

การปรับปรุงวิธีการเรียงลำดับข้อมูลในภาพคิวอาร์โค้ด โดยใช้เทคนิค EVM สำหรับของการฝังข้อมูลแบบกู้คืนกลับได้ Improved Sorting in QR Code images Using EVM technique for Reversible Data Hiding

ชัยพร ปานยืนดี ชีรนัย ลีลาวิวัฒน์ และ ภัทรวดี รังสิริภักษ์

Chaiyaporn Panyindee, Teranai Leelawiwat and Pattarawadee Rangsirak

สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลรัตนโกสินทร์

Department of Computer Engineering, Faculty of Engineering,

Rajamangala University of Technology Rattanakosin (RMUTR), Nakhon Pathom, 73170, Thailand

E-mail: chaiyaporn.pan@gmail.com, Chaiyaporn.pan@rmutr.ac.th

Telephone Number: +662-889-4585-7, fax: +662-889-4585-7

บทคัดย่อ

งานวิจัยฉบับนี้นำเสนอหนึ่งการปรับปรุงวิธีการเรียงลำดับข้อมูลในภาพคิวอาร์โค้ด โดยประยุกต์ใช้เทคนิค EVM เพื่อเรียงลำดับค่าความคลาดเคลื่อนจากการทำนายที่สอดคล้องก่อนการฝัง ลำดับของค่าความคลาดเคลื่อนจากการทำนายถูกคำนวณใหม่ และใช้แทนที่ค่าความแปรปรวนดั้งเดิม ซึ่งคำนวณได้จากสี่พิกเซลข้างเคียง เพื่อให้ได้รับค่า PSNR สูงสุดที่เป็นไปได้ ค่าความแปรปรวนใหม่สามารถขยายระดับของพื้นที่พิกเซลได้สูงสุด 30 ระดับ สำหรับ 40 ภาพคิวอาร์โค้ดที่แตกต่างกัน ผลลัพธ์แสดงให้เห็นการเรียงลำดับที่มีประสิทธิภาพกว่างานในอดีตก่อนหน้านี้ โดยเฉพาะการฝังข้อมูลขนาดเล็ก

คำสำคัญ: ความคลาดเคลื่อนจากการทำนาย การฝังข้อมูลแบบกู้คืนกลับได้ คิวอาร์โค้ด EVM

ABSTRACT

This paper presents an improvement of the data sorting method for reversible data hiding in QR code images by applying an EVM technique to sort the prediction error values before embedding. The order of the predicted error values was recalculated and is replaced the original local variance, which was calculated from the four neighboring pixels to get the maximum possible PSNR value. The new variance can extend the level of the pixel area to a maximum of 30 levels with 40 versions of the QR code images that is used for testing. The results showed a more efficient sorting over previous methods, especially small data embedding.

Keywords: prediction error; reversible data hiding; QR code; EVM

1. บทนำ

การฝังข้อมูลแบบกู้คืนกลับได้ (Reversible Data Hiding: RDH) ถูกนำมาประยุกต์ใช้ในหลายงาน เช่น การฝังข้อมูลที่ไม่ต้องการเปิดเผยในภาพทางการแพทย์ การเข้ารหัสลับสำหรับภาพทางการแพทย์ เป็นต้น ในปัจจุบันภาพคิวอาร์โค้ดได้ถูกนำมาใช้อย่างแพร่หลายใน

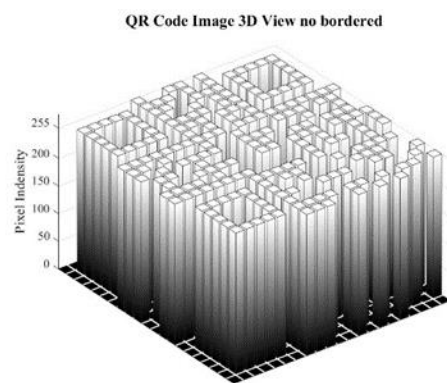
หลายธุรกิจ เนื่องจากประสิทธิภาพที่สูง ต้นทุนต่ำ สามารถเข้าถึงได้ง่าย และยังคงใส่ข้อมูลได้เป็นจำนวนมาก จึงเป็นที่นิยม อย่างไรก็ตาม ข้อมูลบางประเภทยังคงต้องการความเป็นส่วนตัว เช่น เลขที่ธุรกรรม วันเดือนปีเกิด เลขประจำตัวประชาชน จึงต้องอาศัยวิธีการที่สามารถป้องกัน

การเข้าถึงข้อมูลในส่วนที่ไม่ต้องการเปิดเผย แต่ยังคงให้ภาพควอาร์โค้ดที่สามารถเข้าถึงข้อมูลปกติได้

RDH เป็นหนึ่งแขนงงานของการฝังลายน้ำ (Watermarking) ที่ถูกพัฒนามาอย่างต่อเนื่อง มีนักวิจัยจำนวนมากนำเสนอวิธีการใหม่ ๆ สำหรับเพิ่มประสิทธิภาพของการฝังข้อมูลโดยที่ยังคงสามารถกู้คืนข้อมูลต้นฉบับกลับได้ทั้งหมดแบบไม่สูญเสีย หนึ่งในวิธีการที่ถูกกล่าวขานอย่างยาวนานคือ Difference Expansion (DE) [1] ซึ่งถูกนำเสนอโดย Tian เทคนิคของพวกเขากลับปรับปรุงในหลายแง่มุม เช่น Thodi และ Rodriguez นำเสนอกระบวนการที่มีชื่อว่า Prediction Error Expansion (PEE) [2] โดยผู้เขียนใช้ค่าการทำนายแทนที่ค่าความแตกต่าง กระบวนการของผู้เขียนช่วยลดความบิดเบือนอย่างมีนัยสำคัญ สำหรับตัวทำนายรูปแบบใหม่ จึงถูกต่อขยายในอีกหลายงานวิจัย Thodi และ Rodriguez ได้นำเสนออีกหนึ่งการปรับปรุงที่สำคัญ ซึ่งมีผลต่องานยุคสมัยใหม่ คือ Histogram Shift (HS) [3] HS ถูกนำมาใช้แก้ปัญหาการซ้อนทับกันของพิกเซลหลังการฝังโดยใช้เทคนิค PEE เดิม ผู้วิจัยเรียกเทคนิคนี้ว่า PEHS เทคนิคดังกล่าวช่วยให้ขนาดของพื้นที่แมป (location map) เล็กลง หนึ่งในปัญหาใหญ่ของ RDH คือขนาดของพื้นที่แมปมักมีขนาดใหญ่ เนื่องจากพื้นที่แมปต้องถูกฝังไปพร้อมกับข้อมูลเพื่อใช้ในการกู้คืนกลับ ดังนั้น หากลดขนาดของพื้นที่แมปลงได้ พื้นที่ในการฝังจะสูงขึ้น มีหลายเทคนิค [4]-[5] สามารถลดขนาดของพื้นที่แมปได้ อย่างมีนัยสำคัญ หนึ่งในกลยุทธ์ที่มีประสิทธิภาพคือการใช้เทคนิคการเรียงลำดับข้อมูล (Data Sorting: DS) [6] DS สำหรับงาน RDH ถูกนำเสนอครั้งแรกโดย Kamstra และ Heijmans ผู้วิจัยเรียงลำดับข้อมูลสำหรับการฝัง โดยใช้ค่าสหสัมพันธ์จากพิกเซลข้างเคียงสี่ตำแหน่ง แนวคิดของการเรียงลำดับข้อมูลก่อนการฝังถูกพัฒนาอย่างต่อเนื่อง [7]-[9]

