การพัฒนาและเปรียบเทียบประสิทธิภาพของโมเดล YOLOv8 ในการคัดแยกประเภทของพลาสติกสำหรับการรีไซเคิลด้วยการประมวลผลจากภาพถ่ายดิจิทัล

ภูชิต พญาพรหม

สารนิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
บัณฑิตศึกษา สาขาเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีไทย-ญี่ปุ่น
ปีการศึกษา 2568

DEVELOPMENT AND COMPARISON OF YOLOV8 MODELS FOR PLASTIC CLASSIFICATION VIA DIGITAL IMAGE PROCESSING

Poochit Payaprom

A Independent Study Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Information Technology
Graduate Studies
Thai-Nichi Institute of Technology
Academic Year 2025

หัวข้อสารนิพนธ์	การพัฒนาและเปรียบเทียบประสิทธิภาพของโมเดล YOLOv8 ในการคัดแยกประเภทของพลาสติกสำหรับการรีไซเคิลด้วยการประมวลผลจากภาพถ่ายดิจิทัล
โดย	ภูชิต พญาพรหม
สาขาวิชา	เทคโนโลยีสารสนเทศ
อาจารย์ที่ปรึกษาสารนิพนธ์	ดร.ภูวดล ศิริกองธรรม

บัณฑิตศึกษา สถาบันเทคโนโลยีไทย-ญี่ปุ่น อนุมัติให้นับสารนิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... รองอธิการบดีฝ่ายวิชาการ

(รองศาสตราจารย์ ดร.วรากร ศรีเชวงทรัพย์)

วันที่.....เดือน.....พ.ศ.....

คณะกรรมการสอบสารนิพนธ์

..... ประธานกรรมการ

(ว่าที่ร้อยตรี ดร.พิชิตชัย คำอินทร์)

..... กรรมการ

(ดร.อภิชญา นิ้มคุ้มภัย)

..... อาจารย์ที่ปรึกษาสารนิพนธ์

(ดร.ภูวดล ศิริกองธรรม)

ภูชิต พญาพรหม : การพัฒนาและเปรียบเทียบประสิทธิภาพของโมเดล YOLOv8 ในการคัดแยกประเภทของพลาสติกสำหรับการรีไซเคิลด้วยการประมวลผลจากภาพถ่ายดิจิทัล. อาจารย์ที่ปรึกษา : ดร. ภูวดล ศิริกองธรรม, 80 หน้า.

ปัจจุบันปัญหาขยะพลาสติกมีผลกระทบอย่างมากในประเทศไทยและมีแนวโน้มที่จะเพิ่มขึ้นอย่างมากในทุกๆปี นักวิจัยจึงได้เล็งเห็นว่า ระบบการคัดแยกพลาสติกอย่างมีประสิทธิภาพ จึงมีความสำคัญในการนำพลาสติกกลับมาใช้ใหม่และลดผลกระทบต่อสิ่งแวดล้อม ซึ่งปัจจุบันในประเทศไทยพบว่ายังใช้มีการคัดแยกแบบดั้งเดิมที่ใช้แรงงานคน ซึ่งมีข้อจำกัดด้านทักษะของคนงานทำให้การคัดแยกไม่เกิดประสิทธิภาพทำให้พลาสติกไปปะปนกับขยะมูลฝอยชุมชน ส่งผลให้พลาสติกจำนวนมากถูกกำจัดทิ้ง ฝังกลบหรือรั่วไหลลงแม่น้ำและไหลลงทะเลแทนที่จะถูกนำกลับมาใช้ใหม่ งานวิจัยนี้ได้นำเสนอการพัฒนาระบบคัดแยกประเภทของพลาสติกเพื่อการรีไซเคิลจากภาพถ่ายด้วยวิธีการเรียนรู้เชิงลึก ด้วยชุดข้อมูลของพลาสติกประเภทต่างๆ ก็คือ PET, HDPE, PVC, LDPE, PP, PS และ Other จากนั้นทำการฝึกฝนแบบจำลองด้วยเทคนิคเรียนรู้เชิงลึกด้วยอัลกอริทึม YOLO (You Only Look Once) โดยผู้วิจัยเลือกเป็น YOLOv8 เพื่อให้ AI สามารถคัดแยกประเภทของพลาสติกได้อย่างแม่นยำและรวดเร็ว จากผลการทดลองพบว่าแบบจำลอง YOLO แต่ละรุ่นที่พัฒนาขึ้นมามีประสิทธิภาพในการคัดแยกประเภทพลาสติกที่สูง ดังนั้นการวิจัยนี้ทำการทดสอบประสิทธิภาพของแบบจำลอง YOLOv8 แต่ละรุ่นและพบว่า YOLOv8n ให้ประสิทธิภาพสูงสุดในด้านความแม่นยำและความรวดเร็ว ผลการวิจัยยังชี้ให้เห็นถึงศักยภาพของ AI ในการจัดการกับปัญหาขยะพลาสติก และจะเป็นแนวทางในการพัฒนาเทคโนโลยีเพื่อสิ่งแวดล้อมต่อไป

บัณฑิตศึกษา
สาขาวิชา เทคโนโลยีสารสนเทศ
ปีการศึกษา 2568

ลายมือชื่อนักศึกษา .....
ลายมือชื่ออาจารย์ที่ปรึกษา .....

POOCHIT PAYAPROM : DEVELOPMENT AND COMPARISON OF YOLOV8 MODELS FOR PLASTIC CLASSIFICATION VIA DIGITAL IMAGE PROCESSING. ADVISOR : DR PUWADOL SIRIKONGTUM, 80 PP.

Currently plastic problem has caused a lot of problems and has a lot of effected to Thailand and tends to have more plastic waste every year. Researcher points that having effective plastic waste management is essential by reusing plastic which can help the environment.

Now Thailand still uses the legacy management to manage the plastic waste by using human which leads to inefficient result plastic is include with local waste which were burned, buried or leak to the sea not be able to reuse them.

This research proposes the solution of plastic management to recycle plastic by using image using Deep learning with the data of PET, HDPE, PVC, LDPE, PP, PS and Other then training the model using YOLO (You Only Look Once). I use YOLOv8 so AI can separate type of plastic precisely and speedy. From the experiment's result each YOLO version has potential to separate the type of plastic this research tested the performance of YOLOv8 with different version founded that YOLOv8n has given the best result in terms of accuracy and speed. This research shows that the potential of AI can help with plastic waste management and can lead to innovation for environment.

Graduate Studies Student's Signature.....

Field of Study Information Technology Advisor's Signature.....

Academic Year 2025

กิตติกรรมประกาศ

การทำสารนิพนธ์นี้จะสำเร็จลุล่วงไปได้ด้วยดีนั้น ผู้วิจัยขอกราบขอบพระคุณในความกรุณาของ ดร.ภูวดล ศิริกองธรรม ซึ่งได้สละเวลาในการให้คำแนะนำ คำปรึกษา การสนับสนุน รวมถึงตรวจสอบข่อบกพร่องทุกขั้นตอนและกำลังใจเต็มเปี่ยมตลอดระยะเวลาในการทำสารนิพนธ์ฉบับนี้

ทั้งนี้ผู้วิจัยขอกราบขอบพระคุณคณะกรรมการสอบทุกท่าน ได้แก่ ว่าที่ร้อยตรี ดร. พิชิตชัย คำอินทร์ และ ดร. อภิชญา นิ้มคุ้มภัย ที่ได้กรุณาให้ข้อเสนอแนะที่เป็นประโยชน์และช่วยตรวจสอบชี้ข้อบกพร่องของสารนิพนธ์ฉบับนี้ด้วยความเอาใจใส่จนทำให้สารนิพนธ์ฉบับนี้สำเร็จและมีความสมบูรณ์

เหนือสิ่งอื่นใดผู้วิจัยขอกราบขอบพระคุณบิดามารดาของผู้วิจัยที่คอยให้กำลังใจ และให้การสนับสนุนในทุกๆ ด้านอย่างที่สุดเสมอมา

ผู้วิจัยหวังเป็นอย่างยิ่งว่า สารนิพนธ์ฉบับนี้จะเกิดประโยชน์ต่อผู้ที่มีความสนใจ สามารถนำไปต่อยอด องค์ความรู้เพื่อประยุกต์ใช้งานกับหน่วยงานที่ดำเนินการเกี่ยวกับการคัดแยกขยะเพื่อให้เกิดคุณค่าต่อทางสังคมต่อไป

ภูชิต พญาพรหม

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง
บทคัดย่อภาษาอังกฤษ ..... จ
กิตติกรรมประกาศ ..... ฉ
สารบัญ ..... ช
สารบัญตาราง ..... ฌ
สารบัญรูป ..... ญ

บทที่

1 บทนำ 1
1.1 ความเป็นมาและความสำคัญของปัญหา 1
1.2 วัตถุประสงค์ของการศึกษา 2
1.3 ขอบเขตการวิจัย 2
1.4 ประโยชน์ที่คาดว่าจะได้รับ 3

2 เอกสารและงานวิจัยที่เกี่ยวข้อง 4
2.1 ประเภทของพลาสติกและความสำคัญของการรีไซเคิลพลาสติก 4
2.2 เทคโนโลยีการคัดแยกพลาสติกในปัจจุบัน 7
2.3 การเรียนรู้เชิงลึก (Deep Learning) และการประยุกต์ใช้ในการประมวลภาพ 8
2.4 อัลกอริทึม YOLO (You Only Look Once) 9
2.5 งานวิจัยที่เกี่ยวข้องกับการคัดแยกพลาสติกด้วยวิธีการเรียนรู้เชิงลึก 14
2.6 ช่องว่างในงานวิจัย (Research Gap) 16

3 ระเบียบและวิธีดำเนินการวิจัย 18
3.1 การรวบรวมรูปภาพลาสติกและการแบ่งเป็นชุดข้อมูลสำหรับนำไปใช้ 20
3.2 การสร้างแบบจำลอง (Model) 21
3.3 การประเมินผลโมเดล 21
3.4 การแปลง Confusion Matrix ที่เป็นภาพให้กลายเป็นข้อมูล 23
3.5 การนำค่าตัวเลขที่ได้จากวิธี Image Processing ไปคำนวนหาค่าประสิทธิภาพ 24

สารบัญ (ต่อ)

บทที่ หน้า

3.6 ทำการเปรียบเทียบค่าประสิทธิภาพกับโมเดล YOLOv8 24
3.7 ประเมินและวิเคราะห์ผลการทดลอง 25

4 ผลการวิจัย 26
4.1 การประเมินด้วยค่าตัวชี้วัด 26
4.2 การประเมินด้วยผลการทดสอบจากการนำเข้ารูปภาพ 33

5 บทสรุป อภิปราย และข้อเสนอแนะ 49
5.1 สรุปผลการวิจัย 49
5.2 อภิปรายผลวิจัย 50
5.3 ปัญหาและอุปสรรคในการทำวิจัย 51
5.4 ข้อเสนอแนะงานวิจัย 52

บรรณานุกรม 54

ภาคผนวก 58
ภาคผนวก ก. โค้ดระบบคัดแยกประเภทของพลาสติกฯ 59

ประวัติย่อผู้วิจัย 80

TNI
THAI - NICHI INSTITUTE OF TECHNOLOGY

สารบัญตาราง

ตาราง หน้า

3.1 ขั้นตอนงานวิจัย 18

3.2 แผนการดำเนินงานวิจัย 19

3.3 คำอธิบายหลักการของ Confusion Matrix 22

4.1 ผลการทดลองก่อนที่จะทำการประเมินประสิทธิภาพ 32

4.2 ผลการทดลองหลังจากทำการประเมินประสิทธิภาพและเรียงลำดับ 33

4.3 ผลเปรียบเทียบค่าความแม่นยำของแต่ละโมเดลที่ได้ประมวลผลจากภาพพลาสติกแต่ละชนิด 48

สถาบันเทคโนโลยีไทย-ญี่ปุ่น THAI - NICHI INSTITUTE OF TECHNOLOGY TNI

สารบัญรูป

รูป		หน้า
2.1	องค์ประกอบหลักของ Convolutional Neural Network: CNN	8
2.2	กล่อง Anchor Boxes	10
2.3	YOLOv8 ทุกรุ่น	11
2.4	ขั้นตอนการทำงานของ YOLO	12
2.5	ภาพตัวอย่างระบบการตรวจจับทางม้าลายที่ซีดจางโดยใช้การตรวจจับวัตถุ	14
2.6	การพัฒนาระบบคัดแยกขยะรีไซเคิลด้วยเทคโนโลยีไอโอที	15
2.7	การคัดแยกประเภทขวดน้ำ	15
2.8	หลักการทำงานของระบบ	16
3.1	ภาพรวมการออกแบบทางด้านซอฟต์แวร์ (Software Design)	19
3.2	แพลตฟอร์ม Roboflow	20
3.3	การแบ่งคลาสของประเภทพลาสติก	21
3.4	ผลลัพธ์ของการใช้โมเดล	21
3.5	ภาพ Confusion Matrix ที่ได้จากจาก Validate	23
3.6	ภาพ Confusion Matrix ที่ผ่านวิธีการ OCR	24
3.7	ผลลัพธ์ที่ได้จากการคำนวนค่าที่ได้จาก OCR	24
4.1	Confusion Matrix ของการทดสอบด้วยโมเดล YOLOv8n	27
4.2	ค่า Confusion Matrix ของโมเดล YOLOv8n ที่ผ่านการประมวลผลแล้ว	27
4.3	Confusion Matrix ของการทดสอบด้วยโมเดล YOLOv8s	28
4.4	Confusion Matrix ของโมเดล YOLOv8s ที่ผ่านการประมวลผลแล้ว	28
4.5	Confusion Matrix ของการทดสอบด้วยโมเดล YOLOv8m	29
4.6	ค่า Confusion Matrix ของโมเดล YOLOv8m ที่ผ่านการประมวลผลแล้ว	29
4.7	Confusion Matrix ของการทดสอบด้วยโมเดล YOLOv8l	30
4.8	ค่า Confusion Matrix ของโมเดล YOLOv8l ที่ผ่านการประมวลผลแล้ว	30
4.9	Confusion Matrix ของการทดสอบด้วยโมเดล YOLOv8x	31
4.10	ค่า Confusion Matrix ของโมเดล YOLOv8x ที่ผ่านการประมวลผลแล้ว	31
4.11	ภาพพลาสติก PET ที่ใช้ในการทดสอบประสิทธิภาพของโมเดล YOLOv8	34
4.12	ผลลัพธ์การทำนายพลาสติก PET ด้วยโมเดล YOLOv8 ทุกรุ่น	35

สารบัญรูป (ต่อ)

รูป		หน้า
4.13	กราฟเส้นค่าความแม่นยำในการทำนายพลาสติก PET ของโมเดลแต่ละรุ่น	35
4.14	ภาพพลาสติก HDPE ที่ใช้ในการทดสอบประสิทธิภาพของโมเดล YOLOv8	36
4.15	ผลลัพธ์การทำนายพลาสติก HDPE ด้วยโมเดล YOLOv8 ทุกรุ่น	37
4.16	กราฟเส้นค่าความแม่นยำในการทำนายพลาสติก HDPE ของโมเดลแต่ละรุ่น	37
4.17	ภาพพลาสติก PVC ที่ใช้ในการทดสอบประสิทธิภาพของโมเดล YOLOv8	38
4.18	ผลลัพธ์การทำนายพลาสติก PVC ด้วยโมเดล YOLOv8 ทุกรุ่น	39
4.19	กราฟเส้นค่าความแม่นยำในการทำนายพลาสติก PVC ของโมเดลแต่ละรุ่น	39
4.20	ภาพพลาสติก LDPE ที่ใช้ในการทดสอบประสิทธิภาพของโมเดล YOLOv8	40
4.21	ผลลัพธ์การทำนายพลาสติก LDPE ด้วยโมเดล YOLOv8 ทุกรุ่น	41
4.22	กราฟเส้นค่าความแม่นยำในการทำนายพลาสติก LDPE ของโมเดลแต่ละรุ่น	41
4.23	ภาพพลาสติก PP ที่ใช้ในการทดสอบประสิทธิภาพของโมเดล YOLOv8	42
4.24	ผลลัพธ์การทำนายพลาสติก PP ด้วยโมเดล YOLOv8 ทุกรุ่น	43
4.25	กราฟเส้นค่าความแม่นยำในการทำนายพลาสติก PP ของโมเดลแต่ละรุ่น	43
4.26	ภาพพลาสติก PS ที่ใช้ในการทดสอบประสิทธิภาพของโมเดล YOLOv8	44
4.27	ผลลัพธ์การทำนายพลาสติก PS ด้วยโมเดล YOLOv8 ทุกรุ่น	45
4.28	กราฟเส้นค่าความแม่นยำในการทำนายพลาสติก PS ของโมเดลแต่ละรุ่น	45
4.29	ภาพพลาสติก O(Other) ที่ใช้ในการทดสอบประสิทธิภาพของโมเดล YOLOv8	46
4.30	ผลลัพธ์การทำนายพลาสติก O(Other) ด้วยโมเดล YOLOv8 ทุกรุ่น	47
4.31	กราฟเส้นค่าความแม่นยำในการทำนายพลาสติก O(Other) ของโมเดลแต่ละรุ่น	47

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบัน มีปริมาณขยะมูลฝอยที่เกิดขึ้นในประเทศไทยเพิ่มขึ้นอย่างมากในทุกๆปี ซึ่งในปี 2567 ประเทศไทยมีขยะมูลฝอยเกิดขึ้นประมาณ 27.20 ล้านตันหรือประมาณ 74,529 ตัน/วัน โดยคิดเป็นอัตราการเกิดขยะมูลฝอยเมื่อเทียบกับจำนวนประชากรในปี 2567 จำนวน 64,953,661 คน เฉลี่ยเท่ากับ 1.15 กิโลกรัม/คน/วัน และเมื่อเปรียบเทียบกับปริมาณขยะมูลฝอยในปีก่อนหน้านี้ปี 2566 ซึ่งมีขยะมูลฝอยเกิดขึ้นประมาณ 26.95 ล้านตัน พบว่ามีปริมาณที่เพิ่มขึ้นประมาณร้อยละ 0.9 [1]

ประเทศไทยได้เปลี่ยนแปลงพฤติกรรมการใช้และบริโภค โดยเฉพาะในช่วงหลังสถานการณ์โควิด-19 ร้านค้าจะต้องปรับตัวเข้าสู่ ยุค New Normal โดยคำนึงถึงพฤติกรรมผู้บริโภคที่เปลี่ยนไป ผู้บริโภคหันมาใช้บริการเดลิเวอรี่มีแนวโน้มมากขึ้นในทุกๆ ปี จากสถิติการคาดการณ์ในปี 2567 - 2569 ธุรกิจร้านอาหารและเครื่องดื่มคาดว่าจะเติบโต 4.0-5.0% ต่อปี ด้วยรายได้รวมที่ 275-300 พันล้านบาท [2] ดังนั้นการสั่งสินค้าและอาหารออนไลน์เพื่อความสะดวกสบาย จึงทำให้เกิดการเปลี่ยนแปลงรูปแบบของธุรกิจ การค้าขายอาหารในรูปแบบออนไลน์มากยิ่งขึ้น จึงส่งผลให้ปริมาณขยะมูลฝอยและขยะพลาสติกแบบใช้ครั้งเดียวทิ้ง (Single use plastic) เพิ่มมากขึ้นอีกด้วย

จากปัจจัยต่างๆ ที่ได้กล่าวมาการแก้ปัญหาปริมาณขยะมูลฝอยและขยะพลาสติกแบบใช้ครั้งเดียวทิ้งนั้น สิ่งที่เราสามารถทำได้เบื้องต้นในส่วนของผู้บริโภค เช่น การปรับเปลี่ยนพฤติกรรมการบริโภคเน้นบริโภคในร้านอาหารมากยิ่งขึ้นหรือการแยกประเภทขยะก่อนทิ้งลงถังขยะเป็นต้น ส่วนในภาคอุตสาหกรรมนั้น ควรมีการจัดหากระบวนการกำจัดขยะที่มีประสิทธิภาพ ซึ่งเป็นสิ่งที่สำคัญอย่างมากในการจัดการขยะอย่างยั่งยืน โดยเริ่มจากการคัดแยกขยะที่ต้นทาง อาทิเช่น ที่บ้าน หรือสวนสาธารณะ ก่อนที่จะนำไปจัดการต่อด้วยวิธีการต่างๆ ไม่ว่าจะการรีไซเคิล การเผา และการฝัง ซึ่งการคัดแยกขยะที่มีประสิทธิภาพจะช่วยให้นำวัสดุกลับมาใช้งานใหม่อีกครั้งได้ การรีไซเคิล (Recycle) เป็นกระบวนการนำวัสดุที่ใช้งานแล้วนำกลับมาใช้งานใหม่อีกครั้งโดยผ่านการแปรรูปหรือเปลี่ยนแปลงสภาพเป็นผลิตภัณฑ์ใหม่ วัสดุที่สามารถนำมารีไซเคิลนั้นมีหลากหลายรูปแบบ หนึ่งในนั้นก็คือ “พลาสติก”

อย่างไรก็ตามการเผชิญกับปัญหาขยะพลาสติกปริมาณมาก แม้จะมีการพัฒนาระบบจัดการขยะในช่วงทศวรรษที่ผ่านมา เช่น ระบบคัดแยกในเทศบาลนนทบุรี ที่ยังขาดการสนับสนุนจากภาครัฐที่เพียงพอ จึงต้องพึ่งพาอาศัย คนเก็บของเก่า ร้านรับซื้อของเก่า ซึ่งการคัดแยกยังคงใช้แรงงานคนในการคัดแยกจึงทำให้เกิดการคัดแยกที่ไม่มีประสิทธิภาพเนื่องจากขาดคนงานที่มีทักษะ

เป็นสาเหตุหนึ่งที่ทำให้พลาสติกไปปะปนกับขยะมูลฝอยชุมชน ส่งผลให้พลาสติกจำนวนมากถูกกำจัดทิ้ง ฝังกลบหรือรั่วไหลลงแม่น้ำและไหลลงทะเลแทนที่จะถูกนำกลับมาใช้ใหม่ [3] ดังนั้นการพัฒนาระบบคัดแยกพลาสติก โดยการนำเทคโนโลยีปัญญาประดิษฐ์ (AI) ด้วยเทคนิคการเรียนรู้เชิงลึกมา (Deep Learning) ที่เรียกว่า YOLOv8 ซึ่งมีประสิทธิภาพสูงในการคัดแยกประเภทเข้ามาช่วยนั้น เป็นแนวทางนึงในการพัฒนาระบบคัดแยกพลาสติกที่มีประสิทธิภาพ เนื่องจากประเภทของพลาสติกที่สามารถนำมารีไซเคิลได้ในปัจจุบันนั้นมีหลากหลายประเภท และคุณสมบัติแตกต่างกัน ผู้วิจัยจึงเสนอแนวทางในการคัดแยกระบบคัดแยกพลาสติกโดยใช้ปัญญาประดิษฐ์ (AI) ที่สามารถนำไปประยุกต์ใช้กับระบบแบบเรียลไทม์ได้ และโครงสร้างของระบบนั้นสามารถทำความเข้าใจได้ง่าย หากต้องการที่จะปรับปรุงประสิทธิภาพ

1.2 วัตถุประสงค์ของการศึกษา

1. เพื่อพัฒนาระบบคัดแยกประเภทของขยะพลาสติกโดยใช้ปัญญาประดิษฐ์ (AI) ด้วยเทคนิคการเรียนรู้เชิงลึกจากภาพถ่าย
2. เพื่อเปรียบเทียบประสิทธิภาพของโมเดล YOLOv8 แต่ละรุ่น (YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, YOLOv8x) โดยหาโมเดล YOLOv8 ที่มีประสิทธิภาพดีที่สุดสำหรับนำไปใช้เป็นต้นแบบโมเดลองระบบคัดแยกประเภทของขยะพลาสติก

