สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบเก็บฐานข้อมูลลายมือเขียนผ่านเว็บ

HANDWRITTEN WEB-BASED COLLECTION SYSTEM

กิตติคุณ นันตา

คุณากร โสรเนตร

เลขหมู่.....
เลขทะเบียน 62394
วัน,เดือน,ปี 17 ส.ค. 2549

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบเก็บฐานข้อมูลลายมือเขียนผ่านเว็บ

HANDWRITTEN WEB-BASED COLLECTION SYSTEM

โดย

กิตติคุณ นันตา

คุณากร โสรเนตร

อาจารย์ที่ปรึกษา

รศ.ดร. บุญธีร์ เครือตราชู

ผศ. กฤตวัน ศิริบูรณ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2548

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบเก็บฐานข้อมูลลายมือเขียนผ่านเว็บ

HANDWRITTEN WEB-BASED COLLECTION SYSTEM

ผู้จัดทำ

1. นายกิตติคุณ นันตา รหัสนักศึกษา 45010049

2. นายคุณากร โสรเนตร รหัสนักศึกษา 45010086

อาจารย์ที่ปรึกษา

(รศ.ดร. บุญธีร์ เครือตราชู)

อาจารย์ที่ปรึกษา

(ผศ. กฤตวัน ศิริบูรณ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบเก็บฐานข้อมูลลายมือเขียนผ่านเว็บ

กิตติคุณ นันตา	45010049
คุณากร โสรเนตร	45010086
รศ.ดร. บุญธีร์ เครือตราชู	อาจารย์ที่ปรึกษา
ผศ. กฤตวัน ศิริบูรณ์	อาจารย์ที่ปรึกษาร่วม
ปีการศึกษา 2548	

บทคัดย่อ

เทคโนโลยีการนำเข้าข้อมูลด้วย pen-based คือการใช้ปากกาเขียนตัวอักษรด้วยลายมือ หรือเขียนสัญลักษณ์เป็นคำสั่งป้อนให้เครื่อง หรือวาดภาพลงบนเครื่อง ถึงแม้จะมีใช้งานแล้วในปัจจุบัน แต่สาเหตุหนึ่งที่การนำเข้าข้อมูลตัวอักษรด้วยวิธี pen-based ยังไม่เป็นที่นิยมแพร่หลาย เนื่องมาจากขั้นตอนการป้อนข้อมูลนั้น ยังใช้งานได้ไม่ง่ายเท่าที่ควร โดยเฉพาะอย่างยิ่งในการใช้งานกับภาษาไทย ที่มีรูปแบบตัวอักษร สระ วรรณยุกต์ ที่ยากต่อการรู้จำ ดังนั้น ณ ขณะนี้จึงยังมีการพัฒนากระบวนการรู้จำลายมือเขียนอยู่อย่างต่อเนื่อง และในการพัฒนานี้ ก็มีความจำเป็นที่จะต้องใช้ตัวอย่างลายมือเขียนเป็นจำนวนมาก และมีความแตกต่างกัน คือต้องมาจากการเขียนของคนหลายๆ คนเพื่อนำมาใช้ในเป็นข้อมูลสำหรับการ train นอกจากนี้ยังต้องหาอาสาสมัครมาทดสอบอีก ซึ่งเป็นสิ่งที่ลำบากและเสียเวลาไม่น้อย ดังนั้น ในโครงงานฉบับนี้จึงได้เสนอการทำระบบจัดเก็บลายมือเขียนโดยใช้เว็บเป็นสื่อ ซึ่งผู้ที่เป็นอาสาสมัครสามารถเข้ามาใช้เวลาไหนก็ได้จากที่ไหนก็ได้ และยังสามารถใช้ได้หลายๆคนพร้อมๆกัน ซึ่งจะช่วยลดระยะเวลาของการพัฒนาไปได้มากทีเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HANDWRITTEN WEB-BASED COLLECTION SYSTEM

Kittikun Nanta	45010049
Kunakorn Soranate	45010086
Assoc.Prof.Dr. Boontee Kruatrathue	Advisor
Asst.Prof. Kritawan Siriboon	Co-Advisor

Academic Year 2005

ABSTRACT

For pen-based computing, handwriting character recognition is one of the major methods used for character input. There are other methods for pen input for example, gestures recognition or touch screen input, however character recognition is the most natural method. Although it is used in worldwide, the algorithm used for character input needs to be improved in order to provide conveniently to user, especially in some Thai characters that consists of difficult alphabets, vowels and tonal accent in Thai writing. Therefore, developing this technology has been continuing improved. And training data that is one of many points for success recognizable is essentially. In this project we provide web application for Thai handwriting character collecting and recognition testing for algorithm creators that will help them in developing process.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้อย่างดี ด้วยคำแนะนำ และคำปรึกษาจาก รศ.ดร. บุญธีร์ เครือ-ตราชู ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์, ผ.ศ. กฤตวัน ศิริบูรณ์ ข้าพเจ้ารู้สึกทราบซึ้งในความอนุเคราะห์จากท่านอาจารย์ทั้งสองท่าน และขอขอบพระคุณเป็นอย่างสูง

ขอกราบขอบพระคุณคณาจารย์ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ในภาควิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจและให้การสนับสนุนในทุกเรื่องๆ ทำให้ข้าพเจ้าสามารถทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน

กิตติคุณ นันตา

คุณากร โสรเนตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ..... II

กิตติกรรมประกาศ..... III

สารบัญ.....IV

สารบัญรูป.....VI

บทที่ 1 บทนำ.....1

1.1 ความสำคัญและที่มาของโครงงาน..... 1

1.2 วัตถุประสงค์ของโครงงาน.....1

1.3 ขอบเขตของโครงงาน.....2

1.4 ประโยชน์ที่คาดว่าจะได้รับ.....2

1.5 ส่วนประกอบของปริญญานิพนธ์.....2

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง..... 3

2.1 เทคโนโลยี Pen-based computing.....3

2.2 การรู้จำรูปแบบ.....4

2.2.1 การรู้จำรูปแบบของมนุษย์.....4

2.2.2 ระบบของการรู้จำรูปแบบโดยคอมพิวเตอร์.....4

2.2.3 การรู้จำแบบจำแนกตามวิธีการรู้จำ..... 5

2.3 การรู้จำตัวอักษร..... 6

2.3.1 การรู้จำตัวอักษรแบบมีการประมวลผลในขณะเขียน (On-line Recognition)... 6

2.3.2 การรู้จำตัวอักษรแบบไม่มีการประมวลผลขณะเขียน(Off-line Recognition).... 6

2.4 ตัวอักษรที่ใช้ในการรู้จำตัวอักษร.....7

2.4.1 ตัวอักษรตัวพิมพ์ (Printed Character).....7

2.4.2 ตัวอักษรลายมือเขียน (Handwritten Character).....8

บทที่ 3 เทคโนโลยีที่ใช้ในการพัฒนา..... 9

3.1 เทคโนโลยีดอทเน็ต..... 9

3.2 หลักการของซ็อกเก็ต.....11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

3.2.1 รูปแบบของ TCP/IP Software Interface.....12

3.2.2 Interface Functionality.....12

3.2.3 Berkeley Sockets.....13

3.2.4 Working with Sockets.....15

3.3 Java Applet.....16

3.3.1 ข้อจำกัดของ applet.....16

3.3.2 Signed Applet.....17

3.4 Javascript.....18

บทที่ 4 การออกแบบและพัฒนา.....19

4.1 การหาวิธีรับลายมือเขียนผ่านทางการลากเมาส์โดยที่ไม่ต้องคลิกเมาส์.....19

4.2 การออกแบบระบบ.....25

4.2.1 ส่วนรับลายมือเขียน (client).....25

4.2.2 Server.....26

4.2.3 เว็บระบบจัดเก็บฐานข้อมูลลายมือเขียน.....31

บทที่ 5 บทสรุป.....35

บรรณานุกรม.....36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่		หน้า
3.1	แสดงสถาปัตยกรรมของดอทเน็ตเฟรมเวิร์ก	10
3.2	แสดงการแปลจากโค้ดให้อยู่ในรูปของ IL	11
3.3	Diagram ขั้นตอนการสร้าง Passive Sockets	15
3.4	Diagram ขั้นตอนการสร้าง Active Sockets	15
3.5	ตัวอย่างการใช้แท็ก <APPLET> ในเอกสาร HTML	16
3.6	ตัวอย่าง Javascript สำหรับใช้แสดงหน้าเว็บเป็นแบบ popup window	18
4.1	แสดงจุดเริ่มต้นและขอบเขตที่สามารถเขียนตัวอักษรได้	19
4.2	แสดงส่วนของเส้นซึ่งไม่ต้องการและต้องตัดออก	20
4.3	แสดงตัวอักษรที่เขียนต้องมีมุมสำหรับตัดเสมอ	20
4.4	แสดงทิศทางการเขียนที่เป็นไปได้	21
4.5	แสดงการพิจารณาตัดส่วนที่เกินมา	21
4.6	เมื่อแนวโน้มของทิศทางเป็น 1 ถ้าพบทิศ 2 กับ 8 ให้ถือว่าทิศทางยังไม่เปลี่ยน	22
4.7	แสดง statechart ของการเขียนโดยใช้การเซท timer ร่วมกับการใช้คลิกเมาส์	23
4.8	แสดงลำดับการเขียน	24
4.9	แสดง statechart ของการเขียนโดยใช้การเซท timer เพียงอย่างเดียว	24
4.10	แสดง Java Applet สำหรับรับลายมือเขียน	25
4.11	แสดงการทำงานแบบมัลติเทรดในเซิร์ฟเวอร์	27
4.12	แสดงเซิร์ฟเวอร์ขณะรอให้ผู้ใช้ป้อนอัลกอริทึมที่ต้องการใช้ในกระบวนการรู้จำ	28
4.13	แสดงการรอรับการเชื่อมต่อจากไคลเอนท์ของเซิร์ฟเวอร์	29
4.14	แสดงการเขียนลายมือ	29
4.15	แสดงการปิดการเชื่อมต่อของเซิร์ฟเวอร์	30
4.16	แสดงผลการรู้จำลายมือเขียน	30
4.17	แสดงหน้าล็อกอิน	31
4.18	แสดงสถิติการรู้จำลายมือเขียน	32
4.19	แสดงหน้าเพิ่มผู้ใช้	32
4.20	แสดงหน้าแสดงรายชื่อผู้ใช้	33
4.21	แสดงหน้าสำหรับส่งอีเมล์เพื่อแจ้งข่าวสาร	33
4.22	แสดงหน้าตรวจสอบลายมือเขียน	34
4.23	แสดงไฟล์ลายมือเขียน	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มาของโครงงาน

ในปัจจุบันเราพบว่า คอมพิวเตอร์ได้เข้ามามีบทบาทต่อชีวิตประจำวันมากขึ้น และยังมีแนวโน้มที่จะช่วยอำนวยความสะดวกต่อผู้ใช้งานมากยิ่งขึ้นด้วย จึงทำให้มีการพัฒนาระบบเพื่อช่วยให้การติดต่อระหว่างผู้ใช้งานและเครื่องคอมพิวเตอร์มีความง่ายในการใช้มากยิ่งขึ้น ถึงแม้ปัจจุบันนี้จะมีเทคโนโลยีการนำเข้าข้อมูลตัวอักษรอยู่หลายรูปแบบ และเทคโนโลยีที่ใช้งานอย่างแพร่หลายในปัจจุบันนี้คือการใช้คีย์บอร์ด ซึ่งการใช้งานนั้นผู้ใช้ต้องใช้เวลาในการจดจำปุ่มบนแป้นพิมพ์ ทั้งยังต้องฝึกฝนเพื่อให้ใช้งานได้อย่างคล่องแคล่ว ทำให้เกิดความไม่สะดวกต่อผู้ที่เริ่มต้นใช้งาน ดังนั้นวิธีการนำเข้าข้อมูลตัวอักษรที่มีความใกล้เคียงธรรมชาติ และเป็นสิ่งที่ผู้ใช้คุ้นเคยมาก่อน เช่น การใช้ลายมือเขียนของผู้ใช้เอง จึงเป็นสิ่งที่น่าจะช่วยให้ผู้ใช้รู้สึกว่ามีความเป็นธรรมชาติมากขึ้น

เทคโนโลยีการนำเข้าข้อมูลด้วย pen-based คือการใช้ปากกาเขียนตัวอักษรด้วยลายมือหรือเขียนสัญลักษณ์เป็นคำสั่งป้อนให้เครื่อง หรือวาดภาพลงบนเครื่อง ถึงแม้จะมีใช้งานแล้วในปัจจุบัน แต่สาเหตุหนึ่งที่การนำเข้าข้อมูลตัวอักษรด้วยวิธี pen-based ยังไม่เป็นที่นิยมแพร่หลายเนื่องมาจากขั้นตอนการป้อนข้อมูลนั้นยังใช้งานได้ไม่ง่ายเท่าที่ควร โดยเฉพาะอย่างยิ่งในการใช้งานกับภาษาไทยที่มีรูปแบบตัวอักษร สระ วรรณยุกต์ ที่ยากต่อการรู้จำ ดังนั้น ณ ขณะนี้จึงยังมีการพัฒนากระบวนการรู้จำลายมือเขียนอยู่อย่างต่อเนื่อง และในการพัฒนานี้ก็มีความจำเป็นที่จะต้องใช้ตัวอย่างลายมือเขียนเป็นจำนวนมากและมีความแตกต่างกันคือต้องมาจากการเขียนของคนหลายๆคนเพื่อนำมาใช้ในเป็นข้อมูลสำหรับการ train นอกจากนี้ยังต้องหาอาสาสมัครมาทดสอบอีก ซึ่งเป็นสิ่งที่ลำบากและเสียเวลาไม่น้อยสำหรับการหาอาสาสมัครมาทีละคนสองคน

ดังนั้นในโครงงานฉบับนี้จึงได้เสนอการทำระบบจัดเก็บลายมือเขียนโดยใช้เว็บเป็นสื่อซึ่งผู้ที่เป็นอาสาสมัครสามารถเข้ามาใช้เวลาไหนก็ได้จากที่ไหนก็ได้ และยังสามารถใช้ได้หลายๆคนพร้อมๆกัน ซึ่งจะช่วยลดระยะเวลาของการพัฒนาไปได้มากทีเดียว

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อศึกษาหาวิธีการรับข้อมูลลายมือเขียนภาษาไทยโดยผ่านทางเมาส์โดยที่ไม่ต้องคลิกเมาส์

1.2.2 เพื่อศึกษาหลักการทำงานของเวบแอพลิเคชัน

1.2.3 สร้างเวบแอพพลิเคชันสำหรับรับข้อมูลลายมือเขียนภาษาไทยผ่านทางการลากเมาส์ซึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ช่วยทดสอบกระบวนการจดจำลายมือเขียนภาษาไทยและเก็บลายมือเขียนภาษาไทย

1.2.4 สร้างระบบฐานข้อมูลซึ่งเก็บลายมือเขียนภาษาไทยเพื่อนำไปใช้ในการพัฒนาอัลกอริทึมสำหรับรู้จำลายมือเขียนภาษาไทย

1.3 ขอบเขตของโครงงาน

พัฒนาเวบแอพพลิเคชันสำหรับรับข้อมูลลายมือเขียนภาษาไทยผ่านทางการลากเมาส์ซึ่งจะนำไปใช้เป็น training data ของกระบวนการจดจำลายมือเขียนภาษาไทยและทดสอบกระบวนการจดจำลายมือเขียนภาษาไทยไปพร้อมๆกัน

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 เข้าใจหลักการทำงานของเวบแอพพลิเคชันและสามารถสร้างเวบแอพพลิเคชันได้

1.4.2 เข้าใจกระบวนการทำงานของเว็บไคลเอนท์และเว็บเซิร์ฟเวอร์

1.4.3 เวบแอพพลิเคชันสำหรับรับข้อมูลลายมือเขียนภาษาไทยผ่านทางการลากเมาส์ซึ่งช่วยทดสอบกระบวนการจดจำลายมือเขียนภาษาไทย

1.5 ส่วนประกอบของปริญญานิพนธ์

ปริญญานิพนธ์ฉบับนี้ประกอบด้วยเนื้อหา 5 บท ได้แก่

บทที่ 1 บทนำ เป็นการอธิบายโครงงาน วัตถุประสงค์ของโครงงาน ประโยชน์ที่คาดว่าจะได้รับ และขอบเขตของโครงงาน

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง ประกอบด้วย ทฤษฎีพื้นฐานที่เกี่ยวข้องกับโครงงานนี้

