

แอปพลิเคชันผู้ใช้สำหรับระบบเอกสารภาพแบบ DjVu



นางสาวสุชีราภรณ์ สุรัชกุลวัฒนา

สถาบันวิทยบริการ

จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า


คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2548

ISBN 974-17-5833-2

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

CLIENT APPLLET FOR DJVU DOCUMENT IMAGING SYSTEM



Miss Sucheraporn Surachaikulwattana

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering Program in Electrical Engineering

Department of Electrical Engineering

Faculty of Engineering

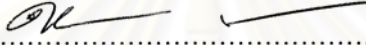
Chulalongkorn University

Academic Year 2005

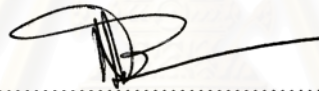
ISBN 974-17-5833-2

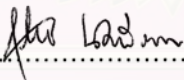
หัวข้อวิทยานิพนธ์ แอปพลิเคชันผู้ใช้สำหรับระบบเอกสารภาพแบบ DjVu
โดย นางสาวสุชีราภรณ์ สุรัชย์กุลวัฒนา
สาขาวิชา วิศวกรรมไฟฟ้า
อาจารย์ที่ปรึกษา อาจารย์สุวิทย์ นาคพีระยุทธ

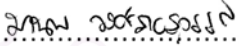
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้หัวข้อวิทยานิพนธ์ฉบับนี้เป็นส่วน
หนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต


..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร. ดิเรก ลาวณย์ศิริ)

คณะกรรมการสอบวิทยานิพนธ์


..... ประธานกรรมการ
(รองศาสตราจารย์ ดร.สมชาย จิตะพันธ์กุล)


..... อาจารย์ที่ปรึกษา
(อาจารย์สุวิทย์ นาคพีระยุทธ)


..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.มานพ วงศ์สายสุวรรณ)

สภามหาวิทยาลัย
จุฬาลงกรณ์มหาวิทยาลัย

สุชีราภรณ์ สุรัชกุลวัฒนา : แอปพลิเคชันด้านผู้ใช้สำหรับระบบเอกสารภาพแบบ DjVu
(CLIENT APPLLET FOR DJVU DOCUMENT IMAGING SYSTEM) อ. ที่ปรึกษา :
อ.สุวิทย์ นาคพีระยุทธ, 53 หน้า. ISBN 974-17-5833-2.

งานวิจัยนี้ได้พัฒนาโปรแกรมสำหรับระบบเรียกคืนภาพเอกสารสแกนผ่านเว็บ โดยได้ทำการพัฒนาโปรแกรมด้วยภาษาจาวา ที่มีชื่อว่า DjVuApplet สำหรับการเรียกดูภาพเอกสาร DjVu ซึ่งเป็นรูปแบบการบีบอัดที่เหมาะสมกับเอกสารภาพสี จากการสแกนในรูปแบบ BMP หรือ JPG ซึ่งมีทั้งตัวอักษรและรูปภาพ

ระบบการเรียกดูเอกสารผ่านเว็บ เอกสารภาพทั้งเล่มจะถูกเก็บไว้ที่เครื่องคอมพิวเตอร์ที่เป็นเว็บเซิร์ฟเวอร์ ผู้ใช้สามารถที่จะเรียกดูภาพเอกสาร DjVu ผ่านโปรแกรมบราวส์เซอร์ที่เครื่องผู้ใช้ได้เลย โดยที่ไม่ต้องเสียเวลาในการดาวน์โหลดโปรแกรมปลั๊กอินมาติดตั้งเพิ่มเติม ซึ่งก็เป็นวิธีหนึ่งที่จะช่วยป้องกันความเสียหายที่จะเกิดขึ้นกับระบบของไคลแอนท์ได้และยังทำให้เกิดความสะดวกกับผู้ใช้งานอีกด้วย โดยที่ภาพเอกสาร DjVu จะแสดงอยู่ภายใต้การทำงานของจาวา แอปพลิเคชันซึ่งจะทำงานผ่าน JVM ของ บราวส์เซอร์ที่เครื่องผู้ใช้

โปรแกรม DjVuApplet มีขนาดของไฟล์เพียง 193 กิโลไบต์เท่านั้น ในขณะที่ปลั๊กอินของ DjVu (เวอร์ชัน 3.5) มีขนาดของไฟล์ประมาณ 2 เมกกะไบต์ และผลลัพธ์ที่ได้จากการทดสอบเวลาที่ใช้ในการเรียกดูหน้าเอกสารและการทำงานในส่วนติดต่อกับผู้ใช้ก็อยู่ในเกณฑ์ที่น่าพอใจ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา.....วิศวกรรมไฟฟ้า.....ลายมือชื่อนิสิต.....
สาขาวิชา.....วิศวกรรมไฟฟ้า.....ลายมือชื่ออาจารย์ที่ปรึกษา.....
ปีการศึกษา.....2548.....

4470722421 : MAJOR ELECTRICAL ENGINEERING

KEY WORD: DJVU / DOCUMENT IMAGING SYSTEM / APPLET

SUCHERAPORN SURACHAIKULWATTANA : CLIENT APPLET FOR DJVU
DOCUMENT IMAGING SYSTEM. THESIS ADVISOR : SUVIT NAKPEERAYUTH, 53
pp. ISBN 974-17-5833-2

This research modified program for web based scanning document image retrieval system. A java applet program, DjVuApplet enable user to view DjVu files in their browser. DjVu is a good compression for scanned color image with texts and pictures inside.

In the document retrieval system, the whole document volume will be uploaded to a web server. With the program DjVuApplet, Internet users will be able to view DjVu files in their browser without a plug-in downloaded. Therefore, the program DjVuApplet can protect the client system from the outside damage and also provide the convenience for the user. When user use a browser to view a djvu page that contains an applet, the applet's code is transferred to client's system and executed by the browser's Java Virtual Machine (JVM).

DjVuApplet program file size is only 193 KB, whereas DjVu plug-in (version 3.5) have the file size about 2 MB. And the response time of the system for browsing a djvu page and user interaction is satisfactory.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

Department..... Electrical Engineering..... Student's Signature.....
Field of study..... Electrical Engineering..... Advisor's signature.....
Academic year..... 2005.....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้เป็นอย่างดีด้วยความช่วยเหลืออย่างดียิ่งของ อ.สุวิทย์ นาคพีระยุทธ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งกรุณาให้ความรู้ คำแนะนำที่เป็นประโยชน์ ในการวิจัย ตลอดจนความเมตตา และเอาใจใส่ต่อผู้ทำวิจัยมาโดยตลอด ผู้วิจัยจึงขอกราบ ขอบพระคุณมา ณ ที่นี้

ขอกราบขอบพระคุณบิดามารดาที่ให้กำเนิด ให้ความรัก ความอบอุ่น การอบรม สั่งสอน การสนับสนุน และเป็นกำลังใจแก่ผู้วิจัยในทุก ๆ ด้านเสมอมา

สุดท้ายนี้ ขอขอบคุณห้องปฏิบัติการวิจัยโทรคมนาคมซึ่งเป็นสถานที่ทำวิจัย รวมถึงเพื่อนพี่น้องนิสิตทุกท่าน ที่มีส่วนช่วยเหลือในการให้ข้อคิดเห็น คำแนะนำ และกำลังใจ จนกระทั่งวิทยานิพนธ์ฉบับนี้ลุล่วงไปได้ด้วยดี



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

| | หน้า |
|---|------|
| บทคัดย่อภาษาไทย..... | ง |
| บทคัดย่อภาษาอังกฤษ..... | จ |
| กิตติกรรมประกาศ..... | ฉ |
| สารบัญ..... | ช |
| สารบัญตาราง..... | ฉ |
| สารบัญภาพ..... | ญ |
| บทที่ | |
| 1 บทนำ..... | 1 |
| 1.1 ความเป็นมาและความสำคัญของปัญหา..... | 1 |
| 1.2 วัตถุประสงค์..... | 3 |
| 1.3 ขอบเขตของวิทยานิพนธ์..... | 3 |
| 1.4 ขั้นตอนและวิธีในการดำเนินงาน..... | 3 |
| 1.5 ประโยชน์ที่คาดว่าจะได้รับ..... | 4 |
| 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง..... | 5 |
| 2.1 ระเบียบวิธีการบีบอัดเอกสารภาพ DjVu..... | 5 |
| 2.1.1 หลักการของระเบียบวิธีการบีบอัดเอกสารภาพ DjVu..... | 5 |
| 2.1.2 เทคนิคที่ใช้ในการแยกฟอร์กราวนด์และแบ็คกราวนด์..... | 7 |
| 2.1.3 หลักการบีบอัดภาพชั้นมาสค์ด้วยวิธี JB2..... | 8 |
| 2.1.4 Model Base Coding..... | 8 |
| 2.1.4.1 Pattern matching and substitution (PM&S)..... | 9 |
| 2.1.4.2 Soft pattern matching (SPM)..... | 9 |
| 2.1.4.2.1 แบบไม่มีการสูญเสียข้อมูล (lossless)..... | 11 |
| 2.1.4.2.2 แบบมีการสูญเสียข้อมูล (lossy)..... | 11 |
| 2.1.5 หลักการบีบอัดภาพชั้นแบ็คกราวนด์ด้วยวิธี IW44..... | 12 |
| 2.2 รูปแบบเพิ่มข้อมูล DjVu..... | 13 |
| 2.3 เปรียบเทียบ DjVu กับรูปแบบการเก็บเพิ่มเอกสารภาพวิธีอื่นๆ..... | 14 |
| 2.4 ระบบภาพเอกสาร..... | 15 |
| 2.4.1 ข้อดีของระบบการจัดเก็บภาพเอกสาร..... | 16 |

| บทที่ | ช หน้า |
|---|-----------|
| 2.5 งานวิจัยที่เกี่ยวข้อง..... | 17 |
| 3 การออกแบบโปรแกรมส่วนเรียกคืนภาพเอกสาร DjVu ผ่านเว็บ..... | 21 |
| 3.1 ระบบจัดเก็บและเรียกคืนภาพเอกสารผ่านอินเทอร์เน็ต..... | 21 |
| 3.1.1 ส่วนจัดเก็บภาพเอกสาร..... | 22 |
| 3.1.2 การออกแบบส่วนแสดงภาพเอกสาร..... | 22 |
| 3.2 เหตุผลที่เลือกการพัฒนาส่วนเรียกคืนภาพเอกสาร DjVu ด้วยภาษาจาวา..... | 23 |
| 3.3 แอปเพล็ต (Applet)..... | 24 |
| 3.3.1 คลาสแอปเพล็ต..... | 24 |
| 3.3.2 ช่วงชีวิตของแอปเพล็ต..... | 25 |
| 3.3.3 APPLET TAG..... | 27 |
| 3.4 การออกแบบโปรแกรม DjVuApplet..... | 28 |
| 3.5 คุณสมบัติของโปรแกรม DjVuApplet..... | 30 |
| 3.6 การค้นหาข้อความในเอกสารภาพ DjVu..... | 32 |
| 4 ผลการดำเนินการ..... | 33 |
| 4.1 การแปลงภาพเอกสาร จากแฟ้มภาพสแกนให้เป็น DjVu | 33 |
| 4.2 การเรียกใช้งานไลบรารีของโปรแกรม DjVuApplet..... | 35 |
| 4.3 ส่วนเรียกคืนภาพเอกสาร..... | 37 |
| 4.3.1 ฟังก์ชันการทำงานในส่วนติดต่อกับผู้ใช้ (UI)..... | 38 |
| 4.3.2 การควบคุมการทำงานของ Applet ผ่านทาง APPLET tag ของเอกสาร HTML..... | 45 |
| 4.4 ตัวอย่างการแสดงผลเอกสารภาพดิจิทัล DjVu ในรูปแบบลายมือเขียน..... | 46 |
| 4.5 ตัวอย่างการสร้างเอกสารภาพ DjVu..... | 47 |
| 5 สรุปและข้อเสนอแนะ..... | 50 |
| 5.1 สรุปผลการวิจัย..... | 50 |
| 5.2 ข้อเสนอแนะ..... | 51 |
| รายการอ้างอิง..... | 52 |
| ประวัติผู้เขียนวิทยานิพนธ์..... | 53 |

สารบัญตาราง

| ตาราง | หน้า |
|--|------|
| 3.1 คุณสมบัติของโปรแกรม DjVuApplet..... | 30 |
| 4.1 อารีกิวเมนต์ที่จำเป็นต้องมีใน APPLETTAG..... | 36 |
| 4.2 ฟังก์ชันการทำงานของทูลบาร์..... | 39 |
| 4.3 เวลาที่ใช้ในการเรียกดูหน้าเอกสาร DjVu..... | 49 |



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญภาพ

| ภาพประกอบ | หน้า |
|--|------|
| รูปที่ 2.1 หลักการบีบอัดภาพเอกสารแบบ DjVu..... | 6 |
| รูปที่ 2.2 แผนภาพวิธีการทำ Pattern matching and substitution..... | 10 |
| รูปที่ 2.3 แผนภาพวิธีการทำ Soft pattern matching..... | 12 |
| รูปที่ 2.4 ภาพแสดงโครงสร้างแฟ้มชนิด EA IFF 85 รูปแบบที่ใช้เก็บเอกสาร DjVu..... | 14 |
| รูปที่ 2.5 แสดงการเปรียบเทียบผลลัพธ์การบีบอัดเอกสารภาพสแกนสีระหว่าง JPEG และ DjVu..... | 15 |
| รูปที่ 2.6 ระบบจัดเก็บและเรียกคืนภาพเอกสารผ่านอินเทอร์เน็ตในงานวิจัยของภาณุมาศ..... | 18 |
| รูปที่ 2.7 ระบบจัดเก็บและเรียกคืนภาพเอกสารผ่านอินเทอร์เน็ตในงานวิจัยของสิทธิชัย..... | 19 |
| รูปที่ 3.1 ระบบจัดเก็บและเรียกคืนภาพเอกสารผ่านอินเทอร์เน็ต..... | 21 |
| รูปที่ 3.2 Applet Class..... | 25 |
| รูปที่ 3.3 Applet Life Cycle..... | 25 |
| รูปที่ 4.1 การบันทึกเอกสารภาพให้อยู่ในรูปแบบ DjVu..... | 34 |
| รูปที่ 4.2 การสร้างความเชื่อมโยงระหว่างแฟ้มเอกสารภาพที่มาจาก สิ่งพิมพ์เดียวกัน..... | 34 |
| รูปที่ 4.3 การเรียกดูเอกสารภาพผ่านเว็บ..... | 38 |
| รูปที่ 4.4 เมนูทูลบาร์ของโปรแกรม DjVuApplet..... | 38 |
| รูปที่ 4.5 การเรียกดูแฟ้มเอกสารภาพ DjVu ที่ระดับความละเอียดเท่ากับ100%..... | 39 |
| รูปที่ 4.6 การเรียกดูแฟ้มเอกสารภาพ DjVu ที่ระดับความละเอียดเท่ากับ150%..... | 40 |
| รูปที่ 4.7 การเรียกดูแฟ้มเอกสารภาพ DjVu ที่ระดับความละเอียดเท่ากับ300%..... | 40 |
| รูปที่ 4.8 การเรียกดูหน้าถัดไป หน้าก่อนหน้าของเอกสาร..... | 41 |
| รูปที่ 4.9 การค้นหาคำที่ต้องการภายในเอกสาร..... | 41 |
| รูปที่ 4.10 เมนู pop-up แสดงคำสั่งการเลือกหน้าที่ต้องการ..... | 42 |
| รูปที่ 4.11 เมนู pop-up แสดงคำสั่งการเลือกความละเอียดของหน้าเอกสาร..... | 43 |
| รูปที่ 4.12 เมนู pop-up แสดงการเลือกคำสั่ง Navigation Pane..... | 43 |
| รูปที่ 4.13 ภาพเอกสาร DjVu ที่เป็นลายมือเขียนแสดงที่ระดับความละเอียดเท่ากับ100%..... | 46 |
| รูปที่ 4.14 ภาพเอกสาร DjVu ที่เป็นลายมือเขียนแสดงที่ระดับความละเอียดเท่ากับ150%..... | 46 |
| รูปที่ 4.15 การเรียกดูเอกสารภาพ index.djvu ผ่านเว็บ..... | 48 |

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันเครือข่ายอินเทอร์เน็ตเข้ามามีบทบาทในชีวิตของมนุษย์ การติดตามข่าวสาร ค้นหาข้อมูลจากสถานที่ต่างๆ สามารถทำได้โดยสะดวก ทำให้จำนวนเครื่องคอมพิวเตอร์ที่ให้บริการเว็บ เกิดขึ้นอย่างมากมาภายในเวลาเพียงไม่กี่ปี ขบวนการจัดทำข้อมูล(Content) ที่จะบรรจุลงบนแต่ละหน้าของเว็บถูกจัดทำขึ้นใหม่จากข้อมูลเก่า อาจจะเป็นโดยการสร้างขึ้นใหม่ หรือจากการสแกนเอกสารเก่าแล้วบรรจุลงเป็นเครื่องคอมพิวเตอร์เครือข่ายในรูปแบบข้อมูลที่เป็นภาพเอกสาร

ปัญหาที่เกิดขึ้นจากการสแกนหน้าเอกสารแล้วเก็บข้อมูลในรูปแบบภาพเอกสารสำหรับการนำเสนอผ่านเว็บคือข้อมูลที่ได้จะมีขนาดใหญ่มาก ตัวอย่างเช่นหน้าเอกสารขนาด A4 ที่ทำการสแกนที่ระดับความละเอียด 300 จุดต่อนิ้ว สำหรับเอกสารภาพสี จะมีขนาดของแฟ้มภาพเอกสารที่ประมาณ 25 MB ทำให้การจัดเก็บเอกสารจำนวนมากต้องใช้เครื่องเครือข่ายที่มีเนื้อที่จัดเก็บข้อมูลขนาดใหญ่มาก และการเรียกใช้จากผู้ใช้งานก็ต้องเสียเวลาในการดาวน์โหลดข้อมูลเป็นเวลานาน ยิ่งไปกว่านั้นทำให้เครือข่ายมีการส่งข้อมูลมากอาจทำให้เกิดความล่าช้าแก่ผู้ใช้งานคนอื่นที่ใช้งานในเครือข่ายเดียวกันก็เป็นได้

แนวทางในการแก้ปัญหานี้จึงได้มีงานวิจัยที่ทำการบีบอัดเอกสารภาพกับภาพสแกนก่อนที่จะบรรจุลงบนคอมพิวเตอร์เครือข่ายเพื่อให้ผู้ใช้งานเข้าถึงได้ผ่านเว็บ โดยได้ทำการบีบอัดภาพเอกสารด้วยระเบียบวิธีที่มีชื่อว่า JBIG1 และได้ทำระบบการเข้าถึงเอกสารผ่านระบบจัดการเอกสาร HTML[1] โดยเรียกถึงเอกสารภาพผ่านโปรแกรมประเภทบราวส์เซอร์(browser) เช่น โปรแกรมอินเทอร์เน็ตเอกซ์พลอร์เรอร์ ซึ่งได้ทำการติดตั้งโปรแกรมปลั๊กอิน(Plug-in) เรียบร้อยแล้ว

การบีบอัดเอกสารภาพด้วยระเบียบวิธี JBIG1 เหมาะกับเอกสารภาพขาวดำสองระดับ ที่มีแต่ตัวหนังสือเท่านั้น แม้ว่า JBIG1 สามารถบีบอัดภาพ Gray Scale ได้ประสิทธิภาพจะไม่ดีนัก

วิทยานิพนธ์ของ สิทธิชัย อ่างวัฒนชัย [2] ได้นำเสนอการประยุกต์ใช้วิธีการบีบอัดที่มีชื่อว่า DjVu เพื่อใช้เป็นระเบียบวิธีในการบีบอัดภาพเอกสารในระบบการจัดเก็บและเรียกคืนเอกสารผ่านเว็บ ซึ่งสามารถทำการจัดเก็บเอกสารที่ประกอบด้วยตัวหนังสือและรูปภาพ ทั้งสีและขาวดำได้อย่างมีประสิทธิภาพทั้งในด้านคุณภาพของเอกสาร และขนาดของแฟ้มเอกสาร

ในวิทยานิพนธ์ของ สิทธิชัย อังกรวิวัฒน์ชัย ได้พัฒนาโปรแกรมประยุกต์บนระบบปฏิบัติการวินโดวส์ที่มีชื่อว่า DjVuDocumentOrganizer เพื่อให้ทำงานในส่วนของจัดเก็บระบบเอกสารภาพ DjVu โปรแกรมนี้ทำงานบนเครื่องไมโครคอมพิวเตอร์ โดยได้พัฒนาจากเครื่องมือในการพัฒนาโปรแกรมที่มีชื่อว่า DjVu SDK ของบริษัท AT&T

โปรแกรม DjVuDocumentOrganizer มีความสามารถดังต่อไปนี้ คือ

1. สามารถเข้ารหัสและถอดรหัสเพิ่มให้เป็นรูปแบบ DjVu ได้
2. สามารถแสดงผลเอกสารภาพชนิด DjVu, BMP และ JPG บนจอภาพได้ รวมถึงสามารถย่อ-ขยายระหว่างการเรียกดูได้
3. สามารถสร้างความเชื่อมโยงระหว่างเอกสารภาพ DjVu ที่มาจากสิ่งพิมพ์เดียวกันเพื่อให้เป็นเอกสารเดียวกันแบบเรียงต่อเนื่องได้ขณะเรียกดูผ่านเว็บหรือแบบออฟไลน์
4. สามารถใส่การเชื่อมโยงกับชื่อเว็บอื่น (URL) ไว้บนเอกสารภาพ DjVu แบบผู้ใช้กรอก ซึ่งสามารถคลิกเพื่อเข้าถึงได้ขณะเรียกดูด้วยโปรแกรมบราวส์เซอร์
5. สามารถทำการรู้จำตัวเลขในหน้าสารบัญและหน้าดัชนีแบบมีความช่วยเหลือจากผู้สร้างเอกสาร แล้วใส่การเชื่อมโยงกับแฟ้มเอกสารภาพที่ตรงกับเลขหน้านั้นๆ ได้

สำหรับการเรียกดูจากผู้ใช้งานผ่านเว็บนั้นผู้ใช้งานไม่สามารถเรียกดูหน้าเอกสารผ่านเครือข่ายได้โดยตรงเพราะโปรแกรมที่เป็นเว็บเบราว์เซอร์ทั้งหลายไม่สามารถแสดงผลภาพชนิดนี้ได้ ดังนั้นการเรียกดูจากผู้ใช้งานเว็บก็จะต้องทำการดาวน์โหลดปลั๊กอินเพิ่มเติม(ที่มีขนาดประมาณ 2 MB) แล้วติดตั้งลงบนเครื่องไมโครคอมพิวเตอร์ของผู้ใช้เพื่อให้โปรแกรมบราวส์เซอร์สามารถแสดงผลภาพชนิด DjVu ได้

ปัญหาสำคัญที่เกิดจากการดาวน์โหลดปลั๊กอินมาติดตั้งโดยที่ผู้ใช้อาจคาดไม่ถึงได้แก่ปัญหาด้านความปลอดภัย เพราะว่าบางทีโปรแกรมที่ดูเหมือนจะทำงานปกติแต่อาจจะทำความเสียหายให้กับระบบ หรือถูกใช้ให้ทำลายระบบรักษาความปลอดภัยที่รัดกุม เช่น โปรแกรมไวรัสคอมพิวเตอร์ เป็นต้น ดังนั้นการที่ผู้ใช้ไม่ต้องดาวน์โหลดปลั๊กอินเพิ่มเติมก็เป็นวิธีหนึ่งที่จะช่วยป้องกันความเสียหายที่จะเกิดขึ้นกับระบบของ client ได้ และยังทำให้เกิดความสะดวกกับผู้ใช้งานอีกด้วย เพราะโปรแกรมบราวส์เซอร์สามารถแสดงผลภาพชนิด DjVu ได้เลย

เนื่องจากงานวิจัยของ สิทธิชัย [1] ไม่ได้ทำการพัฒนาในส่วนของปลั๊กอินจึงไม่สามารถจัดการกับปัญหาอย่างที่กล่าวมาข้างต้น ในวิทยานิพนธ์นี้จึงมีแนวคิดที่จะทำการพัฒนาโปรแกรมในส่วนของเรียกคืนภาพเอกสารด้วยโปรแกรมภาษาจาวา ที่เป็นแอปเพล็ต (Applet) เพื่อให้ผู้ใช้สามารถที่จะเรียกดูเอกสารภาพชนิด DjVu บนเว็บเบราว์เซอร์ ได้เลย โดยที่ไม่ต้องเสียเวลาในการดาวน์โหลดโปรแกรมปลั๊กอิน (ที่มีขนาดประมาณ 2 MB) มาติดตั้งเพิ่มเติม

