

ผลการดำเนินการ

ในบทนี้จะกล่าวถึงรายละเอียดเกี่ยวกับผลการดำเนินการพัฒนาระบบจัดเก็บ และเรียกคืนภาพเอกสารตามที่ได้ออกแบบไว้ในบทที่ 3

4.1 ส่วนจัดเก็บภาพเอกสาร

ได้พัฒนาโปรแกรมชื่อ Document Image Organizer (DocOrg) โดยมีวัตถุประสงค์เพื่อจัดสร้างเอกสาร HTML

โปรแกรม Document Image Organizer พัฒนาโดยใช้ภาษา C และการที่ต้องจัดการกับข้อมูลภาพจำนวนมากๆนั้น (เช่นการเข้ารหัส ถอดรหัสภาพ หรือการแสดงผลภาพ) จึงออกแบบให้ DocOrg เป็นโปรแกรมชนิด 32 บิต เพราะจะมองเนื้อหาในหน่วยความจำเป็นแบบเชิงเส้น (flat-address) จึงสามารถใช้หน่วยความจำทั้งหมดที่มีได้ ซึ่งต่างจากโปรแกรมชนิด 16 บิต จะมองเป็นเซกเมนต์และออฟเซต [8] โปรแกรมที่ได้สามารถทำงานภายใต้ระบบปฏิบัติการ Microsoft Windows เวอร์ชัน 3.1x (Win32s)

นอกจากนั้นได้ออกแบบส่วนติดต่อกับผู้ใช้ของโปรแกรมให้เป็นแบบ MDI (multiple document interface) เพื่อให้เปิดหน้าต่างย่อย(child window) ได้หลายๆหน้าต่าง โดยที่หน้าต่างเหล่านั้นอยู่ในกรอบของหน้าต่างหลักของโปรแกรม หน้าต่างย่อยเหล่านี้ใช้แสดงผลภาพ หรือข้อมูลของเอกสาร

ฟังก์ชันการทำงาน และคุณสมบัติสำคัญของโปรแกรมได้แก่

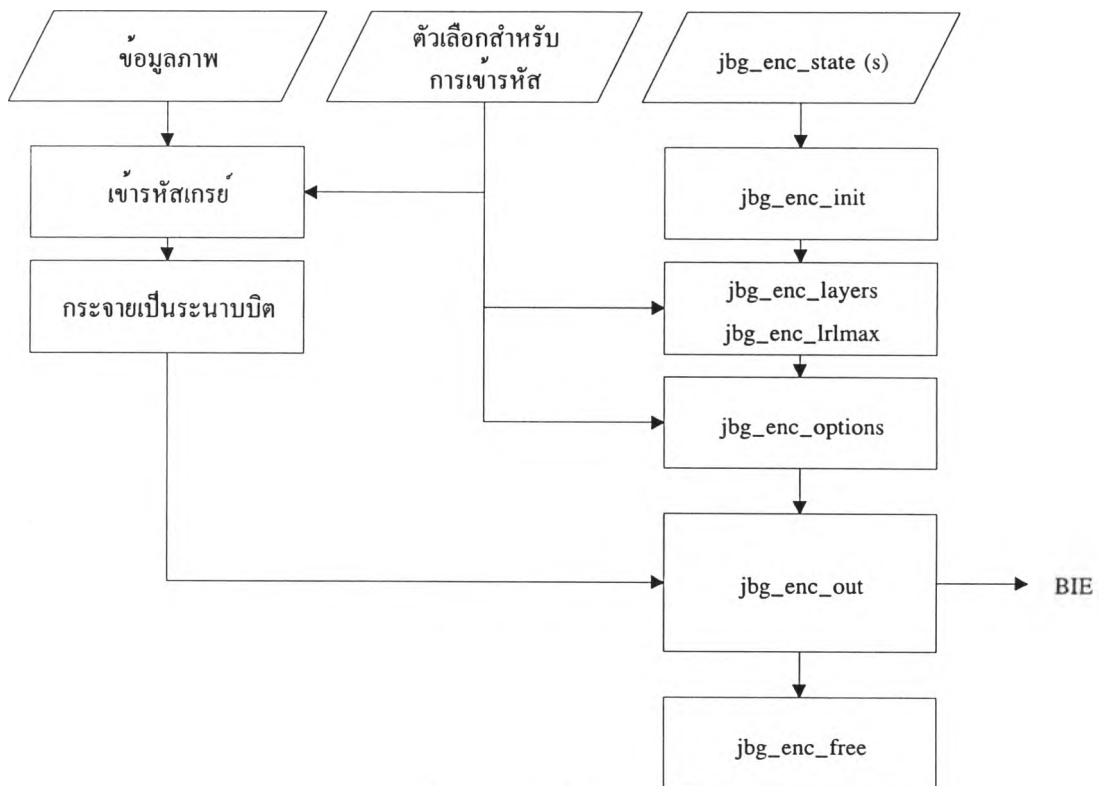
1. เข้ารหัสและถอดรหัสภาพ JBIG
2. แสดงภาพหน้าของเอกสารที่จะนำมาจัดสร้าง
3. แปลงไฟล์ภาพเป็นชุดให้อยู่ในรูป JBIG
4. จัดสร้างเอกสารชนิด HTML

4.1.1 การเข้ารหัสและถอดรหัสภาพ JBIG

ฟังก์ชันส่วนนี้มีไว้สำหรับเข้ารหัสภาพที่ต้องการเก็บให้เป็นแบบ JBIG และถอดรหัสภาพ JBIG เพื่อแสดงผลภาพหรือต้องการอ่านข้อมูลภาพ

ในการเข้ารหัส และถอดรหัสภาพได้นำ JBIG kit [9] มาใช้ ซึ่งพัฒนาโดย Markus Kuhn ชาวเยอรมัน ด้วยภาษา ANSI/ISO C

4.1.1.1 ขั้นตอนในการเข้ารหัสภาพ JBIG



รูปที่ 4.1 การเข้ารหัสภาพ JBIG

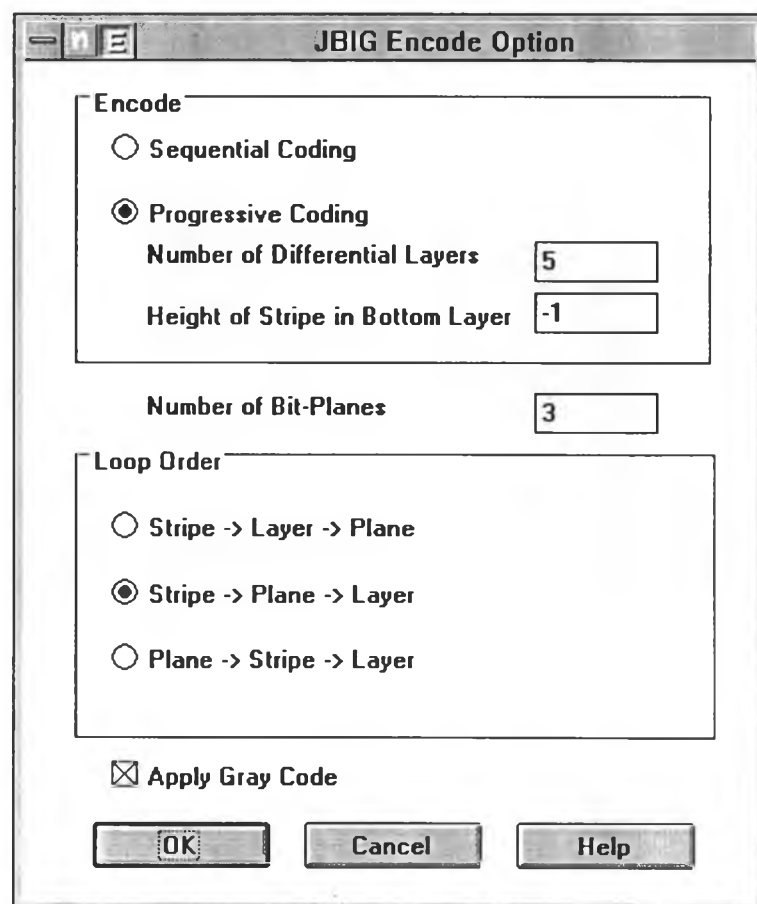
1. จากข้อมูลภาพที่จะเข้ารหัส ให้กระจายออกเป็นระนาบบิต
2. ตั้งค่าตัวเลือกสำหรับการเข้ารหัสที่ได้อะล็อกบ็อกซ์ JBIG Encode Option (รูป 4.2)

ได้แก่

- เลือกความต้องการเข้ารหัสแบบ sequential หรือ progressive
- ถ้าให้การเข้ารหัสเป็นแบบ progressive ต้องเลือกว่าจะให้มีกี่ differential layer หรือให้ความสูงของแถบภาพใน layer ล่างสุดเป็นเท่าใด
- ต้องการบันทึกภาพที่ระนาบบิต
- เลือกลำดับที่จะส่งแถบภาพไปเข้ารหัส (ลำดับของ SDE ใน BIE)
- และเลือกว่าต้องการให้เข้ารหัสเกรย์กับภาพก่อนเข้ารหัส JBIG หรือไม่

3. จัดสรรหน่วยความจำให้กับโครงสร้างของส่วนเข้ารหัส (jbg_enc_state : s)
4. เซ็ตค่าเริ่มต้นให้กับพารามิเตอร์ของ s ด้วยฟังก์ชัน jbg_enc_init()

5. ข้อมูลจากไดอะล็อกบ็อกซ์ กำหนดจำนวน resolution layer ด้วยฟังก์ชัน `jbg_enc_layers()` หรือความสูงของแถบภาพใน layer ล่างสุดด้วยฟังก์ชัน `jbg_enc_lrlmax()` และกำหนดลำดับที่จะส่งแถบภาพไปเข้ารหัสด้วยฟังก์ชัน `jbg_enc_options()`
6. เข้ารหัสภาพ ด้วยฟังก์ชัน `jbg_enc_out()` และส่งผลลัพธ์ไปที่ `data_out()`
7. คืนหน่วยความจำทั้งหมดที่จัดสรรให้กับโครงสร้างของส่วนเข้ารหัส ด้วยฟังก์ชัน `jbg_enc_free()`