บทที่ 3 เทคโนโลยีและเครื่องมือในการพัฒนา

บทที่ 4 การออกแบบและพัฒนา

บทที่ 5 บทสรุป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 เทคโนโลยี Pen-Based Computing

Pen-based computing เป็นเทคโนโลยีในการการติดต่อสื่อสารกับเครื่องคอมพิวเตอร์โดยการใช้ปากกาแทนที่จะใช้คีย์บอร์ดหรือเมาส์ เราสามารถใช้ปากกานี้เขียนตัวอักษรด้วยลายมือ เขียนสัญลักษณ์เป็นคำสั่งป้อนให้เครื่อง หรือวาดภาพลงบนเครื่องได้ง่ายกว่าการใช้อุปกรณ์ป้อนข้อมูลเข้าชนิดอื่นๆ การเชื่อมต่อแบบ pen-based นี้ได้รับความนิยมในการใช้กับเครื่อง PDA (Personal Digital Assistants) เพราะว่าปากกาสามารถพกพาได้สะดวก เทคโนโลยีการรู้จำตัวอักษรทำให้คอมพิวเตอร์สามารถแปลงลายมือเขียนและสัญลักษณ์ต่างๆ เป็นรูปแบบของตัวอักษรแบบเท็กซ์และคำสั่งที่คอมพิวเตอร์เข้าใจได้

Pen-based Computer เป็นเครื่องคอมพิวเตอร์ขนาดเล็กที่สามารถพกพาได้ หรือเรียกว่า Notepad Computer มีการใช้เทคโนโลยีการรู้จำรูปแบบทำให้สามารถรับตัวอักษรแบบลายมือเขียนเข้าสู่เครื่องได้ โดยมี stylus ซึ่งภายในมีวงจรอิเล็กทรอนิกส์ชนิดพิเศษอยู่ ใช้ในการเขียนลงบนหน้าจอของเครื่องหรือบนแท็บเล็ตพีซี ซึ่งทุกวันนี้ด้วยเทคโนโลยีรู้จำรูปแบบทำให้เครื่องสามารถเข้าใจตัวอักษรลายมือภาษาอังกฤษที่เขียนแบบติดกันได้ ถึงแม้ว่าจะยังไม่ถูกต้องเต็มร้อยเปอร์เซ็นก็ตาม [1]

ระบบการจดจำลายมือเขียน (Handwriting-recognition System) สามารถแปลงข้อมูลออกมาเป็นข้อความได้อย่างรวดเร็ว แม้จะเป็นลายมือ (Shaky Hands) ที่ไม่เป็นระเบียบเหมือนกับการพิมพ์ (Block-printing) ก็ตาม ปากกานี้มีความสามารถในการทำงานมากกว่าแป้นพิมพ์

เทคโนโลยีคอมพิวเตอร์ที่รับข้อมูลแบบลายมือได้ถูกนำมาใช้ในคอมพิวเตอร์แบบมือถือและ PDA เครื่องคอมพิวเตอร์เล็กๆ และ PDA เล็กๆ เหล่านี้ได้บรรจุโพรเสซเซอร์ที่ทำงานรวดเร็วและซอฟต์แวร์ที่สามารถจดจำและสามารถแปลงลายมือเขียน ลายมือพิมพ์ และการวาดด้วยมือให้เป็นดิจิตอลได้ พวกมันมี Pressure-sensitive Layer เหมือนกับแผ่นกราฟิกใต้จอภาพ LCD (Liquid Crystal Display) ดังนั้นแทนที่จะเขียนลงในรูปแบบกระดาษ From Fastened to Clipboard หรือ ใช้อุปกรณ์แป้นพิมพ์ เราสามารถใช้ปากกาเพื่อทำการเลือกและส่งไปรษณีย์อิเล็กทรอนิกส์ หรือใส่ข้อมูลด้วยการเขียนโดยตรงบนคอมพิวเตอร์

นอกจากนี้ยังมีอุปกรณ์คล้ายๆ ปากกานี้หลายอย่าง เช่น Digitizer Pen และกราฟิกแท็บเล็ต (Graphic Tablet) เราสามารถเลือก Digital Pen เพื่อเป็นอุปกรณ์ชี้ หรือใช้เพื่อวาดหรือเขียนลงบนพื้นผิวหน้าจอที่ไวต่อการสัมผัส (Pressure-sensitive Surface) ของกราฟิกแท็บเล็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลายมือเขียนของเราจะถูกทำให้เป็นดิจิตอลโดยคอมพิวเตอร์ซึ่งจะป้อนข้อมูลเข้า แสดงบนจอภาพ และนำเข้าสู่ระบบงานในการใช้งานของเรา

2.2 การรู้จำรูปแบบ

มนุษย์มีความสามารถในการรู้จำได้จากการเห็นภาพ การได้ยินเสียง การสัมผัส การได้กลิ่น และการรู้รส ซึ่งทำให้มนุษย์สามารถแยกแยะวัตถุต่าง ๆ ที่อยู่รอบตัวที่มีรูปแบบต่างกันออกจากกันได้

2.2.1 การรู้จำรูปแบบของมนุษย์ สามารถแบ่งเป็น 2 ประเภท คือ [2]

1. การรู้จำในสิ่งที่เป็นรูปธรรม (The Recognition of Concrete Items) การรู้จำประเภทนี้ยังสามารถจำแนกย่อยได้อีกสองกลุ่มคือ การรู้จำในสิ่งที่มีรูปแบบอยู่อย่างถาวร เช่น ตัวอักขระ และวัตถุต่าง ๆ เป็นต้น และการรู้จำในสิ่งที่มีรูปแบบอยู่ชั่วขณะหนึ่ง เช่นคลื่นเสียงอนุกรมเวลา เป็นต้น

2. การรู้จำในสิ่งที่เป็นนามธรรม (The Recognition of Abstract Items) เป็นการรู้จำในสิ่งที่ไม่มีรูปร่าง หรือการรู้จำในสิ่งที่ได้จากความคิด (Conceptual Recognition) เช่น ความสามารถในการรู้จำเหตุผลเก่า ๆ หรือคำตอบของปัญหาได้โดยที่ไม่ต้องนำความสามารถทางตา ทางหู ทางลิ้น หรือทางจมูกมาใช้งาน

2.2.2 ระบบของการรู้จำรูปแบบโดยคอมพิวเตอร์ ประกอบด้วย

1. ส่วนของการนำเข้า ทำหน้าที่ในการแปลงรูปแบบที่ต้องการวิเคราะห์ให้เป็นสัญญาณอิเล็กทรอนิกส์ อุปกรณ์นำข้อมูลเข้าที่นิยมใช้ ได้แก่ กล้องวีดีโอ (Video Camera) เครื่องอ่านพิกัดภาพ (Image Digitizers) เครื่องสแกน เป็นต้น

2. ส่วนการประมวลผลขั้นต้น เป็นการกระทำกับสัญญาณตามเงื่อนไขที่เพิ่มเติมขึ้น และอาจรวมถึงหน้าที่ในการขยายสัญญาณสแปเชียลฟิลเตอร์ริง (Spatial Filtering) การวิเคราะห์สเปคตรัม และการแปลงสัญญาณอนาลอกไปเป็นสัญญาณดิจิตอล เป็นต้น

3. ส่วนของการเฟ้นหาตัวบรรยายลักษณะ หรือเรียกว่าส่วนของการแสดงให้เห็นความแตกต่าง (Discriminators) ทำหน้าที่จับคู่ต้นแบบ (Template Matching) เป็นต้น

4. ส่วนของการเลือกผลลัพธ์ (Response Selector) หรือส่วนของการรู้จำรูปแบบเป็นขั้นตอนของการเปรียบเทียบคุณลักษณะพิเศษของสิ่งที่ทำการรู้จำ เพื่อเลือกรูปแบบเก็บไว้ในกลุ่มที่เหมาะสม ซึ่งเป็นรูปแบบที่เหมือนกับรูปแบบที่ต้องการรู้จำมากที่สุด

5. ส่วนของการแสดงผลลัพธ์ เป็นกระบวนการแสดงผลของสิ่งที่ได้ผ่านกระบวนการรู้จำ ระบบการแสดงผลลัพธ์อาจใช้ ตัวกำเนิดเสียง รูปภาพ หรือจอภาพเป็นสื่อในการแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 การรู้จำแบบจำแนกตามวิธีการรู้จำ สามารถแบ่งเป็น 4 วิธีคือ [3]

1. วิธีการเทียบคู่รูปแบบ เป็นวิธีการแรก ๆ ที่ใช้ในการรู้จำตัวอักษร หลักการโดยทั่วไปคือ จะต้องมีต้นแบบที่สร้างขึ้นมาสำหรับอ่านตัวอักษรปกติจะเป็นข้อมูล 2 มิติ โดยมีการกำหนดตำแหน่งที่สำคัญที่สามารถใช้แยกแยะความแตกต่างระหว่างตัวอักษรแต่ละตัว การทำงานของระบบจะนำรูปภาพที่ต้องการไปเปรียบเทียบกับต้นแบบเพื่อวัดความคล้ายคลึงของภาพกับตัวแบบ จากนั้นก็ระบุว่าเป็นรหัสของตัวอักษรอะไร วิธีการนี้ค่อนข้างจะอ่อนไหวต่อข้อมูลแทรกซ้อนขนาดของภาพ และการเอียงของตัวอักษร

2. วิธีการทางสถิติ เป็นวิธีการที่ใช้ความน่าจะเป็น และ/หรือ ฟังก์ชันความน่าจะเป็นมาใช้ในการตัดสินใจรูปภาพอินพุตที่ได้มาจากขั้นตอนการสกัดลักษณะสำคัญ และจะส่งเข้าไปในส่วนการรู้จำเฉพาะของแต่ละตัวอักษร ซึ่งได้ผลลัพธ์ออกมาเป็นค่าความน่าจะเป็นที่อินพุตเป็นตัวอักษรใด เมื่ออินพุตได้ผ่านส่วนการรู้จำได้ครบทุกตัวแล้วก็นำเอาผลลัพธ์ที่ได้ทั้งหมดมาเปรียบเทียบกันว่าได้ค่าความน่าจะเป็นของตัวใดมากที่สุด ผลลัพธ์จะแสดงออกมาเป็นตัวอักษรนั้น วิธีนี้แต่ละต้นแบบจะถูกแทนด้วยจำนวนข้อมูลที่เป็นค่าตัวแทนจากการสกัดลักษณะสำคัญ ซึ่งจะเป็นชุดตัวเลขที่มีสัมประสิทธิ์ต่าง ๆ กันตามที่ผู้ใช้กำหนด

3. วิธีการวิเคราะห์ทางโครงสร้าง เป็นการนำภาพตัวอักษรแต่ละตัวมาวิเคราะห์โครงสร้าง โดยถือว่าตัวอักษรทุกตัวประกอบด้วยองค์ประกอบพื้นฐานที่ได้มาจากการสกัดลักษณะสำคัญที่ส่งมาให้กับขั้นตอนการรู้จำแบบการวิเคราะห์ทางโครงสร้างนี้มักจะใช้ชื่อหรือค่าที่บอกว่าลักษณะโครงสร้างสำคัญนั้นเป็นอะไร เช่น เส้นตรง วงกลม เป็นต้น แทนที่จะเป็นค่าจำนวนจริง ในขั้นตอนการรู้จำลักษณะสำคัญทั้งหลายที่ประกอบเป็นตัวอักษรนั้น จะถูกส่งเข้าไปให้กับส่วนที่ตรวจวิเคราะห์กฎการเขียนตัวอักษร เช่น ฟอร์มอลแกรมมาแมชชีน (Formal Grammar Machine) โครงสร้างกราฟ หรือโครงสร้างต้นไม้ เป็นต้น เพื่อระบุว่าเป็นตัวอะไรซึ่งจะตัดสินโดยการพิจารณาที่รูปแบบการเชื่อมต่อขององค์ประกอบต่าง ๆ เข้ากับตัวอักษรนั้น วิธีการนี้มีข้อดีตรงที่มีความยืดหยุ่นต่อความหลากหลายของตัวอักษรค่อนข้างมาก แต่อย่างไรก็ตามอัตราความถูกต้องของวิธีนี้ขึ้นอยู่กับการสร้างกฎ และการวิเคราะห์กฎที่มีประสิทธิภาพซึ่งเป็นส่วนสำคัญที่สุดของวิธีการนี้

4. วิธีทางโครงข่ายประสาทเทียม เป็นเทคนิคที่พยายามเลียนแบบการทำงานของมนุษย์ ที่มีโครงข่ายเชื่อมต่อกันของหน่วยความจำย่อยจำนวนมากที่สะสมความรู้เอาไว้ ความรู้เหล่านี้จะได้จากการฝึกสอนไว้ก่อน เช่น การสอนให้รู้จักตัวอักษร “ก” ถึง “ฮ” โดยการส่งภาพตัวอักษรเหล่านี้เข้าไปให้ระบบเรียนรู้ พร้อมทั้งบอกว่ามีค่าเป็นรหัสตัวอักษรอะไร โครงข่ายประสาทเทียมจะเรียนรู้ถึงรูปแบบตัวอักษรที่หลากหลายของตัวอักษรเหล่านั้น เพื่อว่าเวลาทำงานจริงจะได้มีความสามารถพอที่จะรับมือกับภาพตัวอักษรในหลาย ๆ รูปแบบ สิ่งที่สอนให้กับโครงข่ายประสาทเทียมไม่จำเป็นต้องเป็นรูปของตัวอักษรอย่างที่เราเห็นก็ได้ ข้อมูลที่ส่งให้มักจะผ่านขั้นตอนการสกัดลักษณะสำคัญ และกระบวนการประมวลผลเบื้องต้นอื่น ๆ ก่อนเสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การรู้จำตัวอักษร

มนุษย์มีความสามารถในการรู้จำได้จากการเห็นภาพ การได้ยินเสียง การสัมผัส การได้กลิ่น และการรู้รส ซึ่งทำให้มนุษย์สามารถแยกแยะวัตถุต่าง ๆ ที่อยู่รอบตัวที่มีรูปแบบ (Pattern) ต่างกันออกจากกันได้ การรู้จำตัวอักษรสามารถที่จะแบ่งแยกย่อยออกเป็นประเภทที่แตกต่างกันเป็นจำนวนมาก และที่มีความสำคัญเป็นอย่างมากก็คือการแยกคุณลักษณะพิเศษของตัวอักษรที่ต้องการนำไปรู้จำ การรู้จำตัวอักษร ซึ่งการรู้จำตัวอักษรสามารถแยกออกเป็น 2 ประเภทคือ [4]

2.3.1 การรู้จำตัวอักษรแบบมีการประมวลผลในขณะเขียน (On-line Recognition)

การรู้จำตัวอักษรแบบมีการประมวลผลในขณะเขียน ข้อความต้นฉบับ ก็คือข้อมูลเข้าและจะทำการรู้จำในขณะนั้น (Real Time) อุปกรณ์นำข้อมูลเข้าที่ใช้กันมากได้แก่ Tablet ซึ่งสามารถที่จะเก็บข้อมูลในตำแหน่งที่ปากกาเคลื่อนไปบนแผ่นดิจิไตเซอร์ ซึ่งตำแหน่งหรือพิกัดที่ใช้ปากกาเขียนข้อมูลจะถูกส่งเป็นค่าพิกัดของจุดไปยังคอมพิวเตอร์โดยตรง เพื่อทำการรู้จำในช่วงเวลานั้น

การรู้จำตัวอักษรแบบมีการประมวลผลในขณะเขียนมีความแตกต่างจากการรู้จำตัวอักษรแบบไม่มีการประมวลผลในขณะเขียน เนื่องจากข้อมูลแบบไดนามิคนั้นมีความรวดเร็ว และสามารถลดระยะเวลาในการเขียนได้มาก ซึ่งเพียงใช้ปากกาเขียนลงในแผ่นดิจิไตเซอร์ ระบบก็สามารถที่จะใช้งานได้ในทันที ซึ่งเมื่อใช้ลายมือเขียนที่แตกต่างกันออกไป ข้อมูลแบบไดนามิคนั้นสามารถที่จะแยกแยะความแตกต่างของลายมือ และแสดงผลออกมาได้อย่างถูกต้อง ซึ่งในปัจจุบันวิธีการรู้จำแบบตัวอักษรมีการประมวลผลในขณะเขียนนี้ ได้พัฒนามาแทนที่วิธีการรู้จำตัวอักษรแบบไม่มีการประมวลผลในขณะเขียนเป็นที่เรียบร้อยแล้ว ในส่วนของการปรับปรุงคุณภาพของรูปภาพตัวอักษรมีความจำเป็นสำหรับการปรับปรุงแก้ไขเป็นอย่างมาก เพื่อที่ทำให้ข้อมูลแบบไดนามิคมีการรู้จำที่มีความถูกต้องแม่นยำมากขึ้น

