# การเร่งความเร็วอัลกอริธึมการประมวลผลภาพโดยใช้เว็บซีแอล บนหน่วยประมวลผลกราฟิกร่วม

# Accelerated Image Processing using WebCL on Integrated Graphic Processing Unit

ธนพงษ์ แสงสว่างชัย ศักดิ์นริน ปัญญาสาร รุ่งนภา ทารัตน์ อภิสิทธิ์ รัตนาตรานุรักษ์*

*สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะเทคโนโลยีอุตสาหกรรม มหาวิทยาลัยราชภัฏสวนสุนันทา*
**apisit.ra@ssru.ac.th*

**บทคัดย่อ**

*ในบทความนี้นำเสนอวิธีการประมวลผลภาพที่ถูกพัฒนาขึ้นจากการประมวลผลแบบตามลำดับไปเป็นการประมวลผลแบบขนาน ซึ่งอัลกอริธึมการประมวลผลภาพนี้ต้องใช้เวลาในการคำนวณมากขึ้นเมื่อภาพที่ใช้มีขนาดที่ใหญ่ขึ้น บทความนี้เริ่มจากการวิเคราะห์อัลกอริธึมในการประมวลผลภาพที่สามารถนำไปพัฒนาให้มีการทำงานแบบ ขนานได้ ในการเพิ่มประสิทธิภาพ อัลกอริธึมการประมวลผลภาพถูกประยุกต์เข้ากับเว็บซีแอล ซึ่งเป็นเครื่องมือในการพัฒนาโปรแกรมให้สามารถทำงานบนระบบหลายรูปแบบ ผลลัพธ์ของการประมวลผลแบบขนานได้เปรียบเทียบกับการประมวลผลแบบตามลำดับบนเครื่องอินเทลไอ 7 ซึ่งประกอบไปด้วยหน่วยประมวลผลกลางสองแกน และหน่วยประมวลผลกราฟิกร่วม ความเร็วที่เพิ่มขึ้นของวิธีการประมวลผลแบบขนานโดยใช้หน่วยประมวลผลกราฟิกร่วมสูงถึงสองเท่าเมื่อเปรียบเทียบกับวิธีการประมวลผลแบบตามลำดับ ซึ่งสิ่งที่ส่งผลต่อความเร็วที่เพิ่มขึ้นนั้นเกิดจากจำนวนแกนของหน่วยประมวลผลกลาง และหน่วยประมวลผลกราฟิก ความเร็วสัญญาณนาฬิกา หน่วยความจำ ขนาดของรูปภาพ และแบนวิธระหว่างซีพียูและจีพียู*

**คำสำคัญ:** การประมวลผลภาพ, เว็บซีแอล, การประมวลผลแบบขนาน, จีพียูร่วม

**Abstract**

*In this paper, image processing algorithms are developed from sequential processing to parallel processing. These algorithms require higher computational time when the image is larger. This paper starts with analysis the algorithms of image processing that can be developed to parallel. To improve the performance, image processing algorithm is applied to WebCL which is a framework for developing programs to execute on heterogeneous systems. The results of parallel processing are compared with the sequential one on Intel i7 that consists of 2 cores CPU and Intel Integrated GPU. Speedup of parallel processing on integrated GPU can be achieved to 2X compared with sequential one. In addition, the reasons that affect speedup are number of CPU and GPU cores, clock frequency, memory, image size, and bandwidth between CPU and GPU.*

**Keywords:** image processing, WebCL, parallel processing, integrated GPU.

**1. บทนำ**

ปัจจุบันเทคโนโลยีการถ่ายภาพนั้นมีประสิทธิภาพมากขึ้น ขนาดของภาพมีขนาดใหญ่มากยิ่งขึ้นเนื่องจากจำนวนพิกเซล (pixel) ภายในภาพ มีจำนวนมากขึ้นตามคุณสมบัติของกล้องถ่ายภาพ ไม่ว่าจะเป็นกล้องดีเอสแอลอาร์ (DSLR) กล้องมิลเลอร์เลส (Mirrorless) หรือกล้องถ่ายภาพบนโทรศัพท์มือถือ ซึ่งสิ่งที่ตามมาทำให้การตกแต่งภาพถ่ายนั้นใช้พลังในการประมวลผลมากขึ้นเนื่องจากปริมาณข้อมูลมีมากขึ้นตามขนาดของภาพ และส่งผลให้เวลาในการประมวลผลสูงขึ้นตามไปด้วย

การพัฒนาโปรแกรมปกตินั้นจะไม่สามารถดึงทรัพยากรมาใช้ได้อย่างมีประสิทธิภาพเพราะการพัฒนาโปรแกรมปกตินั้นจะใช้ซีพียู (Central Processing Unit : CPU) เพียง 1 แกน (core) แม้ว่าปัจจุบันจะมีการพัฒนาให้มีซีพียูมีหลายแกนก็ตาม ดังนั้นวิธีการที่จะเพิ่มประสิทธิภาพของซอฟต์แวร์นั้นจะต้องมีการพัฒนาซอฟต์แวร์ให้มีการทำงานแบบขนาน เพื่อนำเอาทรัพยากรที่ไม่ได้ทำงานไม่ว่าจะเป็น แกนอื่น ๆ ของซีพียู จีพียู หรือหน่วยประมวลผลร่วม (Co-processor) เพื่อเร่งความเร็วในการทำงานให้มีประสิทธิภาพมากยิ่งขึ้น โดยการเร่งความเร็วนั้นสามารถทำได้โดยการพัฒนาโปรแกรมให้มีการทำงานแบบขนานแบบข้อมูล (Data Parallelism) ซึ่งเป็นการสร้างเธรดให้มีจำนวนเท่ากับข้อมูลที่มีอยู่ ซึ่งข้อมูลในที่นี้คือพิกเซลทั้งหมดภายในภาพ หากจำนวนพิกเซลในภาพมีจำนวนมากจะทำให้ปริมาณข้อมูลมีมากขึ้น และทำให้การทำงานแบบตามลำดับ (Sequential) ใช้เวลามากขึ้น