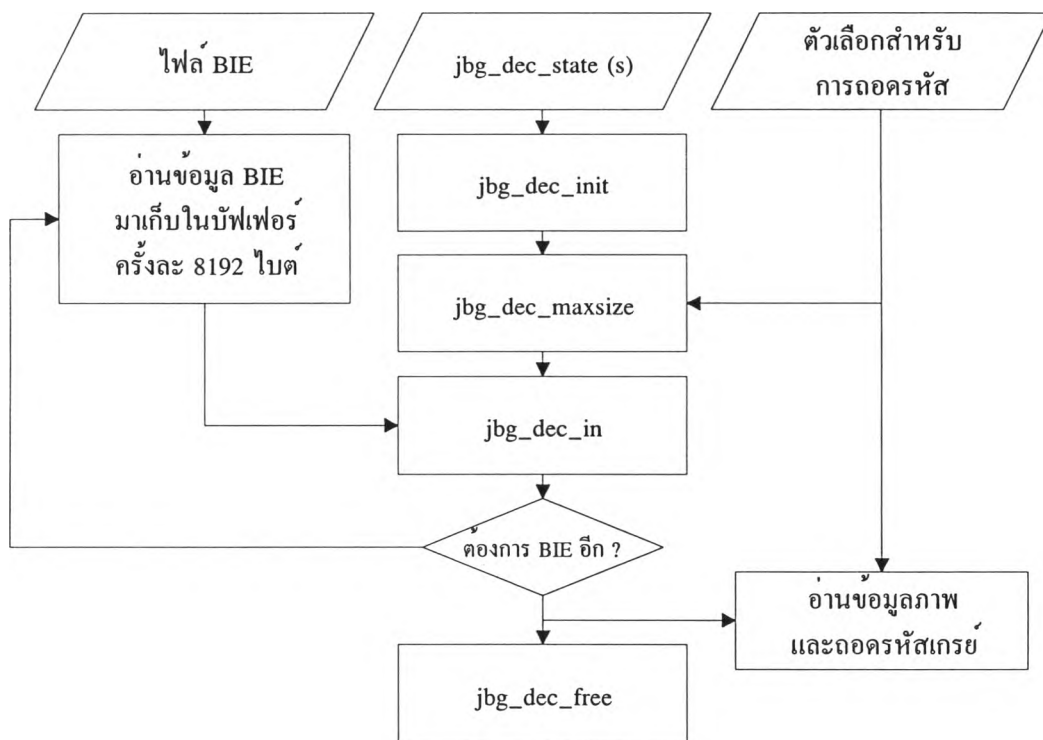


รูปที่ 4.2 ไดอะล็อกบ็อกซ์ JBIG Encode Option

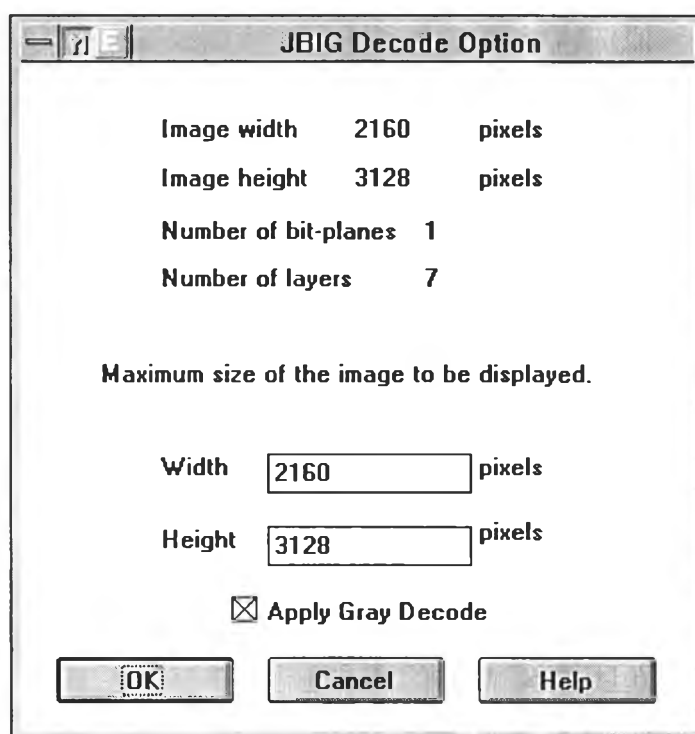
4.1.1.2 ขั้นตอนในการถอดรหัสภาพ JBIG

1. อ่านส่วนหัว(header) ของ BIE จำนวน 20 ไบต์แรก เพื่ออ่านค่า ความกว้าง ความสูง จำนวน layer จำนวนระนาบิต และส่งข้อมูลไปให้ไดอะล็อกบ็อกซ์ JBIG Decode Option (รูปที่ 4.4) ที่ไดอะล็อกบ็อกซ์สามารถตั้งค่าตัวเลือกสำหรับการถอดรหัส เช่น ความกว้าง ความสูงของภาพสูงสุดที่ต้องการถอดรหัส และเลือกว่าการให้ถอดรหัสเกี่ยวข้องกับภาพที่ได้หรือไม่
2. จัดสรรหน่วยความจำให้กับโครงสร้างของส่วนถอดรหัส (`jbg_dec_state : s`)
3. เซ็ตค่าเริ่มต้นให้ `s` ด้วยฟังก์ชัน `jbg_dec_init()`

4. กำหนดความกว้าง ความสูงของภาพสูงสุดที่ต้องการถอดรหัส ด้วยฟังก์ชัน
 jbg_dec_maxsize() โดยรับข้อมูลที่ส่งมาจากไดอะล็อกบ็อกซ์ JBIG Decode Option



รูปที่ 4.3 การถอดรหัสภาพ JBIG



รูปที่ 4.4 ไดอะล็อกบ็อกซ์ JBIG Decode Option

5. อ่านข้อมูลจากไฟล์ BIE มาไว้ที่บัฟเฟอร์ครั้งละ 8162 ไบต์ และส่งให้ฟังก์ชัน `jbg_dec_in()` เพื่อถอดรหัส กระบวนการนี้จะกระทำจนกระทั่งถอดรหัสถึง layer ที่กำหนด หรืออ่านข้อมูลจากไฟล์จนครบ

6. อ่านข้อมูลของภาพที่ได้จากการถอดรหัส ได้แก่ ความกว้าง ความสูงของภาพ ขนาดภาพ จำนวนไบต์ที่ใช้ในหนึ่งบรรทัดของภาพ layer ของภาพที่ได้ จำนวนระนาบของบิต

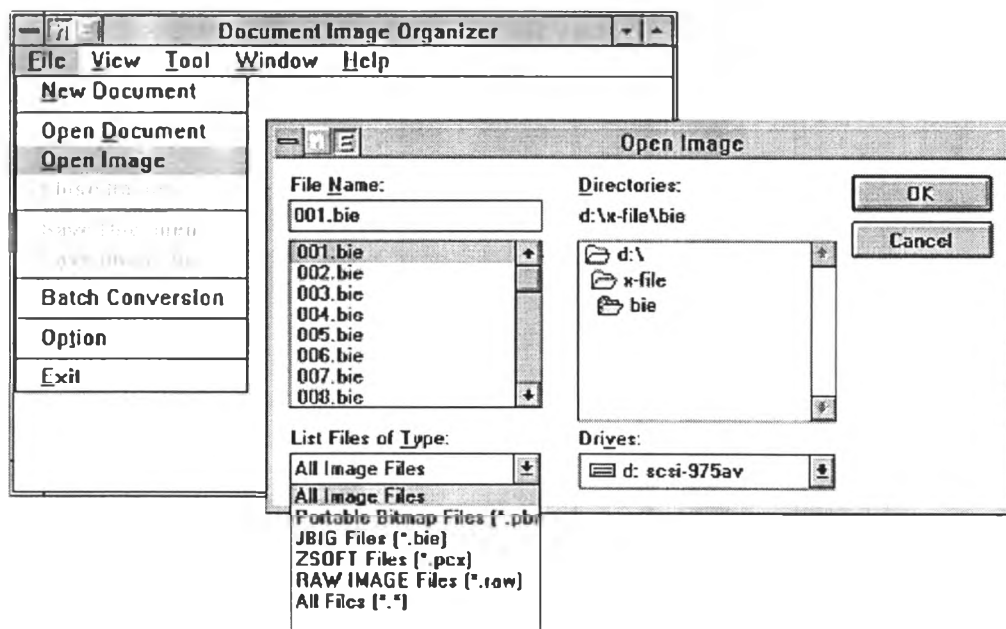
7. อ่านข้อมูลภาพที่ได้จากการถอดรหัส โดยข้อมูลที่ได้อาจจะเป็นชุดของภาพในแต่ละระนาบ บิต จึงต้องรวมระนาบบิตเหล่านี้เข้าด้วยกัน และทำการถอดรหัสเกรย

8. เมื่ออ่านข้อมูลจากโครงสร้างของส่วนถอดรหัสครบแล้ว ต้องคืนหน่วยความจำทั้งหมดที่จัดสรรไว้ ด้วยฟังก์ชัน `jbg_dec_free()`

4.1.2 การแสดงภาพ

ฟังก์ชันส่วนนี้มีไว้สำหรับแสดงภาพหน้าของเอกสารที่จะนำมาจัดสร้างและจัดเก็บ

การอ่านและแสดงภาพ ทำได้โดยเลือก Open Image ที่เมนูไฟล์ของโปรแกรม และเลือกไฟล์ที่ต้องการจากไดอะล็อกบ็อกซ์ Open Image (รูป 4.5)



รูปที่ 4.5 ไดอะล็อกบ็อกซ์ Open Image

โปรแกรมสามารถอ่านและแสดงภาพที่มีรูปแบบและชนิดของไฟล์ตามตาราง 4.1

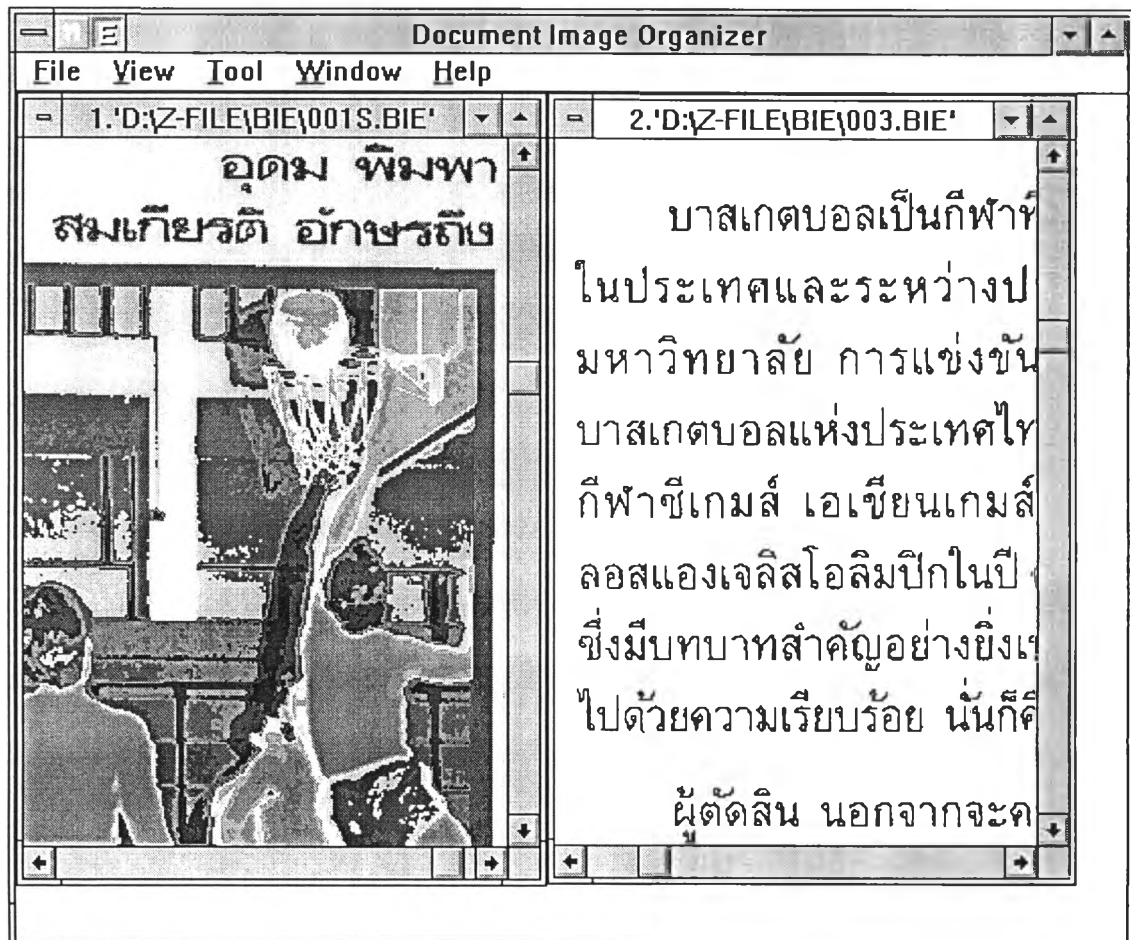
ตารางที่ 4.1 รูปแบบและชนิดของไฟล์ที่สามารถอ่านและแสดงภาพ

รูปแบบไฟล์	นามสกุล	ชนิดของภาพ
JBIG	BIE	1-8 บิตต่อจุดภาพ (sequential / progressive / Gray code)
Portable bitmap	PBM	1 บิตต่อจุดภาพ (text หรือ raw binary format)
ZSOFT	PCX	1 บิตต่อจุดภาพ
Raw Image	RAW	8 บิตต่อจุดภาพ และต้องรู้ขนาดของภาพ (จำนวนจุดภาพตามแนวนอน และแนวตั้ง)

การอ่านและแสดงภาพ ยังสามารถทำได้โดยใช้เมาส์เลือกไฟล์ หรือกลุ่มของไฟล์ที่ต้องการจากโปรแกรม File Manager แล้วลากมาปล่อยไว้ที่หน้าต่างของโปรแกรม Document Image Organizer

โปรแกรมจะแสดงหน้าของเอกสารแต่ละหน้าในหน้าต่างย่อย และสามารถแสดงพร้อมกันทีละหลายหน้าได้

จากรูป 4.6 แสดงภาพหน้าของเอกสารสองหน้าที่เปิดอยู่ ภาพซ้ายเป็นภาพชนิด 3 บิตต่อจุดภาพ และขวาเป็นภาพชนิด 1 บิตต่อจุดภาพ



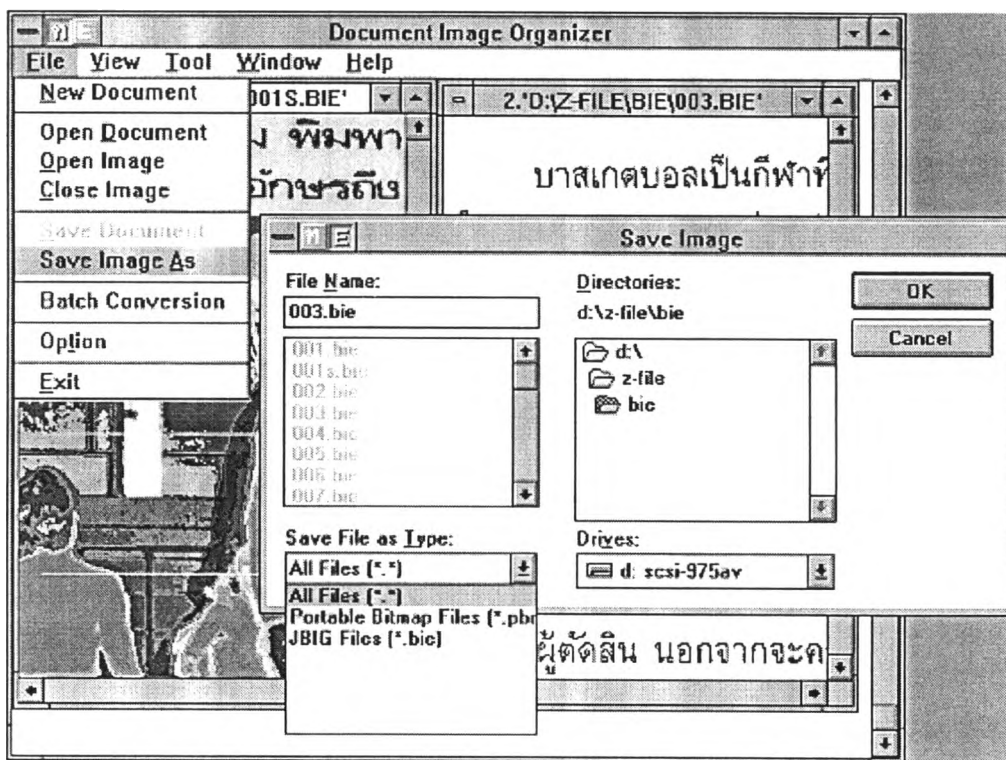
รูปที่ 4.6 ภาพหน้าของเอกสาร

การซูมภาพสามารถทำได้โดยเลือก Zoom In , Zoom Out จากเมนู View หรือใช้แป้นพิมพ์โดยกดปุ่ม “+” เมื่อต้องการซูมเข้า และ “-” เมื่อต้องการซูมออก

