การพัฒนาคิวอาร์โค้ดร่วมกับลายน้ำดิจิตอลเพื่อซ่อนข้อมูล

นางสาวสุพพัต รุ่งเรืองศิลป์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2554
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ที่ส่งผ่านทางบัณฑิตวิทยาลัย
5 3 7 0 3 6 8 1 2 1
The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository(CUIR)
are the thesis authors' files submitted through the Graduate School.

DEVELOPMENT OF QR CODE WITH WATERMARK FOR HIDING DATA

Miss Suppat Rungraungsilp

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering Program in Computer Engineering
Department of Computer Engineering
Facalty of Engineering
Chulalongkorn University
Academic Year 2011
Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การพัฒนาคิวอาร์โค้ดร่วมกับลายน้ำดิจิตอลเพื่อซ่อนข้อมูล
โดย	นางสาวสุพพัต รุ่งเรืองศิลป์
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร.สาธิต วงศ์ประทีป

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

.... คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.บุญสม เลิศหิรัญวงศ์)

คณะกรรมการสอบวิทยานิพนธ์

.... ประธานกรรมการ
(ศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา)

.... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(รองศาสตราจารย์ ดร.สาธิต วงศ์ประทีป)

.... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ์)

.... กรรมการภายนอกมหาวิทยาลัย
(ผู้ช่วยศาสตราจารย์ ดร.วิไลพร แซ่ลี้)

สุพพัต รุ่งเรืองศิลป์ : การพัฒนาคิวอาร์โค้ดร่วมกับลายน้ำดิจิตอลเพื่อซ่อนข้อมูล (DEVELOPMENT OF QR CODE WITH WATERMARK FOR HIDING DATA). อ.ที่ปรึกษาวิทยานิพนธ์หลัก : รศ. ดร.สาธิต วงศ์ประทีป, 169 หน้า.

วิทยานิพนธ์ฉบับนี้เป็นการนำเสนอวิธีการฝังลายน้ำดิจิตอลและการถอดลายน้ำดิจิตอลบนคิวอาร์โค้ด ซึ่งผ่านการทรานฟอร์ม 3 แบบ ได้แก่ Discrete Cosine Transform, Discrete Fourier Transform และ Discrete Wavelet Transform ทำให้สามารถซ่อนข้อมูลในคิวอาร์โค้ดได้ โดยผลการทดลองแสดงให้เห็นว่าการฝังลายน้ำดิจิตอลและการถอดลายน้ำดิจิตอลบนคิวอาร์โค้ด แบบใดที่ทำให้ถอดลายน้ำดิจิตอลออกมาได้สมบูรณ์มากที่สุด แบบใดใช้เวลาในการฝังและถอดลายน้ำดิจิตอลน้อยสุด ทำให้เห็นข้อดี-ข้อเสียของแต่ละแบบ โดยการฝังลายน้ำดิจิตอลและการถอดลายน้ำดิจิตอลต้องผ่านการเข้ารหัส-ถอดรหัสข้อมูลตามมาตรฐานของคิวอาร์โค้ดร่วมด้วย

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต..

สาขาวิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก ....

ปีการศึกษา 2554

5370368121 : MAJOR COMPUTER ENGINEERING

KEYWORDS : QR CODE / WATERMARK / HIDING DATA

SUPPAT RUNGRAUNGSILP : DEVELOPMENT OF QR CODE WITH WATERMARK FOR HIDING DATA. ADVISOR : ASSOC. PROF. SARTID VONGPRADHIP, Ph.D., 169 pp.

In this thesis we proposed embedding and extracting a watermark on QR Code based on Discrete Cosine Transform, Discrete Fourier Transform and Discrete Wavelet Transform that preserve an information hiding text on QR Code. The experimental results shown which one is the best extracting, use the least time in embedding a watermark and use the least time in extracting a watermark. For each transform we obtained advantages and disadvantages by embedding and extracting a watermark through a standard encode and decode QR Code method.

Department: Computer Engineering Student's Signature ....

Field of Study: Computer Engineering Advisor's Signature ....

Academic Year: 2011

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี ขอกราบขอบพระคุณรองศาสตราจารย์ ดร. สาธิต วงศ์ประทีป ซึ่งเป็นอาจารย์ที่ปรึกษาวิทยานิพนธ์ ท่านได้กรุณาสละเวลาและดูแลการวิจัย ให้คำแนะนำเกี่ยวกับการทำวิจัย และช่วยชี้แนะการแก้ไขปัญหาต่างๆ ที่เกิดขึ้นระหว่างการทำวิจัย

ขอกราบขอบพระคุณคณะกรรมสอบวิทยานิพนธ์ทุกท่านเป็นอย่างสูง ได้แก่ ศาสตราจารย์ ดร. ประภาส จงสถิตย์วัฒนา ผู้ช่วยศาสตราจารย์ ดร. อาทิตย์ ทองทักษ์ และผู้ช่วยศาสตราจารย์ ดร. วิไลพร แซ่ลิ้ ในการตรวจแก้ ให้ข้อคิดและคำแนะนำอันเป็นประโยชน์ยิ่งต่องานวิจัย

ขอขอบพระคุณอาจารย์ประจำภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัยทุกท่านเป็นอย่างสูงที่ให้ข้อคิดและแนวทางในการวิจัย ขอขอบคุณเจ้าหน้าที่ประจำภาควิชาวิศวกรรมคอมพิวเตอร์ทุกท่าน รวมถึงเพื่อนๆ และพี่น้องทุกคนในห้องวิจัย ISEL Lab, MIND Lab, SPACE Lab โดยเฉพาะ Digital System Engineering Laboratory (DSEL) ที่ให้ความช่วยเหลือ คำแนะนำ และยังให้กำลังใจตลอดมา ขอบคุณ นายมหศักดิ์ เกตุฉ่ำ ที่เสียสละเวลาในการตรวจและแก้ไขงานพรีเซนต์หลายครั้ง พร้อมทั้งให้คำแนะนำมากมาย นายนริศ หนูหอม และนายประทาน โชติพันธุ์บัณฑิตย์ ที่ให้ไฟล์งานและเอกสารมาศึกษางานวิจัย

ขอกราบขอบพระคุณ ดร.สุรสิทธิ์ รุ่งเรืองศิลป์ นางสมทรง รุ่งเรืองศิลป์ รวมถึงขอขอบคุณนางสาวพจนีย์ พาหุรัตน์ บิดามารดาและคุณป้าของผู้วิจัยซึ่งให้การสนับสนุน คอยเป็นห่วงเป็นใย และให้กำลังใจแก่ผู้วิจัยเสมอมา รวมถึงนางสาวณัฏฐขวัญ พาหุรัตน์ ที่ช่วยเหลือในงานพรีเซนต์และนางสาวอรอภิญญ์ ชัยชลทรัพย์ ที่คอยช่วยเหลือแก้ไขไฟล์งานเอกสารให้ลุล่วงไปได้ด้วยดี

ท้ายนี้ผู้วิจัยขอขอบคุณเพื่อนๆ ที่มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ โดยเฉพาะนายพีรนิษฐ์ ตุวพันธพิพัฒน์ และนายสุเมธ สิริเรืองอำไพ ที่สละเวลารับฟังปัญหาต่างๆ พร้อมถกเถียงทำให้เกิดไอเดียซึ่งเป็นประโยชน์ต่อวิทยานิพนธ์ฉบับนี้ด้วย นอกจากนี้ผู้วิจัยขอขอบคุณทุกท่านที่มีส่วนช่วยเหลือทำให้วิทยานิพนธ์ฉบับนี้สำเร็จเรียบร้อยลงได้ด้วยดีทุกประการ

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง

บทคัดย่อภาษาอังกฤษ ..... จ

กิตติกรรมประกาศ..... ฉ

สารบัญ ..... ช

สารบัญตาราง..... ฎ

สารบัญภาพ ..... ฏ

บทที่

1 บทนำ..... 1

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 วัตถุประสงค์ของการวิจัย 3

1.3 ขอบเขตของการวิจัย..... 3

1.4 ข้อตกลงเบื้องต้น 4

1.5 ประโยชน์ที่คาดว่าจะได้รับ 5

1.6 ลำดับขั้นตอนในการเสนอผลการวิจัย..... 5

2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง..... 6

2.1 โครงสร้างของ QR Code (Structure of QR Code) 6

2.1.1 Finder Patterns 9

2.1.2 Separator..... 10

2.1.3 Timing Pattern..... 10

2.1.4 Alignment Patterns 10

2.1.5 Encoding Region..... 10

2.1.6 Quiet Zone 10

2.2 ขั้นตอนการเข้ารหัส (Encode Procedure Overview) 10

บทที่ หน้า

2.2.1 วิเคราะห์ข้อมูล (Data analysis) 10
2.2.2 การเข้ารหัสข้อมูล (Data encoding) 11
2.2.3 พิจารณา Error Correcting Codewords (Error correction coding) 16
2.2.4 Finally codewords (Structure final message) 21
2.2.5 การวาง Codewords (Module placement in matrix) 22
2.2.6 การวาง Data Masking (Data masking) 25
2.2.7 Format and version information 28
2.3 ขั้นตอนการถอดรหัส (Decode Procedure Overview) 32
2.3.1 Recognize Black/White Modules 32
2.3.2 Decode Format Information 32
2.3.3 Determine Version 32
2.3.4 Release Masking 32
2.3.5 Restore Data and RS Codewords 32
2.3.6 Error Detection 32
2.3.7 Error Correction 32
2.3.8 Decode Data Codewords 32
2.4 พื้นฐานเบื้องต้นเกี่ยวกับภาพ (Principle of image) 41
2.4.1 แบบจำลองสี RGB 41
2.4.2 แบบจำลองสี CMY หรือ CMYK 42
2.4.3 แบบจำลองสี HSV 43
2.5 ลายน้ำดิจิตอล (Digital Watermark) 44
2.5.1 นิยามของลายน้ำดิจิตอล 44
2.5.2 หลักการในการทำลายน้ำดิจิตอล 45
2.5.3 วิธีการฝังลายน้ำดิจิตอล 45
2.5.4 การฝังลายน้ำดิจิตอล (Watermark embedding) 47
2.5.5 การตรวจหาลายน้ำดิจิตอล (Watermark detection) 47
2.5.6 การถอดลายน้ำดิจิตอล (Watermark Extraction) 48

บทที่ หน้า

2.5.7 เกณฑ์การวัดประสิทธิภาพของลายน้ำดิจิตอล 48

2.6 เทคนิคการกระบวนการฝังและถอดลายน้ำในโดเมนความถี่ 50

2.6.1 การแปลงโคซายน์แบบไม่ต่อเนื่อง (Discrete cosine transform, DCT) 51

2.6.2 การแปลงฟูเรียร์แบบไม่ต่อเนื่อง (Discrete fourier transform, DFT) 54

2.6.3 การแปลงเวฟเล็ตแบบไม่ต่อเนื่อง (Discrete wavelet transform, DWT) 55

2.7 เอกสารและงานวิจัยที่เกี่ยวข้อง 58

3 วิธีดำเนินการวิจัย 63

3.1 ภาพรวมทั้งหมดของงานวิจัย 63

3.1.1 การเข้ารหัสข้อมูล QR Code 63

3.1.2 กระบวนการฝังลายน้ำดิจิตอล 63

3.1.3 กระบวนการถอดลายน้ำดิจิตอล 64

3.1.4 การถอดรหัสข้อมูล QR Code 64

3.2 การเข้ารหัสข้อมูล QR Code (Encode QR Code Procedure) 66

3.3 กระบวนการฝังลายน้ำดิจิตอล (Embed Watermark in QR Code Procedure) 66

3.4 การถอดลายน้ำดิจิตอล (Extract Watermark in QR Code Procedure Overview) 77

3.5 การถอดรหัสข้อมูล QR Code (Decode QR Code Procedure) 79

3.6 การวัดค่าประสิทธิภาพ 79

4 การทดลองและผลการทดลอง 81

4.1 เครื่องมือที่ใช้ในการทดลอง 81

4.1.1 ซอฟต์แวร์ที่ใช้ในการเขียนโปรแกรมคอมพิวเตอร์ 81

4.1.2 ฮาร์ดแวร์ที่ใช้ในการทดสอบ 81

4.1.3 รูปภาพที่ใช้ในการทดลอง 81

4.2 ผลการทดลอง 82

5 สรุปผลการวิจัย 121

5.1 บทสรุป 121

5.2 ข้อเสนอแนะ 122

รายการอ้างอิง 123

หน้า

ภาคผนวก.. 130
ประวัติผู้เขียนวิทยานิพนธ์... 169

สารบัญตาราง

ตารางที่ หน้า

2-1 แสดงความจุในแต่ละชนิดของอักขระ....8
2-2 Mode Indicators ของ QR Code 11
2-3 จำนวนบิต Character Count Indicator ของ QR Code 12
2-4 แสดงค่า Encoding และ Decodingของ Alphanumeric Mode....13
2-5 แสดงระดับของ Error Correcting 16
2-6 เงื่อนไขการเลือก Data mask pattern....26
2-7 คะแนนของผล Data Masking 28
2-8 Indicators ของ Error Correction Level ใน QR Code 29
4-1 แสดงผลลัพธ์ในกระบวนการก่อนและหลังฝังลายน้ำดิจิตอล....82
4-2 แสดงผลลัพธ์ในกระบวนการก่อนและหลังฝังลายน้ำดิจิตอล 91

สารบัญภาพ

ภาพที่ หน้า

2-1 ขั้นตอนการถอดรหัสภาพบาร์โค้ดแบบ 1D Barcode 7
2-2 ขั้นตอนการถอดรหัสภาพบาร์โค้ดแบบ QR Code (2D Barcode) 7
2-3 แสดงให้เห็นข้อความมากมายกลายเป็นแค่ 1 สัญลักษณ์ 8
2-4 โครงสร้างของ QR Code 9
2-5 อัตราส่วนโมดูลดำ-ขาวของ Finder Patterns 9
2-6 รูปแบบการวาง Codewords จากตัวอย่าง 22
2-7 การวางบิตในรูปแบบปกติในทิศทางขึ้นและลง 23
2-8 ทิศทางการวางบิต (i) เป็นแบบปกติและ (ii) แบบไม่ปกติมีการเปลี่ยนทิศการวาง 24
2-9 การวางพื้นที่ที่จะวางถูกครอบครองโดยข้อมูลอื่น 24
2-10 ตัวอย่างการจัดเตรียมการวางในเวอร์ชัน 2 ที่ Error Correcting Level ระดับ M 25
2-11 รูปแบบ Data Mask Pattern ทั้งหมด 27
2-12 การแสดงขั้นตอนเลือกData Masking ใน QR Code 27
2-13 ตำแหน่งของ Format Information 30
2-14 ตำแหน่ง Version Information 31
2-15 การเก็บค่าไบนารีของ Version Information ฝั่งซ้ายล่าง 31
2-16 การเก็บค่าไบนารีของ Version Information ฝั่งขวาบน 31
2-17 แสกนเส้นใน Finder Pattern 33
2-18 ความกว้างของ Finder Pattern ซ้ายบนและขวาบน 34
2-19 ตำแหน่ง AC และ AB จากจุด A, B และ C 35
2-20 ตำแหน่ง Finder Patterns และ Alignment Pattern 36
2-21 คู่อันดับกึ่งกลางของ Alignment Pattern 37
2-22 พื้นที่ฝั่งซ้ายบนของ QR Code 39
2-23 แสดงพื้นที่ล่างขวาของ QR Code 40
2-24 โครงสร้างของสีหลัก RGB 42
2-25 โครงสร้างของสีหลักของ CMY หรือ CMYK 43

ภาพที่ หน้า

2-26 โครงสร้างของสีหลักของ HSV 44
2-27 โครงสร้างของวิธีซ่อนข้อมูล 46
2-28 การฝังลายน้ำดิจิตอลลงในข้อมูล QR Code 47
2-29 การตรวจหาลายน้ำดิจิตอลหรือข้อมูลลับที่อาจฝังอยู่ในข้อมูลที่ผ่านการฝังลายน้ำ 48
2-30 การถอดลายน้ำดิจิตอลออกจากข้อมูลที่มีลายน้ำดิจิตอลฝังอยู่ 48
2-31 ความหมายของขอบเขต DCT ช่วงความถี่กลาง 51
2-32 ความหมายของขอบเขต DFT ช่วงความถี่ต่ำ 55
2-33 การแยกส่วนประกอบโดยใช้การแปลงเวฟเล็ต 3 ระดับ 57
2-34 การแปลงแบบ Forward Wavelet Transform 57
3-1 โครงสร้างทั้งหมดของงานวิจัย 65
3-2 กระบวนการเข้ารหัสของ QR Code 66
3-3 โครงร่างของเทคนิคการทำลายน้ำดิจิตอลแบบ DCT ในกระบวนการฝังลายน้ำดิจิตอล...... 67
3-4 โครงร่างของเทคนิคการทำลายน้ำดิจิตอลแบบ DCT ในกระบวนการถอดลายน้ำดิจิตอล... 67
3-5 การแบ่งข้อมูล QR Code ต้นฉบับออกเป็นบริเวณไม่ซ้อนทับกัน 68
3-6 การเลือกค่าสัมประสิทธิ์ของความถี่กลางด้วยวิธี Zigzag 69
3-7 โครงร่างของเทคนิคการทำลายน้ำดิจิตอลแบบ DFT ในกระบวนการฝังลายน้ำดิจิตอล 70
3-8 โครงร่างของเทคนิคการทำลายน้ำดิจิตอลแบบ DFT ในกระบวนการถอดลายน้ำดิจิตอล ... 70
3-9 การแบ่งข้อมูล QR Code ต้นฉบับออกเป็นบริเวณไม่ซ้อนทับกัน 71
3-10 การเลือกค่าสัมประสิทธิ์ของความถี่กลางด้วยวิธี Zigzag 72
3-11 โครงร่างของเทคนิคการทำลายน้ำดิจิตอลแบบ DWT ในกระบวนการฝังลายน้ำดิจิตอล 73
3-12 โครงร่างของเทคนิคการทำลายน้ำดิจิตอลแบบ DWT ในกระบวนการถอดลายน้ำดิจิตอล .. 68
3-13 การแบ่งข้อมูล QR Codeต้นฉบับออกเป็นบริเวณละ 16×16จุดภาพ 74
3-14 เวฟเล็ตแบบไม่ต่อเนื่องระดับที่ 1 และ 2 ของข้อมูล QR Code ต้นฉบับ 75
3-15 การแบ่งภาพลายน้ำดิจิตอลเป็นบริเวณขนาด 4×4 จุดภาพ 75
3-16 แทรกบิตลายน้ำลงในสัมประสิทธิ์ข้อมูล QR Code ต้นฉบับผ่านตารางเทียบค่าไบนารี 77
3-17 การถอดบิตลายน้ำออกจากสัมประสิทธิ์ภาพทดสอบโดยผ่านตารางเทียบค่าไบนารี 78
3-18 กระบวนการถอดรหัสของ QR Code 79

ภาพที่ หน้า

4-1 แผนภาพค่า PSNR ของทุกภาพ 88
4-2 แผนภาพค่า NC ของทุกภาพ 89
4-3 แผนภาพค่า Time แต่ละ PSNR ของทุกภาพ...89
4-4 แผนภาพค่า Time แต่ละ PSNR ของทุกภาพ...90
4-5 แผนภาพค่า PSNR ของทุกภาพ 119
4-6 แผนภาพค่า NC ของทุกภาพ 119
4-7 แผนภาพค่า Time แต่ละ PSNR ของทุกภาพ...120
4-8 แผนภาพค่า Time แต่ละ NC ของทุกภาพ...120

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

QR Code ย่อมาจาก Quick Response Code ซึ่งเป็นที่รู้จักกันดีในวงการบาร์โค้ด เพราะความสามารถในการอ่านได้อย่างถูกต้องและรวดเร็ว ซึ่งกลายเป็นที่รู้จักในกลุ่มคนทั่วไปในประเทศไทยมากขึ้นเมื่อบริษัท Blackberry ให้ผู้ใช้สามารถสร้าง QR Code ได้ด้วยตนเองเพื่อแสดง PIN BB ของเครื่องผู้ใช้งาน นอกจากนั้น QR Code ยังได้รับการยอมรับในระดับสากลดังที่เห็นได้จากการมีมาตรฐานมารองรับในปัจจุบัน [1] QR Code ถูกพัฒนาเรื่อยมาโดยบริษัท Denso Wave [2] โดยครั้งแรกถูกพัฒนาขึ้นเพื่อให้เป็นสัญลักษณ์ที่ง่ายต่อการตีความหมายด้วยแสกนเนอร์ ต่อมาถูกพัฒนาให้ใช้ในสายการผลิตในวงการอุตสาหกรรม [3] จนทำให้เกิดการแข่งขันทางการตลาดที่นำเสนอโดยการอ่าน QR Code ผ่านอุปกรณ์เครื่องมือสื่อสาร [4],[5],[6],[7] หรือกระทั่งการที่ผู้ใช้สามารถสร้าง QR Code ได้เองผ่านเว็บไซต์ผู้ให้บริการ [3] จนปัจจุบันได้มีการพัฒนาให้สามารถเก็บตัวอักขระได้มากขึ้นกว่าสมัยก่อนโดยในปัจจุบันสามารถ
เก็บอักขระที่เป็นเฉพาะตัวเลขได้มากสูงสุดถึง 7,089 ตัว นอกจากนั้น QR Code ยังสามารถบรรจุข้อมูลได้หลายรูปแบบ เช่น Numeric Characters, Alphabetic Characters, Kanji Characters, Kana Characters, Hiragana Characters, Symbols, Binary และ Control Code [8] ด้วยความสามารถของ QR Code ซึ่งถือว่าเป็น 2D Barcode (Two Dimension Barcode) ที่บรรจุข้อมูลได้มาก ราคาถูกและสามารถลดทอนความเสียหายนั้นทำให้มีศักยภาพในการพัฒนาได้หลากหลายแอพพลิเคชั่น ประโยชน์ของ QR Code มีประโยชน์มากมาย เช่น การบริหารการจัดการด้านโลจิสติก ความสามารถในการแลกเปลี่ยนและการระบุข้อมูล [9] เมื่อไม่กี่ปีมานี้รัฐบาลไต้หวันและในส่วนภาคอุตสาหกรรมได้พยายามอย่างมากเพื่อโปรโมท 2D Barcode ในงานวิจัยที่เกี่ยวข้อง ด้วยเหตุนี้เองทำให้เกิดผลตามมาดังนี้ อุตสาหกรรมผลิตถูกประยุกต์ใช้กับเทคโนโลยี 2D Barcode ในวงการธุรกิจของการจัดการ โรงงานอุตสาหกรรมและการให้ค่าตอบแทนต่างๆ เหล่านี้เพิ่มมากขึ้น [10] ลักษณะโดยทั่วไปของ 2D Barcode มีลักษณะความปลอดภัยที่เฉพาะตัวและยากต่อการปลอมแปลง เพราะจากการฝังข้อมูลที่ไม่สามารถถอดความออกมาได้โดยง่ายนักจากการดูด้วยสายตามนุษย์ ถึงอย่างนั้นก็ตามหากมีบุคคลที่คุ้นเคยในเทคโนโลยี 2D Barcode ความปลอดภัยของข้อมูลก็จะไม่สมบูรณ์นัก ดังที่เห็น ZXing Decoder Online [11] เพียงแค่เราอัพ

โหลดไฟล์ภาพ QR Code ผู้ใช้ก็สามารถถอดความหมายของ QR Code ได้ จึงเห็นได้ว่า QR Code มีจุดอ่อนในเรื่องความปลอดภัยดังที่มีงานวิจัยฉบับหนึ่งของ Manabu Hirakawa และ Junichi Iijima ในปี 2009 [12] ได้กล่าวไว้ว่ามีความปลอดภัยหลายด้านตระหนักถึง QR Code ว่ามีจุดอ่อนในด้านการรักษาความลับ ดังนั้นงานวิจัยชิ้นนี้จึงทำการศึกษาด้านความปลอดภัยในการรักษาความลับของ QR Code [13] โดยอาศัยวิธีบางประการเพื่อป้องกันไม่ให้บุคคลที่สามที่ไม่มีส่วนเกี่ยวข้องสามารถเข้าถึงข้อมูลที่ต้องการซ่อนได้ โดยจากการศึกษาพบว่ามีลายน้ำดิจิตอลที่สามารถนำมาประยุกต์ใช้ในงานวิจัยนี้ได้ จากการเติบโตของเทคโนโลยีทางการสื่อสารทางอินเตอร์เน็ตอย่างรวดเร็ว ทำให้เกิดการแพร่กระจายของข้อมูลไปทั่วโลกจากเพียงแหล่งข้อมูลแห่งเดียว ซึ่งหากข้อมูลดังกล่าวไม่ได้รับอนุญาตจากเจ้าของผลงานหรือข้อมูลก่อนที่จะถูกคัดลอกหรือเผยแพร่นั้น ผลกระทบทางตรงก็คือเจ้าของผลงานหรือข้อมูลก็จะเกิดการเสียผลประโยชน์ในส่วนที่ควรจะได้รับ ผลกระทบทางอ้อมก็ คือ ระบบเศรษฐกิจ เพื่อหยุดหรือลดปัญหาดังกล่าวจึงเกิดการจดลิขสิทธิ์ (Copyright) เพื่อยืนยันความเป็นเจ้าของของผลงานหรือข้อมูลนั้นๆ นอกจากการจดลิขสิทธิ์แล้วยังมีวิธีอื่นๆ ที่ใช้ป้องกันการละเมิดลิขสิทธิ์ เช่น การเข้ารหัสลับข้อมูล (Encryption) ที่อนุญาตให้เฉพาะผู้ที่มีกุญแจรหัสรับเท่านั้นจึงจะสามารถเข้าถึงข้อมูลได้หรือการใส่เลขทะเบียน (Serial Numbers) ตราสัญลักษณ์(Logo) ไว้เสมือนเป็นตำหนิ (Label) ไว้ที่ตัวของข้อมูลเพื่อบ่งบอกให้บุคคลทั่วไปทราบว่าข้อมูลมีเจ้าของ แต่ถึงกระนั้นก็ยังไม่สามารถหยุดหรือลดปัญหาดังกล่าวได้ เช่น ข้อมูลที่ผ่านการเข้ารหัสแล้วสามารถนำมาคัดลอกหรือทำซ้ำ ดังนั้นเพื่อหยุดหรือลดปัญหาดังกล่าวจึงเกิดการพัฒนาเทคโนโลยีการทำภาพพิมพ์ลายน้ำดิจิตอล (Digital Watermarking) ซึ่งต่างจากการเข้ารหัสทั่วไปที่การเข้าถึงข้อมูลสามารถเข้าถึงได้เฉพาะผู้ที่ถือกุญแจเท่านั้น แต่เทคโนโลยีการทำภาพพิมพ์ลายน้ำดิจิตอลมีการซ่อนสัญญาณลายน้ำ (Watermark Signal) ลงไปในข้อมูลมัลติมีเดีย ซึ่งหากเกิดการคัดลอกสัญญาณลายน้ำจะติดไปกับข้อมูลมัลติมีเดียที่ถูกคัดลอกไปด้วย ซึ่งส่วนนี้สามารถนำไปใช้ในการสืบหาแหล่งกำเนิดของการคัดลอกข้อมูลหรือใช้แสดงความเป็นเจ้าของได้ด้วย โดยหลักการแล้วกระบวนการฝังลายน้ำดิจิตอลเป็นหลักการที่ทำให้มนุษย์ไม่สามารถรับรู้ได้ทันทีด้วยประสาทสัมผัสทางตาของมนุษย์ หรือไม่สามารถสังเกตความแตกต่างระหว่างข้อมูลต้นฉบับกับข้อมูลที่ผ่านกระบวนการฝังลายน้ำดิจิตอลได้ โดยข้อมูลที่เราใช้ในการฝังเรียกว่า “ลายน้ำดิจิตอล” การทำลายน้ำดิจิตอลบน QR Code ถือว่าเป็นทางเลือกหนึ่งที่ใช้ในการแก้ปัญหาดังกล่าว โดยใช้ข้อมูลที่มีข้อความเป็นลายน้ำดิจิตอลแทรกลงไปในข้อมูล QR Code เพื่อซ่อนข้อมูล โดยจะทำการรวมลายน้ำดิจิตอลเข้ากับข้อมูล QR Code ที่เกิดจากการเข้ารหัสข้อมูลเข้าด้วยกันซึ่งจะทำให้แนวทางการวิจัยชิ้นนี้บรรลุผล

ได้ จากที่กล่าวมาข้างต้นเราทำการศึกษาต่อว่าเมื่อเกิดกระบวนการฝังลายน้ำดิจิตอลบนข้อมูล QR Code การแทรกและถอดลายน้ำดิจิตอลบนโดเมนความถี่ (Frequency Domain) โดยใช้วิธีการแปลง 3 วิธี คือ Discrete Cosine Transform(DCT), Discrete Fourier Transform(DFT) และ Discrete Wavelet Transform(DWT) นั้นแบบใดที่ทำให้ข้อมูลที่ผ่านการฝังแล้วได้ผลที่ดีกว่ากัน

1.2 วัตถุประสงค์ของการวิจัย

เพื่อศึกษา ออกแบบและนำเสนอวิธีการฝังลายน้ำดิจิตอลในข้อมูล QR Code เพื่อซ่อนข้อมูลในโดเมนความถี่โดยใช้วิธีการแปลง 3 วิธี อันได้แก่ การแปลงโคซายน์แบบไม่ต่อเนื่อง (Discrete cosine transform, DCT), การแปลงฟูเรียร์แบบไม่ต่อเนื่อง (Discrete fourier transform, DFT) และ การแปลงเวฟเล็ตแบบไม่ต่อเนื่อง (Discrete wavelet transform, DWT) เปรียบเทียบกัน

1.3 ขอบเขตของการวิจัย

1. เป็นการฝังลายน้ำดิจิตอลในข้อมูล QR Code เพื่อซ่อนข้อมูล
2. ในงานวิจัยนี้สนใจการเข้ารหัส Numeric และ Alphanumeric
3. ข้อมูล QR Code ที่เข้ารหัสใช้ Error Correction ที่ระดับ L
4. การแทรกและถอดลายน้ำดิจิตอลบนโดเมนความถี่ (Frequency Domain) โดยใช้วิธีการแปลง 3 วิธี คือ Discrete Cosine Transform (DCT), Discrete Fourier Transform (DFT) และ Discrete Wavelet Transform (DWT)
5. เกณฑ์การวัดคุณภาพในการแปลง 2 แบบ คือ

   5.1 การใช้เกณฑ์การวัดคุณภาพของภาพด้วย PSNR และ NC

   5.2 เกณฑ์การวัดประสิทธิภาพการคำนวณของเครื่องคอมพิวเตอร์โดยวัดเวลาการทำงานของการประมวลผลของ Clock cycle
6. ข้อมูล QR Code ที่ผ่านการเข้ารหัสขนาด สกุล .bmp

7. รูปภาพลายน้ำดิจิตอลที่ใช้เป็นลายน้ำขนาด สกุล .bmp

8. รหัสลับ (Key) ที่ใช้แสดงความเป็นเจ้าของ คือ รหัสลับที่ใช้เพื่อเข้าสู่การแทรกและการถอดลายน้ำดิจิตอลที่ผ่านการแปลงโคซายน์แบบไม่ต่อเนื่อง การแปลงฟูริเยร์เต็มหน่วยและการแปลงเวฟเล็ตแบบไม่ต่อเนื่อง

9. การแทรกและถอดลายน้ำดิจิตอลแบบ DCT กระทำใน Frequency Domain โดยใช้ฟังก์ชันการแปลง DCT และ IDCT

10. การแทรกและถอดลายน้ำดิจิตอลแบบ DFT กระทำใน Frequency Domain โดยใช้ฟังก์ชันการแปลง FFT และ IFFT

11. การแทรกและถอดลายน้ำดิจิตอลแบบ DWT กระทำใน Frequency Domain โดยใช้ฟังก์ชันการแปลง DWT และ IDWT

12. ขั้นตอนการแทรกและถอดลายน้ำดิจิตอลบน QR Code ใช้โปรแกรม MATLAB Version 7.11.0(R2010b) ในการพัฒนา

1.4 ข้อตกลงเบื้องต้น

1. ภาพต้นฉบับ (Original Image) / ข้อมูล QR Code หมายถึง ภาพหลัก สกุล .bmp ใช้เพื่อการแทรกลายน้ำดิจิตอล มีขนาดตามที่ขอบเขตของการวิจัยกำหนด

2. ลายน้ำดิจิตอล (Digital Watermarking) หมายถึง ภาพลายน้ำที่ใช้เพื่อการแทรกลงไปในข้อมูล QR Code สกุล .bmp มีขนาดตามที่ขอบเขตของการวิจัยกำหนด

3. ภาพลายน้ำดิจิตอล (Watermarked Image) หมายถึง ข้อมูล QR Code ที่ผ่านกระบวนการแทรกลายน้ำดิจิตอล

4. วิธีการแปลงโคซายน์แบบไม่ต่อเนื่อง หมายถึง วิธีการทางคณิตศาสตร์ที่ใช้ในการแปลงภาพจากโดเมนพื้นที่ (Spatial Domain) ไปเป็นโดเมนความถี่ (Frequency Domain) และนำค่าที่ได้มาใช้ในการคำนวณต่อไป

5. วิธีการแปลงฟูเรียร์แบบไม่ต่อเนื่อง หมายถึง วิธีการทางคณิตศาสตร์ที่ใช้ในการแปลงภาพจากโดเมนพื้นที่ (Spatial Domain) ไปเป็นโดเมนความถี่ (Frequency Domain) และนำค่าที่ได้มาใช้ในการคำนวณต่อไป

6. วิธีการแปลงเวฟเล็ตแบบไม่ต่อเนื่อง หมายถึง วิธีการทางคณิตศาสตร์ที่ใช้ในการแปลงภาพจากโดเมนพื้นที่ (Spatial Domain) ไปเป็นโดเมนความถี่ (Frequency Domain) และนำค่าที่ได้มาใช้ในการคำนวณต่อไป

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ทำให้สามารถพัฒนาการซ่อนข้อมูลใน QR Code ได้
2. ทำให้เกิดแนวทางการวิจัยด้าน การพัฒนา QR Code ร่วมกับลายน้ำดิจิตอล ซึ่งกำลังเริ่มเป็นที่สนใจมากขึ้นในขณะนี้
3. ทำให้ได้แนวทางการวิจัยด้านการฝังลายน้ำบนโดเมนความถี่โดยใช้วิธีการแปลง 3 วิธี คือ Discrete Cosine Transform (DCT), Discrete Fourier Transform (DFT) และ Discrete Wavelet Transform (DWT) นั้นได้ผลแตกต่างกันอย่างไร

1.6 ลำดับขั้นตอนในการเสนอผลการวิจัย

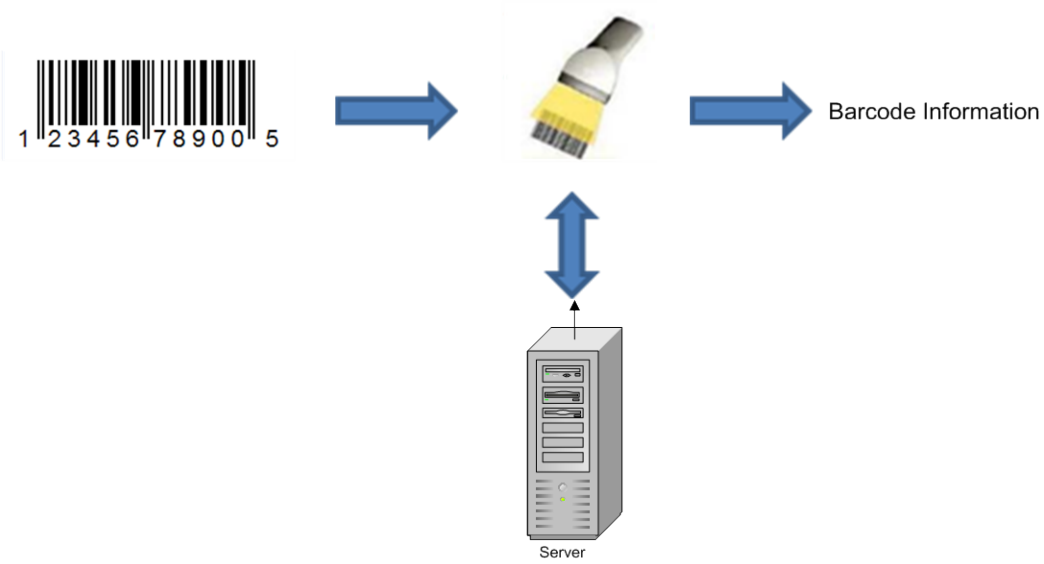
1. ศึกษาทฤษฎี หลักการพื้นฐานที่ใช้ในการวิจัยและงานวิจัยที่เกี่ยวข้อง
2. ศึกษาเทคนิคต่างๆที่มีอยู่ถึงแนวคิด หลักการ ข้อดี และข้อบกพร่องของแต่ละเทคนิค
3. ออกแบบและนำเสนอกระบวนการฝังและการถอดลายน้ำดิจิตอลร่วมกับ QR Code โดยใช้วิธีการแปลง 3 วิธี คือ Discrete Cosine Transform (DCT), Discrete Fourier Transform (DFT) และ Discrete Wavelet Transform (DWT)
4. ทดสอบวิธีการที่นำเสนอ
5. วิเคราะห์ผลการทดลอง
6. สรุปผลและเรียบเรียงวิทยานิพนธ์

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในส่วนนี้เป็นแนวคิดและทฤษฎีเกี่ยวกับโครงสร้างของ QR Code และการทำลายน้ำดิจิตอล ซึ่งจะการกล่าวถึงโครงสร้างของ QR Code ขั้นตอนการเข้ารหัส ขั้นตอนการถอดรหัส พื้นฐานการประมวลภาพดิจิตอล แนวคิดในการทำลายน้ำดิจิตอล ประวัติการทำลายน้ำดิจิตอล ลักษณะการแปลงโคซายน์แบบไม่ต่อเนื่อง (Discrete cosine transform, DCT), การแปลงฟูริเยร์เต็มหน่วย (Discrete fourier transform, DFT) และการแปลงเวฟเล็ต (Discrete wavelet transform, DWT) ตามลำดับ

2.1 โครงสร้างของ QR Code (Structure of QR Code)

บาร์โค้ดถูกนำไปใช้อย่างกว้างขวางในโฆษณาสื่อต่างๆ การบริการด้านไปรษณีย์ การจัดการเกี่ยวกับหนังสือ โกดังสินค้า การผลิตที่มี Process Control ในอุตสาหกรรมและอื่นๆ บาร์โค้ดมีประโยชน์หลายอย่าง เช่น High Input Speed, High Accuracy, Low Cost และ High Reliability ปัจจุบันบาร์โค้ดถูกประยุกต์ใช้ในเทคโนโลยีด้าน Automatic Recognition [14],[15] ด้วยการบรรจุข้อมูลลงไป บาร์โค้ดมีอยู่ทั่วไปในชีวิตประจำวันดังที่เห็นได้จากสินค้าต่างๆ ก็จะมีบาร์โค้ดเป็นของตัวเองในร้านสะดวกซื้อต่างๆและร้านซูเปอร์มาเก็ต ร้านขายหนังสือก็จะมีเลข ISBN (International Standard Book Number) ที่จะเห็นอยู่ด้านหลังหนังสือ โดยบาร์โค้ดพวกนี้จะมีลักษณะแถบขาว-ดำซึ่งเราจะเรียกว่าเป็น 1D Barcode (One Dimension Barcode) ดังภาพที่ 2-1 แต่บาร์โค้ดพวกนี้สามารถบรรจุได้แค่ตัวเลขเท่านั้น บาร์โค้ดชนิดนี้มีความจุไม่มากนักและถ้าเกิดต้องการข้อมูลจากบาร์โค้ดใช้การถอดรหัสกลับโดยใช้อุปกรณ์ที่ทำขึ้นเป็นพิเศษเพื่อแสกน แล้วได้ข้อมูลออกมา นอกจากนั้นเราต้องใช้ดาต้าเบสเพื่อเก็บข้อมูลของบาร์โค้ด เพื่อเก็บข้อมูลได้มากกว่าและสะดวกสบายกว่าเดิมจึงเกิด 2D Barcode (Two Dimension Barcode) ซึ่งไม่ต้องใช้อุปกรณ์ที่ทำขึ้นเป็นพิเศษเพื่อแสกนแล้วยังสามารถเข้ารหัสข้อมูลได้หลายชนิดมากมายดังที่กล่าวไปแล้วข้างต้น ดังภาพที่ 2-2



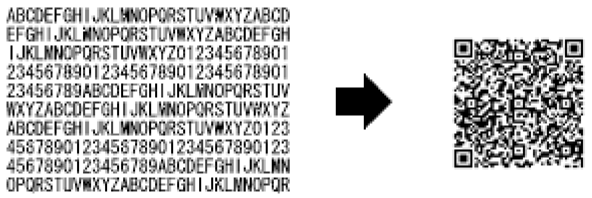
ภาพที่ 2-1 ขั้นตอนการถอดรหัสภาพบาร์โค้ดแบบ 1D Barcode

ภาพที่ 2-2 ขั้นตอนการถอดรหัสภาพบาร์โค้ดแบบ QR Code (2D Barcode)

จากภาพที่ 2-1 และ 2-2 จะเห็นว่า QR Code ซึ่งเป็น 2D Barcode นั้นผู้ใช้สามารถรับข้อมูลได้ง่ายกว่ามากเพียงแค่ถ่ายภาพ QR Code โดยผ่านแอพพลิเคชั่นที่สามารถอ่าน QR Code ได้ [15] ก็สามารถลิ้งไปหน้าเว็บอินเตอร์เน็ตได้ทันทีเมื่อ Content นั้นๆ เป็น URL (Uniform Resource Locators) ด้วยเหตุนี้เองทำให้เราพบ 2D Barcode ตามที่ต่างๆ มากมาย ในขณะที่บาร์โค้ดทั่วไปสามารถเก็บได้มากสุด 20 ดิจิตแต่ QR Code สามารถเก็บได้มากกว่าหลายเท่าทำให้ได้ข้อมูลที่มากกว่า QR Code สามารถจุข้อมูลได้ทุกรูปแบบ เช่น Numeric, Alphanumeric, Kanji, Kana, Hiragana, Symbols, Binary และ Control Code สูงถึง 7,089 อักขระต่อการเข้ารหัสเป็น 1 ภาพของ QR Code ดังตารางต่อไปนี้ [2]

ตารางที่ 2-1 แสดงความจุในแต่ละชนิดของอักขระ

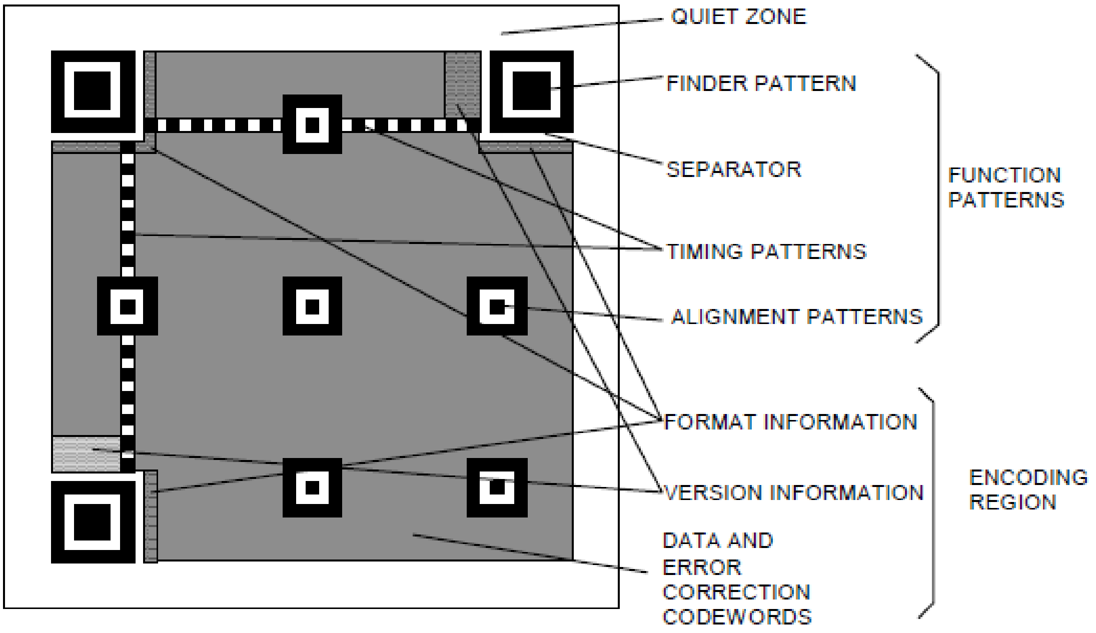
QR Code Data capacity	
Numeric only	Max. 7,089 characters
Alphanumeric	Max. 4,296 characters
Binary (8 bits)	Max. 2,953 bytes
Kanji, full-width Kana	Max. 1,817 characters



ภาพที่ 2-3 แสดงให้เห็นข้อความมากมายกลายเป็นแค่ 1 สัญลักษณ์

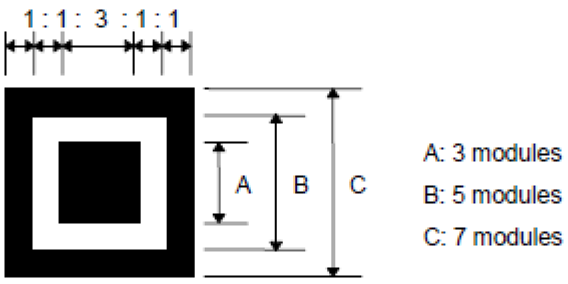
การแสดงข้อมูลของ QR Code โมดูลสีดำเป็นเลขไบนารี 1 และโมดูลสีขาวเป็นเลขไบนารี 0 ขนาดของ QR Code ถูกกำหนดเป็น Version ซึ่ง Version มีตั้งแต่ 1 ถึง 40 โดย Version 1 คือ เมททริกซ์ขนาด 21*21 modules และเมื่อ version เพิ่มขึ้นที่ละหนึ่งจะเป็นการเพิ่มขึ้นขนาดด้านละ 4 modules ในแต่ละด้าน ดังนั้น version 40 คือ เมตทริกซ์ขนาด 177*177 modules (ภาคผนวก ข) ซึ่งเกิดจาก {(39*4)+21}*{(39*4)+21} นอกจากนั้นยังมีคุณสมบัติการหมุน คือ เมื่อทำการหมุนรูป QR Code ก็ยังสามารถถอดรหัสได้

QR Code ถูกสร้างจากโมดูลสี่เหลี่ยมซึ่งประกอบด้วย Encoding Region, Function Patterns, Finder Patterns, Separator, Timing Patterns และ Alignment Patterns ดังภาพซึ่งในเป็นเวอร์ชัน 7 ต่อไปนี้ [1]



ภาพที่ 2-4 แสดงโครงสร้างของ QR Code

2.1.1 Finder Patterns คือ QR Code มีการระบุ Finder Patterns 3 ตำแหน่ง ได้แก่ ตำแหน่งมุมบนซ้าย, มุมบนขวาและมุมล่างซ้าย ซึ่งแต่ละ Finder Patterns มีลักษณะเป็นสี่เหลี่ยมที่ซ้อนทับตรงกลางกันและถูกสร้างจากโมดูลสีดำขนาด 7 x 7 หน่วย โมดูลสีขาวขนาด 5 x 5 หน่วย และโมดูลสีดำขนาด 3 x 3 หน่วยทำให้ได้อัตราส่วนความกว้างของแต่ละ Finder Patterns เป็น 1:1:3:1:1 ดังภาพที่ 2-5 [1]



ภาพที่ 2-5 อัตราส่วนโมดูลดำ-ขาวของ Finder Patterns

2.1.2 Separator คือ ถูกสร้างด้วยโมดูลขาวทั้งหมด ซึ่งถูกวางระหว่างแต่ละ Finder Patterns กับส่วนที่เข้ารหัส ดังภาพที่ 2-4

2.1.3 Timing Pattern ใน QR Code มีทั้งแนวนอนและแนวตั้งที่ประกอบด้วยโมดูลดำ-ขาวสลับกัน โดยเป็นโมดูลดำในช่วงต้นและท้าย ตำแหน่งของ Timing Pattern จะอยู่แถวที่ 6 ของ QR Code ซึ่งอยู่ระหว่าง Separator ดังภาพที่ 2-4

2.1.4 Alignment Patterns ถูกแสดงในเวอร์ชัน 2 ขึ้นไป ซึ่งแต่ละ Alignment Patterns เป็นสี่เหลี่ยม 3 รูปที่ซ้อนทับศูนย์กลางกันโดยถูกสร้างจากโมดูลดำขนาด 5 x 5 หน่วย โมดูลขาวขนาด 3 x 3 หน่วยและโมดูลดำตรงกลางขนาด 1 หน่วย โดยจำนวนของ Alignment Patterns จะขึ้นกับเวอร์ชัน

2.1.5 Encoding Region เป็นการแสดงถึง Error Correcting Codewords, Format Information และ Version Information

2.1.6 Quiet Zone เป็นส่วนที่เป็นอิสระ โดยจะอยู่รอบๆ QR Code ทั้งสี่ด้าน

2.2 ขั้นตอนการเข้ารหัส (Encode Procedure Overview)

ในส่วนนี้เป็นภาพรวมของขั้นตอนที่จำเป็นในการแปลง Input Data เป็น QR Code โดยมีขั้นตอนดังต่อไปนี้

2.2.1 วิเคราะห์ข้อมูล (Data analysis) คือ วิเคราะห์กระแสของ Input Data เพื่อระบุความแตกต่างของ Characters ที่จะเข้ารหัส โดย QR Code มี Extended Channel Interpretation ทำให้ Data ที่มี Character ต่างกันสามารถทำการเข้ารหัสได้ ซึ่งให้คำจำกัดความไว้ใน AIM Inc. International Technical Specification Extended Channel Interpretations จะทำการเลือกเวอร์ชันที่น้อยสุดที่สามารถบรรจุข้อมูลได้ รายชื่อของเวอร์ชันและความจุ โดยที่ทุก Codewords ยาว 8 บิต

