การรู้จำรูปแบบตัวอักษร

(CHARACTERS RECOGNITON)

โดย

นาย สักก์ศิษฎ์ บุญญาวิโรจน์

นาย อนุชา จันทระวิ

นาย อาทิตย์ แย้มไสย

ปริญญานิพนธ์ส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2541

เลขหมู่.....
เลขทะเบียน.....34077.....
วัน, เดือน, ปี.....1 ต.ค. 2542

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2541

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การรู้จำตัวอักษร

ผู้จัดทำ

นายลักก์ศิษฎ์ บุญญาวิโรจน์

นายอนุชา จันทระวี

นายอาทิตย์ แย้มไสย

.....

(อาจารย์สุรเดช ตรีไตรลักษณะ)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

รายงานฉบับนี้จะสำเร็จลงไม่ได้ถ้าปราศจากการแนะนำ และการช่วยเหลือจาก อาจารย์ ดุรุเดช ตรีไตรลักษณะ อาจารย์ที่ปรึกษา และอาจารย์ สุรพันธุ์ เอื้อไพบูลย์ ที่ท่านได้ทำงานวิจัยเรื่องการจดจำตัวอักษรลายมือเขียนมาก่อน ซึ่งทำให้เป็นแนวทางในการทำงานวิจัยเรื่องนี้

สุดท้ายนี้ขอขอบพระคุณบิดาและมารดาผู้ที่ให้การสนับสนุนและส่งเสริมมาโดยตลอด ทางผู้จัดทำต้องขอขอบพระคุณทุกๆท่านไว้ ณ โอกาสนี้ด้วย

นาย สักก์ศิษฎ์ บุญญาวิโรจน์

นาย อนุชา จันทระวี

นาย อาทิตย์ แย้มไสย

ผู้จัดทำ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรู้จำรูปแบบตัวอักษร

นาย สักก์ศิษฎ์ บุญญาวิโรจน์
นาย อนุชา จันทระวี
นาย อาทิตย์ แย้มไสย
อ. สุรเดช ตรีไตรลักษณะ อาจารย์ที่ปรึกษา
ภาคการศึกษาที่ 2 ปีการศึกษา 2541

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้เป็นเรื่องเกี่ยวกับการวิเคราะห์ผลสัญญาณภาพ โดยใช้คอมพิวเตอร์เป็นตัววิเคราะห์และประมวลผล ซึ่งในที่นี้เป็นการวิเคราะห์สัญญาณภาพเกี่ยวกับการจดจำตัวอักษรภาษาไทยลายมือเขียน (Thai Character Recognition) เพื่อเป็นการพัฒนาระบบการป้อนข้อมูลเข้าสู่คอมพิวเตอร์ โดยสามารถอ่านลายมือเขียนโดยไม่ต้องป้อนข้อมูลผ่านแป้นพิมพ์ หรือ อ่านจากแฟ้มข้อมูลในหน่วยความจำ แต่จะป้อนข้อมูลโดยการสแกน ตัวอักษรบนกระดาษเข้าสู่คอมพิวเตอร์โดยตรงและในการจดจำตัวอักษรไทยลายมือเขียนจะนำเอาลักษณะเด่นคือหัวของตัวอักษรมาใช้ในการจำแนกกลุ่มของตัวอักษร สำหรับโปรแกรมที่ใช้ในการประมวลผลจะเป็นโปรแกรมที่เขียนด้วยภาษาซี++

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CHARACTER RECOGNITION

Mr. Saksit Boonyavirod

Mr. Anucha Chuntaravee

Mr. Artit Yamsai

Mr. Suradech treetriluksana. Advisor

2nd Semester , Education Year 1998

Abstract

This Project concerns with the image processing which uses computer as analyzer and processor. The aim of this project is recognition of the hand writing thai character

Hand writing thai character recognition is based on imminent characteristic such as the head of character, to classify group of character. The software that is used in this project is writing in c++ language

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
ABSTRACT	III
สารบัญ	IV
สารบัญรูป	VI
บทที่ 1 บทนำ	1
บทที่ 2 อุปกรณ์ที่ใช้ในการเก็บข้อมูล และลักษณะของข้อมูล	3
2.1 เครื่องตรวจกวาดภาพ และลักษณะการทำงาน	3
2.2 ลักษณะข้อมูลภาพกราฟิก	4
2.2.1 ลักษณะข้อมูลภาพกราฟิก	4
2.2.2 การเปลี่ยนข้อมูลชนิดแมพเป็นข้อมูลไบนารี่	5
บทที่ 3 การเตรียมข้อมูล และขั้นตอนการจดจำตัวอักษร	7
3.1 การกำจัดสัญญาณรบกวน	7
3.2 ขนาดมาตรฐาน	8
3.3 การทำตัวอักษรให้บาง	10
3.4 การคำนวณค่าตัวเลขต่อเนื่อง	15
3.5 การกำจัดส่วนเกิน	16
บทที่ 4 การจำแนกตัวอักษร	19
4.1 การตรวจสอบหัวของตัวอักษร	19
4.2 การแบ่งกลุ่มของตัวอักษรแบบหยาบ	25
4.3 การแบ่งกลุ่มของตัวอักษรแบบละเอียด	27
4.4 เทคนิคและวิธีการแยกตัวอักษรที่คล้ายกัน	28
4.4.1 การแบ่งเขตย่อย	28
4.4.2 การหารหัสรูปร่างบางส่วนของตัวอักษร	30
4.4.3 การหาลักษณะหัวของตัวอักษร	30
4.4.4 การหาอัตราส่วนความกว้างต่อความสูง	31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

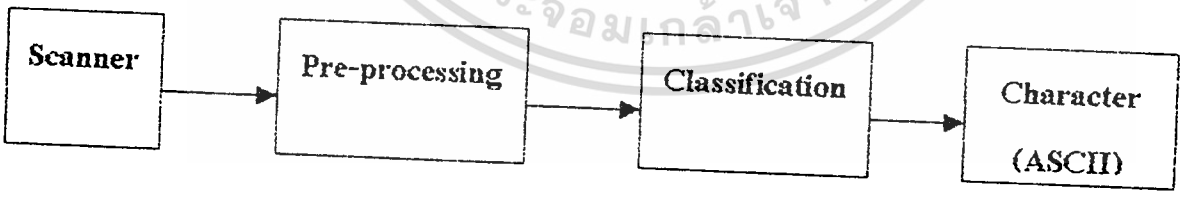
บทที่ 5 ผลการทดลองและสรุปผล
5.1 ผลการทดลอง 40
5.2 สรุปผลการทดลอง 40
บรรณานุกรม 42
ภาคผนวก
โปรแกรมการจดจำตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ปัจจุบัน computer ได้มีบทบาทในงานสาขาต่างๆ ของมนุษย์มากมาย เช่น การใช้ computer เก็บและวิเคราะห์ข้อมูลทางด้านธุรกิจ ด้านวณทางวิศวกรรมศาสตร์ เป็นต้น งานทุกอย่างที่มี computer มาเกี่ยวข้องจะต้องมีการปฏิบัติการอย่างหนึ่งเกิดขึ้นเสมอ ๆ สิ่งนั้นคือการป้อนข้อมูลเข้า อุปกรณ์ในการป้อนข้อมูลที่มีการนำมาใช้กันมากในเครื่อง computer ก็คือ แป้นพิมพ์ การป้อนข้อมูลโดยใช้แป้นพิมพ์นั้นเป็นการส่งรหัสให้แก่เครื่อง Computer เช่น ASCII code ของตัวอักษรต่าง ๆ รวมทั้งตัวอักษรภาษาไทยก็ได้มีการกำหนดรหัสแทนตัวอักษรขึ้นมาใช้กันหลาย ๆแบบ Computer ไม่รับรู้ลักษณะของตัวอักษรที่ถูกพิมพ์เข้าไป แต่จะรับรู้เพียงรหัสแทนตัวอักษรเท่านั้น ดังนั้น จึงทำให้เกิดความยุ่งยากที่จะทำให้ Computer สามารถเข้าใจถึงลักษณะของอักษรแบบต่าง ๆ ที่มนุษย์ใช้กันอยู่

ในขณะที่เรามีข้อมูลซึ่งถูกนำมาเขียนอยู่ในกระดาษอยู่แล้วเราสามารถทำการเก็บข้อมูลลง computer โดยการพิมพ์เข้าไปด้วยแป้นพิมพ์ หรือ มีวิธีการอย่างหนึ่งซึ่งเร็วกว่าคือการป้อนข้อมูลเข้าสู่ Computer ได้ก็คือ โดยการใช้เครื่อง Scanner รูปแบบของตัวอักษรจะถูกอ่านเข้ามาด้วยเครื่อง Scanner แล้วนำข้อมูลที่ได้เข้า computer แต่การนำข้อมูลในลักษณะนี้จะเก็บเป็นรูปภาพซึ่งใช้หน่วยความจำจำนวนมาก แต่เราการเก็บข้อมูลโดยใช้หน่วยความจำให้น้อยลง ดังนั้นก่อนจะเก็บข้อมูลต้องให้ computer ทำการวิเคราะห์ออกมาว่า เป็นตัวอักษรตัวใดเสียก่อน ค่อยทำการเก็บเป็นรหัส ASCII โดยแสดงขั้นตอนดังรูปที่ 1.1



รูปที่ 1.1 แสดงการเก็บรูปแบบของตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ฉบับนี้ได้นำเสนอวิธีในการจดจำตัวอักษรภาษาไทยลายมือเขียน โดยแบ่งเนื้อหาออกเป็นบท ๆ ดังต่อไปนี้

บทที่ 1 บทนำ

บทที่ 2 อุปกรณ์ที่ใช้ในการเก็บข้อมูล และลักษณะของข้อมูล

บทที่ 3 การเตรียมข้อมูล และขั้นตอนการรู้จำตัวอักษร

บทที่ 4 การจำแนกตัวอักษร

บทที่ 5 การทดลอง และผลการทดลอง

บทที่ 5 สรุปและวิจารณ์ผลการทดลองการรู้จำตัวอักษรภาษาไทยลายมือเขียน

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
อุปกรณ์ที่ใช้ในการเก็บข้อมูลและลักษณะของข้อมูล

2.1 เครื่องตรวจกวาดภาพ และลักษณะการทำงาน

เครื่องตรวจกวาดภาพหรืออิมเมจสแกนเนอร์ (Image scanner) เป็นอุปกรณ์ช่วยสนับสนุนการทำงานของเครื่องคอมพิวเตอร์ ใช้เก็บรูปภาพหรือภาพตัวอักษรที่พิมพ์อยู่บนแผ่นกระดาษ นำเข้าไปเก็บไว้ในหน่วยความจำหรืออุปกรณ์เก็บข้อมูลสำรองของเครื่องคอมพิวเตอร์ เช่น ฟลอปปี้ดิสก์ หรือ ฮาร์ดดิสก์ เป็นต้น ได้มีการนำเอาเครื่องสแกนเนอร์มาประยุกต์ใช้เป็นเครื่องอ่านตัวอักษรด้วยแสง(Optical Character Reader , OCR) เพื่อใช้ในงานวิจัยทางด้านการรู้จำตัวอักษร (Character recognition) โดยทำหน้าที่เป็นอุปกรณ์เก็บข้อมูล ตัวอย่างข้อมูลอาจมีลักษณะเป็นหน้ากระดาษ ที่ประกอบด้วยรูปภาพและตัวอักษรพิมพ์เรียงเป็นประโยคข้อความ เมื่อนำแผ่นกระดาษดังกล่าวมาตรวจกวาดหรือสแกนด้วยสแกนเนอร์ ภาพที่ปรากฏบนหน้ากระดาษเอกสารทั้งหมดไม่ว่าจะเป็นรูปภาพหรือตัวอักษรจะถูกเปลี่ยนเป็นข้อมูลไบนารี่และเก็บบันทึกลงเฟ็บข้อมูลในรูปแบบหรือฟอร์แมทแบบกราฟิก ซึ่งจะถูกนำไปประมวลผลในขั้นตอนต่อ ๆ ไป

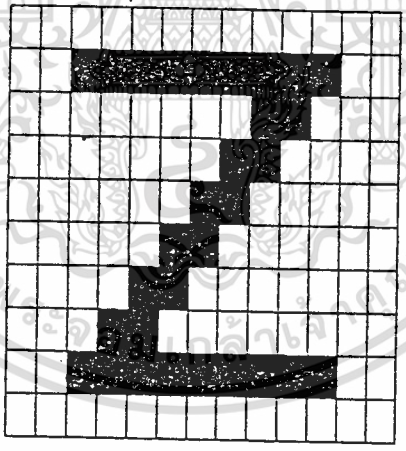
ในการทำงานเครื่องตรวจกวาดภาพทำงานโดยอาศัยหลักการสะท้อนแสง เมื่อฉายแสงไปบนแผ่นกระดาษที่มีรูปภาพหรือข้อความพิมพ์อยู่ แล้วเก็บบันทึกที่สะท้อนกลับออกมาในปริมาณที่แตกต่างกัน ส่วนประกอบของเครื่องตรวจกวาดภาพ ภายในตัวเครื่องจะมีหลอดฟลูออเรสเซนต์ ที่จะให้แสงความถี่ต่ำฉายออกมากระทบกับหน้ากระดาษที่จะตรวจกวาด ส่วนที่เป็นพื้นสีดำซึ่งอาจจะเป็นข้อความหรือรูปภาพ ก็จะดูดซับแสงเอาไว้หรือสะท้อนออกมาบางส่วน และส่วนที่เป็นพื้นขาวก็จะสะท้อนออกมามากกว่า จากนั้นจะมีตัวรับสัญญาณภาพ (Photosensor) ที่เรียกว่า CCD (Charge Couple Device) ซึ่งจะประกอบไปด้วยเซลล์รับแสงเป็นจำนวนมากเรียงเป็นแถว โดยที่เซลล์แต่ละตัวจะสร้างความต่างศักย์ทางไฟฟ้าขึ้นมาเป็นสัดส่วนกับปริมาณแสงที่รับเข้ามา ค่าความต่างศักย์จะถูกแปลงให้เป็นค่าทางตัวเลข ไบนารี่ (Binary data ; 0 หรือ 1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ลักษณะข้อมูล

2.2.1 ลักษณะข้อมูลภาพกราฟิก

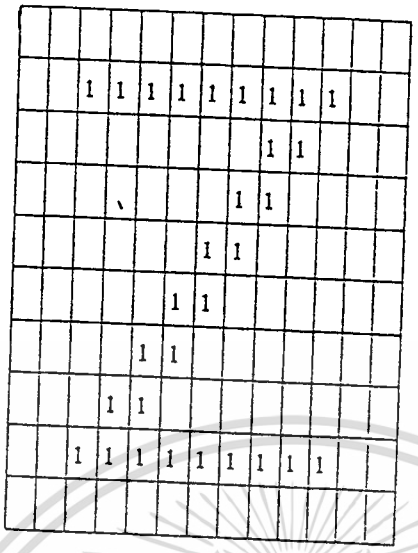
เนื่องจากข้อมูลที่จัดเก็บได้จากเครื่องตรวจกวาดภาพจะอยู่ในแฟ้มข้อมูลภาพกราฟิก นั่นคือเป็นตัวเลขไบนารี่คือ เลข 0 กับ 1 ของจุดภาพหนึ่ง ๆ ถูกย่อรวมกัน จุดเก็บอยู่ในรูปตัวเลขฐาน 16 ก่อนที่จะนำข้อมูลดังกล่าวมาป้อนให้กับเครื่องคอมพิวเตอร์ เพื่อทำการประมวลผลต่อไปจะต้องแปลงข้อมูลกราฟิกในรูปเลขฐาน 16 นี้กลับให้เป็นข้อมูลไบนารี่ในรูปตัวเลข 0 กับ 1 เสียก่อน แฟ้มข้อมูลกราฟิกที่ได้จากเครื่องกวาดตรวจภาพที่ใช้ในงานวิจัยนี้ เป็นแฟ้มข้อมูลรูปแบบ TIFF มีลักษณะเป็นบิทแมพ (Bitmap pattern) ซึ่งก็คือกลุ่มของจุดสี่เหลี่ยมอาจจะเป็นสีขาวหรือสีดำมาเรียงต่อกันเป็นตาราง ซึ่งจะต่อรวมกันแสดงออกมาเป็นรูปภาพ ตัวอย่างของรูปภาพแบบบิทแมพแสดงได้ดังรูปที่ 2.1



รูปที่ 2.1 แสดงตัวอย่างของ Bitmap pattern

Bitmap pattern ที่แสดงดังรูปที่ 2.1 สามารถที่จะแสดงให้อยู่ในรูปตัวเลข 0 กับ 1 ได้ดังรูปที่ 2.2 และ Bitmap data ซึ่งเป็นข้อมูลที่เก็บบันทึกจริงในแฟ้มข้อมูลกราฟิกจะมีลักษณะเป็นตัวเลขฐาน 16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 แสดง Bitmap pattern ในรูปของตัวเลข 0 กับ 1

2.2.2 การเปลี่ยนข้อมูลบิตแมพเป็นข้อมูลไบนารี่

ข้อมูลกราฟิกแบบ Bitmap มีอยู่ 2 ลักษณะ โดยพิจารณาจากโครงสร้างของภาพ และนำมาใช้เป็นเกณฑ์ในการเปลี่ยนข้อมูลจากข้อมูลที่อยู่ในลักษณะเลขฐาน 16 เป็นเลขฐาน 2 คือ

(1) ถ้าความกว้างเป็นผลคูณของ 8 มีเงื่อนไขของโครงสร้างดังนี้

(ก) ข้อมูลในรูปของตัวเลข 0 หรือ 1 จะถูกจับกลุ่มโดยเรียงกันไปจำนวน 8 บิท นั่นก็คือ 1 ไบท์

(ข) ลำดับของไบท์ของข้อมูลจะนับจากบนลงล่าง และซ้ายไปขวา

(ค) ลำดับของบิทในแต่ละไบท์จะนับจากซ้ายไปขวา และบิททางซ้ายมือสุด(Letmost bit) นับเป็นบิทที่ 0 และบิททางขวามือสุด (Rightmost bit) นับเป็นบิทที่ 7

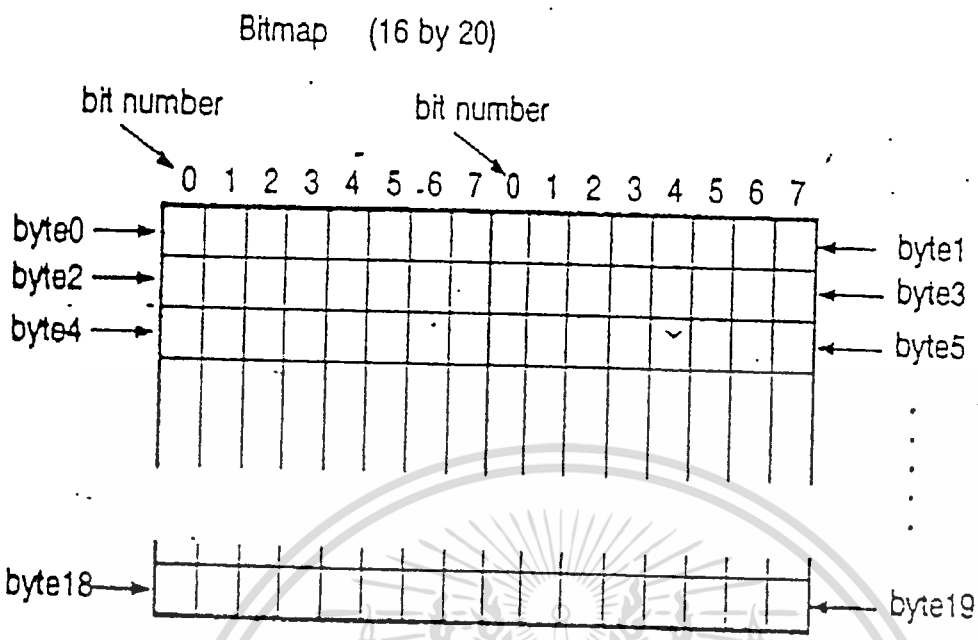
(2) ถ้าความกว้างไม่เป็นผลคูณของ 8จะมีเงื่อนไขดังนี้