การเลื่อนหน้าต่างเพื่อดูภาพสามารถทำได้โดยใช้เมาส์เลื่อนที่ scroll bar หรือใช้ปุ่มลูกศรของแป้นพิมพ์

เมื่อต้องการให้โปรแกรมปรับหน้าต่างใหม่ สามารถทำได้โดยเลือก Refresh จากเมนู View หรือปุ่ม F5

การบันทึกภาพ ทำได้โดยเลือก Save Image As ที่เมนูไฟล์ของโปรแกรม และใส่ชื่อไฟล์ที่ต้องการบันทึกในไดอะล็อกบ็อกซ์ Save Image (รูป 4.7)



รูปที่ 4.7 ไดอะล็อกบ็อกซ์ Save Image

โปรแกรมสามารถบันทึกภาพให้มีรูปแบบและชนิดของไฟล์ตามตาราง 4.2

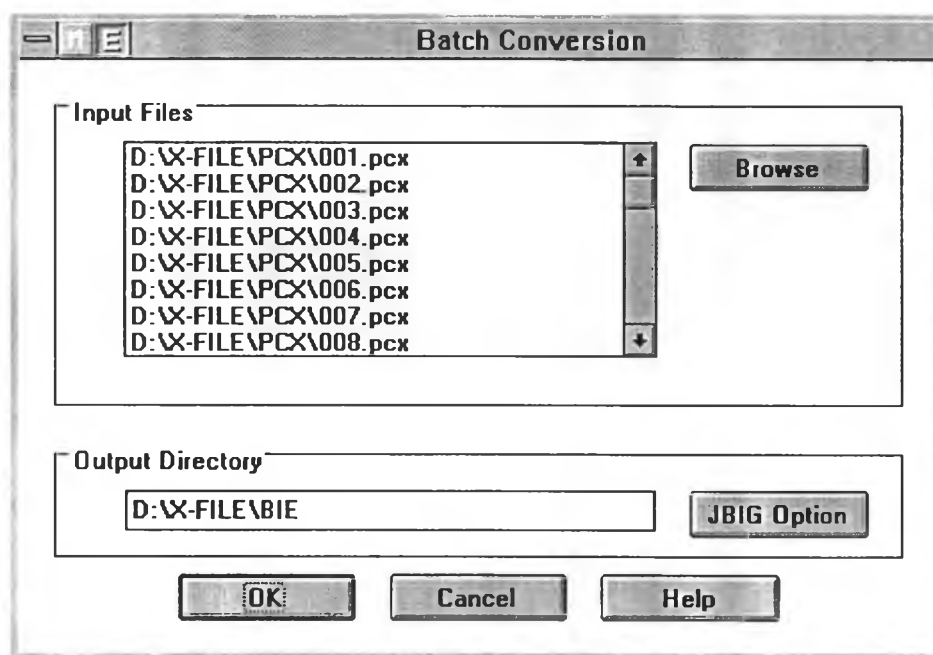
ตารางที่ 4.2 รูปแบบและชนิดของไฟล์ที่สามารถบันทึกได้

รูปแบบไฟล์	นามสกุล	ชนิดของภาพ
JBIG	BIE	1-8 บิตต่อจุดภาพ (sequential / progressive / Gray code)
Portable bitmap	PBM	1 บิตต่อจุดภาพ (raw binary format)

4.1.3 การแปลงไฟล์ภาพเป็นชุด

การแปลงไฟล์ภาพจากรูปแบบหนึ่งไปยังอีกแบบหนึ่ง สามารถทำได้โดยเปิดไฟล์ภาพที่ต้องการแปลง และบันทึกภาพเป็นแบบที่ต้องการ แต่ถ้าต้องการแปลงไฟล์จำนวนมากๆ เช่น ภาพที่ได้จากการสแกนเอกสารทั้งเล่ม วิธีการดังกล่าวจะไม่สะดวกและเสียเวลา จึงต้องมีฟังก์ชันเพื่อให้โปรแกรมสามารถแปลงไฟล์เป็นชุดได้

ฟังก์ชันนี้จะใช้แปลงไฟล์ภาพหน้าของเอกสารที่ได้จากการสแกนให้อยู่ในรูปแบบ JBIG หรือใช้เปลี่ยนตัวเลือกของภาพ JBIG การใช้งานทำได้โดยเลือก Batch Conversion ที่เมนูไฟล์ของโปรแกรม และที่ใดจะลือกบ็อกซ์การแปลงไฟล์เป็นชุด (รูป 4.8) เลือกกลุ่มของไฟล์ที่ต้องการแปลง เลือกไดเรกทอรีที่จะบันทึกผลลัพธ์ และกำหนดค่าตัวเลือกในการเข้ารหัส JBIG



รูปที่ 4.8 ใดจะลือกบ็อกซ์ของการแปลงไฟล์เป็นชุด

ตาราง 4.3 แสดงรูปแบบและชนิดของไฟล์ที่สามารถแปลงได้ และจุดประสงค์ของการใช้งาน

ตารางที่ 4.3 รูปแบบและชนิดของไฟล์ที่สามารถแปลงได้

ไฟล์ที่ต้องการแปลง	ไฟล์ที่ได้	จุดประสงค์
PBM หรือ PCX (1 บิตต่อจุดภาพ)	BIE (1 บิตต่อจุดภาพ)	แปลงภาพเอกสารที่ได้จากการสแกนให้อยู่ในรูปแบบ JBIG
BIE (1 บิตต่อจุดภาพ)	BIE (1 บิตต่อจุดภาพ)	เปลี่ยนจำนวน layer
BIE (2-8 บิตต่อจุดภาพ)	BIE (2-8 บิตต่อจุดภาพ)	เปลี่ยนจำนวน layer และ/หรือ ใช้/ไม่ใช่ Gray code
BIE (2-8 บิตต่อจุดภาพ)	BIE (1-7 บิตต่อจุดภาพ)	ลดจำนวนระนาบบิต

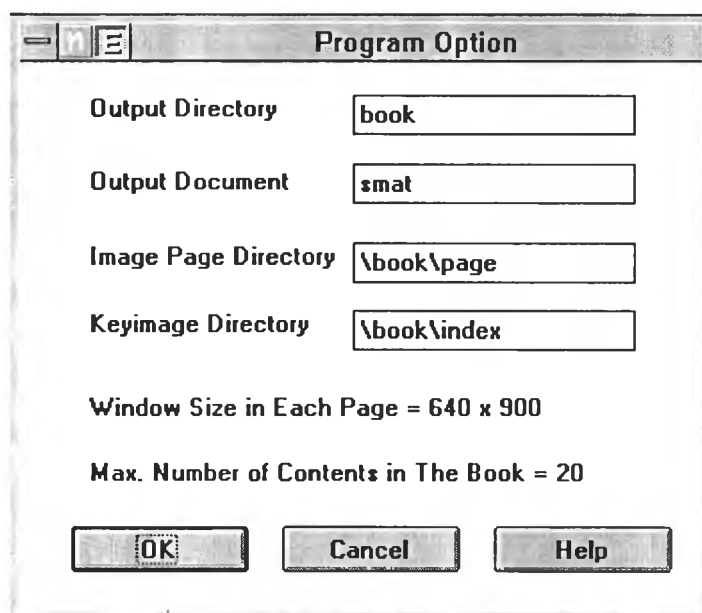
4.1.4 การจัดสร้างเอกสารชนิด HTML

หน้าที่การทำงานหลักของโปรแกรม Document Image Organizer คือ จัดสร้างเอกสารชนิด HTML โดยจะรับข้อมูลภาพที่ได้จากการสแกน ต่อจากนั้นต้องใส่ข้อมูลของเอกสาร ทำสารบัญของเอกสาร และทำดัชนีของเอกสาร โปรแกรมจะสร้างเอกสารซึ่งประกอบด้วยชุดของไฟล์ HTML

4.1.4.1 การเตรียมการจัดสร้างเอกสาร

ในการจัดสร้างเอกสาร เริ่มต้นโดยการสแกนเอกสารที่ต้องการจัดเก็บให้ครบทุกหน้า และบันทึกภาพหน้าของเอกสารที่สแกนได้ในรูปแบบ PCX (1 บิตต่อจุดภาพ) หรือ PBM (1 บิตต่อจุดภาพ) หรือ RAW (8 บิตต่อจุดภาพ) ชื่อของไฟล์ภาพเอกสารควรเป็นตัวเลขเรียงตามลำดับของหน้ากระดาษเพื่อให้ส่วนเรียกคืนภาพเอกสารสามารถหาหน้าของเอกสารได้ ต่อจากนั้นใช้การแปลงไฟล์แบบกลุ่มแปลงไฟล์ภาพหน้าของเอกสารให้อยู่ในรูปของ BIE (JBIG)

เมื่อได้ชุดของไฟล์ภาพเอกสารที่จะนำมาใช้แล้ว ให้ตั้งค่าตัวเลือกโดยเลือก Option จากเมนูไฟล์ ที่ไอคอนบล็อก Program Option (รูปที่ 4.9) สามารถกำหนดไดเรกทอรีที่จะเก็บเอกสาร (output directory) และชื่อไฟล์เอกสาร(output document)

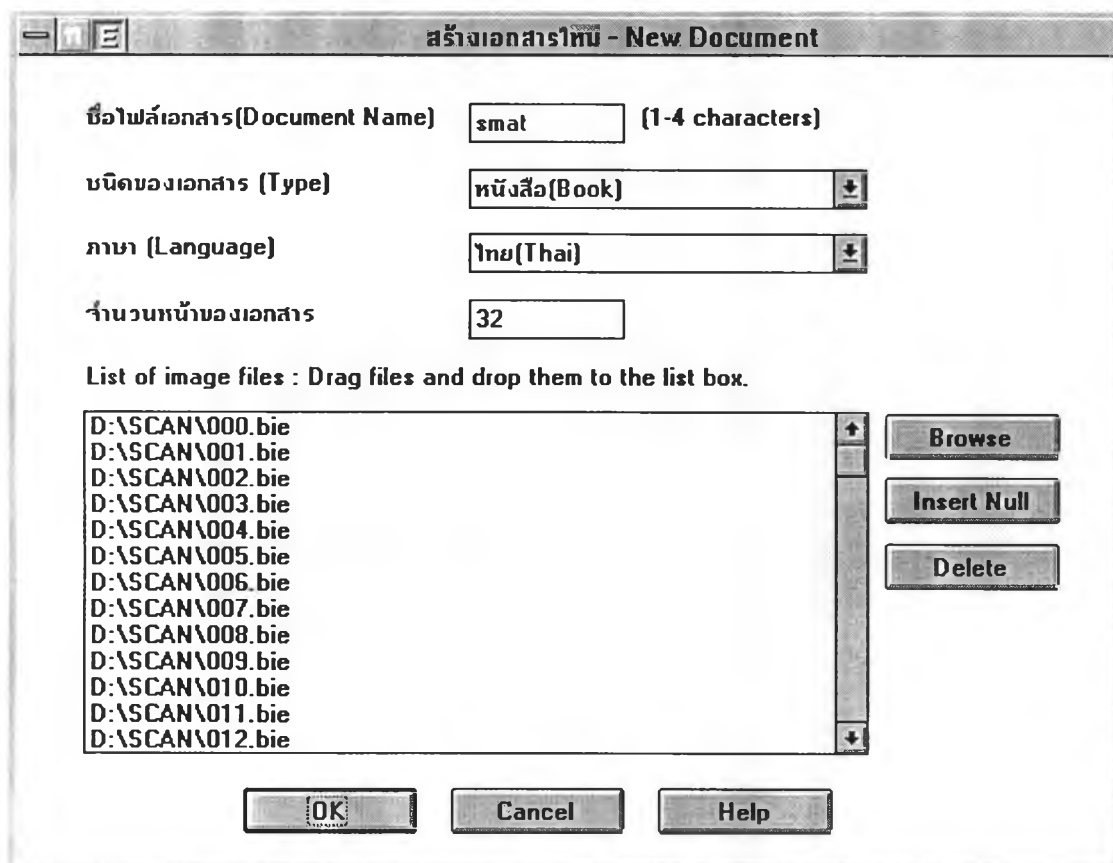


รูปที่ 4.9 ไอคอนบล็อก Program Option

4.1.4.2 การกำหนดจำนวนหน้าและชนิดของเอกสาร

หลังจากได้ไฟล์ภาพเอกสารที่จะนำมาใช้ และตั้งค่าตัวเลือกแล้ว ต่อจากนั้นให้เลือก New Document จากเมนูไฟล์ เพื่อสร้างเอกสารใหม่ ที่ไอคอนบล็อกสร้างเอกสารใหม่ (รูปที่ 4.10)