ในปี ค.ศ.2017 Pandyandee และคณะ [10] นำเสนอการใช้หลายเทคนิคสำหรับหนึ่งขั้นตอนวิธี RDH ร่วมกับการปรับภาพแบบเชิงเส้นสำหรับภาพควอาร์โค้ด

ในหลายเวอร์ชัน วิธีการของพวกเขาบรรลุการฝังขนาดใหญ่ในขณะที่ความบิดเบือนต่ำเมื่อเทียบกับงานก่อนหน้า [11] หนึ่งในปัญหาสำคัญสำหรับการจัดเรียงลำดับข้อมูลโดยใช้เพียงสี่พิกเซลข้างเคียงในภาพควอาร์โค้ด สังเกตจากรูปที่ 1 พบว่า ภาพควอาร์โค้ดมีลักษณะของพิกเซลคล้ายคลึงกับสัญญาณรบกวน กล่าวคือ ค่าเฉลี่ยของพิกเซลมีการเปลี่ยนแปลงอย่างรวดเร็ว เช่น จาก 255 เป็น 0 ซึ่งการใช้ค่าความแปรปรวนจาก 4 ตำแหน่งข้างเคียงไม่เพียงพอเพื่อให้ได้การเรียงลำดับข้อมูลที่ต้องการ



รูปที่ 1 ภาพ QR Code ในมุมมองพิกเซลสามมิติ

งานวิจัยนี้แนะนำเสนอหนึ่งการปรับปรุงสำหรับเทคนิคการเรียงลำดับข้อมูลโดยประยุกต์ใช้ Expand Variance Means (EVM) ร่วมกับหลายเทคนิค อาทิ Histogram Shift, การทำนายโดยใช้ Rhombus, การฝังแบบดับเบิล (Double Embedding Scheme) และการปรับภาพโดยใช้การถ่วงน้ำหนักแบบเชิงเส้น (WL) การคำนวณค่า EVM ใหม่ ในแต่ละการฝังถูกเพิ่มขอบเขตจนได้รับค่าที่เหมาะสมที่สุดเพื่อให้ได้ค่า PSNR สูงสุดที่เป็นไปได้ การเรียงลำดับข้อมูลที่มีประสิทธิภาพ ค่าความคลาดเคลื่อนจากการทำนายที่มีค่าควรถูกเรียงขึ้นก่อนค่าที่มีค่าสูงตามค่า EVM เมื่อฝังข้อมูลโดยใช้ HS ความบิดเบือนที่ได้รับจะต่ำตามค่าความคลาดเคลื่อนจากการทำนายที่มีขนาดเล็ก ในทางเดียวกัน เมื่อค่าความคลาดเคลื่อนที่มีขนาดเล็กยังคงช่วยลดปัญหา Overflow หรือในอีกความหมายหนึ่ง

ขนาดของพื้นที่ที่สามารถฝังได้สูงขึ้น ในขณะที่พื้นที่แม่ปมีขนาดเล็กลง

รายละเอียดส่วนอื่น ๆ ของงานวิจัยมีดังต่อไปนี้ ส่วนที่ 2 อธิบายงานวิจัยที่เกี่ยวข้อง ส่วนที่ 3 อธิบายขั้นตอนวิธีที่นำเสนอ ส่วนที่ 4 อธิบายผลการทดลอง และ ส่วนที่ 5 สรุปผลการทดลอง

2. งานวิจัยที่เกี่ยวข้องกับการเรียงลำดับข้อมูลของ RDH

Kamstra และ Heijmans [6] นำเสนอหนึ่งเทคนิคสำหรับการเรียงลำดับข้อมูล วิธีการของพวกเขารเรียงลำดับของคู่พิกเซลตามค่าสหสัมพันธ์ของพิกเซลข้างเคียงงานของพวกเขารรูปประสิทธิภาพที่ดีเยี่ยมในการลดขนาดของพื้นที่แม่ป อย่างไรก็ตาม การเรียงลำดับใหม่ยังคงอยู่บนพื้นฐานของกระบวนการ DE ดั้งเดิม ซึ่งสามารถฝังข้อมูลสูงสุดได้เพียง 0-5 bpp Sachnev และคณะ [7] ปรับปรุงกระบวนการโดยนำเสนอการฝังแบบดับเบิล กล่าวคือ ฝังในโครสเซต และคอกเซตซึ่งอิสระต่อกัน และใช้ประโยชน์จากสี่ตำแหน่งข้างเคียงในการคำนวณค่าความแปรปรวน (Local Variance: LV) ดังนี้

$$\mu_{i,j} = \frac{1}{4} \sum_{k=1}^4 (\Delta v_k - \Delta \bar{v}_k)^2 \quad (1)$$

โดย $\Delta v_1 = |v_{i,j-1} - v_{i-1,j}|$, $\Delta v_2 = |v_{i-1,j} - v_{i,j+1}|$,

$\Delta v_3 = |v_{i,j+1} - v_{i+1,j}|$, $\Delta v_4 = |v_{i+1,j} - v_{i,j-1}|$,

$\Delta \bar{v}_k = (\Delta v_1 + \Delta v_2 + \Delta v_3 + \Delta v_4) / 4$

Δv_1 คือค่าสมบูรณ์ของผลต่างของสองพิกเซลข้างเคียง $v_{i,j-1}$ และ $v_{i-1,j}$, Δv_2 คือค่าสมบูรณ์ของผลต่างของสองพิกเซลข้างเคียง $v_{i-1,j}$ และ $v_{i,j+1}$, Δv_3 คือค่าสมบูรณ์ของผลต่างของสองพิกเซลข้างเคียง $v_{i,j+1}$ และ $v_{i+1,j}$, Δv_4 คือค่าสมบูรณ์ของผลต่างของสองพิกเซลข้างเคียง $v_{i+1,j}$ และ $v_{i,j-1}$ $\Delta \bar{v}_k$ คือค่าเฉลี่ยของ Δv_1 , Δv_2 , Δv_3 และ Δv_4 การเรียงลำดับของพวกเขามีประสิทธิภาพสูง

