 2 สำหรับเงื่อนไขที่ใช้ในการแยกประเภทของตัวพยัญชนะออกจากกันใช้พื้นที่ที่ 1 และที่ 2 มาเป็นตัวแปร โดยมีเงื่อนไขว่าถ้าพื้นที่ที่ 1 น้อยกว่าพื้นที่ที่ 2 หาส่วนหัวของตัวพยัญชนะที่หัวอยู่ช่วงกลางแต่ถ้าพื้นที่ที่ 2 มีค่าน้อยกว่าพื้นที่ที่ 1 หาส่วนหัวของตัวพยัญชนะที่หัวอยู่ช่วงล่างแทน โดยการหาพื้นที่ที่ 1 และที่ 2 แสดงในรูปที่ 4.49 และ 4.50 ด้านล่าง



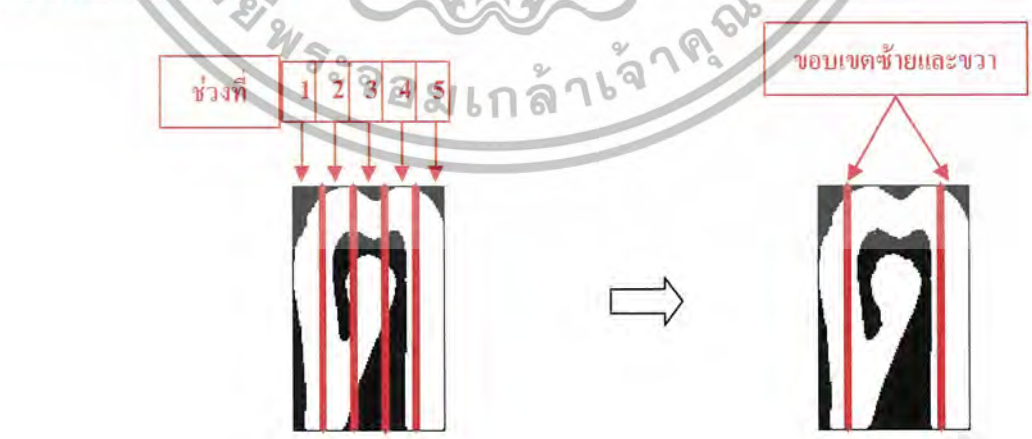

**รูปที่ 4.49** การหาพื้นที่พิกเซลสีขาวที่มากที่สุดของช่วงกลางของตัวพยัญชนะได้เป็นพื้นที่ที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



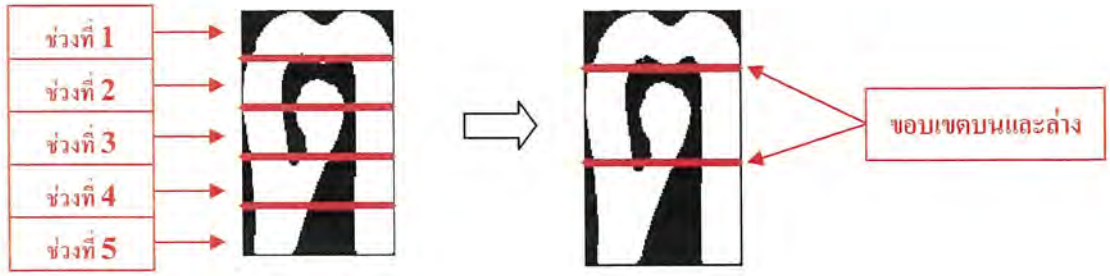

**รูปที่ 4.50** การหาพื้นที่พิกเซลสีขาวที่มากที่สุดของช่วงล่างของตัวพยัญชนะได้เป็นพื้นที่ที่ 2

จากการหาพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดของช่วงกลางและช่วงล่าง พบว่าเมื่อพื้นที่ที่ 1 น้อยกว่าพื้นที่ที่ 2 เป็นตัวพยัญชนะ ต. เต่า ที่มีส่วนหัวอยู่ถึงกลาง จึงใช้ขอบเขตจากการนำความสูงและความกว้างของตัวพยัญชนะแบ่งเป็นห้าช่วงเท่าๆ กัน ขอบเขตบนและล่างใช้ช่วงที่สองถึงสามและขอบเขตซ้ายและขวาใช้ช่วงที่สองถึงสี่ จากนั้นหาส่วนหัวของตัวพยัญชนะโดยการหาพิกัดหัวท้ายของแถว (row) และหลัก (column) ที่มีความกว้างและความสูงของกลุ่มพิกเซลสีขาวที่อยู่ติดกันมากที่สุดในขอบเขตนั้น ดังรูปที่ 4.51

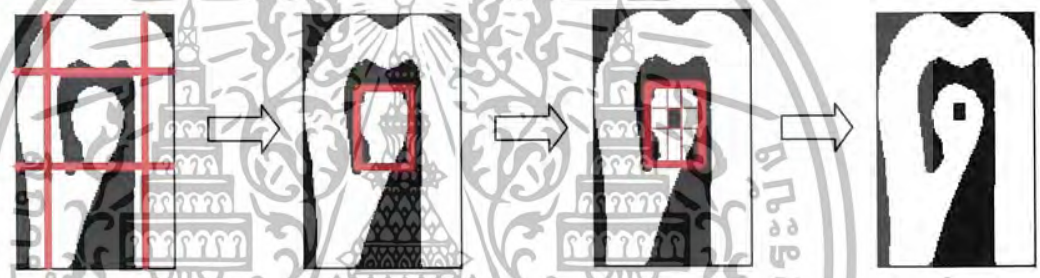



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบ่งความสูงออกเป็น 5 ช่วง



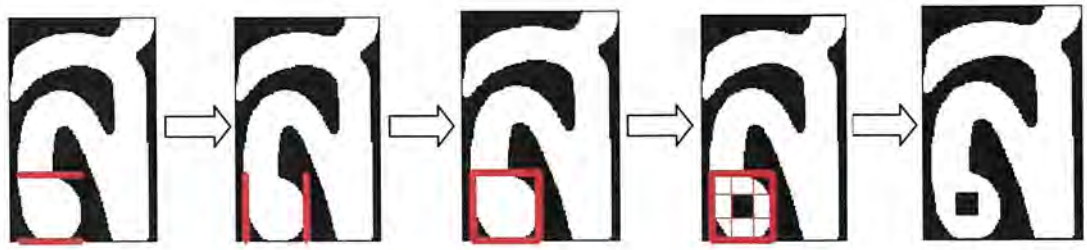

รวมขอบเขตบน-ล่างและซ้าย-ขวา หาพิกัดหัวท้ายของแถว (row) และหลัก (column) ที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตนี้แล้วตีกรอบได้ส่วนหัวของตัวพยัญชนะ จากนั้นสร้างสี่เหลี่ยม 3x3 ในกรอบนี้แล้วเปลี่ยนสี่เหลี่ยมตรงกึ่งกลางเป็นสีดำ จึงเกิดเป็นหัวของตัวพยัญชนะที่หัวอยู่กึ่งกลาง



**รูปที่ 4.51** การหาขอบเขต การหาส่วนหัวและใส่หัวให้กับตัวพยัญชนะที่มีส่วนหัวของตัวพยัญชนะอยู่ในช่วงกึ่งกลาง

ในทางตรงกันข้าม ถ้าพื้นที่ที่ 2 น้อยกว่าพื้นที่ที่ 1 เป็นตัวพยัญชนะ ส.เสือ ที่มีส่วนหัวอยู่ช่วงล่างจึงใช้ขอบเขตช่วงล่างในการหาส่วนหัวของตัวพยัญชนะ โดยการหาพิกัดหัวท้ายของแถว (row) และหลัก (column) ที่มีความกว้างและความสูงของกลุ่มพิกเซลสีขาวที่อยู่ติดกันมากที่สุดในขอบเขตนี้ได้ขอบเขตบน-ล่างและซ้าย-ขวาเป็นกรอบครอบส่วนหัวของตัวพยัญชนะ จากนั้นสร้างสี่เหลี่ยม 3x3 ในกรอบนี้แล้วเปลี่ยนสี่เหลี่ยมที่กึ่งกลางเป็นสีดำ จึงเกิดเป็นหัวของตัวพยัญชนะที่ที่ส่วนหัวอยู่ช่วงล่าง ดังรูปที่ 4.52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 4.52** การหาส่วนหัวและใส่หัวให้กับตัวพยัญชนะ ที่มีส่วนหัวของตัวพยัญชนะอยู่ในช่วงล่าง

สำหรับเงื่อนไขทั้งหมดที่ได้กล่าวมานี้ยังไม่สามารถควบคุมได้ทุกตัวพยัญชนะ และไม่มีเงื่อนไขสำหรับตัวพยัญชนะที่ไม่มีบนป้ายทะเบียนรถยนต์ และตัวพยัญชนะที่มีความซับซ้อนมาก อาทิเช่น ฃ.ขวด และ ฅ.คน เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 5

# การทดลองและผลการทดลอง

ในการทดลองจะมีอยู่สองส่วนหลักๆด้วยกัน คือการใส่หัวให้กับตัวพยัญชนะและการหาเส้นโคร่งร่างของตัวพยัญชนะและเรื่องการรู้จำอักษรด้วยวิธี Template Matching ซึ่งแบ่งผลการทดลองออกเป็น 3 ส่วน คือการแยกส่วนที่เป็นตัวเลขและตัวพยัญชนะออกจากแผ่นป้ายทะเบียน การรู้จำตัวเลขและตัวพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ด้วยวิธีการ Template Matching และการใส่หัวให้กับตัวพยัญชนะรวมทั้งการหาเส้นโครงร่างของตัวพยัญชนะที่ทำการใส่หัวแล้ว

## 5.1 การแยกตัวเลขและตัวพยัญชนะออกจากแผ่นป้ายทะเบียนรถยนต์

การแยกตัวเลขและตัวพยัญชนะออกจากแผ่นป้ายทะเบียน เริ่มต้นโปรแกรมจะโหลดไฟล์ของภาพถ่ายส่วนที่เป็นด้านหน้ารถยนต์จากฐานข้อมูลคอมพิวเตอร์ ภาพที่รับเข้ามานั้นเป็นภาพสี RGB อย่างไรก็ตามถ่ายภาพเหล่านั้นจะถ่ายในช่วงเวลาที่แตกต่างกัน ดังนั้นภาพถ่ายด้านหน้ารถยนต์อาจมีแสงที่มาตกกระทบลงบนพื้นผิววัตถุมากหรือน้อยจนเกินไป ซึ่งจะมีผลต่อการค้นหาภาพส่วนที่เป็นแผ่นป้ายทะเบียน ขั้นตอนแยกมีดังนี้

**ขั้นตอนที่ 1** โหลดภาพถ่ายส่วนที่เป็นด้านหน้ารถยนต์ขึ้นมาด้วยคำสั่ง imread จากไฟล์ข้อมูล ซึ่งเป็นภาพสี RGB ดังรูปที่ 5.1

**รูปที่ 5.1** ภาพถ่ายด้านหน้ารถยนต์ที่เป็นภาพสี RGB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ขั้นตอนที่ 2** แปลงภาพสี RGB ให้เป็นภาพระดับเทาด้วยคำสั่ง rgb2gray

**รูปที่ 5.2** ภาพที่ได้จากการแปลงภาพสี RGB ให้เป็นภาพระดับเทา

**ขั้นตอนที่ 3** แปลงภาพระดับเทาให้เป็นภาพขาว – ดำ หรือภาพสองระดับ ด้วยคำสั่ง im2bw โดยคำสั่งนี้ใช้วิธีการตัดระดับเทรชโฮลแบบอัตโนมัติ ค่าเทรชโฮลจะได้มาโดยอัตโนมัติจากภาพระดับเทาและสามารถแสดงค่าเทรชโฮลได้จากคำสั่ง graythresh (การแปลงจากภาพระดับเทาเป็นภาพสองระดับนั้นด้วยค่าเทรชโฮลนั่น คือพิกเซลใดๆ ที่มีค่าความสว่างน้อยกว่าค่าเทรชโฮล พิกเซลนั้นจะเป็นสีดำ และพิกเซลใดๆ ที่มีค่าความสว่างมากกว่าค่าเทรชโฮล พิกเซลนั้นจะเป็นสีขาว)

**รูปที่ 5.3** ภาพผลลัพธ์จากการแปลงภาพระดับเทาให้เป็นภาพสองระดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ขั้นตอนที่ 4** กำจัดกลุ่มพิกเซลสีขาวที่ไม่เกี่ยวข้องกับแผ่นป้ายทะเบียนออกจากภาพ จะพบว่ามีกลุ่มพิกเซลสีขาวที่ไม่เกี่ยวข้องกับแผ่นป้ายทะเบียนที่อยู่ติดกับขอบภาพจึงใช้ วิธีกำจัดกลุ่มพิกเซลสีขาวที่อยู่ติดกับขอบภาพด้วยคำสั่ง imclearborder

**รูปที่ 5.4** ภาพผลลัพธ์หลังจากกำจัดกลุ่มพิกเซลสีขาวที่ติดกับขอบภาพออกไป

**ขั้นตอนที่ 5** นำคู่พิกัดของกลุ่มพิกเซลสีขาวที่มีพื้นที่มากที่สุดไปคัดลอกจากภาพสี RGB ที่รับเข้ามาในขั้นตอนที่ 1 จะได้รูปส่วนที่เป็นแผ่นป้ายทะเบียนออกมา

**รูปที่ 5.5** ภาพที่ได้จากการนำคู่พิกัดของกลุ่มพิกเซลสีขาวที่มีพื้นที่มากที่สุดจากรูปที่ 5.4 ไปคัดลอกจากภาพสี RGB ที่รับเข้ามาในขั้นตอนแรก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ขั้นตอนที่ 6** กลับภาพสองระดับจากดำเป็นขาวและขาวเป็นดำ เพื่อทำการกำจัดส่วนที่เป็นชื่อจังหวัดออกไป โดยการหาขอบเขตบนสุดและขอบเขตล่างสุด เพื่อหาความสูงของกลุ่มพิกเซลสีขาว ซึ่งถ้ากลุ่มพิกเซลสีขาวมีความสูงต่ำกว่า 0.25 เท่าของความสูงของภาพสองระดับก็จะถูกกำจัดออกไป

**รูปที่ 5.7** ภาพผลลัพธ์จากการกลับภาพสองระดับ

**รูปที่ 5.8** ภาพผลลัพธ์ที่ได้จากการกำจัดส่วนที่เป็นชื่อจังหวัดออกไป

**ขั้นตอนที่ 7** ทำการหาขอบเขตด้านบนและขอบเขตด้านล่างของตัวพยัญชนะ และตัวเลขบนแผ่นป้ายทะเบียนเพื่อที่จะทำการลบพื้นที่ส่วนที่ไม่ใช่ตัวพยัญชนะและตัวเลขออกไป

**รูปที่ 5.9** ภาพผลลัพธ์ที่ได้จากการนำรูปที่ 5.8 ไปตัดขอบเขตด้านบนและขอบเขตด้านล่างให้กับกลุ่มตัวเลขและตัวพยัญชนะ

**ขั้นตอนที่ 8** หาขอบเขตด้านซ้ายและขอบเขตด้านขวาของตัวเลขและตัวพยัญชนะแต่ละตัว เพื่อเก็บค่า หลัก (Column) ที่เป็นขอบเขตด้านซ้ายไว้ในตัวแปร T1 และขอบเขตด้านขวาไว้ในตัวแปร U1 และนำค่าหลัก (Column) ที่เก็บไว้ในตัวแปร T1 และ U1 มาเรียงลำดับจากน้อยไปหามากไว้ ดังรูปที่ 5.10

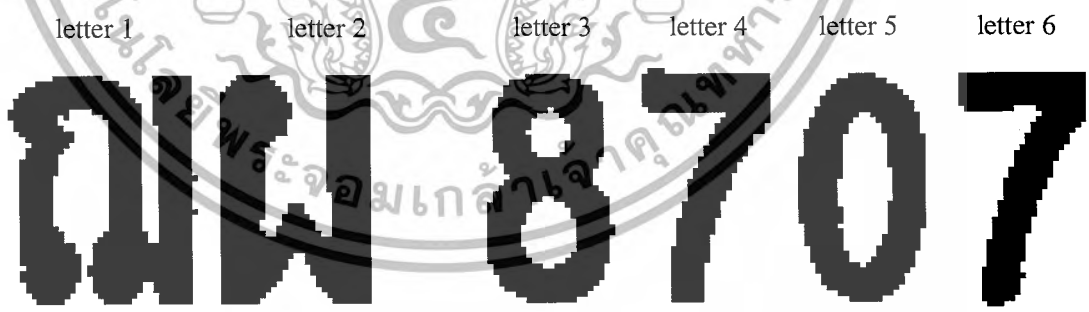
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
T1  =

