

การพัฒนาระบบคัดแยกมะขามหวานด้วยเทคนิคการประมวลผลภาพ Development of Sweet Tamarind Sorting System by Image Processing Technique

ปานิสรา หาดขุนทด^{1*}, ภัทรพล อาจาอาษา² และ ธนากร แสงกุดเลาะ³

Panisara Hadkhuntod^{1*}, Phukkaraphon Ardarsa² and Thanakorn Sangkudluo³

มหาวิทยาลัยราชภัฏพิบูลสงคราม¹ มหาวิทยาลัยราชภัฏชัยภูมิ² และ มหาวิทยาลัยนเรศวร³

Pibulsongkram Rajabhat University¹, Chaiyaphum Rajabhat University² and Naresuan University³

e-mail: panisara.h@psru.ac.th¹, tiwa@cpru.ac.th², thanakorns@nu.ac.th³

Received: September 13, 2023; Revised: November 9, 2023; Accepted: November 29, 2023

บทคัดย่อ

มะขามหวานเป็นสินค้า GI ของจังหวัดเพชรบูรณ์ สามารถสร้างมูลค่าได้หลายพันล้านบาทต่อปี การกระจายมะขามหวานให้ได้ราคาดี มีความจำเป็นต้องคัดแยกคุณภาพ การแยกคุณภาพและขนาดของมะขามหวาน ณ ปัจจุบันจะใช้คนที่มีความชำนาญในการคัดแยก ซึ่งหากมีปริมาณมากก็ต้องใช้เวลาและแรงงานมาก ผู้วิจัยมีแนวคิดที่จะพัฒนาระบบคัดแยกมะขามหวานด้วยเทคนิคการประมวลผลภาพ โดยแบ่งตามมาตรฐานมะขามหวาน เป็น มะขามหวานมีตำหนิ มะขามปกติขนาด 1 มะขามปกติขนาด 2 และ มะขามปกติขนาด 3 จากภาพมะขามหวานสีทองและสีชมพู จำนวน 3,000 ภาพ มาทดสอบในอัตราส่วน 80 : 20 โดยแบ่งเป็น 5-Fold Cross Validation แล้วหาค่าเฉลี่ย โดยใช้อัลกอริทึม YOLOv8 โดยทำการทดสอบ ใน 3 รุ่น คือ n, s and m โดยเลือก รุ่น n ที่มีความเร็วสูงสุด 7.28 เฟรมต่อวินาที ค่า mAP ที่ 0.84 ค่า Precision ที่ 1.0 Recall ที่ 8.9 F1-Score ที่ 0.85 และพื้นที่ในการจัดเก็บบนดิสก์เล็กที่สุด ขนาด 7.52 เมกะไบต์ โดยการนำเข้ารูปภาพแล้วทำนายผลคุณภาพของมะขามหวานบนเว็บแอปพลิเคชัน ช่วยเกษตรกรให้คัดแยกมะขามหวานได้ถูกต้อง รวดเร็ว และมีเครื่องมือที่จะช่วยให้ผู้บริโภคเองเกิดความมั่นใจในตัวสินค้ามากขึ้น

คำสำคัญ: มะขามหวาน การคัดแยก การประมวลผลภาพ

Abstract

The sweet tamarind from Phetchabun province is classified as a Geographical Indication (GI) product and can earn billions of baht in annual revenue. It is imperative to categorise items based on their price, quality, and size. In the present day, the task of categorising sweet tamarinds needs individuals with expertise. When dealing with a substantial amount, it requires significant time and effort. The researcher proposes to create a sorting system for sweet tamarind using image processing techniques. The system will classify the sweet tamarind into categories such as sweet tamarind with flaws, normal tamarind size 1, normal tamarind size 2, and normal tamarind size 3. This will be achieved by employing YOLO version 8 with n, s, and m algorithms. The dataset consists of 3,000 photographs of sweet tamarind, specifically Srithong and Srichompoo varieties. These images were used to carry out a test with a ratio of 80:20, meaning that 80% of the images were used for training and 20% for testing. The dataset was separated into 5-Fold Cross Validation, which involves splitting the data into five subsets and doing the test several times, each

time using a different subset as the test set. The results were then averaged. Following performance testing, the researcher opted for the n model, which has a maximum speed of 7.28 frame/s, a mAP value of 0.84, a precision value of 1.0, a recall value of 8.9, an F1-Score of 0.75, and the smallest size on disc, 7.52 Mb. This model was then utilised to create a web application for predicting the quality of sweet tamarinds. This technology helps farmers in efficiently and precisely categorising sweet tamarind, while also providing customers with tools to enhance their trust and confidence in the product.

Keywords: Sweet Tamarind, Sorting, Image Processing

บทนำ

มะขามหวานเพชรบูรณ์เป็นสินค้า GI (Geographical Indication เครื่องหมายที่ใช้กับสินค้าที่มาจากแหล่งผลิตที่เฉพาะเจาะจงซึ่งคุณภาพหรือชื่อเสียงของสินค้านั้น ๆ) ที่มีชื่อเสียงเป็นที่รู้จักมาอย่างยาวนาน ด้วยลักษณะเด่นที่มีรสชาติหวานหอมเป็นเอกลักษณ์ ผักใหญ่ เนื้อหนานุ่มเหนียว มีสีสวยสม่ำเสมอ โดยในแต่ละปีมีปริมาณการผลิตสูงถึง 52,000 ตัน มีระบบควบคุมคุณภาพสินค้า GI ที่ได้มาตรฐานเป็นที่ยอมรับในระดับสากล และมีตลาดส่งออกสำคัญของไทย ได้แก่ เวียดนาม จีน และหลายประเทศในทวีปยุโรป สร้างเม็ดเงินให้ผู้ประกอบการในจังหวัดเพชรบูรณ์กว่า 3,900 ล้านบาท/ปี (กรมทรัพย์สินทางปัญญา, 2565)

จากสถานการณ์การผลิตมะขามหวาน ปีการผลิต 2562 และ 2563 พื้นที่ย่นต้น 109,849 ไร่ ผลผลิตเฉลี่ยไร่ละ 495 กิโลกรัม ราคาเฉลี่ยกิโลกรัมละ 74.68 บาท คิดเป็นมูลค่าประมาณ 3,899 ล้านบาท พื้นที่ปลูกและผลผลิตสูงสุด คือ มะขามหวานผักไค้ พันธ์สีทอง พื้นที่ย่นต้น 37,736 ไร่ ผลผลิตเฉลี่ยไร่ละ 495 กิโลกรัม ราคาเฉลี่ยกิโลกรัมละ 74.21 บาท คิดเป็นมูลค่าประมาณ 1,279 ล้านบาท อันดับสอง คือ มะขามหวานผักตรง พันธ์ศรีชมพู พื้นที่ย่นต้น 26,554 ไร่ ผลผลิตเฉลี่ยไร่ละ 495 กิโลกรัม ราคาเฉลี่ยกิโลกรัมละ 76.27 บาท คิดเป็นมูลค่าประมาณ 963 ล้านบาท (พีรพล สุธงษา, 2563)

มาตรฐานมะขามหวาน จากสำนักงานมาตรฐานสินค้าเกษตรและอาหารแห่งชาติ (2557) มีข้อกำหนดขั้นต่ำคือ (1) เป็นมะขามหวานทั้งฝักที่มีเปลือกและขั้วติดอยู่ ความยาวขั้วไม่เกิน 0.5 เซนติเมตร (2) ตรงตามพันธุ์ (3) สะอาด ปราศจากสิ่งแปลกปลอมที่มองเห็นได้ (4) เปลือกไม่แตก (5) ไม่มีศัตรูพืชที่มีผลกระทบต่อลักษณะภายนอกของฝักมะขามหวาน (6) ไม่มีร่องรอยความเสียหายที่เปลือกจากศัตรูพืชที่มีผลกระทบต่อคุณภาพของเนื้อมะขามหวาน (7) ไม่มีความผิดปกติจากความชื้นภายนอก โดยไม่รวมหยดน้ำที่เกิดหลังจากนำมะขามหวานออกจากห้องเย็น (8) ไม่มีกลิ่นแปลกปลอม และ/หรือรสชาติที่ผิดปกติ และไม่มี ความผิดปกติด้านรูปร่างและสี ไม่มีตำหนิที่เปลือกในกรณีที่มีความผิดปกติหรือตำหนิต้องมองเห็นได้ไม่เกิน 10% และไม่มีผลกระทบต่อลักษณะภายนอกคุณภาพของเนื้อมะขามหวาน คุณภาพระหว่างการเก็บรักษา และการจัดเรียงเสนอในภาชนะบรรจุ ถึงจะผ่านมาตรฐาน และแบ่งขนาดตามจำนวนข้อ มากกว่าหรือเท่ากับ 7 ข้อ คือขนาด 1, 4-6 ข้อ คือขนาด 2 และ 1-3 ข้อ คือขนาด 3 การแยกคุณภาพ และขนาด ณ ปัจจุบันจะใช้คนที่มีความชำนาญในการคัดแยก ซึ่งหากมีปริมาณมากก็ต้องใช้เวลาและแรงงานมากเช่นเดียวกัน ประกอบกับปัจจุบันผู้บริโภคมีความต้องการสินค้าที่มีคุณภาพมากขึ้น จึงจำเป็นต้องมีเครื่องมือที่จะช่วยให้ผู้บริโภคเองเกิดความมั่นใจในตัวสินค้า

