การเข้ารหัสและการถอดรหัสรูปภาพดิจิตอลโดยใช้ ไฟไนต์ออโตมาตาแบบถ่วงน้ำหนัก

DIGITAL IMAGE ENCODING AND DECODING USING WEIGHTED FINITE AUTOMATA

ธมรัตน์ เลขะวิจิตเลิศ
THOMMARAT LAKHAVIJITLERT

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2549

การเข้ารหัสและการถอดรหัสรูปภาพดิจิตอลโดยใช้ไฟไนต์ออโตมาตาแบบถ่วงน้ำหนัก

DIGITAL IMAGE ENCODING AND DECODING USING WEIGHTED FINITE AUTOMATA

ธมรัตน์ เลขะวิจิตเลิศ

THOMMARAT LAKHAVIJITLERT

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิทยาการคอมพิวเตอร์

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2549

DIGITAL IMAGE ENCODING AND DECODING USING WEIGHTED FINITE AUTOMATA

THOMMARAT LAKHAVIJITLERT

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF SCIENCE IN COMPUTER SCIENCE
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2006

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ การเข้ารหัสและการถอดรหัสรูปภาพดิจิตอลโดยใช้ไฟไนต์ออโตมาตาแบบถ่วงน้ำหนัก

Digital Image Encoding and Decoding Using Weighted Finite Automata

ชื่อนักศึกษา นายธมรัตน์ เลขะวิจิตเลิศ

รหัสประจำตัว 46063613

ปริญญา วิทยาศาสตรมหาบัณฑิต

สาขาวิชา วิทยาการคอมพิวเตอร์

อาจารย์ที่ปรึกษาวิทยานิพนธ์ ผศ.ดร.กรกช ประชุมรักษ์

คณะกรรมการสอบวิทยานิพนธ์		**ลายมือชื่อ**
ผศ.ดร.จีรพร	ศรีสวัสดิ์	
รศ.ดร.วีระ	บุญจริง	
ดร.เฉลิมศักดิ์	เลิศวงศ์เสถียร	
ผศ.ดร.กรกช	ประชุมรักษ์	

วัน / เดือน/ ปี ที่สอบ 9 ตุลาคม 2549 เวลา 13.00-16.00 น.

สถานที่สอบ ณ อาคารจุฬาภรณวลัยลักษณ์ 1 ห้อง 219

บัณฑิตวิทยาลัยรับรองแล้ว

(ผศ.ดร.จารุวัตร เจริญสุข)

คณบดีบัณฑิตวิทยาลัย

วันที่......๒๓......**เดือน**....ธันวาคม......**พ.ศ.**..๒๕๔๙....

หัวข้อวิทยานิพนธ์	การเข้ารหัสและการถอดรหัสรูปภาพดิจิตอลโดยใช้ไฟไนต์ออโตมาตาแบบถ่วงน้ำหนัก
นักศึกษา	นายธมรัตน์ เลขะวิจิตเลิศ
รหัสประจำตัว	46060613
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	วิทยาการคอมพิวเตอร์
พ.ศ.	2549
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ผศ. ดร. กรกช ประชุมรักษ์

บทคัดย่อ

การเข้ารหัสรูปภาพแบบไฟไนต์ออโตมาต้าแบบถ่วงน้ำหนัก (Weighted Finite Automata : WFA) เป็นกระบวนการเข้ารหัสรูปภาพซึ่งพัฒนาโดย Culik และ Kari ซึ่งในขั้นตอนการเตรียมรูปภาพเพื่อจะต้องแบ่งรูปภาพดิจิตอลออกเป็นรูปย่อยๆ สี่ส่วนที่เรียกว่า ควอดทรี (quadtree) และนำแต่ละรูปย่อยมาสร้างเป็นสมการผลรวมเชิงเส้น เมื่อคำนวณหาคำตอบของสมการออกมาได้ก็จะนำมาเป็นค่าถ่วงน้ำหนักของออโตมาต้า แต่ในรูปภาพทั่วๆ ไปแล้ว บางรูปอาจจะไม่สามารถแบ่งออกเป็นสี่ส่วนได้ ดังนั้นในงานวิจัยนี้จึงได้ประยุกต์การบีบอัดแบบไฟไนออโตมาต้าแบบถ่วงน้ำหนักมาใช้กับการแบ่งแบบ 9 ส่วน แทนที่จะใช้การแบ่งแบบสี่ส่วนของ Culik และ Kari

Thesis Title	Digital Image Encoding and Decoding Using Weighted Finite Automata
Student	Mr. Thommarat Lakhavijitlert
Student ID.	46060613
Degree	Master of Science
Programe	Computer Science
Year	2006
Thesis Advisor	Asst. Prof. Dr. Korakot Prachumrak

ABSTRACT

Weight Finite Automata (WFA) image encoding is a method for encoding images which has been brought up by Culik and Kari. They suggest the way to encode a digital image by applying quadtree partition to divide an image into subsquares and then construct subdividing images with linear combination as a wighted automaton. Nonatree is a new way to patition an image into a nine subsquare-tree. Instead of quadtree partition which is used by Culik and Kari, nonatree partition is applied with WFA to encode digital images.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้อย่างดี ด้วยคำแนะนำ และคำปรึกษาจาก ผศ.ดร. กรกช ประชุมรักษ์ ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์ ที่กรุณาให้คำปรึกษา แนวคิดและแนะนำในการดำเนินงานวิจัย ตลอดจบแก้ปัญหาต่างๆ อันเป็นประโยชน์ต่องานวิจัยนี้เป็นอย่างดี ข้าพเจ้ารู้สึกทราบซึ้งในความอนุเคราะห์จากอาจารย์และขอขอบพระคุณเป็นอย่างสูง

ผู้วิจัยขอกราบขอบพระคุณต่อ ผศ.ดร. จีรพร ศรีสวัสดิ์ ซึ่งให้ความกรุณามาเป็นประธานกรรมการสอบ ผศ.ดร. วีระ บุญจริง ซึ่งให้ความกรุณามาเป็นตัวแทนกรรมการจากภาควิชา และ ดร. เฉลิมศักดิ์ เลิศวงศ์เสถียร ซึ่งให้ความกรุณามาเป็นตัวแทนกรรมการจากบุคคลภายนอกทำให้ได้รับคำปรึกษาและคำแนะนำต่างๆ ในการสอบวิทยานิพนธ์

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ในภาควิชาคณิตศาสตร์ประยุกต์และวิทยาการคอมพิวเตอร์ สถาบันเทคโนโลยี พระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา

ขอขอบคุณบัณฑิตศึกษาและบัณฑิตวิทยาลัย คณะวิทยาศาสตร์ให้ความช่วยเหลือ ในเรื่องต่างๆ

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจ และให้การสนับสนุนในทุกเรื่องๆ ทำให้ข้าพเจ้าสามารถทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน

ธมรัตน์ เลขะวิจิตเลิศ

สารบัญ

หน้า

สารบัญ(ต่อ)

หน้า

สารบัญตาราง

ตารางที่ หน้า

สารบัญรูป

รูปที่ หน้า

สารบัญรูป(ต่อ)

รูปที่ **หน้า**

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การสื่อสารข้อมูลสามารถทำได้หลายรูปแบบ ทั้งที่อยู่ในรูปแบบของข้อความ รูปภาพ เสียง หรืออยู่ในรูปแบบที่เป็นภาพยนตร์ เนื่องจากมีคำพูดที่ว่า "ภาพหนึ่งภาพสามารถแทนได้หลายคำพูด" จึงมีความนิยมที่จะเก็บเหตุการณ์ต่างๆ อยู่ในลักษณะของรูปภาพ ซึ่งมีทั้งภาพถ่ายที่พิมพ์ลงกระดาษอัดรูป หรือเก็บอยู่ในรูปของรูปภาพดิจิตอล

ในการเก็บรูปภาพดิจิตอลจำเป็นที่จะต้องเก็บลงอุปกรณ์บันทึกข้อมูลที่มีลักษณะเป็นอุปกรณ์อิเล็กทรอนิคส์ ซึ่งก็มีอยู่หลายรูปแบบด้วยกัน ซึ่งทุกๆ รูปแบบจะมีการจำกัดจำนวนเนื้อที่ของอุปกรณ์ที่จะบันทึกข้อมูลไว้ ซึ่งมักจะทำให้เกิดปัญหาที่ว่าอุปกรณ์ที่ใช้บันทึกข้อมูลเหล่านี้ไม่สามารถจัดเก็บข้อมูลได้เนื่องจากเนื้อที่ที่จะใช้ในการเก็บข้อมูลไม่เพียงพอ

แนวทางในการแก้ปัญหานี้อาจจะแบ่งได้สองหนทางคือการพัฒนาอุปกรณ์บันทึกข้อมูลเพื่อให้สามารถบันทึกข้อมูลให้ได้มากๆ แต่วิธีนี้อาจจะต้องพัฒนาในหลายด้านพร้อมๆ กัน ทั้งในการพัฒนาวัสดุที่จะใช้ทำสื่อบันทึก พัฒนาเทคโนโลยีที่ใช้ในการสร้างสื่อบันทึก ซึ่งอาจจะทำให้สื่อบันทึกที่มีขนาดเล็ก ที่สามารถบันทึกข้อมูลได้มาก แต่ก็จะทำให้ได้ราคาที่แพงยิ่งขึ้น จึงเป็นอุปสรรคที่จะนำสื่อบันทึกนี้มาใช้ แต่ถึงแม้ว่าจะสามารถบันทึกรูปภาพดิจิตอลได้มากขึ้น แต่ปัญหาอีกอย่างหนึ่งที่ก็คือการถ่ายโอนข้อมูลระหว่างอุปกรณ์ ถึงแม้จะไม่เกี่ยวข้องกับการเก็บข้อมูลแต่ก็เป็นปัจจัยที่สำคัญมาก เพราะเนื่องจากขนาดของข้อมูลยังคงใช้เนื้อที่เท่าเหมือนเดิม ดังนั้นถ้าไม่พัฒนาเทคโนโลยีทางด้านการโอนถ่ายข้อมูลอาจทำให้เกิดปัญหาใช้เวลานานในการโอนถ่ายข้อมูล

ส่วนอีกหนทางอย่างหนึ่งที่จะใช้ในการแก้ปัญหานี้ก็คือ การพัฒนาวิธีการบีบอัดรูปภาพดิจิตอลเพื่อให้ใช้เนื้อที่ที่จะใช้บันทึกมีขนาดลดลง โดยที่สามารถใช้สื่อบันทึกขนาดเดิมหรือลักษณะเดิมได้แต่จะสามารถเก็บปริมาณข้อมูลได้มากขึ้นเนื่องจากมีการเข้ารหัสไว้เพื่อที่จะได้ใช้เนื้อที่ในการเก็บข้อมูลได้น้อยลง และทำให้เวลาในการโอนถ่ายข้อมูลได้เร็วขึ้นเนื่องจากปริมาณข้อมูลที่จะโอนถ่ายมีขนาดน้อยลง ด้วยเหตุนี้จึงทำให้เกิดการงานวิจัยนี้ขึ้นเพื่อให้การบีบอัดรูปภาพมีประสิทธิภาพยิ่งขึ้น เพื่อที่จะสามารถนำไปใช้ได้

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

วิทยานิพนธ์นี้มีวัตถุประสงค์พัฒนาวิธีการแบ่งรูปภาพสำหรับวิธีออโตมาตาแบบถ่วงน้ำหนักให้มีประสิทธิภาพยิ่งขึ้น โดยมีแนวความคิดมาจากวิธีการแบบดั้งเดิมที่ใช้หลักการแบ่งภาพแบบสมมาตร หรือแบบควอดทรี (quandtree) แต่ในความจริงภาพทั่วๆไปไม่ได้มีความสมมาตรแบบควอดทรีทั้งหมด การวิจัยนี้จึงนำเสนอแนวทางในการแบ่งรูปภาพให้ความสมมาตรแบบอื่น (โนนาทรี (nonatree)) ซึ่งจะทำให้ใช้พื้นที่ในการเก็บรูปภาพที่เข้ารหัสแล้วน้อยลงกว่าเดิม

1.3 สมมติฐานของการศึกษา

ในการแบ่งรูปภาพออกเป็นรูปย่อยแบบเดิมจะแบ่งเป็นสี่ส่วน ซึ่งในการแบ่งบางครั้งเมื่อพิจารณาแต่ละรูปย่อยเมื่อทำการแบ่งแบบต่างๆ แล้ว จะพบว่าถ้าหากทำการแบ่งแบบอื่น จะทำให้สามารถได้รูปย่อยที่เหมาะสมกว่าการแบ่งแบบสี่ส่วน

ดังนั้นงานวิจัยนี้จึงจะทำการทดลองเปรียบเทียบระหว่างการแบ่งแบบสี่ส่วน (Quadtree) และการแบ่งแบบเก้าส่วน (Nonatree) โดยมีสมมติฐานว่าสำหรับรูปภาพบางแบบบางรูปนั้นถ้าหากทำการแบ่งแบบเก้าส่วนแล้ว จะเหมาะสมกว่าการแบ่งแบบสี่ส่วน โดยทำให้ภาพมีขนาดเล็กลงหลังจากทำการบีบอัด ซึ่งผลที่ได้จะพิจารณาจากที่ว่าพื้นที่ในการจัดเก็บข้อมูลจะน้อยลง พร้อมทั้งค่าความผิดพลาดหรือค่าความเพี้ยนของความเข้มแสงของแต่ละพิกเซลและใช้เวลาในการเข้ารหัสและถอดรหัสไม่ต่างจากภาพเดิม

1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย

ในการเข้ารหัสและถอดรหัสรูปภาพดิจิตอลด้วยไฟไนต์ออโตมาตาแบบถ่วงน้ำหนักจะต้องอาศัยความรู้พื้นฐานทางด้านผลรวมเชิงเส้น พร้อมทั้งการหาผลเฉลยแบบต่างๆ ไปใช้ในการสร้างไฟไนต์ออโตมาตาแบบถ่วงน้ำหนัก เพื่อที่จะใช้ไฟไนต์ออโตมาตาที่สร้างขึ้นมานี้ไปใช้คำนวณหาค่าความเข้มแสงของแต่ละพิกเซลออกมา ส่วนอัลกอริธึมที่ใช้ในการเข้ารหัสจะใช้อัลกอริธึมที่มีกระบวนการที่คล้ายคลึงกันจะแตกต่างกันในส่วนของในการรูปภาพ ซึ่งในการแบ่งรูปภาพนี้จะมีส่วนทำให้ระบบสมการผลรวมเชิงเส้นนี้จะมีตัวแปรเพิ่มมากขึ้น ซึ่งจะทำให้เกิดทางเดินระหว่างโหนดของไฟไนต์ออโตมาตาแบบถ่วงน้ำหนักมากขึ้นแต่จำนวนของโหนดน้อยลง ซึ่งอาจทำให้เวลาในการถอดรหัสมากขึ้นได้ แต่ผลลัพธ์ที่จะได้ดียิ่งขึ้นก็คือพื้นที่ในการจัดเก็บข้อมูลได้น้อยลง

1.5 การเปรียบเทียบระหว่างวิธีการที่นำเสนอกับวิธีการแบบพื้นฐาน

ในการเปรียบเทียบเพื่อหาค่าทางสถิติจะมีการเปรียบเทียบค่าซึ่งจะแบ่งเป็น 2 ชุด โดยชุดแรกจะเป็นการเปรียบระหว่างรูปต้นฉบับที่ทำการเข้ารหัสกับรูปที่ทำการถอดรหัสแล้ว เพื่อที่จะหาค่าความเพี้ยนของความเข้มแสงด้วยวิธีการหาค่าความผิดพลาดของรากกำลังสองของค่าเฉลี่ยกำลังสอง (root mean quare error) ของการแบ่งทั้งสองแบบ และการเปรียบเทียบค่าอีกหนึ่งชุดที่เหลือก็คือการเปรียบเทียบขนาดของภาพที่ถูกบีบอัดด้วยแต่ละวิธีและเวลาในการเข้ารหัสถอดรหัสของการแบ่งรูปภาพทั้งสองแบบ

1.6 ขอบเขตการวิจัย

ขอบเขตของวิทยานิพนธ์ฉบับนี้มีดังต่อไปนี้

1) งานวิจัยนี้จะทดสอบกับรูปภาพดิจิตอลที่มีขนาด $3^n \times 3^n$ พิกเซล และ $2^m \times 2^m$ พิกเซล เมื่อ m, n = 0, 1, 2, ... เท่านั้น
2) รูปภาพต้นฉบับจะใช้เป็นรูปภาพแบบบิทแมพ (Bitmap)
3) ประสิทธิภาพของวิธีการบีบอัดข้อมูลจะวัดจากปริมาณการใช้เนื้อที่ในการบันทึกและอัตราการผิดเพี้ยนของความเข้มแสงของแต่ละพิกเซลเมื่อเทียบกับรูปต้นฉบับ
4) รูปภาพที่ได้จากการบีบอัดด้วยงานวิจัยนี้จะถูกนำไปเปรียบเทียบกับวิธีการบีบอัดด้วยออโตมาตาแบบถ่วงน้ำหนักที่มีการแบ่งรูปภาพต้นฉบับแบบควอดทรี ทางด้านอัตราการใช้พื้นที่และอัตราการผิดเพี้ยนของความเข้มแสง
5) รูปภาพที่จะใช้ทดสอบจะเป็นรูปภาพขาวดำแบบง่ายๆ ที่ใช้ในการทดสอบเกี่ยวกับการบีบอัดรูปภาพดิจิตอล

1.7 ขั้นตอนของการศึกษา

1) ศึกษาเทคนิคการบีบอัดรูปภาพแบบต่างๆ
2) ตั้งข้อสมมติฐานในการทดลอง
3) ออกแบบโครงสร้างและขั้นตอนการบีบอัดรูปภาพด้วยวิธีการบีบอัดที่ศึกษามา
4) ศึกษาภาษาโปรแกรมที่ใช้พัฒนาโปรแกรมต้นแบบ
5) พัฒนาโปรแกรมที่ใช้บีบอัดรูปภาพ
6) ทดสอบบีบอัดรูปภาพด้วยโปรแกรมต้นฉบับ

7) วิเคราะห์ผลการทดลอง โดยการเปรียบเทียบขนาดของพื้นที่ที่ใช้จัดเก็บข้อมูล และความผิดเพี้ยนของเข้มแสงแต่ละพิกเซลของรูปที่ทำการบีบอัดด้วยวิธีออโตมาตาแบบต่างๆ กับรูปต้นฉบับ และเวลาที่ใช้ในการเข้ารหัสและถอดรหัส
8) ประเมินผล พร้อมทั้งสรุปผลการทดลอง
9) เขียนวิทยานิพนธ์

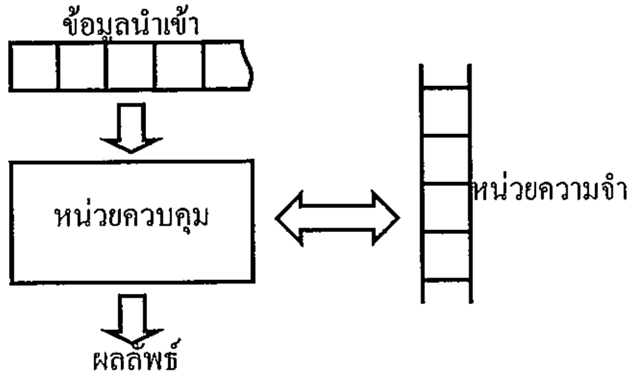
บทที่ 2
เทคนิคการแบ่งรูปภาพและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงรายละเอียดของออโตมาตาแบบต่างๆ ที่นำมาใช้ประยุกต์เกี่ยวกับงานวิจัยนี้เพื่อใช้เป็นพื้นฐานสำหรับการถอดรหัสของรูปภาพดิจิตอล และในส่วนของการเข้ารหัสรูปภาพดิจิตอลก็จะแบ่งเป็นเทคนิคการแบ่งรูปภาพดิจิตอล และการแก้ปัญหาสมการเชิงเส้นแบบต่างๆ

2.1 ออโตมาตา (Automata)

ออโตมาตาเป็นโมเดลทางคณิตศาสตร์ของเครื่องจักรที่มีสถานะจำกัด (Finite state machine) เครื่องจักรที่มีสถานะจำกัดคือเครื่องจักรที่เมื่อรับข้อมูลเข้ามาแล้วจะมีการกระโดดหรือย้ายไปมาระหว่างสถานะต่างๆที่ได้กำหนดไว้ในฟังก์ชันการเปลี่ยนแปลงหรือฟังก์ชันทางผ่าน (transition funtion) ซึ่งเป็นหัวใจสำคัญสำหรับออโตมาตา เพราะเก็บข้อมูลนำเข้าและสถานะถัดไปของแต่ละสถานะเอาไว้ ข้อมูลที่ป้อนเข้าไปในเครื่องจักรจะถูกอ่านทีละตัวอักษรจนกระทั่งอ่านครบทุกตัว ยกตัวอย่างเช่น ถ้าหากข้อมูลที่ถูกป้อนเข้าไปเป็นข้อความ ข้อความนี้จะถูกแบ่งเป็นตัวอักษรแล้วจึงจะนำเข้าไปในออโตมาตาทีละตัวอักษรซึ่งจะอ่านเฉพาะข้อมูลที่ได้รับการยอมรับว่าเป็นชนิดข้อมูลที่ออโตมาตายอมรับ โดยจะอ่านจากทางด้านซ้ายไปยังทางด้านขวาจนกระทั่งครบทุกตัวอักษรที่อยู่ในข้อความ เมื่อข้อมูลถูกอ่านจนจบออโตมาตาก็จะหยุดทำงาน ในสถานะสุดท้ายของออโตมาตานั้น ถ้าเป็นสถานะที่ยอมรับได้ ก็จะกล่าวว่าออโตมาตายอมรับข้อมูลนี้ แต่ถ้าหากอยู่ในสถานะที่ยอมรับไม่ได้ ก็จะถือว่าไม่ยอมรับข้อมูลนี้

เนื่องจากตัวออโตมาตาเป็นเครื่องจักรดังนั้น ออโตมาตาจึงจำเป็นจะต้องมีอุปกรณ์หน่วยความจำ (storage device) ไว้สำหรับเก็บข้อมูลนำเข้าซึ่งแต่ละหน่วยของอุปกรณ์หน่วยความจำจะถือว่าเก็บข้อมูลอยู่หน่วยละ 1 ตัวอักษร ซึ่งสามารถที่จะอ่านและทำการเปลี่ยนแปลงค่าที่อยู่ในอุปกรณ์หน่วยความจำนี้ได้ ส่วนออโตมาตาจะมีหน่วยควบคุม (control unit) ซึ่งจะเก็บสถานะภายใน (internal state) อยู่ ดังรูป 2.1 ซึ่งแสดงโครงสร้างโดยทั่วไปของออโตมาตาที่เมื่อมีข้อมูลนำเข้ามาแล้วจะถูกหน่วยควบคุมนำไปเก็บไว้ในหน่วยความจำ และเมื่อออโตมาตาในหน่วยควบคุมต้องการที่จะใช้ข้อมูลนำเข้าก็จะดึงข้อมูลนำเข้ามาจากหน่วยความจำทีละตัวอักษรจนกระทั่งอ่านข้อมูลนำเข้าครบทุกตัวอักษรของข้อมูลนำเข้าแล้ว ก็จะพิจารณาสถานะสุดท้ายว่าเป็นสถานะที่ออโตมาตานี้ยอมรับหรือไม่ แล้วจึงส่งค่ายอมรับหรือไม่ยอมรับออกมาเป็นผลลัพธ์