(ก) จะนับความกว้างเป็นผลคูณของ 8 ที่น้อยที่สุดที่มีค่ามากกว่าความกว้างที่แท้จริงของรูป

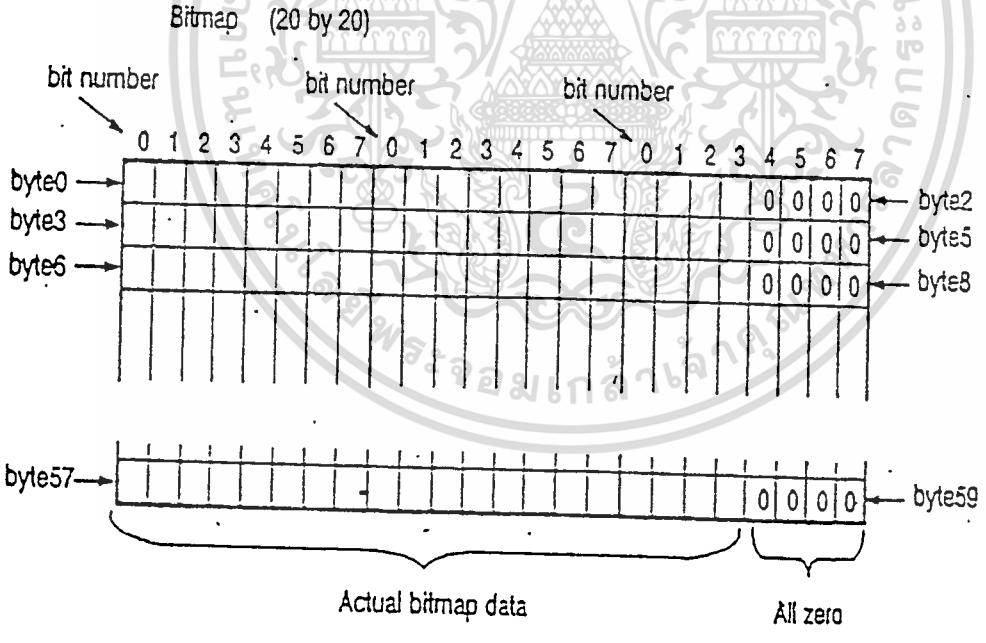
(ข) บิทที่เหลือนอกเหนือจากความกว้างที่แท้จริง จะถูกเติมด้วยเลข 0

เมื่อทราบรูปแบบของ Bitmap ก็ทำการเปลี่ยนข้อมูลภาพที่มีลักษณะเป็นเลขฐาน 16 ให้เป็นเลขฐาน 2 โดยหนึ่งบิทของเลขฐาน 2 จะแทนจุดภาพหนึ่งจุด บิทที่มีค่า เป็น 0ก็จะแทนจุดสว่างหรือจุดที่เป็นพื้นกระดาษสีขาว และบิทที่มีค่าเป็น 1 ก็จะแทนจุดดำหรือมืดทึบ ซึ่งก็คือจุดที่อาจเป็นตัวอักษรหรือจุดดำของรูปภาพ ก็จะได้ข้อมูลภาพตัวอักษรที่ถูกต้องและพร้อมที่จะส่งไปประมวลผลในขั้นตอนต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 แสดงโครงสร้างของ Bitmap ความกว้างเป็นผลคูณของ 8



รูปที่ 2.4 แสดงโครงสร้างของ Bitmap ความกว้างไม่เป็นผลคูณของ 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การเตรียมข้อมูลและขั้นตอนการจดจำตัวอักษร

ปัญหาจะเกิดขึ้นเสมอในระบบการจำแนกรูปแบบของตัวอักษรภาษาไทย คือ รูปแบบตัวอักษรใช้เป็นอินพุท (Input pattern) ของระบบมีสัญญาณมีรบกวน (Noise) ปะปนเข้ามาด้วย สัญญาณรบกวนนี้จะทำให้เกิดการผิดพลาดขึ้นในกระบวนการจำแนกตัวอักษร ดังนั้นจึงมีความจำเป็นที่จะต้องมีการปรับปรุงคุณภาพของรูปแบบตัวอักษร ก่อนที่จะนำไปจำแนกกลุ่มต่อไป จึงเรียกกระบวนการในขั้นตอนนี้ว่า ระบบจัดการล่วงหน้า (Pre-processing)

ระบบจัดการล่วงหน้าที่กล่าวถึงในบทนี้จะประกอบด้วยเทคนิคและวิธีการต่างๆหลายขั้นตอน ดังที่จะกล่าวถึงในหัวข้อต่อๆไป

3.1 การกำจัดสัญญาณรบกวน

สัญญาณรบกวนที่ปรากฏในรูปแบบของตัวอักษรที่พบ คือ สัญญาณรบกวนที่เป็น จุดอิสระ (Isolate point) และรูภายใน (Isolate hold) การกำจัดสัญญาณรบกวนทั้งสองลักษณะนี้จะใช้ตารางหน้าต่าง (Window) ขนาด 3 X 3 pixel ในรูปที่ 3.1 สแกนไปบนรูปแบบสองระดับที่มีสัญญาณรบกวน

X_3	X_2	X_1
X_4	X	X_0
X_5	X_6	X_7

รูปที่ 3.1 ตารางหน้าต่างสำหรับกำจัดสัญญาณรบกวน

เมื่อทำลหสัมพันธ์ (Correlation) ของตารางหน้าต่างในรูปที่ 3.1 กับรูปแบบของตัวอักษร และกระทำตาม Algorithm A:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Algorithm A:

สแกนไปบนรูปแบบ ทุกๆ จุดภาพ

{

ถ้า จุดภาพข้างเคียงของจุดภาพ X ทั้ง 8 จุดภาพมีค่าเป็น 0

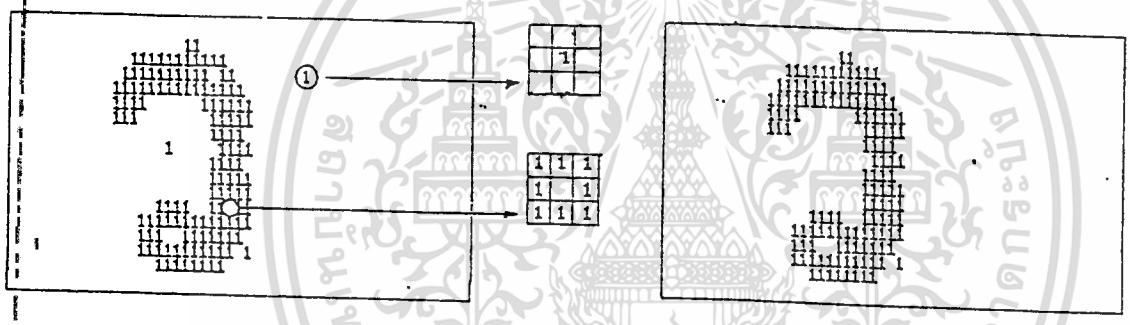
ทำให้จุดภาพ X มีค่าเป็น 0 ตาม

ถ้า จุดภาพข้างเคียงของจุดภาพ X ทั้ง 8 จุดภาพมีค่าเป็น 1

ทำให้จุดภาพ X มีค่าเป็น 1 ตาม

}

ผลที่ได้จากการทำ Algorithm A : จะทำให้สัญญาณรบกวนในรูปแบบถูกกำจัดออกไป ดังแสดงในรูปที่ 3.2



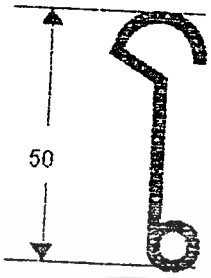
รูปที่ 3.2 การกำจัดสัญญาณรบกวนบนรูปแบบ

3.2 ขนาดมาตรฐาน

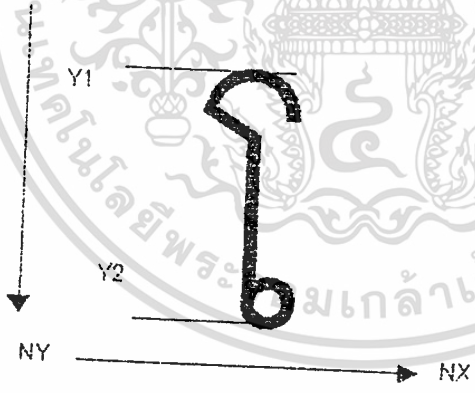
รูปแบบของตัวอักษรซึ่งถูกกำจัดสัญญาณรบกวนแล้วอาจมีขนาดที่แตกต่างกันในการเก็บข้อมูลภาพแต่ละครั้ง ดังนั้นจะต้องทำรูปแบบของตัวอักษรให้อยู่ในรูปแบบที่มีขนาดมาตรฐานเดียวกันก่อนซึ่งจะทำให้มีขนาดมาตราฐาน(สูง 50 จุดภาพ) ดังในรูปที่ 3.3

โดย การย่อรูปทำได้โดยการสแกนหากรอบ (border) ของตัวอักษรทั้ง 4 ด้าน คือ X1, X2, Y1 และ Y2 ซึ่งก็คือกรอบบน ล่าง ซ้าย ขวา ของตัวอักษร ดังแสดงในรูปที่ 3.4 และทิศของการสแกนทั้ง 4 ด้าน แสดงในรูปที่ 3.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

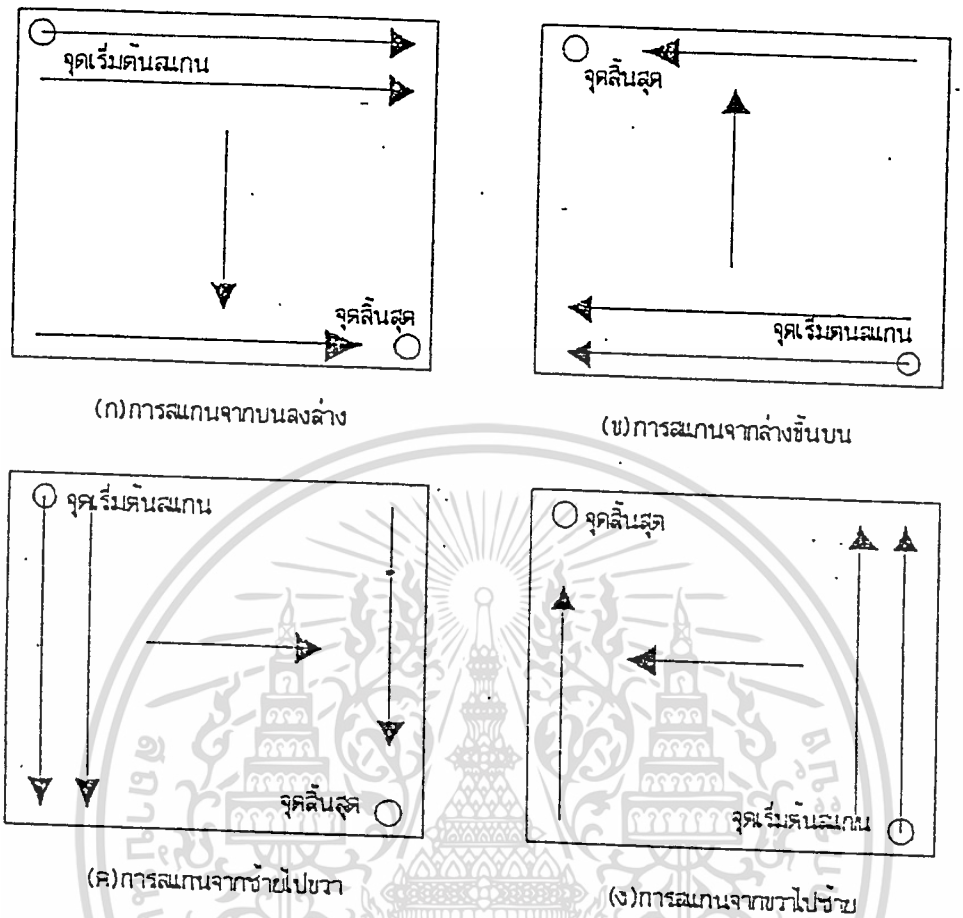


รูปที่ 3.3 ตัวอย่างรูปตัวอักษรที่ถูกทำให้เป็นขนาดมาตราฐานแล้ว



รูปที่ 3.4 แสดงกรอบตัวอักษรที่จะทำการย่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 ทิศทางการสแกนทั้ง 4 แบบ

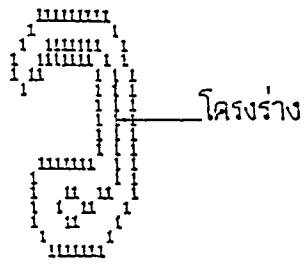
ในกรอบนี้จะนำค่าความกว้าง และความสูงของกรอบมาย่อเป็นขนาดมาตราฐาน โดยด้วยอัตราส่วน

$$Ratio = (Y2-Y1) / 50 \qquad (3.1)$$

3.3 การทำตัวอักษรให้บาง (Thinning)

รูปแบบสองระดับที่ได้มาจากอุปกรณ์รับแสง เช่น OCR หรือ Scanner หรือแม้กระทั่งกล้องทีวีก็ตามจะมีภาพของตัวอักษรในรูปแบบที่มีขนาดความกว้างมากกว่า 1 จุดภาพเสมอ แต่ในระบบการจดจำที่ใช้คุณสมบัติทางโทโพโลยีของรูปแบบเป็นตัวจำแนกกลุ่มนั้น จะต้องการเฉพาะส่วนที่เป็นโครงร่าง (Skeleton) ของตัวอักษรเท่านั้น โครงร่างของตัวอักษรก็คือภาพของตัวอักษรที่มีคววามกว้างของตัวอักษรเพียง 1 จุดภาพและอยู่ในตำแหน่งที่สมมาตรบนเนื้อของตัวอักษรนั้น ดังแสดงในรูปที่ 3.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 โครงร่างของตัวอักษร

n_3	n_2	n_1
n_4	P	n_0
n_5	n_6	n_7

รูปที่ 3.7 ตารางหน้าต่างสำหรับทำให้บาง

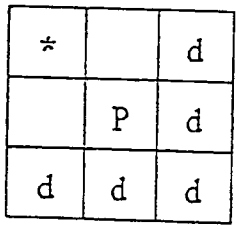
ตารางหน้าต่างในรูปที่ 3.7 นำมาพิจารณาส่วนของภาพตัวอักษรบนรูปแบบที่ต้องการทำให้บาง โดยที่จุดข้างเคียงของจุดภาพ P ซึ่งก็คือจุดภาพ $n_0, n_1, n_2, \ldots\ldots n_7$ จะเป็นตัวกำหนดลักษณะขอบ (Edge) ของจุดภาพ P ใน 4 ลักษณะคือ

1. จุดภาพ P เป็นขอบซ้ายถ้ามี n_4 เป็นจุดสว่าง ($n_4 = 0$)
2. จุดภาพ P เป็นขอบขวาถ้ามี n_0 เป็นจุดสว่าง ($n_0 = 0$)
3. จุดภาพ P เป็นขอบบนถ้ามี n_2 เป็นจุดสว่าง ($n_2 = 0$)
4. จุดภาพ P เป็นขอบล่างถ้ามี n_6 เป็นจุดสว่าง ($n_6 = 0$)

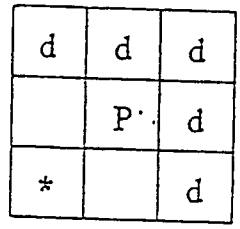
ในขบวนการทำให้บางถือว่าขอบของภาพไม่ใช่โครงร่างของตัวอักษร ดังนั้น เมื่อพบขอบก็จะกำหนดค่าของจุดภาพนั้นให้เป็นค่า ๆ หนึ่ง ที่ไม่เป็น 1 (Flagging) แต่ถ้าพบจุดภาพที่มีความกว้างเพียง 1 จุดภาพก็จะกำหนดค่าของจุดภาพนั้นให้เป็นค่า 1 (Unflagging)

การทดสอบจุดภาพที่เป็นขอบ กระทำโดยการเปรียบเทียบตารางหน้าต่างในรูปที่ 3.7 ที่สแกนจากบนลงล่างบนรูปแบบตัวอักษร กับตารางหน้าต่างในรูปที่ 3.8

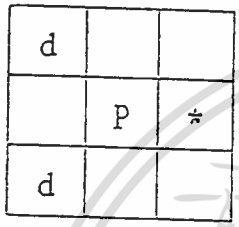
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



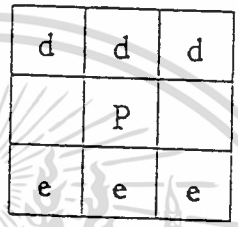
(ก)



(ข)



(ค)



(ง)

รูปที่ 3.8 ตารางหน้าต่างสำหรับการตรวจสอบหาขอบ

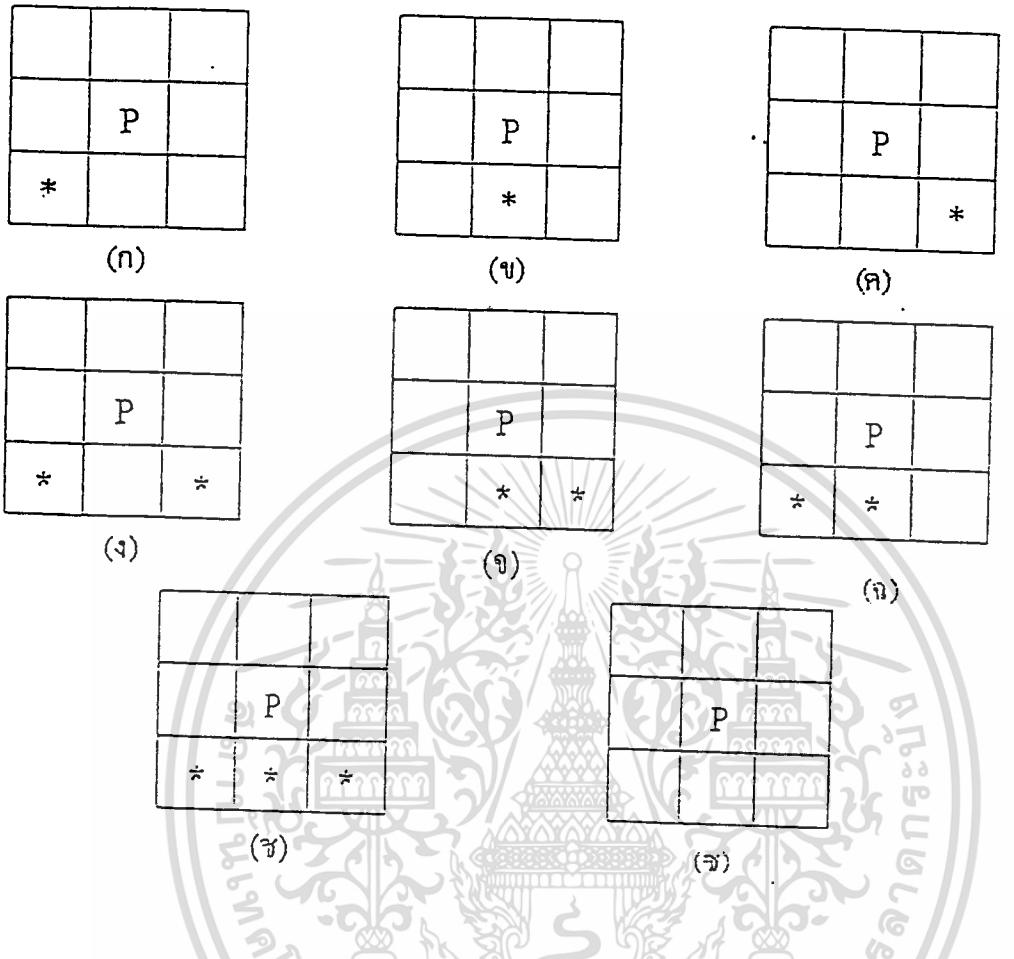
จากรูปที่ 3.8 P และ * เป็นจุดมีคชิค่าเป็น 1 ส่วน d และ e เป็นค่าที่ไม่สนใจ (don't care) เมื่อพิจารณาที่ 3.11 ก) , ข) , ค) จะแบ่งการพิจารณาเป็น 2 ลักษณะคือ

1 ถ้า d ทุกตัวเป็นจุดว่าง (มีค่าเป็น 0) จะได้ว่า P คือจุดภาพที่เป็นจุดปลาย (end point) ของโครงร่าง

2 ถ้า d อย่างน้อย 1 จุดเป็นจุดมืด (มีค่าเป็น 1) จะได้ว่า P คือจุดภาพที่เป็น โครงร่างของตัวอักษร จุด P ใดๆ ที่มีคุณสมบัติตาม 2 ลักษณะดังกล่าวจะถูกกำหนดค่าให้เป็น 1 (Unflaged)

ตารางหน้าต่างในรูปที่ 3.8 จ) จะมีการพิจารณาที่ซับซ้อนขึ้น โดยแยกการพิจรณาออกมาต่างหากแสดงโดยตารางหน้าต่างในรูปที่ 3.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 ตารางหน้าต่างของจุดภาพโครงร่าง

จากรูปที่ 3.8 ง) ถ้า d และ e อย่างน้อย 1 จุดเป็นจุดดำนิดจะกำหนดให้จุด P เป็นจุดโครงร่าง ในกรณีนอกเหนือจากนั้นจะพิจารณาตามตารางหน้าต่างในรูปที่ 3.9

จากรูปที่ 3.9 ตารางหน้าต่าง ก), ข), ค) จะกำหนดให้จุด P เป็นจุดปลายของโครงร่าง ตารางหน้าต่าง ง) กำหนดให้ P เป็นจุดโครงร่าง ตารางหน้าต่าง จ) และ ฉ) กำหนดให้ P เป็นจุดที่ทำให้เกิดความกว้างขนาด 2 จุดในแนวเอียง ตารางหน้าต่าง ช)กำหนดให้จุด P เป็นกิ่งสั้น (Protrusion) ของโครงร่าง ส่วนตารางหน้าต่าง ซ) กำหนดให้จุด P เป็นจุดอิสระที่เกิดขึ้น

เพื่อลดความยุ่งยากในการพิจารณาเงื่อนไขของขอบและโครงร่างลง จึงได้เอาเงื่อนไขทั้งหมดของตารางขอบเขตมาเขียนเป็นนิพจน์ทางตรรก ทำให้ง่ายต่อการเขียนโปรแกรมยิ่งขึ้น ดังแสดงในลักษณะต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขอบขวา : $B0 = N4*(N2+N3+N5+N6)*(N6+\overline{N7})*(\overline{N1}+N2)$ (3.2)

ขอบล่าง : $B6 = N2*(N6+N1+N3+N4)*(N4+\overline{N5})*(\overline{N0}+N7)$ (3.3)

ขอบซ้าย : $B4 = N0*(N1+N2+N6+N7)*(N2+\overline{N3})*(\overline{N5}+N6)$ (3.4)

ขอบบน : $B2 = N6*(N0+N4+N5+N7)*(N0+\overline{N1})*(\overline{N3}+N4)$ (3.5)

เมื่อ * คือ operator AND

+ คือ operator OR

- คือ operator NOT

และ B4 , B0 , B2 , B6 เป็นค่าของจุดภาพ P ที่มีลักษณะเป็นขอบซ้าย ขอบขวา ขอบบน และขอบล่างตามลำดับ

โดยการสแกนไปบนรูปแบบลองระดับที่ต้องการทำให้บาง ทำการสแกนใน 2 ทิศทาง คือ สแกนจากบนลงล่าง และ สแกนจากซ้ายไปขวา (รูปที่ 3.5 (ก) และ (ค)) ในทิศทางแรกจะพิจารณาขอบ B4 และ B0 และในทิศทางที่สองจะพิจารณาขอบ B2 และ B6 ตามลำดับ

ด้วยการกระทำ แบบวนรอบ (Iteration) จนกระทั่งเหลือแต่จุดภาพที่เป็นโครงร่างก็จะได้ตัวอักษรที่บางออกมาดังแสดงในรูปที่ 3.10

ขั้นตอนการทำให้บาง

เมื่อจุดที่ถูกพิจารณาเป็น1

1.สแกนจากบนลงล่าง(หรือล่างขึ้นบน)

พิจารณาที่ เมื่อ n4 เป็น 0 และ B4 เป็น 1 หรือ

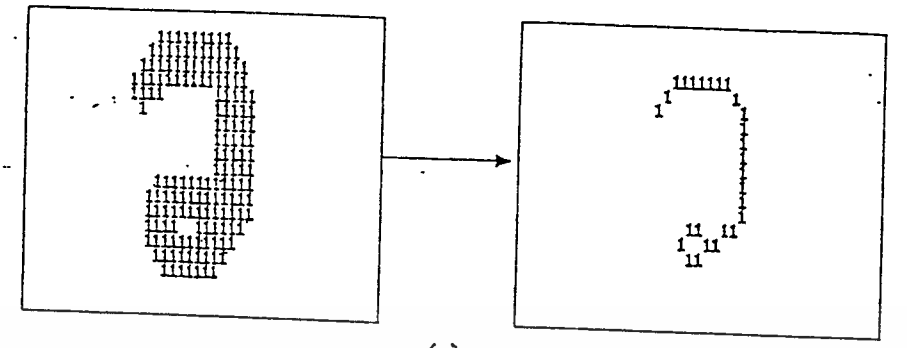
เมื่อ n0 เป็น 0 และ B0 เป็น 1 จะทำการตั้งค่าจุดนั้นให้เป็น 0

2.สแกนจากซ้ายไปขวา(หรือบนลงล่าง)

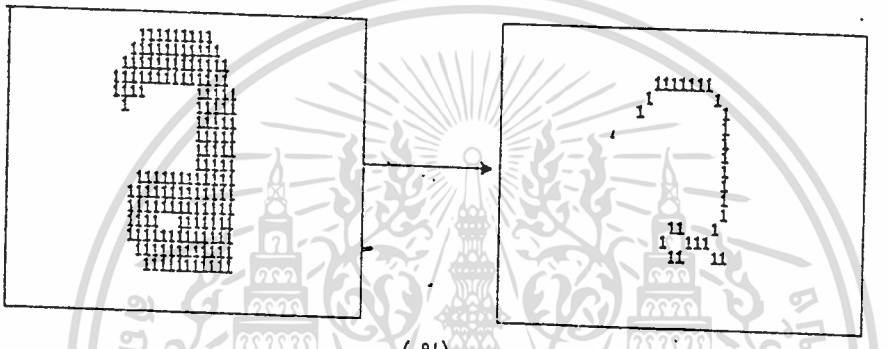
พิจารณาที่ เมื่อ n2 เป็น 0 และ B2 เป็น 1 หรือ

เมื่อ n6 เป็น 0 และ B6 เป็น 1 จะทำการตั้งค่าจุดนั้นให้เป็น 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก)



(ข)

(ก) โครงร่างที่สมบูรณ์ (ข) โครงร่างที่มีส่วนเกิน

รูปที่ 3.10 โครงร่างของตัวอักษร

โครงร่างของตัวอักษรได้จากการทำให้บางจะเก็บ เป็นรูปแบบที่จะนำไปหาค่าคุณสมบัติทาง โท โพโลยีต่อไป

ข้อดีของการทำให้บางก็คือจะลดจำนวนข้อมูลที่เป็นภาพตัวอักษรลงไปได้จำนวนมาก ทำให้ลดเวลาของการประมวลผลในขั้นตอนต่อไปได้ แต่ทั้งนี้การทำให้บางก็อาจจะใช้เวลาในการประมวลผลมากถ้ารูปแบบสองระดับที่ป้อนเข้ามามีขนาดความหนามาก ๆ

3.4 การคำนวณค่าตัวเลขต่อเนื่อง

ค่าตัวเลขต่อเนื่องเป็นคุณสมบัติทาง Topology ของจุดภาพที่อยู่ติดกันในรูปแบบ 2 ระดับ ค่าตัวเลขต่อเนื่องนี้จะเป็นตัวแสดงความสัมพันธ์ระหว่างจุดภาพข้างเคียง จุดภาพข้างเคียงบนรูปแบบสองระดับใด ๆ ใช้สมการดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

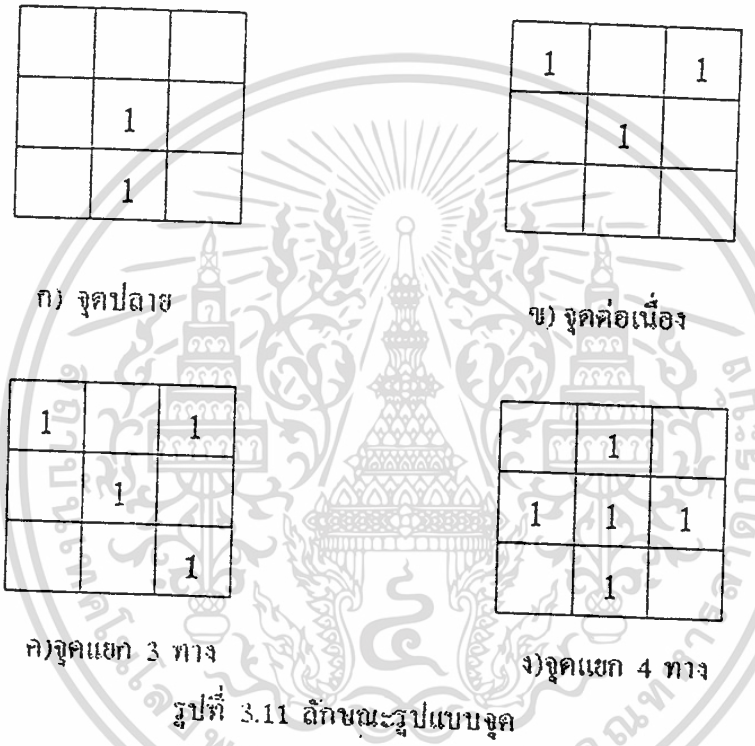
$$Num = (N0*\overline{N1})+(N1*\overline{N2})+(N2*\overline{N3})+(N3*\overline{N4})+(N4*\overline{N5})+(N5*\overline{N6})+(N6*\overline{N7})+(N7*\overline{N0}) \quad (3.6)$$

* = AND

+ = ADD

(Num = 1) : จุดปลาย; (Num = 2) : จุดต่อเนื่อง;

(Num = 3) : จุดแยก 3 ทาง; (Num =41) : จุดแยก 4 ทาง;



รูปที่ 3.11 ลักษณะรูปแบบจุด

3.5 การกำจัดส่วนเกิน (Elimination of Redundancy)

โครงร่างของภาพตัวอักษรที่ได้จากการทำให้บางแล้ว ส่วนมากจะมีส่วนเกินที่เป็นกิ่ง (Redundant bits) ปรากฏออกมาด้วย ดังจะเห็นได้จากรูปที่ 3.10 (ข) ส่วนเกินนี้จะทำให้เกิดความผิดพลาดขึ้นต่อระบบการจำแนกกลุ่มของตัวอักษรในภายภาคหลัง ดังนั้นจึงต้องกำจัดส่วนเกินส่วนนี้ออกไปจากรูปแบบ

วิธีการกำจัดส่วนเกิน จะต้องมีการคำนวณค่าตัวเลขต่อเนื่อง ในสมการ 3.6 และพิจารณาค่าตัวเลขต่อเนื่องจากตารางที่ 3.1

กำหนดให้

$$L = \{ l_{ij} \mid i = 1,2,\ldots,m ;\ j = 1,2,\ldots,n \}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

L เป็นรูปแบบสองระดับขนาด m x n ที่เก็บ โครงร่างของตัวอักษรที่มีส่วนเกินปรากฏอยู่ (ในที่นี้ m = 64 และ n = 85)

กำหนดให้

$M = \{ m_{i,j} \mid i = 1,2,.....,m ; j = 1,2,.....,n \}$

$N = \{ n_{i,j} \mid i = 1,2,.....,m ; j = 1,2,.....,n \}$

M และ N เป็น accumulated pattern ที่มีค่าจุดภาพเริ่มต้นทุกๆ จุดเป็น 0 และกำหนดให้ค่าเริ่มต้นของจุดภาพที่เป็นส่วนเกินคือ จำนวนเต็มบวก K (ในที่นี้ให้ K=5)

X_3	X_2	X_1
X_4	X	X_0
X_5	X_6	X_7

รูปที่ 3.12 ตารางหน้าต่างสำหรับการกำจัดส่วนเกิน

จากการใช้ตารางหน้าต่างในรูปที่ 3.12 สแกนไปบนรูปแบบสองระดับ L ในสองทิศทางคือ สแกนจากบนลงล่าง และ สแกนจากซ้ายไปขวา และพิจารณาการกระทำเป็น 4 ขั้นตอน คือ

ขั้นตอนที่ 1

กระทำแบบขนาน (parallel operation) ต่อจุดภาพที่มีค่า 1 ของรูปแบบ L ในทิศทางสแกนจากบนลงล่าง จำนวน K ครั้ง ดังนี้

if ($NC_s = 1$)

{ $l_{i,j} = 0$; $m_{i,j} = 1$ }

else

if ($NC_s = 2$ || $NC_s = 3$)

{ $m_{i,j} = 2$ }

ขั้นตอนที่ 2

กระทำแบบอนุกรม (Series Operation) ต่อจุดภาพที่มีค่า 1 ของรูปแบบ M ในทิศทางสแกนจากซ้ายไปขวา (ดังแสดงในรูป 3.5 (ค)) จำนวน K ครั้งดังนี้

if (X1 or X2 or X3 or X4 or X5 or X6 or X7 = 2)

{ $m_{i,j} = 2$ }

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

34077

ขั้นตอนที่ 3

กระทำแบบอนุกรมต่อจุดภาพที่มีค่า 2 ของรูปแบบ M ทิศทาง สแกนจากบนลงล่าง ดังนี้

if $(m_{i,j} = 2)$

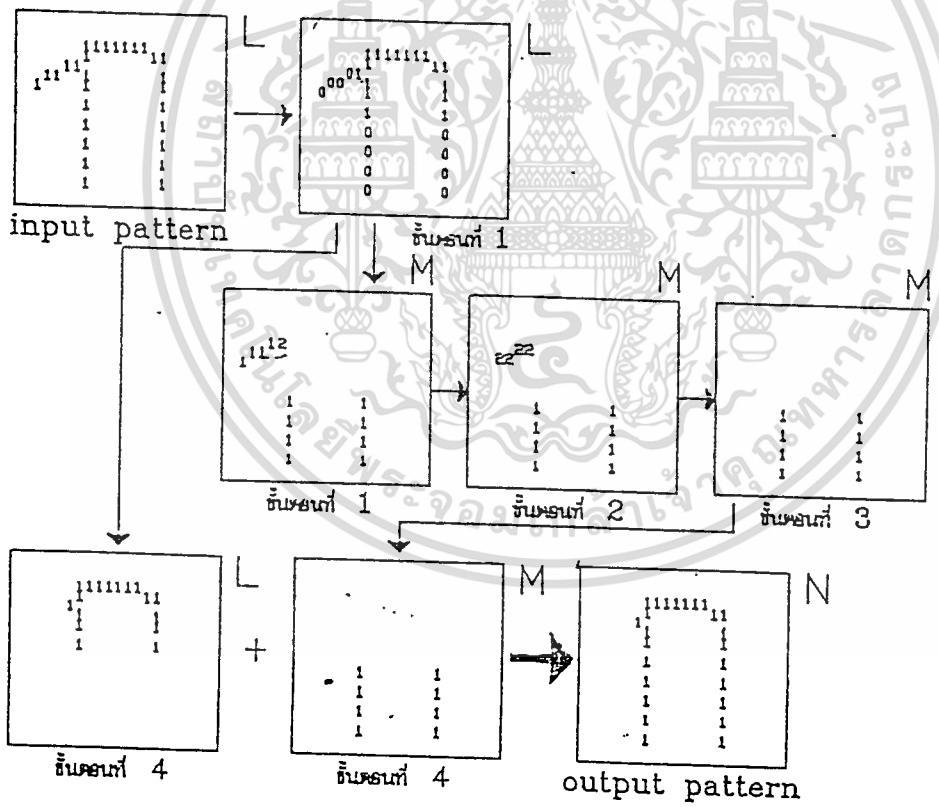
$\{ m_{i,j} = 0 \}$

ขั้นตอนที่ 4

รวมเอารูปแบบ M และรูปแบบ L เข้าด้วยกันและเก็บค่าไว้ในรูปแบบ N ดังนี้

$n_{i,j} = m_{i,j} + l_{i,j}$

รูปแบบฉลองระดับ N ที่ได้จากขั้นตอนที่ 4 จะเป็นโครงร่างของตัวอักษรที่ถูกกำจัดเอาส่วนเกินออกไปแล้ว บล็อกไดอะแกรมของการกำจัดส่วนเกินแสดงไว้ในรูปที่ 3.13



รูปที่ 3.13 บล็อกไดอะแกรมของการกำจัดส่วนเกิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

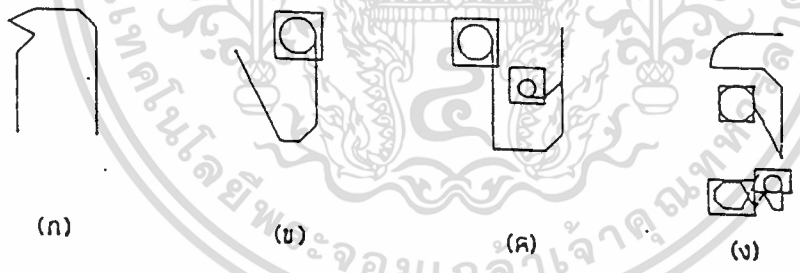
บทที่ 4

การจำแนกตัวอักษร

ภาพของตัวอักษรแต่ละตัวที่ผ่านเข้ามาในระบบการจดจำ จะถูกนำไปเปลี่ยนให้เป็นรูปแบบสองระดับของตัวอักษรด้วยวิธีการดังที่ได้กล่าวไว้แล้วในบทที่ 2 รูปแบบสองระดับที่ได้นี้จะนำไปลอกแบบออกเป็น 2 รูปแบบ รูปแบบของตัวอักษรส่วนที่หนึ่งใช้เป็นรูปแบบสำหรับการตรวจสอบหาหัวของตัวอักษร และรูปแบบของตัวอักษรในส่วนที่สองใช้เป็นรูปแบบสำหรับระบบการจัดการล่วงหน้า ซึ่งระบบการจัดการล่วงหน้านี้ได้แสดงถึงรายละเอียดไว้แล้วในบทที่ 3 ดังนั้นเนื้อหาในบทนี้ก็จะได้กล่าวถึงส่วนที่มีความสำคัญที่สุดของระบบการจำแนกตัวอักษร (Classification) ออกเป็นกลุ่ม ๆ

4.1 การตรวจสอบหัวของตัวอักษร (Head detection)

หัวของตัวอักษรในภาษาไทย จะมีลักษณะเป็นวงปิดขนาดเล็ก (เมื่อเทียบกับส่วนอื่น ๆ ในตัวอักษร) และมีจำนวนต่าง ๆ กันออกไป สำหรับรูปแบบของตัวอักษรแต่ละตัวดังแสดงในรูปที่ 4.1

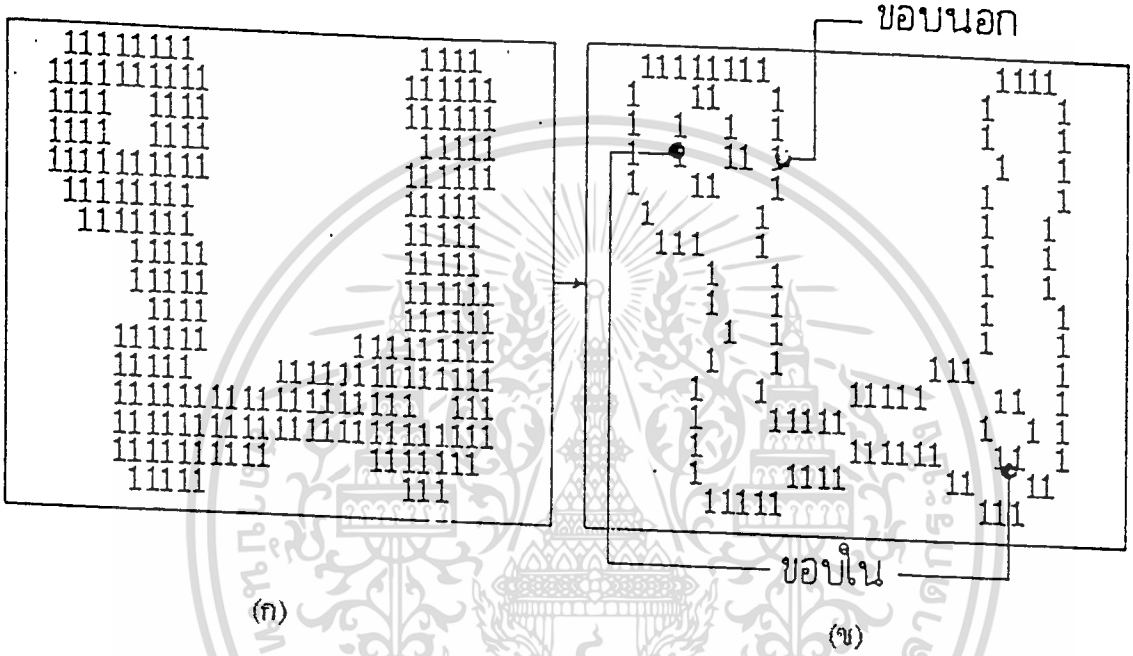


รูปที่ 4.1 จำนวนและตำแหน่งของหัว
(ก) ตัวอักษรที่ไม่มีหัว (ข) ตัวอักษรที่มีหัว 1 หัว
(ค) ตัวอักษรที่มีหัว 2 หัว (ง) ตัวอักษรที่มีหัว 3 หัว

จะพบว่าตัวอักษรในภาษาไทยทั้งหมดสามารถที่จะแบ่งได้เป็น ตัวอักษรที่ไม่มีหัว มีหัว 1 หัว มีหัว 2 หัว และมีหัว 3 หัว ดังแสดงในรูปที่ 4.1 (ก), (ข), (ค) และ (ง) ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

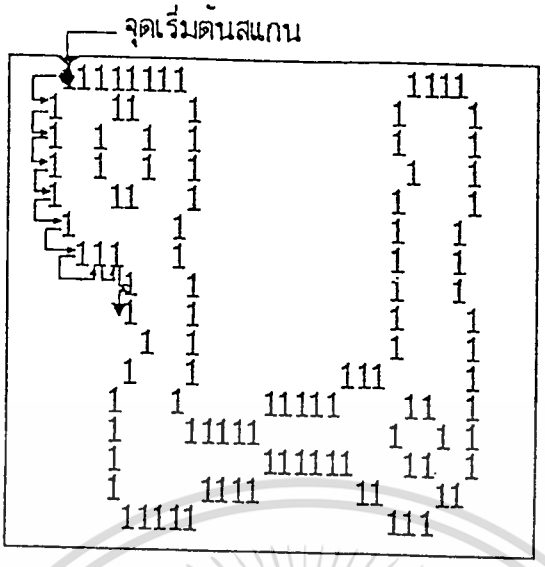
การตรวจสอบหัวของตัวอักษรจะเริ่มต้นโดยการทำ สแกนจากบนลงล่าง (รูปที่ 3.5 (ก)) ไปบนรูปแบบสองระดับที่ยังไม่ได้มีการทำให้บาง แล้วคำนวณหาค่าตัวเลขต่อเนื่องของจุดภาพใด ๆ บนรูปแบบและทำการเก็บค่าไว้ในรูปแบบ ก็จะทำให้สามารถหาส่วนที่เป็นขอบใน และ ขอบนอกของตัวอักษรออกมาได้ดังรูปที่ 4.4



รูปที่ 4.4 ขอบของตัวอักษร

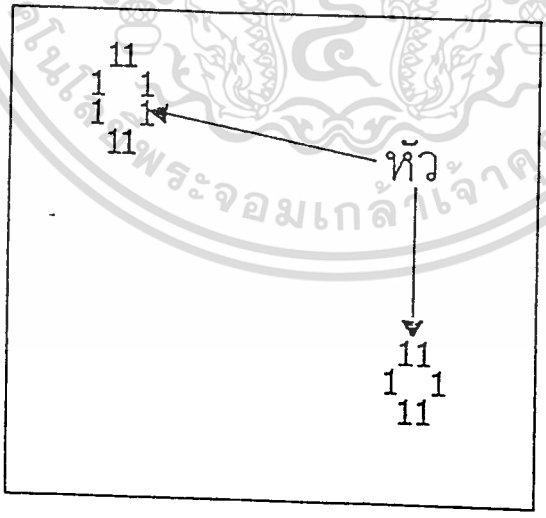
ขั้นตอนต่อไปก็จะทำการสแกนจากบนลงล่าง บนรูปแบบที่เป็นขอบของภาพ (รูปที่4.4(ข)) จนกระทั่งพบจุดภาพที่เป็นขอบจุดแรก จะใช้จุดภาพนี้เป็นจุดเริ่มต้นของกการติดตามของเขตของตัวอักษร ดังรูปที่ 4.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 การติดตามของภาพเพื่อการตรวจสอบหัว

ในขณะที่ทำการติดตามขอบเขตของภาพอยู่นี้ ก็จะบันทึกเอาตำแหน่งและจำนวนของจุดภาพไว้ และเมื่อการติดตามขอบเขตของภาพสิ้นสุดลงที่จุดเริ่มต้น ก็จะใส่ค่า 0 ลงไปในตำแหน่งของจุดภาพที่ได้บันทึกไว้(ตัดขอบนอกออกไปจากรูปแบบ) จะได้รูปแบบที่เป็นขอบภาพตัวอักษรออกมาดังรูปที่ 4.6



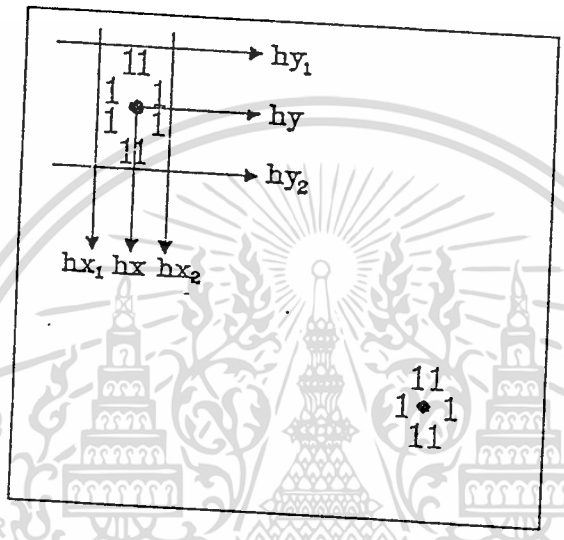
รูปที่ 4.6 ขอบของภาพตัวอักษรที่เป็นหัว

เมื่อถึงขั้นตอนนี้จะพบว่าหัวของตัวอักษรก็จะปรากฏออกมาดังแสดงในรูปที่ 4.6

เอกสารนี้ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้ากำหนดให้ H เป็นจำนวนสูงสุดของหัวในตัวอักษรภาษาไทยแล้ว (ในที่นี้ H = 3) ขั้นตอนต่อไปก็จะทำการคิดตามขอบเขตที่เป็นหัวของตัวอักษร โดยกระทำเป็นจำนวน H ครั้ง ซึ่งในแต่ละครั้งก็จะทำการบันทึกเอาตำแหน่งที่ใช้แทนหัวและจำนวนจุดภาพของแต่ละหัวไว้

ตำแหน่งที่ใช้แทนหัวจะหาได้จากการทำสแกนภาพใน 4 ทิศทาง (รูปที่ 3.5) เพื่อหากรอบ บน ล่าง ซ้าย และ ขวาของหัวดังรูปที่ 4.7



รูปที่ 4.7 การหาตำแหน่งของหัว

ตำแหน่งที่ใช้แทนหัวซึ่งต่อไปจะเรียกว่าตำแหน่งของหัว จะหาได้จากสมการที่ (4.1) และ (4.2)

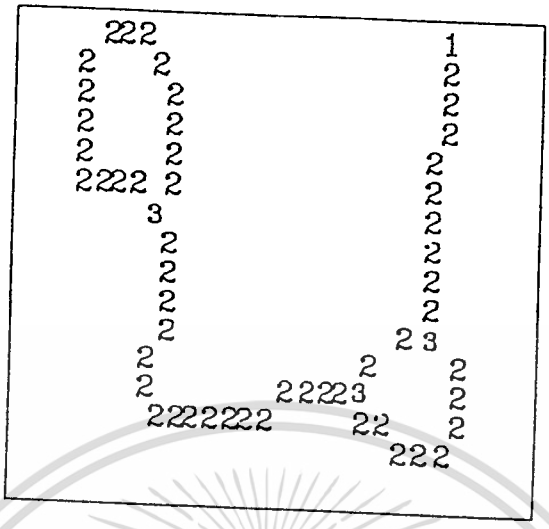
$$hx = (hx_2 - hx_1)/2 \quad (4.1)$$

$$hy = (hy_2 - hy_1)/2 \quad (4.2)$$

เมื่อนำเอารูปแบบสองระดับของตัวอักษรที่ผ่านการเตรียมข้อมูลแล้วมาหาค่าตัวเลขต่อเนื่องก็จะได้รูปแบบตัวอักษรที่ประกอบด้วยค่าตัวเลขต่อเนื่องดังรูปที่ 4.8

ดังนั้นจะเห็นได้ว่ารูปแบบสองระดับสำหรับใช้ตรวจสอบหัว และรูปแบบสองระดับสำหรับการเตรียมข้อมูลจะกระทำแยกจากกัน โดยจะกระทำกระบวนการใดก่อนก็ได้

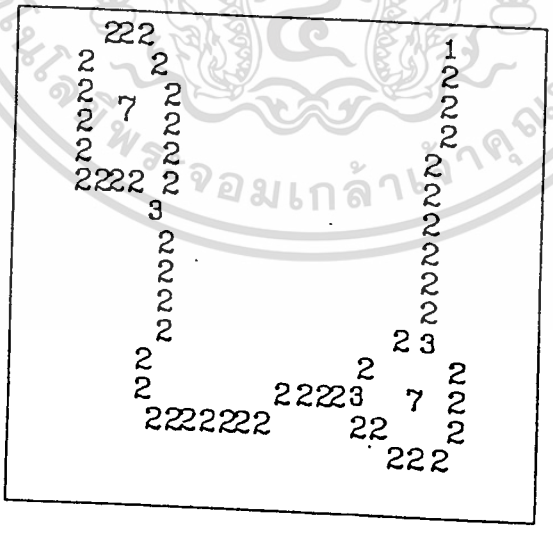
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่4.8 รูปแบบที่มีค่าตัวเลขต่อเนื่อง

1 คือ จุดปลาย : 2 คือ จุดต่อเนื่อง : 3 คือ จุดแยก

รวมเอาตำแหน่งของหัวที่คำนวณได้จาก สมการ (4.1) และ (4.2) เข้ากับรูปแบบที่มีค่าตัวเลขต่อเนื่องของตัวอักษร ก็จะได้รูปแบบของตัวอักษรมาดังรูปที่ 4.9



รูปที่ 4.9 รูปแบบตัวอักษรสำหรับการจำแนกกลุ่ม

1 คือ จุดปลาย : 2 คือ จุดต่อเนื่อง : 3 คือ จุดแยก : 7 คือ จุดหัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบของตัวอักษรในรูปที่ 4.9 นี้จะเป็นรูปแบบที่มีคุณสมบัติต่าง ๆ ครบถ้วนพร้อมที่จะนำไปจำแนกกลุ่มต่อไป

4.2 การแบ่งกลุ่มของตัวอักษรแบบหยาบ (Rought Classification)

จากกรรมวิธีการตรวจสอบหัวของตัวอักษรในหัวข้อที่ผ่านมา ก็จะสามารถหาจำนวนและตำแหน่งในตัวอักษรแต่ละตัวที่ผ่านเข้ามาในระบบการจดจำ โดยจะแบ่งตัวอักษรออกเป็น 4 กลุ่ม (class) ตามจำนวนหัวดังนี้

กลุ่มที่ 1 (class1) คือกลุ่มที่ไม่มีหัว (No head) เช่น ตัวอักษร ก ธ
กลุ่มที่ 2 (class2) คือกลุ่มที่มีหัว 1 หัว (One head) เช่น ตัวอักษร บ ย ภ
กลุ่มที่ 3 (class3) คือกลุ่มที่มีหัว 2 หัว (Two head) เช่น ตัวอักษร ฉ ม ษ
กลุ่มที่ 4 (class4) คือกลุ่มที่มีหัว 3 หัว (Three head) เช่น ตัวอักษร ฐ ล

รูปแบบตัวอักษรในกลุ่มที่ 1 และกลุ่มที่ 3 จะถูกนำมาคำนวณหาจุดศูนย์ถ่วงของตัวอักษร เพื่อที่จะใช้จุดนี้แบ่งตัวอักษรออกเป็น 4 เขต (Zone) ดังรูปที่ 4.10

รูปที่ 4.10 การแบ่งเขตของตัวอักษรกลุ่มที่ 1 และ กลุ่มที่ 3

เมื่อได้ทราบตำแหน่งหัวของตัวอักษรใน 4 เขตที่แบ่งได้ ก็จะสามารถแบ่งกลุ่มย่อย (Sub Class) ของตัวอักษรได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่มย่อย C11 คือกลุ่มย่อยที่มีหัวอยู่ที่ในเขต 1 ทั้ง 2 หัว

กลุ่มย่อย C12 คือกลุ่มย่อยที่มีหัวอยู่ที่ในเขต 1 และ เขต 2

กลุ่มย่อย C13 คือกลุ่มย่อยที่มีหัวอยู่ที่ในเขต 1 และ เขต 3

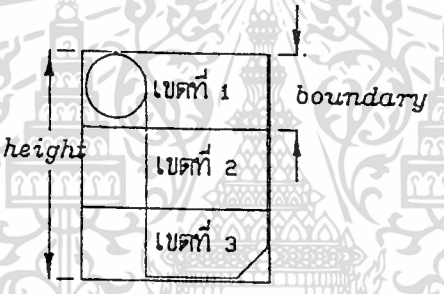
กลุ่มย่อย C14 คือกลุ่มย่อยที่มีหัวอยู่ที่ในเขต 1 และ เขต 4

กลุ่มย่อย C23 คือกลุ่มย่อยที่มีหัวอยู่ที่ในเขต 2 และ เขต 3

กลุ่มย่อย C33 คือกลุ่มย่อยที่มีหัวอยู่ที่ในเขต 3 ทั้ง 2 หัว

กลุ่มย่อย C34 คือกลุ่มย่อยที่มีหัวอยู่ที่ในเขต 3 และ เขต 4

สำหรับตัวอักษรในกลุ่มที่ 2 จะแบ่งเป็น 3 เขตดังรูปที่ 4.11



รูปที่ 4.11 การแบ่งเขตของตัวอักษรกลุ่มที่ 2

โดยที่ความกว้างของแต่ละเขตจะหาได้จาก

boundary = height / 3

จากการสังเกตหัวของตัวอักษรในกลุ่มที่ 2 นี้จะสามารถแบ่งตัวอักษรออกเป็นกลุ่มย่อยตามตำแหน่งของหัวในเขตที่แบ่งได้เป็น 3 กลุ่มย่อยดังนี้

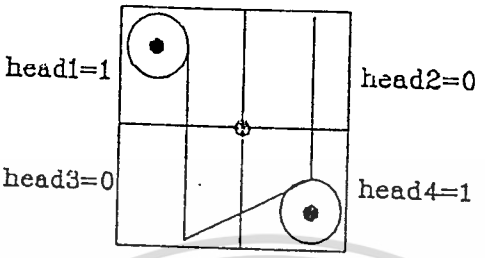
กลุ่มย่อย H1 (Upper Head) คือกลุ่มย่อยที่มีหัวอยู่ในเขต 1

กลุ่มย่อย H2 (Middle Head) คือกลุ่มย่อยที่มีหัวอยู่ในเขต 2

กลุ่มย่อย H3 (Lower Head) คือกลุ่มย่อยที่มีหัวอยู่ในเขต 3

ตัวอักษรในกลุ่มย่อย H1, H2 และ H3 นี้ยังจะนำไปแบ่งเขตออกเป็น 4 เขต ด้วยวิธีการเช่นเดียวกับการแบ่งเขตหัวอักษร ในกลุ่มที่ 1 และกลุ่มที่ 3 เพื่อประโยชน์ในการจำแนกกลุ่มของตัวอักษรอย่างละเอียดต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



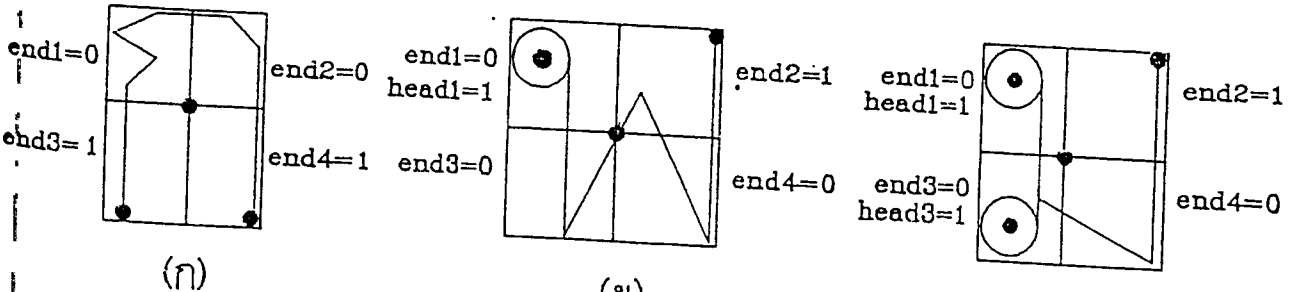
รูปที่ 4.12 ตำแหน่งหัวของตัวอักษรในกลุ่มย่อย

รูปที่ 4.12 แสดงถึงตำแหน่งของหัว ที่ปรากฏอยู่ตามเซคต่างๆ head 1 = 1 และ head 4 = 1 มีความหมายว่า ตัวอักษร "บ" มีหัวอยู่ในเซคที่ 1 และเซคที่ 4 ตามลำดับ ส่วน head 2 = 0 และ head 3 = 0 ก็หมายความว่าไม่มีหัวอยู่ในเซคที่ 2 และเซคที่ 3 ตามลำดับ

ส่วนตัวอักษรในกลุ่มที่4 นั้นมีเพียงตัวเดียวคือ "ฐ" ดังนั้นจึงไม่กระทำการใดๆ กับตัวอักษรในกลุ่มนี้ เพราะถือว่าเป็นผลลัพธ์ขั้นสุดท้ายของระบบการจดจำแล้ว

4.3 การแบ่งกลุ่มของตัวอักษรแบบละเอียด (Fine classification)

ตัวอักษรในกลุ่มย่อยที่ได้จากหัวข้อ 4.2 จะถูกนำมาแยกและความแตกต่างของแต่ละตัวอักษรในหัวข้อนี้ ด้วยการพิจารณาถึงตำแหน่งจุดปลาย (end point) ของตัวอักษรในแต่ละเซคที่แบ่ง ดังรูปที่ 4.3



รูปที่ 4.13 ตำแหน่งของจุดปลาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 4.13 แสดงถึงตำแหน่งของจุดปลายที่อยู่ในเขตต่างๆ ส่วน end3 = 1 และ end4 = 1 ในรูปที่ 4.13 (ก) นั้นหมายความว่า ตัวอักษร "ก" มีจุดปลายอยู่ในเขตที่ 3 และเขตที่ 4 ตามลำดับ เช่นเดียวกัน end1 = 0 และ end 2 = 0 ก็หมายความว่าไม่มีจุดปลายอยู่ในเขตที่ 1 และเขตที่ 2 ตามลำดับ

สำหรับตัวอักษรในรูปที่ 4.13 (ข) และ (ค) ก็จะมีการพิจารณาในลักษณะเดียวกัน

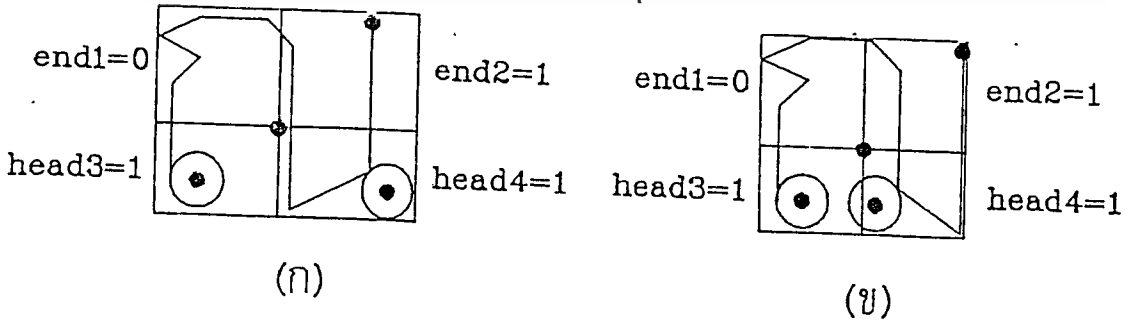
4.4 เทคนิคและวิธีการแยกตัวอักษรที่คล้ายกัน

ในกลุ่มย่อยของตัวอักษรหลายๆ กลุ่ม จะมีตัวอักษรที่คล้ายกันอยู่หลายตัวเช่น ในกลุ่มย่อยของตัวอักษร "ข" หรือ "ฆ" กับ "ฌ" กับ "ฉ" เป็นต้น ดังนั้นจึงได้นำเสนอถึงเทคนิคและวิธีการหลายๆ แบบ ในการแยกตัวอักษรแต่ละตัวออกจากกัน เพื่อเพิ่มประสิทธิภาพของการจดจำให้สูงขึ้น เทคนิคต่างๆที่นำเสนอมีดังนี้

1. การแบ่งเขตย่อย (Sub Zone)
2. การหารหัสรูปร่าง (Feature Code) บางส่วนของตัวอักษร
3. การหาลักษณะหัวของตัวอักษร (Head Style)
4. การหาอัตราส่วนความกว้างต่อความสูงของตัวอักษร (Width per Height ratio)

4.4.1 การแบ่งเขตย่อย (Sub Zone)

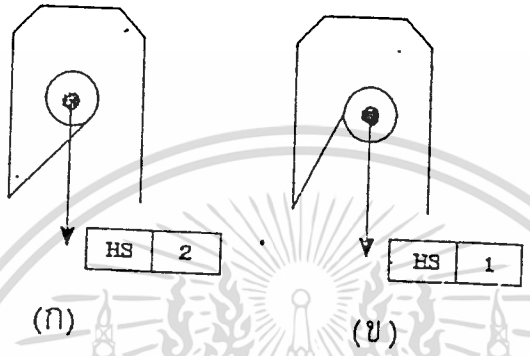
การแบ่งเขตของตัวอักษรเป็น 4 เขต ตามขั้นตอนการแบ่งกลุ่มของตัวอักษรแบบหยาบใน หัวข้อ 4.2 นั้น บางกรณีจะเกิดความไม่แน่นอนในเรื่องตำแหน่งของหัว เช่น ตัวอย่างในรูปที่ 4.14



รูปที่ 4.14 ตำแหน่งของหัวที่อาจจะไม่แน่นอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีง่ายๆ ในการตรวจสอบลักษณะของหัว คือการพิจารณารหัสรูปร่างในทิศทางใดทิศทางหนึ่ง โดยใช้ตำแหน่งหัวเป็นจุดอ้างอิงของการพารหัส ดังรูปที่ 4.19



รูปที่ 4.19 รหัสรูปร่างของหัว
(ก) รหัสรูปร่างของหัวด้านเข้า (ข) รหัสรูปร่างของหัวด้านออก

รหัสของรูปร่าง ของตัวอักษรในรูปที่ 4.19 (ก) และ (ข) จะเป็นตัวบอกถึงลักษณะของหัว HS

HS = 1 ตัวอักษรจะมีหัวด้านออก และ

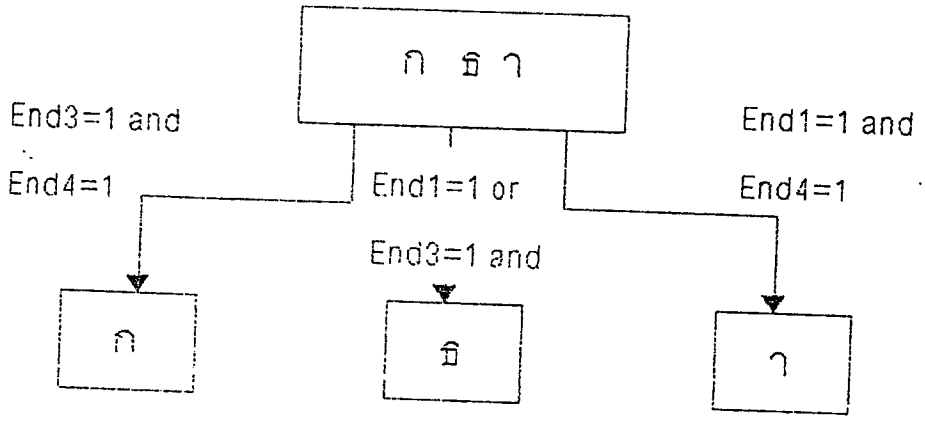
HS = 2 ตัวอักษรจะมีหัวด้านเข้า

4.4.4 การหาอัตราส่วนความกว้างต่อความสูง (Width per Height ratio)

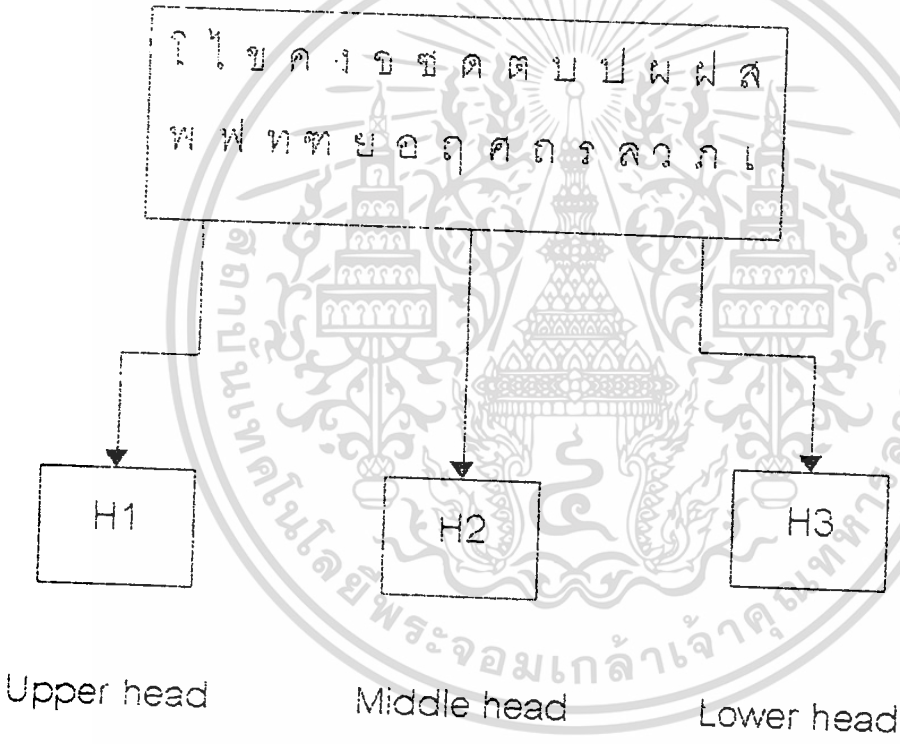
ตัวอักษร "ข" กับตัวอักษร "บ" จะมีลักษณะที่คล้ายกันมากที่สุด แต่ก็ยังมีลักษณะที่มีความแตกต่างกันที่ความกว้างของตัวอักษร ดังนั้นจึงนำเอาอัตราส่วนของความกว้างต่อความสูงของตัวอักษร WH ratio มาใช้เป็นตัวแยกตัวอักษร 2 ตัวนี้ออกจากกัน

จากการแบ่งกลุ่มของตัวอักษร การแบ่งกลุ่มย่อยตลอดจนเทคนิคต่างๆ ในการแยกตัวอักษรออกจากกลุ่มย่อย ที่กล่าวในบทนี้ จะสามารถนำมาจำแนกกลุ่มและแยกแยะตัวอักษรลายมือเขียนภาษาไทยได้โดยการทำโครงร่างแบบต้นไม้ (Tree structure) ดังแสดงในรูปที่ 4.20 ถึงรูปที่4.32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

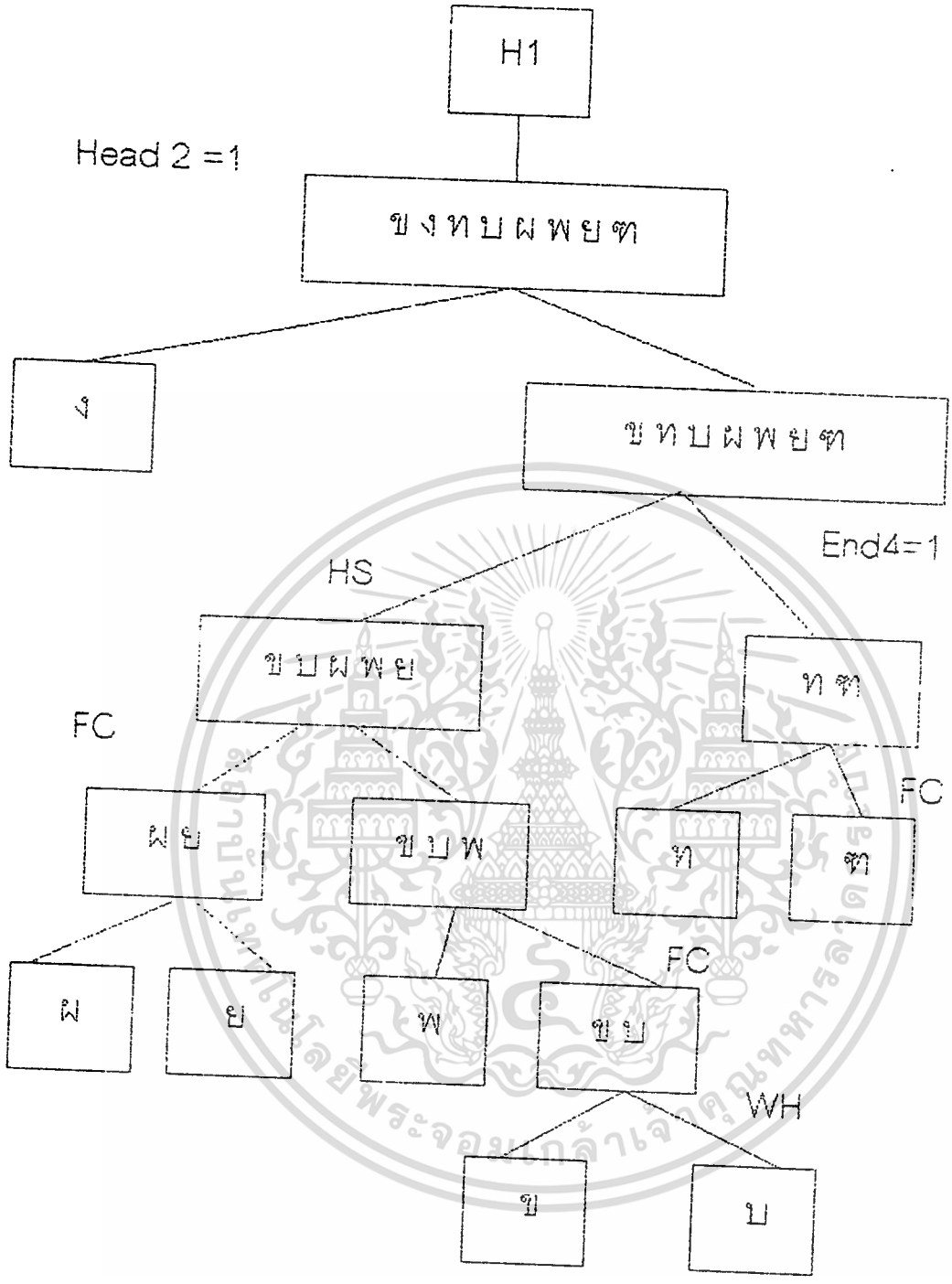


รูปที่ 4.20 โครงร่างแบบต้นไม้สำหรับตัวอักษรในกลุ่มที่ 1



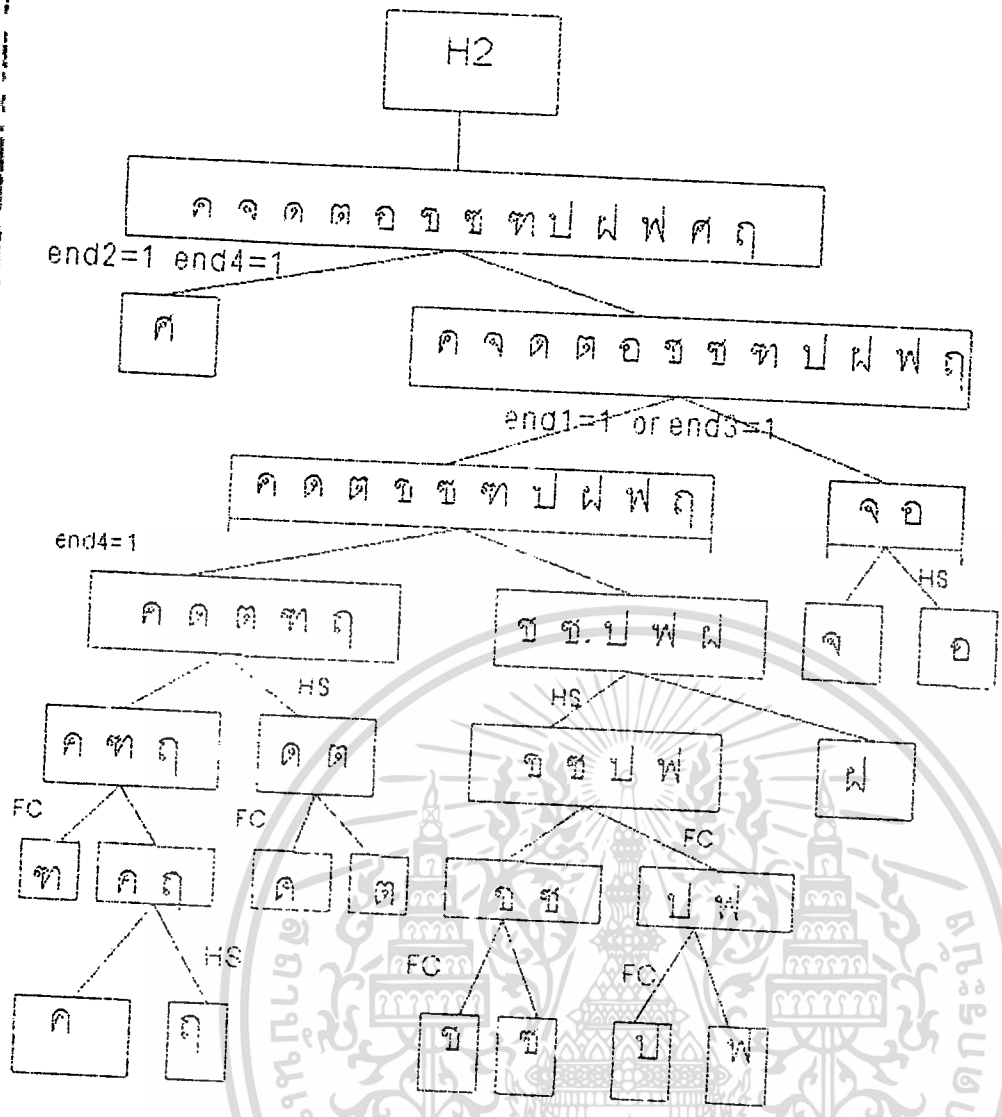
รูปที่ 4.21 โครงร่างแบบต้นไม้สำหรับตัวอักษรในกลุ่มที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



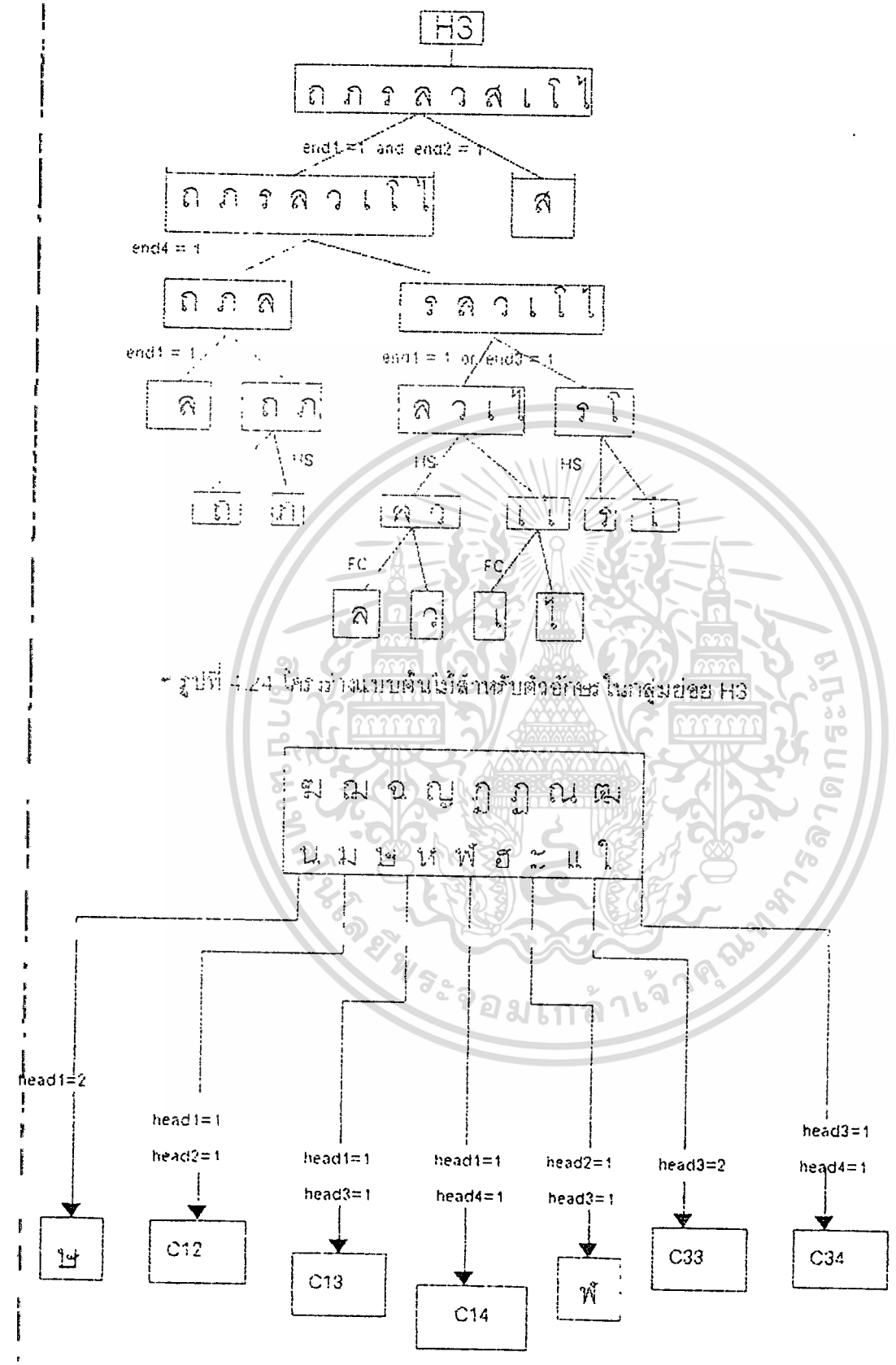
รูปที่ 4.22 โครงร่างแบบต้นไม้สำหรับตัวอักษรในกลุ่มย่อย H1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.23 โครงร่างแบบต้นไม้สำหรับตัวอักษรในกลุ่มย่อย H2

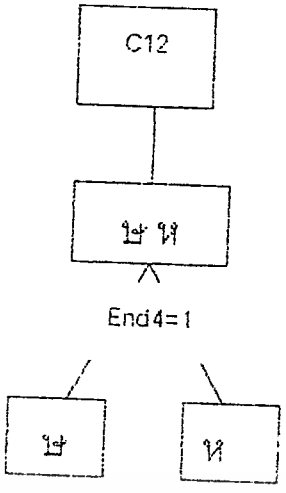
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



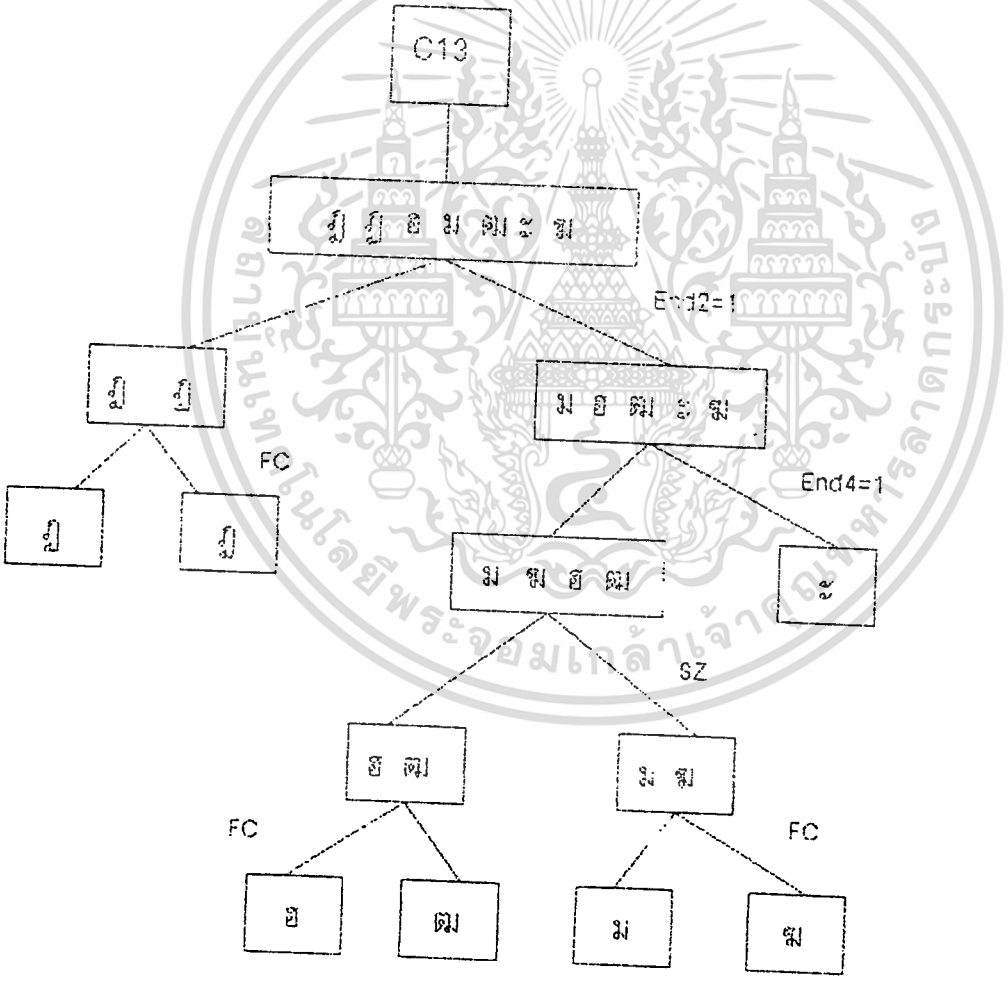
รูปที่ 4.24 โครงร่างแบบต้นไม้สำหรับตัวอักษรในกลุ่มย่อย H3

รูปที่ 4.25 โครงร่างแบบต้นไม้สำหรับตัวอักษรในกลุ่มที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

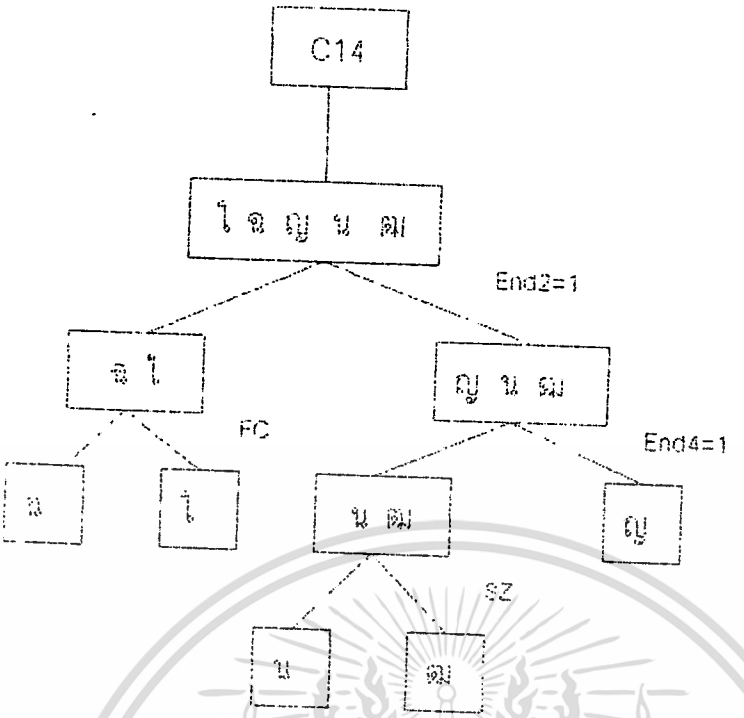


รูปที่ 4 26 โครงร่างแบบต้นไม้สำหรับตัวอักษรในกลุ่มย่อย C12

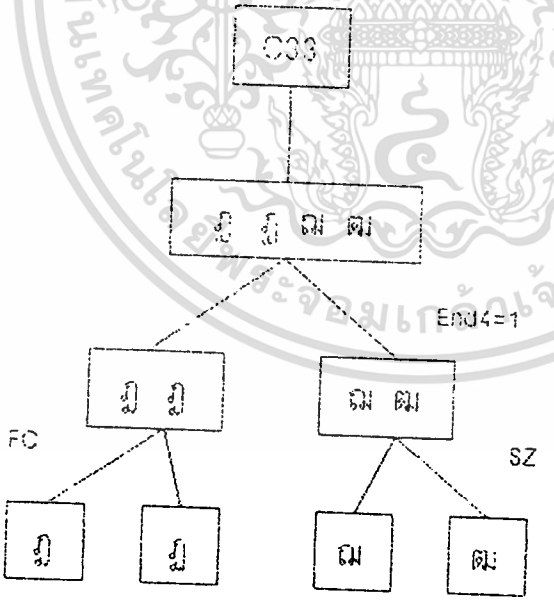


รูปที่ 4.27 โครงสร้างแบบต้นไม้สำหรับตัวอักษรในกลุ่มย่อย C13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

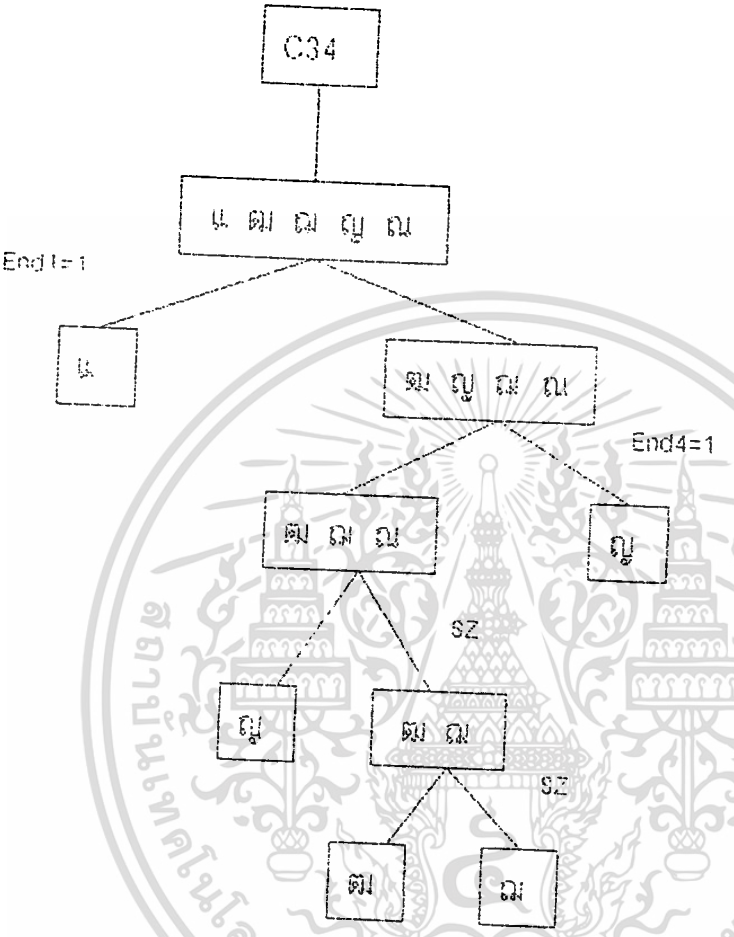


รูปที่ 4.28 โครงร่างแบบต้นไม้สำหรับตัวอักษรในกลุ่มย่อย C14



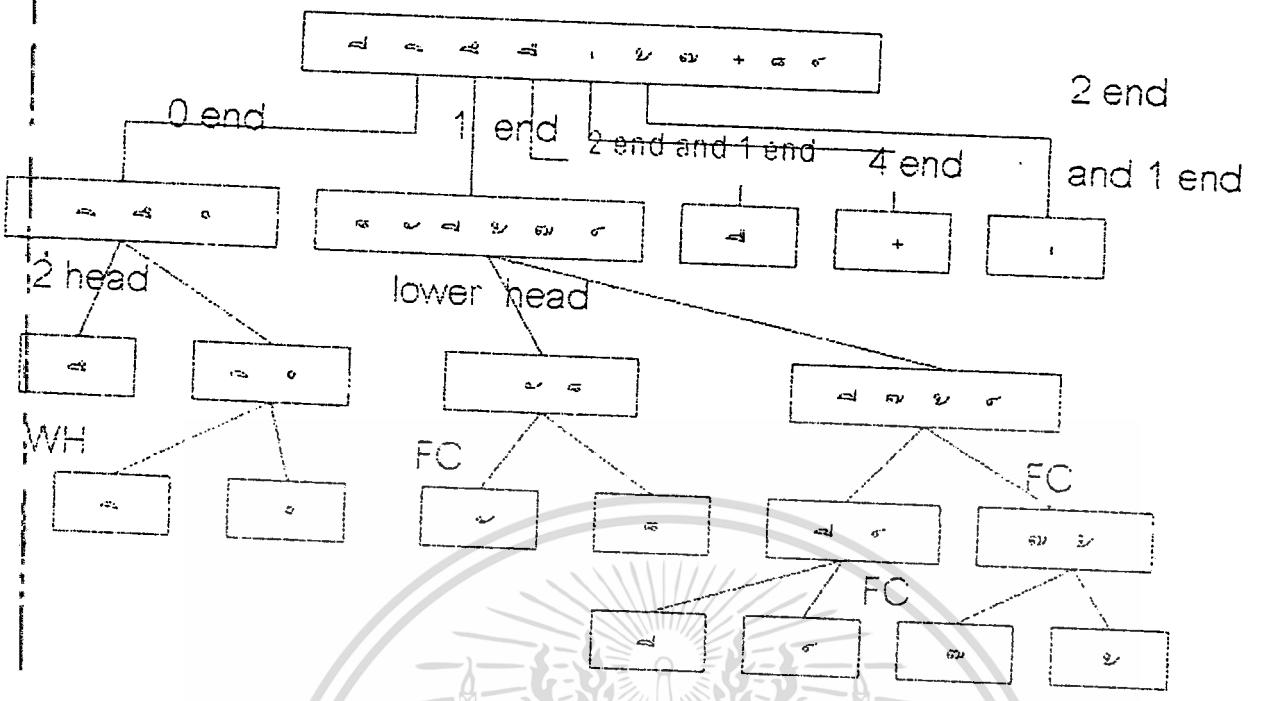
รูปที่ 4.29 โครงสร้างแบบต้นไม้สำหรับตัวอักษรในกลุ่มย่อย C33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

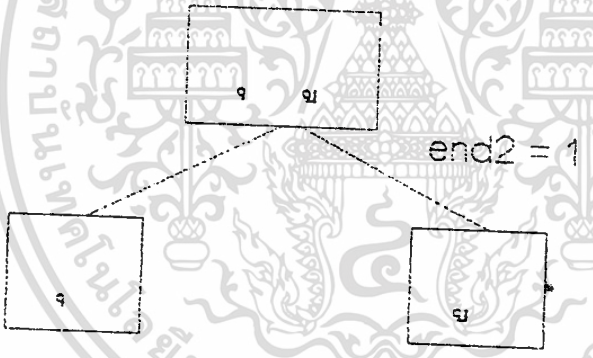


รูปที่ 4.30 โครงสร้างแบบต้นไม้สำหรับตัวอักษรในกลุ่มย่อย C34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.31 โครงร่างแบบต้นไม้สำหรับตัวอักษรระดับบน



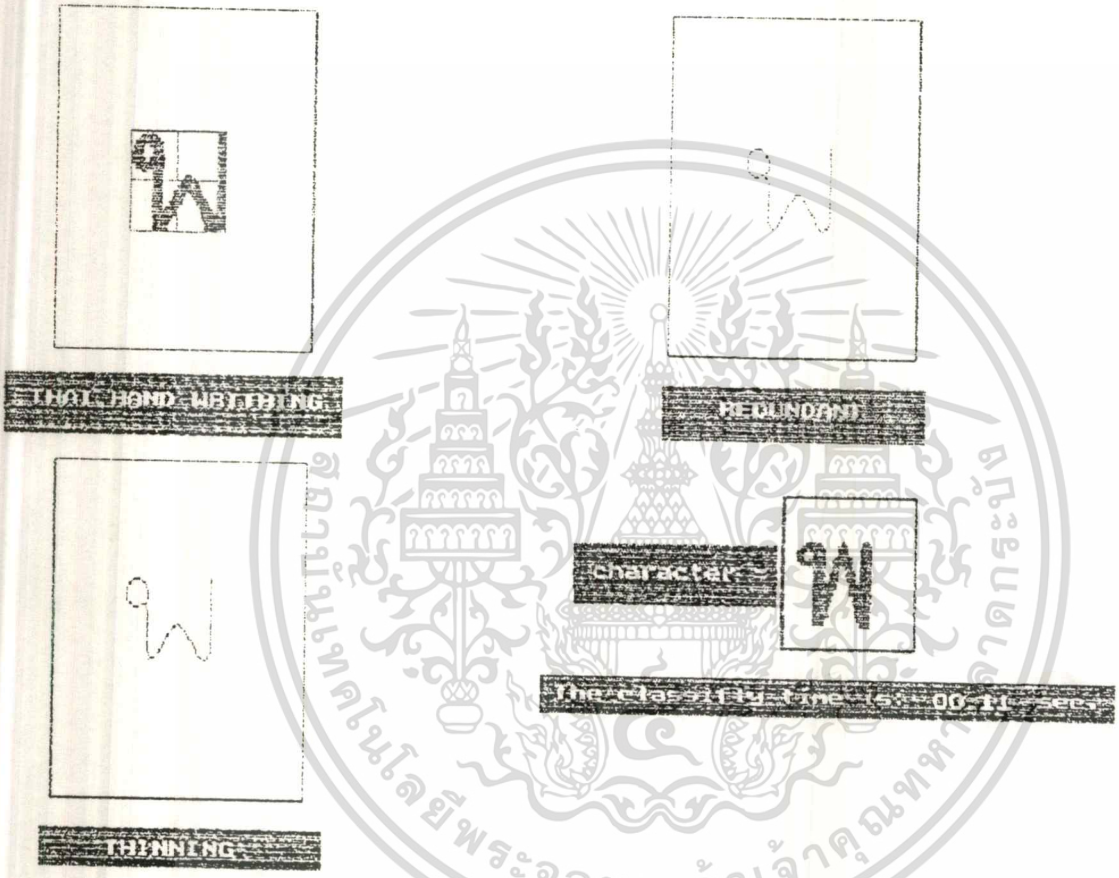
รูปที่ 4.32 โครงร่างแบบต้นไม้สำหรับตัวอักษรระดับล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

ผลการทดลองและสรุปผล

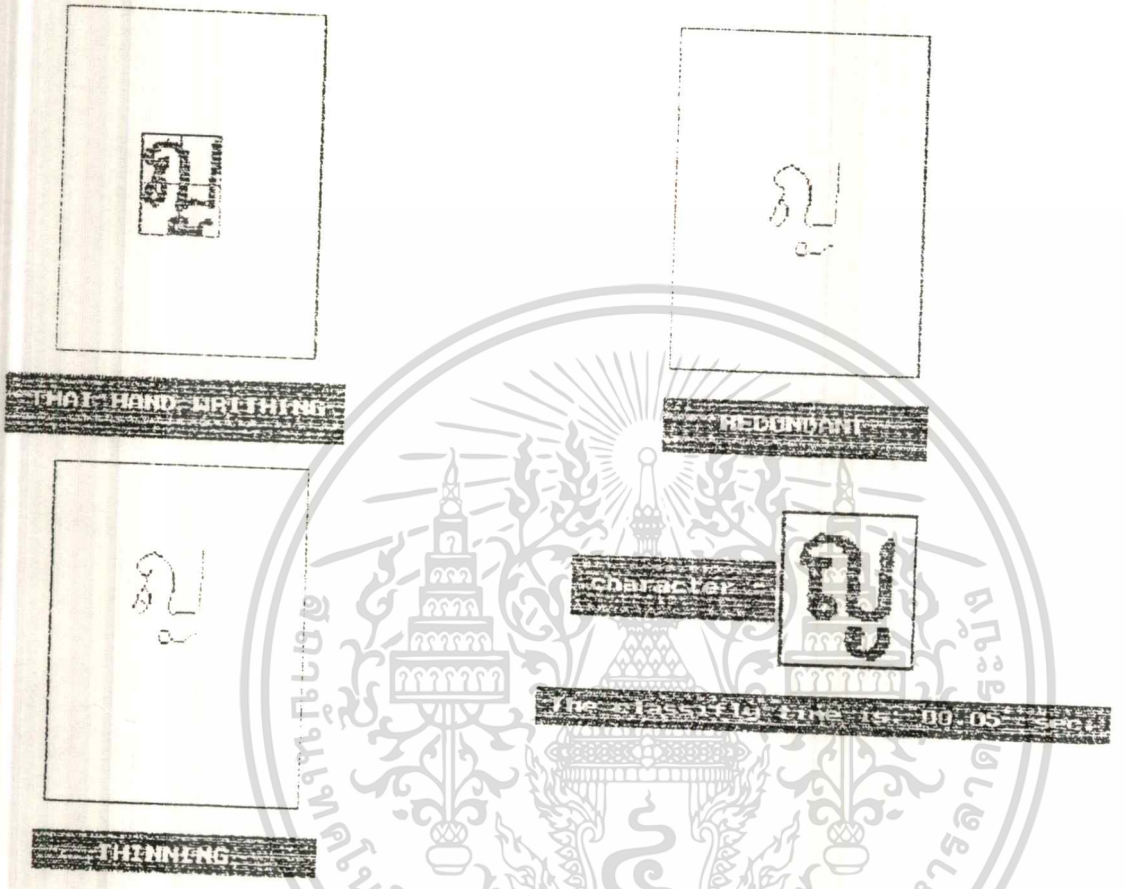
5.1 ผลการทดลอง



รูปที่ 5.1 ตัวอย่างการประมวลผลตัวอักษร "พ"

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 ตัวอย่างการประมวลผลตัวอักษร "ญ"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปอร์เซ็นต์ความถูกต้องของตัวอักษรแต่ละตัวโดยใช้ลายมือ 10 แบบ

ตัวอักษร	เปอร์เซ็นต์ความถูกต้อง
ก	100%
ข	100%
ค	100%
ฆ	80%
ง	100%
จ	100%
ฉ	100%
ช	80%
ซ	70%
ฌ	90%
ญ	80%
ฎ	70%
ฏ	60%
ฐ	100%
ฑ	70%
ฒ	80%
ณ	100%
ด	100%
ต	90%
ถ	100%
ท	100%
ธ	100%
น	100%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอักษร	เปอร์เซ็นต์ความถูกต้อง
บ	100%
ป	90%
ผ	100%
ฝ	90%
พ	100%
ฟ	90%
ภ	100%
ม	100%
ย	90%
ร	100%
ล	100%
ว	100%
ษ	90%
ส	100%
ศ	90%
ห	100%
ฬ	80%
อ	100%
ฮ	90%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 สรุปผลการทดลอง

1. ในการทดลองใช้ลายมือเขียน 10 แบบ ความถูกต้องคิดเป็น 92.38%
2. สาเหตุของความผิดพลาด
   2.1 เกิดจากการทำให้บางอาจจะทำให้เส้นแยกที่ยาวจนไม่สามารถลดส่วนเกินได้
   2.2 เกิดจากขณะทำให้บางกับเส้นที่มีขนาดเปลี่ยนแปลงเร็วอาจทำให้เกิดเส้นขาดได้
   2.3 เกิดจากลักษณะตัวอักษรที่เขียนไม่ถูกต้องตามแบบที่สามารถแบ่งอักษรได้ เช่น เขียนหัวไม่สมบูรณ์ เป็นต้น
   2.4 เกิดจากการเขียนรอยหยักไม่ชัดเจน
   2.5 ความหนาของตัวอักษรมีขนาดไม่เพียงพอในกรอดจำตัวอักษร ซึ่งในการทดลองต้องใช้ดินสอขนาด 2B ขึ้นไปหรือใช้ปากกาหมึกซึม จึงจะทำให้มีความหนาพอที่จะใช้ในการจำแนกตัวอักษรได้
3. เวลาที่ใช้ในการประมวลผล
   3.1 ขึ้นกับขนาดความหนาของตัวอักษร โดยตัวที่มีความหนามากจะใช้เวลาการประมวลผลมาก
   3.2 แต่ละตัวอักษรมีเวลาในการประมวลผลไม่เท่ากัน
   3.3 การทดลองใช้ คอมพิวเตอร์รุ่น Pentium 133 MHz , RAM 16 MB ใช้เวลาในการประมวลผลต่ำกว่า 0.2 วินาทีต่อตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. ดร.ชม กิ้มปาน "การจดจำรูปแบบตัวพิมพ์อักษรภาษาไทย" คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
2. ดร.สุรพันธุ์ เอื้อไพบูลย์ "การจดจำตัวอักษรลายมือเขียนภาษาไทยโดยการพิจารณาหัวของตัวอักษร" คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก
โปรแกรมของการจดจำตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอักษร	ลายมือที่									
	1	2	3	4	5	6	7	8	9	10
ก	ก	ก	ก	ก	ก	ก	ก	ก	ก	ก
ข	ข	ข	ข	ข	ข	ข	ข	ข	ข	ข
ค	ค	ค	ค	ค	ค	ค	ค	ค	ค	ค
ฆ	ฆ	ฆ	ฆ	ฆ	ฆ	ฆ	ฆ	ฆ	ฆ	ฆ
ง	ง	ง	ง	ง	ง	ง	ง	ง	ง	ง
จ	จ	จ	จ	จ	จ	จ	จ	จ	จ	จ
ฉ	ฉ	ฉ	ฉ	ฉ	ฉ	ฉ	ฉ	ฉ	ฉ	ฉ
ช	ช	ช	ช	ช	ช	ช	ช	ช	ช	ช
ซ	ซ	ซ	ซ	ซ	ซ	ซ	ซ	ซ	ซ	ซ
ฌ	ฌ	ฌ	ฌ	ฌ	ฌ	ฌ	ฌ	ฌ	ฌ	ฌ
ญ	ญ	ญ	ญ	ญ	ญ	ญ	ญ	ญ	ญ	ญ
ฎ	ฎ	ฎ	ฎ	ฎ	ฎ	ฎ	ฎ	ฎ	ฎ	ฎ
ฏ	ฏ	ฏ	ฏ	ฏ	ฏ	ฏ	ฏ	ฏ	ฏ	ฏ
ฐ	ฐ	ฐ	ฐ	ฐ	ฐ	ฐ	ฐ	ฐ	ฐ	ฐ
ฑ	ฑ	ฑ	ฑ	ฑ	ฑ	ฑ	ฑ	ฑ	ฑ	ฑ
ฒ	ฒ	ฒ	ฒ	ฒ	ฒ	ฒ	ฒ	ฒ	ฒ	ฒ
ณ	ณ	ณ	ณ	ณ	ณ	ณ	ณ	ณ	ณ	ณ
ด	ด	ด	ด	ด	ด	ด	ด	ด	ด	ด
ต	ต	ต	ต	ต	ต	ต	ต	ต	ต	ต
ถ	ถ	ถ	ถ	ถ	ถ	ถ	ถ	ถ	ถ	ถ
ท	ท	ท	ท	ท	ท	ท	ท	ท	ท	ท
ธ	ธ	ธ	ธ	ธ	ธ	ธ	ธ	ธ	ธ	ธ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอักษร	ลายมือที่									
	1	2	3	4	5	6	7	8	9	10
น	น	น	น	น	น	น	น	น	น	น
บ	บ	บ	บ	บ	บ	บ	บ	บ	บ	บ
ป	ป	ป	ป	ป	ป	ป	ป	ป	ป	ป
ผ	ผ	ผ	ผ	ผ	ผ	ผ	ผ	ผ	ผ	ผ
ฝ	ฝ	ฝ	ฝ	ฝ	ฝ	ฝ	ฝ	ฝ	ฝ	ฝ
พ	พ	พ	พ	พ	พ	พ	พ	พ	พ	พ
ฟ	ฟ	ฟ	ฟ	ฟ	ฟ	ฟ	ฟ	ฟ	ฟ	ฟ
ภ	ภ	ภ	ภ	ภ	ภ	ภ	ภ	ภ	ภ	ภ
ม	ม	ม	ม	ม	ม	ม	ม	ม	ม	ม
ย	ย	ย	ย	ย	ย	ย	ย	ย	ย	ย
ร	ร	ร	ร	ร	ร	ร	ร	ร	ร	ร
ล	ล	ล	ล	ล	ล	ล	ล	ล	ล	ล
ว	ว	ว	ว	ว	ว	ว	ว	ว	ว	ว
ษ	ษ	ษ	ษ	ษ	ษ	ษ	ษ	ษ	ษ	ษ
ส	ส	ส	ส	ส	ส	ส	ส	ส	ส	ส
ศ	ศ	ศ	ศ	ศ	ศ	ศ	ศ	ศ	ศ	ศ
ห	ห	ห	ห	ห	ห	ห	ห	ห	ห	ห
พ	พ	พ	พ	พ	พ	พ	พ	พ	พ	พ
อ	อ	อ	อ	อ	อ	อ	อ	อ	อ	อ
ฮ	ฮ	ฮ	ฮ	ฮ	ฮ	ฮ	ฮ	ฮ	ฮ	ฮ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include<conio.h>
#include<string.h>
#include<stdio.h>
#include<graphics.h>
#include<ctype.h>
#include<mem.h>
#include<dos.h>
#include<math.h>
#define right 1
#define up   2
#define left  3
#define down  4

const constX = 120;
const constY = 53;

struct time t1,t2;
struct BMPHEAD
{
unsigned char id[2];  //file type : must be ASCII text "BM"
long filesize;    //Size of the file : in double words (32 bit integers)
int reserved[2];  //Reserved for future use : must be zero
long headersize;  //byte offset to bitmap data : offset from the bitmap file header (i.e. the start of the file)
long infosize;    //number of bytes in header : currenty 40 bytes
long width;       //width of bitmap : in pixels
long depth;       //height of bitmap : in pixels
int biplanes;     //number of color planese : must be set to 1
int bits;         //number of bits per pixel : valid choices are 1,4,8,24if not 24, determines the size of the
palette
long bicompression;  //type of compression : {0:no compression} {1:run length 8 bits per pixel} { 2:run
length 4 bits per pixel}
long bisizeimage;    //size of image : in byte
long bixpelspermeter; //horizontal resolution : in pixels/meter
long biclrused;      //number of color indexes used by the bitmap : zero indicates all color important
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  long biClrImportant;  //number of color indexes important for displaying bitmap : a value of size
  indicates all colors are important
 } BMPhead;
  char BMPname[30];
 long  MaxX,MaxY,MinX,MinY,maxy=1,maxx=1,miny,minx;
  int   j,MinJ,MaxJ,minj,maxj=0;
  char huge buffer_image[1200/8][1800];
  long x,y;
  long width;
  char temp;
  int pointX,pointY,
  int driver = DETECT, mode;  //setup graphic mode
  unsigned char picZoom[constX][constY];
  unsigned char PT3[constX][constY];
  int n0,n1,n2,n3,n4,n5,n6,n7;
  int XHead[5],YHead[5];
  unsigned char Bran,End;
  unsigned char XBran[10],YBran[10],XEnd[10],YEnd[10];
  int head;
  char EndPos1,EndPos2,EndPos3,EndPos4;
  char HeadPos1,HeadPos2,HeadPos3,HeadPos4;

 int ReadImage(char *st)
 {
  FILE *BMPfile;

  if((BMPfile=fopen(st,"rb+"))==NULL)return(1); //can't open file
  fseek(BMPfile,0,SEEK_SET);
  fread(&BMPhead,sizeof(BMPhead),1,BMPfile);
  if(memcmp(BMPhead.id,"BM",2)) //if not bitmap file return 2
  {
   fclose(BMPfile);
   return(2);
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if(BMPhead.bits!=1) //if not b/w color return 3
    {
     fclose(BMPfile);
     return(3);
    }
    fseek(BMPfile,BMPhead.headersize,SEEK_SET);
    for(y=BMPhead.depth;y>=1;--y)
     for(x=1;x<=BMPhead.bisizeimage/BMPhead.depth;++x)
      buffer_image[x][y] = fgetc(BMPfile);
    fclose(BMPfile);
   // width=BMPhead.width/8;
   // if ((width%8)!=0) width++;
    return(0);
   }

   int PutPic(char *st)
   {
    FILE *BMPfile;
    if((BMPfile=fopen(st,"rb+"))==NULL)return(1); //can't open file
    fseek(BMPfile,0,SEEK_SET);
    fread(&BMPhead,sizeof(BMPhead),1,BMPfile);
    if(memcmp(BMPhead.id,"BM",2)) //if not bitmap file return 2
    {
     fclose(BMPfile);
     return(2);
    }
    if(BMPhead.bits!=1) //if not b/w color return 3
    {
     fclose(BMPfile);
     return(3);
    }
    fseek(BMPfile,BMPhead.headersize,SEEK_SET);
    for(y=BMPhead.depth;y>=1;--y)
    {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        j=0;
        for(x=1;x<=BMPhead.bisizeimage/BMPhead.depth;++x)
        {
        do
        {
         char b;
         if (j==0) b=~fgetc(BMPfile);
         if((b&(0x80>>j))&&((8*(x-1)+j)<BMPhead.width))
              putpixel(8*x+j+460,y+250,BROWN);
              else putpixel(8*x+j+460,y+250,GREEN);
         j++;
         if(j>7)j=0;
        }
        while(j!=0);
       }
      }
      fclose(BMPfile);
      return(0);
     }
     void LowNoise()
     {
      char compare;
      for(y=1;y<=BMPhead.depth;++y)
      {
       for(x=1;x<=width;++x)
       {
        j=0;
        while ((j<=7)&(BMPhead.width>=8*(x-1)+j))
        {
compare = 0x80>>j;
          n0 = (j==7)?buffer_image[x+1][y]&0x80:buffer_image[x][y]&(compare>>1);
          n1 = (j==7)?buffer_image[x+1][y-1]&0x80:buffer_image[x][y-1]&(compare>>1);
          n2 = buffer_image[x][y-1]&compare;
          n3 = (j==0)?buffer_image[x-1][y-1]&0x01:buffer_image[x][y-1]&(compare<<1);
          n4 = (j==0)?buffer_image[x-1][y]&0x01:buffer_image[x][y]&(compare<<1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            n5 = (j==0)?buffer_image[x-1][y+1]&0x01:buffer_image[x][y+1]&(compare<<1);
            n6 = buffer_image[x][y+1]&compare;
            n7 = (j==7)?buffer_image[x+1][y+1]&0x80:buffer_image[x][y+1]&(compare>>1);
             if ( buffer_image[x][y]&(0x80>>j) )
             {
              if (BMPhead.width==8*(x-1)+j) n0=n1=n7=0;
              if ( (n0|n1|n2|n3|n4|n5|n6|n7)==0 )
                      buffer_image[x][y] = buffer_image[x][y]&~(0x80>>j);
             }
            else  if ( (n0&n1&n2&n3&n4&n5&n6&n7)==1 )
                      buffer_image[x][y] = buffer_image[x][y]|(0x80>>j);
            j++;
        }
      }
    }

void border()
{
 for(y=1;y<=BMPhead.depth;y++)
  for(j=0,x=1;8*(x-1)+j+1<=BMPhead.width;j++)
  {
   if(j==8) { j=0; x++; };
   if ((buffer_image[x][y])&(0x80>>j))
   MaxY=y;
  }
 for(y=BMPhead.depth;y>=1;y--)
  for(j=0,x=1;8*(x-1)+j+1<=BMPhead.width;j++)
  {
   if(j==8) { j=0; x++; }
   if ((buffer_image[x][y])&(0x80>>j)) MinY=y;
  }
for(j=0,x=1;8*(x-1)+j+1<=BMPhead.width;j++)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
     if(j==7) { j=0; x++; }
    for(y=BMPhead.depth;y>=1;y--)
       if ((buffer_image[x][y])&(0x80>>j)) { MaxX=x; MaxJ=j;}
    }
    for(j=BMPhead.width%8,x=BMPhead.width/8;8*(x-1)+j+1>=1;j--)
    {
     if(j==-1) { x--; j=7; }
     for(y=BMPhead.depth;y>=1;y--)
       if ((buffer_image[x][y])&(0x80>>j)) { MinX=x; MinJ=j;}
    }
    x=x;
   }

   void LabelN()
   {
    if((x==0)||(y==0)) {n1=n2=n3=n4=n5=0;}
    else
    {
     n1 = picZoom[x+1][y-1];
     n2 = picZoom[x][y-1];
     n3 = picZoom[x-1][y-1];
     n4 = picZoom[x-1][y];
     n5 = picZoom[x-1][y+1];
    }
    n0 = picZoom[x+1][y];
    n6 = picZoom[x][y+1];
    n7 = picZoom[x+1][y+1];
   }



   void zoom()
   {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
long pointYmax,i,x,y,loopX,loopY,count;
float ratioY,ratioX;

setbkcolor(LIGHTBLUE);
setfillstyle(1,LIGHTGREEN);
bar(100,10,64*2+100,(int)(64*8/3.0+10));
bar(100,230,64*2+100,(int)(64*8/3.0+230));
bar(400,10,64*2+400,(int)(64*8/3.0+10));
setcolor(WHITE);
rectangle(100,10,64*2+100,(int)(64*8/3.0+10));
rectangle(100,230,64*2+100,(int)(64*8/3.0+230));
rectangle(400,10,64*2+400,(int)(64*8/3.0+10));
for (x=0;x<constX;x++)
 for (y=0;y<constY;y++)
        picZoom[x][y] = 0;
ratioX = ratioY = (MaxY-MinY)/(constY-3);
if ((ratioX<1)||(ratioY<1)) {ratioX=1.0;ratioY=1.0;}
x=y=j=pointYmax=pointX=pointY=0;
while( ((8*x+j)<=(8*MaxX+MaxJ))||(y<=MaxY) )
{
 loopY=1;
 count=0;
 y=MinY+pointY*ratioY;
 while ( (loopY<=ratioY) && (y<=MaxY))
 {
  x=MinX+(int)(MinJ+pointX*ratioX)/8;
  j=(int)(MinJ+pointX*ratioX)%8;
  loopX=1;
  while ( (loopX<=ratioX) && ((8*x+j)<=(8*MaxX+MaxJ)) )
  {
       if ((buffer_image[x][y]<<j)&0x80) count++;
       j++;
       if (j==8) {j=0; x++;}
       loopX++;
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        y++;
        loopY++;
      }
      pointY++;
      if ( y > MaxY ) {pointX++; pointY = 1;}
      if (pointYmax < (pointY-1)) {pointYmax = pointY-1;}
      if ((pointY-1))
      if (count> 0/*(
      (ratioX*ratioY)*0.7)*/) picZoom[pointX+1][pointY-1]=1;
    }
    pointY = pointYmax;
    for (x=1;x<=pointX;x++)
      for (y=1;y<=pointY;y++)
        if (picZoom[x][y]==1) putpixel(x+140,y+30+(int)(64*4/6.0),BROWN);
    setcolor(BLUE);
    line (140+pointX/2,72,140+pointX/2,72+pointY);
    line (140,(144+pointY)/2,140+pointX,(144+pointY)/2);
    rectangle(140,72,140+pointX,72+pointY);
  }