โปรแกรมหลักของการรู้จำตัวอักษรลายมือเขียนแบบมีการประมวลผลในขณะเขียนมีข้อได้เปรียบก็คือ ระบบการรู้จำตัวอักษรลายมือเขียนสามารถที่จะจำลองเพื่อใช้แทนเมาส์ และคีย์บอร์ด ดังนั้นระบบจึงจำเป็นอย่างมากที่จะใช้ Tablet สำหรับเขียน เพื่อส่งข้อมูลไปทำการรู้จำและผลลัพธ์ที่ได้จากการรู้จำจะถูกส่งกลับไปยังโปรแกรมหลักเพื่อแสดงผลต่อไป

2.3.2 การรู้จำตัวอักษรแบบไม่มีการประมวลผลในขณะเขียน (Off-line Recognition)

การรู้จำตัวอักษรลายมือเขียนถือว่าเป็นเรื่องที่ยุ่งยาก ซึ่งจะขึ้นอยู่กับตัวโปรแกรม (Application) ซึ่งโปรแกรมนี้เป็นระบบซึ่งทำการรู้จำตัวอักษรต้นฉบับเดิมที่เขียนเป็นลายลักษณ์อักษรลงบนกระดาษ จากนั้นแปลงจากแผ่นกระดาษให้อยู่ในลักษณะของรูปภาพสองมิติ ซึ่งเป็นรูปภาพแบบบิตแม็พ ซึ่งเรียกกันว่าการประมวลผลรูปภาพแบบดิจิตอล ซึ่งการรู้จำประเภทนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ หาประโยชน์เชิงการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียกว่าการรู้จำตัวอักษรแบบไม่มีการประมวลผลในขณะเขียน ซึ่งระบบจะยังไม่ทำการรู้จำตัวอักษรในเวลานั้น แต่จะทำการรู้จำเมื่อทำการเขียนตัวอักษรเสร็จสิ้น การรู้จำตัวอักษรแบบไม่มีการประมวลผลในขณะเขียนนั้นเป็นส่วนประกอบหนึ่งของ Optical Character Recognition (OCR)

วิธีการรู้จำตัวอักษรแบบไม่มีการประมวลผลในขณะเขียนไม่เหมาะสำหรับการติดต่อสื่อสารระหว่างมนุษย์ กับเครื่องจักร เพราะไม่ได้อำนวยความสะดวก ไม่ได้ทำการประมวลผลในเวลาขณะนั้นและไม่มีการโต้ตอบกันไปมาระหว่างมนุษย์ กับเครื่องจักร ซึ่งจะเหมาะสำหรับการเปลี่ยนแปลงโดยอัตโนมัติจากเอกสารต้นฉบับไปสู่เอกสารในรูปแบบดิจิตอล บางครั้งอาจต้องมีการปรับปรุงคุณภาพของรูปภาพตัวอักษรโดยคอมพิวเตอร์เสียก่อน ซึ่งโปรแกรม OCR นั้นต้องการที่จะมีระบบที่เป็นอัตโนมัติเพื่อไว้สำหรับรวบรวมข้อมูล จากที่เก็บไว้ในรูปแบบกระดาษ เพื่อให้อยู่ในรูปแบบดิจิตอล ดังนั้นที่ผ่านมาระบบการรู้จำได้มีการพัฒนาโปรแกรมที่เกี่ยวกับตัวเลข แต่ส่วนใหญ่แล้วจะใช้สำหรับเครื่องพิมพ์เอกสาร (Machine-Printed Text) สำหรับโปรแกรมที่เกี่ยวข้องกับตัวอักษรลายมือเขียน ประกอบด้วย เช่นการแยกประเภทของจดหมายโดยอัตโนมัติ การตรวจสอบเอกสารทางด้านการเงิน ได้แก่ เช็ค การตรวจสอบตัวอักษรลายมือเขียนของผู้เขียน การพิสูจน์ลายเซ็นและการคัดชวเลข เป็นต้น

2.4 ตัวอักษรที่นำมาใช้ในการรู้จำตัวอักษร แบ่งเป็น 2 ประเภท คือ [2]

2.4.1 ตัวอักษรตัวพิมพ์ (Printed Character)

รูปแบบของตัวอักษรชนิดนี้จะได้มาจากการพิมพ์ เช่นจากเครื่องพิมพ์ดีด หรืออุปกรณ์การพิมพ์อื่น ๆ โดยมีรูปแบบตัวอักษรที่แน่นอนตามชนิดของแต่ละอุปกรณ์การพิมพ์ การพิมพ์เหล่านี้ ได้แก่ สิ่งพิมพ์ต่าง ๆ เช่น หนังสือ เป็นต้น

การรู้จำตัวอักษรตัวพิมพ์สามารถแบ่งได้เป็น 2 ประเภทคือ

ก) การรู้จำตัวอักษรตัวพิมพ์รูปแบบเดียว (Recognition of Single-Font Characters)

การรู้จำตัวอักษรประเภทนี้มีจุดมุ่งหมาย เพื่อให้เครื่องคอมพิวเตอร์สามารถอ่าน และรู้จำตัวอักษรตัวพิมพ์ได้เฉพาะรูปแบบของตัวอักษรที่เก็บไว้เพียงรูปแบบเดียวเท่านั้น

ข) การรู้จำตัวอักษรตัวพิมพ์หลายรูปแบบ (Recognition of Multiple-Font Characters)

การรู้จำตัวอักษรตัวพิมพ์ประเภทนี้มีจุดมุ่งหมาย เพื่อให้เครื่องคอมพิวเตอร์สามารถอ่านและรู้จำตัวอักษรตัวพิมพ์ได้มากกว่าหนึ่งรูปแบบของตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 ตัวอักษรลายมือเขียน (Handwritten Character)

รูปแบบตัวอักษรลายมือเขียนได้มาจากการเขียนทั่วไปซึ่งไม่สามารถกำหนดรูปแบบที่แน่นอนตายตัวได้ หากแต่จะขึ้นอยู่กับพฤติกรรมการเขียนของแต่ละบุคคล สามารถแบ่งออกเป็น 2 ประเภท คือ [3]

ก) ลายมือเขียนแบบตัวโดด (Isolated Handprint Character)

ตัวอักษรของกลุ่มนี้เป็นลายมือเขียนที่มักจะถูกกำหนดให้เขียนในกรอบที่จักไว้ โดยเขียนครั้งละตัวแยกจากกัน ตัวอย่างเช่น การรู้จำในโปรแกรมโอซีอาร์ที่ใช้แยกซองจดหมาย โดยพิจารณาจากรหัสไปรษณีย์ ซึ่งเขียนเป็นตัวเลขจากลายมือเขียน เป็นต้น การรู้จำลายมือเป็นเรื่องที่ยุ่งยาก เนื่องจากตัวอักษรที่ได้จากการเขียนของแต่ละบุคคลมีลักษณะแตกต่างกันของแต่ละบุคคลมีลักษณะแตกต่างกันมากกว่าตัวอักษรที่ได้จากเครื่องพิมพ์

ข) ลายมือแบบเขียนต่อเนื่อง (Scrip Recognition)

เป็นการรู้จำที่ยากที่สุดในกลุ่มของการรู้จำทั้งหมด การรู้จำแบบนี้บางครั้งเรียกการรู้จำแบบลายมือเขียนอิสระ (Freestyle Handwritten Recognition) เพราะตัวอักษรที่จะต้องรู้จำเป็นตัวอักษรที่เป็นลายมือเขียนโดยไม่มีข้อกำหนดใด ๆ ผู้เขียนสามารถเขียนได้ตามธรรมชาติอย่างต่อเนื่อง ดังนั้นตัวอักษรที่ได้อาจมีเส้นที่ลากเชื่อมตัวอักษรหลาย ๆ ตัวติดกันประกอบกับความแตกต่างอย่างมากของลายมือแต่ละคน จึงส่งผลให้การรู้จำแบบนี้ยากกว่าแบบอื่น ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
เทคโนโลยีและเครื่องมือในการพัฒนา

3.1 เทคโนโลยีดอทเน็ต

ดอทเน็ตเฟรมเวิร์ก คือโครงร่างการพัฒนาโปรแกรมคอมพิวเตอร์ ที่ถูกออกแบบมาเพื่อให้อำนวยความสะดวกในการพัฒนาโปรแกรมสมัยใหม่ที่ใช้งานในระบบเครือข่ายโดยมีเป้าหมายการทำงานหลัก 3 ข้อ ได้แก่

1. การพัฒนาโปรแกรมในรูปแบบของเวบเซอร์วิส ซึ่งจะเป็นหลักของโปรแกรมต่างๆ ที่ใช้งานบนอินเตอร์เน็ต เวบเซอร์วิสจะช่วยให้การติดต่อสื่อสาร ระหว่างแอพลิเคชั่นบนอินเตอร์เน็ตนั้นง่ายขึ้น และเป็นระบบมากยิ่งขึ้น
2. เวบเซอร์วิสขั้นพื้นฐานเช่น การตรวจสอบชื่อผู้ใช้ที่ล็อกอินเข้าสู่ระบบ มีการพัฒนาให้เป็นมาตรฐาน และสามารถนำไปใช้ได้ทั่วไปบนอินเตอร์เน็ต
3. เครื่องคอมพิวเตอร์และอุปกรณ์พกพาต่างๆ ที่ต่อเชื่อมกับอินเตอร์เน็ตได้ เช่น พีดีเอและโทรศัพท์มือถือ จะมีบทบาทและประโยชน์มากขึ้นเมื่อสามารถติดต่อใช้งานโปรแกรมต่างๆ บนอินเตอร์เน็ตได้

การที่จะทำให้ระบบหลายๆ ระบบทำงานต่อเชื่อมกันได้อย่างราบรื่นนั้น ไมโครซอร์ฟได้พัฒนารูปแบบการพัฒนาระบบคอมพิวเตอร์ขึ้นมา เรียกว่า ดอทเน็ตเฟรมเวิร์ก ซึ่งมีผู้คิดค้นพัฒนาจากหลายหน่วยงาน ยกตัวอย่างเช่น ซันไมโครซิสเต็มส์ บริษัทไอบีเอ็ม เป็นต้น แต่ว่า ไมโครซอร์ฟได้นำแนวคิดเหล่านั้นมาออกแบบให้อยู่ในรูปแบบที่สามารถต่อเชื่อมกันได้ง่ายขึ้น และเป็นระบบมากขึ้น โดยดอทเน็ตประกอบไปด้วยส่วนประกอบต่างๆ หลายส่วนด้วยกัน และส่วนประกอบเหล่านี้ถูกออกแบบมาเพื่อให้ทำงานได้เข้ากันได้ดีขึ้น

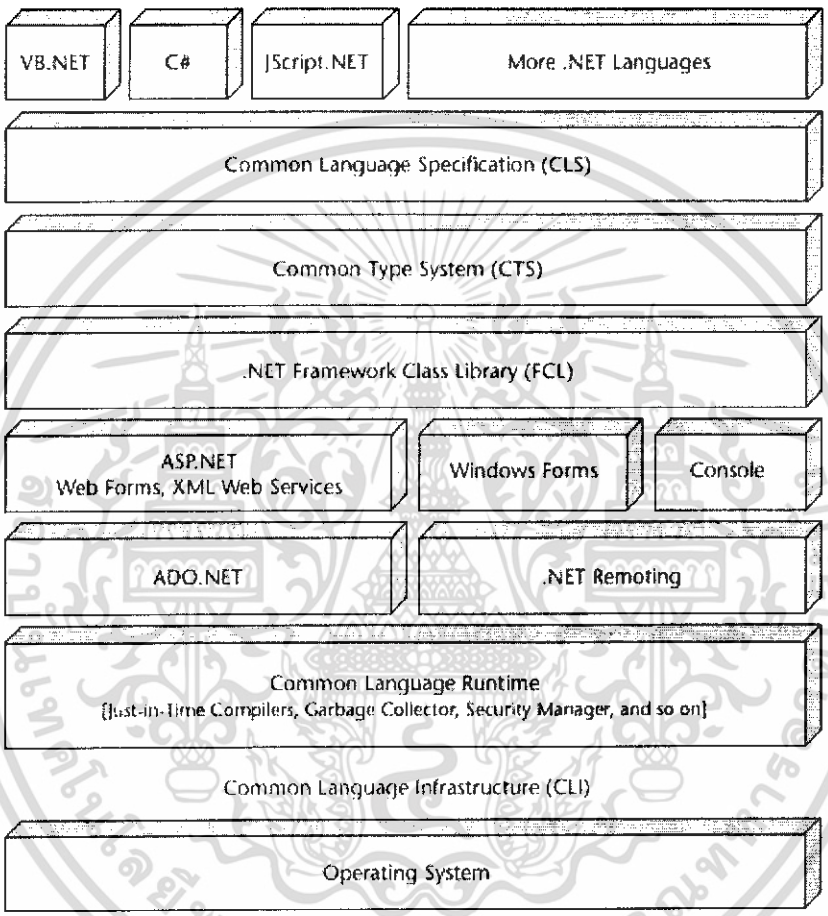
3.1.1 ส่วนประกอบของดอทเน็ตเฟรมเวิร์ก

ส่วนประกอบหลักของ ดอทเน็ตเฟรมเวิร์กแบ่งเป็นชั้นๆ ได้แก่

1. Common Language Runtime (CLR) ถือเป็นรากฐานของแพลตฟอร์มดอทเน็ต โดยเป็นส่วนพื้นฐานที่ติดต่อกับระบบปฏิบัติการวินโดวส์ ทำหน้าที่เป็น run-time environment ให้กับโปรแกรมที่เขียนขึ้นสำหรับใช้บนดอทเน็ต โดยเป็น execution engine ในการประมวลผลและจัดการโปรแกรมที่คอมไพล์แล้ว ให้ทำงานได้บนวินโดวส์ CLR มีส่วนของคอมไพเลอร์ ทั้งที่เป็นแบบปกติ คือคอมไพล์ก่อนที่จะนำโปรแกรมไปใช้ และแบบ Just-In-Time คือคอมไพล์เมื่อจะใช้โปรแกรมนั้นๆ หลังจากนั้นถ้าต้องการนำส่วนอื่นมาใช้งานอีกก็จะแปลงเพิ่มเฉพาะในส่วนนั้น ซึ่งช่วยให้โปรแกรมทำงานได้เร็วขึ้น เนื่องจากไม่ต้องรอให้แปลงเสร็จทั้งหมดก่อนจึงจะทำงานได้ มีส่วนของการจัดการหน่วยความจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ ดัดแปลง เผยแพร่ หรือแสวงหาประโยชน์ทางการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

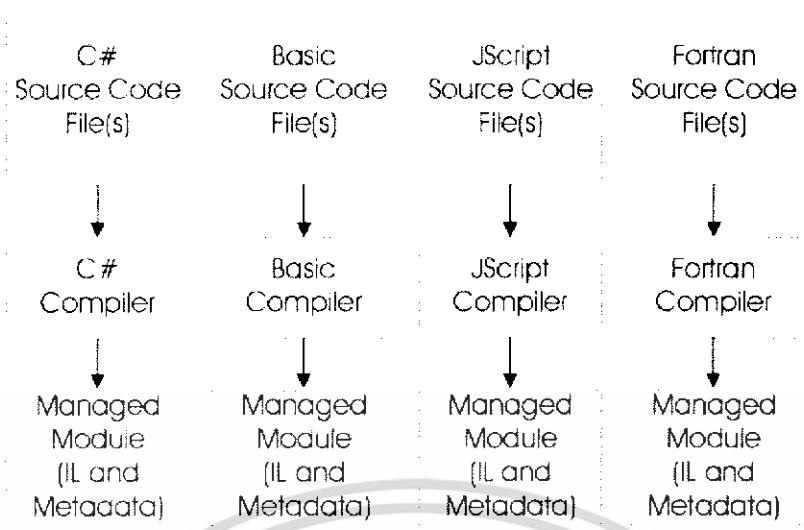
ที่เอาไว้สำหรับจัดสรรหน่วยความจำของเครื่องให้กับโปรแกรม รวมไปถึงการจัดสรรหน่วยความจำให้กับโปรแกรมต่างๆ และคืนหน่วยความจำที่ไม่ถูกใช้งานแล้วให้กับระบบ (Garbage Collection) การจัดการกับข้อผิดพลาดต่างๆ (Exception Handling) รวมถึงดูแลเรื่องความปลอดภัย (security management) ด้วย ส่วนของ Common Type Systems (CTS) ทำให้ภาษาต่าง ๆ ที่เขียนขึ้นบน .Net สามารถทำงานร่วมกันได้ เพราะขนาด และรูปแบบของข้อมูลที่เก็บไว้นั้นเป็นรูปแบบเดียวกัน



