

การประมวลผลภาพในระบบดิจิทัล  
( Image Processing )

ประสงค์      จารุชเนศ  
วิโรจน์      สถาปนิกกุล  
Prasong      Charudhanes  
Wirote      Stapanikkul



อาจารย์ที่ปรึกษา  
ดร. บุญวัฒน์      อัฒชู  
Advisor  
Dr. Boonwat      Attachoo

ปริญญาโทสำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2531



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


023238

ปริญญาโท ประจำปีการศึกษา.....2531.....  
ภาควิชา.....วิศวกรรมคอมพิวเตอร์.....

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
เรื่อง.....การประมวลผลภาพในระบบดิจิทัล.....  
ผู้จัดทำ.....

1. นายประสงค์ จารุธเนศ
2. นายวิโรจน์ สถาปนิกกุล
3. ....

๕

  
.....อาจารย์ที่ปรึกษา  
(.....คร. นณวัฒน์ อิศข.....)  
.....อาจารย์ที่ปรึกษา  
(.....)  
.....อาจารย์ที่ปรึกษา  
(.....)

การประมวลผลภาพในระบบดิจิทัล

ประสงค์            จารุเดช  
วิโรจน์            สถาบันกุล  
ดร. บุญวัฒน์    อัคร            อาจารย์ที่ปรึกษา  
ปีการศึกษา            2531

บทคัดย่อ

ในวิทยานิพนธ์ฉบับนี้ เรียบเรียงขึ้นจากผลงานที่ได้ทำการศึกษาและค้นคว้าทดลองเกี่ยวกับขบวนการประมวลผลภาพในระบบดิจิทัล เน้นด้านการลดข้อมูลภาพ โดยกล่าวถึงระบบทั่วไปของการประมวลผลภาพทางดิจิทัล คณิตศาสตร์พื้นฐานที่สามารถนำมาประยุกต์ใช้กับการลดข้อมูลภาพ ได้รวบรวมอัลกอริทึม (Algorithm) ต่าง ๆ ที่ใช้ลดข้อมูลภาพ โดยใช้ได้ทั้งสำหรับภาพไบนารี (ภาพมี 2 ระดับความเข้มคือ ขาวกับดำ) ภาพเกรย์เลเวล (gray level คือภาพขาวดำที่มีหลายระดับความเข้ม) และภาพสี ทำการวิเคราะห์ข้อได้เปรียบ ข้อเสียเปรียบ และเปรียบเทียบอัลกอริทึมต่าง ๆ ให้ผู้ใช้ได้เลือกอัลกอริทึมที่เหมาะสม สำหรับภาพแต่ละชนิด เพื่อประโยชน์ในการนำไปใช้งาน อีกทั้งกล่าวถึงวิธีการนำอัลกอริทึมมาเข้ากับไมโครคอมพิวเตอร์ได้ทั้ง IBM PC และ IBM PS/II สำหรับภาพขาวดำ ภาพเกรย์เลเวล และภาพสี

## IMAGE PROCESSING

Prasong Charudhanes  
Wrote Stapanikkul  
Dr. Boonwat Attachoo Advisor  
1988

### Abstract

This thesis proposes application of digital image processing to compress image data by using algorithms. It discusses the general system of image processing, the basic mathematics and algorithms for image compression. Algorithms are used to compress image data for binary image, the gray level image and the color image. The advantage and disadvantage of algorithms are analyzed for user to select the suitable algorithm for a kind of images. All of algorithms in this thesis were developed for IBM PC/XT/AT and IBM PS/II.

สารบัญ

หน้า

บทคัดย่อ

สารบัญรูป

สารบัญตาราง

บทที่ 1 บทนำ

1.1 ประวัติและการพัฒนา

1.2 เนื้อหาของวิทยานิพนธ์

บทที่ 2 ขบวนการสร้างภาพด้วยคอมพิวเตอร์

และคณิตศาสตร์พื้นฐานที่ใช้ในการประมวลผลภาพดิจิทัล

2.1 ความหมายและนิยามของภาพคอมพิวเตอร์

2.2 การแทนภาพด้วยข้อมูลแบบดิจิทัล

2.3 ระบบการประมวลผลภาพทางดิจิทัล

2.4 การสุ่มแบบสุ่มสม่ำเสมอและควอนไทเซชัน

2.5 เทคนิคต่าง ๆ สำหรับการประมวลผลภาพทางดิจิทัล

2.6 การทรานส์ฟอร์มภาพ

2.7 ดิสกรีตฟูเรียร์ทรานส์ฟอร์มและดิสกรีตฮาดามาร์ดทรานส์ฟอร์ม

2.8 ฟาสต์ฟูเรียร์ทรานส์ฟอร์มและฟาสต์ฮาดามาร์ดทรานส์ฟอร์ม

บทที่ 3 การลดข้อมูลภาพ

3.1 - การลดข้อมูลภาพด้วยวิธีรันเลงก์โค้ด

3.2 การลดข้อมูลภาพด้วยวิธีฮัฟแมนโค้ด

3.3 การลดข้อมูลภาพด้วยวิธีฮัฟแมนบีฟโค้ด

3.4 การลดข้อมูลภาพด้วยวิธีบีล็คกรุนด์

3.5 การลดข้อมูลภาพด้วยวิธีแอสแอล

3.6 การลดข้อมูลภาพด้วยวิธีฮาดามาร์ด

3.7 การลดข้อมูลภาพด้วยวิธีฟูเรียร์ทรานส์ฟอร์ม

3.8 การลดข้อมูลภาพด้วยวิธีโคซายันทรานส์ฟอร์ม

1

1

4

6

6

7

8

9

11

16

35

36

49

50

57

62

65

71

77

78

81

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่าการนำเอกสารนี้ไปใช้ในรูปแบบใดก็ตาม ผู้ใช้ต้องรับผิดชอบต่อเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
3.9 การลดข้อมูลภาพด้วยวิธีการกำหนดตำแหน่งบิต	86
3.10 การลดข้อมูลภาพด้วยวิธีทำนาย	89
3.11 การลดข้อมูลภาพด้วยวิธีไฮเทรเวดเฟิงก์ที่เฮลสเต็ม	99
บทที่ 4 การนำเทคนิคการลดข้อมูลภาพประยุกต์ใช้กับไมโครคอมพิวเตอร์ และผลการทดลอง	104
บทที่ 5 สรุปและวิจารณ์	125
กิตติกรรมประกาศ	127
หนังสืออ้างอิง	128



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1177

### สารบัญรูป

	หน้า
รูป 1.1 ภาพพิมพ์ก่อนหน้าปี 1921 และภาพถ่ายจากเทปใหม่ 1921	2
รูป 1.2 ภาพตัวอย่างของเทคนิคต่าง ๆ ในการประมวลผลภาพในระบอดิจิตอล	3
รูป 2.1 กระจกและพิกัดที่ใช้ในระบบภาพ	6
รูป 2.2 แสดงภาพดิจิทัลขนาด 64 X 64 จุด <sup>2</sup>	8
รูป 2.3 ระบบประมวลผลภาพดิจิทัล	8
รูป 2.4 เปรียบเทียบภาพเมื่อลดความละเอียดของภาพลง	11
รูป 2.5 แสดงภาพการทำคอนทราสต์เอนฮานเมนต์	13
รูป 2.6 แสดงภาพการทำคอนทราสต์เอนฮานเมนต์	14
รูป 2.7 แสดงภาพการกรองสัญญาณภาพ	15
รูป 2.8 ฟังก์ชันง่าย ๆ และฟูเรียร์สเปกตรัมของมัน	18
รูป 2.9 ฟังก์ชัน 2 มิติ และฟูเรียร์สเปกตรัม	19
รูป 2.10 ฟังก์ชันภาพต่อเนื่องที่กลุ่มในช่องเท่า ๆ กัน	19
รูป 2.11 ภาพและสเปกตรัมของภาพ	21
รูป 2.12 ผลของการใช้ค่าล็อก (log) กับฟูเรียร์สเปกตรัม	22
รูป 2.13 ดาวเสาร์ และผลของ $ F(u,v) $	23
รูป 2.14 หาค่าฟูเรียร์ทรานส์ฟอร์ม 2 มิติ จากภาพทรานส์ฟอร์มมิติเดียว	24
รูป 2.15 สรุปลักษณะสมบัติคาบของฟูเรียร์ทรานส์ฟอร์ม	26
รูป 2.16 รูปตัวอย่างง่าย ๆ และฟูเรียร์ทรานส์ฟอร์มสเปกตรัม	27
รูป 2.17 คุณสมบัติการหมุนของฟูเรียร์ทรานส์ฟอร์ม	28
รูป 2.18 แสดงคุณสมบัติการลุ่ม	29
รูป 2.19 แสดงการลุ่มที่จำกัด	31
รูป 2.20 ฟังก์ชันการลุ่มใน 2 มิติ	32
รูป 2.21 การแทนตัวอย่างที่ลุ่มใน 2 มิติ ของโดเมนความถี่	33
รูป 2.22 แสดงแผนผังการทำงานของฟาสต์ฟูเรียร์ทรานส์ฟอร์มม่อนูญาติให้นำไปใช้ประโยชน์การคำนวณ	37
รูป 2.23 โฟลว์ไดอะแกรมของดิสกรีตฮาร์ดเลย์ทรานส์ฟอร์ม	40

รูป 2.24	แฟคเตอร์ของโคซายน์	42
รูป 2.25	โพลาร์ชาร์ตแสดงอัลกอริทึมของพาส์ฟูเรียร์ทรานสฟอร์ม	47
รูป 2.26	โพลาร์ชาร์ตแสดงอัลกอริทึมของพาส์ฮาร์ดเลย์ทรานสฟอร์ม	48
รูป 3.1.1	ลักษณะการเข้ารหัสรันเลงจ์	51
รูป 3.1.2	ตัวอย่างรันเลงจ์แม่บั้ง	52
รูป 3.1.3	แสดงรูปหลังถอดรหัสด้วยวิธีรันเลงจ์ ภาพ 256 ระดับ	53
รูป 3.1.4	แสดงรูปหลังถอดรหัสด้วยวิธีรันเลงจ์ ภาพ 2 ระดับ	54
รูป 3.1.6	ฮิสโตแกรมของภาพก่อน และหลังการทำเกรย์สเกลทรานสฟอร์มเมชั่น	56
รูป 3.2.1	การสร้างทรีสำหรับฮัฟฟ์แมนโคด	58
รูป 3.2.2	รูปทรีที่สร้างขึ้นใหม่ จากรูป 3.2.1	59
รูป 3.2.3	การใช้โคดสร้างทรีในคอมพิวเตอร์	60
รูป 3.3.1	แสดงภาพขนาด $256 \times 256$ จุด <sup>2</sup>	62
รูป 3.3.2	แสดงวิธีการสร้างทรีโคด	63
รูป 3.4.1	การตัดส่วนของภาพเป็นบล็อกและเปลี่ยนข้อมูล	65
รูป 3.4.2	ภาพต้นแบบขนาด $256 \times 256$ จุด <sup>2</sup>	67
รูป 3.4.3	ภาพที่สร้างขึ้นจากการถอดรหัสด้วยวิธีบล็อกทรานเคชั่น	68
รูป 3.4.5	แสดงภาพหลังการถอดรหัสบล็อกทรานเคชั่น	70
รูป 3.5.1	ภาพขนาด $256 \times 256$ จุด <sup>2</sup>	73
รูป 3.5.2	แบ่งภาพใหญ่เป็นภาพย่อย	74
รูป 3.5.3	ภาพย่อย $16 \times 16$ ภาพ เรียงลำดับ	75
รูป 3.5.4	ภาพที่ได้กลับคืนมา โดยการเก็บค่าสัมประสิทธิ์เคแอลทรานสฟอร์ม	75
รูป 3.5.5	ความแปรปรวนของสัมประสิทธิ์ $F(u, v)$	76
รูป 3.5.6	ภาพจากการควอนไทซ์ภาพที่ใช้สัมประสิทธิ์เคแอลทรานสฟอร์ม	76
รูป 3.6.1	ภาพจากการควอนไทซ์ภาพที่ใช้สัมประสิทธิ์ฟูเรียร์ทรานสฟอร์ม	78

เอกสารนี้เป็นเอกสารที่มอบให้ทางราชการใช้การ ซึ่งควรสงวนและไม่นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น ซึ่งผู้รับมอบให้ต้องดูแลรักษา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
รูป 3.8.1 ภาพตัวอย่างและค่าเฉลี่ยความคลาดเคลื่อนกำลังสอง	82
รูป 3.8.2 ดิสกรีตโคซายน์ทรานส์ฟอร์ม	83
รูป 3.8.3 ภาพจากดิสกรีตโคซายน์ทรานส์ฟอร์ม	84
รูป 3.9.1 a) ภาพที่ใช้ควอนไทเซอร์ 4 ระดับ	87
รูป 3.9.1 b) แสดงผลของบล็อกควอนไทเซชัน	88
รูป 3.10.1 การพิจารณาจุดของการลดข้อมูลภาพด้วยวิธีทำนาย	89
รูป 3.10.2 แสดงระบบที่กำเนิดข้อมูลแตกต่าง $e(m,n)$	92
รูป 3.10.3 ภาพตัวอย่างและภาพที่ได้จากการรบกวน	95
รูป 3.10.4 ค่าความคลาดเคลื่อนเฉลี่ยกำลังสอง สำหรับตัวทำนายแต่ละระดับ	96
รูป 3.10.5 ฮิสโตแกรมของสัญญาณแตกต่าง	96
รูป 3.10.6 ผลของภาพที่ใช้การลดข้อมูลภาพด้วยวิธีทำนาย	97
รูป 3.11.1 การทราเน็ฟฟอร์มภาพสีเหลี่ยม	99
รูป 3.11.3 การทำแรนดอมไฮเทอเรนซ์	102
รูป 3.11.4 การใช้ทฤษฎีคอลลาจ	103
รูป 4.1 ภาพต้นแบบ $256 \times 256$ จุด $256$ ระดับ	107
รูป 4.2 ภาพต้นแบบ $256 \times 256$ จุด $8$ ระดับ	108
รูป 4.3 ภาพต้นแบบ $256 \times 256$ จุด $256$ ระดับ	109
รูป 4.4 ภาพหลังจากการถอดรหัสด้วยบล็อกทรูเนชัน	110
รูป 4.5 ภาพหลังจากการถอดรหัสด้วยบล็อกทรูเนชัน	111
รูป 4.6 ภาพต้นแบบ ขนาด $256 \times 256$ จุด $256$ ระดับ	112
รูป 4.7 ภาพหลังจากการถอดรหัสด้วยบล็อกทรูเนชัน	113
รูป 4.8 ภาพต้นแบบ ขนาด $256 \times 256$ จุด $2$	114
รูป 4.9 ภาพต้นแบบ ขนาด $256 \times 256$ จุด $2$ ระดับ	115
รูป 4.10 ภาพต้นแบบ	118

## สารบัญตาราง

	หน้า
ตาราง 2.1 จำนวนไบท์ที่ใช้ในการเก็บภาพ	11
ตาราง 2.8.2.1 แสดงการคำนวณของชั้นต่าง ๆ ในสาร์ตเล็ทราเนลฟอร์ม	38
ตาราง 2.8.2.2 แสดงการคำนวณ เมื่อแทนค่าโคไซน์และไซน์ในตาราง 2.8.2.1	39
ตาราง 2.8.2.3 ตัวอย่างของการคำนวณฟาสท์สาร์ต เล็ทราเนลฟอร์ม	40
ตาราง 2.8.2.4 ตารางการเปรียบเทียบเวลาของฟาสท์ฟูเรียร์ทรานสฟอร์มและฟาสท์สาร์ต เล็ทราเนลฟอร์ม	43
ตาราง 2.8.2.5 เปรียบเทียบเวลาของฟาสท์ฟูเรียร์และฟาสท์สาร์ต เล็ทราเนลฟอร์ม	45
ตาราง 3.1.5 แสดงอัตราข้อมูลภาพเดียวกัน แต่มีความแตกต่างของระดับสีต่างกัน	55
ตาราง 3.3.1 เปรียบเทียบอัตราข้อมูลหลังจากเข้ารหัสภาพด้วยอัลฟ์แมนซิงก์โคัดและอัลฟ์แมน โคัด	64
ตาราง 3.4.4 ความแตกต่างของปริมาณการลดข้อมูลภาพ	69
ตาราง 3.10.1 ค่าโครีเลชัน	93
ตาราง 3.10.2 ค่าสัมประสิทธิ์ $a_1$ ที่ใช้สำหรับแต่ละตัวทำนาย	94
ตาราง 3.11.1 รหัส ไอเอฟเอส	101
ตาราง 4.1 เปรียบเทียบการลดข้อมูลภาพ 3 วิถี	117
ตาราง 4.1 รหัส ไอเอฟเอสของส่วนที่หนึ่ง	119
ตาราง 4.2 รหัส ไอเอฟเอสของส่วนที่สอง	119
ตาราง 4.3 การเปรียบเทียบประสิทธิภาพการลดข้อมูลภาพ	123
ตาราง 4.4 ภาพที่เหมาะสมกับการลดข้อมูลภาพแต่ละวิถี	124

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

(INTRODUCTION)

ข้อมูลต่างๆ ที่มีเขยิบรับรู้ประมาณ 75% เป็นข้อมูลที่มนุษย์ได้รับจากการมองเห็น มนุษย์จะนำข้อมูลที่รับรู้ไปตีความหมายเพื่อความเข้าใจ เช่นเดียวกับเมื่อคอมพิวเตอร์รับข้อมูลหรือใช้ข้อมูลที่เกี่ยวกับการมองเห็น ซึ่งเราเรียกว่า ขบวนการสร้างภาพและจดจำภาพของคอมพิวเตอร์ (Computer image processing and recognition)

การเชื่อมโยงความสัมพันธ์ระหว่างภาพที่คอมพิวเตอร์เข้าใจกับภาพที่มนุษย์เข้าใจทำได้โดยการให้คอมพิวเตอร์แปลภาพในลักษณะที่มนุษย์เข้าใจ ไปเป็นข้อมูลตัวเลขที่คอมพิวเตอร์เข้าใจ จากนั้นคอมพิวเตอร์ก็จะคำนวณและประเมินผลได้กลับมาเป็นภาพที่มนุษย์คุ้นเคย การกระทำดังกล่าวนี้ได้นั้นต้องการความรู้ทางคอมพิวเตอร์และคณิตศาสตร์ เพื่อหาความสัมพันธ์ระหว่างข้อมูลตัวเลขกับจุดต่างๆ ในภาพ ในบทนี้จะกล่าวถึงประวัติและการพัฒนาขบวนการสร้างภาพในระบบดิจิทัล โดยใช้คอมพิวเตอร์ ตั้งแต่เริ่มจนสามารถนำมาใช้ประโยชน์ได้อย่างมีประสิทธิภาพในปัจจุบัน

### 1.1 ประวัติและการพัฒนาด้านการประมวลผลภาพดิจิทัล

ความสนใจในเทคนิคการทำภาพในระบบดิจิทัล (Digital image) ได้เกิดขึ้นในงานส่งข่าวสารเป็นภาพข้อมูลตัวเลขทางเคเบิลใต้น้ำ ระหว่างลอนดอนกับนิวยอร์ก ซึ่งเป็นการส่งภาพแบบบาร์ทเลน เคเบิล (Bartlane cable) มาใช้ในเดือน ค.ศ. 1920 ซึ่งลดเวลาในการส่งภาพข้ามมหาสมุทรแอตแลนติก จากการส่งภาพธรรมดา 1 สัปดาห์ ลดลงมาเหลือไม่ถึง 3 ชั่วโมง ภาพต่าง ๆ จะนำมาเข้าโคิดเพื่อส่งทางเคเบิลและรับที่ปลายทางด้วยอุปกรณ์รับข้อมูลภาพ และพิมพ์โดยเทเลกราฟพริ้นเตอร์ (Telegraph printer) พิมพ์ภาพเหมือนออกมา ภาพที่ได้ในระยะเริ่มแรกมีปัญหาทางด้านคุณภาพของภาพ และวิธีการในการทำภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบบนำร์ทเลนในระยะเริ่มแรกสามารถเข้าโคัดได้ ในระดับความเข้ม 5 ระดับ และได้เพิ่มขึ้นเป็น 15 ระดับในปี ค.ศ. 1929



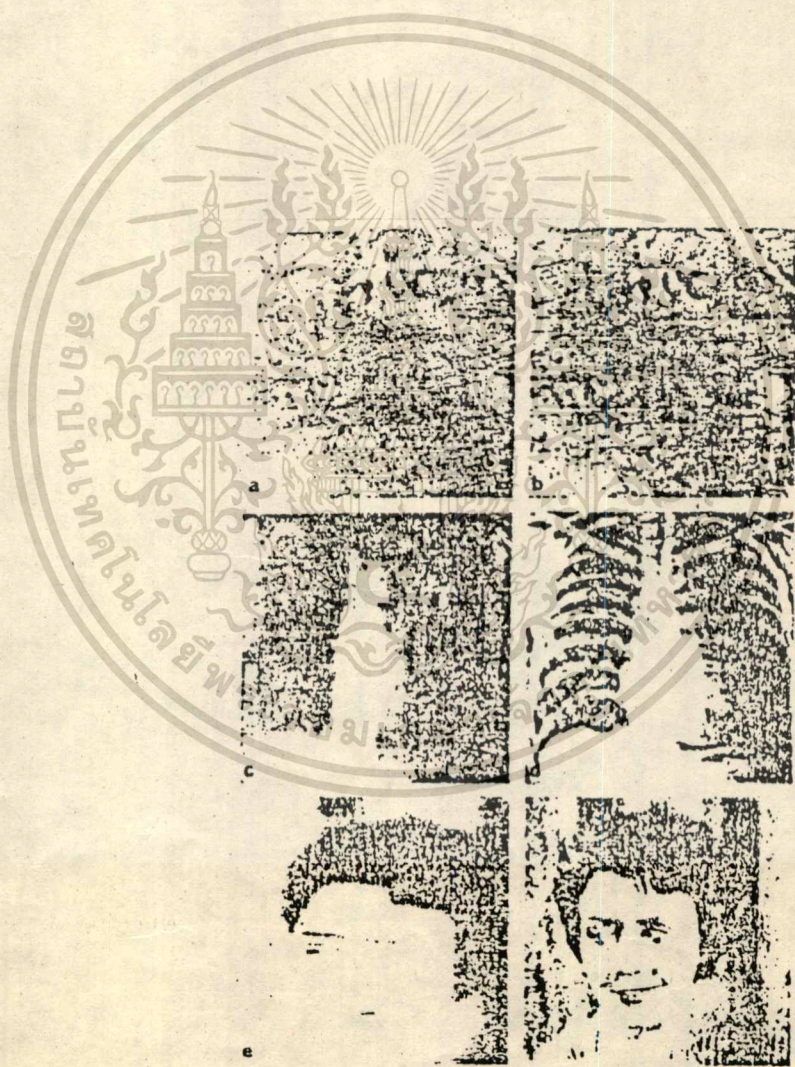
รูป 1.1 a) ภาพพิมพ์ก่อนหน้าปี 1921 b) ภาพถ่ายจากเทปในปี 1921

ในระหว่างนี้เองวิธีการและเทคโนโลยีต่าง ๆ ได้รับการปรับปรุงขึ้น โดยการเาระบบที่ใช้การเข้าโคัดกับลำแสงที่มอดดูเลข แม้ว่าหลังจากนั้นได้มีการพัฒนาและปรับปรุงวิธีการส่งภาพเรื่อยมากกว่า 35 ปี แต่เครื่องคอมพิวเตอร์และโปรแกรมที่ใช้ ยังมีขนาดใหญ่และประสิทธิภาพต่ำอยู่ ในปี ค.ศ. 1920 ห้องทดลองเจทไฟรูล์ทั้น (JET PROPULSION ที่ BASADONA CALIFORNIA) ได้ทำการปรับปรุงคุณภาพของภาพที่ส่งลงมาจากอวกาศ โดยใช้เทคโนโลยี ทางคอมพิวเตอร์ ได้มีการกำจัดสัญญาณรบกวน (Distortion) ที่มากับสัญญาณภาพของภาพดวงจันทร์บนกล้องโทรทัศน์ ซึ่งเทคนิคเหล่านี้เป็นหลักการเบื้องต้น ในการปรับปรุงภาพ และทำภาพกลับ (Enhancement and Restoration)

ตั้งแต่ ค.ศ. 1964 เป็นต้นมา เทคโนโลยีทางด้านขบวนการประมวลผลภาพ ได้ถูกนำมาประยุกต์ใช้งานอย่างแพร่หลาย ในหลาย ๆ สาขาวิชาก็ได้ใช้เทคนิคของการประมวลผลภาพในระบบดิจิทัล แก่ปัญหาต่าง ๆ ที่เกี่ยวข้องกับข้อมูลภาพ เพื่อใช้ในการวิเคราะห์และตีความหมายของภาพที่สำคัญอย่างเช่น ทางด้านทหารแพทย์ ไม่ขบวนการทางคอมพิวเตอร์ได้ช่วยปรับโลกนี้ อีลหรือเข้าโคัดระดับความเข้มของภาพให้อยู่ในระดับที่ง่ายต่อความเข้าใจ

หรือแปลความหมาย ส่วนเทคนิควิทยาศาสตร์ที่ใช้เทคโนโลยีเหล่านี้ ในการศึกษาถึงปัญหาประชากร จากภาพที่ถ่ายจากที่สูง หรือภาพถ่ายดาวเทียม

วิธีการของการนำภาพกลับมา และทำให้ภาพคมชัด ได้ถูกนำมาใช้ในการ แก้ไขปัญหาต่าง ๆ ของภาพ สามารถสร้างส่วนที่มองไม่เห็น ให้เห็นออกมาได้ ทำให้ภาพคม ชัดขึ้น เช่น ภาพทางโบราณคดี ซึ่งเป็นภาพมัว ๆ ทำให้ข้อมูลสำคัญ ๆ หายไป หรือถูกทำลาย ไป ก็สามารถดึงภาพกลับมาได้



รูป 1.2 แสดงตัวอย่างของเทคนิคต่าง ๆ ของการประมวลผลภาพในระบบดิจิทัล

เอกสารนี้เป็นเอกสารที่สงวน โดยทางซ้ายมือ เป็นภาพเดิม และภาพขวามือ เป็นภาพที่ได้จากการดำเนินการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังขอสงวนสิทธิ์ และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รูป 1.2 (a) เป็นภาพพื้นผิวของดาวอังคาร ซึ่งไม่ชัดเจนเนื่องจากถูกรบกวน  
 ในระหว่างที่ส่งสัญญาณมา ยังโลก จะเห็นเป็นเส้นแหว่ง ซึ่งเมื่อผ่าน  
 ขบวนการทางคอมพิวเตอร์แล้วสัญญาณรบกวนจะ หายไปดังรูป 1.2 (b)  
 รูปที่ 1.2 (c) และ 1.2 (d) แสดงการปรับระดับความเข้มของภาพที่ได้จาก  
 การถ่ายเอ็กซ์เรย์ โดยใช้เทคนิคการปรับปรุงความเข้มของขอบภาพ  
 รูป 1.2 (e) เป็นภาพไหวเื่องมาจากการเคลื่อนไหวของกล้อง ขณะเปิดหน้า  
 กล้อง รูป 1.2 (f) เป็นผลที่ได้จากการผ่านอัลกอริทึมเลื่อนจุดใน  
 ภาพต้นแบบให้กลับเป็นภาพนิ่ง

วัตถุโดยทั่ว ๆ ไปมีลักษณะเป็นเสารวมมิติ การสร้างภาพของวัตถุเหล่านี้ใช้เทคนิคพื้นฐานในการสร้างภาพวัตถุสามมิติ จากโปรเจกชัน 2 มิติ ได้ถูกนำมาใช้อย่างกว้างขวาง เช่น การนำภาพจากการฉายรังสีเอ็กซ์ ซึ่งทางการแพทย์ เรียกว่า คอมพิวเตอร์โทโมกราฟี (Computer tomography : CT) การพัฒนาอุปกรณ์ต่าง ๆ ในด้านเรดิโอแอสโตรโนมี (radio astronomy) และยังนำไปประยุกต์ใช้งานทางด้านศิลปะ และการตรวจสอบทางอุตสาหกรรม

ตั้งนับปัญหาของการประมวลผลภาพทางดิจิทัล จึงอยู่ที่การทำความเข้าใจกับระหว่างเครื่องคอมพิวเตอร์กับมนุษย์ และยังต้องการวิชาการใหม่ ๆ เกี่ยวกับ ความสามารถทางเครื่องจักรอุปกรณ์ ในการกวาดภาพ (image scanning) และการแสดงผล (display) ในราคาต่ำลงและได้ประสิทธิภาพสูงขึ้น อย่างไรก็ตาม ถ้าปราศจากอัลกอริทึมฟาสต์ฟูเรียร์ทรานสฟอร์ม (fast fourier transform) ของคูเลย์ (Cooley) และ ทูคีย์ (Tukey) และผู้ร่วมงานที่ค้นพบในปี 1965 แล้ว เทคโนโลยีด้านภาพก็ยังคงอยู่แต่เพียงในเรื่องของออปติก (optic)

## 1.2 เนื้อหาของวิทยานิพนธ์

จุดมุ่งหมายของโครงการนี้ เพื่อศึกษาและวิจัยแนวทางในการนำเอาไมโครคอมพิวเตอร์มาใช้ในงานประมวลผลภาพทางดิจิทัล โดยเฉพาะอย่างยิ่งการสร้างโปรแกรมลดข้อมูลภาพ เมื่อนำไปใช้กับกล้องขาวดำ 2 ระดับ ภาพขาวดำ 256 ระดับสี และภาพสีที่เนื้อหาไปใช้

ของวิทยาลัยเทคนิครายละ เขียด ดังนี้

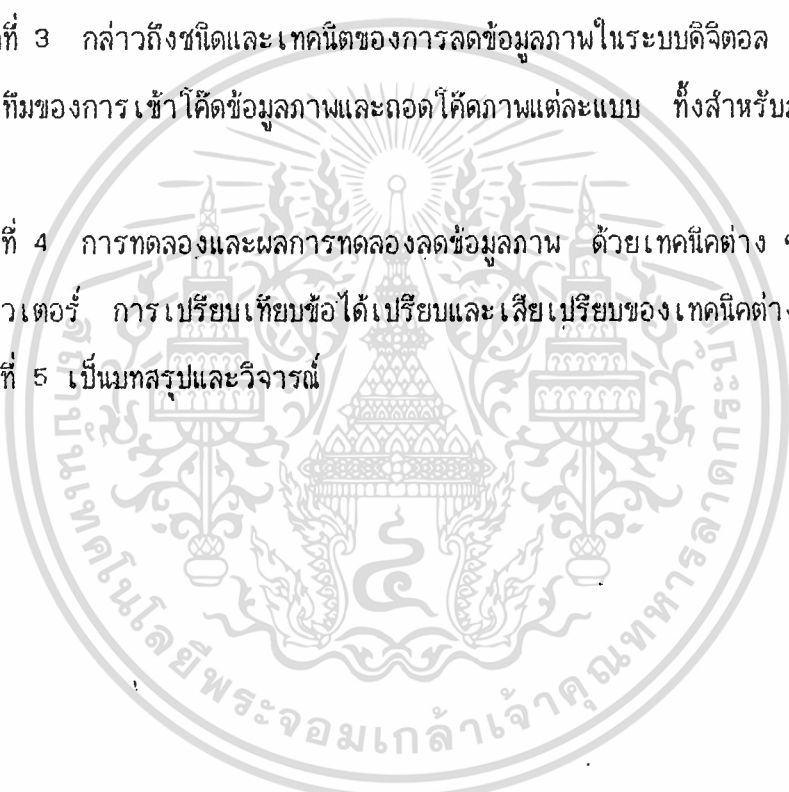
บทที่ 1 กล่าวถึงประวัติความเป็นมาตลอดจนการพัฒนาระบบเทคโนโลยี ทางด้านขบวนการประมวลผลภาพในระบบดิจิทัล

บทที่ 2 กล่าวถึงความรู้พื้นฐาน ด้านสัญญาณภาพที่เป็นสัญญาณทางดิจิทัล โดยจะมีรายละเอียดเกี่ยวกับการแทนภาพในระบบดิจิทัล ระบบการประมวลผลภาพทางดิจิทัล ตลอดจนคณิตศาสตร์พื้นฐานที่เกี่ยวข้องในการลดข้อมูลภาพ

บทที่ 3 กล่าวถึงชนิดและเทคนิคของการลดข้อมูลภาพในระบบดิจิทัล ตลอดจนถึง อัลกอริทึมของการเข้า-ออกข้อมูลภาพและถอด-ใส่ภาพแต่ละแบบ ทั้งสำหรับภาพขาว-ดำ และภาพสี

บทที่ 4 การทดลองและผลการทดลองลดข้อมูลภาพ ด้วยเทคนิคต่าง ๆ กัน โดยใช้ไมโครคอมพิวเตอร์ การเปรียบเทียบข้อได้เปรียบและเสียเปรียบของเทคนิคต่าง ๆ

บทที่ 5 เป็นบทสรุปและวิจารณ์





$$0 < f(x,y) < \infty \quad (2.1.1)$$

โดยธรรมชาติของแสง ซึ่งจะต้องมีแหล่งกำเนิดแสงและส่วนที่สะท้อนของแสง ดังนั้นเราสามารถแยกฟังก์ชัน  $f(x,y)$  ออกเป็น 2 ส่วนคือ อิลลูมินันซ์คอมโพเนนต์ (illumination component) และ รีเฟล็กแทนท์คอมโพเนนต์ (reflectant component) จะได้ว่า

$$f(x,y) = i(x,y) \times r(x,y) \quad (2.1.2)$$

เมื่อ

$$0 < i(x,y) < \infty \quad (2.1.3)$$

และ

$$0 < r(x,y) < 1 \quad (2.1.4)$$

สมการ (2.1.4) แสดงให้เห็นว่า ฟังก์ชันการสะท้อนถูกจำกัดขอบเขตระหว่าง 0 (ซึ่งหมายถึง การดูดซึมโดยสมบูรณ์) และ 1 (ซึ่งหมายถึง การสะท้อนโดยสมบูรณ์) ธรรมชาติของ  $i(x,y)$  ขึ้นอยู่กับแหล่งกำเนิดแสง ในขณะที่  $r(x,y)$  ขึ้นอยู่กับวัตถุที่สะท้อนแสงมาเข้าตา

ดังที่กล่าวมาแล้ว ความเข้มของภาพที่จุด  $(x,y)$  เราเรียกว่า ระดับสีเทา (Gray level)  $I$  จากสมการที่ (2.1.2) ถึง (2.1.4) จะเห็นว่า  $I$  ควรอยู่ในช่วง

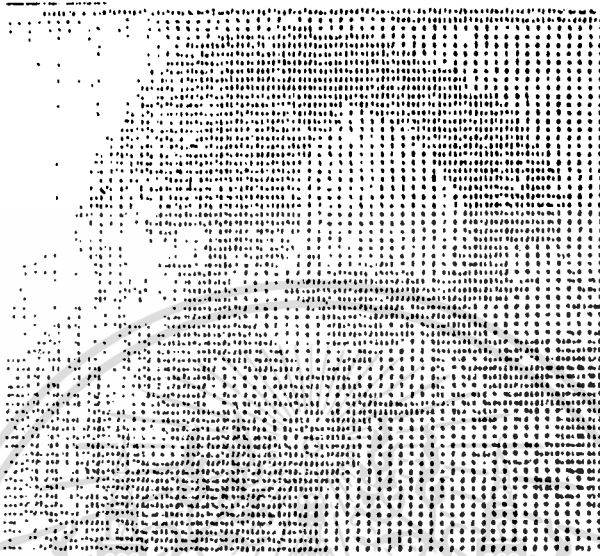
$$L_{\min} \leq I \leq L_{\max} \quad (2.1.5)$$

ในทางทฤษฎี  $L_{\min}$  ต้องมีค่าบวก ในขณะที่  $L_{\max}$  ต้องมีค่าที่น้อยกว่าอนันต์ ในทางปฏิบัติ  $L_{\min} = L_{\min} r_{\min}$  และ  $L_{\max} = L_{\max} r_{\max}$  ช่วงของ  $(L_{\min}, L_{\max})$  เราเรียกว่า ช่วงของระดับสีเทา ในทางปฏิบัติโดยใช้หลักคณิตศาสตร์ เรานิยมปรับช่วง  $(L_{\min}, L_{\max})$  ให้เป็นช่วง  $(0, L)$  โดย  $L = 0$  หมายถึงดำสนิท และ  $L = 1$  หมายถึงขาว

## 2.2 การแทนภาพด้วยข้อมูลแบบดิจิทัล

ภาพดิจิทัล (digital image) เป็นภาพที่ถูกแปลงมาจากภาพอนาล็อก อยู่ในรูปตัวเลข โดยภาพอนาล็อกถูกแบ่งเป็นพื้นที่สี่เหลี่ยมเล็ก ๆ ที่เรียกว่า พิกเซล (pixel) ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

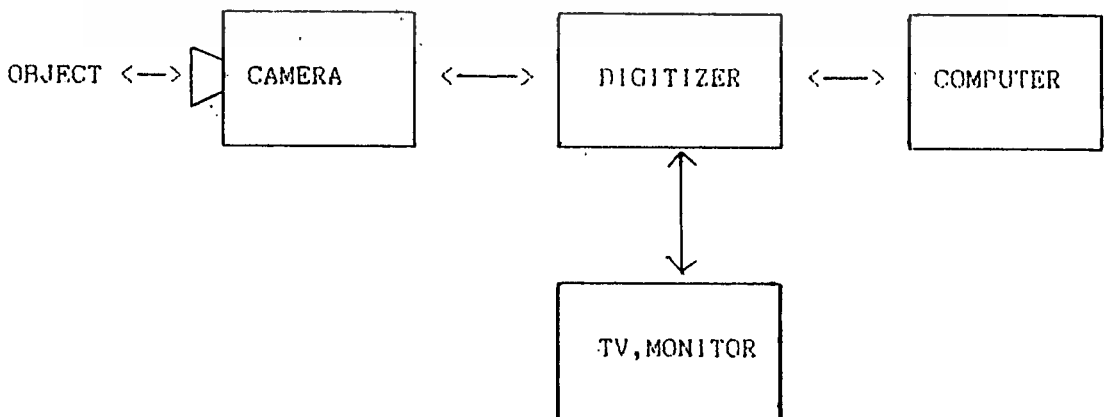
ในแต่ละพิกเซล จะถูกระบุตำแหน่งโดย  $(x,y)$  และค่าระดับสีเทาของพิกเซลนั้น คือค่าของ  $f(x,y)$  รูป 2.2 เป็นภาพดิจิทัลขนาด  $64 \times 64$  พิกเซล<sup>2</sup>



รูป 2.2 แสดงภาพดิจิทัลขนาด  $64 \times 64$  พิกเซล<sup>2</sup>

### 2.3 ระบบการประมวลผลภาพทางดิจิทัล

ระบบการประมวลผลภาพประกอบด้วย 3 ส่วนใหญ่ ๆ คือ ส่วนเปลี่ยนสัญญาณแอนะล็อก ให้เป็นสัญญาณทางด้านดิจิทัล ซึ่งเรียกว่า ดิจิไทเซอร์ (Digitizer) ส่วนประมวลผล (Processing) และส่วนแสดงผล (display) แสดงในรูป 1.1





จากรูปที่ 2.3 ส่วนแรกคือ กล้องถ่ายภาพนิ่งเป็นสัญญาณแอนะล็อก ให้เป็นสัญญาณดิจิทัล กล้อง (CAMERA) เปรียบเสมือนดวงตาของมนุษย์ ทำหน้าที่เปลี่ยนภาพวัตถุ มาเป็นสัญญาณทางไฟฟ้าและส่งให้ดีจิจิตาไลเซอร์ (DIGITIZER) ซึ่งทำหน้าที่เปลี่ยนสัญญาณไฟฟ้าให้เป็นสัญญาณดิจิทัล อุปกรณ์ส่วนนี้ ได้แก่ กล้องโทรทรรศน์ดีจิจิตาไลเซอร์ ซึ่งภายในประกอบด้วยหลอด วัตต์คอน ทำหน้าที่เป็นสื่อนำไฟฟ้าทางแสง ภาพถูกโฟกัสลงบนผิวของหลอด และถูกเปลี่ยนให้เป็นสัญญาณไฟฟ้าที่สอดคล้องกับความสว่างของภาพในแต่ละแห่งนั้น ๆ จากนั้น ทำการควอนไทซ์ (quantizing) ข้อมูลภาพที่ได้เป็นสัญญาณดิจิทัล

ส่วนประมวลผลคือ คอมพิวเตอร์ ซึ่งเปรียบเสมือนสมอง ทำหน้าที่ประมวลผล และวิเคราะห์ข้อมูลภาพ

ส่วนแสดงผล ทำหน้าที่เปลี่ยนข้อมูลตัวเลข (ซึ่งเป็นระดับสีเทา) ที่เก็บเป็นอาร์เรย์ (array) ในคอมพิวเตอร์ ให้อยู่ในรูปที่เหมาะสม และสื่อความหมายกับมนุษย์ได้ คือ เป็นภาพที่ปกติทั่ว ๆ ไป อุปกรณ์ในส่วนนี้ ได้แก่ มอนิเตอร์ ที่รี เครื่องพิมพ์ที่สามารถแสดงผลในรูปแบบกราฟิกได้

ภาพ 1 ภาพ ที่ถูกเปลี่ยนจากสัญญาณดิจิทัล สำหรับคอมพิวเตอร์จะมีขนาดใหญ่ขึ้นอยู่กับความละเอียดของภาพที่ต้องการ และจะมีผลทำให้ใช้เนื้อที่ในหน่วยความจำมาก ในการเก็บข้อมูลภาพ 1 ภาพ เช่น การเก็บภาพ 1 ภาพ ขนาด  $256 \times 256$  จุด<sup>2</sup> ที่มีความแตกต่างของระดับความเข้มของแต่ละจุด เท่ากับ 256 ระดับ จะต้องใช้เนื้อที่ในหน่วยความจำถึง 64 กิโลไบต์ ในการเก็บภาพนี้ ดังนั้น ในปัจจุบันนี้ได้มีการค้นคว้าวิจัย หาวิธีการที่จะเก็บภาพด้วยคอมพิวเตอร์ โดยให้ใช้เนื้อที่ในหน่วยความจำให้น้อยที่สุด และยังรักษาความละเอียดของภาพตามการใช้งานได้อีกด้วย

## 2.4 การสุ่มแบบสม่ำเสมอและควอนไทเซชัน

(Uniform sampling and Quantization)

เพื่อที่จะประมวลสัญญาณภาพด้วยระบบคอมพิวเตอร์ ฟังก์ชันของภาพ  $f(x, y)$

จะถูกทำให้เป็นสัญญาณไม่ต่อเนื่อง ทั้งระนาบของภาพ ซึ่งเราเรียกว่า การสุ่มภาพ (Image sampling) ค่าของฟังก์ชันที่ได้เรียกว่า การควอนไทเซชันระดับสีเทา (gray-level

สมมติว่าสัญญาณภาพต่อเนื่อง  $f(x,y)$  ถูกดัดแปลง ในระนาบ  $x-y$  เป็นช่วงเท่า ๆ กัน เราสามารถจัด  $f(x,y)$  ให้อยู่ในรูปเมตริกซ์ ขนาด  $N \times N$  ได้ดังสมการ (2.4.1)

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0, N-1) \\ f(1,0) & f(1,1) & \dots & f(1, N-1) \\ \dots & \dots & \dots & \dots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1, N-1) \end{bmatrix} \quad (2.4.1)$$

ทางขวาของสมการ จะเรียกว่า ภาพดิจิทัล และทุก ๆ สมาชิกของเมตริกซ์ จะเรียกว่า พิกเซล จากขบวนการสร้างภาพดิจิทัลข้างต้น จะเห็นว่า เราต้องทราบขนาดความละเอียดของภาพ  $N \times N$  พิกเซล และจำนวนระดับของระดับสีเทา ในทางปฏิบัติการทำควอนไทเซชันในระบบภาพดิจิทัล จะเป็นค่าของ 2 ยกกำลังจำนวนเต็ม คือ

$$N = 2^m \quad (2.4.2)$$

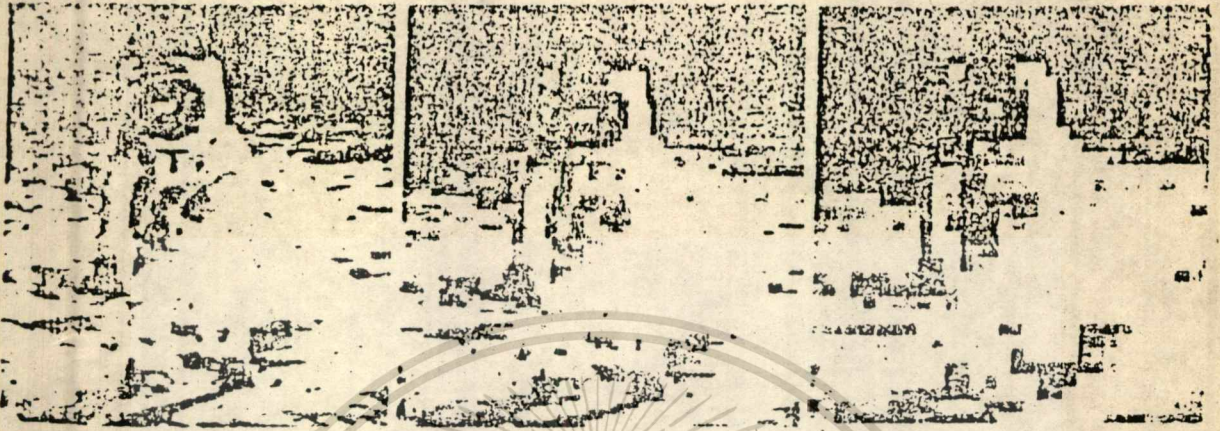
และ

$$G = 2^m \quad (2.4.3)$$

เมื่อ  $G$  คือ จำนวนระดับของระดับของระดับสีเทา ดังนั้นจำนวนบิต (bits) ที่ใช้เก็บภาพหนึ่งภาพที่ถูกดัดแปลง คือ

$$B = N \times N \times m \quad \text{บิต} \quad (2.4.4)$$

ตัวอย่างภาพขนาด  $128 \times 128$  พิกเซล และระดับสีเทาจำนวน 64 ระดับ ต้องใช้หน่วยความจำขนาด 98,304 บิต ในรูป 2.4 ได้แสดงการเปรียบเทียบภาพ เมื่อลดความละเอียดของภาพลง และตาราง 2.1 แสดงจำนวนบิตที่ใช้ในการเก็บภาพ เมื่อ  $N$  และ  $M$  เปลี่ยนไป



128 x 128

64 x 64

32 x 32

รูป 2.4 เปรียบเทียบภาพเมื่อลดความละเอียดของภาพลง

N \ m	1	2	3	4	5	6	7	8
32	128	256	512	512	1024	1024	1024	1024
64	512	1024	2048	2048	4096	4096	4096	4096
128	2048	4096	8192	8192	16384	16384	16384	16384
256	8192	16384	32768	32768	65536	65536	65536	65536
512	32768	65536	131072	131072	262144	262144	262144	262144

ตาราง 2.1 จำนวนไบท์ที่ใช้ในการเก็บภาพ เมื่อ N และ M เปลี่ยนไป

2.5 เทคนิคต่าง ๆ สำหรับการประมวลผลภาพ

เทคนิคต่าง ๆ สำหรับการประมวลผลภาพ แบ่งเป็น 4 พวกใหญ่ ๆ คือ

1. อิมเมจดิจิทัลไลเซชัน (image digitization)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ภายใต้การสงวนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ห้ามสื่อนี้เข้าหาเวกเตอร์และวีลด์ต่อเราเช่น ต้อง image ถึงเจ้า enhancement รั้งที่มีกาทำไปใช้

restoration)

3. อีเมจเอนโค้ดดิ้ง (image encoding)
4. อีเมจรีคอนสตรัคชัน (image reconstruction)