1.3 ขอบเขตการวิจัย

1. การศึกษาประเภทของพลาสติกที่สามารถนำไปรีไซเคิลได้ ซึ่งแบ่งออกเป็น 7 ประเภทตามสัญลักษณ์ตัวเลขบนผลิตภัณฑ์พลาสติก
   (1) PET/PETE/PE (Polyethylene Terephthalate)
   (2) HDPE (High-Density Polyethylene)
   (3) PVC (Polyvinyl Chloride)
   (4) LDPE (Low-Density Polyethylene)
   (5) PP (Polypropylene)
   (6) PS (Polystyrene)
   (7) Other (พลาสติกชนิดอื่น ๆ)
2. การใช้เทคนิคการเรียนรู้เชิงลึก (Deep Learning) เพื่อฝึกโมเดลในการคัดแยกประเภทพลาสติกจากภาพถ่าย

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้ต้นแบบโมเดลของระบบคัดแยกประเภทพลาสติกเพื่อการรีไซเคิลจากภาพถ่ายด้วยวิธีการเรียนรู้เชิงลึก

2. ลดปริมาณขยะพลาสติกที่เพิ่มขึ้น โดยการส่งเสริมการคัดแยกและการรีไซเคิลพลาสติกอย่างมีประสิทธิภาพ

3. สนับสนุนการนำพลาสติกที่สามารถรีไซเคิลกลับมาใช้ใหม่ ลดการใช้ทรัพยากรธรรมชาติและผลกระทบต่อสิ่งแวดล้อม

สถาบันเทคโนโลยีไทย-ญี่ปุ่น THAI - NICHI INSTITUTE OF TECHNOLOGY TNI

บทที่ 2
เอกสารและงานวิจัยที่เกี่ยวข้อง

ในการศึกษา การพัฒนาและเปรียบเทียบประสิทธิภาพของโมเดล YOLOv8 ในการคัดแยกประเภทของพลาสติกสำหรับการรีไซเคิลด้วยการประมวลผลจากภาพถ่ายดิจิทัล ผู้วิจัยได้สืบค้น ทฤษฎี หลักการ และงานวิจัยที่เกี่ยวข้อง โดยมีหัวข้อดังต่อไปนี้

2.1 ประเภทของพลาสติกและความสำคัญของการรีไซเคิล

2.2 เทคโนโลยีการคัดแยกพลาสติกในปัจจุบัน

2.3 การเรียนรู้เชิงลึก (Deep Learning) และการประยุกต์ใช้ในการประมวลภาพ

2.4 อัลกอริทึม YOLO (You Only Look Once)

2.5 งานวิจัยที่เกี่ยวข้องกับการคัดแยกพลาสติกด้วยวิธีการเรียนรู้เชิงลึก

2.6 ช่องว่างในงานวิจัย (Research Gap)

โดยแต่ละหัวข้อมีรายละเอียดดังนี้

2.1 ประเภทของพลาสติกและความสำคัญของการรีไซเคิลพลาสติก

2.1.1 ประเภทของพลาสติก

ปัจจุบันพลาสติก ถือว่าเป็นวัสดุสังเคราะห์ที่มีบทบาทที่สำคัญอย่างมากในยุคปัจจุบัน เพาะมีต้นทุนการผลิตที่ถูก และสามารถใช้ทดแทนวัสดุอื่นๆ ได้ จึงได้รับความนิยมอย่างมากในการนำมาใช้งาน ดังนั้นเมื่อมีการใช้งานพลาสติกอย่างกว้างขวางและมากขึ้น จึงทำให้เกิดขยะพลาสติกจำนวนมาก และเนื่องจากพลาสติกแต่ละชนิดมีคุณสมบัติทางเคมีและกายภาพที่แตกต่างกันออกไป และพลาสติกที่ใช้กันทั่วไปโดยปกติแล้วจะมีอยู่ 7 ประเภทหลัก โดยจะมีสัญลักษณ์ตัวเลขระบุไว้ที่บรรจุภัณฑ์ [4] ดังนี้

1. พอลีเอทิลีนเทเรฟทาเลต (Polyethylene Terephthalate - PET)

คุณสมบัติ : เป็นพลาสติกที่แข็งแรง ทนทาน มีความใส เหนียว และป้องกันการแพร่ผ่านของก๊าซได้ดี ทำให้รักษาคุณภาพของสิ่งที่บรรจุอยู่ภายในได้

ลักษณะ : ใสและมีความเงาคล้ายแก้ว

ผลิตภัณฑ์ : ขวดน้ำดื่ม, ขวดน้ำอัดลม, ขวดน้ำมันพืช, บรรจุภัณฑ์อาหาร

2. พอลีเอทิลีนความหนาแน่นสูง (High-Density Polyethylene - HDPE)

คุณสมบัติ : เป็นพลาสติกที่แข็งแรง มีความหนาแน่นสูง ทำให้ทนทานต่อสารเคมีและแรงกระแทกได้ดี สามารถรับน้ำหนักได้มาก

ลักษณะ : มีสีขาวขุ่นหรือทึบแสง ไม่ใสเหมือน PET แต่ยังคงความยืดหยุ่นได้เล็กน้อย

ผลิตภัณฑ์ : ขวดนม, ขวดแชมพู, ขวดน้ำยาซักผ้า, ถังขยะ, ถุงหิ้ว

3. พอลีไวนิลคลอไรด์ (Polyvinyl Chloride - PVC)

คุณสมบัติ : มีความแข็งแรงและทนทานต่อสภาพอากาศได้ดีเยี่ยม ทนต่อการกัดกร่อนของสารเคมีและน้ำ ป้องกันไขมันได้ดี แต่เมื่อได้รับความร้อนสูงจะอ่อนตัวลงได้ง่าย

ลักษณะ : มีแบบแข็งและแบบอ่อนและสามารถผลิตให้มีสีสันที่หลากหลายได้

ผลิตภัณฑ์ : ท่อน้ำประปา, สายไฟ, กรอบประตูหน้าต่าง, แผ่นปูพื้น

4. พอลีเอทิลีนความหนาแน่นต่ำ (Low-Density Polyethylene - LDPE)

คุณสมบัติ : มีความโปรงแสง ยืดหยุ่นสูง มีความนิ่มและเหนียว ความหนาแน่นต่ำทำให้ทนทานต่อการฉีกขาด และมีคุณสมบัติกันความชื้นได้ดี

ลักษณะ : มีความใส และยืดหยุ่นสูง

ผลิตภัณฑ์ : ถุงพลาสติกใส, ฟิล์มห่ออาหาร, ถุงเย็นสำหรับบรรจุอาหาร

5. พอลีโพรพิลีน (Polypropylene - PP)

คุณสมบัติ : มีน้ำหนักเบา แข็งแรง ทนทานต่อแรงกระแทกและรอยขีดข่วน ทนความร้อนได้สูงกว่าพลาสติกชนิดอื่นๆ ทำให้สามารถนำเข้าไมโครเวฟได้

ลักษณะ : มีลักษณะกึ่งใสหรือทึบแสง มีพื้นผิวค่อนข้างแข็ง

ผลิตภัณฑ์ : กล่องบรรจุอาหารสำหรับไมโครเวฟ, ขวดนมเด็ก, ฝาขวด, ถ้วยโยเกิร์ต

6. พอลีสไตรีน (Polystyrene - PS)

คุณสมบัติ : มีความโปร่งแสง แข็งแรงแต่เปราะบาง แตกหักง่าย มีน้ำหนักเบามากเมื่อทำเป็นโฟม และสามารถขึ้นรูปได้ง่าย

ลักษณะ : มีทั้งแบบใสและแบบโฟมที่มีน้ำหนักเบามาก

ผลิตภัณฑ์ : กล่องโฟมบรรจุอาหาร, ช้อน - ส้อมพลาสติกแบบใช้แล้วทิ้ง, ถาดอาหาร

7. พลาสติกอื่น ๆ (Other - O)

คุณสมบัติ : มีการผสมของพลาสติกและสารหลายชนิด จะนำไปรีไซเคิลค่อนข้างยาก เพราะจะไม่ทราบว่าเป็นพลาสติกชนิดใดผสมอยู่ โดยในงานวิจัยผู้วิจัยจะแบ่งพลาสติกประเภท 7 เป็นสองรูปแบบก็คือ พลาสติกใช้ครั้งเดียวทิ้ง (Single- use plastic) และ พลาสติกหลายชั้น (Multi-layer)

ลักษณะ : มีลักษณะแตกต่างกันไปตามชนิดของพลาสติกที่เป็นส่วนประกอบ

ผลิตภัณฑ์ : ขวดน้ำขนาดใหญ่ (PC), แว่นตา, แผ่นซีดี

2.1.2 ความสำคัญของการรีไซเคิลพลาสติก

การรีไซเคิลไม่ได้เพียงแค่ลดปริมาณขยะพลาสติก แต่ยังยังเป็นการใช้ทรัพยากรที่มีอยู่อย่างจำกัด ให้เกิดความคุ้มค่าอีกด้วย ดังนั้นการรีไซเคิลพลาสติกจึงมีความสำคัญดังต่อไปนี้

1. ช่วยลดการใช้ทรัพยากรธรรมชาติ เนื่องจากการผลิตพลาสติกขึ้นมาใหม่ จะต้องใช้วัตถุดิบจากปิโตรเลียม ซึ่งก็คือน้ำมันดิบและก๊าซธรรมชาติ วัตถุดิบทั้งสองอย่างนั้นเป็นทรัพยากรที่ใช้แล้วหมดไป การนำพลาสติกที่มีอยู่กลับมาใช้ใหม่ช่วยลดความจำเป็นในการขุดเจาะและแปรรูปวัตถุดิบใหม่ได้

2. ในกระบวนการผลิตพลาสติกขึ้นมาใหม่นั้น จะก่อให้เกิดการปล่อยก๊าซเรือนกระจกปริมาณมาก ซึ่งการรีไซเคิลแทนการผลิตขึ้นมาใหม่นั้น จะใช้พลังงานน้อยกว่าถึง 50-70% ดังนั้นการรีไซเคิลพลาสติก จึงช่วยลดมลพิษทางอากาศและน้ำเสียที่เกิดจากกระบวนการผลิตและการกำจัดพลาสติก

3. ช่วยสร้างมูลค่าทางเศรษฐกิจ ขยะพลาสติกที่ถูกคัดแยกอย่างถูกต้องสามารถกลายเป็นวัตถุดิบที่มีคุณภาพสูงและมีมูลค่าเพิ่มขึ้น การรีไซเคิลจึงช่วยให้ภาคอุตสาหกรรมสามารถดำเนินงานเกี่ยวข้องกับการเก็บรวบรวม คัดแยก และแปรรูปพลาสติก สร้างงานและรายได้ให้กับผู้คนในหลากหลายภาคส่วน

2.1.3 ปัญหาและความท้าทายในการคัดแยกพลาสติกในปัจจุบัน

1. ความหลากหลายของประเภทพลาสติกและโครงสร้างบรรจุภัณฑ์ที่มีโพลิเมอร์หลากหลายประเภท และการผลิตบรรจุภัณฑ์หลายชั้น (multilayer) ที่ผสานพลาสติกหลากชนิดเข้าด้วยกัน ทำให้การแยกด้วยเครื่องตรวจจับสเปกตรัมหรือแยกคัดแยกเชิงกลทำได้ยาก จึงทำให้ได้พลาสติกที่นำมารีไซเคิลปนเปื้อนและคุณภาพลดลงจนไม่สามารถขายได้ จนต้องนำไปเผาเพื่อผลิตพลังงานแทน [5]

2. การปนเปื้อนและความสกปรก เช่น เศษอาหาร คราบน้ำมัน ฉลาก ฯลฯ จะทำให้เครื่องจักรคัดแยกประเภทพลาสติกเกิดความขัดข้อง ทำให้คุณภาพของก้อนอัดต่ำ รบกวนการคัดแยกแบบออปติคัลและเกิดของเสียจากการผลิต จึงทำให้มีต้นทุนที่เพิ่มขึ้นจากการที่ต้องทำความสะอาดสิ่งปนเปื้อนและคราบสกปรกก่อนนำไปเข้าสู่กระบวนการคัดแยก [6]

2.2 เทคโนโลยีการคัดแยกพลาสติกในปัจจุบัน

เทคโนโลยีการคัดแยกพลาสติกในปัจจุบันมีความก้าวหน้าไปมาก โดยมีการใช้เทคนิคหลากหลายเพื่อจัดการกับปริมาณขยะพลาสติกที่เพิ่มขึ้น เช่น การคัดแยกด้วยมือ, การคัดแยกด้วยเครื่องตรวจจับ NIR โดยมีรายละเอียดดังนี้

1. การคัดแยกด้วยมือ เป็นวิธีดั้งเดิมที่ทำกันมาอย่างเนิ่นนาน โดยพนักงานคัดขยะบนสายพานจะต้องทำการคัดพลาสติกด้วยเกณฑ์ของโรงงาน เช่น เลือกตามสี รูปร่าง โลโก้ หรือรหัสรีไซเคิล ซึ่งการคัดแยกด้วยมือนั้น มีประสิทธิภาพการคัดแยกอยู่ที่ประมาณ 60 ชิ้นต่อชั่วโมง [7]

2. การคัดแยกด้วยเครื่องตรวจจับ NIR (NIR spectroscopy sorting) ด้วยการใช้การสะท้อนแสงปริมาณ 0.8–2.5 μm เพื่อจดจำโครงสร้างโพลิเมอร์ของพลาสติก โดยนำบรรจุภัณฑ์พลาสติกวางบนสายพาน ผ่านเซ็นเซอร์ แล้วยิงลมอากาศเล็งเป้าหมายให้ลอยไปยังสายพานแยกตามประเภท จากงานวิจัยเปรียบเทียบความแม่นยำ พบว่าการคัดแยกด้วยเครื่องจักรอัตโนมัตินั้น ให้ความแม่นยำกว่าการคัดแยกด้วยมือประมาณ 30% [8]

2.2.1 ข้อจำกัดของเทคโนโลยีที่มีอยู่ในปัจจุบัน

1. การคัดแยกด้วยมือ แม้ปลายทางบรรจุภัณฑ์จะมีรหัส แต่โดยมากผู้คัดไม่มีเวลาอ่าน จึงอาจเกิดข้อผิดพลาดได้และความแม่นยำในการคัดแยกที่ต่ำ

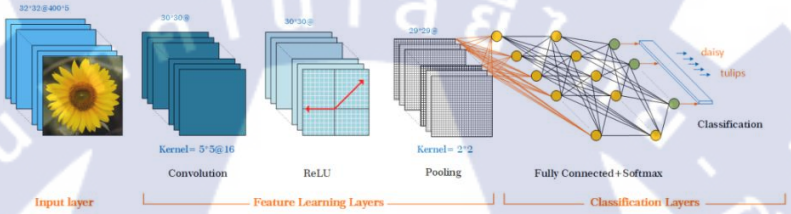
2. การคัดแยกด้วยเครื่องตรวจจับ NIR สามารถคัดแยกได้ประสิทธิภาพดี แต่ว่าการคัดแยกพลาสติกสีดำที่ผสมคาร์บอนแบล็กจะดูดกลืนแสง ทำให้เครื่องจักรตรวจจับไม่พบและมักถูกคัดแยกผิดสายพานคัดแยก อีกทั้งฉลาก (shrink-sleeve) ที่ติดมากับพลาสติกก็รบกวนการตรวจจับ NIR ได้เช่นกัน [9]

2.2.2 การประยุกต์ใช้เทคโนโลยีภาพถ่ายและการประมวลผลภาพในการคัดแยกพลาสติก

ในปัจจุบันเทคโนโลยีปัญญาประดิษฐ์ สามารถจำแนกประเภทของขยะพลาสติกได้จากภาพถ่ายด้วยการฝึกจากชุดข้อมูลภาพขนาดใหญ่ของขยะพลาสติกชนิดต่าง ๆ เพื่อให้สามารถเรียนรู้และแยกแยะลักษณะเฉพาะของแต่ละประเภทได้อย่างแม่นยำ และนำหุ่นยนต์มาใช้งานกับโมเดล AI ที่สามารถระบุชนิดของพลาสติกได้แล้วทำให้การคัดแยกมีประสิทธิภาพสูง [9] ด้วยวิธีดังกล่าวผู้วิจัยจึงเล็งเห็นว่าสามารถนำมาใช้เป็นเครื่องมือในการคัดแยกพลาสติกที่มีประสิทธิภาพได้

2.3 การเรียนรู้เชิงลึก (Deep Learning) และการประยุกต์ใช้ในการประมวลภาพ

การเรียนรู้เชิงลึก (Deep Learning) เป็นส่วนหนึ่งในแขนงของการเรียนรู้ของเครื่อง (Machine Learning) ที่เน้นการสร้างแบบจำลองด้วยโครงข่ายประสาทเทียมหลายชั้น (Deep Neural Networks) โดยเฉพาะอย่างยิ่งโครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Network: CNN) ซึ่งเหมาะสำหรับงานด้านการประมวลผลข้อมูลจากภาพ เช่นการใช้ CNN เพื่อดึงคุณลักษณะ (feature extraction) จากภาพหรือวิดีโอ เพื่อที่จะได้ ขอบ (edges), รูปร่าง (shapes) และวัตถุที่ซับซ้อน ดังนั้น CNN จึงมีจุดเด่นอย่างมากในการเรียนรู้ลักษณะเฉพาะของวัตถุในภาพได้โดยอัตโนมัติ จากการวิเคราะห์คุณลักษณะแบบลำดับชั้น โดยองค์ประกอบหลักของ CNN ประกอบด้วย ดังรูปที่ 2.1



รูปที่ 2.1 องค์ประกอบหลักของ Convolutional Neural Network: CNN

Convolutional Layer ทำหน้าที่กรองข้อมูลจากภาพด้วย kernel เพื่อวิเคราะห์คุณลักษณะที่สำคัญ เช่น เส้น ขอบ รูปร่าง

ReLU Layer ทำหน้าที่เพิ่มความไม่เป็นเชิงเส้น (non-linearity) ให้กับโมเดล ทำให้โมเดลสามารถเรียนรู้ข้อมูลที่ซับซ้อนได้ดีขึ้น

Pooling Layer ทำหน้าที่ย่อขนาดข้อมูล เพื่อเพิ่มความยืดหยุ่นของโมเดล ต่อการเปลี่ยนแปลงของข้อมูลภาพและลดความซับซ้อนในการคำนวณ

Fully Connected Layer หรือ Dense Layer ทำหน้าที่ประมวลผลข้อมูลทั้งหมดที่ได้จากเลเยอร์ก่อนหน้า มีบทบาทสำคัญในการเรียนรู้ความสัมพันธ์ที่ซับซ้อนและมักใช้ในการตัดสินใจขั้นสุดท้ายของโมเดล

2.3.1 การประยุกต์ใช้การเรียนรู้เชิงลึกในงานตรวจจับและจำแนกวัตถุจากภาพถ่าย

การใช้ Convolutional Neural Network (CNN) สำหรับการตรวจจับวัตถุ คือการใช้ Convolutional Neural Network (CNN) ที่สามารถเรียนรู้คุณลักษณะของภาพ (Features) ได้โดยอัตโนมัติ มาทำการประมวลผลข้อมูลภาพจากระดับพิกเซลดิบ (Raw Pixels) ผ่านหลายเลเยอร์

(Convolutional layers, Pooling layers) เพื่อสร้างคุณลักษณะที่ซับซ้อนขึ้นเรื่อยๆ ซึ่งช่วยให้ระบบสามารถจำแนกประเภท (Classification) และระบุตำแหน่งของวัตถุภายในภาพได้พร้อมกัน ซึ่งในอดีตงานจำแนกและงานตรวจจับจะทำแยกกันและการนำ CNN มาประยุกต์ใช้นั้น จะช่วยให้สามารถรวมสองฟังก์ชันนี้เข้าด้วยกันได้ในระบบเดียว และทำให้การตรวจจับวัตถุมีความแม่นยำและรวดเร็วยิ่งขึ้น [10]

2.3.2 ตัวอย่างงานวิจัยที่ใช้การเรียนรู้เชิงลึกในการคัดแยกวัตถุหรือขยะ

วิทยานิพนธ์เรื่อง “การพัฒนาและประยุกต์ใช้เทคนิค Convolutional Neural Network ในการจำแนกขยะชายฝั่งทะเล” โดยงานวิจัยนี้จะเล่าเกี่ยวกับปัญหาขยะทะเลและชายฝั่งกำลังทวีความรุนแรง โดยเฉพาะในประเทศไทยซึ่งต้องพึ่งพาทรัพยากรชายฝั่งและการท่องเที่ยว ปกติแล้วการจำแนกนั้นจะมีการใช้แรงงานคนในการเก็บรวบรวมและจัดหมวดหมู่ของขยะชายฝั่ง ซึ่งใช้เวลามากและอาจเกิดความคลาดเคลื่อนได้ งานวิจัยนี้จึงมีเป้าหมายที่จะนำเทคนิค Convolutional Neural Network (CNN) มาประยุกต์ใช้เพื่อจำแนกขยะจากภาพถ่ายจริง ถือเป็นความพยายามในการทำให้กระบวนการตรวจสอบขยะชายฝั่งทำงานแบบอัตโนมัติและผลลัพธ์ที่ได้นั้นโมเดล CNN สามารถจำแนกขยะได้ด้วย ความแม่นยำประมาณ 80–90% ซึ่งอยู่ในระดับที่น่าพึงพอใจ [11]

2.4 อัลกอริทึม YOLO (You Only Look Once)

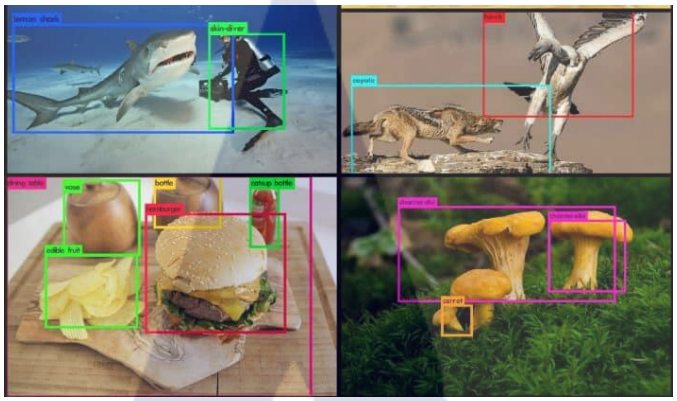
2.4.1 ความเป็นมาของ YOLO และวิวัฒนาการของเวอร์ชันต่างๆ

YOLO (You Only Look Once) ถูกนำเสนอครั้งแรกโดย Joseph Redmon และคณะในปี 2016 ที่นำเสนอว่า “การตรวจจับวัตถุ” จะเป็นการใช้ภาพต้นฉบับ ในการป้อนเข้าโมเดลเพียงครั้งเดียว เพื่อให้โมเดลทำนายค่าตัวเลขบางอย่างออกมาโดยตรงแล้วให้มองเป็นเป็นกรอบสี่เหลี่ยมและคลาสต่างๆ โดยทำให้เร็วแบบเรียลไทม์ และการป้อนข้อมูลเข้าสู่โมเดล แล้วทำให้โมเดลเรียนรู้ไปจนถึงผลลัพธ์สุดท้ายได้ง่ายกว่าวิธีก่อนหน้าอย่าง R-CNN หรือ SSD ในยุคสมัยนั้น [12]

โมเดล YOLO ก็มีการพัฒนาออกมาอยู่หลากหลายรุ่น ซึ่งโมเดล YOLO ยุคแรก (YOLOv1 – YOLOv3) จะมุ่งเน้นการปรับปรุงโครงสร้างพื้นฐานของโมเดลเพื่อเพิ่มความแม่นยำและประสิทธิภาพให้ดียิ่งขึ้นรายละเอียดดังนี้ [13]

- YOLOv1 เป็นการแนะนำแนวคิดที่จะปฏิวัติวงการ แต่ยังมีข้อจำกัดในเรื่องการตรวจจับวัตถุที่มีขนาดเล็กและมีความแม่นยำต่ำกว่าโมเดลอื่นๆ ในยุคสมัยนั้น

- YOLOv2 ได้มีการแก้ไขข้อจำกัดของรุ่นแรกด้วยการเพิ่มเทคนิคอย่าง Anchor Boxes ดังรูปที่ 2.2 และการรวม Batch Normalization โดยจะทำให้แต่ละ Layer ใน Neural Network สามารถเรียนรู้ได้ด้วยตัวเอง



