

การจำแนกโรคใบพืชโดยใช้อัลกอริทึมวีจีจี10-อินเซปชันวีทรี่ที่มีชั้นสำหรับลด  
จำนวนพารามิเตอร์

PLANT LEAF DISEASE CLASSIFICATION USING VGG10 – INCEPTIONV3  
WITH PARAMETER REDUCTION LAYER ALGORITHM



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญาปรัชญาดุษฎีบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์  
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2566

KMITL-2023-SC-D-002-015

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PLANT LEAF DISEASE CLASSIFICATION USING VGG10 – INCEPTIONV3  
WITH PARAMETER REDUCTION LAYER ALGORITHM



A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE  
DEGREE OF DOCTOR OF PHILOSOPHY IN COMPUTER SCIENCE  
DEPARTMENT OF COMPUTER SCIENCE SCHOOL OF SCIENCE  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2023

KMITL-2023-SC-D-002-015

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2023

SCHOOL OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การจำแนกโรคใบพืชโดยใช้อัลกอริทึมวีจีจี10-อินเซปชันวีทรี่ที่มีชั้นสำหรับลดจำนวนพารามิเตอร์
ชื่อนักศึกษา	นางสาวจิราภรณ์ ถมแก้ว
รหัสประจำตัว	63605141
ปริญญา	ปรัชญาดุษฎีบัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชา	วิทยาการคอมพิวเตอร์
พ.ศ.	2566
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ผู้ช่วยศาสตราจารย์.ดร.ศรัณย์ อินทโกสุม

### บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อ 1) เปรียบเทียบประสิทธิภาพการจำแนกใบพืชโดยใช้การกำหนดจุดเด่นบนขอบใบด้วย Scale-Invariant Feature Transform (SIFT) 2) เพื่อนำเสนอแบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer สำหรับจำแนกโรคของใบพืช และ 3) เพื่อเปรียบเทียบประสิทธิภาพ จำนวนพารามิเตอร์และเวลาในการประมวลผลของแบบจำลองที่นำเสนอ กับแบบจำลองอื่น ๆ ของ Convolutional Neural Network เช่น VGG16, InceptionV3, ResNet50, DenseNet121 และ MobileNetV2 แบบจำลองที่นำเสนอปรับปรุงบนพื้นฐานของแบบจำลอง VGG16 และ InceptionV3 มีการเพิ่มชั้นสำหรับการลดจำนวนพารามิเตอร์ของแบบจำลอง InceptionV3 โดยเพิ่มชั้นคอนโวลูชันของ InceptionV3 Block จาก 3 ชั้นเป็น 4 ชั้นเพื่อให้แบบจำลองมีความกว้างมากขึ้น และสามารถสกัดคุณสมบัติของรูปภาพแบบขนานพร้อมกัน รวมทั้งมีการเปลี่ยนขนาดของตัวกรองจาก  $3 \times 3$  เป็น  $1 \times 3$  และ  $3 \times 1$  ซึ่งการเปลี่ยนขนาดของตัวกรองช่วยให้สามารถลดจำนวนพารามิเตอร์และเวลาการทำงานของแบบจำลองลงได้ โดยแบบจำลอง InceptionV3 with Parameter Reduction Layer ที่นำเสนอสามารถลดจำนวนพารามิเตอร์ได้ถึงร้อยละ 41.58 จาก InceptionV3 Block แบบเดิม จากนั้นนำ InceptionV3 with Parameter Reduction Layer ใช้ร่วมกับแบบจำลอง VGG16 โดยการเพิ่มระหว่างชั้นที่ 4 และ 5 ของ VGG16 และปรับลดจำนวนชั้นของ VGG16 จาก 16 ชั้น เป็น 10 ชั้น เมื่อปรับลดความลึกของแบบจำลองจึงส่งผลให้จำนวนพารามิเตอร์และเวลาประมวลผลของแบบจำลองลดลง แบบจำลองที่นำเสนอสามารถลดพารามิเตอร์ได้ร้อยละ 87.20 จาก VGG16 และร้อยละ 91.38 จาก InceptionV3 โดยทำการทดลองกับชุดข้อมูลใบพืชจาก PlantVillage Dataset ผลการทดลองพบว่าแบบจำลองที่นำเสนอให้ค่าความถูกต้องของการจำแนกโรคใบพืชร้อยละ 99.33 สูงที่สุดเมื่อเปรียบเทียบกับแบบจำลอง CNN อื่นๆ รวมทั้งยังใช้จำนวนพารามิเตอร์และหน่วยความจำน้อยกว่าแบบจำลอง

VGG16 และ InceptionV3 และเมื่อเปรียบเทียบเวลาการทำงานกับแบบจำลองอื่นพบว่ายังคงใช้เวลาการทำงานมากกว่าแบบจำลอง MobileNetV2 จึงเป็นแนวทางสำหรับการพัฒนาในอนาคตให้แบบจำลองสามารถลดเวลาการทำงานให้เร็วขึ้นแต่ยังคงมีค่าความถูกต้องของการจำแนกข้อมูลที่สูง ดังนั้นจึงสรุปได้ว่าแบบจำลองที่นำเสนอเป็นแบบจำลองโครงข่ายประสาทเทียมคอนโวลูชันขนาดเล็กที่ให้ค่าความถูกต้องของการจำแนกข้อมูลสูงที่สามารถนำไปประยุกต์ใช้งานบนอุปกรณ์ขนาดเล็กหรืออุปกรณ์ที่มีข้อจำกัดด้านทรัพยากรณได้

**คำสำคัญ :** การจำแนกโรคใบพืช ชั้นสำหรับลดพารามิเตอร์ วิจิจิ10 วิจิจิ16 อินเซปชันวิทรี



Thesis Title            Plant Leaf Disease Classification using VGG10 - InceptionV3  
with Parameter Reduction Layer Algorithm

Student Name           Miss Jiraporn Thomkaew

Student ID             63605141

Degree                 Doctor of Philosophy (Computer Science)

Department            Computer Science

Year                     2023

Thesis Advisor        Asst. Prof. Dr. Sarun Intakosum

### Abstract

This research aims to 1) compare plant leaf classification efficiency by using the key point on the leaf edge with Scale-Invariant Feature Transform (SIFT). 2) To present the VGG10 – InceptionV3 with Parameter Reduction Layer model for classifying plant leaf diseases, and 3) to compare the performance, the number of parameters, and execution time of the model presented to other convolutional neural networks models such as VGG16, InceptionV3, ResNet50, DenseNet121, and MobileNetV2. The proposed model is based on the VGG16 and InceptionV3 models. Added a layer for reducing the number of parameters of the InceptionV3 model by increasing the convolution layer of the InceptionV3 Block from 3 to 4 layers so that the model has more width and can simultaneously extract features of parallel images. The filter size has also been changed from 3 x 3 to 1 x 3 and 3 x 1. Changing the filter size can reduce the number of parameters and model run time. The proposed InceptionV3 with Parameter Reduction Layer can reduce the number of parameters by 41.58% over the base InceptionV3 Block. Next, apply InceptionV3 with Parameter Reduction Layer to the VGG16 model by adding between layers 4th and 5th of VGG16 and reducing the number of layers of VGG16 from 16 layers to 10 layers. As the model depth is reduced, the number of parameters and the model processing time are reduced. The proposed model has a parameter reduction of 87.20% from VGG16 and 91.38% from InceptionV3. Experiment with the plant leaf dataset from the PlantVillage Dataset. The experimental results showed that the proposed model gave the highest plant leaf disease classification accuracy of 99.33% compared to other CNN

ค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

models. It also uses fewer parameters and memory than the VGG16 and InceptionV3 models. And when comparing the run time with other models, it still takes more run time than the MobileNetV2 model. This is a guideline for future developments in which the model can reduce the run time faster but still have a high data classification accuracy. Therefore, it can be concluded that the proposed model is a small convolutional neural network model that provides high accuracy in disease classification of plant leaves, which can be applied to small devices or devices with limited resources.

**Keywords :** Plant Leaf Diseases Classification, Parameter Reduction Layer, VGG10, VGG16, InceptionV3



## กิตติกรรมประกาศ

วิทยานิพนธ์เล่มนี้สำเร็จได้ด้วยดี ข้าพเจ้าขอกราบขอบพระคุณ ผศ.ดร.ศรัณย์ อินทโกสุม อาจารย์ที่ปรึกษาวิทยานิพนธ์ที่ท่านได้สละเวลาให้ความรู้ ให้คำปรึกษา ตรวจสอบ และแนะนำแนวทางการวิจัยจนทำให้วิจัยในครั้งนี้สำเร็จด้วยดี ข้าพเจ้าจึงขอกราบระลึกถึงพระคุณของอาจารย์ไว้ ณ ที่นี้

ขอกราบขอบพระคุณ รองศาสตราจารย์ ดร. อนุชิต จิตพัฒนกุล ซึ่งให้ความกรุณาเป็นประธานกรรมการจากภายนอก ทำให้ได้รับคำแนะนำ คำปรึกษาต่างๆ ที่ทำให้การทำวิทยานิพนธ์นี้สมบูรณ์ยิ่งขึ้น

ขอกราบขอบพระคุณ ผศ.ดร. อนันตพร หารัชคุณาฒย์ ผศ.ดร. นवलสวาท หิรัญสกุลวงศ์ และ ผศ.ดร. ปัทมา เจริญพร กรรมการสอบวิทยานิพนธ์ ที่กรุณาเสียสละเวลา ให้คำแนะนำ ตรวจสอบ และแก้ไขวิทยานิพนธ์ฉบับนี้

ขอกราบขอบพระคุณ คุณพ่อคุณแม่และครอบครัว ที่คอยสนับสนุน ช่วยเหลือและเป็นกำลังใจที่สำคัญยิ่งตลอดเวลาการทำวิทยานิพนธ์ รวมถึงท่านอื่นๆ ที่มีได้กล่าวถึงไว้ ณ ที่นี้ ที่มีส่วนช่วยให้วิทยานิพนธ์ของข้าพเจ้าสำเร็จไปได้ด้วยดี

ท้ายสุดขอขอบพระคุณ มหาวิทยาลัยเทคโนโลยีราชมงคลศรีวิชัย ที่ให้โอกาสข้าพเจ้าได้ลาศึกษาต่อเพื่อพัฒนาความรู้ในวิชาชีพ และขอขอบพระคุณสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่ให้โอกาสได้เข้าศึกษาและประสิทธิ์ประสาทวิชาที่จำเป็นต่อวิชาชีพ

นางสาวจิราภรณ์ ถมแก้ว

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ค
กิตติกรรมประกาศ	จ
สารบัญ	ฉ
สารบัญตาราง	ญ
สารบัญรูปภาพ	ณ
<b>บทที่ 1 บทนำ</b>	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 สมมติฐานการวิจัย	3
1.3 วัตถุประสงค์ของงานวิจัย	3
1.4 ขอบเขตของงานวิจัย	4
1.5 ประโยชน์ที่คาดว่าจะได้รับ	5
<b>บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง</b>	6
2.1 โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks: CNN)	6
2.2 VGGNet	14
2.3 InceptionV3	14
2.4 MobileNet	16
2.5 DenseNet121	21
2.6 ResNet50	23
2.7 การสกัดคุณสมบัติรูปภาพ (Feature Extraction)	26
2.7.1 คุณสมบัติสี (Color Feature)	26
2.7.2 คุณสมบัติรูปร่าง (Shape Feature)	28
2.7.3 คุณสมบัติพื้นผิว (Texture Feature)	30
2.8 การประมวลผลรูปภาพ (Image Processing)	30
2.9 การจำแนกรูปภาพ (Image Classification)	33

๑

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ (ต่อ)

	หน้า
2.10 งานวิจัยที่เกี่ยวข้อง	34
<b>บทที่ 3 วิธีดำเนินการวิจัย</b>	41
3.1 การเตรียมชุดข้อมูลใบพืช (Preparation of Plant Leaf Dataset)	41
3.2 การประมวลผลรูปภาพใบพืช (Plant Leaf Image Processing)	45
3.2.1 การแปลงค่าสี (Color Transform)	45
3.2.2 การเปลี่ยนแปลงทางสัณฐานวิทยา (Morphological Transformations)	45
3.2.3 การตรวจจับขอบภาพ (Edge Detection)	46
3.2.4 การหาจุดเด่นบนภาพด้วย Scale-Invariant Feature Transform (SIFT)	46
3.3 การสกัดคุณสมบัตินี้และจำแนกข้อมูลใบพืช	46
3.4 สถาปัตยกรรมโครงข่ายประสาทเทียม VGG10 – InceptionV3 with Parameter Reduction Layer	47
3.4.1 สถาปัตยกรรมโครงข่ายประสาทเทียม InceptionV3 with Parameter Reduction Layer	48
3.4.2 สถาปัตยกรรมโครงข่ายประสาทเทียม VGG10 – InceptionV3 with Parameter Reduction Layer	49
3.5 การประเมินผล	50
<b>บทที่ 4 ผลการดำเนินการวิจัย</b>	53
4.1 การจำแนกใบพืชโดยใช้การกำหนดจุดเด่นบนขอบใบด้วย Scale-Invariant Feature Transform (SIFT)	53
4.2 การจำแนกโรคของใบพืชด้วยสถาปัตยกรรม VGG10 – InceptionV3 with Parameter Reduction Layer	61
4.3 เปรียบเทียบจำนวนพารามิเตอร์ของสถาปัตยกรรม VGG10 – InceptionV3 with Parameter Reduction Layer	74

## สารบัญ (ต่อ)

	หน้า
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ	76
5.1 สรุปผลการวิจัย	76
5.2 ข้อจำกัดและข้อเสนอแนะของงานวิจัย	78
5.3 แนวทางการประยุกต์ใช้ผลงานวิจัย	79
เอกสารอ้างอิง	80
ภาคผนวก ก: ผลงานวิจัยที่ตีพิมพ์	86
ภาคผนวก ข: ประวัติผู้เขียน	87



# สารบัญตาราง

ตารางที่	หน้า
3.1 รายละเอียดชุดข้อมูลจาก PlantVillage Dataset	43
3.2 พารามิเตอร์สำหรับการทดลองของแบบจำลองที่นำเสนอ	50
3.3 การประเมินผลด้วย Confusion Matrix	51
4.1 ผลการเปรียบเทียบค่าความถูกต้องการจำแนกข้อมูล	57
4.2 ผลการเปรียบเทียบค่าความถูกต้องการจำแนกข้อมูลของวิธีการที่นำเสนอกับแบบจำลองอื่นของ CNN	59
4.3 เปรียบเทียบค่าความถูกต้องจากการปรับลดจำนวนชั้นของแบบจำลอง VGG16 ที่ใช้ร่วมกับ InceptionV3 with Parameter Reduction Layer	62
4.4 เปรียบเทียบค่าความถูกต้องของการจำแนกโรคใบมะเขือเทศระหว่างแบบจำลองที่นำเสนอกับแบบจำลองของ Convolution Neural Network	64
4.5 ผลค่าความถูกต้องเฉลี่ยของการจำแนกโรคใบพืช 38 คลาสด้วยแบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer	65
4.6 ค่าความถูกต้องแยกตามคลาสของการจำแนกโรคใบพืช 38 คลาสด้วยแบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer	67
4.7 เปรียบเทียบค่าความถูกต้องของการจำแนกโรคใบพืช 38 คลาสด้วยแบบจำลองที่นำเสนอกับแบบจำลองของ Convolution Neural Network	68
4.8 เปรียบเทียบค่าความถูกต้องของแบบจำลองที่นำเสนอกับผลงานวิจัยที่เกี่ยวข้อง	69
4.9 เปรียบเทียบค่าความถูกต้องของแบบจำลองที่นำเสนอกับการใช้ชั้น Global Average Pooling	71
4.10 เปรียบเทียบเวลาการทำงานแต่ละรอบของแบบจำลองที่นำเสนอกับการใช้ชั้น Global Average Pooling	72
4.11 จำนวนพารามิเตอร์ของ InceptionV3 Block และ InceptionV3 with Parameter Reduction Layer	75

# สารบัญรูป

รูปที่	หน้า
2.1 สถาปัตยกรรมโครงข่ายประสาทเทียมแบบคอนโวลูชัน	7
2.2 การทำ Convolutional Feature	8
2.3 การเลื่อนของตัวกรองที่มีค่าการเลื่อนเท่ากับ 1	8
2.4 การขยายขอบของภาพ (Padding)	9
2.5 การรวมผลลัพธ์โดยใช้ค่ามากที่สุด (Max Pooling)	9
2.6 การรวมผลลัพธ์โดยใช้ค่าเฉลี่ย (Average Pooling)	10
2.7 โครงข่ายประสาทเทียมแบบ VGGNet	11
2.8 จำนวนพารามิเตอร์ของโครงข่ายประสาทแบบ VGGNet (หน่วยเป็นล้านพารามิเตอร์)	12
2.9 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ VGG16	13
2.10 สถาปัตยกรรมพื้นฐานของโครงข่ายแบบ Inception	14
2.11 สถาปัตยกรรมของโมดูล Inception	15
2.12 สถาปัตยกรรมของโมดูล InceptionV3	16
2.13 การทำงานของชั้นคอนโวลูชันแบบเดียวกับชั้นคอนโวลูชันของ MobileNet	17
2.14 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ MobileNet	18
2.15 สถาปัตยกรรมโมดูล Inverted Residual Structure ของโครงข่าย MobileNetV2	19
2.16 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ MobileNetV2	20
2.17 โครงสร้างการทำงานภายในของ Dense Block	21
2.18 โครงข่ายประสาทเทียมแบบ DenseNet	22
2.19 สถาปัตยกรรมชั้นถ่วงน้ำหนักของโครงข่ายประสาทเทียมแบบ ResNet34	24
2.20 โครงข่ายประสาทเทียมแบบ ResNet34	24
2.21 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ ResNet50	25
3.1 ตัวอย่างใบพืชที่ใช้สำหรับการทดลองจากชุดข้อมูล PlantVillage	42
3.2 ตัวอย่างใบพืชที่ใช้สำหรับการทดลองตามข้อสมมติฐานที่หนึ่ง	44
3.3 กรอบดำเนินงานของการวิจัยตามข้อสมมติฐานที่หนึ่ง	44
3.4 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ Sequential Model	47

ญ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.5 สถาปัตยกรรมของ InceptionV3 with Parameter Reduction Layer	48
3.6 สถาปัตยกรรมโครงข่ายประสาทเทียม VGG10 – InceptionV3 with Parameter Reduction Layer	49
4.1 ผลการแปลงค่าสีของรูปภาพเป็นภาพระดับสีเทาและภาพไบนารี	54
4.2 ผลการเปลี่ยนแปลงลักษณะทางสัณฐานวิทยาของภาพใบพีช	55
4.3 การตรวจจับขอบรูปภาพใบพีชด้วย Canny	55
4.4 การหาจุดเด่นบนภาพด้วย Scale-Invariant Feature Transform (SIFT)	56
4.5 Confusion Matrix ของการจำแนกใบพีชโดยใช้การกำหนดจุดเด่นบนขอบใบด้วย SIFT	58
4.6 ค่าความถูกต้องของการจำแนกข้อมูลด้วยชุดข้อมูลสำหรับการเรียนรู้ของแบบจำลอง และชุดข้อมูลสำหรับการตรวจสอบ	60
4.7 ค่าฟังก์ชันการสูญเสียของการจำแนกข้อมูลด้วยชุดข้อมูลสำหรับการเรียนรู้ของแบบจำลองและชุดข้อมูลสำหรับการตรวจสอบ	60
4.8 Confusion Matrix การจำแนกโรคใบมะเขือเทศด้วย VGG10 – InceptionV3 with Parameter Reduction Layer	63
4.9 Confusion Matrix การจำแนกโรคใบพีช 38 คลาส ด้วย VGG10 – InceptionV3 with Parameter Reduction Layer	66

ฎ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

ด้วยสภาพภูมิอากาศที่เปลี่ยนแปลงไปอย่างรวดเร็วย่อมส่งผลกระทบต่อพืชผลทางการเกษตรให้ได้รับความเสียหาย หากเกษตรกรรับมือกับผลกระทบดังกล่าวไม่ทันเวลาก็ส่งผลให้เกิดความเสียหายในวงกว้างได้ ทำให้ผลผลิตไม่เป็นไปตามมาตรฐานและความต้องการของตลาด ดังนั้นปัจจุบันจึงมีนักวิจัยจำนวนมากจากหลายศาสตร์ความรู้พยายามพัฒนางานวิจัยและเทคโนโลยีใหม่เพื่อช่วยแก้ปัญหาให้กับเกษตรกรสามารถรับรู้และแก้ปัญหาโรคของพืชได้อย่างรวดเร็วและแม่นยำ หนึ่งในเทคโนโลยีที่กำลังเป็นที่นิยมสำหรับการจำแนกโรคของพืชคือการประมวลผลภาพด้วยคอมพิวเตอร์ การจำแนกรูปภาพด้วยคอมพิวเตอร์กำลังได้รับความนิยมและนำมาประยุกต์ใช้กับงานที่หลากหลายมากขึ้น โดยการพัฒนาให้คอมพิวเตอร์สามารถมองเห็นรูปภาพได้เหมือนกับมนุษย์ และสอนให้คอมพิวเตอร์เรียนรู้รูปภาพจนสามารถวิเคราะห์และจดจำรูปภาพและสามารถจำแนกรูปภาพได้เสมือนกับมนุษย์ โดยการประมวลผลรูปภาพด้วยคอมพิวเตอร์สามารถนำไปใช้กับการเรียนรู้ของเครื่อง (Machine Learning) เพื่อให้คอมพิวเตอร์สามารถประมวลผลรูปภาพจำนวนมากๆ ได้อย่างแม่นยำและรวดเร็ว เช่นเดียวกับการวิเคราะห์โรคของพืชที่จำเป็นต้องอาศัยการจำแนกรูปภาพด้วยคอมพิวเตอร์เข้ามาช่วยเพราะสามารถจำแนกรูปภาพได้อย่างแม่นยำและรวดเร็ว

คอมพิวเตอร์วิทัศน์ (Computer vision) เป็นหนึ่งในเทคโนโลยีของปัญญาประดิษฐ์ (Artificial Intelligence) ที่พยายามสอนให้อัลกอริทึมของคอมพิวเตอร์ให้สามารถวิเคราะห์ จำแนก และตรวจจับรูปภาพได้ด้วยตนเองโดยใช้โครงข่ายประสาทเทียมและการเรียนรู้เชิงลึกหลายๆ ชั้นทำงานร่วมกัน ให้สามารถตรวจจับและจำแนกรูปภาพได้ทั้งที่เป็นภาพนิ่ง ภาพจากวิดีโอ รวมไปถึงภาพแบบเรียลไทม์ ซึ่งงานวิจัยในปัจจุบันที่นำเสนอแนวคิดเกี่ยวกับคอมพิวเตอร์วิทัศน์เพื่อเพิ่มประสิทธิภาพการทำงานให้แม่นยำมากขึ้น เช่นแนวคิดเกี่ยวกับการประมวลผลรูปภาพ การแบ่งส่วนรูปภาพ การสกัดคุณสมบัติของรูปภาพ และการจำแนกและจัดกลุ่มรูปภาพ โดยได้มีการนำเอาเทคโนโลยีคอมพิวเตอร์วิทัศน์ มาใช้กันอย่างแพร่หลาย เช่น การวินิจฉัยโรคจากภาพถ่ายทางการแพทย์ [1],[2],[3],[4] การจำแนกภาษามือ [5] การรู้จำอารมณ์บนใบหน้า [6] รวมไปถึงการตรวจจับและระบุโรคของใบพืชต่างๆ เช่น การระบุโรคของใบข้าวโพด [7] ใบพริกหวาน [8] ใบมะม่วง [9] ใบข้าว [10] เป็นต้น นอกจากนี้ยังมีการวิจัยเกี่ยวกับการตรวจจับและระบุโรคของพืชแบบเรียลไทม์ [11], [12], [13] โดยการพัฒนาแบบจำลองให้สามารถใช้

งานบนอุปกรณ์พกพาที่มีข้อจำกัดด้านทรัพยากรให้สามารถใช้งานได้ง่ายและสะดวกมากขึ้น นอกจากนี้ยังมีการพัฒนาให้สามารถใช้งานบนอุปกรณ์พกพาแบบไม่ต้องเชื่อมต่อสัญญาณอินเทอร์เน็ต [14] ก็ยังสามารถทำงานได้ ดังจะเห็นได้ว่าปัจจุบันการพัฒนาหรือปรับปรุงแบบจำลองของโครงข่ายประสาทเทียม (Convolutional Neural Network: CNN) จะมุ่งเน้นเรื่องของการลดจำนวนพารามิเตอร์และต้นทุนการคำนวณของแบบจำลอง เนื่องจากแบบจำลองของโครงข่ายประสาทเทียมในรุ่นก่อนเป็นเครือข่ายเชิงลึกขนาดใหญ่ที่มีหลายชั้นและจำนวนพารามิเตอร์ที่ใช้สำหรับการคำนวณจำนวนมากทำให้ต้องใช้ทรัพยากรของเครื่องจำนวนมากเช่นกันจึงไม่เหมาะสำหรับการใช้งานบนอุปกรณ์พกพาที่มีข้อจำกัดด้านทรัพยากร โครงข่ายประสาทเทียมน้ำหนักเบา (Lightweight Convolutional Neural Network) เป็นโมเดลขนาดเล็กที่พัฒนาเพื่อให้สามารถรองรับการใช้งานบนอุปกรณ์พกพาหรืออุปกรณ์ขนาดเล็กที่มีข้อจำกัดด้านทรัพยากร โดยแบบจำลองโครงข่ายประสาทเทียมน้ำหนักเบาจะปรับลดจำนวนชั้นของแบบจำลองและจำนวนพารามิเตอร์ที่ใช้คำนวณของแบบจำลองให้มีขนาดเล็กลงเพื่อลดเวลาและต้นทุนในการประมวลผล แต่ให้ยังคงมีประสิทธิภาพด้านการทำงานที่ดีกว่าเดิม Bouguettaya Abdelmalek และคณะ [15] ได้อธิบายเกี่ยวกับโครงข่ายประสาทเทียมน้ำหนักเบา เช่น MobileNet, MobileNetV2, SqueezeNet, ShuffleNet และ PeleeNet แบบจำลองเหล่านี้ใช้จำนวนพารามิเตอร์และต้นทุนการคำนวณที่น้อยลงและสามารถทำงานได้เร็วขึ้นโดยที่ประสิทธิภาพด้านค่าความแม่นยำก็ยังคงสูงเมื่อเปรียบเทียบกับแบบจำลองโครงข่ายประสาทเทียมแบบเดิม

ในงานวิจัยนี้ได้นำเสนอแบบจำลองโครงข่ายประสาทเทียม VGG10 – InceptionV3 with Parameter Reduction Layer มีวัตถุประสงค์เพื่อเพิ่มประสิทธิภาพการจำแนกโรคใบพืชด้วยแบบจำลองโครงข่ายประสาทเทียมที่มีขนาดเล็กที่สามารถลดจำนวนพารามิเตอร์ของแบบจำลองและลดต้นทุนด้านการคำนวณให้แบบจำลองสามารถใช้งานบนอุปกรณ์ขนาดเล็กได้แต่ยังคงให้ค่าความแม่นยำของการจำแนกโรคที่สูง โดยนำเสนอแบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer ที่ใช้แบบจำลองพื้นฐานของ VGG16 และ InceptionV3 โดยมีการปรับปรุง InceptionV3 with Parameter Reduction Layer สำหรับลดพารามิเตอร์ของ InceptionV3 Block โดยการเพิ่มชั้นคอนโวลูชัน (Convolutional layer) สำหรับการรับภาพนำเข้าจาก 3 ชั้นเป็น 4 ชั้น เพื่อให้แบบจำลองกว้างขึ้นและปรับขนาดตัวกรอง (Filter Size) ของชั้นคอนโวลูชัน จาก  $3 \times 3$  เป็น  $1 \times 3$  และ  $3 \times 1$  แทนเพื่อช่วยให้สามารถลดจำนวนพารามิเตอร์และเวลาการทำงานของแบบจำลอง จากนั้นเพิ่ม InceptionV3 with Parameter Reduction Layer ใช้งานร่วมกับแบบจำลองของ VGG16 ที่มีทั้งหมด 16 ชั้น โดยเพิ่มระหว่างชั้นที่ 4 และ 5 จากนั้นจึงปรับลดจำนวนชั้นของ VGG16 ทีละชั้นเพื่อพิจารณาค่าความถูกต้องของ

การจำแนกโรคใบพืช จำนวนพารามิเตอร์ เวลาประมวลผล และค่า F1 Score ที่ดีที่สุด ดังนั้นจึงลดจำนวนชั้นจาก 16 ชั้นเหลือเพียง 10 ชั้น จากนั้นทำการทดลองแบบจำลองที่นำเสนอกับการจำแนกโรคของใบพืช โดยใช้ภาพใบพืชจากชุดข้อมูล PlantVillage จำนวน 38 คลาส (Class) เป็นใบพืชที่เป็นโรคและใบพืชที่สุขภาพดีของพืชที่แตกต่างกัน 14 ชนิด ซึ่งวิธีการที่นำเสนอ InceptionV3 with Parameter Reduction Layer สามารถลดพารามิเตอร์ของแบบจำลอง InceptionV3 Block ได้ร้อยละ 41.58 และเมื่อใช้ร่วมกับแบบจำลอง VGG10 ใช้พารามิเตอร์เพียง 1,886,708 พารามิเตอร์ สามารถลดพารามิเตอร์ได้ร้อยละ 87.20 จาก VGG16 และร้อยละ 91.38 จาก InceptionV3 นอกจากนี้ได้ทำการเปรียบเทียบค่าความถูกต้องในการจำแนกโรคของใบพืชกับแบบจำลองโครงข่ายประสาทเทียมแบบเดิม เช่น VGG16, InceptionV3, DenseNet121, MobileNetV2 และ ResNet50

## 1.2 สมมติฐานการวิจัย

1.2.1 การจำแนกใบพืชโดยใช้การกำหนดจุดเด่นบนขอบใบด้วย Scale-Invariant Feature Transform (SIFT) สามารถเพิ่มค่าความถูกต้องในการจำแนกใบพืชได้หรือไม่

1.2.2 การปรับปรุงสถาปัตยกรรมของ VGG16 ร่วมกับ InceptionV3 with Parameter Reduction Layer สามารถเพิ่มค่าความถูกต้องในการจำแนกโรคของใบพืชได้หรือไม่