รูปที่ 3.1 แสดงสถาปัตยกรรมของดอทเน็ตเฟรมเวิร์ก

2. Base Classes เป็นคลาสไลบรารี่พื้นฐาน ที่โปรแกรมต่างๆ ไม่ว่าจะเขียนด้วยภาษาใดบนดอทเน็ตก็สามารถใช้ร่วมกันได้ เช่น การติดต่อระบบฐานข้อมูล การติดต่อกับ ระบบไฟล์ของเซิร์ฟเวอร์ เป็นต้น
3. Programming Languages เป็นเซ็ตของภาษาคอมพิวเตอร์ ที่ถูกออกแบบมาเพื่อการเขียนโปรแกรมบนดอทเน็ตเฟรมเวิร์ก โดยมีภาษาหลักๆ ได้แก่ VB.Net ซึ่งเป็นตัวที่พัฒนาต่อมาจาก VB C# ซึ่งเป็นภาษาใหม่ที่มีรูปแบบการเขียนใกล้เคียงกับ Java และ C++ Visual C++ และ JScript.Net ส่วนภาษาอื่นๆ นั้น มีบริษัท หรือหน่วยงานอื่นๆ เป็นผู้พัฒนาขึ้น ซึ่งมีอีกจำนวนหลายภาษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 แสดงการแปลจากโค้ดให้อยู่ในรูปของ IL

สำหรับบนดอทเน็ตเฟรมเวิร์กนั้นไม่ว่าจะเขียนโปรแกรมด้วยภาษาใดก็ตามคอมไพล์เลอร์ใน CLR ก็จะคอมไพล์โปรแกรมนั้นให้อยู่ในรูปของ Intermediate Language (IL) ซึ่งจะถูกนำไปแปลเป็นภาษาเครื่อง (Native Code) อีกทีเมื่อตอนที่นำไปใช้

3.2 หลักการของซ็อกเก็ต

โดยทั่วไปในการติดต่อสื่อสารที่เป็นที่นิยมใช้กันในปัจจุบัน จะกระทำบนพื้นฐานของ Protocol TCP/IP และชุดของ Software ซึ่งทำงานกับ Packets ตามมาตรฐานของ TCP/IP นั้นจะถูกสร้างขึ้นมาเป็นส่วนหนึ่งของตัวระบบปฏิบัติการ (Operating System, OS) ดังนั้น เมื่อ Application Software ใด ๆ ที่ต้องการสื่อสารผ่านระบบเครือข่ายตามมาตรฐาน TCP/IP Protocol จะต้องติดต่อกับระบบปฏิบัติการ เพื่อขอรับบริการ (Services)

สำหรับทางผู้พัฒนามาตรฐาน TCP/IP Protocol ไม่ต้องการให้ TCP/IP Protocol ใช้งานได้กับ Platform ใดเป็นการเฉพาะ หรือใช้งานได้กับเพียงระบบปฏิบัติการระบบใดระบบหนึ่ง ดังนั้นผู้พัฒนาจึงมีความระมัดระวังในการกำหนดมาตรฐาน มิให้มีการอ้างถึงข้อมูลการเชื่อมต่อภายในซึ่งเป็นรูปแบบเฉพาะของ Platform ใด Platform หนึ่งหรืออ้างถึงวิธีการเชื่อมต่อกับ Application Software ในลักษณะหรือรูปแบบเฉพาะของระบบปฏิบัติการจากผู้ผลิตรายใดรายหนึ่ง และด้วยแนวคิดในการออกแบบที่ไม่ยึดติดกับ Platform หรือระบบปฏิบัติการใด ๆ นี้ทำให้อาจกล่าวได้ว่า Protocol TCP/IP มีลักษณะเป็น Loosely Specified Protocol Software Interface

จากที่กล่าวมาข้างต้น สามารถกล่าวโดยสรุปได้ว่า "มาตรฐาน TCP/IP นั้นมิได้ระบุรายละเอียด เกี่ยวกับการเรียกใช้งาน TCP/IP Protocol Software ของ Application Software หากแต่มีเพียงข้อแนะนำเกี่ยวกับ functions ที่ควรจะมีในชุดของ TCP/IP Protocol Software เท่านั้น

เอกสารนี้สงวนลิขสิทธิ์ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยในส่วนของรายละเอียดที่ นอกเหนือจากนั้นจะเปิดกว้างให้นักออกแบบระบบเป็นกำหนดและลงรายละเอียดเอง”

ข้อดีของการออกแบบ TCP/IP ให้เป็นแบบ Loosely Specified Protocol Software Interface นั้นคือความ Flexible และ Tolerance ทั้งนี้เพราะการไม่กำหนดรายละเอียดปลีกย่อยทำให้นักออกแบบระบบสามารถที่จะใส่ TCP/IP ลงในระบบปฏิบัติการได้โดยง่ายไม่ว่าระบบปฏิบัติการนั้นจะเป็นระบบปฏิบัติการแบบง่าย ๆ อย่างที่ใช้กันใน Embedded System ไปจนถึงระบบปฏิบัติการที่สลับซับซ้อนอย่างในระบบปฏิบัติการที่ใช้ในเครื่อง Super Computer เป็นต้น นอกจากนั้นด้วยการที่ไม่กำหนดวิธีการเชื่อมต่อกับ Application Software ทำให้นักออกแบบระบบสามารถเลือกใช้รูปแบบในการเชื่อมต่อ (Interface) ได้อย่างอิสระ โดยสามารถใช้ได้ทั้งแบบ Procedural หรือ Message-Passing เป็นต้น

แต่ทั้งนี้ Loosely Specified Software Interface นั้นก็ยังมีข้อด้อยเช่นกัน กล่าวคือการที่นักออกแบบระบบมีอิสระในการใส่รายละเอียดในส่วนการเชื่อมต่อ (Interface) จึงทำให้เกิดความหลากหลายของรูปแบบการเชื่อมต่อ ซึ่งนั่นทำให้การเขียน Application Program เพื่อการใช้งาน TCP/IP มีความ ยุ่งยากและซับซ้อนมากยิ่งขึ้น อีกทั้งยังทำให้ Application Program นั้นยากแก่การ Port ไปยังระบบปฏิบัติการหรือ Platform อื่น ๆ

3.2.1 รูปแบบของ TCP/IP Software Interface

แม้ว่านักออกแบบระบบจะสามารถใส่ส่วนการเชื่อมต่อกับซอฟแวร์ (Software Interface) อย่างไรก็ได้ แต่ในทางปฏิบัติกลับมีรูปแบบของ TCP/IP Software Interface อยู่เพียงไม่กี่แบบเท่านั้น และ TCP/IP Software Interface ที่นิยมใช้งานกันอย่างแพร่หลายจะมีอยู่เพียง 2 รูปแบบเท่านั้นคือ

1. Socket Interface เป็น TCP/IP Software Interface ที่พัฒนาขึ้นโดย University of California at Berkley ซึ่งในตอนต้นได้พัฒนารูปแบบการเชื่อมต่อแบบนี้เพื่อใช้กับระบบปฏิบัติการ Berkley Unix
2. Transport Layer Interface หรือ TLI เป็น TCP/IP Software Interface ที่พัฒนาโดย AT&T เพื่อใช้งานกับระบบปฏิบัติการ System V Unix

3.2.2 Interface Functionality

แม้ว่าทางผู้พัฒนา TCP/IP Protocol Software Interface จะไม่ได้มีการกำหนดรูปแบบหรือวิธีการใด ๆ ในการเชื่อมต่อ Application Software กับ TCP/IP Protocol Software แต่ทางผู้พัฒนาเองก็ได้มีข้อแนะนำเกี่ยวกับ Functions ต่าง ๆ ที่จำเป็น เป็นต้นว่า Interface จะต้องสามารถรองรับการทำงานต่าง ๆ ตามหลักการดังต่อไปนี้ เช่น การจองเนื้อที่ทรัพยากรเพื่อใช้ในการสื่อสารข้อมูล สามารถระบุตำแหน่ง end-point ของต้นทางและปลายทางได้ สามารถสถาปนาการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เชื่อมต่อได้ในกรณีที่เป็นไคลเอนท์ สามารถรอรับการเชื่อมต่อได้ ในกรณีที่เป็นเซิร์ฟเวอร์ สามารถรับและส่งข้อมูลได้ บอกได้ว่าข้อมูลจะมาถึงเมื่อไหร่ มีความสามารถในการยุติการเชื่อมต่อ และจัดการได้เมื่อปลายทางต้องการยุติการเชื่อมต่อ เป็นต้น

นอกจาก Concept ในการทำงานดังที่ได้กล่าวมาแล้วข้างต้น ทางผู้พัฒนามาตรฐาน TCP/IP Protocol ได้สร้าง Conceptual Interface ขึ้นมาเพื่อเป็นแนวทางให้นักออกแบบระบบสามารถพัฒนาซอฟแวร์ของตนได้ง่ายและเร็วขึ้นโดย Conceptual Interface ที่ทางผู้พัฒนามาตรฐาน TCP/IP Protocol ได้สร้างขึ้นนั้นจะอยู่ในรูปของ Set of Procedures and Functions เหตุที่ Conceptual Interface อยู่ในรูปแบบของ Procedures และ Functions นั้น เนื่องมาจากในช่วงเวลาที่พัฒนา Conceptual Interface ระบบปฏิบัติการต่าง ๆ ในยุคนั้นนิยมใช้วิธีการซึ่งเรียกกันว่า Procedural Mechanism ในการเคลื่อนย้ายการควบคุม (Transfer control) จาก Application Software ไปยังระบบปฏิบัติการ ดังนั้นจึงอาจกล่าวโดยสรุปเกี่ยวกับ Conceptual Interface ได้ดังนี้ "Conceptual Interface ซึ่งถูกกำหนดโดยผู้พัฒนามาตรฐาน TCP/IP นั้นจะไม่ระบุรูปแบบของการแทนข้อมูล (Data representation) หรือรายละเอียดในการเขียนโปรแกรม หากแต่แสดงเพียงตัวอย่างหนึ่งของของการเชื่อมต่อ ซึ่งมีความเป็นไปได้ในทางปฏิบัติที่ระบบปฏิบัติการจะสามารถให้บริการแก่ Application Software ที่ต้องการใช้งาน TCP/IP Protocol"

3.2.3 Berkeley Sockets

ช่วงต้นทศวรรษที่ 1980 Advanced Research Projects Agency (ARPA) ได้ให้ทุนกับ University of California at Berkeley เพื่อพัฒนา TCP/IP Software บนระบบปฏิบัติการ UNIX และผลส่วนหนึ่งของโครงการดังกล่าวคือการออกแบบและพัฒนาส่วนการเชื่อมต่อ (Interface) ซึ่ง Application Software จะใช้ในการสื่อสารข้อมูลผ่านเครือข่าย และผลที่ได้รับคือ Berkeley Sockets Interface หรือที่เรียกกันสั้น ๆ ว่า Sockets Interface โดยจุดเด่นของ Sockets Interface นั้นคือการสนับสนุนการสื่อสารข้อมูลผ่านเครือข่ายด้วย Protocols ที่หลากหลายที่สุดเท่าที่จะเป็นไปได้ Sockets Interface ได้ถูกนำไปใส่ในระบบปฏิบัติการสำหรับเครื่อง Work station ระดับสูงจากผู้ผลิตรายใหญ่ อาทิเช่น Sun Microsystems Inc., Tektronix Inc., Digital Equipment Corp. เป็นต้น ทำให้ Sockets Interface กลายมาเป็น Interface ที่สำคัญ และมีผู้ใช้งานอย่างกว้างขวาง จนกระทั่งอาจกล่าวได้ว่า Sockets Interface นั้นได้กลายเป็นมาตรฐานสำหรับการเชื่อมต่อ (Interface) ไปโดยปริยาย

3.2.3.1 System Data Structure for Sockets

เมื่อมีการสร้าง Socket ใหม่ ระบบปฏิบัติการจะทำการ allocate พื้นที่ในหน่วยความจำส่วนหนึ่งเพื่อสร้าง Data Structure สำหรับเก็บข้อมูลซึ่งจำเป็นสำหรับการสื่อสารข้อมูลผ่านเครือข่าย ซึ่ง Data Structure นี้จะประกอบไปด้วย field หลาย field ด้วยกัน โดยปกติเมื่อ Sockets ถูกสร้างขึ้น fields ต่าง ๆ เหล่านี้ใน data structure นั้นจะถูกว่างไว้ Application Program ที่สร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Socket นั้นจะต้องเรียกใช้งาน function สำหรับใส่ข้อมูลต่าง ๆ ลงใน fields ที่เว้นว่างไว้เหล่านั้นเอาเอง

3.2.3.2 Sockets Descriptors

โดยปกติในระบบปฏิบัติการจะมีการสร้าง Descriptor Table สำหรับเก็บ Data Structure ซึ่งเกี่ยวข้องกับ I/O ไว้ ซึ่งปกติ Descriptor Table สำหรับแต่ละ Process จะถูกเก็บแยกออกจากกัน โดยทั่วไป Descriptor Table จะมีลักษณะเป็น Array of Pointer ซึ่งชี้ไปยัง Data Structure ที่เกี่ยวข้องสำหรับในกรณีของ Sockets ก็จะเป็นเช่นเดียวกัน ทั้งนี้เพราะ Sockets นั้นนับเป็น I/O ประเภทหนึ่งด้วยเช่นกัน (I/O ซึ่งติดต่อกับเครือข่าย) ดังนั้นเมื่อมีการสร้าง Socket ก็จะมีการ update Descriptor Table โดยเพิ่ม Pointer ที่ชี้ไปยัง Data Structure ที่เก็บข้อมูลของ Sockets ที่สร้างขึ้นใหม่นั้น

3.2.3.3 การใช้งาน Sockets

Sockets ที่ถูกสร้างขึ้นนั้นจะมีวัตถุประสงค์ในการใช้งานอยู่เพียง 2 ลักษณะได้แก่ การรอรับการเชื่อมต่อและการเชื่อมต่อไปยังเครื่องปลายทาง (Initiate a connection) โดย Sockets ที่รอรับการเชื่อมต่อนั้นจะเป็น Sockets ที่ใช้ใน Server Application ซึ่ง Sockets ชนิดนี้มีชื่อเรียกว่า Passive Socket ในขณะที่ Sockets ที่พยายามเชื่อมต่อไปยังเครื่องปลายทางซึ่งใช้ใน Client Application นั้นจะมีชื่อเรียกว่า Active Socket ทั้ง Active Sockets และ Passive Sockets นั้นในชั้นแรกจะถูกสร้างขึ้นมาด้วยวิธีการเดียวกัน ดังนั้นข้อแตกต่างของ Active Sockets และ Passive Sockets จึงอยู่ที่วัตถุประสงค์การใช้งานเท่านั้น

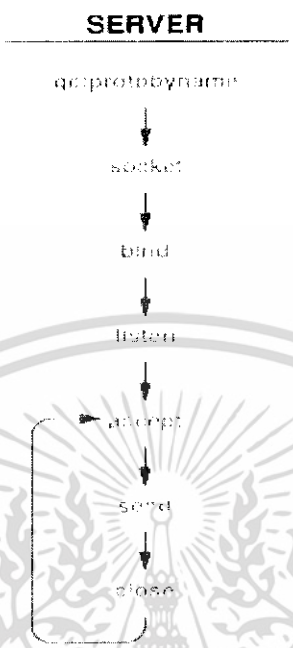
โดย Sockets สามารถแบ่งออกเป็นประเภทตามลักษณะการรับส่งข้อมูลผ่านเครือข่ายได้เป็น 3 ประเภท ดังนี้