รูปที่ 2.1 โครงสร้างโดยทั่วไปของออโตมาตา

คุณลักษณะโดยทั่วไปของออโตมาตาจะมีดังนี้

- ประกอบด้วยสถานะ (states) ฟังก์ชันการเปลี่ยนสถานะ (transition function) สถานะเริ่มต้น (intitial states) และสถานะการยอมรับ (accepting states)
- รับอินพุตจากภายนอกระบบเข้าอย่างต่อเนื่อง เรียกอินพุตที่รับเข้ามานี้ว่าตัวอักษร (alphabets)
- ลำดับของตัวอักษรที่เป็นอินพุตซึ่งรับเข้ามาเรื่อยๆ นั้นจะเป็นสายอักขระที่เรียกว่า คำ (words)
- มีการเปลี่ยนสถานะตามที่กำหนดโดยฟังก์ชันการเปลี่ยนสถานะ ซึ่งจะขึ้นอยู่กับตัวอักษรที่รับอินพุตเข้ามา
- เมื่อหยุดการรับอินพุต หากออโตมาตาอยู่ในสถานการยอมรับถือว่าออโตมาตายอมรับคำที่เป็นอินพุตนั้น แต่ถ้าออโตมาตาอยู่นอกสถานการยอมรับ ถือว่าออโตมาตาปฏิเสธคำที่เป็นอินพุตนั้น
- เซตของคำทั้งหมดที่ออโตมาตานั้นยอมรับเรียกว่า ภาษา (language) ซึ่งยอมรับโดยออโตมาตา

เนื่องจากออโตมาต้ามีสถานะที่จำกัดดังนั้นจึงอาจเรียกได้ว่า ออโตมาตาจำกัด (Finte Automata, FA) ออโตมาตาสามารถที่จะแบ่งออกได้เป็น 2 แบบใหญ่ๆ คือ ออโตมาตาจำกัด เชิงกำหนด (deterministic finite automata, DFA) กับออโตมาตาจำกัดเชิงไม่กำหนด (nondeterministic finite automata, NFA) และจะมีการประยุกต์ออโตมาตาเป็นแบบถ่วงนำหนักคือ ออโตมาตาแบบถ่วงนำหนัก (weighted finite automa, WFA) โดยออโตมาตาจะมีข้อมูลนำเข้าชนิดเดียวกันก็คือ มีลักษณะเป็นสายอักขระ

2.1.1 สายอักขระ (String) และภาษา (Language)

ชุดตัวอักษร คือเซตจำกัดของสัญลักษณ์ซึ่งจะไม่เป็นเซตว่าง โดยจะใช้สัญลักษณ์ Σ แทนชุดตัวอักษรใดๆ เช่น $\Sigma=\{0, 1\}$ หรือ $\Sigma=\{a, b, c\}$ เป็นต้น สายอักขระ จากชุดอักษร Σ คือลำดับจำกัดของสัญลักษณ์จาก Σ ซึ่งเขียนติดกันโดยไม่มีช่องว่างและไม่มีเครื่องหมายใดๆ มาคั่น เช่น $\Sigma=\{0, 1\}$ จะสามารถสร้างสายอักขระ 1100 ได้เป็นต้น

ถ้ากำหนดให้ w เป็นสายอักขระใดๆ ความยาว (length) ของ w เขียนแทนด้วย $|w|$ คือจำนวนตัวอักษรทั้งหมดที่ปรากฏอยู่ในสายอักขระ w เช่น ถ้า w=10011 แล้ว $|w|=5$

สายอักขระว่าง (empty string) เขียนแทนด้วย ε คือสายอักขระที่ไม่มีสัญลักษณ์ใดๆ ปรากฏเลย ดังนั้น $|\varepsilon|=0$

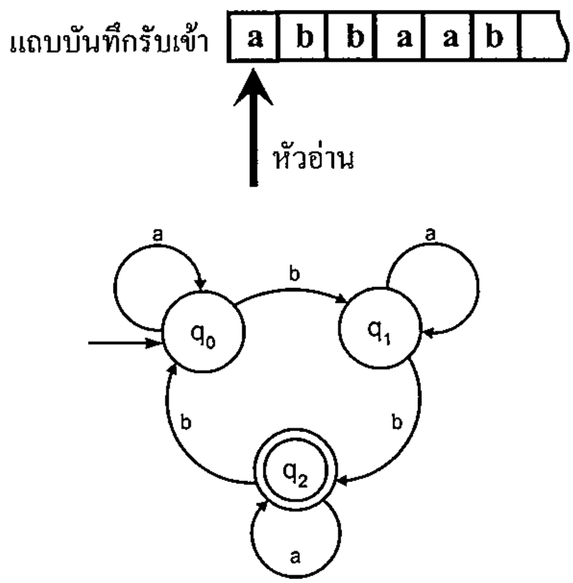
ถ้า w และ v เป็นสายอักขระจากชุดตัวอักษรเดียวกันจะสามารถสร้างสายอักขระใหม่จากการนำ w และ v ได้โดยการนำมาต่อกัน (concatenate) จะได้ wv ดังนั้น w^5=wwwww โดยจะสามารถหาความยาวของสายอักขระจะเท่ากับ $|wv|=|w|+|v|$ และกำหนดให้ $w^0=\varepsilon$ และ $w^1=w$

สำหรับชุดตัวอักษร Σ ใดๆ สัญลักษณ์ Σ^* จะหมายถึงเซตของสายอักขระทั้งหมดจาก Σ ซึ่งจะรวมถึงสายอักขระว่างด้วย และสัญลักษณ์ Σ^+ จะหมายถึง $\Sigma^*-\{\varepsilon\}$

กำหนดให้ L เป็น ภาษา (Language) จาก Σ ก็ต่อเมื่อ $L\subseteq\Sigma^*$ ตัวอย่างเช่น $\Sigma=\{0, 1\}$ จะสามารถสร้างภาษาได้เช่น ε, {0, 1}, {00, 1, 01} เป็นต้น และเรียกสมาชิกทุกตัวใน L ว่า ประโยค (sentence) ของ L

2.1.2 ออโตมาตาเชิงกำหนด (Deterministic Finite Automata, DFA)

ลักษณะของออโตมาตาเชิงกำหนด จะรับข้อมูลเข้าเป็นสายอักขระผ่านทาง แถบบันทึกรับเข้า (input tape) ที่จะอ่านข้อมูลเข้าไปทีละตัวจากทางด้ายซ้ายมือโดยหัวอ่านดังรูปที่ 2.2 และเขียนออโตมาตาเชิงกำหนด อยู่ในรูปของกราฟบ่งบอกทิศทาง (directed graph) โดยจะเริ่มจากสถานะเริ่มต้น(initial state, q_0) โดยในการย้ายสถานะแต่ละครั้ง ออโตมาตาเชิงกำหนด จะอยู่ในสถานะใดสถานหนึ่งเสมอ เช่นในรูปที่ 2.2 แสดงให้เห็นถึงออโตมาตาที่กำลังอยู่ในสถานะที่ q_0 กำลังจะอ่านข้อมูลนำเข้า a เมื่ออ่านเสร็จแล้วหัวอ่านจะย้ายไปทางขวา 1 ช่อง และออโตมาตาเชิงกำหนด จะย้ายสถานะไป q_1 ซึ่งในออโตมาตาเชิงกำหนด ใดๆ อาจจะมีการย้ายสถานะหรือไม่ย้ายก็ได้ขึ้นอยู่กับตัวอักษรที่เป็นข้อมูลนำเข้าและลักษณะของฟังก์ชันทางผ่าน เมื่อหัวอ่านทำการอ่านข้อมูลนำเข้าทีละ 1 ตัวจนหมดสายอักขระ แล้วออโตมาตาเชิงกำหนด จะบ่งบอกว่าสายอักขระนี้ยอมรับโดยออโตมาตาเชิงกำหนด หรือไม่ โดยดูได้จากสถานะสุดท้ายของออโตมาตาเชิงกำหนด ว่าเป็นสถานะที่ยอมรับ (accepting state) ได้หรือไม่ ในรูปที่ 2.2 สถานะที่ยอมรับจะถูกแทนด้วยวงกลม 2 วงซ้อนกัน ก็คือสถานะ q_2

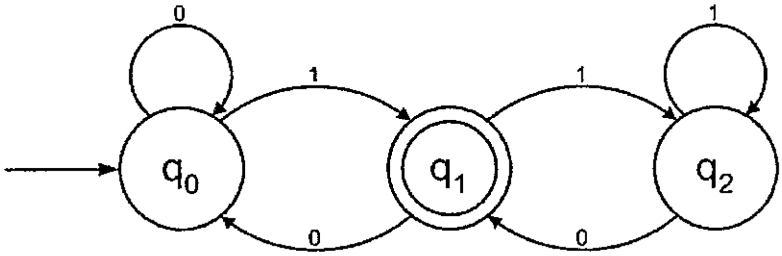


รูปที่ 2.2 แสดงตัวอย่างออโตมาตาเชิงกำหนด

ส่วนประกอบของออโตมาตาเชิงกำหนด M ใดๆ จะประกอบไปด้วยส่วนประกอบ 5 ส่วนคือ $M=(Q,\Sigma,\delta,q_0,F)$ เมื่อ

- Q เป็นเซตจำกัดของสถานะภายใน
- Σ เป็นเซตจำกัดของสัญลักษณ์นำเข้าที่เรียกว่าตัวอักษรนำเข้า (input alphabet)
- δ เป็นฟังก์ชันการเปลี่ยนสถานะของแต่ละสถานะ โดยเป็นฟังก์ชันจาก $Q\times\Sigma$ ไปยัง Q เขียนแทนด้วย $\delta : Q\times\Sigma\rightarrow Q$
- q_0 เป็นสถานะเริ่มต้น
- F เป็นเซตของสถานะสุดท้ายที่เป็นสถานะที่มีการยอมรับสายอักขระ

ตัวอย่างที่ 2.1 จากรูปที่ 2.3 กำหนดให้แทนออโตมาตา M



รูปที่ 2.3 ออโตมาตา M ซึ่งเขียนอยู่ในรูปของกราฟบ่งบอกทิศทาง

จากรูปที่ 2.3 จะแสดงถึงออโตมาตา M ซึ่งเป็นอยู่ในรูปของกราฟบ่งบอกทิศทาง ซึ่งจะกำหนดให้แต่ละจุดยอด (vertex) หรือ โหนดแทนแต่ละสถานะ โดยที่สถานะเริ่มต้นจะมีลูกศรชี้อยู่ และสถานะสุดท้ายทุกจุดใน F จะแทนด้วยวงกลม 2 วงซ้อนกัน แต่ละขอบ (edge) ให้แทนฟังก์ชันการเปลี่ยนสถานะ ซึ่งเชื่อมจากจุด p ไปยังจุด q โดยมีสัญลักษณ์นำเข้าเป็นฉลาก (label) เขียนกำกับ ซึ่งสามารถเขียนแจกแจงรายละเอียดออโตมาตาได้ดังนี้ $M=\{\{q_0, q_1, q_2\}, \{0, 1\}, \delta, \{q_0\},\{q_1\}\}$ ซึ่ง δ หรือ ฟังก์ชันการเปลี่ยนสถานะ สามารถเขียนแสดงเป็นตารางได้ดังตารางที่ 2.1

ตารางที่ 2.1 แสดงฟังก์ชันการเปลี่ยนสถานะของออโตมาตา M

δ	0	1
q_0	q_0	q_1
q_1	q_0	q_2
q_2	q_1	q_2

ในการแสดงฟังก์ชันการเปลี่ยนแปลงของแต่ละสถานะจะอยู่ในรูป

$$\delta(q_i,a)=q_j$$

เมื่อ $q_i,q_j \in Q$;i=0, 1, 2 และ $a \in \{0, 1\}$

เช่น $\delta(q_0,1)=q_1$ แต่เนื่องจากในการรับข้อมูลเข้านั้นไม่ได้รับเพียงแค่สัญลักษณ์นำเข้าเพียงตัวเดียว แต่เป็นสายอักขระ ดังนั้นจึงต้องเขียนให้อยู่ในรูป $\delta^*(q_i,w)$ โดยที่ $q_i \in Q$, $w \in \Sigma^*$ ซึ่งสัญลักษณ์ดาว (star) จะบ่งบอกว่าข้อมูลนำเข้าที่กำลังอ่านอยู่ ยังมีลักษณะเป็นสายอักขระ ซึ่งเมื่อทำการย้ายสถานะเรียบร้อยแล้วก็ให้อ่านข้อมูลนำเข้าตัวต่อไป โดยฟังชันก์การเปลี่ยนแปลงถัดไปจะมีสถานะเป็น $\delta^*(q_j,x)$ เมื่อ $\delta(q_i,a)=qj$ โดยที่ $q_j,q_i \in Q$, $w,a \in \Sigma^*$, $x \in \Sigma$ และ ax=w ซึ่งจะเขียนอยู่ในรูปทั่วไปได้โดย

$$\delta^*(q_i,w) = \delta^*(q_j,x)$$

ในการเริ่มตรวจสอบข้อความว่าออโตมาตาจะยอมรับข้อความนี้หรือไม่ จะเริ่มต้นที่โหนดเริ่มต้น (สถานะเริ่มต้น) และจะอ่านข้อมูลเข้ามาทีละตัวอักษรจากทางด้านซ้ายไปเรื่อยๆ จนกระทั่งครบทุกตัวอักษร เช่น ข้อความ 0111

$$\begin{aligned} &\delta^*(q_0,0111)\\ &= \delta^*(q_0,111)\\ &= \delta^*(q_1,11)\\ &= \delta(q_2,1)\\ &= q_2 \end{aligned}$$

ซึ่งจะเห็นว่าสถานะสุดท้ายเป็นโหนด q_2 แต่โหนด q_2 ไม่ได้เป็นสถานะสุดท้ายที่จะแสดงว่ายอบรับข้อความนี้ ดังนั้นสำหรับข้อความนี้จึงไม่ยอมรับโดยออโตมาตา M ข้อความ 01111001

$$\begin{aligned} &\delta^*(q_0,01111001)\\ &= \delta^*(q_0,1111001) \end{aligned}$$

$= \delta^*(q_1,111001)$
$= \delta^*(q_0,11001)$
$= \delta^*(q_1,1001)$
$= \delta^*(q_2,001)$
$= \delta^*(q_1,01)$
$= \delta(q_0,1)$
$= q_1$

จะเห็นข้อความ 01111001 สถานะสุดท้ายจะอยู่ที่โหนด q_1 ซึ่งเป็นสถานะสุดท้าย ดังนั้นออโตมาตา M จึงยอมรับข้อความนี้

2.1.3 ออโตมาตาจำกัดเชิงไม่กำหนด (Nondeterministic Finite Automata, NFA)

ลักษณะของออโตมาตาจำกัดเชิงไม่กำหนด หรือ เอ็นเอฟเอ จะมีคล้ายกับออโตมาตาเชิงกำหนด จะต่างกันก็เพียงฟังก์ชันทางผ่าน โดยจะสังเกตเห็นว่าออโตมาตาเชิงกำหนด มีข้อมูลออกของฟังก์ชันทางผ่านเป็นสถานะได้เพียงสถานะเดียวเท่านั้น แต่ของเอ็นเอฟเอนั้นจะมีข้อมูลออกของฟังก์ชันทางผ่านเป็นเซตกำลัง (power set) ของ Q หรือ P(Q)

ส่วนประกอบของออโตมาตาเชิงไม่กำหนด N ใดๆ จะประกอบไปด้วยส่วนประกอบ 5 ส่วนเช่นกันคือ $N=(Q,\Sigma,\delta,q_0,F)$ เมื่อ

Q เป็นเซตจำกัดของสถานะภายใน

Σ เป็นเซตจำกัดของสัญลักษณ์นำเข้าที่เรียกว่าตัวอักษรนำเข้า

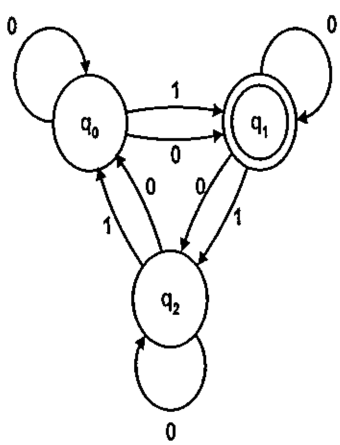
δ เป็นฟังก์ชันจาก $Q\times\Sigma$ ไปยัง P(Q) เขียนแทนด้วย $\delta : Q\times\Sigma\rightarrow P(Q)$

q_0 เป็นสถานะเริ่มต้น

F เป็นเซตของสถานะสุดท้ายที่เป็นสถานะที่มีการยอมรับสายอักขระ

จากฟังก์ชันทางผ่านจะเห็นได้ว่า $\delta(q,x)$ มีได้มากว่าหนึ่งสถานะ เมื่อ $q\in Q$, $x\in\Sigma$ ดังนั้นแต่ละข้อมูลนำเข้าของเอ็นเอฟเอ จึงมีเส้นทางผ่านได้หลายเส้นทาง ซึ่งอาจที่จะมีบางเส้นทางที่ทำให้สามารถไปถึงสถานะสุดท้ายที่มีการยอมรับสายอักขระนี้ได้ ก็จะถือว่าเอ็นเอฟเอนี้ยอมรับสายอักขระนี้ ดังตัวอย่างที่ 2.2

ตัวอย่างที่ 2.2 จากรูปที่ 2.4 แสดงถึงเอ็นเอฟเอ N



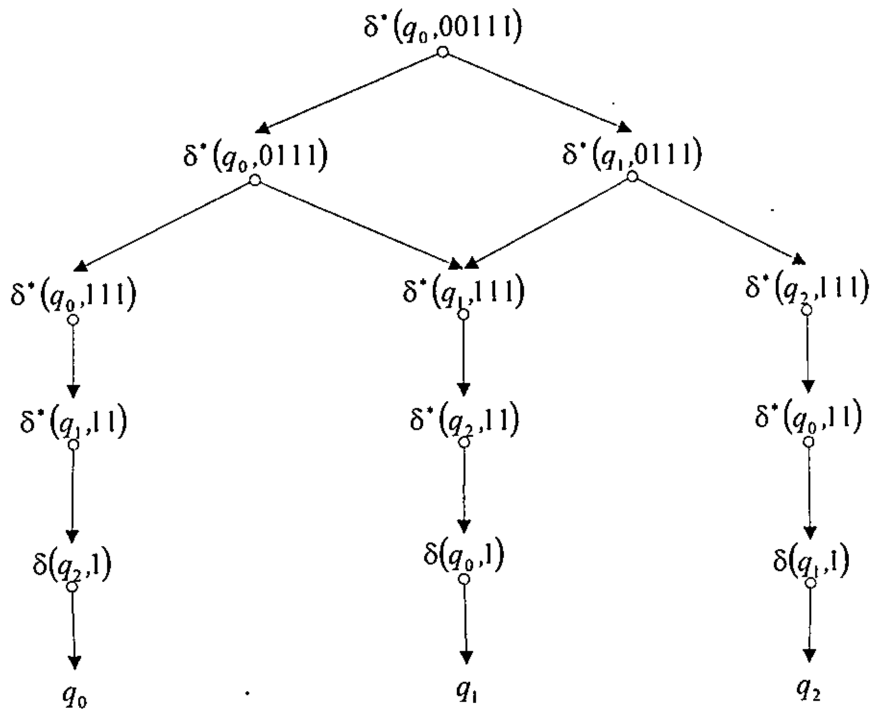
รูปที่ 2.4 แสดงถึงเอ็นเอฟเอ N

การแจกแจงสมาชิกของออโตมาตาจำกัดเชิงไม่กำหนดจะเขียนได้ดังนี้ $N=\{\{q_0,q_1,q_2,q_3\}, \{0,1\},\delta, \{q_0\},\{q_1,q_2\}\}$ โดยที่ฟังก์ชันทางผ่านจะเป็นดังตารางที่ 2.2

ตารางที่ 2.2 แสดงถึงฟังก์ชันทางผ่านของเอ็นเอฟเอ N

δ	0	1
q_0	$\{q_0, q_1\}$	$\{q_1\}$
q_1	$\{q_1, q_2\}$	$\{q_2\}$
q_2	$\{q_2, q_0\}$	$\{q_0\}$

จะสังเกตได้ว่ามีฟังก์ชันทางผ่านบางคู่ที่ได้ผลลัพธ์เป็นสถานะ 2 สถานะ เช่น $\delta(q_0, 0)=\{q_0, q_1\}$ ดังนั้นแต่ละข้อมูลนำเข้าจึงอาจจะสามารถสร้างทางผ่านได้หลายเส้นทาง แต่ขอเพียงแค่มีทางผ่านทางเดียวที่สามารถไปยังสถานะที่ยอมรับได้ ก็จะถือว่าสายอักขระนั้นยอมรับโดยเอ็นเอฟเอ N เช่น 00111 จะสามารถสร้างทางผ่านได้ดังรูปที่ 2.5



รูปที่ 2.5 แสดงเส้นทางผ่านของเอ็นเอฟเอ N

จะเห็นได้ว่าสายอักขระ 00111 สามารถสร้างทางผ่านได้หลายเส้นทางแต่มีอยู่เส้นทางบางเส้นทางที่สามารถให้สถานะที่ยอมรับได้คือ

$$
\begin{aligned}
&\delta^*(q_0,00111)\\
&=\delta^*(q_1,0111)\\
&=\delta^*(q_1,111)\\
&=\delta^*(q_2,11)\\
&=\delta(q_0,1)\\
&=q_1
\end{aligned}
$$

หรือ

$$
\begin{aligned}
&\delta^*(q_0,00111)\\
&=\delta^*(q_0,0111)\\
&=\delta^*(q_1,111)\\
&=\delta^*(q_2,11)\\
&=\delta(q_0,1)\\
&=q_1
\end{aligned}
$$

ก็จะถือว่าสายอักขระหรือข้อมูลนำเข้า 00111 นี้ยอมรับโดยเอ็นเอฟเอ N

2.2 สมการเชิงเส้น (Linear Equation)

ในส่วนนี้จะกล่าวถึงส่วนที่เกี่ยวข้องกับการหาค่าถ่วงน้ำหนักของออโตมาตาซึ่งจะใช้ความรู้ทางคณิตศาสตร์ในการแก้สมการ

2.2.1 ระบบสมการเชิงเส้น (System of Linear Equation)

เมื่อกล่าวถึงสมการโดยทั่วๆ ไปที่กำลังของตัวแปร (variable) หรือตัวไม่ทราบมีกำลังเป็น 1 แล้ว สมการจะอยู่ในรูปของสมการเชิงเส้น ดังนี้

$$a_1x_1 + a_2x_2 + \dots + a_nx_n = b \tag{2.1}$$

เมื่อ a_i, $b \in \Re$ และ x_i เป็นตัวแปรที่ยังไม่รู้ค่า ซึ่งจะเรียก $a_1, a_2, \dots, a_n$ ว่าเป็นสัมประสิทธิ์ของ $x_1, x_2, \dots, x_n$ ตามลำดับ และเรียก b ว่าเป็นค่าคงตัวของสมการ ในกรณีที่กำหนดค่าคงที่ให้กับตัวแปรทุกตัวแล้วทำให้สมการที่ 2.1 เป็นจริง หรือจะกล่าวว่า ถ้ากำหนดให้

$x_1 = k_1, x_2 = k_2, \dots, x_n = k_n$ เมื่อ k_i เป็นค่าคงที่

จะทำให้ $a_1k_1 + a_2k_2 + \dots + a_nk_n = b$

จะเรียกเซต (set) ของ $\{ k_1, k_2, \dots, k_n \}$ ว่า ผลเฉลย (solution) ของสมการที่ 2.1 ซึ่งผลเฉลยของสมการอาจจะมีได้มากกว่า 1 เซต

ในกรณีที่สมการที่ 2.1 มี m สมการแต่มีตัวแปรชุดเดียวกัน ดังนี้

$$\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &= b_1 \\
a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n &= b_2 \\
a_{31}x_1 + a_{32}x_2 + \dots + a_{3n}x_n &= b_3 \\
&\cdots\cdots \\
a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n &= b_m
\end{aligned} \tag{2.2}$$

จะเรียกสมการทุกสมการรวมกันว่า ระบบสมการเชิงเส้น (System of Linear Equation) และจะเรียกผลเฉลยของระบบสมการเชิงเส้นว่า เซตผลเฉลย (solution set) หรือ ผลเฉลยทั่วไป (general solution) ในกรณีที่ $b_1, b_2, \dots, b_m$ มีค่าเท่ากับ 0 ทั้งหมดจะเรียกระบบสมการนี้ว่าเป็น เอกพันธุ์ (hemogeneous) ซึ่งจะเขียนได้อยู่ในรูปของ สมการที่ 2.3

$$\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &= 0 \\
a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n &= 0 \\
&\cdots\cdots \\
a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n &= 0
\end{aligned} \tag{2.3}$$