โดยในงานวิจัยชิ้นนี้ สนใจการเข้ารหัส 2 Modes คือ Numeric Mode และ Alphanumeric Mode ซึ่งมีรายละเอียด ดังต่อไปนี้ [1]

2.2.1.1 Numeric Mode คือ มีการเข้ารหัสจากกลุ่มตัวเลข (0-9) โดยปกติ 3 Data Characters ถูกแสดงเป็น 10 บิต

2.2.1.2 Alphanumeric Mode คือ มีการเข้ารหัสจากกลุ่มของ 45 Characters เช่น ตัวเลข 10 ตัว (0-9), ตัวอักษร 26 ตัว (A-Z) และ สัญลักษณ์ 9 ตัว (SP, $, %, *, +, -, ., /, :) โดยปกติ 2 Input Characters ถูกแสดงเป็น 11 บิต

2.2.2 การเข้ารหัสข้อมูล (Data encoding) คือ แปลงข้อมูลที่เป็น Characters เป็นกระแสบิตที่สอดคล้องกับโหมดที่บังคับ โดยแทรก Mode Indicators เพื่อเปลี่ยนโหมด และ Terminator เป็นท้ายสุดของ Data หลังจากนั้นทำการแบ่งเป็นกระแสของบิตเป็น Codewords โดยแต่ละ Codewords มี 8 บิต แล้วทำการเติม Codewords ให้ครบตามแต่ละเวอร์ชันที่กำหนด โดยเริ่มแรก Input Data จะเริ่มเข้าสู่ Mode Indicators เพื่อระบุว่าเป็นกระแสของมูลแบบใด ดังตารางต่อไปนี้ [1]

ตารางที่ 2-2 Mode Indicators ของ QR Code

Mode	Representation
Numeric	0001
Alphanumeric	0010
Terminator (สิ้นสุดข้อความ)	0000

ตัวอย่างเช่น ถ้าประเภทของอักขระอยู่ในรูปแบบ Alphanumeric เราจะเลือกโหมด Alphanumeric คือ 0010 ต่อไปเป็นการแสดงจำนวนบิตในการนับ Character ของ QR Code ดังตารางต่อไปนี้

ตารางที่ 2-3 จำนวนบิต Character Count Indicator ของ QR Code

Version	Numeric Mode	Alphanumeric Mode
1 ถึง 9	10	9
10 ถึง 26	12	11
27 ถึง 40	14	13

ตัวอย่างเช่น จากตัวอย่างที่แล้วมี 8 ตัว เราจะเข้ารหัสไบนารียาว 9 บิตสำหรับประเภท Alphanumeric คือ 0010 000001000

2.2.2.1 การเข้ารหัส Numeric Mode คือ Input Data มีค่าตามตัวเลขที่เป็น Input Data ถูกแบ่งเป็นกลุ่มกลุ่มละ 3 ดิจิต เช่น ข้อมูล “123456” จะถูกแยกชุดข้อมูลเป็น “123” และ “456” โดยที่แต่ละกลุ่มถูกแปลงเป็นเลขไบนารี 10 บิตอย่างเท่าๆ กัน แต่ในกรณีที่แยกชุดข้อมูลแล้วไม่ครบ 3 ตัว เราจะใช้รหัสไบนารีที่มีความยาว 4 บิต และ 7 บิต แทนชุดข้อมูลขนาด 1 และ 2 ตัว ตามลำดับ ยกตัวอย่างเช่น ข้อมูล “9876” จะถูกแยกชุดข้อมูลออกเป็น “987” ซึ่งมีความยาวไบนารี10 บิต และ “6” ซึ่งมีความยาวไบนารี 4 บิตเป็นผลลัพธ์คือ “1111011011 0110” ดังตารางที่ 2-4 และการจำนวนบิตแสดงไว้ในตางรางที่ 2-5 สำหรับความยาวของ Data Characters ของกระแสข้อมูลใน Numeric Mode ถูกเขียนเป็นสูตรดังต่อไปนี้

$$B = M + C + 10(D \text{ DIV } 3) + R \qquad (2\text{-}1)$$

โดยที่ B เป็นจำนวนบิตของกระแสข้อมูล ส่วน M เป็นจำนวนบิตของ Mode Indicator (ดังตารางที่ 2-4) ส่วน C เป็นจำนวนบิตใน Character Count Indicator ของ QR Code (ดังตารางที่ 2-5) ส่วน D เป็นจำนวน Input Data Characters โดยเลือกค่า R ตามเงื่อนไขสมการต่อไปนี้

$$R = 0 \text{ if } (D \text{ MOD } 3) = 0 \qquad (2\text{-}2)$$

$$R = 4 \text{ if } (D \text{ MOD } 3) = 1 \qquad (2\text{-}3)$$

$$R = 7 \text{ if } (D \text{ MOD } 3) = 2 \qquad (2\text{-}4)$$

2.2.2.2 การเข้ารหัส Alphanumeric Mode ในแต่ละ Input Data Character ถูกกำหนดค่าตั้งแต่ 0 ถึง 44 ดังตารางที่ 2-6

ตารางที่ 2-4 แสดงค่า Encoding และ Decoding ของ Alphanumeric Mode

Characters	Value
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
A	10
B	11
C	12
D	13
E	14
F	15
G	16
H	17
I	18
J	19
K	20
L	21
M	22
N	23
O	24
P	25
Q	26

R	27
S	28
T	29
U	30
V	31
W	32
X	33
Y	34
Z	35
SP	36
$	37
%	38
*	39
+	40
-	41
.	42
/	43
:	44

ใน Input Data Characters ถูกแบ่งเป็นกลุ่มเป็นกลุ่มละ 2 Characters ซึ่งจะถูกเข้ารหัสเป็น 11 บิต โดยค่าของ Characters ตัวแรกจะถูกคูณด้วย 45 แล้วบวกด้วยค่าของ Characters ตัวที่สอง หลักจากนั้นแปลงเป็นเลขไบนารียาว 11 บิต แต่ในกรณีที่แยกชุดข้อมูลแล้วไม่ครบ 2 เราจะใช้ความยาวของรหัส Binary ชนิด 6 บิตแทนชุดข้อมูลขนาด 1 ตัว จากตัวอย่างชุดข้อมูล 0010 000001000 (จากตัวอย่างที่ผ่านมา) ซึ่งจะมีตัวอย่างดังต่อไปนี้

“AB”	“CD”
45*10+11	45*12+13
461	553
00111001101	01000101001

"E1"	"23"
45*14+1	45*2+3
631	93
01001110111	00001011101

สำหรับความยาวของ Data Characters ของกระแสข้อมูลใน Alphanumeric Mode โดยมีการคำนวณดังสมการต่อไปนี้

$$B = M + C + 11(D \text{ DIV } 2) + 6(D \text{ MOD } 2) \quad (2\text{-}5)$$

โดยที่ B เป็นจำนวนบิตของกระแสข้อมูล ส่วน M เป็นจำนวนบิตของ Mode Indicator (ดังตารางที่ 2-4) ส่วน C เป็นจำนวนบิตใน Character Count Indicator ของ QR Code (ดังตารางที่ 2-5) ส่วน D เป็นจำนวน Input Data Characters

2.2.2.3 Terminator คือ เมื่อทำการเข้ารหัสเสร็จ หลังจากนั้นในส่วนของ Terminator เป็นการบ่งบอกถึงการสิ้นสุดของ Data โดยมีลักษณะเป็นบิต 0 เรียงกันซึ่งจะรวมเข้ากับกระแสของข้อมูลที่ผ่านการแปลงเข้าสู่ไบนารีดังที่กล่าวถึงก่อนหน้าแต่ Terminator จะถูกละเว้นเมื่อกระแสของข้อมูลถูกใส่จนเต็มกระแสของข้อมูลจะสอดคล้องกับแต่ละ Mode ซึ่งจะถูกบังคับให้เชื่อมโยงกัน โดยที่ผลรวมของข้อมูลจะถูกแบ่งเป็น Codewords โดยที่แต่ละ Codewords ทั้งหมดจะมีความยาว 8 บิต แต่ถ้าความยาวของกระแสของข้อมูลใน Codewords สุดท้ายไม่ครบ 8 บิตให้เติมบิต 0 ต่อบิตของ Data ตัวสุดท้าย (Least Significant Bit) โดยเพิ่ม 0000 ต่อท้ายเข้าไป แต่ในกรณีที่ความยาวของบิตเต็มจำนวนแล้วก็ไม่จำเป็นต้องใช้ 0000 เพื่อชี้แจงว่าจุดสิ้นสุด ยกตัวอย่างเช่น 0010 000001000 00111001101 01000101001 01001110111 00001011101 <u>0000</u> นอกจากนั้น QR Code ใช้ Reed-Solomon เพื่อควบคุม Error ทำให้สามารถกู้คืนข้อมูลได้ถูกต้องแม้ภาพ QR Code เกิดการเปรอะเปื้อนหรือฉีกขาด ซึ่งสามารถกู้คืนได้ตามระดับตามตารางที่ 2-6

ตารางที่ 2-5 แสดงระดับของ Error Correcting

Error Correcting Level	Recovery Capacity % (approx.)
L	7
M	15
Q	25
H	30

2.2.2.4 Pad Codewords คือ เติม Codewords 11101100 และ 00010001 สลับกันเรื่อยๆ จนครบจำนวน Codewords โดยพิจารณา Number of Data Codewords (ภาคผนวก ค) จากนั้นจะเข้าสู่การเข้ารหัสเป็น Codewords สำหรับข้อมูลผลลัพธ์จากตัวอย่างที่ผ่านมาจะถูกแบ่งออกให้เป็นชุดๆ ชุดละ 8 บิต คือ 00100000 01000001 11001101 01000101 00101001 11011100 00101110 10000
ในกรณีที่ข้อมูลชุดสุดท้ายไม่ครบ 8 ตัว ให้เพิ่ม 0 เข้าไปจนครบ ดังนี้ 00100000 01000001 11001101 01000101 00101001 11011100 00101110 10000000 ถ้าจำนวน Codewords น้อยกว่าความจุของจะเพิ่ม “11101100” และ “00010001” เข้าไปจนกว่าจะเต็มความจุ ดังตัวอย่างความจุของ QR Code เวอร์ชัน 1 ที่มี Error Correcting Level ระดับ H มีค่าเป็น 9 Codewords ดังนี้ 00100000 01000001 11001101 01000101 00101001 11011100 00101110 10000000 11101100 สรุปผลลัพธ์ได้เป็นจำนวนเต็ม 9 ตัว ดังนี้ 32 65 205 69 41 220 46 128 236

2.2.3 พิจารณา Error Correcting Codewords (Error correction coding) [1] คือ เราใช้ Reed-Solomon Error Correcting สำหรับ QR Code เพื่อช่วยในการวิเคราะห์หา Error Correcting Codewords โดยเริ่มจากนำข้อมูลผลลัพธ์จากในหัวข้อที่ผ่านมาเพื่อลดทอนให้เป็นข้อมูล RS block ตามมาตรฐานที่มีอยู่ เนื่องจากค่า RS Block ของ 1-H เท่ากับ 1 ดังนั้นเราไม่ต้องลดทอนข้อมูลจากตัวอย่างของเราพิจารณาค่า RS Block ของ QR Code พิจารณา Number of Error Correction Codewords (ภาคผนวก ง) โดยที่พหุนามทางคณิตศาสตร์ของ QR Code จะถูกคำนวณทางคณิตศาสตร์โดยใช้ Bit-Wise Modulo 2 และ Byte-wise modulo 2 ของ

100011101 ซึ่งพวกนี้อยู่ในเรื่อง Galois Field ของ 2^8 ที่มี 100011101 จะถูกเขียนเป็น Field ของ Prime Modulus Polynomial ได้ดังนี้ $x^8 + x^4 + x^3 + x^2 + 1$ ซึ่งมี Data Codewords เป็นสัมประสิทธิ์ของเงื่อนไขพหุนามที่มีสัมประสิทธิ์ของเทอมที่สูงที่สุดจะเป็น Data Codeword อันแรก ในทางกลับกันสัมประสิทธิ์ของเงื่อนไขพหุนามที่มีสัมประสิทธิ์ของเทอมที่ต่ำที่สุดจะเป็น Data Codeword อันสุดท้ายก่อนที่จะเป็น Error Correction Codeword ตัวแรก ในส่วนของ Error Correction Codewords จะเป็นส่วนที่เหลือหลังจากแบ่ง Data Codewords ด้วยพหุนาม g(x) ซึ่งใช้กับ Error Correction Codes โดยที่สัมประสิทธิ์ที่มีกำลังสูงสุดของส่วนที่เหลือ คือ Error Correction Codeword ตัวแรกและสัมประสิทธิ์ที่มีกำลังเป็น 0 คือ Error Correction Codeword ตัวสุดท้ายซึ่งเป็น Codeword ตัวสุดท้ายใน Block โดยที่มีพหุนามที่แตกต่างกันได้ 36 แบบ (ภาคผนวก จ) เพื่อใช้สร้าง Error Correction Codewords เช่น จากตัวอย่างใน 2.2.2.4 ที่ QR Code มีเวอร์ชัน 1 ที่มี Error Correcting Level ระดับ H จะมีจำนวน Error Correction Codewords (ภาคผนวก ง) ในช่อง Numbers of Error Correction Codewords มีค่าเท่ากับ 17 ส่งผลให้ (ภาคผนวก จ) เรากำหนดพหุนามของ Error Correction Codewords โดยเลือก Generator Polynomial จากช่องที่มี Numbers of Error Correction Codewords มีค่าเท่ากับ 17 หลังจากนั้นทำการคำนวณ Reed Solomon โดยใช้ตัวอย่างจาก 2.2.2.4 ที่มี Numbers of Error Correction Codewords มีค่าเท่ากับ 17 ดังนี้

$$g(x) = x^{17} + \alpha^{43}x^{16} + \alpha^{139}x^{15} + \alpha^{206}x^{14} + \alpha^{78}x^{13} + \alpha^{43}x^{12} + \alpha^{239}x^{11} + \alpha^{123}x^{10} + \alpha^{206}x^{9} + \alpha^{214}x^{8} + \alpha^{147}x^{7} + \alpha^{24}x^{6} + \alpha^{99}x^{5} + \alpha^{150}x^{4} + \alpha^{39}x^{3} + \alpha^{243}x^{2} + \alpha^{163}x + \alpha^{136} \tag{2-6}$$

สามารถเขียนสัมประสิทธิ์แบบจำนวนเต็ม (ภาคผนวก ฉ) ดังนี้

$$g(x)' = x^{17} + 119x^{16} + 66x^{15} + 83x^{14} + 120x^{13} + 119x^{12} + 22x^{11} + 197x^{10} + 83x^{9} + 249x^{8} + 41x^{7} + 143x^{6} + 134x^{5} + 85x^{4} + 53x^{3} + 125x^{2} + 99x + 79 \tag{2-7}$$

และเรามี Data Code Words ที่มีค่าในเลขฐานสิบ ดังนี้

$$32\ 65\ 205\ 69\ 41\ 220\ 46\ 128\ 236 \quad (2\text{-}8)$$

จากนั้นคำนวณหาสัมประสิทธิ์ของ f(x) โดย เริ่มพิจารณา g(x) ให้มีสัมประสิทธิ์เท่ากับ Data Code Words โดยกำหนดให้ x มีกำลังมากที่สุดเท่ากับ 25 ดังนี้

$$\text{Coefficients calculation } f(x) = g(x)*(\alpha^{5})*x^{8} \quad (2\text{-}9)$$

$$f(x) = \alpha^{5}x^{25} + \alpha^{191}x^{24} + \alpha^{12}x^{23} + \alpha^{221}x^{22} + \alpha^{147}x^{21} + \alpha^{187}x^{20} + \alpha^{130}x^{19} + \alpha^{7}x^{18} + \alpha^{122}x^{17} \quad (2\text{-}10)$$

พบว่า f(x) มีสัมประสิทธิ์ (ภาคผนวก ฉ)

$$f(x)' = 32x^{25} + 65x^{24} + 205x^{23} + 69x^{22} + 41x^{21} + 220x^{20} + 46x^{19} + 128x^{18} + 236x^{17} + 0x^{16} + 0x^{15} + 0x^{14} + 0x^{13} + 0x^{12} + 0x^{11} + 0x^{10} + 0x^{9} + 0x^{8} + 0x^{7} + 0x^{6} + 0x^{5} + 0x^{4} + 0x^{3} + 0x^{2} + 0x^{1} + 0x^{0} \quad (2\text{-}11)$$

จากนั้นพิจารณาสมการ 2-6 และ 2-9 เปรียบเทียบสัมประสิทธิ์กัน ดังนี้

1. α ตำแหน่งที่มีเลขยกกำลังของค่า x มากที่สุดทั้ง 2-6 และ 2-8 คือ 0 กับ 5 ทำการรวมกันได้ 5 ส่งผลให้ได้ α^{5} นำค่า α^{5} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 32 นำค่า 32 มา XOR กับค่า 32 ในสมการ 2-10 พบว่าได้ 0
2. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 43 กับ 5 ทำการรวมกันได้ 48 ส่งผลให้ได้ α^{48} นำค่า α^{48} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 70 นำค่า 70 มา XOR กับค่า 65 ในสมการ 2-10 พบว่าได้ 7

3. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 139 กับ 5 ทำการรวมกันได้ 144 ส่งผลให้ได้ α^{144} นำค่า α^{144} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 168 นำค่า 168 มา XOR กับค่า 205 ในสมการ 2-10 พบว่าได้ 101
4. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 206 กับ 5 ทำการรวมกันได้ 211 ส่งผลให้ได้ α^{211} นำค่า α^{211} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 178 นำค่า 178 มา XOR กับค่า 69 ในสมการ 2-10 พบว่าได้ 247
5. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 78 กับ 5 ทำการรวมกันได้ 83 ส่งผลให้ได้ α^{83} นำค่า α^{83} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 187 นำค่า 187 มา XOR กับค่า 41 ในสมการ 2-10 พบว่าได้ 146
6. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 43 กับ 5 ทำการรวมกันได้ 48 ส่งผลให้ได้ α^{48} นำค่า α^{48} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 70 นำค่า 70 มา XOR กับค่า 220 ในสมการ 2-10 พบว่าได้ 154
7. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 239 กับ 5 ทำการรวมกันได้ 244 ส่งผลให้ได้ α^{244} นำค่า α^{244} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 250 นำค่า 250 มา XOR กับค่า 46 ในสมการ 2-10 พบว่าได้ 212
8. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 123 กับ 5 ทำการรวมกันได้ 128 ส่งผลให้ได้ α^{128} นำค่า α^{128} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 133 นำค่า 133 มา XOR กับค่า 128 ในสมการ 2-10 พบว่าได้ 5
9. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 206 กับ 5 ทำการรวมกันได้ 211 ส่งผลให้ได้ α^{211} นำค่า α^{211} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 178 นำค่า 178 มา XOR กับค่า 236 ในสมการ 2-10 พบว่าได้ 94
10. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 214 กับ 5 ทำการรวมกันได้ 219 ส่งผลให้ได้ α^{219} นำค่า α^{219} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 86 นำค่า 86 มา XOR กับค่า 0 ในสมการ 2-10 พบว่าได้ 86
11. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 147 กับ 5 ทำการรวมกันได้ 152 ส่งผลให้ได้ α^{152} นำค่า α^{152} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 73 นำค่า 73 มา XOR กับค่า 0 ในสมการ 2-10 พบว่าได้ 73

12. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 24 กับ 5 ทำการรวมกันได้ 29 ส่งผลให้ได้ α^{29} นำค่า α^{29} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 48 นำค่า 48 มา XOR กับค่า 0 ในสมการ 2-10 พบว่าได้ 48
13. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 99 กับ 5 ทำการรวมกันได้ 104 ส่งผลให้ได้ α^{104} นำค่า α^{104} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 13 นำค่า 13 มา XOR กับค่า 0 ในสมการ 2-10 พบว่าได้ 13
14. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 150 กับ 5 ทำการรวมกันได้ 155 ส่งผลให้ได้ α^{155} นำค่า α^{155} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 114 นำค่า 114 มา XOR กับค่า 0 ในสมการ 2-10 พบว่าได้ 114
15. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 39 กับ 5 ทำการรวมกันได้ 44 ส่งผลให้ได้ α^{44} นำค่า α^{44} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 238 นำค่า 238 มา XOR กับค่า 0 ในสมการ 2-10 พบว่าได้ 238
16. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 243 กับ 5 ทำการรวมกันได้ 248 ส่งผลให้ได้ α^{248} นำค่า α^{248} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 27 นำค่า 27 มา XOR กับค่า 0 ในสมการ 2-10 พบว่าได้ 27
17. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 163 กับ 5 ทำการรวมกันได้ 168 ส่งผลให้ได้ α^{168} นำค่า α^{168} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 252 นำค่า 252 มา XOR กับค่า 0 ในสมการ 2-10 พบว่าได้ 252
18. α ตำแหน่งที่มีเลขยกกำลังของค่า x รองลงมาในสมการ 2-6 และ 2-8 คือ 136 กับ 5 ทำการรวมกันได้ 141 ส่งผลให้ได้ α^{141} นำค่า α^{141} เทียบสัมประสิทธิ์ (ภาคผนวก ฉ) พบว่าได้ 21 นำค่า 21 มา XOR กับค่า 0 ในสมการ 2-10 พบว่าได้ 21

พบว่าได้ Coefficients calculation f(x) ที่ผ่าน $g(x)*(\alpha^{5})*x^{8}$ คือ 7, 101, 247, 146, 154, 212, 5, 94, 86, 73, 48, 13, 114, 238, 27, 252 และ 21 ซึ่งนำไปสร้างเป็นสร้าง $f(x) = g(x)*(\alpha^{198})*x^{7}$ ได้ดังนี้ $7x^{24} + 101x^{23} + 247x^{22} + 146x^{21} + 154x^{20} + 212x^{19} + 5x^{18} + 94x^{17} + 86x^{16} + 73x^{15} + 48x^{14} + 13x^{13} + 114x^{12} + 238x^{11} + 27x^{10} + 252x^{9} +21x^{8}+ 0x^{7} + 0x^{6} + 0x^{5} + 0x^{4} + 0x^{3} + 0x^{2} + 0x^{1} + 0x^{0}$ แล้วทำตามขั้นตอนที่ผ่านมานกระทั้ง $g(x)*(\alpha^{92})*x^{0}$ ที่เลขยกกำลังของ x เป็น 0 เมื่อคำนวณเสร็จได้ค่า Error correcting code words in decimal notation เป็น 42. 159. 74. 221.

244. 169. 239. 150. 138. 70. 237. 85. 224. 96. 74. 219. 61. จำนวนของ Codewords จะเท่ากับจำนวนของความจุ Codewords ที่แสดงให้เห็นในภาพที่ 2-8 กับ 2-9 ซึ่งขั้นตอน

2.2.4 Finally codewords (Structure final message) คือ ต่อไปนี้เป็นการสร้างลำดับสุดท้ายของ Codeword [1] ซึ่งมีขั้นตอนดังต่อไปนี้

2.2.4.1 แบ่ง Data Codewords เป็น n Blocks ที่บอกไว้ (ภาคผนวก ง) ตามเวอร์ชันและ Error Correcting Level

2.2.4.2 ในแต่ละ Data Block คำนวณ Block ที่เกี่ยวข้องกับ Error Correction Codewords (ภาคผนวก ง)

2.2.4.3 รวบรวม Data และ Error Correction Codewords จากแต่ละ Block เช่น ถ้ามี 4 Blocks จะเป็นดังนี้ Data Block 1, Codeword 1; Data Block 2, Codeword 1; ... Data Block 4, Codeword 1; Data block 1, Codeword 2; ... เหมือนกันกับ Data Block 3, Codeword สุดท้าย; … Data Block 4, Codeword สุดท้าย หลังจากนั้นจึงเป็น Error Correction Block 1, Codeword 1; Error Correction Block 2, Codeword 1; … เหมือนกันกับ Error Correction Block 4, Codeword สุดท้าย ซึ่งแสดงให้เห็นดังตารางที่ 2-11 ซึ่งต่อไปจะยกตัวอย่าง QR Code เวอร์ชัน 5 ที่มี Error Correcting Level ระดับ H ที่ประกอบด้วย 4 Data Blocks และ 4 Error Correction Blocks แบ่งเป็น 2 Blocks แรกบรรจุ 11 Data Codewords และ 22 Error Correction Codewords อีก 2 Blocks ถัดมาบรรจุ 12 Data Codewords และ 22 Error Correction Codewords ซึ่งมาจากตารางที่ 2-9 โดยแต่ละ Block ของ Data Codewords เขียนด้วย D_n และ Error Correction Codewords เขียนด้วย E_n การจัดเรียงจะเป็นดังภาพที่ 2-7

	Data Codewords					Error Correction Codewords			
Block 1	D_1	D_2	.....	D_{11}		E_1	E_2	.....	E_{22}
Block 2	D_{12}	D_{13}	.....	D_{22}		E_{23}	E_{24}	.....	E_{44}
Block 4	D_{23}	D_{24}	.....	D_{33}	D_{34}	E_{45}	E_{46}	.....	E_{66}
Block 5	D_{35}	D_{36}	.....	D_{45}	D_{46}	E_{57}	E_{68}	.....	E_{88}

ภาพที่ 2-6 รูปแบบการวาง Codewords จากตัวอย่าง

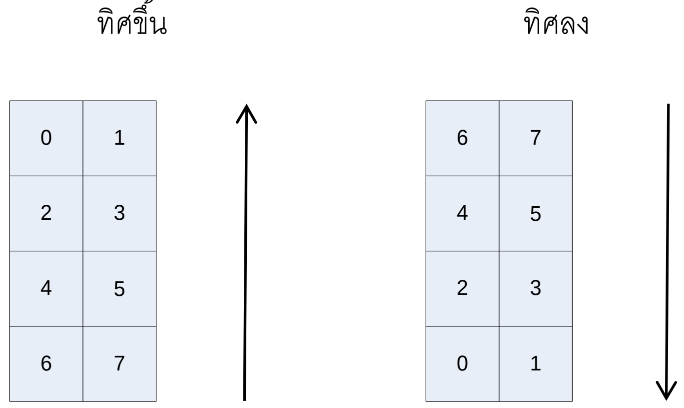
เพราะฉะนั้น Codewords สุดท้ายเรียงกันของเวอร์ชัน 5 ที่มี Error Correcting Level ระดับ H จะมีลักษณะดังนี้ D_1, D_{12}, D_{23}, D_{35}, D_2, D_{13}, D_{24}, D_{36}, ... D_{11}, D_{22}, D_{33}, D_{45}, D_{34}, D_{46}, E_1, E_{23}, E_{45}, E_{67}, E_2, E_{24}, E46, E68, ... E22, E44, E_{66}, E_{88}

2.2.5 การวาง Codewords (Module placement in matrix) คือ มี 2 ลักษณะด้วยกัน คือ แบบปกติจัดเรียงตามแนวตั้งและแบบไม่ปกติถูกใช้เมื่อมีการเปลี่ยนทิศทาง ดังที่แสดงในภาพที่ 2-7, 2-8 และ 2-9 โดยจำนวน Modules ในแนวตั้งและแนวนอนขึ้นอยู่กับเวอร์ชันที่ใช้ ซึ่งตำแหน่งการวางขึ้นอยู่กับ Finder Pattern, Separator, Timing Pattern และ Alignment Patterns โดยตำแหน่ง Module ของ Format Information และ Version Information จะถูกวางในตำแหน่งที่ว่างซึ่งตำแหน่งว่างนี้จะแสดงให้เห็นในภาพที่ 2-10 และ 2-11 ซึ่งเหมือนกันทุกเวอร์ชัน (ยกเว้นเวอร์ชัน 1 ถึง 6) โดยปกติจะเริ่มวางข้อมูลที่ถูกเข้ารหัสที่ตำแหน่งมุมล่างขวาและวางต่อทิศทางขึ้นและลงสลับกันจากขวาไปซ้าย การควบคุมการวางของบิตถูกแสดงให้เห็นในภาพที่ 2-10 และ 2-11 ซึ่งแสดงเวอร์ชัน 2 และ 7 โดยใช้หลักการดังนี้ [1]

2.2.5.1 ลำดับการวางของบิตในคอลัมน์จะวางจากขวาไปซ้ายและถ้าไม่วางในทิศทางขึ้นก็จะเป็นทิศทางลงตามทิศทางการวาง

2.2.5.2 Significant Bit (แสดงเป็น 7 บิต) ของแต่ละ Codeword จะวางที่ตำแหน่ง Module แรกที่ว่าง จากนั้นก็วางในตำแหน่ง Module ถัดมา ใน Module ที่มี Significant Bit มากที่สุดจะถูกวางตำแหน่งขวาล่างในทิศทางการวางทิศทางขึ้น

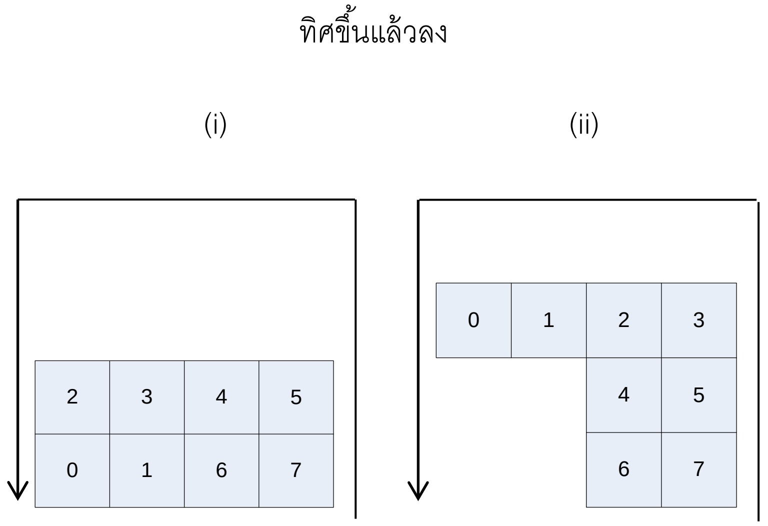
และ Significant Bit มากที่สุดจะถูกวางตำแหน่งขวาบนในทิศทางการวางทิศทางลง มีความเป็นได้ที่ Significant Bit มากที่สุดจะวางที่ตำแหน่ง Module ซ้ายล่าง ถ้าก่อนหน้าสิ้นสุดด้วยคอลัมน์ Module ขวา ซึ่งสามารถดูได้ในภาพที่ 2-9



ภาพที่ 2-7 การวางบิตในรูปแบบปกติในทิศทางขึ้นและลง

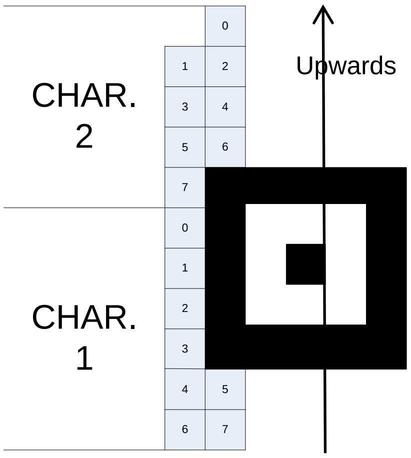
2.2.5.3 เมื่อพบกับขอบเขตแนวนอนของ Alignment Pattern หรือ Timing Pattern การวางยังคงดำเนินการต่อโดยอาจวางอยู่เหนือหรือใต้ Pattern โดยให้ขอบเขตการเข้ารหัสต่อกัน

2.2.5.4 เมื่อสุดขอบบนหรือล่าง (เช่น ขอบของ QR Code หรือ Format Information หรือ Version Information หรือ Separator) บิตที่เหลือใน Codeword จะต้องวางในคอลัมน์ถัดไปทางซ้าย และทิศทางการวางจะตรงกันข้ามกับตอนก่อนสุดขอบบนหรือล่าง ดังภาพที่ 2-8



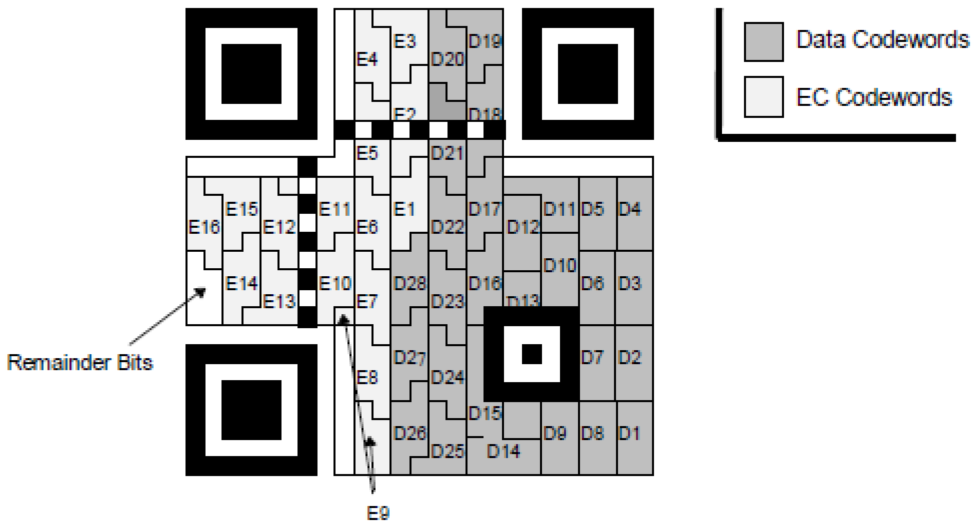
ภาพที่ 2-8 ทิศทางการวางบิต (i) เป็นแบบปกติและ (ii) แบบไม่ปกติมีการเปลี่ยนทิศการวาง

2.2.5.5 เมื่อคอลัมน์ Module ขวามือชนกับ Alignment Pattern หรือพื้นที่ที่ถูกครอบครองด้วย Version Information บิตจะวางในรูปแบบไม่ปกติโดยการขยายคอลัมน์ Module หนึ่งหลักที่ติดกับ Alignment Pattern หรือ Version Information ถ้าจบก่อนที่จะวางในสองคอลัมน์ถัดไป Significant Bit ที่สำคัญที่สุดของการวางครั้งถัดไปนั้นจะถูกวางในหนึ่งคอลัมน์ ดังภาพที่ 2-9



ภาพที่ 2-9 การวางพื้นที่ที่จะวางถูกครอบครองโดยข้อมูลอื่น

เมื่อความจุของ Data มีไม่ครบ 8 บิต จำนวนบิตที่เหลือที่เหมาะสม (3 หรือ 4 หรือ 7) ซึ่งบิตที่เหลือจะมีค่าเป็น 0 ทั้งหมด ซึ่งตำแหน่งของบิตที่เหลือจะวางที่ตำแหน่งดังภาพที่ 2-10



ภาพที่ 2-10 ตัวอย่างการจัดเตรียมการวางในเวอร์ชัน 2 ที่ Error Correcting Level ระดับ M

2.2.6 การวาง Data Masking (Data masking) คือ เป็นการช่วยปรับสมดุลของโมดูลดำและโมดูลขาว ซึ่งมีรายละเอียด [1] ดังต่อไปนี้

2.2.6.1 Data Masking ไม่ถูกนำไปคิดกับ Function Pattern

2.2.6.2 ใช้ XOR ระหว่าง Module Pattern ที่เราได้ก่อนหน้ากับ Data Masking ทุกแบบ ทำการสลับค่าตามผลที่ได้จาก XOR ส่งผลให้อาจมีการสลับโมดูลขาวเป็นดำหรือตรงข้ามซึ่งขึ้นอยู่กับโมดูลดำของ Data Masking Pattern

2.2.6.3 ประเมินทุกการเปลี่ยนแปลง Pattern ด้วยการให้คะแนน Penalty กับรูปแบบที่ไม่ต้องการในการเปลี่ยนแปลงผลลัพธ์แต่ละอัน

2.2.6.4 เลือก Pattern ที่ได้คะแนน Penalty น้อยสุด

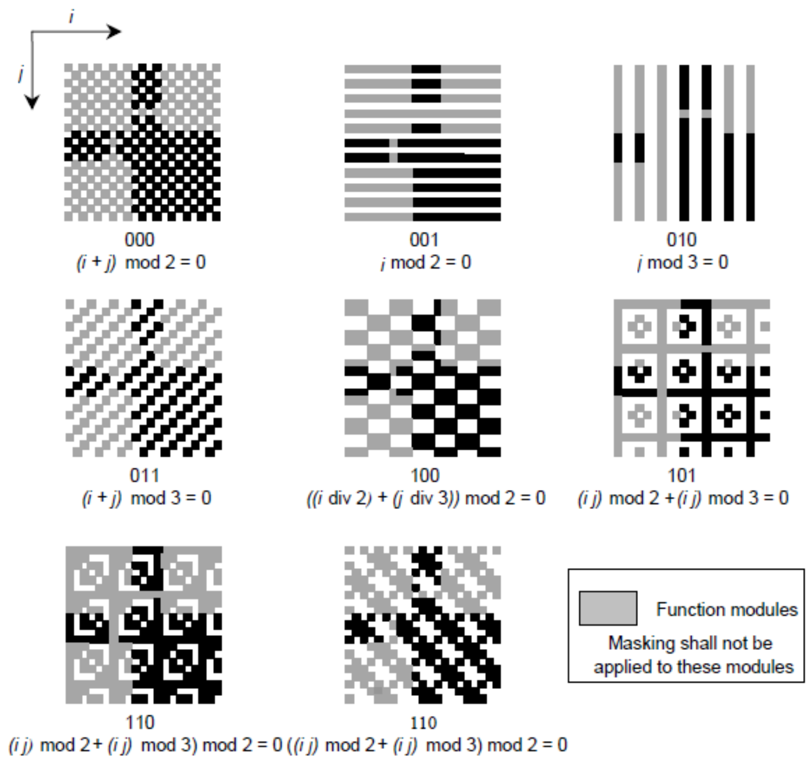
ในตารางที่ 2-6 แสดงรูปแบบ Data Masking อ้างอิง (บิตอ้างอิงสำหรับใช้ใน Format Information) ที่ Data Mask Pattern ถูกสร้างด้วยการกำหนดเป็นโมดูลดำทุกโมดูลในพื้นที่ที่เข้ารหัส (แยกพื้นที่ที่กำหนดไว้สำหรับ Format Information และ Version Information) ที่มี

เงื่อนไขเป็นจริง ซึ่งเงื่อนไข i หมายถึง ตำแหน่งแถวของโมดูลที่สงสัยและ j หมายถึง ตำแหน่งคอลัมน์ของโมดูลที่สงสัย โดยที่ (i, j) = (0, 0) คือ ตำแหน่งโมดูลซ้ายบนสุด

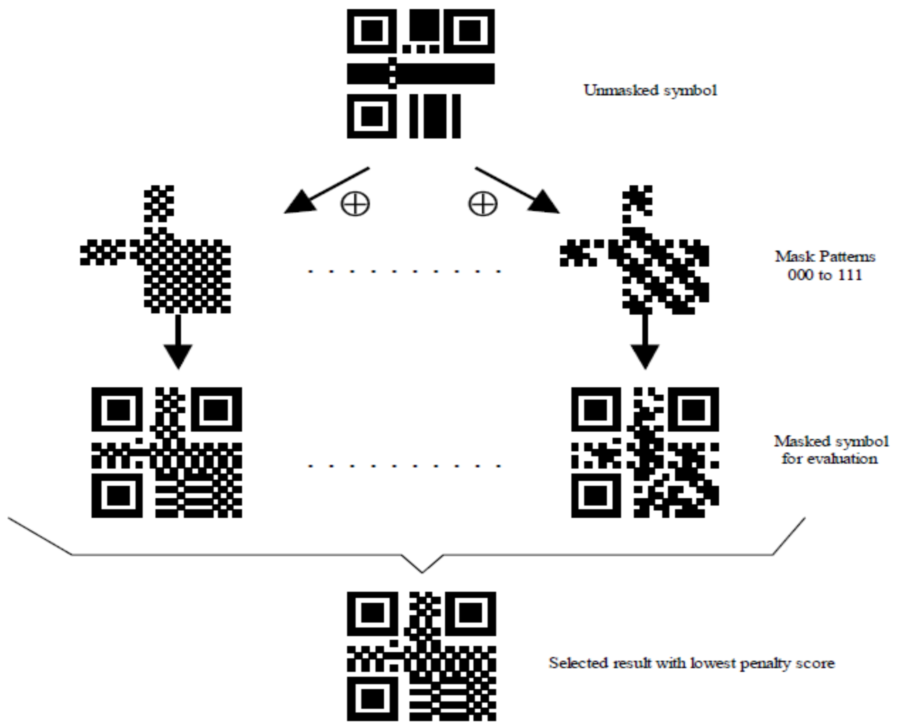
ตารางที่ 2-6 เงื่อนไขการเลือก Data mask pattern

Data mask pattern reference for QR Code symbols	Condition
000	$(i + j) \bmod 2 = 0$
001	$i \bmod 2 = 0$
010	$j \bmod 3 = 0$
011	$(i + j) \bmod 3 = 0$
100	$((i \text{ div } 2) + (j \text{ div } 3)) \bmod 2 = 0$
101	$(i+j) \bmod 2 + (i+j) \bmod 3 = 0$
110	$((i+j) \bmod 2 + (i+j) \bmod 3) \bmod 2 = 0$
111	$((i+j) \bmod 2 + (i+j) \bmod 3) \bmod 2 = 0$

ตัวอย่างเช่น ใน Mask Pattern ที่ 000 : ที่ตำแหน่ง (20, 20) จะได้ (20 + 20) mod 2 = 0 แปลว่าให้สลับสีบิตจากเดิม, ที่ตำแหน่ง (19, 20) จะได้ (19+ 20) mod 2 = 1 แปลว่าบิตสีเดิม, … ทำจนครบทุก Patternต่อไปเป็นการแสดง Data Mask Pattern ซึ่งแสดงอยู่ในเวอร์ชัน 1 ตัวเลข 3 บิตข้างล่างในแต่ละ Pattern แสดง Data Mask Pattern อ้างอิง ที่ตรงตามเงื่อนไขที่แสดงในตารางที่ 2-7



ภาพที่ 2-11 รูปแบบ Data Mask Pattern ทั้งหมด



ภาพที่ 2-12 การแสดงขั้นตอนเลือก Data Masking ใน QR Code

หลังจากทำ Data Masking กับแต่ละ Pattern ทุกอันคำนวณ Point ว่าได้เท่าไหร่สำหรับแต่ละ Pattern ซึ่งถ้า Pattern ใดยิ่งได้คะแนนมากจะไม่นำมาใช้ ในตารางที่ 2-8 ตัวแปร N_1 ถึง N_4 แสดงน้ำหนักของคะแนน Penalty เพื่อดูว่า ตัวไหนที่เราไม่ต้องการ ซึ่งกำหนดให้ค่า $N_1 = 3$, $N_2 = 3$, $N_3 = 40$ และ $N_4 = 10$ ที่มีค่า i เป็นค่าที่จำนวนของโมดูลที่ติดกันของสีเดียวกันเกินกว่า 5 และค่า k เป็นอัตราของความคลาดเคลื่อนของอัตราส่วนของโมดูลดำใน QR Code จาก 50% ใน 5% ถึงแม้ว่าการคำนวณส่วนนี้ไม่ได้รวม Format Information แต่พิจารณาแต่ขอบเขตการเข้ารหัสข้อมูลเท่านั้นก็ยังถือว่าเป็นการคำนวณที่สมบูรณ์ [1]

ตารางที่ 2-7 คะแนนของผล Data Masking

Feature	Evaluation condition	Points
Adjacent modules in row/column in same color	No. of modules = (5 + i)	$N_1 + i$
Block of modules in same color	Block size = m x n	$N_2 \times (m - 1) \times (n - 1)$
1 : 1 : 3 : 1 : 1 ratio (dark:light:dark:light:dark) pattern in row/column, preceded or followed by light area 4 modules wide	Existence of the pattern	N_3
Proportion of dark modules in entire symbol	50 ± (5 × k)% to 50 ± (5 × (k + 1))%	$N_4 \times k$

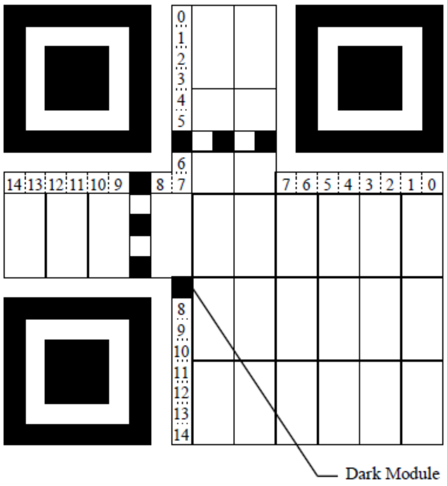
2.2.7 Format and version information คือ ในส่วนนี้จะเกี่ยวข้องกับ Error Correcting และเป็นข้อมูลในการคำนวณ Error Correcting ซึ่งมีรายละเอียด [1] ดังต่อไปนี้

2.2.7.1 Format Information คือ ในส่วนของ Format Information มีความยาว 15 บิตที่บรรจุข้อมูล 5 บิต และ Error Correction 10 บิต ซึ่งคำนวณโดยใช้การเข้ารหัส BCH (15, 5) [1] โดยในส่วนของข้อมูล 5 บิตมี 2 บิตแรกที่บอก Error Correction Level ดังตารางที่ 2-9

ตารางที่ 2-8 Indicators ของ Error Correction Level ใน QR Code

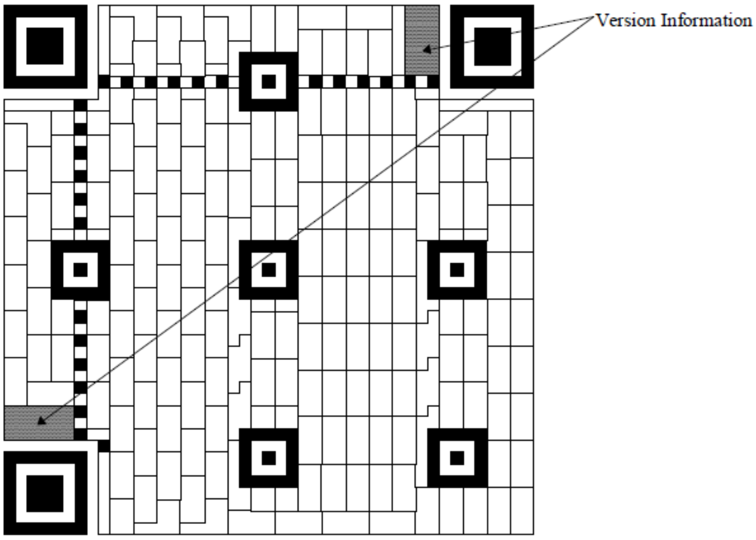
Error Correction Level	Binary Indicator
L	01
M	00
Q	11
H	10

มีบิตที่ 3 ถึง 5 ของข้อมูล 5 บิต บรรจุ Data Mask Pattern ที่กล่าวไปแล้วในตารางที่ 2-11 ที่ถูกเลือก ส่วน 10 บิตที่เป็น Error Correction นั้นการคำนวณจะกล่าวไว้ใน ซึ่งจะถูกรวมเข้ากับข้อมูล 5 บิต โดยทั้ง 15 บิตจะ XOR กับ Mask Pattern ที่มีค่า 101010000010010 เพื่อตรวจดูว่าไม่มีการผสมกันของ Error Correction Level กับ Data Mask Pattern ซึ่งจะให้ผลลัพธ์เป็น String ที่มีค่าเป็น 0 หมด การวาง Format Information ดังภาพที่ 2-13 ซึ่งจะเห็นว่ามี Format Information ปรากฏ 2 ตำแหน่งเพื่อป้องกันการถอดรหัสผิดพลาด บิตที่ความสำคัญน้อยที่สุดจะอยู่ที่โมดูล 14 และ บิตที่ความสำคัญมากที่สุดจะอยู่ที่โมดูล 0 ดังภาพที่ 2-13 โมดูลในตำแหน่ง (4V + 9, 8) ที่ V เป็นเวอร์ชันมักเป็นสีดำและไม่ได้เป็นผลมาจาก Format Information



ภาพที่ 2-13 ตำแหน่งของ Format Information

2.2.7.2 Version Information คือ ในส่วนของ Version Information จะปรากฏที่เวอร์ชัน 7 ขึ้นไป ใน Version Information จะประกอบด้วย 18 บิตที่มีข้อมูล 6 บิตและมี Error Correction 12 บิต ซึ่งคำนวณโดยใช้การเข้ารหัส Golay (18, 6) [1] ซึ่งรายละเอียดการคำนวณ Error Correction ในส่วนของ Version Information ซึ่งในส่วนของข้อมูล 6 บิตจะบรรจุเวอร์ชันของ QR Code โดยที่บิตที่มีความสำคัญมากที่สุดอยู่ในตำแหน่งแรก ในส่วนของ 12 บิตที่เป็นError Correction จะถูกคำนวณโดยรายละเอียดอยู่ใน จะถูกรวมเข้ากับข้อมูลที่มี 6 บิต หลักจากนั้น Version Information จะถูกจับคู่กับพื้นที่ที่ไว้สำหรับใส่ Version Information ดังภาพที่ 2-14 สังเกตว่า Version Information จะปรากฏ 2 ตำแหน่งเพื่อป้องกันการถอดรหัสผิดพลาด โดยบิตที่มีความสำคัญน้อยสุดจะวางไว้ที่ตำแหน่งเลข 0 และ บิตที่มีความสำคัญมากสุดจะวางไว้ที่ตำแหน่งเลข 17 ดังภาพที่ 2-14 ซึ่งมีตัวอย่างแสดงการเข้ารหัส Version Information เช่น เวอร์ชันที่ 7 เมื่อแปลงเป็นเลขไบนารีได้ 000111 แล้วเข้ารหัสแบบ BCH ได้ 110010010100 ทำให้ได้ Format Information Module Pattern เป็น 000111110010010100 พื้นที่ของ Version Information มีขนาด 6 x 3 Module Blocks ซึ่งอยู่เหนือ Timing Pattern ใกล้ Finder Pattern ฝั่งขวาบนและ Finder Pattern ฝั่งซ้ายล่าง ซึ่งทั้งสองถูกกั้นด้วย Separator ดังภาพที่ 2-14