   13     57    112    144    177    210

U1  =

   50     86    137    170    202    235
```

**รูปที่ 5.10** ค่าขอบเขตด้านซ้ายและขอบเขตด้านขวาที่เก็บอยู่ในตัวแปร T1 และ U1 ตามลำดับ

**ขั้นตอนที่ 9** นำค่าหลัก (column) ที่เป็นขอบเขตด้านซ้ายและขอบเขตด้านขวาของตัวพยัญชนะและตัวเลขมาใช้เป็นแกนอ้างอิงในการตัดภาพของตัวเลขและตัวพยัญชนะ โดยเรียงลำดับภาพให้เหมือนแผ่นป้ายทะเบียน จากนั้นทำการกลับภาพสองระดับเพื่อนำไปหาค่าสัมประสิทธิ์ความเหมือนกับชุดภาพแม่แบบที่เป็นภาพสองระดับต่อไป แสดงในรูปที่ 5.11

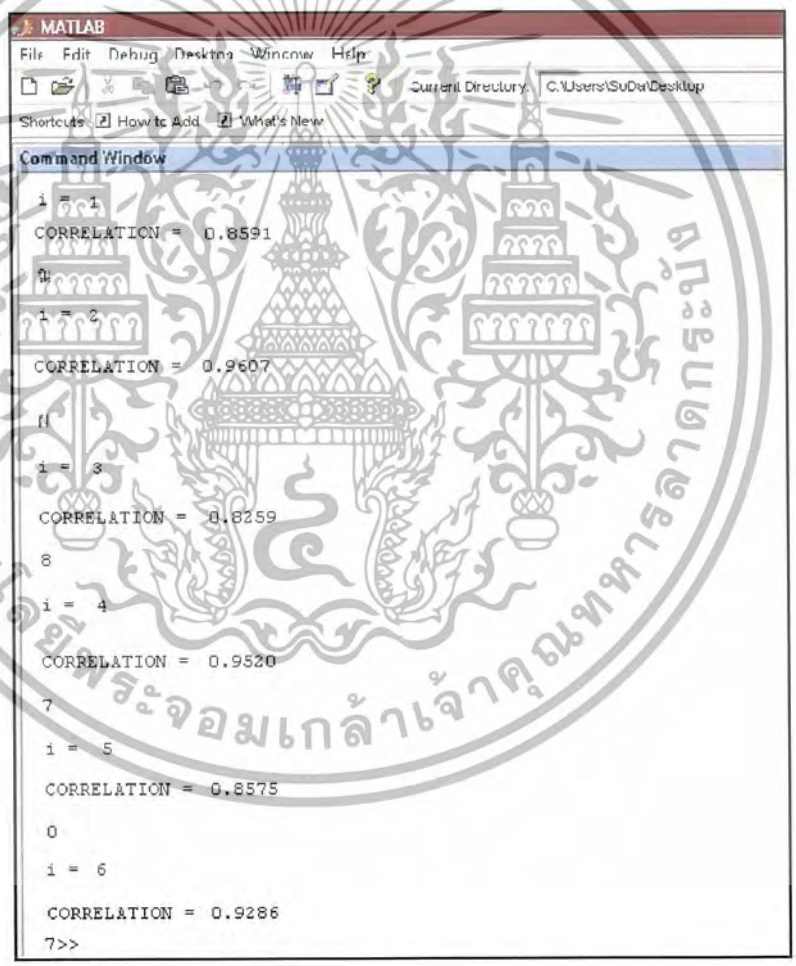
**รูปที่ 5.11** ภาพผลลัพธ์หลังจากกลับภาพสองระดับ




**รูปที่ 5.12** ตัวเลขและตัวพยัญชนะที่ถูกแยกออกมาจากแผ่นป้ายทะเบียน
โดยทำการเรียงลำดับให้เหมือนแผ่นป้ายทะเบียนรถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

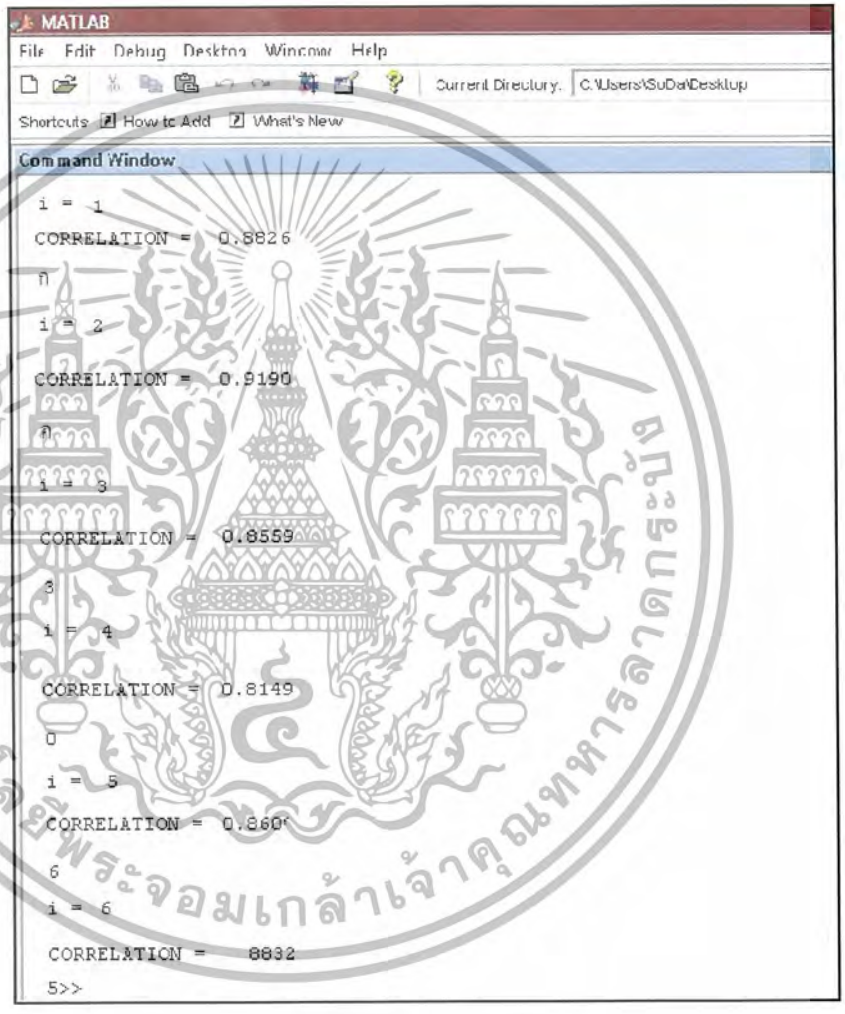
ตัวอย่างผลการทดลองขั้นตอนการรู้จำตัวเลขและตัวพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ด้วยวิธีการ Template matching จากรูปที่ 5.13 – 5.21




**รูปที่ 5.13** ภาพผลการทดลองรถยนต์ทะเบียน ฌผ 8707

จากรูปที่ 5.13 แสดงค่าสัมประสิทธิ์ความเหมือนของภาพที่ต้องการพิจารณากับภาพชุดแม่แบบของแต่ละตัวโดยค่าที่ได้เป็นเพียงค่าเดียวที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้คือ 0.600

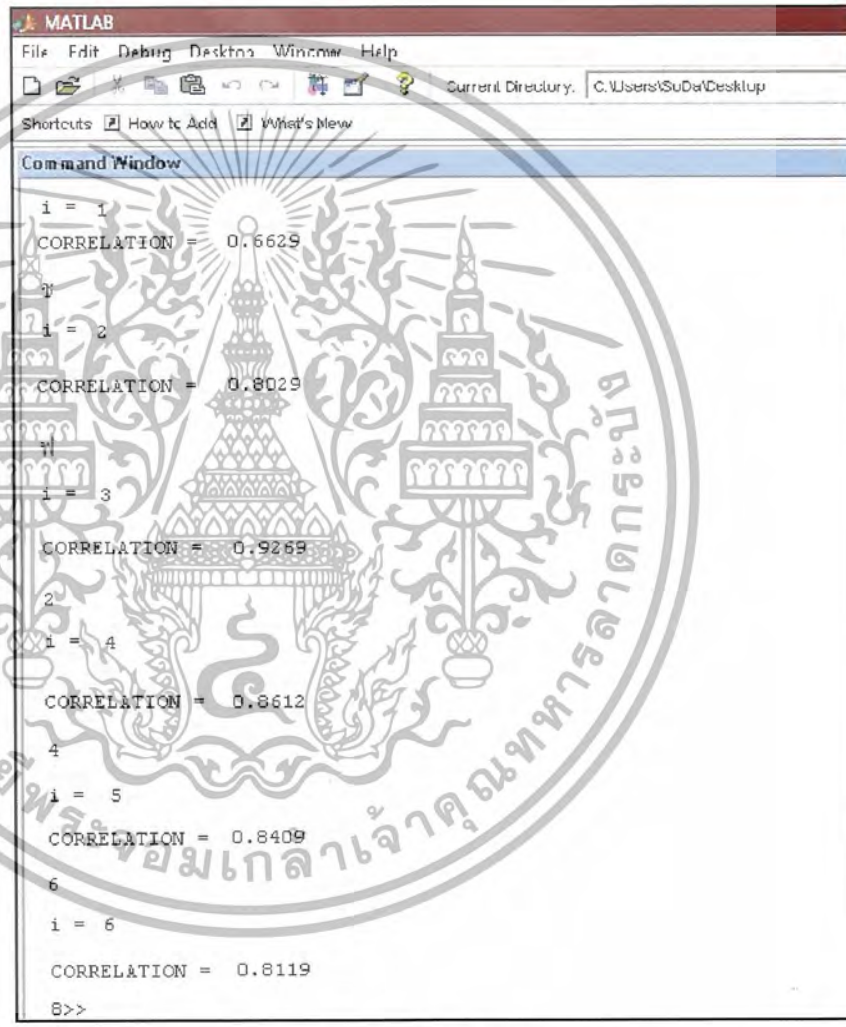
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 5.14** ภาพผลการทดลองรถยนต์ทะเบียน กค 3065

จากรูปที่ 5.14 แสดงค่าสัมประสิทธิ์ความเหมือนของภาพที่ต้องการพิจารณากับภาพชุดแม่แบบของแต่ละตัวโดยค่าที่ได้เป็นเพียงค่าเดียวที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้คือ 0.600

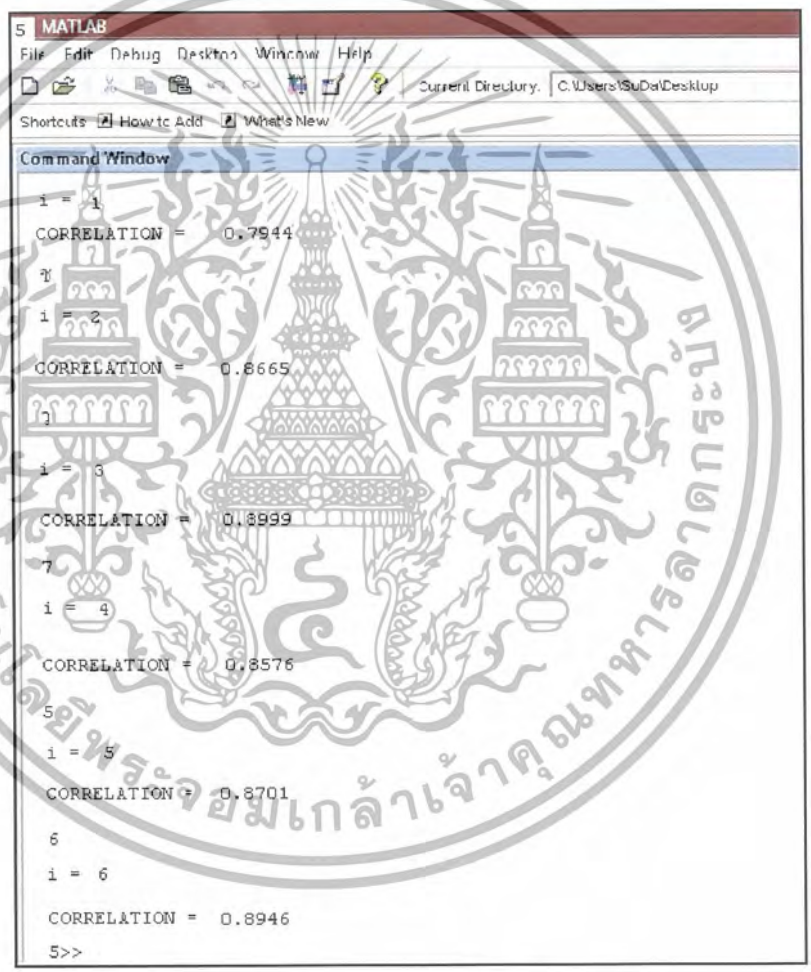
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 5.15** ภาพผลการทดลองรถยนต์ทะเบียน ชฟ 2468

จากรูปที่ 5.15 แสดงค่าสัมประสิทธิ์ความเหมือนของภาพที่ต้องการพิจารณากับภาพชุดแม่แบบของแต่ละตัวโดยค่าที่ได้เป็นเพียงค่าเดียวที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้คือ 0.600

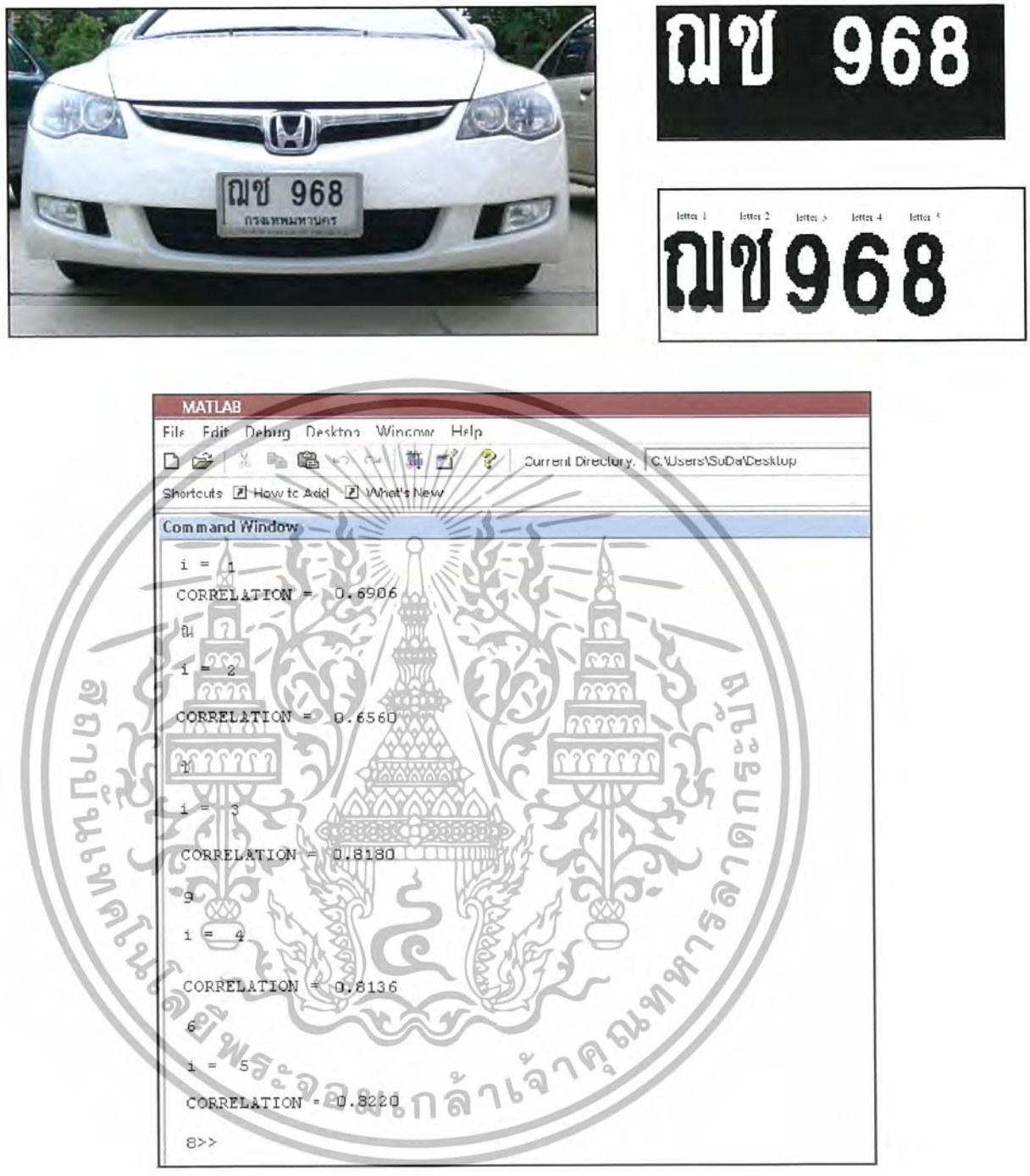
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 5.16** ภาพผลการทดลองรถยนต์ทะเบียน ชว 7565

จากรูปที่ 5.16 แสดงค่าสัมประสิทธิ์ความเหมือนของภาพที่ต้องการพิจารณากับภาพชุดแม่แบบของแต่ละตัวโดยค่าที่ได้เป็นเพียงค่าเดียวที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้คือ 0.600

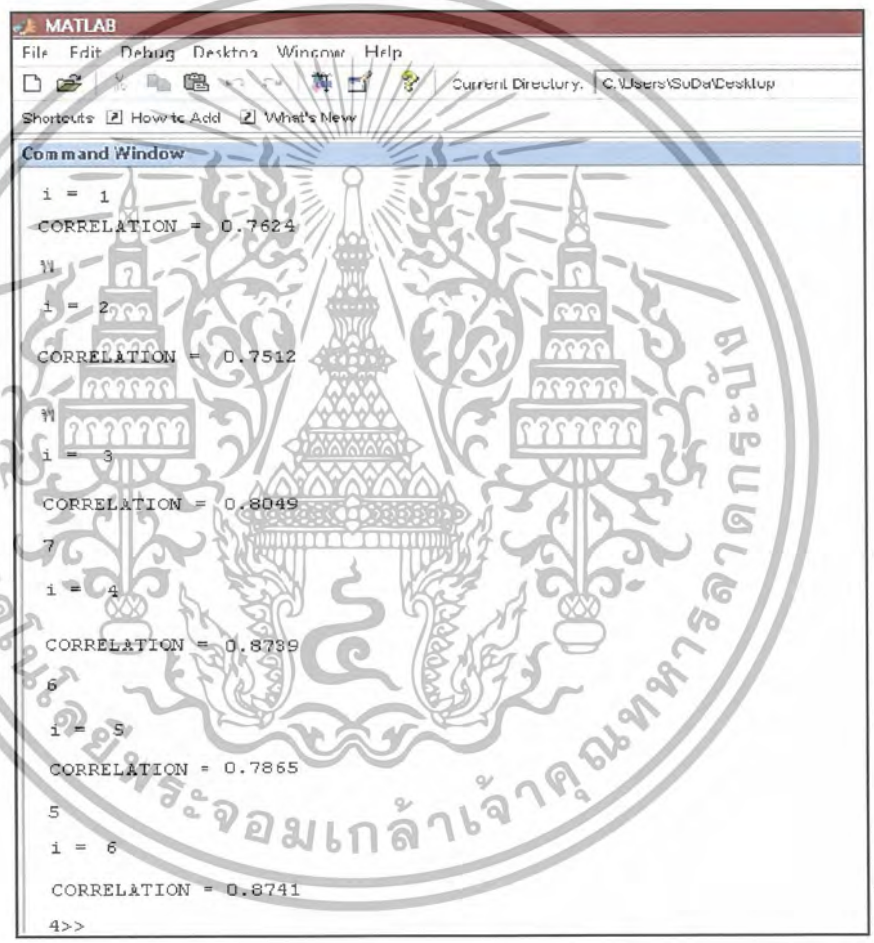
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 5.17** ภาพผลการทดลองรถยนต์ทะเบียน ฌช 968

จากรูปที่ 5.17 แสดงค่าสัมประสิทธิ์ความเหมือนของภาพที่ต้องการพิจารณากับภาพชุดแม่แบบของแต่ละตัวโดยค่าที่ได้เป็นเพียงค่าเดียวที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้คือ 0.600

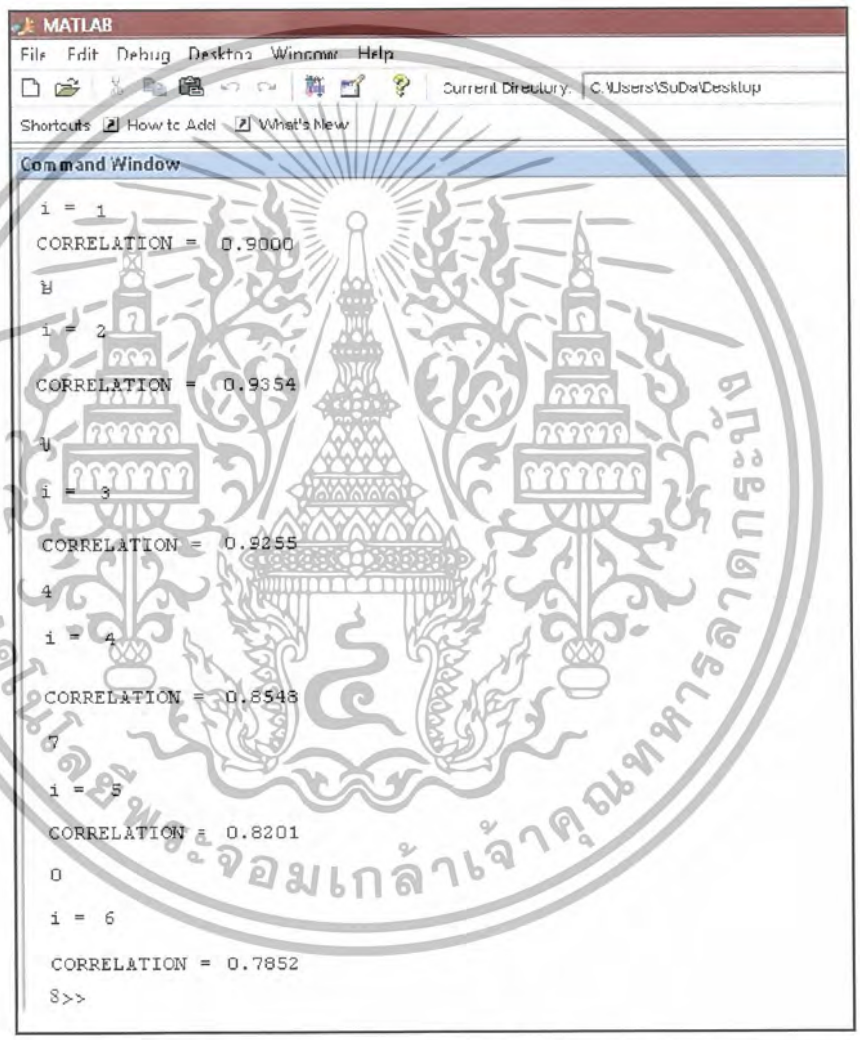
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 5.18** ภาพผลการทดลองรถยนต์ทะเบียน พพ 7654

จากรูปที่ 5.18 แสดงค่าสัมประสิทธิ์ความเหมือนของภาพที่ต้องการพิจารณากับภาพชุดแม่แบบของแต่ละตัวโดยค่าที่ได้เป็นเพียงค่าเดียวที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้คือ 0.600

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 5.19** ภาพผลการทดลองรถยนต์ทะเบียน ษข 4708

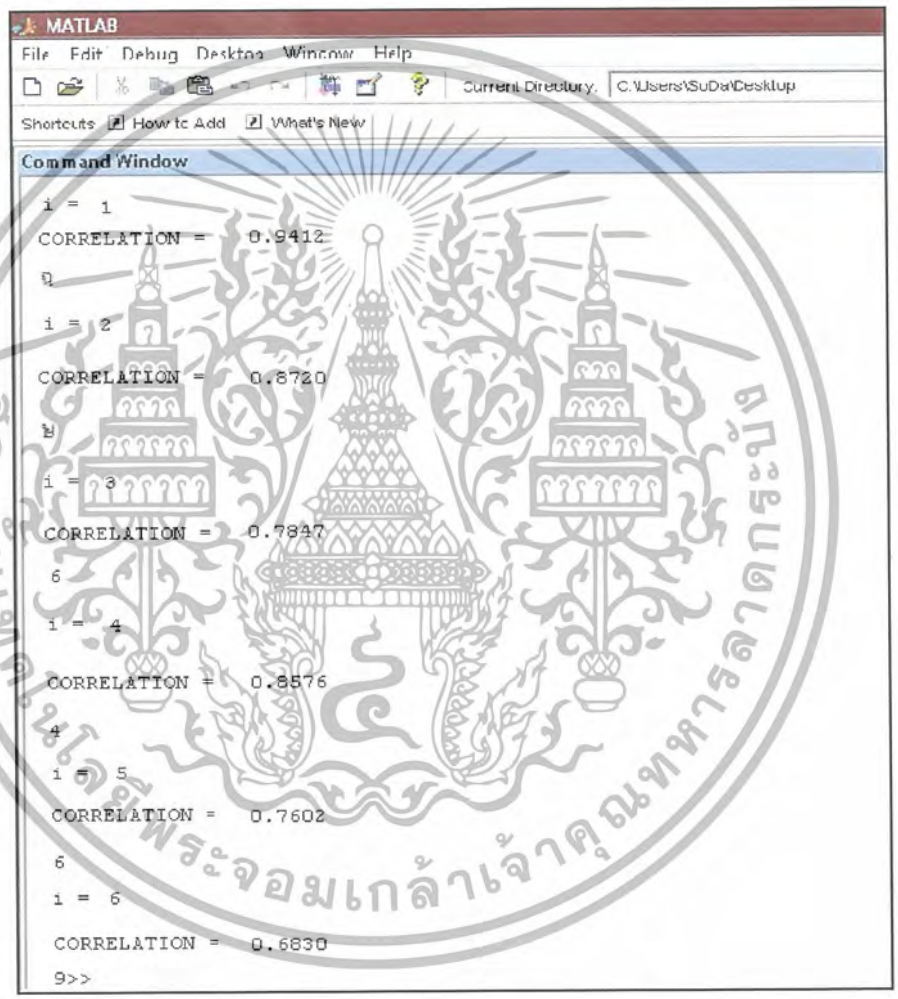
จากรูปที่ 5.19 แสดงค่าสัมประสิทธิ์ความเหมือนของภาพที่ต้องการพิจารณากับภาพชุดแม่แบบของแต่ละตัวโดยค่าที่ได้เป็นเพียงค่าเดียวที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้คือ 0.600

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**รูปที่ 5.20** ภาพผลการทดลองรถยนต์ทะเบียน สล 4677

จากรูปที่ 5.20 แสดงค่าสัมประสิทธิ์ความเหมือนของภาพที่ต้องการพิจารณากับภาพชุดแม่แบบของแต่ละตัวโดยค่าที่ได้เป็นเพียงค่าเดียวที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้คือ 0.600

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 5.21** ภาพผลการทดลองรถยนต์ทะเบียน ฉษ 6469

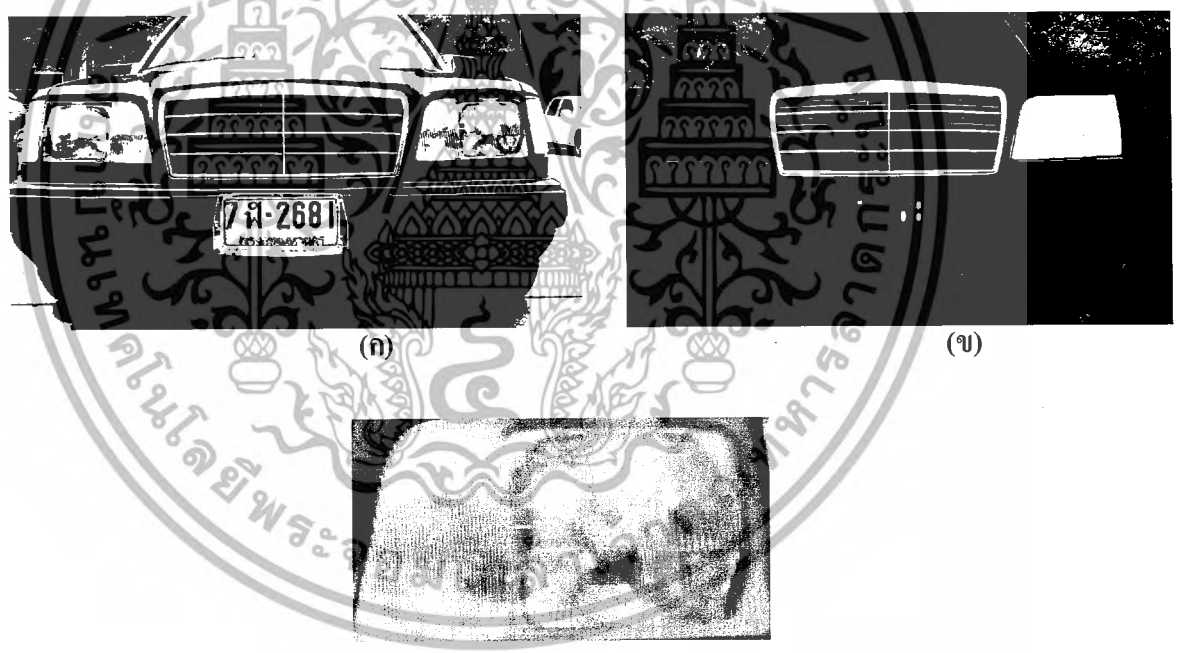
จากรูปที่ 5.13 แสดงค่าสัมประสิทธิ์ความเหมือนของภาพที่ต้องการพิจารณากับภาพชุดแม่แบบของแต่ละตัวโดยค่าที่ได้เป็นเพียงค่าเดียวที่มีค่ามากกว่าค่าตัดระดับเทสโฮลที่ตั้งไว้คือ 0.600

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างภาพผลการทดลองที่พบปัญหาต่างๆ เช่นไม่สามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ได้ และภาพของแผ่นป้ายทะเบียนรถยนต์ที่มีลวดลาย

ตัวอย่างภาพผลการทดลองการรู้จำตัวเลขและตัวพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ ด้วยวิธีการ Template Matching จากผลการทดลอง พบว่ามีบางภาพที่โปรแกรมไม่สามารถทำการแยกตัวเลขและตัวพยัญชนะออกมาจากภาพแผ่นป้ายทะเบียนได้ เนื่องจากเมื่อทำการแปลงภาพสี RGB ให้เป็นภาพสองระดับ โดยการใช้ค่าเทรชโฮลอัตโนมัติ เมื่อทำการกำจัดกลุ่มพิกเซลสีขาวที่ติดอยู่กับขอบภาพ ทำให้ส่วนที่เป็นแผ่นป้ายทะเบียนนั้นหายไป และไม่สามารถแยกภาพส่วนที่เป็นแผ่นป้ายทะเบียนได้ ดังรูปที่ 5.22 (ก) – (ฆ)

ตัวอย่างภาพผลการทดลองที่ใช้การปรับค่าเทรชโฮลอัตโนมัติและกำจัดกลุ่มพิกเซลสีขาวที่อยู่ติดกับขอบภาพออกไปเพื่อค้นหาส่วนที่เป็นแผ่นป้ายทะเบียน ผลที่ได้คือได้ส่วนที่เป็นไฟหน้ารถมาแทนเพราะว่าเมื่อทำการรวมกลุ่มพิกเซลสีขาวเข้าด้วยกันแล้ว กลุ่มพิกเซลสีขาวตรงส่วนไฟหน้ารถยนต์มีพื้นที่มากกว่า กลุ่มพิกเซลสีขาวส่วนที่เป็นกระจังหน้ารถยนต์ ดังภาพที่ 5.22 (ก) – (ฆ)




(ก) (ข)

(ฆ)

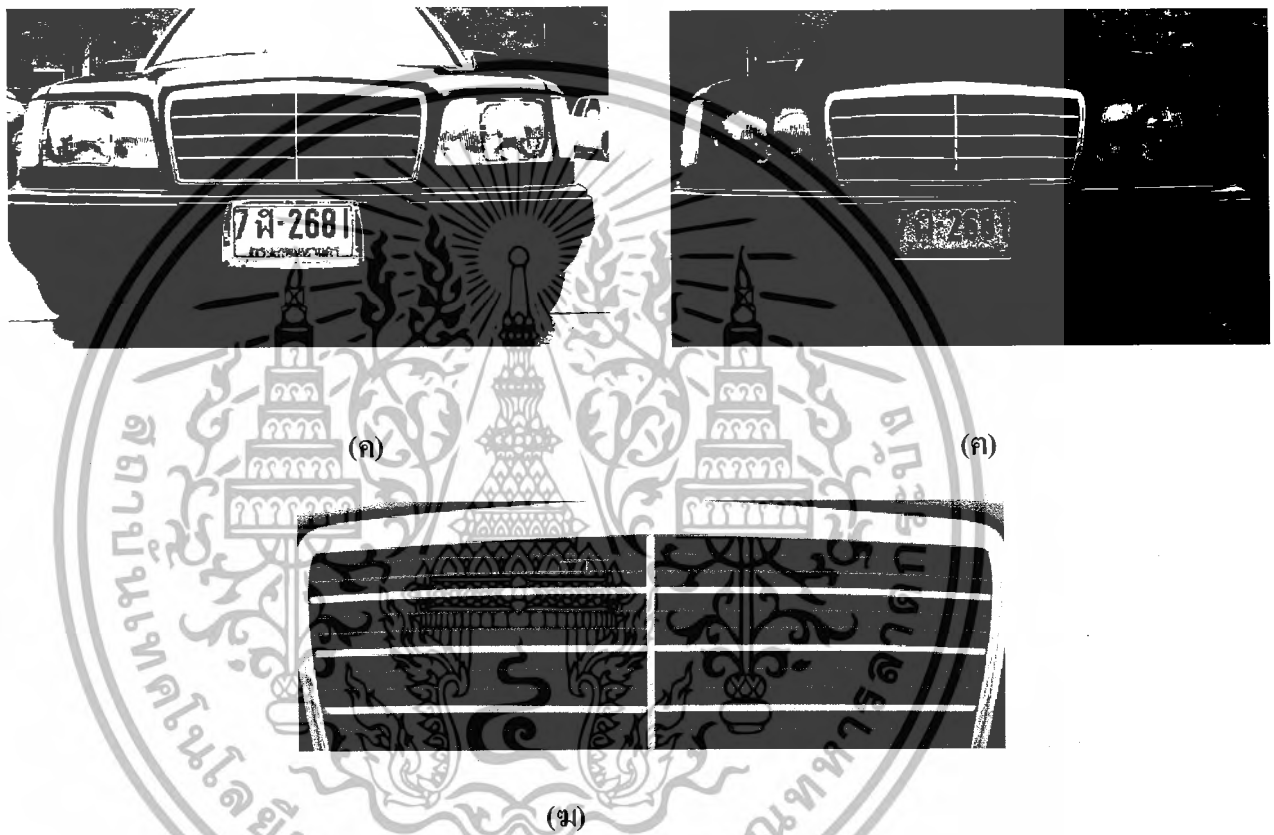
**รูปที่ 5.22** **(ก)** แปลงภาพสี RGB เป็นภาพสองระดับ

**(ข)** ภาพสองระดับที่ได้จากการปรับค่าเทรชโฮลอัตโนมัติ

**(ฆ)** ภาพผลลัพธ์ได้รูปไฟด้านหน้ารถยนต์ไม่สามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างภาพผลการทดลองในการปรับค่าเทรชโอลด้วยตนเอง เมื่อทำการปรับค่าเทรชโฮลด้วยตนเองเพื่อทำให้ภาพมีความสว่างมากขึ้นเพื่อให้เห็นส่วนที่เป็นแผ่นป้ายทะเบียนแต่ จะเห็นได้ว่าส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์มีจำนวนกลุ่มพิกเซลสีขาวน้อยกว่ากลุ่มพิกเซลสีขาวของส่วนที่เป็นกระจังหน้ารถยนต์ เมื่อค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์จึงไม่สามารถหาได้และผลที่ได้ออกมานั้นจึงได้ส่วนที่เป็นไฟหน้ารถมาแทน ดังรูปที่ 5.22 (ค) – (ฆ)




**รูปที่ 5.22** (ค) แปลงภาพสี RGB เป็นภาพสองระดับ
(ต) ภาพสองระดับของภาพสี RGB ที่เกิดจากการปรับค่าเทรชโฮลเอง
(ฆ) ภาพผลลัพธ์จากการค้นหาแผ่นป้ายทะเบียนแต่ได้กระจังหน้ารถยนต์มาแทน

ตัวอย่างภาพผลการทดลองที่ใช้แผ่นป้ายทะเบียนรถยนต์ที่มีพื้นหลังเป็นลวดลาย เมื่อทำการค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนซึ่งไม่สามารถค้นหาได้ จึงได้ทำการปรับค่าเทรชโฮลให้มีค่าที่เพิ่มมากขึ้น เพื่อเพิ่มความสว่างให้กับพื้นหลังของแผ่นป้ายทะเบียนรถยนต์ ทำให้สามารถค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ได้ ดังรูปที่ 5.22 (ง) – (ฌ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ง)

(จ) (ฉ)

**รูปที่ 5.22** (ง) แปลงภาพสี RGB เป็นภาพสองระดับ
(จ) ภาพผลลัพธ์ที่เกิดจากการปรับค่าเทรชโฮลเอง
(ฉ) ภาพผลลัพธ์ที่เกิดจากการปรับค่าเทรชโดยอัตโนมัติ

ใช้โปรแกรม Photoscape เพื่อทดลองปรับความสว่างของภาพให้เพิ่มมากขึ้น เพื่อช่วยในการค้นหาส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ แต่เมื่อปรับความสว่างของภาพรวมทั้งปรับค่าเทรชโฮลด้วยตนเองแล้วก็สามารถค้นหาส่วนของแผ่นป้ายทะเบียนรถยนต์ได้ แต่เมื่อแยกตัวเลขและตัวพยัญชนะออกจากแผ่นป้ายทะเบียนจะไม่สามารถแยกตัวพยัญชนะได้ครบทุกตัว เนื่องจากเมื่อค้นหาแผ่นป้ายทะเบียนรถยนต์และทำการกลับภาพสองระดับดังรูปที่ 5.22 (ญ) สังเกตุเห็นว่าตัวพยัญชนะ ส. เสือ มีบางส่วนที่ติดอยู่กับขอบภาพ เมื่อทำการลบส่วนที่เป็นชื่อจังหวัดออกจากแผ่นป้ายทะเบียน ทำให้ตัวพยัญชนะ ส.เสือ ถูกลบไปด้วย ดังรูปที่ 5.22 (ช) – (ฎ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ช)

(ซ)

(ฌ)

(ญ)

(ฎ)

**รูปที่ 5.22** **(ช)** ภาพที่ทำการปรับค่าความสว่างด้วยโปรแกรม Photoscape
**(ซ)** ภาพจากการแปลงภาพสี RGB เป็นภาพสองระดับ
**(ฌ)** ภาพผลลัพธ์ที่เกิดจากการปรับค่าเทรชโฮลอัตโนมัติ
**(ญ)** ภาพผลลัพธ์การค้นหาแผ่นป้ายทะเบียนรถยนต์และกลับภาพสองระดับ
**(ฎ)** ภาพผลลัพธ์หลังจากทำการลบส่วนที่เป็นชื่อจังหวัดออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลองการค้นหาแผ่นป้ายทะเบียนรถยนต์ ซึ่งใช้ภาพถ่ายด้านหน้ารถยนต์ที่นำมาทดลองทั้งหมด 40 ภาพ สามารถค้นหาแผ่นป้ายทะเบียนได้ทั้งหมด 36 ภาพ คิดเป็น 90% รวมทั้งยังสามารถแยกตัวเลขและตัวพยัญชนะออกจากแผ่นป้ายทะเบียนรถยนต์ได้ครบสมบูรณ์ทั้งหมด 36 ภาพ คิดเป็น 90% ดังตารางที่ 5.1

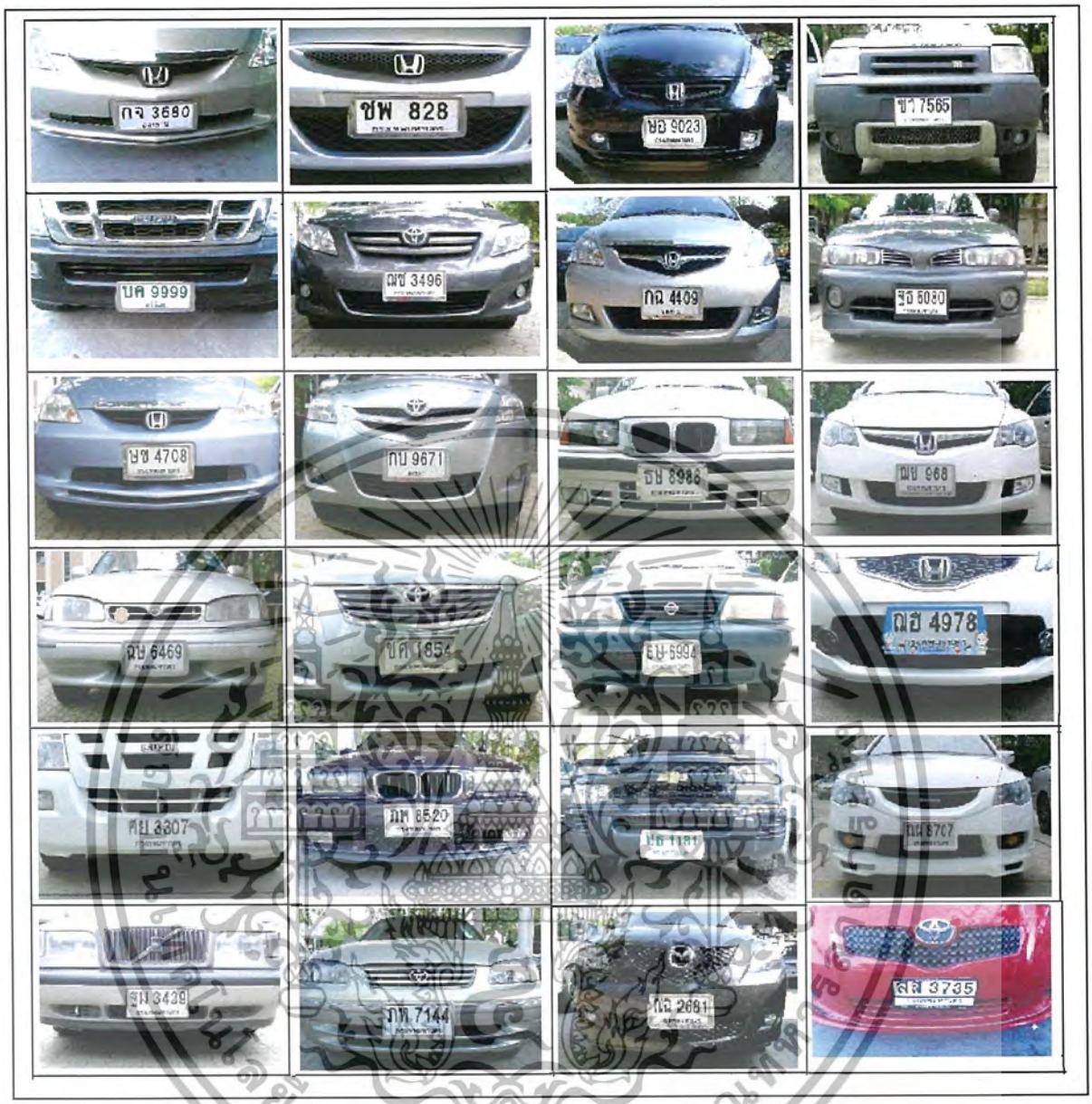
**ตารางที่ 5.1** ผลการทดลองในการค้นหาแผ่นป้ายทะเบียน

| จำนวนรูปที่ใช้ทำการทดลอง | จำนวนรูปที่โปรแกรมสามารถทำการรู้จำแผ่นป้ายทะเบียนได้ถูกต้อง | เปอร์เซนต์ความถูกต้อง |
|---|---|---|
| 40 | 36 | 90% |

**ภาพถ่ายด้านหน้ารถยนต์ที่ใช้ในการทดลอง**

ภาพถ่ายด้านหน้ารถยนต์ที่สามารถค้นหาแผ่นป้ายทะเบียนได้มีทั้งหมด 36 ภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




ภาพถ่ายด้านหน้ารถยนต์ที่ไม่สามารถค้นหาแผ่นป้ายทะเบียนได้มีทั้งหมด 4 ภาพ

**รูปที่ 5.23** ภาพถ่ายด้านหน้ารถยนต์ทั้งหมดที่ใช้ในการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 5.2** การแยกตัวเลขที่อยู่บนแผ่นป้ายทะเบียน

| ตัวเลขที่อยู่บนแผ่นป้ายทะเบียน | จำนวนตัวเลขที่อยู่บนแผ่นป้ายทะเบียนรถยนต์ | เปอร์เซนต์ความถูกต้องในการรู้จำตัวเลข |
|---|---|---|
| 0 | 16 | 86.60% |
| 1 | 8 | 76.44% |
| 2 | 6 | 83.00% |
| 3 | 16 | 87.33% |
| 4 | 18 | 88.80% |
| 5 | 15 | 83.20% |
| 6 | 19 | 88.60% |
| 7 | 19 | 84.20% |
| 8 | 20 | 83.40% |
| 9 | 17 | 86.60% |

จากตางรางที่ 5.2 จะแสดงตัวเลขและจำนวนตัวเลขที่ได้มาจากแผ่นป้ายทะเบียนรถยนต์ รวมทั้งเปอร์เซนต์ความถูกต้องที่สามารถทำการรู้จำตัวเลขแต่ละตัวได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 5.3** การแยกตัวพยัญชนะที่อยู่บนแผ่นป้ายทะเบียน