1. Stream Sockets หรือ Connection Oriented Sockets
2. Datagram Sockets หรือ Connectionless Sockets
3. Raw Sockets

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4 Working with Sockets

3.2.4.1 Passive Sockets หรือ Sockets ซึ่งใช้ใน Server Application นั้นมีขั้นตอนการสร้างดังนี้



รูปที่ 3.3 Diagram ขั้นตอนการสร้าง Passive Sockets

3.2.4.2 Active Sockets หรือ Socket ที่ใช้งานใน Client Application มีขั้นตอนการสร้างดังนี้

รูปที่ 3.4 Diagram ขั้นตอนการสร้าง Active Sockets

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 Java Applet

Applet เป็นโปรแกรมที่เขียนขึ้นโดยภาษา Java ซึ่งเวลาทำงานจะไม่เอ็กซีคิวต์เป็นโปรแกรมเดี่ยวๆ แต่จะต้องทำงานภายใต้โปรแกรมอื่นไม่สามารถควบคุมการดำเนินของตัวเองได้ทั้งหมด เช่น เว็บบราวเซอร์โดยการฝังไปกับเอกสาร HTML applet จะถูกเก็บไว้ที่เว็บเซิร์ฟเวอร์และจะถูกเรียกใช้งานโดยไคลเอนต์ผ่านทางโปรโตคอล HTTP เหมือนกับว่า applet เป็นส่วนหนึ่งในคำสั่ง HTML ของเว็บเพจนั้นเช่นเดียวกับการใส่รูปภาพเข้าไปในหน้าเว็บเพจ โดยจะใช้แท็ก <APPLET> และ </APPLET> เป็นตัวกำหนดขอบเขตของ applet ดังรูปที่ 3.8 ซึ่งคุณสมบัติการทำงานภายใต้บราวเซอร์นี้ทำให้สามารถสร้างโปรแกรมเพื่อให้ใครก็ได้เข้ามาใช้ผ่านทางหน้าเว็บเพจได้

```
<APPLET CODE="Appletclass.class" WIDTH="100" HEIGHT="50">
</APPLET>
```

รูปที่ 3.5 ตัวอย่างการใช้แท็ก <APPLET> ในเอกสาร HTML

บราวเซอร์ที่เอ็กซีคิวต์ applet ทั่วไปเรียกว่า applet container ตัวอย่างเช่น Appletviewer ซึ่งเป็น applet container ที่มาพร้อมกับ Java 2 Software Development Kit (J2SDK) มีไว้สำหรับใช้ทดสอบ applet ที่กำลังอยู่ในขั้นตอนการพัฒนา ก่อนที่จะนำไปใส่ไว้ในเว็บเพจ ในขณะที่เราใช้เว็บบราวเซอร์ที่อนุญาตให้ใช้เทคโนโลยี Java ในการแสดงหน้าเว็บที่มี applet ฝังมาด้วย โคด applet จะถูกส่งมายังระบบของเราและจะถูกเอ็กซีคิวต์โดย Java Virtual Machine (JVM) ของเว็บบราวเซอร์

3.3.1 ข้อจำกัดของ applet

เนื่องจาก applet เป็นโปรแกรมที่ถูกแจกจ่ายไปในระบบเครือข่าย ดังนั้นจึงต้องมีข้อจำกัดบางอย่างเพื่อไม่ให้มีการนำเอา applet ไปใช้ในทางที่จะก่อให้เกิดความเสียหายกับข้อมูลหรือระบบของเครื่องคอมพิวเตอร์ที่รับ applet นั้นไปทำงาน

โดยปกติแล้วแล้วบราวเซอร์จะถือว่า applet ที่รับมาจากเครื่องอื่นในระบบเครือข่ายเป็นโปรแกรมที่ไม่น่าไว้วางใจ (untrusted applet) และจะให้ applet ทำงานภายใต้สภาพแวดล้อมที่ปลอดภัยซึ่งมีการติดตั้ง Security Manager ให้คอยตรวจสอบการทำงานของ applet และไม่ยอมให้มีการละเมิดข้อจำกัด

ข้อจำกัดของ applet ถ้ามีการนำเอาไปใช้ จะเป็นตัวที่ทำให้ applet ไม่น่าไว้วางใจ ซึ่งเว็บบราวเซอร์แต่ละตัวอาจจะมีข้อจำกัดที่แตกต่างกันไป และอาจจะยอมให้ผู้ใช้เลือกผ่อนปรนข้อจำกัดบางอย่างได้ แต่อย่างไรก็ตามเราก็ต้องถือว่า untrusted applet ถูกจำกัดไม่ให้มีกระทำดังต่อไปนี้ [5]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ข้อจำกัดเกี่ยวกับระบบไฟล์ของเครื่องที่รับ applet นั้นไปทำงาน (local filesystem)
   - ห้ามเขียน อ่าน หรือลบไฟล์ หรือแม้แต่แก้ไข permission ของไฟล์
   - ห้ามสร้างไดเร็กทอรีหรือการจัดการเกี่ยวกับไดเร็กทอรี
   - ห้ามเขียนหรืออ่านไฟล์จากออปเจ็ก FileDescriptor
2. ข้อจำกัดเกี่ยวกับระบบเครือข่าย
   - ห้ามสร้างการเชื่อมต่อไปยังเครื่องอื่นที่ไม่ใช้เครื่องที่ applet นั้นถูกโหลดมา
   - ห้าม listen หรือ accept ในพอร์ต (port) ที่หมายเลขน้อยกว่าหรือเท่ากับ 1024
   - ห้ามให้ multicast socket
   - สร้างหรือรีจิสเตอร์ SocketImplFactory, URLStreamHandlerFactory หรือ ContentHandlerFactory
3. ไม่สามารถใช้ฟังก์ชันบางอย่างของระบบได้
   - ออกจาก Java interpreter โดยการเรียกใช้คำสั่ง System.exit() หรือคำสั่ง Runtime.exit()
   - สร้างโปรเซสใหม่โดยการเรียกผ่านทางคำสั่ง Runtime.exec()
   - โหลด native code library ด้วยการเรียกให้เมทธอด load() หรือ loadLibrary() ของคลาส Runtime หรือคลาส System

นอกจากนี้ยังมีข้อจำกัดของ applet อีกมากแต่จะขอยกมาให้เห็นเพียงเท่านี้ก่อน

เว็บบราวเซอร์และ applet viewer อาจจะมีการผ่อนปรนข้อจำกัดบางอย่างให้กับ applet ที่ถูกโหลดมาจาก local filesystem บ้าง เพราะถือว่า local applet ต้องมีความน่าไว้วางใจมากกว่า applet ที่ถูกโหลดมาจากเครือข่ายและให้ถือเป็นความรับผิดของผู้ใช้เอง

3.3.2 Signed Applet

ใน Java 1.1 ได้มีการเพิ่มความสามารถในการลงลายมือชื่อดิจิตอล (digital signature) ให้กับไฟล์ .jar (Java achive file) ซึ่งเป็นไฟล์ที่เก็บรวมไฟล์ .class และไฟล์ที่จำเป็นอื่นๆ ของ applet ไว้ในไฟล์เพียงไฟล์เดียวด้วย ลายมือชื่อดิจิตอลนี้ใช้ระบุถึงเจ้าของหรือแหล่งที่มาของ applet ซึ่งเราสามารถใช้ในการตัดสินใจได้ว่า applet นั้นมีความน่าไว้วางใจมากน้อยแค่ไหน ถ้าเราไว้ใจในแหล่งที่มาของ applet เราก็สามารถที่จะบอกให้เว็บบราวเซอร์หรือ applet viewer ว่า applet นั้นมีความน่าไว้วางใจ (trusted applet) ซึ่งจะทำให้ไม่ถูกจำกัดเหมือน untrusted applet

กระบวนการของลงลายมือชื่อดิจิตอลลงไปในไฟล์ .jar ของ applet ขึ้นอยู่กับแพล็ตฟอร์ม โดยใน Java 1.1 จะใช้โปรแกรม javakey ในขณะที่ Java 1.2 ใช้ jarsigner แทน Netscape และ Microsoft ต่างก็มีโปรแกรมที่ใช้สำหรับลงลายมือชื่อดิจิตอลเองเหมือนกัน เพื่อให้เหมาะสมกับบราวเซอร์ของตนเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

62394

3.4 Javascript

ในการสร้างเว็บเพจที่ต้องการให้มีการโต้ตอบกับผู้ใช้ได้อย่างทันทีทันใดนั้น สามารถทำได้โดยการใช้ภาษาสคริปต์ (Scripting Language) ร่วมกับการทำงานของ HTML ภาษาสคริปต์นี้เป็นชุดคำสั่งที่ไม่ใช่ HTML แต่ถูกแทรกไว้ในส่วนหนึ่งเอกสาร HTML และสามารถทำงานร่วมกับคำสั่งหรือแท็กของ HTML ได้

Javascript ก็เป็นภาษาสคริปต์ภาษาหนึ่งที่พัฒนาขึ้นโดย Sun และแต่เดิมเรียกว่า Livescript และได้ถูกเปลี่ยนชื่อเป็น Javascript เมื่อปี ค.ศ. 1995 โครงสร้างภาษาของ Javascript มีความคล้ายคลึงกับ Java มาก โดย Javascript เป็น complement ของ Java สามารถติดต่อกับส่วนต่างๆ ของ Java Applet โดยสคริปต์ที่เขียนขึ้นมาได้ คำสั่งของ Javascript สามารถนำมาใช้แสดงกำหนดคุณสมบัติ สอบถามสถานะ หรือควบคุมการกระทำของ Applet และปลั๊กอิน นอกจากนี้ Javascript ยังสนับสนุนรูปแบบนิพจน์และการควบคุมพื้นฐานของภาษา Java อีกด้วย

Javascript ได้ถูกออกแบบมาเพื่อใช้เป็นส่วนเพิ่มขยายในภาษา HTML โดยเฉพาะ ช่วยให้สามารถควบคุมเว็บเพจได้อย่างง่ายดายเหมาะกับการทำงานอย่างรวดเร็ว และเน้นที่ความถูกต้องเป็นสำคัญ เนื่องจาก Javascript เป็นภาษาแบบ interpleter ดังนั้นเราจึงสามารถใช้งาน Javascript ร่วมกับเอกสาร HTML โดยแทรกสคริปต์ลงไปในหน้าเอกสาร HTML ได้เลย โดยจะต้องระบุส่วนที่เป็น Javascript ด้วยแท็ก <SCRIPT LANGUAGE="Javascript"> ดังรูปที่ เลยก็ได้ ซึ่งในขณะที่บราวเซอร์ทำงานจะส่งสคริปต์นี้ไปให้ตัวแปรสคริปต์อีกทีหนึ่ง

```
<script language="javascript">
<!--
  function popup(mylink, windowname)
  {
    if (! window.focus) return true;
    var href;
    if (typeof(mylink) == 'string')
      href=mylink;
    else
      href=mylink.href;
      window.open(href, windowname, 'width=950, height=550, scrollbars=no');
                                        return false;
  }
-->
</script>
```

รูปที่ 3.6 ตัวอย่าง Javascript สำหรับใช้แสดงหน้าเว็บเป็นแบบ popup window

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

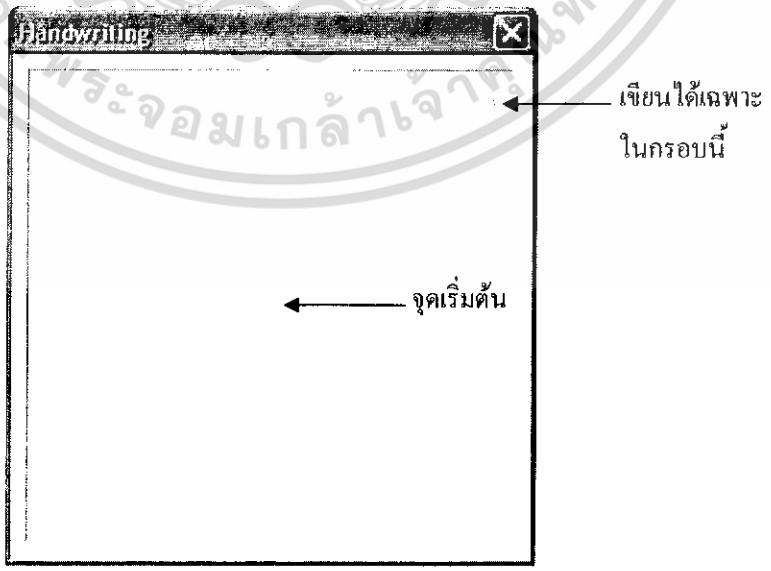
การออกแบบและพัฒนา

4.1 การหาวิธีรับลายมือเขียนผ่านทางการลากเมาส์โดยที่ไม่ต้องคลิกส์เมาส์

จากการที่เราต้องการหาวิธีรับลายมือเขียนโดยผ่านทางการลากเมาส์โดยที่ไม่ต้องคลิกเมาส์ เนื่องจากการคาดหวังว่าเมาส์ซึ่งเป็นอุปกรณ์ที่เครื่องคอมพิวเตอร์ทุกเครื่องในปัจจุบันจำเป็นต้องมีจะสามารถนำมาใช้เขียนตัวอักษรแทนการใช้ปากกาได้ ซึ่งวิธีการเดิมที่ใช้กันมาจะใช้การแคร็กเมาส์ (การกดเมาส์แล้วลาก) แต่การที่ต้องกดปุ่มของเมาส์ขณะลากไปด้วยทำให้เขียนลำบาก ขาดความคล่องตัว ทำให้เราคิดว่าน่าจะลองหาวิธีที่ไม่ต้องคลิกเมาส์ดู

เราได้ทำการศึกษาหาวิธีการรับลายมือเขียนที่คิดว่าเป็นวิธีที่เหมาะสม โดยมุ่งเน้นไปที่การใช้เมาส์ลากโดยที่ไม่ต้องคลิกเมาส์ ซึ่งการที่ไม่ต้องคลิกเมาส์นี้ทำให้การลากเมาส์เป็นไปอย่างอิสระและสามารถเขียนได้เร็วกว่าการที่ต้องคลิกเมาส์อีกด้วย

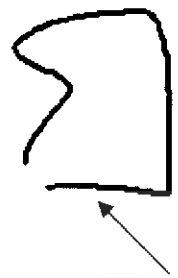
วิธีการที่ได้ใช้ในตอนแรกก็คือได้กำหนดจุดจุดหนึ่งให้เป็นจุดเริ่มต้น การเขียนทำได้โดยลากเมาส์ออกจากจุดนี้และเมื่อลากกลับเมาที่จุดนี้อีกครั้งหนึ่งก็จะถือว่าจบการเขียนตัวอักษรตัวนั้นและพร้อมที่จะเขียนตัวใหม่ต่อไปได้ ในเวลาต่อมาก็ได้มีการกำหนดขอบเขตของการเขียนโดยให้สามารถเขียนได้เฉพาะในพื้นที่ที่กำหนดเท่านั้น ถ้ามีการลากเมาส์ออกจากพื้นที่นี้ก็จะถือว่าจบการเขียนตัวอักษรตัวนั้นแล้วเช่นกันและเซทให้ mouse pointer กลับไปอยู่ที่จุดเริ่มต้นและพร้อมที่จะเขียนตัวใหม่ต่อไป การกำหนดขอบเขตนี้ก็ถือเป็นอีกตัวช่วยหนึ่งสำหรับการตัดตัวอักษรแทนที่จะต้องลากเมาส์กลับไปที่จุดเริ่มต้นเพียงอย่างเดียว ดูรูปที่ 4.1 ประกอบ



รูปที่ 4.1 แสดงจุดเริ่มต้นและขอบเขตที่สามารถเขียนตัวอักษรได้

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเขียนโดยใช้วิธีนี้จำเป็นจะต้องมีกระบวนการในการตัดส่วนที่เกินมาซึ่งเป็นส่วนที่เกิดจากการลากเมาส์กลับไปที่จุดเริ่มต้นหรือเกิดจากการลากเมาส์ออกจากขอบเขตที่กำหนด ดูรูปที่ 4.2 ประกอบ



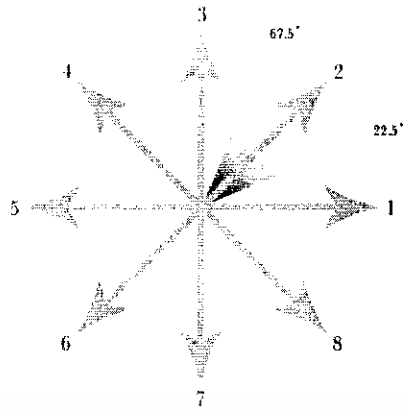
รูปที่ 4.2 แสดงส่วนของเส้นซึ่งไม่ต้องการและต้องตัดออก