รูปที่ 2.2 กล่อง Anchor Boxes

- YOLOv3 มีการปรับปรุงโครงสร้างพื้นฐานโดยใช้ Darknet-53 ซึ่งมีเลเยอร์ Convolution จำนวน 53 เลเยอร์ แทนที่จะเป็น 19 เลเยอร์เหมือน YOLOv2 ซึ่งทำให้มีประสิทธิภาพมากกว่า เพิ่มการทำนายผลในหลายสเกล (multi-scale prediction) ทำให้สามารถตรวจจับวัตถุขนาดเล็กได้ดีขึ้นอย่างมาก

การมาถึงของยุคใหม่ YOLO จาก YOLOv4 ถึง YOLOv10 การพัฒนาโมเดลในช่วงยุคนี้จะมุ่งเน้นการปรับปรุงประสิทธิภาพ, ความเร็ว, และการลดขนาดโมเดล ดังนี้

- YOLOv4 และ YOLOv5 นับเป็นจุดเปลี่ยนที่สำคัญ การนำเทคนิคที่หลากหลายมาใช้เพื่อเพิ่มประสิทธิภาพ ซึ่งจะเรียกเทคนิคเหล่านี้ว่า Bag of Freebies เทคนิคที่ช่วยให้โมเดลฉลาดขึ้นในตอนฝึก แต่ไม่กระทบความเร็วตอนใช้งานจริงและ Bag of Specials เทคนิคที่ทำให้แม่นยำขึ้น แต่ต้องยอมแลกกับการประมวลผลที่ช้าลงเล็กน้อยในตอนใช้งานจริง จึงทำให้โมเดล YOLO เวอร์ชันดังกล่าวมีความสมดุลระหว่างความเร็วและความแม่นยำอย่างมาก

- YOLOv6, YOLOv7 และ YOLOv8 วิวัฒนาการในช่วงนี้มุ่งเน้นการปรับปรุงสถาปัตยกรรม (Architecture) ของโมเดลให้มีประสิทธิภาพสูงขึ้น ทั้งในด้านความเร็วและความแม่นยำ รวมถึงการลดขนาดโมเดลเพื่อรองรับการทำงานบนอุปกรณ์แบบพกพา

- YOLOv9 และ YOLOv10 เป็นรุ่นล่าสุดที่มีการนำเสนอเทคนิคที่เรียกว่า Generalized Efficient Layer Aggregation Networks (GELAN) และ Gradient-based Dynamic Label Assignment (GDLA) ซึ่งช่วยยกระดับประสิทธิภาพของโมเดลให้มีความแม่นยำสูงขึ้นไปอีกขั้น โดยเฉพาะในด้านเกษตรกรรม โมเดลใหม่เหล่านี้สามารถนำไปประยุกต์ใช้เพื่อเพิ่มความแม่นยำในการตรวจจับโรคพืชหรือการนับผลผลิตได้อย่างมีประสิทธิภาพ

จากการวิเคราะห์ YOLO ในแต่ละยุคจะพบว่า YOLO รุ่น YOLOv8 นั้นจะตอบโจทย์การทำงานวิจัยในด้านนี้มากที่สุด ผู้วิจัยจึงได้ทำการศึกษาในรุ่นดังกล่าวแล้วพบว่า YOLOv8 นั้นจะมีโครงสร้างที่ไม่เหมือนกันโดยจะมีแบบจำลองให้เลือกใช้อยู่หลากหลายชนิด ดังรูปที่ 2.3 ซึ่งแต่ละชนิดก็จะมีข้อดี – ข้อเสียที่แตกต่างกัน ดังนี้

Model	size (pixels)	mAP^{val} 50-95	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	params (M)	FLOPs (B)
YOLOv8n	640	37.3	80.4	0.99	3.2	8.7
YOLOv8s	640	44.9	128.4	1.20	11.2	28.6
YOLOv8m	640	50.2	234.7	1.83	25.9	78.9
YOLOv8l	640	52.9	375.2	2.39	43.7	165.2
YOLOv8x	640	53.9	479.1	3.53	68.2	257.8

รูปที่ 2.3 YOLOv8 ทุกรุ่น

1. YOLOv8n (Nano) เป็นโมเดลที่ขนาดเล็กที่สุด มีจำนวนพารามิเตอร์น้อยที่สุดประมาณ 3.2 ล้านพารามิเตอร์ ทำให้มีขนาดไฟล์เล็กและใช้หน่วยความจำน้อยมาก จึงทำให้มีความเร็วสูงที่สุด สามารถประมวลผลได้อย่างรวดเร็วเป็นพิเศษ ทำให้เหมาะสำหรับการใช้งานแบบเรียลไทม์ (real-time) หรือแอปพลิเคชันที่ต้องการเฟรมเรทสูงใช้ทรัพยากรน้อย เหมาะสำหรับงานประเภท การทำแอปพลิเคชันมือถือสำหรับการตรวจจับวัตถุในภาพหรือวิดีโอที่ถ่ายจากสมาร์ทโฟน เป็นต้น

2. YOLOv8s (Small) มีขนาดใหญ่กว่า Nano เล็กน้อยประมาณ 11.2 ล้านพารามิเตอร์มีสมดุลระหว่างความเร็วและความแม่นยำ ให้ประสิทธิภาพที่ดีขึ้นในด้านความแม่นยำเมื่อเทียบกับ Nano โดยที่ยังคงความเร็วในการประมวลผลที่น่าพอใจ เหมาะสำหรับงานประเภท การวิเคราะห์วิดีโอแบบเรียลไทม์ สำหรับงานที่ซับซ้อนขึ้นเล็กน้อย เช่น การติดตามการจราจร, การนับวัตถุที่ละเอียดขึ้น เป็นต้น

3. YOLOv8m (Medium) โมเดลขนาดกลาง มีจำนวนพารามิเตอร์ประมาณ 25.9 ล้านพารามิเตอร์และมีความแม่นยำเพิ่มขึ้นอย่างเห็นได้ชัด เมื่อเทียบกับรุ่น Small และ Nano แต่ในส่วนความเร็ว จะช้ากว่ารุ่นที่ผ่านมา แต่ก็ยังเพียงพอสำหรับการใช้งานในหลายๆ กรณี เหมาะสำหรับงานประเภท การวิเคราะห์ภาพถ่ายดาวเทียมหรือโดรน สำหรับการระบุพื้นที่เพาะปลูก, สิ่งก่อสร้าง, ภัยพิบัติ เป็นต้น

4. YOLOv8l (Large) โมเดลขนาดใหญ่ มีจำนวนพารามิเตอร์ประมาณ 43.7 ล้านพารามิเตอร์ ความแม่นยำสูง ให้ผลลัพธ์ที่แม่นยำมาก โดยเฉพาะอย่างยิ่งในการตรวจจับวัตถุขนาด

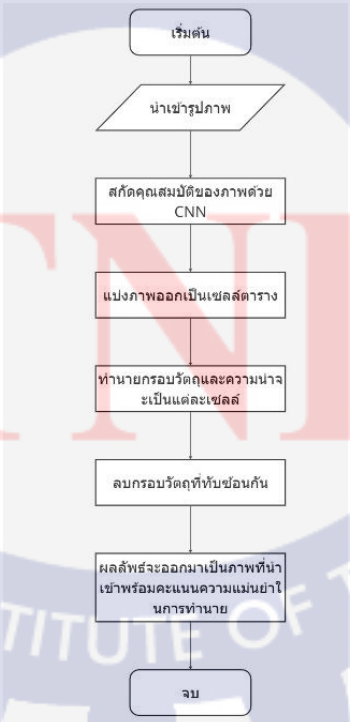
เล็กหรือวัตถุที่ซับซ้อน ต้องการทรัพยากรมากขึ้น ต้องใช้ GPU ที่มีประสิทธิภาพค่อนข้างสูงเหมาะสำหรับงานประเภทงานด้านการแพทย์และชีวภาพ เช่น การตรวจจับเซลล์, เนื้อเยื่อ, หรือความผิดปกติในภาพทางการแพทย์ที่ต้องการความแม่นยำสูงเป็นพิเศษ เป็นต้น

5. YOLOv8x (Extra Large) ขนาดใหญ่ที่สุด: มีจำนวนพารามิเตอร์มากที่สุดประมาณ 68.2 ล้านพารามิเตอร์ มีความแม่นยำสูงสุด ต้องการทรัพยากรสูงมาก จำเป็นต้องใช้ GPU ระดับไฮเอนด์ (เช่น NVIDIA A100, H100 หรือ RTX series ระดับสูง) และหน่วยความจำ GPU จำนวนมาก เหมาะสำหรับงานประเภท แอปพลิเคชันหรือระบบอุตสาหกรรมที่มีความสำคัญต่อการดำเนินงานหลักของธุรกิจ หรือต่อความปลอดภัยของชีวิตและทรัพย์สิน (Mission-critical Industrial Applications) ที่การตรวจจับที่ผิดพลาดเพียงเล็กน้อยอาจนำไปสู่ผลเสียหายอยากมากได้

จากโมเดลดังกล่าวทั้ง 5 ประเภท เราจึงต้องการวิจัยว่าแบบจำลองประเภทไหนให้ประสิทธิภาพมากที่สุดเมื่อนำมาประยุกต์ใช้กับการคัดแยกประเภทพลาสติกสำหรับการรีไซเคิล

2.4.2 หลักการทำงานของ YOLO ในการตรวจจับและจำแนกวัตถุจากภาพ

กระบวนการทำงานของอัลกอริทึม YOLO จะประกอบไปด้วยขั้นตอนดังภาพที่ 2.4 และมีรายละเอียดดังนี้ [14]



รูปที่ 2.4 ขั้นตอนการทำงานของ YOLO

1. โมเดล YOLO จะนำเข้ารูปภาพที่ต้องการตรวจจับวัตถุ

2. ทำการสกัดคุณสมบัติของภาพด้วยโครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN) จากภาพโดยแบ่งออกเป็นส่วนย่อยๆ ที่เรียกว่า “ฟิลเตอร์”ที่ทำหน้าที่ตรวจจับคุณลักษณะของภาพ เช่น เส้น ขอบ หรือสี โดยผลลัพธ์ที่ได้ก็คือตำแหน่งของภาพที่มีลักษณะต่างๆ

3. การแบ่งภาพเป็นเซลล์ตารางหรือกริด (Grid Division) โดยทั่วไปแล้ว YOLO จะแบ่งภาพออกเป็นกริดขนาด 7x7 หรือ 13x13 ซึ่งจะครอบคลุมพื้นที่ของพื้นผิวของภาพ

4. ทำนายกรอบวัตถุ (Bounding Box) และคะแนนความน่าจะเป็นที่มีวัตถุอยู่ในกรอบนั้น (Confidence Score) ในแต่ละกริด

5. ลบกรอบของวัตถุที่ทับซ้อนกันด้วยการรวมผลลัพธ์ (Non-Maximum Suppression) หลังจากการทำนาย YOLO จะได้ Bounding Boxes จำนวนมาก ซึ่งอาจมีกรอบที่ซ้อนทับกัน ด้วยวิธีดังกล่าวจะใช้เพื่อลดกรอบที่ซ้ำซ้อน โดยเก็บเฉพาะกรอบที่มีค่าความน่าจะเป็นวัตถุสูงที่สุด และตัดกรอบที่มีค่า IOU ซ้อนกันเกินค่าที่กำหนดออกไป

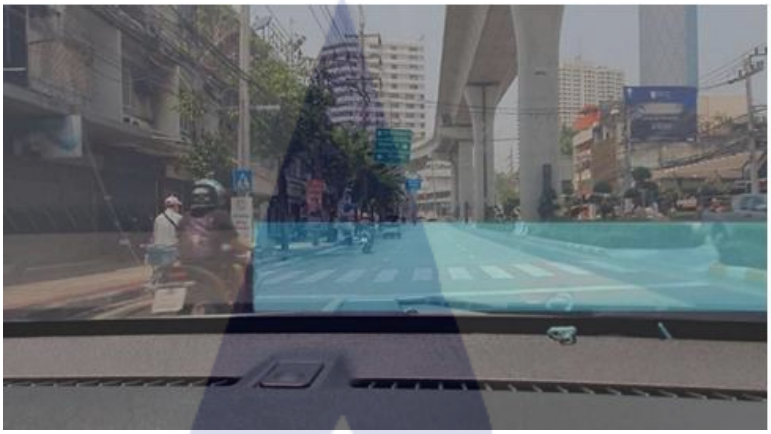
6. แสดงผลลัพธ์ของภาพที่นำเข้าพร้อมให้คะแนนความแม่นยำในการทำนาย

2.4.3 ข้อดีของ YOLO เมื่อเทียบกับอัลกอริทึมอื่น

YOLO (You Only Look Once) เป็น อัลกอริธึม Deep Learning ที่ได้รับความนิยมสูงสุดในยุคปัจจุบัน สำหรับงานด้านการตรวจจับวัตถุ (Object Detection Algorithm) ด้วยขั้นตอนเดียว (Single-Stage) จึงทำให้สามารถตรวจจับวัตถุหลายรายการในภาพเดียวกันได้อย่างรวดเร็วและแม่นยำ และแตกต่างจากอัลกอริธึมเก่าๆ เช่น R-CNN, Faster R-CNN ที่จะต้องตรวจจับวัตถุจากการประมวลผลหลายขั้นตอน (multi-stage) แล้วทำนายตำแหน่งและประเภทของวัตถุทั้งหมดพร้อมกัน

2.4.4 ตัวอย่างงานวิจัยที่ใช้ YOLO ในงานตรวจจับหรือคัดแยกวัตถุ

วิทยานิพนธ์เรื่อง “การตรวจจับทางม้าลายที่ซีดจางโดยใช้การตรวจจับวัตถุ” มหาวิทยาลัยธรรมศาสตร์ (2024) งานวิจัยนี้ศึกษาเกี่ยวกับการตรวจจับ ทางม้าลายซีดจาง (Faded Crosswalk Detection) โดยใช้ YOLOv8 มีวัตถุประสงค์หลักเพื่อแก้ไขปัญหาความปลอดภัยบนท้องถนน โดยเฉพาะในเขตเมืองที่เครื่องหมายจราจรบนพื้นถนนทางม้าลายมักจะเสื่อมสภาพหรือซีดจางจนมองเห็นได้ยาก ซึ่งเป็นปัจจัยเสี่ยงต่อการเกิดอุบัติเหตุ ดังรูปที่ 2.5 ผลการทดลองแสดงให้เห็นว่า YOLOv8 สามารถตรวจจับและจำแนกทางม้าลายซีดจางได้อย่างแม่นยำ และมีประสิทธิภาพในการทำงานแบบเรียลไทม์ เหมาะสำหรับการนำไปประยุกต์ใช้ใน ระบบตรวจสอบสภาพถนนอัจฉริยะ (Smart Road Monitoring Systems) หรือใน เมืองอัจฉริยะ (Smart City) เพื่อช่วยหน่วยงานที่เกี่ยวข้องติดตามและบำรุงรักษาเครื่องหมายจราจรอย่างทันท่วงที [15]



รูปที่ 2.5 ภาพตัวอย่างระบบการตรวจจับทางม้าลายที่ซีดจางโดยใช้การตรวจจับวัตถุ

2.5 งานวิจัยที่เกี่ยวข้องกับการคัดแยกพลาสติกด้วยวิธีการเรียนรู้เชิงลึก

2.5.1 การพัฒนาระบบคัดแยกขยะรีไซเคิลด้วยเทคโนโลยีไอโอที

เป็นงานวิจัยของ ธานิล ม่วงพลู และ วริยา เย็นเปิง ได้ทำการศึกษาหัวข้อ การพัฒนาระบบคัดแยกขยะรีไซเคิลด้วยเทคโนโลยีไอโอที งานวิจัยนี้นำเสนอระบบคัดแยกขยะรีไซเคิลที่ใช้เทคโนโลยี IoT ในการควบคุมการเปิด-ปิดถังขยะ คัดแยกขวดโลหะและขวดพลาสติก พร้อมทั้งบอกจำนวนขยะที่รับเข้าและแสดงพิกัดของถังขยะผ่านแอปพลิเคชัน LINE จากการทดสอบจำนวน 10 รอบพบว่าระบบสามารถคัดแยกพลาสติกได้พร้อมกับระบุจำนวน และพิกัดผ่านโทรศัพท์มือได้ถูกต้อง[16] โดยมีโครงสร้างการทำงานของระบบดังรูปที่ 2.6

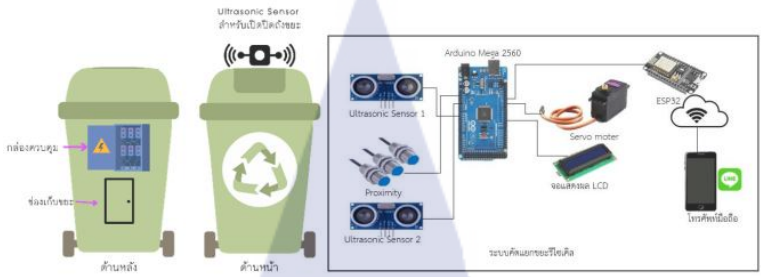
ข้อดี:

1.ระบบสามารถคัดแยกขวดโลหะและขวดพลาสติกได้อย่างมีประสิทธิภาพ

2.การแจ้งเตือนผ่านแอปพลิเคชันไลน์ทำให้ผู้ใช้งานสามารถติดตามสถานะของถังขยะได้แบบเรียลไทม์

ข้อจำกัด:

ระบบอาจต้องการการบำรุงรักษาและการเชื่อมต่ออินเทอร์เน็ตที่เสถียรเพื่อให้ทำงานได้อย่างต่อเนื่อง



รูปที่ 2.6 การพัฒนาระบบคัดแยกขยะรีไซเคิลด้วยเทคโนโลยีไอโอที

2.5.2 การประยุกต์ใช้ Teachable Machine สำหรับระบบคัดแยกขวดน้ำอัตโนมัติ

เป็นงานวิจัยของ บพิตร ไชยนอก และ ฤชานนท์ ศรีราวงค์ งานวิจัยนี้พัฒนาระบบคัดแยกประเภทขวดน้ำอัตโนมัติ โดยใช้เทคโนโลยีการประมวลผลภาพและการเรียนรู้ของเครื่อง เพื่อแก้ปัญหาการใช้แรงงานคนในการคัดแยกขยะ ใช้เทคนิค Teachable machine ในการคัดแยกประเภทของขวดน้ำได้แก่ ขวดพลาสติก ขวดแก้ว กล่องกระดาษ และกระป๋องอะลูมิเนียม รองรับขวดขนาดไม่เกิน 1.5 ลิตร เก็บข้อมูลการคัดแยกและแสดงสถิติบบเว็บไซต์ ผลการทดลองพบว่าสามารถคัดแยกประเภทขวดน้ำสามารถคัดแยกประเกทของขวดน้ำได้อย่างแม่นยำดังรูปที่ 2.7 โดยมีอัตราความถูกต้องเฉลี่ยอยู่ที่ประมาณ 95% ระบบบสามารถประมวลผลภาพและจำแนกประเภทขวดน้ำได้ภายในเวลาไม่ถึง 1 วินาทีต่อขวด ซึ่งผลลัทธ์ที่กล่าวอยู่ในเกณฑ์ที่มีประสิทธิสูง [17]

ข้อดี:

สามารถตรวจจับและคัดแยกขยะในสภาพแวดล้อมจริงได้อย่างมีประสิทธิภาพช่วยลดปัญหาขยะในแม่น้ำและส่งเสริมการรักษาสิ่งแวดล้อม

ข้อจำกัด:

การประมวลผลภาพในสภาพแวดล้อมจริงอาจเผชิญกับความท้าทาย เช่น แสงที่เปลี่ยนแปลงหรือสิ่งกีดขวางและต้องการฮาร์ดแวร์และซอฟต์แวร์ที่มีประสิทธิภาพสูง

รูปที่ 2.7 การคัดแยกประเภทขวดน้ำ

2.5.3 การพัฒนาเว็บแอปพลิเคชันแยกประเภทขยะด้วยเทคโนโลยีปัญญาประดิษฐ์

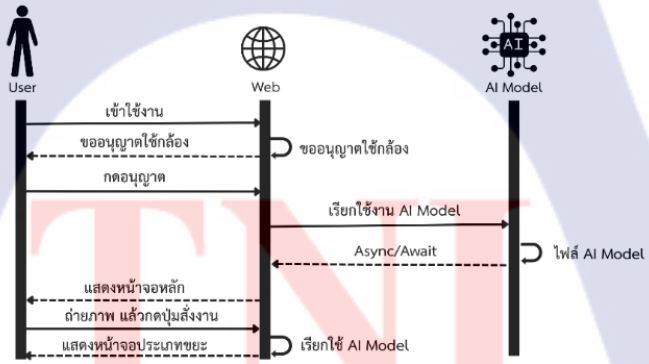
เป็นงานวิจัยของวดีนาถ วรรณสวัสดิ์กุล งานวิจัยนี้เป็นการสร้างเว็บแอปพลิเคชันสำหรับการให้ข้อมูลคัดแยกขยะซึ่งสามารถใช้งานได้ทั้งคอมพิวเตอร์และสมาร์ทโฟน โดยได้นำเทคโนโลยีปัญญาประดิษฐ์ มาใช้วิเคราะห์ภาพขยะเพื่อให้ได้ประเภทขยะที่ถูกต้อง และจะแสดงผลลัพธ์ที่หน้าจออุปกรณ์ โครงการวิจัยนี้ได้นำ Teachable Machine มาใช้เป็นแบบจำลองของปัญญาประดิษฐ์ของขยะจำนวน 4 ประเภท โดยแบ่งข้อมูลออกเป็น 12 คลาส โมเดลปัญญาประดิษฐ์ที่ได้มีความถูกต้องร้อยละ 85.33 จากนั้นใช้เอชทีเอ็มแอล ร่วมกับจาวาสคริปต์ในการพัฒนาแอปพลิเคชัน เว็บไซต์ที่พัฒนาสามารถใช้งานแสดงประเภทขยะได้อย่างถูกต้องร้อยละ 84.67 [18] โดยมีหลักการทำงานของระบบดังรูปที่ 2.8

ข้อดี:

เว็บแอปพลิเคชันสามารถใช้งานได้ทั้งบนคอมพิวเตอร์และสมาร์ทโฟน และมีความถูกต้องในการคัดแยกประเภทขยะเฉลี่ยอยู่ที่ 84.67%

ข้อจำกัด:

ความแม่นยำของระบบยังสามารถปรับปรุงได้ เนื่องจากมีความถูกต้องเฉลี่ยที่ 84.67% ซึ่งอาจไม่เพียงพอสำหรับการใช้งานในบางสถานการณ์



รูปที่ 2.8 หลักการทำงานของระบบ

2.6 ช่องว่างในงานวิจัย (Research Gap)

จากการศึกษางานวิจัยในส่วนของการพัฒนาระบบคัดแยกขยะพลาสติกด้วยเทคนิคการเรียนรู้เชิงลึก (Deep Learning) แม้เทคโนโลยีจะมีความก้าวหน้าอย่างมาก แต่ก็ยังมีช่องว่างที่ของงานวิจัยที่สามารถนำไปพัฒนาและต่อยอด ดังนี้

1. ในด้านการฝึกฝนโมเดล เนื่องจากชุดข้อมูลที่นำมาใช้ในการฝึกโมเดลไม่ครอบคลุมพลาสติกทุกประเภทในสภาพต่างๆ เช่น พลาสติกที่สกปรก หรือเปียก และชุดข้อมูลที่นำมาฝึกนั้นยังไม่มีมาตรฐานที่ชัดเจน

2. ด้านการคัดแยกที่ซับซ้อน ถึงแม้ปัจจุบัน AI จะสามรถช่วยคัดแยกได้แต่พบว่ามีข้อจำกัดในการคัดแยกพลาสติกหลายชั้น (Muti-layer Plastics) ที่ทำได้ค่อนข้างยาก

2.6.1 เหตุผลงานวิจัยนี้จึงเลือกใช้ YOLO และการประมวลผลภาพถ่ายเพื่อแก้ปัญหา

เหตุผลที่งานวิจัยนี้เลือกใช้ YOLO และการประมวลผลภาพถ่ายเพื่อแก้ปัญหาดังกล่าวมีดังนี้