| ตัวพยัญชนะที่อยู่บนแผ่นป้ายทะเบียน | จำนวนตัวพยัญชนะที่อยู่บนแผ่นป้ายทะเบียนรถยนต์ | เปอร์เซนต์ความถูกต้องในการรู้จำตัวพยัญชนะ |
|---|---|---|
| ก | 6 | 85.16% |
| ข | 1 | 93.90% |
| ค | 4 | 83.00% |
| ง | 2 | 86.00% |
| จ | 2 | 89.50% |
| ฉ | 4 | 86.25% |
| ช | 9 | 88.19% |
| ฌ | 6 | 76.60% |
| ฐ | 2 | 99.50% |
| ธ | 4 | 85.50% |
| บ | 3 | 84.50% |
| ป | 1 | 83.54% |

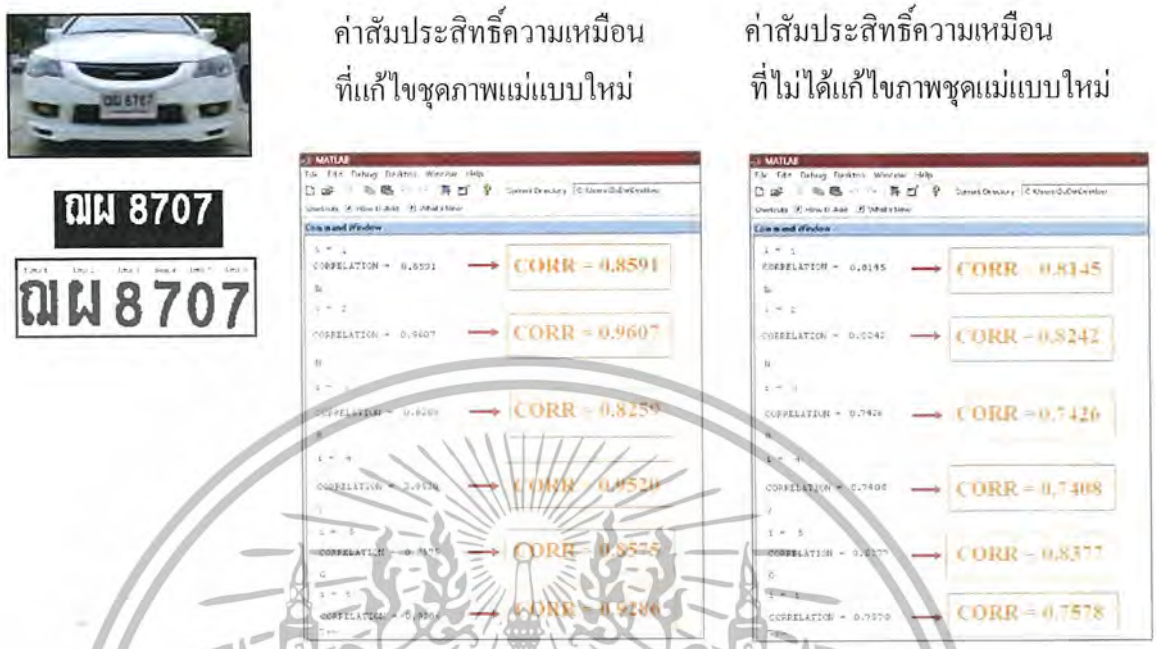
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | |
|---|---|---|
| ผ | 1 | 95.56% |
| พ | 1 | 77.00% |
| ฟ | 1 | 78.53% |
| ม | 2 | 86.50% |
| ย | 2 | 85.13% |
| ล | 1 | 94.16% |
| ว | 1 | 86.22% |
| ศ | 3 | 79.00% |
| ษ | 5 | 88.00% |
| ส | 4 | 83.00% |
| อ | 3 | 74.52% |
| ฮ | 1 | 90.40% |

จากตารางที่ 5.3 จะแสดงตัวพยัญชนะและจำนวนตัวพยัญชนะที่ได้มาจากแผ่นป้ายทะเบียนรถยนต์ รวมทั้งเปอร์เซนต์ความถูกต้องที่สามารถทำการรู้จำตัวพยัญชนะแต่ละตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ผลการเปรียบเทียบวิธี Template Matching กับของรุ่นพี่**




**รูปที่ 5.24** ภาพการเปรียบเทียบค่าสัมประสิทธิ์ความเหมือนในวิธีการ Template Matching

จะเห็นว่าเมื่อทำการแก้ไขวิธีการทำชุดภาพแม่แบบใหม่ เมื่อนำภาพที่กำลังพิจารณามาทำการหาค่าสัมประสิทธิ์ความเหมือนกับภาพชุดแม่แบบ ก็ได้ว่าค่าสัมประสิทธิ์ความเหมือนที่ได้มีค่ามากกว่า ค่าสัมประสิทธิ์ความเหมือนที่รุ่นพี่ได้ทำไว้

**5.2 ผลการทดลองการใส่หัวให้กับตัวพยัญชนะและหาเส้นโครงร่างของตัวพยัญชนะ**

ตัวพยัญชนะทั้งหมดที่หาได้จากแผ่นป้ายทะเบียนรถยนต์ แสดงดังรูปที่ 5.24

**รูปที่ 5.25** ตัวพยัญชนะทั้งหมดที่หาได้บนป้ายทะเบียนรถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหาส่วนหัวของตัวพยัญชนะ แบ่งตามตำแหน่งส่วนหัวของตัวพยัญชนะสามารถแบ่งได้เป็นสามตำแหน่งใหญ่ๆด้วยกัน คือ

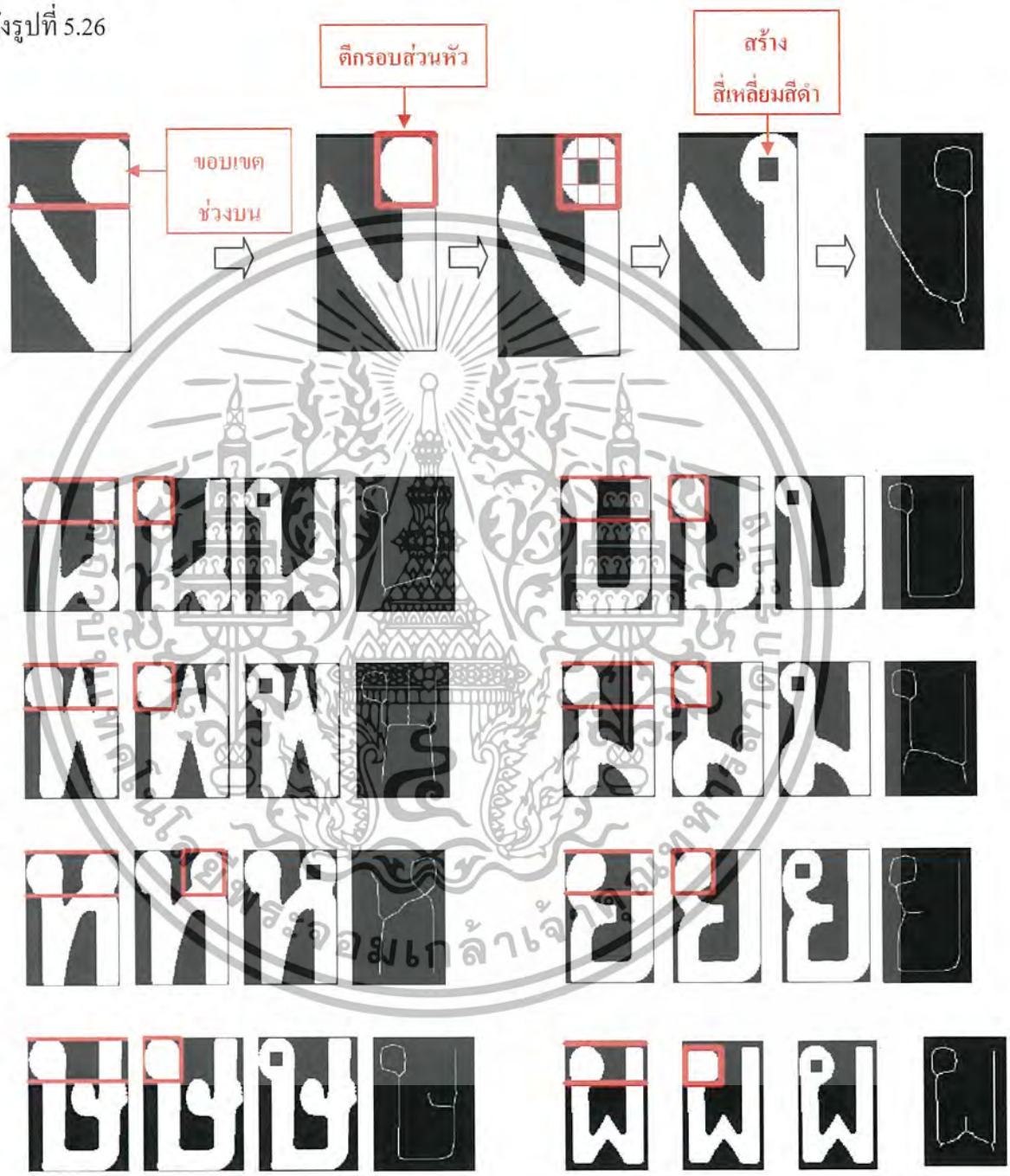
1. ส่วนหัวของตัวพยัญชนะอยู่ช่วงบน มีตัวพยัญชนะคือ ง, น, บ, พ, ม, ย, ษ, ห และ ผ
2. ส่วนหัวของตัวพยัญชนะอยู่ช่วงกลาง มีตัวพยัญชนะคือ ค, จ, ฉ, ด, ต, ศ, อ และ ฮ
3. ส่วนหัวของตัวพยัญชนะอยู่ช่วงล่าง มีตัวพยัญชนะคือ ฌ, ญ, ฒ, ถ, ภ, ร, ล, ว และ ส

1. ส่วนหัวของตัวพยัญชนะอยู่ช่วงบน มีตัวพยัญชนะคือ ง, น, บ, พ, ม, ย, ษ, ห และ ผ วิธีที่ใช้ในการแยกพยัญชนะที่มีหัวอยู่ช่วงบนออกจากตัวอื่นๆ คือการหาเส้นสมมติเปรียบเทียบกับแถวที่มีพิกเซลสีขาวอยู่ติดกันมากที่สุด พบว่าแถวที่มีพิกเซลสีขาวอยู่ติดกันมากที่สุดต้องอยู่ต่ำกว่าเส้นสมมติ แสดงดังรูปที่ 5.25

**รูปที่ 5.26** ภาพแสดงเส้นสมมติบนตัวพยัญชนะแต่ละตัวที่มีส่วนหัวอยู่ช่วงบน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

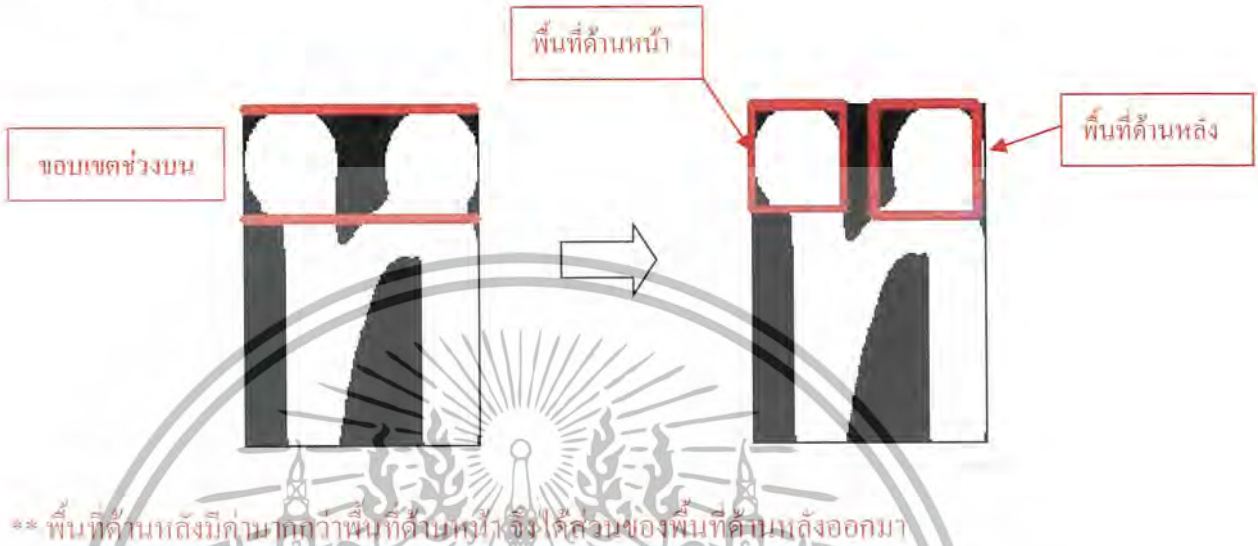
เมื่อสามารถแยกได้แล้วว่าเป็นพยัญชนะที่มีส่วนหัวอยู่ช่วงบน จึงใช้ขอบเขตช่วงบนในการหาพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดแล้วตีกรอบ จากนั้นสร้างสี่เหลี่ยม 3x3 ในกรอบนี้แล้วเปลี่ยนสี่เหลี่ยมที่กึ่งกลางเป็นสีดำเกิดเป็นหัวของตัวพยัญชนะ แล้วหาเส้นโครงร่างของตัวพยัญชนะ แสดงดังรูปที่ 5.26




**รูปที่ 5.27** การหาและใส่หัวให้กับตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงบนและหาเส้นโครงร่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

** สำหรับตัวพยัญชนะ ห.หีบนั้น ได้ส่วนหัวที่อยู่ด้านหลังออกมาเนื่องจากหาพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดในช่วงบนแล้ว พื้นที่ด้านหลังมีค่ามากกว่าด้านหน้าจึงได้ส่วนหัวด้านหลังออกมา แสดงดังรูปที่ 5.27

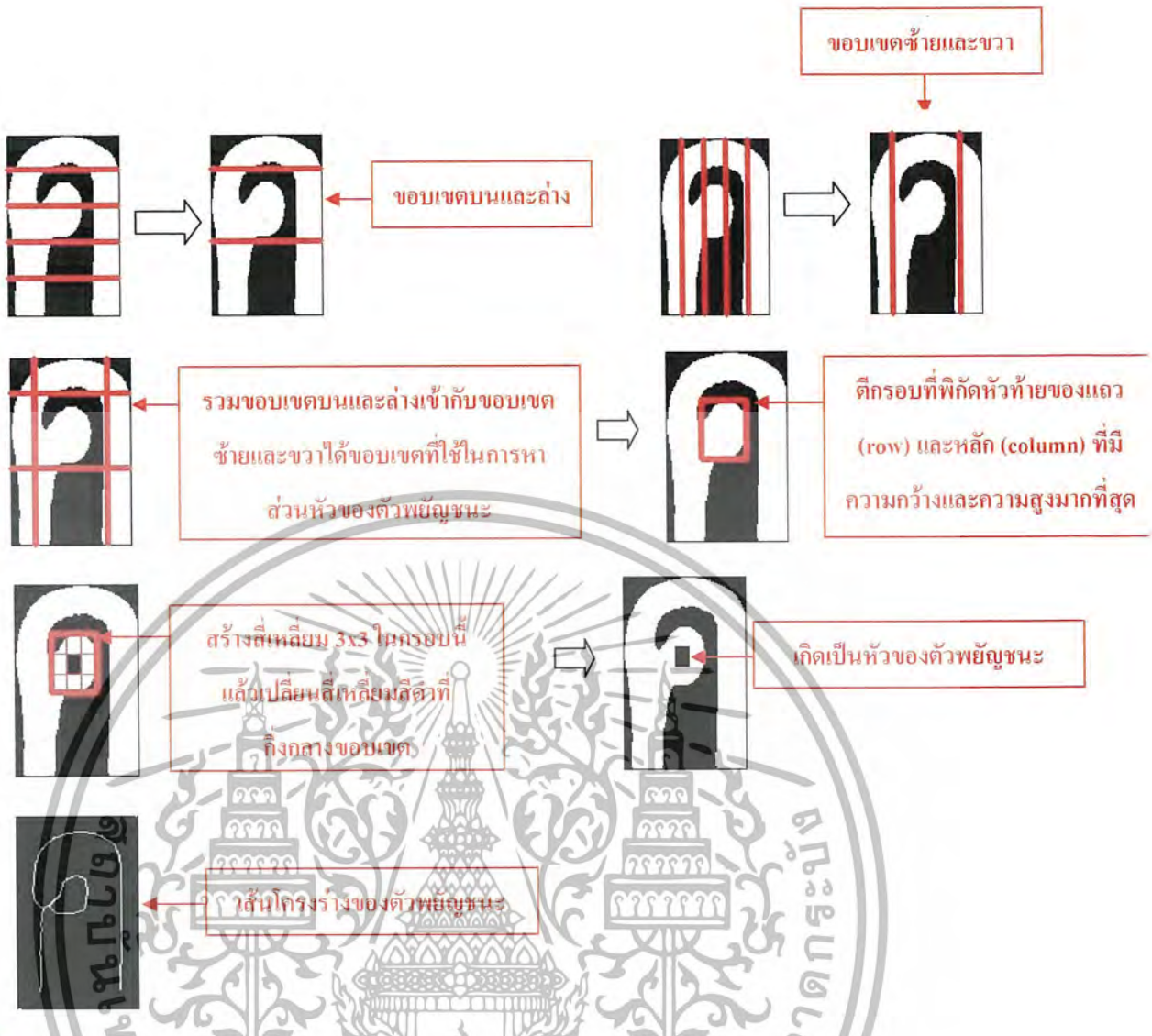



**รูปที่ 5.28** ภาพตัวพยัญชนะห.หีบ ที่ได้ส่วนหัวด้านหลังออกมาแทน

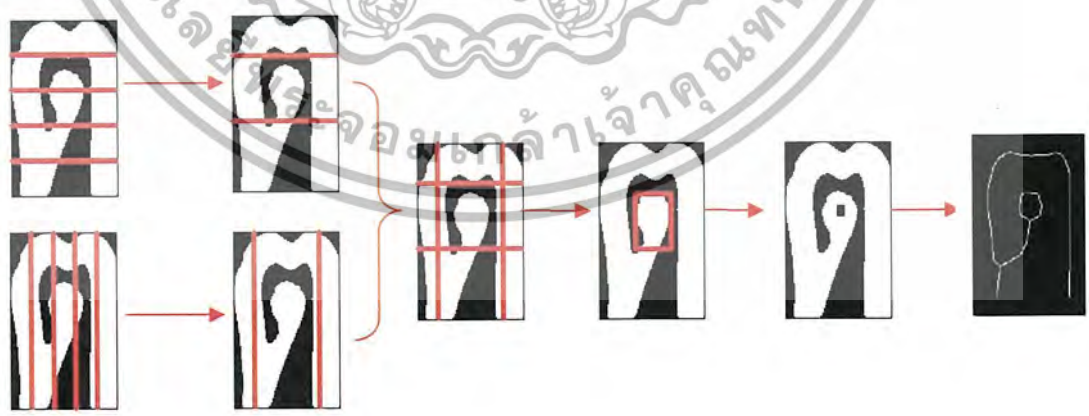
2. ส่วนหัวของตัวพยัญชนะอยู่ช่วงกลาง มีตัวพยัญชนะคือ ค, จ, ฉ, ด, ต, ศ, อ และ ฮ สามารถแบ่งได้เป็นสองประเภทคือส่วนหัวอยู่กึ่งกลางและส่วนหัวอยู่ช่วงกลางด้านหน้า ซึ่งตัวพยัญชนะทั้งสองประเภทนี้ใช้ขอบเขตในการหาส่วนหัวที่แตกต่างกัน ดังนี้

2.1) ส่วนหัวอยู่กึ่งกลาง ใช้ขอบเขตจากการแบ่งความสูงและความกว้างของตัวพยัญชนะเป็นห้าช่วงเท่าๆ กัน ขอบเขตบนและล่างใช้ช่วงที่สองถึงสามจากการแบ่งความสูงและขอบเขตซ้ายและขวาใช้ช่วงที่สองถึงสี่จากการแบ่งความกว้าง จากนั้นหาส่วนหัวของตัวพยัญชนะ โดยการหาพิกัดหัวท้ายของแถว (row) และหลัก (column) ที่มีความกว้างและความสูงมากที่สุดในขอบเขตนั้น จากนั้นสร้างสี่เหลี่ยม 3x3 ในกรอบนี้แล้วเปลี่ยนสี่เหลี่ยมที่กึ่งกลางเป็นสีดำ เกิดเป็นหัวของตัวพยัญชนะแล้วหาเส้นโครงร่างของตัวพยัญชนะ แสดงดังรูปที่ 5.28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



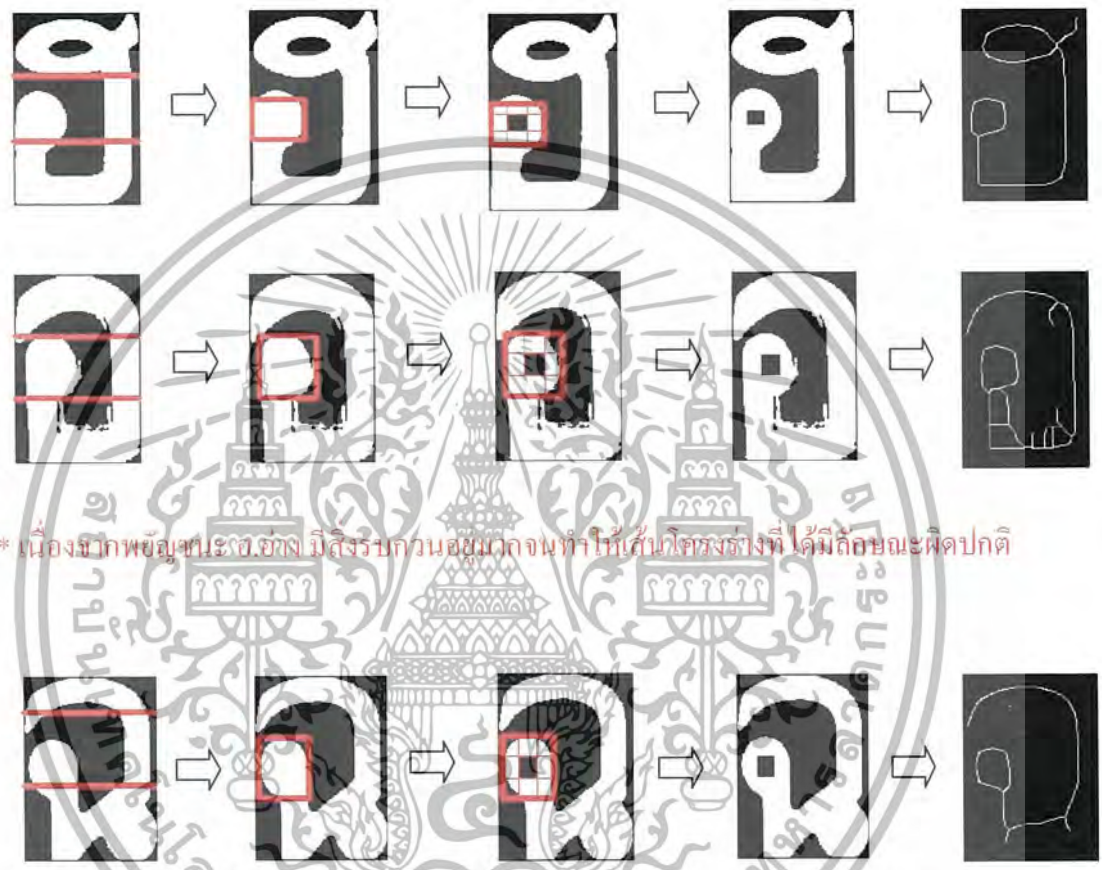

ตัวอย่าง การหาและใส่หัวให้ตัวพยัญชนะที่มีส่วนหัวอยู่กึ่งกลางและหาเส้นโครงร่างของตัวพยัญชนะ



**รูปที่ 5.29** ภาพการหาและใส่หัวให้กับตัวพยัญชนะที่มีส่วนหัวอยู่กึ่งกลาง
และหาเส้นโครงร่างของตัวพยัญชนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2) ส่วนหัวอยู่ช่วงกลางด้านหน้า ใช้ขอบเขตช่วงกลาง จากนั้นหาส่วนหัวของตัวพยัญชนะโดยการหาพิกัดหัวท้ายของแถว (row) และหลัก (column) ที่มีความกว้างและความสูงมากที่สุดในขอบเขตนั้น แล้วสร้างสี่เหลี่ยม 3x3 ในกรอบนี้แล้วเปลี่ยนสี่เหลี่ยมที่กึ่งกลางเป็นสีดำ เกิดเป็นหัวของตัวพยัญชนะ แล้วจึงหาเส้นโครงร่างของตัวพยัญชนะ แสดงดังรูปที่ 5.29



*** เนื่องจากพยัญชนะ ง.งู มีสิ่งรบกวนอยู่มากจนทำให้เส้นโครงร่างที่ได้มีลักษณะผิดปกติ

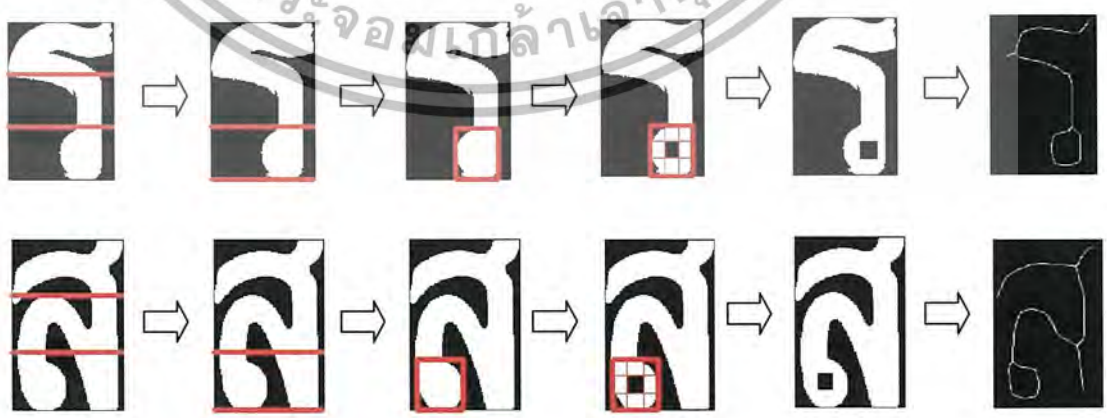
**รูปที่ 5.30** ภาพการหาและใส่หัวให้กับตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงกลางด้านหน้า และหาเส้นโครงร่างของตัวพยัญชนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

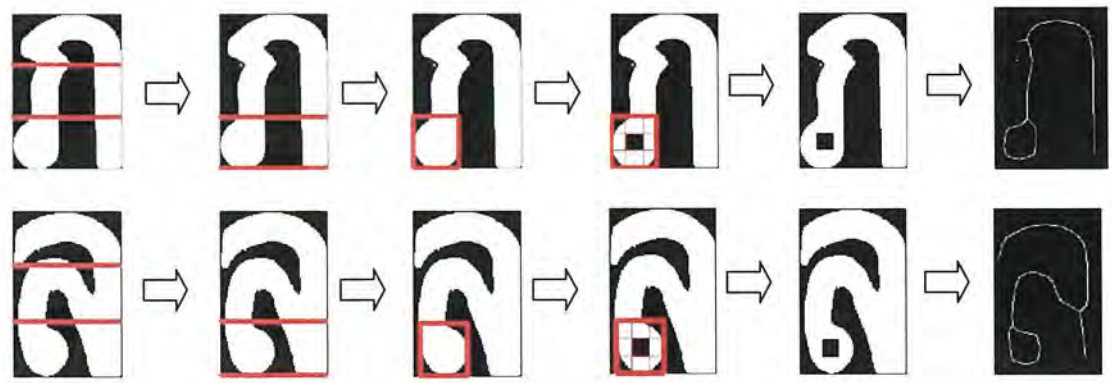
3. ส่วนหัวของตัวพยัญชนะอยู่ช่วงล่าง มีตัวพยัญชนะคือ ฌ, ญ, ฒ, ฏ, ภ, ร, ล, ว และ ส ส่วนหัวของตัวพยัญชนะอยู่ช่วงล่างจึงใช้ขอบเขตช่วงล่าง จากนั้นหาส่วนหัวของตัวพยัญชนะ โดยการหาพิกัดหัวท้ายของแถว (row) และหลัก (column) ที่มีความกว้างและความสูงมากที่สุดในขอบเขตช่วงล่าง จากนั้นสร้างสี่เหลี่ยม 3x3 ในกรอบนี้แล้วเปลี่ยนสี่เหลี่ยมที่กึ่งกลางเป็นสีดำ เกิดเป็นหัวของตัวพยัญชนะ แล้วจึงหาเส้นโครงร่างของตัวพยัญชนะ

**รูป 5.31** ภาพการหาและใส่หัวให้กับตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงล่าง และหาเส้นโครงร่างของตัวพยัญชนะ

ตัวอย่าง การหาและใส่หัวให้ตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงล่างแล้วจึงหาเส้นโครงร่างของตัวพยัญชนะ

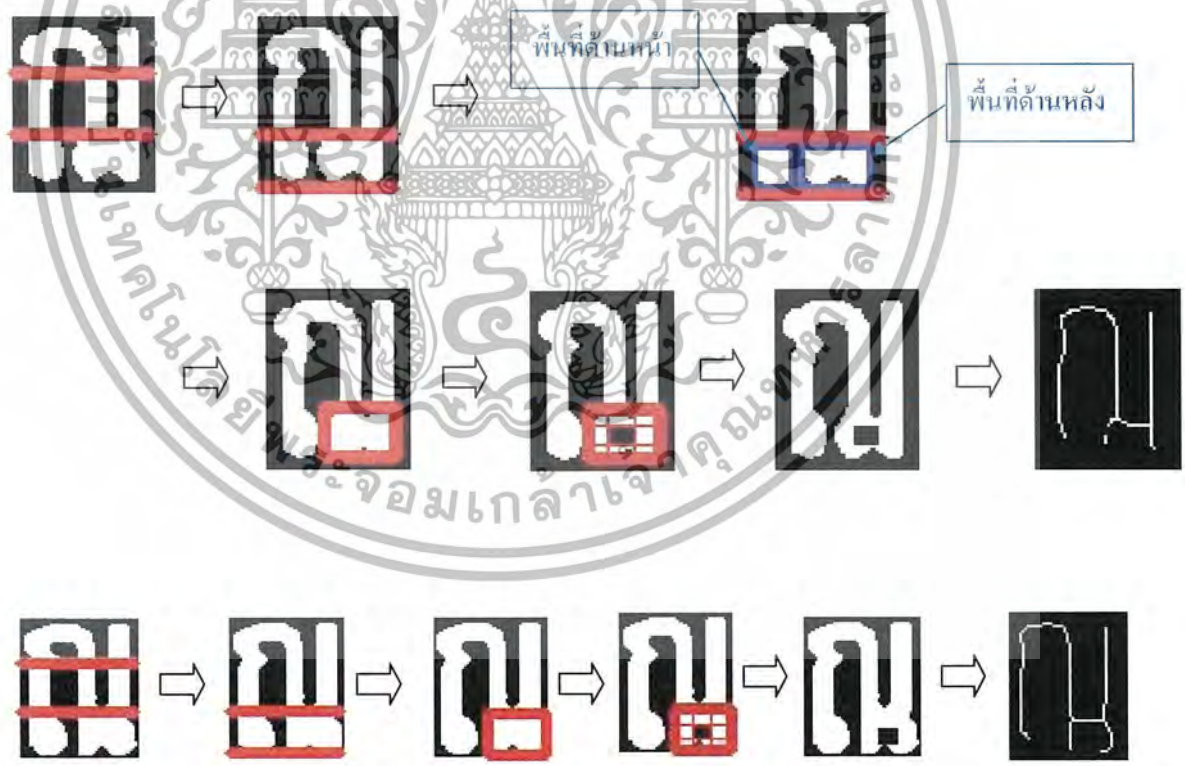


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 5.32** ภาพการหาและใส่หัวให้ตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงล่างแล้วหาเส้นโครงร่าง

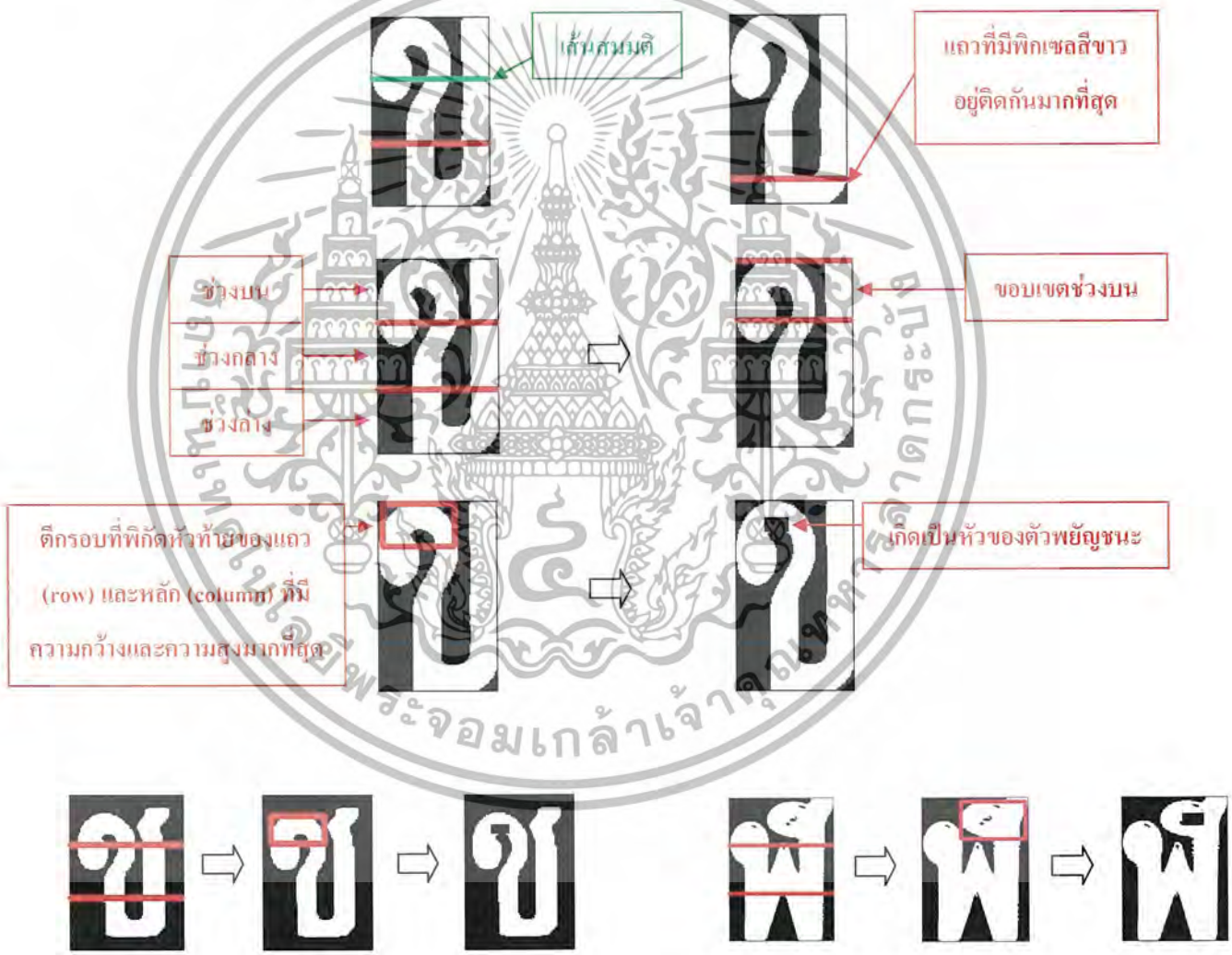
*** การหาและใส่หัวให้กับตัวพยัญชนะ จะเกิดปัญหาขึ้นกับตัวพยัญชนะ ณ.เณร และ ฌ.กระเฌอ เนื่องจากมีสองหัวในช่วงล่าง แต่ได้ส่วนหัวที่อยู่ด้านหลังออกมาเนื่องจากหาพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดในช่วงล่างแล้วพื้นที่ด้านหลังมีค่ามากกว่าด้านหน้าจึงได้ส่วนหัวด้านหลังออกมา




**รูปที่ 5.33** ภาพการหาและใส่หัวให้ตัวพยัญชนะที่มีส่วนหัวอยู่ช่วงล่าง 2 หัว และได้ด้านหลังออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับตัวพยัญชนะที่เกิดปัญหา ส่วนหัวของตัวพยัญชนะไม่อยู่ในสามช่วงนั้นอย่างพอดี แต่จะอยู่ระหว่างช่วงบนกับช่วงกลาง อาทิเช่น ข,ช,พ เป็นต้น ตัวพยัญชนะในกลุ่มนี้ใช้ขอบเขตช่วงบนใบการหาพื้นที่ส่วนหัวของตัวพยัญชนะ เนื่องจากเข้าเงื่อนไขแถวที่มีพิกเซลสีขาวอยู่ติดกันมากที่สุดอยู่ต่ำกว่าเส้นสมมติจึงเข้าเงื่อนไขหัวอยู่ช่วงบน จากนั้นหาพิกัดหัวท้ายของแถว (row) และหลัก (column) ที่มีความกว้างและความสูงมากที่สุดในขอบเขตนั้นแล้วสร้างสี่เหลี่ยม 3x3 ในกรอบนี้แล้วเปลี่ยนสี่เหลี่ยมที่กึ่งกลางเป็นสีดำ เกิดเป็นหัวของตัวพยัญชนะ ดังรูป 5.37 พบว่าสี่เหลี่ยมสีดำอยู่ไม่พอดีกับส่วนหัวเนื่องจากกรอบที่ตีจากหาพิกัดหัวท้ายของแถว (row) และหลัก (column) ที่มีความกว้างและความสูงมากที่สุดในขอบเขตนี้ไม่ใช่ส่วนหัวของตัวพยัญชนะ




**รูป 5.34** ตัวพยัญชนะที่เกิดปัญหาขึ้นจากสาเหตุที่หัวของตัวพยัญชนะไม่อยู่ในตำแหน่งพอดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางสรุปผลการทดลองการหาและใส่หัวให้กับตัวพยัญชนะ**

| ตำแหน่งส่วนหัวของตัวพยัญชนะ | จำนวนตัวพยัญชนะทั้งหมด | ตัวพยัญชนะที่หาได้ |
|---|---|---|
| 1. พยัญชนะที่ส่วนหัวอยู่ช่องบน | ง ท น บ ป ผ ฝ พ ฟ ม ย ษ ห | บ ป ผ ฝ พ ฟ ง น ม ย ษ ห |
| 2. พยัญชนะที่ส่วนหัวอยู่ช่วงกลาง | ค ศ จ ฉ ฌ ด ต ศ อ ฮ | อ ฮ ด ต ค จ |
| 3. พยัญชนะที่ส่วนหัวอยู่ช่วงล่าง | ฌ ณ ญ ฎ ฏ ภ ร ล ว ส ถ | ร ภ ล ส ถ ว ณ ฌ |
| **4. พยัญชนะที่ส่วนหัวอยู่ระหว่างสองช่วง | ข ฃ ช ซ ฆ ฬ ฐ | - |
| 5. พยัญชนะที่ไม่มีหัว | ก ฐ | - |

** พยัญชนะที่เกิดปัญหาไม่สามารถหาส่วนหัวได้ถูกต้อง

สำหรับตัวพยัญชนะที่ไม่มีหัว เช่น ก.ไก่ กับ ธ.ธง ไม่ได้ถูกนำมาทดสอบ เพราะเป็นตัวพยัญชนะที่ไม่มีหัว ซึ่งการหาเส้นโครงร่างของตัวพยัญชนะเหล่านี้สามารถหาได้ตามวิธีดังที่กล่าวไว้แล้วข้างต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 6
# สรุปและวิจารณ์ผลการทดลอง

## ตอนที่ 1 การแยกส่วนที่เป็นตัวเลขและตัวพยัญชนะออกจากแผ่นป้ายทะเบียน การรู้จำตัวเลขและตัวพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ด้วยวิธีการ Template Matching

จากผลการทดลองพบว่า โปรแกรมค้นหาแผ่นป้ายทะเบียนรถยนต์ที่ได้ทำการแก้ไขในส่วนของภาพแม่แบบใหม่ โดยการกำหนดขนาดของแม่แบบใหม่ทั้งความสูงและความกว้างของตัวเลขและตัวพยัญชนะ ซึ่งต่างจากของรุ่นพี่ที่กำหนดแต่ความสูงเพียงของตัวเลขและตัวพยัญชนะเพียงอย่างเดียว เมื่อนำมาหาค่าสัมประสิทธิ์ความเหมือนกับตัวเลขและตัวพยัญชนะที่ได้จากแผ่นป้ายทะเบียนแล้ว ผลลัทธ์ที่ได้พบว่ามีค่าสัมประสิทธิ์ความเหมือนที่ดีขึ้นกว่าเดิม เหมือนเทียบกับผลการทดลองของรุ่นพี่