ในปัจจุบันผู้ผลิตชิปประมวลผลได้มีการนำเอาจีพียู (Graphics Processing Units : GPU) เข้าไปรวมอยู่กับชิปซีพียู ซึ่งจีพียูโดยปกติแล้วจะมีหน้าที่ในการคำนวณภาพกราฟฟิกส์ ซึ่งจะไม่มีการใช้งานในการคำนวณปกติ ทำให้ไม่ได้นำเอาทรัพยากรมาใช้อย่างมีประสิทธิภาพ ซึ่งทำให้นักพัฒนาสามารถนำเอาการประมวลผลที่มีปริมาณมากเช่น การประมวลผลภาพเข้ามาใช้ในการประมวลผลบนจีพียู เพื่อให้ความเร็วของการทำงานที่ใช้น้อยลง โดยมีการนำเอาโอเพนซีแอล (Open Computing Language : OpenCL) และเว็บซีแอล (WebCL) (Khronos, 2010)(Aarnio, T. et al, 2012) ซึ่งเป็นไลบรารีที่ใช้ในการพัฒนาโปรแกรมให้มีการคำนวณแบบขนาน

จากการศึกษาพบว่าในปัจจุบันมีผู้ใช้งานอินเทอร์เน็ตเพิ่มขึ้นอย่างรวดเร็ว ซึ่งทำให้เทคโนโลยีเว็บนั้นเข้ามามีส่วนในการดำรงชีวิตในปัจจุบันของมนุษย์ ดังนั้นการทำเว็บแอปพลิเคชัน (Web Application) จึงทำให้เกิดความสะดวกในการใช้งานกว่าการใช้ แอปพลิเคชัน (Application) ที่ติดตั้งได้ในเครื่องทั่วไป และสามารถใช้งานได้บนหลายแพลตฟอร์ม (Platform) ดังนั้นการนำเอาเว็บแอปพลิเคชันมามีส่วนร่วมในการสร้างโปรแกรมในการตกแต่งภาพ ทำให้เกิดความสะดวกต่อผู้ใช้งาน และสามารถทำงานได้ทุกอุปกรณ์ ทำให้สามารถปฏิบัติงานต่าง ๆ ได้อย่างมีประสิทธิภาพมากยิ่งขึ้น

ในเอกสารนี้จะแสดงเนื้อหาดังต่อไปนี้ วัตถุประสงค์แสดงในส่วนที่ 2 เอกสารและงานวิจัยที่เกี่ยวข้องซึ่งจะกล่าวถึงโอเพนซีแอล และเว็บซีแอล การประมวลผลภาพ ในส่วนที่ 3 ถัดจากนั้นในส่วนที่ 4 จะนำเสนอวิธีการทดลอง จากนั้นจะมีการแสดงผลการวิจัยในส่วนที่ 5 และสรุปผลการทดลองในส่วนที่ 6

**2. วัตถุประสงค์การวิจัย**

ในงานวิจัยชิ้นนี้จะมีการพัฒนาอัลกอริธึมการประมวลผลภาพที่มีการทำงานแบบตามลำดับ มาแปลงให้มีการทำงานแบบขนาน และทำงานบนหน้าเว็บ ด้วยเว็บซีแอล ซึ่งทำให้สามารถเปิดโปรแกรมผ่านหน้าเว็บได้ และสามารถประมวลผลด้วยหน่วยประมวลผลกราฟิกร่วมซึ่งมีการใช้งานมากขึ้นในปัจจุบัน โดยงานวิจัยชิ้นนี้จะแสดงให้เห็นวิธีการเร่งความเร็วในการประมวลผลภาพจากวิธีการประมวลผลผ่านซีพียูแบบเดิม ให้มีความเร็วมากยิ่งขึ้นโดยใช้หน่วยประมวลผลกราฟิกร่วมด้วยเว็บซีแอล

**3. เอกสารและงานวิจัยที่เกี่ยวข้อง**

**3.1 โอเพนซีแอล และเว็บซีแอล**

ในปัจจุบันผู้ผลิตหน่วยประมวลผลได้ออกแบบให้ชิปประมวลผลนั้นมีหลายแกน และได้ใส่หน่วยประมวลผลกราฟิกลงไปในหน่วยประมวลผลกลางอีกด้วย โดยผู้ผลิตรายใหญ่คืออินเทล (Intel) และ เอเอ็มดี (AMD) (AMD, 2011) ได้สร้างหน่วยประมวลผลที่ใส่หน่วยประมวลผลกราฟิกเข้าไปในซีพียูหลายแกน เพื่อลดช่องว่างและเวลาที่ใช้ในการส่งข้อมูลจากหน่วยประมวลผลกลางไปยังหน่วยประมวลผลกราฟิก เพื่อให้งานที่ประมวลผลมีประสิทธิภาพมากยิ่งขึ้น

ภายในหน่วยประมวลผลกราฟิก จะมีแกนประมวลผลสตรีม (Stream Core) จำนวนมาก (NVIDIA, 2009) (AMD, 2011) ซึ่งภายในจะสามารถประมวลผลคำสั่งได้ และมีจำนวนมากกว่าซีพียูหลายแกน ซึ่งหากมีการประมวลผลบนหน่วย

ประมวลผลกราฟิกจะมีโอกาสที่จะได้ประสิทธิภาพในการทำงานสูงกว่าการประมวลผลด้วยซีพียูหลายแกน แม้ว่าความถี่ของสัญญาณนาฬิกาของหน่วยประมวลผลกลางจะสูงกว่าก็ตาม