1. ความเร็วในการประมวลผล (Real-time Processing) การคัดแยกพลาสติกในสายการผลิตจำเป็นต้องทำงานอย่างรวดเร็ว หากใช้วิธีการประมวลผลภาพแบบดั้งเดิมอาจใช้เวลานานและไม่สามารถทำงานแบบเรียลไทม์ได้ ดังนั้น YOLO ที่ถูกออกแบบมาให้ตรวจจับและจำแนกวัตถุได้ในขั้นตอนเดียว (Single-Stage) ทำให้สามารถคัดแยกพลาสติกได้ทันที

2. ความแม่นยำในการจำแนกพลาสติกหลายประเภท ที่มีลักษณะทางกายภาพคล้ายคลึงกันได้ การใช้ YOLO ที่สามารถตรวจจับและจำแนกวัตถุหลายคลาส (multi-class classification) พร้อมกันได้ จึงช่วยเพิ่มความแม่นยำในการระบุประเภทของพลาสติกที่หลากหลายได้

3. ความสามารถในการตรวจจับวัตถุที่มีรูปร่างและสถานะภาพหลากหลาย เมื่อพลาสติกเข้าสู่ระบบรีไซเคิลแล้วพลาสติกอาจอยู่ในรูปแบบที่บิดเบี้ยว ซีดจาง หรือถูกบด YOLO มีความสามารถในการตรวจจับวัตถุที่มีรูปร่างไม่สมบูรณ์ และมีความสามารถในกการตรวจจับการเปลี่ยนแปลงของภาพ เช่น แสง สี หรือมุมมองของภาพ

2.6.2 ความแปลกใหม่ (novelty) หรือจุดเด่นของงานวิจัยนี้เมื่อเทียบกับงานวิจัยก่อนหน้านี้

1. มีการประยุกต์ใช้ YOLO กับการคัดแยกพลาสติกโดยตรง จากงานวิจัยส่วนใหญ่ที่ผ่านมา YOLO มักจะถูกใช้ไปกับการตรวจจับวัตถุทั่วไป เช่น ยานพาหนะ คน สัตว์ แต่ในงานวิจัยนี้จะเป็นการนำมาประยุกต์ใช้กับการคัดแยกขยะพลาสติกเพื่อรีไซเคิล งานนี้จึงถือว่าเป็นการขยายขอบเขตการใช้งาน YOLO ไปสู่ด้านการจัดการขยะและแนวคิดใหม่ๆ ของเศรษฐกิจหมุนเวียน

2. การจำแนกพลาสติกหลายประเภทพร้อมกัน (Multi-class Plastic Classification) จากงานวิจัยที่ผ่านมา บางงานอาจจำแนกได้เพียงเป็นพลาสติกกับไม่เป็นพลาสติก แต่งานวิจัยนี้มุ่งเน้นการจำแนกพลาสติกตาม ประเภทโพลิเมอร์ เช่น PET, HDPE, LDPE, PVC, PP, PS ซึ่งมีความสำคัญต่อการรีไซเคิลในเชิงอุตสาหกรรมอย่างมาก

บทที่ 3
ระเบียบและวิธีดำเนินการวิจัย

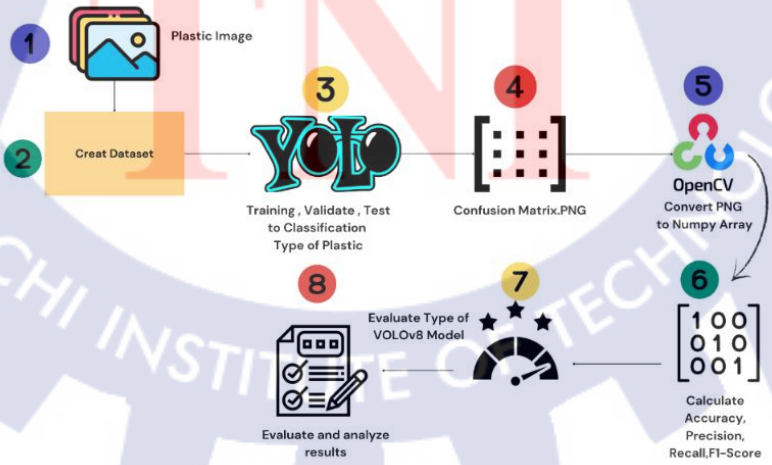
การวิจัยเรื่องการพัฒนาและเปรียบเทียบประสิทธิภาพของโมเดล YOLOv8 ในการคัดแยกประเภทของพลาสติกสำหรับการรีไซเคิลด้วยการประมวลผลจากภาพถ่ายดิจิทัลนี้ สามารถแบ่งวิธีการดำเนินงานตามขั้นตอนได้ดังตารางที่ 3.1 ขั้นตอนงานวิจัย และมีแผนการดำเนินงานวิจัย ดังตารางที่ 3.2 โดยขั้นตอนในการดำเนินงานสามารถอธิบายได้ดังรูปที่ 3.1 ขั้นตอนการดำเนินงาน

ตารางที่ 3.1 ขั้นตอนงานวิจัย

ขั้นตอนหลัก	ขั้นตอนย่อย
ขั้นตอนด้านนิยาม และปัญหา	ขั้นตอนที่ 1 การวิเคราะห์ปัญหาของงานวิจัย
	ขั้นตอนที่ 2 การศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้อง
ขั้นตอนการดำเนินงาน	ขั้นตอนที่ 3 การรวบรวมรูปภาพพลาสติกชนิดต่างๆ โดยแยกประเภทและแบ่งข้อมูลไว้สำหรับการฝึกฝน (Train) ตรวจสอบ (Validate) และทดสอบ (Test)
	ขั้นตอนที่ 4 การสร้างแบบจำลอง (Model) โดยเลือกใช้ VOLOv8 โมเดลรุ่นต่างๆ มาใช้เป็นฐานโมเดลสำหรับประมวลผลภาพ โดยนำภาพที่เตรียมไว้มาฝึกฝน (Train) ตรวจสอบ (Validate) และทดสอบ (Test)
	ขั้นตอนที่ 5 วัดประสิทธิภาพของความแม่นยำของโมเดลแต่ละรุ่นของ VOLOv8 จะให้ผลลัพธ์อยู่ในรูปแบบของ Confusion Matrix
	ขั้นตอนที่ 6 ทำการแปลงข้อมูลจากภาพ เนื่องจาก Confusion Matrix ที่ได้จะอยู่ในรูปแบบของไฟล์ PNG จึงใช้วิธีการของ Image Processing มาช่วยดังค่าจากภาพมาใช้
	ขั้นตอนที่ 7 คำนวณค่าจาก Confusion Matrix จะได้ค่า Accuracy, Precision, Recall,F1-Score เฉลี่ยจากโมเดลVOLOv8 รุ่นต่างๆ
	ขั้นตอนที่ 8 ประเมินและวิเคราะห์ผลการทดลอง
ขั้นตอนการสรุปผล	ขั้นตอนที่ 9 สรุปผลการทดลองและจัดทำเอกสารสารนิพนธ์

ตารางที่ 3.2 แผนการดำเนินงานวิจัย

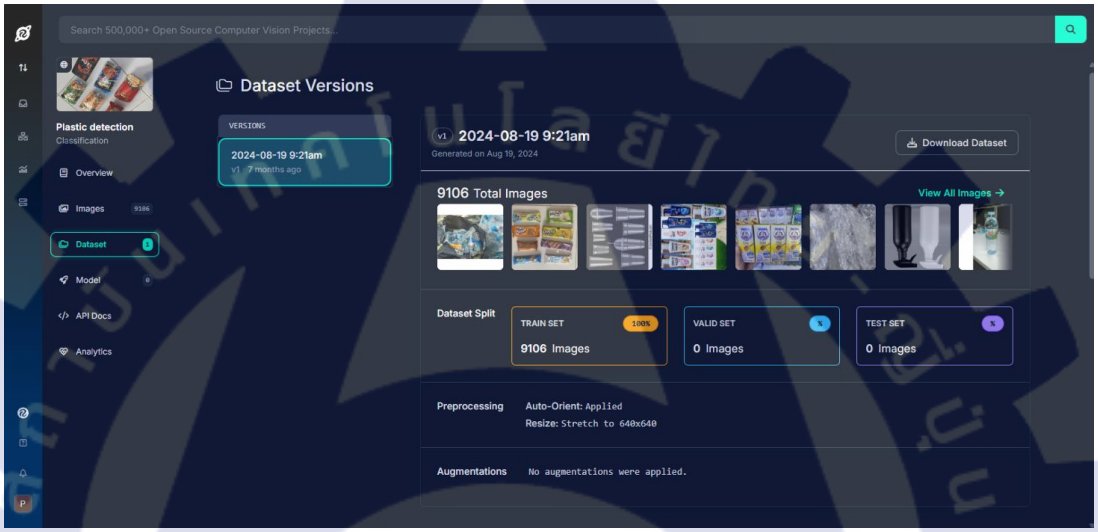
การดำเนินงาน	ระยะเวลา									
	ปี 2567		ปี 2568							
	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	มิ.ย.	ก.ค.	ส.ค.
ขั้นตอนเบื้องต้น										
- การหาหัวข้อวิจัย										
- การทบทวนวรรณกรรม										
- ร่างเค้าโครงสารนิพนธ์										
- นำเสนอเค้าโครงสารนิพนธ์										
ขั้นตอนการดำเนินการ										
- การเก็บรวบรวมข้อมูล										
- การประมวลผลข้อมูล										
ขั้นตอนการสรุปผล										
- สรุปผลงานวิจัย										
- เขียนสารนิพนธ์เล่มสมบูรณ์										
- สอบป้องกันสารนิพนธ์										



รูปที่ 3.1 ภาพรวมการออกแบบทางด้านซอฟต์แวร์ (Software Design)

3.1 การรวบรวมรูปภาพพลาสติกและการแบ่งเป็นชุดข้อมูลสำหรับนำไปใช้

ในการรวบรวมรูปภาพพลาสติกชนิดต่างๆ เพื่อที่จะนำมาใช้เป็นข้อมูล (Dataset) นั้น ผู้วิจัยได้รวบรวม ผ่านแพลตฟอร์มจัดการและเทรนโมเดล Computer Vision โดยช่วยใน การสร้างและจัดการชุดข้อมูลสำหรับการทำวิจัยประเภท Computer Vision ชื่อว่า “Roboflow” ซึ่งช่วยในการรวบรวมรูปภาพปริมาณมากและแบ่งออกเป็นประเภทต่างๆ ไว้ให้เรียบร้อยแล้ว ดังรูปที่ 3.2 และรูปพลาสติกที่สกปรก เปียก หรืออยู่ปะปนกับพื้นผิวอื่นๆจากแหล่งภายนอกจำนวน 50 ภาพ รวม 9156 ภาพ



รูปที่ 3.2 แพลตฟอร์ม Roboflow

เกณฑ์การแบ่งชุดข้อมูลจะแบ่งในรูปแบบ 70% Training, 15% Validation, 15% Test เป็นสัดส่วนที่ นิยมใช้และแนะนำกันทั่วไป ในวงการ Machine Learning และ Deep Learning [19] ซึ่งจากชุดข้อมูลทั้งหมด 9156 ภาพให้เหมาะสม โดยจะได้เกณฑ์ดังนี้

ชุดสำหรับการฝึกฝน (Train) จำนวน 6410 ภาพ (70%)

ชุดสำหรับการตรวจสอบ (Validate) จำนวน 1373 ภาพ (15%)

ชุดสำหรับการทดสอบ (Test) จำนวน 1373 ภาพ (15%)

ซึ่งแต่ละชุดข้อมูลจะแบ่งออกเป็น 8 คลาส ก็คือ PET, HDPE, PVC, LDPE, PP, PS, Single use plastic, Multilayer plastic ดังรูปที่ 3.3

รูปที่ 3.3 การแบ่งคลาสของประเภทพลาสติก

3.2 การสร้างแบบจำลอง (Model)

ผู้วิจัยได้ใช้โปรแกรม Google Colab เพื่อดำเนินการสร้างโมเดลผ่านเว็บเบราเซอร์ และเลือกใช้โมเดลของ YOLOv8 และใช้งานในส่วนของงานด้าน Classification มาเป็นฐานโมเดลสำหรับการพัฒนาโมเดล เพื่อการฝึกฝน ตรวจสอบ และทดสอบ และโดยจะแสดงตัวอย่างของผลลัพธ์ของการทดสอบที่โมเดลทำนายได้ ดังรูปที่ 3.4

รูปที่ 3.4 ผลลัพธ์ของการใช้โมเดล

3.3 การประเมินผลโมเดล

ในงานวิจัยนี้จะใช้ Confusion Matrix ซึ่งเป็นเครื่องมือในการประเมินผลลัพธ์ของการทำนาย ที่ทำนายจากโมเดลที่เราสร้างขึ้น จากการเรียนรู้ของเครื่องโดยวัดว่า สิ่งที่โมเดลทำนายกับสิ่งที่เกิดขึ้นจริง มีสัดส่วนเป็นอย่างไร [20] ผู้วิจัยจึงนำเครื่องมือดังกล่าวมาประยุกต์ใช้ในการประเมินประสิทธิภาพของโมเดลคัดแยกประเภทของพลาสติก

ตาราง 3.3 คำอธิบายหลักการของ Confusion Matrix

	คำอธิบาย
True Positive (TP)	จำนวนภาพที่โมเดลจัดเป็นพลาสติกและเป็นเพลาสติกจริง แสดงว่าโมเดลตรวจจับภาพพลาสติกได้ถูกต้อง
False Negative (FN)	จำนวนภาพที่โมเดลจัดเป็นไม่ใช่พลาสติกและไม่ได้เป็นพลาสติกจริง แสดงว่า โมเดลแยกแยะภาพที่ไม่ใช่พลาสติกได้ถูกต้อง
False Positive (FP)	จำนวนภาพที่โมเดลจัดเป็นพลาสติก แต่ไม่ได้เป็นพลาสติกจริง แสดงว่าโมเดล ตรวจจับภาพที่ไม่ใช่เพลาสติกเป็นพลาสติก
True Negative (TN)	จำนวนภาพที่โมเดลจัดเป็นไม่ใช่พลาสติก แต่เป็นพลาสติกจริง แสดงว่าโมเดล ผิดพลาดในการตรวจจับภาพพลาสติก

เมื่อค่า TP, FP, FN, และ TN แล้ว สามารถนำค่าเหล่านี้ไปใช้ในการคำนวณประสิทธิภาพของแบบจำลอง โดยมี สูตรสำคัญดังนี้

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

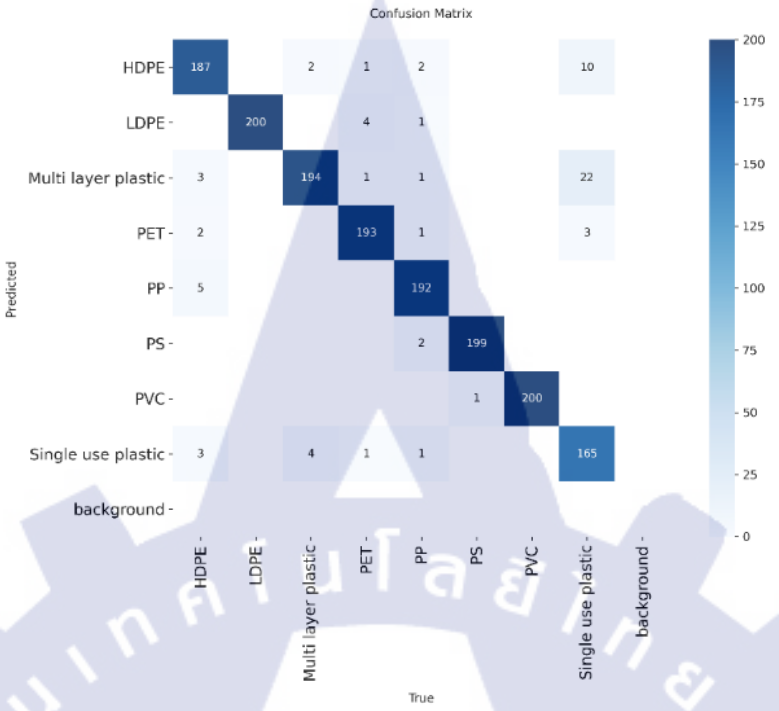
$$Precision = \frac{TP}{TF+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (4)$$

การคำนวณค่าจาก Confusion Matrix สามารถวัดค่าประสิทธิภาพโมเดลได้ดังนี้ Precision ค่าร้อยละของตัวอย่างที่โมเดลจำแนกเป็นคลาสใดคลาสหนึ่ง และเป็นคลาสที่ถูกต้อง Recall ค่าร้อยละของตัวอย่างที่ แท้จริงอยู่ในคลาสใดคลาสหนึ่ง และโมเดลจำแนกได้ถูกต้อง และF1-score ค่าเฉลี่ยฮาร์โมนิกของ Precision และ Recall

ในขั้นตอนการตรวจสอบ (Validate) ของระบบนั้น จะได้ Confusion Matrix ที่ได้จะอยู่ในรูปแบบของไฟล์รูปภาพสำหรับใช้ในการประเมินผลเบื้องต้น แต่หากต้องการใช้ Confusion Matrix ที่เป็นไฟล์ภาพดังกล่าว ดังรูปที่ 3.5 จึงต้องมีการแปลงภาพให้กลายเป็นข้อมูลด้วยวิธีการถัดไป



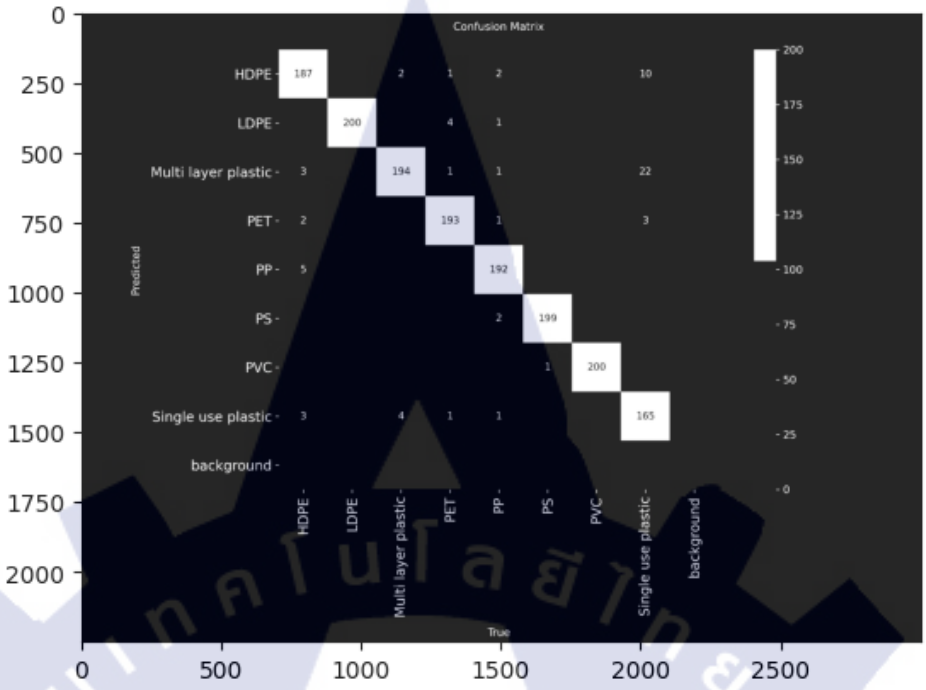
รูปที่ 3.5 ภาพ Confusion Matrix ที่ได้จากจาก Validate

3.4 การแปลง Confusion Matrix ที่เป็นภาพให้กลายเป็นข้อมูล

ด้วยกระบวนการทาง Image Processing จากเครื่องมือที่มีชื่อว่า “EasyOCR” หลักการทำงานของ EasyOCR [21] มีลักษณะเป็นกระบวนการแบบสองขั้นตอน ดังนี้

1. การตรวจจับข้อความ (Text Detection) EasyOCR จะทำหน้าที่ค้นหาและระบุตำแหน่งของข้อความจากภาพ โดยการสร้างขอบเขต (Bounding Boxes) รอบๆ ของข้อความที่พบและตัดภาพข้อความออกมาเพื่อนำไปประมวลผลต่อ

2. การรู้จำข้อความ (Text Recognition) เมื่อได้ตำแหน่งของข้อความแล้ว EasyOCR จะส่งภาพของแต่ละ Bounding Box ที่ทำการตรวจจับได้ ส่งไปยังโมเดลสำหรับขั้นตอนการรู้จำจะได้ผลลัพธ์ดังรูปที่ 3.6



รูปที่ 3.6 ภาพ Confusion Matrix ที่ผ่านวิธีการ OCR

3.5 การนำค่าตัวเลขที่ได้จากวิธี Image Processing ไปคำนวนหาค่าประสิทธิภาพ

เมื่อได้ค่าตัวเลขจากการทำ OCR แล้วจึงนำค่าไปประมวลผลต่อเพื่อคำนวนหาค่าประสิทธิภาพของโมเดล ซึ่งแบ่งออกเป็น 4 ค่า Accuracy, Precision, Recall,F1-Score ดังรูปที่ 3.7 เพื่อนำค่าที่ได้ไปใช้สำหรับเปรียบเทียบประสิทธิภาพของโมเดลYOLOv8 รุ่นต่างๆ

```
Precision per class: [    0.26076     0.40486      0.5452     0.95673   0.0036364   0.0024814]
Recall per class: [    0.49751     0.26882     0.58663     0.25911   0.0044643    0.029412]
F1 per class: [    0.34217      0.3231     0.56515     0.40779    0.004008   0.0045767]
Accuracy per class: [    0.69252     0.66493     0.88125     0.76889     0.80128     0.82607]
Overall Precision: 0.36227679251883593
Overall Recall: 0.2743244026138585
Overall Accuracy: 0.7724910035985606
Overall F1: 0.2744666446946349
```

รูปที่ 3.7 ผลลัพธ์ที่ได้จากการคำนวนค่าที่ได้จาก OCR

3.6 ทำการเปรียบเทียบค่าประสิทธิภาพกับโมเดล YOLOv8 ทุกรุ่น

เปรียบเทียบค่า Overall ของค่า Precision, Recall, Accuracy และ F1-score ของโมเดล YOLOv8 แต่ละรุ่น (YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, YOLOv8x) โดยจัดทำในรูปแบบตารางแสดงผลการเปรียบเทียบ

3.7 ประเมินและวิเคราะห์ผลการทดลอง

ในการประเมินประสิทธิภาพระบบการคัดแยกประเภทของพลาสติกสำหรับการรีไซเคิล งานวิจัยนี้ได้เลือกใช้ตัวชี้วัดการประเมินผล ได้แก่ จำนวนพารามิเตอร์ของโมเดล, Precision ,FLOPs, Accuracy, Precision, Recall, F1 score, top-1 accuracy, top-5 accuracy ทำการเปรียบเทียบค่าประสิทธิภาพกับโมเดล YOLOv8 ทุกรุ่น