1.2 วัตถุประสงค์ของวิทยานิพนธ์

พัฒนาโปรแกรมเรียกคืนภาพเอกสาร DjVu ที่สามารถนำไปใช้งานได้จริงเพื่อเผยแพร่ข้อมูลเอกสารผ่านทางระบบอินเทอร์เน็ต

1.3 ขอบเขตของวิทยานิพนธ์

1. พัฒนาโปรแกรมในส่วนเรียกคืนภาพเอกสาร DjVu ที่มีชื่อว่า DjVuApplet ด้วยโปรแกรมภาษาจาวา ที่เป็นแอปเพล็ต (Applet) เพื่อให้ผู้ใช้สามารถที่จะเรียกดูเอกสารภาพชนิด DjVu บนเว็บเบราว์เซอร์ ได้เลย โดยที่ไม่ต้องเสียเวลาในการดาวน์โหลดโปรแกรมปลั๊กอินมาติดตั้งเพิ่มเติม
2. พัฒนาฟังก์ชันในส่วนติดต่อกับผู้ใช้ (UI) ซึ่งจะช่วยให้การโต้ตอบกับผู้ใช้ทำได้ง่ายขึ้น
3. ทดสอบเอกสารภาพ DjVu หนึ่งฉบับที่ได้มีการสแกนไว้แล้ว เมื่อทำทุกขั้นตอนเสร็จแล้วจากนั้นจึงอัปโหลดไปเก็บไว้ในเครื่องคอมพิวเตอร์ที่เป็นเครื่องเว็บเซิร์ฟเวอร์ เพื่อสามารถเรียกดูได้

1.4 ขั้นตอนและวิธีในการดำเนินงาน

1. ศึกษาระเบียบวิธีการบีบอัดภาพเอกสารแบบ DjVu
2. ศึกษาโครงสร้างแฟ้มข้อมูลที่มีชื่อว่า EA IFF85
3. ศึกษาวิธีการเรียกใช้ไลบรารีสำหรับการบีบอัดรูปแบบ DjVu ที่ทางบริษัท AT&T อนุญาตให้ใช้สำหรับงานวิจัย (DjVu API SDK)
4. ศึกษาโปรแกรมประยุกต์บนระบบปฏิบัติการวินโดวส์ที่มีชื่อว่า DjVuDocumentOrganizer ที่พัฒนาโดย สิทธิชัย อังระวัฒน์ชัย [1]
5. ออกแบบและพัฒนาโปรแกรมเพื่อให้ผู้ใช้สามารถเรียกดูเอกสารภาพชนิด DjVu ได้เลย โดยที่ไม่ต้องเสียเวลาในการดาวน์โหลดโปรแกรมปลั๊กอิน (ที่มีขนาดประมาณ 2 MB) มาติดตั้งเพิ่มเติม
6. ทำการแปลงไลบรารีสำหรับการบีบอัดข้อมูลให้เป็นรูปแบบ DjVu ที่ทางบริษัท AT&T อนุญาตให้ใช้สำหรับงานวิจัย จาก C++ ให้อยู่ในรูปแบบของภาษาจาวา
7. พัฒนาโปรแกรมในส่วนติดต่อกับผู้ใช้ (User Interface)
8. สรุปผลการทดสอบและเขียนวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

- สามารถเรียกดูภาพเอกสาร DjVu ผ่านโปรแกรมบราวเซอร์ได้เลย โดยที่ไม่ต้องดาวน์โหลดโปรแกรมปลั๊กอินมาติดตั้งเพิ่มเติม(ที่มีขนาดประมาณ 2 MB) ซึ่งเป็นวิธีหนึ่งที่จะช่วยป้องกันความเสียหายที่จะเกิดขึ้นกับระบบของไคลเอ็นท์(client) ได้ และยังทำให้เกิดความสะดวกกับผู้ใช้งานอีกด้วย
- นอกจากนี้ยังเป็นการส่งเสริมให้เกิดการพัฒนาซอฟต์แวร์แบบ open source ขึ้นใช้ภายในประเทศเพื่อลดการขาดดุลทางการค้าในอนาคตได้



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

เนื้อหาในบทนี้จะกล่าวถึงความรู้พื้นฐานที่เกี่ยวข้องกับงานวิจัย ได้แก่ ระเบียบวิธีการบีบอัดภาพเอกสารสแกนที่มีชื่อว่า DjVu, ระบบภาพเอกสาร และงานวิจัยที่เกี่ยวข้อง

2.1 ระเบียบวิธีการบีบอัดภาพเอกสาร DjVu

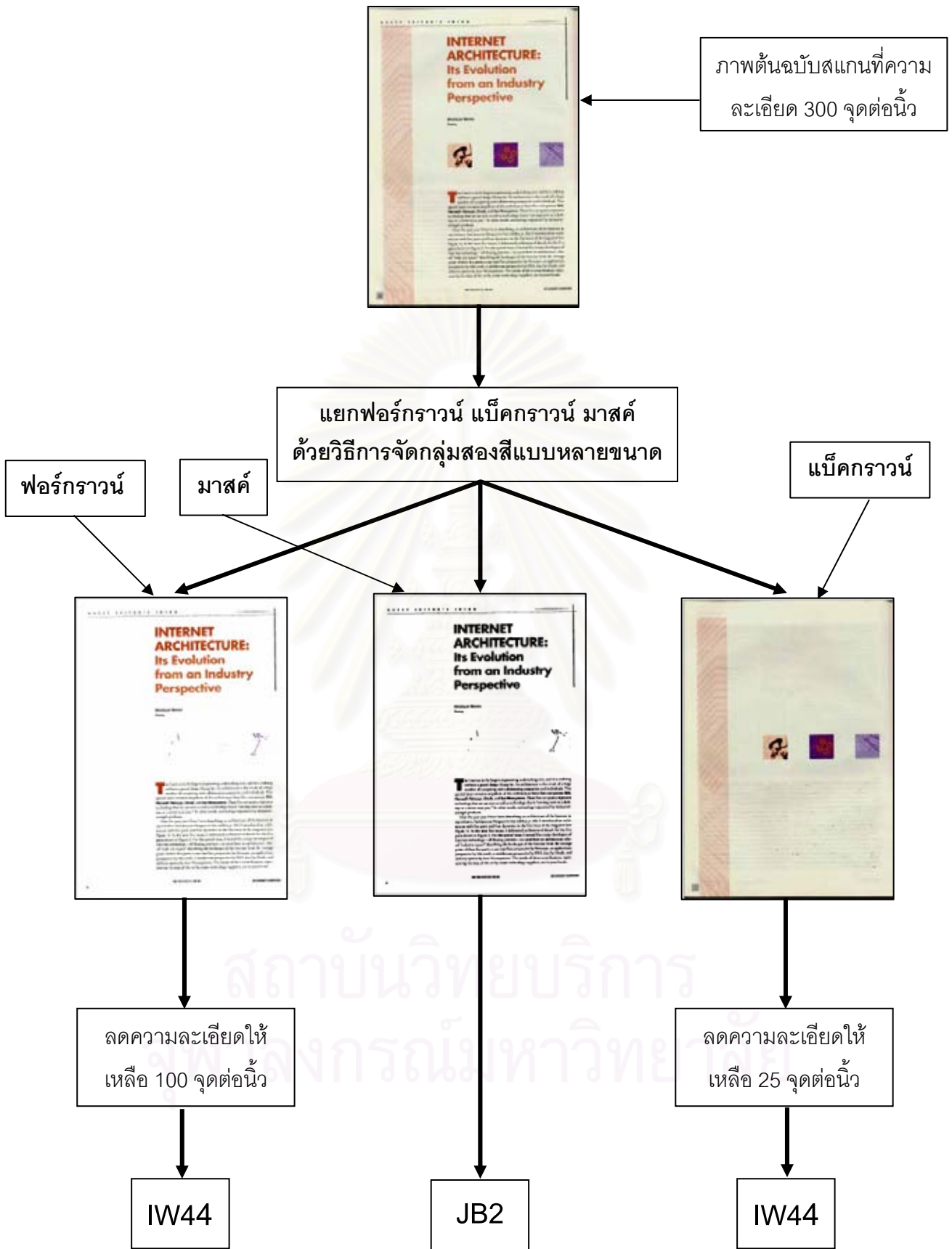
DjVu (อ่านออกเสียงว่า เดจาเวอ หรือ dejavu เป็นภาษาฝรั่งเศสแปลว่าที่ได้ประสบมาแล้ว) เป็นระเบียบวิธีในการบีบอัดเอกสารภาพสแกนที่มีความละเอียดและคุณภาพสูงพัฒนาโดยบริษัท AT&T ด้วยเทคนิคนี้ทำให้สามารถอ่านข้อมูลภาพสแกนจากหนังสือหรือสื่อที่เป็นกระดาษมาแสดงบนจอได้ตรงตามต้นฉบับทั้งแบบตัวอักษร, สี, ลายเส้น, รูปภาพ, และลายกระดาษหน้ากระดาษนิตยสารที่มีความละเอียด 300 จุดต่อนิ้ว ถูกบีบอัดจนขนาดเหลือเพียง 40-60 กิโลไบต์ (ซึ่งอยู่ในระดับเดียวกับแฟ้มเอกสาร HTML โดยเฉลี่ยในปัจจุบัน) ซึ่งดีกว่า JPEG ประมาณ 5-10 เท่าที่ระดับคุณภาพเดียวกัน ส่วนเอกสารภาพขาวดำจะเหลือขนาดประมาณ 15-30 กิโลไบต์ ที่ 300 จุดต่อนิ้วเช่นกัน ซึ่งดีกว่ามาตรฐาน CCITT-G4 ประมาณ 4-8 เท่า

2.1.1 หลักการของระเบียบวิธีการบีบอัดเอกสารภาพ DjVu [3]

หลักการบีบอัดนี้คือทำการแยกภาพสแกนออกเป็นชั้น ๆ ได้แก่

- ชั้นฟอกราวน์ (Foreground) จะเป็นข้อมูลจำพวกลายเส้นของตัวหนังสือ, ลายเส้น
- ชั้นแบ็คกราวน์ (Background) จะเป็นข้อมูลจำพวกรูปภาพซึ่งใช้ความละเอียดเพียงแค่ 50-100 จุดต่อนิ้วก็มีคุณภาพที่ยอมรับได้
- ชั้นมาสก์ (Mask) จะเป็นชั้นที่เป็นภาพสองสีโดยที่แต่ละบิตจะบอกว่าจุดภาพนั้นเป็นข้อมูลที่จะใช้สีของชั้นแบ็คกราวน์หรือจากชั้นฟอกราวน์ ข้อมูลที่ปรากฏอยู่บนชั้นมาสก์จะเป็นประเภทตัวหนังสือ, ลายเส้น

เทคนิคที่ใช้แยกชั้นแบ็คกราวน์ ออกจากชั้นฟอกราวน์มีชื่อว่าการจัดกลุ่มสองสีแบบหลายขนาด (Multi Scale Bicolor Clustering)



รูปที่ 2.1 หลักการบีบอัดภาพเอกสารแบบ DjVu

เมื่อแยกภาพออกเป็นสามส่วนดังกล่าวแล้วจึงใช้ระเบียบวิธีในการบีบอัดข้อมูลที่เหมาะสมกับชนิดข้อมูลในชั้นนั้น ๆ เพื่อทำการบีบอัดข้อมูลอีกทีหนึ่งดังรูปที่ 2.1

เทคนิคที่ใช้สำหรับการบีบอัดข้อมูลในแต่ละชั้นได้แก่

- ชั้นแบ็คกราวนด์จะใช้เทคนิคที่มีชื่อว่า IW44 (Deslauriers-Dubuc interpolating wavelets with four analyzing moments and four vanishing moments [6])
- ชั้นฟอร์กราวนด์จะใช้เทคนิคที่มีชื่อว่า IW44 ในการบีบอัดเช่นเดียวกัน
- ชั้นมาสก์จะใช้เทคนิคที่มีชื่อว่า JB2 ซึ่งเป็นระเบียบวิธีการบีบอัดข้อมูลภาพสองสีโดยจะทำการละเลียดสูงคือที่ระดับความละเอียด 300 จุดต่อนิ้ว

2.1.2 เทคนิคที่ใช้ในการแยกฟอร์กราวนด์และแบ็คกราวนด์ (The Foreground / Background Separation) [3],[4]

สาเหตุที่ DjVu สามารถบีบอัดเอกสารภาพได้เป็นอย่างดีก็เพราะว่าใช้หลักการแยกข้อมูลออกจากกันแล้วทำการบีบอัดที่เหมาะสมกับข้อมูลแต่ละชนิด ดังนั้นการแยกภาพออกเป็นชั้นจึงมีความสำคัญมาก แนวความคิดหลักที่ใช้ในการแยกฟอร์กราวนด์และแบ็คกราวนด์ออกจากกันคือ ขั้นตอนแรกในการบีบอัด DjVu จะต้องทำการแยกส่วนของตัวหนังสือและลายเส้นออกจากรูปภาพให้ได้ มีหลายเทคนิคที่สามารถนำมาใช้สำหรับงานนี้ได้ แบ่งเป็นสองลักษณะใหญ่คือ

1. วิธีที่ทำจากบนลงล่าง (Top-down) โดยการแบ่งภาพออกเป็นบริเวณเล็ก ๆ แล้วค่อยตัดสินใจว่าแต่ละบริเวณเป็นส่วนของตัวหนังสือ, ลายเส้น หรือภาพ วิธีนี้จะใช้ได้ดีกับเอกสาร 2 สี หรือเอกสารที่มีสีของแบ็คกราวนด์เพียงสีเดียวเท่านั้น
2. วิธีที่ทำจากล่างขึ้นบน (Bottom-up) วิธีนี้จะพยายามดึงคุณลักษณะของตัวหนังสือให้ได้ ที่นิยมกันมากคือเราจะใช้คุณสมบัติของตัวหนังสือที่ไม่เหมือนแบ็คกราวนด์ สำหรับตัดสินใจแยกทั้งสองส่วนออกจากกัน

เทคนิคที่ใช้แยกแบ็คกราวนด์ออกจากฟอร์กราวนด์ที่สนใจคือระเบียบวิธีการจัดกลุ่ม 2 สีแบบหลายขนาด (Multi – scale bicolor clustering algorithm) ซึ่งเป็นวิธีกลาง ๆ ระหว่างวิธี top-up และ bottom-down คือจะทำการแยกส่วนฟอร์กราวนด์ออกจากแบ็คกราวนด์โดยตรง เริ่มต้นจะทำการแบ่งภาพออกเป็นช่องสี่เหลี่ยมจัตุรัสเท่า ๆ กันของหลาย ๆ จุดภาพ แล้วทำการหาสีหลัก 2 สี สำหรับแต่ละช่องจากนั้นจะใช้ระเบียบวิธี realization ทำให้แน่ใจว่าช่องข้างเคียงกำหนดให้สีเดียวกันเป็นของฟอร์กราวนด์หรือแบ็คกราวนด์ หลังจากเสร็จขั้นตอนนี้แล้ว แต่ละ

จุดภาพจะถูกกำหนดให้เป็นสีที่มาจากฟอร์กราวน์หรือแบ็คกราวน์ทันที โดยการหาระยะห่างของสีของแต่ละจุดภาพว่าใกล้กับสีที่กำหนดให้เป็นสีของฟอร์กราวน์หรือแบ็คกราวน์มากกว่ากัน

แม้ว่าวิธีนี้ค่อนข้างง่ายแต่ก็ให้ความถูกต้องในระดับที่สูงมาก จึงเลือกเปรียบเทียบวิธีนี้มาใช้ในการแยกองค์ประกอบของการบีบอัดภาพ DjVu

2.1.3 หลักการบีบอัดภาพชั้นมาสก์ด้วยวิธี JB2 (Coding the Bilevel Mask Using JB2)

เดิมแม้ว่าจะมีวิธีการบีบอัดภาพสองระดับที่มีประสิทธิภาพอยู่แล้วคือ JBIG1 แต่ก็มีข้อจำกัดอยู่ที่ว่าใช้สำหรับการบีบอัดภาพในโหมดที่ไม่มีการสูญเสีย (lossless) ข้อมูลเท่านั้น ต่อมาจึงได้มีการเสนอมาตรฐานใหม่คือ JBIG2 ขึ้นมาเพื่อให้มีการบีบอัดในโหมดสูญเสีย (lossy)

แนวความคิดพื้นฐานของ JB2 ได้แก่ Model base coding ดังรายละเอียดต่อไปนี้ [5]

- ภาพจะถูกแบ่งออกเป็นส่วน ๆ ที่เรียกว่า Mark ก่อน (บางครั้งเรียกว่า Symbol คือเป็นภาพบิตแม็พขนาดเล็กนั่นเอง) ซึ่งจะเป็นส่วนของจุดภาพสีดำที่ต่อเนื่องกัน
- Mark แต่ละตัวจะถูกจัดกลุ่มเป็นลำดับชั้นตามความเหมือน โดยวัดจากระยะห่างที่เหมาะสม
- Mark บางส่วนจะถูกบีบอัดและเข้ารหัสโดยตรงโดยใช้แบบจำลองทางสถิติ และการเข้ารหัสเลขคณิต (arithmetic-coding)
- Mark ที่เหลือจะถูกบีบอัดและเข้ารหัสทางอ้อม โดยอาศัยความรู้จาก Mark ก่อนหน้าที่ผ่านการเข้ารหัสแล้ว
- ภาพถูกเข้ารหัสโดยการระบุหมายเลขประจำแต่ละ Mark และตำแหน่งของ Mark นั้น ๆ เทียบกับ Mark ก่อนหน้า

มีหลายวิธีที่ใช้ในการแบ่งกลุ่มและเข้ารหัสแบบมีเงื่อนไขของ mark แต่วิธีที่เลือกใช้สำหรับเปรียบเทียบวิธีการเข้ารหัสแบบ DjVu คือวิธีที่มีชื่อว่า “soft pattern matching”

2.1.4 Model Base Coding [5]

ในมาตรฐาน JBIG1 จะมีการคาดเดา pixel แต่ละจุดโดยใช้ความรู้จากจุดภาพข้างเคียงมาช่วย โดยจะมีการกำหนดรูปแบบไว้ตายตัว (fix template) โมเดลในลักษณะนี้จะเป็นการเก็บโครงสร้างของตัวอักษรบางส่วนมาด้วย

ในมาตรฐาน JBIG2 จะมีการใช้ประโยชน์จากความรู้ที่ว่าส่วนใหญ่เอกสารภาพสองระดับมักจะประกอบด้วยตัวหนังสือจึงมีการอนุญาตให้ใช้โมเดลที่ถูกออกแบบเฉพาะสำหรับข้อมูลประเภทนั้น ๆ ทำให้ JBIG2 จะไม่ขึ้นกับชนิดของตัวอักษร

สำหรับเอกสารที่ประกอบด้วยตัวหนังสือจะมีความซ้ำกันมากจึงมีการเข้ารหัสตัวอักษรแล้วเก็บไว้ในพจนานุกรม (dictionary) ไว้เพื่ออ้างอิง ในหัวข้อนี้จะกล่าวถึงวิธีการเข้ารหัส 2 วิธีได้แก่ Pattern matching and substitution และ Soft pattern matching ซึ่งวิธีหลังจะเป็นวิธีที่ใช้ในระเบียบวิธี JB2 สำหรับการเข้ารหัสภาพขั้นมาสดค้ของระเบียบวิธีการบีบอัดเอกสารภาพ DjVu

2.1.4.1 Pattern matching and substitution (PM&S)

ทำตามขั้นตอนดังต่อไปนี้ (รูปที่ 2.2)

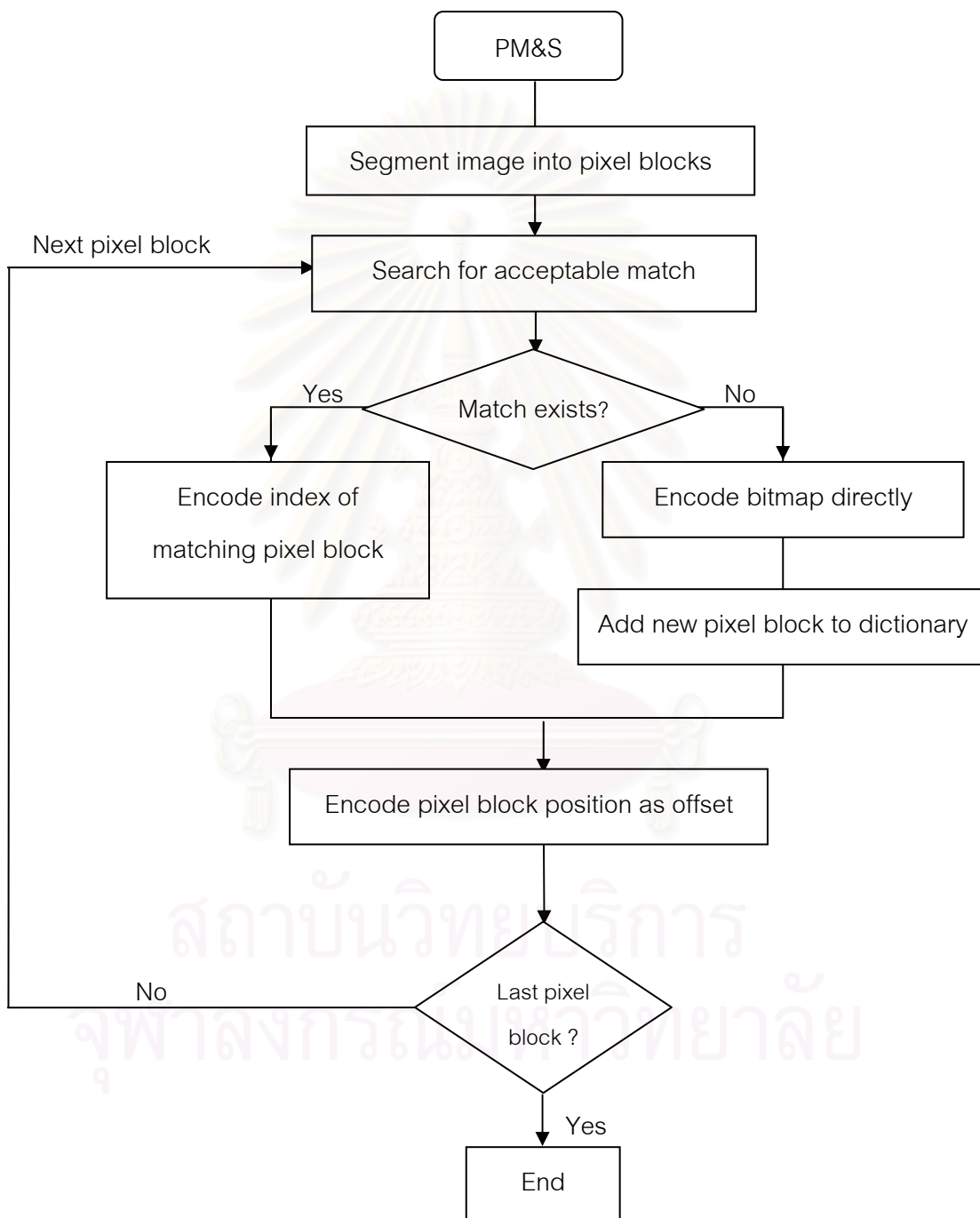
1. แบ่งภาพออกเป็น pixel block ที่เรียกว่า mark โดยใช้เทคนิคการแบ่งส่วน (segmentation) แบบมาตรฐาน
2. หาส่วนที่เหมือนกับ pixel block นี้ใน dictionary ที่พอยอมรับได้ โดยจะพิจารณาข้ามไปถ้าคุณลักษณะเด่นของ pixel block ที่นำมาเปรียบเทียบกับ ส่วนสูง ความกว้าง หรือพื้นที่ของบริเวณนั้นไม่ใกล้เคียงกับ pixel block อันที่กำลังค้นหา
3. ถ้ามี pixel block ที่เหมือนก็จะทำการเข้ารหัสเลขดัชนีของ block นั้น ๆ ด้วยวิธีการเข้ารหัสแบบ Huffman
4. ถ้าไม่มี pixel block ใดเหมือนก็จะทำการเพิ่มเข้าไปในพจนานุกรม คือทำการเข้ารหัส pixel block ภาพนั้น ๆ โดยใช้เทคนิค MMR (modified modified READ) ก็ได้
5. ทำการเข้ารหัสตำแหน่งของ pixel block เป็น offset
6. ทำซ้ำจนหมด pixel block