ผู้จัดทำต้องระบุชนิดเอกสาร(มีด้วยกัน 6 ตัวเลือก คือ หนังสือ รายงานการประชุมวิชาการ บทความทางวิชาการ วารสารทางวิชาการ วิทยานิพนธ์ และอื่นๆ) ระบุภาษาของเอกสาร(มีด้วยกัน 6 ตัวเลือก คือ ไทย อังกฤษ ญี่ปุ่น เยอรมัน ฝรั่งเศส และอื่นๆ) พร้อมกับกำหนดจำนวนหน้าทั้งหมดของเอกสาร โดยเลือกไฟล์ภาพเอกสารที่จะนำมาใช้ 1 ไฟล์ คือ 1 หน้า บางหน้าอาจจะไม่เป็นภาพก็ได้(ต้องใส่เป็น NULL)



รูปที่ 4.10 ไดอะล็อกบ็อกซ์สำหรับสร้างเอกสารใหม่

ไฟล์ภาพของหน้าเอกสารทั้งหมดที่เลือก จะถูกสำเนาไปยังไดเรกทอรีย่อย page และจะปรากฏไดอะล็อกบ็อกซ์ให้ใส่ข้อมูลของเอกสาร และไดอะล็อกบ็อกซ์สร้างสารบัญของเอกสาร

4.1.4.3 การทำข้อมูลของเอกสาร

ข้อมูลของเอกสารจะบอกถึงรายละเอียดของเอกสารที่สำคัญๆ เช่น ชื่อเอกสาร ชื่อผู้แต่ง สำนักพิมพ์ สถานที่พิมพ์ วันที่พิมพ์ ครั้งที่พิมพ์ ฯลฯ รวมทั้งบทคัดย่อซึ่งจะกล่าวถึงเนื้อหาโดยย่อของเอกสารนั้นๆ ในการทำข้อมูลของเอกสาร หลังจากผู้จัดทำเลือกชนิดของเอกสารและเลือกไฟล์ภาพสำหรับหน้าเอกสารแล้ว โปรแกรมจะขึ้นไดอะล็อกบ็อกซ์ (รูป 4.11) ให้ผู้จัดทำใส่รายละเอียดของเอกสาร ซึ่งสัมพันธ์กับชนิดของเอกสารตามที่สรุปไว้ในตาราง 4.4

รายละเอียดของเอกสาร - Document Information

ชนิดของเอกสาร (Type) หนังสือ(Book)

ชื่อเอกสาร (Title) วิธีทำสมาธิเบื้องต้น

ชื่อผู้แต่ง (Authors) พุทธทาสภิกขุ

สำนักพิมพ์ (Publisher) หก. ภาพพิมพ์

สถานที่พิมพ์ กรุงเทพฯ

ปีพิมพ์ (Date) 2539

ครั้งที่พิมพ์ 1

ภาษา (Language) ไทย(Thai)

จำนวนหน้า 32

บทคัดย่อ (Abstract) วิธีทำสมาธิ คือการฝึกจิต

Next >> Cancel Help

รูปที่ 4.11 ไอคอนบล็อกบุ๊กสำหรับใส่รายละเอียดของเอกสารที่เป็นหนังสือ

ตารางที่ 4.4 แสดงส่วนข้อมูลของเอกสารในเอกสารแต่ละชนิด

ชนิดของเอกสาร	ส่วนข้อมูลของเอกสาร
หนังสือ	ชื่อผู้แต่ง ชื่อหนังสือ สำนักพิมพ์ สถานที่พิมพ์ ปีพิมพ์ ครั้งที่พิมพ์ ลำดับที่ จำนวนเล่ม จำนวนหน้าทั้งหมด บทคัดย่อ
บทความทางวิชาการ	ชื่อบทความ ชื่อผู้เขียนบทความ ชื่อเรื่อง เลขหน้า สำนักพิมพ์ สถานที่พิมพ์ ปีพิมพ์ บทคัดย่อ

ตารางที่ 4.4 (ต่อ)

ชนิดของเอกสาร	ส่วนข้อมูลของเอกสาร
รายงานการประชุมวิชาการ และวารสารทางวิชาการ	ชื่อรายงานการประชุมหรือวารสาร สำนักพิมพ์ สถานที่พิมพ์ ปีพิมพ์ บทคัดย่อ
วิทยานิพนธ์	ผู้เขียนวิทยานิพนธ์ ชื่อวิทยานิพนธ์ ระดับวิทยานิพนธ์ ชื่อสาขาวิชาหรือภาควิชา คณะ ชื่อมหาวิทยาลัย ปีพิมพ์ บทคัดย่อ

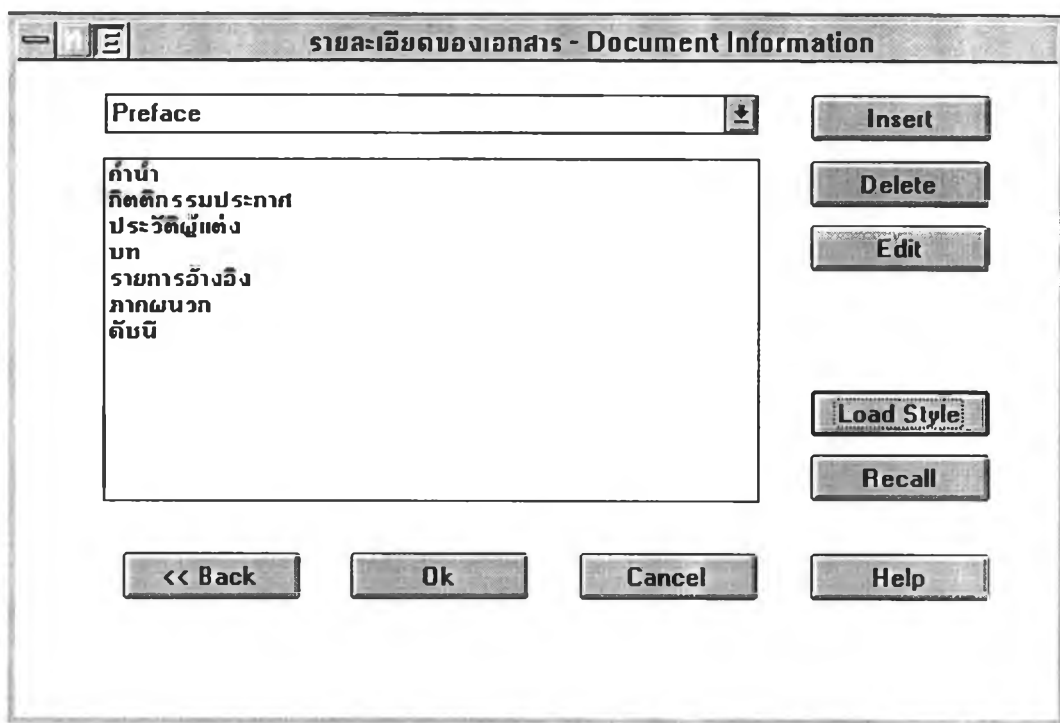
4.1.4.4 การทำสารบัญของเอกสาร

ในภายเอกสารเล่มหนึ่ง ย่อมประกอบไปด้วยส่วนต่างๆ ซึ่งแต่ละส่วนจะมีหน้าที่และความหมายที่ต่างกัน เมื่อรวมส่วนต่างๆ เหล่านี้เข้าด้วยกันจะได้โครงสร้างของเอกสารขึ้นมา

ส่วนประกอบหรือ content ที่กล่าวถึงนี้ ได้แก่ หน้าปก บทคัดย่อ บทนำ เนื้อหา บทสรุป ภาคผนวก รายการอ้างอิง ฯลฯ ซึ่งเอกสารแต่ละชนิดจะมีส่วนประกอบที่ต่างกันไป

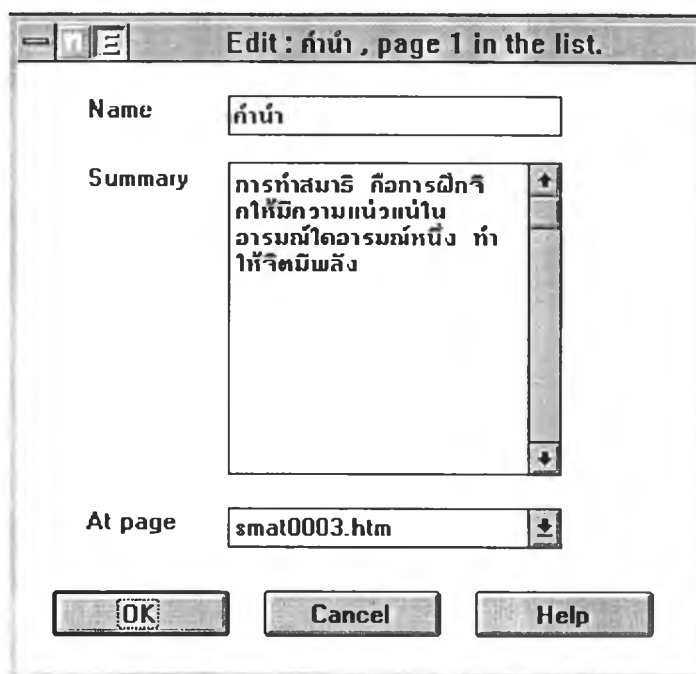
ที่ได้อะลือกบ็อกซ์สำหรับสร้างสารบัญของเอกสาร (รูปที่ 4.12) สามารถเลือกได้ว่าเอกสารนั้นมีส่วนประกอบอะไรบ้าง โดยเลือกจากรายการในคอมโบบ็อกซ์(combo box) หรือสามารถเลือกรูปแบบ(style)ที่โปรแกรมจัดให้ ได้แก่

1. หนังสือ ประกอบด้วย คำนำ กิตติกรรมประกาศ ประวัติผู้แต่ง สารบัญ บท รายการอ้างอิง ภาคผนวก และดัชนี
2. รายงานการประชุมวิชาการ (conference) ประกอบด้วย คำนำ สารบัญ บทความ และดัชนีผู้เขียนบทความ
3. บทความทางวิชาการ (paper) ประกอบด้วย บทคัดย่อ บทนำ เนื้อหา สรุป กิตติกรรมประกาศ รายการอ้างอิง และประวัติผู้เขียน
4. วารสารทางวิชาการ (journal) ประกอบด้วย คำนำ สารบัญ บทความ และดัชนีผู้เขียนบทความ
5. วิทยานิพนธ์ (thesis) ประกอบด้วย หน้าปก หน้าอนุมัติ บทคัดย่อ กิตติกรรมประกาศ สารบัญ บทนำ ตัวเรื่อง ขอสรุปและเสนอแนะ รายการอ้างอิง ภาคผนวก และประวัติผู้เขียน



รูปที่ 4.12 ไอคอนเลือกบ็อกซ์สำหรับสร้างสารบัญของเอกสาร

เมื่อเลือกส่วนประกอบของเอกสารแล้ว ผู้จัดทำต้องใส่รายละเอียดของแต่ละส่วน ได้แก่ ชื่อของส่วนประกอบ เนื้อความโดยย่อในส่วนนั้น และชื่อของไฟล์ที่อ้างอิง ซึ่งเป็นหน้าเริ่มต้นของส่วนประกอบนั้น ตามรูป 4.13



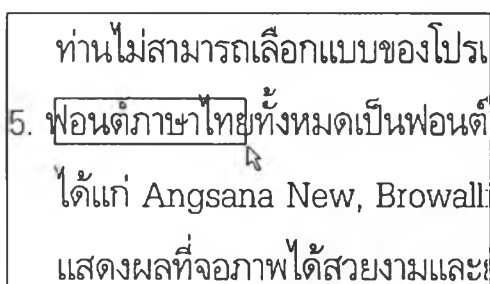
รูปที่ 4.13 ไอคอนเลือกบ็อกซ์สำหรับใส่รายละเอียดในแต่ละส่วนประกอบของเอกสาร