อันเนื่องมาจากความสัมพันธ์ของตำแหน่งที่ใช้สำหรับการคำนวณค่าความคลาดเคลื่อนจากการทำนาย และ LV ใช้ตำแหน่งเดียวกัน ซึ่งไม่กระทบต่อการฝังในโครสเซต และคอกเซต อย่างไรก็ตาม หนึ่งปัญหาสำหรับภาพที่มีการเปลี่ยนแปลงแบบฉับพลัน เช่น ภาพ Baboon Barbara เป็นต้น การใช้สี่ตำแหน่งข้างเคียงสำหรับคำนวณค่า LV ไม่เพียงพอ เพื่อให้ได้ลำดับการเรียงที่ถูกต้อง Kotvicha และคณะ [9] นำเสนอเทคนิค EVM เพื่อแก้ปัญหานี้ โดยขยายพิกเซลสำหรับคำนวณค่าเฉลี่ยของค่า LV เดิม ดังสมการที่ (2) และ (3)

$$\bar{\mu}_{i,j}^{ex} = \frac{\sum_{l=-\frac{ex}{2}}^{\frac{ex}{2}} \mu_{i+l,j+l} + \sum_{k=1}^{\frac{ex}{2}} \sum_{l=-\frac{ex}{2}}^{\frac{ex}{2}-2k} (\mu_{i+l,j+l+2k} + \mu_{i+l+2k,j+l})}{\frac{ex^2}{2} + ex + 1} \quad (2)$$

$$\bar{\mu}_{i,j}^{ex} = \frac{\sum_{l=-\frac{ex-1}{2}}^{\frac{ex-1}{2}} \mu_{i+l,j+l} + \sum_{k=1}^{\frac{ex-1}{2}} \sum_{l=-\frac{ex-1}{2}}^{\frac{ex-1}{2}-2k} (\mu_{i+l,j+l+2k} + \mu_{i+l+2k,j+l})}{\frac{ex^2}{2} + 2ex - \frac{3}{2}} \quad (3)$$

งานของพวกเขาก็ให้ผลลัพธ์ที่น่าพึงพอใจสำหรับขนาดพื้นที่แม่ปที่ลดลง อย่างไรก็ตาม การขยายพื้นที่พิกเซลสำหรับคำนวณค่าการเรียงลำดับ EVM (ดูรูปที่ 2 (ค) ด้านซ้ายมือ) ใหม่ทุกครั้ง ส่งผลโดยตรงต่อเวลาในการประมวลผลที่สูงขึ้นตามการขยายในแต่ละครั้ง ดังกล่าวพื้นที่ที่เหมาะสมสำหรับการขยายออกมกอยู่ใกล้ ๆ ตำแหน่งศูนย์กลางการทำนาย ดังนั้น การขยายพื้นที่ออกไปไกลมาก ๆ เป็นสิ่งที่ไม่จำเป็น

3. วิธีการที่นำเสนอ

การเรียงลำดับพิกเซลยังคงเป็นสิ่งจำเป็นสำหรับ RDH พิกเซลที่ไม่ถูกเรียงมีโอกาสน้อยให้เกิดปัญหา Overflow และ Underflow สูงกว่าพิกเซลที่มีการเรียงลำดับ โดยเฉพาะอย่างยิ่งภาพที่มีความแปรปรวนสูง เช่น ภาพควอร์โคว์ (ดูรูปที่ 2 (ก) ด้านซ้ายมือ) ข้อเท็จจริง

บางประการของแนวคิดการเรียงลำดับข้อมูลพบว่า พิกเซลที่มีค่าความคลาดเคลื่อนจากการทำนายต่ำ เมื่อทำการฝังข้อมูลลงในพิกเซลเหล่านั้นโดยใช้ HS จะมีค่าความผิดพลาดที่ต่ำตามค่าความคลาดเคลื่อนจากการทำนายนั้น ในทางกลับกัน พิกเซลที่ไม่ถูกเรียงลำดับ หรือ พิกเซลที่มีบางค่าความคลาดเคลื่อนจากการทำนายสูงถูกฝังก่อน (ดูรูปที่ 2 (ก) ด้านขวามือ) จะมีค่าความผิดพลาดที่สูง และมีโอกาสเกิดปัญหา Overflow และ Underflow ดังที่กล่าวไว้ข้างต้น หนึ่งตัวอย่างสำหรับการฝังโดยใช้ HS

กรณี $d = 0$ เมื่อ $u' = 200, b = 1, T_p = 1$

$$U = 2d + b + u' = 2(0) + 1 + 200 = 201$$

กรณี $d = 2$ เมื่อ $u' = 200, b = 1, T_p = 1$

$$U = 2d + b + u' = 2(2) + 1 + 200 = 205$$

สังเกต การฝังข้อมูลลงใน $d = 0$ ให้ความบิดเบือนที่ต่ำกว่า $d = 2$ ที่ 4 ระดับ ดังนั้น การเรียงลำดับพิกเซลที่มีประสิทธิภาพเป็นสิ่งที่ต้องการ

หนึ่งปัญหาสำคัญของการเรียงลำดับที่ใช้บริเวณพิกเซลข้างเคียงของภาพที่มีการเปลี่ยนแปลงอย่างรวดเร็ว สำหรับคำนวณค่าความแปรปรวนที่สอดคล้อง สังเกตว่าลำดับของค่าความคลาดเคลื่อนจากการทำนายเรียงผิดพลาด (ดูรูปที่ 2 (ข) ด้านขวามือที่มีวงประสีแดง) ในหลายตำแหน่ง พบได้ในงานวิจัยก่อนหน้า [10] ในบทความเดียวกัน ผู้เขียนได้นำเสนอหนึ่งขั้นตอนวิธี RDH ร่วมกับอีกหนึ่งกลยุทธ์ใหม่สำหรับการปรับปรุงภาพคิวอาร์โค้ดโดยใช้การถ่วงน้ำหนักแบบเชิงเส้น (WL) วิธีการของพวกเขาบรรลุการแก้ปัญหาการฝังที่ก่อให้เกิด Overflow อย่างไรก็ตาม การเรียงลำดับข้อมูลโดยใช้ LV (ดูรูปที่ 2 (ข) ด้านซ้ายมือ) สำหรับภาพคิวอาร์โค้ด ยังคงให้ผลลัพธ์การทำนายที่ไม่ถูกต้อง สังเกตด้านขวามือในรูปเดียวกันพบบางค่าความคลาดเคลื่อนจากการทำนายที่สูงอยู่ก่อนค่าที่ต่ำตามเส้นประสีแดง ดังนั้น งานวิจัยฉบับนี้ นำเสนอหนึ่งการปรับปรุงเทคนิคการเรียงลำดับ โดยประยุกต์ใช้เทคนิค EVM ร่วมกับ WL ที่เหมาะสมเพื่อให้ได้การฝังมีประสิทธิภาพสูงสุด การใช้งานสำหรับการคำนวณ ex ที่เป็นคู่ (even) และคี่ (odd) ในงานวิจัย [9] ดังสมการที่ (2) และ (3) ถูกเขียนใหม่โดยใช้สมการที่ (4)