สถาบันเทคโนโลยีไทย-ญี่ปุ่น THAI - NICHI INSTITUTE OF TECHNOLOGY TNI

บทที่ 4
ผลการวิจัย

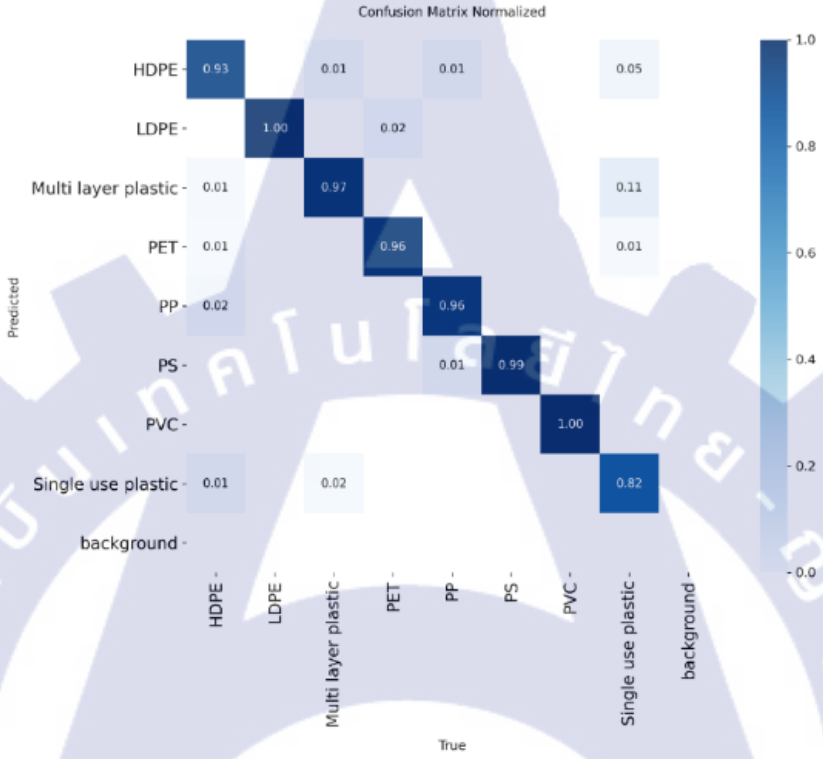
การศึกษาวิจัยเรื่องการพัฒนาและเปรียบเทียบประสิทธิภาพของโมเดล YOLOv8 ในการคัดแยกประเภทของพลาสติกสำหรับการรีไซเคิลด้วยการประมวลผลจากภาพถ่ายดิจิทัล มีวัตถุประสงค์เพื่อเพิ่มประสิทธิภาพการคัดแยกประเภทพลาสติกสำหรับการนำไปรีไซเคิล ดังนั้นผู้วิจัยจึงได้ศึกษาการนำเทคโนโลยีการคัดแยกประเภทในปัจจุบัน อย่างการเรียนรู้เชิงลึกมาใช้งาน โดยใช้ภาพขยะประเภทพลาสติกจำนวน 9156 ภาพ สำหรับการฝึนฝน ประเมินและทดสอบประสิทธิภาพของโมเดลโดยในการประเมินผลประสิทธิภาพของระบบการคัดแยกประเภทของพลาสติกสำหรับการรีไซเคิลจากนี้จะสามารถแบ่งการทดสอบได้ออกเป็น 2 การทดสอบดังต่อไปนี้

4.1 การประเมินด้วยค่าตัวชี้วัด

ในการประเมินประสิทธิภาพระบบการคัดแยกประเภทของพลาสติกสำหรับการรีไซเคิล งานวิจัยนี้ได้เลือกใช้ตัวชี้วัดการประเมินผล จำนวนพารามิเตอร์ของโมเดล, FLOPs, Accuracy, Precision, Recall, F1 score, top-1 accuracy, top-5 accuracy ที่ได้จากการทดลองมาทำการประเมินประสิทธิภาพระบบการคัดแยกประเภทของพลาสติกสำหรับการรีไซเคิล โดยจะต้องมีการหาค่า Overall (Accuracy, Precision, Recall, F1 score) ของ YOLOv8 แต่ละรุ่น (YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, YOLOv8x) ก่อนและจะให้ผลลัพธ์การทดสอบดังนี้

4.1.1 ผลการทดสอบด้วยการใช้โมเดล YOLOv8n (Nano)

การทดสอบด้วยการใช้โมเดล YOLOv8n (Nano) จะให้ผลลัพธ์ดังรูปที่ 4.1 และรูปที่ 4.2



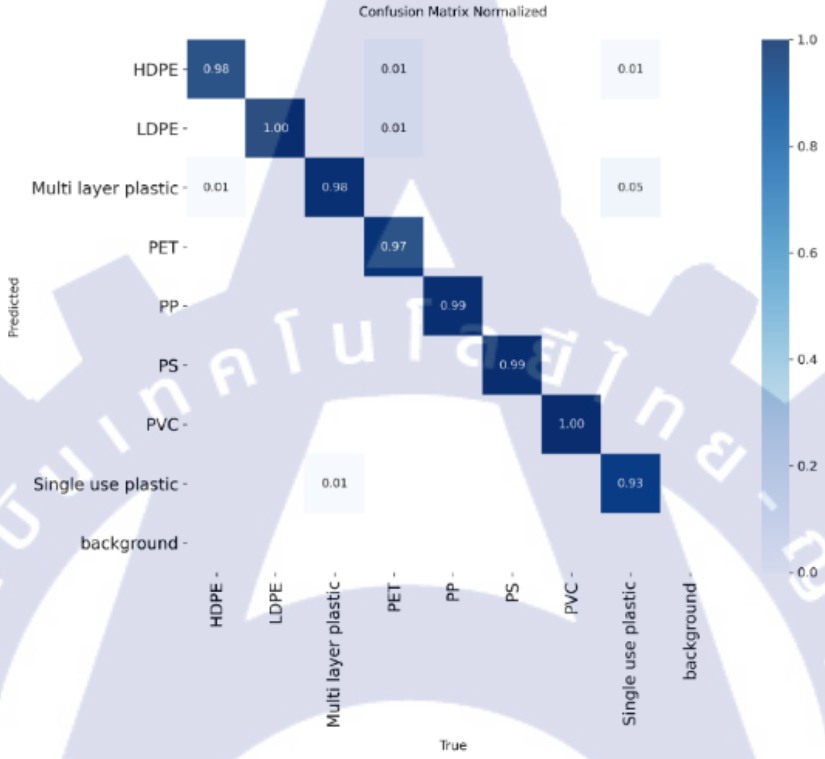
รูปที่ 4.1 Confusion Matrix ของการทดสอบด้วยโมเดล YOLOv8n

```
Precision per class: [    0.26076     0.40486      0.5452     0.95673   0.0036364   0.0024814]
Recall per class: [    0.49751     0.26882     0.58663     0.25911   0.0044643    0.029412]
F1 per class: [    0.34217      0.3231     0.56515     0.40779    0.004008   0.0045767]
Accuracy per class: [    0.69252     0.66493     0.88125     0.76889     0.80128     0.82607]
Overall Precision: 0.3622679251883593
Overall Recall: 0.2743244026138585
Overall Accuracy: 0.7724910035985606
Overall F1: 0.2744666446946349
```

รูปที่ 4.2 ค่า Confusion Matrix ของโมเดล YOLOv8n ที่ผ่านการประมวลผลแล้ว

4.1.2 ผลการทดสอบด้วยการใช้โมเดล YOLOv8s (Small)

การทดสอบด้วยการใช้โมเดล YOLOv8s (Small) จะให้ผลลัพธ์ดังรูปที่ 4.3 และรูปที่ 4.4



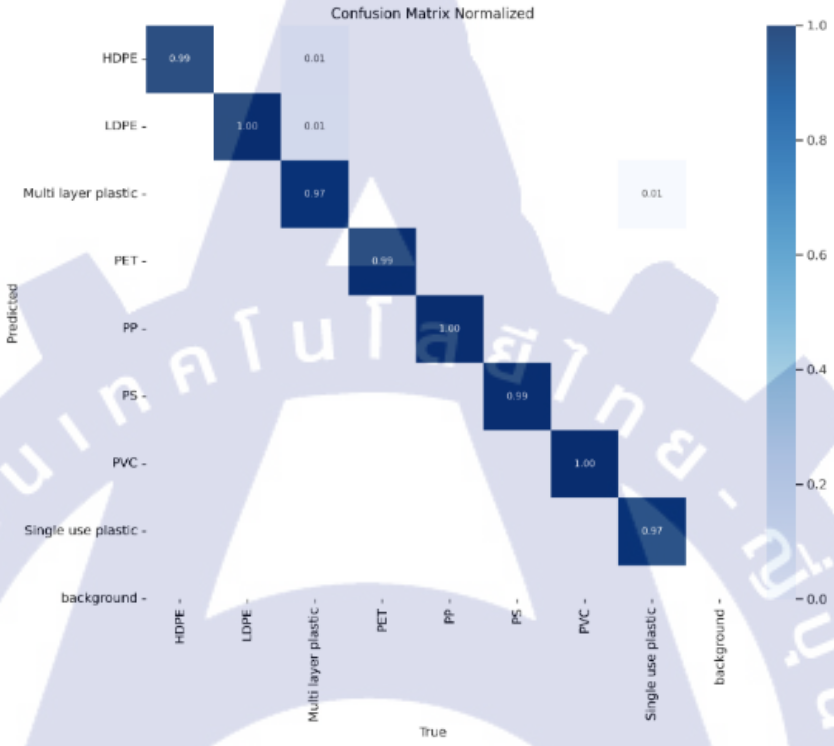
รูปที่ 4.3 Confusion Matrix ของการทดสอบด้วยโมเดล YOLOv8s

```
Precision per class: [    0.71429     0.16181    0.012195    0.002451    0.010274    0.014981]
Recall per class: [    0.25773     0.30928    0.001443   0.0030395    0.013761     0.36364]
F1 per class: [    0.37879     0.21246   0.0025806   0.0027137    0.011765    0.028777]
Accuracy per class: [    0.73999     0.55925     0.69362     0.70868     0.80024     0.78597]
Overall Precision: 0.15266656006244966
Overall Recall: 0.15814844265423925
Overall Accuracy: 0.7146254458977408
Overall F1: 0.10618141694835953
```

รูปที่ 4.4 ค่า Confusion Matrix ของโมเดล YOLOv8s ที่ผ่านการประมวลผลแล้ว

4.1.3 ผลการทดสอบด้วยการใช้โมเดล YOLOv8m (Medium)

การทดสอบด้วยการใช้โมเดล YOLOv8m (Medium) จะให้ผลลัพธ์ดังรูปที่ 4.5 และรูปที่ 4.6



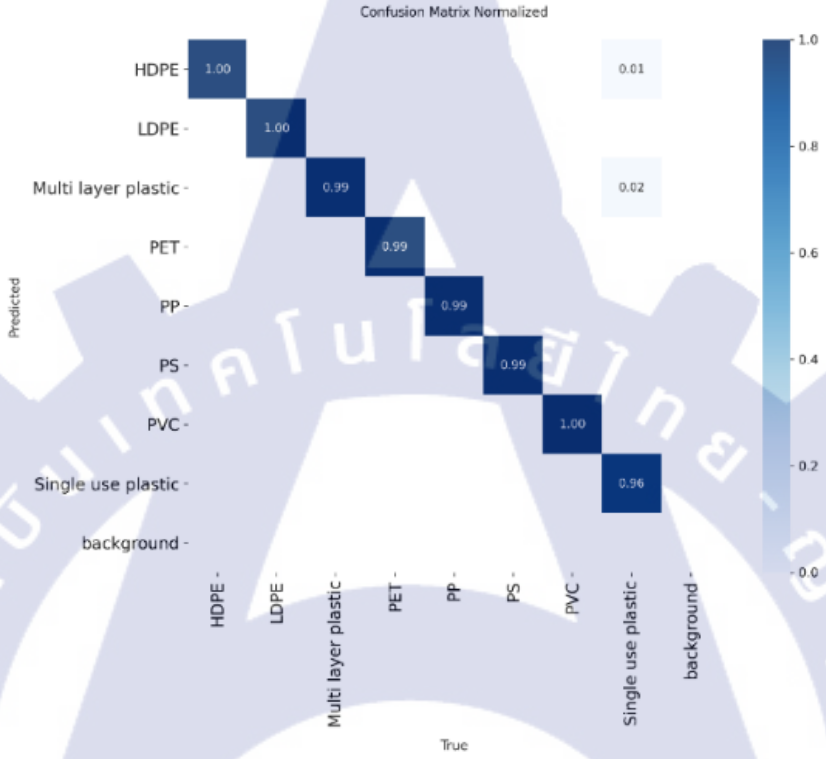
รูปที่ 4.5 Confusion Matrix ของการทดสอบด้วยโมเดล YOLOv8m

```
Precision per class: [      0.25     0.24917      0.0025     0.64784   0.0049751]
Recall per class: [    0.34602     0.22288   0.0017391     0.29148         0.2]
F1 per class: [    0.29028     0.23529   0.0020513     0.40206   0.0097087]
Accuracy per class: [    0.60958     0.61078     0.61158     0.76846     0.83713]
Overall Precision: 0.2308970182311036
Overall Recall: 0.21242446549285554
Overall Accuracy: 0.6875049900199601
Overall F1: 0.18787835104127937
```

รูปที่ 4.6 ค่า Confusion Matrix ของโมเดล YOLOv8m ที่ผ่านการประมวลผลแล้ว

4.1.4 ผลการทดสอบด้วยการใช้โมเดล YOLOv8l (Large)

การทดสอบด้วยการใช้โมเดล YOLOv8l (Large) จะให้ผลลัพธ์ดังรูปที่ 4.5 และรูปที่ 4.6



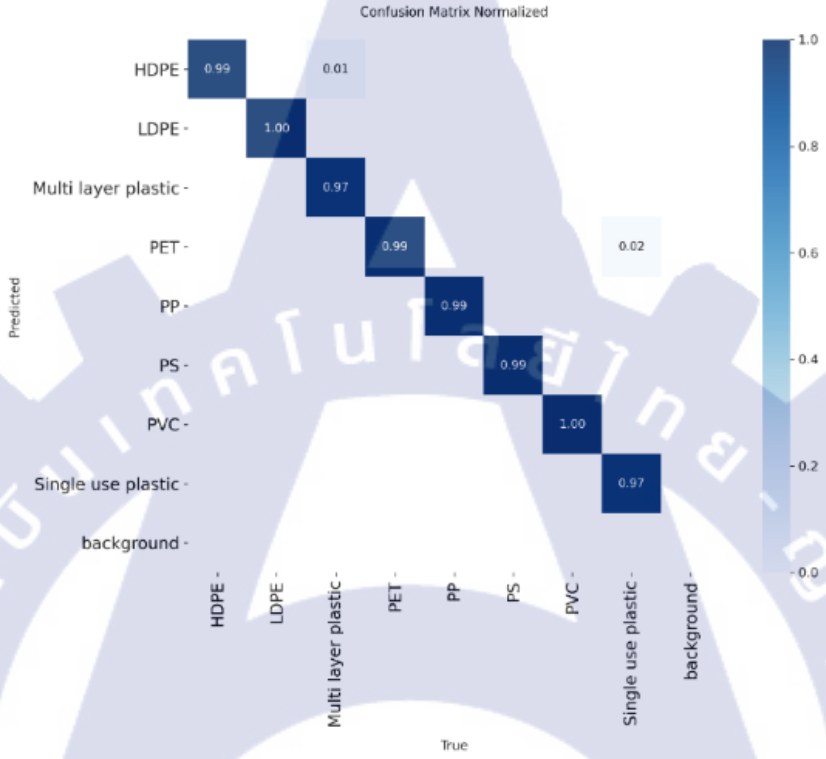
รูปที่ 4.7 Confusion Matrix ของการทดสอบด้วยโมเดล YOLOv8l

```
Precision per class: [    0.36101     0.30556     0.33113    0.047619   0.0041841]
Recall per class: [    0.25707     0.29247     0.17422    0.053305     0.22222]
F1 per class: [     0.3003     0.29887     0.22831    0.050302   0.0082136]
Accuracy per class: [    0.62824     0.62944     0.73036     0.62345     0.80734]
Overall Precision: 0.2098990723464985
Overall Recall: 0.19985586560572882
Overall Accuracy: 0.6837654567211807
Overall F1: 0.17719881806765894
```

รูปที่ 4.8 ค่า Confusion Matrix ของโมเดล YOLOv8l ที่ผ่านการประมวลผลแล้ว

4.1.5 ผลการทดสอบด้วยการใช้โมเดล YOLOv8x (Extra Large)

การทดสอบด้วยการใช้โมเดล YOLOv8x (Extra Large) จะให้ผลลัพธ์ดังรูปที่ 4.9 และรูปที่ 4.10



รูปที่ 4.9 Confusion Matrix ของการทดสอบด้วยโมเดล YOLOv8x

```
Precision per class: [       0.4     0.2008   0.0033445      0.5277   0.0052083]
Recall per class: [    0.2574     0.3178   0.0024876     0.32051    0.013274]
F1 per class: [   0.31323     0.2461   0.0028531      0.3988   0.0074813]
Accuracy per class: [    0.64934     0.63255     0.72051      0.7589     0.68173]
Overall Precision: 0.22741210265564332
Overall Recall: 0.18229431731096202
Overall Accuracy: 0.6886045581767293
Overall F1: 0.1936950917943303
```

รูปที่ 4.10 ค่า Confusion Matrix ของโมเดล YOLOv8x ที่ผ่านการประมวลผลแล้ว

หลังจากได้ค่า Overall (Accuracy, Precision, Recall, F1 score) ของ YOLOv8 แต่ละรุ่นแล้วจึงนำค่า จำนวนพารามิเตอร์ของโมเดล, FLOPs , F1 score, top-1 accuracy, top-5 accuracy เข้าไปใส่ซึ่งจะให้ผลลัพธ์ดังตารางที่ 4.1

ตารางที่ 4.1 ผลการทดลองก่อนที่จะทำการประเมินประสิทธิภาพ

Method	Params	FLOPs	Overall Accuracy	Overall Precision	Overall Recall	Overall F1 score	Top- 1 accuracy	Top- 5 accuracy
YOLOv8n	1.44m	3.3g	0.77	0.36	0.27	0.27	0.96	1
YOLOv8s	5.08m	12.5g	0.71	0.15	0.16	0.11	0.98	1
YOLOv8m	15.77m	41.6g	0.69	0.23	0.21	0.19	0.99	1
YOLOv8l	36.19m	98.7g	0.68	0.21	0.20	0.18	0.99	1
YOLOv8x	56.13m	153.8g	0.69	0.23	0.18	0.19	0.99	1

จากตารางที่ 4.1 ได้แสดงค่า Accuracy, Precision, Recall, F1-score, Top-1 accuracy และ Top-5 accuracy ของโมเดล YOLOv8ในแต่ละรุ่น ได้แก่ YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l และ YOLOv8x สรุปผลการเปรียบทียบประสิทธิภาพแล้วให้ผลดังนี้

1. ด้านของความถูกต้อง (Accuracy)โมเดล YOLOv8n ให้ค่า Accuracy สูงที่สุดที่ 0.77 ซึ่งให้ค่าที่มากกว่าโมเดลที่มีขนาดใหญ่กว่าอย่าง เช่น YOLOv8s (0.71), YOLOv8m (0.69), YOLOv8l (0.68) และ YOLOv8x (0.69) ที่กลับมีค่า Accuracy ลดลงอย่างเห็นได้ชัด แสดงให้เห็นว่าการเพิ่มขนาดโมเดลและจำนวนพารามิเตอร์ ไม่ได้ช่วยเพิ่มความถูกต้องโดยรวม

2. ส่วนของผลลัพธ์ด้านความแม่นยำ (Precision) และความครอบคลุม (Recall) ค่า Precision ของ YOLOv8n = 0.36 ซึ่งสูงสุดในกลุ่มค่าและ Recall ของ YOLOv8n ก็สูงที่สุดเช่นกันที่ 0.27 ในขณะที่รุ่นอื่น ๆ อยู่ในช่วง 0.16–0.23 แสดงว่า YOLOv8n มีความสามารถในการทำนายผลที่ถูกต้องและครอบคลุมดีกว่ารุ่นอื่น

3. ค่า F1-score ของ YOLOv8n ให้ค่า F1-score สูงที่สุดที่ 0.27 ซึ่งจะให้ผลลัพธ์สอดคล้องกับ Precision และ Recall ที่จะมีค่าสูงกว่าโมเดลอื่น ที่มีค่า F1-score ต่ำกว่าอย่างชัดเจน เช่น YOLOv8s (0.11), YOLOv8m (0.19), YOLOv8l (0.18), YOLOv8x (0.19)

4. ผลลัพธ์ด้าน Top-1 และ Top-5 Accuracy ทุกๆ โมเดลจะมีค่า Top-5 Accuracy = 1.0 เท่ากัน หมายความว่า ผลลัพธ์ลัพธ์ที่ทำนายถูกต้องมักจะปรากฏใน 5 อันดับแรกเสมอ ส่วนค่าของ Top-1 Accuracy ของทุกโมเดลจะมีค่าใกล้เคียงกัน (0.96–0.99) หมายความว่าโมเดลทุกตัวมีความสามารถในการจำแนกประเภทของพลาสติกที่ถูกต้องในอันดับหนึ่งใกล้เคียงกัน

5. ในส่วนของขนาดโมเดลและการใช้ทรัพยากร (Params และ FLOPs) โมเดล YOLOv8n ใช้พารามิเตอร์เพียง 1.44M และ FLOPs 3.3G ซึ่งเล็กที่สุดในบรรดาโมเดลทั้งหมด ในขณะที่ YOLOv8x ใช้พารามิเตอร์สูงถึง 56.13M และ FLOPs 153.8G แต่ก็ไม่ได้ให้ผลลัพธ์ที่ดีกว่า YOLOv8n เท่าที่ควร จึงแสดงให้เห็นว่า YOLOv8n มีประสิทธิภาพเชิงการเปรียบเทียบสูงที่สุดเมื่อเปรียบเทียบกับประสิทธิภาพที่ได้ โดยผลการประเมินประสิทธิภาพตามตารางที่ 4.2

ตารางที่ 4.2 ผลการทดลองหลังจากทำการประเมินประสิทธิภาพและเรียงลำดับ

Method	Params	FLOPs	Overall Accuracy	Overall Precision	Overall Recall	Overall F1 score	Top- 1 accuracy	Top- 5 accuracy
YOLOv8n	1.44m	3.3g	0.77	0.36	0.27	0.27	0.96	1
YOLOv8m	15.77m	41.6g	0.69	0.23	0.21	0.19	0.99	1
YOLOv8x	56.13m	153.8g	0.69	0.23	0.18	0.19	0.99	1
YOLOv8l	36.19m	98.7g	0.68	0.21	0.20	0.18	0.99	1
YOLOv8s	5.08m	12.5g	0.71	0.15	0.16	0.11	0.98	1

4.2 การประเมินด้วยผลการทดสอบจากการนำเข้ารูปภาพ

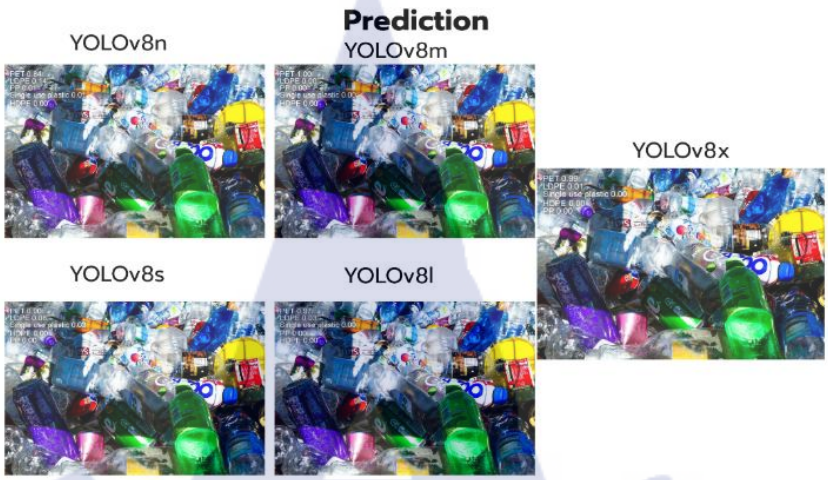
ผู้วิจัยได้ทำการทดลองนำเข้าภาพที่มีองค์ประกอบของวัตถุหลากหลายชนิด เช่น พลาสติกที่สกปรก เปียก หรืออยู่ปะปนกับพื้นผิวอื่นๆ โดยพลาสติกส่วนใหญ่ในภาพนั้นจะเป็นพลาสติกชนิดต่างๆ 7 ประเภท แล้วทำการทดลองหาผลการทำนายจากภาพว่าโมเดล YOLOv8 แต่ละรุ่นจะแสดงผลลัพธ์ค่าความมั่นใจแค่ไหนว่าการทำนายนั้นถูกต้อง ซึ่งค่านี้จะอยู่ระหว่าง 0 ถึง 1 ยิ่งใกล้ 1 ยิ่งแม่นยำ โดยจะแสดงผลเป็นกล่องล้อมรอบวัตถุ พร้อมระบุชนิดของพลาสติกและค่าความแม่นยำที่โมเดลทำนายได้ ดังนี้

4.2.1 ผลการทดสอบในการระบุพลาสติกประเภทพอลีเอทิลีนเทเรฟทาเลต (Polyethylene Terephthalate - PET) โดยรูปภาพที่นำเข้าจะใช้รูปที่ 4.11