วิธีการที่เราได้นำมาใช้สำหรับการตัดส่วนที่เกินมาก็คือ ใช้การพิจารณาจากทิศทางการเขียนว่ามีการเปลี่ยนทิศหรือไม่ ซึ่งจะเริ่มพิจารณาจากจุดท้ายสุดก่อนแล้วไล่ต่อขึ้นไปเรื่อยๆ ถ้าทิศทางการเขียนเปลี่ยนตรงไหนก็จะเริ่มตัดตั้งแต่จุดนั้นไปจนถึงจุดสุดท้ายทิ้งไป (การเก็บข้อมูลของลายมือเขียน เบื้องต้นได้เก็บเป็นลิสของคู่อันดับ x กับ y) ดังนั้นจึงมีความจำเป็นที่ตัวอักษรที่เขียนขึ้นจะต้องมีมุมซึ่งเอาไว้บอกว่าเป็นจุดที่จะต้องตัดเสมอ ดังรูปที่ 4.3 ไม่เช่นนั้นแล้วการตัดตัวอักษรอาจเกิดความผิดพลาดได้

รูปที่ 4.3 แสดงตัวอักษรที่เขียนต้องมีมุมสำหรับตัดเสมอ

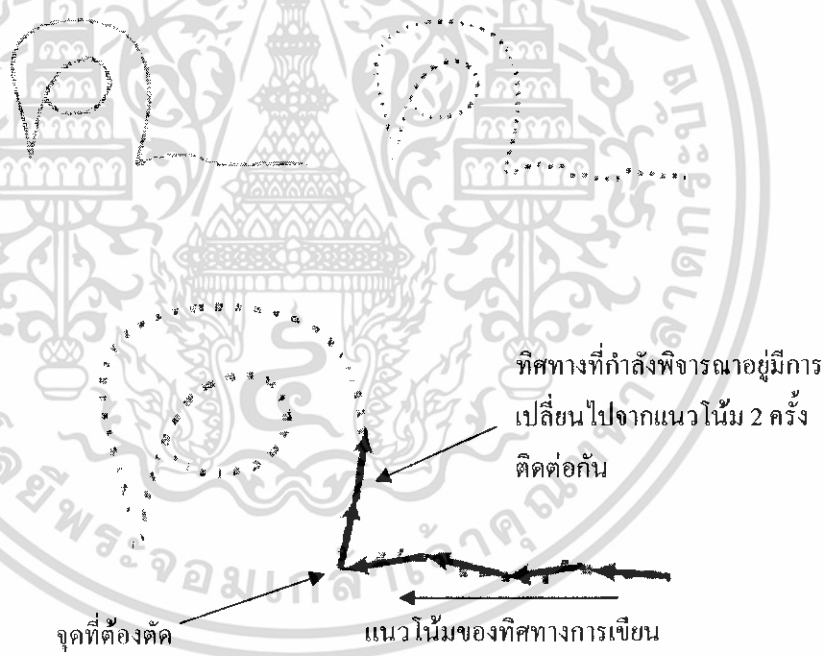
สำหรับทิศทางการเขียนเราได้แบ่งออกเป็น 8 ทิศทางแต่ละทิศทางครอบคลุมบริเวณใกล้เคียงรวม 45° ดังรูปที่ 4.4 ซึ่งทิศทางดังกล่าวหาได้จากการหามุมระหว่างจุดสองจุดเทียบกับแนวนอน โดยในที่นี้เราเลือกจุดที่อยู่ห่างกัน 4 ช่วงจุด (เช่น 1 กับ 5 และ 5 กับ 9 ต่อไปเรื่อยๆ) เมื่อได้มุมมาแล้วเราก็จะพิจารณาว่าอยู่ในทิศทางไหน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 แสดงทิศทางการเขียนที่เป็นไปได้

ในการที่จะตัดสินว่าเกิดการเปลี่ยนทิศทางการเขียนที่จุดไหน เราจะดูจากแนวโน้มของทิศทางว่าโดยรวมแล้วทิศทางกำลังไปทางไหนแล้วทิศทางที่กำลังพิจารณาอยู่มีการเปลี่ยนไปจากแนวโน้มหรือไม่ ถ้าทิศทางมีการเปลี่ยนไปจากแนวโน้ม (ในที่นี้เรากำหนดว่าต้องเปลี่ยนติดต่อกัน 2 ครั้ง) ก็จะถือว่าจุดนั้นเป็นจุดที่มีการเปลี่ยนทิศทาง ดังรูปที่ 4.5

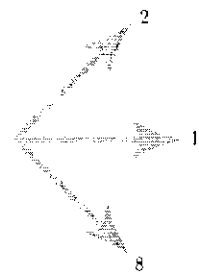


รูปที่ 4.5 แสดงการพิจารณาตัดส่วนที่เกินมา

ในทางปฏิบัติเราได้พบว่า ในการลากเมาส์อย่างช้าๆมีโอกาสเป็นไปได้สูงมากที่ทิศทางที่กำลังพิจารณาอยู่จะถูกตัดสินว่าได้เปลี่ยนทิศไปแล้วแต่ถ้าดูจากแนวโน้มแล้วมันไม่ได้เปลี่ยน ที่เป็นแบบนี้มีเนื่องมาจากการที่เราได้พิจารณาทิศทาง ทีละ 4 ช่วงจุด เมื่อมีการลากเมาส์อย่างช้าๆ จะทำให้จุดที่อ่านได้อยู่ชิดกันมากเมื่อนำมาหาทิศทางจึงทำให้ได้ทิศทางที่ผิดไปจากแนวโน้มมาก ดังนั้นเราจึงได้แก้ไขโดยการให้ทิศทางที่ทำให้เกิดการตีความว่าเกิดการเปลี่ยนทิศไปแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลายเป็นว่ายังคงไม่มีการเปลี่ยนทิศ ดังรูปที่ 4.6 เมื่อแนวโน้มของทิศทางเป็น 1 ถ้าพบทิศ 2 กับ 8 ให้ถือว่าทิศทางยังไม่เปลี่ยน



รูปที่ 4.6 เมื่อแนวโน้มของทิศทางเป็น 1 ถ้าพบทิศ 2 กับ 8 ให้ถือว่าทิศทางยังไม่เปลี่ยน

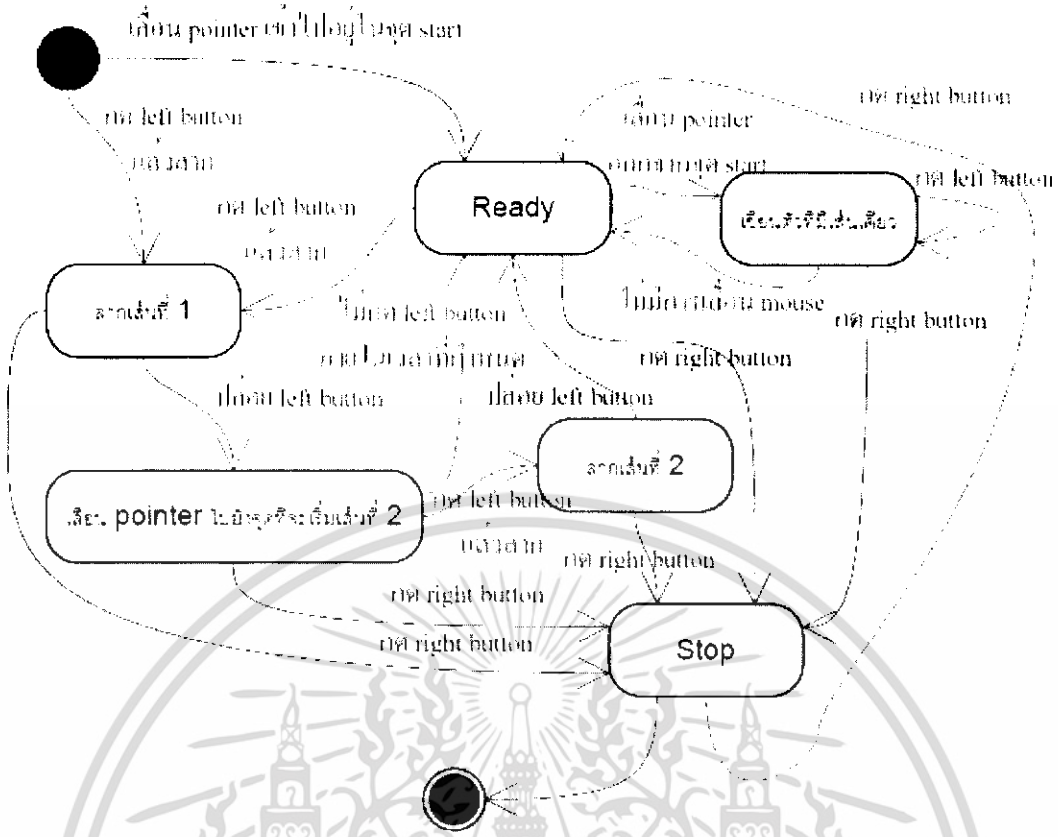
นอกจากวิธีการข้างต้นแล้ว เรายังได้มองหาวิธีการใหม่ที่คิดว่าน่าจะดีกว่า ซึ่งเราก็ได้นำเอา timer เข้ามาเป็นตัวช่วย โดยให้ถือว่าในขณะที่มีการลากเส้นถ้ามีการหยุดเลื่อนเมาส์ (ไม่มีการเลื่อนเมาส์เป็นช่วงเวลาหนึ่งซึ่งเท่ากับ timer ที่ตั้งไว้) ก็จะให้ mouse pointer กลับไปเริ่มที่จุดเริ่มต้นใหม่ทันที ซึ่งการใช้ timer นี้ได้ผลเป็นที่น่าพอใจมากทีเดียว คือ สามารถเขียนได้อย่างรวดเร็วและมีความเป็นธรรมชาติมากกว่าวิธีแรกและที่สำคัญคือไม่มีส่วนที่ลากเกินมาซึ่งไม่ต้องการด้วย

อย่างไรก็ตามทั้งสองวิธีข้างต้นยังไม่สามารถใช้ได้กับตัวอักษรที่มีสองเส้น เช่น *ส ศ ษ* เป็นต้น (ตัวอักษรในภาษาไทยมีเพียง 1 เส้นกับ 2 เส้นเท่านั้น) จึงจำเป็นที่ต้องหาวิธีการใหม่ที่สามารถใช้ได้กับตัวอักษรที่มีสองเส้นด้วย

วิธีแรกที่เราได้คิดขึ้นมาเพื่อให้สามารถเขียนได้ทั้งตัวอักษรที่มีเส้นเดียวและตัวอักษรที่มีสองเส้นก็คือ ได้ยอมให้มีการคลิกเมาส์ได้เมื่อจะเขียนตัวอักษรที่มีสองเส้น ส่วนการเขียนตัวอักษรที่มีเส้นเดียวได้เลือกใช้วิธีเซท timer เพราะให้ผลที่ดีกว่า คือ เขียนได้เร็วและเป็นธรรมชาติมากกว่า

วิธีนี้มีข้อดี คือ สำหรับตัวอักษรที่มีเส้นเดียวจะสามารถเขียนได้อย่างรวดเร็วและเป็นรู้สึกเป็นธรรมชาติ แต่สำหรับการเขียนตัวอักษรที่มีสองเส้นซึ่งต้องใช้การคลิกเมาส์ช่วย ทำให้รู้สึกไม่สะดวกและยังทำให้เกิดความสับสนด้วยว่า เมื่อไหร่จะต้องคลิกเมาส์เมื่อไหร่ไม่ต้องคลิกเมาส์ และทำให้เขียนผิดได้บ่อยครั้ง รูปที่ 4.7 แสดง statechart ของการเขียนโดยใช้การเซท timer ร่วมกับการใช้คลิกเมาส์

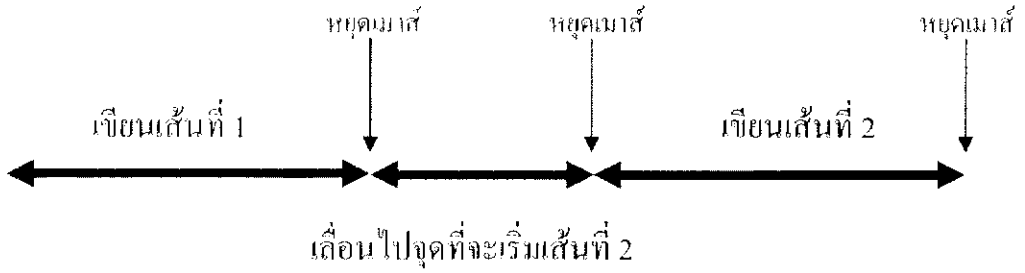
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



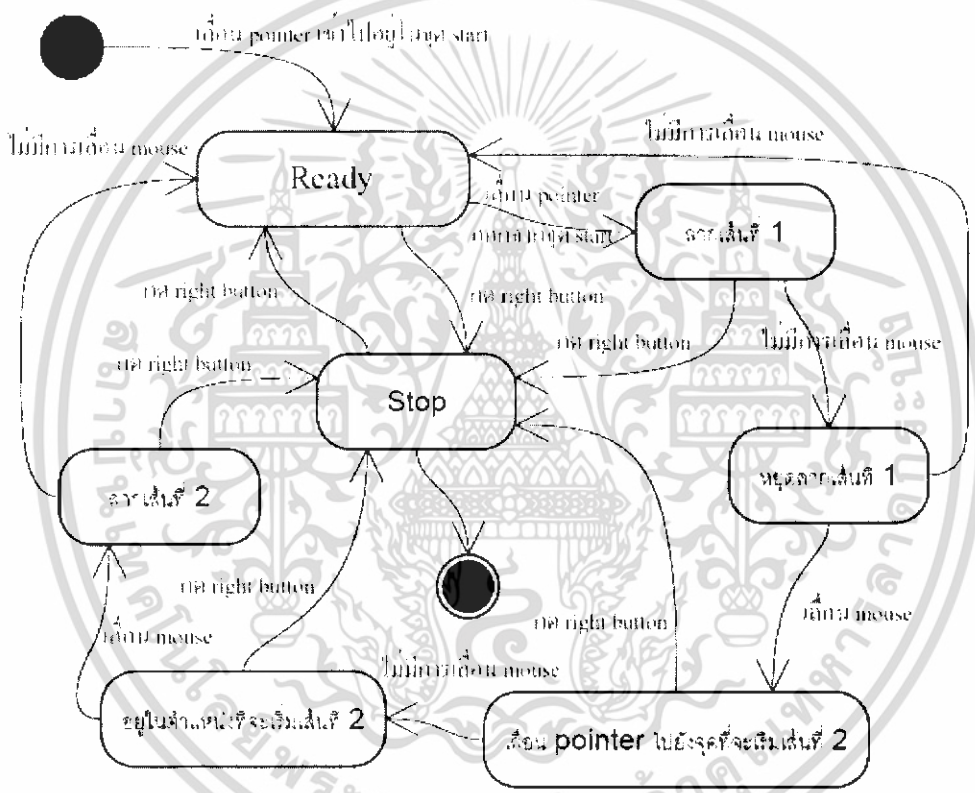
รูปที่ 4.7 แสดง statechart ของการเขียนโดยใช้การเซท timer ร่วมกับการใช้คลิกเมาส์

วิธีที่สองที่ได้คิดขึ้นมาเพื่อให้สามารถเขียนได้ทั้งตัวอักษรที่มีเส้นเดียวและตัวอักษรที่มีสองเส้นก็คือ เปลี่ยนจากการที่จะต้องคลิกเมาส์เมื่อจะเขียนตัวอักษรที่มีสองเส้นมาเป็นการใช้ timer ทั้งหมด โดยการเขียนตัวอักษรหนึ่งตัวจะถูกแบ่งออกเป็น 3 ช่วง คือ ช่วงที่หนึ่งสำหรับการลากเส้นที่ 1 และเมื่อมีการหยุดลากเมาส์ก็จะเข้าสู่ช่วงที่สองซึ่งจะให้เลื่อน mouse pointer ไปยังจุดที่จะเริ่มเส้นที่ 2 ถ้าไม่มีการเลื่อนเมาส์จนครบกำหนดเวลาในช่วงที่สองก็จะถือว่าจบการเขียนตัวอักษรตัวนั้นและ mouse pointer ก็จะกลับไปเริ่มที่จุดเริ่มต้นใหม่ซึ่งกรณีนี้จะเกิดขึ้นกับการเขียนตัวอักษรที่มีเส้นเดียว ดังรูปที่ 4.8 แต่ถ้ามีการเลื่อนเมาส์ในช่วงที่สองก็จะถือเป็นการเลื่อน mouse pointer ไปยังจุดที่จะเริ่มเส้นที่ 2 และเมื่อมีการหยุดเลื่อนเมาส์ก็จะเข้าสู่ช่วงที่สามซึ่งจะให้ลากเส้นที่ 2 เมื่อมีการหยุดลากเมาส์ก็จะถือว่าจบการเขียนตัวอักษรตัวนั้นและ mouse pointer ก็จะกลับไปเริ่มที่จุดเริ่มต้นใหม่ และใช้การเปลี่ยนรูปแบบ mouse pointer เพื่อเป็นการบอกว่าตอนไหนให้เขียนเส้นหรือตอนไหนให้เลื่อนเมาส์ไปอยู่ในตำแหน่งที่จะเขียนเส้นต่อไป รูปที่ 4.9 แสดง statechart ของการเขียนโดยใช้การเซท timer เพียงอย่างเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 แสดงลำดับการเขียน