เพื่อให้สมการกระชับขึ้น หนึ่งความแปรปรวนใหม่ตามรูปแบบขยาย EVM จำนวนได้ดังนี้

$$\bar{\mu}_{i,j}^{ex} = \frac{\sum_{l=-\lfloor ex/2 \rfloor}^{\lfloor ex/2 \rfloor} \mu_{i+l,j+l} + \sum_{k=1}^{\lfloor ex/2 \rfloor} \sum_{l=-\lfloor ex/2 \rfloor}^{\lfloor ex/2 \rfloor - 2k} (\mu_{i+l,j+l+2k} + \mu_{i+l+2k,j+l})}{2\lceil ex/2 \rceil^2 + 2\lceil ex/2 \rceil + 1 - 4(ex \bmod 2)} \quad (4)$$

โดยที่ ex คือ ลำดับของการขยายพิกเซลสำหรับคำนวณ EVM ใหม่

l คือ ความยาวของการขยายพิกเซลออกในแต่ละมาร์ก

k คือ จำนวนของระดับการขยายพิกเซล

ข้อเท็จจริงพบว่า การขยายระดับของ ex (ดูรูปที่ 2 (ค)) มีผลต่อการใช้จำนวนพิกเซลที่สูงขึ้นสำหรับการคำนวณ EVM ใหม่ การขยายระดับของ ex ออกไปเป็นสิ่งที่ดี แต่ในบางเวอร์ชันของภาพคิวอาร์โค้ด การขยาย ex ออกไปไกลมาก ๆ มักเป็นสิ่งไม่จำเป็น ซึ่งไม่ได้ช่วยให้ค่า PSNR สูงขึ้น ดังนั้นในงานนี้ 30 ระดับของ ex จึงถูกใช้ในทดสอบ เพื่อให้ได้ผลลัพธ์ที่มีประสิทธิภาพสูงสุด รายละเอียดของขั้นตอนการฝัง และการกู้คืนอธิบายต่อในหัวข้อ 3.1 และ 3.2 ตามลำดับ

3.1 ขั้นตอนการฝังข้อมูล

1) นำภาพคิวอาร์โค้ดสีมาแยกเลเยอร์ทั้ง 3 ออกจากกัน (R, G, B)

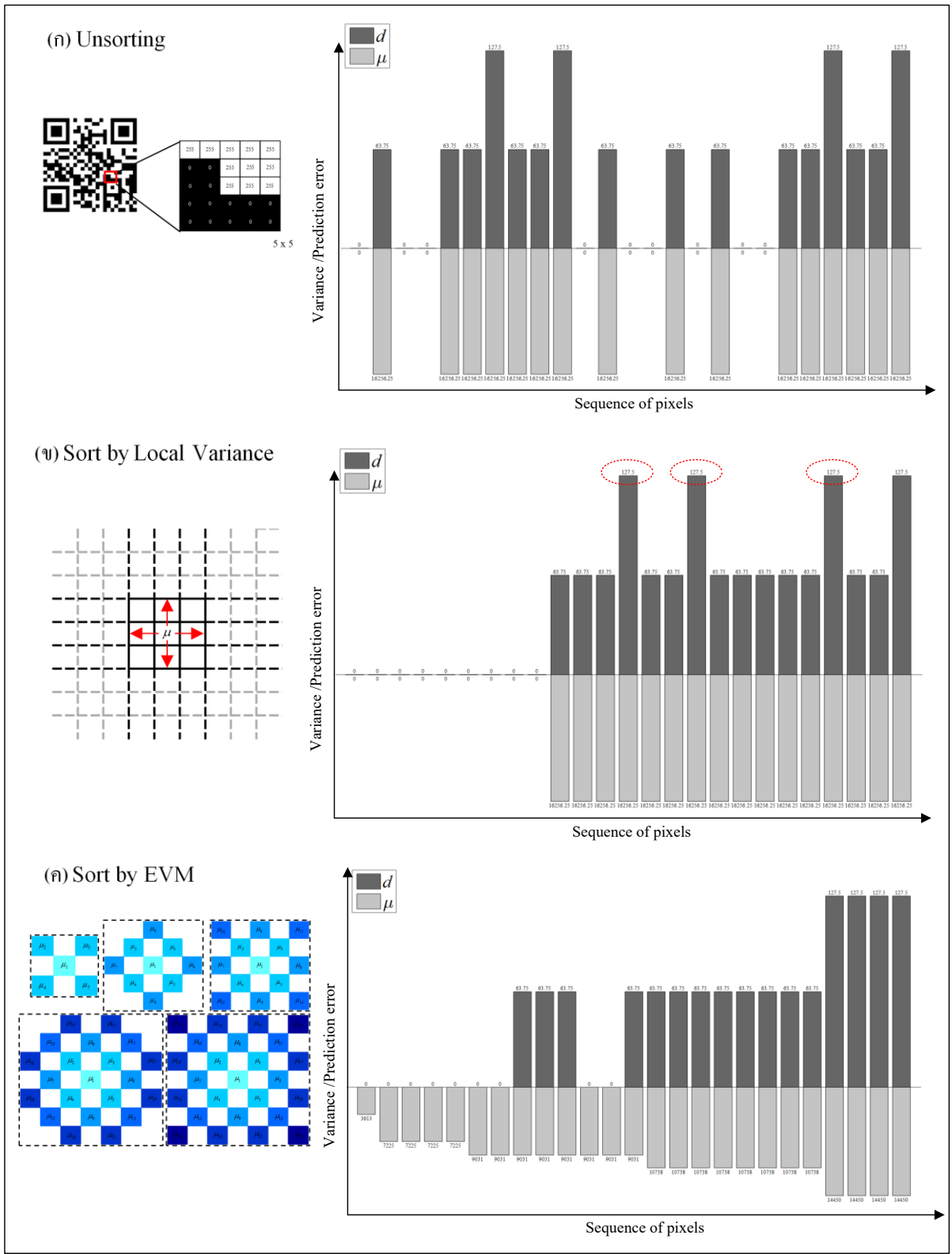
2) แบ่งพิกเซลในแต่ละเลเยอร์ออกเป็น 2 เซต (ครอสเซต และคอตเซต ตาม [9]) และปรับภาพด้วยค่าถ่วงน้ำหนัก W เริ่มจากค่าที่ 1 ตาม [10]