จุดอ่อนของวิธีนี้คือความผิดพลาดที่เกิดจากการแทนผิด เพราะความต้องการที่จะให้อัตราการบีบอัดสูงจึงต้องบังคับให้มีความเหมือนกันมากที่สุด ซึ่งจะส่งผลให้ mark ที่มีความต่างกันแต่แตกต่างกันไม่มากพอจะถูกแทนที่ผิดพลาดได้

2.1.4.2 Soft pattern matching (SPM)

คล้าย ๆ กับวิธีการ Pattern matching and substitution แต่แทนที่เราจะแทนที่ mark ที่เหมือนด้วย mark อันที่อยู่ใน dictionary วิธีการนี้จะใช้บิตแม็พนั้นเพื่อปรับปรุงค่าคอนเท็กที่ใช้ใน

การเข้ารหัสแทนสำหรับกรณีการบีบอัดแบบไม่มีการสูญเสียข้อมูล (lossless) แต่สำหรับกรณีการบีบอัดที่มีการสูญเสียข้อมูล (lossy) จะมีขั้นตอนการ preprocessing หรือ ขั้นตอนการ post-processing ก่อนและหลังการทำ pattern matching นั้นเอง แต่ในที่นี้จะกล่าวถึงแต่ขั้นตอนการ preprocessing เท่านั้น



รูปที่ 2.2 แผนภาพวิธีการทำ Pattern matching and substitution [5]

2.1.4.2.1 แบบไม่มีการสูญเสียข้อมูล (lossless)

ส่วนแรกของขบวนการ soft pattern matching ทำตามขั้นตอนต่อไปนี้ (รูปที่ 2.3)

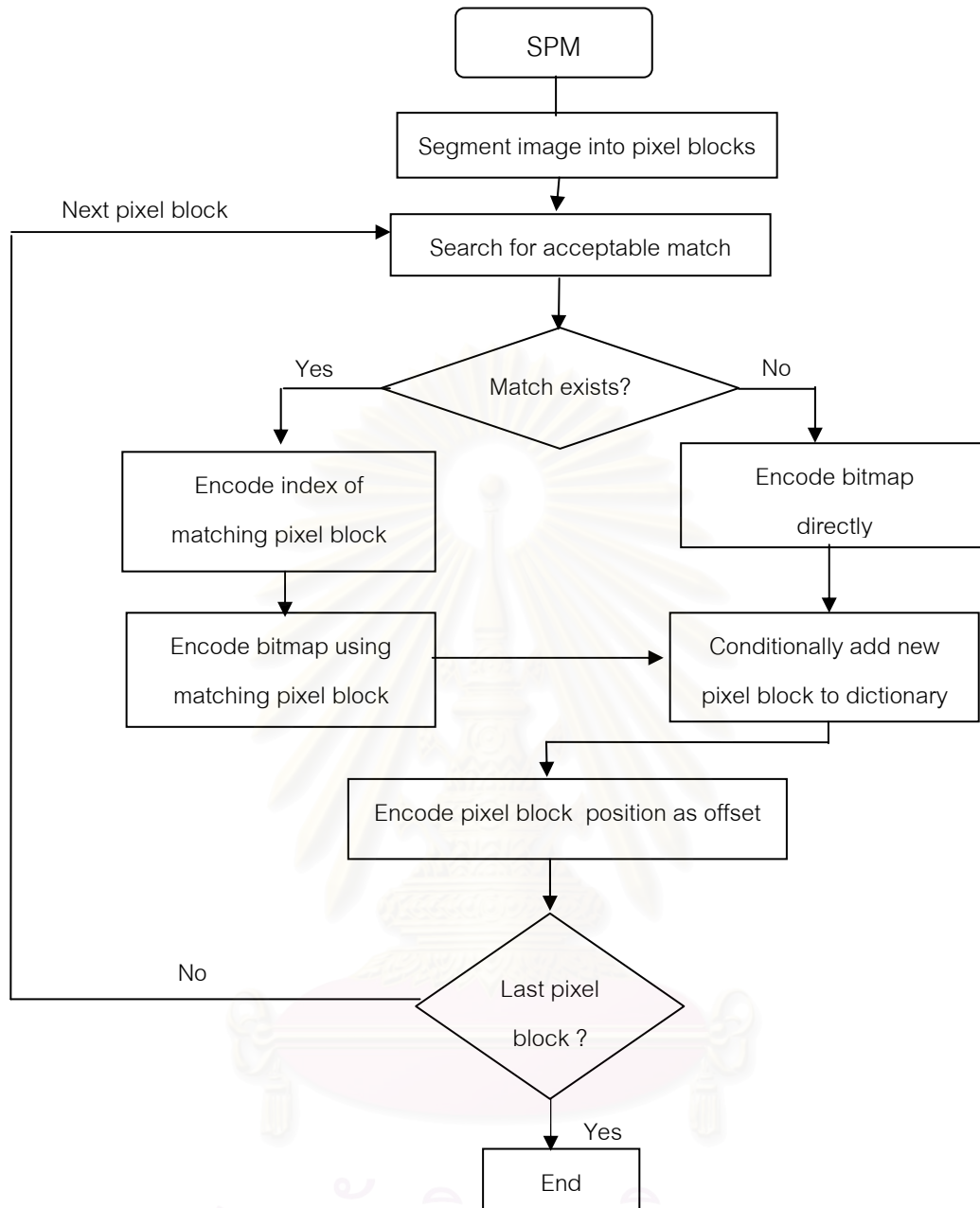
1. แบ่งภาพออกเป็น pixel block โดยใช้เทคนิคการแบ่งส่วน แบบมาตรฐาน
2. หาส่วนที่เหมือนกับ pixel block นี้ใน dictionary ที่พอยอมรับได้
3. ถ้ามี pixel block ที่เหมือนกันก็จะทำการเข้ารหัสข้อมูลเลขดัชนีของ pixel block นั้น ๆ โดยใช้ arithmetic code และเข้ารหัสภาพบิตแมพโดยใช้ mark ที่เข้าคู่กับอันนั้น ๆ หลังจากนั้นก็จะทำการเพิ่ม mark เข้าสู่ dictionary แบบมีเงื่อนไข
4. ถ้าไม่มีที่เหมือนก็ทำการเข้ารหัสภาพบิตแมพโดยตรง แล้วเพิ่ม mark เข้าสู่พจนานุกรมแบบมีเงื่อนไขเช่นกัน
5. ทำการเข้ารหัสตำแหน่งของ pixel block เป็นค่า offset
6. ทำซ้ำจนครบทุก pixel block

ส่วนที่สองของขบวนการจะประกอบด้วยการเข้ารหัสภาพบิตแมพของ pixel block แบบปราศจากความสูญเสีย โดยการจัดให้ pixel block ที่ตรงกันกับที่ค้นหา แล้วทำการเข้ารหัสจุดภาพทั้งหมดในขอบของ pixel block โดยใช้ตัวเข้ารหัส Arithmetic แบบเดียวกับที่ใช้ในมาตรฐาน JBIG1 แต่ใช้เทมเพลตที่ต่างไปจากที่ใช้ใน JBIG1

2.1.4.2.2 แบบมีการสูญเสียข้อมูล (lossy)

ปกติการเข้ารหัสเลขคณิต (Arithmetic coding) จะเป็นแบบไม่มีการสูญเสียข้อมูล (lossless) แม้จะมีการทำ Soft pattern matching ก็ตาม การทำ preprocessing บางอย่างสามารถลดขนาดความยาวรหัสของภาพได้โดยไม่มีผลกระทบต่อลักษณะที่ปรากฏแม้จะมีการสูญเสียข้อมูลไปบ้างก็ตาม ขบวนการ preprocessing เช่นนี้ได้แก่

- การควอนไทซ์ค่าออฟเซต (Quantization of Offsets)
- การยอมละเลย mark ที่มีขนาดเล็กและการทำให้เรียบ (Noise Removal and Smoothing)
- การกลับบิต (Bit Flipping)



รูปที่ 2.3 แผนภาพวิธีการทำ Soft pattern matching [5]

2.1.5 หลักการบีบอัดภาพชั้นแบ็คกราวนด์ด้วยวิธี IW44 [6]

เดจาอูได้นำเอาการแปลง wavelet มาใช้สำหรับการบีบอัดภาพชั้นแบ็คกราวนด์ โดยหลักการที่ว่าสายตาของคนไม่สามารถสังเกตเห็นส่วนที่มีความถี่สูงในรูปภาพได้ เราจึงสามารถตัดข้อมูลส่วนนั้นออกไปได้โดยที่คุณภาพของภาพยังเป็นที่ยอมรับได้

การแปลง wavelet มีคุณสมบัติในการแปลงข้อมูลไปเป็นข้อมูลอีกรูปแบบหนึ่ง ซึ่งข้อมูลที่มีความถี่ต่างกันจะถูกแยกออกมาเป็นระดับทำให้เราสามารถเลือกเก็บข้อมูลเฉพาะส่วนที่มีความสำคัญต่อการมองเห็นของคน

การแยกส่วนประกอบความละเอียดหลายระดับ (Multi-resolution wavelet decomposition) เป็นระเบียบวิธีที่มีประสิทธิภาพมากที่สุดอย่างหนึ่งสำหรับการเข้ารหัสภาพสี และจะเป็นทางเลือกหนึ่งที่สำคัญสำหรับมาตรฐานการบีบอัดภาพสีหรือ gray scale โดยเริ่มต้นภาพจะถูกแทนด้วยผลรวมเชิงเส้นของ wavelet และจากสมมติฐานที่ว่าภาพมีความราบเรียบจึงทำให้แน่ใจได้ว่าการกระจายของค่าสัมประสิทธิ์ของ wavelet จะกระจุกตัวและหนาแน่นอยู่บริเวณศูนย์กลาง ด้วยเหตุนี้การบีบอัดที่มีประสิทธิภาพสามารถทำได้โดยการใช้ Quantization และการเข้ารหัสที่ใช้แนวความคิดที่ว่ามีการกระจุกตัวที่ศูนย์กลาง

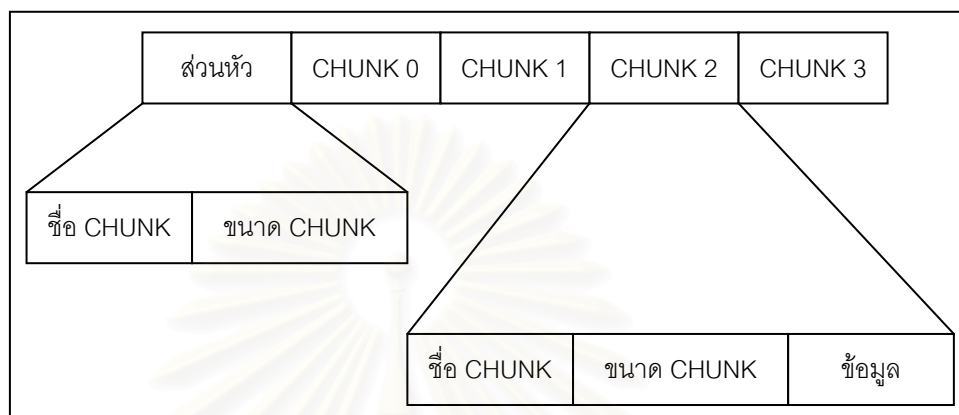
เนื่องจากสมมติฐานที่ว่ามีความราบเรียบจึงใช้ระเบียบวิธีการบีบอัด Wavelet ในการเข้ารหัสรูปฉากหลังสำหรับการบีบอัดภาพแบบ DjVu นั้นเอง ซึ่งระเบียบวิธีการบีบอัด Wavelet ที่เลือกมาใช้คือ Deslauriers-Dubuc-Limire interpolating wavelet โดยใช้ analyzing moment เท่ากับ 4 และ Vanishing moment เท่ากับ 4 [5] เมื่อได้ค่าสัมประสิทธิ์ของ wavelet แล้วก็จะทำการเข้ารหัสโดยอาศัยวิธีเข้ารหัสเลขคณิต (Arithmetic coding) และใช้เทคนิคที่มีชื่อว่า hierarchical Set Difference

ระหว่างที่ทำการคลายการบีบอัด ค่าสัมประสิทธิ์ Wavelet จะถูกแทนในรูปของ compact sparse array ซึ่งแทบจะไม่ใช้หน่วยความจำเลยในกรณีของสัมประสิทธิ์เท่ากับศูนย์กลาง การใช้เทคนิคนี้จะทำให้ภาพฉากหลังที่มีขนาดใหญ่จะใช้เนื้อที่ของหน่วยความจำเพียงหนึ่งในสี่ที่ต้องการโดยจุดภาพทั้งหมด ชิ้นส่วนของภาพที่ถูกแสดงบนจอภาพสามารถสร้างตามความต้องการเท่านั้นจาก sparse array นี้มันเอง ด้วยเหตุนี้ทำให้โปรแกรมที่ใช้ดูภาพจึงใช้หน่วยความจำน้อยกว่าที่ควร

2.2 รูปแบบแฟ้มข้อมูล DjVu [7]

แฟ้มข้อมูลชนิด DjVu เป็นแฟ้มที่มีรูปแบบตามมาตรฐานที่มีชื่อว่า EA IFF 85 ซึ่งข้อมูลจะถูกแบ่งออกเป็นองค์ประกอบย่อยที่เรียกว่า chunk เรียงต่อเนื่องกันโดยแต่ละ chunk จะประกอบด้วยชื่อของ chunk (chunk id), ขนาดของ chunk (chunk size) และตามด้วยข้อมูลของ chunk นั้นๆ โดยข้อมูลภายใน chunk อาจประกอบด้วย chunk ย่อยๆ ได้ โดยสำหรับ chunk ที่ติดกัน ถ้า chunk ก่อนหน้ามีความยาวเป็นเลขคี่ จะต้องเพิ่ม 0 ลงท้าย chunk ก่อนนั้นแล้วจึงค่อยต่อด้วย chunk ปัจจุบันแต่ไม่รวมเป็นความยาวของ chunk โครงสร้างแฟ้มแบบ EA IFF 85 ดังแสดงในรูปที่ 2.4

นอกจากนี้แฟ้ม DjVu จะมีส่วนหัวเพิ่มเติมได้คือ AT&T (0x41, 0x54, 0x26, 0x54) เพื่อให้โปรแกรมอินเทอร์เน็ตเอ็กพลอร์เรอร์เรียกปลั๊กอิน DjVu เพื่อแสดงผลบนจอภาพ มิฉะนั้นโปรแกรมจะคิดว่าแฟ้มที่ขึ้นแฟ้มด้วยส่วนหัวคือ FORM เป็นแฟ้มชนิดเสียงจึงไม่ส่งให้ปลั๊กอิน DjVu ทำการถอดรหัส



รูปที่ 2.4 ภาพแสดงโครงสร้างแฟ้มชนิด EA IFF 85 รูปแบบที่ใช้เก็บเอกสาร DjVu

แฟ้มเอกสาร DjVu จะแบ่งได้เป็นสองประเภทได้แก่ ชนิดหนึ่งแฟ้มหนึ่งหน้า หมายความว่า ในหนึ่งแฟ้มเอกสารจะประกอบด้วยเอกสารภาพ DjVu เพียงหนึ่งหน้า และชนิดหนึ่งแฟ้มหลายหน้า ซึ่งจะประกอบด้วยเอกสารภาพ DjVu หลายๆ หน้า รวมกันอยู่เป็นแฟ้มเอกสารแฟ้มเดียว

2.3 เปรียบเทียบ DjVu กับรูปแบบการเก็บแฟ้มเอกสารภาพวิธีอื่น ๆ [4]

รูปที่ 2.5 แสดงถึงการเปรียบเทียบเอกสารภาพสแกนสี ทั้งหมด 7 ภาพ ที่มีความละเอียด 300 จุดต่อนิ้ว และ 24 บิตต่อจุดภาพ โดยจะเป็นการเปรียบเทียบเทคนิคการบีบอัดระหว่าง JPEG และ DjVu สาเหตุที่นำเทคนิคการบีบอัดแบบ JPEG มาทำการเปรียบเทียบเพราะว่า JPEG เป็นหนึ่งในมาตรฐานการบีบอัดภาพที่ใช้กันอย่างแพร่หลายในปัจจุบัน

จากรูปที่ 2.5 จะพบว่าผลลัพธ์ที่ได้จากการบีบอัดระหว่าง JPEG และ DjVu เมื่อทำการเปรียบเทียบกับภาพต้นฉบับที่ไม่ได้ผ่านกระบวนการบีบอัด จะสังเกตได้ว่าภาพเอกสารสีที่บีบอัดด้วยวิธี DjVu ตัวหนังสือจะมีความชัดเจนกว่าและยังคงรักษาพื้นฉากหลังไว้ได้ ในขณะที่เมื่อบีบอัดด้วยวิธี JPEG ภาพส่วนที่มีความถี่สูง (ประเภทตัวหนังสือ) จะมีคุณภาพต่ำกว่า นอกจากนี้การบีบอัดด้วยวิธี DjVu จะให้อัตราส่วนของการบีบอัด (Compression ratio) ที่สูงกว่าและยังมีขนาดของแฟ้ม (file) ที่เล็กกว่าอีกด้วย

| Image Description | Raw image detail | JPEG, 300dpi, quality 20 | JPEG, 100dpi, size=DjVu | DjVu compressed |
|--|------------------|--------------------------|-------------------------|-----------------|
| Magazine Ad % image= 56 ads-freehand-300 | 20640K | 292K 70:1 | 50K 412:1 | 52K 396:1 |
| Brattain Notebook % image= 22 brattain-0001 | 9534K | 116K 82:1 | 17K 560:1 | 19K 501:1 |
| Scientific Article % image= 46 graham-001 | 22013K | 383K 57:1 | 41K 536:1 | 38K 579:1 |
| Newspaper Article % image= 50 lrr-wpost-lrr-wpost-1 | 12990K | 250K 51:1 | 38K 341:1 | 40K 324:1 |
| Cross-Section of Jupiter % image= 73 planets-jupiter | 24405K | 284K 85:1 | 47K 519:1 | 47K 519:1 |
| XVIIIth Century book % image= 45 cuisine-p006 | 12128K | 206K 58:1 | 35K 346:1 | 37K 327:1 |
| US First Amendment % image= 30 usa-amend1 | 31059K | 388K 80:1 | 77K 403:1 | 73K 425:1 |

รูปที่ 2.5 แสดงการเปรียบเทียบผลลัพธ์การบีบอัดเอกสารภาพสแกนสีระหว่าง JPEG และ DjVu (ค่า "% image" หมายถึงค่าเปอร์เซ็นต์ของจำนวนบิตที่ต้องใช้สำหรับเข้ารหัสภาพฉากหลัง)

2.4 ระบบภาพเอกสาร

ระบบภาพเอกสารคือวิธีการแปลงข้อมูลในรูปสิ่งพิมพ์บนกระดาษหรือที่เรียกว่า hard copy ให้เป็นข้อมูลประเภทดิจิทัลหรือ soft copy โดยใช้ความสามารถของเครื่องสแกนเนอร์หรือกล้องดิจิทัลซึ่งสามารถถ่ายลักษณะของหน้าเอกสารที่เป็นสิ่งพิมพ์และสามารถบันทึกเป็นแฟ้มภาพเอกสารดิจิทัลได้ทันที

ขบวนการสำหรับวิธีแปลงสิ่งพิมพ์สำหรับเผยแพร่ผ่านอินเทอร์เน็ต เริ่มจากนำสิ่งพิมพ์ที่ต้องการแปลงให้อยู่ในรูปดิจิทัลสแกนหรือถ่ายให้เป็นแฟ้มดิจิทัลที่ละหน้า โดยบันทึกให้อยู่ในรูปแบบแฟ้มภาพดิจิทัลมาตรฐาน หลังจากนั้นก็ทำการเคลื่อนย้ายแฟ้มหน้าเหล่านี้ไปเก็บไว้บน

เครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นเครื่องเว็บเซิร์ฟเวอร์ ซึ่งได้เชื่อมต่อกับเครือข่ายอินเทอร์เน็ต เพื่อให้ผู้ใช้สามารถเข้ามาเรียกชมต่อไป

2.4.1 ข้อดีของระบบการจัดเก็บภาพเอกสาร

- ง่ายเมื่อเปรียบเทียบกับ การสร้างเอกสารขึ้นมาใหม่
- ข้อมูลประเภทรูปวาดและรูปภาพจะไม่สูญหายหรือเปลี่ยนแปลงไปจากต้นฉบับเหมือนกับการสร้างใหม่เพราะจะต้องวาดใหม่ด้วยคอมพิวเตอร์ซึ่งอาจผิดเพี้ยนไปจากต้นฉบับโดยง่าย
- สามารถสร้างเอกสารดิจิทัลขึ้นได้อย่างรวดเร็ว ใช้เวลาน้อยและเผยแพร่ได้ทันทีไม่ต้องมีการตรวจทานเหมือนกับการพิมพ์ใหม่
- สร้างเอกสารดิจิทัลที่เหมือนสิ่งพิมพ์ต้นฉบับได้ทั้งหมด ทั้งที่เป็นรูปแบบ, ขนาดของตัวหนังสือ รูปภาพบนหน้าต่าง ๆ ลายของกระดาษที่ใช้พิมพ์ ลายมือซึ่งรายละเอียดต่าง ๆ เหล่านี้จะเสียไปถ้าใช้วิธีการสร้างใหม่ไม่ว่าจะเป็นการพิมพ์ใหม่ หรือว่าการทำการรู้จำตัวหนังสือ
- รักษาข้อมูลของเอกสารได้อย่างเต็มที่ การสร้างเอกสารขึ้นมาใหม่อาจมีความผิดพลาดในการสร้าง การแปลที่ไม่ถูกต้อง เป็นต้น ขบวนการนี้จึงเหมาะกับเอกสารเก่า, เอกสารทางประวัติศาสตร์, เอกสารที่ไม่มีการตีพิมพ์แล้ว

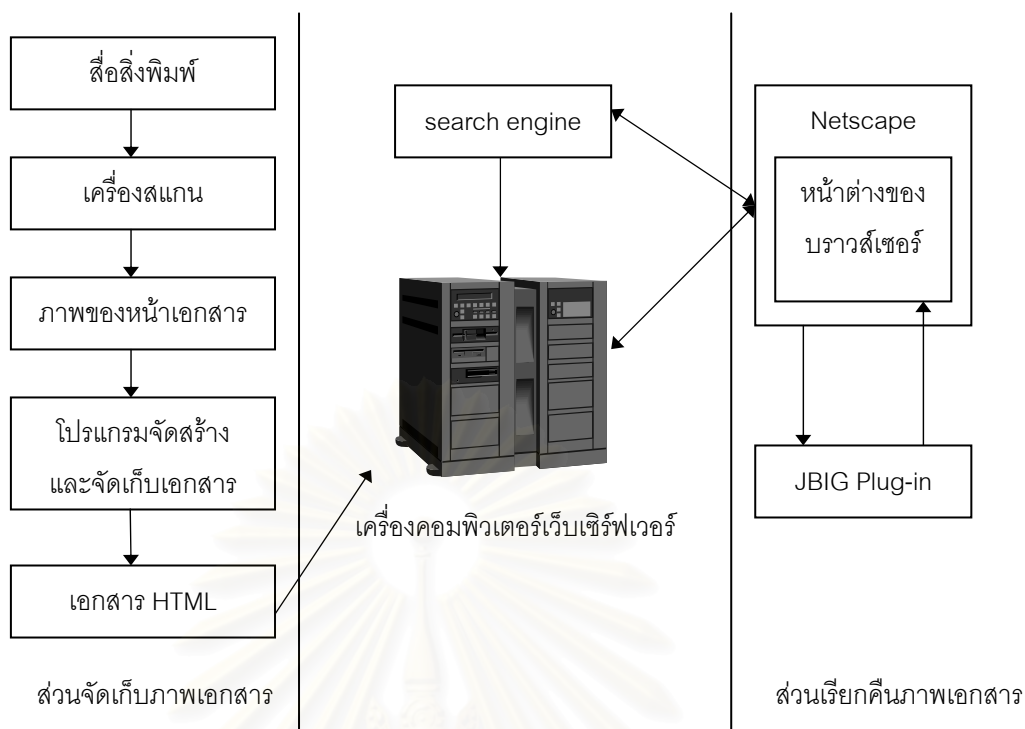
ภาพเอกสารสแกนที่ได้ในรูปแบบการบันทึกแฟ้มเป็นดิจิทัลแล้วจะมีขนาดของแฟ้มต่อหน้าที่ใหญ่กว่าไม่สามารถจัดเก็บได้โดยตรง มีข้อเสียคือเปลืองเนื้อที่ ๆ ใช้สำหรับจัดเก็บ ยิ่งไปกว่านั้นสำหรับการเผยแพร่ผ่านอินเทอร์เน็ต ผู้ใช้ต้องเสียเวลาในการดาวน์โหลดแต่ละหน้าของเอกสารเป็นจำนวนมาก ดังนั้นจึงต้องมีการจัดการทำให้ขนาดของแฟ้มข้อมูลเล็กลงเสียก่อนแล้วจึงเคลื่อนย้ายภาพเอกสารทั้งหมดไปเก็บไว้ที่เครื่องคอมพิวเตอร์เว็บเซิร์ฟเวอร์ทำให้ต้องมีระบบสำหรับการจัดเก็บภาพเอกสารเพิ่มเติมเข้ามาในขบวนการแปลงสิ่งพิมพ์ให้เป็นข้อมูลในรูปแบบดิจิทัล

