SWU Engineering Journal (2017) 12(1), 20-32
วารสารวิศวกรรมศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ
ปีที่ 12 ฉบับที่ 1 เดือนมกราคม – มิถุนายน พ.ศ.2560

การรวมกันของวิทยาการอำพรางข้อมูลกับวิทยาการเข้ารหัสลับสำหรับภาพทางการแพทย์

Combination of Steganography with Cryptography for Medical Images

ชัยพร ปานยินดี พุธภรณ์ เอี่ยมภาษี และนิษฐา อรุณสินประเสริฐ
สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลรัตนโกสินทร์
*Corresponding author: chaiyaporn.pan@gmail.com

บทคัดย่อ

งานวิจัยนี้นำเสนอการรวมกันของสองขั้นตอนวิธีประกอบด้วยการอำพรางข้อมูลแบบที่สามารถกู้คืนกลับได้ (Reversible Data Hiding: RDH) และการเข้ารหัสลับ (Advanced Encryption Standard: AES) เพื่อเพิ่มประสิทธิภาพความปลอดภัยในการเข้าถึงข้อมูล หลายเทคนิคของ RDH ถูกใช้ร่วมกันเพื่อให้ได้รับความบิดเบือนต่ำสุดสำหรับการซ่อนข้อมูล หนึ่งตัวทำนาย Linear Fitting Rhombus Pattern (LFRP) ถูกใช้สำหรับการทำนาย, Local variance ใช้สำหรับการเรียงค่าความผิดพลาดจากการทำนาย, Double Modification Testing (DMT) ใช้เพื่อการตรวจสอบสถานะของพิกเซล และเทคนิค Histogram Shifting ใช้ในการฝัง มากไปกว่านั้น ขั้นตอนวิธี AES ถูกประยุกต์ใช้ร่วมในงานนี้สำหรับการเข้ารหัสลับอีกชั้นหนึ่งสำหรับข้อมูล Header 128 บิต ของขั้นตอนวิธีการเข้ารหัส RDH เพื่อให้แน่ใจสำหรับการป้องกันการเข้าถึงข้อมูลโดยบุคคลที่ไม่ได้รับอนุญาต การทดสอบภาพแบบไบนารีหลายขนาดถูกใช้ฝังลงในภาพทางการแพทย์ซึ่งได้รับมาจากเครื่องมือที่แตกต่างกัน อาทิเช่น Magnetic Resonance Image (MRI) Ultrasound (US) และ X-ray ผลลัพธ์ขั้นตอนวิธีที่นำเสนอแสดงให้เห็นความบิดเบือนของการฝังที่ต่ำ และความปลอดภัยของการเข้าถึงข้อมูลที่สูงขึ้น

คำสำคัญ: การอำพรางข้อมูลแบบที่สามารถกู้คืนกลับได้ (RDH) การเข้ารหัสลับ (AES)

ABSTRACT

This paper presents two algorithms, Reversible Data Hiding (RDH) and Advanced Encryption Standard (AES) to enhance the security of unauthorized data access. Many techniques of RDH can be shared to achieve minimal distortion when hiding information. A Linear Fitting Rhombus Pattern Predictor (LFRPP) was used for prediction, with, local variance to sort prediction error values. Double Modification Testing (DMT) was used to check the status of pixels with Histogram Shifting (HS) employed for data embedding. The AES algorithm was applied for encryption 128 bit RDH encoder algorithm Header to ensure data protection and restrict access by unauthorized persons. Various quantities of binary information embedded into medical imaging and derived from the diverse sources of Magnetic Resonance Image (MRI), Ultrasound (US) and X-ray were tested. Results showed a distortion between embedding low and higher data security.

Keyword: Reversible Data Hiding, Advanced Encryption Standard

1. บทนำ

ความก้าวหน้าของเทคโนโลยีการสื่อสารไร้สาย (Wireless LAN Technology) เป็นที่นิยมอย่างมากในปัจจุบัน การรับส่งข้อมูลทางการแพทย์ถือว่าเป็นสิ่งจำเป็นเพื่อให้การรักษามีประสิทธิภาพ และลดต้นทุนสำหรับการจ้างบุคลลากรผู้เชี่ยวชาญ แพทย์หรือผู้วินิจฉัยสามารถวางแผนการรักษาให้ผู้ป่วยได้โดยไม่จำเป็นต้องอยู่ที่โรงพยาบาล ขั้นตอนวิธีการฝังข้อมูลแบบกู้คืนกลับได้ (Reversible Data Hiding) มักถูกประยุกต์ใช้ในหลายงาน โดยเฉพาะงานทางการแพทย์ โดยทั่วไปข้อมูลของผู้ป่วย เช่น ชื่อ นามสกุล น้ำหนัก ส่วนสูง ประวัติการรักษา ฯลฯ จะถูกปกป้องไม่ให้ผู้อื่นเข้าถึงข้อมูลได้เพื่อเอาไปใช้ประโยชน์อื่นใด ซึ่งถือเป็นจรรณยาบรรณของแพทย์ ข้อมูลดังกล่าวจะถูกฝังไปกับภาพทางการแพทย์ เพื่อลดพื้นที่ในการจัดส่งข้อมูล รวมถึงป้องกันการเข้าถึง ในอดีตมีหลายงานวิจัยสำหรับวิธีการฝังข้อมูลที่สามารถกู้คืนกลับได้ ซึ่งพัฒนาอย่างต่อเนื่อง อาทิเช่น Difference Expansion (DE) [1]-[3], Sorting [4]-[5], Prediction-error Expansion (PEE) [6] และ Histogram Shifting (HS) [7]-[9] เป็นต้น เทคนิคเหล่านี้มีวัตถุประสงค์เดียวกัน คือ ต้องการฝังข้อมูลในปริมาณที่สูง และหลังการฝังต้องมีความบิดเบือนที่ต่ำ อย่างไรก็ตาม การอำพรางข้อมูล หรือ ซ่อนข้อมูลเพียงอย่างเดียว ยังคงมีข้อจำกัดในส่วนของ Header ที่ต้องใส่ข้อมูลโดยตรงเพื่อให้ผู้รับสามารถถอดรหัสข้อมูลที่ฝังได้ การเข้ารหัสลับถูกนำมาใช้ร่วมด้วย โดยปกติ ศาสตร์ของการเข้ารหัสลับจะแปลงข้อมูลปกติ (Plaintext) ให้กลายเป็นข้อความลับ (Ciphertext) ที่ไม่สามารถเข้าใจได้

วิทยาการเข้ารหัสลับสามารถแบ่งออกได้เป็น 2 แบบเรียกว่า Symmetric Algorithm และ Asymmetric Algorithm ขั้นตอนวิธีของการเข้ารหัสลับมีอยู่ด้วยกันหลายเทคนิค อาทิเช่น Data Encryption Standard (DES) [14], RC4 [15] และ RSA [16] การประยุกต์ใช้ขึ้นอยู่กับเงื่อนไข และองค์ประกอบที่นำไปใช้ในงานนั้นๆ

สำหรับบทความนี้ได้นำเสนอการประยุกต์ใช้ AES ร่วมกับ RDH เพื่อประสิทธิภาพสูงสุดสำหรับการป้องกันการเข้าถึงข้อมูลที่ไม่ต้องการเปิดเผย โดยแนวคิดที่นำเสนอ Header ของขั้นตอนวิธี RDH จะนำมาเข้ารหัสลับโดยใช้ขั้นตอนวิธี AES ก่อนจะถูกส่งให้ผู้รับเพื่อถอดรหัสข้อมูลดังกล่าว

รายละเอียดส่วนอื่น ๆ ของบทความมีดังต่อไปนี้ ส่วนที่ 2 อธิบายงานวิจัยที่เกี่ยวข้อง ส่วนที่ 3 อธิบายวิธีการที่นำเสนอโดยประยุกต์ใช้ RDH ร่วมกับวิทยาการเข้ารหัสลับ AES ส่วนที่ 4 อธิบายผลการทดลอง และส่วนที่ 5 สรุปผลการทดลอง