3) หาค่าการทำนาย u' , ค่าความคลาดเคลื่อนจากการทำนาย $d_{i,j}$ และค่าความแปรปรวน μ ตาม [7]

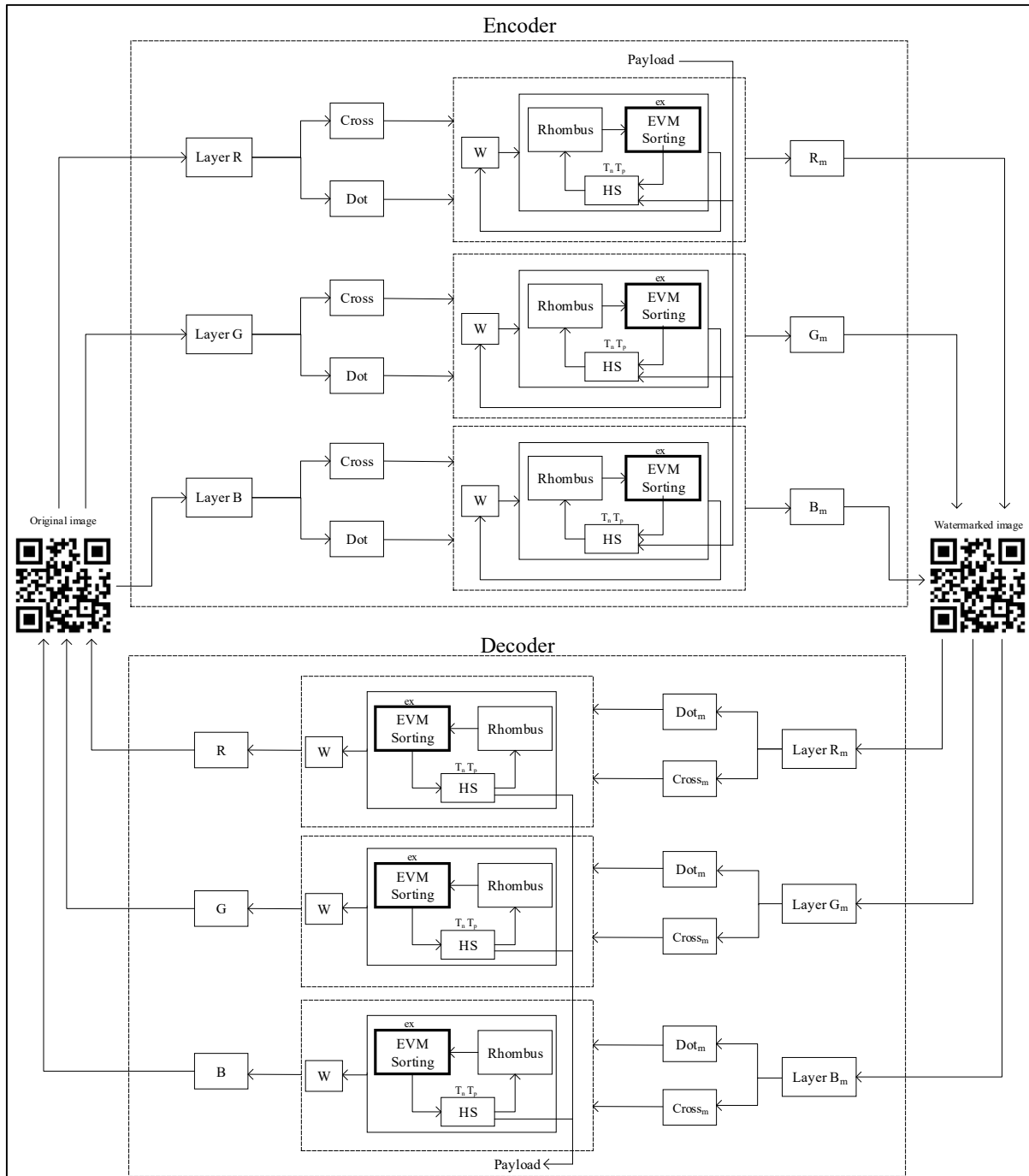
4) เรียงลำดับค่าความคลาดเคลื่อนจากการทำนาย $d_{i,j}$ โดยเรียงตามค่า EVM ใหม่ เริ่มจากค่าที่ 1

5) ใช้เทคนิค DMT ตาม [10] เพื่อตรวจสอบสถานะของทุกพิกเซลก่อนการฝัง

6) หาช่วงค่าขีดแบ่ง (T_n และ T_p) ที่เหมาะสมสำหรับการฝังข้อมูลโดยใช้เทคนิค HS ตาม [3]



รูปที่ 2 (ก) การเปรียบเทียบค่า d และ μ โดยไม่ใช้เทคนิค Sorting
 (ข) การเปรียบเทียบค่า d และ μ_{LV} โดยใช้เทคนิค Sorting
 (ค) การเปรียบเทียบค่า d และ μ_{EVM} โดยใช้เทคนิค Sorting



รูปที่ 3 โดอะแกรมกระบวนการฝังข้อมูล และกู้คืนข้อมูล

7) ทำซ้ำขั้นตอน 3-6 โดยเรียงลำดับค่าความกว้างของ EVM ขึ้นไปจนครบ 30 ระดับ

8) ปรับภาพด้วยค่าถ่วงน้ำหนัก W ค่าถัดไปและทำตามขั้นตอนที่ 4-10 จนถึงค่า W สุดท้าย

9) แยก LSB เสดเดอร์ขนาด 208 บิตสำหรับผู้รับเพื่อให้สามารถกู้คืนต้นฉบับ และข้อมูลที่ฝังได้ ซึ่งประกอบไปด้วยเพย์โหลด 56 บิต, ค่าจัดแบ่ง (T_n และ T_p) 16 บิต และค่าถ่วงน้ำหนักแบบเชิงเส้น 136 บิต

10) รวมเลขอร์สีทั้ง 3 เฟรมที่ถูกส่งข้อมูลแล้ว (R_m , G_m , B_m) เข้าด้วยกัน เสร็จสิ้นขั้นตอนการส่งข้อมูลสำหรับครอสเซต ในคอตเซตทำเหมือนกันกับครอสเซตตามลำดับ

3.2 ขั้นตอนการกู้คืนข้อมูล และภาพต้นฉบับ

1) แยกเลขอร์สีที่ถูกส่งข้อมูลทั้ง 3 ออกจากกัน (R_m , G_m , B_m)