แนวทางในการแก้ปัญหาที่นี้ได้มีงานวิจัยที่ทำการบีบอัดภาพเอกสารกับภาพสแกนก่อนที่จะบรรจุลงบนคอมพิวเตอร์เครือข่ายเพื่อให้ผู้ใช้งานเข้าถึงได้ผ่านเว็บ

2.5 งานวิจัยที่เกี่ยวข้อง

งานวิจัยของ ภาณุมาศ หาดทรายทอง [1] ได้ทำการบีบอัดภาพเอกสารด้วยระเบียบวิธีที่มีชื่อว่า JBIG1 โดยได้ทำการออกแบบระบบจัดเก็บและเรียกคืนภาพเอกสาร ดังแสดงในรูปที่ 2.6 ซึ่งสามารถสรุปผลการวิจัยได้ดังนี้

1. การเก็บเอกสารเป็นภาพทำให้รักษาข้อมูลเดิมของเอกสารได้
2. การนำเทคนิค JBIG มาเข้ารหัสภาพ ทำให้ใช้เนื้อที่ในการเก็บเอกสารลดลง และเนื่องจากปริมาณข้อมูลลดลงทำให้เวลาเรียกคืนภาพเอกสารผ่านอินเทอร์เน็ตจะใช้เวลาถ่ายโอนข้อมูลน้อยลง ประกอบกับภาพที่เข้ารหัสแบบ progressive ทางส่วนเรียกคืนจะรับข้อมูลภาพความละเอียดต่ำก่อน ซึ่งมีปริมาณข้อมูลน้อยจึงสามารถแสดงภาพได้อย่างรวดเร็ว ทำให้ผู้ใช้ดูเอกสารอย่างรวดเร็ว ได้ทันที โดยไม่ต้องรอให้ข้อมูลส่งมาจนครบ ภาพเอกสารที่นำมาจัดสร้างสามารถเป็นได้ทั้ง 2 ระดับ(ขาวดำ) หรือ 2-8 บิตระดับเกรย์ ขึ้นอยู่กับจุดประสงค์ของการใช้งาน
3. เก็บข้อมูลในส่วนคำสำคัญเพื่อใช้ในการสืบค้นเป็นแบบรูปภาพ เพื่อลดภาระของผู้จัดทำเอกสารในการป้อนข้อมูล
4. โครงสร้างของเอกสารเชื่อมโยงเข้าด้วยกันแบบไฮเปอร์ลิงค์ โดยใช้ภาษา HTML ซึ่งเป็นมาตรฐานบนอินเทอร์เน็ต
5. เอกสารชนิด HTML ที่ได้ทำการสร้างจะอยู่ในโลกคอลไดเรกทอรีก่อน เมื่อผู้จัดทำการแก้ไขเรียบร้อยแล้ว จึงถ่ายโอนข้อมูลไปที่เว็บเซิร์ฟเวอร์เพื่อให้ผู้ใช้เรียกคืนภาพเอกสารผ่านทางอินเทอร์เน็ตได้
6. ส่วนเรียกคืนภาพเอกสารจะกระทำผ่านอินเทอร์เน็ตโดยใช้โปรแกรม Netscape Navigator เป็นส่วนติดต่อกับผู้ใช้ โดยผู้ใช้สามารถสืบค้นเอกสารได้โดยใช้บริการของ search engine ที่ให้บริการในอินเทอร์เน็ต และมี JBIG plug-in มอดูลทำหน้าที่แสดงภาพเอกสาร และบรรทัดภาพเอกสาร JBIG เช่น ชุมภาพเพื่อแสดงที่ความละเอียดต่าง ๆ กัน, แสดงภาพแบบ thumbnail หรือดูหน้าเอกสารก่อนหน้าหรือถัดไป
7. การเรียกคืนภาพเอกสารมีลักษณะเป็น page-on-demand อย่างแท้จริง กล่าวคือ ผู้ใช้สามารถเรียกหน้าที่ต้องการดูได้โดยตรง หรือไปยังส่วนต่าง ๆ ของเอกสารได้ โดยไม่จำเป็นต้องอ่านหน้าทั้งหมดของเอกสาร



รูปที่ 2.6 ระบบจัดเก็บและเรียกคืนภาพเอกสารผ่านอินเทอร์เน็ตในงานวิจัยของภานุมาศ

ปัญหาของระบบการเข้าถึงและระบบการจัดเก็บเอกสารข้างต้น ได้แก่ สำหรับเอกสารที่เป็นหนังสือ 1 เล่ม จะต้องประกอบด้วย เอกสาร HTML 1 แฟ้ม สำหรับการจัดเก็บภาพเอกสารสแกน 1 ภาพเอกสาร ทำให้การจัดเก็บข้อมูลประกอบด้วยแฟ้มเอกสารเป็นจำนวนมาก ไม่สะดวกแก่การสำเนาและเคลื่อนย้าย ปัญหาอีกอย่างหนึ่งคือการใช้ระบบเอกสาร HTML เป็นไฟล์แสดงผลภาพทำให้ไม่สามารถใส่การเชื่อมโยง (Hyperlink) ภายในภาพเอกสารได้เนื่องจากข้อจำกัดของ HTML เอง นอกจากนี้การเข้าถึงแฟ้มเอกสารแต่ละภาพก็ทำได้โดยการเลือกจากการเลือกหน้าของผู้ใช้งานจากภาพเอกสารขนาดเล็ก (thumbnail) ทำให้มีแฟ้มเอกสารเพิ่มขึ้นไปอีก

การบีบอัดเอกสารภาพด้วยระเบียบวิธี JBIG1 เหมาะกับเอกสารภาพขาวดำ ที่มีแต่ตัวหนังสือเท่านั้น แม้ว่า JBIG1 สามารถบีบอัดภาพ Gray Scale ได้แต่สำหรับภาพเอกสารที่มีตัวหนังสือประกอบอยู่จะทำให้ยากต่อการอ่าน

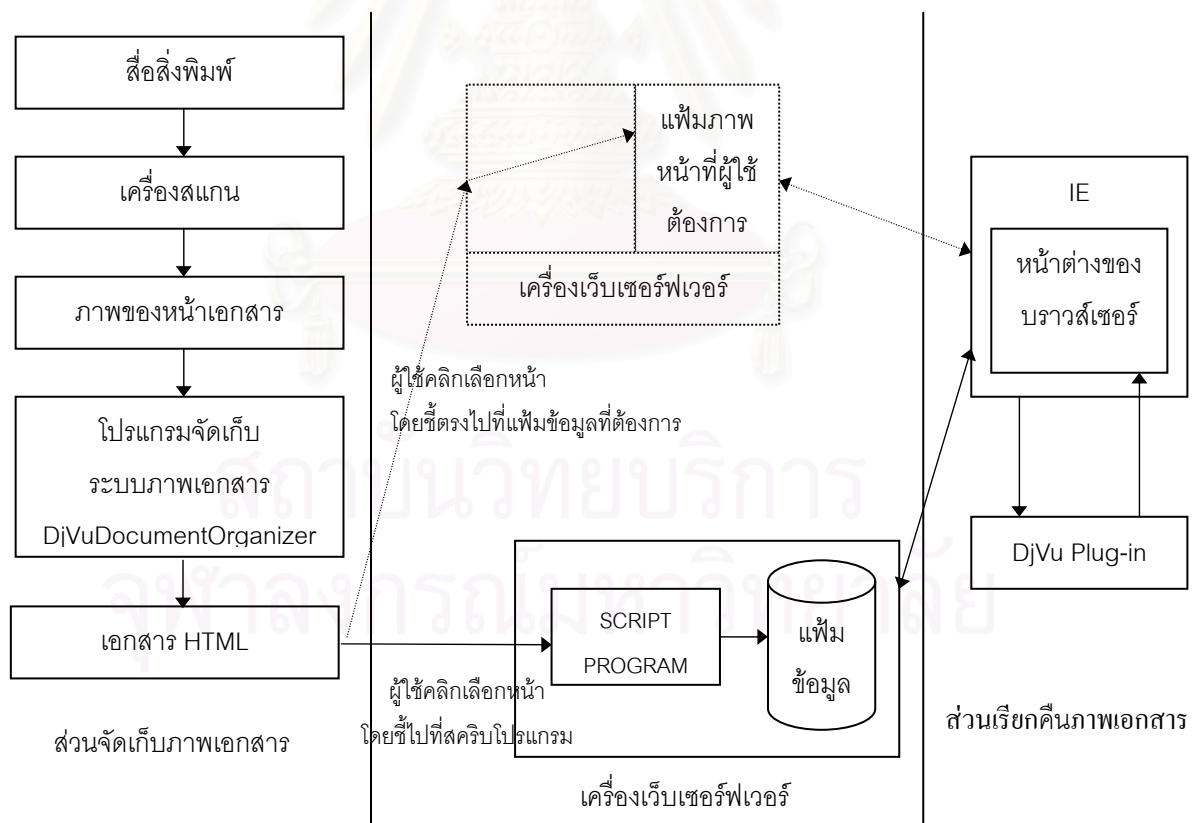
งานวิจัยของ สิทธิชัย อังกรวัฒน์ชัย [2] ได้นำเสนอการประยุกต์ใช้วิธีการบีบอัดที่มีชื่อว่า DjVu เพื่อใช้เป็นระเบียบวิธีในการบีบอัดภาพเอกสารในระบบการจัดเก็บและเรียกคืนเอกสารผ่านเว็บ ซึ่งสามารถทำการจัดเก็บเอกสารที่ประกอบด้วยตัวหนังสือและรูปภาพ ทั้งสีและขาวดำได้อย่างมีประสิทธิภาพทั้งในด้านคุณภาพของเอกสาร และขนาดของแฟ้มเอกสาร

ในวิทยานิพนธ์ของ สิทธิชัย อังกรวัฒน์ชัย ได้ทำการพัฒนาโปรแกรมประยุกต์บนระบบปฏิบัติการวินโดวส์ที่มีชื่อว่า DjVuDocumentOrganizer เพื่อให้ทำงานในส่วนการจัดเก็บ

ระบบเอกสารภาพ DjVu โปรแกรมนี้ทำงานบนเครื่องไมโครคอมพิวเตอร์ โดยได้พัฒนาจากเครื่องมือในการพัฒนาโปรแกรมที่มีชื่อว่า DjVu SDK ของบริษัท AT&T

โปรแกรม DjVuDocumentOrganizer มีความสามารถดังต่อไปนี้ คือ

1. สามารถเข้ารหัสและถอดรหัสแฟ้มให้เป็นรูปแบบ DjVu ได้
2. สามารถแสดงผลเอกสารภาพชนิด DjVu, BMP และ JPG บนจอภาพได้ รวมถึงสามารถย่อ-ขยายระหว่างการเรียกดูได้
3. สามารถสร้างความเชื่อมโยงระหว่างเอกสารภาพ DjVu ที่มาจากสิ่งพิมพ์เดียวกันเพื่อให้เป็นเอกสารเดียวกันแบบเรียงต่อเนื่องได้ขณะเรียกดูผ่านเว็บหรือแบบออฟไลน์
4. สามารถใส่การเชื่อมโยงกับชื่อเว็บอื่น (URL) ไว้บนเอกสารภาพ DjVu แบบผู้ใช้กรอก ซึ่งสามารถคลิกเพื่อเข้าถึงได้ขณะเรียกดูด้วยโปรแกรมบราวส์เซอร์
5. สามารถทำการรู้จำตัวเลขในหน้าสารบัญและหน้าดัชนีแบบมีความช่วยเหลือจากผู้สร้างเอกสาร แล้วใส่การเชื่อมโยงกับแฟ้มเอกสารภาพที่ตรงกับเลขหน้านั้นๆ ได้



รูปที่ 2.7 ระบบจัดเก็บและเรียกคืนภาพเอกสารผ่านอินเทอร์เน็ตในงานวิจัยของสิทธิชัย

สำหรับการเรียกดูจากผู้ใช้งานเว็บนั้นผู้ใช้งานไม่สามารถเรียกดูหน้าเอกสารผ่านเครือข่าย
ได้โดยตรงเพราะโปรแกรมที่เป็นเว็บเบราว์เซอร์ทั้งหลายไม่สามารถแสดงผลภาพชนิดนี้ได้ ดังนั้น
การเรียกดูจากผู้ใช้งานเว็บก็ต้องทำการดาวน์โหลดปลั๊กอินเพิ่มเติม(ที่มีขนาดประมาณ 2 MB)
แล้วติดตั้งลงบนเครื่องไมโครคอมพิวเตอร์ของผู้ใช้เพื่อให้โปรแกรมเบราว์เซอร์สามารถแสดงผล
ภาพชนิด DjVu ได้



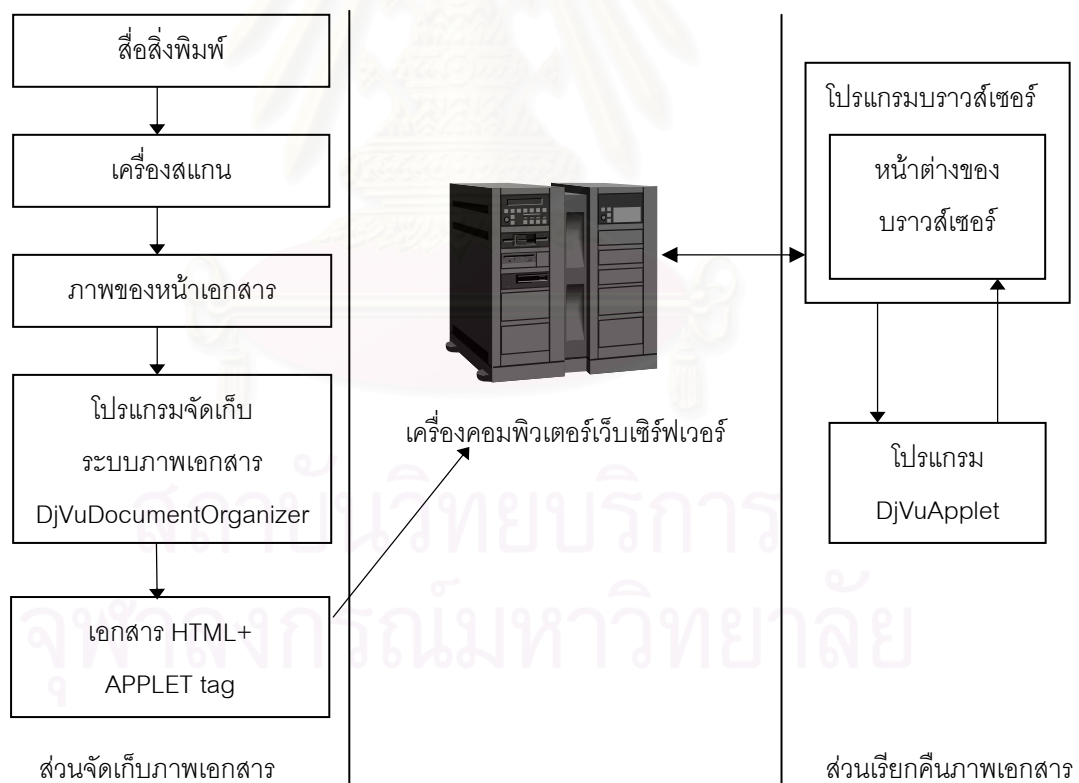
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 3

การออกแบบโปรแกรมส่วนเรียกคืนภาพเอกสาร DjVu ผ่านเว็บ

เนื้อหาในบทนี้กล่าวถึงแนวทางในการออกแบบโปรแกรมส่วนเรียกคืนภาพเอกสาร DjVu ผ่านเว็บ ในวิทยานิพนธ์นี้มีแนวคิดที่จะทำการพัฒนาโปรแกรมในส่วนการเรียกคืนภาพเอกสาร ด้วยโปรแกรมภาษาจาวา ที่เป็นแอปเพล็ต (Applet) ที่มีชื่อว่า DjVuApplet เพื่อให้ผู้ใช้สามารถที่จะเรียกดูภาพเอกสาร DjVu บนเว็บเบราว์เซอร์ (browser) ได้เลย โดยที่ไม่ต้องเสียเวลาในการดาวน์โหลดโปรแกรมปลั๊กอิน (ที่มีขนาดประมาณ 2 MB) มาติดตั้งเพิ่มเติม โดยได้ทำการแปลงไลบรารี (เวอร์ชัน 2) สำหรับการบีบอัดข้อมูลให้เป็นรูปแบบ DjVu ที่ทางบริษัท AT&T แจกสำหรับการพัฒนาที่ไม่เกี่ยวข้องกับการค้าจาก C++ ให้อยู่ในรูปของภาษาจาวา

3.1 ระบบจัดเก็บและเรียกคืนภาพเอกสารผ่านอินเทอร์เน็ต



รูปที่ 3.1 ระบบจัดเก็บและเรียกคืนภาพเอกสารผ่านอินเทอร์เน็ต

รูปที่ 3.1 แสดงระบบทั้งหมดซึ่งสามารถแบ่งได้เป็นสองส่วนใหญ่ ๆ คือ ส่วนจัดเก็บภาพเอกสารและส่วนเรียกคืนภาพเอกสาร

3.1.1 ส่วนจัดเก็บภาพเอกสาร

- ในวิทยานิพนธ์นี้ได้นำเอาโปรแกรม DjVuDocumentOrganizer [2] มาใช้ทำงานในส่วนการจัดเก็บระบบภาพเอกสาร DjVu
- จากนั้นก็นำเอาไฟล์ DjVuApplet.class (137 bytes) และ javadjvu.jar (192 KB) (ได้จากการ build, debug, และ test โปรแกรม DjVuApplet) ไปใส่ไว้ใน APPLET tag (APPLET tag คือข้อความที่อยู่ระหว่าง <applet> กับ </applet> ใช้สำหรับเรียก applet ให้ทำงานจากใน HTML page)
- จากนั้นนำผลลัพธ์ที่ได้จากโปรแกรม DjVuDocumentOrganizer และชุดของไฟล์เอกสารจากขั้นตอนที่สองไปเก็บไว้ในโหนดคอลโดเร็กทอรีก่อน เพื่อให้ผู้จัดทำได้ทดสอบดูความถูกต้อง หรือแก้ไขก่อนที่จะส่งไปเก็บไว้บนเครื่องคอมพิวเตอร์เว็บเซิร์ฟเวอร์

3.1.2 การออกแบบส่วนแสดงภาพเอกสาร

ส่วนเรียกคืนภาพเอกสาร ได้ออกแบบให้ใช้โปรแกรมเว็บเบราว์เซอร์ทำหน้าที่เป็นตัวแสดงภาพเอกสาร DjVu โดยที่ผู้ใช้ไม่ต้องดาวน์โหลดโปรแกรมปลั๊กอิน (ที่มีขนาดประมาณ 2 MB) มาติดตั้งเพิ่มเติม ผู้ใช้สามารถเรียกดูเอกสาร DjVu ได้เลยจากการทำงานของโปรแกรม DjVuApplet ซึ่งก็คือโปรแกรมที่ถูกสร้างขึ้นเพื่อส่งไปกับ HTML page ให้ไปทำงานภายใต้เบราว์เซอร์ ที่มี Java interpreter บนเครื่องที่รับ HTML page นั้นไปแสดงผล ที่จริงแล้ว HTML page นั้นจะถูกโหลดไปที่เครื่องของผู้ใช้ก่อน และเมื่อเบราว์เซอร์อ่านถึงตำแหน่งที่มีการเรียก applet นั้น จึงจะโหลดโปรแกรมของ applet ตามมา และส่งให้ Java interpreter ทำงาน หาก applet ทำการวาดสิ่งใดออกมา ก็จะปรากฏบนพื้นที่ของ page นั้น

ในวิทยานิพนธ์นี้ได้ออกแบบให้โปรแกรม DjVuApplet ต้องมีฟังก์ชันการทำงาน และคุณสมบัติดังต่อไปนี้

1. สามารถแสดงผลเอกสารภาพ DjVu ได้ ภายใต้การทำงานของโปรแกรม DjVuApplet
2. การถอดรหัสภาพจะทำขณะที่เบราว์เซอร์รับข้อมูลจากเครือข่าย และถ้าภาพได้รับการเข้ารหัสเป็นแบบ progressive ก็ยังสามารถแสดงภาพที่ความละเอียดต่ำก่อนแล้วจึงเพิ่มความละเอียดขึ้นเรื่อย ๆ ตามข้อมูลที่รับมา

3. มีฟังก์ชันในส่วนติดต่อกับผู้ใช้ (UI) ซึ่งจะช่วยให้การโต้ตอบกับผู้ใช้ทำได้ง่ายขึ้น
4. กรณียที่เป็นภาพของหน้าเอกสารต้องสามารถดูหน้าก่อนหรือถัดจากหน้าปัจจุบันที่แสดงอยู่ได้ และต้องสามารถแสดงหน้าที่อยู่ติดกันพร้อม ๆ กันได้
5. การเรียกคืนภาพเอกสารมีลักษณะเป็น page-on-demand กล่าวคือผู้ใช้สามารถเรียกหน้าที่ต้องการดูได้โดยตรงหรือไปยังส่วนต่าง ๆ ของเอกสารได้ โดยไม่จำเป็นต้องโหลดหาทุกหน้าในคราวเดียว

3.2 เหตุผลที่เลือกการพัฒนาส่วนเรียกคืนภาพเอกสาร DjVu ด้วยภาษาจาวา

เหตุผลที่เลือกการพัฒนาโปรแกรมด้วยภาษาจาวาเนื่องจาก [8]

1. ภาษาจาวาประสบความสำเร็จอย่างมาก ในการใช้เทคนิคของภาษาเชิงวัตถุ ช่วยให้สร้างโปรแกรมที่ยู่ยากเป็นไปได้อย่างขึ้น ดังจะเห็นว่าหากเป็นภาษาอื่น การสร้างโปรแกรมที่เกี่ยวกับ graphic user interfaces, multitasking, network ผู้โปรแกรมจะต้องมีความรู้ขั้นสูงจึงจะทำงานได้ หรือไม่ก็ต้องอาศัยโปรแกรมประเภท wizard อย่างใน Visual C++ ช่วยสร้างโปรแกรม แต่ภาษาจาวามีกลไกของการโปรแกรมเชิงวัตถุที่ส่งเสริมการนำโปรแกรมที่สร้างไว้แล้วมาใช้งานใหม่ (reuse) โดยสามารถเปลี่ยนแปลงหรือเพิ่มเติมโปรแกรมบางส่วน ให้เหมาะกับงานที่ต้องการได้ โดยไม่จำเป็นต้องทราบรายละเอียดของการสร้างโปรแกรมเดิมทั้งหมด จึงสามารถสร้างโปรแกรมสำหรับงานที่ยู่ยากขึ้นจากโปรแกรมที่มีผู้สร้างไว้แล้วได้โดยง่าย

2. โปรแกรมที่เขียนด้วยภาษาจาวาเป็นโปรแกรมที่ทำงานโดยไม่ขึ้นกับระบบปฏิบัติการจึงเป็นโปรแกรมที่เหมาะสมอย่างยิ่งในการนำมาใช้งานบนอินเทอร์เน็ต เนื่องจากอินเทอร์เน็ตเป็นระบบเครือข่ายที่ไม่สามารถระบุเจาะจงได้อย่างแน่ชัดว่าผู้ใช้งานจะใช้อินเทอร์เน็ตอยู่บนระบบปฏิบัติการใด เช่น ผู้ใช้ทั่วไปอาจใช้เครื่องคอมพิวเตอร์ส่วนบุคคลเชื่อมต่อกับอินเทอร์เน็ต แต่นักวิจัยบางคนอาจใช้เครื่อง UNIX ต่อกับอินเทอร์เน็ต เป็นต้น ดังนั้นจึงจะสะดวกกว่าถ้าเรามีโปรแกรมเพียงโปรแกรมเดียวที่ผู้ใช้อินเทอร์เน็ตทุกคนสามารถเรียกใช้ได้ และในปัจจุบันนี้เบราว์เซอร์มาตรฐานเช่น Internet Explorer ต่างก็มีความสามารถที่จะใช้งานกับโปรแกรมภาษาจาวาได้แล้วทั้งสิ้น