### 2.5.1 อีเมจดิจิตาไลเซชัน

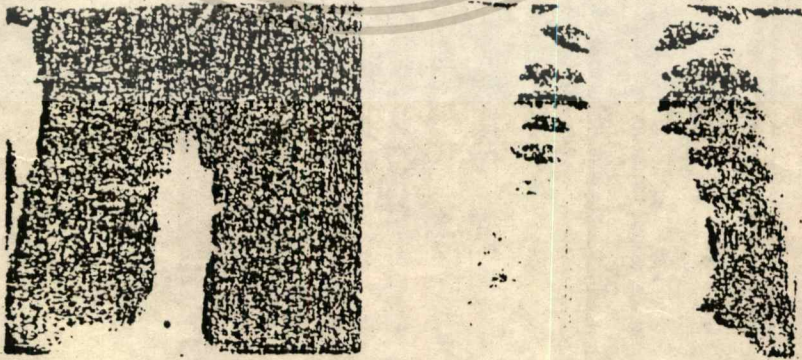
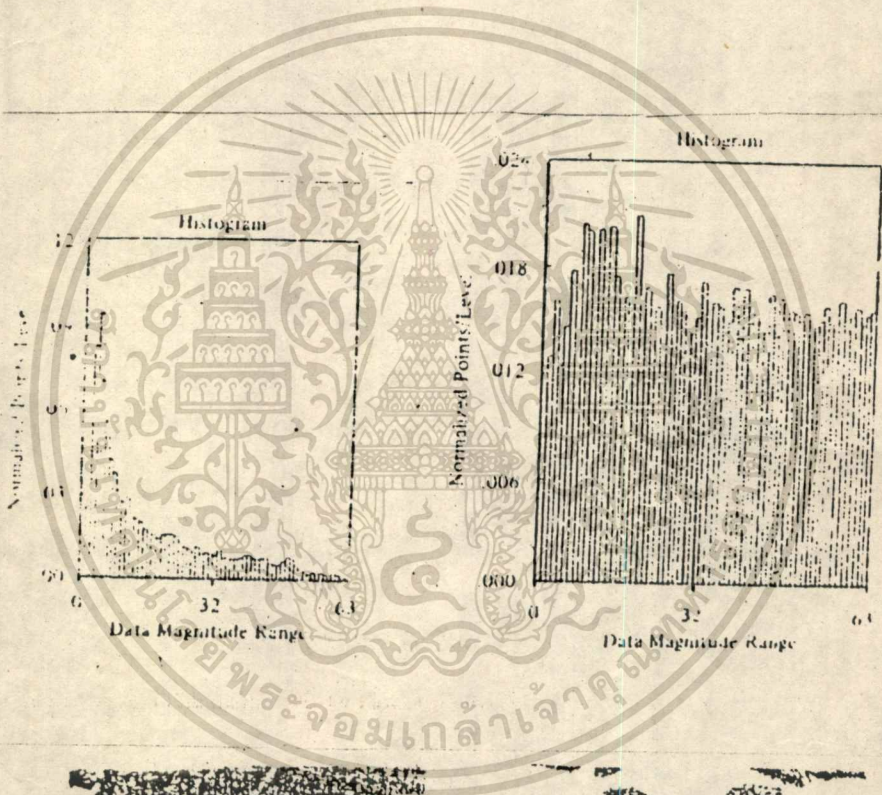
ดังได้กล่าวมาแล้วถึงความหมายของการดิจิตาไลเซชัน ซึ่งความละเอียดของภาพที่ได้ก็ขึ้นกับการควมละเอียดของภาพ ในปัจจุบันเครื่องมือที่ใช้ทำขบวนการนี้ที่เรียกว่า ดิจิตาไลเซอร์ (digitizer) ดิจิตาไลเซอร์สามารถเปลี่ยนสัญญาณแอนาลอกเป็นสัญญาณดิจิตอลได้ ดังนั้น ข้อมูลภาพที่ได้จึงเก็บเป็นเลขไบนารี โดยใช้ดิจิตาไลเซอร์เป็นตัวจัดการ

### 2.5.2 อีเมจเอนแฮนเม้นต์และรีสโตเรชัน

อีเมจเอนแฮนเม้นต์เป็นการทำภาพให้อยู่ในรูปที่เหมาะสมขึ้น มีความคมชัดมากยิ่งขึ้น สำหรับการนำไปใช้งานเฉพาะอย่าง กล่าวคือ วิธีที่ทำภาพ หรือปรับปรุงภาพเอ็กซ์เรย์ อาจจะไม่เป็นวิธีที่ดี เมื่อนำมาปรับปรุงภาพถ่ายดาวเคราะห์ที่ส่งมาจากการสำรวจทางอวกาศ

วิธีปรับปรุงคุณภาพของภาพ (enhancement) มีหลายวิธี ดังนี้

1. คอนทราสต์เอนฮานเมนท์ (Contrast enhancement) เป็นวิธีที่ทำการเพิ่มความชัดขึ้น โดยอาศัยฮิสโตแกรม อาจใช้แบบลิเนียร์สเตรท (linear stretch), ฟิลล์เพียร์สเตรท (piecewise linear stretch) หรือ อีควอลไลเซชัน (equalization)



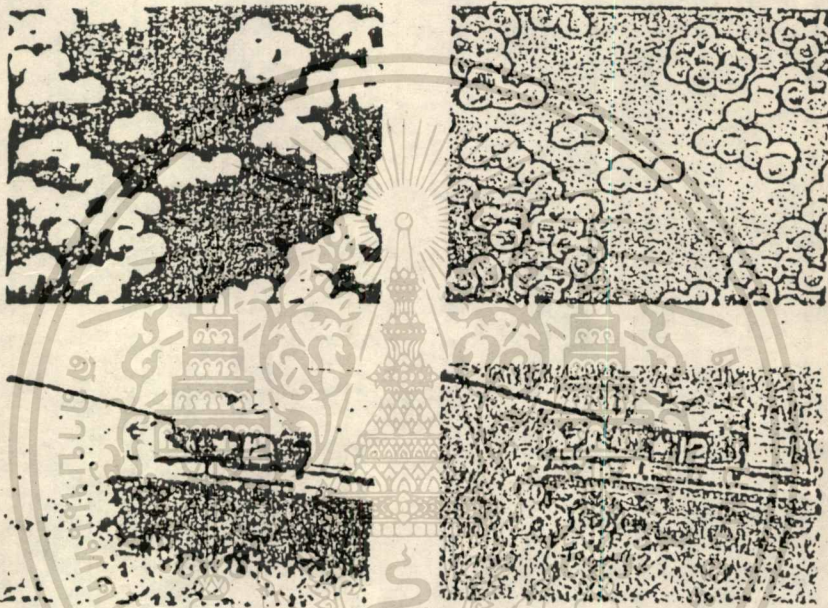
รูป 2.5 แสดงการทำคอนทราสต์เอนฮานเมนท์ จากภาพที่มีการกระจายของเกรย์โทน (gray

tone) ต่ำซึ่งภาพจะมีดี ให้มีการกระจายสูงขึ้น จะได้ภาพที่ชัดเจน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้เผยแพร่โดยไม่ขออนุญาตจากเจ้าของลิขสิทธิ์

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เอดจ์เอนฮานเม้นท์ (Edge enhancement) เป็นการแยกความแตกต่าง  
ของจุดภาพที่ใกล้เคียงกัน เพื่อหาขอบเขตของภาพ



รูป 2.6 แสดงการทำคอนทราสต์ที่เอนฮานเม้นท์ เพื่อให้ได้ลักษณะ โครงร่างของภาพออกมา

2.6 a) สามารถใช้เอดจ์เอนฮานเม้นท์ นับจำนวนเม็ดเลือดในภาพได้

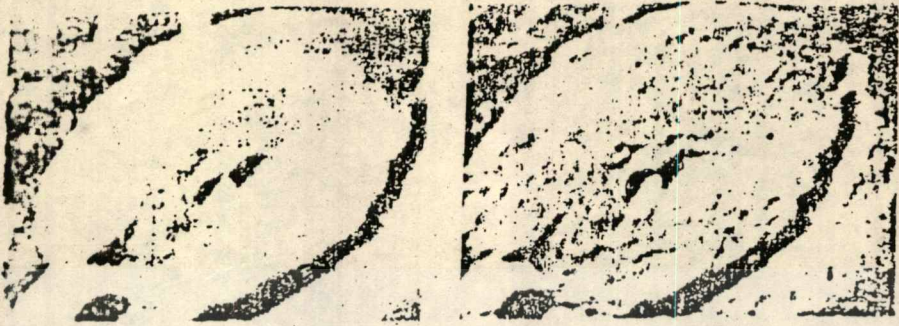
2.6 b) แสดงโครงร่างของรถยนต์บนถนนที่ชัดเจนนั่น

3. การประมวลผลภาพทูโตลเลอร์ (Pseudo-color image processing)

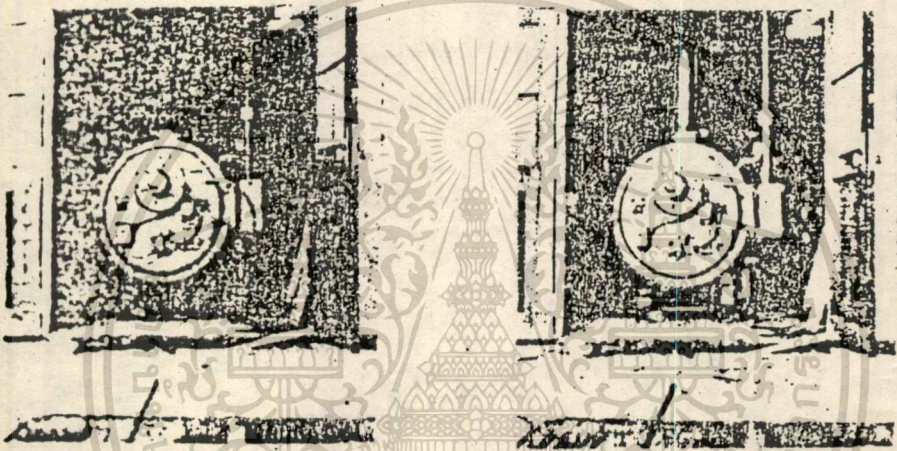
เป็นการใช้เทคนิคของการทำเดนซิตีสไลซิง (density slicing) และการใส่สีเทียมให้  
กับภาพขาว-ดำ ที่มีระดับสีเทาต่าง ๆ กัน

4. การกรองภาพ (Filtering) เพื่อให้ภาพเรียบ (smoothing) หรือ

คมชัด (sharpening) โดยใช้ตัวกรองความถี่ต่ำ (low pass filter) หรือ ตัว  
กรองความถี่สูง (high pass filter) ตามลำดับ



a)



b)

รูป 2.7 แสดงการทำการกรองสัญญาณภาพ เพื่อให้ภาพคมชัดยิ่งขึ้น (filtering enhancement)

- a) แสดงการทำให้ภาพชัดขึ้น (sharpening) ของภาพนี้เห็นผิวดวงจันทร์ ซึ่งได้ภาพคมชัดขึ้น
- b) แสดงการทำการกรองสัญญาณภาพ (high pass filter) กับภาพเพื่อให้แสดงรายละเอียดภายในห้องได้ชัดเจนขึ้น

**อิมเมจรีสตอเรชั่น**

เป็นขบวนการในการสร้างภาพกลับคืน โดยการหาค่า ขาดหาย และแก้ความ

คลาดเคลื่อน เนื่องมาจากข้อมูลในภาพผิดเพี้ยนไป หรือเป็นขบวนการสร้างภาพกลับคืนจากการคำนวณการคำนวณที่สูญเสียไปทั้งหมดเนื่องจากปรากฏการณ์ต่างตัวอย่างนี้โดยใช้หลักการของนิพจน์ฟูริเยร์เชิงเส้นไปใช้

(linear algebra)

### 2.5.3 อีเมจเอนโค๊ดดิ้ง

เป็นการใช้เทคนิคต่าง ๆ เพื่อเข้าโค๊ดข้อมูล เนื่องจากข้อมูลภาพที่ได้จะถูกเก็บในลักษณะเป็นจำนวนไบท์ ดังตาราง 2.1 ซึ่งถ้าภาพมีขนาดใหญ่ ก็ต้องใช้พื้นที่ในการเก็บมาก ด้วยข้อจำกัดของเครื่องไมโครคอมพิวเตอร์ ที่มีขนาดหน่วยความจำจำกัด

การเข้าโค๊ดข้อมูลจึงมีประโยชน์ ในด้านการลดพื้นที่ในการเก็บข้อมูลภาพดังกล่าว มาก ผลของการเข้าโค๊ดข้อมูลนี้ เรียกว่าเป็น การลดข้อมูลภาพ (Data reduction หรือ Data compression) ซึ่งเป็นเนื้อหาที่ทำในโครงการนี้ รายละเอียดของการลดข้อมูลภาพ ได้อธิบายในบทที่ 3 นอกจากนี้ การเข้าโค๊ดข้อมูลยังมีประโยชน์ในการช่วยลดปริมาณข้อมูลภาพ ที่ใช้ในระบบการสื่อสาร เช่น การส่งภาพถ่ายจากอวกาศมายังโลก การส่งข้อมูลภาพผ่าน โมเด็ม (modem) เป็นต้น

### 2.5.4 อีเมจรีคอนสตรัคชัน

เป็นวิธีการสร้างภาพตัดขวางของวัตถุ โดยไม่ต้องผ่า หรือทำลายวัตถุ เพื่อประมวลผลโดยใช้คอมพิวเตอร์ เราเรียกการสร้างภาพตัดขวางด้วยคอมพิวเตอร์ว่า คอมพิวเตอร์โทโมกราฟี (Computer tomography)

## 2.6 การทรานส์ฟอร์มภาพ (Image transform)

คณิตศาสตร์ที่ใช้ทรานส์ฟอร์ม เน้นฟูเรียร์ทรานส์ฟอร์ม เนื่องจากมีการนำไปใช้งานได้อย่างกว้างขวาง เพื่อแก้ปัญหาทางด้านขบวนการประมวลผลภาพ ในหัวข้อ

2.6.1 ที่จะแนะนำถึงฟูเรียร์ทรานส์ฟอร์ม ของตัวแปรตัวเดียว หรือ 2 ตัวแปร ที่ต่อเนื่อง และได้เ้าหลักการนี้ไปใช้ในหัวข้อ 2.6.2 ในรูปของตัวแปรที่ไม่ต่อเนื่อง (discrete) คุณสมบัติต่าง ๆ ที่สำคัญของฟูเรียร์ทรานส์ฟอร์ม 2 มิติ ได้พัฒนาขึ้นและสรุปในหัวข้อ 2.6.3 และสุดท้ายจะอธิบายถึง อัลกอริทึม ของฟาสต์ฟูเรียร์ทรานส์ฟอร์ม (Fast Fourier Transform)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# 1. พูรีเยร์ทรานส์ฟอร์มเบื้องต้น (Concept of the Fourier Transform)

ให้  $f(x)$  เป็นฟังก์ชันต่อเนื่องของตัวแปรจำนวนจริง พูรีเยร์ทรานส์ฟอร์มของ  $f(x)$  ซึ่งใช้สัญลักษณ์  $\mathcal{F}\{f(x)\}$  นิยามโดยสมการ

$$\mathcal{F}\{f(x)\} = F(u) = \int_{-\infty}^{\infty} f(x) \exp[-j2\pi ux] dx$$

$$j = \sqrt{-1} \quad (2.6.1)$$

เมื่อกำหนดฟูรีเยร์ทรานส์ฟอร์ม  $F(u)$  เราสามารถหา  $f(x)$  ได้จากการทำ

## อินเวอร์สฟูรีเยร์ทรานส์ฟอร์ม (Inverse Fourier Transform)

$$\mathcal{F}^{-1}\{F(u)\} = f(x)$$

$$= \int_{-\infty}^{\infty} F(u) \exp[j2\pi ux] dx \quad (2.6.2)$$

สมการ (2.6.1) และ (2.6.2) เรียกว่า คู่สมการของฟูรีเยร์ทรานส์ฟอร์ม ซึ่งจริงหรือเกิดได้ (exist) ก็ต่อเมื่อ  $f(x)$  เป็นฟังก์ชันต่อเนื่อง และสามารถอินทิเกรตได้ และ  $F(u)$  ก็ต้องสามารถอินทิเกรตได้ด้วย กรณีเช่นนี้ในทางปฏิบัติมักจะเป็นจริง

ฟูรีเยร์ทรานส์ฟอร์มของฟังก์ชันตัวแปรจำนวนจริง  $f(x)$  มักจะให้ เป็นจำนวนเชิง

คือ

$$F(u) = R(u) + jI(u) \quad (2.6.3)$$

โดยที่  $R(u)$  และ  $I(u)$  คือส่วนจริงและจินตภาพของ  $F(u)$  ตามลำดับ เพื่อความ

สะดวก เราจะกระจายสมการ (2.6.3) ในรูปเอกซ์โพเนนเชียล คือ

$$F(u) = |F(u)| e^{j\phi(u)} \quad (2.6.4)$$

$$|F(u)| = [R^2(u) + I^2(u)]^{1/2} \quad (2.6.5)$$

$$\phi(u) = \tan^{-1} [I(u)/R(u)] \quad (2.6.6)$$

ขนาดของฟังก์ชัน  $F(u)$  เรียกว่า สเปกตรัมฟูรีเยร์ของ  $f(x)$  และ  $\phi(u)$  คือ

เฟสของฟังก์ชันของสเปกตรัมจะได้

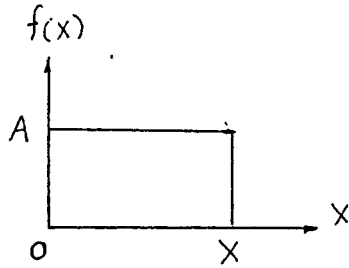
$$E(u) = |F(u)|^2$$

$$= R^2(u) + I^2(u) \quad (2.6.7)$$

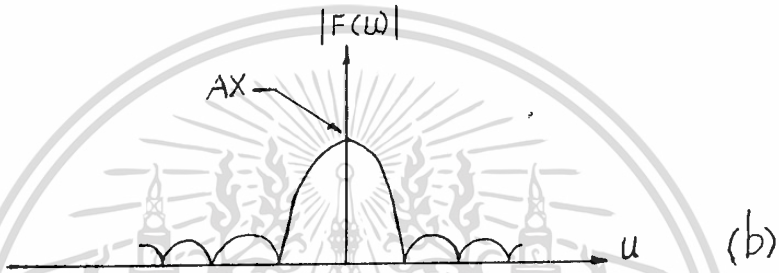
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งเรียกว่า สเปกตรัมพลังงานของ  $f(x)$  มีลักษณะดังรูป 2.8



(a)



(b)

รูป 2.8 พิงก์ชันง่าย ๆ และฟูเรียร์สเปกตรัมของมัน

ฟูเรียร์ทรานส์ฟอร์มสามารถใช้กับฟังก์ชัน  $f(x,y)$  ซึ่งมีตัวแปรอิสระ 2 ตัวได้ ถ้าฟังก์ชัน  $f(x,y)$  เป็นฟังก์ชันที่ต่อเนื่อง และสามารถอินทิเกรตได้ และ  $F(u,v)$  สามารถอินทิเกรตได้ คู่สมการของฟูเรียร์ทรานส์ฟอร์มจะเป็น

$$\mathcal{F}\{f(x,y)\} = F(u,v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) \exp[-j2\pi(ux+vy)] dx dy \quad (2.6.8)$$

และ

$$\mathcal{F}\{F(u,v)\} = f(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u,v) \exp[j2\pi(ux+vy)] du dv \quad (2.6.9)$$

เมื่อ  $u$  และ  $v$  เป็นตัวแปรทางความถี่

ในทำนองเดียวกันกรณี 1 มิติ จะได้สเปกตรัมฟูเรียร์ เฟส และสเปกตรัมพลังงาน ดังสมการต่อไปนี้

$$|F(u,v)| = [R^2(u,v) + I^2(u,v)]^{1/2} \quad (2.6.10)$$

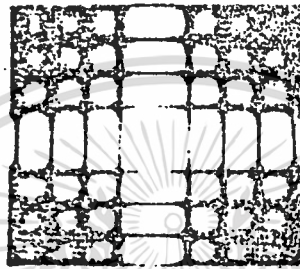
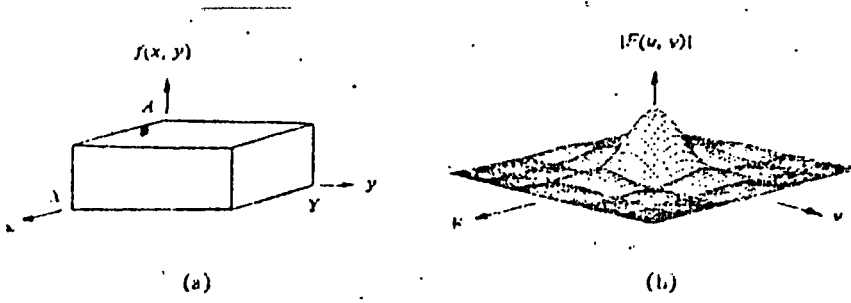
$$\phi(u,v) = \tan^{-1} [ I(u,v) / R(u,v) ] \quad (2.6.11)$$

และ

$$E(u,v) = R^2(u,v) + I^2(u,v) \quad (2.6.12)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาติให้นำไปใช้ประโยชน์ด้านการค้า

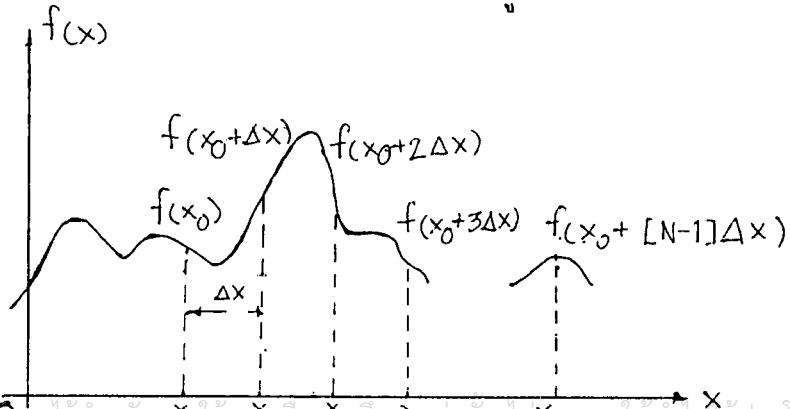
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.9 a) ฟังก์ชัน 2 มิติ  
 b) ฟูเรียร์สเปกตรัมของมัน  
 c) แสดงสเปกตรัมในรูปฟังก์ชันความเข้ม

2.6.2 ฟูเรียร์ทรานส์ฟอร์มแบบไม่ต่อเนื่อง (The Discrete Fourier Transform)

สมมติให้ฟังก์ชัน  $f(x, y)$  เป็นฟังก์ชันที่ถูกสุ่ม (sampling) ในช่วงห่างเท่า ๆ กัน คือ  $f(x_0), f(x_0 + \Delta x), f(x_0 + 2\Delta x), f(x_0 + 3\Delta x), \dots, f(x_0 + [N-1]\Delta x)$  เป็นจำนวน  $N$  ตัวอย่าง โดยมีระยะห่าง  $x$  หน่วย ดังรูป 2.10



เอกสารนี้เป็นเอกสารที่รวบรวมไว้สำหรับนักศึกษาชั้นปีที่ 1 ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี เพื่อใช้ในการเรียนการสอน  
 วิชาคณิตศาสตร์ทุกชั้นปี ไม่ควรนำเอกสารนี้ไปเผยแพร่โดยไม่ได้รับอนุญาตจากทางมหาวิทยาลัย

รูป 2.10 ฟังก์ชันภาพต่อเนื่องที่ถูกสุ่มในช่วงเท่า ๆ กัน

เราอาจนิยาม

$$f(x) = f(x + x\Delta x) \tag{2.6.13}$$

เมื่อค่า  $x$  เป็นค่าของจุดไม่ต่อเนื่อง ที่  $0, 1, 2, \dots, N-1$  อีกนัยหนึ่ง ลำดับ  $\{ f(0), f(1), f(2), \dots, f(N-1) \}$  จะเป็น  $N$  ตัวอย่างที่ถูกสุ่มในช่วงเท่า ๆ กัน จากฟังก์ชันต่อเนื่อง

จากนิยามข้างต้น เราสามารถเขียนคู่สมการฟูเรียร์ทรานส์ฟอร์มของฟังก์ชันสุ่ม  $f(x)$  ซึ่งเรียกว่า ฟูเรียร์ทรานส์ฟอร์มแบบไม่ต่อเนื่อง โดย

$$F(u) = 1/N \sum_{x=0}^{N-1} f(x) \exp[-j2\pi ux/N] \tag{2.6.14}$$

เมื่อ  $u = 0, 1, 2, \dots, N-1$  และ

$$f(x) = \sum_{u=0}^{N-1} F(u) \exp[j2\pi ux/N] \tag{2.6.15}$$

เมื่อ  $x = 0, 1, 2, \dots, N-1$

ในสมการที่ (2.6.15) ค่าของ  $u = 0, 1, 2, \dots, N-1$  จะสมมูลกับค่าที่สุ่มมาที่จุด  $0, \Delta u, 2\Delta u, \dots, (N-1)\Delta u$  ซึ่งจะเห็นได้ว่า  $F(u)$  ในกรณีนี้ จะใช้แทน  $F(u)$  และ  $u$  กับ  $x$  มีความสัมพันธ์กันโดย

$$u = 1/N\Delta x \tag{2.6.16}$$

ในกรณีฟังก์ชัน 2 ตัวแปร คู่สมการของฟูเรียร์ทรานส์ฟอร์มจะกำหนดโดย

$$F(u, v) = 1/MN \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} f(x, y) \exp[-j2\pi(ux/m + vy/n)] \tag{2.6.17}$$

เมื่อ  $u = 0, 1, 2, \dots, M-1$  ;  $v = 0, 1, 2, \dots, N-1$

และ

$$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) \exp[j2\pi(ux/M + uv/N)] \tag{2.6.18}$$

เมื่อ  $u = 0, 1, 2, \dots, M-1$  และ  $y = 0, 1, 2, \dots, N-1$

จะเห็นว่าในกรณีนี้  $f(x, y)$  จะถูกสุ่มทั้งสองแกนด้วยระยะห่าง  $x$  และ  $y$  เท่า ๆ กัน จากการที่เราเขียน  $f(x, y)$  ในกรณีนี้จะหมายถึง ฟังก์ชันที่ถูกสุ่มแล้วนั่นก็คือ  $f(x + x\Delta x, y + y\Delta y)$  เมื่อ  $x = 0, 1, 2, \dots, M-1$  และ  $y = 0, 1, 2, \dots, N-1$

นี่เป็นสิ่งที่เราสามารถหาความสัมพันธ์ของฟังก์ชันความถี่ นั้น และฟังก์ชันภาพได้ดังนี้ ประโยชน์ด้านการคำนวณว่ากรณีใดๆทั้งสิ้น อีกหนึ่งข้อที่หัดแปลงเนื้อหา และต้องอ้างอิงเอกสารทุกครั้งที่มีการนำไปใช้

และ

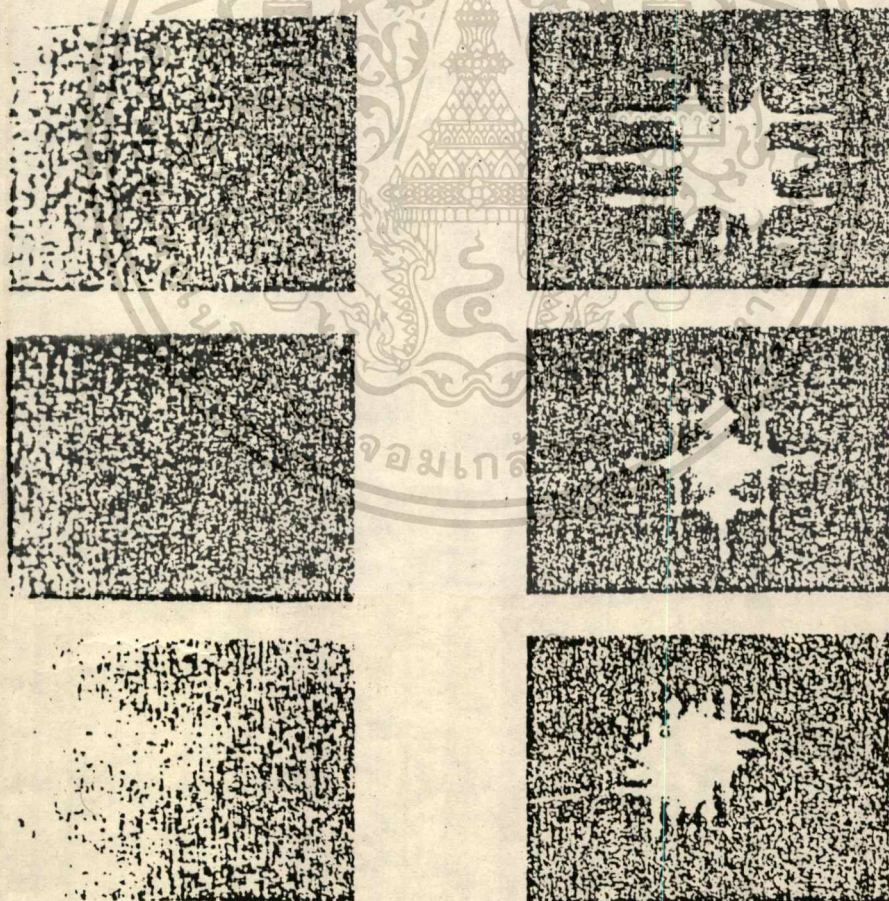
$$v = 1/N\Delta y \quad (2.6.20)$$

สำหรับฟังก์ชันของภาพที่ถูกสุ่มโดยที่  $M = N$  เราจะได้คู่ของสมการฟูเรียร์ทรานส์

ฟอร์ม

$$F(u,v) = 1/N \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \exp[-j2\pi(ux+vy)/N] \quad (2.6.21)$$

$$f(x,y) = 1/N \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u,v) \exp[j2\pi(ux+vy)/N] \quad (2.6.22)$$

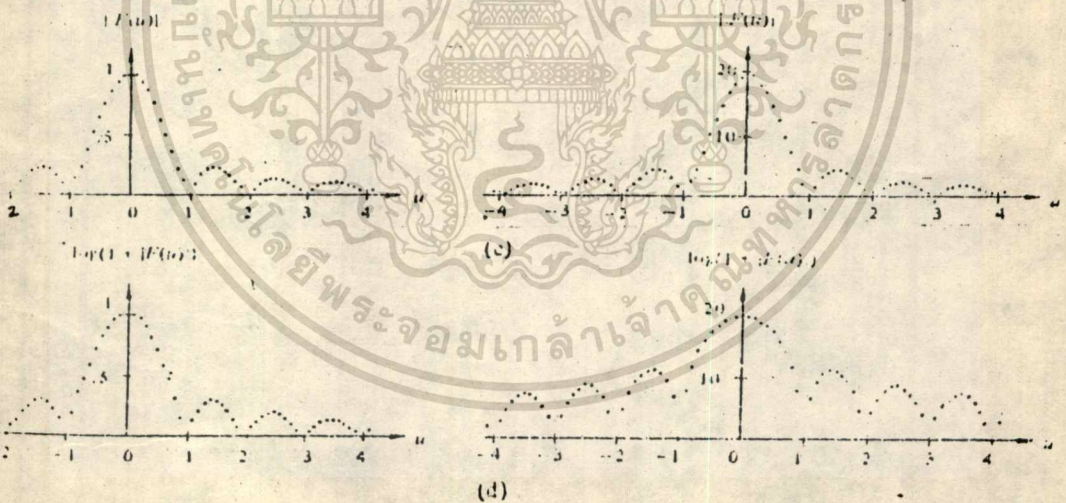


2.6.3 คุณสมบัติบางประการของฟูเรียร์ทรานส์ฟอร์ม 2 มิติ

หัวข้อต่าง ๆ ในส่วนนี้ จะเกี่ยวข้องกับภาพ และสเปกตรัมของภาพ ในรูปของ ฟังก์ชันความเข้มของภาพ ข้อมูลต่าง ๆ ที่เกี่ยวข้องกับ การทราบก่อนที่จะกล่าวถึง คุณสมบัติของฟูเรียร์ทรานส์ฟอร์ม คือ สเปกตรัมของภาพ โดยส่วนใหญ่จะลดลงอย่างรวดเร็ว เมื่อความถี่ของฟังก์ชันเพิ่มขึ้น ดังนั้น ข้อมูลที่มีความถี่สูง จะไม่ชัด เมื่ออยู่ในรูปภาพ เทคนิคสำคัญที่ใช้แก้ปัญหาที่แก้คือใช้

$$D(u, v) = 10^5 (1 + |F(u, v)|) \tag{2.6.23}$$

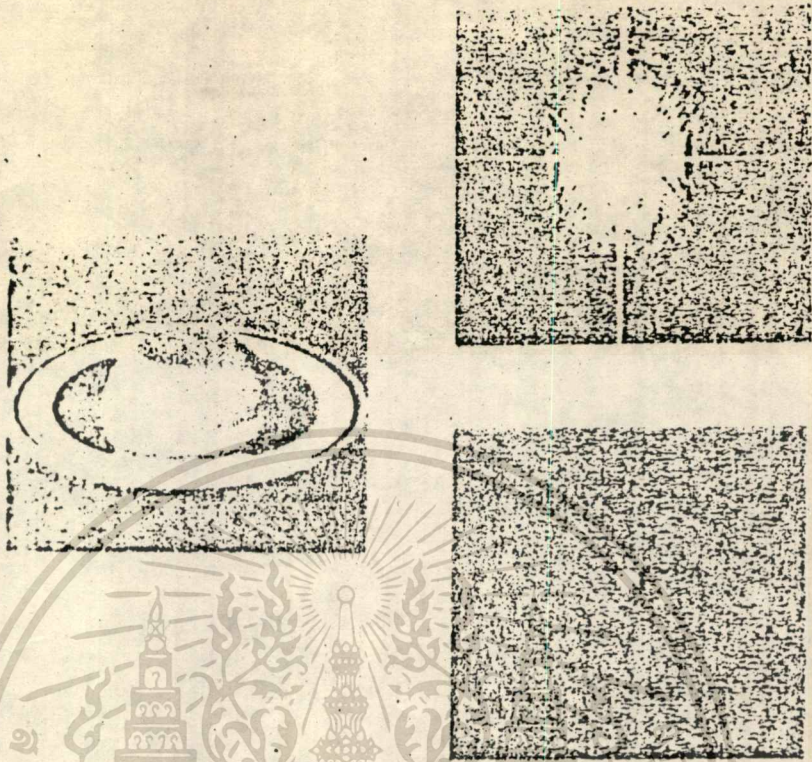
แทน  $|F(u, v)|$  สมการนี้ให้ค่าศูนย์ในระนาบความถี่ เพราะ  $D(u, v) = 0$  เมื่อ  $|F(u, v)|$  และยังมีค่าเป็นบวกเสมอด้วย รูปที่ 3.6 แสดงให้เห็นผลของ อัลกอริทึม ที่มีต่อฟูเรียร์ทรานส์ฟอร์มของฟังก์ชัน ในรูป 2.12



รูป 2.12 ผลของการใช้ค่าลอการิทึม (log) กับฟูเรียร์สเปกตรัม

การใช้สมการ (2.6.23) ในการแสดงผลของสเปกตรัม 2 มิติ ทำให้เห็นความหมายของการทำฟูเรียร์ทรานส์ฟอร์มได้ชัดเจน ดังแสดงในรูป 2.13

เอกสารนี้เป็นเอกสารที่สวชนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่จ่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.13 a) รูปดาวเสาร์ b) ผลของ  $|F(u,v)|$  c) ผลของ  $\log(1+|F(u,v)|)$

2.6.3.1 แบ่งการคำนวณ (Separability)

คู่สมการฟูเรียร์ทรานส์ฟอร์มแบบไม่ต่อเนื่อง ในสมการ (2.6.17) และ (2.6.18)

เขียนแยกกันได้ดังนี้

$$F(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} \exp[-j2\pi ux/N] \sum_{y=0}^{N-1} f(x,y) \exp[-j2\pi vy/N]$$

เมื่อ  $u,v = 0, 1, 2, \dots, N-1$  (2.6.24)

และ

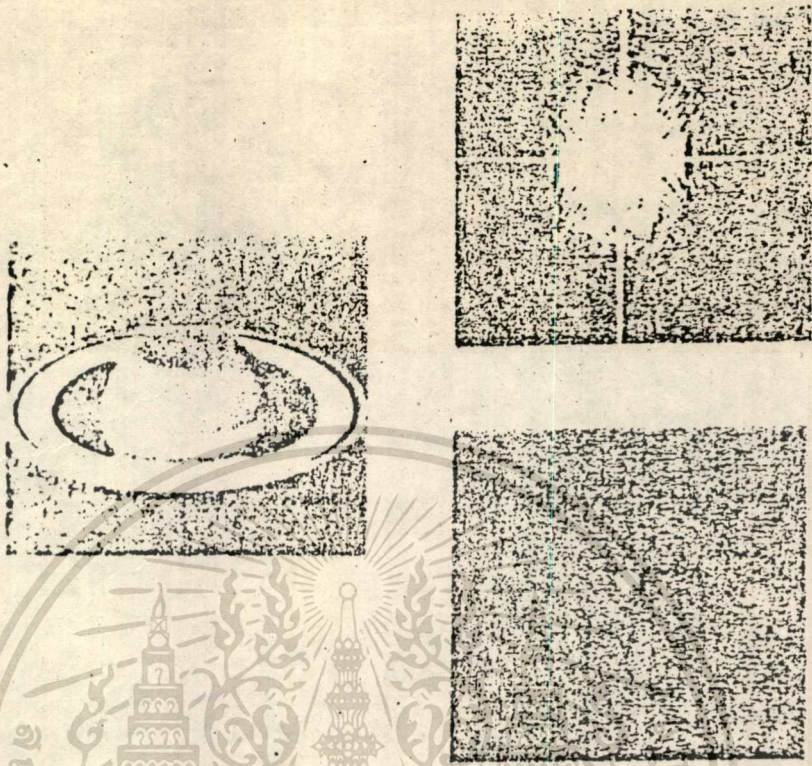
$$f(x,y) = \frac{1}{N} \sum_{u=0}^{N-1} \exp[j2\pi ux/N] \sum_{v=0}^{N-1} F(u,v) \exp[j2\pi vy/N]$$

เมื่อ  $x,y = 0, 1, \dots, N-1$  (2.6.25)

การที่ทำได้เช่นนี้ ก็เพื่อหาค่า  $F(u,v)$  หรือ  $f(x,y)$  จากฟูเรียร์ทรานส์ฟอร์ม หรืออินเวอร์สฟูเรียร์ทรานส์ฟอร์ม มิติเดียว 2 ครั้ง ต่อเนื่องกัน จะเห็นได้ชัดว่า สมการ

(2.6.24) เขียนได้ดังรูป

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ผู้เขียนขอสงวนสิทธิ์ในข้อนี้  $\sum_{x=0}^{N-1}$  เนื้อหา (และ)  $\exp[j2\pi ux/N]$  เอกสาร (2.6.25) การนำไปใช้



รูป 2.13 a) รูปดาวเสาร์ b) ผลของ  $|F(u,v)|$  c) ผลของ  $\log(1+|F(u,v)|)$

2.6.3.1 แบ่งการคำนวณ (Separability)

คู่สมการฟูเรียร์ทรานส์ฟอร์มแบบไม่ต่อเนื่อง ในสมการ (2.6.17) และ (2.6.18)

เขียนแยกกันได้ดังนี้

$$F(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} \exp[-j2\pi ux/N] \sum_{y=0}^{N-1} f(x,y) \exp[-j2\pi vy/N]$$

เมื่อ  $u,v = 0, 1, 2, \dots, N-1$  (2.6.24)

และ

$$f(x,y) = \frac{1}{N} \sum_{u=0}^{N-1} \exp[j2\pi ux/N] \sum_{v=0}^{N-1} F(u,v) \exp[j2\pi vy/N]$$

เมื่อ  $x,y = 0, 1, \dots, N-1$  (2.6.25)

การที่ทำได้เช่นนี้ ก็เพื่อหาค่า  $F(u,v)$  หรือ  $f(x,y)$  จากฟูเรียร์ทรานส์ฟอร์ม หรืออินเวอร์สฟูเรียร์ทรานส์ฟอร์ม มิติเดียว 2 ครั้ง ต่อเนื่องกัน จะเห็นได้ชัดว่า สมการ

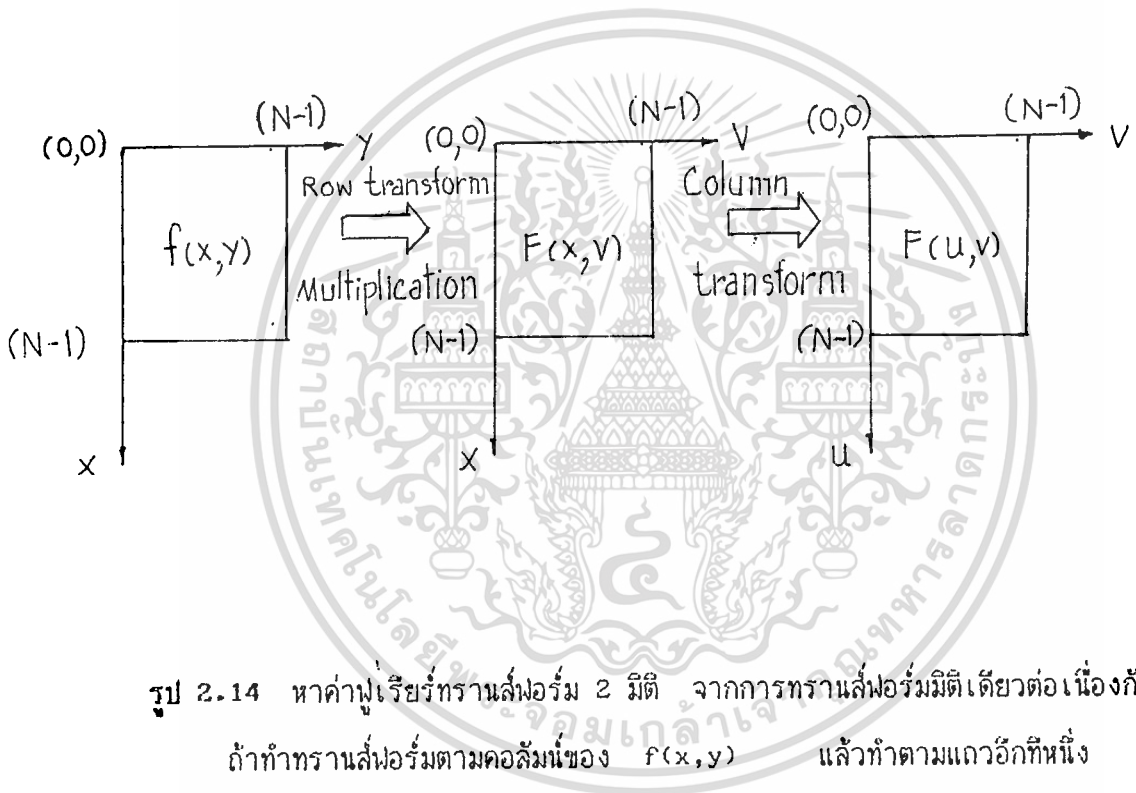
เอกส. 2.6.24 และ 2.6.25 เขียนได้ดังรูป สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรรมใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอก และต้องอ้างอิงถึงที่มาของเอกสารฉบับนี้  

$$F(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} f(x,y) \exp[-j2\pi ux/N]$$
 (2.6.25) การนำไปใช้

เมื่อ

$$F(x,v) = \frac{1}{N} \sum_{y=0}^{N-1} f(x,y) \exp(-j2\pi yv/N) \quad (2.6.27)$$

$x$  แต่ละค่าในวงเล็บเป็นทรานส์ฟอร์มมิติเดียว ที่ความถี่  $v = 0, 1, 2, \dots, N-1$  ดังนั้นฟังก์ชัน 2 มิติ  $F(x,v)$  จะได้จากการทำทรานส์ฟอร์ม ตามแถวของ  $f(x,y)$  และคูณผลที่ได้ด้วย  $N$  และ  $F(u,v)$  ก็จะได้จากการทำทรานส์ฟอร์มตามคอลัมน์หนึ่ง ๆ ของ  $F(x,v)$  ตามสมการ (2.6.26) ขั้นตอนต่าง ๆ สรุปได้ตามรูป 2.14



รูป 2.14 หาค่าฟูรีเยร์ทรานส์ฟอร์ม 2 มิติ จากการทรานส์ฟอร์มมิติเดียวต่อเนื่องกัน

ถ้าทำทรานส์ฟอร์มตามคอลัมน์ของ  $f(x,y)$  แล้วทำตามแถวอีกทีหนึ่ง ก็จะได้

ให้ผลเช่นเดียวกัน

2.5.3.2 การแปลเคลื่อน (Translation)

คุณสมบัติ ข้อนี้ของฟูเรียร์ทรานส์ฟอร์ม อยู่ในรูป

$$f(x,y)\exp[j2\pi(u_0x + v_0y)/N] \Leftrightarrow F(u-u_0, v-v_0) \quad (2.6.28)$$

และ

$$f(x-x_0, y-y_0) \Leftrightarrow F(u, v)\exp[-j2\pi(ux_0 - vy_0)/N] \quad (2.6.29)$$

เครื่องหมายลูกศร หมายถึง สมมูลกัน ระหว่างฟังก์ชันหนึ่ง กับฟูเรียร์ทรานส์ฟอร์มของมัน หรือกลับกัน

สมการ (2.6.28) แสดงให้เห็นว่า การคูณ  $f(x,y)$  ด้วย เทอมเอ็กส์โปเนนเชียล แล้วทำทรานส์ฟอร์มผลคูณนั้น จะทำให้ระนาบความถี่ เลื่อนจุดกำเนิด (Origin) ไปอยู่ที่  $(u_0, v_0)$  เช่นเดียวกับ เมื่อคูณ  $F(u,v)$  ด้วยเทอมเอ็กส์โปเนนเชียลดังสมการ แล้วทำอินเวอร์สทรานส์ฟอร์ม จะทำให้จุดกำเนิดของระนาบพิกัดเลื่อนมายัง  $(x_0, y_0)$

$$\begin{aligned} \text{เมื่อใช้ } u_0 = v_0 = N/2 \text{ กับสมการ (2.6.28) จะได้} \\ \exp[j2\pi(u_0x + v_0y)/N] &= e^{j\pi(x+y)} \\ &= (-1)^{x+y} \end{aligned}$$

และ

$$f(x,y)(-1)^{x+y} = F(u-N/2, v-N/2) \quad (2.6.30)$$

ดังนั้น จุดกำเนิดของฟูเรียร์ทรานส์ฟอร์มของ  $f(x,y)$  สามารถย้ายมายังจุดกึ่งกลางภาพความถี่ ขนาด  $N \times N$  โดยการคูณ  $f(x,y)$  ด้วย  $(-1)^{x+y}$  ในกรณีตัวแปรเดียว จะคูณ  $f(x)$  ด้วยเทอม

$$f(x,y) \text{ จะไม่ทำให้ขนาดของฟูเรียร์ทรานส์ฟอร์มเปลี่ยนเนื่องจาก} \\ |F(u,v)\exp[-j2\pi(ux_0 + vy_0)/N]| = |F(u,v)| \quad (2.6.31)$$

ฟูเรียร์ทรานส์ฟอร์มแบบไม่ต่อเนื่อง และอินเวอร์สของมันเป็นฟังก์ชันคาบเวลา ที่มี คาบเวลา  $N$  นั่นคือ

$$F(u,v) = F(v+N, v) = F(u, v+N) = F(u+N, v+N) \quad (2.6.32)$$

คุณสมบัตินี้ได้จากการแทนค่าโดยตรง ของตัวแปร  $(u+N)$  และ  $(v+N)$  ในสมการ (2.6.17) พบว่า สิ่งจำเป็นในการกำหนดค่า  $F(u,v)$  ไม่อย่างสมบูรณ์ในโดเมนความถี่คือการค่า (frequency domain) คือการทรานส์ฟอร์มเพียง 1 คาบเวลา สำหรับ  $f(x,y)$  ใน

สเปซฮิลโดเมนกี้เช่นเดียวกัน

ฟูเรียร์ทรานส์ฟอร์มที่มีคอนจูเกทสมมาตร คือ

$$F(u, v) = F^* (-u, -v) \tag{2.6.33}$$

และที่สำคัญคือ

$$|F(u, v)| = |F(-u, -v)| \tag{2.6.34}$$

เรามักจะใช้ขนาด (magnitude) ของฟูเรียร์ทรานส์ฟอร์ม เพื่อแยกความหมายในการที่จะดูผลของสมการ (2.6.32) และ (2.6.34) เพื่อแสดงขนาดทรานส์ฟอร์ม

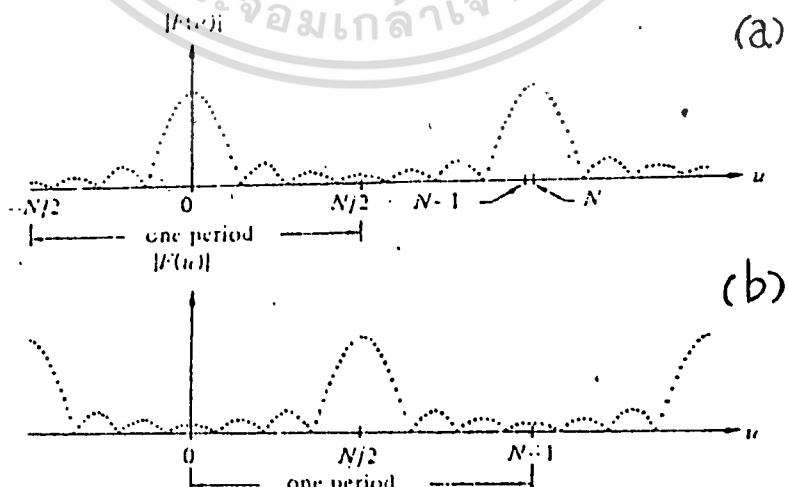
เราจะพิจารณากรณีตัวแปรเดียวกัน ซึ่ง

$$F(u) = F(u+N) \tag{2.6.35}$$

และ

$$|F(u)| = |F(-u)| \tag{2.6.36}$$

พิจารณารูป 2.1 จากคุณสมบัติคาบเวลา (periodicity) พบว่า  $F(u)$  มีคาบเวลายาว  $N$  จากคุณสมบัติการสมมาตร พบว่า ขนาดของทรานส์ฟอร์มประมาณได้ที่จุดกำเนิด ขนาดของทรานส์ฟอร์มจาก  $(N/2J+1)$  ถึง  $(N-1)$  เป็นค่าของภาพในครึ่งคาบเวลาซ้ายของจุดกำเนิด เนื่องจากฟูเรียร์ทรานส์ฟอร์มของ ถูกกำหนดในช่วง  $[0, N-1]$  ในช่วงนี้ จะเป็นแต่ละครึ่งของคาบซิกเนต ดังนั้นจึงเลื่อนจุดกำเนิดไปยังจุด  $u = N/2$  จากรูปที่ 2.1 ซึ่งทำได้โดยการคูณ  $f(x)$  ด้วย  $(-1)^x$  ดังที่กล่าวมาข้างต้น