การพัฒนาโปรแกรมในยุคปัจจุบันสามารถนำเอาประมวลผลกลางและหน่วยประมวลผลกราฟิกเข้ามาร่วมการทำงานกันได้เพื่อเพิ่มประสิทธิภาพในการประมวลผลแบบขนาน ซึ่งจุดเด่นของหน่วยประมวลผลกราฟิกคือมีจำนวนแกนมากกว่าหน่วยประมวลผลกลาง ทำให้สามารถประมวลผลแบบขนานได้อย่างมีประสิทธิภาพสูงกว่าการประมวลผลด้วยหน่วยประมวลผลกลาง อย่างไรก็ตามการนำเอาหน่วยประมวลผลกราฟิก เข้ามาประมวลผลด้วย จะต้องมีการพัฒนาโปรแกรมจากการทำงานแบบตามลำดับ (Sequential) ให้มีการทำงานแบบขนานได้ เพื่อแบ่งงานที่มีจำนวนมากให้กับหน่วยประมวลผลแต่ละหน่วยได้ทำงานอย่างมีประสิทธิภาพ

โอเพนซีแอลเป็นมาตรฐานกลางในการพัฒนาโปรแกรมแบบขนานที่สามารถทำงานได้บนหน่วยประมวลผลกลาง หน่วยประมวลผลกราฟิก และหน่วยประมวลผลอื่น ซึ่งมาตรฐานนี้สามารถนำเอาไปพัฒนาได้บนทุกแพลตฟอร์ม

ในปัจจุบันมีเฟรมเวิร์ค เว็บซีแอล (WebCL) (Khronos, 2010)(Aarnio, T. et al, 2012) ในการพัฒนาโปรแกรมเพื่อให้โปรแกรมสามารถทำงานได้ระหว่างแพลทฟอร์ม ซีพียู จีพียู หรือหน่วยประมวลผลอื่น ซึ่งเว็บซีแอลนั้นจะมีรูปแบบการทำงานคล้ายกับโอเพนซีแอล และมีความแตกต่างจากโอเพนซีแอลเล็กน้อย โดยเว็บซีแอลนั้นจะทำงานอยู่บนเว็บบราวเซอร์ โดยมีภาษาที่ใช้พัฒนาฝั่งโฮสต์คือ ภาษาจาวาสคริปต์ แต่เคอร์นอลจะใช้ภาษา C99 ในการพัฒนาเช่นเดียวกันกับการพัฒนาโอเพนซีแอลปกติ

เว็บซีแอลมีความแตกต่างกับโอเพนซีแอลเพียงเล็กน้อย โดยแบ่งออกเป็น 3 ส่วนหลักคือ ส่วนของจาวาสคริปต์ จะมีการจัดการบนโฮสต์เท่านั้น ไม่ได้ติดต่อกับอุปกรณ์โดยตรง ส่วนนี้ใช้ในการจองพื้นที่ ไม่ว่าจะเป็นการจองพื้นที่รูป หรือข้อมูลในฝั่งโฮสต์ หรือพื้นที่ส่วนอื่น ส่วนของเว็บบราวเซอร์เป็นส่วนที่จาวาสคริปต์ติดต่อกับส่วนเสริมเว็บซีแอลส่วนนี้จะเป็นคำสั่งเกี่ยวกับการจองพื้นที่ที่จำเป็นในการพัฒนาโปรแกรมด้วยเว็บซีแอล และส่วนของโอเพนซีแอล เป็นส่วนที่ทำงานร่วมกับอุปกรณ์ เป็นส่วนที่ใช้ในการจัดการอุปกรณ์เช่นหน่วยประมวลผลกลาง หน่วยประมวลผลกราฟิก เพื่อให้คำนวณและประมวลผลบนอุปกรณ์ได้อย่างถูกต้อง

**3.2 การประมวลผลภาพ**

การประมวลผลภาพ คือการนำภาพมาประมวลผลหรือคิดคำนวณด้วยคอมพิวเตอร์ เพื่อให้ได้ข้อมูลภาพที่ต้องการ เช่น การทำให้ภาพมีความคมชัดมากขึ้น การจัดการสัญญาณรบกวน การแบ่งส่วนของวัตถุที่สนใจ ซึ่งกระบวนการเหล่านี้ทำให้ผู้ใช้สามารถนำภาพไปวิเคราะห์และนำไปใช้ประโยชน์ที่ต้องการ เช่น ระบบรู้จำลายนิ้วมือ การสแกนม่านตา การคัดแยกจดหมายของระบบไปรษณีย์ การเก็บข้อมูลรถเข้าออกในอาคารโดยการตรวจสอบป้ายทะเบียนรถ การใช้ระบบรู้จำใบหน้าของบุคคลในการตรวจสอบคนเข้าเมือง ซึ่งระบบเหล่านี้ใช้การประมวลผลภาพ และต้องมีการทำซ้ำ ซึ่งหากใช้มนุษย์ในการวิเคราะห์จะทำให้ใช้เวลามากกว่าการใช้คอมพิวเตอร์ และสิ้นเปลืองแรงงาน ทำให้เกิดความล้าระหว่างการทำงานและเกิดความผิดพลาดในที่สุด ซึ่งคอมพิวเตอร์จึงเข้ามามีบทบาทในการทำงานแทนมนุษย์