3. โปรแกรมจาวาทุกตัวจะต้องทำงานภายใต้ Java Virtual Machine (JVM) เสมอเมื่อเราคอมไพล์โปรแกรมเป็นไบต์โค้ด (ไฟล์ *.class) แล้ว JVM จะทำหน้าที่แปลงไบต์โค้ดเหล่านี้ไปเป็นภาษาเครื่องและทำงานตามคำสั่งนั้น ๆ ต่อไป ด้วยวิธีนี้เองโปรแกรมจาวาจึงสามารถทำงานได้บนทุกระบบปฏิบัติการ ขอเพียงแต่ให้มี JVM บนระบบปฏิบัติการที่เราต้องการนำโปรแกรมจาวาไปรันก็พอ

4. เนื่องจากโปรแกรมภาษาจาวามักจะถูกส่งผ่านระบบเครือข่ายไปทำงานบนเครื่องคอมพิวเตอร์ของผู้อื่น ดังนั้นภาษาจาวาต้องมีหลักประกันให้แก่ผู้รับโปรแกรมนั้นไปทำงานว่าจะไม่ก่อให้เกิดอันตรายต่อเครื่องหรือระบบของเขา จาวาจึงมีข้อกำหนดหลายอย่างเพื่อให้โปรแกรมไม่สามารถทำอันตราย หรือสิ่งที่ไม่สมควรทำต่อระบบที่รับโปรแกรมนั้นไปทำงาน คุณสมบัติของภาษาอย่างนี้เรียกว่า ปลอดภัย(security) อย่างไรก็ดี ไม่มีภาษาใดที่ปลอดภัยร้อยเปอร์เซ็นต์ ภาษาจาวาถูกจัดว่ามีความปลอดภัยในระดับสูงเท่านั้น เพราะถูกออกแบบมาเพื่อความปลอดภัยมากกว่าภาษาอื่น

3.3 แอปเพล็ต (Applet)

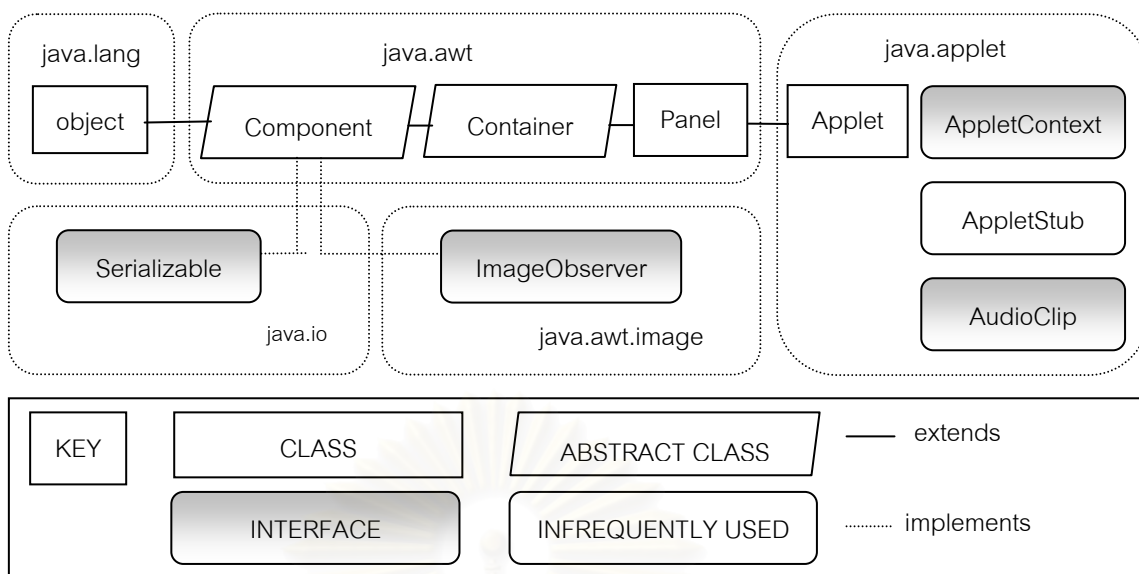
แอปเพล็ตคือโปรแกรมที่ถูกสร้างขึ้นเพื่อส่งไปกับ HTML page ให้ไปทำงานภายใต้บราวส์เซอร์ที่มี Java interpreter บนเครื่องที่รับ HTML page นั้นไป ที่จริงแล้ว HTML page นั้นจะถูกโหลดไปที่เครื่องของผู้ใช้ก่อน และเมื่อบราวส์เซอร์อ่านถึงตำแหน่งที่มีการเรียกแอปเพล็ตนั้น จึงจะโหลดโปรแกรมของแอปเพล็ตตามมาและสั่งให้ Java interpreter ทำงาน หากแอปเพล็ตทำการวาดสิ่งใดออกมา ก็จะปรากฏบนพื้นที่ของ page นั้น

โปรแกรมภาษาจาวาที่เป็น application จะทำงานภายใต้ Java interpreter ไม่จำเป็นต้องอาศัยโปรแกรมอื่น และสามารถควบคุมการดำเนินของตนเอง จึงเรียกว่า standalone โปรแกรม แต่โปรแกรมที่เป็นแอปเพล็ตจะต้องทำงานในสภาวะแวดล้อมที่กำหนดให้โดยบราวส์เซอร์และถูกบราวส์เซอร์ควบคุมการดำเนินของโปรแกรมให้ผ่านขั้นตอนต่าง ๆ เราจึงเรียกการทำงานของแอปเพล็ตว่าเป็นแบบ embedded เพราะต้องอาศัยบราวส์เซอร์ในตอนที่ทำงาน

โปรแกรมของแอปเพล็ตเมื่อถูกคอมไพล์แล้วจะได้ไฟล์เป็น .class แต่จะไม่สามารถทำงานได้โดย Java interpreter (java.exe) เหมือนกับคลาสของ application การเรียกให้แอปเพล็ตทำงานนั้นต้องทำจากภายใน HTML page โดยใช้ APPLET tag และแอปเพล็ตนั้นจะทำงานในสภาวะแวดล้อมของบราวส์เซอร์ที่มี Java interpreter อยู่ภายใน อย่างเช่น Microsoft Internet Explorer

3.3.1 คลาสแอปเพล็ต (Applet Class)

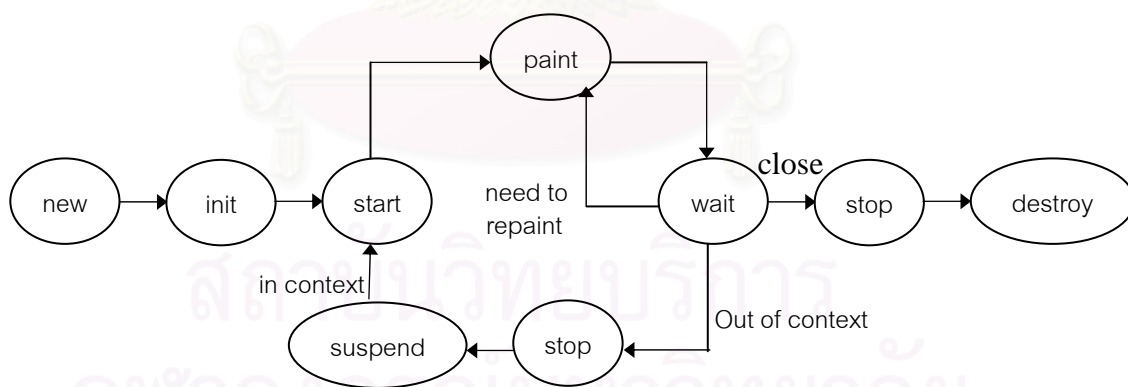
คลาสแอปเพล็ตถูกกำหนดไว้ใน package java.applet คลาสแอปเพล็ตขยายมาจากคลาส Panel ซึ่งขยายมาจากคลาส Container ซึ่งขยายมาจากคลาส Component ดังรูปที่ 3.2



รูปที่ 3.2 Applet Class

3.3.2 ช่วงชีวิตของแอปเพล็ต (Applet Life Cycle)

แอปเพล็ตจะมีช่วงชีวิตผ่านสถานะต่าง ๆ ดังนี้ คือ init, start, paint, wait, suspended, stop และ destroy ทุก ๆ แอปเพล็ตจะถูกบราวส์เซอร์ควบคุมให้ดำเนินการผ่านสถานะเหล่านี้ตามที่กำหนดไว้ สามารถแสดงเป็นรูปการเปลี่ยนสถานะได้ดังรูปที่ 3.3



รูปที่ 3.3 Applet Life Cycle

เมื่อเริ่มต้นแอปเพล็ตถูกสร้างขึ้นโดยการ new() เหมือน instances ทั่วไป และจะเริ่มทำงานทันทีคือ

1. เข้าสู่สถานะ init แล้วเข้าสู่สถานะ start แล้วเข้าสู่สถานะ paint และเข้าสู่สถานะ wait ซึ่งจะหยุดรอเหตุการณ์ (Event) จากผู้ใช้ (คืออินพุตที่รับเข้าไปทาง mouse, keyboard หรือ

- graphic user interface) ที่จะเข้ามาให้เกิดการทำงานของ method ที่กำหนดไว้สำหรับจัดการกับเหตุการณ์นั้น เมื่อเสร็จแล้วจะกลับเข้าสู่สถานะ wait เพื่อรอเหตุการณ์อื่นต่อไป
2. ระหว่างที่แอปพลิเคชันอยู่ในสถานะ wait หากผู้ใช้เปลี่ยนแปลง HTML page จนทำให้
 - 2.1 พื้นที่แสดงผลของแอปพลิเคชันนั้นต้องมีการวาดใหม่ เช่น เมื่อแอปพลิเคชันถูกเปลี่ยนขนาด (resize), ขยายขนาด (maximize) หรือมี windows อื่นมาทับแล้วนำออกไป แอปพลิเคชันนั้นจะถูกทำให้เกิดการวาดอีกครั้ง ส่งผลให้แอปพลิเคชันเข้าสู่สถานะ paint เมื่อเสร็จแล้วจะกลับเข้าสู่สถานะ wait รอเหตุการณ์อื่นต่อไป
 - 2.2 พื้นที่แสดงผลของแอปพลิเคชันไม่ปรากฏใน page ของบราวส์เซอร์ในตอนนั้น เช่น เมื่อผู้ใช้เลื่อน page ไปดูหน้าอื่น หรือกดปุ่ม minimize จะทำให้บราวส์เซอร์ออกคำสั่งให้แอปพลิเคชันเข้าสู่สถานะ stop และเข้าสู่สถานะ suspended ซึ่งแอปพลิเคชันจะหยุดรออยู่ในสถานะนี้โดยไม่ยอมรับอินพุตจากผู้ใช้ และไม่ได้ตอบโต้ ใดๆ ไปจนกว่าพื้นที่แสดงผลของแอปพลิเคชันนั้นจะถูกทำให้กลับมาปรากฏอีกครั้ง บราวส์เซอร์ก็จะออกคำสั่งให้แอปพลิเคชันเข้าสู่สถานะ start, paint และ wait เพื่อเริ่มทำงานอีก
 3. ระหว่างที่แอปพลิเคชันอยู่ในสถานะ wait หากผู้ใช้กดปุ่ม close ของบราวส์เซอร์เพื่อหยุดการทำงาน จะทำให้แอปพลิเคชันเข้าสู่สถานะ stop และไปสู่สถานะ destroy แล้วสิ้นสุดการทำงาน

ข้อสังเกตเกี่ยวกับช่วงชีวิตของแอปพลิเคชันพอสรุปได้ดังนี้

- เมื่อถูกสร้างขึ้น แอปพลิเคชันจะเข้าสู่สถานะ init, start และ paint
- ตอนถูกทำลาย (บราวส์เซอร์เลิกทำงาน) จะเข้าสู่สถานะ stop และ destroy
- แอปพลิเคชันจะผ่านสถานะ init และสถานะ destroy ได้ครั้งเดียว นั่นคือ ผ่าน init เมื่อตอนเริ่มต้นทำงานครั้งแรก และผ่าน destroy เมื่อถูกทำลาย ซึ่งจะกลับมาทำงานอีกไม่ได้
- แอปพลิเคชันจะผ่านสถานะ start หรือ stop ได้หลายครั้ง ขึ้นกับการควบคุมของผู้ใช้
- แอปพลิเคชันจะหยุดรอเหตุการณ์อยู่ในสถานะ wait หรือ suspended ได้เป็นเวลานาน ส่วนสถานะอื่น ๆ นั้น เมื่อเข้าไปทำงานเสร็จแล้วจะผ่านออกไปทันที

เราไม่สามารถเขียนโปรแกรมเพื่อเปลี่ยนแปลงให้แอปพลิเคชันมีการดำเนินผ่านสถานะในแบบที่แตกต่างไปจากนี้ได้ เนื่องจากบราวส์เซอร์เป็นผู้ควบคุมการดำเนินไปของแอปพลิเคชันผ่านสถานะต่าง ๆ อย่างไรก็ตาม เราสามารถกำหนดการทำงานของแอปพลิเคชันขณะที่ดำเนินผ่านสถานะหนึ่ง ๆ ได้

3.3.3 APPLETTAG

APPLET tag คือข้อความที่อยู่ระหว่าง <applet> กับ </applet> สำหรับใช้เรียกแอปพลิเคชันให้ทำงานจากใน HTML page

```
<applet
    code=applet_program
    width=pixels
    height=pixels
    [codebase=URL]
    [archives=jarFile]
    [name=applet_name]
    [align=alignment]
    [vspace=pixels]
    [hspace=pixels]
    [alt=alternateText]>
    [<param name=parameter_name value=parameter_value>] *
    [HTML text displayed in the absence of Java]
</applet>
```

สิ่งที่อยู่ระหว่าง [กับ] คือ option ซึ่งจะมีหรือไม่มีก็ได้ หากตามด้วย * จะหมายถึง มีซ้ำอีกกี่ครั้งก็ได้ ชื่อที่อยู่ทางด้านซ้ายของ = คือชื่อ attribute ของ APPLET tag และค่าที่อยู่ด้านขวาของ= คือค่าของ attribute นั้น จะเห็นว่าทุก ๆ APPLET tag จะต้องมี attributes อย่างน้อยคือ code, width และ height

- code ระบุโปรแกรมของ applet ที่จะถูกทำงาน ซึ่งเป็นไฟล์ .class หากเป็น applet ที่อยู่ในเว็บไซต์อื่นจะต้องอ้างถึงโดยใช้ URL ถ้าอ้างถึงแต่เพียงชื่อไฟล์จะเป็นชื่อที่ relative กับ URL ของเว็บไซต์ ที่มีไฟล์ HTML นั้นอยู่
 - width ระบุความยาวของพื้นที่ในแนวนอนที่จะแสดง applet ใน HTML page
 - height ระบุความกว้างของพื้นที่ในแนวตั้งที่จะแสดง applet ใน HTML page
- ส่วน attributes ของ APPLET tag ที่เป็น options มีรายละเอียดดังนี้
- codebase ระบุ base URL ของ applet class ที่จะถูกเรียกมาทำงาน ซึ่งก็คือ โดเมนทอริที่จะถูกค้นหา applet นั้น หากไม่ระบุ attribute นี้ base URL ของไฟล์ HTML จะถูกใช้เป็น base URL ของ applet

- archives ระบุชื่อของ jar ไฟล์ที่มี applet ที่จะถูกทำงาน ใช้ในกรณีที่มีการบีบอัดไฟล์ของ applet มาใน jar ไฟล์
- name ระบุชื่อของ applet มีประโยชน์ในกรณีที่บาง HTML page อาจมีหลาย ๆ applets ทำงานพร้อมกัน ซึ่งจะทำให้ applet หนึ่งสามารถอ้างถึงอีก applet หนึ่งได้ ในการที่ applet จะได้ชื่อของ applet หนึ่งจะต้องใช้ getApplet() ที่กำหนดไว้ใน AppletContext interface
- align ระบุการวาง (alignment) พื้นที่ของ applet ใน HTML page ซึ่งคล้ายกับการวางรูปภาพโดย IMG tag ของ HTML ค่าที่เป็นไปได้คือ LEFT, RIGHT, TOP, BOTTOM, MIDDLE, BASELINE, TEXTTOP, ABSMIDDLE และ ABSBOTTOM
- vspace และ hspace ระบุช่องว่างระหว่างพื้นที่ของ applet กับข้อความหรือรูปภาพใน HTML page นั้น โดยระบุเป็นจำนวน pixels ของระยะห่างในแนวตั้งและแนวนอนตามลำดับ ซึ่งคล้ายกับการกำหนดระยะห่างของ IMG tag เช่นกัน
- alt ระบุข้อความที่จะถูกแสดงใน HTML page ถ้าบราวส์เซอร์นั้นสามารถเข้าใจ APPLET tag แต่ไม่สามารถทำงานโปรแกรม applet นั้นได้ในขณะนั้น ซึ่งอาจเป็นเพราะขาดช่องที่ Java interpreter หรือไม่สามารถโหลด applet นั้นมาได้

เราสามารถส่งพารามิเตอร์จาก HTML page ไปสู่ applet ได้โดยใช้ PARAM tag ซึ่งจะต้องระบุอยู่ระหว่าง <applet> กับ </applet> มีรูปแบบดังนี้

```
<param name=parameter_name value=parameter_value>
```

หนึ่ง PARAM tag ใช้ส่งพารามิเตอร์ได้หนึ่งตัว แต่ applet หนึ่งอาจจะมีพารามิเตอร์ก็ได้ทุก ๆ พารามิเตอร์จะต้องมีทั้ง attribute เป็นชื่อ (name) และค่า (value) ในโปรแกรมของ applet เราย่นาค่าพารามิเตอร์ออกมาได้โดยใช้ method ของคลาส Applet คือ

```
Public String getParameter(String name);
```

โดยต้องระบุชื่อพารามิเตอร์นั้นให้เป็น String name เนื่องจากค่าของ value เป็น String และจะได้ค่าออกมาเป็น String เท่านั้น

3.4 การออกแบบโปรแกรม DjVuApplet

ในงานวิจัยนี้ได้ทำการออกแบบ packages และกลุ่ม class สำหรับการพัฒนาโปรแกรม DjVuApplet ดังต่อไปนี้

1. package com.kik.djvu

ภายใน package นี้ประกอบไปด้วยกลุ่ม class ที่ได้จากการแปลงไลบรารี (version 2) สำหรับการบีบอัดข้อมูลให้เป็นรูปแบบ DjVu ที่ทางบริษัท AT&T แจกสำหรับการพัฒนา จาก C++ ให้อยู่ในรูปของภาษาจาวา

1.1 package com.kik.djvu.anno

package นี้ใช้สำหรับการ decoding annotations

1.2 package com.kik.djvu.outline

package นี้ใช้สำหรับการ decoding bookmarks

1.3 package com.kik.djvu.text

package นี้ใช้สำหรับการ decoding และการค้นหา hidden text

2. package com.kik.djvubean

ภายใน package นี้ประกอบไปด้วยกลุ่ม class ที่ใช้สำหรับการสร้างส่วนติดต่อกับผู้ใช้ (Graphic User Interface)

2.1 package com.kik.djvubean.menu

package นี้ใช้สำหรับการสร้าง pop-up เมนู

2.2 package com.kik.djvubean.outline

package นี้ใช้สำหรับการสร้าง navigation pane

2.3 package com.kik.djvubean.text

package นี้ใช้สำหรับการแสดงและการค้นหาอักษร (searching text)

2.4 package com.kik.djvubean.toolbar

package นี้ใช้สำหรับการสร้าง toolbar navigation

3. package com.kik.djview

ภายใน package นี้ประกอบไปด้วยกลุ่ม class ที่ใช้สำหรับแสดงผลเอกสารภาพ DjVu ในรูปแบบของแอปเพล็ต

3.1 package com.kik.frame

package นี้ถูกใช้ในขั้นตอนการทดสอบและดีบั๊กโปรแกรม ซึ่งผลลัพธ์ที่ได้จะอยู่ในรูปของ stand-alone browser

สามารถสรุปองค์ประกอบของโปรแกรม DjVuApplet ได้ดังนี้

- applet : มีขนาดประมาณ 137 ไบต์

ใช้สำหรับสร้างไฟล์ DjVuApplet.class เพื่อนำไปใส่ไว้ใน APPLET tag ของเอกสาร HTML

- menu : มีขนาดประมาณ 5452 ไบต์
ใช้สำหรับเพิ่ม pop-up เมนูให้กับ applet เมื่อผู้ใช้คลิกที่ปุ่มเมาส์ข้างขวา หรือกดปุ่มเมาส์พร้อมกับ CTRL key pop-up ก็จะมีปรากฏออกมา
- toolbar : มีขนาดประมาณ 34079 ไบต์
ใช้สำหรับสร้างทูลบาร์ที่ด้านบนของแต่ละหน้าเอกสาร
- outline : มีขนาดประมาณ 8502 ไบต์
ใช้สำหรับจัดการฟังก์ชัน outline navigation ของ pop-up เมนู ในกรณีที่เอกสาร DjVu มีหลายหน้า
- frame : มีขนาดประมาณ 4065 ไบต์
ใช้ในขั้นตอนการทดสอบและดีบักโปรแกรม ซึ่งผลลัพธ์ที่ได้จะอยู่ในรูปของ stand-alone browser

3.5 คุณสมบัติของโปรแกรม DjVuApplet

โปรแกรม DjVuApplet มีคุณสมบัติดังรายละเอียดตามตารางที่ 3.1 เพื่อให้ผู้ใช้สามารถควบคุมลักษณะการทำงานของ applet ได้ โดยที่ค่าเหล่านี้จะถูกกำหนดผ่านทางพารามิเตอร์ <APPLET>, <OBJECT> หรือ <EMBED> tags

ตารางที่ 3.1 คุณสมบัติของโปรแกรม DjVuApplet

| Name, Values | คำอธิบาย |
|-----------------|--|
| AboutURL, <url> | สำหรับชี้ไปที่ URL ที่เป็นข้อมูลเกี่ยวกับ applet นี้ ตัวอย่าง: <param name= "abouturl" value="http://www.chula.ac.th"/> |
| Cache, [yes no] | สำหรับ enables หรือ disables ในส่วนของ predecoding และ caching ตัวอย่าง: <param name= "cache" value="no"/> <param name= "cache" value="yes"/> |
| Data, <url> | สำหรับระบุตำแหน่งและการเข้าถึงเอกสาร DjVu <param name= "data" value="../sample/book.djvu"/> |

ตารางที่ 3.1 (ต่อ) คุณสมบัติของโปรแกรม DjVuApplet

| Name, Values | คำอธิบาย |
|-------------------------------|--|
| Menu,[yes no] | สำหรับ enables หรือ disables ในส่วนของ pop-up เมนู ตัวอย่าง: <param name= "menu" value="no"/> <param name= "menu" value="yes"/> |
| Toolbar,[yes no] | สำหรับ enables หรือ disables ในส่วนของทูลบาร์ ตัวอย่าง: <param name= "toolbar" value="no"/> <param name= "toolbar" value="yes"/> |
| NavPane,[outline none] | สำหรับ enables หรือ disables ในส่วนของ navigation pane ตัวอย่าง: <param name= "navpane" value="no"/> <param name= "navpane" value="yes"/> |
| Zoom [<number> width page] | สำหรับระบุค่าเป็นเปอร์เซ็นต์ ที่ใช้ในการซูมหน้าเอกสาร ตัวอย่าง: <param name= "zoom" value="300%"/> <param name= "zoom" value="width"/> |
| MouseMode [pan zoom text] | สำหรับกำหนดพฤติกรรมของเมาส์ : panning, zoom ตัวอย่าง: <param name= "mousemode" value="pan"/> <param name= "mousemode" value="zoom"/> |
| Page,[<number>] | สำหรับกำหนดหน้าที่ต้องการแสดง <param name= "page" value="5"/> |
| LogoURL,<url> | ใช้สำหรับกำหนดตำแหน่งของ url เมื่อผู้ใช้คลิกไปที่โลโก้ <param name= "logourl" value="http://www.chula.ac.th"/> |

3.6 การค้นหาข้อความในเอกสารภาพ DjVu

ในงานวิจัยนี้ได้นำเอาไลบรารีที่เป็น open source ซึ่งพัฒนาขึ้นด้วยภาษาจาวามาใช้ในส่วนของการค้นหาข้อความเพื่อเพิ่มเติมความสามารถให้กับโปรแกรม DjVuApplet โดยทำการสร้าง package ดังต่อไปนี้