รูปที่ 4.9 แสดง statechart ของการเขียนโดยใช้การเซท timer เพียงอย่างเดียว

ข้อดีของวิธีที่สองนี้ก็ คือ สามารถเขียนได้อย่างเป็นธรรมชาติโดยไม่ต้องอาศัยการคลิกเมาส์เข้าช่วยและสามารถใช้ได้กับตัวอักษรที่มีในภาษาไทยได้ครบทุกตัว แต่ก็มีข้อเสียคือทำให้การเขียนตัวอักษรที่มีเส้นเดียวช้าไปด้วย แต่ว่าวิธีนี้ไม่ได้เป็นปัญหาสำหรับโปรแกรมของเรา เนื่องจากเราได้กำหนดตัวอักษรที่จะเขียนไว้ก่อนแล้วจึงให้ผู้ใช้เขียนตาม ดังนั้นเราจึงสามารถกำหนดได้ว่าตัวอักษรตัวนั้นควรจะใช้การกำหนดแบบไหน เช่น เรากำหนดให้เขียน ก เรารู้ว่า ก ต้องเขียนครั้งเดียว เราจึงกำหนดให้ใช้อัลกอริทึมสำหรับเขียนเส้นเดียวไปเลย แต่อย่างไรก็ตามสำหรับการนำไปใช้จริงอาจจะยังใช้งานได้ไม่ดีสำหรับตัวอักษรที่ต้องเขียนสองครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การออกแบบระบบ

ระบบเก็บฐานข้อมูลลายมือเขียนผ่านเว็บประกอบไปด้วยส่วนหลักๆ 3 ส่วน คือ

1. ส่วนรับลายมือเขียน (client)
2. server ซึ่งทำหน้าที่ recognize ลายมือเขียนและจับเก็บลายมือเขียนลงฐานข้อมูล
3. เว็บระบบจัดการฐานข้อมูลลายมือเขียน

4.2.1 ส่วนรับลายมือเขียน (client)

เป็นส่วนที่อินเตอร์เฟสสำหรับรับลายมือเขียนแล้วทำการส่งลายมือเขียนนั้นไปให้ server ทำการ recognize แล้วส่งผลกลับคืนมาให้ผู้ใช้ตรวจสอบว่าถูกต้องหรือไม่ ผู้ใช้สามารถเลือกได้ว่าจะบันทึกลายมือลงในระบบหรือไม่ ส่วนรับลายมือเขียนนี้สามารถเรียกใช้โดยผ่านทางหน้าเว็บของระบบจัดเก็บฐานข้อมูลลายมือเขียน

รูปที่ 4.10 แสดง Java Applet สำหรับรับลายมือเขียน

สำหรับการพัฒนาส่วนนี้ได้พยายามหาวิธีที่ทำให้ผู้ใช้สามารถใช้งานได้สะดวก และไม่จำเป็นต้องใช้เพียงปากกาเท่านั้น ผู้ใช้สามารถใช้เมาส์ซึ่งเป็นอุปกรณ์ที่คอมพิวเตอร์ทุกเครื่องมีอยู่แล้วเขียนได้ โดยทั่วๆไปแล้วการใช้เมาส์เขียนจะต้องอาสัยการคลิกเมาส์แล้วลากเพื่อให้ได้ตัวอักษรตามที่ต้องการ ซึ่งกลไกนี้เป็นกลไกเดียวกับการใช้ปากกาเขียนเพราะการคลิกเมาส์แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลากก็เหมือนกับการกดปากกาแล้วลาก (เกิด event เดียวกัน) แต่ว่าการคลิกเมาส์แล้วลากทำให้รู้สึกไม่สะดวกเหมือนกับการเขียนด้วยปากกา จึงทำให้เกิดความต้องการหาวิธีที่ทำให้ไม่ต้องคลิกเมาส์ ดูหัวข้อที่ 4.1

4.2.2 Server

Server มีหน้าที่คอยรับการติดต่อจาก client เพื่อรับลายมือเขียนไปทำการ recognize แล้วส่งผลกลับไปให้ client และมีหน้าที่จัดการเก็บลายมือเขียนที่ส่งมาจาก client เมื่อผู้ใช้ได้ทำการตรวจสอบและเลือกบันทึกลายมือลงในระบบ

การพัฒนาในส่วนของเซิร์ฟเวอร์แอพลิเคชั่น ซึ่งใช้ในการติดต่อระหว่างผู้ใช้ผ่านเวบแอพลิเคชั่นและในการติดต่อกับฐานข้อมูลเก็บลายมือเขียนนั้น พัฒนาโดยใช้เทคโนโลยีดอทเนต ซึ่งช่วยอำนวยความสะดวก ให้สามารถเรียกใช้อัลกอริทึมการรู้จำที่สร้างขึ้นมาจากภาษาต่างๆ ได้อย่างหลากหลายและได้โดยง่าย

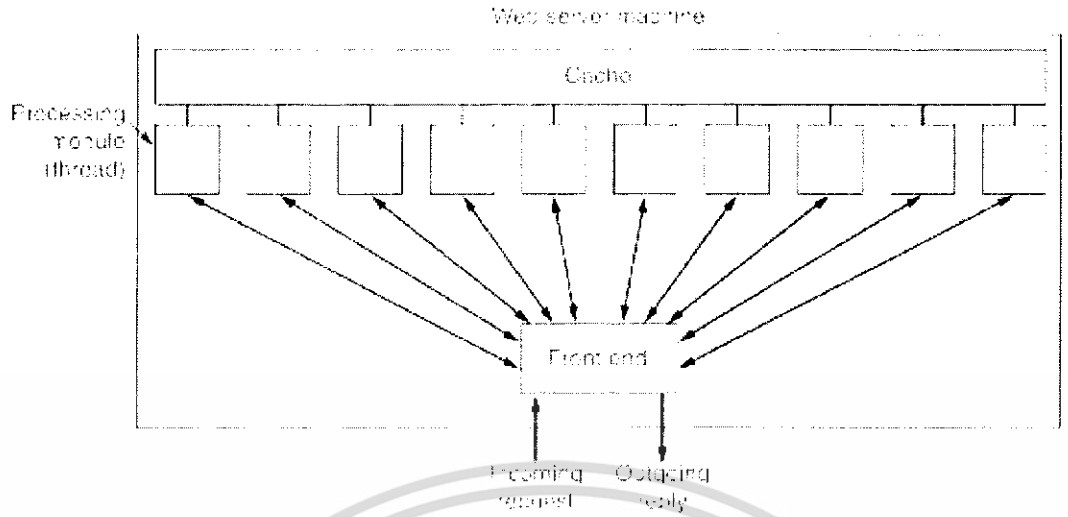
สำหรับการทำงานนั้นมีการใช้เทคโนโลยีของซ็อกเก็ต ซึ่งเป็น TCP Software Interface มาตรฐานที่ใช้ในการติดต่อถึงกันผ่านระบบเครือข่าย โดยใน .Net นั้น การทำงานที่เกี่ยวข้องกับ Sockets จะต้องเรียกใช้งาน Namespace ที่ชื่อว่า System.Net.Sockets ซึ่งเป็น Namespace ที่บรรจุ Classes ต่างๆ ที่สนับสนุนการทำงานกับ Sockets โดย Class ที่เกี่ยวข้องกับการทำงานหลักๆ กับ Sockets เช่น NetworkStream, TcpClient, TcpListener, UdpClient, Socket เป็นต้น โดยคลาส Socket จะมีฟังก์ชั่นการทำงานพื้นบานต่างๆ ที่จำเป็นสำหรับการสร้างแอพลิเคชั่นที่ใช้งาน Socket สำหรับ Properties ที่สำคัญสำหรับ System.Net.Sockets.Socket เช่น

- AddressFamily ใช้ในการรับค่า address family ของ Socket
- LocalEndPoint ใช้ในการรับค่า end point ซึ่งเป็น network address ของ local
- ProtocolType ใช้ในการรับค่าชนิดของโปรโตคอลใน Socket
- SocketType ใช้ในการรับค่าชนิดของ Socket

นอกจากนี้ยังมี Method ที่สำคัญที่ใช้สำหรับการสร้างเซิร์ฟเวอร์แอพลิเคชั่นอีก เช่น

- Accept ใช้สร้าง Socket ใหม่สำหรับการเชื่อมต่อที่เกิดขึ้น
- Bind ใช้ในการเชื่อมต่อ Socket เข้ากับ local endpoint
- Listen ใช้ในการให้ Socket รอรับการติดต่อจากไคลเอนท์
- Send ใช้ในการส่งข้อมูลให้กับ Socket ที่เชื่อมต่อกันอยู่
- Receive ใช้ในการรับข้อมูลจาก Socket ที่เชื่อมต่อกันอยู่
- Close ใช้ในการปิดการเชื่อมต่อกับ Socket และเคลียร์ทรัพยากรของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 แสดงการทำงานแบบมัลติเทรดในเซิร์ฟเวอร์

หลักการสำคัญอีกอย่างหนึ่งที่ใช้ในการพัฒนาเซิร์ฟเวอร์แอพลิเคชั่นนี้คือ การทำงานแบบเป็นมัลติเทรด ซึ่งช่วยให้รองรับการใช้งานได้พร้อมกันโดยไม่มีปัญหา ซึ่งการสร้าง Thread สำหรับ .Net นั้น สามารถทำได้โดยเรียกใช้ Namespace ที่ชื่อ System.Threading โดยที่คลาส Thread นั้นจะเป็นตัวชี้ไปยังส่วนของโปรแกรมซึ่งจะทำงานบน Thread ใหม่ที่ถูกสร้างขึ้น สำหรับ method ที่สำคัญใน Thread Class ได้แก่ Start ใช้สำหรับให้ OS ทำการเปลี่ยนสถานะของ Thread ให้ทำงาน Abort ใช้ในการจบการทำงานของ Thread เป็นต้น นอกจาก Thread แล้ว ใน .Net ยังมี Namespace อีกตัวชื่อ ThreadPool ซึ่งเป็น Static Class โดยการทำงานนั้นจะเริ่มทำงานทันทีเมื่อเรียกใช้ ThreadPool โดยไม่ต้องสั่งให้ Start

นอกจากนี้แล้ว เซิร์ฟเวอร์แอพลิเคชั่นนี้ สามารถทำการเพิ่มอัลกอริทึมในกระบวนการรู้จำเข้าไปได้ โดยการสร้างคลาสแบบ Managed ในการห่อหุ้ม Dynamic Link Library (DLL) ของการรู้จำ เพื่อเรียกใช้อัลกอริทึมในการรู้จำผ่าน .NET Interoperation Services ซึ่งช่วยให้การทำงานระหว่างภาษาต่างชนิดกัน โดยช่วยให้ .NET สามารถทำงานร่วมกันโค้ดอื่นๆ ที่อยู่นอกจาก .NET Environment ทำได้โดยง่าย เช่น C# สามารถเรียกใช้คลาสจาก Unmanaged C++ ได้โดยตรง สำหรับขั้นตอนการเพิ่มอัลกอริทึมในการรู้จำ มีขั้นตอนหลัก 3 ส่วน ดังต่อไปนี้

ส่วนแรก ในส่วนของการรู้จำ ทำการสร้างไฟล์ .Dll ขึ้นมา จากโค้ดการรู้จำตัวอักษรอัลกอริทึมแบบต่างๆ ที่มีอยู่

ส่วนที่สอง ในส่วนของคลาสที่ใช้ห่อหุ้มไลบรารี่ที่ได้สร้างไว้ (Wrapper Class) เนื่องจากไฟล์ .Dll ที่ได้จาก .NET มีสถาปัตยกรรมแตกต่างจากอย่างอื่น โดย .NET จะสร้างไฟล์คอมโพเน้นต์ในรูปแบบของตัวเอง เรียกกันตรงๆ ว่า .NET component แม้ว่าผลลัพธ์คือไฟล์นามสกุล .DLL เหมือนกัน แต่ภายในมีโครงสร้างที่ต่างกันอย่างสิ้นเชิง การทำให้สามารถเรียกใช้โค้ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

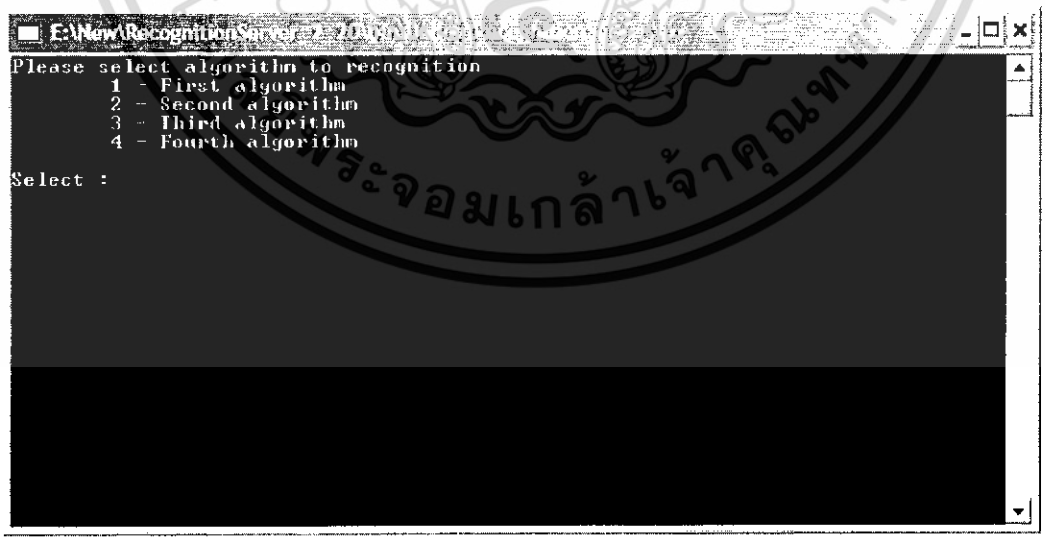
Unmanaged Dll จาก .NET component เรียกว่าการทำ Interop โดย Wrapper Class ที่สร้างขึ้นมานี้จะทำให้ .NET มองเห็นไฟล์ .Dll เสมือนเป็น .NET Class ทั่วๆ ไป

ส่วนที่สาม ในส่วนโปรแกรมหลักของเซิร์ฟเวอร์ ซึ่งในที่นี้เขียนโดย C# ซึ่งเป็น Managed Extension (การเขียนโปรแกรมใน Managed Environment ซึ่งเป็นส่วนหนึ่งของ .Net Framework) สามารถเรียกใช้งานไลบรารี่ของการรู้จำโดยโดยตรง ผ่าน Wrapper Class ที่สร้างขึ้น

4.2.2.1 การทำงานของเซิร์ฟเวอร์แอพลิเคชั่น

เซิร์ฟเวอร์แอพลิเคชั่นทำงานในโหมดคอนโซล โดยเมื่อเริ่มต้นเปิดใช้งาน ผู้ใช้จะพบกับหน้าจอให้เลือกอัลกอริทึมที่จะใช้ในการรู้จำ ซึ่งในขั้นนี้มีเพียงอัลกอริทึมเดียว (โดยได้อธิบายวิธีการในการเพิ่มอัลกอริทึมให้ผู้ที่ต้องการใช้ในหัวข้อก่อนหน้าแล้ว) หลังจากเลือกอัลกอริทึมเรียบร้อยแล้ว จะเริ่มเข้าสู่กระบวนการเทรนข้อมูล เพื่อเตรียมไว้ใช้ในกระบวนการรู้จำตัวอักษร ต่อมาเซิร์ฟเวอร์จะทำการเปิดการเชื่อมต่อและรอรับการเชื่อมต่อจากไคลเอนท์ โดยเป็นแบบมัลติเทรด เพื่อให้สามารถรองรับการทำงานของผู้ใช้ได้หลายคนพร้อมกัน ซึ่งตัวไคลเอนท์นี้ เป็นเวบแอพลิเคชั่น ที่ทำหน้าที่รับและจัดการข้อมูลลายมือเขียนจากผู้ใช้ ผ่านทางเวบบราวส์เซอร์ และจัดส่งให้กับเซิร์ฟเวอร์