โดยในการทดลองโปรแกรมการค้นหาแผ่นป้ายทะเบียนรถยนต์ ให้ค้นหาแผ่นป้ายจากรูปถ่ายด้านหน้ารถยนต์ซึ่งเป็นรูปสี RGB ซึ่งมีจำนวนภาพที่นำมาทดลองทั้งหมด 40 รูป โดยการค้นหาแผ่นป้ายทะเบียน สามารถค้นหาส่วนของแผ่นป้ายทะเบียนออกมาได้ทั้งหมด 36 รูป คิดเป็น 90% และยังสามารถแยกตัวเลขและตัวพยัญชนะออกจากแผ่นป้ายทะเบียนทีละตัวได้ครบสมบูรณ์ทั้งหมด 36 รูป คิดเป็น 90%

การค้นหาแผ่นป้ายทะเบียนจากภาพสี RGB ได้นั้น ในขั้นตอนการแปลงเป็นภาพสองระดับจะต้องไม่มีพิกเซลสีขาวตรงแผ่นป้ายทะเบียนไปติดกับขอบภาพ เพราะจะทำให้แผ่นป้ายทะเบียนถูกกำจัดออกไปเมื่อใช้คำสั่ง imclearborder ซึ่งทำให้ไม่สามารถหาแผ่นป้ายทะเบียนออกมาได้

เมื่อได้แผ่นป้ายทะเบียนออกมาแล้ว ก็จะทำการแยกตัวเลขและตัวพยัญชนะทั้งหมดบนแผ่นป้ายออกมาทีละตัวโดยต้องเรียงลำดับให้เหมือนบนแผ่นป้าย การแยกตัวเลขและตัวพยัญชนะออกมาทีละตัว จะต้องทำการกำจัดสิ่งอื่นที่ไม่ใช่ตัวเลขหรือตัวพยัญชนะออกไปก่อน เช่น ลวดลายของแผ่นป้าย สิ่งสกปรกที่ติดอยู่น๊อตหรือตะปูที่คั่นระหว่างตัวเลขกับตัวพยัญชนะ เพราะน๊อตและตะปูจะเชื่อมตัวเลขหรือตัวพยัญชนะให้มีบางส่วนติดกัน ซึ่งในการกำหนดหมายเลขเพื่อแยกตัวเลขและตัวพยัญชนะออกจากแผ่นป้ายทะเบียน โปรแกรมจะประมวลผลให้ตัวเลขและตัวพยัญชนะที่ติดกันเป็นตัวเลขหรือตัวพยัญชนะตัวเดียวกัน ทำให้ไม่สามารถแยกตัวเลขและตัวพยัญชนะออกจากแผ่นได้ทะเบียนรถยนต์ได้ครบสมบูรณ์ทุกตัว

ในการรู้จำตัวเลขหรือตัวพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ จากการทดลองพบว่าถ้าโปรแกรมสามารถแยกตัวเลขและตัวพยัญชนะออกจากแผ่นป้ายทะเบียนได้ครบทุกตัว ก็จะสามารถรู้จำตัวเลขและตัวพยัญชนะที่แยกจากแผ่นป้ายได้ครบทุกตัวเช่นกัน แต่สิ่งสำคัญคือ ตัวเลขและตัวพยัญชนะจะต้องมีความคมชัด มีความชัดเจนและไม่เอียงจนเกินไป เพราะเมื่อนำไปวาบทาบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กับภาพชุดแม่แบบเพื่อหาค่าสัมประสิทธ์ความเหมือนแล้ว ถ้าค่าสัมประสิทธ์ความเหมือนต่ำกว่าที่กำหนด โปรแกรมก็จะไม่แสดงผลการรู้จำตัวเลขหรือตัวพยัญชนะออกมาหรืออาจจะแสดงตัวเลขหรือตัวพยัญชนะที่ไม่ตรงตามแผ่นป้ายทะเบียนที่ทำการทดลอง

**ตอนที่ 2 การใส่หัวให้กับตัวพยัญชนะและการหาเส้นโคร่งร่างของตัวพยัญชนะ**

จากการทดลองพบว่า เมื่อนำตัวพยัญชนะที่ได้จากแผ่นป้ายทะเบียนซึ่งตัวพยัญชนะจะมีลักษณะเป็นตัวทึบมาทำการหาเส้นโครงร่าง โปรแกรมการหาเส้นโครงร่างของตัวพยัญชนะก็สามารถหาโครงร่างได้ทั้งหมด 44 พยัญชนะไทย โดยไม่สนใจว่าขนาดของพยัญชนะที่ใช้จะเล็กหรือใหญ่ ก็สามารถทำการหาเส้นโครงร่างได้แต่เกิดปัญหาตามมาหลังจากนั้น คือหลังจากที่ทำการหาเส้นโครงร่างเสร็จแล้ว ความคล้ายคลึงกันของตัวพยัญชนะไทยบางตัวทำให้ไม่สามารถแยกแยะได้ว่าเส้นโครงร่างนี้เป็นตัวพยัญชนะตัวใด เมื่อแยกแยะได้ยากแล้วถ้านำเส้นโครงร่างนี้ไปทำการรู้จำตัวอักษร ก็อาจจะเกิดข้อผิดพลาดได้

จึงได้มีคิดวิธีแก้ไขลักษณะความคล้ายคลึงกันของตัวพยัญชนะได้โดย การนำพื้นที่ที่มากที่สุดบนตัวพยัญชนะมาใช้ เนื่องจากมีความคิดที่ว่าพื้นที่ที่มากที่สุดของตัวพยัญชนะแต่ละตัวอาจจะเป็นส่วนหัวของตัวพยัญชนะจึงทำการดึงส่วนนั้นออกมาเพื่อที่จะสร้างสี่เหลี่ยมลงไป เพื่อทำให้หัวของตัวพยัญชนะจากที่มีลักษณะทึบจะกลายเป็นลักษณะโปร่งแทน เมื่อทำการหาเส้นโครงร่างก็จะทำให้เส้นโครงร่างของตัวพยัญชนะแต่ละตัวสามารถแยกแยะได้ง่ายยิ่งขึ้น

จากการทดลองพบว่าการดึงพื้นที่ที่มากที่สุดขึ้นมาแล้ว สร้างสี่เหลี่ยมลงไปแทนนั้น โปรแกรมสามารถดึงพื้นที่ที่มากที่สุดซึ่งเป็นหัวของตัวพยัญชนะได้ถูกต้องเพียง 8 ตัวพยัญชนะเท่านั้น คือ ย ม น ภ ห บ ป และ ษ ส่วนอีก 36 ตัวที่เหลือ พื้นที่ที่มากที่สุดไม่ได้เป็นส่วนหัวตัวพยัญชนะอย่างที่ต้องการ แต่เป็นส่วนอื่นของตัวพยัญชนะแทน เช่นส่วนฐาน เป็นต้น

ดังนั้นจึงมีแนวความคิดใหม่ในการแก้ไขความคล้ายคลึงกันของตัวพยัญชนะ คือการหาความหนาของตัวพยัญชนะ จากการหาความหนาของตัวพยัญชนะจำเป็นจะต้องจำแนกตัวพยัญชนะออกเป็น 3 จำพวกดังนี้

1. ตัวพยัญชนะที่ส่วนหัวอยู่ด้านบน มีดังนี้ ข ฃ ง ช ซ ฌ ท น บ ป ผ ฝ พ ฟ ม ย ษ ห ฬ
2. ตัวพยัญชนะที่ส่วนหัวอยู่ตรงกลาง มีดังนี้ ค ฅ จ ฉ ฐ ฒ ด ต ศ อ ฮ
3. ตัวพยัญชนะที่ส่วนหัวอยู่ด้านล่าง มีดังนี้ ฌ ณ ญ ฎ ฏ ภ ร ล ว ส ฬ

แต่ตัวพยัญชนะที่นำมาใช้ในการทดลอง จะเป็นตัวพยัญชนะที่ได้มาจากแผ่นป้ายทะเบียนซึ่งบางตัวไม่สามารถหาได้

เมื่อนำตัวพยัญชนะที่ได้จากแผ่นป้ายทะเบียนมาแยกจำพวก และเข้าโปรแกรมการหาความหนาของตัวพยัญชนะ เพื่อให้ได้ส่วนหัวของตัวพยัญชนะขึ้นมาแล้วทำการสร้างสี่เหลี่ยมลงไปเพื่อแก้ไขลักษณตัวพยัญชนะจากหัวทึบให้มีลักษณะหัวโปร่ง ผลการทดลองมีดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ตัวพยัญชนะที่ส่วนหัวอยู่ด้านบน ที่โปรแกรมสามารถทำได้มีทั้งหมด 12 ตัว จาก 20 ตัว คือ บ ป ผ ฝ พ ฟ ง น ม ย ษ ห จาก ข ฃ ง ช ซ ฌ ท น บ ป ผ ฝ พ ฟ ม ย ษ ห ฬ คิดเป็น 60% จากทั้งหมด 20 ตัวพยัญชนะ สำหรับตัวพยัญชนะที่ไม่สามารถทำได้ เนื่องจากหัวของตัวพยัญชนเหล่านั้นมีขนาดเล็กหรือไม่อยู่ในตำแหน่งด้านบนแบบปกติ

2. ตัวพยัญชนะที่ส่วนหัวอยู่ตรงกลาง ที่โปรแกรมสามารถทำได้มีทั้งหมด 6 ตัว จาก 10 ตัว คือ อ ฮ ด ต ค จ จาก ค ศ จ ฉ ฐ ฒ ด ต ศ อ ฮ คิดเป็น 60% จากทั้งหมด 10 ตัวพยัญชนะ สำหรับตัวพยัญชนะที่ไม่สามารถทำได้เนื่องจากมีความซับซ้อนมากกว่าตัวพยัญชนะปกติ

3. ตัวพยัญชนะที่ส่วนหัวอยู่ด้านล่าง ที่โปรแกรมสามารถทำได้มีทั้งหมด 8 ตัว จาก 10 ตัว คือ ร ภ ล ส ถ ว ณ ฌ จาก ญ ฎ ฏ ร ล ว ส ถ คิดเป็น 80% จากทั้งหมด 10 ตัวพยัญชนะ สำหรับตัวพยัญชนะที่ไม่สามารถทำได้เนื่องจากมีความยาวเกินกว่าความยาวตัวพยัญชนะปกติ

ผลรวมทั้งหมดที่โปรแกรมสามารถทำได้ก็คือ 26 ตัว จากพยัญชนะไทยทั้งหมด 44 ตัว คิดเป็น 59.09% จากทั้งหมด 44 ตัว

หลังจากนั้นก็ตัวพยัญชนะที่ผ่านการทดลองที่ได้ 26 ตัว ไปทำการหาเส้นโครงร่าง จะเห็นได้ชัดเจนว่า เส้นโครงร่างของตัวพยัญชนะ 26 ตัว ที่ผ่านการแก้ไขตามแนวคิดนี้จะมีความชัดเจนและสามารถแยกแยะตัวพยัญชนะแต่ละตัวออกจากกันได้ง่ายมากยิ่งขึ้น เมื่อเทียบกับเส้นโครงร่างของตัวพยัญชนะตัวเดียวกันแต่ไม่ได้ผ่านการแก้ไขตามแนวคิดนี้

อย่างไรก็ตามการทำวิจัย ในส่วนการรู้จำตัวอักษรบนพื้นฐานลักษณะเด่นไม่ก้าวหน้าเท่าที่ควรจะเป็น เนื่องจากตัวพยัญชนะที่ใช้บนแผ่นป้ายทะเบียนรถยนต์เป็นชนิดหัวทึบทำให้ไม่เป็นเรื่องง่ายเลยที่จะหาส่วนที่เป็นหัวของพยัญชนะได้ คณะผู้วิจัยได้ศึกษาค้นคว้าหาวิธีการต่างๆ พร้อมกับทดสอบกับภาพจริงจนในที่สุดก็สามารถหาได้ อย่างไรก็ตามเนื่องจากเสียเวลาส่วนใหญ่กับการหาส่วนที่เป็นหัวของตัวพยัญชนะ ทำให้เหลืองานส่วนเป็นการหาลักษณะเด่นบนตัวพยัญชนะ ดังนั้นงานในส่วนนี้จึงยังไม่เป็นไปตามเป้าหมายที่ได้กำหนดไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ข้อเสนอแนะ

**ตอนที่ 1 การแยกส่วนที่เป็นตัวเลขและตัวพยัญชนะออกจากแผ่นป้ายทะเบียน การรู้จำตัวเลขและตัวพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ด้วยวิธีการ Template Matching**

ความผิดพลาดโดยส่วนมากจะเกิดขึ้น ในขั้นตอนการค้นหาส่วนของแผ่นป้ายทะเบียนออกมาจากด้านหน้าของรถยนต์ จึงมีวิธีแก้ไขดังนี้

ในการเก็บภาพถ่ายด้านหน้าของรถยนต์นั้น จะต้องมีการตั้งระยะการถ่ายภาพให้ห่างจากด้านหน้ารถยนต์อย่างคงที่ มีการปรับความสว่างของภาพที่กำลังจะถ่ายให้เหมาะสมกับช่วงเวลาที่ทำการถ่ายภาพ รวมทั้งจะต้องทำการตั้งกล้องถ่ายภาพให้ตรงเพื่อให้ได้ภาพด้านหน้ารถยนต์ที่มีลักษณะตรง ถ้าทำได้ครบตามที่กล่าวทั้งหมดเมื่อนำภาพที่ถ่ายมาเข้าโปรแกรมการค้นหาแผ่นป้ายทะเบียนจะมีความเป็นไปได้ที่จะได้ส่วนของแผ่นป้ายทะเบียนออกมามากขึ้นกว่าเดิม

**ตอนที่ 2 การใส่หัวให้กับตัวพยัญชนะ**

ในการหาความหนาของตัวพยัญชนะ เพื่อที่จะแก้ไขความคล้ายคลึงกันของตัวพยัญชนะบางตัว จะต้องทำการจำแนกจำพวกของตัวพยัญชนะให้ถูกต้อง คือตัวพยัญชนะที่ส่วนหัวอยู่ด้านบน ตัวพยัญชนะที่ส่วนหัวอยู่ตรงกลางและตัวพยัญชนะที่ส่วนหัวอยู่ด้านล่าง ถ้าทำการจำแนกผิดจำพวกเมื่อเข้าโปรแกรมการหาความหนาของตัวพยัญชนะ ก็จะเกิดข้อผิดพลาดขึ้นอย่างแน่นอน เนื่องจากโปรแกรมที่ใช้จะมีเงื่อนไขการทำงานที่แตกต่างกันตามจำพวกของพยัญชนะ และก็ยังขึ้นอยู่กับโปรแกรมที่เขียนว่ามีเงื่อนไขครอบคุมมากน้อยเพียงใด

## วิจารณ์ผลการทดลอง

**ตอนที่ 1 การแยกส่วนที่เป็นตัวเลขและตัวพยัญชนะออกจากแผ่นป้ายทะเบียนการรู้จำตัวเลขและตัวพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ด้วยวิธีการ Template Matching**

จากผลการทดลองพบว่าความผิดพลาดจากผลการทดลองเกิดจากนำภาพที่ต้องการพิจารณาของตัวเลขและตัวพยัญชนะมีความไม่สมบูรณ์ของแผ่นป้ายทะเบียน เช่น มีสิ่งรบกวนอยู่บนแผ่นป้ายทะเบียนทำให้แผ่นป้ายทะเบียนมีความไม่ชัดเจน เมื่อนำภาพที่ต้องการพิจารณาของตัวเลขและตัวพยัญชนะไปวางทาบกันกับภาพแม่แบบ ทำให้ค่าสัมประสิทธิ์ความเหมือน(Correlation) มีค่าที่ลดน้อยลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตอนที่ 2 การใส่หัวให้กับตัวพยัญชนะ**

จากผลการทดลองหาเส้นโครงร่างของตัวพยัญชนะนั้น พบว่าตัวพยัญชนะบนแผ่นป้ายทะเบียนที่มีลักษณะหัวทึบเมื่อหาเส้นโครงร่างแล้วเกิดความคลาดเคลื่องกันขึ้นจึงมีแนวคิดในการเพิ่มลักษณะเด่น คือการใส่หัวให้กับตัวพยัญชนะจากการทดลองการใส่หัวให้กับตัวพยัญชนะโดยแยกประเภทตามตำแหน่งส่วนหัวของตัวพยัญชนะ พบว่าตัวพยัญชนะที่มีลักษณะของส่วนหัวซับซ้อน อาทิเช่น ตัวพยัญชนะ ข, ช, ฑ, ฒ, ฎ เป็นต้น ที่มีส่วนหัวอยู่ในตำแหน่งที่ไม่ใช่ช่วงบน ช่วงกลาง และช่วงล่างอย่างชัดเจนจะมีปัญหาคือไม่สามารถใส่หัวได้ตามตำแหน่งที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บรรณานุกรม

[1] Rael C. Gonzalez University of Ten nesses, Richard E. Woods MedData Interactive, Steven L. Eddins "Digital Image Processing Using MATLAB"

[2] Gerard Blanchet, Maurice Charbit "Digital Singnal and Image Processing MATLAB"

[3] หทัยทิพย์ สิริพงศ์พนธ์, อธิศักดิ์ สาธุประคัลภ์, อัญมณี เกิดเพชร "License Plate Detectionn การรู้จำป้ายทะเบียนรถยนต์" สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปริญญานิพนธ์ปีการศึกษา 2551

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการเรียกข้อมูลรูปภาพขึ้นมาจากไฟล์ภาพโดยใช้นามสกุลภาพเป็น .JPG เก็บไว้ในตัวแปร I