ซึ่งระบบสมการเอกพันธุ์นี้จะมีเซตของผลเฉลยเสมออย่างน้อย 1 เซต คือ {0, 0, … , 0} ซึ่งจะมีชื่อเฉพาะสำหรับเรียกผลเฉลยนี้ว่า ผลเฉลยที่เป็นศูนย์ หรือผลเฉลยที่มีความสำคัญน้อย (trivial solution)

ในกรณีที่นำค่าคงที่ c_i คูณกับสมการที่ i ของระบบสมการ 2.2 ซึ่งจะได้ดังนี้

$$
\begin{aligned}
c_1a_{11}x_1 + c_1a_{12}x_2 + \dots + c_1a_{1n}x_n &= c_1b_1 \\
c_2a_{21}x_1 + c_2a_{22}x_2 + \dots + c_2a_{2n}x_n &= c_2b_2 \qquad (2.4) \\
&\dots\dots\dots \\
c_ma_{m1}x_1 + c_ma_{m2}x_2 + \dots + c_ma_{mn}x_n &= c_mb_m
\end{aligned}
$$

แล้วนำแต่ละสมการมาบวกกันแล้วจัดรูปจะได้ดังสมการที่ 2.4 ดังนี้

$$
\begin{aligned}
&(c_1a_{11} + c_2a_{21} + c_3a_{31} + \dots + c_ma_{m1})x_1 + \\
&(c_1a_{12} + c_2a_{22} + c_3a_{32} + \dots + c_ma_{m2})\, x_2 + \\
&\dots\dots\dots\dots + (c_1a_{1n} + c_2a_{2n} + c_3a_{3n} + \dots + c_ma_{mn})x_n \\
&= c_1b_1 + c_2b_2 + c_3b_3 + \dots + c_mb_m \qquad (2.4)
\end{aligned}
$$

จะเรียกสมการในลักษณะนี้ว่า การรวมเชิงเส้น (linear combination) ซึ่งจะสามารถกล่าวได้ ผลเฉลยของระบบสมการเชิงการใดๆ จะเป็นผลเฉลยของการรวมเชิงเส้นของระบบสมการนั้นด้วย หรือจะกล่าวว่าถ้าเซตของ $\{ k_1, k_2, \dots , k_n \}$ เป็นผลเฉลยของสมการ 2.2 แล้วจะเป็นผลเฉลยของสมการ 2.4 ด้วย ซึ่งจะแสดงได้ดังนี้

จาก $\{ x_1, x_2, \dots , x_n \}=\{ k_1, k_2, \dots , k_n \}$ ดังนั้น

$$
\begin{aligned}
a_{11}k_1 + a_{12}k_2 + \dots + a_{1n}k_n &= b_1 \\
a_{21}k_1 + a_{22}k_2 + \dots + a_{2n}k_n &= b_2 \\
a_{31}k_1 + a_{32}k_2 + \dots + a_{3n}k_n &= b_3 \qquad (2.5) \\
&\dots\dots\dots \\
a_{m1}k_1 + a_{m2}k_2 + \dots + a_{mn}k_n &= b_m
\end{aligned}
$$

หรือ $a_{i1}k_1 + a_{i2}k_2 + \dots + a_{in}k_n = b_i$ เมื่อ $i = 1, 2, 3, \dots , m$

จะต้องแสดงว่า

$$(c_1a_{11}+\ \dots + c_ma_{m1})k_1+\dots+ \qquad (c_1a_{1n}+\dots + c_ma_{mn})k_n = c_1b_1+\ \dots + c_mb_m$$

จัดรูปแบบพจน์ทางด้านซ้ายมือของเครื่องเท่ากับใหม่จะได้ว่า

$$c_1(a_{11}k_1+\dots+a_{1n}k_n) + \qquad c_2(a_{21}k_1+\dots+a_{2n}k_n) + \dots + c_m(a_{m1}k_1+\dots+a_{mn}k_n)$$

จากสมการ 2.5 จะได้ว่าเท่ากับ $c_1b_1+c_2b_2+c_3b_3+\ \dots\ +\ c_mb_m$

นอกจากนี้ยังสามารถที่จะรูปแบบสมการ 2.5 ให้อยู่ในรูปเมตริกซ์ดังนี้

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \tag{2.6}$$

ซึ่งสามารถเขียนอยู่ในรูปย่อคือ $[A]\times[X]=[B]$ โดยที่ [A] เป็นเมตริกซ์ขนาด $m\times n$ และ [X], [B] เป็นเมตริกซ์คอลัมน์ แต่อาจจะเรียกได้อีกอย่างหนึ่งก็คือจัดให้เมตริกซ์ A อยู่ในรูปของเซตของเวกเตอร์คอลัมน์ A หรือ $A=\{A_1, A_2, \dots , A_n\}$ และ เวกเตอร์คอลัมน์ X กับเวกเตอร์คอลัมน์ B

2.2.2 การแยกส่วนประกอบด้วยคิวอาร์ (QR Factorization)

การแยกส่วนประกอบด้วยคิวอาร์ นิยมใช้ในการคำนวณหาค่าไอเกนของเมตริกซ์ (eigenvalues of a matrix) ในการแก้ระบบสมการเชิงเส้น (linear systems)

ถ้า A เป็นเมตริกซ์ที่มีขนาด $m\times n$ ซึ่งแต่ละคอลัมน์ (columns) เป็นอิสระต่อกัน แล้ว A สามารถแยกส่วนประกอบได้เป็น A = QR เมื่อ Q เป็นเมตริกซ์ที่มีขนาดเป็น $m\times n$ โดยแต่ละคอลัมน์มีขนาดเป็น 1 และเป็นอิสระต่อกันและ R เป็นเมตริกซ์สามเหลี่ยมบนที่มีขนาดเท่ากับ $n\times n$

ขั้นตอนในการหาส่วนประกอบด้วยคิวอาร์

ขั้นที่ 1: หาเมตริกซ์ $[w_1\ w_2 \dots w_n]$ ที่มีขนาดเป็น 1 และแต่ละคอลัมน์เป็นอิสระต่อกันของเมตริกซ์ $A=[u_1\ u_2 \dots u_n]$ โดยการใช้วิธีแกรม-ชมิตท์ (Gram-Schmidt) ที่กล่าวว่า เซตตั้งฉากปกติ $[u_1\ u_2 \dots u_n]$ เป็นเซตอิสระเชิงเส้น แล้ว

$$v_i = u_i - \langle w_1,u_i\rangle w_1 - \langle w_2,u_i\rangle w_2 - \dots - \langle w_{i-1},u_i\rangle w_{i-1}\ ,$$

$$w_i = \frac{v_i}{\|v_i\|} \; ; i = 1, 2, \ldots \tag{2.7}$$

โดยให้ $Q = [w_1 \ w_2 \ldots w_n]$

ขั้นที่ 2: คำนวณหา R จาก $A = QR$

$$R = Q^t A$$

ตัวอย่าง 1 หาส่วนประกอบคิวอาร์ของ

$$A = \begin{bmatrix} 1 & -1 & -1 \\ 1 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}.$$

ขั้นที่ 1: จะได้เวกเตอร์ u_i คือ

$$u_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \qquad u_2 = \begin{bmatrix} -1 \\ 0 \\ -1 \\ 0 \end{bmatrix}, \qquad u_3 = \begin{bmatrix} -1 \\ 0 \\ 0 \\ -1 \end{bmatrix}$$

อาศัยวิธีการของแกรมชมิดท์ในสมการที่ (2.7) จะได้ว่า

$v_1 = u_1 = [1 \ \ 1 \ \ 1 \ \ 0]^t$และ $w_1 = \dfrac{u_1}{\|u_1\|} = \dfrac{[1 \ \ 1 \ \ 1 \ \ 0]^t}{\sqrt{1^2+1^2+1^2+0^2}} = \left[\dfrac{1}{\sqrt{3}} \ \ \dfrac{1}{\sqrt{3}} \ \ \dfrac{1}{\sqrt{3}} \ \ 0\right]^t$

$$v_2 = u_2 - \langle w_1, u_2 \rangle w_1 = \begin{bmatrix} -1 \\ 0 \\ -1 \\ 0 \end{bmatrix} - \left\langle \begin{bmatrix} 1/\sqrt{3} \\ 1/\sqrt{3} \\ 1/\sqrt{3} \\ 0 \end{bmatrix}, \begin{bmatrix} -1 \\ 0 \\ -1 \\ 0 \end{bmatrix} \right\rangle \begin{bmatrix} 1/\sqrt{3} \\ 1/\sqrt{3} \\ 1/\sqrt{3} \\ 0 \end{bmatrix} = \begin{bmatrix} -1/3 \\ 2/3 \\ -1/3 \\ 0 \end{bmatrix}$$

และ $w_2 = \dfrac{v_2}{\|v_2\|} = \dfrac{\left[-\frac{1}{3} \ \ \frac{2}{3} \ \ -\frac{1}{3} \ \ 0\right]}{\sqrt{\left(-\frac{1}{3}\right)^2+\left(\frac{2}{3}\right)^2+\left(-\frac{1}{3}\right)^2+(0)^2}} = \left[-\dfrac{1}{\sqrt{6}} \ \ \dfrac{2}{\sqrt{6}} \ \ -\dfrac{1}{\sqrt{6}} \ \ 0\right]^t$

$v_3 = u_3 - \langle w_1,u_3\rangle w_1 - \langle w_2,u_3\rangle w_2$

$$=\begin{bmatrix}-1\\0\\0\\-1\end{bmatrix}-\left\langle\begin{bmatrix}1/\sqrt{3}\\1/\sqrt{3}\\1/\sqrt{3}\\1/\sqrt{3}\end{bmatrix},\begin{bmatrix}-1\\0\\0\\-1\end{bmatrix}\right\rangle\begin{bmatrix}1/\sqrt{3}\\1/\sqrt{3}\\1/\sqrt{3}\\1/\sqrt{3}\end{bmatrix}-\left\langle\begin{bmatrix}-1/\sqrt{6}\\5/\sqrt{6}\\-1/\sqrt{6}\\0\end{bmatrix},\begin{bmatrix}-1\\0\\0\\-1\end{bmatrix}\right\rangle\begin{bmatrix}-1/\sqrt{6}\\5/\sqrt{6}\\-1/\sqrt{6}\\0\end{bmatrix}$$

$$=\begin{bmatrix}-\frac{1}{2} & \frac{3}{2} & \frac{1}{2} & 0\end{bmatrix}^t$$

$$w_1=\begin{bmatrix}1/\sqrt{3}\\1/\sqrt{3}\\1/\sqrt{3}\\0\end{bmatrix},w_2=\begin{bmatrix}-1/\sqrt{15}\\2/\sqrt{15}\\-1/\sqrt{15}\\3/\sqrt{15}\end{bmatrix},w_3=\begin{bmatrix}-4/\sqrt{35}\\3/\sqrt{35}\\1/\sqrt{35}\\-3/\sqrt{35}\end{bmatrix}.$$

โดย $Q = [w_1\ w_2 \dots w_n]$

ขั้นที่ 2:

$$R=\begin{bmatrix}r_{11} & r_{12} & r_{13}\\0 & r_{22} & r_{23}\\0 & 0 & r_{33}\end{bmatrix}=\begin{bmatrix}3/\sqrt{3} & -2/\sqrt{3} & -1/\sqrt{3}\\0 & 5/\sqrt{15} & -2/\sqrt{15}\\0 & 0 & 7/\sqrt{35}\end{bmatrix},$$

ดังนั้น

$$r_{11}=u_1\cdot w_1=\frac{3}{\sqrt{3}},r_{12}=u_2\cdot w_1=\frac{-2}{\sqrt{3}},r_{22}=u_2\cdot w_2=\frac{5}{\sqrt{15}},$$

$$r_{13}=u_3\cdot w_1=\frac{-1}{\sqrt{3}},r_{23}=u_3\cdot w_2=\frac{-2}{\sqrt{15}},r_{33}=u_3\cdot w_3=\frac{7}{\sqrt{35}}$$

ในการแก้สมการจะสามารถแก้ได้ดังต่อไปนี้

จาก $[A][X] = [B]$

ให้ $[A]=[Q][R]$ ดังนั้นจะได้ว่า

$$[Q][R][X] = [B]$$

$$[Q]^t[Q][R][X] = [Q]^t[B]$$

$$[R][X] = [Z]$$

ในขั้นตอนสุดท้าย จะสามารถคำนวณหาค่า x ได้ด้วยวิธีการคำนวณค่าถอยหลัง (back substitution)

$$\begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ 0 & r_{22} & \cdots & r_{2n} \\ 0 & 0 & \cdots & \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & r_{nn} \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_n \end{bmatrix}$$

ตัวอย่างที่ 2

ให้

$$[A] = \begin{bmatrix} 1 & 2 & -1 & 3 \\ 2 & 1 & 1 & 2 \\ -2 & 3 & 4 & 1 \\ 4 & 2 & 1 & 0 \\ 0 & 2 & 1 & 3 \\ 1 & -1 & 2 & 0 \end{bmatrix}, [B] = \begin{bmatrix} 1 \\ 5 \\ -2 \\ 1 \\ 3 \\ 5 \end{bmatrix}$$

ด้วยวิธีการของแกรมชมิตท์ จะได้

$$[Q] = [w_1 \quad w_2 \quad w_3 \quad w_4] = \begin{bmatrix} -0.1961 & -0.3851 & 0.5099 & 0.3409 \\ -0.3922 & -0.1311 & -0.1768 & 0.4244 \\ 0.3922 & -0.7210 & -0.4733 & -0.2177 \\ -0.7845 & -0.2622 & -0.1041 & -0.5076 \\ 0 & -0.4260 & 0.0492 & 0.4839 \\ -0.1961 & 0.2540 & -0.6867 & 0.4055 \end{bmatrix}$$ และ

$$[R] = \begin{bmatrix} -5.0990 & -0.9806 & 0.1961 & -0.9806 \\ 0 & -4.6945 & -2.8102 & -3.4164 \\ 0 & 0 & -4.0081 & 0.8504 \\ 0 & 0 & 0 & 3.1054 \end{bmatrix}.$$

แล้ว

$$[Z] = [Q]^t \cdot [Z] = \begin{bmatrix} 4.7068 \\ -0.1311 \\ 2.8172 \\ 5.8699 \end{bmatrix}.$$

ขั้นตอนสุดท้ายแก้สมการหาค่า x_i

$$[R]\cdot[X]=\begin{bmatrix} -5.0990 & -0.9806 & 0.1961 & -0.9806 \\ 0 & -4.6945 & -2.8102 & -3.4164 \\ 0 & 0 & -4.0081 & 0.8504 \\ 0 & 0 & 0 & 3.1054 \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

$$=\begin{bmatrix} 4.7068 \\ -0.1311 \\ 2.8172 \\ 5.8699 \end{bmatrix} = [Z] = [Q]^t \cdot [B]$$

จะได้

$$[X]=\begin{bmatrix} 0.999 \\ -2.0643 \\ 1.1039 \\ 1.8902 \end{bmatrix}.$$

2.3 เทคนิคพื้นฐานในการแบ่งรูปภาพดิจิตอล

ในการบีบอัดรูปภาพดิจิตอลด้วยวิธีออโตมาตาแบบถ่วงน้ำหนัก (Weighted Finite Automata, WFA) ซึ่งโดยปกติแล้วจะกำหนดขนาดของรูปภาพเป็น 2 ลักษณะคือ รูปภาพที่มีขนาดจำกัด (finite-resolution image) จะหมายถึงรูปภาพดิจิตอลที่เป็นรูปไล่ระดับสีเทา (grey-scale) ที่มีขนาดเป็น $2^m \times 2^m$ พิกเซล(pixels) (พิกเซล คือจุดที่ประกอบกันเป็นรูปภาพดิจิตอลขึ้นมา เช่น รูปภาพดิจิตอลขนาด 27 × 27 = 128 × 128 พิกเซล หมายถึง รูปภาพดิจิตอลนี้จะมีจำนวนพิกเซลในแนวนอนเท่ากับ 128 จุด และ จำนวนพิกเซลในแนวตั้ง 128 จุด รวมกันทั้งสิ้น 128 × 128 = 16,384) โดยปกติแล้วจะให้มีค่าอยู่ระหว่าง $7 \leq m \leq 11$ โดยความหมายของ รูปภาพที่มีขนาดไม่จำกัด (multirelolution image) จะหมายถึง รูปภาพดิจิตอลที่มีขนาดเท่ากับ $2^n \times 2^n$ พิกเซล สำหรับ $n = 0, 1, 2, \ldots$ ดังนั้นจะสามารถกำหนดนิยามของรูปภาพที่มีขนาดไม่จำกัดเป็นฟังก์ชันได้ดังนี้

$$F:\Sigma^{*}\rightarrow\Re \tag{2.8}$$

เมื่อ Σ เป็นตำแหน่งที่อยู่ซึ่งได้จากการแบ่งรูปภาพ และ

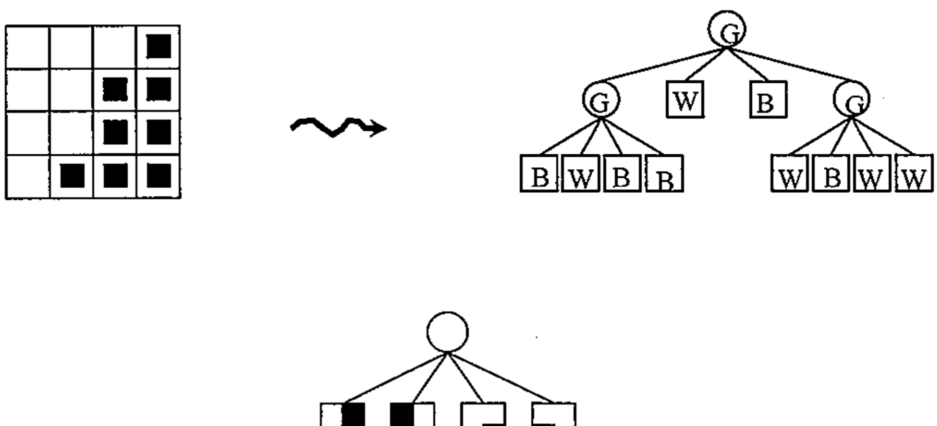
$F(x)$ เป็นค่าเฉลี่ยความเข้มแสงซึ่งจะเป็นจำนวนเต็มที่มีอยู่ระหว่าง 0 ถึง 255 โดยที่ค่าความเข้มแสง 0 จะมีสีดำ และค่าความเข้มแสง 255 จะมีสีขาว

การบีบอัดรูปภาพด้วยวิธีออโตมาตาแบบถ่วงน้ำหนักนี้จะเป็นการบีบอัดรูปภาพแบบไม่จำกัดขนาด โดยก่อนทำการบีบอัดจะต้องทำการแบ่งรูปภาพออกเป็นรูปย่อย โดยแต่ละรูปย่อยจะมีที่อยู่ หรือแอดเดรส (Address) เพื่อบ่งบอกตำแหน่งเพื่อที่จะนำแต่ละส่วนมาหาความสัมพันธ์กันเพื่อที่จะนำรูปย่อยนั้นมาสร้างเป็นโหนด (node) และนำความสัมพันธ์มาสร้างเป็นขอบทางเดิน (edge) ,พาธ (path) หรือ ทรานสิชั่น (transition) ของออโตมาตาแบบถ่วงน้ำหนัก ซึ่งเทคนิคในการแบ่งรูปภาพมีดังนี้

2.3.1 ควอดทรี (Quadtrees)

ควอดทรีเป็นเทคนิคที่ใช้ในการแบ่งรูปภาพเพื่อทำการบีบอัดด้วยเทคนิคออโตมาตาแบบถ่วงน้ำหนักซึ่ง K. Culik II และ J. Kari [1] เป็นผู้คิดค้นการบีบอัดรูปดิจิตอลด้วยออโตมาตาแบบถ่วงน้ำหนัก ซึ่งการแบ่งแบบควอดทรีจะทำให้ได้โครงสร้างข้อมูลแบบเป็นลำดับขั้น โดยแต่ละขั้นจะทำแบบการแบ่งรูปภาพออกเป็น 4 ส่วนไปเรื่อยๆ จนกระทั่งได้ส่วนที่เหมาะสม (ในกรณีที่แย่ที่สุดคือแบ่งจนกระทั่งได้ขนาดของรูปย่อยเป็น 1×1 พิกเซล) ขอบเขตของการแบ่งแบบควอดทรี (region quadtree) คือจะแบ่งรูปภาพต้นฉบับเป็นสี่ส่วนซ้ำไปเรื่อยๆ ด้วยอัตราส่วนของขนาดที่เท่ากัน ด้วยเหตุนี้ควอดทรีจะเป็นสามารถสร้างเป็นต้นไม้ (tree) ได้ โดยที่แต่ละโหนดจะสามารถมีโหนดลูก (Child)ได้ 4 โหนด หรือมีใบ (leaves) ได้ 4 ใบ โดยที่แต่ละโหนดจะสามารถแทนค่าได้ด้วยรูปย่อยซึ่งจะแทนค่าเป็นค่าความเข้มแสง (grayness values) เฉลี่ยของแต่ละรูปย่อย

รูปที่ 2.1 จะแสดงให้เห็นถึงเทคนิคเบื้องต้นของการแบ่งรูปภาพแบบควอดทรี โดยการแบ่งจะเริ่มจากมุมบนขวาของรูปภาพ มุมบนซ้าย มุมล่างขวา และมุมล่างซ้าย ตามลำดับ โดยจะมีตัวอักษร B W G แทนด้วยความเข้มแสงที่เป็นสีดำ ขาว เทา ตามลำดับ โดยจะสังเกตได้ว่าโหนดภายใน (inner node) จะมีเฉพาะตัวอักษร G และ ในส่วนที่เป็นใบ (leaves) ของทรีจะมีแต่ตัวอักษร B และ W ที่เป็นเช่นนี้เพราะว่าเนื่องจากโหนดภายในนั้นยังสามารถแบ่งออกไปได้อีก



รูปที่2.6 แสดงเทคนิคการแบ่งรูปแบบควอทรี

ควอดทรีเป็นวิธีที่เหมาะสำหรับการบีบอัดรูปภาพในลักษณะที่มีขนาดรูปเป็น $2^n \times 2^n$ เพราะไม่ว่ารูปภาพจะมีขนาดใหญ่จะสามารถแบ่งรูปไปได้เรื่อยๆ จนกระทั่งได้โหนดเดี่ยว (single node) ที่เหมาะสม แต่การแบ่งที่ทำซ้ำกันไปเรื่อยๆ จะหยุดในกรณีพิเศษคือ ถ้าทุกๆจุดมีความเข้มแสงเป็นสีขาว หรือดำ ทั้งหมด (homogeneity property) โดยที่ปกติแล้วจะหยุดก็ต่อเมื่อรูปย่อยจะมีขนาดเป็น 1×1 พิกเซล ซึ่งจะเห็นได้ดังนั้นตัวอย่างรูปที่ 2.1 ซึ่งถ้าเปรียบแต่ละช่องสี่เหลี่ยมเล็กๆ เป็นพิกเซล จะสังเกตเห็นว่าโหนดที่ 2 และ 3 ที่เป็นรูปนั้นจะมีขนาดเป็น 2 × 2 พิกเซล แต่ก็หยุดทำการแบ่ง เพราะเนื่องจากรูปย่อยทั้งสองรูปเป็นสีขาวและสีดำทั้งหมด แต่โหนดที่ 1 และ 4 ยังคงเป็นสีเทาดังนั้นจึงยังคงต้องแบ่งอีกจนกระทั่งเป็นพิกเซลเดียว โดยการแบ่งแต่ละครั้งแต่ละรูปย่อยจะมีตำแหน่งแอดเดรสดังนี้

