

การบีบอัดภาพของกล้องจุลทรรศน์อิเล็กตรอน

โดยใช้การแปลงเวฟเลตแบบดิสครีท

EM Image Compression Using the Discrete Wavelet Transform

ณัฐนันท์ ทัดพิทักษ์กุล¹ กิตติ อรรถกิจมงคล² สราวุฒิ สุจิตจร³

¹นักศึกษาระดับบัณฑิตศึกษา ²อาจารย์ ³รองศาสตราจารย์

สาขาวิชาวิศวกรรมไฟฟ้า สำนักวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

ABSTRACT – The image from the Electron Microscope (EM) is effectively used to analyze the fine details of the object's surface. When the discrete wavelet transform is applied to the image, the plain surface of the object will be in the low-frequency subband and the edge will be in the high-frequency subbands. Thus, compression algorithm for the EM image must take every subband of the wavelet coefficients into account. A powerful image compression algorithm we consider is the Set Partitioning in Hierarchical Tree (SPIHT). This coding scheme exploits the self-similarity of the wavelet coefficients across different scales and searches for the high magnitude coefficients in every subband. In this paper, we propose an improvement of the SPIHT algorithm. Several wavelets are then applied for comparison in order to find the best wavelet basis for the EM image with the improved algorithm.

KEY WORD – Image compression, SPIHT, Wavelet

บทคัดย่อ – ภาพของกล้องจุลทรรศน์อิเล็กตรอน (Electron Microscope, EM) เป็นภาพที่ใช้วิเคราะห์รายละเอียดของโครงสร้างระดับไมโครบนพื้นวัตถุ ทำให้ภาพ EM เป็นภาพที่มีรายละเอียดมาก เมื่อนำมาแปลงด้วยวิธีเวฟเลตแบบดิสครีท ข้อมูลที่เป็นพื้นผิววัตถุจะอยู่ที่สัมประสิทธิ์ที่แบนด์ย่อยความถี่ต่ำ และข้อมูลที่เป็นขอบหรือลายเส้นของวัตถุจะอยู่ที่สัมประสิทธิ์ที่แบนด์ย่อยความถี่สูง ดังนั้นการบีบอัดข้อมูลภาพ EM ด้วยการแปลงเวฟเลตจะต้องให้ความสำคัญกับสัมประสิทธิ์ทุกแบนด์ย่อย อัลกอริทึม Set Partitioning in Hierarchical Tree (SPIHT) เป็นอัลกอริทึมหนึ่งที่เหมาะสม เนื่องจากอัลกอริทึมนี้เข้ารหัสโดยให้ความสำคัญกับขนาดของสัมประสิทธิ์และไม่สนใจว่าสัมประสิทธิ์ตัวนั้นจะอยู่ในระดับแบนด์ย่อยใด ในบทความนี้เสนอวิธีการพัฒนาอัลกอริทึม SPIHT ให้สามารถบีบอัดข้อมูลได้เพิ่มขึ้น พร้อมทั้งหาเวฟเลตที่เหมาะสมกับอัลกอริทึมที่ทำการพัฒนา

คำสำคัญ – การบีบอัดข้อมูลภาพ, SPIHT, เวฟเลต

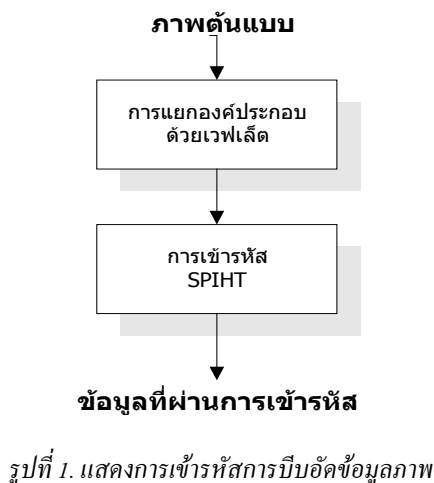
1. คำนำ

ปัจจุบันการศึกษาค้นคว้าสิ่งเร้นลับทางชีวภาพและกายภาพขนาดเล็ก ซึ่งไม่สามารถมองเห็นได้ด้วยตาเปล่าได้ เช่น การวิเคราะห์รายละเอียดของโครงสร้างระดับไมโครบนพื้นผิววัตถุ ไม่ว่าจะเป็นการกระจายของอนุภาค ความพรุน ความหนาแน่น หรือร่องรอยอันเกิดจากความผิดปกติของผิววัสดุ เป็นสิ่งจำเป็นอย่างยิ่งในการควบคุมคุณภาพของงานด้านโลหกรรมและวัสดุศาสตร์ และการผลิตประดิษฐ์กรรมทางไมโครอิเล็กทรอนิกส์ สิ่งหนึ่งที่ใช้ในการวิเคราะห์คือภาพไมโครกราฟที่ได้จากกล้อง EM จะมีอยู่ด้วยกัน 2 ประเภทคือ ภาพที่ได้จากกล้องโพลาไรซ์

(Polaroid) ภาพที่ได้นั้นมักไม่สามารถแยกความเปรียบต่าง (contrast) ของภาพ รวมถึงความแตกต่างของรูปลักษณะตามบริเวณต่างๆ ได้ชัดเจน ทำให้เกิดความคลาดเคลื่อนในการวิเคราะห์สูง [12] และภาพที่ได้จากการเปลี่ยนสัญญาณภาพจากระบบสแกนความเร็วต่ำของ EM ที่อยู่ในรูปสัญญาณอนาล็อก จะถูกแปลงเป็นสัญญาณเชิงตัวเลข และบันทึกเป็นไฟล์ข้อมูล ซึ่งภาพจากไฟล์ข้อมูลยังสามารถนำไปผ่านกระบวนการปรับแต่งรายละเอียดภาพด้วยวิธีการกรองสิ่งรบกวน และผ่านกระบวนการชดเชยให้มีความคมชัด และความเปรียบต่างของภาพดีขึ้น ทำให้มีความเชื่อมั่นในผลการวิเคราะห์ได้สูงขึ้น [12]