1.2.3 การปรับปรุงสถาปัตยกรรมของ VGG16 ร่วมกับ InceptionV3 with Parameter Reduction Layer สามารถลดจำนวนพารามิเตอร์ของสถาปัตยกรรมและเวลาในการประมวลผลได้หรือไม่

## 1.3 วัตถุประสงค์ของงานวิจัย

1.3.1 เพื่อเปรียบเทียบประสิทธิภาพการจำแนกใบพืชโดยใช้การกำหนดจุดเด่นบนขอบใบด้วย Scale-Invariant Feature Transform (SIFT)

1.3.2 เพื่อนำเสนอแบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer สำหรับจำแนกโรคของใบพืช

1.3.3 เพื่อเปรียบเทียบประสิทธิภาพ จำนวนพารามิเตอร์และเวลาในการประมวลผลของสถาปัตยกรรม VGG10 – InceptionV3 with Parameter Reduction Layer ที่นำเสนอกับสถาปัตยกรรมอื่น ๆ ของ CNN เช่น VGG16, InceptionV3, ResNet50, DenseNet121 และ MobileNetV2

## 1.4 ขอบเขตของงานวิจัย

งานวิจัยนี้มุ่งเน้นศึกษาการเพิ่มประสิทธิภาพของการจำแนกโรคของใบพืช โดยแบ่งการทำงานออกเป็น 2 ขั้นตอน ดังนี้

1.4.1 การจำแนกใบพืชโดยการกำหนดจุดเด่นบนขอบใบพืชด้วย Scale-Invariant Feature Transform (SIFT) ขั้นตอนนี้ประกอบด้วยขั้นตอนการประมวลผลภาพ เช่น การแปลงค่าสีของภาพ การเปลี่ยนแปลงลักษณะทางสัณฐานวิทยา การหาขอบภาพ เพื่อช่วยให้สามารถกำหนดจุดเด่นบนขอบภาพได้แม่นยำมากขึ้นก่อนการจำแนกใบพืช ในขั้นตอนการทดลองใช้รูปภาพใบพืชจากชุดข้อมูล PlantVillage จำนวน 10 คลาส แบ่งเป็นใบพืชที่เป็นโรค 5 คลาส และไม่เป็นโรค 5 คลาส สกัดคุณสมบัติของใบพืชด้วยโครงข่ายประสาทเทียมแบบคอนโวลูชันและจำแนกชนิดใบพืชด้วยแบบจำลองแบบป่าสุ่ม (Random Forest) จากนั้นเปรียบเทียบประสิทธิภาพการจำแนกใบพืชของวิธีการที่นำเสนอกับแบบจำลองอื่นของโครงข่ายประสาทเทียม

1.4.2 จากผลการเปรียบเทียบประสิทธิภาพการจำแนกใบพืชด้วยการกำหนดจุดเด่นบนขอบใบพืชในขั้นตอนแรก จึงเป็นแนวทางในการพัฒนาแบบจำลองโครงข่ายประสาทเทียมที่มีขนาดเล็ก โดยการปรับปรุงสถาปัตยกรรมของโครงข่ายประสาทเทียม VGG16 ด้วยการปรับลดจำนวนชั้นของโครงข่ายเพื่อให้แบบจำลองมีขนาดเล็กลง และมีการปรับปรุง InceptionV3 with Parameter Reduction Layer โดยการเพิ่มขนาดของแบบจำลองให้มีความกว้างมากขึ้น จากนั้นจึงประยุกต์ใช้งาน VGG16 ร่วมกับ InceptionV3 with Parameter Reduction Layer ที่ปรับปรุงแล้วเพื่อช่วยลดจำนวนพารามิเตอร์และเวลาในการประมวลผล ในขั้นตอนการทดลองประสิทธิภาพเบื้องต้นใช้รูปภาพใบมะเขือเทศจากชุดข้อมูล PlantVillage มีรูปภาพใบพืชจำนวน 18,159 รูปภาพ จากนั้นทดลองประสิทธิภาพของแบบจำลองด้วยชุดข้อมูลขนาดใหญ่ ที่มีรูปภาพจำนวน 54,304 รูปภาพ และมีข้อมูลรูปภาพใบพืชที่หลากหลาย โดยมีขั้นตอนการทำงานดังนี้

1.4.2.1 รูปภาพใบพืชที่ใช้สำหรับการทดลองประสิทธิภาพของแบบจำลองใช้รูปภาพใบพืชทั้งใบที่เป็นโรคและใบที่สุขภาพดี จำนวน 38 คลาส เป็นภาพสีอาร์จีบี ขนาด 224 x 224 พิกเซล

1.4.2.2 ปรับปรุง InceptionV3 with Parameter Reduction Layer โดยการเพิ่มชั้นคอนโวลูชันสำหรับรับข้อมูลนำเข้าให้มีความกว้างของโมเดลที่เพิ่มขึ้นเพื่อให้โมเดลสามารถทำงานแบบขนานกันได้รวดเร็วมากขึ้น และปรับขนาดของตัวกรอง (Filter Size) เพื่อลดจำนวนพารามิเตอร์ของแบบจำลอง

1.4.2.3 ประยุกต์ใช้ InceptionV3 with Parameter Reduction Layer ร่วมกับแบบจำลอง VGG โดยการเพิ่มระหว่างชั้นที่ 4 และ 5 ของ VGG16 จากนั้นทำการปรับลดจำนวนชั้นของโมเดล VGG16 ลงทีละชั้นจนได้ค่าความถูกต้องที่ดีที่สุดและใช้จำนวนพารามิเตอร์และเวลาการประมวลผลที่ลดลง

## 1.5 ประโยชน์ที่คาดว่าจะได้รับ

ได้แบบจำลองโครงข่ายประสาทเทียมชื่อ VGG10 – InceptionV3 with Parameter Reduction Layer ซึ่งเป็นโครงข่ายประสาทเทียมขนาดเล็กที่สามารถช่วยเพิ่มประสิทธิภาพการจำแนกโรคของใบพืชได้แม่นยำมากขึ้น และสามารถลดจำนวนพารามิเตอร์และต้นทุนการคำนวณของแบบจำลองได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

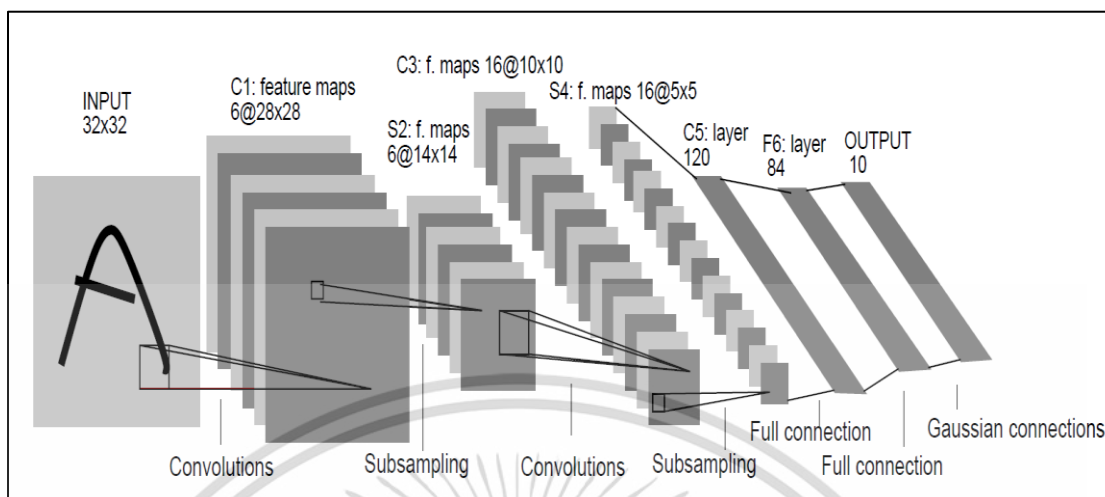
## บทที่ 2

# ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้นำเสนอเกี่ยวกับแนวคิดทฤษฎีและงานวิจัยที่เกี่ยวข้องกับการดำเนินงานวิจัย โดยมีวัตถุประสงค์เพื่อให้มีความรู้ ความเข้าใจเกี่ยวกับแนวคิดพื้นฐานของทฤษฎีต่างๆ ที่นำมาใช้สำหรับการวิจัย โดยประกอบด้วยทฤษฎีเกี่ยวกับโมเดลของโครงข่ายประสาทเทียม การสกัดคุณสมบัติของรูปภาพ การประมวลผลภาพ การจำแนกรูปภาพ และงานวิจัยที่เกี่ยวข้อง โดยมีรายละเอียดต่างๆ ดังนี้

### 2.1 โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks: CNN)

โครงข่ายประสาทเทียมแบบคอนโวลูชันเป็นโครงข่ายประสาทเทียมชนิดหนึ่งที่แตกต่างกันจากโครงข่ายประสาทเทียม (Artificial Neural Network: ANN) แบบเดิม โดย Keiron O'Shea และคณะ [16] กล่าวว่าโครงข่ายประสาทเทียมแบบคอนโวลูชันพัฒนาขึ้นเพื่อแก้ปัญหาข้อจำกัดของโครงข่ายประสาทเทียมแบบเดิมที่มีความซับซ้อนและข้อจำกัดในการคำนวณเกี่ยวกับรูปภาพ ดังนั้นโครงข่ายประสาทเทียมแบบคอนโวลูชันจึงพัฒนาให้สามารถดำเนินเกี่ยวกับรูปภาพได้เสมือนกับสมองของมนุษย์ โดยการจำลองการมองเห็นของโครงข่ายให้สามารถแยกคุณสมบัติของรูปภาพ (Feature Extraction) ออกมาเป็นส่วนๆ เช่น เส้นขอบ สี ขนาดของรูปภาพ มุมของภาพ เป็นต้น โดยการกำหนดขนาดตัวกรอง (Filter Size) หรือเคอร์เนล (Kernel Size) เพื่อดึงคุณสมบัติของรูปภาพออกมา และมีการกำหนดค่าการเลื่อน (Stride) ของตัวกรองและการขยายขอบภาพ (Padding) ซึ่งการกำหนดค่าเหล่านี้จะทำให้ได้แผนที่คุณสมบัติ (Feature Map) ของภาพ จากนั้นจึงนำแต่ละส่วนมารวมกันจนได้เป็นผลลัพธ์ของรูปภาพ โดยดำเนินการผ่านชั้นต่างๆ ของโครงข่าย ได้แก่ ชั้นนำเข้าข้อมูล (Input Layer) ชั้นซ่อน (Hidden Layer) และชั้นผลลัพธ์ (Output Layer) และนอกจากนี้โครงข่ายประสาทเทียมแบบคอนโวลูชันยังประกอบด้วยชั้นรวมผลลัพธ์ (Pooling Layer) และชั้นการเชื่อมต่อสมบูรณ์ (Fully Connected Layer) สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบคอนโวลูชันแสดงดังรูปที่ 2.1

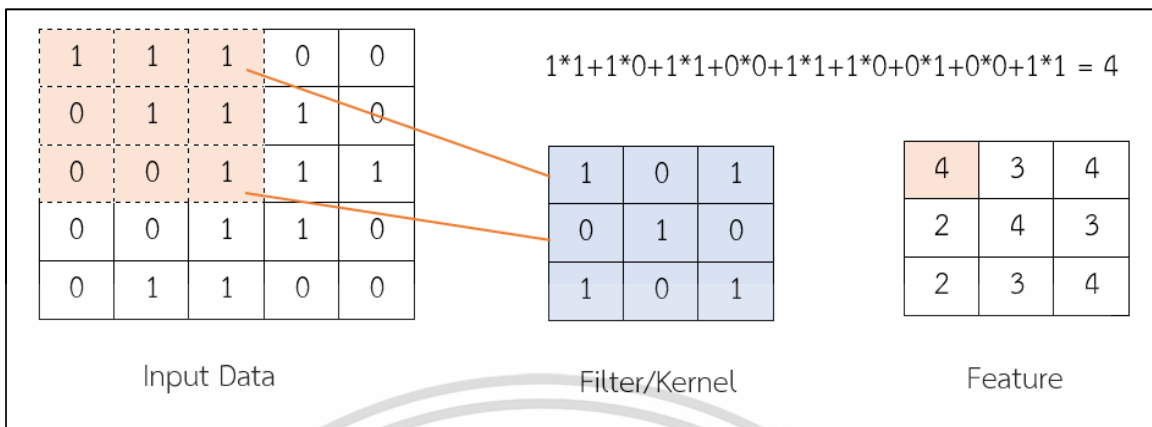


รูปที่ 2.1 สถาปัตยกรรมโครงข่ายประสาทเทียมแบบคอนโวลูชัน [16]

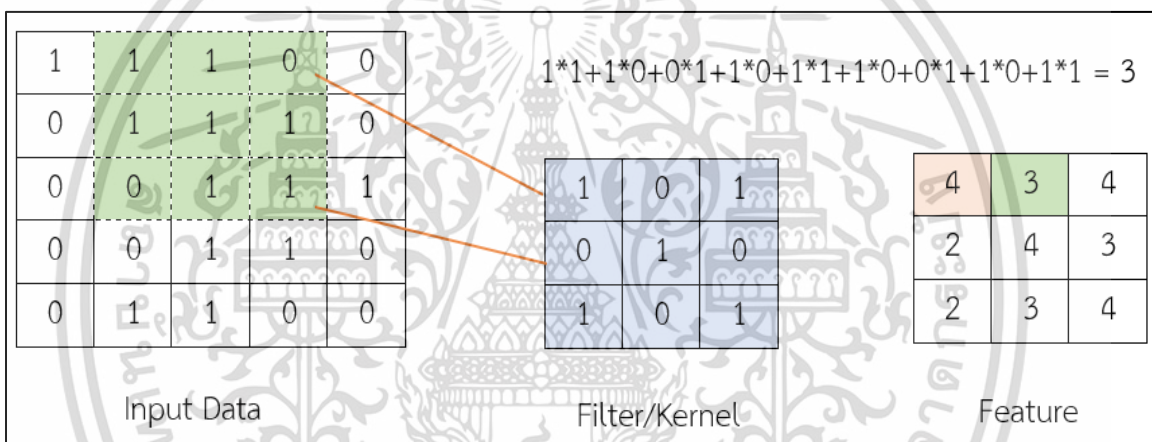
จากรูปที่ 2.1 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบคอนโวลูชันประกอบด้วยชั้นนำเข้าข้อมูล (Input Layer) ชั้นคอนโวลูชัน (Convolutional Layer) ชั้นรวมผลลัพธ์ (Pooling Layer) ชั้นการเชื่อมต่อสมบูรณ์ (Fully Connected Layer) และชั้นผลลัพธ์ (Output Layer) แต่ละชั้นมีรายละเอียดดังนี้

**ชั้นนำเข้าข้อมูล (Input Layer)** เป็นชั้นสำหรับการรับข้อมูลรูปภาพเข้าสู่การทำงานของโครงข่าย ซึ่งจะต้องระบุขนาดของรูปภาพเป็นพิกเซล (Pixel) ที่มีขนาดความกว้าง  $\times$  ความสูง  $\times$  ขนาดของช่องสี เช่น  $224 \times 224 \times 3$  กรณีเช่นนี้คือรับเข้าข้อมูลรูปภาพขนาดความกว้าง 224 พิกเซล ความสูง 224 พิกเซล และรูปภาพเป็นภาพสีอาร์จีบี (RGB) เนื่องจากภาพที่รับเข้าเป็นภาพสีอาร์จีบีจึงทำให้ขนาดของช่องสีมีค่าเท่ากับ 3 และ  $224 \times 224 \times 1$  กรณีภาพที่รับเข้าเป็นภาพระดับสีเทาจะทำให้ขนาดของช่องสีเท่ากับ 1

**ชั้นคอนโวลูชัน (Convolutional Layer)** เป็นชั้นสำหรับการสกัดคุณสมบัติ (Feature Extraction) ของรูปภาพ Abdelmalek และคณะ [1] กล่าวว่า ชั้นคอนโวลูชันเป็นชั้นที่สำคัญของสถาปัตยกรรมโครงข่ายแบบคอนโวลูชัน เป็นชั้นสำหรับเชื่อมต่อนำเข้าเพื่อสร้างแผนที่คุณสมบัติ (Feature Map) ซึ่งจำนวนของคุณสมบัติที่ได้จะขึ้นอยู่กับข้อกำหนดค่าของขนาดตัวกรองหรือเคอร์เนลและค่าการเลื่อนของตัวกรอง โดยการนำเมทริกซ์ของรูปภาพคูณแบบดอทเมทริกซ์กับค่าตัวกรองหรือเคอร์เนลตามจำนวนที่ต้องการ โดยการนำขนาดของตัวกรองวางทาบลงบนเมทริกซ์ของรูปภาพจากด้านซ้ายและทำการเลื่อนไปทางขวา การเลื่อนของตัวกรองขึ้นอยู่กับค่าการเลื่อน (Stride) ดังรูปที่ 2.2 และ 2.3

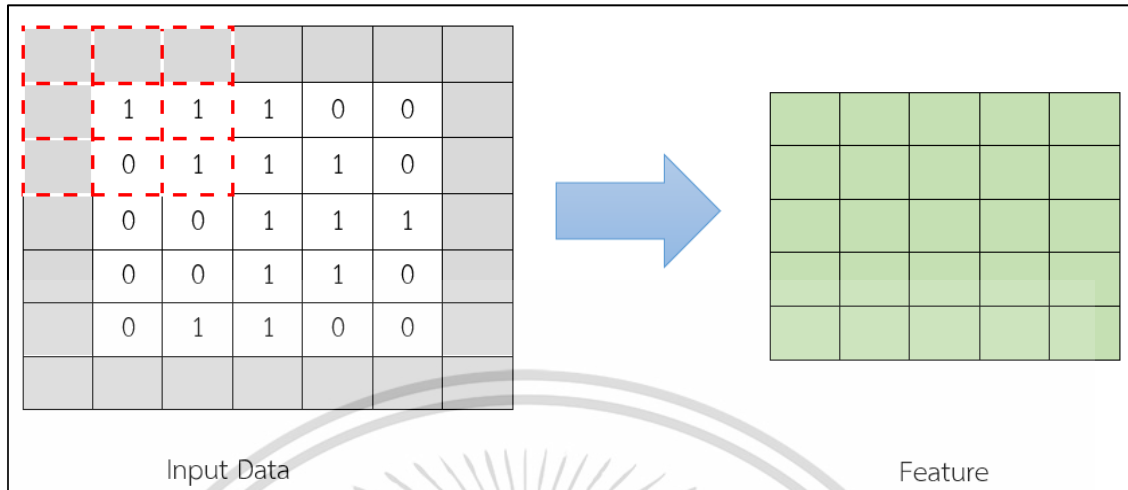


รูปที่ 2.2 การทำ Convolutional Feature



รูปที่ 2.3 การเลื่อนของตัวกรองที่มีค่าการเลื่อนเท่ากับ 1

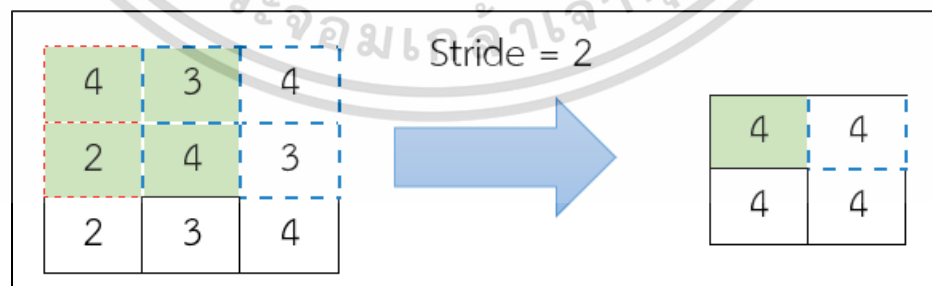
จากรูปที่ 2.2 เป็นการสกัดคุณสมบัติของรูปภาพนำเข้าโดยกำหนดให้ตัวกรองมีขนาด 3 x 3 และเมื่อกำหนดค่าการเลื่อนของตัวกรองให้เลื่อนจากทางซ้ายไปขวามีค่าเท่ากับ 1 ตามรูปที่ 2.3 จะทำให้ได้คุณสมบัติของภาพออกมาจำนวน 9 คุณสมบัติ ซึ่งมีขนาดเล็กกว่าข้อมูลรูปภาพนำเข้า โดยพิกเซลที่เป็นส่วนของขอบภาพจะไม่ถูกนำมาคำนวณเนื่องจากมีขนาดเล็กกว่าขนาดของตัวกรองจึงส่งผลให้ไม่ได้รับคุณสมบัติของภาพในส่วนที่เป็นขอบภาพ ดังนั้นจึงสามารถแก้ปัญหานี้ได้ด้วยการกำหนดการขยายภาพ (Padding) ดังรูปที่ 2.4 โดยการขยายขอบของภาพออกด้านข้างขนาดเท่ากันทุกด้าน ซึ่งจะทำให้ตัวกรองสามารถตรวจจับคุณสมบัติของภาพที่อยู่บริเวณขอบภาพได้ถูกต้องมากขึ้น เพราะรูปภาพบางภาพคุณสมบัติที่ขอบภาพอาจเป็นตัวช่วยในการตัดสินใจเกี่ยวกับรูปภาพได้มากขึ้น



รูปที่ 2.4 การขยายขอบของภาพ (Padding)

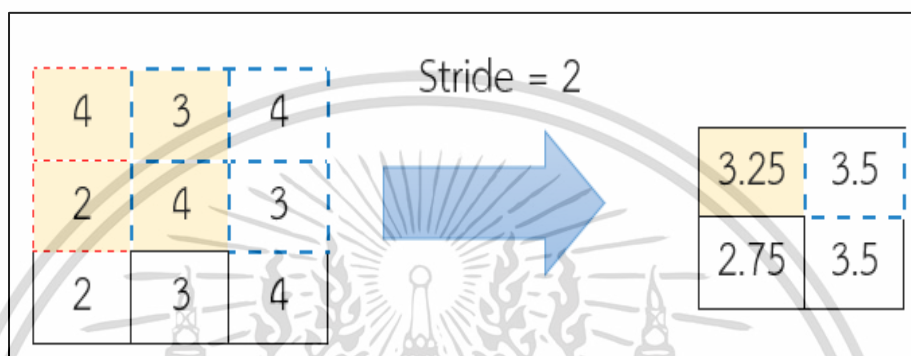
จากรูปที่ 2.4 รูปทางด้านซ้ายมือในส่วนที่เป็นสีเทาคือส่วนของขอบภาพที่ทำการขยายขอบภาพเพิ่มเข้าไป โดยค่าที่เพิ่มเติมเข้าไปอาจจะเป็นค่า 0 หรือค่าใดๆ เพื่อให้ตัวกรองสามารถวางทับลงบนภาพนำเข้าได้ครบทุกพิกเซล ซึ่งจะทำได้จำนวนของคุณสมบัติที่มีขนาดเท่ากับรูปภาพนำเข้า

ชั้นรวมผลลัพธ์ (Pooling Layer) ผลลัพธ์ของแผนที่คุณสมบัติ (Feature Map) ที่ได้รับจากชั้นคอนโวลูชันจะเป็นจำนวนของคุณสมบัติทั้งหมดที่ตัวกรองสามารถสกัดออกมาได้จากรูปภาพนำเข้า ซึ่งเมื่อจะนำไปใช้งานต่อในชั้นต่อไป อาจส่งผลให้มีจำนวนคุณสมบัติที่มากเกินไปและอาจจะไม่ใช่คุณสมบัติที่เด่นของรูปภาพนั้นๆ ดังนั้นชั้นรวมผลลัพธ์จึงเป็นชั้นที่นำผลลัพธ์แผนที่คุณสมบัติที่ได้จากชั้นคอนโวลูชันมาทำการคัดเลือกเอาเฉพาะคุณสมบัติเด่นของรูปภาพ ซึ่งสามารถทำได้ 2 วิธีคือ การรวมผลลัพธ์โดยเลือกค่าที่มากที่สุด (Max Pooling) และการรวมผลลัพธ์โดยใช้ค่าเฉลี่ย (Average Pooling) แสดงรายละเอียดดังรูปที่ 2.5 และ 2.6



รูปที่ 2.5 การรวมผลลัพธ์โดยใช้ค่ามากที่สุด (Max Pooling)

จากรูปที่ 2.5 แสดงการรวมผลลัพธ์โดยเลือกใช้ค่ามากที่สุดและกำหนดให้มีค่าการเลื่อนเท่ากับ 2 ซึ่งวิธีนี้จะพิจารณาเลือกค่าที่มากที่สุดในขณะที่ตัวกรองกำลังทาบบนแผนผังคุณสมบัติ ดังนั้นจากคุณสมบัติของภาพนำเข้าจำนวน 9 คุณสมบัติ เมื่อผ่านชั้นรวมผลลัพธ์โดยใช้ค่ามากที่สุดจะได้คุณสมบัติจำนวน 4 คุณสมบัติ



รูปที่ 2.6 การรวมผลลัพธ์โดยใช้ค่าเฉลี่ย (Average Pooling)

จากรูปที่ 2.6 แสดงการรวมผลลัพธ์โดยเลือกใช้ค่าเฉลี่ยและกำหนดให้มีค่าการเลื่อนเท่ากับ 2 ซึ่งวิธีนี้จะพิจารณาโดยนำค่าของคุณสมบัติในขณะที่ตัวกรองกำลังทาบบนแผนที่คุณสมบัติมารวมกันและหาค่าเฉลี่ย ดังนั้นจากคุณสมบัติของภาพนำเข้าจำนวน 9 คุณสมบัติ เมื่อผ่านชั้นรวมผลลัพธ์โดยใช้ค่ามากที่สุดจะได้คุณสมบัติจำนวน 4 คุณสมบัติเท่ากับวิธีการรวมผลลัพธ์โดยเลือกค่ามากที่สุด

**ชั้นการเชื่อมต่อสมบูรณ์ (Fully Connected Layer)** จากการทำงานของชั้นคอนโวลูชันและชั้นรวมผลลัพธ์จะทำให้ได้แผนที่คุณสมบัติขนาด 2 มิติ ในขณะที่ชั้นผลลัพธ์ที่เป็นชั้นสำหรับการจำแนกข้อมูลรูปภาพจำเป็นต้องใช้แผนที่คุณสมบัติขนาด 1 มิติเท่านั้น ดังนั้นในชั้นการเชื่อมต่อสมบูรณ์จึงเป็นชั้นของการรวมเอาผลลัพธ์จากชั้นคอนโวลูชันและชั้นรวมผลลัพธ์ทั้งหมดที่อยู่ก่อนหน้าเข้าด้วยกัน จากนั้นทำการแปลงมิติของแผนที่คุณสมบัติจาก 2 มิติ เป็น 1 มิติ เพื่อเชื่อมต่อไปยังชั้นผลลัพธ์ต่อไป จำนวนชั้นการเชื่อมต่อสมบูรณ์อาจจะใช้ 2 หรือ 3 ชั้นก็ได้ขึ้นอยู่กับการออกแบบโครงข่ายของผู้ใช้งาน

**ชั้นผลลัพธ์ (Output Layer)** หรือเรียกว่าชั้นซอฟต์แม็กซ์ (Softmax) เป็นชั้นที่อยู่สุดท้ายของโครงข่ายประสาทเทียมแบบคอนโวลูชัน มีการใช้งานฟังก์ชันการเปิดใช้งาน (Activation Functions) สำหรับการประมวลผลผลลัพธ์และตัดสินใจว่าแผนที่คุณสมบัติที่ได้รับมาจากชั้นก่อนหน้านั้นมีความน่าจะเป็นของรูปภาพใด โดยปกติในชั้นผลลัพธ์นี้จะมีฟังก์ชันการเปิดใช้งานที่หลากหลาย เช่น ฟังก์ชัน Rectified Linear Unit (ReLU) ฟังก์ชัน Tanh ฟังก์ชัน Sigmoid และฟังก์ชันซอฟต์แม็กซ์ ส่วนการเลือกใช้งานฟังก์ชันใดนั้นขึ้นอยู่กับลักษณะของงาน เช่น หากเป็นการจำแนกข้อมูลรูปภาพแบบไบนารี

คลาส (Binary Class) ก็สามารเลือกใช้ฟังก์ชัน Sigmoid ได้ และหากเป็นการจำแนกข้อมูลรูปภาพแบบหลายคลาส (Multiple Classes) ก็เลือกใช้เป็นฟังก์ชันซอฟต์แม็กซ์ เป็นต้น

## 2.2 VGGNet

VGGNet พัฒนาโดย Karen Simonyan [17] จาก Visual Geometry Group, Department of Engineering Science, University of Oxford โดยชนะการแข่งขัน ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) 2014 ผู้พัฒนามีแนวคิดในการปรับปรุงโครงข่ายประสาทเทียมแบบคอนโวลูชันโดยการปรับปรุงให้โครงข่ายมีความลึกมากขึ้นด้วยการเพิ่มชั้นคอนโวลูชันให้มากขึ้นเป็น 16 ชั้น และ 19 ชั้น และปรับขนาดของตัวกรองให้เล็กลงเพียง 3x3 และใช้ตัวกรองขนาดเดียวกันทั้งโครงข่าย นอกจากนี้ยังมีการเพิ่มชั้นการรวมผลลัพธ์จำนวน 5 ชั้น และชั้นการเชื่อมต่อสมบูรณ์จำนวน 3 ชั้น แสดงรายละเอียดโครงข่ายดังรูปที่ 2.7

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 LRN	conv3-64 <b>conv3-64</b>	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 <b>conv3-128</b>	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 <b>conv1-256</b>	conv3-256 conv3-256 <b>conv3-256</b>	conv3-256 conv3-256 <b>conv3-256</b>
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 <b>conv1-512</b>	conv3-512 conv3-512 <b>conv3-512</b>	conv3-512 conv3-512 conv3-512 <b>conv3-512</b>
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 <b>conv1-512</b>	conv3-512 conv3-512 <b>conv3-512</b>	conv3-512 conv3-512 conv3-512 <b>conv3-512</b>
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