ภาพที่ 2-14 ตำแหน่ง Version Information

0	3	6	9	12	15
1	4	7	10	13	16
2	5	8	11	14	17

ภาพที่ 2-15 การเก็บค่าไบนารีของ Version Information ฝั่งซ้ายล่าง

0	1	2
3	4	5
6	7	8
9	10	11
12	13	14
15	16	17

ภาพที่ 2-16 การเก็บค่าไบนารีของ Version Information ฝั่งขวาบน

2.3 ขั้นตอนการถอดรหัส (Decode Procedure Overview)

ขั้นตอนการถอดรหัสจากการอ่าน QR Code ให้กลายเป็นข้อมูล Characters ออกมานั้นมีขั้นตอนตรงกันข้ามกับขั้นตอนการเข้ารหัส [1] ซึ่งภาพที่ 2-17 แสดงเค้าโครงของกระบวนการทั้งหมดซึ่งกระบวนการมีดังต่อไปนี้

2.3.1 Recognize Black/White Modules คือ นำภาพ QR Code ที่ได้มาตรวจจับโมดูลสีดำกับโมดูลสีขาวเป็นบิต “0” และ “1” แล้วทำการระบุ Reflectance Polarity จากสีโมดูล Finder Pattern

2.3.2 Decode Format Information คือ อ่าน Format Information ปลด Masking pattern และแสดง Error Correction บนโมดูล Format Information ถ้าสำเร็จจะอยู่ในตำแหน่งปกติ แต่ในกรณีอื่นๆ จะเกิดการถอดรหัสของ Format Information เป็นแบบ Mirror Image (สลับโมดูลขาวเป็นดำ หรือ โมดูลดำเป็นขาว) จากนั้นระบุ Error Correction Level และ Data Mask Pattern

2.3.3 Determine Version คือ อ่าน Version Information (กรณี QR Code เวอร์ชัน 7 ขึ้นไป) หลังจากนั้นให้พิจารณาเวอร์ชันของ QR Code

2.3.4 Release Masking คือ ปลด Data Masking ด้วยวิธี XOR บริเวณที่เข้ารหัสด้วย Data Mask Pattern จะถูกสกัดออกมาจาก Format Information

2.3.5 Restore Data and RS Codewords คือ อ่าน Characters ตามกฎการวางเพื่อแก้ไข Data และ Error Correction Codewords

2.3.6 Error Detection using Error Correction Codewords คือ ตรวจสอบความผิดพลาดโดยใช้ Error Correction Codewords ที่สอดคล้องกับ Level Information ถ้าตรวจพบข้อผิดพลาดให้แก้ไข

2.3.7 Error Correction คือ แบ่ง Data Codewords เป็นส่วนๆ ที่สอดคล้องกับ Mode Indicators และ Character Count Indicators

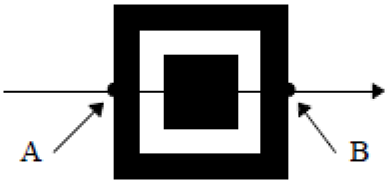
2.3.8 Decode Data Codewords คือ ถอดรหัส Data Characters ตาม Modes ที่ใช้จะได้ผลลัพธ์ออกมา

2.3.9 ส่วนต่อไปจะเป็นอัลกอริทึมของการถอดรหัสซึ่งจะทำการหาภาพ QR Code และทำการถอดรหัส ซึ่งจะมีขั้นตอนดังต่อไปนี้

2.3.9.1 แปลงรูปภาพเป็นกลุ่มของพิกเซลดำและขาวโดยใช้การตัด Threshold

2.3.9.2 หาตำแหน่งของ Finder Pattern ซึ่งมีทั้งหมด 3 ตำแหน่งซึ่งอยู่ตามมุม 3 ใน 4 ของ QR Code ซึ่ง Finder Pattern มีขนาดอัตราส่วนเป็น 1 : 1 : 3 : 1 : 1 ซึ่งความคลาดเคลื่อนยินยอมเป็น 0,5 เช่น ช่วงของ 0,5 ถึง 1,5 สำหรับ Single Module Box และ 2,5 ถึง 3,5 สำหรับ Three Module Box โดยที่

2.3.9.2.1 เมื่อพื้นที่ที่อยู่ในความสนใจ (Finder Pattern) มีการตรวจพบตำแหน่งของตำแหน่งแระและตำแหน่งสุดท้าย A และ B ตามลำดับ ดังรูปที่ 2.19 ที่เส้นของพิกเซลในรูปพบขอบด้านนอกของ Finder Pattern ทำซ้ำกับเส้นพิกเซลที่อยู่ติดกันจนเส้นทั้งหมดจนผ่านกึ่งกลางของ Finder Pattern ในแนวแกน x



ภาพที่ 2-17 แสกนเส้นใน Finder Pattern

2.3.9.2.2 ทำขั้นตอนที่ 2.3.9.2.1 ซ้ำจนคอลัมน์พิกเซลในแนวแกน y ผ่านกึ่งกลางของ Finder Pattern

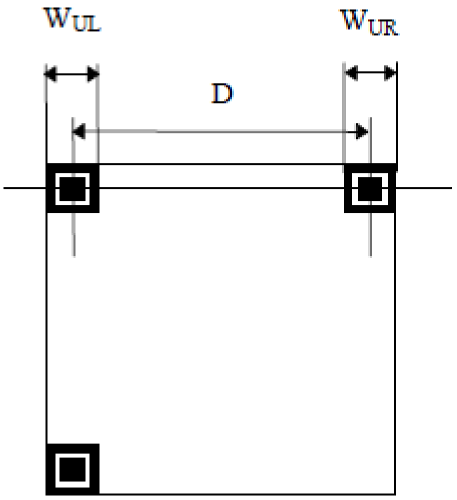
2.3.9.2.3 หาตำแหน่งกลางของ Pattern โดยสร้างเส้นผ่านกึ่งกลางระหว่างตำแหน่ง A และ B บนเส้นพิกเซลด้านนอกสุดซึ่งข้ามกึ่งกลางของ Finder Pattern ในแนวแกน x และ สร้างเส้นผ่านกึ่งกลางระหว่างตำแหน่ง A และ B บนเส้นพิกเซลด้านนอกสุดซึ่งข้ามกึ่งกลางของ Finder Pattern ในแนวแกน y จะหาตำแหน่งกึ่งกลางของ Pattern จากการตัดกันของทั้งสองเส้น

2.3.9.2.4 ทำขั้นตอนที่ 2.3.9.2.1 – 2.3.9.2.3 ซ้ำเพื่อหาตำแหน่งของ Finder Pattern อีก 2 ตำแหน่ง

2.3.9.2.5 เมื่อไม่มีพื้นที่ที่สนใจอีก (หา Finder Pattern ครบ) ให้แปลงสีของพิกเซลขาวและดำแล้วเริ่มทำใหม่ที่ขั้นตอนที่ 2.3.9.2 เพื่อถอดรหัส

2.3.9.3 พิจารณาแนวการหมุนของ QR Code ด้วยการวิเคราะห์คู่อันดับกึ่งกลางของ Finder Pattern เพื่อระบุ Pattern ใดอยู่ Pattern ตำแหน่งซ้ายบนและมุมของการหมุนของ QR Code

2.3.9.4 พิจารณาขั้นตอนที่ 2.3.9.1 ความกว้างของ D ซึ่งวัดจากกึ่งกลางของ Finder Pattern ซ้ายบนและขวาบนและพิจารณาขั้นตอนที่ 2 ความกว้างของ Finder Pattern ซ้ายบนและขวาบน ซึ่งเขียนด้วย W_{UL} และ W_{UR} ดังภาพที่ 2-18



ภาพที่ 2-18 ความกว้างของ Finder Pattern ซ้ายบนและขวาบน

2.3.9.5 คำนวณ Nominal X Dimension ของ QR Code ดังสมการต่อไปนี้

$$X = \frac{(W_{UL}+W_{UR})}{14} \qquad (2\text{-}11)$$

2.3.9.6 เงื่อนไขในการคำนวณเวอร์ชันแทนด้วย V ของ QR Code ดังสมการต่อไปนี้

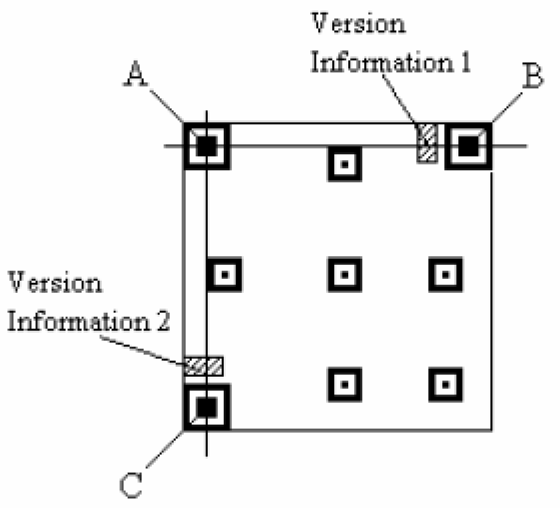
$$V = \frac{\left[\left(\frac{D}{X}\right)-10\right]}{4} \qquad (2\text{-}12)$$

2.3.9.7 ถ้าเงื่อนไขสรุปออกมาเป็นเลข 6 หรือน้อยกว่าให้หมายถึงเวอร์ชันเลย แต่ถ้าเงื่อนไขสรุปออกมาเป็นเวอร์ชัน 7 ให้ถอดรหัส Version Information ดังขั้นตอนต่อไปนี้

2.3.9.7.1 แบ่งความกว้างของ W_{UR} ของ Finder Pattern บนขวาด้วยการหาร 7 เพื่อคำนวณหาขนาดของ CP_{UR} ดังสมการต่อไปนี้

$$CP_{UR} = \frac{W_{UR}}{7} \tag{2-13}$$

2.3.9.7.2 หาเส้น Guide Line AC และ AB จากจุด A, B และ C ซึ่งแต่ละจุดผ่านกึ่งกลางของ Finder Pattern ทั้ง 3 อัน ดังภาพที่ 2-20 Sampling Grid ของแต่ละกึ่งกลาง Module ในพื้นที่ Version Information 1 ถูกคำนวณโดยขึ้นกับเส้นที่ขนานกับ Guide Line ซึ่งคู่อันดับกึ่งกลางของ Finder Pattern ที่มีขนาดของโมดูล CPUR ค่าไบนารี "0" และ "1" ถูกพิจารณาจาก Light Pattern หรือ Dark Pattern บน Sampling Grid



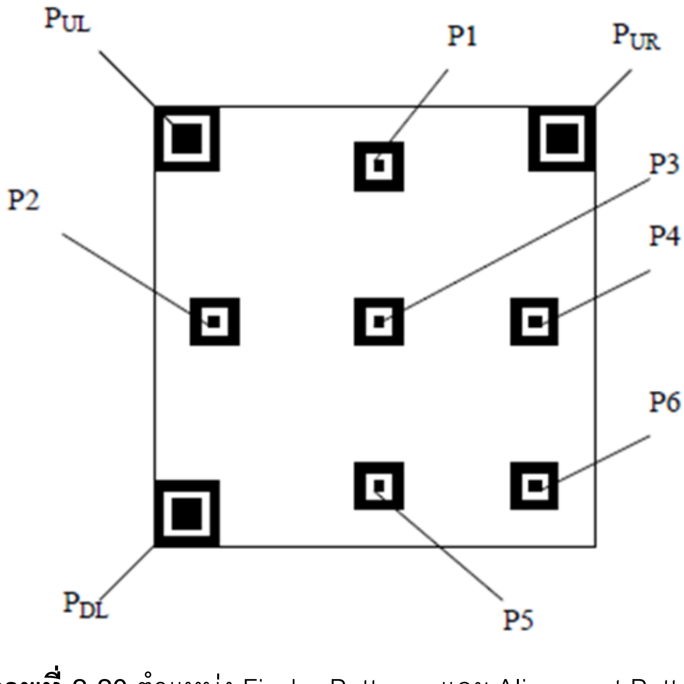
ภาพที่ 2-19 ตำแหน่ง AC และ AB จากจุด A, B และ C

2.3.9.7.3 พิจารณาเวอร์ชันด้วยการตรวจจับ Error Correction ซึ่งขึ้นกับตารางใน

2.3.9.7.4 จากภาพที่ 2-20 ถ้าพบ Error เกินกว่า Error Correction ที่ถูกตรวจพบ ให้คำนวณความกว้างของ W_{DL} ของ Finder Pattern ซ้ายล่าง

และทำตามขั้นตอน 2.3.9.7.1 – 2.3.9.7.3 ข้างบนเพื่อถอดรหัส Version Information 2

2.3.9.8 สำหรับเวอร์ชัน 1 ให้กำหนดค่าใหม่โดยให้ X เป็นการประมาณช่องว่างของจุดกึ่งกลางของโมดูลดำและโมดูลขาวใน Timing Pattern โดยทำนองเดียวกันคำนวณให้ Y เป็นการประมาณช่องว่างของจุดกึ่งกลางของโมดูลดำและโมดูลขาวใน Timing Pattern ฝั่งซ้าย สร้าง Sampling Grid ขึ้นกับขั้นตอนที่ 1 ที่เส้นแนวแกนนอนลากผ่าน Timing Pattern ด้านบนด้วยเส้นที่ขนานกันที่ตำแหน่งช่องว่างแนวตั้งของ Y ซึ่งประกอบด้วย 6 เส้นเหนือ Timing Pattern และทุกเส้นใต้ Timing Pattern เป็นเวอร์ชันของ QR Code และขั้นตอนที่ 2 เส้นแนวตั้งลากผ่าน Timing Pattern ฝั่งซ้ายด้วยเส้นที่ขนานกันที่ตำแหน่งช่องว่างแนวนอนของ X ซึ่งประกอบด้วย 6 เส้นของเส้นอ้างอิงแนวตั้งและทุกเส้นทางขวาของ Timing Pattern ถูกกำหนดให้เป็นเวอร์ชันของ QR Code สำหรับเวอร์ชัน 2 หรือมากกว่าให้พิจารณาคู่อันดับกึ่งกลางของแต่ละ Alignment Pattern จากคู่อันกับที่กำหนดใน Annex E และสร้าง Sampling Grid ด้วยแนวช่องว่างที่มีระยะห่างเท่ากันระหว่างจุด ดังภาพที่ 2-21 ซึ่งมีขั้นตอนดังกระบวนการต่อไปนี้



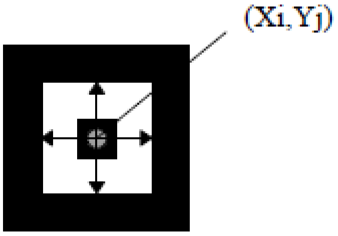
ภาพที่ 2-20 ตำแหน่ง Finder Patterns และ Alignment Pattern

2.3.9.8.1 แบ่งความกว้าง Pattern ด้วย W_{UL} ของ Finder Pattern ซ้ายบน P_{UL} ด้วยการหารด้วย 7 เพื่อคำนวณหาขนาดของโมดูล CP_{UL} ดังนี้

$$CP_{UL} = \frac{W_{UL}}{7} \tag{2-14}$$

2.3.9.8.2 พิจารณาคู่อันดับของ Alignment Patterns P1 และ P2 (ตามภาพที่ 2-20) ซึ่งขึ้นกับคู่อันดับกึ่งกลาง A ของ Finder Pattern ฝั่งซ้ายบน P_{UL} เส้นขนานกับ Guide Lines เส้น AB และ AC รวมถึงขนาดโมดูลของ CP_{UL}

2.3.9.8.3 แสกนขอบนอกของสี่เหลี่ยมสีขาวใน Alignment Pattern P1 และ P2 เริ่มจากพิกเซลของคู่อันดับกึ่งกลางเพื่อหาคู่อันดับกึ่งกลาง Xi และ Yj ดังภาพที่ 2-22



ภาพที่ 2-21 คู่อันดับกึ่งกลางของ Alignment Pattern

2.3.9.8.4 ประมาณคู่อันดับกึ่งกลางของ Alignment Pattern P3 โดยขึ้นกับคู่อันดับกึ่งกลางของ Finder Pattern ซ้ายบน P_{UL} และคู่อันดับกึ่งกลางของ Alignment Pattern P1 และ P2 ที่ได้จากขั้นตอนที่ 2.3.9.8.3

2.3.9.8.5 หาคู่อันดับกึ่งกลางของ Alignment Pattern P3 ตามขั้นตอนที่ 2.3.9.8.3

2.3.9.8.6 หา L_X ซึ่งวัดระยะทางจากกึ่งกลางถึงกึ่งกลางของ Alignment Pattern P2 และ P3 ซึ่ง L_Y ซึ่งวัดระยะทางจากกึ่งกลางถึงกึ่งกลางของ Alignment Pattern P1 และ P3 ทำการแบ่ง L_X และ L_Y ด้วยการกำหนดพื้นที่ว่างของ Alignment Pattern เพื่อหา CP_X ส่วนล่าง

และ CP_Y ในส่วนขวาในพื้นที่บนซ้ายของ QR Code (ดังภาพที่ 2-21) ได้ดังนี้

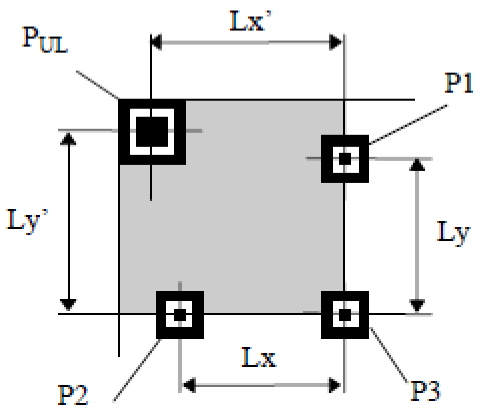
$$CP_X = \frac{L_X}{AP} \quad (2\text{-}15)$$

$$CP_Y = \frac{L_Y}{AP} \quad (2\text{-}16)$$

โดยที่ AP เป็นพื้นที่ว่างของกึ่งกลาง Alignment Pattern (ดูใน Table E.1) ในทางเดียวกันหา LX’ ซึ่งเป็นระยะทางแนวนอนระหว่างกึ่งกลางคู่อันดับของ Finder Pattern บนซ้าย PUL และคู่อันดับกึ่งกลางของ Alignment Pattern P1 และ LY’ ซึ่งเป็นระยะทางแนวตั้งระหว่างกึ่งกลางคู่อันดับของ Finder Pattern บนซ้าย PUL และคู่อันดับกึ่งกลางของ Alignment Pattern P2 แบ่ง LX’ และ LY’ ด้วยสูตรข้างล่างเพื่อหา CPX’ ในฝั่งซ้ายบนและ CPY’ ในฝั่งซ้ายในพื้นที่ซ้ายบนของ QR Code ดังสมการต่อไปนี้

CP_X’ = L_X’ / (Column Coordinate ของคู่อันดับกึ่งกลางของ Alignment Pattern P1 - Column Coordinate กึ่งกลางของ Finder Pattern P_{UL} ฝั่งซ้ายบน) (2-17)

CP_Y’ = L_Y’ / (Row Coordinate ของคู่อันดับกึ่งกลางของ Alignment Pattern P2 - Row Coordinate กึ่งกลางของ Finder Pattern P_{UL} ฝั่งซ้ายบน) (2-18)

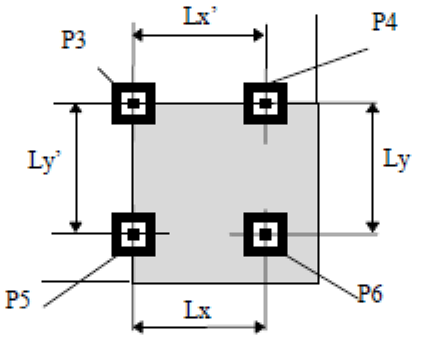


ภาพที่ 2-22 พื้นที่ฝั่งซ้ายบนของ QR Code

2.3.9.8.7 พิจารณา Sampling Grid ที่ปกคลุมพื้นที่ฝั่งซ้ายบนของ QR Code ซึ่งขึ้นกับ CP_x , CP_x', CP_Y และ CP_Y' ซึ่งแสดงแต่ละด้านของพื้นที่ฝั่งซ้ายบนของ QR Code

2.3.9.8.8 ในทางเดียวกันพิจารณา Sampling Grid สำหรับพื้นที่ฝั่งขวาบน (ปกคลุมด้วย Finder Pattern ฝั่งขวาบน P_{UR} , Alignment Pattern P1, Alignment Pattern P3 และ Alignment Pattern P4) และพื้นที่ฝั่งซ้ายล่าง (ปกคลุมด้วย Finder Pattern ฝั่งขวาบน P_{UR} , Alignment Pattern P2, Alignment Pattern P3 และ Alignment Pattern P5) ของ QR Code

2.3.9.8.9 สำหรับ Alignment Pattern P6 (ในภาพที่ 2-22) ประมาณคู่อันดับกึ่งกลางจากระดับของ CP_X' และ CP_Y' ซึ่งค่าแต่ละตัวมาจากพื้นที่ว่างของ Alignment Pattern P3, P4 และ P5 ที่มี Guide Lines ลากผ่านกึ่งกลางของ Alignment Pattern P3, P4 และ P3, P5 ตามลำดับซึ่งเป็นคู่อันดับกึ่งกลางของ Pattern



ภาพที่ 2-23 แสดงพื้นที่ล่างขวาของ QR Code

2.3.9.8.10 ทำขั้นตอนที่ 2.3.9.8.5 ถึง 2.3.9.8.8 ซ้ำเพื่อพิจารณา Sampling Grid สำหรับพื้นที่ล่างขวาของ QR Code

2.3.9.8.11 ด้วยหลักการเดียวกันจะต้องนำมาใช้เพื่อตรวจสอบ Sampling Grids สำหรับพื้นที่ใดๆ ของ QR Code ที่ยังไม่ถูก Cover

2.3.9.9 สุ่มพื้นที่ตัวอย่างขนาด 3 X 3 พิกเซลที่กึ่งกลางบนแต่ละส่วนที่ซ้อนทับกันของ Grid Lines และพิจารณาว่าเป็นโมดูลดำหรือโมดูลขาวโดยขึ้นกับการตัด Threshold สร้างเมตริกซ์จับคู่กับโมดูลดำให้เป็นไบนารีค่า “1” และโมดูลสีขาวให้เป็นไบนารีค่า “0”

2.3.9.10 ถอดรหัส Format Information ติดกันกับ Finder Pattern ซ้ายบนซึ่งรายละเอียดอยู่ใน เพื่อให้ Error Correction Level และ Data Mask Pattern ถูกใช้กับ QR Code ถ้ามี Error Correction เมื่อ Format Information ถูกพบ หลังจากนั้นให้ทำกระบวนการเดิมเพื่อถอดรหัส Format Information ที่ติดกันกับ Finder Pattern บนขวาและ Finder Pattern ล่างซ้าย

2.3.9.11 ถ้า Format Information หาไม่ได้ให้คิดว่าลำดับของ Format Information ถูกอ่านในทิศสวนทางและดำเนินการถอดรหัสต่อเป็น Mirror Image ที่แถวและคอลัมน์ของคู่อันดับทรานสโพส

2.3.9.12 ทำ XOR กับ Data Mask Pattern ในพื้นที่เข้ารหัสของ QR Code เพื่อปลด Data Masking และแก้ไข Characters และ Error Correction Codewords ซึ่งทำให้สลับผลของกระบวนการ Data Masking ซึ่งถูกใช้ระหว่างขั้นตอนเข้ารหัส

2.3.9.13 พิจารณา Codewords ตามกฎการวาง QR Code ตามหัวข้อ 2.2.2.7 ซึ่งมีหลักการโดยรวม ดังต่อไปนี้

2.3.9.13.1 จัดลำดับ Codeword ใหม่เป็น Blocks ตามข้อกำหนดของเวอร์ชัน และ Error Correction Level

2.3.9.13.2 ไล่ตรวจข้อผิดพลาดและแก้ไขให้ถูกในการถอดรหัสตามขั้นตอนใน Annex B เพื่อแก้ไขข้อผิดพลาดให้ถูกต้องและ Erasures เพิ่มขึ้นเพื่อเพิ่มขนาดของการแก้ไขข้อผิดพลาดสำหรับเวอร์ชันและ Error Correction Level

2.3.9.13.3 กู้คืนข้อความต้นฉบับที่เป็นกระแสบิตข้อมูลด้วยการประกอบเป็น Data Block ตามลำดับ

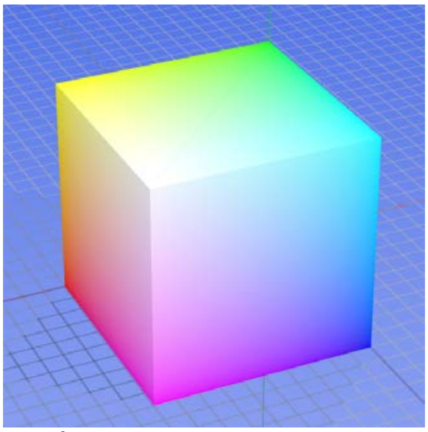
2.3.9.13.4 แบ่งกระแสข้อมูลบิตอย่างละเอียดเป็น Segment ที่เริ่มต้นด้วย Mode Indicator และความยาวที่ถูกพิจารณาด้วย Character Count Indicator ตาม Mode Indicator

2.3.9.13.5 ถอดรหัสตามกฎของ Mode ที่ใช้

2.4 พื้นฐานเบื้องต้นเกี่ยวกับภาพ (Principle of image)

ในการแสดงลักษณะสีของแต่ละรูปภาพที่ใช้กันอยู่ ในคอมพิวเตอร์นั้นใช้การเลือกแบบจำลองสี (Color Model) โดยแต่ละแบบจำลองสีมีคุณสมบัติที่แตกต่างกันออกไป ส่งผลให้นำไปใช้งานที่แตกต่างกัน ซึ่งยังขึ้นกับเทคโนโลยีที่ใช้ของแต่ละประเทศ แบบจำลองสีจะใช้สีหลักซึ่งเราเรียกว่าแม่สีมาผสมกันในการแสดงค่าสีอื่นๆ [16], [17] โดยแม่สีหลักแตกต่างตามแต่ละชนิด ดังต่อไปนี้

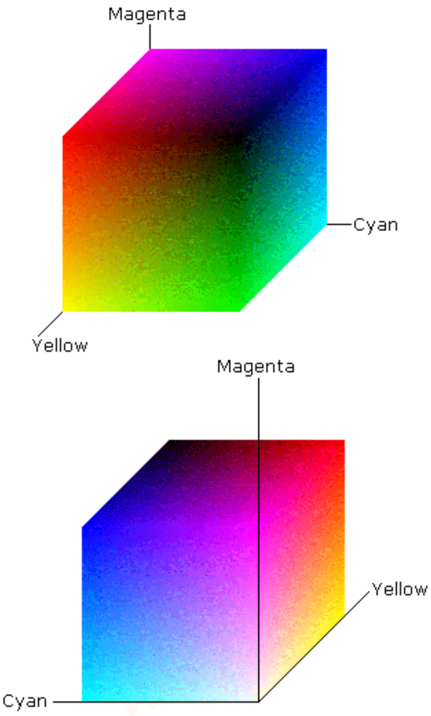
2.4.1 แบบจำลองสี RGB คือ มีสีหลัก 3 สี ได้แก่ แดง (Red) เขียว (Green) และน้ำเงิน (Blue) ซึ่งใช้แสดงสีทั้งหมด ตามโครงสร้างของสีหลัก ดังรูปต่อไปนี้



ภาพที่ 2-24 โครงสร้างของสีหลัก RGB

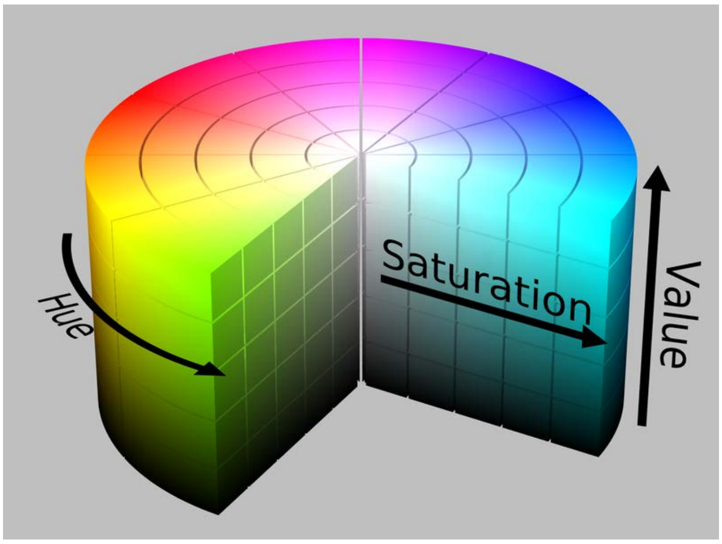
โดยจุดกำเนิด (R,G,B) = (0, 0, 0) เป็นสีดำ ที่จุด (1, 1, 1) = เป็นสีขาว โดยที่ค่าในแต่ละแกนเป็นการแสดงถึงความเข้มของแต่ละแม่สี การแสดงค่าของแต่ละแม่สีอยู่ระหว่าง 0 ถึง 1 เพื่อให้อยู่ในลักษณะอัตราส่วนและเป็นอิสระต่อกัน โดยฮาร์ดแวร์ไม่มีผลต่อการแสดงสีในกรณีที่เราพิจารณาค่าสี ในส่วนการแสดงค่าความเข้มในระดับเทา (Gray Scale) จะแสดงในแนวเส้นทแยงมุมพุ่งออกจากจุดกำเนิด (0, 0, 0) เป็นเส้นตรงไปยังตำแหน่ง (1, 1, 1) ที่เป็นสีขาว การผสมสีของแบบจำลอง RGB มักใช้ในการแสดงผลทางหน้าจอคอมพิวเตอร์ เนื่องจากจอแสดงผลคอมพิวเตอร์ทั่วไปใช้หลักการผสมสีแบบเดียวกัน ซึ่งเป็นสาเหตุให้แบบจำลองสี RGB ถูกนำมาใช้ในงานทำภาพพิมพ์ลายน้ำดิจิตอล

2.4.2 แบบจำลองสี CMY หรือ CMYK คือ แบบจำลองสี CMY หรือ CMYK ได้แก่ สีเขียวน้ำเงิน (Cyan) สีม่วงแดง (Magenta) และสีเหลือง (Yellow) ส่วนใหญ่มักใช้ในการพิมพ์ภาพผ่านเครื่องพิมพ์ โดยหลักการของการแสดงสีแบบจำลองสีนี้จะใช้การลบออก ถ้าแสงสีขาวเกิดการตกระทบกับหมึกสีเหลืองจะพบว่าสีเหลืองจะประกอบด้วยสีแดงผสมสีเขียวในแบบจำลองของ RGB ซึ่งแสงที่สะท้อนจากสีเหลืองจะไม่หรากฎแสงสีน้ำเงินเนื่องจากแสงสีน้ำเงินถูกดูดซับเอาไว้ด้วยหมึกสีเหลืองลบจากแสงสีขาว หรือ สีน้ำเงินถูกลบออกจากสีขาว ทำให้สะท้อนเห็นเป็นสีเหลือง แบบจำลอง CMY หรือ CMYK จุดกำเนิดอยู่ที่ (0, 0, 0) เป็นสีขาว ส่วนจุด (1, 1, 1) เป็นสีดำ ทำให้เราอาจมองได้ว่าแบบจำลองสี CMY หรือ CMYK เพิ่มอีกหนึ่งสี คือ สีดำ (Black) ดังนั้นจึงเห็นชื่อแบบจำลองสีเขียน CMYK แบบจำลอง CMY หรือ CMYK มีลักษณะดังรูป [17] ต่อไปนี้



ภาพที่ 2-25 โครงสร้างของสีหลักของ CMY หรือ CMYK

2.4.3 แบบจำลองสี HSV คือ ในแบบจำลองสีที่ได้กล่าวไปถึงก่อนหน้า ในการจะที่จะแสดงค่าความสว่างในระดับใดระดับหนึ่งนั้นทำได้ยาก เพื่อแก้ปัญหาให้สามารถเพิ่มหรือลดค่าความสว่างของสีใดๆ ได้นั้นจึงเกิดการคิดค้นแบบจำลองที่ตัวแปรหลัก 3 ตัว คือ ค่าสี (Hue), ค่าความอิ่มตัว (Saturation) และค่าความสว่าง (Value) โดยใช้หลักการบอกเข้าไปในแบบจำลอง RGB ในทิศที่ขนานกับแกนสีเทาเริ่มต้นจากตำแหน่งสีขาวเข้าไป ดังแสดงในภาพที่ 2-27 [17] จะเห็นว่ารูปร่างเหมือนทรงกระบอก โดยแกนมุมสูง คือ ค่าความสว่าง ส่วนแกนรัศมี คือ ค่าอิ่มตัว และมุมของแกนรัศมี คือ ค่าของสี



ภาพที่ 2-26 โครงสร้างของสีหลักของ HSV

2.5 **ลายน้ำดิจิตอล** (Digital Watermark)

ลายน้ำดิจิตอลถูกนำมาใช้ในกิจกรรมต่าง ๆ มากมาย โดยวัตถุประสงค์คือป้องกันการปลอมแปลงแต่ในงานวิจัยนี้เราต้องการนำมาใช้ในการซ่อนข้อมูลเพื่อป้องกันไม่ให้ข้อมูลบางส่วนถูกเปิดเผยต่อบุคคลที่สามหรือคนที่ไม่มีส่วนเกี่ยวข้อง เมื่อระบบในปัจจุบันเปลี่ยนรูปแบบไปเป็นระบบดิจิตอล ทำให้ลายน้ำอนาลอกที่เคยใช้กันได้ถูกเปลี่ยนรูปแบบมาเป็นลายน้ำดิจิตอล [18] ในงานวิจัยนี้จึงทดลองนำลายน้ำดิจิตอลเข้ามาใช้กับข้อมูล QR Code เพื่อใช้ในการซ่อนข้อมูล

2.5.1 นิยามของลายน้ำดิจิตอล คือ ลายน้ำดิจิตอลคือข้อมูลดิจิตอลที่ถูกฝังลงไปในสื่อประสม เช่น ภาพนิ่ง เสียง และวีดีทัศน์ เป็นต้น ในงานวิจัยนี้ใช้สนใจการฝังลายน้ำดิจิตอลลงในข้อมูล QR Code โดยทั่วไปลายน้ำดิจิตอลมักถูกนำมาใช้แสดงความเป็นเจ้าของผลงานหรือนวัตกรรม และป้องกันการละเมิดสิทธิ์ทางปัญญา [18] ในงานวิจัยชิ้นนี้เล็งเห็นว่าการฝังลายน้ำสามารถนำมาใช้ในการซ่อนข้อมูลเพื่อช่วยในการรักษาความลับของเจ้าของข้อมูล QR Code ได้ โดยปกติลายน้ำดิจิตอลที่นำไปฝังแบ่งเป็นแบบที่มองเห็นได้ คือ จะแสดงให้เห็นลายน้ำดิจิตอลได้อย่างชัดเจนจากการมองเห็นด้วยสายตาและแบบที่ไม่สามารถมองเห็นได้ด้วยสายตา คือ ใช้วิธีการเฉพาะเพื่อการสืบค้น ซึ่งในงานวิจัยชิ้นนี้เราสนใจแบบที่สอง คือ แบบที่ไม่สามารถมองเห็นได้ด้วยสายตา โดยหลักแล้วลายน้ำดิจิตอลทั้งสองแบบที่กล่าวมานั้นจะต้องไม่สามารถถูกดึงออก

จากสื่อได้โดยง่าย โดยในงานวิจัยชิ้นนี้ได้กำหนดการเข้ารหัสและถอดรหัสโดยใช้กุญแจลับหรือคีย์เป็นรูปภาพ

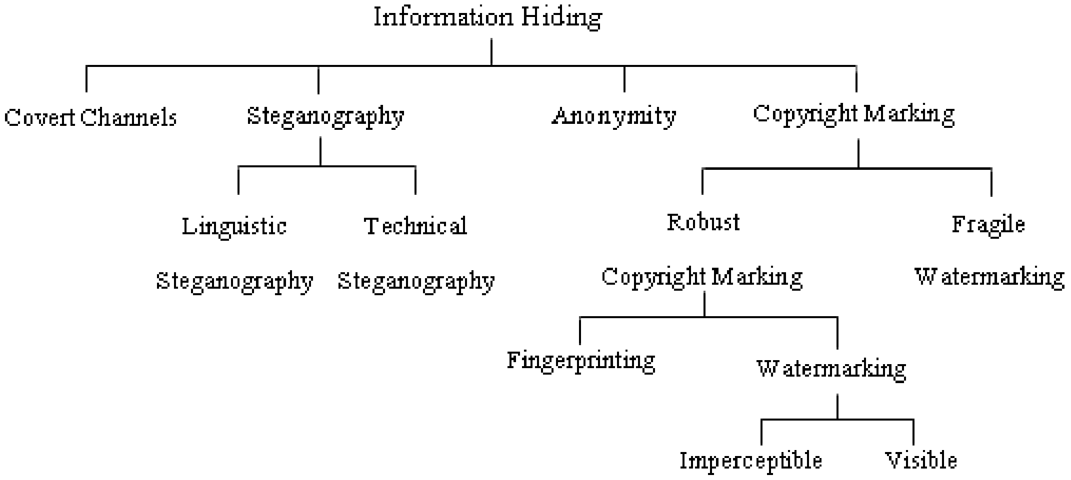
2.5.2 หลักการในการทำลายน้ำดิจิตอล คือ โดยทั่วไปเทคนิคในการทำลายน้ำดิจิตอลมีรูปแบบคล้ายๆ กัน คือ เป็นการใส่สัญญาณเข้าไปในตัวข้อมูลหลักก่อนที่จะทำการเผยแพร่หรือแจกจ่าย โดยงานวิจัยชิ้นนี้ทำการใส่สัญญาณลายน้ำดิจิตอลซึ่งเป็นข้อมูลที่เราต้องการให้เป็นเป็นความลับลงในข้อมูล QR Code โดยทั่วไปการทำลายน้ำดิจิตอลทุกประเภทประกอบด้วยขั้นตอนทั่ว ๆ ไปคือ

2.5.2.1 การใส่ลายน้ำดิจิตอล (Watermark embedding)

2.5.2.2 การตรวจสอบ (Detection) หรือการนำลายน้ำดิจิตอลออกมา ข้อมูล QR Code จะถูกใส่สัญญาณลายน้ำเข้าไปโดยจะมีค่าขึ้นอยู่กับกุญแจลับหรือคีย์ ซึ่งใช้ในการเข้ารหัสโดยสิ่งที่คาดหมาย คือ มีเพียงผู้ที่ถือกุญแจลับหรือคีย์นี้เท่านั้น ที่สามารถตรวจสอบสัญญาณลายน้ำหรือเปลี่ยนแปลงข้อมูลภายในได้ ซึ่งการเข้าถึงข้อมูลภายในจำเป็นต้องใช้กุญแจลับในการนำสัญญาณลายน้ำที่ถูกต้องกลับคืนมา

2.5.3 วิธีการฝังลายน้ำดิจิตอล คือ วิธีการฝังลายน้ำดิจิตอลเข้าไปในข้อมูล QR Code ต้นฉบับแบ่งออกเป็นสองวิธี วิธีแรกคือการฝังลายน้ำดิจิตอลในรูปแบบโดเมนพื้นที่ เป็นการเลือกจุดตำแหน่งของ QR Code แล้วทำการปรับแต่งค่าโดยพิจารณาจากตำแหน่งจุดของข้อมูล QR Code ต้นฉบับนั้น ๆ เป็นสำคัญ ซึ่งมักจะเรียกวิธีการแบบนี้กระทำบน “Spatial Domain” ส่วนอีกวิธีหนึ่งนั้นเป็นการฝังลายน้ำดิจิตอลในรูปแบบโดเมนความถี่ หรือที่มักเรียกกันว่า “Frequency Domain” ซึ่งเป็นผลมาจากการใช้วิธีการแปลง (Transform) แบบต่าง ๆ ซึ่งในงานวิจัยนี้สนใจการแปลง 3 แบบเปรียบเทียบกัน คือการแปลงโคซายน์แบบไม่ต่อเนื่อง (Discrete Cosine Transform, DCT), การแปลงดีสครีตฟูเรียร์ทรานฟอร์ม (Discrete Fourier Transform, DFT) และ การแปลงเวฟเล็ตไม่ต่อเนื่อง (Discrete wavelet transform, DWT) โดยทำการแก้ไขค่าสัมประสิทธิ์ของการแปลงค่าที่มีอยู่ในช่วงความถี่ที่เหมาะสม วิธีการฝังลายน้ำดิจิตอลเป็นส่วนหนึ่งของวิธีซ่อนข้อมูล (Information Hiding) แสดงดังภาพที่ 2-28 ซึ่งนับว่าเป็นวิธีการที่สามารถนำประยุกต์ใช้ได้

อย่างกว้างขวางไม่ว่าจะเป็นในภาพนิ่ง วีดีทัศน์และเสียง รวมถึงงานวิจัยนี้ที่นำวิธีการซ่อนข้อมูลมาใช้กับข้อมูล QR Code ซึ่งแต่ละวิธีในภาพจะมีวิธีการฝังลายน้ำดิจิตอลที่แตกต่างกันออกไป [21]



ภาพที่ 2-27 โครงสร้างของวิธีซ่อนข้อมูล

จากภาพที่ 2-27 จะมีรายละเอียด โดยจำแนกเป็นหกประเภทแบบคร่าวๆ ดังนี้ [19, 20, 21]

2.5.3.1 Perceptible Technique การฝังวิธีนี้มองเห็นลายน้ำดิจิตอลซ้อนทับอยู่บนข้อมูลต้นฉบับอย่างชัดเจน ทำให้วิธีการนี้จะมนุษย์เห็นลายน้ำดิจิตอลได้อย่างชัดเจน

2.5.3.2 Imperceptible Technique วิธีนี้ทำให้มนุษย์ไม่สามารถเห็นลายน้ำดิจิตอลได้อย่างชัดเจน

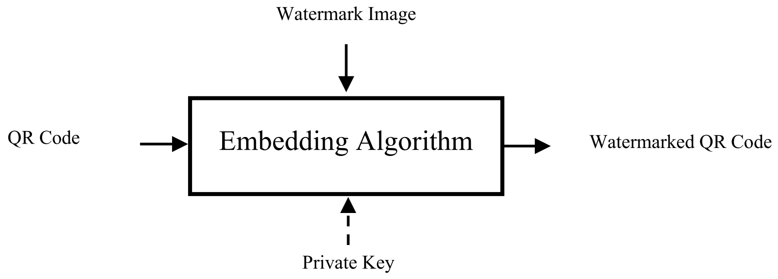
2.5.3.3 Fragile Technique วิธีนี้พิจารณาความเปราะบางของลายน้ำดิจิตอลที่ถูกฝังไว้ ซึ่งมีความไวต่อการเสียหายและการเปลี่ยนแปลงแม้เพียงเล็กน้อย เมื่อมีการแก้ไขข้อมูลต้นฉบับที่ผ่านการฝังลายน้ำจะทำให้ไม่สามารถถอดลายน้ำดิจิตอลออกมาได้ ซึ่งเป็นรูปแบบที่งานวิจัยชิ้นนี้สนใจโดยงานวิจัยไม่ได้มุ่งเน้นการโจมตีข้อมูลต้นฉบับที่ผ่านการฝังลายน้ำเรียบร้อย

2.5.3.4 Robust Technique วิธีนี้ข้อมูลต้นฉบับที่ผ่านการฝังลายน้ำแล้ว ลายน้ำดิจิตอลจะยังอยู่หากการโจมตีไม่รุนแรงเกินไป

2.5.3.5 Source Extraction Technique หรือ Non-Blind Technique วิธีนี้ลายน้ำดิจิตอลต้องใช้ข้อมูลต้นฉบับช่วยในการถอดลายน้ำดิจิตอลที่ซ่อนไว้ จากข้อมูลต้นฉบับที่ผ่านการฝังลายน้ำเรียบร้อยแล้วเพื่อให้สามารถถอดลายดิจิตอลออกมาได้

2.5.3.6 Destination Extraction Technique หรือ Blind Technique วิธีนี้ไม่ใช้ข้อมูลต้นฉบับในการถอดลายน้ำดิจิตอลออกมา ซึ่งเป็นรูปแบบที่งานวิจัยชิ้นนี้สนใจ

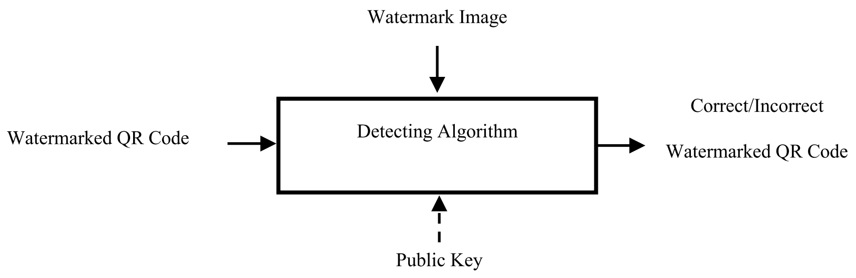
2.5.4 การฝังลายน้ำดิจิตอล (Watermark embedding) คือ ในงานวิจัยนี้ การฝังลายน้ำดิจิตอลเป็นกระบวนการใส่ลายน้ำดิจิตอลให้ฝังเข้าไปอยู่ในข้อมูล QR Code ต้นฉบับ โดยข้อมูลเริ่มต้นของกระบวนการฝังลายน้ำดิจิตอลนี้จะประกอบด้วย ข้อมูล QR Code, ลายน้ำดิจิตอลที่จะนำมาใส่ลงไปในข้อมูล QR Code เพื่อใช้ในการซ่อนข้อมูล และรหัสลับส่วนตัว (Private Key) ด้วย [22] เมื่อเตรียมข้อมูลทั้งสองส่วนนี้เสร็จ หลังจากนั้นจะถูกนำเข้าสู่กระบวนการฝัง (Embedding Algorithm) หลังจากผ่านกระบวนการดังกล่าวแล้วข้อมูลที่ได้ก็คือผลลัพธ์ที่มีลายน้ำดิจิตอลฝังอยู่ดังภาพที่ 2-28



ภาพที่ 2-28 การฝังลายน้ำดิจิตอลลงในข้อมูล QR Code

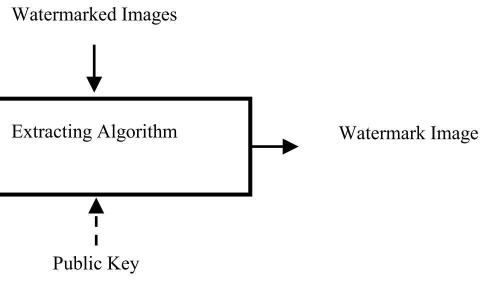
2.5.5 การตรวจหาลายน้ำดิจิตอล (Watermark detection) คือ เมื่อมีความต้องการลายน้ำดิจิตอลที่เราทำการฝังในข้อมูล QR Code ต้นฉบับเรียบร้อยแล้วออกมา เริ่มต้นด้วยการค้นหาว่าในข้อมูล QR Code ที่สงสัยว่ามีลายน้ำดิจิตอลหรือข้อมูลลับที่ฝังอยู่หรือไม่ จึงนำไปสู่กระบวนการแสดงดังภาพที่ 2-30 ข้อมูลนำเข้าของกระบวนการตรวจหาลายน้ำดิจิตอลนี้ คือ

ข้อมูลที่ต้องสงสัยว่ามีลายน้ำดิจิตอลฝังอยู่ ซึ่งในงานวิจัยนี้มีการกำหนดรหัสลับในกระบวนการฝังและการถอดลายน้ำดิจิตอลด้วยเพื่อให้ผู้ที่ถือกุญแจเท่านั้นจึงจะสามารถแก้ไขข้อมูลต่างๆ ได้



ภาพที่ 2-29 การตรวจหาลายน้ำดิจิตอลหรือข้อมูลลับที่อาจฝังอยู่ในข้อมูลที่ผ่านการฝังลายน้ำ

2.5.6 การถอดลายน้ำดิจิตอล (Watermark Extraction) คือ หลังจากนั้นขั้นถัดมาคือ การถอดลายน้ำดิจิตอล การถอดลายน้ำดิจิตอลจะมีแนวทางปฏิบัติอยู่สองแบบ ได้แก่ แบบแรก คือ การถอดลายน้ำดิจิตอลโดยไม่ใช้ข้อมูลต้นฉบับ (Blind Watermarking) ซึ่งงานวิจัยชิ้นนี้ทำการศึกษา แบบสอง คือ เป็นการถอดลายน้ำดิจิตอลที่ใช้ข้อมูลต้นฉบับ ซึ่งทั้งสองรูปแบบเป็นกระบวนการที่ทำให้ได้ลายน้ำดิจิตอลและข้อมูลต้นฉบับก่อนถูกฝังลายน้ำดิจิตอลกลับคืนมา แสดงดังภาพที่ 2-30



ภาพที่ 2-30 การถอดลายน้ำดิจิตอลออกจากข้อมูลที่มีลายน้ำดิจิตอลฝังอยู่

2.5.7 เกณฑ์การวัดประสิทธิภาพของลายน้ำดิจิตอล

2.5.7.1 เกณฑ์การวัดประสิทธิภาพของการฝังลายน้ำดิจิตอล [23, 24, 21] โดยแบ่งออกเป็น

ก) เกณฑ์เชิงคุณภาพ (Subjective Fidelity Criteria) วัดประสิทธิภาพโดยแบ่ง เป็นสองกลุ่ม คือกลุ่มคนที่มีพื้นฐานทางการประมวลผลภาพ และกลุ่มคนที่ไม่มีพื้นฐานด้านนี้มาก่อน หลังจากนั้นให้กลุ่มตัวอย่างทั้งสองมาพิจารณาภาพที่ฝังภาพลายน้ำดิจิตอลแล้วพร้อมกัน จากนั้นให้คะแนนตามเกณฑ์ที่กำหนด (Subjective Fidelity Scoring Scale) ข้อเสียวิธีนี้คือผลการทดลองจะไม่มีเกณฑ์มาตรฐานเพียงพอ ซึ่งอาจเกิดจากกลุ่มตัวอย่างมีความลำเอียงหรือความพึงพอใจในการให้คะแนน

ข) เกณฑ์เชิงปริมาณ (Objective Fidelity Criteria) วัดประสิทธิภาพโดยการใช้การคำนวณทางคณิตศาสตร์เป็นเกณฑ์ ข้อดีคือสะดวกต่อการใช้งาน และมีเกณฑ์มาตรฐานที่น่าเชื่อถือหรือรองรับ งานวิจัยนี้ใช้เกณฑ์เชิงปริมาณเป็นเกณฑ์การวัดผลการทดลอง แบ่งเป็น 3 ส่วนตามกระบวนการวิจัย คือ กระบวนการฝังลายน้ำดิจิตอล ใช้ค่า PSNR (Peak signal-to-noise ratio), กระบวนการถอดลายน้ำดิจิตอลใช้ค่า NC (Normalized Cross Correlation) และ เกณฑ์การวัดเวลาที่เครื่องคอมพิวเตอร์ใช้โดยแบ่งเป็น การวัดเวลาในกระบวนการฝังลายน้ำดิจิตอลและการวัดเวลาในกระบวนการถอดลายน้ำดิจิตอล

2.5.7.2 การตรวจสอบหาค่าความผิดเพี้ยนของข้อมูล QR Code โดยใช้ค่า PSNR เป็นตัววัดเกณฑ์เชิงปริมาณที่ใช้ประเมินค่าลายน้ำดิจิตอลที่ใส่เพิ่มลงไปในข้อมูลหลัก (ข้อมูล QR Code ต้นฉบับ) โดยมีหน่วยวัดเป็นเดซิเบล (dB:Decibel) ซึ่งวิธีคำนวณแสดงได้ดังสมการที่ 2-19 และสมการที่ 2-20 [21]

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right) \quad dB \tag{2-19}$$

โดยสมการของการหาค่า *MSE* (Mean-squared-error) เขียนได้ดังนี้

$$MSE = \frac{\sum (f_w(x,y) - f(x,y))^2}{n} \tag{2-20}$$

เมื่อค่า $f_w(x,y)$ เป็นข้อมูล QR Code ที่มีลายน้ำดิจิตอลฝังอยู่

$f(x,y)$ เป็นข้อมูล QR Code ต้นฉบับ

n เป็นจำนวนจุดภาพในภาพที่มีลายน้ำดิจิตอลฝังอยู่และข้อมูล QR Code ต้นฉบับ

โดยคุณภาพการผิดเพี้ยนของข้อมูล QR Code ที่ผ่านการฝังลายน้ำดิจิตอลแล้วจะแปรผกผันตามค่า PSNR คือหากค่า PSNR น้อยแสดงว่าข้อมูล QR Code ที่ผ่านการฝังลายน้ำดิจิตอลแล้วมีการผิดเพี้ยนจากข้อมูล QR Code ต้นฉบับมากและหากค่า PSNR มากแสดงว่า มีการผิดเพี้ยนจากข้อมูล QR Code ต้นฉบับน้อย

2.5.7.3 การวัดค่าความเหมือนของภาพ (Similarity Measurement) ด้วยค่า *NC* คือจะใช้วัดค่าความเหมือนระหว่างลายน้ำดิจิตอลต้นฉบับและลายน้ำดิจิตอลที่ถอดออกมาได้ทั้งก่อนและหลังจะอยู่ในช่วงระหว่างค่า 0 ถึงค่า 1 สำหรับวิธีการคำนวณค่าความเหมือนของข้อมูลสองมิติ สามารถทำได้ตามสมการที่ 2-21 [21]

$$NC = \frac{\sum_{i}\sum_{j} W_{(i,j)} - W'_{(i,j)}}{\sum_{i}\sum_{j} \left[W_{(i,j)}\right]^{2}} \qquad (2\text{-}21)$$

เมื่อ $W_{(i,j)}$ และ $W'_{(i,j)}$ แทนค่าความเข้มสีของจุดภาพ ณ ตำแหน่ง (*i,j*) ในลายน้ำดิจิตอลต้นฉบับและลายน้ำดิจิตอลที่ถอดออกมาตามลำดับ โดยคุณภาพของลายน้ำดิจิตอลจะแปรผันตาม ค่าความเหมือนของภาพคือค่าความเหมือนสูงมากเท่าใดแสดงว่าลายน้ำดิจิตอลที่ถอดออกมาจะเหมือนกับภาพต้นฉบับมากขึ้นด้วยเช่นกัน

2.5.7.4 เกณฑ์การวัดเวลาที่เครื่องคอมพิวเตอร์ใช้ในการฝังและถอดลายน้ำดิจิตอล (Time for Embedding and Extracting) เป็นการคำนวณเวลาที่ใช้ในการประมวลผลแต่ละการแปลงข้อมูลเข้าสู่โดเมนความถี่ โดยคำนวณได้ตามสมการที่ 2-22

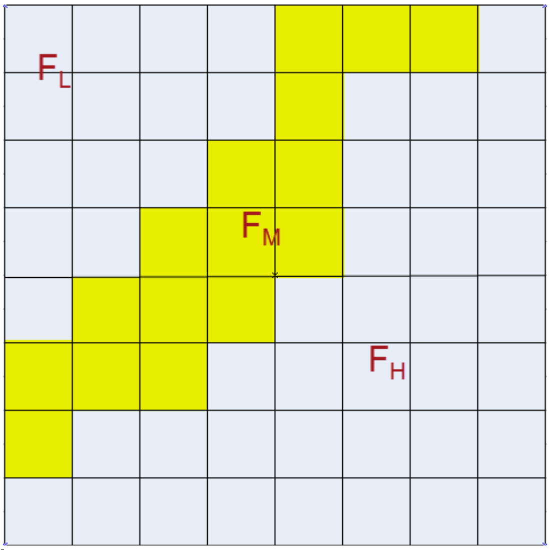
$$\Delta T = T_{START} - T_{STOP} \qquad (2\text{-}22)$$

2.6 เทคนิคการกระบวนการฝังและถอดลายน้ำในโดเมนความถี่

จะเป็นการแปลงภาพด้วยวิธีทางคณิตศาสตร์ก่อนการฝังลายน้ำลงไป โดยในงานวิจัยชิ้นนี้ใช้การแปลงแบบ การแปลงโคซายน์แบบไม่ต่อเนื่อง (Discrete cosine transform, DCT), การแปลงฟูริเยร์เต็มหน่วย (Discrete fourier transform, DFT) และการแปลงเวฟเล็ตแบบไม่ต่อเนื่อง (Discrete wavelet transform, DWT) ซึ่งจะได้ค่าสัมประสิทธิ์จากการแปลงดังกล่าว

จากนั้นจะทำการปรับแต่งค่าสัมประสิทธิ์เหล่านั้น โดยมีเงื่อนไขว่าจะต้องไม่ทำให้คุณภาพของข้อมูล QR Code ต้นฉบับต่ำลงมากนัก โดยจะกล่าวถึงการแปลงแต่ละแบบ มีคุณลักษณะดังต่อไปนี้

2.6.1 การแปลงโคซายน์แบบไม่ต่อเนื่อง (Discrete cosine transform, DCT) การแปลงโคซายน์แบบทำทีละ Block เป็นกระบวนการคำนวณทางคณิตศาสตร์ที่คล้ายกับ FFT (Fast fourier transform) เนื่องจากเอาสัญญาณมาแปลงจากรูปแบบหนึ่งเป็นอีกรูปแบบหนึ่งมักถูกใช้กับการประมวลผลข้อมูลดิจิตอล เป็นการแปลงกลุ่มของข้อมูลให้เข้าสู่ค่าของความถี่ โดยในยุคเริ่มต้น [24] ซึ่งได้นำเสนอวิธีการในการทำภาพพิมพ์ลายน้ำดิจิตอลที่อ้างอิงกับวิธีการแปลงสัญญาณโคซายน์แบบไม่ต่อเนื่อง (Discrete cosine transform, DCT) โดยสัญญาณลายน้ำจะถูกนำไปแทรกลงภายในช่วงความถี่กลางของค่าสัมประสิทธิ์ที่ได้จากการแปลง DCT ของข้อมูล QR Code ต้นฉบับ ดังภาพที่ 2-31 [25, 26]



ภาพที่ 2-31 ความหมายของขอบเขต DCT ช่วงความถี่กลาง

ในกระบวนการที่สร้างค่าตัวอย่างข้อมูลที่ได้จากชุดของสัมประสิทธิ์ที่ผ่านการแปลงฟังก์ชันการแปลงสัญญาณโคซายน์แบบไม่ต่อเนื่อง เรียกว่า IDCT (Inverse discrete cosine transform) โดยถ้ามีข้อมูลมากกว่า 8 ค่า เราสามารถที่จะแบ่งข้อมูลเป็นกลุ่ม กลุ่มละ 8 ค่าแล้วทำการแปลงแยกกันได้ เพราะฟังก์ชันจะถูกเรียกใช้เหมือนกันทุกกลุ่ม มีเพียงค่าสัมประสิทธิ์เท่านั้นที่แตกต่างในแต่ละกลุ่ม โดยจะมีสมการ การแปลง DCT ในแบบ Two-dimensional (ในแนวแกนตั้งและ

แนวแกนนอน) และการแปลงกลับ หรือ IDCT ดังสมการ 2-23 และ 2-29 ตามลำดับดังต่อไปนี้ [27, 28]

$$Y = DCT\big(f(x,y)\big) \tag{2-23}$$

$f(x,y)$ คือ บริเวณของข้อมูล QR Code ต้นฉบับ ซึ่งถูกจำกัดความด้วยสมการต่อไปนี้

$$X = [f(x,y)]_{N\times N} \tag{2-24}$$

โดยที่ X คือ ความเข้มของจุดภาพที่อยู่ในโดเมนเวลา เมื่อ $0 \le x,y \le N-1$ ตำแหน่ง โดย Y คือ บริเวณของข้อมูล QR Code ต้นฉบับที่มีค่าอยู่ในโดเมนความถี่ ซึ่งถูกจำกัดความด้วยสมการต่อไปนี้

$$Y = [C(u,v)]_{N\times N} \tag{2-25}$$

โดยที่ $C(u,v)$ คือ ค่าที่อยู่ในโดเมนความถี่ เมื่อ $0 \le u,v \le N-1$ ตำแหน่ง ซึ่งกำหนด DCT ดังนี้

$$C(u,v) = \alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)cos\left(\frac{(2x+1)u\pi}{2N}\right)cos\left(\frac{(2y+1)v\pi}{2N}\right) \tag{2-26}$$

เมื่อ

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}}, & u = 0 \\ \sqrt{\frac{2}{N}}, & u = 1,2,\dots,N-1 \end{cases} \tag{2-27}$$

$$\alpha(v) = \begin{cases} \sqrt{\frac{1}{N}}, & v = 0 \\ \sqrt{\frac{2}{N}}, & v = 1,2,\dots,N-1 \end{cases} \tag{2-28}$$

ในการแปลงค่าซึ่งอยู่ในสัญญาณความถี่ให้กลับเป็นสัญญาณดิจิตอลดังเดิมด้วยการทำ IDCT ทุกบริเวณจนทั่วทั้งภาพสามารถเขียนสมการได้ดังนี้

$$\hat{X} = IDCT\left(\widehat{f(x,y)}\right) \quad (2\text{-}29)$$

$\hat{Y}$ คือ บริเวณของข้อมูล QR Code ต้นฉบับที่มีค่าอยู่ในโดเมนความถี่ซึ่งถูกกำหนดด้วยสมการ

$$\hat{Y} \equiv [C(u,v)]_{N\times N} \quad (2\text{-}30)$$

โดยที่ $C(u,v)$ คือ ค่าที่อยู่ในโดเมนความถี่ เมื่อ $0 \le u,v \le N-1$ ตำแหน่ง โดยที่ $\hat{X}$ คือ บริเวณบริเวณของข้อมูล QR Code ต้นฉบับที่มีค่าอยู่ในโดเมนความถี่ ซึ่งถูกเขียนด้วยสมการต่อไปนี้

$$\hat{X} \equiv [f(x,y)]_{N\times N} \quad (2\text{-}31)$$

โดยที่ $f(x,y)$ คือ ค่าที่อยู่ในโดเมนความถี่ เมื่อ $0 \le x,y \le N-1$ ตำแหน่ง ซึ่งกำหนดให้ IDCT มีสมการดังต่อไปนี้

$$f(x,y) = \alpha(u)\alpha(v)\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} C(u,v)cos\left(\frac{(2x+1)u\pi}{2N}\right)cos\left(\frac{(2y+1)v\pi}{2N}\right) \quad (2\text{-}32)$$

เมื่อ

$$\alpha(u) = \begin{cases}\sqrt{\frac{1}{N}}, & u = 0\\ \sqrt{\frac{2}{N}}, & u = 1,2,\ldots,N-1\end{cases} \quad (2\text{-}33)$$

$$\alpha(v) = \begin{cases}\sqrt{\frac{1}{N}}, & v = 0\\ \sqrt{\frac{2}{N}}, & v = 1,2,\ldots,N-1\end{cases} \quad (2\text{-}34)$$

2.6.2 การแปลงฟูเรียร์แบบไม่ต่อเนื่อง (Discrete fourier transform, DFT) ในการแปลงแปลงฟูเรียร์แบบไม่ต่อเนื่อง ใช้การคำนวณพิกัดลอการิธึมเชิงขั้ว (Log-Polar coordinate) เมื่อ *F* (*u*, *v*) คือสัมประสิทธิ์ที่ได้จาก Fourier transform ของข้อมูล QR Code จาก *I* (*x, y*) ซึ่งพิจารณาสมการต่อไปนี้ [29, 30]

$$F(u,v) = \frac{1}{XY}\sum_{x=0}^{X-1}\sum_{y=0}^{Y-1} I(x,y)\cdot e^{-j2\pi\left(\frac{ux}{X}+\frac{vy}{Y}\right)} \qquad (2\text{-}35)$$

$$I(x,y) = \sum_{x=0}^{X-1}\sum_{y=0}^{Y-1} F(u,v)\cdot e^{j2\pi\left(\frac{ux}{X}+\frac{vy}{Y}\right)} \qquad (2\text{-}36)$$

โดยข้อมูล QR Code ที่ถูกซ่อนลายน้ำดิจิตอล *F* ' (*u*, *v*) ในโดเมนความถี่นั้นได้จากการใช้แปลงแบบฟูเรียร์ (Fourier transform) ซึ่งพิจารณาสมการต่อไปนี้

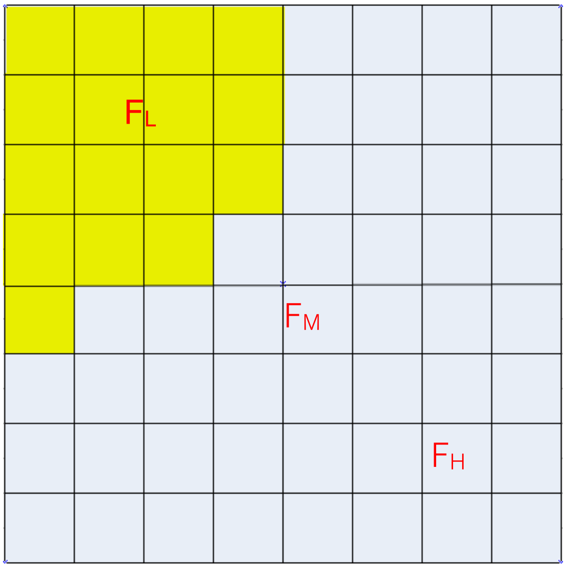
$$F'(u,v) = \left(\left|F(u,v)\right|\cdot(1+\alpha W(x,y))\right)\cdot e^{j\Phi(u,v)} \qquad (2\text{-}37)$$

โดยที่ $|F(u, v)|$ และ $\Phi(u, v)$ คือ Fourier spectrum และ Phase angle ของ $F(u, v)$ ซึ่งได้จากสมการที่ 2-36 และ 2-37 ตามลำดับ ส่วน α คือค่าที่ใช้กำหนดความเข้ม (Amplitude factor) ของลายน้ำดิจิตอลที่จะซ่อนลงในรูปภาพ สำหรับรูปภาพที่ถูกซ่อนลายน้ำดิจิตอล $I'(x, y)$ ในโดเมนปกติจะได้จากการทำ Inverse Fourier transform ตามสมการที่ 2-38 ดังนี้

$$I'(x,y) = \sum_{x=0}^{X-1}\sum_{y=0}^{Y-1} F'(u,v)\cdot e^{j2\pi\left(\frac{ux}{X}+\frac{vy}{Y}\right)} \qquad (2\text{-}38)$$

อย่างไรก็ดี ลายน้ำดิจิตอล *W* (*x*, *y*) นี้ อาจมีขนาดเท่ากับรูปภาพ *I* (*x*, *y*) หรือเล็กกว่าก็ได้ ในกรณีที่ลายน้ำดิจิตอลมีขนาดเล็กกว่ารูปภาพ ลายน้ำดิจิตอลดังกล่าวนี้ควรจะถูกเติมขอบด้วยค่า 0 (สีดำ) ให้เต็มจนมีขนาดเท่ากับรูปภาพ หากขนาดของลายน้ำดิจิตอลมีขนาดเล็กมากเกินไปจะทำให้มีเฉพาะส่วนความถี่สูงของรูปภาพเท่านั้นที่ถูกซ่อน ซึ่งจะทำให้ลายน้ำดิจิตอลมีความคงทน (Robust) ต่อกระบวนการเปลี่ยนแปลงแก้ไขรูปภาพต่างๆเป็นอย่างสูง แต่ในทางกลับกันก็

ส่งผลโดยตรงต่อคุณภาพต่อการมองเห็น (Perceptibility) ของรูปภาพ เนื่องจากบริเวณดังกล่าวเป็นบริเวณที่มีนัยสำคัญต่อการมองเห็นสูงสุด ดังนั้นเพื่อเป็นการประนีประนอมระหว่างความคงทนกับคุณภาพของรูปภาพที่ได้ ใน [26] และ [27] ได้แนะนำให้ซ่อนลายน้ำดิจิตอลลงในบริเวณตรงกลางระหว่างส่วนที่มีความถี่สูงกับความถี่ต่ำของรูปภาพ โดยในงานวิจัยเราให้การซ่อนลายน้ำดิจิตอลในบริเวณความถี่ต่ำ ดังภาพที่ 2-32



ภาพที่ 2-32 ความหมายของขอบเขต DFT ช่วงความถี่ต่ำ

2.6.3 การแปลงเวฟเล็ตแบบไม่ต่อเนื่อง (Discrete wavelet transform, DWT) มีสมการที่ใช้อธิบายดังต่อไปนี้ [21, 31, 32]

$$DWT(m,n) = \frac{1}{\sqrt{a_0^m}} \int_{-\infty}^{\infty} f(t)\psi\left(\frac{t - nb_0 a_0^m}{a_0^m}\right) dt \tag{2-39}$$

โดยที่	a_0^m	คือ	การสเกล
	$nb_0 a_0^m$	คือ	การเลื่อนตำแหน่ง
	m,n	คือ	เลขจำนวนเต็มบวก

ในทางปฏิบัติการนำมาใช้งานจริง สัญญาณที่เข้ามานี้จะอยู่ในรูปของการสุ่ม (Sample) ดังนั้นสมการที่ 2-39 จึงพัฒนามาสู่สมการที่ 2-40

$$DWT(m,n) = \frac{1}{\sqrt{a_0^m}} \sum_k f(k) \psi\left(\frac{n - kb_0 a_0^m}{a_0^m}\right) \tag{2-40}$$

เมื่อ *m,n,k* เป็นเลขจำนวนเต็ม โดยที่

n คือ จำนวนข้อมูล

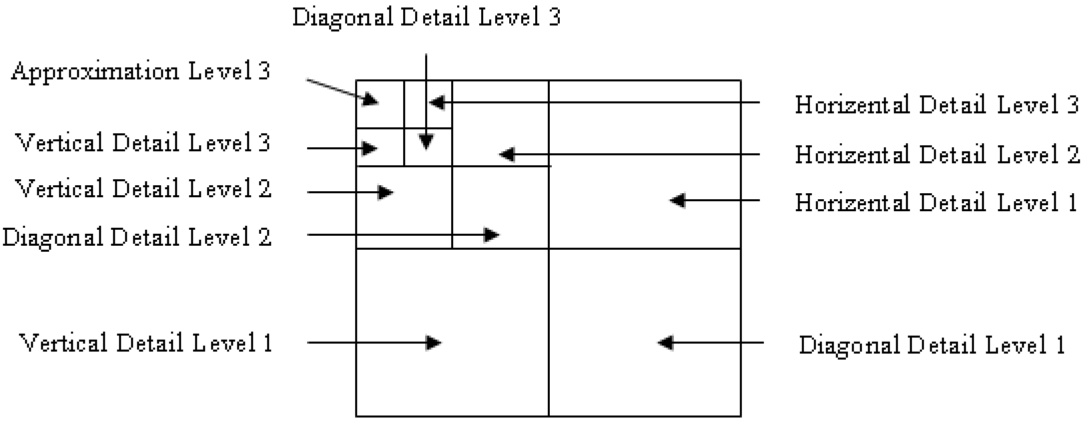
m คือ เลขแสดงการเปลี่ยนแปลงของสเกล

k คือ เลขแสดงการเลื่อนตำแหน่ง

เมื่อพิจารณาในรูปแบบของการวิเคราะห์หลายระดับความละเอียดแล้ว โดยที่จะมีการเปลี่ยนแปลงสเกลในการวิเคราะห์ให้ลดลงครั้งละสอง เท่า (a_0 = 2 ; b_0 = 1) แล้วจะได้รูปแบบการแปลงเวฟเล็ตแบบเต็มหน่วย ดังสมการต่อไปนี้

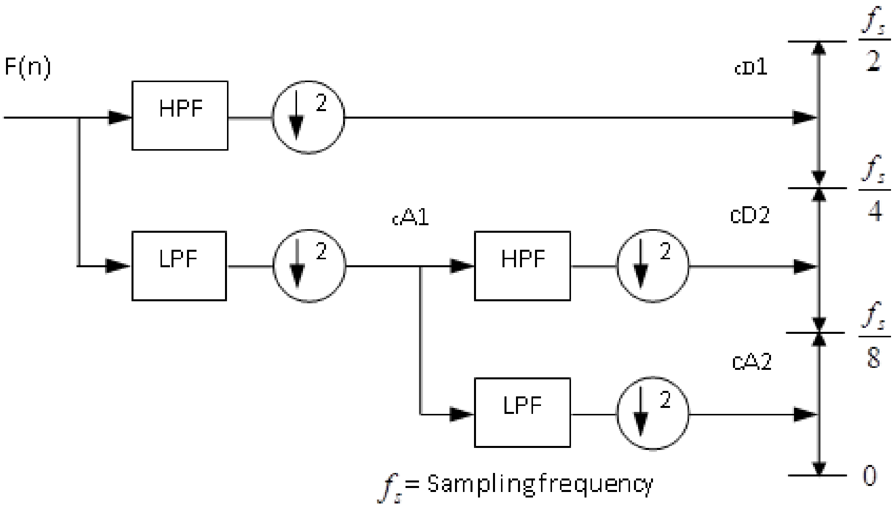
$$DWT(m,n) = \frac{1}{\sqrt{2^m}} \sum_k f(k) \psi\left(\frac{n - k2^m}{2^m}\right) \tag{2-41}$$

โดยในงานวิจัยเลือกใช้ชนิด Haar Wavelet Transform โดยการเป็นการแปลงภาพแบบไม่ต่อเนื่อง ซึ่งมีการแบ่งภาพออกเป็นส่วนๆ โดยแต่ละส่วนนั้นมีองค์ประกอบในเชิงความถี่ที่แตกต่างกันออกไป เวฟเลตเป็นการวิเคราะห์รายละเอียดของภาพจากหยาบไปละเอียด ซึ่งการแปลงแบบอื่นข้อมูลหลังจากแปลงจะไม่มีความสัมพันธ์กันกับข้อมูลต้นฉบับ โดยจะทำการแยกส่วนประกอบของภาพในลักษณะเชิงลำดับชั้น (Hierarchical) ที่จะแยกส่วนประกอบของภาพแบบเรียกซ้ำ (Recursive) และเปลี่ยนข้อมูลต้นฉบับให้อยู่ในรูปข้อมูลความถี่ โดยผ่านตัวกรองแบบ High Pass (G) และ ตัวกรองสัญญาณแบบ Low Pass (H) ส่วนที่ผ่านตัวกรองสัญญาณแบบ High Pass คือ ข้อมูลบริเวณความถี่สูงหรือรายละเอียดของข้อมูลที่มีความละเอียดน้อย และส่วนที่ผ่านตัวกรองสัญญาณแบบ Low Pass คือ ข้อมูลบริเวณความถี่ต่ำหรือรายละเอียดของข้อมูลที่มีความละเอียดมาก ดังภาพที่ 2-33 [21]



ภาพที่ 2-33 การแยกส่วนประกอบโดยใช้การแปลงเวฟเล็ต 3 ระดับ

จากภาพที่ 2-33 แสดงการแปลงแบบ Forward Wavelet Transform โดยเริ่มจากข้อมูลต้นฉบับ f(x,y) ผ่านการกรองสัญญาณทางแกน x ผลที่ได้เป็นข้อมูล Lowpass $f_L(x,y)$ และ Highpass $f_H(x, y)$ ซึ่งจะทำให้ $f_L(x,y)$ และ $f_H(x,y)$ ทางแกน x ถูกแบ่งครึ่ง Bandwidth ของสัญญาณ และทำ Downsample โดยไม่มีการสูญเสียข้อมูล จากนั้นทั้ง $f_L(x,y)$ และ $f_H(x,y)$ จะถูกกรองสัญญาณทางแกน y ผลลัพธ์ที่ได้ 4 ข้อมูลย่อย (Sub Data) ประกอบด้วย $f_{LL}(x,y)$, $f_{LH}(x,y)$, $f_{HL}(x,y)$ และ $f_{HH}(x,y)$ แล้วจึงทำ Downsample ข้อมูลย่อยอีกครั้ง ซึ่งเราจะแยกส่วนประกอบของข้อมูลเป็น $f_{LL}(x,y)$ (Approximation Signal หรือ Lowpass-Lowpass) อีก 3 ข้อมูลย่อย คือ Detail Signal ได้แก่ $f_{LH}(x,y)$ (Horizontal Detail หรือ Lowpass-Highpass), $f_{HL}(x,y)$ (Vertical Detail หรือ Highpass-Lowpass) และ $f_{HH}(x,y)$ (Diagonal Detail หรือ Highpass-Highpass) [21]



ภาพที่ 2-34 การแปลงแบบ Forward Wavelet Transform

หลังจากการแปลงข้อมูลแบบเวฟเลตแล้ว จะได้เมทริกซ์ของสัมประสิทธิ์ที่ประกอบด้วยสัญญาณเฉลี่ย (Approximation Signal) และสัญญาณรายละเอียด (Detail Signal) ส่วนการแปลง Inverse Wavelet Transform ของข้อมูลเป็นกระบวนการย้อนกลับจากการแปลงแบบ Forward Wavelet Transform โดยเริ่มจากนำสัญญาณค่าเฉลี่ยและสัญญาณละเอียดมาทำอัพแซมเปิลทางแกน y ด้วยตัวกรองสัญญาณแบบ Lowpass Inverse Filter และตัวกรองสัญญาณแบบ Highpass Inverse Filter ทั้ง 4 ข้อมูล การทำอัพแซมเปิลจะกระทำโดยการใส่ 0 ระหว่างคู่ของค่าสัญญาณในแกน y หลังจากนั้นจะทำการอัพแซมเปิลทางแกน x และสร้างข้อมูลกลับขึ้นมาใหม่ จะเหมือนภาพที่ 2-34 แต่กลับลูกศรทุกอัน

2.7 เอกสารและงานวิจัยที่เกี่ยวข้อง

จากการศึกษา QR Code พบว่ามีงานวิจัยหลายด้านที่สนใจมากมาย โดยงานวิจัยนี้เราสนใจปัญหาด้านความปลอดภัยของข้อมูล QR Code เนื่องจากบุคคลทั่วไปหากมีโปรแกรมก็สามารถถอดความหมายของ QR Code ดังที่มี

งานวิจัยของ Hirakawa M. และ Iijima J. [12] ได้กล่าวว่าผู้ใช้อินเตอร์เน็ตในปัจจุบันมีมากขึ้นกว่าก่อนมาก เนื่องจากเทคโนโลยีที่ก้าวหน้าขึ้นอย่างมาก ทำให้เกิดการคัดลอกได้ง่ายซึ่งมีผลต่อการละเมิดลิขสิทธิ์ตามมา โดยเฉพาะ QR Code ซึ่งเป็นที่สนใจอย่างมาก ณ ขณะนี้ เนื่องจากเทคโนโลยีแบบใหม่ด้าน Data Management ซึ่งมีจุดอ่อนในด้านการรักษาความลับ ทำให้มีงานวิจัยด้านความปลอดภัยหลายด้านตระหนักถึงความปลอดภัยของ QR Code โดยมีการกล่าวถึงการแก้ปัญหาหลากหลายรูปแบบ การใช้ลายน้ำดิจิตอลเข้ามาแก้ไขปัญหาดังกล่าวก็เป็นหนึ่งในหลายๆ ทางเลือกที่งานวิจัยนี้ได้กล่าวถึง

งานวิจัยของ I. Rakocevic และคณะ [13] ซึ่งได้นำเสนอวิธีการในการทำภาพพิมพ์ลายน้ำดิจิตอลที่อ้างอิงกับวิธีการแปลงสัญญาณโคซายน์แบบไม่ต่อเนื่อง (Discrete Cosine Transform, DCT) โดยสัญญาณลายน้ำจะถูกนำไปแทรกลงไปภายในช่วงความถี่กลางของค่าสัมประสิทธิ์ที่ได้จากการแปลง DCT ของภาพต้นฉบับ อย่างไรก็ตามหากภาพที่ใส่ลายน้ำข้างต้นนี้ผ่านกระบวนการบีบอัดข้อมูลด้วยอัตราบีบอัดสูงๆ สัญญาณลายน้ำที่ถูกใส่ไว้ในสัมประสิทธิ์ช่วงความถี่กลางของ DCT ก็จะถูกทำลายไปด้วย

งานวิจัยของ I. Cox และคณะ [33] ซึ่งได้เสนอวิธีการที่จะเลือกตำแหน่งเพื่อใส่สัญญาณลายน้ำ ลงไปในภาพโดยการแก้ไขค่าของสัมประสิทธิ์จากการแปลง DCT ของภาพต้นฉบับ

เช่นเดียวกับวิธีแรก แต่วิธีการนี้จะเลือกเฉพาะตำแหน่งที่สัมประสิทธิ์มีค่าสูงสุดจำนวน 1000 ค่าเท่านั้น ในวิธีการนี้สัญญาณ ลายน้ำจะถูกจัดเตรียมขึ้นด้วยเทคนิคการกระจายแถบความถี่ (Spread Spectrum) ดังนั้นสัญญาณลายน้ำ จะสามารถถูกกู้กลับคืนมาได้อย่างมีประสิทธิภาพ แม้ว่าจะถูกรบกวนจากการประมวลผลทางสัญญาณ ต่างๆ ก็ตาม

งานวิจัยของ K. S. Ng และคณะ [34] เป็นวิธีการซึ่งเรียกว่าการย้ายตำแหน่งพิกเซล (Pixels Position Shifting) เริ่มต้นด้วยการคำนวณค่าผลรวมของค่าพิกเซลทั้ง 64 ค่าจากแต่ละช่วงพื้นที่ของภาพที่มีขนาด 8×8 พิกเซล จากนั้นผลลัพธ์ที่ได้จะถูกหารด้วย 16 เพื่อจะให้ได้ค่าเศษเหลือออกมา ค่าของเศษเหลือที่ได้มานี้จะถูกนำไปใช้เป็นค่าเริ่มต้นเพื่อใส่สัญญาณลายน้ำลงไปในสัมประสิทธ์ DCT ของภาพยิ่งกว่านั้นแล้วรหัสแฮมมิง (Hamming Code) ขนาด (8, 4) ได้ถูกนำมาประยุกต์ใช้เพื่อตรวจสอบและแก้ไขความผิดพลาดที่อาจจะเกิดขึ้นในกระบวนการกู้คืนสัญญาณลายน้ำ อย่างไรก็ตามข้อด้อยของวิธีการนี้ก็คือสัญญาณลายน้ำที่ถูกใส่จะไม่สามารถทนทานต่อการประมวลผลในลักษณะของการกรองแบบความถี่ต่ำผ่าน (Lowpass Filter) และการกรองแบบค่าเฉลี่ยกลาง (Median Filter) ได้

งานวิจัยของ M. George และคณะ [35] ระบบการทำภาพพิมพ์ลายน้ำดิจิตอลที่อ้างอิงมาจากเทคนิคการกระจายแถบความถี่แบบอันดับโดยตรง (Direct Sequence Spread Spectrum Technique) ด้วยวิธีการนี้ข้อมูลลับจะถูกขยายออกด้วยค่าคงที่ค่าหนึ่งเรียกว่า อัตราขยาย (Chip-Rate) ทำให้ได้ข้อมูลที่ถูกกระจาย (Spread Sequence) โดยข้อมูลนี้จะถูกนำไปมอดูเลตกับสัญญาณรบกวนที่ถูกสร้างขึ้นที่เรียกว่า Pseudo-Noise แล้วทำการปรับค่าสัญญาณสุดท้ายนี้ด้วยค่าแฟกเตอร์ α ซึ่งเป็นค่าความแรงของสัญญาณลายน้ำก่อนที่จะนำไปบวกโดยตรงกับภาพต้นฉบับแบบพิกเซลต่อพิกเซล ด้วยวิธีการนี้สัญญาณลายน้ำที่ได้จะมีความคงทนและปลอดภัยเนื่องมาจากคุณสมบัติเฉพาะตัวของเทคนิคการสร้างสัญญาณลายน้ำดังกล่าวจึงทำให้สัญญาณลายน้ำที่ได้จากวิธีการนี้ยากต่อการหาตำแหน่งและการตรวจจับจากผู้ไม่ประสงค์ดี

งานวิจัยของ B. Mobasseri [36] ได้นำเสนอวิธีการใส่สัญญาณลายน้ำดิจิตอลในโดเมนของเวลากับวีดีทัศน์ (Video Sequence) ที่ไม่มีการบีบอัดข้อมูลมาก่อน และด้วยวิธี Direct

Sequence CDMA สัญญาณลายน้ำก็จะถูกกระจายออก และถูกนำไปรวมกับแต่ละเฟรมของวีดีทัศน์ภายใต้การควบคุมของ M-Sequence สองชุดที่แยกกัน ในเวลาต่อมางานของ B. Mobasseri ได้มีการพัฒนามาจนถึงการทำภาพพิมพ์ลายน้ำดิจิตอลกับข้อมูลวีดีทัศน์ที่ถูกบีบอัดโดยมาตรฐาน MPEG

งานวิจัยของ Yeung และ Mintzer [37] ใช้เทคนิคการแทรกลายน้ำในโดเมนพื้นที่ (Spatial Domain) โดยกระบวนการแทรกภาพจะมีรหัสตรวจสอบ (Verification Key) ซึ่งได้จากการสุ่มเทียมตารางเทียบค่าไบนารี เพื่อใช้ตรวจสอบการเปลี่ยนแปลงในกระบวนการถอดลายน้ำ และสามารถระบุตำแหน่งของการปลี่ยนแปลงได้ ซึ่งเทคนิคการแทรกนี้มีประสิทธิภาพและคงทนกว่าการแทรกค่าไบนารี (Binary) ในบิตที่มีความสำคัญน้อยที่สุด (LSB) การใช้รหัสลับในการสุ่มเทียมสร้างตารางเทียบค่าไบนารีทำให้สามารถควบคุมการปรับปรุงค่าความเข้มบนจุดภาพ และยากต่อการปลอมแปลงรายละเอียดเนื้อหาภาพ แต่มีข้อเสียของการแทรกลายน้ำในโดเมนพื้นที่คือ จะมีความคงทนต่อการโจมตีด้วยกระบวนการประมวลผลสัญญาณ (Signal Processing) น้อยกว่าการแทรกลายน้ำโดเมนความถี่

งานวิจัยของ M. Wu และ B. Liu [38] นำเทคนิคการเข้ารหัส JPEG มาใช้ การแทรกบิตลายน้ำนั้นจะทำการปรับปรุงค่าสัมประสิทธิ์ซึ่งผ่านการแปลง DCT และการแทรกลายน้ำผ่านตารางเทียบค่าไบนารีซึ่งสร้างจากการใช้รหัสลับ ขั้นตอนการถอดลายน้ำก็ผ่านกระบวนการเดิม จากนั้นได้ทำการทดสอบปลอมแปลงภาพลายน้ำโดยทำการลบรายละเอียดบางส่วนของภาพทิ้ง แล้วเพิ่มรายละเอียดอื่นลงไปในภาพแทน ผลการทดลองพบว่าลายน้ำที่ถอดออกมาสามารถระบุตำแหน่งปลอมแปลงได้

งานวิจัยของ S. Anahita Mortazavi และ คณะ [39] ได้ใช้ Diffie-Hellman key ซึ่งพัฒนาวิทยาการรหัสลับชนิดกุญแจสาธารณะ (Public-key cryptography) เพื่อแก้ปัญระบบการจัดการกุญแจ (Key management) โดยใช้กุญแจสองดอก โดยกุญแจดอกแรกเรียกว่า กุญแจสาธารณะ (Public key) ส่วนกุญแจดอกสองเรียกว่า กุญแจส่วนตัว (Private key) ใช้ในการถอดรหัสลับข้อความ โดยนำเสนอ Many-to-many group key management

งานวิจัยของ Schyndel และคณะ [40] ได้เสนอวิธีการฝังลงในบิตที่มีความสำคัญน้อยสุด (Least Significant Bit : LSB) ของแต่ละจุดภาพในรูปภาพ โดยอิงสมมุติฐานที่ว่าการเปลี่ยนแปลงที่ตำแหน่ง LSB จะส่งผลกระทบต่อคุณภาพของภาพน้อยที่สุด ข้อเสียของงานวิจัย คือ จะไม่มีความทนทานต่อการโจมตีที่รวมสัญญาณรบกวนกับตัวข้อมูลแล้วเปลี่ยนแปลงค่าของ LSB ให้เป็น 0 หรือ 1 ทั้งหมด

งานวิจัยของ สิริพร ผลสมบูรณ์ [29] ได้นำเสนอเทคนิคการทำลายน้ำดิจิตอลสำหรับรูปภาพเพื่อใช้พิสูจน์ลิขสิทธิ์ความเป็นเจ้าของ (Ownership identification) และตรวจสอบการถูกเปลี่ยนแปลงแก้ไขบนตัวสื่อ (Media authentication) ซึ่งในวิทยานิพนธ์นี้เราจะพิจารณาเฉพาะสื่อรูปภาพเท่านั้น โดยมีคุณสมบัติในด้านความคงทนต่อกระบวนการประมวลผลใดๆ ที่อาจเกิดขึ้นกับรูปภาพ โดยในการฝังลายน้ำใช้การทรานฟอร์ม DFT ข้อเสียของงานวิจัย คือ เมื่อเพิ่มขนาดของภาพให้ใหญ่ขึ้น ส่งผลให้ถอดลายน้ำออกมาได้ไม่ดี

งานวิจัยของ ประทาน โชติพันธุ์บัณฑิตย์ [41] ได้นำเสนอเทคนิคการซ่อนข้อมูลและเพิ่มความมั่นคงสำหรับเอกสารสิ่งพิมพ์ โดยการนำเทคโนโลยีดาต้ากลิฟส์และการเพิ่มสัญลักษณ์ เพื่อเพิ่มความปลอดภัยและป้องกันไม่ให้ผู้ที่ไม่มีสิทธิ์ในการเข้าถึงข้อมูล สามารถเข้าถึงข้อมูลได้ โดยใช้เทคโนโลยีดาต้ากลิฟส์มาแปลงข้อมูลที่ต้องการปกปิดให้อยู่ในรูปแบบที่ไม่สามารถอ่านหรือรับรู้ได้ด้วยสายตาแลเพิ่มสัญลักษณ์ในการเพิ่มความจุข้อมูล ข้อเสียของงานวิจัย คือ ถ้ามีสัญลักษณ์มาก ก็มีโอกาสที่ทำให้การถอดข้อมูลออกมานั้นผิดพลาดได้มากกว่าสัญญลักษณ์น้อย

งานวิจัยของ มหศักดิ์ เกตุฉ่ำ [21] ได้นำเสนอเทคนิคการทำลายน้ำดิจิตอลสำหรับสื่อประเภทเสียงเพื่อใช้พิสูจน์ความเป็นเจ้าของ (Ownership identification) ซึ่งพิจารณาเฉพาะสื่อประเภทเสียงและรูปภาพที่เป็นลายน้ำ และลายน้ำมีคุณสมบัติในด้านความคงทนต่อกระบวนการประมวลผลใด ๆ และไม่ลดทอนคุณภาพของสัญญาณเสียง โดยพัฒนาอัลกอริทึมร่วมกับวิธีเชิงพันธุกรรม (Genetic Algorithms) เพื่อหาตำแหน่งที่ดีที่สุดที่ใช้ในการฝังลายน้ำ

งานวิจัยของ นริศ หนูหอม [42] ได้นำเสนอเทคนิควิธีการฝังและวิธีการถอดลายน้ำอเนกประสงค์บนรูปภาพดิจิตอล เพื่อให้สามารถนำมาใช้ตรวจสอบรับรองความเป็นต้นฉบับและ

พิสูจน์ความเป็นเจ้าของลิขสิทธิ์ได้ในคราวเดียวกัน โดยวิธีการฝังและวิธีการถอดลายน้ำอเนกประสงค์ดังกล่าวจะกระทำบนเอสวีดี (Single value decomposition, SVD)

งานวิจัยของ สุวรรฐ สุตันยสุวรรณ [43] ได้นำเสนอนำเอาเทคโนโลยีรหัสแท่ง (Barcode) และ เทคโนโลยีดาต้ากลิฟส์ (Data glyphs) มาใช้ฝังข้อมูลลงบนสื่อสิ่งพิมพ์ โดยทำการปรับปรุงให้มีความหนาแน่นมากขึ้นและมีความถูกต้องสูงเมื่อทำการอ่านค่าที่บันทึกกลับ ข้อเสียของงานวิจัย คือ การใช้ระดับเฉดสีมีข้อจำกัดจำนวนมาก ทั้งขอบเขตและช่องว่างระหว่างสี ทำให้ไม่สามารถแบ่งแยกสีที่มีเฉดเดียวกันได้มาก

งานวิจัยของ Kundur และ Hatzinakos [44] อธิบายการใช้เทคนิคการแปลงเวฟเล็ต และการทำการแทรกลายน้ำ เน้นที่กระบวนการปรับปรุงค่าสัมประสิทธิ์ ผลการทดลองสามารถระบุตำแหน่งปลอมแปลงได้ โดยแยกองค์ประกอบด้วยวิธีเวฟเล็ต (Wavelet) ซึ่งให้ข้อมูลทั้งโดเมนความถี่ (Frequency) และโดเมนพื้นที่ (Spatial) ทำให้สามารถตรวจสอบสัญญาณที่ถูกเปลี่ยนแปลงได้

งานวิจัยของ Ruanaidh และ คณะ [45] นำเสนอวิธี DFT โดยการปรับแต่งเฟส (Phase) การทำลายน้ำโดยการปรับแต่งเฟสนั้นทำให้ลายน้ำทนทานต่อความเสียหายที่เกิดจากการปรับความเข้มจากนั้นแสดงให้เห็นวิธีการใช้การแปลงแบบ Fourier Mellin ในการทำลายน้ำดิจิตอล

งานวิจัยของ Li Li และ คณะ [46] นำเสนอวิธี DCT โดยมุ่งเน้นการใส่ Noise ที่แตกต่างกัน ได้แก่ Fuzzy, Gaussian, Random และ Salt and pepper แล้วพิจารณาสัมประสิทธิ์ในการฝังลายน้ำ โดยในการทดลองได้กล่าวว่าจากการทดลองการนำ 2D Barcode มาประยุกต์ใช้นั้นเป็นไปได้อย่างมีประสิทธิภาพ

บทที่ 3
วิธีดำเนินการวิจัย

ในบทนี้จะกล่าวถึงการดำเนินงานวิจัยซึ่งประกอบด้วย การเข้ารหัสข้อมูล QR Code (จากหัวข้อ 2.2), นำข้อมูล QR Code เข้าสู่กระบวนการฝังผ่านการทรานฟอร์ม 3 แบบ ได้แก่ DCT, DFT และ DWT (จากหัวข้อ 2.6), นำข้อมูล QR Code ที่ผ่านการฝังเข้าสู่กระบวนการถอดลายน้ำดิจิตอล (จากหัวข้อ 2.6), การถอดรหัสข้อมูล QR Code (จากหัวข้อ 2.3) และการวัดประสิทธิภาพของลายน้ำดิจิตอล (จากหัวข้อ 2.5.7) โดยมีแผนภาพรวมเป็นดังภาพที่ 3-1 โดยแบ่งออกไป 3 ส่วนใหญ่ๆ ดังต่อไปนี้

3.1 ภาพรวมทั้งหมดของงานวิจัย

3.1.1 การเข้ารหัสข้อมูล QR Code

3.1.1.1 Input Data คือ เริ่มต้นด้วยกำหนดข้อความที่เราใช้เข้ารหัส QR Code

3.1.1.2 Encode QR Code คือ เป็นการนำข้อความ (ผลลัพธ์ 3.1.1.1) ผ่านกระบวนการเข้ารหัส QR Code

3.1.1.3 QR Code Original คือ ข้อมูลที่ได้หลังจากผ่านการเข้ารหัส QR Code (ผลลัพธ์จาก 3.1.1.2)

3.1.2 กระบวนการฝังลายน้ำดิจิตอล

3.1.2.1 Embedding (DCT/DFT/DWT) คือ ทำการเลือกการทรานฟอร์มเพื่อแปลง (ผลลัพธ์จาก 3.1.1.3) ให้อยู่ในโดเมนความถี่ การเลือกการทรานฟอร์มในภาพ 3-1 มีทั้งหมด 3 แบบ ได้แก่ DCT, DFT และ DWT เพื่อเข้าสู่กระบวนการฝังลายน้ำดิจิตอลใน โดยจะกระทำผ่าน Look-up table และ Key ได้ Watermarked QR Code

3.1.2.2 Calculate PSNR and Time คือ คำนวณหา PSNR และ Time จาก

Watermarked QR Code (ที่ได้จาก 3.1.2.1)

3.1.2.3 Result คือ ค่าที่ได้จากการคำนวณหา PSNR และ Time (ที่ได้จาก 3.1.2.2)

3.1.3 กระบวนการถอดลายน้ำดิจิตอล

3.1.3.1 Extracting (DCT/DFT/DWT) คือ ทำการเลือกการทรานฟอร์มในการแปลงกลับเพื่อมาผ่านกระบวนการถอดลายน้ำดิจิตอล โดยผ่าน Look-up table และ Key

3.1.3.2 Watermark คือ ทำให้เราได้ลายน้ำดิจิตอลจากการทรานฟอร์มในการแปลงกลับ (ผลลัพธ์จาก 3.1.3.1)

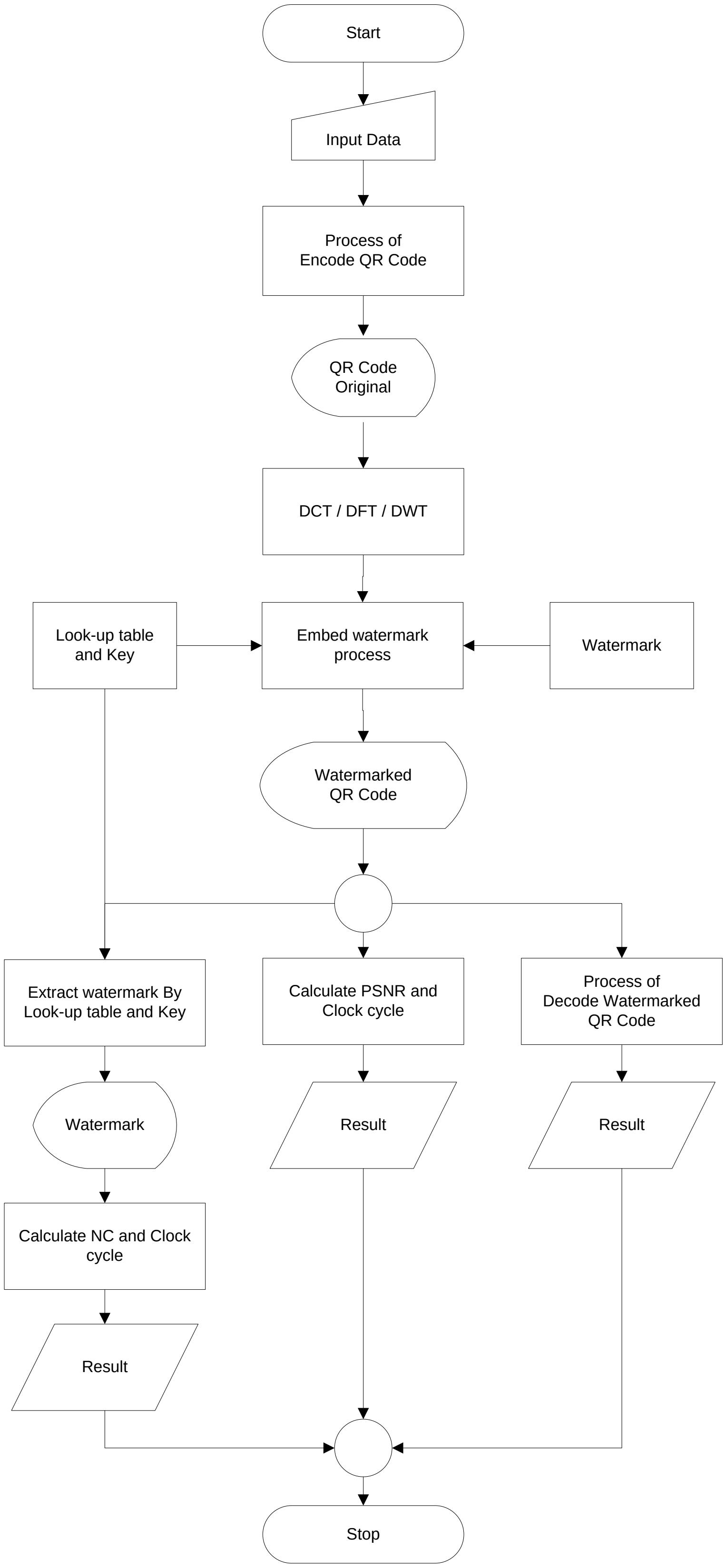
3.1.3.3 Calculate NC and Time คือ คำนวณหา NC และ Time จากลายน้ำดิจิตอล (ผลลัพธ์ที่ได้จาก 3.1.3.2)

3.1.3.4 Result คือ ค่าที่ได้จากการคำนวณหา NC และ Time (ที่ได้จาก 3.1.3.3)

3.1.4 การถอดรหัสข้อมูล QR Code

3.1.4.1 Decode Watermarked QR Code คือ การถอดรหัสข้อมูล Watermarked QR Code (ที่ได้จาก 3.1.2.1)

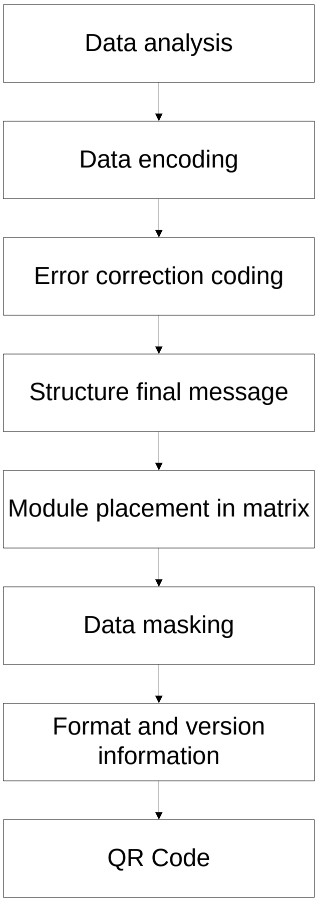
3.1.4.2 Result คือ ผลลัพธ์ที่ได้จากการถอดรหัสข้อมูล (ที่ได้จาก 3.1.4.1)



ภาพที่ 3-1 โครงสร้างทั้งหมดของงานวิจัย

3.2 **การเข้ารหัสข้อมูล** QR Code (Encode QR Code Procedure)

ในงานวิจัยนี้ เราสนใจการเข้ารหัสการแปลงจากข้อมูลชนิด Numeric และ Alphanumeric เท่านั้น ส่วนในการคำนวณ Error Correction Level เราสนใจแค่ระดับ L เท่านั้น เนื่องจากงานวิจัยมุ่งเน้นการหาวิธีแก้ไขปัญหา QR Code ไม่สามารถซ่อนข้อมูลได้ เราจึงแก้ปัญหาโดยการนำลายน้ำมาใช้ ในรายละเอียดการเข้ารหัส QR Code ได้กล่าวไว้ในหัวข้อ 2.2 โดยมีขั้นตอนการคิดในการเข้ารหัส ดังภาพที่ 3-2



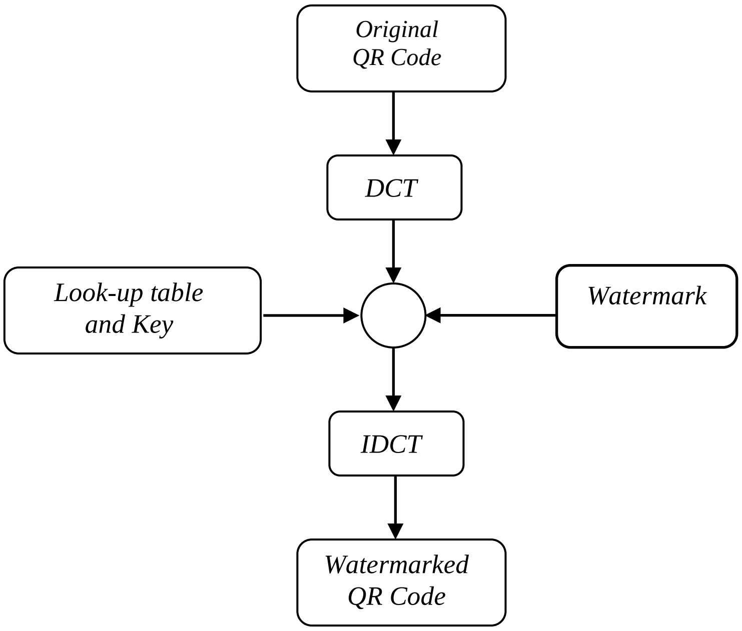
ภาพที่ 3-2 กระบวนการเข้ารหัสของ QR Code

3.3 **กระบวนการฝังลายน้ำดิจิตอล** (Embed Watermark in QR Code Procedure)

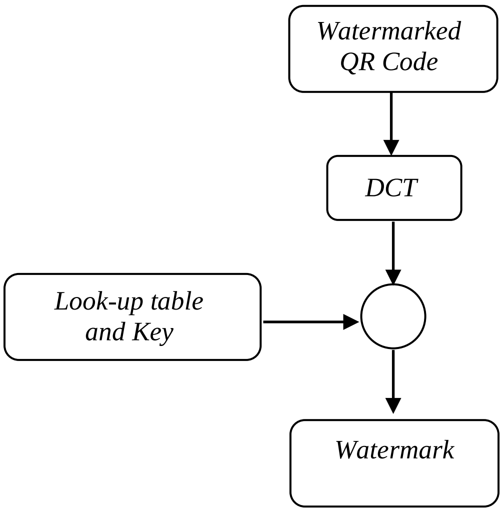
ในงานวิจัยนี้ เราใช้การฝังลายน้ำดิจิตอลรวมเข้ากับข้อมูล QR Code เพื่อแก้ปัญหา ซึ่งในกระบวนการฝังลายน้ำดิจิตอล เราต้องการดูรูปแบบการทรานฟอร์ม 3 แบบ ซึ่งได้แก่ DCT, DFT และ DWT ซึ่งมีขั้นตอนการฝังลายน้ำดิจิตอล ดังต่อไปนี้

3.3.1 กระบวนการฝังลายน้ำดิจิตอลด้วย DCT

แผนภาพกระบวนการฝังลายน้ำดิจิตอลด้วย DCT แสดงดังภาพที่ 3-3 และ 3-4

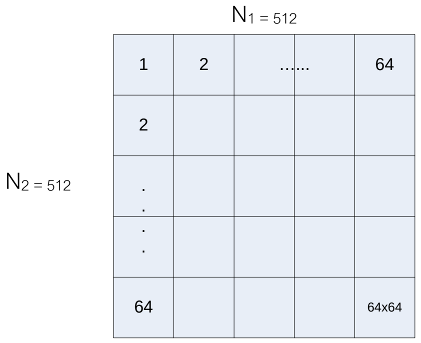


ภาพที่ 3-3 โครงร่างของเทคนิคการทำลายน้ำดิจิตอลแบบ DCT ในกระบวนการฝังลายน้ำดิจิตอล



ภาพที่ 3-4 โครงร่างของเทคนิคการทำลายน้ำดิจิตอลแบบ DCT ในกระบวนการถอดลายน้ำดิจิตอล

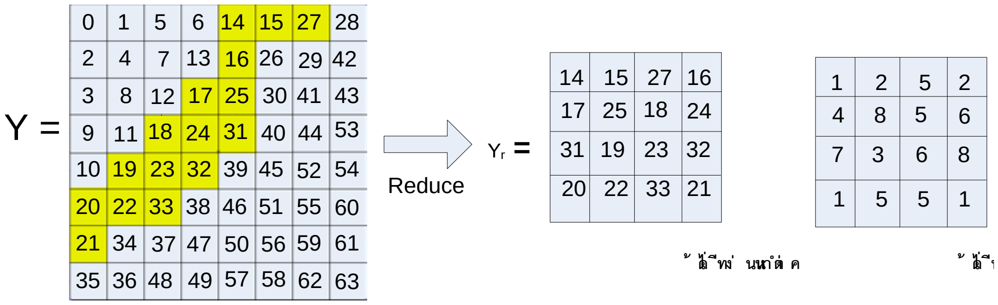
เมื่อแบ่งข้อมูล QR Code ต้นฉบับออกเป็นบริเวณแล้ว จำนวนบริเวณทั้งหมดต้องสัมพันธ์กับข้อมูล QR Code ต้นฉบับ คือ $\frac{N_1}{8} \times \frac{N_2}{8}$ บริเวณ เช่น ถ้า $N_1 = N_2 = 512$ จุดภาพ จึงทำให้ได้บริเวณในข้อมูล QR Code ต้นฉบับเป็น 64×64 บริเวณ ดังภาพที่ 3-5



ภาพที่ 3-5 การแบ่งข้อมูล QR Code ต้นฉบับออกเป็นบริเวณไม่ซ้อนทับกัน

ภาพลายน้ำดิจิตอลก็แบ่งในลักษณะเดียวกัน โดยขนาดแต่ละบริเวณบนภาพลายน้ำดิจิตอลนั้น มีความสัมพันธ์กับข้อมูล QR Code ต้นฉบับและภาพลายน้ำดิจิตอล คือ สามารถแบ่งบริเวณภาพลายน้ำดิจิตอลได้ $\left(M_1 \times \frac{8}{N_1}\right) \times \left(M_2 \times \frac{8}{N_2}\right)$ บริเวณ ในงานวิจัยถ้าข้อมูล QR Code ต้นฉบับขนาด 512×512 จุดภาพ แทนค่าลงในความสัมพันธ์ จะได้ขนาดของบริเวณแต่ละบริเวณบนภาพลายน้ำดิจิตอล คือ $\left(64 \times \frac{8}{512}\right) \times \left(64 \times \frac{8}{512}\right) = 1 \times 1$ จุดภาพ เมื่อทำการแบ่งภาพข้อมูล QR Code ต้นฉบับเป็นบริเวณไม่ซ้อนทับกัน โดยแต่ละบริเวณมีขนาด 8×8 จุดภาพ จะทำการแปลงข้อมูล QR Code ต้นฉบับที่อยู่ในรูปของสัญญาณดิจิตอลซึ่งอยู่ในโดเมนเวลา (Time Domain) ไปเป็นสัญญาณที่อยู่ในโดเมนความถี่ (Frequency Domain) การแปลงข้อมูล QR Code ต้นฉบับใช้การแปลงแบบ DCT แล้วค่าความถี่ที่ได้ในแต่ละจุดภาพจะอยู่ในรูปของจำนวนจริงซึ่งสามารถนำค่าเหล่านี้ไปทำการเปลี่ยนแปลงแก้ไข ซึ่งการแปลงแบบ DCT มีรายละเอียดอยู่ในหัวข้อ 2.6.1 โดยในการเลือกค่าสัมประสิทธิ์จะเลือกที่อยู่ในความถี่ช่วงกลางดังภาพที่ 2-31 เนื่องจากเมื่อทำการฝังภาพลายน้ำแล้วภาพผลลัพธ์จะเกิดการเปลี่ยนแปลงน้อยสุด โดยทำการเลือกค่าสัมประสิทธิ์ความถี่ช่วงกลางด้วยวิธีจัดเรียงแบบ Zigzag และจะต้องเก็บตำแหน่งที่เลือกไว้ด้วย แล้วนำมาจัดเรียงใหม่ซึ่งจะได้ค่า

สัมประสิทธิ์ที่มีขนาด 4×4 จุดภาพ ดังภาพที่ 3-5 โดยตำแหน่งของจุดภาพที่ทำการเลือกนั้นจะเหมือนกันทุกบริเวณและทำเช่นนี้ไปจนครบทุกบริเวณ



ภาพที่ 3-6 การเลือกค่าสัมประสิทธิ์ของความถี่กลางด้วยวิธี Zigzag

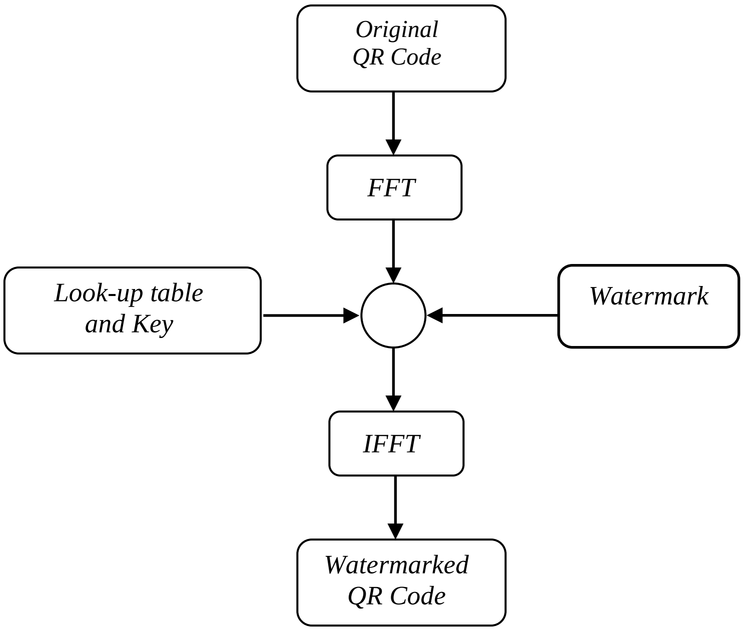
โดยในงานวิจัยใช้เทคนิคการฝัง PN Sequence ไปยังความถี่กลางของ DCT Block ที่ได้ ซึ่งก็คือค่าสัมประสิทธิ์ที่ได้จากภาพที่ 3-6 โดยเราสามารถคำนวณได้โดยใช้สมการต่อไปนี้

$$I_{W\,x,y}(u,v) = \begin{cases} I_{x,y}(u,v) + k * W_{x,y}(u,v), & u,v \in F_M \\ Ix,y(u,v), & u,v \notin F_M \end{cases} \qquad (3\text{-}1)$$

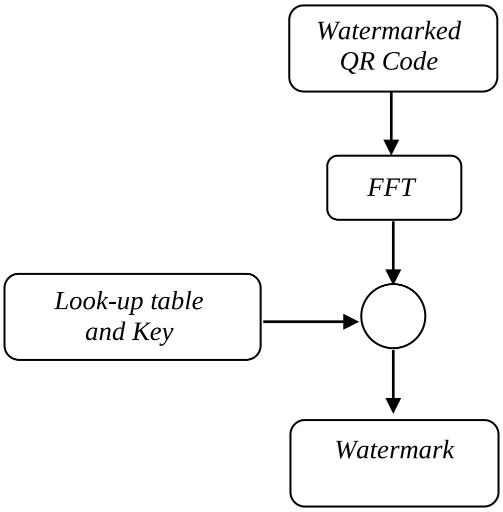
จากสมการที่ 3-1 เป็นการฝัง CDMA Watermark ไปยังความถี่กลางของ DCT โดยการนำค่าสัมประสิทธิ์ที่ได้จากภาพที่ 3-6 เพิ่มค่า PN Sequence เข้าไป จากสมการที่ 3-1 นั้น $I_{W\,x,y}(u,v)$ เป็นค่าสัมประสิทธิ์ที่ได้จากภาพที่ 3-6 ถ้าอยู่ช่วงความถี่กลางให้เพิ่มค่า PN Sequence ที่กระทำการคูณกับค่า Gain factor (k) ลงไป ซึ่งจะกระทำผ่าน Look-up table และ Key ซึ่งจะกล่าวถึงในหัวข้อ 3.3.4 หลังจากนั้นให้ทำการทรานฟอร์มกลับโดยใช้ IDCT ซึ่งมีรายละเอียดอยู่ในหัวข้อ 2.6.1

3.3.2 กระบวนการฝังลายน้ำดิจิตอลด้วย DFT

แผนภาพกระบวนการฝังลายน้ำดิจิตอลด้วย DFT แสดงดังภาพที่ 3-7 และ 3-8

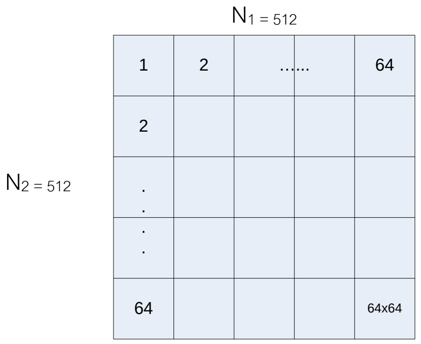


ภาพที่ 3-7 โครงร่างของเทคนิคการทำลายน้ำดิจิตอลแบบ DFT ในกระบวนการฝังลายน้ำดิจิตอล



ภาพที่ 3-8 โครงร่างของเทคนิคการทำลายน้ำดิจิตอลแบบ DFT ในกระบวนการถอดลายน้ำดิจิตอล

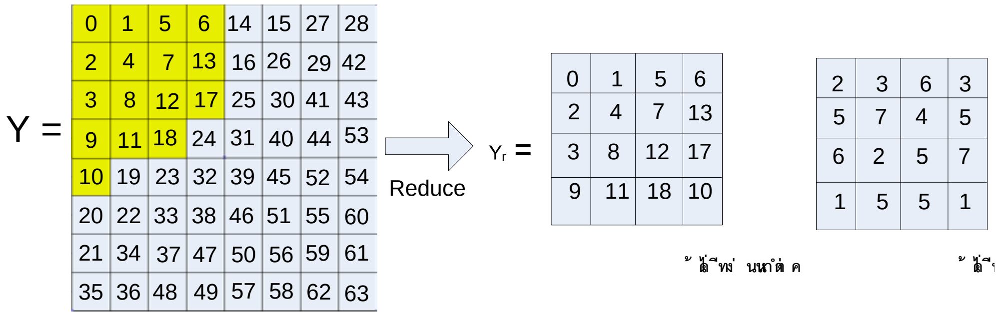
โดยในการแปลง DFT เพื่อทำการฝังลายน้ำจะมีหลักการคล้ายกับการแปลง DCT ซึ่งมีรายละเอียดดังต่อไปนี้ เมื่อแบ่งข้อมูล QR Code ต้นฉบับออกเป็นบริเวณแล้ว จำนวนบริเวณทั้งหมดต้องสัมพันธ์กับข้อมูล QR Code ต้นฉบับ คือ $\frac{N_1}{8} \times \frac{N_2}{8}$ บริเวณ เช่น ถ้า $N_1 = N_2 = 512$ จุดภาพ จึงทำให้ได้บริเวณในข้อมูล QR Code ต้นฉบับเป็น 64×64 บริเวณ ดังภาพที่ 3-9



ภาพที่ 3-9 การแบ่งข้อมูล QR Code ต้นฉบับออกเป็นบริเวณไม่ซ้อนทับกัน

ภาพลายน้ำดิจิตอลก็แบ่งในลักษณะเดียวกัน โดยขนาดแต่ละบริเวณบนภาพลายน้ำดิจิตอลนั้น มีความสัมพันธ์กับข้อมูล QR Code ต้นฉบับและภาพลายน้ำดิจิตอล คือ สามารถแบ่งบริเวณภาพลายน้ำดิจิตอลได้ $\left(M_1 \times \frac{8}{N_1}\right) \times \left(M_2 \times \frac{8}{N_2}\right)$ บริเวณ ในงานวิจัยถ้าข้อมูล QR Code ต้นฉบับขนาด 512×512 จุดภาพ แทนค่าลงในความสัมพันธ์ จะได้ขนาดของบริเวณแต่ละบริเวณบนภาพลายน้ำดิจิตอล คือ $\left(64 \times \frac{8}{512}\right) \times \left(64 \times \frac{8}{512}\right) = 1 \times 1$ จุดภาพ เมื่อทำการแบ่งภาพข้อมูล QR Code ต้นฉบับเป็นบริเวณไม่ซ้อนทับกัน โดยแต่ละบริเวณมีขนาด 8×8 จุดภาพ จะทำการแปลงข้อมูล QR Code ต้นฉบับที่อยู่ในรูปของสัญญาณดิจิตอลซึ่งอยู่ในโดเมนเวลา (Time Domain) ไปเป็นสัญญาณที่อยู่ในโดเมนความถี่ (Frequency Domain) การแปลงข้อมูล QR Code ต้นฉบับใช้การแปลงแบบ DFT แล้วค่าความถี่ที่ได้ในแต่ละจุดภาพจะอยู่ในรูปของจำนวนจริงซึ่งสามารถนำค่าเหล่านี้ไปทำการเปลี่ยนแปลงแก้ไข ซึ่งการแปลงแบบ DFT มีรายละเอียดอยู่ในหัวข้อ 2.6.1 โดยในการเลือกค่าสัมประสิทธิ์จะเลือกที่อยู่ในความถี่ช่วงต่ำดังภาพที่ 2-32 เนื่องจากเมื่อทำการฝังภาพลายน้ำแล้วภาพผลลัพธ์จะเกิดการเปลี่ยนแปลงน้อยสุด โดยทำการเลือกค่าสัมประสิทธิ์ความถี่ช่วงต่ำด้วยวิธีจัดเรียงแบบ Zigzag และจะต้องเก็บตำแหน่งที่เลือกไว้ด้วย แล้วนำมาจัดเรียงใหม่ซึ่งจะได้ค่า

สัมประสิทธิ์ที่มีขนาด 4×4 จุดภาพ ดังภาพที่ 3-10 โดยตำแหน่งของจุดภาพที่ทำการเลือกนั้นจะเหมือนกันทุกบริเวณและทำเช่นนี้ไปจนครบทุกบริเวณ



ภาพที่ 3-10 การเลือกค่าสัมประสิทธิ์ของความถี่กลางด้วยวิธี Zigzag

โดยในงานวิจัยใช้เทคนิคการฝัง PN Sequence ไปยังความถี่ต่ำของ DFT Block ที่ได้ ซึ่งก็คือค่าสัมประสิทธิ์ที่ได้จากตารางภาพที่ 3-8 โดยเราสามารถคำนวณได้โดยใช้สมการต่อไปนี้

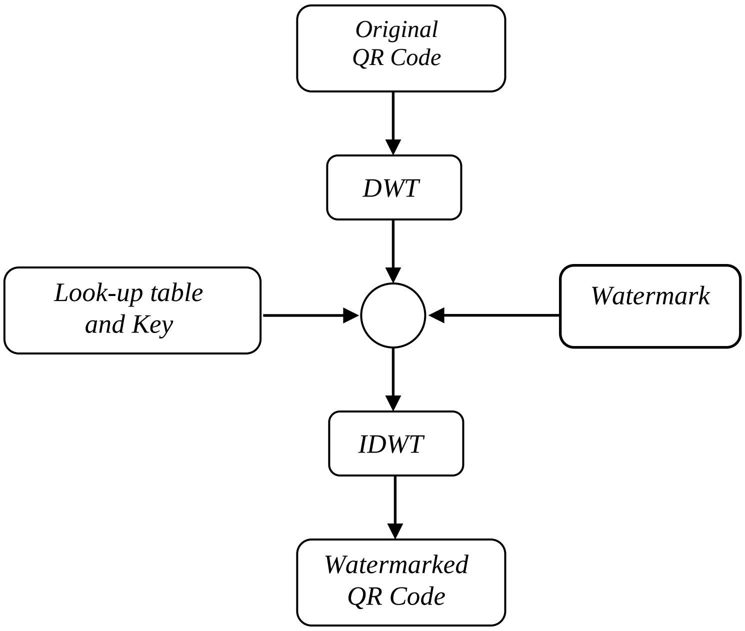
$$I_{W\,x,y}(u,v) = \begin{cases} I_{x,y}(u,v) + k * W_{x,y}(u,v), & u,v \in F_L \\ Ix, y(u,v), & u,v \notin F_L \end{cases} \quad (3\text{-}2)$$

จากสมการที่ 3-2 เป็นการฝัง CDMA Watermark ไปยังความถี่ต่ำของ DFT โดยการนำค่าสัมประสิทธิ์ที่ได้จากภาพที่ 3-8 เพิ่มค่า PN Sequence เข้าไป จากสมการที่ 3-2 นั้น $I_{W\,x,y}(u,v)$ เป็นค่าสัมประสิทธิ์ที่ได้จากภาพที่ 3-8 ถ้าอยู่ช่วงความถี่ต่ำให้เพิ่มค่า PN Sequence ที่กระทำการคูณกับค่า Gain factor (k) ลงไป ซึ่งจะกระทำผ่าน Look-up table และ Key ซึ่งจะกล่าวถึงในหัวข้อ 3.3.4 หลังจากนั้นให้ทำการทรานฟอร์มกลับ โดยใช้ IDCT ซึ่งมีรายละเอียดอยู่ในหัวข้อ 2.6.1

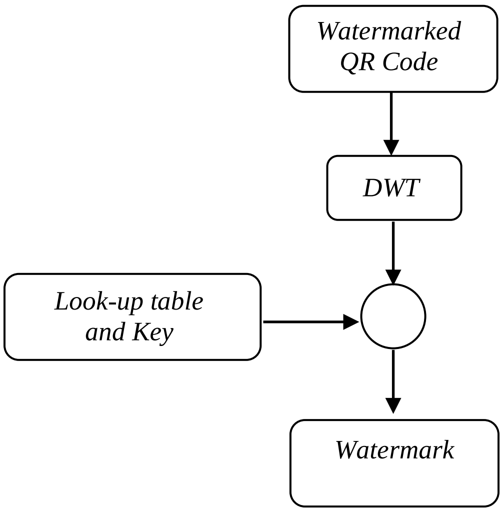
3.3.3 กระบวนการฝังลายน้ำดิจิตอลด้วย DWT

การฝังลายน้ำดิจิตอลด้วยการทรานฟอร์มวิธีนี้ จะมีหลักการแตกต่างจาก 2 วิธีแรก ที่กระทำบนข้อมูล QR Code ต้นฉบับทั้งหมด ซึ่งได้มีการพัฒนามาจากการแปลงสัญญาณพื้นฐานที่มีอยู่เดิม โดยวิธีนี้ผลการวิเคราะห์จะออกมาในรูปของการแตก

องค์ประกอบของสัญญาณใดๆ ออกมา ดังในภาพที่ 2-32 ที่มีแผนภาพกระบวนการฝังลายน้ำดิจิตอลด้วย DWT แสดงดังภาพที่ 3-11 และ 3-12



ภาพที่ 3-11 โครงร่างของเทคนิคการทำลายน้ำดิจิตอลแบบ DWT ในกระบวนการฝังลายน้ำดิจิตอล

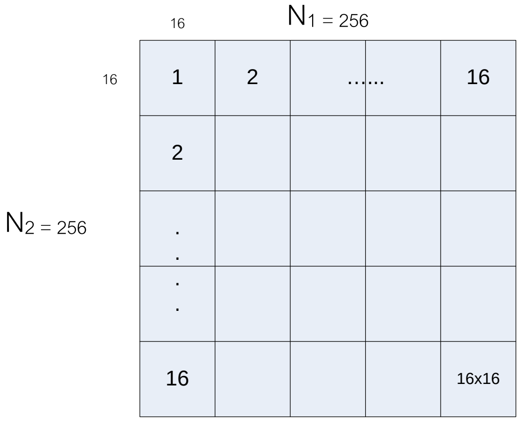


ภาพที่ 3-12 โครงร่างของเทคนิคการทำลายน้ำดิจิตอลแบบ DWT ในกระบวนการถอดลายน้ำดิจิตอล

กำหนดให้ X แทนข้อมูล QR Code ต้นฉบับ ขนาด $N_1 \times N_2$ จุดภาพ ข้อมูล QR Code ต้นฉบับ สามารถแทนด้วย สมการที่ (3-3)

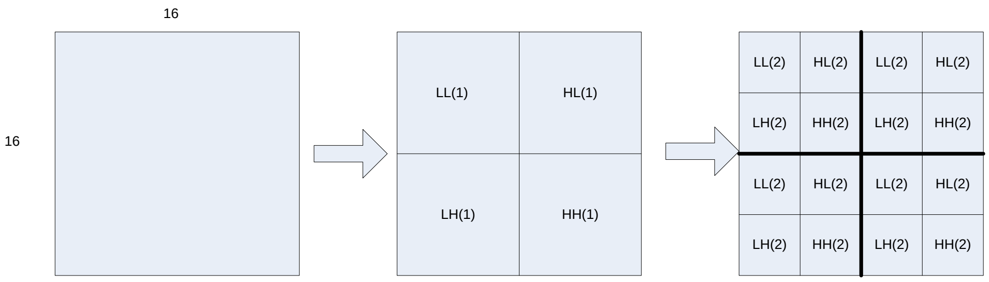
$$X = \{X(i,j); 0 \leq i < N_1, 0 \leq j < N_2\} \tag{3-3}$$

เมื่อ $X(i,j)$ คือความเข้มของจุดภาพที่ตำแหน่ง(i,j) ในแต่ละจุดภาพ จากนั้นทำการแบ่งข้อมูล QR Code ต้นฉบับออกเป็นบริเวณ (Block) ขนาด 16×16 จุดภาพ จำนวนบริเวณทั้งหมดที่ได้มีความสัมพันธ์กับข้อมูล QR Code ต้นฉบับคือ $\frac{N_1}{16} \times \frac{N_2}{16}$ บริเวณ เช่น ถ้า $N_1 = N_2 = 256$ จุดภาพ จะทำให้ได้จำนวนบริเวณในข้อมูล QR Code ต้นฉบับดังกล่าวเป็น 16×16 บริเวณ ดังภาพที่ 3-13



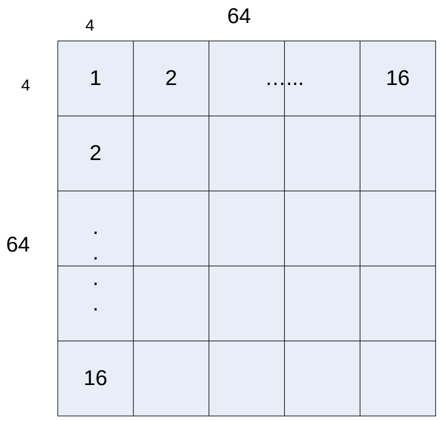
ภาพที่ 3-13 การแบ่งข้อมูล QR Code ต้นฉบับออกเป็นบริเวณละ 16×16 จุดภาพ

ข้อมูล QR Code ต้นฉบับขนาดบริเวณละ 16×16 จุดภาพ จะถูกแปลงจากโดเมนปกติให้อยู่ในโดเมนความถี่ด้วยวิธีการแปลงเวฟเล็ตแพกเกตส์ โดยค่าระดับในการแปลงข้อมูล QR Code ต้นฉบับเมื่อผ่านการแปลงแล้ว จะมีขนาดเท่ากับลายน้ำดิจิตอล เช่น ลายน้ำดิจิตอลที่ถูกแบ่งให้มีขนาด 4×4 จุดภาพ และข้อมูล QR Code ต้นฉบับขนาด 16×16 จุดภาพนั้น แสดงว่าระดับการแปลงข้อมูล QR Code ต้นฉบับโดยใช้การแปลงเวฟเล็ตแบบฮาร์ (Haar Wavelet) จะมีค่าระดับเป็น 2 จึงจะทำให้ข้อมูล QR Code ต้นฉบับที่จะเลือกแทรกลายน้ำมีขนาดเท่ากับขนาดของลายน้ำดิจิตอล คือ 4×4 จุดภาพ ดังภาพที่ 3-14



ภาพที่ 3-14 เวฟเล็ตแบบไม่ต่อเนื่องระดับที่ 1 และ 2 ของข้อมูล QR Code ต้นฉบับ

การแบ่งพื้นที่ภาพลายน้ำดิจิตอลจะทำการแบ่งในลักษณะเดียวกัน โดยแต่ละบริเวณบนภาพลายน้ำนั้นมีความสัมพันธ์กับข้อมูล QR Code ต้นฉบับและภาพลายน้ำคือ $\left(M_i \times \frac{16}{N_1}\right) \times \left(M_j \times \frac{16}{N_2}\right)$ ซึ่งในที่นี้จะได้บริเวณบนภาพลายน้ำที่มีขนาด 4×4 จุดภาพ ดังภาพที่ 3-15



ภาพที่ 3-15 การแบ่งภาพลายน้ำดิจิตอลเป็นบริเวณขนาด 4×4 จุดภาพ

เนื่องจากการแทรกลายน้ำดิจิตอลลงในข้อมูล QR Code ต้นฉบับจะกระทำเป็นบริเวณต่อบริเวณ ดังนั้นจำนวนบริเวณที่ได้จากการแบ่งข้อมูล QR Code ต้นฉบับกับการแบ่งลายน้ำดิจิตอลจะต้องเป็น 16×16 บริเวณเท่ากัน การแปลงเทคนิคนี้ใช้ DWT ในการทรานฟอร์ม และทำการแปลงกลับเวฟเล็ต IDWT ทำจนครบทุกบริเวณของภาพ จากนั้นทำบริเวณทั้งหมดรวมกลับคืนจะได้ภาพที่มีลายน้ำแฝงอยู่

3.3.4 ตารางเทียบค่าไบนารี (Look-up table)

การแทรกลายน้ำดิจิตอลบนข้อมูล QR Code ต้นฉบับจะกระทำผ่านตารางเทียบค่าไบนารี ซึ่งมีรหัสลับที่ใช้ในการสร้างตารางเทียบค่าไบนารี โดยรหัสลับที่ใช้สร้างตารางนี้จะต้องเป็นรหัสลับเดียวกันทั้งตอนแทรกและถอดลายน้ำดิจิตอล จึงจะได้ลายน้ำดิจิตอลกลับคืนมา ตารางเทียบค่าไบนารี คือ ตารางที่เป็นไปได้ทั้งหมดของสัมประสิทธิ์จากการแปลงเป็นโดเมนความถี่เทียบกับค่าไบนารีที่ได้จากการสุ่ม การแทรกบิตลายน้ำลงในสัมประสิทธิ์ภาพต้นฉบับ ณ ตำแหน่งใดๆ หากค่าบิตลายน้ำนั้นมีค่าเท่ากับค่าไบนารี ณ ตำแหน่งสัมประสิทธิ์ค่าเดียวกัน ไม่ต้องทำการเปลี่ยนค่าสัมประสิทธิ์นั้น ๆ ให้ใช้ค่าสัมประสิทธิ์เดิม แต่หากบิตลายน้ำที่แทรกมีค่าไม่ตรงกับค่าไบนารี ณ ตำแหน่งนั้น ให้หาค่าสัมประสิทธิ์ใหม่ข้างเคียงที่มีค่าไบนารีตรงกับค่าบิตลายน้ำนั้น แล้วเปลี่ยนค่าสัมประสิทธิ์เดิมเป็นค่าสัมประสิทธิ์ใหม่แทนที่ เช่น การแทรกบิตลายน้ำค่า 1 ลงในสัมประสิทธิ์ของข้อมูล QR Code ต้นฉบับ สัมประสิทธิ์ของข้อมูล QR Code ต้นฉบับจะไม่ถูกเปลี่ยนแปลง ถ้าค่าไบนารี ณ ตำแหน่งสัมประสิทธิ์เดียวกันในตารางเทียบค่าไบนารีมีค่าเป็น 1 เหมือนกัน แต่ถ้าค่าไบนารี ณ ตำแหน่งสัมประสิทธิ์นั้นในตารางเทียบค่าดังกล่าว มีค่าเป็น 0 สัมประสิทธิ์ของข้อมูล QR Code ต้นฉบับจะถูกเปลี่ยนเป็นค่าสัมประสิทธิ์ข้างเคียงที่มีค่าไบนารีเป็น 1 แทน การแทรกบิตลายน้ำค่า 0 ก็ใช้วิธีเดียวกัน

$$x_{\mathrm{i}}' = \begin{cases} x_i & \text{ถ้า } LUT\ (x_i) = w_i \\ x_i + \delta & \text{ถ้า } LUT\ (x_i) \neq w_i \text{ และ} \\ \delta = min_{|d|}\ \{d = x_i' - x_i\ : Lut\ (x_i') = w_i\} \end{cases} \quad (3\text{-}4)$$

โดยที่ x_i เป็นสัมประสิทธิ์ของข้อมูล QR Code ต้นฉบับ

x_i' เป็นสัมประสิทธิ์ที่ถูกใส่ลายน้ำ

w_i' บิตลายน้ำดิจิตอลที่จะแทรกในข้อมูล QR Code ต้นฉบับ

$LUT(\cdot)$ การแทรกผ่านตารางเทียบค่าไบนารี

ตัวอย่างการแทรกบิตลายน้ำแสดงดังภาพที่ 3-13 เราจะทำการแทรกค่าบิตลายน้ำ ณ ตำแหน่งแถวที่ 1 คอลัมน์ที่ 1 ซึ่งมีค่าเท่ากับ 1 ลงในสัมประสิทธิ์ของข้อมูล QR Code ต้นฉบับ ณ ตำแหน่งที่ตรงกันซึ่งมีค่าเท่ากับ -4 แต่ค่าไบนารีซึ่งตรงกับค่า ณ ตำแหน่ง -4 ของตารางเทียบค่ามีค่าเป็น 0 ซึ่งต้องตรวจดูทั้งสองด้านซ้ายขวาของตาราง จึงแทนที่สัมประสิทธิ์ค่า -4 ของข้อมูล QR Code ต้นฉบับดังกล่าวด้วยค่าใหม่ คือ -2 กรณีแทรกแทรกบิตลายน้ำค่า 1 ตัวถัดมาลงมาในสัมประสิทธิ์ข้อมูล QR Code ต้นฉบับซึ่งมีค่าเป็น 0 และค่าไบนารี ณ ตำแหน่งดังกล่าวในตารางเทียบค่ามีค่าเป็น 1 เหมือนกัน ได้คงสัมประสิทธิ์ค่า 0 เดิมไว้ไม่ต้องเปลี่ยนแปลง กรณีการแทรกบิตลายน้ำค่า 0 ก็กระทำเช่นเดียวกัน

1	1	1	1
1	0	0	1
1	1	0	0
0	0	1	0

-4	0	7	1
1	-2	4	0
-7	1	8	-5
6	0	0	-1

=

-2	0	7	0
0	-3	4	0
-7	0	8	-5
4	1	0	-3

Look-up Table

-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8
0	1	0	0	0	0	1	1	1	0	1	1	0	1	1	1	0

ภาพที่ 3-16 แทรกบิตลายน้ำลงในสัมประสิทธิ์ข้อมูล QR Code ต้นฉบับผ่านตารางเทียบค่าไบนารี

3.4 **การถอดลายน้ำดิจิตอล** (Extract Watermark in QR Code Procedure Overview)

การถอดลายน้ำดิจิตอลออกจากภาพลายน้ำ ใช้กระบวนการเดียวกับการแทรกลายน้ำดิจิตอล ลายน้ำดิจิตอลสามารถถอดได้โดยตรงจากภาพลายน้ำ โดยไม่ต้องใช้ภาพต้นฉบับผ่าน Look-up table เริ่มจากนำภาพลายน้ำที่ต้องการทดสอบหาลายน้ำดิจิตอล มาทำการแบ่งพื้นที่ภาพออกเป็นบริเวณจุดภาพตามที่ทำก่อนหน้า ทำการแปลงโดเมนของภาพลายน้ำจากโดเมนปกติ

เป็นโดเมนความถี่ด้วยวิธีการแปลง DCT, DFT และ DWT เลือกบริเวณที่ใช้แทรกลายน้ำ โดยใช้รหัสลับที่ตรงกันในการเลือกบริเวณของภาพที่มีลายน้ำดิจิตอลแทรกอยู่ ทำการปรับปรุงค่าสัมประสิทธิ์ในบริเวณนั้นๆ สร้างตารางเทียบค่าไบนารีจากรหัสลับที่ใช้ในขั้นตอนการแทรกลายน้ำ ซึ่งจะเป็นตารางเดียวกันทั้งกระบวนการแทรกและถอดลายน้ำ จากนั้นทำการถอดบิตลายน้ำจากตารางเทียบค่าซึ่งเป็นค่าบิตไบนารี ณ ตำแหน่งที่สัมประสิทธิ์ในตารางเทียบค่า มีค่าตรงกับค่าสัมประสิทธิ์ในบริเวณที่เลือกแทรกลายน้ำ การถอดลายน้ำแสดงได้ดังสมการที่ 3-5

$$w_i' = LUT(x_i') \tag{3-5}$$

โดยที่ x_i' เป็นสัมประสิทธิ์ของภาพทดสอบ

w_i' เป็นลายน้ำดิจิตอลที่ถอดออกมาได้

ตัวอย่างการถอดบิตลายน้ำจากสัมประสิทธิ์ภาพทดสอบแสดงดังภาพที่ 3-17 ถอดบิตลายน้ำจากภาพทดสอบสัมประสิทธิ์ตัวแรกแถวที่ 1 คอลัมน์ที่ 1 มีค่าเท่ากับ -2 ตารางเทียบค่า ณ ตำแหน่งค่า -2 มีค่าไบนารีเป็น 1 ดังนั้นบิตลายน้ำที่ถอดได้จงมีค่าเป็น 1 การถอดบิตตัวถัดไปก็เช่นเดียวกัน จะเห็นได้ว่าสามารถถอดลายน้ำได้โดยตรงจากตารางเทียบค่าไบนารี โดยไม่ต้องอาศัยภาพต้นฉบับ

X'

-2	0	7	0
0	-3	4	0
-7	0	8	-5
4	1	0	-3

W

1	1	1	1
1	0	0	1
1	1	0	0
0	0	1	0

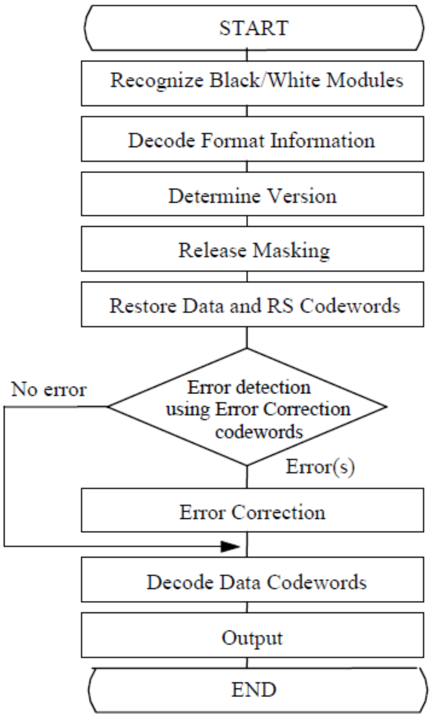
-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8
0	1	0	0	0	0	1	1	1	0	1	1	0	1	1	1	0

ภาพที่ 3-17 การถอดบิตลายน้ำออกจากสัมประสิทธิ์ภาพทดสอบโดยผ่านตารางเทียบค่าไบนารี

เนื่องจากการถอดบิตลายน้ำตามขั้นตอนจนครบทุบริเวณแล้ว นำบิตลายน้ำที่ได้ในแต่ละบริเวณมารวมกลับ ก็จะได้ลายน้ำดิจิตอลที่ถอดออกมาจากภาพทดสอบคืนมา

3.5 **การถอดรหัสข้อมูล** QR Code (Decode QR Code Procedure)

ในรายละเอียดการถอดรหัส QR Code ได้กล่าวไว้ในหัวข้อ 2.3 โดยมีขั้นตอนการคิดในการเข้ารหัส ดังภาพที่ 3-18



ภาพที่ 3-18 กระบวนการถอดรหัสของ QR Code

3.6 **การวัดค่าประสิทธิภาพ**

3.6.1 การวัดประสิทธิภาพการแทรกภาพลายน้ำดิจิตอล คือ การทดสอบประสิทธิภาพของการแทรกลายน้ำที่ทำให้มองไม่เห็นลายน้ำด้วยสายตาปกติและวัดความผิดเพี้ยนของภาพที่แทรกลายน้ำแล้วเปรียบเทียบกับภาพต้นฉบับเดิม โดยใช้เกณฑ์เชิงปริมาณ คือ ค่า $PSNR$ มีค่าสูงแสดงว่าภาพที่แทรกลายน้ำมีคุณภาพที่ดี ใกล้เคียงภาพต้นฉบับมาก คือมองไม่เห็นความผิดเพี้ยนจากการแทรกลายน้ำ (สมการคำนวณค่าความเหมือนของภาพอธิบายอยู่ในหัวข้อที่ 2.5.7.2)

3.6.2 การวัดประสิทธิภาพการถอดลายน้ำดิจิตอลออกจากภาพลายน้ำ คือ การทดสอบความเหมือนของลายน้ำดิจิตอลที่ถอดออกมาได้เทียบกับลายดิจิตอลต้นฉบับ โดยใช้เกณฑ์เชิงปริมาณ คือ ค่า NC เป็นค่าความเหมือนของภาพจะอยู่ในช่วงระหว่างค่า 0 ถึง 1 ซึ่งคุณภาพของลายน้ำดิจิตอลจะแปรผันตามค่าความเหมือนของภาพ คือ ถ้า NC มีค่าใกล้เคียง 1 มากเท่าใดคุณภาพของลายน้ำที่ถอดออกมาได้จะเหมือนลายน้ำต้นฉบับมากขึ้นเท่านั้น (สมการคำนวณค่าความเหมือนของภาพอธิบายอยู่ในหัวข้อ 2.5.7.3)

3.6.3 การวัดเวลาที่เครื่องคอมพิวเตอร์ใช้ในการฝังและถอดลายน้ำดิจิตอล คือ เป็นการคำนวณเวลาที่ใช้ในการประมวลผลแต่ละการแปลงข้อมูลเข้าสู่โดเมนความถี่ (อยู่ในหัวข้อ 2.5.7.4)

บทที่ 4

การทดลองและผลการทดลอง

ในบทนี้จะกล่าวถึงเครื่องมือที่ใช้ในการทดลอง ผลการทดลองของวิธีการฝังและการถอดลายน้ำดิจิตอลโดยใช้การทรานฟอร์ม 3 แบบ ได้แก่ DCT, DFT และ DWT ซึ่งผ่าน Look-up table และ Key โดยทดสอบประสิทธิภาพโดยคำนวณค่า PSNR, NC และ Time เปรียบเทียบกัน

4.1 เครื่องมือที่ใช้ในการทดลอง

4.1.1 ซอฟต์แวร์ที่ใช้ในการเขียนโปรแกรมคอมพิวเตอร์ คือ MATLAB เวอร์ชัน 7.11.0 (R2010b)

4.1.2 ฮาร์ดแวร์ที่ใช้ในการทดสอบคือ ซีพียู Intel® Core™ i3-330M Processor (3M Cache, 2.13 GHz) หน่วยความจำหลัก 4096 MB ฮาร์ดดิสก์ 300 GB บนระบบปฏิบัติการ Microsoft Window 7

4.1.3 รูปภาพที่ใช้ในการทดลอง มีดังนี้

4.1.3.1 ข้อมูล QR Code ต้นฉบับขนาด 512×512 จุดภาพ จำนวน 10 ภาพที่ผ่านกระบวนการเข้ารหัสจาก 10 ข้อความ มีไฟล์ภาพนามสกุล .BMP ซึ่งใช้ในการทรานฟอร์มแบบ DCT และ DFT

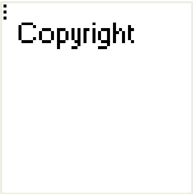
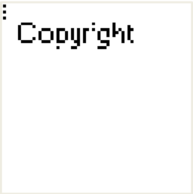
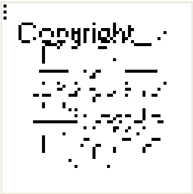
4.1.3.2 ข้อมูล QR Code ต้นฉบับขนาด 256×256 จุดภาพ จำนวน 10 ภาพที่ผ่านกระบวนการเข้ารหัสจาก 10 ข้อความ มีไฟล์ภาพนามสกุล .BMP ซึ่งใช้ในการทรานฟอร์มแบบ DWT

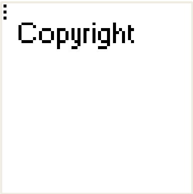
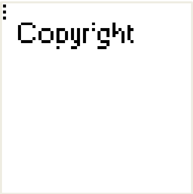
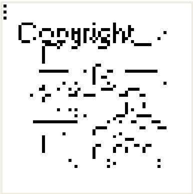
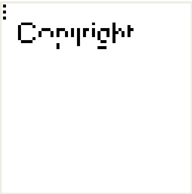
4.1.3.3 รูปภาพที่มีข้อความขนาด 64×64 จุดภาพ จำนวน 1 ภาพ มีไฟล์ภาพนามสกุล .BMP ใช้แทนลายน้ำดิจิตอล ซึ่งใช้แทนข้อความที่จะใช้ซ่อนในข้อมูล QR Code ต้นฉบับ

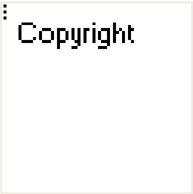
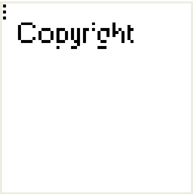
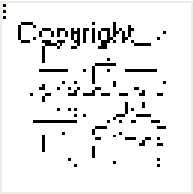
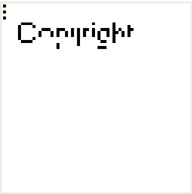
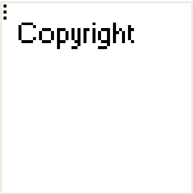
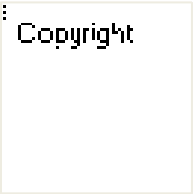
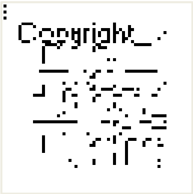
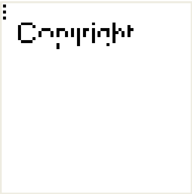
4.2 ผลการทดลอง

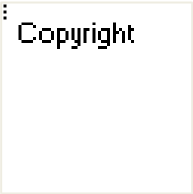
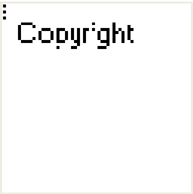
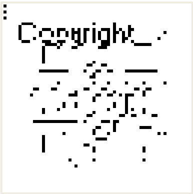
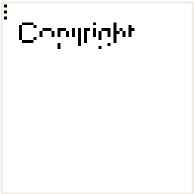
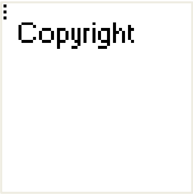
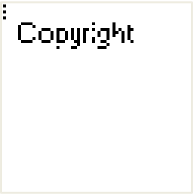
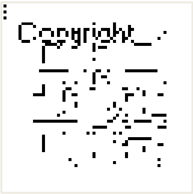
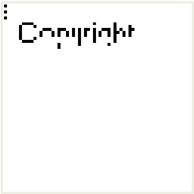
การทดลองนี้จะใช้รูปภาพที่มีข้อความขนาด 64×64 จุดภาพ แทนลายน้ำดิจิตอล 1 ภาพ ฝังลงในตำแหน่งที่ไม่ทับซ้อนกันบนข้อมูล QR Code ต้นฉบับ ซึ่งมีขนาด 512×512 จุดภาพ จำนวน 10 ภาพ ที่ผ่านกระบวนการเข้ารหัสจาก 10 ข้อความ จากนั้นทำการเปรียบเทียบรูปภาพต้นฉบับกับรูปภาพผลลัพธ์ที่ผ่านการฝังลายน้ำดิจิตอลที่ผ่านการทรานฟอร์ม 3 แบบ ได้แก่ DCT, DFT และ DWT ซึ่งวัดประสิทธิภาพด้วยการวัดค่า PSNR ต่อมาทำถอดลายน้ำดิจิตอลจากรูปภาพผลลัพธ์ทั้งหมด แล้วทำการเปรียบเทียบลายน้ำดิจิตอลต้นฉบับกับลายน้ำดิจิตอลที่ถอดได้ ซึ่งวัดประสิทธิภาพด้วยการวัดค่า NC และ เกณฑ์การวัดเวลาที่เครื่องคอมพิวเตอร์ใช้ที่ผ่านการฝังและถอดลายดิจิตอลโดยการทรานฟอร์มทั้ง 3 แบบ ดังแสดงในตารางต่อไปนี้

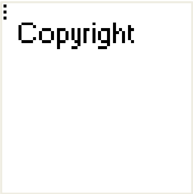
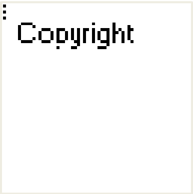
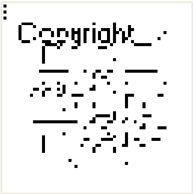
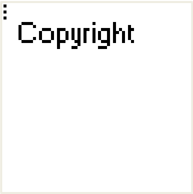
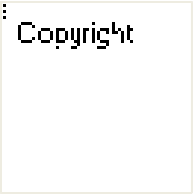
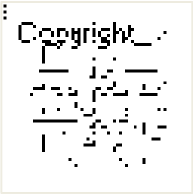
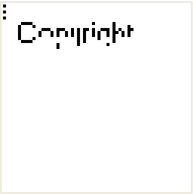
ตารางที่ 4-1 แสดงผลลัพธ์ในกระบวนการก่อนและหลังฝังลายน้ำดิจิตอล

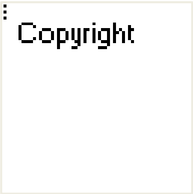
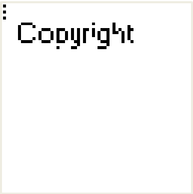
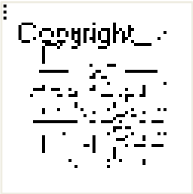
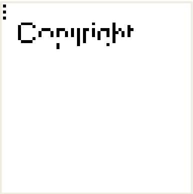
ลำดับ	ข้อมูลทั้งหมด			
	ข้อมูล	Discrete Cosine Transform	Discrete Fourier Transform	Discrete Wavelet Transform
	Chulalongkorn			
1				
	PSNR	11.1831	47.3690	67.0528
	Time	7.6908	0.5928	6.1620
				

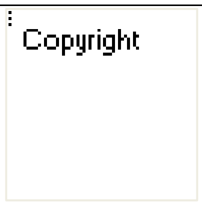
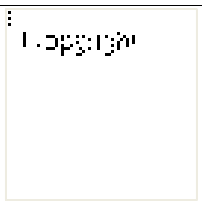
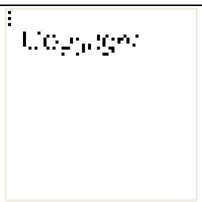
	NC	0.9980	0.1686	0.9805
	Time	1.8096	2.1840	3.1668
	Decode QR Code	Yes	Yes	Yes
2	Computer			
	PSNR	11.1831	47.2418	67.0528
	Time	7.9249	0.5928	6.3960
				
	NC	0.9976	0.1772	0.9805
	Time	1.7160	2.1684	2.9016
	Decode QR Code	Yes	Yes	Yes
3	Engineer			
	PSNR	11.1831	47.2526	67.0528
	Time	7.6128	0.6084	6.9264

				
	NC	0.9976	0.1747	0.9805
	Time	1.7940	2.0748	2.9952
	Decode QR Code	Yes	Yes	Yes
4	QR Code			
	PSNR	11.1831	47.3112	67.0528
	Time	7.7064	0.5616	6.2244
				
	NC	0.9988	0.1816	0.9812
	Time	1.7316	2.0592	2.8860
	Decode QR Code	Yes	Yes	Yes
5	Watermark			

	PSNR	11.1831	47.3692	67.0528
	Time	7.6596	0.5772	6.0216
				
	NC	0.9988	0.1696	0.9807
	Time	1.7472	2.1996	3.2136
	Decode QR Code	Yes	Yes	Yes
6	Embedding			
	PSNR	11.1831	47.3881	67.0528
	Time	7.6752	0.5772	6.9420
				
	NC	0.9983	0.1734	0.9812
	Time	1.7160	2.1372	2.9328
	Decode QR Code	Yes	Yes	Yes

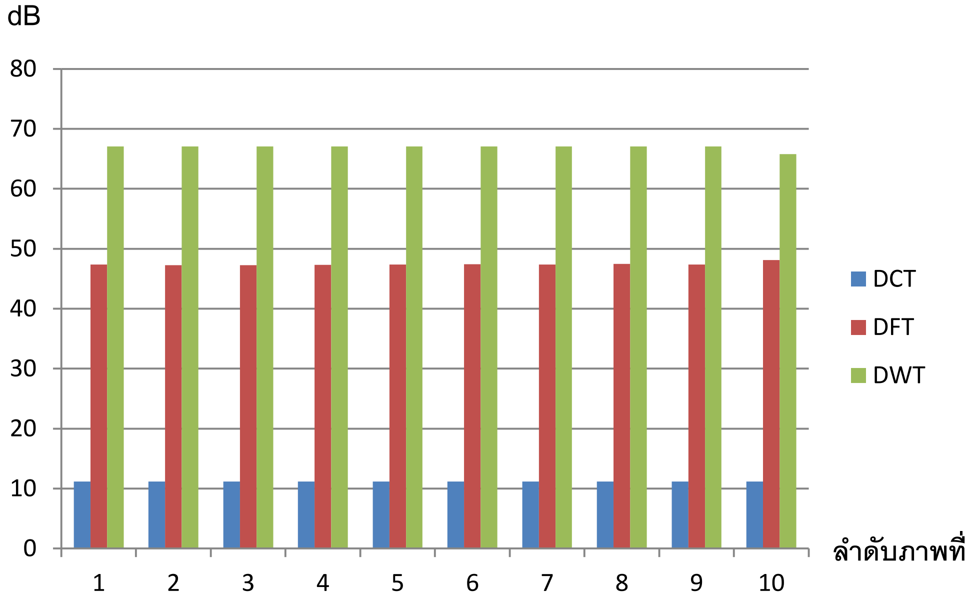
7	Extracting			
	PSNR	11.1831	47.3374	67.0528
	Time	7.8313	0.5772	7.0824
				
	NC	0.9988	0.1693	0.9802
	Time	1.7628	2.1684	3.0420
	Decode QR Code	Yes	Yes	Yes
8	Look-up table			
	PSNR	11.1831	47.4420	67.0528
	Time	7.7064	0.6396	6.9108
				
	NC	0.9985	0.1702	0.9810
	Time	1.7004	2.1684	2.8860
	Decode QR Code	Yes	Yes	Yes

9	Key			
	PSNR	11.1831	47.3541	67.0528
	Time	7.7844	0.5928	6.8016
				
	NC	0.9985	0.0486	0.9810
	Time	1.8408	2.1684	3.0732
	Decode QR Code	Yes	Yes	Yes
10	Suppat Rungraungsilp 5370368121 Master of Engineering Program in Computer Engineering Chulalongkorn University			
	PSNR	11.1831	48.1313	65.7744
	Time	7.8157	0.5772	6.0528

				
	NC	0.9890	0.1650	0.9468
	Time	1.7316	2.1684	3.0264
	Decode QR Code	Yes	Yes	Yes

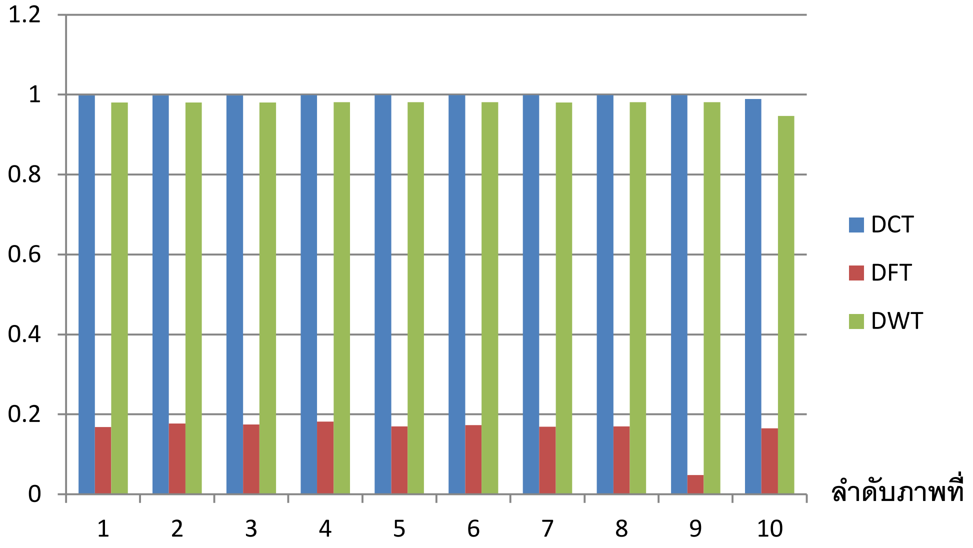
จากการทดลองสามารถวิเคราะห์ผลออกเป็น 4 กลุ่ม ดังนี้

4.2.1 PSNR (Peak signal-to-noise ratio) เมื่อเรานำมาพล็อตกราฟ โดยให้แกนตั้งเป็นค่า PSNR ส่วนแกนนอนเป็นลำดับภาพที่ ซึ่งได้กราฟดังนี้



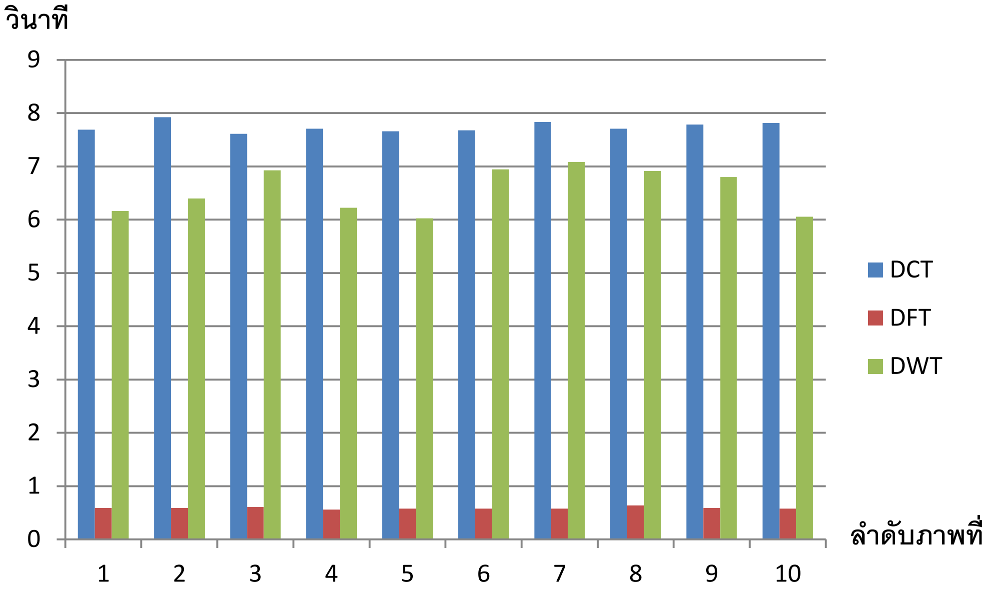
ภาพที่ 4-1 แผนภาพค่า PSNR ของทุกภาพ

4.2.2 NC (Normalized Cross Correlation) เมื่อเรานำมาพล็อตกราฟ โดยให้แกนตั้งเป็นค่า NC ส่วนแกนนอนเป็นลำดับภาพที่ ซึ่งได้กราฟดังนี้



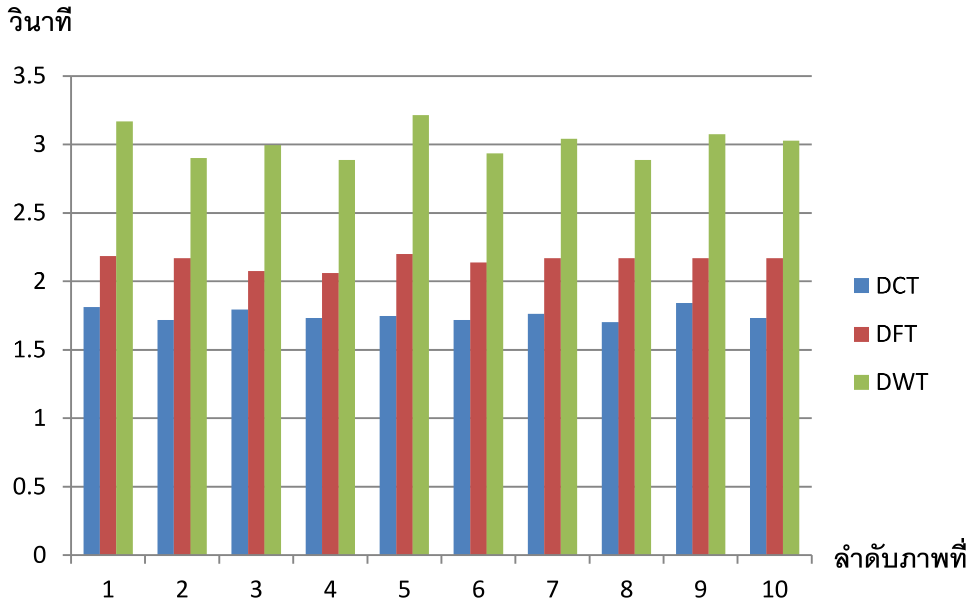
ภาพที่ 4-2 แผนภาพค่า NC ของทุกภาพ

4.2.3 Time (PSNR) เมื่อเรานำมาพล็อตกราฟ โดยให้แกนตั้งเป็นค่า Time หน่วยเป็นวินาที ส่วนแกนนอนเป็นลำดับภาพที่ ซึ่งได้กราฟดังนี้



ภาพที่ 4-3 แผนภาพค่า Time แต่ละ PSNR ของทุกภาพ

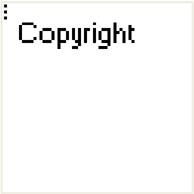
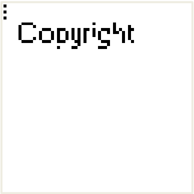
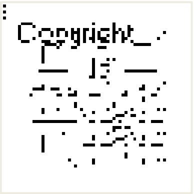
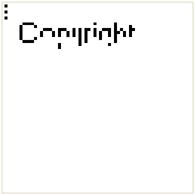
4.2.4 Time (NC) เมื่อเรานำมาพล็อตกราฟ โดยให้แกนตั้งเป็นค่า Time หน่วยเป็นวินาที ส่วนแกนนอนเป็นลำดับภาพที่ ซึ่งได้กราฟดังนี้

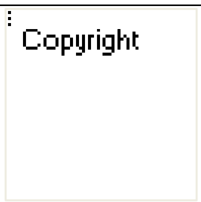
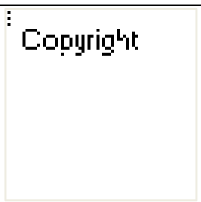
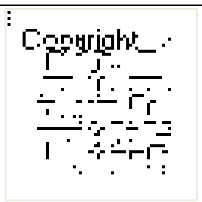
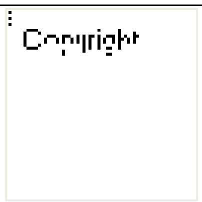
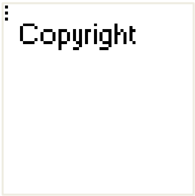
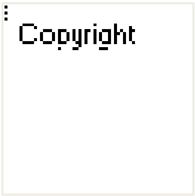
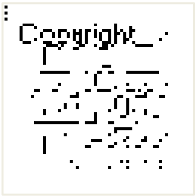
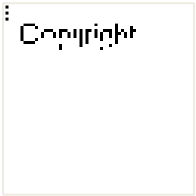


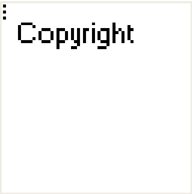
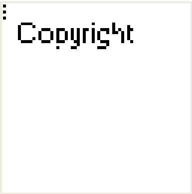
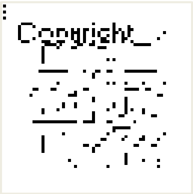
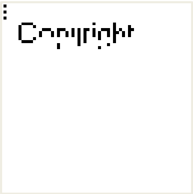
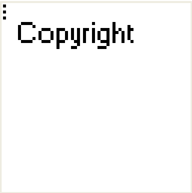
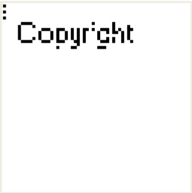
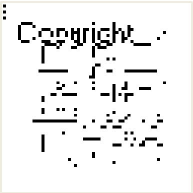
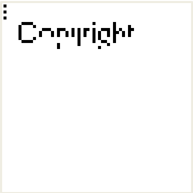
ภาพที่ 4-4 แผนภาพค่า Time แต่ละ NC ของทุกภาพ

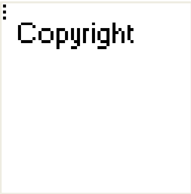
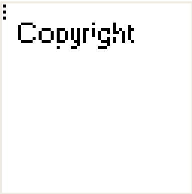
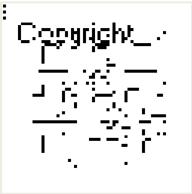
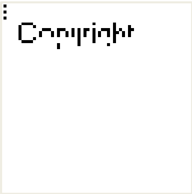
จากตารางที่ 4-1 ถึง 4-7 พบว่าเมื่อเราเพิ่มตัวอักษรในการเข้ารหัสข้อมูล QR Code มีผลต่อการฝังและถอดลายน้ำ เราจึงทดลองต่อเพื่อดูว่าการทรานฟอร์มแบบไหนสามารถถอดลายน้ำได้ออกมาได้ดีที่สุด เมื่อเราทำการเพิ่มตัวอักขระในการเข้ารหัส QR Code เรื่อยๆ โดยในการทดลองที่ผ่านมาเราเห็นว่าเมื่อมีการเพิ่มตัวอักขระมากขึ้นทำให้มีผลต่อ QR Code ต้นฉบับซึ่งจะนำไปสู่กระบวนการฝังและถอดลายน้ำ ซึ่งจากการทดลองที่ผ่านมาพบว่า เมื่อมีการเข้ารหัสอักขระมากยิ่งทำให้ถอดลายน้ำออกมาได้ไม่ดี เราจึงทดลองเพิ่มอักขระทีละ 1 ตัวเพื่อดูการเปลี่ยนแปลงแต่ในที่นี้จะขอแสดงเฉพาะการเข้ารหัสอักขระที่เป็นตัวอักษร เนื่องจากได้ลองทำการทดลองกับอักขระแบบอื่นพบว่าให้ค่าที่เหมือนกัน เพื่อไม่ให้ตางรางผลการทดลองเยอะจนเกินไป โดยสร้างตารางเหมือนการทดลองที่ผ่านมา โดยเรากำหนดขอบเขตจำนวนของการเข้ารหัสขึ้นอยู่กับเมื่อ 2 ใน 3 การทรานฟอร์มไม่สามารถอ่านลายน้ำได้ด้วยสายตาเมื่อผ่านกระบวนการถอดลายน้ำดิจิตอล จึงหยุดการเข้ารหัส QR Code ซึ่งได้ผลการทดลองดังตารางต่อไปนี้

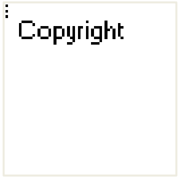
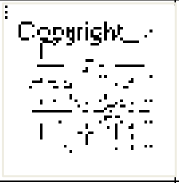
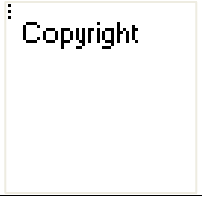
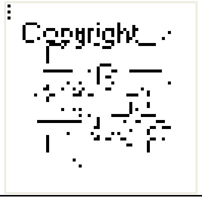
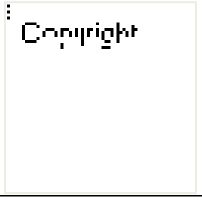
ตารางที่ 4-2 แสดงผลลัพธ์ในกระบวนการก่อนและหลังฝังลายน้ำดิจิตอล

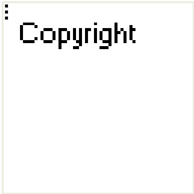
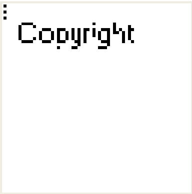
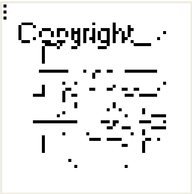
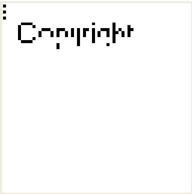
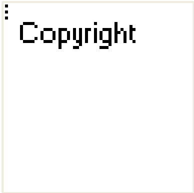
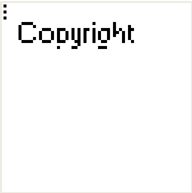
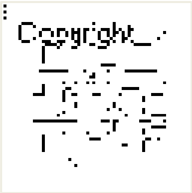
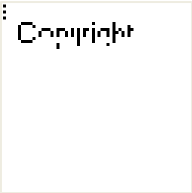
ลำดับ	ข้อมูลทั้งหมด			
	ข้อมูล	Discrete Cosine Transform	Discrete Fourier Transform	Discrete Wavelet Transform
1	A			
	PSNR	11.1831	47.3933	67.0528
	Time	9.1417	0.6396	6.3648
				
	NC	0.9980	0.0503	0.9810
	Time	1.8096	2.1996	2.5896
	Decode QR Code	Yes	Yes	Yes
2	AB			
	PSNR	11.1831	47.3205	67.0528
	Time	8.3773	0.5772	6.2244

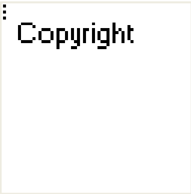
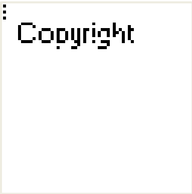
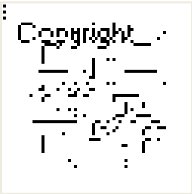
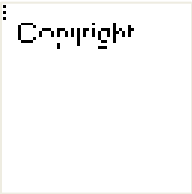
				
	NC	0.9988	0.0508	0.9800
	Time	1.9188	2.1684	2.6832
	Decode QR Code	Yes	Yes	Yes
3	ABC			
	PSNR	11.1831	47.3181	67.0528
	Time	8.3305	0.5772	6.2556
				
	NC	0.9995	0.0488	0.9807
	Time	1.9032	2.1840	2.6832
	Decode QR Code	Yes	Yes	Yes

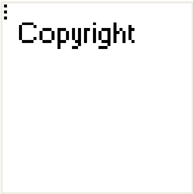
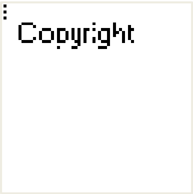
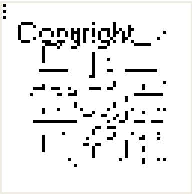
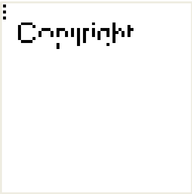
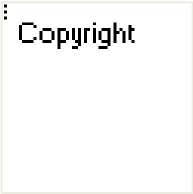
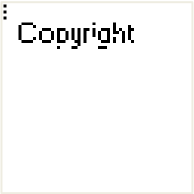
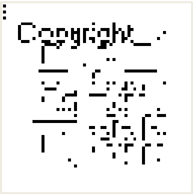
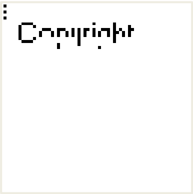
4	ABCD			
	PSNR	11.1831	47.4256	67.0528
	Time	8.8921	0.5772	6.0840
				
	NC	0.9985	0.0459	0.9807
	Time	1.7784	2.2464	2.6364
	Decode QR Code	Yes	Yes	Yes
5	ABCDE			
	PSNR	11.1831	47.3462	67.0528
	Time	8.2837	0.6084	6.0528
				

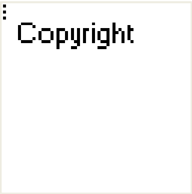
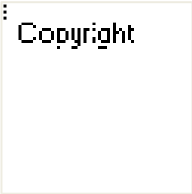
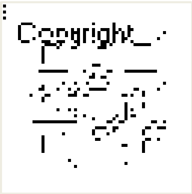
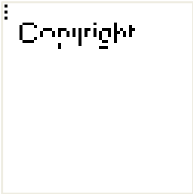
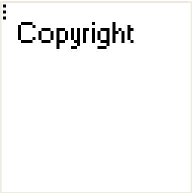
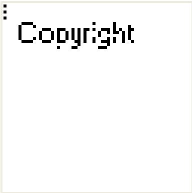
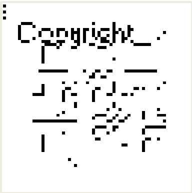
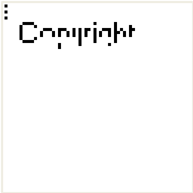
	NC	0.9980	0.0459	0.9802
	Time	1.8876	2.3244	2.6052
	Decode QR Code	Yes	Yes	Yes
6	ABCDEF			
	PSNR	11.1831	47.3845	67.0528
	Time	8.0653	0.6552	5.9904
				
	NC	0.9988	0.0464	0.9812
	Time	1.8096	2.1216	2.9172
	Decode QR Code	Yes	Yes	Yes
7	ABCDEFG			
	PSNR	11.1831	47.3559	67.0528

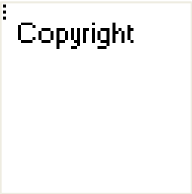
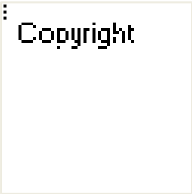
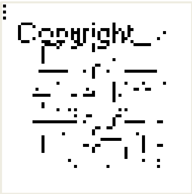
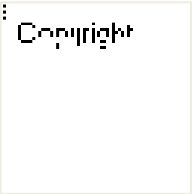
	Time	8.0497	0.5772	5.9280
				
	NC	0.9993	0.0444	0.9810
	Time	1.8408	2.1060	2.5116
	Decode QR Code	Yes	Yes	Yes
8	ABCDEFGH			
	PSNR	11.1831	47.3949	67.0528
	Time	8.1433	0.5772	6.5520
				
	NC	0.9995	0.0449	0.9805
	Time	1.7628	2.1216	3.0108

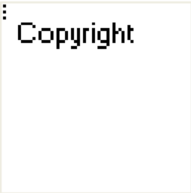
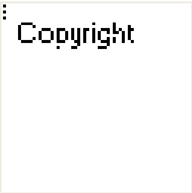
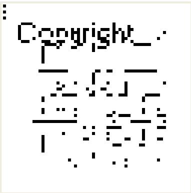
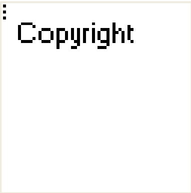
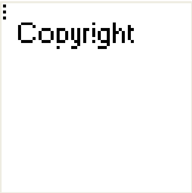
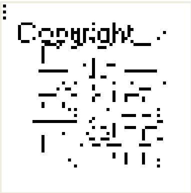
9	ABCDEFGHI			
	PSNR	11.1831	47.3808	67.0528
	Time	8.0497	0.5928	6.0840
				
	NC	0.9985	0.0435	0.9812
	Time	1.8096	2.1372	2.7768
	Decode QR Code	Yes	Yes	Yes
10	ABCDEFGHIJ			
	PSNR	11.1831	47.4600	67.0528
	Time	8.0341	0.5928	5.9904
				

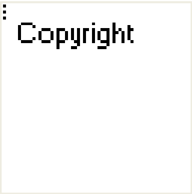
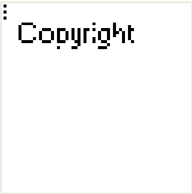
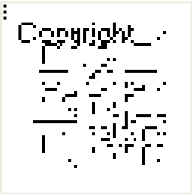
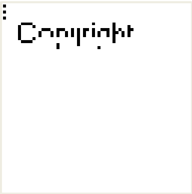
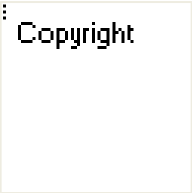
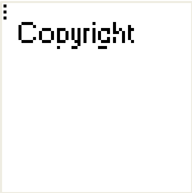
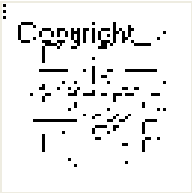
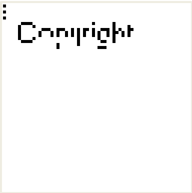
	NC	0.9988	0.0439	0.9812
	Time	1.7628	2.2776	2.5584
	Decode QR Code	Yes	Yes	Yes
11	ABCDEFGHIJK			
	PSNR	11.1831	47.4131	67.0528
	Time	8.2525	0.5772	5.8968
				
	NC	0.9985	0.0464	0.9805
	Time	1.7316	2.1528	2.5272
	Decode QR Code	Yes	Yes	Yes
12	ABCDEFGHIJKL			
	PSNR	11.1831	47.3771	67.0528

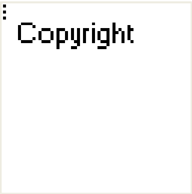
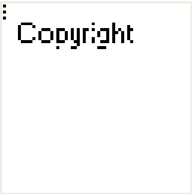
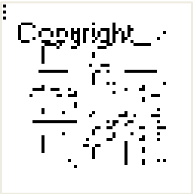
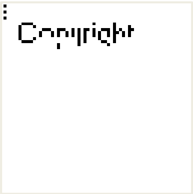
	Time	8.3461	0.6396	6.3180
				
	NC	0.9988	0.0496	0.9810
	Time	1.7472	2.1216	2.4492
	Decode QR Code	Yes	Yes	Yes
13	ABCDEFGHIJKLM			
	PSNR	11.1831	47.3884	67.0528
	Time	8.1901	0.5772	5.9592
				
	NC	0.9988	0.0515	0.9817
	Time	1.8408	2.1996	2.5428
	Decode QR Code	Yes	Yes	Yes
14	ABCDEFGHIJKLMN			

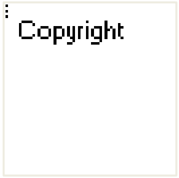
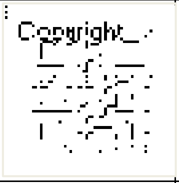
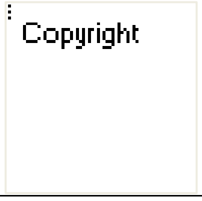
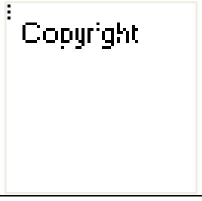
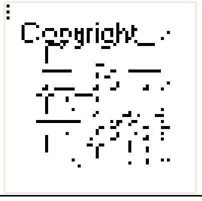
	PSNR	11.1831	47.3913	67.0528
	Time	8.4085	0.5772	6.1308
				
	NC	0.9990	0.0439	0.9805
	Time	1.8564	2.1372	2.6208
	Decode QR Code	Yes	Yes	Yes
15	ABCDEFGHIJKLMNO			
	PSNR	11.1831	47.4651	67.0528
	Time	8.3617	0.5772	6.2400
				
	NC	0.9985	0.0471	0.9812

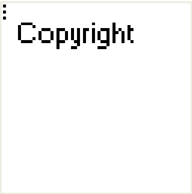
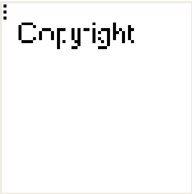
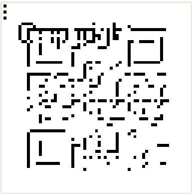
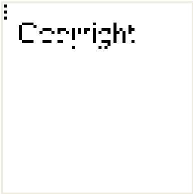
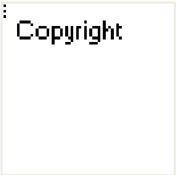
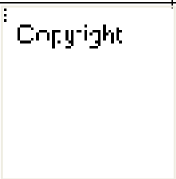
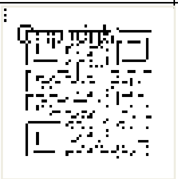
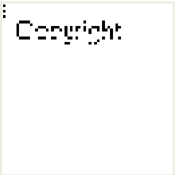
	Time	1.8408	2.0904	2.4960
	Decode QR Code	Yes	Yes	Yes
16	ABCDEFGHIJKLMNOP			
	PSNR	11.1831	47.3749	67.0528
	Time	8.2213	0.6240	6.5988
				
	NC	0.9995	0.0442	0.9800
	Time	1.8408	2.1216	2.6520
	Decode QR Code	Yes	Yes	Yes
17	ABCDEFGHIJKLMNOPQ			
	PSNR	11.1831	47.3871	67.0528
	Time	8.2213	0.5928	5.9436

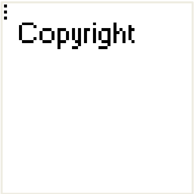
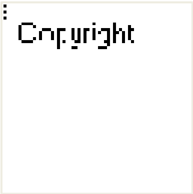
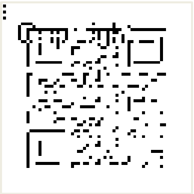
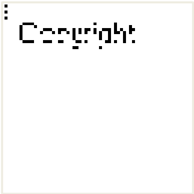
				
	NC	0.9993	0.0466	0.9802
	Time	1.7160	2.1060	2.6520
	Decode QR Code	Yes	Yes	Yes
18	ABCDEFGHIJKLMNOPQR			
	PSNR	11.1831	47.2750	67.0528
	Time	8.9701	0.5772	6.0216
				
	NC	0.9995	0.0508	0.9797
	Time	1.8408	2.2464	2.4492
	Decode QR Code	Yes	Yes	Yes

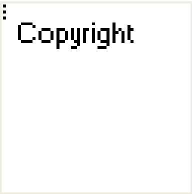
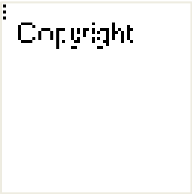
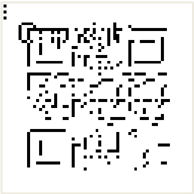
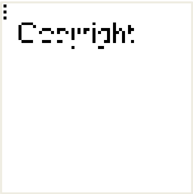
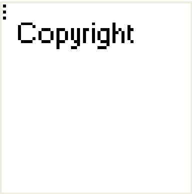
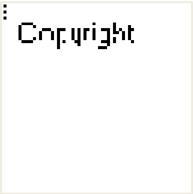
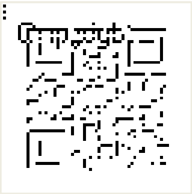
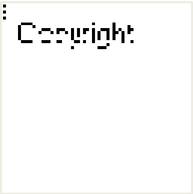
19	ABCDEFGHIJKLMNOPQRS			
	PSNR	11.1831	47.4113	67.0528
	Time	8.7361	0.6552	5.9748
				
	NC	0.9985	0.0518	0.9817
	Time	1.8876	2.0904	2.5272
	Decode QR Code	Yes	Yes	Yes
20	ABCDEFGHIJKLMNOPQRST			
	PSNR	11.1831	47.4709	67.0528
	Time	8.9233	0.5928	6.0840
				

	NC	0.9990	0.0444	0.9805
	Time	1.8408	2.0904	2.5896
	Decode QR Code	Yes	Yes	Yes
21	ABCDEFGHIJKLMNOPQRSTU			
	PSNR	11.1831	47.2605	67.0528
	Time	8.2369	0.5928	6.2088
				
	NC	0.9988	0.0457	0.9805
	Time	1.7940	2.2620	2.7612
	Decode QR Code	Yes	Yes	Yes
22	ABCDEFGHIJKLMNOPQRSTUV			
	PSNR	11.1831	47.3204	67.0528

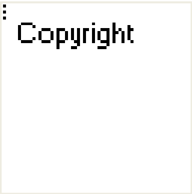
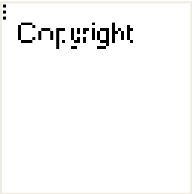
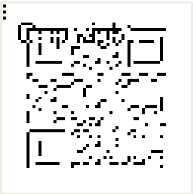
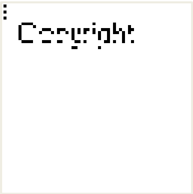
	Time	8.1745	0.6396	5.7252
				
	NC	0.9998	0.0457	0.9797
	Time	1.7940	2.0436	2.6832
	Decode QR Code	Yes	Yes	Yes
23	ABCDEFGHIJKLMNOPQRSTUVW			
	PSNR	11.1831	47.2830	67.0528
	Time	7.7532	0.5772	6.2712
				
	NC	0.9985	0.0459	0.9790
	Time	1.7316	2.1060	2.3712
	Decode QR Code	Yes	Yes	Yes
24	ABCDEFGHIJKLMNOPQRSTUVWX			

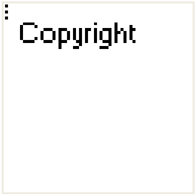
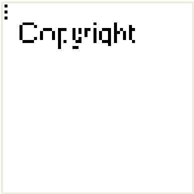
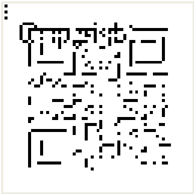
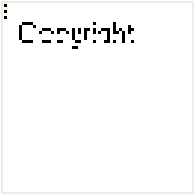
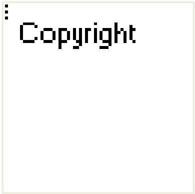
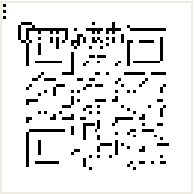
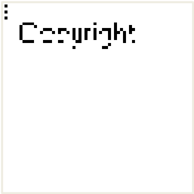
	PSNR	11.1831	47.7398	66.6108
	Time	8.6113	0.6864	6.0996
				
	NC	0.9966	0.0911	0.9646
	Time	1.7628	2.0748	2.6520
	Decode QR Code	Yes	Yes	Yes
25	ABCDEFGHIJKLMNOPQRSTUVWXY			
	PSNR	11.1831	47.7921	66.6577
	Time	8.5177	0.5928	5.8500
				
	NC	0.9961	0.0884	0.9661

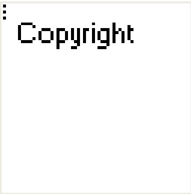
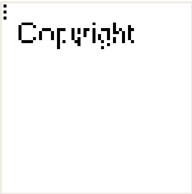
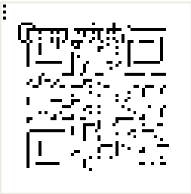
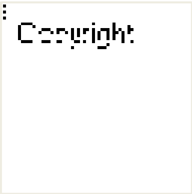
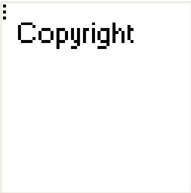
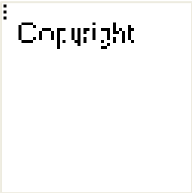
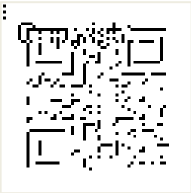
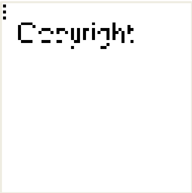
	Time	1.7316	2.0904	2.5740
	Decode QR Code	Yes	Yes	Yes
26	ABCDEFGHIJKLMNOPQRSTUVWXYZ			
	PSNR	11.1831	47.8014	66.5968
	Time	8.3929	0.5928	6.1620
				
	NC	0.9971	0.0879	0.9656
	Time	1.7628	2.1528	2.4024
	Decode QR Code	Yes	Yes	Yes
27	ABCDEFGHIJKLMNOPQRSTUVWXYZA			
	PSNR	11.1831	47.6777	66.5184
	Time	7.8000	0.5928	6.1152

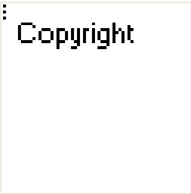
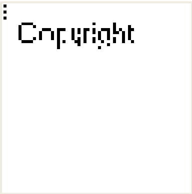
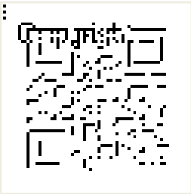
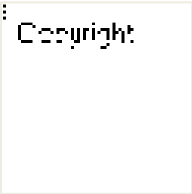
				
	NC	0.9961	0.0947	0.9641
	Time	1.7940	2.1216	2.6208
	Decode QR Code	Yes	Yes	Yes
28	ABCDEFGHIJKLMNOPQRSTUVWXYZAB			
	PSNR	11.1831	47.6402	66.6014
	Time	8.5957	0.5772	6.2244
				
	NC	0.9961	0.0918	0.9658
	Time	1.7628	2.0436	2.6208
	Decode QR Code	Yes	Yes	Yes

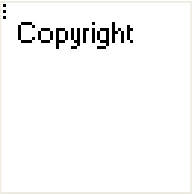
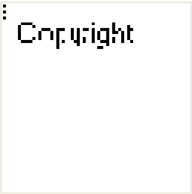
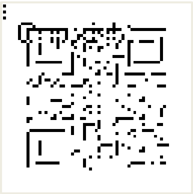
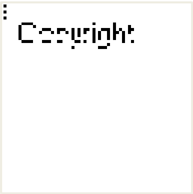
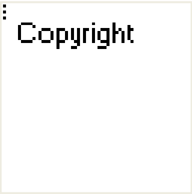
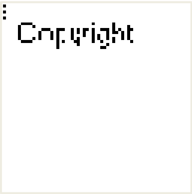
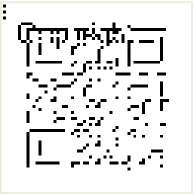
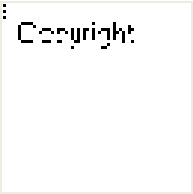
29	ABCDEFGHIJKLMNOPQRSTUVWXYZABC			
	PSNR	11.1831	47.5467	66.6980
	Time	8.4397	0.6084	6.5208
	NC	0.9963	0.0894	0.9663
	Time	1.8564	2.1684	2.6364
	Decode QR Code	Yes	Yes	Yes
30	ABCDEFGHIJKLMNOPQRSTUVWXYZABCD			
	PSNR	11.1831	47.6147	66.6648
	Time	8.5333	0.5928	5.9280

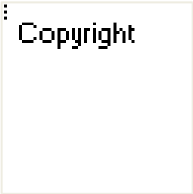
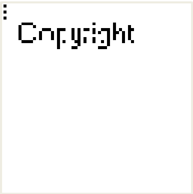
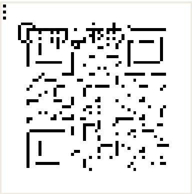
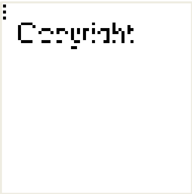
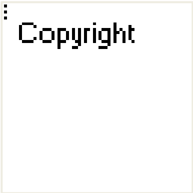
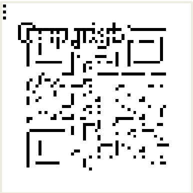
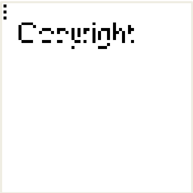
	NC	0.9968	0.0911	0.9658
	Time	1.7940	2.1840	2.4492
	Decode QR Code	Yes	Yes	Yes
31	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDE			
	PSNR	11.1831	47.7450	66.5689
	Time	8.7673	0.5772	6.1620
				
	NC	0.9971	0.0886	0.9658
	Time	1.7940	2.0592	2.4804
	Decode QR Code	Yes	Yes	Yes
32	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEF			
	PSNR	11.1831	47.7906	66.6154

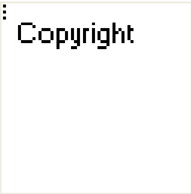
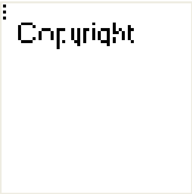
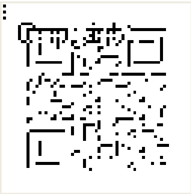
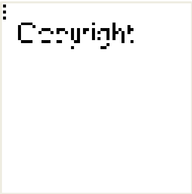
	Time	8.8921	0.6396	5.9124
				
	NC	0.9963	0.0857	0.9646
	Time	1.7628	2.1060	2.5116
	Decode QR Code	Yes	Yes	Yes
33	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFG			
	PSNR	11.1831	47.6324	66.6506
	Time	8.7049	0.6240	6.2244
				
	NC	0.9963	0.0911	0.9668
	Time	1.8096	2.1684	2.5428
	Decode QR Code	Yes	Yes	Yes

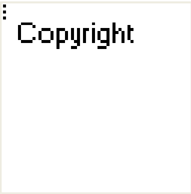
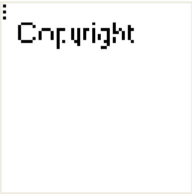
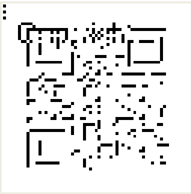
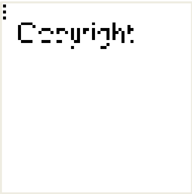
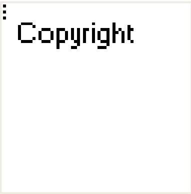
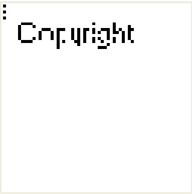
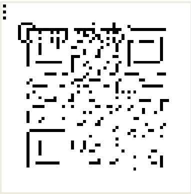
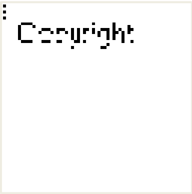
34	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGH			
	PSNR	11.1831	47.5841	66.6248
	Time	8.0185	0.6084	5.8344
				
	NC	0.9956	0.0898	0.9644
	Time	1.7628	2.2308	2.6676
	Decode QR Code	Yes	Yes	Yes
35	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHI			
	PSNR	11.1831	47.6559	66.6648
	Time	8.5177	0.5928	6.0528
				

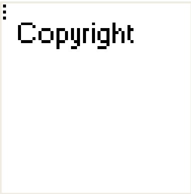
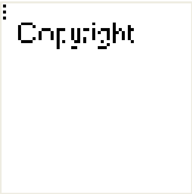
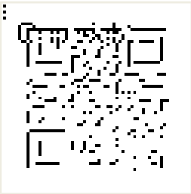
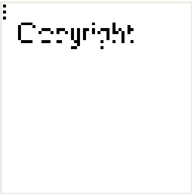
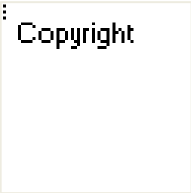
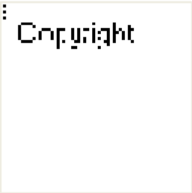
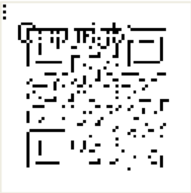
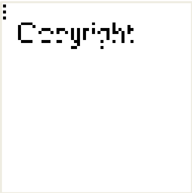
	NC	0.9956	0.0896	0.9670
	Time	1.7472	2.1684	2.6208
	Decode QR Code	Yes	Yes	Yes
36	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJ			
	PSNR	11.1831	47.6200	66.6459
	Time	7.8625	0.6240	5.8344
				
	NC	0.9961	0.0911	0.9656
	Time	1.8252	2.1216	2.6052
	Decode QR Code	Yes	Yes	Yes
37	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJK			
	PSNR	11.1831	47.6653	66.5968

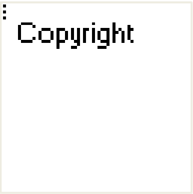
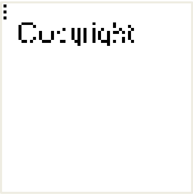
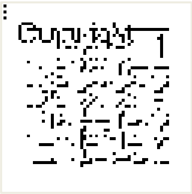
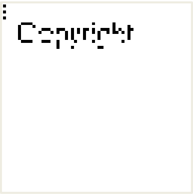
	Time	7.8000	0.6084	6.1776
				
	NC	0.9968	0.0913	0.9646
	Time	1.9188	2.0748	2.6988
	Decode QR Code	Yes	Yes	Yes
38	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKL			
	PSNR	11.1831	47.6528	66.5991
	Time	7.7376	0.5772	5.9904
				
	NC	0.9956	0.0938	0.9646
	Time	1.7940	2.2152	2.7144
	Decode QR Code	Yes	Yes	Yes

39	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLM			
	PSNR	11.1831	47.7622	66.6201
	Time	8.5333	0.5772	6.2868
				
	NC	0.9966	0.0864	0.9658
	Time	1.7628	2.1060	2.5896
	Decode QR Code	Yes	Yes	Yes
40	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLMN			
	PSNR	7.8000	47.6367	66.6061
	Time	11.1831	0.6708	6.0216
				

	NC	0.9966	0.0916	0.9646
	Time	1.7472	2.1684	2.5584
	Decode QR Code	Yes	Yes	Yes
41	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLMNO			
	PSNR	11.1831	47.6348	66.6957
	Time	7.8781	0.5928	6.0684
				
	NC	0.9963	0.0901	0.9663
	Time	1.7472	2.1372	2.4024
	Decode QR Code	Yes	Yes	Yes
42	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLMNOP			
	PSNR	11.1831	47.6816	66.6365

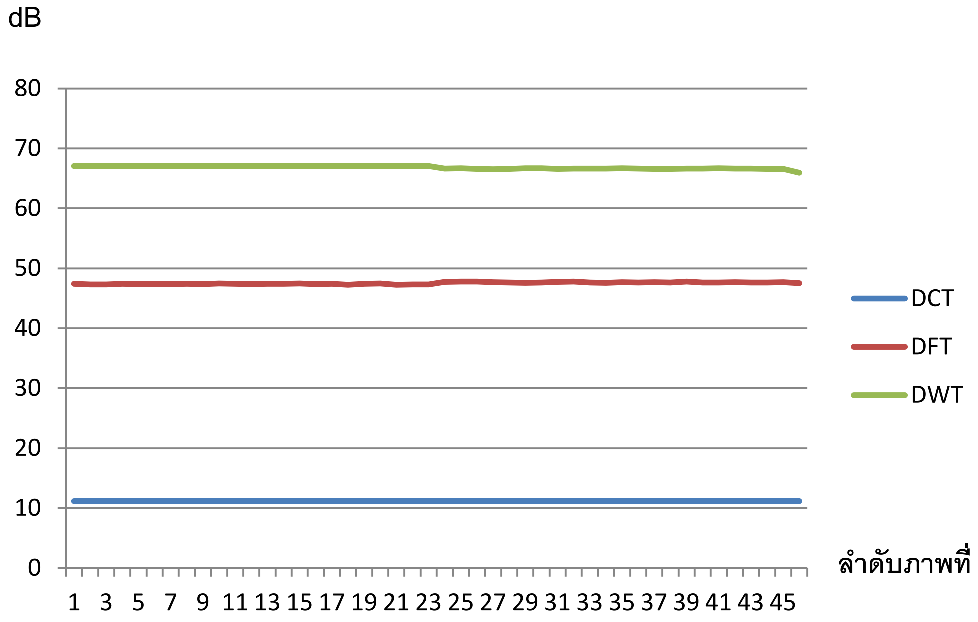
	Time	8.8453	0.5928	6.0840
				
	NC	0.9963	0.0942	0.9658
	Time	1.7628	2.1528	2.5116
	Decode QR Code	Yes	Yes	Yes
43	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLMNOPQ			
	PSNR	11.1831	47.6012	66.6295
	Time	8.5177	0.5928	6.0996
				
	NC	0.9966	0.0918	0.9653
	Time	1.8720	2.2152	2.5584
	Decode QR Code	Yes	Yes	Yes

44	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLMNOPQR			
	PSNR	11.1831	47.6103	66.5643
	Time	8.6269	0.6396	6.2400
				
	NC	0.9961	0.0867	0.9644
	Time	1.9032	2.0436	2.5272
	Decode QR Code	Yes	Yes	Yes
45	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLMNOPQRS			
	PSNR	11.1831	47.6551	66.5921
	Time	8.1121	0.6552	6.0060
				
	NC	0.9963	0.0840	0.9634
	Time	1.8252	2.1684	2.7300
	Decode QR Code	Yes	Yes	Yes

46	ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLMNOPQRST			
	PSNR	11.1831	47.5249	65.9413
	Time	7.8625	0.6084	5.8968
				
	NC	0.9934	0.0823	0.8977
	Time	1.8252	2.1528	2.7144
	Decode QR Code	Yes	Yes	Yes

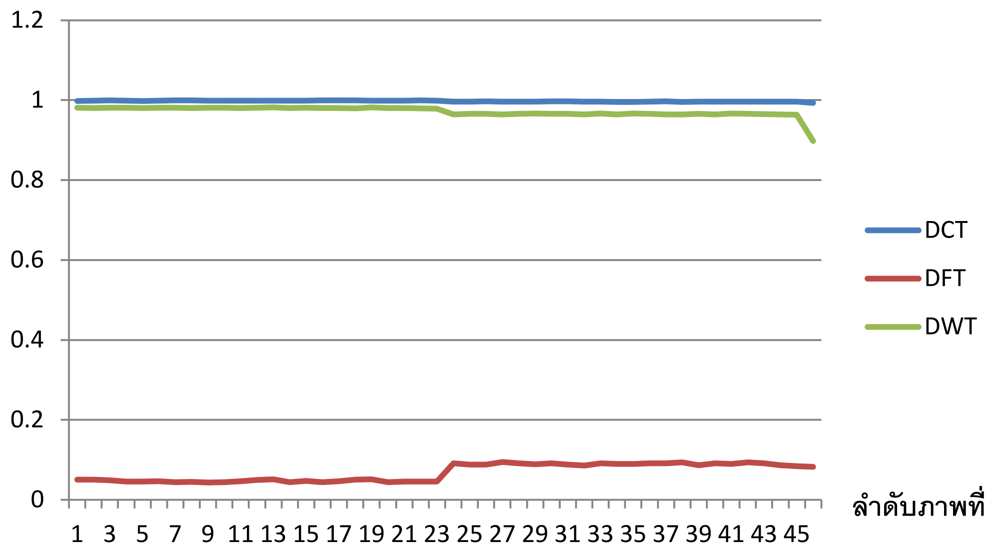
จากการทดลองสามารถวิเคราะห์ผลออกเป็น 4 กลุ่ม ดังนี้

4.2.5 PSNR (Peak signal-to-noise ratio) เมื่อเรานำมาพล็อตกราฟ โดยให้แกนตั้งเป็นค่า PSNR ส่วนแกนนอนเป็นลำดับภาพที่ ซึ่งได้กราฟดังนี้



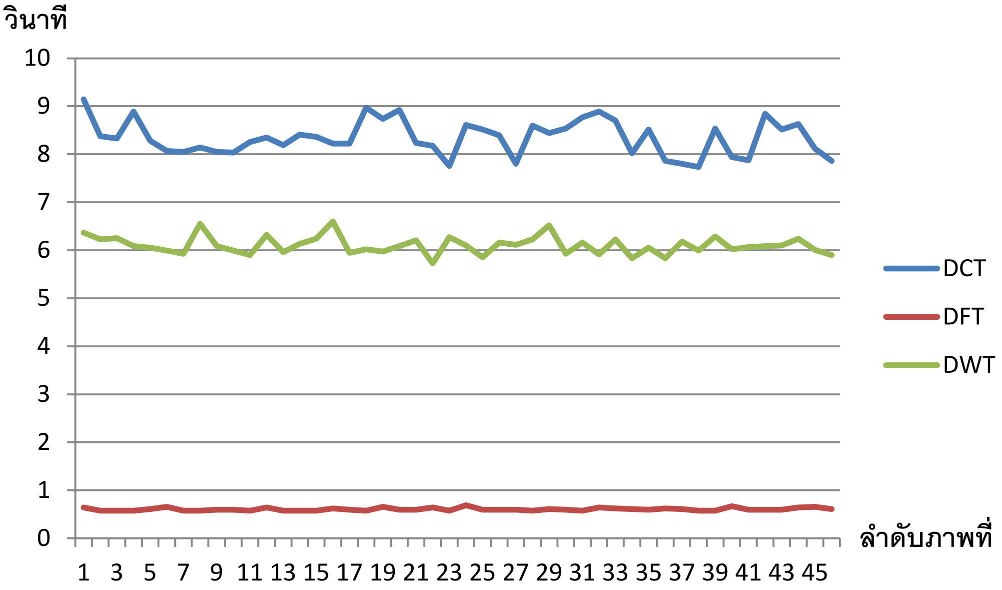
ภาพที่ 4-5 แผนภาพค่า PSNR ของทุกภาพ

4.2.6 NC (Normalized Cross Correlation) เมื่อเรานำมาพล็อตกราฟ โดยให้แกนตั้งเป็นค่า NC ส่วนแกนนอนเป็นลำดับภาพที่ ซึ่งได้กราฟดังนี้



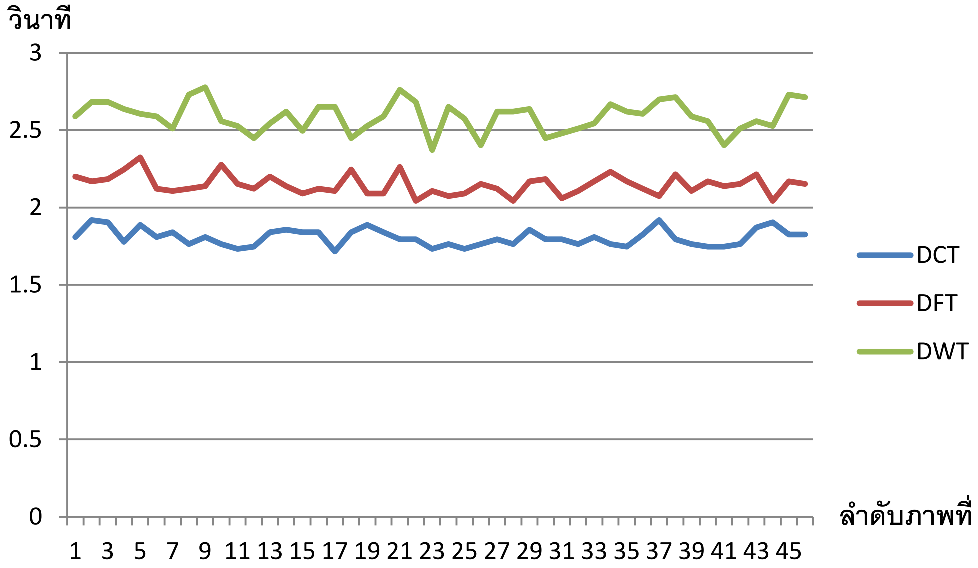
ภาพที่ 4-6 แผนภาพค่า NC ของทุกภาพ

4.2.7 Time (PSNR) เมื่อเรานำมาพล็อตกราฟ โดยให้แกนตั้งเป็นค่า Time หน่วยเป็นวินาที ส่วนแกนนอนเป็นลำดับภาพที่ ซึ่งได้กราฟดังนี้



ภาพที่ 4-7 แผนภาพค่า Time แต่ละ PSNR ของทุกภาพ

4.2.8 Time (NC) เมื่อเรานำมาพล็อตกราฟ โดยให้แกนตั้งเป็นค่า Time หน่วยเป็นวินาที ส่วนแกนนอนเป็นลำดับภาพที่ ซึ่งได้กราฟดังนี้



ภาพที่ 4-8 แผนภาพค่า Time แต่ละ NC ของทุกภาพ

บทที่ 5

สรุปผลการวิจัย

ในบทนี้จะกล่าวถึงผลสรุปงานวิจัย ปัญหาที่พบ และข้อเสนอแนะอันจะเป็นแนวทางการพัฒนา QR Code ร่วมกับลายน้ำดิจิตอลเพื่อซ่อนข้อมูล

5.1 บทสรุป

วิทยานิพนธ์ฉบับนี้ได้นำเสนอการฝังลายน้ำดิจิตอลบน QR Code เพื่อซ่อนข้อมูล โดยกระทำใน 3 โดเมนเปรียบเทียบกัน ได้แก่ Discrete Cosine Transform (DCT), Discrete Fourier Transform (DFT) และ Discrete Wavelet Transform (DWT) โดยใช้เกณฑ์การวัดประสิทธิภาพของลายน้ำดิจิตอลแบ่งเป็น 3 ส่วนตามกระบวนการวิจัย คือ กระบวนการฝังลายน้ำดิจิตอลใช้ค่า PSNR (Peak signal-to-noise ratio), กระบวนการถอดลายน้ำดิจิตอลใช้ค่า NC (Normalized Cross Correlation) และการวัดเวลาที่เครื่องคอมพิวเตอร์ใช้ในการฝังและถอดลายน้ำดิจิตอล

จากการทดลองช่วงแรกสามารถวิเคราะห์ผลออกเป็น 4 กลุ่ม คือ 4.2.1 - 4.2.4 เมื่อเราพิจารณาการคำนวณค่า PSNR พบว่า DWT ให้ค่าที่ดีสุดรองลงมาเป็น DFT และ DCT ตามลำดับ เมื่อเราพิจารณาการคำนวณค่า NC พบว่า DCT ให้ค่าที่ดีสุดรองลงมาเป็น DWT และ DFT ตามลำดับ แต่ผลที่ตามมาพบว่า QR Code ที่ผ่านการฝังลายน้ำดิจิตอลที่ผ่านการทรานฟอร์มแบบ DCT เกิดการเปลี่ยนแปลงไปจากเดิมมากที่สุดจากการพิจารณาค่า PSNR ที่ให้ค่าน้อยที่สุด โดยจากการทดลอง เรากำหนดค่า k ในการประมวลผลภาพเป็น 200 (ใน DCT และ DFT) เพื่อให้สามารถถอดลายน้ำดิจิตอลออกมาให้ได้มากที่สุด โดยเราทดสอบมาแล้วว่าค่านี้ให้ผลได้ดีที่สุด เมื่อเราพิจารณาเวลาในการคำนวณ PSNR พบว่า DFT ให้ค่าที่ดีที่สุดรองลงมาเป็น DCT และ DWT ตามลำดับ เมื่อเราพิจารณาเวลาในการคำนวณ NC พบว่า DCT ให้ค่าที่ดีที่สุดรองลงมาเป็น DFT และ DWT ตามลำดับ

จากการทดลองช่วงสองสามารถวิเคราะห์ผลออกเป็น 4 กลุ่ม คือ 4.2.5 - 4.2.8 พบว่า ใน DCT และ DWT สามารถถอดลายน้ำหรือยอมรับลายน้ำที่อ่านออกได้ด้วยสายตาซึ่งผ่านการเข้ารหัส QR Code ไม่เกิน 2947 อักขระ หรือ ค่า NC อยู่ในช่วงที่มากกว่า 0.8 แต่ไม่เกิน 1 ใน DFT

สามารถถอดลายน้ำหรือยอมรับลายน้ำที่อ่านออกได้ด้วยสายตาซึ่งผ่านการเข้ารหัส QR Code ไม่เกิน 23 อักขระ หรือ มีค่า NC อยู่ในช่วงที่มากกว่า 0.04 แต่ไม่เกิน 0.10 ซึ่งเป็นผลมาจากใช้การทรานฟอร์มแบบ DFT ได้ค่าจำนวนเชิงซ้อน แต่ฟังก์ชัน Correlation ใน Matlab กำหนดให้รับค่าจำนวนจริงเท่านั้น รวมถึงเกิดจากจำนวนสัญลักษณ์ในบล็อกยิ่งมีมากยิ่งมีผลต่อการถอดลายน้ำดิจิตอล [41] โดยปัญหาที่พบเมื่อทำการเข้ารหัสคิวอาร์โค้ดมากกว่า 2947 อัขระ ไม่สามารถเข้ารหัสได้ เนื่องจากขนาดของภาพคิวอาร์โค้ดต้นฉบับในงานวิจัยเรากำหนดขอบเขตการวิจัยให้มีขนาด 512x512 พิกเซล ซึ่งขนาดภาพไม่เพียงพอต่อการเข้ารหัสคิวอาร์โค้ด นอกจากนั้นสามารถถอดรหัสคิวอาร์โค้ดได้มากที่สุดที่ 421 อักขระ ใน DFT และ DWT จากการทดลองทั้งหมดแสดงให้เห็นว่าการทรานฟอร์มแต่ละแบบมีข้อดีและข้อเสียที่แตกต่างกันขึ้นอยู่กับการเลือกใช้ให้เหมาะสม

5.2 ข้อเสนอแนะ

จากผลการทดลอง พบว่าวิธีการที่ได้พัฒนาขึ้นมีข้อจำกัดบางประการ ควรปรับปรุงโดยให้มีการทำการวิจัยและศึกษาต่อไปเพื่อให้มีประสิทธิภาพมากขึ้น ข้อจำกัดที่ใช้ในการฝังลายน้ำดิจิตอล นั้นจำเป็นจะต้องมีการผันกลับกระบวนการเพื่อให้ได้รูปภาพที่มีลักษณะใกล้เคียงกับของเดิมให้มากที่สุดก่อนเข้าสู่กระบวนการถอดลายน้ำได้ จึงควรมีการนำเทคนิคอื่นๆ เข้ามาประยุกต์ใช้เพิ่มเติม เช่น การฝังลายน้ำดิจิตอลบนฟูเรียร์ร่วมกับโดเมนอื่น เช่น ฝังลายน้ำดิจิตอลบน DFT ร่วมกับ DWT ในงานวิจัยของ Kang และคณะ [47] หรือนำเทคนิคการค้นหาคุณลักษณะ (Feature-based scheme) ซึ่งจะทำการตรวจหาคุณลักษณะของข้อมูล QR Code ที่ถูกเปลี่ยนแปลงไปก่อน จากนั้นทำการผันกลับกระบวนการให้ข้อมูล QR Code มีลักษณะใกล้เคียงกับของเดิมให้มากที่สุดก่อนเข้าสู่กระบวนการถอดลายน้ำ โดยเทคนิคนี้จะมีจุดเด่นตรงที่สามารถนำไปประยุกต์ใช้ได้กับทุกโดเมนความถี่ เช่น ในงานวิจัยของ Mikolajczyk และ Schmid [48] หรือในงานวิจัยของ Tuytelaars และ Gool [49]

รายการอ้างอิง

[1] ISO/IEC. Information technology-Automatic identification and data capture techniques-Bar code symbology-QR Code. First Edition. 18004. Switzerland: ISO copyright office, 2000.

[2] Denso wave incorporated. About QR Code [Online]. 1994. Available from : http://www.denso-wave.com/qrcode/index-e.html [2011, April 20]

[3] Kaywa Services for the mobile internet incorporated. QR-Code Generator [Online]. Available from : http://qrcode.kaywa.com [2011, April 20]

[4] Kato, H. and Tan, K. T. Pervasive 2D barcodes for camera phone applications. IEEE Pervasive Computing. (2007) : 76-85.

[5] Toye, E., Sharp, R., Madhavapeddy, A., Scott, D., Upton, E. and Blackwell, A. Interacting with mobile services : an evaluation of camera-phones and visual tags. Journal of Personal and Ubiquitous Computing (2007) : 97-106.

[6] Ballagas, R. and Rohs, M. Mobile phones as pointing devices. Workshop on Pervasive Mobile Interaction Devices (2005) : 27-30.

[7] Chu, C.H., Yang, D.N. and Chen, M.S. Image stabilization for 2D barcode in handheld devices. Proceedings of the 15th international conference on Multimedia (2007) : 697-706.

[8] Tsung-Yu Liu, Tan-Hsu Tan and Yu-Ling Chu. 2D barcode and augmented reality supported english learning system. Proceedings of the 6th IEEE/ACIS International Conference on Computer and Information Science (2007) : 5-10.

[9] Claycomb, W. and Dongwan Shin. Using a two dimensional colorized barcode solution for Authentication in pervasive computing. Proceedings of the 6th IEEE/ACIS International Conference on Pervasive Services (2006) : 173-180.

[10] Jeng-Fang Weng, Yo-Ping Huang, and Shang-Lin Hsieh. The Study of RSA Algorithm on QR Code Design. Thesis for Master of Science Department of Computer Science and Engineering Tatung University, 2008.

[11] ZXing. ZXing Decoder Online [Online]. Available from : http://zxing.org/w/decode.jspx [2011, April 20]

[12] Hirakawa, M. and Iijima, J. A Study on Data Management Using Mobile Computing With Digital Watermark Technology. International Conference on Service Systems and Service Management (2009) : 186 – 191.

[13] Rakocevic, I., Reljin, B. and Reljin I. A method for providing digital image authenticity. Telecommunications in modern satellite, cable and broadcasting services 1 (1999) : 173-176.

[14] Yue Liu and Mingjun Liu. Automatic Recognition Algorithm of Quick Response Code Based on Embedded System. Proceedings of the 6th IEEE International Conference on Intelligent Systems Design and Applications 2 (2006) : 783-788.

[15] Peizhuo Lv, Shengli Lai, Leian Liu, Yongming Zhuo and Kai Liu. Research and implementation of automatic recognition of two dimensional barcode mobile computing. IEEE International Conference on Mobile Technology, Applications and Systems (2005) : 1-4.

[16] Donald D. Hearn and M. Pauline Baker. Computer Graphics with Open GL. Fourth Edition. Printice Hall Inc., 2010.

[17] Thumrongrat Amornraksa and Bundit Thipakorn. เอกสารประกอบการเรียนการสอนภายในมหาวิทยาลัยฯ. Department of Computer Engineering King Mongkut's University of Technology Thonburi, 2548.

[18] สุพจน์ นิตย์สุวัฒน์. ลายน้ำดิจิตอล. วารสารพัฒนาเทคนิคศึกษาสถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ. 15 (ตุลาคม-ธันวาคม 2545) : 11-16.

[19] Tong Won Seok and Jin Woo Hong. Audio Watermarking for Copyright Protection of Digital Audio Data. Electronics Letters 37 (2001) : 60-61.

[20] Cvejic, N., Keskinarkaus, A. and Seppanen, T. Audio watermarking using m-sequences and temporal masking. IEEE Workshops on Applications of Signal Processing to Audio and Acoustics (2001) : 227-230.

[21] มหศักดิ์ เกตุฉ่ำ. การฝังลายน้ำดิจิตอลบนสัญญาณเสียงโดยใช้วิธีการเชิงพันธุกรรม และดีสครีตเวฟเล็ตทรานฟอร์ม. วิทยานิพนธ์ปริญญาดุษฎีบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2552.

[22] Fabien A. P. Petitcolas, Ross J. Anderson and Markus G. Kuhn. Information Hiding-A Survey. Proceedings of the IEEE 87 (1999): 1062-1078.

[23] Katzenbeisser, S. and Petitcolas, F. A. P. Information Hiding Techniques for Steganography and Digital Watermarking Artech House, Boston, London, 1999.

[24] นริศรา โชคอำนวยชัย. การทำภาพพิมพ์ลายน้ำดิจิตอลในภาพกราฟฟิกส์โดยใช้วิธีเวฟเลตฟังชัน. วิทยานิพนธ์ปริญญามหาบัณฑิต คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี, 2544.

[25] Liwei Chen and Mingfu Li. An effective blind watermark algorithm based on DCT. World Congress on Intelligent Control and Automation (2008) : 6822–6825.

[26] Vongpradhip S. and Rungraungsilp S. QR code using invisible watermarking in frequency domain. International Conference on ICT and Knowledge Engineering (ICT & Knowledge Engineering) (January 2012) : 47–52.

[27] Chiou-Ting Hsu and Ja-Ling Wu. Hidden Digital Watermarks in Images. IEEE Transaction on image processing 8 (1999) : 58-68.

[28] สิทธิพงศ์ กำเนิดศิริ. การแทรกคำอธิบายคุณลักษณะลายไทยโดยใช้การทำลายน้ำดิจิตอลแบบมองไม่เห็น. วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาคอมพิวเตอร์ศึกษา คณะอุตสาหกรรมบัณฑิต มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 2546

[29] สิริพร ผลสมบูรณ์. ลายน้ำดิจิตอลสำหรับรูปภาพที่คืนสภาพได้จากการหมุน การปรับขนาดและการเคลื่อนที่ โดยใช้ฟังก์ชันเอ็กซ์โพเนนเชียลเชิงซ้อน. วิทยานิพนธ์ปริญญาดุษฎีบัณฑิต,ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2547.

[30] Xiangui Kang, Rui Yang and Jiwu Huang. Geometric Invariant Audio Watermarking Based on an LCM Feature. IEEE Transactions on Multimedia 13 (2011) : 181 – 190.

[31] P.W. Wong and E. J. Delp. Security and watermarking of multimedia contents. Proceedings of the Society of Photo-optical Instrumentation Engineers 1 (1999) : 156-166.

[32] วิจิตรา ตรีพลกุล. การทำลายน้ำดิจิตอลบนเอกสารราชการ. วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาคอมพิวเตอร์ศึกษา คณะอุตสาหกรรมบัณฑิต มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 2546.

[33] Cox, I. J., Kilian, J., Leighton, F. T. and Shamoon, T. Secure spread spectrum watermarking for multimedia. IEEE Transactions on Image Processing 6 (December 1997) : 1673-1687.

[34] Ng, K. S., Cheng, L. M., Cheng, L. L. and Wong, M. K. Adaptive watermarking by using pixel position shifting technique. IEEE Transactions on Consumer Electronics 45 (November 1999) : 1057-1064.

[35] George, M., Chouinard, J. V. and Georganas, N. Digital watermarking of images and video using direct sequence spread spectrum techniques. IEEE Canadian Conference on Electrical and Computer Engineering 1 (1999) : 116-121.

[36] Mobasseri, B.G. A spatial digital video watermark that survives MPEG. Proceedings International Conference on Information Technology: Coding and Computing (2000): 68-73.

[37] Yeung, M. M. and Mintzer, F. An invisible watermarking technique for image verification. Proceedings International conference on Image Processing 2 (October 1997) : 680-683.

[38] Qingzhong Liu, Sung, A. H., Ribeiro, B. M. and Ferreira, R. Steganalysis of multi-class JPEG images based on expanded Markov features and polynomial fitting. IEEE International Joint Conference on Neural Networks

(IEEE World Congress on Computational Intelligence) (June 2008) : 3352-3357.

[39] Mortazavi, S. A., Pour, A. N. and Kato, T. An efficient distributed group key management using hierarchical approach with Diffie-Hellman and Symmetric Algorithm: DHSA. International Symposium on Computer Networks and Distributed Systems (CNDS) (February 2011) : 49-54.

[40] Schyndel, R. G., Tirkel, A. Z. and Osborne, C. F. A digital watermark. IEEE International Conference on Image Processing 2 (November 1994) : 86-90.

[41] ประทาน โชติพันธุ์บัณฑิตย์. การซ่อนข้อมูลและความมั่นคงสำหรับเอกสารสิ่งพิมพ์. วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2552.

[42] นริศ หนูหอม. การฝังลายน้ำอเนกประสงค์บนรูปภาพดิจิตอลเพื่อตรวจสอบรับรองความเป็นต้นฉบับและพิสูจน์ความเป็นเจ้าของลิขสิทธิ์. วิทยานิพนธ์ปริญญาดุษฎีบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2553.

[43] สุวรฐ สุตันยสุวรรณ. การปรับปรุงความหนาแน่นของเทคโนยีการฝังข้อมูลลงบนกระดาษ. วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2546

[44] Kundur D. and Hatzinakos D. Watermarking for Image Authentication. Proceedings of the IEEE International conference on Image Processing 87 (July 1998) : 1167-1180.

[45] Ruanaidh, J. J. K. O., Dowling, W. J. and Boland, F. M. Phase watermarking of digital images. Proceedings of the IEEE International conference on Image Processing 3 (1996) : 239-242.

[46] Li Li, Rui-Ling Wang and Chin-Chen Chang. A Digital Watermark Algorithm for QR Code. International Journal of Intelligent Information Processing 2 (2011) : 29-36.

[47] Kang, X. J., Dong L. J. and Wang Y. A Digital Watermarking Algorithm Based on Image Segmentation and DFT. Proceedings of the First International Conference on Information Science and Engineering (2009) : 1511-1514.

[48] Mikolajczyk K. and Schmid C. Scale and affine invariant interest point detectors. International Journal of Computer Vision 60 (2004) : 63-86.

[49] Tuytelaars, T. and Gool, L. V. Matching widely separated views based on affine invariant regions. International Journal of Computer Vision 59 (2004) : 61-85.

ภาคผนวก

ภาคผนวก ก
คำจำกัดความที่เกี่ยวข้องกับ QR Code

1. Alignment Pattern หมายถึง ตำแหน่งในข้อมูล QR Code เป็นส่วนที่ re-synchronise coordinate mapping module ของข้อมูล QR Code กรณีที่ ข้อมูล QR Code เกิดการบิดเบือน
2. Character Count Indicator หมายถึง จำนวนของข้อมูล
3. Data Masking หมายถึง กระบวนการ XOR บิตระหว่างส่วนที่เข้ารหัสกับ Mask Pattern เพื่อให้ได้ข้อมูล QR Code ที่มีสีขาวและสีดำที่สมดุลกันซึ่งช่วยให้ไม่รบกวนการประมวลผลข้อมูล QR Code
4. Data Mask Pattern Reference หมายถึง เลขไบนารี 3 บิตซึ่งเป็นตัวแทนในแต่ละ Data Masking Patterns
5. Encoding Region หมายถึง ส่วนที่เกิดจากการเข้ารหัสของข้อมูลอินพุต, Error Correction, Version และ Format Information
6. Format Information หมายถึง มีข้อมูล 15 บิตที่ประกอบด้วยข้อมูล 5 บิตและ Bose-Chaudhuri-Hocquenghem ซึ่งเป็นเรื่องการเข้ารหัสที่เกี่ยวข้องกับ Error Correcting 10 บิต
7. QR Code หมายถึง บาร์โค้ด 2 มิติที่มีลักษณะการเก็บข้อมูลเป็นแบบเมตริกซ์ มีตั้งแต่เวอร์ชัน 1 ถึง 40
8. Function Pattern หมายถึง โดยทั่วไปจะเป็นส่วนประกอบหนึ่งของสัญลักษณ์ เช่น Finder, Separator, Timing Patterns และ Alignment Patterns ซึ่งมีไว้สำหรับการวางสัญลักษณ์หรือแยกแยะคุณลักษณะที่ช่วยในการถอดรหัส
9. Mode หมายถึง ตัวแทนที่แสดงให้ทราบถึงกลุ่มของอักขระ
10. Mode Indicator หมายถึง เลขไบนารี 4 บิตที่ใช้บอกว่าข้อมูลของเราอยู่กลุ่มไหน

11. Padding Bit หมายถึง การเติมบิต 0 ไปยังตำแหน่งที่ว่างใน Codeword ท้ายสุดหลัง Terminator

12. Finder Pattern หมายถึง 1 ใน 3 ที่มีลักษณะเหมือนกันทั้ง 3 ของ Finder Pattern

13. Remainder Bit หมายถึง การเติมบิต 0 ซึ่งไม่ได้หมายถึงการแสดงถึงข้อมูล ใช้เติมไปยังตำแหน่งที่ว่างของส่วนการเข้ารหัสเมื่อการเข้ารหัสไม่สามารถแบ่งเป็นชุดละ 8 บิตได้ครบ

14. Remainder Codeword หมายถึง การเติม Codeword หลัง Error Correcting เพื่อเติมตำแหน่ง Codeword ที่ว่างให้เต็มถ้าขาด

15. Segment หมายถึง Function Pattern เป็นสีขาวซึ่งใช้แยก Finder Patterns

16. Terminator หมายถึง บิต 0 ทั้งหมดที่มีจำนวนตามที่กำหนด เพื่อบอกจุดสิ้นสุดของข้อมูล

17. Timing Pattern หมายถึง โมดูลขาวกับดำสลับกันทำให้สามารถคิดพิกัดของโมดูลภายในได้

18. Version หมายถึง ขนาดของ QR Code โดยมี Version 21 x 21 modules (Version 1) ถึง 177 x 177 modules (Version 40)

19. Version Information หมายถึง จะมีใน QR Code ตั้งแต่เวอร์ชัน 7 ขึ้นไป ซึ่งประกอบด้วย 18 บิต โดยที่ เป็นข้อมูล 6 บิตเป็น Error Correcting 12 บิต เพื่อเป็นข้อมูลในการคำนวณ Error Correcting

20. BCH หมายถึง การเข้ารหัสแบบ Bose-Chaudhuri-Hocquenghem

21. RS หมายถึง การเข้ารหัสแบบ Reed-Solomon

22. Module Positions หมายถึง เพื่อให้ง่ายในการอ้างถึงตำแหน่งของโมดูลเลยถูกกำหนดด้วย Row กับ Column ร่วมกันใน QR Code โดยรูปแบบ (i, j) ซึ่ง i เป็นการระบุ Row (นับจากบนสุดไล่ลงมา) และ j เป็นการระบุ Column (นับจากซ้ายไปขวา)

ที่แต่ละตำแหน่งของโมดูลโดยเริ่มนับจากตำแหน่ง 0 หรือ โมดูล (0,0) ซึ่งคือตำแหน่งบนสุดซ้ายมือของ QR Code

23. Byte notation หมายถึง ปริมาณของ Byte ถูกแสดงเป็น 6 ค่า

24. Version references หมายถึง QR Code ถูกเรียกในรูปแบบ V-E ซึ่ง V เป็นการระบุเลขเวอร์ชัน (1 ถึง 40) และ E เป็นการระบุระดับของ Error Correcting (L, M, Q, H) โดยในงานวิจัยสนใจแค่ระดับ L

ภาคผนวก ข

ตารางแสดงจำนวน Codewords ทุกเวอร์ชัน

Version	No. of Modules/ side (A)	Function pattern modules (B)	Format and version information modules (C)	Data modules except (C) ($D=A^2-B-C$)	Data capacity [codewords][a] (E)	Remainder Bits
1	21	202	31	208	26	0
2	25	235	31	359	44	7
3	29	243	31	567	70	7
4	33	251	31	807	100	7
5	37	259	31	1,079	134	7
6	41	267	31	1,383	172	7
7	45	390	67	1,568	196	0
8	49	398	67	1,936	242	0
9	53	406	67	2,336	292	0
10	57	414	67	2,768	346	0
11	61	422	67	3,232	404	0
12	65	430	67	3,728	466	0
13	69	438	67	4,256	532	0
14	73	611	67	4,651	581	3
15	77	619	67	5,243	655	3
16	81	627	67	5,867	733	3
17	85	635	67	6,523	815	3
18	89	643	67	7,211	901	3
19	93	651	67	7,931	991	3
20	97	659	67	8,683	1,085	3

21	101	882	67	9,252	1,156	4
22	105	890	67	10,068	1,258	4
23	109	898	67	10,916	1,364	4
24	113	906	67	11,796	1,474	4
25	117	914	67	12,708	1,588	4
26	121	922	67	13,652	1,706	4
27	125	930	67	14,628	1,828	4
28	129	1,203	67	15,371	1,921	3
29	133	1,211	67	16,411	2,051	3
30	137	1,219	67	17,483	2,185	3
31	141	1,227	67	18,587	2,323	3
32	145	1,235	67	19,723	2,465	3
33	149	1,243	67	20,891	2,611	3
34	153	1,251	67	22,091	2,761	3
35	157	1,574	67	23,008	2,876	0
36	161	1,582	67	24,272	3,034	0
37	165	1,590	67	25,568	3,196	0
38	169	1,598	67	26,896	3,362	0
39	173	1,606	67	28,256	3,532	0
40	177	1,614	67	29,648	3,706	0

ภาคผนวก ค
ตารางแสดงคุณสมบัติ Data Codewords ของ QR Code

Version	Error Correction Level	Number of Data Codewords	Number of Data Bits	Data Capacity	
				Numeric	Alphanumeric
1	L	19	152	41	25
	M	16	128	34	20
	Q	13	104	27	16
	H	9	72	17	10
2	L	34	272	77	47
	M	28	224	63	38
	Q	22	176	48	29
	H	16	128	34	20
3	L	55	440	127	77
	M	44	352	101	61
	Q	34	272	77	47
	H	26	208	58	35
4	L	80	640	187	114
	M	64	512	149	90
	Q	48	384	111	67
	H	36	288	82	50
5	L	108	864	255	154
	M	86	688	202	122
	Q	62	496	144	87
	H	46	368	106	64
6	L	136	1,088	322	195
	M	108	864	255	154
	Q	76	608	178	108
	H	60	480	139	84

7	L	156	1,248	370	224
	M	124	992	293	178
	Q	88	704	207	125
	H	66	528	154	93
8	L	194	1,552	461	279
	M	154	1,232	365	221
	Q	110	880	259	157
	H	86	688	202	122
9	L	232	1,856	552	335
	M	182	1,456	432	262
	Q	132	1,056	312	189
	H	100	800	235	143
10	L	274	2,192	652	395
	M	216	1,728	513	311
	Q	154	1,232	364	221
	H	122	976	288	174
11	L	324	2,592	772	468
	M	254	2,032	604	366
	Q	180	1,440	427	259
	H	140	1,120	331	200
12	L	370	2,960	883	535
	M	290	2,320	691	419
	Q	206	1,648	489	296
	H	158	1,264	374	227
13	L	428	3,424	1,022	619
	M	334	2,672	796	483
	Q	244	1,952	580	352
	H	180	1,440	427	259
14	L	461	3,688	1,101	667
	M	365	2,920	871	528

	Q	261	2,088	621	376
	H	197	1,576	468	283
15	L	523	4,184	1,250	758
	M	415	3,320	991	600
	Q	295	2,360	703	426
	H	223	1,784	530	321
16	L	589	4,712	1,408	854
	M	453	3,624	1,082	656
	Q	325	2,600	775	470
	H	253	2,024	602	365
17	L	647	5,176	1,548	938
	M	507	4,056	1,212	734
	Q	367	2,936	876	531
	H	283	2,264	674	408
18	L	721	5,768	1,725	1,046
	M	563	4,504	1,346	816
	Q	397	3,176	948	574
	H	313	2,504	746	452
19	L	795	6,360	1,903	1,153
	M	627	5,016	1,500	909
	Q	445	3,560	1,063	644
	H	341	2,728	813	493
20	L	861	6,888	2,061	1,249
	M	669	5,352	1,600	970
	Q	485	3,880	1,159	702
	H	385	3,080	919	557
21	L	932	7,456	2,232	1,352
	M	714	5,712	1,708	1,035
	Q	512	4,096	1,224	742
	H	406	3,248	969	587

22	L	1,006	8,048	2,409	1,460
	M	782	6,256	1,872	1,134
	Q	568	4,544	1,358	823
	H	442	3,536	1,056	640
23	L	1,094	8,752	2,620	1,588
	M	860	6,880	2,059	1,248
	Q	614	4,912	1,468	890
	H	464	3,712	1,108	672
24	L	1,174	9,392	2,812	1,704
	M	914	7,312	2,188	1,326
	Q	664	5,312	1,588	963
	H	514	4,112	1,228	744
25	L	1,276	10,208	3,057	1,853
	M	1,000	8,000	2,395	1,451
	Q	718	5,744	1,718	1,041
	H	538	4,304	1,286	779
26	L	1,370	10,960	3,283	1,990
	M	1,062	8,496	2,544	1,542
	Q	754	6,032	1,804	1,094
	H	596	4,768	1,425	864
27	L	1,468	11,744	3,517	2,132
	M	1,128	9,024	2,701	1,637
	Q	808	6,464	1,933	1,172
	H	628	5,024	1,501	910
28	L	1,531	12,248	3,669	2,223
	M	1,193	9,544	2,857	1,732
	Q	871	6,968	2,085	1,263
	H	661	5,288	1,581	958
29	L	1,631	13,048	3,909	2,369
	M	1,267	10,136	3,035	1,839

	Q	911	7,288	2,181	1,322
	H	701	5,608	1,677	1,016
30	L	1,735	13,880	4,158	2,520
	M	1,373	10,984	3,289	1,994
	Q	985	7,880	2,358	1,429
	H	745	5,960	1,782	1,080
31	L	1,843	14,744	4,417	2,677
	M	1,455	11,640	3,486	2,113
	Q	1,033	8,264	2,473	1,499
	H	793	6,344	1,897	1,150
32	L	1,955	15,640	4,686	2,840
	M	1,541	12,328	3,693	2,238
	Q	1,115	8,920	2,670	1,618
	H	845	6,760	2,022	1,226
33	L	2,071	16,568	4,965	3,009
	M	1,631	13,048	3,909	2,369
	Q	1,171	9,368	2,805	1,700
	H	901	7,208	2,157	1,307
34	L	2,191	17,528	5,253	3,183
	M	1,725	13,800	4,134	2,506
	Q	1,231	9,848	2,949	1,787
	H	961	7,688	2,301	1,394
35	L	2,306	18,448	5,529	3,351
	M	1,812	14,496	4,343	2,632
	Q	1,286	10,288	3,081	1,867
	H	986	7,888	2,361	1,431
36	L	2,434	19,472	5,836	3,537
	M	1,914	15,312	4,588	2,780
	Q	1,354	10,832	3,244	1,966
	H	1,054	8,432	2,524	1,530

37	L	2,566	20,528	6,153	3,729
	M	1,992	15,936	4,775	2,894
	Q	1,426	11,408	3,417	2,071
	H	1,096	8,768	2,625	1,591
38	L	2,702	21,616	6,479	3,927
	M	2,102	16,816	5,039	3,054
	Q	1,502	12,016	3,599	2,181
	H	1,142	9,136	2,735	1,658
39	L	2,812	22,496	6,743	4,087
	M	2,216	17,728	5,313	3,220
	Q	1,582	12,656	3,791	2,298
	H	1,222	9,776	2,927	1,774
40	L	2,956	23,648	7,089	4,296
	M	2,334	18,672	5,596	3,391
	Q	1,666	13,328	3,993	2,420
	H	1,276	10,208	3,057	1,852

ภาคผนวก ง
ตารางแสดงคุณสมบัติ Error Correction ของ QR Code

Version	Total Number of Codewords	Error Correction Level	Number of Error Correction Codewords	Value of p	Number of Error Correction Blocks	Error Correction Code per Block (c, k, r)[a]
1	26	L	7	3	1	(26,19,2)[b]
		M	10	2	1	(26,16,4)[b]
		Q	13	1	1	(26,13,6)[b]
		H	17	1	1	(26,9,8)[b]
2	44	L	10	2	1	(44,34,4)[b]
		M	16	0	1	(44,28,8)
		Q	22	0	1	(44,22,11)
		H	28	0	1	(44,16,14)
3	70	L	15	1	1	(70,55,7)[b]
		M	26	0	1	(70,44,13)
		Q	36	0	2	(35,17,9)
		H	44	0	2	(35,13,11)
4	100	L	20	0	1	(100,80,10)
		M	36		2	(50,32,9)
		Q	52		2	(50,24,13)
		H	64		4	(25,9,8)
5	134	L	26	0	1	(134,108,13)
		M	48		2	(67,43,12)
		Q	72		2	(33,15,9)
		H	88		2	(34,16,9)
					2	(33,11,11)
					2	(34,12,11)

6	172	L	36	0	2	(86,68,9)
		M	64		4	(43,27,8)
		Q	96		4	(43,19,12)
		H	112		4	(43,15,14)
7	196	L	40	0	2	(98,78,10)
		M	72		4	(49,31,9)
		Q	108		2	(32,14,9)
		H	130		4	(33,15,9)
					4	(39,13,13)
					1	(40,14,13)
8	242	L	48	0	2	(121,97,12)
		M	88		2	(60,38,11)
		Q	132		2	(61,39,11)
		H	156		4	(40,18,11)
					2	(41,19,11)
					4	(40,14,13)
					2	(41,15,13)
9	292	L	60	0	2	(146,116,15)
		M	110		3	(58,36,11)
		Q	160		2	(59,37,11)
		H	192		4	(36,16,10)
					4	(37,17,10)
					4	(36,12,12)
					4	(37,13,12)
10	346	L	72	0	2	(86,68,9)
		M	130		2	(87,69,9)
		Q	192		4	(69,43,13)
		H	224		1	(70,44,13)
					6	(43,19,12)
					2	(44,20,12)

					6	(43,15,14)
					2	(44,16,14)
11	404	L	80	0	4	(101,81,10)
		M	150		1	(80,50,15)
		Q	224		4	(81,51,15)
		H	264		4	(50,22,14)
					4	(51,23,14)
					3	(36,12,12)
					8	(37,13,12
12	466	L	96	0	2	(116,92,12)
		M	176		2	(117,93,12)
		Q	260		6	(58,36,11)
		H	308		2	(59,37,11)
					4	(46,20,13)
					6	(47,21,13)
					7	(42,14,14)
					4	(43,15,14)
13	532	L	104	0	4	(133,107,13)
		M	198		8	(59,37,11)
		Q	288		1	(60,38,11)
		H	352		8	(44,20,12)
					4	(45,21,12)
					12	(33,11,11)
					4	(34,12,11)
14	581	L	120	0	3	(145,115,15)
		M	216		1	(146,116,15)
		Q	320		4	(64,40,12)
		H	384		5	(65,41,12)
					11	(36,16,10)
					5	(37,17,10)

					11	(36,12,12)
					5	(37,13,12)
15	655	L	132	0	5	(109,87,11)
		M	240		1	(110,88,11)
		Q	360		5	(65,41,12)
		H	432		5	(66,42,12)
					5	(54,24,15)
					7	(55,25,15)
					11	(36,12,12)
					7	(37,13,12)
16	733	L	144	0	5	(122,98,12)
		M	280		1	(123,99,12)
		Q	408		7	(73,45,14)
		H	480		3	(74,46,14)
					15	(43,19,12)
					2	(44,20,12)
					3	(45,15,15)
					13	(46,16,15)
17	815	L	168	0	1	(135,107,14)
		M	308		5	(136,108,14)
		Q	448		10	(74,46,14)
		H	532		1	(75,47,14)
					1	(50,22,14)
					15	(51,23,14)
					2	(42,14,14)
					17	(43,15,14)
18	901	L	180	0	5	(150,120,15)
		M	338		1	(151,121,15)
		Q	504		9	(69,43,13)
		H	588		4	(70,44,13)

					17	(50,22,14)
					1	(51,23,14)
					2	(42,14,14)
					19	(43,15,14)
19	991	L	196	0	3	(141,113,14)
		M	364		4	(142,114,14)
		Q	546		3	(70,44,13)
		H	650		11	(71,45,13)
					17	(47,21,13)
					4	(48,22,13)
					9	(39,13,13)
					16	(40,14,13)
20	1,085	L	224	0	3	(135,107,14)
		M	416		5	(136,108,14)
		Q	600		3	(67,41,13)
		H	700		13	(68,42,13)
					15	(54,24,15)
					5	(55,25,15)
					15	(43,15,14)
					10	(44,16,14)
21	1,156	L	224	0	4	(144,116,14)
		M	442		4	(145,117,14)
		Q	644		17	(68,42,13)
		H	750		17	(50,22,14)
					6	(51,23,14)
					19	(46,16,15)
					6	(47,17,15)
22	1,258	L	252	0	2	(139,111,14)
		M	476		7	(140,112,14)
		Q	690		17	(74,46,14)

		H	816		7	(54,24,15)
					16	(55,25,15)
					34	(37,13,12)
23	1,364	L	270	0	4	(151,121,15)
		M	504		5	(152,122,15)
		Q	750		4	(75,47,14)
		H	900		14	(76,48,14)
					11	(54,24,15)
					14	(55,25,15)
					16	(45,15,15)
					14	(46,16,15)
24	1,474	L	300	0	6	(147,117,15)
		M	560		4	(148,118,15)
		Q	810		6	(73,45,14)
		H	960		14	(74,46,14)
					11	(54,24,15)
					16	(55,25,15)
					30	(46,16,15)
					2	(47,17,15)
25	1,588	L	312	0	8	(132,106,13)
		M	588		4	(133,107,13)
		Q	870		8	(75,47,14)
		H	1,050		13	(76,48,14)
					7	(54,24,15)
					22	(55,25,15)
					22	(45,15,15)
					13	(46,16,15)
26	1,706	L	336	0	10	(142,114,14)
		M	644		2	(143,115,14)
		Q	952		19	(74,46,14)

		H	1,110		4	(75,47,14)
					28	(50,22,14)
					6	(51,23,14)
					33	(46,16,15)
					4	(47,17,15)
27	1,828	L	360	0	8	(152,122,15)
		M	700		4	(153,123,15)
		Q	1,020		22	(73,45,14)
		H	1,200		3	(74,46,14)
					8	(53,23,15)
					26	(54,24,15)
					12	(45,15,15)
					28	(46,16,15)
28	1,921	L	390	0	3	(147,117,15)
		M	728		10	(148,118,15)
		Q	1,050		3	(73,45,14)
		H	1,260		23	(74,46,14)
					4	(54,24,15)
					31	(55,25,15)
					11	(45,15,15)
					31	(46,16,15)
29	2,051	L	420	0	7	(146,116,15)
		M	784		7	(147,117,15)
		Q	1,140		21	(73,45,14)
		H	1,350		7	(74,46,14)
					1	(53,23,15)
					37	(54,24,15)
					19	(45,15,15)
					26	(46,16,15)
30	2,185	L	450	0	5	(145,115,15)

		M	812		10	(146,116,15)
		Q	1,200		19	(75,47,14)
		H	1,440		10	(76,48,14)
					15	(54,24,15)
					25	(55,25,15)
					23	(45,15,15)
					25	(46,16,15)
31	2,323	L	480	0	13	(145,115,15)
		M	868		3	(146,116,15)
		Q	1,290		2	(74,46,14)
		H	1,530		29	(75,47,14)
					42	(54,24,15)
					1	(55,25,15)
					23	(45,15,15)
					28	(46,16,15)
32	2,465	L	510	0	17	(145,115,15)
		M	924		10	(74,46,14)
		Q	1,350		23	(75,47,14)
		H	1,620		10	(54,24,15)
					35	(55,25,15)
					19	(45,15,15)
					35	(46,16,15)
33	2,611	L	540	0	17	(145,115,15)
		M	980		1	(146,116,15)
		Q	1,440		14	(74,46,14)
		H	1,710		21	(75,47,14)
					29	(54,24,15)
					19	(55,25,15)
					11	(45,15,15)
					46	(46,16,15)

34	2,761	L	570	0	13	(145,115,15)
		M	1,036		6	(146,116,15)
		Q	1,530		14	(74,46,14)
		H	1,800		23	(75,47,14)
					44	(54,24,15)
					7	(55,25,15)
					59	(46,16,15)
					1	(47,17,15)
35	2,876	L	570	0	12	(151,121,15)
		M	1,064		7	(152,122,15)
		Q	1,590		12	(75,47,14)
		H	1,890		26	(76,48,14)
					39	(54,24,15)
					14	(55,25,15)
					22	(45,15,15)
					41	(46,16,15)
36	3,034	L	600	0	6	(151,121,15)
		M	1,120		14	(152,122,15)
		Q	1,680		6	(75,47,14)
		H	1,980		34	(76,48,14)
					46	(54,24,15)
					10	(55,25,15)
					2	(45,15,15)
					64	(46,16,15)
37	3,196	L	630	0	17	(152,122,15)
		M	1,204		4	(153,123,15)
		Q	1,770		29	(74,46,14)
		H	2,100		14	(75,47,14)
					49	(54,24,15)
					10	(55,25,15)

					24	(45,15,15)
					46	(46,16,15)
38	3,362	L	660	0	4	(152,122,15)
		M	1,260		18	(153,123,15)
		Q	1,860		13	(74,46,14)
		H	2,220		32	(75,47,14)
					48	(54,24,15)
					14	(55,25,15)
					42	(45,15,15)
					32	(46,16,15)
39	3,532	L	720	0	20	(147,117,15)
		M	1,316		4	(148,118,15)
		Q	1,950		40	(75,47,14)
		H	2,310		7	(76,48,14)
					43	(54,24,15)
					22	(55,25,15)
					10	(45,15,15)
					67	(46,16,15)
40	3,706	L	750	0	19	(148,118,15)
		M	1,372		6	(149,119,15)
		Q	2,040		18	(75,47,14)
		H	2,430		31	(76,48,14)
					34	(54,24,15)
					34	(55,25,15)
					20	(45,15,15)
					61	(46,16,15)

ซึ่งจากตารางจะมี c เป็นจำนวน Codewords ทั้งหมด ส่วน k เป็นจำนวน Data Codewords ทั้งหมด ส่วน r เป็น Error Correction capacity ซึ่งจะมีจำนวนน้อยกว่า Error Correction Codewords ครึ่งนึงเพื่อป้องกันความเป็นไปได้ที่จะเกิดความเข้าใจผิด

ภาคผนวก จ

ตารางกำหนดพหุนามการคำนวณ Error Correction Codewords

Number of Error Correction Codewords	Generator Polynomials
2	$x^{2} + \alpha^{25}x + \alpha$
5	$x^{5} + \alpha^{113}x^{4} + \alpha^{164}x^{3} + \alpha^{166}x^{2} + \alpha^{119}x + \alpha^{10}$
6	$x^{6} + \alpha^{116}x^{5} + x^{4} + \alpha^{134}x^{3} + \alpha^{5}x^{2} + \alpha^{176}x + \alpha^{15}$
7	$x^{7} + \alpha^{87}x^{6} + \alpha^{229}x^{5} + \alpha^{146}x^{4} + \alpha^{149}x^{3} + \alpha^{238}x^{2} + \alpha^{102}x + \alpha^{21}$
8	$x^{8} + \alpha^{175}x^{7} + \alpha^{238}x^{6} + \alpha^{208}x^{5} + \alpha^{249}x^{4} + \alpha^{215}x^{3} + \alpha^{252}x^{2} + \alpha^{196}x + \alpha^{28}$
10	$x^{10} + \alpha^{251}x^{9} + \alpha^{67}x^{8} + \alpha^{46}x^{7} + \alpha^{61}x^{6} + \alpha^{118}x^{5} + \alpha^{70}x^{4} + \alpha^{64}x^{3} + \alpha^{94}x^{2} + \alpha^{32}x + \alpha^{45}$
13	$x^{13} + \alpha^{74}x^{12} + \alpha^{152}x^{11} + \alpha^{176}x^{10} + \alpha^{100}x^{9} + \alpha^{86}x^{8} + \alpha^{100}x^{7} + \alpha^{106}x^{6} + \alpha^{104}x^{5} + \alpha^{130}x^{4} + \alpha^{218}x^{3} + \alpha^{206}x^{2} + \alpha^{140}x + \alpha^{78}$
14	$x^{14} + \alpha^{199}x^{13} + \alpha^{249}x^{12} + \alpha^{155}x^{11} + \alpha^{48}x^{10} + \alpha^{190}x^{9} + \alpha^{124}x^{8} + \alpha^{218}x^{7} + \alpha^{137}x^{6} + \alpha^{216}x^{5} + \alpha^{87}x^{4} + \alpha^{207}x^{3} + \alpha^{59}x^{2} + \alpha^{22}x + \alpha^{91}$
15	$x^{15} + \alpha^{8}x^{14} + \alpha^{183}x^{13} + \alpha^{61}x^{12} + \alpha^{91}x^{11} + \alpha^{202}x^{10} + \alpha^{37}x^{9} + \alpha^{51}x^{8} + \alpha^{58}x^{7} + \alpha^{58}x^{6} + \alpha^{237}x^{5} + \alpha^{140}x^{4} + \alpha^{124}x^{3} + \alpha^{5}x^{2} + \alpha^{99}x + \alpha^{105}$
16	$x^{16} + \alpha^{120}x^{15} + \alpha^{104}x^{14} + \alpha^{107}x^{13} + \alpha^{109}x^{12} + \alpha^{102}x^{11} + \alpha^{161}x^{10} + \alpha^{76}x^{9} + \alpha^{3}x^{8} + \alpha^{91}x^{7} + \alpha^{191}x^{6} + \alpha^{147}x^{5} + \alpha^{169}x^{4} + \alpha^{182}x^{3} + \alpha^{194}x^{2} + \alpha^{225}x + \alpha^{120}$
17	$x^{17} + \alpha^{43}x^{16} + \alpha^{139}x^{15} + \alpha^{206}x^{14} + \alpha^{78}x^{13} + \alpha^{43}x^{12} + \alpha^{239}x^{11} + \alpha^{123}x^{10} + \alpha^{206}x^{9} + \alpha^{214}x^{8} + \alpha^{147}x^{7} + \alpha^{24}x^{6} + \alpha^{99}x^{5} + \alpha^{150}x^{4} + \alpha^{39}x^{3} + \alpha^{243}x^{2} + \alpha^{163}x + \alpha^{136}$
18	$x^{18} + \alpha^{215}x^{17} + \alpha^{234}x^{16} + \alpha^{158}x^{15} + \alpha^{94}x^{14} + \alpha^{184}x^{13} + \alpha^{97}x^{12} + \alpha^{118}x^{11} + \alpha^{170}x^{10} + \alpha^{79}x^{9} + \alpha^{187}x^{8} + \alpha^{152}x^{7} + \alpha^{148}x^{6} + \alpha^{252}x^{5} + \alpha^{179}x^{4} + \alpha^{5}x^{3} + \alpha^{98}x^{2} + \alpha^{96}x + \alpha^{153}$

20	$x^{20} + \alpha^{17}x^{19} + \alpha^{60}x^{18} + \alpha^{79}x^{17} + \alpha^{50}x^{16} + \alpha^{61}x^{15} + \alpha^{163}x^{14} + \alpha^{26}x^{13} + \alpha^{187}x^{12} + \alpha^{202}x^{11} + \alpha^{180}x^{10} + \alpha^{221}x^{9} + \alpha^{225}x^{8} + \alpha^{83}x^{7} + \alpha^{239}x^{6} + \alpha^{156}x^{5} + \alpha^{164}x^{4} + \alpha^{212}x^{3} + \alpha^{212}x^{2} + \alpha^{188}x + \alpha^{190}$
22	$x^{22} + \alpha^{210}x^{21} + \alpha^{171}x^{20} + \alpha^{247}x^{19} + \alpha^{242}x^{18} + \alpha^{93}x^{17} + \alpha^{230}x^{16} + \alpha^{14}x^{15} + \alpha^{109}x^{14} + \alpha^{221}x^{13} + \alpha^{53}x^{12} + \alpha^{200}x^{11} + \alpha^{74}x^{10} + \alpha^{8}x^{9} + \alpha^{172}x^{8} + \alpha^{98}x^{7} + \alpha^{80}x^{6} + \alpha^{219}x^{5} + \alpha^{134}x^{4} + \alpha^{160}x^{3} + \alpha^{105}x^{2} + \alpha^{165}x + \alpha^{231}$
24	$x^{24} + \alpha^{229}x^{23} + \alpha^{121}x^{22} + \alpha^{135}x^{21} + \alpha^{48}x^{20} + \alpha^{211}x^{19} + \alpha^{117}x^{18} + \alpha^{251}x^{17} + \alpha^{126}x^{16} + \alpha^{159}x^{15} + \alpha^{180}x^{14} + \alpha^{169}x^{13} + \alpha^{152}x^{12} + \alpha^{192}x^{11} + \alpha^{226}x^{10} + \alpha^{228}x^{9} + \alpha^{218}x^{8} + \alpha^{111}x^{7} + x^{6} + \alpha^{117}x^{5} + \alpha^{232}x^{4} + \alpha^{87}x^{3} + \alpha^{96}x^{2} + \alpha^{227}x + \alpha^{21}$
26	$x^{26} + \alpha^{173}x^{25} + \alpha^{125}x^{24} + \alpha^{158}x^{23} + \alpha^{2}x^{22} + \alpha^{103}x^{21} + \alpha^{182}x^{20} + \alpha^{118}x^{19} + \alpha^{17}x^{18} + \alpha^{145}x^{17} + \alpha^{201}x^{16} + \alpha^{111}x^{15} + \alpha^{28}x^{14} + \alpha^{165}x^{13} + \alpha^{53}x^{12} + \alpha^{161}x^{11} + \alpha^{21}x^{10} + \alpha^{245}x^{9} + \alpha^{142}x^{8} + \alpha^{13}x^{7} + \alpha^{102}x^{6} + \alpha^{48}x^{5} + \alpha^{227}x^{4} + \alpha^{153}x^{3} + \alpha^{145}x^{2} + \alpha^{218}x + \alpha^{70}$
28	$x^{28} + \alpha^{168}x^{27} + \alpha^{223}x^{26} + \alpha^{200}x^{25} + \alpha^{104}x^{24} + \alpha^{224}x^{23} + \alpha^{234}x^{22} + \alpha^{108}x^{21} + \alpha^{180}x^{20} + \alpha^{110}x^{19} + \alpha^{190}x^{18} + \alpha^{195}x^{17} + \alpha^{147}x^{16} + \alpha^{205}x^{15} + \alpha^{27}x^{14} + \alpha^{232}x^{13} + \alpha^{201}x^{12} + \alpha^{21}x^{11} + \alpha^{43}x^{10} + \alpha^{245}x^{9} + \alpha^{87}x^{8} + \alpha^{42}x^{7} + \alpha^{195}x^{6} + \alpha^{212}x^{5} + \alpha^{119}x^{4} + \alpha^{242}x^{3} + \alpha^{37}x^{2} + \alpha^{9}x + \alpha^{123}$
30	$x^{30} + \alpha^{41}x^{29} + \alpha^{173}x^{28} + \alpha^{145}x^{27} + \alpha^{152}x^{26} + \alpha^{216}x^{25} + \alpha^{31}x^{24} + \alpha^{179}x^{23} + \alpha^{182}x^{22} + \alpha^{50}x^{21} + \alpha^{48}x^{20} + \alpha^{110}x^{19} + \alpha^{86}x^{18} + \alpha^{239}x^{17} + \alpha^{96}x^{16} + \alpha^{222}x^{15} + \alpha^{125}x^{14} + \alpha^{42}x^{13} + \alpha^{173}x^{12} + \alpha^{226}x^{11} + \alpha^{193}x^{10} + \alpha^{224}x^{9} + \alpha^{130}x^{8} + \alpha^{156}x^{7} + \alpha^{37}x^{6} + \alpha^{251}x^{5} + \alpha^{216}x^{4} + \alpha^{238}x^{3} + \alpha^{40}x^{2} + \alpha^{192}x + \alpha^{180}$
32	$x^{32} + \alpha^{10}x^{31} + \alpha^{6}x^{30} + \alpha^{106}x^{29} + \alpha^{190}x^{28} + \alpha^{249}x^{27} + \alpha^{167}x^{26} + \alpha^{4}x^{25} + \alpha^{67}x^{24} + \alpha^{209}x^{23} + \alpha^{138}x^{22} + \alpha^{138}x^{21} + \alpha^{32}x^{20} + \alpha^{242}x^{19} + \alpha^{123}x^{18} + \alpha^{89}x^{17} + \alpha^{27}x^{16} + \alpha^{120}x^{15} + \alpha^{185}x^{14} + \alpha^{80}x^{13} + \alpha^{156}x^{12} + \alpha^{38}x^{11} + \alpha^{69}x^{10} + \alpha^{171}x^{9} + \alpha^{60}x^{8} + \alpha^{28}x^{7} + \alpha^{222}x^{6} + \alpha^{80}x^{5} + \alpha^{52}x^{4} + \alpha^{254}x^{3} + \alpha^{185}x^{2} + \alpha^{220}x + \alpha^{241}$
34	$x^{34} + \alpha^{111}x^{33} + \alpha^{77}x^{32} + \alpha^{146}x^{31} + \alpha^{94}x^{30} + \alpha^{26}x^{29} + \alpha^{21}x^{28} + \alpha^{108}x^{27}$

	$+ \alpha^{19}x^{26} + \alpha^{105}x^{25} + \alpha^{94}x^{24} + \alpha^{113}x^{23} + \alpha^{193}x^{22} + \alpha^{86}x^{21} + \alpha^{140}x^{20}$ $+ \alpha^{163}x^{19} + \alpha^{125}x^{18} + \alpha^{58}x^{17} + \alpha^{158}x^{16} + \alpha^{229}x^{15} + \alpha^{239}x^{14} + \alpha^{218}x^{13}$ $+ \alpha^{103}x^{12} + \alpha^{56}x^{11} + \alpha^{70}x^{10} + \alpha^{114}x^{9} + \alpha^{61}x^{8} + \alpha^{183}x^{7} + \alpha^{129}x^{6}$ $+ \alpha^{167}x^{5} + \alpha^{13}x^{4} + \alpha^{98}x^{3} + \alpha^{62}x^{2} + \alpha^{129}x + \alpha^{51}$
36	$x^{36} + \alpha^{200}x^{35} + \alpha^{183}x^{34} + \alpha^{98}x^{33} + \alpha^{16}x^{32} + \alpha^{172}x^{31} + \alpha^{31}x^{30} + \alpha^{246}x^{29}$ $+ \alpha^{234}x^{28} + \alpha^{60}x^{27} + \alpha^{152}x^{26} + \alpha^{115}x^{25} + x^{24} + \alpha^{167}x^{23} + \alpha^{152}x^{22}$ $+ \alpha^{113}x^{21} + \alpha^{248}x^{20} + \alpha^{238}x^{19} + \alpha^{107}x^{18} + \alpha^{18}x^{17} + \alpha^{63}x^{16} + \alpha^{218}x^{15}$ $+ \alpha^{37}x^{14} + \alpha^{87}x^{13} + \alpha^{210}x^{12} + \alpha^{105}x^{11} + \alpha^{177}x^{10} + \alpha^{120}x^{9} + \alpha^{74}x^{8}$ $+ \alpha^{121}x^{7} + \alpha^{196}x^{6} + \alpha^{117}x^{5} + \alpha^{251}x^{4} + \alpha^{113}x^{3} + \alpha^{233}x^{2} + \alpha^{30}x + \alpha^{120}$
40	$x^{40} + \alpha^{59}x^{39} + \alpha^{116}x^{38} + \alpha^{79}x^{37} + \alpha^{161}x^{36} + \alpha^{252}x^{35} + \alpha^{98}x^{34} + \alpha^{128}x^{33}$ $+ \alpha^{205}x^{32} + \alpha^{128}x^{31} + \alpha^{161}x^{30} + \alpha^{247}x^{29} + \alpha^{57}x^{28} + \alpha^{163}x^{27} + \alpha^{56}x^{26}$ $+ \alpha^{235}x^{25} + \alpha^{106}x^{24} + \alpha^{53}x^{23} + \alpha^{26}x^{22} + \alpha^{187}x^{21} + \alpha^{174}x^{20} + \alpha^{226}x^{19}$ $+ \alpha^{104}x^{18} + \alpha^{170}x^{17} + \alpha^{7}x^{16} + \alpha^{175}x^{15} + \alpha^{35}x^{14} + \alpha^{181}x^{13} + \alpha^{114}x^{12}$ $+ \alpha^{88}x^{11} + \alpha^{41}x^{10} + \alpha^{47}x^{9} + \alpha^{163}x^{8} + \alpha^{125}x^{7} + \alpha^{134}x^{6} + \alpha^{72}x^{5} + \alpha^{20}x^{4}$ $+ \alpha^{232}x^{3} + \alpha^{53}x^{2} + \alpha^{35}x + \alpha^{15}$
42	$x^{42} + \alpha^{250}x^{41} + \alpha^{103}x^{40} + \alpha^{221}x^{39} + \alpha^{230}x^{38} + \alpha^{25}x^{37} + \alpha^{18}x^{36} + \alpha^{137}x^{35}$ $+ \alpha^{231}x^{34} + x^{33} + \alpha^{3}x^{32} + \alpha^{58}x^{31} + \alpha^{242}x^{30} + \alpha^{221}x^{29} + \alpha^{191}x^{28} + \alpha^{110}x^{27}$ $+ \alpha^{84}x^{26} + \alpha^{230}x^{25} + \alpha^{8}x^{24} + \alpha^{188}x^{23} + \alpha^{106}x^{22} + \alpha^{96}x^{21} + \alpha^{147}x^{20} +$ $\alpha^{15}x^{19} + \alpha^{131}x^{18} + \alpha^{139}x^{17} + \alpha^{34}x^{16} + \alpha^{101}x^{15} + \alpha^{223}x^{14} + \alpha^{39}x^{13} + \alpha^{101}x^{12}$ $+ \alpha^{213}x^{11} + \alpha^{199}x^{10} + \alpha^{237}x^{9} + \alpha^{254}x^{8} + \alpha^{201}x^{7} + \alpha^{123}x^{6} + \alpha^{171}x^{5} +$ $\alpha^{162}x^{4} + \alpha^{194}x^{3} + \alpha^{117}x^{2} + \alpha^{50}x + \alpha^{96}$
44	$x^{44} + \alpha^{190}x^{43} + \alpha^{7}x^{42} + \alpha^{61}x^{41} + \alpha^{121}x^{40} + \alpha^{71}x^{39} + \alpha^{246}x^{38} + \alpha^{69}x^{37} +$ $\alpha^{55}x^{36} + \alpha^{168}x^{35} + \alpha^{188}x^{34} + \alpha^{89}x^{33} + \alpha^{243}x^{32} + \alpha^{191}x^{31} + \alpha^{25}x^{30} + \alpha^{72}x^{29}$ $+ \alpha^{123}x^{28} + \alpha^{9}x^{27} + \alpha^{145}x^{26} + \alpha^{14}x^{25} + \alpha^{247}x^{24} + \alpha x^{23} + \alpha^{238}x^{22} + \alpha^{44}x^{21}$ $+ \alpha^{78}x^{20} + \alpha^{143}x^{19} + \alpha^{62}x^{18} + \alpha^{224}x^{17} + \alpha^{126}x^{16} + \alpha^{118}x^{15} + \alpha^{114}x^{14} +$ $\alpha^{68}x^{13} + \alpha^{163}x^{12} + \alpha^{52}x^{11} + \alpha^{194}x^{10} + \alpha^{217}x^{9} + \alpha^{147}x^{8} + \alpha^{204}x^{7} + \alpha^{169}x^{6} +$ $\alpha^{37}x^{5} + \alpha^{130}x^{4} + \alpha^{113}x^{3} + \alpha^{102}x^{2} + \alpha^{73}x + \alpha^{181}$
46	$x^{46} + \alpha^{112}x^{45} + \alpha^{94}x^{44} + \alpha^{88}x^{43} + \alpha^{112}x^{42} + \alpha^{253}x^{41} + \alpha^{224}x^{40} + \alpha^{202}x^{39} +$ $\alpha^{115}x^{38} + \alpha^{187}x^{37} + \alpha^{99}x^{36} + \alpha^{89}x^{35} + \alpha^{5}x^{34} + \alpha^{54}x^{33} + \alpha^{113}x^{32} + \alpha^{129}x^{3}1$

	$+ \alpha^{44}x^{30} + \alpha^{58}x^{29} + \alpha^{16}x^{28} + \alpha^{135}x^{27} + \alpha^{216}x^{26} + \alpha^{169}x^{25} + \alpha^{211}x^{24} + \alpha^{36}x^{23} + \alpha x^{22} + \alpha^{4}x^{21} + \alpha^{96}x^{20} + \alpha^{60}x^{19} + \alpha^{241}x^{18} + \alpha^{73}x^{17} + \alpha^{104}x^{16} + \alpha^{234}x^{15} + \alpha^{8}x^{14} + \alpha^{249}x^{13} + \alpha^{245}x^{12} + \alpha^{119}x^{11} + \alpha^{174}x^{10} + \alpha^{52}x^{9} + \alpha^{25}x^{8} + \alpha^{157}x^{7} + \alpha^{224}x^{6} + \alpha^{43}x^{5} + \alpha^{202}x^{4} + \alpha^{223}x^{3} + \alpha^{19}x^{2} + \alpha^{82}x + \alpha^{15}$
48	$x^{48} + \alpha^{228}x^{47} + \alpha^{25}x^{46} + \alpha^{196}x^{45} + \alpha^{130}x^{44} + \alpha^{211}x^{43} + \alpha^{146}x^{42} + \alpha^{60}x^{41} + \alpha^{24}x^{40} + \alpha^{251}x^{39} + \alpha^{90}x^{38} + \alpha^{39}x^{37} + \alpha^{102}x^{36} + \alpha^{240}x^{35} + \alpha^{61}x^{34} + \alpha^{178}x^{33} + \alpha^{63}x^{32} + \alpha^{46}x^{31} + \alpha^{123}x^{30} + \alpha^{115}x^{29} + \alpha^{18}x^{28} + \alpha^{221}x^{27} + \alpha^{111}x^{26} + \alpha^{135}x^{25} + \alpha^{160}x^{24} + \alpha^{182}x^{23} + \alpha^{205}x^{22} + \alpha^{107}x^{21} + \alpha^{206}x^{20} + \alpha^{95}x^{19} + \alpha^{150}x^{18} + \alpha^{120}x^{17} + \alpha^{184}x^{16} + \alpha^{91}x^{15} + \alpha^{21}x^{14} + \alpha^{247}x^{13} + \alpha^{156}x^{12} + \alpha^{140}x^{11} + \alpha^{238}x^{10} + \alpha^{191}x^{9} + \alpha^{11}x^{8} + \alpha^{94}x^{7} + \alpha^{227}x^{6} + \alpha^{84}x^{5} + \alpha^{50}x^{4} + \alpha^{163}x^{3} + \alpha^{39}x^{2} + \alpha^{34}x + \alpha^{108}$
50	$x^{50} + \alpha^{232}x^{49} + \alpha^{125}x^{48} + \alpha^{157}x^{47} + \alpha^{161}x^{46} + \alpha^{164}x^{45} + \alpha^{9}x^{44} + \alpha^{118}x^{43} + \alpha^{46}x^{42} + \alpha^{209}x^{41} + \alpha^{99}x^{40} + \alpha^{203}x^{39} + \alpha^{193}x^{38} + \alpha^{35}x^{37} + \alpha^{3}x^{36} + \alpha^{209}x^{35} + \alpha^{111}x^{34} + \alpha^{195}x^{33} + \alpha^{242}x^{32} + \alpha^{203}x^{31} + \alpha^{225}x^{30} + \alpha^{46}x^{29} + \alpha^{13}x^{28} + \alpha^{32}x^{27} + \alpha^{160}x^{26} + \alpha^{126}x^{25} + \alpha^{209}x^{24} + \alpha^{130}x^{23} + \alpha^{160}x^{22} + \alpha^{242}x^{21} + \alpha^{215}x^{20} + \alpha^{242}x^{19} + \alpha^{75}x^{18} + \alpha^{77}x^{17} + \alpha^{42}x^{16} + \alpha^{189}x^{15} + \alpha^{32}x^{14} + \alpha^{113}x^{13} + \alpha^{65}x^{12} + \alpha^{124}x^{11} + \alpha^{69}x^{10} + \alpha^{228}x^{9} + \alpha^{114}x^{8} + \alpha^{235}x^{7} + \alpha^{175}x^{6} + \alpha^{124}x^{5} + \alpha^{170}x^{4} + \alpha^{215}x^{3} + \alpha^{232}x^{2} + \alpha^{133}x + \alpha^{205}$
52	$x^{52} + \alpha^{116}x^{51} + \alpha^{50}x^{50} + \alpha^{86}x^{49} + \alpha^{186}x^{48} + \alpha^{50}x^{47} + \alpha^{220}x^{46} + \alpha^{251}x^{45} + \alpha^{89}x^{44} + \alpha^{192}x^{43} + \alpha^{46}x^{42} + \alpha^{86}x^{41} + \alpha^{127}x^{40} + \alpha^{124}x^{39} + \alpha^{19}x^{38} + \alpha^{184}x^{37} + \alpha^{233}x^{36} + \alpha^{151}x^{35} + \alpha^{215}x^{34} + \alpha^{22}x^{33} + \alpha^{14}x^{32} + \alpha^{59}x^{31} + \alpha^{145}x^{30} + \alpha^{37}x^{29} + \alpha^{242}x^{28} + \alpha^{203}x^{27} + \alpha^{134}x^{26} + \alpha^{254}x^{25} + \alpha^{89}x^{24} + \alpha^{190}x^{23} + \alpha^{94}x^{22} + \alpha^{59}x^{21} + \alpha^{65}x^{20} + \alpha^{124}x^{19} + \alpha^{113}x^{18} + \alpha^{100}x^{17} + \alpha^{233}x^{16} + \alpha^{235}x^{15} + \alpha^{121}x^{14} + \alpha^{22}x^{13} + \alpha^{76}x^{12} + \alpha^{86}x^{11} + \alpha^{97}x^{10} + \alpha^{39}x^{9} + \alpha^{242}x^{8} + \alpha^{200}x^{7} + \alpha^{220}x^{6} + \alpha^{101}x^{5} + \alpha^{33}x^{4} + \alpha^{239}x^{3} + \alpha^{254}x^{2} + \alpha^{116}x + \alpha^{51}$
54	$x^{54} + \alpha^{183}x^{53} + \alpha^{26}x^{52} + \alpha^{201}x^{51} + \alpha^{87}x^{50} + \alpha^{210}x^{49} + \alpha^{221}x^{48} + \alpha^{113}x^{47} + \alpha^{21}x^{46} + \alpha^{46}x^{45} + \alpha^{65}x^{44} + \alpha^{45}x^{43} + \alpha^{50}x^{42} + \alpha^{238}x^{41} + \alpha^{184}x^{40} + \alpha^{249}x^{39} + \alpha^{225}x^{38} + \alpha^{102}x^{37} + \alpha^{58}x^{36} + \alpha^{209}x^{35} + \alpha^{218}x^{34} + \alpha^{109}x^{33} + \alpha^{165}x^{32} + \alpha^{26}x^{31} + \alpha^{95}x^{30} + \alpha^{184}x^{29} + \alpha^{192}x^{28} + \alpha^{52}x^{27} + \alpha^{245}x^{26} + \alpha^{35}x^{25} + \alpha^{254}x^{24}$

	$+ \alpha^{238}x^{23} + \alpha^{175}x^{22} + \alpha^{172}x^{21} + \alpha^{79}x^{20} + \alpha^{123}x^{19} + \alpha^{25}x^{18} + \alpha^{122}x^{17} + \alpha^{43}x^{16} + \alpha^{120}x^{15} + \alpha^{108}x^{14} + \alpha^{215}x^{13} + \alpha^{80}x^{12} + \alpha^{128}x^{11} + \alpha^{201}x^{10} + \alpha^{235}x^{9} + \alpha^{8}x^{8} + \alpha^{153}x^{7} + \alpha^{59}x^{6} + \alpha^{101}x^{5} + \alpha^{31}x^{4} + \alpha^{198}x^{3} + \alpha^{76}x^{2} + \alpha^{31}x + \alpha^{156}$
56	$x^{56} + \alpha^{106}x^{55} + \alpha^{120}x^{54} + \alpha^{107}x^{53} + \alpha^{157}x^{52} + \alpha^{164}x^{51} + \alpha^{216}x^{50} + \alpha^{112}x^{49} + \alpha^{116}x^{48} + \alpha^{2}x^{47} + \alpha^{91}x^{46} + \alpha^{248}x^{45} + \alpha^{163}x^{44} + \alpha^{36}x^{43} + \alpha^{201}x^{42} + \alpha^{202}x^{41} + \alpha^{229}x^{40} + \alpha^{6}x^{39} + \alpha^{144}x^{38} + \alpha^{254}x^{37} + \alpha^{155}x^{36} + \alpha^{135}x^{35} + \alpha^{208}x^{34} + \alpha^{170}x^{33} + \alpha^{209}x^{32} + \alpha^{12}x^{31} + \alpha^{139}x^{30} + \alpha^{127}x^{29} + \alpha^{142}x^{28} + \alpha^{182}x^{27} + \alpha^{249}x^{26} + \alpha^{177}x^{25} + \alpha^{174}x^{24} + \alpha^{190}x^{23} + \alpha^{28}x^{22} + \alpha^{10}x^{21} + \alpha^{85}x^{20} + \alpha^{239}x^{19} + \alpha^{184}x^{18} + \alpha^{101}x^{17} + \alpha^{124}x^{16} + \alpha^{152}x^{15} + \alpha^{206}x^{14} + \alpha^{96}x^{13} + \alpha^{23}x^{12} + \alpha^{163}x^{11} + \alpha^{61}x^{10} + \alpha^{27}x^{9} + \alpha^{196}x^{8} + \alpha^{247}x^{7} + \alpha^{151}x^{6} + \alpha^{154}x^{5} + \alpha^{202}x^{4} + \alpha^{207}x^{3} + \alpha^{20}x^{2} + \alpha^{61}x + \alpha^{10}$
58	$x^{58} + \alpha^{82}x^{57} + \alpha^{116}x^{56} + \alpha^{26}x^{55} + \alpha^{247}x^{54} + \alpha^{66}x^{53} + \alpha^{27}x^{52} + \alpha^{62}x^{51} + \alpha^{107}x^{50} + \alpha^{252}x^{49} + \alpha^{182}x^{48} + \alpha^{200}x^{47} + \alpha^{185}x^{46} + \alpha^{235}x^{45} + \alpha^{55}x^{44} + \alpha^{251}x^{43} + \alpha^{242}x^{42} + \alpha^{210}x^{41} + \alpha^{144}x^{40} + \alpha^{154}x^{39} + \alpha^{237}x^{38} + \alpha^{176}x^{37} + \alpha^{141}x^{36} + \alpha^{192}x^{35} + \alpha^{248}x^{34} + \alpha^{152}x^{33} + \alpha^{249}x^{32} + \alpha^{206}x^{31} + \alpha^{85}x^{30} + \alpha^{253}x^{29} + \alpha^{142}x^{28} + \alpha^{65}x^{27} + \alpha^{165}x^{26} + \alpha^{125}x^{25} + \alpha^{23}x^{24} + \alpha^{24}x^{23} + \alpha^{30}x^{22} + \alpha^{122}x^{21} + \alpha^{240}x^{20} + \alpha^{214}x^{19} + \alpha^{6}x^{18} + \alpha^{129}x^{17} + \alpha^{218}x^{16} + \alpha^{29}x^{15} + \alpha^{145}x^{14} + \alpha^{127}x^{13} + \alpha^{134}x^{12} + \alpha^{206}x^{11} + \alpha^{245}x^{10} + \alpha^{117}x^{9} + \alpha^{29}x^{8} + \alpha^{41}x^{7} + \alpha^{63}x^{6} + \alpha^{159}x^{5} + \alpha^{142}x^{4} + \alpha^{233}x^{3} + \alpha^{125}x^{2} + \alpha^{148}x + \alpha^{123}$
60	$x^{60} + \alpha^{107}x^{59} + \alpha^{140}x^{58} + \alpha^{26}x^{57} + \alpha^{12}x^{56} + \alpha^{9}x^{55} + \alpha^{141}x^{54} + \alpha^{243}x^{53} + \alpha^{197}x^{52} + \alpha^{226}x^{51} + \alpha^{197}x^{50} + \alpha^{219}x^{49} + \alpha^{45}x^{48} + \alpha^{211}x^{47} + \alpha^{101}x^{46} + \alpha^{219}x^{45} + \alpha^{120}x^{44} + \alpha^{28}x^{43} + \alpha^{181}x^{42} + \alpha^{127}x^{41} + \alpha^{6}x^{40} + \alpha^{100}x^{39} + \alpha^{247}x^{38} + \alpha^{2}x^{37} + \alpha^{205}x^{36} + \alpha^{198}x^{35} + \alpha^{57}x^{34} + \alpha^{115}x^{33} + \alpha^{219}x^{32} + \alpha^{101}x^{31} + \alpha^{109}x^{30} + \alpha^{160}x^{29} + \alpha^{82}x^{28} + \alpha^{37}x^{27} + \alpha^{38}x^{26} + \alpha^{238}x^{25} + \alpha^{49}x^{24} + \alpha^{160}x^{23} + \alpha^{209}x^{22} + \alpha^{121}x^{21} + \alpha^{86}x^{20} + \alpha^{11}x^{19} + \alpha^{124}x^{18} + \alpha^{30}x^{17} + \alpha^{181}x^{16} + \alpha^{84}x^{15} + \alpha^{25}x^{14} + \alpha^{194}x^{13} + \alpha^{87}x^{12} + \alpha^{65}x^{11} + \alpha^{102}x^{10} + \alpha^{190}x^{9} + \alpha^{220}x^{8} + \alpha^{70}x^{7} + \alpha^{27}x^{6} + \alpha^{209}x^{5} + \alpha^{16}x^{4} + \alpha^{89}x^{3} + \alpha^{7}x^{2} + \alpha^{33}x + \alpha^{240}$
62	$x^{62} + \alpha^{65}x^{61} + \alpha^{202}x^{60} + \alpha^{113}x^{59} + \alpha^{98}x^{58} + \alpha^{71}x^{57} + \alpha^{223}x^{56} + \alpha^{248}x^{55} + \alpha^{118}x^{54} + \alpha^{214}x^{53} + \alpha^{94}x^{52} + x^{51} + \alpha^{122}x^{50} + \alpha^{37}x^{49} + \alpha^{23}x^{48} + \alpha^{2}x^{47} +$

	$\alpha^{228}x^{46} + \alpha^{58}x^{45} + \alpha^{121}x^{44} + \alpha^{7}x^{43} + \alpha^{105}x^{42} + \alpha^{135}x^{41} + \alpha^{78}x^{40} + \alpha^{243}x^{39} + \alpha^{118}x^{38} + \alpha^{70}x^{37} + \alpha^{76}x^{36} + \alpha^{223}x^{35} + \alpha^{89}x^{34} + \alpha^{72}x^{33} + \alpha^{50}x^{32} + \alpha^{70}x^{31} + \alpha^{111}x^{30} + \alpha^{194}x^{29} + \alpha^{17}x^{28} + \alpha^{212}x^{27} + \alpha^{126}x^{26} + \alpha^{181}x^{25} + \alpha^{35}x^{24} + \alpha^{221}x^{23} + \alpha^{117}x^{22} + \alpha^{235}x^{21} + \alpha^{11}x^{20} + \alpha^{229}x^{19} + \alpha^{149}x^{18} + \alpha^{147}x^{17} + \alpha^{123}x^{16} + \alpha^{213}x^{15} + \alpha^{40}x^{14} + \alpha^{115}x^{13} + \alpha^{6}x^{12} + \alpha^{200}x^{11} + \alpha^{100}x^{10} + \alpha^{26}x^{9} + \alpha^{246}x^{8} + \alpha^{182}x^{7} + \alpha^{218}x^{6} + \alpha^{127}x^{5} + \alpha^{215}x^{4} + \alpha^{36}x^{3} + \alpha^{186}x^{2} + \alpha^{110}x + \alpha^{106}$
64	$x^{64} + \alpha^{45}x^{63} + \alpha^{51}x^{62} + \alpha^{175}x^{61} + \alpha^{9}x^{60} + \alpha^{7}x^{59} + \alpha^{158}x^{58} + \alpha^{159}x^{57} + \alpha^{49}x^{56} + \alpha^{68}x^{55} + \alpha^{119}x^{54} + \alpha^{92}x^{53} + \alpha^{123}x^{52} + \alpha^{177}x^{51} + \alpha^{204}x^{50} + \alpha^{187}x^{49} + \alpha^{254}x^{48} + \alpha^{200}x^{47} + \alpha^{78}x^{46} + \alpha^{141}x^{45} + \alpha^{149}x^{44} + \alpha^{119}x^{43} + \alpha^{26}x^{42} + \alpha^{127}x^{41} + \alpha^{53}x^{40} + \alpha^{160}x^{39} + \alpha^{93}x^{38} + \alpha^{199}x^{37} + \alpha^{212}x^{36} + \alpha^{29}x^{35} + \alpha^{24}x^{34} + \alpha^{145}x^{33} + \alpha^{156}x^{32} + \alpha^{208}x^{31} + \alpha^{150}x^{30} + \alpha^{218}x^{29} + \alpha^{209}x^{28} + \alpha^{4}x^{27} + \alpha^{216}x^{26} + \alpha^{91}x^{25} + \alpha^{47}x^{24} + \alpha^{184}x^{23} + \alpha^{146}x^{22} + \alpha^{47}x^{21} + \alpha^{140}x^{20} + \alpha^{195}x^{19} + \alpha^{195}x^{18} + \alpha^{125}x^{17} + \alpha^{242}x^{16} + \alpha^{238}x^{15} + \alpha^{63}x^{14} + \alpha^{99}x^{13} + \alpha^{108}x^{12} + \alpha^{140}x^{11} + \alpha^{230}x^{10} + \alpha^{242}x^{9} + \alpha^{31}x^{8} + \alpha^{204}x^{7} + \alpha^{11}x^{6} + \alpha^{178}x^{5} + \alpha^{243}x^{4} + \alpha^{217}x^{3} + \alpha^{156}x^{2} + \alpha^{213}x + \alpha^{231}$
66	$x^{66} + \alpha^{5}x^{65} + \alpha^{118}x^{64} + \alpha^{222}x^{63} + \alpha^{180}x^{62} + \alpha^{136}x^{61} + \alpha^{136}x^{60} + \alpha^{162}x^{59} + \alpha^{51}x^{58} + \alpha^{46}x^{57} + \alpha^{117}x^{56} + \alpha^{13}x^{55} + \alpha^{215}x^{54} + \alpha^{81}x^{53} + \alpha^{17}x^{52} + \alpha^{139}x^{51} + \alpha^{247}x^{50} + \alpha^{197}x^{49} + \alpha^{171}x^{48} + \alpha^{95}x^{47} + \alpha^{173}x^{46} + \alpha^{65}x^{45} + \alpha^{137}x^{44} + \alpha^{178}x^{43} + \alpha^{68}x^{42} + \alpha^{111}x^{41} + \alpha^{95}x^{40} + \alpha^{101}x^{39} + \alpha^{41}x^{38} + \alpha^{72}x^{37} + \alpha^{214}x^{36} + \alpha^{169}x^{35} + \alpha^{197}x^{34} + \alpha^{95}x^{33} + \alpha^{7}x^{32} + \alpha^{44}x^{31} + \alpha^{154}x^{30} + \alpha^{77}x^{29} + \alpha^{111}x^{28} + \alpha^{236}x^{27} + \alpha^{40}x^{26} + \alpha^{121}x^{25} + \alpha^{143}x^{24} + \alpha^{63}x^{23} + \alpha^{87}x^{22} + \alpha^{80}x^{21} + \alpha^{253}x^{20} + \alpha^{240}x^{19} + \alpha^{126}x^{18} + \alpha^{217}x^{17} + \alpha^{77}x^{16} + \alpha^{34}x^{15} + \alpha^{232}x^{14} + \alpha^{106}x^{13} + \alpha^{50}x^{12} + \alpha^{168}x^{11} + \alpha^{82}x^{10} + \alpha^{76}x^{9} + \alpha^{146}x^{8} + \alpha^{67}x^{7} + \alpha^{106}x^{6} + \alpha^{171}x^{5} + \alpha^{25}x^{4} + \alpha^{132}x^{3} + \alpha^{93}x^{2} + \alpha^{45}x + \alpha^{105}$
68	$x^{68} + \alpha^{247}x^{67} + \alpha^{159}x^{66} + \alpha^{223}x^{65} + \alpha^{33}x^{64} + \alpha^{224}x^{63} + \alpha^{93}x^{62} + \alpha^{77}x^{61} + \alpha^{70}x^{60} + \alpha^{90}x^{59} + \alpha^{160}x^{58} + \alpha^{32}x^{57} + \alpha^{254}x^{56} + \alpha^{43}x^{55} + \alpha^{150}x^{54} + \alpha^{84}x^{53} + \alpha^{101}x^{52} + \alpha^{190}x^{51} + \alpha^{205}x^{50} + \alpha^{133}x^{49} + \alpha^{52}x^{48} + \alpha^{60}x^{47} + \alpha^{202}x^{46} + \alpha^{165}x^{45} + \alpha^{220}x^{44} + \alpha^{203}x^{43} + \alpha^{151}x^{42} + \alpha^{93}x^{41} + \alpha^{84}x^{40} + \alpha^{15}x^{39} +$

	$\alpha^{84}x^{38} + \alpha^{253}x^{37} + \alpha^{173}x^{36} + \alpha^{160}x^{35} + \alpha^{89}x^{34} + \alpha^{227}x^{33} + \alpha^{52}x^{32} + \alpha^{199}x^{31}$ $+ \alpha^{97}x^{30} + \alpha^{95}x^{29} + \alpha^{231}x^{28} + \alpha^{52}x^{27} + \alpha^{177}x^{26} + \alpha^{41}x^{25} + \alpha^{125}x^{24} +$ $\alpha^{137}x^{23} + \alpha^{241}x^{22} + \alpha^{166}x^{21} + \alpha^{225}x^{20} + \alpha^{118}x^{19} + \alpha^{2}x^{18} + \alpha^{54}x^{17} + \alpha^{32}x^{16}$ $+ \alpha^{82}x^{15} + \alpha^{215}x^{14} + \alpha^{175}x^{13} + \alpha^{198}x^{12} + \alpha^{43}x^{11} + \alpha^{238}x^{10} + \alpha^{235}x^{9} +$ $\alpha^{27}x^{8} + \alpha^{101}x^{7} + \alpha^{184}x^{6} + \alpha^{127}x^{5} + \alpha^{3}x^{4} + \alpha^{5}x^{3} + \alpha^{8}x^{2} + \alpha^{163}x + \alpha^{238}$

ภาคผนวก ฉ

ตารางกำหนดพหุนามการคำนวณ Error Correction Codewords

เลขชี้กำลังของ α เป็น primitive element ของ GF(2,8) ซึ่งลักษณะของ GF(2,8) มีดังนี้

1. $\alpha^{255} = 1$
2. เราสามารถแปลงไปกลับระหว่างค่า exponential ใน α ไปเป็นจำนวนเต็มได้ โดยใช้ตารางต่อไปนี้เพื่อแปลงค่าระหว่างจำนวนเต็มและ exponential ของ α

Exponent of α	Integer
0	1
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	29
9	58
1	116
1	232
12	205
13	135
14	19
15	38
□6	76
17	152
18	45
19	90

Integer	Exponent of α
1	0
2	1
3	25
4	2
5	50
6	26
7	198
8	3
9	223
10	51
11	238
12	27
13	104
14	199
15	75
16	4
17	100
18	224
19	14

20	180	20	52
21	117	21	141
22	234	22	239
23	201	23	129
24	143	24	28
25	3	25	193
26	6	26	105
27	12	27	248
28	24	28	200
29	48	29	8
30	96	30	76
31	192	31	113
32	157	32	5
33	39	33	138
34	78	34	101
35	156	35	47
36	37	36	225
37	74	37	36
38	148	38	15
39	53	39	33
40	106	40	53
41	212	41	147
42	181	42	142
43	119	43	218
44	238	44	240
45	193	45	18
46	159	46	130
47	35	47	69
48	70	48	29

49	140
50	5
51	10
52	20
53	40
54	80
55	160
56	93
57	186
58	105
59	210
60	185
61	111
62	222
63	161
64	95
65	190
66	97
67	194
68	153
69	47
70	94
71	188
72	101
73	202
74	137
75	15
76	30
77	60

49	181
50	194
51	125
52	106
53	39
54	249
55	185
56	201
57	154
58	9
59	120
60	77
61	228
62	114
63	166
64	6
65	191
66	139
67	98
68	102
69	221
70	48
71	253
72	226
73	152
74	37
75	179
76	16
77	145

78	120	78	34
79	240	79	136
80	253	80	54
81	231	81	208
82	211	82	148
83	187	83	206
84	107	84	143
85	214	85	150
86	177	86	219
87	127	87	189
88	254	88	241
89	225	89	210
90	223	90	19
91	163	91	92
92	91	92	131
93	182	93	56
94	113	94	70
95	226	95	64
96	217	96	30
97	175	97	66
98	67	98	182
99	134	99	163
100	17	100	195
101	34	101	72
102	68	102	126
103	136	103	110
104	13	104	107
105	26	105	58
106	52	106	40

107	104		107	84
108	208		108	250
109	189		109	133
110	103		110	186
111	206		111	61
112	129		112	202
113	31		113	94
114	62		114	155
115	124		115	159
116	248		116	10
117	237		117	21
118	199		118	121
119	147		119	43
120	59		120	78
121	118		121	212
122	236		122	229
123	197		123	172
124	151		124	115
125	51		125	243
126	102		126	167
127	204		127	87
128	133		128	7
129	23		129	112
130	46		130	192
131	92		131	247
132	184		132	140
133	109		133	128
134	218		134	99
135	169		135	13

136	79		136	103
137	158		137	74
138	33		138	222
139	66		139	237
140	132		140	49
141	21		141	197
142	42		142	254
143	84		143	24
144	168		144	227
145	77		145	165
146	154		146	153
147	41		147	119
148	82		148	38
149	164		149	184
150	85		150	180
151	170		151	124
152	73		152	17
153	146		153	68
154	57		154	146
155	114		155	217
156	228		156	35
157	213		157	32
158	183		158	137
159	115		159	46
160	230		160	55
161	209		161	63
162	191		162	209
163	99		163	91
164	198		164	149

165	145	165	188
166	63	166	207
167	126	167	205
168	252	168	144
169	229	169	135
170	215	170	151
171	179	171	178
172	123	172	220
173	246	173	252
174	241	174	190
175	255	175	97
176	227	176	242
177	219	177	86
178	171	178	211
179	75	179	171
180	150	180	20
181	49	181	42
182	98	182	93
183	196	183	158
184	149	184	132
185	55	185	60
186	110	186	57
187	220	187	83
188	165	188	71
189	87	189	109
190	174	190	65
191	65	191	162
192	130	192	31
193	25	193	45

194	50
195	100
196	200
197	141
198	7
199	14
200	28
201	56
202	112
203	224
204	221
205	167
206	83
207	166
208	81
209	162
210	89
211	178
212	121
213	242
214	249
215	239
216	195
217	155
218	43
219	86
220	172
221	69
222	138

194	67
195	216
196	183
197	123
198	164
199	118
200	196
201	23
202	73
203	236
204	127
205	12
206	111
207	246
208	108
209	161
210	59
211	82
212	41
213	157
214	85
215	170
216	251
217	96
218	134
219	177
220	187
221	204
222	62

223	9	223	90
224	18	224	203
225	36	225	89
226	72	226	95
227	144	227	176
228	61	228	156
229	122	229	169
230	244	230	160
231	245	231	81
232	247	232	11
233	243	233	245
234	251	234	22
235	235	235	235
236	203	236	122
237	139	237	117
238	11	238	44
239	22	239	215
240	44	240	79
241	88	241	174
242	176	242	213
243	125	243	233
244	250	244	230
245	233	245	231
246	207	246	173
247	131	247	232
248	27	248	116
249	54	249	214
250	108	250	244
251	216	251	234

252	173
253	71
254	142
255	1

252	168
253	80
254	88
255	175

ประวัติผู้เขียนวิทยานิพนธ์

สุพพัต รุ่งเรืองศิลป์ เกิดวันที่ 20 พฤศจิกายน พ.ศ. 2528 มีภูมิลำเนาอยู่ที่จังหวัดนนทบุรี เป็นบุตรคนเดียวของดร.สุรสิทธิ์ รุ่งเรืองศิลป์ และนางสมทรง รุ่งเรืองศิลป์ เข้ารับการศึกษาในระดับประถมศึกษาถึงมัธยมศึกษาตอนต้นที่โรงเรียนสาธิตมหาวิทยาลัยราชภัฏสวนสุนันทาและมัธยมศึกษาตอนปลายที่โรงเรียนสตรีวิทยา

ในปีการศึกษา 2548 เข้าศึกษาในระดับปริญญาบัณฑิตที่มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ คณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมไฟฟ้า สาขาวิศวกรรมคอมพิวเตอร์ และสำเร็จการศึกษาในปีการศึกษา 2551

ในปีการศึกษา 2553 เข้าศึกษาในระดับปริญญามหาบัณฑิตที่จุฬาลงกรณ์มหาวิทยาลัย คณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ สาขาวิศวกรรมคอมพิวเตอร์