1	**3**
0	**2**

รูปที่ 2.7 แสดงตำแหน่งแอดเดรสของแต่ละรูปย่อย

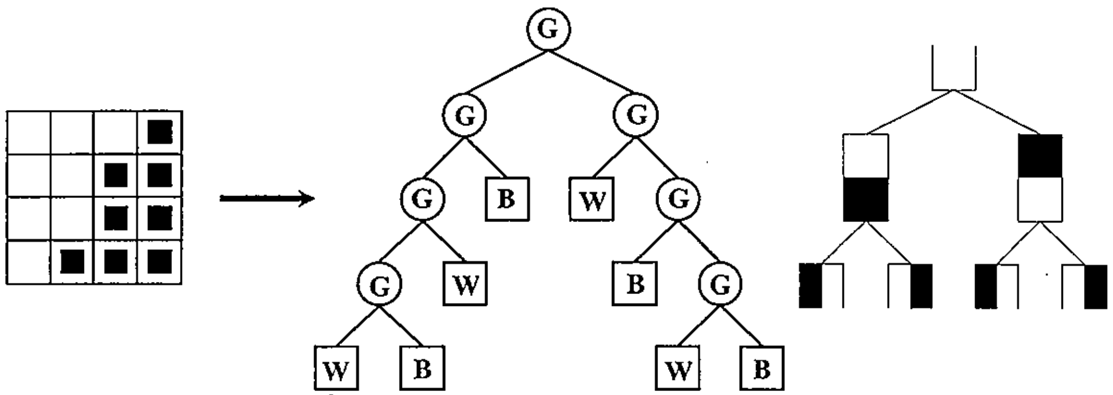
จากนิยามของรูปภาพที่มีขนาดไม่จำกัดจะทำให้สามารถนิยามเอเวอร์เรจพรีเซิฟวิงก์ (Average preservint) หรือ เอพีฟังก์ชัน (ap-funtion) ดังนี้

$$F(x) = \frac{1}{4}\left[F(x_0) + F(x_1) + F(x_2) + F(x_3)\right] \tag{2.9}$$

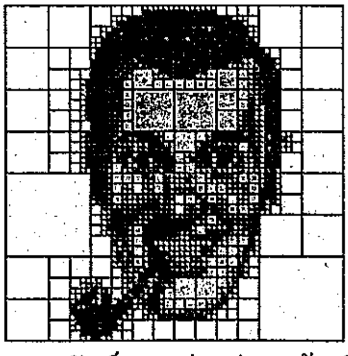
จากนิยามของเอพีฟังก์ชันจะทำให้ทราบได้ว่าเศษหนึ่งส่วนสี่ของผลรวมของค่าเฉลี่ยความเข้มแสงของทุกๆรูปย่อยจะมีค่าเท่ากับค่าเฉลี่ยความเข้มแสงของรูปภาพนั้นๆ ซึ่งจะพิจารณาได้จาก แต่ละรูปจะหาค่าเฉลี่ยความเข้มแสงได้จากผลรวมค่าความเข้มแสงทั้งหมดของรูปนั้นแล้วหารด้วยจำนวนพิกเซลทั้งหมด ด้วยวิธีการหาค่าเฉลี่ยความเข้มแสงถ้ากำหนดให้รูปภาพใดๆ ที่ประกอบด้วยรูปภาพย่อย 4 รูป โดยที่แต่ละรูปย่อยจะมีค่าเฉลี่ยความเข้มแสงเท่ากับ $F(x_0)$, $F(x_1)$, $F(x_2)$, และ $F(x_3)$ ดังนั้นรูปภาพใดๆ จะสามารถหาค่าเฉลี่ยความเข้มแสงได้จากผลรวมของค่าเฉลี่ยความเข้มแสงของทุกๆ รูปย่อยซึ่งมีทั้งหมด 4 รูป และหารด้วยจำนวนรูปย่อย คือ 4

2.3.2 ไบนารีทรี (Binary Trees)

การแบ่งรูปแบบไบนารีทรีเป็นการประยุกต์การแบ่งมาจากควอดทรี โดยที่แต่ละโหนดของรูปภาพจะถูกแบ่งเป็นสองส่วนแทนที่จะเป็นสี่ส่วน การแบ่งแนวนอนและแนวตั้งเป็นการประยุกต์ในเรื่องของเลเวล (level) และดีกรีของแต่ละโหนดของต้นไม้ ซึ่งจะสามารถเห็นได้ว่าไบนารีทรีจะมีเลเวลของทรีเป็นจำนวนคู่ซึ่งจะคล้ายกับการแบ่งแบบควอดทรี ซึ่งในส่วนนี้จะใช้การแบ่งแบบไบนารีทรีมาประยุกต์ใช้กับการบีบอัดรูปภาพด้วยออโตมาตาแบบถ่วงน้ำหนัก สำหรับรูปที่ 2.2 จะแสดงถึงการแบ่งรูปแบบไบนารีทรี ในการตัดสินใจของการแบ่งว่าจะแบ่งแบบแนวนอนหรือแนวตั้งยังคงไม่มีกฎเกณฑ์ในการตัดสินใจที่แน่นอน ดังจะเห็นได้ในรูปที่ 2.3 ว่าแต่ละส่วนของรูปภาพจะมีการแบ่งแต่ละส่วนไม่เท่ากัน



รูปที่ 2.8 แสดงเทคนิคการแบ่งรูปแบบไบนารีทรี



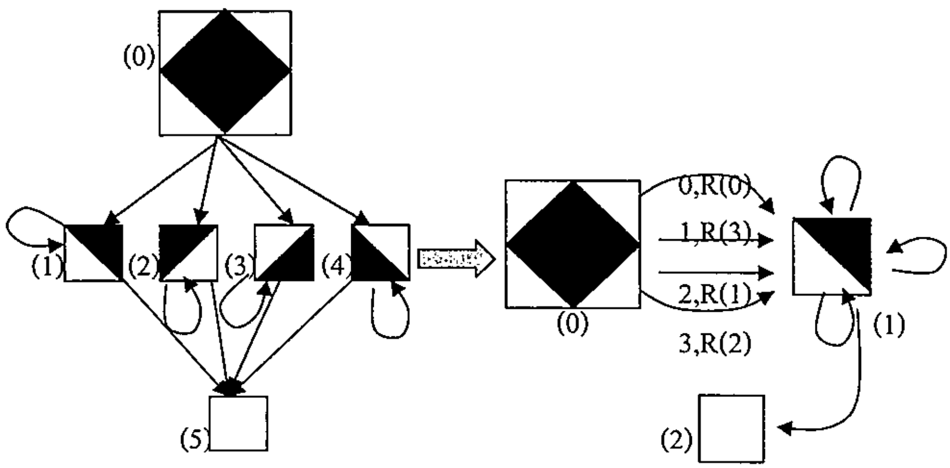
รูปที่ 2.9 แสดงผลลัพธ์การแบ่งรูปภาพด้วยวิธีไบนารีทรี

2.4 การประยุกต์การแบ่งรูป

เมื่อได้ทำการแบ่งรูปออกเป็นส่วนย่อยๆ แล้วและในการหาความสัมพันธ์ระหว่างรูปย่อยด้วยกัน โดยปกติแล้ว เพื่อที่จะให้สามารถหาความสัมพันธ์ระหว่างรูปย่อยได้ง่ายยิ่งขึ้น จึงอาจที่จะเพิ่มขั้นตอนขึ้นมาอีกหนึ่งขั้น นั่นก็คือการหมุน และการย่อขยาย

2.4.1 การหมุน

เมื่อได้ทำการแบ่งรูปย่อยออกเป็นส่วนๆ แล้ว ก่อนที่จะนำมาสร้างเป็นไฟไนต์ออโตมาตาแบบถ่วงน้ำหนักแล้ว อาจจะนำมาหมุน ตามที่ Zhuhan Jiang , Bruce Litow และ Olivier de Vel [7] ได้ทำการวิจัยไว้แล้ว เพื่อที่จะทำให้สามารถใช้เนื้อที่หน่วยความจำได้น้อยลง จึงได้ทำการหมุนรูปย่อยก่อนเพื่อให้ที่จะได้หาความสัมพันธ์ได้ง่ายขึ้น ดังรูป



รูปที่ 2.10 แสดงการหมุนรูปเพื่อลดจำนวนโหนด

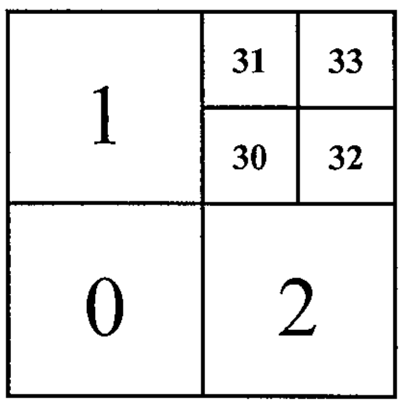
ในรูปที่ 2.5 จะสังเกตเห็นออโตมาตาที่อยู่ทางด้านซ้ายมือ จะเห็นว่าในโหนดที่ (0) เป็นรูปภาพรูปเดียวกับรูปที่จะทำการบีบอัด ซึ่งเมื่อทำการแบ่งรูปภาพออกเป็นสี่ส่วนตามแอดเดรส 0, 1, 2, 3 แล้วก็จะได้รูปย่อยซึ่งจะนำมาสร้างเป็นโหนดที่ (1), (2) ,(3) และ (4) ตามลำดับ ซึ่งถ้าสังเกตจะเห็นว่าโหนดที่ (1), (2), (3) และ (4) เป็นรูปที่มีลักษณะเหมือนกันเพียงแต่มีการหมุนรูปที่ไม่เท่ากัน กล่าวคือรูปของโหนดที่ (2) เกิดจากการหมุนรูปของโหนดที่ (1) ในทิศตามเข็มนาฬิกา ครั้งละ 90 องศา 3 ครั้ง รูปของโหนดที่ (3) ก็เกิดจากรูปของโหนดที่ (1) หมุนไปในทิศทางเดียวกัน 1 ครั้ง และรูปของโหนดที่ (4) เกิดจากการหมุนรูปของโหนดที่ (1) ไปในทิศทางเดียวกัน 2 ครั้ง ซึ่งจะได้ส่วนขยายของทางเดินของโหนดที่ (1), (2), (3) และ (4) เป็น R(0), R(3), R(1) และ R(2) เมื่อเทียบกับรูปของโหนดที่ (1) ตามลำดับ ในการหมุนรูปเพื่อลดจำนวนโหนด ทำให้สามารถใช้เนื้อจัดเก็บได้น้อยลง แต่จะต้องเสียเวลาในการคำนวณมากขึ้นทั้งขั้นตอนในการเข้ารหัสและถอดรหัส เนื่องจากในขั้นตอนการหมุนแต่ละครั้ง เนื่องจากต้องหมุนรูปแล้วจึงหาความสัมพันธ์ จนครบ 4 รอบ โดยการหมุนแต่ละครั้ง นิยมที่จะหมุนครั้งละ 90 องศา

2.4.2 การย่อหรือการขยาย

การย่อหรือการขยายก็เป็นพื้นฐานอย่างหนึ่งของกราฟฟิกส์เช่นกัน แต่โดยพื้นฐานของการเข้ารหัสแบบไฟในต์ออโตมาตาแล้วจะรวมการย่อหรือการขยายเข้าไปอยู่แล้ว ดังนั้น ในพื้นฐานของการย่อยหรือขยายจึงไม่จำเป็นที่จะต้องทำการตรวจสอบ

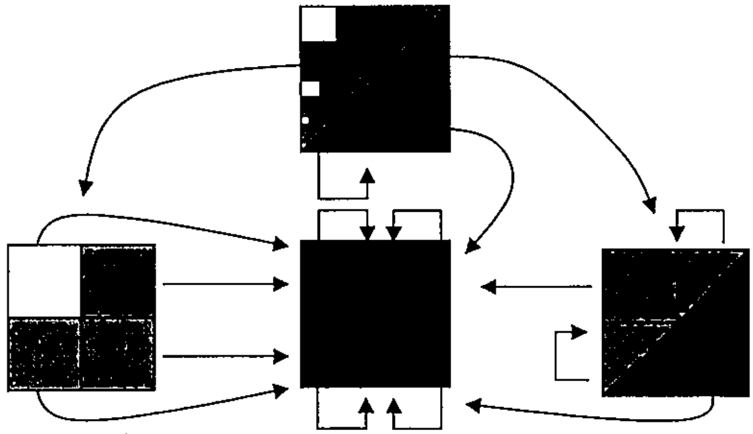
2.5 ความรู้เบื้องต้นในการบีบอัดภาพด้วยวิธีออโตมาตาแบบถ่วงน้ำหนัก

ในแต่ละจุดของรูปภาพดิจิตอลที่มีขนาด $2^n \times 2^n$, $n = 0, 1, 2, \ldots$ ที่จะทำการแบ่งเพื่อทำการบีบอัดรูปภาพ จะมีการกำหนดตำแหน่งแอดเดรสที่มีความยาว *n* ซึ่งจะประกอบด้วยตัวอักษรที่จะบ่งบอกถึงตำแหน่งแอดเดรสของพิกเซลนั้นๆ โดยที่ตัวอักษรแต่ละตัวจะแสดงถึงควอดเดรนท์ของรูปย่อยซึ่งเกิดจากการแบ่งรูปภาพดิจิตอลดังรูป



รูปที่ 2.11 แสดงตำแหน่งแอดเดรสของการแบ่งรูปแบบควอดทรี

โดยเมื่อทำการแบ่งเสร็จสิ้นแล้วก็ทำการสร้างออโตมาตาแบบถ่วงน้ำหนักเพื่อใช้ในเข้ารหัสและถอดรหัสรูปภาพดิจิตอล เมื่อทำการแบ่งรูปภาพเสร็จเรียบร้อยแล้ว ก็นำมาหาความสัมพันธ์กันเพื่อนำมาสร้างเป็นออโตมาตาแบบถ่วงน้ำหนัก ดังรูปที่ 2.7 จะแสดงให้เห็นถึงเทคนิคพื้นฐานในการบีบอัดรูปภาพแบบออโตมาตาแบบถ่วงน้ำหนัก โดยแบ่งแบบควอดทรี



รูปที่ 2.12 แสดงถึงเทคนิคการสร้างกราฟสำหรับการบีบอัดรูปภาพ

จากรูปที่ 2.7 รูปที่อยู่บนสุดจะเป็นรูปที่จะทำการเข้ารหัส จะเป็นถูกกำหนดให้เป็นโหนดที่หนึ่งจากนั้นจึงทำการแบ่งออกเป็นสี่ส่วน จะสังเกตได้ว่าในแอดเดรสที่ 0 ของโหนดที่หนึ่งจะมีลักษณะเหมือนกับโหนดที่หนึ่งเองดังนั้นจึงสร้าง ทางเดิน (Path) วนเข้าหาโหนดที่หนึ่งเอง จากนั้นจึงพิจารณาในส่วนของแอดเดรสที่ 1 ของโหนดที่หนึ่ง ซึ่งจะสังเกตได้ว่าไม่เหมือนกับโหนดที่หนึ่ง ดังนั้น จึงนำรูปย่อยที่อยู่ในส่วนของแอดเดรสที่ 1 ของโหนดที่หนึ่งมาสร้างเป็นโหนดที่สองและสร้างทางเดินจากโหนดที่หนึ่งมายังโหนดที่สอง จากนั้นจึงพิจารณารูปย่อยในส่วนของแอดเดรสที่ 2 และ 3 ของโหนดที่หนึ่ง ซึ่งจะสังเกตได้ว่ามีลักษณะไม่เหมือนกับโหนดที่หนึ่งและโหนดที่สอง ดังนั้นจึงสร้างเป็นโหนดใหม่ได้แก่โหนดที่สามและโหนดที่สี่ตามลำดับ จากนั้นจึงสร้างทางเดินมายังโหนดที่สอง เมื่อพิจารณาทุกส่วนของรูปย่อยของโหนดที่หนึ่งเสร็จเรียบร้อยแล้วจึงพิจารณาโหนดที่สองต่อ

ในโหนดที่สอง จะสังเกตเห็นได้ว่ารูปย่อยที่มีแอดเดรส 0, 2 และ 3 จะมีความเข้มแสงเป็นครึ่งหนึ่งของโหนดที่สาม ดังนั้นจึงสามารถสร้างทางเดินของรูปย่อยที่มีแอดเดรสจากโหนดที่สองไปยังโหนดที่สามได้ ส่วนรูปย่อยที่มีแอดเดรสเป็น 1 จะมีลักษณะเหมือนกับโหนดที่สามดังนั้นจึงสามารถสร้างทางเดินไปยังโหนดที่สามได้เลย เมื่อพิจารณาโหนดที่สองเสร็จครบทุกรูปย่อยแล้วจึงพิจารณาโหนดที่สามต่อ

ในโหนดที่สาม จะสังเกตได้ว่าแต่ละรูปย่อยจะมีลักษณะเหมือนกับโหนดที่สามเองดังนั้นแต่ละรูปย่อยจึงสามารถสร้างทางเดินจากโหนดที่สามวนเข้าหาโหนดที่สามเองได้หลังจากนั้นจึงพิจารณาโหนดสุดท้ายคือโหนดที่สี่ ซึ่งในรูปย่อยที่มีแอดเดรสเป็น 0 กับ 4 จะมีลักษณะเหมือนกับตัวเอง ดังนั้นจึงวนเข้าหาตัวเอง และในรูปย่อยที่มีแอดเดรสเป็น 2 จะมีความเข้มแสงเป็นครึ่งหนึ่งของโหนดสามดังนั้นจึงสร้างทางเดินไปยังโหนดที่สาม ส่วนในรูปย่อยที่มีแอดเดรสเป็น 2 จะมีลักษณะเหมือกับโหนดที่สาม ดังนั้นจึงสร้างทางเดินจากรูปย่อยที่ 2 ของโหนดที่สามไปยังโหนดที่

2

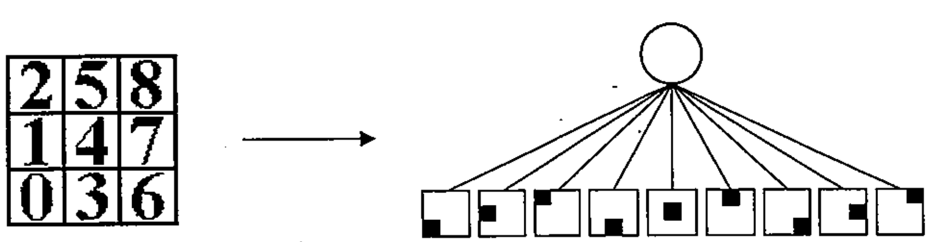
บทที่ 3

การบีบอัดรูปภาพด้วยออโตมาตาแบบถ่วงน้ำหนัก โดยใช้การแบ่งแบบโนนาทรี

จากบทที่แล้วจะเห็นว่ารูปภาพที่จะทำการบีบอัดโดยส่วนมากแล้วจะมีขนาดเป็น 2^nx2^n แต่โดยความจริงแล้วอาจจะมีรูปขนาดอื่นๆ อยู่ด้วย ดังนั้นในบทนี้จะนำเสนอวิธีการแบ่งรูปแบบโนนาทรีเพื่อใช้ในการบีบอัดรูปภาพ

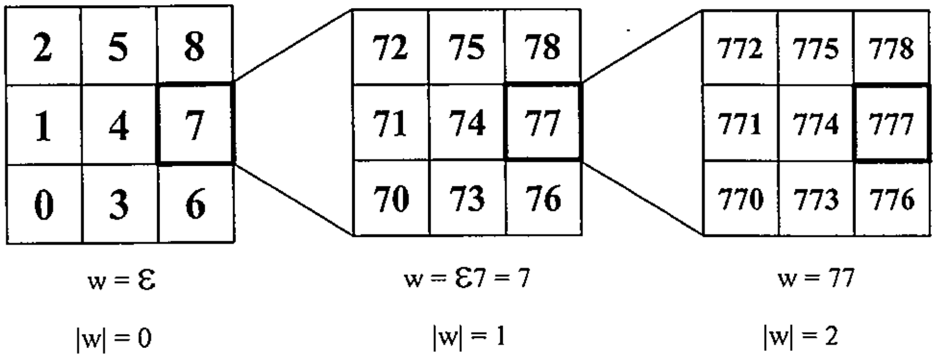
3.1 รูปภาพดิจิตอลและการแบ่งแบบโนนาทรี (Nona – Tree)

ดังที่กล่าวมาแล้วในบทที่ 2 รูปภาพดิจิตอลที่จะทำการบีบอัดจะต้องทำการแบ่งรูปภาพเป็นรูปย่อยๆ เสียก่อน โดยในบทนี้จะกำหนดให้ขนาดของรูปภาพเป็น $3^n \times 3^n$ พิกเซล กล่าวคือ จะกำหนด ขนาดของรูปภาพที่มีขนาดจำกัด (finite-resolution image) จะหมายถึง รูปภาพดิจิตอลที่เป็นรูประดับสีเทา (gray-scale) ที่ประกอบด้วยจุดจำนวน $3^m \times 3^m$ พิกเซล ซึ่ง $5 \leq m \leq 8$ และ ขนาดของรูปภาพที่ไม่จำกัดขนาด (multiresolution image) จะหมายถึง รูปภาพดิจิตอลที่มีขนาด $3^n \times 3^n$ พิกเซล โดยที่ $n = 0, 1, 2, \ldots$ โดยที่แต่ละพิกเซลจะมีชุดของตัวอักษร w ที่มีความยาว n เพื่อบ่งบอกถึงตำแหน่ง หรือแอดเดรส ของแต่ละพิกเซล ซึ่ง w จะประกอบด้วยตัวอักษรใน $\Sigma=\{0, 1, 2, 3, 4, 5, 6, 7, 8\}$ ซึ่งแต่ละตัวอักษรใน Σ จะแสดงถึงตำแหน่งของรูปย่อยที่ถูกแบ่งด้วยวิธีโนนาทรี ในส่วนของรูปภาพต้นฉบับจะกำหนดให้มีตำแหน่งเป็น ε ซึ่งเมื่อเขียนเป็นต้นไม้จะได้ว่า รูปภาพต้นฉบับเป็นราก (root) และจะได้ว่าแต่ละรูปย่อยซึ่งมีตัวอักษร w เป็นสมาชิกใน Σ จะเป็นลูกดังที่ได้แสดงดังรูป 3.1



รูปที่ 3.1 แสดงถึงตำแหน่งของรูปย่อย

ทุกๆ $w \in \Sigma^*$ ที่มีความยาว k โดยที่แต่ละ w ที่เป็นแอดเดรสของโหนดจะมีเพียงหนึ่งเดียวของโนนาทรี ซึ่งจะมีความลึก (depth) k โดยที่ลูกๆของ w จะมีแอดเดรสเป็น w0, w1, w2, w3, w4, w5, w6, w7, w8 ซึ่งจะมีความยาว k+1 ดังที่ได้แสดงดังรูป 3.2



รูปที่3.2 แสดงแอดเดรสของรูปย่อยและความยาวของแอดเดรส

3.2 ออโตมาตาแบบถ่วงน้ำหนัก

ดังที่กล่าวไว้ในหัวข้อที่แล้วว่าแต่ละพิกเซลจะมีตัวอักษรกำกับเพื่อบ่งบอกแอดเดรส จึงทำให้สามารถที่จะกำหนดได้รูปแบบฟังก์ชันของรูปภาพใดๆ ว่าเป็นฟังก์ชันจำนวนจริงบน Σ^* ซึ่งมีความสัมพันธ์กับ w ซึ่งเป็นแอดเดรสของรูปย่อย และค่าเฉลี่ยความเข้มแสงของรูปย่อยนั้น ว่าเป็นฟังก์ชัน $f:\Sigma^* \rightarrow \Re$ ซึ่งเรียกว่าเป็นเอเวอเรจพรีเซิฟวิ่งฟังก์ชัน (average preserving function) หรือเรียกย่อๆ ว่า เอพีฟังก์ชัน (ap-function)

โดยกำหนดว่า $f:\Sigma^* \rightarrow \Re$ เป็นเอพีฟังก์ชันก็ต่อเมื่อ

$$f(w) = \frac{1}{|\Sigma|}\sum_{i=0}^{3} w_i \qquad (3.1)$$

สำหรับแต่ละ $w \in \Sigma^*$

เอพีฟังก์ชัน *f* จะถูกแทนค่าด้วยแอดเดรส *w* ของรูปภาพย่อย เมื่อรูปภาพต้นฉบับหรือรากจะถูกแทนค่าด้วย $f(\varepsilon)$ ดังนั้นรูปภาพย่อยจะเป็น $f(0), f(1), \ldots, f(8)$ เมื่อ *w* มีค่า $k=1$ จะสังเกตได้ว่า $f(w)$ จะเป็นค่าเฉลี่ยความเข้มแสงของรูปย่อย *w* เมื่อรูปที่กำหนดมาให้เป็นรูปที่อยู่ในลักษณะการไล่ระดับสีเทา (grey-tone)