รูปที่ 2.7 สถาปัตยกรรมโครงข่ายประสาทเทียมแบบ VGGNet [17]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.7 แสดงรายละเอียดของโครงข่ายประสาทเทียมแบบ VGGNet ที่ปรับปรุงโดยการเพิ่มชั้นคอนโวลูชันที่มีจำนวนที่ต่างกันไปตั้งแต่ 11 ชั้น จนถึง 19 ชั้น โครงข่ายทั้งหมดที่ปรับปรุงกำหนดให้รับค่ารูปภาพนำเข้ามาขนาด  $224 \times 224$  พิกเซลเท่ากันทุกเครือข่าย ตามด้วยชั้นคอนโวลูชันจำนวนชั้นแตกต่างกันตามความลึกของโครงข่าย ใช้ขนาดตัวกรอง  $3 \times 3$  ตลอดทั้งโครงข่ายและมีการใช้ชั้นรวมผลลัพธ์แบบเลือกค่าผลรวมมากที่สุดจำนวน 5 ชั้น จากนั้นตามด้วยชั้นการเชื่อมต่อสมบูรณ์จำนวน 3 ชั้น โดยในชั้นที่ 1 และ 2 กำหนดให้มีค่าเท่ากับ 4096 คลาส และชั้นที่ 3 กำหนดให้มีค่าเท่ากับ 1000 คลาส และสุดท้ายเป็นชั้นผลลัพธ์มีการใช้ฟังก์ชันการเปิดใช้งานคือฟังก์ชันซอฟต์แวร์แม็ก ซึ่งโครงข่ายที่ปรับปรุงแต่ละแบบจะมีจำนวนพารามิเตอร์ที่ต่างกัน ดังรูปที่ 2.8

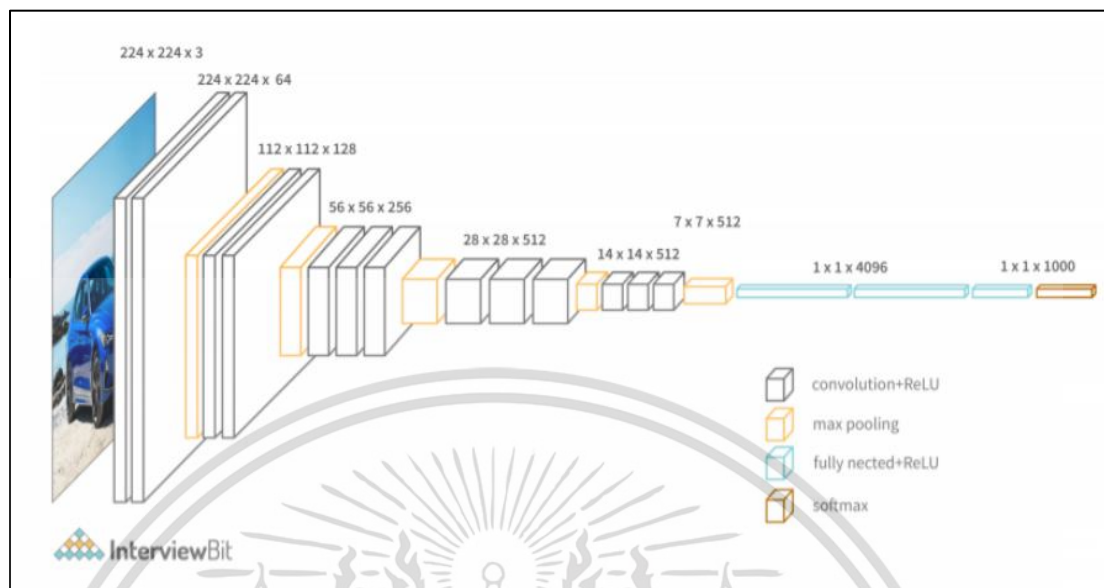
Network	A,A-LRN	B	C	D	E
Number of parameters	133	133	134	138	144

รูปที่ 2.8 จำนวนพารามิเตอร์ของโครงข่ายประสาทแบบ VGGNet (หน่วยเป็นล้านพารามิเตอร์) [17]

จากรูปที่ 2.8 แสดงให้เห็นว่าโครงข่ายประสาทเทียมแบบ VGGNet แต่ละแบบที่มีจำนวนชั้นที่ต่างกันก็จะมีจำนวนพารามิเตอร์ที่ต่างกันตามไปด้วย โครงข่ายที่มีจำนวนพารามิเตอร์มากที่สุดคือ VGGNet ที่มีจำนวน 19 ชั้น มีพารามิเตอร์ทั้งหมด 144 ล้านพารามิเตอร์ รองลงมาคือ VGGNet ที่มีจำนวน 16 ชั้น มีพารามิเตอร์ทั้งหมด 138 ล้านพารามิเตอร์ ซึ่งโครงข่ายทั้ง 2 แบบนี้คือโครงข่าย VGGNet ที่ปรับปรุงแล้วมีประสิทธิภาพดีที่สุดใน

#### VGG16

VGG16 เป็นโครงข่ายประสาทเทียมแบบ VGGNet ที่มีจำนวนทั้งหมด 16 ชั้น จึงมีชื่อว่า VGG16 ประกอบด้วยชั้นรับภาพนำเข้า ชั้นคอนโวลูชัน และชั้นผลลัพธ์ จำนวนทั้งหมด 16 ชั้น โดยชั้นรับภาพนำเข้าโครงข่ายจะรับเข้าข้อมูลรูปภาพขนาด  $224 \times 224$  พิกเซล ตามด้วยชั้นคอนโวลูชัน ประกอบด้วย Conv2D จำนวน 13 ชั้น โดยใช้จำนวนแชนแนล (Number of Channels) เท่ากับ 64 จำนวน 2 Conv2D จำนวนแชนแนลเท่ากับ 128 จำนวน 2 Conv2D จำนวนแชนแนลเท่ากับ 256 จำนวน 3 Conv2D และจำนวนแชนแนล เท่ากับ 512 จำนวน 6 Conv2D ใช้ขนาดตัวกรองเท่ากับ  $3 \times 3$  ตลอดทั้งโครงข่าย หลัง Conv2D ชั้นที่ 2 ชั้นที่ 4 ชั้นที่ 7 ชั้นที่ 10 และชั้นที่ 13 จะตามด้วยชั้นรวมผลลัพธ์แบบเลือกค่ามากที่สุด (Max Pooling) จำนวน 5 ชั้น ที่มีค่าของการเลื่อนเท่ากับ 2 ทั้งโครงข่าย และตามด้วยชั้นการเชื่อมต่อสมบูรณ์จำนวน 3 ชั้น โดย 2 ชั้นแรกใช้ 4096 คลาส และชั้นสุดท้ายใช้ 1000 คลาส ส่วนในชั้นผลลัพธ์ใช้ซอฟต์แวร์แม็กเป็นฟังก์ชันการเปิดใช้งาน สถาปัตยกรรมของ VGG16 แสดงดังรูปที่ 2.9



รูปที่ 2.9 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ VGG16

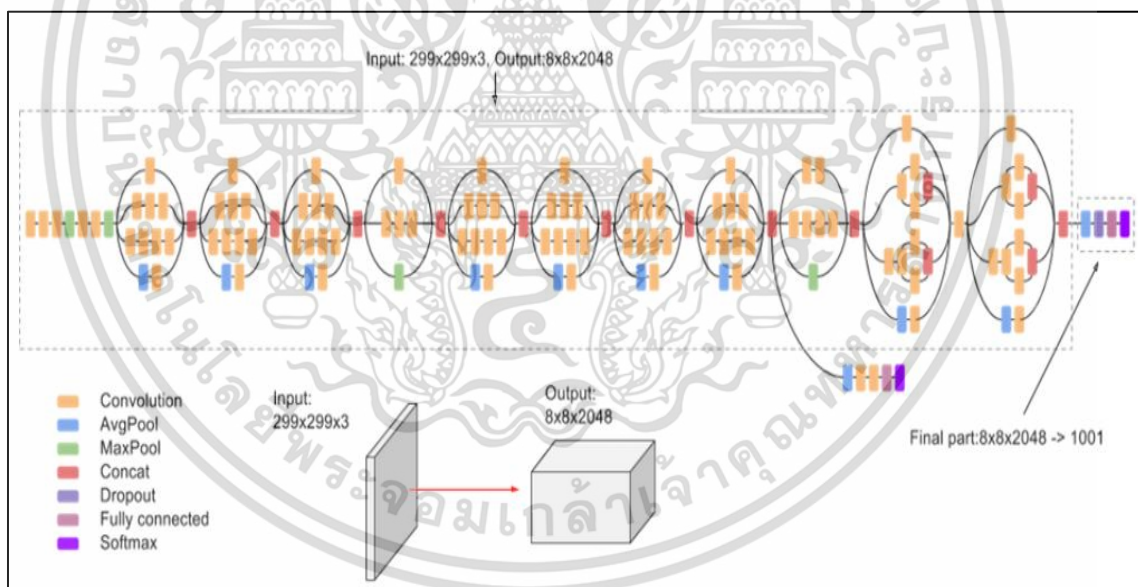
(ที่มา: <https://www.interviewbit.com/blog/cnn-architecture/>)

จากรูปที่ 2.9 แสดงสถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ VGG16 เป็นโครงข่ายที่มีความลึกจำนวน 16 ชั้น ซึ่งไม่นับรวมชั้นรวมผลลัพธ์แบบเลือกค่ามากที่สุดและชั้นผลลัพธ์ โครงข่าย VGG16 เป็นโครงข่ายที่มีการทำงานแบบลำดับขั้น กล่าวคือเมื่อรับรูปภาพนำเข้าสู่โครงข่ายและผ่านชั้นคอนโวลูชันชั้นที่ 1 แล้ว ผลลัพธ์ที่ได้จากชั้นที่ 1 ก็ส่งต่อไปให้ชั้นที่ 2 ผลลัพธ์ที่ได้จากชั้นที่ 2 ก็ส่งต่อไปให้ชั้นที่ 3 ถัดไปตามลำดับ ซึ่งข้อดีของโครงข่ายนี้คือง่ายต่อการออกแบบโครงข่าย เนื่องจากมีการทำงานแบบเป็นลำดับขั้น และมีการใช้ขนาดตัวกรองที่เท่ากันตลอดทั้งโครงข่าย แต่ก็ยังคงมีข้อจำกัดเนื่องจากโครงข่ายนี้พัฒนาขึ้นเพื่อเพิ่มความลึกของโครงข่าย ดังนั้นเมื่อโครงข่ายมีความลึกมากขึ้นก็ส่งผลให้โครงข่ายมีขนาดใหญ่ขึ้นและต้องใช้พารามิเตอร์จำนวนมากหลายล้านตัว รวมถึงเวลาที่ใช้ในการประมวลผลและต้นทุนการคำนวณต่างๆ ก็เพิ่มขึ้นด้วยเช่นกัน

VGG19 มีสถาปัตยกรรมโครงข่ายเหมือนกับสถาปัตยกรรมของโครงข่ายแบบ VGG16 แต่จะมีจำนวนชั้นคอนโวลูชันเพิ่มขึ้นจาก 13 ชั้นเป็น 16 ชั้น (แสดงรายละเอียดตามรูปที่ 2.7 โครงข่ายแบบ E คือสถาปัตยกรรมของโครงข่าย VGG19) มีการปรับจำนวนชั้นคอนโวลูชันในชั้นที่ใช้จำนวนแกนแนลเท่ากับ 256 เพิ่มอีกหนึ่งชั้น และชั้นคอนโวลูชันในชั้นที่ใช้จำนวนแกนแนลเท่ากับ 512 เพิ่มขึ้นอีก 2 ชั้น ดังนั้นเมื่อเพิ่มจำนวนชั้นของโครงข่ายมากขึ้นย่อมส่งผลให้โครงข่ายมีความลึกมากขึ้นและต้องใช้พารามิเตอร์ทั้งหมด 144 ล้านพารามิเตอร์ ซึ่งถือได้ว่าเป็นโครงข่ายประสาทเทียมขนาดใหญ่โครงข่ายหนึ่ง

## 2.3 InceptionV3

InceptionV3 พัฒนาโดย Szegedy, C. และคณะ [18] เป็นนักวิจัยของกูเกิ้ล (Google) ในปี 2015 ได้รับรางวัลรองชนะเลิศอันดับหนึ่งจากการแข่งขัน ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) 2015 โดยมีแนวคิดในการปรับสถาปัตยกรรมของโครงข่ายประสาทเทียมเชิงลึกให้โครงข่ายมีความกว้างมากขึ้น กล่าวคือนักวิจัยส่วนใหญ่มุ่งเน้นการพัฒนาโครงข่ายประสาทเทียมที่เน้นให้โครงข่ายมีความลึกมากขึ้น แต่ผู้พัฒนาโครงข่ายนี้เน้นพัฒนาให้โครงข่ายมีความกว้างมากขึ้น โดยการเพิ่มชั้นคอนโวลูชันในระดับเดียวกันเพื่อเป็นการเพิ่มชั้นให้โครงข่ายสามารถสกัดคุณสมบัติของรูปภาพนำเข้าแบบขนานให้ชั้นคอนโวลูชันในระดับเดียวกันสามารถทำงานขนานไปพร้อมๆ กันได้ จากนั้นจึงนำเอาผลลัพธ์ของชั้นคอนโวลูชันแต่ละชั้นมารวมกันด้วยชั้นการต่อข้อมูล (Concatenation Layer) InceptionV3 เป็นรุ่นที่ 3 ของ Inception โดยมีวัตถุประสงค์เพื่อเพิ่มประสิทธิภาพของการจำแนกข้อมูลรูปภาพแต่ต้องการลดจำนวนต้นทุนการคำนวณ พารามิเตอร์และเวลาในการทำงาน สถาปัตยกรรมรุ่นแรกของ Inception ดังรูปที่ 2.10



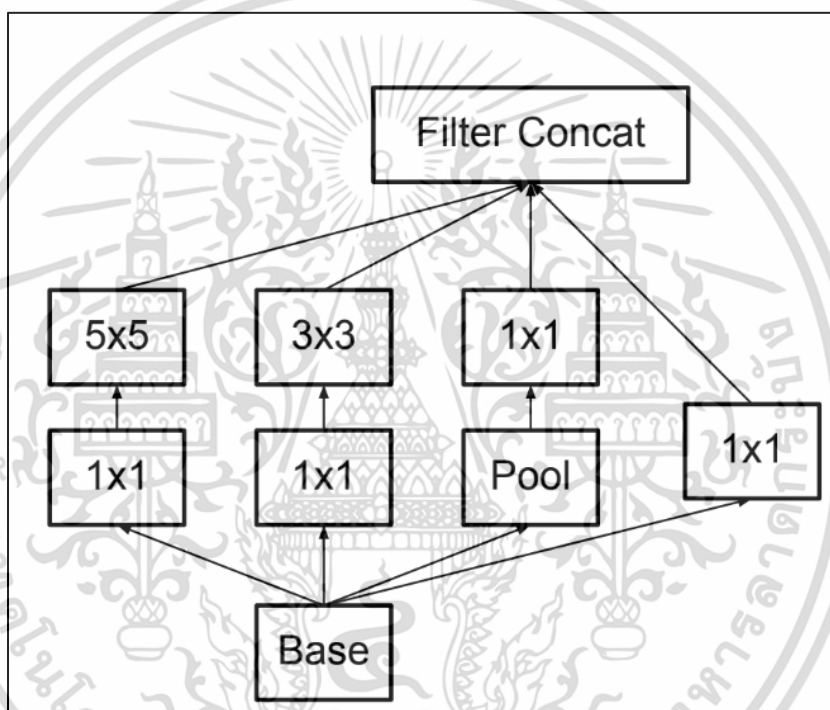
รูปที่ 2.10 สถาปัตยกรรมพื้นฐานของโครงข่ายแบบ Inception

(ที่มา: <https://cloud.google.com/tpu/docs/inception-v3-advanced>)

จากรูปที่ 2.10 แสดงสถาปัตยกรรมของโครงข่ายแบบ Inception ที่ประกอบด้วยชั้นนำเข้ารูปภาพ ชั้นคอนโวลูชัน ชั้นการรวมผลลัพธ์ ชั้นการเชื่อมต่อข้อมูล (Concatenation Layer) ชั้นลบข้อมูลออก (Dropout Layer) ชั้นการเชื่อมต่อสมบูรณ์ และชั้นผลลัพธ์ โดยในโครงข่ายจะแบ่งชั้นคอนโวลูชันออกเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

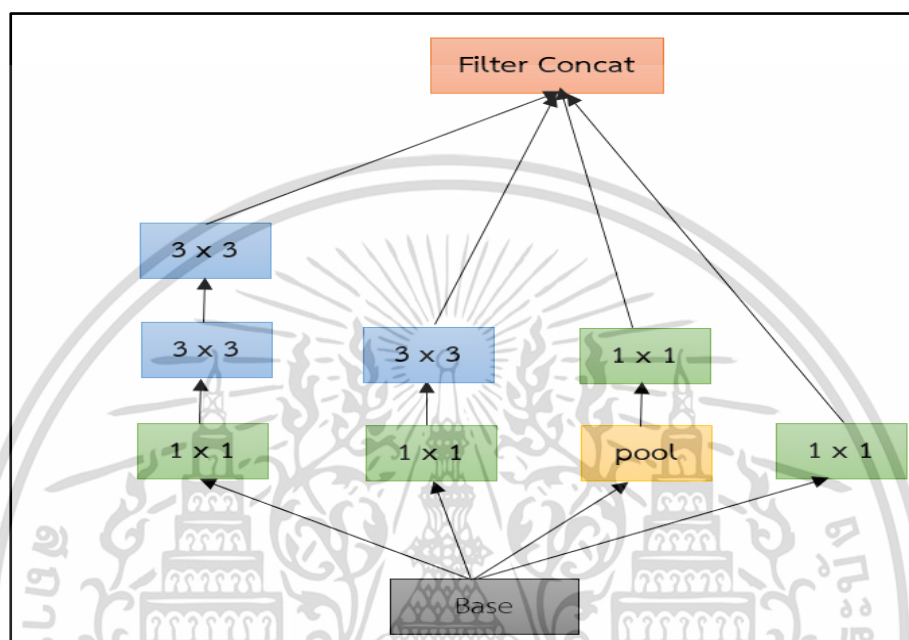
บล็อก เรียกว่า Inception Block หรือ Inception Modul ในแต่ละบล็อกหรือโมดูลจะประกอบด้วยชั้นคอนโวลูชันจำนวน 3 ชั้น และชั้นการรวมผลลัพธ์ด้วยค่ามากที่สุดจำนวน 1 ชั้น โดยใช้หลักการสกัดคุณลักษณะของรูปภาพแบบขนาน มีการใช้ตัวกรองที่มีขนาดแตกต่างกัน จึงทำให้ได้แผนที่คุณสมบัติที่มีขนาดแตกต่างกัน โดยสถาปัตยกรรมพื้นฐานของบล็อกประกอบด้วย ชั้นคอนโวลูชัน  $1 \times 1$  คอนโวลูชัน  $1 \times 1$  และตามด้วยชั้นคอนโวลูชัน  $5 \times 5$  ชั้นคอนโวลูชัน  $1 \times 1$  ตามด้วยชั้นคอนโวลูชัน  $3 \times 3$  และชั้นการรวมผลลัพธ์ด้วยค่าเฉลี่ยตามด้วยชั้นคอนโวลูชัน  $1 \times 1$  แสดงดังรูปที่ 2.11



รูปที่ 2.11 สถาปัตยกรรมของโมดูล Inception [18]

จากรูปที่ 2.11 เมื่อชั้นคอนโวลูชันแต่ละชั้นที่มีขนาดตัวกรองที่แตกต่างกันและทำงานแบบขนานไปพร้อมกันจึงทำให้ได้แผนที่คุณสมบัติจากชั้นคอนโวลูชันแต่ละชั้นที่แตกต่างกัน ดังนั้นขั้นตอนต่อไปคือการนำแผนที่คุณสมบัติจากแต่ละชั้นมาเชื่อมต่อกันด้วยชั้นการเชื่อมต่อข้อมูลเพื่อใช้สำหรับเป็นข้อมูลน้ำเข้าให้กับชั้นถัดไป ระหว่างแต่ละบล็อกก็เชื่อมต่อข้อมูลด้วยชั้นการเชื่อมต่อข้อมูล และในส่วนท้ายของโครงข่ายก่อนชั้นการเชื่อมต่อสมบูรณ์มีการลบข้อมูลคุณสมบัติบางตัวออกไปด้วยชั้นลบข้อมูลออก (Dropout Layer) ซึ่งจะช่วยให้ลดจำนวนของคุณสมบัติบางตัวที่อ่อนแอออกไป ข้อดีของบล็อก Inception คือทำให้ได้แผนที่คุณสมบัติของรูปภาพที่มากขึ้นและสามารถลดปัญหาคอขวดสำหรับการสกัดคุณสมบัติของรูปภาพได้ จึงทำให้สามารถลดเวลาการประมวล จำนวนพารามิเตอร์ของโครงข่าย และ

ต้นทุนการคำนวณลงได้ และต่อมาได้มีการพัฒนาบล็อกของ Inception รุ่นที่ 2 รุ่นที่ 3 และรุ่นที่ 7 ซึ่งในงานวิจัยนี้เลือกใช้สถาปัตยกรรมของบล็อก InceptionV3 จึงนำเสนอรายละเอียดเฉพาะสถาปัตยกรรมของบล็อก InceptionV3 เท่านั้น โดยสถาปัตยกรรมของบล็อก InceptionV3 แสดงดังรูปที่ 2.12



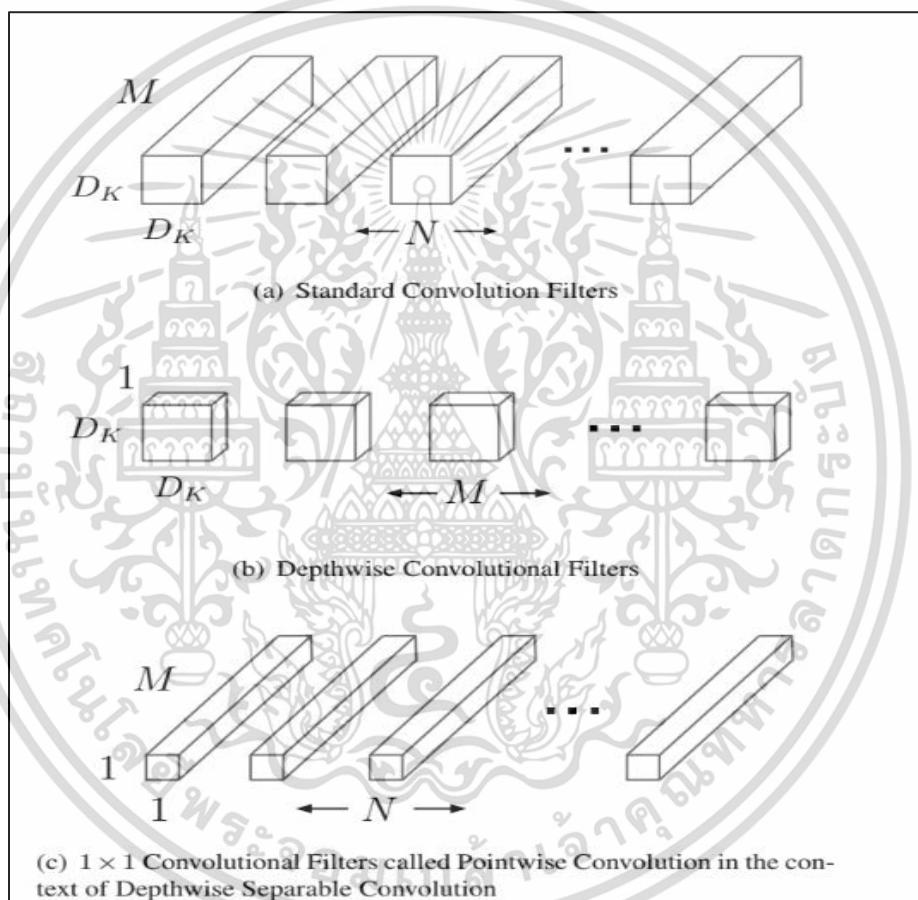
รูปที่ 2.12 สถาปัตยกรรมของโมดูล InceptionV3

จากรูปที่ 2.12 สถาปัตยกรรมของบล็อก InceptionV3 ได้มีการปรับสถาปัตยกรรมโดยการปรับขนาดของตัวกรองให้มีขนาดเล็กลงจากเดิมชั้นคอนโวลูชันขนาด  $5 \times 5$  ปรับเปลี่ยนเป็นชั้นคอนโวลูชันขนาด  $3 \times 3$  ซ้อนกันจำนวน 2 ชั้นแทน โดยสถาปัตยกรรมของ InceptionV3 ประกอบด้วยชั้นคอนโวลูชัน  $1 \times 1$  ชั้นคอนโวลูชัน  $1 \times 1$  ตามด้วยชั้นคอนโวลูชัน  $3 \times 3$  จำนวนสองชั้น ชั้นคอนโวลูชัน  $1 \times 1$  ตามด้วยชั้นคอนโวลูชัน  $3 \times 3$  และ ชั้นการรวมผลลัพธ์ตามด้วยชั้นคอนโวลูชัน  $1 \times 1$  ซึ่งการแทนที่ชั้นคอนโวลูชัน  $5 \times 5$  ด้วยชั้นคอนโวลูชัน  $3 \times 3$  จำนวนสองชั้นทำให้จำนวนของพารามิเตอร์ลดลงจาก  $5 \times 5 = 25$  เป็น  $(3 \times 3) + (3 \times 3) = 18$  พารามิเตอร์

## 2.4 MobileNet

MobileNet พัฒนาโดย Howard, G.A. และคณะ [19] ในปี 2017 โดยมีแนวคิดในการพัฒนาโครงข่ายประสาทเทียมแบบคอนโวลูชันที่มีน้ำหนักเบา (Lightweight Convolutional Neural Network) สำหรับการใช้งานคอมพิวเตอร์วิทัศน์ (Computer Vision) บนอุปกรณ์พกพาขนาดเล็กและอุปกรณ์แบบฝังตัว (Embedded Device) ที่มีทรัพยากรการคำนวณที่จำกัด เนื่องจากโครงข่ายประสาท

เทียบแบบคอนโวลูชันรุ่นก่อนๆ จะเน้นการเพิ่มประสิทธิภาพของโครงข่ายด้วยวิธีการเพิ่มความลึกและขนาดของโครงข่ายซึ่งทำให้ต้องใช้ทรัพยากรการคำนวณที่สูงขึ้นตามไปด้วย ผู้พัฒนาจึงนำเสนอแนวคิดการเพิ่มประสิทธิภาพการทำงานของโครงข่ายโดยทำให้โครงข่ายมีขนาดเล็กลง โดยนำเสนอการทำงานของชั้นคอนโวลูชันของโครงข่ายที่มีชื่อว่า Depthwise Separable Convolution ซึ่งประกอบด้วย 2 ชั้นย่อย คือชั้น Depthwise convolution ใช้สำหรับการแยกตัวประกอบของคุณสมบัติ และชั้น Pointwise convolution ใช้สำหรับการปรับมิติของแผนที่คุณสมบัติให้อยู่ในรูป 1 มิติ แสดงดังรูปที่ 2.13



รูปที่ 2.13 รูปแบบการทำงานของชั้นคอนโวลูชันแบบเดียวกับชั้นคอนโวลูชันของ MobileNet [19]

จากรูปที่ 2.13 (a) แสดงลักษณะตัวกรองของชั้นคอนโวลูชันแบบมาตรฐานประกอบด้วยจำนวนช่องสัญญาณของข้อมูลเข้า ( $M$ ) ความกว้างและความสูงของแผนผังคุณสมบัติของข้อมูลเข้า ( $D_K$ ) จำนวนช่องสัญญาณของข้อมูลออก ( $N$ ) และความกว้างและความสูงของแผนที่คุณสมบัติของข้อมูลออก ( $D_F$ ) คำนวณได้จาก

$$D_K \cdot D_K \cdot M \cdot N \cdot D_F \cdot D_F$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งในโครงข่ายของ MobileNet มีการปรับปรุงโดยใช้ชั้นคอนโวลูชัน 2.13 (b) แทนโดยใช้จำนวนช่องสัญญาณเพียงหนึ่งตัวสำหรับข้อมูลเข้าและข้อมูลออก (M) จากนั้นทำการสร้างผลลัพธ์ที่เป็นเชิงเส้นจากชั้น Depthwise convolution เพื่อสร้างตัวกรองให้กับชั้น Depthwise Separable Convolution นอกจากนี้ผู้พัฒนาได้มีการนำเสนอค่าพารามิเตอร์ 2 พารามิเตอร์ คือตัวคูณความกว้าง (Width Multiplier:  $\alpha$ ) และตัวคูณความละเอียดภาพ (Resolution Multiplier:  $\rho$ ) สำหรับทำให้ชั้นคอนโวลูชันของโครงข่ายมีความเบาบางและรักษาความละเอียดของภาพ คำนวณได้จาก

$$D_K \cdot D_K \cdot \alpha M \cdot \rho D_F \cdot \rho D_F + \alpha M \cdot \alpha N \cdot \rho D_F \cdot \rho D_F$$

โดยค่า  $\alpha \in (0,1)$  มีค่าเท่ากับ 1, 0.75, 0.50 และ 0.25 และ  $\rho$  มีค่าเท่ากับ 224, 192, 160 และ 128 สถาปัตยกรรมของ MobileNet แสดงดังรูปที่ 2.14

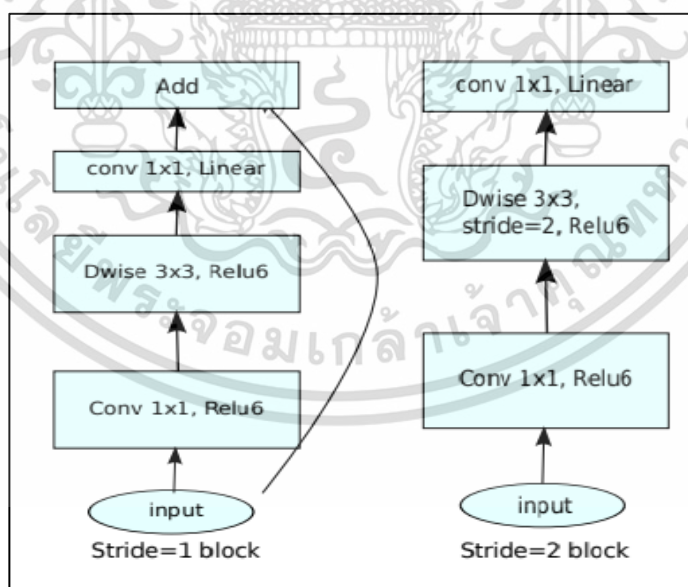
Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
5× Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool $7 \times 7$	$7 \times 7 \times 1024$
FC / s1	$1024 \times 1000$	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

รูปที่ 2.14 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ MobileNet [19]