YOLO (Redmon และคณะ, 2016) เป็นการใช้วิธีที่เรียกว่า “Fast Single-Shot Detection” ซึ่งจะเป็นวิธีการที่สามารถตรวจจับวัตถุได้จากการส่งผ่านรูปภาพเข้าไปในระบบเพียงครั้งเดียวโดยใช้หลักการโครงข่ายประสาทแบบคอนโวลูชัน อัลกอริทึม YOLO มีการพัฒนาอย่างต่อเนื่อง โดย YOLO มีหลายรุ่น

(Version) ซึ่งแต่ละรุ่นจะมีโครงสร้างสถาปัตยกรรมที่แตกต่างกัน มีนักวิจัยที่เคยทำการการคัดแยกมะขามหวานด้วยหลักการประมวลผลภาพเช่นเดียวกัน (Jarimopas & Jaisin, 2008) โดยการแยกมะขามหวานมีคำนิยามที่รอยแตกด้วยหลักการ Thresholding ซึ่งเป็นการตรวจหาความเข้มแสงที่แตกต่างกันบนผิว มะขามโดยผิวที่แตกที่สามารถเห็นถึงเนื้อใน สีเนื้อจะมีความเข้มแสงมากกว่าสีผิวเปลือก แต่สมรรถนะทำได้น้อยกว่าเพราะด้วยลักษณะของผิวที่กลมทำให้การวางค่อนข้างมีปัญหา จึงทำให้เกิดความล่าช้า และงานวิจัยของ ญัฐพล ชัยวิชานันท์ (2555) ในงานวิจัยนี้อธิบายถึงข้อบกพร่องในการคัดแยกผิวมะขามหวานด้วยการประมวลผลภาพโดยใช้เทคนิคการส่องสว่าง แต่เทคโนโลยีในสมัยนั้นยังสามารถแยกคำนิยามได้เพียงรอยแตกเท่านั้น ไม่สามารถแยกคำนิยามลักษณะอื่นได้ ไม่สามารถแยกขนาดของมะขามหวานตามมาตรฐานมะขามหวานได้ และความเร็วในการประมวลผลได้ช้า มีนักวิจัยรุ่นใหม่ ๆ ได้นำเทคโนโลยีการประมวลผลภาพด้วย YOLO มาใช้งาน เช่น จันธิรัก น้ำใส และคณะ (2565) ได้ทำระบบการตรวจจับคู่ด้วยวิธีการ YOLO ในกรณีตัวอย่างงูเข้าที่ฟักอาศัย ซึ่งมีความรวดเร็วและแม่นยำ ความถูกต้องเฉลี่ยมากกว่า 80% และ ชนาธิป จันทนะชาติ และคณะ (2564) ได้การตรวจหาช่องจอตรดแบบเรียลไทม์โดยใช้โครงข่ายประสาทเทียม YOLO ระบบสามารถตรวจหาช่องจอตรดและแสดงผลมีค่าร้อยละความถูกต้อง 67.13 และมีค่าเฉลี่ยความคลาดเคลื่อนร้อยละสัมบูรณ์ 32.9

งานวิจัยนี้ทำการเปรียบเทียบ อัลกอริทึม YOLO 8 Version n m และ s เพื่อหาความเหมาะสมในการพัฒนาระบบคัดแยกมะขามหวานด้วยเทคนิคการประมวลผลภาพ เพื่อเป็นแนวทางในการพัฒนาเครื่องคัดแยกมะขามหวานด้วยเทคนิคการประมวลผลภาพ เพิ่มความแม่นยำในการคัดแยก ลดจำนวนแรงงานที่ใช้ในการคัดแยก และช่วยให้ผู้บริโภคเกิดความมั่นใจในสินค้ามากยิ่งขึ้นต่อไป

วัตถุประสงค์

เพื่อพัฒนาระบบคัดแยกมะขามหวานสีทองและสีชมพูด้วยเทคนิคการประมวลผลภาพ YOLOv8

วิธีดำเนินการวิจัย

ผู้วิจัยได้ดำเนินการวิจัยโดยอาศัยขั้นตอนที่ประยุกต์จากการวิเคราะห์และออกแบบระบบตามแนวทางของ Dennis และคณะ (2012) ที่แบ่งขั้นตอนออกเป็น 4 ขั้นตอนดังนี้

ขั้นตอนที่ 1 การวางแผน (Planning) แบ่งออกเป็น 3 กระบวนการย่อย ๆ คือ

1. การวิเคราะห์ความต้องการ จากการศึกษาเอกสาร และงานวิจัยที่เกี่ยวข้อง จากข้อมูลมาตรฐานของมะขามหวาน จากสำนักงานมาตรฐานสินค้าเกษตรและอาหารแห่งชาติ (2557) มีข้อกำหนดขั้นต่ำ คือ ไม่มีตำหนิที่เปลือกในกรณีที่มีความผิดปกติหรือตำหนิต้องมองเห็นได้ไม่เกิน 10% จึงจะผ่านมาตรฐาน และแบ่งขนาดตามจำนวนข้อ มากกว่าหรือเท่ากับ 7 ข้อ คือขนาด 1, 4-6 ข้อ คือ ขนาด 2 และ 1-3 ข้อ คือ ขนาด 3 ในช่วงฤดูกาลมะขามหวานจะมีผลผลิตออกมาเป็นจำนวนมาก ต้องใช้ผู้คัดแยกเป็นจำนวนมาก และใช้เวลานาน หากเกษตรกรสามารถคัดแยกผลผลิตได้ตามเกรดและคุณภาพ จะส่งผลให้ได้ราคาดีกว่ามะขามหวานแบบละขนาด

2. การศึกษาความเป็นไปได้ จากการศึกษาความเป็นไปได้ นักวิจัยลงพื้นที่เก็บภาพมะขามหวานฝักโค้งพันธุ์สีทอง และมะขามหวานฝักตรงพันธุ์ศรีชมพู จากงานมะขามหวานเพชรบูรณ์ 2566 และจากตลาดพื้นที่จังหวัดเพชรบูรณ์ และทำการคัดแยกโดยแบ่งเป็นมะขามหวานที่ได้มาตรฐานและไม่ได้มาตรฐาน มะขามหวานที่ได้มาตรฐานแบ่งเป็น 3 ขนาด คือ ขนาด 1 ขนาด 2 และขนาด 3 ชุดละ 500 ภาพ รวม 3,000 ภาพ แบ่งข้อมูลรูปภาพเพื่อทำการเทรนและการทดสอบในอัตราส่วน 80 : 20

ตารางที่ 1

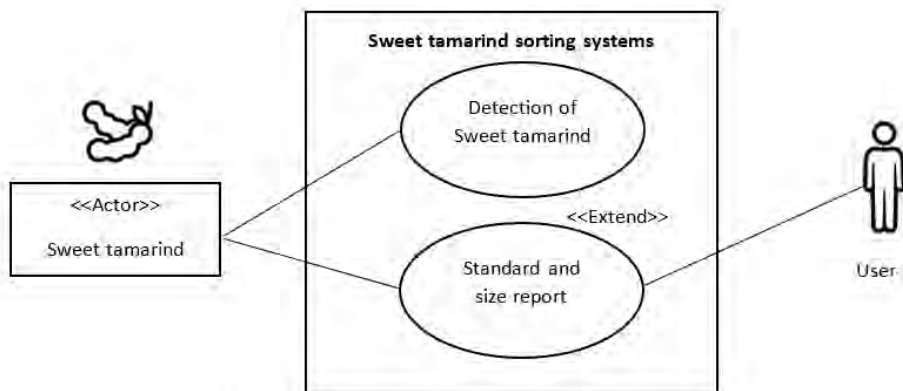
จำนวนมะขามที่ใช้ในการทดลอง

ประเภทข้อมูล	จำนวน			
	มะขามมีตำหนิ (Not)	มะขามปกติขนาด 1 (Size 1)	มะขามปกติขนาด 2 (Size 2)	มะขามปกติขนาด 3 (Size 3)
ข้อมูลฝึกสอน	400	400	400	400
ข้อมูลทดสอบ	100	100	100	100

มีการนำ 5-Fold Cross Validation มาใช้ในการแบ่งข้อมูลเพื่อฝึกสอนและทดสอบ จำนวน 5 ชุด แล้วทำการฝึกสอนจำนวน 100 epochs ทำการเปรียบเทียบโดยใช้ อัลกอริทึม YOLOv8n, m และ s เพื่อหาค่าเฉลี่ยเมื่อได้โมเดลที่มีประสิทธิภาพแล้ว จะทำการพัฒนาเว็บแอปพลิเคชันจาก Flask Framework ในการแสดงผลผ่านเว็บเบราว์เซอร์ต่อไป