ภาพของ EM เป็นภาพที่มีรายละเอียดภาพสูง เมื่อนำมาแปลงเวฟเลตแบบดิสครีต ข้อมูลที่เป็นพื้นผิววัตถุจะอยู่ที่สัมประสิทธิ์ที่แบนด์ย่อยความถี่ต่ำ และข้อมูลที่เป็นขอบหรือลายเส้นของวัตถุจะอยู่ที่สัมประสิทธิ์ที่แบนด์ย่อยความถี่สูง ดังนั้นการบีบอัดข้อมูลภาพ EM ด้วยการแปลงเวฟเลตจะต้องให้ความสำคัญกับสัมประสิทธิ์ทุกแบนด์ย่อย อัลกอริทึม Embedded Zerotrees Wavelet (EZW) เป็นอัลกอริทึมหนึ่งที่เหมาะสมเนื่องจากอัลกอริทึมนี้เข้ารหัสโดยให้ความสำคัญกับขนาดของสัมประสิทธิ์และไม่สนใจว่าสัมประสิทธิ์ตัวนั้นจะอยู่ในระดับแบนด์ย่อยใด [8,9] และให้อัตราการบีบอัดข้อมูลดีกว่าการบีบอัดข้อมูลภาพด้วยวิธี Joint Photographic Group (JPEG) [9] และ EZW ยังไม่ทำให้เกิดบล็อกอาร์ตifacts (Block Artifacts) เหมือน JPEG [10,11] ได้มีการพัฒนา EZW ให้มีประสิทธิภาพเพิ่มมากขึ้นหลายวิธี แต่ในบทความนี้เลือกใช้ อัลกอริทึม Set Partitioning in Hierarchical Tree (SPIHT) ซึ่งให้อัตราการบีบอัดข้อมูลภาพที่ดีกว่า EZW [2,8] และเป็นอัลกอริทึมที่เร็วและมีประสิทธิภาพ [3] นอกจากนี้ในบทความนี้ยังทำการพัฒนาอัลกอริทึม SPIHT ให้มีอัตราการบีบอัดข้อมูลภาพที่ดีขึ้นด้วยการเพิ่ม List of Forbidden Coefficients (LFC) และเพิ่มเงื่อนไขการเข้ารหัสและการถอดรหัส พร้อมทั้งหาเวฟเลตแม่ที่เหมาะสมกับอัลกอริทึม SPIHT ที่ทำการพัฒนา

สำหรับการบีบอัดข้อมูลภาพในบทความนี้ ประกอบด้วยขั้นตอนการเข้ารหัส (Encode) และขั้นตอนการถอดรหัส (Decode) ซึ่งแต่ละขั้นตอนได้แสดงไว้ในรูปที่ 1 และรูปที่ 2 ตามลำดับ



ข้อมูลที่ผ่านการเข้ารหัส

การถอดรหัส SPIHT

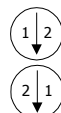
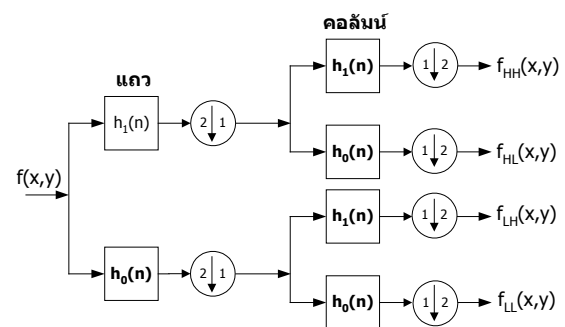
การสร้างกลับจากองค์ประกอบทางเวฟเลต

ภาพที่ผ่านการบีบอัดข้อมูล

รูปที่ 2. แสดงการถอดรหัสการบีบอัดข้อมูลภาพ

2. การแปลงเวฟเลตแบบดิสครีต (Discrete Wavelet Transform)

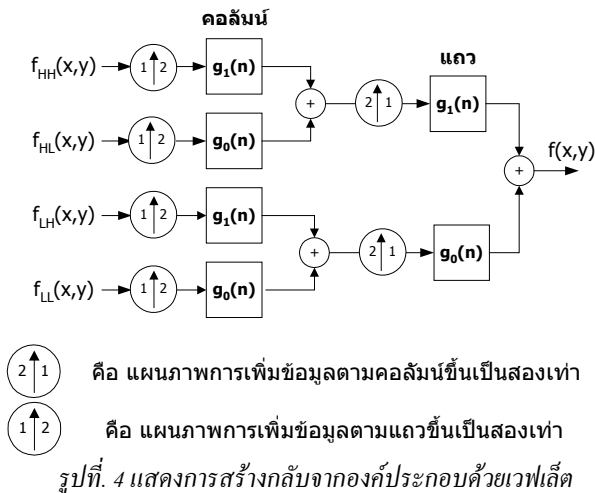
การแปลงเวฟเลตของข้อมูลภาพ เป็นการแปลงเวฟเลตแบบดิสครีตแบบ 2 มิติ [1,5,7] ใช้หลักการแยกองค์ประกอบเป็นแบนด์ย่อย (Subband Decomposition) โดยมีวิธีแยกองค์ประกอบด้วยเวฟเลต (Wavelet Decomposition) เป็นดังรูปที่ 3 กำหนดให้ $f(x,y)$ คือภาพต้นแบบ $f_{LL}(x,y)$, $f_{LH}(x,y)$, $f_{HL}(x,y)$ และ $f_{HH}(x,y)$ คือสัมประสิทธิ์เวฟเลต $h_0(n)$ และ $h_1(n)$ คือสัมประสิทธิ์ของเวฟเลตของการแยกองค์ประกอบ (ตัวฟิลเตอร์ที่กรองความถี่ต่ำและสูงตามลำดับ) และมีวิธีการสร้างกลับจากองค์ประกอบด้วยเวฟเลต (Wavelet Reconstruction) เป็นดังรูปที่ 4 กำหนดให้ $g_0(n)$ และ $g_1(n)$ คือสัมประสิทธิ์ของเวฟเลตของการสร้างกลับจากองค์ประกอบ (ตัวฟิลเตอร์ที่กรองความถี่ต่ำและสูงตามลำดับ)



คือ แผนภาพการลดข้อมูลตามแวลลงครึ่งหนึ่ง

คือ แผนภาพการลดข้อมูลตามคอลัมน์ลงครึ่งหนึ่ง

รูปที่ 3 แสดงการแยกองค์ประกอบด้วยเวฟเลต



3. การเข้ารหัส SPIHT [3,8]