จากรูปที่ 2.14 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ MobileNet ประกอบด้วยทั้งหมด 28 ชั้น ซึ่งไม่นับรวมชั้นการเชื่อมต่อสมบูรณ์กับชั้นผลลัพธ์ โดยในชั้น Depthwise Separable

convolutions จะใช้การทำให้เป็นมาตรฐานแบบแบตช์ (Batch Normalization) และฟังก์ชัน ReLU ตามหลังชั้น Depthwise convolution และชั้น Pointwise convolution และเมื่อทดลองกับชุดข้อมูล จาก ImageNet ปรากฏว่าโครงข่าย MobileNet ได้ค่าความถูกต้อง (Accuracy) Top-1 เท่ากับ 70.4% และค่าความถูกต้อง Top-5 เท่ากับ 89.5% และเมื่อเปรียบเทียบกับโครงข่ายประสาทเทียมแบบคอนโวลูชันแบบเดิม เช่น VGG16 และ GoogleNet ก็พบว่าโครงข่าย MobileNet มีค่าความถูกต้องน้อยกว่า VGG16 ประมาณ 0.9% แต่สามารถลดจำนวนพารามิเตอร์จาก 138 ล้านพารามิเตอร์เหลือเพียง 4.2 ล้านพารามิเตอร์ และมีค่าความถูกต้องมากกว่า GoogleNet ประมาณ 0.8% และลดจำนวนพารามิเตอร์จาก 6.8 ล้านพารามิเตอร์ เหลือเพียง 4.2 ล้านพารามิเตอร์

จากนั้นในปี 2019 Sandler, M. และคณะ [20] ได้พัฒนาโครงข่ายประสาทเทียมแบบ MobileNetV2 โดยมีแนวคิดในการพัฒนาแบบจำลองที่สามารถใช้งานแอปพลิเคชันบนอุปกรณ์พกพาและอุปกรณ์ฝังตัว ให้สามารถลดความจำเป็นในการใช้หน่วยความจำของอุปกรณ์ลงแต่ยังคงให้ค่าความแม่นยำที่สูง โดยการนำเสนอโมดูลใหม่เพื่อเป็นชั้นหนึ่งของโครงข่ายคือ Inverted Residual Structure ซึ่งจะทำหน้าที่ในการลดมิติของข้อมูลเป็นลักษณะคอขวด ก่อนที่จะมีการขยายมิติของข้อมูลอีกครั้งในชั้น Depthwise Convolution และจะทำการลดมิติของข้อมูลอีกครั้งเพื่อส่งไปยังชั้นถัดไป โดยโมดูลที่นำเสนอมี 2 แบบ แสดงดังรูปที่ 2.15



รูปที่ 2.15 สถาปัตยกรรมโมดูล Inverted Residual Structure ของโครงข่าย MobileNetV2 [20]

จากรูปที่ 2.15 สถาปัตยกรรมของโมดูล Inverted Residual Structure มี 2 แบบคือ แบบที่มีค่าการเลื่อน (Stride) เท่ากับ 1 และค่าการเลื่อนเท่ากับ 2 โดยทั้ง 2 แบบ ประกอบด้วยเริ่มต้นด้วยชั้นคอน

โวลูชัน ขนาด  $1 \times 1$  ที่ใช้ฟังก์ชัน ReLU6 จากนั้นตามด้วยชั้น Depthwise Convolution ขนาด  $3 \times 3$  ที่ใช้ฟังก์ชัน ReLU6 และตามด้วยชั้นคอนโวลูชัน ขนาด  $1 \times 1$  โดยในชั้นนี้จะทำการลบฟังก์ชัน ReLU6 ออกเพื่อให้ข้อมูลมีลักษณะเป็นเชิงเส้นและต้องการลดมิติของข้อมูลให้มีลักษณะเป็นคอขวดหรือที่เรียกว่า Bottleneck Residual Block เนื่องจากการลบข้อมูลที่ไม่เป็นเชิงเส้นในส่วนที่เป็นคอขวดออกจะสามารถเพิ่มค่าความแม่นยำในการจำแนกข้อมูลได้สูงขึ้น สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ MobileNetV2 แสดงดังรูปที่ 2.16

Input	Operator	$t$	$c$	$n$	$s$
$224^2 \times 3$	conv2d	-	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	conv2d 1x1	-	1280	1	1
$7^2 \times 1280$	avgpool 7x7	-	-	1	-
$1 \times 1 \times 1280$	conv2d 1x1	-	k	-	-

รูปที่ 2.16 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ MobileNetV2 [20]

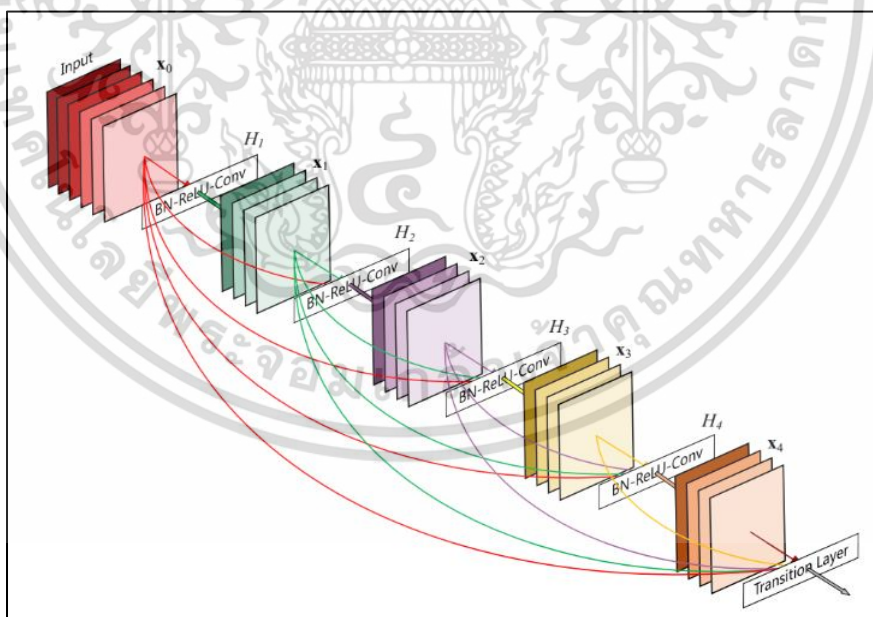
จากรูปที่ 2.16 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ MobileNetV2 ประกอบด้วยชั้นรับข้อมูลเข้า (ชั้น conv2d) และตามด้วยชั้น Bottleneck Residual Block จำนวน 7 ชั้น หลังจากนั้นตามด้วยชั้น conv2d ขนาด  $1 \times 1$  ชั้นรวมผลลัพธ์ด้วยค่ามากที่สุดขนาด  $7 \times 7$  และตามด้วยชั้น conv2d ขนาด  $1 \times 1$  อีกหนึ่งชั้น ซึ่งชั้นนี้จะทำหน้าที่ในการแปลงข้อมูลให้เป็นเชิงเส้นและบีบอัดข้อมูลให้อยู่ในลักษณะคอขวด โดย  $n$  คือจำนวนครั้งในการทำซ้ำของชั้นนั้นๆ  $c$  คือจำนวนช่องผลลัพธ์ของชั้นนั้นๆ ในระดับเดียวกัน และชั้นแรกๆในระดับเดียวกันจะมีค่าการเลื่อนเท่ากับ  $s$  ส่วนชั้นอื่นๆ มีค่าเท่ากับ 1 ใช้ขนาดของเคอร์เนลเท่ากับ  $3 \times 3$  ทั้งหมด และ  $t$  คือค่าปัจจัยของการขยายขนาดของข้อมูลเข้า นอกจากนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ยังได้มีการปรับปรุง SSDLite เพื่อใช้บนอุปกรณ์พกพาพร้อมกับโครงข่าย MobileNetV2 ซึ่งก็พบว่าโครงข่าย MobileNetV2 ที่ใช้ร่วมกับ SSDLite ที่ปรับปรุงแล้วสามารถลดจำนวนพารามิเตอร์และต้นทุนการคำนวณได้มากกว่าการใช้ SSD แบบเดิม

## 2.5 DenseNet121

โครงข่ายประสาทเทียมแบบ DenseNet121 ถูกนำเสนอในปี 2018 พัฒนาโดย Gao Huang [21] จากมหาวิทยาลัย Cornell University และคณะ โดยมีแนวคิดต่อยอดจากโครงข่ายประสาทเทียมแบบ ResNet ที่มีการเชื่อมต่อแบบทางลัดเพื่อข้ามไปยังชั้นถัดไป จากชั้นปัจจุบันเชื่อมต่อโดยตรงไปยังชั้นถัดไป ดังนั้นแนวคิดหลักของโครงข่ายประสาทเทียมแบบ DenseNet คือการเชื่อมต่อโดยตรงแบบไปข้างหน้า (Feed-Forward) ของชั้นปัจจุบันไปยังชั้นที่อยู่ในลำดับถัดไปทุกชั้น จึงทำให้แผนที่คุณสมบัติจากชั้นนำเข้าจึงถูกส่งไปยังทุกชั้นที่อยู่ในบล็อกเดียวกัน ซึ่งวิธีดังกล่าวสามารถช่วยลดปัญหาการไล่ระดับสีของรูปภาพเมื่อผ่านการทำงานด้วยโครงข่ายประสาทเทียมที่มีจำนวนหลายชั้นได้ และสามารถลดจำนวนพารามิเตอร์และจำนวนคุณสมบัติได้เนื่องจากวิธีนี้จะช่วยให้สามารถหลีกเลี่ยงการใช้คุณสมบัติซ้ำได้ การทำงานของชั้นนี้มีการรวมเข้าเป็นบล็อกที่เรียกว่า Dense Block แสดงโครงสร้างการทำงานภายในของ Dense Block ดังรูปที่ 2.17



รูปที่ 2.17 โครงสร้างการทำงานภายในของ Dense Block [21]

จากรูปที่ 2.17 แสดงโครงสร้างการทำงานภายในของ Dense Block โดยชั้นที่รับข้อมูลเข้าจะทำการส่งแผนที่คุณสมบัติแบบเชื่อมต่อโดยตรงไปยังชั้นที่อยู่ถัดไปทุกชั้นในบล็อกเดียวกัน จึงทำให้ทุกๆ ชั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้รับแผนที่คุณสมบัติของข้อมูลต้นฉบับที่เหมือนกันทุกชั้นทำให้การไหลของข้อมูลดีขึ้น โดยภายในบล็อกเดียวกันเมื่อได้ผลลัพธ์ของชั้นปัจจุบันก็จะทำการส่งผลลัพธ์ไปยังชั้นถัดไปที่อยู่ติดกันโดยผ่านการทำให้ข้อมูลเป็นมาตรฐานด้วยชั้น Batch Normalization และผ่านฟังก์ชัน Rectified Linear Unit (ReLU) ก่อนการทำคอนโวลูชันที่มีขนาด  $3 \times 3$  ซึ่งเรียกชั้นตอนนี้ว่า Composite function

หลังการทำงานของชั้นสุดท้ายภายในบล็อกเดียวกันโครงข่ายจะทำการลดมิติข้อมูลด้วยชั้นคอนโวลูชันขนาด  $1 \times 1$  เพื่อทำให้ขนาดข้อมูลเป็นคอคอดและเล็กลง จากนั้นจึงรวมข้อมูลจากทุกชั้นที่อยู่ในบล็อกเดียวกันด้วยการรวมข้อมูลด้วยค่าเฉลี่ย (Average Pooling) ขนาด  $2 \times 2$  และมีค่าการเลื่อนเท่ากับ 2 จากนั้นส่งผลลัพธ์ไปยังบล็อกถัดไปของโครงข่าย เรียกชั้นตอนนี้ว่า Transition Layer โดยจะใช้ชั้นนี้เชื่อมต่อระหว่างบล็อกของโครงข่ายประสาทเทียม DenseNet สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ DenseNet แสดงดังรูปที่ 2.18

Layers	Output Size	DenseNet-121	DenseNet-169	DenseNet-201	DenseNet-264
Convolution	$112 \times 112$	$7 \times 7$ conv, stride 2			
Pooling	$56 \times 56$	$3 \times 3$ max pool, stride 2			
Dense Block (1)	$56 \times 56$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$
Transition Layer (1)	$56 \times 56$ $28 \times 28$	$1 \times 1$ conv $2 \times 2$ average pool, stride 2			
Dense Block (2)	$28 \times 28$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$
Transition Layer (2)	$28 \times 28$ $14 \times 14$	$1 \times 1$ conv $2 \times 2$ average pool, stride 2			
Dense Block (3)	$14 \times 14$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 64$
Transition Layer (3)	$14 \times 14$ $7 \times 7$	$1 \times 1$ conv $2 \times 2$ average pool, stride 2			
Dense Block (4)	$7 \times 7$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$
Classification Layer	$1 \times 1$	$7 \times 7$ global average pool 1000D fully-connected, softmax			

รูปที่ 2.18 โครงข่ายประสาทเทียมแบบ DenseNet [21]

จากรูปที่ 2.18 แสดงสถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ DenseNet ขนาดต่างๆ โดยตัวเลขคือจำนวนชั้นของโครงข่าย เช่น DenseNet121 คือโครงข่าย DenseNet ที่มี 121 ชั้น ซึ่งโครงข่าย

แต่ละแบบจะมีจำนวนชั้นและจำนวนช่องสัญญาณของข้อมูลที่แตกต่างกัน งานวิจัยนี้เลือกใช้โครงข่ายแบบ DenseNet121 จึงขออธิบายรายละเอียดเฉพาะโครงข่ายแบบนี้

โครงข่ายประสาทเทียมแบบ DenseNet121 ประกอบด้วยชั้นนำเข้าไปรับข้อมูลรูปภาพขนาด  $224 \times 224$  พิกเซล จากนั้นผ่านชั้นคอนโวลูชันขนาด  $7 \times 7$  ที่มีค่าการเลื่อนเท่ากับ 2 และทำการรวมผลลัพธ์ด้วยค่ามากที่สุดขนาด  $3 \times 3$  มีค่าการเลื่อนเท่ากับ 2 จากนั้นจึงผ่าน Dense Block ที่ 1 ภายในบล็อกที่ 1 ประกอบด้วยชั้นคอนโวลูชันขนาด  $1 \times 1$  จำนวน 6 ชั้น ตามด้วยชั้นคอนโวลูชันขนาด  $3 \times 3$  จำนวน 6 ชั้น จากนั้นผ่านชั้น Transition Layer ชั้นที่ 1 ต่อด้วย Dense Block ที่ 2 ภายในบล็อกที่ 2 ประกอบด้วยชั้นคอนโวลูชันขนาด  $1 \times 1$  จำนวน 12 ชั้น ตามด้วยชั้นคอนโวลูชันขนาด  $3 \times 3$  จำนวน 12 ชั้น จากนั้นผ่านชั้น Transition Layer ชั้นที่ 2 ต่อมา Dense Block ที่ 3 ภายในบล็อกที่ 3 ประกอบด้วยชั้นคอนโวลูชันขนาด  $1 \times 1$  จำนวน 24 ชั้น ตามด้วยชั้นคอนโวลูชันขนาด  $3 \times 3$  จำนวน 24 ชั้น จากนั้นผ่านชั้น Transition Layer ชั้นที่ 3 และ Dense Block ที่ 4 ภายในบล็อกที่ 4 ประกอบด้วยชั้นคอนโวลูชันขนาด  $1 \times 1$  จำนวน 16 ชั้น ตามด้วยชั้นคอนโวลูชันขนาด  $3 \times 3$  จำนวน 16 ชั้น จากนั้นทำการรวมผลลัพธ์จาก Dense Block ทั้ง 4 บล็อกด้วยวิธี Global Average Pooling ขนาด  $7 \times 7$  และตามด้วยชั้นเชื่อมต่อสมบูรณ์และจำแนกข้อมูลด้วยฟังก์ชันซอฟต์แวร์แม็กซ์ จากสถาปัตยกรรมของ DenseNet121 จะเห็นว่าในทุกๆ บล็อกจะประกอบด้วยชั้นคอนโวลูชันขนาด  $1 \times 1$  ก่อนชั้นคอนโวลูชันขนาด  $3 \times 3$  เนื่องจากสามารถช่วยลดขนาดของแผนที่คุณสมบัติและเวลาการคำนวณได้

## 2.6 ResNet50

โครงข่ายประสาทเทียมแบบ Residual Network (ResNet) นำเสนอโดย Kaiming He และคณะ [22] ในปี 2015 สถาปัตยกรรมดั้งเดิมคือ ResNet34 ที่มีชั้นถ่วงน้ำหนัก 34 ชั้น เป็นโครงข่ายประสาทเทียมที่แก้ปัญหาเรื่องการหายไปของระดับสีระหว่างชั้นคอนโวลูชัน โดยใช้วิธีการเชื่อมต่อทางลัด กล่าวคือโครงข่ายประสาทเทียม ResNet ใช้วิธีการเชื่อมต่อข้อมูลจากชั้นนำเข้าไปยังชั้นที่อยู่ห่างกันออกไปสองชั้นหรือเรียกว่าการเชื่อมต่อทางลัด เพื่อข้ามไปยังอีกชั้นหนึ่งที่อยู่ห่างออกไปจากชั้นปัจจุบัน ในแต่ละชั้นจะมีขนาดของตัวกรองที่เท่ากันทั้งหมดขึ้นอยู่กับขนาดของแผนที่คุณสมบัติของผลลัพธ์ แต่หากมีขนาดของแผนที่คุณสมบัติลดลงจะทำให้มีขนาดของตัวกรองเพิ่มขึ้นเพื่อแก้ปัญหาเรื่องความซับซ้อนของการคำนวณ ชั้นถ่วงน้ำหนักของโครงข่าย ResNet34 แสดงดังรูปที่ 2.19



เท่ากับ 2 ซึ่งส่งผลให้สามารถลดจำนวนช่องสัญญาณให้เท่ากันและเชื่อมต่อกันแบบโดยตรงได้อีกครั้ง จากนั้นก่อนขั้นการเชื่อมต่อสมบูรณ์จะทำการรวมผลลัพธ์ด้วยค่าเฉลี่ย สถาปัตยกรรมของโครงข่ายแบบ ResNet มีให้ใช้หลายโครงข่าย แต่ละโครงข่ายจะแตกต่างกันตามจำนวนชั้นของบล็อก งานวิจัยนี้เลือกใช้โครงข่ายแบบ ResNet50 แสดงสถาปัตยกรรมดังรูปที่ 2.21

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		$1.8 \times 10^9$	$3.6 \times 10^9$	$3.8 \times 10^9$	$7.6 \times 10^9$	$11.3 \times 10^9$

รูปที่ 2.21 โครงข่ายประสาทเทียมแบบ ResNet50 [22]

จากรูปที่ 2.21 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ ResNet50 ประกอบด้วยชั้นคอนโวลูชันขนาด  $7 \times 7$  มีช่องสัญญาณเท่ากับ 64 มีค่าการเลื่อนเท่ากับ 2 จากนั้นทำการรวมผลลัพธ์ด้วยค่ามากที่สุด ขนาด  $3 \times 3$  มีค่าการเลื่อนเท่ากับ 2 บล็อกที่ 1 ประกอบด้วยชั้นคอนโวลูชันขนาด  $1 \times 1$  จำนวน 64 ช่องสัญญาณ คอนโวลูชันขนาด  $3 \times 3$  จำนวน 64 ช่องสัญญาณ และคอนโวลูชันขนาด  $1 \times 1$  จำนวน 256 ช่องสัญญาณ แต่ละชั้นมีการทำซ้ำ 3 รอบ บล็อกที่ 2 ประกอบด้วยชั้นคอนโวลูชันขนาด  $1 \times 1$  จำนวน 128 ช่องสัญญาณ คอนโวลูชันขนาด  $3 \times 3$  จำนวน 128 ช่องสัญญาณ และคอนโวลูชันขนาด  $1 \times 1$  จำนวน 512 ช่องสัญญาณ แต่ละชั้นมีการทำซ้ำ 4 รอบ บล็อกที่ 3 ประกอบด้วยชั้นคอนโวลูชันขนาด  $1 \times 1$  จำนวน 256 ช่องสัญญาณ คอนโวลูชันขนาด  $3 \times 3$  จำนวน 256 ช่องสัญญาณ และคอนโวลูชันขนาด  $1 \times 1$  จำนวน 1024 ช่องสัญญาณ แต่ละชั้นมีการทำซ้ำ 6 รอบ และบล็อกที่ 4 ประกอบด้วยชั้นคอนโวลูชันขนาด  $1 \times 1$  จำนวน 512 ช่องสัญญาณ คอนโวลูชันขนาด  $3 \times 3$  จำนวน 512 ช่องสัญญาณ และคอนโวลูชันขนาด  $1 \times 1$  จำนวน 2048 ช่องสัญญาณ แต่ละชั้นมีการทำซ้ำ 3 รอบ จะสังเกตได้ว่าชั้นแรกและ

ขั้นสุดท้ายของแต่ละบล็อกมีการปรับขนาดช่องสัญญาณให้เท่ากับบล็อกถัดไปเพื่อให้สามารถเชื่อมต่อทาง ลัดกันได้โดยตรง จากนั้นจึงรวมผลลัพธ์ด้วยค่าเฉลี่ยก่อนส่งต่อผลลัพธ์ไปยังขั้นการเชื่อมต่อสมบูรณ์

## 2.7 การสกัดคุณสมบัติรูปภาพ (Feature Extraction)

การทำงานด้านคอมพิวเตอร์วิทัศน์ (Computer Vision) คือการทำงานเกี่ยวกับการใช้รูปภาพ ดิจิตอลเพื่อให้คอมพิวเตอร์สามารถมองเห็น รับรู้ วิเคราะห์และแยกรูปภาพได้เสมือนกับมนุษย์ที่กำลัง ได้รับความนิยมนปัจจุบันได้มีนักวิจัยมากมายที่นำเสนอแนวคิดเกี่ยวกับคอมพิวเตอร์วิทัศน์เพื่อเพิ่ม ประสิทธิภาพการทำงานให้แม่นยำมากขึ้น เช่นแนวคิดเกี่ยวกับ การประมวลผลภาพ (Image Preprocessing) การแบ่งส่วนรูปภาพ (Image Segmentation) การสกัดคุณสมบัติของภาพ (Feature Extraction) และการจำแนกรูปภาพ (Image Classification) ขั้นตอนการสกัดคุณสมบัติรูปภาพเป็น ขั้นตอนที่สำคัญและส่งผลกระทบต่อประสิทธิภาพของแบบจำลอง (Model) กล่าวคือเป็นขั้นตอนการแปลงข้อมูล ให้อยู่ในรูปแบบที่คอมพิวเตอร์สามารถเข้าใจได้และสามารถนำไปใช้งานกับการเรียนรู้ของเครื่องได้ (Machine Learning) เช่น ข้อมูลที่เป็นข้อความ ข้อมูลรูปภาพ วิดีโอ เสียงพูด เป็นต้น หากสามารถสกัด คุณสมบัติของรูปภาพได้ถูกต้องและครบถ้วนก็จะทำให้แบบจำลองมีข้อมูลสำหรับการคำนวณที่ง่ายขึ้น วิธีการสกัดคุณสมบัติรูปภาพที่นิยมใช้กันมีดังนี้

### 2.7.1 คุณสมบัติสี (Color Feature)

การรับรู้ข้อมูลภาพของมนุษย์สามารถรับรู้และแยกรูปภาพได้ด้วยการมองเห็นสี ขนาด และรูปร่าง ของวัตถุที่เป็นส่วนประกอบของภาพ แต่คอมพิวเตอร์ไม่สามารถมองเห็นสีของภาพได้เช่นเดียวกับมนุษย์ ดังนั้นจึงต้องมีการใช้โมเดลสี (Color Model) เพื่อแยกคุณสมบัติ (Color Feature) เป็นคุณสมบัติของ รูปภาพที่เกิดจากการสกัดเอาข้อมูลสีออกมาจากภาพเพื่อแบ่งแยกวัตถุและขอบเขตของส่วนประกอบของ ภาพให้อยู่ในรูปแบบของตัวเลขเพื่อให้คอมพิวเตอร์สามารถเรียนรู้และแยกรูปภาพได้ แบบจำลองสีที่นิยม ใช้สำหรับงานด้านคอมพิวเตอร์วิทัศน์มีดังนี้

แบบจำลองสีอาร์จีบี (RGB Color Model) เป็นแบบจำลองสีที่ประกอบด้วยค่าสีแดง (Red) สีเขียว (Green) และสีน้ำเงิน (Blue) แต่ละสีจะมีอัตราส่วนเป็นค่าตัวเลขตั้งแต่ 0 ถึง 255 โดย 0 หมายถึงสีดำ และ 255 หมายถึงสีขาว แต่ละสีจะมีค่าสีเท่ากับ 256 ค่าสี และเมื่อผสมสีรวมกันจะได้ค่าสีทั้งหมด ประมาณ 16 ล้านสี ( $256 \times 256 \times 256 = 16,777,216$ ) แบบจำลองสีอาร์จีบี เกิดจากการผสมกันของสี แดง สีเขียว และสีน้ำเงิน เพื่อให้ได้เป็นสีต่างๆ ตามต้องการโดยการกำหนดอัตราส่วนของสีทั้ง 3 สีที่ แตกต่างกัน เช่น  $R = 255 \ G = 0 \ B = 0$  ผลลัพธ์คือสีแดง  $R = 255 \ G = 255 \ B = 0$  ผลลัพธ์คือสีเหลือง และ  $R = 0 \ G = 255 \ B = 255$  ผลลัพธ์คือสีฟ้า เป็นต้น แต่คอมพิวเตอร์สามารถแสดงค่าสีที่เกิดจากการ

ผสมกันของทั้ง 3 สี ได้เพียงแค่ 256 สี ดังนั้นจึงต้องทำการแปลงค่าสีให้อยู่ในรูปที่คอมพิวเตอร์สามารถแสดงออกมาได้ในรูปภาพแบบระดับสีเทา คำนวณได้จากสมการที่ 2.1

$$Gray = (0.3 \times R) + (0.59 \times G) + (0.11 \times B) \quad (2.1)$$

แบบจำลองสี YCbCr (YCbCr Color Model) เป็นแบบจำลองสีที่นิยมใช้สำหรับงานที่เกี่ยวข้องกับรูปภาพ วิดีโอ ที่แสดงผลผ่านทางอุปกรณ์อิเล็กทรอนิกส์ เช่น คอมพิวเตอร์และโทรทัศน์ดิจิทัล แบบจำลองสี YCbCr เป็นแบบจำลองสีที่เลียนแบบการมองเห็นด้วยตาของมนุษย์ ที่มีค่าความสว่างเข้ามาช่วยในการแบ่งแยกสี โดยแบบจำลองสี YCbCr แปลงค่าสีจากอาร์จีบี ให้มีค่าความสว่างเพิ่มมากขึ้น ประกอบด้วย Y คือค่าความสว่าง (Luminance) แทนค่าความเข้มของแสงที่แยกออกมาจากค่าสีแดง สีเขียว และสีน้ำเงิน Cb คือค่าความแตกต่างของสีน้ำเงิน ใช้สำหรับเข้ารหัสสีน้ำเงิน และ Cr คือค่าความแตกต่างของสีแดง ใช้สำหรับการเข้ารหัสสีแดง จากนั้นจึงทำการเข้ารหัสรวมกันด้วย YCbCr การแปลงแบบจำลองสีจากอาร์จีบี เป็น YCbCr คำนวณได้จากสมการต่อไปนี้

$$Y = (0.299 \times R) + (0.587 \times G) + (0.114 \times B) \quad (2.2)$$

$$Cb = (-0.169 \times R) - (0.331 \times G) + (0.5 \times B) \quad (2.3)$$

$$Cr = (0.5 \times R) - (0.419 \times G) - (0.081 \times B) \quad (2.4)$$

จากสมการ กำหนดให้ R คือค่าของสีแดง G คือค่าของสีเขียว และ B คือค่าของสีน้ำเงิน ที่มีค่าตั้งแต่ 0 ถึง 255 ซึ่งจะเห็นได้ว่าแบบจำลอง YCbCr จะไม่มีการคำนวณของสีเขียวเนื่องจากแบบจำลองมีช่องสัญญาณของความสว่าง Y และค่าสีจาก Cb และ Cr อยู่แล้วจึงไม่จำเป็นต้องคำนวณค่าสีของสีเขียวซึ่งสามารถช่วยประหยัดเวลาของการแปลงค่าสีลงได้

แบบจำลองสี HSV (HSV Color Model) เป็นแบบจำลองสีที่ใช้ความอิ่มตัวของสีเพื่อแบ่งแยกสีที่ใกล้เคียงกันหรือเฉดสีเดียวกันออกจากกัน ซึ่งจะให้ความรู้สึกได้ดีกว่า แบบจำลองประกอบด้วยค่า H ย่อมาจาก Hue, S ย่อมาจาก Saturation และ V ย่อมาจาก Value

H คือค่าที่ใช้บอกมุมมองสีที่ใช้บอกเฉดของสี มีค่าตั้งแต่ 0 ถึง 360 องศา โดยองศา 0 – 60 แทนด้วยเฉดสีแดง 60 – 120 แทนด้วยเฉดสีเหลือง 120 – 180 แทนด้วยเฉดสีเขียว 180 – 240 แทนด้วยเฉดสี Cyan 240 – 300 แทนด้วยเฉดสีน้ำเงิน และ 300 – 360 แทนด้วยเฉดสี Magenta

S คือค่าความอิ่มตัวของสี ยิ่งสีมีความอิ่มตัวมากเท่าไรก็จะทำให้สีนั้นมีความเข้มมากขึ้น และหากสีมีความอิ่มตัวย่น สีนั้นก็จะมีค่าความสว่างมากขึ้น โดยค่าความอิ่มตัวของสีนี้จะทำให้สามารถแยก

ลักษณะของสีที่แตกต่างกันได้อย่างชัดเจน เช่น สีแดงกับสีชมพู ที่มีเฉดสีคือสีแดงเหมือนกัน แต่สีแดงจะมีค่าความอิ่มตัวของสีมากกว่าของสีชมพู หรือระหว่างสีฟ้ากับสีน้ำเงิน สีฟ้าอ่อนมีค่าความอิ่มตัวของเฉดสีน้อยกว่าสีน้ำเงิน เป็นต้น ถ้าความอิ่มตัวของสีเท่ากับ 100% หมายถึงมีการเติมสีทั้งหมด แต่ถ้าความอิ่มตัวของสีเท่ากับ 0% หมายความว่าไม่มีการเติมสีใดๆ จึงทำให้รูปภาพเป็นเฉดสีเทาหรือสีดำ

V คือค่าความสว่างที่เกี่ยวข้องกับค่าอิ่มตัวของสี หากค่า V มีค่าเท่ากับ 0 จะหมายถึงสีดำไม่ว่าจะมีค่า H และ S หรือไม่ และ V มีค่าสูงสุดจะหมายถึงสีขาว

### 2.7.2 คุณสมบัติรูปร่าง (Shape Feature)

คุณสมบัติรูปร่าง (Shape Feature) หรือ คุณสมบัติรูปร่างทางสัณฐานวิทยา (Geometric Shape Feature) เป็นคุณสมบัติของรูปภาพด้านคุณลักษณะทางเรขาคณิตที่ช่วยให้คอมพิวเตอร์และการเรียนรู้ของเครื่องสามารถทำงานด้านการประมวลผลภาพและคอมพิวเตอร์วิทัศน์ได้ โดยการสกัดคุณสมบัติด้วยวิธีนี้จะทำการกำหนดจุดต่างๆ บนรูปภาพเพื่อให้ได้เป็นรูปร่างทางเรขาคณิต จากนั้นจึงทำการคำนวณพื้นที่ภายในจุดที่กำหนด ซึ่งเมื่อนำคุณสมบัติที่ได้จากการสกัดด้วยวิธีนี้ไปใช้กับแบบจำลองการเรียนรู้ของเครื่องจะทำให้สามารถเรียนรู้และจำแนกรูปภาพได้ว่าภายในรูปภาพนั้นประกอบด้วยวัตถุใดบ้าง วัตถุแต่ละชิ้นมีรูปร่างและขนาดเท่าไร ในงานวิจัยนี้ขอยกตัวอย่างการคำนวณคุณสมบัติรูปร่างทางสัณฐานวิทยาของใบพืชเนื่องจากงานวิจัยดำเนินงานเกี่ยวกับโรคของใบพืช แบ่งการคำนวณออกเป็น 2 แบบ คือ คุณสมบัติทางเรขาคณิตพื้นฐาน (Basic Geometrical Features) และลักษณะทางสัณฐานวิทยาตามรูปร่าง (Shape based Morphological feature) [23]

คุณสมบัติทางเรขาคณิตพื้นฐาน (Basic Geometrical Features) สามารถคำนวณได้ ดังนี้

*เส้นผ่านศูนย์กลาง (Diameter: D)* เป็นการคำนวณระยะทางเส้นผ่านศูนย์กลางของใบพืช โดยใช้ระยะทางที่ยาวที่สุดระหว่างสองพิกเซลที่อยู่บนขอบของใบพืช

*ความยาวทางสรีรวิทยา (Physiological Length:  $L_p$ )* เป็นการคำนวณความยาวแกนหลักของวงรีที่อยู่รอบใบพืช

*ความกว้างทางสรีรวิทยา (Physiological Width:  $W_p$ )* เป็นการคำนวณความกว้างแกนหลักของวงรีที่อยู่รอบใบพืช

*พื้นที่ของใบ (Leaf Area: A)* เป็นการคำนวณพื้นที่ทั้งหมดของใบพืช

ขนาดเส้นรอบวงของใบ (*Leaf Perimeter: P*) เป็นการคำนวณระยะของเส้นรอบวงของใบพืชโดยวัดจากส่วนที่เป็นขอบใบพืช

ลักษณะทางสัณฐานวิทยาตามรูปร่าง (*Shape based Morphological feature*) สามารถคำนวณได้ ดังนี้

*Smooth Factor* คือการคำนวณผลกระทบของสัญญาณรบกวนที่ใช้เพื่อแสดงถึงความเรียบของใบไม้เมื่อใช้ตัวกรองที่มีขนาดต่างกันมากกว่าหนึ่งตัว

*Aspect Ratio* คือการคำนวณอัตราส่วนของความยาวทางสรีรวิทยาต่อความกว้างทางสรีรวิทยา คำนวณได้จากสมการที่ 2.5

$$AR = \frac{L_P}{W_P} \quad (2.5)$$

*Form Factor* คือการคำนวณความแตกต่างระหว่างใบพืชกับวงกลมที่ล้อมรอบใบพืช คำนวณได้จากสมการที่ 2.6

$$Form_F = \frac{4 \times \pi \times A}{P^2} \quad (2.6)$$

*Rectangularity* คือการคำนวณความคล้ายคลึงกันระหว่างพื้นที่ใบพืชกับสี่เหลี่ยมผืนผ้าที่ล้อมรอบใบพืช คำนวณได้จากสมการที่ 2.7

$$Rectangularity = \frac{L_P \times W_P}{A} \quad (2.7)$$

*Narrow Factor* คือการคำนวณอัตราส่วนของเส้นผ่านศูนย์กลางต่อความยาวทางสรีรวิทยาของใบพืช คำนวณได้จากสมการที่ 2.8

$$Narrow_F = \frac{D}{L_P} \quad (2.8)$$

*Ratio of Perimeter to Diameter* คือการคำนวณอัตราส่วนของเส้นรอบวงต่อเส้นผ่านศูนย์กลาง คำนวณได้จากสมการที่ 2.9

$$P_D = \frac{P}{D} \quad (2.9)$$

*Perimeter ratio of physiological length and physiological width* คือการคำนวณอัตราส่วนของเส้นรอบวงต่อผลรวมของความยาวทางสรีรวิทยาและความกว้างทางสรีรวิทยา คำนวณได้จากสมการที่ 2.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$P_{LW} = \frac{P}{L_P + W_P} \quad (2.10)$$

### 2.7.3 คุณสมบัติพื้นผิว (Texture Feature)

การสกัดคุณสมบัติของรูปภาพอีกวิธีหนึ่งที่นิยมใช้สำหรับงานด้านการจำแนกรูปภาพคือ การสกัดคุณสมบัติพื้นผิว (Texture Feature) เพื่อคำนวณหาค่าความต่างของความราบเรียบและเนื้อละเอียดของภาพเพื่อใช้แยกวัตถุที่อยู่ในรูปภาพ โดยปกติการสกัดคุณสมบัติพื้นผิวของรูปนั้น รูปภาพนำเข้าจะต้องอยู่ในรูปแบบภาพระดับสีเทา เนื่องจากการวิธีนี้จะวัดความแตกต่างจากระดับสีเทาที่อยู่ในรูปภาพ งานวิจัยนี้ขออธิบายวิธีการสกัดคุณสมบัติพื้นผิวรูปภาพด้วย Gray Level Co-occurrence Matrix (GLCM) โดย GLCM จะคำนวณค่าความละเอียดหรือความหยาบของรูปภาพจากภาพระดับสีเทา โดยอาศัยพิกเซลบนภาพ 2 พิกเซลคือ พิกเซลอ้างอิงและพิกเซลใกล้เคียงที่มีความเข้มของสีเทา ค่าความน่าจะเป็นของทิศทางของพิกเซล และระยะห่าง คุณสมบัติ GLCM คำนวณได้จากค่า Contrast, Dissimilarity, Inverse Different Moment, Angular Second Moment, Energy, Correlation และ Sum of Squares Variance

## 2.8 การประมวลผลรูปภาพ (Image Processing)

การประมวลผลรูปภาพ (Image Processing) เป็นขั้นตอนสำคัญในการปรับปรุงคุณภาพของภาพก่อนจะนำเข้ารูปภาพให้กับแบบจำลองการเรียนรู้ หากรูปภาพที่นำเข้ามีคุณภาพดีก็จะทำให้สามารถดึงคุณสมบัติของรูปภาพได้ครบถ้วนและถูกต้องมากขึ้น เพราะแบบจำลองการเรียนรู้ด้วยเครื่องจะเรียนรู้รูปภาพจากคุณสมบัติของภาพ ดังนั้นหากรูปภาพมีความคมชัด ไม่มีสัญญาณรบกวนก็จะทำให้สามารถสกัดคุณสมบัติเด่นของภาพได้แม่นยำมากขึ้น การประมวลผลรูปภาพมีด้วยกันหลายวิธี ในงานวิจัยนี้อธิบายการประมวลผลภาพเฉพาะในส่วนของการแปลงค่าสีของภาพ (Color Transform) การแปลงทางสัณฐานวิทยา (Morphological Transformations) การตรวจจับขอบภาพ (Edge Detection) และการหาจุดเด่นบนภาพด้วย Scale-Invariant Feature Transform (SIFT) ที่เกี่ยวข้องกับการจำแนกรูปใบพืช

*การแปลงค่าสีภาพ (Color Transform)* เป็นขั้นตอนการประมวลผลรูปภาพ โดยการเปลี่ยนแบบจำลองสีของภาพจากรูปภาพที่มีแบบจำลองสีแบบอาร์จีบี เป็นภาพระดับสีเทาและทำการแปลงภาพระดับสีเทาเป็นภาพแบบไบนารี โดยใช้การปรับค่าเกณฑ์ต่ำสุดและเกณฑ์สูงสุดของ threshold ให้เหมาะสมกับรูปภาพ และมีการใช้ค่า Otsu's threshold สำหรับการกำหนดค่าเกณฑ์ threshold แบบอัตโนมัติเพื่อให้ภาพเป็นไบนารี

การเปลี่ยนแปลงทางสัณฐานวิทยา (*Morphological Transformations*) การแปลงค่าสีของรูปภาพบางครั้งอาจทำให้ได้ผลลัพธ์เป็นภาพไบนารีที่รูปภาพยังคงมีสัญญาณรบกวน (*Noise*) ของรูปภาพ การเปลี่ยนแปลงทางสัณฐานวิทยาจึงใช้เพื่อลบสัญญาณรบกวนของภาพ เป็นกระบวนการประมวลผลภาพล่วงหน้าโดยใช้ลักษณะทางสัณฐานวิทยาของรูปภาพซึ่งจะทำงานกับภาพที่เป็นไบนารี โดยจะมีการกำหนดค่า structuring element หรือค่าเคอร์เนลที่เป็นพารามิเตอร์ การเปลี่ยนแปลงทางสัณฐานวิทยา มีให้เลือกใช้งานหลายฟังก์ชัน เช่น Erosion, Dilation, Opening, Closing, Morphological Gradient, Top Hat, และ Black Hat ในงานวิจัยนี้ใช้ฟังก์ชัน Opening และ Closing ซึ่งมีหลักการทำงานพื้นฐานมาจาก Erosion และ Dilation โดยจะดำเนินการกับรูปภาพที่เป็นไบนารี

Opening operation จะมีขั้นตอนการทำงานคล้ายกับ Erosion แต่มีการกัดเซาะน้อยกว่า โดยจะทำการลบพิกเซลด้านหน้าที่มีความสว่างบางส่วนที่ไม่ต้องการออกจากขอบเขตของรูปภาพ ด้านหน้าแต่ยังคงต้องรักษาพื้นที่ส่วนหน้าให้มีลักษณะคงเดิมคล้ายกับรูปภาพนำเข้า ซึ่งผลลัพธ์ที่ได้ขึ้นอยู่กับค่า structuring element ซึ่งงานวิจัยโดยงานวิจัยนี้ใช้ *Opening operation* สำหรับขยายเส้นขอบของใบพืชให้เชื่อมต่อกันและลบบางพิกเซลของขอบใบที่มีผลมาจากแสงและเงาของใบพืช

Closing operation มีขั้นตอนการทำงานคล้ายกับ Dilation กล่าวคือ Closing จะทำการขยายพื้นที่ด้านหน้าในภาพเพื่อลดพื้นที่ด้านหลังของพื้นที่นั้นๆ เพื่อเป็นการปิดรูรั่วที่อยู่บนรูปภาพให้มีลักษณะเป็นพื้นที่ด้านหน้าที่สมบูรณ์ โดยงานวิจัยนี้ใช้ *Closing operation* สำหรับลบสัญญาณรบกวนของภาพที่อยู่บนรูปภาพใบพืชที่ได้มาหลังจากการแปลงภาพสีเทาเป็นไบนารี

การตรวจจับขอบภาพ (*Edge Detection*) เป็นวิธีการหาเส้นขอบที่เป็นของของวัตถุที่อยู่ในรูปภาพ โดยพิจารณาจากความต่างของความเข้มของสีจากจุดหนึ่งไปยังอีกจุดหนึ่งที่มีความต่อเนื่องกัน เพื่อให้สามารถคำนวณขนาดพื้นที่หรือจดจำรูปร่างของวัตถุได้ มีการนำเทคนิคการตรวจจับขอบภาพไปใช้สำหรับงานด้านการตรวจจับรูปภาพหรือวัตถุ งานด้านการจำแนกรูปภาพ เป็นต้น การตรวจจับขอบภาพมิให้เลือกใช้หลายวิธี งานวิจัยนี้เลือกใช้การตรวจจับขอบภาพด้วยแคนนี่ (*Canny Edge Detection*) การหาขอบภาพด้วยแคนนี่จะสามารถทำงานได้ดีกับรูปภาพแบบไบนารี (*Binary Image*) โดยจะทำการปรับค่าสีของแต่ละพิกเซลด้วยตัวกรองแบบ Guassian จากนั้นจะคำนวณหาขนาดและทิศทางของการเปลี่ยนแปลงค่าสีเพื่อกำหนดพิกเซลที่คาดว่าจะจะเป็นขอบของภาพโดยจะพิจารณาจากความเข้มของสี

การหาจุดเด่นบนภาพด้วย *Scale-Invariant Feature Transform (SIFT)* พัฒนาโดย Lowe, D.G. และคณะ [24] เป็นอัลกอริทึมสำหรับการค้นหาจุดสำคัญบนรูปภาพเพื่อทำการเปรียบเทียบความ

เหมือนกันของภาพ โดยนำจุดสำคัญเหล่านั้นใช้สำหรับการคำนวณหาคุณลักษณะเด่นของภาพที่ไม่ขึ้นกับขนาด ตำแหน่ง และทิศทางของภาพ ซึ่งช่วยให้สามารถแยกคุณสมบัติเด่นออกจากรูปภาพได้ โดยมีการทำงาน 4 ขั้นตอนคือ

1) Scale-space Extrema Detection เป็นขั้นตอนการค้นหาค่าจุดสำคัญบนรูปภาพ โดยรูปภาพที่รับเข้ามาจะผ่านการเบลอภาพด้วยเกาส์เซียน (Gaussian Blur) ซึ่งภาพจะมีความเบลอแตกต่างกันตามค่า  $\sigma$  ของแต่ละมาตราส่วน (Scale) ซึ่งจะทำให้ได้ค่าความแตกต่างเกาส์เซียน (Difference of Gaussian: DoG) และจัดเก็บเป็นพีระมิดเกาส์เซียน (Gaussian Pyramid) ซึ่งถือว่าเป็นจุดที่คาดว่าจะป็นจุดสำคัญคำนวณได้จากสมการที่ 2.11

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (2.11)$$

โดยที่  $G(x, y, \sigma)$  คือ Gaussian Kernel

$I(x, y)$  คือ ค่าพิกเซลที่ตำแหน่ง ของรูปภาพ

$\sigma$  คือ ค่าความกว้างของ Gaussian Kernel

2) Keypoint Localization เป็นขั้นตอนการกำหนดจุดสำคัญ เมื่อเจอค่าความแตกต่างเกาส์เซียนก็จะทำการเปรียบเทียบกับพิกเซลที่อยู่ใกล้เคียงรอบๆ จำนวน 8 พิกเซลในมาตราส่วนเดียวกันและจำนวน 9 พิกเซลในมาตราส่วนก่อนหน้าและมาตราส่วนถัดไป รวม 26 พิกเซล หากพิกเซลที่มีค่ามากที่สุดหรือน้อยสุดในระดับมาตราส่วนเดียวกันก็จะถูกกำหนดให้เป็นจุดสำคัญคู่แข่ง (Candidate Keypoints) จากนั้นทำการเปรียบเทียบจุดสำคัญใดที่มีค่าความแตกต่างต่ำและเป็นจุดสำคัญที่เส้นขอบก็จะถูกลบออกเพื่อลดจำนวนของจุดสำคัญ

3) Orientation Assignment เป็นขั้นตอนการคำนวณขนาดและทิศทางของการไล่ระดับสี (Gradient) โดยการกำหนดจุดสำคัญแต่ละจุดให้มีค่าคงที่สำหรับการหมุนภาพ จุดสำคัญจะถูกนำไปคำนวณกับพื้นที่ใกล้เคียงโดยขึ้นกับมาตราส่วนและขนาดและทิศทางของการไล่ระดับสีของแต่ละพื้นที่ เก็บไว้ในฮิสโตแกรม (Histogram) ขนาด 36 ครอบคลุม 360 องศา โดยให้ค่าน้ำหนักตามการไล่ระดับสีและถ่วงน้ำหนักแบบเกาส์เซียน จุดสูงสุดของฮิสโตแกรมที่มีค่ามากกว่า 80% ของจุดสูงสุดถือว่าเป็นการวางแนวใหม่ การสร้างจุดสำคัญที่มีตำแหน่งและขนาดที่เหมือนกัน แต่ทิศทางต่างกันจะทำให้การจับคู่ภาพมีประสิทธิภาพมากขึ้น คำนวณได้จากสมการที่ 2.12 และ 2.13

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad (2.12)$$

$$\theta(x, y) = \tan^{-1} \left( \frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)} \right) \quad (2.13)$$

4) Keypoint Descriptor เป็นขั้นตอนการสร้างตัวอธิบายจุดสำคัญโดยการใช้พื้นที่ใกล้เคียงจุดสำคัญที่มีขนาด  $16 \times 16$  เพื่อเป็นตัวกำหนดทิศทาง ซึ่งแบ่งออกเป็นขนาด  $16$  บล็อกที่มีขนาด  $4 \times 4$  บล็อกย่อย และมีการสร้างฮิสโตแกรมขนาด  $8$  สำหรับเก็บข้อมูลขนาดและทิศทางของการไล่ระดับสี  $8$  ทิศทาง ดังนั้นทำให้ได้เวกเตอร์ขนาด  $4 \times 4 \times 8$  หรือ  $128$  สำหรับคุณสมบัติแต่ละตัว

## 2.9 การจำแนกรูปภาพ (Image Classification)

การจำแนกรูปภาพ (Image Classification) หรือการจัดหมวดหมู่รูปภาพจัดเป็นการเรียนรู้แบบมีผู้สอน (Supervised Learning) กล่าวคือ แบบจำลองหรือโมเดลด้านการจำแนกรูปภาพจำเป็นต้องมีชุดของข้อมูลรูปภาพจำนวนหนึ่งที่ใช้เป็นชุดเรียนรู้ (Training Set) สำหรับการฝึกสอนให้แบบจำลองรู้จักว่ารูปภาพมีรูปร่างอะไรบ้าง แบ่งออกเป็นกี่หมวดหมู่ จากนั้นจะมีชุดข้อมูลอีกหนึ่งชุดเรียกว่าชุดทดสอบ (Testing Set) เพื่อทำการทดสอบให้แบบจำลองจำแนกรูปภาพออกมาเป็นหมวดหมู่ แบบจำลองการจำแนกรูปภาพมีทั้งแบบโครงข่ายประสาทเทียม (Neural Network) (ได้อธิบายไว้ในหัวข้อ 2.1 – 2.6) และแบบจำลองการเรียนรู้ของเครื่อง (Machine Learning) งานวิจัยนี้มีการใช้แบบจำลองการจำแนกรูปภาพทั้งแบบโครงข่ายประสาทเทียมและการเรียนรู้ของเครื่อง ในส่วนนี้จะขออธิบายแบบจำลองการจำแนกรูปภาพแบบการเรียนรู้ของเครื่องด้วยโมเดลแบบป่าสุ่ม (Random Forest Model)

โมเดลแบบป่าสุ่ม (Random Forest Model) เป็นโมเดลการจำแนกข้อมูลแบบป่าสุ่ม กล่าวคือ โมเดลจะประกอบด้วยต้นไม้ตัดสินใจ (Decision Tree) หลายๆ ต้นมาทำงานร่วมกัน โดยต้นไม้ตัดสินใจแต่ละต้นจะได้รับชุดข้อมูลสำหรับการเรียนรู้ที่เหมือนกัน แต่ใช้ชุดข้อมูลสำหรับการเรียนรู้ที่ไม่เหมือนกัน และต้นไม้ตัดสินใจแต่ละต้นจะตัดสินใจว่าชุดข้อมูลที่ได้รับนั้นควรอยู่ในหมวดหมู่หรือประเภทใด โดยต้นไม้ตัดสินใจแต่ละต้นต้องเป็นอิสระจากกันและสามารถตัดสินใจได้ด้วยตัวเอง จากนั้นจึงนำผลจากต้นไม้ตัดสินใจทั้งหมดมาโหวตเพื่อหาคำตอบว่าต้นไม้ตัดสินใจทั้งหมดระบุให้ชุดข้อมูลอยู่ในหมวดหมู่หรือประเภทใดมากที่สุดก็จะเป็นคำตอบให้กับโมเดลแบบป่าสุ่ม

## 2.10 งานวิจัยที่เกี่ยวข้อง

ในส่วนนี้นำเสนอบทความที่เกี่ยวข้องกับการจำแนกโรคของพืชโดยใช้การสกัดคุณสมบัติด้วยการจำแนกข้อมูลด้วยการเรียนรู้ของเครื่อง และการปรับปรุงสถาปัตยกรรมของ CNN ในรูปแบบต่าง ๆ โดยงานวิจัยที่ไม่มีการใช้ CNN สำหรับการจำแนกโรคของใบพืชต้องอาศัยการสกัดคุณสมบัติของใบพืชด้วยมือ [25] เช่น คุณสมบัตีสี คุณสมบัติน้ำหนัก คุณสมบัติพื้นผิว เป็นต้น แต่หากใช้ CNN จะมีการสกัดคุณสมบัติของภาพโดยอัตโนมัติ

Monika Bansal และคณะ [26] นำเสนอการเปรียบเทียบประสิทธิภาพการสกัดคุณสมบัติด้วยวิธี SIFT, SURF และ ORB และวิเคราะห์ผลการเปรียบเทียบระหว่างการใช้คุณสมบัติแต่ละตัวและการใช้คุณสมบัติแบบผสมผสาน สำหรับการรู้จำวัตถุ 2 มิติ ทดลองกับชุดข้อมูล Caltech-101 ลดมิติของคุณสมบัติที่สนใจด้วย Locality preserving projection (LPP) จากนั้นจำแนกประเภทของรูปภาพด้วย K-Nearest Neighbor, Naïve Bayes, Decision Tree และ Random Forest เพื่อเปรียบเทียบประสิทธิภาพโดยใช้คุณสมบัติแบบร่วมกันระหว่าง SIFT, SURF และ ORB ผลการวิเคราะห์พบว่าการใช้วิธีการสกัดคุณสมบัติด้วย SIFT, SURF และ ORB ร่วมกับ Random Forest ให้ผลลัพธ์ที่ดีที่สุด มีความแม่นยำร้อยละ 83.27

Payal Chhabra และคณะ [27] นำเสนอการค้นคืนรูปภาพโดยใช้วิธีการสกัดคุณสมบัติของรูปภาพและค้นคืนรูปภาพตามเนื้อหา ใช้การสกัดคุณสมบัติของภาพด้วย Oriented Fast and Rotated BRIEF (ORB) และ Scale-Invariant Feature Transform (SIFT) ใช้ K-mean สำหรับการกำหนดจำนวนกลุ่มของชุดข้อมูล และใช้ Locality preserving projection (LPP) สำหรับการลดมิติของคุณสมบัติ ทดลองกับชุดข้อมูล Wang database และ Corel database ผลการทดลองพบว่าการใช้คุณสมบัติ ORB ร่วมกับ SIFT และจำแนกประเภทด้วย Random Forest ให้ค่าความแม่นยำสูงที่สุดร้อยละ 99.53 สำหรับชุดข้อมูล Coral database และร้อยละ 86.20 สำหรับชุดข้อมูล Wang database

Surbhi Gupta และคณะ [28] ปรับปรุงการรู้จำวัตถุโดยใช้การสกัดคุณสมบัติของภาพด้วย Oriented Fast and Rotated BRIEF (ORB) และ Scale-Invariant Feature Transform (SIFT) ใช้ K-means Clustering สำหรับการจัดกลุ่มข้อมูลและลดมิติของคุณสมบัติด้วย Locality Preserving Projection (LPP) ทดลองกับชุดข้อมูล Caltech 101 ผลการทดลองพบว่าการใช้เวกเตอร์คุณสมบัติที่ผสมผสานคุณสมบัติที่สกัดด้วยวิธี ORB และ SIFT จำนวน 16 คุณสมบัตินี้ร่วมกับการจำแนกประเภทด้วย Random Forest มีอัตราความแม่นยำสูงที่สุดร้อยละ 85.60

Gayathri Devi และคณะ [29] นำเสนอการตรวจจับใบข้าวที่เป็นโรคและไม่เป็นโรคจำนวน 6 คลาส โดยใช้เทคนิคการประมวลผลภาพ แบ่งส่วนรูปภาพส่วนใบข้าวที่เป็นโรคด้วย K-means Clustering จากนั้นทำการสกัดคุณสมบัติของภาพโดยใช้คุณสมบัติแบบผสมผสานที่ได้จากการสกัดคุณสมบัตินิรूपภาพ จาก SIFT, DWT และ GLCM ผลการทดลองพบว่าการใช้คุณสมบัตินิรूपภาพจากการสกัดคุณสมบัตินิรूपภาพด้วย SIFT, DWT และ GLCM ร่วมกับการจำแนกประเภทด้วย Multiclass SVM มีค่าความแม่นยำสูงที่สุด ร้อยละ 98.63

Lu Zhang และคณะ [30] นำเสนอการระบุชนิดของใบพืชโดยใช้การวิเคราะห์องค์ประกอบหลัก และการจำแนกแบบเชิงเส้น โดยแปลงภาพสีเป็นภาพระดับสีเทาและภาพไบนารีโดยใช้ค่า OTSU's Thresholding Method และ Morphology Method (Opening Operation และ Closing Operation) จากนั้นสกัดคุณสมบัตินิรूपภาพ โดยใช้คุณสมบัตินิรूपภาพ (Geometric Feature, Hu Moment Invariants Features และ Structural Characteristics) คุณสมบัตินิรूपภาพ (GLCM, Fractal Dimension, LBP และ Gabor features) จากนั้นลดมิติของคุณสมบัตินิรूपภาพโดยการประยุกต์ใช้ CPA ร่วมกับ LDA และจำแนกใบพืช ด้วย Back Propagation Neural Network ทดลองกับชุดข้อมูล Flavia Leaf Image และ ICL Leaf Image Datasets ผลการทดลองพบว่าการใช้วิธีสกัดคุณสมบัตินิรूपภาพหลายตัวจากรูปภาพ (GC + Hu + SC + GLCM + FD + LBP + Gabor) ร่วมกับการลดมิติคุณสมบัตินิรूपภาพด้วย CPA และ LDA และจำแนกประเภทด้วย BPNN ให้ค่าความแม่นยำสูงที่สุดคือร้อยละ 94.22 สำหรับชุดข้อมูล Flavia และร้อยละ 87.82 สำหรับชุดข้อมูล ICL

Satti R. G. Reddy และคณะ [31] นำเสนอการจำแนกชนิดของพืชด้วยโครงข่ายประสาทเทียม แบบคอนโวลูชัน ใช้คุณสมบัตินิรूपภาพและคุณสมบัตินิรूपภาพ โครงข่ายประสาทเทียมที่นำเสนอประกอบด้วย ชั้นคอนโวลูชันจำนวน 4 ชั้น ตามด้วยชั้นการรวมผลลัพธ์ 4 ชั้น และชั้นการเชื่อมต่อสมบูรณ์ 2 ชั้น ใช้ฟังก์ชันซอฟต์แวร์แม็ก สำหรับการจำแนกประเภท ทดลองกับ 5 ชุดข้อมูลคือ Leaf Snap, UCI Leaf, PlantVillage, Flavia, และ Swedish ผลการทดลองพบว่าโครงข่ายประสาทเทียมที่นำเสนอสามารถสกัดคุณสมบัตินิรूपภาพได้ดีกว่าการสกัดคุณสมบัตินิรूपภาพด้วยมือ และมีค่าความแม่นยำในการจำแนกข้อมูลดีกว่าจากการทดลองกับชุดข้อมูล Flavia, Swedish, UCI leaf, PlantVillage และ Leaf snap มีค่าความแม่นยำร้อยละ 100, 100, 100, 89.99, 97.99 ตามลำดับ

Jagadeesh Basavaiah และคณะ [32] นำเสนอการเทคนิคการระบุโรคของใบมะเขือเทศ โดยปรับปรุงความแม่นยำในการจำแนกและลดเวลาการประมวลผล ด้วยการใช้หลายคุณสมบัตินิรूपภาพร่วมกันคือ Color histograms, Hu Moments, Haralick และ Local Binary Pattern features ทดลองกับชุด

ข้อมูลโคมะเชื้อเทศที่เป็นโรคและไม่เป็นโรคจำนวน 5 คลาส จำแนกข้อมูลด้วย Random Forest และ Decision Tree ผลการทดลองพบว่าการจำแนกข้อมูลด้วย Random Forest มีค่าความแม่นยำร้อยละ 94 สูงกว่าการจำแนกข้อมูลด้วย Decision Tree มีความแม่นยำร้อยละ 90

Changjian Zhou และคณะ [33] นำเสนอแบบจำลองสำหรับการระบุโรคของโคมะเชื้อเทศ โดยการปรับปรุง Residual Dense Network เป็น Restructure Deep Residual Dense Network เป็นการพัฒนาโดยรวม Deep Residual Networks และ Dense Networks เข้าด้วยกันเพื่อสกัดคุณสมบัติของภาพผ่าน Dense Connected Convolutional Layers ซึ่ง RDB จะมีการเชื่อมต่อโดยตรงไปยังทุกชั้นถัดไปโดยรูปภาพนำเข้าจะถูกรวมเข้าที่ Res-Dense-Block (RDB) หลังจากนั้นจะมีการเพิ่มชั้น BatchNormalized ตามหลังชั้นคอนโวลูชันและมีการเพิ่มฟังก์ชัน LeakyReLU Activity Function ส่วนชั้น Residual Concatenate Tensor เป็นการรวม RDB Block ที่ปรับปรุงแล้วกับชั้นนำเข้าข้อมูลเข้าด้วยกัน ทดลองกับชุดข้อมูลโคมะเชื้อเทศจาก AI CHALLENGER Dataset จำนวน 13,185 รูปภาพ ผลการทดลองพบว่าวิธีการที่นำเสนอให้ค่าความแม่นยำในการจำแนกโรคโคมะเชื้อเทศคือร้อยละ 95 และให้ค่าความแม่นยำสูงสุดเมื่อเปรียบเทียบกับโครงข่ายประสาทเทียมอื่น เช่น Deep CNN, ResNet50 และ DenseNet121 มีค่าความแม่นยำร้อยละ 93.21, 88.49, 91.96 ตามลำดับ

Alain B. Djimeli-Tsajio และคณะ [34] นำเสนอวิธีการตรวจหาและระบุโรคของมะเชื้อเทศแบบอัตโนมัติโดยใช้โครงข่ายประสาทเทียมที่มีการถ่ายโอนการเรียนรู้จากแบบจำลอง ResNet101 และ ResNet152 เพื่อสกัดคุณสมบัติของรูปภาพโคมะเชื้อเทศ จากนั้นจำแนกโรคของโคมะเชื้อเทศด้วย Multi-Layer Perceptron การถ่ายโอนการเรียนรู้จากการฝึกฝนล่วงหน้าของ ResNet101 และ ResNet152 ทำให้ได้คุณสมบัติรูปร่าง สี และคุณสมบัติอื่นๆ ที่มีขนาดเวกเตอร์ของคุณสมบัติที่แตกต่างกัน จากนั้นจึงทำการปรับและรวมคุณสมบัติของรูปภาพที่ได้จาก ResNet101 และ ResNet152 โดยพิจารณาจากความแปรปรวนและส่วนเบี่ยงเบนเฉลี่ยของคุณสมบัติ ค่าเฉลี่ยของคุณสมบัติทั้งสอง และการเชื่อมต่อคุณสมบัติทั้งสองเข้าด้วยกัน จำแนกโรคโคมะเชื้อเทศด้วย MLP ผลการทดลองพบว่าการจำแนกประเภทด้วยคุณสมบัติที่มีการรวมกันของคุณสมบัติจาก ResNet101 และ ResNet152 นั้นให้ค่าความแม่นยำสูงสุดคือร้อยละ 98.90

Rajasekaran Thangaraj และคณะ [35] นำเสนอแบบจำลองเพื่อระบุโรคของโคมะเชื้อเทศโดยใช้โครงข่ายประสาทเทียมเชิงลึก Modified-Xception Model ใช้การถ่ายโอนการเรียนรู้สำหรับถ่ายโอนน้ำหลักและพารามิเตอร์การเรียนรู้ มีการแบ่งการทำงานของโมเดลออกเป็นสองส่วนคือการสกัด

คุณลักษณะและการจำแนกโรค ในขั้นตอนการสกัดคุณสมบัติประกอบด้วยชั้นคอนโวลูชัน ชั้นการรวมผลลัพธ์ใช้ฟังก์ชันกระตุ้น Rectified linear unit (ReLU) ส่วนขั้นตอนการจำแนกโรคใช้ Global average pooling (GAP) แทนชั้นการเชื่อมต่ออย่างสมบูรณ์ในเครือข่ายแบบเดิม GAP จะคำนวณค่าเฉลี่ยของแผนผังคุณสมบัติ ชั้นผลลัพธ์ใช้ฟังก์ชันซอฟต์แวร์ นอกจากนี้ยังมีการใช้เทคนิคการปรับจูน (Fine-tuning) เพื่อปรับปรุงค่าน้ำหนักที่ได้จากการฝึกแบบจำลองด้วยชุดข้อมูลใบมะเขือเทศ มีการเปรียบเทียบการใช้เครื่องมือเพิ่มประสิทธิภาพ (Optimizer) เช่น Adaptive Moment Estimation (Adam), Stochastic Gradient Descent (SGD) และ Root-Mean-Square Propagation (RMSprop) โดยทดลองกับชุดข้อมูลใบมะเขือเทศจาก PlantVillage Dataset ผลจากการทดลองพบว่าการใช้ Adam และ RMSprop ให้ค่าความแม่นยำที่ดีกว่า SGD โดย Adam ให้ค่าความแม่นยำสูงที่สุดคือร้อยละ 99.55 และตามด้วย ร้อยละ RMSprop 99.01 และ SGD ร้อยละ 81.77 ตามลำดับ

Poornima Singh Thakur และคณะ [36] นำเสนอแบบจำลองสำหรับการระบุโรคพืชด้วยโครงข่าย Convolutional Neural Network ชื่อ VGG-ICNN แบบจำลองที่นำเสนอเป็นการทำงานร่วมกันของ VGG-16 และ GoogleNet InceptionV7 Block โดยมีวัตถุประสงค์เพื่อลดขนาดของโครงข่ายให้เล็กลงแต่ยังคงมีประสิทธิภาพในการจำแนกข้อมูลที่สูง แบบจำลองที่นำเสนอประกอบด้วยชั้นคอนโวลูชันขนาด 64 จำนวนสองชั้นตามด้วยชั้นรวมผลลัพธ์ด้วยค่ามากที่สุด จากนั้นตามด้วยคอนโวลูชันขนาด 128 จำนวนสองชั้นและตามด้วยชั้นรวมผลลัพธ์ด้วยค่ามากที่สุดเช่นเดียวกับแบบจำลองพื้นฐานของ VGG-16 จากนั้นมีการเพิ่ม InceptionV7 Block จำนวนสามบล็อก ต่อท้ายชั้นรวมผลลัพธ์ด้วยค่ามากที่สุดของ VGG16 ใน InceptionV7 Block ใช้ขนาด 1024 ตามด้วยชั้น Global Average Pooling (GAP) จากนั้นตามด้วยชั้น การเชื่อมต่อสมบูรณ์และใช้ฟังก์ชันซอฟต์แวร์สำหรับการจำแนกข้อมูล แบบจำลองที่นำเสนอสามารถลดจำนวนพารามิเตอร์เหลือเพียงประมาณ 6 ล้านพารามิเตอร์ ซึ่งเทียบได้กับโครงข่ายประสาทเทียมน้ำหนักเบา (Lightweight CNN) ทดสอบกับชุดข้อมูลจำนวนห้าชุดข้อมูล ได้แก่ PlantVillage Dataset, Embrapa Dataset, Apple Dataset, Maize Dataset และ Rice Dataset ผลการทดลองพบว่าชุดข้อมูล PlantVillage Dataset ให้ค่าความแม่นยำสูงที่สุดเป็นร้อยละ 99.16 รองลงมาคือ Rice Dataset, Apple Dataset, Embrapa Dataset และ Maize Dataset มีค่าความแม่นยำร้อยละ 96.67, 94.24, 93.66 และ 91.36 ตามลำดับ

Adem Tuncer และคณะ [37] นำเสนอแบบจำลองสำหรับการตรวจหาและระบุโรคของใบพืชด้วยโครงข่ายประสาทเทียมเชิงลึกแบบไฮบริดที่มีการทำงานร่วมกันของสถาปัตยกรรม Inception และ

Depthwise Separable Convolutions มีวัตถุประสงค์เพื่อลดจำนวนพารามิเตอร์และต้นทุนด้านการคำนวณ แต่ยังคงมีประสิทธิภาพด้านการจำแนกข้อมูลที่สูง สถาปัตยกรรม Inception สามารถดึงคุณสมบัติของรูปภาพได้แบบขนานไปพร้อมกันและสามารถรวมคุณสมบัติเข้าด้วยกัน ส่วน Depthwise Separable Convolution เป็นการสกัดคุณสมบัติเพื่อทำการบิดเชิงพื้นที่อย่างอิสระของช่องรับสัญญาณของภาพที่รับเข้าและทำการรวมจุดเป็นผลลัพธ์จากชั้นคอนโวลูชัน แบบจำลองที่นำเสนอได้ทำการแทนที่ชั้นคอนโวลูชันที่มีตัวกรองขนาด  $1 \times 1$  ตามด้วย  $3 \times 3$  ชั้นคอนโวลูชันที่มีตัวกรองขนาด  $1 \times 1$  และ  $5 \times 5$  ด้วยชั้น Depthwise ขนาด  $3 \times 3$  และชั้น Pointwise ขนาด  $1 \times 1$  และ Depthwise ขนาด  $5 \times 5$  และชั้น Pointwise ขนาด  $1 \times 1$  ใน Inception Block โดยแบบจำลองที่นำเสนอประกอบด้วยสถาปัตยกรรม Modified Inception จำนวนสองบล็อก ชั้น Depthwise Separable Convolution จำนวนสี่ชั้น ชั้นรวมผลลัพธ์จำนวนสี่ชั้น และตามด้วยชั้นการเชื่อมต่อสมบูรณ์และใช้ฟังก์ชันซอฟต์แวร์ฝึกทดลองกับชุดข้อมูล PlantVillage จำนวน 30 คลาส ผลการทดลองพบว่าแบบจำลองที่นำเสนอมีค่าความแม่นยำในการจำแนกโรคของพืชเป็นร้อยละ 99.27 และสามารถลดจำนวนพารามิเตอร์ได้ร้อยละ 75 จากโมเดล CNN มาตรฐาน เช่นเดียวกับงานวิจัยของ Sk Mahmudul Hassan ใช้การแทนที่ชั้นคอนโวลูชันด้วย Depthwise Separable Convolution เพื่อช่วยลดจำนวนพารามิเตอร์และต้นทุนการคำนวณของโมเดล

Trong-Yen Lee และคณะ [38] นำเสนอแบบจำลองสำหรับการตรวจหาโรคใบมันฝรั่ง ประกอบด้วยชั้นคอนโวลูชันขนาด 64 จำนวนหนึ่งชั้น ตามด้วยชั้นคอนโวลูชันขนาด 128 จำนวนสองชั้น ถัดไปตามด้วยชั้น คอนโวลูชันขนาด 256 จำนวนสามชั้น ชั้นการรวมผลลัพธ์ และสุดท้ายตามด้วยชั้นการเชื่อมต่อสมบูรณ์ที่ใช้ฟังก์ชันซอฟต์แวร์ฝึก ในขั้นตอนการทดลองเพื่อป้องกันการเกิดการเรียนรู้ของแบบจำลองที่มากเกินไป จึงได้ทำการเพิ่มจำนวนรอบในการทดลอง เพื่อรักษาค่าความแม่นยำของแบบจำลองที่นำเสนอให้ทดลองกับชุดข้อมูลใบมันฝรั่งจาก PlantVillage dataset และทำการเปรียบเทียบประสิทธิภาพของแบบจำลองที่นำเสนอกับ VGG16 และ VGG19 ผลการทดลองพบว่าวิธีการที่นำเสนอสามารถให้ค่าความแม่นยำสูงที่สุดเป็นร้อยละ 99.53 รองลงมาคือ VGG16 และ VGG19 มีค่าความแม่นยำเป็นร้อยละ 98.15 และ 48.55 ตามลำดับ และวิธีการที่นำเสนอสามารถลดจำนวนการใช้พารามิเตอร์ได้ประมาณร้อยละ 99.39

Peng Jiang และคณะ [39] นำเสนอแบบจำลองสำหรับการตรวจหาโรคของใบแอปเปิ้ลแบบเรียลไทม์ที่ชื่อ INAR-SSD พัฒนาแบบจำลองจาก VGGNet ร่วมกับ Inception ในขั้นตอนของการสกัด

คุณสมบัติของรูปภาพ จากนั้นใช้ Rainbow Concatenation Method เพื่อปรับปรุงการรวมกันของแผนผังคุณสมบัติ Pooling และ Deconvolution ทำงานไปพร้อมกันเพื่อสร้างแผนที่คุณสมบัติระหว่างชั้น แบบจำลองที่นำเสนอมีการเพิ่ม Inception Modules จำนวนสองชั้นในเครือข่ายของ VGG16 เพื่อปรับปรุงความสามารถในการสกัดคุณสมบัติของรูปภาพแบบหลายขนาด จึงทำให้แบบจำลองที่นำเสนอสามารถตรวจหาโรคที่มีจุดขนาดต่างๆ บนใบพืชเดียวกันได้ โดยชั้นที่หนึ่งถึงชั้นที่เจ็ดยังคงใช้สถาปัตยกรรมเดิมของ VGG16 จากนั้นในชั้นที่แปดและเก้าจะถูกแทนที่ด้วย Inception จำนวนสองโมดูล ส่วนในชั้นที่สิบถึงชั้นที่สิบสามก็ยังคงใช้สถาปัตยกรรมเดิมของ VGG16 ส่วนชั้นการเชื่อมต่อสมบูรณ์ถูกแทนที่ด้วยชั้นคอนโวลูชันขนาด  $1 \times 1$  จำนวนสามชั้นและตามด้วยฟังก์ชันซอฟต์แวร์แมก ทดลองกับชุดข้อมูลใบแอปเปิ้ลที่ได้จากการถ่ายภาพจำนวน 26,377 รูปภาพ ผลการทดลองพบว่าวิธีการที่นำเสนอให้ค่าความแม่นยำในการจำแนกโรคใบแอปเปิ้ลคือร้อยละ 97.14 และให้ค่าความแม่นยำสูงที่สุดเมื่อเปรียบเทียบกับแบบจำลองอื่น เช่น AlexNet, GoogLeNet, InceptionV3, ResNet101, ResNet50, ResNet34, ResNet18 และ VGG-16 ดังนั้นแสดงให้เห็นว่าโมเดล INAR-SSD มีประสิทธิภาพการตรวจจ็บร้อยละ 78.80 mAP และมีความเร็วในการตรวจจ็บ 23.13 FPS

Reva Nagi และคณะ [40] นำเสนอแบบจำลองสำหรับการระบุโรคของใบองุ่นโดยใช้โครงข่ายประสาทเทียมน้ำหนักเบาที่ต้องการลดจำนวนพารามิเตอร์สำหรับการคำนวณแต่ยังคงมีค่าความแม่นยำที่ดีสำหรับการจำแนกข้อมูลโรคใบองุ่น แบบจำลองที่นำเสนอพัฒนาบนพื้นฐานของ VGG16 ซึ่งเดิมมีจำนวนชั้นทั้งหมด 16 ชั้น ดังนั้นจึงปรับลดจำนวนชั้นของ VGG16 เหลือเพียง 6 ชั้นและปรับขนาดตัวกรองของชั้นคอนโวลูชัน โดยแบบจำลองที่นำเสนอประกอบด้วยชั้นนำเข้รับข้อมูลรูปภาพเข้าขนาด  $128 \times 128$  ชั้น คอนโวลูชันขนาด 128 ตามด้วยชั้นรวมผลลัพธ์มากที่สุดขนาด  $2 \times 2$  ชั้นคอนโวลูชันขนาด 64 ตามด้วยชั้นรวมผลลัพธ์มากที่สุดขนาด  $2 \times 2$  จากนั้นตามด้วยชั้นการเชื่อมต่อสมบูรณ์และฟังก์ชันซอฟต์แวร์แมก การทดลองใช้ Adam Optimizer ทดลองกับชุดข้อมูลใบองุ่นจาก PlantVillage Dataset จำนวน 3,423 รูปภาพ ผลการทดลองพบว่าวิธีการที่นำเสนอให้ค่าความแม่นยำในการจำแนกโรคของใบองุ่นสูงที่สุดคือ 98.4% เมื่อเปรียบเทียบกับ MobileNet, VGG16 และ AlexNet ให้ค่าความแม่นยำร้อยละ 98.10, 96.60 และ 95.70 ตามลำดับ นอกจากนี้วิธีที่นำเสนอสามารถลดจำนวนพารามิเตอร์เหลือเพียงประมาณสองล้านพารามิเตอร์

Prabhjot Kaur และคณะ [41] นำเสนอแบบจำลอง Modified InceptionResNet-V2 (MIR-V2) เพื่อระบุโรคของใบมะเขือเทศ โดยปรับปรุงมาจากพื้นฐานของ InceptionResNet-V2 (IR-V2) เป็น

การทำงานร่วมกันของ InceptionV1 และ ResNetV2 และใช้เทคนิคการถ่ายโอนการเรียนรู้จาก InceptionResNet-V2 เพื่อเพิ่มค่าความแม่นยำในการจำแนกโรคของใบมะเขือเทศ ปรับปรุงแบบจำลอง InceptionResNet-V2 ที่แตกต่างกันสี่แบบโดยการปรับเปลี่ยนชั้นการรวมผลลัพธ์ด้วยค่ามากที่สุดภายในสถาปัตยกรรมของ InceptionResNet-V2 ดังนี้ 1) 3-Max pool layer InceptionResNet-V2 (3MPL-IR-V2) 2) 3 -Max pool layer with skip connection (3 MPL-SC) 3) 2 -Max pool-layer InceptionResNet-V2 (2MPL-IR-V2) และ 4) 2-Max pool layer with skip connection (2MPL-SC) จากนั้นทำการทดลองกับชุดข้อมูลใบมะเขือเทศ ผลการทดลองพบว่าการปรับเปลี่ยนชั้นการรวมผลลัพธ์ด้วยค่ามากที่สุดของ InceptionResNet-V2 ที่ให้ค่าความแม่นยำสูงที่สุดคือแบบ 2MPL-SC ให้ค่าความแม่นยำเท่ากับร้อยละ 98.92 เมื่อเปรียบเทียบกับ การปรับปรุงอีกสามแบบที่เหลือ

Mariam Moussafir และคณะ [42] นำเสนอแบบจำลองเพื่อเพิ่มประสิทธิภาพของการจำแนกโรคของใบมะเขือเทศ โดยใช้แบบจำลอง CNN ร่วมกับการถ่วงค่าน้ำหนักด้วย Genetic Algorithm ในขั้นตอนแรกทำการเปรียบเทียบผลการจำแนกโรคของใบมะเขือเทศจากชุดข้อมูล PlantVillage จำนวน 10 คลาส โดยทดลองกับแบบจำลองของ CNN เช่น VGG16, ResNet50, EfcientNetB0, EfcientNetB1, EfcientNetB2, EfcientNetB3 และ EfcientNetB4 โดยใช้เทคนิคการถ่ายโอนการเรียนรู้และการปรับจูนเพื่อคัดเลือกแบบจำลองที่ให้ค่าความแม่นยำในการจำแนกโรคของใบมะเขือเทศสูงที่สุดจำนวนสองแบบ จากนั้นนำสองโมเดลที่ให้ค่าความแม่นยำสูงที่สุดไปปรับใช้งานร่วมกับ Genetic Algorithm โดย Genetic Algorithm จะทำการคัดเลือกค่าน้ำหนักที่เกิดขึ้นระหว่างการเรียนรู้ด้วย CNN เพื่อสร้างเป็นเมทริกซ์น้ำหนักใหม่ จากนั้นจะพิจารณาจากค่าเฉลี่ยถ่วงน้ำหนัก ผลการทดลองการจำแนกโรคของใบมะเขือเทศด้วยแบบจำลอง CNN พบว่าแบบจำลองที่ให้ค่าความแม่นยำสูงที่สุดคือ ResNet50 และ EfcientNetB0 มีค่าความแม่นยำร้อยละ 96.60 และ 95.60 จากนั้นเมื่อมีการใช้ Genetic Algorithm เพื่อสร้างเป็นเมทริกซ์น้ำหนักใหม่จากโมเดล ResNet50 และ EfcientNetB0 ผลค่าความแม่นยำของการจำแนกโรคใบมะเขือเทศเพิ่มขึ้นเป็นร้อยละ 98.10

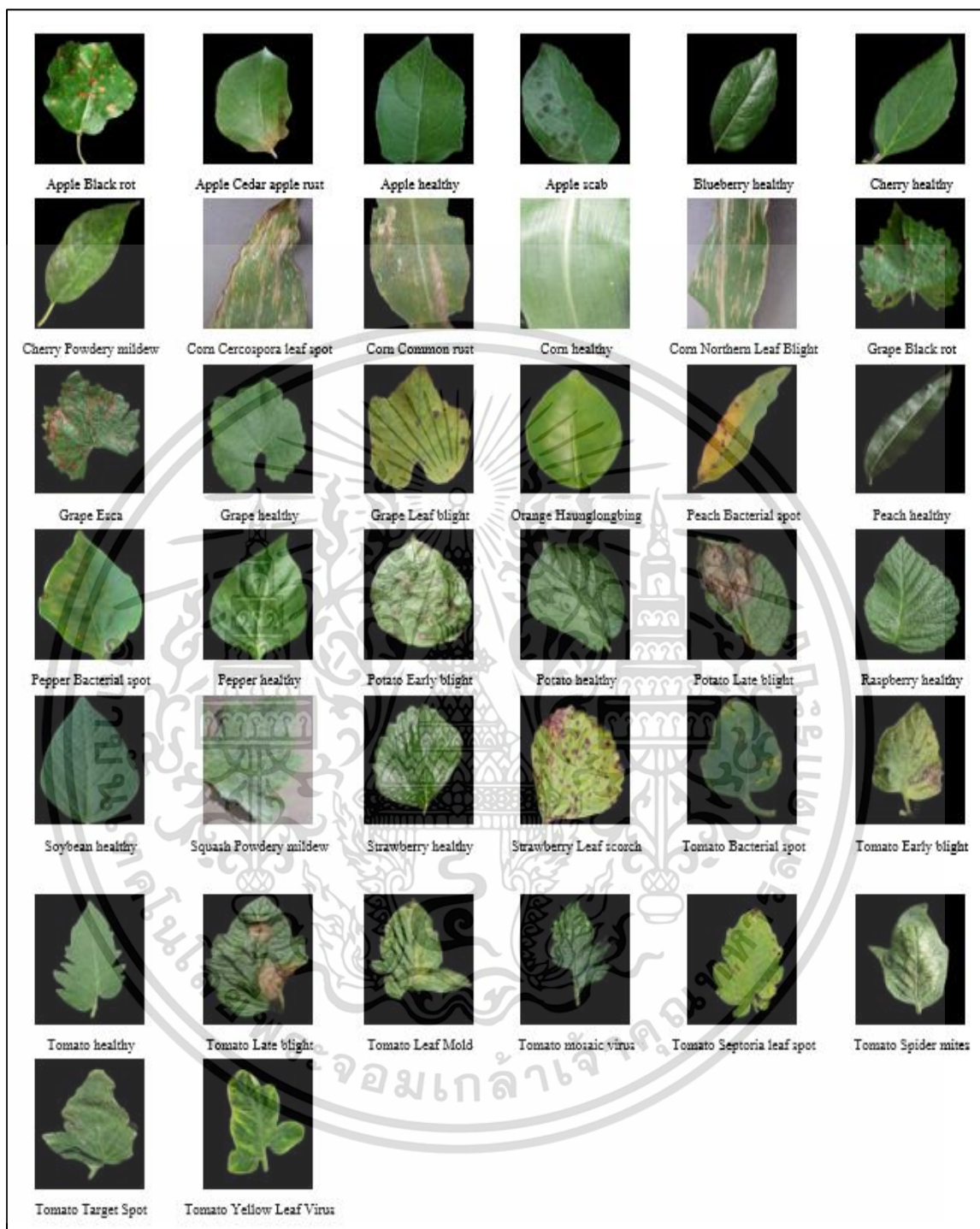
## บทที่ 3

### วิธีดำเนินการวิจัย

ตามข้อสมมติฐานการวิจัยในบทที่ 1 มีข้อสมมติฐานการวิจัยดังนี้ 1) การจำแนกใบพืชโดยใช้การกำหนดจุดเด่นบนขอบใบด้วย Scale-Invariant Feature Transform (SIFT) สามารถเพิ่มค่าความแม่นยำในการจำแนกใบพืชได้หรือไม่ 2) การปรับปรุงสถาปัตยกรรมของ VGG16 ร่วมกับ InceptionV3 with Parameter Reduction Layer สามารถเพิ่มค่าความแม่นยำในการจำแนกโรคของใบพืชได้หรือไม่ และ 3) การปรับปรุงสถาปัตยกรรมของ VGG16 ร่วมกับ InceptionV3 with Parameter Reduction Layer สามารถลดจำนวนพารามิเตอร์ของสถาปัตยกรรมและเวลาในการประมวลผลได้หรือไม่ บทนี้จึงนำเสนอขั้นตอนการดำเนินการวิจัยโดยมีขั้นตอนการทำงานดังนี้

#### 3.1 การเตรียมชุดข้อมูลใบพืช (Preparation of Plant Leaf Dataset)

ชุดข้อมูลที่ใช้สำหรับการทดลองในงานวิจัยคือชุดข้อมูลใบพืชจาก PlantVillage Dataset จาก [www.kaggle.com](http://www.kaggle.com) ซึ่งเป็นชุดข้อมูลมาตรฐานที่นิยมใช้สำหรับการทำงานด้านการประมวลผลภาพและการจำแนกข้อมูลภาพ ประกอบด้วยภาพใบพืชที่เป็นโรคและไม่เป็นโรค จำนวน 38 คลาส เป็นพืชที่แตกต่างกัน 14 ชนิด ได้แก่ ใบแอปเปิ้ล ใบบลูเบอร์รี่ ใบเชอร์รี่ ใบข้าวโพด ใบองุ่น ใบส้ม ใบพีช ใบพริกหวาน ใบมันฝรั่ง ใบราสเบอร์รี่ ใบถั่ว ใบฟัก ใบสตรอเบอร์รี่ และใบมะเขือเทศ โดยมีรูปภาพทั้งหมด 54,304 ภาพ ตัวอย่างชุดข้อมูลดังรูปที่ 1 และมีรายละเอียดจำนวนรูปภาพของแต่ละคลาสดังตารางที่ 1



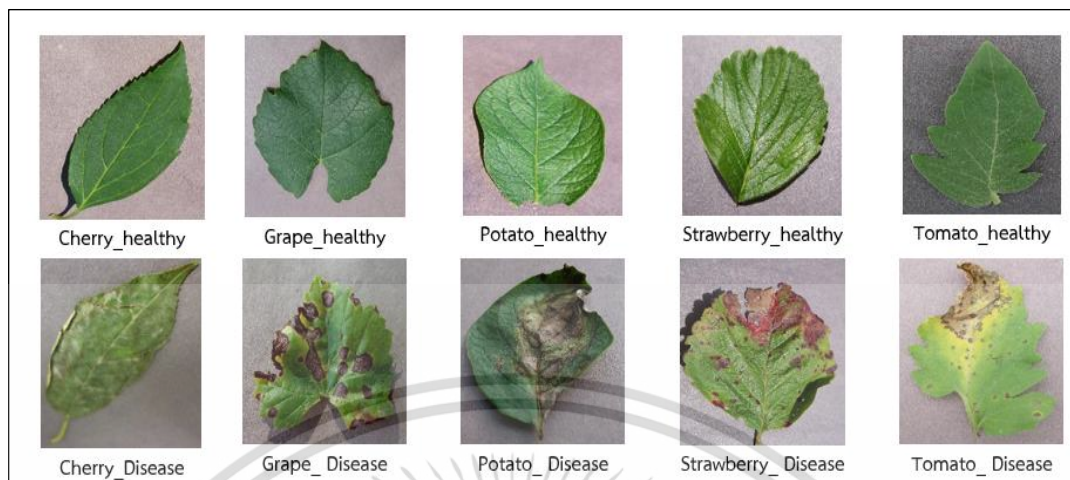
รูปที่ 3.1 ตัวอย่างใบพืชที่ใช้สำหรับการทดลองจากชุดข้อมูล PlantVillage

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

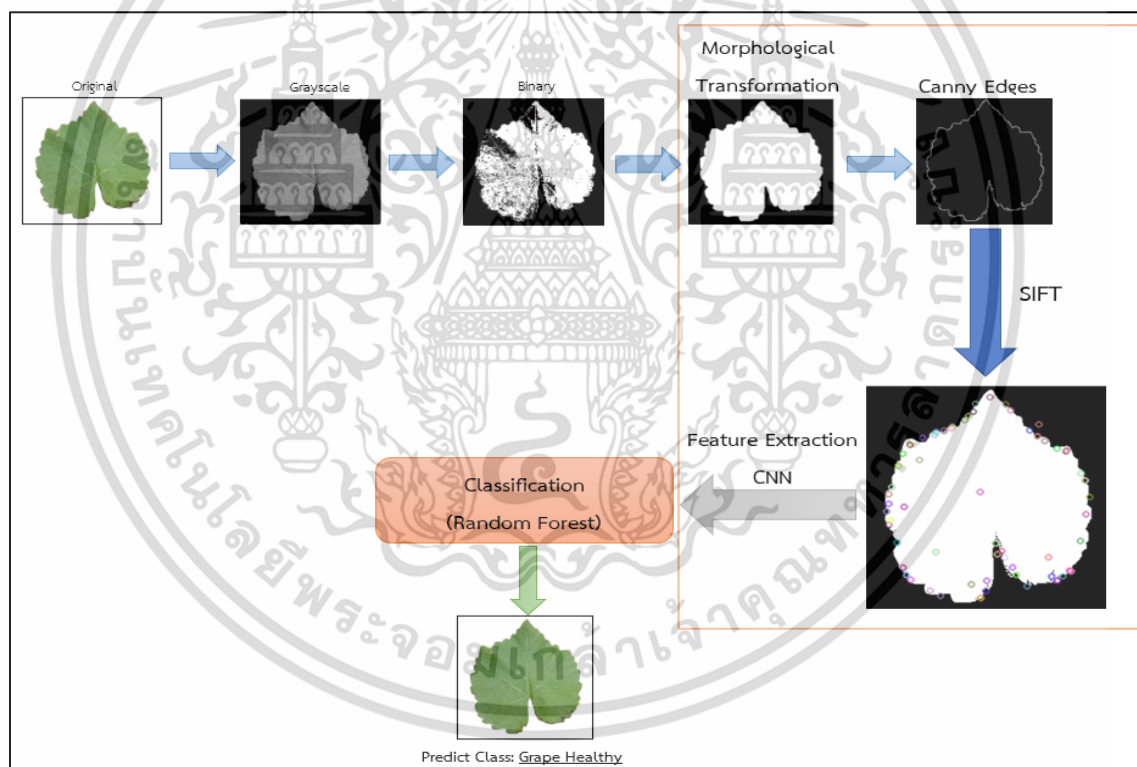
ตารางที่ 3.1 รายละเอียดชุดข้อมูลจาก PlantVillage Dataset