```
I=imread('C:\Users\SoDa\Desktop\mm\39.jpg');
```

โปรแกรมแสดงรูปภาพที่เรียกขึ้นมาจากตัวแปร I

```
figure(1),imshow(I);
```

โปรแกรมการแปลงภาพสี RGB ให้เป็นภาพ Gray scale โดยใช้คำสั่ง rgb2gray

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

โปรแกรมการแปลงภาพ Gray scale ให้เป็นภาพ Binary โดยใช้วิธีการตัดระดับเทรชโฮลอัตโนมัติ

```
%--หาค่าเทรชโฮลจากภาพ Gray Scale ที่เก็บไว้ในตัวแปร J แล้วเก็บไว้ที่ตัวแปร level
level=graythresh(J)

%----แปลงภาพ Gray Scale ที่เก็บไว้ในตัวแปร J ให้เป็นภาพ Binary โดยใช้ค่าเทรชโฮลที่ได้จากตัวแปร
level แล้วเก็บไว้ในตัวแปร bw
bw=im2bw(J,level)
```

โปรแกรมวิธีการตัดระดับเทรชโฮลแบบ Manual

```
%--หาค่าเทรชโฮลโดยการปรับค่าเองโดยการบวกหรือลบค่าเพื่อให้ได้ค่าเทรชโฮลที่เหมาะสม
level = level-0.2;
```

โปรแกรมการลบพื้นที่ส่วนติดกับขอบออกไป โดยการเปลี่ยนกลุ่มพิกเซลสีขาวให้เป็นกลุ่มพิกเซลสีดำ

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการค้นหาภาพส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์

```
%----กำหนดโครงสร้างที่จะนำไปกัดภาพเป็นโครงสร้างแบบ square แล้วเก็บไว้ที่ตัวแปร se
se=strel('square',4);
%----ทำการกัดภาพ Binary ที่เก็บไว้ในตัวแปร BWnobord ด้วยโครงสร้างที่เก็บไว้ในตัวแปร
%----se แล้วนำภาพผลลัพทธ์ที่ได้จากการกัดไปเก็บไว้ในตัวแปร erodedBW
erodebw=imerode(bwnobord,se);
%----ทำการ Label ให้กับภาพที่เก็บไว้ในตัวแปร erodedBW ส่วนตัวแปร numObjects
%----จะเป็นตัวแปรที่เก็บจำนวนกลุ่มพิกเซลสีขาวที่มีอยู่ภาพ Binary นั้นไว้
[labeled,numobjects]=bwlabel(erodebw,4);
squaredata11=regionprops(labeled,'basic');
%----ตัวแปร allsquares11 จะเก็บค่าจำนวนพิกเซลของทุกกลุ่มพิกเซลสีขาวไว้
allsquares11=[squaredata11.Area];
%----ตัวแปร maxArea11 จะเก็บค่าจำนวนพิกเซลของกลุ่มพิกเซลสีขาวที่มีพื้นที่มากที่สุดจาก
%----ตัวแปร allsquares
maxArea11=max(allsquares11);
%----ตัวแปร biggestGrains11 จะหากลุ่มพิกเซลสีขาวในภาพ Binary ที่มีพื้นที่พื้นที่
%----เท่ากับจำนวนพิกเซลที่เก็บไว้ในตัวแปร 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 แล้วเก็บภาพที่ได้จากการคัดลอกไว้ในตัวแปร U
U=I(i11min:i11max,j11min:j11max);
```

โปรแกรมการ Inverse Binary ของภาพแผ่นป้ายทะเบียนรถยนต์ที่ได้

```
%----หาค่าเทรชโฮลอัตโนมัติจากภาพ RGB ที่เก็บไว้ในตัวแปร U แล้วเก็บค่าเทรชโฮลที่ได้
%----ไว้ในตัวแปร level2
level2=graythresh(U);
%----แปลงภาพ RGB ที่เก็บไว้ในตัวแปร A ให้เป็นภาพ Binary โดยใช้ค่าเทรชโฮลที่
%----เก็บไว้ในตัวแปร level2 แล้วเก็บภาพ Binary ที่ได้ไว้ในตัวแปร bw2
bw2=im2bw(U,level2);
[m,n]=size(bw2);
for j=1:n
    for i=1:m
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----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
```

โปรแกรมการกำจัดส่วนที่เป็นชื่อจังหวัดของแผ่นป้ายทะเบียนออกไป

```
%----ทำการ Label ให้กับภาพ Binary ของส่วนที่เป็นแผ่นป้ายทะเบียนรถยนต์ที่โปรแกรมค้นหา
%----ได้ซึ่งเก็บไว้ในตัวแปร PICTURE แล้วเก็บภาพผลลัพธ์ที่ได้จากการ Label ไว้ที่ตัวแปร Labeled2
[labeled2,numObjects2]=bwlabel(picture,4);
squaredata2=regionprops(labeled2,'basic');
%----ทำการสำรวจทุกกลุ่มพิกเซลสีขาวที่มีอยู่ภาพ Binary ที่เก็บไว้ในตัวแปร labeled2
for u2=1:numObjects2
    [rr2,cc2]=find([labeled2==u2])
%----หาขอบเขตของแต่ละกลุ่มพิกเซลสีขาว
    ii2max=max(rr2); %----เก็บค่าขอบล่างไว้ในตัวแปร ii16max
    ii2min=min(rr2); %----เก็บค่าขอบบนไว้ในตัวแปร ii16min
    jj2max=max(cc2);%----เก็บค่าขอบขวาไว้ในตัวแปร jj16max
    jj2min=min(cc2); %----เก็บค่าขอบซ้ายไว้ในตัวแปร jj16min
%----นำค่าขอบล่างลบด้วยขอบบน จะทำให้ได้ความสูงของกลุ่มพิกเซลสีขาว แล้วเก็บไว้ในตัวแปร h
    h=ii2max-ii2min
%----นำค่าขอบขวาลบด้วยค่าขอบซ้าย ทำให้ได้ความกว้างของกลุ่มพิกเซลสีขาว แล้วเก็บไว้ในตัวแปร w
    w=jj2max-jj2min
%----ถ้ากลุ่มพิกเซลสีขาวนั้นมีความสูงน้อยกว่า 0.25 เท่าของภาพ Binary ที่เก็บไว้ในตัวแปร PICTURE
%----เปลี่ยนกลุ่มพิกเซลสีขาวนั้นให้เป็นพิกเซลสีดำทั้งหมด
      if w<(0.25*m)
       picture(labeled2==u2)=0;
    end
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการกำจัดส่วนที่เป็นพิกเซลสีขาวที่ติดกับขอบภาพและกำจัดสิ่งรบกวนที่อยู่บนแผ่นป้ายทะเบียนรถยนต์

```
[j,k]=size(picture);
%----ทำให้แถวที่ 1 ถึง 3 ของภาพที่เก็บไว้ในตัวแปร PICTURE เป็นสีดำทั้งหมด เพื่อป้องกัน
%----ไม่ให้ส่วนที่เป็นตัวเลขและพยัญชนะที่อยู่บนแผ่นป้ายถูกลบออกไปด้วยคำสั่ง 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 แล้วเก็บภาพผลลัพธ์ไว้ในตัวแปร labeled17
[labeled17,numObjects17]=bwlabel(BW2,4);
[j1,k1]=size(labeled17);
%----ทำการสำรวจทุกกลุ่มพิกเซลสีขาว
for u17=numObjects17
    [rr17,cc17]=find(labeled17==u17);
%----หาขอบเขตของแต่ละกลุ่มพิกเซลสีขาว
    ii17max=max(rr17);  %----เก็บค่าขอบล่างของกลุ่มพิกเซลสีขาวไว้ในตัวแปร ii17max
    ii17min=min(rr17); %----เก็บค่าขอบบนของกลุ่มพิกเซลสีขาวไว้ในตัวแปร  ii17min
    jj17max=max(cc17);  %---- เก็บค่าขอบขวาของกลุ่มพิกเซลสีขาวไว้ในตัวแปร jj17max
    jj17min=min(cc17);  %----เก็บค่าขอบซ้ายของกลุ่มพิกเซลสีขาวไว้ในตัวแปร jj17min
%----หาความสูงของกลุ่มพิกเซลสีขาว ซึ่งหาได้จากนำค่าขอบล่างลบด้วยค่าขอบบน แล้วเก็บไว้ในตัว
%----แปร high
    high=ii17max-ii17min;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----หาความกว้างของกลุ่มพิกเซลสีขาว ซึ่งหาได้จากนำค่าขอบขวาลบด้วยค่าขอบซ้าย แล้วเก็บ
%----ไว้ในตัวแปร width
width=jj17max-jj17min;
%----ถ้ากลุ่มพิกเซลสีขาวนั้นมีความสูงน้อยกว่า 0.3 หรือมากกว่า 0.75 เท่าของความสูงของภาพ
%----Binary หรือ มีความกว้างน้อยกว่า 0.025 เท่าของความกว้างของภาพ Binary หรือมี
%----อัตราส่วนความสูงต่อความกว้างน้อยกว่าหรือเท่ากับ 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
```

โปรแกรมการแบ่งส่วนที่เป็นขอบเขตบนและขอบเขตล่างของตัวเลขและตัวพยัญชนะ

```
%----ทำการหาขนาดความกว้างและความสูงแล้วเก็บไว้ที่ตัวแปร BW2
[x,y]=size(BW2);
%----
for i=1:15
    A=0;
    for j=1:x-1
        A=A+BW2(i,j);
    end
    if A>1
        TOP=i,break
    end
end
TOP=0;
LOW=0;
for i=1:y-1
    A=0;
    for j=1:x-1
        A=A+BW2(i,j);
    end
    if (A>1&&TOP==0)
        TOP=i;
    elseif (A==0&&TOP>1)
        LOW=i-1,break
    end
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการหาส่วนที่เป็นขอบเขตซ้ายและขอบเขตขวาของส่วนที่เป็นตัวเลขและตัวพยัญชนะของแผ่นป้ายทะเบียนรถยนต์

```
%----ใช้ค่าที่เก็บไว้ในตัวแปร TOP และ LOW ไปคัดลอกส่วนของภาพมาจากภาพที่เก็บไว้ใน
%----ตัวแปร BW2 แล้วเก็บภาพผลลัพทธ์ที่ได้จากการคัดลอกไว้ในตัวแปร another
another=BW2(TOP:LOW,1:y);
%----หาขนาดของภาพที่เก็บไว้ในตัวแปร another
[ll,kk]=size(another);
%----ทำการ Label ภาพที่เก็บไว้ในตัวแปร another เพื่อหาขอบซ้ายและขอบขวาของตัวเลขและ
%----พยัญชนะบนป้ายทะเบียนรถ แล้วเก็บภาพผลลัพธ์ที่ได้จากการ Label ไว้ในตัวแปร labeled3
[labeled3,numObjects3] = bwlabel(another,4);
figure(9),imshow(labeled3);
%----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
%----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)];
           elseif u3==5
               T1=[left(1,1) left(1,2) left(1,3) left(1,4)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
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
```

โปรแกรมการแบ่งขอบเขตของตัวเลขและตัวพยัญชนะออกมาทีละตัวโดยเรียงลำดับให้เหมือนแผ่นป้ายทะเบียนรถยนต์

```
%----หาค่าขอบซ้ายที่น้อยที่สุดที่เก็บไว้ในเมตริกซ์ T1 แล้วเก็บไว้ในตัวแปร cutpoint1
%----ซึ่งค่าขอบซ้ายที่น้อยที่สุดนี้ก็คือขอบซ้ายของพยัญชนะตัวแรกที่อยู่บนแผ่นป้ายทะเบียน
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
%----ซึ่งค่าขอบขวาที่น้อยที่สุดนี้ก็คือขอบขวาของพยัญชนะตัวแรกที่อยู่บนแผ่นป้ายทะเบียน
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
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
T1(:,point1)=[];
%----นำเมตริกซ์ที่ลบค่าที่น้อยที่สุดตัวแรกออกไปแล้ว ไปเก็บไว้ในตัวแปร T2
T2=T1;
%----หาค่าขอบซ้ายที่น้อยที่สุดในเมตริกซ์ T2 ซึ่งก็คือขอบซ้ายของพยัญชนะตัวที่ 2 ที่อยู่บน
%----แผ่นป้ายทะเบียน 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 เพื่อที่จะหาขอบ
%----ขวาของตัวเลขตัวที่ 2 ที่อยู่บนแผ่นป้ายทะเบียนแล้วนำเมตริกซ์ที่ได้ไปเก็บไว้ในตัวแปร U2
U1(:,point11)=[];
U2=U1;
%----ขอบขวาของตัวเลขหรือพยัญชนะตัวที่สองที่อยู่บนแผ่นป้ายทะเบียน
cutpoint4=min(U2);
[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)=[];
%----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;
    end
end
%----ลบค่าหลักที่เก็บไว้ในตัวแปร point 22 ออกไปจากเมตริกซ์ U2 เพื่อที่จะหาขอบขวา
%----ของตัวเลขตัวที่ 3 ที่อยู่บนแผ่นป้ายทะเบียน แล้วนำเมตริกซ์ที่ได้ไปเก็บไว้ในตัวแปร U3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
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
%----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
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
T4(:,point4)=[];
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
%----ตรวจสอบดูว่าตัวแปร cutpoint10 อยู่หลักใดของเมตริกซ์ U5
    if U5(1,num55)==cutpoint10
            point55=num55;
    end
end