รูปที่ 4.11 ภาพพลาสติก PET ที่ใช้ในการทดสอบประสิทธิภาพของโมเดล YOLOv8

จากการทดสอบดังกล่าวดังรูปที่ 4.12 พบว่า YOLOv8m เป็นรุ่นที่ทำนายชนิดพลาสติกนี้ได้แม่นยำมากที่สุด ค่าความแม่นยำ 1 ซึ่งทำนายได้ถูกต้องว่าพลาสติกในรูปเป็นชนิด PET ตามมาด้วยYOLOv8x และ YOLOv8l ก็มีความแม่นยำสูงมากเช่นกัน ค่าความแม่นยำ 0.99 และ 0.97 ในส่วนของ YOLOv8s และ YOLOv8n นั้นให้ผลลัพธ์ที่น่าพอใจ ก็คือ 0.90 และ 0.84 และจากกราฟ ดังรูปที่ 4.13 จะชี้ให้เห็นว่าทุกๆ การใช้โมเดลที่มีขนาดใหญ่ขึ้นจะทำให้ค่าความแม่นยำเพิ่มขึ้น แต่ทว่า โมเดล YOLOv8l และ YOLOv8x ซึ่งเป็นรุ่นที่ใหญ่กว่า YOLOv8m กลับมีประสิทธิภาพลดลงเล็กน้อยที่ 0.97 และ 0.99 ตามลำดับแสดงให้เห็นว่าการเพิ่มขนาดโมเดลไม่ได้หมายถึงประสิทธิภาพจะเพิ่มขึ้นเสมอไป อาจเกิดจากปัญหา โมเดลจดจำข้อมูลที่ใช้ในการฝึกมากเกินไปจนไม่สามารถทำงานได้ดีกับข้อมูลใหม่ ดังนั้นจากการคำนวนค่าเฉลี่ยของค่าความถูกต้องของกราฟด้วยโมเดล YOLOv8 ทั้ง 5 ตัวจะมีค่าอยู่ที่ 0.94 หรือ 94% สำหรับงานในการทำนายประเภทพลาสติกชนิด PET โมเดลมีความเสถียรอย่างมาก

ในการเปรียบเทียบนี้ชี้ให้เห็นว่า YOLOv8m ที่เป็นโมเดลที่สุดในรุ่นของ YOLOv8 สามารถให้ผลลัพธ์การทำนายที่แม่นยำที่สุดสำหรับการจำแนกพลาสติกชนิดพอลีเอทิลีนเทเรฟทาเลต (Polyethylene Terephthalate - PET)



รูปที่ 4.12 ผลลัพธ์การทำนายพลาสติก PET ด้วยโมเดล YOLOv8 ทุกรุ่น

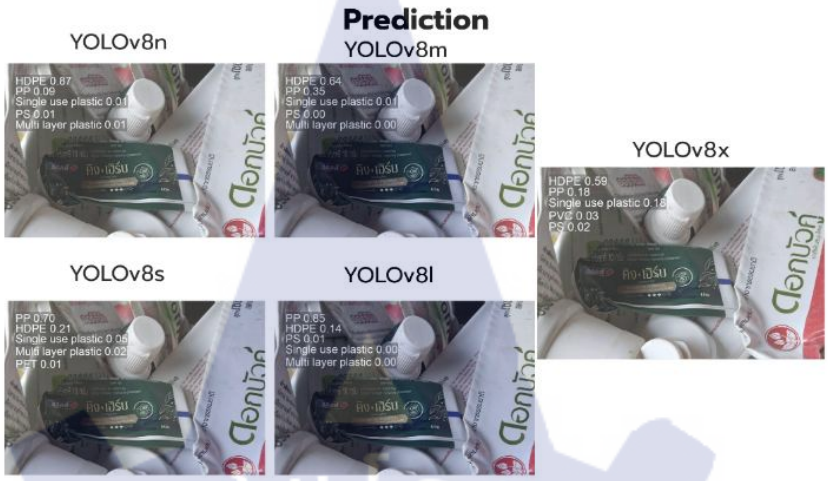
รูปที่ 4.13 กราฟเส้นค่าความแม่นยำในการทำนายพลาสติก PET ของโมเดลแต่ละรุ่น

4.2.2 ผลการทดสอบในการระบุพลาสติกประเภทพอลีเอทิลีนความหนาแน่นสูง (High-Density Polyethylene - HDPE) โดยรูปภาพที่นำเข้าจะใช้รูปที่ 4.14

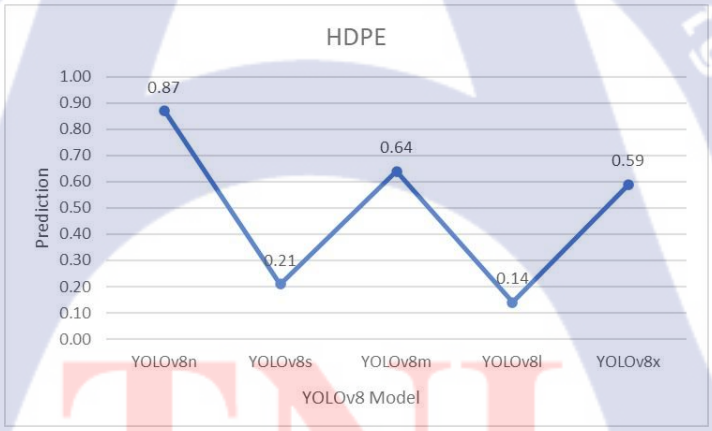
รูปที่ 4.14 ภาพพลาสติก HDPE ที่ใช้ในการทดสอบประสิทธิภาพของโมเดล YOLOv8

จากการทดสอบดังกล่าวดังรูปที่ 4.15 พบว่า YOLOv8n เป็นรุ่นที่ทำนายชนิดพลาสติกนี้ได้แม่นยำมากที่สุด ค่าความแม่นยำที่ 0.85 ซึ่งทำนายได้ถูกต้องว่าพลาสติกในรูปเป็นชนิด HDPE ตามมาด้วยYOLOv8l ก็มีความแม่นยำสูงมากเช่นกัน ค่าความแม่นยำที่ 0.85 และ YOLOv8s และ YOLOv8m ค่าความแม่นยำที่ 0.70 และ 0.64 ตามลำดับ ถือว่าให้ผลลัพธ์ที่น่าพอใจ ส่วน YOLOv8x ให้ค่าความแม่นยำที่ 0.59 เป็นความแม่นยำที่อยู่ในช่วง 50% ซึ่งควรปรับปรุงและจากกราฟ ดังรูปที่ 4.16 จะชี้ให้เห็นว่ารูปแบบของกราฟนี้มีการผันผวนอย่างรุนแรง ชี้ถึงปัญหาดังนี้คือ สำหรับโมเดล YOLOv8l อาจมีการเกิด Overfitting โมเดลเรียนรู้จากข้อมูลที่ใช้ฝึกฝนได้ดีเกินไป สามารถจดจำรายละเอียดเล็กๆได้ ทำให้เมื่อนำโมเดลไปใช้กับข้อมูลใหม่ โมเดลจะทำนายผลได้แย่ลงอย่างเห็นได้ชัดทำให้ไม่สามารถทำนายได้อย่างแม่นยำ ในส่วนของ YOLOv8s ก็เกิดจากอีกปัจจัยก็คือ ชุดข้อมูลที่ใช้ฝึกอาจไม่เพียงพอด้วยจำนวนรอบการฝึก (Epochs) น้อยเกินไป ทำให้โมเดลไม่สามารถเรียนรู้ลักษณะเฉพาะของพลาสติกชนิดพอลีเอทิลีนความหนาแน่นสูงได้เพียงพอ ดังนั้นจากการคำนวนค่าเฉลี่ยของค่าความถูกต้องของกราฟด้วยโมเดล YOLOv8 ทั้ง 5 ตัวจะมีค่าอยู่ที่ 0.49 หรือ 49% สำหรับงานในการทำนายประเภทพลาสติกชนิด HDPE มีความเสถียรค่อนข้างต่ำ

ในการเปรียบเทียบนี้ชี้ให้เห็นว่า YOLOv8n ที่เป็นโมเดลที่สุดในรุ่นของ YOLOv8 สามารถให้ผลลัพธ์การทำนายที่แม่นยำที่สุดสำหรับการจำแนกพลาสติกชนิดพอลีเอทิลีนความหนาแน่นสูง (High-Density Polyethylene - HDPE)



รูปที่ 4.15 ผลลัพธ์การทำนายพลาสติก HDPE ด้วยโมเดล YOLOv8 ทุกรุ่น



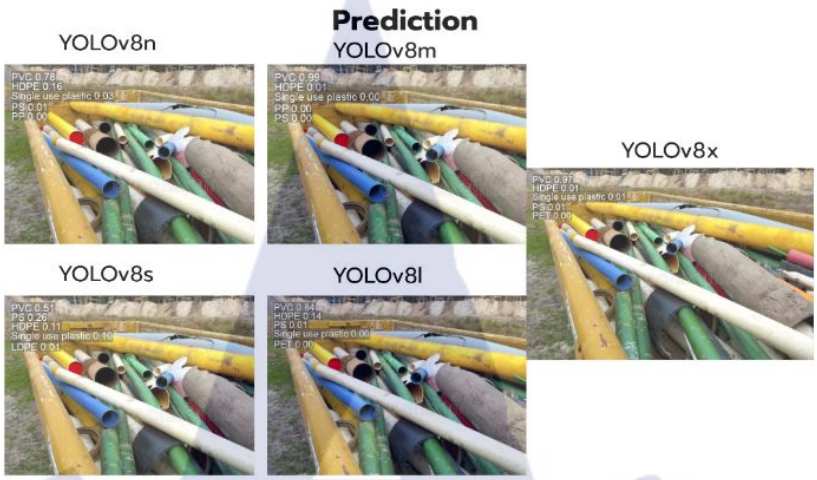
รูปที่ 4.16 กราฟเส้นค่าความแม่นยำในการทำนายพลาสติก HDPE ของโมเดลแต่ละรุ่น

4.2.3 ผลการทดสอบในการระบุพลาสติกประเภทพอลีไวนิลคลอไรด์ (Polyvinyl Chloride - PVC) โดยรูปภาพที่นำเข้าจะใช้รูปที่ 4.17

รูปที่ 4.17 ภาพพลาสติก PVC ที่ใช้ในการทดสอบประสิทธิภาพของโมเดล YOLOv8

จากการทดสอบดังกล่าวดังรูปที่ 4.18 พบว่า YOLOv8m เป็นรุ่นที่ทำนายชนิดพลาสติกได้แม่นยำมากที่สุด ค่าความแม่นยำที่ 0.99 ซึ่งทำนายได้ถูกต้องว่าพลาสติกในรูปเป็นชนิด PVC ตามมาด้วยYOLOv8x YOLOv8l และ YOLOv8n ที่ให้ค่าความแม่นยำที่ 0.97 0.84 และ 0.78 ตามลำดับ ซึ่งอยู่ในระดับที่น่าพึงพอใจอย่างมากแต่ในส่วนของ YOLOv8s ให้ค่าความแม่นยำที่ 0.51 เป็นความแม่นยำที่อยู่ในช่วง 50% ซึ่งถือว่าควรปรับปรุงและจากกราฟดังรูปที่ 4.19 พบว่า YOLOv8s มีประสิทธิภาพลดลงอย่างมากแสดงให้เห็นว่าในกรณีนี้อาจเกิดปัญหา Underfitting โมเดลเรียนรู้จากข้อมูลที่ใช้ฝึกฝนได้เรียบง่ายเกินไป จนไม่สามารถเรียนรู้รูปแบบพื้นฐานที่ซ่อนอยู่ในข้อมูลได้ ทำให้ประสิทธิภาพการทำนายแย่ตั้งแต่ขั้นตอนการฝึกฝน และเมื่อนำไปใช้กับข้อมูลใหม่ก็ทำนายได้ไม่ดี ดังนั้นจากการคำนวนค่าเฉลี่ยของค่าความถูกต้องของกราฟด้วยโมเดล YOLOv8 ทั้ง 5 ตัวจะมีค่าอยู่ที่ 0.81 หรือ 81% สำหรับงานในการทำนายประเภทพลาสติกชนิด PVC โมเดลมีความเสถียรอย่างมาก

ในการเปรียบเทียบนี้ชี้ให้เห็นว่า YOLOv8m ที่เป็นโมเดลที่สุดในรุ่นของ YOLOv8 สามารถให้ผลลัพธ์การทำนายที่แม่นยำที่สุดสำหรับการจำแนกพลาสติกชนิดพอลีไวนิลคลอไรด์ (Polyvinyl Chloride - PVC)



รูปที่ 4.18 ผลลัพธ์การทำนายพลาสติก PVC ด้วยโมเดล YOLOv8 ทุกรุ่น

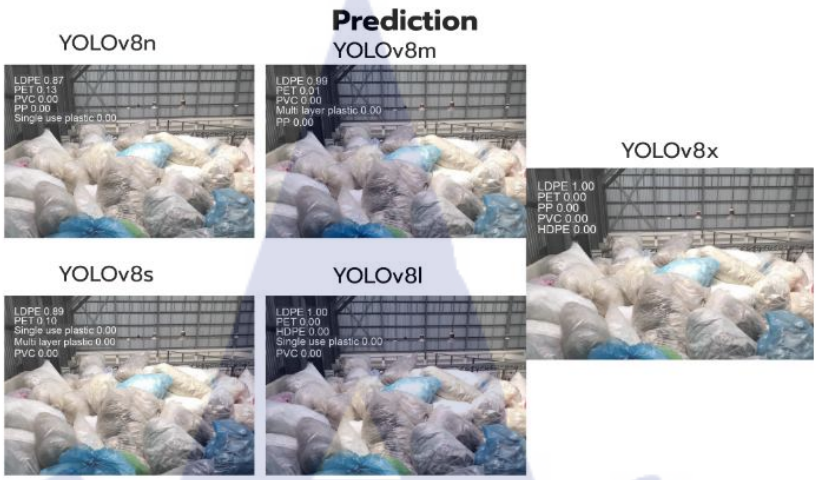
รูปที่ 4.19 กราฟเส้นค่าความแม่นยำในการทำนายพลาสติก PVC ของโมเดลแต่ละรุ่น

4.2.4 ผลการทดสอบในการระบุพลาสติกประเภทพอลีเอทิลีนความหนาแน่นต่ำ (Low-Density Polyethylene - LDPE) โดยรูปภาพที่นำเข้าจะใช้รูปที่ 4.20

รูปที่ 4.20 ภาพพลาสติก LDPE ที่ใช้ในการทดสอบประสิทธิภาพของโมเดล YOLOv8

จากการทดสอบดังกล่าวดังรูปที่ 4.21 พบว่า YOLOv8x และ YOLOv8l เป็นรุ่นที่ทำนายชนิดพลาสติกได้แม่นยำมากที่สุด ค่าความแม่นยำที่ 1 ซึ่งทำนายได้ถูกต้องว่าพลาสติกในรูปเป็นชนิด LDPE ตามมาด้วยYOLOv8m ที่ให้ค่าความแม่นยำที่ 0.99 ซึ่งใกล้เคียงมากกับโมเดล 2 ตัวแรกและ YOLOv8s และ YOLOv8n ก็ให้ค่าความแม่นยำที่ 0.89 และ 0.87 ตามลำดับซึ่งถือเป็นผลลัพธ์ที่พอใจอย่างมากและจากกราฟ ดังรูปที่ 4.22 จะชี้ให้เห็นว่า การจำแนกประเภทพลาสติกชนิดพอลีเอทิลีนความหนาแน่นต่ำ เป็นงานที่โมเดล YOLOv8 สามารถทำได้อย่างมีประสิทธิภาพมากจากแนวโน้มของกราฟที่มีการเพิ่มขึ้นอย่างต่อเนื่องเมื่อใช้โมเดลที่มีขนาดใหญ่ สิ่งนี้บ่งชี้ว่าข้อมูลพลาสติกชนิดพอลีเอทิลีนความหนาแน่นต่ำอาจมีลักษณะเฉพาะที่สามารถเรียนรู้ได้ง่ายกว่าพลาสติกประเภทอื่น ๆ ดังนั้นจากการคำนวนค่าเฉลี่ยของค่าความถูกต้องของกราฟด้วยโมเดล YOLOv8 ทั้ง 5 ตัวจะมีค่าอยู่ที่ 0.95 หรือ 95% สำหรับงานในการทำนายประเภทพลาสติกชนิด LDPE โมเดลมีความเสถียรอย่างมาก

ดังนั้นในการเปรียบเทียบนี้ชี้ให้เห็นว่า YOLOv8x และ YOLOv8l ที่เป็นโมเดลที่สุดในรุ่นของ YOLOv8 สามารถให้ผลลัพธ์การทำนายที่แม่นยำที่สุดสำหรับการจำแนกพลาสติกชนิดพอลีเอทิลีนความหนาแน่นต่ำ (Low-Density Polyethylene - LDPE)



รูปที่ 4.21 ผลลัพธ์การทำนายพลาสติก LDPE ด้วยโมเดล YOLOv8 ทุกรุ่น

รูปที่ 4.22 กราฟเส้นค่าความแม่นยำในการทำนายพลาสติก LDPE ของโมเดลแต่ละรุ่น

4.2.5 ผลการทดสอบในการระบุพลาสติกประเภทพอลีโพรพิลีน (Polypropylene - PP) โดยรูปภาพที่นำเข้าจะใช้รูปที่ 4.23

รูปที่ 4.23 ภาพพลาสติก PP ที่ใช้ในการทดสอบประสิทธิภาพของโมเดล YOLOv8

จากการทดสอบดังกล่าวดังรูปที่ 4.24 พบว่า YOLOv8m และ YOLOv8l เป็นรุ่นที่ทำนายชนิดพลาสติกได้แม่นยำมากที่สุด ค่าความแม่นยำที่ 1 ซึ่งทำนายได้ถูกต้องว่าพลาสติกในรูปเป็นชนิด PP ตามมาด้วยYOLOv8s และ YOLOv8x ที่ให้ค่าความแม่นยำที่ 0.99 ซึ่งใกล้เคียงมากกับโมเดล 2 และ YOLOv8n ก็ให้ค่าความแม่นยำที่ 0.96 ซึ่งถือเป็นผลลัพธ์ที่พอใจอย่างมากและจากกราฟ ดังรูปที่ 4.25 พบว่าการจำแนกชนิดพอลีโพรพิลีนเป็นงานที่โมเดล YOLOv8 สามารถทำได้ดีเยี่ยมแต่ค่าความแม่นยำที่ลดลงเล็กน้อยของ YOLOv8x อาจชี้ให้เห็นว่าโมเดลนี้เริ่มมีความซับซ้อนเกินความจำเป็นสำหรับงานนี้ แต่ก็ยังคงอยู่ในระดับที่ยอมรับได้ ดังนั้นจากการคำนวนค่าเฉลี่ยของค่าความถูกต้องของกราฟด้วยโมเดล YOLOv8 ทั้ง 5 ตัวจะมีค่าอยู่ที่ 0.99 หรือ 99% สำหรับงานในการทำนายประเภทพลาสติกชนิด PP โมเดลมีความเสถียรอย่างมากที่สุด

ดังนั้นในการเปรียบเทียบนี้ชี้ให้เห็นว่า YOLOv8m และ YOLOv8l ที่เป็นโมเดลที่สุดในรุ่นของ YOLOv8 สามารถให้ผลลัพธ์การทำนายที่แม่นยำที่สุดสำหรับการจำแนกพลาสติกชนิดพอลีโพรพิลีน (Polypropylene - PP)

รูปที่ 4.24 ผลลัพธ์การทำนายพลาสติก PP ด้วยโมเดล YOLOv8 ทุกรุ่น

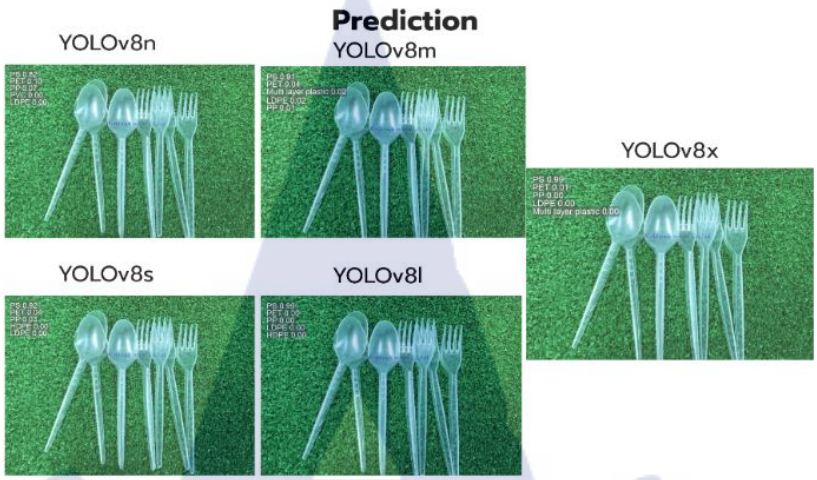
รูปที่ 4.25 กราฟเส้นค่าความแม่นยำในการทำนายพลาสติก PP ของโมเดลแต่ละรุ่น

4.2.6 ผลการทดสอบในการระบุพลาสติกประเภทพอลีสไตรีน (Polystyrene - PS) โดยรูปภาพที่นำเข้าจะใช้รูปที่ 4.26

รูปที่ 4.26 ภาพพลาสติก PS ที่ใช้ในการทดสอบประสิทธิภาพของโมเดล YOLOv8

จากการทดสอบดังกล่าวดังรูปที่ 4.27 พบว่า YOLOv8l และ YOLOv8x เป็นรุ่นที่ทำนายชนิดพลาสติกได้แม่นยำมากที่สุด ค่าความแม่นยำที่ 0.99 ซึ่งทำนายได้ถูกต้องว่าพลาสติกในรูปเป็นชนิด PS ตามมาด้วยYOLOv8s และ YOLOv8m ที่ให้ค่าความแม่นยำที่ 0.92 และ 0.91 ตามลำดับ ซึ่งใกล้เคียงมากกับโมเดล 2 แรกและ YOLOv8n ก็ให้ค่าความแม่นยำที่ 0.82 ซึ่งถือเป็นผลลัพธ์ที่พอใจอย่างมากและจากกราฟ ดังรูปที่ 4.28 การจำแนกประเภทพลาสติกชนิดพอลีสไตรีนเป็นงานที่โมเดล YOLOv8 สามารถทำได้อย่างน่าพอใจ การที่ YOLOv8m มีประสิทธิภาพต่ำกว่า YOLOv8s เล็กน้อยนั้นอาจเกิดจากความผันผวนในการฝึกฝนโมเดล แต่โดยรวมแล้วแนวโน้มประสิทธิภาพก็ยังคงเป็นบวก ดังนั้นจากการคำนวนค่าเฉลี่ยของค่าความถูกต้องของกราฟด้วยโมเดล YOLOv8 ทั้ง 5 ตัวจะมีค่าอยู่ที่ 0.93 หรือ 93% สำหรับงานในการทำนายประเภทพลาสติกชนิด PS โมเดลมีความเสถียรอย่างมาก

ดังนั้นในการเปรียบเทียบนี้ชี้ให้เห็นว่า YOLOv8l และ YOLOv8x ที่เป็นโมเดลที่สุดในรุ่นของ YOLOv8 สามารถให้ผลลัพธ์การทำนายที่แม่นยำที่สุดสำหรับการจำแนกพลาสติกชนิดพอลีสไตรีน (Polystyrene - PS)



รูปที่ 4.27 ผลลัพธ์การทำนายพลาสติก PS ด้วยโมเดล YOLOv8 ทุกรุ่น

รูปที่ 4.28 กราฟเส้นค่าความแม่นยำในการทำนายพลาสติก PS ของโมเดลแต่ละรุ่น

4.2.7 ผลการทดสอบในการระบุพลาสติกประเภทพลาสติกอื่น ๆ (Other - O) โดยรูปภาพที่นำเข้าจะใช้รูปที่ 4.29