- package com.kik.djvu.text ภายใน package นี้จะใช้สำหรับการ decoding และการค้นหา hidden text
- package com.kik.djvubean.text ซึ่งภายใน package นี้จะใช้สำหรับการแสดงและค้นหาข้อความภายในเอกสาร

ส่วนจัดเก็บเอกสารภาพ

นำเอกสารภาพ DjVu หนึ่งฉบับที่ได้มีการสแกนไว้แล้วอัปโหลดไฟล์ไปที่ <http://any2djvu.djvuzone.org/> เพื่อทำไอซีอาร์ให้กับเอกสารภาพ DjVu

ส่วนเรียกดูเอกสารภาพ

เมื่อผู้ใช้ทำการค้นหาข้อความในเอกสาร โดยการใส่คำที่ต้องการค้นหา โปรแกรม DjVuApplet จะทำการตอบสนองกับผู้ใช้ให้ทราบว่าพบข้อความหรือไม่ ในกรณีที่พบข้อความที่ต้องการ โปรแกรม DjVuApplet จะแสดงข้อความที่ค้นพบ โดยการแสดงสีที่ทับตำแหน่งข้อความที่ค้นพบ โดยวิธีการค้นหาข้อความนั้นจะทำการค้นหาได้เพียงแค่หน้าปัจจุบันที่ทำการแสดงอยู่เท่านั้น ไม่สามารถค้นหาข้อความได้ทั้งเล่มเอกสาร

บทที่ 4

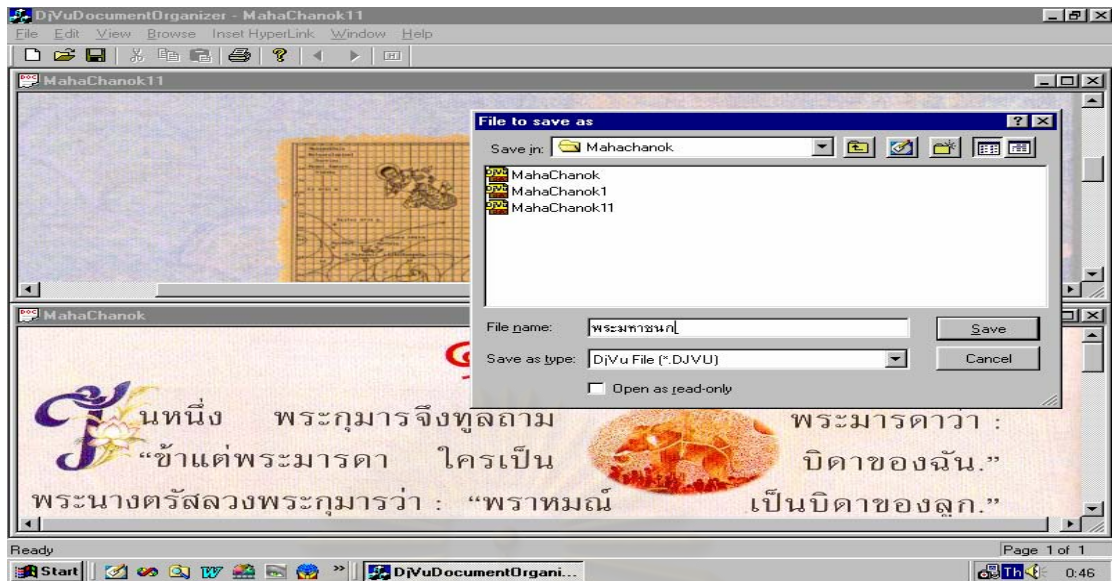
ผลการดำเนินการ

เนื้อหาในบทนี้จะนำเสนองานวิจัยที่ได้ทำ เพื่อให้การเผยแพร่สิ่งพิมพ์ผ่านเว็บเป็นไปโดยสะดวก งานวิจัยนี้ได้พัฒนาโปรแกรมจาวาแอปเพล็ต ที่มีชื่อว่า DjVuApplet (มีขนาดประมาณ 193 กิโลไบต์หลังจากทำเป็นจาร์ไฟล์แล้ว) เพื่อให้ทำงานในส่วนการเรียกคืนเอกสารภาพ DjVu ผ่านโปรแกรมประเภทบราวส์เซอร์ ซึ่งผู้ใช้สามารถที่จะเรียกดูเอกสารภาพชนิด DjVu ได้เลย โดยที่ไม่ต้องเสียเวลาในการดาวน์โหลดโปรแกรมปลั๊กอิน (ที่มีขนาดประมาณ 2 MB) มาติดตั้งเพิ่มเติม ซึ่งเป็นวิธีหนึ่งที่จะช่วยป้องกันความเสียหายที่จะเกิดขึ้นกับระบบของไคลเอนท์(client) ได้ และยังทำให้เกิดความสะดวกกับผู้ใช้งานอีกด้วย

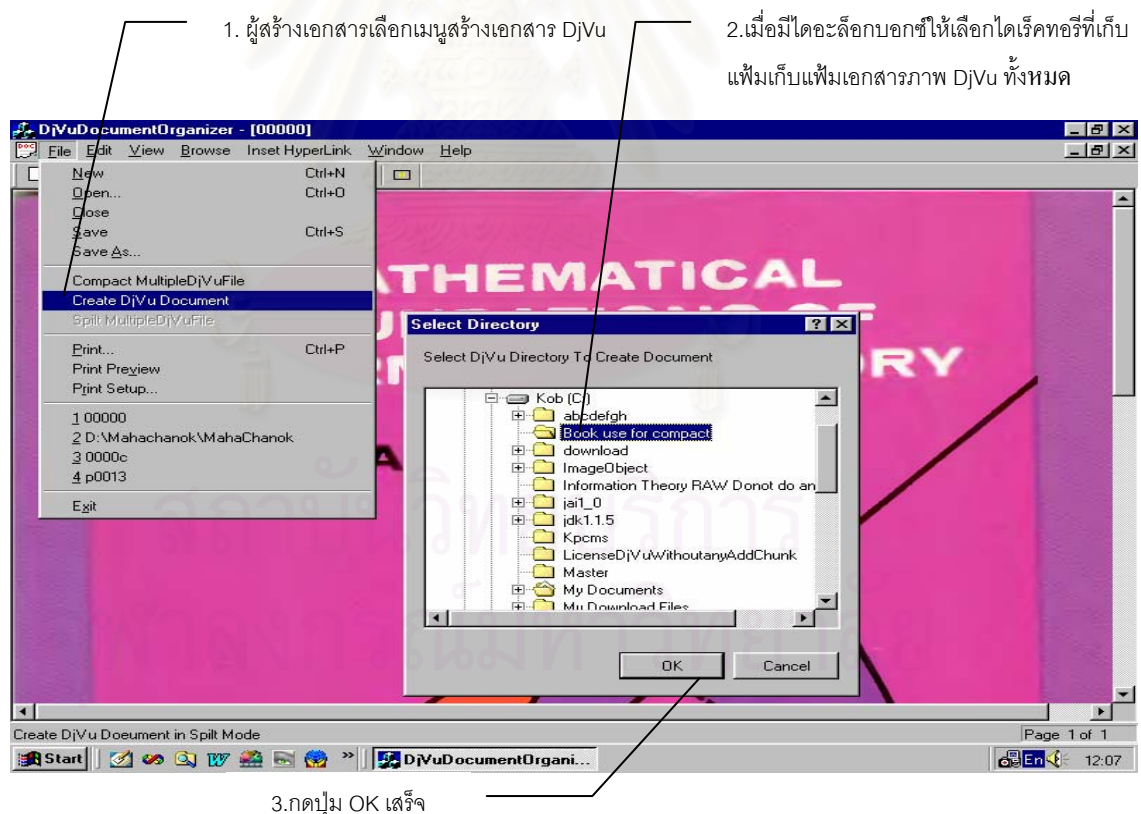
4.1 การแปลงภาพเอกสารจากแฟ้มภาพสแกนให้เป็น DjVu

เนื่องจากผู้วิจัยได้เลือกรูปแบบในการบีบอัดแฟ้มข้อมูลที่มีชื่อว่า DjVu ดังนั้นในขั้นตอนแรกของระบบเอกสารภาพ คือการสแกนภาพเอกสารให้อยู่ในรูปแบบมาตรฐานซึ่งอาจจะเป็น BMP หรือ JPG ก็ได้ แล้วใช้โปรแกรม DjVuDocumentOrganizer ที่พัฒนาขึ้นโดย สิทธิชัย [2] มาทำการเข้ารหัสข้อมูลที่เป็นสี RGB ซึ่งเก็บอยู่ในหน่วยความจำให้เป็นรูปแบบ DjVu แล้วทำการบันทึกลงฮาร์ดดิสก์ตามต้องการ โดยการเลือกเปิดภาพเอกสารสแกนที่บันทึกในรูปแบบ BMP หรือ JPG ก็ได้แล้วบันทึกให้เป็น DjVu โดยการเลือกเมนู File->SaveAs ดังแสดงในภาพที่ 4.1 ซึ่งโปรแกรมจะขึ้นไดอะล็อกบ็อกซ์เพื่อถามความละเอียดของภาพต้นฉบับ เพราะว่าไลบรารีการเข้ารหัส DjVu จะทำงานได้ประสิทธิภาพดีที่สุดที่ 300 จุดต่อนิ้ว

จากนั้นทำการสร้างความเชื่อมโยงระหว่างเอกสารภาพ DjVu ที่มาจากสิ่งพิมพ์เดียวกัน ซึ่งจะทำให้ผู้ใช้สามารถเรียกดูเอกสารแบบเรียงหน้าต่อเนื่องกันไปได้โดยไม่ขาดความต่อเนื่อง คล้ายๆ เป็นการรวมเล่มเอกสารทุกแฟ้มไว้ด้วยกัน มิใช่เก็บเป็นหน้าๆ แล้วมาสร้างความเชื่อมโยงจากภายนอก โดยการเลือกเมนู File->Create DjVu Document ซึ่งโปรแกรมจะเรียกไดอะล็อกบ็อกซ์เพื่อให้ผู้สร้างเอกสารเลือกไดเรกทอรีที่บรรจุแฟ้มที่ต้องการสร้างความเชื่อมโยง ดังรูปที่ 4.2



รูปที่ 4.1 การบันทึกเอกสารภาพให้อยู่ในรูปแบบ DjVu



รูปที่ 4.2 การสร้างความเชื่อมโยงระหว่างแฟ้มเอกสารภาพที่มาจากสิ่งพิมพ์เดียวกัน

4.2 การเรียกใช้งานไลบรารีของโปรแกรม DjVuApplet

ในวิทยานิพนธ์นี้ได้เลือกใช้โปรแกรม NetBeans เป็น IDE สำหรับการพัฒนาโปรแกรม DjVuApplet เมื่อเราได้ทำการคอมไพล์และรันโปรแกรม DjVuApplet แล้ว NetBeans project ก็ จะทำการสร้างไฟล์มาให้สองไฟล์ คือ

1. DjVuApplet.class (มีขนาดประมาณ 137 bytes)

ใช้สำหรับแสดงผลเอกสารภาพ DjVu ในรูปแบบของแอปเพล็ต

2. javadjvu.jar (มีขนาดประมาณ 192 KB)

ประกอบไปด้วยกลุ่มคลาสสำหรับการบีบอัดข้อมูลให้เป็นรูปแบบ DjVu และกลุ่มคลาสที่ใช้สำหรับการสร้างส่วนติดต่อกับผู้ใช้ (User Interface)

ปัญหาในการส่ง applet จำนวนมากไปทำงานที่เบราว์เซอร์ของผู้ใช้คือ การส่งไฟล์ไปในเครือข่ายครั้งละไฟล์จะเสียเวลามากกว่าการส่งไฟล์ใหญ่หนึ่งไฟล์ที่เป็นผลรวมของไฟล์ย่อยทั้งหมด และหากเราใช้การบีบอัดเข้าช่วยด้วย การบีบอัดไฟล์ใหญ่ ๆ จะได้ประสิทธิภาพดีกว่าการบีบอัดไฟล์เล็ก ๆ ดังนั้นเพื่อให้การส่ง applet ผ่านเครือข่ายไปได้เร็วขึ้น เราจึงควรรนำ applets เหล่านั้นบีบอัดลงเป็น jar ไฟล์เพียงไฟล์เดียว เมื่อส่ง jar ไฟล์นี้ไปถึงเบราว์เซอร์แล้ว ก็ใช้ option archives ของ APPLET tag เลือกทำงาน applet ตัวที่ต้องการใน jar ไฟล์นั้นได้

จากนั้นก็นำเอาไฟล์ DjVuApplet.class (137 ไบต์) และ javadjvu.jar (192 กิโลไบต์) ไปใส่ไว้ใน APPLET tag ที่อยู่ใน HTML page ดังนี้

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD>
<TITLE> Sucheraporn </TITLE>
<META NAME="Generator" CONTENT="EditPlus">
<META NAME="Author" CONTENT="">
<META NAME="Keywords" CONTENT="">
<META NAME="Description" CONTENT="">
</HEAD>
```

```

<BODY>
<applet
  style="border:0px none;margin:0px 0px 0px 0px;width:100%;height:100%;"
  code="DjVuApplet.class"
  archive="javadjvu.jar"
  width="720"
  height="480" >

<param name="cache_archive" value="javadjvu.jar" >
<param name="data" value="index.djvu">
<param name="abouturl" value="http://www.ee.eng.chula.ac.th/">
<param name="helpurl" value="http://www.ee.eng.chula.ac.th/">