%----ถ้ามีตัวเลขและพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ทั้งหมด 5 ตัวให้หยุดไว้ที่ขั้นตอนนี้
if u3>5
%----ลบค่าหลักที่เก็บไว้ในตัวแปร point 5 ออกไปจากเมตริกซ์ T5 เพื่อที่จะหาขอบซ้ายของตัวเลขตัวที่ 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
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
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
%---- 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
```

โปรแกรมการ Inverse Binary ของตัวเลขและตัวอักษรที่ถูกตัดออกทีละตัว

```
[m,n]=size(labeled3)
for j=1:n
    for i=1:m
%----ถ้าพิกเซลที่กำลังพิจารณาเป็นสีขาว ให้เปลี่ยนพิกเซลนั้นเป็นสีดำ
        if labeled3(i,j) == 0;
            labeled3(i,j) = 1;
%----ถ้าพิกเซลที่กำลังพิจารณาเป็นสีดำ ให้เปลี่ยนพิกเซลนั้นเป็นสีขาว
        else labeled3(i,j) == 1;
           labeled3(i,j) = 0;
        end
    end
end
```

โปรแกรมการคัดลอกส่วนที่เป็นตัวเลขและตัวพยัญชนะที่ถูกตัดแบ่งออกทีละตัว โดยเรียงลำดับให้เหมือนแผ่นป้ายทะเบียนรถยนต์

```
labeled3=another;
newone=(~another);
%----เมื่อทำ Inverse Binary ของภาพที่เก็บไว้ในตัวแปร another แล้วเก็บภาพผลลัพธ์ไว้ในตัวแปร
%----newone นำค่าที่น้อยที่สุดตัวแรกจากจากเมตริกซ์ที่เก็บค่า Column ที่เป็นขอบซ้ายและขอบขวาของ
%----ตัวเลขและพยัญชนะไว้ ซึ่งเก็บไว้ในตัวแปร cutpoint1 และ cutpoint2
%----มาเป็นขอบเขตในการแยกพยัญชนะตัวแรกที่อยู่บนแผ่นป้ายทะเบียนออกมา
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
labeled3=another;
newone=(~another);
%----เมื่อทำ Inverse Binary ของภาพที่เก็บไว้ในตัวแปร another แล้ว เก็บภาพผลลัพธ์ไว้ในตัวแปร newone
%---- นำค่าที่น้อยที่สุดตัวแรกจากจากเมตริกซ์ที่เก็บค่า Column ที่เป็นขอบซ้ายและขอบขวาของ
%----ตัวเลขและพยัญชนะไว้ ซึ่งเก็บไว้ในตัวแปร cutpoint1 และ cutpoint2
%----มาเป็นขอบเขตในการแยกพยัญชนะตัวแรกที่อยู่บนแผ่นป้ายทะเบียนออกมา
 letter1=newone(1:ll,cutpoint1:cutpoint2);
 %---- ll เป็นตัวแปรที่เก็บค่าความสูงของภาพที่เก็บไว้ในตัวแปร 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;
                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 ที่ใช่เป็นขอบเขตในการตัดขอบล่าง
 %----คือ Row สุดท้ายของภาพนั้น
   elseif sum(letter1(x1,1:y1)) ~= y1
                X2=x1;
          end
   end
 %----ใช้ Row ที่เป็นขอบบนและขอบล่างของตัวเลขหรือพยัญชนะนั้นคัดลอกส่วนของภาพมาจาก
 %-----ภาพที่เก็บ ไว้ในตัวแปร letter1 แล้วเก็บภาพผลลัพธ์ที่ได้จากการคัดลอกไว้ในตัวแปร letter1neww
 letter1neww=letter1(X1:X2,1:y1);
 %----แสดงภาพที่ได้ที่เก็บ ไว้ในตัวแปร 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);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----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
          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 พร้อมกับการตัดขอบบนและขอบล่าง
%----เรียบร้อยแล้ว 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
%---Column อ้างอิงในการคัดลอกภาพตัวเลขตัวที่ 3 ออกมาจากภาพแผ่นป้ายทะเบียน
letter3=newone(1:11,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;
end
    end
%----ได้ภาพตัวเลขตัวที่ 3 ที่อยู่บนแผ่นป้ายทะเบียนออกมา พร้อมกับการตัดขอบบนและ
%----ขอบล่างเรียบร้อยแล้ว เก็บภาพนี้ไว้ในตัวแปร letter3neww
letter3newww=letter3(X5:X6,1:y3);
%----แสดงภาพที่ได้ที่เก็บไว้ในตัวแปร letter3neww โดยมี Title ของภาพว่า letter3
figure(12),imshow(letter3neww);
title('letter3');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----ถ้ามีตัวเลขและพยัญชนะบนแผ่นป้ายทะเบียนรถยนต์ทั้งหมด 3 ตัวให้หยุดไว้เพียงแค่นี้
if u3>3
%----นำค่าขอบซ้ายและขอบขวาที่เก็บไว้ในตัวแปร 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 พร้อมกับการตัดขอบบนและขอบล่าง
%----เรียบร้อยแล้ว เก็บภาพนี้ไว้ในตัวแปร letter4neww
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
%---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
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
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
letter5newww=letter5(X9:X10,1:y5);
%----แสดงภาพที่ได้ที่เก็บไว้ในตัวแปร letter5neww โดยมี Title ของภาพว่า letter5
figure(14),imshow(letter5neww);
title('letter5');
end


%----ถ้ามีตัวเลขและพยัญชนะบนแผ่นป้ายทะเบียน 5 ตัวให้หยุดไว้ที่ขั้นตอนนี้
if u3>5
%----นำค่าขอบซ้ายและขอบขวาที่เก็บไว้ในตัวแปร cutpoint11 และ cutpoint12 ตามลำดับ
%---Column อ้างอิงในการคัดลอกภาพตัวเลขตัวที่ 6 ออกมาจากภาพแผ่นป้ายทะเบียน
letter6=newone(1:11,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
letter6newww=letter6(X11:X12,1:y6);
%----แสดงภาพที่ได้ที่เก็บไว้ในตัวแปร letter6neww โดยมี Title ของภาพว่า letter6
figure(15),imshow(letter6neww);
title('letter6');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการเรียกชุดแม่แบบขึ้นมาจากไฟล์ภาพโดยใช้นามสกุลภาพเป็น .JPG ทำการแปลงภาพสี RGB เป็นภาพ Gray Scale และแปลงเป็นภาพ Binary

```
%----เรียกข้อมูลชุดแม่แบบขึ้นมาจากไฟล์ภาพโดยใช้นามสกุลภาพเป็น .JPG เก็บไว้ในตัวแปร z
z=imread('C:\Users\SoDa\Desktop\newtem.jpg');
%----แปลงภาพสี RGB ให้เป็นภาพ Gray scale โดยใช้คำสั่ง rgb2gray
p=rgb2gray(z);
%----แปลงภาพ Gray Scale ที่เก็บไว้ในตัวแปร p ให้เป็นภาพ Binary โดยใช้ค่าเทรชโฮลที่ได้
%----จากตัวแปร p แล้วเก็บไว้ในตัวแปร w
w=im2bw(p);
```

โปรแกรมการหาขนาดภาพของชุดแม่แบบ และนำค่า Input ที่ได้จากการหาตัวเลขและตัวอักษรบนแผ่นป้ายทะเบียนรถยนต์ทุกตัวมาทำการหาค่าสัมประสิทธิ์ความเหมือนหรือค่า Correlation

```
%----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 มีจำนวน
%----หลักตามที่ตัวแปร 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 มีจำนวน
%----หลักตามที่ตัวแปร count4 นับได้ในตัวแปร YY
        YY(count4)=jj;
    end
end
[x3,y3]=size(XX);
[x4,y4]=size(YY);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for j3=1:y3
%----คัดลอกตัวเลขและพยัญชนะออกมาจากภาพ Template ที่เก็บไว้ในตัวแปร template ออกมา
%----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 ไปคัดลอกส่วนของภาพจาก
%----ภาพ AA จะได้ภาพตัวเลขและพยัญชนะที่ตัดขอบบนและขอบล่างแล้วเก็บภาพนี้ไว้ในตัวแปร 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
[x,y]=size(B);
%----ทำการปรับขนาดภาพในตัวแปร Input ให้มีความสูงเท่ากับค่าที่เก็บไว้ในตัวแปร x แล้วเก็บภาพผลลัพธ์
%----ที่ปรับความสูงแล้วไว้ในตัวแปร 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
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
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);
%---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 ให้ค่าสัมประสิทธิ์ความเหมือนเมื่อนำภาพนี้
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%----ไปวางทาบกับภาพตัวเลข 1 ซึ่งอยู่ตำแหน่งที่ 77 ของภาพชุด Template มีค่าเป็น 1
if n/m<=0.25
    CORR(1,77)=1;
%----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
%----ถ้าค่าสัมประสิทธิ์ความเหมือนที่เก็บไว้ในตัวแปร CORRELATION มีค่ามากกว่า 0.6 โปรแกรมจะ
%----ทำงานในส่วนนี้ คือจะแสดงผลการรู้จำภาพพยัญชนะตัวนี้ทางหน้าต่างCommand Window
if CORRELATION>=0.6
%----ถ้าค่าสัมประสิทธิ์ความเหมือนที่มีค่ามากที่สุดอยู่ที่ Column 1 หรือ 2 หรือ 3 ของเมตริกซ์
%----CORR แสดงว่าภาพพยัญชนะตัวนี้มีค่าความเหมือนมากที่สุดเมื่อนำไปวางทาบหาค่าสัมประสิทธิ์
%----ความเหมือนกันภาพพยัญชนะ ก ของภาพชุด Template ให้แสดงผลการรู้จำเป็นตัวพยัญชนะ
%---- ทางหน้าต่าง Command Window
if CORRELATION>=0.6
        if W1==1
            fprintf('ก')
            elseif W1==2
                fprintf('ข')
                elseif W1==3
                    fprintf('ค')
                    elseif W1==4
                        fprintf('ง')
                        elseif W1==5
                            fprintf('จ')
                            elseif W1==6
                                fprintf('ฉ')
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
elseif W1==7
printf('ช')

elseif W1==9
fprintf('ฌ')

elseif W1==10
fprintf('ฎ')

elseif W1==11
fprintf('ฐ')

elseif W1==12
fprintf('ณ')

elseif W1==13
fprintf('ต')

elseif W1==14
fprintf('ถ')

elseif W1==15
fprintf('ท')

elseif W1==16
fprintf('ธ')

elseif W1==17
fprintf('น')

elseif W1==18
fprintf('บ')

elseif W1==19
fprintf('ป')

elseif W1==20
fprintf('ผ')

elseif W1==21
fprintf('ฝ')

elseif W1==22
fprintf('ฝ')

fprintf('พ')
elseif W1==23
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
fprintf('พ')
elseif W1==24

fprintf('ภ')
elseif W1==25

fprintf('ม')
elseif W1==26

fprintf('ย')
elseif W1==27

fprintf('ร')
elseif W1==28

fprintf('ล')
elseif W1==29

fprintf('ว')
elseif W1==30

fprintf('ศ')
elseif W1==31

fprintf('ษ')
elseif W1==32

fprintf('ส')
elseif W1==33

fprintf('ห')
elseif W1==34

fprintf('ฬ')
elseif W1==35

fprintf('อ')
elseif W1==36

fprintf('0')
elseif W1==37
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                    fprintf('1')
                                    elseif W1==38

                                    fprintf('2')

                                    elseif W1==39
                                    fprintf('3')

                                    elseif W1==40
                                    fprintf('4')

                                    elseif W1==41
                                    fprintf('5')

                                    elseif W1==42
                                    fprintf('6')

                                    elseif W1==43
                                    fprintf('7')

                                    elseif W1==44
                                    fprintf('8')

                                    elseif W1==45
                                    fprintf('9')
        end
    end
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียกภาพตัวพยัญชนะขึ้นมาแล้วเปลี่ยนจากภาพ RGB เป็นภาพ binary จากนั้นทำการ inverse ภาพจากขาวเป็นดำจากดำเป็นขาว จะได้ภาพตัวพยัญชนะสีขาวและพื้นหลังสีดำ จากนั้นทำให้ขอบของภาพความหนาหนึ่งพิกเซลกลายเป็นสีดำทั้งหมดเพื่อไม่ให้ตัวพยัญชนะติดขอบภาพ

```
A = imread('C:\Documents and                                    %%% เรียกรูปภาพจากไฟล์
Settings\Administrator\Desktop\PRojEct\alphabet\204.bmp'); เก็บไว้ในตัวแปร A %%%
B = rgb2gray(A); %%% เปลี่ยนรูปภาพสี RGB เป็นสีเทาแล้วเก็บไว้ในตัวแปร B %%%
p = im2bw(B); %%% เปลี่ยนรูปภาพใน B เป็น binary แล้วเก็บไว้ใน p %%%
w = medfilt2(p,[3 3]); %%% กำจัดสิ่งรบกวนในรูปภาพ p แล้วเก็บไว้ใน w %%%
[m n]=size(w); %%% ขนาดรูปภาพ w โดย m = เก็บจำนวนแถว , n = เก็บจำนวนหลัก %%%
for i = 1:m
    for j = 1:n
        if w(i,j) == 1;
            w(i,j) = 0;
        else w(i,j) == 0;          } %%% inverse รูปภาพ w จากพิกเซลขาวเป็นดำ จากดำเป็นขาว %%%
            w(i,j) = 1;
        end
    end
end
[x,y] = size(w); %%% ขนาดรูปภาพ w โดย x = เก็บจำนวนแถว , y = เก็บจำนวนหลัก %%%
for i = 1:x
    for j = 1:y
        if w(1,j) == 1
            w(1,j) = 0;
        end
        if w(x,j) == 1
            w(x,j) = 0;
        end                        } %%% เปลี่ยนขอบภาพขนาด 1 พิกเซล เป็นสีดำทั้งหมด %%%
        if w(i,1) == 1
            w(i,1) = 0;
        end
        if w(i,y) == 1
            w(i,y) = 0;
        end
    end
end
```

จากนั้นแบ่งความสูงของตัวพยัญชนะเป็นสามส่วนแล้วทำค่าที่ได้จากการหารให้เป็นเลขจำนวนเต็ม แล้วให้เส้นแรกเป็นเส้นสมมติ ขั้นตอนนี้จะได้ดังรูปที่ 4.21

```
[m,n] = size(w);
a1=m/3; %%% นำความสูงของตัวพยัญชนะหรือจำนวนแถวหารสามแล้วเก็บไว้ใน a1 %%%
a2=int16(a1); %%% แปลงค่าใน a1 ให้เป็นเลขจำนวนเต็มแล้วเก็บไว้ใน a2 %%%
(a2 คือแถวที่เก็บเส้นสมมติหรือแถวที่หารสาม)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นหาแถวที่ตัวพยัญชนะมีความหนาของพิกเซลสีขาวมากที่สุด ขั้นตอนนี้จะได้ดังรูปที่ 4.22 (ข้อกำหนดของหน้าต่าง 3x3)

| | | |
|---|---|---|
| | i-1,j | |
| i,j+1 | i,j | i,j+1 |
| | i+1,j | |

⇨

| | | |
|---|---|---|
| 1,1 | 1,2 | 1,3 |
| 2,1 | 2,2 | 2,3 |
| 3,1 | 3,2 | 3,3 |

```
%%%%%%%%%%%%%%%%%%% หาแถมที่มีพิกเซลสีขาวมากที่สุด %%%%%%%%%%%%%%%%%%%%%%%%%
a=0;b=0;c=0;max=0;xmax=0;xmin=0;  %%% กำหนดตัวแปรเริ่มต้นเป็นศูนย์ %%%
for i = 2:m-1 %%% วนตามหลัง(บนลงล่าง)จากพิกเซลที่สองถึงแถวรองสุดท้าย %%%
    for j = 2:n-1 %%% วนตามแถว(ซ้ายไปขวา)จากพิกเซลที่สองถึงหลักรองสุดท้าย %%%
        if w(i,j) ~= 0 && w(i,j-1)==0 %%% ถ้าพิกเซล (2,2) ไม่เท่ากับ 0 และพิกเซล(2,1)เท่ากับ 0 %%%
            a = j; %%% จะวนที่ละแถวเก็บหลักที่เป็นพิกเซลสีขาวหลักแรกไว้ใน a %%%
        elseif w(i,j) ~= 0 && w(i,j+1)==0 %%% แต่ถ้าพิกเซล (2,2) ไม่เท่ากับ 0 และพิกเซล(2,3)
                                                เท่ากับ 0 %%%
            b = j; %%% เก็บหลักที่เป็นพิกเซลสีขาวหลักสุดท้ายในแถวนั้นๆ ไว้ใน b %%%
            c = b-a; %% นำหลักสุดท้ายลบหลักแรกจะได้ความหนาของพิกเซลสีขาวในแถวนั้นๆเก็บไว้ใน c %
            if c > max %%% ถ้าค่าที่เก็บไว้ใน c มีค่ามากกว่าค่าใน max %%%
                max = c; %%% หาแถวที่มีพิกเซลสีขาวอยู่ติดกันมากที่สุด %%%
                g = i; %%% เก็บแถวที่มีพิกเซลสีขาวมากที่สุดไว้ใน g %%%
            end
        end
    end
end
```

จากนั้นถ้าแถวที่มีความหนามากที่สุดมีค่ามากกว่าเส้นสมมติ จะเข้าเงื่อนไขหัวอยู่ส่วนบนของตัวพยัญชนะจึงเป็นโปรแกรมสำหรับหาหัวที่อยู่ส่วนบนของตัวพยัญชนะแล้วทำการใส่หัวลงไปให้กับตัวพยัญชนะนั้น ขั้นตอนนี้เป็นไปตามรูปที่ 4.23 ถึง 4.31

```
if g > a2 %%% ถ้าแถวที่มีพิกเซลสีขาวมากที่สุดมีค่ามากกว่าแถวของเส้นสมมติจะเข้าเงื่อนไขหาหัว %%%
%%% ที่มีหัวอยู่ด้านบน โดยใช้ขอบเขตช่วงบนในสามช่วงหาแถวและหลักที่มีพิกเซลสีขาวติดกันมากที่สุด %%%
%%% หาแถวที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงบน %%%
    a=0;b=0;c=0;maxx=0;xmax=0;xmin=0;  %%% กำหนดตัวแปรเริ่มต้นเป็นศูนย์ %%%
    for i = 2:a2 %%% ใช้หน้าต่าง 3x3 วนตั้งแต่แถวที่ 2 ถึงแถวที่ a2 (เป็นขอบเขตช่วงบน) %%%
        for j = 2:n-1 %%% ใช้หน้าต่าง 3x3 วนตั้งแต่หลักที่ 2 ถึงหลักรองสุดท้าย %%%
            if w(i,j) ~= 0 && w(i,j-1) == 0 %%% ถ้าพิกเซล (2,2) ไม่เท่ากับ 0 และพิกเซล
                                                (2,1)เท่ากับ 0 %%%
                a=j; %%% จะวนที่ละแถวเก็บหลักที่เป็นพิกเซลสีขาวหลักแรกไว้ใน a %%%
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        elseif w(i,j) ~= 0 && w(i,j+1) ==0 %%% แต่ถ้าพิกเซล (2,2) ไม่เท่ากับ 0 และพิกเซล
                                                    (2,3) เท่ากับ 0 %%%
            b=j; %%% เก็บหลักที่เป็นพิกเซลสีขาวหลักสุดท้ายในแถวนั้นๆ ไว้ใน b %%%
            c=b-a; %% นำหลักสุดท้ายลบหลักแรกจะได้ความหนาของพิกเซลสีขาวในแถวนั้นๆเก็บไว้
                    ใน c %%%
            if c>maxx %%% ถ้าค่าที่เก็บไว้ใน c มีค่ามากกว่าค่าใน maxx %%%
                maxx=c; %%% หาแถวที่มีพิกเซลสีขาวอยู่ติดกันมากที่สุด %%%
            end
        end
    end
end
```

```
for i = 2:a2 %%% ใช้หน้าต่าง 3x3 วนตั้งแต่แถวที่ 2 ถึงแถวที่ a2 (เป็นขอบเขตช่วงบน) %%%
    for j = 2:n-1 %%% ใช้หน้าต่าง 3x3 วนตั้งแต่หลักที่ 2 ถึงหลักรองสุดท้าย %%%
        if w(i,j) ~= 0 && w(i,j-1) == 0 %%% ถ้าพิกเซล (2,2) ไม่เท่ากับ 0 และพิกเซล
                                                (2,1)เท่ากับ 0 %%%
            a=j; %%% จะวนที่ละแถวเก็บหลักที่เป็นพิกเซลสีขาวหลักแรกไว้ใน a %%%
        elseif w(i,j) ~= 0 && w(i,j+1) ==0 %%% แต่ถ้าพิกเซล (2,2) ไม่เท่ากับ 0 และพิกเซล
                                                (2,3) เท่ากับ 0 %%%
            b=j; %%% เก็บหลักที่เป็นพิกเซลสีขาวหลักสุดท้ายในแถวนั้นๆ ไว้ใน b %%%
            g=i; %%% เก็บแถวที่เป็นพิกเซลสีขาวติดกันมากที่สุดนี้ไว้ใน g %%%
            c=b-a; %% นำหลักสุดท้ายลบหลักแรกจะได้ความหนาของพิกเซลสีขาวในแถวนั้นๆเก็บไว้
                    ใน c %%%
            if c==maxx %%% ถ้าค่าที่เก็บไว้ใน c มีค่าเท่ากับค่าใน maxx %%%
                xmin=a; %%% เก็บหลักแรกที่เป็นสีขาวในแถวที่มีพิกเซลสีขาวมากสุดไว้ใน
                        xmin %%%
                xmax=b;,break %%% เก็บหลักสุดท้ายที่เป็นสีขาวในแถวที่มีพิกเซลสีขาว
                                มากสุดไว้ใน xmax %%%
            end
        end
    end
end
.....
%%% หาหลักที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงบน %%%
d=0;e=0;f=0;maxy=0;ymin=0;ymax=0; %%% กำหนดตัวแปรเริ่มต้นเป็นศูนย์ %%%
for j = 2:n-1 %%% ใช้หน้าต่าง 3x3 วนตั้งแต่หลักที่ 2 ถึงหลักรองสุดท้าย %%%
    for i = 2:a2 %%% ใช้หน้าต่าง 3x3 วนตั้งแต่แถวที่ 2 ถึงแถวที่ a2 (เป็นขอบเขตช่วงบน) %%%
        if w(i,j) ~= 0 && w(i-1,j) == 0 %%% ถ้าพิกเซล (2,2) ไม่เท่ากับ 0 และพิกเซล
                                                (1,2)เท่ากับ 0 %%%
            d = i; %%% จะวนที่ละหลักเก็บแถวที่เป็นพิกเซลสีขาวหลักแรกไว้ใน d %%%
        elseif w(i,j) ~= 0 & w(i+1,j) == 0 %%% แต่ถ้าพิกเซล (2,2) ไม่เท่ากับ 0 และพิกเซล
                                                (3,2) เท่ากับ 0 %%%
            e = i; %%% เก็บแถวที่เป็นพิกเซลสีขาวแถวสุดท้ายในหลักนั้นๆไว้ใน e %%%
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                f = e-d;  %% นำแถวสุดท้ายลบแถวแรกจะได้ความหนาของพิกเซลสีขาวในหลักนั้นๆเก็บ
                          ไว้ใน f %%%
                if f>maxy  %%% ถ้าค่าที่เก็บไว้ใน f มีค่ามากกว่าค่าใน maxy %%%
                    maxy=f;  %%% หาหลักที่มีพิกเซลสีขาวอยู่ติดกันมากที่สุด %%%
                end
            end
        end
    end
    for j = 2:n-1  %%% ใช้หน้าต่าง 3x3 วนตั้งแต่หลักที่ 2 ถึงหลักรองสุดท้าย %%%
        for i = 2:a2  %%% ใช้หน้าต่าง 3x3 วนตั้งแต่แถวที่ 2 ถึงแถวที่ a2 (เป็นขอบเขตช่วงบน) %%%
            if w(i,j) ~= 0 && w(i-1,j) == 0  %%% ถ้าพิกเซล (2,2) ไม่เท่ากับ 0 และพิกเซล
                                               (1,2)เท่ากับ 0 %%%
                d = i;  %%% จะวนที่ละหลักเก็บแถวที่เป็นพิกเซลสีขาวหลักแรกไว้ใน d %%%
            elseif w(i,j) ~= 0 && w(i+1,j) == 0  %%% แต่ถ้าพิกเซล (2,2) ไม่เท่ากับ 0 และ
                                                   พิกเซล (3,2) เท่ากับ 0 %%%
                e = i;  %%% เก็บแถวที่เป็นพิกเซลสีขาวแถวสุดท้ายในหลักนั้นๆไว้ใน e %%%
                f = e-d;  %% นำแถวสุดท้ายลบแถวแรกจะได้ความหนาของพิกเซลสีขาวในหลักนั้นๆเก็บ
                          ไว้ใน f %%%
                if f==maxy  %%% ถ้าค่าที่เก็บไว้ใน f มีค่าเท่ากับค่าใน maxy %%%
                    ymin=d;  %%% เก็บแถวแรกที่เป็นสีขาวในหลักที่มีพิกเซลสีขาวมากสุดไว้ใน
                             ymin %%%
                    ymax=e;,break  %%% เก็บแถวแรกที่เป็นสีขาวในหลักที่มีพิกเซลสีขาวมากสุดไว้ใน
                                   ymin %%%
                end % เก็บพิกเซลสุดท้ายที่เป็นสีขาวในแถวที่มีพิกเซลสีขาวมากสุดไว้ใน ymax %
            end
        end
    end

%%%%%% ตีกรอบโดยใช้ความกว้างและสูงจากพิกเซลแรกและสุดท้ายที่เก็บไว้ตามแถวและหลัก %%%%%%
    for x1=xmin-1
        for y1=ymin-1:ymax+1
            hold on
            plot(x1,y1,'-r.')        %%% ตีกรอบด้านซ้าย ใช้หลักที่ xmin-1 และใช้แถวที่ ymin-1 ถึง
            hold off                       ymax+1 %%%
        end
    end
    for x1=xmin-1:xmax+1
        for y1=ymax+1
            hold on
            plot(x1,y1,'-r.')        %%% ตีกรอบด้านล่าง ใช้หลักที่ xmin-1 ถึง xmax+1 และใช้แถว
            hold off                       ที่ ymax+1 %%%
        end
    end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for x1=xmin-1:xmax+1
    for y1=ymin-1
        hold on
        plot(x1,y1,'-r.')        %%% ตีกรอบด้านบน ใช้หลักที่ xmin-1 ถึง xmax+1 และใช้แถวที่
        hold off                     ymin-1 %%%
    end
end
for x1=xmax+1
    for y1=ymin-1:ymax+1
        hold on
        plot(x1,y1,'-r.')        %%% ตีกรอบด้านขวา ใช้หลักที่ xmax+1 และใช้แถวที่ ymin-1
        hold off                     ถึง ymax+1 %%%
    end
end
%%%%%%%%%%%%%%%%%% ใส่สี่เหลี่ยมสีดำลงไปตรงกึ่งกลางของกรอบที่ตีไว้ %%%%%%%%%%%%%%%%%%
yy = maxy/3; %%% แบ่งความสูงของกรอบสี่เหลี่ยมเป็นสามส่วนเก็บไว้ใน yy %%%
yyy = int16(yy); %%% แปลงค่าใน yy ให้เป็นจำนวนเต็มแล้วเก็บไว้ใน yyy %%%
xx = maxx/3; %%% แบ่งความกว้างของกรอบสี่เหลี่ยมเป็นสามส่วนเก็บไว้ใน xx %%%
xxx = int16(xx); %%% แปลงค่าใน xx ให้เป็นจำนวนเต็มแล้วเก็บไว้ใน xxx %%%
xminx1xxx = xmin+xxx; %%% ขอบบนของสีเหลี่ยมสีดำ %%%
xminx2xxx = xmin+xxx*2; %%% ขอบล่างของสีเหลี่ยมสีดำ %%%
yminyyy = ymin+yyy; %%% ขอบซ้ายของสีเหลี่ยมสีดำ %%%
ymin2yyy = ymin+yyy*2; %%% ขอบขวาของสีเหลี่ยมสีดำ %%%
for i = yminyyy:ymin2yyy
    for j = xminx1xxx:xminx2xxx
        if w(i,j) == 1
            w(i,j) = 0;              %%% สีเหลี่ยมสีดำตามขอบเขตด้านบน %%%
        end
    end
end
```