3. การวางแผน ทีมวิจัยมีระยะเวลาในการศึกษาความต้องการ ศึกษาความเป็นไปได้ หาภาพถ่ายพัฒนาระบบ ทดลองใช้และแก้ไขปรับปรุงระบบ ในช่วงระยะเวลา มกราคม-พฤศจิกายน 2566

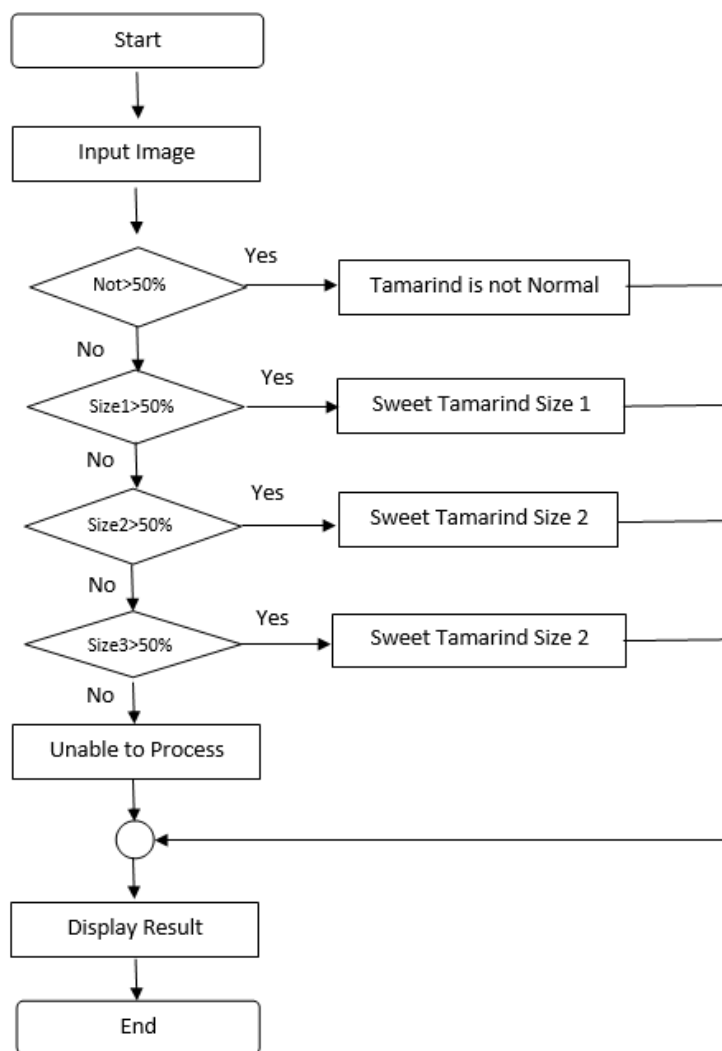
ขั้นตอนที่ 2 การวิเคราะห์ (Analysis) ขั้นตอนนี้ทีมวิจัยได้ใช้ Use Case Diagram เพื่อวิเคราะห์การทำงานของระบบดังนี้



ภาพที่ 1 แผนภาพกรณี (Use Case Diagram) ระบบคัดแยกมะขามหวานด้วยเทคนิคการประมวลผลภาพ

จากภาพที่ 1 มะขามหวานจะถูกถ่ายภาพ หรือนำเข้าโดยใช้รูปภาพมะขามหวาน หรือสามารถนำเว็บแอปพลิเคชันส่งไปที่ตัวมะขามแล้วทำนายผลได้เลย เมื่อเข้ามาในระบบจะทำการตรวจจับภาพที่ได้ หลังจากนั้นระบบจะทำนายผลคุณภาพของมะขามหวานออกมาให้ผู้ใช้งาน

ขั้นตอนที่ 3 การออกแบบ (Design) ทีมวิจัยได้นำข้อมูลจากขั้นตอนการวิเคราะห์ข้อมูลมาออกแบบการเขียนผังการปฏิบัติงาน (Flow Chart) ของระบบ



ภาพที่ 2 ผังงานการปฏิบัติงานของระบบ

จากภาพที่ 2 เมื่อนำเข้าภาพถ่ายแล้ว จะเข้าสู่กระบวนการทำนายผล เงื่อนไขที่ 1 หากขึ้น not มากกว่า 50% จะแสดงผลคำทำนายว่า “Tamarind is not Normal” เงื่อนไขที่ 2 หากขึ้น size 1 มากกว่า 50% ระบบจะแสดงผลคำทำนายว่า “Sweet Tamarind Size 1” เงื่อนไขที่ 3 หากขึ้น size 2 มากกว่า 50% ระบบจะแสดงผลคำทำนายว่า “Sweet Tamarind Size 2” เงื่อนไขที่ 4 หากขึ้น size 3 มากกว่า 50% ระบบจะแสดงผลคำทำนายว่า “Sweet Tamarind Size 3” และหากไม่ขึ้น % ใด ๆ ระบบจะแสดงผลลัพท์ “Unable to Process”

1. การนำเข้ารูปภาพ (Input Image) ผู้วิจัยได้ออกแบบกระบวนการนำเข้ารูปภาพ โดยสร้างส่วนติดต่อผู้ใช้งานด้วย Flask (เว็บเฟรมเวิร์ก) เป็นไมโครเว็บเฟรมเวิร์กที่เขียนด้วย Python จัดอยู่ในประเภทไมโครเฟรมเวิร์ก เนื่องจากไม่จำเป็นต้องใช้เครื่องมือหรือไลบรารีเฉพาะ ไม่มีไลเอร์นามธรรมของฐานข้อมูล การตรวจสอบ

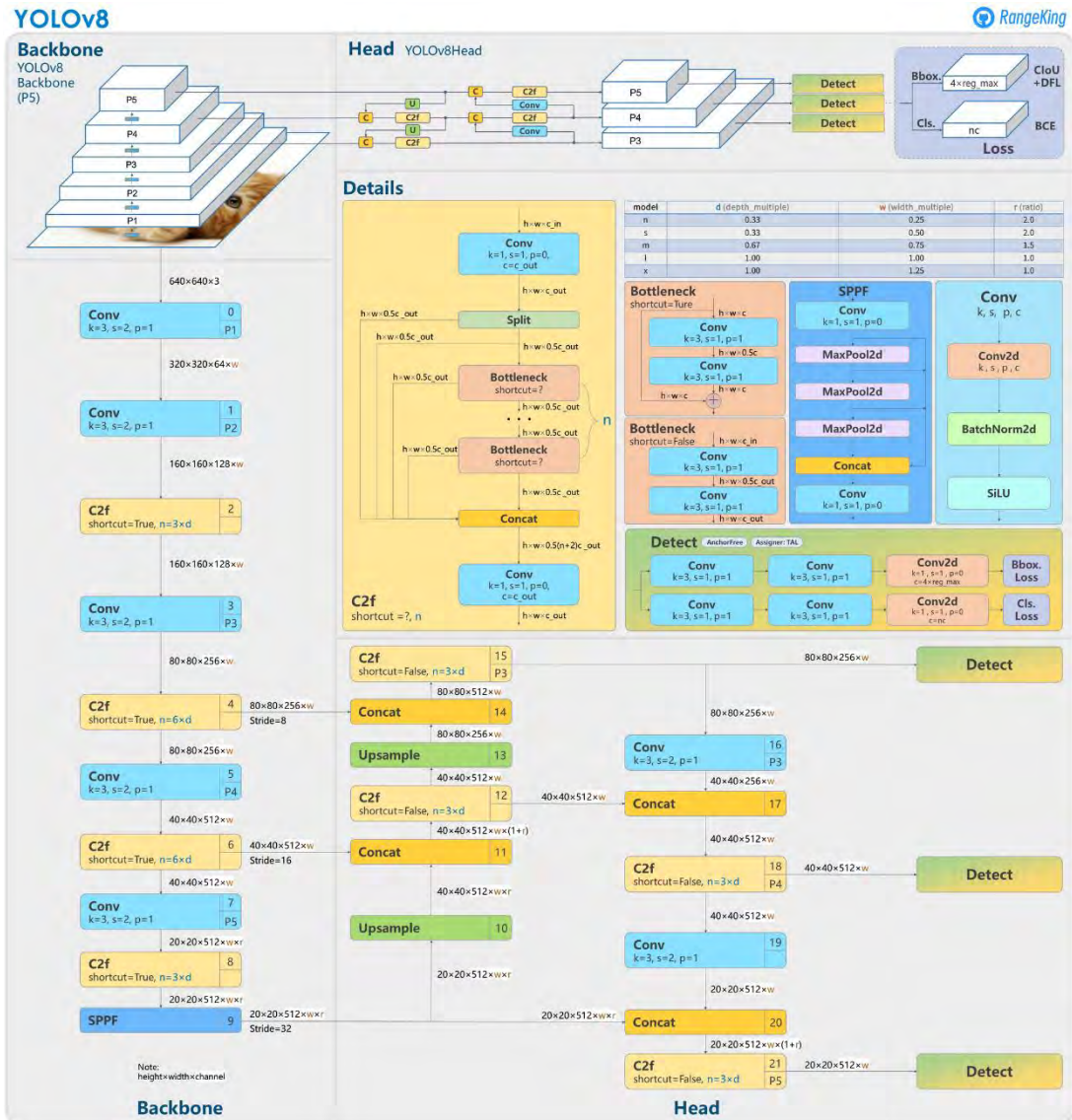
ความถูกต้องของแบบฟอร์ม หรือส่วนประกอบอื่น ๆ ที่ไลบรารีของบุคคลที่สามที่มีอยู่แล้วมีฟังก์ชันทั่วไป อย่างไรก็ตาม Flask รองรับส่วนขยายที่สามารถเพิ่มคุณสมบัติของแอปพลิเคชันได้เหมือนกับว่าถูกนำไปใช้ใน Flask เอง มีส่วนขยายสำหรับผู้ทำแผนที่เชิงวัตถุ การตรวจสอบความถูกต้องของแบบฟอร์ม การจัดการการอัปโหลด เทคโนโลยีการตรวจสอบความถูกต้องแบบเปิดต่าง ๆ และเครื่องมือที่เกี่ยวข้องกับกระบวนการทั่วไปหลายอย่าง (“Extensions,” 2010) ให้สามารถรับข้อมูลจากการถ่ายภาพด้วยสมาร์ทโฟน และคอมพิวเตอร์ที่มีกล้องถ่ายภาพ โดยออกแบบเป็นปุ่มรูปกล้องถ่ายภาพ หรือสามารถนำเว็บแอปพลิเคชันส่งไปที่ตัวมะขามแล้วทำนายผลได้เลย พร้อมคำแนะนำในการใช้โปรแกรมเบื้องต้น