2) แยกค่าเฮดเดอร์ขนาด 208 บิตที่กั้นไว้ก่อนหน้าสำหรับผู้รับของแต่ละเลขอร์ออกมา ซึ่งประกอบด้วยเพย์โหลดขนาด 56 บิต, ค่าขีดแบ่ง T_n และ T_p ขนาด 16 บิต และค่าถ่วงน้ำหนักแบบเชิงเส้นที่ได้จากการประมวลผลเบื้องต้นขนาด 136 บิต

3) แบ่งพิกเซลของแต่ละเลขอร์ออกเป็น 2 เซตตามขั้นตอนการส่ง คำนวณค่าการทำนาย u' และค่า EVM โดยเรียงลำดับค่าความคลาดเคลื่อนจากการทำนาย $d_{i,j}$ ตามค่า EVM

4) ตรวจสอบสถานะของพิกเซลโดยใช้ DMT และกู้คืนข้อมูล, กู้คืนค่าการทำนายคลาดเคลื่อน

5) นำภาพคิวอาร์โค้ดที่ได้กู้คืนปรับภาพด้วยค่าถ่วงน้ำหนัก W เพื่อปรับค่าเฉลี่ยให้กลับเป็นค่าดั้งเดิม

6) รวมเลขอร์สีดั้งเดิมทั้ง 3 เฟรม (R, G, B) เข้าด้วยกัน เสร็จสิ้นขั้นตอนการกู้คืนข้อมูล และภาพต้นฉบับ

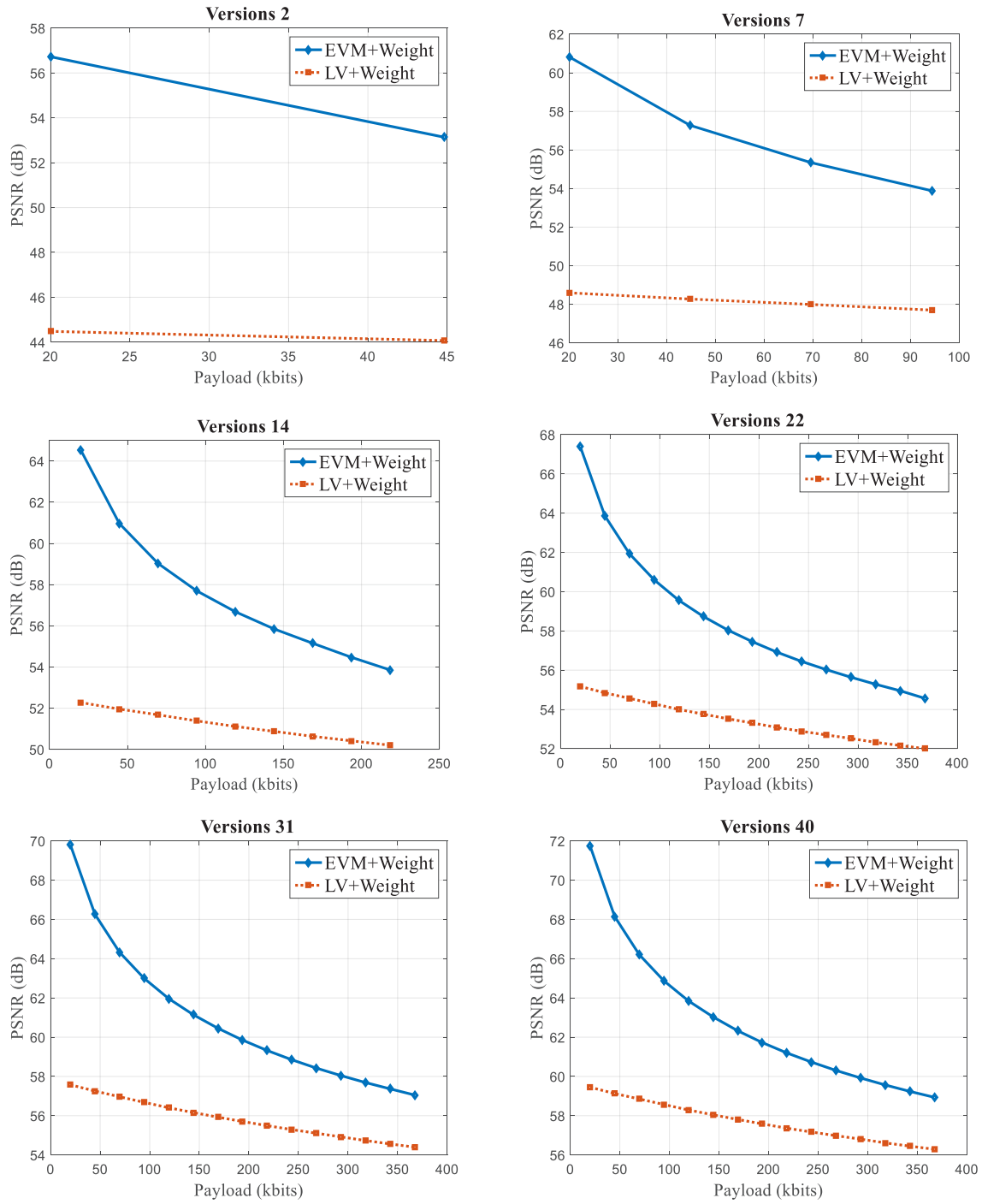
4. ผลการทดลอง

สำหรับการทดสอบประสิทธิภาพของวิธีการที่นำเสนอ ผู้วิจัยใช้ค่า PSNR (Peak Signal to Noise Ratio) ในการวัดผลลัพธ์เปรียบเทียบกับงาน [10] ภาพคิวอาร์โค้ดทั้งหมดมีการตั้งค่ามาตรฐานการสร้าง

เช่นเดียวกับในก่อนหน้า ข้อมูลที่ใช้ในการทดสอบเป็นบิตแบบสุ่ม ค่า WL ที่เหมาะสมถูกใช้ร่วมกับกระบวนการ RDH ที่ปรับปรุงประสิทธิภาพการเรียงลำดับ เพื่อให้ได้รับค่า PSNR สูงสุดที่เป็นไปได้ ในส่วนนี้ EVM ใหม่ถูกประยุกต์ใช้ ตาราง 1 แสดงประสิทธิภาพของวิธีการที่นำเสนอโดยส่งเพย์โหลดหลายขนาดกับการขยาย ex 1-30 ระดับกับภาพคิวอาร์โค้ดเวอร์ชัน 40 (เวอร์ชันสูงสุด ขนาดใหญ่สุด) สังเกตว่า ที่ปริมาณการส่งขนาดเล็ก (20 kbit) ให้ผลลัพธ์ PSNR ที่สูงกว่าการส่งขนาดใหญ่ (367.20 kbit) เฉลี่ยประมาณ 2 dB ในเกือบทุกการขยาย ระดับการขยายที่เหมาะสมมักสูงขึ้นตามขนาดของการส่ง เนื่องจาก 2 ปัจจัย คือ ขนาดของภาพ และความแปรปรวนของพิกเซลภายในภาพ ในตารางที่ 2 สังเกตขนาดของการส่งที่ 367.20 kbits ภาพคิวอาร์โค้ดที่ส่งได้จะเริ่มต้นที่เวอร์ชัน 20-40 การขยาย ex ให้ผลลัพธ์ค่า PSNR ที่สูงกว่าแบบที่ไม่ได้ขยาย ในรูปที่ 6 สังเกตว่าที่เพย์โหลดขนาดเล็กให้ความบิดเบือนที่ต่ำมาก ในทางเดียวกันค่า PSNR ที่สูงตั้งแต่เริ่มต้นการส่งเมื่อเทียบกับ non- ex [10] กราฟค่า PSNR แสดงให้เห็นผลลัพธ์การบิดเบือนที่ต่ำกว่า หรือในอีกความหมายหนึ่งการเรียงลำดับข้อมูลโดยการขยายพิกเซลสำหรับการทำนายใหม่ให้ผลลัพธ์ที่ถูกต้องเข้าใกล้ค่าความคลาดเคลื่อนจากการทำนายสูงกว่าแบบที่ไม่มีการขยายพิกเซล พิกเซลที่มีการแปรปรวนสูง ๆ ในภาพให้การตอบสนองที่ดีมากสำหรับการประยุกต์ใช้ EVM ร่วมกับ WL ที่เหมาะสม ถึงแม้ว่าการคำนวณจะสูงขึ้น ค่า PSNR ที่ได้รับ และขนาดของการส่งที่ใหญ่ขึ้นเป็นสิ่งที่น่าพึงพอใจสำหรับการนำเสนอ

ตารางที่ 1 ผลลัพธ์การฝังข้อมูล ex 1-30 ของภาพ QR Code เวอร์ชัน 40

ex	Payload (kbits)							
	20.00		119.20		367.20		367.20	
	PSNR	Weight	PSNR	Weight	PSNR	Weight	PSNR	Weight
1	71.558	0.98431	63.792	0.98431	61.695	0.98431	58.918	0.97647
2	71.562	0.98431	63.816	0.98431	61.708	0.98431	58.916	0.97647
3	71.582	0.98431	63.817	0.98431	61.724	0.98431	58.925	0.97647
4	71.599	0.98431	63.816	0.98431	61.703	0.98431	58.919	0.97647
5	71.660	0.98431	63.822	0.98431	61.705	0.98431	58.922	0.97647
6	71.590	0.98431	63.821	0.98431	61.705	0.98431	58.913	0.97647
7	71.594	0.98431	63.830	0.98431	61.708	0.98431	58.919	0.97647
8	71.603	0.98431	63.821	0.98431	61.708	0.98431	58.920	0.97647
9	71.581	0.98431	63.825	0.98431	61.705	0.98431	58.921	0.97647
10	71.611	0.98431	63.822	0.98431	61.716	0.98431	58.930	0.97647
11	71.607	0.98431	63.826	0.98431	61.717	0.98431	58.926	0.97647
12	71.600	0.98431	63.831	0.98431	61.709	0.98431	58.932	0.97647
13	71.600	0.98431	63.828	0.98431	61.715	0.98431	58.927	0.97647
14	71.621	0.98431	63.823	0.98431	61.711	0.98431	58.919	0.97647
15	71.612	0.98431	63.847	0.98431	61.713	0.98431	58.927	0.97647
16	71.618	0.98431	63.820	0.98431	61.715	0.98431	58.928	0.97647
17	71.609	0.98431	63.820	0.98431	61.715	0.98431	58.922	0.97647
18	71.634	0.98431	63.829	0.98431	61.711	0.98431	58.922	0.97647
19	71.625	0.98431	63.823	0.98431	61.710	0.98431	58.923	0.97647
20	71.634	0.98431	63.831	0.98431	61.712	0.98431	58.930	0.97647
21	71.617	0.98431	63.820	0.98431	61.708	0.98431	58.927	0.97647
22	71.595	0.98431	63.832	0.98431	61.715	0.98431	58.921	0.97647
23	71.606	0.98431	63.828	0.98431	61.707	0.98431	58.923	0.97647
24	71.642	0.98431	63.828	0.98431	61.718	0.98431	58.927	0.97647
25	71.668	0.98431	63.829	0.98431	61.713	0.98431	58.926	0.97647
26	71.611	0.98431	63.829	0.98431	61.715	0.98431	58.926	0.97647
27	71.629	0.98431	63.833	0.98431	61.708	0.98431	58.932	0.97647
28	71.614	0.98431	63.834	0.98431	61.711	0.98431	58.927	0.97647
29	71.600	0.98431	63.829	0.98431	61.718	0.98431	58.930	0.97647
30	71.606	0.98431	63.828	0.98431	61.707	0.98431	58.932	0.97647



รูปที่ 6 กราฟแสดงค่า PSNR VS Payload ของวิธีการที่นำเสนอ (EVM+Weight) โดยเปรียบเทียบกับงานในอดีต (LV+Weight) [10] สำหรับ 6 ภาพคิวอาร์โค้ด

ตารางที่ 2 ผลลัพธ์ค่า PSNR และค่าถ่วงน้ำหนักที่ดีที่สุดสำหรับภาพคิวอาร์โค้ด 40 เวอร์ชัน