การประมวลผลภาพมีรูปแบบต่าง ๆ เช่น เบลอ (Blur) เป็นการลดความละเอียดของภาพลงตามส่วนที่ต้องการ การเปลี่ยนแปลงความสว่าง (Brightness) เป็นการปรับความสว่าง คอนทราสต์ (Contrast) เป็นการเปลี่ยนแปลงความแตกต่างระหว่างโทนมืดและโทนสว่าง การตัดภาพ (Crop) เป็นการตัดบางส่วนของภาพตามที่ต้องการ การเปลี่ยนภาพสี เป็นขาวดำ (Black & White) การกลับสี (Invert Color) เป็นการกลับสีที่อยู่ตรงข้ามกับสีเดิม การเปลี่ยนแปลงสัญญาณรบกวน (Noise) สัญญาณรบกวนนั้นเป็นสิ่งที่เกิดขึ้นกับภาพซึ่งทำให้ภาพหยาบ ซึ่งในการถ่ายภาพมักเกิดจากการเปลี่ยนแปลงค่า ISO ให้เพิ่มมากขึ้นเพื่อให้สามารถรับภาพที่มีความสว่างน้อยได้ จะทำให้เกิดสัญญาณรบกวนมากขึ้น ซึ่งทำให้ภาพไม่สวยงาม การหมุนภาพ (Rotate) เป็นการหมุนภาพไปในแนวที่ต่างกัน

**3.3 งานวิจัยที่เกี่ยวข้อง**

เว็บบราวเซอร์สมัยใหม่ส่วนใหญ่จะสามารถสร้างแอปพลิเคชัน โดยใช้ภาษาจาวาสคริปต์ในการพัฒนา โดยสามารถใช้เว็บซีแอล ในการเขียนโปรแกรมโดยให้ทำงานบนซีพียู และจีพียู ในการทำงาน อย่างไรก็ตามในการพัฒนาโปรแกรมปกติสามารถทำงานได้เพียง 1 อุปกรณ์ โดยงานวิจัยของ Xianglan Piao, et al. ได้ทดสอบเว็บซีแอลด้วยสี่โปรแกรม ซึ่งการใช้เฟรมเวิร์คนี้สามารถเพิ่มความเร็วได้เพิ่มขึ้น 65% เมื่อเทียบกับการทำงานด้วยจาวาสคริปต์ปกติ และเพิ่มขึ้น 33% เมื่อเปรียบเทียบกับการทำงานบนจีพียูอย่างเดียว (Piao, Xianglan, et al, 2014)

งานวิจัย CrowdCL ของ Tommy MacWilliam, et al. ได้นำเสนอเฟรมเวิร์คที่ใช้ในการพัฒนาแอปพลิเคชันบนเว็บโดยใช้โอเพนซีแอลที่ทำงานบนจีพียู โดยในการทำงานนั้นมีประสิทธิภาพมากยิ่งขึ้นเมื่อเทียบกับการทำงานบนจาวาสคริปต์ โดย

ที่ผลลัพธ์ปัญหาทอมสัน ซึ่งผลลัพธ์ที่ได้ CrowedCL นั้นทำงานได้อย่างมีประสิทธิภาพเมื่อเทียบกับการทำงานโดยใช้แกนเดียว (MacWilliam, et al., 2013)

ในการเขียนโปรแกรมบนเว็บนั้นจำเป็นจะต้องใช้ภาษาจาวาสคริปต์ซึ่งเป็นภาษาที่ทำงานได้บนหน่วยประมวลผลเพียง 1 แกน ซึ่งในงานวิจัยการพัฒนาโปรแกรมแบบขนานบนเว็บของ Stephen Herhut et al. ได้มีการทำคอมไพเลอร์โพรโทไทป์แบบ JIT ในบราวเซอร์ไฟร์ฟอกซ์ (Firefox) โดยมีการทดสอบกับการสร้างภาพสามมิติ ซึ่งผลลัพธ์ที่ได้เมื่อเพิ่มงานที่ทำแบบขนานให้มากขึ้น เฟรมต่อวินาที ของการทำงานแบบขนานจะมากยิ่งขึ้นตามไปด้วย (Herhut, et al.,2012).

นอกจากนี้ยังมีงานวิจัยเว็บซีแอลสำหรับการเพิ่มความเร็วเว็บแอปพลิเคชันบนฮาร์ดแวร์โดย Won Jeon et al. ซึ่งในงานวิจัยชิ้นนี้ได้มีการนำเอาภาษาเอชทีเอ็มแอลรุ่น 5 (HTML5) ซึ่งเป็นเอชทีเอ็มแอลรุ่นใหม่ มาทำงานร่วมกับภาษาจาวาสคริปต์ โดยมีการนำเอาเว็บแอปพลิเคชันมาพัฒนาร่วมกันกับการประมวลผลร่วมระหว่างซีพียู จีพียู โดยทดสอบร่วมกับ Sobel Filter N-body simulation และ Deformable body simulation ซึ่งทั้ง 3 อัลกอริธึมที่นำมาทดสอบนั้นใช้พลังในการประมวลผลสูงมาก โดยความเร็วที่สูงขึ้นของ Sobel filter ขึ้นถึง 13 เท่า ที่ภาพขนาด 256 x 256 พิกเซล N Body simulation ที่ 1024 พาติเคิล มีความเร็วเพิ่มขึ้นสูงถึง 23 เท่า และ Deformable body simulation มีความเร็วสูงสุดถึง 116 เท่า (Jeon Won et al, 2012)

งานวิจัยการใช้จาวาสคริปต์และเว็บซีแอลสำหรับการคำนวณทางคณิตศาสตร์ ของ Faiz Khan et al. ได้มีการเปรียบเทียบประสิทธิภาพของการทำงานโดยใช้ภาษาจาวาสคริปต์ เทียบกับการนำเอาเว็บซีแอลมาพัฒนาแบบขนาน โดยใช้อัลกอริธึมของ Dense Linear Algebra, Sparse Linear Algebra, Spectral Methods, N-Body, Structured grids, Unstructed grids, Map Reduce, Combinatorial logic, Graph traversal, Dynamic programming, Backtrack and Branch-and-bound และ Graphical models ซึ่งผลลัพธ์ที่ได้พบว่าอัลกอริธึมส่วนใหญ่จะทำงานได้ดีกว่าการทำงานบนจาวาสคริปต์ปกติเพียงอย่างเดียว นอกจากนี้ยังพบว่าเวลาที่ใช้ส่วนใหญ่สูญเสียไปกับการตั้งค่าฟังก์ชันก่อนการใช้เว็บซีแอล แต่อย่างไรก็ตามการคำนวณแบบขนานส่วนใหญ่ยังใช้เวลาในการทำงานน้อยกว่าการทำงานแบบตามลำดับขั้นปกติ (Khan Faiz et al, 2014)