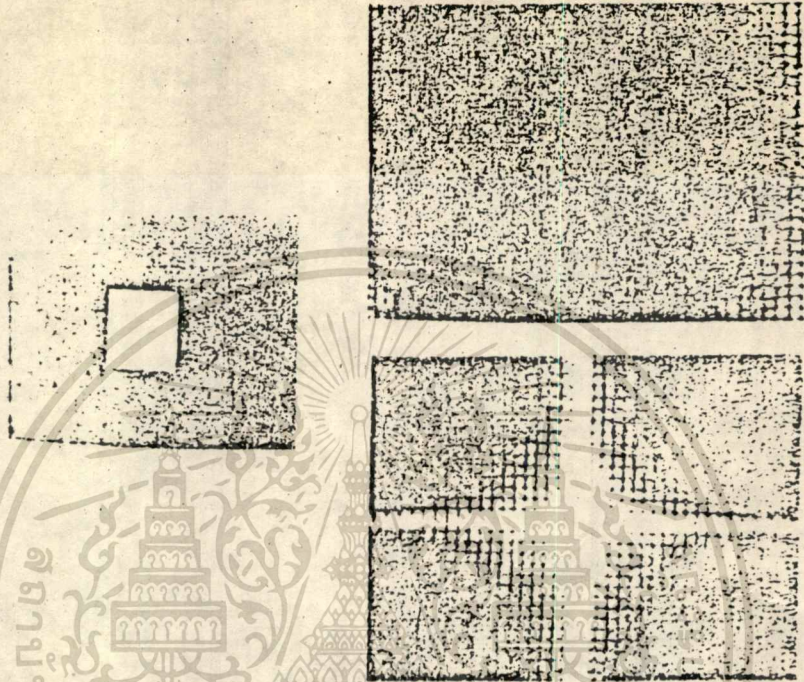
รูป 2.15 สรุปลักษณะคาบของฟูเรียร์ทรานส์ฟอร์ม

a) ฟูเรียร์สเปกตรัม แสดงครึ่งคาบเวลา ซิกเนตในช่วง  $[0, N-1]$

b) เลื่อนสเปกตรัม เมื่อดูเต็มคาบเวลา ในช่วงที่เหนือกัน

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น

สำหรับฟูรีเยร์ทรานส์ฟอร์ม 2 มิติ ที่ใช้ข้อพิจารณาเดียวกัน จะเห็นว่าเราจะแปลความหมายได้ยาก ถ้าจุดกำเนิดของการทรานส์ฟอร์มไม่ได้เลื่อนมาที่จุดความถี่



รูป 2.16 a) รูปตัวอย่างง่าย ๆ  
 b) ฟูรีเยร์สเปกตรัมที่ไม่มีกการเลื่อน  
 c) ฟูรีเยร์สเปกตรัมที่เลื่อน ไปยังจุดกึ่งกลางภาพความถี่

2.6.3.4 การหมุน (Rotation)

ใช้โพลาร์โคออร์ดิเนต (polar coordinate) กำหนดให้

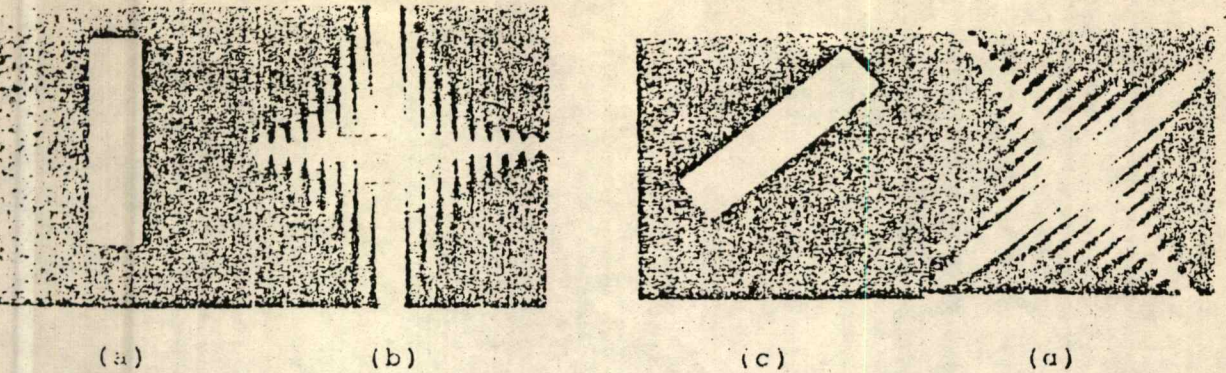
$$x = r \cos \theta \quad y = r \sin \theta \quad u = w \cos \phi \quad v = w \sin \phi$$

ดังนั้น  $f(x,y)$  และ  $F(u,v)$  กลายเป็น  $f(r,\phi)$  และ  $F(w,\phi)$  ตามลำดับ โดยการแทนค่าโดยตรง ในคู่สมการฟูรีเยร์ทรานส์ฟอร์ม แบบต่อเนื่องหรือไม่ต่อเนื่อง อย่างไรก็ตามอย่างหนึ่ง จะได้ว่า

$$f(r, \theta + \theta_0) \iff F(w, \phi + \phi_0) \tag{2.6.37}$$

หรืออีกนัยหนึ่งก็คือ ถ้า  $f(x,y)$  หมุนไปเป็นมุม แล้ว  $F(u,v)$  ก็จะมีมุม

เอกลิปไปด้วยมุมเดียวกัน งานในทำนองเดียวกันเกิดการหมุน  $F(u,v)$  จะทำให้  $f(x,y)$  ี่ถูกหมุนด้วยมุมการค้ำ  
 ไม่สามารถที่จะห้กลับคืนทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้  
 แท้กันดังรูป 2.17



รูป 2.17 คุณสมบัติการหมุนของฟูเรียร์ทรานส์ฟอร์ม

- a) ภาพตัวอย่างง่าย ๆ
- b) สเปกตรัม
- c) หมุนภาพ
- d) ผลในสเปกตรัม

2.6.3.5 ความไม่ต่อเนื่องและสเกลิ่ง (Distributivity and Scaling)

จากนิยามคุณสมบัติการของการทรานส์ฟอร์มแบบต่อเนื่องหรือไม่ต่อเนื่อง คือ

$$\{f_1(x,y) + f_2(x,y)\} = \{f_1(x,y)\} + \{f_2(x,y)\} \quad (2.6.38)$$

แต่

$$\{f_1(x,y) * f_2(x,y)\} = \{f_1(x,y)\} * \{f_2(x,y)\} \quad (2.6.39)$$

คุณสมบัติการกระจายทำได้ในการบวก แต่ในการคูณทำไม่ได้ และเช่นเดียวกัน

พบว่า สำหรับสเกลลาร์  $a, b$  ใดๆ

$$af(x,y) \iff aF(u,v)$$

และ

$$f(ax,by) \iff |1/ab|F(u/a,x/b) \quad (2.6.40)$$

2.6.3.6 ค่าเฉลี่ย (Average value)

นิยามค่าเฉลี่ยของฟังก์ชันไม่ต่อเนื่อง 2 มิติ ที่ใช้โดยทั่วไป คือ

$$\bar{f}(x,y) = 1/N^2 \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \quad (2.6.41)$$

แทน  $u = v = 0$  ในสมการ (2.6.17) จะได้

$$F(0,0) = 1/N \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \quad (2.6.42)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในมหาวิทยาลัยเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆก็ตามที่อ้างว่าคัดลอก (x,y) หรือหาความสัมพันธ์กับฟูเรียร์ทรานส์ฟอร์มของ f(x,y) ด้วย

สมการ

$$\bar{f}(x,y) = 1/N (F(0,0)) \tag{2.6.43}$$

2.6.3.7 ลاپลาเซียน (Laplacian)

ค่าลาปลาเซียนของฟังก์ชัน 2 ตัวแปร  $f(x,y)$  ถูกกำหนดโดย

$$\nabla^2 f(x,y) = \left( \frac{\partial^2 f}{\partial x^2} \right) + \left( \frac{\partial^2 f}{\partial y^2} \right) \tag{2.6.44}$$

และจากนิยามฟูเรียร์ทรานส์ฟอร์ม 2 มิติจะได้

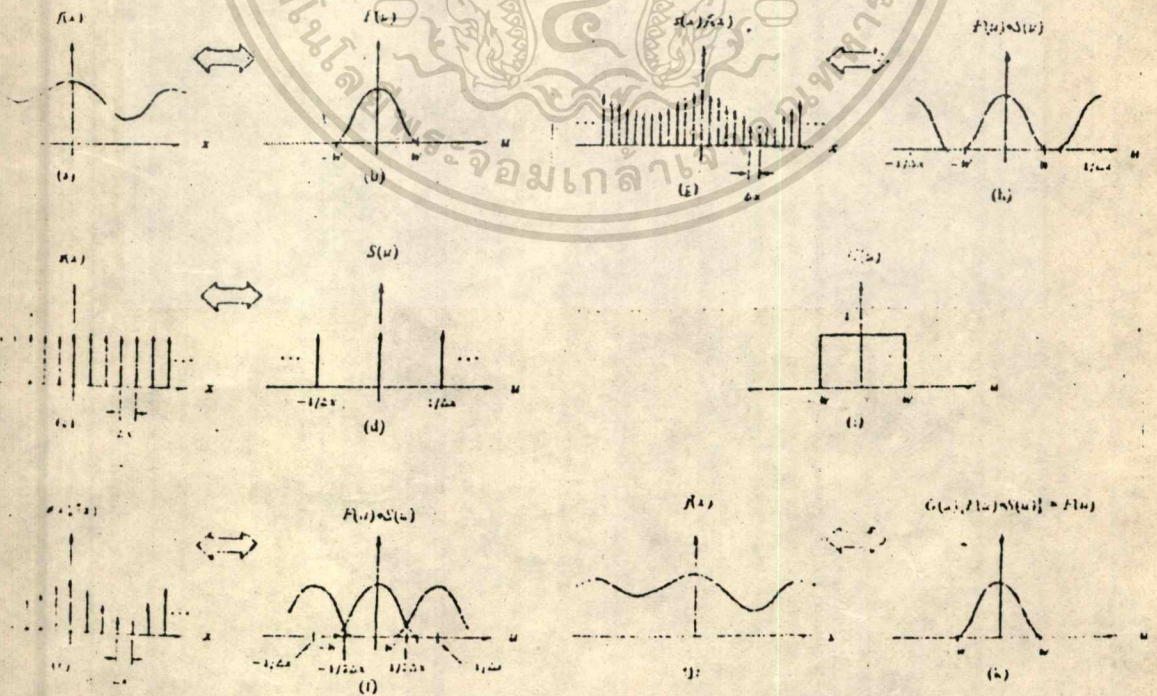
$$\mathcal{F} \{ \nabla^2 f(x,y) \} \iff -(2\pi)^2 (u^2 + v^2) F(u,v) \tag{2.6.45}$$

การทำลาปลาเซียนมีประโยชน์ในการลากขอบเขตของภาพ

2.6.4 การสุ่ม (Sampling)

ปัญหาการเลือกจำนวนการสุ่ม เพื่อไม่ให้ข้อมูลหาย จะศึกษาถึงกรณี 1 มิติก่อน

พิจารณารูป 2.18



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้รูป 2.18 นี้แสดงทุกสิ่งที่มีชื่อของกิจการสุ่มองเอกสารทุกครั้งที่มีการนำไปใช้

รูป (a) ฟังก์ชันต่อเนื่องจาก  $-\infty$  ถึง  $\infty$  มีฟูเรียร์ทรานส์ฟอร์ม  $f(x)$  ซึ่งค่าที่อยู่นอกช่วง  $[-w, w]$  เป็นศูนย์หมด เรียกฟังก์ชันที่ทรานส์ฟอร์มแล้วได้ลักษณะเช่นนี้ว่าแบบเลอิมิตต์ฟังก์ชัน (band - limited function)

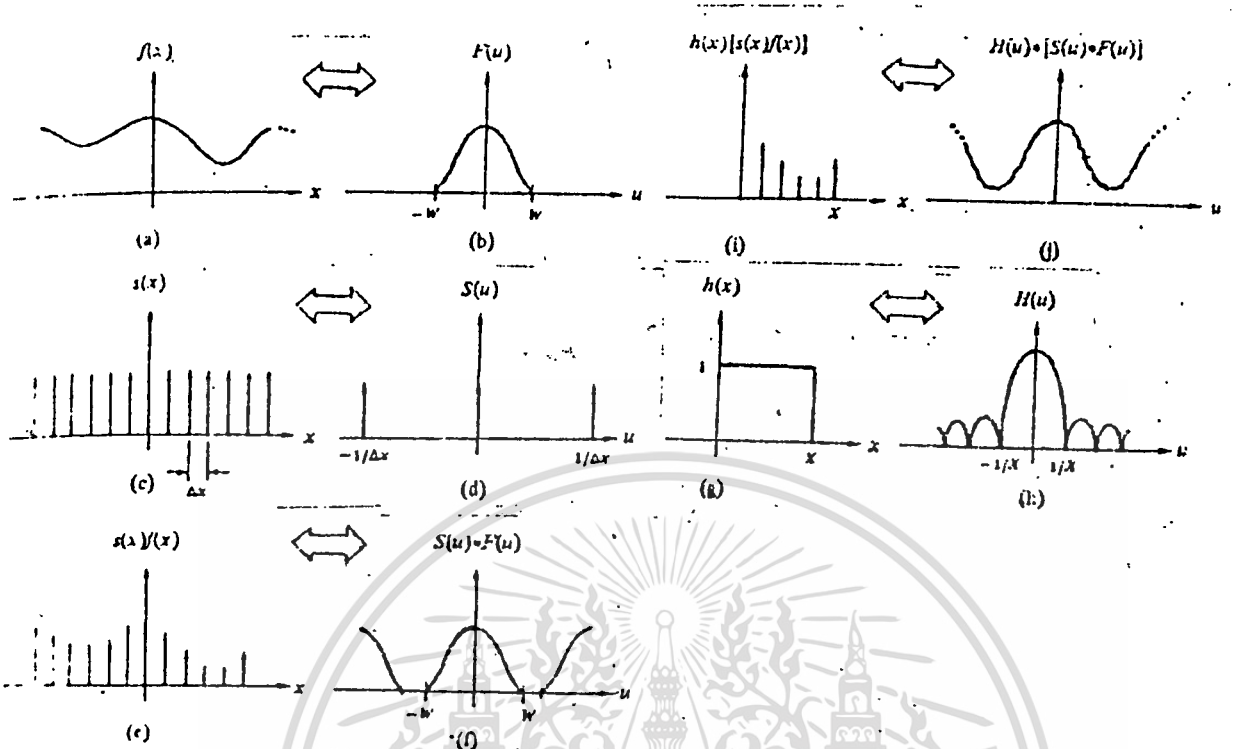
การสุ่ม (sampling) ฟังก์ชัน  $f(x)$  คือการคูณฟังก์ชันด้วยฟังก์ชัน  $S(x)$  ซึ่งประกอบด้วยแนวของอิมพัลส์ (impulse) ห่างกัน  $\Delta x$  จากทฤษฎีของคอนโวลูชัน การคูณกันในโดเมน  $x$  สมัย การคอนโวลูชันในโดเมนความถี่ เราได้ฟูเรียร์ทรานส์ฟอร์มในรูป  $(f)$  สำหรับผลคูณ  $S(x)$   $f(x)$  จะเห็นว่าการทรานส์ฟอร์มเป็นคาบเวลา ที่มีคาบเวลา  $1/\Delta x$  และแต่ละค่าที่ซ้ำกันของ  $F(u)$  ซ้อนกันได้ ตัวอย่างเช่นในคาบเวลาแรก จุดกึ่งกลางของบริเวณที่ซ้อนกันเกิดที่  $u = 1/2\Delta x$  ถ้าค่า  $1/2\Delta x$  น้อยกว่า  $w$  เพื่อหลีกเลี่ยงปัญหานี้ เราจึงเลือกช่วงการสุ่ม  $\Delta x$  ที่  $1/2\Delta x \geq w$  หรือ

$$\Delta x \leq 1/2w \quad (2.6.46)$$

ผลของการลด  $x$  แสดงในรูป 2.19 (g) และ (h) ถ้าเราคูณทรานส์ฟอร์มด้วยฟังก์ชัน

$$G(u) = \begin{cases} 1 & -w \leq u \leq w \\ 0 & \text{ที่อื่น ๆ} \end{cases} \quad (2.6.47)$$

จะทำให้  $F(u)$  แยกจากกันโดยเด็ดขาด ดังรูป 2.18 (k) ซึ่งเมื่อมาอินเวิร์สฟูเรียร์ทรานส์ฟอร์ม ก็จะได้ฟังก์ชันต่อเนื่องเดิมกลับมา ผลที่ได้แบบเลอิมิตต์มาจากการสุ่มด้วยระยะ  $\Delta x$  ที่สอดคล้องกัน (2.6.46) ก็คือหลักการของทฤษฎีการสุ่มวิลเทเกอร์ชานอน (Whittaker Shannon sampling theorem) ซึ่งมีข้อจำกัดคือ ข้อมูลในโดเมนความถี่อยู่ในช่วง  $[-w, w]$  ถ้าสมการ (2.6.46) ไม่เป็นจริง การทรานส์ฟอร์มในช่วงนี้จะเสียไปเนื่องจากการซ้อนกันของค่าข้างเคียง เหตุการณ์เช่นนี้ เรียกว่า อเลซิง (aliasing)



รูป 2.19 แสดงการสุมที่จำกัด

เมื่อใช้กับฟังก์ชันต่อเนื่องในโดเมน  $x$  ซึ่งเสมือนมีช่วงการสุม ได้จำนวนมาก การเลือกบริเวณการสุมในบริเวณจำกัดที่หนึ่ง จึงเป็นกรณีในทางปฏิบัติที่สำคัญ ในสภาวะเช่นนี้ แสดงให้เห็นในรูป 2.19 ส่วนของ (a) ถึง (f) เหมือนรูป 2.18 เว้นแต่มีช่วงการสุมที่ สอดคล้องกับพฤติกรรมการสุม ซึ่งไม่เกิดต่อเนื่อง ช่วงการสุม  $[0, x]$  ถูกแทนด้วยคณิตศาสตร์ โดยการคูณผลของการสุม ในรูป 2.19 (e)

รูป 2.19(e) ด้วยฟังก์ชัน

$$h(x) = \begin{cases} 1 & 0 \leq x \leq x \\ 0 & \text{นอกเหนือจากนั้น} \end{cases} \quad (2.6.48)$$

$h(x)$  กับฟูเรียร์ทรานส์ฟอร์ม ดังรูป 2.16 (q) และ (h) ตามลำดับ ผลการคูณจะได้ รูป 2.16 (i) และ (j). จะพบว่าผลสุดท้ายในโดเมนความถี่ได้จาก คอนโวลว (convolve) ฟังก์ชัน  $S(u) * F(u)$  ด้วย  $H(u)$  ซึ่งเป็นฟูเรียร์ทรานส์ฟอร์มของ  $h(x)$

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ เนื่องจาก  $h(u)$  ใช้ มีตัวประกอบ ขาดความถี่ที่มีค่าไม่แน่นอน การทำคอนโวลวกับ ฟังก์ชันของฟังก์ชันเหล่านี้ ทำให้เกิดความไม่แน่นอนในโดเมนความถี่ขึ้น แทนฟังก์ชันที่ถูกสุมการนำไปใช้

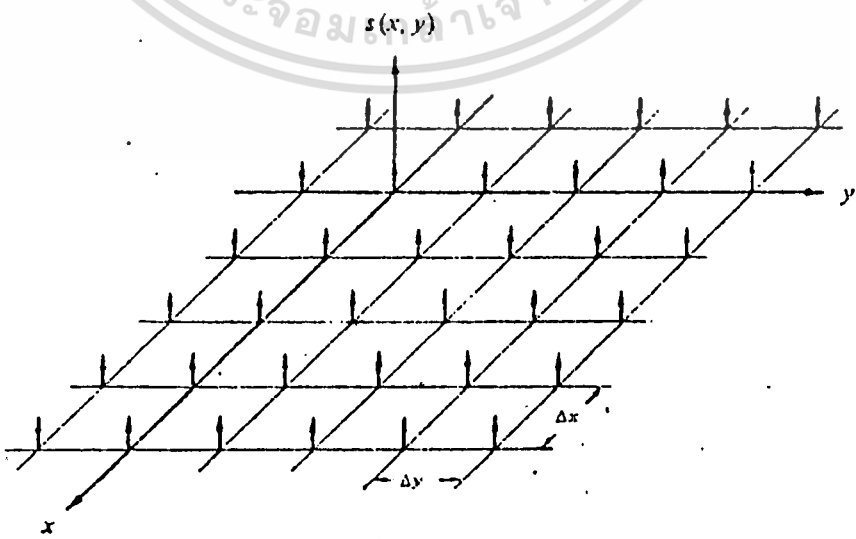
และจำกัดในหน้าที่หนึ่งโดย  $h(x)$  ดังแสดงในรูป 2.19 (j) ดังนั้นแม้ว่าการสุมที่แยกกันสอดคล้องกับทฤษฎีการสุม การที่จะได้ฟังก์ชันที่ถูกสุม

ในช่วงจำกัดของโดเมน  $x$  กลับคืนมาโดยสมบูรณ์นั้นเป็นไปได้ กล่าวคือ ภายใต้เงื่อนไขข้างต้น เราไม่มีทางแยกฟูเรียร์ทรานส์ฟอร์มเต็มออกมา เว้นเสียแต่ว่า  $f(x)$  เป็นแบบลิมิต และเป็นฟังก์ชันคาบเวลามีคาบเวลาเท่ากับ  $x$  ซึ่งผลของ  $h(x)$  จะหายไป และจะได้  $f(x)$  เพื่อสอดคล้องตามทฤษฎีการสุม สิ่งสำคัญคือ ฟังก์ชันที่ได้ยังคงอยู่  $(-\infty$  ถึง  $\infty)$  และไม่เป็นศูนย์นอกช่วงที่  $h(x)$  เป็นศูนย์ ข้อมูลดังกล่าวทำให้ได้ข้อสรุปสำคัญว่า ไม่มีฟังก์ชัน  $f(x)$  ในช่วงจำกัดใดเป็นแบบลิมิตได้ ในทางกลับกัน คือฟังก์ชันที่เป็นแบบลิมิตได้ต้องอยู่ในช่วง  $-\infty$  ถึง  $\infty$  ในโดเมน  $x$  นี้คือผลในทางปฏิบัติที่สำคัญ เพราะทำให้เราได้ข้อจำกัดหลักที่จะใช้ฟังก์ชันเดิจิตอล

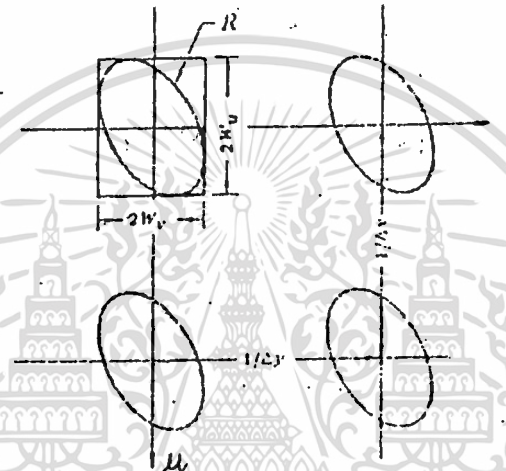
สำหรับฟังก์ชัน 2 มิติ ด้วยหลักการเดียวกัน การสุมฟังก์ชันต่าง ๆ เขียนในรูปคณิตศาสตร์โดยใช้อิมพัลส์ฟังก์ชัน (impulse function) 2 มิติ  $(x, y)$  ซึ่งนิยามโดย

$$\iint_{-\infty}^{\infty} f(x, y) \delta(x-x_0, y-y_0) dx dy = f(x_0, y_0) \quad (2.6.49)$$

ฟังก์ชันสุม 2 มิติประกอบด้วยแถวของอิมพัลส์ห่างกัน  $x$  ในแนวแกน  $x$  และห่างกัน  $\Delta y$  ในแนวแกน  $y$  ดังรูป 2.20



กำหนดให้  $f(x,y)$  เป็นฟังก์ชันหนึ่ง ซึ่ง  $x$  และ  $y$  ต่อเนื่อง ฟังก์ชันการสุ่ม ที่ได้อยู่ในรูปผลคูณ  $s(x,y) f(x,y)$  ในโดเมนความถี่ก็คือในรูปของการทำคอนโวลูชัน ของ  $S(u,v)$  และ  $F(u,v)$  เมื่อ  $s(u,v)$  เป็นอิมพัลส์ที่ห่างกัน  $1/\Delta x$  และ  $1/\Delta y$  ในทิศทางของ  $u$  และ  $v$  ตามลำดับ ถ้า  $f(x,y)$  เป็นแบนด์ลิมิต (อีกนัยหนึ่งค่าฟูเรียร์ทรานส์ฟอร์ม เป็นศูนย์นอกบริเวณที่หาค่าได้  $R$ ) ผลของการทำคอนโวลูชันดังกล่าวคือ กรณีในรูป 2.21 โดยฟังก์ชันที่ใช้เป็นคาบเวลา ใน 2 มิติ



รูป 2.21 การแทนตัวอย่างที่สุ่มใน 2 มิติของโดเมนความถี่ ใช้แบนด์ลิมิตฟังก์ชัน

ให้  $2w_u$  และ  $2w_v$  แทนความกว้างในทิศทางของ  $u$  และ  $v$  ตามลำดับ ของสี่เหลี่ยมผืนผ้าที่เล็กที่สุด และคลุมพื้นที่  $R$  ดังนั้น จากรูป 2.21 ถ้า  $1/\Delta x > 2w_u$  และ  $1/\Delta y > 2w_v$  (คือไม่เกิด aliasing) เราสามารถได้คาบเวลา หนึ่งที่สมบูรณ์

ถ้าเราคูณ  $S(u,v) * F(u,v)$  ด้วยฟังก์ชัน

$$G(u,v) = \begin{cases} 1 & \text{อยู่ในสี่เหลี่ยมผืนผ้าที่ครอบคลุม } R \\ 0 & \text{นอกเหนือจากนั้น} \end{cases} \quad (2.6.50)$$

อินเวิร์สฟูเรียร์ทรานส์ฟอร์มของ  $G(u,v) [S(u,v) * F(u,v)]$  คือ  $f(x,y)$  จากข้อมูลที่ผ่านมาทำให้เราทราบถึงทฤษฎีการสุ่ม ใน 2 มิติ ซึ่งกล่าวว่าฟังก์ชันแบนด์ลิมิต  $f(x,y)$  สามารถทำกลับมาได้โดยสมบูรณ์จากการสุ่มที่ห่างกัน ตามความสัมพันธ์ต่อไปนี้

$$\Delta x < 1/2w_u \quad (2.6.51)$$

เอกสารและเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกนี้  $\Delta y < 1/2w_v$  ลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่นำไปใช้ (2.6.52)

เมื่อ  $f(x,y)$  เป็น สเปเชียลลิมิต (spatial-limited) โดยใช้กรอบ (window) 2 มิติ  $h(x,y)$  เหมือนกัน ที่ใช้ในมิติเดียว ปัญหาในลักษณะเดียวกัน คือ ค่าทราานส์ฟอร์มของฟังก์ชันการสุ่มถูกทำให้เห็นไป เนื่องจากลักษณะทั่วไป ของสเปเชียลลิมิตของภาพดิจิทัล ทำให้ไม่ได้  $f(x,y)$  อย่างสมบูรณ์ และก็เช่นเดียวกับใน 1 มิติ คือ เว้นเสียแต่ว่า ใช้ฟังก์ชันคาบเวลา แต่ภาพที่สอดคล้อง เงื่อนไขหาได้ยากในทางปฏิบัติ



## 2.7 ดิสกรีตฟูเรียร์ทรานสฟอร์มและดิสกรีตฮาร์ตเลย์ทรานสฟอร์ม

(Discrete Fourier Transform and Discrete Hardley Transform)

ดิสกรีตฟูเรียร์ทรานสฟอร์มเป็นการส่งค่าของฟังก์ชันเวลา  $X(t)$  ซึ่งเป็นค่าเชิงซ้อน (Complex Sequence) ไปยังฟังก์ชันความถี่ ซึ่งเป็นค่าเชิงซ้อนเช่นกัน แต่สำหรับดิสกรีตฮาร์ตเลย์ทรานสฟอร์มนั้นเป็นการส่งค่าจริง ไปยังค่าจริง ดังนั้นการคำนวณของดิสกรีตฮาร์ตเลย์ทรานสฟอร์มนั้น จะไม่ยุ่งยากซับซ้อนเท่ากับการคำนวณของดิสกรีตฟูเรียร์ทรานสฟอร์ม จึงทำงานได้เร็วกว่า

### 2.7.1 รูปแบบทั่วไป

(Definition of Discrete Fourier Transform  
and Discrete Hardley Transform)

$$F(f) = 1/N \sum_{t=0}^{N-1} [X(t)e^{-j2\pi ft/n}] \quad (2.7.1)$$

$$X(t) = \sum_{f=0}^{N-1} [X(f)e^{j2\pi ft/n}] \quad (2.7.2)$$

$$e = \cos(2\pi ft/n) - j\sin(2\pi ft/n)$$

ดิสกรีตฮาร์ตเลย์ทรานสฟอร์ม

$$H(f) = 1/N \sum_{t=0}^{N-1} [X(t)\cos(2\pi ft/n)] \quad (2.7.3)$$

$$X(t) = \sum_{f=0}^{N-1} [H(f)\cos(2\pi ft/n)] \quad (2.7.4)$$

$$\cos(2\pi ft/n) = \cos(2\pi ft/n) + \sin(2\pi ft/n)$$

จากสมการ 2.7.2 และ 2.7.4 เป็นอินเวอร์สทรานสฟอร์มของฟาสท์ฟูเรียร์ทรานสฟอร์มและฟาสท์ฮาร์ตเลย์ทรานสฟอร์มตามลำดับ จะเห็นว่ารูปสมการของดิสกรีตฟูเรียร์ทรานสฟอร์มและดิสกรีตฮาร์ตเลย์ทรานสฟอร์มตามลำดับ เพียงแต่เราแทน  $e^{-j2\pi ft/n} =$

$\cos(2\pi ft/n) - j\sin(2\pi ft/n)$  ของฟูเรียร์ทรานสฟอร์มด้วย  $\cos(2\pi ft/n)$  ก็จะได้

เอกสารนี้เป็นเอกสารที่สวชนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ได้รูปสมการของฮาร์ตเลย์ทรานสฟอร์ม

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.7.2 ความสัมพันธ์ระหว่างดิסקรีตฟูเรียร์ทรานสฟอร์มและดิסקรีตอาร์ดเลย์ทรานสฟอร์ม

จากรูปสมการของดิסקรีตฟูเรียร์ทรานสฟอร์ม และดิסקรีตอาร์ดเลย์ทรานสฟอร์ม มีลักษณะคล้ายกัน ทำให้เราสามารถเขียนสมการของอาร์ดเลย์ทรานสฟอร์ม ให้อยู่ในรูปของฟูเรียร์ทรานสฟอร์มได้จากสมการ 2.7.4 ดังนี้

$$H(f) = E(f) + O(f) \quad \text{โดยที่ } E(f) \text{ (even part) ซึ่งมีความสมบัติ } E(-f) = E(f)$$

$$O(f) \text{ (odd part) ซึ่งมีความสมบัติ } O(-f) = -O(f)$$

$$E(f) = [H(f) + H(N-f)]/2 \quad O(f) = [H(f) - H(N-f)]/2$$

ส่วนดิסקรีตฟูเรียร์ทรานสฟอร์ม เราสามารถเขียนสมการใหม่ได้เป็น

$$F(f) = E(f) - iO(f) \quad F \text{ (real part)} = E(f)$$

$$F \text{ (imaginary part)} = -O(f)$$

ดังนั้น ความสัมพันธ์ระหว่างอาร์ดเลย์ทรานสฟอร์ม กับฟูเรียร์ทรานสฟอร์ม คือ

$$H(f) = F_{re}(f) - F_{im}(f)$$

$$F_{re} = [H(f) + H(N-f)]/2 \quad F_{im} = [H(N-f) - H(f)]/2$$

## 2.8 ฟาสท์ฟูเรียร์ทรานสฟอร์มและฟาสท์อาร์ดเลย์ทรานสฟอร์ม

ในการประมวลผลสัญญาณดิจิทัล เพื่อนำไปใช้ประยุกต์กับงานในสาขาวิชาต่างๆ โดยทั่วไป มักจะใช้การคำนวณดิסקรีตฟูเรียร์ทรานสฟอร์มในการแปลงสัญญาณจากโดเมน (time domain) ไปเฟรเควนซีโดเมน (frequency domain) ซึ่งส่วนใหญ่ใช้อัลกอริทึม ฟาสท์ฟูเรียร์ทรานสฟอร์ม ที่ช่วยลดการคำนวณทางคณิตศาสตร์ลงได้ แต่ในปัจจุบันได้มีการพัฒนาดิסקรีตฟูเรียร์ทรานสฟอร์ม เป็นดิסקรีตอาร์ดเลย์ทรานสฟอร์ม โดยใช้อัลกอริทึมฟาสท์อาร์ดเลย์ทรานสฟอร์มที่คำนวณได้เท่ากับ หรือเร็วกว่าฟาสท์ฟูเรียร์ทรานสฟอร์ม

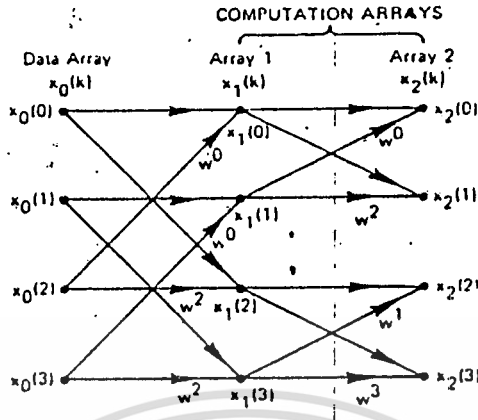
### 2.8.1 ฟาสท์ฟูเรียร์ทรานสฟอร์ม

จากสมการของดิסקรีตฟูเรียร์ทรานสฟอร์ม

$$X(n) = (1/N) \sum_{s=0}^{N-1} x(s) w^{ns} \quad n = 0, 1, 2, \dots, N-1 \quad (2.8.1)$$

โดยที่  $w = e^{-j2\pi/n}$ ,  $j = \sqrt{-1}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ถ้าให้ N = 4 สามารถเขียนแผนผังการทำงานได้ดังรูปที่ 2.22



รูป 2.22 แสดงแผนผังการทำงานของฟาสต์ฟูเรียร์ทรานสฟอร์ม โดย  $N=4$   $r = 2$  ( $N = 2^r$ )

จากสมการ 2.7.1 สมการที่ในการคำนวณของอัลกอริทึมฟาสต์ฟูเรียร์ทรานสฟอร์ม

คือ

$$x_1(k) = x_{l-1}(k) + W x_{l-1}(k+N/2) \quad (2.8.1)$$

$$x_1(k+N/2) = x_{l-1}(k) - W x_{l-1}(k+N/2) \quad (2.8.2)$$

โดย  $l$  เป็นลำดับที่ของอาร์เรย์ (array) และ  $p$  หาได้โดย

- เขียน  $k$  ในรูปของเลขไบนารีจำนวน  $r$  บิต
- เลื่อนเลขไบนารีไปทางขวา  $r-1$  บิต และเติมตำแหน่งทางซ้ายด้วย 0
- กลับบิต (bit reversal)

แล้วนำผลจากสมการ 2.8.1 และ 2.8.2 มากลับบิตและหารด้วย  $N$  ก็จะได้ผล

ลัพธ์เป็น  $X(n)$

### 2.8.2 ฟาสต์อาร์ตเลย์ทราสฟอร์ม

มีอัลกอริทึมที่สามารถลดเวลาการคำนวณได้ เมื่อเทียบกับการคำนวณดิสกรีตอาร์ตเลย์ทราสฟอร์ม ตามรูปแบบสมการที่กำหนดไว้ อัลกอริทึมนั้นคือ ฟาสต์อาร์ตเลย์ทราสฟอร์ม

ซึ่งมีลักษณะคล้ายคลึงกับฟาสต์ฟูเรียร์ทรานสฟอร์ม

เอกสารนี้เป็นลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งนี้เราสามารถแสดงอัลกอริทึมได้ โดยรูป 2.23 และตารางสัจจริงนี้ ครั้งที่มีการนำไปใช้

ตาราง 2.8.2.1 สมการแสดงความสัมพันธ์ในการคำนวณของกันต่าง ๆ โดยมี L เป็นตัวบอกระดับ

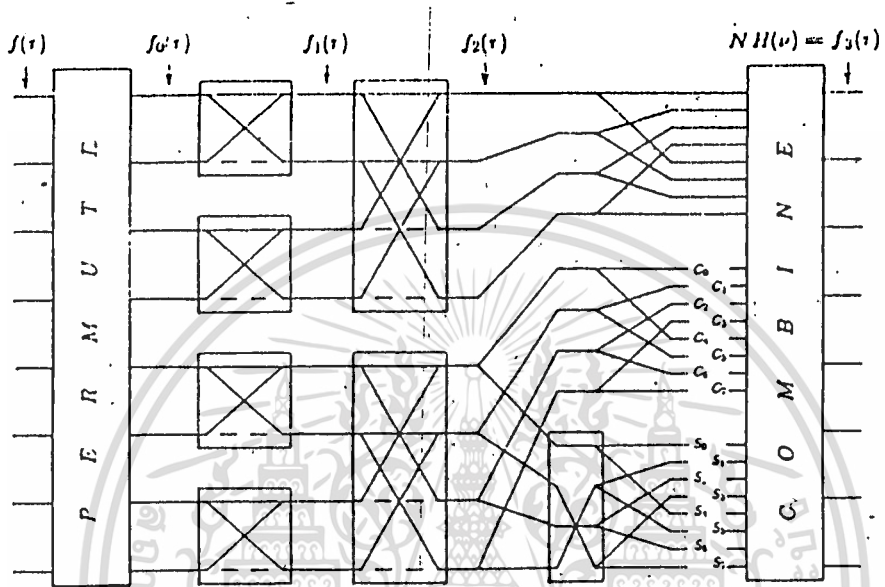
Data	Permute	Level 1	Level 2	Level 3	Level 4	16 x DHT
$f(0,0)$	$f(0,0) - f(0,0)$	$f(1,0) - f(0,0) + f(0,1)$	$f(2,0) - f(1,0) + f(1,2)C_0 + f(1,2)S_0$	$f(3,0) - f(2,0) + f(2,4)C_0 + f(2,4)S_0$	$f(4,0) - f(3,0) + f(3,8)C_0 + f(3,8)S_0$	
$f(0,1)$	$f(0,1) - f(0,0)$	$f(1,1) - f(0,1) - f(0,1)$	$f(2,1) - f(1,1) + f(1,3)C_1 + f(1,3)S_1$	$f(3,1) - f(2,1) + f(2,5)C_1 + f(2,5)S_1$	$f(4,1) - f(3,1) + f(3,9)C_1 + f(3,9)S_1$	
$f(0,2)$	$f(0,2) - f(0,1)$	$f(1,2) - f(0,2) + f(0,3)$	$f(2,2) - f(1,2) + f(1,2)C_2 + f(1,2)S_2$	$f(3,2) - f(2,2) + f(2,6)C_2 + f(2,6)S_2$	$f(4,2) - f(3,2) + f(3,10)C_2 + f(3,10)S_2$	
$f(0,3)$	$f(0,3) - f(0,2)$	$f(1,3) - f(0,3) - f(0,3)$	$f(2,3) - f(1,3) + f(1,3)C_3 + f(1,3)S_3$	$f(3,3) - f(2,3) + f(2,7)C_3 + f(2,7)S_3$	$f(4,3) - f(3,3) + f(3,11)C_3 + f(3,11)S_3$	
$f(0,4)$	$f(0,4) - f(0,3)$	$f(1,4) - f(0,4) + f(0,5)$	$f(2,4) - f(1,4) + f(1,6)C_4 + f(1,6)S_4$	$f(3,4) - f(2,4) + f(2,4)C_4 + f(2,4)S_4$	$f(4,4) - f(3,4) + f(3,12)C_4 + f(3,12)S_4$	
$f(0,5)$	$f(0,5) - f(0,4)$	$f(1,5) - f(0,5) - f(0,5)$	$f(2,5) - f(1,5) + f(1,7)C_5 + f(1,7)S_5$	$f(3,5) - f(2,5) + f(2,5)C_5 + f(2,5)S_5$	$f(4,5) - f(3,5) + f(3,13)C_5 + f(3,13)S_5$	
$f(0,6)$	$f(0,6) - f(0,5)$	$f(1,6) - f(0,6) + f(0,7)$	$f(2,6) - f(1,6) + f(1,6)C_6 + f(1,6)S_6$	$f(3,6) - f(2,6) + f(2,6)C_6 + f(2,6)S_6$	$f(4,6) - f(3,6) + f(3,14)C_6 + f(3,14)S_6$	
$f(0,7)$	$f(0,7) - f(0,6)$	$f(1,7) - f(0,7) - f(0,7)$	$f(2,7) - f(1,7) + f(1,7)C_7 + f(1,7)S_7$	$f(3,7) - f(2,7) + f(2,7)C_7 + f(2,7)S_7$	$f(4,7) - f(3,7) + f(3,15)C_7 + f(3,15)S_7$	
$f(0,8)$	$f(0,8) - f(0,7)$	$f(1,8) - f(0,8) + f(0,9)$	$f(2,8) - f(1,8) + f(1,10)C_8 + f(1,10)S_8$	$f(3,8) - f(2,8) + f(2,8)C_8 + f(2,8)S_8$	$f(4,8) - f(3,8) + f(3,16)C_8 + f(3,16)S_8$	
$f(0,9)$	$f(0,9) - f(0,8)$	$f(1,9) - f(0,9) - f(0,9)$	$f(2,9) - f(1,9) + f(1,11)C_9 + f(1,11)S_9$	$f(3,9) - f(2,9) + f(2,9)C_9 + f(2,9)S_9$	$f(4,9) - f(3,9) + f(3,17)C_9 + f(3,17)S_9$	
$f(0,10)$	$f(0,10) - f(0,9)$	$f(1,10) - f(0,10) + f(0,11)$	$f(2,10) - f(1,10) + f(1,10)C_{10} + f(1,10)S_{10}$	$f(3,10) - f(2,10) + f(2,10)C_{10} + f(2,10)S_{10}$	$f(4,10) - f(3,10) + f(3,18)C_{10} + f(3,18)S_{10}$	
$f(0,11)$	$f(0,11) - f(0,10)$	$f(1,11) - f(0,11) - f(0,11)$	$f(2,11) - f(1,11) + f(1,11)C_{11} + f(1,11)S_{11}$	$f(3,11) - f(2,11) + f(2,11)C_{11} + f(2,11)S_{11}$	$f(4,11) - f(3,11) + f(3,19)C_{11} + f(3,19)S_{11}$	
$f(0,12)$	$f(0,12) - f(0,11)$	$f(1,12) - f(0,12) + f(0,13)$	$f(2,12) - f(1,12) + f(1,14)C_{12} + f(1,14)S_{12}$	$f(3,12) - f(2,12) + f(2,12)C_{12} + f(2,12)S_{12}$	$f(4,12) - f(3,12) + f(3,20)C_{12} + f(3,20)S_{12}$	
$f(0,13)$	$f(0,13) - f(0,12)$	$f(1,13) - f(0,13) - f(0,13)$	$f(2,13) - f(1,13) + f(1,15)C_{13} + f(1,15)S_{13}$	$f(3,13) - f(2,13) + f(2,13)C_{13} + f(2,13)S_{13}$	$f(4,13) - f(3,13) + f(3,21)C_{13} + f(3,21)S_{13}$	
$f(0,14)$	$f(0,14) - f(0,13)$	$f(1,14) - f(0,14) + f(0,15)$	$f(2,14) - f(1,14) + f(1,14)C_{14} - f(1,14)S_{14}$	$f(3,14) - f(2,14) + f(2,14)C_{14} + f(2,14)S_{14}$	$f(4,14) - f(3,14) + f(3,22)C_{14} + f(3,22)S_{14}$	
$f(0,15)$	$f(0,15) - f(0,14)$	$f(1,15) - f(0,15) - f(0,15)$	$f(2,15) - f(1,15) + f(1,15)C_{15} + f(1,15)S_{15}$	$f(3,15) - f(2,15) + f(2,15)C_{15} + f(2,15)S_{15}$	$f(4,15) - f(3,15) + f(3,23)C_{15} + f(3,23)S_{15}$	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่ควรนำออกเผยแพร่โดยไม่ได้รับอนุญาตจากเจ้าของลิขสิทธิ์

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 2.8.2.2 แสดงความสัมพันธ์ของการคำนวณ เมื่อแทนค่าโดยขายน และ ไซ้ ในตารางที่ 2.8.2.1 ซึ่งมีหลายค่าที่เริ่ม 0, 1, -1

Data	Permute	Level 1	Level 2	Level 3
$F(0,0)$	$F(0,0) - F(0,0)$	$F(1,0) - F(0,0) + F(0,1)$	$F(2,0) - F(1,0) + F(1,2)$	$F(3,0) - F(2,0) + F(2,4)$
$F(0,1)$	$F(0,1) - F(0,8)$	$F(1,1) - F(0,0) - F(0,1)$	$F(2,1) - F(1,1) + F(1,3)$	$F(3,1) - F(2,1) + F(2,5) + F(2,7)$
$F(0,2)$	$F(0,2) - F(0,4)$	$F(1,2) - F(0,2) + F(0,3)$	$F(2,2) - F(1,0) - F(1,2)$	$F(3,2) - F(2,2) + F(2,6)$
$F(0,3)$	$F(0,3) - F(0,12)$	$F(1,3) - F(0,2) - F(0,3)$	$F(2,3) - F(1,1) - F(1,3)$	$F(3,3) - F(2,3) - F(2,7) + F(2,5)$
$F(0,4)$	$F(0,4) - F(0,2)$	$F(1,4) - F(0,4) + F(0,5)$	$F(2,4) - F(1,4) + F(1,6)$	$F(3,4) - F(2,4) - F(2,4)$
$F(0,5)$	$F(0,5) - F(0,10)$	$F(1,5) - F(0,4) - F(0,5)$	$F(2,5) - F(1,5) + F(1,7)$	$F(3,5) - F(2,5) - F(2,5)$
$F(0,6)$	$F(0,6) - F(0,6)$	$F(1,6) - F(0,6) + F(0,7)$	$F(2,6) - F(1,4) - F(1,6)$	$F(3,6) - F(2,6) - F(2,6)$
$F(0,7)$	$F(0,7) - F(0,14)$	$F(1,7) - F(0,6) - F(0,7)$	$F(2,7) - F(1,5) - F(1,7)$	$F(3,7) - F(2,7) + F(2,7) - F(2,5)$
$F(0,8)$	$F(0,8) - F(0,1)$	$F(1,8) - F(0,8) + F(0,9)$	$F(2,8) - F(1,8) + F(1,10)$	$F(3,8) - F(2,8) + F(2,12)$
$F(0,9)$	$F(0,9) - F(0,9)$	$F(1,9) - F(0,8) - F(0,9)$	$F(2,9) - F(1,9) + F(1,11)$	$F(3,9) - F(2,9) + F(2,13) + F(2,15)$
$F(0,10)$	$F(0,10) - F(0,5)$	$F(1,10) - F(0,10) + F(0,11)$	$F(2,10) - F(1,8) - F(1,10)$	$F(3,10) - F(2,10) + F(2,14)$
$F(0,11)$	$F(0,11) - F(0,13)$	$F(1,11) - F(0,10) - F(0,11)$	$F(2,11) - F(1,9) - F(1,11)$	$F(3,11) - F(2,11) - F(2,15) + F(2,13)$
$F(0,12)$	$F(0,12) - F(0,3)$	$F(1,12) - F(0,12) + F(0,13)$	$F(2,12) - F(1,12) + F(1,14)$	$F(3,12) - F(2,8) - F(2,12)$
$F(0,13)$	$F(0,13) - F(0,11)$	$F(1,13) - F(0,12) - F(0,13)$	$F(2,13) - F(1,13) + F(1,15)$	$F(3,13) - F(2,9) - F(2,13) + F(2,15)$
$F(0,14)$	$F(0,14) - F(0,7)$	$F(1,14) - F(0,14) + F(0,15)$	$F(2,14) - F(1,12) - F(1,14)$	$F(3,14) - F(2,10) - F(2,14)$
$F(0,15)$	$F(0,15) - F(0,15)$	$F(1,15) - F(0,14) - F(0,15)$	$F(2,15) - F(1,13) - F(1,15)$	$F(3,15) - F(2,11) + F(2,15) - F(2,13)$