2. กระบวนการประมวลผลภาพ (Image Processing) ผู้วิจัยใช้อัลกอริทึม YOLO YOLO (You Only Look Once) เป็นที่รู้จักกันในรูปแบบของโมเดลที่ใช้ประมวลผลภาพที่มีความเร็วสูงและมีขนาดเล็ก สามารถฝึกสอนได้โดยใช้ GPU (Graphics Processing Unit) ตัวเดียว ทำให้ผู้ปฏิบัติงานด้านการประมวลผลภาพใช้งานบนเทคโนโลยี Edge และ Clouds Computing หรือการประมวลผลต้นทุนต่ำได้ โดยที่ YOLO เป็นการรู้จำวัตถุที่ใช้วิธีการที่เรียกว่า “Fast Single-Shot Detection” ซึ่งจะเป็นวิธีการที่สามารถตรวจจับวัตถุได้จากการส่งผ่านรูปภาพเข้าไปในระบบเพียงครั้งเดียว โดยใช้หลักการโครงข่ายประสาทแบบคอนโวลูชัน อัลกอริทึม YOLO มีการพัฒนาอย่างต่อเนื่อง โดย YOLO มีหลายรุ่น (Version) ในอัลกอริทึม YOLO การดำเนินการจำแนกวัตถุว่าเป็นชนิดอะไร (Classification) และดำเนินการหาดำแหน่งของวัตถุ (Localization) โดยใช้กรอบล้อมวัตถุ (Bounding Box) ขั้นตอนการทำงานของ YOLO (Redmon และคณะ, 2016) จะเป็นในลักษณะการแบ่งรูปภาพเป็นส่วน ๆ หรือกริด (Grid) จากนั้นทำการเลื่อนการคำนวณไปที่ละจุดตามที่แบ่งกริดไว้ (Sliding Windows) พร้อมกับคำนวณหาว่าวัตถุจะมีอยู่จริงหรือไม่มาน้อยเพียงไร

YOLO เปิดตัวครั้งแรกในปี 2559 โดย Redmon และคณะ ในช่วงแรก ๆ (เวอร์ชัน 1-4) โดยใช้ภาษา C เฟรมเวิร์กการเรียนรู้เชิงลึกที่เรียกว่า Darknet ต่อมา Glenn Jocher ผู้พัฒนา YOLOv8 ได้ทำการปรับปรุง YOLOv3 repo ใน PyTorch (เฟรมเวิร์กการเรียนรู้เชิงลึกจาก Facebook) ให้ดีขึ้นแล้วทำการเปิดตัว YOLOv5

YOLOv5 ได้รับความนิยมทั่วโลกเนื่องจากมีโครงสร้าง Pythonic (เขียนด้วยภาษา Python) ที่มีความยืดหยุ่น และมีที่จัดเก็บด้วยเฟรมเวิร์ก PyTorch ที่สามารถแบ่งปันให้ชุมชนช่วยกันคิดค้นและปรับปรุงโมเดลใหม่ ๆ ได้อย่างรวดเร็ว

ในช่วงระหว่างปี 2564-2566 ได้มีการปรับปรุงโมเดล YOLOv5 PyTorch เช่น Scaled-YOLOv4, YOLOR และ YOLOv7 และจาก เฟรมเวิร์ก PyTorch เอง เช่น YOLX และ YOLOv6 ซึ่งมีการนำเทคนิคใหม่ ๆ มาใช้เพื่อปรับปรุงประสิทธิภาพของโมเดล (Solawetz & Francesco, 2023)



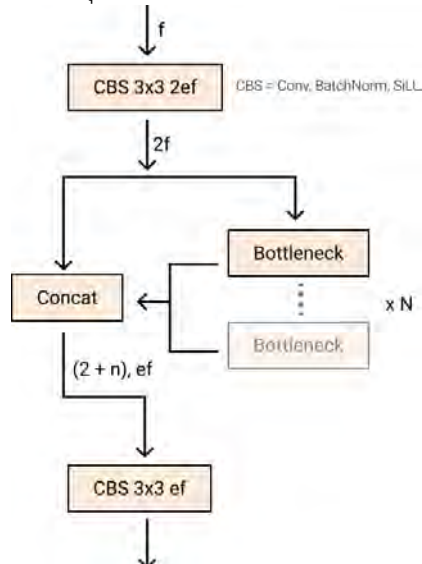
ภาพที่ 3 สถาปัตยกรรมของ YOLOv8
ที่มา (Solawetz & Francesco, 2023)

สิ่งที่เพิ่มขึ้นมาใหม่จาก YOLO รุ่นอื่น ๆ คือ

การตรวจจับแบบไม่มีจุดยึด (Anchor Free Detection) YOLOv8 เป็นรุ่นที่ไม่มีจุดยึด ซึ่งหมายความว่าระบบจะคาดการณ์จุดศูนย์กลางของวัตถุโดยตรง แทนที่จะคาดการณ์ออฟเซตจาก Anchor Box ที่รู้จัก แก้ปัญหาการกระจายตัวของ Dataset ที่เกิดจากการ Anchor Box ของรุ่นก่อนหน้านี้ ตรวจจับแบบไม่มีจุดยึดจะช่วยลดจำนวนการคาดการณ์แบบกลอง ซึ่งจะช่วยให้ความเร็ว Non-Maximum Suppression (NMS) ซึ่งเป็นขั้นตอนหลังการประมวลผลที่ซับซ้อนซึ่งจะกรองผ่านการตรวจจับผู้สมัครหลังจากการอนุมาน

Convolutions แบบใหม่ (New Convolutions) Conv 6x6 แรกจะถูกแทนที่ด้วย 3x3 โครงสร้างหลักเปลี่ยนไป และ C2f แทนที่ C3 ตามภาพที่ 4 โดยที่ "f" คือจำนวนคุณลักษณะ "e" คือ อัตราการขยายตัว และ CBS คือ บล็อกที่ประกอบด้วย Conv, BatchNorm และ SiLU

C2f คือ เอาต์พุตทั้งหมด (ชื่อสำหรับ 3x3 convs สองตัวที่มีการเชื่อมต่อที่เหลือ) จะต่อกัน ในขณะที่ C3 จะใช้เฉพาะเอาต์พุตของ Bottleneck สุดท้ายเท่านั้น



ภาพที่ 4 YOLOv8 C2f Module
ที่มา (Solawetz & Francesco, 2023)

ส่วนของ The Bottleneck จะเหมือนกับ YOLOv5 แต่ขนาดเคอร์เนลของ Conv แรกเปลี่ยนจาก 1x1 เป็น 3x3 ที่ส่วน Neck คุณลักษณะต่าง ๆ จะต่อกันโดยตรงโดยไม่ต้องบังคับให้มีขนาดช่องสัญญาณเดียวกัน ซึ่งจะช่วยลดจำนวนพารามิเตอร์และขนาดโดยรวมของเทรนเนอร์

การปิด Mosaic Augmentation ซึ่งเกี่ยวข้องกับการต่อภาพสี่ภาพเข้าด้วยกัน เพื่อบังคับให้โมเดลเรียนรู้วัตถุในตำแหน่งใหม่ การบดบังบางส่วน และเทียบกับฟิสิกส์โดยรอบที่แตกต่างกัน (Solawetz & Francesco, 2023)

โดยนำภาพมะขามหวานฝักโค้งพันธุ์สีทอง และมะขามหวานฝักตรงพันธุ์ศรีชมพู จากงานมะขามหวานเพชรบูรณ์ 2566 และจากตลาดพื้นที่จังหวัดเพชรบูรณ์ และทำการคัดแยกโดยแบ่งเป็นมะขามหวานที่ได้มาตรฐานขนาด 1 ขนาด 2 ขนาด 3 และไม่ได้มาตรฐาน ขนาด 640 x 640 พิกเซล ชุดละ 500 ภาพ รวม 3,000 ภาพ แบ่งข้อมูลเพื่อทำการเทรน 80 : 20 มีการใช้ 5-Fold Cross Validation เพื่อแบ่งข้อมูลแล้วหาค่าเฉลี่ย ทำการเปรียบเทียบโดยใช้ อัลกอริทึม YOLOv8 n, s และ m การเขียนโปรแกรมทั้งหมดโดยใช้ภาษา Python ร่วมโมเดล YOLO บน Google Colab ที่สามารถใช้ทรัพยากร GPU ได้ฟรี ช่วยแก้ปัญหาความเข้ากันได้ของไลบรารีต่าง ๆ ที่จำเป็นต้องใช้ในการเทรนโมเดล แล้วหาโมเดลที่เหมาะสมนำไปพัฒนาเป็นเว็บแอปพลิเคชัน

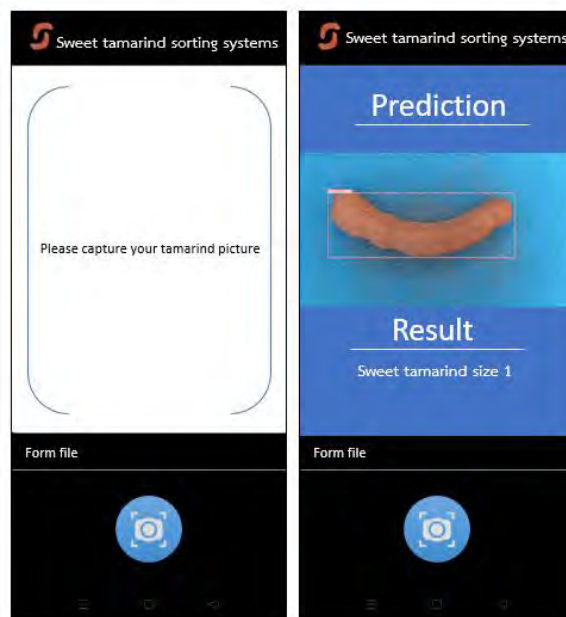
3. การกำหนดความแม่นยำ (Precision) ผู้วิจัยได้กำหนดเงื่อนไขของระบบไว้ว่าหากทำนายว่าเป็นมะขามหวานที่ได้มาตรฐาน ขนาด 1 ขนาด 2 ขนาด 3 หรือไม่ได้มาตรฐาน มากกว่าหรือเท่ากับ 50% โดยเลือกค่าทำนายที่ให้ผลดีที่สุด

4. การแสดงผล (Display) ผู้วิจัยได้ออกแบบให้แสดงผลได้ทั้งในสมาร์ทโฟน และคอมพิวเตอร์ส่วนบุคคล โดยแสดงภาพที่รับเข้า พร้อมเปอร์เซ็นต์คำทำนาย พร้อมข้อความผลการทำนายในรูปแบบข้อความ แสดงว่าเป็นมะขามหวานในกลุ่มใด ที่มิวิจัยได้แบ่งผลของการพัฒนาออกเป็นสองส่วน ได้แก่ ส่วนติดต่อผู้ใช้งาน ระบบ และประสิทธิภาพของระบบ ซึ่งในแต่ละส่วนมีรายละเอียด ดังนี้

ส่วนติดต่อผู้ใช้งานระบบ ประกอบด้วย

4.1 หน้าจอรับข้อมูล เป็นส่วนในการรับข้อมูลเข้า ทำงานผ่านเว็บเบราว์เซอร์ ในรูปแบบของเว็บแอปพลิเคชัน มีการนำเข้าข้อมูล 3 รูปแบบคือ นำเข้าโดยการถ่ายภาพ นำเข้าในรูปแบบไฟล์ หรือหรือสามารถนำเว็บแอปพลิเคชันส่งไปที่ตัวมะขามแล้วทำนายผลได้เลย

4.2 หน้าจอแสดงผล เป็นหน้าจอสำหรับทำนายภาพหลังจากการประมวลผลข้อมูล และแสดงผลลัพธ์ของข้อมูลในแบบรูปภาพและข้อความตามชื่อคลาสที่ได้จากการทำนายผล



ภาพที่ 5 ส่วนติดต่อกับผู้ใช้ของระบบคัดแยกมะขามหวานด้วยเทคนิคการประมวลผลภาพ

จากภาพที่ 5 ผู้ใช้สามารถนำเข้าข้อมูลได้ด้วยการถ่ายภาพ การนำภาพถ่ายที่ทำการถ่ายไว้แล้ว หรือสามารถนำเว็บแอปพลิเคชันส่งไปที่ตัวมะขาม นำมาประมวลผลด้วยโมเดล YOLOV8n ที่ผู้วิจัยได้นำมาใช้กับระบบ โดยใช้งานผ่านเว็บเบราว์เซอร์ เมื่อมีการนำเข้ารูปภาพแล้ว รูปภาพจะทำการแยกกลุ่ม หากโมเดลทำการทำนายว่ามะขามหวานที่ได้ Not มากกว่า 50% จะเข้าเงื่อนไขที่ 1 ระบบจะแสดงผลคำทำนายว่า “Tamarind is not Normal” หากโมเดลทำการทำนายว่ามะขามหวานที่ได้มีขนาด 1 มากกว่า 50% จะเข้าเงื่อนไขที่ 2 ระบบจะแสดงผลคำทำนายว่า “Sweet Tamarind Size 1” หากโมเดลทำการทำนายว่ามะขามหวานที่ได้มีขนาด 2 มากกว่า 50% จะเข้าเงื่อนไขที่ 3 ระบบจะแสดงผลคำทำนายว่า “Sweet Tamarind Size 2” หากโมเดลทำการทำนายว่ามะขามหวานที่ได้มีขนาด 3 มากกว่า 50% จะเข้าเงื่อนไขที่ 4 ระบบจะแสดงผลคำทำนายว่า “Sweet Tamarind Size 1” และหากไม่ขึ้น % ใด ๆ ระบบจะแสดงผลลัพธ์ “Unable to Process” พร้อม

แสดงภาพถ่ายที่ผ่านการประมวลผล โดยจะมีกรอบที่ใช้ในการทำนายพร้อมค่าเปอร์เซ็นต์ และคำอธิบายขึ้นกำกับไว้ด้วย

ขั้นตอนที่ 4 การนำไปใช้ (Implementation) พร้อมปรับแก้ไขข้อผิดพลาด โดยการทดสอบด้วยข้อมูลชุดทดสอบ แล้วหาค่าความถูกต้องจาก mAP, Precision, Recall และ F1-score จากสมการต่อไปนี้

1. mAP (Mean Average Precision) เป็นการวัดประสิทธิภาพของระบบการตรวจจับวัตถุหรือค้นหาวัตถุในคอมพิวเตอร์วิชัน (Computer Vision) โดยเฉพาะ มักใช้ในการประเมินคุณภาพของระบบตรวจจับวัตถุหรือระบบค้นหาวัตถุ โดยสรุปค่า Precision-Recall ของการตรวจจับหลาย ๆ วัตถุในรูปภาพเป็นค่าเฉลี่ย (Average) และนับระหว่าง 0 ถึง 1 โดยค่า mAP สูงถือว่าระบบมีประสิทธิภาพสูงขึ้น

2. ค่า Precision, Recall และ F1-score จากสมการ Confusion Matrix ได้พิจารณาเปรียบเทียบค่าต่าง ๆ ดังต่อไปนี้ ค่า True Positive Rate (TP Rate) ค่า False Positive Rate (FP Rate) ค่า Precision ค่า Recall ค่า f-Measure และค่า Accuracy วิเคราะห์จากตาราง Confusion Matrix (Gulshan, 2014)

ตารางที่ 2

Confusion Matrix

Predict	Actual	
	Positive	Negative
Positive	True Positive (TP)	False Positive (FP)
Negative	True Negative (TN)	False Negative (FN)

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP}) \quad (1)$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN}) \quad (2)$$

$$\text{F-Measure/F1-Score} = 2 \times \text{Precision} \times \text{Recall} / (\text{Precision} + \text{Recall}) \quad (3)$$