</applet>
</BODY>
</HTML>

```

ตารางที่ 4.1 อารักิวเมนต์ที่จำเป็นต้องมีใน APPLET tag

| อารักิวเมนต์ที่จำเป็นต้องมีภายใน APPLET TAG | ความหมาย |
|---|---|
| code="DjVuApplet.class" archive="javadjvu.jar" width = "xxx" height = "xxx" <param name="data" value="index.djvu"> | ระบุโปรแกรมของ applet ที่จะถูกทำงาน ระบุชื่อของ jar ไฟล์ที่มี applet ที่จะถูกทำงาน กำหนดความกว้างของหน้าต่าง (หน่วยเป็น จุดภาพ) กำหนดความสูงของหน้าต่าง (หน่วยเป็นจุดภาพ) ตำแหน่งของไฟล์ภาพ DjVu |

- code ใช้สำหรับระบุโปรแกรมของ applet ที่จะถูกทำงาน ซึ่งเป็นไฟล์ .class หากเป็น applet ที่อยู่ในเว็บไซต์ (Web site) อื่นจะต้องอ้างถึงโดยใช้ URL ถ้าอ้างถึงแต่เพียงชื่อไฟล์จะเป็นชื่อที่สัมพันธ์กับ URL ของเว็บไซต์ ที่มีไฟล์ HTML นั้นอยู่
- Archives ใช้สำหรับระบุชื่อของ jar ไฟล์ที่มี applet ที่จะถูกทำงาน ใช้ในกรณีที่มีการบีบอัดไฟล์ของ applet มาใน jar ไฟล์
- `<param name="data" value="index.djvu">` เป็นการส่งพารามิเตอร์จาก HTML page ไปสู่ applet โดยใช้ PARAM tag ซึ่งจะต้องระบุอยู่ระหว่าง `<applet>` กับ `</applet>` ในที่นี้หมายถึงให้ applet แสดงภาพเอกสารที่มีชื่อว่า "index.djvu"

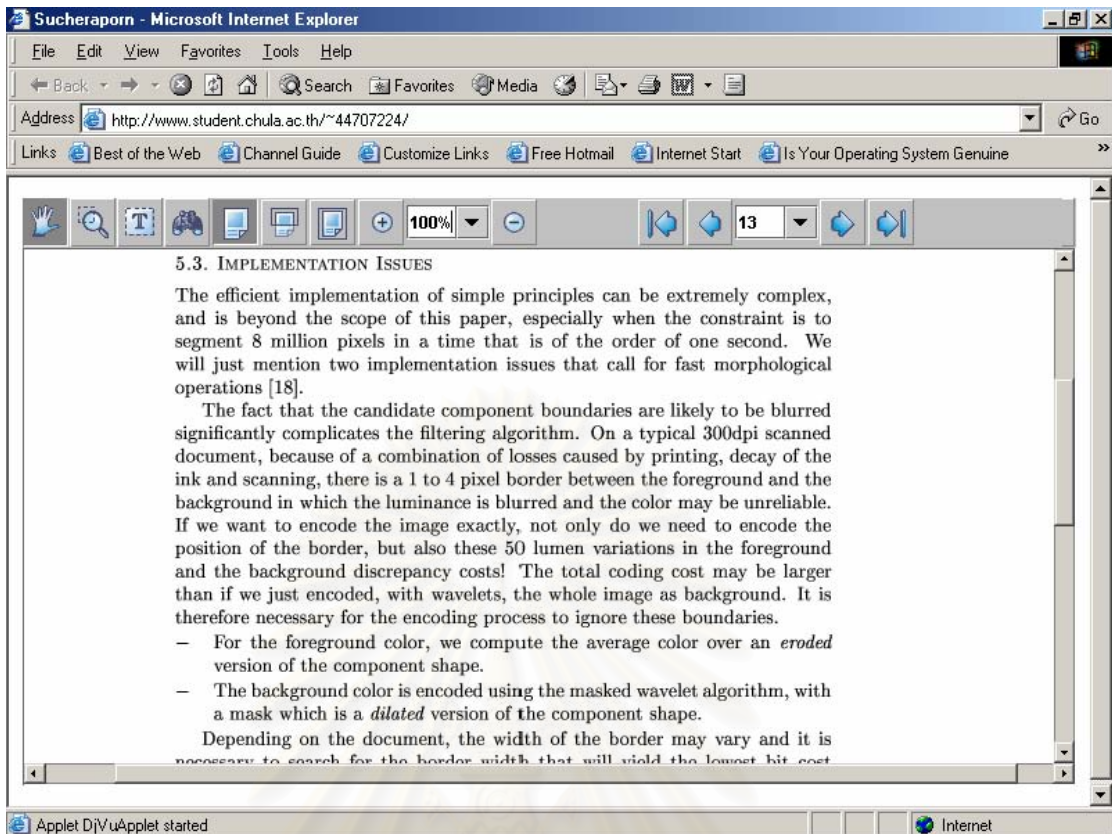
4.3 ส่วนเรียกคืนภาพเอกสาร

ส่วนเรียกคืนภาพเอกสารจะใช้โปรแกรมอินเทอร์เน็ตเอ็กพลอร์เรอร์เพื่อแสดงภาพเอกสาร HTML และเป็นส่วนติดต่อกับผู้ใช้

ฟังก์ชันการทำงานและคุณสมบัติที่สำคัญของส่วนเรียกคืนภาพเอกสาร ได้แก่

1. การบราวซ์เอกสาร เมื่อได้เอกสารที่ต้องการแล้วผู้ใช้สามารถอ่านเนื้อหาภายในเอกสารได้โดยอาศัยฟังก์ชันในการบราวซ์ เช่น
 - สามารถเรียกดูหน้าถัดไป หน้าก่อนหน้าได้ด้วยการคลิกที่ปุ่มกด
 - ซুমเข้าเพื่อดูรายละเอียดเฉพาะส่วน หรือซুমออกเพื่อดูรายละเอียดทั้งหน้าของเอกสาร
2. การแสดงผลบนหน้าจอ จะมีการดาวน์โหลดข้อมูลหน้าเฉพาะที่ผู้ใช้เรียกดูเท่านั้นกลับมาแสดงผล ไม่มีการโหลดภาพเอกสารทั้งฉบับ ซึ่งมีขนาดใหญ่ และทำให้เสียเวลาในการดาวน์โหลด
3. การถอดรหัสภาพจะทำขณะที่โปรแกรมบราวส์เซอร์รับข้อมูลจากเครือข่าย เนื่องจาก DjVu มีคุณสมบัติเป็น progressive จึงกำหนดขนาดภาพสูงสุดที่ต้องการแสดงได้ และขณะส่งข้อมูลภาพมาให้บราวส์เซอร์จะทำการแสดงภาพที่ความละเอียดต่ำกว่าก่อน แล้วจึงเพิ่มความละเอียดขึ้นเรื่อย ๆ ตามข้อมูลที่ได้รับมา

ตัวอย่างการเรียกดูภาพเอกสาร DjVu ผ่านเว็บ ดังแสดงในรูปที่ 4.3



รูปที่ 4.3 การเรียกดูเอกสารภาพผ่านเว็บ

4.3.1 ฟังก์ชันการทำงานในส่วนติดต่อกับผู้ใช้ (UI)

เมื่อภาพเอกสาร DjVu ถูกโหลดเข้าไปในเบราว์เซอร์ ก็จะมีเมนูทูลบาร์ปรากฏดังรูปที่ 4.4







รูปที่ 4.4 เมนูทูลบาร์ของโปรแกรม DjVuApplet

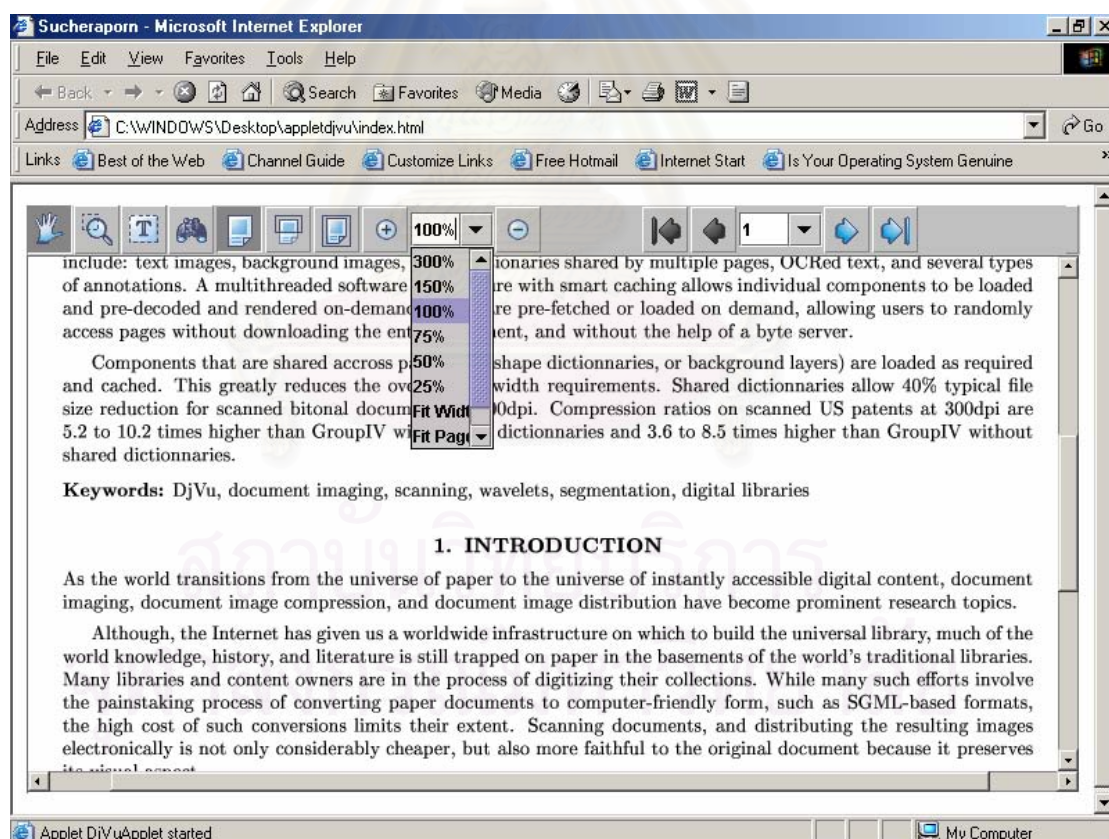
ฟังก์ชันการทำงานของทูลบาร์ของโปรแกรม DjVuApplet สามารถสรุปได้ดังตารางที่ 4.2

ดังนี้

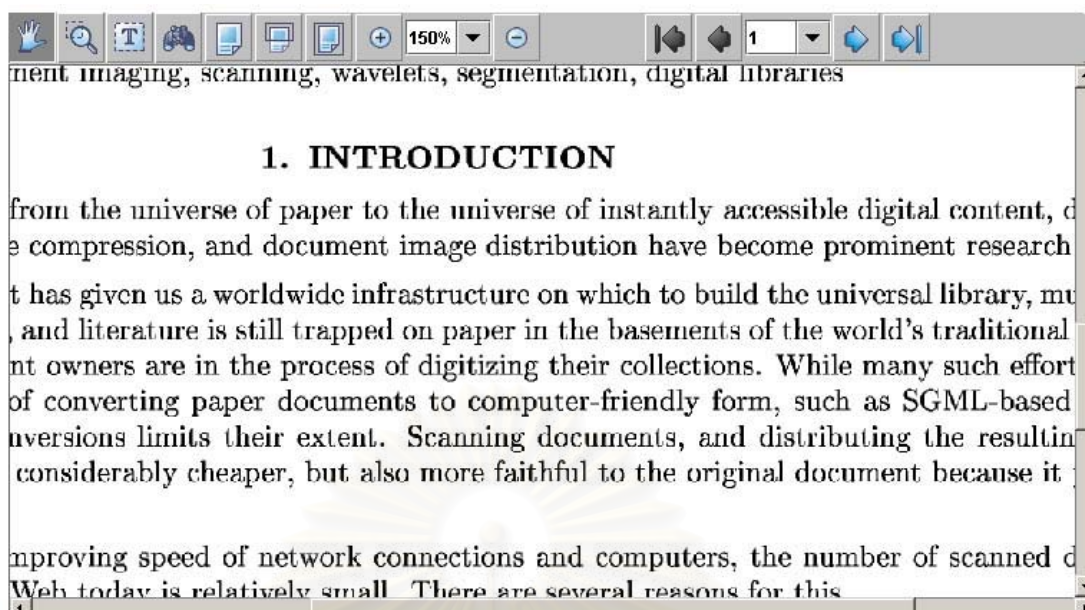
ตารางที่ 4.2 ฟังก์ชันการทำงานของทูลบาร์

| คำสั่ง | การทำงาน |
|---|---|
|  | ใช้สำหรับเลื่อนหน้าเอกสาร |
|  | ใช้สำหรับการค้นหาคำภายในเอกสาร |
|  | ใช้สำหรับแสดงภาพเอกสารที่ความละเอียดต่าง ๆ |
|  | ใช้สำหรับเรียกดูหน้าถัดไป หน้าก่อนหน้าของเอกสาร |

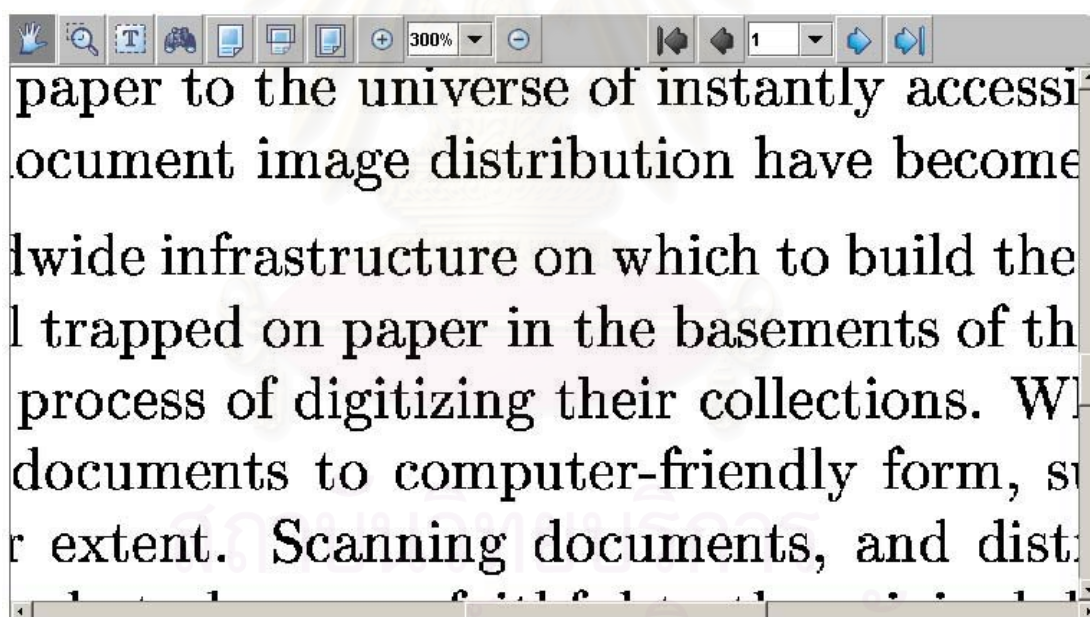
เนื่องจากภาพเอกสาร DjVu ได้รับการเข้ารหัสแบบ progressive ดังนั้นผู้ใช้จึงสามารถชมภาพเพื่อดูที่ความละเอียดต่าง ๆ ที่ต้องการได้ซึ่งจะเป็นระดับที่ใช้แสดงผลไปตลอดทุกหน้าของเอกสาร ดังแสดงในรูปที่ 4.5, รูปที่ 4.6 และรูปที่ 4.7



รูปที่ 4.5 การเรียกดูแฟ้มเอกสารภาพ DjVu ที่ระดับความละเอียดเท่ากับ 100%

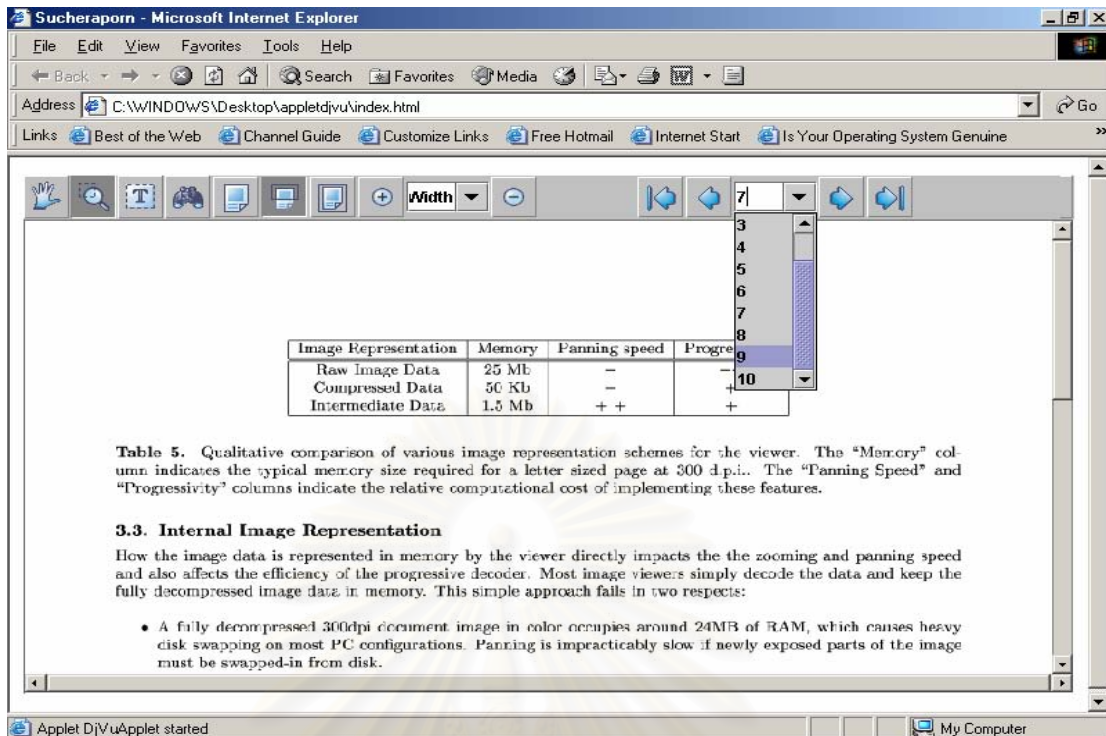


รูปที่ 4.6 การเรียกดูแฟ้มเอกสารภาพ DjVu ที่ระดับความละเอียดเท่ากับ150%

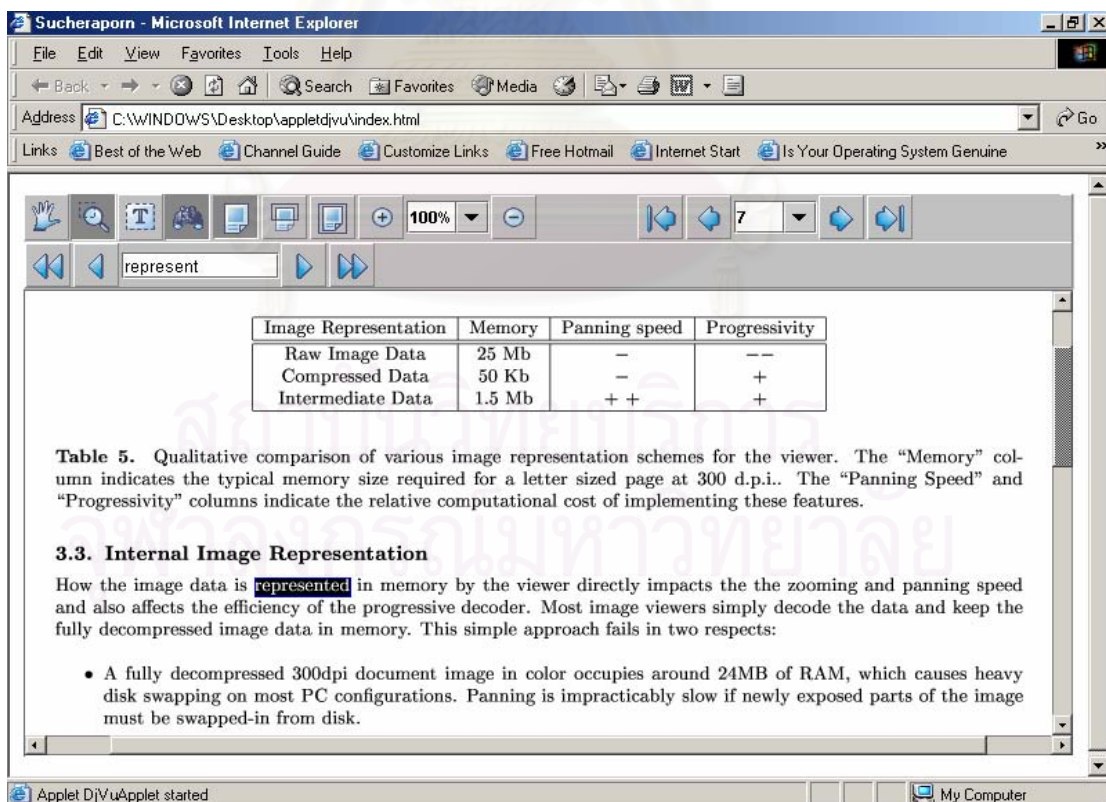


รูปที่ 4.7 การเรียกดูแฟ้มเอกสารภาพ DjVu ที่ระดับความละเอียดเท่ากับ300%


เมื่อได้เอกสารที่ต้องการแล้วผู้ใช้สามารถอ่านเนื้อหาภายในเอกสารได้ โดยอาศัยฟังก์ชันสำหรับเรียกดูหน้าถัดไป หน้าก่อนหน้าได้ด้วยวิธีการคลิกที่ปุ่มกดบนทูลบาร์ ดังแสดงในรูปที่ 4.8



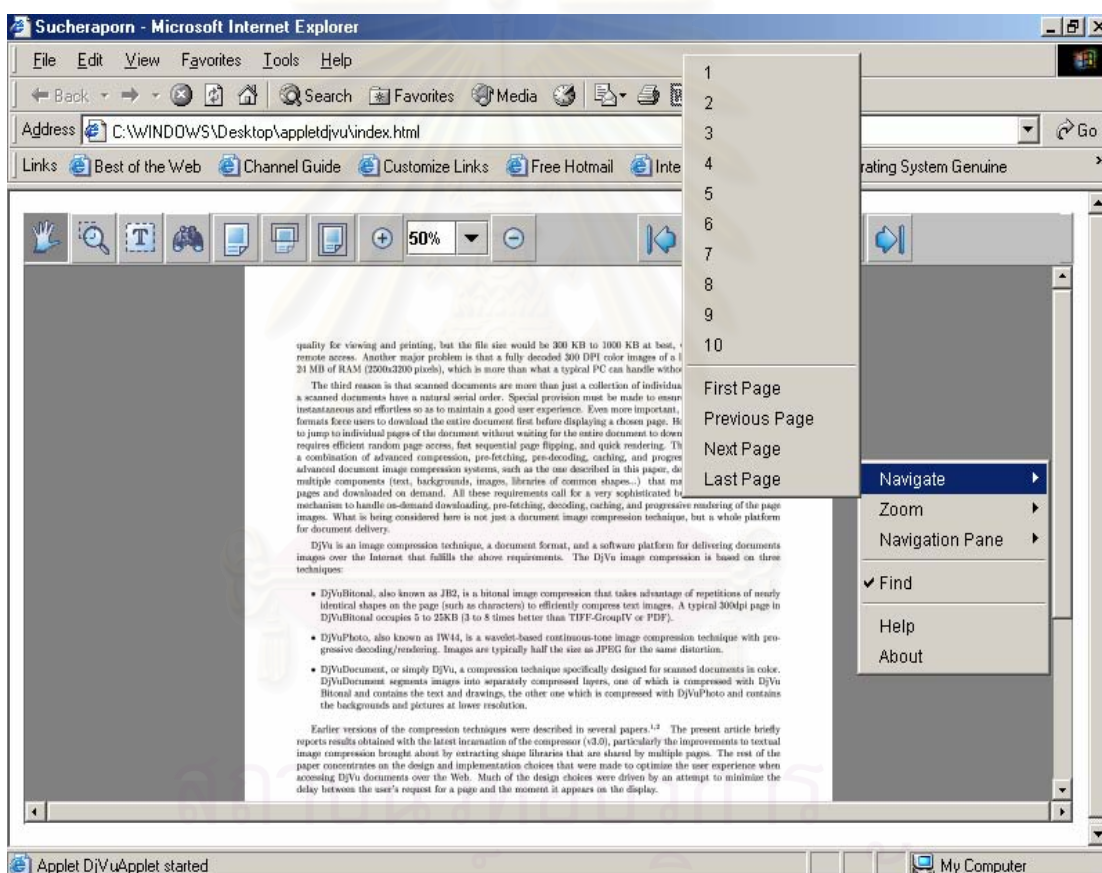
รูปที่ 4.8 การเรียกดูหน้าถัดไป หน้าก่อนหน้าของเอกสาร



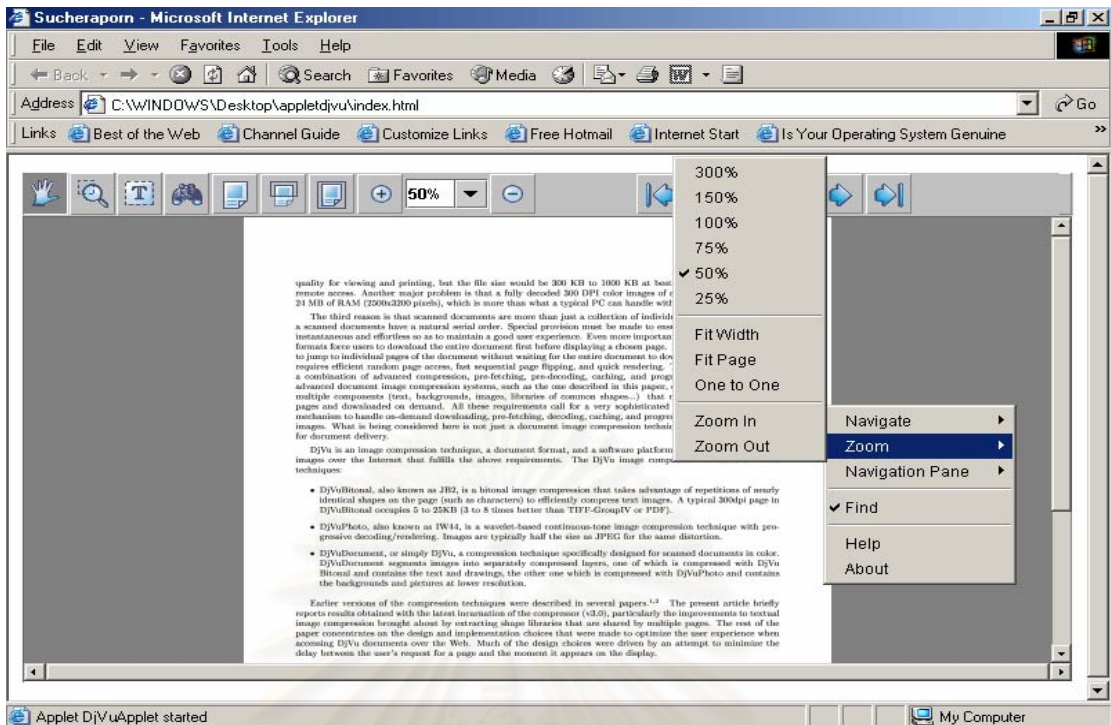
รูปที่ 4.9 การค้นหาคำที่ต้องการภายในเอกสาร

ผู้ใช้สามารถค้นหาคำที่ต้องการได้โดยการกดที่ปุ่ม  บนทูลบาร์ จากนั้นก็จะปรากฏกล่องข้อความออกมาบนทูลบาร์ ให้ผู้ใช้ใส่คำที่ต้องการค้นหา โปรแกรมก็จะทำการค้นหาคำที่ต้องการภายในหน้าเอกสารนั้น ๆ ดังแสดงในรูปที่ 4.9

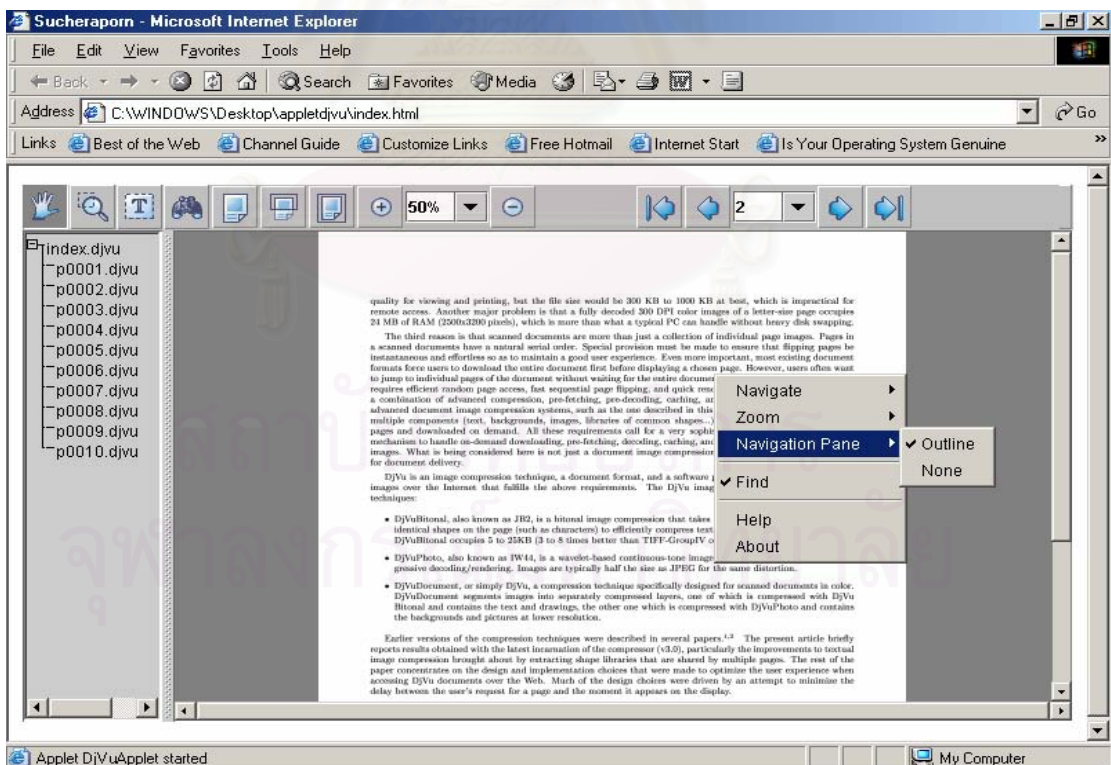
เมื่อผู้ใช้กดปุ่มขวาของเมาส์ก็จะปรากฏเมนู pop-up ดังรูปที่ 4.10, รูปที่ 4.11 และรูปที่ 4.12 ให้ผู้ใช้เลือกคำสั่งในการบราวซ์เอกสาร คำสั่งต่าง ๆ และผลลัพธ์ได้สรุปไว้ในตารางที่ 4.2



รูปที่ 4.10 เมนู pop-up แสดงคำสั่งการเลือกหน้าที่ต้องการ



รูปที่ 4.11 เมนู pop-up แสดงคำสั่งการเลือกความละเอียดของหน้าเอกสาร



รูปที่ 4.12 เมนู pop-up แสดงการเลือกคำสั่ง Navigation Pane

เมื่อผู้ใช้เลือกคำสั่ง Navigation Pane แล้วตามด้วยคำสั่ง outline จากเมนู pop-up ที่เฟรมทางด้านซ้ายของหน้าต่างบราวส์เซอร์ก็จะปรากฏชื่อไฟล์ทั้งหมดที่รวมอยู่ในเอกสารนี้ และเมื่อผู้ใช้ต้องการอ่านรายละเอียดของหน้าไหนก็สามารถใช้เมาส์คลิกไปที่ชื่อไฟล์ของหน้านั้นได้จากรูปที่ 4.10 จะพบว่าชื่อไฟล์ของ DjVu จะมีลักษณะการตั้งชื่อดังนี้ p0001.djvu, p0002.djvu, p0003.djvu ที่เป็นเช่นนี้ก็เพราะว่าเพื่อให้การเรียกดูหน้าเอกสารเป็นไปโดยสะดวกชื่อของแฟ้มเอกสารดิจิทัลจึงต้องมีความสัมพันธ์กับเลขหน้าที่ปรากฏในสิ่งพิมพ์ต้นฉบับ ซึ่งจะทำให้การอ้างอิงเพื่อเรียกดูหน้าเอกสารที่ต้องการทำได้โดยง่าย โดยเฉพาะการเก็บแฟ้มข้อมูลจำนวนมาก ถ้าไม่มีการตั้งชื่อแฟ้มเอกสารให้สื่อถึงข้อมูลที่อยู่ภายใน การเข้าถึงข้อมูลที่ต้องการจะเป็นไปได้ยาก

ตัวอย่างเช่นพิจารณาสิ่งพิมพ์ที่เป็นหนังสือ ข้อมูลที่มีอยู่โดยทั่วไป ประกอบด้วยสองส่วนคือส่วนที่เป็นเนื้อหาภายในซึ่งจะมีเลขหน้ากำกับและเรียงลำดับกันตามเลขหน้าเหล่านั้นจากน้อยไปมาก อีกส่วนหนึ่งคือส่วนที่ไม่มีเลขหน้ากำกับอาทิเช่น ปกหน้า ปกหลัง คำนำ สารบัญฯลฯ เป็นต้น เพื่อให้การเรียงลำดับของแฟ้มเอกสารภาพดิจิทัลซึ่งทำตามลำดับของตัวอักษรของชื่อแฟ้มเป็นไปตามแบบการเรียงลำดับของหน้าเอกสารสิ่งพิมพ์ DjVu จึงได้ออกแบบให้การบันทึกชื่อแฟ้มเป็นไปตามกฎดังต่อไปนี้ คือ

- เอกสารภาพที่เป็นส่วนข้อมูลที่มีเลขหน้ากำกับให้บันทึกชื่อแฟ้มเอกสารภาพเหล่านั้นตามเลขหน้าและมีการใส่เลข "0" หน้าเลขหน้าเหล่านั้น จนกระทั่งครบ 4 หลัก เพื่อให้ความยาวของชื่อแฟ้มเอกสารมีขนาดเท่ากันทุกแฟ้ม
- เพิ่มตัวอักษร "p" ขึ้นต้นทุกแฟ้มที่เป็นหน้าเอกสารที่มีเลขหน้ากำกับเพื่อให้ทราบว่าเป็นชื่อแฟ้มนั้นเป็นข้อมูลในส่วนที่มีเลขหน้ากำกับ
- เอกสารภาพส่วนที่ไม่มีเลขหน้ากำกับเช่น ปกหน้า, สารบัญ เพื่อให้เรียงไปตามลำดับตรงกับสิ่งพิมพ์ต้นฉบับ จึงให้ตั้งชื่อแฟ้มเหล่านี้ตามลำดับ โดยเริ่มจาก 00000, 00001, 00002, ... จนหมด
- เอกสารภาพซึ่งอยู่ส่วนหลังก็ให้ตั้งชื่อแฟ้มเป็นตัวเลขที่ต่อเนื่องกันไปได้เลยแต่และเติมเลขศูนย์ข้างหน้าจนครบสี่หลักเช่นกันหลังจากนั้น เพิ่มตัวอักษร "p" ข้างหน้าของชื่อแฟ้มเป็นลำดับสุดท้าย

เมื่อตั้งชื่อแฟ้มเอกสารตามกฎข้างบน จะทำให้แฟ้มข้อมูลมีการเรียงลำดับเหมือนกับต้นฉบับสิ่งพิมพ์ และมีชื่อแฟ้มทุกแฟ้มขนาดเท่ากันหมด

4.3.2 การควบคุมการทำงานของ Applet ผ่านทาง APPLET tag ของเอกสาร HTML

ผู้วิจัยได้ทำการออกแบบโปรแกรม DjVuApplet เพื่อให้ผู้ใช้งานสามารถควบคุมการทำงานของ Applet ผ่านทาง APPLET tag ของเอกสาร HTML โดยที่ผู้ใช้งานไม่ต้องเข้าไปยุ่งเกี่ยวกับโค้ดของโปรแกรม DjVuApplet รายละเอียดได้กล่าวไว้ในบทที่ 3 แสดงดังตารางที่ 3.1

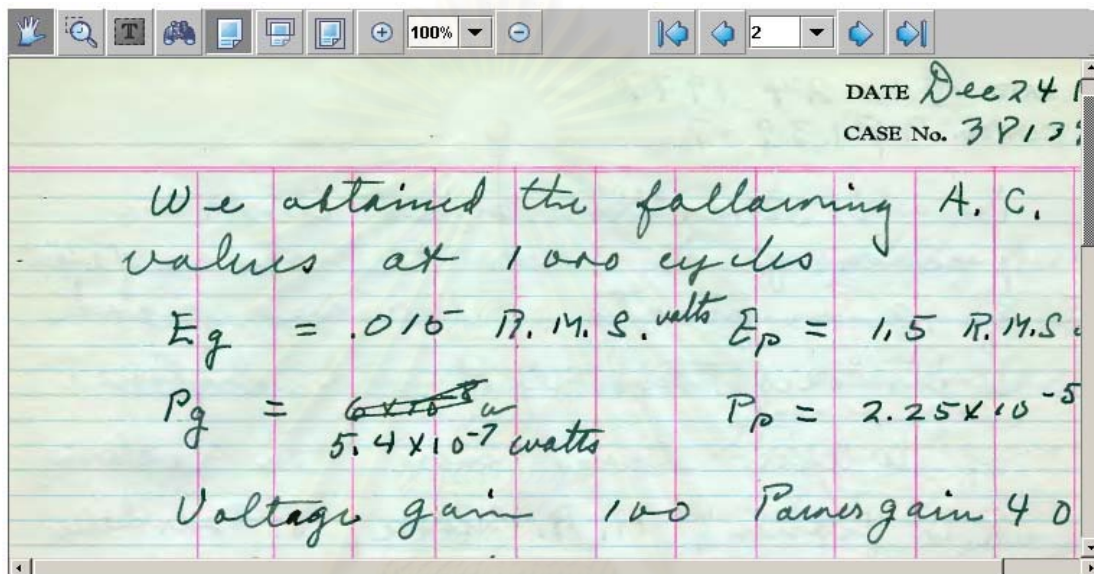
ตัวอย่างเช่น ถ้าผู้ใช้ไม่ต้องการให้โปรแกรม DjVuApplet แสดงเมนู pop-up ก็อาจจะควบคุมการทำงานใน APPLET tag ได้ดังนี้ <param name="menu" value="no">

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML><HEAD>
<TITLE> Sucheraporn </TITLE>
<META NAME="Generator" CONTENT="EditPlus">
<META NAME="Author" CONTENT="">
<META NAME="Keywords" CONTENT="">
<META NAME="Description" CONTENT="">
</HEAD><BODY>
<applet style="border:0px none;margin:0px 0px 0px 0px;width:100%;height:100%;"
code="DjVuApplet.class"
archive="javadjvu.jar"
width="720"
height="480" >
<param name="cache_archive" value="javadjvu.jar" >
<param name="data" value="index.djvu">
<param name="abouturl" value="http://www.ee.eng.chula.ac.th/">
<param name="menu" value="no">
</applet>
</BODY></HTML>
```