รูป 2.23 โฟลว์ไดอะแกรม (Flow diagram) ของคิสิกวิทอาร์ดเลย์ทธานสฟอร์ม ซึ่งมี  $N = 8, P=3$

$\tau$	$f(\tau)$	$\pi$	$f_0(\tau)$	$f_1(\tau)$	$f_2(\tau)$	$f_3(\tau)$	$H(\tau)$	$\nu$
0	1	1	1	6	16	36	4.5	0
1	2	3	5	-4	-8	-13.6	-1.7	1
2	3	5	3	10	-4	-8	-1	2
3	4	7	7	-4	0	-5.6	-0.7	3
4	5	2	2	8	20	-4	-0.5	4
5	6	4	6	-4	-8	-2.4	-0.3	5
6	7	6	4	12	-4	0	0	6
7	8	8	8	-4	0	5.6	0.7	7

ตาราง 2.8.2.3 ตัวอย่างของการคำนวณเฟาสท์อาร์ดเลย์ทธานสฟอร์ม ซึ่งมี  $N = 8, P=3$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมการ 2.7.4 ได้แสดงว่าสูตรดีคอมโพสิชัน (decomposition) ทั่วไป คือ

$$H(f) = H_2(f)\cos(2\pi f/N) + H(N-f)\sin(2\pi f/N) + H_1(f)$$

จำนวนข้อมูลที่ใช้จะต้องเป็น  $N = 2^P$   $P = 1, 2, 3, \dots$

จากไดอะแกรม เมื่อข้อมูลเข้ามาครั้งแรกนั้น จะต้องทำการเพอร์มูท (Permute) ซึ่งมีหลักการเกี่ยวกับการกลับทิศทางของฟาส์ฟูเรียร์ทรานสฟอร์ม เมื่อข้อมูลผ่านมายังระดับที่ 1 ซึ่งในระดับนี้ โคไซน์และไซน์เฟเดเตอร์ มีค่าเฉพาะ 0, 1 หรือ -1 เท่านั้น ดังนั้นไดอะแกรมจึงมีเส้น 2 ประเภท คือ เส้นทับ จะหมายถึง การนำข้อมูลมารวมกัน ส่วนเส้นประ หมายถึง จะนำข้อมูลมาลบกัน

ในระดับถัดไปนั้น จะมีค่าของโคไซน์และไซน์อยู่ด้วยโดย  $C_n = \cos(2\pi n/2^L)$ ,  $S_n = \sin(2\pi n/2^L)$  โดยที่ L แทนระดับที่ L

หลักการสำคัญของฟาส์ฟูเรียร์ทรานสฟอร์ม ก็คือ การใช้บัตเตอร์ฟลาย (butterfly) ซึ่งอาศัยคุณสมบัติของโคไซน์และไซน์ ช่วยลดการคำนวณลง ดังนี้ จุดที่ห่างกัน  $N/2$  จุด จะมีความมุมเปลี่ยนไป  $\pi$

$$\cos(\pi+t) = -\cos(t), \quad \sin(\pi+t) = -\sin(t)$$

จะได้  $H(k) = H(k) + H(k+N/2)\cos(2\pi k/N) + H(N-k)\sin(2\pi k/N)$

$$H(k+N/2) = H(k) - [H(k+N/2)\cos(2\pi k/N) + H(N-k)\sin(2\pi k/N)]$$

หรือ  $\cos(\pi-t) = -\cos(t), \quad \sin(\pi-t) = \sin(t)$

$$H(N-k) = H(N-k) + H(k+N/2)\sin(2\pi k/N) - H(N-k)\cos(2\pi k/N)$$

$$H(N/2-k) = H(N-k) - [H(k+N/2)\sin(2\pi k/N) - H(N-k)\cos(2\pi k/N)]$$

เพื่อให้มองเห็นชัดเจน จะขอยกตัวอย่างมาประกอบ ดังตาราง 2.8.2.2 และ 2.8.2.3

สมมุติ จำนวนข้อมูลทั้งหมด 8 ตัว เมื่อลำดับข้อมูล { 1 2 3 4 5 6 7 8 } ผ่านการเพอร์มูท จะได้ { 1 5 3 7 2 6 4 8 }

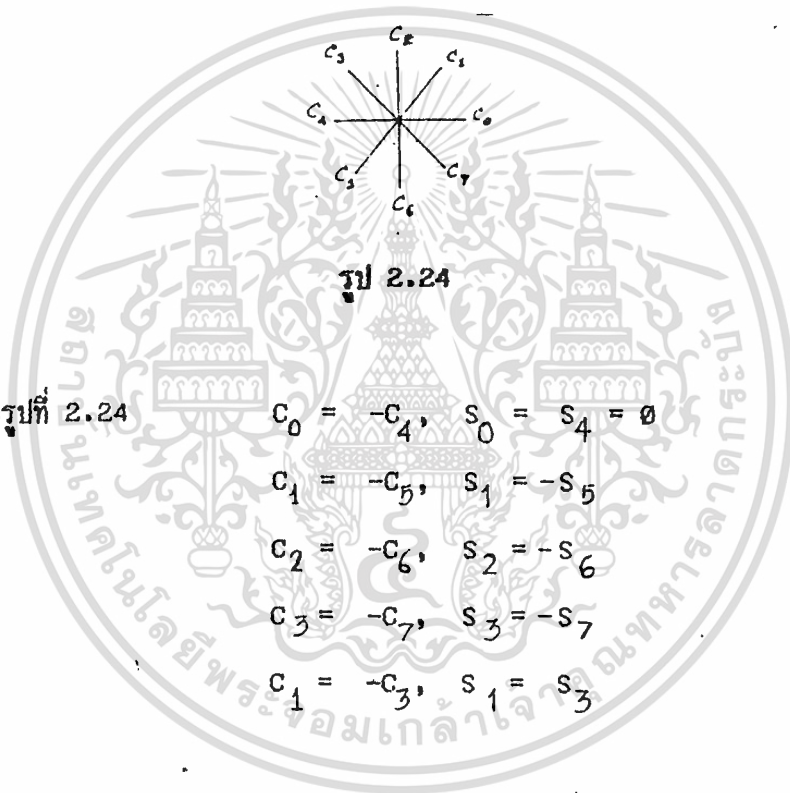
ในขั้นที่ 1 นำข้อมูลมาพิจารณากลุ่ม ๆ ละ 2 ตัว ได้ดังนี้ { 1 5 } { 3 7 } { 2

6 } { 4 8 } แล้วนำสมาชิกในแต่ละกลุ่มมาหาผลบวกและผลต่าง ไม่เช่น { 1 5 } จะได้  $(1+5)$  และการค้า  $(1-5)$  มี  $(6-4)$  อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในขั้นที่ 2 แบ่งข้อมูลเป็นกลุ่ม ๆ ละ 4 โดยทำดังในไดอะแกรม เช่น (6 -4 10 -4) จะได้  $((6+10) (-4 + -4) (6-10) (-4-(-4))) = (16 -8 -4 0)$  ซึ่งในที่สุดจะได้  $f(t)$

สำหรับในขั้นต่อไป ซึ่งในขั้นตอนนี้ จะมีแพลตฟอร์มของโคไซน์และไซน์เป็นตัวคูณอยู่ด้วย จาก  $C_n = \cos(2\pi n/2^L)$ ,  $S_n = \sin(2\pi n/2^L)$  ในที่นี้  $L = 3$

$$\text{จะได้ } C_n = \cos(2\pi n/8) \quad S_n = \sin(2\pi n/8)$$



$$\begin{aligned} F(3,0) &= F(2,0) + F(2,4)C_0 & F(3,4) &= F(2,0) - F(2,4)S_0 \\ F(3,1) &= F(2,1) + F(2,5)C_1 + F(2,7)S_1 & F(3,5) &= F(2,1) - [F(2,5)C_1 + F(2,7)S_1] \\ F(3,3) &= F(2,3) + F(2,5)S_2 - F(2,7)C_2 & F(3,7) &= F(2,3) - [F(2,5)S_2 - F(2,7)C_2] \end{aligned}$$

จากคุณสมบัติของโคไซน์และไซน์ ช่วยลดการคำนวณได้ สำหรับข้อมูลที่ห่างกัน  $N/2=4$  จุด หลังจากได้  $f(t)$  ก็นำผลลัพธ์นั้นมาหารด้วยจำนวนข้อมูลทั้งหมด ก็จะได้ผลลัพธ์เป็น  $H(f)$

### 2.8.3 วิเคราะห์ผลการทำงานของฟาสต์ฟูเรียร์ทรานสฟอร์มและฟาสต์อาร์ตเลย์ทรานสฟอร์ม

สำหรับเรื่องการเปรียบเทียบความเร็วของการทำงานระหว่างฟาสต์ฟูเรียร์ทรานสฟอร์มและฟาสต์อาร์ตเลย์ทรานสฟอร์ม มีหลายปัจจัยที่ต้องคำนึงถึง เช่น ภาษาที่ใช้ ประเภทเครื่องคอมพิวเตอร์ เป็นต้น ซึ่งหากปัจจัยแตกต่างกัน ก็จะได้ผลการเปรียบเทียบที่ต่างกันไปด้วย ตารางทดสอบการทำงานบนโปรเซสเซอร์ (processor) 80286 (speed 10 Mhz) ที่มี 80287 Math-Co. และฟังก์ชันที่ใช้ในการทดสอบคือ  $\cos(2 \pi f t)$  โดยที่ N เริ่มจาก 4 ถึง 1024 จุด

N	Time (Secs)		
	FFT	FHT	FFT/FHT
4	0.000	0.000	-
8	0.000	0.000	-
16	0.000	0.000	-
32	0.109	0.054	2.000
64	0.218	0.109	2.000
128	0.386	0.218	1.767
256	0.878	0.496	1.771
512	1.921	1.152	1.667
1024	4.066	2.582	1.574

เอกสารนี้เป็น **ตาราง 2.8.2.4** ตารางการเปรียบเทียบเวลาของฟาสต์ฟูเรียร์ทรานสฟอร์ม

บนขบวนการคำนวณ และฟาสต์อาร์ตเลย์ทรานสฟอร์ม

ถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตาราง 2.8.2.4 จะเห็นได้ว่า ความเร็วในการคำนวณของฟาสต์ฮาร์ดลีย์ทรานสฟอร์ม จะเร็วกว่าอยู่ในช่วงประมาณ 2 - 1.5 เท่าของฟาสต์ฟูเรียร์ทรานสฟอร์ม เนื่องจากในฟาสต์ฟูเรียร์ทรานสฟอร์ม มีการคำนวณค่าเชิงซ้อน ซึ่งยุ่งยากกว่า เช่น การคูณค่าเชิงซ้อน ก็จะต้องมีการคูณค่าจริง 4 ครั้ง และการบวกค่าจริง 2 ครั้ง

และจากสมการข้างต้น ได้กล่าวถึง จำนวนเครื่องหมาย (Operation) ที่ใช้ในการคำนวณไว้ดังนี้

ใน FFT  $A = N(3\ln(N) - 2) + 2$  สำหรับการบวกค่าจริง (real additions)  
 $M = 2N(\ln(N) - 2) + 4$  สำหรับการคูณค่าจริง (real multiplications)

ส่วน FHT  $A = 2N(\ln(N) - 1) + 2$  สำหรับการบวกค่าจริง (real additions)  
 $M = N(\ln(N) - 2) + 2$  สำหรับการคูณค่าจริง (real multiplications)

ในกรณีที่ N มีค่ามาก ๆ

$$\frac{FFT_A}{FHT_A} \approx \frac{N(3\ln(N) - 2)}{2N(\ln(N) - 1)} \approx \frac{3}{2} \quad \frac{FFT_M}{FHT_M} \approx \frac{2N(\ln(N) - 2)}{N(\ln(N) - 2)} \approx 2$$

เมื่อเปรียบเทียบจำนวนเครื่องหมายที่ใช้ในการคำนวณของฟาสต์ฟูเรียร์ทรานสฟอร์ม และฟาสต์ฮาร์ดลีย์ทรานสฟอร์ม พบว่า ในฟาสต์ฮาร์ดลีย์ทรานสฟอร์ม มีการคำนวณน้อยกว่า ซึ่งทำให้ความเร็วในการทำงานของฟาสต์ฮาร์ดลีย์ทรานสฟอร์ม เร็วกว่าหรือเท่ากับ ฟาสต์ฟูเรียร์ทรานสฟอร์ม แม้ว่าเมื่อจำนวน N มากขึ้น อัตราส่วนของฟาสต์ฟูเรียร์ทรานสฟอร์ม และฟาสต์ฮาร์ดลีย์ทรานสฟอร์ม จะมีแนวโน้มที่ลดลงบ้าง (ซึ่งเป็นไปตามสมการของจำนวนเครื่องหมายที่กล่าวข้างต้น) แต่ก็ยังอยู่ในช่วงที่ใกล้เคียงกัน เมื่อเปรียบเทียบกับตารางที่ 2.8.2.5 มีผลในลักษณะเดียวกัน

N	Time (Secs)		
	FFT	FHT	FFT/FHT
32	0.110	0.055	2.00
64	0.220	0.165	1.33
128	0.549	0.330	1.66
256	1.153	0.714	1.61
512	2.417	1.593	1.52
1024	5.163	3.405	1.52

ตาราง 2.8.2.5 เปรียบเทียบเวลาของฟาสต์ฟูเรียร์ทรานสฟอร์มและฟาสต์ฮาร์ดเลย์

#### 2.8.4 ข้อเปรียบเทียบระหว่างฟาสต์ฮาร์ดเลย์ทรานสฟอร์มและฟาสต์ฟูเรียร์ทรานสฟอร์ม

1. ข้อมูลและผลลัพธ์ของฟาสต์ฟูเรียร์ทรานสฟอร์ม มีทั้งส่วนจริงและส่วนจินตภาพ ซึ่งมีการคำนวณเป็นค่าเชิงซ้อน ซึ่งต้องใช้การคูณค่าจริง 4 ครั้ง และการบวกค่าจริง 2 ครั้ง ส่วนฟาสต์ฮาร์ดเลย์ทรานสฟอร์ม ข้อมูลและผลลัพธ์เป็นค่าจริงเท่านั้น ทำให้การคำนวณง่ายกว่า ฟาสต์ฟูเรียร์ทรานสฟอร์ม

2. จากสาเหตุข้อ 1 ฟาสต์ฮาร์ดเลย์ทรานสฟอร์ม สามารถทำงานได้ เร็วกว่า หรือเท่ากับฟาสต์ฟูเรียร์ทรานสฟอร์ม แม้ว่าฟาสต์ฮาร์ดเลย์ทรานสฟอร์ม จะเร็วกว่าฟาสต์ฟูเรียร์ทรานสฟอร์ม ไม่มากนัก แต่ก็ให้ความสะดวกสบายกว่า เพราะผลลัพธ์ที่ได้จากฟาสต์ฮาร์ดเลย์ทรานสฟอร์ม เป็นค่าจริงที่เรานำมาใช้ได้เลย ส่วนฟาสต์ฟูเรียร์ทรานสฟอร์มนั้นให้ผลลัพธ์เป็นค่าเชิงซ้อน ซึ่งมักจะต้องนำผลลัพธ์ค่าเชิงซ้อนนั้น มาแปลงให้เป็นค่าจริงอีกครั้ง

3. การใช้เนื้อที่ในการเก็บข้อมูลของฟาสต์ฮาร์ดเลย์ทรานสฟอร์ม น้อยกว่าฟาสต์ฟูเรียร์ทรานสฟอร์ม เนื่องจากข้อมูลของฟาสต์ฟูเรียร์ทรานสฟอร์มนั้น มีทั้งส่วนจริงและส่วนจินตภาพ ทำให้ต้องใช้เนื้อที่เก็บข้อมูลมากกว่าเป็น 2 เท่า ทั้งถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. อินเวอร์สอาร์ดเลย์ทรานสฟอร์ม นั้น มีรูปแบบเหมือนฟอร์วาร์ดทรานสฟอร์ม (forward transform) เพียงแต่เอาที่พุกของการไทม์ทูปรีควเอนซีทรานสฟอร์ม (time to frequency transform) จะต้องหารด้วยจำนวนข้อมูล (N) ด้วย แต่สำหรับเอาที่พุกของพรีควเอนซีทูปรีควเอนซีทรานสฟอร์ม (frequency to time transform) ไม่มีการเปลี่ยนแปลงใด ๆ

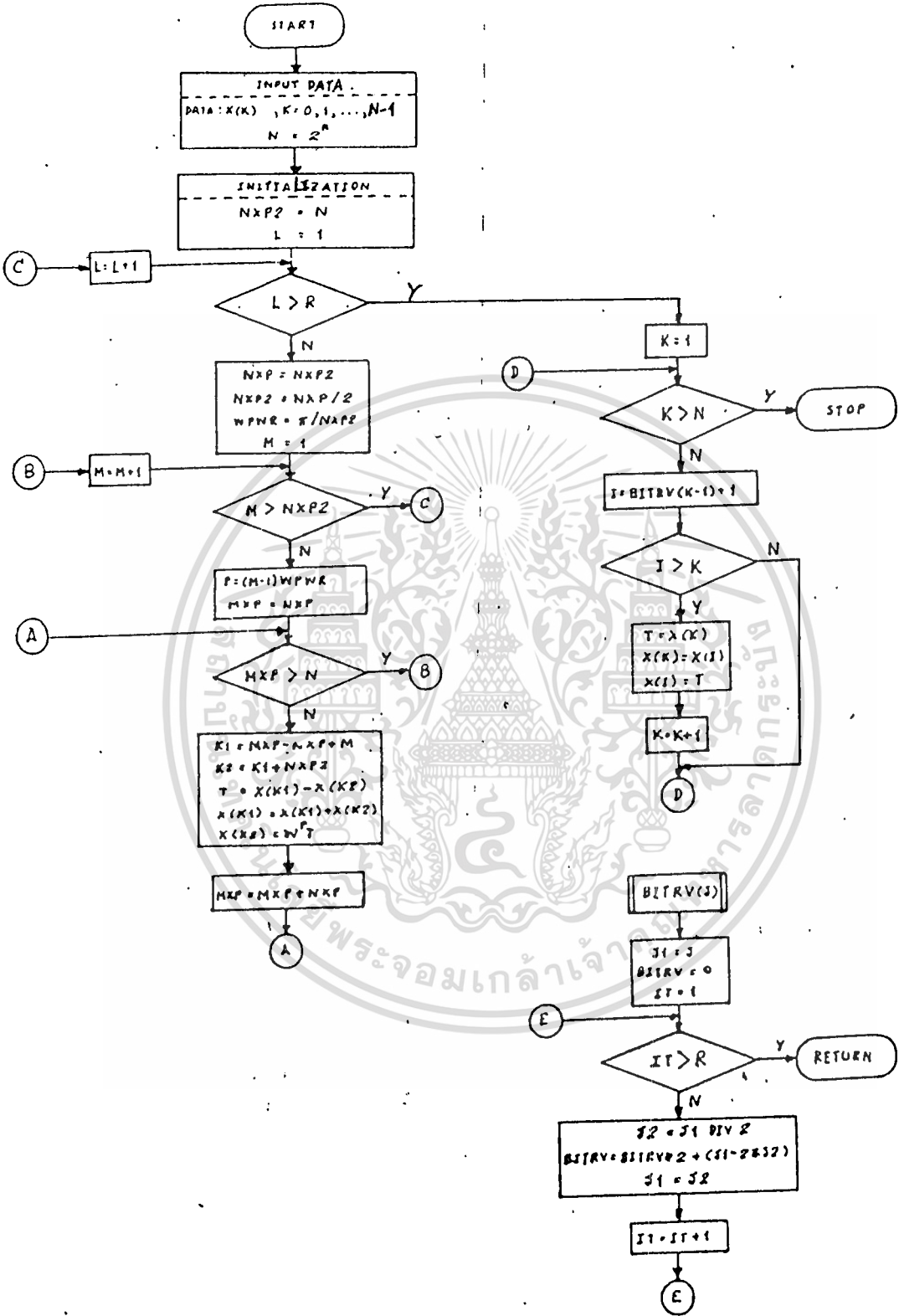
กรณีอินเวอร์สฟูเรียร์ทรานสฟอร์ม จะมีรูปต่างไปจากฟอร์วาร์ดทรานสฟอร์ม ที่  $e^{-j2\pi ft/n}$  เป็นตัวคูณของฟอร์วาร์ด แต่  $e^{j2\pi ft/n}$  เป็นตัวคูณของอินเวอร์ส

5. เราสามารถนำผลที่ได้จากอาร์ดเลย์ทรานสฟอร์ม ที่เป็นค่าจริงมาแปลงให้อยู่ในรูปของฟูเรียร์ทรานสฟอร์ม ที่เป็นค่าเชิงซ้อนได้ โดยความสัมพันธ์ดังนี้

$$H(f) = F_{re}(f) - F_{im}(f)$$

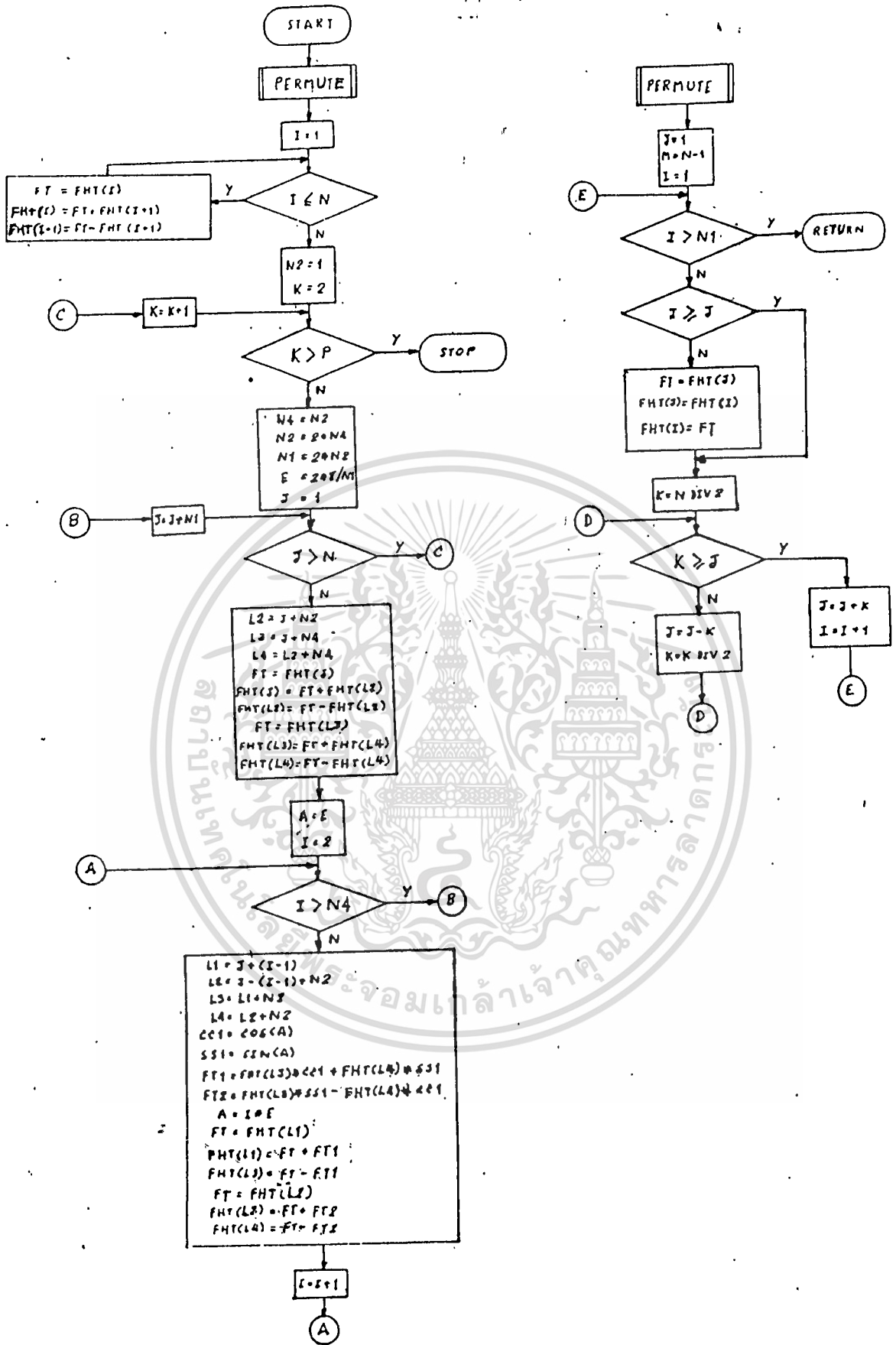
$$F = [H(f) + H(N-f)]/2 \quad F_{im} = [H(N-f) - H(f)]/2$$

จากข้อเปรียบเทียบข้างต้น จะเห็นได้ว่า ฟาสท์อาร์ดเลย์ทรานสฟอร์ม มีข้อได้เปรียบกว่า ฟาสท์ฟูเรียร์ทรานสฟอร์ม เพราะฟูเรียร์อาร์ดเลย์ทรานสฟอร์ม เป็นการคำนวณเฉพาะค่าจริงเท่านั้น ทำให้การคำนวณง่ายกว่าฟาสท์ฟูเรียร์ทรานสฟอร์ม ที่มีการคำนวณเป็นค่าเชิงซ้อน จึงทำให้การทำงานของฟาสท์อาร์ดเลย์ทรานสฟอร์ม เร็วกว่าหรือเท่ากับฟาสท์ฟูเรียร์ เนื่องจากในการเก็บข้อมูลก็น้อยกว่า ข้อมูลส่วนใหญ่ก็อยู่ในรูปของค่าจริง และการประยุกต์ใช้งานต่าง ๆ ที่ใช้ฟาสท์ฟูเรียร์ทรานสฟอร์ม เช่น สเปกทรัลแอนาไลซิส (spectral analysis) , การประมวลผลภาพในระบบดิจิทัล เป็นต้น



รูปที่ 2.25 โพลีชาร์ตแสดงอัลกอริทึมของฟาสต์ฟูเรียร์ทรานสฟอร์ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.26 โพลีชาร์ตแสดงอัลกอริทึมของฟาสต์ฟิวรีแตรนส์ฟอร์ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### บทที่ 3

#### การลดข้อมูลภาพ

(DATA REDUCTION OR DATA COMPRESSION)

การลดข้อมูลภาพ มีประโยชน์เด่นชัดในด้านช่วยลดพื้นที่สำหรับเก็บข้อมูล ลดปริมาณข้อมูลที่สื่อสาร โดยเฉพาะข้อมูลที่เป็ภาพ ซึ่งถ้าหากต้องการภาพที่ละเอียดข้อมูลภาพจะมีปริมาณมาก (จำนวนพิกเซลมาก) ดังนั้นเทคนิคต่าง ๆ จึงได้ถูกนำมาใช้เพื่อลดข้อมูลภาพลงก่อนจะกล่าวถึงเทคนิคต่าง ๆ ในการลดข้อมูลภาพ จะอธิบายลักษณะข้อมูลภาพดังต่อไปนี้

#### ลักษณะข้อมูลภาพ

##### ข้อมูลภาพขาวดำ

1 จุดของภาพมีขนาด 8 บิต สามารถบอกความแตกต่างของระดับสีได้ 256 ระดับ แต่ข้อมูลภาพจริงจะมีเพียง 64 ระดับ ภาพ 1 ภาพ ขนาด 256X256 จุด จะใช้เนื้อที่เก็บ 64 Kbytes

##### ข้อมูลภาพสี

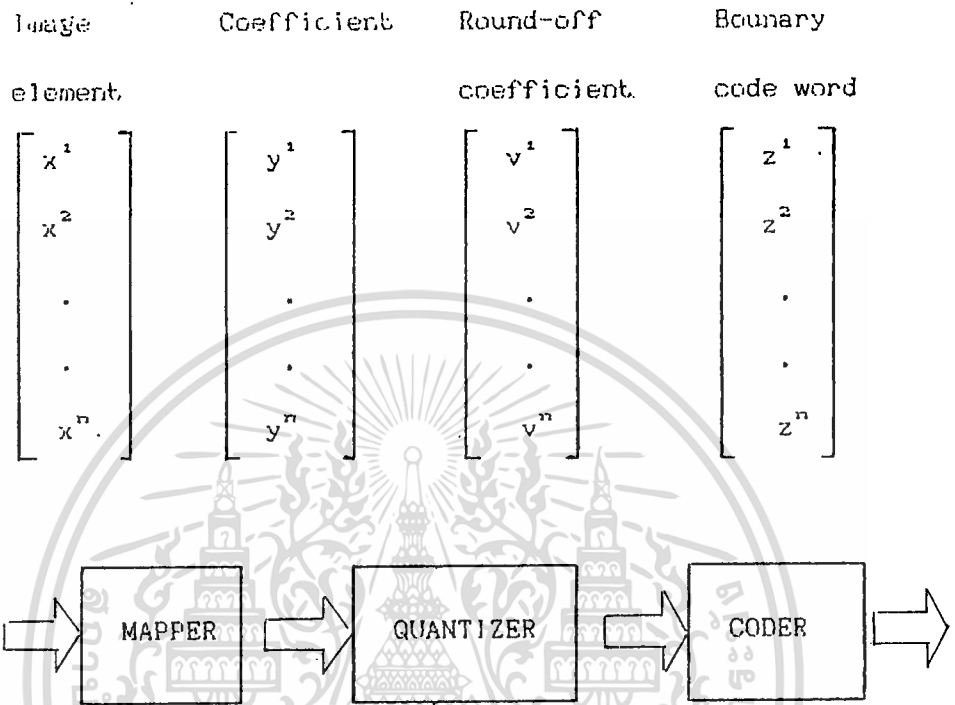
จะประกอบด้วย 2 ส่วน คือ

- 1) ส่วนที่เป็น attribute ของสี จะมีเนื้อที่ 768 bytes ส่วนนี้จะเป็นดัชนีที่จะบอกให้รู้ว่า ข้อมูลนั้น ๆ มีระดับสีเท่าไร
- 2) เป็นส่วนข้อมูลภาพที่มีขนาด 8 บิต สามารถให้ความแตกต่างของระดับสีได้ 256 ระดับ จะมีเนื้อที่ 64,๑๑๑ bytes

เทคนิคสำคัญที่จะกล่าวถึงต่อไปนี้ เป็นการเข้าได้ดข้อมูลภาพ (image encoding) ด้วยวิธีต่าง ๆ

## การเข้ารหัสข้อมูลภาพ (image encoding)

ขบวนการในการเข้ารหัสข้อมูลมี 3 ขั้นตอน ดังรูป



การแมปปีง (mapping) คือ การเปลี่ยนค่าของตัวเลข (พิกเซลทั้งหมด)

ให้อยู่ในอีกรูปหนึ่ง กล่าวคือเป็นการเข้ารหัสข้อมูลวิธีหนึ่ง ซึ่งการเข้ารหัสข้อมูล มีหลายวิธี โดยแต่ละวิธีจะมีคุณสมบัติเฉพาะของมัน วิธีต่าง ๆ ได้แก่

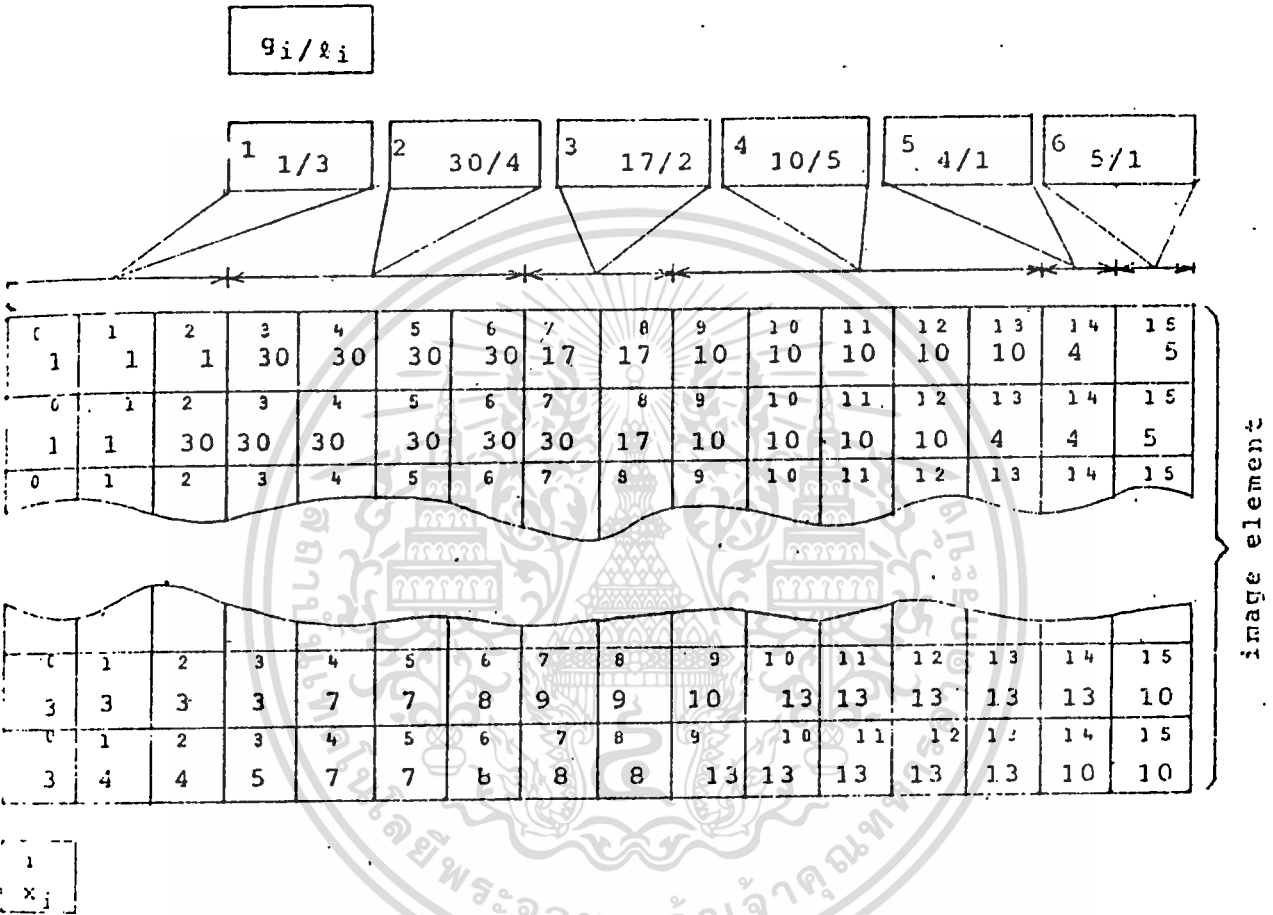
### 3.1 การลดข้อมูลภาพด้วยวิธีรันเลงจ์

(Runlength Compression)

การลดข้อมูลด้วยวิธีนี้ อาศัยลักษณะทั่วไปของข้อมูลภาพที่จะต้องมีส่วนของแบล็คกราว์ (background) และฟอร์กราว์ (foreground) ในส่วนของแบล็คกราว์จะมีการรายละเอียดของภาพไม่มากนัก ส่วนนี้เองจะมีการเปลี่ยนแปลงของข้อมูลน้อย เมื่อเทียบกับส่วนของฟอร์กราว์ ซึ่งมีรายละเอียด และการเปลี่ยนแปลงของข้อมูลมาก ในส่วนที่ทำการเปลี่ยนแปลงข้อมูลน้อยนี้เอง ที่เราสามารถนำการเข้ารหัสแบบรันเลงจ์ มาประยุกต์ใช้ได้ได้อย่างมีประสิทธิภาพ

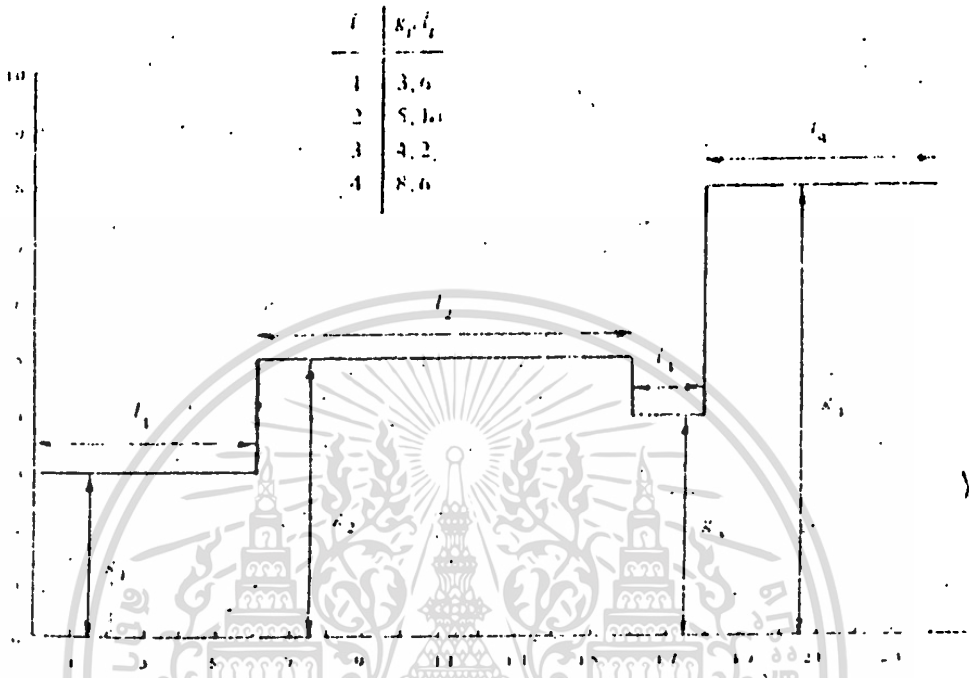
เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการเรียนการสอนเท่านั้น ไม่สามารถนำออกจำหน่ายหรือเผยแพร่โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ  $l_i$  แทนความยาวของข้อมูลหรือ จำนวนจุดที่มีระดับสีเทา  $g_i$ ,  
 เมื่อ  $i = 1, 2, 3, \dots, K$



รูปที่ 3.1.1 ลักษณะการเข้ารหัส Runlength โดยพิจารณาตามลำดับข้อมูลในแถวเดียวกัน (row major)

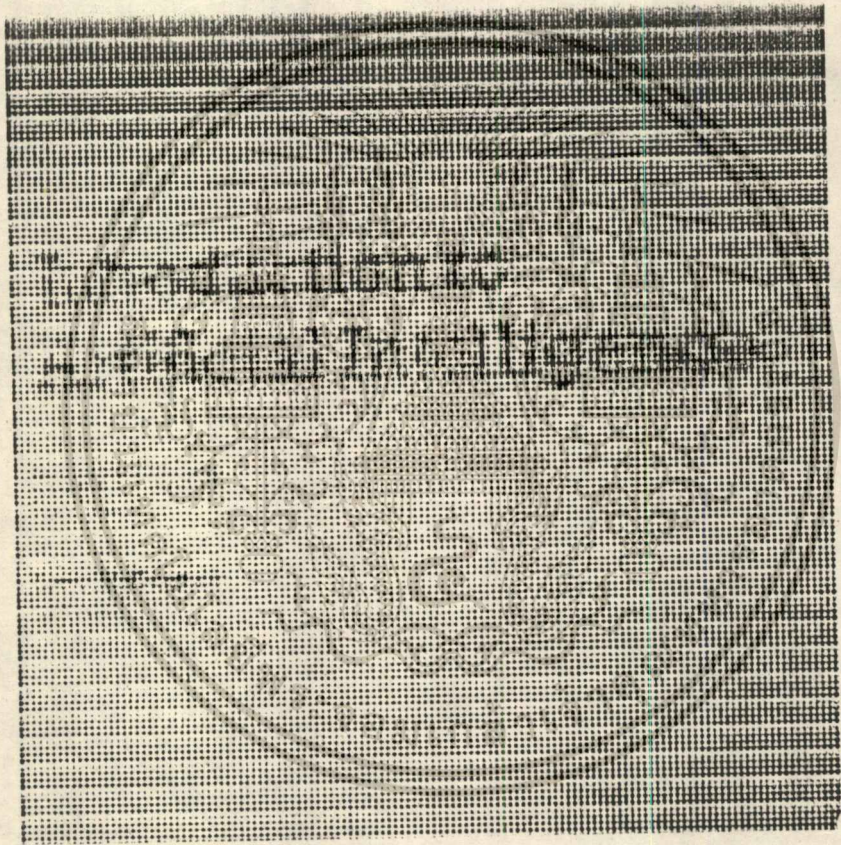
เราสามารถแสดงเป็นกราฟของการแมปปีง (mapping) ได้ดังรูป 3.1.2 สำหรับการถอดรหัส เราก็สามารถทำได้ โดยการ mapping ย้อนกลับ



รูปที่ 3.1.2 ตัวอย่างรันเลงจ์แมปปีง (Runlength mapping)

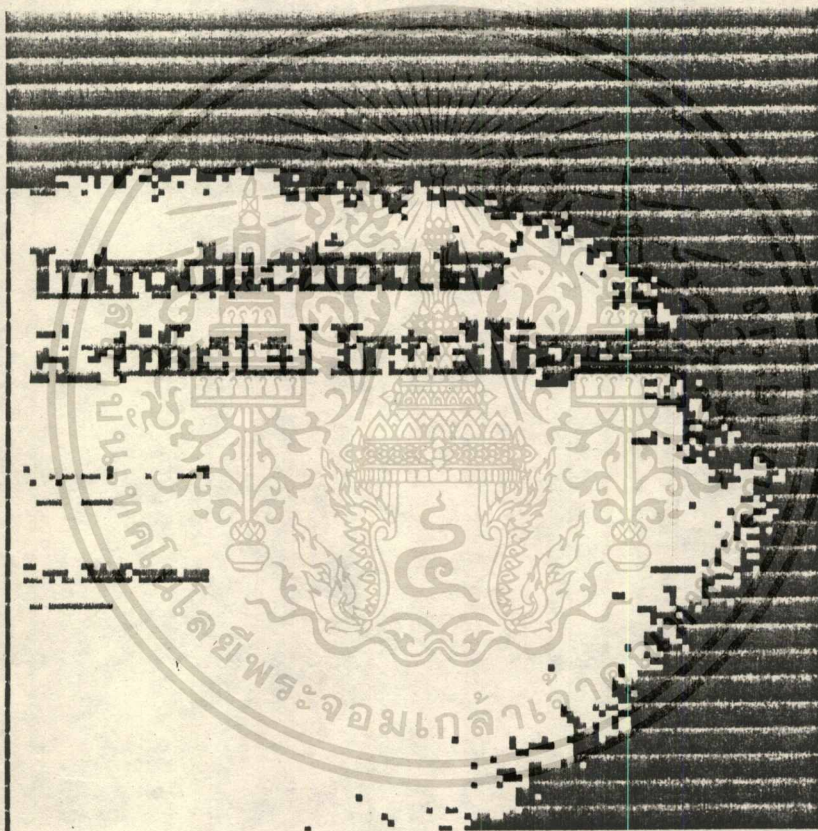
### ความสามารถในการลดข้อมูลงานด้วยรันเลงจ์

ในกรณีที่ข้อมูลภาพ มีความแตกต่างของระดับสีเทามาก อัตราของข้อมูลต่อ 1 จุดภาพ (Data Rates.) จะมากตามด้วย เนื่องจากที่ความเป็นไปได้ที่ข้อมูลจะมีการเปลี่ยนแปลงสูง ดังแสดงในรูปที่ 3.1.3 และรูปที่ 3.1.4 เปรียบเทียบอัตราข้อมูลของรูปที่มีระดับสีเทา 256 ระดับ และรูปเดียวกัน แต่มีระดับสีเทาเพียง 2 ระดับ ตามลำดับ



รูปที่ 3.1.3 แสดงรูปหลังถอดรหัสมีอัตราข้อมูล 6.18 บิตต่อ 1 จุด ความคลาดเคลื่อน  
อิสระ (error free) ขนาด  $256 \times 256$  จุด<sup>2</sup> 256 ระดับสี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1.4 แสดงรูปหลังถอดรหัส มีอัตราข้อมูล 0.75 บิตต่อจุด

ความคลาดเคลื่อน ขนาด 256 x 256 จุด<sup>2</sup> 2 ระดับสี

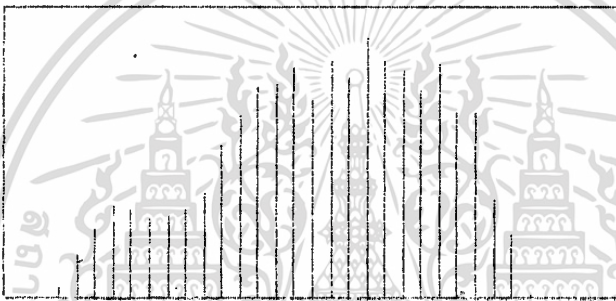
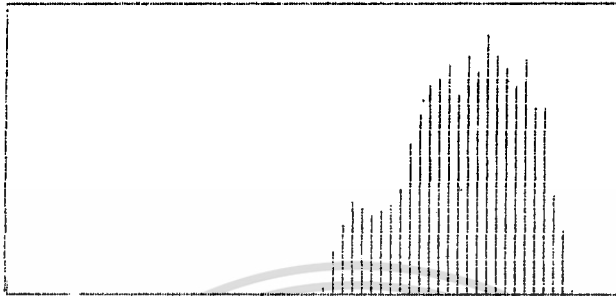
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความแตกต่างของระดับสีเทา	อัตราข้อมูลภาพบิตต่อจุด
256 ระดับ	8
64 ระดับ	6
32 ระดับ	5
8 ระดับ	3
2 ระดับ	1

ตารางที่ 3.1.5 แสดงอัตราข้อมูลของภาพเดียวกัน แต่มีความแตกต่างของระดับสีต่างกัน

การลดข้อมูลด้วยรันเลงจ์ จะไม่มีความคลาดเคลื่อนเกิดขึ้น หลังจากการถอดรหัส  
 เลข แต่การใช้วิธีนี้ต้องพิจารณาถึง ลักษณะของภาพต้นแบบว่ามีความเหมาะสมหรือไม่  
 ภาพที่เหมาะสม ควรมีรายละเอียดของภาพน้อย เช่น ภาพเอกสาร ภาพทางงานกราฟ  
 ฟิก (graphic) จากตาราง เราสามารถลดอัตราข้อมูลภาพได้ โดยการปรับข้อมูล  
 ภาพต้นแบบให้มี ความแตกต่างของระดับสีเทาให้น้อยลง ด้วยวิธีของเกรย์สเกลทรานส์ฟอร์  
 เมชัน (Gray Scale Transformation)

การปรับข้อมูลให้มี ความแตกต่างของระดับสีเทาให้น้อยลง ด้วยวิธีของเกรย์สเกล  
 ทรานส์ฟอร์เมชัน เป็นการปรับขยายช่วงระดับสีเทา ซึ่งอยู่ในช่วงที่เหมาะสม



รูปที่ 3.1.6 ฮิสโตแกรมของภาพก่อน และหลังการทำเกรย์สเกลทราشهอร์เมชั่น

ภาพเดิมมีปริมาณของข้อมูลมาก ให้กระจายออกโดยเปลี่ยนค่า ระดับสีเทา ดังสมการ

$$W_1 = \frac{(W_1 - a) \times 255}{(b - a)}$$

หลังจากปรับค่าระดับสีเทาให้น้อยลงแล้ว จะทำให้ภาพมีความคมชัดมากขึ้นด้วย  
เนื่องจากมีการกระจายของระดับสีเทา ให้กว้างขึ้น (contrast มาก) ตาคาเราจะสามารถแยกรายละเอียดได้ดีขึ้น

### 3.2 การลดข้อมูลภาพด้วยวิธีฮัฟฟ์แมนโคด

(Huffman coding compression)

การเข้ารหัสด้วย ฮัฟฟ์แมนโคด อาศัยลักษณะทั่วไปของภาพที่ว่า ภาพต่าง ๆ จะมีระดับสีเทาที่แตกต่างกัน ด้วยอัตราความถี่ที่ไม่เท่ากัน ระดับสีเทาใดมีมาก ก็จะถูกแทนด้วยข้อมูลระดับบิตที่น้อย ระดับสีเทาใดมีน้อย ก็จะถูกแทนด้วยข้อมูลระดับบิตที่มาก ตัวอย่างการหา ฮัฟฟ์แมนโคด เราสมมติให้ภาพ ๗ หน้า มีระดับสีเทา 8 ระดับ ขนาด  $256 \times 256$  จุด<sup>2</sup> ซึ่งแต่ละระดับ มีความถี่ของข้อมูลที่ระดับสีเทานั้น ดังนี้

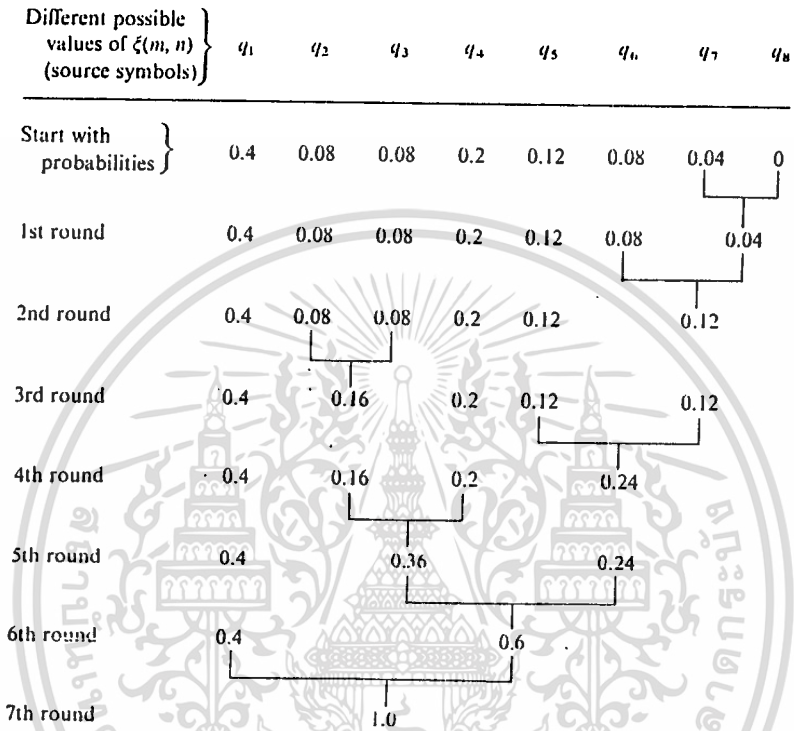
$$\begin{array}{ll}
 f_1 = 26214 & f_5 = 7865 \\
 f_2 = 5243 & f_6 = 5243 \\
 f_3 = 5243 & f_7 = 2621 \\
 f_4 = 13107 & f_8 = 0
 \end{array} \quad (3.2.1)$$

เราสามารถหาความน่าจะเป็น ของแต่ละระดับสี ได้ดังนี้

$$\begin{array}{ll}
 p_1 = p\{f_1\} = 0.4 & p_5 = p\{f_5\} = 0.12 \\
 p_2 = p\{f_2\} = 0.08 & p_6 = p\{f_6\} = 0.08 \\
 p_3 = p\{f_3\} = 0.08 & p_7 = p\{f_7\} = 0.04 \\
 p_4 = p\{f_4\} = 0.2 & p_8 = p\{f_8\} = 0
 \end{array} \quad (3.3.2)$$

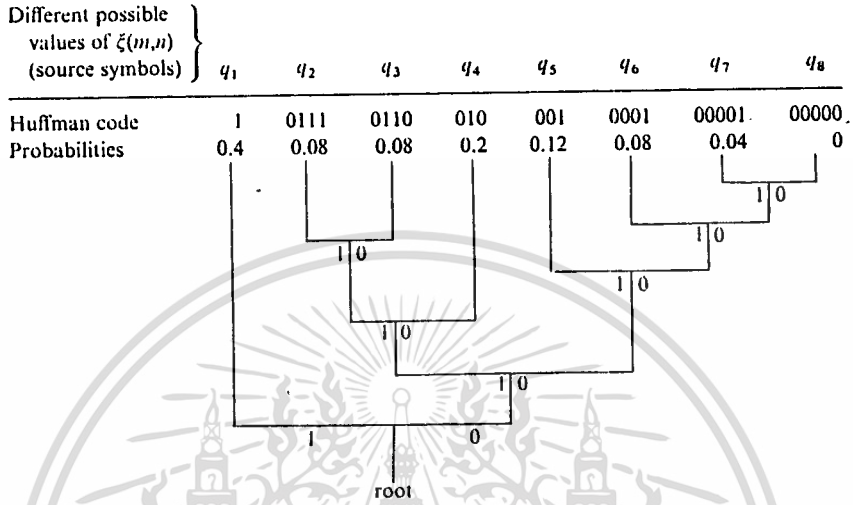
ในทางปฏิบัติ อาจหาค่า  $p_i$  ได้ด้วยแผนภาพฮิสโตแกรม จากนั้นให้ทำตามขั้นตอนต่าง ๆ (procedure) ดังนี้

- ขั้นตอนที่ 1 หาค่าความน่าจะเป็นที่มีค่าน้อยที่สุด 2 ค่า ในตัวอย่างนี้คือ ลำดับที่ 7 และ 8 ซึ่งมีค่าความน่าจะเป็น 0.04 และ 0 ตามลำดับ
- ขั้นตอนที่ 2 บวกค่าความน่าจะเป็นทั้งสอง ผลทำให้ค่าความน่าจะเป็นทั้งหมด ลดลงไป 1 ค่า โดยมีค่าความน่าจะเป็นตัวใหม่ ที่เป็นผลบวกนี้เกิดขึ้นด้วย
- ขั้นตอนที่ 3 ทำขั้นตอนที่ 1 และ 2 ซ้ำ จนได้ค่าความน่าจะเป็นเพียงตัวเดียว



รูปที่ 3.2.1 การสร้างทรี (tree) สำหรับ ฮัฟฟ์แมนโค้ด

ในรูป 3.2.1 จะเห็นว่าการทำตามขั้นตอนที่ 1 และ 2 (รวมกันเป็น 1 รอบ) หลังจากทำรอบแรก จะทำให้ค่าความน่าจะเป็นเหลือเพียง 7 ค่า ทำเช่นนี้เรื่อยไป จนสุดท้าย จะเหลือค่าความน่าจะเป็นเพียง 1 ค่า ในรูป 3.2.1 สามารถเขียนได้ดังรูป 3.3.2 เพื่อให้มีลักษณะเหมือนทรี การสร้าง โค้ดเวิร์ด เริ่มจากรูท (root) และ กำหนดค่าให้กับบรานซ์ (branch) หนึ่ง ค่า 0 ให้กับอีกบรานซ์หนึ่ง การกำหนดค่า 1 ให้กับบรานซ์ที่มีค่าเท่ากัน จะทำที่แต่ละโหนด (node) ของทรี ถึงตอนเห็นโค้ดเวิร์ด สำหรับข้อมูลที่กำหนดหาได้ โดยการไล่ตามทางเดินรูท ของทรีไปจนหมด ซึ่งจะได้ชุดข้อมูลของ 0 กับ 1



รูปที่ 3.2.2 รูปทรี ที่สร้างขึ้นใหม่ จากรูป 3.2.1 ถูกแสดงให้ชัดเจนขึ้น เพื่อหาขีดเวรต์

ค่าต่ำสุด (lower bound) ที่จะสามารถเป็นได้ ของอัตราข้อมูล หาได้จาก

สมการ

$$H = - \sum_{i=1}^K P_i \log P_i \quad (3.3.3)$$

โดย K คือ จำนวนความน่าจะเป็น ( $P_i$ ) ของระดับสีทั้งหมด

เรียก H ว่า ค่าต่ำสุดที่สามารถเป็นได้ของอัตราข้อมูล (entropy) จากสม

การ 3.2.2 และ 3.2.3 จะได้

$$H = - [0.4 \log_2 0.4 + 3(0.08 \log_2 0.08) + 0.2 \log_2 0.2 + 0.12 \log_2 0.12 + 0.04 \log_2 0.04]$$

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ 2.42 การใช้งานเพื่อการศึกษานี้ ไม่อนุญาตให้นำไปใช้ (3.3.4) ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบกับค่าเฉลี่ยของโค้ดเวอร์ด (average length, R)

$$\begin{aligned}
 R &= (1 \times 0.4) + (3 \times 4 \times 0.08) + (3 \times 0.2) + (3 \times 0.12) + (5 \times 0.04) \\
 &= 2.52 \quad \text{บิต} \qquad (3.3.5)
 \end{aligned}$$

	Entry number	List of probabilities	Left branch	Right branch
Initial entries in the list	1	0.4	—	—
	2	0.08	—	—
	3	0.08	—	—
	4	0.2	—	—
	5	0.12	—	—
	6	0.08	—	—
	7	0.04	—	—
	8	0	—	—
	9	0.04	7	8
	10	0.12	6	9
	11	0.16	2	3
	12	0.24	5	10
	13	0.36	4	11
	14	0.6	12	13
	15	1.0	1	14

รูปที่ 3.2.3 การใช้โครงสร้างทรี ในคอมพิวเตอร์

วิธีการสร้างฮัฟฟ์แมนโค้ดดังกล่าว มีลักษณะดังนี้

อธิบายโดยใช้ตัวอย่างเดิม เริ่มจากการเขียนความน่าจะเป็นทั้งหมด ที่มีคือ 8 ค่า เรียงลำดับดังรูป 3.2.3 จากนั้นหาความน่าจะเป็นสองตัว ที่มีค่าน้อยที่สุด บวกกัน ผลบวกที่ได้ใส่เป็นลำดับที่ 9 ในตาราง โดยคอลัมน์เลขปารานซ์ (column left branch) และไลต์ปารานซ์ (right branch) จะใส่ลำดับที่มาของผลบวกที่ตำแหน่งนั้น จากนั้นตัดลำดับ และทิ้งแล้วหาคู่ที่มีความน่าจะเป็นน้อยที่สุดอีกได้ ลำดับที่ 6 กับ 9 (ลำดับที่ 6 เท่ากับ 2, 3 ซึ่งเราอาจจะเลือกอันใดอันหนึ่งก็ได้) ผลบวกที่ได้ใส่เป็นลำดับที่ 10 จากนั้นตัดลำดับ 6 กับ 9 ออกใส่เป็นยรรานส์ซ้ายขวา ของลำดับ 10 ทำเช่นนี้

จนกระทั่งได้ค่าความน่าจะเป็นเท่ากับ 1 ซึ่งสมนัยกับรากของทรี ในรูป 3.2.3 ใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

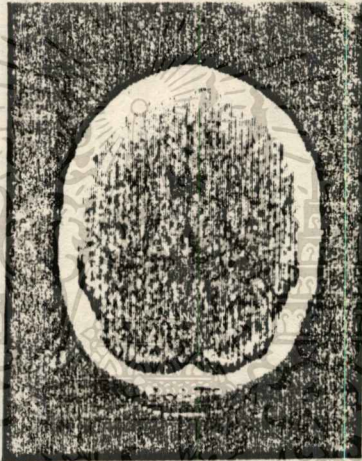
การหาค่าไบนารีโค้ดเวิร์ดก็สามารถทำได้โดย เริ่มที่ลำดับหลังสุดของค่าความน่าจะเป็นในตารางข้างต้น โดยกำหนดค่า 1 ให้กับบรานซ์ซ้าย และ 0 ให้กับบรานซ์ขวา ในรูป 3.2.3 ลำดับหลังสุดคือ 15 ดังนั้น เรากำหนดค่า 1 ให้กับลำดับ 1 และ 0 ให้กับลำดับ 14 เนื่องจากลำดับ 1 ไม่มีบรานซ์ ดังนั้น ไบนารีโค้ดเวิร์ดที่สมมุติกับลำดับนี้คือ 1 อีกด้านหนึ่ง คือ ลำดับ 14 มี 2 บรานซ์ เราก็กำหนด 1 ให้กับลำดับ 12 และ 0 ให้กับลำดับ 13 วิธีการนี้ทำไปจนกระทั่งถึงลำดับที่ไม่มีบรานซ์เลย โค้ดเวิร์ดได้จากชุดของ 0 กับ 1 ที่ไล่ตามทางเดินจากกรูท ไปยังจุดที่เริ่ม

ฮังเงอแมโค๊ด เป็นยูนิโค๊ดโค๊ดเดเบิลโค๊ด (uniquely decodable code) เพราะค่าไบนารีเหล่านี้สามารถถอดรหัสได้ในทางเดียว (one way) ตัวอย่างเช่น เรามีโค๊ดของภาพ ๗ หน้า เป็น ๐11๐๑๐1 ค่าตัวเลขไบนารีที่อาจเป็น โค๊ดเวิร์ด อาจเป็น ๐, ๐1 หรือ ๐11 ก็ได้ อย่างไรก็ตามค่าเหล่านี้ไม่ตรงกับ โค๊ดเวิร์ด,  $q_i$  ใด ๆ เลย เราสามารถถอดรหัสได้เป็น ๐11๐ แทน โค๊ด เวิร์ด  $q_3$  และไม่ต้องสงสัย ตัวถัดมาคือ 1 ซึ่งแทน โค๊ดเวิร์ด ของ  $q_1$  สุดท้ายคือ ๑๐๑๐1 แทน โค๊ด เวิร์ด  $q_7$  เพราะค่าไบนารีที่เป็นได้ คือ ๐, ๐๑, ๑๐๑ โค๊ดเวิร์ด,  $q_i$  ใด ๆ เลย

### 3.3 การลดข้อมูลภาพโดยวิธีฮัฟฟ์แมนเชิงที่โคัด

( Huffman shift coding compression )

การเข้ารหัสโดยใช้ ฮัฟฟ์แมนโคัด นั้น มีข้อเสียเกิดขึ้นในกรณีที่ ภาพที่เราจะนำไปเข้ารหัสนั้น มี ระดับสีเทามาก ตัวอย่างดังรูป 3.3.1 ซึ่งเป็นภาพขนาด  $256 \times 256$  จุด มีระดับสีเทา 4,096 ระดับ ( 12 บิตต่อจุด ) ถ้าเราจะอาศัย ฮัฟฟ์แมนโคัด ช่วยในการเข้ารหัส จะต้องใช้ โคัด ที่มีขนาดใหญ่มาก ซึ่งจะทำการถอดรหัสเป็นไปได้ช้า และยากขึ้น

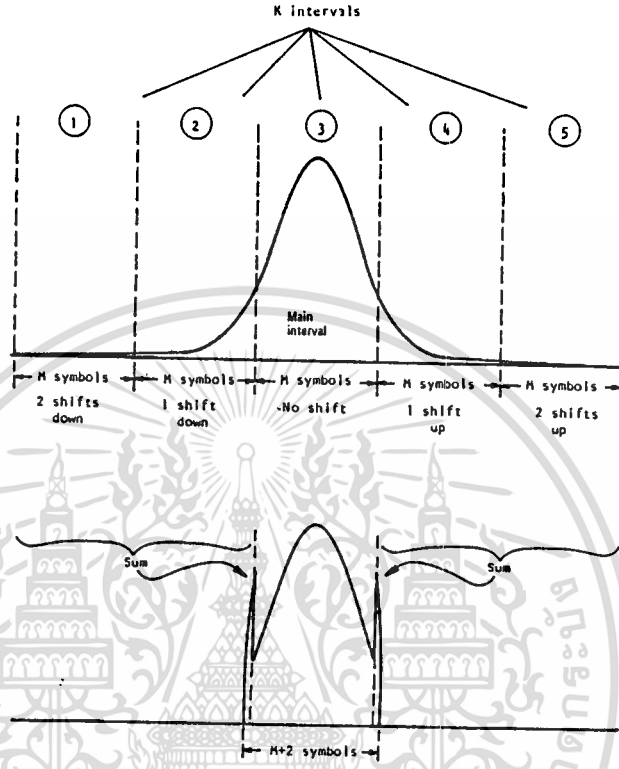


รูปที่ 3.3.1 แสดงภาพขนาด  $256 \times 256$  จุด<sup>2</sup> อัตราข้อมูล 12 บิตต่อจุด

การแก้ปัญหาเหล่านี้ ทำได้โดยอาศัย ฮัฟฟ์แมนเชิงที่โคัด โคัดซึ่งมีหลักการเข้ารหัส คือ พิจารณาแผนภาพฮิสโตแกรมของภาพต้นแบบ แล้วแบ่งภาพฮิสโตแกรมออกเป็น L ส่วน แต่ละส่วนมีจำนวนระดับสีเทา M ระดับ จากนั้นเราจะมองภาพเป็น 3 ส่วน คือ ด้านซ้าย กลาง และ ด้านขวา จากนั้น ทำการบวกค่าความเ่างจะเป็นทั้งหมดของภาพด้านซ้ายเข้าด้วยกัน แล้วแทนค่าความเ่างจะเป็นนี้ ให้กับระดับสีเทา ที่อยู่ขอบซ้าย (left margin) ของภาพส่วนกลาง และดำเนินตามขั้นตอนเดียวกันกับ ภาพด้านขวา ซึ่งสามารถแสดงได้ ดังรูป 3.3.2 จากรูปจะทำให้ เราสามารถลดจำนวน ระดับสีเทาลงเหลือเพียง  $M+2$  ระดับ หลังจากนั้น ก็ทำการเข้ารหัส ตามวิธีของฮัฟฟ์แมนเชิงที่โคัดตัวแรก เรียกว่า ชิฟต์ดาวนเวิร์ด (shift down word) และ โคัดเวิร์ด ตัวสุดท้าย

(M+2) เรียกว่า ชิฟต์อัพเวิร์ด (shift up word) ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3.2 แสดงวิธีการสร้าง ซิงท์ โค้ด ภาพบนแสดงฮิสโตแกรมของภาพต้นแบบ ซึ่งถูกแบ่งเป็น  $L$  ส่วน มีส่วนที่ 3 เป็นส่วนกลาง (main interval) ภาพล่างเป็นภาพที่ทำการ ซิงท์ โค้ด เรียบร้อยแล้ว

ตามรูปที่ 3.3.2 สมมติว่าเราต้องการเข้ารหัสของระดับสีที่  $m$  ในส่วนของเมอเอ็นเทอร์วัล เราสามารถหาได้โดยตรง ก็คือ ยังงั้นแมนโค้ดเวิร์ด ตัวที่  $m+1$  ถ้าเราต้องการเข้ารหัสของระดับสีที่  $m$  ในส่วนที่ 1 ซิงท์อับ เราจะได้รหัสซึ่งเกิดจากซิงท์อับเวิร์ด ตามด้วย ยังงั้นแมนโค้ดเวิร์ด ตัวที่  $m$  ของส่วนเมอเอ็นเทอร์วัล ถ้าเราต้องการเข้ารหัสของระดับสีที่  $m$  ในส่วนที่ 3 ซิงท์อับ เราจะได้รหัสซึ่งเกิดจาก ซิงท์อับ

ต่อกันเป็นจำนวน  $J$  ตัว แล้วตามด้วย ยังงั้นแมนโค้ดเวิร์ดตัวที่  $m$  ของช่วงเมอเอ็นเทอร์วัล  
 ไปเทอร์วัลใดในส่วนเซอร์วิล ซึ่งยังคงเป็น ก็ยังสามารถทำได้ในลักษณะเดียวกันกับบริเวณซิงท์อับ

### ความสามารถในการลดข้อมูลภาพโดยฮัฟฟ์แมนเชิงที่โคัด

อัตราการลดข้อมูลขึ้นอยู่กับ การแบ่งภาพฮิสโตแกรม ออกเป็นกี่ส่วน ซึ่งมีปริมาณการแบ่งช่วงมาก (  $M$  น้อย  $L$  มาก ) ก็จะทำให้อัตราการลดข้อมูลเมื่อเปรียบเทียบกับฮัฟฟ์แมนโคัดลดลง แต่ถ้ามีการแบ่งช่วงน้อย (  $M$  มาก  $L$  น้อย ) ก็จะทำให้อัตราการลดข้อมูลเมื่อเปรียบเทียบกับฮัฟฟ์แมนโคัดมากขึ้น ดังในตารางที่ 3.3.1 ถ้าเราได้สร้างโคัดขึ้นมา 16 โคัดเวิร์ด ( นั่นหมายถึง  $M = 14$  ) เราจะสามารถลดข้อมูลขนาด 12 บิตต่อ 1 จุด ให้เหลือเพียง 6.75 บิตต่อ 1 จุด คิดเป็นอัตราการลดข้อมูล 44 % เมื่อเปรียบเทียบกับ การเข้ารหัสแบบฮัฟฟ์แมนโดยตรง พบว่ามีอัตราการลดข้อมูล 49 % ซึ่งแตกต่างกันเพียงเล็กน้อย แต่ฮัฟฟ์แมนเชิงที่โคัด จะได้เปรียบในเรื่องของเวลาในการเข้ารหัส และถอดรหัส เพราะมีจำนวนโคัดเวิร์ดที่น้อยกว่า

วิธีลดข้อมูลภาพ	อัตราข้อมูลบิตต่อจุด	ความสามารถในการลดข้อมูล (%)
ฮัฟฟ์แมนเชิงโคัด	8 โคัดเวิร์ด	32
	16 โคัดเวิร์ด	44
	32 โคัดเวิร์ด	47
	64 โคัดเวิร์ด	47
	128 โคัดเวิร์ด	48
ฮัฟฟ์แมนโคัด	8192 โคัดเวิร์ด	49
เอนโทรปี (entropy)	6.09	49

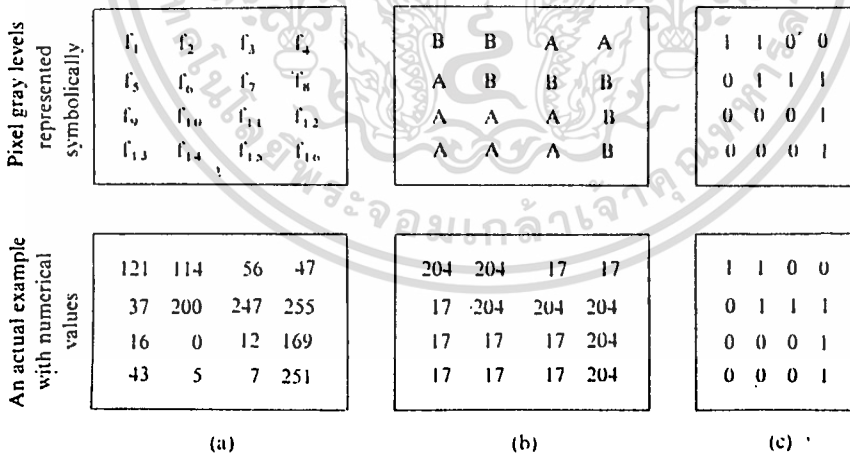
ตารางที่ 3.3.1 เปรียบเทียบอัตราข้อมูล หลังจากเข้ารหัสภาพในรูปแบบที่ 3.3.1

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งด้วยฮัฟฟ์แมนเชิงที่โคัด และ ฮัฟฟ์แมนโคัด ของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4 การลดข้อมูลภาพด้วยวิธีบีบอัดทรวงเคชั่น

(Block Truncation compression)

เป็นวิธีการเข้ารหัสที่อาศัยการแบ่งภาพออกเป็นบล็อก จำนวน  $N \times N$  บล็อก แต่ละบล็อก มีขนาด  $M \times M$  จุด โดย  $M \ll N$  จากนั้นจะอาศัยทฤษฎีทางสถิติ ทำการเปลี่ยนข้อมูลที่มีระดับสีเทาเพียง 2 ระดับ โดยข้อมูลที่ถูกลบเปลี่ยนนี้จะเปลี่ยนไปอย่างเหมาะสม เมื่อข้อมูลในแต่ละบล็อก ถูกเปลี่ยนให้มีระดับสีเทาเพียง 2 ระดับ เราก็จะสามารถแทนข้อมูลเหล่านั้น ด้วยตัวเลขไบนารีได้ ซึ่งจะช่วยให้ลดข้อมูลลงได้ ดังแสดงในรูป 3.4.1



- รูปที่ 3.4.1 a) ส่วนของภาพที่ถูกตัดออกเป็นบล็อก ขนาด  $4 \times 4$  จุด<sup>2</sup>  
 b) ส่วนของภาพหลังจากถูกเปลี่ยนเป็นข้อมูล 2 ระดับสี  
 c) การแทนข้อมูลรูป 3.4.1 (b) ด้วยตัวเลขไบนารี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป 3.4.1 a) แสดงส่วนของภาพขนาด  $4 \times 4$  จุด<sup>2</sup> ซึ่งเราจะแทนระดับสีเทา ด้วย  $f_1, f_2, \dots$  เราจะคำนวณหา ค่าเฉลี่ย,  $\bar{f}$  ของแต่ละบล็อก ได้ดังนี้

$$\bar{f} = 1/M \sum_{i=1}^m f_i \quad (3.4.1)$$

โดย  $m = M^2$  (กรณี = 16)

ในการเปลี่ยนระดับสีเทาต่าง ๆ ของแต่ละบล็อก ให้เหลือเพียง 2 ระดับ เราจะนำระดับสีเทาของข้อมูลจริง,  $f_i$  ไปเปรียบเทียบกับ ค่าเฉลี่ยของระดับสีเทาในแต่ละบล็อก,  $\bar{f}$

ถ้า  $f_i < \bar{f}$  เราจะเปลี่ยนข้อมูล  $f_i$  ให้มีค่าเป็น A

ถ้า  $f_i > \bar{f}$  เราจะเปลี่ยนข้อมูล  $f_i$  ให้มีค่าเป็น B

ดังได้แสดงไว้ในรูป 3.4.1 b) ซึ่งเราสามารถหาค่าของ A และ B ได้ จากสมการ ต่อไถ่

$$\bar{f} = \frac{1}{m} \sum_{i=1}^m f_i = \frac{1}{m} [(m-q) A + qB] \quad (3.4.2)$$

$$f^2 = \frac{1}{m} \sum_{i=1}^m f_i^2 = \frac{1}{m} [(m-q) A^2 + qB^2] \quad (3.4.3)$$

โดย คือ จำนวนของจุด ซึ่งมี  $f_i \geq \bar{f}$  เมื่อแก้สมการทั้งสองจะได้ ค่า A และ B ดังนี้

$$A = \hat{f} - \bar{\sigma} \sqrt{\frac{q}{m-q}} \quad (3.4.4)$$

และ

$$B = \hat{f} + \bar{\sigma} \sqrt{\frac{m-q}{q}} \quad (3.4.5)$$

โดย

$$\bar{\sigma} = \sqrt{f^2 - (\bar{f})^2}$$

ในการเข้ารหัส เพื่อเก็บลงไฟล์ (file) เราจะเก็บข้อมูล 2 ส่วน คือ

1) ส่วนของตัวเลขไบนารี (Bit Plane), ดังรูป 3.4.1 c) และ รูป 3.4.2 ส่วนของค่า

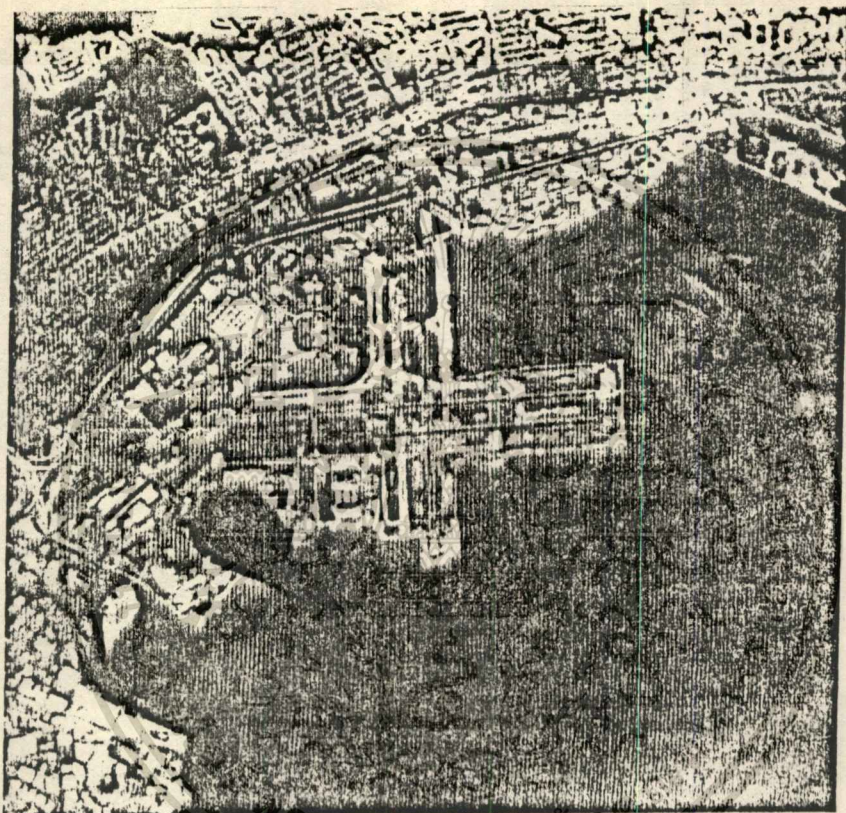
$\bar{f}$  และ ในกรณีที่เราแบ่งภาพออกเป็นบล็อก: ขนาด  $4 \times 4$  จุด<sup>2</sup> เราสามารถลดข้อ

มูลภาพต้นแบบ ที่มี ระดับสีเทา 256 ระดับ 8 บิตต่อ 1 จุด ให้เหลือเพียง 2 บิตต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

1 จุด

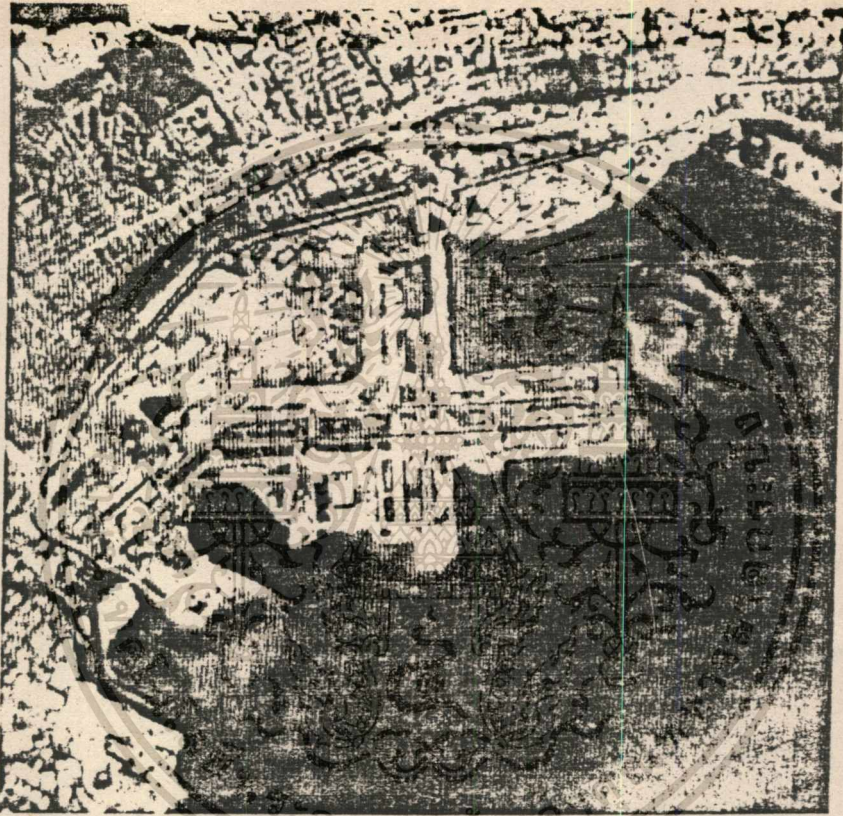
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4.2 ภาพต้นแบบขนาด  $256 \times 256$  จุด<sup>2</sup>

อัตราข้อมูลภาพ 8 บิตต่อ 1 จุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4.3 ภาพที่สร้างขึ้นจากการถอดรหัส

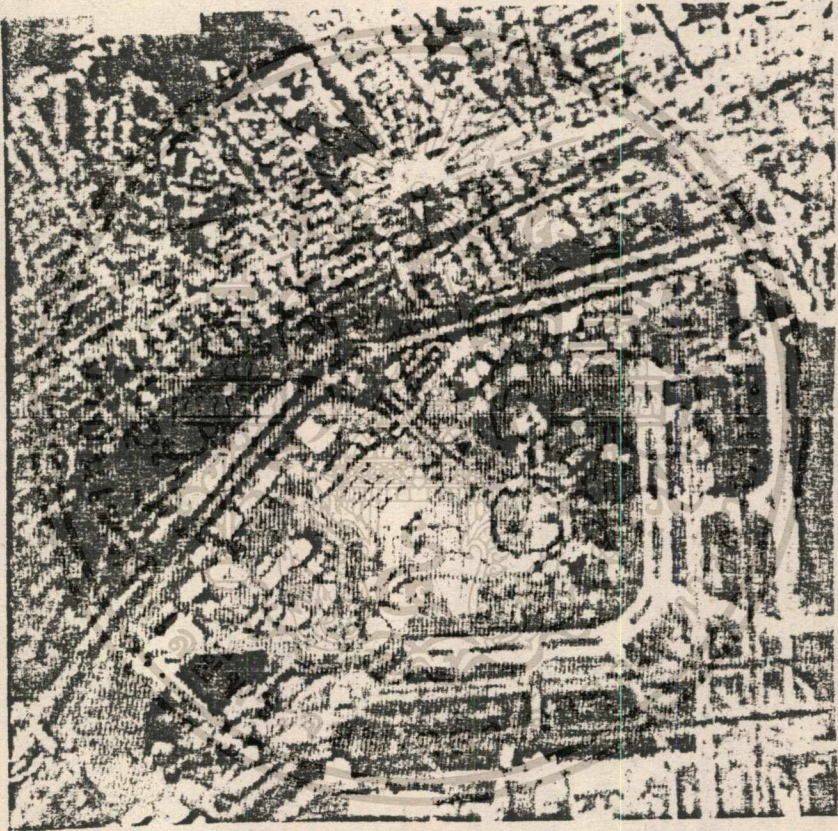
อัตราข้อมูล 1.63 บิตต่อ 1 จุด

### ความสามารถในการลดข้อมูลด้วยวิถึบล็อกทรนเคชั่น

ด้วยวิธีของบล็อกทรนเคชั่นนี้ เรายังสามารถลดข้อมูลลงให้มีอัตราข้อมูลน้อยกว่า 2 บิตต่อ 1 จุด กล่าวคือ ค่า  $\bar{r}$  และ  $\bar{\sigma}$  สามารถแทนได้ด้วยข้อมูลขนาด 6 และ 4 บิตตามลำดับ ซึ่งจะไม่ทำให้ภาพหลังการถอดรหัสแตกต่างจากภาพเดิมมากนัก หมายความว่า อัตราข้อมูลต่อ 1 จุดภาพ จะเหลือเพียง 1.63 บิตต่อ 1 จุด ดังแสดงใน รูป 3.4.3 นอกจากนี้แล้วบล็อกที่มีค่า  $\bar{\sigma} = 0$  แล้ว จะมีค่าของ A และ B เท่า กันดังสมการที่ 3.4.4 และ 3.4.5 ซึ่งทำให้ส่วนของบิตเฟลน ถูกตัดทิ้งไปได้ นั่นคือ อัตราของข้อมูล ต่อ 1 จุดภาพ จะลดลงเป็นลำดับ และ สุดท้าย คือการกำหนดขนาดของ บล็อกที่เหมาะสม ยังช่วยให้อัตราข้อมูลลดลงด้วย เช่น ภาพขนาด  $256 \times 256$  จุด  $256^2$  ระดับสี ขนาดของบล็อก คือ  $4 \times 4$  จุด  $4^2$  จะมีอัตราข้อมูล 2 บิตต่อ 1 จุด ถ้าขนาดของ บล็อก คือ  $8 \times 8$  จุด  $8^2$  จะมีอัตราข้อมูล 1.25 บิตต่อ 1 จุด แต่ก็จะต้องคำนึงถึงค่าความ คลาดเคลื่อนของภาพ หลังการถอดรหัสด้วย เพราะยิ่งบล็อกมีขนาดใหญ่ ค่าความคลาดเคลื่อน จะยิ่งมากขึ้นด้วย

ขนาดของบล็อก	อัตราของข้อมูลภาพต่อ 1 จุดภาพ	ปริมาณการลดข้อมูล(%)
2 x 2	3 บิตต่อ 1 จุด	66.67
4 x 4	2 บิตต่อ 1 จุด	75
8 x 8	1.25 บิตต่อ 1 จุด	84.375
16 x 16	1.0625 บิตต่อ 1 จุด	86.72

เมื่อกำหนดขนาดของบล็อกต่างๆ กัน



รูปที่ 3.4.5 แสดงภาพหลังการถอดรหัสบล็อกทรูเนชัน  
อัตราข้อมูลภาพ บิตต่อ 1 จุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5 การลดข้อมูลภาพด้วยวิธีเคแอล

(Karhunen-Loeve Compression)

#### 3.5.1 การลดข้อมูลภาพด้วยวิธีเคแอล สำหรับกรณีข้อมูลภาพต่อเนื่อง

ใช้ลดข้อมูลภาพในกรณีที่เรารู้ลำดับข้อมูลภาพ (priori) ที่เก็บไว้ ซึ่งเป็นลักษณะซิงูซoidal (Sinusoidal Pattern) ภาพทั่วไปจะแสดงลักษณะสำคัญแตกต่างกันไป ค่าเกรย์เลเวล (gray level) ของจุดทุกจุดในแต่ละภาพมีความสัมพันธ์กัน เราจึงทำการลดข้อมูลภาพโดยทำการทรานส์ฟอร์มภาพเก็บไว้ ข้อมูลที่เหมือนกันจะเก็บบางส่วนไว้ การนำภาพกลับคืนมาโดยการทำนายค่าเกรย์เลเวลของจุดที่เหมือนกัน โดยให้ค่าความถี่, เฟส, ความกลมกลืนของภาพ และแอมพลิจูดของภาพใกล้เคียงกับภาพเดิม

ภาพเดิมข้อมูลภาพมีความสัมพันธ์กัน (correlated data) แต่ภาพที่ได้จากการทรานส์ฟอร์ม (transform) กลับมาจะมีข้อมูลไม่สัมพันธ์กัน (uncorrelated data) ข้อมูลที่ไม่สัมพันธ์กันจะถูกจัดเป็นแถวในลำดับความสำคัญ ในตัวอย่างต่อไปจะแสดงข้อมูลเป็นลำดับจากสูงมาต่ำ การลดข้อมูลภาพวิธีนี้ทำได้โดยใช้คณิตศาสตร์ เริ่มจากการสุ่มข้อมูลที่ไม่สัมพันธ์กัน จากภาพที่ให้ตามทฤษฎีต่อไปนี้

ทฤษฎีเคแอลทรานส์ฟอร์มเมชัน สำหรับภาพต่อเนื่อง

ให้  $-A/2 \leq X \leq A/2$ ,  $-B/2 \leq Y \leq B/2$  บ่งขอบเขต  $L$  บนระนาบ  $XY$   $(x, y)$  เป็นกลุ่มของออร์ทอนอรัล ฟังก์ชัน (Orthornormal function) ที่ครอบคลุมพื้นที่  $L$  แรนดอมฟิลด์ (random field) ที่อยู่ในขอบเขตของ  $L$  แสดงดังสมการ

$$f(x, y) = \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} a_{mn} \psi_{mn}(x, y) \quad (3.5.1)$$

โดย 
$$a_{mn} = \int_{-B/2}^{B/2} \int_{-A/2}^{A/2} f(x, y) \psi_{mn}^*(x, y) dx dy \quad (3.5.2)$$

ค่า  $\psi_{mn}(x, y)$  เป็นผลจากการสุ่มค่าที่ไม่สัมพันธ์กัน  $a_{mn}$  ต้องสอดคล้องกับสมการ

$$\int_{-B/2}^{B/2} \int_{-A/2}^{A/2} R(x, y, x', y') \psi_{mn}(x', y') dx dy = \gamma_{mn} \psi_{mn}(x, y) \quad (3.5.3)$$

สำหรับ  $-A/2 \leq x \leq A/2$ ,  $-B/2 \leq y \leq B/2$  ที่ซึ่ง

$$\gamma_{mn} = E\{|a_{mn}|^2\} \quad (3.5.4)$$

จะเห็นว่าเราจะแสดงภาพต่อเนื่องโดยใช้ข้อมูลที่ไม่สัมพันธ์กัน ทำได้โดยใช้ฮาดแวร์

แต่ที่นิยมทำกันคือ หลังจากทำการดิฟิโวลชันโดยการสุ่มเป็นตารางได้ข้อมูลภาพ แล้วนำมา

ทรานส์ฟอร์มเป็นข้อมูลภาพที่ไม่สัมพันธ์กันเก็บไว้ ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.2 การลดข้อมูลภาพด้วยวิธีเคนเอล สำหรับภาพไม่ต่อเนื่อง

ข้อมูลภาพจะอยู่ในรูปแรนดอมฟิลด์  $f(x, y)$  ถูกสุ่มบนภาพ ซึ่งแบ่งเป็น  $N \times N$  ช่อง ค่าที่สุ่มนี้จะเขียนแทนด้วย  $f(m, n)$  โดย  $m, n$  มีค่าตั้งแต่ 0 ถึง  $N-1$  ได้เมตริกซ์  $[f]$  จะได้

$[f]$  เป็นข้อมูลภาพเดิม ข้อมูลที่ไม่สัมพันธ์กัน เป็นข้อมูลภาพที่เราเก็บไว้ และ  $[f]$  ได้จากการอินเวอร์สทรานส์ฟอร์ม ข้อมูลที่ไม่สัมพันธ์กัน

ให้  $[P]$  และ  $[Q]$  เป็น ดีเทอร์มิแนนติกอนซิงกูลาร์เมตริกซ์ (deterministic nonsingular matrix) ขนาด  $N \times N$  ของ  $[f]$  จะได้

$$[F] = [P][f][Q] \quad (3.5.5)$$

อินเวอร์สทรานส์ฟอร์มคือ

$$[f] = [P]^{-1} [F][Q]^{-1} \quad (3.5.6)$$

ให้  $[P]^{-1} = [P']$  และ  $[Q]^{-1} = [Q']$  จะได้

$$f(m, n) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u, v) P'(m, u) Q'(v, n) \quad (3.5.7)$$

สำหรับ  $m = 0, \dots, N-1$

$$f(m, n) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} [\psi^{(u, v)}] F(u, v) \quad (3.5.8)$$

ที่ซึ่ง  $[\psi^{(u, v)}]$  เป็น  $N \times N$  เมตริกซ์ ซึ่ง สมาชิก  $(m, n)$  มาจาก  $P'(m, u) Q'(v, n)$

เราให้ เมตริกซ์  $N \times N$   $[\psi^{(u, v)}]$  อยู่ฟอร์มออร์ทอนอมอล จะได้

$$[\psi^{(u, v)}] [\psi^{(r, s)}] = \begin{cases} 0, & u \neq r \text{ หรือ } v \neq s \\ 1, & u = r \text{ และ } v = s \end{cases} \quad (3.5.9)$$

คูณทั้งสองข้างด้วย  $[\psi^{(r, s)}]$  ได้

$$[f] * [\psi^{(r, s)}] = F(r, s) \quad (3.5.10)$$

สรุปว่า ให้กลุ่มของออร์ทอนอมอลเมตริกซ์  $[\psi^{(u, v)}]$  มา เราขยาย  $[f]$  ได้เป็น

$$[f] = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u, v) [\psi^{(u, v)}] \quad (3.5.11)$$

ซึ่งสัมประสิทธิ์ของการขยาย  $F(u, v)$  คือ

$$F(u, v) = [f] * [\psi^{(u, v)}] \quad (3.5.12)$$

สมการที่ (11) และ (11b) สะดวกสำหรับการทรานส์ฟอร์ม  $[f]$  สู่มเมตริกซ์  $[F]$  ถ้าไม่ว่าให้  $[F]$  ก็สามารรถหา  $[f]$  ได้โดยใช้สมการที่ (11a) สำหรับการบีบภาพไม่ต่อเนื่อง การขยายใน

สมการที่ 11 จะเกี่ยวข้องกับจำนวนเทอมเท่านั้น

ทฤษฎีเคออสกราฟานลเฟอร์เมชัน สำหรับกรณีภาพไม่ต่อเนื่อง

ให้  $R(m, n, p, q)$  เป็นออโตโครีเลชันฟังก์ชัน (autocorrelation function) ของ

$[f]$  คือ

$$R(m, n, p, q) = E\{f(m, n)f(p, q)\} \quad (3.5.13)$$

ออร์ทอนอโมลเมตริกซ์  $[\psi^{u,v}]$  ซึ่งมีผลใน  $F(u, v)$  ในสมการที่ 3.5.11 จะได้

$$\sum_{p=0}^{N-1} \sum_{q=0}^{N-1} R(m, n, p, q) \psi^{(u,v)}(p, q) = \gamma_{uv} \psi^{(u,v)}(m, n) \quad (3.5.14)$$

ที่  $[\psi^{u,v}]$  และ  $[\psi^{m,n}]$  มีสมาชิกที่  $(p, q)$  และ  $(m, n)$  ของ  $[\psi^{u,v}]$

ตามลำดับ ซึ่ง

$$\gamma_{uv} = E\{|F(u, v)|^2\} \quad (3.5.15)$$

เมตริกซ์  $[\psi^{u,v}]$  เรียกว่า โอเกนเมตริกซ์ (eigenmatrices) หรือ เมลิส

เมตริกซ์ของ  $R(m, n, p, q)$

การนำการลดข้อมูลด้วยวิธี เคออสมาประยุกต์ใช้กับภาพ

สำหรับภาพที่จะเป็นตัวอย่าง แสดงดังรูปที่ 3.5.1

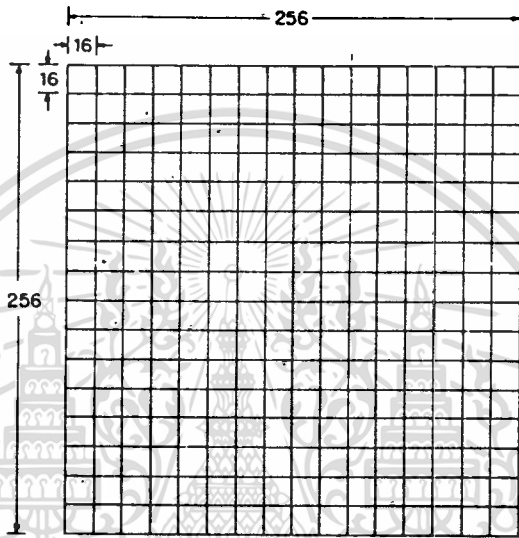


รูป 3.5.1 ภาพขนาด  $256 \times 256$  จุด<sup>2</sup> แต่ละจุดของภาพใช้ 8 บิต

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติเห็นไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่มีข้อมูลเป็นเมตริกซ์ขนาด  $256 \times 256$  แต่ละจุดใช้ 8 บิต 1 ภาพใช้บิตทั้งหมดจำนวน  $2^8 \times 2^8 \times 2^8 = 2^{24}$  บิต เราทำการแบ่งภาพเป็นภาพย่อย ๆ จำนวน  $16 \times 16$  ภาพเพื่อสะดวกในการคำนวณ ดังรูปที่ 3.5.2



รูป 3.5.2 แบ่งภาพใหญ่เป็นภาพย่อย  $16 \times 16$  ภาพย่อย

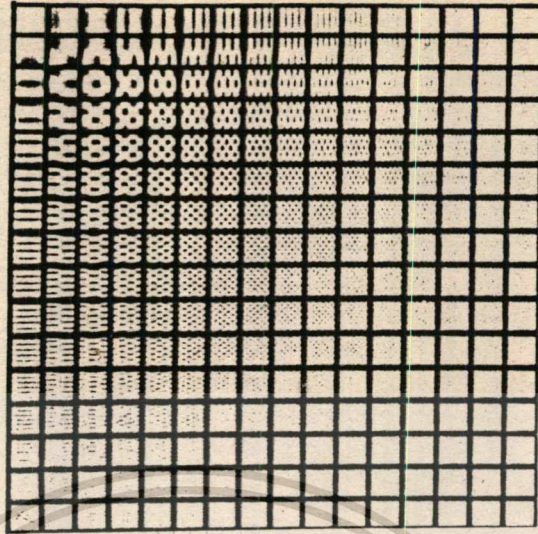
แต่ละภาพเล็กจะถูกทราบสี่ฟอร์มเป็นข้อมูลที่ไม่วัดกัน โดยการหาไอเกนเวกเตอร์และสำหรับค่า  $[K]$  จะมีค่าเหมือนกันสำหรับแต่ละภาพย่อย เพื่อใช้ในการอินเวอร์สเมตริกซ์  $256 \times 256$  จุด

ข้อมูลในภาพที่ 3.5.1 อยู่ในรูปย่อยได้โครีเลขัน ซึ่งออกแบบจากสมการ

$$E(x, y, x', y') = 0.125|x-x'| - 0.249|y-y'| \quad (3.5.16)$$

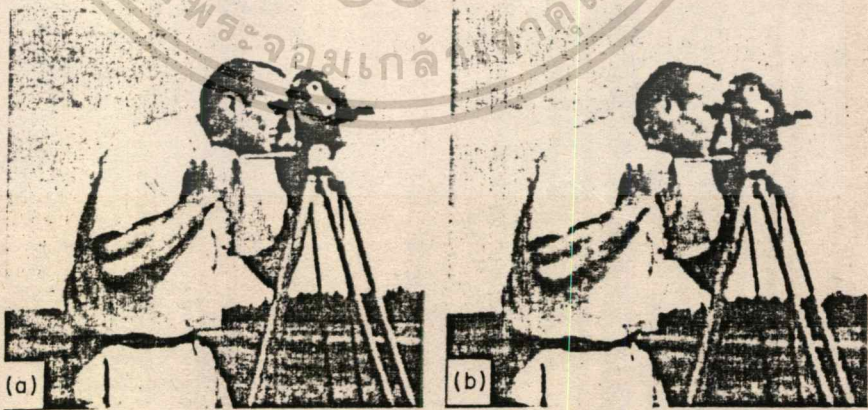
คำตอบของสมการที่ (3) สำหรับอโต้โครีเลขัน เป็นคำตอบสำหรับภาพย่อย ๆ (ซึ่งเราให้จุดศูนย์กลางอยู่ที่  $(0,0)$  เป็น  $\psi_{uv}(x,y)$  โดย  $u$  และ  $v$  เป็นจำนวนเต็มบวก คำตอบเหล่านี้ถูกจัดเป็นแถวตามค่าไอเกนวาลิวส์ (eigenvalues)

ถ้าแต่ละออร์ทอนอมอลฟังก์ชัน  $\psi_{uv}(x,y)$  ถูกสุ่มจากภาพย่อย  $16 \times 16$  และผลของเมตริกซ์เป็น  $\psi^{(u,v)}$  โดย  $\psi^{(u,v)}$  จะถูกจัดเป็นลำดับ ตั้งแต่  $\psi^{(0,0)}$  อยู่ที่มุมซ้ายบนในภาพที่ 3.5.3 เมตริกซ์  $\psi^{(0,0)}$  เป็นช่องที่ 2 จากแถวบนซ้าย



รูป 3.5.3 ภาพย่อย 16 x 16 ภาพ เรียงลำดับ หลังจากผ่านเคแอลทรานส์ฟอร์ม

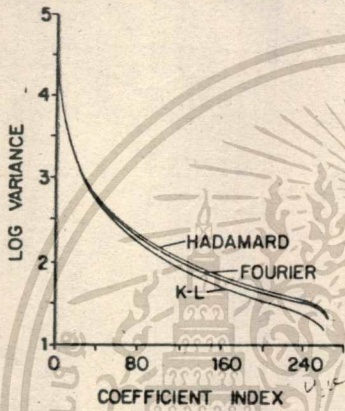
เมื่อหา  $\varphi^{(u,v)}$  ได้ก็หาสัมประสิทธิ์ที่ไม่สัมพันธ์กัน  $F(u,v)$  สำหรับแต่ละภาพย่อยได้ โดยใช้สมการ 3.5.11 ความสัมพันธ์กันระหว่างค่าสัมประสิทธิ์  $F(u,v)$  จะน้อยกว่าความสัมพันธ์ระหว่าง 256 จุด ของภาพย่อย ตามสมการ 3.5.11 สำหรับแต่ละภาพย่อยจะเปลี่ยนเป็นสัมประสิทธิ์  $F(u,v)$  256 ค่า ภาพย่อยเหล่านี้จะถูกนำกลับคืนมาได้จากค่าสัมประสิทธิ์เหล่านี้ โดยใช้สมการ 3.5.10



รูป 3.5.4 a) ภาพที่ได้กลับคืนมาโดยการ เก็บค่าสัมประสิทธิ์ เคแอลทรานส์ฟอร์ม 128 ค่าจากทั้งหมด 256 ค่า สำหรับแต่ละภาพย่อย

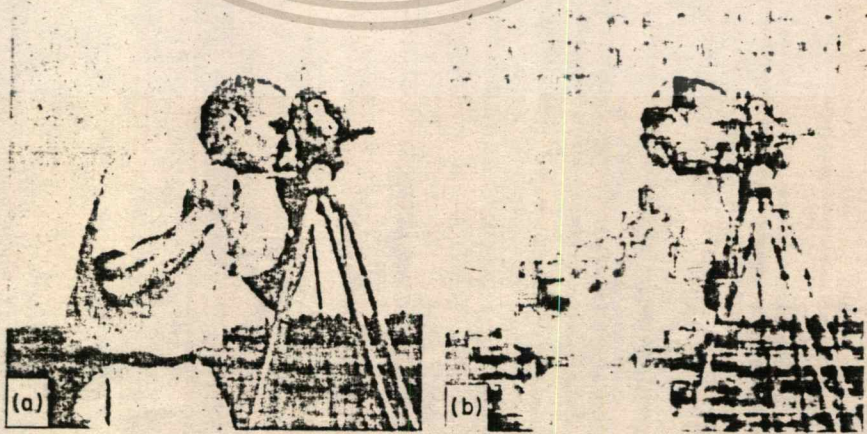
เอกสารนี้เป็นเอกสาร b) ภาพที่ได้กลับคืนมาโดยการ เก็บค่าสัมประสิทธิ์ เคแอลทรานส์ฟอร์ม 64 ค่าจาก...  
ไม่ว่ากรณีใดๆทั้งสิ้น อีทั้งหมด 256 ค่า สำหรับแต่ละภาพย่อย ถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราเก็บค่า  $F(u,v)$  256 ค่าสำหรับแต่ละภาพย่อย จะสามารถลดข้อมูลภาพได้มากขึ้น ถ้าเราเก็บเพียง 128 ค่าโดยการสุ่ม โดยค่าเหล่านี้จะถูกจัดเป็นลำดับตามไอเกนวาลู เมื่อเราใช้สมการ 3.5.10 ทราન્ส์ฟอร์มภาพภาพกลับคืนมาจะได้ภาพดังรูปที่ 3.5.4 a) จะเห็นว่าการเก็บค่าสัมประสิทธิ์เพียงครึ่งเดียว สำหรับแต่ละภาพย่อยยังทำให้ภาพใกล้เคียงกับภาพเดิมมาก ส่วนภาพในรูป 3.5.4 b) ได้การนำภาพกลับคืนมาโดยใช้เพียงสัมประสิทธิ์ 64 ค่า สำหรับแต่ละภาพย่อย ซึ่งความละเอียดของภาพลดลง



รูป 3.5.5 แสดงค่าความแปรปรวนของสัมประสิทธิ์  $F(u,v)$  แสดงในรูป  $|F(u,v)|^2$  สำหรับภาพย่อยทั้งหมดในรูป 3.5.1 โดย  $E\{F(u,v)\} = 0$  สำหรับภาพซีโรมีน (zero-mean picture)

ถ้าเก็บสัมประสิทธิ์ 128 ค่า จาก 256 ค่าสำหรับแต่ละภาพย่อย เราทำให้เป็นมาตรฐานด้วยค่าความแปรปรวนและทำการควอนไทเซชัน เป็นสัมประสิทธิ์มาตรฐานเป็น 16 ระดับสี (4 บิต) ใน 1 ภาพต้องการจำนวนบิตทั้งหมด  $= 256 \times 128 \times 4 = 2^{17}$  จะเห็นว่าข้อมูลภาพลดลงเหลือ 2 บิต/จุด แต่ละภาพย่อยถูกนำกลับคืนมาโดยใช้สมการ 3.5.10 ได้ภาพที่ 3.5.6 a)



รูป 3.5.6 a) ภาพที่ได้กลับคืนมาโดยการควอนไทซ์ภาพที่ใช้สัมประสิทธิ์ เคแอลทรานส์ฟอร์ม

เอกสารนี้เป็นเอกสารที่สวมน 128 ค่า ภาพ 16 ระดับ (4 บิต) นั้น ไม่นูนญาติให้นำไปใช้ประโยชน์ด้านการคำนวณว่ากรณีใดๆทั้งสิ้น ลึกทั้งหน่วยบิตต่อจุด และข้ออ้างถึงถึงแล้วคงเอกสารทุกครั้งที่มีการนำไปใช้

b) ภาพเดียวกันแต่ใช้การควอนไทซ์ภาพ 4 ระดับ

เราสามารถลดข้อมูลภาพได้อีก โดยการควอนไทเซชันภาพเป็นความเข้ม 4 ระดับ ( 2 บิต) ข้อมูลภาพก็จะลดลงเหลือเพียง 1 บิต/pixel ภาพที่กลับคืนมาแสดงดังรูปที่ 3.5.6 b)

### 3.6 การลดข้อมูลภาพด้วยฟูเรียร์ทรานส์ฟอร์ม

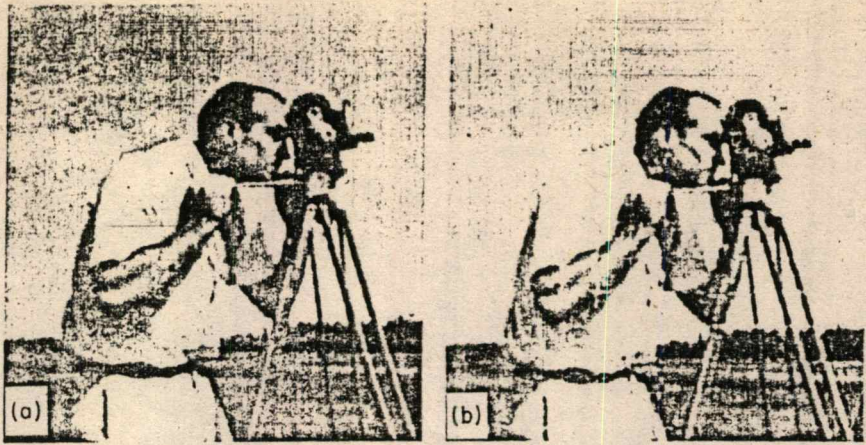
(FOURIER TRANSFORM COMPRESSION)

ฟูเรียร์ทรานส์ฟอร์มที่แสดงในสมการ 3.5.10 และ 3.5.11 ทำให้แสดง  $\psi^{(u,v)}$  ง่าย สำหรับฟูเรียร์ทรานส์ฟอร์มได้เป็น

$$\psi^{(u,v)}(m,n) = (1/N) \exp [j2\pi/N (mu+nv)] \quad (3.6.1)$$

ฟูเรียร์ทรานส์ฟอร์มเป็นการใช้สัมประสิทธิ์  $F(u,v)$  ซึ่งมีสมมูล (equivalent) กับเคแอลทรานส์ฟอร์ม จากสมการที่ (3.5.8) ถ้าค่า  $N$  ใกล้เคียงอนันต์ (infinity) สัมประสิทธิ์ของฟูเรียร์จะไม่สัมพันธ์กัน การใช้ฟูเรียร์ทรานส์ฟอร์มจะสามารถลดข้อมูลภาพได้ใกล้เคียงกับเคแอลทรานส์ฟอร์ม

สำหรับตัวอย่างการใช้ฟูเรียร์ทรานส์ฟอร์ม จะพิจารณาภาพไม่ต่อเนื่องในรูปที่ 3.5.1 ภาพนี้จะถูกแบ่งเป็น  $16 \times 16$  ภาพย่อย เพื่อสะดวกในการใช้วีเอ็มเอริคอลล ฟูเรียร์ทรานส์ฟอร์มจะทรานส์ฟอร์มภาพย่อยเป็นสัมประสิทธิ์ฟูเรียร์ 256 ค่า ความแปรปรวนของแต่ละค่าสัมประสิทธิ์ฟูเรียร์  $F(u,v)$  ถูกประมาณโดยค่าเฉลี่ย  $|F(u,v)|^2$  ของภาพทั้งหมด 256 สัมประสิทธิ์ฟูเรียร์ในส่วนจินตภาพของแต่ละภาพย่อยถูกจัดเป็นลำดับด้วยความแปรปรวนดังรูปที่ 3.5.5 สัมประสิทธิ์ฟูเรียร์ส่วนจินตภาพ 256 ค่านี้ เราจะเก็บเพียง 128 ค่า เพราะมีเพียง 128 ค่าเท่านั้นที่แตกต่างกันชัดเจน ถ้าเรามีค่าสัมประสิทธิ์ฟูเรียร์ส่วนจินตภาพ 128 ค่า จะมี 64 ค่าเท่านั้นที่แตกต่างกัน ค่าสัมประสิทธิ์ฟูเรียร์ส่วนจินตภาพ รวมกับค่าสัมประสิทธิ์ฟูเรียร์ในส่วนจริง 128 ค่า ก็จะได้ภาพย่อย ๆ 1 ภาพ



รูป 3.6.1 a) ภาพที่ได้กลับคืนมาจากการใช้ 128 สัมประสิทธิ์ฟูเรียร์ ของทั้งหมด 256  
 b) เป็นภาพที่ได้จากค่า 64 สัมประสิทธิ์ฟูเรียร์

สำหรับภาพย่อย ๆ แต่ละภาพถูกสร้างกลับมาจากค่าสัมประสิทธิ์ฟูเรียร์ 128 ค่า โดยการใช้สมการ 3.5.10 และ 3.5.11 ดังแสดงในรูป 3.6.1 a) ถ้าเก็บค่าสัมประสิทธิ์ฟูเรียร์เพียง 64 ค่า ภาพที่สร้างกลับคืนมาจะเป็นดังรูปที่ 3.6.1 b) รูป 3.6.1 a) และ 3.6.1 b) มีลักษณะเหมือนกับรูป 3.5.1 a) และ 3.5.1 b) ของการใช้โคแอสตรานส์ฟอร์ม ตามลำดับ ขึ้นต่อไปคือการทำสัมประสิทธิ์ฟูเรียร์ที่เก็บไว้ให้อยู่ในรูปมาตรฐาน โดยใช้ความแปรปรวนของมัน เมื่อใช้ควอนไทเซชันและระดับของเอาท์พุทของควอนไทเซชันเป็นอีควอลเลงท์โค้ดเวิร์ด (equal length code words) ผลที่ได้จะเหมือนกับที่แสดงในรูป 3.5.7 a) และ 3.5.7 b) สำหรับการลดข้อมูลภาพที่มีประสิทธิภาพมากกว่าฟูเรียร์ทรานส์ฟอร์ม ได้แก่ การลดข้อมูลภาพด้วยวิธีฮาดามาร์ดทรานส์ฟอร์ม (HADAMARD TRANSFORM COMPRESSION)

### 3.7 การลดข้อมูลภาพด้วยวิธีฮาดามาร์ดทรานส์ฟอร์ม

(HADAMARD TRANSFORM COMPRESSION)

ฮาดามาร์ดเมตริกซ์ประกอบด้วย +1 และ -1 เท่านั้น การคำนวณของฮาดามาร์ดทรานส์ฟอร์มไม่ต้องการผลคูณ สำหรับรูปขนาด  $N \times N$  สัมประสิทธิ์ฮาดามาร์ดทรานส์ฟอร์มด้วยจำนวนเงื่อนไขหรือ การลบกับ ซึ่งคล้ายกับสัมประสิทธิ์ฟูเรียร์ทรานส์ฟอร์ม สัมประสิทธิ์ฮาดามาร์ดทรานส์ฟอร์มจะมีลักษณะไม่สัมพันธ์กันมากกว่าจุดต่าง ๆ ของภาพ

เมตริกซ์พื้นฐานสำหรับฮาดามาร์ดทรานส์ฟอร์มคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้กันเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

$$\psi^{(u,v)}(m,n) = (1/N)(-1)^{b(u,v,m,n)} \quad (3.7.1)$$

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับ  $N$  กำลังสอง ที่  $1/N$  จำเป็นสำหรับเงื่อนไข normalization จะได้

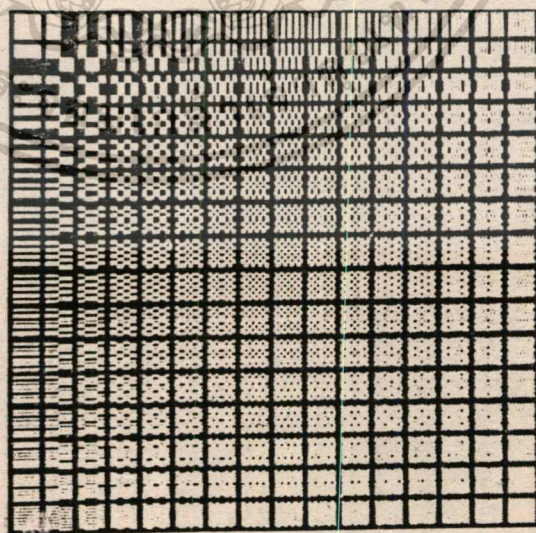
$$b(u,v,m,n) = \sum_{h=0}^{\log_2 N-1} [b_h(u)b_h(v) + b_h(m)b_h(n)] \quad (3.7.2)$$

เมื่อแทนสมการ (17) ใน (20b) จะได้

$$F(u,v) = (1/N) \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} f(m,n) * (-1)^{b(u,v,m,n)} \quad (3.7.3)$$

ในรูปที่ 3.7.1 เรามี 256 ฮาดามาร์ดเมตริกซ์,  $[ \psi^{(u,v)} ]$  แต่ละ  $16 \times 16$  ภาพย่อย สำหรับ  $u = 0, 1, \dots, 15$  และ  $v = 0, 1, \dots, 15$

ตัวอย่างของการลดข้อมูลภาพโดยใช้ฮาดามาร์ดทรานส์ฟอร์ม แต่ละ  $16 \times 16$  ภาพย่อยของรูป 3.5.1 ถูกทรานส์ฟอร์มเป็น 256 สัมประสิทธิ์ฮาดามาร์ด  $F(u,v)$  ความแปรปรวนของแต่ละสัมประสิทธิ์ถูกประมาณโดยค่าเฉลี่ยของแต่ละ  $|F(u,v)|^2$  ของรูปย่อยทั้งหมด สัมประสิทธิ์ในแต่ละภาพย่อย จัดเป็นแถวตามค่าความแปรปรวน (รูป 3.5.5) และ 128 สัมประสิทธิ์ที่เก็บไว้ เมื่อเราเ้าภาพกลับมา แต่ละภาพย่อยจาก 128 สัมประสิทธิ์ที่เก็บไว้ โดยการใชสมการ 3.5.10 และ 3.7.1 ผลแสดงดังรูป 3.7.2 ในเคแอลทรานส์ฟอร์มและฟูเรียร์ทรานส์ฟอร์มการลบสัมประสิทธิ์ลงครึ่งหนึ่งในแต่ละภาพย่อยไม่ได้มีผลต่อภาพ

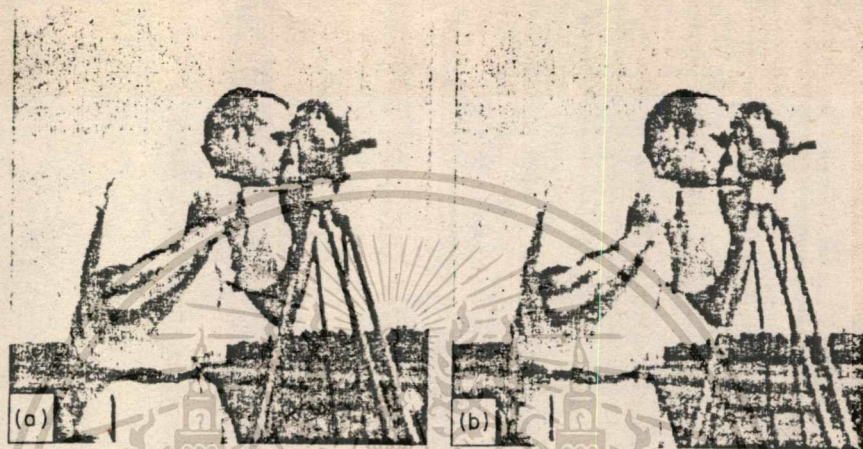


รูป 3.7.1 แสดงภาพย่อย  $16 \times 16$  ภาพย่อย ซึ่งมีการเรียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษา **ลำดับตามฮาดามาร์ดทรานส์ฟอร์ม** ญาติให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราเก็บสัมประสิทธิ์การทราสเฟอร์มี 64 ค่าแรกเท่านั้น สำหรับแต่ละภาพย่อย การนำภาพกลับมาคล้ายรูป 3.7.2 b) และความละเอียดภาพยังชัดเจน ถ้าเทคนิคการควอนไทซ์เหมือนกัน ผลที่ได้ใกล้เคียงกันมากในกรณีเคแอลสทราสเฟอร์มีและฟูเรียร์ทราสเฟอร์มี



รูป 3.7.2 a) ภาพที่ได้กลับคืนมาจากสัมประสิทธิ์ 128 ค่าจากทั้งหมด 256 สัมประสิทธิ์ฮาดามาร์ด  
b) ภาพที่ใช้สัมประสิทธิ์ฮาดามาร์ด 64 ค่า

### 3.8 การลดข้อมูลภาพด้วยวิธีดิสกรีตโคซายน์ทรานส์ฟอร์ม

(Discrete cosine transform compression)

จะอธิบายการทรานส์ฟอร์มภาพที่น่าสนใจ โดย ดิสกรีตโคซายน์ทรานส์ฟอร์ม

ของข้อมูล  $N$  ตัว ซึ่งถูกกำหนดโดย

$$F(u) = \frac{2c(u)}{N} \sum_{m=0}^{N-1} f(m) \cos \frac{(2m+1)u\pi}{2N}, \quad u = 0, 1, \dots, N-1 \quad (3.8.1)$$

ซึ่ง  $c(u) = 1/\sqrt{2}$  เมื่อ  $u = 0$

$$1 \quad \text{เมื่อ } u = 1, \dots, N-1 \quad (3.8.2)$$

อินเวอร์สทรานส์ฟอร์มเป็น

$$f(m) = \sum_{u=0}^{N-1} c(u) F(u) \cos \frac{(2m+1)u\pi}{2N}, \quad m = 0, 1, \dots, N-1 \quad (3.8.3)$$

ดิสกรีตโคซายน์ทรานส์ฟอร์ม สำหรับ 2 มิติ เป็นดังสมการ

$$F(u, v) = \frac{4c(u, v)}{N^2} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} f(m, n) \cos \frac{(2m+1)u\pi}{2N} \cos \frac{(2n+1)v\pi}{2N} \quad (3.8.4)$$

$u, v = 0, 1, \dots, N-1$

$$\text{ที่ซึ่ง } c(u, v) = 1/2 \quad \text{เมื่อ } u = v = 0 \quad (3.8.5)$$

$$1 \quad \text{เมื่อ } u, v = 1, 2, \dots, N-1$$

อินเวอร์สทรานส์ฟอร์มของ (3.8.4)

$$f(m, n) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} c(u, v) F(u, v) \cos \frac{(2m+1)u\pi}{2N} \cos \frac{(2n+1)v\pi}{2N} \quad (3.8.6)$$

$m, n = 0, 1, \dots, N-1$

ดิสกรีตโคซายน์ทรานส์ฟอร์มถูกนำไปใช้งานจริง โดยการใช้นาฬ 2 เท่าของ

พลาสก์ฟูเรียร์อัลกอริทึมพิจารณาการทรานส์ฟอร์ม 1 มิติ ในสมการ (3.8.1) โดยทำให้ข้อ

มูลมี  $2N$  ตัว:

$$\begin{aligned} f_p(m) &= f(m) \quad m = 0, 1, \dots, N-1 \\ &= 0 \quad m = N, N+1, \dots, 2N-1 \end{aligned} \quad (3.8.7)$$

เราสามารถเขียนสมการ ( ) ได้เป็น

$$F(u) = \frac{2c(u)}{N} \operatorname{Re} \left\{ \exp(j\pi u/2N) \sum_{m=0}^{2N-1} f_p(m) \exp(j\pi m/2N) \right\} \quad (3.8.8)$$

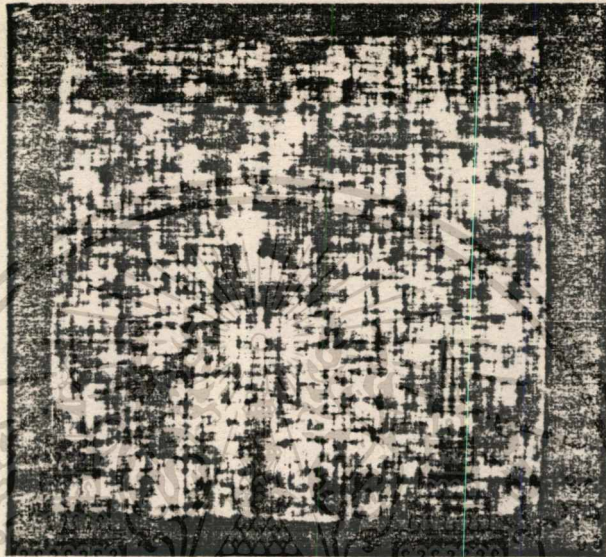
ผลรวมของดิสกรีตโคซายน์ทรานส์ฟอร์ม ซึ่งนำไปใช้งานจริงด้วย พลาสก์ฟูเรียร์

ทรานส์ฟอร์มอัลกอริทึมเร็วที่เราสามารถเขียนพลาสก์อัลกอริทึม (fast algorithm) โดยหลักการคำนวณ

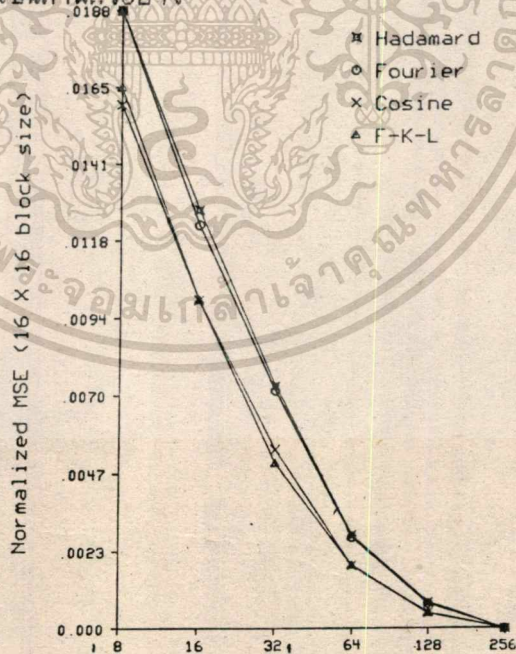
พลาสก์ฟูเรียร์ ซึ่งต้องการค่า  $N = 2^m$  เมื่อ  $m > 2$  ซึ่งต้องใช้เรกิสเตอร์เป็นจำนวนมาก

(real operation) โดยมีสมาชิก  $N$  จุด ซึ่งอัลกอริทึมนี้จะทำงานได้เร็วกว่าฟาสต์ฟูเรียร์ทรานส์ฟอร์ม

สำหรับการนำเทคนิคการทรานส์ฟอร์มต่าง ๆ ไปใช้งานกับภาพจริง ใช้การทรานส์ฟอร์มที่ต่างกัน ตัวอย่างแสดงดังรูปที่ 3.8.1



รูป 3.8.1 a) เป็นภาพตัวอย่าง



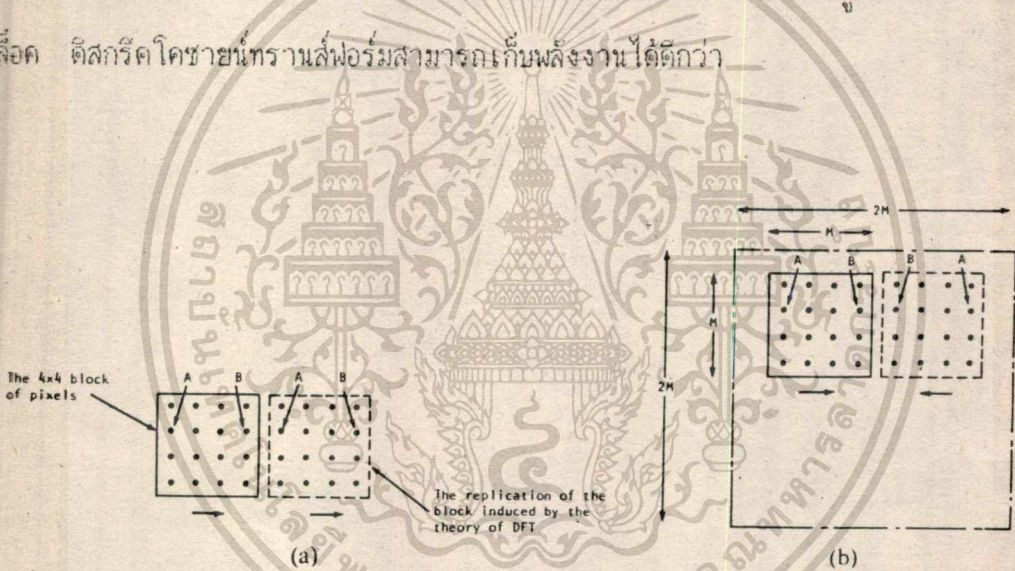
รูป 3.8.1 b) แสดงค่าเฉลี่ยความคลาดเคลื่อนกำลังสอง (integrated mean square error) ที่เกิดขึ้นจากค่าสัมประสิทธิ์ที่เก็บไว้จำนวนต่าง ๆ

กัน จะเห็นว่า ดีสกรีตโคไซน์ทรานส์ฟอร์ม มีค่าความคลาดเคลื่อนต่ำสุด และยิ่งเก็บค่าสัมประสิทธิ์ไว้จำนวนน้อย จะเกิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น มิใช่เพื่อเผยแพร่โดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ความคลาดเคลื่อนมาก ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 3.8.1 c) แสดงความคลาดเคลื่อนเฉลี่ยกำลังสองที่เกิดขึ้น เมื่อแบ่งขนาดบล็อกที่ต่าง ๆ กัน โดยภาพถูกแบ่งเป็น  $M \times M$  บล็อก แต่ละบล็อกถูกทรานส์ฟอร์มและเลือก 25% ของสัมประสิทธิ์เท่านั้นที่เก็บไว้ โดยเอาช่วงที่มีความแปรปรวนมากที่สุด จะเห็นว่าเทคนิคฟาสต์โคซายน์ทรานส์ฟอร์ม (Fast Cosine Transform) ให้ความคลาดเคลื่อนต่ำสุดและเมื่อขนาดบล็อกใหญ่ขึ้น ค่าความคลาดเคลื่อนที่เกิดขึ้นน้อยลงตามลำดับ

จากการทดลองพบว่า ดิสกรีตโคซายน์ทรานส์ฟอร์ม มีข้อได้เปรียบกว่า ดิสกรีตฟูเรียร์ทรานส์ฟอร์ม ที่น่าสนใจคือ ในการใช้งานจริงภาพขนาด  $N \times N$  จะถูกแบ่งเป็น  $M \times M$  บล็อก ดิสกรีตโคซายน์ทรานส์ฟอร์มสามารถเก็บพลังงานได้ดีกว่า



รูป 3.8.2

ในรูป 3.8.2 a) เราแสดง  $4 \times 4$  บล็อกของจุด ใช้ดิสกรีตฟูเรียร์ทรานส์ฟอร์มของบล็อก ได้บล็อกในเส้นประ ทฤษฎีของการทรานส์ฟอร์มจะทำให้บล็อกนี้ มีลักษณะซ้ำกันตามคาบเวลา จุด B จะถูกตามด้วยจุด A

รูปที่ 3.8.2 b) แสดงการใช้ดิสกรีตโคซายน์ทรานส์ฟอร์ม ซึ่งมีโครงสร้างแตกต่างจากภาพรูป 3.8.2 a) ถ้าบล็อกเดิมเป็น  $4 \times 4$  จุด<sup>2</sup> ดิสกรีตโคซายน์ทรานส์ฟอร์มแล้วจะได้บล็อกในเส้นประ ซึ่งมีลักษณะซ้ำกัน พิจารณาบล็อกใหญ่เป็นขนาด  $8 \times 8$  จุด<sup>2</sup>

ภายในประกอบด้วยบล็อกขนาด  $4 \times 4$  จุด<sup>2</sup> 4 บล็อก ซึ่งเป็นเหมือนภาพเงาของบล็อกแต่ละการคำนวณ ซึ่งจุด B จะถูกตามด้วยจุดซึ่งเหมือนกัน การตัดส่วนที่ซ้ำกันนี้ทำให้สามารถลดข้อมูลภาพใน

บล็อกได้



รูป 3.8.3 ตัวอย่างการลดข้อมูลภาพโดยการ ใช้ดีคอสกรีดโคซายน์ทรานส์ฟอร์ม สำหรับภาพ ขนาด  $256 \times 256$  จุด<sup>2</sup>

- a) ภาพที่ถูกแบ่งเป็น  $16 \times 16$  บล็อก ใช้เพียง 128 บิตประสิทธิภาพการทรานส์ฟอร์ม ด้วยค่าความแปรปรวนมากที่สุดที่เก็บไว้แต่ละบล็อก ลดอัตราข้อมูลลงเหลือ 2

บิตต่อจุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดทั้งภาพที่ได้กลับคืนมาแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างของการลดข้อมูลภาพด้วยติสกริตโคซายน์ทรานส์ฟอร์ม แสดงในรูป 3.8.3  
 $256 \times 256$  จุด<sup>2</sup> ถูกสร้างเป็นบล็อก  $16 \times 16$  จุด<sup>2</sup> 128 สัมประสิทธิ์การทรานส์ฟอร์ม  
 เท่านั้นที่มีค่าความแปรปรวนสูงถูกเก็บสำหรับแต่ละบล็อก (ความแปรปรวนของแต่ละสัมประสิทธิ์  
 ถูกประมาณโดยรวมค่าเฉลี่ยของบล็อกทั้งหมด) สัมประสิทธิ์ที่เก็บไว้ถูกทำให้มาตรฐานด้วยส่วน  
 เบี่ยงเบนมาตรฐาน และถูกควอนไทซ์ด้วย 16 ระดับ เมื่อใช้ติสกริตโคซายน์ทรานส์ฟอร์มจะ  
 ลดข้อมูลเหลือ 2 บิตต่อจุด ภาพที่ได้กลับมาจากข้อมูลที่ลดลงถูกแสดงดังรูป 3.8.2 b)



3.9 การกำหนดตำแหน่งของบิตในการทรานส์ฟอร์มเพื่อลดข้อมูลภาพ

( BIT ALLOCATION IN TRANSFORM COMPRESSION )

แต่ละลัมประสิทธิ์การทรานส์ฟอร์ม ถูกทำให้เป็นมาตรฐาน โดยค่าส่วนเบี่ยงเบนมาตรฐาน แล้วทำการควอนไทเซชัน และอีควอลเลงจ์ไบนารีโค้ดเวิร์ด ( equal length binary code word ) สุดท้ายเป็นเข้าที่พู่ของควอนไทเซอร์ ด้วยการวิจัยการกำหนดตำแหน่งของบิต ข้อมูลภาพจากเดิมใช้ 8 บิตต่อ 1 จุด ถูกลดลงเหลือ 2 บิตต่อจุด โดยให้ภาพมีความคมชัดใกล้เคียงภาพเดิม

เราใช้หลักการกำหนดบิตให้มีค่าเป็นสแควร์เออริอร์น้อยที่สุด (ค่าเฉลี่ยของมัน) สำหรับลัมประสิทธิ์ทั้งหมด ให้  $f(m,n)$   $m,n = 0,1,2,\dots,M-1$  จุดใน  $M \times M$  บล็อกของภาพ และ  $F(u,v)$   $u,v = 0,1,2,\dots,M-1$  เป็นเอกลักษณ์การทรานส์ฟอร์ม ให้  $F_q(u,v)$  แสดงค่าของลัมประสิทธิ์ที่ควอนไทซ์ และ  $f_q(m,n)$  เป็นบล็อกที่สร้างกลับมาจากลัมประสิทธิ์ที่ควอนไทซ์ ค่าเฉลี่ยมันสแควร์เออริอร์ระหว่างบล็อกเดิมและบล็อกที่สร้างกลับมา มีค่าเป็น

$$D = \frac{1}{M^2} E \sum_{m=0}^{M-1} \sum_{n=0}^{M-1} [F(m,n) - f_q(m,n)]^2 \quad (3.9.1)$$

D คือ การตัดทอน (distortion)

คุณสมบัติของเอกลักษณ์การทรานส์ฟอร์ม เขียนได้เป็น

$$D = \frac{1}{M^2} E \sum_{u=0}^{M-1} \sum_{v=0}^{M-1} [F(u,v) - F_q(u,v)]^2 \quad (3.9.2)$$

ให้ค่า  $(u,v)$  เป็นค่าความแปรปรวนของลัมประสิทธิ์การทรานส์ฟอร์มที่  $(u,v)$

ใส่ค่าความคาดหวังด้านขวาของสมการจะได้

$$E[F(u,v) - F_q(u,v)]^2 = \sigma^2(u,v) d_{u,v}(b_{u,v}) \quad (3.9.3)$$

ค่าเฉลี่ยของผลรวมความผิดพลาดเคลื่อนจากการควอนไทซ์ สำหรับบล็อกเดิม เขียนได้

เป็น

$$D = \frac{1}{M} \sum_{u=0}^{M-1} \sum_{v=0}^{M-1} \sigma^2(u,v) d_{u,v}(b_{u,v}) \quad (3.9.4)$$

การกำหนดบิต  $(b_{u,v})$  จะต้องถูกทำโดยมีการตัดทอนน้อยที่สุด โดย

$$\sum_{u=0}^{M-1} \sum_{v=0}^{M-1} b_{u,v} = m^2 b_{aver} \quad (3.9.5)$$

ในการหาค่าน้อยที่สุดของสมการ 3.9.4

โดยข้อจำกัดในสมการ

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ 3.9.5 ใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมมติ

$$d_{u,v}(b_{u,v}) = K \exp(-2b_{u,v} \ln 2) \quad (3.9.6)$$

สมการนี้เป็นการศึกษาทดสอบผลของนิวเมอริคอล (numerical) จากการคำนวณของ

ฮวง (Huang) จะได้ผลเป็น

$$b_{u,v} = b_{aver} + \frac{1}{2} \log_2 \sigma_{u,v}^2 - \frac{1}{2M^2} \sum_{u=0}^{M-1} \sum_{v=0}^{M-1} \log_2 \sigma_{u,v}^2 \quad (3.9.7)$$

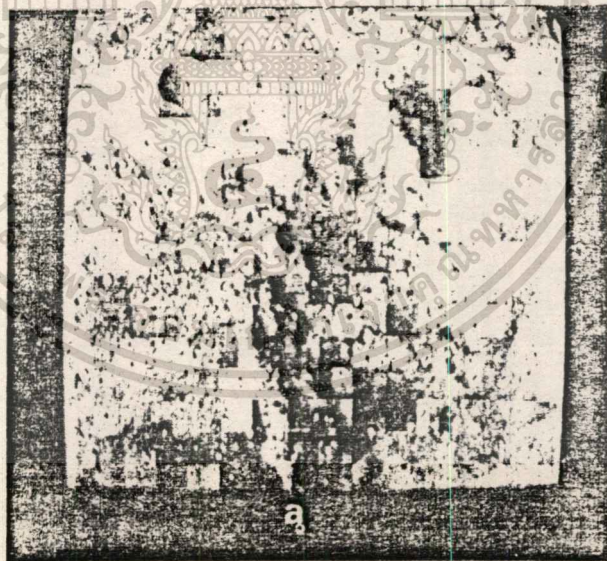
สำหรับกรณีค่าเหมาะสมสมมุติฟอร์มควอนไทเซอร์ สมมติ

$$d_{u,v}(b_{u,v}) = \exp\left(-\frac{1}{2} b_{u,v} \ln 10\right) \quad (3.9.8)$$

แทนสมการ (3.9.8) ใน 3.9.4 และทำให้ได้ค่าความคลาดเคลื่อน

น้อยที่สุด เราจะได้

$$b_{u,v} = b_{aver} + 2 / \ln 10 \left[ \ln \sigma_{u,v}^2 - \frac{1}{M^2} \sum_{u=0}^{M-1} \sum_{v=0}^{M-1} \ln \sigma_{u,v}^2 \right] \quad (3.9.9)$$



รูป 3.9.1 ก) ภาพที่ใช้ควอนไทเซอร์ 4 ระดับ ใช้สัมประสิทธิ์ที่เก็บไว้ทำให้ภาพใช้

เพียง 1 บิตต่อจุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.9.1 b) แสดงผลของบล็อกควอนไทเซชัน อัตราข้อมูลลดลงเหลือ 1 บิตต่อจุด  
สมการ 3.9.9 ถูกใช้คำนวณจำนวนบิตสำหรับแต่ละสัมประสิทธิ์  
ภาพที่ได้คมชัดขึ้น

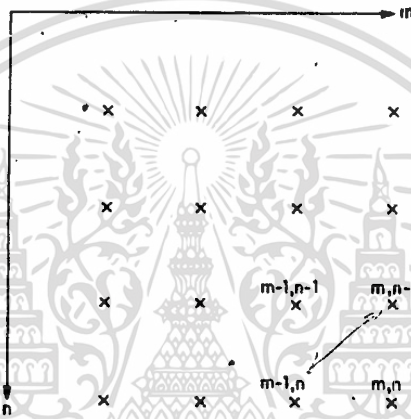
ในภาพ 3.9.1 a) เราแสดงภาพที่ถูกลดข้อมูลลงเหลือ 1 บิตต่อจุด เมื่อสัมประ  
สิทธิ์การทรานส์ฟอร์มทั้งหมดถูกทำให้เป็นมาตรฐาน โดยส่วนเบี่ยงเบนมาตรฐาน ใช้ควอนไท  
เซชัน 2 บิต มี  $16 \times 16$  บล็อก จะเก็บค่าสัมประสิทธิ์ 128 ค่าที่มีค่าความแปรปรวนมากที่สุด  
เก็บไว้ รูป 3.9.1 b) แสดงผลจากอัตราข้อมูล 1 บิตต่อจุด เมื่อการกำหนดบิตโดยสม  
การ 3.9.9 จะทำให้เทคนิคการลดข้อมูลภาพมีประสิทธิภาพมากขึ้นและภาพมีคุณภาพดีขึ้น

### 3.10 การลดข้อมูลภาพด้วยวิธีทำนาย

(PREDICTIVE COMPRESSION)

ได้กล่าวถึงการลดข้อมูลภาพโดยข้อมูลภาพซึ่งไม่สัมพันธ์กัน โดยลิเนียร์ทรานสฟอร์ม (linear transform) มาแล้ว การลดข้อมูลภาพด้วยวิธีทำนายเป็นการลดข้อมูลภาพที่ให้ผลเหมือนกัน และมีข้อได้เปรียบคือ ง่ายในการนำไปใช้งาน ซึ่งจะอธิบายต่อไป

เราพิจารณาภาพ แล้วเก็บเป็นเมตริกซ์  $[f]$  ให้  $f(m, n)$  เป็นเมตริกซ์ของจุดภาพ ซึ่งอยู่ในคอลัมน์ที่  $m$  และแถวที่  $n$  ดังรูปที่ 3.10.1



รูป 3.10.1 จุด  $(m-1, n)$ ,  $(m-1, n-1)$ , และ  $(m, n-1)$  ถูกใช้ในรูปแบบการประมาณที่จุด  $(m, n)$  ในทิศทางการเพิ่มค่า  $m$  และ  $n$

ให้  $R(m, n, p, q)$  เป็นออโต้โครีเลชันฟังก์ชันของฟิลด์ค่าจริงที่สุ่ม (real random field) ซึ่งภาพแสดงดังสมการ

$$R(m, n, p, q) = E\{f(m, n)f(p, q)\} \quad (3.10.1)$$

เพราะจุดต่างๆ ของภาพ  $f(m, n)$  สัมพันธ์กัน เราสามารถประมาณหรือทำนาย ค่าจุดต่าง ๆ ที่ถูกให้เป็น  $\hat{f}(m, n)$  จาก  $f(m, n)$  ซึ่งเป็นจุดของภาพเดิม

ความแตกต่าง  $f(m, n) - \hat{f}(m, n)$  เป็นความคลาดเคลื่อนที่ประมาณได้ ค่าที่ทำนายได้  $\hat{f}(m, n)$  ขึ้นอยู่กับความสัมพันธ์ของภาพระหว่าง  $f(m, n)$  และส่วนที่เหลือของภาพ (the rest of picture) ลำดับของตัวแปรที่สุ่มได้ถูกจัดเป็นรูปแบบโดยสัญญาณแตกต่าง  $e(m, n)$  ซึ่งจะมีความสัมพันธ์น้อยกว่าองค์ประกอบของภาพเดิม

การประมาณที่ดีที่สุดของ  $\hat{f}(m, n)$   $[\hat{f}(m, n)]$  จะดีที่สุดคือให้ค่ามีนสแควร์

เออเรอร์น้อยที่สุด] อยู่ในฟังก์ชันไม่เป็นเชิงเส้น (nonlinear function) ของจุดภาพ ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งถูกใช้เป็นแบบการประมาณ การประมาณค่าความคลาดเคลื่อนที่ดีที่สุดคือ การประมาณเชิงเส้น

เราสมมุติว่าภาพถูกสแกน (scan) ตามแถวจากแถวบนสุดสู่ล่าง จุดข้างเคียงที่ใกล้ที่สุด 3 จุด พร้อมทั้งจะถูกสแกน ทำให้กำหนดรูปแบบการประมาณเชิงเส้น สำหรับจุดต่าง ๆ ของภาพ  $f(m,n)$  ได้ และในทำนองเดียวกัน ก็กำหนด  $f(m,n)$  ได้ในรูปแบบนี้

$$\hat{f}(m,n) = a_1 f(m-1,n) + a_2 f(m-1,n-1) + a_3 f(m,n-1) \quad (3.10.2)$$

ที่ซึ่ง  $a_1, a_2, a_3$  ยังไม่ทราบค่า เป็นค่าที่ทำให้ความคลาดเคลื่อนเฉลี่ยกำลังสองมีค่าน้อยที่สุด

$$E\{[f(m,n) - \hat{f}(m,n)]^2\} \quad (3.10.3)$$

แทนค่าสมการ (3.10.2) ในสมการ (3.10.3) ทำการดิฟเฟอเรนเชียล (differentiating) ผลด้วยค่า  $a_1, a_2,$  และ  $a_3$  มีค่าใกล้ 0 และใช้สมการที่จะได้ 3 สมการคือ

$$\begin{aligned} a_1 R(m-1,n,m-1,n) + a_2 R(m-1,n-1,m-1,n) \\ + a_3 R(m,n-1,m-1,n) &= R(m,n,m-1,n) \\ a_1 R(m-1,n,m-1,n-1) + a_2 R(m-1,n-1,m-1,n-1) \\ + a_3 R(m,n-1,m-1,n-1) &= R(m,n,m-1,n-1) \\ a_1 R(m-1,n,m,m-1) + a_2 R(m-1,n-1,m,m-1) \\ + a_3 R(m,n-1,m,m-1) &= R(m,n,m,m-1) \end{aligned} \quad (3.10.4)$$

ซึ่งทำให้หาค่า  $a_1, a_2$  และ  $a_3$  ได้ ถ้าเราสมมุติแรนดอมฟิลด์ เป็นโฮโมจีเนียส (homogeneous) มีซีโรมีน (zero mean) และมีอัตราโครีเลชันฟังก์ชันให้โดยสมการ

$$\begin{aligned} a_1 R(0,0) + a_2 R(0,1) + a_3 R(1,1) &= R(1,0) \\ a_1 R(0,1) + a_2 R(0,0) + a_3 R(1,0) &= R(1,1) \\ a_1 R(1,1) + a_2 R(1,0) + a_3 R(0,0) &= R(0,1) \end{aligned} \quad (3.10.5)$$

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ที่กรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$R(\alpha, \beta) = R(0, 0) \exp(-c, |\alpha| - c |\beta|) \quad (3.10.6)$$

สำหรับข้อโต้แย้งเชิงฟังก์ชัน  $R(1, 1) = R(1, 0)R(0, 1)/R(0, 0)$

ใช้ความสัมพันธ์และแกสมการ 3.10.5 สำหรับ  $a_1, a_2$  และ  $a_3$  เราได้

$$a_1 = R(1, 0)/R(0, 0) \quad a_2 = -R(1, 1)/R(0, 0) \quad a_3 = R(0, 1)/R(0, 0)$$

สัญญาณแตกต่าง  $e(m, n)$  สำหรับแต่ละจุดของภาพเป็นดังสมการ

$$\begin{aligned} e(m, n) &= f(m, n) - f(m, n) \\ &= f(m, n) - [a_1 f(m-1, n) + a_2 f(m-1, n-1) + a_3 f(m, n-1)] \end{aligned} \quad (3.10.7)$$

ถ้าฟิล์ม  $f$  มีอีโรมีน

$$E\{e(m, n)\} = 0 \quad (3.10.8)$$

ความแปรปรวนของสัญญาณแตกต่างที่ทุก ๆ จุด คือ

$$\begin{aligned} E\{e^2(m, n)\} &= E\{[f(m, n) - [a_1 f(m-1, n) + a_2 f(m-1, n-1) \\ &\quad + a_3 f(m, n-1)]]^2\} \end{aligned} \quad (3.10.9)$$

โดยการขยายด้านขวามือของสมการ (3.10.8) และใช้สมการ 3.10.5 จะได้

$$E\{e^2(m, n)\} = R(0, 0) - [a_1 R(1, 0) + a_2 R(1, 1) + a_3 R(0, 1)] \quad (3.10.10)$$

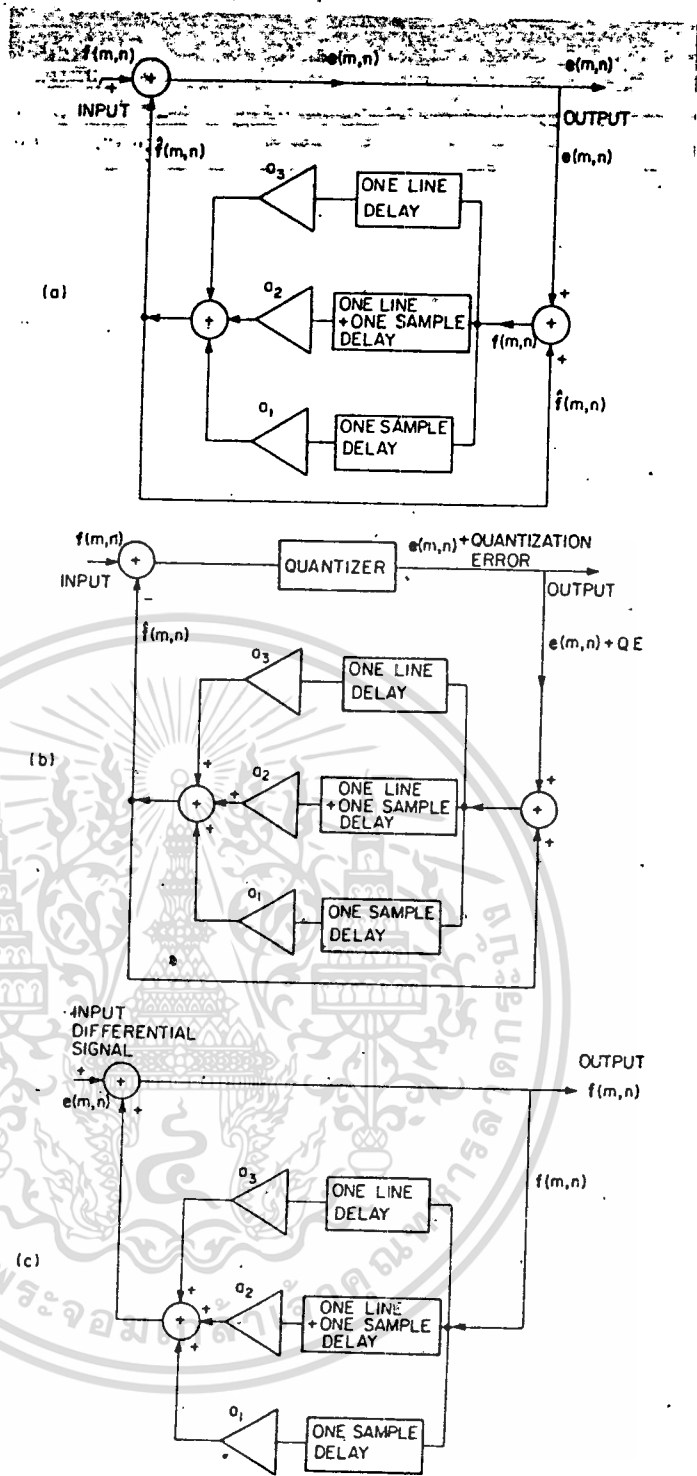
เนื่องจาก  $f$  เป็นไอโมจีเนียส ความแปรปรวนของแต่ละจุดเหมือนกัน คือ ยิ่งกว่านั้นสมการ 3.10.6 แสดงว่า  $R(1, 0), R(0, 1)$  และ  $R(1, 1)$  ต้องน้อยกว่า  $R(0, 0)$  โดยแทนค่าของ  $a_1, a_2$  และ  $a_3$  จากสมการ 3.10.7 ใน 3.10.10 ความแปรปรวนของข้อมูลที่แตกต่าง  $e(m, n)$  น้อยกว่าความแปรปรวนของจุดภาพ  $f(m, n)$

โดยการแทนค่าของ  $a_1, a_2$  และ  $a_3$  จากสมการ 3.10.6 ใน 3.10.7 แสดงค่าอีโรมีนแเรนดอมฟิล์มด้วยข้อโต้แย้งเชิงฟังก์ชัน ด้วยสมการ 3.10.5 ผลที่ตามมาคือ

$$E\{e(m, n)e(p, q)\} = 0, \quad m = p \text{ หรือ } n = q \quad (3.10.11)$$

ข้อมูลแตกต่าง  $e(m, n)$  ไม่สัมพันธ์กัน

ถ้าระดับสีเทาในภาพบนแถบมอดและซ้ายสุด (รูป 3.10.1) ภาพสามารถนำกลับ  
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่หรือนำไปใช้ในการค้า  
ไม่มาจากข้อมูลซึ่งแตกต่างห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.10.2 a) แสดงระบบที่กำเนิดข้อมูลแตกต่าง  $e(m,n)$  จากภาพที่ถูกคิ  
 จิตไรซ์ ภาพเป็นแถวซึ่งถูกสแกน และจุดของภาพถูกแสดง โดย  
 วงจรบรรทัดที่ซ้ายบนสุดของภาพ

เราควรจะมีควอนไทเซอร์ด้วยดังในรูป 3.10.2 b) แสดงถึงระบบที่มีควอนไทเซอร์

ในลูปป้อนกลับสู่ภาพ ภาพที่ได้กลับมากจะไม่ชัดเจนเนื่องจากผลของสัญญาณรบกวน (noise)

โดยควอนไทเซอร์ในรูป 3.10.2 b)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ รูป 3.10.3 c) บล็อกโคออร์ดิเนตการสร้างภาพกลับมาจากข้อมูลที่แตกต่าง การนำไปใช้

$S_1, S_2, \dots, S_N$

ในรูปแบบ

$$\hat{S}_0 = a_1 S_1 + a_2 S_2 + \dots + a_n S_n \quad (3.10.12)$$

ที่ซึ่ง  $a_1, a_2, \dots, a_n$  เป็นเมทริกซ์แควร์เออเรอร์  $E\{(S_0 - \hat{S}_0)^2$  มีค่าต่ำสุด โดยการใช้นั้นตอนการคำนวณจะได้

$$R_S(0, i) = \sum_{j=1}^n a_j R_S(i, j) \quad , \quad i = 1, 2, \dots, n \quad (3.10.13)$$

Correlations between  $S_0$  and  $S_i$  for the picture in Fig. 16a [31]

Correlations	Interlaced	Noninterlaced
$R_s(0, 1)$	0.9771	0.9866
$R_s(0, 2)$	0.9534	0.9771
$R_s(0, 3)$	0.9479	0.9570
$R_s(0, 4)$	0.9648	0.9648
$R_s(0, 5)$	0.9026	0.9026
$R_s(0, 6)$	0.8679	0.8901
$R_s(0, 7)$	0.9197	0.9479
$R_s(0, 8)$	0.8901	0.8981
$R_s(0, 9)$	0.9483	0.9573
$R_s(0, 10)$	0.9193	0.9483
$R_s(0, 11)$	0.7569	0.8927
$R_s(0, 12)$	0.7120	0.8997
$R_s(0, 13)$	0.8416	0.8463
$R_s(0, 14)$	0.8254	0.8416
$R_s(0, 15)$	0.8421	0.8828
$R_s(0, 16)$	0.8866	0.9338
$R_s(0, 17)$	0.9079	0.9591
$R_s(0, 18)$	0.8878	0.9337
$R_s(0, 19)$	0.8406	0.8800
$R_s(0, 20)$	0.8186	0.8371
$R_s(0, 21)$	0.8371	0.8439
$R_s(0, 22)$	0.8478	0.8478

ตาราง 3.10.1

ตาราง 3.10.1 แสดงค่า  $R_S(i, j)$  ซึ่งได้จากค่าเฉลี่ยของผลคูณ  $S_0 S_i$  หลังจากภาพที่ดิจิทัลได้ในรูปแบบ 3.10.3 a) ถูกทำเป็นมาตรฐานโดยการลบค่ามัน จากจุดแต่ละภาพ และหารด้วยค่าส่วนเบี่ยงเบนมาตรฐาน ทำให้  $R_S(0, 0) = 1$  เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ถ้าแรนดอมฟิลด์เป็นโอโมจีเนียส ค่า  $R_S(i, j)$  อื่น ๆ ถูกหาจาก ไม่ว่ากรณีใดๆทางสน อักทงห้ามมิให้ดัดแปลงเนื้อหา และต้งอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$R_{\beta}(\theta, i)$  รูป 3.10.3 b)  $R_{\beta}(1,7)$  ต้องเท่ากับ  $R_{\beta}(\theta, 3)$  จากข้อมูลในตารางที่ 3.10.2 ค่าที่ได้สามารถคำนวณ  $a_1$  ในสมการ 3.10.12

A List of Numerical Values of the Coefficients  $a_i$  Used for Each Predictor [31]

Predictor:	1st order horizontal	1st order vertical	2nd order	3rd order	4th order	6th order	12th order	18th order	22nd order
		0.977	0.617	0.826	0.897	0.855	0.664	0.648	0.630
					-0.059	-0.057	-0.073	-0.063	-0.081
	0.965			-0.594	-0.574	-0.534	-0.460	-0.446	-0.429
		0.379	0.746	0.729	0.827	0.834	0.831	0.831	0.865
						-0.108	-0.135	-0.131	-0.261
						0.006	0.002	0.008	0.006
							0.021	0.037	0.035
							-0.001	-0.011	0.139
							0.117	0.142	0.002
							0.051	-0.005	-0.124
							-0.125	-0.107	0.069
							0.096	0.048	0.029
								0.041	0.006
								0.001	-0.049
								-0.057	0.123
								0.112	-0.076
								-0.040	0.071
								-0.014	-0.086
									0.061
									-0.086
									0.129
									0.016

ตาราง 3.10.2 แสดงค่าของสัมประสิทธิ์  $a_i$  ที่ใช้สำหรับตัวทำนายแต่ละตัว

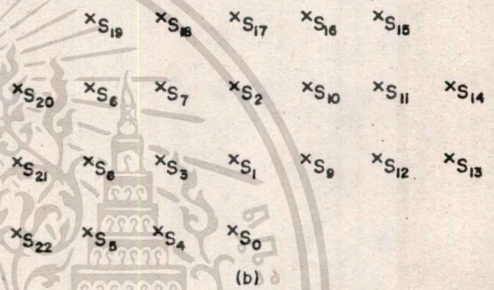
ระบบการลดข้อมูลภาพโดยการทำนาย จะเรียกว่าเป็นตัวทำนายตามแนวอนระดับที่ 1 (first-order) ถ้าจุดของภาพ  $S_{i,j}$  ถูกใช้ประมาณในสมการ 3.10.11 ซึ่งผลรวมทางขวามือในสมการ 3.10.11 มี 1 เทอมเท่านั้น  $a_1 s_{i,j}$  ตัวทำนายจะถูกเรียกว่าเป็นตัวทำนายตามแนวตั้งระดับที่ 1 ถ้า  $s_{i,j}$  ถูกใช้ประมาณ  $S_{i,j}$  ตัวทำนายระดับที่ 2  $s_{i,j}$  และ  $s_{i,j+1}$  ถูกใช้สำหรับการประมาณ สำหรับการทำนาย ขณะที่ตั้งตัวทำนาย ระดับที่ 3 ใช้  $s_{i,j}$ ,  $s_{i,j+1}$  และ  $s_{i,j+2}$  สำหรับการประมาณ สำหรับ  $n > 3$  ตัวทำนายระดับที่  $n$  ใช้  $s_{i,j}, s_{i,j+1}, \dots, s_{i,j+n-1}$  สำหรับการประมาณ

ไม่ว่ากรณีใด ตารางที่ 3.10.2 แสดงค่าคงที่  $a_i$  สำหรับตัวทำนายระดับต่าง ๆ ค่าเหล่านี้

ข้อมูลที่แตกต่างสำหรับภาพซึ่งแสดงมีขนาดเล็กกว่าข้อมูลในภาพเดิม เพราะภาพนี้  
 บรรจุบรรจุพื้นที่ของค่าคงที่ หรือระดับสีเทาที่เปลี่ยนแปลงช้า ๆ ในขอบเขตนี้ มีขนาด  
 เล็กมาก เมื่อความแตกต่างเหล่านี้เล็กน้อยจนไม่สามารถตรวจจับโดยผู้ใช้ได้ ซึ่งประมาณ  
 เป็นศูนย์



(a)

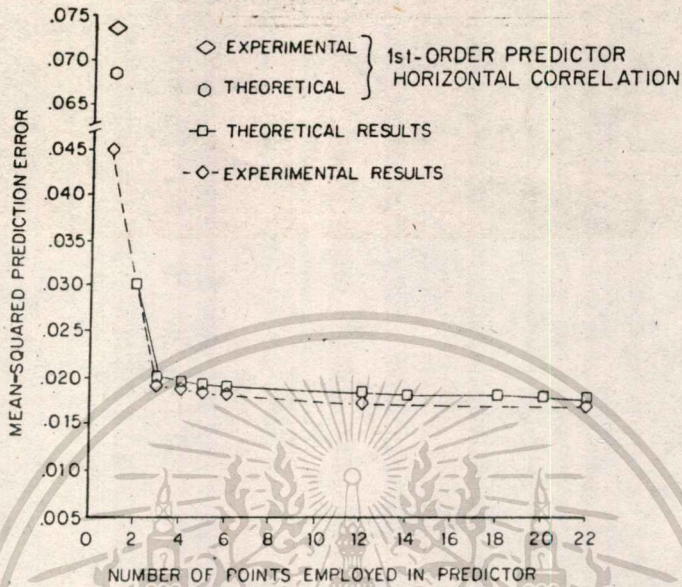


รูป 3.10.3 a) ภาพตัวอย่าง  
 b) ภาพซึ่งได้จากการสแกนตามแถว  $s_0$  แสดงการสุ่มปัจจุบัน  
 $s_1$  แสดงการสุ่มเหนือขึ้นไป 1 เส้น

ถ้าออดีรีเลขขึ้นนึ่งกัซัน ไม่สามารถถูกจำลองโดยสมการ 3.10.5 จำนวนจุด  
 ของภาพต้องการความสัมพันธ์ที่เล็กมาก ตัวอย่างพิจารณารูป 3.10.3 a) ภาพถูกสุ่มได้เมตริกซ์  
 ขนาด  $208 \times 250$  จุด เริ่มเส้นสแกนจากบน ให้  $s_0$  แสดงการสุ่มปัจจุบัน  $s_1$   
 เป็นการสุ่ม 1 เส้นบน และแสดงดังรูป 3.10.3 b) ให้  $R_{ij}(i, j)$  เป็นความคาดหวัง  
 $E(S_i, s_j)$  เราใช้สับสคริป (subscript)  $s$  ใน  $R_{ij}(i, j)$  เพื่อหลีกเลี่ยง

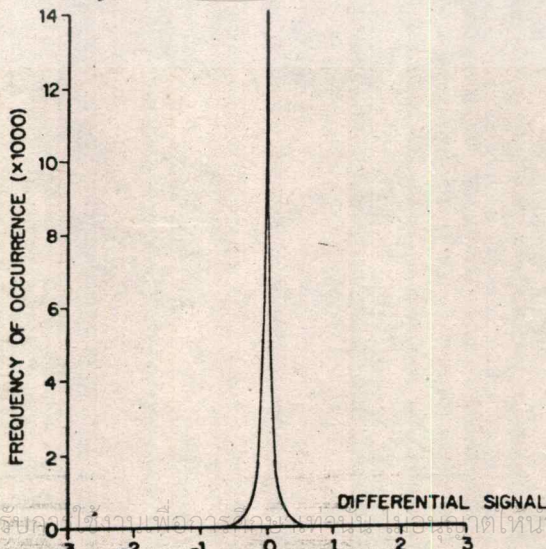
ความสับสนกับ  $R(\dots)$  ของโอโวกีเนียสแรนดอมนิสค์  
 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 การประมาณเชิงเส้น  $s_0$  ของการสุ่มปัจจุบัน  $s_0$  ในเทอมของการสุ่ม  $s_1$  ไปใช้  
 ไม่จำกัด

ถูกหาโดยการแก้สมการ 3.10.12 สำหรับตัวทำนายแต่ละตัว



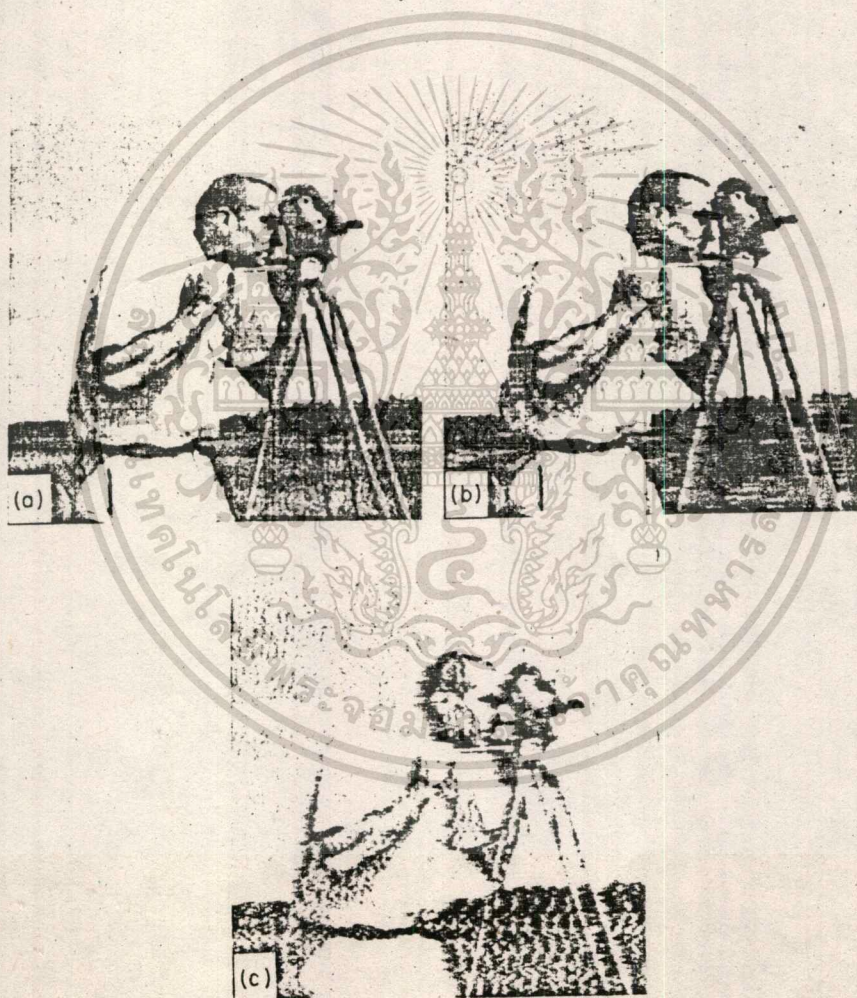
รูป 3.10.4 ค่าความคลาดเคลื่อนเฉลี่ยกำลังสองสำหรับตัวทำนายแต่ละอันดับ  
ที่ได้จากการทดลองและได้จากทฤษฎี

รูป 3.10.4 แสดงค่าความคลาดเคลื่อนเฉลี่ยกำลังสอง สำหรับแต่ละตัวทำนาย โดยค่าเฉลี่ย  $e^2(m,n)$  จากจุดต่าง ๆ ของภาพในรูป 3.10.3a) กราฟแสดงการลดความคลาดเคลื่อน เมื่อจำนวนจุดที่ใช้ในการทำนายเพิ่มขึ้นเป็น 3 ผลในตาราง 3.10.2 และรูป 3.10.4 แสดงค่าที่ได้จาก รูป 3.10.3 a)



รูป 3.10.5 ฮิสโตแกรมของสัญญาณแตกต่าง  $e(m,n)$  สำหรับตัวทำนายอันดับที่ 3 มีการนำไปใช้

รูป 3.10.5 แสดงฮิสโตแกรมของสัญญาณแตกต่าง  $e(m,n)$  ในกรณีของตัวทำนายนระดับที่ 3 สำหรับภาพในรูป 3.10.3 a) หลังจากมันถูกทำเป็นมาตรฐาน ยกเว้นตัวประกอบมาตรฐาน (normalization factor) ฮิสโตแกรมนี้ประมาณฟังก์ชันความหนาแน่นของข้อมูลแตกต่าง ด้วยค่าพิก (peak) ฟังก์ชันความหนาแน่นสูง ค่าข้อมูลที่แตกต่างจะดีที่สุดในโดยการควอนไทซ์โดยนอนยูนิฟอร์มควอนไทเซอร์ (Nonuniform quantizer) รูป 3.10.5 แสดงฮิสโตแกรมที่ได้จากการคำนวณ



รูป 3.10.6 a) ผลของภาพที่ใช้การลดข้อมูลภาพด้วยการทำนายน เหลือ 3 บิตต่อจุด

b) ภาพที่ได้กลับคืนมา โดยใช้ 2 บิตต่อจุด

c) ภาพที่ได้กลับคืนมา โดยใช้ 1 บิตต่อจุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลของการลดข้อมูลภาพเหลือ 3 บิทต่อ 1 จุด โดยใช้ การลดข้อมูลภาพด้วยวิธีทำ  
 นายแสดงในรูปแบบที่ 3.10.6 a) ผลในภาพ 3.10.6 เป็นตัวทำนายระดับที่ 1 ใช้จุดข้าง  
 เคียงที่ใกล้ที่สุดตามแนวอน การลดข้อมูลภาพด้วยวิธีทำนาย 3 บิท หมายความว่า เรา  
 ใช้เฉลี่ย 3 บิทต่อ 1 จุด สำหรับภาพที่กลับคืนมาในรูปแบบ 3.10.6 a) โดยคอนโตเซอรัมี  
 เอาที่พูด 8 ระดับ รูป 3.10.6 b) และ 3.10.6 c) แสดงภาพที่กลับคืนมาโดยใช้ 2 บิท  
 และ 1 บิทต่อ 1 จุดตามลำดับ



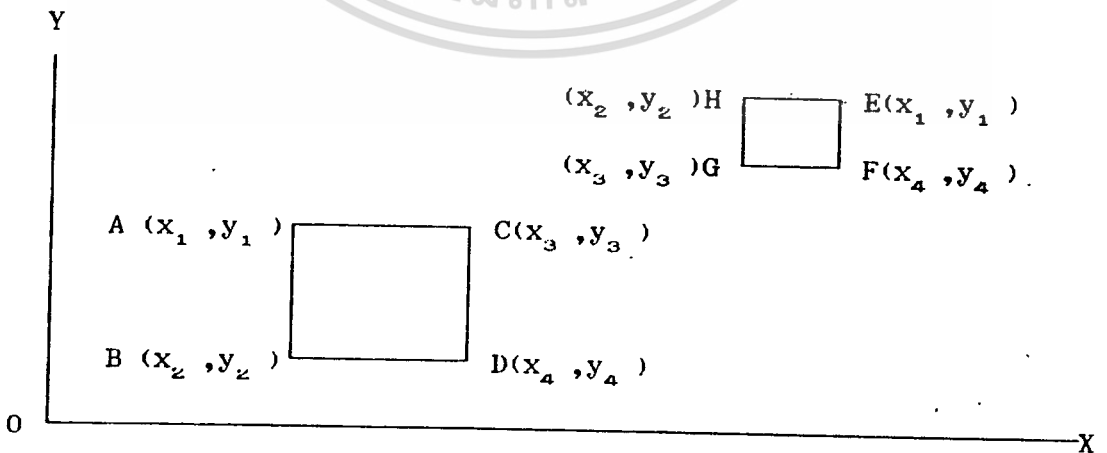
### 3.11 การลดข้อมูลภาพโดยไอเทอร์เรตฟังกชันซิสเต็ม

(IMAGE COMPRESSION USING ITERATED FUNCTION SYSTEM)

การลดข้อมูลภาพ (Image Compression) โดยใช้หลักของไอเทอร์เรตฟังกชันซิสเต็ม (IFS) ซึ่งสามารถลดขนาดของข้อมูลลงได้มาก หลักการนี้จะทำให้รูปขนาดใหญ่ถูกลดสเกล (scale) ลงเป็นรูปเล็กๆ ไปอยู่ในระนาบใหม่ เมื่อเราทำการ เข้ารหัส (transform) ในจำนวนครั้งที่มากแล้ว รายละเอียดของภาพที่จะถอดรหัสกลับมา ก็จะมีมาก และต้องอาศัยทฤษฎีคอลลาจ (Collage) มาช่วย กล่าวคือ การบีบอัดจากรูปเดิมไปเป็นรูปเล็ก จะต้องให้พื้นที่ของรูปเล็ก ปกคลุมในส่วนของรูปใหญ่ด้วย ในบทความนี้ จะนำเสนอเฉพาะการลดข้อมูลที่มีความแตกต่างของระดับสี (grey level) เพียง 2 ระดับ เท่านั้น

#### 3.11.1 ไอเทอร์เรตฟังกชันซิสเต็ม (IFS)

ไอเอฟเอส เป็นทฤษฎีจากเรขาคณิต (geometry) ที่ใช้การทรานส์ฟอร์ม เพื่อแสดงความสัมพันธ์ระหว่างส่วนของภาพ ซึ่งความสัมพันธ์นี้ ทำให้สามารถมองโครงสร้างของภาพที่ยากๆ ให้ อยู่ในรูปโครงสร้างของภาพที่ง่ายขึ้นได้ โดยมองความสัมพันธ์ในรูปแบบของรหัส ไอเอฟเอส การทรานส์ฟอร์มในที่นี้หมายถึงการรวมกันของการหมุนและการเลื่อน การเปลี่ยนสัดส่วนของแกนโคออร์ดิเนต ใน  $n$ -มิติสเปซ เพื่อให้เข้าใจถึงรหัสไอเอฟเอส ขอยกตัวอย่างการทรานส์ฟอร์ม ใน 2 มิติดังนี้



รูป 3.11.1 แสดงการทรานส์ฟอร์มรูปสี่เหลี่ยม

รูปสี่เหลี่ยม EFGH

รูปแบบทั่วไปของการทรานส์ฟอร์มจุด  $(x, y)$  ไปเป็น  $(x', y')$  สามารถเขียนได้ดัง  
แสดงในสมการ 3.11.1

$$\begin{aligned} \begin{bmatrix} x' \\ y' \end{bmatrix} &= W \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} \\ &= \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} \\ &= \begin{bmatrix} ax + by + e \\ cx + dy + f \end{bmatrix} \end{aligned} \quad (3.11.1)$$

โดย  $W$  เป็นการรวมกันของการหมุน การเปลี่ยนสัดส่วน และการเลื่อน ดังนั้น เราสามารถหาค่า  
สัมประสิทธิ์ได้ ดังสมการ 3.11.2

$$\begin{aligned} a &= r \cos A, & b &= -s \sin B \\ c &= r \sin A, & d &= s \cos B \\ e &= \text{เป็นการเลื่อนของแกน } x \\ f &= \text{เป็นการเลื่อนของแกน } y \end{aligned} \quad (3.11.2)$$

โดย  $r$  เป็นสเกลิ่งแฟคเตอร์ (scaling factor) ของแกน  $x$   
 $s$  เป็นสเกลิ่งแฟคเตอร์ของแกน  $y$   
 $A$  เป็นมุมที่หมุนไปบนแกน  $x$   
 $B$  เป็นมุมที่หมุนไปบนแกน  $y$

ถ้าเราต้องการค่าสเกลิ่งแฟคเตอร์เท่าไร รู้ว่าต้องการหมุนไปเป็นมุมกี่องศา  
และทำการเลื่อนไปเป็นเท่าไร เราก็สามารถคำนวณหาค่าสัมประสิทธิ์ จากสมการ 3.11.2  
ได้โดยง่ายและสุดท้ายคำนวณหา  $(x', y')$  จากสมการได้ในการทรานส์ฟอร์ม รูปสี่เหลี่ยม ABCD  
ก็เพียงพอแต่ทำการทรานส์ฟอร์มจุด A จุด B จุด C และจุด D ไปเป็นจุด E จุด F จุด G และจุด H แต่  
ในทางตรงข้าม ถ้าเรารู้จุด A, B, C, D และรู้จุด E, F, G, H ต้องการคำนวณหาอันนี้สามารถทำได้  
โดย แก้มสมการเชิงเส้น (Linear equation) ต่อไปนี้

$$x_1 + ay_2 + bte = x_1$$

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น  $x_2 + ay_2 + bte = x_2$  เปลี่ยนเนื้อหา (3.11.3) ถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการนี้สามารถคำนวณค่า  $a, b, c$  ได้ และเราสามารถคำนวณค่า  $c, d, f$  ได้จากสมการ

$$\begin{aligned} x_1 c + y_1 d + f &= y_1 \\ x_2 c + y_2 d + f &= y_2 \\ x_3 c + y_3 d + f &= y_3 \end{aligned} \quad (3.11.5)$$

จากตัวอย่างนี้เรารู้จุดที่แน่นอนของจุด  $A, B, C, D$  และ  $a, b, c, d$  เราก็สามารถคำนวณหา  $a, b, c, d, e$  และ  $f$  ซึ่งเป็นค่าของ สัมประสิทธิ์ ของ  $P$  ได้ กำหนดให้เป็น  $w_1$  และถ้าเรารู้จุดอื่นบนขอบสี่เหลี่ยมผืนผ้า  $ABCD$  ที่สัมพันธ์กับจุดอื่นบนขอบสี่เหลี่ยมผืนผ้า  $EFGH$  เราก็สามารถคำนวณหาค่าสัมประสิทธิ์ ของ  $P$  ได้และทำนองเดียวกัน เราสามารถหาได้ถึง  $w_n$

ตาราง 3.11.1 แสดงตัวอย่างรหัส ไอเอฟเอส ซึ่งก็คือค่าสัมประสิทธิ์ของ  $P$  ที่มีค่าความน่าจะเป็นรวมอยู่ด้วยมี  $w_1, w_2, w_3$  และ  $w_4$  ค่าความน่าจะเป็นต้องมีค่ารวมกันเท่ากับ 1

$w$	$a$	$b$	$c$	$d$	$e$	$f$	$p$
1	0	.5	-.5	0	5	35	.25
2	0	.5	-.5	0	25	35	.25
3	0	.5	-.5	0	5	55	.25
4	0	.5	-.5	0	25	55	.25

ตาราง 3.11.1 แสดงรหัส ไอเอฟเอส ของการทรานส์ฟอร์มรูปไป 4 ครั้ง

### 3.11.2 แบนดอมไลเทอเรชันอัลกอริทึม

(The Random Iteration Algorithm)

เราใช้เป็นวิธีถอดรหัส ไอเอฟเอส ได้ดี จากไอเอฟเอส ได้ดี เราให้  $m$  เป็นจำนวนครั้งของการทรานส์ฟอร์ม จากการทรานส์ฟอร์ม เราจะใช้ แบนดอมไลเทอเรชันอัลกอริทึม ดังนี้

- 1) ให้ค่าเริ่มต้น (Initialize);  $x=0, y=0$ .
- 2) สำหรับค่า  $n = 1$  ถึง 2500 ทำขั้นตอนที่ 1 ถึง 7

3) เลือกค่า  $k$  จาก  $1, 2, \dots, m$ , ด้วยค่าความน่าจะเป็น  $P_k$  นำไปใช้ประโยชน์ด้วยการคำนวณ  $x, y$  ใหม่

4) ทำการทรานส์ฟอร์มเมชัน  $w_k$  นี้ไปยังจุด  $(x, y)$  ถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) เช็ท  $(x, y)$  เท่ากับจุดใหม่  $x=x \text{ bar}, y=y \text{ bar}$ .

6) ถ้า  $n > 10$ , พล็อต  $(\text{plot})(x, y)$ .

7) วาดรูป

แรนคอมไอบ์เทอเรชั่นอัลกอริทึม จะสร้างรูปที่เหมือนกันขึ้นเรื่อยๆ ซึ่งสามารถทดสอบการทำงานว่าทำงานได้จริง จากการทดลองทางคณิตศาสตร์ (mathematics) หลังจากนำไอเอฟเอสโค้ดผ่านแรนคอมไอบ์เทอเรชั่นอัลกอริทึม จะได้ภาพเดิมกลับคืนมา

ตัวอย่างการทำ แรนคอมไอบ์เทอเรชั่นอัลกอริทึม โดยใช้รหัส ไอเอฟเอส ในตารางกลับมาเป็นสี่เหลี่ยม ABCD ได้ ดังแสดงในรูป a, b และ c. การทำซ้ำ (Iteration) 500, 1000, 2500 ครั้ง ตามลำดับ



(a)



(b)



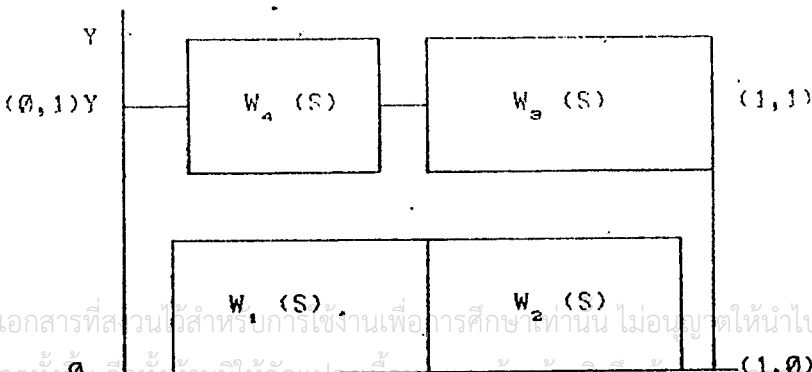
(c)

รูป 3.11.3 แสดงการทำแรนคอมไอบ์เทอเรชั่น

### 3.11.3 ทฤษฎีคอลลาจ

(The Collage Theorem)

เป็นทฤษฎีที่ใช้แสดงวิธีการทราเซ่ฟอร์มเพื่อสร้าง ไอเอฟเอสโค้ด ดังรูป



(๑, 1)Y	$W_3 (S)$	$W_4 (S)$	(1, 1)
0	$W_1 (S)$	$W_2 (S)$	(1, 0)

รูป 3.11.4

เราเริ่มค่าจากรูปในสี่เหลี่ยม ในระนาบ  $(x, y)$  ด้วยจุด  $(๑, ๑), (1, ๑), (1, 1), (๑, 1)$  เลือกการทราวนซ์เฟอร์มีเป็น  $W_1 (S), W_2 (S), W_3 (S), W_4 (S)$  เพื่อให้สี่เหลี่ยมถูกสร้างขึ้นจาก  $W_1 (S) \cup W_2 (S) \cup W_3 (S) \cup W_4 (S)$  ดังรูป ๓.11.4 รูปซึ่งเกิดจากการทราวนซ์เฟอร์มีก็จะครอบคลุมสี่เหลี่ยมใหญ่ได้

เราใช้วิธีของ ไอเอฟเอส เก็บ แก้มการ จะได้ค่า ไอเอฟเอส ได้ค แล้วมา ได้ค ที่ได้ฟ้าแระนตอมไอเทอเซซังแอลกอวิม ก็จะได้รูปสี่เหลี่ยมเกิดขึ้น จากตัวอย่างที่ผ่านมา เราต้องหากลุ่มของการทราวนซ์เฟอร์มีที่เหมาะสม เพื่อเ้าทวูโยเเน (union) กัน ให้ได้ภาพที่ ต้องการออกมา

3.11.4 การหาค่าความน่าจะเป็นในรหัส ไอเอฟเอส

จุดที่แตกต่างกันของความน่าจะเป็น ไม่ได้อยู่ที่ความแตกต่างของภาพ แต่มันแสดงถึงอัตรา ที่ขอบเขต (หรือ distribution) ของภาพ ซึ่งถูกับรจอยู่ไนบริเวณที่ผ่านการทราวนซ์เฟอร์มี จากสมการการทราวนซ์เฟอร์มี

$$W_i \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix}$$

ที่  $i = 1, 2, 3, \dots, n$

เวลาของการที่จุดเคลื่อนที่แบบสุ่ม จะใช้เวลาในภาพย่อย (sub-image)  $W_i =$  ฝั่ง ที่ของ  $W_i$  / ฝั่งที่ของ 1

ความน่าจะเป็นหาได้จาก

$$P_i = a_i, d_i, -b_i, c_i / \text{ผลรวมค่าสัมบูรณ์ของ } a_i, d_i, -b_i, c_i \text{ โดย } k=1 \text{ to } n$$

กำหนดว่าค่า  $P_i$  ต้องไม่เท่ากับ ๑ ถ้าค่าวนได้ค่า ๑ เราจะแทนด้วย ๑.๑๑1 และถ้าไม่ว่า ค่าความน่าจะเป็นทั้งหมดจะต้องรวมกัน  $= 1$  และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### บทที่ 4

### การนำเทคนิคการลดข้อมูลภาพประยุกต์ใช้กับ ไมโครคอมพิวเตอร์

#### และผลการทดลอง

จากการศึกษาถึงเทคนิคในการลดข้อมูลภาพ ดังได้กล่าวมาแล้วในบทที่ 3 นั้น พบว่าแต่ละเทคนิค มีรูปแบบและความเหมาะสม สำหรับภาพแต่ละประเภทแตกต่างกันไป บางเทคนิคสามารถลดข้อมูลได้มากถึง 4 หรือ 8 เท่า แต่ภาพที่ได้ก็จะมีข้อมูลพลาดมาก บางวิธีสามารถลดข้อมูลได้น้อยตั้งแต่ 10-60 % แต่ภาพที่ได้จากการถอดรหัส จะมีข้อผิดพลาด น้อยหรือไม่ผิดพลาดเลย เพราะฉะนั้นในการเลือกใช้เทคนิคที่เหมาะสมก็ขึ้นอยู่กับ ลักษณะข้อมูลภาพ และการนำไปใช้งานว่าต้องการประสิทธิภาพ หรือต้องการปริมาณการลดข้อมูลมาก ๆ

จากการศึกษาและทดสอบความเป็นไปได้ที่จะลดข้อมูลภาพ โดยทดสอบกับภาพดิจิทัล ขนาด 256 X 256 จุด<sup>2</sup> ใช้ระดับสีเทา 256 ระดับ โดยมีขั้นตอนดังนี้

#### ขั้นตอนการวิจัย

จัดเตรียมข้อมูลภาพ

แบ่งแยกประเภทภาพตามลักษณะการกระจายของข้อมูล

จัดเตรียมอัลกอริทึมที่คาดว่าจะลดข้อมูลภาพได้ โดยสอดคล้องกับประเภทของภาพ

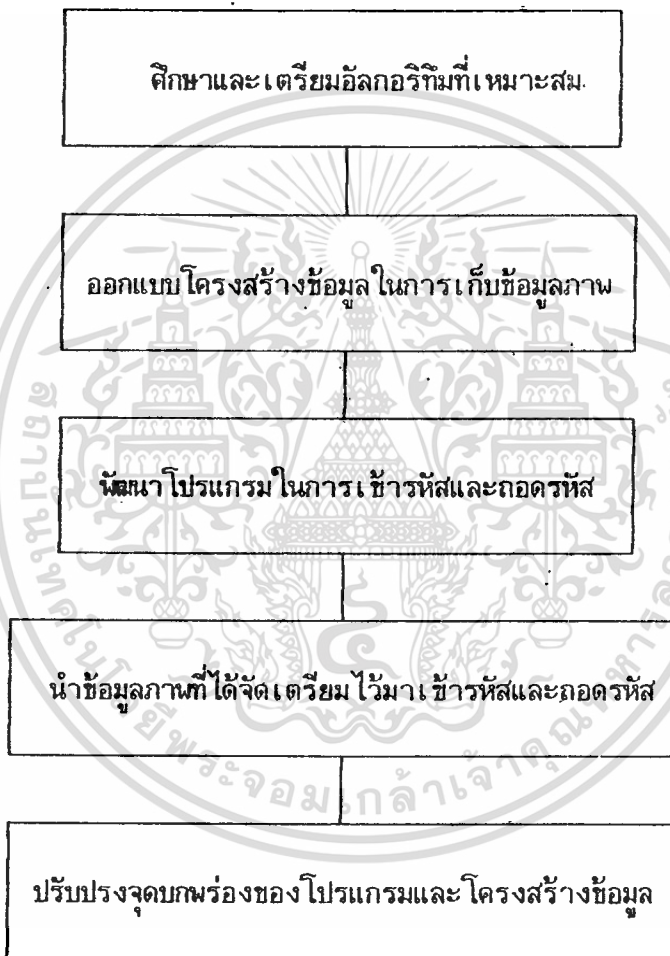
วิเคราะห์แบบแผน เพื่อทำการเลือกฟังก์ชันต่าง ๆ ที่เหมาะสมกับภาพแต่ละชนิด

ประเมินผลที่ได้จากการใช้ฟังก์ชันต่าง ๆ เพื่อลดข้อมูลภาพ

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากได้ทำการประเมินผลที่ได้ จากการใช้ฟังก์ชันต่าง ๆ เพื่อลดข้อมูลภาพ  
แบบต่าง ๆ แล้ว เราก็ต้องทำการพัฒนาโปรแกรม เพื่อให้สามารถนำไปใช้งานได้จริง โดย  
มีขั้นตอนในการพัฒนาโปรแกรม ดังนี้

### ขั้นตอนในการพัฒนาโปรแกรม



วัตถุประสงค์ของการลดข้อมูลภาพ ที่ได้พัฒนาขึ้น ในโครงการวิจัยนี้ เพื่อประหยัด  
เนื้อที่หน่วยความจำ และช่วยในการส่งข้อมูลให้ได้อย่างรวดเร็ว เพราะฉะนั้นการเลือกอัลกอ  
ริทึม จากเทคนิคที่ได้ศึกษามาทั้งหมด ประกอบกับวัตถุประสงค์ ในการส่งข้อมูลที่ได้ทำ  
การเข้าใจผิด เพื่อประหยัดค่าใช้จ่ายและความรวดเร็ว สิ่งที่ต้องคำนึงถึงสำหรับการลดข้อมูล  
ภาพคือ ความเร็วในการเข้าใจผิด และถอดรหัส จะต้องอยู่ในช่วงเวลาที่เหมาะสม คือ  
ไม่ช้าจนเกินไป เทคนิคที่เหมาะสมมีหลายวิธี เหมาะกับภาพต่าง ๆ กัน สำหรับในโครงการงาน

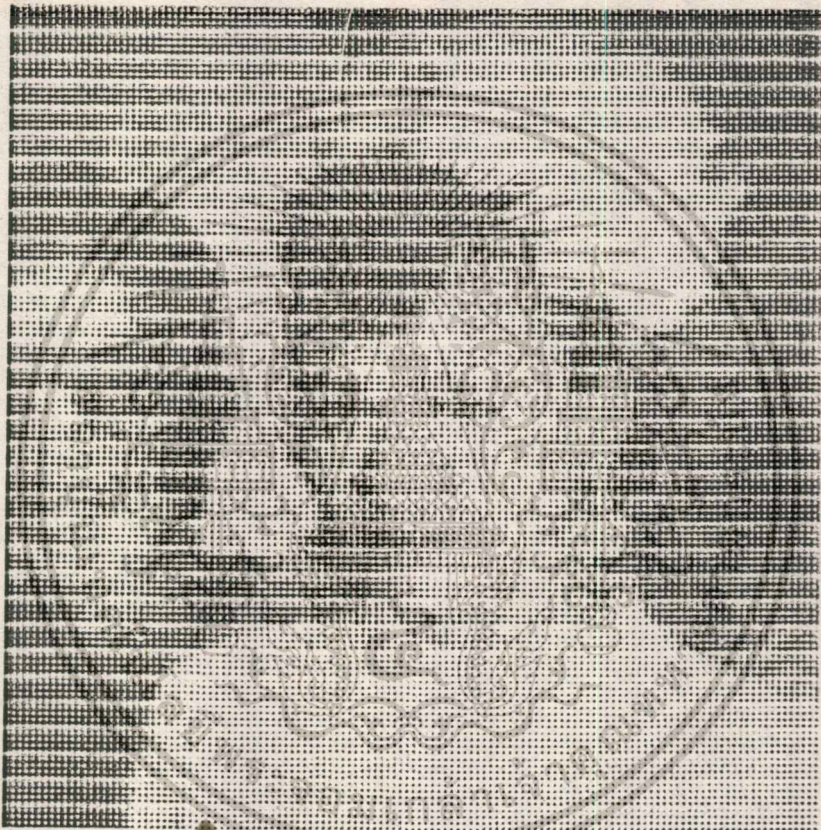
นี้ได้พิจารณาตัวอย่างเทคนิคลดข้อมูลภาพแตกต่างกัน 4 วิธี เพื่อทำการโปรแกรมใช้กับไมโครคอมพิวเตอร์ ดังนี้

1. การลดข้อมูลภาพด้วยวิธีรันเลงจ์โค้ด (Runlength coding)
2. การลดข้อมูลภาพด้วยวิธีบล็อกทรังก์ชัน (Block Truncation coding)
3. การลดข้อมูลภาพด้วยวิธีฮัฟฟ์แมนโค้ด (Huffman coding)
4. การลดข้อมูลภาพด้วยวิธีไอเอฟเอส (Iterated function system)

ผลการทดลอง

มีดังนี้

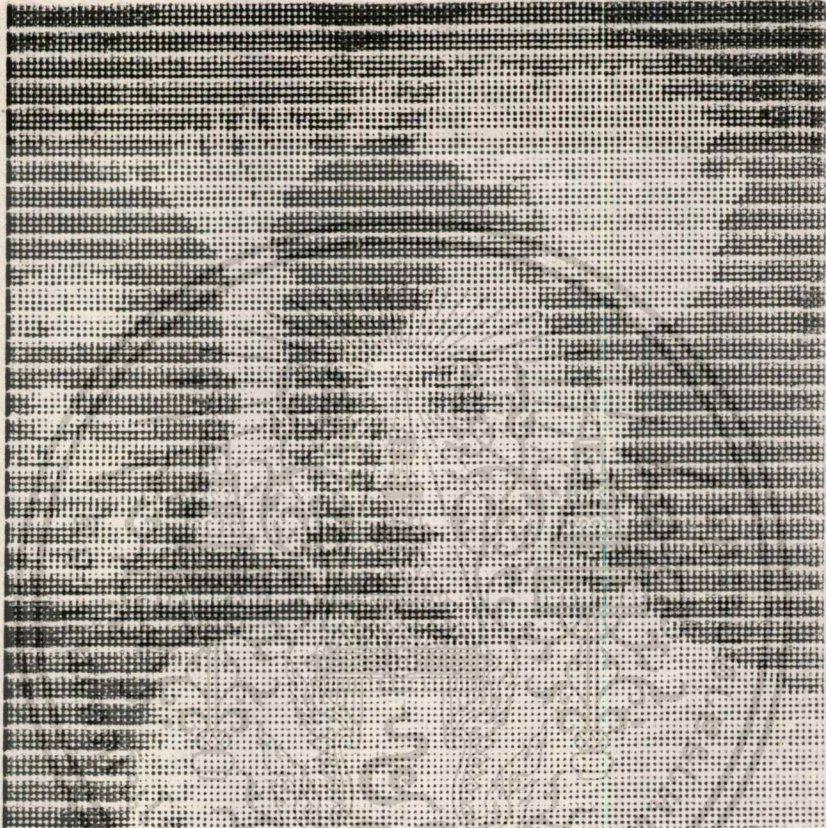




รูป 4.1 แสดงภาพต้นแบบขนาด  $256 \times 256$  จุด<sup>2</sup> 256 ระดับสี

เมื่อเข้ารหัสแบบรันเลงจ์โค้ด จะไม่สามารถลดข้อมูลได้

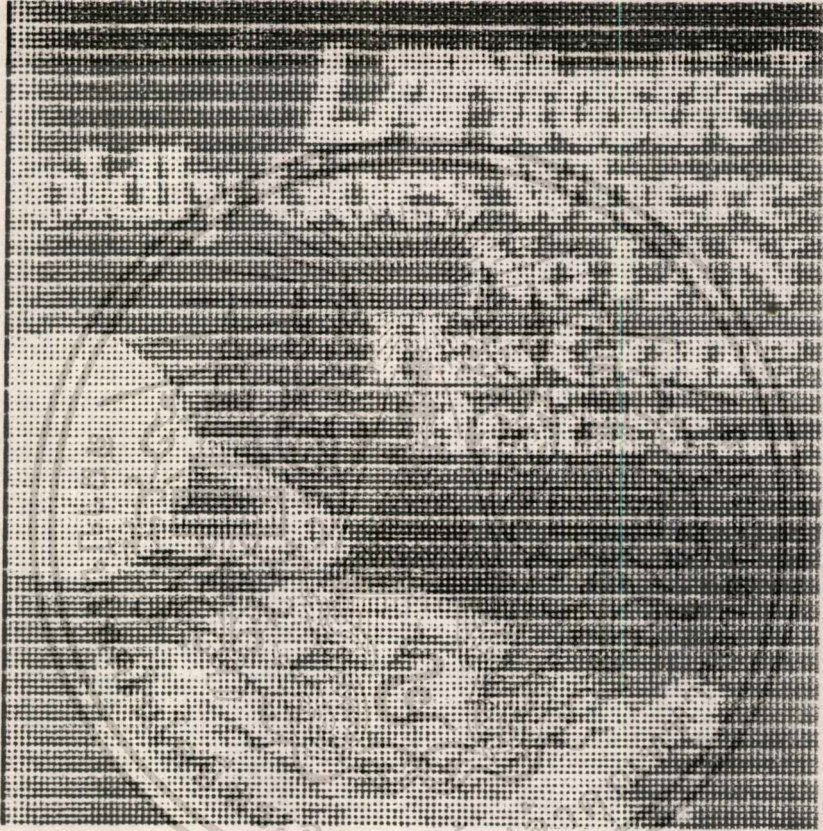
เมื่อเข้ารหัสแบบฮัฟฟ์แมนโค้ด จะสามารถลดข้อมูลภาพลงเหลือ 5.32 บิต/จุด



รูป 4.2 แสดงภาพต้นแบบขนาด  $256 \times 256$  จุด<sup>2</sup> 8 ระดับสี

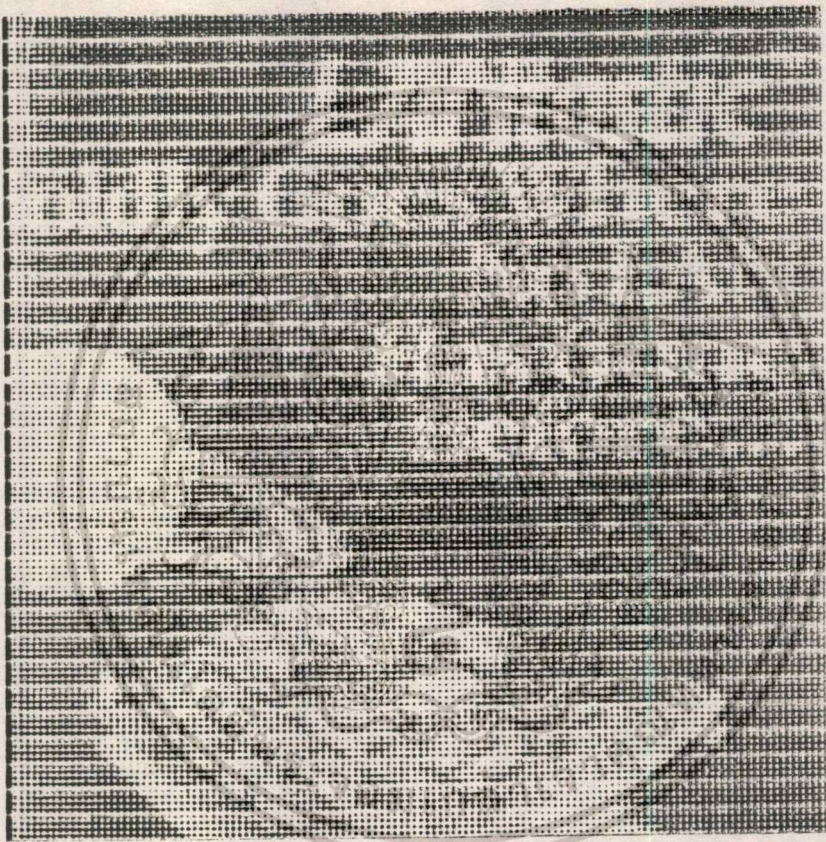
เมื่อเข้ารหัสแบบรันเลงจ์โค้ด จะสามารถลดข้อมูลได้เหลือ 1.92 บิตต่อจุด

เมื่อเข้ารหัสแบบฮัฟฟ์แมนโค้ด จะสามารถลดข้อมูลภาพลงเหลือ 2.57 บิตต่อจุด



รูป 4.3 แสดงภาพต้นแบบขนาด  $256 \times 256$  จุด<sup>2</sup> 256 ระดับสี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



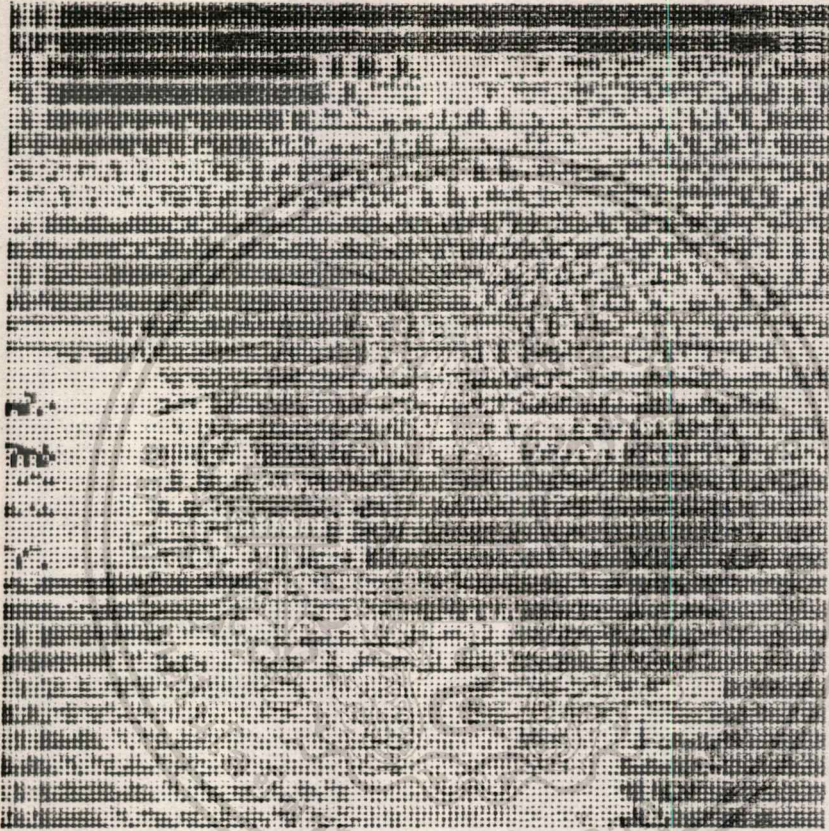
รูป 4.4 แสดงภาพหลังจากถอดรหัสด้วยวิธีของ บล็อกทรุนเคชั่น

โดยการเข้ารหัส กำหนดให้บล็อกมีขนาด  $2 \times 2$  จุด<sup>2</sup>

สามารถลดข้อมูลลงเหลือเพียง 5 บิตต่อจุด

ภาพที่ได้หลังจากถอดรหัสจะมีลักษณะ ใกล้เคียงกับภาพต้นแบบมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

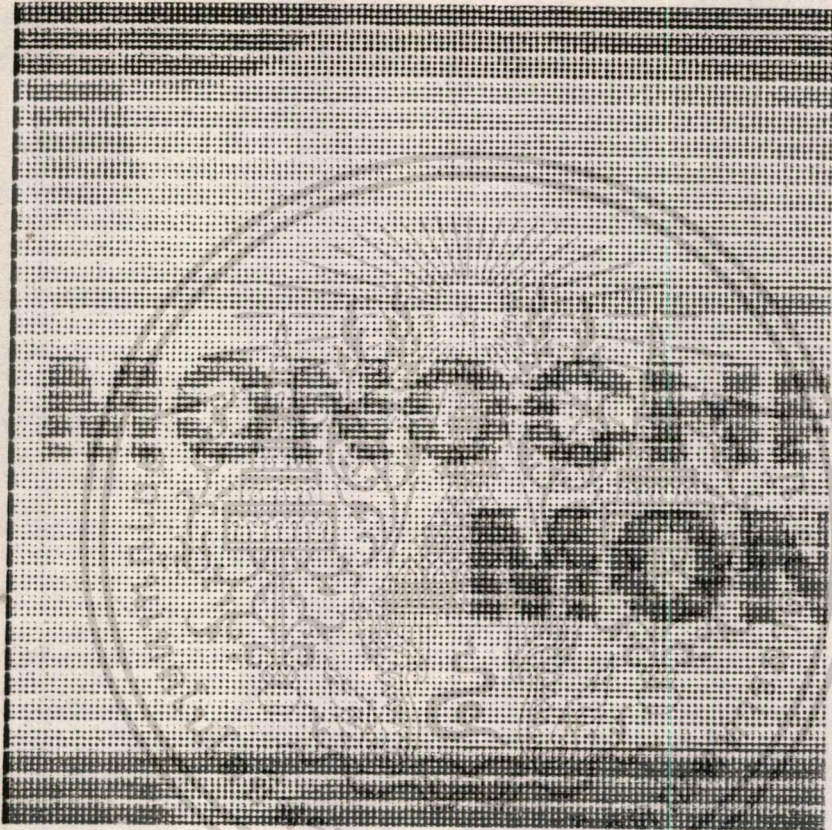


รูป 4.5 แสดงภาพหลังจากถอดรหัสแล้ว ด้วยวิธีบีบอัดทรมุเคชั่น

โดยการเข้ารหัส กำหนดขนาดของบล็อกไม่เหมาะสม คือมีขนาดใหญ่เกินไป คือ  
ขนาด  $8 \times 8$  จุด<sup>2</sup> สามารถลดข้อมูลภาพลงเหลือเพียง 1 บิตต่อจุด

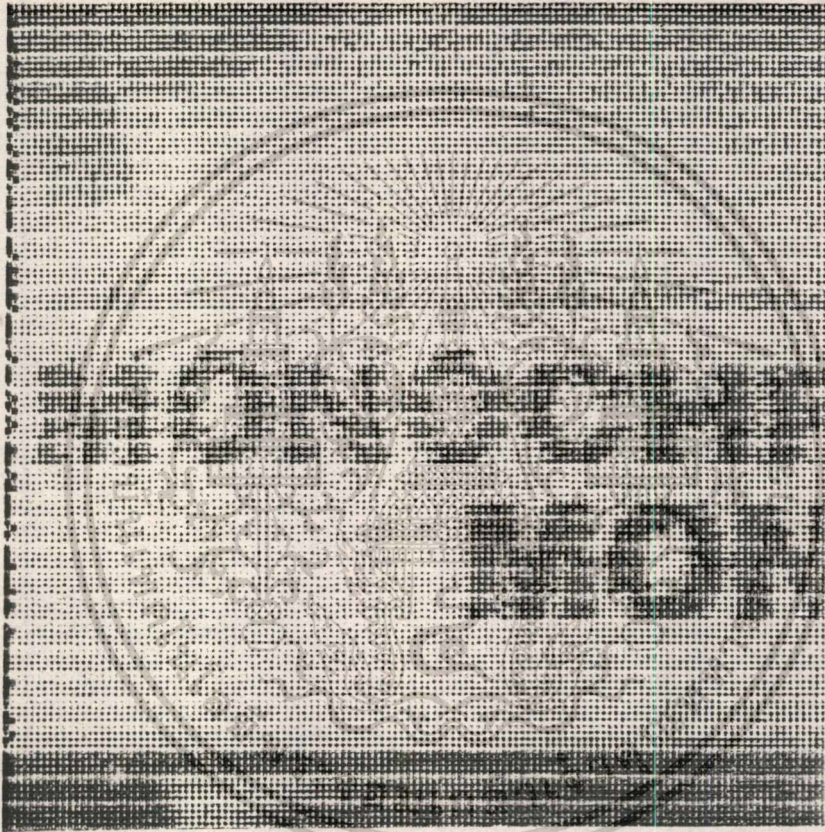
แต่ภาพที่ได้จากการถอดรหัสมีความคลาดเคลื่อนมาก และมีลักษณะ เป็นบล็อกอย่าง  
เห็นได้ชัด คือเกิดภาวะบล็อกกี้ (blocky)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.6 แสดงภาพต้นแบบ ขนาด 256 X 256 จุด<sup>2</sup> 256 ระดับสี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.7 แสดงภาพหลังจากการถอดรหัสด้วยวิธีบิตลดทอนเคชัน

โดยการเข้ารหัส กำหนดให้บิตลดขนาด  $4 \times 4$  จุด<sup>2</sup>

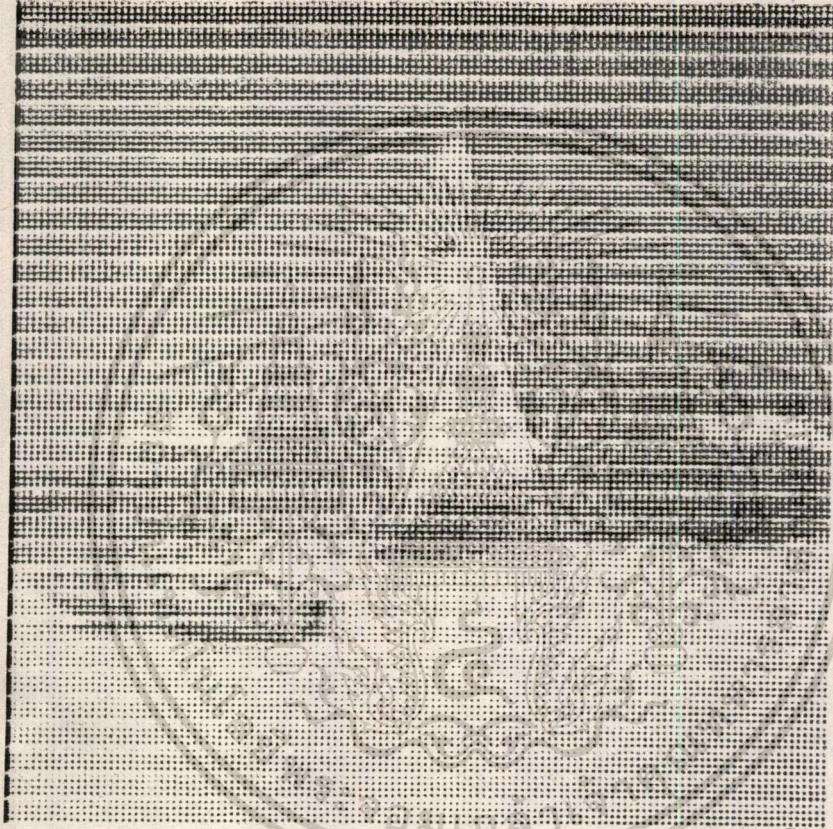
สามารถลดข้อมูลลงเหลือเพียง 2 บิตต่อจุด

ภาพที่ได้หลังถอดรหัสมีความคลาดเคลื่อนบ้างและเกิดภาวะบิตคึก

แต่ก็ยังคง

เอกลักษณ์เค้าโครงของภาพได้ ทรัพยากรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.8 แสดงภาพต้นแบบ ขนาด  $256 \times 256$  จุด<sup>2</sup> 16 ระดับ

เมื่อเข้ารหัสด้วยวิธีรันเลจแล้วลดข้อมูลลงเหลือ 2.71 บิตต่อจุด

เมื่อเข้ารหัสด้วยวิธีฮัฟฟ์แมนไคต์แล้วลดข้อมูลลงเหลือ 3.39 บิตต่อจุด



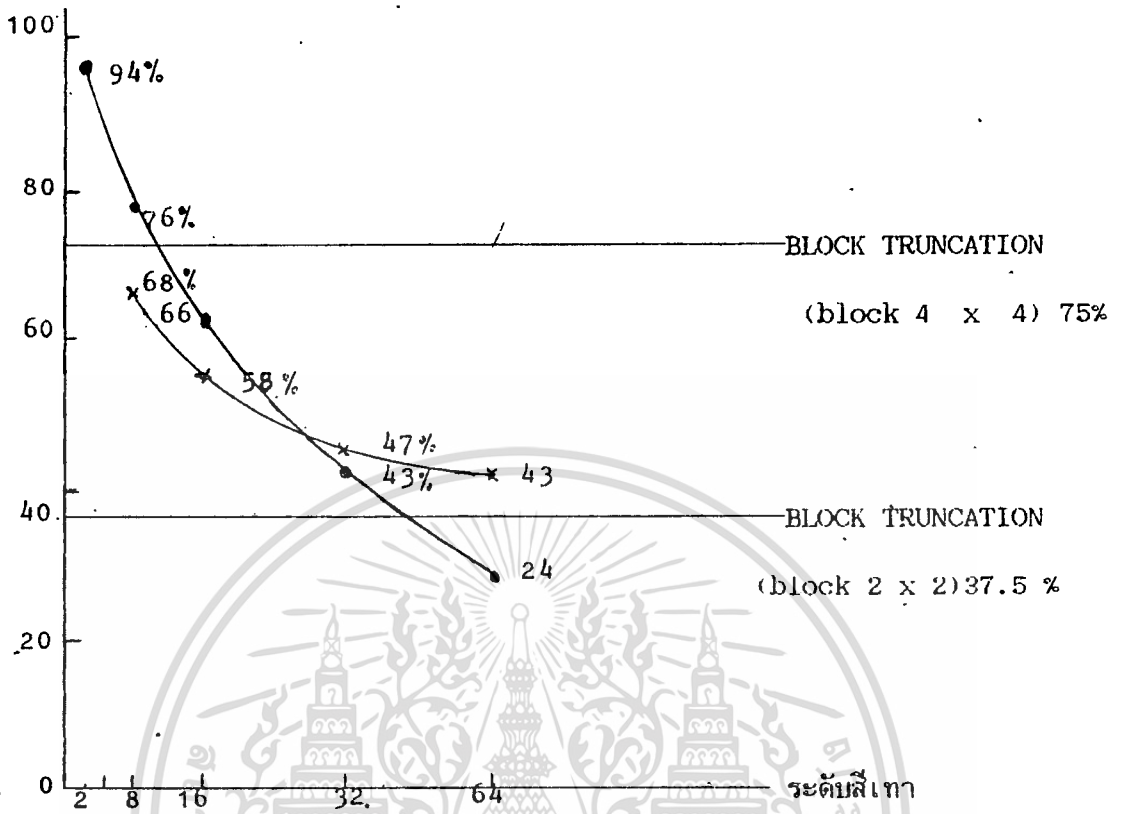
รูป 4.9 ภาพต้นแบบขนาด  $256 \times 256$  จุด<sup>2</sup> 2 ระดับสี

เข้ารหัสรันเลงจ์ ลดข้อมูลภาพลงเหลือ 0.49 บิตต่อจุด

เข้ารหัสฮัฟฟ์แมนโคัด ไม่สามารถลดข้อมูลลงได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

% การลดข้อมูลภาพ (เทียบกับมาตรฐานการเก็บไฟล์)



กราฟแสดงอัตราการลดข้อมูลภาพ ด้วยเทคนิคต่าง ๆ เมื่อทดลองกับภาพที่มีจำนวน

ระดับสีเทามากขึ้น

- แทน รันเลงจ์
- x แทน ฮัฟฟ์แมน โค้ด

จากกราฟและตารางเป็นผลที่ได้จากการทดลองจริง ๆ เมื่อนำข้อมูลภาพผ่าน

กระบวนการลดข้อมูลภาพ ด้วยเทคนิคต่าง ๆ สามารถสรุปข้อดี และข้อเสียของแต่ละเทคนิค ดังนี้

ตาราง 4.1 เปรียบเทียบการลดข้อมูลภาพ 3 วิธี

ข้อมูลภาพต้นแบบ	RUNLENGTH		HUFFMAN		BTC(block4x4)		BTC(block2x2)	
	BIT/PEL	ERROR	BIT/PEL	ERROR	BIT/PEL	ERROR	BIT/PEL	ERROR
ภาพผู้หญิง 8 ระดับสี	1.91	free	2.57	free	2	not free	5	not free
ภาพผู้หญิง 256 ระดับสี	ลดไม่ได้	-	5.32	free	2	not free	5	not free
ภาพตัวหนังสือ	6.18	free	4.55	free	2	not free	5	not free

#### การลดข้อมูลภาพด้วยวิธีรันเลงจ์โค้ด

ข้อดี

- เป็นเทคนิคที่มีประสิทธิภาพสูง คือ จะไม่มีความผิดพลาดของข้อมูลภาพ (error) หลังจากการถอดโค้ด
- ใช้เวลาในการเข้าโค้ดและถอดโค้ดน้อย ประมาณ 30 วินาที / 1 ภาพ (เฉพาะเข้ารหัส)
- ใช้ได้กับทั้งภาพขาว-ดำ ภาพเกรย์เลเวล และภาพสี

ข้อเสีย

- ปริมาณการลดข้อมูลภาพจะทำได้ไม่มากนัก
- มีข้อจำกัดคือ เหมาะสมกับภาพที่มีระดับสีซ้ำกันมาก ๆ เช่น ภาพตัวหนังสือ

#### การลดข้อมูลภาพด้วยวิธีบล็อกทรูเนชั่น

ข้อดี

- สามารถลดข้อมูลภาพได้มาก

เอกสารนี้เป็นเอกสารใช้ได้กับทั้งภาพขาว-ดำที่ ภาพเกรย์เลเวล ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น ห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อเสีย

- เสียเวลาในการเข้ารหัสและถอดรหัสมาก เนื่องจากต้องทำการคำนวณ
- มีความผิดพลาดบ้าง ขึ้นอยู่กับปริมาณการลดข้อมูลภาพ ถ้ายังลดข้อมูลภาพมาก เเปอร์เซ็นต์ความผิดพลาดจะสูง ถ้าลดข้อมูลภาพน้อย ความผิดพลาดจะน้อยลงตามลำดับ

### การลดข้อมูลภาพด้วยวิธีฮัฟฟ์แมน โค้ด

ข้อดี

- ไม่มีความคลาดเคลื่อน ภาพที่ได้จากการถอดรหัสจะเหมือนเดิมทุกประการ
- เหมาะสมกับภาพเกรย์เลเวล

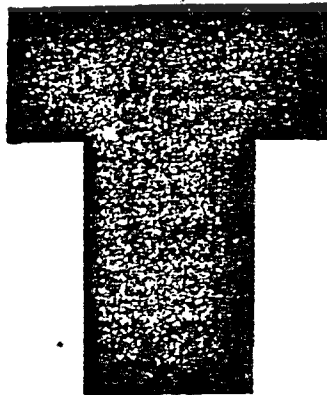
ข้อเสีย

- เสียเวลาในการเข้าโค้ดและถอดโค้ดมาก เนื่องจากใช้การคำนวณซับซ้อน

### ผลการทดลองลดข้อมูลภาพด้วยวิธี ไอเอฟเอส (Iterated Function System)

ไอเอฟเอสเป็นเทคนิคการลดข้อมูลภาพที่มีระดับความเข้ม 2 ระดับ (ขาว-ดำ) โดยวิธี ไอเอฟเอส (IFS) ซึ่งสามารถลดข้อมูลภาพได้มาก เมื่อเปรียบเทียบกับวิธีการลดข้อมูลภาพแบบรันเลงจ์โค้ด

กำหนดให้มีรูปภาพต้นแบบดังแสดงในรูป 4.8



จากรูป 4.8 เป็นรูปภาพตัวอักษร เป็นภาพที่มีระดับความเข้ม 2 ระดับและมีขนาด  $64 \times 64$  จุด<sup>2</sup> ซึ่งภาพนี้ถ้านำไปเก็บในเครื่องคอมพิวเตอร์จะใช้เนื้อที่ 4 กิโลไบต์ (Kbytes) และถ้าทำการลดข้อมูลภาพโดยใช้ รันเลงจ์โค้ด จะใช้เนื้อที่ 140 ไบต์

การทดลองลดขนาดข้อมูลของภาพ ในรูป 4.8 นี้ทำได้โดยใช้เทคนิคการแบ่งรูป T ออกเป็น 2 ส่วน คือ แบ่งเป็นสี่เหลี่ยม 2 รูป จากนั้นทำการคำนวณหารหัส ไอเอฟเอส ของแต่ละส่วน ซึ่งรหัส ไอเอฟเอส ของส่วนแรก คำนวณหารหัส คำนวณได้ผลลัพธ์ดังแสดงในตาราง 4.1 และ รหัส ไอเอฟเอส ของส่วนที่ 2 คำนวณได้ผลลัพธ์ ดังแสดงในตาราง 4.2

ตาราง 4.1 แสดงรหัสไอเอฟเอส ของส่วนที่หนึ่ง

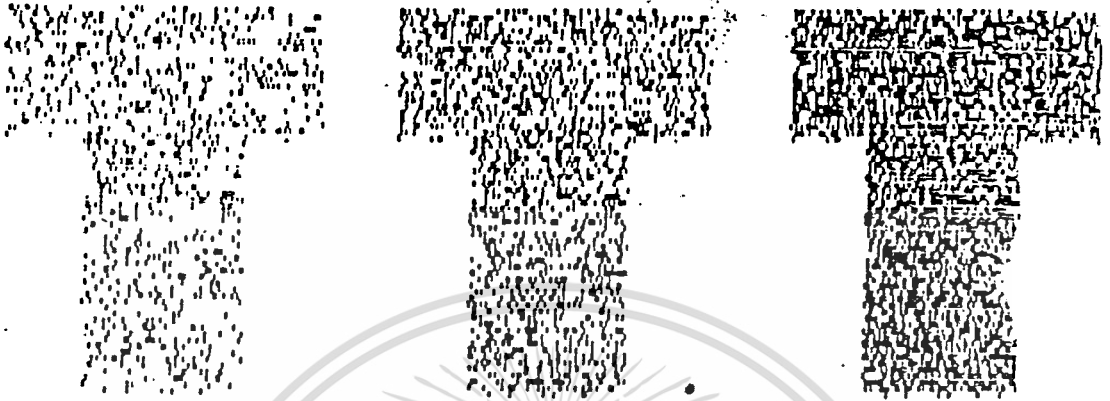
W	a	b	c	d	e	f	p
1	.5	0	0	.5	5	15	.25
2	.5	0	0	.5	25	15	.25
3	.5	0	0	.5	5	5	.25
4	.5	0	0	.5	25	5	.25

ตาราง 4.2 แสดงรหัสไอเอฟเอส ของส่วนที่ 2

W	a	b	c	d	e	f	p
1	.5	0	0	.5	10	35	.25
2	.5	0	0	.5	20	35	.25
3	.5	0	0	.5	10	15	.25
4	.5	0	0	.5	20	15	.25

จากทั้งสองตารางนี้ จะเห็นได้ว่า เราใช้เนื้อที่ในหน่วยความจำโดยประมาณ เพื่อทำการเก็บรหัส ไอเอฟเอส นี้ซึ่งต่างจากการเก็บข้อมูลภาพโดยตรงและการเก็บโดยวิธี รันเลงจ์โค้ด และการนำภาพกลับออกมาแสดงนั้น กระทำได้โดยการถอดรหัส ไอเอฟเอส นี้ด้วยวิธีแวนเดอมไฮเทอเวดอัลกอริทึม ที่ได้อธิบายไว้ตอนที่ 3 ซึ่งได้ผลลัพธ์ ดังแสดงในรูป 4.9

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(a) 500 ครั้ง

(b) 1000 ครั้ง

(c) 2500 ครั้ง

รูป 4.11 แสดงผลการถอทรหัส ไอเอฟเอส  
ตามจำนวนครั้งของการทำซ้ำ (Iteration)

จากการทดลอง เทคนิคของการลดขนาดของข้อมูลภาพลง โดยใช้วิธี ไอเอฟเอส สามารถลดขนาดข้อมูลภาพลงได้มาก เมื่อเปรียบเทียบกับวิธีการลดขนาดของข้อมูลภาพโดยวิธี รันเลงจ์โค้ด ทำให้การเก็บข้อมูลภาพ 1 ภาพ จะเปลืองเนื้อที่ในหน่วยความจำเท่ากับข้อมูล ในตารางรหัสไอเอฟเอสเท่านั้น โครงการนี้ศึกษาเฉพาะตัวอย่างข้อมูลภาพ ที่มีระดับความ เข้มเพียง 2 ระดับเท่านั้น ซึ่งจริงๆแล้ว สามารถประยุกต์นำไปใช้กับข้อมูลภาพที่เป็นสีหลายสี ได้โดยแบ่งภาพออกเป็นหลายๆ ให้แต่ละส่วนเหลือเพียง 2 สี (สีหลักกับสีพื้น) หลังจากนั้นก็สามารถใช้วิธี ไอเอฟเอส กับแต่ละส่วนได้ นอกจากนี้ยังเหมาะสำหรับนำไปใช้กับการส่งข้อมูล ภาพจากเครื่องคอมพิวเตอร์เครื่องหนึ่ง ไปยังเครื่องคอมพิวเตอร์อีกเครื่องหนึ่ง เนื่อง จากวิธี ไอเอฟเอสนี้ใช้เนื้อที่ไม่มากนัก ในหน่วยความจำ ใช้เนื้อที่ในหน่วยความจำไม่มากนัก

### การลดข้อมูลภาพด้วยวิธี ไอเอฟเอส

ข้อดี

- สามารถลดข้อมูลภาพได้มาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น ห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การเข้ารหัสทำได้ยาก อาศัยการทราบสฟอว์รภาพที่ยุ่งยาก ขึ้นซ้อน
- การถอดรหัสใช้เวลามาก ประมาณ 2 นาที / 1 ภาพ
- ในโครงการนี้ยังใช้ได้กับภาพไบนารีเท่านั้น

จากการศึกษาค้นคว้าการลดข้อมูลภาพด้วยวิธีอื่น ๆ ทำการวิเคราะห์ข้อดีและเสีย

ได้ดังนี้

#### การลดข้อมูลภาพด้วยวิธีฮัฟฟ์แมนบิตโค้ด

ข้อดี

- ใช้เวลาในการเข้ารหัสและถอดรหัส น้อยกว่าฮัฟฟ์แมนโค้ด
- สามารถใช้ได้กับภาพที่มีระดับสีเทาмаกระดับ
- ภาพที่ได้ไม่มีความคลาดเคลื่อน

ข้อเสีย

- อัตราการลดข้อมูลภาพได้น้อยกว่าฮัฟฟ์แมนโค้ดเล็กน้อย

#### การลดข้อมูลภาพด้วยวิธีเคเนอล

ข้อดี

- ภาพที่ได้กลับคืนมาใกล้เคียงกับภาพเดิม
- ลดข้อมูลภาพได้มาก

ข้อเสีย

- เสียเวลาในการคำนวณเข้ารหัส และถอดรหัส

#### การลดข้อมูลภาพด้วยวิธีฟูเรียร์ทรานสฟอร์ม

ข้อดี

- ภาพที่ได้กลับคืนมาใกล้เคียงกับภาพเดิม
- ลดข้อมูลภาพได้มาก

ข้อเสีย

- ใช้เนื้อที่หน่วยความจำมาก สำหรับการรันโปรแกรม
- การคำนวณเข้ารหัส และถอดรหัสยังช้า

#### การลดข้อมูลภาพด้วยวิธีดิสกรีต โครมากรานสฟอร์ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ภาพที่ได้กลับคืนมาใกล้เคียงกับภาพเดิม
  - ลดข้อมูลภาพได้มาก
  - ในการใช้งานภาพจริง การเก็บพลังงานของสัญญาณภาพทำได้ดีกว่าฟูเรียร์ทรานสฟอร์ม
- ข้อเสีย
- เวลาในการเข้ารหัส และถอดรหัสค่อนข้างช้า

#### การลดข้อมูลภาพด้วยวิธีฮาดามาร์ดทรานสฟอร์ม

ข้อดี

- ภาพที่ได้กลับคืนมาใกล้เคียงกับภาพเดิม
- ลดข้อมูลภาพได้มาก

ข้อเสีย

- เวลาในการเข้ารหัส และถอดรหัสค่อนข้างช้า

#### การลดข้อมูลภาพด้วยวิธีกำหนดตำแหน่งบิตในการทรานสฟอร์ม

ข้อดี

- ใช้เทคนิคการควอนไทซ์ภาพ ร่วมกับการกำหนดบิตให้มีความละเอียดเคลื่อนกำลังสองน้อยที่สุด ทำให้ลดข้อมูลภาพได้มากขึ้น

#### การลดข้อมูลภาพด้วยวิธีทานาย

ข้อดี

- การนำไปใช้งานได้ง่าย
- ลดข้อมูลภาพได้มาก

ข้อเสีย

- ความคลาดเคลื่อนเพิ่มมากขึ้น เมื่อลดข้อมูลภาพมากขึ้น

ตาราง 4.3 แสดงการเปรียบเทียบประสิทธิภาพการลดข้อมูลภาพแต่ละวิธี  
ความเร็วในการเข้ารหัสและถอดรหัสขึ้นกับความเร็วของคอมพิวเตอร์ คอมพิวเตอร์ที่ใช้ IBM PC XT

การลดข้อมูลภาพด้วยวิธี	เวลาเข้าและ ถอดรหัส	ความคลาดเคลื่อน	อัตราลดข้อมูลภาพ
1) วันเลขจ้	ทำได้รวดเร็ว	0%	ขึ้นกับข้อมูลภาพ
2) ฮัฟฟ์แมน	ค่อนข้างช้า	0%	ขึ้นกับความถี่ระดับ สีเทา
3) ฮัฟฟ์แมนบีพท์โคัด	เร็วกว่าฮัฟฟ์แมน	0%	ขึ้นกับความถี่ระดับ สีเทา
4) บล็อกทรูนเคชั่น	เร็วกว่าฮัฟฟ์แมน	ขึ้นกับขนาดบล็อก	ขึ้นกับขนาดบล็อก
5) เคแอล	ค่อนข้างช้า	ขึ้นกับจำนวนสัม ประสิทธิ์	ขึ้นกับจำนวนสัมประ สิทธิ์ที่เก็บ
6) พูเรียร์ทรานสฟอร์ม	ค่อนข้างช้า	ขึ้นกับจำนวนสัม ประสิทธิ์	ขึ้นกับจำนวนสัมประ สิทธิ์ที่เก็บ
7) ดิสกรีต โคซาย์ทราน สฟอร์ม	ค่อนข้างช้า	ขึ้นกับจำนวนสัม ประสิทธิ์	ขึ้นกับจำนวนสัมประ สิทธิ์ที่เก็บ
8) ฮาดามาร์ด	ค่อนข้างช้า	ขึ้นกับจำนวนสัม ประสิทธิ์	ขึ้นกับจำนวนสัมประ สิทธิ์ที่เก็บ
9) การกำหนดตำแหน่งบิต	-	-	ลดข้อมูลภาพได้มาก
10) ทำนาย	ใช้การกำหนดบิต ค่อนข้างช้า	-	ลดข้อมูลภาพได้มาก
11) ไอเอฟเอส	ค่อนข้างช้า	ขึ้นกับรายละเอียดของภาพ	ลดข้อมูลภาพได้มาก

ตาราง 4.4 ภาพที่เหมาะสมกับการลดข้อมูลแต่ละวิธี

การลดข้อมูลภาพด้วยวิธี	ภาพที่ใช้ได้เหมาะสม	ข้อจำกัด
1) วันเลขจ้	ภาพไบนารี ภาพเกรย์เลเวล ภาพสี	ใช้กับภาพที่ต่างระดับสีกัน มากไม่ได้
2) ยัฟฟ์แมน	ภาพเกรย์เลเวล ภาพสี	ภาพที่มีระดับสีเทามาก จะ ได้โคดยววมมาก
3) ยัฟฟ์แมนบิพท์โคด	ภาพเกรย์เลเวล ภาพสี	-
4) บล๊อคทรูนเคชั่น	ภาพเกรย์เลเวล	ไม่เหมาะสมกับภาพที่มีระ ดับเทามาก
5) เคนแอล	ภาพเกรย์เลเวล	-
6) ฟูเรียร์ทรานส์ฟอร์ม	ภาพเกรย์เลเวล	-
7) ดิสกรีตโคไซน์ทราน ส์ฟอร์ม	ภาพเกรย์เลเวล	-
8) ฮาดามาร์ด	ภาพเกรย์เลเวล	-
9) การกำหนดตำแหน่งบิท	ภาพเกรย์เลเวล	-
10) ทำนาศ	ภาพเกรย์เลเวล	-
11) ไอเอฟเอส	ภาพไบนารี	ภาพที่รายละเอียดมาก จะ เข้ารหัสได้ยาก

สรุปและวิจารณ์

การประมวลผลภาพทางดิจิทัลเป็นขบวนการที่กว้างขวาง ไม่ว่าจะเป็น การดิจิทัลไคร์ภาพ การทำให้ภาพคมชัด การสร้างภาพกลับคืนโดยการแก้ความคลาดเคลื่อน การสร้างภาพตัดขวางของวัตถุ การลดข้อมูลภาพ ได้รับความสนใจและถูกพัฒนาอย่างรวดเร็วเพื่อใช้งานในวงการต่าง ๆ อย่างกว้างขวาง

การลดข้อมูลภาพ เป็นงานวิจัยที่ต้องให้ความรู้ทางคณิตศาสตร์มาประยุกต์ใช้ในการเข้าได้ ถอดได้ การทรานส์ฟอร์มภาพด้วยเทคนิคต่าง ๆ กัน มีความเหมาะสมกับภาพแต่ละชนิด ทำให้ได้อัตราการลดข้อมูล ความคลาดเคลื่อนที่แตกต่างกัน

จากการศึกษาวิจัย ทำให้ผู้ใช้สามารถเลือกเทคนิคที่เหมาะสมกับภาพที่ต้องการได้ แล้วนำความรู้ที่ มาคิดค้นอัลกอริทึมที่เหมาะสม เพื่อนำมาโปรแกรมใช้กับไมโครคอมพิวเตอร์ ได้ทั้งภาพขาว-ดำ ภาพเกรย์เลเวล และภาพสี อีกทั้งเป็นแนวทางในการศึกษาเทคนิคที่ยิ่งขึ้นไป จนสามารถนำไปใช้งานได้ยังมีประสิทธิภาพ

นอกจากภาพเกรย์เลเวลที่มีความเข้มต่างกันถึง 256 ระดับ ดิสเพลย์บนจอทีวีให้ความคมชัดสูงแล้ว ปัจจุบันได้มีการนำภาพสีมาใช้งาน ดิสเพลย์บนจอสี ความละเอียดสูง สีเส้นสมจริง ได้รับความสนใจมาก แม้ข้อมูลภาพทั้ง 2 ชนิดมีความแตกต่างกัน ในโครงการนี้ก็ได้นพยายามทดลองลดข้อมูลภาพทั้ง 2 ชนิด

- สำหรับภาพเกรย์เลเวล สามารถลดข้อมูลภาพได้มาก น้อยตามเทคนิคแต่ละชนิด โดยใช้เครื่องคอมพิวเตอร์ IBM PC

- สำหรับภาพสี ใช้อุปกรณ์ที่มีประสิทธิภาพสูงในการประมวลผลภาพ ใช้ซอฟต์แวร์ช่วยในการดิจิทัลไคร์ภาพ การเก็บข้อมูล (ใช้ PC STORYBOARD) และทำการลดข้อมูลภาพโดยใช้เครื่อง IBM PS / II ทำให้สามารถลดข้อมูลภาพได้รวดเร็ว เมื่อใช้เทคนิคที่มีประสิทธิภาพสูงขึ้น ก็จะสามารถใช้ลดข้อมูลภาพ เพื่อใช้งานได้กว้างขวางขึ้นอีกมาก

โครงการนี้สำเร็จลงด้วยดี ประกอบด้วย

- การนำความรู้ด้านคณิตศาสตร์มาประยุกต์ใช้กับภาพ ซอฟต์แวร์ช่วยในการคำนวณ

แนวคิด (เช่น EUREKA) ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การศึกษาค้นคว้าอัลกอริทึมที่เหมาะสม ทำงานได้รวดเร็วถูกต้อง
- เครื่องมือที่มีประสิทธิภาพในการประมวลผลภาพ ได้แก่ ดวอนไตเซอร์ ดิจิตัลไตเซอร์ จอภาพทีวี กล้อง และซอฟต์แวร์ควบคุมการควอนไตซ์ ดิสเพลย์ภาพ การเก็บข้อมูลภาพ นอกจากนี้ความเร็วในการทำงานของเครื่อง จะช่วยในการประหยัดเวลาในการทำงานของโปรแกรม การเข้าไค้ด และถอดไค้ดอีกด้วย
- ผู้เชี่ยวชาญทางด้านการประมวลผลภาพทางดิจิทัล ให้คำแนะนำ
- ผู้ชำนาญด้านการโปรแกรม ภาษาที่ใช้ได้แก่ ภาษาซี เพราะทำงานได้รวดเร็ว มีความอ่อนตัว ใช้งานได้ไม่ขึ้นกับฮาร์ดแวร์แต่ละชนิด

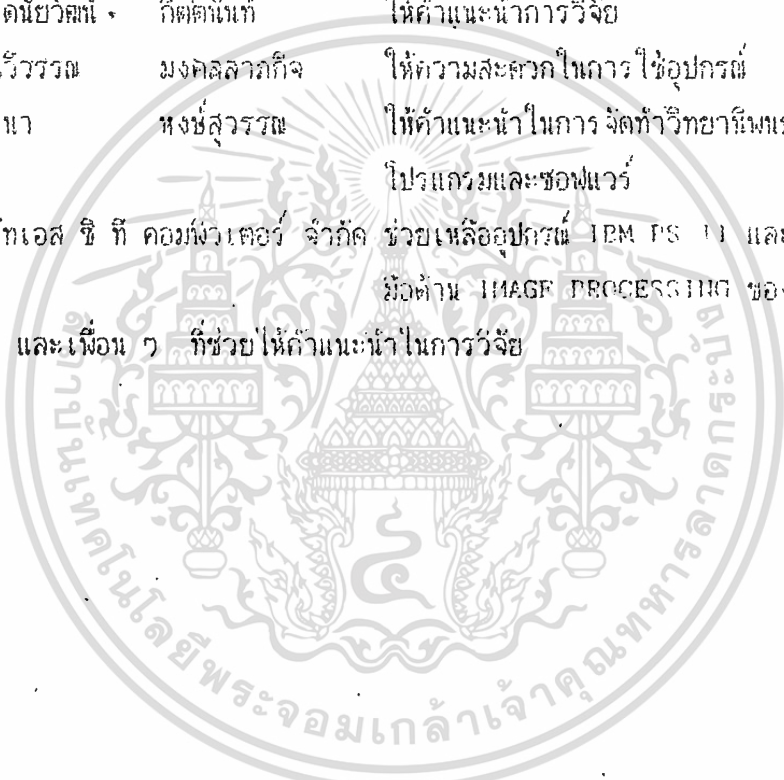
### วิจารณ์

- การควอนไตซ์ภาพเกรย์เลเวลต้องการความคมชัด ในบางครั้งจะมีนอส์ย (noise) รบกวน ทำให้ได้ข้อมูลภาพผิดพลาดบ้าง
- การนำอัลกอริทึมมาโปรแกรม มักมีปัญหากทำให้เสียเวลา และงานอาจล่าช้าได้ จำเป็นต้องให้ความพยายามในการทำโครงงานมาก

### กิติกรรมประกาศ

วิทยาลัยนอร์ทอีสต์ได้สำเร็จลุล่วงด้วยดี ด้วยคำแนะนำ ชี้นำ คำปรึกษาและการสนับสนุนทั้งทางด้านอุปกรณ์และความรู้ จาก ดร. บุญวัฒน์ อัครชู ซึ่งเป็นอาจารย์ที่ปรึกษาของโครงการงานนี้ อีกทั้งยังได้รับความช่วยเหลือจากบุคคลหลายฝ่าย จึงขอขอบคุณอย่างสูงมา ณ ที่นี้ด้วย

1. ดร. วรวัฒน์ ลิ้มโกศา ให้คำแนะนำในเนื้อหาของวิทยาลัยนอร์ท
2. ดร. ศุภมิตร จิตตะยโธธร ให้ความช่วยเหลือในการจัดทำวิทยาลัยนอร์ท
3. อ. ดนัยวัฒน์ กิตต์พันธ์ ให้คำแนะนำการวิจัย
4. คุณฉวีวรรณ มงคลลาภกิจ ให้ความช่วยเหลือในการใช้อุปกรณ์
5. คุณธนา หงษ์สุวรรณ ให้คำแนะนำในการจัดทำวิทยาลัยนอร์ท การโปรแกรมและซอฟต์แวร์
6. บริษัทเอส ซี ที คอมพิวเตอร์ จำกัด ช่วยเหลืออุปกรณ์ IBM PS 11 และเครื่องมื่อต้น IMAGE PROCESSING ของภาพสี
7. พี่ ๆ และเพื่อน ๆ ที่ช่วยให้คำแนะนำในการวิจัย



หนังสืออ้างอิง

1. R.P. Abbott, A differential pulse code modulation coded for video telephony using four bits per sample. IEEE Tran. Commu. Technol. COM-19. 1971,907-912.
2. N. Ahmed, T. Natarajan. and K.R.Rao, Discrete cosine transform, IEEE Tran. Commu. Technol. COM-23, 1974,90-93.
3. G. B. Anderson and T.S. Huang, Picture bandwidth compression by piecewise Fourier transformation, IEEE Tran. Commu. Technol. COM-19,1971,133-140.
4. H. C. Andrews, Some unitary transformatoons in pattern recognition and image procsssing, in "Information Processing 71" pp. 155-160 North-Holland Publ. Amsterdam,1972.
5. R.J. Arguello, H.R. Sellner, and J.A. Stuller, The effect of channel errors in the differential pulse-code modulation transmission of sampled imagery, IEEE Tran. Commu. Technol. COM-19,1971,926-933.
6. Michael F. Barnsley and Alan D. Sloan ,Byte January 1988, vol.13 Number 1 pp.215-223.
7. J.S. Wezska, "A survey of threshold selection techniques," Comput. Graph. Image Process. vol. 7, pp.259-265,1978.
8. W.Chen and C.H. Smith, Adaptive coding of monochrome and color images, IEEE Tran. Commu. Technol. COM-25,1977,1285-1291.
9. W. Chen, C.H. Smith, and S. Fralick, A fast computational algorithm for the discrete cosine transform, IEEE Tran. Commu. Technol. COM-25,1977,1285-1291.
10. W.H. Chen and W.K. Pratt, Color Image Coding with the Slant Transform, Proc.1973 Symp. Appl. Walsh Functions pp.155-161,

April 1973.

11. C.Cherry, M.H.Kubba, D.E.Pearson and M.P.Barton, An experimental study of the possible bandwidth compression of visual image signals, Proc. IEEE 51, 1963, 1507-1517.
12. D.J. Connor, R.F.W. Peade, and W.G. Scholes, Television coding using two dimensional spatial prediction, BSTJ 50, 1971, 1049-1061.
13. C.C.Culter, Differential Quantization of Communication Signals, Patent No. 2,605,361, July 29, 1952.
14. L.D.Davisson, Rate distortion theory and application, Proc. IEEE 60, 1972, 800-808.
15. E.J.Delp and P.R.Mitchell, Image truncation using block truncation coding, IEEE Tran. Comm. Technol. COM-27, 1979, 1335-1342.
16. H.Enomoto and K.Shibata, Orthogonal transform coding system for television signals, Television J, Inst, TV Eng. Japan 24, 1970, 99-108; also in Proc. 1971 Symp. Appl. Walsh Functions pp.11-17. April 1971.
17. D.Estournet, Compression d'information de signaux d'images par les systeme differential codes, Onde Elec. 49, 1969, 858-867.
18. B.Fox, Discrete optimization via marginal analysis, Management Sci, November 1966, 201-216.
19. R.G.Gallager, "Information Theory and Reliable Communication." Wiley, New York, 1968.
20. N.C.Gallagher and B. Liu, Statistical properties of the Fourier transform of random phase diffusers, Pptik 42, 1976, 65-86.
21. N.C.Gallagher, Jr., Quantizing schemes for the discrete Fourier transform of a random time series, IEEE Trans. Informat, Theory IT-24, 1978, 156-164.

22. A.Gersho, Asymptotically optimal block quantization, IEEE Tran.

23. L.S.Golding and R.Garlow, Frequency interleaved sampling of a color television signal, IEEE Trans. Informat, COM-19, 1971, 972-979.
24. R.C. gonzalez and P.A. Wintq, "Digital Image Processing." Addison-Wesley, Reading, Massachusetts, 1977.
25. G.G.Gouriet, Bandwidth compression of television signals, IEE Proc(London) 104B, 1957, 256-272.
26. D.N.Graham, Image transmission by two-dimension contour coding, Proc IEEE 55, 1967, 336, 346.
27. R.E.Graham, Predictive quantizing of television signals, IRE Wescon Content. Record Pt.4, 1958, 147-156.
28. V.Grenander and G.Szego. "Toeplitz Forms and their Applecations." "Springer-Verlag, Berlin and New York, 1969.
29. A. Habibi and P.A. Wintz, Image coding by linear transformation and block quantization, IEEE Trans. Commun. Technol. COM-19, 1971, 50-62.
30. A.Habibi, Comparison of nth-order DPCM encoder with linear transformation and block quantization, IEEE Trans. Commun. Technol. COM-19, 1971, 948-956.
31. A.Habibi and G.S.Robinson, A survey of digital coding, Computer 7, 1974, 22-33.
32. A.Habibi, Hybrid coding of pictoreal data, IEEE Trans. COM-22, 1974, 614-612.
33. B.G. Haskell, F.W.Mounts, and T.C.Candy, Tnterframe coding of videotelephone pictures, Proc. IEEE 60, 1972, 792-800.
34. B.G. Haskell, P.L. Gordon, R.L.Schmidt, and J.V.Scattaglia, Interframe coding of 525-line, monochrome television at 1.5 M bits/sec, IEEE Trans Commun. COM-25, 1977, 1339-1348.