รูปที่ 4.29 ภาพพลาสติก O(Other) ที่ใช้ในการทดสอบประสิทธิภาพของโมเดล YOLOv8

จากการทดสอบดังกล่าวดังรูปที่ 4.30 พบว่า YOLOv8n เป็นรุ่นที่ทำนายชนิดพลาสติกได้แม่นยำมากที่สุด ค่าความแม่นยำที่ 0.95 ซึ่งทำนายได้ถูกต้องว่าพลาสติกในรูปเป็นชนิด Other (Single use plastic) ตามมาด้วย YOLOv8x ที่ให้ค่าความแม่นยำที่ 0.91 ซึ่งใกล้เคียงมากกับโมเดลแรกมากๆ และ YOLOv8s และ YOLOv8l ก็ให้ค่าความแม่นยำที่ 0.87 และ 0.82 ตามลำดับ ซึ่งถือเป็นผลลัพธ์ที่พอใจอย่างมาก แต่ว่า YOLOv8m นั้นมีการทำนายพลาสติกเป็นพลาสติกชนิด HDPE ที่ความแม่นยำ 0.74 และทำนายว่าเป็น Other เพียง 0.26 ซึ่งถือว่าให้ผลลัพธ์ไม่ตรงกับความต้องการและจากกราฟ ดังรูปที่ 4.31 พบว่าค่าของกราฟมีการหักเหลงอย่างมากในการทำงานของโมเดล YOLOv8m ชี้ให้เห็นว่าโมเดลนี้อาจมีการเรียนรู้ที่ไม่เหมาะสมสำหรับข้อมูลประเภทนี้ ซึ่งอาจเกิดจากปัญหา Overfitting หรือการตั้งค่าที่ไม่เหมาะสม ดังนั้นจากการคำนวนค่าเฉลี่ยของค่าความถูกต้องของกราฟด้วยโมเดล YOLOv8 ทั้ง 5 ตัวจะมีค่าอยู่ที่ 0.76 หรือ 76% สำหรับงานในการทำนายประเภทพลาสติกชนิด O(Other) โมเดลมีความเสถียรอยู่ในระดับพอใจ

ดังนั้นในการเปรียบเทียบนี้ชี้ให้เห็นว่า YOLOv8n ที่เป็นโมเดลที่สุดในรุ่นของ YOLOv8 สามารถให้ผลลัพธ์การทำนายที่แม่นยำที่สุดสำหรับการจำแนกพลาสติกพลาสติกอื่น ๆ (Other - O)

รูปที่ 4.30 ผลลัพธ์การทำนายพลาสติก O(Other) ด้วยโมเดล YOLOv8 ทุกรุ่น

รูปที่ 4.31 กราฟเส้นค่าความแม่นยำในการทำนายพลาสติก O(Other) ของโมเดลแต่ละรุ่น

จากการทดลองดังกล่าว ผู้วิจัยได้นำค่าที่ได้จากการเปรียบเทียบโมเดลมาจัดทำให้อยู่ในรูปแบบของตารางเพื่อเปรียบเทียบค่าความแม่นยำของแต่ละโมเดลที่ได้ประมวลผลจากภาพพลาสติกแต่ละชนิด และได้ผลลัพธ์ดังตารางที่ 4.3

ตารางที่ 4.3 ผลเปรียบเทียบค่าความแม่นยำของแต่ละโมเดลที่ได้ประมวลผลจากภาพพลาสติกแต่ละชนิด

Method	YOLOv8n	YOLOv8s	YOLOv8m	YOLOv8l	YOLOv8x
PET	0.84	0.90	1.00	0.97	0.99
HDPE	0.87	0.21	0.64	0.14	0.59
PVC	0.78	0.51	0.99	0.84	0.97
LDPE	0.87	0.89	0.99	1.00	1.00
PP	0.96	0.99	1.00	1.00	0.99
PS	0.82	0.92	0.91	0.99	0.99
O	0.95	0.87	0.26	0.82	0.91
Avg.	0.87	0.76	0.83	0.82	0.92

จากตารางที่ 4.3 จะสรุปได้ว่าจากค่าเฉลี่ยความแม่นยำของโมเดล YOLOv8 ทุกรุ่นจะมีค่าอยู่ที่ 0.84 โดยโมเดลที่มีค่ามากกว่าค่าเฉลี่ยจะเป็นโมเดลที่มีความเสถียรในการนำมาใช้งาน ดังนั้นจึงสรุปได้ว่า YOLOv8x เป็นโมเดลที่ดีที่สุด เนื่องจากการคำนวนค่าเฉลี่ยของทุกคลาสได้ค่าเท่ากับ YOLOv8x ประมาณ 0.92 และ YOLOv8n ประมาณ 0.87 เป็นโมเดลที่รองลงมา แต่พบว่า YOLOv8m ประมาณ 0.83, YOLOv8l ประมาณ 0.82 และ YOLOv8s ประมาณ 0.76 นั้นมีค่าต่ำกว่าค่าเฉลี่ยของโมเดล YOLOv8 ทุกรุ่นจึงสรุปได้ว่าทั้ง 3 รุ่นดังกล่าวเป็นโมเดลที่ไม่เสถียรต่อการนำใช้งานจริง ดังนั้นผลการประเมินจากการทดสอบจากการนำเข้ารูปภาพ YOLOv8x มีประสิทธิภาพสูงที่สุด

บทที่ 5
บทสรุป อภิปราย และข้อเสนอแนะ

การคัดแยกประเภทของพลาสติกเพื่อการรีไซเคิลจากภาพถ่ายด้วยวิธีการเรียนรู้เชิงลึก ซึ่งการวิจัยนี้มีวัตถุประสงค์ตามบทที่ 1 ได้นำเสนอข้อสรุปตาม

5.1 สรุปผลการวิจัย

5.2 อภิปรายผลวิจัย

5.3 ปัญหาและอุปสรรคในการทำวิจัย

5.4 ข้อเสนอแนะงานวิจัย

5.1 สรุปผลการวิจัย

ในการสรุปผลงานวิจัย ผู้จัดทำวิจัยได้ทำสรุปผลตามวัตถุประสงค์ของงานวิจัยในบทที่ 1 ดังนี้

สำหรับการพัฒนาระบบคัดแยกประเภทของขยะพลาสติกโดยใช้ปัญญาประดิษฐ์ (AI) และเทคนิคการเรียนรู้เชิงลึกจากภาพถ่ายนั้น ผู้วิจัยได้ทำการวิเคราะห์ว่าจะนำโมเดลไหนมาใช้ จากที่ได้ศึกษาและเรียนรู้มาพบว่าโมเดล YOLOv8 เป็นโมเดลที่การตรวจจับหรือคัดแยกประเภทวัตถุที่มีประสิทธิภาพสูง และสามารถใช้ในการคัดแยกประเภทของพลาสติกได้อย่างแม่นยำ

ในส่วนของการประเมินประสิทธิภาพของโมเดล YOLOv8 แต่ละรุ่นนั้น ผู้วิจัยได้ทำการประเมินด้วยค่าตัวชี้วัด เช่น จำนวนพารามิเตอร์ของโมเดล, FLOPs, Accuracy, Precision, Recall, F1 score, top-1 accuracy, top-5 accuracy ของโมเดล YOLOv8 แต่ละรุ่นและจากการประเมินด้วยผลการทดสอบจากการนำเข้ารูปภาพจะพบว่า

สามารถสรุปได้ว่า โมเดล YOLOv8n เป็นโมเดลที่ให้ ประสิทธิภาพดีที่สุดโดยรวม ถึงแม้ประสิทธิภาพในการทำนายผลด้วยการประเมินจากการทดสอบนำเข้ารูปภาพจะให้ผลลัพธ์ไม่ดีเท่า YOLOv8x ด้วยความต่างของค่าเฉลี่ยเพียง 0.05 และเหตุผลอื่นๆ เช่น

1. ให้ค่า Overall Accuracy, Precision, Recall และ F1-score สูงที่สุด

2. มี Top-1 และ Top-5 Accuracy อยู่ในระดับสูงและใกล้เคียงกับโมเดลขนาดใหญ่กว่า

3. ใช้ จำนวนพารามิเตอร์และ FLOPs ต่ำที่สุด จึงส่งผลให้มีความเหมาะสมต่อการนำไปใช้งานจริง โดยเฉพาะในอุปกรณ์ที่มีข้อจำกัดด้านทรัพยากร

ดังนั้น YOLOv8n ถือเป็นโมเดลที่สมดุลทั้งในแง่ ความแม่นยำ ประสิทธิภาพ ที่เหมาะสมที่สุดสำหรับงานวิจัยนี้

5.2 อภิปรายผลวิจัย

5.2.1 จากสรุปผลการวิจัยการพัฒนาและเปรียบเทียบประสิทธิภาพของโมเดล YOLOv8 ในการคัดแยกประเภทของพลาสติกสำหรับการรีไซเคิลด้วยการประมวลผลจากภาพถ่ายดิจิทัล

พบว่าโมเดล YOLOv8 แต่ละรุ่นมีความแตกต่างกันทั้งในด้าน ความแม่นยำ ประสิทธิภาพการประมวลผลและปัจจัยต่างๆ ที่ประกอบในรูปภาพที่นำมาทดสอบ ซึ่งสอดคล้องกับแนวคิดของโมเดล YOLOv8 ที่ออกแบบให้มีหลากหลายขนาด (nano, small, medium, large, extra-large) เพื่อรองรับเงื่อนไขการใช้งานในบริบทที่แตกต่างกัน

การทดสอบนี้ชี้ให้เห็นว่าโมเดลขนาดเล็กอย่าง YOLOv8n สามารถทำงานได้อย่างรวดเร็วและใช้ทรัพยากรน้อย เหมาะสมกับงานที่ต้องการการตรวจจับแบบเรียลไทม์หรือใช้บนอุปกรณ์ที่มีข้อจำกัดด้านพลังประมวลผล เช่น กล้องอัจฉริยะหรือระบบ edge computing โดยให้ค่าเฉลี่ยของความแม่นยำในการทำนายพลาสติกทุกประเภทรวมกันที่ 87%

ในส่วนของ โมเดล YOLOv8s, YOLOv8m, YOLOv8l ก็แสดงให้เห็นถึงความไม่เสถียรต่อการนำใช้งานในการคัดแยกพลาสติกเนื่องจากค่าเฉลี่ยของความแม่นยำในการทำนายผลของ YOLOv8 ทุกโมเดลจะอยู่ที่ 0.84 หรือ 84% ซึ่งทั้ง 3 โมเดลดังกล่าวให้ค่าความแม่นยำโดยรวมที่ต่ำกว่าค่าเฉลี่ยดังกล่าว ซึ่งสำหรับโมเดลที่มีขนาดกลางถึงใหญ่อย่าง YOLOv8m กับ YOLOv8l อาจเกิดจากภาวะการเรียนรู้มากเกินไป (Overfitting) โดยโมเดลเรียนรู้ข้อมูลฝึกได้ละเอียดจนเกินไป โดยสามารถจับรายละเอียดหรือสัญญาณรบกวน ของข้อมูลได้ด้วย ส่งผลให้โมเดลมีความแม่นยำสูงมากบนชุดข้อมูลฝึก แต่เมื่อทดสอบกับชุดข้อมูลใหม่ กลับมีประสิทธิภาพต่ำลงอย่างมาก โมเดลจึงขาดความสามารถในการนำความรู้ไปใช้กับข้อมูลที่ไม่เคยเห็นมาก่อน แต่สำหรับโมเดลขนาดเล็กอย่าง YOLOv8s นั้นอาจเกิดจากภาวะการเรียนรู้น้อยเกินไป (Underfitting) โมเดลไม่สามารถเรียนรู้โครงสร้างหรือรูปแบบของข้อมูลได้เพียงพอ ทำให้ผลลัพธ์การทำนายมีความแม่นยำต่ำ ซึ่งสะท้อนได้ว่าโมเดลไม่มีความซับซ้อนเพียงพอหรือได้รับการฝึกไม่เพียงพอจำนวนรอบการฝึก (Epochs) น้อยเกินไป ดังนั้นจึงสรุปได้ว่าโมเดล YOLOv8 ทั้ง 3 รุ่นไม่เหมาะสมสำหรับการนำไปใช้ในงานคัดแยกประเภทพลาสติกสำหรับการรีไซเคิล

ในทางกลับกัน โมเดล YOLOv8x แสดงให้เห็นถึงความสามารถในการคัดแยกพลาสติกได้แม่นยำกว่าโมเดล YOLOv8 ทุกรุ่นโดยเฉพาะในกรณีที่ต้องแยกประเภทพลาสติกที่มีลักษณะใกล้เคียงกันหรือมีสิ่งรบกวนอยู่ในภาพ อย่างไรก็ตามโมเดลดังกล่าวต้องการทรัพยากรในการประมวลผลสูงกว่าโมเดลรุ่นอื่นๆ ซึ่งอาจมีค่าใช้จ่ายที่สูงและไม่เหมาะกับโรงงานรีไซเคิลที่มีข้อจำกัดต้นทุนที่จำกัด

ผลการวิจัยนี้สะท้อนให้เห็นว่า การเลือกใช้โมเดล YOLOv8 ที่เหมาะสมขึ้นอยู่กับบริบทของการใช้งาน หากโรงงานรีไซเคิลต้องการประสิทธิภาพการคัดแยกในเชิงปริมาณโดยการคัดแยกจำนวนมากต่อวินาที โมเดลขนาดเล็กอย่าง YOLOv8n ย่อมตอบโจทย์ได้ดีกว่า แต่หากต้องการความ

แม่นยำสูงที่สุดเพื่อให้การคัดแยกพลาสติกมีความบริสุทธิ์มากที่สุดและสามารถนำกลับมาใช้เป็นวัตถุดิบคุณภาพสูง โมเดลขนาดใหญ่อย่าง YOLOv8x ก็จะมีความเหมาะสมมากกว่า

5.2.2 ประโยชน์และคุณค่าของงานวิจัยการพัฒนาและเปรียบเทียบประสิทธิภาพของโมเดล YOLOv8 ในการคัดแยกประเภทของพลาสติกสำหรับการรีไซเคิลด้วยการประมวลผลจากภาพถ่ายดิจิทัลสามารถนำไปประยุกต์ใช้งานจริงในอุตสาหกรรมรีไซเคิลพลาสติกได้ดังนี้

1. ภาคอุตสาหกรรม โรงงานรีไซเคิลพลาสติกสามารถนำเทคโนโลยีการคัดแยกประเภทพลาสติกแบบอัตโนมัติมาประยุกต์ใช้ในสายพานคัดแยกได้ ลดการพึ่งพาแรงงานคน เพิ่มความรวดเร็ว และได้ผลลัพธ์การคัดแยกที่มีประสิทธิภาพสูงขึ้นจากการคัดแยกด้วยมือ

2. ภาคธุรกิจ ผู้ประกอบการที่ทำธุรกิจเกี่ยวข้องกับวัตถุดิบพลาสติก เนื่องจากการรีไซเคิลพลาสติกที่มีประสิทธิภาพ จะช่วยให้ได้วัตถุดิบพลาสติกรีไซเคิลที่มีคุณภาพสูงและบริสุทธิ์มากขึ้น ซึ่งช่วยลดต้นทุนในการผลิตวัตถุดิบใหม่และสอดคล้องกับแนวทางเศรษฐกิจหมุนเวียน

3. ภาคประชาชน การคัดแยกที่แม่นยำช่วยลดการปนเปื้อนของขยะ ลดปัญหาการฝังกลบหรือการกำจัดที่ไม่เหมาะสม ลดการเกิดมลพิษทางน้ำและอากาศ

4. ภาครัฐ ข้อมูลที่ได้จากการประยุกต์ใช้ระบบคัดแยกประเภทพลาสติกสามารถนำไปใช้ในการจัดทำสถิติหรือนโยบายด้านการจัดการขยะและสิ่งแวดล้อมได้อย่างมีประสิทธิภาพมากยิ่งขึ้น

5.3 ปัญหาและอุปสรรคในการทำวิจัย

ในการทดสอบประสิทธิภาพสำหรับเปรียบเทียบโมเดล YOLOv8 แต่ละรุ่น ผลลัพธ์ที่ได้ในขั้นตอนการทำ Validate จะให้ผลลัพธ์ในรูปแบบของรูปภาพไฟล์ PNG ซึ่งทำให้ไม่สามารถนำไปประมวลต่อได้ ทางผู้วิจัยจึงจำเป็นที่จะต้องมีการศึกษาเพิ่มเติมในเรื่องของเทคนิคการรู้จำอักขระด้วยแสง(Optical Character Recognition : OCR) และหาเครื่องมือสำหรับเทคนิคดังกล่าวก็คือ EasyOCR ที่เป็นไลบรารี Python เข้ามาใช้งานเพื่อที่จะได้ผลลัพธ์สำหรับนำไปเปรียบเทียบประสิทธิภาพของโมเดล YOLOv8 แต่ละรุ่นได้ต่อไป

5.4 ข้อเสนอแนะงานวิจัย

จากการทดสอบประสิทธิภาพในการคัดแยกประเภทพลาสติก งานวิจัยนี้ให้ผลลัพธ์ที่ค่อนข้างดี แต่ก็ยังมีช่องทางให้พัฒนาต่อได้โดยแบ่งออกเป็น 2 ปัจจัยดังนี้

5.4.1 ปัจจัยภายในเกี่ยวกับระบบข้อเสนอแนะสิ่งที่ต้องเตรียมสำหรับผู้ที่จะทำงานวิจัยลักษณะเดียวกันในเชิงเทคนิค

1. การจัดเตรียมชุดข้อมูลสำหรับฝึนฝน โดยเพิ่มจำนวนรูปภาพพลาสติกให้มีความหลากหลายคุณลักษณะดังนี้ การเพิ่มภาพพลาสติกที่อยู่ในสภาวะแสงที่แตกต่างกันเพื่อให้โมเดลเรียนรู้พลาสติกในมุมมองต่างๆ การถ่ายภาพพลาสติกควรจะถ่ายเป็นรูปพลาสติกชนิดนั้น แบบโดดเด่นเพื่อให้โมเดลเรียนรู้คุณลักษณะข้อมูลอย่างมีประสิทธิภาพ หลีกเลี่ยงการถ่ายภาพพลาสติกหลากหลายชนิดในภาพเดียวกัน

2 การเปรียบเทียบประสิทธิภาพของโมเดล YOLOv8 กับ โมเดล YOLO รุ่นใหม่ๆ อย่างเช่น YOLOv11, YOLOv12

3. การพัฒนาเว็บไซต์ในการจัดทำ Dashboard สำหรับติดตามผลการตรวจจับ เช่น สถิติพลาสติกแต่ละประเภทที่ถูกคัดแยกโดยเชื่อมต่อระบบกับฐานข้อมูลเพื่อติดตามปริมาณพลาสติกที่เข้าสู่กระบวนการรีไซเคิลและอาจพัฒนาเป็น API เพื่อให้โรงงานอื่น ๆ หรือผู้ประกอบการสามารถนำไปใช้ในระบบที่มีอยู่แล้วได้

4. การจัดหากล้องอุตสาหกรรม (Industrial Camera) ที่มีความละเอียดสูง 1080p – 4K เพื่อจับรายละเอียดของพลาสติกที่อยู่ในสภาวะแสงที่แตกต่างกันเพื่อให้โมเดลเรียนรู้พลาสติกในมุมมองต่างๆ

5.4.2 ปัจจัยภายนอกสิ่งที่ต้องทำเพื่อนำงานวิจัยไปบูรณาการร่วมหรือใช้งานจริงในภาคธุรกิจมีดังนี้

อุตสาหกรรมหรือภาคส่วนที่จะต้องร่วมมือกัน โรงงานรีไซเคิลหากต้องการนำงานวิจัยไปใช้จริงเพื่อทดสอบต้นแบบในสภาพแวดล้อมจริง สิ่งที่สำคัญที่จะต้องคำนึงถึงคือ โครงสร้างพื้นฐานและเทคโนโลยีในโรงงาน โดยโรงงานต้องมี สายพานลำเลียง ที่เหมาะสมกับการติดตั้งกล้องตรวจจับและแขนกลคัดแยกและระบบไฟฟ้าและเครือข่าย ต้องรองรับการทำงานของอุปกรณ์ AI เช่น GPU Server และค่าใช้จ่ายด้านฮาร์ดแวร์และซอฟต์แวร์

ผู้พัฒนาเทคโนโลยีหรือสตาร์ทอัพด้าน AI ก็จะมีบทบาทอย่างมากสำหรับที่จะต้องทำงานร่วมกันกับโรงงานรีไซเคิล โดยจะต้องวิเคราะห์ความต้องการของโรงงานรีไซเคิลให้มีความ

ชัดเจนว่าโรงงานรีไซเคิลและผู้ประกอบการต้องการโซลูชัน AI เพื่อเพิ่มประสิทธิภาพการคัดแยกจริง และต้องได้รับความร่วมมือจากโรงงานรีไซเคิลในการเปิดให้เก็บข้อมูล Dataset พลาสติกในสภาพแวดล้อมจริงและจะต้องมีการต้องมีทำสัญญาข้อตกลงการแบ่งปันข้อมูลส่วนบุคคลระหว่างผู้พัฒนา AI และผู้ประกอบการเพื่อที่จะนำข้อมูลมาใช้แล้วจึงปรับแต่งโมเดล YOLOv8 ให้เหมาะสมกับ Dataset และเงื่อนไขการใช้งาน

ทั้งนี้หน่วยงานรัฐก็จะมีบทบาทสำคัญในการสร้างเงื่อนไขภายนอก ในการบูรณาการเทคโนโลยี YOLOv8 ในการคัดแยกพลาสติกเกิดขึ้นทั้งด้าน นโยบาย กฎหมาย งบประมาณ โครงสร้างพื้นฐานของประเทศ ความร่วมมือระหว่างประเทศ และการสร้างการยอมรับของสังคม หากปัจจัยเหล่านี้ได้รับการจัดการอย่างเหมาะสม จะช่วยเร่งให้โรงงานรีไซเคิลและผู้พัฒนาเทคโนโลยีสามารถประยุกต์ใช้งานได้อย่างแพร่หลายและยั่งยืน

สถาบันเทคโนโลยีไทย-ญี่ปุ่น THAI - NICHI INSTITUTE OF TECHNOLOGY TNI

บรรณานุกรม

บรรณานุกรม

[1] กรมควบคุมมลพิษ, *รายงานสถานการณ์สถานที่กำจัดขยะมูลฝอยชุมชนของประเทศไทย ปี พ.ศ. 2567*, กรุงเทพฯ: กระทรวงทรัพยากรธรรมชาติและสิ่งแวดล้อม, 2567.

[2] ศุภกร กรบุญตรีทศ, *"แนวโน้มธุรกิจ/อุตสาหกรรม ปี 2567 – 2569: ธุรกิจร้านอาหารและเครื่องดื่ม,"* [Online]. Available: https://www.krungsri.com/th/research/industry/industry-outlook/services/food-beverages/io/io-food-beverage-restaurant-2024-2026. [Accessed: 24 สิงหาคม 2568].

[3] S. K. M. Huno et al., *Manual on Plastics Leakage Prevention from Formal and Informal Recycling Facilities in Nonthaburi*, Pathum Thani: Asian Institute of Technology, 2024.

[4] กรมควบคุมมลพิษ, *คู่มือการจัดการขยะพลาสติกเพื่อการรีไซเคิลอย่างยั่งยืน,* กรุงเทพฯ: กระทรวงทรัพยากรธรรมชาติและสิ่งแวดล้อม, 2565.

[5] P. Sambyal et al., "Plastic recycling: Challenges and opportunities," *Journal of Chemical Engineering,* vol. 103, no. 6, pp. 2462–2498, November 2024.

[6] C. A. Correa et al., "Challenges to reducing post-consumer plastic rejects from the MSW selective collection at two MRFs in São Paulo city, Brazil," *Journal of Material Cycles and Waste Management,* vol. 24, no. 3, pp. 1140-1155, March 2022.

[7] X. Wu et al., "Auto-sorting commonly recovered plastics from waste household appliances and electronics using near-infrared spectroscopy," *Journal of Cleaner Production,* vol. 246, pp. 118732, February 2020.

[8] T. N. Taneepanichskul et al., "A review of sorting and separating technologies suitable for compostable and biodegradable plastic packaging," *Frontiers in Sustainability,* vol. 3, Art. no. 901885, May 2022.

[9] C. Lubongo and P. Alexandridis, "Assessment of performance and challenges in use of commercial automated sorting technology for plastic waste," *Recycling,* vol. 7, no. 2, Art. no. 11, February 2022.

[10] A. R. Pathak et al., "Application of deep learning for object detection," *Procedia Computer Science,* vol. 132, pp. 1706–1717, May 2018.

[11] ภัทรมน หุ่นลำพูน, *“การพัฒนาและประยุกต์ใช้เทคนิค Convolutional Neural Network ในการจำแนกขยะชายฝั่งทะเล และการติดตามการแพร่กระจายของชนิดของขยะพลาสติกในทะเลบริเวณชายฝั่งทะเลอ่าวไทยตะวันออก,”* วิทยานิพนธ์ วท.ม. (เทคโนโลยีทางทะเล), มหาวิทยาลัยบูรพา, 2566.

[12] J. Redmon et al., “You only look once: Unified, real-time object detection,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* Las Vegas, NV, USA, June 27-30, 2016, pp. 779–788.

[13] M. A. R. Alif et al., *“YOLOv1 to YOLOv10: A Comprehensive Review of YOLO Variants and Their Application in the Agricultural Domain,”* arXiv 2406.10139 [Online]. Available: https://arxiv.org/abs/2406.10139, [Accessed: August 24, 2025].

[14] N. Wiangkam and S. Jiriwibhakorn, “Comparison of YOLOv8 models for aircraft detection in airport apron using digital image processing,” *Engineering and Technology Horizons,* vol. 41, no. 3, pp. 410309, September 2024.

[15] T. Sakda and T. Suthisarn, “Conditional classification of faded crosswalk detection using YOLOv8,” *Journal of Intelligent Transportation Systems,* vol. 27, no. 4, pp. 512–526, July 2023.

[16] ธานิล ม่วงพูล, “การพัฒนาระบบคัดแยกขยะรีไซเคิลด้วยเทคโนโลยีไอโอที,” *วารสารวิจัยและพัฒนา มหาวิทยาลัยราชภัฏนครราชสีมา*, เล่มที่ 12, ฉบับที่ 2, หน้า 45–52, พฤษภาคม 2563.

[17] บพิตร ไชยนอก และ ฤชานนท์ ศรีราวงค์, “การประยุกต์ใช้ Teachable Machine สำหรับระบบคัดแยกขวดน้ำอัตโนมัติ,” *วารสารเทคโนโลยีและวิศวกรรมก้าวหน้า,* เล่มที่ 2, ฉบับที่ 1–2, หน้า 13–20, มกราคม–กุมภาพันธ์ 2567.

[18] วดีนาถ วรรณสวัสดิ์กุล, “การพัฒนาเว็บแอปพลิเคชันแยกประเภทขยะด้วยเทคโนโลยีปัญญาประดิษฐ์,” *วารสารวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏเทพสตรี,* เล่มที่ 3, ฉบับที่ 1, หน้า 45–52, มกราคม 2567.

[19] H. Bichri et al., “Investigating the impact of train/test split ratio on the performance of pre-trained models with custom datasets,” *International Journal of Advanced Computer Science and Applications,* vol. 15, no. 2, pp. 333–340, February 2024.

[20] P. Gatchalee, *“Confusion Matrix เครื่องมือสำคัญในการประเมินผลลัพธ์ของการทำนายใน Machine Learning,”* Medium [Online]. Available: https://medium.com/@pagongatchalee/confusion-matrix-เครื่องมือสำคัญในการประเมินผลลัพธ์ของการทำนาย-ในmachine-learning-fba6e3f9508c. [Accessed: 24 สิงหาคม 2568].

[21] B. K. Pattanayak et al., *"A Novel Technique for Handwritten Text Recognition Using Easy OCR,"* ResearchGate, 2023 [Online]. Available: https://www.researchgate.net/publication/376355415_A_Novel_Technique_for_Handwritten_Text_Recognition_using_Easy_OCR. [Accessed: August 24, 2025].

สถาบันเทคโนโลยีไทย-ญี่ปุ่น
THAI - NICHI INSTITUTE OF TECHNOLOGY
TNI

ภาคผนวก

ภาคผนวก ก.
โค้ดระบบคัดแยกประเภทของพลาสติกฯ

TNI

โค้ดระบบคัดแยกฯ แบบจำลอง YOLOv8n

```
# Make sure that we have access to GPU.
!nvidia-smi
import os
HOME = os.getcwd()
print(HOME)

# Install YOLOv8
!pip install ultralytics
from IPython import display
display.clear_output()
import ultralytics
ultralytics.checks()
from ultralytics import YOLO
from IPython.display import display, Image

# Preparing a custom dataset
!mkdir '{HOME}/datasets'
DATA_DIR='/content/datasets'
from google.colab import drive
drive.mount('/content/drive/', force_remount=True)
!cp /dataset/.zip "/content/drive/MyDrive/datasets"
!unzip "/content/drive/MyDrive/datasets/dataset.zip" -d "datasets

# Custom Training YOLOv8 Model N
%cd {HOME}
!yolo task=classify mode=train model=yolov8n-cls.pt data='{DATA_DIR}'
epochs=5 imgsz=128
```

```
# Validate Custom Model
from ultralytics import YOLO
model_path = '/content/drive/MyDrive/TNIIS/ModelN/bestModelN.pt'
model = YOLO(model_path)
%cd {HOME}
!yolo task=detect mode=val
model='/content/drive/MyDrive/TNIIS/ModelN/bestModelN.pt'
data='{DATA_DIR}'
from sklearn.metrics import confusion_matrix

#Confusion Matrix
Image(filename=f'{HOME}/runs/classify/val/confusion_matrix.png')
Image(filename=f'{HOME}/runs/classify/val/confusion_matrix_normalized.p
ng')
Image(filename=f'{HOME}/runs/classify/train/results.png')

#Predict Image
%cd {HOME}
!yolo task=classify mode=predict
model="/content/drive/MyDrive/TNIIS/ModelN/bestModelN.pt" conf=0.25
source="/content/plastic11.jpg"

#OCR
!apt install tesseract-ocr
!apt install libtesseract-dev
!pip install pytesseract
!pip install easyocr
```

```
import cv2
import numpy as np
import easyocr
import matplotlib.pyplot as plt
image_path = "/content/runs/classify/val/confusion_matrix.png"
img = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
_, img_bin = cv2.threshold(img, 128, 255, cv2.THRESH_BINARY_INV +
cv2.THRESH_OTSU)
plt.imshow(img_bin, cmap='gray')
plt.show()
reader = easyocr.Reader(['en'])
results = reader.readtext(image_path, detail=0)
numbers = [int(num) for num in results if num.isdigit()]
print(f"Extracted Numbers: {numbers}")
matrix_size = int(np.sqrt(len(numbers)))
if len(numbers) > matrix_size * matrix_size:
    numbers = numbers[:matrix_size * matrix_size]
elif len(numbers) < matrix_size * matrix_size:
    print(f" ⚠️ Warning: Found only {len(numbers)} numbers, expected
{matrix_size * matrix_size}.")
    print("Please check the OCR results manually.")
conf_matrix = np.array(numbers).reshape(matrix_size, matrix_size)
print(conf_matrix)
```

```
#Accuracy Recall Precision F1 Score
import numpy as np

TP = np.diag(conf_matrix)
FP = np.sum(conf_matrix, axis=0) - TP
FN = np.sum(conf_matrix, axis=1) - TP
TN = np.sum(conf_matrix) - (FP + FN + TP)

# Precision recall F1score and accuracy
precision = TP / (TP + FP)
recall = TP / (TP + FN)
F1 = 2 * TP / (2 * TP + FP + FN)
accuracy = (TP + TN) / (TP + TN + FP + FN)

# For overall metrics, you might average or sum these depending on the
context
precision_overall = np.mean(precision)
recall_overall = np.mean(recall)
accuracy_overall = np.mean(accuracy)
F1_overall = np.mean(F1)

print(f"Precision per class: {precision}")
print(f"Recall per class: {recall}")
print(f"F1 per class: {F1}")
print(f"Accuracy per class: {accuracy}")
print(f"Overall Precision: {precision_overall}")
print(f"Overall Recall: {recall_overall}")
print(f"Overall Accuracy: {accuracy_overall}")
print(f"Overall F1: {F1_overall}")
```

โค้ดระบบคัดแยกฯ แบบจำลอง YOLOv8s

```
# Make sure that we have access to GPU.
!nvidia-smi
import os
HOME = os.getcwd()
print(HOME)

# Install YOLOv8
!pip install ultralytics
from IPython import display
display.clear_output()
import ultralytics
ultralytics.checks()
from ultralytics import YOLO
from IPython.display import display, Image

# Preparing a custom dataset
!mkdir '{HOME}/datasets'
DATA_DIR='/content/datasets'
from google.colab import drive
drive.mount('/content/drive/', force_remount=True)
!cp /dataset/.zip "/content/drive/MyDrive/datasets"
!unzip "/content/drive/MyDrive/datasets/dataset.zip" -d "datasets

# Custom Training YOLOv8 Model S
%cd {HOME}
!yolo task=classify mode=train model=yolov8s-cls.pt data='{DATA_DIR}'
epochs=5 imgsz=128
```

```
# Validate Custom Model
from ultralytics import YOLO
model_path = '/content/drive/MyDrive/TNIIS/ModelS/bestModelS.pt'
model = YOLO(model_path)
%cd {HOME}
!yolo task=detect mode=val
model='/content/drive/MyDrive/TNIIS/ModelS/bestModelS.pt'
data='{DATA_DIR}'
from sklearn.metrics import confusion_matrix

#Confusion Matrix
Image(filename=f'{HOME}/runs/classify/val/confusion_matrix.png')
Image(filename=f'{HOME}/runs/classify/val/confusion_matrix_normalized.p
ng')
Image(filename=f'{HOME}/runs/classify/train/results.png')

#Predict Image
%cd {HOME}
!yolo task=classify mode=predict
model="/content/drive/MyDrive/TNIIS/ModelS/bestModelS.pt" conf=0.25
source="/content/plastic11.jpg"

#OCR
!apt install tesseract-ocr
!apt install libtesseract-dev
!pip install pytesseract
!pip install easyocr
```

```
import cv2
import numpy as np
import easyocr
import matplotlib.pyplot as plt
image_path = "/content/runs/classify/val/confusion_matrix.png"
img = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
_, img_bin = cv2.threshold(img, 128, 255, cv2.THRESH_BINARY_INV +
cv2.THRESH_OTSU)
plt.imshow(img_bin, cmap='gray')
plt.show()
reader = easyocr.Reader(['en'])
results = reader.readtext(image_path, detail=0)
numbers = [int(num) for num in results if num.isdigit()]
print(f"Extracted Numbers: {numbers}")
matrix_size = int(np.sqrt(len(numbers)))
if len(numbers) > matrix_size * matrix_size:
    numbers = numbers[:matrix_size * matrix_size]
elif len(numbers) < matrix_size * matrix_size:
    print(f" ⚠️ Warning: Found only {len(numbers)} numbers, expected
{matrix_size * matrix_size}.")
    print("Please check the OCR results manually.")
conf_matrix = np.array(numbers).reshape(matrix_size, matrix_size)
print(conf_matrix)
```

```
#Accuracy Recall Precision F1 Score
import numpy as np

TP = np.diag(conf_matrix)
FP = np.sum(conf_matrix, axis=0) - TP
FN = np.sum(conf_matrix, axis=1) - TP
TN = np.sum(conf_matrix) - (FP + FN + TP)

# Precision recall F1score and accuracy
precision = TP / (TP + FP)
recall = TP / (TP + FN)
F1 = 2 * TP / (2 * TP + FP + FN)
accuracy = (TP + TN) / (TP + TN + FP + FN)

# For overall metrics, you might average or sum these depending on the
context
precision_overall = np.mean(precision)
recall_overall = np.mean(recall)
accuracy_overall = np.mean(accuracy)
F1_overall = np.mean(F1)

print(f"Precision per class: {precision}")
print(f"Recall per class: {recall}")
print(f"F1 per class: {F1}")
print(f"Accuracy per class: {accuracy}")
print(f"Overall Precision: {precision_overall}")
print(f"Overall Recall: {recall_overall}")
print(f"Overall Accuracy: {accuracy_overall}")
print(f"Overall F1: {F1_overall}")
```

โค้ดระบบคัดแยกฯ แบบจำลอง YOLOv8m

```
# Make sure that we have access to GPU.
!nvidia-smi
import os
HOME = os.getcwd()
print(HOME)

# Install YOLOv8
!pip install ultralytics
from IPython import display
display.clear_output()
import ultralytics
ultralytics.checks()
from ultralytics import YOLO
from IPython.display import display, Image

# Preparing a custom dataset
!mkdir '{HOME}/datasets'
DATA_DIR='/content/datasets'
from google.colab import drive
drive.mount('/content/drive/', force_remount=True)
!cp /dataset/.zip "/content/drive/MyDrive/datasets"
!unzip "/content/drive/MyDrive/datasets/dataset.zip" -d "datasets

# Custom Training YOLOv8 Model M
%cd {HOME}
!yolo task=classify mode=train model=yolov8m-cls.pt data='{DATA_DIR}'
epochs=5 imgsz=128
```

```
# Validate Custom Model
from ultralytics import YOLO
model_path = '/content/drive/MyDrive/TNIIS/ModelM/bestModelM.pt'
model = YOLO(model_path)
%cd {HOME}
!yolo task=detect mode=val
model='/content/drive/MyDrive/TNIIS/ModelM/bestModelM.pt'
data='{DATA_DIR}'
from sklearn.metrics import confusion_matrix

#Confusion Matrix
Image(filename=f'{HOME}/runs/classify/val/confusion_matrix.png')
Image(filename=f'{HOME}/runs/classify/val/confusion_matrix_normalized.p
ng')
Image(filename=f'{HOME}/runs/classify/train/results.png')

#Predict Image
%cd {HOME}
!yolo task=classify mode=predict
model="/content/drive/MyDrive/TNIIS/ModelM/bestModelM.pt" conf=0.25
source="/content/plastic11.jpg"

#OCR
!apt install tesseract-ocr
!apt install libtesseract-dev
!pip install pytesseract
!pip install easyocr
```

```
import cv2
import numpy as np
import easyocr
import matplotlib.pyplot as plt
image_path = "/content/runs/classify/val/confusion_matrix.png"
img = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
_, img_bin = cv2.threshold(img, 128, 255, cv2.THRESH_BINARY_INV +
cv2.THRESH_OTSU)
plt.imshow(img_bin, cmap='gray')
plt.show()
reader = easyocr.Reader(['en'])
results = reader.readtext(image_path, detail=0)
numbers = [int(num) for num in results if num.isdigit()]
print(f"Extracted Numbers: {numbers}")
matrix_size = int(np.sqrt(len(numbers)))
if len(numbers) > matrix_size * matrix_size:
    numbers = numbers[:matrix_size * matrix_size]
elif len(numbers) < matrix_size * matrix_size:
    print(f" ⚠️ Warning: Found only {len(numbers)} numbers, expected
{matrix_size * matrix_size}.")
    print("Please check the OCR results manually.")
conf_matrix = np.array(numbers).reshape(matrix_size, matrix_size)
print(conf_matrix)
```

```
#Accuracy Recall Precision F1 Score
import numpy as np

TP = np.diag(conf_matrix)
FP = np.sum(conf_matrix, axis=0) - TP
FN = np.sum(conf_matrix, axis=1) - TP
TN = np.sum(conf_matrix) - (FP + FN + TP)

# Precision recall F1score and accuracy
precision = TP / (TP + FP)
recall = TP / (TP + FN)
F1 = 2 * TP / (2 * TP + FP + FN)
accuracy = (TP + TN) / (TP + TN + FP + FN)

# For overall metrics, you might average or sum these depending on the
context
precision_overall = np.mean(precision)
recall_overall = np.mean(recall)
accuracy_overall = np.mean(accuracy)
F1_overall = np.mean(F1)

print(f"Precision per class: {precision}")
print(f"Recall per class: {recall}")
print(f"F1 per class: {F1}")
print(f"Accuracy per class: {accuracy}")
print(f"Overall Precision: {precision_overall}")
print(f"Overall Recall: {recall_overall}")
print(f"Overall Accuracy: {accuracy_overall}")
print(f"Overall F1: {F1_overall}")
```

โค้ดระบบคัดแยกฯ แบบจำลอง YOLOv8l

```
# Make sure that we have access to GPU.
!nvidia-smi
import os
HOME = os.getcwd()
print(HOME)

# Install YOLOv8
!pip install ultralytics
from IPython import display
display.clear_output()
import ultralytics
ultralytics.checks()
from ultralytics import YOLO
from IPython.display import display, Image

# Preparing a custom dataset
!mkdir '{HOME}/datasets'
DATA_DIR='/content/datasets'
from google.colab import drive
drive.mount('/content/drive/', force_remount=True)
!cp /dataset/.zip "/content/drive/MyDrive/datasets"
!unzip "/content/drive/MyDrive/datasets/dataset.zip" -d "datasets

# Custom Training YOLOv8 Model L
%cd {HOME}
!yolo task=classify mode=train model=yolov8l-cls.pt data='{DATA_DIR}'
epochs=5 imgsz=128
```

```
# Validate Custom Model
from ultralytics import YOLO
model_path = '/content/drive/MyDrive/TNIIS/ModelL/bestModelL.pt'
model = YOLO(model_path)
%cd {HOME}
!yolo task=detect mode=val
model='/content/drive/MyDrive/TNIIS/ModelL/bestModelL.pt'
data='{DATA_DIR}'
from sklearn.metrics import confusion_matrix

#Confusion Matrix
Image(filename=f'{HOME}/runs/classify/val/confusion_matrix.png')
Image(filename=f'{HOME}/runs/classify/val/confusion_matrix_normalized.p
ng')
Image(filename=f'{HOME}/runs/classify/train/results.png')

#Predict Image
%cd {HOME}
!yolo task=classify mode=predict
model="/content/drive/MyDrive/TNIIS/ModelL/bestModelL.pt" conf=0.25
source="/content/plastic11.jpg"

#OCR
!apt install tesseract-ocr
!apt install libtesseract-dev
!pip install pytesseract
!pip install easyocr
```

```
import cv2
import numpy as np
import easyocr
import matplotlib.pyplot as plt
image_path = "/content/runs/classify/val/confusion_matrix.png"
img = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
_, img_bin = cv2.threshold(img, 128, 255, cv2.THRESH_BINARY_INV +
cv2.THRESH_OTSU)
plt.imshow(img_bin, cmap='gray')
plt.show()
reader = easyocr.Reader(['en'])
results = reader.readtext(image_path, detail=0)
numbers = [int(num) for num in results if num.isdigit()]
print(f"Extracted Numbers: {numbers}")
matrix_size = int(np.sqrt(len(numbers)))
if len(numbers) > matrix_size * matrix_size:
    numbers = numbers[:matrix_size * matrix_size]
elif len(numbers) < matrix_size * matrix_size:
    print(f" ⚠️ Warning: Found only {len(numbers)} numbers, expected
{matrix_size * matrix_size}.")
    print("Please check the OCR results manually.")
conf_matrix = np.array(numbers).reshape(matrix_size, matrix_size)
print(conf_matrix)
```

```
#Accuracy Recall Precision F1 Score
import numpy as np

TP = np.diag(conf_matrix)
FP = np.sum(conf_matrix, axis=0) - TP
FN = np.sum(conf_matrix, axis=1) - TP
TN = np.sum(conf_matrix) - (FP + FN + TP)

# Precision recall F1score and accuracy
precision = TP / (TP + FP)
recall = TP / (TP + FN)
F1 = 2 * TP / (2 * TP + FP + FN)
accuracy = (TP + TN) / (TP + TN + FP + FN)

# For overall metrics, you might average or sum these depending on the
context
precision_overall = np.mean(precision)
recall_overall = np.mean(recall)
accuracy_overall = np.mean(accuracy)
F1_overall = np.mean(F1)

print(f"Precision per class: {precision}")
print(f"Recall per class: {recall}")
print(f"F1 per class: {F1}")
print(f"Accuracy per class: {accuracy}")
print(f"Overall Precision: {precision_overall}")
print(f"Overall Recall: {recall_overall}")
print(f"Overall Accuracy: {accuracy_overall}")
print(f"Overall F1: {F1_overall}")
```

โค้ดระบบคัดแยกฯ แบบจำลอง YOLOv8X

```
# Make sure that we have access to GPU.
!nvidia-smi
import os
HOME = os.getcwd()
print(HOME)

# Install YOLOv8
!pip install ultralytics
from IPython import display
display.clear_output()
import ultralytics
ultralytics.checks()
from ultralytics import YOLO
from IPython.display import display, Image

# Preparing a custom dataset
!mkdir '{HOME}/datasets'
DATA_DIR='/content/datasets'
from google.colab import drive
drive.mount('/content/drive/', force_remount=True)
!cp /dataset/.zip "/content/drive/MyDrive/datasets"
!unzip "/content/drive/MyDrive/datasets/dataset.zip" -d "datasets

# Custom Training YOLOv8 Model X
%cd {HOME}
!yolo task=classify mode=train model=yolov8x-cls.pt data='{DATA_DIR}'
epochs=5 imgsz=128
```

```
# Validate Custom Model
from ultralytics import YOLO
model_path = '/content/drive/MyDrive/TNIIS/ModelX/bestModelX.pt'
model = YOLO(model_path)
%cd {HOME}
!yolo task=detect mode=val
model='/content/drive/MyDrive/TNIIS/ModelX/bestModelX.pt'
data='{DATA_DIR}'
from sklearn.metrics import confusion_matrix

#Confusion Matrix
Image(filename=f'{HOME}/runs/classify/val/confusion_matrix.png')
Image(filename=f'{HOME}/runs/classify/val/confusion_matrix_normalized.p
ng')
Image(filename=f'{HOME}/runs/classify/train/results.png')

#Predict Image
%cd {HOME}
!yolo task=classify mode=predict
model="/content/drive/MyDrive/TNIIS/ModelX/bestModelX.pt" conf=0.25
source="/content/plastic11.jpg"

#OCR
!apt install tesseract-ocr
!apt install libtesseract-dev
!pip install pytesseract
!pip install easyocr
```

```
import cv2
import numpy as np
import easyocr
import matplotlib.pyplot as plt
image_path = "/content/runs/classify/val/confusion_matrix.png"
img = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
_, img_bin = cv2.threshold(img, 128, 255, cv2.THRESH_BINARY_INV +
cv2.THRESH_OTSU)
plt.imshow(img_bin, cmap='gray')
plt.show()
reader = easyocr.Reader(['en'])
results = reader.readtext(image_path, detail=0)
numbers = [int(num) for num in results if num.isdigit()]
print(f"Extracted Numbers: {numbers}")
matrix_size = int(np.sqrt(len(numbers)))
if len(numbers) > matrix_size * matrix_size:
    numbers = numbers[:matrix_size * matrix_size]
elif len(numbers) < matrix_size * matrix_size:
    print(f"⚠️ Warning: Found only {len(numbers)} numbers, expected
{matrix_size * matrix_size}.")
    print("Please check the OCR results manually.")
conf_matrix = np.array(numbers).reshape(matrix_size, matrix_size)
print(conf_matrix)
```

```
#Accuracy Recall Precision F1 Score
import numpy as np

TP = np.diag(conf_matrix)
FP = np.sum(conf_matrix, axis=0) - TP
FN = np.sum(conf_matrix, axis=1) - TP
TN = np.sum(conf_matrix) - (FP + FN + TP)

# Precision recall F1score and accuracy
precision = TP / (TP + FP)
recall = TP / (TP + FN)
F1 = 2 * TP / (2 * TP + FP + FN)
accuracy = (TP + TN) / (TP + TN + FP + FN)

# For overall metrics, you might average or sum these depending on the
context
precision_overall = np.mean(precision)
recall_overall = np.mean(recall)
accuracy_overall = np.mean(accuracy)
F1_overall = np.mean(F1)

print(f"Precision per class: {precision}")
print(f"Recall per class: {recall}")
print(f"F1 per class: {F1}")
print(f"Accuracy per class: {accuracy}")
print(f"Overall Precision: {precision_overall}")
print(f"Overall Recall: {recall_overall}")
print(f"Overall Accuracy: {accuracy_overall}")
print(f"Overall F1: {F1_overall}")
```