  int check_head()
  {
   int  x1,y1,direct,count;
   char pic_head[constX][constY];
   char pic_head1[constX][constY];
   char  flag,Ycut,Xcut;
   int  XBorderMin,XBorderMax,YBorderMin,YBorderMax;
   for ( y=0; y<=pointY+1; y++)
    for ( x=0; x<=pointX+1; x++)
     {
          pic_head[x][y] =0;
          pic_head1[x][y]=1;
     }
  for (y=0; y<=pointY+1; y++)
    for (x=0; x<=pointX+1; x++)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        if(picZoom[x][y]==0)
        {
            LabelN();
            if ( (n0||n2||n4||n6)==1) pic_head[x][y] = 1;
        }
      y= 1;
      do
      {
       y++;
       Ycut = y;
       for (x=1;x<pointX;x++)
         if(picZoom[x][y]) { Ycut=pointY; }
      }
      while((Ycut!=y)&&(y<=pointY));
      x=1;
      do
      {
       x++;
       Xcut = x;
       for (y=1;y< pointY;y++)
         if(picZoom[x][y]) { Xcut=pointX; }
      }
      while((Xcut!=x)&&(x<=pointX));
      j=0;
      do
      {
       flag = 1;
       x = 0;
       y = 1;
       do
       {
        x++;
        if(x>pointX)   {y++;x=0;}
        if((x==pointX)&&(y==pointY)) flag=0;
       }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    while  ( (pic_head[x][y]==0) && (flag==1));
    if(flag==1)
    {
     head++;
     XBorderMin = XBorderMax = x1 = x;
     YBorderMin = YBorderMax = y1 = y;
     if ((head==0)||(Ycut<y)||(Xcut<x)) {direct = 3;} else {direct = 6;}
     do
     {
          if(pic_head[x][y]==0)
          {
           count++;
           if((direct==0)||(direct==7)) { y--;}
           else if((direct==1)||(direct==2)) { x--;}
           else if((direct==3)||(direct==4)) { y++;}
           else if((direct==5)||(direct==6)) { x++;}
           if (direct <=6){direct++;}else {direct=0;}
          }
          else
          {
           count=0;
           if(x<XBorderMin) XBorderMin = x;
           if(x>XBorderMax) XBorderMax = x;
           if(y<YBorderMin) YBorderMin = y;
           if(y>YBorderMax) YBorderMax = y;
           pic_head1[x][y]=0;
           if((direct==0)||(direct==1)) { y++; direct=6;}
           else if((direct==2)||(direct==3)) { x++;direct=0;}
           else if((direct==4)||(direct==5)) { y--;direct=2;}
           else if((direct==6)||(direct==7)) { x--;direct=4;}
          }
     }
     while( ((x1!=x)||(y1!=y))&&(flag==1)&&(count<9));
     if (count>=9) pic_head1[x][y] = 0;
     for (x=0;x<=pointX+1;x++)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            for (y=0;y<=pointY+1;y++)
            {
             pic_head[x][y]=pic_head[x][y]&&pic_head1[x][y];
            // if(pic_head1[x][y]==0)putpixel(x+140,y+30+(int)(64*4/6.0),WHITE);
            }
        }
        //getch();
        if(Xcut<XBorderMin)
        {
         Xcut=pointX; head--;
        }
        else if(Ycut<YBorderMin)
        {
         Ycut=pointY; head--;
        }
        else
        {
         XHead[j] = (XBorderMax+XBorderMin)/2;
         YHead[j] = (YBorderMax+YBorderMin)/2;
         j++;
        }
      }
      while(flag==1);
      return(head);
    }

    void thinning()
    {
     int flag = 1;
     int bLEFT,bRIGHT,bUP,bDOWN;
     while(flag == 1)
     {
      flag = 0;
      for (x=1; x<=pointX; x++)
       for (y=1; y<=pointY; y++)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            if(picZoom[x][y]==1)
            {
             LabelN();
             bUP  = n6&(n0|n4|n5|n7)&(n0|(1-n1))&((1-n3)|n4);
             bDOWN =  n2&(n0|n1|n3|n4)&(n4|(1-n5))&((1-n7)|n0);
             if(n2==0)
             {
              if(bUP == 1) picZoom[x][y] = 8;
             }
             else
             if(n6==0)
             {
              if(bDOWN==1) picZoom[x][y] = 8;
             }
            }
      for (y=0;y<=pointY;y++)
       for (x=0;x<=pointX;x++)
            if (picZoom[x][y] == 1)
            {
             LabelN();
             bRIGHT = n4&(n2|n3|n5|n6)&(n5|(1-n7))&((1-n1)|n2);
             bLEFT  = n0&(n1|n2|n6|n7)&(n2|(1-n3))&((1-n5)|n6);
             if(n4==0)
             {
              if (bLEFT == 1) picZoom[x][y]=8;
             }
             else if(n0==0)
             {
              if (bRIGHT == 1) picZoom[x][y]=8;
             }
            }
      for (x=1; x<=pointX; x++)
      {
       for (y=1; y<=pointY; y++)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            if (picZoom[x][y] == 8)
            {
             flag = 1;
             picZoom[x][y] = 0;
            }
        }
       }
      }
      for (x=1;x<=pointX;x++)
       for (y=1;y<=pointY;y++)
            if (picZoom[x][y]== 1) putpixel(x+140,y+230+(int)(64*4/6.0),BROWN);
    }

    int NumBorder()
    {
     LabelN();
    // return(n0+n1+n2+n3+n4+n5+n6+n7);
     return((n0&&(n1==0))+(n1&&(n2==0))+(n2&&(n3==0))+(n3&&(n4==0))
            +(n4&&(n5==0))+(n5&&(n6==0))+(n6&&(n7==0))+(n7&&(n0==0)));
    }

    void redun()
    {
     int j,t,i=0;
     int PT2[constX][constY];
     int YT[10],XT[10];
     int round=8;
     for ( y=0; y<constY; y++)
      for ( x=0; x<constX; x++)
            PT2[x][y] =0;
     for ( t=0; t<round; t++)
     {
      for ( y=0; y<constY; y++)
       for ( x=0; x<constX; x++)
        if (picZoom[x][y] == 1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            {
             //NC4_NC8();
             if ( NumBorder()==1 )
               {
                if (PT2[x][y]!=2) { YT[i]=y; XT[i]=x; i++; PT2[x][y]=1; }
               }
             else if ( NumBorder()>2 )
               {PT2[x][y] = 2;}
            }
      for (j=0; j<i; j++) { picZoom[XT[j]][YT[j]] = 0;}
      i=0;
    }
    for (t=0; t<round; t++)
     for (x=0; x<constX; x++)
      for (y=0; y<constY; y++)
           if (PT2[x][y] == 1)
           {
            if ( (PT2[x-1][y-1]==2) || (PT2[x][y-1]==2) || (PT2[x+1][y-1]==2)
                 || (PT2[x-1][y]==2) || (PT2[x+1][y]==2)||(PT2[x-1][y+1]==2)
                                  ||(PT2[x][y+1]==2)||(PT2[x+1][y+1]==2) )
              {PT2[x][y]=2;}
           }
    for (y=0; y<constY; y++)
     for (x=0; x<constX; x++)
          if(PT2[x][y]==2) PT2[x][y] = 0;}
    for (y=0; y<constY; y++)
     for (x=0; x<constX; x++)
          picZoom[x][y] = picZoom[x][y] + PT2[x][y];
    for (x=1;x<=pointX;x++)
     for (y=1;y<=pointY;y++)
      if (picZoom[x][y]==1) putpixel(x+415+32,y+30+(int)(64*4/6.0),BROWN);
}

void property()
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    for ( y=0; y<constY; y++)
     for ( x=0; x<constX; x++)
       PT3[x][y] =0;
    for (y=0; y<constY; y++)
     for (x=0; x<constX; x++)
      if(picZoom[x][y] == 1)
      {
            if (NumBorder()==1) {End++;YEnd[End]=y; XEnd[End]=x; PT3[x][y]=1;}
            else if (NumBorder()==2) {PT3[x][y]=2;}
            else if (NumBorder()==3) {Bran++; YBran[Bran]=y; XBran[Bran]=x; PT3[x][y] =
3;}
            else if (NumBorder()==4) {Bran++; YBran[Bran]=y; XBran[Bran]=x; PT3[x][y] =
4;}
      }
    }

    char HeadLightTail()
    {
     char HeadPoint = 1;
     if (Bran != 1)
     {
      for(x=2; x<=Bran; x++)
      {
       if (  (pow(XBran[x-1]-XHead[1],2)+pow(YBran[x-1]-YHead[1],2))
            > (pow(XBran[x]-XHead[1],2)+pow(YBran[x]-YHead[1],2)))
            { HeadPoint = x; }
      }
     }
     return (XHead[1]> XBran[HeadPoint]); //true = head in right
    }

    char HeadUpTail()
    {
     char HeadPoint = 1;
     if (Bran != 1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
     for(x=2; x<=Bran; x++)
     {
      if (  (pow(XBran[x-1]-XHead[1],2)+pow(YBran[x-1]-YHead[1],2))
            > (pow(XBran[x]-XHead[1],2)+pow(YBran[x]-YHead[1],2)))
            { HeadPoint = x; }
     }
    }
    return (YHead[1]<YBran[HeadPoint]); // true = head in righ
   }

   void Position()
   {
    HeadPos1=HeadPos2=HeadPos3=HeadPos4=0;
    for(x=1;x<=head;x++)
    {
     if ((XHead[x]<pointX/2)&&(YHead[x]<pointY/2)) HeadPos1++;
     if ((XHead[x]>=pointX/2)&&(YHead[x]<pointY/2)) HeadPos2++;
     if ((XHead[x]<pointX/2)&&(YHead[x]>=pointY/2)) HeadPos3++;
     if ((XHead[x]>=pointX/2)&&(YHead[x]>=pointY/2)) HeadPos4++;
    }
   EndPos1=EndPos2=EndPos3=EndPos4=0;
   for(x=1;x<=End;x++)
   {
    if ((XEnd[x]<pointX/2)&&(YEnd[x]<pointY/2)) EndPos1 = 1;
    if ((XEnd[x]>=pointX/2)&&(YEnd[x]<pointY/2)) EndPos2 = 1;
    if ((XEnd[x]<pointX/2)&&(YEnd[x]>=pointY/2)) EndPos3 = 1;
    if ((XEnd[x]>=pointX/2)&&(YEnd[x]>=pointY/2)) EndPos4 = 1;
   }
  }

  char head0()
  {
   if(EndPos3&&EndPos4)          return(1);
   if((EndPos1||EndPos3)&&EndPos2)    return(24);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        return(51);
      }

      char Zigzag(char y)
      {
       char num=0;
       for(x=1;x<=pointX;x++)
        if(picZoom[x][y]&&(picZoom[x-1][y]==0)) num++;
       return(num>2);
      }

      char ZigzagUp()
      {
       return(Zigzag(5));
      }

      char ZigzagDown()
      {
       return(Zigzag(pointY-9));
      }

      char ZigzagHead()
      {
       char flag=0;
       for(x=1;x<pointX;x++)
       {
        if(picZoom[x][YBran[1]-1]&&(picZoom[x-1][YBran[1]-1]==0)) flag++;
//      putpixel(x+415+32,YBran[1]-1+30+(int)(64*4/6.0),WHITE);
       }
       return(flag>4);
      }

      char WHratio()
      {
       char x1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  for(x=pointX;x--;x<=1)
    if(picZoom[x][pointY/2]) x1=x;
  return((pointX-x1)<20);
 }

 char LargeWidth()
 {
  char x1,x2;
  for(x=1;x<=pointX;x++)
    if(picZoom[x][pointY-10]) x1=x;
  for(x=pointX;x--;x>=1)
    if(picZoom[x][pointY-10]) x2=x;
  return((x1-x2)>12);
 }

 char Segment13Light()
 {
  return(XHead[1]>=pointX/4);
 }

 char Segment24Light()
 {
  return(XHead[1]>=(pointX*3)/4);
 }

 char Segment24Light2()
 {
  return(XHead[2]>=(pointX*3)/4);
 }

 char Segment12Up()
 {
  return(YHead[1]<=pointY/4);
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
char Segment12Up2()
{
 return(YHead[2]<=pointY/4);
}

char Segment34Up()
{
 return(YHead[1]<=(pointY*3)/4);
}

char NumCut(char num)
{
 char flag=0;
 for(x=1;x<=pointX;x++)
  if(picZoom[x][num]&&(picZoom[x-1][num]==0)) flag++;
 return(flag);
}

char UpHead()
{
 if(HeadPos2)          return(7);
 if (EndPos4)
 {
  if(ZigzagHead())     return (17);
  return (23);
 }
 if(HeadLightTail())
 {
  if(ZigzagDown())     return(28);
  return(34);
 }
 if(ZigzagDown())      return(30);
 if(WHratio())         return(2);
 return(26);
}
```

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}

char MidHead()
{
 if(EndPos2&&EndPos4)       return(38);
 if(EndPos1||EndPos3)
 {
  if(EndPos4)               return(17);
  if(HeadLightTail())       return(43);
  return(8);
 }
 if(EndPos4)
 {
  if(HeadLightTail())
  {
   if(ZigzagHead())         return(17);
   if(HeadUpTail())         return(4);
   return(45);
  }
  if(NumCut(pointY*3/4)>1)
  {
   if (ZigzagUp())          return(21);
   return(20);
  }
  return(46);
 }
 if(HeadLightTail())
 {
  if(ZigzagDown())          return(29);
  return(11);
 }
 if(LargeWidth())
 {
  if(ZigzagDown())          return(31);
  return(27);
 }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if(ZigzagHead())          return(11);
    return(10);
   }

   char DownHead()
   {
    if(EndPos1&&EndPos2)      return(40);
    if(EndPos4)
    {
     if(EndPos1)              return(36);
     if(HeadLightTail())      return(22);
     return(32);
    }
    if(EndPos1||EndPos3)
    {
     if(HeadLightTail())
     {

          if(HeadPos3)        return(36);
          if(NumCut(5)>1)     return(61);
      return(58);
     }
     return(37);
    }
    if(HeadLightTail())       return(60);
    return(35);
   }

  char head1()
  {
   if (YHead[1]<(pointY/3.))   return(UpHead());
   if  (YHead[1]> pointY*2/3.)  return(DownHead());
   return(MidHead());
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
char Dorchada_Torpatug()
{
 char flag=0;
 if(YBran[1] > YBran[2]) y=YBran[1]; else y=YBran[2];
 for(x=1;x<=pointX;x++)
 {
  if(picZoom[x][y]&&(picZoom[x-1][y]==0)) flag++;
  putpixel(x+415+32,y+30+(int)(64*4/6.0),WHITE);
 }
 return(flag<5);
}
char C12()
{
 if(EndPos4)                return(41);
 if(Segment12Up2())         return(42);
 return(39);
}
char C13()
{
 if(End==0)                 return(44);
 if(EndPos2)
 {
  if(EndPos4)               return(50);
  if(Segment13Light())
  {
   if(NumCut(pointY-8)<4)  return(44);
   return(18);
  }
  if(ZigzagHead())          return(6);
  return(33);
 }
if(Dorchada_Torpatug())    return(14);
return(15);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
char C14()
{
 if(EndPos2)
 {
  if(EndPos4)            return(13);
  if(Segment12Up())      return(25);
  return(18);
 }
 if(NumCut(2*pointY/3)> 1)   return(9);
 return(62);
}

char C33()
{
 if(EndPos2)
 {
  if(EndPos4)            return(13);
  if(Segment13Light())   return(16);
  return(12);
 }
 if(Dorchada_Torpatug())  return(14);
 return(15);
}

char C34()
{
 if(EndPos1&&EndPos2)     return(59);
 if(EndPos1||EndPos3)     return(9);
 if(EndPos4)              return(13);
 if(Segment34Up())        return(18);
 if(Segment24Light2())    return(19);
 return(12);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
char HeadMore1()
{
 if(head>2)              return(16);
 if(HeadPos2&&HeadPos3)  return(42);
 if(HeadPos1==2)         return(39);
 if(HeadPos1&&HeadPos2)  return(C12());
 if(HeadPos1&&HeadPos3)  return(C13());
 if(HeadPos1&&HeadPos4)  return(C14());
 if(HeadPos3==2)         return(C33());
 if(HeadPos3&&HeadPos4)  return(C34());
 return(0);
}

char Display(char ChPos)
{
 LowNoise();
  {
   End =0; Bran = 0; head =-1;
   border();
   zoom();
   setcolor(WHITE);
   gettime(&t1);
   head =check_head();
   thinning();
   redun();
   property();
   Position();
   if(ChPos==1)
   {
       if(End==4)          return(67);
       if(End==3)          return(55);
       if(End==2)          return(64);
       if(End==0)
       {
        if(head==2)        return(54);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                if(pointX>1.5*pointY)  return(52);
                return(63);
              }
              if(HeadLightTail())
              {
               if(NumCut(pointY/2)>1) return(66);
               return(68);
              }
             if(EndPos2)
             {
              if(HeadPos2||HeadPos4)
              {
               if(XHead[1]>3*pointX/4)  return(70);
               return(53);
              }
              if(XBran[1]>pointX/2)   return(53);
              if((YHead[1]<3*pointY/4)&&(NumCut(XBran[1]+3)>1)) return(65);
              if(YHead[1]>YBran[1])   return(68);
              return(69);
             }
            return (0);
        }
        else if(ChPos==2)
        {
            if (head==0)          return(head0());
            else if (head==1)     return(head1());
            return(HeadMore1());
        }
        else
        {
            if(EndPos2)           return(57);
            return(56);
        }
      }
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void main()
{
 char Name[10];
 int ChPos;
 clrscr();
 printf ("\nfile : ");
 scanf("%s",BMPname);
 printf ("Position(1=up;2=mid;3=down) : ");
 scanf("%d",&ChPos);
 strncat(BMPname,".bmp",7);
 if (ReadImage(BMPname)==0)
 {
  int driver = DETECT, mode;  //setup graphic mode
  initgraph(&driver,&mode,"C:\\BC5\\BGI");
  setcolor(LIGHTGREEN);
  outtextxy(90,200,"THAI HAND WRITHING");
  outtextxy(430,200,"REDUNDANT");
  outtextxy(130,420,"THINNING");
  int b = Display(ChPos);
  gettime(&t2);
  outtextxy(360,280," character = ");
  char*  buffer;
  sprintf(buffer,"The classify time is: %02d.%02d  sec.",
  60*(t2.ti_min-t1.ti_min)+t2.ti_sec-t1.ti_sec,100*60*(t2.ti_min-t1.ti_min)
  +100*(t2.ti_sec-t1.ti_sec)+t2.ti_hund-t1.ti_hund);
  outtextxy(350,340,buffer);
  Name[0] = 't';
  Name[1] = b/10+48;
  Name[2] = b%10+48;
  Name[3] = NULL;
  strncat(Name,".bmp",7);
  PutPic(Name);
  getch();
  closegraph();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
    else if (ReadImage(BMPname)==1)
    {
     printf("can't open file");
     getch();
    }
    else if (ReadImage(BMPname)==2)
    {
     printf("It's not BMP file.");
     getch();
    }
    else if (ReadImage(BMPname)==3)
    {
     printf("It's not black/white color.");
     getch();
    }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้