4.1.4.5 การทำดัชนีของเอกสาร

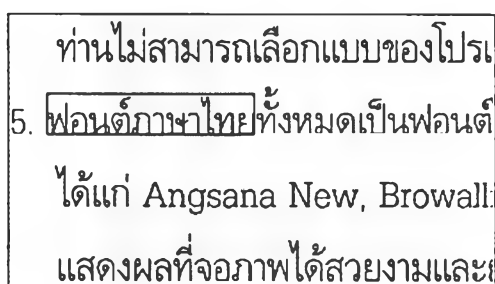
เนื่องจากภาพเอกสารเป็นข้อมูลบิตแมพ เราไม่สามารถทราบได้เลยว่าภายในหน้าหนึ่งๆ ประกอบด้วยอะไร มีข้อความอะไรบ้าง จึงจำเป็นต้องสร้างฐานข้อมูลเพื่อให้สามารถค้นหาบางคำในเอกสารได้ ฐานข้อมูลที่กล่าวถึงนั้นก็คือดัชนีของเอกสาร

โดยทั่วไปข้อมูลที่นำมาใช้คือ คำสำคัญ (keyword) ซึ่งผู้จัดทำจะเลือกคำสำคัญ และหาว่ามีคำสำคัญนี้อยู่ที่ใดหน้าใดบางของเอกสาร การวิจัยนี้เสนอการทำดัชนีของเอกสาร โดยที่เก็บคำสำคัญเป็นภาพ หรือเรียกว่าภาพคำสำคัญ (keyimage)

ในการทำดัชนีของเอกสาร ผู้จัดทำจะใช้เมาส์เลือกบริเวณภาพที่ต้องการจะนำไปทำดัชนี และสามารถให้โปรแกรมปรับบริเวณที่เลือกให้เหมาะสมถูกต้องได้โดยกดปุ่ม ESC ขณะที่เมาส์เลือก (ตามรูป 4.14)

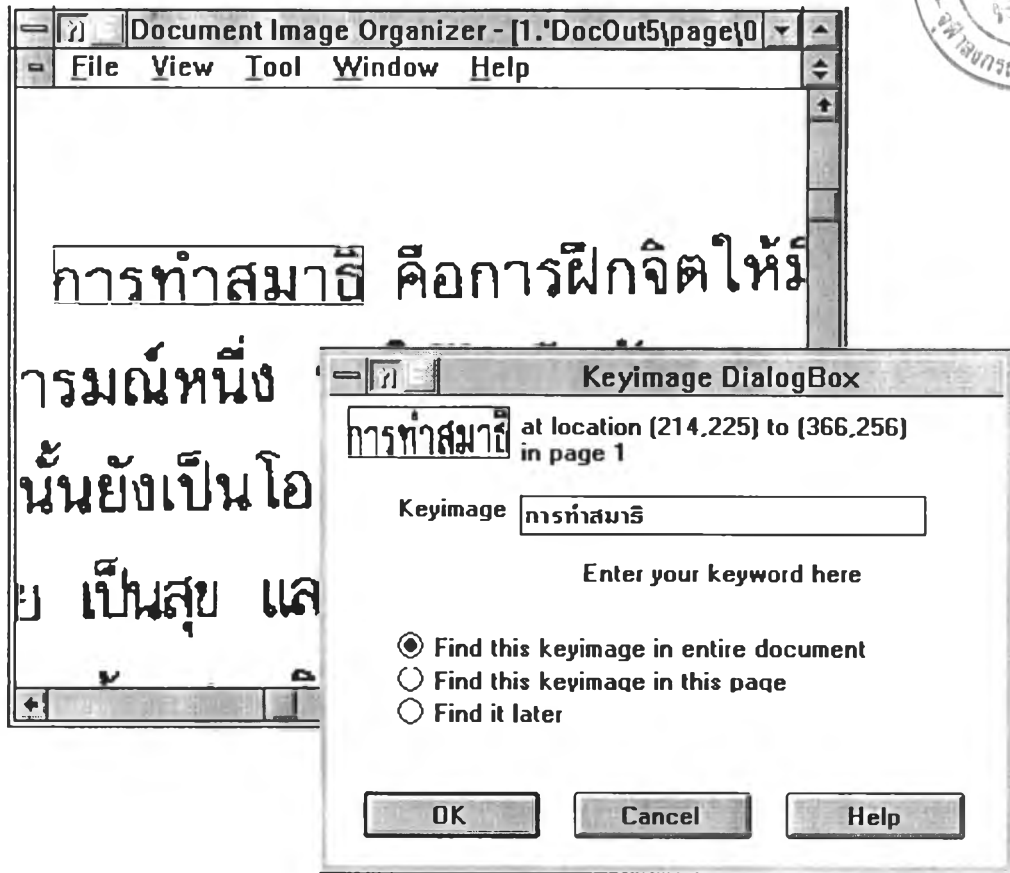


รูปที่ 4.14 ก. ตัวอย่างการเมาส์เลือกบริเวณที่เป็นคำสำคัญ



ข. โปรแกรมจะปรับบริเวณที่เลือกให้เหมาะสมถูกต้อง

จากรูป 4.15 แสดงตัวอย่างของการเลือก keyimage หลังจากที่ได้เลือกบริเวณคำที่ต้องการได้แล้ว จะปรากฏไดอะล็อกบ็อกซ์ให้ใส่ชื่อของ keyimage และผู้จัดทำสามารถเลือกได้ว่าจะให้โปรแกรมหาว่ามี keyimage นี้ที่ตำแหน่งอื่นอีกหรือไม่ในเอกสาร หรือต้องการให้หาเฉพาะในหน้านี้เท่านั้น หรือไม่ต้องการเก็บเฉพาะตำแหน่งนี้ตำแหน่งเดียว



รูปที่ 4.15 ไดอะล็อกบ็อกซ์สำหรับเลือก keyimage

keyimage ที่เลือกนี้จะถูกบันทึกไว้ในไคเร็กทอรีย่อย index ในรูปไฟล์ JBIG และชื่อของไฟล์จะเป็นเลขจำนวนเต็มบวก เริ่มตั้งแต่ 0 และมีค่าเพิ่มขึ้นตามจำนวนของ keyimage

ในการหาว่ามี keyimage ปรากฏอยู่ที่ใดบ้างในเอกสาร โปรแกรมจะใช้วิธีทำแมตซิงระหว่าง keyimage กับหน้าต่างๆของภาพเอกสาร

4.1.4.6 การจัดเก็บเอกสาร

เมื่อจัดสร้างเอกสารเรียบร้อยแล้ว เลือก Save Document จากเมนูไฟล์ โปรแกรมจะบันทึกข้อมูลที่ใส่สร้างเอกสารทั้งหมดในไฟล์ DIO (ย่อจาก Document Image Organizer) ซึ่งสามารถแก้ไขเอกสารได้ในภายหลัง และโปรแกรมจะบันทึกชุดของไฟล์ที่เป็นส่วนประกอบของเอกสาร HTML ลงไปยังโลคอลไคเร็กทอรีที่เลือกไว้

เอกสาร HTML ประกอบด้วยชุดของไฟล์ต่างๆดังต่อไปนี้

ไฟล์ xxxx.htm

- เป็นไฟล์ที่แสดงข้อมูลคุณสมบัติของเอกสาร (เป็นตัวอักษรล้วน)
- xxxx เป็นชื่อของไฟล์ที่อ้างถึงชื่อเอกสารนั้น

- รายละเอียดของไฟล์นี้คือ

```
<HTML>
<HEAD>
  <TITLE> ชื่อเอกสาร </TITLE>
  <META NAME="KeyWords" CONTENT="ชื่อของ keyimage" >
  <META NAME="Author" CONTENT="ชื่อผู้แต่งหนังสือ">
  <META NAME="Description" CONTENT="บทคัดย่อของหนังสือ ">
</HEAD>
<BODY BGCOLOR = "#FFFFFF" TEXT = "#000000" LINK = "#000FFF" ALINK = "#FF0000" >
  ... ข้อมูลของเอกสาร เช่น ชื่อเอกสาร, ชื่อผู้แต่ง, สำนักพิมพ์ ฯลฯ ซึ่งขึ้นกับชนิดของเอกสาร
  ... ไฮเปอร์ลิงค์ไปยังตัวเอกสาร <A HREF="xxxxffff.htm" ></A>
</BODY>
</HTML>
```

ไฟล์ xxxxxxxx.htm

- เป็นตัวเอกสาร โดยแบ่งหน้าต่างของบราวเซอร์ออกเป็นสองเฟรม เฟรมแรกอยู่ทางด้านซ้ายมือชื่อ left มีความกว้าง 20% ทำหน้าที่เป็นส่วนควบคุมการแสดงผลโดยจะอ่านไฟล์ชื่อ xxxxiiii.htm (ตัวเลือกแบบข้อความ) เฟรมที่สองอยู่ทางด้านขวามือชื่อ right มีความกว้าง 80% ทำหน้าที่เป็นส่วนแสดงผลโดยจะอ่านไฟล์ชื่อ xxxccccc.htm (สารบัญของเอกสาร)

- รายละเอียดของไฟล์นี้คือ

```
<HTML>
<HEAD>
<TITLE> ชื่อเอกสาร : ชื่อไฟล์เอกสาร (xxx) </TITLE>
</HEAD>
<FRAMESET COLS=20%,80%>
<FRAME SRC=xxxxiiii.htm name=left margin width=1 scrolling=auto>
<FRAME SRC=xxxxccccc.htm name=right margin width=1 scrolling=auto>
</FRAMESET>
</BODY>
</HTML>
```

ไฟล์ xxxxiiii.htm และ xxxxtttt.htm

- เป็นตัวเลือกแบบข้อความ และแบบภาพชนิด thumbnail ในส่วนควบคุม โดยมีไฮเปอร์ลิงค์ไปยังหน้าต่างๆ หนาสารบัญ หนาดัชนี

ไฟล์ xxxxxxxx.htm

- เป็นสารบัญชของเอกสาร ซึ่งจะมีไฮเปอร์ลิงค์ไปยังหน้าเริ่มต้นของส่วนประกอบต่างๆ ในเอกสาร

ไฟล์ xxxx0001.htm , xxxx0002.htm , ...

- เป็นหน้าของภาพเอกสารต่างๆ
- ชื่อไฟล์ 4 ตัวท้ายเป็นตำแหน่งหน้า โดยเริ่มจาก 0001, 0002, ... , 9999 (จำนวนหน้าของเอกสารสูงสุดเท่ากับ 9,999 หน้า)
- ส่วนหัวของไฟล์ จะมีชื่อ keyimage และเนื้อหาโดยย่อของหน้านั้น ในตัวเอกสาร HTML จะมี EMBED tag สำหรับแสดงข้อมูลภาพ JBIG
- รายละเอียดของไฟล์นี้คือ

```
<HTML>
<HEAD>
<TITLE> ชื่อเอกสาร ตำแหน่งหน้า </TITLE>
  <META NAME="KeyWords" CONTENT="ชื่อของ keyimage" >
  <META NAME="Description" CONTENT="เนื้อหาโดยย่อของหน้า">
</HEAD>
<BODY BGCOLOR = "#FFFFAF" TEXT = "#000000" LINK = "#000FFF" ALINK = "#FF0000" >
<EMBED SRC=page/000.bie TYPE=image/x-jbig
      WIDTH=640 HEIGHT=900
      AUTODOC=TRUE ></EMBED>
<BODY>
</HTML>
```

ไฟล์ xxxxxkkkk.htm

- เป็นดัชนีของเอกสาร ซึ่งจะมีไฮเปอร์ลิงค์ไปยังหน้าเริ่มต้นของส่วนประกอบต่างๆ
- ส่วนหัวของไฟล์ จะมีชื่อ keyimage ทั้งหมดในเอกสาร
- ในตัวเอกสาร HTML จะมี EMBED tag สำหรับแสดงภาพ keyimage ชื่อของ keyimage และมีไฮเปอร์ลิงค์ไปยังหน้าที่มี keyimage นั้นปรากฏอยู่

4.2 ส่วนเรียกคืนภาพเอกสาร

ส่วนเรียกคืนภาพเอกสาร จะใช้โปรแกรม Netscape Navigator เพื่อแสดงภาพเอกสาร HTML และเป็นส่วนติดต่อกับผู้ใช้

ฟังก์ชันการทำงาน และคุณสมบัติสำคัญของส่วนเรียกคืนภาพเอกสาร ได้แก่

1. การค้นหาเอกสาร ทำโดยอาศัย search engine ที่ให้บริการอยู่ในอินเทอร์เน็ต ผู้ใช้สามารถค้นหาเอกสารที่ต้องการได้โดยพิมพ์ชื่อเอกสาร หรือข้อมูลเกี่ยวกับเอกสาร หรือชื่อของ keyimage

2. บrowsable เอกสาร เมื่อได้เอกสารที่ต้องการแล้วผู้ใช้สามารถอ่านเนื้อหาภายในเอกสารได้ โดยอาศัยฟังก์ชันในการ browsed เช่น

- ภาพหน้าของเอกสารก่อนหน้า หรือถัดจากหน้าปัจจุบันที่แสดงอยู่
- ซุมเข้าเพื่อดูรายละเอียดเฉพาะส่วน หรือซุมออกเพื่อดูรายละเอียดทั้งหน้าของเอกสาร
- ภาพหน้าของเอกสารในลักษณะ thumbnail คือ จะเห็นหน้าเอกสารเป็นรูปเล็กๆ หลายๆหน้าที่อยู่ติดกันบนจอภาพเดียว และเมื่อต้องการอ่านรายละเอียดของหน้าไหน ก็สามารถชี้เมาส์คลิกไปที่รูป thumbnail ของหน้านั้นได้