แต่ถ้าแถวที่มีความหนามากที่สุดมีค่าน้อยกว่าเส้นสมมติจะเป็นตัวพยัญชนะที่มีหัวอยู่ในช่วงกลางและช่วงล่าง จากนั้นแบ่งความกว้างของตัวพยัญชนะออกเป็นสองส่วนแล้วหาจำนวนพิกเซลสีขาวในแถวกลางนี้ จะได้เงื่อนไขออกมาอีกสามเงื่อนไขคือ 1 กลุ่มพิกเซลสีขาว, 2 กลุ่มพิกเซลสีขาว และ 3 กลุ่มพิกเซลสีขาว ดังรูปที่ 4.32 และ 4.33

```
elseif g < a2 %%% แต่ถ้าแถวที่มีพิกเซลสีขาวมากที่สุดมีค่าน้อยกว่าแถวของเส้นสมมติจะเข้าเงื่อนไขนี้แทน %%%
 [m,n] = size(w); %%% ขนาดรูปภาพ w โดย m = เก็บจำนวนแถว , n = เก็บจำนวนหลัก %%%
    b1=n/2; %%% นำความกว้างหรือจำนวนหลักมาแบ่งเป็นสองส่วนเก็บไว้ใน b1 %%%
    b2=int16(b1); %%% แปลงค่าใน b1 ให้เป็นเลขจำนวนเต็มแล้วเก็บไว้ใน b2 %%%
    r=0;s=0;t=0; %%% กำหนดตัวแปรเริ่มต้นเป็นศูนย์ %%%
    for j = b2 %%% วนเฉพาะหลัก b2 %%%
        for i = 2:m-1 %%% วนหลัก b2 ตั้งแต่แถวที่ 2 ถึง แถวรองสุดท้าย %%%
            if w(i-1,j) ~= 1 & w(i,j) == 1 %%% ถ้าหน้าต่าง 3x3 พิกัด(1,2) ไม่เท่ากับ 1 และพิกัด
                                               (2,2) เท่ากับ 1 จะทำเงื่อนไขด้านล่างต่อ %%%
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            if (r == 0)& (s == 0)& (t == 0) %%% ถ้า r=0 และ s=0 และ t=0 จะทำให้ r=1 %%
                r = 1;
            elseif (r ~= 0)& (t == 0)& (s == 0) %%% แต่ถ้า r ไม่เท่ากับ 0 และ s=0 และ t=0
                                                    จะทำให้ s=1 %%%
                s = 1;
            else %%% แต่ถ้าเป็นเงื่อนไขอื่นๆ จะทำให้ t = 1 %%%
                t = 1;
            end
        else %%% ถ้าไม่เข้าเงื่อนไข if ด้านบน จะมาเข้าเงื่อนไขด้านล่างนี้แทน %%%
            if w(i,j) == 1 %%% ถ้าหน้าต่าง 3x3 พิกัด(2,2)เท่ากับ 1 จะเข้าเงื่อนไขด้านล่าง %%%
                if t > 0 %%% ถ้า t มากกว่า 0 จะทำให้ t มีค่าเพิ่มขึ้น 1 %%%
                    t = t+1;
                elseif s > 0; %%% แต่ถ้า s มากกว่า 0 จะทำให้ s มีค่าเพิ่มขึ้น 1 %%%
                    s = s+1;
                else r > 0; %%% แต่ถ้า r มากกว่า 0 จะทำให้ r มีค่าเพิ่มขึ้น 1 %%%
                    r = r+1;
                end
            end
        end
    end
end
```

เงื่อนไขที่ 1 คือมีกลุ่มพิกเซลสีขาวหนึ่งกลุ่ม จะเป็นตัวพยัญชนะที่มีหัวอยู่ช่วงล่างแล้วจึงเข้าโปรแกรมหาหัวที่อยู่ช่วงล่างของตัวพยัญชนะจากนั้นจึงใส่หัวลงไปให้กับตัวพยัญชนะนั้น ขั้นตอนนี้เป็นไปตามรูปที่ 4.34 ถึง 4.36

```
    if (r ~= 0) && (s == 0) && (t == 0)%%% ถ้ามี 1 กลุ่มพิกเซลสีขาว จะเข้าเงื่อนไขด้านล่างนี้
เป็นเงื่อนไขหาหัวที่มีหัวอยู่ด้านล่าง โดยใช้ขอบเขตช่วงล่างในสามช่วงหาแถวและหลักที่มีพิกเซลสีขาวติดกันมากที่สุด %%%
%%% หาแถวที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงล่าง %%%
    a=0;b=0;c=0;maxx=0;xmax=0;xmin=0;a1=0;B=0;v=0;
    for i = a2*2:m-1
        for j = 2:n-1
            if w(i,j) ~= 0 && w(i,j-1) == 0
                a=j;
            elseif w(i,j) ~= 0 && w(i,j+1) ==0
                b=j;
                c=b-a;
                if c>maxx
                    maxx=c;
                end
            end
        end
    end
    for i = a2*2:m-1
        for j = 2:n-1
            if w(i,j) ~= 0 && w(i,j-1) == 0
                a=j;
            elseif w(i,j) ~= 0 && w(i,j+1) ==0
                b=j;
                c=b-a;
                if c==maxx
                    xmin=a;
                    xmax=b;,break
                end
            end
        end
    end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%%% หาหลักที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงล่าง %%%
        d=0;e=0;f=0;maxy=0;ymin=0;ymax=0;
        for j = 2:n-1
            d=0;
            for i = a2*2:m-1
                if w(i,j) ~= 0 && w(i-1,j) == 0
                    d = i;
                elseif w(i,j) ~= 0 & w(i+1,j) == 0
                    e = i;
                    if d == 0
                        f = 0;
                    elseif d ~= 0
                        f = e-d;
                    end
                    if f>maxy
                       maxy=f;
                    end
                end
            end
        end
        for j = 2:n-1
            d=0;
            for i = a2*2:m-1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%
                if w(i,j) ~= 0 && w(i-1,j) == 0
                    d = i;
                elseif w(i,j) ~= 0 & w(i+1,j) == 0
                    e = i;
                    if d == 0
                        f = 0;
                    elseif d ~= 0
                        f = e-d;
                    end
                    if f==maxy
                        ymin=d;
                        ymax=e;,break
                    end
                end
            end
        end
%%%%%%% ตีกรอบโดยใช้ความกว้างและสูงจากพิกเซลแรกและสุดท้ายที่เก็บไว้ตามแถวและหลัก %%%%%%
        for x1=xmin-1
            for y1=ymin-1:ymax+1
                hold on
                plot(x1,y1,'-r.')
                hold off
            end
        end
        for x1=xmin-1:xmax+1
            for y1=ymax+1
                hold on
                plot(x1,y1,'-r.')
                hold off
            end
        end
        for x1=xmin-1:xmax+1
            for y1=ymin-1
                hold on
                plot(x1,y1,'-r.')
                hold off
            end
        end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        for x1=xmax+1
            for y1=ymin-1:ymax+1
                hold on
                plot(x1,y1,'-r.')
                hold off
            end
        end
%%%%%%%%%%%%%%%%% ใส่สี่เหลี่ยมสีดำลงไปตรงกึ่งกลางของกรอบที่ตีไว้ %%%%%%%%%%%%%%%%%
        ymax
        yy = maxy/3;
        yyy = int16(yy);
        xx = maxx/3;
        xxx = int16(xx);
        xminx1xxx = xmin+xxx;
        xminx2xxx = xmin+xxx*2;
        yminyyy = ymin+yyy
        ymin2yyy = ymin+yyy*2
        for i = yminyyy:ymin2yyy
            for j = xminx1xxx:xminx2xxx
                if w(i,j) == 1
                    w(i,j) = 0;
                end
            end
        end
```

แต่ถ้าเข้าเงื่อนไขที่ 3 คือมีกลุ่มพิกเซลสีขาวสามกลุ่ม จะเป็นตัวพยัญชนะที่มีหัวอยู่ช่วงกลางและช่วงล่าง เพราะฉะนั้นจึงต้องแยกพยัญชนะที่มีหัวอยู่ช่วงกลางออกจากช่วงล่างก่อนโดยแบ่งความสูงของตัวพยัญชนะออกเป็นสองส่วน จากนั้นหาจำนวนกลุ่มพิกเซลสีขาวบนเส้นแบ่งนั้นจะเกิดเงื่อนไขสองเงื่อนไขคือ 1 มีหนึ่งกลุ่มพิกเซลสีขาวจะมีหัวอยู่ช่วงล่างและ 2 มีสองกลุ่มพิกเซลสีขาวจะมีหัวอยู่ช่วงกลาง ขั้นตอนนี้เป็นไปตามรูปที่ 4.38 และ 4.39

```
elseif (r ~= 0) && (s ~= 0) && (t ~= 0) %%% แต่ถ้ามี 3 กลุ่มพิกเซลสีขาว จะเข้าเงื่อนไขด้านล่างนี้
        [m,n] = size(w);
        c1 = m/2; %%% แบ่งความสูงหรือจำนวนแถวออกเป็นสองส่วน จะได้แถวที่แบ่งครึ่งภาพเก็บไว้ใน c1 %%
        c2 = int16(c1) %%% แปลงค่าใน c1 ให้เป็นจำนวนเต็ม เก็บไว้ใน c2 %%%
        for i = c2
            r1=0;s1=0;t1=0;
            for j = 2:n-1
                if w(i,j-1) ~= 1 && w(i,j) == 1
                    if (r1== 0) && (s1 == 0) && (t1 == 0)
                        r1 = 1;
                    elseif (r1 ~= 0) && (s1 == 0) && (t1 == 0)
                        s1 = 1;
                    else
                        t1 = 1;
                    end
                else
                    if w(i,j) == 1
                        if t1 > 0
                            t1 = t1+1;
                        elseif s1 > 0
                            s1 = s1+1;
                        elseif r1 > 0
                            r1 = r1+1;
                        end
                    end
                end
            end
```

%%% หาจำนวนกลุ่มพิกเซลสีขาวบนแถวกลางนี้ แล้วเก็บแต่ละกลุ่มไว้ใน r1 s1 t1 %%%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (r1 ~= 0) && (s1 == 0) && (t1 == 0) )%%% ถ้ามี 1 กลุ่มพิกเซลสีขาว จะเข้าเงื่อนไข
```
หาหัวที่มีหัวอยู่ด้านล่าง โดยใช้ขอบเขตช่วงล่างในสามช่วงหาแถวและหลักที่มีพิกเซลสีขาวติดกันมากที่สุด %%%

%%% หาแถวที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงล่าง %%%

```
a=0;b=0;c=0;maxx=0;xmax=0;xmin=0;a1=0;B=0;v=0;
for i = a2*2:m-1
    for j = 2:n-1
        if w(i,j) ~= 0 && w(i,j-1) == 0
            a=j;
        elseif w(i,j) ~= 0 && w(i,j+1) ==0
            b=j;
            c=b-a;
            if c>maxx
                maxx=c;
            end
        end
    end
end
for i = a2*2:m-1
    for j = 2:n-1
        if w(i,j) ~= 0 && w(i,j-1) == 0
            a=j;
        elseif w(i,j) ~= 0 && w(i,j+1) ==0
            b=j;
            c=b-a;
            if c==maxx
                xmin=a;
                xmax=b;,break
            end
        end
    end
end
.....
```

%%% หาหลักที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงล่าง %%%

```
d=0;e=0;f=0;maxy=0;ymin=0;ymax=0;
for j = 2:n-1
    d=0;
    for i = a2*2:m-1
        if w(i,j) ~= 0 && w(i-1,j) == 0
            d = i;
        elseif w(i,j) ~= 0 & w(i+1,j) == 0
            e = i;
            if d == 0
                f = 0;
            elseif d ~= 0
                f = e-d;
            end
            if f>maxy
                maxy=f;
            end
        end
    end
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            for j = 2:n-1
                d=0;
                for i = a2*2:m-1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                    if w(i,j) ~= 0 && w(i-1,j) == 0
                        d = i;
                    elseif w(i,j) ~= 0 & w(i+1,j) == 0
                        e = i;
                        if d == 0
                            f = 0;
                        elseif d ~= 0
                            f = e-d;
                        end
                        if f==maxy
                            ymin=d;
                            ymax=e;,break
                        end
                    end
                end
            end
```

%%%%%% ตีกรอบโดยใช้ความกว้างและสูงจากพิกเซลแรกและสุดท้ายที่เก็บไว้ตามแถวและหลัก %%%%%%

```
            for x1=xmin-1
                for y1=ymin-1:ymax+1
                    hold on
                    plot(x1,y1,'-r.')
                    hold off
                end
            end
            for x1=xmin-1:xmax+1
                for y1=ymax+1
                    hold on
                    plot(x1,y1,'-r.')
                    hold off
                end
            end
            for x1=xmin-1:xmax+1
                for y1=ymin-1
                    hold on
                    plot(x1,y1,'-r.')
                    hold off
                end
            end
            for x1=xmax+1
                for y1=ymin-1:ymax+1
                    hold on
                    plot(x1,y1,'-r.')
                    hold off
                end
            end
```

%%%%%%%%%%%%%%%%%% ใส่สี่เหลี่ยมสีดำลงไปตรงกึ่งกลางของกรอบที่ตีไว้ %%%%%%%%%%%%%%%%%%

```
            ymin;
            yy = maxy/3;
            yyy = int16(yy);
            xx = maxx/3;
            xxx = int16(xx);
            xminx1xxx = xmin+xxx;
            xminx2xxx = xmin+xxx*2;
            yminyyy = ymin+yyy
            ymin2yyy = ymin+yyy*2
            for i = yminyyy:ymin2yyy
                for j = xminx1xxx:xminx2xxx
                    if w(i,j) == 1
                        w(i,j) = 0;
                    end
                end
            end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Else )%%% แต่ถ้าไม่ใช่ทั้ง 1 และ 3 กลุ่มพิกเซลสีขาว จะเข้าเงื่อนไขหาหัวที่มีหัวอยู่ช่วงกลาง โดยใช้ขอบเขตช่วงกลางในสามช่วงหาแถวและหลักที่มีพิกเซลสีขาวติดกันมากที่สุด %%%

%%% หาแถวที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงกลาง %%%

```
a=0;b=0;c=0;maxx=0;xmax=0;xmin=0;
for i = a2:a2*2
    for j = 2:n-1
        if w(i,j) ~= 0 && w(i,j-1)==0
            a=j;
        elseif w(i,j) ~= 0 && w(i,j+1)==0
            b=j;
            c=b-a;
            if c>maxx
                maxx=c;
            end
        end
    end
end
for i = a2:a2*2
    for j = 2:n-1
        if w(i,j) ~= 0 && w(i,j-1)==0
            a=j;
        elseif w(i,j) ~= 0 && w(i,j+1)==0
            b=j;
            c=b-a;
            if c==maxx
                xmin=a;
                xmax=b;o = i;,break
            end
        end
    end
end
```

%%% หาหลักที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงกลาง %%%

```
d=0;e=0;f=0;maxy=0;ymin=0;ymax=0;
for j = 2:n-1
    d=0;
    for i = a2:a2*2
        if w(i,j) ~= 0 && w(i-1,j) == 0
            d = i;
        elseif w(i,j) ~= 0 & w(i+1,j) == 0
            e = i;
            if d == 0
                f = 0;
            elseif d ~= 0
                f = e-d;
            end
            if f>maxy
                maxy=f;
            end
        end
    end
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        for j = 2:n-1
            d=0;
            for i = a2:a2*2
                if w(i,j) ~= 0 && w(i-1,j) == 0
                    d = i;
                elseif w(i,j) ~= 0 & w(i+1,j) == 0
                    e = i;
                    if d == 0
                        f = 0;
                    elseif d ~= 0
                        f = e-d;
                    end
                    if f==maxy
                        ymin=d;
                        ymax=e;,break
                    end
                end
            end
        end
```

%%%%%% ตีกรอบโดยใช้ความกว้างและสูงจากพิกเซลแรกและสุดท้ายที่เก็บไว้ตามแถวและหลัก %%%%%%

```
        for x1=xmin-1
            for y1=ymin-1:ymax+1
                hold on
                plot(x1,y1,'-r.')
                hold off
            end
        end
        for x1=xmin-1:xmax+1
            for y1=ymax+1
                hold on
                plot(x1,y1,'-r.')
                hold off
            end
        end
        for x1=xmin-1:xmax+1
            for y1=ymin-1
                hold on
                plot(x1,y1,'-r.')
                hold off
            end
        end
        for x1=xmax+1
            for y1=ymin-1:ymax+1
                hold on
                plot(x1,y1,'-r.')
                hold off
            end
        end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%%%%%%%%%%%%%%%%%% ใส่สี่เหลี่ยมสีดำลงไปตรงกึ่งกลางของกรอบที่ตีไว้ %%%%%%%%%%%%%%%%%%
                ymax
                yy = maxy/3;
                yyy = int16(yy);
                xx = maxx/3;
                xxx = int16(xx);
                xminx1xxx = xmin+xxx;
                xminx2xxx = xmin+xxx*2;
                yminyyy = ymin+yyy
                ymin2yyy = ymin+yyy*2
                for i = yminyyy:ymin2yyy
                    for j = xminx1xxx:xminx2xxx
                        if w(i,j) == 1
                            w(i,j) = 0;
                        end
                    end
                end
        end
```

แต่ถ้าเข้าเงื่อนไขที่ 2 คือมีกลุ่มพิกเซลสีขาวสองกลุ่มและกลุ่มหน้ามีค่ามากกว่ากลุ่มหลังจะเป็นตัวพยัญชนะที่มีหัวอยู่ช่วงล่าง ฉะนั้นจึงเข้าโปรแกรมหาหัวที่อยู่ช่วงล่างแล้วใส่หัวลงไปให้กับตัวพยัญชนะนั้น ดังรูปที่ 4.41

```
elseif r > s %%% แต่ถ้ากลุ่มพิกเซลสีขาว r มากกว่า s จะเข้าเงื่อนไขด้านล่าง ซึ่งเป็นเงื่อนไขหาหัวที่มีหัวอยู่ด้านล่าง
โดยใช้ขอบเขตช่วงล่างในสามช่วงหาแถวและหลักที่มีพิกเซลสีขาวติดกันมากที่สุด %%%
%%% หาแถวที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงล่าง %%%
            a=0;b=0;c=0;maxx=0;xmax=0;xmin=0;a1=0;B=0;v=0;
            for i = a2*2:m-1
                for j = 2:n-1
                    if w(i,j) ~= 0 && w(i,j-1)==0
                        a=j;
                    elseif w(i,j) ~= 0 && w(i,j+1)==0
                        b=j;
                        c=b-a;
                        if c>maxx
                            maxx=c;
                        end
                    end
                end
            end
            for i = a2*2:m-1
                for j = 2:n-1
                    if w(i,j) ~= 0 && w(i,j-1)==0
                        a=j;
                    elseif w(i,j) ~= 0 && w(i,j+1)==0
                        b=j;
                        c=b-a;
                        if c==maxx
                            xmin=a;
                            xmax=b;,break
                        end
                    end
                end
            end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%%% หาหลักที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงล่าง %%%
            d=0;e=0;f=0;maxy=0;ymin=0;ymax=0;
            for j = 2:n-1
                d=0;
                for i = a2*2:m-1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                    if w(i,j) ~= 0 && w(i-1,j) == 0
                        d = i;
                    elseif w(i,j) ~= 0 & w(i+1,j) == 0
                        e = i;
                        if d == 0
                            f = 0;
                        elseif d ~= 0
                            f = e-d;
                        end
                        if f>maxy
                            maxy=f;
                        end
                    end
                end
            end
            for j = 2:n-1
                d=0;
                for i = a2*2:m-1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                    if w(i,j) ~= 0 && w(i-1,j) == 0
                        d = i;
                    elseif w(i,j) ~= 0 & w(i+1,j) == 0
                        e = i;
                        if d == 0
                            f = 0;
                        elseif d ~= 0
                            f = e-d;
                        end
                        if f==maxy
                            ymin=d;
                            ymax=e;,break
                        end
                    end
                end
            end
%%%%%%% ตีกรอบโดยใช้ความกว้างและสูงจากพิกเซลแรกและสุดท้ายที่เก็บไว้ตามแถวและหลัก %%%%%%%
            for x1=xmin-1
                for y1=ymin-1:ymax+1
                    hold on
                    plot(x1,y1,'-r.')
                    hold off
                end
            end
            for x1=xmin-1:xmax+1
                for y1=ymax+1
                    hold on
                    plot(x1,y1,'-r.')
                    hold off
                end
            end
            for x1=xmin-1:xmax+1
                for y1=ymin-1
                    hold on
                    plot(x1,y1,'-r.')
                    hold off
                end
            end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            for x1=xmax+1
                for y1=ymin-1:ymax+1
                    hold on
                    plot(x1,y1,'-r.')
                    hold off
                end
            end
%%%%%%%%%%%%%%%%% ใส่สี่เหลี่ยมสีดำลงไปตรงกึ่งกลางของกรอบที่ตีไว้ %%%%%%%%%%%%%%%%%%
            ymin;
            yy = maxy/3;
            yyy = int16(yy);
            xx = maxx/3;
            xxx = int16(xx);
            xminx1xxx = xmin+xxx;
            xminx2xxx = xmin+xxx*2;
            yminyyy = ymin+yyy
            ymin2yyy = ymin+yyy*2
            for i = yminyyy:ymin2yyy
                for j = xminx1xxx:xminx2xxx
                    if w(i,j) == 1
                        w(i,j) = 0;
                    end
                end
            end
```

แต่ถ้ากลุ่มหน้ามีค่าน้อยกว่ากลุ่มหลังจะเป็นตัวพยัญชนะที่มีหัวอยู่ช่วงล่างและช่วงกลาง ฉะนั้นจึงต้องแยกก่อนว่าหัวอยู่ช่วงกลางหรือช่วงล่างโดยการแบ่งความสูงออกเป็นสองส่วนเท่าๆกันแล้วหาจำนวนกลุ่มพิกเซลสีขาวบนเส้นแบ่งนั้น และได้เงื่อนไขออกมาสามเงื่อนไข ขั้นตอนนี้แสดงดังรูปที่ 4.43

```
elseif s > r %%% แต่ถ้ากลุ่มพิกเซลสีขาว s มากกว่า r จะต้องหากลุ่มพิกเซลสีขาวบนแถวที่แบ่งความสูงออกเป็น
สองส่วน อีกครั้ง แล้วเก็บไว้ในตัวแปร r2 , s2 และ t2 %%%
        [m,n] = size(w);
        d1 = m/2;
        d2 = int16(d1);
        for i = d2
            r2=0;s2=0;t2=0;
            for j = 2:n-1
                if w(i,j-1) ~= 1 && w(i,j) == 1
                    if (r2 == 0) && (s2 == 0) && (t2 == 0)
                        r2 = 1;
                    elseif (r2 ~= 0) && (s2 == 0) && (t2 == 0)
                        s2 = 1;
                    else
                        t2 = 1;
                    end
                else
                    if w(i,j) == 1
                        if t2 > 0
                            t2 = t2+1;
                        elseif s2 > 0
                            s2 = s2+1;
                        elseif r2 > 0
                            r2 = r2+1;
                        end
                    end
                end
            end
        end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยถ้าเข้าเงื่อนไขที่ 2 คือ มีกลุ่มพิกเซลสีขาวสองกลุ่มและกลุ่มหลังมากกว่าหรือเท่ากับกลุ่มหน้าจะเป็นตัวพยัญชนะที่มีหัวอยู่ช่วงล่าง จึงเข้าโปรแกรมหาหัวที่อยู่ช่วงล่างแล้วใส่หัวให้กับตัวพยัญชนะนั้น ดังรูปที่ 4.48

```
        if s2 >= r2  %%% ถ้า s2 มากกว่า r2 จะเงื่อนไขหาหัวที่มีหัวอยู่ด้านล่าง โดยใช้ขอบเขตช่วงล่างในสาม
ช่วงหาแถวและหลักที่มีพิกเซลสีขาวติดกันมากที่สุด %%%
%%% หาแถวที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงล่าง %%%
            a=0;b=0;c=0;maxx=0;xmax=0;xmin=0;a1=0;B=0;v=0;
            for i = a2*2:m-1
                for j = 2:n-1
                    if w(i,j) ~= 0 && w(i,j-1) == 0
                        a=j;
                    elseif w(i,j) ~= 0 && w(i,j+1) ==0
                        b=j;
                        c=b-a;
                        if c>maxx
                            maxx=c;
                        end
                    end
                end
            end
            for i = a2*2:m-1
                for j = 2:n-1
                    if w(i,j) ~= 0 && w(i,j-1) == 0
                        a=j;
                    elseif w(i,j) ~= 0 && w(i,j+1) ==0
                        b=j;
                        c=b-a;
                        if c==maxx
                            xmin=a;
                            xmax=b;,break
                        end
                    end
                end
            end
            ..............................................
 %%% หาหลักที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงล่าง %%%
            d=0;e=0;f=0;maxy=0;ymin=0;ymax=0;
            for j = 2:n-1
                d=0;
                for i = a2*2:m-1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                    if w(i,j) ~= 0 && w(i-1,j) == 0
                        d = i;
                    elseif w(i,j) ~= 0 & w(i+1,j) == 0
                        e = i;
                        if d == 0
                            f = 0;
                        elseif d ~= 0
                            f = e-d;
                        end
                        if f>maxy
                            maxy=f;
                        end
                    end
                end
            end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        for j = 2:n-1
            d=0;
            for i = a2*2:m-1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                if w(i,j) ~= 0 && w(i-1,j) == 0
                    d = i;
                elseif w(i,j) ~= 0 & w(i+1,j) == 0
                    e = i;
                    if d == 0
                        f = 0;
                    elseif d ~= 0
                        f = e-d;
                    end
                    if f==maxy
                        ymin=d;
                        ymax=e;,break
                    end
                end
            end
        end
```

%%%%%% ตีกรอบโดยใช้ความกว้างและสูงจากพิกเซลแรกและสุดท้ายที่เก็บไว้ตามแถวและหลัก %%%%%%

```
        figure(12),imshow(w)
        for x1=xmin-1
            for y1=ymin-1:ymax+1
                hold on
                plot(x1,y1,'-r.')
                hold off
            end
        end
        for x1=xmin-1:xmax+1
            for y1=ymax+1
                hold on
                plot(x1,y1,'-r.')
                hold off
            end
        end
        for x1=xmin-1:xmax+1
            for y1=ymin-1
                hold on
                plot(x1,y1,'-r.')
                hold off
            end
        end
        for x1=xmax+1
            for y1=ymin-1:ymax+1
                hold on
                plot(x1,y1,'-r.')
                hold off
            end
        end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%%%%%%%%%%%%%%%%%% ใส่สี่เหลี่ยมสีดำลงไปตรงกึ่งกลางของกรอบที่ตีไว้ %%%%%%%%%%%%%%%%%%
            ymax;
            ymin;
            yy = maxy/3;
            yyy = int16(yy);
            yyy;
            xx = maxx/3;
            xxx = int16(xx);
            xminx1xxx = xmin+xxx;
            xminx2xxx = xmin+xxx*2;
            yminyyy = ymin+yyy
            ymin2yyy = ymin+yyy*2
            for i = yminyyy:ymin2yyy
                for j = xminx1xxx:xminx2xxx
                    if w(i,j) == 1
                        w(i,j) = 0;
                    end
                end
            end
```

แต่ถ้าเข้าเงื่อนไขสองกลุ่มพิกเซลสีขาวที่กลุ่มหน้ามากกว่ากลุ่มหลังจะเป็นตัวพยัญชนะที่มีหัวอยู่ช่วงกลางตรงกึ่งกลางจึงต้องใช้ขอบเขตใหม่ในการหาส่วนหัวและใส่หัวลงไป ดังรูปที่ 4.46

```
    elseif r2 > s2 && (t2 ~=0) %%% แต่ถ้า r2 มากกว่า s2 และ t2เป็นศูนย์ จะเข้าเงื่อนไขหาหัวที่อยู่กึ่งกลาง จึงต้องใช้ขอบเขตที่ต่างออกไปจากเงื่อนไขก่อนๆ ดังนี้ %%%
                [m,n] = size(w);
                a1=m/5; %%% แบ่งความสูงเป็นห้าช่วง %%%
                a2=int16(a1); %%% แปลงค่าที่ได้เป็นจำนวนเต็ม %%%
                b1=n/5; %%% แบ่งความกว้างเป็นห้าช่วง %%%
                b2=int16(b1); %%% แปลงค่าที่ได้เป็นจำนวนเต็ม %%%
%%% หาแถวที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงกึ่งกลาง %%%
                a=0;b=0;c=0;maxx=0;xmax=0;xmin=0;a1=0;B=0;v=0;
                for i = a2:a2*3 %%% ขอบเขตความสูงในช่วงที่สองถึงสามจากในห้าช่วง %%%
                    for j = b2-5:b2*4 %%% ขอบเขตความกว้างในช่วงที่สองถึงสี่จากในห้าช่วง %%

                        if w(i,j) ~= 0 && w(i,j-1)==0
                            a=j;
                        elseif w(i,j) ~= 0 && w(i,j+1)==0
                            b=j;
                            c=b-a;
                            if c>maxx
                                maxx=c;
                            end
                        end
                    end
                end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                for i = a2:a2*3
                    for j = b2-5:b2*4
                        if w(i,j) ~= 0 && w(i,j-1) == 0
                            a=j;
                        elseif w(i,j) ~= 0 && w(i,j+1) ==0
                            b=j;
                            c=b-a;
                            if c==maxx
                                xmin=a;
                                xmax=b;,break
                            end
                        end
                    end
                end
```

%%% หาหลักที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงกึ่งกลาง %%%

```
d=0;e=0;f=0;maxy=0;ymin=0;ymax=0;
            for j = b2-5:b2*4  %% ขอบเขตความกว้างในช่วงที่สองถึงสี่จากในห้าช่วง %%
                d=0;
                for i = a2:a2*3  %%% ขอบเขตความสูงในช่วงที่สองถึงสามจากในห้าช่วง %%%
                    if w(i,j) ~= 0 && w(i-1,j) == 0
                        d = i;
                    elseif w(i,j) ~= 0 & w(i+1,j) == 0
                        e = i;
                        if d == 0
                            f = 0;
                        elseif d ~= 0
                            f = e-d;
                        end
                        if f>maxy
                            maxy=f;
                        end
                    end
                end
            end
            for j = b2-5:b2*4
                d=0;
                for i = a2:a2*3
                    if w(i,j) ~= 0 && w(i-1,j) == 0
                        d = i;
                    elseif w(i,j) ~= 0 & w(i+1,j) == 0
                        e = i;
                        if d == 0
                            f = 0;
                        elseif d ~= 0
                            f = e-d;
                        end
                        if f==maxy
                            ymin=d;
                            ymax=e;,break
                        end
                    end
                end
            end
```

%%%%%% ตีกรอบโดยใช้ความกว้างและสูงจากพิกเซลแรกและสุดท้ายที่เก็บไว้ตามแถวและหลัก %%%%%%

```
            for x1=xmin-1
                for y1=ymin-1:ymax+1
                    hold on
                    plot(x1,y1,'-r.')
                    hold off
                end
            end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                for x1=xmin-1:xmax+1
                    for y1=ymax+1
                        hold on
                        plot(x1,y1,'-r.')
                        hold off
                    end
                end
                for x1=xmin-1:xmax+1
                    for y1=ymin-1
                        hold on
                        plot(x1,y1,'-r.')
                        hold off
                    end
                end
                for x1=xmax+1
                    for y1=ymin-1:ymax+1
                        hold on
                        plot(x1,y1,'-r.')
                        hold off
                    end
                end
%%%%%%%%%%%%%%%%%% ใส่สี่เหลี่ยมสีดำลงไปตรงกึ่งกลางของกรอบที่ตีไว้ %%%%%%%%%%%%%%%%%%
                yy = maxy/3;
                yyy = int16(yy);
                xx = maxx/3;
                xxx = int16(xx);
                xminx1xxx = xmin+xxx;
                xminx2xxx = xmin+xxx*2;
                yminyyy = ymin+yyy;
                ymin2yyy = ymin+yyy*2;
                for i = yminyyy:ymin2yyy
                    for j = xminx1xxx:xminx2xxx
                        if w(i,j) == 1
                            w(i,j) = 0;
                        end
                    end
                end
```

ถ้าเข้าเงื่อนไขมีกลุ่มพิกเซลสามกลุ่มจะเป็นตัวพยัญชนะที่มีหัวอยู่กึ่งกลางและช่วงล่าง จึงต้องมีวิธีในการตรวจสอบว่าหัวอยู่ช่วงใด โดยการหาพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดของช่วงกลางและช่วงล่าง

โปรแกรมหาพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดของช่วงกลาง เก็บไว้ใน pt1 รูปที่ 4.50 และ 4.52

```
elseif (r2~=0)&&(s2~=0)&&(t2~=0)% แต่ถ้ามี 3 กลุ่มพิกเซลสีขาวจะต้องหาพื้นที่ในช่วงกลางและช่วงล่าง %
            [m,n] = size(w);
            a1 = m/3; %%% แบ่งความสูงออกเป็นสามช่วง %%%
            a2 = int16(a1);
%%% หาแถวที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงกลาง %%%
            a=0;b=0;c=0;maxxx=0;xmaxx=0;xminx=0;a1=0;B=0;v=0;
            for i = a2:a2*2
                for j = 2:n-1
                    if w(i,j) ~= 0 && w(i,j-1) == 0
                        a=j;
                    elseif w(i,j) ~= 0 && w(i,j+1) ==0
                        b=j;
                        c=b-a;
                        if c>maxxx
                            maxxx=c;
                        end
                    end
                end
            end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for i = a2:a2*2
    for j = 2:n-1
        if w(i,j) ~= 0 && w(i,j-1) == 0
            a=j;
        elseif w(i,j) ~= 0 && w(i,j+1) ==0
            b=j;
            c=b-a;
            if c==maxxx
                xminx=a;
                xmaxx=b;,break
            end
        end
    end
end
.....................................................
```

%%% หาหลักที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงกลาง %%%

```
d=0;e=0;f=0;maxyy=0;yminy=0;ymaxy=0;
for j = 2:n-1
    d=0;
    for i = a2:a2*2 %%%%%%%%%%%%%%%%%%%%%%%%%%%%
        if w(i,j) ~= 0 && w(i-1,j) == 0
            d = i;
        elseif w(i,j) ~= 0 & w(i+1,j) == 0
            e = i;
            if d == 0
                f = 0;
            elseif d ~= 0
                f = e-d;
            end
            if f>maxyy
                maxyy=f;
            end
        end
    end
end
for j = 2:n-1
    d=0;
    for i = a2:a2*2 %%%%%%%%%%%%%%%%%%%%%%%%%%%%
        if w(i,j) ~= 0 && w(i-1,j) == 0
            d = i;
        elseif w(i,j) ~= 0 & w(i+1,j) == 0
            e = i;
            if d == 0
                f = 0;
            elseif d ~= 0
                f = e-d;
            end
            if f==maxyy
                yminy=d;
                ymaxy=e;,break
            end
        end
    end
end
```

P = maxxx %%% จำนวนพิกเซลสีขาวตามความยาวในแถวที่มีพิกเซลสีขาวมากที่สุด %%%

L = maxyy %%% จำนวนพิกเซลสีขาวตามความสูงในหลักที่มีพิกเซลสีขาวมากที่สุด %%%

pt1 = P*L; %%% แถวและหลักที่มีจำนวนพิกเซลสีขาวติดกันมากที่สุดมาคูณกันได้พื้นที่ที่มากที่สุดในช่วงกลาง %%%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมหาพื้นที่ของกลุ่มพิกเซลสีขาวที่มากที่สุดของช่วงกลาง เก็บไว้ใน pt2 รูปที่ 4.50 และ 4.52

```
%%% หาแถวที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงล่าง %%%
            a=0;b=0;c=0;maxx=0;xmax=0;xmin=0;a1=0;B=0;v=0;
            for i = a2*2:m-1
                for j = 2:n-1
                    if w(i,j) ~= 0 && w(i,j-1) == 0
                        a=j;
                    elseif w(i,j) ~= 0 && w(i,j+1) ==0
                        b=j;
                        c=b-a;
                        if c>maxx
                            maxx=c;
                        end
                    end
                end
            end
            for i = a2*2:m-1
                for j = 2:n-1
                    if w(i,j) ~= 0 && w(i,j-1) == 0
                        a=j;
                    elseif w(i,j) ~= 0 && w(i,j+1) ==0
                        b=j;
                        c=b-a;
                        if c==maxx
                            xmin=a;
                            xmax=b;,break
                        end
                    end
                end
            end
            .....
%%% หาหลักที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงล่าง %%%
            d=0;e=0;f=0;maxy=0;ymin=0;ymax=0;
            for j = 2:n-1
                d=0;
                for i = a2*2:m-1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                    if w(i,j) ~= 0 && w(i-1,j) == 0
                        d = i;
                    elseif w(i,j) ~= 0 & w(i+1,j) == 0
                        e = i;
                        if d == 0
                            f = 0;
                        elseif d ~= 0
                            f = e-d;
                        end
                        if f>maxy
                            maxy=f;
                        end
                    end
                end
            end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        for j = 2:n-1
            d=0;
            for i = a2*2:m-1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                if w(i,j) ~= 0 && w(i-1,j) == 0
                    d = i;
                elseif w(i,j) ~= 0 & w(i+1,j) == 0
                    e = i;
                    if d == 0
                        f = 0;
                    elseif d ~= 0
                        f = e-d;
                    end
                    if f==maxy
                        ymin=d;
                        ymax=e;,break
                    end
                end
            end
        end
        pt2 = maxx*maxy; %%% แถวและหลักที่มีจำนวนพิกเซลสีขาวติดกันมากที่สุดมาคูณกัน ได้พื้นที่
                                        ที่มากที่สุดในช่วงล่าง %%%
```

นำพื้นที่ทั้งสองมาเปรียบเทียบกัน ถ้าพื้นที่ pt2 มากว่า pt1 จะเป็นพยัญชนะที่มีหัวอยู่ช่วงกึ่งกลางของตัวพยัญชนะ จึงทำการหาหัวที่อยู่ช่วงกึ่งกลางนั้นแล้วใส่หัวลงไปดังรูปที่ 4.51

```
        if pt2>pt1 % ถ้าพื้นที่ช่วงล่างมากกว่าช่วงกลาง จะเข้าเงื่อนไขหาหัวช่วงกลางโดยใช้ขอบเขตดังนี้
            [m,n] = size(w);
            v = m/5; %%% แบ่งความสูงออกเป็นห้าช่วง %%%
            vv = int16(v);
%%% หาแถวที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงกึ่งกลาง %%%
            a=0;b=0;c=0;maxx=0;xmax=0;xmin=0;
            for i = vv:vv*3 %%% ขอบเขตความสูงในช่วงที่สองถึงสามจากในห้าช่วง %%%
                for j = 2:n-1 %%% ขอบเขตความกว้างจากพิกเซลที่สองถึงรองสุดท้าย %%%
                    if w(i,j) ~= 0 && w(i,j-1) == 0
                        a=j;
                    elseif w(i,j) ~= 0 && w(i,j+1) ==0
                        b=j;
                        c=b-a;
                        if c>maxx
                            maxx=c;
                        end
                    end
                end
            end
            for i = vv:vv*3
                for j = 2:n-1
                    if w(i,j) ~= 0 && w(i,j-1) == 0
                        a=j;
                    elseif w(i,j) ~= 0 && w(i,j+1) ==0
                        b=j;
                        c=b-a;
                        if c==maxx
                            xmin=a;
                            xmax=b;o = i;,break
                        end
                    end
                end
            end
        end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%%% หาหลักที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงกึ่งกลาง %%%
                d=0;e=0;f=0;maxy=0;ymin=0;ymax=0;
                for j = 2:n-1 %%% ขอบเขตความกว้างจากพิกเซลที่สองถึงรองสุดท้าย %%%
                    d=0;
                    for i = vv:vv*3 %%% ขอบเขตความสูงในช่วงที่สองถึงสามจากในห้าช่วง %%%
                        if w(i,j) ~= 0 && w(i-1,j) == 0
                            d = i;
                        elseif w(i,j) ~= 0 & w(i+1,j) == 0
                            e = i;
                            if d == 0
                                f = 0;
                            elseif d ~= 0
                                f = e-d;
                            end
                            if f>maxy
                                maxy=f;
                            end
                        end
                    end
                end
                for j = 2:n-1
                    d=0;
                    for i = vv:vv*3
                    if w(i,j) ~= 0 && w(i-1,j) == 0
                        d = i;
                    elseif w(i,j) ~= 0 & w(i+1,j) == 0
                        e = i;
                        if d == 0
                            f = 0;
                        elseif d ~= 0
                            f = e-d;
                        end
                        if f==maxy
                            ymin=d;
                            ymax=e;,break
                        end
                    end
                end
            end
%%%%%%% ตีกรอบโดยใช้ความกว้างและสูงจากพิกเซลแรกและสุดท้ายที่เก็บไว้ตามแถวและหลัก %%%%%%
            for x1=xmin-1
                for y1=ymin-1:ymax+1
                    hold on
                    plot(x1,y1,'-r.')
                    hold off
                end
            end
            for x1=xmin-1:xmax+1
                for y1=ymax+1
                    hold on
                    plot(x1,y1,'-r.')
                    hold off
                end
            end
            for x1=xmin-1:xmax+1
                for y1=ymin-1
                    hold on
                    plot(x1,y1,'-r.')
                    hold off
                end
            end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                for x1=xmax+1
                    for y1=ymin-1:ymax+1
                        hold on
                        plot(x1,y1,'-r.')
                        hold off
                    end
                end
%%%%%%%%%%%%%%%%%% ใส่สี่เหลี่ยมสีดำลงไปตรงกึ่งกลางของกรอบที่ตีไว้ %%%%%%%%%%%%%%%%%%
                ymax;
                yy = maxy/3;
                yyy = int16(yy);
                xx = maxx/3;
                xxx = int16(xx);
                xminx1xxx = xmin+xxx;
                xminx2xxx = xmin+xxx*2;
                yminyyy = ymin+yyy;
                ymin2yyy = ymin+yyy*2;
                for i = yminyyy:ymin2yyy
                    for j = xminx1xxx:xminx2xxx
                        if w(i,j) == 1
                            w(i,j) = 0;
                        end
                    end
                end
```

แต่ถ้าพื้นที่ pt1 มากว่า pt2 จะเป็นพยัญชนะที่มีหัวอยู่ช่วงล่างของตัวพยัญชนะ จึงทำการหาหัวที่อยู่ช่วงล่างนั้นแล้วใส่หัวลงไปดังรูปที่ 4.53

```
            elseif pt1 > pt2 %%% แต่ถ้าพื้นที่ช่วงกลางมากกว่าช่วงล่าง จะเข้าเงื่อนไขหาหัวที่อยู่ช่วงล่าง %%
%%% หาแถวที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงล่าง %%%
                a=0;b=0;c=0;maxx=0;xmax=0;xmin=0;a1=0;B=0;v=0;
                for i = a2*2:m-1
                    for j = 2:n-1
                        if w(i,j) ~= 0 && w(i,j-1) == 0
                            a=j;
                        elseif w(i,j) ~= 0 && w(i,j+1) ==0
                            b=j;
                            c=b-a;
                            if c>maxx
                                maxx=c;
                            end
                        end
                    end
                end
                for i = a2*2:m-1
                    for j = 2:n-1
                        if w(i,j) ~= 0 && w(i,j-1) == 0
                            a=j;
                        elseif w(i,j) ~= 0 && w(i,j+1) ==0
                            b=j;
                            c=b-a;
                            if c==maxx
                                xmin=a;
                                xmax=b;,break
                            end
                        end
                    end
                end
                .....
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

%%% หาหลักที่มีพิกเซลสีขาวติดกันมากที่สุดในขอบเขตช่วงล่าง %%%

```
d=0;e=0;f=0;maxy=0;ymin=0;ymax=0;
for j = 2:n-1
    d=0;
    for i = a2*2:m-1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        if w(i,j) ~= 0 && w(i-1,j) == 0
            d = i;
        elseif w(i,j) ~= 0 & w(i+1,j) == 0
            e = i;
            if d == 0
                f = 0;
            elseif d ~= 0
                f = e-d;
            end
            if f>maxy
                maxy=f;
            end
        end
    end
end
for j = 2:n-1
    d=0;
    for i = a2*2:m-1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        if w(i,j) ~= 0 && w(i-1,j) == 0
            d = i;
        elseif w(i,j) ~= 0 & w(i+1,j) == 0
            e = i;
            if d == 0
                f = 0;
            elseif d ~= 0
                f = e-d;
            end
            if f==maxy
                ymin=d;
                ymax=e;,break
            end
        end
    end
end
```

%%%%%% ตีกรอบโดยใช้ความกว้างและสูงจากพิกเซลแรกและสุดท้ายที่เก็บไว้ตามแถวและหลัก %%%%%%

```
for x1=xmin-1
    for y1=ymin-1:ymax+1
        hold on
        plot(x1,y1,'-r.')
        hold off
    end
end
for x1=xmin-1:xmax+1
    for y1=ymax+1
        hold on
        plot(x1,y1,'-r.')
        hold off
    end
end
for x1=xmin-1:xmax+1
    for y1=ymin-1
        hold on
        plot(x1,y1,'-r.')
        hold off
    end
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                for x1=xmax+1
                    for y1=ymin-1:ymax+1
                        hold on
                        plot(x1,y1,'-r.')
                        hold off
                    end
                end
%%%%%%%%%%%%%%%%% ใส่สี่เหลี่ยมสีดำลงไปตรงกึ่งกลางของกรอบที่ตีไว้ %%%%%%%%%%%%%%%%%
                ymax;
                ymin;
                yy = maxy/3;
                yyy = int16(yy);
                xx = maxx/3;
                xxx = int16(xx);
                xminx1xxx = xmin+yyy;
                xminx2xxx = xmin+yyy*2;
                yminyyy = ymin+yyy
                ymin2yyy = ymin+yyy*2
                for i = yminyyy:ymin2yyy
                    for j = xminx1xxx:xminx2xxx
                        if w(i,j) == 1
                            w(i,j) = 0;
                        end
                    end
                end
            end
        end
    end
end
```

.....

โปรแกรมหาเส้นโครงร่างของตัวเลขและตัวพยัญชนะหลังจากทำภาพเป็นภาพเป็นไบนารี่ และ inverse ให้ตัวเลขและพยัญชนะเป็นสีขาวแล้ว

```
[m1,n1]=size(f);
for v=1:25  %%% กำหนดจำนวนรอบในการวนทั้งภาพ %%%
    for j=2:n1-1  %%% วนจากซ้ายไปขวาจากหลักที่สองถึงหลักรองสุดท้าย %%%
        for i=2:m1-1  %%% วนจากบนลงล่างจากแถวที่สองถึงแถวรองสุดท้าย %%%
            if f(i,j) == 1  % ถ้าพิกเซลกึ่งกลางของพิกเซล 3*3 เป็นหนึ่ง จะตามข้อกำหนดด้านล่าง%
                A(1)=f(i-1,j-1);
                A(2)=f(i-1,j);
                A(3)=f(i-1,j+1);
                A(4)=f(i,j+1);
                A(5)=f(i+1,j+1);
                A(6)=f(i+1,j);
                A(7)=f(i+1,j-1);
                A(8)=f(i,j-1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    s = 0; %%% กำหนดตัวแปร s เป็นศูนย์ %%%
                    for u = 1:7 %%% วนตั้งแต่พิกเซล A(1) ถึง A(8) %%%
                          a = A(u)-A(u+1); %%% โดนนำพิกเซลหน้าลบหลังแล้วเก็บค่าไว้ใน a %%%
                          if a < 0 %%% a น้อยกว่าศูนย์หรือติดลบหรือหน้าเป็นศูนย์และหลังเป็นหนึ่ง %%%
                                s = s+1; %%% ให้ s เป็นหนึ่งและบวกหนึ่งไปเรื่อยๆถ้า a ที่ตำแหน่งถัดไป
                          end                    ติดลบอีก %%%
                    end
                    a = A(8)-A(1); %%% พิกเซลสุดท้ายคือ 8 ต้องลบกับพิกเซลแรก %%%
                    if a < 0
                         s = s+1;
                    end
                    N = A(1)+A(2)+A(3)+A(4)+A(5)+A(6)+A(7)+A(8); %%% นำค่าทั้งแปด
                                                                  พิกเซลมารวมกัน %%%
                    R = A(2) & A(4) & A(6); %%% นำค่าในพิกเซล 2,4และ 6 มาคูณกัน %%%
                    L = A(4) & A(6) & A(8); %%% นำค่าในพิกเซล 4,6และ 8 มาคูณกัน %%%
                    if (s == 1 &  R == 0 & L == 0  & (N >= 2 && N <= 6)) %%% ถ้า s
เป็นหนึ่ง และ R เป็นศูนย์ และ L เป็นศูนย์ และ N มากกว่าหรือเท่ากับสองและน้อยกว่าหรือเท่ากับหก %%%
                         I(i,j) = 0; %% จะเปลี่ยนพิกเซลกึ่งกลางเป็นศูนย์แล้วเก็บไว้ในตัวแปรใหม่ I %%
                    else %%% แต่ถ้าไม่เข้าเงื่อนไขทั้งหมด %%%
                         I(i,j) = f(i,j); %%% พิกเซลกึ่งกลางเป็นเหมือนเดิมแล้วเก็บไว้ใน I %%%
                    end
               else %%% แต่ถ้าไม่เข้าตั้งแต่เงื่อนแรกที่ให้พิกเซลกึ่งกลางเป็นหนึ่ง %%%
                    I(i,j) = f(i,j); %%% ทำให้พิกเซลกึ่งกลางเป็นเหมือนเดิมแล้วเก็บไว้ใน I %%%
               end
          end
     end
     f=I; %%% เปลี่ยนจากตัวแปร I ไปเป็น f แล้ววนทั่วทั้งภาพอีกครั้ง %%%
     for j=2:n1-1 %%% วนจากซ้ายไปขวาจากหลักที่สองถึงหลักรองสุดท้าย %%%
          for i=2:m1-1 %%% วนจากบนลงล่างจากแถวที่สองถึงแถวรองสุดท้าย %%%
               if f(i,j) == 1 % ถ้าพิกเซลกึ่งกลางของพิกเซล 3*3 เป็นหนึ่ง จะตามข้อกำหนดด้านล่าง%
                    A(1)=f(i-1,j-1);
                    A(2)=f(i-1,j);
                    A(3)=f(i-1,j+1);
                    A(4)=f(i,j+1);
                    A(5)=f(i+1,j+1);
                    A(6)=f(i+1,j);
                    A(7)=f(i+1,j-1);
                    A(8)=f(i,j-1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    s = 0; %%% กำหนดตัวแปร s เป็นศูนย์ %%%
                    for u = 1:7
                         a = A(u)-A(u+1);
                         if a < 0
                              s = s+1;
                         end
                    end
                    a = A(8)-A(1);
                    if a < 0
                         s = s+1;
                    end
                    N = A(1)+A(2)+A(3)+A(4)+A(5)+A(6)+A(7)+A(8);
                    H = A(2) & A(4) & A(8); %%% นำค่าในพิกเซล 2,4และ 8 มาคูณกัน %%%
                    K = A(2) & A(6) & A(8); %%% นำค่าในพิกเซล 2,6และ 8 มาคูณกัน %%%
                    if (s == 1 &  K == 0 & H == 0  & (N >= 2 && N <= 6)) %%% ถ้า s
เป็นหนึ่ง และ K เป็นศูนย์ และ H เป็นศูนย์ และ N มากกว่าหรือเท่ากับสองและน้อยกว่าหรือเท่ากับหก %%%
                         I(i,j) = 0; %% จะเปลี่ยนพิกเซลกึ่งกลางเป็นศูนย์แล้วเก็บไว้ในตัวแปรใหม่ I %%
                    Else %%% แต่ถ้าไม่เข้าเงื่อนไขทั้งหมด %%%
                         I(i,j) = f(i,j); %%% พิกเซลกึ่งกลางเป็นเหมือนเดิมแล้วเก็บไว้ใน I %%%
                    end
               else %%% แต่ถ้าไม่เข้าตั้งแต่เงื่อนแรกที่ให้พิกเซลกึ่งกลางเป็นหนึ่ง %%%
                    I(i,j) = f(i,j); %%% ทำให้พิกเซลกึ่งกลางเป็นเหมือนเดิมแล้วเก็บไว้ใน I %%%
               end
          end
     end
     f=I;
end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้