ตำแหน่งสัมประสิทธิ์เวฟเลตในการเข้ารหัสมีความสัมพันธ์แบบ Spatial Orientation Trees โดยมีความสัมพันธ์เป็นดังรูปที่ 5 โดยปกติพลังงานของภาพ (Image Energy) ส่วนใหญ่จะอยู่ในส่วนที่มีความถี่ต่ำ ดังนั้นเราจะเคลื่อนที่จากแบนด์ย่อยที่สูงไปยังแบนด์ย่อยที่ต่ำกว่า นอกจากนี้จากการสังเกต spatial ของสัมประสิทธิ์มีความคล้ายกันระหว่างแบนด์ย่อยที่อยู่ในทิศทางเดียวกัน และคาดหวังว่าขนาดของสัมประสิทธิ์จะมีค่าลดลง โดยเริ่มจากแบนด์ย่อยสูงไปยังแบนด์ย่อยต่ำตามทิศทาง spatial ที่มีทิศทางเดียวกัน

การกำหนดตำแหน่ง (Coordinates) ของการเข้ารหัส

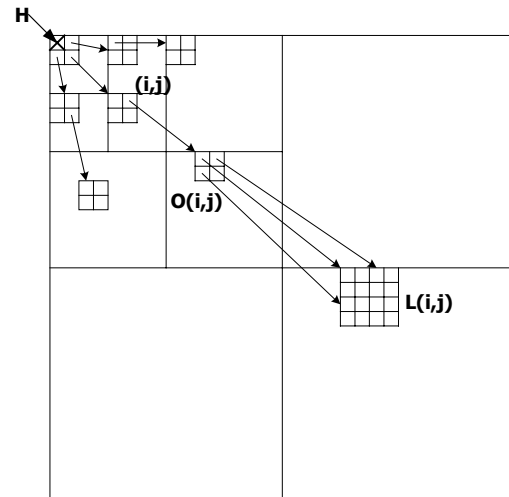
- (i,j) คือตำแหน่งของสัมประสิทธิ์เวฟเลตที่โหนด(node)
- $O(i,j)$ คือกลุ่มสัมประสิทธิ์เวฟเลตที่ตำแหน่งของการสืบทอด (offspring) ของโหนด และ 1 โหนดมีการสืบทอด 4 ตำแหน่ง
- (k,l) คือตำแหน่งของสัมประสิทธิ์เวฟเลตที่สืบทอดจากโหนดนั้น คือ $O(i,j)$ จะมี (k,l) อยู่ 4 ตำแหน่ง
- $L(i,j)$ คือกลุ่มสัมประสิทธิ์เวฟเลตที่ตำแหน่งของการสืบทอดของ $O(i,j)$ โดยมีการสืบทอดไปจนกว่าจะถึงแบนด์ย่อยต่ำสุด
- $D(i,j) = O(i,j) + L(i,j)$ คือตำแหน่งของการสืบทอดทั้งหมดของโหนด
- H คือตำแหน่งทั้งหมดที่อยู่ในแบนด์ย่อยสูงสุดที่เป็นต้นกำเนิด โหนดและแต่ละตำแหน่งจะมีตำแหน่งของการสืบทอด 3 ตำแหน่ง

ในการเข้ารหัสและถอดรหัส จะมีตารางในการเข้ารหัสอยู่ด้วยกัน 3 ตารางคือ

1. List of Insignificant Set (LIS) คือตารางที่ใช้เก็บตำแหน่งของกลุ่มสัมประสิทธิ์เวฟเลตที่ยังไม่มีความสำคัญ ซึ่ง LIS นี้จะมีอยู่ด้วยกัน 2

ชนิดคือ ชนิด A และ ชนิด B ซึ่งค่าที่เก็บใน LIS ชนิด A คือ $D(i,j)$ และค่าที่เก็บใน LIS ชนิด B คือ $L(i,j)$

2. List of Insignificant Pixel (LIP) คือตารางที่ใช้เก็บตำแหน่งของสัมประสิทธิ์เวฟเลตที่ยังไม่มีความสำคัญ
3. List of Significant Pixel (LSP) คือตารางที่ใช้เก็บตำแหน่งของสัมประสิทธิ์เวฟเลตที่มีความสำคัญ



รูปที่.5 แสดงความสัมพันธ์แบบ Spatial Orientation Trees

มีฟังก์ชันการเข้ารหัส $S_T(k)$ เป็นดังสมการ

$$S_T(k) = \begin{cases} 1, & \max_{(i,j) \in k} |c(i,j)| \geq T \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

$$n \leq \log_2(\max |c(i,j)|) \quad (2)$$

$$T = 2^n \quad (3)$$

เมื่อ k คือ สัมประสิทธิ์เวฟเลต $((i,j)$ หรือ (k,l)) หรือกลุ่มของสัมประสิทธิ์เวฟเลต $(D(i,j)$ หรือ $L(i,j)$)

3.1 อัลกอริทึมการเข้ารหัส

มีขั้นตอนดังนี้

1. กำหนดค่าเริ่มต้น
 - หาค่า n และ T จากสมการที่ 2 และ 3 ตามลำดับ
 - LSP กำหนดเป็นตารางว่าง
 - LIP กำหนดเป็นค่าสมาชิกของ H
 - LIS กำหนดเป็นค่าสมาชิกของ H และกำหนดให้เป็นชนิด A
2. การจัดลำดับ (Sorting)
 - สำหรับแต่ละ (i,j) ใน LIP
 - ส่งค่า $S_T((i,j))$ ออก

- ถ้าบิตเท่ากับ 1 ให้ส่งบิตเช็คเครื่องหมาย
 - สำหรับแต่ละ (i,j) ใน LIS
 - ถ้าเป็น LIS ชนิด A
 - * ส่งค่า $S_T(D(i,j))$ ออก
 - * ถ้า $S_T(D(i,j))$ เท่ากับ 1
 - สำหรับแต่ละ (k,l) ของ $O(i,j)$
 - ~ ส่งค่า $S_T((k,l))$ ออก
 - ~ ถ้าค่า $S_T((k,l))$ เท่ากับ 1 ให้ส่งบิตเช็คเครื่องหมายและให้ส่ง (k,l) ไป LSP
 - ~ ถ้าค่า $S_T((k,l))$ เท่ากับ 0 ให้ส่ง (k,l) ไป LIP
 - * ถ้า $L(i,j)$ มีสมาชิก ให้ส่ง (i,j) ไปท้าย LIS แล้วกำหนดให้เป็นชนิด B
 - ถ้าเป็น LIS ชนิด B
 - * ส่งค่า $S_T(L(i,j))$ ออก
 - ~ ถ้าค่า $S_T(L(i,j))$ เท่ากับ 1 ให้ส่งแต่ละสมาชิกของ $O(i,j)$ ไปท้าย LIS แล้ว กำหนดให้เป็นชนิด A และย้าย (i,j) ออกจาก LIS
3. Quantization
 - $n = n-1$
 - สำหรับแต่ละ (i,j) ใน LSP
 - ส่งค่าบิตที่ n ของ (i,j) ออก
4. Update
 - ทำซ้ำขั้นตอนที่ 2 ถึงขั้นตอนที่ 3 และยุติเมื่อมีการส่งข้อมูลการบีบอัดครบตามต้องการ

3.2 การจัดเก็บเป็นไฟล์การบีบอัดข้อมูล

การจัดเก็บไฟล์แบ่งออกเป็น 2 ส่วน คือ ไฟล์ส่วนหัวและไฟล์ข้อมูลที่ได้จากการเข้ารหัส โดยไฟล์แต่ละส่วนจะมีการจัดเก็บค่าดังนี้

1. ไฟล์ส่วนหัวจะทำการจัดเก็บข้อมูลคือ
 - จำนวนข้อมูลทั้งหมดที่ทำการบีบอัด
 - ค่า n
 - จำนวนคอลัมน์ของภาพ
 - จำนวนแถวของภาพ
 - จำนวนระดับการแปลงเวฟเลต
2. ไฟล์ข้อมูลจะทำการจัดเก็บข้อมูลทั้งหมดที่ได้จากการเข้ารหัสการบีบอัดข้อมูลด้วยวิธี SPIHT

3.3 อัลกอริทึมการถอดรหัส

ในการถอดรหัสจะใช้อัลกอริทึมเดียวกับการเข้ารหัส เพียงแต่เปลี่ยนจากการส่งค่าเป็นการรับค่า และนำค่านั้นมาพิจารณาเหมือนการเข้ารหัส

4. การพัฒนาอัลกอริทึม SPIHT

จากอัลกอริทึมที่กล่าวมามีจุดที่สามารถปรับปรุงพัฒนาได้ 2 จุดกล่าวคือ

1. กรณีที่ $O(i,j)$ ไม่มีระดับความสำคัญแต่ $L(i,j)$ มีระดับความสำคัญ จะมีการส่งบิตออกมา 2 บิต (ไม่นับรวมการเช็คแต่ละสมาชิกของ $O(i,j)$) ในกรณีนี้สามารถปรับอัลกอริทึมให้ส่งออกมาเพียง 1 บิต โดยการเพิ่มการเช็คการส่ง (k,l) เข้า LIP คือถ้ามีการส่ง (k,l) ทั้งหมดของ $O(i,j)$ แสดงว่าได้เกิดกรณีนี้ขึ้น ดังนั้นสามารถทำการส่ง (k,l) เข้า LIS ให้เป็นชนิด A ได้เลย
2. กรณีที่ $O(i,j)$ มีระดับความสำคัญแต่ $L(i,j)$ ไม่มีระดับความสำคัญ จะมีการส่งบิตออกมาเหมือนกับข้อ 1 ในกรณีนี้สามารถปรับอัลกอริทึมให้ส่งข้อมูลให้น้อยลงได้ โดยการเพิ่ม List of Forbidden Coefficients (LFC) มีลักษณะการหา LFC ดังนี้

- ทำการแบ่งสัมประสิทธิ์เวฟเลตแต่ละแบนด์ย่อยออกเป็น 4 ส่วน โดยเรียกแต่ละส่วนที่ทำการแบ่งว่า Label โดยแต่ละแบนด์ย่อยที่มีการแบ่งจะมีตำแหน่ง Label ไม่ซ้ำกัน
- หาค่าสัมประสิทธิ์สูงสุดของแต่ละ Label
- หาค่าระดับของการควอนไทซ์ในแต่ละ Label และนำมาทำเป็น LFC โดยหาได้จากสมการดังนี้

$$LFC(w) \leq \log_2 (\max(c_{label}(w))) \quad (4)$$

เมื่อ W คือตำแหน่งของ Label
 $C_{label}(W)$ คือสัมประสิทธิ์เวฟเลตทั้งหมดที่ W
 $LFC(W)$ คือระดับการควอนไทซ์ ที่ W

โดย LFC จะเป็นตัวบ่งบอกสถานะว่า $O(i,j)$ และ $L(i,j)$ ควรมีการส่งบิตออกมาหรือไม่ โดยจะดูว่า $O(i,j)$ หรือ $L(i,j)$ นี้อยู่ที่ Label ไหนและมีระดับการควอนไทซ์ที่เท่าใด ถ้าระดับของการควอนไทซ์ที่มีอยู่มีค่ามากกว่าระดับการควอนไทซ์ของ $O(i,j)$ หรือ $L(i,j)$ ก็ไม่ต้องการเช็ค $O(i,j)$ หรือ $L(i,j)$ สามารถเขียนเป็นฟังก์ชันได้ดังสมการที่ 3

$$F_n(p) = \begin{cases} 1, & \max_{w \in p} LFC(w) \leq n \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

เมื่อ $F_n(p)$ คือฟังก์ชันที่ใช้เป็นเงื่อนไขในการเช็คค่าระดับการควอนไทซ์ของ $O(i,j)$ หรือ $L(i,j)$
 p คือค่า LFC ทั้งหมดที่เป็นของสมาชิกใน $O(i,j)$ หรือ $L(i,j)$

4.1 อัลกอริทึมการเข้ารหัสที่มีการพัฒนา

มีการพัฒนาเฉพาะในส่วนการจัดลำดับในอัลกอริทึม SPIHT ส่วนอื่นจะเหมือนเดิม ดังนั้นจะแสดงเฉพาะในส่วนที่เป็นการจัดลำดับ และส่วนกำหนดค่าเริ่มต้น มีขั้นตอนดังนี้

1. กำหนดค่าเริ่มต้น

- หาค่า n และ T จากสมการที่ 2 และ 3 ตามลำดับ
- LSP กำหนดเป็นตารางว่าง
- LIP กำหนดเป็นค่าสมาชิกของ H
- LIS กำหนดเป็นค่าสมาชิกของ H และกำหนดให้เป็นชนิด A
- LFC กำหนดค่าเป็นไปตามสมการที่ 4

2. การจัดลำดับ

- สำหรับแต่ละ (i,j) ใน LIP
 - ส่งค่า $S_T(i,j)$ ออก
 - ถ้าบิตเท่ากับ 1 ให้ส่งบิตเช็คเครื่องหมาย
- สำหรับแต่ละ (i,j) ใน LIS
 - เช็คค่า FO จาก $F_n(O(i,j))$
 - เช็คค่า FL จาก $F_n(L(i,j))$
 - ถ้าเป็น LIS ชนิด A และ FO หรือ FL เท่ากับ 1
 - * ส่งค่า $S_T(D(i,j))$ ออก
 - * ถ้า $S_T(D(i,j))$ เท่ากับ 1
 - สำหรับแต่ละ (k,l) ของ $O(i,j)$
 - ~ ส่งค่า $S_T(k,l)$ ออก
 - ~ ถ้าค่า $S_T(k,l)$ เท่ากับ 1 ให้ส่งบิตเช็คเครื่องหมายและให้ส่ง (k,l) ไป LSP
 - ~ ถ้าค่า $S_T(k,l)$ เท่ากับ 0 ให้ส่ง (k,l) ไป LIP และเพิ่มค่าตัวเช็คการส่ง (k,l)
 - * ถ้า $L(i,j)$ มีสมาชิก
 - ~ ถ้าสมาชิกทุกตัวของ $O(i,j)$ ถูกส่งไป LIP ให้ทำการส่งสมาชิกทุกตัวของ $O(i,j)$ ไปท้าย LIS และกำหนดให้เป็นชนิด A
 - ~ ถ้าสมาชิกบางตัวของ $O(i,j)$ ไม่ถูกส่งไป LIP ให้ทำการส่ง (i,j) ไปท้าย LIS แล้วกำหนดให้เป็นชนิด B
 - ถ้าเป็น LIS ชนิด B และ FL เท่ากับ 1
 - * ส่งค่า $S_T(L(i,j))$ ออก
 - ~ ถ้าค่า $S_T(L(i,j))$ เท่ากับ 1 ให้ส่งแต่ละสมาชิกของ $O(i,j)$ ไปท้าย LIS แล้วกำหนดให้เป็นชนิด A และทำการย้าย (i,j) ออกจาก LIS

4.2 การจัดเก็บเป็นไฟล์การบีบอัดข้อมูล

การจัดเก็บไฟล์แบ่งออกเป็น 2 ส่วนคือ ไฟล์ส่วนหัวและไฟล์ข้อมูลที่ได้จากการเข้ารหัส โดยไฟล์แต่ละส่วนจะมีการจัดเก็บค่าดังนี้

1. ไฟล์ส่วนหัวจะทำการจัดเก็บข้อมูลคือ
 - จำนวนข้อมูลทั้งหมดที่ทำการบีบอัด
 - ค่า n
 - จำนวนคอลัมน์ของภาพ
 - จำนวนแถวของภาพ
 - จำนวนระดับการแปลงเวฟเลต
 - ค่า LFC

2. ไฟล์ข้อมูลจะทำการจัดเก็บข้อมูลทั้งหมดที่ได้จากการเข้ารหัสการบีบอัดข้อมูลด้วยวิธี SPIHT

4.3 อัลกอริทึมการถอดรหัสที่มีการพัฒนา

ในการถอดรหัสจะใช้อัลกอริทึมเดียวกับการเข้ารหัส เพียงแต่เปลี่ยนจากการส่งค่าเป็นการรับค่า และนำค่านั้นมาพิจารณาเหมือนการเข้ารหัส

5. การทดสอบ

5.1 วิธีการทดสอบ

1. ทำการทดสอบหาตระกูลเวฟเลตแม่ ที่ให้ผลการบีบอัดข้อมูลภาพ EM ด้วยอัลกอริทึม SPIHT และอัลกอริทึม SPIHT ที่ผ่านการปรับปรุง ให้ได้ผลการบีบอัดข้อมูลภาพ EM ได้ดีที่สุด โดยใช้ข้อมูลของภาพต้นแบบจำนวน 1 ภาพคือ “Zucchini” (เป็นภาพตัวอย่างจากกล้อง EM รุ่น JEM 2010 ของบริษัท JEOL LTD) ซึ่งมีข้อมูลภาพขนาด 512×512 พิกเซล แต่ละพิกเซลมี 8 บิต และตระกูลเวฟเลตแม่ที่ใช้ในการทดสอบมีอยู่ด้วยกัน 4 ตระกูลคือ bi9-7 [1] db4 [8] sym8 [5] และ coif5 [4] และทำการวัดประสิทธิภาพการบีบอัดข้อมูลภาพโดยใช้วิธี อัตราส่วนของสัญญาณต่อสัญญาณรบกวนสูงสุด (Peak signal-to-noise ratio, PSNR)

$$\text{PSNR(dB)} = 10 \log \frac{255^2}{\text{MSE}} \quad (6)$$

$$\text{MSE} = \frac{1}{M \times N} \sum_{x=1}^M \sum_{y=1}^N \left[f(x, y) - \hat{f}(x, y) \right]^2 \quad (7)$$

เมื่อ	M	คือจำนวนพิกเซลตามความกว้างภาพ
	N	คือจำนวนพิกเซลตามความสูงภาพ
	$f(x, y)$	คือค่าของพิกเซลที่ตำแหน่ง (x, y) ของภาพต้นแบบ
	$\hat{f}(x, y)$	คือค่าของพิกเซลที่ตำแหน่ง (x, y) ของภาพที่ผ่านการบีบอัดข้อมูล

ค่า PSNR จะสะท้อนถึงภาพที่ได้จากการบีบอัดว่ามีความผิดเพี้ยนจากภาพต้นแบบมากน้อยเพียงใด นั่นคือถ้าค่า PSNR มีค่ามากแสดงว่าภาพที่ได้จากการบีบอัดมีความผิดเพี้ยนน้อย และถ้าค่า PSNR มีค่าน้อยแสดงว่าภาพที่ได้จากการบีบอัดมีความผิดเพี้ยนมาก และวิธีอัตราบิต (Bit rate) คือค่าเฉลี่ยของจำนวนบิตต่อพิกเซล (Bits per pixel, bpp) ของภาพที่ผ่านการบีบอัดข้อมูล

$$\text{Bit rate (bpp)} = \frac{\text{bit compress}}{\text{pixel image}} \quad (8)$$

เมื่อ bit compress คือจำนวนบิตทั้งหมดของ ภาพที่ผ่านการบีบอัดข้อมูล

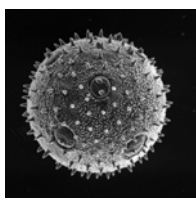
pixel image คือจำนวนพิกเซลทั้งหมดของภาพต้นแบบ

โดยอัตราบิต จะมีความสัมพันธ์กับขนาดของไฟล์ภาพที่ผ่านการบีบอัดข้อมูล เป็นดังตารางที่ 1 และทำการทดสอบที่อัตราบิตเดียวกันคือ 0.5, 1, 1.5 และ 2 bpp

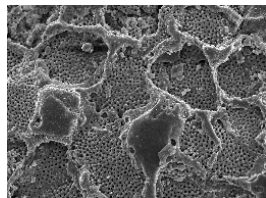
ตารางที่ 1. แสดงความสัมพันธ์ระหว่างอัตราบิต และขนาดของไฟล์ภาพที่ผ่านการบีบอัดข้อมูล

อัตราบิต (bpp)	ขนาดของไฟล์ภาพที่ผ่านการบีบอัดข้อมูล
0.25	1/32 ของไฟล์ภาพต้นแบบ
0.5	1/16 ของไฟล์ภาพต้นแบบ
1	1/8 ของไฟล์ภาพต้นแบบ
2	1/4 ของไฟล์ภาพต้นแบบ

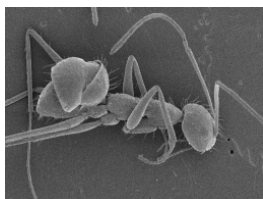
- ทำการทดสอบอัลกอริทึม SPIHT และอัลกอริทึม SPIHT ที่ผ่านการปรับปรุง ที่อัตราบิตเดียวกันคือ 0.25, 0.5 และ 1 bpp โดยใช้ข้อมูลของภาพต้นแบบจำนวน 4 ภาพคือ “Zucchini” และ “sand” “ant” และ “LED” (เป็นภาพที่ถ่ายจากกล้อง EM รุ่น JSM-5800 LV ของบริษัท JEOL LTD) ซึ่งมีข้อมูลภาพขนาด 960x1280 พิกเซล แต่ละพิกเซลมี 8 บิต ซึ่งภาพที่นำมาทดสอบได้แสดงไว้ในรูปที่ 6 (a-d)



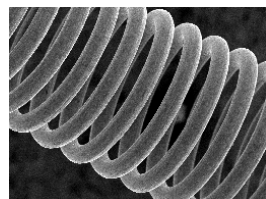
รูปที่ 6.a (Zucchini)



รูปที่ 6.b (sand)



รูปที่ 6.c (ant)



รูปที่ 6.d (LED)

รูปที่ 6 (a-d) แสดงรูปภาพที่ใช้ทดสอบการบีบอัดข้อมูลภาพ

5.2 ผลการทดสอบ

- ผลการทดสอบหาตระกูลเวฟเลตแม่ที่ให้ผลการบีบอัดข้อมูลภาพ EM ได้ดีที่สุด พบว่าทั้งอัลกอริทึม SPIHT และอัลกอริทึม SPIHT ที่ผ่านการปรับปรุง ที่ใช้ bi9-7 เป็นเวฟเลตแม่ในการแปลงเวฟเลต ให้ผลการบีบอัดข้อมูลภาพ EM ได้ดีที่สุด เป็นดัง ตารางที่ 2
- ผลการทดสอบการเปรียบเทียบประสิทธิภาพการบีบอัดข้อมูลภาพ EM ด้วยอัลกอริทึม SPIHT และอัลกอริทึม SPIHT ที่ผ่านการปรับปรุง พบว่า อัลกอริทึม SPIHT ที่ผ่านการปรับปรุงให้ผลการบีบอัดข้อมูลภาพ EM ได้ดีกว่า เป็นดังตารางที่ 3 และมีตัวอย่างภาพที่ผ่านการบีบอัดข้อมูลด้วยอัลกอริทึม SPIHT ที่ผ่านการปรับปรุง เป็นดังรูปที่ 7
- เวลาที่ใช้ในการบีบอัดข้อมูล ระหว่างอัลกอริทึม SPIHT และอัลกอริทึม SPIHT ที่ผ่านการปรับปรุง มีการใช้เวลาในการบีบอัดข้อมูลภาพที่ใกล้เคียงกันมาก เนื่องจากเงื่อนไขในการเข้ารหัสที่เพิ่มขึ้นในอัลกอริทึม SPIHT ที่ผ่านการปรับปรุง ไม่มีความซับซ้อน

ตารางที่ 2. แสดงผลการบีบอัดข้อมูลภาพ EM ที่ตระกูลเวฟเลตแม่ต่างๆ

ตระกูลเวฟเลตแม่	อัตราบิต (bpp)	PSNR (dB)	
		SPIHT	SPIHT*
bi9-7	0.5	25.9437	26.4209
	1	30.7854	31.4849
	1.5	35.0367	35.2730
	2	38.4622	38.7493
Db4	0.5	25.6627	25.7600
	1	30.0797	30.4873
	1.5	30.1865	34.7129
	2	37.3909	38.3268
Sym8	0.5	25.7788	25.8553
	1	30.1865	30.5499
	1.5	33.9292	34.6861
	2	37.3499	38.2538
Coif 5	0.5	25.5975	25.7016
	1	29.9516	30.2969
	1.5	33.6847	34.4694
	2	37.0617	38.0273

SPIHT * คือ SPIHT ที่ผ่านการปรับปรุง

ตารางที่ 3. แสดงผลการบีบอัดข้อมูลภาพ EM ด้วยอัลกอริทึม SPIHT และอัลกอริทึม SPIHT ที่ผ่านการปรับปรุง

ภาพ	อัตราบิต (bpp)	PSNR (dB)	
		SPIHT	SPIHT*
Zucchini	0.25	22.9363	23.1409
	0.5	25.9437	26.4209
	1	30.7854	31.4849
Sand	0.25	23.3655	23.5302
	0.5	25.1323	25.3597
	1	27.7760	28.0055
Ant	0.25	34.5026	34.7356
	0.5	35.4126	35.4504
	1	37.1842	37.2121
LED	0.25	27.5024	27.6934
	0.5	28.8230	29.0162
	1	30.7168	30.9553

SPIHT * คือ SPIHT ที่ผ่านการปรับปรุง

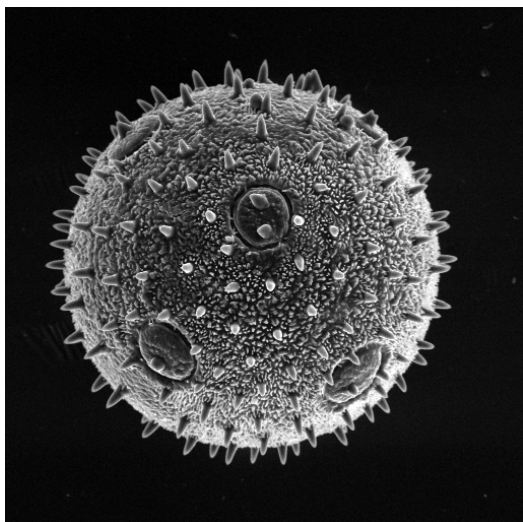
6. สรุป

จากผลการวิจัยที่ได้นำเสนอ จากตารางที่ 1 พบว่าการใช้ bi9-7 เป็นเวฟเลตแม่ ให้ผลการบีบอัดข้อมูลภาพ EM ได้มีประสิทธิภาพมากที่สุด และจากตารางที่ 2 อัลกอริทึม SPIHT ที่ผ่านการปรับปรุง โดยการเพิ่ม LFC และเพิ่มเงื่อนไขการเข้ารหัสและการถอดรหัส ให้ผลการบีบอัดข้อมูลภาพที่ดีขึ้น

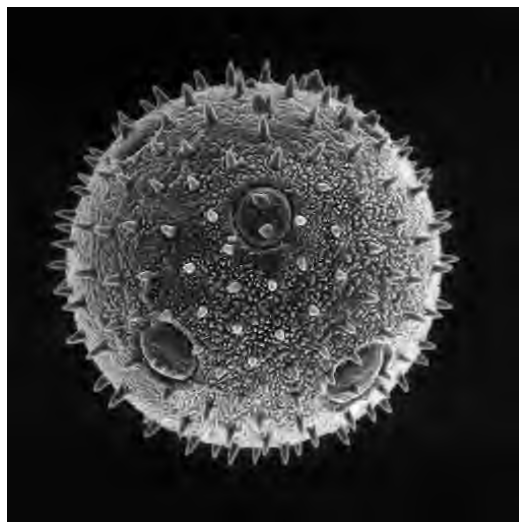
สำหรับงานวิจัยในอนาคต จะทำการพัฒนาการบีบอัดข้อมูลให้มีผลการบีบอัดข้อมูลที่ดีขึ้น ด้วยการนำมาเข้ารหัสเลขคณิต (Arithmetic Coding) และทำการหาข้อสรุปในการหาระดับอัตราบิตที่เหมาะสมของการบีบอัดข้อมูลภาพ EM จากการประเมินผลด้วยแบบสอบถาม

เอกสารอ้างอิง

- [1] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies. "Image coding using wavelet transform" IEEE Transaction on Image Processing. Vol 1, April 1992. pp 205-220.
- [2] A. Bovik, "Handbook of Image & Video Processing" New York: Academic Press. 2000
- [3] B. A. Banister and T. R. Fischer. "Quadtree Classification and TCQ Image Coding." IEEE Transactions on Circuit and Systems for Video Technology. Vol. 11, No 1, January 2001. pp 3-8.
- [4] C. S. Burrus and J. E. Odegard. "Coiflet System and Zero Moments." IEEE Transaction on Signal Processing. Vol 46, No 3, March 1998. pp. 761-766
- [5] C. S. Burrus, R. A. Gopinath and H. Guo. "Introduction to Wavelets and Wavelet Transforms." New York: Prentice-Hall International, Inc. 1998.
- [6] M. J. Lai. "On the Digital-Associated with Daubechies's Wavelets" IEEE Transaction on Signal Processing. VOL 43, September 1995. pp 2203-2205.
- [7] S. G. Mallat. "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation." IEEE Transaction on Pattern and Machine Intelligence. Vol 11, July 1989. pp.674-693.
- [8] A. Said and W. A. Pearlman., "A New Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees." IEEE Transaction on Circuit and Systems for Video Technology. Vol. 6, June 1996. pp 243-250.
- [9] J. M. Shapiro. "Embedded image coding using zerotrees of wavelets coefficients." IEEE Transaction on Signal Processing. Vol 41, Jan 1993. pp 3445 -3462.
- [10] ขวัญฤทัย ไพรีพ่ายฤทธิ์. ตระกูลเวฟเลตที่เหมาะสมสำหรับการเข้ารหัสแบบเอ็มเบดเดด ซีโวลฟ์เวฟเลตในการประยุกต์ใช้งานการแพทย์ทางไกล. วิทยา-นิพนธ์ปริญญาวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าธนบุรี. 2542.
- [11] ศิริพร เคชะศิริรักษ์. การลดขอบบล็อคในภาพ JPEG ด้วยวิธีเวฟเลตเรซอลต์ที่ปรับตัวเองได้. วิทยานิพนธ์ปริญญาวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าคุณทหารลาดกระบัง. 2543
- [12] เดโช ทองอร่าม และ สุวิทย์ ปุณณชัยยะ. กระบวนการทางภาพและการวิเคราะห์ภาพสำหรับกล้องจุลทรรศน์อิเล็กตรอนแบบสแกน. วารสารเครื่องมือวิจัยวิทยาศาสตร์และเทคโนโลยี. Vol. 5, 1995. หน้า 21 - 48.



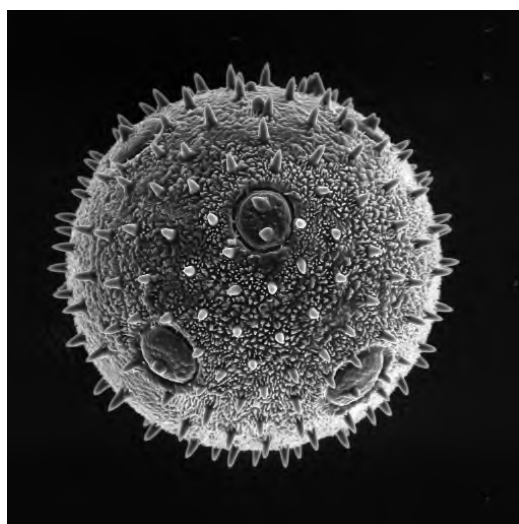
รูปที่ A1 ภาพต้นแบบ (Zucchini)
ขนาดไฟล์เท่ากับ 262,144 ไบต์



รูปที่ A2 อัตราบิต 0.25 bpp
ขนาดไฟล์เท่ากับ 8,192 ไบต์

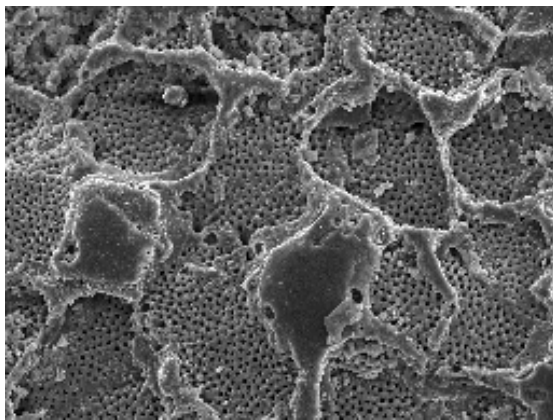


รูปที่ A3 อัตราบิต 0.5 bpp
ขนาดไฟล์เท่ากับ 16,384 ไบต์

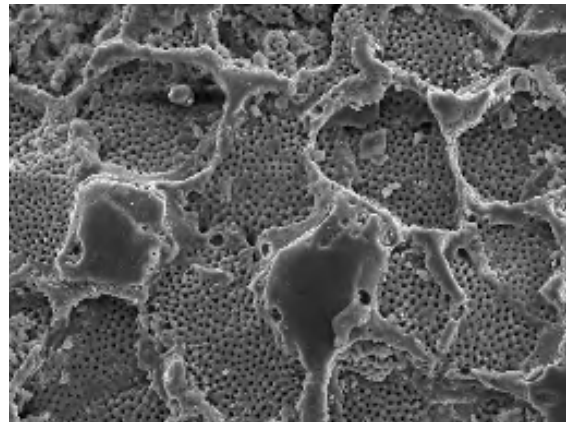


รูปที่ A4 อัตราบิต 1.00 bpp
ขนาดไฟล์เท่ากับ 32,768 ไบต์

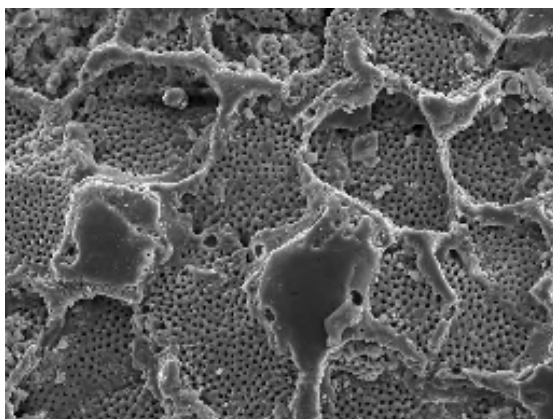
รูปที่ 7.1 แสดงรูปภาพต้นแบบ (A1) และภาพที่ผ่านการบีบอัดข้อมูลด้วยอัลกอริทึม SPIHT ที่ผ่านการปรับปรุง (A2-A4)



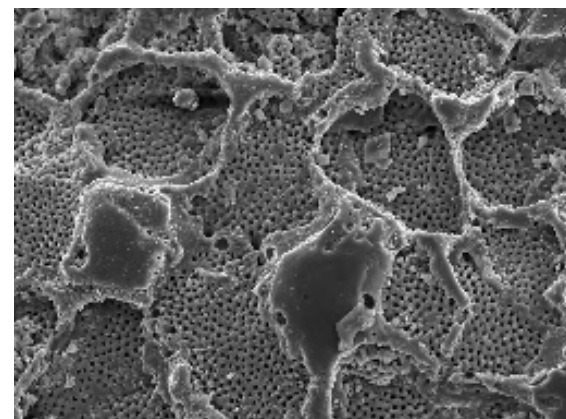
รูปที่ B1 รูปภาพต้นแบบ (sand)
ขนาดไฟล์เท่ากับ 1,228,800 ไบต์



รูปที่ B2 อัตราบิต 0.25 bpp
ขนาดไฟล์เท่ากับ 38,400 ไบต์



รูปที่ B3 อัตราบิต 0.50 bpp
ขนาดไฟล์เท่ากับ 76,800 ไบต์



รูปที่ B4 อัตราบิต 1.00 bpp
ขนาดไฟล์เท่ากับ 153,600 ไบต์

รูปที่ 7.2 แสดงรูปภาพต้นแบบ (B1) และภาพที่ผ่านการบีบอัดข้อมูลด้วยอัลกอริทึม SPIHT ที่ผ่านการปรับปรุง (B2-B4)



ณัฐนันท์ ทัดพิทักษ์กุล สำเร็จปริญญาตรีในสาขา
วิศวกรรมโทรคมนาคม จากมหาวิทยาลัย
เทคโนโลยีสุรนารี เมื่อ พ.ศ. 2543 ปัจจุบันกำลัง
ศึกษาระดับปริญญาโทที่มหาวิทยาลัยเทคโนโลยี
สุรนารี มีความสนใจงานวิจัยทางด้าน Digital

Signal Processing, Image Processing และ Wavelet Transform



กิตติ อรรถกิจมงคล สำเร็จปริญญาเอกในสาขา
วิศวกรรมไฟฟ้า จาก Vanderbilt University
ประเทศสหรัฐอเมริกา เมื่อ พ.ศ. 2542 ปัจจุบันดำรง
ตำแหน่งเป็นอาจารย์ประจำสาขาวิศวกรรม

ไฟฟ้า สำนักวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี มีความสนใจงานวิจัยทางด้าน Digital Signal Processing, Image Processing และ Wavelet Transform



สราวุฒิ สุจิตจอร์ สำเร็จปริญญาเอกในสาขา
วิศวกรรมไฟฟ้าจากมหาวิทยาลัยเบอร์มิงแฮม
ประเทศอังกฤษ ปัจจุบันดำรงตำแหน่งรอง
ศาสตราจารย์ และหัวหน้าสาขาวิศวกรรมไฟฟ้า
สำนักวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยี

สุรนารี ดำเนินงานวิจัยทางด้านสัญญาณระบบและการควบคุม