3. ไฮเปอร์ลิงค อาศัยโครงสร้างของเอกสารที่เป็นแบบไฮเปอร์ลิงคจึงสามารถไปยังตำแหน่งต่างๆของเอกสารได้ เช่น ไฮเปอร์ลิงคที่ส่วนควบคุมการแสดงผลภาพ(เฟรมด้านขวาของเอกสาร) ไฮเปอร์ลิงคที่หน้าสารบัญจะไปยังส่วนต่างๆของเอกสาร ไฮเปอร์ลิงคที่หน้าดัชนีจะไปยังหน้าต่างๆของเอกสารที่มี keyimage นั้นๆปรากฏอยู่ Netscape Navigator

ฟังก์ชันในข้อ 1 และ 3 เป็นผลของงานที่ส่วนจัดเก็บภาพเอกสารทำให้แก่ส่วนเรียกคืนภาพเอกสาร โดยสร้างชุดเอกสาร HTML ให้ search engine สามารถค้นหาค่าได้ตามข้อ 1 และมีโครงสร้างไฮเปอร์ลิงคตามข้อ 3 แต่ฟังก์ชัน browsed ในข้อ 2 นั้นเป็นหน้าที่ของส่วนเรียกคืนภาพเอกสารที่เป็นตัวจัดการ ซึ่งจะต้องพัฒนามอดูลอันหนึ่งขึ้นมาเพื่อให้ Netscape Navigator สามารถทำหน้าที่ browsed เอกสารได้ และมอดูลนั้นก็คือ JBIG plug-in

4.2.1 การพัฒนา JBIG plug-in สำหรับ Netscape Navigator

ในการพัฒนา JBIG plug-in ได้ใช้ plug-in SDK (software development kit) [10] ที่บริษัทผู้ผลิตโปรแกรม Netscape ได้เตรียมไว้ให้สำหรับนักพัฒนาโปรแกรมได้ใช้สร้าง plug-in มอดูล โดยใช้ภาษา C และสร้างมอดูลเป็นไดนามิกไลบรารีลิงค (DLL) ชนิด 32 บิต

plug-in ที่ได้มีฟังก์ชันการทำงานและคุณสมบัติดังต่อไปนี้

1. สามารถแสดงข้อมูลภาพ JBIG ทั้งแบบ embedded หรือ full-page ได้
2. การถอดรหัสภาพจะทำขณะที่ Netscape รับข้อมูลจากเครือข่าย และถ้าภาพได้รับการเข้ารหัสเป็นแบบ progressive ก็จะสามารถแสดงภาพที่ความละเอียดต่ำได้ก่อน และจะแสดงภาพที่ความละเอียดสูงขึ้นเรื่อยๆเมื่อรับข้อมูลที่ส่งมาจากเครือข่าย
3. ชุมภาพได้
4. กรณีที่เป็นภาพของหน้าเอกสาร สามารถดูหน้าก่อน หรือถัดจากหน้าปัจจุบันที่แสดงอยู่ได้ และดูภาพในลักษณะ thumbnail คือ สามารถแสดงหน้าที่อยู่ติดกันพร้อมๆกันได้

4.2.2 การติดตั้ง JBIG plug-in

ขั้นตอนในการติดตั้ง JBIG plug-in ให้แก่ส่วนเรียกคืนภาพเอกสาร มีดังนี้

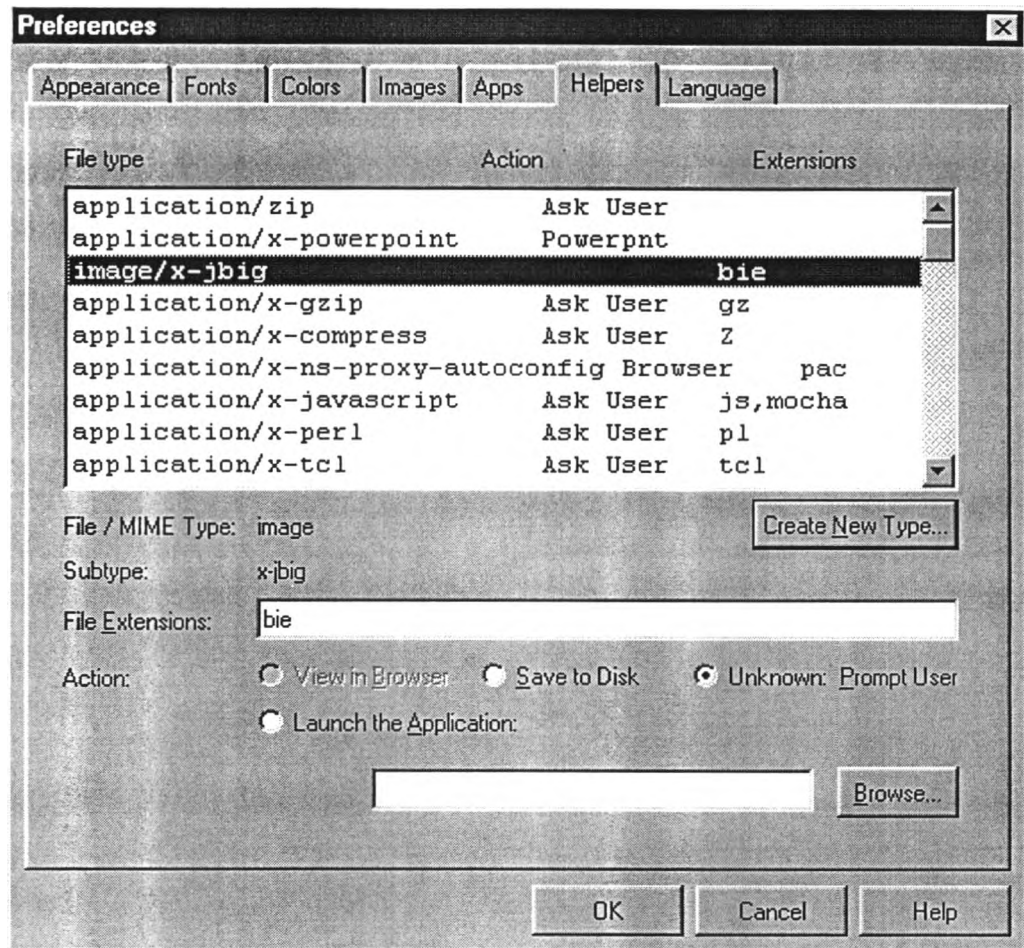
1. สำเนาไฟล์ที่เป็น JBIG plug-in มอดูล (NPJBIG16.DLL สำหรับ Windows 3.1x หรือ NPJBIG32.DLL สำหรับ Windows 95/NT) ไปยังไดเรกทอรี plugins ของโปรแกรม Netscape Navigator

2. เรียกโปรแกรม Netscape Navigator ที่เมนู Options เลือก General Preferences, ที่ไดอะล็อกบ็อกซ์ Preferences เลือกแท็บ Helpers (รูป 4.16) ใส่ชนิดของ MIME type ใหม่โดยกดปุ่ม Create New Type

Mime Type ให้ใส่ image

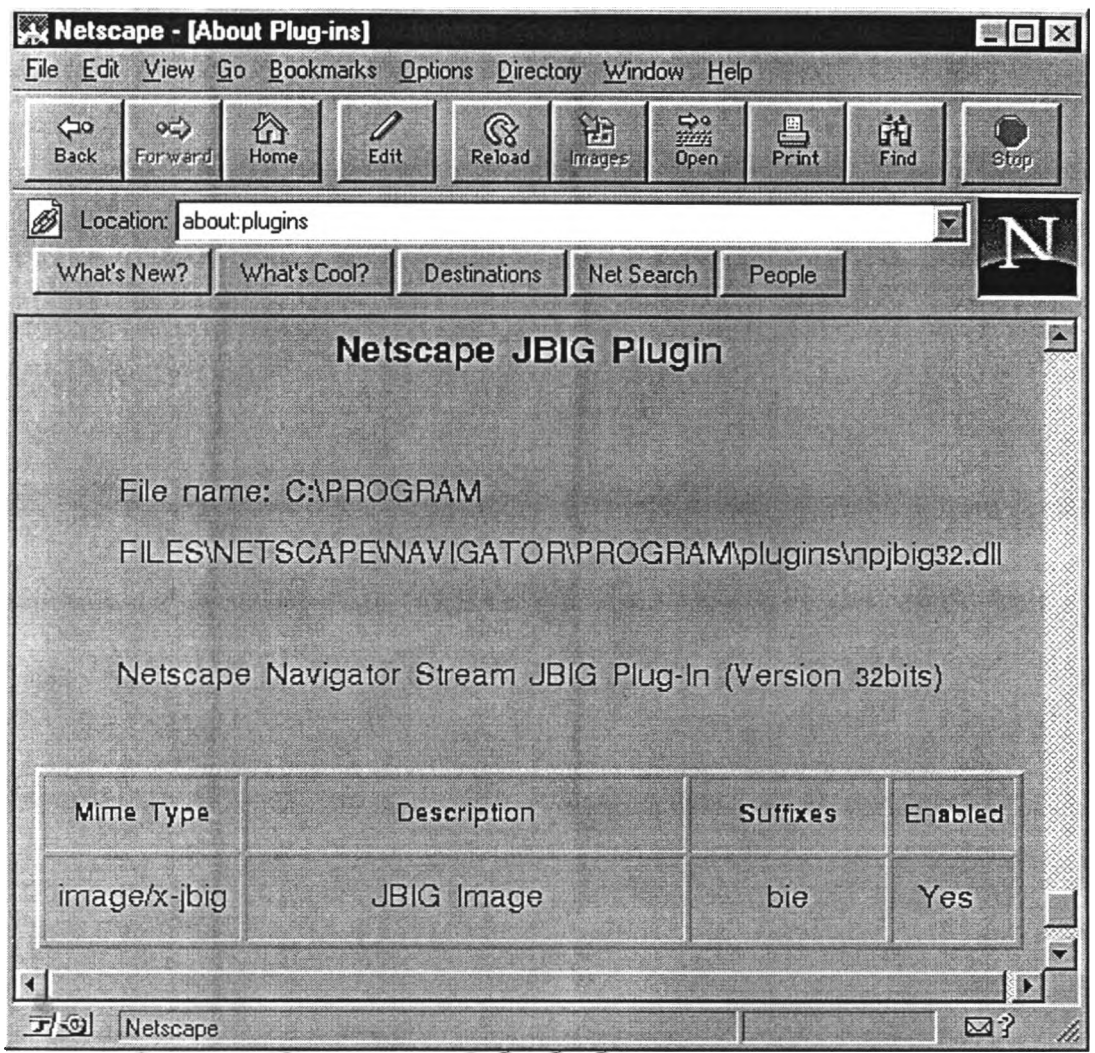
Mime Sub Type ให้ใส่ x-jbig

File Extensions ให้ใส่ bie



รูปที่ 4.16 ไดอะล็อกบ็อกซ์ Preferences ของโปรแกรม Netscape Navigator

3. ตรวจสอบความถูกต้องในการติดตั้งโดยเลือก About Plug-ins ที่ Help เมนู ถ้าติดตั้งถูกต้องจะปรากฏรายชื่อของ JBIG plug-in มอดูล ดังรูป 4.17



รูปที่ 4.17 รายชื่อของ plug-in ที่ได้ติดตั้งให้กับโปรแกรม Netscape Navigator

4.2.3 การใช้งาน JBIG plug-in

JBIG plug-in จะรับข้อมูลที่ส่งผ่านมาจาก EMBED tag ในเอกสาร HTML

```
<EMBED argument></EMBED>
```

และอาร์กิวเมนต์ (argument) ซึ่งใช้ส่งค่าผ่าน EMBED tag มีดังนี้

ตารางที่ 4.5 อาร์กิวเมนต์ที่จำเป็นต้องมีใน EMBED tag

อาร์กิวเมนต์ที่จำเป็นต้องมี	ความหมาย
SRC = "URL"	ตำแหน่งของไฟล์ภาพ JBIG (นามสกุล bie)
WIDTH = "xxx"	กำหนดความกว้างของหน้าต่าง (หน่วยเป็นจุดภาพ)
HEIGHT = "xxx"	กำหนดความสูงของหน้าต่าง (หน่วยเป็นจุดภาพ)
TYPE = "image/x-jbig"	ระบุชนิดของ MIME type เป็น JBIG

ตารางที่ 4.6 อาร์กิวเมนต์อื่นๆ ใน EMBED tag

อาร์กิวเมนต์อื่นๆ	ความหมาย
AUTODOC = "TRUE/FALSE" HREF = "URL" TARGET = "window"	กำหนดโมดมาให้แสดงภาพ หรือภาพเอกสาร ตำแหน่งของไฟล์ที่จะลิงคไปเมื่อใช้เมาส์คลิกไปบนภาพ (เฉพาะ โมดแสดงภาพ) หน้าต่างสำหรับใช้แสดงข้อมูลไฟล์ที่ลิงคไป (เฉพาะ โมด แสดงภาพ)