ในงานวิจัย ParalleUS : การทำงานของจาวาสคริปต์บนระบบที่มีหลากหลายรูปแบบของ Jin Wang et al. ได้มีการนำเสนอเฟรมเวิร์คในการนำไปทำงานบนระบบที่มีหลากหลายแบบคือสามารถทำงานบนซีพียู จีพียู โดยการพัฒนาโปรแกรมด้วยภาษาระดับสูงซึ่งผลลัพธ์ที่ได้มีความเร็วเพิ่มขึ้น 26.8 เท่าเมื่อเทียบกับการพัฒนาด้วยเว็บบราวเซอร์ที่ใช้จาวาสคริปต์ปกติ (Wang Jin et al, 2014)

ในงานวิจัยการประมวลผลแบบกระจาย (Distributed Computing) บนบราวเซอร์ของ Cushing, Reginald, et al. ได้กล่าวถึงประสิทธิภาพของภาษาจาวาสคริปต์ การประมวลผลของเว็บบราวเซอร์ โดยใช้ WeevilScount และได้นำเอาเว็บบราวเซอร์มาทดสอบประสิทธิภาพของการประมวลผลในงานทางด้านชีวสารสนเทศ (bio-informatics) โดยใช้ภาษาจาวาสคริปต์ในการพัฒนา ซึ่งในการประมวลผลงานได้อย่างมีประสิทธิภาพมากยิ่งขึ้น (Cushing, Reginald, et al, 2013)

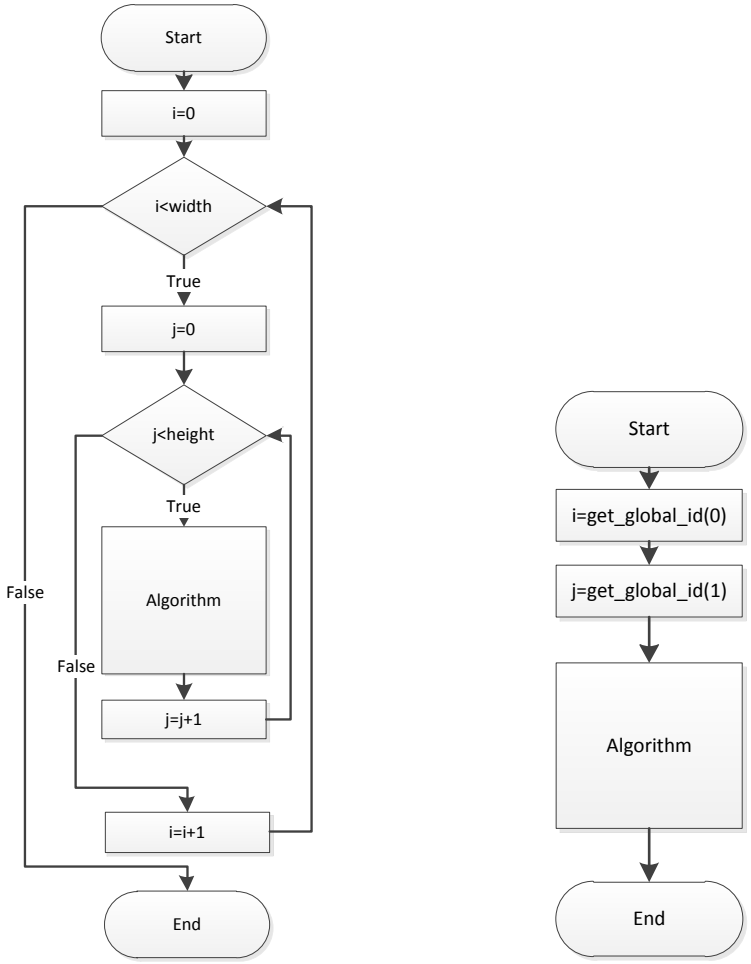
## 4. วิธีการวิจัย

จากการวิจัยที่เกี่ยวข้องจะเห็นว่าการพัฒนาโปรแกรมบนหน้าเว็บจะต้องใช้ภาษาจาวาสคริปต์ทำงานร่วมกับภาษาเอชทีเอ็มแอล ซึ่งทำให้สามารถพัฒนาโปรแกรมการประมวลผลภาพ อย่างไรก็ตามการใช้ภาษาจาวาสคริปต์นั้นสามารถสั่งงานได้เพียงซีพียู เพียง 1 แกนเท่านั้น ทำให้การประมวลผลไม่เต็มที่ อีกทั้งยังไม่สามารถนำเอาทรัพยากรที่มีอยู่ เช่น ซีพียูแกนที่เหลือ และหน่วยประมวลผลกราฟิกมาใช้ประโยชน์อย่างสูงสุด ดังนั้นในงานวิจัยนี้จะมีการนำเอาเว็บซีแอลเข้ามาใช้ในการพัฒนาโปรแกรมประมวลผลภาพบนเว็บให้มีการดึงเอาทรัพยากรภายในเครื่องคอมพิวเตอร์มาประมวลผลภาพแบบขนาน เพื่อลดระยะเวลาในการทำงาน

### 4.1 การออกแบบ

ในการพัฒนาโปรแกรมประมวลผลภาพบนเว็บ จะใช้เว็บบราวเซอร์โมซิลลา ไฟร์ฟอกซ์ ที่ติดตั้งเว็บซีแอล ในการพัฒนาและทดสอบโปรแกรม ซึ่งภายในโปรแกรมนั้นจะมีการทดสอบอัลกอริธึมในการประมวลผลภาพบนจาวาสคริปต์ โดยการนำเอาฟังก์ชันในการประมวลผลภาพมาทดสอบการประมวลผลด้านเวลา เช่น Brightness Contrast Inversion Posterize และ Threshold ถัดจากนั้นจะมีการพัฒนาโปรแกรมให้มีการทำงานแบบขนานโดยใช้เว็บซีแอลเพื่อนำงานในการประมวลผลภาพนั้นไปให้กับหน่วยประมวลผลกราฟิก จากนั้นจับเวลาและนำมาคำนวณหาความเร็วที่เพิ่มขึ้น เพื่อทดสอบประสิทธิภาพในการทำงาน

ในฟังก์ชันที่นำมาทดลองนั้นสามารถใช้ฟังก์ชัน get_global_id(0) เพื่อแทนตำแหน่งของพิกเซลข้อมูลในแกน x และ get_global_id(1) เพื่อแทนตำแหน่งของพิกเซลข้อมูลในแกน y ได้เช่นกัน เนื่องจากการทำงานภายในฟังก์ชันเป็นอิสระต่อกันทุกพิกเซล ทำให้สามารถปรับการทำงานให้มีการทำงานแบบขนานได้ง่าย ซึ่งอัลกอริธึมแบบตามลำดับจะมีรูปแบบการประมวลผลดังรูปภาพที่ 1(ก) และมีการพัฒนาเพื่อให้มีการทำงานแบบขนานดังรูปภาพที่ 1(ข) เมื่อ width คือ ความกว้างของรูปภาพ height คือความยาวของรูปภาพ i คือ ตำแหน่งของพิกเซลในแกน x j คือ ตำแหน่งของพิกเซลในแกน y




(ก) (ข)

**รูปภาพที่ 1** ผังงานของวิธีการประมวลผลภาพที่มีรูปแบบการประมวลผลแบบตามลำดับ(ก) และแบบขนาน(ข)

### 4.2 การทดสอบ

ในการทดสอบจะใช้รูปภาพ 3 ขนาด โดยหากภาพขนาดใหญ่จะมีจำนวนพิกเซลในภาพมากยิ่งขึ้น ส่งผลให้ปริมาณข้อมูลมีมากตามไปด้วย โดยจะมีการนำเอาภาพทั้ง 3 ขนาดไปทดสอบกับฟังก์ชันดังต่อไปนี้ โดยทดสอบทั้งการทำงานบนซีพียูแบบแกนเดียว และการทำงานบนจีพียูร่วม

#### 4.2.1 Brightness

ในฟังก์ชันนี้จะมีการเพิ่ม และลดค่าสีในภาพ เมื่อเพิ่ม Brightness จะมีการเพิ่มค่าลงไปในตัวแปรค่าสีแดง เขียว และน้ำเงิน เพื่อให้ภาพสว่างยิ่งขึ้น และถ้าลดค่า Brightness จะมีการลดค่าตัวแปรสีแดง เขียวและน้ำเงิน ทำให้ความสว่างของภาพมีค่าลดลงแทน

#### 4.2.2 Contrast

ในฟังก์ชันนี้จะมีการปรับค่า Contrast โดยการหาหาค่าตัวประกอบเพื่อคูณค่ากลับไปยัง ตัวแปรแต่ละสี นั่นคือ สีแดง สีเขียว และสีน้ำเงิน

#### 4.2.3 Desaturation

ในฟังก์ชันนี้จะมีการปรับลดความสดของสี ซึ่งจะมีการหาค่าตัวประกอบเพื่อคูณค่ากลับไปยัง ตัวแปรสีแดง สีเขียว และสีน้ำเงิน เพื่อลดความสดของสี

#### 4.2.4 Inversion

ในฟังก์ชันนี้จะมีเปลี่ยนค่าสีโดยการนำเอาค่าสีเดิมในแต่ละพิกเซล หลังจากนั้นคำนวณค่าสีแดง เขียว และน้ำเงินใหม่ โดยการหาค่าความแตกต่างระหว่างสีขาว และสีอื่น

#### 4.2.5 Posterize

ฟังก์ชันนี้จะเป็นการเปลี่ยนค่าสีใหม่ โดยการนำเอาค่าตัวประกอบที่ได้จากการคำนวณไปคูณเข้ากับค่าสีเดิมเพื่อให้ได้สีใหม่ที่ต้องการ และใส่ค่ากลับไปยังแต่ละพิกเซล

#### 4.2.6 Threshold

ฟังก์ชันนี้จะเป็นการปรับภาพให้เป็นสองสี คือ ขาว (255) และดำ (0) โดยหาค่ากลาง และเปรียบเทียบกับค่า Threshold ที่ต้องการ

จะเห็นว่าในผังงานของอัลกอริธึมในรูปแบบตามลำดับจะเป็นการวนซ้ำซึ่งจำนวนรอบของการวนซ้ำจะขึ้นอยู่กับความกว้าง และความยาวของภาพ ซึ่งหากมีภาพขนานใหญ่จะมีจำนวนรอบของการวนซ้ำมากยิ่งขึ้น ในทางกลับกันผังงานของอัลกอริธึมในรูปแบบขนาน จะไม่มีการวนซ้ำของอัลกอริธึมเนื่องจากการทำงานจะทำพร้อมกันครั้งเดียวโดยการแจกงานไปในแต่ละหน่วยประมวลผลของหน่วยประมวลผลกราฟิก

## 5. ผลการวิจัย

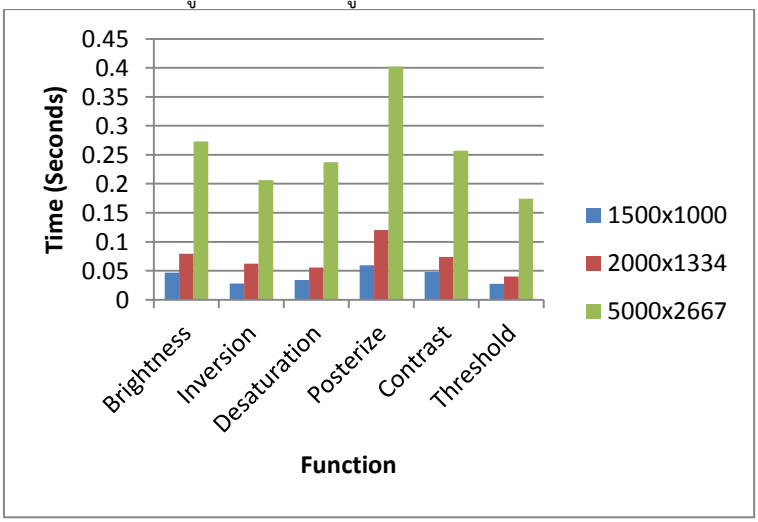
### 5.1 สภาพแวดล้อมในการทดลอง

ในการทดสอบการทำงานของโปรแกรมจะมีใช้เครื่อง Intel i7-4510U ความถี่ 2 GHz ซึ่งประกอบไปด้วย 2 แกนประมวลผล และมีเทอร์โบบูสต์ถึง 3.1 GHz หน่วยความจำ 8 GB บนระบบปฏิบัติการวินโดวส์ 8.1 Pro 64 บิต หน่วยประมวลผลกราฟิกร่วม Intel HD Graphics 4400 หน่วยความจำ 2 GB โดยใช้ Mozilla Firefox 24.0 และ Nokia WebCL 1.0 ในการทดสอบ

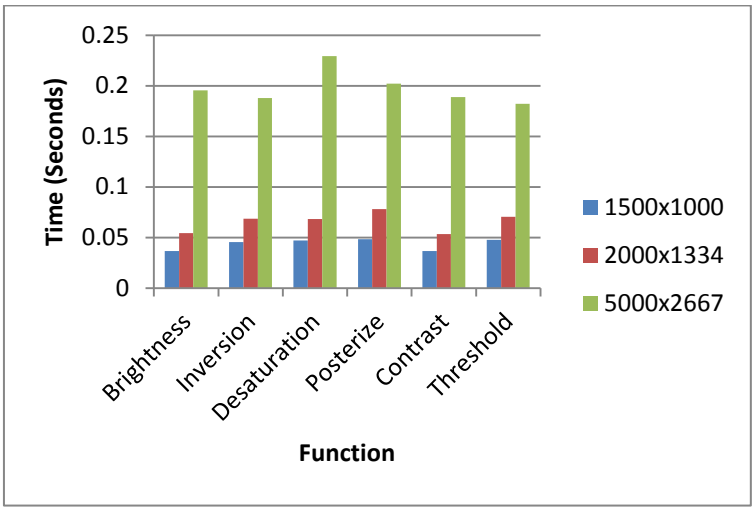
ในการทดสอบนี้จะมีการเปลี่ยนรูปภาพขนาดต่างกันสามภาพคือ 5000x2667 , 2000 x 1334 และ 1500 x 1000 pixel เนื่องจากเป็นขนาดของภาพที่ได้จากกล้องดิจิตอล

### 5.2 ผลการทดสอบ

ในการทดสอบภาพทั้ง 3 ขนาดกับทุกฟังก์ชันโดยการจับเวลา โดยมีการทดสอบกับรุ่นที่ใช้จาวาสคริปต์ที่ทำงานแบบ 1 แกน เพียงอย่างเดียว และใช้เว็บซีแอลที่ทำงานบนหน่วยประมวลผลกราฟิกที่รวมกับหน่วยประมวลผลกลาง (Integrated GPU) โดยฟังก์ชัน Posterize จะใช้เวลาในการประมวลผลมากที่สุด และเวลาที่ใช้ในแต่ละฟังก์ชันจะมากยิ่งขึ้นเมื่อขนาดของภาพมีค่ามากยิ่งขึ้น ซึ่งผลลัพธ์ที่ได้แสดงดังรูปภาพที่ 2 และรูปภาพที่ 3



**รูปภาพที่ 2** เวลาที่ใช้ในการประมวลผลในแต่ละฟังก์ชัน โดยใช้หน่วยประมวลผลกลาง โดยใช้ขนาดภาพต่างกัน



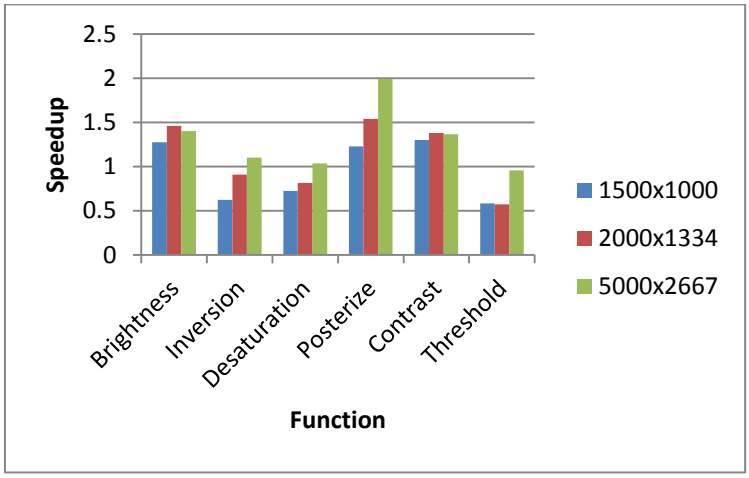

**รูปภาพที่ 3** เวลาที่ใช้ในการประมวลผลในแต่ละฟังก์ชัน โดยใช้หน่วยประมวลผลกราฟิกร่วม โดยใช้ขนาดภาพต่างกัน

### 5.3 Speedup

Speedup เป็นสัดส่วนของเวลาที่ใช้ของโปรแกรมที่ใช้จาวาสคริปต์ประมวลผลแบบ 1 แกน กับโปรแกรมที่ใช้เว็บซีแอล ดังสมการ (1)

$$Speedup = \frac{Javascript_Runtime}{WebCL_Runtime} \quad (1)$$

ในการหาค่า Speedup นั้นจะทำให้ทราบความเร็วที่เพิ่มขึ้นตามผลลัพธ์ที่แสดงดังรูปภาพที่ 4




**รูปภาพที่ 4** Speedup ของแต่ละฟังก์ชันระหว่างหน่วยประมวลผลกลาง และหน่วยประมวลผลกราฟิกร่วม

จากการทดลองจะพบว่าเมื่อขนาดภาพมากยิ่งขึ้นจะทำให้ปริมาณข้อมูลมากขึ้นทำให้การประมวลผลแบบขนานมีประสิทธิภาพสูงขึ้นเนื่องจากค่าของ Speedup มีค่าที่สูงขึ้น โดยจะเห็นค่าจาก Speedup ของฟังก์ชัน Posterize สูงสุดที่ 2 เท่า ที่ขนาด 5000x2667 พิกเซล อย่างไรก็ตามบางฟังก์ชันที่ไม่ซับซ้อนจะมีค่า Speedup น้อยกว่า 1 เท่า เนื่องจากการประมวลผลแบบขนานโดยใช้หน่วยประมวลผลกราฟิกนั้นจะมีการส่งข้อมูลไปยังหน่วยประมวลผลกราฟิก ทำให้มีโอเวอร์เฮด (Overhead) ในการทำงานมากกว่าหน่วยประมวลผลกลาง

**6. สรุปผลการวิจัย**

ในการวิจัยครั้งนี้ได้มีการพัฒนาอัลกอริธึมในการประมวลผลภาพแบบตามลำดับ ให้เป็นการประมวลผลภาพแบบขนาน โดยใช้เว็บซีแอล จากนั้นทดสอบเวลาในการประมวลผลภาพทั้งสองแบบ โดยการประมวลผลภาพแบบขนานมีการนำเอาหน่วยประมวลผลกราฟิกร่วม ซึ่งอยู่ในหน่วยประมวลผลกลางรุ่นใหม่ เข้ามาใช้ในการประมวลผล โดยทดสอบกับอัลกอริธึมประมวลผลภาพ Brightness Inversion Desaturation Posterize Contrast และ Threshold ซึ่งผลลัพธ์ที่ได้อัลกอริธึมส่วนใหญ่จะใช้เวลาในการทำงานน้อยลง เมื่อพัฒนาให้มีการทำงานแบบขนาน โดยฟังก์ชัน Posterize ใช้เวลาในการประมวลผลภาพน้อยลง 2 เท่าที่ภาพขนาด 5000x2667 พิกเซล

การวิจัยนี้สามารถพัฒนาต่อโดยการแบ่งการประมวลผลแบบขนานไปยังหน่วยประมวลผลกลางหลายแกน (Multi-core CPU) หน่วยประมวลผลกราฟิกร่วม (Integrated GPU) หน่วยประมวลผลกราฟิกแยก (Dedicated GPU) และหน่วยประมวลผลอื่น ซึ่งการพัฒนาลักษณะนี้จะทำให้ความเร็วในการคำนวณมีค่าลดลง ซึ่งทำให้การประมวลผลภาพทางด้านเวลามีประสิทธิภาพมากยิ่งขึ้น

**7. กิตติกรรมประกาศ**

งานวิจัยนี้ได้รับการสนับสนุนโดยมหาวิทยาลัยราชภัฏสวนสุนันทา ที่สนับสนุนมอบทุนอุดหนุนการวิจัย สัญญาที่ 236/2558 ซึ่งทำให้งานวิจัยนี้สำเร็จลุล่วงไปได้ด้วยดี

**8. เอกสารอ้างอิง**

Khronos (2010). **The opencl specification, version 1.1.** ค้นเมื่อวันที่ 9 มกราคม 2557 จาก http://www.khronos.org/opencl/

NVIDIA(2009). **Nvidia opencl jumpstart guide.** ค้นเมื่อวันที่ 9 มกราคม 2557 จาก http://developer.download.nvidia.com/OpenCL/

AMD (2011). **Amd accelerated parallel processing**. ค้นเมื่อวันที่ 9 มกราคม 2557 จาก http://developer.amd.com

Piao, Xianglan, et al (2014). Efficient CPU-GPU work sharing for data-parallel JavaScript workloads. **Proceedings of the companion publication of the 23rd international conference on World wide web companion.** International World Wide Web Conferences Steering Committee.

MacWilliam, et al (2013). CrowdCL: Web-based volunteer computing with WebCL. **High Performance Extreme Computing Conference (HPEC)**, 2013 IEEE. IEEE.

Herhut, et al(2012). Parallel programming for the web. **Proceedings of the 4th USENIX conference on Hot Topics in Parallelism**. HotPar Vol. 12.

Jeon Won et al(2012), **WebCL for Hardware-Accelerated Web Applications.**

Khan Faiz et al(2014). **Using JavaScript and WebCL for Numerical Computations: A Comparative Study of Native and Web Technologies.**

Wang Jin et al(2014). ParalleJS: An Execution Framework for JavaScript on Heterogeneous Systems. **Proceedings of Workshop on General Purpose Processing Using GPUs**. ACM.

Aarnio, T. et al(2012). **WebCL working draft**. Khronos WebCL Working Group.

Cushing, Reginald, et al(2013). Distributed computing on an ensemble of browsers. **Internet Computing**, IEEE 17.5. 54-61.