QR Code Image	Payload (20 kbits)			Payload (193.60 kbits)			Payload (367.20 kbits)		
	PSNR (dB)	Weight	<i>ex</i>	PSNR (dB)	Weight	<i>ex</i>	PSNR (dB)	Weight	<i>ex</i>
Version 1	55.575	0.98824	12	-	-	-	-	-	-
Version 2	56.724	0.98039	30	-	-	-	-	-	-
Version 3	57.703	0.97255	30	-	-	-	-	-	-
Version 4	58.611	0.98039	15	-	-	-	-	-	-
Version 5	59.409	0.97255	30	-	-	-	-	-	-
Version 6	60.134	0.98824	12	-	-	-	-	-	-
Version 7	60.824	0.97255	30	-	-	-	-	-	-
Version 8	61.463	0.98039	15	-	-	-	-	-	-
Version 9	62.068	0.98824	12	-	-	-	-	-	-
Version 10	62.639	0.98824	30	-	-	-	-	-	-
Version 11	63.118	0.98039	8	-	-	-	-	-	-
Version 12	63.625	0.97255	12	53.649	0.97255	30	-	-	-
Version 13	64.056	0.98039	30	53.956	0.97255	8	-	-	-
Version 14	64.547	0.98039	15	54.471	0.97255	5	-	-	-
Version 15	64.957	0.98039	15	54.975	0.98431	30	-	-	-
Version 16	65.336	0.98824	30	55.367	0.98039	30	-	-	-
Version 17	65.710	0.98824	30	55.758	0.98039	12	-	-	-
Version 18	66.076	0.98824	12	56.115	0.97255	30	-	-	-
Version 19	66.419	0.97255	12	56.466	0.98039	5	-	-	-
Version 20	66.792	0.97255	30	56.814	0.98824	30	53.814	0.97255	12
Version 21	67.126	0.98431	30	57.139	0.98824	12	54.208	0.98431	15
Version 22	67.388	0.98039	8	57.447	0.98039	30	54.566	0.98039	12
Version 23	67.719	0.98039	8	57.751	0.98431	30	54.948	0.97255	12
Version 24	68.003	0.98039	15	58.037	0.98039	30	55.252	0.98431	30
Version 25	68.276	0.98039	30	58.322	0.98039	8	55.532	0.98431	30
Version 26	68.554	0.97255	30	58.601	0.97255	30	55.799	0.97255	30
Version 27	68.822	0.97255	8	58.858	0.97255	30	56.068	0.97647	12
Version 28	69.076	0.98039	30	59.117	0.98039	5	56.326	0.97647	12
Version 29	69.309	0.97255	12	59.374	0.98039	8	56.575	0.98039	30
Version 30	69.564	0.98824	12	59.616	0.97255	12	56.816	0.97255	8
Version 31	69.802	0.97647	30	59.852	0.98824	30	57.053	0.98824	30
Version 32	70.025	0.98039	30	60.074	0.98824	12	57.287	0.97255	12
Version 33	70.284	0.97255	12	60.311	0.98039	15	57.510	0.97255	30
Version 34	70.475	0.97647	30	60.524	0.98824	30	57.725	0.98824	30
Version 35	70.688	0.97647	30	60.738	0.98824	30	57.939	0.98824	30
Version 36	70.896	0.97647	30	60.946	0.98824	30	58.147	0.98824	30
Version 37	71.092	0.98039	30	61.142	0.98824	12	58.354	0.97255	12
Version 38	71.326	0.97255	12	61.353	0.98039	15	58.551	0.97255	30
Version 39	71.487	0.98039	30	61.543	0.97255	30	58.747	0.98039	30
Version 40	71.743	0.98824	15	61.726	0.98039	3	58.935	0.97255	12

5. สรุป

งานวิจัยนี้นำเสนอหนึ่งการปรับปรุงวิธีการเรียงลำดับข้อมูลบนพื้นฐานกระบวนการ RDH ดั้งเดิม สำหรับภาพคิวอาร์โค้ดโดยประยุกต์ใช้เทคนิค EVM ร่วมกับ WL การปรับระดับที่เหมาะสมถูกใช้ประโยชน์อย่างมีนัยสำคัญต่อความบิดเบือนของภาพหลังการส่ง การขยายพิกเซล

สำหรับคำนวณค่าการเรียงลำดับใหม่ของค่าความคลาดเคลื่อนจากการทำนายเป็นสิ่งจำเป็น โดยเฉพาะภาพที่มีความแปรปรวนของพิกเซลสูง ๆ วิธีการที่นำเสนอให้ปริมาณการส่งที่สูงกว่า ในขณะที่ความบิดเบือนหลังการส่งต่ำกว่าเมื่อเปรียบเทียบกับงานเมื่อเร็ว ๆ นี้

6. กิตติกรรมประกาศ

ขอขอบคุณสถาบันวิจัยและพัฒนา มหาวิทยาลัยเทคโนโลยีราชมงคลรัตนโกสินทร์ สำหรับทุนสนับสนุนการทำวิจัย

เอกสารอ้างอิง

- [1] Tian, J. Reversible Data Embedding Using a Difference Expansion. *IEEE Transactions on Circuits and Systems for Video Technology*, 2003; 13(8): 890-896.
- [2] Thodi, D. M. and Rodriguez, J. J. Reversible Watermarking by Prediction-Error Expansion. 6th IEEE Southwest Symposium on Image Analysis and Interpretation, 28-30 March, Lake Tahoe, NV, USA, 2004; 21-25.
- [3] Thodi, D. M. and Rodriguez, J. J. Expansion Embedding Techniques for Reversible Watermarking. *IEEE Transactions on Image Processing*, 2007; 16(3): 721-730.
- [4] Ni, Z., Shi, Y., Ansari, Q. N. and Su, W. Reversible Data Hiding. *IEEE Transactions on Circuits and Systems for Video Technology*, 2006; 16(3): 354-362.
- [5] Kim, H. J., Sachnev, V., Shi, Y. Q., Nam, J. and Choo, H. G. A Novel Difference Expansion Transform for Reversible Data Embedding. *IEEE Transactions on Information Forensics and Security*, 2008; 3(3): 456-465.
- [6] Kamstra, L. H. J. and Heijmans, A. M. Reversible Data Embedding into Images Using Wavelet Techniques and Sorting. *IEEE Transactions on Image Processing*, 2005; 14(12): 2082-2090.
- [7] Sachnev, V., Kim, H. J., Nam, J. Suresh, S. and Shi, Y.-Q. Reversible Watermarking Algorithm Using Sorting and Prediction. *IEEE Transactions on Circuits and Systems for Video Technology*, 2009; 19(7): 989-999.
- [8] Kang, S., Hwang, H. J. and Kim, H. J. Reversible Watermarking Using an Accurate Predictor and Sorter Based on Payload Balancing. *ETRI Journal*, 2012; 34(4): 410-420.
- [9] Kotvicha, A., Sanguansat, P. and Kasemsan, M.L.K. Expand Variance Mean sorting for Reversible Watermarking. 2012 IEEE International Conference on Signal Processing, Communications and Computing, 12-15 August, Hong Kong, 2012; 668-673.
- [10] ชัยพร ปานยินดี ชีรนัย สิลาวิวัฒน์ และภัทรวดี รังสิริรักษ์. การศึกษาและพัฒนาขั้นตอนวิธีการฝังข้อมูลแบบกู้คืนกลับได้โดยใช้การถ่วงน้ำหนักแบบเชิงเส้นสำหรับภาพคิวอาร์โค้ด. *วารสารวิศวกรรมศาสตร์มหาวิทยาลัยเชียงใหม่*. 2560; 1.
- [11] ปรีญาทิพย์ แดงมี และวราพล สิลาเกียรติสกุล. การอำพรางข้อมูลลับด้วย Steganography บน QR Code. *Journal of Information Science and Technology*, 2015; 5(1): 35-41.