4.2.4 ฟังก์ชันของ JBIG plug-in

JBIG plug-in จะแบ่งการทำงานเป็น 2 โมดใหญ่ๆ คือ โมดสำหรับแสดงภาพ โดยมองข้อมูลภาพ JBIG (ไฟล์นามสกุล BIE) เป็นเพียงภาพธรรมดาทั่วไป และโมดสำหรับแสดงภาพเอกสาร โดยมองว่าภาพที่แสดงนั้นเป็นหน้าหนึ่งของเอกสาร

4.2.4.1 โมดสำหรับแสดงภาพ

ในโมดนี้สามารถเลือกได้ว่าจะให้แสดงภาพเป็นแบบ full page คือจะใช้เนื้อที่ทั้งหมดของ Netscape แสดงภาพ หรือเป็นแบบ embed อยู่ในเอกสาร HTML

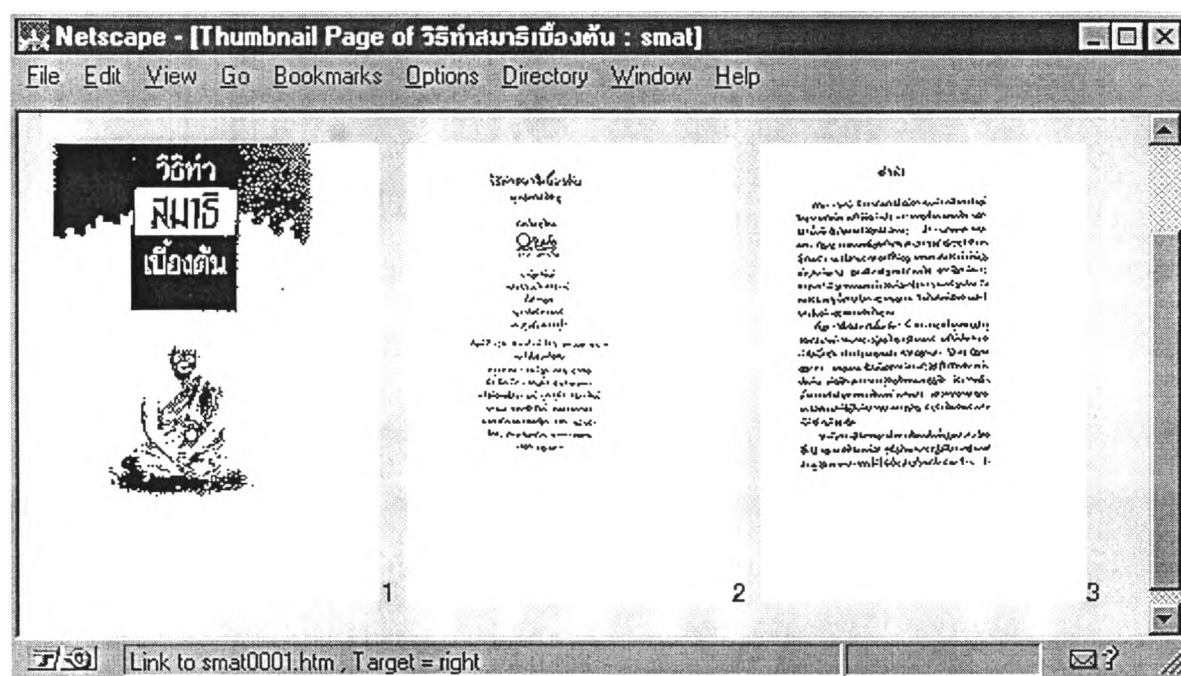
จากรูป 4.18 เป็นตัวอย่างของการแสดงภาพแบบ full page เมื่อไฟล์ b20_g8.bie (เป็นภาพ JBIG) ถูกเรียก Netscape จะโหลด JBIG plug-in และส่งข้อมูลไปบอกว่า โมดของ plug-in เป็น full-page mode สามารถใช้เนื้อที่ทั้งหมดของหน้าต่างได้ ดังนั้น plug-in จึงแสดงภาพที่ความละเอียดสูงสุด และวางตำแหน่งของภาพให้อยู่กึ่งกลางหน้าต่าง

ถ้าภาพ JBIG ได้รับการเข้ารหัสแบบ progressive จะสามารถดูที่ความละเอียดต่ำลงมาได้ โดยใช้เมาส์ปุ่มขวากดไปที่รูป แล้วจึงเลือก Zoom Out จากเมนู pop-up



รูปที่ 4.18 ตัวอย่างการแสดงผลภาพแบบ full page

รูป 4.19 เป็นตัวอย่างของการแสดงผลภาพแบบ embed อยู่ในเอกสาร HTML



รูปที่ 4.19 ตัวอย่างการแสดงผลภาพแบบ embed

และมีรายละเอียดของ EMBED tag ดังนี้

```
<EMBED SRC=page/000.bie TYPE=image/x-jbig
      WIDTH=160 HEIGHT=225
      TARGET=right HREF=smat0001.htm>
</EMBED>
```

Netscape จะสร้างหน้าต่างขนาด WIDTH x HEIGHT ให้กับ plug-in ตามที่กำหนดไว้ใน EMBED tag และ plug-in จะถอดรหัสและแสดงภาพที่ความละเอียดสูงสุดที่ไม่เกินขนาดของหน้าต่างที่กำหนดไว้ และวางตำแหน่งของภาพให้ชิดมุมบนซ้ายของหน้าต่าง แต่ถ้าวางภาพ JBIG มีขนาดใหญ่กว่าหน้าต่าง plug-in จะแสดง scroll bar ให้สามารถเลื่อนดูภาพที่ตำแหน่งต่างๆได้

ถ้าวางภาพ JBIG ได้รับการเข้ารหัสแบบ progressive ผู้ใช้สามารถซูมภาพเพื่อดูที่ความละเอียดต่างๆได้ โดยใช้เมาส์ปุ่มขวากดไปที่รูป แล้วจึงเลือก Zoom In/Out จากเมนู pop-up

เมื่อมีการเลื่อนเมาส์ไปบนภาพ plug-in จะแสดงชื่อไฟล์ที่จะลิงก์ไป พร้อมกับหน้าต่างที่จะแสดงไฟล์นั้นบน status bar ของ Netscape และเมื่อใช้เมาส์ปุ่มซ้ายกดไปที่ภาพ Netscape จะอ่านข้อมูลจาก HREF และส่งไปให้กับหน้าต่างหรือเฟรมชื่อ TARGET

4.2.4.2 โมดสำหรับแสดงภาพเอกสาร

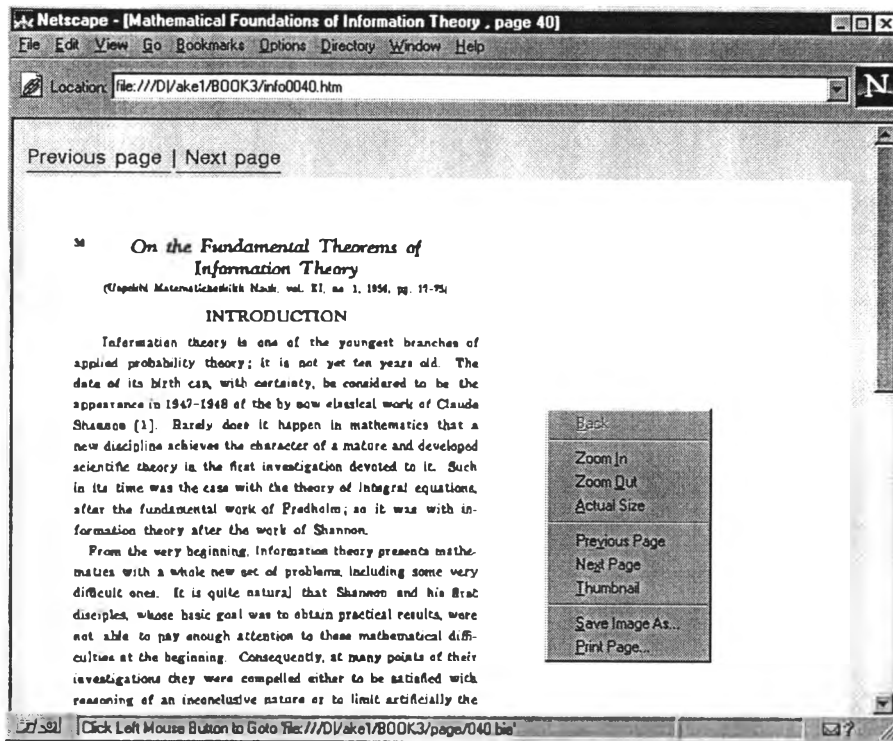
รูป 4.20 เป็นตัวอย่างของโมดแสดงภาพเอกสาร ซึ่งมีรายละเอียดของ EMBED tag ดัง

นี้

```
<EMBED SRC=page/040.bie TYPE=image/x-jbig
      WIDTH=640 HEIGHT=900
      AUTODOC=TRUE >
</EMBED>
```

จาก EMBED tag จะเห็นว่ากำหนดโมดสำหรับแสดงภาพเอกสาร (AUTODOC=TRUE)

การทำงานของโมดนี้ จะคล้ายกับการแสดงภาพแบบ embed แต่จะต่างกันตรงที่กำหนดให้ภาพ JBIG ที่อยู่ในไฟล์ HTML เป็นภาพของหน้า



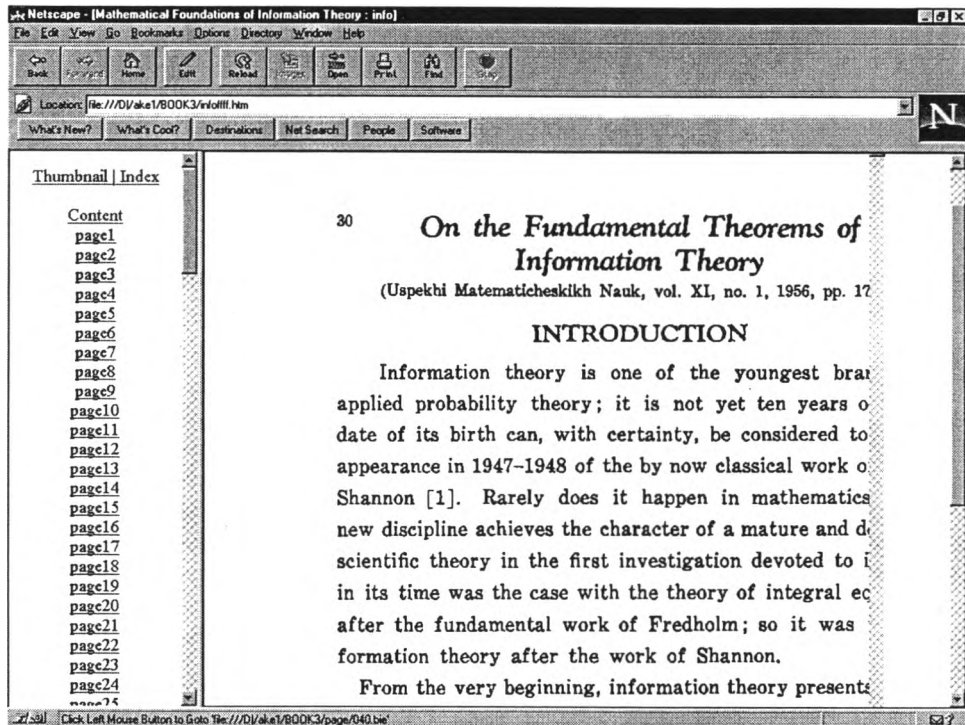
รูปที่ 4.20 ตัวอย่างของโมดแสดงภาพเอกสาร

เมื่อคลิกปุ่มขวาของเมาส์จะปรากฏเมนู pop-up ให้ผู้ใช้เลือกคำสั่งในการบราวซ์เอกสาร คำสั่งต่างๆและผล ได้สรุปไว้ในตาราง 4.7