กำหนดให้เซตของฟังก์ชั่น $F:\Sigma^n \rightarrow \Re$ มีนิยามการกระทำบนการบวกและการคูณกับจำนวนจริงดังนี้

$$(f_1 + f_2)(w) = f_1(w) + f_2(w)$$

สำหรับทุกๆ ฟังก์ชัน $f_1, f_2 : \Sigma^* \rightarrow \Re$ และ $w \in \Sigma^*$

$$(cf)(w) = cf(w)$$

สำหรับทุกๆ ฟังก์ชั่น $f : \Sigma^* \rightarrow \Re$ และ $c \in \Re$

เซตของเอพีฟังก์ชัน f จะอยู่ในรูปของปริภูมิย่อยเชิงเส้น (lenear subspace) เพราะทุกๆ ผลบวกเชิงเส้นของ เอพีฟังก์ชันเป็นเอเวอเรจพรีเซิฟวิ่ง (average preserving) โดยที่ผลบวกจะหมายถึงการที่เกิดจากการรวมกันของความเข้มแสง และการคูณโดยจำนวนจริงจะเป็นการเพิ่มความเข้มแสงเป็นจำนวนทวีคูณ

การกระทำทั้ง 2 นี้จะใช้สำหรับการคำนวณเพื่อการสร้างออโตมาตาแบบถ่วงน้ำหนักโดยจะหาความสัมพันธ์กันระหว่างรูปย่อยซึ่งจะคิดจากค่าเฉลี่ยความเข้มแสง ความสัมพันธ์นี้จะหาได้จากการรวมเชิงเส้น (linear combination) และนำความสัมพันธ์ที่หาได้มาเขียนเป็นออโตมาตาแบบถ่วงน้ำหนักโดยรายละเอียดของออโตมาตามีดังนี้

ส่วนประกอบของออโตมาตาแบบถ่วงน้ำหนัก A จะประกอบด้วยสมาชิก 5 ตัว คือ Q, Σ, W_a, I, F ซึ่งมีรายละเอียดดังนี้

1. Q เป็นเซตจำกัดของโหนด หรือจะเรียกว่าสถานะ (states) ซึ่งจะใช้เป็นตัวแทนของรูปย่อยแต่ละรูป โดยการเก็บค่าเฉลี่ยความเข้มแสงไว้
2. Σ เป็นเซตจำกัดของตัวอักษร, จะเป็นฉลากของขอบ (edge) หรือของทางเดิน
3. $W_a : QxQ \longrightarrow \Re$ เป็นฟังก์ชันถ่วงน้ำหนัก (weight function) ซึ่งเป็นการเปลี่ยนสถานะหนึ่งไปยังอีกสถานะหนึ่งโดยที่มีฉลาก $a \in \Sigma$ กำกับอยู่แต่ละทางเดินพร้อมกับค่าน้ำหนักที่จะได้จากการเปลี่ยนสถานะหรือโหนด
4. $I : Q \longrightarrow (-\infty, \infty)$ เป็นการแจกแจงค่าเริ่มต้น (initial distribution) เพื่อบอกโหนดเริ่มต้นของออโตมาตาแบบถ่วงน้ำหนัก
5. $F: Q \longrightarrow (-\infty, \infty)$ เป็นการแจกแจงค่าสิ้นสุด (final distribution) ซึ่งค่าที่ได้จะเป็นค่าเฉลี่ยความเข้มแสงของแต่ละรูปย่อย

จะกล่าวว่า $(p,a,q) \in Q \times \Sigma \times Q$ เป็นทางเดินเพื่อที่จะเปลี่ยนสถานะ(transition) ของ A ซึ่งถ้า $W_a(p,q) \neq 0$ ซึ่งเป็นค่าถ่วงน้ำหนัก โดยที่ $a \in \Sigma$ และ $p, q \in Q$

สำหรับ $|Q| = n$ จะสามารถเขียนการเปลี่ยนสถานะให้อยู่ในรูปของเมตริกซ์ W_a ได้ซึ่งมีขนาดเป็น $n \times n$ ทุกๆ $a \in \Sigma$ และ I, F ซึ่งเป็นเวกเตอร์ที่มีขนาด n โดยที่เวกเตอร์ $I = [1 \;\; 0 \;\; 0 \;\; \ldots \;\; 0]$ เพื่อที่จะบ่งบอกว่าโหนดที่มีค่าการแจกแจงการเริ่มต้นเป็น 1 เป็นโหนดเริ่มต้น ดังนั้นโหนดอื่นๆ จึงมีค่าเป็น 0

เนื่องจากการกำหนดให้ WFA A เป็นรูปภาพดิจิตอล f_A จึงทำให้ได้ว่า

$$f_A(a_1 a_2 \ldots a_k) = IW_{a_1} W_{a_2} \ldots W_{a_k} F \tag{3.2}$$

สำหรับทุกๆ $k \geq 0$ และ $a_1 a_2 \ldots a_k \in \Sigma^*$

3.2.1 วิธีการเข้ารหัสแบบโนนาทรี

ในการแบ่งแบบโนนาทรี หรือ แบ่งเก้าจะได้ว่า $\Sigma=\{0, 1, 2, 3, 4, 5, 6, 7, 8\}$ และเนื่องจากรูปดิจิตอลใดๆ จะสามารถเขียนให้อยู่ในรูปของเอฟีฟังก์ชันได้ ดังนี้เราจะสามารถเขียนการเอฟีฟังก์ชันเมื่อทำการแบ่งแบบโนนาทรีได้ดังนี้

$$f(w) = \frac{1}{|\Sigma|} \sum_{i=0}^{9} w_i \tag{3.3}$$

สำหรับแต่ละ $w \in \Sigma^*$

หรือถ้าหากทำการแจกแจงออกมาก็จะได้ดังนี้

$$f(w) = \frac{1}{9}(w0 + w1 + w2 + w3 + w4 + w5 + w6 + w7 + w8) \tag{3.4}$$

โดยคุณสมบัตินี้ทำให้สามารถนำมาคำนวณความสัมพันธ์เพื่อสร้างออโตมาตาถ่วงน้ำหนักได้ดังนี้

กระบวนการการสร้างออโตมาตาแบบถ่วงน้ำหนัก (Inference Algorithm)

สำหรับรูปภาพใดๆ ที่กำหนดให้เป็นเอฟีฟังก์ชัน $f{:}\Sigma^* \to \Re$ ดังนั้นจะสามารถสร้าง WFA M ได้ ซึ่งในการสร้างจะมีตัวแปรและกระบวนการสร้างดังต่อไปนี้

N	เป็นตัวชี้ไปยังสเตทสุดท้ายที่พึ่งสร้าง
i	เป็นตัวชี้ไปยังสเตทที่จะทำการสร้างขอบไปยังโหนดอื่นๆ
ϕ_p	เป็นรูปย่อยซึ่งถูกแทนด้วนโหนด p
F(x)	เป็นการแจกแจงค่าสิ้นสุดของรูปย่อยซึ่งจะมีแอดเดรสเป็น x ซึ่งจะหาค่าได้จาก $f_{avg}(x) = f(x)$
$\gamma{:}Q \to \Sigma^*$	เป็นฟังก์ชันระหว่างสเตทและรูปย่อย

ค่านำเข้า : รูปภาพ M ซึ่งมีขนาด $2^k \times 2^k$

ค่าส่งออก : WFA M ที่เป็นการเข้ารหัสของรูปภาพ M

1. กำหนดให้ $N = 0, i = 0, F(q_0) = f(\varepsilon), \gamma(q_0) = \varepsilon$
2. พิจารณาโหนด q_i เมื่อ $w = \gamma(q_i)$ สำหรับทุกๆ แต่ละ

   $a \in \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

   2.1 ถ้าสามารถหาค่าสัมประสิทธิ์ $c_0, c_1, \ldots, c_N$ โดยที่

   $$f_{wa}=c_0 f_{\gamma(q0)} + c_1 f_{\gamma(q1)} + \ldots + c_N f_{\gamma(qN)}$$

   แล้วจะกำหนดให้ $W_a(q_i,q_j)=c_j$ สำหรับ $j=0, 1, \ldots, N$

   2.2 ถ้าไม่สามารถหาค่าสัมประสิทธิ์ได้ ให้กำหนด

   $\gamma(q_{N+1})=wa$, $F(q_{N+1})= f_{wa}$ และ $W_a(q_i, q_{N+1}) = 1, N = N + 1$
3. กำหนดให้ $i = i + 1$, ถ้า $i \leq N$ ให้กลับไปทำยังข้อ 2.
4. กำหนด $I(q_0) = 1, I(q_j) = 0$ สำหรับ $j = 1, \ldots, N$ เมื่อ I เป็นค่าการแจกแจงค่าเริ่มต้นของ M

ในกระบวนการสร้างนี้จะสังเกตเห็นได้ว่าจะมีการสร้างโหนดใหม่ขึ้นมาเมื่อไม่สามารถหาค่าสัมประสิทธิ์ได้ของแต่ละโหนดได้ ดังนั้นจะสามารถกล่าวได้ว่าทุกๆ โหนดนั้นเป็นอิสระต่อกัน (Independent) เช่น สำหรับโหนด ϕ_i ใดๆ เมื่อ $i = 1, 2, \ldots, n$ จะไม่สามารถแสดงได้ด้วยผลรวมเชิงเส้นของโหนดอื่นๆ ดังสมการ

$$c_0\phi_0 + c_1\phi_1 + \ldots + c_n\phi_n = 0 \tag{3.5}$$

ซึ่งจะมีคำตอบของผลรวมเชิงเส้นคือ $[c_0, c_1, \ldots, c_n] = [0, 0, \ldots, 0]$ เพียงคำตอบเดียว ดังนั้นกระบวนการสร้างไฟไนต์ออโตมาต้าแบบถ่วงน้ำหนักนี้จึงสามารถสร้างไฟไนต์ออโตมาต้าแบบถ่วงน้ำหนักที่มีจำนวนโหนดน้อยที่สุดแล้ว

ในกระบวนการขั้นตอนในข้อที่ 2. ซึ่งจะต้องแก้สมการเพื่อที่จะหาค่าสัมประสิทธิ์คงที่ $[c_0, c_1, \ldots, c_N]$ จากสมการ

$$f_{wa}=c_0\phi_0 + c_1\phi_1 + \ldots + c_N\phi_N \tag{3.6}$$

ในส่วนนี้จะนำรูปย่อยของรูป w มาสร้างเป็นระบบสมการผลรวมเชิงเส้นของรูปภาพซึ่งถูกแทนที่ด้วยโหนดที่ได้ถูกสร้างขึ้นมาแล้ว ซึ่งในการบวนการสร้างระบบสมการเชิงเส้นนี้ จะอาศัย

คุณสมบัติที่ว่ารูปภาพ M ที่จะทำการเข้ารหัสนี้จะมีคุณสมบัติของเอฟีฟังก์ชัน และ ϕ_j เมื่อ $j = 0, 1, \ldots, N$ เป็นตัวแปรที่เก็บค่าเฉลี่ยความเข้มแสงของแต่ละโหนดซึ่งเป็นตัวแทนของรูปย่อยแต่ละรูปที่ถูกนำมาสร้างเป็นโหนดใหม่เมื่อไม่สามารถแก้สมการค่าสัมประสิทธิ์ได้ดังนั้น

$$\phi_i = \frac{1}{9}[\phi_i(0) + \phi_i(1) + \phi_i(2) + \phi_i(3) + \phi_i(4) + \phi_i(5) + \phi_i(6) + \phi_i(7) + \phi_i(8) + \phi_i(9)] \qquad (3.7)$$

จากสมการ (3.7) และสมการที่ (3.6) ก็จะสามารถสร้างระบบสมการเชิงเส้นได้ดังนี้

$$\begin{gathered} f_{wa} = c_0\phi_0 + c_1\phi_1 + \ldots + c_N\phi_N \\ \Downarrow \\ f_{wa}(0) = c_0\phi_0(0) + c_1\phi_1(0) + \ldots + c_N\phi_N(0) \\ f_{wa}(1) = c_0\phi_0(1) + c_1\phi_1(1) + \ldots + c_N\phi_N(1) \\ f_{wa}(2) = c_0\phi_0(2) + c_1\phi_1(2) + \ldots + c_N\phi_N(2) \\ f_{wa}(3) = c_0\phi_0(3) + c_1\phi_1(3) + \ldots + c_N\phi_N(3) \\ f_{wa}(4) = c_0\phi_0(4) + c_1\phi_1(4) + \ldots + c_N\phi_N(4) \\ f_{wa}(5) = c_0\phi_0(5) + c_1\phi_1(5) + \ldots + c_N\phi_N(5) \\ f_{wa}(6) = c_0\phi_0(6) + c_1\phi_1(6) + \ldots + c_N\phi_N(6) \\ f_{wa}(7) = c_0\phi_0(7) + c_1\phi_1(7) + \ldots + c_N\phi_N(7) \\ f_{wa}(8) = c_0\phi_0(8) + c_1\phi_1(8) + \ldots + c_N\phi_N(8) \end{gathered} \qquad (3.8)$$

หรือจะกล่าวได้ว่า ถ้ากำหนดให้ขนาดของรูปย่อยที่กำลังถูกพิจารณามีขนาดเป็น 2^kx2^k แล้ว จะสามารถสร้างสมการให้อยู่ในรูปต่อไปนี้ได้

$$f_{wa,2^k\times 2^k} = c_0\phi_{0,2^k\times 2^k} + c_1\phi_{1,2^k\times 2^k} + \ldots + c_N\phi_{N,2^k\times 2^k} \qquad (3.9)$$

เมื่อ $\phi_{i,2^k\times 2^k}$ หมายถึงรูปที่มีขนาด 2^kx2^k ที่ถูกแทนด้วนโหนดที่ q_i และ

$f_{wa,2^k\times 2^k}$ หมายถึงรูปย่อยที่กำลังถูกพิจารณาซึ่งมีขนาด 2^kx2^k ซึ่งจะมีตำแหน่งแอดเดรสเป็น wa

ดังนั้นจึงจะสามารถสร้างระบบสมการผลรวมเชิงเส้นได้ดังนี้

$$
\begin{aligned}
f_{wa}(1,1) &= c_0\phi_0(1,1)+c_1\phi_1(1,1)+\ldots+c_N\phi_N(1,1) \\
f_{wa}(1,2) &= c_0\phi_0(1,2)+c_1\phi_1(1,2)+\ldots+c_N\phi_N(1,2) \\
&\ldots \\
f_{wa}(1,2^k) &= c_0\phi_0(1,2^k)+c_1\phi_1(1,2^k)+\ldots+c_N\phi_N(1,2^k) \\
f_{wa}(2,1) &= c_0\phi_0(2,1)+c_1\phi_1(2,1)+\ldots+c_N\phi_N(2,1) \\
&\ldots \\
f_{wa}(2,2^k) &= c_0\phi_0(2,2^k)+c_1\phi_1(2,2^k)+\ldots+c_N\phi_N(2,2^k) \\
&\ldots \\
f_{wa}(2^k,2^k) &= c_0\phi_0(2^k,2^k)+c_1\phi_1(2^k,2^k)+\ldots+c_N\phi_N(2^k,2^k)
\end{aligned} \tag{3.10}
$$

เมื่อสามารถสร้างระบบสมการผลรวมเชิงเส้นได้แล้ว จะสามารถสร้างให้อยู่ในรูปของเมตริกซ์ได้ดังนี้

$$[A][C] = [B] \tag{3.11}$$

เมื่อ [A] เป็นเมตริกซ์ขนาด $4^k \times (n+1)$ ซึ่งในแนวคอลัมภ์จะมีสมาชิกตั้งแต่ (1,1) (1,2)…(1,2k) (2,1)…(2,2k)…(2k,1)…(2k,2k) เป็นค่าความเข้มแสงของพิกเซลของ $\phi_i(x,y)$ ที่มีจำนวนทั้งหมด $2^k \times 2^k$ พิกเซล

[C] เป็นเมตริกซ์ขนาด (n+1)x1 ซึ่งเป็นค่าสัมประสิทธิ์ที่จะต้องการหา

[B] เป็นเมตริกซ์ขนาด $4^k \times 1$ ซึ่งเป็นเมตริกซ์ที่มีสมาชิกเป็นค่าความเข้มแสงของแต่ละพิกเซลของรูป f_{wa}

เมื่อได้ระบบสมการผลรวมเส้นแล้วก็จะทำการหาผลเฉลยของระบบสมการนี้ ซึ่งกระบวนการวิธีที่จะใช้แก้ปัญหาของสมการนี้ก็มีได้หลายอย่าง เช่น Gaussian Elimination, Row Echelon, LU-Decomposition แต่ในการแก้สมการเหล่านี้ในแต่ละครั้งจำเป็นที่จะต้องให้จำนวนของสมการมากกว่าตัวแปร

3.2.2 วิธีการถอดรหัส

เมื่อสามารถหา WFA M ได้แล้วโดยปกติเมื่อจะทำการหาค่าของ $f_M(w)$ สำหรับ $w \in \Sigma^n$, เมื่อกำหนดให้ $n \geq 0$ จะต้องหาทางเดินทุกๆ เส้นทางที่มีความยาว n ที่อยู่ใน WFA *M* โดยการหาผลคูณของน้ำหนักของขอบที่ผ่าน และกำหนดค่าลงในพิกเซลของรูปที่สอดคล้องกัน แต่เนื่องจากวิธีนี้จะมีกรณีที่แย่ที่สุดเมื่อทุกๆ โหนดมีเส้นทางเชื่อมหากันครบทุกฉลากจะทำให้ทางเดิน

มีมาก หลังจากนั้นจึงนำผลรวมของน้ำหนักของขอบคูณกับค่าแจกแจงการสิ้นสุด มาหาค่าของพิกเซล

เมื่อกำหนดค่า Wa ; $a \in \Sigma$ พร้อมทั้ง I และ F ก็จะสามารถคำนวณหาหาค่า $f_M(w)$ สำหรับ $w \in \Sigma^n$, เมื่อกำหนดให้ $n \geq 0$

โดยกำหนดให้ $\psi_p(w)$ สำหรับ $p \in Q$ เป็นรูปภาพของโหนด p

กระบวนการถอดรหัส

ค่านำเข้า : WFA M, I และ F และจำนวนเต็มบวก n

ผลลัพธ์ : ค่าของ $f_A(w)$ สำหรับทุกๆ $w \in \Sigma^*$

1. กำหนด $\psi_p(\varepsilon) = F(p)$ สำหรับทุกๆ $p \in Q$
2. ทำตามข้อ 3. สำหรับ $i = 1, 2, \ldots, n$
3. สำหรับทุกๆ $p \in Q$, $w \in \Sigma^{i-1}$ และ $a \in \Sigma$

$$\psi_p(aw) = \sum_{q \in Q} W_a(p,q)\psi_q(w)$$

4. สำหรับแต่ละ $w \in \Sigma^n$ คำนวณ $f_M(w) = \sum_{q \in Q} I(q)\psi_q(w)$

สำหรับการวิเคราะห์การซับซ้อนของเวลา กำหนดให้ m เป็นจำนวนโหนด ค่าความซับซ้อนในข้อที่ 1, 3 และ 4 คือ O(m), $O(m \cdot m \cdot |\Sigma|^{i-1} \cdot |\Sigma|)$= O(m2·|Σ|i) และ O(m·|Σ|n) แต่จะเห็นว่าในข้อ 3 ยังต้องวนลูปตามข้อ 2 อีก ดังนั้นค่าความซับซ้อนของเวลาในข้อ 2,3 คือ O(m(|Σ|+|Σ|2+|Σ|3+…+|Σ|n)) = $O(m^2 \cdot |\Sigma|n)$ ดังนั้นจึงสามารถสรุปค่าการซับซ้อนของเวลาในกระบวนการนี้ได้คือ O(m+m2·|Σ|n+m·|Σ|n)=$O(m^2 \cdot |\Sigma|n)$

3.2.3 ตัวอย่างในการการเข้ารหัสและถอดรหัส

ในหัวข้อนี้จะแสดงให้เห็นการเข้ารหัสและถอดรหัสที่ทำการแบ่งแบบโนนาทรี โดยจะใช้รูปภาพที่มีในลักษณะที่ต่างๆ กัน เพื่อแสดงให้เห็นถึงกระบวนการในการเข้ารหัสและถอดรหัส

ตัวอย่างที่ 1 การถอดรหัสและเข้ารหัสรูปภาพ

การเข้ารหัส

ในตัวอย่างนี้จะใช้รูปที่ 3.3 เป็นตัวอย่างทำการบีบอัดซึ่งแต่ละพิกเซลจะมีค่าความเข้มแสงดังรูปที่ 3.4 ซึ่งจะมีขนาดของรูปเท่ากับ 27x27

รูปที่ 3.3 ตัวอย่างรูปที่จะทำการบีบอัด

27	28	29	......	51	52	53
26	27	28		50	51	52
25	26	27		49	50	51
	......				......	
3	4	5	......	27	28	29
2	3	4		26	27	28
1	2	3		25	26	27

รูปที่ 3.4 แสดงค่าความเข้มแสงของแต่ละพิกเซลของรูปที่ 3.3

ในขั้นตอนแรกจะกำหนดให้รูปที่ต้องการจะทำการเข้ารหัสจะเป็นโหนดที่หนึ่ง โดยที่โหนดนั้นจะเก็บค่าความเข้มแสงเฉลี่ยของรูปนั้นไว้ ดังนั้นจึงได้โหนดที่หนึ่งของออโตมาตาแบบถ่วงน้ำหนักของรูปที่ 3.10 ซึ่งมีค่าความเข้มแสงเฉลี่ยเท่ากับ 27 จากนั้นจึงทำการแบ่งรูปภาพออกเป็นเก้าส่วนซึ่งจะได้ส่วนที่หนึ่งหรือ แอดเดรส 0 คือ รูปที่ 3.5

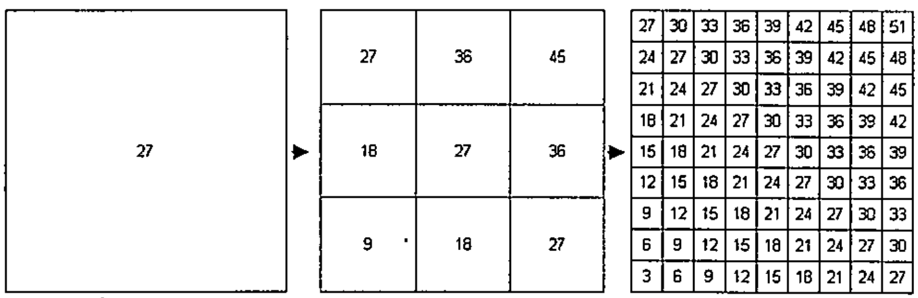
9	10	11	12	13	14	15	16	17
8	9	10	11	12	13	14	15	16
7	8	9	10	11	12	13	14	15
6	7	8	9	10	11	12	13	14
5	6	7	8	9	10	11	12	13
4	5	6	7	8	9	10	11	12
3	4	5	6	7	8	9	10	11
2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9

รูปที่ 3.5 แสดงค่าความเข้มแสงในส่วนที่หนึ่ง หรือ มีแอดเดรสเป็น 0

ซึ่งจะนำค่าสีในส่วนที่มีแอดเดรสเป็น 0 มาเปรียบเทียบค่ากับโหนดที่หนึ่งหรือรูปที่จะทำการบีบอัด ดังนั้นจะเขียนสมการได้เป็น

$$f_0 = c_0 \psi_0 \tag{3.12}$$

แต่เนื่องจากรูปที่ถูกแทนที่ด้วยโหนดที่หนึ่งนั้นมีขนาด 27x27 แต่รูปย่อยที่มีแอดเดรสเป็น 0 นั้นมีขนาดเป็น 9x9 ดังนั้นจึงจำเป็นต้องทำการลดขนาดของรูปด้วยคุณสมบัติของเอพีฟังก์ชัน หรือ ทำการแบ่งส่วนรูปภาพแค่ 2 ครั้ง โดยเมื่อแบ่งครั้งที่ 2 เสร็จแล้วจะได้รูปย่อยที่มีขนาด 3x3 พิกเซล ก็ทำการหาค่าเฉลี่ยความเข้มแสงของทุกๆ รูปย่อย ซึ่งก็จะได้รูปที่มีขนาด 9x9 ซึ่งเป็นรูปของโหนดที่หนึ่ง ดังแสดงในรูปที่ 3.6 โดยตัวเลขที่แสดงอยู่ในแต่ละช่องคือค่าเฉลี่ยความเข้มแสงของแต่ละรูปย่อย



รูปที่ 3.6 แสดงการแบ่งหรือ การย่อขนาดของรูปที่แทนด้วยโหนดที่หนึ่ง

จากนั้นจึงจำค่าสีแต่ละค่ามาสร้างเป็นสมการได้ดังนี้

$$\begin{aligned} 9 &= c_0 27 \\ 10 &= c_0 30 \\ 11 &= c_0 33 \\ 12 &= c_0 36 \\ 13 &= c_0 39 \\ &\cdots \\ 6 &= c_0 18 \\ 7 &= c_0 21 \\ 8 &= c_0 24 \\ 9 &= c_0 27 \end{aligned} \tag{3.13}$$

เมื่อทำการหาค่าสัมประสิทธิ์ c_0 ในทุกสมการแล้วจะได้ค่าเท่ากับ 1/3 ดังนั้นจึงสามารถสร้างเส้นทางจากโหนดที่หนึ่งไปยังโหนดที่หนึ่งได้ โดยที่มีฉลากเป็น 0 และมีค่าถ่วงน้ำหนักเป็น 1/3 ซึ่งอาจจะเขียนสมการแสดงความสัมพันธ์ได้ดังนี้

$$f_0 = \frac{1}{3}\psi_0 \tag{3.14}$$

จากนั้นจึงทำการพิจารณาในส่วนที่มีแอดเดรส 1 ซึ่งได้แสดงในรูปที่ 3.7

18	19	20	21	22	23	24	25	26
17	18	19	20	21	22	23	24	25
16	17	18	19	20	21	22	23	24
15	16	17	18	19	20	21	22	23
14	15	16	17	18	19	20	21	22
13	14	15	16	17	18	19	20	21
12	13	14	15	16	17	18	19	20
11	12	13	14	15	16	17	18	19
10	11	12	13	14	15	16	17	18

รูปที่ 3.7 แสดงค่าความเข้มแสงของรูปย่อยที่มีแอดเดรสเป็น 1

ดังนั้นจะสามารถสร้างสมการได้เป็น

$$\begin{aligned} 18 &= c_0 27 \\ 19 &= c_0 30 \\ 20 &= c_0 33 \\ 21 &= c_0 36 \\ 22 &= c_0 39 \\ &\ldots \\ 15 &= c_0 18 \\ 16 &= c_0 21 \\ 17 &= c_0 24 \\ 18 &= c_0 27 \end{aligned} \tag{3.15}$$

เมื่อแก้สมการแล้วจะพบว่าค่าสัมประสิทธิ์ที่หามาได้แต่ละค่าจะมีค่าไม่เท่ากัน หรืออาจจะกล่าวได้ว่าโหนดที่หนึ่งไม่มีความสัมพันธ์กับรูปย่อยที่มีแอดเดรส 1 ดังนั้นจึงจำเป็นที่จะต้องสร้างโหนดใหม่ซึ่งเป็นตัวแทนของรูปย่อยที่มีแอดเดรสเป็น 1 และสร้างเส้นทางจากโหนดที่หนึ่งไปยังโหนดที่สอง โดยที่มีฉลากเป็น 1 และมีค่าถ่วงน้ำหนักเป็น 1 และในโหนดที่สองนี้จะเก็บค่าเฉลี่ยความเข้มแสงของรูปย่อยที่มีแอดเดรสเป็น 1 ไว้ด้วย เพราะฉะนั้นจึงได้โหนดที่สองของออโตมาตาของรูปที่ 3.10

จากนั้นจึงพิจารณารูปย่อยที่มีแอดเดรสเป็น 2 ซึ่งจะสามารถแก้สมการได้ค่าคงที่ดังนี้

$$f_2 = -\frac{1}{3}\psi_0 + 2\psi_1 \tag{3.16}$$

ดังนั้นจะได้ว่าค่าคงที่ $c_0 = -\frac{1}{3}$ และ $c_1 = 2$ ซึ่งจะสามารถสร้างเส้นทางที่มีฉลาก 2 จากโหนดหนึ่งไปยังโหนดหนึ่งและโหนดหนึ่งไปยังโหนดสอง ที่มีค่าถ่วงน้ำหนักเป็น $-\frac{1}{3}$ และ 2 ตามลำดับ จากนั้นจึงคำนวณรูปย่อยที่มีแอดเดรสเป็น 3, 4, 5, 6, 7, 8 ซึ่งจะแก้สมการได้ค่าคงที่ดังนี้

$$
\begin{aligned}
f_3 &= 0\psi_0 + 1\psi_1 \\
f_4 &= -\frac{1}{3}\psi_0 + 2\psi_1 \\
f_5 &= -\frac{2}{3}\psi_0 + 3\psi_1 \\
f_6 &= -\frac{1}{3}\psi_0 + 3\psi_1 \\
f_7 &= -\frac{2}{3}\psi_0 + 3\psi_1 \\
f_8 &= -1\psi_0 + 4\psi_1
\end{aligned}
\tag{3.13}
$$

ซึ่งจะสามารถสร้างเส้นทางที่มีฉลากเป็น 3, 4, 5, 6, 7, 8 ซึ่งจะออกจากโหนดหนึ่งไปยังโหนดหนึ่ง มีค่าถ่วงน้ำหนักเป็น $0, \frac{1}{3}, -\frac{1}{3}, -\frac{2}{3}, -\frac{1}{3}, -\frac{2}{3}$ และ -1 ตามลำดับ และจากโหนดหนึ่งไปยังโหนดสองจะมีค่าถ่วงน้ำหนักเป็น 1, 2, 3, 3, 3 และ 4 ตามลำดับ

เมื่อพิจารณารูปย่อยในโหนดที่หนึ่งครบหมดแล้วจึงพิจารณาแต่รูปย่อยของรูปที่ถูกแทนด้วยโหนดที่สอง ก็คือ รูปย่อยที่มีแอดเดรสเป็น 1 ดังรูปที่ 3.7 ดังนั้นรูปย่อยที่มีแอดเดรสเป็น 0 ของโหนดที่สอง หรือรูปย่อยที่มีแอดเดรสเป็น 10 จะแสดงได้ดังรูปที่ 3.8

12	13	14
11	12	13
10	11	12

รูปที่ 3.8 แสดงรูปย่อยที่มีแอดเดรสเป็น 0 ของโหนดที่สอง

หรือ รูปย่อยที่มีแอดเดรสเป็น 10

เนื่องจากรูปที่แทนด้วยโหนดที่หนึ่งจะมีขนาดเป็น 27x27 ดังนั้นจึงจะต้องย่อขนาดของรูปให้เท่ากับรูปย่อยที่มีแอดเดรสเป็น 10 ซึ่งจะย่อรูปได้ดังรูปที่ 3.9 ก. และ รูปที่แทนด้วยโหนดที่สองจะมีขนาดเป็น 9x9 ดังนั้นจะต้องย่อขนาดของรูปให้เท่ากับรูปย่อยด้วย จึงได้ค่าสีแต่ละพิกเซลดังรูปที่ 3.9 ข.

27	36	45
18	27	36
9	18	27

ก.

18	21	24
15	18	21
12	15	18

ข.

รูปที่ 3.9 แสดงผลลัพธ์การย่อขนาดของรูป
ที่ถูกแทนด้วยโหนดที่หนึ่งและโหนดที่สอง

จากนั้นจึงนำค่าความเข้มแสงของแต่ละพิกเซลมาสร้างสมการได้ดังนี้

$$\begin{aligned}
12 &= c_0 37 + c_1 18 \\
13 &= c_0 36 + c_1 21 \\
14 &= c_0 45 + c_1 24 \\
11 &= c_0 18 + c_1 15 \\
12 &= c_0 27 + c_1 18 \\
13 &= c_0 36 + c_1 21 \\
10 &= c_0 9 + c_1 12 \\
11 &= c_0 18 + c_1 15 \\
12 &= c_0 27 + c_1 18
\end{aligned} \tag{3.14}$$

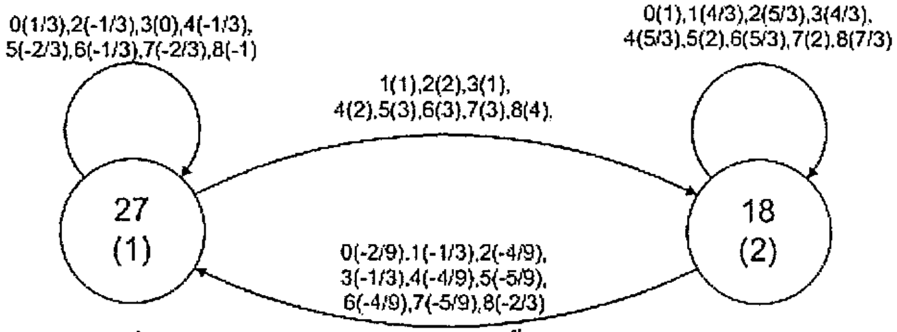
ซึ่งเมื่อแก้สมการแล้วจะสามารถหาค่าสัมประสิทธิ์และเขียนให้อยู่ในรูปสมการได้ดังนี้

$$f_{10} = -\frac{2}{9}\psi_0 + 1\psi_1 \tag{3.15}$$

เมื่อคำนวณหาค่าสัมประสิทธิ์ c_0 และ c_1 ได้แล้วจึงนำไปสร้างเส้นทางสำหรับโหนดที่สองซึ่งมีฉลากเป็น 0 ได้ดังนี้ เส้นทางที่ออกจากโหนดที่สองไปยังโหนดที่หนึ่งจะมีค่าถ่วงน้ำหนักเป็น $-\frac{2}{9}$ และเส้นทางที่ออกจากโหนดที่สองไปยังโหนดที่สอง หรือเส้นทางที่วนเข้าหาตัวเองจะมีค่าถ่วงน้ำหนักเป็น 1 ในกรณีเดียวกันสำหรับรูปย่อยอื่นๆ ของโหนดที่สองก็จะสามารถหาค่าสัมประสิทธิ์ได้ดังนี้

$$
\begin{aligned}
f_{11} &= -\frac{1}{3}\psi_0 + \frac{4}{3}\psi_1 \\
f_{12} &= -\frac{4}{9}\psi_0 + \frac{5}{3}\psi_1 \\
f_{13} &= -\frac{1}{3}\psi_0 + \frac{4}{3}\psi_1 \\
f_{14} &= -\frac{4}{9}\psi_0 + \frac{5}{3}\psi_1 \\
f_{15} &= -\frac{5}{9}\psi_0 + 2\psi_1 \\
f_{16} &= -\frac{4}{9}\psi_0 + \frac{5}{3}\psi_1 \\
f_{17} &= -\frac{5}{9}\psi_0 + 2\psi_1 \\
f_{18} &= -\frac{2}{3}\psi_0 + \frac{7}{3}\psi_1
\end{aligned}
\tag{3.16}
$$

เมื่อได้พิจารณารูปย่อยของแต่ละภาพที่ถูกแทนที่ด้วยโหนดเสร็จแล้ว ก็จะได้ไฟไนต์ออโตมาต้าแบบถ่วงน้ำหนักดังรูปที่ 3.10



รูปที่ 3.10 แสดงออโตมาตาแบบถ่วงน้ำหนักที่แทนรูปที่ 3.3

การถอดรหัส

ในการถอดรหัสของไฟไนต์ออโตมาต้าถ่วงนำหนักจะใช้วิธีการหาค่าความเข้มแสงของแต่ละพิกเซลออกมา โดยที่จะมีการกำหนดขนาดของรูปไว้ก่อนแล้วจะหาความยาวของแอดเดรสของพิกเซลเพื่อที่จะนำแอดเดรสนี้มาคำนวณหาค่าความเข้มแสง เช่น ขนาดของรูปเป็น 3×3 จะได้ความยาวของแต่ละแอดเดรสเป็น 1, $3^2 \times 3^2$ ก็จะได้ความยาวของแอดเดรสเป็น 2 เป็นต้น ส่วนการหาความเข้มแสงของแต่ละพิกเซลจะหาได้จากผลบวกของผลคูณค่าเริ่มต้นกับค่าถ่วงน้ำหนักแต่ละเส้นทางที่ผ่านกับค่าสิ้นสุดของโหนดสุดท้ายของทุกๆ เส้นทาง โดยจะสามารถเขียนเป็นสมการได้ดังนี้

$$
f(a_1a_2\ldots a_n) = I(j_0)W_{a_1}(j_0, j_1)W_{a_2}(j_1, j_2)\ldots W_{a_n}(j_{n-1}, j_n)F(j_n) \tag{3.17}
$$

เมื่อ $I(a_1)$ = เป็นค่าการแจกแจงค่าเริ่มต้นของโหนดเริ่มต้น

$F(a_n)$=เป็นค่าการแจกแจงค่าสิ้นสุดของโหนดสุดท้าย

$W_{ai}(i,j)$=เป็นค่าน้ำหนักที่ฉลากที่ a_i จากโหนด i ไปยังโหนด j โดยที่โหนด i จะได้มาจากโหนด j ของฉลากตัวข้างหน้าซึ่งจะได้มาพร้อมกับค่าถ่วงน้ำหนัก

เนื่องจากค่าการแจกแจงค่าเริ่มต้นของโหนดอื่นๆ ที่ไม่ใช่โหนดที่ 0 จะมีค่าเป็น 0 ดังนั้นค่าเฉลี่ยของความเข้มแสงที่เริ่มต้นจากโหนดอื่นๆ จะมีค่าเป็น 0 เสมอ

ถ้าหากกำหนดให้ขนาดของรูปเป็น $3^3 \times 3^3$ ซึ่งจะได้ว่าค่าความยาวของแอดเดรสเป็น 2 ดังนั้นจำนวนแอดเดรสของพิกเซลจะเท่ากับ 9×9=81 แอดเดรส ก็ได้แก่ 00, 01, 02, … , 08, 20, 21, … , 28, … , 80, 81, … , 88 ดังนั้นจากออโตมาตาแบบถ่วงน้ำหนักในรูปที่ 3.10 จะหาค่าเฉลี่ยความเข้มแสงได้ดังนี้

แอดเดรส 00 : จะได้ว่า

$$\begin{aligned} f(00) &= I(0)W_0(0, j_1)W_0(j_1, j_2)F(j_2) \\ &= 1\times 1/3\times W_0(0, j_2)F(j_2) \\ &= 1\times 1/3\times 1/3\times F(0) \\ &= 1\times 1/3\times 1/3\times 27 \\ &= 3 \end{aligned}$$

แอดเดรส 01 : จะได้ว่า

$$\begin{aligned} f(01) &= I(0)W_0(0, j_1)W_1(j_1, j_2)F(j_2) \\ &= 1\times 1/3\times W_1(0, j_2)F(j_2) \\ &= 1\times 1/3\times 1\times F(1) \\ &= 1\times 1/3\times 1\times 18 \\ &= 6 \end{aligned}$$

แอดเดรส 02 : จะได้ว่า

$$\begin{aligned} f(02) &= I(0)W_0(0, j_1)W_2(j_1, j_2)F(j_2) \\ &= 1\times 1/3\times W_2(0, j_2)F(j_2) \end{aligned}$$

จะสังเกตเห็นว่า $W_2(0,j_2)$ มีโหนดปลายทาง 2 โหนดคือ โหนด 0 และ โหนด 1 ดังนั้นจึงต้องหาผลคูณของทั้งสองเส้นทางแล้วนำมาบวกกัน คือ

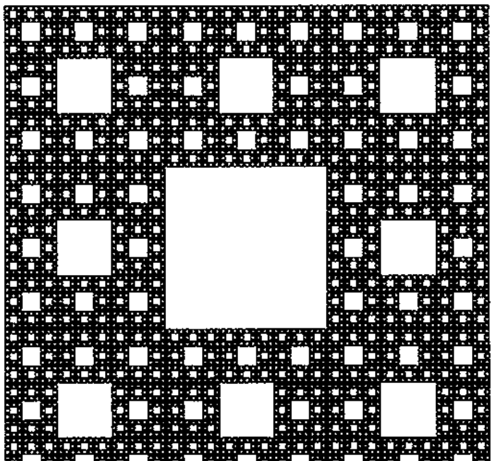
$$\begin{aligned} f(02) &= 1\times 1/3\times W_2(0, 0)F(0)+1\times 1/3\times W_2(0,1)F(1) \\ &= 1\times 1/3\times (-1/3) \times 27+1\times 1/3\times 2\times 18 \\ &= -3+12 = 9 \end{aligned}$$

จากนั้นจึงหาให้ครบทุกพิกเซล จะได้รูปที่มีค่าความเข้มแสงดังรูปที่ 3.11

27	30	33	36	39	42	45	48	51
24	27	30	33	36	39	42	45	48
21	24	27	30	33	36	39	42	45
18	21	24	27	30	33	36	39	42
15	18	21	24	27	30	33	36	39
12	15	18	21	24	27	30	33	36
9	12	15	18	21	24	27	30	33
6	9	12	15	18	21	24	27	30
3	6	9	12	15	18	21	24	27

รูปที่ 3.11 แสดงความเข้มแสงของแต่ละพิกเซลของรูปที่มีขนาด $3^2 \times 3^2$ พิกเซล

จะสังเกตเห็นว่าค่าความเข้มแสงของแต่ละพิกเซลของรูปที่ทำการถอดรหัสแล้วจะไม่เท่ากับค่าความเข้มแสงของรูปที่ทำการบีบอัด เนื่องจากรูปที่ทำการบีบอัดมีขนาดเท่ากับ $3^3 \times 3^3$ พิกเซล แต่รูปที่ทำการบีบอัดมีขนาด $3^2 \times 3^2$ พิกเซล แต่จะสังเกตเห็นได้อีกว่ารูปที่ถอดรหัสจะมีค่าเฉลี่ยความเข้มแสงแต่ละพิกเซลเท่ากับ รูปที่ 3.6 ซึ่งเป็นค่าเฉลี่ยความเข้มแสงของรูปที่ทำการลดขนาดด้วยคุณสมบัติของเอพี-ฟังก์ชัน



รูปที่ 3.12 รูปสำหรับตัวอย่างที่ 2 เพื่อการบีบอัด

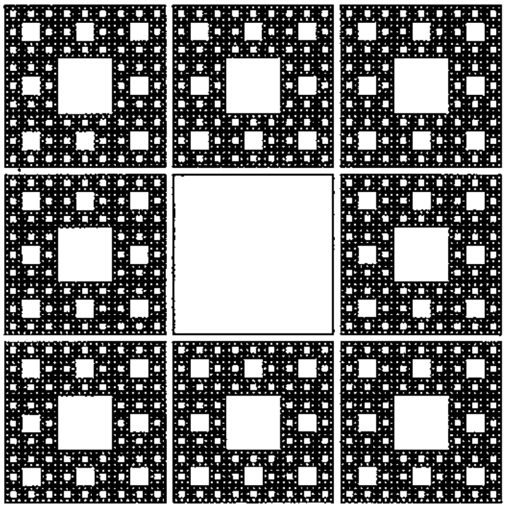
ตัวอย่างที่ 2 การถอดรหัสและเข้ารหัสรูปภาพ

การเข้ารหัส

ในการเข้ารหัสในตัวอย่างที่ 2 จะใช้รูปภาพที่ 3.12 เป็นตัวอย่างในการบีบอัดซึ่งจะแสดงให้เห็นได้ในกรณีที่รูปที่พิจารณานั้นถ้าเป็นรูปเดียวกัน จะสามารถสร้างทางเดินวนเข้าหารูปนั้นได้ทันที

โดยในขั้นตอนแรกของการบีบอัดจะต้องกำหนดให้รูปที่ต้นฉบับซึ่งมีแอดเดรสเป็น ε เป็นโหนดแรกของออโตมาตาแบบถ่วงน้ำหนัก

จากรูปที่ 3.12 เมื่อทำการแบ่งเป็น 9 ส่วนแล้วจะได้ดังรูปที่ 3.13

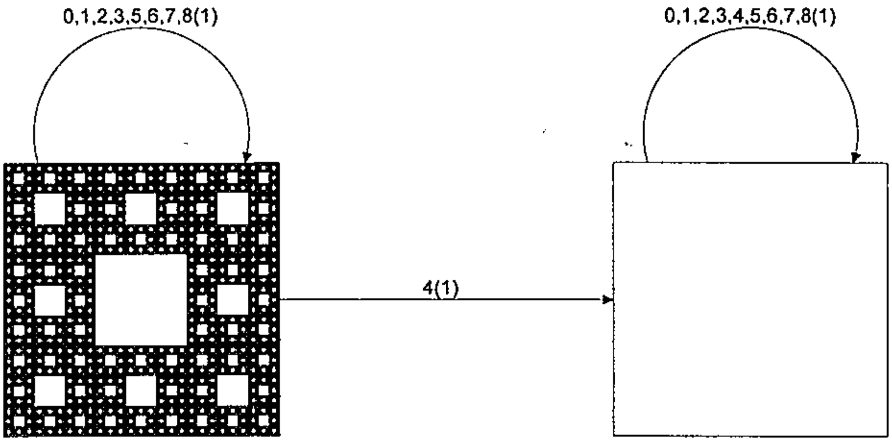


รูปที่ 3.13 แสดงการแบ่งออกเป็น 9 ส่วนของรูปที่ 3.12

เมื่อพิจารณารูปย่อยที่มีแอดเดรสเป็น 0 แล้วจะสังเกตเห็นว่าเหมือนรูปต้นฉบับที่ต้องการทำการบีบอัดดังนั้นในจะหาค่าคงที่ของสมการที่ 3.6 ได้เท่ากับ 1 ดังนั้นจึงจะได้เส้นทางของรูปย่อยแอดเดรส 0 ได้จากโหนดที่หนึ่งไปยังโหนดที่หนึ่ง ซึ่งหาพิจารณาดีๆ แล้ว จะเห็นว่ารูปย่อยที่มีแอดเดรสเป็น 2, 3, 4, 6, 7 และ 8 ก็มีลักษณะเดียวกันดังนั้นจึงสามารถสร้างทางเดิน จากโหนดที่หนึ่งไปยังโหนดที่หนึ่งได้โดยที่มีฉลากเป็น 0, 1, 2, 3, 5, 6, 7 และ 8 ซึ่งมีค่าถ่วงน้ำหนักเป็น 1 ดังรูปที่ 3.13

ส่วนรูปย่อยที่ 4 นั้นไม่เหมือนรูปโหนดที่หนึ่งดังนั้นจึงต้องสร้างโหนดใหม่โดยให้เป็นโหนดที่สองและมีฉลากเป็น 4 พร้อมทั้งค่าถ่วงน้ำหนักเป็น 1 ก็จะเสร็จสิ้นการพิจารณารูปที่แทนด้วยโหนดที่หนึ่ง

ในโหนดที่สอง จะเห็นว่าทุกๆ รูปย่อยเป็นสีขาวเหมือนกับโหนดที่สอง ดังนั้นจึงสร้างทางเดินจากโหนดที่สองไปยังโหนดที่สองโดยที่มีฉลากเป็น 0, 1, 2, 3, 4, 5, 6, 7, 8 และ9 พร้อมทั้งค่าถ่วงน้ำหนักเป็น 1 ดังนั้นจึงจะได้ออโตมาตาที่แทนด้วยรูปของรูปที่ 3.12 ได้ดังรูป 3.13



รูปที่ 3.14 แสดงออโตมาตาแบบถ่วงน้ำหนักที่แทนรูปที่ 3.12

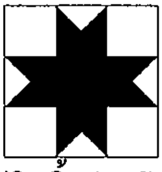
การถอดรหัส

ในการถอดรหัสออโตมาตาในรูปที่ 3.13 จะได้รูปที่มีลักษณะของรูปย่อยที่ 0, 1, 2, 3, 5, 6, 7, 8 และ9 จะมีลักษณะเหมือนโหนดที่หนึ่ง และรูปย่อยที่ 4 จะเป็นสีขาวเสมอ หรือจะกล่าวได้ว่าทุกๆ รูปย่อยที่มีแอดเดรสเป็น w4 เมื่อ $w \in \Sigma^*$ จะมีค่าความเข้มแสงเป็นสีขาว

ตัวอย่างที่ 3 การเข้ารหัสและถอดรหัสรูปภาพ

การเข้ารหัส

ในตัวอย่างนี้จะแสดงการสร้างออโตมาตาแบบถ่วงน้ำหนักสำหรับรูปริบบิ้นที่แสดงในรูปที่ 3.15



รูปที่ 3.15 รูปริบบิ้นสำหรับการเข้ารหัส

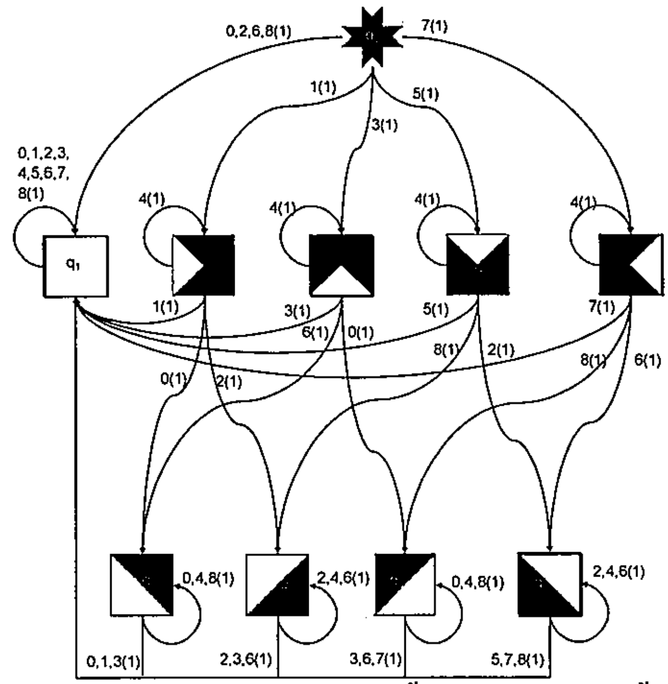
รูปริบบิ้นนี้มีขนาด 243x243 พิกเซล ในขั้นแรกจะกำหนดให้รูปริบบิ้นเป็น q_0 และ $F(q_0) = 1275/9$ แล้วจึงพิจารณารูปย่อยต่างๆ ของ q_0 ตั้งแต่ 0 ถึง 8 ซึ่งแต่ละรูปจะมีความยาวของแอดเดรสเป็น 1 โดยจะสังเกตเห็นว่ารูปย่อย f_0 และ f_1 ไม่สามารถแสดงอยู่ในรูปของผลรวมเชิงเส้นของ f_ε ได้ เมื่อกำหนดให้ f_w เป็นค่าความเข้มแสงของรูปย่อยที่มีแอดเดรสเป็น w จากนั้นจึงทำขั้นตอนที่ 2.2 ของอัลกอริทึมในการเข้ารหัส คือกำหนด $W_0(q_0,q_1) = 1$ และ $F(q_1) = 255$; $W_0(q_0,q_2) = 1$ และ $F(q_2) = 576/9$ ในส่วนของรูปย่อยที่ 2 นั้นจะเหมือนกับโหนด q_1 ซึ่งมันจะสามารถแสดงอยู่ในรูปของ $f_2 = 1 \cdot f_1$ ได้ ดังนั้นจึงกำหนดให้ $W_2(q_0,q_1) = 1$ แล้วพิจารณารูปย่อยที่มีแอดเดรสเป็น 3 จะไม่สามารถ

แสดงเป็นผลรวมเชิงเส้นของทุกๆ โหนดได้ ดังนั้นจึงให้รูปย่อยที่มีแอดเดรสเป็น 3 เป็นโหนดใหม่ ซึ่ง $W_3(q_0,q_3) = 1$ และ $F(q_3) = 576/9$ ค่าความเข้มแสงของรูปย่อย f_4 เป็น 0 ซึ่งจะสามารถหาค่าสัมประสิทธิ์ได้เป็น 0 ดังนั้นจึงไม่ต้องสร้างทางเดินสำหรับรูปย่อยนี้ สำหรับรูปย่อย f_5 นี้ก็ไม่สามารถที่จะแสดงอยู่ในรูปของผลรวมเชิงเส้นกับโหนดอื่นๆ ได้ ดังนั้นต้องสร้างโหนดใหม่ โดยให้รูปย่อย f_5 นี้เป็นโหนดใหม่ พร้อมทั้งกำหนด $W_5(q_0,q_4)$ และ $F(q_4) = 576/9$ รูปย่อย f_6 นี้จะเหมือนกับรูปย่อย f_0 ดังนั้นกำหนดให้ $W_6(q_0,q_1) = 1$ รูปย่อย f_7 ก็ไม่สามารถแสดงอยู่ในรูปของผลรวมเชิงเส้นได้อีกเช่นกัน ดังนั้นจึงสร้างโหนดใหม่ขึ้นมา โดยใช้รูปย่อย f_7 พร้อมทั้งกำหนด $W_7(q_0,q_5)$ และ $F(q5) = 576/9$

ขั้นตอนต่อไป พิจารณารูปย่อยที่มีความยาวของที่อยู่เป็น 2 พิจารณาโหนด q_1 ทุกๆ รูปย่อยจะเหมือนกับโหนด q_1 ดังนั้นจึงสร้างทางเดิน 8 ทางเดินสำหรับ q_1 ไปยัง q_1 ซึ่งมีฉลากเป็น 0, 1, ... , 8 ซึ่งมีค่าถ่วงน้ำหนักเป็น 1 โหนดต่อไปที่จะพิจารณาคือโหนด q_2 เมื่อพิจารณาแต่ละรูปย่อยของโหนด q_2 แล้วจะพบว่า รูปย่อยที่มีแอดเดรสเป็น 0 ไม่สามารถแสดงเป็นผลรวมเชิงเส้นได้ จึงต้องสร้างโหนดใหม่ซึ่งมีทางเดินเป็น $W_0(q_2,q_6) = 1$ และ $F(q_6) = 128$ แต่ในรูปย่อยที่ 1 จะสามารถแสดงอยู่ในรูปของ $f_{11} = f_0$ ได้ จึงสามารถสร้างทางเดินจาก $W_1(q_2,q_1) = 1$ ในส่วนของรูปย่อยที่ 2 ของโหนด q_2 ก็ไม่สามารถแสดงให้อยู่ในรูปของผลรวมเชิงเส้นกับโหนดใดๆ ได้ จึงกำหนดให้ $W_2(q_2,q_7) = 1$ และ $F(q_7) = 128$ รูปย่อยที่ 4 ของโหนด q_2 เหมือนกับโหนด q_2 จึงกำหนดให้ $W_4(q_2,q_2) = 1$ ส่วนรูปย่อยที่ 3 และรูปย่อยที่เหลือทั้งหมดของโหนด q_2 เป็นรูปสี่เหลี่ยมสีดำมีค่าความเข้มแสงเป็น 0 ซึ่งสามารถหาค่าสัมประสิทธิ์ของผลรวมเชิงเส้นได้เป็น 0 ในกรณีนี้จึงไม่จำเป็นต้องสร้างทางเดิน สำหรับโหนด q_3, q_4 และ q_5 ก็เช่นเดียวกันกับโหนด q_2 คือพิจารณาแต่ละรูปย่อยของแต่ละโหนด ถ้าไม่สามารถแสดงให้อยู่ในรูปของผลรวมเชิงเส้นได้ ก็ให้สร้างโหนดใหม่

สำหรับขั้นตอนต่อไปทุกๆ โหนดจะมีความลึกเท่ากับ 3 หรือความยาวของแอดเดรสเป็น 3 สำหรับโหนดแรกของขั้นตอนนี้ก็คือ โหนด q_6 จะสังเกตเห็นว่ารูปย่อยที่ 0, 4 และ 8 ของโหนด q_6 จะเหมือนกับโหนด q_6 เอง จึงสร้างทางเดิน $W_0(q_6,q_6)=1$, $W_4(q_6,q_6)=1$ และ $W_8(q_6,q_6)=1$ ส่วนรูปย่อยที่ 1, 2 และ 5 เหมือนกับโหนด q_1 จึงกำหนดให้ $W_1(q_6,q_1)=1$, $W_2(q_6,q_1)=1$ และ $W_5(q_6,q_1)=1$ ส่วนรูปย่อยอื่นๆ เป็นรูปสี่เหลี่ยมสีดำซึ่งหาค่าสัมประสิทธิ์ได้เป็น 0 จึงไม่ต้องสร้างทางเดินไปยังโหนดใดๆ สำหรับโหนด q_7, q_8 และ q_9 ก็เช่นเดียวกันกับโหนด q_6 รูปย่อยที่เหมือนกับตัวเองก็ให้สร้างทางเดินเข้าหาตัวเอง ส่วนรูปย่อยที่มีสีขาวก็ให้สร้างทางเดินไปยังโหนด q_1

สุดท้ายกำหนดค่าของการแจกแจงค่าเริ่มต้นให้แต่ละโหนด โดยที่โหนด $I(q_0) = 1$ ส่วนโหนดอื่นๆ เท่ากับ 0 จะได้ออโตมาตาถ่วงน้ำหนักที่แทนรูปที่3.14 ได้ดังรูปที่ 3.16



รูปที่ 3.16 รูปออโตมาตาแบบถ่วงน้ำหนักที่แทนรูปริบบิ้น

การถอดรหัส

ในส่วนนี้จะแสดงวิธีการถอดรหัสออโตมาตาแบบถ่วงน้ำหนักในรูปที่ 3.15 ไปยังรูปต้นฉบับ โดยที่ในรูปที่ 3.15 จะแสดงให้เห็นถึงทางเดินจากโหนด i ไปยังโหนด j ซึ่งมีฉลากเป็น a ก็ต่อเมื่อ $Wa(q_i,q_j) = r;\ r \neq 0$

ค่าความเข้มแสงของแต่ละพิกเซลจะหาได้จากออโตมาต้าแบบถ่วงน้ำหนักดังนี้ : สำหรับแอดเดรสของแต่ละพิกเซลแล้ว ให้ใช้แอดเดรสของพิกเซลนั้นเป็นทางเดิน และนำค่าถ่วงน้ำหนักของแต่ละทางเดินที่เดินผ่านคูณกับค่าเริ่มต้นของโหนดแรกและคูณกับค่าสิ้นสุดของโหนดสุดท้าย ดังนั้น $f_A(w)$ เป็นเป็นผลรวมของของค่าถ่วงน้ำหนักของทุกทางเดินที่คูณกับค่าเริ่มต้นของโหนดแรกกับค่าสิ้นสุดของโหนดสุดท้ายของ w

เช่น ค่าความเข้มแสงของรูปย่อยที่มีแอดเดรสเป็น 142 ซึ่งจะแทนด้วย f(142) ซึ่งจะสามารถคำนวณได้จาก

$$
\begin{aligned}
f(142) &= I(q_0)\cdot W_1(q_0,q_2)\cdot W_4(q_2,q_2)\cdot W_2(q_2,q_7)\cdot F(q_7)\\
&= 1\cdot 1\cdot 1\cdot 128\\
&=128
\end{aligned}
$$

ถ้าในโหนดใดไม่มีทางเดินไปยังโหนดอื่น จะกำหนดให้โหนดนั้นมีค่าถ่วงน้ำหนักเป็น 0 ซึ่งจะทำให้ผลคูณทั้งหมดเป็นศูนย์ไปด้วย เช่น $f(50) = 1\cdot 1\cdot 0\cdot 0=0$

บทที่ 4
การทดสอบประสิทธิภาพการแปลงข้อมูล

ในบทนี้จะกล่าวถึงลักษณะและเหตุผลที่นำข้อมูลมาใช้ในการทดสอบ และผลของการบีบอัดข้อมูล

4.1 รูปภาพที่ใช้ในการทดสอบ

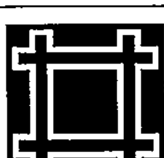
รูปภาพที่ใช้ในการทดสอบจะเป็นรูปที่มีลักษณะง่ายๆ ที่ประกอบไปด้วยรูปทางเรขาคณิต และรูปมาตรฐานในงานวิจัยทางด้านรูปดิจิตอล แต่ละรูปจะมีความเข้มแสงอยู่ระหว่างสีดำและสีขาว (ค่าความเข้มแสงอยู่ระหว่าง 0 ถึง 255) และจะมีชนิดของข้อมูลเป็นบิทแมพ โดยจะคัดเลือกเฉพาะรูปที่มีขนาด 2^mx2^m และ 3^nx3^n, เมื่อ m, n =0, 1, 2,... ที่เหมือนกัน รูปที่มีขนาดเป็น 3^mx3^m จะทำให้การแบ่งแบบโนนาทรี และรูปที่มีขนาดเป็น 2^mx2^m การแบ่งแบบควอดทรี โดยจะแบ่งเป็น 2 กลุ่มคือ

กลุ่มที่ 1 ลักษณะของรูปจะประกอบไปด้วยรูปทรงทางเรขาคณิต โดยขนาดของรูปจะมีขนาดเป็น $3^5 \times 3^5$ และ $2^8 \times 2^8$ พิกเซล ซึ่งเป็นขนาดใหญ่ที่สุดของรูปที่จะทำการทดสอบและไล่ระดับเฉดสีซึ่งจะเป็นรูปสีเทา

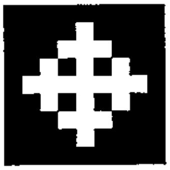
กลุ่มที่ 2 จะเป็นรูปมาตรฐานสำหรับการทดสอบทางด้านรูปภาพดิจิตอลทั่วๆไป โดยในที่นี้จะใช้รูปของเลนน่า (Lenna) ที่เป็นรูปขาวดำ และรูปที่มีลักษณะเป็นรูปสีเทา โดยจะเป็นรูปที่มีขนาดต่างๆ กัน

รายละเอียดของรูปภาพในกลุ่มที่ 1 จะเป็นดังตารางที่ 4.1

ตารางที่ 4.1 แสดงข้อมูลของรูปภาพที่จะทำการบีบอัด

รูป	ขนาดของรูปภาพ	
	$3^5 \times 3^5$	$2^8 \times 2^8$
	Ribbon.BMP ขนาดรูปภาพ 3.46 กิโลไบต์	Ribbon. BMP ขนาดรูปภาพ 3. 38 กิโลไบต์
	Sharp. BMP ขนาดรูปภาพ 3.64 กิโลไบต์	Shapr. BMP ขนาดรูปภาพ 3.87 กิโลไบต์

ตารางที่ 4.1 (ต่อ)

รูป	ขนาดของรูปภาพ	
	$3^5 \times 3^5$	$3^5 \times 3^5$
	Plus. BMP ขนาดรูปภาพ 2.76 กิโลไบต์	Plus. BMP ขนาดรูปภาพ 3.49 กิโลไบต์
	triangle. BMP ขนาดรูปภาพ 8.00 กิโลไบต์	triangle. BMP ขนาดรูปภาพ 8.00 กิโลไบต์
	Grayscale.BMP ขนาดรูปภาพ 5.36 กิโลไบต์	Grayscale.BMP ขนาดรูปภาพ 5.4 กิโลไบต์

ส่วนรายละเอียดของรูปเลนน่าในกลุ่มที่ 2 ที่จะทำการทดสอบ โดยในรูปที่ 4.1 จะเป็นรูปเลนน่าที่เป็นรูปขาวดำ และรูปที่ 4.2 จะเป็นรูปสีเทา ส่วนในตารางที่ 4.2 จะแสดงถึงรายละเอียดของรูปภาพเลนน่าที่มีขนาดต่างๆ กัน

รูปที่ 4.1 รูปเลนน่าที่เป็นรูปขาวดำ

รูปที่ 4.2 รูปเลนน่าที่เป็นรูปสีเทา

ตารางที่ 4.2 แสดงขนาดของรูปเลนน่าที่มีขนาดและลักษณะต่างกัน

ประเภทของรูป / ขนาดของรูป	รูปขาวดำ	รูปสีเทา
$3^3 \times 3^3$	170 ไบต์	1834 ไบต์
$2^5 \times 2^5$	190 ไบต์	2102 ไบต์
$2^6 \times 2^6$	574 ไบต์	5174 ไบต์
$3^4 \times 3^4$	1034 ไบต์	7882 ไบต์

4.2 การเข้ารหัสและถอดรหัสข้อมูล

ในการเข้ารหัสจะใช้วิธีตามอัลกอริธึมที่ได้เขียนไว้ในบทที่ 3 โดยแบ่งเป็นโปรแกรมสำหรับการเข้ารหัสและถอดรหัสรูปภาพที่มีขนาด 2^mx2^m และ 3^nx3^n, เมื่อ m,n =0, 1, 2,...

4.3 เครื่องมือที่ใช้ในการวิจัย

เครื่องคอมพิวเตอร์ที่ใช้ในการทดสอบมีคุณสมบัติดังนี้

หน่วยประมวลผลกลาง (CPU)	:	AMD Duron 700 MHz
หน่วยความจำหลัก (RAM)	:	128 MB SD-Ram
หน่วยความจำสำรอง (HARD DISK)	:	20 GB
ระบบปฏิบัติการ (OS)	:	Windows XP Professional
โปรแกรมที่ใช้ในการพัฒนา	:	Visual Studio C#.NET
โปรแกรมที่ใช้วิเคราะห์ข้อมูลทางสถิติ	:	Microsoft Excel Version XP

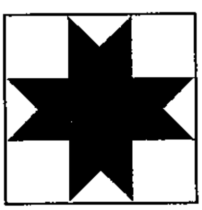
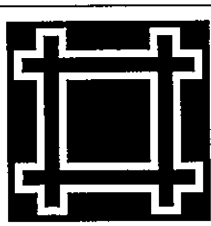
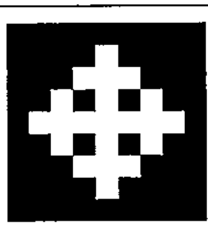
4.4 การทดสอบ

ในการทดสอบกับรูปภาพทั้ง 2 กลุ่ม ได้แก่ กลุ่มที่ 1 ได้แก่รูปภาพที่ประกอบด้วยรูปทรงทางเรขาคณิต และกลุ่มที่ 2 ซึ่งเป็นรูปของเลนน่า ซึ่งจะแบ่งเป็นรูปที่มีลักษณะเป็นแค่สีขาวดำเท่านั้น กับรูปที่มีความเข้มแสงอยู่ระหว่างสีขาวกับสีดำ (สีขาวจะเข้มขึ้นเรื่อยๆ จนกระทั่งเป็นสีดำ) โดยรายละเอียดสำหรับผลการทดสอบ จะกล่าวถึงในลำดับต่อๆ ไป

4.4.1 การบีบอัดข้อมูล

ในการบีบอัดข้อมูลจะเปรียบเทียบปริมาณของข้อมูลก่อนทำการบีบอัดและหลังทำการบีบอัดเพื่อหาปริมาณข้อมูลที่ลดลง ผลการทดสอบของกลุ่มที่ 1 จะเป็นดังตารางที่ 4.3

ตารางที่ 4.3 แสดงผลข้อมูลของไฟล์ที่ทำการบีบอัดแล้ว

รูป.	**ขนาดของรูป**	
	$3^5 \times 3^5$	$2^8 \times 2^8$
	Ribbon.WFA Compress Size 662 ไบต์ ขนาดรูปภาพ 3.46 กิโลไบต์ เวลาในการถอดรหัส 1.6093750 วินาที	Ribbon.WFA Compress Size 910 ไบต์ ขนาดรูปภาพ 3.38 กิโลไบต์ เวลาในการถอดรหัส 4.6093750 วินาที
	Sharp.WFA Compress Size 1.12 กิโลไบต์ ขนาดรูปภาพ 3.64 กิโลไบต์ เวลาในการถอดรหัส1.6875000 วินาที	Sharp.WFA Compress Size 1.29 กิโลไบต์ ขนาดรูปภาพ 3.87 กิโลไบต์ เวลาในการถอดรหัส 16.4062500 วินาที
	Snow.WFA Compress Size 433 ไบต์ ขนาดรูปภาพ 2.76 กิโลไบต์ เวลาในการถอดรหัส 1.0625000 วินาที	Snow.WFA Compress Size 908 ไบต์ ขนาดรูปภาพ 3.49 กิโลไบต์ เวลาในการถอดรหัส 7.3125000 วินาที
	Triangle.WFA Compress Size 5.02 กิโลไบต์ Original ขนาดรูปภาพ 8.00 กิโลไบต์ เวลาในการถอดรหัส 19.7656250 วินาที	Triangle.WFA Compress Size 4.00 กิโลไบต์ ขนาดรูปภาพ 8.00 กิโลไบต์ เวลาในการถอดรหัส 1.5312500 วินาที

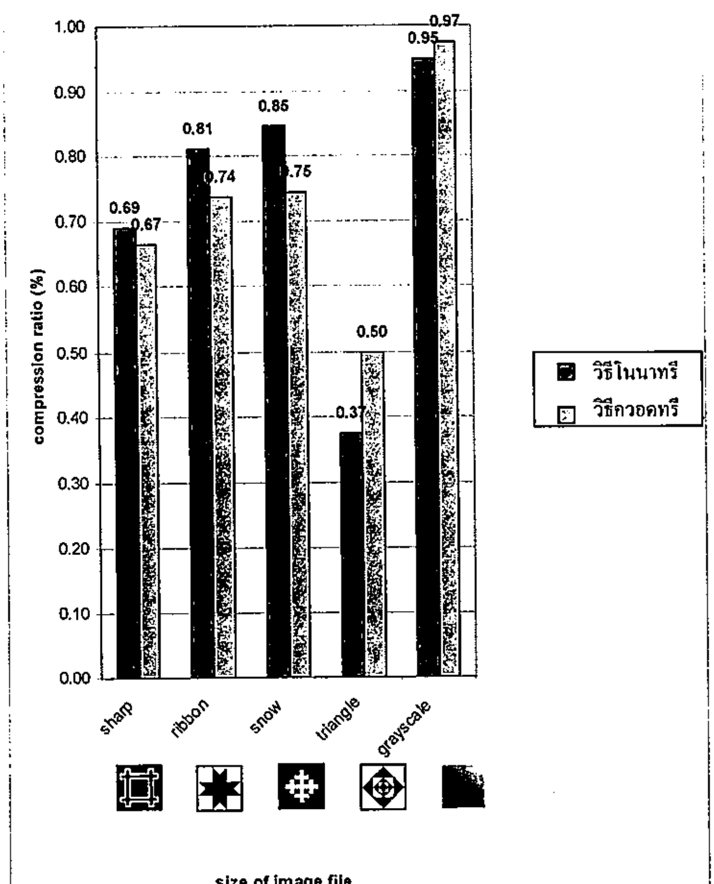
ตารางที่ 4.3 (ต่อ)

	Grayscale.WFA Compress Size 444 ไบต์ ขนาดรูปภาพ 8.4 กิโลไบต์ เวลาในการถอดรหัส 0.8906250 วินาที	Grayscale.WFA Compress Size 139 ไบต์ ขนาดรูปภาพ 5.36 กิโลไบต์ เวลาในการถอดรหัส 0.6875000 วินาที

โดยเมื่อนำข้อมูลแต่ละรูปมาทำเป็นกราฟเปรียบเทียบกันแล้วจะได้ดังกราฟรูปที่ 4.3 เมื่อ แกนนอน คือ รูปภาพที่ทำการบีบอัด โดยแถบที่มีสีเข้มจะเป็นการบีบอัดแบบการแบ่งแบบโนนาทรี และแถบที่มีสีอ่อนกว่าเป็นการแบ่งแบบควอดทรี โดยเรียงจากขนาดของไฟล์เล็กไปยังไฟล์ใหญ่

แกนตั้ง คือ ปริมาณข้อมูลที่ลดลงหลังผ่านการบีบอัดข้อมูล

$$\text{ปริมาณข้อมูลที่ลดลง} = \frac{\text{ปริมาณข้อมูลต้นฉบับ - ปริมาณข้อมูลหลังทำการบีบอัด}}{\text{ปริมาณข้อมูลต้นฉบับ}} \quad (4.1)$$



รูปที่4.3 กราฟแสดงอัตราการบีบอัด ของรูปต่างๆ จากกลุ่มที่ 1

ในส่วนของการทดลองในกลุ่มที่ 2 ซึ่งเป็นรูปของเลนน่า ผลการทดลองที่ได้จะพบว่าขนาดของไฟล์มีขนาดใหญ่ขึ้นตามตารางที่ 4.4

ตารางที่ 4.4 ผลของการบีบอัดรูปเลนน่า

ประเภทของรูป	รูปขาวดำ		
ขนาดของข้อมูล / ขนาดของรูป	ก่อนการบีบอัด	หลังการบีบอัด	อัตราการบีบอัด
$3^3 \times 3^3$	170 ไบต์	5007 ไบต์	-283453
$2^5 \times 2^5$	190 ไบต์	22640ไบต์	-118.158
$2^6 \times 2^6$	574 ไบต์	26553 ไบต์	-23.889
$3^4 \times 3^4$	1034 ไบต์	27323 ไบต์	-25.425

ก. ผลการบีบอัดรูปเลนน่าที่เป็นรูปขาวดำ

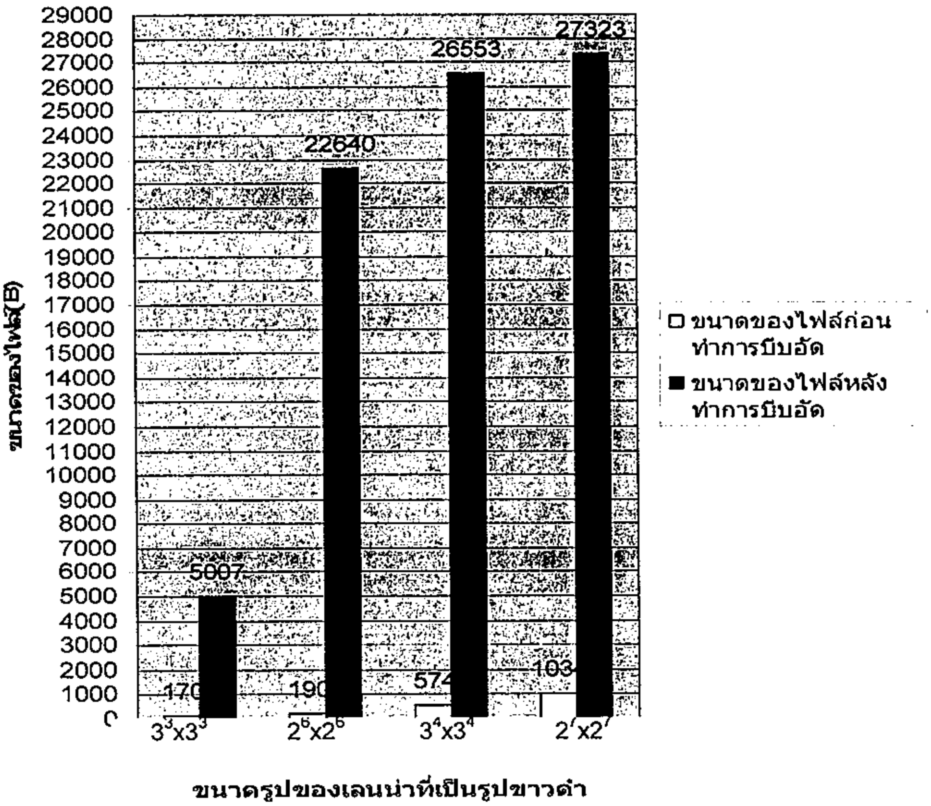
ประเภทของรูป	รูปสีเทา		
ขนาดของข้อมูล / ขนาดของรูป	ก่อนการบีบอัด	หลังการบีบอัด	อัตราการบีบอัด
$3^3 \times 3^3$	1834 ไบต์	11002 ไบต์	-4.999
$2^5 \times 2^5$	2102 ไบต์	18691 ไบต์	-7.892
$2^6 \times 2^6$	5174 ไบต์	26553 ไบต์	-4.132
$3^4 \times 3^4$	7882 ไบต์	124674 ไบต์	-14.818

ข. ผลการบีบอัดรูปเลนน่าที่เป็นรูปสีเทา

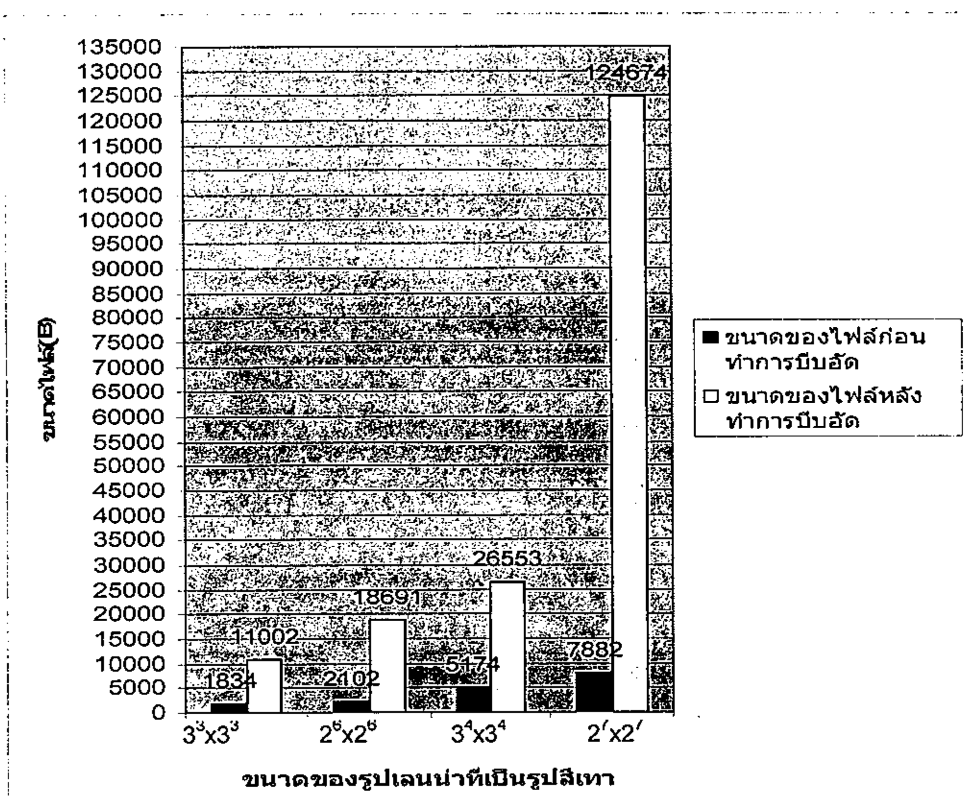
โดยจะสังเกตเห็นว่าขนาดของไฟล์ที่ทำการบีบอัดเพิ่มมากขึ้น และอัตราการบีบอัดที่คำนวณได้มีค่าติดลบ ทำให้ไม่สามารถแสดงเป็นกราฟอัตราการบีบอัดออกมาได้ ดังนั้นจึงแสดงได้เพียงกราฟเปรียบเทียบขนาดของไฟล์รูปภาพต้นฉบับและไฟล์บีบอัด ดังกราฟรูปที่ 4.4 สำหรับรูปเลนน่าที่เป็นรูปขาวดำ และ กราฟรูปที่ 4.5 สำหรับรูปเลนน่าที่เป็นรูปสีเทา

เมื่อ แกนนอน คือ ขนาดของรูปเลนน่าที่เป็นรูปขาวดำที่ทำการบีบอัด

แกนตั้ง คือ ปริมาณข้อมูล หรือ ขนาดของไฟล์



รูปที่ 4.4 กราฟแสดงขนาดของอัตราขนาดของไฟล์ที่เพิ่มมากขึ้นของรูปเลนน่าขาวดำ



รูปที่ 4.5 กราฟแสดงขนาดของไฟล์ที่เพิ่มมากขึ้นของรูปเลนน่าที่เป็นรูปสีเทา

4.4.2 การวิเคราะห์ข้อมูล

รูปที่ 4.3 เป็นกราฟแสดงขนาดของไฟล์ที่ลดลงของรูปภาพในกลุ่มที่ 1 ซึ่งแต่ละรูปจะประกอบด้วยรูปทรงทางเรขาคณิต โดยจะสังเกตเห็นว่า ในรูปเดียวกัน ที่ทำการบีบอัดด้วยออโตมาต้าแบบถ่วงน้ำหนัก ด้วยวิธีการแบ่งแบบโนนาทรี และแบบควอดทรี จะมีอัตราการบีบอัดที่ไม่เท่ากัน ซึ่งการที่จะสามารถบีบอัดรูปได้มากหรือน้อยนั้นขึ้นอยู่กับการแบ่งรูปภาพ ว่าจำนวนรูปย่อยที่ทำการแบ่งนั้นมีความเหมือนกับรูปย่อยอื่นๆ มากน้อยเพียงไร ดังนั้น ถ้าทำการแบ่งรูปแล้วปรากฏว่า จำนวนรูปย่อยที่ได้มานั้นมีลักษณะเหมือนกันเป็นจำนวนมาก จะทำให้รูปต้นฉบับที่ถูกทำการแบ่งนั้นมีอัตราการบีบอัดของรูปที่สูง แต่ถ้าจำนวนรูปย่อยที่แบ่งนั้น มีลักษณะที่เหมือนกันน้อย จะทำให้อัตราการบีบอัดนั้นน้อยลงไปด้วย

ดังนั้น จากกราฟในรูปที่ 4.3 จะสังเกตเห็นได้ว่า ในรูปของ sharp , ribbon และ snow นั้น เมื่อทำการแบ่งด้วยวิธีโนนาทรีจะได้จำนวนรูปย่อยที่มีลักษณะที่เหมือนกันมากกว่าการแบ่งแบบควอดทรี ในทางกลับกันรูปของ triangle และ grayscale เมื่อทำการแบ่งทั้งสองแบบแล้วจะพบว่าเหมาะสำหรับการแบ่งแบบควอดทรีมากกว่า

สำหรับรูปภาพของเลนน่าจะพบว่ามีขนาดที่เพิ่มมากขึ้นหลังที่ทำการบีบอัด ดังกราฟในรูปที่ 4.4 และ 4.5 เพราะว่า ในงานวิจัยนี้จะเป็นการบีบอัดแบบไม่สูญเสียข้อมูล (Lossless compression) ซึ่งในความเป็นจริงแล้ว โอกาสที่รูปย่อยแต่ละรูปจะมีความเหมือนกันนั้นน้อยมากหรือไม่ก็จะมีความเหมือนในลักษณะอื่นๆ ที่จะต้องพิจารณาเพิ่ม เช่น อาจจะมีขนาดของรูปที่ต่างกัน หรือ ต้องทำการหมุน (Rotate) การพลิกภาพ (Flip) อื่นๆ ทำให้รูปย่อยของเลนน่าแต่ละรูปเมื่อคำนวณหาค่าความสัมพันธ์กันออกมาแล้วจะไม่สามารถหาค่าความสัมพันธ์ได้ทำให้ต้องสร้างโหนดใหม่ทุกๆ ครั้ง ที่มีการคำนวณ หรือการแบ่งรูป จึงทำให้เมื่อจัดเก็บเป็นไฟล์บีบอัดรูปเลนน่านั้นนอกจากจะต้องเก็บค่าความเข้มแสงของแต่ละรูปย่อย ซึ่งถูกแบ่งจนเกือบเท่ากับ 1 พิกเซลแล้วยังต้องรวมกับโหนดที่อยู่ในระดับต้นๆ ของกราฟควอดทรี และค่าฉลากของแต่ละเส้นทางเดินของออโตมาต้าแบบถ่วงน้ำหนักอีกด้วย

รูปที่ 4.4 กราฟแสดงขนาดของไฟล์ที่เพิ่มมากขึ้นของรูปเลนน่าที่เป็นรูปขาวดำ ในกรณีที่เป็นรูปเล็กๆ เช่น 3^3x3^3 ที่มีอัตราการบีบอัดไฟล์เพิ่มมากขึ้นน้อยที่สุดก็เนื่องจากมีจำนวนพิกเซลที่น้อย ในขณะที่อีก 3 รูปที่เหลือ ถ้ารูปมีขนาดใหญ่ขึ้นบริเวณรูปย่อยที่มีแค่สีขาวหรือสีดำเพียงอย่างเดียวก็จะมีโอกาสมากขึ้น จึงทำให้อัตราการบีบอัดขนาดของไฟล์ที่ทำการบีบอัดมีขนาดเล็กลงเรื่อยๆ ดังตารางที่ 4.4 ก. และสำหรับรูปที่ 4.5 กราฟแสดงขนาดของขนาดของไฟล์ที่เพิ่มมากขึ้นของรูปเลนน่าที่เป็นรูปสีเทา ก็จะเป็นเช่นเดียวกับรูปขาวดำ คือเมื่อรูปมีขนาดใหญ่ขึ้นโอกาสที่รูปที่ทำการแบ่งจะมีลักษณะเหมือนกันก็จะมากขึ้นตามไปด้วย ซึ่ง

สำหรับรูปภาพอื่นๆ ที่มีขนาดเป็น 3^nx3^n, n = 0, 1, 2, ... ที่อยู่ในลักษณะของรูปภาพกลุ่มที่ 2 คือเป็นภาพถ่ายคน หรือธรรมชาติต่างๆ ในกรณีที่ใช้การคำนวณแบบงานวิจัยนี้ก็จะได้ไฟล์ที่มี

ขนาดใหญ่เพิ่มขึ้นเช่นเดียวกัน แต่จะเพิ่มมากน้อยแค่ไหนก็ขึ้นอยู่กับความซับซ้อนของของภาพว่าจะมีรายละเอียดมากแค่ไหน ถ้ามีความซับซ้อนมาก โอกาสที่รูปย่อยที่ทำการแบ่งแล้วจะมีความสัมพันธ์กับรูปอื่นก็จะน้อย แต่ถ้ามีความซับซ้อนน้อย โอกาสที่รูปย่อยจะมีความสัมพันธ์กับรูปอื่นก็จะมีมาก

สรุปว่า การบีบอัดแบบออโตมาต้าถ่วงน้ำหนักในงานวิจัยนี้ จะเป็นการบีบอัดแบบไม่สูญเสียข้อมูล จึงทำให้เหมาะสำหรับการบีบอัดรูปภาพที่ประกอบไปด้วยรูปทางเรขาคณิต เนื่องจากเมื่อแบ่งรูปแล้ว จะได้รูปที่มีลักษณะเหมือนกันมาก แต่ถ้านำไปใช้ใน การบีบอัดรูปภาพธรรมชาติ จะได้ขนาดของไฟล์บีบอัดเพิ่มมากขึ้น เพราะเมื่อทำการแบ่งเป็นรูปย่อยแล้ว จำนวนรูปย่อยที่จะมีลักษณะเหมือนกันนั้นน้อยมาก

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

จากงานวิจัยเรื่องการเข้ารหัสและถอดรหัสรูปภาพดิจิตอลด้วยวิธีไฟไนต์ออโตมาตาแบบถ่วงน้ำหนักด้วยวิธีการแบ่งแบบโนนาทรี โดยมีวัตถุประสงค์เพื่อที่จะศึกษาทดสอบการแบ่งรูปภาพออกเป็นเก้าส่วน ซึ่งสำหรับงานวิจัยที่เกี่ยวข้องกับรูปภาพดิจิตอลส่วนมากจะยึดการแบ่งรูปภาพออกเป็นแบบควอดทรีเป็นพื้นฐานแล้วถึงได้พิจารณาในส่วนขยายอื่นๆ เช่น การหมุน เป็นต้น แต่ในความเป็นจริง รูปภาพดิจิตอลทั่วๆไปนั้นไม่จำเป็นต้องแบ่งออกได้เป็นแบบควอดทรีทั้งหมด อาจจะมีบางรูปที่เหมาะสำหรับการแบ่งเป็นโนนาทรี ซึ่งจากผลการทดลองจะสามารถสรุปผลได้ดังนี้

1) ในการบีบอัดรูปที่ประกอบด้วยรูปทรงทางเรขาคณิตด้วยวิธีออโตมาต้าแบบถ่วงน้ำหนักจะเป็นการบีบอัดรูปภาพแบบไม่มีการสูญเสียข้อมูล (Lossless compression) โดยที่ขนาดของไฟล์จะมีขนาดเล็กลง เนื่องจากโอกาสที่แต่ละรูปย่อยจะมีลักษณะเหมือนกันมีโอกาสมาก

2) ในการบีบอัดรูปภาพในธรรมชาติทั่วไป เช่นในงานวิจัยนี้คือรูปเลนน่า ด้วยวิธีออโตมาต้าแบบถ่วงน้ำหนักนี้จะเป็นการบีบอัดรูปภาพแบบ ที่ไม่มีการสูญเสียข้อมูล เนื่องจากการคำนวณหาค่าความสัมพันธ์ของแต่ละรูปย่อย ใช้การคำนวณซึ่งจะคำนวณค่าความสัมพันธ์ของแต่ละรูปย่อยจริงๆ ไม่ใช่ค่าประมาณ (Aproximation value) ซึ่งทำให้การบีบอัดรูปธรรมชาติทั่วๆ ไปขนาดไม่ได้มีขนาดที่เล็กลง เพราะการที่แต่ละรูปย่อยที่ทำการแบ่งนั้นจะมีความสัมพันธ์กันได้ เป็นเรื่องที่ค่อนข้างยาก แต่ถ้าหากใช้การประมาณค่าช่วยในการคำนวณหาค่าความสัมพันธ์ ซึ่งจะทำให้ค่าความเข้มแสงบางจุดไม่ใช่ค่าที่แท้จริง ก็จะทำให้เป็นการบีบอัดแบบสูญเสียข้อมูล (Lossy compression) โดยที่ข้อมูลที่สูญเสียไปจะไม่สามารถกู้กลับมาได้ แต่ก็จะสามารถทำให้มีอัตราการบีบอัดได้มากขึ้น ซึ่งวิธีที่ใช้ในการคำนวณจะส่งผลต่อการใช้เวลาในการบีบอัดด้วย โดยที่ถ้าอัตราการบีบอัดมากก็จะใช้เวลาในการบีบอัดมากตามไปด้วย

3) ในการบีบอัดรูปภาพด้วยวิธีออโตมาต้าแบบถ่วงน้ำหนักนั้นในการที่อัตราการบีบอัดข้อมูลจะมากหรือน้อยขึ้นอยู่กับว่ารูปที่ทำการบีบอัดนั้นจะมีเป็นรูปที่มีรูปย่อยที่เหมือนกัน หรือมีรูปที่ซ้ำๆ กันมากแค่ไหน ซึ่งถ้ายิ่งมีรูปที่ซ้ำกันมากๆ ก็จะทำให้อัตราการบีบอัดเพิ่มมากขึ้นด้วย เนื่องจากจะสามารถใช้รูปย่อยเพียงรูปเดียวแทนรูปย่อยอื่นๆ ได้

4) ผลพลอยได้จากการทำการบีบอัดก็คือการเข้ารหัสข้อมูล ทำให้สามารถป้องกันไม่ให้ข้อมูลรั่วไหลได้ ถ้าหากไม่ทราบวิธีที่จะทำการถอดรหัสออกมาให้ได้ค่าความเข้มแสงของแต่ละพิกเซล แล้วจึงนำค่าความเข้มแสงนั้นมากำหนดให้ถูกตำแหน่งของรูปภาพ

5.2 ข้อเสนอแนะ

1) ในการคำนวณหาความสัมพันธ์ของแต่ละรูปย่อย หรือการหาค่าสัมประสิทธิ์ของผลรวมเชิงเส้น อาจจะใช้การประมาณค่าเพื่อที่จะได้อัตราการบีบอัดที่เพิ่มมากขึ้น

2) หากจะประยุกต์ไปใช้เป็นโปรแกรมเพื่อลดขนาดของรูปภาพ ควรจะมีอัลกอริทึมที่จะใช้ตัดสินใจว่าควรจะทำการแบ่งรูปแบบใด เพราะการเตรียมรูปภาพโดยการแบ่งรูปแบบสี่ส่วนกับเก้าส่วน อาจจะมีอัตราการบีบอัดที่ต่างกันมาก ซึ่งถ้าหากไม่ตัดสินใจให้ดีๆ แล้วจะทำให้การทำการบีบอัดไม่ได้ผลดีเท่าที่ควร

3) ในการเตรียมรูปภาพด้วยการแบ่งรูปออกเป็นรูปย่อยๆ อาจจะใช้ส่วนขยายมาพิจารณาเพิ่มเติม ซึ่งได้แก่ การหมุนรูป ซึ่งจะทำให้อัตราการบีบอัดเพิ่มมากขึ้น แต่อาจจะต้องใช้เวลาในการคำนวณหาความสัมพันธ์ของรูปมากขึ้น

5.3 แนวทางในการวิจัย

1) การบีบอัดด้วยวิธีออโตมาต้าแบบถ่วงน้ำหนักจะมีการเตรียมรูปโดยการแบ่งรูปออกเป็นรูปย่อยๆ ดังนั้น จึงแสดงว่าในออโตมาต้าแบบถ่วงน้ำหนักที่สร้างขึ้นนั้นย่อยมีรูปย่อยแทรกอยู่ ดังนั้นจึงควรศึกษาการค้นหารูปย่อยที่อยู่ในออโตมาต้าแบบถ่วงน้ำหนัก ซึ่งอาจจะใช้ในการค้นหาจุดสำคัญของแต่ละรูปได้ เช่น การเปรียบเทียบลายนิ้วมือ การเปรียบเทียบรูปถ่ายของบุคคลว่าเป็นบุคคลเดียวกัน ฯ

2) ประยุกต์การบีบอัดด้วยวิธีออโตมาต้าแบบถ่วงน้ำหนักกับภาพถ่ายวิดีโอ หรือคลื่นเสียง เพราะภาพถ่ายวิดีโอ จะมีลักษณะที่ซ้ำกันมากกว่ารูปถ่าย และ ในกรณีเดียวกันคลื่นเสียงก็จะมีความซ้ำซ้อนในเสียงของตัวเองที่มากกว่าด้วยเช่นกัน ดังนั้นจึงน่าจะมีอัตราการบีบอัดมากกว่า

เอกสารอ้างอิง

[1] K. Culik II and J. Kari, **Image Compression Using Weighted Finite Automata**. Computers and Graphics, vol. 17, no. 3, pp. 305-313, May/June 1993.

[2] Stephans Mallat and Zhifeng Zhang, **mathcing Pursuit With Time-Frequency dictionaries**, 1993, IEEE

[3] F.Katritzke, W. Merzenich, M. Thomas, **Enhancements of partitioning techniques for image compression using weighted fintie automata**, 2003 Elsevier

[4] Ullrich Hafner, Lehrstuhl fur Informatik II, Universitat Wurzburg, **Refining Image Compression with Weighted Finite Automata**, 1996 IEEE

[5] Karel Culik II, **IMAGE-DATA COMPRESSION USING EDGE-OPTIMIZENG ALGORITHM FOR WFA INFERENCE**, 1994 Elsevier

[6] K. Culik II and J. Kari, **Image compression using weighted finite automata**, 1993 computer and graphics, 17, 305-313

[7] Frank Katritzke, **Refinements of Data Compression Using Weighted Finite Automata**, Dissertation

[8] Zhuhan Jiang, Bruce Litow, Olivier de Vel, **An Inference Implementation Based on Extended Weighted Finite Automata**, 2001 IEEE

[9] Kamala Krithivasan, **Weighted Finite Automata and Representation of Image**, lecture note.

ประวัติผู้เขียน

ชื่อ - สกุล	นาย ธมรัตน์ เลขะวิจิตเลิศ
วัน เดือน ปีเกิด	8 ธันวาคม 2522
ที่อยู่	9/112 ซอย พหลโยธิน 21 เขตจตุจักร กรุงเทพฯ 10900
ประวัติการศึกษา	2546 จบการศึกษาวิทยาศาสตร์บัณฑิต สาขา คณิตศาสตร์ประยุกต์ สถาบันเทคโนโลยีพระจอมเกล้า พระนครเหนือ