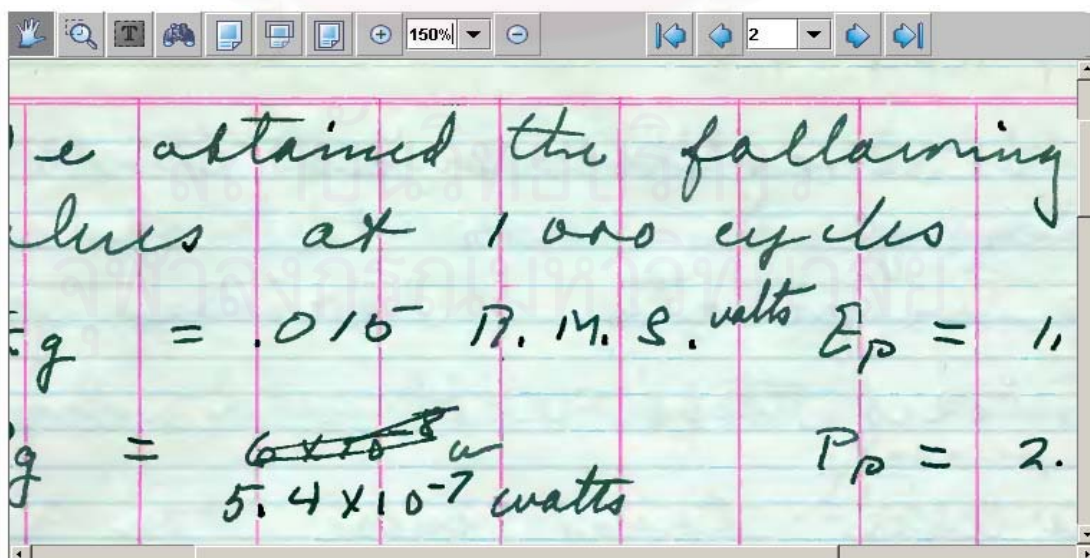
สำหรับการควบคุมคุณสมบัติอื่น ๆ รายละเอียดได้กล่าวไว้ในบทที่ 3 แสดงดังตารางที่ 3.1

4.4 ตัวอย่างการแสดงผลเอกสารภาพดิจิทัล DjVu ในรูปแบบลายมือเขียน

ในหัวข้อนี้จะทำการแสดงผลเอกสารภาพ DjVu ที่เป็นรูปแบบลายมือเขียน เพื่อแสดงให้เห็นถึงความคมชัดของภาพเอกสารเมื่อเรียกดูผ่านโปรแกรมบราวส์เซอร์ภายใต้การทำงานของโปรแกรม DjVuApplet จากรูปที่ 4.13 และรูปที่ 4.14 จะสังเกตเห็นได้ว่าภาพส่วนที่มีความถี่สูง (ประเภทตัวหนังสือ) ของภาพเอกสารยังมีความชัดเจนอยู่และยังคงรักษาพื้นฉากหลังไว้ได้



รูปที่ 4.13 ภาพเอกสาร DjVu ที่เป็นลายมือเขียนแสดงที่ระดับความละเอียดเท่ากับ 100%



รูปที่ 4.14 ภาพเอกสาร DjVu ที่เป็นลายมือเขียนแสดงที่ระดับความละเอียดเท่ากับ 150%

4.5 ตัวอย่างการสร้างเอกสารภาพ DjVu

ในหัวข้อนี้ผู้วิจัยได้ทำการทดสอบเอกสารภาพ DjVu หนึ่งฉบับที่ได้มีการสแกนไว้แล้ว โดยมีขั้นตอนการทำงานดังต่อไปนี้

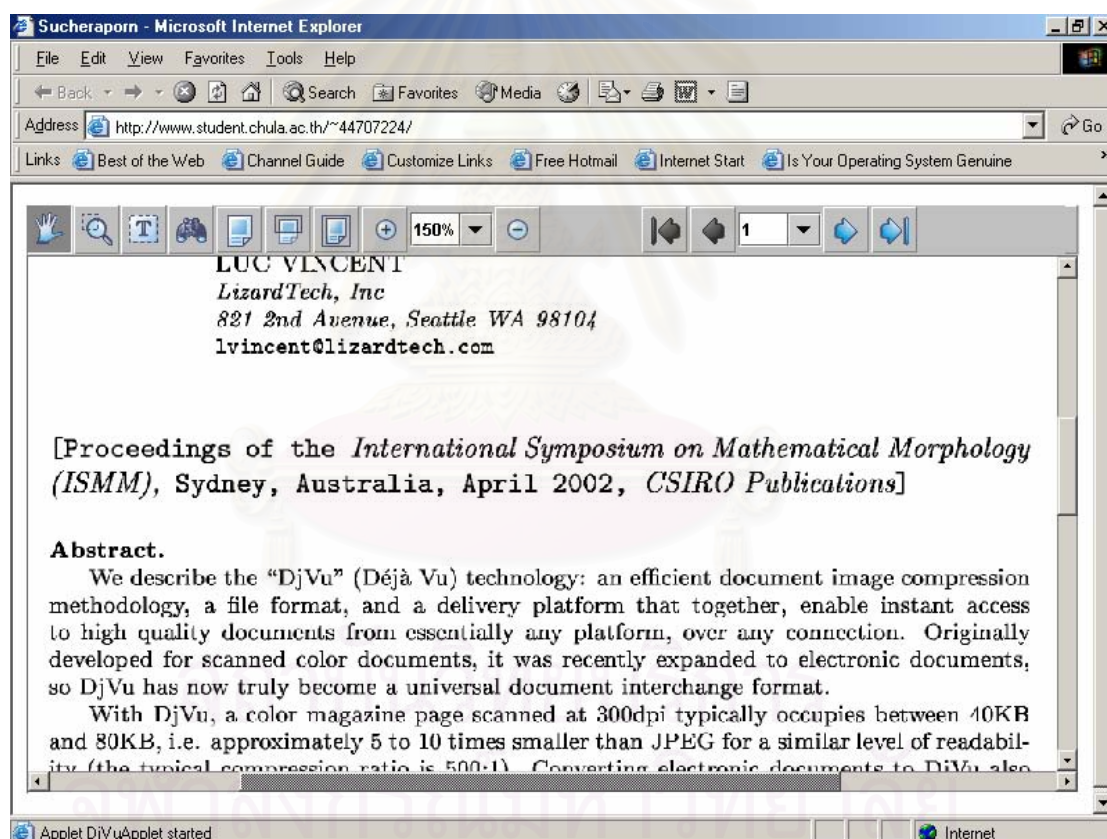
1. ตั้งชื่อไฟล์ว่า index.djvu โดยที่เอกสารดังกล่าวมีขนาดไฟล์เท่ากับ 197 กิโลไบต์ ประกอบไปด้วยหน้าเอกสารทั้งหมด 18 หน้า
2. จากนั้นทำการสร้างเอกสาร HTML ขึ้นมาหนึ่งไฟล์ตั้งชื่อไฟล์ว่า index.html
3. นำเอาไฟล์ DjVuApplet.class (137 ไบต์) ,javadjvu.jar (192 กิโลไบต์) และ index.djvu ไปใส่ไว้ใน APPLET tag ที่อยู่ใน HTML page ดังนี้

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML><HEAD>
<TITLE> Sucheraporn </TITLE>
<META NAME="Generator" CONTENT="EditPlus">
<META NAME="Author" CONTENT="">
<META NAME="Keywords" CONTENT="">
<META NAME="Description" CONTENT="">
</HEAD>
<BODY>
<applet
  style="border:0px none;margin:0px 0px 0px 0px;width:100%;height:100%;"
  code="DjVuApplet.class"
  archive="javadjvu.jar"
  width="720"
  height="480" >
  <param name="cache_archive" value="javadjvu.jar" >
  <param name="data" value="index.djvu">
  <param name="abouturl" value="http://www.ee.eng.chula.ac.th/">
  <param name="helpurl" value="http://www.ee.eng.chula.ac.th/">
</applet></BODY></HTML>
```

4. เมื่อทำทุกขั้นตอนเสร็จแล้วจึงอัปโหลดไฟล์ทั้งหมด (index.djvu, index.html, DjVuApplet.class และ javadjvu.jar) ไปเก็บไว้ในเครื่องคอมพิวเตอร์ที่เป็นเครื่องเว็บเซิร์ฟเวอร์เพื่อให้ผู้ใช้สามารถเรียกดูหน้าเอกสารผ่านโปรแกรมบราวส์เซอร์ได้

รูปที่ 4.15 แสดงหน้าเอกสารเมื่อเรียกดูผ่านโปรแกรมบราวส์เซอร์ภายใต้การทำงานของโปรแกรม DjVuApplet โดยเครื่องคอมพิวเตอร์ที่เชื่อมต่อทางฝั่งไคลเอนท์มีคุณสมบัติดังนี้

- CPU 450 MHz Inter Pentium III processor
- หน่วยความจำ 128 MB
- Microsoft Internet Explorer 6.0
- 56 Kbps modem



รูปที่ 4.15 การเรียกดูเอกสารภาพ index.djvu ผ่านเว็บ

ผู้วิจัยได้ทำการทดสอบเวลาที่ใช้ในการเรียกดูหน้าเอกสาร DjVu ภายใต้การทำงานของโปรแกรม DjVuApplet สามารถสรุปการทำงานได้ดังตารางที่ 4.3

ตารางที่ 4.3 เวลาที่ใช้ในการเรียกดูหน้าเอกสาร DjVu

| การกระทำ (Action) | เวลาที่ใช้ |
|---------------------------------------|----------------------|
| เวลาที่ใช้ในการโหลดหน้าเอกสารครั้งแรก | < 10 วินาที |
| Zooming/Panning | โดยทันที (Immediate) |
| Next/Previous Page | < 1 วินาที |
| Random Page Access | < 2 วินาที |

จากตารางที่ 4.3 จะพบว่าผลลัพธ์ที่ได้จากการทดสอบเวลาที่ใช้ในการเรียกดูหน้าเอกสาร และการทำงานในส่วนติดต่อกับผู้ใช้อยู่ในเกณฑ์ที่น่าพอใจ

สรุปคือการใช้ผู้ใช้สามารถที่จะเรียกดูเอกสารภาพชนิด DjVu ได้ทันที โดยที่ไม่ต้องเสียเวลาในการดาวน์โหลดโปรแกรมปลั๊กอิน (ที่มีขนาดประมาณ 2 MB) มาติดตั้งเพิ่มเติม ก็เป็นวิธีหนึ่งที่จะช่วยป้องกันความเสียหายที่จะเกิดขึ้นกับระบบของไคลเอนท์ (client) ได้ และยังทำให้เกิดความสะดวกกับผู้ใช้งานอีกด้วย

บทที่ 5

สรุปและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

งานวิจัยนี้นำเสนอแนวทางในการแก้ปัญหาระบบการเรียกคืนภาพเอกสาร DjVu ผ่านเว็บ เพื่อเผยแพร่ภาพเอกสาร DjVu ผ่านเว็บให้ง่ายแก่ทั้งผู้เผยแพร่ผลงานและผู้ใช้ในการเรียกดู โดยได้ทำการพัฒนาโปรแกรมในส่วนการเรียกคืนภาพเอกสารด้วยโปรแกรมภาษาจาวา ที่เป็นแอปเพล็ต (Applet) ที่มีชื่อว่า DjVuApplet เพื่อให้ผู้ใช้สามารถที่จะเรียกดูภาพเอกสาร DjVu บนเว็บเบราว์เซอร์ (browser) ได้เลย โดยที่ไม่ต้องเสียเวลาในการดาวน์โหลดโปรแกรมปลั๊กอิน (ที่มีขนาดประมาณ 2 MB) มาติดตั้งเพิ่มเติม ซึ่งเป็นวิธีหนึ่งที่จะช่วยป้องกันความเสียหายที่จะเกิดขึ้นกับระบบของไคลเอนท์(client) ได้ และยังทำให้เกิดความสะดวกกับผู้ใช้งานอีกด้วย

ผู้วิจัยได้ทำการแปลงไลบรารี (version 2) สำหรับการบีบอัดข้อมูลให้เป็นรูปแบบ DjVu ที่ทางบริษัท AT&T แจกสำหรับการพัฒนาจาก C++ ให้อยู่ในรูปของภาษาจาวา เพื่อให้โปรแกรมแอปเพล็ตสามารถที่จะถอดรหัสและแสดงภาพ DjVu บนเบราว์เซอร์ได้

ในวิทยานิพนธ์นี้ได้เลือกใช้โปรแกรม NetBeans เป็น IDE สำหรับการพัฒนาโปรแกรม DjVuApplet เมื่อเราได้ทำการคอมไพล์และรันโปรแกรม DjVuApplet แล้ว NetBeans project ก็ จะทำการสร้างไฟล์มาให้สองไฟล์ คือ

1. DjVuApplet.class (มีขนาดประมาณ 137 bytes)

ใช้สำหรับแสดงผลเอกสารภาพ DjVu ในรูปแบบของแอปเพล็ต

2. javadjvu.jar (มีขนาดประมาณ 192 KB)

ประกอบไปด้วยกลุ่มคลาสสำหรับการบีบอัดข้อมูลให้เป็นรูปแบบ DjVu และกลุ่มคลาสที่ใช้สำหรับการสร้างส่วนติดต่อกับผู้ใช้ (User Interface)

จากผลลัพธ์ของโปรแกรมที่ได้จะพบว่าขนาดของไฟล์ที่ต้องการใช้ในการแสดงผลภาพเอกสาร DjVu มีขนาดเพียง 193 KB เท่านั้น ในขณะที่ปลั๊กอินของ DjVu มีขนาดไฟล์ประมาณ 2 MB (ปลั๊กอินเวอร์ชัน 3.5)

จากนั้นก็นำเอาไฟล์ DjVuApplet.class (137 bytes) และ javadjvu.jar (192 KB) ไปใส่ไว้ใน APPLET tag (APPLET tag คือข้อความที่อยู่ระหว่าง <applet> กับ </applet> ใช้สำหรับเรียก applet ให้ทำงานจากใน HTML page)

เมื่อทำทุกขั้นตอนเสร็จแล้วจึงอัปโหลดไฟล์ทั้งหมด (ไฟล์เอกสารภาพ DjVu, ไฟล์เอกสาร HTML, DjVuApplet.class และ javadjvu.jar) ไปเก็บไว้ในเครื่องคอมพิวเตอร์ที่เป็นเครื่องเว็บเซิร์ฟเวอร์เพื่อให้ผู้ใช้สามารถเรียกดูหน้าเอกสารผ่านโปรแกรมบราวส์เซอร์ได้

การเรียกคืนภาพเอกสาร DjVu ผ่านเว็บโดยใช้โปรแกรมบราวส์เซอร์ ผู้วิจัยได้ทำการออกแบบและพัฒนาฟังก์ชันการทำงานในส่วนติดต่อกับผู้ใช้ (User Interface) ให้ผู้ใช้สามารถใช้งานได้สะดวกขึ้นโดยมีความสามารถเทียบเท่ากับปลั๊กอินมาตรฐานของปลั๊กอิน DjVu

การเรียกคืนภาพเอกสารมีลักษณะเป็น page-on-demand อย่างแท้จริง กล่าวคือผู้ใช้สามารถเรียกหน้าที่ต้องการดูได้โดยตรง หรือไปยังส่วนต่าง ๆ ของเอกสารได้ โดยไม่จำเป็นต้องโหลดหาทุกหน้าในคราวเดียว

ผลลัพธ์ที่ได้จากการทดสอบเวลาที่ใช้ในการเรียกดูหน้าเอกสารและการทำงานในส่วนติดต่อกับผู้ใช้อยู่ในเกณฑ์ที่น่าพอใจ สำหรับสาเหตุของเวลาที่เกิดจากความล่าช้า (delay) ในการเรียกดูหน้าเอกสารน่าจะมีสาเหตุมาจากการทำงานของจาวาแอปเพล็ตเองและประสิทธิภาพของเครื่องทางฝั่งผู้ใช้ แต่ในอนาคตอันใกล้นี้การเรียกใช้งานแอปเพล็ตผ่านทางเว็บบราวส์เซอร์จะมีความเร็วมากกว่าในปัจจุบัน เพราะว่าทางบริษัทซัมไมโครซิสเต็มก็ได้พยายามปรับปรุงการทำงานของจาวาในเรื่องของความเร็วในการใช้งานเพื่อตอบสนองต่อความต้องการของนักพัฒนา

5.2 ข้อเสนอแนะ

1. ควรออกแบบให้แอปเพล็ตมีขนาดไฟล์เล็กลงกว่านี้เพื่อที่จะไม่ต้องใช้เวลาในการดาวน์โหลดหน้าเอกสารนานเกินไป
2. พัฒนาฟังก์ชันเพิ่มเติมให้โปรแกรม DjVuDocumentOrganizer ให้สามารถสร้างและบันทึกชุดของไฟล์ที่เป็นส่วนประกอบของเอกสาร HTML ที่มี APPLETTAG อยู่ภายในลงไปยังไดรฟ์คอลลิดเร็กทอรีที่เลือกไว้ได้เลย โดยที่ผู้สร้างเอกสารไม่ต้องมาทำการสร้าง เอกสาร HTML เอง
3. พัฒนาฟังก์ชันการทำงานเพิ่มเติมให้กับโปรแกรม DjVuApplet ให้สามารถทำงานบนอุปกรณ์ขนาดเล็กซึ่งมีทรัพยากร เช่น การแสดงผล ขนาดของหน่วยความจำ และความสามารถในการประมวลผลจำกัด ตัวอย่างของอุปกรณ์เหล่านี้ได้แก่ โทรศัพท์มือถือ และ PDA เป็นต้น โดยการสร้างโปรแกรม MIDlet เพิ่มเติม ซึ่ง MIDlet ก็คือ แอปพลิเคชันที่สร้างขึ้นเพื่อทำงานบนอุปกรณ์ประเภท MIDP (Mobile Information Device Profile เป็นกลุ่มคลาสไลบรารีที่รองรับการเขียนโปรแกรมบนโทรศัพท์มือถือ) ซึ่งลักษณะของมันก็จะคล้ายกับแอปเพล็ตที่ทำงานบนบราวส์เซอร์นั่นเอง

รายการอ้างอิง

1. ภาณุมาศ หาดทรายทอง. การพัฒนาระบบจัดเก็บและเรียกคืนภาพเอกสารแบบ JBIG ผ่านอินเทอร์เน็ต. วิทยานิพนธ์ปริญญาโทมหาบัณฑิต, สาขาวิศวกรรมไฟฟ้า จุฬาลงกรณ์มหาวิทยาลัย, 2540.
2. สิทธิชัย อังกรวัฒน์ชัย. ระบบจัดเก็บและเรียกคืนเอกสารภาพ DjVu ผ่านเว็บ. วิทยานิพนธ์ปริญญาโทมหาบัณฑิต, สาขาวิศวกรรมไฟฟ้า จุฬาลงกรณ์มหาวิทยาลัย, 2544.
3. L. Bottou, P. Haffner, P. G. Howard, P. Simard, Y. Bengio, and Y. LeCun. High Quality Document Image Compression with DjVu. Journal of Electronic Imaging 7,3 (1998) : 410-428.
4. Patrick Haffner, Leon Bottou, Paul G. Howard, Patrice Simard, Yoshua Beugio, Yann Le Cun. Browsing through High Quality Document Images with DjVu. Proceeding of IEEE International Forum on Research and Technology Advances in Digital Libraries ADL98 (22-24 April 1998)
5. Paul G. Howard, Faouzi Kossentini, Bo Martins, Soren Forchhammer, and William J. Rucklidge. The Emerging JBIG2 Standard. IEEE Transactions on circuits and systems for video technology Vol.8 No.7 (November 1998)
6. Wim Sweldens. The lifting scheme : A custom-design construction of biorthogonal wavelets. Journal of Applied Computing and Harmonic Analysis 3(1996):186-200.
7. Jerry Morrison. EA IFF 85 Standard for Interchange Format Files [online]. Available from : <http://www.concentric.net/~Bradds/iff.html>
8. Bruno R. Preiss. Data Structures And Algorithms with Object-Oriented Design Pattern In Java. John Wiley & Sons ,1999.

ประวัติผู้เขียนวิทยานิพนธ์

นางสาวสุชีราภรณ์ สุรัชย์กุลวัฒนา เกิดเมื่อวันที่ 31 ธันวาคม สำเร็จการศึกษา วิศวกรรมศาสตรบัณฑิต (วศ.บ.) สาขาวิศวกรรมไฟฟ้าสื่อสาร ภาควิชาวิศวกรรมไฟฟ้า คณะ วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ ในปีการศึกษา 2540 และได้เข้า ศึกษาต่อในระดับปริญญาโท สาขาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า คณะ วิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปี พ.ศ. 2544



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย