



ระบบตรวจจับและระบุประเภทไม้ประดับบอนสีด้วยการประมวลผลภาพ
Caladium ornamental plant species detection and identification system
with image processing

ณัฐพร รอดสไต

นลพรรณ อมรโชติพันธ์

เพชรชริน สุขสม

โครงการศึกษาหัวข้อพิเศษนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

วิทยาศาสตร์บัณฑิต สาขาวิชาระบบสารสนเทศ

หลักสูตรการวิเคราะห์ข้อมูลเชิงธุรกิจ

มหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร

ปีการศึกษา 2565

ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร



ระบบตรวจจับและระบุประเภทไม้ประดับบอนสีด้วยการประมวลผลภาพ
Caladium ornamental plant species detection and identification system
with image processing

ณัฐพร รอดสไต

นลพรรณ อมรโชติพันธ์

เพชรชริน สุขสม

อาจารย์ที่ปรึกษาโครงการ

อาจารย์ ดร. สุจิตรา ไชยกุลสินธุ์

อาจารย์เกียรติศักดิ์ ลาภพาณิชยกุล

อาจารย์ดารุณี บุญมา

โครงการศึกษาหัวข้อพิเศษนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

วิทยาศาสตรบัณฑิต สาขาวิชาระบบสารสนเทศ

หลักสูตรการวิเคราะห์ข้อมูลเชิงธุรกิจ

มหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร

ปีการศึกษา 2565

ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร

- ชื่อโครงการ** : ระบบตรวจจับและระบุประเภทไม้ประดับบอนสีด้วยการประมวลผลภาพ
- ชื่อผู้ดำเนินโครงการ** : นางสาวณัฐพร รอดสดีใส
นางสาวนลพรรณ อมรโชติพันธ์
นางสาวเพชรชริน สุขสม
- สาขาวิชา** : ระบบสารสนเทศ คณะบริหารธุรกิจ
มหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร
- ที่ปรึกษาโครงการ** : อาจารย์ ดร. สุจิตรา ไชยกุลสินธุ์ ประธานที่ปรึกษาโครงการ
อาจารย์เกียรติศักดิ์ ลาภพาณิชย์กุล รองประธานที่ปรึกษาโครงการ
อาจารย์ดารุณี บุญมา
- ปีการศึกษา** : 2565

บทคัดย่อ

บอนสีที่เป็นไม้ประดับที่ถือว่าเป็นราชินีแห่งไม้ใบ และได้รับความนิยมในการปลูกมาตั้งแต่อดีตจนถึงปัจจุบัน ด้วยความงดงามของรูปแบบใบ และสีเส้นที่หลากหลายนวมทั้งการแต่งแต้มสีเส้นของแต่ละใบที่สวยงามแตกต่างกัน บอนสีมีหลากหลายสายพันธุ์ และมีลักษณะที่คล้ายคลึงกันจึงอาจเกิดความสับสนในการจำแนกประเภทของบอนสี งานวิจัยนี้ได้ใช้เทคนิคการประมวลผลภาพร่วมกับการใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันจำแนกประเภทของบอนสี โดยใช้ข้อมูลของบอนสี 4 ประเภท ได้แก่ ใบไทย ใบยาว ใบกลม และใบแผ่ ซึ่งข้อมูลภาพทั้งหมด 1,500 ภาพ แบ่งเป็นข้อมูลสอน (Training set) ประเภทละ 300 ภาพ เป็นทั้งหมดจำนวน 1,200 ภาพ และข้อมูลทดสอบ (Test set) ประเภทละ 75 ภาพ เป็นทั้งหมดจำนวน 300 ภาพ เพื่อลดทรัพยากร และเวลาในการประมวลผล ผู้วิจัยได้ใช้เทคนิคการถ่ายโอนการเรียนรู้จาก 4 โมเดล คือ MobilenetV2 InceptionV3 VGG16 และ ResNet50 จากผลการทดสอบพบว่าทั้งสี่โมเดลสามารถจำแนกประเภทของบอนสีได้มากกว่าร้อยละ 80 โดยโมเดลที่มีการจำแนกประเภทของบอนสีได้แม่นยำมากที่สุดคือ Resnet50V2 มีความถูกต้องร้อยละ 88.67 รองลงมาคือ MobilenetV2 มีความถูกต้องร้อยละ 87.33 ส่วน VGG16 และ InceptionV3 มีความถูกต้องอยู่ที่ร้อยละ 83.67 และ 82 ตามลำดับ

คำสำคัญ : ระบบตรวจจับ การประมวลผลภาพและโครงข่ายประสาทเทียมแบบคอนโวลูชัน บอนสี

Project Title : Caladium ornamental plant species detection and
identification system with image processing

Student's Name : Ms. Nathaporn Rodsodsai
Ms. Nalaphan Amornthotiphan
Ms. Petcharin Suksom

Degree an Program : Bachelor of Science (Business Data Analytics)

Advisory Committee : Dr. Suchira Chigusin
Mr. Kreadtisak Lappanitchayakul
Ms. Darunee Bunma

Academic Year : 2022

Abstract

Caladium is an ornamental plant with a beautiful variety of pattern and color of the leaves. Including the appearance of tinging the color of each leaf that is beautiful and different, Caladium is considered the queen of foliage plants that have been popular in planting from the past to the present. There are many species or types of Caladium that look similar, therefore, there may be confusion in classification of the types of Caladium. In this research, an image processing technique was combined with a convolutional neural network to classify the types of Caladium. By using the data of 4 types of Caladium, namely, Thai leaf, long leaf, round leaf and bamboo leaf, which has a total of 1,500 images, the data divided into 300 images of each type of a training set, totaling 1,200 images, and the rest 75 images of each type for a total of 300 images are used as a test set. In order to reduce resources and processing time, the researchers used Transfer Learning techniques from four convolutional neural network models, MobilenetV2, InceptionV3, VGG16 and ResNet50. The results showed that all four models were able to classify more than 80 percent of the Caladium classification. The most accurate was Resnet50V2 with 88.67 percent accuracy, followed by MobilenetV2 with 87.33 percent accuracy, and VGG16 and InceptionV3 with 83.67 and 82 percent accuracy, respectively.

Keyword: Caladium classification, convolutional neural networks, image processing

กิตติกรรมประกาศ

งานวิจัยนี้สำเร็จได้ด้วยการสนับสนุน และความช่วยเหลือจากหลายส่วน ขอบพระคุณที่ปรึกษา อาจารย์ ดร. สุจิรา ไชยกุลสินธุ์ อาจารย์เกียรติศักดิ์ ลาภพาณิชย์กุล และอาจารย์ดารุณี บุญมา ที่ได้สละเวลามาให้คำปรึกษาคำแนะนำ และแนวทางสำหรับการทำวิจัยนี้ รวมทั้งเป็นผู้ประสานงานให้ความช่วยเหลือแก่นักศึกษาที่ทำวิจัยทุกคน

ขอบคุณเจ้าของสวนมาลัยทอง และเจ้าของสวนนพดลที่สละเวลา และให้ข้อมูลที่เป็นประโยชน์ และความร่วมมือในการเก็บรวบรวมข้อมูลในงานวิจัยนี้

สุดท้ายนี้ผู้วิจัยขอขอบคุณเพื่อน น้อง และสมาชิกผู้วิจัยทุกคนที่มีส่วนร่วมในการช่วยเหลือในทุก ๆ เรื่อง และคอยให้กำลังใจจนงานวิจัยนี้สำเร็จลุล่วงไปได้ด้วยดี

ณัฐพร รอดสดใส

นลพรรณ อมรโชติพันธ์

เพชรชริน สุขสม



สารบัญ

	หน้า
บทคัดย่อ.....	ก
กิตติกรรมประกาศ.....	ข
สารบัญ.....	ง
สารบัญตาราง.....	จ
สารบัญภาพ.....	ช
บทที่ 1 บทนำ	
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์.....	3
1.3 ขอบเขตของการศึกษา.....	3
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	4
1.5 นิยามศัพท์เฉพาะ.....	5
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	
2.1 ความรู้ทั่วไปเกี่ยวกับพันธุ์ไม้ประดับบอนสี.....	6
2.2 การประมวลผลภาพ (Image Processing).....	10
2.3 ภาษา Python.....	12
2.4 Computer Vision.....	13
2.5 มาตรฐานของสี.....	14
2.6 การตรวจจับวัตถุ (Object Detection).....	15
2.7 การจำแนกประเภทข้อมูลภาพ (Image Classification).....	15
2.8 การเรียนรู้ของเครื่อง (Machine Learning).....	16
2.9 ทฤษฎีการเรียนรู้เชิงลึก (Deep Learning).....	17
2.10 โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks: CNN).....	17
2.11 หลักการ Keras.....	23
2.12 ทฤษฎีการถ่ายโอนการเรียนรู้ (Transfer Learning).....	24
2.13 การวัดประสิทธิภาพ (K-Fold Cross Validation).....	26
2.14 InceptionV3.....	28

สารบัญ (ต่อ)

	หน้า
2.15 VGG16.....	29
2.16 MobileNetV2.....	31
2.17 ResNet.....	32
2.18 งานวิจัยที่เกี่ยวข้อง.....	33
บทที่ 3 วิธีดำเนินงานวิจัย	
3.1 แบบแผนการวิจัย.....	38
3.2 เครื่องมือที่ใช้ในการวิจัย.....	38
3.3 การเก็บรวบรวมข้อมูล.....	38
3.4 การเตรียมข้อมูล (Data Preparation and Image augmentation).....	40
3.5 สร้างแบบจำลอง และถ่ายโอนการเรียนรู้ (Create Model and Transfer Learning).....	47
บทที่ 4 ผลการดำเนินงาน	
4.1 ผลลัพธ์จากข้อมูลหลังทำการฝึก โดยไม่ใส่ K-Fold Cross Validation.....	50
4.2 วิเคราะห์ความสัมพันธ์จากข้อมูลหลังทำการฝึก โดยใส่ K-Fold Cross Validation.....	55
4.3 การเปรียบเทียบประสิทธิภาพของโมเดล (Model) เมื่อทำการทดสอบ (Test).....	65
4.4 การเปรียบเทียบความแม่นยำในการทำนายภาพของ Model เมื่อทำการ Test.....	66
บทที่ 5 สรุปผลและข้อเสนอแนะ	
5.1 สรุปผลการวิจัย.....	69
5.2 อภิปรายผล.....	72
5.3 ข้อเสนอแนะ.....	76
5.4 ข้อจำกัด.....	76
บรรณานุกรม	77
ภาคผนวก ก คู่มือโครงการ	80
ประวัติผู้ดำเนินโครงการ	105

สารบัญตาราง

ตารางที่		หน้า
3-1	ข้อมูลหมวดหมู่ (Class) ของประเภทบอนสีที่ทำการเก็บข้อมูล.....	39
4-1	การเปรียบเทียบประสิทธิภาพของโมเดล (Model).....	65
4-2	การเปรียบเทียบความแม่นยำในการทำนายภาพของโมเดล (Model).....	66



สารบัญภาพ

ภาพที่	หน้า
2-1	บอนใบไทย..... 8
2-2	บอนใบกลม..... 8
2-3	บอนใบยาว..... 9
2-4	บอนใบกาบ..... 9
2-5	บอนใบไข่..... 9
2-6	การแทนค่าพิกเซลของรูปสีขา-ดำ นั้นด้วยตัวเลขตั้งแต่ 0-255..... 19
2-7	จำลองเมทริกซ์ที่ได้จากชุดข้อมูลรับเข้า และเมทริกซ์ฟิลเตอร์ (filter)..... 19
2-8	หลักการทํางานของค่าฟิลเตอร์ (filter) และการกำเนิดเมทริกซ์ ชุดใหม่หรือพีเจอร์แมพ..... 20
2-9	โครงข่ายประสาทเทียมแบบคอนโวลูชัน..... 21
2-10	การพูลลิ่งแบบมากที่สุด (max pooling)..... 22
2-11	การเชื่อมต่อกันของแต่ละชั้นอย่างสมบูรณ์ (Full connected layer)..... 23
2-12	โมเดล (Model) ของ CNN ที่ได้ทำการฝึกฝนบน ImageNet และคุณลักษณะ แต่ละโมเดล (Model)..... 25
2-13	สถาปัตยกรรม (Architecture) ConvNet..... 26
2-14	ตัวอย่าง 5-Fold Cross-Validation..... 27
2-15	สถาปัตยกรรม (Architecture) InceptionV3..... 28
2-16	โครงสร้างโมเดล (Model) ของ InceptionV3..... 29
2-17	สถาปัตยกรรม (Architecture) VGG16..... 30
2-18	โครงสร้างโมเดล (Model) ของ VGG16..... 30
2-19	โครงสร้างโมเดล (Model) ของ Mobilenet..... 32
2-20	สถาปัตยกรรม (Architecture) Resnet..... 33
3-1	ตัวอย่าง Data Augmentation..... 40
3-2	ค้นหา Roboflow ผ่าน Google Chrome..... 41
3-3	เลือกเว็บแอปพลิเคชัน Roboflow..... 41
3-4	เลือก Sign in การใช้งานสามารถ Sign in ผ่าน Email..... 41
3-5	ทำการสร้าง Project ขึ้นมา..... 42

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
3-6	ทำการ Create Project..... 42
3-7	นำรูปภาพที่ต้องการ Augmentation มาใส่..... 43
3-8	ทำการเลือกรูปภาพ Unannotated..... 43
3-9	สร้าง Group ของภาพ..... 43
3-10	Overview จำนวนภาพที่ Annotated แล้ว..... 44
3-11	เลือก Add image to data set..... 44
3-12	กำหนดค่า Augmentation..... 45
3-13	การ Generate รูปภาพเพิ่ม..... 46
3-14	การ Generate ภาพเพิ่มให้เพียงพอต่อการ Train..... 46
3-15	การ Export เพื่อบันทึกภาพ..... 46
3-16	ตัวอย่าง 5-Fold Cross Validation..... 48
4-1	การคำนวณค่า Evaluate ของโมเดล (Model) Keres..... 50
4-2	Report และ Confusion Matrix ของโมเดล (Model) Keres..... 50
4-3	การคำนวณค่า Evaluate ของโมเดล (Model) Mobilenetv2..... 51
4-4	Report และ Confusion Matrix ของโมเดล (Model) Mobilenetv2..... 51
4-5	การคำนวณค่า Evaluate ของโมเดล (Model) InceptionV3..... 52
4-6	Report และ Confusion Matrix ของโมเดล (Model) InceptionV3..... 52
4-7	การคำนวณค่า Evaluate ของโมเดล (Model) VGG16..... 53
4-8	Report และ Confusion Matrix ของโมเดล (Model) VGG16..... 53
4-9	การคำนวณค่า Evaluate ของโมเดล (Model) Resnet50V2..... 54
4-10	Report และ Confusion Matrix ของโมเดล (Model) Resnet50V2..... 54
4-11	ค่า Average Scores ของทุก Fold ของโมเดล (Model) Keras..... 55
4-12	ค่า Evaluate ของการทดสอบ (Test) ของโมเดล (Model) Keras..... 55
4-13	Report และ Confusion Matrix เมื่อทำการทดสอบ (Test)..... 56
4-14	ค่า Average Scores ของทุก Fold ของโมเดล (Model) MobilenetV2..... 57
4-15	ค่า Evaluate ของการทดสอบ (Test) ของโมเดล (Model) MobilenetV2..... 57

สารบัญภาพ (ต่อ)

ภาพที่		หน้า
4-16	Report และ Confusion Matrix เมื่อทำการทดสอบ (Test) ของ โมเดล (Model) MobilenetV2.....	58
4-17	ค่า Average Scores ของทุก Fold ของโมเดล (Model) InceptionV3.....	59
4-18	ค่า Evaluate ของการทดสอบ (Test) ของโมเดล (Model) InceptionV3.....	59
4-19	Report และ Confusion Matrix เมื่อทำการทดสอบ (Test) ของ โมเดล (Model) InceptionV3.....	60
4-20	ค่า Average Scores ของทุก Fold ของโมเดล (Model) VGG16.....	61
4-21	ค่า Evaluate ของการทดสอบ (Test) ของโมเดล (Model) VGG16.....	61
4-22	Report และ Confusion Matrix เมื่อทำการทดสอบ (Test) ของ โมเดล (Model) VGG16.....	62
4-23	ค่า Average Scores ของทุก Fold ของโมเดล (Model) Resnet50V2.....	63
4-24	ค่า Evaluate ของการทดสอบ (Test) ของโมเดล (Model) Resnet50V2.....	63
4-25	Report และ Confusion Matrix เมื่อทำการทดสอบ (Test) ของ โมเดล (Model) Resnet50V2.....	64
5-1	บอนใบไทย.....	69
5-2	บอนใบกลม.....	69
5-3	บอนใบยาว.....	70
5-1	บอนใบไผ่.....	70
5-1	เปรียบเทียบชุดข้อมูลทดสอบ (Test) ระหว่าง ใบยาว (ด้านซ้าย) และใบไทย (ด้านขวา).....	74
5-2	เปรียบเทียบชุดข้อมูลทดสอบ (Test) ระหว่าง ใบยาว (ด้านซ้าย) และใบไผ่ (ด้านขวา).....	74
5-3	เปรียบเทียบชุดข้อมูลทดสอบ (Test) ระหว่าง ใบยาว (ด้านซ้าย) และใบไทย (ด้านขวา).....	75
5-4	เปรียบเทียบชุดข้อมูลทดสอบ (Test) ระหว่าง ใบยาว (ด้านซ้าย) และใบกลม (ด้านขวา).....	75

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในอดีตต้นกำเนิดของ “บอนสี” อยู่ในทวีปอเมริกาใต้ จากนั้นได้แพร่หลายได้เข้ามาทางทวีปยุโรป อินเดีย จนถึงประเทศอินโดนีเซีย โดยคาดว่าได้เข้ามาในประเทศตั้งแต่สมัยสุโขทัย ต่อมาเมื่อสมัยกรุงศรีอยุธยาได้มีหลักฐานพบว่า “สมเด็จพระนารายณ์มหาราช” ได้มีพืชต่างถิ่นที่ชาวต่างประเทศเป็นผู้นำทูลเกล้าฯ ถวายจากเมืองชวา เมืองจีน แหลมบาลายู และอินเดียตอนใต้ (อรรพรรณ วิชัยลักษณ์, 2548) จนถึงสมัยกรุงรัตนโกสินทร์ การปลูกเลี้ยงบอนสีก็ยิ่งมีความหลากหลายมากยิ่งขึ้น และเมื่อพระบาทสมเด็จพระจุลจอมเกล้าเจ้าอยู่หัว (รัชกาลที่ 5) เสด็จประพาสยุโรปครั้งแรกในปี พ.ศ. 2440 ได้ทรงนำบอนสีกลับเข้ามาปลูกในพระบรมมหาราชวัง สร้างความนิยมให้กับฝ่ายใน โดยยุคนั้นสายพันธุ์ที่มีชื่อเสียง คือ “บอนสีเจ้ากรุงโกเซอร์” และ “บอนสีเจ้ากรุงเดนมาร์ก” (กวี ลิมอัสโน) โดยมีการพัฒนาปรับปรุง และขยายพันธุ์บอนสีออกไปอย่างมาก โดยพันธุ์ใหม่ ๆ ได้มีการตั้งชื่อเป็นตัวละครในวรรณคดี จังหวัด บุคคลสำคัญในประวัติศาสตร์ ฯลฯ เป็นต้น ในความเป็นจริงประเทศไทยมีบอนสีพื้นบ้านซึ่งถือเป็นบอนสีโบราณ ตัวอย่างบอนสีโบราณ เช่น พันธุ์วัวแดง ช้างเผือกใบบัว พระเสวก โดยต่อมาก็มีพันธุ์กรมเจ้าท่า เสือเหลือง สาวใหม่ ซึ่งปัจจุบันเรียกว่าสาวน้อยปะแป้ง เป็นต้น โดยมีความเชื่อว่าเป็นต้นไม้ที่จะทำให้ธุรกิจการค้าขายรุ่งเรืองเป็นสิริมงคล เมตตามหานิยม และความสงบสุขมาให้แก่ผู้อยู่อาศัยจนช่วงปี พ.ศ. 2472-2475 การปลูกเริ่มแพร่หลายไปสู่ประชาชน เริ่มจากบรรดาขุนนางที่ได้นำไปถวายพระสงฆ์ บุคคลที่สนิทสนม และบุคคลสำคัญ (ไทยรัฐ, 2564)

บอนสีเป็นไม้ประดับที่ได้รับความนิยมมาตั้งแต่อดีต แม้แต่ในยุคที่สื่อเข้ามามีบทบาทกับทุกคน บอนสีก็ยังครองใจคนรักต้นไม้ยังไม่เปลี่ยนแปลง ที่เป็นเช่นนี้เพราะความงามของใบสีสันทึบหลากลาย ทั้งแดง ชมพู เขียว ขาว เหลือง น้ำตาล ฯลฯ และลีลาการแตงแต้มที่แตกต่างกัน ป้ายสีหนึ่งกับสีหนึ่ง มีเม็ดสีเล็ก-ใหญ่กระจายไปทั่ว หรือมีเส้นสีที่จัดจ้านพาดผ่านทางใบ ไม่นับรวมรูปใบ และทรงพุ่มที่มีเสน่ห์เป็นเอกลักษณ์ชวนมอง ที่สำคัญความงามของบอนสีไม่ได้หยุดอยู่เพียงแค่พันธุ์ใดพันธุ์หนึ่ง หากแต่ได้รับการปรับปรุงสายพันธุ์จากนักปลูกเลี้ยงบอนสีอยู่ตลอดเวลา ซึ่งเกิดลูกผสมแปลกใหม่สวยงามให้ได้ชมอย่างหลากหลาย จึงได้รับสมญานามว่า “ราชินีแห่งไม้ใบ” (สมาคมบอนสีแห่งประเทศไทย, 2552)

เมื่อเกิดภาวะโรคระบาด COVID-19 เริ่มต้นขึ้นในเดือนธันวาคม พ.ศ. 2562 ซึ่งทำให้สถานประกอบการหลายแห่งจึงต้องปรับการทำงาน โดยให้พนักงานทำงานที่บ้าน (Work From Home: WFH) เมื่อผู้คนเริ่มอยู่บ้านมากขึ้น ทำให้ต้องปรับตัวในหลาย ๆ ด้าน อีกทั้งใน

ปัจจุบันเกิดกระแสการเลี้ยงต้นไม้ขนาดเล็กถึงปานกลางเป็นที่นิยม ไม้ประดับบอนสีจึงเป็นหนึ่งในตัวเลือกของหลาย ๆ คน และข่าวจากมติชนได้มีการสำรวจถึงการค้นหาที่คนไทยนิยมในช่วงเดือนมกราคม-พฤษภาคม พ.ศ. 2564 โดยอันดับที่ 1 คือ ต้นบอนสี เพิ่มขึ้นร้อยละ 1000 รองลงมา คือ เครื่องวัดออกซิเจนร้อยละ 600 (มติชน, 2564) ด้วยความหลากหลายของขนาดทำให้ผู้ที่พื้นที่ใช้สอยภายในบ้านน้อยสามารถเลี้ยงได้ และผู้คนเริ่มหันมาสนใจจึงซื้อไม้ประดับบอนสี แต่ด้วยในหนึ่งประเภท มีหลายชนิด เช่น สายพันธุ์บอนสีพื้นเมือง ได้แก่ บอนสีวัวแดง บอนสีเจ้ากรุงเดนมาร์ก เป็นต้น จึงเกิดความเข้าใจผิดเนื่องจากลักษณะใบที่คล้ายคลึงกัน และในปัจจุบันเทคโนโลยีมีความก้าวหน้ามากยิ่งขึ้นจากอดีตในหลายด้าน และเข้ามามีบทบาทต่อการดำเนินชีวิต ซึ่งมีการพัฒนาเกิดขึ้นอย่างต่อเนื่อง ทั้งในด้านการคิดค้นนวัตกรรม ด้านการติดต่อสื่อสาร ด้านธุรกิจ ด้านการเงิน ด้านอุตสาหกรรม ด้านการเกษตร ฯลฯ จึงส่งผลให้เกิดการเปลี่ยนแปลงวิถีชีวิตในปัจจุบัน ทำให้ต้องมีการปรับตัวให้เข้ากับสภาพแวดล้อมทางสังคมที่เปลี่ยนแปลง เช่น การสแกนจ่ายเงินผ่านแอปพลิเคชัน (Application) การซื้อขายออนไลน์ผ่านแพลตฟอร์มต่าง ๆ ซึ่งในการค้นหาสามารถค้นหาได้จากการถ่ายภาพ หรือนำเข้าภาพ ดังนั้น การจำแนกสิ่งต่าง ๆ ทั้งมีชีวิต และไม่มีชีวิตต้องอาศัยความเชี่ยวชาญ ประสบการณ์ และอุปกรณ์ ทั้งนี้ต้องมีองค์ความรู้ในเรื่องนั้น ๆ ตัวอย่างเช่น การระบุประเภทไม้ประดับบอนสี การสำรวจหาทรัพยากร การศึกษาทางด้านความหลากหลายทางชีวภาพ เป็นต้น เนื่องจากในปัจจุบันเทคโนโลยีเข้ามามีบทบาทในชีวิตประจำวัน ทำให้ข้อมูลต่าง ๆ สามารถเข้าถึง ศึกษา เรียนรู้ ได้ง่ายแต่ก็ยังมีข้อจำกัดในด้านต่าง ๆ ทำให้การระบุประเภทไม้ประดับบอนสีด้วยการมองเห็นเป็นที่น่าสนใจมากขึ้น

เทคโนโลยีในปัจจุบันมีการพัฒนาไปอย่างมากทำให้เกิดข้อมูลในรูปแบบของรูปภาพมากขึ้น ข้อมูลในลักษณะรูปภาพนี้เป็นที่นิยม และยังแพร่กระจายไปอย่างรวดเร็วในสังคมออนไลน์ ซึ่งเทคโนโลยีทางด้านคอมพิวเตอร์วิทัศน์ (Computer Vision) ได้มีการพัฒนาไปอย่างก้าวหน้ารวดเร็วโดยมีการประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึก (Deep learning) โดยเฉพาะการใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolution Neural Networks: CNN) ซึ่งมีโครงสร้างที่ใช้ในการแยกคุณลักษณะสำคัญ แล้วนำคุณลักษณะนั้นมาช่วยในการสร้างโมเดลที่มีประสิทธิภาพสูงในการทำนายวัตถุ

ด้วยเหตุนี้การนำเทคโนโลยีการประมวลผลภาพ (Image Processing) มาประยุกต์ใช้กับไม้ประดับบอนสีทำให้สามารถนำภาพถ่ายของไม้ประดับบอนสีมาทำการประมวลได้เพิ่มความสะดวกสบายในการระบุประเภทของไม้ประดับบอนสีในกรณีที่ไม้ทราบประเภทของไม้ประดับบอนสี

ปัจจุบันมีหลายงานวิจัยที่ใช้เทคโนโลยีในการระบุประเภทของสิ่งต่างๆ ด้วยวิธีการประมวลผลภาพ (Image processing) เทคโนโลยีทางด้านคอมพิวเตอร์วิทัศน์ (Computer Vision) โดยประยุกต์ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolution Neural Networks:

CNN) ตัวอย่างเช่น งานวิจัย การตรวจหาต้นไม้เป็นโรคโดยอัตโนมัติด้วยภาพถ่ายมุมสูงจากโดรน และวิธีการเรียนรู้เชิงลึก โดยใช้ Mask R-CNN (รัตนโชติ พันธุ์วิไล, 2562) งานวิจัย การตรวจจับโลโก้สินค้าแอลกอฮอล์ด้วยแบบจำลองการเรียนรู้เชิงลึก (พิชิตชัย พิมพ์โคตร, 2562) โดยใช้ชุดข้อมูล Logos-32Plus มีจำนวน 6 ยี่ห้อ โดยใช้ Mask R-CNN งานวิจัย Comparison of Vision-based and CNN-based Classifiers for Fish Monitoring in Complex Environment (Ling Yi Jun and Lau Phooi Yee, 2565) โดยใช้วิธีในการตรวจจับปลา 2 วิธีคือ 1) Extended UTAR Aquaculture Farm Fish Monitoring System Framework (UFFMS) และ 2) Faster Region Convolutional Neural Network (Faster R-CNN) และงานวิจัย การจำแนกผลึกน้ำตาลด้วยวิธีการเรียนรู้เชิงลึก (สุริยะ ชยะธรรมกุล, 2563) โดยโมเดล VGG16 ที่ผ่านการปรับแต่งมีความแม่นยำในการจำแนกผลึกน้ำตาลระหว่าง 80% ถึง 94%

จากประเด็นดังกล่าว จึงทำให้เกิดแนวคิดในการพัฒนาระบบตรวจจับ และระบุประเภทไม้ประดับบอนสีด้วยการประมวลผลภาพ ซึ่งเป็นการประยุกต์ใช้กับเทคนิคการประมวลผลภาพ (Image Processing) เพื่อช่วยในการระบุประเภทของไม้ประดับบอนสีด้วยตัวเอง อีกทั้งช่วยลดระยะเวลาในการค้นหา

1.2 วัตถุประสงค์

เพื่อให้สามารถระบุประเภทไม้ประดับบอนสีโดยใช้เทคนิคการประมวลผลภาพ และโครงข่ายประสาทเทียมแบบคอนโวลูชัน

1.3 ขอบเขตของการศึกษา

1.3.1 ด้านเนื้อหา

ศึกษาเกี่ยวกับพันธุ์ไม้ประดับบอนสี และกระบวนการทำงานของการตรวจจับการประมวลผลภาพ

1.3.2 ด้านกลุ่มตัวอย่าง

ประเภทไม้บอนสี จำนวนทั้งหมด 4 ประเภท ซึ่งแต่ละประเภทจะมีหลากหลายชนิดทางผู้วิจัยจึงกำหนดใช้กลุ่มตัวอย่างจากประเภทบอนสี ซึ่งเก็บภาพตัวอย่างชนิดละ 375 ภาพ โดยจัดเป็นชุด Training Set จำนวน 300 ภาพ และชุด Test Set จำนวน 75 ภาพ รวมทั้งสิ้น 1,500 ภาพ

ตัวแปรที่ใช้ในการวิจัย

- ตัวแปรอิสระ ได้แก่ ภาพถ่ายไม้ประดับบอนสี
- ตัวแปรตาม ได้แก่ ประเภทไม้ประดับบอนสี

1.3.3 ด้านพื้นที่

- สวนมาลัยทอง อำเภอไทรน้อย จังหวัดนนทบุรี
- สวนนพดล อำเภอไทรน้อย จังหวัดนนทบุรี

1.3.4 การดำเนินโครงการ

กิจกรรม	ปีงบประมาณ พ.ศ. 2566											
	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	มิ.ย.	ก.ค.	ส.ค.	ก.ย.
1. ศึกษาและเก็บรวบรวมข้อมูล												
2. เก็บตัวอย่างชุดข้อมูล												
3. เขียนโปรแกรมภาษา Python												
4. ทดสอบและแก้ไขข้อผิดพลาด												
5. ปรับปรุง												
6. จัดทำเอกสารประกอบ												
7. สรุปผลดำเนินโครงการ												
8. สอบจบโครงการ												

1.3.5 เครื่องมือที่ใช้ในการวิจัย

- ฮาร์ดแวร์ (Hardware) ได้แก่
 1. สมาร์ทโฟน Apple รุ่น iPhone XR 1 เครื่อง
 2. เครื่องคอมพิวเตอร์ (Computer) 1 เครื่อง
INBook_X1 Processor Intel® Core(TM) i3-1005G1
CPU @1.20GHz 1.19 GHz Ram 8.00 GB System
type 64-bit
- ซอฟต์แวร์ (Software) ได้แก่
 1. ภาษา Python
 2. Google Colaboratory

1.4 ประโยชน์ที่คาดว่าจะได้รับ

สามารถระบุประเภทไม้ประดับบอนสี โดยใช้เทคนิคการประมวลผลภาพในการวิเคราะห์โดยสามารถให้ความถูกต้อง ความแม่นยำไม่น้อยกว่าร้อยละ 80

1.5 นิยามศัพท์เฉพาะ

1. ระบบตรวจจับ หมายถึง ชุดอุปกรณ์ เช่น เซอร์ วังจร หรือ ระบบ ที่ทำหน้าที่ตรวจวัดการเปลี่ยนแปลงคุณสมบัติ หรือลักษณะของสิ่งต่าง ๆ โดยรอบวัตถุเป้าหมาย และนำข้อมูลจำนวนมหาศาล (Big Data) ที่ได้จากการตรวจวัด เข้าสู่กระบวนการแจกแจง และวิเคราะห์พฤติกรรมของการเปลี่ยนแปลง ประมวลผลเป็นองค์ความรู้และปัญญาประดิษฐ์ (Artificial Intelligence) ให้มนุษย์สามารถนำองค์ความรู้มาใช้เพื่อเพิ่มประสิทธิภาพลดขั้นตอนของกระบวนการทำงาน

2. ประเภทบอนสี หมายถึง สายพันธุ์บอนสีแบ่งออกเป็น 3 กลุ่มใหญ่ๆ ด้วยกัน ดังนี้

- สายพันธุ์บอนสีพื้นเมือง เช่น บอนสีอิเหนา บอนสีฮกหลง บอนสีแดงวัว บอนสีมานนางพิมพ์ บอนสีปาเต๊ะ และบอนสีเจ้ากรุงเดนมาร์ก

- สายพันธุ์บอนสีลูกผสมในไทย บอนสีที่เกิดจากการผสมเกสรหรือการผ่าหัวแล้วกลายพันธุ์ เช่น บอนสีสายชล บอนสีหนูมาน บอนสีรุ่งสามสี บอนสีม่วงมงคล บอนสีเทพอัมรินทร์ บอนสีเพชรนิลจินดา บอนสีหนูมานเข้าเฝ้า บอนสีแสงประกายสร บอนสีเทพเทวฤทธิ์ บอนสีใบบุญ บอนสีปิ่นรัตน์ บอนสีเจ้าชาย บอนสีเทพทรงทาน บอนสีเทพประทาน บอนสีศรีสหัส บอนสียุทธหัตถี

- สายพันธุ์ต่างประเทศ เช่น ไวท์คริสต์มาส โปสแมน และฟิงค์กิม

3. ไม้ประดับ หมายถึง พืชที่ปลูกขึ้นเพื่อใช้ประโยชน์จากรูปร่าง รูปทรง สีสีนของลำต้น และใบ พืชชนิดนี้จะมีรูปทรง รูปร่าง สีสีนของลำต้นและใบสวยงามแตกต่างกันไป นิยมปลูกประดับตกแต่งอาคารสถานที่ทั้งในพื้นที่ดิน และในกระถาง มีทั้งไม้ยืนต้นขนาดใหญ่ ไม้พุ่ม และไม้ล้มลุก เช่น ปาล์มต่าง ๆ ช่อย สนชนิดต่าง ๆ ไทรยอดต่าง ฤๅษีผสม เฟิร์นชนิดต่าง ๆ สาวน้อยประแป้ง ว่านกาบหอย เป็นต้น

4. การประมวลผลภาพ (Image Processing) หมายถึง การประมวลผลภาพเป็นกระบวนการจัดการวิเคราะห์สารสนเทศของรูปภาพ โดยใช้คอมพิวเตอร์ในการประมวลผลภาพ

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในการศึกษางานวิจัยเรื่อง “ระบบตรวจจับและระบุชนิดพันธุ์ไม้ประดับบอนสีด้วยการประมวลผลภาพ” ผู้วิจัยได้ศึกษาค้นคว้าทฤษฎี เอกสาร และงานวิจัยที่เกี่ยวข้อง ที่สามารถนำมาประยุกต์เพื่อการพัฒนาาระบบงานวิจัย โดยศึกษาค้นคว้าด้านความรู้ทั่วไปเกี่ยวกับไม้ประดับบอนสี การประมวลผลภาพ

2.1 ความรู้ทั่วไปเกี่ยวกับพันธุ์ไม้ประดับบอนสี

2.1.1 ลักษณะทางพฤกษศาสตร์

บอนสีเป็นพืชในสกุล Caladium วงศ์ Araceae มีชื่อวิทยาศาสตร์ว่า Caladium bicolor เป็นไม้ล้มลุกอายุหลายปี ทุกส่วนอวบน้ำ ดูดซึมน้ำ และคายน้ำได้เร็ว (อรรชรณ วิชัยลักษณ์, 2548) มีลำต้นใต้ดินเป็นหัวแบบมันฝรั่ง (Tuber) ทำหน้าที่สะสมอาหาร ใบมีลักษณะแตกเป็นก้อมีก้านสูงยาวเหนือพื้นดิน ใบใหญ่แผ่กว้าง รูปทรงและสีสันของใบมีหลายแบบขึ้นอยู่กับสายพันธุ์ อาทิ ใบรูปหัวใจ ใบทรงกลม ใบหอก และอื่น ๆ แฉกแต่ละชนิดก็มีสีสัน และลวดลายที่โดดเด่น และสวยงามแตกต่างกันไป มีตาหรือที่เรียกกันว่า “เขี้ยว” รอบหัว สามารถแตกรากฝอยได้รอบมีถิ่นกำเนิดในแถบอเมริกาใต้ ยุโรป อินเดีย และอินโดนีเซีย สันนิษฐานว่ามีการนำเข้ามาปลูกในสมัยสุโขทัยและสมัยรัตนโกสินทร์ ช่วงรัชกาลที่ 5 (สมาคมบอนสีแห่งประเทศไทย, 2564)

- **หัวบอนสี** มี 2 ลักษณะคือ หัวรูปทรงกลม และหัวเป็นแท่งทรงกระบอกคล้ายดินสอ เนื้อในเป็นสีขาวหรือสีเหลืองอ่อน มีน้ำยางที่มีแคลเซียมออกซาเลต (Calcium Oxalate) ซึ่งระคายเคืองต่อผิวหนัง ทำให้คัน ในฤดูหนาวจะพักตัว เมื่อเข้าฤดูร้อนถึงฤดูฝนจะผลิยอดใหม่ที่ตารอบหัว

- **ดอก** เป็น ดอกสมบูรณ์เพศ ประกอบด้วยช่อเชิงลดมีกาบ (Squadix) หรือที่เรียกว่า ปรีดอก และกาบหุ้มช่อดอก (Spathe) หรือที่เรียกว่า จานรองดอก บนปรีดอกมีดอกทั้งเพศเมีย และดอกเพศผู้ที่สมบูรณ์ เมื่อบานจะมีกลิ่นหอมอ่อน ๆ โดยดอกเพศเมียที่ห่อหุ้มด้วยจานดอกด้านล่าง อาจจะบานก่อนดอกเพศผู้ ดังนั้น กว่าดอกเพศผู้จะบานดอกเพศเมียก็ไม่พร้อมผสม ถ้าดอกเพศเมียสามารถติดเมล็ดได้แสดงว่าได้รับละอองเกสรเพศผู้จากต้นอื่น จึงทำให้บอลสีผสมข้ามต้นเกิดเป็นพันธุ์ใหม่

- **ผล** ติดเป็นช่อ เป็นผลมีเนื้อนุ่ม ภายในมี 1-2 เมล็ดสีขาวนวล ขนาดประมาณ 2 มิลลิเมตร แต่ละช่อติดพ่นได้มากถึง 500 เมล็ด

- **ใบ** มีรูปทรงหลายแบบ ทั้งรูปทรงหัวใจ รูปกลม รูปแถบ และรูปขอบขนาน ขึ้นอยู่กับสายพันธุ์ มีสีสันแตกต่างกัน บางสายพันธุ์มีจุดแต้มเรียกว่า “เม็ด” หรือแถบต่างเรียกว่า “ป้าย” บนใบขอบใบเรียบหรือย่นเป็นคลื่น ก้านใบกลมออกจากกึ่งกลางของใบเรียกว่า “สะดือ” บางพันธุ์ก้านใบแผ่ออกเป็นกาบ และมียางค้ำยื่นออกมาจากก้าน

- **ราก** จะเป็นรากฝอยที่งอกออกจากหัวด้านบนระหว่างรอยต่อของหัวบอนกับลำต้น
- **เมล็ด** บอนสีจะมีสีน้ำตาลไหม้ขนาดเล็กเท่าเมล็ดงา

2.1.2 ชนิดพันธุ์บอนสี

สายพันธุ์บอนสีแบ่งออกเป็น 3 กลุ่มใหญ่ ๆ ด้วยกัน (คณะบรรณาธิการสำนักพิมพ์บ้านและสวน, 2564) ดังนี้

- **สายพันธุ์บอนสีพื้นเมือง** เช่น บอนสีโอเหนา บอนสียกหลง บอนสีแดงวัว บอนสีมานนางพิมพ์ บอนสีปาเต๊ะ และบอนสีเจ้ากรุงเดนมาร์ก มักพบในเห็นอยู่ในตามสวนยางในภาคใต้

- **สายพันธุ์บอนสีลูกผสมในไทย** บอนสีที่เกิดจากการผสมเกสรหรือการผ่าหัวแล้วกลายพันธุ์ เช่น บอนสีสายชล บอนสีหนูมาน บอนสีรุ่งสามสี บอนสีม่วงมงคล บอนสีเทพอัมรินทร์ บอนสีเพชรนิลจินดา บอนสีหนูมานเข้าเฝ้า บอนสีแสงประกายสร บอนสีเทพเทวฤทธิ์ บอนสีใบบุญ บอนสีปิ่นรัตน์ บอนสีเจ้าชาย บอนสีเทพทรงทาน บอนสีเทพประทาน บอนสีศรีสหัส บอนสียุทธหัตถี เป็นต้น

- **สายพันธุ์ต่างประเทศ** เป็นสายพันธุ์ที่ปลูกในประเทศสหรัฐอเมริกาเพื่อผลิตหัว เช่น ไวท์คริสต์มาส โปสแมน และฟิงค์กิม เป็นต้น

2.1.3 ประเภทบอนสี

ความแตกต่างของบอนสีแต่ละชนิด แต่ละพันธุ์ อยู่ที่ลักษณะของใบที่มีสีสันลวดลายสวยงามจำแนกรูปใบเป็น 5 ประเภท (อุไร จิรมงคลการ, 2538) ได้แก่

- **บอนใบไทย** : บอนสีไทยโบราณ ลักษณะใบเป็นรูปหัวใจ หูใบยาว เว้าลึกเกือบถึงสะดือ (ส่วนปลายก้านใบที่จรดกับเส้นกลางใบ) ก้านใบอยู่กึ่งกลางใบ มีทั้งปลายใบแหลม และปลายใบมนขึ้นอยู่กับสายพันธุ์ ใบแผ่กว้าง มีขนาดใหญ่ สีสันสวยงาม ใบดก และไม่ทิ้งใบ



ภาพที่ 2-1 บอนใบไทย

ที่มา: อรรรรณ วิชัยลักษณ์ (2548)

- **บอนใบกลม** : สายพันธุ์บอนสีที่พัฒนามาจากบอนใบไทย ลักษณะใบกลมหรือรี หูใบสั้น ปลายใบมน และก้านใบกลมอยู่กึ่งกลางใบ คล้ายกับใบบัว



ภาพที่ 2-2 บอนใบกลม

ที่มา: อรรรรณ วิชัยลักษณ์ (2548)

- **บอนใบยาว** : ลักษณะใบทรงหัวใจคล้ายกับบอนใบไทยเช่นเดียวกัน แต่แตกต่างกันที่ใบของบอนสีใบยาวจะเรียวยาว และมีปลายใบแหลมกว่า หูใบยาวฉีกถึงสะดือ และสามารถแยกได้อีก 3 ลักษณะย่อยคือ บอนใบยาวธรรมดา บอนใบยาวรูปหอก และบอนใบยาวรูปใบไผ่



ภาพที่ 2-3 บอนใบยาว

ที่มา: อรรรรณ วิชัยลักษณ์ (2548)

- บอนใบกาบ : ลักษณะคล้ายกับบอนใบไทย แต่บอนใบกาบจะมีก้านใบที่แผ่แบนตั้งแต่โคนใบไปถึงข้าง (ใบขนาดเล็กที่ยื่นออกจากกาบใบ อยู่กึ่งกลางของก้านหรือต่ำกว่าใบจริงเล็กน้อย)



ภาพที่ 2-4 บอนใบกาบ

ที่มา: อรรรรณ วิชัยลักษณ์ (2548)

- บอนใบไผ่ : ลักษณะใบรูปแถบ รูปใบหอกแคบหรือเป็นเส้น ปลายใบเรียวแหลมคล้ายใบไผ่ หูใบสั้นมาก ความกว้างของใบไม่เกิน 2 นิ้ว



ภาพที่ 2-5 บอนใบไผ่

ที่มา: อรรรรณ วิชัยลักษณ์ (2548)

นอกจากการจำแนกตามรูปร่างแล้ว สามารถจำแนกตามสีสันไปได้
(อุไร จิรมงคลการ, 2538) ดังนี้

- **บอนไม่กัตสี** คือ บอนสีที่มีสีคงที่ไม่เปลี่ยนแปลงตั้งแต่ยังเล็กจนโตเต็มที่หรืออาจมีสีเปลี่ยนแปลงเล็กน้อยแต่ยังคงสีเดิมให้เห็น เช่น นายจันหนวดเขี้ยว บอนสีตบวีรชน เป็นบอนใบไทยที่มีสีแดงตั้งแต่ต้นเล็กจนโตเต็มที่

- **บอนกัตสี** คือ บอนสีที่มีการเปลี่ยนแปลงของสี เมื่อยังเล็กใบเป็นสีเขียวหรือสีน้ำตาล พอโตเต็มที่ก็จะเปลี่ยนเป็นสีชมพูหรือแดง และอาจมีจุดหรือแต้มสี เกิดบนใบส่วนใหญ่เป็นบอนลูกผสมพันธุ์ใหม่

- **บอนป้าย** คือ บอนสีที่มีแถบต่างสีแดงพาดทับบนแผ่นใบสีเขียว ซึ่งเริ่มแสดงลักษณะตั้งแต่ใบที่ 1 หรือใบที่ 2 เช่น อัปสรสวรรค์

- **บอนต่าง** คือ บอนสีที่มีป็นต่างสีขาวอมเขียวหรือขาวอมแดง บนพื้นใบสีเขียวหรือต่างเหลือง

2.2 การประมวลผลภาพ (Image Processing)

การประมวลผลภาพ หมายถึง การเรียกใช้ขั้นตอนหรือกรรมวิธีใด ๆ มากระทำกับภาพ โดยมีวัตถุประสงค์เพื่อปรับปรุงคุณภาพของภาพ ให้ได้ภาพใหม่ที่มีคุณภาพตามที่ต้องการ เช่น ความคมชัด การประหยัดพื้นที่จัดเก็บ หรือใช้สำหรับการประมวลผลในระดับสูง เช่น การหาขอบภาพ และการจดจำรูปร่างลักษณะได้อย่างแม่นยำ โดยทั่วไปวัตถุประสงค์ของการประมวลผลภาพ คือ

- การประมวลผลภาพ (Image Processing) วิธีนี้จะใช้ในกระบวนการประมวลผลภาพ เพื่อให้ได้ภาพออกมา เช่น การตกแต่งด้วยโปรแกรม Adobe Photoshop เป็นต้น

Image Processing: Image in → Image out

- การวิเคราะห์ภาพ (Image Analysis) วิธีนี้จะใช้ในกระบวนการประมวลผลภาพ เพื่อให้ได้ค่าการวัดออกมา เช่น การวัดขนาดในงานอุตสาหกรรม

Image Analysis: Image in → Measurements out

- ความสามารถในการเข้าใจภาพ (Image Understanding) วิธีนี้จะใช้กระบวนการประมวลผลภาพ เพื่อให้ได้ผลลัพธ์ออกมาเป็นความหมาย ตัวอย่างเช่น การจดจำตัวอักษร (Optical Character Recognition : OCR) เป็นต้น (นิตยา สายสวาท และคณะ, 2551)

Image Understanding: image → High-Level Description out

2.2.1 ความรู้เบื้องต้นเกี่ยวกับการประมวลผลภาพดิจิทัล

การประมวลผลภาพดิจิทัล (Digital Image Processing) เป็นการแปลงข้อมูลภาพให้อยู่ในรูปแบบ ข้อมูลดิจิทัล (digital format) หรือข้อมูลเชิงตัวเลข โดยวัตถุประสงค์ของการประมวลผลภาพ แบ่งเป็นสองประเภท คือการปรับปรุงคุณภาพของภาพเพื่อให้มนุษย์สามารถมองเห็นรายละเอียดได้ ชัดเจนมากขึ้น และเพื่อให้คอมพิวเตอร์สามารถแปลความหมายภาพได้ (computer interpretation) ซึ่งจะกล่าวถึงทฤษฎี และรายละเอียดดังต่อไปนี้

2.2.1.1 ภาพดิจิทัล (digital Image) ภาพดิจิทัล คือภาพที่แสดงเป็นแอเรย์ (array) หลายมิติ หรือเป็นเมทริกซ์ (matrix) เชิงตัวเลขภาพดิจิทัลจะแสดงในลักษณะสองมิติ จะมีขนาดความกว้างและความสูงของภาพบน แนวแกน X และแกน Y ส่วนจุดใด ๆ ที่อยู่บนระนาบ XY จะเรียกแทนจุดนั้นว่า พิกเซล (pixel) แต่ละพิกเซลจะแสดงค่าความเข้มแสงของภาพ (นุชรี ธรรมโชติ, 2557)

2.2.1.2 ประเภทของภาพ (Types Of Digital Images) มีการจำแนกภาพออกเป็น 3 ประเภท ได้แก่ ภาพขาวดำ (black and white image (B/W)) หรือภาพไบนารี (binary) ที่แต่ละพิกเซล มีค่า หนึ่งบิต สามารถแสดงได้เพียงสองสี คือสีขาวแทนด้วย 1 และสีดำแทนด้วย 0 ภาพระดับเทา (gray scale image) คือภาพที่มีระดับความเข้มแต่ละพิกเซลสอดคล้องกับความเข้มแสงปกติแสดงใน ระดับสีเทา ซึ่งแต่ละพิกเซลสามารถแสดงเฉดสีจาก 0 (สีดำ) ถึง 255 (สีขาว) มีขนาด 8 บิต หรือ 1 ไบต์ และภาพสี (color image) หรือภาพสี RGB (RGB image) คือภาพสีที่แต่ละพิกเซลมีส่วนประกอบของเวกเตอร์แทนสีแดง (Red) สีเขียว (Green) และสีน้ำเงิน (Blue) แต่ละสีสามารถแสดงค่าความเข้มแสงได้ในช่วง 0-255 ทำให้แต่ละพิกเซลสามารถแสดงสีที่เป็นไปได้ทั้งหมด 256³ สี และแต่ละพิกเซลมีขนาด 24 บิต (นุชรี ธรรมโชติ, 2557)

2.2.1.3 การประมวลผลภาพดิจิทัล (Digital Image Processing) การประมวลผลภาพดิจิทัลเป็นการนำข้อมูลดิจิทัลผ่านขั้นตอนต่าง ๆ ด้วยคอมพิวเตอร์ เช่น การทำให้ภาพมีความคมชัดมากขึ้นการกำจัดสัญญาณรบกวนออกจากภาพ การแบ่งส่วนของวัตถุที่สนใจออก

มาจากภาพ เพื่อให้ได้ข้อมูลที่ต้องการทั้งในเชิงคุณภาพและปริมาณ ซึ่งสามารถอธิบายขั้นตอนพื้นฐานของการประมวลผลภาพด้วยคอมพิวเตอร์ได้ดังต่อไปนี้ (นุชรี ธรรมโชติ, 2557)

1) การจัดสภาพแวดล้อม (scene constraint) เพื่อลดความซับซ้อนในการประมวลผลภาพให้มากที่สุด เนื่องจากความสามารถในการมองเห็น และรับรู้ของอุปกรณ์ประมวลผลมีจำกัด และไม่เทียบเท่ามนุษย์ จึงต้องช่วยลดความยุ่งยากของการประมวลผล ได้แก่ การจัดการกับวัตถุ ระยะระหว่างกล้องถึงวัตถุ การจัดการเรื่องแสง

2) การดึงข้อมูลภาพ (image acquisition) เป็นกระบวนการที่เริ่มตั้งแต่การถ่ายภาพ โดย กล้อง ตลอดจนถึงการดึงข้อมูลภาพเข้าสู่คอมพิวเตอร์ หรืออุปกรณ์ประมวลผล

3) การประมวลผลภาพเบื้องต้น (pre-processing) มีด้วยกันหลากหลายกระบวนการ ตัวอย่างเช่น การลดสัญญาณรบกวนที่ปรากฏขึ้นในภาพ การตรวจจับขอบของวัตถุที่อยู่ในภาพ การแปลงคุณสมบัติทางกายภาพ เช่นการหมุน การเลื่อน การย่อ และการขยาย เป็นต้น การแปลงสี การวิเคราะห์ภาพในเชิงความถี่ การบีบอัดภาพ และอื่น ๆ

4) การแยกบริเวณ (segmentation) เป็นกระบวนการแยกบริเวณภาพที่มีลักษณะร่วมกัน ออกเป็นส่วน ๆ ซึ่งมุ่งเน้นในการแยกวัตถุออกจากพื้นหลัง

5) การคำนวณหาคุณลักษณะของวัตถุ (feature extraction) เป็นการคำนวณหาหรือวัด คุณลักษณะต่าง ๆ ของแต่ละบริเวณหรือของวัตถุแต่ละชิ้นที่อยู่ในภาพ เช่น การคำนวณหาพื้นที่ ซึ่งเป็นเพียงการนับจำนวนพิกเซลของบริเวณหรือพื้นที่นั้น ๆ หรือการหาเส้นรอบวงของวัตถุ ผลที่ได้ของกระบวนการนี้ คือ ค่าคุณลักษณะต่าง ๆ ของวัตถุแต่ละชิ้นที่อยู่ในภาพคุณลักษณะเหล่านี้จะจัดวางอยู่ในรูปเวกเตอร์

6) การจำแนกวัตถุและการแปลความหมาย (Classification And Interpretation) การจำแนกวัตถุคือกระบวนการจัดกลุ่มให้วัตถุที่กำลังพิจารณาอยู่นั้นว่าเป็นวัตถุที่อยู่ในกลุ่มใด โดยอาศัยข้อมูลที่ได้จากการวัดหรือการคำนวณซึ่งเป็นคุณสมบัติของวัตถุนั้น ๆ เปรียบเทียบกับวัตถุตัวอย่างที่อยู่ในแต่ละกลุ่มก่อนที่ระบบจะสามารถตัดสินใจดังกล่าวได้ ระบบจะต้องมีตัวอย่างของวัตถุในแต่ละกลุ่ม สำหรับงานวิจัยที่ผ่านมาที่เกี่ยวข้องกับการคัดแยกขนาดมีการจำแนกด้วยกันหลายวิธี ยกตัวอย่างเช่น ระเบียบวิธีของ K-Nearest Neighbor Classifier (K-NN) ฟัซซี่ (Fuzzy) ซัพพอร์ตเวกเตอร์แมชชีน (SVM) การเรียนรู้ของเครื่องจักร (Machine Learning) และโครงข่ายประสาทเทียม (Artificial Neural Network: ANN) (นศพร ธรรมโชติ และคณะ, 2563)

2.3 ภาษา Python

เป็นภาษาการเขียนโปรแกรมระดับสูงที่นำข้อดีของภาษาต่าง ๆ มารวมไว้ด้วยกัน ถูกออกแบบมาให้เรียนรู้ได้ง่าย และมีไวยากรณ์ที่ช่วยให้เขียนโค้ดสั้นกว่าภาษาอื่น ๆ มีความสามารถ

ใช้ชนิดข้อมูลแบบไดนามิก จัดการหน่วยความจำอัตโนมัติ สนับสนุนกระบวนทัศน์การเขียนโปรแกรม (Programming paradigms) ประกอบด้วย การเขียนโปรแกรมเชิงวัตถุ (OOP : Object Oriented Programming) การเขียนโปรแกรมเชิงคำสั่ง (Imperative Programming) การเขียนโปรแกรมเชิงฟังก์ชัน (Functional) และการเขียนโปรแกรมเชิงกระบวนการ มีลักษณะเป็นภาษาสคริปต์ที่ทำงานร่วมกับภาษาอื่นได้ มีไลบรารีมาตรฐาน และใช้แปลภาษาโปรแกรมให้ทำงานบนระบบปฏิบัติการได้หลากหลาย (Interpreters) ทั้งบน Windows MAC Linux และ Unix นอกจากนี้ยังเป็นโปรแกรมแบบ Open Source ที่นำใช้ได้ฟรี เหมาะสำหรับโปรแกรมทั้งขนาดเล็กและขนาดใหญ่เช่น การสร้างเกม เฟรมเวิร์คพัฒนาเว็บ โปรแกรมที่ใช้กราฟิกติดต่อกับผู้ใช้งาน (GUI) งานคำนวณทางวิทยาศาสตร์ และสถิติ งานพัฒนาซอฟต์แวร์ และซอฟต์แวร์ควบคุมระบบ เป็นต้น (bornToDev Creators, 2563)

2.4 Computer Vision

Computer Vision คือ เทคโนโลยีที่เป็นแขนงหนึ่งของวิทยาการปัญญาประดิษฐ์ (AI) ซึ่งทำการฝึกฝนคอมพิวเตอร์ และระบบให้สามารถเข้าใจ และตอบสนองต่อข้อมูลภาพได้อย่างชาญฉลาดด้วยภาพดิจิทัลจากกล้องถ่ายภาพ และวิดีโอต่าง ๆ และแบบจำลองการเรียนรู้เชิงลึก (Deep Learning) เริ่มจากการทดลองครั้งแรกเริ่มที่เกี่ยวข้องกับเทคโนโลยี Computer Vision นั้น เริ่มต้นขึ้นในทศวรรษ 1950 โดยใช้ประโยชน์จากวงจรโครงข่ายประสาทเทียม (Neural Networks) รุ่นแรกด้วยความพยายามที่จะค้นหาขอบ และมุมของวัตถุต่าง ๆ และจัดหมวดหมู่ให้แก่รูปทรงอย่างง่าย เช่น รูปทรงกลม หรือรูปสี่เหลี่ยม เป็นต้น ต่อมาในช่วงทศวรรษ 1970 ได้มีการนำเทคโนโลยีนี้มาใช้ในการพาณิชย์เป็นครั้งแรกโดยการตีความตัวอักษรที่ถูกเขียนหรือพิมพ์ด้วยเทคนิคการประมวลผลที่เรียกว่า “Optical Character Recognition” ซึ่งนำไปสู่การตีความตัวหนังสือ และข้อความที่เกิดจากการเขียนหรือสิ่งตีพิมพ์ให้แก่ผู้พิการทางสายตา

การพัฒนาสูงสุดสูงสุดของอินเทอร์เน็ตในช่วงทศวรรษที่ 1990 ส่งผลให้รูปภาพปริมาณมากถูกนำขึ้นยังระบบออนไลน์ และสามารถถูกนำมาทำการวิเคราะห์ได้อย่างไม่หยุดยั้ง ซึ่งเป็นปัจจัยกระตุ้นที่ดีสำหรับการเติบโตของโปรแกรมการจดจำใบหน้า ข้อมูลปริมาณที่นับไม่ถ้วนเหล่านี้เติบโตอยู่ตลอดเวลา และช่วยให้อุปกรณ์ต่าง ๆ สามารถทำการระบุตัวตน และจดจำผู้คนต่าง ๆ ได้จากภาพถ่าย และวิดีโอ ทุกวันนี้มีหลายปัจจัยที่เป็นการพัฒนาอย่างรวดเร็วในวิทยาการด้าน Computer Vision ได้แก่ อุปกรณ์พกพา และโทรศัพท์ ซึ่งมีกล้องถ่ายภาพในตัวทำให้ปัจจุบันนี้ผู้คนสามารถเก็บภาพได้ด้วยภาพถ่าย และวิดีโอต่าง ๆ อุปกรณ์ฮาร์ดแวร์ที่ออกแบบมาสำหรับงานด้าน Computer Vision และการวิเคราะห์ด้วยระบบคอมพิวเตอร์ หรืออัลกอริทึมที่ทันสมัย เช่น Convolutional Neural Networks สามารถใช้ประโยชน์จากฮาร์ดแวร์ และซอฟต์แวร์ที่พัฒนาอย่างดีขึ้นในยุค

ปัจจุบันได้อย่างมีประสิทธิภาพ และผลของความก้าวหน้าทางเทคโนโลยี การพัฒนาด้าน Computer Vision เห็นได้อย่างชัดเจน อัตราความแม่นยำของการระบุวัตถุต่าง ๆ และการจัดหมวดหมู่ข้อมูลได้ดีขึ้นจากความแม่นยำร้อยละ 50 มาอยู่ที่ร้อยละ 99 และระบบในปัจจุบันนี้สามารถตรวจจับ และตอบสนองต่อข้อมูลเชิงภาพต่าง ๆ ได้อย่างแม่นยำมากขึ้น (sas.com, ออนไลน์)

กล่าวคือ Computer Vision เป็นการจัดสภาพแวดล้อมทั้งทางด้านซอฟต์แวร์ และฮาร์ดแวร์ของคอมพิวเตอร์ให้พร้อมในการรับรู้ข้อมูลภาพทั้งภาพนิ่ง และภาพเคลื่อนไหว จากนั้นนำมาวิเคราะห์ภาพที่ได้ในด้านต่าง ๆ ด้วยคอมพิวเตอร์เปรียบได้กับดวงตาของมนุษย์เมื่อมองภาพ ทำให้สามารถนำภาพนั้นมาวิเคราะห์ด้วยสมมติว่าเป็นภาพอะไร คุณลักษณะเด่นของภาพมีอะไรบ้าง และเป็นไลบรารีฟังก์ชันการเขียนโปรแกรม (Library Of Programming Functions) โดยส่วนใหญ่จะมุ่งเข้าไปไปที่การแสดงผลด้วยคอมพิวเตอร์แบบเรียลไทม์ (Real-Time Computer Vision) (Nuttakan Chuntra, 2562)

2.5 มาตรฐานของสี

มาตรฐานของสีที่ใช้อยู่ในปัจจุบันมีอยู่หลายระบบด้วยกัน ทั้งนี้จะขึ้นอยู่กับ การนำไปใช้แต่โดยทั่วไปแล้วทุกมาตรฐานจะมีแนวคิดเดียวกันคือการแทนจุดสีด้วยจุดที่อยู่ภายใน สเปส 3 มิติ โดยจะมีแกนอ้างอิงสำหรับจุดสีนั้นในสเปส ซึ่งแต่ละแกนจะมีความเป็นอิสระต่อกัน ตัวอย่างเช่น ในระบบ RGB จะมีแกนสี คือ แกนสีแดง เขียว และน้ำเงินในระบบ HLS จะมีแกน เป็น ค่าสี (hue) 14 ความสว่าง (lightness) และความบริสุทธิ์ ของสี (saturation) ตัวอย่างระบบสี ที่นิยม ใช้กัน ได้แก่ ระบบ RGB HSV (Hue Saturation Value) และ HLS (Hue Lightness Saturation) จำนวนสีสูงสุดที่เป็นไปได้ของแต่ละจุดภาพขึ้นอยู่กับจำนวนบิตที่ใช้ เมื่อมีการกำหนดให้ ขนาดของบิตต่อจุดมากขึ้นจะทำให้จำนวนของสีมากขึ้นด้วย ตัวอย่างเช่น

$$1 \text{ บิต} = 2^1 = 2 \text{ สี}$$

$$2 \text{ บิต} = 2^2 = 4 \text{ สี}$$

$$4 \text{ บิต} = 2^4 = 16 \text{ สี}$$

$$8 \text{ บิต} = 2^8 = 256 \text{ สี}$$

$$16 \text{ บิต} = 2^{16} = 65536 \text{ สี เป็นต้น}$$

2.5.1 ระบบสี RGB

เป็นระบบสีที่เกิดจากการรวมกันของแสงสีแดง เขียวและน้ำเงินโดยมีการรวมเข้าด้วยกัน ซึ่งโดยปกติจะนำไปใช้ในจอภาพแบบ CRT (Cathode ray tube) ในการใช้งานระบบสี RGB ยังมีการสร้างมาตรฐานที่แตกต่างกันออกไปที่นิยมใช้งาน ได้แก่ RGBCIE และ RGBNTSC

2.5.2 ระบบสี HSV

ระบบสี HSV (Hue Saturation Value) เป็นการพิจารณาสีโดยใช้ Hue Saturation และ Value ซึ่ง Hue คือ ค่าสีของสีหลัก (แดง เขียว และน้ำเงิน) ในทางปฏิบัติจะอยู่ระหว่าง 0 และ 255 ซึ่งถ้า Hue มีค่าเท่ากับ 0 จะแทนสีแดงและเมื่อ Hue มีค่าเพิ่มขึ้นเรื่อย ๆ สีก็จะเปลี่ยนแปลงไปตามสเปกตรัมของสีจนถึง 256 จึงจะกลับมาเป็นสีแดงอีกครั้ง ซึ่งสามารถแทนให้อยู่ในรูปขององศาได้ ดังนี้ คือ สีแดง เท่ากับ 0 องศา สีเขียวเท่ากับ 120 องศา สีน้ำเงินเท่ากับ 240 องศา Hue สามารถคำนวณได้จากระบบสี RGB (ยูภาวดี อินตาเครือ และรัฐสิทธิ์ สุขะหุต, 2554) ดังนี้

$$\text{red h} = \text{red} - \min(\text{red green blue})$$

$$\text{green h} = \text{green} - \min(\text{red green blue})$$

$$\text{blue h} = \text{blue} - \min(\text{red green blue})$$

2.6 การตรวจจับวัตถุ (Object Detection)

หนึ่งในพีเจอร์หลักของ AI (Artificial Intelligence) ที่ใช้กับกล้องวงจรปิดสามารถค้นหาสิ่งของโดยใช้ AI มาวิเคราะห์ข้อมูลจากการมองเห็นของคอมพิวเตอร์ (Computer Vision) และการประมวลผลภาพ (Image Processing) เพื่อตรวจจับวัตถุที่อยู่ในรูปหรือวิดีโอ เช่น มนุษย์ สัตว์ สิ่งของ รถยนต์ อาคาร และวัตถุอื่น ๆ ที่อยู่ในรูปภาพหรือวิดีโอ โดยตามหลักแล้วก่อนที่จะพัฒนามาเป็นเทคโนโลยีตรวจจับวัตถุ (Object detection) จะต้องผ่านการจัดหมวดหมู่ของวัตถุ (Object Classification) โดยที่การจัดหมวดหมู่ของวัตถุจะเป็นการจัดหมวดหมู่ของรูปภาพว่ารูปภาพนั้นคือภาพอะไร แต่เทคโนโลยีตรวจจับวัตถุจะเป็นการระบุเลยว่าในรูปภาพนั้นมีวัตถุอะไรบ้าง ซึ่งจะต้องอาศัยการทำงานของ AI เข้ามาช่วยในการวิเคราะห์ข้อมูล เช่น การจัดหมวดหมู่ของวัตถุจะสามารถระบุได้ว่าวัตถุที่อยู่ในภาพ คือ ต้นไม้ ในขณะที่เทคโนโลยีตรวจจับวัตถุจะระบุได้ว่าวัตถุที่อยู่ในภาพ มีต้นไม้ และนก (Beam Warasanan, 2564)

2.7 การจำแนกประเภทข้อมูลภาพ (Image Classification)

การจำแนกประเภทข้อมูลภาพเป็นการประมวลผลในทางสถิติ เพื่อแยกข้อมูลจุดภาพทั้งหมดที่ประกอบเป็นพื้นที่ศึกษาออกเป็นกลุ่มย่อย โดยใช้ลักษณะทางสถิติเป็นตัวกำหนดความแตกต่างระหว่างกลุ่มจุดภาพ โดยจุดภาพที่ถูกจัดให้อยู่กลุ่มเดียวกันจะมีลักษณะทางสถิติเฉพาะกลุ่มเป็นไปในทิศทางเดียวกัน แต่ละกลุ่มจุดภาพที่จำแนกได้นั้นจะแสดงถึงสิ่งปกคลุมพื้นดินประเภทใดประเภทหนึ่งแตกต่างกันไป การจำแนกประเภทข้อมูลภาพด้วยระบบคอมพิวเตอร์แบ่งออกได้เป็น 2 วิธี คือ การจำแนกประเภทข้อมูลภาพแบบมีผู้สอน (Supervised Classification) และการจำแนก

ประเภทข้อมูลภาพแบบไม่มีผู้สอน (Unsupervised Classification) การจำแนกประเภทข้อมูลภาพทั้งสองวิธีให้ได้ผลลัพธ์ที่มีประสิทธิภาพนั้น ก่อนเริ่มจำแนกประเภทข้อมูลภาพ ควรศึกษาสถิติของข้อมูลภาพในแต่ละช่วงคลื่นก่อน เพื่อให้ได้ช่วงคลื่นที่เหมาะสมในการใช้จำแนกประเภทข้อมูลภาพ ค่าสถิติเบื้องต้นที่ใช้ในการเลือกช่วงคลื่นที่เหมาะสม ได้แก่

- ค่าการสะท้อนต่ำสุด-สูงสุด (Minimum-Maximum value) ของแต่ละช่วงคลื่นเป็นค่าที่แสดงการสะท้อนของข้อมูลภาพในแต่ละช่วงคลื่นว่ามีค่าการสะท้อนตกอยู่ในช่วงใด ระหว่างค่า 0-255 หากเป็นค่าที่อยู่ใกล้ 0 มาก หมายถึง ช่วงคลื่นนั้นจะให้ข้อมูลเกี่ยวกับวัตถุที่มีการดูดกลืนพลังงานมาก หากค่าค่อนข้าง 255 หมายถึงช่วงคลื่นนั้นจะให้ข้อมูลเกี่ยวกับวัตถุที่มีการสะท้อนพลังงานสูง และหากมีช่วงค่ากว้าง คือ มีทั้งค่าต่ำสุดใกล้ 0 และมีค่าสูงสุดใกล้ 255 หมายถึง มีข้อมูลของกลุ่มวัตถุทั้งที่ดูดกลืนพลังงาน และที่สะท้อนพลังงาน ถือเป็นช่วงคลื่นที่มีความหลากหลายของข้อมูลมาก

- ค่าเฉลี่ยเลขคณิต (Mean) เป็นค่าเฉลี่ยค่าการสะท้อนทั้งหมดของแต่ละช่วงคลื่นสามารถใช้เป็นตัวแทนภาพรวมข้อมูลจุดภาพทั้งหมดของช่วงคลื่นหนึ่ง ๆ ได้ ค่าเฉลี่ยเลขคณิตสามารถคำนวณโดยนำผลบวกของค่าการสะท้อนทั้งหมดมาหารด้วยจำนวนจุดภาพทั้งหมด (ตำราเทคโนโลยีอวกาศและภูมิสารสนเทศศาสตร์)

2.8 การเรียนรู้ของเครื่อง (Machine Learning)

การเรียนรู้ของเครื่อง เป็นศาสตร์หนึ่งของปัญญาประดิษฐ์ (Artificial Intelligence: AI) เป็นกระบวนการพัฒนาอัลกอริทึมที่ทำให้คอมพิวเตอร์มีความสามารถในการเรียนรู้ด้วยตนเอง สามารถแบ่งการเรียนรู้ได้ 3 แบบ (นงเยาว์ สอนจะโปะ, 2561) ได้แก่

- 1) การเรียนรู้แบบมีผู้สอน (Supervised Learning) เป็นการเรียนรู้โดยอาศัยข้อมูลที่ป้อนเข้าไปเก็บไว้เป็นตัวอย่าง เพื่อให้คอมพิวเตอร์ใช้ในการเปรียบเทียบกับข้อมูลที่เข้ามาใหม่แล้วทำนาย หรือจัดหมวดหมู่ที่มีความเหมือนกันมากที่สุดให้อยู่ด้วยกัน ผลลัพธ์ที่ได้คือ การจัดหมวดหมู่ (Classification) และการวิเคราะห์การถดถอย (Regression)

- 2) การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) การเรียนรู้จะพยายามรู้จำและสร้างโมเดลตัวแทนจากข้อมูลฝึกสอนโดยไม่มีคลาสเป้าหมาย กลไกการเรียนรู้จะพยายามหาความสัมพันธ์ของข้อมูลสอน และจัดกลุ่มข้อมูลสอนตามเงื่อนไขที่สนใจ

- 3) การเรียนรู้แบบกึ่งมีผู้สอน (Semi-supervised Learning) ข้อมูลนำเข้ามีคลาสเป้าหมายเพียงบางส่วน กลไกการเรียนรู้จะเรียนรู้จากส่วนที่มีผู้สอนก่อนที่จะปรับปรุงการเรียนรู้จากส่วนที่ไม่มีผู้สอนนอกจาก 3 รูปแบบข้างต้นแล้วปัจจุบันการเรียนรู้แบบเสริมกำลัง (Reinforcement learning) ที่เป็นที่สนใจในงานด้านการออกแบบความฉลาดของหุ่นยนต์ หรือระบบที่ต้องมี

การต่อประสานกับสิ่งแวดล้อมภายนอก ตอบสนองต่อผลย้อนกลับที่รับเข้ามาแล้วใช้ความฉลาดในการตัดสินใจ (นศพร ธรรมโชติ และคณะ, 2563)

งานวิจัยนี้จะใช้วิธีการเรียนรู้แบบมีผู้สอนชุดหลักฐานที่นำมาใช้จะมีคลาสเป้าหมาย เพื่อให้กลไกการเรียนรู้ได้สร้างโมเดลตัวแทนเพื่อใช้ในการทำนายข้อมูลที่ไม่เคยฝึกสอนมาก่อน

2.9 ทฤษฎีการเรียนรู้เชิงลึก (Deep learning)

วิธีการเรียนรู้เชิงลึก เป็นซอฟต์แวร์คอมพิวเตอร์ที่เลียนแบบให้คล้ายกับเครือข่ายเซลล์ประสาทในสมองของมนุษย์ (Network of Neuron) ซึ่งการเรียนรู้เชิงลึก (Deep Learning) นั้นเป็นส่วนหนึ่งเล็ก ๆ อยู่ใน การเรียนรู้ของเครื่อง (Machine Learning) ส่วน Algorithm ของการเรียนรู้เชิงลึก (Deep Learning) ถูกสร้างขึ้นจากการนำเอา Neural Network หลาย ๆ layer มาต่อกัน โดย layer แรกสุดจะทำหน้าที่ในการรับข้อมูล (Input Layer) layer สุดท้ายจะทำหน้าที่ส่งผลลัพธ์การประมวลผลออกมา (Output layer) ส่วน layer ระหว่าง layer แรกสุด และ layer สุดท้ายจะถูกเรียกว่า Hidden layer โดยที่ Hidden layer ของแต่ละ layer จะเปรียบเสมือนว่าประกอบด้วยเซลล์ประสาท (Neural) จำนวนมาก มีหน้าที่ในการประมวลผลรับข้อมูลจาก layer ที่อยู่เหนือกว่า และส่งข้อมูลที่ประมวลผลเสร็จแล้วไปยัง layer ที่อยู่ต่ำกว่า ข้อดีของการส่งข้อมูลแบบนี้ก็คือ layer แต่ละ layer สามารถที่จะมีค่าถ่วงน้ำหนัก (Weight) ค่าความเอนเอียงของข้อมูล (Bias) และวิธีการประมวลผลทางคณิตศาสตร์ (Activation Function) ที่เป็นอิสระต่อกัน เมื่อมีข้อมูล Input ให้กับโมเดล (Model) มากเท่าไร layer แต่ละ layer ก็จะสามารถสกัด Feature ที่มีความซับซ้อนมากยิ่งขึ้น (รัตนโชติ พันธุ์วิไล, 2562)

การเรียนรู้เชิงลึก (Deep learning) เหมาะกับโครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional neural networks: CNN) ใน layer แรกของ neural network จะเรียนรู้ในรายละเอียดเล็ก ๆ จากรูปภาพ layer ถัดมาจะประกอบไปด้วยความรู้ที่เรียนรู้มาจาก layer ก่อนหน้านี้เพื่อสร้างข้อมูลที่มีความซับซ้อนมากขึ้นใน CNN, feature extraction จะทำการใช้งานฟิลเตอร์ (filter) โครงข่ายจะประยุกต์ฟิลเตอร์ (filter) กับรูปภาพเพื่อทำให้มองเห็นถ้ามีสิ่งๆ เหมือนกันเกิดขึ้นกับที่เคยเจอหรือรูปทรงของ feature นั้นเหมือนกับส่วนใดส่วนหนึ่งของรูปภาพ ซึ่งกระบวนการของการสกัดคุณสมบัติ (feature extraction) จะทำโดยอัตโนมัติ (Nessessence, 2562)

2.10 โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks: CNN)

โครงข่ายประสาทเทียมเป็นโครงข่ายที่มีรูปแบบโครงสร้าง และการประมวลผล เหมือนกับสมองมนุษย์ที่มีการปรับเปลี่ยนตัวเองต่อการตอบสนองของอินพุตตามกฎการเรียนรู้หรือ

อาจกล่าวได้ว่าโครงข่ายประสาทเทียมเป็นการเลียนแบบการทำงานของสมองมนุษย์โครงข่ายประสาทเทียมที่ได้รับความนิยม คือ โครงข่ายแบบคอนโวลูชัน (Convolutional Neural Networks: CNN)

โครงข่ายประสาทเทียมแบบคอนโวลูชันจะตัดขั้นตอนในส่วนของการหาคุณลักษณะพิเศษ (Feature Extraction) แต่ในขั้นตอนของการคอนโวลูชันเป็นขั้นตอนหนึ่งในการคำนวณหาคุณลักษณะพิเศษ การทำงานของโครงข่ายประสาทเทียมแบบคอนโวลูชันจะทำงานควบคู่กับชั้นพูลลิ่งเพื่อช่วยลดขนาดของเมทริกซ์ (Matrix) ที่ใช้ในการคำนวณลง ดังนั้น การเรียนรู้เชิงลึกที่ใช้โครงสร้างแบบ CNN จึงเหมาะสมกับข้อมูลประเภทรูปภาพ (จักรินทร์ สุนุกแสน และโอฬาริก สุรินตะ, 2562)

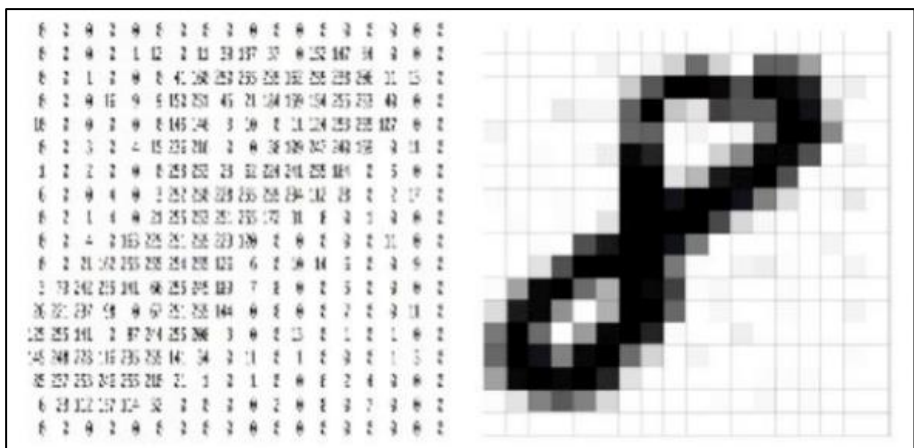
การทำงานของ Convolutional Neural Networks เป็นกระบวนการแบบ Feed-Forward มีส่วนของ Hidden Layer นั้นจะมี Convolutional Layers เพิ่มขึ้นเป็นส่วนที่ใช้ในการฟิลเตอร์ภาพ โดยมี Kernel Function เพื่อใช้แปลงคุณลักษณะและเพื่อแยกองค์ประกอบออกมา เช่น ขอบรูป สี ลักษณะรูปทรง เป็นต้น ก่อนจะนำมาผ่าน Activation Function ที่ช่วยแปลงค่าให้อยู่ในรูปที่ผลให้ได้ผลลัพธ์ดียิ่งขึ้น รวมถึงส่วนที่เรียกว่า พูลลิ่ง (Pooling) โดยส่วนนี้จะทำหน้าที่ปรับขนาดของข้อมูลให้มีขนาดเล็กลง โดยที่รายละเอียดของข้อมูลนั้นยังคงเดิม และในส่วนสุดท้ายก็จะเป็น Fully connected layer ที่จะคอยเชื่อมต่อในแต่ละชั้นเข้าด้วยกัน

โดยกระบวนการที่กล่าวมาข้างต้นนี้จะแบ่ง ขั้นตอนออกเป็น 4 ขั้นตอนหลัก

- การทำคอนโวลูชัน (Convolution channel kernel)
- การขจัดความเป็นเชิงเส้น (Relu: Activation function rectified linear)
- การทำพูลลิ่ง (Max average pooling)
- การเชื่อมต่อกันของแต่ละเลเยอร์อย่างสมบูรณ์ (Fully connected)

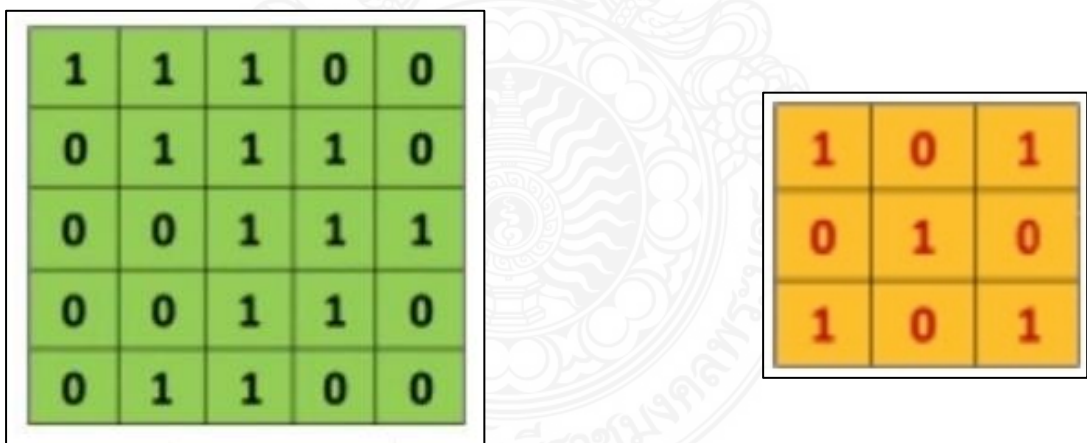
2.10.1 คอนโวลูชัน (Convolution)

คอนโวลูชันจะเป็นกระบวนการที่ให้เราสามารถสกัดเอาลักษณะเด่นของรูปถ่ายออกมา โดยใช้ค่าพิกเซล และค่าพิกเซลจะประกอบด้วยกันใน 3 แชนแนล คือ สีแดง (Red) สีฟ้า (Blue) สีเขียว (Green)



ภาพที่ 2-6 การแทนค่าพิกเซลของรูปลีขาว-ดำ นั้นด้วยตัวเลขตั้งแต่ 0-255
 ที่มา: นายสุริยะ ชยะธรรมกุล (2563)

จากภาพที่ 2-6 แสดงถึงการแทนค่าด้วยตัวเลขที่บอกความเข้มของสีนั้น ๆ โดยมีค่าตั้งแต่ 0 ถึง 255 (แทนความเข้มจากน้อยไปหามาก) โดยในการทำรูปขาว-ดำ แชนแนลของรูปนั้นจะมีเพียงหนึ่งแชนแนลเท่านั้น คือ แชนแนลของสีดำซึ่งตัวเลข 0 คือสีขาวแล้วจะไล่ระดับความเข้มไปจนถึงตัวเลข 255 คือ สีดำ



ภาพที่ 2-7 จำลองเมทริกซ์ที่ได้จากชุดข้อมูลรับเข้าและเมทริกซ์ฟิลเตอร์ (filter)
 ที่มา: นายสุริยะ ชยะธรรมกุล (2563)

1 _{x1}	1 _{x0}	1 _{x1}	0	0
0 _{x0}	1 _{x1}	1 _{x0}	1	0
0 _{x1}	0 _{x0}	1 _{x1}	1	1
0	0	1	1	0
0	1	1	0	0

4		

ภาพที่ 2-8 แสดงหลักการทำงานของฟิลเตอร์ (filter) ค่าและการกำเนิดเมทริกซ์ชุดใหม่หรือพีเจอร์แมพ

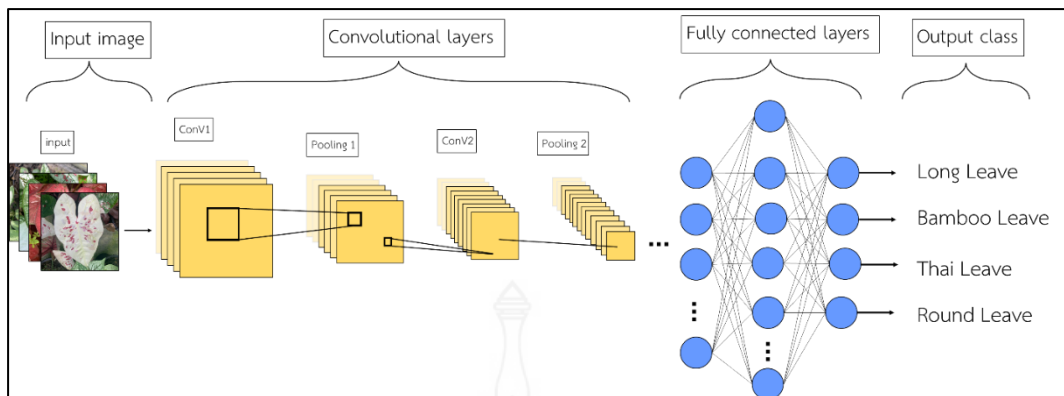
ที่มา: นายสุริยะ ชยะธรรมกุล (2563)

จากภาพที่ 2-8 แสดงถึงแต่ละชุดการคำนวณนี้จะทำการเก็บค่าไว้เป็นข้อมูลในแบบจำลองเมทริกซ์ของคอนโวลูชัน ซึ่งในแต่ละรูปที่ถ่ายได้นั้นจะมีชุดค่าเมทริกซ์ที่ต่างกันไป โดยข้อมูลนำเข้า 5x5 พิกเซลที่เป็นรูปขาวดำโดยที่นี้กำหนดค่าเป็น 0 พิกเซลเป็นสีขาวส่วนค่า 1 เป็นพิกเซลสีดำ และมีการกำหนดเมทริกซ์อีกชุดหนึ่งขึ้นมาเพื่อใช้เป็นตัวกรองค่าฟิลเตอร์ (filter) ไปเก็บไว้ในเมทริกซ์ชุดที่เล็กกว่า เรียกเมทริกซ์กรองค่าชุดนี้ว่า เคอเนล (Kernel) เพื่อทำหน้าที่ตรวจจับลักษณะ และสกัดข้อมูลที่สำคัญ feature detector (นายสุริยะ ชยะธรรมกุล, 2563)

ตัวอย่างการคำนวณ

$$\begin{array}{l}
 1 \times 1 = 1, 1 \times 0 = 0, 1 \times 1 = 1 \qquad 1, 0, 1 \\
 0 \times 0 = 0, 1 \times 1 = 1, 1 \times 0 = 0 \qquad 0, 1, 0 \\
 0 \times 1 = 0, 0 \times 0 = 0, 1 \times 1 = 1 \qquad 0, 0, 1 \quad = 1 + 1 + 1 + 1 = 4
 \end{array}$$

เมทริกซ์ที่ทำหน้าที่เป็นตัวกรองค่าจะเคลื่อนไปทั่วภาพ และคูณค่าเก็บไว้ในเมทริกซ์ชุดใหม่ดังภาพที่ 2-8 ซึ่งเราเรียกเมทริกซ์ใหม่นี้ว่าคอนโวลูชันพีเจอร์ (convolved feature) หรือพีเจอร์แมพ



ภาพที่ 2-9 โครงข่ายประสาทเทียมแบบคอนโวลูชัน

ที่มา: คณะผู้จัดทำ (2565)

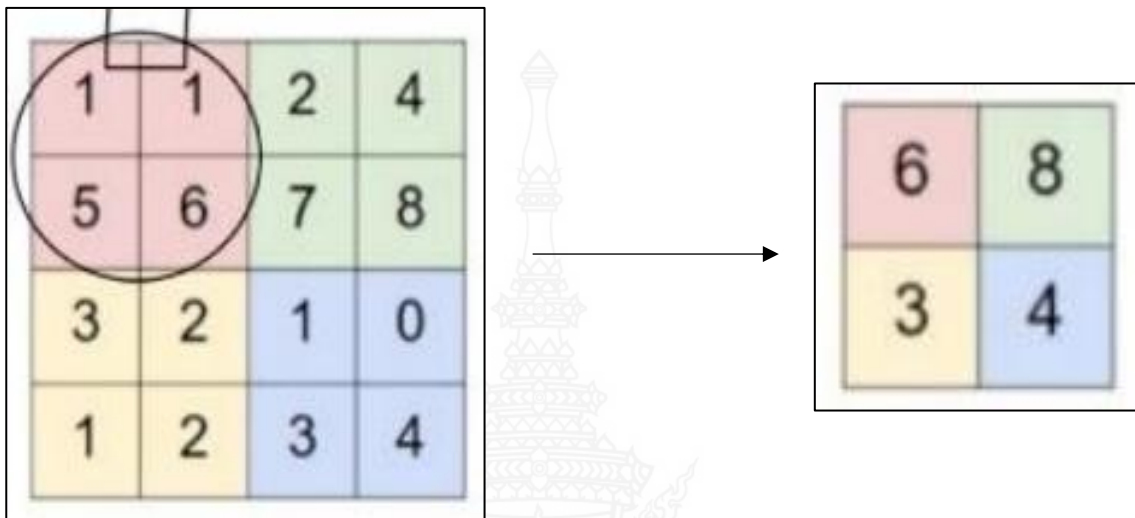
2.10.2 การขจัดความเป็นเชิงเส้น (Relu: Activation Function Rectified Linear)

Relu คือการเติมความสัมพันธ์ที่ไม่เป็นเชิงเส้น (Nonlinear) ให้กับโมเดลเพื่อให้สามารถเรียนรู้ความสัมพันธ์ที่ซับซ้อนระหว่างข้อมูลอินพุต และผลลัพธ์ได้ ฟังก์ชัน Relu หรือ Piecewise Linear เป็นฟังก์ชันไม่เชิงเส้นที่นิยมใช้เป็น Activation Function ในปัจจุบันเนื่องจากทำให้การเรียนรู้ง่ายขึ้นจุดประสงค์ของการทำ Relu นั้นเพื่อให้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks) เรียนรู้ข้อมูลที่ไม่เป็นเชิงเส้น เมื่อนำรูปนั้นเข้าสู่กระบวนการที่เป็นพีเจอร์แมพเข้ามาทำ Relu โดยสีดำ ในรูปเป็นค่าเชิงลบส่วนสีขาวในรูปจะมีค่าเป็นเชิงบวก เมื่อทำการขจัดด้วย Relu จะมีค่าที่ได้เพียงค่าที่เป็นเชิงบวกเท่านั้น ซึ่งส่งผลให้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks) นี้เกิดการเรียนรู้ข้อมูลที่ได้รับมาจากขั้นตอนการตรวจสอบรูปก่อนหน้าตลอดจนข้อมูลนั้นจะถูกแปลงมาอยู่ในรูปของ Non-Linear ซึ่งจะส่งผลให้สามารถเพิ่มประสิทธิภาพรูปของ Network นี้ด้วย (รัสรินทร์ เมธาเฉลิมพัฒน์, 2565)

2.10.3 การทำพูลลิ่ง (Pooling)

การพูลลิ่ง ช่วยลดมิติของพีเจอร์แมพแมปแต่ยังคงรักษาข้อมูลสำคัญไว้ ซึ่งการพูลลิ่ง (Pooling) สามารถจำแนกเป็นประเภทต่าง ๆ ได้เช่น พูลลิ่งด้วยค่าสูงสุด (Max Pooling) ค่าเฉลี่ย (Average Pooling) ผลรวม ซึ่งการพูลลิ่งทำให้ผลลัพธ์ที่ได้มานั้นมีขนาดเล็ก และจัดการได้ง่ายขึ้น นอกจากนี้ยังช่วยลดจำนวนของพารามิเตอร์ และการคำนวณที่เกินจำเป็นใน Network ในกรณีที่ต้องการพูลลิ่งด้วยค่าสูงสุด เราจะกำหนดหน้าต่างหนึ่งขึ้นมา ตัวอย่างในที่นี้สมมติให้หน้าต่างมีขนาด 2x2 พิกเซล และหน้าต่างนี้จะทำการเคลื่อนที่ทีละ 2 พิกเซล ไปจนทั่วเมทริกซ์ของ

พีเจอร์แมพเพื่อทำการเก็บค่าสูงสุดในทุก ๆ 2 พิกเซล ซึ่งนอกจากนี้การพูลลิ่ง (Pooling) จะทำตามจำนวนแชนแนล (Channel) ของรูปซึ่งในกรณีรูปสี่เราจะได้ผลลัพธ์ออกมาถึงสามผลลัพธ์ เมื่อพีเจอร์แมพที่ผ่านการทำ Relu เข้ามาทำการพูลลิ่ง (Pooling)

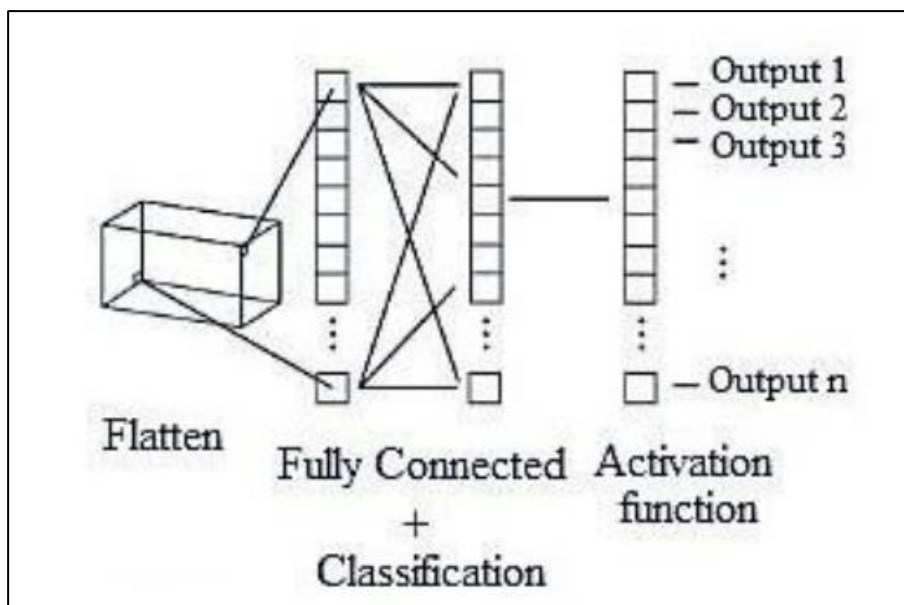


ภาพที่ 2-10 แสดงการพูลลิ่งแบบมากที่สุด (max pooling)

ที่มา: นายสุริยะ ชยะธรรมกุล (2563)

2.10.4 การเชื่อมต่อกันของแต่ละเลเยอร์อย่างสมบูรณ์ (Full Connected Layer)

กระบวนการ Convolution Relu และการ Pooling กระบวนการทั้งสามนี้เป็นกระบวนการที่ทำซ้ำได้หลายครั้ง และในขั้นสุดท้ายจะมีการเชื่อมต่อกันของแต่ละชั้นอย่างสมบูรณ์ Fully Connected Layer ผลลัพธ์จากคอนโวลูชัน และพูลลิ่งนั้นให้ลักษณะเด่น (High-Level Features) ของรูปที่รับเข้ามา และขั้นสุดท้ายเพื่อนำลักษณะเด่นไปทำการคัดกรองรูปที่รับเข้าให้อยู่ในรูปของ Classes



ภาพที่ 2-11 การเชื่อมต่อกันของแต่ละชั้นอย่างสมบูรณ์ (Full connected layer)

ที่มา: นายสุริยะ ชยะธรรมกุล (2563)

2.11 หลักการของ Keras

Keras เป็น Open-Source Neural Networks (นิเวรอล เน็ตเวิร์ค) ที่เขียนด้วยภาษา Python ซึ่งเป็น library Open Source ภาษา Python สำหรับการพัฒนาโครงข่ายประสาทเทียมที่สามารถทำงานบน TensorFlow R Theano หรือ PlaidML ได้ Keras ออกแบบมาให้ผู้ใช้สามารถพัฒนาโปรแกรมด้วยการเรียนรู้เชิงลึกได้อย่างรวดเร็วจึงใช้งานง่าย และมีฟังก์ชันให้เลือกหลากหลายทำงานเป็นสัดเป็นส่วน ซึ่ง Keras ถูกพัฒนาขึ้นในปี พ.ศ. 2560 โดย ฟรอนซ์วส์ ซอลเลต์ วิศวกรของ Google

หลักการของ Keras ถูกสร้างขึ้นเพื่อให้เป็นมิตรกับผู้ใช้แบบแยกส่วนง่ายต่อการขยายและทำงานกับ Python API ได้รับการออกแบบมาเพื่อมนุษย์ไม่ใช่เครื่องจักร และปฏิบัติตามแนวทางปฏิบัติที่ดีที่สุดในการลดภาระการรับรู้เลเยอร์ประสาทฟังก์ชันต้นทุนเครื่องมือเพิ่มประสิทธิภาพ โครงร่างการเริ่มต้นฟังก์ชันการเปิดใช้งาน และโครงร่างมาตรฐานเป็น standalone module ทั้งหมดที่สามารถรวมเข้าด้วยกันเพื่อสร้างโมเดลใหม่ โมเดลใหม่นั้นง่ายต่อการเพิ่มเช่นเดียวกับคลาส และฟังก์ชันใหม่ตัวแบบถูกกำหนดไว้ในรหัส Python ไม่ใช่ไฟล์แยกแบบจำลอง

Keras มีข้อได้เปรียบดังต่อไปนี้

1. Keras อนุญาตให้ใช้ code เดียวกันในการรันทั้ง CPU และ GPU ได้เหมือนกัน
2. API ง่ายต่อการใช้งานและทำให้ตัวต้นแบบในการสร้างโมเดลการเรียนรู้เชิงลึก (Deep Learning) เสร็จเร็ว

3. มีฟังก์ชัน Build-In ในการใช้งาน Convolution Network สำหรับงาน Computer Vision และงาน Recurrent Network สำหรับงาน Sequence Processing

4. สนับสนุนการทำงานของ network รูปแบบใด ๆ เช่น MIMO models Layer Sharing model sharing และอื่น ๆ สิ่งนี้หมายความว่า Keras เหมาะสำหรับการสร้างโมเดลการเรียนรู้เชิงลึก (Deep Learning) ในทุก ๆ แบบจาก Generative Adversarial Network ไปจนถึง Neural Turing Machine (ธเนศ วงศ์หงษ์, 2561)

2.12 ทฤษฎีการถ่ายโอนการเรียนรู้ (Transfer Learning)

เป็นเทคนิคที่ช่วยแก้ปัญหาดังกล่าวมีวิธีการ คือ นำโมเดล (Model) ที่ฝึก (Train) ไว้แล้ว เรียกว่า Pre-trained model มาสกัดเอาคุณลักษณะเด่นแล้วนำมาใช้ฝึก (Train) สร้างโมเดล (Model) ใหม่กับชุดข้อมูล Dataset ชุดใหม่ซึ่งมีจำนวนน้อยกว่า ซึ่งเทคนิคนี้เป็นที่นิยมเนื่องจากจะทำให้ได้โมเดล (Model) ที่มีความแม่นยำสูง โดยที่ไม่ต้องเตรียมข้อมูลจำนวนมาก โดย CNN หรือ ConvNet ประกอบด้วย 2 ส่วนหลัก คือ

1. Feature Extraction ประกอบด้วย Convolution และ Pooling หรือเพิ่มเติมส่วนอื่นๆเช่น Normalization Dropout ซึ่งทั้งหมดจะทำหน้าที่จับคุณลักษณะเด่นของภาพ

2. Classification ทำหน้าที่จำแนกหรือ Predict ทำนาย โดยส่วนนี้ประกอบด้วย Neural

การใช้เทคนิคการถ่ายโอนการเรียนรู้ (Transfer Learning) ทำให้การสร้างโมเดล (Model) ใหม่เป็นเรื่องง่ายขึ้นโดยการนำโมเดล (Model) ที่ผ่านการสอน (Train) แล้วด้วย Dataset ขนาดใหญ่ที่มีรูปแบบ ช่าง สุนัข หรืออื่น ๆ มาตัดต่อบางส่วน แล้วฝึก (Train) ด้วย Dataset ขนาดเล็กกว่า (ขนาดเล็กในที่นี้หมายถึง Dataset มีจำนวนรูปภาพน้อยกว่า) คือมีจำนวนรูปไม่มากการฝึก (Train) ในลักษณะนี้จะทำให้เกิดการส่งถ่ายความรู้เปรียบเหมือนกับการนำ Dataset ขนาดใหญ่จาก Mode ต้นฉบับ Source โมเดล (Model) คือ Pre-trained model มาทำการฝึก (Train) ด้วย ทำให้โมเดล (Model) ที่ได้มีประสิทธิภาพสูงเกือบเทียบเท่ากับโมเดล (Model) ต้นฉบับ (กอบเกียรติ สระอุบล, 2565)

Model	Size (MB)	Top-1 Accuracy	Top-5 Accuracy	Parameters	Depth	Time (ms) per inference step (CPU)	Time (ms) per inference step (GPU)
Xception	88	79.0%	94.5%	22.9M	81	109.4	8.1
VGG16	528	71.3%	90.1%	138.4M	16	69.5	4.2
VGG19	549	71.3%	90.0%	143.7M	19	84.8	4.4
ResNet50	98	74.9%	92.1%	25.6M	107	58.2	4.6
ResNet50V2	98	76.0%	93.0%	25.6M	103	45.6	4.4
ResNet101	171	76.4%	92.8%	44.7M	209	89.6	5.2
ResNet101V2	171	77.2%	93.8%	44.7M	205	72.7	5.4
ResNet152	232	76.6%	93.1%	60.4M	311	127.4	6.5
ResNet152V2	232	78.0%	94.2%	60.4M	307	107.5	6.6
InceptionV3	92	77.9%	93.7%	23.9M	189	42.2	6.9
InceptionResNetV2	215	80.3%	95.3%	55.9M	449	130.2	10.0
MobileNet	16	70.4%	89.5%	4.3M	55	22.6	3.4
MobileNetV2	14	71.3%	90.1%	3.5M	105	25.9	3.8
DenseNet121	33	75.0%	92.3%	8.1M	242	77.1	5.4
DenseNet169	57	76.2%	93.2%	14.3M	338	96.4	6.3
DenseNet201	80	77.3%	93.6%	20.2M	402	127.2	6.7
NASNetMobile	23	74.4%	91.9%	5.3M	389	27.0	6.7
NASNetLarge	343	82.5%	96.0%	88.9M	533	344.5	20.0
EfficientNetB0	29	77.1%	93.3%	5.3M	132	46.0	4.9

ภาพที่ 2-12 โมเดล (Model) ของ CNN ที่ได้ทำการฝึกฝนบน ImageNet และคุณลักษณะแต่ละ

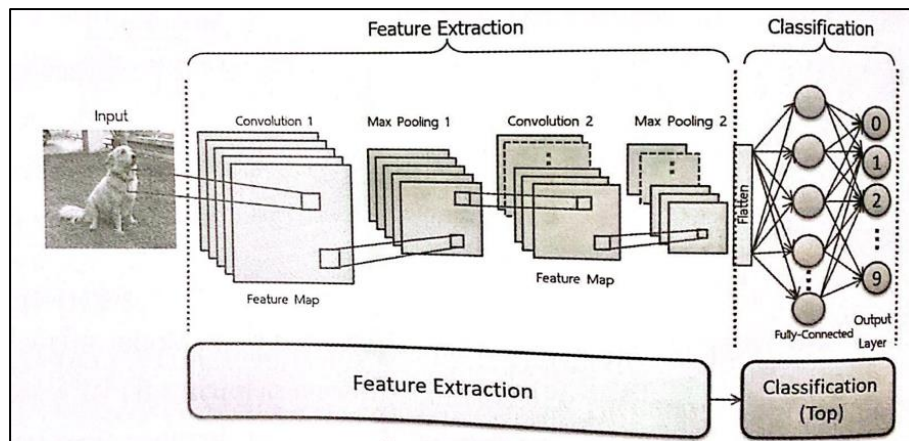
โมเดล (Model)

ที่มา: keras.io (ออนไลน์)

ความแม่นยำ TOP-1 และ TOP-5 หมายถึง ประสิทธิภาพของโมเดล (Model) ในชุดข้อมูลการตรวจสอบความถูกต้องของ ImageNet

Depth หมายถึง ความลึกของรูปแบบการเชื่อมต่อเครือข่ายในแบบต่าง ๆ ซึ่งรวมถึง Layer การเปิดใช้งาน Layer

Time (ms) per inference step โดยใช้ทั้ง CPU และ GPU เป็นค่าเฉลี่ย 30 ชุด และ batches 10 ครั้ง ความลึกนับจำนวน Layer ด้วยพารามิเตอร์



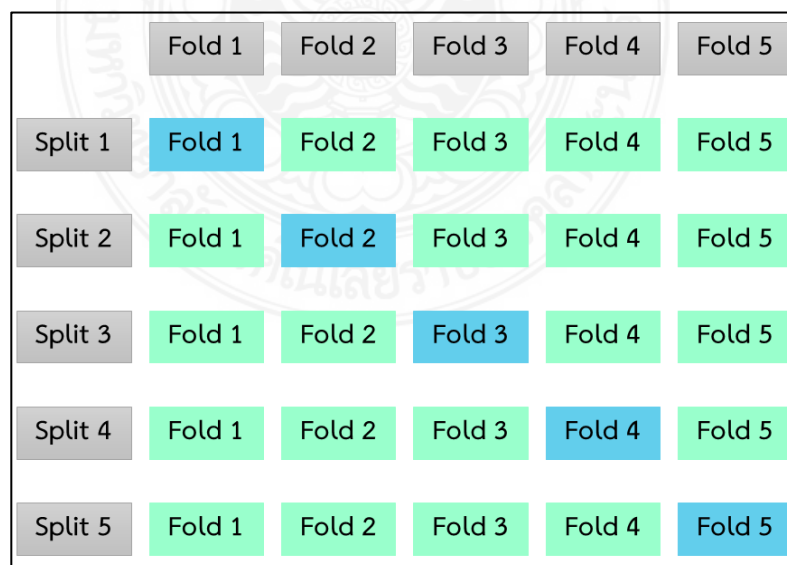
ภาพที่ 2-13 สถาปัตยกรรม (Architecture) ConvNet

ที่มา: กอบเกียรติ สระอุบล (2565)

2.13 การวัดประสิทธิภาพ (K-Fold Cross Validation)

ใช้เพื่อประเมินประสิทธิภาพของโมเดล (Model) และเปรียบเทียบโมเดล (Model) ต่าง ๆ อย่างเป็นกลาง และยังเครื่องมือที่ช่วยให้เราตัดสินใจได้ว่า จะแบ่งข้อมูลส่วนใดของ Training Data ซึ่งการทำ k-fold cv คือการแบ่งข้อมูลเป็น K ส่วนเท่า ๆ กันเพื่อสร้างและทดสอบโมเดล (Train และ Validate) คำนวณค่าเฉลี่ยค่าความถูกต้อง (Accuracy) หรือ Error ก่อนที่จะนำโมเดลไปใช้ทำนายข้อมูลทดสอบ (Test Set) การแบ่งข้อมูลเป็น 5 ชุดข้อมูลเท่า ๆ กัน จะเป็นการแบ่งข้อมูลที่เป็นแบบสุ่ม (Random) เพราะการสุ่ม (Randomness) จะช่วยให้ข้อมูลในแต่ละ fold มีการกระจายตัวใกล้เคียงกัน ช่วยลด bias เวลาที่เราสร้างและทดสอบโมเดล (Model) จำนวน K ที่นิยมใช้กันในทางปฏิบัติมีสองค่าคือ K=5 หรือ K=10 เมื่อแบ่งข้อมูลเสร็จแล้ว K=5 จะสร้าง และทดสอบโมเดล (Model) จนกว่าข้อมูลทุก fold จะถูกนำมาใช้ ถ้า K=5 เราต้องสอน (train) โมเดล ทั้งหมด 5 รอบด้วย (Train Folds) และทดสอบโมเดล (Model) ทั้งหมด 5 รอบด้วย (Validation Fold) ในแต่ละรอบซึ่งควรบันทึกค่า Validation Error ไว้เพื่อนำไปสรุปผลหลังจบกระบวนการ Cross Validation ทั้งหมด เพื่อสามารถคำนวณค่าเฉลี่ยค่าข้อผิดพลาดในการตรวจสอบ (Validation Error) (Kasidis Satangmongkol, 2564) การใช้การทดสอบการตรวจสอบข้าม K-fold กับชุดข้อมูลขนาดใหญ่ต้องใช้เวลาอย่างมาก นอกจากนี้การตรวจสอบความถูกต้องของโมเดลนั้น จะใช้พลังงานคอมพิวเตอร์มากขึ้น แบบจำลองจะใช้เวลาเพิ่มขึ้นในการฝึกสำหรับค่าที่มากขึ้นของ K ถ้า K เป็น 5 แบบจำลองจะผ่านการฝึกฝน 5 ครั้งแยกกันโดยใช้ 5 เท่าที่แตกต่างกันของชุดข้อมูล ซึ่ง K-Fold Cross Validation ดำเนินการจากการสุ่ม (Randomness) ชุดข้อมูลและแยกตัวอย่างย่อยออกเป็นจำนวนที่มีขนาดเท่ากันในการวนซ้ำครั้งแรก ชุดย่อยแรกจะถูกใช้เป็นข้อมูลทดสอบ ในขณะที่ชุดย่อยอื่น ๆ ทั้งหมดถือเป็นข้อมูลการฝึกในการวนซ้ำครั้งถัดไป ให้เลือกเซตย่อยอื่นเป็นชุด

ข้อมูลทดสอบ และทำให้ส่วนอื่น ๆ (รวมถึงชุดทดสอบที่ใช้ในการทำซ้ำครั้งก่อน) เป็นส่วนหนึ่งของข้อมูลการฝึกทำซ้ำ K ครั้งข้างต้นต่อไปแต่ละตัวอย่างย่อยของข้อมูลจะถูกใช้ในแต่ละการวนซ้ำ จนกว่าข้อมูลทั้งหมดจะเสร็จสิ้นลงด้วยผลการประเมินจำนวนหนึ่งซึ่งอัตราข้อผิดพลาดทั้งหมด คือ ค่าเฉลี่ยของคะแนนการประเมินแต่ละรายการทั้งหมด (Praveen Nellihela, 2565) การตรวจสอบความถูกต้อง โดยผู้จัดทำได้ปรับใช้ตามวิจัยของ ไพศาล สิมิลาเตา และจรัญ แสนราช (2562) ซึ่งมีข้อมูลจำนวนที่สามารถใช้ในงานวิจัยได้ 152,850 ชุด ทำการแบ่งข้อมูลด้วยวิธี Cross-validation Test ออกเป็น 5 ส่วน หรือ 5-fold cross-validation ได้ผลออกมา โดยมีค่าความแม่นยำร้อยละ 90.60 ค่าความระลึกร้อยละ 98.39 ค่าความแม่นยำร้อยละ 99.01 และค่าถ่วงดุล 0.321 วิจัยของ พรนภา ชุมเชื้อ (2562) ซึ่งได้ทดสอบเด็กที่มีความบกพร่องทางการเรียนรู้ด้วยเทคนิค 5-fold cross-validation โดยให้ค่าความถูกต้องร้อยละ 98.30 ค่าความคลาดเคลื่อน RMSE = 0.069 และ วิจัยวาทัญญา นิลภาตระกูล และชุตินา เปี้ยวไข่มุก (2562) ได้การทดสอบประสิทธิภาพของตัวแบบใช้ 5-fold Cross-Validation โดยให้ค่าความถูกต้องร้อยละ 91.03 และยังมีวิจัยบางส่วนที่กำหนด K=10 เช่น วิจัยภาภรณ์ เหล่าพิสัย และจรัญ แสนราช (2563) ซึ่งมีข้อมูลจำนวน 14 แอททริบิวต์ และ 10,151 ชุดข้อมูล ทดสอบผลลัพธ์ด้วยวิธีการ 10-Fold Cross Validation มีค่าเฉลี่ยความถูกต้องร้อยละ 94.70 และวิจัย ธรรมชนก นิลมณี และจรัญ แสนราช (2564) ได้ทำการเปรียบเทียบประสิทธิภาพของโมเดลด้วยวิธีการ 10-Fold Cross Validation ได้ค่าความถูกต้องร้อยละ 83.20 ทางผู้จัดทำจึงเลือก K=5 เพราะ ค่าความแม่นยำ ค่าความถูกต้องของวิจัยที่กล่าวมาข้างต้นมีผลลัพธ์ที่ใกล้เคียงกัน และมีระยะเวลาในการ Train ที่สั้นกว่า



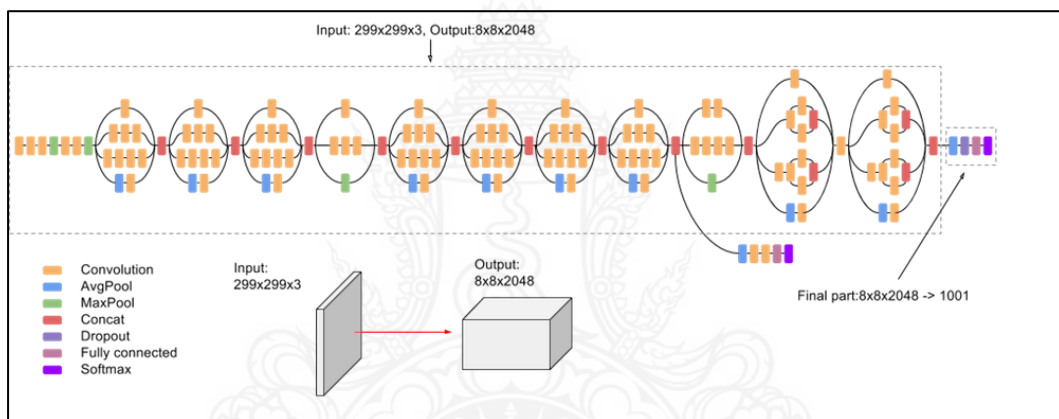
ภาพที่ 2-14 ตัวอย่าง 5-Fold Cross-Validation

ที่มา: Ajitesh Kumar (ออนไลน์)

2.14 InceptionV3

เป็นโครงข่ายประสาทเทียมแบบ Convolutional เพื่อช่วยในการวิเคราะห์ภาพ และการตรวจจับวัตถุ และเริ่มต้นเป็นโมดูลสำหรับ GoogLeNet เป็นรุ่นที่ 3 ของ Inception Convolutional Neural Network ของการออกแบบ InceptionV3 มีวัตถุประสงค์เพื่อให้เครือข่ายลึกขึ้นในขณะเดียวกันก็รักษาจำนวนพารามิเตอร์ไม่ให้เติบโตมากเกินไป

InceptionV3 เป็นโมเดลการเรียนรู้จำภาพที่แสดงให้เห็นว่ามีความแม่นยำมากกว่าร้อยละ 78.1 ในชุดข้อมูล ImageNet แบบจำลองนี้เป็นสุดยอดของแนวคิดมากมายที่พัฒนาโดยนักวิจัยหลายคนในช่วงหลายปีที่ผ่านมา (Rethinking the Inception Architecture for Computer Vision) (Szegedy, et. Al, 2015)



ภาพที่ 2-15 สถาปัตยกรรม (Architecture) InceptionV3

ที่มา: OpenGenus (ออนไลน์)

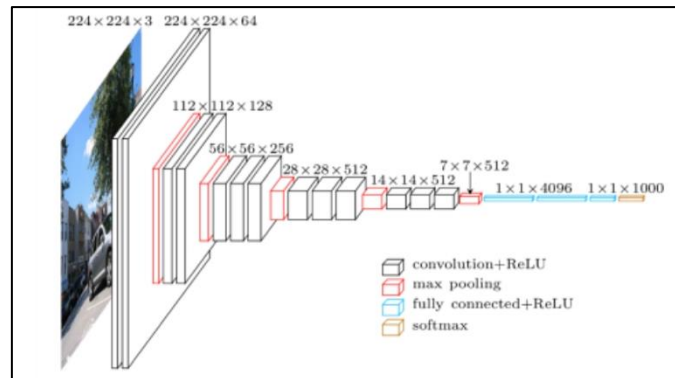
TYPE	PATCH / STRIDE SIZE	INPUT SIZE
Conv	3×3/2	299×299×3
Conv	3×3/1	149×149×32
Conv padded	3×3/1	147×147×32
Pool	3×3/2	147×147×64
Conv	3×3/1	73×73×64
Conv	3×3/2	71×71×80
Conv	3×3/1	35×35×192
3 × Inception	Module 1	35×35×288
5 × Inception	Module 2	17×17×768
2 × Inception	Module 3	8×8×1280
Pool	8 × 8	8 × 8 × 2048
Linear	Logits	1 × 1 × 2048
Softmax	Classifier	1 × 1 × 1000

ภาพที่ 2-16 โครงสร้างโมเดล (Model) ของ InceptionV3
ที่มา: OpenGenus (ออนไลน์)

จากภาพที่ 2-16 โมเดล (Model) InceptionV3 ประกอบด้วย เลเยอร์ 42 เลเยอร์ ซึ่งมีประสิทธิภาพที่สูงกว่า InceptionV1 และ V2 โดยขนาด output ของแต่ละโมดูลคือขนาด input ของโมดูลถัดไป

2.15 VGG16

VGG16 เป็นอัลกอริทึมที่นิยมใช้ในการทำ Image classification โดยวิธีการเรียนรู้แบบมีผู้สอน (Supervised Learning) ซึ่งเป็นการจำแนกข้อมูลประเภทรูปภาพโดยใช้ผลทางสถิติ และ VGG ย่อมาจาก Visual Geometry Group เป็น CNN ที่ปรับปรุงจาก AlexNet นำเสนอโดย K. Simonyan และ A. Zisser-man และชนะการแข่งขัน ILSVRC ปี 2014 โดยการทดสอบได้ค่าแม่นยำสูงสุดถึง 90.1% ซึ่งเป็นการแข่งขันการตรวจจับวัตถุและการจำแนกหมวดหมู่รูปภาพ (VGG คือ Visual Geometry Group อยู่ใน Department of Science and Engineering of Oxford University) ขนาดของ Input VGG16 คือ 224×224 (3 Ch RGB) เช่นกัน แต่ VGG16 ได้ปรับปรุงจาก AlexNet โดยเปลี่ยน Kernel filter ขนาด 11×11 และ 5×5 Stride=4 ในชั้นแรกของ AlexNet มาเป็นขนาด 3×3 Stride=1 (กอบเกียรติ สระอุบล, 2565)



ภาพที่ 2-17 สถาปัตยกรรม (Architecture) VGG16

ที่มา: นายสุริยะ ชยะธรรมกุล (2563)

	Layer	Feature Map	Size	Kernel Size	Stride	Activation
Input	Image	1	224 x 224 x 3	-	-	-
1	2 X Convolution	64	224 x 224 x 64	3x3	1	relu
	Max Pooling	64	112 x 112 x 64	3x3	2	relu
3	2 X Convolution	128	112 x 112 x 128	3x3	1	relu
	Max Pooling	128	56 x 56 x 128	3x3	2	relu
5	2 X Convolution	256	56 x 56 x 256	3x3	1	relu
	Max Pooling	256	28 x 28 x 256	3x3	2	relu
7	3 X Convolution	512	28 x 28 x 512	3x3	1	relu
	Max Pooling	512	14 x 14 x 512	3x3	2	relu
10	3 X Convolution	512	14 x 14 x 512	3x3	1	relu
	Max Pooling	512	7 x 7 x 512	3x3	2	relu
13	FC	-	25088	-	-	relu
14	FC	-	4096	-	-	relu
15	FC	-	4096	-	-	relu
Output	FC	-	1000	-	-	Softmax

ภาพที่ 2-18 โครงสร้างโมเดล (Model) ของ VGG16

ที่มา: Laxman singh tomar (2564)

จากภาพด้านบนเลเยอร์ที่หนึ่งและสอง: อินพุตเป็นภาพ RGB 224x224x3 ซึ่งผ่านเลเยอร์ convolutional ชั้นที่หนึ่งและที่สองพร้อมกับ 64 ฟิวเจอร์แมปหรือ filter ที่มีขนาด 3x3 และการรวมขนาดเดียวกัน โดยมี stride เท่ากับ 1 ขนาดของรูปภาพเปลี่ยนเป็น 224x224x64 จากนั้น VGG16 จะใช้เลเยอร์ร่วมสูงสุดหรือเลเยอร์การสุ่มตัวอย่างด้วยขนาดตัวกรอง 3x3 และระยะของทั้งสองขนาดภาพที่ได้จะลดลงเป็น 112x112x64

ชั้นที่สามและสี่: ถัดไปมีสองเลเยอร์ convolutional พร้อมกับพ็เจอร์ 128 แผนที่มีมีขนาด 3×3 และทำ stide ที่ 1 จากนั้นจะมีเลเยอร์รวมสูงสุดอีกครั้งที่มีขนาดตัวกรอง 3×3 และทำ sride ที่ 2 เลเยอร์นี้เหมือนกับเลเยอร์รวมก่อนหน้านี้ยกเว้นมีพ็เจอร์แม่ปคุณสมบัติ 128 ชุดดังนั้นผลลัพธ์จะลดลงเหลือ $56 \times 56 \times 128$

ชั้นที่ห้าและหก: เลเยอร์ที่ห้าและหกเป็นเลเยอร์ convolutional ที่มีขนาดตัวกรอง 3×3 และระยะทั้งสองใช้แผนที่พ็เจอร์ 256 รายการเลเยอร์ convolutional สองเลเยอร์จะตามด้วยเลเยอร์สูงสุดรวมกับขนาดตัวกรอง 3×3 ทำ stride ของ 2 และมี 256 พ็เจอร์แม่ป

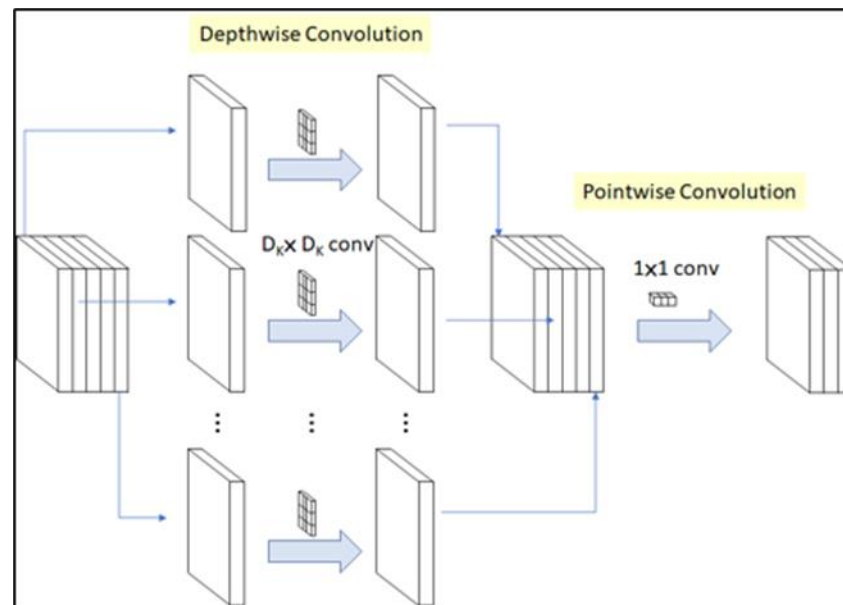
ชั้นที่เจ็ดถึงชั้นสิบสอง: ถัดไปคือชุด convolutional 3 เลเยอร์สองชุดตามด้วยชั้นการรวมสูงสุดเลเยอร์ convolutional ทั้งหมดมีตัวกรอง 512 ขนาด 3×3 และระยะขนาดสุดท้ายจะลดลงเป็น $7 \times 7 \times 512$

ชั้นที่สิบสาม: Output เลเยอร์ Convolutional จะผ่านเลเยอร์ที่เชื่อมต่ออย่างเต็มที่ด้วย feature map 25088 แต่ละขนาด 1×1

เลเยอร์ที่สิบสี่และสิบห้า: ถัดไปเป็นอีกสองเลเยอร์ที่เชื่อมต่ออย่างสมบูรณ์ 4096 หน่วย เลเยอร์เอาต์พุต: ชั้นเอาต์พุต Softmax กับ 1,000 ค่าที่เป็นไปได้ (รัตนโชติ พันธุ์วิไล, 2562)

2.16 MobileNetV2

MobileNet (โดย Google) ออกแบบมาให้มีขนาดเล็กเน้นสำหรับอุปกรณ์ Mobile ตามชื่อ ประมวลผลได้เร็วกว่าเหมาะสำหรับอุปกรณ์ เช่น อุปกรณ์พกพา Mobile terminal และ อุปกรณ์สวมกมลฝังตัว (Embedded device) ตัว Model มีขนาดเพียง 1x MB เท่านั้น ในขณะที่ ResNet มีขนาดถึง 9x MB แต่ประสิทธิภาพของ MobileNet ร้อยละ 70.4 ซึ่งอยู่ในเกณฑ์ที่ใช้ได้ดีพอสมควร Convolution โดยทั่วไปจะทำงานในลักษณะ Channel-wise และ Spatial-wise MobileNet 2 และ 3 (กอบเกียรติ สระอุบล, 2565)



ภาพที่ 2-19 โครงสร้างโมเดล (Model) ของ Mobilenet
ที่มา: กอบเกียรติ สระอุบล (2565)

2.17 Resnet

Resnet (Deep Residual Network หรือ Residual Network) ชนะการแข่งขัน ILSVRC ปี 2015 โดยการทดสอบได้ค่าแม่นยำสูงสุดถึงร้อยละ 90.1 เป็น Neural Network ที่ได้รับความนิยมสำหรับงาน Computer Vision มาก สถาปัตยกรรมรวมจะมีชั้น Convolution จำนวน 5 ชั้น (มีจำนวนพารามิเตอร์ 25,636,712 ซึ่งน้อยกว่า VGG16)

โดยปกติแล้ว Deep Learning Network ที่มีจำนวนชั้นลึกมาก ๆ จะมีปัญหาเรื่อง Vanishing gradient แต่ Resnet ได้นำเสนอวิธีการแก้ปัญหาดังกล่าวโดยการใช้วิธีทางลัดหรือกระโดดข้าม (Shortcut) เครือข่ายบางชั้น เช่น รูป 2-17 จะเห็นว่าการข้ามบางเลเยอร์ (กอบเกียรติ สระอุบล, 2565)

ResNet-50 มีสถาปัตยกรรมตามแบบจำลองที่แสดงด้านบน แต่มีความแตกต่างที่สำคัญประการหนึ่ง ResNet 50 เลเยอร์ใช้การออกแบบคอขวดสำหรับบล็อกส่วนประกอบบล็อกส่วนที่เหลือของคอขวดใช้การบิดเกลียว 1×1 หรือที่เรียกว่า "คอขวด" ซึ่งช่วยลดจำนวนพารามิเตอร์และการคูณเมทริกซ์ สิ่งนี้ทำให้การฝึกแต่ละเลเยอร์เร็วขึ้นมาก ใช้ซ้อนกันสามชั้นแทนที่จะเป็นสองชั้น (datagen.tech, ออนไลน์)

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
		3×3 max pool, stride 2				
conv2_x	56×56	$\begin{bmatrix} 3\times 3, 64 \\ 3\times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times 3, 64 \\ 3\times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1, 64 \\ 3\times 3, 64 \\ 1\times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1, 64 \\ 3\times 3, 64 \\ 1\times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1, 64 \\ 3\times 3, 64 \\ 1\times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3\times 3, 128 \\ 3\times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times 3, 128 \\ 3\times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times 1, 128 \\ 3\times 3, 128 \\ 1\times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times 1, 128 \\ 3\times 3, 128 \\ 1\times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times 1, 128 \\ 3\times 3, 128 \\ 1\times 1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3\times 3, 256 \\ 3\times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times 3, 256 \\ 3\times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times 1, 256 \\ 3\times 3, 256 \\ 1\times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times 1, 256 \\ 3\times 3, 256 \\ 1\times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1\times 1, 256 \\ 3\times 3, 256 \\ 1\times 1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3\times 3, 512 \\ 3\times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times 3, 512 \\ 3\times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1, 512 \\ 3\times 3, 512 \\ 1\times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1, 512 \\ 3\times 3, 512 \\ 1\times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1, 512 \\ 3\times 3, 512 \\ 1\times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

ภาพที่ 2-20 สถาปัตยกรรม (Architecture) Resnet

ที่มา: datagen.tech (ออนไลน์)

สถาปัตยกรรม ResNet 50 เลเยอร์ประกอบด้วยองค์ประกอบต่อไปนี้ ดังแสดงในตารางด้านบน การหมุนเคอร์เนลขนาด 7×7 ควบคู่ไปกับเคอร์เนลอื่น ๆ อีก 64 เคอร์เนลด้วยระยะ 2 ขนาด อีก 9 เลเยอร์ 3×3 64 kernel convolution อีกชั้นมี 1×1 64 kernel และชั้นที่สามมี 1×1 256 kernels 3 ชั้นนี้ทำซ้ำ 3 ครั้ง 12 เลเยอร์ที่มีเคอร์เนล 1×1 128 เคอร์เนล 3×3 128 และเคอร์เนล 1×1 512 ทำซ้ำ 4 ครั้ง 18 เลเยอร์ที่มี 1×1 256 คอร์ และ 2 คอร์ 3×3 256 และ 1×1 1024 วนซ้ำ 6 ครั้ง 9 เลเยอร์ที่มี 1×1 512 คอร์ 3×3 512 คอร์ และ 1×1 2048 คอร์ วนซ้ำ 3 ครั้ง (datagen.tech, ออนไลน์)

2.18 งานวิจัยที่เกี่ยวข้อง

ณัฐวดี หงส์บุญมี และพิรยุทธ ขุนบุญ (2564) ได้ศึกษาเรื่อง การพัฒนาแอปพลิเคชันทำนายระดับความหวานของส้มโอ ผ่านสมาร์ตโฟนด้วยเทคนิคการเรียนรู้เชิงลึก ผลการศึกษาวิจัยพบว่า เริ่มจากการรวบรวมข้อมูลกลุ่มตัวอย่างภาพส้มโอจำนวน 4 ประเภท ได้แก่ พันธุ์ขาวแตงกวาหวาน พันธุ์ขาวแตงกวาไม่หวาน พันธุ์ท่าช้อยหวาน และพันธุ์ท่าช้อยไม่หวาน จำนวนภาพทั้งหมด 600 ภาพ นำภาพมาวิเคราะห์และสร้างโมเดลจำแนกภาพโดยใช้โครงข่ายประสาทเทียมเชิงลึกแบบคอนโวลูชัน ผลการศึกษาพบว่า โมเดลจากโครงข่ายประสาทเทียมคอนโวลูชันมีประสิทธิภาพสูงสามารถจำแนกระดับความหวานส้มโอได้อย่างแม่นยำโดยได้ผลลัพธ์ที่ถูกต้องคิดเป็นความแม่นยำเฉลี่ยเท่ากับ 97 เปอร์เซ็นต์ จึงนำโมเดลดังกล่าวไปพัฒนาในส่วนการติดต่อผู้ใช้ในรูปแบบแอปพลิเคชันบนสมาร์ตโฟนโดยแอปพลิเคชันจะทำการเรียกใช้โมเดลผ่านชุดคำสั่งเพื่อวิเคราะห์ระดับความหวานจากภาพถ่ายของส้มโอการทดสอบประสิทธิภาพการทำงาน

ของแอปพลิเคชัน พบว่า แอปพลิเคชันสามารถจำแนกภาพและวัดระดับความหวานส้มโอ ได้จริงโดยมีค่าความถูกต้องเฉลี่ยถึง 81.25 เปอร์เซ็นต์ และผลการประเมินความพึงพอใจของระบบโดยผู้ใช้งานอยู่ในระดับพึงพอใจระดับดี ค่าเฉลี่ยเท่ากับ 4.19 ผลการทดลองแสดงให้เห็นว่าแอปพลิเคชันนี้ สามารถนำไปใช้ประโยชน์ในการช่วยทำนายความหวานของส้มโอได้ช่วยเพิ่มความสะดวกรวดเร็วต่อการใช้งานอีกทั้งเป็นการส่งเสริมการขาย

Ling Yi Jun and Lau Phooi Yee (2563) ได้ศึกษาเรื่อง Comparison of Vision-based and CNN based Classifiers for Fish Monitoring in Complex Environment ผลการศึกษาวิจัยพบว่า การเพาะเลี้ยงสัตว์น้ำสามารถช่วยลดผลกระทบต่อสิ่งแวดล้อมจากการเสียหายจากความต้องการอาหารทะเลที่ครบถ้วนด้วยฟาร์มเลี้ยงสัตว์ การดูแลรักษาฟาร์มขนาดใหญ่เป็นเรื่องที่ค่อนข้างยาก โดยทางผู้จัดทำได้ใช้กล้องใต้น้ำติดอยู่ในกรงของฟาร์ม จึงสามารถนำระบบที่ใช้การมองเห็นมาใช้ในการแก้ไขข้อมูลที่เป็นประโยชน์จากภาพวิดีโอได้โดยอัตโนมัติ งานนี้เสนอให้แก้ปัญหาดังกล่าวโดยใช้ 2 วิธีในการตรวจจับ 1) Extended UTAR Aquaculture Farm Fish Monitoring System Framework (UFFMS) วิธีที่ทำด้วยมือ 2) Faster Region Convolutional Neural Network (R-CNN ที่เร็วขึ้น) ซึ่งเป็นวิธีการที่ใช้ CNN สองวิธีนี้ดึงข้อมูลมาจากวิดีโอ ผลการทดลองแสดงให้เห็นว่าสำหรับฟุตเทจที่มีแสงสว่างเพียงพอ R-CNN ที่เร็วกว่าทำงานได้ดีกว่า UFFMS แบบขยาย อย่างไรก็ตาม ความแม่นยำของ R-CNN ที่เร็วกว่านั้นลดลงอย่างมากสำหรับฟุตเทจที่มีแสงน้อย โดยเฉลี่ยร้อยละ 28.57 แม้ว่าจะมีคะแนนความแม่นยำที่สมบูรณ์แบบอยู่ที่ตามความแม่นยำโดยเฉลี่ยของวิธี Extended-UFFMS และ Faster R-CNN คือ ร้อยละ 57.89 และร้อยละ 71.77 ตามลำดับ

อุมาภรณ์ สายแสงจันทร์ รพีพร ชำของ และอรรถพล สุวรรณษา (2565) ได้ศึกษาเรื่อง การเปรียบเทียบวิธีการวิเคราะห์โรคใบมะนาวด้วยการเรียนรู้เชิงลึก งานวิจัยนี้ได้ศึกษาวิธีการวิเคราะห์ความเป็นโรคจากใบมะนาวโดยอาศัยการเรียนรู้เชิงลึกโดยเทคนิคที่ใช้คือการจำแนกใบมะนาวโดยใช้คอนโวลูชันนิรอลเน็ตเวิร์ก ซึ่งสถาปัตยกรรมที่ศึกษาในที่นี้ได้แก่ LeNet-5 VGG16 RestNet-50 และสถาปัตยกรรมที่นำเสนอบนพื้นฐานของ VGGNet โดยการปรับปรุง Hyperparameter การปรับปรุงสถาปัตยกรรมนี้เป็นการลดขนาดของชั้นการเรียนรู้ และช่วยลดจำนวนพารามิเตอร์ของโครงข่ายซึ่งจะทำให้การประมวลผลเร็วขึ้นแต่ประสิทธิภาพยังคงเดิมจากการทดลองเก็บรวบรวมข้อมูลภาพถ่ายใบมะนาวเดี่ยวจำนวน 5,710 ภาพ เป็นภาพสี RGB โดยแบ่ง เป็นใบมะนาวที่เป็นโรคและไม่เป็นโรคเท่ากัน และทำการแบ่งชุดข้อมูลเป็นชุดสอนร้อยละ 80 และชุดทดสอบร้อยละ 20 พบว่าประสิทธิภาพการวิเคราะห์ ด้วย LeNet-5 มีความถูกต้องร้อยละ 78.90 ซึ่งมีประสิทธิภาพต่ำสุด ในขณะที่สถาปัตยกรรมที่นำเสนอมีความถูกต้องร้อยละ 89.06 ซึ่งมีค่าสูงสุดแต่ไม่แตกต่างกันกับ ResNet-50 มากนัก

สุริยะ ชยะธรรมกุล (2563) ได้ศึกษาเรื่อง การจำแนกผลึกน้ำตาลด้วยวิธีการเรียนรู้เชิงลึก ผลการศึกษาวิจัย พบว่า งานวิจัยนี้นำวิธีการเรียนรู้เชิงลึกมาจำแนกภาพการตกผลึกน้ำตาลจากกระบวนการน้ำตาลเคี้ยวตกผลึก โดยได้ทำการทดลองจริงที่โรงงานน้ำตาลในประเทศไทย โดยทำการสร้างโมเดลที่แตกต่างกัน และนำมาฝึกฝนผ่านชุดข้อมูลภาพผลึกน้ำตาล จากกระบวนการผลิตนี้แล้วทำการวัด และเปรียบเทียบประสิทธิภาพ โดยโมเดล VGG16 ที่ผ่านการปรับแต่ง มีความแม่นยำในการจำแนกผลึกน้ำตาลระหว่างร้อยละ 80 ถึงร้อยละ 94 ของภาพผลึกน้ำตาลทั้ง 4 ประเภท ที่แยกตามสภาวะการเกิดผลึก ผลการทดลองนี้ยังแสดงให้เห็นว่าโมเดลนี้มีความแม่นยำมากกว่าโมเดลอื่น และสามารถใช้ตรวจสอบการตกผลึกของกระบวนการผลิตน้ำตาล อีกทั้งระบบควบคุมแบบปัญญาประดิษฐ์โดยอาศัยการถ่ายทอดความชำนาญจากมนุษย์

นพรุจ พัฒนสาร และณัฐวุฒิ ศรีวิบูลย์ (2563) ได้ศึกษาเรื่อง การประมวลผลภาพสำหรับการจำแนกคุณภาพมะม่วงพันธุ์โชคอนันต์ โดยการจำลองการมองเห็นของมนุษย์ด้วยวิธีการเรียนรู้เชิงลึก ผลการศึกษาวิจัย พบว่า ดำเนินการเก็บรวบรวมข้อมูล ภาพถ่ายมะม่วงพันธุ์โชคอนันต์ และเก็บข้อมูลการจำแนกคุณภาพกำหนดระดับคุณภาพออกเป็น 4 ระดับประกอบด้วย คุณภาพระดับเกรด A คุณภาพระดับเกรด B คุณภาพระดับเกรด C และคุณภาพระดับเกรด D คือ มะม่วงเน่า ผลของการวิจัยแสดงให้เห็นว่าการใช้วิธีการเรียนรู้เชิงลึกด้วยอัลกอริทึมโครงข่ายประสาทเทียมแบบสังวัตนาการ (Convolutional Neural Networks: CNN) ในการประมวลผลภาพเพื่อสร้างแบบจำลองแสดงค่าความแม่นยำสูงสุด คือร้อยละ 99.79 จากนั้นนำแบบจำลองพัฒนาเป็นระบบต้นแบบสำหรับจำแนกคุณภาพมะม่วงพันธุ์โชคอนันต์ พบว่าอัตราความสำเร็จในการจำแนกคือร้อยละ 100

นศพร ธรรมโชติ ชัยสิทธิ์ ปรีชา และปิติพัฒน์ บุตรโคตร (2563) ได้ศึกษาเรื่อง การจำแนกโรคพืชที่ปรากฏทางใบมะนาวด้วยวิธีการประมวลผลภาพบนสมาร์ตโฟนระบบปฏิบัติการแอนดรอยด์ ผลการศึกษาวิจัย พบว่า โดยมีการดำเนินงานเริ่มจาก 1) เก็บข้อมูลภาพโรคพืชที่ปรากฏทางใบมะนาวในพื้นที่เป้าหมายจังหวัดนครศรีธรรมราช จำแนกภาพตามกลุ่มโรคโดยผู้เชี่ยวชาญทางด้านโรคพืช 2) พัฒนาอัลกอริทึมจากชุดคำสั่งของ TensorFlow Framework ซึ่งเป็น open source library ของ Google โดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน โครงสร้าง Inception-v3 โดยกำหนดจำนวนภาพสำหรับการฝึกต่อการทดสอบเป็น 80:20 โดยทำการฝึกจำนวน 120 รอบ และกำหนดเอพ็อด เท่ากับจำนวนโรคเท่ากับ 6 โรค 3) สร้างแอปพลิเคชันสมาร์ตโฟนในระบบปฏิบัติการแอนดรอยด์ โดยใช้ Android Studio software สำหรับการใช้งาน ซึ่งจะเก็บไฟล์นามสกุล .apk 4) ทดสอบแอปพลิเคชันบนสมาร์ตโฟนเพื่อหาค่าความแม่นยำและความผิดพลาดในการจำแนกจากตัวอย่าง ภาพใหม่ 50 ภาพต่อโรค ผลลัพธ์การทดสอบใช้งานแอปพลิเคชันที่สร้างขึ้นมีความถูกต้องมากกว่า 90 เปอร์เซ็นต์ทุกโรค ซึ่งจะเห็นได้

ว่าอัลกอริทึมที่พัฒนาขึ้นง่ายต่อการปรับปรุงโดย เพิ่มจำนวนภาพ หรือโรคที่ค้นพบใหม่ แอปพลิเคชันนี้จะเป็นประโยชน์แก่เกษตรกรผู้ปลูกมะนาว

รัตน์โชติ พันธุ์วิไล (2562) ได้ศึกษาเรื่อง การตรวจหาต้นไม้ที่เป็นโรคโดยอัตโนมัติด้วย ภาพถ่ายมุมสูงจากโดรน และวิธีการเรียนรู้เชิงลึก ผลการศึกษาวิจัย พบว่า งานวิจัยฉบับนี้ได้นำเทคนิคทางคอมพิวเตอร์วิทัศน์มาช่วยตรวจจับต้นไม้ที่เป็นโรคโดยอัตโนมัติโดยประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึก (Deep learning) ซึ่งใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันกับภาพที่ได้จากโดรนที่ถ่ายจากมุมสูงภายในสวนของเกษตรกร โดยเสนอโมเดลแบบลำดับชั้น (Cascade Model) ในการตรวจจับต้นไม้ที่เป็นโรคใน 2 ขั้นตอน ขั้นตอนแรกทำการตรวจจับตำแหน่งต้นไม้ในรูปภาพ (Object detection) โดยใช้เทคนิค Mask R-CNN ขั้นตอนที่สองทำการจำแนกต้นไม้ที่เป็นโรคจากต้นไม้ที่ตรวจจับได้ในขั้นตอนแรก (Classification) โดยการใช้เทคนิคโอนถ่ายการเรียนรู้จากฐานข้อมูล ImageNet ของโมเดล VGG16 (Transfer Learning) ร่วมกับการใช้ฟังก์ชัน Focal Loss ซึ่งให้ผลดีกับข้อมูลแบบไม่สมดุล ผลการทดลองจากภาพถ่ายมุมสูงจากสวนทุเรียนตัวอย่างแสดงให้เห็นว่าโมเดลแบบลำดับชั้นที่นำเสนอให้ความแม่นยำที่สูง และครบถ้วนกว่าการใช้โมเดลในการตรวจจับต้นไม้ที่เป็นโรคเพียงอย่างเดียว รวมถึงลดความแตกต่างของค่าความน่าจะเป็นในกรณีที่มีการทำนายผิด ซึ่งทำให้สามารถนำไปแจ้งเตือนให้เกษตรกรตรวจสอบต้นไม้นี้ได้อย่างละเอียดได้ในภายหลัง

พิชิตชัย พิมพ์โคตร (2564) ได้ศึกษาเรื่อง การตรวจจับโลโก้สินค้าแอลกอฮอล์ด้วยแบบจำลองการเรียนรู้เชิงลึก ผลการศึกษาวิจัย พบว่า งานวิจัยนี้แนะนำขั้นตอนวิธีการเลือกใช้และเตรียมข้อมูลโดยเลือกยี่ห้อสินค้าที่ได้รับความนิยมในประเทศไทยเป็นโลโก้สุดท้ายที่แต่ละห้อเลือกใช้ในปี 2362 จำนวน 6 ยี่ห้อ ได้แก่ เบียร์สิงห์ เบียร์ลีโอ เบียร์ไฮเนเกน เป๊ปซี่ โคคาโคล่า สตาร์บัค โดยใช้ชุดข้อมูลที่ผู้วิจัยเก็บรวบรวมมาใช้ร่วมกับชุดข้อมูล Logos-32Pius และการเพิ่มความหลากหลายให้กับชุดข้อมูลฝึกสอน (Data Augmentation) ให้เหมาะสม รวมไปถึงการเลือกใช้โมเดล Mask R-CNN และการกำหนดค่าโมเดลที่เหมาะสมทำให้ได้ประสิทธิภาพมากที่สุดที่ mAP 0.982 และยังมีประสิทธิภาพที่ดีเมื่อทดสอบกับชุดข้อมูลที่มีคุณลักษณะที่หลากหลาย ได้แก่ ขนาดรูปภาพเล็ก และมีความละเอียดต่ำ รูปภาพที่มีดี แสงสว่างมากกว่าปกติภาพไม่คมชัดภาพมีสิ่งรบกวนรวมไปถึงรูปภาพที่โลโก้ที่ไม่สมบูรณ์ด้วย

ณัฐวดี หงษ์บุญมี และณัฐพงศ์ จันตะวงศ์ (2563) ได้ศึกษาเรื่อง การประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึกเพื่อวัดระดับความหวานของแตงโมผ่านสมาร์ตโฟนโดยประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึกสำหรับใช้งานบนสมาร์ตโฟนเพื่ออำนวยความสะดวกสำหรับบุคคลทั่วไปที่ต้องการทราบชื่อพันธุ์ และความหวานแตงโมส่วนประกอบหลักของระบบประกอบด้วย 1. การสร้างโมเดลจำแนกพันธุ์ และระดับความหวานของแตงโมด้วยโครงข่ายประสาทเทียมการเรียนรู้เชิงลึก

ผ่านไลบรารี TensorFlow โดยนำอัลกอริทึม Inceptionv3 และ MobileNet มาทำการทดลองเปรียบเทียบการจำแนกภาพ ซึ่งฝึกสอนให้สามารถจำแนกภาพจำนวน 4 ประเภท ประเภทละ 100 ภาพ ฝึกสอนจำนวน 500 รอบ ผลการทดลองพบว่าโมเดลจากอัลกอริทึม InceptionV3 มีค่าความถูกต้องที่เท่ากับโมเดลจากอัลกอริทึม MobileNet ค่าความถูกต้องเท่ากับร้อยละ 97.20 แต่จากการพิจารณาขนาดของโมเดลที่ได้ จากการเรียนรู้พบว่า MobileNet มีขนาดของโมเดลเล็กกว่า Inceptionv3 ดังนั้นจึงเลือกโมเดลจาก MobileNet ไปพัฒนาระบบต่อไป 2. การนำโมเดลจากอัลกอริทึม MobileNet ไปพัฒนาเป็นแอปพลิเคชันบนสมาร์ตโฟน ดำเนินการพัฒนาด้วยโปรแกรม Android Studio ผลการทดสอบความพึงพอใจการใช้แอปพลิเคชันจากผู้ใช้งานพบว่าความพึงพอใจเฉลี่ยเท่ากับ 4.34 ส่วนเบี่ยงเบนมาตรฐาน 0.62 อยู่ในเกณฑ์ระดับดีสามารถสรุปได้ว่าแอปพลิเคชันนี้มีประสิทธิภาพสามารถนำไปใช้งานได้จริง



บทที่ 3

วิธีดำเนินงานวิจัย

การวิจัยเรื่อง ระบบตรวจจับและระบุประเภทไม้ประดับบอนสีด้วยการประมวลผลภาพ โดยมีวัตถุประสงค์ เพื่อศึกษาข้อมูลของไม้ประดับบอนสีประเภทต่างๆ เป็นการจำแนกกลุ่มประเภทใบ โดยการใช้เทคนิคการประมวลผลภาพในการวิเคราะห์ ผู้วิจัยได้ดำเนินการวิจัยตามขั้นตอน ดังนี้

3.1 แบบแผนการวิจัย

การวิจัยครั้งนี้ผู้วิจัยได้ใช้แบบแผนการวิจัยเป็นการวิจัยเชิงปฏิบัติการ

3.2 เครื่องมือที่ใช้ในการวิจัย

เครื่องมือที่ใช้ในการวิจัยครั้งนี้ประกอบด้วย

- ฮาร์ดแวร์ (Hardware) ได้แก่
 1. สมาร์ทโฟน Apple รุ่น iPhone XR 1 เครื่อง
 2. เครื่องคอมพิวเตอร์ (Computer) 1 เครื่อง
Processor Intel® Core(TM) i3-1005G1
CPU @1.20GHz 1.19 GHz Ram 8.00 GB System type 64-bit
- ซอฟต์แวร์ (Software) ได้แก่
 1. ภาษา Python
 2. Google Colaboratory

3.3 การเก็บรวบรวมข้อมูล

ทำการเก็บรวบรวมข้อมูลตัวอย่างของประเภทไม้ประดับบอนสี จำนวน 4 ประเภท ได้แก่ ใบไทย ใบยาว ใบกลม และใบไผ่ โดยใช้การถ่ายภาพผ่านกล้องสมาร์ทโฟน Apple รุ่น iPhone XR โดยมีจำนวนของภาพถ่ายบอนสีทั้งหมด (Data set) 1,500 ภาพ แบ่งเป็นชุด Training set จำนวน 1,200 ภาพ จำนวนประเภทละ 300 ภาพ และชุด Test set จำนวน 300 ภาพ จำนวนประเภทละ 75 ภาพ โดยการนำรูปที่ถ่ายจากต้นบอนสีนั้นมาคัดแยกตามประเภทของลักษณะใบแบ่งออกมาได้ทั้งหมด 4 ประเภท โดยแบ่งตามลักษณะของใบบอนสี คือ ใบไผ่ (Bamboo Leaves) ใบกลม (Round Leaves) ใบไทย (Thai Leaves) และ ใบยาว (Long Leaves) ดังแสดงในตารางที่ 3-1

Class	No. Images			
	Training set		Testing set	รวม
	Training	K-Fold Cross Validation		
ใบไผ่ (Bamboo Leaves)	240	60	75	375
ใบกลม (Round Leaves)	240	60	75	375
ใบไทย (Thai Leaves)	240	60	75	375
ใบยาว (Long Leaves)	240	60	75	375
รวม	960	240	300	1,500

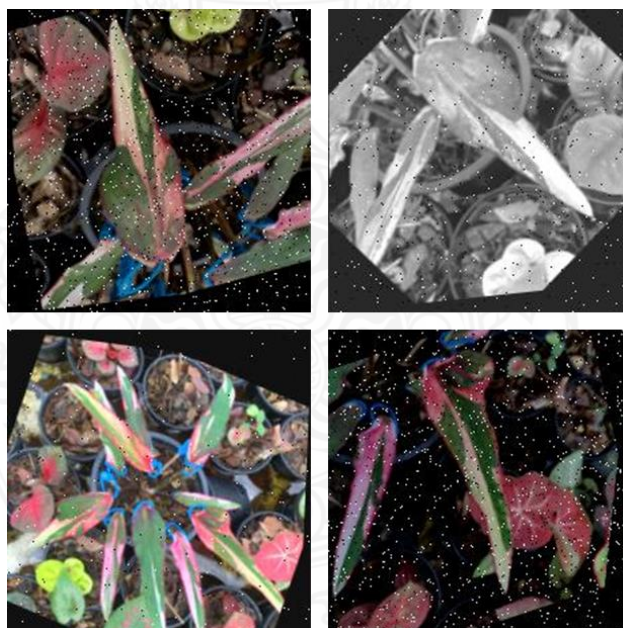
ตารางที่ 3-1 ข้อมูลหมวดหมู่ (Class) ของประเภทบอนสี

จากตารางที่ 3-1 แสดงถึงข้อมูลหมวดหมู่ (Class) ของประเภทบอนสีที่ทำการเก็บข้อมูล โดยได้ทำการเก็บข้อมูลทั้งหมด 4 Class แบ่งเป็น ใบไผ่ (Bamboo Leaves) ใบกลม (Round Leaves) ใบไทย (Thai Leaves) และ ใบยาว (Long Leaves) ซึ่งเก็บเป็น Data set ทั้งหมด 1,500 ภาพ แบ่งออกเป็น Training set ทั้งหมด 1,200 ภาพ แยกเป็น K-Fold Cross Validation จำนวน 240 ภาพ Training จำนวน 960 ภาพ และเป็น Test set ทั้งหมด 300 ภาพ

3.4.1 การเตรียมข้อมูล (Data Preparation and Image augmentation)

เพื่อเพิ่มประสิทธิภาพในชุด Training set จึงได้นำภาพถ่ายมาปรับเพิ่มประสิทธิภาพ โดยใช้ผ่าน Web Application ที่ชื่อว่า roboflow.com ซึ่งเป็นการทำ Data Augmentation ซึ่งได้ทำการกำหนดค่าการดั่งแสดงในภาพที่ 3-1

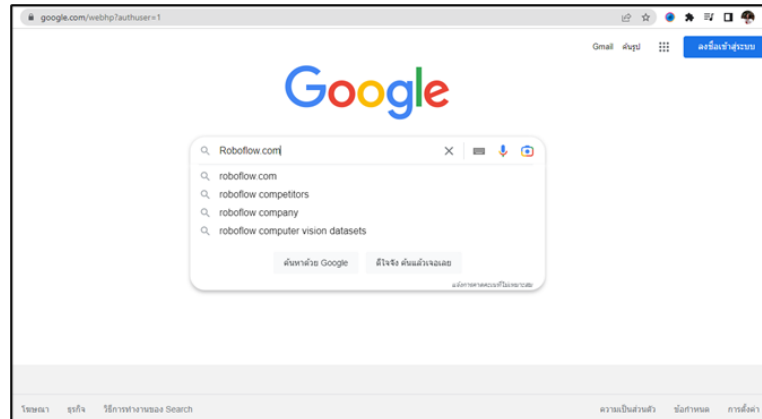
- Flip: Horizontal, Vertical
- Rotation: Between -45° and $+45^{\circ}$
- Shear: $\pm 20^{\circ}$ Horizontal, $\pm 20^{\circ}$ Vertical
- Grayscale: Apply to 35% of images
- Saturation: Between -50% and +50%
- Brightness: Between -35% and +35%
- Blur: Up to 1px
- Noise: Up to 5% of pixels



ภาพที่ 3-1 แสดงตัวอย่าง Data Augmentation

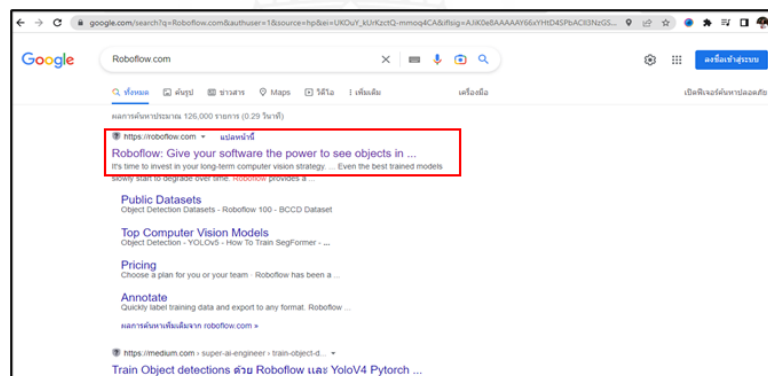
วิธีการเตรียมข้อมูล Data Preparation and Image Augmentation มีดังนี้

1. ค้นหา Roboflow ผ่าน Google Chrome



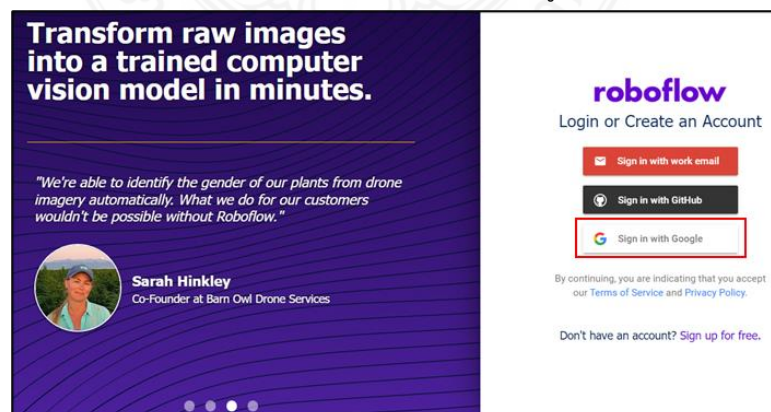
ภาพที่ 3-2 ค้นหา Roboflow ผ่าน Google Chrome

2. เลือกเว็บแอปพลิเคชัน Roboflow



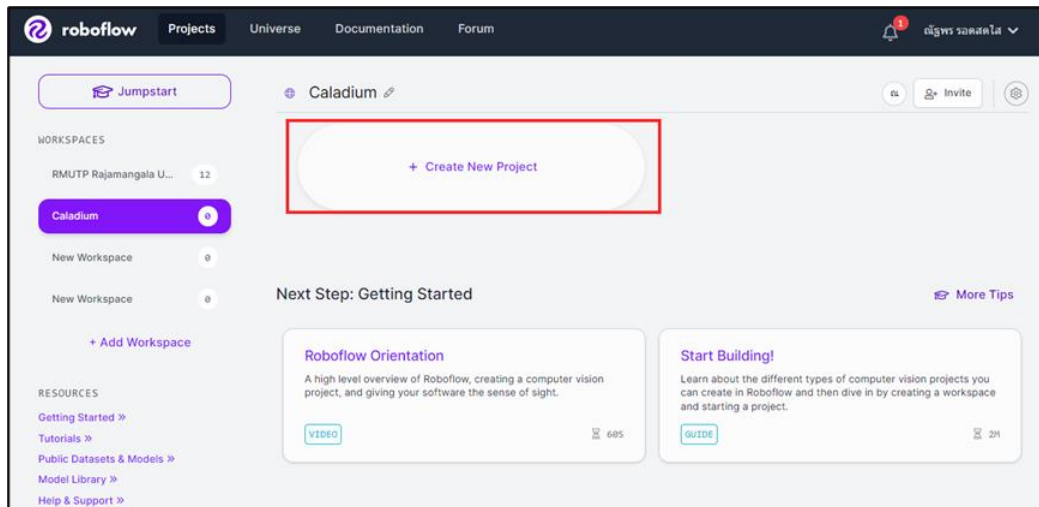
ภาพที่ 3-3 เลือกเว็บแอปพลิเคชัน Roboflow

3. เมื่อทำการคลิกเข้ามาจะเจอหน้าต่าง Sign in การใช้งานสามารถ Sign in ผ่าน Email GitHub หรือ Sign in with Google (ในที่นี้คณะผู้จัดทำจะ Sign in with Google)



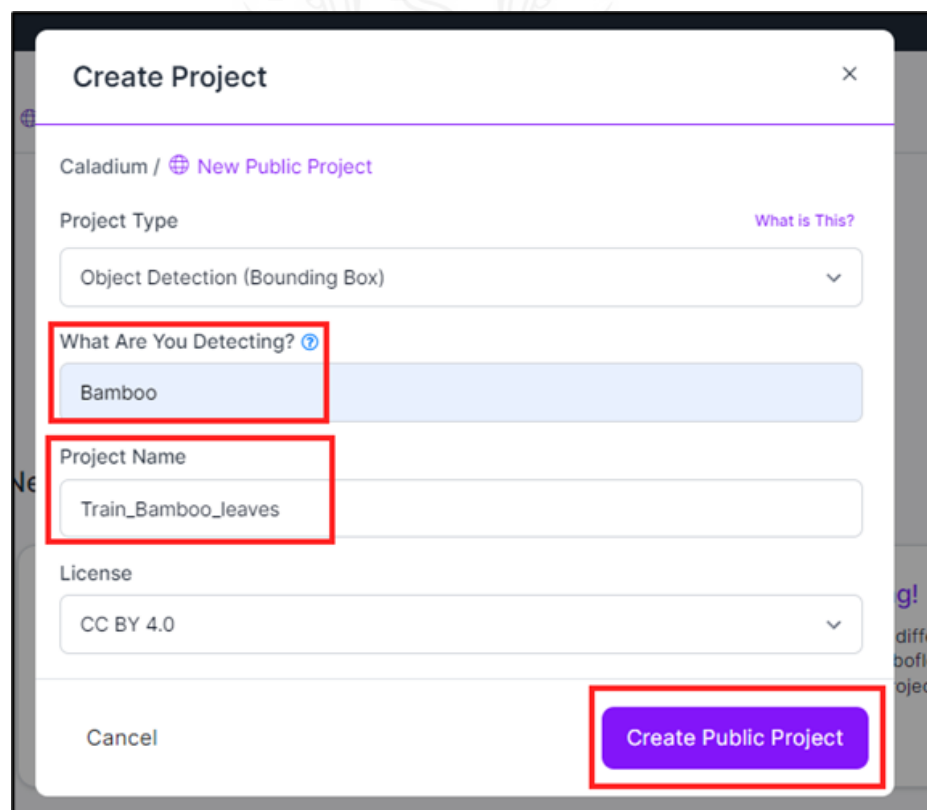
ภาพที่ 3-4 เลือก Sign in การใช้งานสามารถ Sign in ผ่าน Email

4. เมื่อ Login แล้ว ให้ทำการสร้าง Project ขึ้นมาใหม่



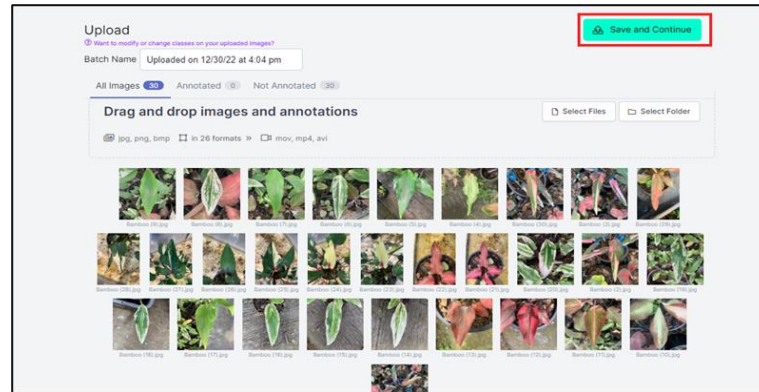
ภาพที่ 3-5 ทำการสร้าง Project ขึ้นมา

5. แล้วทำการใส่ชื่อ Project Name และ สิ่งที่ต้องการให้ตรวจจับ ในที่นี้คณะผู้จัดทำได้ใส่สิ่งที่ต้องการตรวจจับ คือ ลักษณะของประเภทใบ ได้แก่ Bamboo และ Project Name คือ Train_Bamboo_leaves จากนั้นทำการ Create Project



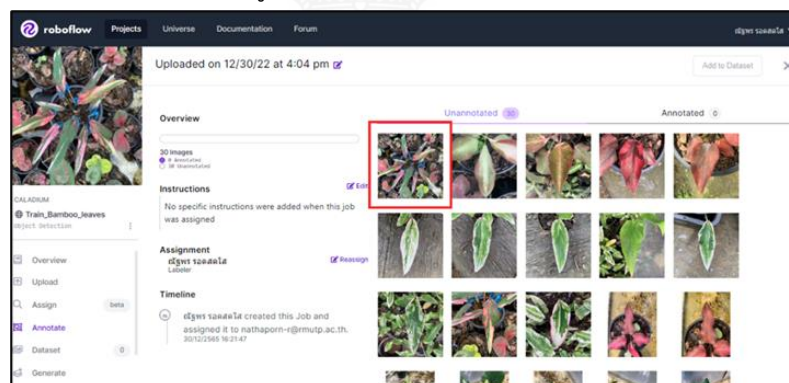
ภาพที่ 3-6 ทำการ Create Project

6. จากนั้นจะเป็นการนำไฟล์รูปภาพที่ต้องการนำมาใส่ เมื่อเลือกภาพที่ต้องการได้แล้วให้คลิก Save and Continue



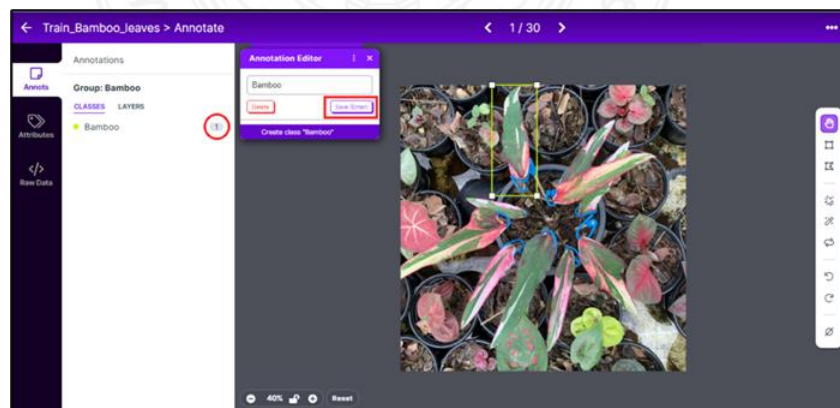
ภาพที่ 3-7 นำรูปภาพที่ต้องการ Augmentation มาใส่

7. จากนั้นให้ทำการเลือกรูปภาพตามกรอบสีแดง



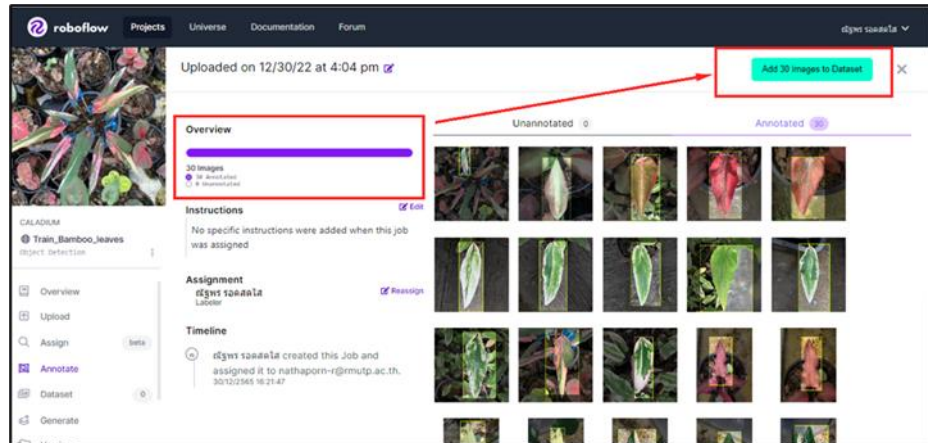
ภาพที่ 3-8 ทำการเลือกรูปภาพ Unannotated

8. ให้ลากเมาส์ครอบรูปใบที่เราต้องการเมื่อทำการครอบแล้วจะมีกรอบสีเขียวขึ้นตามภาพ แล้วทำการคลิก Enter ให้ทำให้ครบทุกภาพ



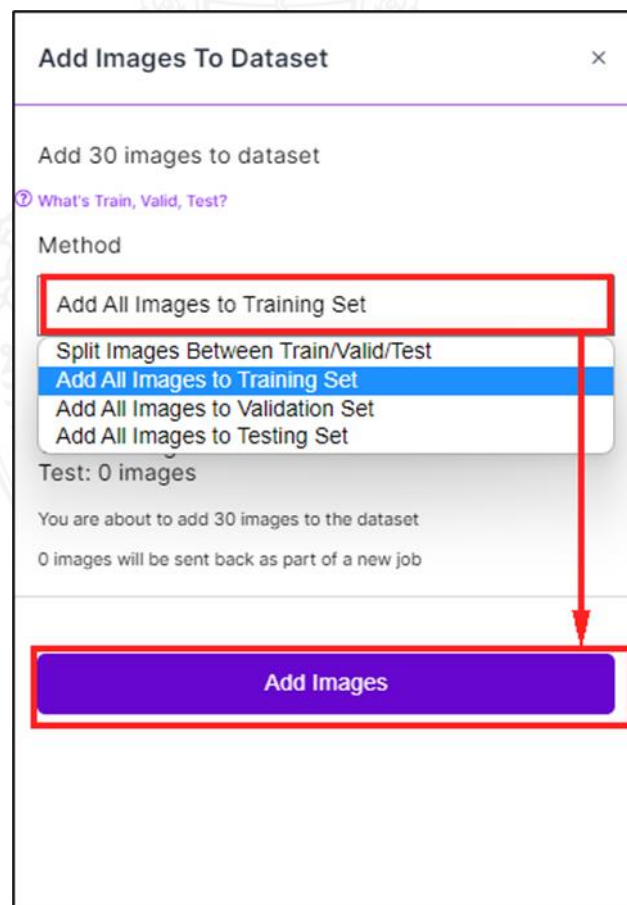
ภาพที่ 3-9 สร้าง Group ของภาพ

9. เมื่อทำครบทุกภาพแล้วตรง Overview จะแสดง 30 Annotated (30 คือจำนวนภาพที่นำเข้ามา) จากนั้น คลิก Add Images to Data set ตามลูกศร



ภาพที่ 3-10 แสดง Overview จำนวนภาพที่ Annotated แล้ว

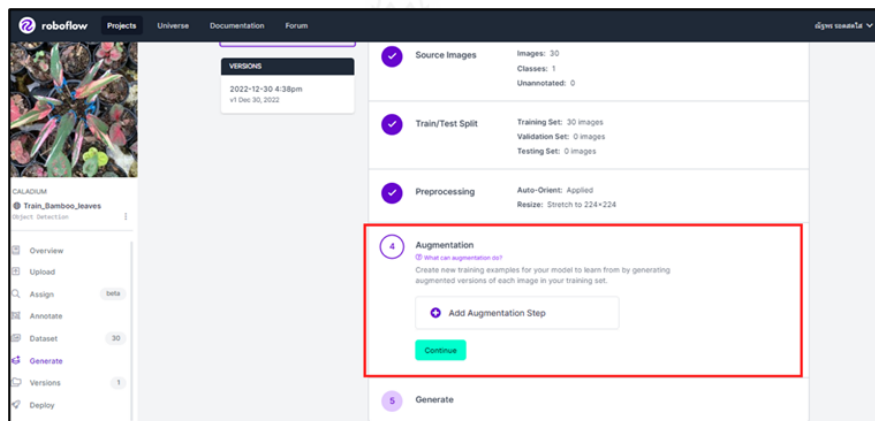
10. จากนั้นทำการเลือก Method เป็น Add all Images to Training Set เมื่อเลือกแล้วทำการคลิก Add Images



ภาพที่ 3-11 เลือก Add image to data set

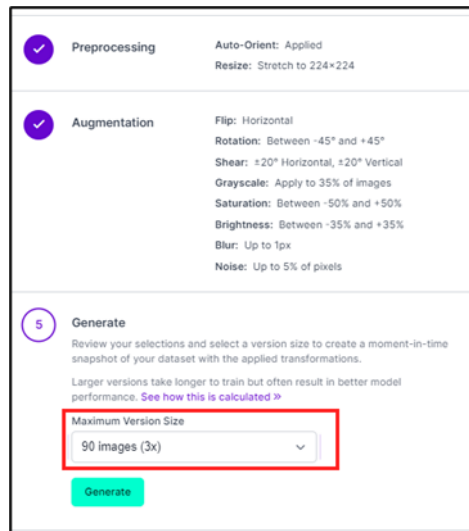
11. จากนั้นทำการเพิ่มปริมาณ Training data โดยทำการ Add Augmentation ให้ทำการ กำหนดค่าการดังนี้

- Flip: Horizontal, Vertical
- Rotation: Between -45° and $+45^\circ$
- Shear: $\pm 20^\circ$ Horizontal, $\pm 20^\circ$ Vertical
- Grayscale: Apply to 35% of images
- Saturation: Between -50% and +50%
- Brightness: Between -35% and +35%
- Blur: Up to 1px
- Noise: Up to 5% of pixels



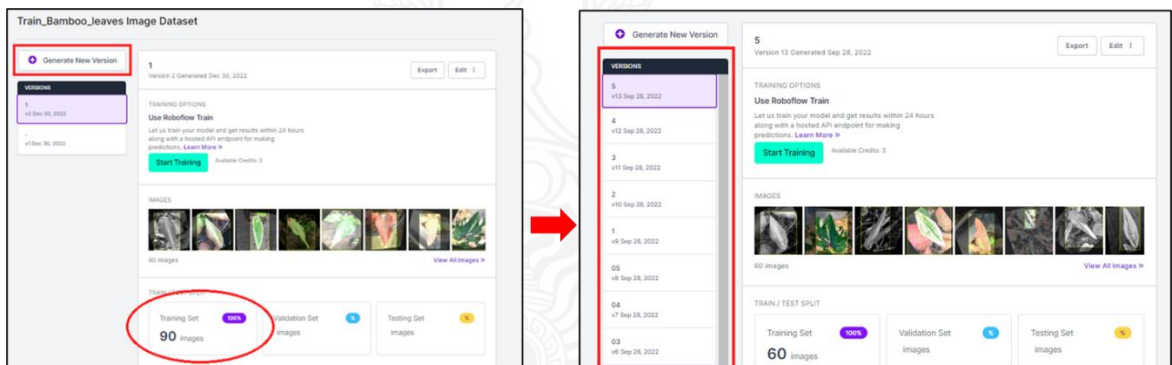
ภาพที่ 3-12 กำหนดค่า Augmentation

12. เมื่อกำหนดค่า Augmentation เรียบร้อย ก็ทำการ Generate รูปภาพเพิ่ม (ในการนี้ที่ใช้เพื่อการศึกษาสามารถใช้ได้สูงสุด 90 ภาพ)



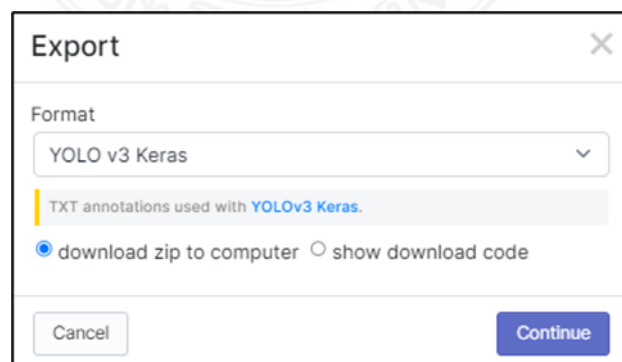
ภาพที่ 3-13 แสดงการ Generate รูปภาพเพิ่ม

13. เมื่อทำการ Generate จะแสดงผลดังนี้ จะเห็นว่าตอนนี้มีภาพในการ Training set ทั้งหมด 90 ภาพ หากต้องการให้ได้ภาพที่มากขึ้นให้ทำการคลิก Generate New Version (ทำเหมือนขั้นตอนที่ 11) เพื่อทำการเพิ่มรูปภาพให้เพียงพอต่อการ Train



ภาพที่ 3-14 แสดงการ Generate ภาพเพิ่มให้เพียงพอต่อการ Train

14. จากนั้นให้ทำการกด Export เพื่อบันทึกภาพ โดย Format ให้เลือกดังภาพ และคลิก Continue เป็นการเสร็จสิ้น



ภาพที่ 3-15 แสดงการ Export เพื่อบันทึกภาพ

15. จากที่ทำมาทั้งหมดตั้งแต่ขั้นตอนที่ 1 เป็นการเตรียมข้อมูล Data Preparation and Image augmentation ของภาพแค่ประเภทเดียวคือ ประเภทใบไม้ ซึ่งเหลือประเภท ใบบอนสีอีก 3 ประเภท คือ ใบกลม ใบไทย และใบยาว ซึ่งวิธีการทำเหมือนกันทุกขั้นตอนเพียงแค่เปลี่ยนชื่อ Project Name ในขั้นตอนที่ 5
16. เมื่อดาวน์โหลดไฟล์แล้วเราจะทำการเก็บไฟล์ไว้บน Google Drive โดยการทำให้ไฟล์ zip และ upload ขึ้นบน Drive ของ Google เพื่อสามารถนำมาใช้งานต่อไป

3.5 Create Model and Transfer Learning

3.5.1 ใช้ภาษา Python 3.8.16 Tensorflow version: 2.9.2 Keras.models version: 2.9.0 tqdm version: 4.64.1 OpenCV version: 4.6.0 บนเครื่องคอมพิวเตอร์ 1 เครื่อง INBook_X1 Processor Intel® Core(TM) i3-1005G1 CPU @1.20GHz 1.19 GHz การออกแบบโมเดล CNN มี Input layer = 224x224x3 จากนั้นเข้ากระบวนการ convolution โดยแบบออกเป็น 4 ขั้นตอน ซึ่งใช้ Conv2D มี filter ขนาด 5x5 โดยขยับทีละ strides=1 จำนวน layer 16 ,32, 64,128 ตามลำดับ โดยทุกขั้นตอน จะทำการลดขนาดมิติผ่าน MaxPooling2D ที่มีขนาด filter=2x2 ซึ่งการถ่ายโอน (Transfer learning) เป็นเทคนิคในการช่วยลดเวลาในการฝึก (Train) โมเดล (Model) ด้วยการนำ โมเดล (Model) ที่ฝึก (Train) เรียบร้อยแล้ว มาใช้เป็นส่วนหนึ่งของ Mode ใหม่ที่จะใช้กับงาน

3.5.2 การฝึกชุดทดสอบโดยใส่ K-Fold Cross Validation

ทำการฝึกโดยใส่ K-Fold Cross Validation เพื่อประเมินประสิทธิภาพของโมเดลและเปรียบเทียบโมเดลต่าง ๆ อย่างเป็นกลาง และ K-Fold Cross Validation ยังเป็นเครื่องมือที่ช่วยให้เราตัดสินใจได้ว่าจะแบ่งข้อมูลส่วนใดของ Training Data ซึ่งการทำ K-fold cv คือการแบ่งข้อมูลเป็น K ส่วนเท่า ๆ กันเพื่อสร้างและทดสอบโมเดล (train และ validate) คำนวณค่าเฉลี่ยค่าความถูกต้อง (Accuracy) หรือ error ก่อนที่จะนำโมเดลไปใช้ทำนายข้อมูล test set การแบ่งข้อมูลเป็น 5 ชุดข้อมูลเท่า ๆ กัน จะเป็นการแบ่งข้อมูลที่เป็นแบบสุ่ม (random) เพราะการสุ่ม (randomness) จะช่วยให้ข้อมูลในแต่ละ fold มีการกระจายตัวใกล้เคียงกัน ช่วยลด bias เวลาที่เราสร้างและทดสอบโมเดล จำนวน K ที่นิยมใช้กันในทางปฏิบัติมีสองค่าคือ K=5 หรือ K=10 ตัวอย่างเช่น วิจัยของ ไพศาล สิวาเลาเต่า และจรัญ แสนราช (2562) ซึ่งมีข้อมูลจำนวนที่สามารถใช้ในงานวิจัยได้ 152,850 ชุด ทำการแบ่งข้อมูลด้วยวิธี Cross Validation Test ออกเป็น 5 ส่วน หรือ 5-Fold Cross Validation ได้ผลออกมาโดยมีค่าความแม่นยำ 90.60% ค่าความระลึกลับ 98.39% ค่าความแม่นยำ 99.01% และ ค่าถ่วงดุล 0.321 จากวิจัยข้างต้นทางผู้จัดทำจึงเลือก K=5 มาปรับใช้กับโมเดล (model) เพราะ ค่า

ความแม่นยำ ค่าความถูกต้องของวิจัยที่กล่าวมาข้างต้นมีผลลัพธ์ที่ใกล้เคียงกัน และมีระยะเวลาในการ Train ที่สั้นกว่า

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 1	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 2	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 3	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 4	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 5	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5

ภาพที่ 3-16 ตัวอย่าง 5-Fold Cross Validation

ที่มา: Ajitesh Kumar (ออนไลน์)



บทที่ 4

ผลการดำเนินงาน

จากการทดลองระบบตรวจจับและระบุประเภทไม้ประดับบอนสี ด้วยการประมวลผลภาพ ผู้วิจัยได้ดำเนินการศึกษาตามวัตถุประสงค์ ในการทดสอบประสิทธิภาพของงานวิจัยที่นำเสนอได้ทำการทดลองบนคอมพิวเตอร์ Processor Intel® Core(TM) i3-1005G1 CPU @1.20GHz 1.19 GHz โดยมีข้อมูลภาพประเภทของไม้ประดับบอนสีจำนวน 4 ประเภท ได้แก่ ใบแผ่ ใบยาว ใบไทย และใบกลม มีจำนวน Data set ประเภทละ 375 ภาพ โดยแบ่งเป็น Training set จำนวน 300 ภาพ และ Test set จำนวน 75 ภาพ Data set ของประเภทใบทั้ง 4 ประเภท จำนวน 1,500 ภาพ

โดยได้ดำเนินการตามขั้นตอนโดยใช้โมเดล 5 โมเดลในการทดสอบ โดยนำค่า Accuracy และ Loss ระหว่างการฝึกมาแสดงความสัมพันธ์ในแต่ละขั้นตอนของ Epoch ที่ทำการฝึก ระหว่างค่า Accuracy และค่า Loss ในชุด Train และ Validation



4.1 ผลลัพธ์จากข้อมูลหลังทำการฝึก โดยไม่ใส่ K-fold Validation

4.1.1 ผลลัพธ์จาก Model Keres

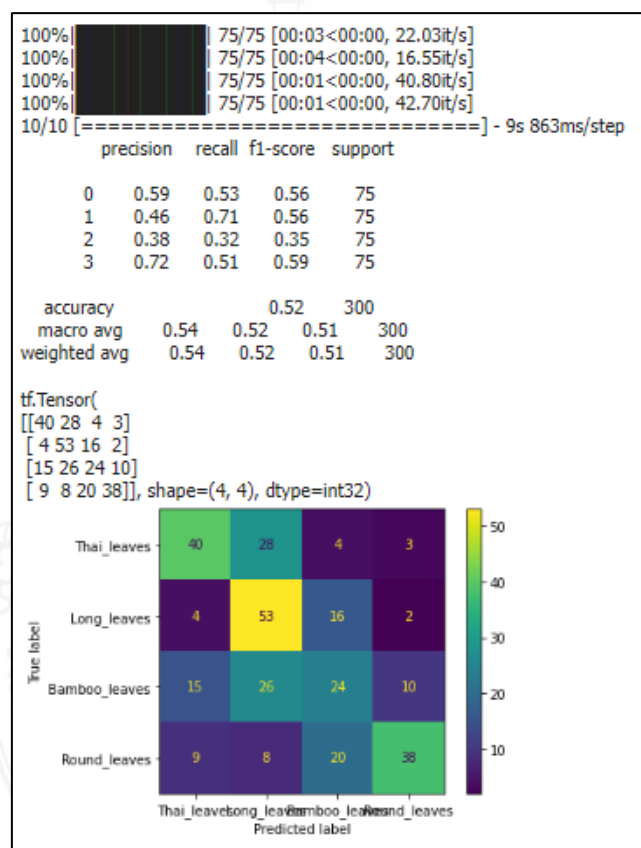
```

1 # Evaluate the restored model
2 loss, acc = new_model.evaluate(x_test, y_test, verbose=2)
3 print('Restored model, accuracy: {:.2f}%'.format(100 * acc))
4
5 print(new_model.predict(x_test).shape)

10/10 - 10s - loss: 3.2482 - accuracy: 0.5167 - 10s/epoch - 969ms/step
Restored model, accuracy: 51.67%
10/10 [=====] - 7s 721ms/step
(300, 4)

```

ภาพที่ 4-1 แสดงการคำนวณค่า Evaluate ของ Model Keres



ภาพที่ 4-2 แสดง Report และ Confusion Matrix ของ Model Keres

จากภาพที่ 4-2 แสดงค่าความถูกต้องของแต่ละคลาส โดยคลาสของใบไทย แทนค่าด้วย 0 คลาสของใบยาวแทนค่าด้วย 1 คลาสของใบไผ่ แทนค่าด้วย 2 และ คลาสของใบกลม แทนค่าด้วย 3 สามารถสรุปได้ว่า คลาสใบไทย มีความถูกต้องร้อยละ 56 ทำนายถูก 40 ภาพ จากทั้งหมด 75 ภาพ คลาสใบยาว มีความถูกต้องร้อยละ 56 ทำนายถูก 53 ภาพ จากทั้งหมด 75 ภาพ คลาสใบไผ่ มีความถูกต้องร้อยละ 35 ทำนายถูก 24 ภาพ จากทั้งหมด 75 ภาพ คลาสใบกลมมีความถูกต้องร้อยละ 59 ทำนายถูก 38 ภาพจากทั้งหมด 75 ภาพ

4.1.2 ผลลัพธ์จาก Model MobilenetV2

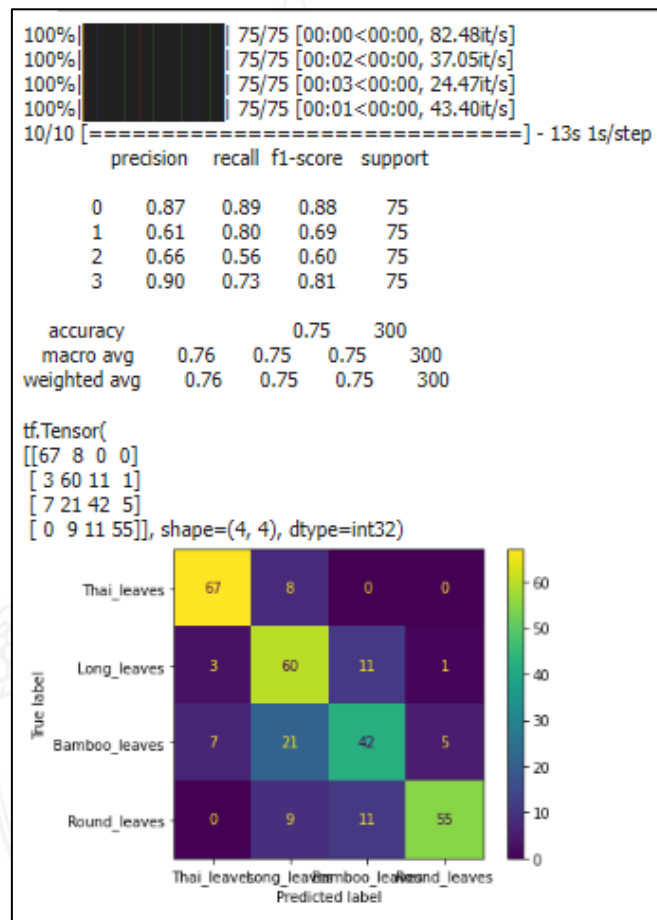
```

1 # Evaluate the restored model
2 loss, acc = mobilenetv2_model.evaluate(x_test, y_test, verbose=2)
3 print('Restored model, accuracy: {:.2f}%'.format(100 * acc))
4
5 print(mobilenetv2_model.predict(x_test).shape)

10/10 - 13s - loss: 1.8776 - accuracy: 0.7467 - 13s/epoch - 1s/step
Restored model, accuracy: 74.67%
10/10 [=====] - 12s 1s/step
(300, 4)

```

ภาพที่ 4-3 แสดงการคำนวณค่า Evaluate ของ Model Mobilenetv2



ภาพที่ 4-4 แสดง Report และ Confusion Matrix ของ Model Mobilenetv2

จากภาพที่ 4-4 แสดงค่าความถูกต้องของแต่ละคลาส โดยคลาสของใบไทย แทนค่าด้วย 0 คลาสของใบยาวแทนค่าด้วย 1 คลาสของใบไผ่ แทนค่าด้วย 2 และ คลาสของใบกลม แทนค่าด้วย 3 สามารถสรุปได้ว่า คลาสใบไทย มีความถูกต้องร้อยละ 88 ทำนายถูก 67 ภาพ จากทั้งหมด 75 ภาพ คลาสใบยาว มีความถูกต้องร้อยละ 69 ทำนายถูก 60 ภาพ จากทั้งหมด 75 ภาพ คลาสใบไผ่ มีความถูกต้องร้อยละ 60 ทำนายถูก 42 ภาพ จากทั้งหมด 75 ภาพ คลาสใบกลม มีความถูกต้องร้อยละ 81 ทำนายถูก 55 ภาพ จากทั้งหมด 75 ภาพ

4.1.3 ผลลัพธ์จาก Model InceptionV3

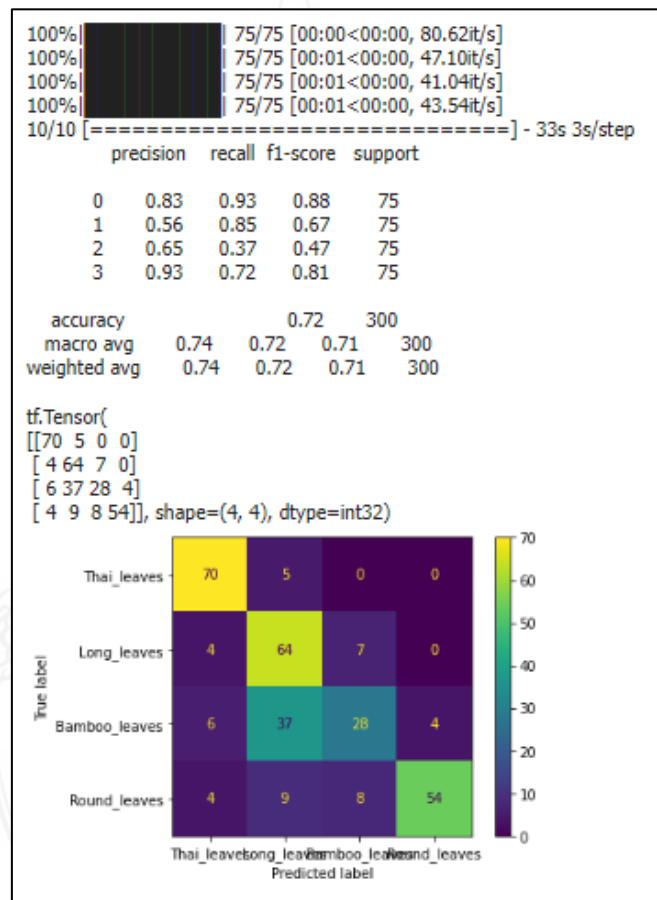
```

1 # Evaluate the restored model
2 loss, acc = inception_model.evaluate(x_test, y_test, verbose=2)
3 print('Restored model, accuracy: {:.5.2f}%'.format(100 * acc))
4
5 print(inception_model.predict(x_test).shape)

10/10 - 35s - loss: 1.9501 - accuracy: 0.7200 - 35s/epoch - 4s/step
Restored model, accuracy: 72.00%
10/10 [=====] - 35s 3s/step
(300, 4)

```

ภาพที่ 4-5 แสดงการคำนวณค่า Evaluate ของ Model InceptionV3



ภาพที่ 4-6 แสดง Report และ Confusion Matrix ของ Model InceptionV3

จากภาพที่ 4-6 แสดงค่าความถูกต้องของแต่ละคลาส โดยคลาสของใบไทย แทนค่าด้วย 0 คลาสของใบยาวแทนค่าด้วย 1 คลาสของใบไผ่ แทนค่าด้วย 2 และ คลาสของใบกลม แทนค่าด้วย 3 สามารถสรุปได้ว่า คลาสใบไทย มีความถูกต้องร้อยละ 88 ทำนายถูก 70 ภาพ จากทั้งหมด 75 ภาพ คลาสใบยาว มีความถูกต้องร้อยละ 67 ทำนายถูก 64 ภาพ จากทั้งหมด 75 ภาพ คลาสใบไผ่ มีความถูกต้องร้อยละ 47 ทำนายถูก 28 ภาพ จากทั้งหมด 75 ภาพ คลาสใบกลม มีความถูกต้องร้อยละ 81 ทำนายถูก 54 ภาพ จากทั้งหมด 75 ภาพ

4.1.4 ผลลัพธ์จาก Model VGG16

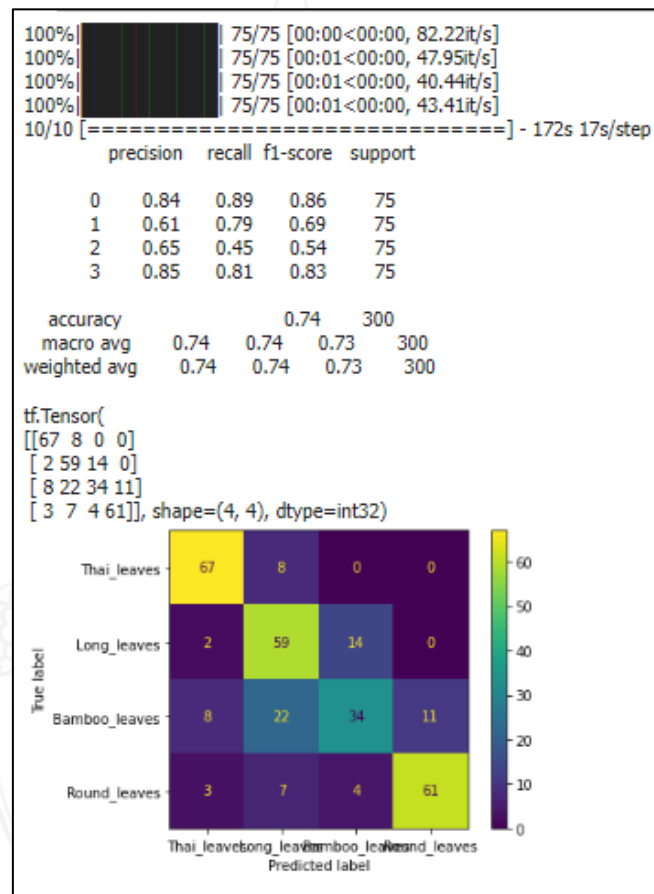
```

1 # Evaluate the restored model
2 loss, acc = vgg16_model.evaluate(x_test, y_test, verbose=2)
3 print("Restored model, accuracy: {:.2f}%".format(100 * acc))
4
5 print(vgg16_model.predict(x_test).shape)

10/10 - 167s - loss: 1.6212 - accuracy: 0.7367 - 167s/epoch - 17s/step
Restored model, accuracy: 73.67%
10/10 [=====] - 167s 16s/step
(300, 4)

```

ภาพที่ 4-7 แสดงการคำนวณค่า Evaluate ของ Model VGG16



ภาพที่ 4-8 แสดง Report และ Confusion Matrix ของ Model VGG16

จากภาพที่ 4-8 แสดงค่าความถูกต้องของแต่ละคลาส โดยคลาสของใบไทย แทนค่าด้วย 0 คลาสของใบยาวแทนค่าด้วย 1 คลาสของใบไผ่ แทนค่าด้วย 2 และ คลาสของใบกลม แทนค่าด้วย 3 สามารถสรุปได้ว่าคลาสใบไทย มีความถูกต้องร้อยละ 86 ทำนายถูก 67 ภาพ จากทั้งหมด 75 ภาพ คลาสใบยาว มีความถูกต้องร้อยละ 69 ทำนายถูก 59 ภาพ จากทั้งหมด 75 ภาพ คลาสใบไผ่ มีความถูกต้องร้อยละ 54 ทำนายถูก 34 ภาพ จากทั้งหมด 75 ภาพ คลาสใบกลม มีความถูกต้องร้อยละ 83 ทำนายถูก 61 ภาพ จากทั้งหมด 75 ภาพ

4.1.5 ผลลัพธ์จาก Model Resnet50V2

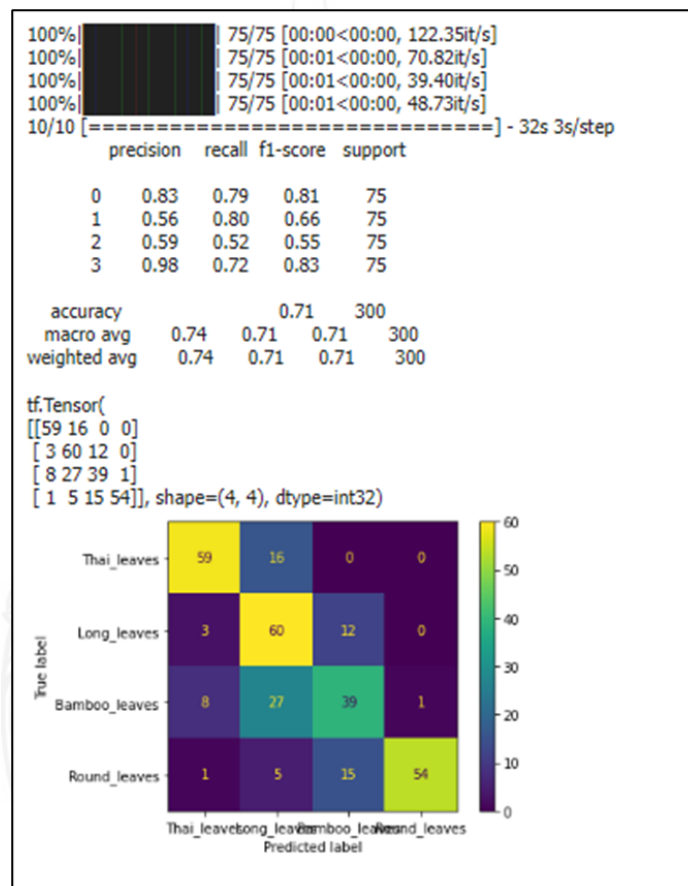
```

1 # Evaluate the restored model
2 loss, acc = resnetv2_model.evaluate(x_test, y_test, verbose=2)
3 print('Restored model, accuracy: {:.2f}%'.format(100 * acc))
4
5 print(resnetv2_model.predict(x_test).shape)

10/10 - 34s - loss: 1.3365 - accuracy: 0.7067 - 34s/epoch - 3s/step
Restored model, accuracy: 70.67%
10/10 [=====] - 34s 3s/step
(300, 4)

```

ภาพที่ 4-9 แสดงการคำนวณค่า Evaluate ของ Model Resnet50V2



ภาพที่ 4-10 แสดง Report และ Confusion Matrix ของ Model Resnet50V2

จากภาพที่ 4-10 แสดงค่าความถูกต้องของแต่ละคลาส โดยคลาสของใบไทย แทนค่าด้วย 0 คลาสของใบยาวแทนค่าด้วย 1 คลาสของใบไผ่ แทนค่าด้วย 2 และ คลาสของใบกลม แทนค่าด้วย 3 สามารถสรุปได้ว่า คลาสใบไทย มีความถูกต้องร้อยละ 81 ทำนายถูก 59 ภาพ จากทั้งหมด 75 ภาพ คลาสใบยาว มีความถูกต้องร้อยละ 66 ทำนายถูก 60 ภาพ จากทั้งหมด 75 ภาพ คลาสใบไผ่ มีความถูกต้องร้อยละ 55 ทำนายถูก 39 ภาพ จากทั้งหมด 75 ภาพ คลาสใบกลม มีความถูกต้องร้อยละ 83 ทำนายถูก 54 ภาพ จากทั้งหมด 75 ภาพ

4.2 วิเคราะห์ความสัมพันธ์จากข้อมูลหลังทำการฝึก โดยใช้ K-fold Validation

4.2.1 ผลลัพธ์จาก Model Keras

```

-----
Score per fold
-----
> Fold 1 - Loss: 0.9392790794372559 - Accuracy: 69.41666603088379%
-----
> Fold 2 - Loss: 0.3905743956565857 - Accuracy: 89.33333158493042%
-----
> Fold 3 - Loss: 0.18702389299869537 - Accuracy: 98.0833351612091%
-----
> Fold 4 - Loss: 0.8151196241378784 - Accuracy: 70.58333158493042%
-----
> Fold 5 - Loss: 0.5919873714447021 - Accuracy: 80.33333420753479%
-----
Average scores for all folds:
> Accuracy: 81.5499997138977 (+- 10.980867972037524)
> Loss: 0.5847968727350235
-----

```

ภาพที่ 4-11 แสดงค่า Average Scores ของทุก Fold ของ Model Keras

จากภาพที่ 4-11 แสดงค่า Average Scores ของทุก Fold โดยมีค่าดังต่อไปนี้ Fold 1 มีค่า Accuracy คือ 69.416 ค่า Loss คือ 0.939 Fold 2 มีค่า Accuracy คือ 89.333 ค่า Loss คือ 0.390 Fold 3 มีค่า Accuracy คือ 98.083 ค่า Loss คือ 0.187 Fold 4 มีค่า Accuracy คือ 70.583 ค่า Loss คือ 0.815 Fold 5 มีค่า Accuracy คือ 80.333 ค่า Loss คือ 0.591 และค่า Average Scores ของทุก Fold มีค่า Accuracy คือ 81.549 และค่า Loss คือ 0.584

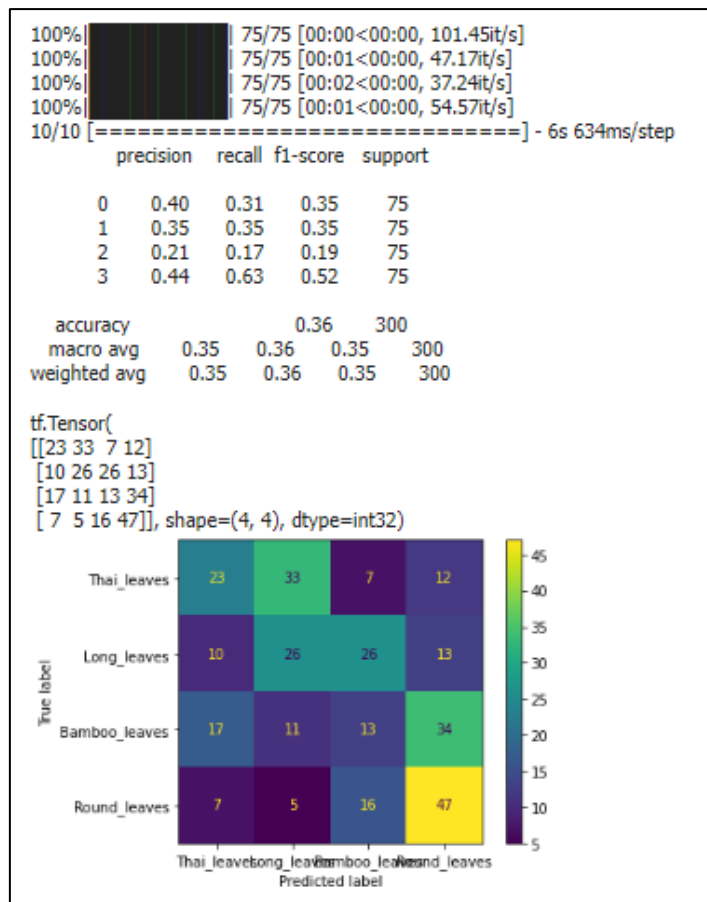
```

1 # Evaluate model's accuracy
2 score = model.evaluate(x_test, y_test, verbose=0)
3 print("Loss is", score[0])
4 print("Accuracy score is", score[1])

Loss is 1.4689892530441284
Accuracy score is 0.3633333444595337

```

ภาพที่ 4-12 แสดงค่า Evaluate ของการ Test ของ Model Keras



ภาพที่ 4-13 แสดง Report และ Confusion Matrix เมื่อทำการ Test ของ Model Keras

จากภาพที่ 4-13 แสดงค่าความถูกต้องของแต่ละคลาสเมื่อทำการ Test โดยคลาสของใบไทย แทนค่า ด้วย 0 คลาสของใบยาวแทนค่าด้วย 1 คลาสของใบไผ่ แทนค่าด้วย 2 และ คลาสของใบกลม แทนค่าด้วย 3 สามารถสรุปได้ว่า คลาสใบไทย มีความถูกต้องร้อยละ 35 ทำนายถูก 23 ภาพ จากทั้งหมด 75 ภาพ คลาสใบยาว มีความถูกต้องร้อยละ 35 ทำนายถูก 26 ภาพ จากทั้งหมด 75 ภาพ คลาสใบไผ่ มีความถูกต้องร้อยละ 19 ทำนายถูก 13 ภาพ จากทั้งหมด 75 ภาพ คลาสใบกลม มีความถูกต้องร้อยละ 52 ทำนายถูก 47 ภาพ จากทั้งหมด 75 ภาพ

4.2.2 ผลลัพธ์จาก Model MoblienetV2

```

-----
Score per fold
-----
> Fold 1 - Loss: 0.03678108751773834 - Accuracy: 99.91666674613953%
-----
> Fold 2 - Loss: 0.052413459867239 - Accuracy: 99.58333373069763%
-----
> Fold 3 - Loss: 0.022623775526881218 - Accuracy: 100.0%
-----
> Fold 4 - Loss: 0.08805211633443832 - Accuracy: 98.91666769981384%
-----
> Fold 5 - Loss: 0.04298389330506325 - Accuracy: 99.83333349227905%
-----
Average scores for all folds:
> Accuracy: 99.65000033378601 (+- 0.39228636908420234)
> Loss: 0.04857086651027202
-----

```

ภาพที่ 4-14 แสดงค่า Average Scores ของทุก Fold ของ Model MoblienetV2

จากภาพที่ 4.14 แสดงค่า Average Scores ของทุก Fold โดยมีค่าดังต่อไปนี้ Fold 1 มีค่า Accuracy คือ 99.916 ค่า Loss คือ 0.036 Fold 2 มีค่า Accuracy คือ 99.583 ค่า Loss คือ 0.052 Fold 3 มีค่า Accuracy คือ 100.0 ค่า Loss คือ 0.022 Fold 4 มีค่า Accuracy คือ 98.916 ค่า Loss คือ 0.088 Fold 5 มีค่า Accuracy คือ 99.833 ค่า Loss คือ 0.042 และค่า Average Scores ของทุก Fold มีค่า Accuracy คือ 99.650 และค่า Loss คือ 0.048

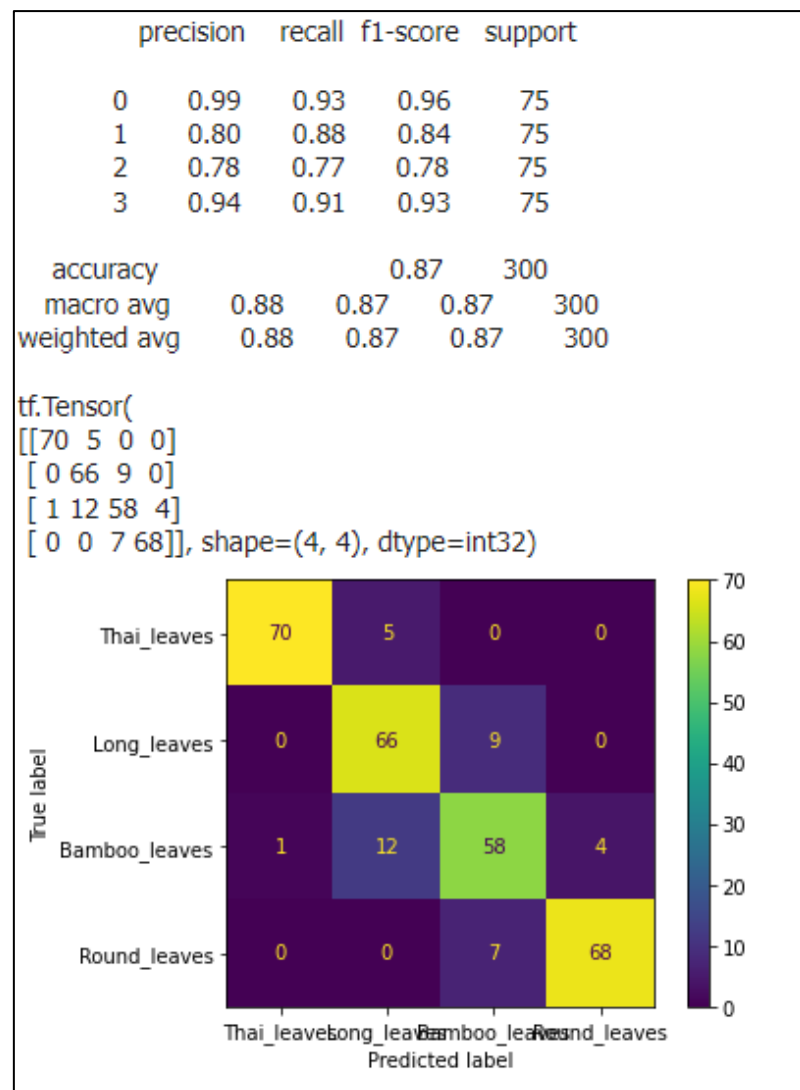
```

1 # Evaluate model's accuracy
2 score = model.evaluate(x_test, y_test, verbose=0)
3 print("Loss is", score[0])
4 print("Accuracy score is", score[1])

Loss is 0.38815316557884216
Accuracy score is 0.8733333349227905

```

ภาพที่ 4-15 แสดงค่า Evaluate ของการ Test ของ Model MoblienetV2



ภาพที่ 4-16 แสดง Report และ Confusion Matrix เมื่อทำการ Test ของ Model MobilenetV2

จากภาพที่ 4-16 แสดงค่าความถูกต้องของแต่ละคลาสเมื่อทำการ Test โดยคลาสของใบไทย แทนค่า ด้วย 0 คลาสของใบยาวแทนค่าด้วย 1 คลาสของใบไผ่ แทนค่าด้วย 2 และ คลาสของใบกลม แทนค่าด้วย 3 สามารถสรุปได้ว่า คลาสใบไทย มีความถูกต้องร้อยละ 96 ทำนายถูก 70 ภาพ จากทั้งหมด 75 ภาพ คลาสใบยาว มีความถูกต้องร้อยละ 84 ทำนายถูก 66 ภาพ จากทั้งหมด 75 ภาพ คลาสใบไผ่ มีความถูกต้องร้อยละ 78 ทำนายถูก 58 ภาพ จากทั้งหมด 75 ภาพ คลาสใบกลม มีความถูกต้องร้อยละ 93 ทำนายถูก 68 ภาพ จากทั้งหมด 75 ภาพ

4.2.3 ผลลัพธ์จาก Model InceptionV3

```

-----
Score per fold
-----
> Fold 1 - Loss: 0.11631492525339127 - Accuracy: 97.91666865348816%
-----
> Fold 2 - Loss: 0.11144030094146729 - Accuracy: 97.83333539962769%
-----
> Fold 3 - Loss: 0.615064263343811 - Accuracy: 75.58333277702332%
-----
> Fold 4 - Loss: 0.9504403471946716 - Accuracy: 52.58333086967468%
-----
> Fold 5 - Loss: 0.338371604681015 - Accuracy: 88.66666555404663%
-----

Average scores for all folds:
> Accuracy: 82.5166666507721 (+- 17.050856866122036)
> Loss: 0.4263262882828712
-----

```

ภาพที่ 4-17 แสดงค่า Average Scores ของทุก Fold ของ Model InceptionV3

จากภาพที่ 4-17 แสดงค่า Average Scores ของทุก Fold โดยมีค่าดังต่อไปนี้ Fold 1 มีค่า Accuracy คือ 97.916 ค่า Loss คือ 0.116 Fold 2 มีค่า Accuracy คือ 97.833 ค่า Loss คือ 0.111 Fold 3 มีค่า Accuracy คือ 75.583 ค่า Loss คือ 0.615 Fold 4 มีค่า Accuracy คือ 52.583 ค่า Loss คือ 0.950 Fold 5 มีค่า Accuracy คือ 88.666 ค่า Loss คือ 0.338 และค่า Average Scores ของทุก Fold มีค่า Accuracy คือ 82.516 และค่า Loss คือ 0.426

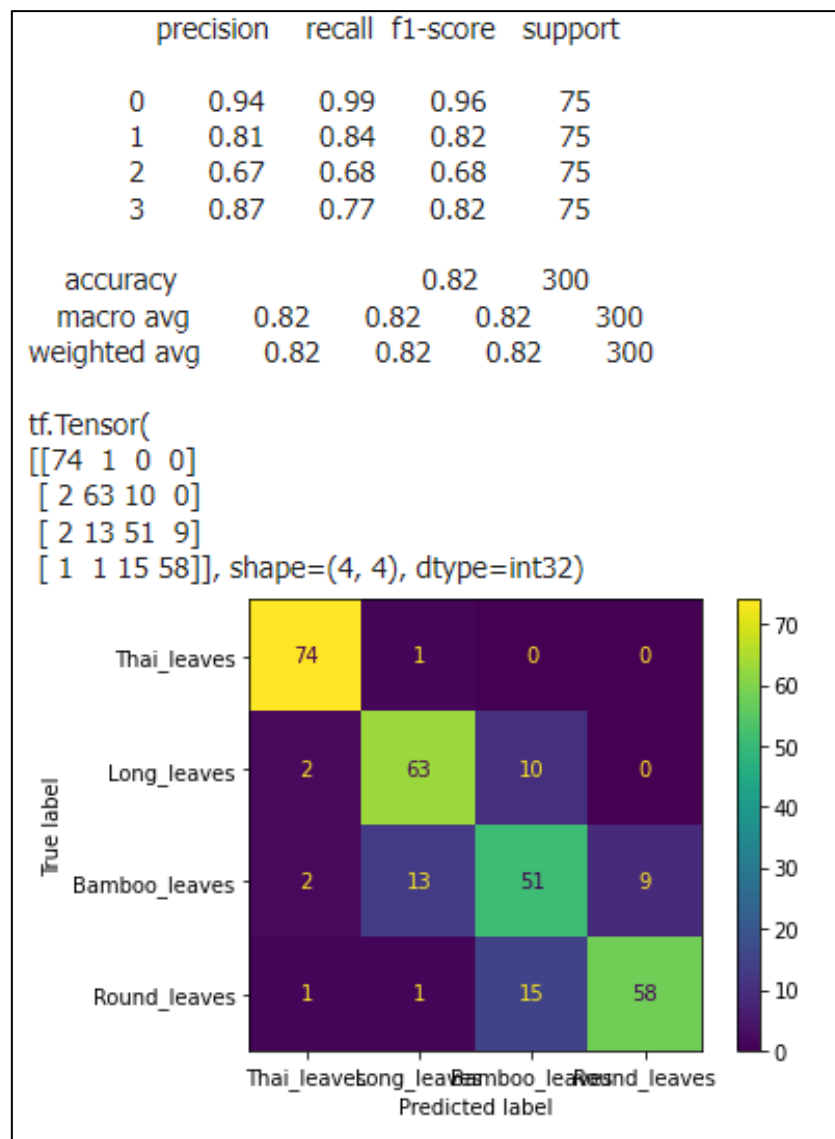
```

1 # Evaluate model's accuracy
2 score = model.evaluate(x_test, y_test, verbose=0)
3 print("Loss is", score[0])
4 print("Accuracy score is", score[1])

Loss is 0.4841155409812927
Accuracy score is 0.8199999928474426

```

ภาพที่ 4-18 แสดงค่า Evaluate ของการ Test ของ Model InceptionV3



ภาพที่ 4-19 แสดง Report และ Confusion Matrix เมื่อทำการ Test ของ Model InceptionV3

จากภาพที่ 4-19 แสดงค่าความถูกต้องของแต่ละคลาสเมื่อทำการ Test โดยคลาสของใบไทย แทนค่า ด้วย 0 คลาสของใบยาวแทนค่าด้วย 1, คลาสของใบไผ่ แทนค่าด้วย 2 และคลาสของใบกลม แทนค่าด้วย 3 สามารถสรุปได้ว่า คลาสใบไทย มีความถูกต้องร้อยละ 96 ทำนายถูก 74 ภาพ จากทั้งหมด 75 ภาพ คลาสใบยาว มีความถูกต้องร้อยละ 82 ทำนายถูก 63 ภาพ จากทั้งหมด 75 ภาพ คลาสใบไผ่ มีความถูกต้องร้อยละ 68 ทำนายถูก 51 ภาพ จากทั้งหมด 75 ภาพ คลาสใบกลม มีความถูกต้องร้อยละ 82 ทำนายถูก 58 ภาพ จากทั้งหมด 75 ภาพ

4.2.4 ผลลัพธ์จาก Model VGG16

```

-----
Score per fold
-----
> Fold 1 - Loss: 0.28378352522850037 - Accuracy: 90.66666960716248%
-----
> Fold 2 - Loss: 0.2984534502029419 - Accuracy: 89.083331823349%
-----
> Fold 3 - Loss: 0.32165074348449707 - Accuracy: 90.08333086967468%
-----
> Fold 4 - Loss: 0.2592080235481262 - Accuracy: 93.00000071525574%
-----
> Fold 5 - Loss: 0.2374015599489212 - Accuracy: 93.16666722297668%
-----
Average scores for all folds:
> Accuracy: 91.20000004768372 (+- 1.619843076686429)
> Loss: 0.2800994604825974
-----

```

ภาพที่ 4-20 แสดงค่า Average Scores ของทุก Fold ของ Model VGG16

จากภาพที่ 4-20 แสดงค่า Average Scores ของทุก Fold โดยมีค่าดังต่อไปนี้
 Fold 1 มีค่า Accuracy คือ 90.666 ค่า Loss คือ 0.283 Fold 2 มีค่า Accuracy คือ 89.083
 ค่า Loss คือ 0.298 Fold 3 มีค่า Accuracy คือ 90.083 ค่า Loss คือ 0.321 Fold 4 มีค่า Accuracy
 คือ 93.000 ค่า Loss คือ 0.259 Fold 5 มีค่า Accuracy คือ 93.166 ค่า Loss คือ 0.237 และค่า
 Average Scores ของทุก Fold มีค่า Accuracy คือ 91.200 และค่า Loss คือ 0.280

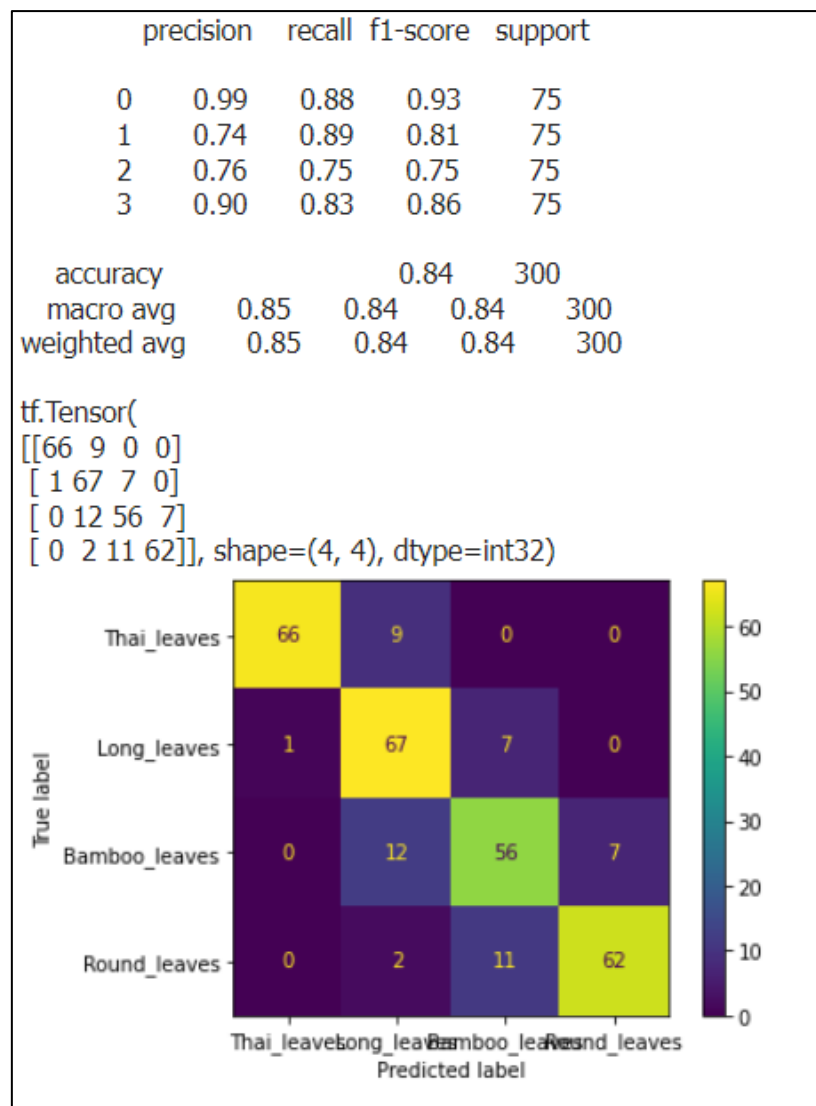
```

1 # Evaluate model's accuracy
2 score = model.evaluate(x_test, y_test, verbose=0)
3 print("Loss is", score[0])
4 print("Accuracy score is", score[1])

Loss is 0.4506162106990814
Accuracy score is 0.8366666436195374

```

ภาพที่ 4-21 แสดงค่า Evaluate ของการ Test ของ Model VGG16



ภาพที่ 4-22 แสดง Report และ Confusion Matrix เมื่อทำการ Test ของ Model VGG16

จากภาพที่ 4-22 แสดงค่าความถูกต้องของแต่ละคลาสเมื่อทำการ Test โดยคลาสของใบไทย แทนค่า ด้วย 0 คลาสของใบยาวแทนค่าด้วย 1 คลาสของใบไม้ แทนค่าด้วย 2 และคลาสของใบกลม แทนค่าด้วย 3 สามารถสรุปได้ว่า คลาสใบไทย มีความถูกต้องร้อยละ 93 ทำนายถูก 66 ภาพ จากทั้งหมด 75 ภาพ คลาสใบยาว มีความถูกต้องร้อยละ 81 ทำนายถูก 67 ภาพ จากทั้งหมด 75 ภาพ คลาสใบไม้ มีความถูกต้องร้อยละ 75 ทำนายถูก 56 ภาพ จากทั้งหมด 75 ภาพ คลาสใบกลม มีความถูกต้องร้อยละ 86 ทำนายถูก 62 ภาพ จากทั้งหมด 75 ภาพ

4.2.5 ผลลัพธ์จาก Model Resnet50V2

```

-----
Score per fold
-----
> Fold 1 - Loss: 0.00526235019788146 - Accuracy: 100.0%
-----
> Fold 2 - Loss: 0.01586051844060421 - Accuracy: 100.0%
-----
> Fold 3 - Loss: 0.0456710010766983 - Accuracy: 99.6666669845581%
-----
> Fold 4 - Loss: 0.0502135194838047 - Accuracy: 99.83333349227905%
-----
> Fold 5 - Loss: 0.012030099518597126 - Accuracy: 100.0%
-----
Average scores for all folds:
> Accuracy: 99.90000009536743 (+- 0.1333332061767578)
> Loss: 0.02580749774351716
-----

```

ภาพที่ 4-23 แสดงค่า Average Scores ของทุก Fold ของ Model Resnet50V2

จากภาพที่ 4-23 แสดงค่า Average Scores ของทุก Fold โดยมีค่าดังต่อไปนี้ Fold 1 มีค่า Accuracy คือ 100.0 ค่า Loss คือ 0.005 Fold 2 มีค่า Accuracy คือ 100.0 ค่า Loss คือ 0.015 Fold 3 มีค่า Accuracy คือ 99.666 ค่า Loss คือ 0.045 Fold 4 มีค่า Accuracy คือ 99.833 ค่า Loss คือ 0.050 Fold 5 มีค่า Accuracy คือ 100.0 ค่า Loss คือ 0.012 และค่า Average Scores ของทุก Fold มีค่า Accuracy คือ 99.900 และค่า Loss คือ 0.025

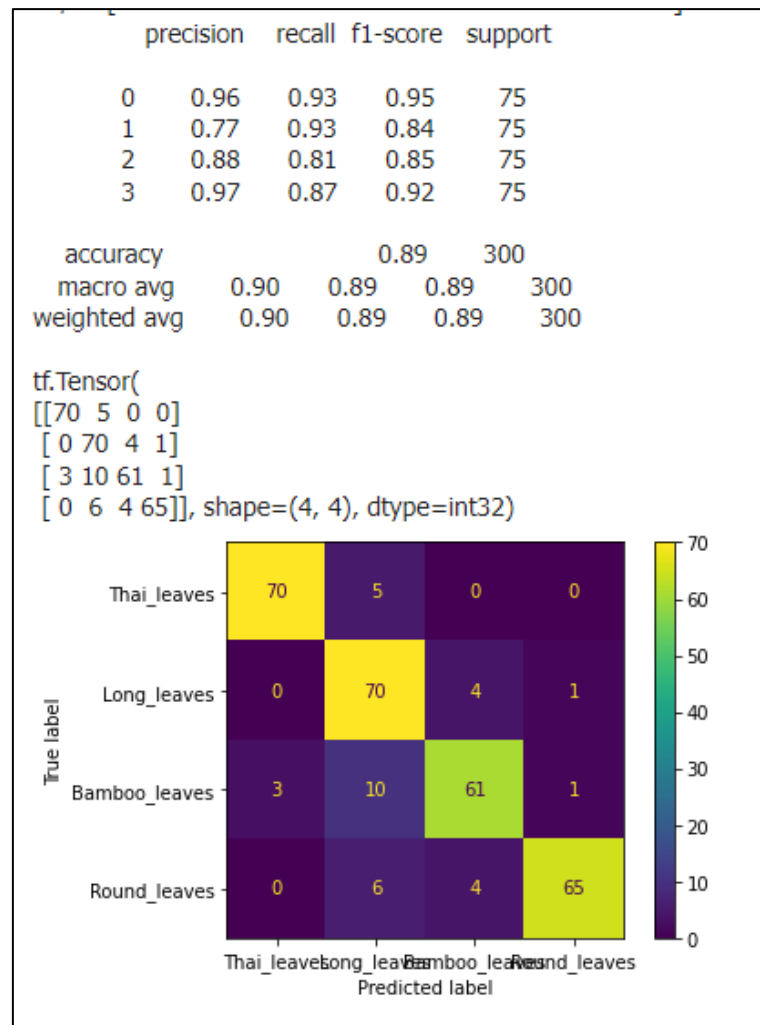
```

1 # Evaluate model's accuracy
2 score = model.evaluate(x_test, y_test, verbose=0)
3 print("Loss is", score[0])
4 print("Accuracy score is", score[1])

Loss is 0.5073531866073608
Accuracy score is 0.8866666555404663

```

ภาพที่ 4-24 แสดงค่า Evaluate ของการ Test ของ Model Resnet50V2



ภาพที่ 4-25 แสดง Report และ Confusion Matrix เมื่อทำการ Test ของ Model Resnet50V2

จากภาพที่ 4-25 แสดงค่าความถูกต้องของแต่ละคลาสเมื่อทำการ Test โดยคลาสของใบไทย แทนค่า ด้วย 0 คลาสของใบยาวแทนค่าด้วย 1 คลาสของใบไผ่ แทนค่าด้วย 2 และคลาสของใบกลม แทนค่าด้วย 3 สามารถสรุปได้ว่า คลาสใบไทย มีความถูกต้องร้อยละ 95 ทำนายถูก 70 ภาพ จากทั้งหมด 75 ภาพ คลาสใบยาว มีความถูกต้องร้อยละ 84 ทำนายถูก 70 ภาพ จากทั้งหมด 75 ภาพ คลาสใบไผ่ มีความถูกต้องร้อยละ 85 ทำนายถูก 61 ภาพ จากทั้งหมด 75 ภาพ คลาสใบกลม มีความถูกต้องร้อยละ 92 ทำนายถูก 65 ภาพ จากทั้งหมด 75 ภาพ

4.3 การเปรียบเทียบประสิทธิภาพของ Model เมื่อทำการทดสอบ (Test)

ตารางที่ 1 การเปรียบเทียบประสิทธิภาพของ Model เมื่อทำการทดสอบ (Test) ที่มีการฝึกโดยไม่ใส่ K-fold Validation และใส่ K-fold Validation

Models	การฝึกโดยไม่ใส่ K-fold Validation		การฝึกโดยใส่ K-fold Validation	
	Accuracy	Precision	Accuracy	Precision
Resnet50V2	0.71	0.74	0.89	0.90
MobilenetV2	0.76	0.75	0.87	0.88
VGG16	0.74	0.74	0.84	0.85
InceptionV3	0.74	0.72	0.82	0.82
Model Keras	0.52	0.54	0.36	0.35

ตารางที่ 4-1 การเปรียบเทียบประสิทธิภาพของโมเดล (Model)

จากตารางที่ 4-1 การเปรียบเทียบประสิทธิภาพของ Model เมื่อทำการทดสอบ (Test) ที่มีการฝึกโดยไม่ใส่ K-fold Validation ทั้ง 5 Model ได้แก่ Model Keras Model MobilenetV2 Model InceptionV3 Model VGG16 และ Model Resnet50V2 แสดงให้เห็นว่า Model MobilenetV2 มีค่าความถูกต้องมากที่สุดเมื่อเทียบทั้ง 5 โมเดลโดยมีค่า Accuracy คือ 0.76 ค่า Precision คือ 0.75 Model InceptionV3 มีค่า Accuracy คือ 0.74 ค่า Precision คือ 0.72 Model VGG16 มีค่า Accuracy คือ 0.74 ค่า Precision คือ 0.74 Model Resnet50V2 มีค่า Accuracy คือ 0.71 ค่า Precision คือ 0.74 และค่าที่ต่ำที่สุดคือ Model Keras มีค่าความถูกต้องคือ 0.52 และค่า Precision คือ 0.54 ซึ่งเป็นค่าที่ต่ำที่สุดเมื่อเปรียบเทียบทั้ง 5 โมเดล

การเปรียบเทียบประสิทธิภาพของ Model เมื่อทำการทดสอบ (Test) ที่มีการฝึกโดยใส่ K-fold Validation ทั้ง 5 Model ได้แก่ Model Keras, Model MobilenetV2 Model InceptionV3 Model VGG16 และ Model Resnet50V2 แสดงให้เห็นว่า Model Resnet50V2 มีค่าความถูกต้องมากที่สุดเมื่อเทียบทั้ง 5 โมเดลโดยมีค่า Accuracy คือ 0.89 ค่า Precision คือ 0.90 Model MobilenetV2 มีค่า Accuracy คือ 0.87 ค่า Precision คือ 0.88 Model VGG16 มีค่า Accuracy คือ 0.84 ค่า Precision คือ 0.85 Model InceptionV3 มีค่า Accuracy คือ 0.82 ค่า Precision คือ

0.82 และค่าที่ต่ำที่สุดคือ Model Keras มีค่าความถูกต้องคือ 0.36 และค่า Precision คือ 0.35 ซึ่งเป็นค่าที่ต่ำที่สุดเมื่อเปรียบเทียบทั้ง 5 โมเดล

โดย Model Keras ที่ผู้วิจัยสร้างขึ้นมีประสิทธิภาพไม่ถึงตามผลที่คาดว่าจะได้รับเนื่องจากมีข้อจำกัดในเรื่องของระยะเวลาที่ใช้ในการฝึกฝนโมเดล (Model Training) และทรัพยากรในส่วนของคอมพิวเตอร์มีความจำกัดทางผู้วิจัยจึงได้เลือกใช้โมเดล 4 โมเดล ดังนี้ในการทดสอบ

4.4 ผลลัพธ์ในการฝึกฝนโมเดล (Model Training) โดยใช้วิธีการ k-Fold Cross Validation ผลลัพธ์จากการทดสอบเป็นดังตาราง

ตารางที่ 2 ผลลัพธ์ในการฝึกฝนโมเดล (Model Training) โดยใช้วิธีการ k-Fold Cross Validation
ผลลัพธ์จากการทดสอบเป็นดังตาราง

Models	ความถูกต้อง ร้อยละ	การฝึกโดยใส่ k-Fold Cross Validation		รวม (จำนวนภาพ)
		ถูก (จำนวนภาพ)	ผิด (จำนวนภาพ)	
Resnet50V2	88.67	266	34	300
MobilenetV2	87.33	262	38	300
VGG16	83.67	251	49	300
InceptionV3	82.00	246	54	300

ตารางที่ 4-2 ผลลัพธ์ในการฝึกฝนโมเดลโดยใช้วิธีการ k-Fold Cross Validation

จากตารางที่ 4-2 การเปรียบเทียบประสิทธิภาพของ Model เมื่อทำการทดสอบ (Test) ทั้ง 4 Model ได้แก่ Model MobilenetV2 Model InceptionV3 Model VGG16 และ Model Resnet50V2 แสดงให้เห็นว่า Model Resnet50V2 มีค่าความถูกต้องสูงที่สุดร้อยละ 88.67 โดยทำนายภาพถูก 266 ภาพ จากทั้งหมด 300 ภาพ Model MobilenetV2 มีค่าความถูกต้องร้อยละ 87.33 โดยทำนายภาพถูก 262 ภาพ จากทั้งหมด 300 ภาพ Model VGG16 มีค่าความถูกต้องร้อยละ 83.67 โดยทำนายภาพถูก 251 ภาพ จากทั้งหมด 300 ภาพ Model InceptionV3 มีค่าความถูกต้องร้อยละ 82.00 โดยทำนายภาพถูก 246 ภาพ จากทั้งหมด 300 ภาพ ซึ่งเป็นค่าที่ต่ำที่สุดเมื่อเปรียบเทียบทั้ง 4 โมเดล

บทที่ 5

สรุปผลและข้อเสนอแนะ

การศึกษานี้นำเสนอระบบตรวจจับ และระบุประเภทไม้ประดับบอนสีด้วยการประมวลผลภาพด้วยโมเดล (Model) Keras โมเดล (Model) Mobilenetv2 โมเดล (Model) inceptionV3 โมเดล (Model) VGG16 และโมเดล (Model) Resnet50V2 ทางคณะผู้จัดทำเลือกไม้ประดับบอนสีมาในการวิจัยเนื่องจากเกิดสภาวะโรคระบาด COVID-19 เริ่มต้นขึ้นในเดือนธันวาคม พ.ศ. 2562 ซึ่งทำให้สถานประกอบการหลายแห่งจึงต้องปรับการทำงาน โดยให้พนักงานทำงานที่บ้าน (Work From Home: WFH) และเกิดกระแสการเลี้ยงต้นไม้ขนาดเล็กถึงปานกลางเป็นที่นิยมไม้ประดับบอนสีจึงเป็นหนึ่งในตัวเลือกของหลาย ๆ คน ด้วยความหลากหลายของขนาดทำให้ผู้ที่ไม่มีพื้นที่ใช้สอยภายในบ้านน้อยสามารถเลี้ยงได้ และผู้คนเริ่มหันมาสนใจจึงซื้อไม้ประดับบอนสี แต่ด้วยในหนึ่งชนิดพันธุ์ มีหลายชนิด จึงเกิดความเข้าใจผิดเนื่องจากในลักษณะที่คล้ายคลึงกัน และเทคโนโลยีในปัจจุบันที่มีการพัฒนาไปอย่างมากทำให้เกิดข้อมูลในรูปแบบของรูปภาพมากขึ้น ข้อมูลในลักษณะรูปภาพนี้เป็นที่นิยม และยังแพร่กระจายไปอย่างรวดเร็วในสังคมออนไลน์ ซึ่งเทคโนโลยีทางด้านคอมพิวเตอร์วิทัศน์ (Computer Vision) ได้มีการพัฒนาไปอย่างก้าวหน้ารวดเร็วโดยมีการประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึก (Deep learning) โดยเฉพาะการใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolution Neural Networks: CNN) ด้วยเหตุนี้ การนำเทคโนโลยีการประมวลผลภาพ (Image Processing) มาประยุกต์ใช้กับไม้ประดับบอนสีทำให้สามารถนำภาพถ่ายของไม้ประดับบอนสีมาทำการประมวลได้เพิ่มความสะดวกสบายในการระบุประเภทของไม้ประดับบอนสีในกรณีที่ไม่ทราบประเภทของไม้ประดับบอนสี ซึ่งงานวิจัยนี้มีการศึกษาประเภทไม้ประดับบอนสี และเพื่อศึกษาการใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน โดยวิจัยที่นำมาอ้างอิงใช้โมเดล (Model) โครงข่ายประสาทเทียมแบบคอนโวลูชันเป็นส่วนใหญ่ ตัวอย่างเช่น วิจัยเรื่อง การจำแนกผลึกน้ำตาลด้วยวิธีการเรียนรู้เชิงลึก ผลการศึกษาวิจัยพบว่า โมเดล VGG16 ที่ผ่านการปรับแต่งมีความแม่นยำในการจำแนกผลึกน้ำตาลระหว่างร้อยละ 80 ถึงร้อยละ 94 ของภาพผลึกน้ำตาลทั้ง 4 ประเภทที่แยกตามสภาวะการเกิดผลึก และวิจัยเรื่อง การประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึกเพื่อวัดระดับความหวานของแตงโมผ่านสมาร์ตโฟน ผลการทดลองพบว่า โมเดลจากอัลกอริทึม InceptionV3 มีค่าความถูกต้องที่เท่ากับโมเดลจากอัลกอริทึม MobileNetv2 ค่าความถูกต้องเท่ากับร้อยละ 97.20 และเพื่อให้สามารถระบุประเภทไม้ประดับบอนสีโดยใช้เทคนิคการประมวลผลภาพ และโครงข่ายประสาทเทียมแบบคอนโวลูชัน

5.1 สรุปผลการวิจัย

ประเภทไม้ประดับบอนสี

การเลี้ยงต้นไม้ขนาดเล็กถึงปานกลางเป็นที่นิยมไม้ประดับบอนสีจึงเป็นหนึ่งในตัวเลือกของหลาย ๆ คน ด้วยความหลากหลายของขนาดทำให้ผู้ที่ไม่มีพื้นที่ใช้สอยภายในบ้านน้อยสามารถเลี้ยงได้ ด้วยในหนึ่งชนิดพันธุ์ มีหลายชนิด จึงเกิดความเข้าใจผิดเนื่องจากในลักษณะที่คล้ายคลึงกัน จากการศึกษาไม้ประดับบอนสีจากสวน ผู้วิจัยจึงเลือกบอนสี 4 ประเภท คือ

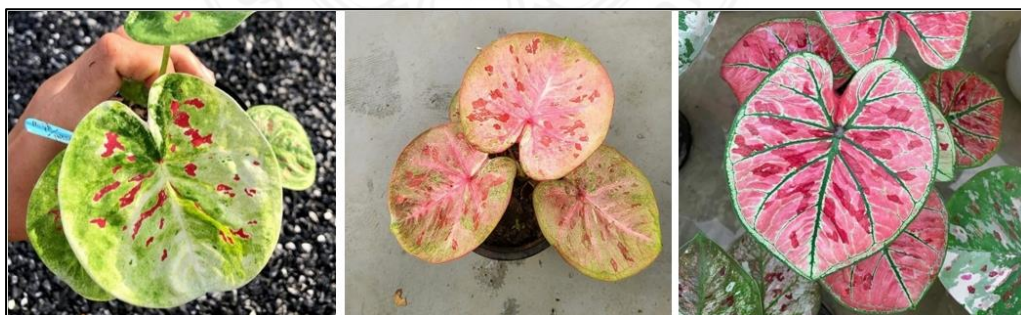
- บอนใบไทย : บอนสีไทยโบราณ ลักษณะใบเป็นรูปหัวใจ หูใบยาว เว้าลึกเกือบถึงสะดือ (ส่วนปลายก้านใบที่จรดกับเส้นกลางใบ) ก้านใบอยู่กึ่งกลางใบ มีทั้งปลายใบแหลม และปลายใบมนขึ้นอยู่กับสายพันธุ์ ใบแผ่กว้าง มีขนาดใหญ่ สีสันสวยงาม ใบดกและไม่ทิ้งใบ



ภาพที่ 5-1 บอนใบไทย

ที่มา: คณะผู้จัดทำ (2565)

- บอนใบกลม : สายพันธุ์บอนสีที่พัฒนามาจากบอนใบไทย ลักษณะใบกลมหรือรี หูใบสั้น ปลายใบมน และก้านใบกลมอยู่กึ่งกลางใบ คล้ายกับใบบัว



ภาพที่ 5-2 บอนใบกลม

ที่มา: คณะผู้จัดทำ (2565)

- **บอนใบยาว** : ลักษณะใบทรงหัวใจคล้ายกับบอนใบไทยเช่นเดียวกัน แต่แตกต่างกันที่ใบของบอนสีใบยาวจะเรียวยาว และมีปลายใบแหลมกว่า หูใบยาวฉีกถึงสะดือ และสามารถแยกได้อีก 3 ลักษณะย่อยคือ บอนใบยาวธรรมดา บอนใบยาวรูปหอก และบอนใบยาวรูปใบไผ่



ภาพที่ 5-3 บอนใบยาว

ที่มา: คณะผู้จัดทำ (2565)

- **บอนใบไผ่** : ลักษณะใบรูปแถบ รูปใบหอกแคบหรือเป็นเส้น ปลายใบเรียวแหลมคล้ายใบไผ่ หูใบสั้นมาก ความกว้างของใบไม่เกิน 2 นิ้ว



ภาพที่ 5-4 บอนใบไผ่

ที่มา: คณะผู้จัดทำ (2565)

โดยจากภาพที่ 5-3 และ 5-4 บอนใบยาว และบอนใบไผ่ที่มีลักษณะใบที่คล้ายคลึงกัน โดยใบยาวจะลักษณะใบทรงหัวใจคล้ายกับบอนใบไทยเช่นเดียวกัน แต่แตกต่างกันที่ ใบของบอนสีใบยาวจะเรียวยาว และมีปลายใบแหลมกว่า หูใบยาวฉีกถึงสะดือ และสามารถแยกได้อีก 3 ลักษณะย่อยคือ บอนใบยาวธรรมดา บอนใบยาวรูปหอก และบอนใบยาวรูปใบไผ่ ส่งผลให้ตรงส่วนของบอนใบยาวรูปใบไผ่จะมักเข้าใจผิดว่าเป็นบอนใบไผ่

งานวิจัยนี้จึงนำเทคโนโลยีการประมวลผลภาพ (Image Processing) มาประยุกต์ใช้กับไม้ประดับบอนสีทำให้สามารถนำภาพถ่ายของไม้ประดับบอนสีมาทำการประมวลได้เพิ่มความสะดวกสบายในการระบุประเภทของไม้ประดับบอนสีในกรณีที่ไม่ทราบประเภทของไม้ประดับบอนสีผู้จัดทำได้ไปเก็บรวบรวมรูปภาพไม้ประดับบอนสีที่ สวนมาลัยทอง อำเภอไทรน้อย จังหวัดนนทบุรี และสวนนพดล อำเภอไทรน้อย จังหวัดนนทบุรี โดยกำหนดประเภทไม้ประดับบอนสี จำนวน 4 ประเภท ได้แก่ ใบไทย ใบยาว ใบกลม และใบแผ่ ซึ่งเก็บภาพตัวอย่างทั้งหมด 1,500 ภาพ แบ่งเป็นชุด Training set ประเภทละ 300 ภาพ เป็นทั้งหมดจำนวน 1,200 ภาพ และชุด Test set 75 ภาพ เป็นทั้งหมดจำนวน 300 ภาพ

การศึกษาคำอธิบายโครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolution Neural Networks: CNN) และการใช้การเรียนรู้แบบถ่ายโอน (Transfer Learning) เป็นวิธีการที่นิยมในการทำวิจัยที่ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน และจากการทบทวนวรรณกรรมที่เกี่ยวข้องพบว่า pre-trained Model ที่ใช้ในงานวิจัยลักษณะคล้ายกันนี้มีหลายโมเดล (Model) คณะผู้วิจัยได้ทำการสร้างโครงข่ายประสาทเทียมแบบคอนโวลูชันขึ้นเอง 1 โมเดล (Model) และใช้ pre-trained Model อีก 4 โมเดล (Model) ที่มีประสิทธิภาพในการจำแนกได้ดี คือ Mobilenetv2 InceptionV3 VGG16 และ Resnet50V2

การระบุประเภทไม้ประดับบอนสีโดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันผู้จัดทำใช้ Keras เป็นโมเดล (Model) พื้นฐานในการเริ่มต้นการจำแนก และได้เลือกโมเดล (Model) เพิ่มอีก 4 ตัว คือ MobileNetV2 InceptionV3 VGG16 และ Resnet50V2 หลังจากนั้นทำการปรับโมเดล (Model) และปรับค่าการสอน (Train) โมเดล (Model) เพื่อจำแนกประเภทไม้ประดับบอนสีโดยใช้ค่าความถูกต้อง (Accuracy) ที่ปรากฏอยู่ในคอนฟิวชันเมทริกซ์ (Confusion Matrix) ซึ่งโมเดล (Model) ใดที่ให้ค่าความถูกต้องมากที่สุดในการทำนายประเภทไม้ประดับบอนสี ถือว่ามีประสิทธิภาพในการทำนายดีที่สุดในงานวิจัย ในการสอน (Train) มีการใช้เทคนิคการวัดประสิทธิภาพ (K-Fold Cross Validation) และแบบไม่ใช้การวัดประสิทธิภาพ (K-Fold Cross Validation) ซึ่งผลที่ได้เป็นดังนี้

1. ผลการวิเคราะห์การเปรียบเทียบประสิทธิภาพของโมเดล (Model) เมื่อทำการทดสอบ (Test) ที่มีการฝึกโดยไม่ใส่การวัดประสิทธิภาพ (K-Fold Cross Validation) ทั้ง 5 โมเดล (Model) ได้แก่ โมเดล (Model) Keras โมเดล (Model) Mobilenetv2 โมเดล (Model) inceptionV3 โมเดล (Model) VGG16 และโมเดล (Model) Resnet50V2 แสดงให้เห็นว่าโมเดล (Model) MobilenetV2 มีค่าความถูกต้อง (Accuracy) มากที่สุดเมื่อเทียบทั้ง 5 โมเดล (Model) โดยมีค่าความถูกต้อง (Accuracy) คือ 0.76 ค่าความแม่นยำ (Precision) คือ 0.75 โมเดล (Model) InceptionV3 มีค่าความถูกต้อง (Accuracy) คือ 0.74 ค่าความแม่นยำ (Precision) คือ 0.72 โมเดล

(Model) VGG16 มีค่าความถูกต้อง (Accuracy) คือ 0.74 ค่าความแม่นยำ (Precision) คือ 0.74 โมเดล (Model) ResNet50V2 มีค่าความถูกต้อง (Accuracy) คือ 0.71 ค่าความแม่นยำ (Precision) คือ 0.74 และค่าที่ต่ำที่สุดคือ โมเดล (Model) Keras มีค่าความถูกต้อง (Accuracy) คือ 0.52 และค่าความแม่นยำ (Precision) คือ 0.54 ซึ่งเป็นค่าที่ต่ำที่สุดเมื่อเปรียบเทียบทั้ง 5 โมเดล (Model)

2. ผลการวิเคราะห์การเปรียบเทียบประสิทธิภาพของโมเดล (Model) เมื่อทำการทดสอบ (Test) ที่มีการฝึกโดยใช้การวัดประสิทธิภาพ (K-Fold Cross Validation) ทั้ง 5 โมเดล (Model) ได้แก่ โมเดล (Model) Keras โมเดล (Model) Mobilenetv2 โมเดล (Model) InceptionV3 โมเดล (Model) VGG16 และโมเดล (Model) Resnet50V2 แสดงให้เห็นว่าโมเดล (Model) Resnet50V2 มีค่าความถูกต้อง (Accuracy) มากที่สุดเมื่อเทียบทั้ง 5 โมเดลโดยมีค่าความถูกต้อง (Accuracy) คือ 0.89 ค่าความแม่นยำ (Precision) คือ 0.90 โมเดล (Model) MobilenetV2 มีค่าความถูกต้อง (Accuracy) คือ 0.87 ค่าความแม่นยำ (Precision) คือ 0.88 โมเดล (Model) VGG16 มีค่าความถูกต้อง (Accuracy) คือ 0.84 ค่าความแม่นยำ (Precision) คือ 0.85 โมเดล (Model) InceptionV3 มีค่าความถูกต้อง (Accuracy) คือ 0.82 ค่าความแม่นยำ (Precision) คือ 0.82 และค่าที่ต่ำที่สุดคือโมเดล (Model) Keras มีค่าความถูกต้อง (Accuracy) คือ 0.36 และค่าความแม่นยำ (Precision) คือ 0.35 ซึ่งเป็นค่าที่ต่ำที่สุดเมื่อเปรียบเทียบทั้ง 5 โมเดล (Model)

5.2 อภิปรายผล

การวิจัยเรื่อง ระบบตรวจจับและระบุประเภทไม้ประดับบอนสีด้วยการประมวลผลภาพ สามารถสรุปได้ ดังนี้

ในการศึกษานี้รูปภาพที่ใช้ในการวิจัยถูกปรับให้มีขนาด 224x224 พิกเซล โดยชุดข้อมูลถูกแบ่งด้วยสัดส่วน 80:20 เพื่อใช้เป็นชุดเรียนรู้ (Training Set) และข้อมูลชุดทดสอบ (Test Set) ตามลำดับ ใช้วิธี K-Fold Cross-validation กำหนดให้มีจำนวน K=5-Fold Cross-Validation ซึ่งสอดคล้องกับการการศึกษาของ จักรินทร์ สนุกแสน (2562) กับการใช้ K=5-Fold Cross-Validation โดยใช้ค่าความถูกต้อง (Accuracy) และค่าความผิดพลาด (Loss) สำหรับการประเมินประสิทธิภาพของการทดลอง โดยใช้ชุดข้อมูลในการทดลอง 1 ชุด ทั้งหมด 1,500 ภาพ แบ่งเป็นชุด Training set ประเภทละ 300 ภาพ เป็นทั้งหมดจำนวน 1,200 ภาพ และชุด Test set 75 ภาพ เป็นทั้งหมดจำนวน 300 ภาพ จะถูกนำมาใช้เพื่อทดสอบประสิทธิภาพของโมเดล (Model) ทั้ง 5 โมเดล ประกอบด้วยโมเดล โมเดล (Model) Mobilenetv2 โมเดล (Model) InceptionV3 โมเดล (Model) VGG16 และโมเดล (Model) Resnet50V2 ซึ่งเป็นเทคนิคที่ได้รับ ความนิยมในการใช้ทดสอบ

โดยโมเดล (Model) Keras ที่ผู้วิจัยสร้างขึ้นมีประสิทธิภาพต่ำ เนื่องจากมีข้อจำกัดในเรื่องของระยะเวลาที่ใช้ในการฝึก (Train) ทรัพยากรในส่วนของคอมพิวเตอร์มีความจำกัด การใช้ pre-trained Model จึงเป็นทางเลือกที่มีประสิทธิภาพในการจำแนกหรือระบุประเภทของบอนสีได้ดีกว่า และช่วยลดข้อจำกัดในเรื่องระยะเวลาในการฝึก (Train) และทรัพยากรคอมพิวเตอร์

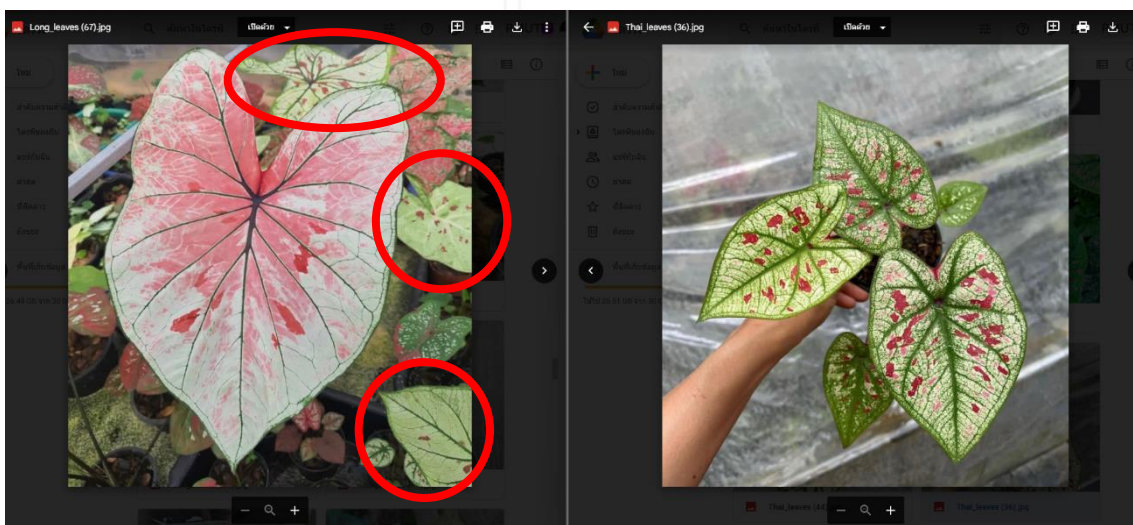
โดยโมเดล (Model) แรกคือโมเดล (Model) MobilenetV2 ซึ่งสอดคล้องกับงานวิจัยเรื่องการรู้จำการแสดงอารมณ์ออกทางใบหน้าด้วยวิธีโครงข่ายประสาทเทียมแบบคอนโวลูชันร่วมกับการประมวลผลภาพ 18 รูปแบบของพนเมฆ ญาณฐิติรัตน์ และคณะ (2561) มีประสิทธิภาพของค่าความถูกต้อง (Accuracy) อยู่ที่ 78.16 ของชุดข้อมูลภาพ Crop40-60 with Flip เมื่อเปรียบเทียบกับค่าความถูกต้อง (Accuracy) ของผู้วิจัยมีค่าความถูกต้องอยู่ที่ 74.67 ของการฝึกโดยไม่ใส่ K-Fold Cross Validation และค่าความถูกต้อง (Accuracy) อยู่ที่ 87.33 ของการฝึกโดยใส่ K-Fold Cross Validation

โมเดล (Model) ที่สองคือโมเดล (Model) InceptionV3 ซึ่งสอดคล้องกับงานวิจัยเรื่องการประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึกเพื่อวัดระดับความหวานของแตงโมผ่านสมาร์ตโฟน ของณัฐวดี หงษ์บุญมี และณัฐพงษ์ จันทะวงศ์ (2563) มีค่าความถูกต้อง 97.20 เรื่องการจำแนกผลึกน้ำตาลด้วยวิธีการเรียนรู้เชิงลึก ของสุริยะ ชยะธรรมกุล (2563) มีค่าความถูกต้อง 78.5 และเรื่องการจำแนกโรคพืชที่ปรากฏทางใบมะนาวด้วยวิธีการประมวลผลภาพบนสมาร์ตโฟน ระบบปฏิบัติการแอนดรอยด์ของนศพร ธรรมโชติ และคณะ (2563) มีค่าความถูกต้อง (Accuracy) 75.66 เมื่อเปรียบเทียบค่าความถูกต้อง (Accuracy) ของผู้วิจัยมีค่าความถูกต้อง 72 ของการฝึกโดยไม่ใส่ K-Fold Cross Validation และค่าความถูกต้อง (Accuracy) 81.99 ของการฝึกโดยใส่ K-Fold Cross Validation

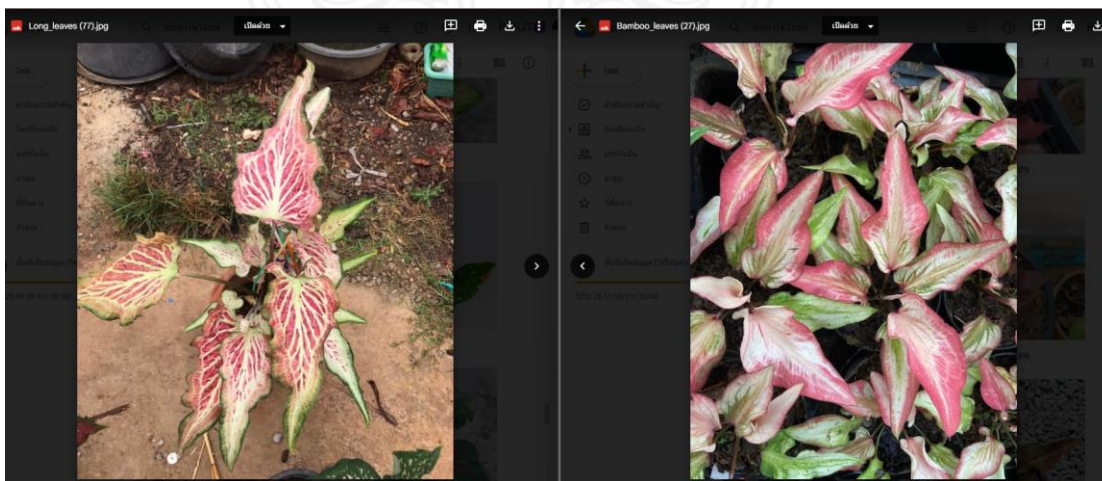
โมเดล (Model) ที่สามคือโมเดล (Model) VGG16 ซึ่งสอดคล้องกับงานวิจัยเรื่องการจำแนกผลึกน้ำตาลด้วยวิธีการเรียนรู้เชิงลึก ของสุริยะ ชยะธรรมกุล (2563) มีค่าความถูกต้อง 85.0 เรื่องการเปรียบเทียบวิธีการวิเคราะห์โรคใบมะนาวด้วยวิธีเรียนรู้เชิงลึก ของอุมาภรณ์ สายแสงจันทร์ และคณะ (2565) มีค่าความถูกต้อง (Accuracy) 86.30 เมื่อเปรียบเทียบค่าความถูกต้องของผู้วิจัยมีค่าความถูกต้อง (Accuracy) 73.67 ของการฝึกโดยไม่ใส่ K-Fold Cross Validation และค่าความถูกต้อง (Accuracy) 83.66 ของการฝึกโดยใส่ K-Fold Cross Validation

โมเดล (Model) ที่สี่คือโมเดล (Model) Resnet50V2 ซึ่งสอดคล้องกับงานวิจัยเรื่องการเปรียบเทียบวิธีการวิเคราะห์โรคใบมะนาวด้วยวิธีเรียนรู้เชิงลึกของอุมาภรณ์ สายแสงจันทร์ และคณะ (2565) มีค่าความถูกต้อง (Accuracy) 88.26 เมื่อเปรียบเทียบค่าความถูกต้องของผู้วิจัยมีค่าความถูกต้อง (Accuracy) 70.67 ของการฝึกโดยไม่ใส่ K-Fold Cross Validation และค่าความถูกต้อง (Accuracy) 88.66 ของการฝึกโดยใส่ K-Fold Cross Validation

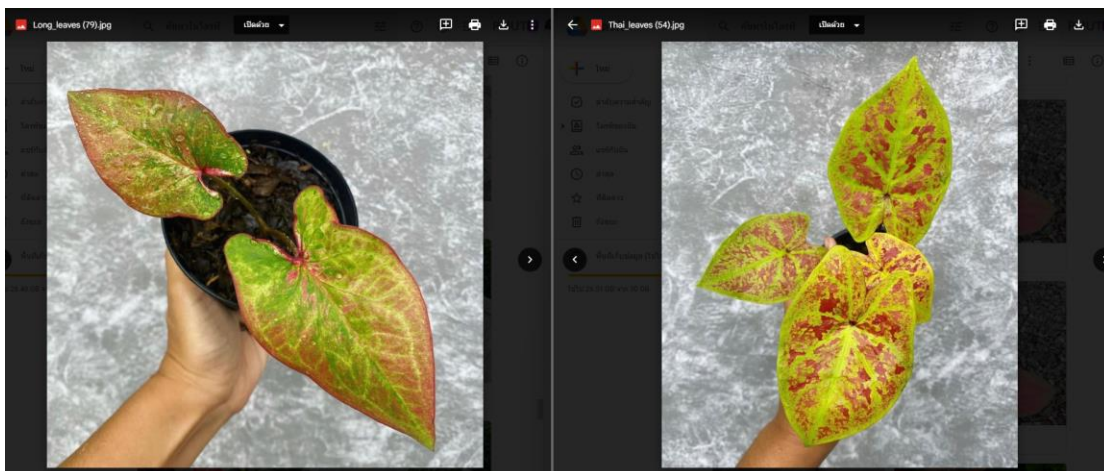
จากที่กล่าวมาข้างต้นประสิทธิภาพของโมเดลที่ได้ทำนายผลของประเภทไม้ประดับบอนสีที่มีค่าความถูกต้อง (Accuracy) สูงสุดคือโมเดล (Model) Resnet50V2 มีค่าความถูกต้อง (Accuracy) คือ 88.66 และจากทั้ง 4 โมเดล (Model) คลาสที่มีการทำนายผิดมากที่สุดคือ คลาสใบไม้ ทำนายผิดเป็น คลาสใบยาว เนื่องมาจากภาพของชุดข้อมูลทดสอบมีความใกล้เคียงกัน ทำให้การทำนายเกิดข้อผิดพลาดสูงดังภาพที่ 5-1



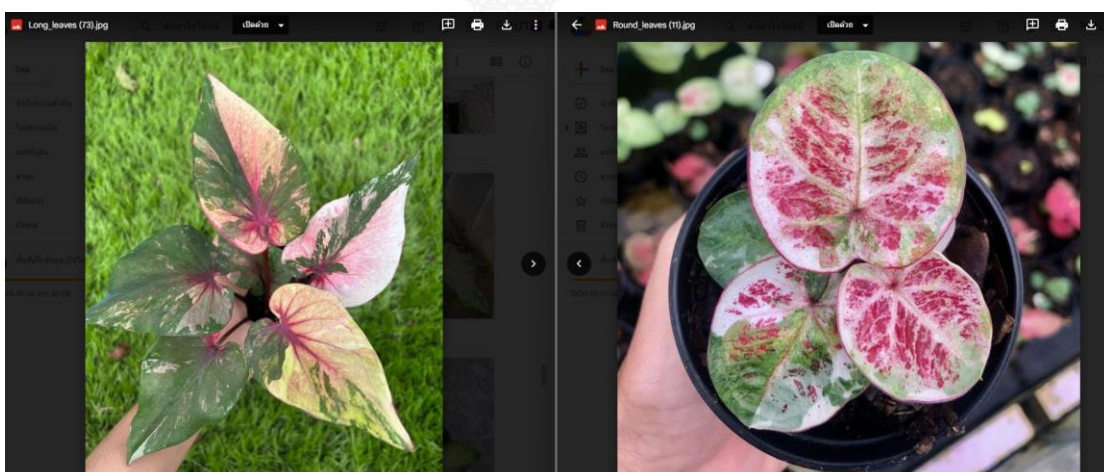
ภาพที่ 5-5 เปรียบเทียบชุดข้อมูลทดสอบ (Test) ระหว่างใบยาว (ด้านซ้าย) และใบไทย (ด้านขวา)



ภาพที่ 5-6 เปรียบเทียบชุดข้อมูลทดสอบ (Test) ระหว่างใบยาว (ด้านซ้าย) และใบไม้ (ด้านขวา)



ภาพที่ 5-7 เปรียบเทียบชุดข้อมูลทดสอบ (Test) ระหว่างใบยาว (ด้านซ้าย) และใบไทย (ด้านขวา)



ภาพที่ 5-8 เปรียบเทียบชุดข้อมูลทดสอบ (Test) ระหว่างใบยาว (ด้านซ้าย) และใบกลม (ด้านขวา)

จากภาพที่ 5-5 จะเห็นได้ว่าลักษณะของใบด้านหลังที่วงกลมสีแดงในภาพใบยาว มีความใกล้เคียงกับภาพของใบไทย ซึ่งส่งผลให้การทำนายประสิทธิภาพลดลง จากภาพที่ 5-6 จะเห็นได้ว่าใบยาว มีลักษณะที่ใกล้เคียงกับใบใฝ่มาก ภาพใบยาวและใบไทย จากภาพที่ 5-7 ก็ใกล้เคียงกันมากส่งผลให้การทำนายมีประสิทธิภาพลดลง ซึ่งแตกต่างจากภาพที่ 5-8 เนื่องจากใบกลม และใบยาวมีความแตกต่างกันชัดเจน ดังนั้นการทำนายผลจึงมีประสิทธิภาพมากกว่า

ใบยาวจะลักษณะใบทรงหัวใจคล้ายกับบอนใบไทยเช่นเดียวกัน แต่แตกต่างกันที่ใบของบอนสีใบยาวจะเรียวยาว และมีปลายใบแหลมกว่า หูใบยาวฉีกถึงสะดือ และสามารถแยกได้อีก 3 ลักษณะย่อยคือ บอนใบยาวธรรมดา บอนใบยาวรูปหอก และบอนใบยาวรูปใบไผ่ ส่งผลให้ตรงส่วนของบอนใบยาวรูปใบไผ่จะมักเข้าใจผิดว่าเป็นบอนใบไผ่ ซึ่งเป็นอีกหนึ่งสาเหตุที่ทำให้การประมวลผลภาพมีประสิทธิภาพลดลง อีกทั้งใบไทย และใบใฝ่ ที่ยังไม่เต็มที่จะมีลักษณะของใบใกล้เคียงกันมาก คือ ใบเรียวยาว หว่างหู และหูใบจะมีลักษณะเหมือนกัน ถ้าเป็นใบไทยจะยังลงไม่ถึงสะดือใบหาก

เรียกตามภาษาทั่วไป จะเรียกว่า “ใบยังไม่กัดสี” คือลักษณะของใบที่ยังไม่ชัดเจน สียังไม่สวย ขอบใบยังไม่ชัด

5.3 ข้อเสนอแนะ

5.3.1 ควรเก็บภาพหลาย ๆ มุม เพื่อที่เวลาทดสอบโมเดลจะได้มีประสิทธิภาพในการจำแนกมากขึ้น ซึ่งให้มีการถ่ายใบแบบหลากหลาย เช่น ใบที่เหี่ยว หรือใบที่ไม่สมบูรณ์

5.3.2 การจำแนกชนิดของประเภทใบนั้นเป็นเพียงต้นแบบของการจำแนกประเภทใบ เนื่องจากทรัพยากรเครื่องที่จำกัดจึงไม่สามารถนำข้อมูลจำนวนมากมาใช้งานได้

5.3.3 ในการศึกษาครั้งนี้ผู้จัดทำได้ศึกษาเกี่ยวกับระบบตรวจจับและระบุประเภทไม้ประดับบอนสีด้วยการประมวลผลภาพ โดยทางผู้จัดทำไม่ได้คำนึงถึงแสง และมุมของการถ่ายภาพใบ และความสัมพันธ์ของรูปภาพกับโมเดล ปัจจัยอื่นที่เกี่ยวข้อง ควรคำนึงถึงการถ่ายภาพแสง มุม และโมเดลที่ใช้เหมาะสม

5.3.4 โมเดล (Model) บางตัว อาจเหมาะกับรูปภาพประเภทของ Gray Scale และโมเดล (Model) บางตัว อาจเหมาะกับรูปภาพประเภทของ RGB

5.3.5 ควรปรับ Parameter ให้เหมาะสมกับโมเดล เพื่อเพิ่มประสิทธิภาพให้ดียิ่งขึ้น

5.4 ข้อจำกัด

5.4.1 การจำกัดการใช้ GPU ของ Google colabortory ทำให้ไม่สามารถ Runtime ได้ตลอดเวลา

5.4.2 รูปภาพของทางผู้จัดทำ อาจจะไม่เหมาะกับโมเดลที่ใช้ ซึ่งท่านที่สนใจสามารถนำไปพัฒนาเพื่อทำให้ประสิทธิภาพดีขึ้น

5.4.3 ควรกำหนดขนาดของรูปภาพตามข้อจำกัดของโมเดล เพื่อให้ค่าดีขึ้น

บรรณานุกรม

ภาษาไทย

- กวี ลิมอุสันโน. (2564). [ออนไลน์]. เทคนิคการปรับปรุงพันธุ์บอนสีโดยการผสมพันธุ์. [สืบค้นวันที่ 10 ตุลาคม 2565]. จาก http://www.sdoae.doe.go.th/News/g_product/1.pdf.
- จิตรพงษ์ เจริญจิตร และคณะ. (2561). ระบบแยกประเภทไข่มุกด้วยวิธีการประมวลผลภาพ. *การประชุมมหาดใหญ่วิชาการ ระดับชาติและนานาชาติ*, 9. 1,679-1,691.
- จักรินทร์ สุนกแสน. (2562). การประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึกสำหรับการรู้จำพรรณไม้ที่อยู่ในสิ่งแวดล้อมทางธรรมชาติ. (ปริญญาวิทยาศาสตรมหาบัณฑิต, มหาวิทยาลัยมหาสารคาม).
- ณัฐวดี หงส์บุญมี และณัฐพงศ์ จันทะวงศ์. (2563). การประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึกเพื่อวัดระดับความหวานของแตงโม. *JIST Journal of Information Science and Technology*, 1(10). 59-69.
- ณัฐวดี หงส์บุญมี และพิรยุทธ ชุนบุญ. (2564). การพัฒนาแอปพลิเคชันทำนายระดับความหวานของส้มโอผ่านสมาร์ทโฟนด้วยเทคนิคการเรียนรู้เชิงลึก. *RMUTSB Acad. J.* 9(1). 92-105.
- นพรุจ พัฒนสาร และณัฐวดี ศรีวิบูลย์. (2563). การประมวลผลภาพสำหรับการจำแนกคุณภาพมะม่วงพันธุ์โชคอนันต์โดยการจำลองการมองเห็นของมนุษย์ด้วยวิธีการเรียนรู้เชิงลึก. *RMUTSB Acad. J.* 9(1). 24-29.
- นศพร ธรรมโชติ, ชัยสิทธิ์ และปิติพัฒน์ บุตรโครต. (2563). การจำแนกโรคพืชที่ปรากฏทางใบมะนาวด้วยวิธีการประมวลผลภาพบนสมาร์ทโฟนระบบปฏิบัติการแอนดรอยด์. (มหาวิทยาลัยเทคโนโลยีราชมงคลศรีวิชัย).
- บอนสี Caladium. (2564). พิมพ์ครั้งที่ 1. กรุงเทพมหานคร : สำนักพิมพ์บ้านและสวน.
- พิชิตชัย พิมพ์โครต. (2564). การตรวจจับโลโก้สินค้าแอลกอฮอล์ด้วยแบบจำลองการเรียนรู้เชิงลึก. (วิทยานิพนธ์มหาบัณฑิต, มหาวิทยาลัยธุรกิจบัณฑิต).

บรรณานุกรม (ต่อ)

- พนเมษ ญาณฐิติรัตน์ และคณะ. (2561). การรู้จำการแสดงอารมณ์ออกทางใบหน้าด้วยวิธี
 โครงข่ายประสาทเทียมแบบคอนโวลูชันร่วมกับการประมวลผลภาพ 18 รูปแบบ
 Facial Expression Recognition using Convolutional Neural Networks Based
 on 18 Variations of Image Processing. วารสารเทคโนโลยีสารสนเทศ. ปีที่ 16.
 ฉบับที่ 1. 98-109
- รัตน์โชติ พันธุ์วิไล. (2562). การตรวจหาต้นไม้เป็นโรคโดยอัตโนมัติด้วยภาพถ่ายมุมสูงจากโดรน
 และวิธีการเรียนรู้เชิงลึก. (วิทยานิพนธ์มหาบัณฑิต, มหาวิทยาลัยธุรกิจ บัณฑิตย).
- ศิริเรือง พัฒน์ช่วย และอาทิตย์ อยู่เย็น. (2561). ระบบคัดแยกเกรดปลาหมึกแห้งสำหรับกลุ่ม
 ชาวประมงลุ่มน้ำชายฝั่งทะเลประจวบคีรีขันธ์ด้วยหลักการ Deep Learning.
 (มหาวิทยาลัยเทคโนโลยีราชมงคลรัตนโกสินทร์).
- สุริยะ ชยะธรรมกุล. (2563). การจำแนกผลึกน้ำตาลด้วยวิธีการเรียนรู้เชิงลึก. (สารนิพนธ์ปริญญา
 มหาบัณฑิต, จุฬาลงกรณ์มหาวิทยาลัย).
- อุมาภรณ์ สายแสงจันทร์, รพีพร ชำชอง และอรรณพ สุวรรณษา. (2565). การเปรียบเทียบวิธี
 วิเคราะห์โรคมะนาวด้วยการเรียนรู้เชิงลึก. *Journal of Applied Informatics and
 Technology*, 4(1). 71-86.
- อรวรรณ วิชัยลักษณ์. (2548). บอนสี. [ม.ป.ท.]: กรมส่งเสริมการเกษตร สำนักส่งเสริมและจัดการ
 สินค้าเกษตรส่วนส่งเสริมการผลิตผัก ไม้ดอกไม้ประดับ กลุ่มส่งเสริมการผลิตไม้ดอกไม้
 ประดับ.

ภาษาอังกฤษ

- Ling Yi Jun and Lau Phooi Yee. (2020). Comparison of Vision-based and CNN-based
 Classifiers for Fish Monitoring Environment. *ECTI TRANSACTIONS ON
 COMPUTER AND INFORMATION TECHNOLOGY*, 15(2). 232-244.

ไม่มีเนื้อหาจากต้นฉบับ



ประวัติผู้ดำเนินโครงการ



- ชื่อ** : นางสาวณัฐพร รอดสดใส
- ชื่อโครงการ** : ระบบตรวจจับและระบุประเภทไม้ประดับบอนสีด้วยการประมวลผลภาพ
- สาขาวิชา** : การวิเคราะห์ข้อมูลเชิงธุรกิจ คณะบริหารธุรกิจ
- ประวัติส่วนตัว**
- วัน-เดือน-ปีเกิด : วันอังคารที่ 29 พฤษภาคม พ.ศ. 2544
- ภูมิลำเนา : 55/30 ซอยพระรามที่ 2 ซอย 60 แขวงสามตำ
เขตบางขุนเทียน กรุงเทพมหานคร 10150
- ที่อยู่ปัจจุบัน : 183/4 ม.3 แขวงบางบอน เขตบางบอน กรุงเทพมหานคร
10150
- โทรศัพท์เคลื่อนที่ : 095-7499339
- อีเมล : nathaporn.rodsodsai@gmail.com
- ประวัติการศึกษา**
- ปีการศึกษา 2556 : ระดับมัธยมศึกษาตอนต้น โรงเรียนวัดราชโอรส
- ปีการศึกษา 2559 : ระดับมัธยมศึกษาตอนปลาย วิทยาลัยพณิชยการเซตุน
- ปีการศึกษา 2562 : ระดับปริญญาตรี สาขาวิชาการระบบสารสนเทศ
หลักสูตรการวิเคราะห์ข้อมูลเชิงธุรกิจ คณะบริหารธุรกิจ
มหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร

ประวัติผู้ดำเนินโครงการ



ชื่อ : นางสาวนลพรรณ อมรโชติพันธ์
ชื่อโครงการ : ระบบตรวจจับและระบุประเภทไม้ประดับบอนสีด้วยการประมวลผลภาพ
สาขาวิชา : การวิเคราะห์ข้อมูลเชิงธุรกิจ คณะบริหารธุรกิจ

ประวัติส่วนตัว

วัน-เดือน-ปีเกิด : วันจันทร์ที่ 23 เมษายน พ.ศ. 2544
ภูมิลำเนา : 43/136 ม.4 ต.คลองพระอุดม อ.ปากเกร็ด จ.นนทบุรี 11120
ที่อยู่ปัจจุบัน : 43/136 ม.4 ต.คลองพระอุดม อ.ปากเกร็ด จ.นนทบุรี 11120
โทรศัพท์เคลื่อนที่ : 097-2126239
อีเมล : gam2544_@hotmail.co.th

ประวัติการศึกษา

ปีการศึกษา 2556 : ระดับมัธยมศึกษาตอนต้น โรงเรียนเตรียมอุดมศึกษาน้อมเกล้า นนทบุรี
ปีการศึกษา 2559 : ระดับมัธยมศึกษาตอนปลาย โรงเรียนปากเกร็ด
ปีการศึกษา 2562 : ระดับปริญญาตรี สาขาวิชาการระบบสารสนเทศ
หลักสูตรการวิเคราะห์ข้อมูลเชิงธุรกิจ คณะบริหารธุรกิจ
มหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร

ประวัติผู้ดำเนินโครงการ



ชื่อ : นางสาวเพชรชริน สุขสม
ชื่อโครงการ : ระบบตรวจจับและระบุประเภทไม้ประดับบอนสีด้วยการประมวลผลภาพ
สาขาวิชา : การวิเคราะห์ข้อมูลเชิงธุรกิจ คณะบริหารธุรกิจ

ประวัติส่วนตัว

วัน-เดือน-ปีเกิด : วันพฤหัสบดีที่ 28 ธันวาคม 2543
ภูมิลำเนา : 943/14 ซอยลาดพร้าว 87 แขวงคลองจั่น เขตบางกะปิ กรุงเทพมหานคร 10240
ที่อยู่ปัจจุบัน : 77 ซอยลาดพร้าว 94 แขวงพลับพลา เขตวังทองหลาง กรุงเทพมหานคร 10310
โทรศัพท์เคลื่อนที่ : 098-937-4494
อีเมล : petcharin.suksom@gmail.com

ประวัติการศึกษา

ปีการศึกษา 2556 : ระดับมัธยมศึกษาตอนต้น โรงเรียนเกษมพิทยา
ปีการศึกษา 2559 : ระดับมัธยมศึกษาตอนปลาย โรงเรียนเกษมพิทยา
ปีการศึกษา 2562 : ระดับปริญญาตรี สาขาวิชาระบบสารสนเทศ
หลักสูตรการวิเคราะห์ข้อมูลเชิงธุรกิจ คณะบริหารธุรกิจ
มหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร