



การเพิ่มความจุข้อมูลของรหัสภาพคิวอาร์โค้ดโดยใช้ระดับเทา

โดย

นางสาววาทีตา พันธุมะโน

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
วิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)  
ภาควิชาวิทยาการคอมพิวเตอร์  
คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์  
ปีการศึกษา 2558  
ลิขสิทธิ์ของมหาวิทยาลัยธรรมศาสตร์

การเพิ่มความจุข้อมูลของรหัสภาพคิวอาร์โค้ดโดยใช้ระดับเทา

โดย

นางสาววาทีตา พันธุมะโน



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

วิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)

ภาควิชาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์

ปีการศึกษา 2558

ลิขสิทธิ์ของมหาวิทยาลัยธรรมศาสตร์



# QR Code Capacity Improvement via Gray Scale

BY

MS. WATITA PHANTUMANO



A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF MASTER OF SCIENCE (COMPUTER SCIENCE)

DEPARTMENT OF COMPUTER SCIENCE

FACULTY OF SCIENCE AND TECHNOLOGY

THAMMASAT UNIVERSITY

ACADEMIC YEAR 2015

COPYRIGHT OF THAMMASAT UNIVERSITY

มหาวิทยาลัยธรรมศาสตร์  
คณะวิทยาศาสตร์และเทคโนโลยี

วิทยานิพนธ์

ของ

นางสาววาทีตา พันธะโม

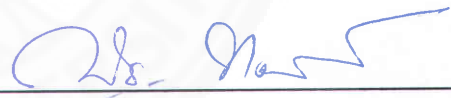
เรื่อง

การเพิ่มความจุข้อมูลของรหัสภาพคิวอาร์โค้ดโดยใช้ระดับเทา

ได้รับการตรวจสอบและอนุมัติ ให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
วิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)

เมื่อวันที่ 14 กรกฎาคม พ.ศ. 2559

ประธานกรรมการสอบวิทยานิพนธ์



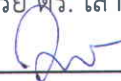
(ผู้ช่วยศาสตราจารย์ ดร. ณัฐชนน หงส์วาทิธีธร)

กรรมการและอาจารย์ที่ปรึกษาวิทยานิพนธ์



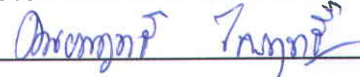
(ผู้ช่วยศาสตราจารย์ ดร. เสาวลักษณ์ วรรณภา)

กรรมการสอบวิทยานิพนธ์



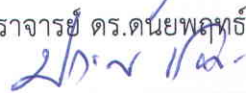
(ผู้ช่วยศาสตราจารย์ ดร. วิรัตน์ จาริวงษ์ไพบูลย์)

กรรมการสอบวิทยานิพนธ์



(รองศาสตราจารย์ ดร. ดนัยพุดท์ ไกรฤทธิ)

คณบดี



(รองศาสตราจารย์ ปกรณ์ เสริมสุข)

หัวข้อวิทยานิพนธ์	การเพิ่มความจุข้อมูลของรหัสภาพคิวอาร์โค้ดโดยใช้ระดับเทา
ชื่อผู้เขียน	นางสาววาทีตา พันธุ์มะโน
ชื่อปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา/คณะ/มหาวิทยาลัย	สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ผู้ช่วยศาสตราจารย์ ดร. เสาวลักษณ์ วรรณภา
ปีการศึกษา	2558

### บทคัดย่อ

ปัจจุบันรหัสคิวอาร์โค้ดนิยมใช้อย่างแพร่หลาย ทั้งในด้านอุตสาหกรรมการผลิต ด้านการบริหารจัดการ ด้านการบริการสาธารณสุข แต่เนื่องด้วยความสามารถในการบรรจุข้อมูลของรหัสคิวอาร์โค้ดนั้นมีจำกัด รวมทั้งหากข้อมูลที่ต้องการนำมาเข้ารหัสคิวอาร์โค้ดมีปริมาณที่มากจะทำให้ภาพรหัสคิวอาร์โค้ดที่เข้ารหัสนั้นใหญ่มากตามไปด้วย ซึ่งไม่เป็นผลดีต่อการนำมาใช้ในพื้นที่ที่ถูกลimit

ทางผู้พัฒนาจึงได้คิดค้นหาวิธีเพื่อให้รหัสคิวอาร์โค้ดสามารถบรรจุข้อมูลปริมาณที่มากขึ้นในพื้นที่ที่จำกัด โดยการประยุกต์ใช้ระดับเทาพร้อมกับการเข้าและถอดรหัสคิวอาร์โค้ดตามมาตรฐาน ISO ที่ใช้กันอย่างแพร่หลายในปัจจุบัน เพื่อให้รหัสคิวอาร์โค้ดสามารถนำไปใช้ประโยชน์ในด้านต่างๆ ได้มากขึ้น และเพื่อสามารถนำไปพัฒนาต่อยอดได้อีกมากมาย

**คำสำคัญ:** รหัสภาพคิวอาร์โค้ด, การแทนที่ข้อมูล, รหัสแท่งสองมิติ

Thesis Title	QR Code Capacity Improvement via Gray Scale
Author	Ms. Watita Phantumano
Degree	Master of Science
Major Field/Faculty/University	Computer Science Faculty of Science and Technology Thammasat University
Thesis Advisor	Asst. Prof. Dr. Saowaluk Watanapa
Academic Years	2015

### ABSTRACT

Nowadays, A QR code is also widely used in a manufacture industry, a management, and a public health service. Because of the limited ability of containing the QR code data with a plenty of the required encryption data, it could make a picture of the QR code getting bigger than it should be. So, this is not an advantage of using the QR code in the limited space.

The developer found a way to contain a plenty of data in a limited space of the QR code by applying a gray scale for encode and decode a QR code in standard. Therefore, for getting more benefit from the QR code, and also develop it in many ways.

**Keywords:** QR Code, Representation, two-dimensional code

## กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้เป็นอย่างดี เนื่องจากได้รับความอนุเคราะห์จากผู้ช่วยศาสตราจารย์ ดร.เสาวลักษณ์ วรรณานา ซึ่งเป็นอาจารย์ที่ปรึกษาและควบคุมวิทยานิพนธ์ เป็นผู้ให้คำปรึกษาและแนะนำอย่างอ่อนโยน ตลอดจนให้กำลังใจแก่ผู้วิจัยเสมอมา ผู้วิจัยซาบซึ้งในความกรุณา และขอขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้ด้วย

ขอขอบคุณ ผู้ช่วยศาสตราจารย์ ดร. ณัฐชนนท์ หงส์วิทธิธร ประธานกรรมการสอบวิทยานิพนธ์ ผู้ช่วยศาสตราจารย์ ดร. วิรัตน์ จาริวงศ์ไพบูลย์ และ รองศาสตราจารย์ ดร. ดนยพฤทธิ์ ไกรฤทธิ์ กรรมการสอบวิทยานิพนธ์ ที่ให้ข้อเสนอแนะในการปรับปรุงแก้ไขวิทยานิพนธ์ให้มีความสมบูรณ์ยิ่งขึ้น

ขอบคุณหัวหน้างานบริษัท ยิบอินซอย จำกัด คุณวรรณิ์ นิลุมล คุณพิชิต มุสิกบุตร และคุณนวลจันทร์ เทียงทัศน์ ที่อนุญาตให้ทำงานเพื่อให้มาทำวิจัยได้อย่างเต็มที่ ส่งผลให้งานวิจัยมีความคืบหน้า รวมทั้งเพื่อนร่วมงานที่ทำหน้าที่รับผิดชอบงานต่อจากผู้วิจัยในช่วงที่ลางาน และเพื่อนร่วมงานคนอื่นๆที่คอยให้กำลังใจเสมอมา

ขอขอบคุณครอบครัว คุณพ่อ คุณแม่ พี่สาวและญาติๆ ที่ผลักดันให้เรียนและให้กำลังใจแก่ผู้วิจัยเสมอมา รวมถึงพี่หมีที่ให้การช่วยเหลือผู้วิจัยทุกอย่าง ทั้งคำปรึกษา คำแนะนำ รวมถึงให้กำลังใจมาโดยตลอด

นางสาววาทีตา พันธะโน

## สารบัญ

	หน้า
บทคัดย่อภาษาไทย	(1)
บทคัดย่อภาษาอังกฤษ	(2)
กิตติกรรมประกาศ	(3)
สารบัญตาราง	(5)
สารบัญภาพ	(7)
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของงานวิจัย	2
1.3 ขอบเขตงานวิจัย	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	3
บทที่ 2 วรรณกรรมและงานวิจัยที่เกี่ยวข้อง	4
2.1 ทฤษฎีที่เกี่ยวข้อง	4
2.1.1 รหัสแท่งสองมิติ (Two-Dimension Barcode)	4
2.1.2 รหัสรีดโซโลมอน (Reed-Solomon)	13
2.1.3 ภาพระดับเทา (Grayscale Image)	15
2.1.4 โมเดลสี RGB	15
2.1.5 การคำนวณระยะทางแบบยูคลิด (Euclidean distance)	16
2.2 งานวิจัยที่เกี่ยวข้อง	16



บทที่ 3 วิธีการวิจัย	28
3.1 แนวคิดและวิธีดำเนินงานวิจัย	28
3.2 ขั้นตอนการวิจัย	29
3.2.1 การเข้ารหัสคิวอาร์โค้ดระดับเทา	29
3.2.2 การถอดรหัสคิวอาร์โค้ดระดับเทา	33
3.3 การทดสอบประสิทธิภาพระบบเบื้องต้น	35
บทที่ 4 ผลการวิจัยและอภิปรายผล	45
4.1 ผลการทดลองวัดความสามารถในการถอดรหัสคิวอาร์โค้ดมาตรฐาน	45
4.2 ผลการทดลองเปรียบเทียบค่าองค์ประกอบสี RGB ของคิวอาร์โค้ดระดับเทา และแบบสีที่เปลี่ยนแปลงไปเมื่อถ่ายภาพในสภาพแสงสีขาวและแสงสีส้ม	46
4.3 ผลการทดลองจำนวนระดับเทาที่เหมาะสมในการใช้งานคิวอาร์โค้ดระดับเทา	49
4.4 ผลการทดลองเพื่อหาความสามารถจุข้อมูลของคิวอาร์โค้ดระดับเทาเปรียบเทียบกับ คิวอาร์โค้ดมาตรฐาน	51
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ	56
5.1 สรุปการศึกษาวิจัย	56
5.2 สรุปผลการทดลอง	57
5.3 ข้อเสนอแนะและแนวทางวิจัยต่อไปในอนาคต	58
รายการอ้างอิง	59
ภาคผนวก	
ภาคผนวก ก.	61
ภาคผนวก ข.	63

ภาคผนวก ค.

หน้า

69

ภาคผนวก ง.

77

ภาคผนวก จ.

83



## สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงชนิดของข้อมูลและความจุที่สามารถเก็บข้อมูลได้ของรหัสแท่งสองมิติ ชนิดคิวอาร์โค้ด	5
2.2 ชนิดของข้อมูลแทนด้วยเลขฐาน 2	8
2.3 จำนวนบิตที่ต้องแปลงตามเวอร์ชัน	9
2.4 จำนวนชุดข้อมูลบิตของคิวอาร์โค้ดมาตรฐานเวอร์ชันหนึ่ง	11
2.5 ชุดระดับความเข้มแสงแต่ละชุดสำหรับทดสอบ	19
2.6 ผลการทดลองประสิทธิภาพการถอดรหัสดิจิทัลที่ถ่ายจากกระดาษ และจากจอภาพ ด้วย Decoder สำหรับคิวอาร์โค้ดมาตรฐาน	20
2.7 ปริมาณความจุข้อมูลของรหัสดิจิทัลแต่ละประเภท ที่พิมพ์ละเอียด 600 DPI	23
2.8 การใช้งานของรหัสดิจิทัลชนิด HCC2D ในความละเอียดที่ต่างกัน และรุ่นเวอร์ชันที่ต่างกัน	24
2.9 แสดงเลขฐานสองและค่าระดับเทา	25
2.10 สรุปรงานวิจัยที่เกี่ยวข้อง	27
3.1 ชนิดของข้อมูลแทนด้วยเลขฐาน 2	31
3.2 ค่า RGB ของแต่ละสี/ระดับเทาที่ใช้ในการทดลอง	38
3.3 ค่า RGB ของระดับเทา 4 8 และ 16 ระดับเทาที่ใช้ในการทดลอง	41
4.1 ผลการถอดรหัสดิจิทัลมาตรฐานระดับความผิดพลาด L ที่ถ่ายได้จากสมาร์ทโฟน	46
4.2 ค่าองค์ประกอบสี RGB ที่เปลี่ยนแปลงไปของสีที่ใช้ในคิวอาร์โค้ดและระดับเทา ที่ใช้ในคิวอาร์โค้ดระดับเทา จากภาพที่ถ่ายจากสมาร์ทโฟน	47
4.3 ค่า Paired T-Test ระหว่างความแตกต่างขององค์ประกอบ R G B ของสี และระดับเทาในสภาพแสงสีขาวและแสงสีส้ม	48
4.4 ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 4 ระดับจากภาพ ที่ถ่ายโดยสมาร์ทโฟน A	49
4.5 ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 4 ระดับจากภาพ ที่ถ่ายโดยสมาร์ทโฟน B	50
4.6 ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 4 ระดับจากภาพ ที่ถ่ายโดยสมาร์ทโฟน C	50

ตารางที่	หน้า
4.7 ผลการทดลองจำนวนระดับเทาที่เหมาะสมสำหรับคิวอาร์โค้ดระดับเทา	51
4.8 สรุปผลการทดลองความจุที่เพิ่มขึ้นเมื่อเข้ารหัสคิวอาร์โค้ดระดับเทา 4 ระดับ เทียบกับความจุของคิวอาร์โค้ดมาตรฐาน	52



## สารบัญภาพ

ภาพที่	หน้า
2.1 การเปรียบเทียบรหัสแท่งสองมิติกับรหัสแท่งหนึ่งมิติ	4
2.2 ตัวอย่างของรหัสแท่งสองมิติชนิดคิวอาร์โค้ดในแต่ละรุ่น	6
2.3 โครงสร้างของคิวอาร์โค้ด	7
2.4 คำรหัสของอักขระและตัวเลขใน Alphanumeric Mode	9
2.5 ทิศทางการบันทึกข้อมูลในคิวอาร์โค้ดของเวอร์ชัน 1	12
2.6 หน้ากากของคิวอาร์โค้ดเวอร์ชัน 1	13
2.7 ระดับสีเทา	15
2.8 โมเดลสี	16
2.9 สมการสูตรการคำนวณระยะทางแบบยูคลิด	16
2.10 ผลลัพธ์ของรหัสคิวอาร์โค้ดสี	17
2.11 บริเวณแถบที่บอกสีที่ใช้	21
2.12 ภาพผลลัพธ์คิวอาร์โค้ดสี (HCC2D) แบบ 4 สี (a) และ 16 (b) สี	22
2.13 ภาพผลลัพธ์คิวอาร์โค้ดระดับเทา	25
3.1 การเข้ารหัสคิวอาร์โค้ดให้อยู่ในรูปแบบระดับเทา	29
3.2 การเข้ารหัสคิวอาร์โค้ดให้อยู่ในรูปแบบระดับเทา (ต่อ)	30
3.3 ตัวอย่างภาพคิวอาร์โค้ดระดับเทา 4 ระดับ	32
3.4 การถอดรหัสคิวอาร์โค้ดให้อยู่ในรูปแบบระดับเทา	33
3.5 สมการระยะห่างของแต่ละระดับเทา	37
3.6 ลักษณะของกระดาษที่ใช้ในการทดลองที่ 2	37
3.7 ตัวอย่างการคำนวณระยะห่างแบบยูคลิดของค่า RGB ที่วัดได้กับค่า RGB ตั้งต้น	39
3.8 ตัวอย่างการเฉลี่ยแต่ละองค์ประกอบสี	39
3.9 ตัวอย่างการหาระยะห่างแบบยูคลิดแต่ละระดับเทาและระยะห่างแบบยูคลิดเฉลี่ยของระดับเทาทั้งหมด	40
3.10 ลักษณะของกระดาษที่ใช้ในการทดลองที่ 3	41
3.11 ตัวอย่างการหาค่าเฉลี่ยมากที่สุดและค่าเฉลี่ยน้อยที่สุด	42
3.12 ตัวอย่างการทดลองหาความจุกิวอาร์โค้ดระดับเทาที่ระดับแก้ไขข้อผิดพลาด L	43

ภาพที่	หน้า
4.1 แสดงการเปรียบเทียบค่าองค์ประกอบสี RGB ที่เปลี่ยนแปลงไปของสีและระดับเทา จากภาพถ่ายจากกล้องสมาร์ทโฟน ในสภาพแสงสีขาว	47
4.2 แสดงการเปรียบเทียบค่าองค์ประกอบสี RGB ที่เปลี่ยนแปลงไปของสีและระดับเทา จากภาพถ่ายจากกล้องสมาร์ทโฟน ในสภาพแสงสีส้ม	48
4.3 ความจุข้อมูลคิวอาร์โค้ดมาตรฐานและคิวอาร์โค้ดระดับเทาที่ระดับแก้ไขข้อผิดพลาด L	52
4.4 ความจุข้อมูลคิวอาร์โค้ดมาตรฐานและคิวอาร์โค้ดระดับเทาที่ระดับแก้ไขข้อผิดพลาด M	53
4.5 ความจุข้อมูลคิวอาร์โค้ดมาตรฐานและคิวอาร์โค้ดระดับเทาที่ระดับแก้ไขข้อผิดพลาด Q	53
4.6 ความจุข้อมูลคิวอาร์โค้ดมาตรฐานและคิวอาร์โค้ดระดับเทาที่ระดับแก้ไขข้อผิดพลาด H	54



## บทที่ 1

### บทนำ

#### 1.1 ที่มาและความสำคัญของปัญหา

สื่อประชาสัมพันธ์ ในปัจจุบันมีอิทธิพลต่อผู้คนที่พบเห็นค่อนข้างสูง และสามารถพบเห็นได้ตามที่สาธารณะทั่วไป ซึ่งประกอบด้วย ข้อความ และภาพปรากฏอยู่ รวมถึง รหัสคิวอาร์โค้ด (Quick Response Code) อยู่บนสื่อต่างๆ และสามารถพบเห็นได้ทั่วไป อีกทั้งนวัตกรรมทางด้านเทคโนโลยี โทรศัพท์เคลื่อนที่ปัจจุบันมีกล้องถ่ายรูปสามารถอ่านข้อมูลต่างๆ ที่บันทึกอยู่บนรหัสคิวอาร์โค้ดได้อย่างรวดเร็ว และแม่นยำ จึงทำให้รหัสคิวอาร์โค้ดได้รับความนิยมอย่างมากในด้านการตลาดของประเทศไทย โดยมักจะเห็นในป้ายโฆษณาตามสถานที่สาธารณะต่างๆ ไม่ว่าจะเป็นป้ายโฆษณาบนสถานีรถไฟฟ้า รวมถึงใบปลิวที่แจกกันโดยทั่วไป แม้แต่ในสื่อโทรทัศน์ หรือบนเว็บไซต์ต่างๆ

รหัสคิวอาร์โค้ดมาประยุกต์ใช้งานในด้านต่างๆ มากมาย เช่น ด้านอุตสาหกรรมการผลิตใช้ในการกำกับสินค้าเพื่อเพิ่มระบุข้อมูลต่างๆ เช่น ชื่อสินค้า ราคา และแหล่งผลิต รวมถึงบนชิ้นส่วนอิเล็กทรอนิกส์ รหัสคิวอาร์โค้ดสามารถระบุวิธีการประกอบชิ้นส่วนต่างๆ เข้าด้วยกันได้ ด้านการบริหารจัดการ เช่น การใช้บันทึกข้อมูลการตรวจคนเข้าเมืองให้มีความแม่นยำและสะดวกเร็วมากยิ่งขึ้น ด้านการบริการสาธารณสุข ใช้ระบุข้อมูลคนไข้ในโรงพยาบาลลดข้อผิดพลาดในการรักษา เป็นต้น

รหัสคิวอาร์โค้ด เป็นรูปภาพที่มีลักษณะเป็นรหัสแท่งสองมิติประเภทหนึ่ง สร้างในปี พ.ศ. 2537 โดย บริษัทเดนมาร์ก-เวฟ รหัสคิวอาร์โค้ดที่ใช้กันอย่างแพร่หลายในปัจจุบัน ซึ่งความสามารถในการบันทึกข้อมูลของรหัสคิวอาร์โค้ดถูกจำกัด โดยรหัสคิวอาร์โค้ดสามารถบรรจุข้อมูลตัวอักษร (Alphanumeric) ซึ่งรูปแบบของรหัสคิวอาร์โค้ดตามมาตรฐานนั้นมีตั้งแต่ 1 - 40 เวอร์ชัน และเวอร์ชันที่ 40 นั้นเป็นเวอร์ชันที่สามารถบรรจุได้มากที่สุดซึ่งสามารถบรรจุได้ 4,296 ตัวอักษร แต่อีกปัญหาหนึ่งของการบรรจุข้อมูลปริมาณมากนั้นจะมาพร้อมกับขนาดของภาพรหัสคิวอาร์โค้ดที่ใหญ่ ซึ่งไม่เป็นผลดีต่อการนำมาใช้ในพื้นที่จำกัด จึงทำให้เกิดงานวิจัยมากมายที่ต้องการให้รหัสคิวอาร์โค้ดมีขนาดเล็กลงและสามารถบรรจุข้อมูลได้ปริมาณที่มากขึ้น โดยเทคนิคหนึ่งที่ถูกเสนอขึ้นมาคือการนำสี มาช่วยเพิ่มการบรรจุข้อมูลในรหัสคิวอาร์โค้ดให้มากขึ้น

ตัวอย่างงานวิจัยเช่นงานวิจัยที่เกี่ยวข้อง ได้แก่ "A New Color QR Code Forward Compatible with the Standard QR Code Decoder." (Kikuchi, Fujiyoshi & Kiya, 2013) ได้ทำการวิจัยวิธีเพิ่มความจุของคิวอาร์โค้ด โดยการนำสีมาใช้ ซึ่งสามารถช่วยเพิ่มความจุได้มากขึ้น

ประมาณสามเท่าของรหัสคิวอาร์โค้ดมาตรฐาน อีกทั้งในส่วนของการถอดรหัสข้อความ (Decode) สามารถใช้วิธีเดียวกับแบบ QR Code มาตรฐานได้ รวมถึงงานวิจัย "High capacity colored two dimensional codes." (Grillo, Lentini & Italiano, 2010) ได้ทำการเพิ่มความจุด้วยวิธีการที่ทำให้หนึ่งหน่วยที่เล็กที่สุดของรหัสคิวอาร์โค้ดสามารถเก็บข้อมูลได้มากกว่าหนึ่งบิต โดยการนำสีเข้ามาช่วยเช่นกัน วิธีนี้สามารถช่วยเพิ่มความจุได้ตามจำนวนบิตที่เพิ่มขึ้นในแต่ละหน่วย

เทคนิคข้างต้นยังคงมีปัญหาอยู่ นั่นคือ ค่าใช้จ่ายที่สูงขึ้น ความคมชัดของภาพและสีซึ่งขึ้นอยู่กับคุณภาพของเครื่องพิมพ์รวมถึงคุณภาพของหมึกสีแต่ละยี่ห้อ หรือแม้แต่สภาพแวดล้อม แสง ซึ่งมีส่วนที่ทำให้สีที่ได้ไม่เหมือนกัน ส่งผลให้การถอดรหัสคิวอาร์โค้ดผิดพลาดได้ง่าย

งานวิจัยนี้นำเสนอวิธีการที่สามารถเพิ่มการบรรจุข้อมูลของคิวอาร์โค้ด โดยนำระดับเทา (Gray Scale) เข้ามาปรับปรุงเพื่อลดความผิดพลาดในการถอดรหัสคิวอาร์โค้ด เนื่องจากการพิมพ์ภาพที่เป็นภาพระดับเทาไม่จำเป็นต้องนำสีมาผสมกันจึงทำให้ภาพที่ได้สีไม่ผิดเพี้ยนมากนัก รวมถึงลดค่าใช้จ่ายในการพิมพ์รหัสคิวอาร์โค้ดได้

## 1.2 วัตถุประสงค์ของงานวิจัย

ทำการศึกษาและวิจัยแนวทางพัฒนาเพื่อเพิ่มความสามารถของรหัสคิวอาร์โค้ดให้สามารถบรรจุข้อมูลได้มากขึ้นจากรหัสคิวอาร์โค้ดมาตรฐาน โดยใช้ระดับเทา โดยพัฒนาจาก Algorithm ของรหัสคิวอาร์โค้ดมาตรฐาน

## 1.3 ขอบเขตงานวิจัย

1.3.1 ศึกษาหาวิธีการเข้ารหัสคิวอาร์โค้ดด้วยระดับเทา เพื่อเพิ่มความสามารถในการบรรจุข้อมูลของรหัสคิวอาร์โค้ด โดยปรับปรุงอัลกอริธึม จากการเข้ารหัสคิวอาร์โค้ดมาตรฐาน

1.3.2 ศึกษาหาวิธีการถอดรหัสคิวอาร์โค้ดที่ได้จากการเข้ารหัสด้วยระดับเทา โดยปรับปรุงอัลกอริธึม จากการถอดรหัสคิวอาร์โค้ดมาตรฐาน



#### 1.4 ประโยชน์ที่คาดว่าจะได้รับ

สามารถเพิ่มความจุในการบันทึกข้อความลงในรหัสคิวอาร์โค้ดระดับเทาได้มากกว่ารหัสคิวอาร์โค้ดมาตรฐานเมื่อขนาดของภาพเท่ากัน หรือภาพของรหัสคิวอาร์โค้ดระดับเทาที่มีขนาดเล็กกว่ารหัสคิวอาร์โค้ดมาตรฐาน เมื่อบรรจุข้อมูลเท่ากัน และสามารถจุข้อมูลได้เท่ากับหรือใกล้เคียงกับคิวอาร์โค้ดสีที่จำนวนสี/ระดับเทาเท่ากัน รวมถึงทราบประสิทธิภาพและข้อจำกัดของรหัสคิวอาร์โค้ดระดับเทา



## บทที่ 2

### วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

#### 2.1 ทฤษฎีที่เกี่ยวข้อง

##### 2.1.1 รหัสแท่งสองมิติ (Two-Dimension Barcode)

รหัสแท่งสองมิติ (Gao, Prakash & Jagatesan, 2007) เป็นรหัสแท่งที่ประกอบไปด้วยข้อมูลทั้งแนวนอนและแนวตั้ง สามารถเก็บข้อมูลได้ในปริมาณที่มากกว่ารหัสแท่งแบบหนึ่งมิติ (one-dimension barcode) และใช้พื้นที่ในการจัดเก็บน้อยกว่าหรือมีขนาดเล็กกว่ามาก เนื่องจากรหัสแท่งสองมิตินั้นสามารถเก็บข้อมูลได้ในสองทิศทาง นอกจากนี้รหัสแท่งสองมิติโดยทั่วไปจะมีการจัดการกับความผิดพลาดและการแก้ไขความผิดพลาดของข้อมูล หากรหัสแท่งสองมิตินั้นเกิดฉีกขาดหรือถูกทำลายไปบางส่วนก็ยังสามารถที่จะอ่านข้อมูลได้อย่างถูกต้อง ดังภาพที่ 2.1 แสดงลักษณะและรูปแบบของรหัสแท่งสองมิติสองชนิดคือ คิวอาร์โค้ด และ PDF417 Code (Rong, Zhen-ya, Yan-hu, Yi & Li-yu, 2011) เมื่อเปรียบเทียบกับรหัสแท่งหนึ่งมิติในปัจจุบัน



ภาพที่ 2.1 การเปรียบเทียบรหัสแท่งสองมิติกับรหัสแท่งหนึ่งมิติ

ในปัจจุบันรหัสแท่งสองมิตินั้นได้รับความนิยมเป็นอย่างมาก เนื่องจากความสามารถในการอ่านข้อมูลได้อย่างรวดเร็วและด้วยความสามารถในการเก็บข้อมูลได้ในปริมาณมาก โดยเฉพาะรหัสคิวอาร์โค้ด

### คิวอาร์โค้ด (QR Code : Quick Response Code)

คิวอาร์โค้ดเป็นสัญลักษณ์ สองมิติ ประดิษฐ์ขึ้นในปี 1994 โดยบริษัท เดนโซ ซึ่งเป็นส่วนหนึ่งของบริษัท โตโยต้าและผ่านการตรวจสอบจากมาตรฐานสากล ISO (ISO/IEC18004) โดยโครงสร้างคิวอาร์โค้ดเป็นสัญลักษณ์เมทริกซ์ มีโครงสร้างเซลล์จัดเรียงในช่องสี่เหลี่ยม ประกอบด้วยรูปแบบการทำงานสำหรับการทำให้อ่านง่ายและพื้นที่ข้อมูลที่จัดเก็บข้อมูล ลักษณะการทำงานของคิวอาร์โค้ดเป็นรหัสแบบเมทริกซ์ ใช้ในการบันทึกข้อมูลขนาดใหญ่ โดยมีลักษณะเป็นรูปสี่เหลี่ยมจัตุรัสสามารถอ่านข้อมูลด้วยความเร็วสูงในทุกทิศทางโดยอ่านข้อมูลจากการสแกนภาพแล้วจัดเก็บไว้ใน หน่วยความจำ ทำให้สามารถอ่านข้อมูลด้วยความเร็วสูงใน ทุกทิศทาง 360 องศา ความสามารถในการจุข้อมูลของคิวอาร์โค้ดแสดงดังตาราง 2.1 ดังนี้

#### ตารางที่ 2.1

แสดงชนิดของข้อมูลและความจุที่สามารถเก็บข้อมูลได้ของรหัสแท่งสองมิติ ชนิดคิวอาร์โค้ด

ชนิดข้อมูล	ความจุสูงสุด	ข้อมูลที่ใช้
ตัวเลข	7,089 ตัวอักษร	0 – 9
ตัวอักษร	4,296 ตัวอักษร	0-9 , A-Z (ตัวพิมพ์ใหญ่เท่านั้น)
ไบนารี (8 บิต)	2,953 ไบต์	ISO-88991

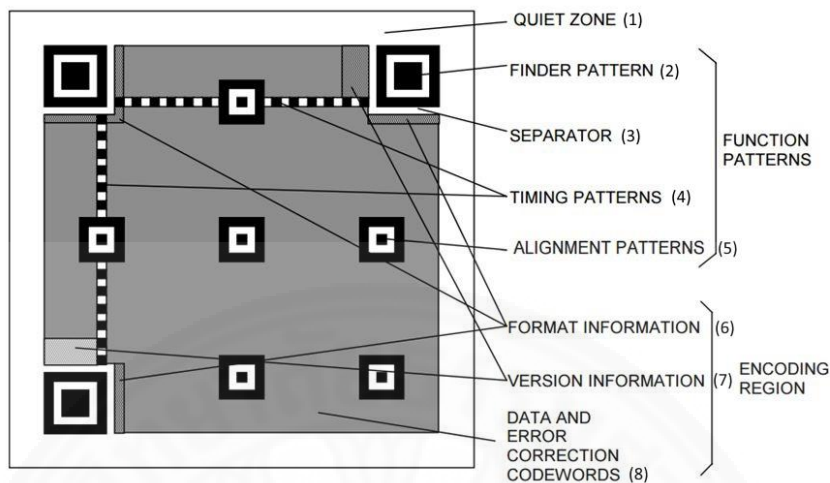
รหัสแท่งสองมิติชนิดคิวอาร์โค้ดนั้นปัจจุบันมีตั้งแต่เวอร์ชัน 1 ถึง เวอร์ชัน 40 ซึ่งแต่ละรุ่นนั้นจะมี จำนวนจุดพิกเซลบนรหัสแท่งที่แตกต่างกัน ซึ่งจะเรียกว่าโมดูล (module) ประกอบด้วยจุดพิกเซลสี่ด้านบนพื้นสีขาวที่อยู่บนคิวอาร์โค้ด โดยคิวอาร์โค้ดเวอร์ชัน 1 นั้นจะมีจำนวนโมดูลทั้งหมดประกอบด้วย 21 คอลัมน์และ 21 แถวและเวอร์ชัน 40 มีจำนวนโมดูลทั้งหมดประกอบด้วย 177 คอลัมน์และ 177แถว ดังภาพ 2.2



ภาพที่ 2.2 ตัวอย่างของรหัสแท่งสองมิติชนิดคิวอาร์โค้ดในแต่ละรุ่น

อีกสิ่งหนึ่งที่ต้องให้ความสำคัญคือการป้องกันความเสียหาย ของข้อมูลแบ่งเป็น 4 ระดับคือ L, M, Q และ H โดยสามารถป้องกันความเสียหายได้ 7%, 15%, 25% และ 30% ตามลำดับ ต่อพื้นที่สัญลักษณ์การแก้ไขข้อผิดพลาดจะมีฟังก์ชันการทำงาน จะถูกนำมาใช้ตาม จุดข้อบกพร่องหรือพื้นที่ ความเสียหายทันที โดยจะใช้รหัสรีดโซโลมอนเพื่อป้องกันการความผิดพลาดนั้น โดยรหัสรีดโซโลมอนจะจัดเรียงในพื้นที่ข้อมูลคิวอาร์โค้ดใหม่ โดยการแก้ไขข้อผิดพลาด ณ จุดนั้นๆ ซึ่งฟังก์ชันการทำงาน ทำให้รหัสยังคงสามารถอ่านได้อย่างถูกต้องแม้ว่าจะมีจุดหรือรอยเปื้อนหรือความเสียหายที่เกิดขึ้นจนกว่าจะมีการแก้ไข ข้อผิดพลาดนั้นเสร็จสิ้น ระดับการแก้ไขข้อผิดพลาดสามารถถูกกำหนดค่าโดยผู้ใช้เมื่อผู้ใช้ได้สร้างสัญลักษณ์ขึ้น ถ้าหากรหัสมี ความละเอียดมากโอกาสที่จะได้รับความเสียหายขึ้นจากการใช้งานไม่ควรมีข้อผิดพลาดเกิน 30%

## โครงสร้างของคิวอาร์โค้ด



ภาพที่ 2.3 โครงสร้างของคิวอาร์โค้ด

### ประกอบด้วย

1. Quiet Zone เป็นพื้นที่ขอบของคิวอาร์โค้ด
2. Finder Pattern สำหรับการตรวจหาตำแหน่งในการอ่านข้อมูล
3. Separator สำหรับแยก Finder Pattern กับสัญลักษณ์ที่เหลือ
4. Timing Patterns สำหรับการตรวจตำแหน่งของแต่ละโมดูล
5. Alignment Patterns สำหรับการแก้ไขการบิดเบือนของรหัสคิวอาร์โค้ด (QR Code distortion)
6. Format Information สำหรับเก็บรูปแบบหน้ากาก (Masking Pattern)
7. Version Information สำหรับบอกเวอร์ชันของ QR Code โดยมี 21 x 21 โมดูล (เวอร์ชัน 1) ถึง 177 x 177 โมดูล (เวอร์ชัน 40)
8. Data And Error Correction Codewords สำหรับเก็บข้อมูลและรหัสควบคุมการผิดพลาด

## การเข้ารหัสคิวอาร์โค้ด (Encode procedure)

**ขั้นตอนที่ 1** การวิเคราะห์ข้อมูล (Data analysis) เป็นขั้นตอนสำหรับกำหนดชนิดของข้อมูล โดยแทนชนิดของข้อมูลด้วยเลขฐาน 2 จำนวน 4 บิต ตามตารางที่ 2.2

ตารางที่ 2.2

ชนิดของข้อมูลแทนด้วยเลขฐาน 2

Mode	Indicator
Numeric	0001
Alphanumeric	0010
8-bit Byte	0100
Kanji	1000

หมายเหตุ. จาก ISO/IEC, Information technology (2006). Automatic identification and data capture techniques QR code 2005 bar code symbology specification, (pp. 22), IS 18004:2006.

**ขั้นตอนที่ 2** การเข้ารหัสข้อมูล (Data encoding) นำจำนวนอักขระของข้อความที่ต้องการเข้ารหัสแปลงให้เป็นเลขฐานสอง โดยจำนวนบิตของเลขฐานสองนั้นสามารถดูได้จากตาราง 2.3 ตามเวอร์ชัน จากนั้นนำข้อความที่ต้องการเข้ารหัสแต่ละตัวแทนด้วยค่าตามภาพที่ 2.4 แล้วจับคู่อักขระ จากนั้นนำค่าตัวแรกคูณด้วย 45 แล้วนำค่าตัวที่ 2 มาบวก เมื่อได้ผลลัพธ์ ให้แปลงเป็นข้อมูลเลขฐานสอง ขนาด 11 บิต กรณีที่ตัวสุดท้ายไม่มีคู่ให้แปลงเป็นเลขฐานสอง ขนาด 6 บิต

## ตารางที่ 2.3

จำนวนบิตที่ต้องแปลงตามเวอร์ชัน

เวอร์ชัน	จำนวนบิต
1 - 9	9
10 - 26	11
27 - 40	13

หมายเหตุ. จาก ISO/IEC, Information technology (2006). Automatic identification and data capture techniques QR code 2005 bar code symbology specification, (pp. 23), IS 18004:2006.

สำหรับการปิดท้ายข้อมูล จำนวนความจุข้อมูลสูงสุดในแต่ละชนิดรุ่นและระดับแก้ไขความผิดพลาดกำหนดมา หากจำนวนบิตที่ได้น้อยกว่าที่กำหนดจะปิดท้ายข้อมูลด้วย 0 ไม่เกิน 4 ตัว

Char.	Value	Char.	Value	Char.	Value	Char.	Value	Char.	Value	Char.	Value	Char.	Value
0	0	6	6	C	12	I	18	O	24	U	30	SP	36
1	1	7	7	D	13	J	19	P	25	V	31	\$	37
2	2	8	8	E	14	K	20	Q	26	W	32	%	38
3	3	9	9	F	15	L	21	R	27	X	33	*	39
4	4	A	10	G	16	M	22	S	28	Y	34	+	40
5	5	B	11	H	17	N	23	T	29	Z	35	-	41
												.	42
												/	43
												:	44

ภาพที่ 2.4 คาร์รหัสของอักขระและตัวเลขใน Alphanumeric Mode จาก ISO/IEC, Information technology (2006). Automatic identification and data capture techniques QR code 2005 bar code symbology specification, (pp. 26), IS 18004:2006.

**ตัวอย่าง** การเข้ารหัสคิวอาร์โค้ดเวอร์ชัน 1 ระดับการแก้ไขความผิดพลาดระดับ Q โดยข้อมูลอินพุต เป็น Alphanumeric Mode

Input data: HELLO WORLD

1. แทนชนิดของข้อมูลด้วยเลขฐาน 2 จำนวน 4 บิต จากตารางที่ 2.2 โดยในตัวอย่างนี้ข้อมูล เป็น Alphanumeric ดังนั้นจะได้ 0010

2. แปลง จำนวนอักขระของข้อมูลให้เป็นไบนารี โดยจำนวนบิตขึ้นอยู่กับเวอร์ชันของควออาร์โค้ด และระดับการแก้ไขข้อผิดพลาด ซึ่งในตัวอย่างนี้ เป็น Version 1 ระดับการแก้ไขข้อผิดพลาด Q จึงใช้จำนวน 9 บิต

จากจำนวนอักขระ 11 จะได้ 000001011

3. แทนตัวอักขระแต่ละตัวด้วยค่าจากภาพที่ 2.4

จาก HELLO WORLD จะได้ (17,14,21,21,24,36,32,24,27,21,13)

4. จัดกลุ่มผลลัพธ์ที่ได้จากข้อ 3 โดยจัดกลุ่มละ 2 ตัว

จาก (17,14,21,21,24,36,32,24,27,21,13)

จะได้ (17,14) (21,21) (24,36) (32,24) (27,21) (13)

5. แปลงค่าแต่ละกลุ่มให้เป็น ไบนารี 11 บิต

จาก (17,14) จะได้  $17 * 45 + 14 = 779$  แปลงเป็น ไบนารี 01100001011

จาก (21,21) จะได้  $21 * 45 + 21 = 966$  แปลงเป็น ไบนารี 01111000110

จาก (24,36) จะได้  $24 * 45 + 36 = 1116$  แปลงเป็น ไบนารี 10001011100

จาก (32,24) จะได้  $32 * 45 + 24 = 1464$  แปลงเป็น ไบนารี 10110111000

จาก (27,21) จะได้  $27 * 45 + 21 = 1236$  แปลงเป็น ไบนารี 10011010100

จาก (13) จะได้ 13 แปลงเป็น ไบนารี 001101

6. นำค่าไบนารีที่ได้จากข้อ 5 มาเรียงต่อกันตามลำดับ

จะได้ 01100001011 01111000110 10001011100 10110111000 10011010100  
001101

7. นำไบนารีที่ได้จากข้อ 1 และ 2 มาไว้ข้างหน้า ไบนารีที่ได้จากข้อ 6 จะได้ ดังนี้

0010 000001011 01100001011 01111000110 10001011100 10110111000  
10011010100 001101



8. ปิดท้ายข้อมูลด้วย 0 ไม่เกิน 4 ตัว

0010 000001011 01100001011 01111000110 10001011100 10110111000  
10011010100 001101 0000

9. จากนั้นทำการจัดกลุ่มข้อมูลใหม่ให้แต่ละกลุ่มมีขนาด 8 บิต หากจำนวนชุดของข้อมูลบิต (codewords) ยังไม่ครบตามที่กำหนดตามตาราง 2.4 จะให้เพิ่มข้อมูล 2 ชุด ประกอบด้วย “11101100” และ “00010001” จนกว่าจะครบ

00100000 01011011 00001011 01111000 11010001 01110010 11011100  
01001101 01000011 01000000 11101100 00010001 11101100

ตารางที่ 2.4

จำนวนชุดข้อมูลบิตของคิวอาร์โค้ดมาตรฐานเวอร์ชันหนึ่ง

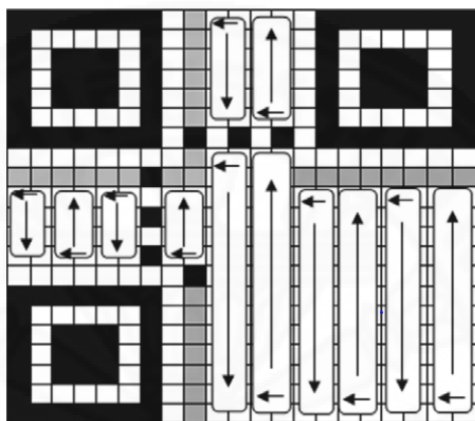
เวอร์ชัน	ระดับแก้ไขข้อผิดพลาด	จำนวนชุดข้อมูลบิต
1	L	19
	M	16
	Q	13
	H	9

หมายเหตุ จำนวนชุดข้อมูลบิตของคิวอาร์โค้ดมาตรฐานเวอร์ชัน 2 – 40 สามารถดูได้ที่ภาคผนวก ก. จาก ISO/IEC, Information technology (2006). Automatic identification and data capture techniques QR code 2005 bar code symbology specification, (pp. 33), IS 18004:2006.

**ขั้นตอนที่ 3** การเข้ารหัสเพื่อใช้แก้ไขข้อผิดพลาด (Error correction coding) ซึ่งการเข้ารหัสเพื่อใช้แก้ไขข้อผิดพลาดของคิวอาร์โค้ดนั้น จะใช้การรหัสรีดโซโลมอนเพื่อป้องกันการความผิดพลาดนั้น โดยรหัสรีดโซโลมอนจะจัดเรียงในพื้นที่ข้อมูลคิวอาร์โค้ดใหม่ โดยการแก้ไขข้อผิดพลาด ณ จุดนั้นๆ ซึ่งฟังก์ชันการทำงาน ทำให้รหัสยังคงสามารถอ่านได้อย่างถูกต้องแม้ว่าจะมีจุดหรือรอยเปื้อนหรือความเสียหายที่เกิดขึ้นจนกว่าจะมีการแก้ไขข้อผิดพลาดนั้นเสร็จสิ้น ซึ่งผู้ใช้จะกำหนดระดับการป้องกันความผิดพลาดเอง

**ขั้นตอนที่ 4** รวมข้อมูลสุดท้าย (Structure final message) นำข้อมูลที่ได้จากขั้นตอนที่ 2 และขั้นตอนที่ 3 มารวมกันเพื่อให้เป็นข้อความเดียวกัน

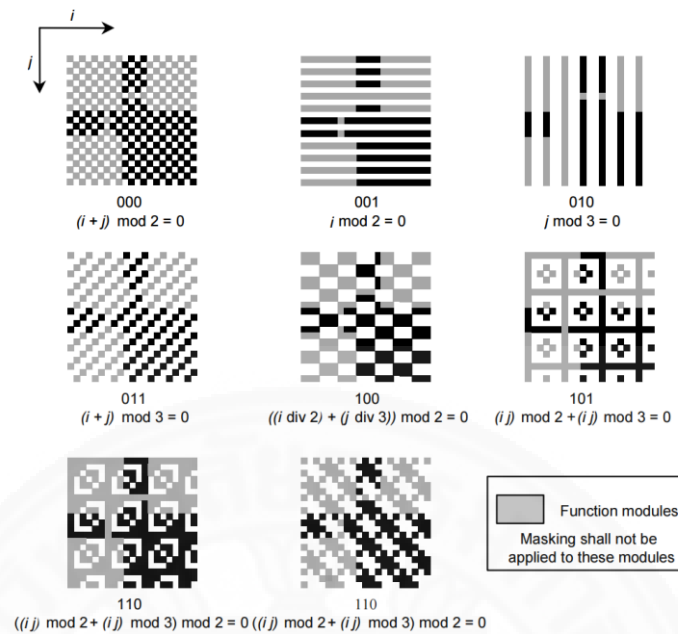
**ขั้นตอนที่ 5** จัดสรรพื้นที่ (Module placement in matrix) นำข้อความที่ได้จากขั้นตอนที่ 4 มาจัดสรรพื้นที่ในส่วน Data And Error Correction Codewords ตามทิศทางในภาพที่ 2.5 ในโครงสร้าง QR Code รวมกับส่วน Finder Pattern, Separators, Timing Pattern และ Alignment Patterns



ภาพที่ 2.5 ทิศทางการบันทึกข้อมูลในคิวอาร์โค้ดของเวอร์ชัน 1

**ขั้นตอนที่ 6** เลือกรูปแบบหน้ากาก (Data masking) เลือกรูปแบบหน้ากาก (Mask Pattern) ที่เหมาะสมกับข้อมูล ซึ่งหน้ากากมีทั้งหมด 8 แบบ ดังภาพที่ 2.6

**ขั้นตอนที่ 7** กำหนดรูปแบบและเวอร์ชัน (Format and version information) กำหนดรูปแบบและเวอร์ชันของคิวอาร์โค้ดนี้ที่ส่วน Version Information



ภาพที่ 2.6 หน้ากากของคิวอาร์โค้ดเวอร์ชัน 1

### 2.1.2 รหัสรีดโซโลมอน (Reed-Solomon)

รหัสแก้ไขข้อผิดพลาดมีมากมายที่แตกต่างกันออกไปตามชนิดของรหัสแก้ไขข้อผิดพลาดที่ได้ถูกสร้างขึ้นในหลายปีที่ผ่านมา แต่ชนิดที่เป็นที่นิยมใช้ในคอมพิวเตอร์คือวิธีการรีดโซโลมอน สร้างขึ้นโดยสองนักวิจัยคือ Irving Reed และ Gustave Solomon เป็นคนแรกที่ทำให้คิดค้นวิธีการนี้และได้นำมาใช้ รหัสรีดโซโลมอน ได้ถูกนำมาใช้อย่างแพร่หลายในการตรวจสอบข้อผิดพลาดและการแก้ไขข้อผิดพลาดของข้อมูล ในการคำนวณหลายๆ ด้าน และด้านเป็นตัวกลางการสื่อสาร รวมทั้งการบันทึกข้อมูลด้วยแม่เหล็ก การบันทึกข้อมูลทางแสง ในโมเดมความเร็วสูง และการส่งข้อมูลผ่านช่องสัญญาณ ก็ได้มีการเลือกที่จะใช้การเข้ารหัสแบบรีดโซโลมอนเพราะว่ามีความง่ายในการเข้ารหัสกว่ารหัสแบบอื่นๆ มาก สามารถตรวจจับข้อผิดพลาด และทำการแก้ไขข้อผิดพลาดที่จำนวนบิตของข้อมูลมากๆ ได้ ต้องการจำนวนบิต Error Correction Code (ECC) น้อยๆ ก็พอสำหรับบิตของข้อมูลได้ (Mamidi, Iancu, Iancu, Schulte and Glossner, 2005)

สมการรหัสรีดโซโลมอนแทนด้วย

$$Rs(n, k)$$

กล่าวคือขนาดของบล็อก  $n$  เทอม และข้อมูลจำนวน  $k$  เทอม โดย  $(n - k) = 2t$  เมื่อ  $t$  คือ จำนวนสูงสุดของสัญลักษณ์ที่อาจผิดพลาดได้ตั้งนั้น

$$t = \left\lceil \frac{d_{\min} - 1}{2} \right\rceil = \left\lceil \frac{n - k}{2} \right\rceil$$

และเมื่อ  $\rho$  คือ Erasure รหัสรีดโซโลมอนจะแก้ไขได้คือ

$$\rho = (n - k) = 2t$$

หรือเมื่อ  $\alpha$  และ  $\beta$  คือจำนวนสัญลักษณ์ผิดพลาดและจำนวนสัญลักษณ์ที่โดนลบแท้จริงตามลำดับ

$$2\alpha + \beta \leq (n - k)$$

Erasure ก็คือสัญลักษณ์ที่ถูกลบไป มักจะใส่ข้อมูลเป็นศูนย์ไว้ (ดังนั้นจึงสามารถรู้แค่ตำแหน่งของ ข้อมูลแต่ไม่รู้ขนาดของข้อมูล) โดยการเข้ารหัสรีดโซโลมอน จะใช้พหุนามกำเนิด (generator - polynomial) ของรหัสรีดโซโลมอน เขียนได้ คือ

$$g(x) = \sum_{i=0}^{2t} g_i x^i = g_0 + g_1 x + g_2 x^2 + \dots + g_{2t} x^{2t}$$

เมื่อ  $g \in (2t)$  ดีกรีของพหุนามกำเนิดจำนวนพหุนามกำเนิด คือ  $2t$  ดังนั้นจึงสามารถที่จะกำหนดจากจำนวนรากที่ต่อเนื่องจำนวน  $2t$  รากหรือเมื่อเขียนเป็นรูปทั่วไปจะได้

$$g(x) = \prod_{i=j_0}^{j_0+2t-1} (x - \alpha^i)$$

ปกติแล้วจะให้  $j_0 = 1$  แต่บางกรณี  $j_0 \neq 1$  ก็ได้เช่น  $2(j_0 + t) = 2^m$  ในกรณีต้องการสัมประสิทธิ์  $g(x)$  แบบสมมาตร ในส่วนของการเข้ารหัสเชิงระบบนั้นเนื่องจากว่ารหัสรีดโซโลมอนเป็นรหัส วนรอบการเลื่อนของคำรหัสก็ยังคงเป็นคำรหัสอยู่เช่นนี้หากลองเลื่อน  $k$  ข้อมูลไปจนขวาสุดแล้ว เติมส่วน  $2t$  ตำแหน่งทางด้านหน้าก็จะได้รหัสรีดโซโลมอนเชิงระบบการกระทำดังกล่าวในเชิงคณิตศาสตร์ก็คือการทำ

$$x^{(n-k)} m(x)$$

และเมื่อหาร  $x^{(n-k)} m(k)$  ด้วย  $g(x)$  ก็จะได้เศษซึ่งก็คือพาริตีนั้นเอง ดังนั้น

$$x^{(n-k)} m(x) = Q(x)g(x) + p(x)$$

เมื่อ  $Q(x)$  คือผลหารและ  $g(x)$  คือเศษ และเมื่อสนใจเฉพาะเศษจึงเขียนใหม่ได้เป็น

$$x^{(n-k)} m(x) \bmod g(x) \text{ หรือ } p(x) = x^{(n-k)} m(x) \bmod g(x)$$

ดังนั้นจึงเขียนรหัสเชิงระบบได้คือ  $c(x) = p(x) + x^{2t}m(x)$

### 2.1.3 ภาพระดับเทา (Grayscale Image)

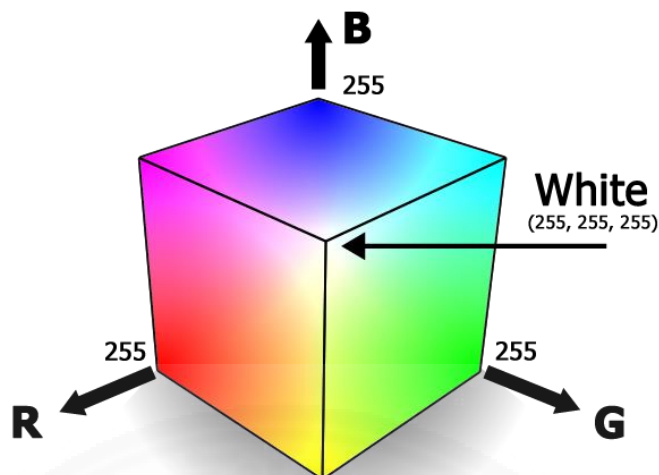
ภาพระดับเทา คือภาพที่ทุกๆ พิกเซล (Pixel) เก็บค่าความเข้มของแสง โดยพิกเซลที่มีระดับความเข้มแสงน้อยที่สุดเป็นดำ และความเข้มมากที่สุดเป็นขาว ส่วนเทา คือระดับความเข้มระหว่างทั้งสอง ภาพระดับเทาโดยทั่วไปจะมีระดับความเข้มแสงตั้งแต่ดำถึงขาว จำนวน 256 ระดับ (Johnson, Stephen. Stephen Johnson on digital photography. O'Reilly Media, Inc., 2006.)



ภาพที่ 2.7 ระดับสีเทา

### 2.1.4 โมเดลสี RGB

โมเดลสี RGB เป็นระบบสีที่แบ่งข้อมูลของสีออกเป็น R (แดง) G (เขียว) B (น้ำเงิน) ซึ่งเป็นสามสีหลักของแสงสี เลียนแบบการรับสีของตามนุษย์ ความเข้มแสงเมื่อผ่านการควอนไทซ์แล้ว จะถูกแบ่งออกเป็น 256 ระดับ (ระดับ 0 - 255) ซึ่งสอดคล้องกับ ระดับสีเทา (Grayscale) ตั้งแต่สีขาวถึงสีดำจำนวน 256 ระดับ ซึ่งหมายถึงหนึ่งจุดภาพจะมีข้อมูลแปดบิต



ภาพที่ 2.8 โมเดลสี RGB

### 2.1.5 การคำนวณระยะทางแบบยูคลิด (Euclidean distance)

การคำนวณระยะทางแบบยูคลิด มีสมการคำนวณระยะทางระหว่างจุดภาพสองจุด  $p$  และ  $q$  ถ้า  $p = (p_1, p_2, p_3, \dots, p_n)$  และ  $q = (q_1, q_2, q_3, \dots, q_n)$  มีสมการเป็นดังนี้

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \dots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^n (p_i - q_i)^2}$$

ภาพที่ 2.9 สมการสูตรการคำนวณระยะทางแบบยูคลิด

## 2.2 งานวิจัยที่เกี่ยวข้อง

การปรับปรุงการเพิ่มความจุของข้อมูลในรหัสคิวอาร์โค้ดในปัจจุบัน ได้มีนักวิจัยที่ทำการศึกษาค้นคว้าด้วยวิธีต่างๆ ดังนี้

**2.2.1 A New Color QR Code Forward Compatible with the Standard QR Code Decoder (Kikuchi, Fujiyoshi & Kiya, 2013)** ได้ทำการวิจัยวิธีเพิ่มความจุของคิวอาร์โค้ด โดยนำคิวอาร์โค้ดแบบมาตรฐานสามภาพ มาแปลงให้อยู่ในโมเดล  $YCbCr$  ซึ่งเป็นระบบสัญญาณสีหรือโมเดลสีที่ใช้สำหรับดิจิทัลวิดีโอ โดยจะแยกข้อมูลของจุดภาพตามการรับรู้ของประสาทตาของคน ซึ่งเป็นโมเดลสีที่จะแยกข้อมูลเกี่ยวกับความสว่างออกจากข้อมูลเกี่ยวกับระดับสี นั่นคือใช้ช่องประกอบ (Channel)  $Y$  เก็บค่าความสว่างของจุดภาพ ส่วน  $C_b$  คือค่าความแตกต่างระหว่างค่าสีน้ำเงินกับค่าความสว่าง ช่องประกอบ  $C_r$  คือความแตกต่างระหว่างค่าสีแดงกับความสว่าง ส่วนค่าความแตกต่าง

ระหว่างค่าสีเกี่ยวข้องกับค่าความสว่างนั้นไม่จำเป็นต้องเก็บเนื่องจากว่าค่าของ  $C_b + C_r + C_g$  มีค่าคงที่จึงสามารถคำนวณหาได้

ในส่วนการถอดรหัสข้อความ (Decode) สามารถใช้วิธีเดียวกับแบบคิวอาร์โค้ดมาตรฐานได้ เนื่องจากการเข้ารหัสของด้วยวิธีนี้ได้ปรับปรุงมาจากคิวอาร์โค้ดมาตรฐาน

### ขั้นตอนการเข้ารหัสข้อมูล มีดังนี้

1. แบ่งข้อความทั้งหมดเป็น 3 ส่วน แล้วนำข้อความแต่ละส่วนมาสร้างคิวอาร์โค้ดแบบมาตรฐาน โดย version และ ระดับแก้ไขความผิดพลาดสามารถแตกต่างกันได้
2. กำหนดระดับความเข้มสูงสุดและต่ำสุด ของแต่ละ Channel ทั้ง Y,  $C_b$ , และ  $C_r$  แล้วนำคิวอาร์โค้ดมาตรฐานทั้ง 3 ของข้อ 1 มาแปลงให้อยู่ในรูปแบบของ  $YC_bC_r$  ตามระดับความเข้มที่กำหนด
3. นำ  $YC_bC_r$  QR Code ทั้ง 3 ที่ได้จากข้อ 2 มาแปลงให้อยู่ในรูปแบบ RGB โดยใช้สมการ (Poynton & Kaufmann, 2003) ดังนี้

$$R = 1.164(Y - 16) + 1.596(C_r - 128),$$

$$G = 1.164(Y - 16) - 0.391(C_b - 128) - 0.391(C_r - 128),$$

$$B = 1.164(Y - 16) + 2.018(C_b - 128)$$



(a) Y channel.



(b)  $C_b$  channel.



(c)  $C_r$  channel.



(d) A color QR code.

ภาพที่ 2.10 ผลลัพธ์ของรหัสคิวอาร์โค้ดสี จาก “A new color QR code forward compatible with the standard QR code decoder” by Kikuchi M, Fujiyoshi M, Kiya H , Intelligent Signal Processing and Communications Systems (ISPACS), 2013, p. 28

### ขั้นตอนการถอดรหัสข้อมูล มีดังนี้

1. ภาพรหัสคิวอาร์โค้ดที่ได้จากการถ่ายด้วยอุปกรณ์ จะอยู่ในรูปแบบสี RGB ให้เปลี่ยนเป็นรูปแบบ  $YCbCr$  ด้วยสมการ (Poynton & Kaufmann, 2003)
 
$$Y = 0.257R + 0.504G + 0.098B + 16,$$

$$Cb = -0.148R - 0.291G + 0.439B + 128,$$

$$Cr = 0.439R - 0.368G - 0.071B + 128$$
2. แยก Channel ของสี  $Y, C_b, C_r$
3. ถอดรหัสข้อมูลโดยใช้วิธีเดียวกับแบบรหัสคิวอาร์โค้ดมาตรฐาน

### การทดลอง

#### ข้อจำกัดการทดลอง

- ทดสอบโดยการถ่ายภาพจากหน้าจอคอมพิวเตอร์ (Monitor) และจากภาพคิวอาร์โค้ดที่ถูกพิมพ์ออกกระดาษ (Print out)
- ถ่ายภาพใต้แสงของหลอดฟลูออเรสเซนต์ (Fluorescent)
- ใช้จอ Monitor Princeton PTFWHF-19W
- ใช้ Printer KONICA MONOLTA magcolor 1650EN
- ใช้กระดาษธรรมดา (Plain paper)
- ถ่ายภาพโดยใช้ iPhone 4S (8 Mega Pixel, CMOS)
- เก็บภาพแบบ BMP
- ใช้ QR Code Version 3H
- ขนาด 3 CM
- ทดสอบโดยใช้ 3 decoder สำหรับ Standard QR
  1. Qrafter
  2. ICONIT
  3. QR Code Decode Library



### การทดลองที่ 1 ความเข้ากันได้

- แบ่งเป็น 3 เซต โดยกำหนดให้
  - Set A: ให้ความสำคัญกับ Channel  $C_b$  และ  $C_r$  มากกว่า  $Y$
  - Set B: ให้ความสำคัญกับ ทุกๆ Channel เท่าๆ กัน
  - Set C: ให้ความสำคัญกับ Channel  $Y$  และ  $C_r$  มากกว่า  $C_b$

### ตารางที่ 2.5

ชุดระดับความเข้มแสงแต่ละชุดสำหรับทดสอบ

Intensity sets	$(Y_1, Y_2), \Delta Y$	$(C_{b1}, C_{b2}), \Delta C_b$	$(C_{r1}, C_{r2}), \Delta C_r$
A	(120, 160), 40	(68, 170), 102	(68, 180), 120
B	(60, 160), 100	(102, 170), 68	(97, 170), 73
C	(80, 180), 100	(100, 160), 60	(82, 170), 88

หมายเหตุ. จาก “A new color QR code forward compatible with the standard QR code decoder” by Kikuchi M, Fujiyoshi M, Kiya H , Intelligent Signal Processing and Communications Systems (ISPACS), 2013, p. 30.

- แบ่งตามแต่ละเซตเป็นสีมืดกับสว่าง

**ผลการทดลอง** คือถ้า Channel  $Y$  มีช่วงความแตกต่างมากเพียงพอ ก็จะสามารถถอดรหัสคิวอาร์โค้ดได้ ด้วย Decoder สำหรับ Standard QR Code

### การทดลองที่ 2 ประสิทธิภาพ

- Print Color QR ที่นำเสนอลงกระดาษ
- ถ่ายภาพด้วยกล้อง เป็นภาพแบบ RGB
- แปลง RGB เป็น  $YCbCr$
- แปลงเป็นระดับเทาในแต่ละ Channel
- ดูว่าคิวอาร์โค้ดของแต่ละเซต เมื่อแบ่งเป็นคิวอาร์โค้ดมาตรฐานแยกแต่ละ Channel แล้วตามข้อ 4 มี Decoder ของคิวอาร์โค้ดมาตรฐานอันไหนบ้างที่ Decode ออกมาได้

## ผลการทดลอง การวัดประสิทธิภาพเป็นดังตารางที่ 2.6

### ตารางที่ 2.6

ผลการทดลองประสิทธิภาพการถอดรหัสรหัสคิวอาร์โค้ดสี ที่ถ่ายจากกระดาษ และจากจอภาพ ด้วย Decoder สำหรับ คิวอาร์โค้ดมาตรฐาน (O : สำเร็จ และ X : ไม่สำเร็จ)

	รหัสคิวอาร์โค้ดสีที่ได้จากกระดาษ			รหัสคิวอาร์โค้ดสีที่ได้จากจอภาพ		
	เซตระดับความเข้มแสง			เซตระดับความเข้มแสง		
	(a)	(b)	(c)	(a)	(b)	(c)
Decoder1	X	O	O	X	O	O
Decoder2	X	O	O	X	O	O
Decoder3	X	O	O	X	O	O

หมายเหตุ. จาก “A new color QR code forward compatible with the standard QR code decoder” by Kikuchi M, Fujiyoshi M, Kiya H , Intelligent Signal Processing and Communications Systems (ISPACS), 2013, p. 31.

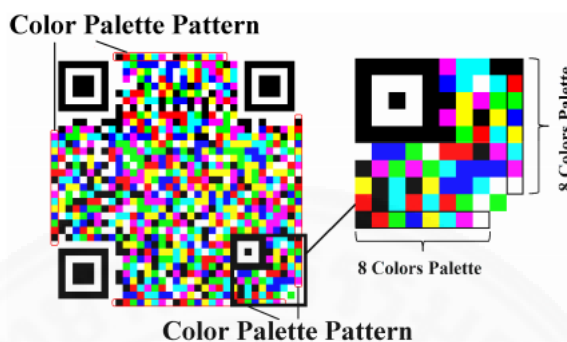
รหัสคิวอาร์โค้ดสีนี้สามารถเพิ่มจำนวนปริมาณความจุได้ประมาณ 3 เท่าของรหัสคิวอาร์โค้ดมาตรฐาน และสามารถใช้อัลกอริธึมการถอดรหัสของรหัสคิวอาร์โค้ดมาตรฐานได้หากค่าความสว่างของ Channel Y มีช่วงความแตกต่างมากเพียงพอ และสามารถถอดรหัสภาพรหัสคิวอาร์โค้ดสีที่ได้จากกระดาษและมอนิเตอร์

**2.2.2 High Capacity Colored Two Dimensional Code (Grillo, Lentini & Italiano, 2010)** ได้ทำการวิจัยวิธีเพิ่มความจุของคิวอาร์โค้ดเดิม ซึ่งโดยปกติคิวอาร์โค้ดมาตรฐานใน 1 โมดูลจะเก็บข้อมูล 1 บิต แต่ทางผู้วิจัยได้ปรับปรุงให้ 1 โมดูลสามารถเก็บข้อมูลได้มากกว่า 1 บิต โดยนำสีเข้ามาช่วย ที่เรียกว่า High Capacity Colored Two Dimensional Code หรือ HCC2D

ใน 1 โมดูลจะถูกแทนที่ด้วยสีอย่างน้อย 4 สี ซึ่งหากใช้จำนวน 4 สี ใน 1 โมดูลจะสามารถจุข้อมูลได้เพิ่มขึ้น 1 บิต โดยจำนวนบิตต่อโมดูล (BpM) เป็นไปตามสมการ ดังนี้

$$BpM = \log_2(\text{จำนวนสี})$$

เนื่องจากการใช้สีเข้ามา โครงสร้างของภาพรหัส จะมีแถบสีเพื่อบอกว่า รหัสภาพนั้น ประกอบด้วยสีอะไรบ้าง ซึ่งจะปรากฏอยู่ด้านขวาของภาพ ดังภาพที่ 2.11



ภาพที่ 2.11 บริเวณแถบที่บอกสีที่ใช้. จาก “2D Color Barcodes for Mobile Phones” by Querini, M., Grillo, A., Lentini, A., & Italiano, G. F, IJCSA, 2010, p. 148.

#### ขั้นตอนการเข้ารหัสข้อมูล ดังนี้

ส่วนที่เหมือนการเข้ารหัสคิวอาร์โค้ดมาตรฐาน คือ

1. วิเคราะห์ประเภทของข้อมูล
2. เข้ารหัสข้อมูล
3. เข้ารหัสเพื่อใช้แก้ไขข้อผิดพลาด
4. รวบรวมข้อมูลก่อนนำไปจัดสรรพื้นที่

ส่วนที่เพิ่มเติมจากการเข้ารหัสคิวอาร์โค้ดมาตรฐาน คือ

5. กำหนดจำนวนบิตต่อหนึ่งโมดูล จำนวนสี และ สี  
โดยได้จากสมการ บิตต่อหนึ่งโมดูล (BpM) =  $\log_2(\text{จำนวนสี})$
6. จัดสรรข้อมูล โดยทิศทางการจัดวางเหมือนคิวอาร์โค้ดมาตรฐาน แต่ในหนึ่งโมดูลจะเก็บตามจำนวนบิตที่กำหนดดังข้อ 5 จะได้ผลลัพธ์เป็นคิวอาร์โค้ดสี ตัวอย่างดังภาพที่ 14



ภาพที่ 2.12 ภาพผลลัพธ์คิวอาร์โค้ดสี (HCC2D) แบบ 4 สี (a) และ 16 (b) สี. จาก “High Capacity Colored Two Dimensional Code” by Grillo, Lentini & Italiano, Computer Science and Information Technology (IMCSIT), 2010, p. 712

#### การทดลอง

1. ใช้ Decoder ที่อ่าน Standard QR ได้ ของ *zxing* เป็น Java Opensource สำหรับ 1D/2D
2. ส่วนของโปรแกรมตัว Decode ที่ไม่มีการเปลี่ยนแปลงจากเดิม คือ
  - Recognition the position detection pattern
  - Recognition and exploiting the alignment pattern
  - Reading version and format information
3. จากนั้นปรับปรุงเพิ่มวิธีการ detect color palette
  - ถ้าไม่เจอทำต่อเป็นแบบมาตรฐาน
  - ถ้าเจอทำต่อโดยคิดว่าเป็น HCC2D
4. กระบวนการประมวลผลภาพจบโดยได้เมตริกซ์ออกมา
  - ถ้าเป็น Standard ก็คือ matrix เก็บค่า 0,1
  - ถ้าเป็น HCC2D จะได้เก็บเป็น Vector ตามค่าของ โดยใช้ความต่างของสีเทียบกับ color palette บนรูป
5. แปลงเป็นค่าไบนารี
6. ใช้ reed-solomon error correction แก้ไขข้อผิดพลาด (ถ้ามี)
7. ถอดรหัสไบนารีกลับมาจะได้ข้อความ

### การทดลองที่ 1 วัดความหนาแน่นของข้อมูล

- โดยเปรียบเทียบระหว่าง Standard QR, HCCB (M\$) และ HCC2D (Proposed)
- ใช้เครื่อง Scan นามบัตรความละเอียด 600 dpi
- หนึ่งโมดูลของคิวอาร์โค้ดมาตรฐานจะมีขนาด 0.17mm
- ทดสอบที่ เวอร์ชัน 19 ระดับแก้ไขข้อผิดพลาด M

ได้ผลการทดลองตาม ดังนี้

### ตารางที่ 2.7

ปริมาณความจุข้อมูลของรหัสคิวอาร์โค้ดแต่ละประเภท ที่พิมพ์ละเอียด 600 DPI

ชนิดของ บาร์โค้ด	ความหนาแน่นของข้อมูล (กิโลไบต์ ต่อ ตารางนิ้ว)
คิวอาร์โค้ด	0.627
HCCB	2.0
HCC2D	1.881

หมายเหตุ. จาก “High Capacity Colored Two Dimensional Code” by Grillo, Lentini & Italiano, Computer Science and Information Technology (IMCSIT), 2010, p. 715

### การทดลองที่ 2 วัดเวลาที่ใช้ของการถอดรหัสข้อมูล

- ใช้มากกว่า 100 รหัสคิวอาร์โค้ด
- ใช้รหัสคิวอาร์โค้ดหลายเวอร์ชัน
- เริ่มจับเวลาหลังการทดลอง alignment and detection เป็น มิลลิวินาที (Milisecond)
- เครื่องที่ใช้
  - OS : Linux Slackware 13.0
  - 1.73 GHz Intel Dual core
  - 1 GB Ram

ผลการทดลองที่ได้ คือ HCC2D ใช้เวลามากกว่า คิวอาร์โค้ดมาตรฐาน

- เวลาที่ใช้น้อยที่สุด คือ มากกว่าแบบมาตรฐานอยู่ 3% เมื่อรหัสคิวอาร์โค้ดสีเป็นแบบ 4 สีเวอร์ชัน 1H
- เวลาที่ใช้มากที่สุด คือ มากกว่าแบบมาตรฐานอยู่ 38% เมื่อรหัสคิวอาร์โค้ดสีเป็นแบบ 16 มีเวอร์ชัน 20L
- รหัสคิวอาร์โค้ดแบบ 4 สี มี ค่าเฉลี่ยอยู่ที่ 15% ของเวลาของคิวอาร์โค้ดมาตรฐาน
- รหัสคิวอาร์โค้ดแบบ 16 สี มี ค่าเฉลี่ยอยู่ที่ 25% ของเวลาของคิวอาร์โค้ดมาตรฐาน

เหตุผลคิดว่า เป็นเพราะ Error collection แต่ยังไม่มั่นใจ เพราะ

- 4 สี มีแนวโน้มว่าเมื่อ version ใหญ่ขึ้น จะมี % การใช้เวลามากกว่า version น้อยลง
- 16 color กลับไม่เป็นแบบนี้ ดูอื่นๆ

**การทดลองที่ 3** ทดสอบใช้จริงพิมพ์และสแกน

- พิมพ์ HCC2D ออกมาขนาด 1 นิ้ว
- ตั้งค่าเครื่องพิมพ์ 4 DPI คือ 180, 360, 720 และ 1440
- ใช้ version 5L, 10L และ 15L

**ผลการทดลอง** การทดสอบใช้จริงพิมพ์และสแกนสรุปได้ดังตารางที่ 2.8

ตารางที่ 2.8

การใช้งานของรหัสคิวอาร์โค้ดสีชนิด HCC2D ในความละเอียดที่ต่างกัน และรุ่นเวอร์ชันที่ต่างกัน (O : สำเร็จ และ X : ไม่สำเร็จ)

ความละเอียด	180 dpi	360 dpi	720 dpi	1440 dpi
เวอร์ชัน 5	X	O	O	O
เวอร์ชัน 10	X	O	O	O
เวอร์ชัน 15	X	X	O	O

หมายเหตุ. จาก “High Capacity Colored Two Dimensional Code” by Grillo, Lentini & Italiano, Computer Science and Information Technology (IMCSIT), 2010, p. 716

จากผลการทดสอบการใช้งานจริง สรุปได้ว่า ความละเอียดของภาพรหัสคิวอาร์โค้ดที่น้อยเกินไปไม่สามารถถอดรหัสได้

### 2.2.3 Gray scale two-dimensional code and its application (Furumoto, Watanabe & Morii, 2012) หลักการทำงานเป็นดังนี้

1. แบ่งข้อมูลเป็นส่วนๆ ตามจำนวนระดับบิต (bit depth) ของ grayscale ที่ต้องการ
2. นำแต่ละส่วนไปทำเป็นคิวอาร์โค้ดมาตรฐาน
3. นำคิวอาร์โค้ดที่ได้แต่ละส่วนในข้อ 2 มาซ้อนกันเป็นเลเยอร์
4. อ่านค่าบิตของตำแหน่งที่ตรงกันในแต่ละเลเยอร์ จากนั้นแปลงเป็น ระดับสีเทา ตามที่กำหนดไว้ ดังตารางที่ 2.9

ตารางที่ 2.9

แสดงเลขฐานสองและค่าระดับเทา

เลขฐานสองจำนวน 2 บิต	ค่าระดับเทา
00	255
10	170
01	85
11	0

หมายเหตุ. จาก “Gray scale two-dimensional code and its application. In The Institute of Electronics” by Furumoto K, Watanabe Y & Morii M, The Institute of Electronics, Information and Communication Engineers (IEICE Tech), 2012, p. 9



ภาพที่ 2.13 ภาพผลลัพธ์คิวอาร์โค้ดระดับเทา . จาก “Gray scale two-dimensional code and its application” by Furumoto, Watanabe & Morii, The Institute of Electronics, Information and Communication Engineers (IEICE Tech) , 2012, p. 9

### การทดลองที่ 1 ความเข้ากันได้กับอัลกอริธึมของการถอดรหัสคิวอาร์โค้ดมาตรฐาน

- ถ่ายภาพได้แสงของหลอด Fluorescent
- ใช้ Printer Canon MP630
- ใช้กระดาษธรรมดา (Plain paper)
- ถ่ายภาพโดยใช้ iPhone 4S (8 Mega Pixel, CMOS)
- เก็บภาพแบบ JPEG
- ใช้ QR Code Version 1L
- ขนาด 2, 4 และ 6 CM

ผลการทดลอง คือ ไม่สามารถทำการถอดรหัสได้

#### 2.2.4 สรุปงานวิจัยที่เกี่ยวข้อง

งานวิจัยที่กล่าวถึงนั้นเป็นงานวิจัยที่พยายามเพิ่มความสามารถในการจุข้อมูลในรหัสภาพคิวอาร์โค้ดให้มากขึ้น โดยการนำระบบสีเข้ามาช่วย ซึ่งงานวิจัยของทีม Kikuchi, Fujiyoshi และ Kiya นั้นสามารถเพิ่มปริมาณการจุข้อมูลได้ประมาณ 3 เท่าของรหัสคิวอาร์โค้ดปกติ เมื่อใช้เวอร์ชัน ขนาด และ ระดับแก้ไขความผิดพลาดเดียวกัน

ทางด้านงานวิจัยของทีม Grillo, Lentini และ Italiano สามารถเพิ่มความสามารถการจุข้อมูลได้มากกว่าคิวอาร์โค้ดมาตรฐาน โดยขึ้นอยู่กับจำนวนบิตต่อโมดูลที่กำหนด ซึ่งโดยปกติแล้วคิวอาร์โค้ดมาตรฐานจะมีจำนวนบิตต่อโมดูล เท่ากับ 1 เท่านั้น

งานวิจัยข้างต้นเป็นการนำสีมาใช้ แต่การใช้สีช่วยนั้นก็ยังมีปัญหาอยู่ เนื่องจากรหัสคิวอาร์โค้ดเป็นที่ที่สีนั้นควบคุมสีค่อนข้างยาก ความคมชัดของสีหรือของภาพ ซึ่งมักจะขึ้นกับอุปกรณ์การพิมพ์ ซึ่งหากเป็นเครื่องพิมพ์ทั่วไปอาจไม่เหมาะสมนัก รวมถึงสภาพแวดล้อมที่ภาพปรากฏความสว่างของแสง ระยะเวลาในการที่รหัสคิวอาร์โค้ดต้องโดนแดดเป็นเวลานาน ทำให้สีบนรหัสคิวอาร์โค้ดซีดและเพี้ยนไป ซึ่งสามารถส่งผลให้การถอดรหัสคิวอาร์โค้ดเป็นไปได้ยากและสามารถให้ผลลัพธ์ผิดไปได้ง่าย

ทางด้านงานวิจัยของทีม Furumoto, Watanabe และ Morii มีชื่อว่า Gray scale two-dimensional code and its application เป็นการเพิ่มความสามารถโดยการใช้ระดับเทาเข้ามาช่วย แต่การทดลองแจ้งว่าถอดรหัสไม่ได้จึงไม่สามารถระบุได้ว่าสามารถเพิ่มปริมาณการจุข้อมูลเท่าไร

งานวิจัยที่เกี่ยวข้องสามารถสรุปผลได้ตามตารางที่ 2.10



ตารางที่ 2.10

สรุปงานวิจัยที่เกี่ยวข้อง

งานวิจัย	ถอดรหัสได้	รูปแบบ	จำนวนสี/ ระดับเทา	ความจุโดยประมาณ
Kikuchi et al.	/	สี	8	3
Grillo et al.	/	สี	16	4
Furumoto et al.	x	ระดับเทา	4	ไม่ได้ระบุ

หมายเหตุ ความจุโดยประมาณ เทียบกับ คิวอาร์โค้ดมาตรฐาน

### บทที่ 3 วิธีการวิจัย

ในการจัดทำวิทยานิพนธ์ฉบับนี้มีวัตถุประสงค์เพื่อเพิ่มความสามารถในการบันทึกข้อมูลให้มากขึ้นจากระหัสคิวอาร์โค้ดแบบมาตรฐานโดยมีแนวคิด วิธีดำเนินงานวิจัย และขั้นตอนการทำวิจัย ดังนี้

#### 3.1 แนวคิดและวิธีดำเนินงานวิจัย

จากงานวิจัยที่เกี่ยวข้องสามารถทำให้รหัสคิวอาร์โค้ดที่เป็นมาตรฐานนั้นสามารถเพิ่มปริมาณความจุให้แก่รหัสคิวอาร์โค้ดได้ แต่เนื่องจากในงานวิจัยเกี่ยวกับการเพิ่มความจุของคิวอาร์โค้ดที่ผ่านมา (Kikuchi, Fujiyoshi & Kiya, 2013) (Grillo, Lentini & Italiano, 2010) มักออกแบบเป็นคิวอาร์โค้ดสี แต่โดยปกติทั่วไป เรามักจะเห็นรหัสคิวอาร์โค้ดอยู่บนกระดาษ หรือป้ายโฆษณา หรือพิมพ์รหัสคิวอาร์โค้ดเพื่อให้อยู่บนผลิตภัณฑ์ หรือสิ่งของที่ต้องการ ซึ่งคิวอาร์โค้ดสีอาจไม่เหมาะกับการใช้งานดังกล่าวนัก เนื่องจากรหัสคิวอาร์โค้ดเป็นที่ที่สีนั้นควบคุมสีรวมทั้งความคมชัดของสีหรือของภาพได้ค่อนข้างยาก เนื่องจากมักขึ้นกับคุณภาพของอุปกรณ์การพิมพ์ และหากเป็นเครื่องพิมพ์ทั่วไปอาจไม่เหมาะสมนัก รวมถึงความสว่างของแสงในสภาพแวดล้อม ซึ่งสามารถส่งผลให้การถอดรหัสคิวอาร์โค้ดผิดพลาดได้ รวมถึงระดับเทาสามารถใช้เครื่องพิมพ์แบบใดก็ได้ซึ่งสะดวกกว่าและประหยัดค่าใช้จ่ายมากกว่าในการใช้หมึกสีอื่นๆ

แม้ว่าในงานวิจัยที่ผ่านมาจะมีการเพิ่มความจุระดับเทา (Furumoto, Watanabe & Morii, 2012) มาแล้วแต่ยังไม่สามารถนำมาถอดรหัสได้ ทางผู้จัดทำจึงคิดที่จะพัฒนาระบบสำหรับเข้ารหัสและถอดรหัสคิวอาร์โค้ดโดยใช้ระดับเทาช่วยเพิ่มปริมาณความจุของรหัสคิวอาร์โค้ด

ทางผู้วิจัยจึงตั้งสมมติฐานว่า

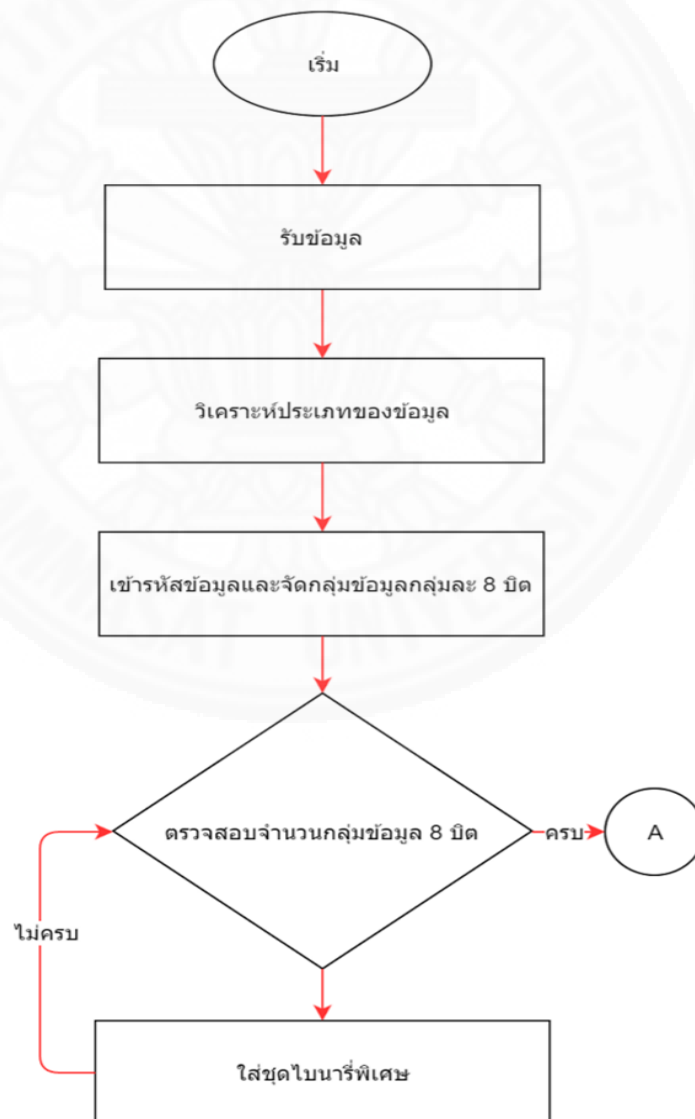
1. ขนาดของรหัสภาพคิวอาร์โค้ดใหญ่ขึ้นทำให้มีโอกาสถอดรหัสสำเร็จมากขึ้น
2. เมื่อขนาดภาพคงที่คิวอาร์โค้ดมาตรฐานเวอร์ชันต่ำกว่าจะมีโอกาสถอดรหัสสำเร็จมากกว่าคิวอาร์โค้ดที่มีเวอร์ชันสูงกว่า
3. ภายใต้สภาพแสงสีขาว (Daylight) จะมีโอกาสถอดรหัสสำเร็จมากกว่าภาพแสงสีส้ม (Warm White)
4. ระดับเทาที่มีการเปลี่ยนแปลงค่าองค์ประกอบ RGB น้อยกว่าสีที่สภาพแสงสีขาว (Daylight) และสภาพแสงสีส้ม (Warm White)

5. รหัสควอาร์โค้ดโดยใช้ระดับเทาสามารถให้ความจุที่ใกล้เคียงกับรหัสควอาร์โค้ดสี ที่จำนวนสีและจำนวนระดับเทาเท่ากัน

### 3.2 ขั้นตอนการวิจัย

ในงานวิจัยเพิ่มปริมาณความจุของรหัสควอาร์โค้ดด้วยระดับเทา นี้ สามารถแบ่งขั้นตอน ออกเป็นสองส่วน ดังนี้

3.2.1 การเข้ารหัสควอาร์โค้ดระดับเทา ลำดับการทำงานของ การเข้ารหัสควอาร์โค้ดระดับเทาเป็นไป ดังภาพ



ภาพที่ 3.1 การเข้ารหัสควอาร์โค้ดให้อยู่ในรูปแบบระดับเทา



ภาพที่ 3.2 การเข้ารหัสคิวอาร์โค้ดให้อยู่ในรูปแบบระดับเทา (ต่อ)

จากรูปสามารถอธิบายขั้นตอนได้เป็น 5 ขั้นตอนได้แก่

1. การวิเคราะห์ข้อมูล (Data analysis)
2. การเข้ารหัสข้อมูล (Data encoding)
3. กำหนดจำนวนระดับเทาที่จะใช้
4. หาค่าระยะห่างของแต่ละระดับเทา
5. จัดสรรพื้นที่ (Module placement in matrix)

**ขั้นตอนที่ 1** การวิเคราะห์ข้อมูล (Data analysis) เป็นขั้นตอนสำหรับกำหนดชนิดของข้อมูล โดยแทนชนิดของข้อมูลด้วยเลขฐานสองจำนวน 4 บิต ตามตาราง ดังนี้

ตารางที่ 3.1

ชนิดของข้อมูลแทนด้วยเลขฐาน 2

Mode	Indicator
Numeric	0001
Alphanumeric	0010
8-bit Byte	0100
Kanji	1000

**ขั้นตอนที่ 2** การเข้ารหัสข้อมูล (Data encoding) การเข้ารหัสของคิวอาร์โค้ดระดับเท่า นั้นมีการทำงานเหมือนกับการเข้ารหัสข้อมูลของคิวอาร์โค้ดมาตรฐานที่ได้กล่าวไว้ในบทที่ 2 ในหน้าที่ 7 ขั้นตอนที่ 2

**ขั้นตอนที่ 3** กำหนดจำนวนระดับเท่าที่ใช้ เพื่อหาจำนวนบิตต่อโมดูล ซึ่งหาได้จากสูตร

$$\text{BpM} = \log_2 (\text{จำนวนระดับเท่า})$$

โดย BpM คือ จำนวนบิตต่อหนึ่งโมดูล

**ตัวอย่าง** การคำนวณหาจำนวนบิตต่อโมดูล โดยกำหนดระดับเท่ามา 4 ระดับ นำเข้าสมการ

$$\text{BpM} = \log_2 (4)$$

ดังนั้น หากต้องการใช้ระดับเท่า 4 ระดับ จำนวนบิตต่อโมดูล (BpM) จะเท่ากับ 2 ซึ่งหมายความว่าใน หนึ่งโมดูลของภาพคิวอาร์โค้ดระดับเท่าสามารถเก็บข้อมูลได้ 2 บิต

**ขั้นตอนที่ 4** หาค่าระยะห่างของแต่ละระดับเท่า โดยนำจำนวนของระดับเท่าหาค่าระยะห่างของแต่ละระดับเท่าเพื่อแทนโมดูลด้วยระดับเท่าที่คำนวณได้ในขั้นตอนจัดสรรพื้นที่ โดยสมการ

$$\text{ระยะห่างของแต่ละระดับเท่า} = 256 / \text{จำนวนระดับเท่า} - 1$$

**ตัวอย่าง** การหาค่าระดับเท่าหากต้องการใช้ 4 ระดับเท่า ดังนั้น

$$\text{ระยะห่างของแต่ละระดับเท่า} = 256 / (4 - 1) = 85 \text{ ระดับ}$$

ดังนั้น หาค่าระดับเท่า 4 ระดับ ได้แก่ 0 , 85 , 170 , 255

**ขั้นตอนที่ 5** จัดสรรพื้นที่ (Module placement in matrix) นำข้อความที่ได้จากขั้นตอนที่ 4 มาจัดสรรพื้นที่ในส่วน Data And Error Correction Codewords โดยนำข้อมูลแต่ละชุดแทนด้วยระดับเทาที่คำนวณได้จากขั้นตอนที่ 4 ในโครงสร้าง QR Code รวมกับส่วน Finder Pattern, Separators, Timing Pattern และ Alignment Patterns

**ตัวอย่าง** การแทนชุดข้อมูลด้วยระดับเทา ในกรณียกตัวอย่าง 2 บิต ต่อ 1 โมดูล

ข้อมูล 2 บิตที่สามารถเป็นไปได้คือ

11 แทนค่าด้วยระดับเทา 0

10 แทนค่าด้วยระดับเทา 85

01 แทนค่าด้วยระดับเทา 170

00 แทนค่าด้วยระดับเทา 225

เมื่อทำการเข้ารหัสคิวอาร์โค้ดระดับเทาตามขั้นตอนขั้นต้น ได้ภาพคิวอาร์โค้ดระดับเทา ดังนี้



ภาพที่ 3.3 ตัวอย่างภาพคิวอาร์โค้ดระดับเทา 4 ระดับ

3.2.2 การถอดรหัสคิวอาร์โค้ดระดับเทา ลำดับการทำงานของ การถอดรหัสคิวอาร์โค้ดระดับเทาเป็นไป ดังภาพ



ภาพที่ 3.4 การถอดรหัสคิวอาร์โค้ดให้อยู่ในรูปแบบระดับเทา

จากภาพ 3.4 สามารถอธิบายขั้นตอนการถอดรหัสคิวยาร์โค้ดได้เป็น 5 ขั้นตอน ได้แก่

1. ระบุตำแหน่งของสัญลักษณ์คิวยาร์โค้ด
2. หาเวอร์ชันและจำนวนโมดูล
3. อ่านค่าสีของแต่ละโมดูล
4. แบ่งข้อมูลไบนารี
5. ถอดรหัสคิวยาร์โค้ดระดับเทา

ซึ่งแต่ละขั้นตอนมีรายละเอียด ดังนี้

**ขั้นตอนที่ 1** ระบุตำแหน่งของสัญลักษณ์คิวยาร์โค้ดระดับเทา โดยการหาสัญลักษณ์ของการระบุตำแหน่งของคิวยาร์โค้ดทั้ง สามมุม (Finder pattern)

**ขั้นตอนที่ 2** อ่านเวอร์ชันจากภาพรหัสคิวยาร์โค้ดระดับเทาเพื่อใช้ในการอ้างอิงในการถอดรหัสข้อมูล เมื่อทราบเวอร์ชันของคิวยาร์โค้ดระดับเทาจะสามารถทราบได้ว่ารูปคิวยาร์โค้ดระดับเทาที่ทำการถอดรหัสนั้นมีจำนวนโมดูลเท่าไร โดยจำนวนโมดูลของรหัสคิวยาร์โค้ดแต่ละเวอร์ชันสามารถดูได้ที่ ภาคผนวก ก

**ขั้นตอนที่ 3** อ่านค่าสีของแต่ละโมดูลตามหลักเกณฑ์การวางตำแหน่งของรูปแบบของแต่ละเวอร์ชัน จากนั้นแปลงค่าสีเป็นไบนารี โดยใช้เทรชโฮลด์ ดังตัวอย่าง

**ตัวอย่าง** ภาพรหัสคิวยาร์โค้ดระดับเทาใช้ระดับเทา 4 ระดับ

ดังนั้น ระยะห่างของแต่ละระดับเทา =  $256 / (4 - 1)$

$$= 85 \text{ ระดับ}$$

ค่าระดับเทา 4 ระดับ ได้แก่ 0 , 85 , 170 , 255

เนื่องจากภาพรหัสคิวยาร์โค้ดระดับเทา 4 ระดับ ใน 1 โมดูลสามารถเก็บข้อมูลได้ 2 บิต ดังนั้น ข้อมูลไบนารี 2 บิตที่สามารถเป็นไปได้โดยใช้เทรชโฮลด์จะอยู่ในช่วง

11 แทนค่าด้วยระดับเทา 0 - 42

10 แทนค่าด้วยระดับเทา 43 - 128

01 แทนค่าด้วยระดับเทา 129 -212

00 แทนค่าด้วยระดับเทา 213 - 255



**ขั้นตอนที่ 4** แบ่งข้อมูลไบนารีที่ได้มาด้วยการแบ่งข้อมูลออกเป็นกลุ่ม กลุ่มละ 8 บิต ตามจำนวนตัวอักษร

**ขั้นตอนที่ 5** ถอดรหัสข้อมูลตัวอักษรและแสดงผล

### 3.3 การทดสอบประสิทธิภาพระบบเบื้องต้น

การทดสอบประสิทธิภาพระบบเบื้องต้น เป็นการวัดประสิทธิภาพและข้อจำกัดของระบบเพื่อนำมาใช้ปรับปรุง และทดลองวิจัยต่อไป การทดสอบนี้จะใช้วิธีถ่ายภาพด้วยสมาร์ทโฟน (Smart Phone) สามารถและนำภาพที่ได้มาวิเคราะห์ด้วยเครื่องคอมพิวเตอร์ ซึ่งการทดลองมีทั้งหมดสี่การทดลองดังนี้

การทดลองที่ 1 วัดความสามารถในการถอดรหัสคิวอาร์โค้ดมาตรฐาน

การทดลองที่ 2 เปรียบเทียบค่าองค์ประกอบสี RGB ของคิวอาร์โค้ดระดับเทาและแบบสีที่เปลี่ยนแปลงไปเมื่อถ่ายภาพในสภาพแสงสีขาว (Daylight) และแสงสีส้ม (Warm White)

การทดลองที่ 3 หาจำนวนระดับเทาที่เหมาะสมในการใช้งานคิวอาร์โค้ดระดับเทา

การทดลองที่ 4 หาความจุข้อมูลของคิวอาร์โค้ดระดับเทา

โดยแต่ละการทดลองมีสมมติฐานและรายละเอียด ดังนี้

#### การทดลองที่ 1 วัดความสามารถในการถอดรหัสคิวอาร์โค้ดมาตรฐาน

การทดลองนี้ ต้องการทดสอบข้อจำกัดการถอดรหัสของคิวอาร์โค้ดมาตรฐาน โดยมีสมมติฐาน 3 ข้อ คือ 1) ขนาดของภาพใหญ่ขึ้นทำให้มีโอกาสถอดรหัสสำเร็จมากขึ้น 2) ที่ขนาดภาพคงที่ คิวอาร์โค้ดมาตรฐานเวอร์ชันต่ำกว่า จะมีโอกาสถอดรหัสสำเร็จมากกว่า และ 3) ภายใต้สภาพแสงสีขาว (Daylight) ซึ่งคล้ายแสงธรรมชาติในตอนกลางวัน จะมีโอกาสถอดรหัสสำเร็จมากกว่าภาพแสงสีส้ม (Warm White) ซึ่งส่วนใหญ่เป็นแสงหลอดไฟ

การทดลองจะใช้ภาพรหัสคิวอาร์โค้ดมาตรฐานสี่เวอร์ชัน ได้แก่ เวอร์ชัน 2 เป็นเวอร์ชันที่เป็นที่นิยมเนื่องจากสามารถบรรจุข้อมูลได้ 47 ตัวอักษร ซึ่งส่วนใหญ่เพียงพอต่อการบรรจุข้อมูลที่อยู่เว็บไซต์ (URL) รวมถึงเป็นเวอร์ชันที่น้อยที่สุดที่มี Alignment Patterns ซึ่งมีไว้สำหรับตรวจสอบความโค้งของภาพคิวอาร์โค้ด (ISO/IEC, 2006) และเวอร์ชัน 5 10 15 ซึ่งเป็นเวอร์ชันที่ใช้ในการทดลองของ Grillo (Grillo, Lentini & Italiano, 2010) และคิวอาร์โค้ดที่ใช้ในการทดลองจะกำหนด

ระดับแก้ไขข้อผิดพลาด L ซึ่งเป็นระดับแก้ไขข้อผิดพลาดได้น้อยที่สุด โดยสามารถแก้ไขข้อผิดพลาดได้ 7% ขนาดภาพของคิวอาร์โค้ดแต่ละเวอร์ชันจะมีขนาดภาพหนึ่งนิ้ว ซึ่งเป็นขนาดภาพในการทดสอบใช้จริงพิมพ์และสแกนของ Grillo (Grillo, Lentini & Italiano, 2010) สองนิ้ว และ สามนิ้ว เป็นขนาดที่มากขึ้น 2 เท่าและ 3 เท่าเช่นเดียวกับการทดลองความเข้ากันได้กับอัลกอริธึมของการถอดรหัสคิวอาร์โค้ดมาตรฐานของ Furomoto (Furumoto, Watanabe & Morii, 2012)

เก็บภาพรหัสคิวอาร์โค้ดด้วยสมาร์ทโฟนสามรุ่น ดังนี้

1. iPhone 4 เป็นตัวแทนของสมาร์ทโฟนที่เป็นที่นิยม เนื่องจากเป็นสมาร์ทโฟนที่มียอดขายประมาณ 50 ล้านเครื่อง เป็นอันดับ 8 ของโลก (เฉพาะสมาร์ทโฟนที่วางจำหน่ายระหว่างปี 2010 - 2016) ปัจจุบัน iPhone 4 ไม่ได้รับการสนับสนุนระบบปฏิบัติการใหม่ ระบบปฏิบัติการล่าสุดที่รองรับคือ iOS 7 (Wikipedia, 2016)
  2. HTC One X เป็นตัวแทนของสมาร์ทโฟนที่ใช้ระบบปฏิบัติการแอนดรอยด์รุ่นเรือธง (Flagship) ที่วางจำหน่ายในช่วงเวลาใกล้เคียงกับ iPhone 4 (GSMArena, 2016)
  3. AIS Lava Iris 405+ เป็นตัวแทนของสมาร์ทโฟนที่ไม่มีการจับไฟก์ส้ออัตโนมัติ และเป็นตัวแทนของสมาร์ทโฟนระดับเริ่มต้น (Low Entry) (GSMArena, 2016)
- (โดยคุณสมบัติของสมาร์ทโฟนแสดงในภาคผนวก จ)

การเก็บภาพสมาร์ทโฟนแต่ละรุ่นจะเก็บภาพทุกเวอร์ชัน ทุกขนาด ทั้งในสภาพแสงสีขาวและแสงสีส้มที่ได้จากหลอดไฟ โดยหลอดไฟที่ใช้ในการทดลองได้แก่ Philips LED Bulb Scene Switch 9.5W อย่างละห้าภาพ

ในการทดลองนี้จะนำภาพที่ได้จากการถ่ายด้วยกล้องสมาร์ทโฟนสามรุ่น มาถอดรหัสภาพด้วยอัลกอริธึมการถอดรหัสคิวอาร์โค้ดมาตรฐาน (ISO/IEC, 2006)

**การทดลองที่ 2** เปรียบเทียบค่าองค์ประกอบสี RGB ของคิวอาร์โค้ดระดับเทาและแบบสีที่เปลี่ยนแปลงไปเมื่อถ่ายภาพในสภาพแสงสีขาวกับแสงสีส้ม

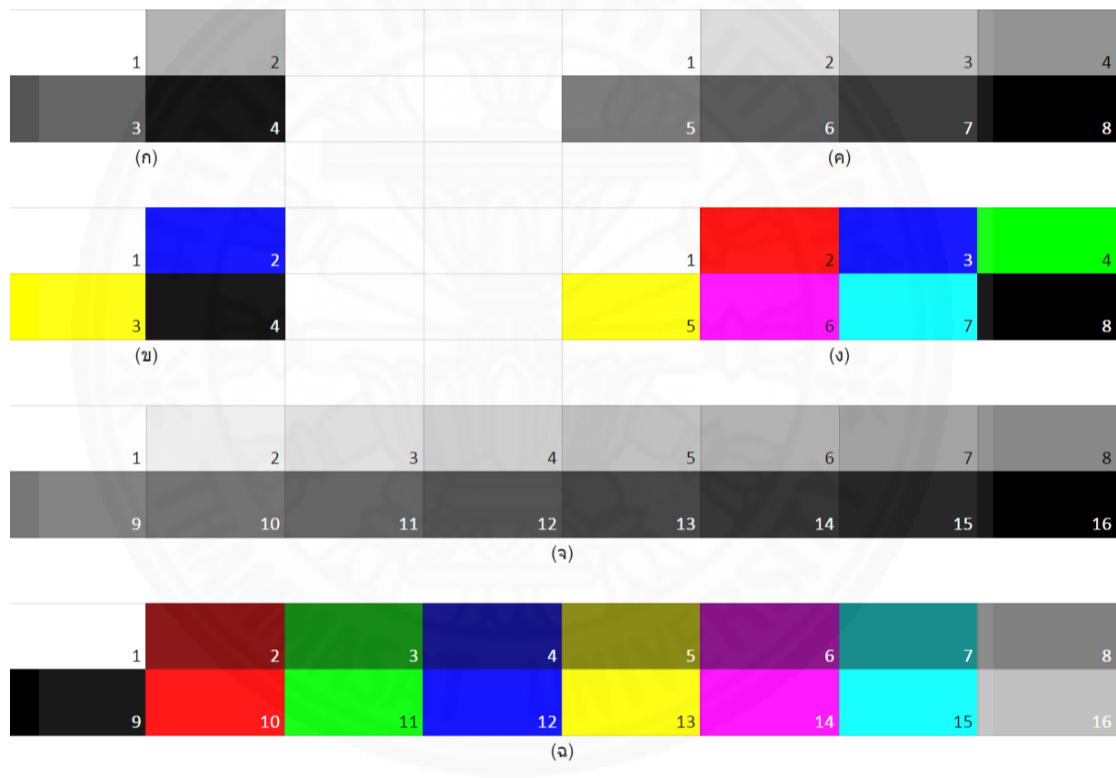
การทดลองนี้ต้องการทดสอบเปรียบเทียบค่า RGB ของสีและระดับเทาที่เปลี่ยนแปลงไปในสภาพแสงสีขาว (Daylight) และแสงสีส้ม (Warm White) โดยมีสมมติฐาน คือ ค่าองค์ประกอบ RGB ของระดับเทาเปลี่ยนแปลงน้อยกว่าค่าองค์ประกอบ RGB ของสีทั้งสภาพแสงสีขาวและแสงสีส้ม

การทดลองนี้จะกำหนดค่าสีและระดับเทา ระดับ โดยสีที่กำหนดในการทดลองอ้างอิงมาจากงานวิจัยของ Grillo (Grillo, Lentini & Italiano, 2010) ซึ่งมีสี 3 ชุด ได้แก่ 4, 8 และ 16 สี และใช้

ภาพระดับเทา 3 ชุด ได้แก่ 4, 8 และ 16 ระดับ ซึ่งค่า RGB ของระดับเทา จะทำการแบ่งช่วงเท่ากันตามจำนวนระดับเทาของแต่ละชุด ดังสมการในภาพ 3.5 ค่า RGB ของสีทั้งสามชุด และระดับเทาทั้งสามชุด แสดงในตาราง 3.2

$$\text{ระยะห่างของแต่ละระดับเทา} = 256 / (\text{จำนวนระดับเทา} - 1)$$

ภาพที่ 3.5 สมการระยะห่างของแต่ละระดับเทา



ภาพที่ 3.6 ลักษณะของกระดาษที่ใช้ในการทดลองที่ 2

ภาพที่ใช้ในการทดลองประกอบด้วย ค่า RGB สี 3 ชุดและระดับเทา 3 ชุด โดยแสดงสีและระดับเทา ในลักษณะตาราง ดังภาพที่ 3.6 เก็บภาพด้วยสมาร์ทโฟน 3 รุ่น (รายละเอียดสมาร์ทโฟนแต่ละรุ่นแสดงในภาคผนวก จ) จะใช้สมาร์ทโฟนแต่ละรุ่นในการเก็บภาพทั้งในสภาพแสงสีขาวและแสงสีส้ม อย่างละ 5 ภาพ

## ตารางที่ 3.2

ค่า RGB ของแต่ละสี/ระดับเทาที่ใช้ในการทดลอง

จำนวนสี/ระดับเทา	ค่า RGB	
4 ระดับ	(255 , 255 , 255)	(170 , 170 , 170)
	(85 , 85 , 85)	(0 , 0 , 0)
4 สี	(255 , 255 , 255)	(0 , 0 , 255)
	(0 , 255 , 255)	(0 , 0 , 0)
8 ระดับ	(255 , 255 , 255)	(219 , 219 , 219)
	(183 , 183 , 183)	(147 , 147 , 147)
	(111 , 111 , 111)	(75 , 75 , 75)
	(39 , 39 , 39)	(0 , 0 , 0)
8 สี	(255 , 255 , 255)	(255 , 0 , 0)
	(0 , 0 , 255)	(0 , 255 , 0)
	(255 , 255 , 0)	(255 , 0 , 255)
	(0 , 255 , 255)	(0 , 0 , 0)
16 ระดับ	(255 , 255 , 255)	(238 , 238 , 238)
	(221 , 221 , 221)	(204 , 204 , 204)
	(187 , 187 , 187)	(170 , 170 , 170)
	(153 , 153 , 153)	(136 , 136 , 136)
	(119 , 119 , 119)	(102 , 102 , 102)
	(85 , 85 , 85)	(68 , 68 , 68)
	(51 , 51 , 51)	(34 , 34 , 34)
	(17 , 17 , 17)	(0 , 0 , 0)
16 สี	(255 , 255 , 255)	(128 , 0 , 0)
	(0 , 128 , 0)	(0 , 0 , 128)
	(128 , 128 , 0)	(128 , 0 , 128)
	(0 , 128 , 128)	(128 , 128 , 128)
	(192 , 192 , 192)	(255 , 0 , 0)
	(0 , 0 , 255)	(0 , 255 , 0)

จำนวนสี/ระดับเทา	ค่า RGB	
	(255 , 255 , 0)	(255 , 0 , 255)
	(0 , 255 , 255)	(0 , 0 , 0)

ในการทดลองนี้จะทำการเก็บค่า RGB ของสีและระดับเทาจากภาพโดยใช้เครื่องมือ ColorZilla ซึ่งเป็นเครื่องมือเพิ่มเติม (extension) บนเว็บเบราว์เซอร์โครม (chrome) จากนั้นนำค่าสีที่ได้แต่ละช่องของทั้ง 5 รูป ที่มาจากอุปกรณ์และสภาพแสดงเดียวกัน มาหาค่าเฉลี่ย RGB โดยแยกเฉลี่ยแต่ละองค์ประกอบสี ได้แก่ R, G และ B ดังแสดงในภาพ 3.7 นำค่าเฉลี่ยที่ได้มาหาค่าความต่างกับของค่า RGB ตั้งต้น โดยใช้วิธีการคำนวณระยะห่างแบบยูคลิด (Euclidean Distance) โดยสมการสูตรการคำนวณแสดงดังภาพ 2.9 ในบทที่ 2

ค่าตั้งต้น (255,255,255)	ค่าที่วัดได้เฉลี่ย (215,210,214)
Distance R = (255 - 215) = 40 , 40 x 40 = 1,600	
Distance G = (255 - 210) = 45 , 45 x 45 = 2,025	
Distance B = (255 - 214) = 41 , 41 x 41 = 1,681	
Distance RGB = sqrt(1,600 + 2,025 + 1,681) = 72.84	

ภาพ 3.7 ตัวอย่างการคำนวณระยะห่างแบบยูคลิดของค่า RGB ที่วัดได้กับค่า RGB ตั้งต้น

	องค์ประกอบสี	ภาพที่ 1	ภาพที่ 2	ภาพที่ 3	ภาพที่ 4	ภาพที่ 5	เฉลี่ย
ระดับเทาที่ 1	R	1	3	3	2	4	2.6
	G	8	2	1	3	0	2.8
	B	2	5	5	6	9	5.4
ระดับเทาที่ 2	R	90	92	90	94	90	91.2
	G	89	90	87	91	85	88.4
	B	94	88	88	90	86	89.2
ระดับเทาที่ 3	R	180	173	175	170	178	175.2
	G	179	174	177	180	179	177.8
	B	178	174	173	179	177	176.2
ระดับเทาที่ 4	R	246	245	251	245	249	247.2
	G	246	250	247	254	254	250.2
	B	249	251	252	247	246	249

ภาพ 3.8 ตัวอย่างการเฉลี่ยแต่ละองค์ประกอบสี

ตัวอย่างการคำนวณในภาพที่ 3.8 ทำการหาค่าเฉลี่ยของแต่ละองค์ประกอบ โดยค่าแต่ละองค์ประกอบได้มาจากเครื่องมือเพิ่มเติม (extension) บนเว็บเบราว์เซอร์โครม (chrome) จากภาพถ่าย

	องค์ประกอบสี	ตั้งต้น	เฉลี่ย	ความแตกต่าง	ระยะห่างแบบยุคลิดเฉลี่ยแต่ละระดับเทา
ระดับเทาที่ 1	R	0	2.6	2.6	6.615134163
	G	0	2.8	2.8	
	B	0	5.4	5.4	
ระดับเทาที่ 2	R	85	91.2	6.2	8.224354078
	G	85	88.4	3.4	
	B	85	89.2	4.2	
ระดับเทาที่ 3	R	170	175.2	5.2	11.23921705
	G	170	177.8	7.8	
	B	170	176.2	6.2	
ระดับเทาที่ 4	R	255	247.2	7.8	10.94897255
	G	255	250.2	4.8	
	B	255	249	6	
ระยะห่างแบบยุคลิดเฉลี่ยของระดับเทาทั้งหมด					8.692901765

ภาพ 3.9 ตัวอย่างการหาระยะห่างแบบยุคลิดแต่ละระดับเทาและระยะห่างแบบยุคลิดเฉลี่ยของระดับเทาทั้งหมด

เมื่อได้ผลค่าเฉลี่ยจากภาพ 3.8 แล้วนำมาคำนวณด้วยวิธีการคำนวณระยะห่างแบบยุคลิดของแต่ละระดับเทากับค่าตั้งต้น จากนั้นนำค่าระยะห่างของแต่ละระดับเทามาเฉลี่ย เพื่อหาระยะห่างแบบยุคลิดเฉลี่ยของระดับเทาทั้งหมด ดังภาพ 3.9 จากนั้นหาระยะห่างแบบยุคลิดเฉลี่ยของสีทั้งหมด ด้วยวิธีการเดียวกัน แล้วนำผลลัพธ์ระยะห่างแบบยุคลิดของสีและระดับเทามาเปรียบเทียบกัน

**การทดลองที่ 3** หาจำนวนระดับเทาที่เหมาะสมในการใช้งานคิวอาร์โค้ดระดับเทา

การทดลองนี้ต้องการหาจำนวนระดับเทาที่เหมาะสมในการใช้งานคิวอาร์โค้ดระดับเทา โดยจำนวนระดับเทาที่เหมาะสม หมายถึง เมื่อทำการถ่ายภาพด้วยอุปกรณ์ต่างๆ ค่า RGB ของระดับเทาแต่ละระดับ จะอยู่ในช่วงที่ไม่ซ้อนทับกับระดับเทาอื่น ตัวอย่างเช่น กรณี ระดับเทา 2 ระดับ ค่า RGB ที่เป็นไปได้ทั้งหมดของระดับเทาที่ 1 ทั้งหมดต้องมีค่าน้อยกว่าค่า RGB ที่เป็นไปได้ทั้งหมดของระดับเทาที่ 2 สมมติฐานการทดลอง คือ ระดับเทา 4 ระดับ เหมาะสมในการใช้งานคิวอาร์โค้ดระดับเทา

หลักการในการตรวจสอบความเหมาะสมของระดับเทา 4 ระดับ ดังนี้

$RGB_{Gray 1}$  คือค่า RGB ที่เป็นไปได้ของระดับเทาที่ 1

$RGB_{Gray 2}$  คือค่า RGB ที่เป็นไปได้ของระดับเทาที่ 2

$RGB_{Gray 3}$  คือค่า RGB ที่เป็นไปได้ของระดับเทาที่ 3

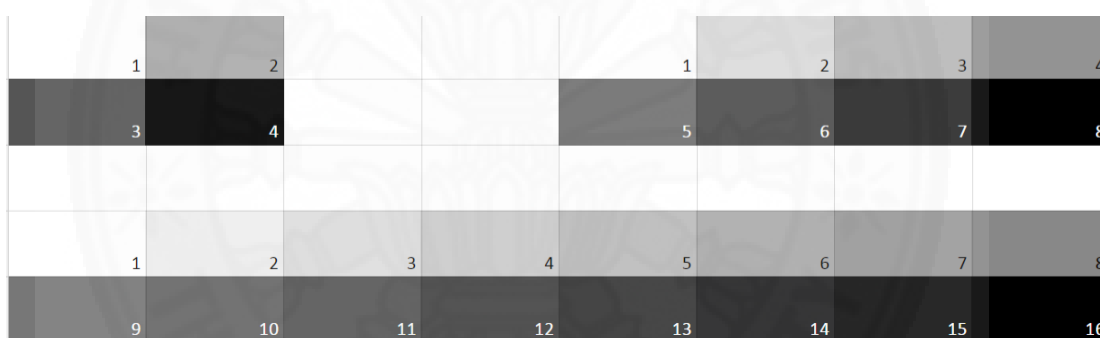
$RGB_{Gray 4}$  คือค่า RGB ที่เป็นไปได้ของระดับเทาที่ 4

$Max(RGB_{Gray 1}) < Min(RGB_{Gray 2})$

$Max(RGB_{Gray 2}) < Min(RGB_{Gray 3})$

$Max(RGB_{Gray 3}) < Min(RGB_{Gray 4})$

ในการทดลองนี้ จะแบ่งช่วงระดับเทา แต่ละช่วงเท่ากันตามจำนวนระดับเทาที่ต้องการ ดังสมการในการทดลองที่ 2 ภาพที่ 3.5



ภาพที่ 3.10 ลักษณะของกระดาษที่ใช้ในการทดลองที่ 3

จากภาพที่ 3.10 ค่า RGB เริ่มต้นของแต่ละระดับเทา มีดังนี้

ตารางที่ 3.3

ค่า RGB ของระดับเทา 4 8 และ 16 ระดับเทาที่ใช้ในการทดลอง

จำนวนระดับเทา	ค่า RGB	
4 ระดับ	(255 , 255 , 255)	(170 , 170 , 170)
	(85 , 85 , 85)	(0 , 0 , 0)
8 ระดับ	(255 , 255 , 255)	(219 , 219 , 219)
	(183 , 183 , 183)	(147 , 147 , 147)

จำนวนระดับเทา	ค่า RGB	
	(111 , 111 , 111)	(75 , 75 , 75)
	(39 , 39 , 39)	(0 , 0 , 0)
16 ระดับ	(255 , 255 , 255)	(238 , 238 , 238)
	(221 , 221 , 221)	(204 , 204 , 204)
	(187 , 187 , 187)	(170 , 170 , 170)
	(153 , 153 , 153)	(136 , 136 , 136)
	(119 , 119 , 119)	(102 , 102 , 102)
	(85 , 85 , 85)	(68 , 68 , 68)
	(51 , 51 , 51)	(34 , 34 , 34)
	(17 , 17 , 17)	(0 , 0 , 0)

ภาพที่ใช้ในการทดลองประกอบด้วย ค่า RGB ทั้งสามชุด โดยแสดงระดับเทาแต่ละระดับของแต่ละชุด ในลักษณะตาราง ดังภาพที่ 3.10 เก็บภาพด้วยสมาร์ทโฟน 3 รุ่น โดยรายละเอียดสมาร์ทโฟนทั้ง 3 รุ่นแสดงในภาคผนวก จ ในสมาร์ทโฟนแต่ละรุ่นจะเก็บภาพทั้งในสภาพแสงสีขาวและแสงสีส้ม ที่ได้จากหลอดไฟ Philips LED Bulb Scene Switch 9.5W อย่างละห้าภาพ

ในการทดลองนี้จะทำการเก็บค่า RGB ของระดับเทาจากภาพ โดยใช้เครื่องมือ ColorZilla ซึ่งเป็นเครื่องมือเพิ่มเติม (extension) บนเว็บเบราว์เซอร์โครม (chrome) จากนั้น หาค่าระดับเทาโดยการเฉลี่ย RGB ของแต่ละภาพ ดังตัวอย่างในภาพ 3.11 และเปรียบเทียบค่าระดับเทาที่มากที่สุดกับและระดับเทาที่น้อยที่สุดของระดับถัดไป

ระดับเทาที่ 1	R	G	B	Avg
ภาพที่ 1	172	165	157	164.67
ภาพที่ 2	176	165	156	165.67
ภาพที่ 3	172	164	154	163.33
ภาพที่ 4	172	164	156	164.00
ภาพที่ 5	173	166	149	162.67

ภาพที่ 3.11 ตัวอย่างการหาค่าเฉลี่ยมากที่สุดและค่าเฉลี่ยน้อยที่สุด



#### การทดลองที่ 4 หาความจุข้อมูลของคิวอาร์โค้ดระดับเทา

การทดลองนี้ต้องการหาความจุข้อมูลของคิวอาร์โค้ดระดับเทาเทียบกับคิวอาร์โค้ดมาตรฐาน โดยการเปรียบเทียบเวอร์ชันที่ใช้ในการเข้ารหัสคิวอาร์โค้ดระดับเทาและคิวอาร์โค้ดมาตรฐาน เมื่อจำนวนอักขระและระดับแก้ไขข้อผิดพลาดเดียวกัน การทดลองจะตัวอักษร A เป็นอินพุตเริ่มต้น จากนั้นทำการเข้ารหัสคิวอาร์โค้ดมาตรฐานและคิวอาร์โค้ดระดับเทา บันทึกเวอร์ชันที่น้อยที่สุดที่สามารถบรรจุข้อความได้ ซึ่งเวอร์ชันของคิวอาร์โค้ดระดับเทาจะเท่ากับเวอร์ชันของคิวอาร์โค้ดมาตรฐานที่มีจำนวนโมดูลมากกว่าหรือเท่ากับจำนวนโมดูลที่ใช้ในการเข้ารหัสคิวอาร์โค้ดระดับเทาเสมอ ดังแสดงในภาพที่ 3.12 จากนั้นเพิ่มความยาวของข้อความด้วยตัวอักษร A ทีละตัว จนกว่าข้อความจะยาวเกินข้อกำหนดของคิวอาร์โค้ดมาตรฐานทั้ง 40 เวอร์ชัน (คิวอาร์โค้ดมาตรฐาน เวอร์ชัน 40 ซึ่งจุข้อมูลได้สูงที่สุด 4296 ตัวอักษร) ในการทดลองนี้ จะทดลองโดยกำหนดระดับแก้ไขข้อผิดพลาด L, M, Q และ H ตามลำดับ สมมติฐานการทดลอง คือ เมื่อใช้จำนวนระดับเทาสี่ระดับ คิวอาร์โค้ดระดับเทาจะสามารถจุข้อมูลได้มากกว่าคิวอาร์โค้ดมาตรฐานประมาณสองเท่า

จำนวนอักษร	จำนวนโมดูลคิวอาร์โค้ดมาตรฐาน	เวอร์ชันคิวอาร์โค้ดมาตรฐาน	จำนวนโมดูลคิวอาร์โค้ดระดับเทา	เวอร์ชันคิวอาร์โค้ดระดับเทา
1	208	1	104	1
2	208	1	104	1
3	208	1	104	1
⋮				
25	208	1	104	1
26	352	2	176	1
27	352	2	176	1
28	352	2	176	1
29	352	2	176	1
30	352	2	176	1
⋮				
46	352	2	176	1
47	352	2	176	1
48	560	3	280	2
49	560	3	280	2
50	560	3	280	2
⋮				
4293	29648	40	14824	28
4294	29648	40	14824	28
4295	29648	40	14824	28
4296	29648	40	14824	28

ภาพที่ 3.12 ตัวอย่างการทดลองหาความจุคิวอาร์โค้ดระดับเทาที่ระดับแก้ไขข้อผิดพลาด L

จากภาพ 3.12 ตัวอย่างการทดลองเริ่มต้นด้วยตัวอักษรหนึ่งตัว เวอร์ชันที่ควอาร์โค้ดมาตรฐานใช้คือเวอร์ชัน 1 และเมื่อเข้ารหัสควอาร์โค้ดระดับเทาจะได้เวอร์ชัน 1 เช่นกัน เมื่อเพิ่มจำนวนตัวอักษรเป็น 26 ตัวเวอร์ชันที่ควอาร์โค้ดมาตรฐานใช้คือเวอร์ชัน 2 แต่เมื่อเข้ารหัสควอาร์โค้ดระดับเทาจะยังคงใช้เวอร์ชัน 1 เช่นเดิม และเมื่อเพิ่มจำนวนตัวอักษรไปจน 48 ตัว ควอาร์โค้ดมาตรฐานใช้เวอร์ชัน 3 แต่หากทำการเข้ารหัสด้วยควอาร์โค้ดระดับเทาจะใช้เวอร์ชัน 2 เปรียบเทียบลักษณะนี้ไปจนกว่าข้อความจะยาวเกินข้อกำหนดของควอาร์โค้ดมาตรฐานทั้ง 40 เวอร์ชัน



## บทที่ 4

### ผลการวิจัยและอภิปรายผล

งานวิจัยนี้มีวัตถุประสงค์เพื่อออกแบบรหัสคิวอาร์โค้ดระดับเทาเพื่อให้สามารถบรรจุข้อมูลได้มากขึ้นกว่ารหัสคิวอาร์โค้ดมาตรฐานที่ใช้กันอยู่ในปัจจุบัน ในบทนี้จะเป็นการกล่าวถึงผลการทดลองและอภิปรายผลการทดลอง เพื่อทราบถึงข้อดีและข้อจำกัดของคิวอาร์โค้ดระดับเทา โดยมีผลการวิเคราะห์ข้อมูลดังนี้

- 4.1. ผลการทดลองวัดความสามารถในการถอดรหัสคิวอาร์โค้ดมาตรฐาน
- 4.2. ผลการทดลองเปรียบเทียบค่าองค์ประกอบสี RGB ของคิวอาร์โค้ดระดับเทาและแบบสีที่เปลี่ยนแปลงไปเมื่อถ่ายภาพในสภาพแสงสีขาว (Daylight) และแสงสีส้ม (Warm White)
- 4.3. ผลการทดลองหาจำนวนระดับเทาที่เหมาะสมในการใช้งานคิวอาร์โค้ดระดับเทา
- 4.4. ผลการทดลองเพื่อหาหาความจุข้อมูลของคิวอาร์โค้ดระดับเทา

สัญลักษณ์ที่ใช้ในการวิเคราะห์ เพื่อให้เกิดความเข้าใจตรงกันในการแปลความหมายจากการวิเคราะห์ข้อมูล ผู้วิจัยจึงกำหนดความหมายของสัญลักษณ์ดังนี้

- Sig. แทน ระดับนัยสำคัญทางสถิติ (Significances)  
 \* แทน ความมีนัยสำคัญทางสถิติที่ระดับความเชื่อมั่น 0.05  
 v. แทน เวอร์ชันของรหัสคิวอาร์โค้ด

#### 4.1 ผลการทดลองข้อจำกัดเกี่ยวกับขนาดของภาพกับเวอร์ชันของคิวอาร์โค้ดมาตรฐาน

การทดลองนี้ต้องการหาข้อจำกัดเกี่ยวกับขนาดของภาพกับเวอร์ชันของคิวอาร์โค้ดมาตรฐาน โดยมีสมมติฐานคือ เมื่อขนาดภาพใหญ่ขึ้นทำให้มีโอกาสถอดรหัสสำเร็จมากขึ้น และที่ขนาดภาพคงที่คิวอาร์โค้ดมาตรฐานเวอร์ชันต่ำกว่าจะมีโอกาสถอดรหัสสำเร็จมากกว่า และภายใต้สภาพแสงสีขาวจะมีโอกาสถอดรหัสสำเร็จมากกว่าสภาพแสงสีส้ม

จากการนำภาพรหัสคิวอาร์โค้ดมาตรฐานเวอร์ชัน 2 5 10 และ 15 ที่ระดับแก้ไขข้อผิดพลาด L ที่ถ่ายได้จากอุปกรณ์ถ่ายภาพ ในสภาพแสงสีขาว (Daylight) และแสงสีส้ม (Warm White) และถ่ายภาพด้วยสมาร์ตโฟนมาทำการถอดรหัสด้วยอัลกอริธึมการถอดรหัสของคิวอาร์โค้ดมาตรฐาน

ผลการทดลองภาพรหัสคิวอาร์โค้ดมาตรฐานที่สามารถถอดรหัสของทุกภาพที่ได้จากทุกอุปกรณ์มาคำนวณหาร้อยละ โดยแบ่งตามเวอร์ชันของรหัสคิวอาร์โค้ด ได้ผลตามตารางต่อไปนี้ (ผลการทดลองโดยละเอียดแสดงใน ภาคผนวก ข)

## ตารางที่ 4.1

ผลการถอดรหัสคิวอาร์โค้ดมาตรฐานระดับความผิดพลาด L ที่ถ่ายได้จากสมาร์ทโฟน

เวอร์ชัน	สภาพแสง	1 นิ้ว	2 นิ้ว	3 นิ้ว
2	สีขาว	66.67%	73.33%	73.33%
	สีส้ม	53.33%	60%	73.33%
5	สีขาว	60%	86.67%	86.67%
	สีส้ม	33.33%	80%	80%
10	สีขาว	60%	93.33%	100%
	สีส้ม	46.67%	73.33%	93.33%
15	สีขาว	20%	33.33%	40%
	สีส้ม	20%	20%	33.33%

จากผลการทดลองดังตาราง 4.1 พบว่า ภาพคิวอาร์โค้ดขนาดใหญ่มีโอกาสถอดรหัสได้มากกว่าขนาดเล็ก สภาพแสงสีขาวมีโอกาสถอดรหัสได้มากกว่าสภาพแสงสีส้ม หากภาพคิวอาร์โค้ดขนาดเท่ากัน ภาพที่เวอร์ชันต่ำกว่า มีโอกาสถอดรหัสได้มากกว่า ซึ่งตรงตามสมมติฐาน แต่มีบางกรณี ภาพที่เวอร์ชันต่ำกว่าไม่ได้มีโอกาสถอดรหัสได้มากกว่าเวอร์ชันสูงเมื่อขนาดรหัสคิวอาร์โค้ดเท่ากัน ได้แก่ เวอร์ชัน 5 สามารถถอดรหัสได้มากกว่าเวอร์ชัน 2 ที่ภาพขนาดสองนิ้วและสามนิ้ว รวมถึงเวอร์ชัน 10 สามารถถอดรหัสได้มากกว่าเวอร์ชัน 5 ที่ภาพขนาดสองนิ้วและสามนิ้ว ซึ่งไม่ตรงตามสมมติฐาน

#### 4.2 ผลการทดลองเปรียบเทียบค่าองค์ประกอบ RGB ของสีและระดับเทาที่เปลี่ยนแปลงไปในสภาพแสงสีขาวและแสงสีส้ม

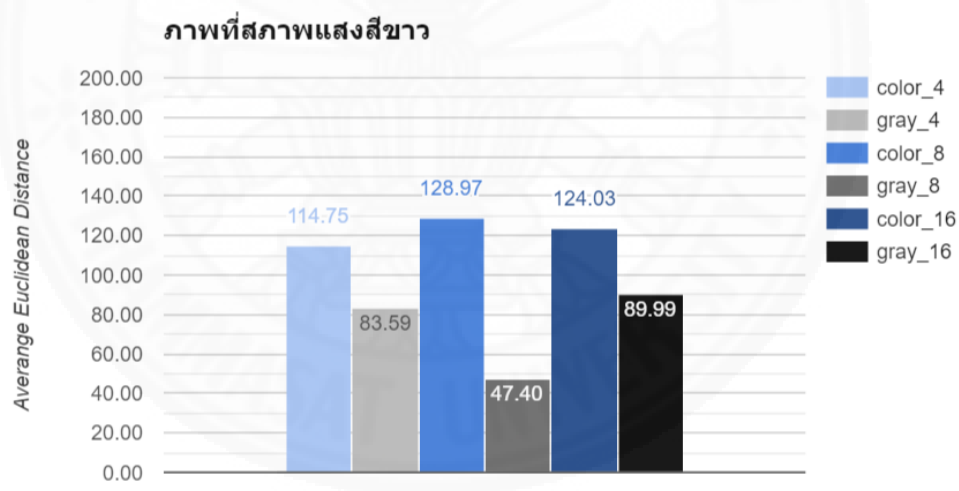
การทดลองนี้มีสมมติฐาน คือ ค่าองค์ประกอบ RGB ของระดับเทาเปลี่ยนแปลงน้อยกว่าค่าองค์ประกอบ RGB ของสีทั้งสภาพแสงสีขาวและแสงสีส้ม

ผลการทดลองแสดงผลต่างเฉลี่ยของค่าองค์ประกอบสี RGB จากภาพถ่ายกับค่าตั้งต้น ทั้งระดับเทาและสี ด้วยวิธีการคำนวณระยะทางแบบยูคลิด (Euclidean Distance) ดังนี้

## ตารางที่ 4.2

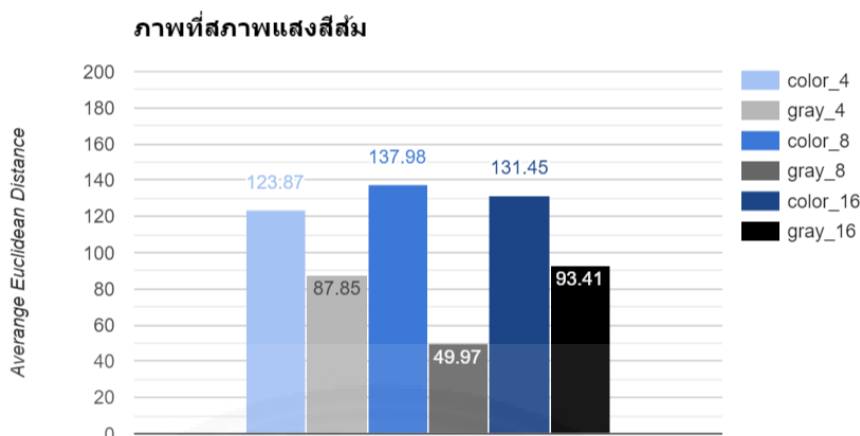
ค่าองค์ประกอบสี RGB ที่เปลี่ยนแปลงไปของสีที่ใช้ในควอาร์โค้ดสีและระดับเทาที่ใช้ในควอาร์โค้ดระดับเทา จากภาพที่ถ่ายจากสมาร์ทโฟน

ประเภทสี	ค่า RGB ที่เปลี่ยนแปลงไปในสภาพแสงโดยเฉลี่ย	
	แสงสีขาว	แสงสีส้ม
4 สี	114.75	123.87
4 ระดับเทา	83.59	87.85
8 สี	128.97	137.98
8 ระดับเทา	47.40	49.97
16 สี	124.03	131.45
16 ระดับเทา	89.99	93.41



ภาพที่ 4.1 แสดงการเปรียบเทียบค่าองค์ประกอบสี RGB ที่เปลี่ยนแปลงไปของสีและระดับเทาจากภาพถ่ายจากกล้องสมาร์ทโฟน ในสภาพแสงสีขาว

จากภาพที่ 4.1 สรุปได้ว่าภาพระดับเทามีการเปลี่ยนแปลงของค่า RGB จากภาพถ่ายน้อยกว่าสีที่สภาพแสงสีขาวทั้ง ชุด 4, 8 และ 16 ระดับเทา



ภาพที่ 4.2 แสดงการเปรียบเทียบค่าองค์ประกอบสี RGB ที่เปลี่ยนแปลงไปของสีและระดับเทาจากภาพถ่ายจากกล้องสมาร์ทโฟน ในสภาพแสงสีส้ม

จากภาพที่ 4.2 สรุปได้ว่าระดับเทามีการเปลี่ยนแปลงของค่าองค์ประกอบสี RGB จากภาพถ่ายน้อยกว่าสีที่สภาพแสงสีส้มทั้ง ชุด 4, 8 และ 16 ระดับเทา

เปรียบเทียบความต่างค่าองค์ประกอบ RGB ระหว่างระดับเทาและสี แบบแยกองค์ประกอบสี ได้แก่ องค์ประกอบ R, G และ B ในสภาพแสงสีขาวและแสงสีส้ม เพื่อตรวจสอบนัยสำคัญของค่าความต่างของแต่ละองค์ประกอบสี ระหว่างระดับเทาและสี ด้วยวิธี Paired T-Test ได้ผลตามตารางดังนี้

ตารางที่ 4.3

ค่า Paired T-Test ระหว่างความแตกต่างขององค์ประกอบ R G B ของสีและระดับเทาในสภาพแสงสีขาวและแสงสีส้ม

องค์ประกอบสี	สภาพแสงสีขาว	สภาพแสงสีส้ม
R	0.000*	0.000*
G	0.000*	0.000*
B	0.000*	0.000*

Sig < 0.05\*

จากตาราง 4.3 สรุปได้ว่าค่าความแตกต่างขององค์ประกอบ R G B ของภาพสีแตกต่างกับภาพระดับเทาแบบมีนัยสำคัญ

จากตาราง 4.2 – 4.3 และ ภาพที่ 4.1 – 4.2 แสดงให้เห็นว่า ค่าองค์ประกอบ RGB ของระดับเทาเปลี่ยนแปลงจากค่าตั้งต้นน้อยกว่าค่าองค์ประกอบ RGB ของสี ทั้งในสภาพแสงสีขาวและแสงสีส้ม ซึ่งเป็นไปตามสมมติฐานที่ตั้งไว้

การทดลองนี้แสดงให้เห็นถึงข้อดีของระดับเทา คือ มีการเปลี่ยนแปลงค่าองค์ประกอบน้อยกว่าสี ดังนั้น หากใช้เพิ่มความจุกวอาร์โค้ดโดยใช้ระดับเทา จะทำให้มีโอกาสถอดรหัสผิดพลาดน้อยกว่าแบบใช้สี ในการทดลองถัดไปจะหาจำนวนระดับเทาที่เหมาะสม เนื่องจากจำนวนระดับเทามากขึ้น จะทำให้ข้อมูลได้มากขึ้น แต่จำนวนระดับเทาที่มากขึ้นก็อาจทำให้ถอดรหัสผิดพลาดได้มากขึ้นเช่นกัน

#### 4.3 ผลการทดลองจำนวนระดับเทาที่เหมาะสมในการใช้งานคิวอาร์โค้ดระดับเทา

การทดลองนี้มีสมมติฐาน คือ ระดับเทาที่เหมาะสมในการใช้งานคิวอาร์โค้ดระดับเทา คือ 4 ระดับ

จากการนำภาพถ่ายที่ประกอบด้วยช่องที่มีระดับเทา 4 8 และ 16 ระดับ โดยภาพถ่ายได้จากสมาร์ตโฟนสามรุ่นมาวิเคราะห์หาค่า RGB ในสภาพแสงสีขาวและแสงสีส้ม ด้วย Google Chrome Extension ชื่อว่า Colorzilla บนเครื่องคอมพิวเตอร์ เพื่อหาจำนวนระดับเทาที่เหมาะสม โดยหมายถึง เมื่อทำการถ่ายภาพด้วยอุปกรณ์ต่างๆ ค่า RGB ของระดับเทาแต่ละระดับ จะอยู่ในช่วงที่ไม่ซ้อนทับกับระดับเทาอื่น โดยผลการทดลอง ดังนี้

ตารางที่ 4.4

ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 4 ระดับ จากภาพที่ถ่ายโดยสมาร์ตโฟน A

ระดับเทาที่	สภาพแสงสีขาว		สภาพแสงสีส้ม	
	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)
1	171	180	163	172
2	131	138	114	128
3	50	58	48	60
4	30	40	28	33

ตารางที่ 4.5

ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 4 ระดับ จากภาพที่ถ่ายโดยสมาร์ทโฟน B

ระดับเทาที่	สภาพแสงสีขาว		สภาพแสงสีส้ม	
	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)
1	138	156	131	155
2	96	120	93	114
3	40	47	34	35
4	19	33	20	29

ตารางที่ 4.6

ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 4 ระดับ จากภาพที่ถ่ายโดยสมาร์ทโฟน C

ระดับเทาที่	สภาพแสงสีขาว		สภาพแสงสีส้ม	
	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)
1	189	210	200	211
2	117	141	120	131
3	25	40	29	42
4	5	17	11	17

จากตาราง 4.4 – 4.6 ทำให้พบว่าที่ภาพระดับเทา 4 ระดับที่ถ่ายด้วยสมาร์ทโฟนทั้ง 3 รุ่น ช่วงค่า RGB ของทั้ง 4 ระดับเทาไม่มีการซ้อนทับกับระดับเทาอื่นทั้งสภาพแสงสีขาวและแสงสีส้ม แสดงให้เห็นว่าที่ระดับเทา 4 ระดับสามารถถอดรหัสคิวอาร์โค้ดระดับเทาได้โดยผลลัพธ์จะไม่ผิดเพี้ยนไปและเหมาะสมที่จะนำมาใช้ในคิวอาร์โค้ดระดับเทา ในขณะที่ 8 และ 16 ระดับมีการซ้อนทับในบางช่วง

เมื่อนำผลของระดับเทาทั้งสามชุดที่ได้จากทุกอุปกรณ์ได้ผลการทดลองดังแสดงในตาราง 4.7 ดังนี้ (ผลการทดลองโดยละเอียดของระดับเทา 8 ระดับ และ 16 ระดับสามารถดูได้ในภาคผนวก ค)



## ตารางที่ 4.7

ผลการทดลองจำนวนระดับเทาที่เหมาะสมสำหรับคิวอาร์โค้ดระดับเทา

จำนวน ระดับเทา	สภาพแสงสีขาว			สภาพแสงสีส้ม		
	เครื่อง A	เครื่อง B	เครื่อง C	เครื่อง A	เครื่อง B	เครื่อง C
4	○	○	○	○	○	○
8	○	X	○	X	X	○
16	X	X	X	X	X	X

หมายเหตุ ○ คือ ระดับเทาไม่ซ้อนทับกัน , X คือ ระดับเทามีการซ้อนทับกัน

จากตาราง 4.7 และตารางผลการทดลองโดยละเอียดที่ภาคผนวก ค พบว่าช่วงค่า RGB ของแต่ละระดับจากภาพระดับเทา 4 ระดับ ไม่มีช่วงใดที่ซ้อนทับกัน แต่ช่วงค่า RGB ของบางระดับจากภาพระดับเทา 8 ระดับ และ 16 ระดับ ซ้อนทับกับระดับอื่น จึงสรุปได้ว่า การสร้างคิวอาร์โค้ดระดับเทา ควรใช้ระดับเทา 4 ระดับ

การทดลองนี้แสดงให้เห็นว่าจำนวนระดับเทาที่เหมาะสมกับการสร้างคิวอาร์โค้ดระดับเทา คือ 4 ระดับ ดังนั้นในการทดลองถัดไปจะทดลองเพื่อหาความจุข้อมูลของคิวอาร์โค้ดระดับเทาเมื่อใช้จำนวนระดับเทา 4 ระดับ

#### 4.4 ผลการทดลองเพื่อหาความสามารถจุข้อมูลของคิวอาร์โค้ดระดับเทาเปรียบเทียบกับคิวอาร์โค้ดมาตรฐาน

จากผลการทดลองจำนวนระดับเทาที่เหมาะสมในการใช้งานคิวอาร์โค้ดระดับเทา สามารถสรุปได้ว่า จำนวนระดับเทาที่เหมาะสมกับสร้างคิวอาร์โค้ดระดับเทานั้น คือ 4 ระดับ เนื่องจากการใช้ระดับเทาสูงกวานั้น อาจทำให้มีการเหลื่อมกันของช่วงระดับเทาที่ใกล้เคียงกัน ทำให้มีโอกาสถอดรหัสออกมาผิดพลาด การทดลองความสามารถในการจุข้อมูลของคิวอาร์โค้ดระดับเทา จึงใช้ระดับเทา 4 ระดับในการทดลอง ผลการทดลองโดยสรุป ดังนี้ (ผลการทดลองโดยละเอียดแสดงไว้ที่ภาคผนวก ง)

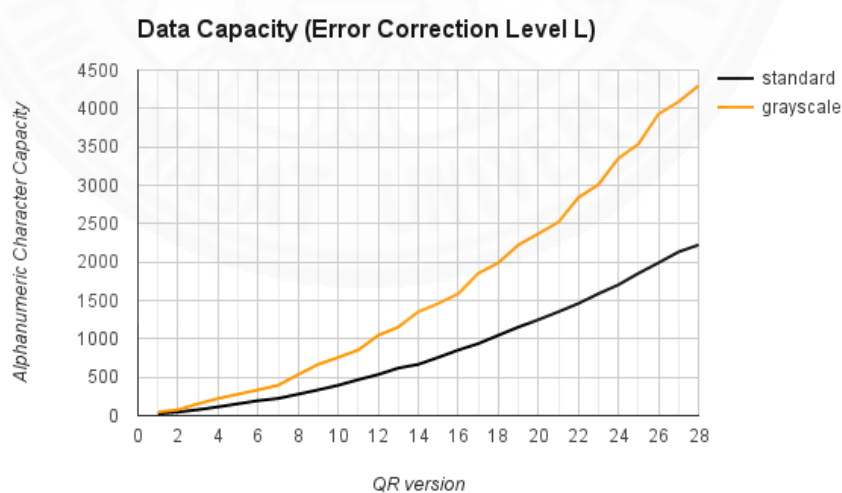
## ตารางที่ 4.8

สรุปผลการทดลองความจุที่เพิ่มขึ้นเมื่อเข้ารหัสคิวอาร์โค้ดระดับเทา 4 ระดับเทียบกับความจุของคิวอาร์โค้ดมาตรฐาน

ระดับแก้ไขข้อผิดพลาด	ความจุเพิ่มขึ้น (%)		
	เฉลี่ย	มากที่สุด	น้อยที่สุด
L	189.87	202.70 (v.14)	163.83 (v. 2)
M	190.11	201.53 (v. 9)	160.53 (v. 2)
Q	188.81	199.36 (v. 26)	162.07 (v. 2)
H	190.03	207.42 (v. 14)	170.24 (v. 6)

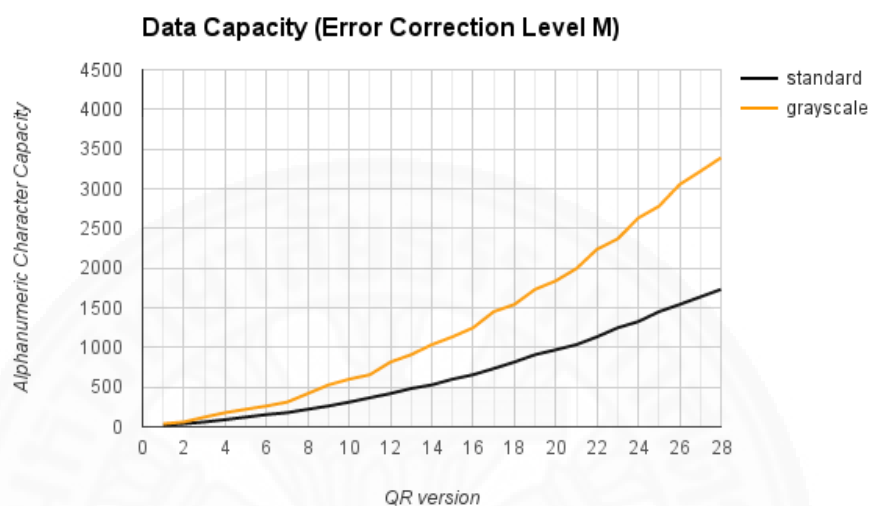
จากการเปรียบเทียบความจุ ในตาราง 4.8 สามารถสรุปได้ว่า คิวอาร์โค้ดระดับเทา 4 ระดับสามารถจุข้อมูลได้มากกว่าคิวอาร์โค้ดมาตรฐานในเวอร์ชันและระดับแก้ไขข้อผิดพลาดเดียวกัน ประมาณ 2 เท่าโดยเฉลี่ยทุกระดับแก้ไขข้อผิดพลาด (189.70%)

เมื่อพิจารณาแต่ละระดับแก้ไขข้อผิดพลาด กราฟเปรียบเทียบความจุข้อมูลคิวอาร์โค้ดมาตรฐานและคิวอาร์โค้ดระดับเทามีลักษณะ ดังนี้



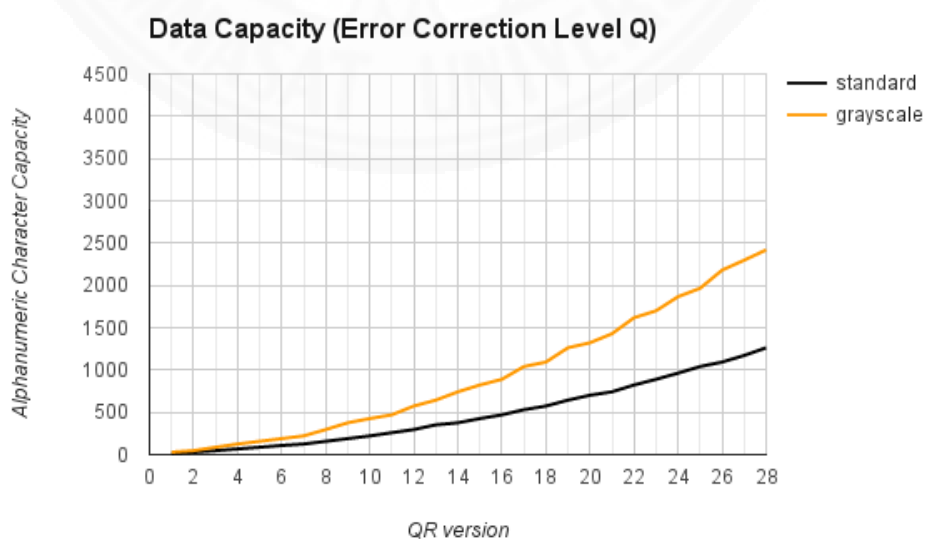
ภาพที่ 4.3 ความจุข้อมูลคิวอาร์โค้ดมาตรฐานและคิวอาร์โค้ดระดับเทาที่ระดับแก้ไขข้อผิดพลาด L

จากภาพ 4.3 แสดงแนวโน้มผลความจุข้อมูลของคิวอาร์โค้ดมาตรฐานและคิวอาร์โค้ดระดับเทา โดยคิวอาร์โค้ดระดับเทาสามารถจุข้อมูลได้มากกว่าคิวอาร์โค้ดมาตรฐานประมาณ 2 เท่าในทุกเวอร์ชันที่ระดับแก้ไขข้อผิดพลาด L



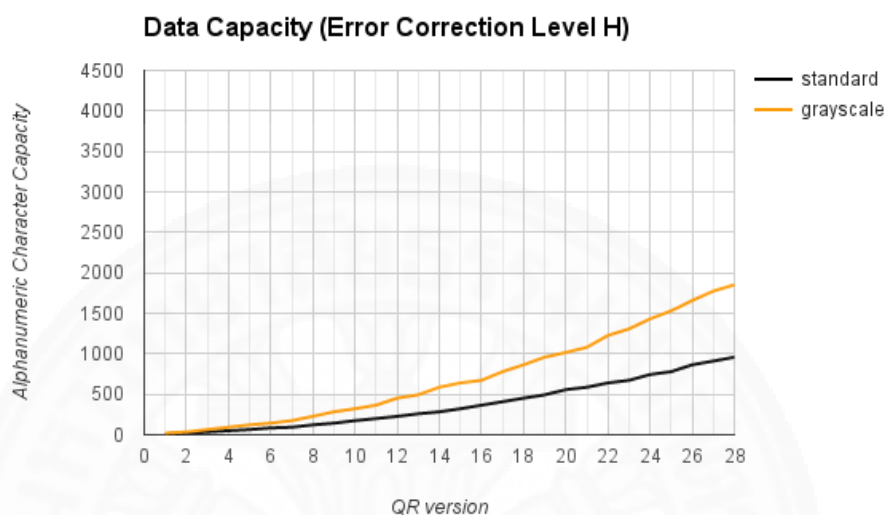
ภาพที่ 4.4 ความจุข้อมูลคิวอาร์โค้ดมาตรฐานและคิวอาร์โค้ดระดับเทาที่ระดับแก้ไขข้อผิดพลาด M

จากภาพ 4.4 แสดงแนวโน้มผลความจุข้อมูลของคิวอาร์โค้ดมาตรฐานและคิวอาร์โค้ดระดับเทา โดยคิวอาร์โค้ดระดับเทาสามารถจุข้อมูลได้มากกว่าคิวอาร์โค้ดมาตรฐานประมาณ 2 เท่าในทุกเวอร์ชันที่ระดับแก้ไขข้อผิดพลาด M



ภาพที่ 4.5 ความจุข้อมูลคิวอาร์โค้ดมาตรฐานและคิวอาร์โค้ดระดับเทาที่ระดับแก้ไขข้อผิดพลาด Q

จากภาพ 4.5 แสดงแนวโน้มผลความจุข้อมูลของคิวอาร์โค้ดมาตรฐานและคิวอาร์โค้ดระดับเทา โดยคิวอาร์โค้ดระดับเทาสามารถจุข้อมูลได้มากกว่าคิวอาร์โค้ดมาตรฐานประมาณ 2 เท่าในทุกเวอร์ชันที่ระดับแก้ไขข้อผิดพลาด Q



ภาพที่ 4.6 เปรียบเทียบความจุข้อมูลคิวอาร์โค้ดมาตรฐานและคิวอาร์โค้ดระดับเทาที่ระดับแก้ไขข้อผิดพลาด H

จากภาพ 4.6 แสดงแนวโน้มผลความจุข้อมูลของคิวอาร์โค้ดมาตรฐานและคิวอาร์โค้ดระดับเทา โดยคิวอาร์โค้ดระดับเทาสามารถจุข้อมูลได้มากกว่าคิวอาร์โค้ดมาตรฐานประมาณ 2 เท่าในทุกเวอร์ชันที่ระดับแก้ไขข้อผิดพลาด H

จากการทดลองเปรียบเทียบความสามารถสรุปได้ว่า คิวอาร์โค้ดระดับเทา 4 ระดับ สามารถจุข้อมูลได้มากกว่าคิวอาร์โค้ดมาตรฐานในเวอร์ชันและระดับแก้ไขข้อผิดพลาดเดียวกันประมาณ 2 เท่า โดยเฉลี่ยทุกระดับแก้ไขข้อผิดพลาด (189.70%) ซึ่งเป็นไปตามสมมติฐานที่ตั้งไว้

### สรุปผลการทดลอง

จากการทดลองทั้ง 4 การทดลองพบว่า เมื่อภาพคิวอาร์โค้ดขนาดใหญ่มีโอกาสถอดรหัสได้มากกว่าขนาดเล็ก รวมถึงเมื่อถอดรหัสภาพที่สภาพแสงสีขาวมีโอกาสถอดรหัสได้มากกว่าสภาพแสงสีส้ม อย่างไรก็ตามเวอร์ชันของคิวอาร์โค้ดไม่ได้มีผลต่อโอกาสที่จะถอดรหัสได้

ระดับเทามีการเปลี่ยนแปลงค่าองค์ประกอบสี RGB น้อยกว่าสีทั้งสภาพแสงสีขาว (Daylight) และแสงสีส้ม (Warm White) และค่าความแตกต่างขององค์ประกอบ RGB ของภาพสีแตกต่างกับภาพระดับเทาแบบมีนัยสำคัญ

การเข้ารหัสคิวอาร์โค้ดระดับเทาควรใช้ระดับเทา 4 ระดับ เพราะค่าองค์ประกอบ RGB ของแต่ละช่วงระดับเทาไม่ซ้อนทับกับช่วงระดับเทาใกล้เคียง ทำให้ถอดรหัสได้ถูกต้อง ซึ่งสามารถจุข้อมูลได้มากกว่าคิวอาร์โค้ดมาตรฐานในเวอร์ชันและระดับแก้ไขข้อผิดพลาดเดียวกันประมาณ 2 เท่าโดยเฉลี่ยทุกระดับแก้ไขข้อผิดพลาด



## บทที่ 5

### สรุปผลการวิจัยและข้อเสนอแนะ

วิทยานิพนธ์ฉบับนี้ต้องการเพิ่มความจุกิวอาร์โค้ดมาตรฐานด้วยการใช้ระดับเทา ในบทนี้จะกล่าวถึงการสรุปการศึกษาวิจัย สรุปผลการทดลอง และข้อเสนอแนะแนวทางในการศึกษาวิจัยต่อไปในอนาคต

#### 5.1 สรุปการศึกษาวิจัย

วิทยานิพนธ์ฉบับนี้ต้องการเพิ่มความจุกิวอาร์โค้ดด้วยการใช้ระดับเทา วิธีการสร้างคิวอาร์โค้ดระดับเทาโดยพัฒนาเพิ่มเติมจากอัลกอริธึมสร้างคิวอาร์โค้ดมาตรฐาน (ISO/IEC, 2006) การสรุปการศึกษาวิจัย แบ่งออกเป็นสองส่วน คือวัตถุประสงค์ของการศึกษา และวิธีการศึกษาวิจัย ดังนี้

##### 5.1.1 วัตถุประสงค์ของการศึกษา

งานวิจัยฉบับนี้ได้บรรลุวัตถุประสงค์ที่กำหนดขึ้น คือ

1. สามารถเข้ารหัสและถอดรหัสคิวอาร์โค้ดระดับเทา ซึ่งเป็นวิธีการเพิ่มความจุกิวอาร์โค้ดที่นำเสนอ
2. สามารถหาจำนวนระดับเทาที่เหมาะสมสำหรับคิวอาร์โค้ดระดับเทา
3. เปรียบเทียบผลลัพธ์ข้อเด่นข้อด้อยของคิวอาร์โค้ดระดับเทากับวิธีการเพิ่มความจุกิวอาร์โค้ดของงานวิจัยที่เกี่ยวข้อง

##### 5.1.2 วิธีศึกษาวิจัย

งานวิจัยฉบับนี้ศึกษาการเข้ารหัสและถอดรหัสคิวอาร์โค้ดระดับเทา โดยมีขั้นตอนและหลักการโดยสรุป ดังนี้

1. ขั้นตอนการเข้ารหัสคิวอาร์โค้ดระดับเทา

รับข้อมูลเข้าเป็นตัวอักษร ทำการเข้ารหัสด้วยอัลกอริธึมของคิวอาร์โค้ดมาตรฐาน (ISO/IEC, 2006) จนถึงขั้นตอนแปลงข้อมูลเข้าเป็นไบนารี จากนั้นจัดข้อมูลไบนารีเป็นกลุ่ม กลุ่มละสองบิต หลังจากนั้น สร้างภาพคิวอาร์โค้ดระดับเทา 4 ระดับ ด้วยวิธีการสร้างภาพที่พัฒนาจากอัลกอริธึมสำหรับคิวอาร์โค้ดมาตรฐาน โดย

การใช้ระดับเทาที่หนึ่งในโมดูลของภาพสำหรับกลุ่มข้อมูลไบนารีที่มีข้อมูล 00 ระดับเทาที่สองสำหรับกลุ่มที่มีข้อมูล 01 ระดับเทาที่สามสำหรับกลุ่มที่มีข้อมูล 10 และระดับเทาที่สี่สำหรับกลุ่มที่มีข้อมูล 11

## 2. ขั้นตอนการถอดรหัสคิวอาร์โค้ด

รับภาพคิวอาร์โค้ดระดับเทา ทำการถอดรหัสโดยการอ่านค่าองค์ประกอบสี RGB ของแต่ละโมดูล การอ่านค่า RGB จะเริ่มอ่านจากโมดูลมุมล่างขวาของภาพ ตามทิศทางที่กำหนดไว้ของคิวอาร์โค้ดมาตรฐาน (ISO/IEC, 2006) จากนั้นหารเฉลี่ยค่าองค์ประกอบสี RGB นำค่าเฉลี่ยที่ได้ เทียบกับช่วงค่า RGB ของระดับเทาแต่ละระดับ เมื่อได้ระดับเทาของแต่ละโมดูล แปลงระดับเป็นข้อมูลไบนารี โดยระดับเทาที่หนึ่งจะถูกแปลงเป็นข้อมูลไบนารี 00 ระดับเทาที่สองจะถูกแปลงเป็นข้อมูลไบนารี 01 ระดับเทาที่สามจะถูกแปลงเป็นข้อมูลไบนารี 10 และระดับเทาที่สี่จะถูกแปลงเป็นข้อมูลไบนารี 11 เมื่อแปลงข้อมูลเป็นไบนารีครบทุกโมดูลแล้ว ขั้นตอนต่อไปคือการแปลงข้อมูลไบนารีเป็นอักษร ด้วยวิธีการของคิวอาร์โค้ดมาตรฐาน

## 5.2 สรุปผลการทดลอง

จากการทดลองพบว่าเมื่อภาพคิวอาร์โค้ดขนาดใหญ่ขึ้นจะมีโอกาสถูกถอดรหัสได้มากกว่าขนาดเล็ก รวมถึงเมื่อภาพที่ถ่ายในสภาพแสงสีขาว (Daylight) มีโอกาสถอดรหัสได้มากกว่า สภาพแสงสีส้ม (Warm White) และค่า RGB จากภาพถ่ายของระดับเทา มีการเปลี่ยนแปลงจากค่า RGB ตั้งต้นที่กำหนด น้อยกว่าแบบสีของ Grillo et al. (Grillo, Lentini & Italiano, 2010) อย่างมีนัยสำคัญ ทั้งในสภาพแสงสีขาว และแสงสีส้ม ดังนั้น การเพิ่มความจุคิวอาร์โค้ด โดยการใช้ระดับเทาจะมีโอกาสถอดรหัสผิดพลาดน้อยกว่าการเพิ่มความจุคิวอาร์โค้ด ด้วยวิธีการใช้สี

การเข้ารหัสคิวอาร์โค้ดระดับเทาควรใช้ระดับเทา 4 ระดับ เนื่องจากในการทดลอง พบว่าหากใช้ระดับเทา 8 ระดับ หรือ 16 ระดับ เมื่ออ่านค่า RGB การถ่ายภาพ จะไม่สามารถระบุระดับเทาที่ถูกต้องได้ คิวอาร์โค้ดระดับเทา 4 ระดับที่นำเสนอ สามารถจุข้อมูลได้มากกว่าคิวอาร์โค้ดมาตรฐานโดยเฉลี่ยประมาณ 2 เท่า ในเวอร์ชันและระดับแก้ไขข้อผิดพลาดเดียวกัน

การเพิ่มความจุคิวอาร์โค้ด โดยการใช้ระดับเทา 4 ระดับ ทำให้สามารถจุข้อมูลได้มากขึ้นประมาณ 2 เท่า เมื่อเทียบกับคิวอาร์โค้ดมาตรฐาน บนพื้นที่ขนาดเดียวกัน ความจุที่เพิ่มขึ้นใกล้เคียงกับวิธีการเพิ่มความจุคิวอาร์โค้ดโดยใช้สีจากวิธีการของ Grillo et al. (Grillo, Lentini & Italiano, 2010) แบบ 4 สี แต่การใช้ระดับเทา มีโอกาสถอดรหัสผิดพลาดน้อยกว่าสี ตลอดจนการพิมพ์คิวอาร์โค้ดระดับเทา มีค่าใช้จ่ายน้อยกว่าแบบสีและเท่ากับค่าใช้จ่ายที่ใช้ในการพิมพ์คิวอาร์โค้ดมาตรฐาน

แบบขาว-ดำ ดังนั้นจึงเป็นอีกทางเลือกหนึ่งของผู้ที่ต้องการบรรจุข้อมูลคิวอาร์โค้ดบนพื้นที่จำกัดและประหยัดค่าใช้จ่าย

### 5.3 ข้อเสนอแนะและแนวทางวิจัยต่อไปในอนาคต

1. การทำ masking ในคิวอาร์โค้ดมาตรฐาน เพื่อให้จำนวนโมดูลขาวและดำมีจำนวนใกล้เคียงกันซึ่งทำให้ถอดรหัสได้ถูกต้องมากขึ้น งานวิจัยนี้ยังไม่มีการทำ masking เพื่อให้ระดับเทาแต่ละระดับมีจำนวนใกล้เคียงกัน ซึ่งอาจทำให้สามารถรหัสได้ถูกต้องมากขึ้น
2. การแก้ไขข้อผิดพลาด (error correction) ซึ่งทำให้สามารถกู้คืนข้อมูลจากภาพคิวอาร์โค้ดเสียหาย หรือไม่ชัดเจน ได้ 7-25% ตามระดับแก้ไขข้อผิดพลาด แต่เนื่องจากวิธีการของคิวอาร์โค้ดมาตรฐานอาศัยข้อมูลไบนารีแต่ละบิต แต่ในงานวิจัยนี้ได้ทำการจัดกลุ่มข้อมูลไบนารีเป็นกลุ่ม กลุ่มละ 2 บิต วิธีการของคิวอาร์โค้ดมาตรฐานจึงไม่สามารถใช้ได้
3. การถอดรหัสคิวอาร์โค้ดระดับเทาในงานวิจัยนี้ ยังไม่มีการคำนึงถึงการหมุน เอียง หรือการบิดเบี้ยวของภาพ ผู้วิจัยคาดว่าวิธีการปรับแก้การหมุน เอียง และการบิดเบี้ยวของคิวอาร์โค้ดมาตรฐาน สามารถนำมาใช้ในกระบวนการถอดรหัสคิวอาร์โค้ดระดับเทาได้
4. ถ้าออกแบบให้มีแถบแสดงระดับเทาทั้งหมดที่ใช้ในภาพรหัสคิวอาร์โค้ดระดับเทานั้น ผู้วิจัยคาดว่าจะสามารถช่วยในการถอดรหัสภาพคิวอาร์โค้ดระดับเทาให้ง่ายและแม่นยำมากขึ้น แม้ว่าค่าองค์ประกอบสี RGB ของระดับเทาจะเปลี่ยนแปลงไปตามสภาพแสงต่างๆ



## รายการอ้างอิง

### Book

- ISO/IEC, Information technology (2006). Automatic identification and data capture techniques QR code 2005 bar code symbology specification, IS 18004:2006.
- Johnson, Stephen. (2006). Stephen Johnson on digital photography, O'Reilly Media, Inc.
- Charles, Poynton (2003). Digital Video and HDTV, (pp. 291-292). United States: Morgan Kaufmann Publishers.

### Conferences

- Kikuchi, M., Fujiyoshi, M., & Kiya, H. (2013, November). A new color QR code forward compatible with the standard QR code decoder. In Intelligent Signal Processing and Communications Systems (ISPACS), 2013 International Symposium on (pp. 26-31). IEEE.
- Grillo, A., Lentini, A., Querini, M., & Italiano, G. F. (2010, October). High capacity colored two dimensional codes. In Computer Science and Information Technology (IMCSIT), Proceedings of the 2010 International Multiconference on (pp. 709-716). IEEE.
- Furumoto K, Watanabe Y & Morii M. (2012, Nov). Gray scale two-dimensional code and its application. In The Institute of Electronics, Information and Communication Engineers (IEICE Tech) on (pp. 364-369).
- Mamidi, Suman, et al. "Instruction set extensions for Reed-Solomon encoding and decoding." Application-Specific Systems, Architecture Processors, 2005. ASAP 2005. 16th IEEE International Conference on. IEEE, 2005.
- Rong, C., Zhen-ya, L., Yan-hu, J., Yi, Z., & Li-yu, T. (2011, June). Coding principle and implementation of two-dimensional PDF417 bar code. In Industrial Electronics and Applications (ICIEA), 2011 6th IEEE Conference on (pp. 466-468). IEEE.

- Biao, L. (2007, August). A DataMatrix-based mutant code design and recognition method research. In Image and Graphics, 2007. ICIG 2007. Fourth International Conference on (pp. 570-574). IEEE.
- Gao, J. Z., Prakash, L., & Jagatesan, R. (2007, July). Understanding 2d-barcode technology and applications in m-commerce-design and implementation of a 2d barcode processing solution. In Computer Software and Applications Conference, 2007. COMPSAC 2007. 31st Annual International (Vol. 2, pp. 49-56). IEEE.
- Querini, M., Grillo, A., Lentini, A., & Italiano, G. F. (2011). 2D Color Barcodes for Mobile Phones. IJCSA, 8(1), 136-155.

#### **บทความจาก Websites**

- Lava Iris 405+. (n.d.). Retrieved July 10, 2016, from [http://www.gsmarena.com/lava\\_iris\\_405+-6313.php](http://www.gsmarena.com/lava_iris_405+-6313.php)
- HTC One X. (n.d.). Retrieved July 10, 2016, from [http://www.gsmarena.com/htc\\_one\\_x-4320.php](http://www.gsmarena.com/htc_one_x-4320.php)
- Apple iPhone 4. (n.d.). Retrieved July 10, 2016, from [http://www.gsmarena.com/apple\\_iphone\\_4-3275.php](http://www.gsmarena.com/apple_iphone_4-3275.php)
- List of best-selling mobile phones. (n.d.). Retrieved July 10, 2016, from [https://en.wikipedia.org/wiki/List\\_of\\_best-selling\\_mobile\\_phones](https://en.wikipedia.org/wiki/List_of_best-selling_mobile_phones)



ภาคผนวก

## ภาคผนวก ก

## จำนวนโมดูลของรหัสคิวอาร์โค้ดแต่ละเวอร์ชัน

จำนวนโมดูลของรหัสคิวอาร์โค้ดมาตรฐานตั้งแต่เวอร์ชัน 1 – 40 มีดังนี้

ตาราง 1ก

จำนวนโมดูลของรหัสคิวอาร์โค้ดแต่ละเวอร์ชัน

เวอร์ชัน	จำนวนโมดูล
1	21 × 21
2	25 × 25
3	29 × 29
4	33 × 33
5	37 × 37
6	41 × 41
7	45 × 45
8	49 × 49
9	53 × 53
10	57 × 57
11	61 × 61
12	65 × 65
13	69 × 69
14	73 × 73
15	77 × 77
16	81 × 81
17	85 × 85
18	89 × 89
19	93 × 93
20	97 × 97
21	101 × 101

เวอร์ชัน	จำนวนโมดูล
22	105 × 105
23	109 × 109
24	113 × 113
25	117 × 117
26	121 × 121
27	125 × 125
28	129 × 129
29	133 × 133
30	137 × 137
31	141 × 141
32	145 × 145
33	149 × 149
34	153 × 153
35	157 × 157
36	161 × 161
37	165 × 165
38	169 × 169
39	173 × 173
40	177 × 177

หมายเหตุ. จาก ISO/IEC, Information technology (2006). Automatic identification and data capture techniques QR code 2005 bar code symbology specification, (pp. 19), IS 18004:2006.

## ภาคผนวก ข

### ผลการทดลองข้อจำกัดเกี่ยวกับขนาดของภาพกับเวอร์ชันของคิวอาร์โค้ดมาตรฐานโดยละเอียด

จากการทดลองที่ 4.1 เมื่อนำภาพรหัสคิวอาร์โค้ด เวอร์ชัน 2 5 10 และ 15 ที่ระดับแก้ไขข้อผิดพลาด L ที่ถ่ายจากอุปกรณ์ถ่ายภาพ ในสภาพแสงสีขาว (Daylight) และแสงสีส้ม (Warm White) โดยแบ่งผลการทดลองตามอุปกรณ์การถ่ายภาพ ได้แก่ สมาร์ทโฟน A B และ C (คุณสมบัติของสมาร์ทโฟนแสดงในภาคผนวก จ) มาถอดรหัสด้วยอัลกอริธึมการถอดรหัสของคิวอาร์โค้ดมาตรฐาน โดยหากถอดรหัสภาพได้จะแทนด้วยสัญลักษณ์ (O) และหากไม่สามารถถอดรหัสได้แทนด้วยสัญลักษณ์ (X) ผลการทดลองดังนี้

ตารางที่ 1ข

ผลการถอดรหัสคิวอาร์โค้ดมาตรฐาน เวอร์ชัน 2 ระดับความผิดพลาด L ที่ถ่ายจากสมาร์ทโฟน A

ภาพที่	แสงสีขาว			แสงสีส้ม		
	1 นิ้ว	2 นิ้ว	3 นิ้ว	1 นิ้ว	2 นิ้ว	3 นิ้ว
1	O	O	O	X	O	O
2	X	O	O	X	O	O
3	X	O	O	X	O	O
4	O	O	O	X	O	O
5	X	O	O	X	O	O

ตารางที่ 2ข

ผลการถอดรหัสคิวอาร์โค้ดมาตรฐาน เวอร์ชัน 5 ระดับความผิดพลาด L ที่ถ่ายจากสมาร์ทโฟน A

ภาพที่	แสงสีขาว			แสงสีส้ม		
	1 นิ้ว	2 นิ้ว	3 นิ้ว	1 นิ้ว	2 นิ้ว	3 นิ้ว
1	X	O	O	X	O	O
2	X	O	O	X	O	O
3	X	O	O	X	O	O
4	X	O	O	X	O	O

ภาพที่	แสงสีขาว			แสงสีส้ม		
	1 นิ้ว	2 นิ้ว	3 นิ้ว	1 นิ้ว	2 นิ้ว	3 นิ้ว
5	X	O	O	X	O	O

ตารางที่ 3ข

ผลการถอดรหัสคิวอาร์โค้ดมาตรฐาน เวอร์ชัน 10 ระดับความผิดพลาด L ที่ถ่ายจากสมาร์โฟน A

ภาพที่	แสงสีขาว			แสงสีส้ม		
	1 นิ้ว	2 นิ้ว	3 นิ้ว	1 นิ้ว	2 นิ้ว	3 นิ้ว
1	X	O	O	X	O	O
2	X	O	O	X	O	O
3	X	O	O	X	O	O
4	X	O	O	X	O	O
5	X	O	O	X	O	O

ตารางที่ 4ข

ผลการถอดรหัสคิวอาร์โค้ดมาตรฐาน เวอร์ชัน 15 ระดับความผิดพลาด L ที่ถ่ายจากสมาร์โฟน A

ภาพที่	แสงสีขาว			แสงสีส้ม		
	1 นิ้ว	2 นิ้ว	3 นิ้ว	1 นิ้ว	2 นิ้ว	3 นิ้ว
1	X	X	O	X	X	O
2	X	X	O	X	X	O
3	X	X	O	X	X	O
4	X	X	O	X	X	O
5	X	X	O	X	X	O

## ตารางที่ 5ข

ผลการถอดรหัสคิวอาร์โค้ดมาตรฐาน เวอร์ชัน 2 ระดับความผิดพลาด L ที่ถ่ายจากสมาร์โฟน B

ภาพที่	แสงสีขาว			แสงสีส้ม		
	1 นิ้ว	2 นิ้ว	3 นิ้ว	1 นิ้ว	2 นิ้ว	3 นิ้ว
1	○	○	○	○	○	○
2	○	○	○	○	○	○
3	○	○	X	○	X	○
4	○	X	○	○	○	○
5	X	○	X	○	X	○

## ตารางที่ 6ข

ผลการถอดรหัสคิวอาร์โค้ดมาตรฐาน เวอร์ชัน 5 ระดับความผิดพลาด L ที่ถ่ายจากสมาร์โฟน B

ภาพที่	แสงสีขาว			แสงสีส้ม		
	1 นิ้ว	2 นิ้ว	3 นิ้ว	1 นิ้ว	2 นิ้ว	3 นิ้ว
1	○	○	○	○	○	○
2	○	○	○	○	○	○
3	○	○	○	○	○	○
4	○	○	○	○	○	○
5	○	○	○	○	○	○



## ตารางที่ 7ข

ผลการถอดรหัสคิวอาร์โค้ดมาตรฐาน เวอร์ชัน 10 ระดับความผิดพลาด L ที่ถ่ายจากสมาร์ทโฟน B

ภาพที่	แสงสีขาว			แสงสีส้ม		
	1 นิ้ว	2 นิ้ว	3 นิ้ว	1 นิ้ว	2 นิ้ว	3 นิ้ว
1	○	○	○	○	○	○
2	○	○	○	○	x	○
3	○	○	○	○	○	○
4	○	○	○	○	x	○
5	○	○	○	○	○	○

## ตารางที่ 8ข

ผลการถอดรหัสคิวอาร์โค้ดมาตรฐาน เวอร์ชัน 15 ระดับความผิดพลาด L ที่ถ่ายจากสมาร์ทโฟน B

ภาพที่	แสงสีขาว			แสงสีส้ม		
	1 นิ้ว	2 นิ้ว	3 นิ้ว	1 นิ้ว	2 นิ้ว	3 นิ้ว
1	x	x	x	○	○	x
2	○	x	x	○	x	x
3	○	○	x	x	○	x
4	○	x	x	x	x	x
5	○	x	○	x	○	x

ตารางที่ 9ข

ผลการถอดรหัสคิวอาร์โค้ดมาตรฐาน เวอร์ชัน 2 ระดับความผิดพลาด L ที่ถ่ายจากสมาร์โฟน C

ภาพที่	แสงสีขาว			แสงสีส้ม		
	1 นิ้ว	2 นิ้ว	3 นิ้ว	1 นิ้ว	2 นิ้ว	3 นิ้ว
1	○	○	×	×	×	×
2	○	×	○	○	×	×
3	○	×	○	○	○	○
4	×	×	○	×	×	×
5	○	○	×	○	×	×

ตารางที่ 10ข

ผลการถอดรหัสคิวอาร์โค้ดมาตรฐาน เวอร์ชัน 5 ระดับความผิดพลาด L ที่ถ่ายจากสมาร์โฟน C

ภาพที่	แสงสีขาว			แสงสีส้ม		
	1 นิ้ว	2 นิ้ว	3 นิ้ว	1 นิ้ว	2 นิ้ว	3 นิ้ว
1	○	○	×	×	○	○
2	○	○	×	×	×	×
3	○	×	○	×	×	×
4	×	×	○	×	×	○
5	○	○	○	×	○	×

ตารางที่ 11ข

ผลการถอดรหัสคิวอาร์โค้ดมาตรฐาน เวอร์ชัน 10 ระดับความผิดพลาด L ที่ถ่ายจากสมาร์โฟน C

ภาพที่	แสงสีขาว			แสงสีส้ม		
	1 นิ้ว	2 นิ้ว	3 นิ้ว	1 นิ้ว	2 นิ้ว	3 นิ้ว
1	○	○	○	X	○	X
2	○	○	○	X	X	○
3	○	○	○	○	○	○
4	○	X	○	X	X	○
5	X	○	○	○	○	○

ตารางที่ 12ข

ผลการถอดรหัสคิวอาร์โค้ดมาตรฐาน เวอร์ชัน 15 ระดับความผิดพลาด L ที่ถ่ายจากสมาร์โฟน C

ภาพที่	แสงสีขาว			แสงสีส้ม		
	1 นิ้ว	2 นิ้ว	3 นิ้ว	1 นิ้ว	2 นิ้ว	3 นิ้ว
1	○	X	X	X	X	X
2	X	X	X	X	X	X
3	X	X	X	○	X	X
4	X	○	X	X	X	X
5	X	○	X	X	X	X

### ภาคผนวก ค

#### ผลการทดลองจำนวนระดับเทาที่เหมาะสมในการใช้งานคิวอาร์โค้ดระดับเทาโดยละเอียด

จากการทดลองที่ 4.2 นำภาพถ่ายที่ประกอบด้วยช่องที่มีระดับเทา 4 8 และ 16 ระดับ ดังแสดงในภาพ 3.5 โดยภาพถ่ายได้สมาร์ทโฟนรุ่น A B และ C มาวิเคราะห์หาค่า RGB น้อยที่สุดและมากที่สุดของแต่ละอุปกรณ์ มีผลการทดลองดังนี้

ตารางที่ 1ค

ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 4 ระดับ จากภาพที่ถ่ายโดยสมาร์ทโฟน A

ระดับเทาที่	สภาพแสงสีขาว		สภาพแสงสีส้ม	
	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)
1	171	180	163	172
2	131	138	114	128
3	50	58	48	60
4	30	40	28	33

ตารางที่ 2ค

ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 4 ระดับ จากภาพที่ถ่ายโดยสมาร์ทโฟน B

ระดับเทาที่	สภาพแสงสีขาว		สภาพแสงสีส้ม	
	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)
1	138	156	131	155
2	96	120	93	114
3	40	47	34	35
4	19	33	20	29

## ตารางที่ 3ค

ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 4 ระดับ จากภาพที่ถ่ายโดยสมาร์ทโฟน C

ระดับเทาที่	สภาพแสงสีขาว		สภาพแสงสีส้ม	
	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)
1	189	210	200	211
2	117	141	120	131
3	25	40	29	42
4	5	17	11	17

จากตาราง 1ค – 3ค ทำให้พบว่าที่ภาพระดับเทา 4 ระดับที่ถ่ายด้วยสมาร์ทโฟนทั้ง 3 รุ่น ช่วงค่า RGB ของทั้ง 4 ระดับเทาไม่มีการซ้อนทับกับระดับเทาอื่นทั้งสภาพแสงสีขาวและแสงสีส้ม แสดงให้เห็นว่าที่ระดับเทา 4 ระดับสามารถถอดรหัสคิวอาร์โค้ดระดับเทาได้แล้วผลลัพธ์จะไม่ผิดเพี้ยนไปและเหมาะสมที่จะนำมาใช้ในคิวอาร์โค้ดระดับเทา

## ตารางที่ 4ค

ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 8 ระดับ จากภาพที่ถ่ายโดยสมาร์ทโฟน A

ระดับเทาที่	สภาพแสงสีขาว		สภาพแสงสีส้ม	
	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)
1	211	216	200	211
2	192	198	175	191
3	155	168	145	155
4	118	127	111	130
5	113	122	79	92
6	69	79	58	69

ระดับเทาที่	สภาพแสงสีขาว		สภาพแสงสีส้ม	
	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)
7	45	53	38	46
8	34	46	30	39

จากตาราง 4ค ทำให้พบว่าที่ภาพระดับเทา 8 ระดับที่ถ่ายด้วยสมาร์ทโฟน A ช่วงค่า RGB ของระดับเทาที่ 8 มีการซ้อนทับกับระดับเทาที่ 7 ที่สภาพแสงสีส้ม แสดงให้เห็นว่าที่ระดับเทา 8 ระดับ เมื่อทำการถอดรหัสคิวอาร์โค้ดระดับเทาแล้วมีโอกาสที่ผลลัพธ์จะผิดพลาด จึงไม่เหมาะที่จะนำมาใช้ในคิวอาร์โค้ดระดับเทา

ตารางที่ 5ค

ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 8 ระดับ จากภาพที่ถ่ายโดยสมาร์ทโฟน B

ระดับเทาที่	สภาพแสงสีขาว		สภาพแสงสีส้ม	
	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)
1	158	178	149	176
2	143	168	140	160
3	119	138	117	136
4	89	105	84	104
5	81	107	76	99
6	49	68	48	66
7	30	45	26	40
8	14	34	17	27

จากตาราง 5ค ทำให้พบว่าที่ภาพระดับเทา 8 ระดับที่ถ่ายด้วยสมาร์ทโฟน B ที่สภาพแสงสีขาว ช่วงค่า RGB ของระดับเทาที่ 2 มีการซ้อนทับกับระดับเทาที่ 1 และระดับเทาที่ 5 มีการซ้อนทับ

กับระดับเทาที่ 4 และที่สภาพแสงสีส้ม ช่วงค่า RGB ของระดับเทาที่ 2 มีการซ้อนทับกับระดับเทาที่ 1 และระดับเทาที่ 5 มีการซ้อนทับกับระดับเทาที่ 4 และระดับเทาที่ 8 มีการซ้อนทับกับระดับเทาที่ 7 แสดงให้เห็นว่าที่ระดับเทา 8 ระดับ เมื่อทำการถอดรหัสคิวอาร์โค้ดระดับเทาแล้วมีโอกาสที่ผลลัพธ์จะผิดพลาด จึงไม่เหมาะที่จะนำมาใช้ในคิวอาร์โค้ดระดับเทา

ตารางที่ 6ค

ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 8 ระดับ จากภาพที่ถ่ายโดยสมาร์ทโฟน C

ระดับเทาที่	สภาพแสงสีขาว		สภาพแสงสีส้ม	
	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)
1	217	238	222	235
2	191	211	203	215
3	159	172	158	173
4	97	116	107	127
5	76	95	90	101
6	47	58	55	62
7	22	36	24	37
8	4	13	11	14

จากตาราง 6ค ทำให้พบว่าที่ภาพระดับเทา 8 ระดับที่ถ่ายด้วยสมาร์ทโฟน C ที่สภาพแสงสีขาวและสีส้ม ช่วงค่า RGB ของระดับเทาทั้ง 8 ระดับไม่มีการซ้อนทับกัน แต่ระยะห่างของระดับเทาที่ 1 กับ 2 มีความใกล้เคียงกันมาก มีโอกาสที่ผลลัพธ์จะผิดพลาดได้ง่าย จึงไม่เหมาะที่จะนำมาใช้ในคิวอาร์โค้ดระดับเทา

## ตารางที่ 7ค

ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 16 ระดับ จากภาพที่ถ่ายโดยสมาร์ท A

ระดับเทาที่	สภาพแสงสีขาว		สภาพแสงสีส้ม	
	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)
1	155	162	147	162
2	159	166	155	169
3	161	172	159	168
4	158	170	156	168
5	152	160	142	157
6	137	142	124	135
7	109	115	92	107
8	86	98	68	91
9	53	65	49	58
10	53	62	43	56
11	47	60	45	55
12	45	55	42	51
13	43	52	41	47
14	38	49	32	42
15	32	40	24	35
16	27	39	21	34



## ตารางที่ 8ค

ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 16 ระดับ จากภาพที่ถ่ายโดยสมาร์ทโฟน B

ระดับเทาที่	สภาพแสงสีขาว		สภาพแสงสีส้ม	
	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)
1	131	149	124	146
2	130	146	124	143
3	122	140	120	139
4	117	138	110	135
5	111	128	101	128
6	93	116	89	115
7	77	97	76	100
8	59	80	59	78
9	48	58	41	53
10	40	53	31	50
11	36	44	33	43
12	30	40	29	43
13	30	40	24	37
14	23	37	21	32
15	12	26	16	22
16	10	27	15	21

## ตารางที่ 9ค

ผลการหาค่าน้อยที่สุดและมากที่สุดของระดับเทาแบบ 16 ระดับ จากภาพที่ถ่ายโดยสมาร์ทโฟน C

ระดับเทาที่	สภาพแสงสีขาว		สภาพแสงสีส้ม	
	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)	ค่าน้อยสุด (min)	ค่ามากที่สุด (max)
1	178	202	180	189
2	165	189	170	183
3	162	180	160	174
4	145	164	154	166
5	134	153	139	153
6	119	131	115	132
7	85	111	90	111
8	62	75	55	76
9	38	60	45	55
10	31	44	40	46
11	25	38	29	40
12	19	31	24	31
13	17	27	19	28
14	12	21	13	19
15	6	15	8	15
16	2	10	0	9

จากตาราง 7ค – 9ค ทำให้พบว่าที่ภาพระดับเทา 16 ระดับที่ถ่ายด้วยสมาร์ทโฟนทั้ง 3 รุ่น มีช่วงค่า RGB ระดับเทามีการซ้อนทับกับระดับเทาอื่นอยู่หลายระดับ ทั้งสภาพแสงสีขาวและแสงสีส้ม

แสดงให้เห็นว่าที่ระดับเทา 16 ระดับ เมื่อทำการถอดรหัสคิวอาร์โค้ดระดับเทาแล้วมีโอกาสที่ผลลัพธ์จะผิดพลาด จึงไม่เหมาะที่จะนำมาใช้ในคิวอาร์โค้ดระดับเทา



## ภาคผนวก ง

## ผลการทดลองความสามารถจู้ข้อมูลของคิวอาร์โค้ดระดับเทาเปรียบเทียบกับคิวอาร์โค้ดมาตรฐาน

จากผลการทดลองจำนวนระดับเทาที่เหมาะสมในการใช้งานคิวอาร์โค้ดระดับเทา ทำให้ทราบแล้วว่าระดับเทาที่เหมาะสมกับคิวอาร์โค้ดระดับเทานั้น คือ 4 ระดับเทา ดังนั้นผลการทดลอง 4.3 เพื่อหาความสามารถในการจู้ข้อมูลของคิวอาร์โค้ดระดับเทาโดยใช้ระดับเทา 4 ระดับ มีผลการทดลองแสดงในตาราง ดังนี้

ตารางที่ 1ง

เปรียบเทียบจำนวนตัวอักษรที่คิวอาร์โค้ดมาตรฐานกับคิวอาร์โค้ดระดับเทา 4 ระดับสามารถจู้ได้ ที่ระดับแก้ไขข้อผิดพลาด  $L$

เวอร์ชัน	จำนวนตัวอักษรที่จู้ได้ (ตัว)		เปอร์เซ็นต์ที่เพิ่มขึ้น
	คิวอาร์โค้ดมาตรฐาน	คิวอาร์โค้ดระดับเทา	
1	25	47	188.00%
2	47	77	163.83% *
3	77	154	200.00%
4	114	224	196.49%
5	154	279	181.17%
6	195	335	171.79%
7	224	395	176.34%
8	279	535	191.76%
9	335	667	199.10%
10	395	758	191.90%
11	468	854	182.48%
12	535	1046	195.51%
13	619	1153	186.27%
14	667	1352	202.70% **
15	758	1460	192.61%
16	854	1588	185.95%

เวอร์ชัน	จำนวนตัวอักษรที่จุได้ (ตัว)		เปอร์เซ็นต์ที่เพิ่มขึ้น
	คิวอาร์โค้ดมาตรฐาน	คิวอาร์โค้ดระดับเทา	
17	938	1853	197.55%
18	1046	1990	190.25%
19	1153	2223	192.80%
20	1249	2369	189.67%
21	1352	2520	186.39%
22	1460	2840	194.52%
23	1588	3009	189.48%
24	1704	3351	196.65%
25	1853	3537	190.88%
26	1990	3927	197.34%
27	2132	4087	191.70%
28	2223	4296	193.25%
		เฉลี่ย	189.87%

หมายเหตุ \* ค่าน้อยสุด (min) , \*\* ค่ามากที่สุด (max)

จากตาราง 1ง พบว่าคิวอาร์โค้ดระดับเทาสามารถจุข้อมูลได้มากกว่าคิวอาร์โค้ดมาตรฐาน ประมาณ 2 เท่าในทุกเวอร์ชันที่ระดับแก้ไขข้อผิดพลาด L

ตารางที่ 2ง

เปรียบเทียบจำนวนตัวอักษรที่คิวอาร์โค้ดมาตรฐานกับคิวอาร์โค้ดระดับเทา 4 ระดับสามารถจุได้ ที่ระดับแก้ไขข้อผิดพลาด M

เวอร์ชัน	จำนวนตัวอักษรที่จุได้ (ตัว)		เปอร์เซ็นต์ที่เพิ่มขึ้น
	คิวอาร์โค้ดมาตรฐาน	คิวอาร์โค้ดระดับเทา	
1	20	38	190.00%
2	38	61	160.53% *
3	61	122	200.00%
4	90	178	197.78%

เวอร์ชัน	จำนวนตัวอักษรที่จับได้ (ตัว)		เปอร์เซ็นต์ที่เพิ่มขึ้น
	คิวอาร์โค้ดมาตรฐาน	คิวอาร์โค้ดระดับเทา	
5	122	221	181.15%
6	154	262	170.13%
7	178	311	174.72%
8	221	419	189.59%
9	262	528	201.53% **
10	311	600	192.93%
11	366	656	179.23%
12	419	816	194.75%
13	483	909	188.20%
14	528	1035	196.02%
15	600	1134	189.00%
16	656	1248	190.24%
17	734	1451	197.68%
18	816	1542	188.97%
19	909	1732	190.54%
20	970	1839	189.59%
21	1035	1994	192.66%
22	1134	2238	197.35%
23	1248	2369	189.82%
24	1326	2632	198.49%
25	1451	2780	191.59%
26	1542	3054	198.05%
27	1637	3220	196.70%
28	1732	3391	195.79%
		เฉลี่ย	190.11%

หมายเหตุ \* ค่าน้อยสุด (min) , \*\* ค่ามากที่สุด (max)

จากตาราง 2ง พบว่าคิวอาร์โค้ดระดับเทาสามารถจุข้อมูลได้มากกว่าคิวอาร์โค้ดมาตรฐาน  
ประมาณ 2 เท่าในทุกเวอร์ชันที่ระดับแก้ไขข้อผิดพลาด M

ตารางที่ 3ง

เปรียบเทียบจำนวนตัวอักษรที่คิวอาร์โค้ดมาตรฐานกับคิวอาร์โค้ดระดับเทา 4 ระดับสามารถจุได้ ที่  
ระดับแก้ไขข้อผิดพลาด Q

เวอร์ชัน	จำนวนตัวอักษรที่จุได้ (ตัว)		เปอร์เซ็นต์ที่เพิ่มขึ้น
	คิวอาร์โค้ดมาตรฐาน	คิวอาร์โค้ดระดับเทา	
1	16	29	181.25%
2	29	47	162.07% *
3	47	87	185.11%
4	67	125	186.57%
5	87	157	180.46%
6	108	189	175.00%
7	125	221	176.80%
8	157	296	188.54%
9	189	376	198.94%
10	221	426	192.76%
11	259	470	181.47%
12	296	574	193.92%
13	352	644	182.95%
14	376	742	197.34%
15	426	823	193.19%
16	470	890	189.36%
17	531	1041	196.05%
18	574	1094	190.59%
19	644	1263	196.12%
20	702	1322	188.32%
21	742	1429	192.59%

เวอร์ชัน	จำนวนตัวอักษรที่จู่ได้ (ตัว)		เปอร์เซ็นต์ที่เพิ่มขึ้น
	คิวนาร์โค้ดมาตรฐาน	คิวนาร์โค้ดระดับเทา	
22	823	1618	196.60%
23	890	1700	191.01%
24	963	1867	193.87%
25	1041	1966	188.86%
26	1094	2181	199.36% **
27	1172	2298	196.08%
28	1263	2420	191.61%
		เฉลี่ย	188.81%

หมายเหตุ \* คำน้อยสุด (min) , \*\* ค่ามากที่สุด (max)

จากตาราง 3ง พบว่าคิวนาร์โค้ดระดับเทาสามารถจู่ข้อมูลได้มากกว่าคิวนาร์โค้ดมาตรฐาน ประมาณ 2 เท่าในทุกเวอร์ชันที่ระดับแก้ไขข้อผิดพลาด Q

ตารางที่ 4ง

เปรียบเทียบจำนวนตัวอักษรที่คิวนาร์โค้ดมาตรฐานกับคิวนาร์โค้ดระดับเทา 4 ระดับสามารถจู่ได้ ที่ระดับแก้ไขข้อผิดพลาด H

เวอร์ชัน	จำนวนตัวอักษรที่จู่ได้ (ตัว)		เปอร์เซ็นต์ที่เพิ่มขึ้น
	คิวนาร์โค้ดมาตรฐาน	คิวนาร์โค้ดระดับเทา	
1	10	20	200.00%
2	20	35	175.00%
3	35	64	182.86%
4	50	93	186.00%
5	64	122	190.63%
6	84	143	170.24% *
7	93	174	187.10%
8	122	227	186.07%
9	143	283	197.90%



เวอร์ชัน	จำนวนตัวอักษรที่จับได้ (ตัว)		เปอร์เซ็นต์ที่เพิ่มขึ้น
	คิวอาร์โค้ดมาตรฐาน	คิวอาร์โค้ดระดับเทา	
10	174	321	184.48%
11	200	365	182.50%
12	227	452	199.12%
13	259	493	190.35%
14	283	587	207.42% **
15	321	640	199.38%
16	365	672	184.11%
17	408	779	190.93%
18	452	864	191.15%
19	493	958	194.32%
20	557	1016	182.41%
21	587	1080	183.99%
22	640	1226	191.56%
23	672	1307	194.49%
24	744	1431	192.34%
25	779	1530	196.41%
26	864	1658	191.90%
27	910	1774	194.95%
28	958	1852	193.32%
		เฉลี่ย	190.03%

หมายเหตุ \* ค่าน้อยสุด (min) , \*\* ค่ามากที่สุด (max)

จากตาราง 4ง พบว่าคิวอาร์โค้ดระดับเทาสามารถจับข้อมูลได้มากกว่าคิวอาร์โค้ดมาตรฐานประมาณ 2 เท่าในทุกเวอร์ชันที่ระดับแก้ไขข้อผิดพลาด Q

ดังนั้นจากการทดลองเปรียบเทียบความสามารถสรุปได้ว่า คิวอาร์โค้ดระดับเทา 4 ระดับสามารถจับข้อมูลได้มากกว่าคิวอาร์โค้ดมาตรฐานในเวอร์ชันและระดับแก้ไขข้อผิดพลาดเดียวกันประมาณ 2 เท่า โดยเฉลี่ยทุกระดับแก้ไขข้อผิดพลาด (189.70%)

## ภาคผนวก จ

### คุณสมบัติของสมาร์ทโฟนที่ใช้ในการทดลอง

ในงานวิจัยฉบับนี้ใช้อุปกรณ์สมาร์ทโฟนในการทดลอง 3 รุ่น ดังนี้

1. iPhone 4 เป็นตัวแทนของสมาร์ทโฟนที่เป็นที่นิยม เนื่องจากเป็นสมาร์ทโฟนที่มียอดขายประมาณ 50 ล้านเครื่อง เป็นอันดับ 8 ของโลก (เฉพาะสมาร์ทโฟนที่วางจำหน่ายระหว่างปี 2010 - 2016) ปัจจุบัน iPhone 4 ไม่ได้รับการสนับสนุนระบบปฏิบัติการใหม่ ระบบปฏิบัติการล่าสุดที่รองรับคือ iOS 7 (Wikipedia, 2016)
2. HTC One X เป็นตัวแทนของสมาร์ทโฟนที่ใช้ระบบปฏิบัติการแอนดรอยด์รุ่นเรือธง (Flagship) ที่วางจำหน่ายในช่วงเวลาใกล้เคียงกับ iPhone 4 (GSMArena, 2016)
3. AIS Lava Iris 405+ เป็นตัวแทนของสมาร์ทโฟนที่ไม่มีการจับโฟกัสอัตโนมัติ และเป็นตัวแทนของสมาร์ทโฟนระดับเริ่มต้น (Low Entry) (GSMArena, 2016)

ในส่วน ผลการทดลอง iPhone 4 จะใช้คำว่า สมาร์ทโฟน C, HTC One X จะใช้คำว่าสมาร์ทโฟน B และ AIS Lava Iris 405+ จะใช้คำว่าสมาร์ทโฟน A เพื่อป้องกันการแปลผลการทดลองแบบเปรียบเทียบระหว่างสมาร์ทโฟน โดยคุณสมบัติจำเพาะของสมาร์ทโฟนทั้ง 3 รุ่น ดังต่อไปนี้

ตาราง 1จ

#### คุณสมบัติของสมาร์ทโฟนที่ใช้ในการทดลอง

ยี่ห้อ	ระบบปฏิบัติการ	หน่วยประมวลผลกลาง	ความละเอียดกล้อง (ล้านพิกเซล)	ออดิโอโฟกัส
iPhone4	iOS 7	1.0 GHz Cortex-A8	5	มี
HTC One X	Android 4.2 (Ice Cream Sandwich)	Quad-core 1.5 GHz	8	มี
AIS Lava 405+	Android 4.2 (Jelly Bean)	Dual-core 1.3 GHz	5	ไม่มี

หมายเหตุ. จาก GSMArena.com - GSM phone reviews, news, opinions, votes, manuals and more... (n.d.). Retrieved July 10, 2016, from <http://www.gsmarena.com/>