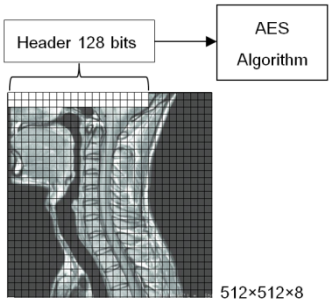
2. งานวิจัยที่เกี่ยวข้อง

หนึ่งขั้นตอนวิธีของ RDH ที่มีประสิทธิภาพสูงในการลดความบิดเบือนของภาพหลังการฝัง ได้แก่งานของ Sachnev และคณะ [10] งานของพวกเขารวมหลายเทคนิค ซึ่งประกอบด้วย PEE, HS, Sorting, Double Modification Testing และ double embedding เป็นต้น หนึ่งตัวทำนายที่มีชื่อว่า Rhombus ถูกใช้ในงานนี้ด้วยเพื่อให้ได้ค่า PE ที่มีค่าต่ำ และสามารถเรียงลำดับได้ งานของพวกเขาถูกปรับปรุงในหลายแง่มุม หนึ่งในการปรับปรุงถูกนำเสนอโดย Panyindee และคณะ [11] งานพวกเขาปรับปรุงตัวทำนาย Rhombus โดยใช้ฟังก์ชั่น Linear fitting ซึ่งเป็นข้อได้เปรียบสำหรับการถ่วงน้ำหนักที่เหมาะสมในแต่ละภาพ และแต่ละขนาดของการฝัง เพื่อให้ได้รับค่า PE ที่มีค่าต่ำสุด สังเกตว่าค่าถ่วงน้ำหนักเดิมของตัวทำนาย Rhombus คือ 0.25 เท่ากันทั้งสี่ตำแหน่ง ซึ่งคงที่ไม่มีการเปลี่ยนแปลง ค่าถ่วงน้ำหนักนี้เหมาะเฉพาะบางภาพ และบางขนาดของการฝังเท่านั้น ตัวนำนาย LFRP ของ [11] ให้ผลลัพธ์ที่ถูกต้องสูงกว่าตัวทำนาย Rhombus เดิม อย่างไรก็ตาม การปรับปรุงตัวทำนายเพียงอย่างเดียวไม่สามารถยกระดับผลลัพธ์ทั้งหมดของกระบวนการ Panyindee และคณะ [13] นำเสนออีกหนึ่งการปรับปรุงที่สำคัญ ตัวทำนายใหม่ที่สามารถปรับเปลี่ยนได้ถูกเรียกว่า Gaussian Weighted Predictor ถูกนำเสนอ มากไปกว่านั้น ประสิทธิภาพที่สูงขึ้นของกระบวนการ ขั้นตอนวิธีเชิงพันธุกรรม (GA) ถูกใช้ร่วมเพื่อค้นหาพารามิเตอร์ที่เหมาะสมสำหรับการทำนาย และการเรียงลำดับ ผลลัพธ์ของพวกเขาดีกว่างานก่อนหน้าทั้งหมดที่กล่าวถึง อย่างไรก็ตาม การอำ

พรางข้อมูลโดยอาศัยการลดความบิดเบือนเพียงอย่างเดียวยังคงไม่สามารถเป็นหลักประกันสำหรับการป้องกันการเข้าถึงข้อมูลได้ ดังนั้น งานวิจัยนี้จึงเพิ่มประสิทธิภาพในส่วนของการป้องกันการเข้าถึงข้อมูลโดยใช้การเข้ารหัสลับ AES ร่วมกับ RDH [11] รายละเอียดของขั้นตอนต่างๆ จะอธิบายต่อในหัวข้อ 2.1-2.2 ตามลำดับ

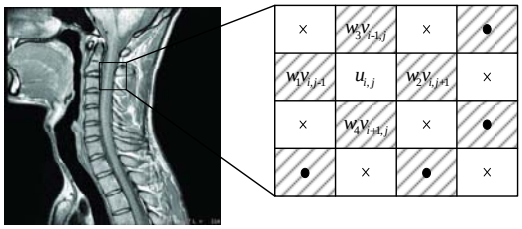
2.1 Reversible Data Hiding [10]

การรวมกันอย่างมีประสิทธิภาพของหลายเทคนิค RDH นำเสนอโดย [10] เพื่อให้การฝังมีการบิดเบือนต่ำสุดสำหรับการอำพรางข้อมูล และสามารถกู้คืนกลับได้ทั้งภาพต้นฉบับ และข้อมูลที่ถูกฝัง จำเป็นต้องฝังข้อมูลในส่วนของ Header สำหรับผู้รับข้อมูลในส่วนนี้ประกอบด้วย ขนาดของข้อมูลที่ต้องการฝัง (Payload) และช่วงของค่าขีดแบ่ง(Thresholds) $[T_n, T_p]$ แสดงดังรูปที่ 1



รูปที่ 1 การประยุกต์ใช้ AES ร่วมกับ RDH สำหรับภาพทางการแพทย์

2.1.1 Prediction-Error Expansion [6] เป็นอีกหนึ่งเทคนิคของ RHD ซึ่งเป็นที่รู้จักอย่างกว้างขวางในหมู่นักวิจัยด้านนี้ เทคนิค PEE ปฏิวัติการฝังข้อมูลใหม่โดยฝังบิตข้อมูลลงในค่าความผิดพลาดจากการทำนายแทนการฝังโดยตรงลงในค่าความแตกต่าง (DE) [1] ซึ่งให้ผลลัพธ์ความบิดเบือนที่ต่ำกว่าเทคนิค-DE เดิม หนึ่งตัวทำนายถูกใช้ PEE ถูกปรับปรุงอย่างต่อเนื่องโดยการใช้ตัวทำนายในรูปแบบใหม่ๆ LFRP เป็นหนึ่งในนั้น โดยรายละเอียดสามารถอ่านเพิ่มเติมได้ใน [11] กระบวนการเริ่มต้นจากการแบ่งพิกเซล



รูปที่ 2 การแบ่งครอสเซต และดอทเซต

ทั้งหมดของภาพออกเป็น 2 เซต คือ ครอสเซต (Cross set) และดอทเซต (Dot set) กระบวนการฝังแบบดับเบิลเกี่ยวข้องกับการใช้งานที่ต่อเนื่องกันของการฝังครอส (Cross Embedding Scheme) และการฝังดอท (Dot Embedding Scheme) ผลลัพธ์ที่ได้เกือบสองเท่าของความจุในการฝัง ความจุสูงสุดจะเพิ่มขึ้นถึง 1 บิต/พิกเซล (จาก 0.5 บิต/พิกเซล เมื่อใช้เพียงการฝังครอส) กระบวนการฝังครอส รูปแบบการฝังในครอสเซตจะถูกคำนวณค่าการทำนายโดยใช้ดอทเซต และฝังข้อมูลโดยใช้ครอสเซต สังเกตว่า พิกเซล $u_{i,j}$ อยู่ในครอสเซตมีสี่พิกเซลข้างเคียง $v_{i-1,j}$, $v_{i+1,j}$, $v_{i,j+1}$, และ $v_{i,j-1}$ ซึ่งอยู่ในตำแหน่งของดอทเซต ค่าการทำนาย $u'_{i,j}$ ที่ใช้สี่พิกเซลข้างเคียงถูกคำนวณโดยสมการที่ (1) และค่าความผิดพลาดจากการทำนาย $d_{i,j}$ ถูกคำนวณเพื่อที่จะซ่อนข้อมูล หลังการซ่อนข้อมูลโดยใช้รูปแบบการฝังครอส พิกเซลจากครอสเซต $u_{i,j}$ ถูกเปลี่ยนเป็น $U_{i,j}$ เหมือนกันในรูปแบบการฝังดอท พิกเซลศูนย์กลาง $v_{i,j}$ อยู่ในดอทเซต และสี่บริเวณพิกเซลข้างเคียง $U_{i-1,j}$, $U_{i+1,j}$, $U_{i,j+1}$, และ $U_{i,j-1}$ อยู่ในครอสเซต นอกจากนี้ขนาดของข้อมูลที่ต้องการฝัง ควรจะถูกแบ่งออกเป็นสองส่วนที่มีขนาดใกล้เคียงกัน หรือ เท่ากัน สำหรับรูปแบบการฝังในครอส และในดอท เห็นได้ชัดเจนว่าความจุของกระบวนการฝังแบบเดี่ยว (Single Embedding) มีค่าต่ำกว่าเมื่อเปรียบเทียบกับกระบวนการฝังแบบดับเบิล (Dabble Embedding) การแบ่งพิกเซลในลักษณะนี้ส่งผลให้พิกเซลสามารถเรียงลำดับได้ซึ่งจะอธิบายต่อในหัวข้อย่อยที่ 2.1.3 ตัวทำนายใหม่ LFRP ถูกใช้เพื่อทำนายค่า u' จากพิกเซลข้างเคียงสามารถคำนวณได้ดังนี้

$$u' = \left\lfloor \frac{w_1 v_{i,j-1} + w_2 v_{i,j+1} + w_3 v_{i-1,j} + w_4 v_{i+1,j}}{\sum w_{i,j}} \right\rfloor \quad (1)$$