เมื่อ

TP (True Positive) = อัตราผลบวกจริง

TN (True negative) = อัตราผลบวกปลอม

FP (False Positive) = อัตราผลลบจริง

FN (False Negative) = อัตราผลลบปลอม

Precision = ค่า ความแม่นยำ

Recall = ค่าความไว

F-Measure = ค่าถ่วงดุล

การทดลองใช้นั้นพบว่า YOLOv8 n มีความเร็วสูงในการประมวลผลข้อมูลเร็วกว่า Version อื่น ๆ โดยที่ความถูกต้องของข้อมูลไม่แตกต่างกันมาก ผู้วิจัยจึงเลือกใช้ Version n ในการพัฒนาเว็บแอปพลิเคชัน

ผลการวิจัย

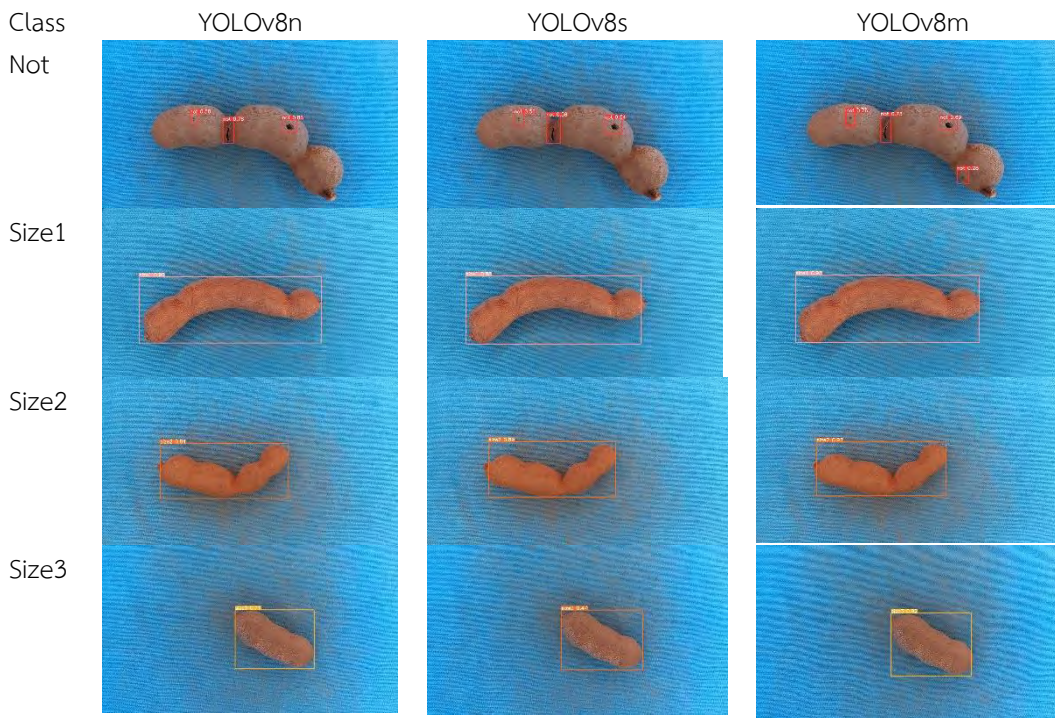
ผลของประสิทธิภาพของระบบจากกระบวนการใช้อัลกอริทึม YOLO เนื่องจากอัลกอริทึม YOLO มีการใช้งานในการประมวลผลภาพที่เน้นเรื่องความเร็ว YOLOv8 เป็น YOLO เวอร์ชันล่าสุดโดย Ultralytics ในฐานะโมเดลที่ล้ำสมัย YOLOv8 ต่อยอดความสำเร็จจากเวอร์ชันก่อนหน้า โดยนำเสนอคุณสมบัติใหม่และการปรับปรุงเพื่อประสิทธิภาพ ความยืดหยุ่น และประสิทธิภาพที่ได้รับการปรับปรุง YOLOv8 รองรับงาน Vision AI อย่างเต็มรูปแบบ รวมถึงการตรวจจับ การแบ่งส่วน การประมาณท่าทาง การติดตาม และการจัดหมวดหมู่ ความเก่งกาจนี้ทำให้ผู้ใช้สามารถใช้ประโยชน์จากความสามารถของ YOLOv8 ในแอปพลิเคชันและโดเมนที่หลากหลาย ผู้วิจัยจึงนำรุ่นต่าง ๆ ของ YOLOv8 มาทำการฝึกสอน จำนวน 100 Epochs โดยแบ่งเป็น 5-Fold Cross Validation แล้วหาค่าเฉลี่ย แล้วหาประสิทธิภาพของโมเดล ปรากฏผลดังตารางที่ 2

ตารางที่ 3

เปรียบเทียบประสิทธิภาพของแต่ละพารามิเตอร์

Parameter	FPS	mAP	Precision	Recall	F1-score	พื้นที่จัดเก็บ บนดิสก์ (MB)
YOLOv8n	7.28	0.84	1.00	0.89	0.85	7.52
YOLOv8s	3.34	0.86	1.00	0.89	0.86	30.47
YOLOv8m	2.32	0.87	1.00	0.86	0.88	60.25

จากตาราง YOLOv8n มีความเร็วในการประมวลผลสูงสุดที่ 7.28 เฟรมต่อวินาที อันดับสองคือ YOLOv8s 3.34 เฟรมต่อวินาที และอันดับสาม คือ YOLOv8m ที่ 2.32 เฟรมต่อวินาที YOLOv8m มีค่า mAP สูงที่สุดที่ 0.87 YOLOv8s มีค่า mAP ที่ 0.86 และ YOLOv8n มีค่า mAP ที่ 0.83 ทั้ง 3 อัลกอริทึมมีค่า Precision เท่ากันที่ 1.00 YOLOv8n และ YOLOv8s มีค่า Recall เท่ากันที่ 0.89 YOLOv8m มีค่า Recall ที่ 0.86 ค่า F1-score สูงสุดที่ YOLOv8m ที่ 0.88 อันดับสอง YOLOv8s ที่ 0.86 และ YOLOv8n เท่ากับ 0.85 พื้นที่ในการจัดเก็บบนดิสก์ (Size on Disk) เล็กที่สุดคือ YOLOv8n ที่ 7.52 เมกะไบต์ อันดับสองคือ YOLOv8s ที่ 30.47 เมกะไบต์ และใหญ่ที่สุดคือ YOLOv8m ที่ 60.25 เมกะไบต์



ภาพที่ 6 เปรียบเทียบภาพที่ได้จากการประมวลผลอัลกอริทึม YOLOv8

งานวิจัยนี้ผู้วิจัยเลือกใช้ YOLOv8n ที่มีความเร็วในการประมวลผลสูงสุด คือ 7.28 เฟรมต่อวินาที และมีขนาดเบาที่สุดคือ 7.52 เมกะไบต์ โดยที่มีความแตกต่างของค่า mAP, Recall and F1-score เพียงเล็กน้อย เพราะการทำงานในรูปแบบของการคัดแยกมะขามหวานจะเน้นเรื่องของความเร็วและถูกต้องเป็นหลัก หากโมเดลที่ใช้ทำนายว่ามากกว่า 50% แล้วก็จะทำนายผลได้

สรุปผลการวิจัย

ผลการออกแบบและพัฒนา ระบบคัดแยกมะขามหวานด้วยเทคนิคการประมวลผลภาพ ที่มียุทธวิธีได้แบ่งผลของการพัฒนาออกเป็นสองส่วน ได้แก่ ส่วนติดต่อผู้ใช้งานระบบ และประสิทธิภาพของระบบ

1. หน้าจอรับข้อมูล เป็นส่วนในการรับข้อมูลเข้า ทำงานผ่านเว็บเบราว์เซอร์ ในรูปแบบของเว็บแอปพลิเคชัน มีการนำเข้าข้อมูล 3 รูปแบบคือ นำเข้าโดยการถ่ายภาพ นำเข้าในรูปแบบไฟล์ หรือสามารถนำเว็บแอปพลิเคชันส่งไปที่ตัวมะขามแล้วทำนายผลได้เลย
2. หน้าจอแสดงผล เป็นหน้าจอสำหรับทำนายภาพหลังจากการประมวลผลข้อมูล และแสดงผลลัพธ์ของข้อมูลในรูปแบบรูปภาพและข้อความตามชื่อคลาสที่ได้จากการทำนายผล

ประสิทธิภาพของระบบ ผู้วิจัยเลือกใช้ YOLOv8n ที่มีความเร็วในการประมวลผลสูงสุด คือ 7.28 เฟรมต่อวินาที และมีขนาดเบาที่สุดคือ 7.52 เมกะไบต์ โดยที่มีความแตกต่างของค่า mAP, Recall and F1-score เพียง

เล็กน้อย เพราะการทำงานในรูปแบบของการคัดแยกมะขามหวานจะเน้นเรื่องความเร็วและถูกต้องเป็นหลัก หากโมเดลที่ใช้ทำนายค่ามากกว่า 50% ก็สามารถนำไปใช้ในการทำนายได้

การอภิปรายผล

สิ่งที่ค้นพบคือ YOLOv8 มีการฝึกสอนการตรวจจับล่วงหน้าไว้ในโมเดลการตรวจจับชุดข้อมูล ImageNet แต่ในชุดฝึกสอนไม่ได้มีการฝึกสอนการตรวจจับมะขามหวาน (Sweet Tamarind) ไว้ ผู้วิจัยได้ทำการศึกษาเอกสารมาตรฐานมะขามหวาน โดยเอกสารดังกล่าวได้แบ่งเป็นมะขามหวานที่ได้มาตรฐาน ออกเป็นขนาด 1 ขนาด 2 ขนาด 3 และไม่ได้มาตรฐาน ขนาด 640×640 พิกเซล ชุดละ 500 ภาพ รวม 3,000 ภาพ ทำการแบ่งข้อมูลเพื่อทำการเทรน 80 : 20 โดยแบ่งเป็น 5-Fold Cross Validation แล้วหาค่าเฉลี่ยทำการเปรียบเทียบโดยใช้ อัลกอริทึม YOLO 8 version n, s และ m ด้วย Google Colab เพื่อหาโมเดลที่เหมาะสมในการตรวจจับ โดยที่ YOLOv8n มีความเร็วในการประมวลผลสูงสุดที่ 7.28 เฟรมต่อวินาที อันดับสองคือ YOLOv8s 3.34 เฟรมต่อวินาที และอันดับสามคือ YOLOv8m ที่ 2.32 เฟรมต่อวินาที เนื่องจาก YOLOv8n มีขนาดของพารามิเตอร์น้อยที่สุด ที่ 3.2 M จึงทำให้การประมวลผลเป็นไปอย่างรวดเร็ว ในขณะที่ YOLOv8s มีขนาดของพารามิเตอร์ 11.2 M จึงทำให้การประมวลผลมีลำดับรองลงมา และ YOLOv8m ขนาดของพารามิเตอร์ 25.9 M แสดงถึงจำนวนขั้นที่มากกว่าจึงประมวลผลได้ช้ากว่านั่นเอง

YOLOv8m มีค่า mAP เท่ากันที่ 0.87 และ YOLOv8s มีค่า mAP เท่ากันที่ 0.86 อาจเป็นเพราะจำนวนพารามิเตอร์ที่มากกว่าและ YOLOv8n มีค่า mAP ที่ 0.84 แต่คุณภาพของการประมวลผลภาพโดยเฉลี่ยมีลักษณะค่อนข้างยากกว่าการแยกคลาสแบบปกติ จึงทำให้ค่า mAP มีค่าน้อย เนื่องจากผู้วิจัยออกแบบไว้ 4 คลาสจากการตรวจจับคุณภาพของมะขามหวานจะดูจาก ขนาด 1 จะมีขนาดมากกว่าหรือเท่ากับ 7 ข้อ ขนาด 2 มีขนาด 4-6 ข้อ และขนาด 3 มี 1-3 ข้อ ซึ่งในมะขามหวานข้อตรงจะดูได้ยากกว่ามะขามหวานข้ออ้อมที่สามารถมองเห็นจำนวนข้อได้ค่อนข้างชัดเจนกว่า และอีกประการหนึ่งที่ทำให้ตรวจจับได้ยากคือมะขามมีตำหนิ จะมีเปลือกไม่แตก มีศัตรูพืชที่มีผลกระทบต่อลักษณะภายนอกของฝักมะขามหวาน มีร่องรอยความเสียหายที่เปลือก มีความผิดปกติจากความชื้นภายนอก มีกลิ่นแปลกปลอม และมีความผิดปกติด้านรูปทรงและสี ซึ่งสามารถตรวจจับได้ค่อนข้างยาก เพราะมีขนาดเล็ก บางภาพอาจเห็นได้ไม่ชัดเจน ในกรณีเงื่อนไขที่ 1 จึงมีเปอร์เซ็นต์ในการตรวจจับได้ยากกว่า เงื่อนไขที่ 2-4 ทำให้แปลตเปอร์เซ็นต์ของความถูกต้องโดยรวมของโมเดลลง

ทั้ง 3 อัลกอริทึมมีค่า Precision เท่ากันที่ 1.00 ทั้งนี้อาจเพราะโมเดลทั้ง 3 สามารถตรวจจับเป็นการเปรียบเทียบ การทำนายที่ถูกต้องว่า จริง และก็เกิดขึ้นจริง (TP) กับ การทำนายว่า จริง แต่สิ่งที่เกิดขึ้น คือ ไม่จริง (FP) ได้อย่างแม่นยำคือตรวจว่าจริงได้ 100%

YOLOv8n และ YOLOv8s มีค่า Recall เท่ากันที่ 0.89 YOLOv8m มีค่า Recall ที่ 0.86 เมื่อเทียบความถูกต้องของการทำนายว่าจะเป็น “จริง” เทียบกับ จำนวนครั้งของเหตุการณ์ทั้งทำนาย และ เกิดขึ้นว่าเป็นจริงอาจเพราะ YOLOv8m มีความซับซ้อนมากกว่าจึงตรวจจับรายละเอียดได้มากทำให้เมื่อทำนายหลาย ๆ ครั้งจึงมีผลการทำนายคลาดเคลื่อนมากกว่า YOLOv8n และ YOLOv8s

ค่า F1-score สูงสุดที่ YOLOv8m ที่ 0.88 อันดับสอง YOLOv8s ที่ 0.86 และ YOLOv8n เท่ากับ 0.85 เป็นค่าเฉลี่ยแบบ harmonic mean ระหว่าง precision และ recall โดยเมื่อเฉลี่ยแล้ว YOLOv8m อาจเพราะโมเดลที่มีจำนวนพารามิเตอร์มากที่สุด จะได้ค่า F1-score สูงสุด คือทำนายได้แม่นยำที่สุด YOLOv8s มีจำนวนพารามิเตอร์รองลงมา และ YOLOv8n จำนวนพารามิเตอร์น้อยที่สุด จึงทำให้ค่าประสิทธิภาพมีค่าตามลำดับ

พื้นที่ในการจัดเก็บบนดิสก์เล็กที่สุดคือ YOLOv8n ที่ 7.52 เมกะไบต์ อันดับสองคือ YOLOv8s ที่ 30.47 เมกะไบต์ และใหญ่ที่สุดคือ YOLOv8m ที่ 60.25 เมกะไบต์ เมื่อทำการประมวลผลโมเดลแล้วจะได้ขนาดของ

โมเดลตามลำดับ YOLOv8m มีขนาดเป็นสองเท่าของ YOLOv8s และ YOLOv8s มีขนาดมากกว่า 9 เท่าของ YOLOv8n

งานวิจัยนี้ผู้วิจัยเลือกใช้ YOLOv8n ที่มีความเร็วในการประมวลผลสูงสุด คือ 7.28 เฟรมต่อวินาที และมีขนาดเบาที่สุดคือ 7.52 เมกะไบต์ ค่า mAP 0.84 โดยที่มีความแตกต่างของค่า mAP, Recall and F1-score จากโมเดลอื่น ๆ เพียงเล็กน้อย เพราะการทำงานในรูปแบบของการคัดแยกมะขามหวานจะเน้นเรื่องของความเร็วและถูกต้องเป็นหลัก หากโมเดลที่ใช้ทำนายว่ามากกว่า 50% แล้วก็จะทำนายผลได้ สอดคล้องกับงานวิจัย จันธิรัก น้ำใส และคณะ (2565) ได้ทำการระบบตรวจจำแนกด้วยวิธีการ YOLO ในกรณีตัวอย่างเข้าที่พักอาศัย ซึ่งมีความรวดเร็วและแม่นยำ ความถูกต้องเฉลี่ยมากกว่า 80% และ ชนาธิป จันทนะชาติ และคณะ (2564) ได้ทำการตรวจหาช่องจอตลอดแบบเรียลไทม์โดยใช้โครงข่ายประสาทเทียม YOLO ระบบสามารถตรวจหาช่องจอตลอดและแสดงผลมีค่าร้อยละความถูกต้อง 67.13 และมีค่าเฉลี่ยความคลาดเคลื่อนร้อยละสมบูรณ์ 32.9 ที่ใช้ YOLO ในการพัฒนาระบบเช่นเดียวกัน

หลังจากคัดเลือกโมเดลแล้ว นำโมเดลมาพัฒนาระบบคัดแยกมะขามหวานด้วยเทคนิคการประมวลผลภาพ โดยผู้ใช้งานสามารถนำเข้าข้อมูลได้ด้วยการถ่ายภาพ การนำภาพถ่ายที่ทำการถ่ายไว้แล้ว หรือสามารถนำเว็บแอปพลิเคชันส่องไปที่ตัวมะขามแล้วนำมาประมวลผลด้วยโมเดล YOLOv8n ใช้งานผ่านเว็บแอปพลิเคชัน รูปภาพจะทำการแยกกลุ่ม หากโมเดลทำการทำนายว่ามะขามหวานที่ได้ Not มากกว่า 50% คือมะขามหวานมีตำหนิจะมีเปลือกไม่แตก มีศัตรูพืชที่มีผลกระทบต่อลักษณะภายนอกของฝักมะขามหวาน มีร่องรอยความเสียหายที่เปลือก มีความผิดปกติจากความขึ้นภายนอก มีความผิดปกติด้านรูปทรงและสี จะเข้าเงื่อนไขที่ 1 ระบบจะแสดงผลคำทำนายว่า “Tamarind is not Normal” หากโมเดลทำการทำนายว่ามะขามหวานที่ได้มีขนาด 1 มากกว่า 50% คือ มีจำนวนข้อมากกว่าหรือเท่ากับ 7 ข้อ โดยไม่มีตำหนิ จะเข้าเงื่อนไขที่ 2 ระบบจะแสดงผลคำทำนายว่า “Sweet Tamarind Size 1” หากโมเดลทำการทำนายว่ามะขามหวานที่ได้มีขนาด 2 มากกว่า 50% คือ มีจำนวนข้อ 4-6 ข้อ โดยไม่มีตำหนิ จะเข้าเงื่อนไขที่ 3 ระบบจะแสดงผลคำทำนายว่า “Sweet Tamarind Size 2” หากโมเดลทำการทำนายว่ามะขามหวานที่ได้มีขนาด 3 มากกว่า 50% คือ มีจำนวนข้อ 1-3 ข้อ โดยไม่มีตำหนิ จะเข้าเงื่อนไขที่ 4 ระบบจะแสดงผลคำทำนายว่า “Sweet Tamarind Size 1” และหากไม่ขึ้น % ใด ๆ ระบบจะแสดงผลลัพธ์ “Unable to Process” พร้อมแสดงภาพถ่ายที่ผ่านการประมวลผล โดยจะมีกรอบที่ใช้ในการทำนายพร้อมค่าเปอร์เซ็นต์ และคำอธิบายขึ้นกำกับไว้ด้วย สอดคล้องกับงานวิจัยของ Jarimopas และ Jaisin (2008) ที่มีการคัดแยกมะขามหวานด้วยหลักการประมวลผลภาพเช่นเดียวกัน โดยการแยกมะขามหวานมีตำหนิจากรอยแตกด้วยหลักการ Thresholding (Sonka และคณะ, 2015) ซึ่งเป็นการตรวจหาความเข้มแสงที่แตกต่างกันบนฝักมะขาม โดยฝักแตกที่สามารถเห็นถึงเนื้อใน สีเนื้อจะมีความเข้มแสงมากกว่าสีผิวเปลือก การคัดแยกทั้งหมดถูกทดสอบทั้งในแบบฝักมะขามอยู่ร่วมกับที่และแบบมีการเคลื่อนที่ แสดงให้เห็นว่าอุปกรณ์คัดแยกมะขามหวานพันธุ์สีชมพูให้ Ew, Cr, Q เป็น 94.3%, 5.7 และ 1491 ฝัก/ชม. ในขณะที่ พันธุ์สีทองให้ Ew, Cr, Q เป็น 89.8%, 10.2 และ 1517 ฝัก/ชม. พันธุ์ศรีชมพูให้ประสิทธิภาพและอัตราส่วนของสิ่งเจือปนที่ดีกว่า พันธุ์สีทอง เพราะว่าลักษณะทางกายภาพของพันธุ์นั้นไม่มีส่วนหยักส่วนเว้าเหมือนพันธุ์สีทอง เครื่องมือคัดแยกสามารถประมวลได้ถูกต้องกว่า แต่สมรรถนะจะทำได้น้อยกว่าเพราะว่าด้วยลักษณะของฝักที่กลมทำให้การวางค้อนข้างมีปัญหา จึงทำให้เกิดความล่าช้า และสอดคล้องกับงานวิจัยของ ณัฐพล ชัยทวีชนานนท์ (2555) ในงานวิจัยนี้อธิบายถึงข้อบกพร่องในการคัดแยกฝักมะขามหวานด้วยการประมวลผลภาพโดยใช้เทคนิคการส่องสว่าง ในการทดลองเพื่อหาระดับความสว่างมี 2 ขั้นตอน ขั้นตอนที่ 1 ทดสอบด้วยตัวอย่างฝักมะขามหวานศรีชมพู จำนวน 30 ผล โดยมีระดับความสว่างตั้งแต่ 0.10 ถึง 0.30 ผลจากการสัมผัสกับเครื่องหมายที่เหมาะสมในช่วง 0.15 ถึง 0.20

ขั้นตอนที่ 2 ทดสอบด้วยตัวอย่างฝักมะขามหวานศรีชมพูจำนวน 100 ฝัก โดยมีระดับความสว่างตั้งแต่ 0.15 ถึง 0.20 ผลจากการสัมผัสกับเครื่องหมายที่เหมาะสมในช่วง 0.15 ถึง 0.16 แต่เทคโนโลยีในสมัยนั้นยังสามารถแยกตำหนิได้เพียงรอยแตกเท่านั้น ไม่สามารถแยกตำหนิลักษณะอื่นได้ ไม่สามารถแยกขนาดของมะขามหวานตามมาตรฐานมะขามหวานได้ และความเร็วในการประมวลผลยังช้ากว่าระบบที่ผู้วิจัยได้พัฒนาขึ้น

เอกสารอ้างอิง

- กรมทรัพย์สินทางปัญญา. (2565, 14 มีนาคม). คนไทยเฮ! “มะขามหวานเพชรบูรณ์” สินค้า GI ไทย ได้รับการคุ้มครองในประเทศเวียดนาม “สินิทย” รมช.พาณิชย์ ชี้ช่วยขยายมูลค่าการส่งออกของไทยพร้อมลุยจัด GI ในต่างประเทศเพิ่มเติม. <https://www.ipthailand.go.th/th/dip-news/item/gi20220314.html>
- จันธิรัก น้ำใส, ศรศักดิ์ ทาวงษ์, ภาณุ ดงทอง, ธนะพัฒน์ เชี่ยวชาญวัฒนา และศรุต อัสวเรืองสุข. (2565). ระบบตรวจจับด้วยวิธีการ YOLO ในกรณีตัวอย่างงูเข้าที่ฟักอาศัย. *วารสารวิชาการวิศวกรรมศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏอุดรธานี*, 1(1), 59-72.
- ชนาธิป จันทนะชาติ, สวรรคพร ดาวรีรัมย์ และเก่ง จันทน์นวล. (2564). การตรวจหาช่องจอตรูปแบบเรียลไทม์โดยใช้โครงข่ายประสาทเทียม YOLO. ใน *การประชุมวิชาการระดับชาติการจัดการเทคโนโลยีและนวัตกรรม ครั้งที่ 7* (น.1679-1687). คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยราชภัฏมหาสารคาม.
- ณัฐพล ชัยทวิชานันท์. (2555). *การตรวจสอบฝักมะขามหวานเสียโดยวิธีการประมวลผลภาพ*. มหาวิทยาลัยราชภัฏเพชรบูรณ์.
- พีรพล สุธงษา. (2563). *สถานการณ์การผลิตพืชจังหวัดเพชรบูรณ์ ปี 2563*. กลุ่มยุทธศาสตร์และสารสนเทศ สำนักงานเกษตรจังหวัดเพชรบูรณ์.
- สำนักงานมาตรฐานสินค้าเกษตรและอาหารแห่งชาติ. (2557). *มาตรฐานสินค้าเกษตร มกษ. 23-2557: มะขามหวาน*. https://www.acfs.go.th/standard/download/SWEET_TAMARIND.pdf
- Dennis, A., Wixom, B. H., & Roth, R. M. (2012). *System analysis and design* (5th ed.). John Wiley & Sons.
- Extensions. (2010). Flask. <https://flask.palletsprojects.com/en/3.0.x/extensions/>
- Gulshan, K. (2014). Evaluation metrics for intrusion detection systems-A study. *International Journal of Computer Science and Mobile Applications*, 2(11), 11-17.
- Jarimopas, B., & Jaisin, N. (2008). An experimental machine vision system for sorting sweet tamarind. *Journal of Food Engineering*, 89(3), 291-297.
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection, In *Proceedings 29th IEEE Conference on Computer Vision and Pattern Recognition CVPR 2016* (pp. 779-788). IEEE X plore.<https://doi.org/10.1109/CVPR.2016.91>
- Solawetz, J., & Francesco. (2023, January 11). *What is YOLOv8? The ultimate guide*. Roboflow. <https://blog.roboflow.com/whats-new-in-yolov8/>
- Sonka, M., Hlavac, V., & Boyle, R. (2015). *Image processing, analysis, and machine vision* (4th ed.). Cengage Learning.