หลังจากเซิร์ฟเวอร์ได้รับข้อมูลลายมือเขียนกลับมาแล้ว ก็จะทำการส่งข้อมูลนั้นเข้าสู่กระบวนการรู้จำ และส่งค่าตัวอักษรที่ได้จากการรู้จำนั้นกลับคืน เพื่อแสดงผลการทำงานสู่ผู้ใช้ โดยที่ผู้ใช้สามารถจัดการกับข้อมูลลายมือเขียนของตนเองได้ และยังสามารถจัดเก็บลายมือเขียนที่สามารถรู้จำได้อย่างถูกต้องนั้น เพื่อเก็บไว้ใช้ประโยชน์ต่อไป



รูปที่ 4.12 แสดงเซิร์ฟเวอร์ขณะรอให้ผู้ใช้ป้อนอัลกอริทึมที่ต้องการใช้ในกระบวนการรู้จำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้เลือกอัลกอริทึมเรียบร้อยแล้ว เซิร์ฟเวอร์จะทำการเทรนข้อมูลเพื่อเตรียมไว้ใช้ในกระบวนการรู้จำ หลังจากนั้นจึงเปิดรอรับการเชื่อมต่อจากไคลเอนท์แบบมัลติเทรด ดังรูปที่ 4.13

รูปที่ 4.13 แสดงการรอรับการเชื่อมต่อจากไคลเอนท์ของเซิร์ฟเวอร์

ที่โปรแกรมไคลเอนท์ เมื่อผู้ใช้ป้อนข้อมูลลายมือเขียนเข้ามา โดยการลากเมาส์เป็นตัวอักษร ไคลเอนท์จะติดต่อเพื่อขอรับบริการจากเซิร์ฟเวอร์ โดยจะส่งข้อมูลลายมือเขียนมาให้เซิร์ฟเวอร์นำเข้าสู่กระบวนการรู้จำต่อไป

รูปที่ 4.14 แสดงการเขียนลายมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้น เมื่อเซิร์ฟเวอร์ได้รับข้อมูล และตรวจสอบเรียบร้อยแล้ว เซิร์ฟเวอร์จะส่งผลลัพธ์ที่ได้จากกระบวนการรู้จำไปให้ไคลเอนท์ และปิดการเชื่อมต่อ ดังรูปที่ 4.15

รูปที่ 4.15 แสดงการปิดการเชื่อมต่อของเซิร์ฟเวอร์

ที่ไคลเอนท์ จะแสดงผลลัพธ์ที่ได้รับจากเซิร์ฟเวอร์ เพื่อให้ผู้ใช้ตรวจสอบว่าได้ผลถูกต้องหรือไม่ ดังรูปที่ 4.16

รูปที่ 4.16 แสดงผลการรู้จำลายมือเขียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้ต้องการจัดเก็บข้อมูลลายมือเขียนของตนไว้ จะทำการเลือกข้อมูลนั้นๆ และคลิกที่บันทึกลายมือ โดยมีการตรวจสอบความถูกต้องของข้อมูลที่ได้รับมาก่อนด้วย จากนั้นจึงจะส่งข้อมูลเหล่านี้ไปให้เซิร์ฟเวอร์

เมื่อเซิร์ฟเวอร์ได้รับข้อมูลที่ต้องการให้มีการจัดเก็บ จะทำการจัดเก็บข้อมูลลายมือเขียนเหล่านั้นเข้าสู่ฐานข้อมูล โดยจะแสดงจำนวนข้อมูลที่ผ่านการตรวจสอบจากผู้ใช้แล้วว่ามีความถูกต้องและผิดพลาดเป็นจำนวนเท่าไร และเก็บข้อมูลเหล่านี้ไว้เป็นสถิติเพื่อใช้งานต่อไปได้

4.3 เว็บระบบจัดเก็บฐานข้อมูลลายมือเขียน

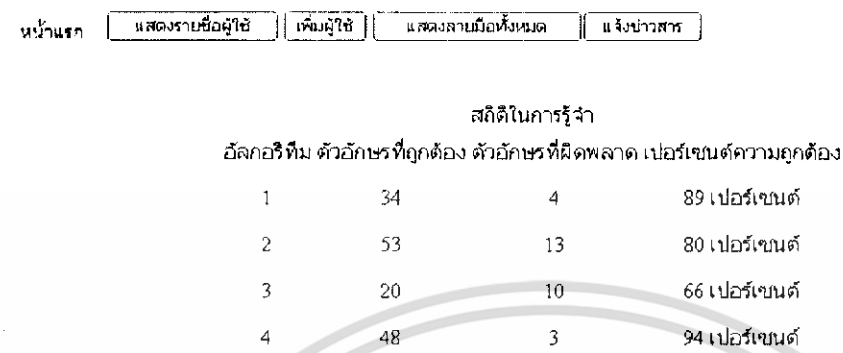
เป็นเว็บสำหรับให้ผู้ดูแลระบบและผู้ใช้เข้ามาใช้ระบบจัดเก็บฐานข้อมูลลายมือเขียนได้ โดยมีคุณสมบัติของหน้าเว็บสำหรับผู้ดูแลระบบและผู้ใช้แตกต่างกัน และเมื่อจะเข้าไปใช้งานในระบบผู้ใช้จะต้องทำการป้อนชื่อผู้ใช้และรหัสผ่านก่อน ดังรูปที่ 4.17 แต่ถ้าผู้ใช้ลืมรหัสผ่านก็ให้เลือกลิงค์สำหรับผู้ที่ลืมรหัสผ่าน ซึ่งในหน้านี้จะมีช่องให้ใส่ชื่อผู้ใช้แล้วระบบจะทำการส่งรหัสผ่านไปให้ทางอีเมล์แอดเดรสของผู้ใช้

รูปที่ 4.17 แสดงหน้าล็อกอิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

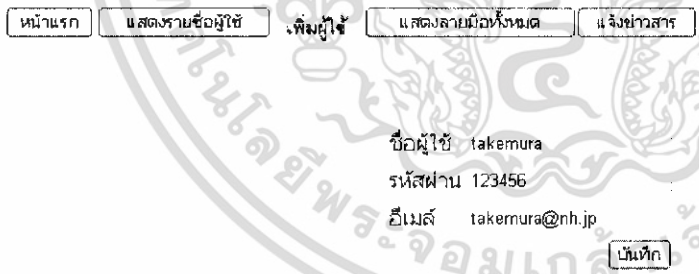
4.3.1 สำหรับผู้ดูแลระบบ

เมื่อเข้ามาสู่หน้าแรกจะมีตารางแสดงจำนวนลายมือที่ได้ทำการทดสอบการรู้จำแยกตามอัลกอริทึมที่ใช้ในการทดสอบ ดังรูปที่ 4.18



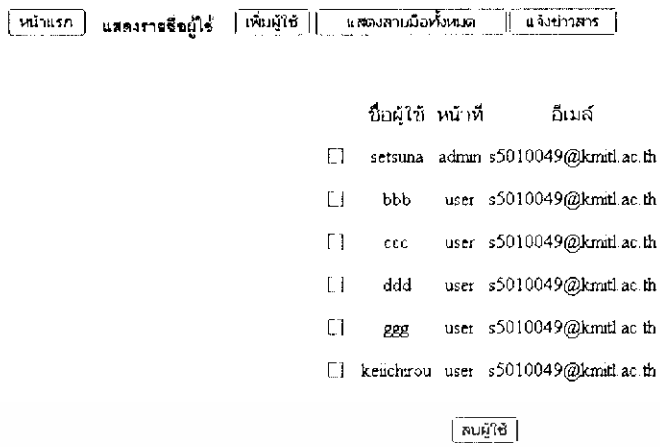
รูปที่ 4.18 แสดงสถิตการรู้จำลายมือเขียน

มีส่วนสำหรับให้ผู้ดูแลระบบใช้ในการเพิ่ม- ถอนรายชื่อผู้ใช้หรือแก้ไขข้อมูลของผู้ใช้ เนื่องจากผู้ดูแลระบบเป็นผู้ที่มีหน้าที่ในการเพิ่ม- ถอนรายชื่อผู้ใช้เข้าไปในระบบ ผู้ใช้คือคนที่ถูกเลือกให้เข้ามาใช้ระบบได้เท่านั้น ไม่มีหน้าเว็บสำหรับให้ผู้ใช้สมัครเข้าเป็นสมาชิก ดังรูปที่ 4.19และ 4.20



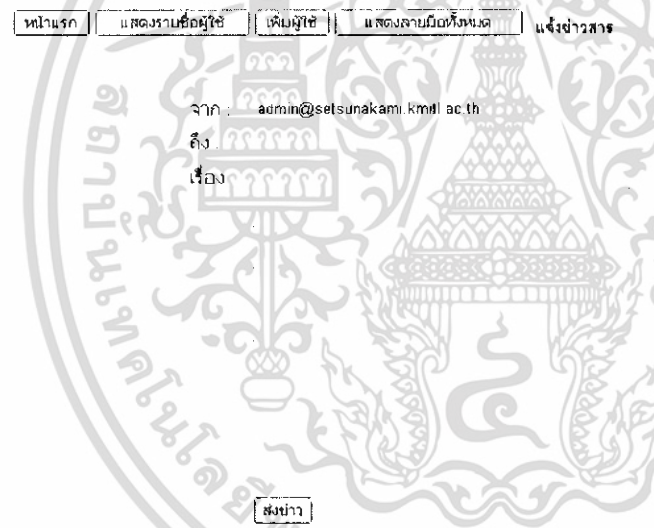
รูปที่ 4.19 แสดงหน้าเพิ่มผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.20 แสดงหน้าแสดงรายชื่อผู้ใช้

มีหน้าเว็บสำหรับให้ผู้ดูแลระบบสามารถส่งอีเมล์ไปยังผู้ใช้ได้ เมื่อมีข่าวสารที่ต้องต้องการแจ้งให้ผู้ใช้ทราบ



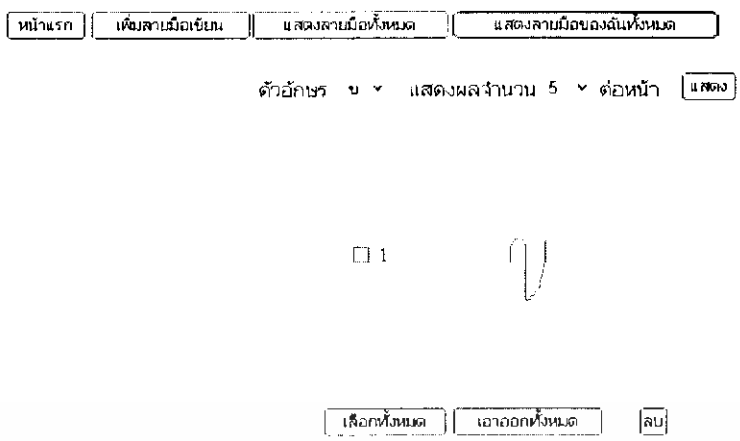
รูปที่ 4.21 แสดงหน้าสำหรับส่งอีเมล์เพื่อแจ้งข่าวสาร

ผู้ดูแลระบบสามารถเข้าไปดูลายมือเขียนของผู้ใช้ได้ และสามารถลบข้อมูลลายมือเขียนของผู้ใช้ได้ (ในกรณีที่มีข้อมูลลายมือเขียนที่ไม่ถูกต้อง)

4.3.2 สำหรับผู้ใช้

มีส่วนสำหรับผู้ใช้จมีส่วนที่ใช้สำหรับเขียนลายมือและส่วนสำหรับเข้าไปดูลายมือเขียนของตนเองและสามารถลบข้อมูลลายมือเขียนของตนเองได้ (ในกรณีที่มีข้อมูลลายมือเขียนที่ไม่ถูกต้อง) ดังรูปที่ 4.22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 แสดงหน้าตรวจสอบลายมือเขียน

สามารถดาวน์โหลดข้อมูลลายมือเขียนทั้งของตนเองและทั้งฐานข้อมูลไปใช้ได้ โดยข้อมูลจะถูกดึงมาจากฐานข้อมูลแล้วบันทึกเป็นเท็กไฟล์ ดังรูปที่ 4.23

รูปที่ 4.23 แสดงไฟล์ลายมือเขียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุป

จากการที่เราต้องการหาวิธีรับลายมือเขียนโดยผ่านทางการลากเมาส์โดยที่ไม่ต้องคลิกเมาส์ เนื่องจากการคาดหวังว่าเมาส์ซึ่งเป็นอุปกรณ์ที่เครื่องคอมพิวเตอร์ทุกเครื่องในปัจจุบันจำเป็นต้องมี จะสามารถนำมาใช้เขียนตัวอักษรแทนการใช้ปากกาได้ ซึ่งวิธีการเดิมที่ใช้กันมาจะใช้การแดร็กเมาส์ (การกดเมาส์แล้วลาก) แต่การที่ต้องกดปุ่มของเมาส์ขณะลากไปด้วยทำให้เขียนลำบาก ขาดความคล่องตัว ทำให้เราคิดว่าน่าจะลองหาวิธีที่ไม่ต้องคลิกเมาส์ดู และในที่สุดเราได้เลือกใช้วิธีที่ใช้การหยุดเลื่อนเมาส์เป็นตัวแบ่งระหว่างการเขียนเส้นที่หนึ่งและเส้นที่สอง กับใช้เป็นตัวแบ่งระหว่างแต่ละตัวอักษร ซึ่งจากการทดสอบใช้งานจริงสามารถใช้ได้ดีสำหรับตัวอักษรที่เขียนทีเดียว (ทดสอบโดยใช้ออปติคอลเมาส์) แต่ไม่เหมาะสำหรับตัวอักษรที่ต้องเขียนหลายที (ตัวอักษรที่ถ้าใช้ปากกาเขียนต้องมีการยกปากกา)

เนื่องจากว่าอัลกอริทึมที่เรานำเอามาใช้ในการทดสอบการรู้จำลายมือเขียนนี้ สามารถรู้จำลายมือเขียนภาษาไทยได้เฉพาะตัวอักษรที่ต้องเขียนครั้งเดียวเท่านั้น เช่น ก ข ค แต่ไม่สามารถรู้จำลายมือเขียนที่ต้องเขียนหลายครั้งได้ เช่น ส ษ เป็นต้น ดังนั้นเราจึงเลือกเก็บเฉพาะลายมือเขียนของตัวอักษรที่เขียนครั้งเดียวเท่านั้นลงในฐานข้อมูล

ระบบที่พัฒนาขึ้นมานี้คาดว่าจะสามารถนำไปใช้ได้จริงและสามารถอำนวยความสะดวกในการทดสอบอัลกอริทึมในการรู้จำลายมือเขียนได้ ช่วยลดระยะเวลาของการทดสอบได้และลดความยุ่งยากจากเดิมที่ต้องให้ผู้ที่เป็นอาสาสมัครมาเขียนทีละคน แต่ถ้าใช้ระบบที่เราได้พัฒนาขึ้นซึ่งสามารถใช้งานได้ผ่านทางเว็บมาแทนก็คงจะทำให้ลดเวลาของการพัฒนาลงไปได้บ้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] James A. O'Brien, 2000, "Management Information Systems", McGraw-Hill

[2] สมยศ จินดาผล "การรู้จำตัวพิมพ์อักขระไทยหลายรูปแบบจากประโยค" วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต สาขาวิชาคอมพิวเตอร์ ปีการศึกษา 2540 บัณฑิตวิทยาลัย สถาบันบัณฑิตพัฒนบริหารศาสตร์

[3] กานต์ เสารยะวิเศษ "การรู้จำลายมือเขียนตัวเลขไทยโดยใช้การจัดกลุ่มแบบฟัซซี่" วิทยานิพนธ์คุรุศาสตร์อุตสาหกรรมมหาบัณฑิต สาขาวิชาเทคโนโลยีคอมพิวเตอร์ ปีการศึกษา 2545 บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ

[4] Vuokko Vuori, "Adaptation in On-line Recognition of Handwriting" , 1999 , Department of Engineering Physics and Mathematics, Helsinki University of Technology

[5] O'Reilly & Associates., 2001, "Java Foundation Classes in a Nutshell", [Online] URL : http://www.unix.org.ua/orelly/java-ent/jfc/ch07_03.htm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้