ค่าความผิดพลาดจากการทำนาย $d_{i,j}$ (Prediction Error: PE) สามารถคำนวณได้ดังนี้

$$d_{i,j} = u_{i,j} - u'_{i,j} \quad (2)$$

การขยายค่า PE สำหรับการฝังหนึ่งบิต b ซึ่งในที่นี้มีค่าที่เป็นไปได้คือ [1, 0] สามารถคำนวณได้ดังนี้

$$D_{i,j} = 2d_{i,j} + b \quad (3)$$

พิกเซลใหม่ที่ถูกโมดิฟาย ($U_{i,j}$) สามารถคำนวณได้ดังสมการต่อไปนี้

$$U_{i,j} = D_{i,j} + u'_{i,j} \quad (4)$$

เมื่อทำการฝังในครอสเซตเสร็จสิ้น กระบวนการฝังในดอทเซตจะเริ่มต้น รูปแบบการฝังในครอสเซตและดอทเซตจะเหมือนกัน ถูกเรียกว่า Double Embedding [10] ที่กล่าวไว้ก่อนหน้า

กระบวนการกู้คืนข้อมูลสำหรับครอสเซตสามารถคำนวณได้ดังนี้

$$D_{i,j} = U_{i,j} - u'_{i,j} \quad (5)$$

บิตข้อมูลที่ถูกฝังคำนวณได้จาก

$$b = D_{i,j} \bmod 2 \quad (6)$$

ค่า PE สามารถคำนวณได้ดังนี้

$$d_{i,j} = \left\lfloor D_{i,j}/2 \right\rfloor \quad (7)$$

และค่าพิกเซลต้นฉบับสามารถกู้คืนได้ดังนี้

$$u_{i,j} = d_{i,j} + u'_{i,j} \quad (8)$$

2.1.2 Histogram Shifting (HS) [7] เป็นหนึ่งเทคนิคที่ช่วยลดความบิดเบือนของภาพหลังการฝังการกำหนดช่วงของค่าขีดแบ่งที่เหมาะสม T_n (ค่าขีดแบ่งฝั่งลบ) และ T_p (ค่าขีดแบ่งฝั่งบวก) เป็นสิ่งจำเป็นในการแยกบริเวณที่ต้องการฝังข้อมูล และบริเวณที่ไม่ต้องการฝังข้อมูล เพื่อให้บรรลุความบิดเบือนต่ำสุดที่เป็นไปได้ ประโยชน์อื่นๆ ของเทคนิคนี้ยังช่วยหลีกเลี่ยงปัญหาการซ้อนทับระหว่างการฝังข้อมูล ละเอียดสามารถอ่านเพิ่มเติมได้ที่ [7] กระบวนการฝังข้อมูลของ HS สามารถคำนวณได้ดังนี้

$$D_{i,j} = \begin{cases} 2d_{i,j} + b, & \text{if } d_{i,j} \in \left[T_n\,;T_p\right] \\ d_{i,j} + T_p + 1, & \text{if } d_{i,j} > T_p \text{ and } T_p \geq 0 \\ d_{i,j} + T_n, & \text{if } d_{i,j} < T_n \text{ and } T_n < 0 \end{cases} \quad (9)$$

การกู้คืนข้อมูลต้นฉบับ และบิตที่ถูกฝังสำหรับเทคนิค HS สามารถคำนวณได้ดังสมการต่อไปนี้

$$d_{i,j} = \begin{cases} \left\lfloor D_{i,j}/2 \right\rfloor, & \text{if } D_{i,j} \in \left[2T_n\,;2T_p+1\right] \\ D_{i,j} - T_p - 1, & \text{if } D_{i,j} > 2T_p+1 \text{ and } T_p \geq 0 \\ D_{i,j} - T_n, & \text{if } D_{i,j} < 2T_n \text{ and } T_n < 0 \end{cases} \quad (10)$$

$$b = D_{i,j} \bmod 2, \quad D_{i,j} \in \left[2T_n\,;2T_p+1\right] \quad (11)$$

2.1.3 Sorting Data [4] ถูกใช้เรียงลำดับค่า PE ใหม่ก่อนการฝังเพื่อให้ได้รับผลลัพธ์ PSNR สูงสุดหลังการฝัง แนวคิดของการเรียงลำดับค่า PE (ในตำแหน่งพิกเซลในเซลนั้น) ถ้าสามารถฝังข้อมูลลงใน PE ที่มีค่าต่ำ โดยใช้สมการที่ (3) $U_{i,j}$ ที่ได้รับจะมีผลลัพธ์การบิดเบือนที่ต่ำตามค่า PE นั้น สังเกตว่า ค่า PE จะไม่สามารถเรียงลำดับได้โดยตรงอันเนื่องมาจากค่า PE จะเปลี่ยนหลังจากการฝัง ซึ่งส่งผลให้การกู้คืนกลับของข้อมูลไม่สามารถทำได้ ดังนั้น ค่าความแปรปรวน ($\mu'_{i,j}$) จากพิกเซลข้างเคียงสี่ตำแหน่งจึงถูกพิจารณานำมาใช้แทนสามารถคำนวณได้ดังนี้

$$\mu'_{i,j} = \sum_{k=1}^{4} (\Delta v_k - \Delta \bar{v}_k)^2 \qquad (12)$$

โดย $\Delta v_1 = \left| v_{i,j-1} - v_{i-1,j} \right|$, $\Delta v_2 = \left| v_{i-1,j} - v_{i,j+1} \right|$, $\Delta v_3 = \left| v_{i,j+1} - v_{i+1,j} \right|$, $\Delta v_4 = \left| v_{i+1,j} - v_{i,j-1} \right|$, $\Delta \bar{v}_k = \left(\Delta v_1 + \Delta v_2 + \Delta v_3 + \Delta v_4 \right) / 4$ ข้อจำกัดสำหรับเทคนิคนี้คือ การฝังข้อมูลในปริมาณที่สูง ผลลัพธ์ค่า PSNR ที่ได้รับจะต่ำเนื่องจากค่า PE สูงๆ ถูกใช้ในการฝัง

2.1.4 Double Modification Testing (DMT) [13] เป็นหนึ่งเทคนิคที่ใช้สำหรับการตรวจสอบสถานะของพิกเซลที่ก่อให้เกิดปัญหา Underflow และ Overflow ก่อนการฝัง เทนนิคนี้สามารถแยกออกได้เป็น 7 กรณีตามความเป็นไปได้ดังนี้

- EE (Expand-Expand) เป็นเซตของพิกเซลที่สามารถขยายค่าได้สองครั้ง โดยไม่เกิดปัญหา Underflow และ Overflow
- ES (Expand-Shift) เป็นเซตที่สามารถขยายได้ในครั้งแรก และสามารถเลื่อนค่าของพิกเซลได้ในครั้งที่สอง โดยไม่เกิดปัญหา Underflow และ Overflow
- SS (Shift-Shift) เป็นเซตของพิกเซลที่สามารถเลื่อนค่าได้สองครั้ง โดยไม่เกิดปัญหา Underflow และ Overflow
- E (Expand) เป็นเซตของพิกเซลที่สามารถขยายค่าได้เพียงครั้งเดียว หากทำการแก้ไขพิกเซลครั้งที่สองจะทำให้เกิดปัญหา Underflow และ Overflow
- S (Shift) เป็นเซตของพิกเซลที่สามารถเลื่อนค่าได้เพียงครั้งเดียว หากทำการเลื่อนค่าของพิกเซลครั้งที่สองจะทำให้เกิดปัญหา Underflow และ Overflow
- NE (None-Expand) เป็นเซตของพิกเซลที่ไม่สามารถขยายค่าของพิกเซลได้
- NS (None-Shift) เป็นเซตของพิกเซลที่ไม่สามารถเลื่อนค่าของพิกเซลได้

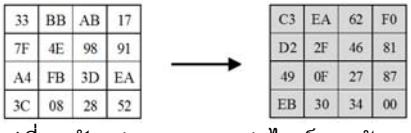
โดยที่เซตของ EE และ ES ค่า PE ที่อยู่ในสองเซตดังกล่าวนี้จะถูกเก็บไว้สำหรับการฝังบิตข้อมูล พิกเซลที่อยู่ในเซต SS จะถูกเลื่อน พิกเซลในเซต E และ S จะต้องกำหนดพื้นที่แมป (Location map) เป็น “0” และเซตของ NE, NS กำหนดพื้นที่แมปเป็น “1” เพื่อให้สามารถกู้คืนข้อมูลกลับได้ รายละเอียดของเทคนิคนี้สามารถอ่านเพิ่มเติมได้ที่ [13] เป็นที่กล่าวขานว่าการตรวจสอบสถานะโดยใช้ DMT ช่วยให้ขนาดของพื้นที่แมปมีขนาดเล็ก หรือ ในบางภาพไม่พบแมป ดังนั้นเครื่องมือในการบีบอัดไม่จำเป็นต้องใช้สำหรับการลดขนาดของพื้นที่แมป

2.2 AES Algorithm [12]

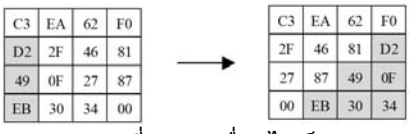
กระบวนการเข้ารหัสแบบ AES เป็นแบบ Symmetric โดยแบ่งเป็นบล็อกขนาด 4×4 และสามารถเลือกใช้กุญแจได้ 3 ขนาดได้แก่ 128, 192 หรือ 256 บิต ตามความเหมาะสม จำนวนรอบของการทำงานขึ้นอยู่กับขนาดของกุญแจ เช่น 10, 12, 14 ตามลำดับ การทำงานของ AES แบ่งเป็น 2 ส่วนคือ การเข้ารหัส/ถอดรหัส และการขยายกุญแจ (Key Expansion) โดยเริ่มต้นจาก AddRoundKey ตามด้วย SubBytes, ShiftRows, MixColumns และ AddRoundKey สำหรับรอบสุดท้ายจะไม่มีการทำงานของ MixColumns โดยกุญแจของแต่ละรอบในการทำงานจะได้มาจากการขยายกุญแจในขั้นตอนแรก

2.2.1 SubBytes เป็นการแทนค่าไบต์ของข้อมูลทั้งหมดด้วยเลขฐาน 16 จากตาราง S-Box ดังตารางที่ 1 โดยแสดงตัวอย่างไว้ดังรูปที่ 3

2.2.2 ShiftRows แถวบนสุดจะไม่ถูกเลื่อน แถวที่ 2 ถูกเลื่อนไปทางซ้าย 1 ตำแหน่ง แถวที่ 3 ถูกเลื่อนไป 2 ตำแหน่ง และแถวที่ 4 จะถูกเลื่อนไป 3 ตำแหน่งแสดงได้ดังรูปที่ 4



รูปที่ 3 ตัวอย่างการแทนค่าไบต์ของข้อมูล



รูปที่ 4 การเลื่อนไบต์

ตารางที่ 1 การแทนค่าสำหรับข้อมูลไบต์ในรูปแบบเลขฐานสิบหก (S-Box) [6]

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	26	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

2.2.3 MixColumns เป็นการนำแต่ละหลักในบล็อกมาคูณกับเมทริกซ์ค่าคงที่ดังรูปที่ 5

$$\begin{bmatrix} C3 \\ 2F \\ 27 \\ 00 \end{bmatrix} \times \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} = \begin{bmatrix} CB \\ F4 \\ A2 \\ 56 \end{bmatrix}$$

รูปที่ 5 ตัวอย่างขั้นตอนการ MixColumns

2.2.4 AddRoundKey เป็นการนำบล็อกข้อมูลเอ็กซ์คลูซีฟ-ออร์กับ RoundKey ที่สร้างจาก Key Expasion แสดงดังรูปที่ 6

01	38	80	02	⊕	32	83	2B	15	=	33	BB	AB	17
01	01	38	80		7E	4F	A0	11		7F	4E	98	91
02	01	00	00		A6	FA	3D	EA		A4	FB	3D	EA
00	00	00	00		3C	08	28	52		3C	08	28	52

รูปที่ 6 ตัวอย่างการเอ็กซ์คลูซีฟ-ออร์ระหว่างข้อมูลกับกุญแจ

2.2.5 Key Expansion ซึ่งประกอบไปด้วย 2 กระบวนการย่อย คือ กระบวนการย่อย Rotword เป็นการนำหลักที่ 4 ของกุญแจทำการเลื่อนไบต์ แต่ละค่าเป็นวงกลมไปทางซ้าย 1 ไบต์ แสดงดังรูปที่ 7 ต่อมากระบวนการย่อย Subword จะทำการแทนที่ไบต์ข้อมูลโดยอ้างอิงจาก S-box นำผลลัพธ์ที่ได้จากกระบวนการข้างต้นนำมาเอ็กซ์คลูซีฟ-ออร์กับหลักที่ 1 ของกุญแจในขั้นตอนที่ 1 และเอ็กซ์คลูซีฟ-ออร์กับค่าคงที่เรียกว่า Rcon [i] แสดงดังรูปที่ 8 และสำหรับลำดับของ i แสดงไว้ในตารางที่ 2 ขั้นตอนทั้งหมดอธิบายตามลำดับไว้ดังต่อไปนี้

ขั้นตอนแรก คือ การเลื่อนไบต์ดังรูปที่ 7

ขั้นตอนที่สอง คือ การแทนที่ข้อมูลจากตาราง SubBytes S-Box ในหลักขวาสุดของบล็อก

ขั้นตอนที่สาม คือ การนำข้อมูลหลักแรกของบล็อกเอ็กซ์คลูซีฟ-ออร์กับผลลัพธ์จากขั้นตอนที่สอง แสดงดังรูปที่ 8 และเอ็กซ์คลูซีฟ-ออร์กับ Rcon ดังตารางที่ 2 การคำนวณกุญแจหลักถัดไปเป็นการนำหลักที่ 2 ของบล็อกไปเอ็กซ์คลูซีฟ-ออร์กับผลลัพธ์ของขั้นตอนที่สามจะได้กุญแจในหลักถัดไป

32	83	2B	15	⟶	32	83	2B	11
7E	4F	A0	11		7E	4F	A0	EA
A6	FA	3D	EA		A6	FA	3D	52
3C	08	28	52		3C	08	28	15

รูปที่ 7 ตัวอย่างการเลื่อนข้อมูลในขั้นตอน Key Expansion

32	⊕	82	⊕	01	=	B1
7E		87		00		F9
A6		00		00		A6
3C		59		00		65

รูปที่ 8 ตัวอย่างการคำนวณหาผลลัพธ์ของกุญแจ

ตารางที่ 2 ค่าคงที่สำหรับกุญแจของแต่ละรอบในการเข้ารหัส (Round Constant: RCon)

01	02	04	08	10	20	40	80	1B	36
00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00

3. กระบวนการเข้ารหัส และถอดรหัส

ในส่วนนี้ได้อธิบายขั้นตอนการประยุกต์ใช้ AES [12] ร่วมกับขั้นตอนวิธี RDH [11] รายละเอียด และแนวความคิดอธิบายการทำงานดังรูปที่ 9 ซึ่งแสดงหนึ่งบล็อกไดอะแกรมซึ่งเป็นตัวแทนการเข้ารหัส และการถอดรหัส ก่อนการฝังข้อมูลทุกพิกเซลควรถูกแบ่งออกเป็นสองเซต ได้แก่ ครอสเซต และดอทเซต ขนาดของข้อมูลที่ต้องการฝัง สำหรับกระบวนการเข้ารหัสครอสเซต และดอทเซต คือ P_{Cross} และ P_{Dot} ตามลำดับ สำหรับการกู้คืนข้อมูล ค่าขีดแบ่ง T_{nCross}

และ T_{pCross} และขนาดของข้อมูลที่ต้องการฝัง $|P_{Cross}|$ หรือ (สำหรับกระบวนการฝังครอส) $|P_{Dot}|$ (สำหรับกระบวนกรฝังดอท) ควรถูกส่งให้ผู้รับ (หรือ ผู้ถอดรหัส) ค่า LSB ของ 36 ตำแหน่งแรกในค่าความผิดพลาดจากการทำนายจะถูกแทนที่ด้วยค่าขีดแบ่ง T_{nCross} (8 บิต) และ T_{pCross} (8 บิต) ขนาดของข้อมูลที่ต้องการฝัง $|P_{Dot}|$ (20 บิต) หรือ $|P_{Cross}|$ (20 บิต) ค่าต้นฉบับ LSB ของทั้ง 36 ตำแหน่งจะถูกเก็บลงในเซตเรียกว่า S_{LSB} เซต และถูกรวมเป็นส่วนหนึ่งกับขนาดของข้อมูลที่ต้องการฝัง ค่าความคลาดผิดพลาดจากการทำนายทั้ง 36 ตำแหน่งเหล่านี้ถูกแยกออก จำนวนของบิตที่ถูกเก็บเป็นไปตามการประยุกต์ใช้งาน ซึ่ง 36 ตำแหน่งเหล่านี้จะถูกเข้ารหัสแบบ AES ด้วยขนาดข้อมูล 128 บิต และใช้กุญแจขนาด 128 บิต โดยจำนวนรอบของการทำงาน (n) ขึ้นอยู่กับขนาดของกุญแจ Header ของ RDH กระบวนการทั้งหมดมีลำดับขั้นตอนดังนี้

3.1 การเข้ารหัส

(1) แบ่งพิกเซลทั้งหมดของภาพต้นฉบับออกเป็น 2 เซตคือ ครอสเซต และดอทเซต

(2) คำนวณค่าทำนาย (u') จากสมการ (1), ค่าความผิดพลาดจากการทำนาย (d) จากสมการ (2) และค่าความแปรปรวน (μ') จากสมการ (12)

(3) ทำการเรียงลำดับค่า PE จากน้อยไปมากตามขนาดของค่าความแปรปรวน (μ)

(4) หาค่าขีดแบ่ง $[T_n;T_p]$ ที่เหมาะสมสำหรับขนาดของข้อมูลที่ต้องการฝัง

(5) ตรวจสอบทุกพิกเซลที่สามารถฝังข้อมูลได้และไม่สามารถฝังข้อมูลได้ (Location map) โดยใช้ DMT

(6) ฝังข้อมูล และพื้นที่แมปโดยใช้เทคนิค HS

(7) แยก Header 128 บิต สำหรับใส่ Payload, T_n , T_p ของครอสเซต

(8) สุ่มกุญแจขนาด 128 บิต และนำกุญแจคำนวณ Key Expansion สำหรับ 10 รอบการทำงาน

(9) นำ Header เอ็กซ์คลูซีฟ-ออร์กับกุญแจตัวแรก(AddRoundKey) ดังรูปที่ 6

(10) นำผลลัพธ์จากขั้นตอนที่ (9) คำนวณ SubBytes, ShiftRows, MixColumns และ AddRoundKey กับกุญแจตัวถัดไป ทำซ้ำถึงรอบที่ 9 และรอบสุดท้ายไม่มีการคำนวณ MixColumns

(11) แทนค่า Cipher text ที่ได้จากการเข้ารหัสลงใน Header เป็นอันเสร็จสิ้นกระบวนการเข้ารหัส

3.2 การถอดรหัส

(1) แยก Cipher text (Header) ที่ถูกเข้ารหัสลับออกจากภาพ

(2) นำ Cipher text เอ็กซ์คลูซีฟ-ออร์กับกุญแจตัวสุดท้าย

(3) นำผลลัพธ์จากขั้นตอนที่ (2) คำนวณ Inv. ShifRows, Inv. SubByte, Inv. AddRoundKey, Inv. MixColumns ทำซ้ำจนถึงรอบที่ 9 และรอบสุดท้ายไม่ต้องคำนวณ Inv.MixColumns

(4) เมื่อได้ Header ต้นฉบับแบ่งพิกเซลออกเป็น 2 เซต

(5) คำนวณค่าทำนาย (u') ค่าความผิดพลาดจากการทำนาย (d) ค่าความแปรปรวน (μ)

(6) เรียงลำดับข้อมูลตามขนาดของค่าความแปรปรวนจากน้อยไปมาก (μ)

(7) กู้คืนข้อมูลภาพต้นฉบับ และข้อมูลที่ฝังตามขั้นตอนวิธี [11]

(8) ใส่ LSB ของ Header คืนลงใน 128 บิตแรก เป็นอันเสร็จสิ้นกระบวนการ หมายเหตุ สำหรับการกู้คืนข้อมูลจะเริ่มถอดจากเซตท้ายสุดของการฝัง

4. ผลการทดลอง

ภาพทางการแพทย์ที่ได้รับจากเครื่องมือที่แตกต่างกันถูกใช้สำหรับการทดสอบประกอบด้วยภาพ MRI จำนวน 25 รูป ภาพ Ultrasound จำนวน 10 รูป และ ภาพ X-ray จำนวน 5 รูป โดยภาพเหล่านี้มีขนาด 512×512 พิกเซล แสดงไว้ดังรูปที่ 10 ตัวอย่างข้อมูลที่ใช้ฝังสำหรับการทดสอบประกอบด้วย ภาพลายนิ้วมือจำนวน 10 รูป ภาพบัตรประชาชนจำนวน 28 รูป และภาพหนังสือเดินทางจำนวน 4 รูป ซึ่งแบ่งออกเป็น 4 ขนาด ได้แก่ 100×100 พิกเซล, 200×200 พิกเซล, 300×300 พิกเซล และ 400×400 พิกเซล ตามลำดับ

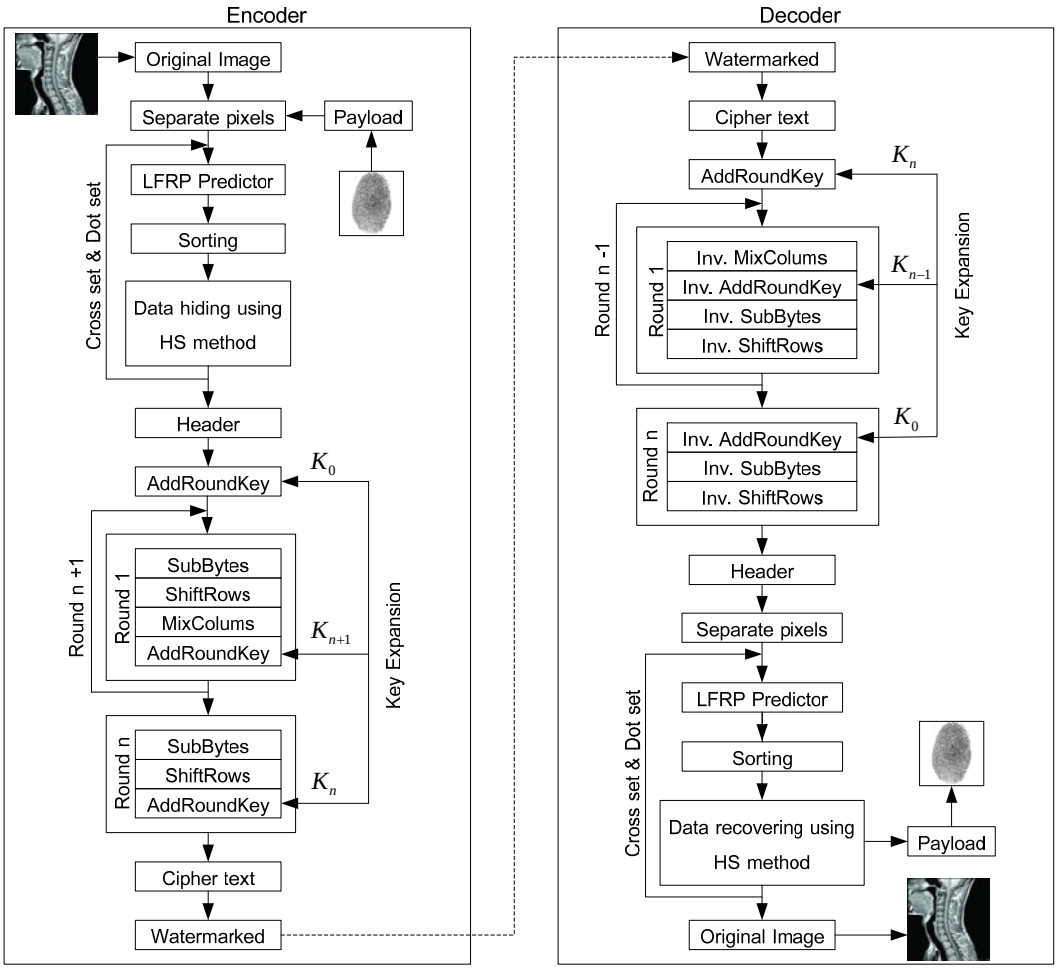
แสดงไว้ดังรูปที่ 11-13 ประสิทธิภาพของการอำพรางข้อมูล กระบวนการของเราใช้ค่า PSNR (Peak Signal to Noise Ratio) ผลการทดลองของวิธีการที่นำเสนอแสดงดังตารางที่ 3-5 และรูปที่ 14-16 สังเกตว่าในทุกขนาดของความจุสำหรับการฝัง 0.04 - 0.61 bpp ภาพ MRI 13 และ MRI 15 มีค่า PSNR สูงกว่าทุกภาพ อันเนื่องมาจากการทำนายที่มีประสิทธิภาพของ LFRP มีค่า PE = 0 จำนวนสูงถึง 34,211 ตำแหน่ง ข้อเท็จจริงพบว่า เมื่อค่า PE เป็นศูนย์จำนวนมาก การฝังข้อมูลโดยใช้เทคนิค HS ความบิดเบือนจะลดลงตามค่า PE ในทางตรงกันข้าม ภาพ MRI 19 ได้รับค่า PSNR ต่ำสุดเนื่องจากภาพมีความแปรปรวนของพิกเซลสูงเมื่อเทียบกับภาพอื่น ๆ สังเกตว่า ค่าเฉดสีภายในภาพส่วนใหญ่มีค่าเป็น 0 และ 255 ในพิกเซลที่ติดกันมีค่าเปลี่ยนแปลงอย่างฉับพลัน ยกตัวอย่างเช่น จาก 0 เป็น 255 การเปลี่ยนแปลงดังกล่าวส่งผลกระทบโดยตรงต่อการทำนาย ซึ่งฮิสโตรแกรมของค่า PE = 0 จำนวนต่ำกว่าภาพอื่นๆ ซึ่งได้รับเพียง 28,113 ตำแหน่ง ในภาพที่ไม่สามารถฝังข้อมูลขนาดใหญ่ได้ เช่น ภาพ MRI 1, MRI 19, Ultrasound 5, Ultrasound 10, Ultrasound 11, Ultrasound 12 และ X-ray 14 เป็นผลมาจากฟังก์ชันการทำนายที่ทำนายค่า PE ผิดพลาดเป็นจำนวนมาก เมื่อฝังข้อมูลลงในค่า PE เหล่านี้ มักก่อให้เกิดปัญหา Overflow ตามมา ซึ่งปัญหาดังกล่าวส่งผลให้พื้นที่แมปมีขนาดใหญ่ เป็นที่รู้กันว่า ขนาดของพื้นที่แมปจะถูกฝังเป็นส่วนหนึ่งของข้อมูลที่ต้องการฝัง หนึ่งตัวอย่างสำหรับภาพที่สามารถฝังข้อมูลขนาดใหญ่ได้ แสดงดังรูปที่ 17 ภาพ MRI 1 โดยฝังข้อมูลลายนิ้วมือขนาด 400x400 พิกเซล (0.61 bpp) สังเกตว่าภาพหลังการฝังมีความบิดเบือนต่ำ และไม่สามารถมองเห็นได้ด้วยตาเปล่า การอำพรางข้อมูลทำได้อย่างมีประสิทธิภาพ อย่างไรก็ตามสำหรับ การป้องกันการเข้าถึงข้อมูลในส่วนของ Header ก่อนการฝังจะถูกเข้ารหัสลับโดยประยุกต์ใช้ขั้นตอนวิธี AES ซึ่งผลลัพธ์ไม่กระทบต่อการบิดเบือนในพื้นที่ส่วนใหญ่ของภาพ

ตารางที่ 3 ผลลัพธ์ PSNR vs. Payload ของภาพลายนิ้วมือ

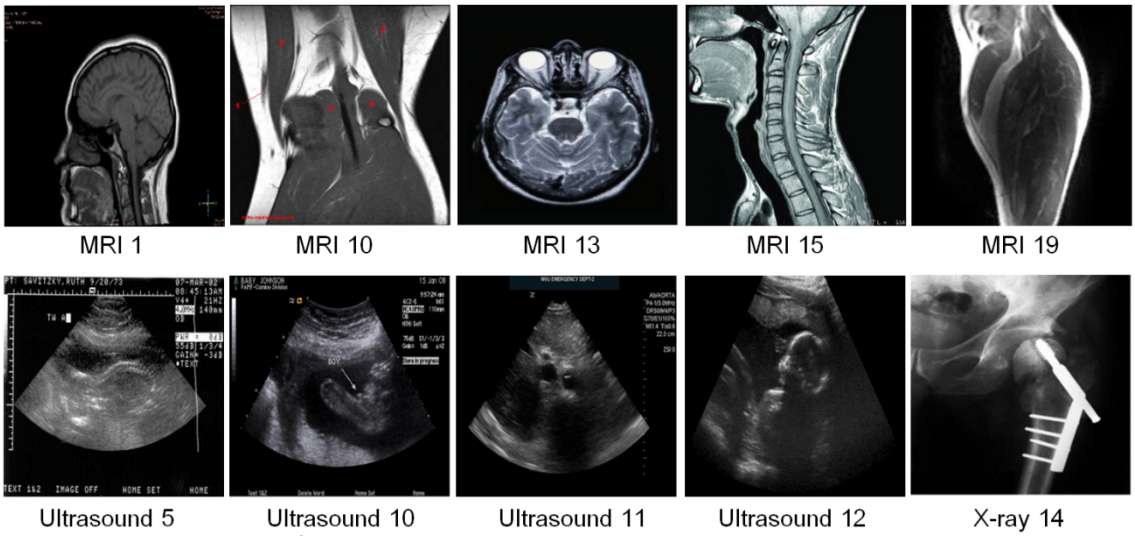
Payload (bpp)	PSNR (dB)			
	0.04	0.15	0.34	0.61
MRI 1	60.90	54.17	-	-
MRI 10	62.25	56.61	52.98	48.63
MRI 13	64.87	58.80	55.63	52.35
MRI 15	65.07	59.11	55.51	45.73
MRI 19	60.65	55.08	52.21	-
Ultrasound 5	64.67	58.73	54.61	-
Ultrasound 10	60.89	53.46	-	-
Ultrasound 11	60.89	53.89	-	-
Ultrasound 12	60.89	54.22	-	-
X-ray 14	64.05	57.90	54.63	52.15

ตารางที่ 4 ผลลัพธ์ PSNR vs. Payload ของภาพบัตรประชาชน

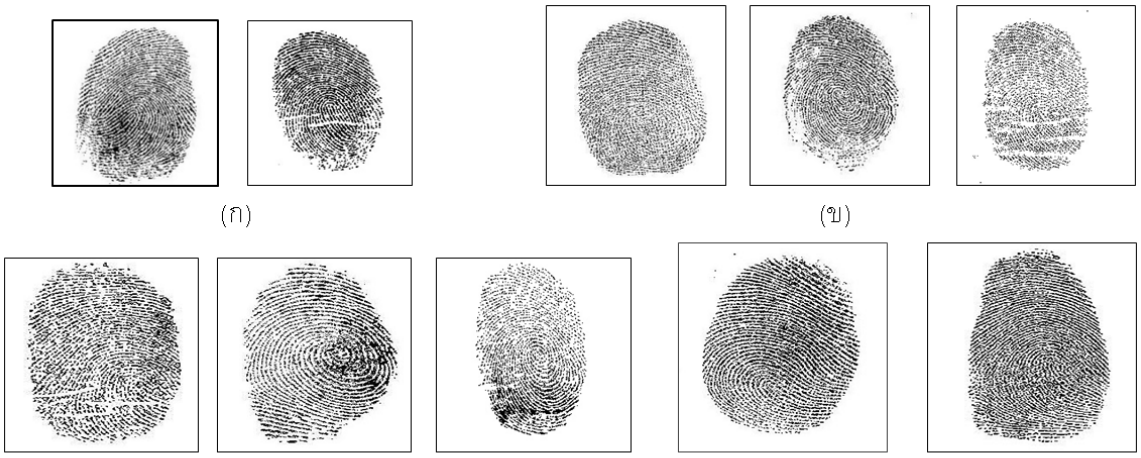
Payload (bpp)	PSNR (dB)			
	0.04	0.15	0.34	0.61
MRI 1	60.79	53.61	-	-
MRI 10	62.24	56.63	53.01	48.66
MRI 13	64.76	58.82	55.68	48.66
MRI 15	64.95	59.23	55.64	45.77
MRI 19	60.78	54.75	52.24	-
Ultrasound 5	64.59	58.80	54.73	-
Ultrasound 10	61.04	51.89	-	-
Ultrasound 11	61.05	53.00	-	-
Ultrasound 12	61.04	53.78	-	-
X-ray 14	64.00	57.87	54.71	52.30



รูปที่ 9 บล็อกไดอะแกรมกระบวนการเข้ารหัส และถอดรหัส



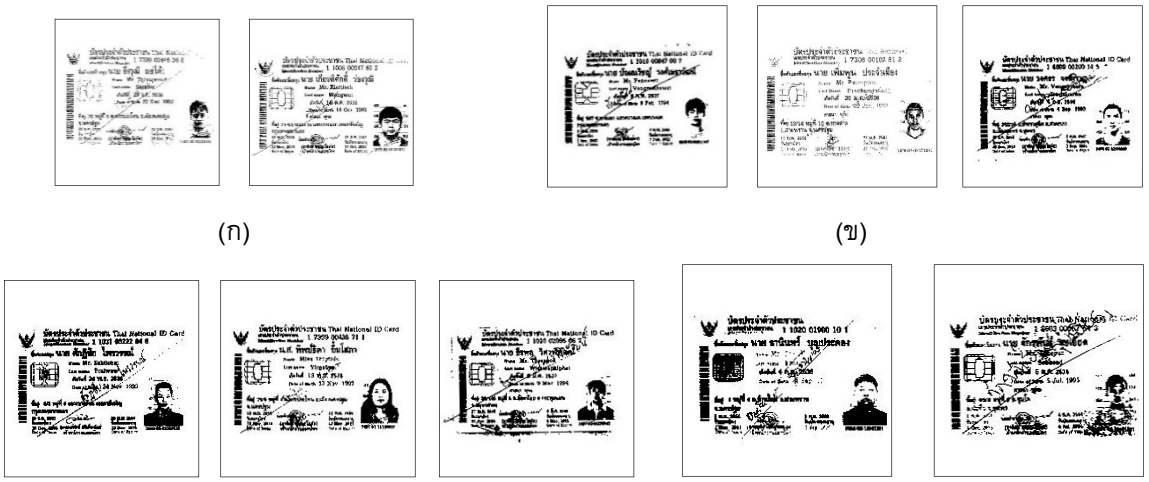
รูปที่ 10 ตัวอย่างภาพต้นฉบับระดับเทาขนาด 512×512 พิกเซล



(ก) (ข)

(ค) (ง)

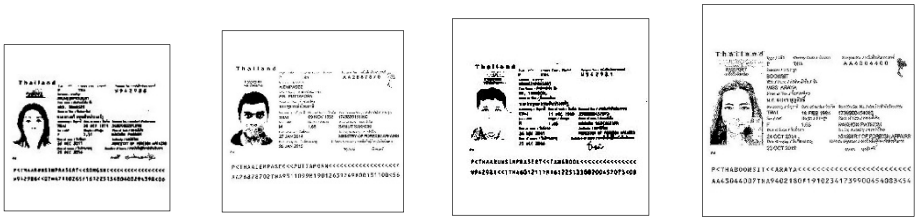
รูปที่ 11 ตัวอย่างภาพลายนิ้วมือ (ก) ภาพลายนิ้วมือขนาด 100×100 พิกเซล (ข) ภาพลายนิ้วมือขนาด 200×200 พิกเซล (ค) ภาพลายนิ้วมือขนาด 300×300 พิกเซล (ง) ภาพลายนิ้วมือขนาด 400×400 พิกเซล



(ก) (ข)

(ค) (ง)

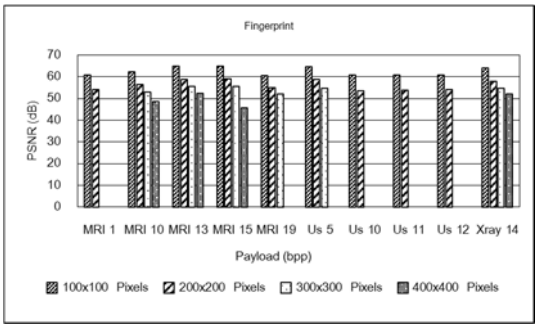
รูปที่ 12 ตัวอย่างภาพบัตรประชาชน (ก) ภาพบัตรประชาชนขนาด 100×100 พิกเซล (ข) ภาพบัตรประชาชนขนาด 200×200 พิกเซล (ค) ภาพบัตรประชาชนขนาด 300×300 พิกเซล (ง) ภาพบัตรประชาชนขนาด 400×400 พิกเซล



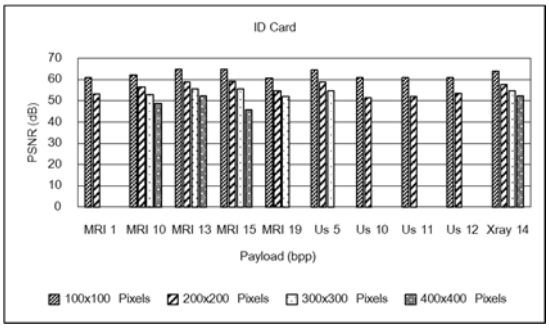
รูปที่ 13 ตัวอย่างภาพหนังสือเดินทาง (ก) ภาพหนังสือเดินทางขนาด 100×100 พิกเซล (ข) ภาพหนังสือเดินทางขนาด 200×200 พิกเซล (ค) ภาพหนังสือเดินทางขนาด 300×300 พิกเซล (ง) ภาพหนังสือเดินทางขนาด 400×400 พิกเซล

ตารางที่ 5 ผลลัพธ์ PSNR vs. Payload ของภาพหนังสือเดินทาง

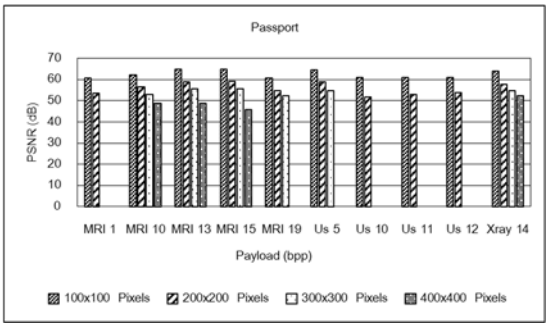
Payload (bpp)	PSNR (dB)			
	0.04	0.15	0.34	0.61
MRI 1	60.89	53.37	-	-
MRI 10	62.23	56.64	53.00	48.65
MRI 13	64.81	59.06	55.64	52.47
MRI 15	65.01	59.18	55.57	45.76
MRI 19	60.64	54.71	52.21	-
Ultrasound 5	64.63	58.79	54.66	-
Ultrasound 10	60.89	51.52	-	-
Ultrasound 11	60.89	51.98	-	-
Ultrasound 12	60.88	53.61	-	-
X-ray 14	64.00	57.80	54.66	52.28



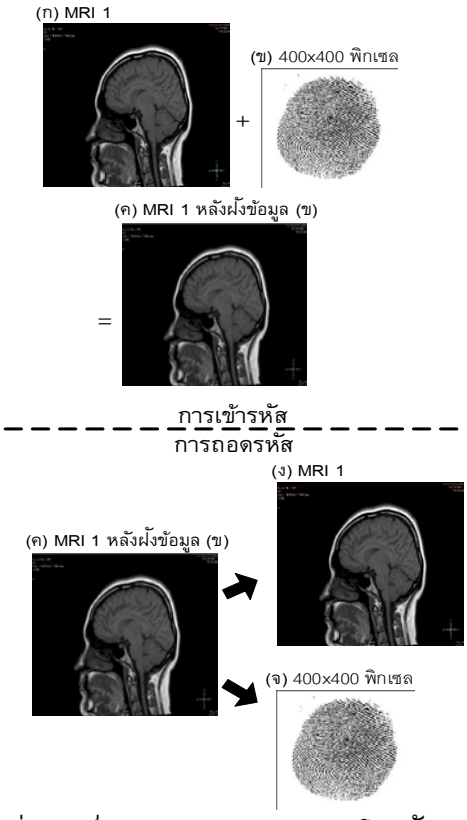
รูปที่ 14 กราฟแท่งเปรียบเทียบค่า PSNR ของแต่ละภาพ แต่ละขนาดการฝังสำหรับภาพลายนิ้วมือ



รูปที่ 15 กราฟแท่งเปรียบเทียบค่า PSNR ของแต่ละภาพ แต่ละขนาดการฝังสำหรับภาพบัตรประชาชน



รูปที่ 16 กราฟแท่งเปรียบเทียบค่า PSNR ของแต่ละภาพ แต่ละขนาดการฝังสำหรับภาพหนังสือเดินทาง



รูปที่ 17 หนึ่งตัวอย่างของการเข้ารหัสโดยฝังข้อมูลลายนิ้วมือขนาด 400x400 พิกเซลลงในภาพ MRI 1 และการกู้คืนข้อมูลกลับ

5. สรุป

ในงานวิจัยนี้นำเสนอการรวมกันของสองวิทยาการ สำหรับขั้นตอนวิธีการอำพรางข้อมูล และขั้นตอนวิธีการเข้ารหัสลับ สองวิทยาการถูกร่วมกันอย่างมีนัยสำคัญเพื่อป้องกันการเข้าถึงข้อมูลโดยผู้ที่ไม่ได้รับอนุญาต หนึ่งขั้นตอนวิธีของการฝังข้อมูลที่สามารถกู้คืนกลับได้ถูกใช้ในการอำพราง AES ถูกใช้ในการเข้ารหัสลับ ผลลัพธ์สำหรับงานนี้แสดงให้เห็นถึงความบิดเบือนที่ต่ำสำหรับประสิทธิภาพในการอำพรางและความปลอดภัยที่สูงขึ้นสำหรับการเข้าถึงข้อมูล

6. กิตติกรรมประกาศ

ขอขอบคุณ BOSIS Lab สำหรับฐานข้อมูลภาพทางการแพทย์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง และขอขอบคุณสถาบันวิจัยและพัฒนา มหาวิทยาลัยเทคโนโลยีราชมงคลรัตนโกสินทร์สำหรับทุนสนับสนุนการทำวิจัย

7. เอกสารอ้างอิง

[1] J. Tian, “Reversible watermarking using a difference expansion,” *IEEE Transactions on Circuit System and Video Technology.*, vol. 13, pp. 890-896, 2003.

[2] A. M. Alattar, “Reversible Watermark Using the Difference Expansion of a Generalized Integer Transform,” *IEEE Transactions on Image Processing.*, vol. 13, pp. 1147-1156, 2004.

[3] Y. Hu, H.-K. Lee, K. Chen and J. Li, “Difference Expansion Based Reversible Data Hiding Using Two Embedding Directions,” *IEEE Transactions on Multimedia*., vol. 10, pp. 1500-1512, 2008.

[4] L. H. J. Kamstra and A. M. Heijmans, “Reversible data embedding into images using wavelet and sorting,” *IEEE Transactions on Image Processing.*, vol. 14, pp. 2082-2090, 2005.

[5] M. Afsharizadeh and M. Mohammadi, “A Reversible Watermarking Prediction Based Scheme using a New Sorting Technique,” *in the Proc. of Information Security and Cryptology.*, pp.1-5 2013.

[6] D. M. Thodi and J. J. Rodriguez, “Reversible Watermarking by Prediction-Error Expansion,” *in the Proc. of IEEE Southwest Symposium on Image Analysis and Interpretation.*, pp. 21-25, 2004.

[7] D. M. Thodi and J. J. Rodriquez, “Expansion embedding techniques for reversible watermarking,” *IEEE Transactions on Image Processing.*, vol. 16, pp. 721-730, 2007.

[8] R. Liu, R. Ni and Y. Zhao, “A reversible data hiding based on adaptive prediction technique and histogram shifting,” *in the Proc. of Asia-Pacific Signal and Information Processing Association.*, pp.1-6, 2014.

[9] N.-K. Chen, C.-Y. Su, C.-Y. Shih and Y.-T. Chen, “Reversible Watermarking for Medical Images Using Histogram Shifting with Location Map Reduction,” *in the Proc. of IEEE International Conference on Industrial Technology.*, pp. 792-797, 2016.

[10] V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Y.-Q. Shi, “Reversible watermarking algorithm using sorting and prediction,” *IEEE Transactions on Circuit System and Video Technology.*, vol. 19, pp. 989-999, July. 2009.

[11] C. Panyindee and C. Pintavirooj, “Reversible Data Hiding Scheme Using Optimal Weight Predictor Based on DMT in Medical Imaging,”*in the Proc. of Biomedical Engineering International Conference.*, pp. 1-4, 2014.

[12] J. Daemen and V. Rijmen, “The Design of Rijndael AES-The Advanced Encryption Standard,” Springer-Verlag, New York, pp.1-238, 2002.

[13] C. Panyindee and C. Pintavirooj, “Optiml Gaussian Weight Predictor and Sorting Using Genetic Algorithm for Reversible Watermarking Based on PEE and HS,” *IEICE Transactions on Information and Systems.*, vol. E99-D, pp. 2306-2319, 2016.

[14] D. Coppersmith, “The Data Encryption Standard (DES) and its Strength Against Attacks,” *IBM Journal of Research & Development.*, vol. 38, pp.243-250, 1994.

[15] A. Mousa and A. Hamad, “Evaluation of the RC4 Algorithm for Data Encryption,” *International Journal of Computer Science and Applications.*, vol. 3, pp.44-56, 2006.

[16] ดร.วรากร ศรีเชวงทรัพย์. “ระบบรหัสลับที่เพิ่มความปลอดภัยให้กับข้อมูล,” *วารสารปัญญาภิวัฒน์*. (ปีที่ 2 ฉบับที่ 2): เลขหน้า 93-101, 2011.