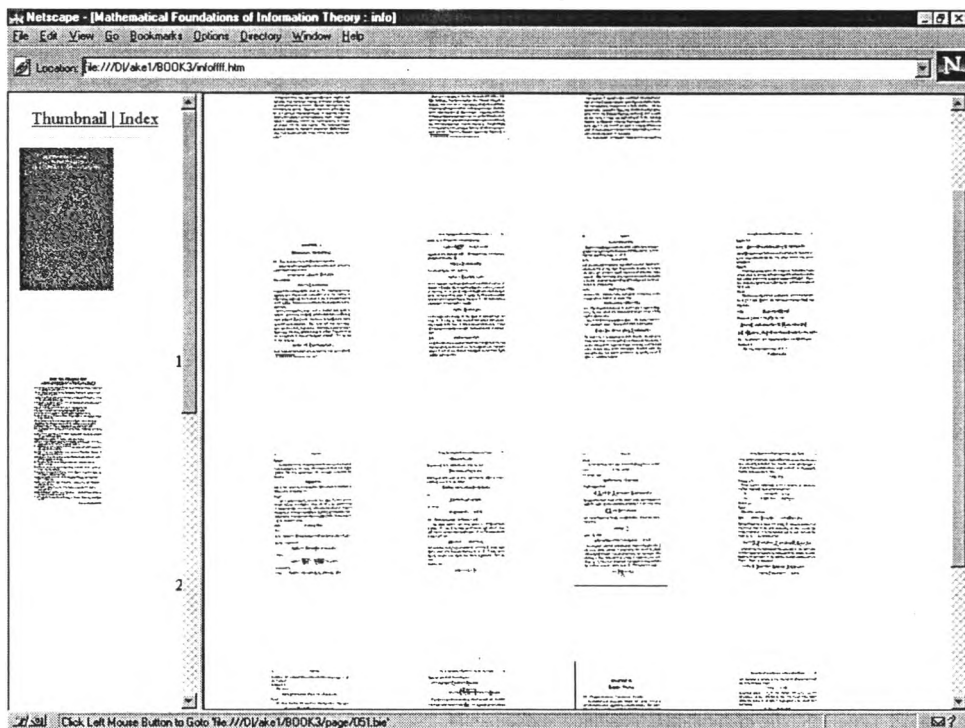
ตารางที่ 4.7 คำสั่งในการบราวซ์เอกสาร

คำสั่ง	การทำงาน
Back	แสดงภาพก่อนหน้า
Zoon In	แสดงภาพที่ความละเอียดมากกว่าเดิม 1 ชั้น
Zoom Out	แสดงภาพที่ความละเอียดน้อยกว่าเดิม 1 ชั้น
Actual Size	แสดงภาพที่ตัวประกอบของการซูมเท่ากับ 1
Previous Page	แสดงภาพเอกสารก่อนหน้า
Next Page	แสดงภาพเอกสารถัดไป
Thumbnail	แสดงภาพที่ตัวประกอบของการซูมเท่ากับ 4
Save Image As	บันทึกข้อมูลภาพลงในดิสก์
Print Page	พิมพ์ภาพผ่านทางเครื่องพิมพ์

- เมื่อมีการเลื่อนเมาส์ไปบนภาพหน้าของเอกสาร จะคำนวณตำแหน่งของเมาส์ และแสดงชื่อของไฟล์ภาพที่แสดงอยู่ ณ ตำแหน่งนั้น บน status bar ของ Netscape และเมื่อกดปุ่มซ้ายของเมาส์ไปที่หน้าเอกสารใด plug-in จะแสดงภาพหน้าของเอกสารนั้นที่ตัวประกอบการซูมเท่ากับ 1



รูปที่ 4.21 แสดงการซูมภาพเข้า



รูปที่ 4.22 การแสดงภาพแบบ thumbnail

4.3 ตัวอย่างเอกสาร

ได้ทดลองจัดสร้างเอกสาร HTML จากหนังสือขนาดกว้าง 5.25 นิ้ว สูง 8 นิ้ว สแกนด้วยความละเอียด 300 จุดต่อนิ้ว จำนวนหน้าทั้งหมดที่สแกนเท่ากับ 134 หน้า ภาพทั้งหมด มีขนาดกว้าง 1,550 จุดภาพ สูง 2,350 จุดภาพ และเก็บเป็นแบบภาพสองระดับ(ขาว-ดำ)

แปลงไฟล์ภาพทั้งหมดให้อยู่ในรูปแบบ JBIG โดยเข้ารหัสแบบ progressive ให้มีจำนวน layer ทั้งหมด 6 layer

ในการจัดสร้างได้ใส่ข้อมูลของเอกสารซึ่งมีรายละเอียดตามรูป 4.23

Document Information#1	
Type	Book
Title	Mathematical Foundations of Information Theory
Authors	A.I.Khinchin
Publisher	Dover Publication, Inc.
Date	1957
Language	English
Abstract	Information theory is one of the youngest branches
No. of Pages	134
HTML File	info.htm
Keyimages	7

รูปที่ 4.23 หน้าต่างแสดงรายละเอียดของเอกสาร

เอกสารประกอบด้วยส่วนต่างๆ ดังนี้ Front Cover , Content , Part 1 , Part 2 , Introduction , Chapter 1 , Chapter 2 , Chapter 3 , Chapter 4 , Chapter 5 , Conclusion , Reference และ Back Cover

ดัชนีของเอกสารประกอบด้วย keyimage ที่เลือกมาจำนวน 7 ภาพ ดังรูป 4.24

uncertainty	<u>uncertainty</u>
Markov chains	<u>Markov chains</u>
$H(A) = -\sum_{i=1}^n p(A_i) \lg p(A_i)$	<u>entropy of the space A</u>
McMillan's Theorem	<u>McMillan's theorem</u>
Doob's Theorem	<u>Doob's theorem</u>
first Shannon	<u>the first Shannon theorem</u>
second Shannon	<u>the second Shannon theorem</u>

รูปที่ 4.24 ภาพและชื่อของ keyimage

4.3.1 เนื้อที่ในการเก็บข้อมูล

เอกสารที่สร้างได้จะประกอบด้วย ไฟล์เอกสาร HTML จำนวน 140 ไฟล์ มีขนาดรวม 88,591 ไบต์ ไฟล์ภาพเอกสารจำนวน 134 ไฟล์ มีขนาดรวม 3,307,608 และไฟล์ภาพ keyimage จำนวน 7 ไฟล์ มีขนาดรวม 3,721 ไบต์ ซึ่งเมื่อรวมไฟล์ทั้งหมดแล้วจะมีขนาด 3,399,920 ไบต์ (3.24 เมกะไบต์)

เมื่อเปรียบเทียบเฉพาะขนาดข้อมูลภาพเอกสาร (ตามตาราง 4.8) พบว่า JBIG สามารถลดปริมาณข้อมูลที่ใช้เก็บภาพลงได้มาก

ตารางที่ 4.8 เปรียบเทียบขนาดของไฟล์ที่เก็บอยู่ในรูปแบบต่างๆ

	RAW	BIE		GIF (LZW)		TIF (FAX G.3)	
	size	size	ratio	size	ratio	size	ratio
134 ภาพ	61,090,600	3,307,608	18.47	8,646,527	7.07	8,856,950	6.90

4.3.2 ชนิดของภาพเอกสาร

โดยทั่วไปแล้วจะเก็บภาพของหน้าเอกสารแบบสองระดับ (1 บิตต่อจุดภาพ) แต่อาจจะสูญเสียข้อมูลบางส่วนไป ผู้จัดทำอาจจะเลือกเก็บเป็นแบบระดับเกรย์ 2-3 บิตต่อจุดภาพได้ แต่การเพิ่มจำนวนบิตเพื่อเก็บจุดภาพจะทำให้ขนาดไฟล์ภาพเพิ่มขึ้น และอัตราการบีบอัดข้อมูลของ JBIG จะลดลงตามจำนวนบิตที่เพิ่มขึ้น

ตัวอย่างต่อไปนี้ได้ทดลองเก็บภาพเอกสารเป็นแบบ 3 บิตต่อจุดภาพ โดยนำบทหนึ่งของหนังสือที่จัดสร้างมาสแกนใหม่ โดยยอมลดความละเอียดของภาพจาก 300 จุดต่อนิ้ว ลงเหลือ 100 จุดต่อนิ้ว จำนวนหน้าทั้งหมดที่สแกนคือ 12 หน้า ภาพทั้งหมดมีขนาดกว้าง 512 จุดภาพ สูง 776 จุดภาพ เข้รหัสภาพโดยใช้เกรย์โคดร่วมกับ JBIG ให้เป็นแบบ progressive โดยมีจำนวน layer ทั้งหมด 3 layer

จากตารางที่ 4.9 จะเห็นว่าภาพมีขนาดภาพลง 9 เท่า เพื่อแลกกับจำนวนบิตที่ใช้เก็บเก็บจุดภาพเพิ่มขึ้น 3 เท่า จะมีขนาดใกล้เคียงกับภาพขนาดเดิมที่เก็บแบบ 1 บิตต่อจุดภาพ

ตารางที่ 4.9 เปรียบเทียบขนาดของไฟล์ JBIG ที่เก็บอยู่ในรูปแบบต่างๆ

	300 จุดต่อนิ้ว 1 บิตต่อจุดภาพ			100 จุดต่อนิ้ว 3 บิตต่อจุดภาพ		
	RAW	size	ratio	RAW	size	ratio
12 ภาพ	5,470,800	264,650	20.67	1,787,904	241,689	7.4

This article represents a first
In writing it, I relied mainly on
mational Theory of Communicat
treatment is not always sufficien
cally correct, so that besides ha
practical details, in many instanc
both the statement of definition
of theorems. There is no doubt
study of entropy will become a
theory; the work I have done s
stage in the development of this

ก.

This article represents a first
In writing it, I relied mainly on
mational Theory of Communica
treatment is not always sufficie
cally correct, so that besides ha
practical details, in many instanc
both the statement of definition
of theorems. There is no doubt
study of entropy will become a
theory; the work I have done s
stage in the development of thi

ข.

รูปที่ 4.25 ภาพเอกสารที่ได้จากการสแกนด้วยความละเอียดและจำนวนบิตต่างกัน

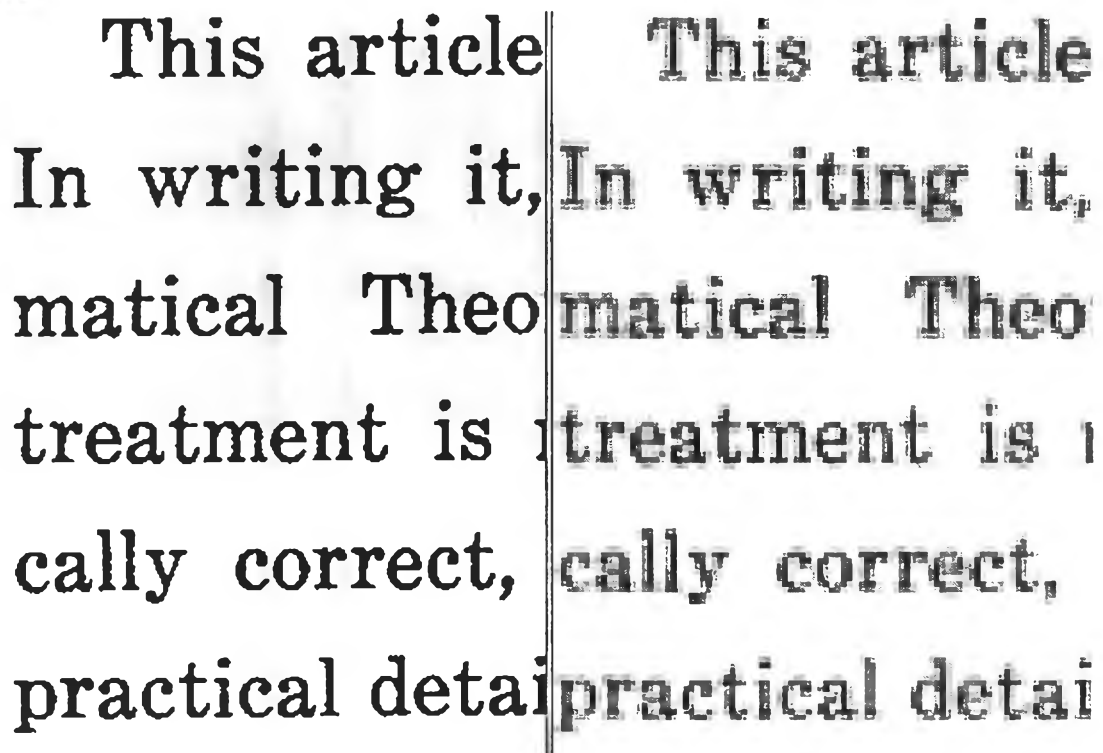
(ก) 300dpi 1bpp (ข) 100dpi 3bpp และ (ค) 100dpi 1 bpp

This article represents a first
 In writing it, I relied mainly on
 mathematical Theory of Communication
 treatment is not always sufficiently
 cally correct, so that besides handling
 practical details, in many instances
 both the statement of definitions
 of theorems. There is no doubt
 study of entropy will become a
 theory; the work I have done is
 a stage in the development of this

ก.

รูปที่ 4.25 (ต่อ)

รูป 4.25 เปรียบเทียบภาพเอกสารที่ได้จากการสแกนด้วยความละเอียดและจำนวนบิตต่าง
 กัน



ก.

ข.

รูปที่ 4.26 ภาพเอกสารขนาดสูง 3 เท่า (ก) 300dpi 1bpp (ข) 100dpi 3bpp

รูป 4.26 แสดงให้เห็นว่าถึงแม้จะเก็บภาพด้วยความละเอียดเพียง 100 จุดต่อนิ้ว แต่ใช้ 3 บิต
 ในการเก็บหนึ่งจุดภาพ ยังคงให้ข้อมูลที่เพียงพอ และเป็นทางเลือกหนึ่งเพื่อเก็บภาพไว้ใช้ทำ OCR ใน
 ภายหลัง