สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การรู้จำตัวอักษรภาษาไทยสำหรับโทรศัพท์มือถือ

Thai Character Recognition for mobile phone

โดย

น.ส. ปอลิน สีหาวงษ์

น.ส. ปัณฑ์ชนิต อรรธนิศาสุข

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขหมู่.....
เลขทะเบียน 72218
วัน,เดือน,ปี 12 มิ.ย. 2550

.b 11716210
.i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thai Character Recognition for mobile phone

BY

Miss. Pauline Sihawong

Miss. Phunchanit Autanisasuk

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENT FOR THE DEGREE OF

BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2006

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	การรู้จำตัวอักษรภาษาไทยสำหรับโทรศัพท์มือถือ
ชื่อนักศึกษา	น.ส. ปอลิน สีหาวงษ์ รหัสนักศึกษา 46010442
	น.ส. ปัณฑ์ชนิต อรรธนิศาสุข รหัสนักศึกษา 46010445
อาจารย์ที่ปรึกษา	อาจารย์ ภุชงค์ หงษ์สุวรรณ
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตร์บัณฑิต
	สาขาวิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2549

ปริญญานิพนธ์ฉบับนี้ได้รับความเห็นชอบจากอาจารย์ที่ปรึกษาเป็นที่เรียบร้อยแล้ว

.....

(อาจารย์ ภุชงค์ หงษ์สุวรรณ)

อาจารย์ผู้ควบคุมปริญญานิพนธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ การรู้จำตัวอักษรภาษาไทยสำหรับโทรศัพท์มือถือ

นักศึกษา นางสาวปอลิน สีหาวงษ์ รหัสนักศึกษา 46010442
นางสาวปัณฑ์ชนิต อรรธนิศาสุข รหัสนักศึกษา 46010445

อาจารย์ที่ปรึกษา อาจารย์ ภุชงค์ หงษ์สุวรรณ

ระดับการศึกษา ปริญญาตรี วิศวกรรมศาสตร์บัณฑิต
สาขาวิศวกรรมสารสนเทศ

ภาควิชา วิศวกรรมสารสนเทศ

ปีการศึกษา 2549

บทคัดย่อ

ในปัจจุบัน โทรศัพท์มือถือได้เข้ามามีบทบาทอย่างมาก จนเปรียบเสมือนเป็นปัจจัยที่ 5 ของมนุษย์ จึงมีความพยายามที่จะเพิ่มขีดความสามารถให้กับโทรศัพท์มือถือในหลายๆทาง เพื่อเพิ่มประโยชน์ใช้สอยและความสะดวกต่างๆ ให้กับผู้ใช้ได้มากขึ้น ผู้จัดทำจึงได้นำเสนอโครงงานการรู้จำตัวอักษรภาษาไทยสำหรับโทรศัพท์มือถือ ซึ่งจะสามารถอำนวยความสะดวกให้กับผู้ใช้โทรศัพท์มือถือในการเก็บข้อความที่ได้พบเห็นโดยการถ่ายรูป(ไม่ต้องผ่านการพิมพ์จากแป้นกด) มีแนวคิดคือ การแปลงภาพตัวอักษรให้เป็นข้อความตัวอักษรที่สามารถแก้ไขได้ และในกรณีที่ต้องการส่งภาพข้อความก็สามารถส่งเป็นข้อความแบบ SMS แทนการส่งแบบ MMS ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Thai Character Recognition for Mobile Phone.	
Student	Miss. Pauline Sihawong	ID. 46010442
	Miss. Phunchanit Autanisasuk	ID. 46010445
Advisor	Mr. Puchong Hongsuwan	
Graduate Level	Bachelor Degree of Information Engineering	
Department	Information Engineering	
Academic Year	2006	

Abstracts

At present, the mobile phone seems to be one of the most essential requisites of human beings. There have been many attempts to raise their capabilities in several ways in order to improve their utilities and conveniences to the users. In this project, we present the "Thai Characters Recognition for Mobile Phone". Our project aims to facilitate the users to collect any messages by capturing the pictures (no typing needed). The concept is to transform the message image into the text message in order that the users can edit it as a regular text message. According to that concept, the messages will be sent in SMS format instead of MMS format.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบพระคุณ อาจารย์ภุชงค์ หงส์สุวรรณ และ อาจารย์สมเกียรติ อุดมหรรษากุล ที่ได้ให้คำปรึกษาและคำแนะนำเป็นอย่างดีจนทำให้ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วง และขอขอบคุณน้องปอม ผู้เต็มใจอุทิศโทรศัพท์มือถือให้ใช้ในการทดลองเสมอมา เป็นเวลาเกือบๆครึ่งปี ขอบคุณเพื่อนๆ พี่ๆ น้องๆ ที่คอยช่วยเหลือให้คำปรึกษาและเป็นกำลังใจให้ สุดท้ายที่จะขาดไม่ได้ ต้องขอขอบคุณครอบครัวสีหาวงษ์ และครอบครัวอรรธนิศาสุขในความสนับสนุนทุกๆอย่างจนประสบความสำเร็จ ณ วันนี้

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทคัดย่อภาษาไทย ก

บทคัดย่อภาษาอังกฤษ ข

สารบัญ ค

สารบัญรูปประกอบ ง

สารบัญตาราง ฉ

บทที่ 1 บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 ความมุ่งหมายและวัตถุประสงค์ 2

1.3 ขอบเขตของโครงงาน 2

1.4 สถาปัตยกรรมของระบบ 3

1.5 ขั้นตอนการดำเนินโครงงาน 4

1.6 ฮาร์ดแวร์และซอฟท์แวร์ 4

บทที่ 2 ทฤษฎีและหลักการที่นำมาใช้

2.1 คุณสมบัติทางโทโปโลยีของภาพดิจิตอล 6

2.2 การติดตามขอบภาพ 8

2.3 การหาขนาดความกว้างและความสูงของตัวอักษร 9

2.4 Noise Reduction 11

2.5 Threshold 12

2.6 การตัดตัวอักษร 13

2.7 การทำลายเส้นตัวอักษรให้บาง 15

2.8 พิกเซลที่มีความหนา 1 จุดภาพ 16

2.9 การกำจัดส่วนเกิน 17

2.10 การวิเคราะห์ หารหัสแทนตัวอักษร 19

2.11 การหาเส้นบรรทัด 23

2.12 การรู้จำตัวอักษร 24

2.13 J2ME 25

บทที่ 3 การออกแบบ และ Flow-chart 33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4 ผลการวิเคราะห์ภาพอักษร 43

บทที่ 5 สรุป 46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปประกอบ

รูปที่

1.1 Block Diagram เปรียบเทียบการรู้จำตัวอักษรของมนุษย์กับโทรศัพท์มือถือ 1

1.2 สถาปัตยกรรมของระบบ การรู้จำตัวอักษร ภาษาไทยสำหรับโทรศัพท์มือถือ 3

2.1 หน้าต่างขนาด 3 x 3 บิต 6

2.2 ลักษณะคุณสมบัติทางโทโปโลจิคอลของจุดภายใน 6

2.3 ลักษณะคุณสมบัติทางโทโปโลจิคอลของจุดโดดเดี่ยว 6

2.4 ลักษณะคุณสมบัติทางโทโปโลจิคอลของจุดตัด 7

2.5 ลักษณะคุณสมบัติทางโทโปโลจิคอลของจุดต่อ 7

2.6 ลักษณะคุณสมบัติทางโทโปโลจิคอลของจุดแยก 7

2.7 ลักษณะคุณสมบัติทางโทโปโลจิคอลของจุดปลาย 7

2.8 การติดตามหาจุดภาพที่เป็นขอบภาพ 8

2.9 การติดตามหาจุดภาพสีดำในลำดับถัดไปในทิศทางตามเข็มนาฬิกาที่ตำแหน่ง Pn 9

2.10 การติดตามหาจุดภาพสีดำในลำดับถัดไปในทิศทางตามเข็มนาฬิกาที่ตำแหน่ง Pn+1 9

2.11 ลักษณะจุดพิกัดของตัวอักษร ที่นำมาคำนวณหาค่าความกว้างและความสูงของตัวอักษร 10

2.12 มาส์คแบบเกาส์เซียนฟิลเตอร์(σ =1) 11

2.13 การคอนโวลูชันข้อมูลรูปภาพด้วยมาส์ค 12

2.14 ผลการทำให้เป็นภาพขาวดำ โดยใช้ค่า Threshold ต่างๆกัน 13

2.15 การสแกนภาพของแนวเส้นแสกน 13

2.16 การสแกนภาพเพื่อหาสระบน สระล่าง และวรรณยุกต์อื่นๆ 14

2.17 การสแกนภาพของเส้นแสกน ต่อจากจุดเดิม 14

2.18 ภาพอักษรก่อนทำให้บาง และหลังทำให้บาง 15

2.19 พิกเซลที่จะถูกลบ เมื่อทำการตรวจกวาดในทิศทางจากซ้ายไปขวา 1 ครั้ง 15

2.20 ภาพที่ผ่านกระบวนการทำให้บาง จนเหลือความหนาเพียง 1 จุดภาพ 16

2.21 พิกเซลที่มีความหนา 1 จุดภาพแบบ ก 16

2.22 พิกเซลที่มีความหนา 1 จุดภาพแบบ ข 17

2.23 กรณียกเว้นของพิกเซลที่มีความหนา 1 จุดภาพแบบ ก 17

2.24 ภาพอักษรก่อนการกำจัดส่วนเกิน และหลังการกำจัดส่วนเกิน 18

2.25 การทำงานของการกำจัดส่วนเกิน 18

2.26 ตัวอย่างของจุดภาพที่จะนำมาใช้วิเคราะห์หารหัส Q-Code 20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.27 ตัวอย่างลักษณะของคุณสมบัติโทโปโลยีในภาพอักษร 21
2.28 การตรวจสอบภาพเพื่อปรับกรอบภาพใหม่ 21
2.29 ภาพตัวอักษรที่ปรับกรอบภาพใหม่แล้ว 22
2.30 การแบ่งภาพอักษรออกเป็น 9 ส่วน 22
2.31 การเก็บข้อมูลคุณสมบัติทางโทโปโลยีในหน่วยความจำขนาด 1 ไบต์ (X = 0, 1) 22
2.32 การทำหาเส้นบรรทัด 23
2.33 ค่าต่างๆ ที่ต้องพิจารณาในการหาคุณสมบัติเกี่ยวกับตำแหน่งบนเส้นบรรทัด ของตัวอักษร 25
2.34 วงจรการทำงานของมิดเลท 28
2.35 องค์ประกอบของชุดมิดเลท 30
2.36 ขั้นตอนการพัฒนามิดเลท 32
3.1 Block Diagram แสดงการทำงานของระบบ 33
3.2 Flow-chart การทำงานของขั้นตอนการทำเป็นภาพขาวดำ 34
3.3 Flow-chart การทำงานของขั้นตอนการตัดแยกตัวอักษร 35
3.4 Flow-chart การทำงานของขั้นตอนการทำตัวอักษรให้บาง 36
3.5 Flow-chart การทำงานของขั้นตอนการทำกำจัดส่วนเกินของตัวอักษร 37
3.6 Flow-chart การทำงานของขั้นตอนการหาเส้นบรรทัดตัว 38
3.7 หน้าจอภาพ 44
3.8 ปุ่มถ่ายภาพและปุ่มตรงกลาง 44
3.9 หน้าจอโทรศัพท์มือถือที่แสดงข้อความเมื่อกดถ่ายภาพ 45
3.10 หน้าจอโทรศัพท์มือถือที่แสดงภาพที่ถ่ายไว้ 45
3.11 ปุ่มประมวลผล 46
3.12 หน้าจอโทรศัพท์มือถือที่แสดงข้อความเมื่อกดประมวลผล 46
3.13 รายชื่อ Access point 47
3.14 กล่องข้อความแสดงข้อความ "ผู้หญิงพิมพ์ฎีกา" บนหน้าจอโทรศัพท์มือถือ 47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่

4.1 ผลการทดสอบตัวอักษรจากการสุ่มตัวอย่าง 150 ภาพข้อความ 43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

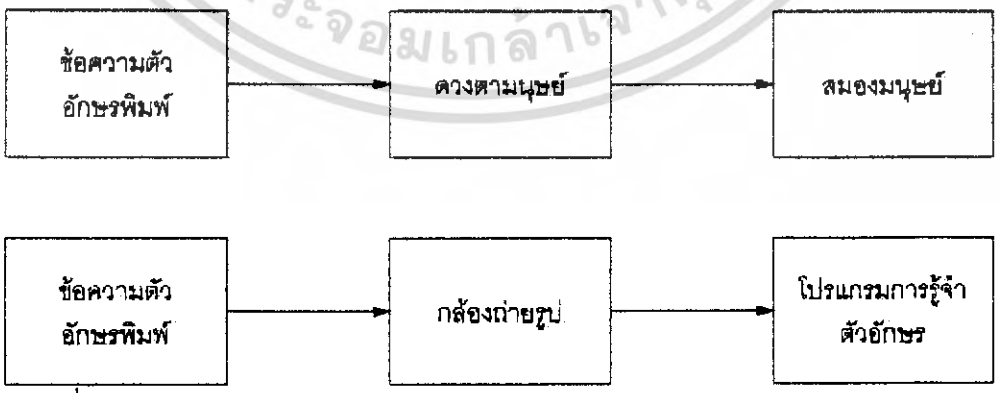
บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันซึ่งเป็นยุคของการสื่อสารโทรคมนาคม โทรศัพท์มือถือเป็นสิ่งอำนวยความสะดวกที่มีการใช้กันอย่างแพร่หลาย และมีการพัฒนาโปรแกรมประยุกต์ออกมามากมายหลายรูปแบบ กล้องในโทรศัพท์มือถือมีความละเอียดมากขึ้น และโทรศัพท์มือถือก็มีความสามารถในการประมวลผลเพิ่มขึ้นเรื่อยๆ เพื่อรองรับกับการพัฒนาโปรแกรมประยุกต์ใหม่ ๆ ขึ้นมาเพื่อเพิ่มประโยชน์ใช้สอยและอำนวยความสะดวกให้กับผู้ใช้

จะเห็นได้ว่าวิทยาการการแปลงตัวอักษรบนผืนภาพมาเป็นไฟล์ตัวอักษรที่สามารถจัดการ (เพิ่มเติม – แก้ไข)ได้ หรือ Optical Character Recognition (OCR) นั้น มีการใช้งานกันอยู่แล้วในปัจจุบัน ในรูปแบบของการสแกนภาพจากเครื่องสแกนเนอร์ แล้วผ่านกระบวนการแปลงภาพสีเป็นภาพขาว-ดำ จากนั้นนำมาผ่านกระบวนการ โอซีอาร์ และพัฒนามาจนถึงขั้นใช้กล้องดิจิตอลถ่ายภาพแทนเครื่องสแกนเนอร์ แต่ยังไม่มีการใช้งานในรูปแบบของกล้องในโทรศัพท์มือถือโดยเฉพาะอย่างยิ่งในการทำการรู้จำตัวอักษรภาษาไทยให้ได้ใช้งานกันอย่างแพร่หลาย

โครงงานการรู้จำตัวอักษรภาษาไทยสำหรับโทรศัพท์มือถือนี้ เป็นการประยุกต์ใช้งานโทรศัพท์มือถือในอีกลักษณะหนึ่งที่พยายามจะทำให้โทรศัพท์มือถือมองลักษณะแล้วทำการเลียนแบบความสามารถในการรู้จำรูปแบบอักษร หรืออ่านอักษรพิมพ์ที่พบเห็นโดยการถ่ายรูป(ไม่ต้องผ่านการพิมพ์จากแป้นกด)ได้อย่างสมองมนุษย์ที่สามารถรู้จำได้จากการมองเห็นด้วยตา โดยต้องคำนึงถึงข้อจำกัดในด้านความสามารถในการประมวลผล และหน่วยความจำของโทรศัพท์มือถือ รวมทั้งคุณภาพของกล้องในโทรศัพท์มือถือที่ใช้งานกันในปัจจุบันด้วย



รูปที่ 1.1 Block Diagram เปรียบเทียบการรู้จำตัวอักษรของมนุษย์กับโทรศัพท์มือถือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 ความมุ่งหมายและวัตถุประสงค์

1.2.1 เพื่อศึกษาและวิเคราะห์แนวทางในการถ่ายภาพโดยใช้โทรศัพท์มือถือเพื่อนำมาประมวลผลให้ได้ประสิทธิภาพสูงสุด

1.2.2 เพื่อศึกษากระบวนการปรับคุณภาพของภาพข้อความให้ชัดเจนสำหรับนำไปเข้าสู่ขั้นตอนของการทำระบบรู้จำตัวอักษร (OCR)

1.2.3 เพื่อศึกษากระบวนการทำระบบรู้จำตัวอักษรภาษาไทยโดยพิจารณาโครงสร้างของตัวอักษรหรือ โทโปโลจิคอลสำหรับภาพดิจิตอลทั่วไป (Topological Properties Extraction for general Digital Picture)

1.2.4 เพื่อศึกษาถึงความเป็นไปได้ในการประมวลผลเกี่ยวกับด้านการประมวลผลภาพ (Image Precessing) บนเครื่องโทรศัพท์มือถือ

1.3 ขอบเขตของโครงงาน

1.3.1 ชุดข้อมูลที่ใช้ประกอบไปด้วยภาพข้อความตัวอักษรภาษาไทยที่เป็นตัวพิมพ์ (พยัญชนะ, สระ และวรรณยุกต์) ในหลายๆ รูปแบบ

1.3.2 มีเส้นสแกนในการถ่ายรูป และประมวลผลภาพเพียง 1 บรรทัดที่เส้นสแกนลากผ่านในการถ่ายรูปแต่ละครั้ง

1.3.3 ผลการทดสอบจะวัดจากประสิทธิภาพในการรู้จำตัวอักษร

1.3.4 จะทำการเขียนโปรแกรมภาษา C++ บนเครื่องคอมพิวเตอร์ในการประมวลผลเพื่อรู้จำข้อมูล

1.3.5 จะเขียนโปรแกรมทำหน้าที่เป็นเหมือนโปรแกรมประยุกต์เซิร์ฟเวอร์ ในการติดต่อกับโทรศัพท์มือถือโดยใช้ J2SE

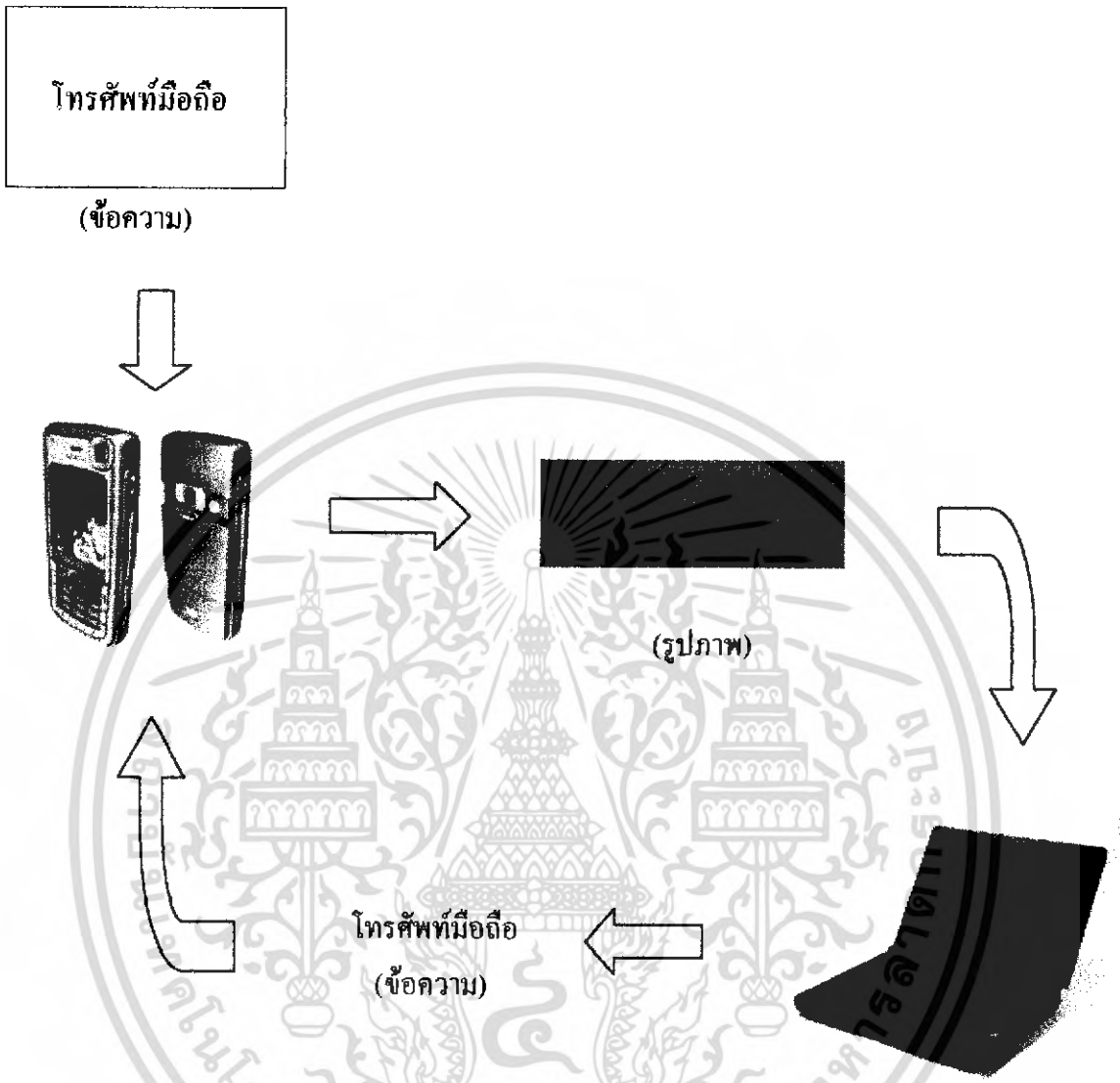
1.3.6 จะพัฒนาโปรแกรมประยุกต์บนโทรศัพท์มือถือ โดยใช้ J2ME

1.3.7 การติดต่อระหว่างโทรศัพท์มือถือ และเครื่องแม่ข่าย จะทำการติดต่อผ่านจีพีอาร์เอส (GPRS)

1.3.8 ภาพตัวอักษรที่ใช้เป็นภาพพื้นหลังสีอ่อน และตัวอักษรสีเข้มเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 สถาปัตยกรรมของระบบ



รูปที่ 1.2 สถาปัตยกรรมของระบบ การรู้จำตัวอักษรภาษาไทยสำหรับโทรศัพท์มือถือ

กระบวนการทำงานของระบบ แบ่งออกได้เป็น 3 ส่วนหลัก คือ

1.4.1 กระบวนการจัดเตรียมข้อมูลตัวอักษร (Image Processing) คือ กระบวนการในการเตรียมภาพตัวอักษรก่อนที่จะนำไปวิเคราะห์รู้จำตัวอักษร ได้แก่ การลดสัญญาณรบกวนในภาพ, การปรับภาพให้เป็นภาพขาวดำ, การแยกตัวอักษรแต่ละตัวออกจากกัน, การทำตัวอักษรให้บาง และสุดท้าย การกำจัดส่วนเกิน ซึ่งกระบวนการทั้งหมดจะทำบนเครื่องเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4.2 กระบวนการวิเคราะห์รู้จำตัวอักษร (Recognition) คือ กระบวนการในการวิเคราะห์หาข้อความที่ถูกต้อง ได้แก่ การหารหัสแทนตัวอักษรคิวโค้ด (Q-code), วิเคราะห์หารหัสตัวอักษรเป็นรายตัว และปรับปรุงข้อความให้ถูกต้องยิ่งขึ้น ซึ่งกระบวนการทั้งหมดจะทำบนเครื่อง เซิร์ฟเวอร์

1.4.3 กระบวนการติดต่อระหว่าง ไคลเอนท์ – เซิร์ฟเวอร์ (GPRS Connection) คือ กระบวนการในการส่งข้อมูลระหว่างโทรศัพท์มือถือและ เซิร์ฟเวอร์ ได้แก่ การส่งภาพข้อความที่ได้จากกล้องจากโทรศัพท์มือถือ ไปยัง เซิร์ฟเวอร์ ซึ่งเป็นกระบวนการที่ทำบนโทรศัพท์มือถือ และการส่งข้อความที่วิเคราะห์ได้จากเซิร์ฟเวอร์ ไปยังโทรศัพท์มือถือ ซึ่งเป็นกระบวนการที่ทำบนเซิร์ฟเวอร์

1.5 ขั้นตอนของการทำโครงงาน

1.5.1 ศึกษาค้นคว้าข้อมูลเกี่ยวกับ การเขียนโปรแกรมด้วย Visual C++, กระบวนการประมวลผลภาพ, อัลกอริธึมในการรู้จำ และ ขีดความสามารถและข้อจำกัดของโทรศัพท์มือถือ

1.5.2 เก็บข้อมูลตัวอย่าง(ภาพข้อความ) จากแหล่งต่างๆ ด้วยกล้องในโทรศัพท์มือถือ

1.5.3 นำข้อมูลที่ได้มาผ่านกระบวนการปรับปรุงภาพ (Image Enhancing) ได้แก่ การกรองสัญญาณรบกวนในภาพ (Noise reduction) และ การทำเป็นภาพขาวดำ (Adaptive Threshold)

1.5.4 ทำกระบวนการแยกตัวอักษร (Segmentation) เพื่อแยกตัวอักษร พยัญชนะ สระ และ วรรณยุกต์ ต่างๆออกจากกัน

1.5.5 ทำตัวอักษรให้บาง และ กำจัดส่วนเกิน

1.5.6 นำข้อมูลที่ได้มาทำการหาค่ารหัสคิวโค้ด ของตัวอักษร

1.5.7 ทำการวิเคราะห์หากระบวนการปรับปรุงข้อความให้มีความถูกต้องยิ่งขึ้น

1.5.8 ศึกษาค้นคว้าข้อมูลเกี่ยวกับ การเชื่อมต่อโทรศัพท์มือถือเข้าสู่ระบบเครือข่าย ผ่านเครือข่ายจีพีอาร์เอส, การเชื่อมต่อโปรแกรมเข้าสู่ระบบเครือข่าย ผ่านเครือข่ายอินเตอร์เน็ต และการติดต่อระหว่างโปรแกรมบนเครื่องคอมพิวเตอร์ผ่านซอคเกต (Socket)

1.5.9 ทำการทดสอบโปรแกรม

1.5.10 ประเมินผลประสิทธิภาพการรู้จำที่ได้จากการทดสอบ และสรุปผลการทดสอบ

1.6 ซอฟท์แวร์ และฮาร์ดแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6.1 ใช้โปรแกรม Visual C++ และไลบรารี OpenCV ในการสร้างโปรแกรมการรู้จำตัวอักษร

1.6.2 ใช้โทรศัพท์มือถือที่เป็นระบบซิมเบียน (Symbian) และมีกล้องถ่ายรูปในตัว

1.6.3 พัฒนาโปรแกรมประยุกต์บนโทรศัพท์มือถือ โดยใช้ J2SE

1.6.4 ติดต่อกับโทรศัพท์มือถือผ่านซอคเกต โดยใช้ J2ME

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการต่างๆที่นำมาใช้

2.1 คุณสมบัติทางโทโปโลยีของภาพดิจิตอล (*Topological Properties*)

คุณสมบัติทางโทโปโลยีของภาพดิจิตอล จะใช้จำแนกกลุ่มของจุดภาพ โดยวิธีการหาค่าคุณสมบัติทางโทโปโลยีของภาพดิจิตอลนี้ จะใช้วิธีการกำหนดหน้าต่างขนาด 3 x 3 ครอบบิตที่ต้องการหา ซึ่งหน้าต่างขนาด 3 x 3 มีลักษณะดังรูปที่ 2.1 ค่า X0, X1, X2, X3, X4, X5, X6, X7 และ X8 เป็นบิตทางดิจิตอลมีค่าเป็น 0 หรือ 1 โดย X0 คือ จุดที่เราจะพิจารณาและจะพิจารณาเฉพาะเมื่อ X0 = 1(จุดภาพเป็นสีดำ) เท่านั้น

X2	X3	X4
X1	X0	X5
X8	X7	X6

รูปที่ 2.1 หน้าต่างขนาด 3 x 3 บิต

การหาค่าคุณสมบัติทางโทโปโลจิคอล สำหรับภาพดิจิตอลทั่วไป สามารถกำหนดความแตกต่างของกลุ่มจุดภาพในตารางหน้าต่าง 3 x 3 ออกได้เป็น 6 กลุ่ม ดังนี้

1) จุดภายใน (Internal)

รูปที่ 2.2 ลักษณะคุณสมบัติทางโทโปโลจิคอลของจุดภายใน

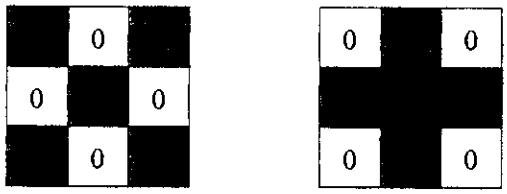
2) จุดโดดเดี่ยว (Isolate)

0	0	0
0		0
0	0	0

รูปที่ 2.3 ลักษณะคุณสมบัติทางโทโปโลจิคอลของจุดโดดเดี่ยว

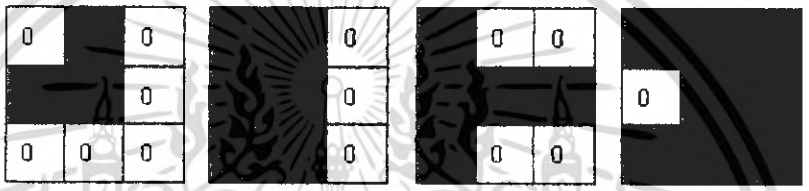
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) จุดตัด (Cross)



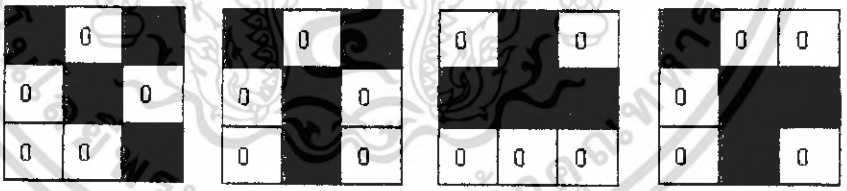
รูปที่ 2.4 ลักษณะคุณสมบัติทางโทโปโลจิคอลของจุดตัด

4) จุดต่อ (Connect)



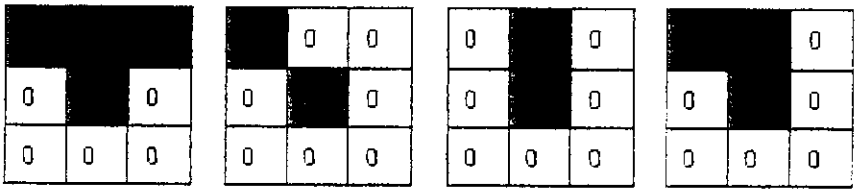
รูปที่ 2.5 ลักษณะคุณสมบัติทางโทโปโลจิคอลของจุดต่อ

5) จุดแยก (Branch)



รูปที่ 2.6 ลักษณะคุณสมบัติทางโทโปโลจิคอลของจุดแยก

6) จุดปลาย (End)



รูปที่ 2.7 ลักษณะคุณสมบัติทางโทโปโลจิคอลของจุดปลาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.2 เป็นจุดภายใน จะไม่มีส่วนประกอบรอบๆที่มีการเปลี่ยนแปลงจากค่า 1, 0 ไปเป็น 0, 1 เลย และส่วนประกอบรอบๆจะมีค่าเป็น 1 ทั้งหมด

รูปที่ 2.3 เป็นจุดโดดเดี่ยว จะไม่มีส่วนประกอบรอบๆที่มีการเปลี่ยนแปลงจากค่า 0, 1 ไปเป็น 1, 0 เลย และส่วนประกอบรอบๆจะมีค่าเป็น 0 ทั้งหมด

รูปที่ 2.4 เป็นจุดตัด ที่มีส่วนประกอบรอบๆที่มีการเปลี่ยนแปลงจากค่า 1, 0 ไปเป็น 0, 1อยู่ 8 ช่วง

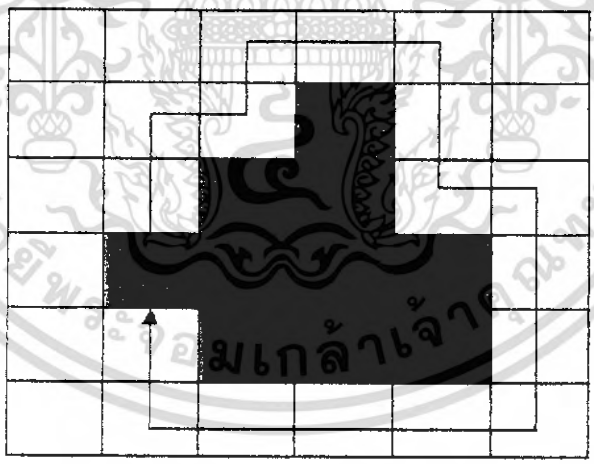
รูปที่ 2.6 เป็นจุดแยก ที่มีส่วนประกอบรอบๆที่มีการเปลี่ยนแปลงจากค่า 1, 0 ไปเป็น 0, 1 อยู่ 6 ช่วง

รูปที่ 2.7 เป็นจุดปลาย ที่มีส่วนประกอบรอบๆที่มีการเปลี่ยนแปลงจากค่า 1, 0 ไปเป็น 0, 1 อยู่ 2 ช่วง

ในกรณีของรูปที่ 2.5 ได้จัดแสดงไว้เพียงบางส่วนเท่านั้น ซึ่งส่วนที่ไม่มีในรูป 2.5 ก็คือจุดต่อทั้งหมด ซึ่งสามารถสังเกตเห็นความแตกต่างได้ชัดเจนในการจำแนกเป็นจุดภาพจากรูป

2.2 การติดตามขอบภาพ (Contour)

จะทำการติดตามขอบภาพโดยเริ่มจากจุดภาพสีดำที่มีค่าเป็น 1 ติดตามหาจุดภาพสีดำในลำดับถัดไปในทิศทางตามเข็มนาฬิกา แล้วเปลี่ยนตำแหน่งที่พิจารณาไปที่จุดนั้นๆ ทำเช่นนี้ไปเรื่อยๆจนกระทั่งจุดที่พิจารณาปัจจุบันกลับมาตรงกับจุดที่พิจารณาจุดแรกอีกครั้ง ก็จะถือว่าเสร็จสิ้นกระบวนการติดตามขอบภาพ ดังรูปที่ 2.8



รูปที่ 2.8 การติดตามหาจุดภาพที่เป็นขอบภาพ

เมื่อ Pn คือ ตำแหน่งของภาพที่พิจารณาจุดแรก

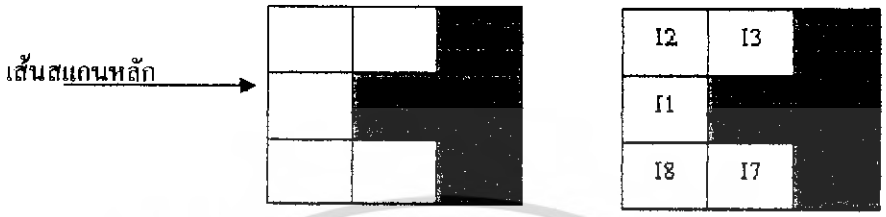
Pn+1 คือ ตำแหน่งของภาพที่จะพิจารณาถัดจาก Pn

Pn+2 คือ ตำแหน่งของภาพที่จะพิจารณาถัดจาก Pn+1

...

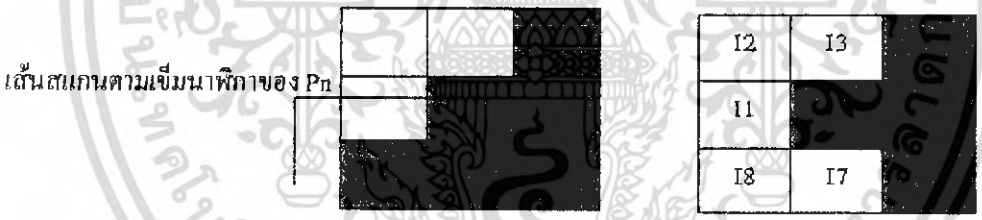
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พิจารณาการติดตามหาจุดภาพสีดำในลำดับถัดไปในทิศทางตามเข็มนาฬิกาที่ตำแหน่ง Pn จะเริ่มพิจารณาจากจุดสแกน(ของเส้นสแกนหลัก)ก่อนที่จะพบจุด Pn เป็นจุดแรก ซึ่งแน่นอนว่าจุดนี้จะเป็นจุดภาพสีขาว และจะพิจารณาจุดถัดไปในทิศตามเข็มนาฬิการอบๆ จุด Pn ถ้าพบจุดใดเป็นจุดสีดำ ก็จะทำการเปลี่ยนจุดที่พิจารณาจุดถัดไปมาเป็นจุดนั้น



รูปที่ 2.9 การติดตามหาจุดภาพสีดำในลำดับถัดไปในทิศทางตามเข็มนาฬิกาที่ตำแหน่ง Pn

ในทำนองเดียวกัน ที่ตำแหน่ง Pn+1 จะเริ่มพิจารณาจากจุดสแกน(ของเส้นสแกนตามเข็มนาฬิกาของ Pn)ก่อนที่จะพบจุด Pn+1 เป็นจุดแรก ซึ่งแน่นอนว่าจุดนี้จะเป็นจุดภาพสีขาว และจะพิจารณาจุดถัดไปในทิศตามเข็มนาฬิการอบๆ จุด Pn+1 ถ้าพบจุดใดเป็นจุดสีดำ ก็จะทำการเปลี่ยนจุดที่พิจารณาจุดถัดไปมาเป็นจุดนั้น



รูปที่ 2.10 การติดตามหาจุดภาพสีดำในลำดับถัดไปในทิศทางตามเข็มนาฬิกาที่ตำแหน่ง Pn+1

เมื่อ I1 คือ ตำแหน่งของภาพที่จะพิจารณาเป็นจุดแรก

I2 คือ ตำแหน่งของภาพที่จะพิจารณาเป็นจุดที่ 2

...

2.3 การหาขนาดความกว้างและความสูงของตัวอักษร

การหาขนาดความกว้างและความสูงของตัวอักษร ทำได้โดยการเก็บค่าพิกัดสูงสุดในแกน X, พิกัดต่ำสุดในแกน X, พิกัดสูงสุดในแกน Y, พิกัดต่ำสุดในแกน Y แล้วนำไปคำนวณหาตามสมการข้างล่างนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$I = Xmax - Xmin + 1 \quad (2.1)$$

$$J = Ymax - Ymin + 1 \quad (2.2)$$

เมื่อ

I คือ ขนาดความกว้างของอักษรแต่ละตัว

J คือ ขนาดความสูงของอักษรแต่ละตัว

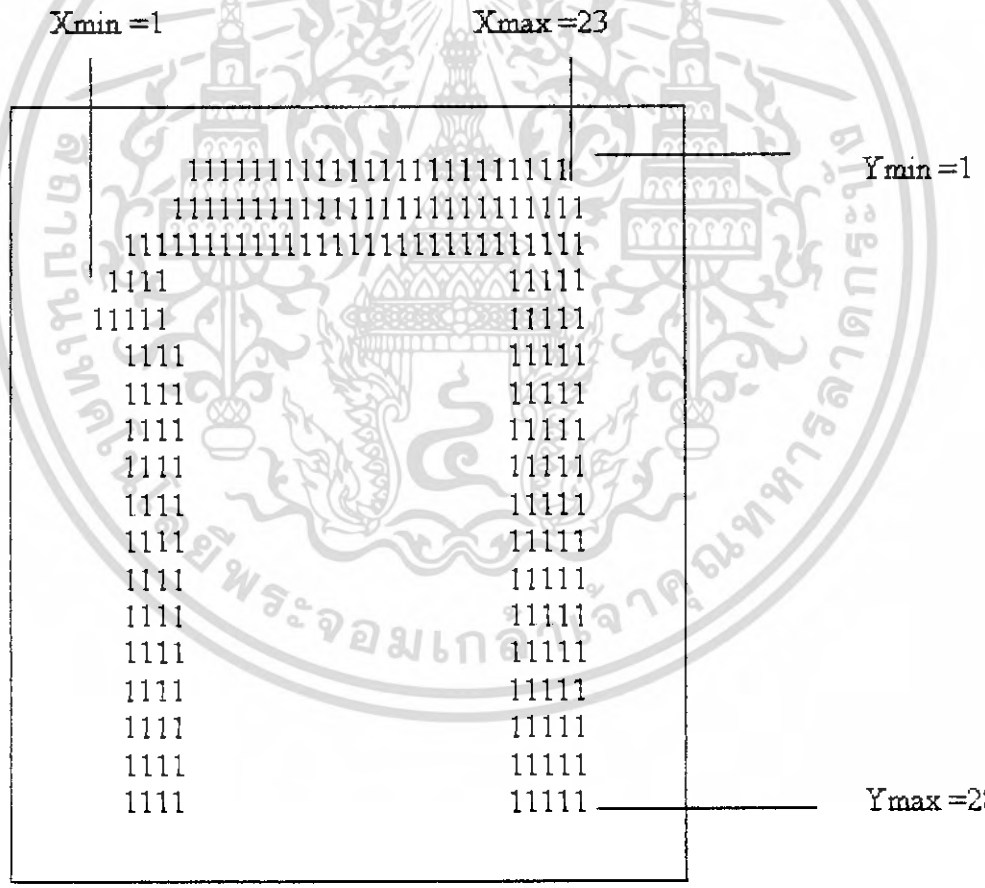
Xmax = ค่าพิกัดสูงสุดในแกน X

Ymax = ค่าพิกัดสูงสุดในแกน Y

Xmin = ค่าพิกัดต่ำสุดในแกน X

Ymin = ค่าพิกัดต่ำสุดในแกน Y

ตัวอย่างการหาค่าขนาดความกว้างความสูงของตัวอักษร แสดงดังรูปที่ 2.11



รูปที่ 2.11 ลักษณะจุดพิกัดของตัวอักษร ที่นำมาคำนวณหาค่าความกว้างและความสูงของตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.11 ได้ค่าจุด Coordinate สูงสุด และต่ำสุดดังนี้

Xmax = 23 Ymax = 28

Xmin = 1 Ymin = 1

จากสมการหาความกว้างและความสูง ทำการแทนค่า

$I = Xmax - Xmin + 1$

$= 23 - 1 + 1$

$= 23$

และ $J = Ymax - Ymin + 1$

$= 28 - 1 + 1$

$= 28$

ดังนั้นภาพอักษรดังรูปที่ 2.11 จึงมีขนาดความกว้างเท่ากับ 28 และความสูงเท่ากับ 23

2.4 การกรองสัญญาณรบกวน (Noise Reduction)

ในการถ่ายภาพ ข้อมูลภาพที่ได้ออกมาทุกครั้งจะมีสัญญาณรบกวน(noise)ปนอยู่ด้วย โดยสัญญาณรบกวนสำหรับงานทางด้านการประมวลผลภาพดิจิตอลในลักษณะนี้จะอยู่ในรูปของฟังก์ชันการกระจาย(Distribution Functions) และส่วนใหญ่เป็นสัญญาณรบกวนที่มีการกระจายแบบเกาส์เซียน ซึ่งเป็นสัญญาณรบกวนที่เกิดขึ้นเนื่องจากปรากฏการณ์ทางธรรมชาติ

การกำจัดสัญญาณรบกวนที่มีการกระจายแบบเกาส์เซียนทำได้โดยการใช้มาส์คแบบ เกาส์เซียนมาเป็นตัวกรองสัญญาณ โดยตัวกรองสัญญาณเกาส์เซียนนี้ จะมีค่าตามสมการข้างล่าง

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \qquad (2.3)$$

โดย σ คือ ค่า standard deviation (constant)

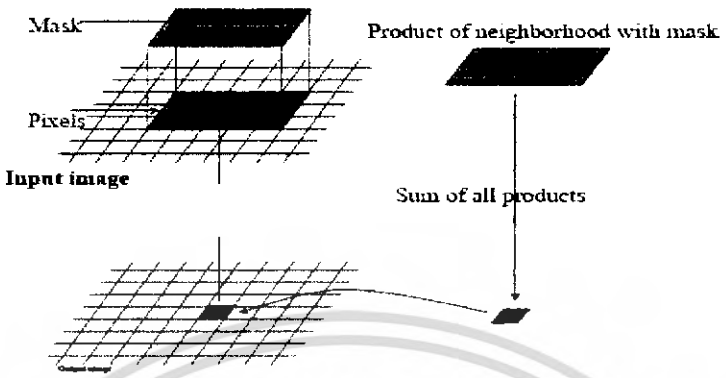
$\frac{1}{273}$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

รูปที่ 2.12 มาส์คแบบเกาส์เซียนฟิลเตอร์(σ =1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราสามารถกรองสัญญาณรบกวนที่มีการกระจายแบบเกาส์เซียนออกได้โดยการนำมาส์คของเกาส์เซียนฟิลเตอร์มาทำคอนโวลูชันกับข้อมูลรูปภาพ



รูปที่ 2.13 การคอนโวลูชันข้อมูลรูปภาพด้วยมาส์ค

2.5 การแปลงเป็นภาพขาวดำ (Thresholding)

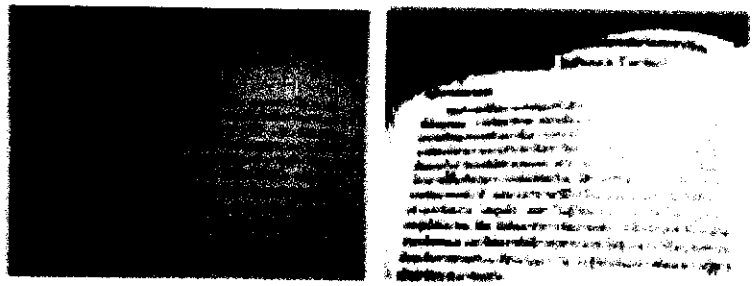
การแปลงภาพเป็นภาพขาดำ คือการแปลงข้อมูลภาพระดับเทา (Gray Scale) ซึ่งมีระดับความเข้ม 256 ระดับ ไปเป็นภาพข้อมูลภาพแบบขาวดำ ซึ่งมีระดับความเข้ม 2 ระดับ โดยจะมีขั้นตอนในการหาค่าเทรสโฮล (Threshold) แล้วจึงจะนำค่าความเข้มแสง (Intensity) ของแต่ละพิกเซล มาเปรียบเทียบกับค่าเทรสโฮลของพิกเซล นั้น ถ้าพิกเซลนั้นมีค่าความเข้มแสง มากกว่าค่าเทรสโฮล ก็จะกำหนดให้พิกเซลนั้นมีค่าเป็นสีดำ แต่ถ้าพิกเซลนั้นมีค่าความเข้มแสง น้อยกว่าค่าเทรสโฮล ก็จะกำหนดให้พิกเซลนั้นมีค่าเป็นสีขาว เราอาจจำแนก ลักษณะการหาค่า เทรสโฮล ได้เป็น 3 แบบ ดังนี้

2.5.1 Manual Threshold คือ การกำหนดค่าเทรสโฮลขึ้นมาโดยผู้ใช้ 1 ค่าสำหรับใช้กับทั้งภาพ โดยเลือกค่าที่ผู้ใช้คิดว่าให้ผลดีที่สุด

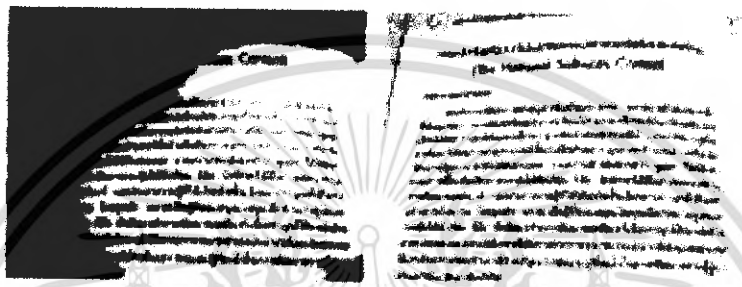
2.5.2 Automatic Threshold คือ การคำนวณหาค่าเทรสโฮลที่เหมาะสมขึ้นมาโดยโปรแกรม 1 ค่าสำหรับใช้กับทั้งภาพ โดยเลือกค่าความเข้มสีที่ทำให้ภาพภายหลังจากการทำเทรสโฮลแล้ว มีพิกเซลสีขาวพอ ๆ กับพิกเซลสีดำ โดยใช้ ฮิสโทแกรมมาช่วย

2.5.3 Adaptive Threshold คือ การคำนวณหาค่าเทรสโฮลที่เหมาะสมขึ้นมาโดยโปรแกรมหลาย ๆ ค่าสำหรับใช้กับทั้งภาพ (คำนวณหาค่าเทรสโฮล 1 ค่าสำหรับใช้กับส่วนย่อยของภาพ) โดยจะพิจารณาจากในขอบเขตที่จำกัด เช่น หน้าต่าง ขนาด 10x10 แล้วใช้ค่ากลางของบริเวณที่พิจารณานั้นเป็นค่าเทรสโฮล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพเริ่มต้น *Threshold Segmentation ที่ Threshold value = 110*

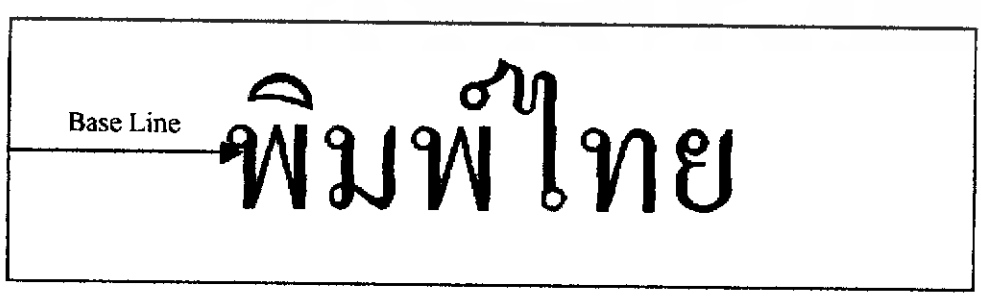


Automatic Threshold *Adaptive Threshold*

รูปที่ 2.14 ผลการทำให้เป็นภาพขาวดำ โดยใช้ค่า เทรส โฮล ต่างๆกัน

2.6 การสแกนเพื่อตัดตัวอักษร (Segmentation)

ในการที่เราจะวิเคราะห์ตัวอักษรได้นั้น เราต้องทำการตัดตัวอักษรออกจากข้อความก่อน แล้วจึงนำอักษรแต่ละตัวนั้นไปผ่านกระบวนการอื่นๆได้ โดยการตัดตัวอักษรออกจากข้อความนั้น เริ่มจากการสแกนตามแนวเส้นแสกน (Base Line) ซึ่งอยู่กึ่งกลางภาพ ในทิศทางจากซ้ายไปขวา เมื่อสแกนพบจุดภาพที่เป็นสีดำ ก็จะทำการติดตามขอบภาพของตัวอักษร และตัดตัวอักษรนั้นออกจากภาพไปแยกไว้ต่างหาก โดยในระหว่างการติดตามขอบภาพนั้น โปรแกรมจะต้องเก็บค่าพิกัดสูงสุดในแกน X และ พิกัดต่ำสุดในแกน X เอาไว้ด้วย



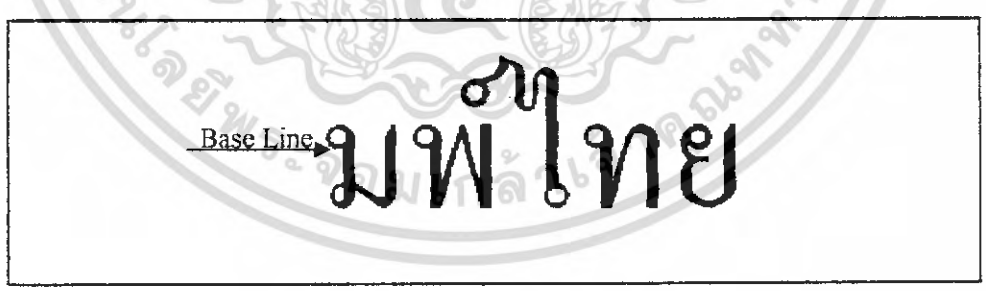
รูปที่ 2.15 การสแกนภาพของแนวเส้นแสกน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อจบขั้นตอนการติดตามขอบภาพที่สแกนพบโดย เส้นแสกน และตัดอักษรออกจากข้อความแล้ว จะต้องมีการสแกนในทิศทางจากล่างขึ้นบนในบริเวณจากพิกัดต่ำสุดในแกน X จนถึงพิกัดสูงสุดในแกน X ของตัวอักษรที่ถูกลบไป เพื่อหาสระบน สระล่าง และวรรณยุกต์ อื่นๆที่อาจมีอยู่ ถ้าพบจุดภาพที่เป็นสีดำ ก็จะทำการติดตามขอบภาพของตัวอักษร และตัดตัวอักษรนั้นออกจากภาพไปแยกไว้ต่างหาก และทำการสแกนต่อจนถึงค่าพิกัดสูงสุดในแกน X ตามที่กำหนดไว้อย่างไม่มีข้อยกเว้น

รูปที่ 2.16 การสแกนภาพเพื่อหาสระบน สระล่าง และวรรณยุกต์อื่นๆ

เมื่อสแกนถึงค่าพิกัดสูงสุดของ X ตามที่กำหนดไว้แล้ว ก้อจะกลับมาสแกนด้วย เส้นแสกนต่อจากจุดเดิมที่พบจุดภาพสีดำจุดแรก ทำเช่นนี้ไปเรื่อยๆจนกระทั่ง เส้นแสกน สแกนไปจนถึงจุดสุดท้ายทางขวาสุดของภาพ ก็ถือว่าสิ้นสุดการสแกนเพื่อตัดตัวอักษร และจะได้ตัวอักษรแต่ละตัวแยกต่างหาก พร้อมสำหรับกระบวนการต่อไป



รูปที่ 2.17 การสแกนภาพของเส้นแสกน ต่อจากจุดเดิม

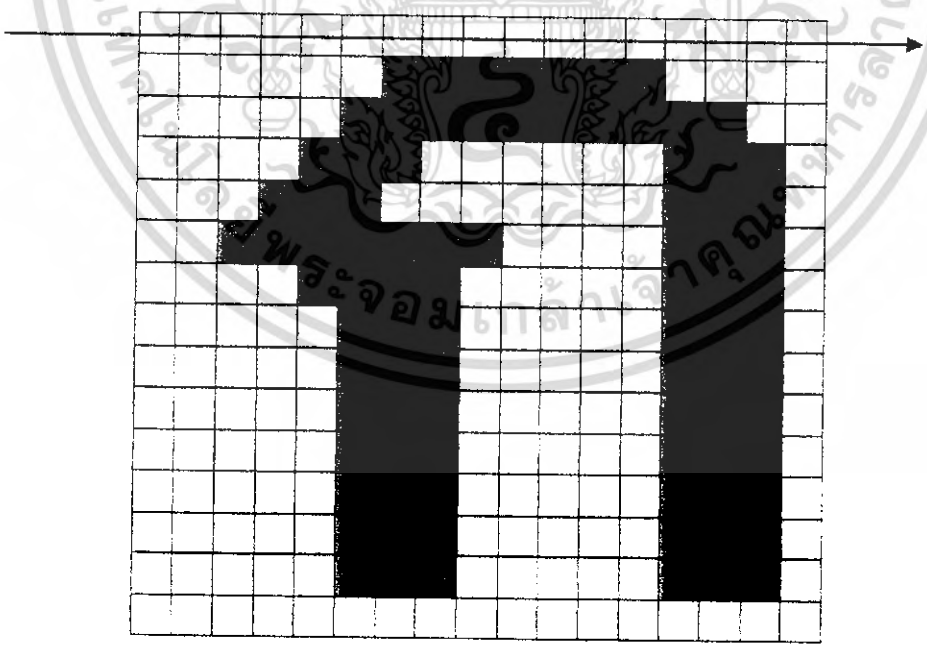
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 การทำลายเส้นตัวอักษรให้บาง (Thinning)

การทำลายเส้นตัวอักษรให้บาง เป็นวิธีการหนึ่งที่ช่วยกำจัดสัญญาณรบกวนเพื่อปรับปรุงรูปแบบของตัวอักษรแต่ละตัวให้ดีขึ้น ทำให้สะดวกในการวัดความสูงและความกว้างตัวอักษร และง่ายต่อการนำไปวิเคราะห์ โดยภาพที่ผ่านกระบวนการนี้แล้วจะเหลือเฉพาะโครงร่าง (หนา 1 พิก-เซล) เท่านั้น

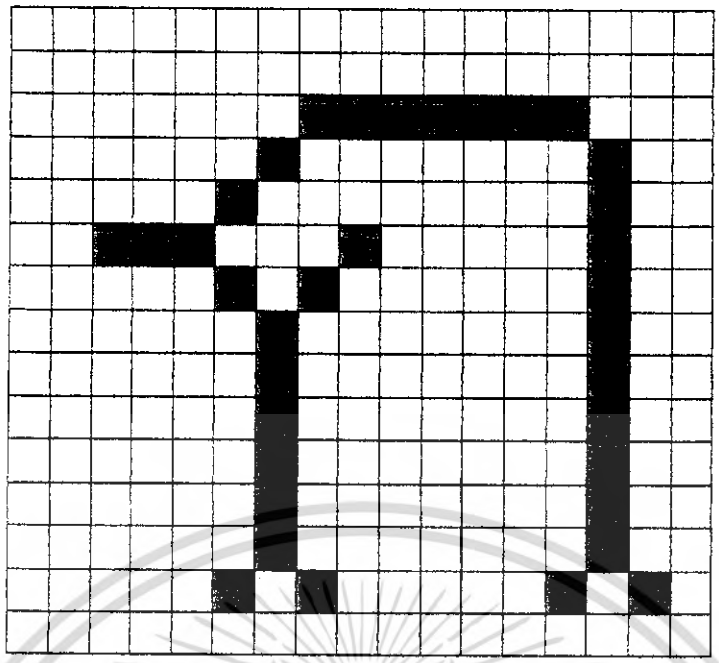
รูปที่ 2.18 ภาพอักษรก่อนทำให้บาง และหลังทำให้บาง

การทำลายเส้นตัวอักษรให้บางนั้นใช้วิธีการตรวจกวาดภาพทั้ง 4 ทิศทาง โดยเมื่อตรวจกวาดพบพิกเซลที่เป็นสีดำพิกเซลแรกของพื้นที่สีดำ ก็จะใช้ หน้าต่าง ขนาด 3 x 3 ในการตรวจสอบเพื่อแยกแยะว่าพิกเซลนั้นมีความหนา 1 จุดภาพหรือไม่ ถ้าใช่ก็จะทำการทำการตรวจกวาดหา พื้นที่สีดำถัดไป แต่ถ้าไม่ใช่จะทำการเปลี่ยนพิกเซลนั้นให้เป็นสีขาวก่อน แล้วจึงทำการตรวจกวาดหาพื้นที่สีดำถัดไป



รูปที่ 2.19 พิกเซลที่จะถูกลบ เมื่อทำการตรวจกวาดในทิศทางจากซ้ายไปขวา 1 ครั้ง

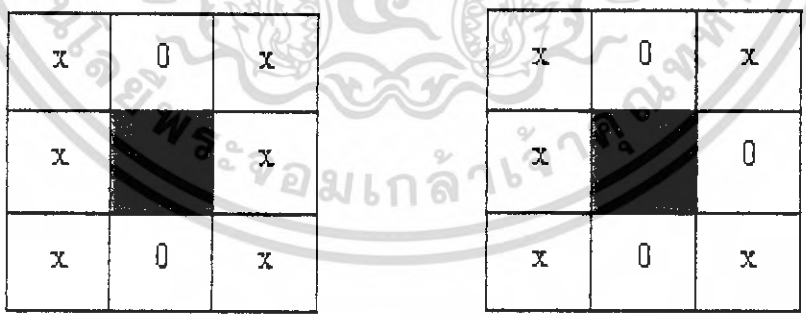
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 ภาพที่ผ่านกระบวนการทำให้บาง จนเหลือความหนาเพียง 1 จุดภาพ

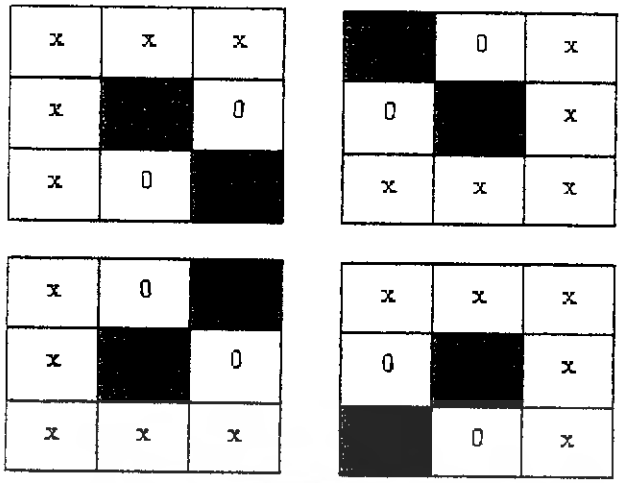
2.8 พิกเซลที่มีความหนา 1 จุดภาพ

พิกเซลที่มีความหนา 1 จุดภาพ คือ พิกเซลที่ไม่สามารถลบทิ้งได้ หรือก็คือพิกเซลที่เป็นโครงร่างของภาพ ได้แก่ พิกเซลที่มีลักษณะดังต่อไปนี้



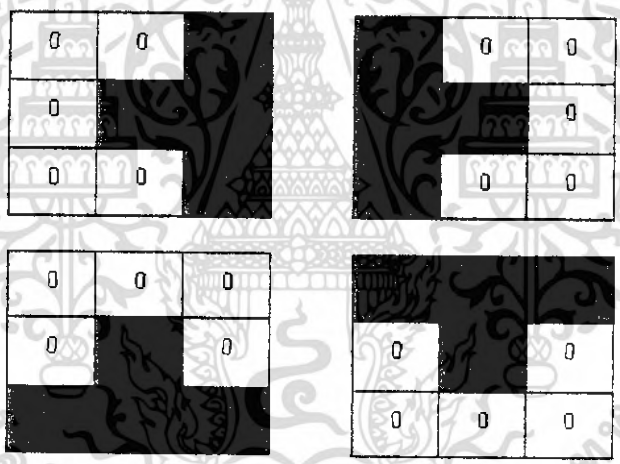
รูปที่ 2.21 พิกเซลที่มีความหนา 1 จุดภาพแบบ ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.22 พิกเซลที่มีความหนา 1 จุดภาพแบบ ข

กรณีพิกเซลที่มีความหนา 1 จุดภาพแบบ ก จะมีข้อยกเว้น ได้แก่ พิกเซลที่มีลักษณะดังต่อไปนี้



รูปที่ 2.23 กรณียกเว้นของพิกเซลที่มีความหนา 1 จุดภาพแบบ ก

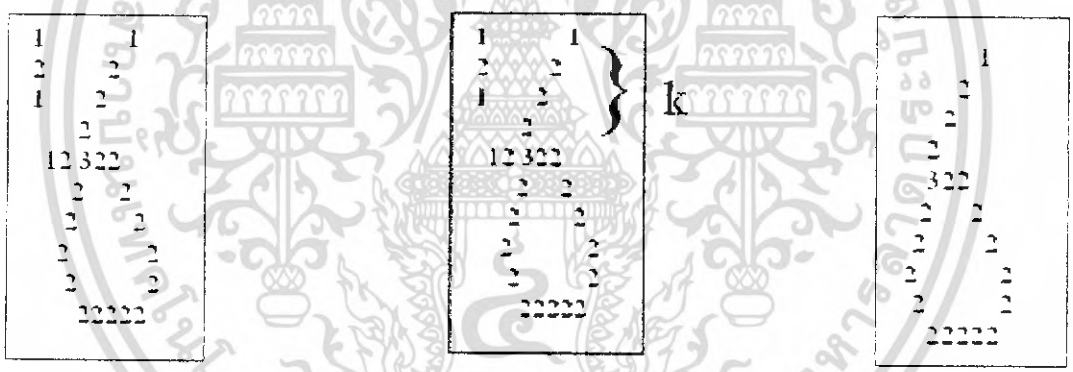
2.9 การกำจัดส่วนเกิน (Enhancing)

ข้อมูลภาพอักษรที่ผ่านการทำให้บางแล้วจะเหลือเฉพาะโครงร่าง ซึ่งรวมทั้งส่วนที่เกินที่ไม่ต้องการด้วย ในการกำจัดส่วนเกินจะใช้คุณสมบัติทางโทโปโลยีของจุดภาพ และใช้วิธีการติดตามจุดภาพมาช่วยในการวิเคราะห์ เมื่อทำการกำจัดส่วนเกินแล้วข้อมูลภาพอักษรที่ได้จะมีความสมบูรณ์พอที่จะนำไปวิเคราะห์เพื่อรู้จำตัวอักษรต่อไป

72218
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.24 ภาพอักษรก่อนการกำจัดส่วนเกิน และหลังการกำจัดส่วนเกิน

การกำจัดส่วนเกินจะเริ่มจากการสแกนหาจุดภาพที่มีคุณสมบัติทางโทโปโลยีของจุดภาพเป็นจุดปลาย จากนั้นจะทำการติดตามจุดภาพถัดไปซึ่งมีคุณสมบัติทางโทโปโลยีของจุดภาพเป็นจุดต่อไปเรื่อยๆพร้อมทั้งทำการนับค่าพิกเซล ที่ได้ทำการติดตามมา จนกระทั่งถึงจุดปลาย, จุดแยก หรือจุดตัด ถ้าค่าพิกเซลที่นับได้เกิน k จุด (ค่าที่กำหนดให้เป็นส่วนเกิน) แสดงว่าพิกเซลที่ติดตามมาทั้งหมดนี้ไม่ใช่ส่วนเกิน ในทางตรงกันข้ามถ้าค่าพิกเซล ที่นับได้ยังไม่เกิน k จุด แสดงว่าพิกเซลที่ติดตามมาทั้งหมดนี้เป็นส่วนเกิน จะทำการเปลี่ยนค่าพิกเซลที่ติดตามมาทั้งหมดให้เป็นสีขาว(ลบจุดภาพ)



รูปที่ 2.25 การทำงานของการกำจัดส่วนเกิน

เมื่อ

1 = จุดปลาย

2 = จุดต่อ

3 = จุดตัด

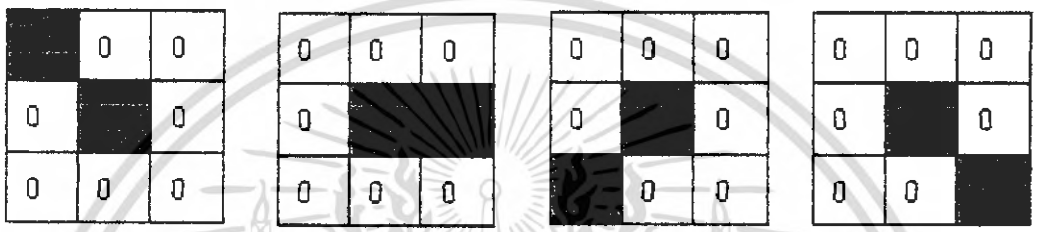
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 การวิเคราะห์หารหัสแทนตัวอักษร (ดิวโค้ด)

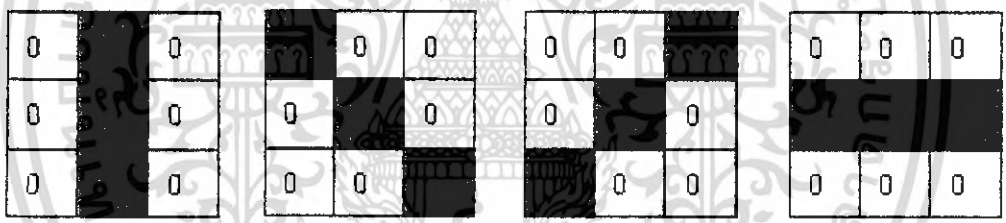
การวิเคราะห์หารหัส ดิวโค้ด ของตัวอักษรนั้น จะใช้วิธีแบ่งส่วนของตัวอักษรเพื่อพิจารณาคุณสมบัติทางโทโปโลยีของจุดภาพในแต่ละส่วน

ภาพตัวอักษรที่จะนำมาใช้วิเคราะห์หารหัส ดิวโค้ด นั้น เป็นภาพที่ผ่านการทำให้บาง และการกำจัดส่วนเกินมาแล้ว ซึ่งจะคงเหลือคุณสมบัติทางโทโปโลยีของจุดภาพเพียง 4 แบบ คือ จุดปลาย(End พิกเซล), จุดต่อ(Connect พิกเซล), จุดแยก(Branch พิกเซล) และ จุดตัด(Cross พิกเซล)

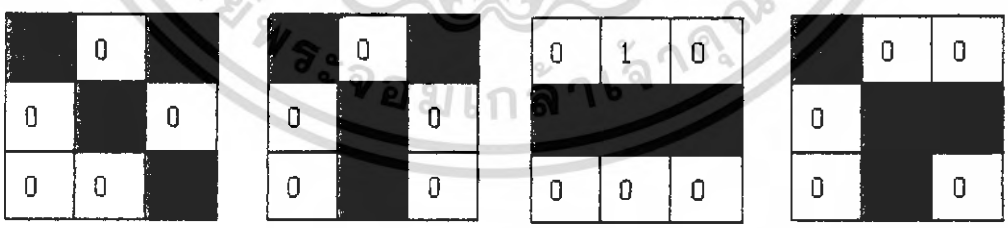
จุดปลาย



จุดต่อ

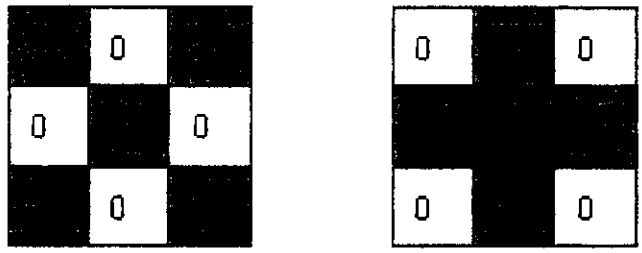


จุดแยก



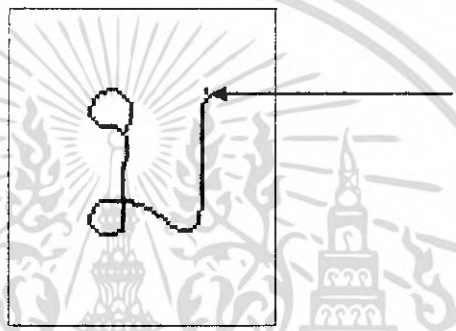
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จุดตัด

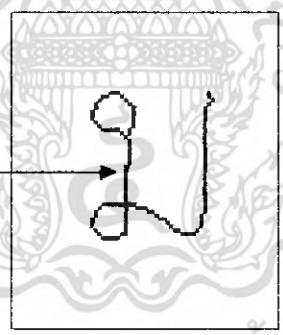


รูปที่ 2.26 ตัวอย่างของจุดภาพที่จะนำมาใช้วิเคราะห์หารหัสคิวโค้ด

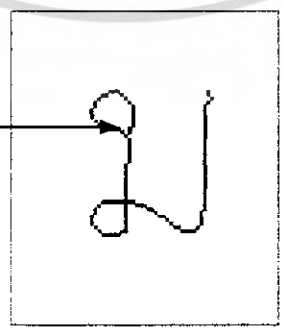
จุดปลาย



จุดต่อ

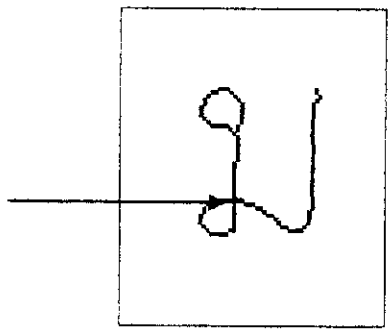


จุดแยก



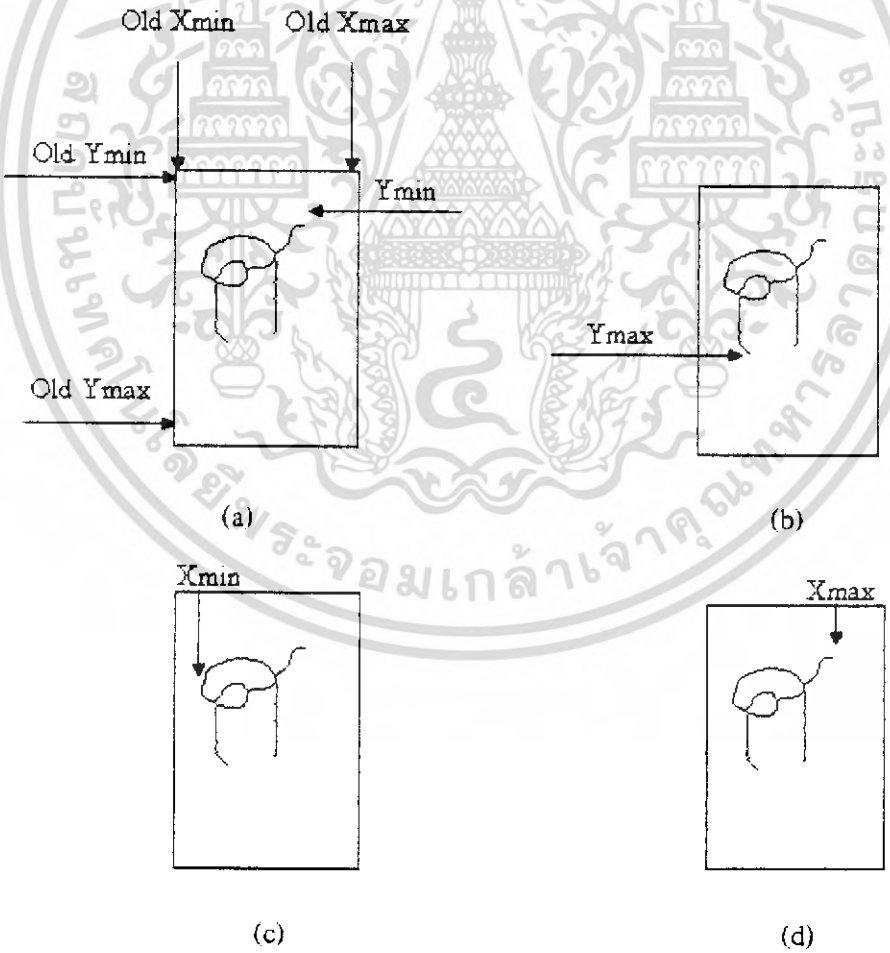
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จุดตัด



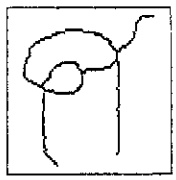
รูปที่ 2.27 ตัวอย่างลักษณะของคุณสมบัติโทโปโลยีในภาพอักษร

ก่อนการแบ่งส่วนภาพที่จะนำมาวิเคราะห์ จะต้องทำการปรับกรอบภาพให้มีขนาดพอดีกับตัวอักษรเสียก่อน โดยกำหนดให้กรอบใหม่มีขนาดใหญ่กว่าพิกัดสูงสุดในแกน X, พิกัดต่ำสุดในแกน X, พิกัดสูงสุดในแกน Y และ พิกัดต่ำสุดในแกน Y อยู่ข้างละ 1 พิกเซล



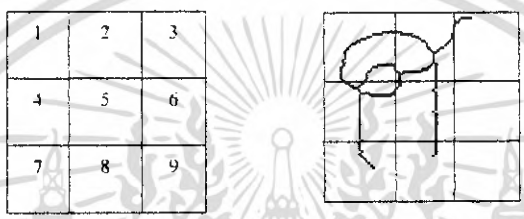
รูปที่ 2.28 การตรวจสอบภาพเพื่อปรับกรอบภาพใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



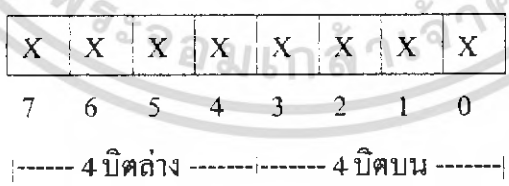
รูปที่ 2.29 ภาพตัวอักษรที่ปรับกรอบภาพใหม่แล้ว

ในโครงงานนี้จะแบ่งส่วนภาพที่จะนำมาวิเคราะห์ซึ่งผ่านการปรับกรอบภาพใหม่แล้ว ออกเป็น 9 ส่วนด้วยกัน



รูปที่ 2.30 การแบ่งภาพอักษรออกเป็น 9 ส่วน

โดยภายในแต่ละส่วนจะทำการพิจารณาเพื่อหาคุณสมบัติทางโทโปโลยีของจุดภาพแล้วนำมากำหนดเป็นรหัสขนาด 4 บิต จะเห็นว่าหน่วยความจำขนาด 1 ไบต์ จะสามารถเก็บข้อมูลคุณสมบัติทางโทโปโลยีของจุดภาพได้ 2 ส่วน ดังนั้นตัวอักษรแต่ละตัวซึ่งถูกแบ่งเป็น 9 ส่วนนั้นจะต้องใช้หน่วยความจำในการเก็บข้อมูลขนาด 5 ไบต์ โดยจะกำหนดให้ 4 บิต บนของ ไบต์สุดท้ายมีค่าเป็น 0 ทั้งหมด



รูปที่ 2.31 การเก็บข้อมูลคุณสมบัติทางโทโปโลยีในหน่วยความจำขนาด 1 ไบต์(X = 0, 1)

เมื่อ บิตที่ 0 เก็บค่าคุณสมบัติของจุดปลายของส่วนที่ 2

บิตที่ 1 เก็บค่าคุณสมบัติของจุดต่อของส่วนที่ 2

บิตที่ 2 เก็บค่าคุณสมบัติของจุดแยกของส่วนที่ 2

บิตที่ 3 เก็บค่าคุณสมบัติของจุดตัดของส่วนที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิตที่ 4 เก็บค่าคุณสมบัติของจุดปลายของส่วนที่ 1
บิตที่ 5 เก็บค่าคุณสมบัติของจุดต่อของส่วนที่ 1
บิตที่ 6 เก็บค่าคุณสมบัติของจุดแยกของส่วนที่ 1
บิตที่ 7 เก็บค่าคุณสมบัติของจุดตัดของส่วนที่ 1

2.11 การหาเส้นบรรทัด(Line Segment)

ในการวิเคราะห์ตัวอักษรในภาพใดๆนั้น จะทำการวิเคราะห์ตัวอักษรเพียงบรรทัดเดียว คือ บรรทัดที่เส้น เส้นแสกน ลากผ่านเท่านั้น ส่วนบรรทัดอื่นๆ ที่อาจถ่ายติดมาในภาพ โปรแกรมจะทำการตัดทิ้งไปไม่นำมาพิจารณาร่วมด้วย

หลักการหาเส้นบรรทัด ทำได้โดยการสแกนในแนวขนานกับเส้นแสกน และค่อยๆเว้นระยะห่างจากเส้นแสกน ออกมาเรื่อยๆจนพบเส้นสแกนที่มีจำนวนพิกเซลสีดำลดลงจาก เส้นแสกนผิดปกติอย่างเห็นได้ชัด ก็จะถือว่าเส้นนั้นเป็นเส้นบรรทัดหลักที่ต้องการ หลังจากพบเส้นบรรทัดหลักทั้งด้านบน และด้านล่าง เส้นแสกน แล้ว จะทำการตัดเป็นบรรทัดหลักรวมทั้งสแกนขึ้นมาด้านบน และลงไปด้านล่างของเส้นบรรทัดหลัก เป็นระยะเท่ากับความสูงของเส้นบรรทัดหลักนั้น

รูปที่ 2.32 การหาเส้นบรรทัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12 การรู้จำตัวอักษร

การรู้จำตัวอักษรนั้น ทำได้โดยการเปรียบเทียบรหัสคิวโค้ด ของภาพอักษรที่วิเคราะห์ได้กับรหัส คิวโค้ด ของตัวอักษรที่ทำการเก็บข้อมูลไว้ โดยจะวิเคราะห์หาข้อมูลที่มีความคล้ายคลึงกับภาพตัวอักษรมากที่สุด

จะเห็นได้ว่าการวิเคราะห์รหัส คิวโค้ด นั้นสามารถสร้างความแตกต่างได้ถึง $2^4 = 16$ สถานะต่อหนึ่งส่วน หรือ $16^9 = 68719476736$ สถานะ ต่อหนึ่งตัวอักษร ดังนั้นจึงพบว่าตัวอักษรหนึ่งๆนั้นไม่จำเป็นต้องเก็บรหัส คิวโค้ด ไว้เพียงรหัสเดียว ซึ่งการเก็บรหัส คิวโค้ด ไว้หลายๆรหัส สำหรับ 1 ตัวอักษรนั้น ก็จะทำให้การรู้จำตัวอักษรมีความถูกต้องมากขึ้นด้วย ในกรณีที่ตัวอักษรมีรูปแบบต่างกัน

ในทางตรงกันข้าม ตัวอักษรบางตัวกลับมีรหัสคิวโค้ด ที่ถูกต้องเหมือนกัน กรณีเช่นนี้ เราจะต้องใช้คุณสมบัติอื่นๆมาช่วยในการรู้จำตัวอักษร ซึ่งจะกล่าวถึงในลำดับต่อไป

คุณสมบัติของสัดส่วนตัวอักษร

จะทำการหาคุณสมบัติพิเศษของตัวอักษรเกี่ยวกับสัดส่วนความกว้างต่อความสูง โดยแยกตัวอักษรออกเป็น 2 กลุ่ม คือ ตัวอักษรที่มีสัดส่วนปกติ และตัวอักษรที่มีสัดส่วนแบบพิเศษ กำหนดสมการในการพิจารณาคุณสมบัติของสัดส่วนตัวอักษรดังนี้

$$X = (W \times 1.8) - H \qquad (2.4)$$

เมื่อ X = ค่าที่ใช้ตัดสินใจ

W = ค่าความกว้างของตัวอักษร

H = ค่าความสูงของตัวอักษร

โดยถ้าค่า X มากกว่า 0 แสดงว่าเป็นตัวอักษรที่มีสัดส่วนปกติ แต่ถ้าค่า X น้อยกว่า 0 แสดงว่าเป็นตัวอักษรที่มีสัดส่วนแบบพิเศษ

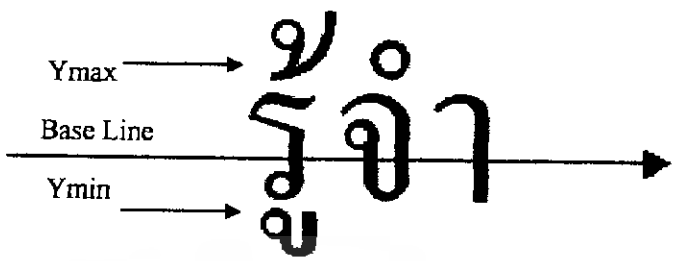
คุณสมบัติเกี่ยวกับตำแหน่งบนเส้นบรรทัดของตัวอักษร

คุณสมบัติเกี่ยวกับตำแหน่งบนเส้นบรรทัดของตัวอักษรนี้ จะพิจารณาโดยเทียบกับเส้นแสกนเป็นหลัก โดยแยกตัวอักษรออกเป็น 3 กลุ่ม ตัวอักษรกลุ่มบน ตัวอักษรกลุ่มกลาง และตัวอักษรกลุ่มล่าง ซึ่งตัวอักษรแต่ละกลุ่ม มีคุณสมบัติดังนี้

1. ตัวอักษรกลุ่มบน คือ กลุ่มของตัวอักษรที่มีค่าพิกัดสูงสุดในแกน Y ต่ำกว่าค่า เส้นแสกน
2. ตัวอักษรกลุ่มล่าง คือ กลุ่มของตัวอักษรที่มีค่าพิกัดต่ำสุดในแกน Y สูงกว่าค่า เส้นแสกน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ตัวอักษรกลุ่มกลาง คือ ตัวอักษรกลุ่มที่นอกเหนือจาก ตัวอักษรกลุ่มบนและกลุ่มล่าง ทั้งหมด



รูปที่ 2.33 ค่าต่างๆ ที่ต้องพิจารณาในการหาคุณสมบัติเกี่ยวกับตำแหน่งบนเส้นบรรทัดของตัวอักษร

2.13 J2ME

J2ME หรือ JAVA 2 Micro Edition คือ เทคโนโลยีหนึ่งของตระกูลจาวา (java) ซึ่งเป็นทั้งสภาวะแวดล้อมของการพัฒนาและของตัวรันไทม์ (Run Time) ไปพร้อม ๆ กัน ซึ่งเป้าหมายของ J2ME คือการนำซอร์ฟแวร์ของจาวา เข้าไปรัน และทำงานอยู่บนอุปกรณ์อิเล็กทรอนิกส์อย่างเช่น โทรศัพท์มือถือ เป็นต้น ซึ่งอุปกรณ์เหล่านี้มีขนาดเล็ก หน่วยความจำน้อย และมีขีดความสามารถในการประมวลผลต่ำ

สถาปัตยกรรมของ J2ME

สถาปัตยกรรมของ J2ME แบ่งออกเป็น 2 ส่วน คือ คอนฟิกูเรชั่น (Configuration) และ โพรไฟล์ (Profiles) ถือเป็นส่วนหลักสำคัญของ J2ME ซึ่งจุดประสงค์ของการกำหนดสองส่วนนี้ก็เพื่อที่จะให้ได้เวอร์ชวลแมทชีน (virtual machines) และคลาสไลบรารี (libraries) ที่เหมาะสมสำหรับอุปกรณ์ แต่ละกลุ่มประเภท

1. คอนฟิกูเรชั่น (Configuration)

เป็นกลุ่มของ API ระดับต่ำ (Low-level API) ทำหน้าที่เป็นตัวกำหนดคุณลักษณะในระหว่างการรันแอพพลิเคชั่นปัจจุบันได้แบ่ง คอนฟิกูเรชั่น สำหรับ J2ME ออกเป็น 2 ประเภทคือ CDC (Connected Device Configuration) และ CLDC (Connected Limited Device Configuration) โดยการจัดกลุ่มจะแบ่งตาม อุปกรณ์ ซึ่งมีความคล้ายกันในเรื่องของขนาดของหน่วยความจำ และกำลังการประมวล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• CLDC (Connected Limit Device Configuration)

เป็น คอนฟิกูเรชั่น สำหรับอุปกรณ์ที่มีหน่วยความจำน้อย มีกำลังของหน่วยประมวลผลต่ำ กินไฟน้อยและมีข้อจำกัดในด้านการเชื่อมต่อเครือข่าย ซึ่ง คอนฟิกูเรชั่น CLDC นี้จะใช้ กิโลไบต์ เวอร์ชวลแมทชีน หรือ KVM เป็น เวอร์ชวลแมทชีน ซึ่งอุปกรณ์ที่จะใช้ คอนฟิกูเรชั่น นี้ได้แก่ โทรศัพท์มือถือ , พีดีเอ เป็นต้น

CLDC มีข้อจำกัดในเรื่องของ ไม่รองรับกับ ข้อมูลทศนิยมชนิดโฟลท (Floating-Point) และข้อมูลทศนิยมชนิดดับเบิ้ล (Double), ไม่สามารถเรียกใช้เมธอดไฟนอล (Finalizion Method) และมีข้อจำกัดในเรื่อง การจัดการข้อผิดพลาด ที่เป็นอย่างนี้ เพราะว่า CLDC จะต้องมีการทำงานที่อยู่บนเครื่องหรืออุปกรณ์ ที่มีข้อจำกัดทางด้านหน่วยความจำ และ หน่วยประมวลผล จึงจำเป็นจะต้องตัดในส่วน ที่ต้องใช้งานหน่วยความจำ มากๆ ออกไป

• CDC (Connect Device Configuration)

เป็น คอนฟิกูเรชั่น สำหรับอุปกรณ์ที่มีความสามารถสูงกว่า CLDC โดยสนับสนุนอุปกรณ์ที่มีหน่วยความจำตั้งแต่ 512 กิโลไบต์จนถึง 2 เมกะไบต์ โดยใช้ ซีเวอร์ชวลแมทชีน หรือ CVM เป็นเวอร์ชวลแมทชีน ตัวอย่างอุปกรณ์ที่ใช้ คอนฟิกูเรชั่น นี้ได้แก่ พอกเก็ตพีซี (Pocket PC) และอุปกรณ์จำพวก เซ็ททอปบ๊อกซ์ (set-top box) ของทีวี เป็นต้น

2. โพรไฟล์

เป็นข้อกำหนดทางด้านคุณลักษณะของอุปกรณ์ที่จะรัน J2ME ซึ่งอาจจะเป็นโพรไฟล์สำหรับอุปกรณ์หนึ่งชิ้นโดยเฉพาะ หรือเป็นโพรไฟล์ที่ทำงานคล้ายๆกันก็ได้ ตัวอย่างข้อกำหนดของโพรไฟล์ เช่น อุปกรณ์มีส่วนที่ติดต่อกับผู้ใช้อย่างไร (ในการรับและแสดงข้อมูล) , อุปกรณ์จะติดต่อกับเครือข่ายอย่างไรหรือเก็บข้อมูลไว้อย่างไร เป็นต้น ดังนั้นในการพัฒนาแอพพลิเคชั่น J2ME จำเป็นจะต้องเลือกโพรไฟล์อย่างน้อย 1 ตัว เพื่อเป็นข้อกำหนดแอพพลิเคชั่นที่พัฒนาขึ้น ว่าสามารถจะนำไปรันกับอุปกรณ์ใดได้บ้าง โพรไฟล์ที่ได้รับความนิยมมากที่สุด คือ MIDP (Mobile Information Device Profile)

เวอร์ชวลแมทชีน ของ J2ME

จาวาเวอร์ชวลแมทชีน เป็นตัวกลางที่ภาษาจาวานำมาใช้เพื่อทำการแปลโปรแกรมแล้วส่งต่อไปให้ระบบปฏิบัติการจากนั้นก็จะทำแปลงคลาสให้เป็นภาษาเครื่องและทำงานต่อไป โดยโปรแกรมภาษาจาวาทุกตัวจะต้องทำงานอยู่ในภายใต้ จาวาเวอร์ชวลแมทชีน ซึ่งมีอยู่ในตัวไม่ว่าจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็น แอพเพลท(Applet), เซิร์ฟเลท (Servlet) และอื่น ๆ นอกจากนี้แล้ว จาวาเวอร์ชวลแมทชีน ยังมีหน้าที่ความรับผิดชอบเตรียมความพร้อมสำหรับความปลอดภัยอีกด้วย

สำหรับ CDC (Connect Device คอนฟิกูเรชั่น) มีการกำหนดให้สามารถใช้ เวอร์ชวลแมทชีน ชุดเดียวกันกับ J2SE ได้ แต่ CLDC (Connected Limit Device คอนฟิกูเรชั่น) ไม่ได้กำหนดให้ใช้กลุ่มคลาสเดียวกันทั้งหมด ทางบริษัทซัน จึงได้พัฒนาเครื่องมือสำหรับสำหรับ เวอร์ชวลแมทชีน ขึ้นมาใหม่เรียกว่า เคเวอร์ชวลแมทชีน หรืออาจจะเรียกอีกอย่างหนึ่งว่า KVM สำหรับ เวอร์ชวล-แมทชีน นี้ได้รับการออกแบบพัฒนาเป็นพิเศษเหมาะสำหรับอุปกรณ์ประเภทมือถือ โดย KVM เป็น จาวาเวอร์ชวลแมทชีน ขนาดเล็กที่พัฒนา จาวาเวอร์ชวลแมทชีน ขึ้นมาใหม่ ซึ่งมีคุณสมบัติดังนี้

- เวอร์ชวลแมทชีน ต้องการหน่วยความจำ 40 และ 80 กิโลไบต์เท่านั้น
- ต้องการหน่วยความจำแบบไดนามิค (Dynamic) 20-40 กิโลไบต์ เท่านั้น
- สามารถรันบน 16 บิต โดยมีความเร็วในการประมวลผล 25 เมกะเฮิรตซ์

การพัฒนาแอพพลิเคชั่นด้วยจาวานั้น สำหรับ J2ME แล้วแอพพลิเคชั่นที่ได้ออกมาจะเรียกว่า มิดเลท (MIDlet) ซึ่งจะนำไปรันบนอุปกรณ์ต่างๆที่สนับสนุนเทคโนโลยี J2ME ซึ่ง มิดเลทที่พัฒนาขึ้นจะประกอบด้วยไฟล์ 2 ไฟล์ คือไฟล์นามสกุล .jar และไฟล์นามสกุล .jad ซึ่งไฟล์นามสกุล .jar นั้นเป็นที่เก็บคลาสไฟล์ต่าง ๆ (นามสกุล .class) ของ มิดเลท นั้น ส่วนไฟล์ .jad (Java Description) เป็นไฟล์ข้อความธรรมดาที่ใช้บรรยายหรืออธิบายไฟล์ .jar ว่ามีชื่ออะไร หรือมีขนาดไฟล์เท่าไหร่ เป็นต้น มิดเลทยังเป็นคลาสที่อยู่ในแพคเก็จ ของ javax.microedition.midlet และเป็นโปรแกรมที่พัฒนาขึ้นจาก MIDP และสืบทอดจากมิดเลท

MIDlet

มิดเลท คือ โปรแกรมจาวา ที่รันบนสภาวะแวดล้อมของ MIDP หรือ เป็นคลาสที่อยู่ในแพ็คเกจของ javax.microedition.midlet หรือโปรแกรมที่พัฒนาขึ้นจาก MIDP และ สืบทอดจาก มิดเลท หรือ คลาสที่สืบทอดจากมิดเลทคลาส และ เรียกใช้ (Implements) 3 เมธอด (method) คือ startApp(), pauseApp() และ destroyApp()

โปรแกรมประยุกต์ ที่เขียนขึ้นมาเพื่อใช้กับอุปกรณ์ที่เป็น MIDP จะเรียกว่า มิดเลท ซึ่งมีลักษณะเหมือนกับ จาวาแอพเพลท ที่เขียนจาวาบนเว็บ ใน มิดเลท จะต้องสืบทอดคลาส javax.microedition.midlet.MIDlet และ เรียกใช้ 3 เมธอด คือ startApp(), pauseApp() และ destroyApp()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MIDlet's Life Cycle

เมื่อได้สร้างคลาสที่สืบทอดมาจากมิดเลทแพคเกจที่ชื่อ javax.microedition.midlet.MIDlet แล้ว การทำงานของคลาสจะเริ่มจากการสร้าง หลังจากนั้นจะอยู่ในสถานะหยุด (Paused) และเมื่อต้องการให้มีการทำงานเกิดขึ้น ก็จะต้องเรียกใช้เมธอด startApp() จะทำให้สถานะของคลาสก็จะเปลี่ยนเป็นสถานะทำงาน (Active) และหากต้องการให้คลาสหยุดการทำงาน ก็สามารถทำได้โดยการเรียกใช้เมธอด pauseApp() หากต้องการจบการทำงาน ก็ให้เรียกใช้เมธอด destroyApp() โปรแกรมก็จะเข้าสู่สถานะ ถูกทำลาย (Destroyed) สถานะต่าง ๆ ของ มิดเลท มีอยู่ 3 สถานะ คือ

• Paused State: สถานะหยุดการทำงาน จะเกิดขึ้นเมื่อทำการเรียกใช้เมธอด pauseApp() หรือ notifyPaused()

• Active State : สถานะการทำงาน จะเกิดขึ้นเมื่อทำการเรียกใช้เมธอด startApp()หรือ มีการเรียกใช้ method resumeRequest() ในตอนที่สถานะการทำงานยังเป็น pauseApp()

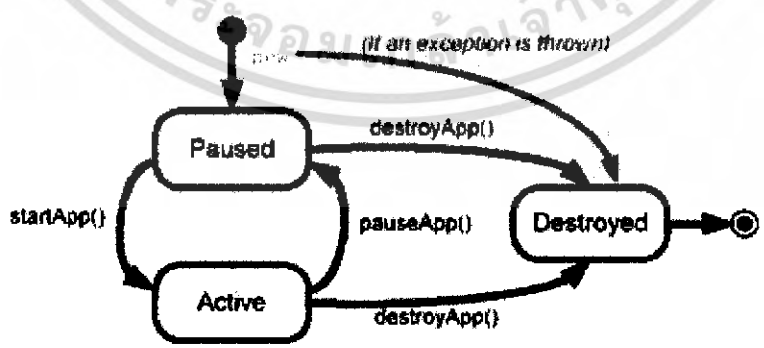
• Destroyed State: สถานะถูกทำลาย จะเกิดขึ้นหลังจากที่มีการเรียกใช้เมธอด destroyApp()หรือ notifyDestroy()

ในการเปลี่ยนสถานะ (state) ของการทำงาน จะใช้ method 3 อันด้วยกัน คือ

- resumeRequest() ใช้เพื่อ ให้กลับมาอยู่ในสถานะทำงาน อีกครั้ง

- notifyPaused() ใช้เพื่อ สั่งหยุดการทำงาน

- notifyDestroy() ใช้เพื่อ สั่งทำลาย เพื่อจบการทำงาน



รูปที่ 2.34 วงจรการทำงานของมิดเลท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MIDlet suites

ชุดมิดเลท (MIDlet suites) เป็นที่รวมของคลาสของมิดเลท และ ไฟล์รูปภาพต่างๆ ที่เกี่ยวข้องกับการทำงาน ชุดมิดเลทจะประกอบด้วย 2 ไฟล์หลักๆ คือ

1. ไฟล์ .jad เป็น Text file ที่ทำหน้าที่ บอกลักษณะ และคุณสมบัติของโปรแกรม (Application Descriptor) จะเก็บ รายละเอียดของ ชุดมิดเลทต่างๆ ใช้สำหรับให้อุปกรณ์ (เช่น โทรศัพท์มือถือ) ดาว์นโหลด ไปทำการตรวจสอบรายละเอียดของโปรแกรม ก่อนการติดตั้ง

ไฟล์ .jad จะประกอบด้วย 7 แอตทริบิวต์ (attribute) หลักคือ

MIDlet-Name :

MIDlet-Version :

MIDlet-Vendor :

MIDlet-Jar-URL :

MIDlet-Jar-Size :

MicroEdition-Profile :

MicroEdition-คอนฟิกูเรชั่น :

ส่วนที่ละเว้นได้ (Optional) ได้แก่

MIDlet-Description :

MIDlet-Icon :

MIDlet-Info-URL :

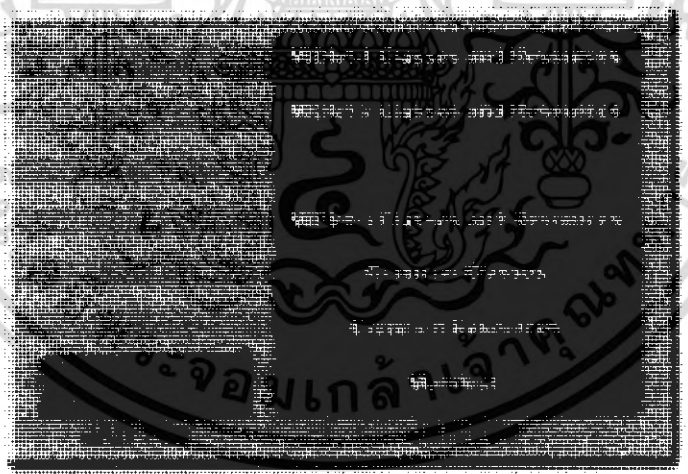
MIDlet-Date-Size :

ก่อนที่ ชุดมิดเลทจะถูกดาว์นโหลดลงอุปกรณ์ จะมีตัวที่เป็น ซอฟต์แวร์ที่คอยจัดการโปรแกรมประยุกต์ (Application Management Software) คอยตรวจสอบค่าแอตทริบิวต์ ใน ไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

.jad 7 ค่าแรก เพื่อดูว่าเหมาะสมกับอุปกรณ์ นั้นหรือเปล่า ตัวอย่างเช่น ถ้าอุปกรณ์ ไม่รองรับเวอร์ชัน ของ โพรไฟล์ ซึ่งอยู่ในแอตทริบิวต์ "MicroEdition-Profile" ตัว ไฟล์ .jar ก็จะไม่ถูกดาว์นโหลด หรือ ถ้าขนาดของ ไฟล์ .jar ที่อยู่ใน MIDlet-Jar-Size ใหญ่เกินกว่าที่อุปกรณ์ จะรองรับได้ก็จะไม่ถูกดาว์นโหลด มา

2. ไฟล์ .jar เป็นไฟล์ที่เก็บมิดเลทต่าง ๆ เอาไว้ โดยปกติ ไฟล์ .jar ของ ชุดมิดเลทจะประกอบด้วยคลาสไฟล์ ทั้งหมดของโปรแกรมประยุกต์ มิดเลท รวมไปถึง รีซอร์สไฟล์ (resource file) และ แมนิเฟสไฟล์ (manifest file) โดยที่ คลาสไฟล์ จะต้องถูกตรวจสอบความถูกต้อง (Preverify) ก่อน ส่วน รีซอร์สไฟล์ มักประกอบด้วย ไฟล์ข้อความ , ไฟล์รูปภาพ ที่ถูกใช้โดย มิดเลท ในตอน ประมวลผล แต่ละ มิดเลท ที่อยู่ใน ชุดมิดเลทเดียวกันสามารถใช้ คลาสไฟล์ ร่วมกันได้ ถ้าเราสร้าง มิดเลท หลาย ๆ ตัวโดยทำเป็น แพ็คเกจ เดียวกันก็จะประหยัดขนาดแพ็คเกจ ที่ต้องดาว์นโหลดได้มาก เพราะว่าสามารถใช้วิธีการใช้ไฟล์ร่วมกัน ได้ ถ้าเราต้องใช้ Third party class library ภายใน มิดเลท เช่น ไลบรารี สำหรับการคำนวณ ทศนิยมเป็นต้น เราก็จะต้อง รวม (include) ตัว Third party class file ลงใน ไฟล์ .jar ด้วย ถ้าหากว่าอุปกรณ์ ที่ใช้งานไม่ได้ถูกโหลด ตัว คลาสไฟล์ นี้ไว้ในอุปกรณ์ อยู่แล้วจากผู้ผลิต



รูปที่ 2.35 องค์ประกอบของชุดมิดเลท

สามารถ แพ็คเกจ มิดเลท ลงใน ไฟล์ .jar โดยใช้คำสั่งดังนี้ :

```
jar cvmf MANIFEST.MF HelloMidlet.jar HelloMidlet.class icon.pn
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมประยุกต์มิดเลท เมื่อถูกสร้างขึ้นจะต้องนำมา แพ็คเกจ รวมกันเป็น ไฟล์ .jar จำนวน 1 ไฟล์ โดยสามารถมี โปรแกรมประยุกต์มิดเลท หลายตัวใน ไฟล์ .jar เดียวกันได้ ซึ่งเราเรียก กลุ่ม ของ โปรแกรมประยุกต์มิดเลท นี้ว่า ชุดมิดเลท หลังจากทำให้เป็นแพกเก็จเดียวกัน (Packaging) แล้วก็จะสามารถ ดาว์นโหลด และ ติดตั้ง ลงในอุปกรณ์ ได้โดยอาจจะผ่านทาง สายเชื่อม ต่อเข้าโดยตรงจากคอมพิวเตอร์ หรือ จาก เครือข่ายไร้สาย

แมนิเฟสไฟล์ (manifest file) เป็นไฟล์ หนึ่งที่จะต้องถูก แพ็คเกจ อยู่ใน ไฟล์ .jar ของ ชุดมิดเลทมีไว้เพื่อเป็นตัวบอกว่าใน ไฟล์ .jar นั้น ๆ มี ส่วนประกอบ อะไรบ้าง และ ประกอบด้วยข้อมูลอื่น ๆ เช่น name, version, vendor ของ ชุดมิดเลทเป็นต้น แมนิเฟสไฟล์ ประกอบด้วยรายการของแอตทริบิวต์ ดังตัวอย่างต่อไปนี้

MIDlet-Name: HelloMidlet

MIDlet-Vendor: MidletZone.com

MIDlet-Version: 1.0

MIDlet-1: HelloMidlet, /Icon.png, HelloMidlet

MicroEdition-คอนฟิกูเรชั่น: CLDC-1.0

MicroEdition-Profile: MIDP-1.0

MIDlet-Data-Size: 0

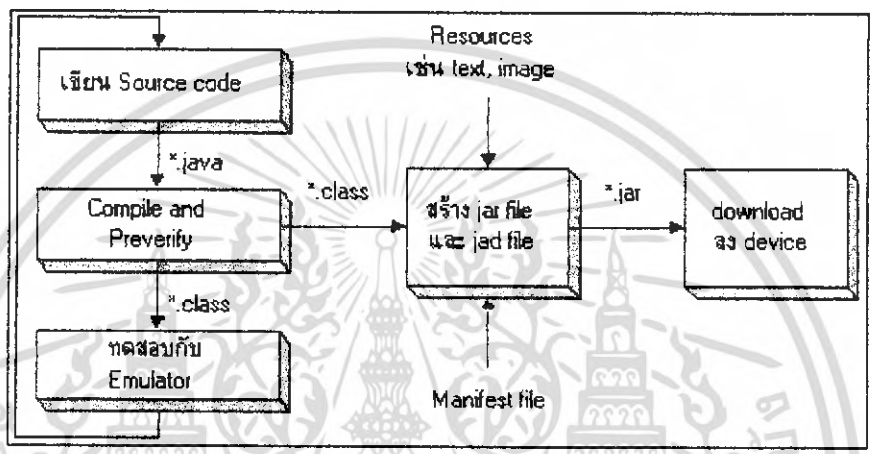
MIDlet Development Steps

ขั้นตอนในการการพัฒนามิดเลท จะมีขั้นตอนดังนี้

1. ทำการสร้างมิดเลท และ ข้อมูลอื่น ๆ ที่ต้องใช้ในโปรแกรม เช่น ไฟล์รูปภาพ, ไอคอนต่าง ๆ
2. สร้างชุดมิดเลท
3. สร้างไฟล์ .jad
4. ทดลองรันโปรแกรมผ่าน โปรแกรมจำลอง (Emulator)
5. ทำการตรวจสอบแก้ไข หากมีข้อผิดพลาดให้กลับไปทำตั้งแต่ 1-3 ใหม่อีกครั้ง
6. สุดท้าย เป็นการทำตัวติดตั้ง สำหรับการติดตั้งเข้าสู่โทรศัพท์มือถือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการพัฒนามิดเลท หลังจากที่คุณได้ทำการเขียนโค้ดขึ้นมาแล้ว คุณจะต้องทำการคอมไพล์ โค้ดนั้นด้วย Javac (Java Compiler) สิ่งที่ได้คือไฟล์ .class ขั้นตอนต่อไปก็คือคุณต้อง นำไฟล์ .class ของคุณไปทำการตรวจสอบความถูกต้อง อีกครั้ง ก่อนจะส่งไฟล์ ที่ได้ไปให้กับมือถือทดลองรัน เพื่อตรวจสอบการทำงาน ว่าถูกต้องอย่างที่ต้องการหรือไม่ จะเห็นได้ว่าเป็นเรื่องที่ยุ่งยากมาก ในการที่จะเขียน และพัฒนามิดเลท ขึ้นมาสักตัว ด้วยเหตุนี้ จึงเกิดการพัฒนาเครื่องมือที่ใช้ในการพัฒนา มิดเลท เพื่อให้ผู้พัฒนา มีความสะดวกใน สร้างและพัฒนามิดเลท

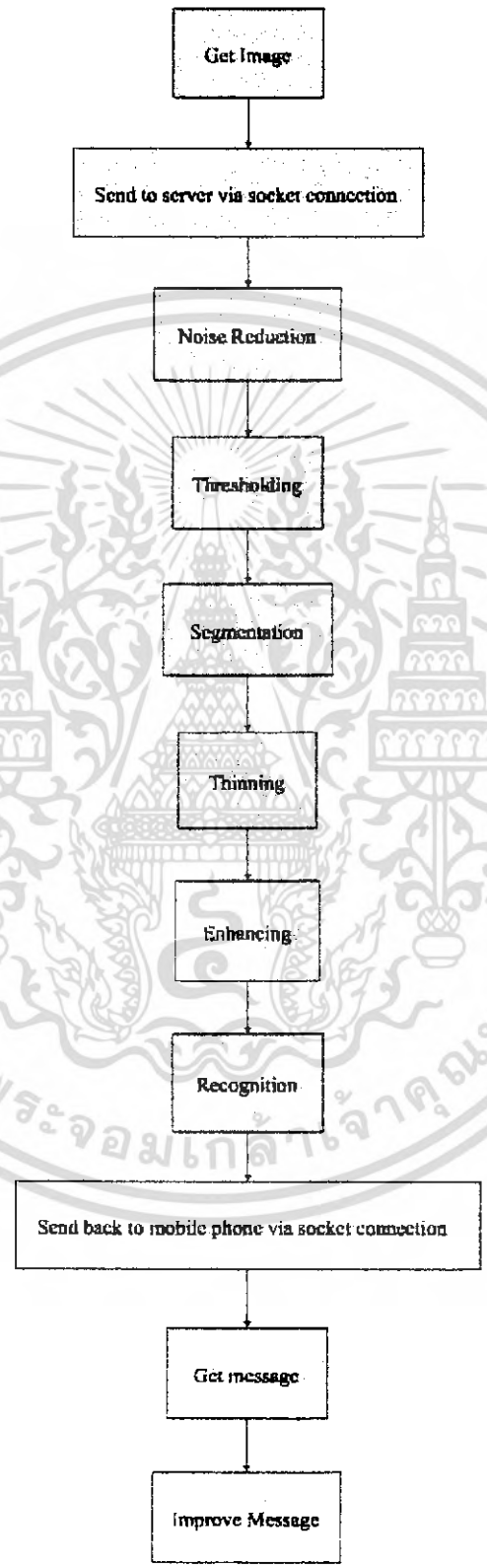


รูปที่ 2.36 ขั้นตอนการพัฒนามิดเลท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

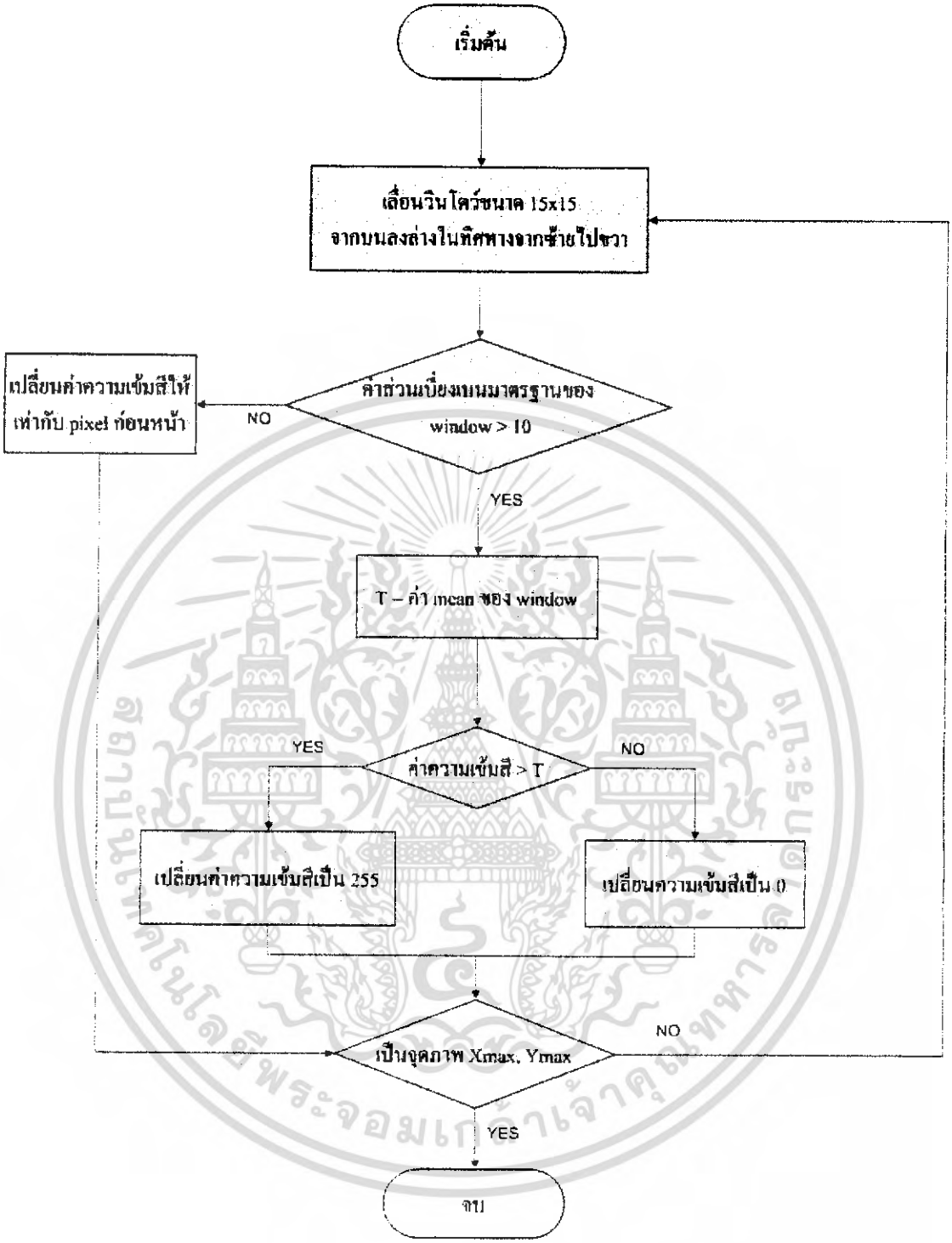
บทที่ 3

การออกแบบ



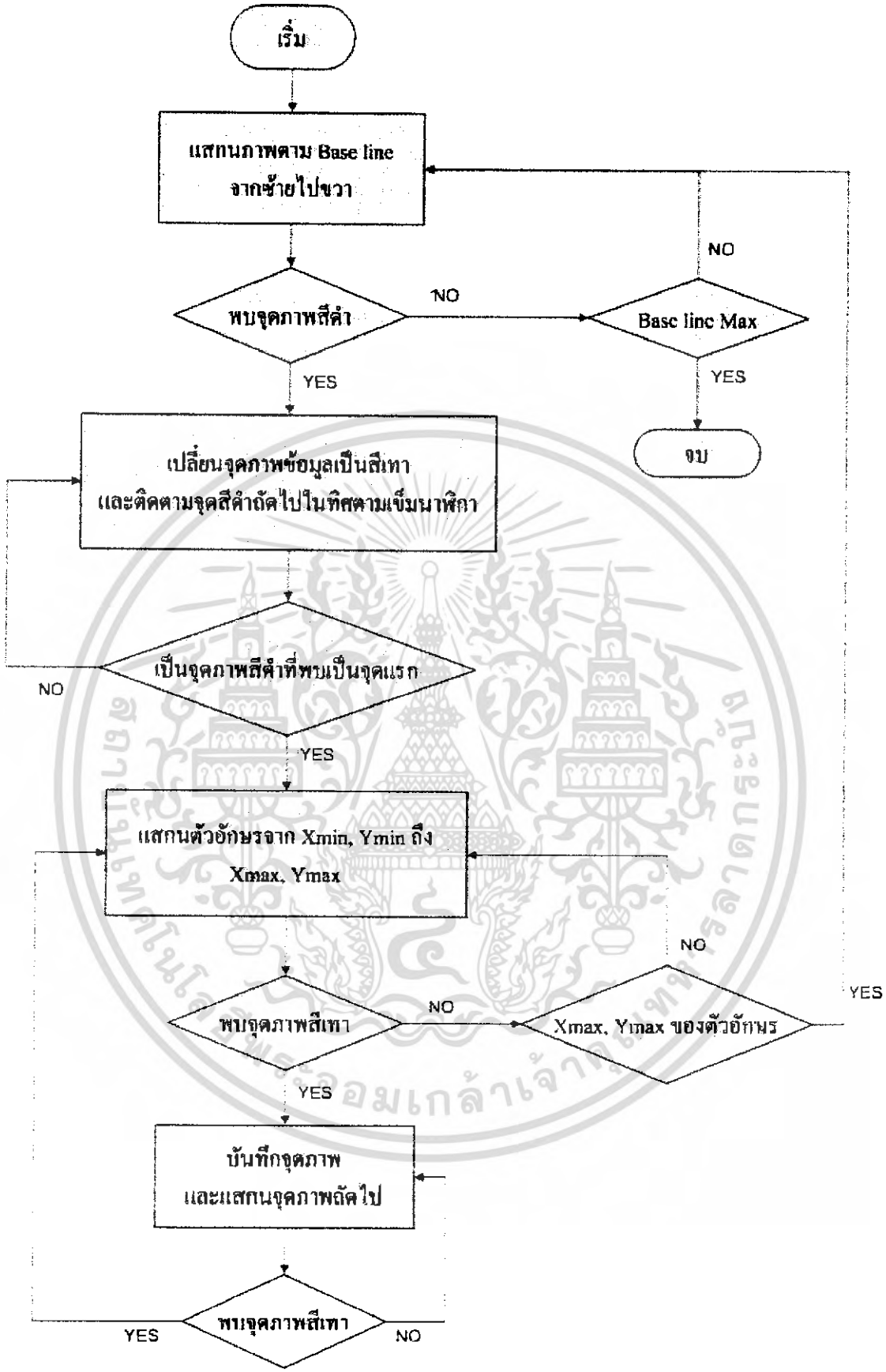
รูปที่ 3.1 Block Diagram การทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



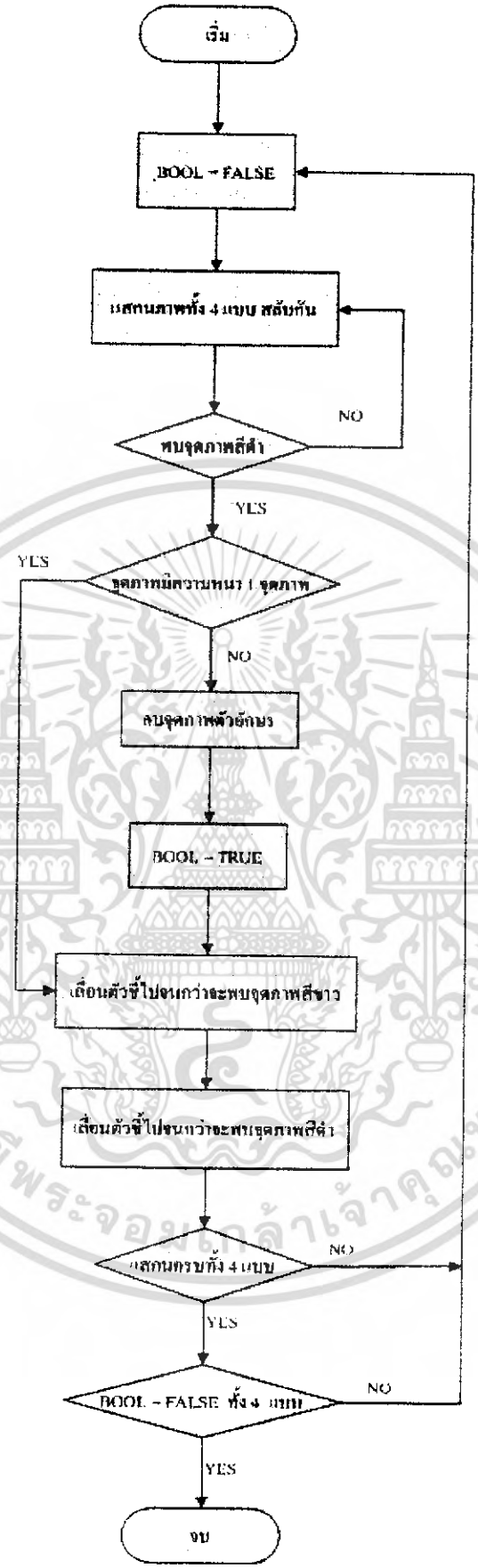
รูปที่ 3.2 Flow-chart การทำงานของขั้นตอนการทำเป็นภาพขาวดำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



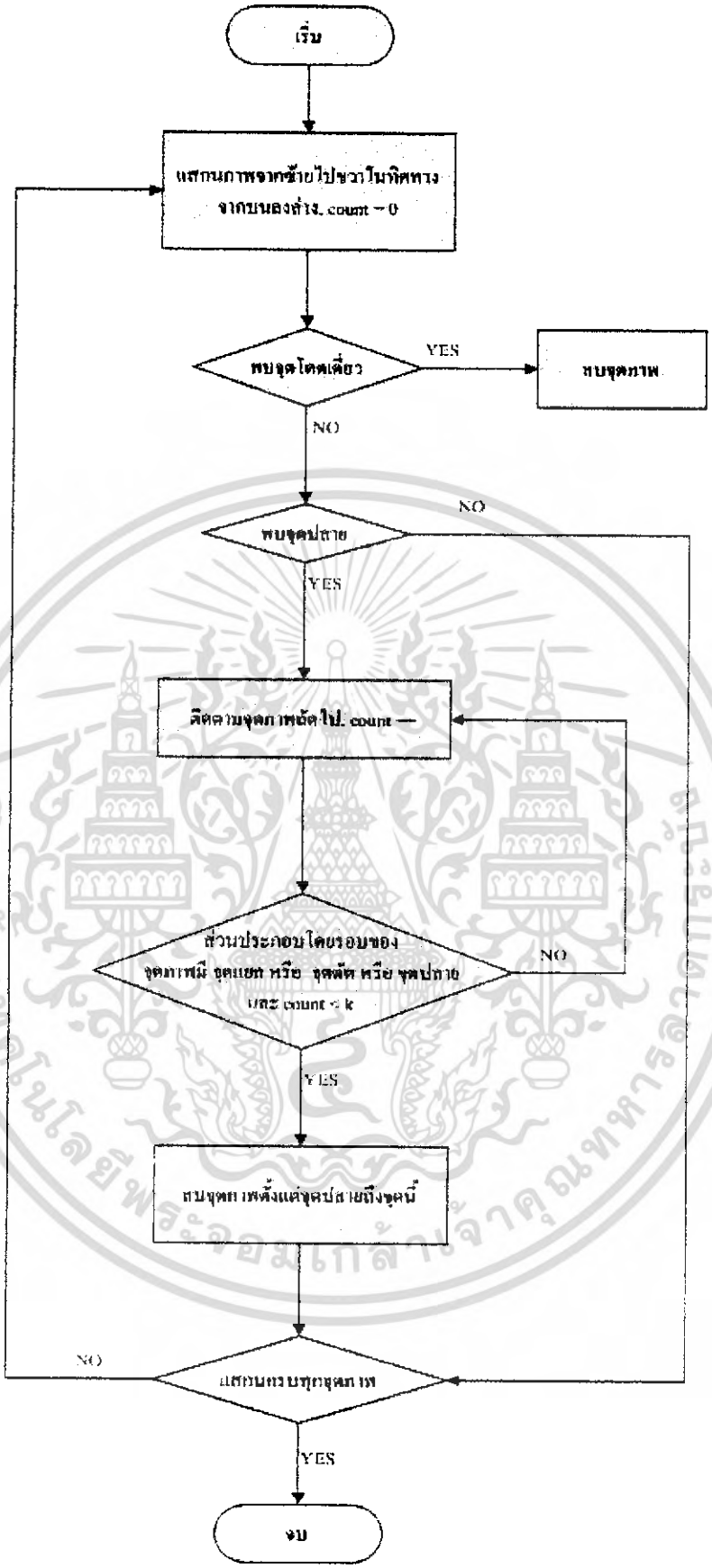
รูปที่ 3.3 Flow-chart การทำงานของขั้นตอนการตัดแยกตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



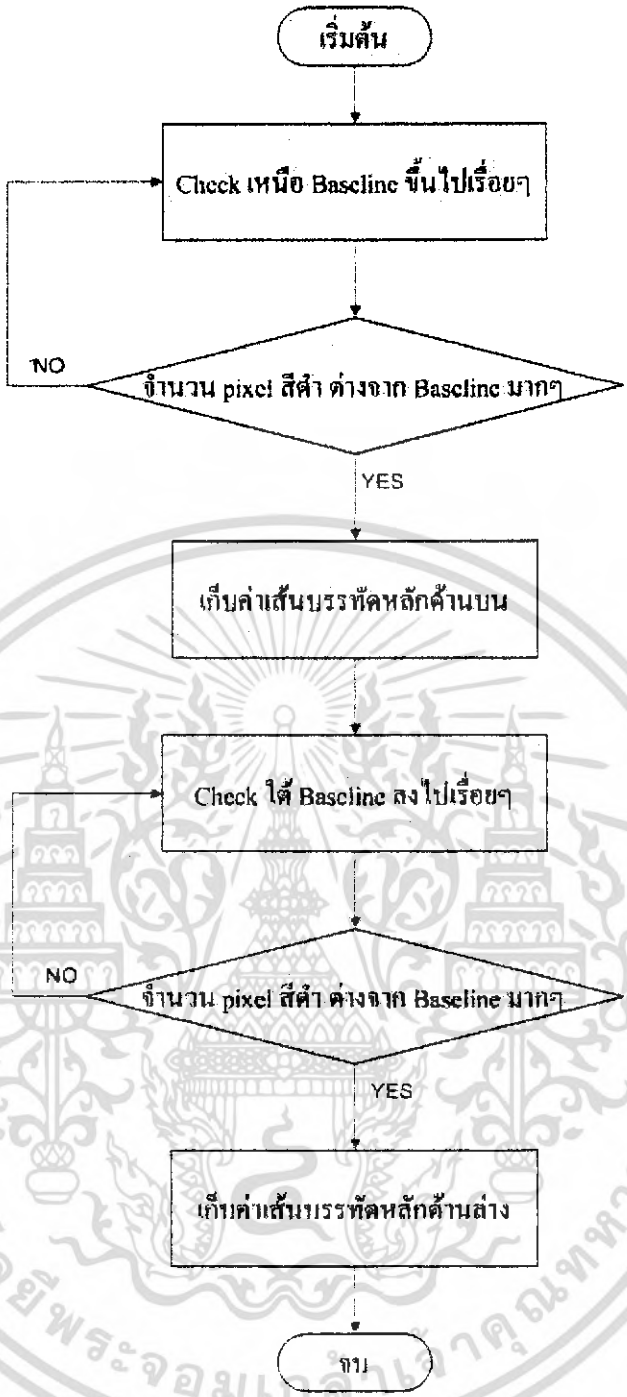
รูปที่ 3.4 Flow-chart การทำงานของขั้นตอนการทำตัวอักษรให้บาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 Flow-chart การทำงานของขั้นตอนการทำกำจัดส่วนเกินของตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



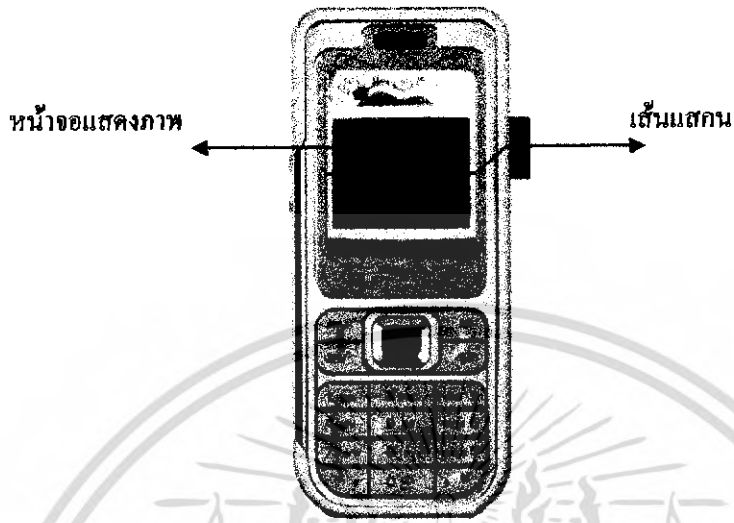
รูปที่ 3.6 Flow-chart การทำงานของขั้นตอนการหาเส้นบรรทัด

ส่วนติดต่อกับผู้ใช้ (User Interface)

1. หน้าจอขณะจับภาพ (Viewfinder) จะมีเส้นแสกนปรากฏอยู่บริเวณขอบด้านซ้ายและด้านขวาของหน้าจอโทรศัพท์มือถือ โดยในขณะจะทำการกดถ่ายภาพ ต้องเล็งให้ข้อความที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องการถ่าย อยู่ตรงกลางหน้าจอแสดงภาพ (ข้อความจะต้องอยู่ตรงกลางระหว่างเส้นแสกนด้านซ้ายและด้านขวา)



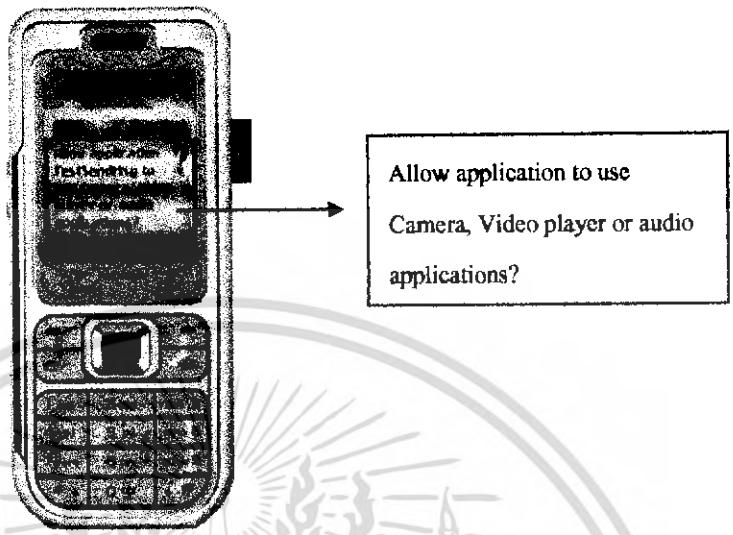
รูปที่ 3.7 หน้าจอแสดงภาพ

2. เมื่อต้องการกดถ่ายภาพ ให้กดปุ่มถ่ายภาพ (Capture) หรือกดตรงกลางของปุ่มเลื่อน(Scroll key)

รูปที่ 3.8 ปุ่มถ่ายภาพและปุ่มตรงกลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เมื่อกดถ่ายภาพจะมีข้อความขึ้นมาถามว่าจะอนุญาตให้ โปรแกรมประยุกต์ ใช้กล้องหรือไม่ ให้ตอบตกลง (Yes)



รูปที่ 3.9 หน้าจอโทรศัพท์มือถือที่แสดงข้อความเมื่อกดถ่ายภาพ

4. เมื่อกด "Yes" หน้าจอโทรศัพท์มือถือจะแสดงภาพที่ได้ถ่ายไว้

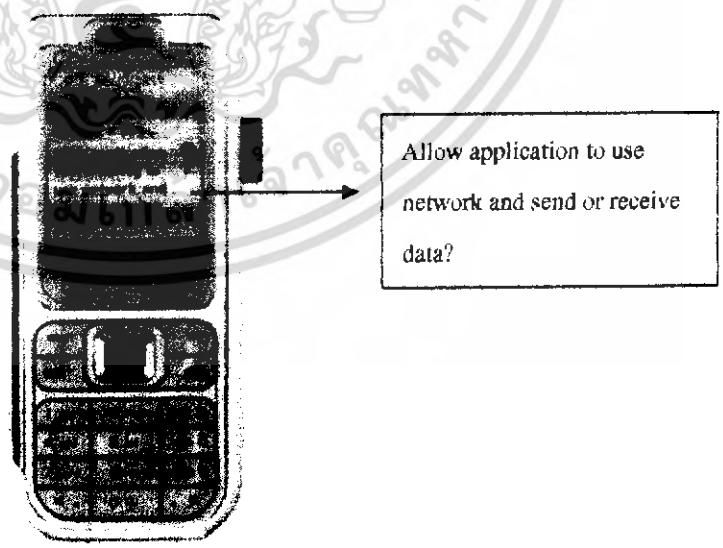
รูปที่ 3.10 หน้าจอโทรศัพท์มือถือที่แสดงภาพที่ถ่ายไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. กดปุ่มประมวลผล (Process) เพื่อทำการเรียกใช้ GPRS ในการส่งภาพไปประมวลผลที่เครื่อง เซิร์ฟเวอร์

รูปที่ 3.11 ปุ่มประมวลผล

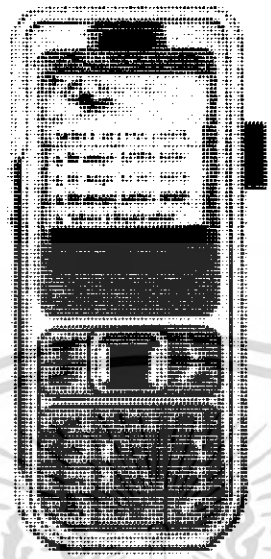
6. เมื่อกดปุ่ม จะมีข้อความขึ้นมาถามว่าจะอนุญาตให้ โปรแกรมประยุกต์ ใช้ เครือข่าย และ รับ - ส่งข้อมูล หรือไม่ ให้ตอบตกลง (Yes)



รูปที่ 3.12 หน้าจอโทรศัพท์มือถือที่แสดงข้อความเมื่อกดประมวลผล

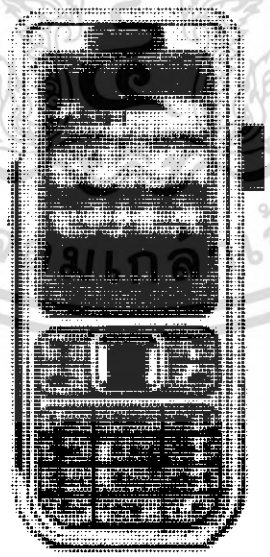
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. เมื่อกด "Yes" แล้ว ให้เลือก Access point ที่ต้องการใช้



รูปที่ 3.13 รายชื่อ Access point

8. รอให้โทรศัพท์ทำการเชื่อมต่อไปที่เครื่อง เซิร์ฟเวอร์ และประมวลผลสักครู่ เมื่อ เซิร์ฟเวอร์ประมวลผลเสร็จ จะส่งข้อความ (text) กลับมาแสดงใน Textbox บนหน้าจอโทรศัพท์มือถือ



รูปที่ 3.14 กล่องข้อความ แสดงข้อความ "ผู้หญิงพิมพ์ฎีกา" บนหน้าจอโทรศัพท์มือถือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

จากโปรแกรมที่ได้พัฒนาขึ้นมาตามทฤษฎีที่ได้กล่าวไว้ข้างต้น ผู้จัดทำได้ทำการเก็บข้อมูลรหัส คิวโค้ด ตัวอักษรไว้ 2 รูปแบบด้วยกัน คือ ตัวอักษรแบบ Angsana New และ Angsana UPC และจะทำการประเมินประสิทธิภาพการรู้จำของโปรแกรม โดยการทดสอบกับข้อมูลจริง ในกรณีต่างๆ ดังต่อไปนี้

4.1 การทดสอบตัวอักษรที่มีรูปแบบเดียวกับข้อมูลต้นแบบ

ตารางที่ 4.1 ผลการทดสอบตัวอักษรจากการสุ่มตัวอย่าง 150 ภาพข้อความ

อักษร	จำนวนตัวอักษรที่วิเคราะห์ได้ (ตัว)	จำนวนตัวอักษรที่วิเคราะห์ได้ถูกต้อง (ตัว)	จำนวนตัวอักษรที่วิเคราะห์ได้จากการเดา (ตัว)
ก	56	34	22
ข	4	4	0
ฆ	4	1	3
ค	29	18	11
ฅ	9	4	5
ฆ	1	0	1
ง	11	1	10
จ	18	13	5
ฉ	1	0	1
ช	1	0	1
ซ	2	2	0
ฌ	-	-	-
ญ	35	13	22
ฎ	4	3	1
ฏ	-	-	-
ฐ	5	0	5
ฑ	9	5	4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 1 ผลการทดสอบตัวอักษรจากการสุ่มตัวอย่าง 150 ภาพข้อความ (ต่อ)

ตัวอักษร	จำนวนตัวอักษรที่วิเคราะห์ (ตัว)	จำนวนตัวอักษรที่วิเคราะห์ได้ถูกต้อง (ตัว)	จำนวนตัวอักษรที่วิเคราะห์ได้จากการเดา (ตัว)
ฌ	3	0	3
ณ	20	13	7
ด	12	0	12
ต	16	0	16
ถ	2	2	0
ท	18	10	8
ธ	13	9	4
น	49	27	22
บ	26	3	23
ป	17	6	11
ผ	14	12	2
ฝ	8	0	8
พ	11	2	9
ฟ	1	0	1
ภ	1	1	0
ม	37	21	16
ย	5	4	1
ร	96	34	62
ล	15	11	4
ว	20	15	5
ศ	22	4	18
ษ	9	1	8
ส	16	3	13
ห	15	6	9
ฬ	9	4	5
อ	26	22	4
ฮ	23	0	23
ฤ	6	0	6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 1 ผลการทดสอบตัวอักษรจากการสุ่มตัวอย่าง 150 ภาพข้อความ (ต่อ)

ตัวอักษร	จำนวนตัวอักษรที่วิเคราะห์ได้ (ตัว)	จำนวนตัวอักษรที่วิเคราะห์ได้ถูกต้อง (ตัว)	จำนวนตัวอักษรที่วิเคราะห์ได้จากการเดา (ตัว)
ะ	34	13	21
ั	28	1	27
า	125	69	56
ำ	-	-	-
ิ	20	11	9
ี	38	27	11
ึ	1	1	0
ื	-	-	-
ุ	12	0	12
ู	13	5	8
เ	32	1	31
แ	4	0	4
โ	1	0	1
ใ	2	2	0
ไ	-	-	-
ๆ	9	1	8
็	1	1	0
่	8	3	5
้	22	19	3
๊	-	-	-
๋	-	-	-
์	10	7	3
!	9	-	-

หมายเหตุ* เครื่องหมาย ! คือ ตัวสัญลักษณ์แสดงครั้งที่ไม่สามารถเดาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุป

5.1 สรุปผลการทดลอง

จากการทดลองพบว่า โปรแกรมที่พัฒนาขึ้นสามารถรู้จำตัวอักษรได้อย่างมีประสิทธิภาพ โดยอาจแบ่งเป็นกลุ่มๆ ได้ดังนี้

1. กลุ่มที่ผลการรู้จำเชื่อถือได้ในระดับสูงสุด

- ก, ข ,ง, จ, ฉ, ช, ซ, ญ, ณ, ท, น, บ, ม, ย, ร, ล, ว, ห, อ , ั , า , ำ , ิ , ี , ึ , ุ , เ , โ , ไ , ็ , ้ , + , ์

2. กลุ่มที่ผลการรู้จำเชื่อถือได้ในระดับกลาง

- ข, ค, ฅ, ฆ, ฎ, ฑ, ต, ถ, ธ, ป, ผ, ฟ, ศ, ษ, ส, ื , ฯ, ่ , ๊

3.กลุ่มที่ผลการรู้จำเชื่อถือได้ในระดับต่ำ

- ฆ, ฎ, ฐ, ฒ, ด, ฝ, พ, ภ, ห, ฮ, ฤ, ะ , ู , ใ , ๆ

5.2 ปัญหาและอุปสรรค

ปัญหาและอุปสรรคที่พบในการพัฒนาโปรแกรม ได้แก่

1. ปัญหาเกี่ยวกับคุณภาพของภาพข้อความ

- ปัญหาตัวอักษรเบลอ อันเนื่องมาจากคุณภาพของกล้องในโทรศัพท์มือถือ
- อุปกรณ์ (โทรศัพท์มือถือ) ไม่รองรับ Java Platform 7 ซึ่งสามารถเรียกใช้ฟังก์ชันโฟกัสภาพได้
- คุณภาพของภาพข้อความที่ถ่ายจากโปรแกรม มีคุณภาพด้อยกว่าภาพที่ถ่ายจากฟังก์ชันของกล้องในโทรศัพท์มือถืออยู่มาก
- ปัญหาแสงและเงาในภาพข้อความ

2. ปัญหาเกี่ยวกับลักษณะของตัวอักษรภาษาไทย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รูปแบบของตัวอักษรที่ต่างกัน ทำให้การวิเคราะห์ตัวอักษรทำได้ยากขึ้น เช่น ตัวอักษรในบางรูปแบบมีหัว บางรูปแบบไม่มีหัว
- ตัวอักษรหลายตัวมีลักษณะแตกต่างกันน้อยมาก เช่น หัวเข้าและหัวออก มีหยักหรือไม่มีหยัก
- ปัญหาตัวอักษรที่มีการติด หรือซ้อนทับกัน

3. จีพีอาร์เอส

- การส่งข้อมูลผ่าน จีพีอาร์เอส ในทางปฏิบัติแล้วทำได้ช้ากว่าทฤษฎี ทำให้ระยะเวลาการประมวลผลล่าช้ามากขึ้น

5.3 แนวทางการพัฒนา

1. อาจพัฒนาโปรแกรมให้สามารถประมวลผลบนโทรศัพท์มือถือได้
2. อาจพัฒนาให้ใช้งานร่วมกับโปรแกรมการสะกดคำในโทรศัพท์มือถือ เพื่อให้ผลออกมาถูกต้องมากยิ่งขึ้น
3. อาจพัฒนาเพื่อใช้กับภาษาอื่นๆ ได้
4. อาจพัฒนาเพื่อให้สามารถเชื่อมต่อประโยคยาวๆ จากการถ่ายรูปหลายๆรูปมาต่อกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. http://www.thai-programmer.com
2. ced.kmitnb.ac.th/tawa/WinSocket.doc
3. พ.อ.เจนวิทย์ เหลืองอร่าม และปิยวิทย์ เหลืองอร่าม.2546.การเขียนโปรแกรมสำหรับ Wireless Application ด้วย J2ME. กรุงเทพฯ:ซีเอ็ดยูเคชั่น.
4. นรินทร์ ทัศนลีลพร และสมศักดิ์ รัตนฤชัตร.2539.Printed Thai Character Recognition.พิมพ์ครั้งที่ 1.กรุงเทพฯ.ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
5. J.Sauvola* and M.Pietikäinen.1999.Adaptive document image binarization.Pattern Recognition 33 (2000) 225-236
6. พ.อ.เจนวิทย์ เหลืองอร่าม และปิยวิทย์ เหลืองอร่าม.2543.การเขียนโปรแกรมสำหรับ Application ด้วย C/C++. กรุงเทพฯ:ซีเอ็ดยูเคชั่น.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

จากโปรแกรมที่ได้พัฒนาขึ้นมาตามทฤษฎีที่ได้กล่าวไว้ข้างต้น ผู้จัดทำได้ทำการเก็บข้อมูลรหัส คิวโค้ด ตัวอักษรไว้ 2 รูปแบบด้วยกัน คือ ตัวอักษรแบบ Angsana New และ Angsana UPC และจะทำการประเมินประสิทธิภาพการรู้จำของโปรแกรม โดยการทดสอบกับข้อมูลจริง ในกรณีต่างๆ ดังต่อไปนี้

4.1 การทดสอบตัวอักษรที่มีรูปแบบเดียวกับข้อมูลต้นแบบ

ตารางที่ 4.1 ผลการทดสอบตัวอักษรจากการสุ่มตัวอย่าง 150 ภาพข้อความ

อักษร	จำนวนตัวอักษรที่วิเคราะห์ได้ (ตัว)	จำนวนตัวอักษรที่วิเคราะห์ได้ถูกต้อง (ตัว)	จำนวนตัวอักษรที่วิเคราะห์ได้จากการเดา (ตัว)
ก	56	34	22
ข	4	4	0
ฃ	4	1	3
ค	29	18	11
ฅ	9	4	5
ฆ	1	0	1
ง	11	1	10
จ	18	13	5
ฉ	1	0	1
ช	1	0	1
ซ	2	2	0
ฌ	-	-	-
ญ	35	13	22
ฎ	4	3	1
ฏ	-	-	-
ฐ	5	0	5
ฑ	9	5	4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 1 ผลการทดสอบตัวอักษรจากการสุ่มตัวอย่าง 150 ภาพข้อความ (ต่อ)

ตัวอักษร	จำนวนตัวอักษรที่วิเคราะห์ (ตัว)	จำนวนตัวอักษรที่วิเคราะห์ได้ถูกต้อง (ตัว)	จำนวนตัวอักษรที่วิเคราะห์ได้จากการเดา (ตัว)
ฆ	3	0	3
ณ	20	13	7
ด	12	0	12
ต	16	0	16
ถ	2	2	0
ท	18	10	8
ธ	13	9	4
น	49	27	22
บ	26	3	23
ป	17	6	11
ผ	14	12	2
ฝ	8	0	8
พ	11	2	9
ฟ	1	0	1
ภ	1	1	0
ม	37	21	16
ย	5	4	1
ร	96	34	62
ล	15	11	4
ว	20	15	5
ศ	22	4	18
ษ	9	1	8
ส	16	3	13
ห	15	6	9
ฬ	9	4	5
อ	26	22	4
ฮ	23	0	23
ฤ	6	0	6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 1 ผลการทดสอบตัวอักษรจากการสุ่มตัวอย่าง 150 ภาพข้อความ (ต่อ)

ตัวอักษร	จำนวนตัวอักษรที่วิเคราะห์ได้ (ตัว)	จำนวนตัวอักษรที่วิเคราะห์ได้ถูกต้อง (ตัว)	จำนวนตัวอักษรที่วิเคราะห์ได้จากการเดา (ตัว)
ะ	34	13	21
ั	28	1	27
า	125	69	56
ำ	-	-	-
ิ	20	11	9
ี	38	27	11
ึ	1	1	0
ื	-	-	-
ุ	12	0	12
ู	13	5	8
เ	32	1	31
แ	4	0	4
โ	1	0	1
ใ	2	2	0
ไ	-	-	-
็	9	1	8
่	1	1	0
้	8	3	5
๊	22	19	3
๋	-	-	-
์	-	-	-
ๆ	10	7	3
!	9	-	-

หมายเหตุ* เครื่องหมาย ! คือ ตัวสัญลักษณ์แสดงครั้งที่ไม่สามารถเดาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุป

5.1 สรุปผลการทดลอง

จากการทดลองพบว่า โปรแกรมที่พัฒนาขึ้นสามารถรู้จำตัวอักษรได้อย่างมีประสิทธิภาพ โดยอาจแบ่งเป็นกลุ่มๆ ได้ดังนี้

1. กลุ่มที่ผลการรู้จำเชื่อถือได้ในระดับสูงสุด

- ก, ข ,ง, จ, ฉ, ช, ซ, ญ, ณ, ท, น, บ, ม, ย, ร, ล, ว, ห, อ , ั , า , ำ , ิ , ี , ึ , ื , ุ , เ , โ , ไ , ็ , ้ , ่ , ์

2. กลุ่มที่ผลการรู้จำเชื่อถือได้ในระดับกลาง

- ข, ค, ศ, ฌ, ฏ, ฑ, ต, ถ, ธ, ป, ผ, ฟ, ศ, ษ, ส, ึ , ฯ, ๋ , ๊

3.กลุ่มที่ผลการรู้จำเชื่อถือได้ในระดับต่ำ

- ฆ, ฏ, ฐ, ฒ, ด, ฝ, พ, ภ, ฬ, ฮ, ฤ, ะ , ู , ใ , ๆ

5.2 ปัญหาและอุปสรรค

ปัญหาและอุปสรรคที่พบในการพัฒนาโปรแกรม ได้แก่

1. ปัญหาเกี่ยวกับคุณภาพของภาพข้อความ

- ปัญหาตัวอักษรเบลอ อันเนื่องมาจากคุณภาพของกล้องในโทรศัพท์มือถือ
- อุปกรณ์ (โทรศัพท์มือถือ) ไม่รองรับ Java Platform 7 ซึ่งสามารถเรียกใช้ฟังก์ชันโฟกัสภาพได้
- คุณภาพของภาพข้อความที่ถ่ายจากโปรแกรม มีคุณภาพด้อยกว่าภาพที่ถ่ายจากฟังก์ชันของกล้องในโทรศัพท์มือถืออยู่มาก
- ปัญหาแสงและเงาในภาพข้อความ

2. ปัญหาเกี่ยวกับลักษณะของตัวอักษรภาษาไทย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รูปแบบของตัวอักษรที่ต่างกัน ทำให้การวิเคราะห์ตัวอักษรทำได้ยากขึ้น เช่น ตัวอักษรในบางรูปแบบมีหัว บางรูปแบบไม่มีหัว
- ตัวอักษรหลายตัวมีลักษณะแตกต่างกันน้อยมาก เช่น หัวเข้าและหัวออก มีหยักหรือไม่มีหยัก
- ปัญหาตัวอักษรที่มีการติด หรือซ้อนทับกัน

3. จีพีอาร์เอส

- การส่งข้อมูลผ่าน จีพีอาร์เอส ในทางปฏิบัติแล้วทำได้ช้ากว่าทฤษฎี ทำให้ระยะเวลาการประมวลผลล่าช้ามากขึ้น

5.3 แนวทางการพัฒนา

1. อาจพัฒนาโปรแกรมให้สามารถประมวลผลบนโทรศัพท์มือถือได้
2. อาจพัฒนาให้ใช้งานร่วมกับโปรแกรมการสะกดคำในโทรศัพท์มือถือ เพื่อให้ผลออกมาถูกต้องมากยิ่งขึ้น
3. อาจพัฒนาเพื่อใช้กับภาษาอื่นๆ ได้
4. อาจพัฒนาเพื่อให้สามารถเชื่อมต่อประโยคยาวๆ จากการถ่ายรูปหลายๆรูปมาต่อกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. http://www.thai-programmer.com
2. ced.kmitnb.ac.th/tawa/WinSocket.doc
3. พ.อ.เจนวิทย์ เหลืองอร่าม และปิยวิทย์ เหลืองอร่าม.2546.การเขียนโปรแกรมสำหรับ Wireless Application ด้วย J2ME. กรุงเทพฯ:ซีเอ็ดยูเคชั่น.
4. นรินทร์ ทัศนลีลพร และสมศักดิ์ รัตนฤขัตร.2539.Printed Thai Character Recognition.พิมพ์ครั้งที่ 1.กรุงเทพฯ.ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
5. J.Sauvola* and M.Pietikäinen.1999.Adaptive document image binarization.Pattern Recognition 33 (2000) 225-236
6. พ.อ.เจนวิทย์ เหลืองอร่าม และปิยวิทย์ เหลืองอร่าม.2543.การเขียนโปรแกรมสำหรับ Application ด้วย C/C++. กรุงเทพฯ:ซีเอ็ดยูเคชั่น.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้