คลาส		จำนวนรูป	คลาส		จำนวนรูป
1.	Apple scab	630	20.	Pepper healthy	1,478
2.	Apple Black rot	621	21.	Potato Early blight	1,000
3.	Apple Cedar rust	275	22.	Potato healthy	152
4.	Apple healthy	1,645	23.	Potato Late blight	1,000
5.	Blueberry healthy	1,502	24.	Raspberry healthy	371
6.	Cherry healthy	853	25.	Soybean healthy	5,090
7.	Cherry Powdery mildew	1,052	26.	Squash Powdery mildew	1,835
8.	Corn Cercospora leaf spot	513	27.	Strawberry healthy	456
9.	Corn Common rust	1,192	28.	Strawberry Leaf scorch	1,109
10.	Corn healthy	1,162	29.	Tomato Bacterial spot	2,127
11.	Corn Northern Leaf Blight	985	30.	Tomato Early blight	1,000
12.	Grape Black rot	1,180	31.	Tomato healthy	1,591
13.	Grape Esca Black Measles	1,383	32.	Tomato Late blight	1,909
14.	Grape healthy	423	33.	Tomato Leaf Mold	952
15.	Grape Leaf blight	1,076	34.	Tomato Septoria leaf spot	1,771
16.	Orange Haunglongbing	5,507	35.	Tomato Spider mites	1,676
17.	Peach Bacterial spot	2,297	36.	Tomato Target Spot	1,404
18.	Peach healthy	360	37.	Tomato mosaic virus	373
19.	Pepper Bacterial spot	997	38.	Tomato Yellow leaf Virus	5,357
<b>Total Number of Images</b>				<b>54,304</b>	

ตามข้อสมมติฐานการวิจัยที่หนึ่ง การจำแนกใบพืชโดยใช้การกำหนดจุดเด่นบนขอบใบด้วย Scale-Invariant Feature Transform (SIFT) เลือกใช้รูปภาพใบพืชที่สุขภาพดีจำนวน 5 คลาส และใบพืชที่เป็นโรคจำนวน 5 คลาส รวม 10 คลาส ประกอบด้วย ใบเชอร์รี่ ใบองุ่น ใบมันฝรั่ง ใบสตรอเบอร์รี่ และใบมะเขือเทศ เนื่องจากขั้นตอนนี้เป็น การตรวจจับเส้นขอบของใบพืชและกำหนดจุดเด่นบนขอบใบพืช ดังนั้นจึงเลือกคลาสของใบพืชที่มีลักษณะรอยหยักกรอบใบที่คล้ายๆ กัน และในส่วนของใบพืชที่เป็นโรคใช้วิธีการเลือกคลาสที่เป็นโรคแล้วไม่ทำให้ลักษณะรูปร่างของใบเปลี่ยนแปลงไป รวมรูปภาพทั้งหมด 4,015 รูป เป็นรูปภาพที่มีสีแบบอาร์จีบี ขนาด 256 x 256 พิกเซล ดังแสดงตัวอย่างในรูปที่ 3.2 และมีกรอบการดำเนินงานดังรูปที่ 3.3



รูปที่ 3.2 ตัวอย่างใบพืชที่สำหรับการทดลองตามข้อสมมติฐานที่หนึ่ง



รูปที่ 3.3 กรอบดำเนินงานของการวิจัยตามข้อสมมติฐานที่หนึ่ง

จากรูปที่ 3.3 แสดงขั้นตอนการดำเนินการวิจัยตามข้อสมมติฐานของการวิจัยข้อที่หนึ่ง ประกอบด้วยขั้นตอนการประมวลผลรูปภาพ การสกัดคุณสมบัตินิรูปภาพและจำแนกข้อมูลใบพืช

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.2 การประมวลผลรูปภาพใบพืช (Plant Leaf Image Processing)

ขั้นตอนการประมวลผลรูปภาพใบพืชเป็นขั้นตอนแรกของการดำเนินงานเพื่อเพิ่มประสิทธิภาพให้กับรูปภาพก่อนนำไปใช้สำหรับขั้นตอนต่อไป โดยมีขั้นตอนการประมวลผลรูปภาพใบพืชดังนี้

### 3.2.1 การแปลงค่าสี (Color Transform)

สำหรับข้อมูลรูปภาพใบพืชจากชุดข้อมูล PlantVillage Dataset เป็นภาพที่มีแบบจำลองสีอาร์จีบี (RGB) งานวิจัยนี้จึงมีกระบวนการประมวลผลรูปภาพ ขั้นตอนแรกทำการลบพื้นหลังของรูปภาพ จากนั้นทำการแปลงภาพสีอาร์จีบีเป็นภาพระดับสีเทา และทำการแปลงภาพระดับสีเทาเป็นภาพแบบไบนารี เนื่องจากในขั้นตอนการตรวจจับขอบภาพจะต้องใช้ภาพแบบไบนารี โดยใช้ไลบรารีพื้นฐานของ OpenCV และมีการปรับค่าเกณฑ์ต่ำสุดและเกณฑ์สูงสุดของ Threshold ให้เหมาะสมกับรูปภาพดังนี้

```
cv2.threshold(input_image, 127, 255, cv2.THRESH_BINARY)
```

และมีการใช้ค่า Otsu's Threshold สำหรับการกำหนดค่าเกณฑ์ Threshold แบบอัตโนมัติเพื่อทำให้ภาพเป็นไบนารีดังนี้

```
cv2.threshold(input_image, 127, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)
```

### 3.2.2 การเปลี่ยนแปลงทางสัณฐานวิทยา (Morphological Transformations)

จากการดำเนินการในขั้นตอนการแปลงค่าสีของรูปภาพใบพืชจากภาพสีอาร์จีบีเป็นภาพระดับสีเทา และภาพไบนารี ในส่วนของภาพใบพืชที่เป็นโรคที่มีแสงและสีของใบพืชที่แตกต่างกัน ดังนั้นเมื่อแปลงค่าสีแล้วยังคงมีสัญญาณรบกวนของภาพ ดังนั้นในขั้นตอนการเปลี่ยนแปลงลักษณะทางสัณฐานวิทยาของภาพใบพืชจึงใช้ตัวดำเนินการ Opening และ Closing เพื่อลบสัญญาณรบกวนของภาพเพื่อให้มีเฉพาะเส้นขอบที่เป็นรูปร่างของใบพืชเท่านั้น ซึ่งในการวิจัยนี้ได้มีการกำหนดค่าองค์ประกอบโครงสร้าง (Structuring Element) ของตัวดำเนินการทั้งสอง ดังนี้

Opening

**Structuring element:**

```
input_image = BinaryImage
```

```
kernel = 3x3
```

```
cv.morphologyEx = cv.MORPH_OPEN
```

```
output_image = OpeningImage
```

Closing

**Structuring element:**

```
input_image = OpeningImage
```

```
kernel = 3x3
```

```
cv.morphologyEx = cv.MORPH_CLOSE
```

```
output_image = ClosingImage
```

การกำหนดค่าองค์ประกอบโครงสร้างของตัวดำเนินการ Opening และ Closing ดังกล่าวเพื่อขยายเส้นขอบของไบพีชให้เชื่อมต่อกันและลบบางพิกเซลของขอบไบพีชที่มีผลมาจากแสงและเงาของไบพีช และเพื่อปิดรูรั่วที่อยู่บนรูปภาพให้มีลักษณะเป็นพื้นที่ยื่นออกมาที่ด้านหน้าทีสมบูรณ์ โดยลบสัญญาณรบกวนของภาพที่อยู่บนรูปภาพไบพีชที่ได้มาหลังจากการแปลงภาพสีเทาเป็นไบนารี ทำให้ได้รูปภาพแบบไบนารีที่มีเฉพาะเค้าโครงรูปร่างของไบพีชเพื่อจะได้นำไปตรวจจับเส้นขอบของไบพีช

### 3.2.3 การตรวจจับขอบภาพ (Edge Detection)

หลังจากการเปลี่ยนแปลงลักษณะทางสีของรูปภาพเพื่อลบสัญญาณรบกวนของรูปภาพ ทำให้ได้รูปไบพีชแบบไบนารี จากนั้นจึงทำการตรวจจับเส้นขอบของไบพีชด้วยวิธีของแคนนี่ (Canny Edge Detection) โดยใช้งานผ่านไลบรารีพื้นฐานของ OpenCV

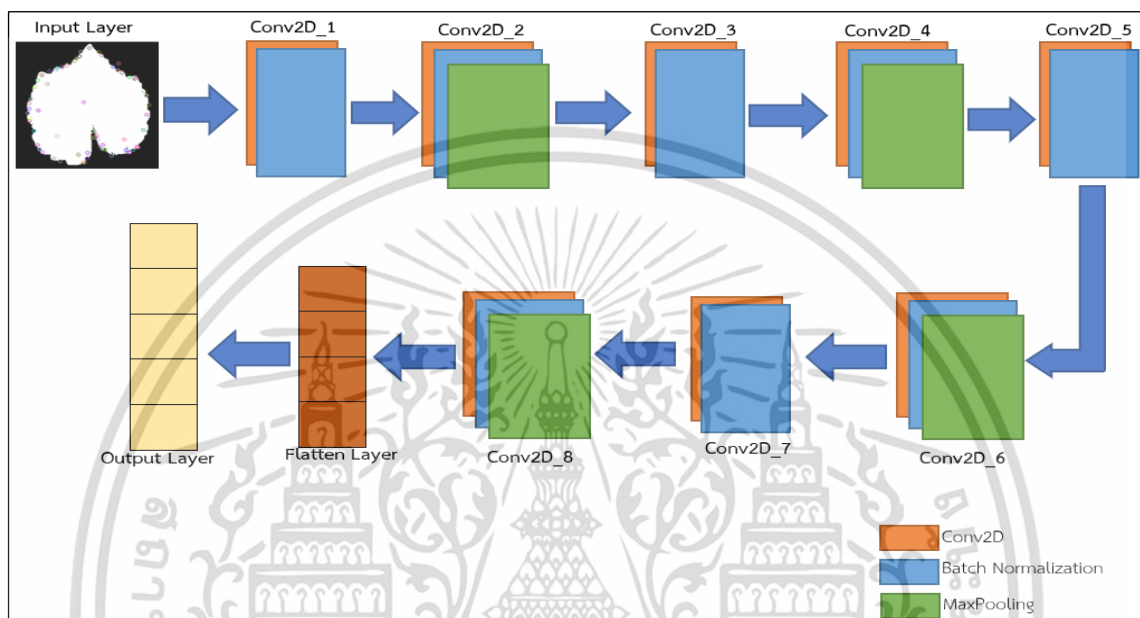
### 3.2.4 การหาจุดเด่นบนภาพด้วย Scale-Invariant Feature Transform (SIFT)

รูปภาพไบพีชที่ผ่านขั้นตอนการประมวลผลรูปภาพด้วยการแปลงค่าสี การเปลี่ยนแปลงลักษณะทางสีของรูปภาพและตรวจจับขอบของรูปภาพแล้ว ขั้นตอนต่อไปคือการกำหนดจุดสำคัญบนขอบไบพีชด้วยอัลกอริทึม Scale-Invariant Feature Transform (SIFT) เพื่อทำการเปรียบเทียบความเหมือนกันของรูปภาพ มีการใช้งานผ่านไลบรารีพื้นฐานของ OpenCV โดย SIFT จะกำหนดจุดสำคัญลงบนเส้นขอบของไบพีชตามลักษณะรูปร่างของไบพีช

## 3.3 การสกัดคุณสมบัติและจำแนกข้อมูลไบพีช

ขั้นตอนการสกัดคุณสมบัติและจำแนกข้อมูลไบพีชแบ่งการทำงานออกเป็น 2 ขั้นตอน คือ การสกัดคุณสมบัติรูปภาพและการจำแนกข้อมูลไบพีช ขั้นตอนการสกัดคุณสมบัติรูปภาพจะใช้รูปภาพไบพีชที่ผ่านขั้นตอนการประมวลผลรูปภาพและกำหนดจุดสำคัญบนเส้นขอบไบพีชแล้ว โดยใช้แบบจำลองโครงข่ายประสาทเทียมที่เป็นแบบจำลองตามลำดับ (Sequential Model) ดึงคุณสมบัติของรูปภาพทั้งหมดเพื่อเป็นชุดคุณสมบัติสำหรับการจำแนกชนิดของไบพีชต่อไป การใช้โครงข่ายประสาทเทียมสกัดคุณสมบัตินั้น จะใช้งานผ่านฟังก์ชัน Sequential Model ของไลบรารี Keras ซึ่งประกอบด้วย ขั้นตอนนำเข้าข้อมูลรูปภาพ

ชั้นคอนโวลูชันจำนวน 8 ชั้น หลังชั้นคอนโวลูชันแต่ละชั้นจะตามด้วย BatchNormalization เพื่อช่วยปรับชุดคุณสมบัติให้เป็นมาตรฐาน ชั้นการรวมผลลัพธ์ด้วยค่ามากที่สุดจำนวน 4 ชั้น ชั้นการเชื่อมต่อสมบูรณ์ และชั้นผลลัพธ์ แสดงดังรูปที่ 3.4



รูปที่ 3.4 สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบ Sequential Model

จากรูปที่ 3.4 การเพิ่มจำนวนชั้นคอนโวลูชันในชั้นซ่อนให้มีจำนวนมากขึ้นสามารถช่วยให้แบบจำลองสามารถสกัดคุณสมบัติของรูปภาพได้ครบถ้วนและแม่นยำมากขึ้น นอกจากนี้การเพิ่ม BatchNormalization ทำให้สามารถเพิ่มความเร็วในเรียนรู้ข้อมูลของแบบจำลองและเพิ่มความแม่นยำในการสกัดคุณลักษณะได้ และขั้นตอนต่อไปคือการจำแนกข้อมูลใบพืช หลังจากได้ชุดคุณสมบัติของรูปภาพขั้นตอนต่อไปคือการจำแนกข้อมูลใบพืชด้วยแบบจำลองป่าสุ่ม (Random Forest)

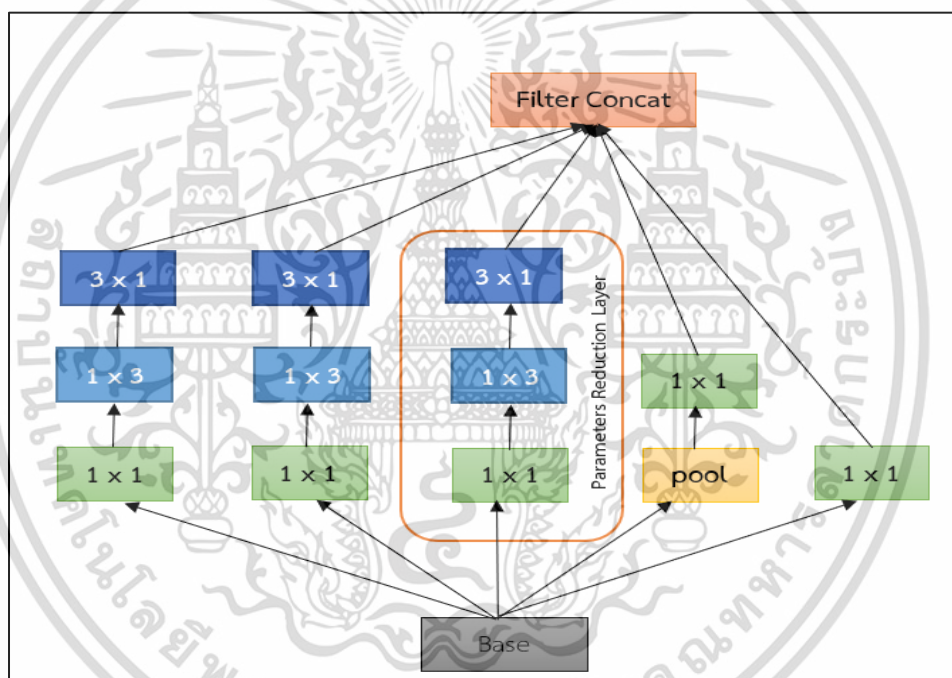
### 3.4 สถาปัตยกรรมโครงข่ายประสาทเทียม VGG10 – InceptionV3 with Parameter Reduction Layer

จากข้อสมมติฐานการวิจัยข้อที่สองคือการปรับปรุงสถาปัตยกรรมของ VGG16 ร่วมกับ InceptionV3 with Parameter Reduction Layer สามารถเพิ่มความแม่นยำในการจำแนกโรคของใบพืชได้หรือไม่ และข้อที่สามคือการปรับปรุงสถาปัตยกรรมของ VGG16 ร่วมกับ InceptionV3 with Parameter Reduction Layer สามารถลดจำนวนพารามิเตอร์ของสถาปัตยกรรมและเวลาในการ

ประมวลผลได้หรือไม่ ดังนั้นในส่วนนี้จึงอธิบายเกี่ยวกับการปรับปรุงสถาปัตยกรรมของโครงข่ายประสาทเทียม VGG10 – InceptionV3 with Parameter Reduction Layer โดยมีขั้นตอนการดำเนินงาน ดังนี้

#### 3.4.1 สถาปัตยกรรมโครงข่ายประสาทเทียม InceptionV3 with Parameter Reduction Layer

จากข้อดีของสถาปัตยกรรม InceptionV3 คือการกระจายการสกัดคุณสมบัติของรูปภาพขนานไปพร้อมๆ กัน ทำให้ได้แผนที่คุณสมบัติที่มีขนาดแตกต่างกันและสามารถลดจำนวนพารามิเตอร์และเวลาในการประมวลผลลงได้ ดังนั้นในงานนี้จึงได้ปรับปรุงสถาปัตยกรรมของ InceptionV3 with Parameter Reduction Layer โดยการเพิ่มความกว้างของเครือข่ายโดยให้มีจำนวนชั้นคอนโวลูชันเพิ่มขึ้นจาก 3 ชั้น เป็น 4 ชั้น แสดงรายละเอียดดังรูปที่ 3.5



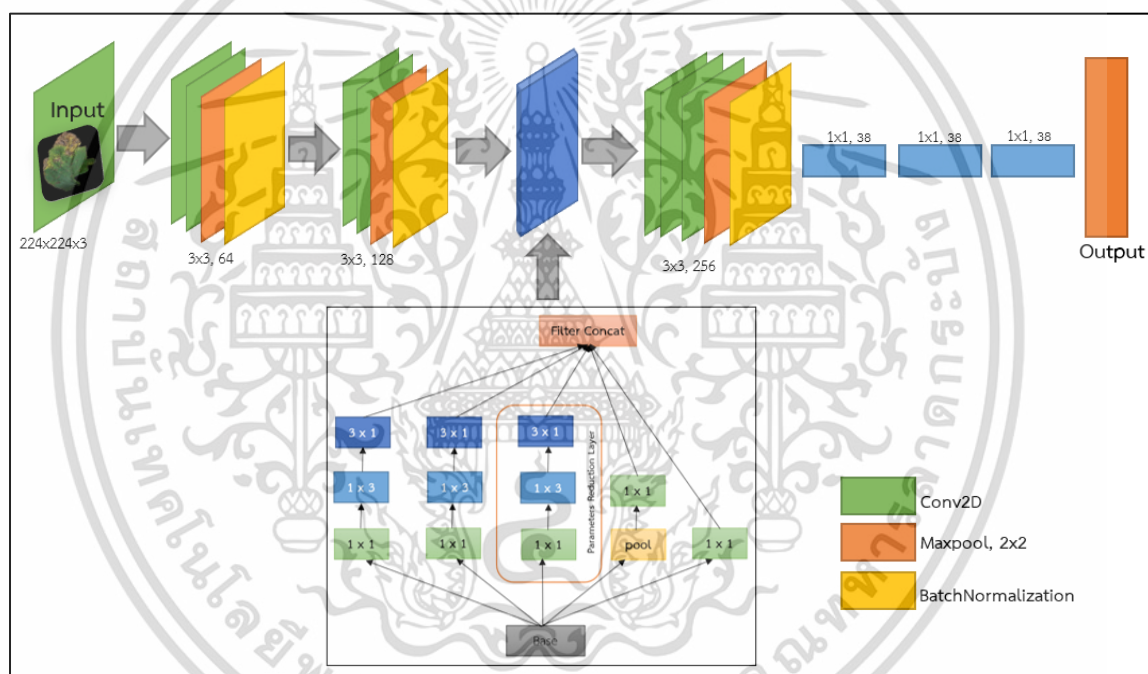
รูปที่ 3.5 สถาปัตยกรรมของ InceptionV3 with Parameter Reduction Layer

จากรูปที่ 3.5 สถาปัตยกรรมของ InceptionV3 with Parameter Reduction Layer ที่ปรับปรุงประกอบด้วยชั้นคอนโวลูชันขนาด  $1 \times 1$  ชั้นคอนโวลูชันขนาด  $1 \times 1$  ตามด้วย ชั้นคอนโวลูชันขนาด  $1 \times 3$  และชั้นคอนโวลูชันขนาด  $3 \times 1$  เหมือนกันทั้ง 3 ชั้น และชั้นการรวมผลลัพธ์ด้วยค่ามากที่สุดตามด้วยชั้นคอนโวลูชันขนาด  $1 \times 1$  สกัดคุณสมบัติแบบขนานไปพร้อมกันทั้ง 4 ชั้น และทำการลดจำนวนขนาดคุณสมบัติด้วยชั้นการรวมผลลัพธ์ด้วยค่ามากที่สุด ซึ่งการแทนที่ชั้นคอนโวลูชันขนาด  $3 \times 3$  ด้วยชั้นคอน

โวลูชันขนาด  $1 \times 3$  และตามด้วยชั้นคอนโวลูชันขนาด  $3 \times 1$  เป็นการแยกชั้นคอนโวลูชันให้เล็กลง จึงสามารถทำงานได้เร็วขึ้น ช่วยลดจำนวนพารามิเตอร์ของสถาปัตยกรรมและลดต้นทุนด้านการคำนวณลงได้

### 3.4.2 สถาปัตยกรรมโครงข่ายประสาทเทียม VGG10 – InceptionV3 with Parameter Reduction Layer

วิธีการที่นำเสนอพัฒนาโดยใช้ VGG16 เป็นแบบจำลองพื้นฐานร่วมกับการปรับปรุงสถาปัตยกรรม Inception V3 with Parameter Reduction Layer โดยมีวัตถุประสงค์เพื่อเพิ่มค่าความแม่นยำในการจำแนกโรคของใบพืช และลดต้นทุนด้านการคำนวณทั้งในส่วนของจำนวนพารามิเตอร์และเวลาที่ใช้ในการประมวลผล โดยแบบจำลองที่พัฒนา แสดงดังรูปที่ 3.6



รูปที่ 3.6 สถาปัตยกรรมโครงข่ายประสาทเทียม VGG10 – InceptionV3 with Parameter Reduction Layer

จากรูปที่ 3.6 แสดงแสดงสถาปัตยกรรมโครงข่ายประสาทเทียม VGG10 – InceptionV3 with Parameter Reduction Layer โดยทำการเพิ่มสถาปัตยกรรม InceptionV3 with Parameter Reduction Layer ที่ปรับปรุงแล้วร่วมกับสถาปัตยกรรมของโครงข่ายประสาทเทียม VGG16 โดยทำการเพิ่มระหว่างชั้นที่ 4 และ 5 ของ VGG16 จากนั้นทำการปรับลดจำนวนชั้นของ VGG16 จาก 16 ชั้น ลดลงทีละชั้นไปเรื่อยๆ จนสามารถให้ค่าความถูกต้องในการจำแนกโรคของใบพืชสูงที่สุด จึงเหลือจำนวนชั้น

เพียง 10 ชั้น แบบจำลองที่นำเสนอประกอบด้วยชั้นนำเข้าข้อมูลรูปภาพ ที่รับข้อมูลรูปภาพเป็นภาพสีอาร์จีบี ขนาด 224 x 224 พิกเซล จากนั้นผ่านชั้นคอนโวลูชัน ชั้นที่หนึ่งและชั้นที่สองใช้ช่องสัญญาณขนาด 64 ถัดไปชั้นที่สามและชั้นที่สี่ใช้ช่องสัญญาณขนาด 128 เช่นเดียวกับแบบจำลองพื้นฐานของ VGG16 ส่วนในชั้นถัดไปเป็นการเพิ่ม InceptionV3 with Parameter Reduction Layer จำนวนหนึ่งบล็อก จากนั้นชั้นที่ห้าถึงชั้นที่เจ็ดตามด้วยชั้นคอนโวลูชันที่มีช่องสัญญาณขนาด 256 โดยชั้นคอนโวลูชันแต่ละชั้นตามด้วยชั้นการรวมผลลัพธ์ด้วยค่ามากที่สุดขนาด 2 x 2 และ BatchNormalization ทุกชั้น โดยแบบจำลองที่นำเสนอมีการใช้ตัวกรองขนาด 3 x 3 ตลอดทั้งโครงข่ายเช่นเดียวกับแบบจำลองพื้นฐานของ VGG16 จากนั้นในชั้นที่แปดจนถึงชั้นที่สิบตามด้วยชั้นการเชื่อมต่อสมบูรณ์และชั้นผลลัพธ์ใช้ฟังก์ชันซอฟต์แวร์สำหรับการจำแนกโรคของใบพืช ที่มี 38 คลาส โดยมีการกำหนดค่าพารามิเตอร์ต่างๆ สำหรับการทดลองของแบบจำลองที่นำเสนอดัง ตารางที่ 2

ตารางที่ 3.2 พารามิเตอร์สำหรับการทดลองของแบบจำลองที่นำเสนอ

	ชื่อพารามิเตอร์	พารามิเตอร์
1.	Optimizer	Adam optimizer
2.	Learning rate	0.001
3.	Batch size	64
4.	Epochs	50
5.	Training data	70%
6.	Validation data	20%
7.	Testing data	10%

### 3.5 การประเมินผล

การประเมินผลของสถาปัตยกรรมโครงข่ายประสาทเทียม VGG10 – InceptionV3 with Parameter Reduction Layer ทดลองกับชุดข้อมูลใบพืชจาก PlantVillage Dataset ที่มีจำนวนรูปภาพทั้งหมด 54,304 รูป ประกอบด้วยใบพืชที่เป็นโรคและไม่เป็นโรคจำนวน 38 คลาส ขั้นตอนการทดลองได้ทำการแบ่งชุดข้อมูลออกเป็น 3 ส่วน คือชุดข้อมูลสำหรับการเรียนรู้ของแบบจำลอง (Training Set) 70% ชุดข้อมูลสำหรับการตรวจสอบ (Validation Set) 20% และชุดข้อมูลสำหรับทดสอบ (Testing Set) 10%

ทำการแบ่งชุดข้อมูลรูปภาพโดยใช้ไลบรารี split-folders ด้วยภาษาไพทอน (Python) การประเมินผลของแบบจำลองสามารถประเมินจากค่าต่างๆ ดังนี้

### 3.5.1 Confusion Matrix

Confusion Matrix เป็นแบบประเมินผลจากการจำแนกข้อมูลหรือผลจากการทำนายของแบบจำลองที่สร้างขึ้น โดยประเมินผลในรูปแบบของตารางที่ประกอบด้วยค่าที่ต้องการทำนาย ผลที่ได้จากการทำนาย และสัดส่วนความถูกต้องของการทำนาย แสดงดังตารางที่ 3

ตารางที่ 3.3 การประเมินผลด้วย Confusion Matrix

	Actually Positive	Actually Negative
Predicted Positive	True Positive (TP)	False Positive (FP)
Predicted Negative	False Negative (FN)	True Negative (TN)

จากตารางที่ 3 Confusion Matrix ประกอบด้วยค่าต่างๆ ดังนี้

*True Positive (TP)* คือ สิ่งที่แบบจำลองสามารถทำนายได้ถูกต้องตรงกับสิ่งที่เกิดขึ้นจริง ในกรณีทำนายว่าจริงและสิ่งที่เกิดขึ้นก็คือจริง

*True Negative (TN)* คือ สิ่งที่แบบจำลองสามารถทำนายตรงกับสิ่งที่เกิดขึ้น ในกรณีที่ทำนายว่าไม่จริงและสิ่งที่เกิดขึ้นก็คือไม่จริง

*False Positive (FP)* คือ สิ่งที่แบบจำลองทำนายออกมาไม่ตรงกับสิ่งที่เกิดขึ้น คือทำนายว่าจริง แต่สิ่งที่เกิดขึ้นคือไม่จริง

*False Negative (FN)* คือ สิ่งที่แบบจำลองทำนายออกมาไม่ตรงกับที่ที่เกิดขึ้นจริง คือทำนายว่าไม่จริง แต่สิ่งที่เกิดขึ้นคือจริง

### 3.5.2 ค่าความถูกต้อง (Accuracy)

ค่าความถูกต้อง (Accuracy) คือการประเมินประสิทธิภาพการทำงานของแบบจำลองที่สามารถจำแนกข้อมูลหรือทำนายข้อมูลได้ถูกต้องเป็นอัตราร้อยละหรือเปอร์เซ็นต์ คำนวณได้จาก

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3.1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.3 ค่าความแม่นยำ (Precision)

ค่าความแม่นยำ (Precision) เป็นการเปรียบเทียบผลการทำนายที่แบบจำลองทำนายได้ถูกต้องว่าจริง และก็เกิดขึ้นจริง (TP) กับการทำนายว่าจริงแต่สิ่งที่เกิดขึ้น คือ ไม่จริง (FP) คำนวณได้จาก

$$Precision = \frac{TP}{TP+FP} \quad (3.2)$$

### 3.5.4 ค่าเรียกคืน (Recall)

ค่าเรียกคืน (Recall) เป็นการเปรียบเทียบความถูกต้องของแบบจำลองจากการทำนายว่าจะป็นจริง เปรียบเทียบกับจำนวนครั้งของเหตุการณ์ทั้งทำนายและเกิดขึ้น ว่า เป็นจริง คำนวณได้จาก

$$Recall = \frac{TP}{TP+FN} \quad (3.3)$$

### 3.5.5 F-Measure

ค่า F-Measure หรือ ค่า F1-Score เป็นค่าเฉลี่ยแบบระหว่างค่าความแม่นยำ (Precision) และค่าเรียกคืน (Recall) เพื่อประเมินประสิทธิภาพโดยรวมของแบบจำลอง คำนวณได้จาก

$$F1 - Score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \quad (3.4)$$

งานวิจัยนี้ประเมินประสิทธิภาพของสถาปัตยกรรม VGG10 - Parameters Reduction Layer InceptionV3 โดยการทดลองกับชุดข้อมูลใบพืชจาก PlantVillage Dataset จากนั้นประเมินจากค่าความถูกต้อง ค่าความแม่นยำ ค่าเรียกคืน และ ค่า F-Measure ของแบบจำลองในการจำแนกโรคของใบพืช

## บทที่ 4

### ผลการดำเนินการวิจัย

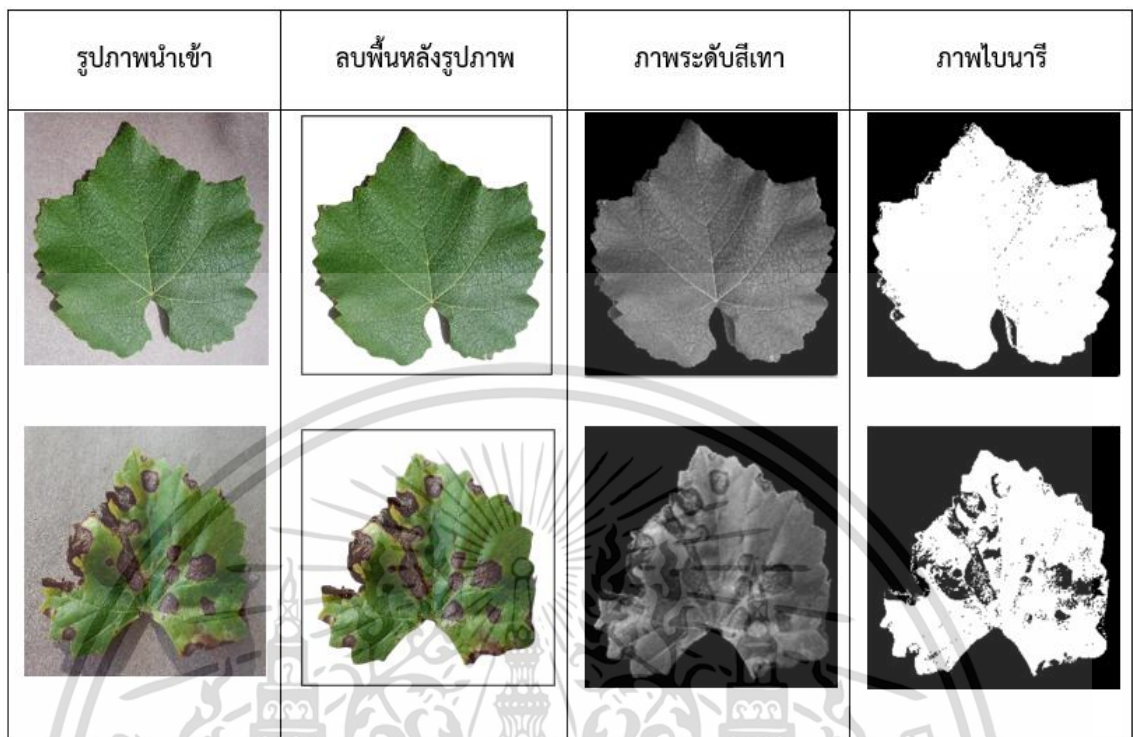
ส่วนนี้นำเสนอผลการดำเนินการวิจัย โดยแบ่งการนำเสนอผลการดำเนินการวิจัยตามข้อสมมติฐานของการวิจัยทั้ง 3 ข้อ ดังนี้ 1) การจำแนกใบพืชโดยใช้การกำหนดจุดเด่นบนขอบใบด้วย Scale-Invariant Feature Transform (SIFT) สามารถเพิ่มค่าความถูกต้องในการจำแนกใบพืชได้หรือไม่ 2) การปรับปรุงสถาปัตยกรรมของ VGG16 ร่วมกับ InceptionV3 with Parameter Reduction Layer สามารถเพิ่มค่าความแม่นยำในการจำแนกโรคของใบพืชได้หรือไม่ และ 3) การปรับปรุงสถาปัตยกรรมของ VGG16 ร่วมกับ InceptionV3 with Parameter Reduction Layer สามารถลดจำนวนพารามิเตอร์ของสถาปัตยกรรมและเวลาในการประมวลผลได้หรือไม่ มีผลการดำเนินการวิจัยดังนี้

#### 4.1 การจำแนกใบพืชโดยใช้การกำหนดจุดเด่นบนขอบใบด้วย Scale-Invariant Feature Transform (SIFT)

ตามข้อสมมติฐานการวิจัยที่ 1 การดำเนินการวิจัยเลือกใช้รูปภาพใบพืชที่สุภาพดีจำนวน 5 คลาส และใบพืชที่เป็นโรคจำนวน 5 คลาส รวม 10 คลาส โดยการหาเส้นขอบของใบพืชและกำหนดจุดเด่นบนขอบใบพืชก่อนจำแนกชนิดของใบพืชด้วยแบบจำลองป่าสุ่ม (Random Forest) มีผลการดำเนินงานของแต่ละขั้นตอนดังนี้

##### 4.1.1 การแปลงค่าสี (Color Transform)

การแปลงค่าสีของรูปภาพใบพืชเป็นขั้นตอนแรกของการประมวลผลรูปภาพใบพืชสำหรับการดำเนินงานของงานวิจัยในส่วนนี้ โดยข้อมูลรูปภาพนำเข้ามีแบบจำลองสีอาร์จีบีและมีพื้นหลังของภาพที่มีแสงและเงาที่แตกต่างกัน ดังนั้นจึงทำการลบพื้นหลังของภาพให้เหลือเพียงรูปภาพของใบพืช จากนั้นแปลงค่าสีจากอาร์จีบีเป็นภาพระดับสีเทาและภาพไบนารีด้วยไลบรารีพื้นฐานของ OpenCV มีการปรับค่าเกณฑ์ต่ำสุดและเกณฑ์สูงสุดของ Threshold ผลการแปลงค่าสีของใบพืชแสดงดังรูปที่ 4.1

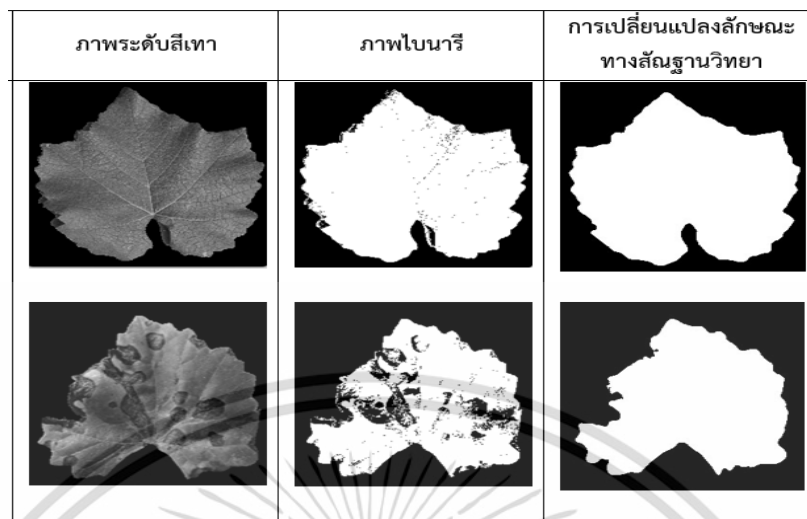


รูปที่ 4.1 ผลการแปลงค่าสีของรูปภาพเป็นภาพระดับสีเทาและภาพไบนารี

จากรูปที่ 4.1 แสดงผลการแปลงค่าสีของภาพใบพืชที่เป็นโรคและใบพืชสุขภาพดี โดยขั้นตอนแรกลบพื้นหลังของรูปภาพเพื่อให้ได้รูปภาพใบพืชที่มีพื้นหลังสีขาว จากนั้นแปลงค่าสีแบบอาร์จีบีเป็นภาพระดับสีเทาและภาพไบนารี ผลจากการแปลงค่าสีของภาพเป็นไบนารีปรากฏว่ายังมีสัญญาณรบกวนของรูปภาพและเมื่อตรวจจับเส้นขอบของใบและขั้นตอนการกำหนดจุดสำคัญบนขอบใบด้วย SIFT ทำให้อัลกอริทึมตรวจจับตรงบริเวณที่เป็นสัญญาณรบกวนของภาพด้วย ดังนั้นจึงลบสัญญาณรบกวนของภาพด้วยการเปลี่ยนแปลงลักษณะทางสัณฐานวิทยาของใบพืชเป็นขั้นตอนต่อไป

#### 4.1.2 การเปลี่ยนแปลงทางสัณฐานวิทยา (Morphological Transformations)

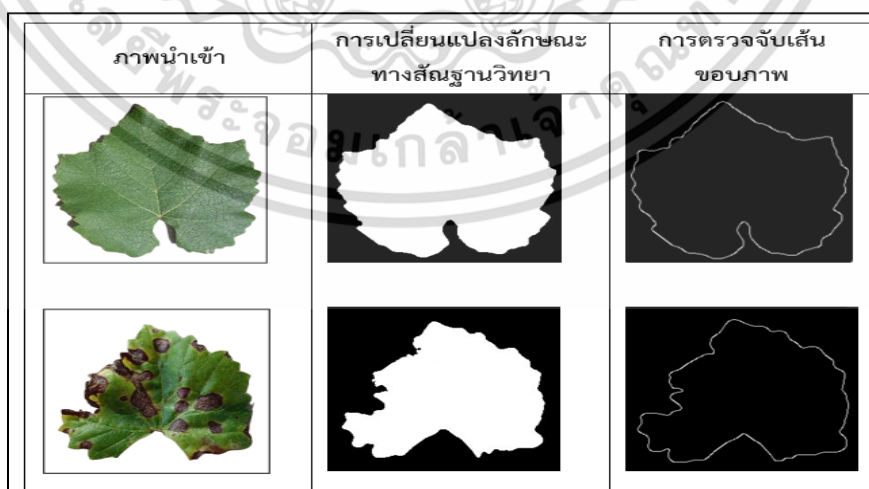
รูปภาพใบพืชที่มีแสงและสีของใบพืชที่แตกต่างกันเมื่อแปลงค่าสีแล้วส่งผลให้ยังคงมีสัญญาณรบกวนของภาพ ดังนั้นในขั้นตอนการเปลี่ยนแปลงลักษณะทางสัณฐานวิทยาของภาพใบพืชจึงใช้ตัวดำเนินการ Opening และ Closing เพื่อลบสัญญาณรบกวนของภาพเพื่อให้มีเฉพาะเส้นขอบที่เป็นรูปร่างของใบพืชเท่านั้น ผลการเปลี่ยนแปลงลักษณะทางสัณฐานวิทยาของภาพใบพืชแสดงดังรูปที่ 4.2



รูปที่ 4.2 ผลการเปลี่ยนแปลงลักษณะทางสัณฐานวิทยาของภาพใบพืช

จากรูปที่ 4.2 เมื่อลบสัญญาณรบกวนของรูปภาพด้วยการเปลี่ยนแปลงลักษณะทางสัณฐานวิทยาของภาพใบพืชโดยใช้ตัวดำเนินการ Opening และ Closing ทำให้ได้รูปภาพใบพืชแบบไบนารีเพื่อใช้ในการตรวจจับขอบภาพและการหาจุดเด่นบนขอบใบพืชในขั้นตอนถัดไป

4.1.3 การตรวจจับขอบภาพ (Edge Detection) หลังจากลบสัญญาณรบกวนของภาพด้วยการเปลี่ยนแปลงลักษณะทางสัณฐานวิทยาทำให้ได้รูปภาพไบนารีที่มีรูปภาพใบพืชสีขาวและพื้นหลังสีดำ ขั้นตอนต่อไปคือการตรวจจับขอบภาพด้วยแคนนี่เพื่อให้ได้เส้นขอบที่เป็นรูปร่างของใบพืช แสดงดังรูปที่ 4.3

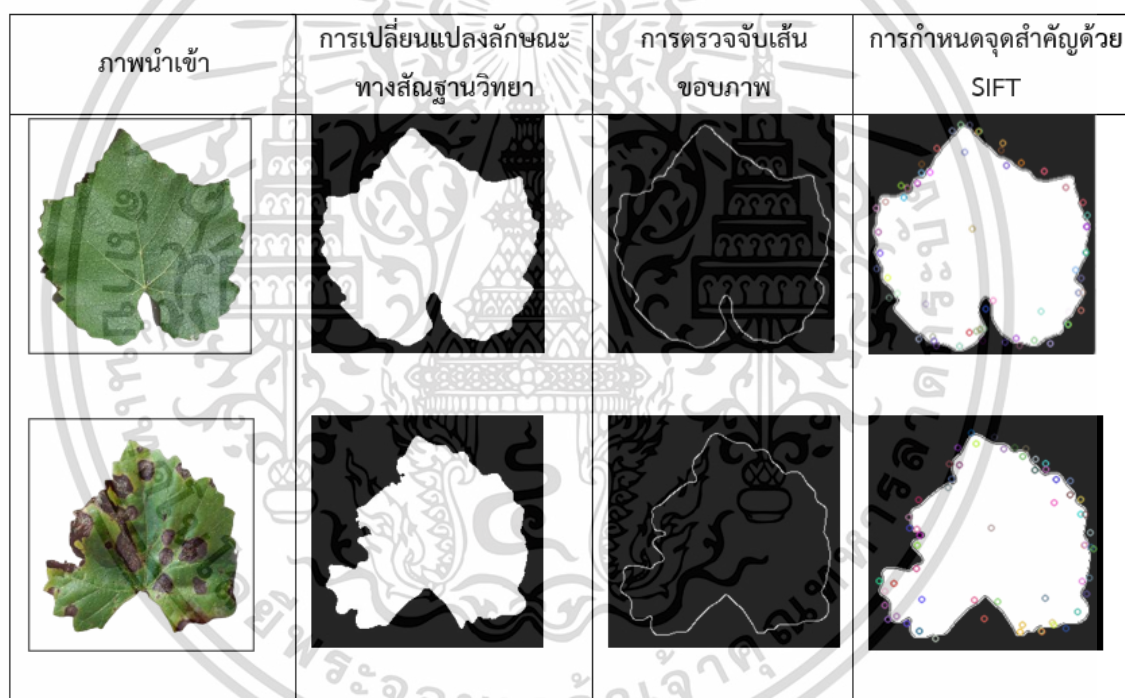


รูปที่ 4.3 การตรวจจับขอบรูปภาพใบพืชด้วย Canny

จากรูปที่ 4.3 เมื่อตรวจจับเส้นขอบของใบพืชด้วยแค่นี้ทำให้ได้ลักษณะเส้นขอบตามรูปร่างของใบพืช โดยใบพืชที่เป็นโรคก็จะมีลักษณะเส้นขอบรูปร่างของใบที่เปลี่ยนแปลงไป

#### 4.1.4 การหาจุดเด่นบนภาพด้วย Scale-Invariant Feature Transform (SIFT)

เมื่อรูปภาพผ่านกระบวนการประมวลผลรูปภาพและตรวจจับขอบภาพด้วยแค่นี้ทำให้ได้รูปภาพไบนารีที่มีลักษณะเป็นรูปร่างของใบพืช ถ้าเป็นใบพืชสุขภาพดีก็จะมีเส้นขอบใบที่มีรอยหยักตามลักษณะของใบพืช แต่หากใบพืชที่เป็นโรคลักษณะรูปร่างของใบพืชจะเปลี่ยนแปลงไป และถัดไปเป็นขั้นตอนการกำหนดจุดสำคัญของขอบใบด้วยอัลกอริทึม SIFT โดย SIFT จะทำการกำหนดจุดสำคัญตามเส้นขอบที่ตรวจจับได้จากแค่นี้แสดงดังรูปที่ 4.4



รูปที่ 4.4 การหาจุดเด่นบนภาพด้วย Scale-Invariant Feature Transform (SIFT)

จากรูปที่ 4.4 เมื่อใช้ SIFT หาจุดเด่นบนภาพใบพืชทำให้ได้จุดเด่นตามเส้นขอบรูปร่างของใบพืช เพื่อใช้เป็นรูปภาพนำเข้าสำหรับการสกัดคุณสมบัติรูปภาพและการจำแนกประเภทใบพืชด้วยแบบจำลอง CNN

#### 4.1.5 การสกัดคุณสมบัติและจำแนกข้อมูลใบพืช











ขั้นตอนการสกัดคุณสมบัติและจำแนกข้อมูลใบพืชโดยใช้คุณสมบัติรูปร่างของใบพืชที่ผ่านการประมวลผลรูปภาพและกำหนดจุดเด่นบนขอบใบ โดยใช้แบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชันที่เป็น Sequential Model สกัดคุณสมบัติของรูปภาพทั้งหมดเพื่อเป็นชุดคุณสมบัติสำหรับการจำแนกชนิดของใบพืชต่อไป การใช้ CNN สกัดคุณสมบัตินั้นจะใช้งานผ่านฟังก์ชัน Sequential Model ของไลบรารี Keras ส่วนการจำแนกชนิดของใบพืชใช้ Random Forest เป็นตัวจำแนกข้อมูล โดยแบ่งชุดข้อมูลออกเป็นชุดข้อมูลการเรียนรู้ (Training Set) 70% ชุดข้อมูลสำหรับการตรวจสอบ (Validation Set) 20% และชุดข้อมูลทดสอบ (Testing Set) 10% การทำงานจำนวน 70 epochs โดยทำการทดลองและเปรียบเทียบกับชุดข้อมูลใบพืชที่กำหนดจุดสำคัญบนใบพืชแต่ยังไม่ผ่านการลบสัญญาณรบกวนของภาพ การใช้คุณสมบัติจากรูปร่าง และคุณสมบัติจากพื้นผิว ผลการทดลองพบว่าวิธีที่นำเสนอสามารถให้ค่าความถูกต้องของการจำแนกชนิดใบพืชสูงสุดร้อยละ 95.62 ดังตารางที่ 4.1

ตารางที่ 4.1 ผลการเปรียบเทียบค่าความถูกต้องการจำแนกข้อมูล

	คุณสมบัติ	ค่าความถูกต้อง (%)
1.	คุณสมบัติรูปร่าง (Shape feature)	75.84
2.	คุณสมบัติพื้นผิว (GLCM)	70.89
3.	คุณสมบัติรูปร่าง + คุณสมบัติพื้นผิว	81.84
4.	การกำหนดจุดเด่นบนขอบใบ โดยไม่มีการใช้การเปลี่ยนแปลงลักษณะทางสัณฐานวิทยา	92.44
5.	การกำหนดจุดเด่นบนขอบใบ + การเปลี่ยนแปลงลักษณะทางสัณฐานวิทยา (วิธีการที่นำเสนอ)	95.62

จากตารางที่ 4.1 ผลการทดลองพบว่าการจำแนกชนิดของใบพืชโดยใช้คุณสมบัติจากการกำหนดจุดเด่นบนขอบใบพืชร่วมกับการเปลี่ยนแปลงลักษณะทางสัณฐานวิทยา (วิธีการที่นำเสนอ) ให้ค่าความถูกต้องในการจำแนกสูงสุดร้อยละ 95.62 รองลงมาคือการใช้คุณสมบัติจากการกำหนดจุดเด่นบนขอบใบพืชโดยไม่มีการเปลี่ยนแปลงลักษณะทางสัณฐานวิทยา ค่าความถูกต้องร้อยละ 92.44 การจำแนกโดยใช้คุณสมบัติรูปร่างร่วมกับคุณสมบัติพื้นผิวค่าความถูกต้องร้อยละ 81.84 ส่วนการใช้คุณสมบัติรูปร่าง

ค่าความถูกต้องร้อยละ 75.84 และใช้คุณสมบัติพื้นผิวค่าความถูกต้องร้อยละ 70.89 ตามลำดับ โดยคลาสที่สามารถทำนายได้แม่นยำที่สุดคือ Tomato\_healthy, Cherry\_Disease และ Potato\_healthy มีค่าความถูกต้องร้อยละ 98.04, 98.00, 96.00 ตามลำดับ ส่วนคลาสที่ทำนายแม่นยำน้อยที่สุดคือ Tomato\_Disease, Potato\_Disease และ Grape\_Disease มีค่าความถูกต้องร้อยละ 81.55, 87.00, 92.00 ตามลำดับ จะเห็นได้ว่าคลาส Tomato\_Disease มีค่าความถูกต้องน้อยที่สุดเนื่องจากใบมะเขือเทศที่เป็นโรคส่งผลให้เส้นขอบใบและรอยหยักขอบใบเปลี่ยนแปลงไป ดังนั้นเมื่อนำใบมะเขือเทศที่เป็นโรคผ่านกระบวนการประมวลผลรูปภาพจึงทำให้รูปร่างของใบมะเขือเทศคล้ายกับใบมันฝรั่งในคลาส Potato\_Disease เล็กน้อย เช่นเดียวกับคลาสของใบพืชที่เป็นโรค เพราะโรคบางโรคทำให้สีของใบเปลี่ยนแปลงไปแต่รูปร่างของใบเปลี่ยนแปลงเล็กน้อย จึงทำให้ยังคงมีรูปร่างที่คล้ายกับคลาสที่เป็นใบพืชสุขภาพดี แสดงดังรูปที่ 4.5

	Cherry_Disease	Cherry_healthy	Grape_Disease	Grape_healthy	Potato_Disease	Potato_healthy	Strawberry_Disease	Strawberry_healthy	Tomato_Disease	Tomato_healthy	No. Leaves
	98	0	0	0	1	0	0	0	1	0	100
	0	93	0	2	0	1	0	4	0	0	100
	0	0	92	1	1	0	2	0	4	0	100
	0	1	2	93	0	0	1	2	1	0	100
	0	2	0	0	87	5	2	2	1	1	100
	0	1	0	0	1	96	2	0	0	0	100
	0	0	1	0	0	1	93	3	2	0	100
	0	2	1	1	0	1	0	93	1	1	100
	4	0	2	1	5	3	1	0	84	3	103
	0	0	0	0	0	2	0	0	0	100	102

รูปที่ 4.5 Confusion Matrix ของการจำแนกใบพืชโดยใช้การกำหนดจุดเด่นบนขอบใบด้วย SIFT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

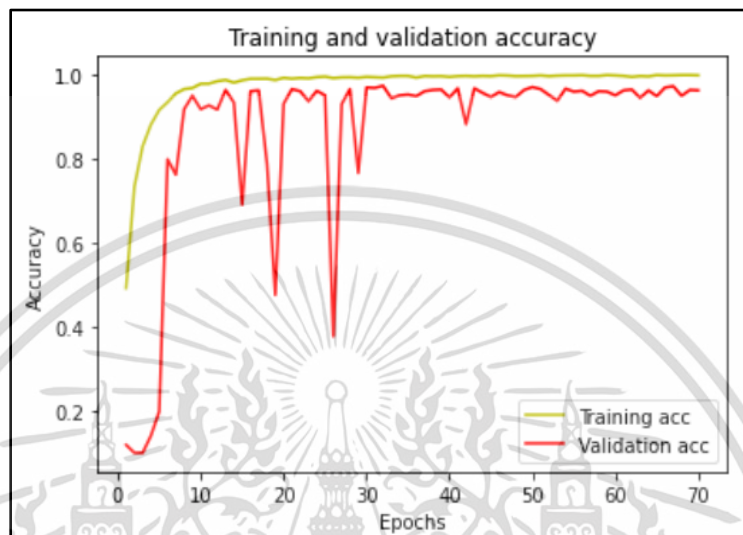
นอกจากนี้เพื่อประเมินผลการทำงานของวิธีการที่นำเสนอ จึงได้ทำการเปรียบเทียบค่าความถูกต้องของวิธีการที่นำเสนอกับแบบจำลองของ CNN อื่นๆ เช่น ResNet50, VGG16, InceptionV3, InceptionResNetV2, Xception และ DenseNet121 แสดงดังตารางที่ 4.2

ตารางที่ 4.2 ผลการเปรียบเทียบค่าความถูกต้องการจำแนกข้อมูลของวิธีการที่นำเสนอกับแบบจำลองอื่นของ CNN

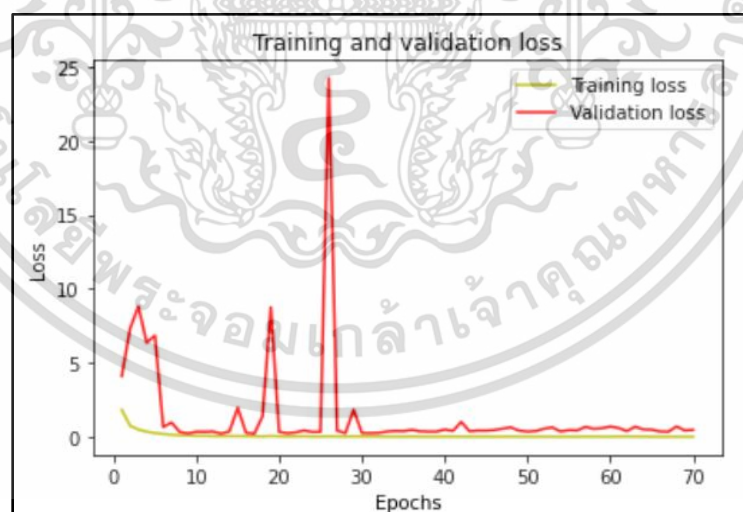
แบบจำลอง CNN		ค่าความถูกต้อง (%)
1.	ResNet50	86.07
2.	VGG16	93.73
3.	InceptionV3	95.42
4.	InceptionResNetV2	95.62
5.	Xception	96.52
6.	DenseNet121	96.82
7.	<b>Proposed Methodology</b>	<b>95.62</b>

จากตารางที่ 4.2 แสดงผลการเปรียบเทียบพบว่าวิธีการที่นำเสนอโดยการกำหนดจุดเด่นบนขอบใบพืชและสกัดคุณสมบัติของรูปภาพด้วยแบบจำลอง CNN จำแนกใบพืชด้วย Random Forest มีค่าความถูกต้องในการจำแนกข้อมูลใบพืชที่ดีกว่า ResNet50, VGG16 และ InceptionV3 มีค่าความถูกต้องเท่ากับ InceptionResNetV2 ดังนั้นแสดงให้เห็นว่าวิธีการที่นำเสนอสามารถเพิ่มประสิทธิภาพการจำแนกข้อมูลของ Random Forest ให้ดีขึ้นได้เพราะการสกัดคุณลักษณะของรูปภาพด้วยแบบจำลอง CNN จะทำให้ได้คุณสมบัติของรูปภาพที่ซับซ้อนและครบถ้วนมากขึ้น และเมื่อจำแนกข้อมูลด้วย Random Forest ต้นไม้ตัดสินใจแต่ละต้นจะได้รับข้อมูลคุณสมบัติที่แตกต่างกันและเป็นอิสระจากกัน ดังนั้นการทำงานของต้นไม้ตัดสินใจแต่ละต้นจะไม่มีความสัมพันธ์กัน แต่อย่างไรก็ตามวิธีการที่นำเสนอยังคงมีประสิทธิภาพในการจำแนกข้อมูลใบพืชน้อยกว่า Xception และ DenseNet121 เนื่องจาก Xception มีการใช้ Depthwise Separable Convolutions จึงทำให้มีประสิทธิภาพในการสกัดคุณลักษณะและจำแนกข้อมูลที่ดีกว่า และ DenseNet121 มีการรับค่าคุณสมบัติโดยตรงจากเลเยอร์ก่อนหน้าทุกเลเยอร์ จึงทำให้ได้คุณสมบัติที่ครบถ้วนมากกว่า และเมื่อพิจารณาค่าความถูกต้องและค่าฟังก์ชันการสูญเสียของ

การจำแนกข้อมูลด้วยชุดข้อมูลสำหรับการเรียนรู้ของแบบจำลองและชุดข้อมูลสำหรับการตรวจสอบ แสดงดังรูปที่ 4.6 และรูปที่ 4.7



รูปที่ 4.6 ค่าความถูกต้องของการจำแนกข้อมูลด้วยชุดข้อมูลสำหรับการเรียนรู้ของแบบจำลองและชุดข้อมูลสำหรับการตรวจสอบ



รูปที่ 4.7 ค่าฟังก์ชันการสูญเสียของการจำแนกข้อมูลด้วยชุดข้อมูลสำหรับการเรียนรู้ของแบบจำลองและชุดข้อมูลสำหรับการตรวจสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.6 และรูปที่ 4.7 แสดงค่าความถูกต้องและค่าฟังก์ชันการสูญเสียของการจำแนกข้อมูล ด้วยชุดข้อมูลสำหรับการเรียนรู้ของแบบจำลองและชุดข้อมูลสำหรับการตรวจสอบพบว่า การทำงานของแบบจำลองตั้งแต่เริ่มทำงานจนถึงรอบการทำงานที่ 45 แบบจำลองยังคงมีค่าความถูกต้องและค่าฟังก์ชันการสูญเสียที่ไม่คงที่ แต่เมื่อรอบการทำงานตั้งแต่ 46 จนถึงรอบที่ 70 พบว่าค่าความถูกต้องของแบบจำลองเริ่มมีค่าคงที่ที่สูงขึ้นโดยมีค่าความถูกต้องประมาณร้อยละ 90 และค่าฟังก์ชันการสูญเสียของแบบจำลองลดลงและมีค่าคงที่เข้าหาค่าศูนย์มากขึ้น

จากข้อสมมติฐานการวิจัย การจำแนกใบพืชโดยใช้การกำหนดจุดเด่นบนขอบใบด้วย Scale-Invariant Feature Transform (SIFT) สามารถเพิ่มค่าความถูกต้องในการจำแนกใบพืชได้หรือไม่ ผลการดำเนินงานพบว่าวิธีการที่นำเสนอมีค่าความถูกต้องในการจำแนกข้อมูลร้อยละ 95.62 ซึ่งมีความถูกต้องในการจำแนกข้อมูลได้ดีกว่าการใช้คุณสมบัติรูปร่าง คุณสมบัติพื้นผิว และคุณสมบัติรูปร่างร่วมกับคุณสมบัติพื้นผิว และวิธีการที่นำเสนอสามารถทำงานได้ดีกับชุดข้อมูลใบพืชที่มีรูปร่างของใบแตกต่างกัน ส่วนใบพืชที่มีลักษณะรูปร่างที่คล้ายกันอาจทำให้ค่าความแม่นยำในการจำแนกข้อมูลลดลงจึงอาจจะต้องใช้คุณสมบัติอื่นร่วมด้วย นอกจากนี้เมื่อเปรียบเทียบกับวิธีการที่นำเสนอกับแบบจำลองอื่นของ CNN เช่น ResNet50, VGG16, InceptionV3, InceptionResNetV2, Xception และ DenseNet121 พบว่าวิธีการที่นำเสนอสามารถมีค่าความถูกต้องของการจำแนกข้อมูลดีกว่าแบบจำลอง ResNet50, VGG16 และ InceptionV3 แต่ก็ยังคงมีค่าความถูกต้องน้อยกว่า Xception และ DenseNet121

จากผลการดำเนินงานของข้อสมมติฐานที่ 1 พบว่าแบบจำลองโครงข่ายประสาทเทียม VGG16 และ InceptionV3 มีค่าความถูกต้องของการจำแนกข้อมูลใกล้เคียงกับวิธีการที่นำเสนอ แต่แบบจำลอง VGG16 เป็นโครงข่ายขนาดใหญ่และใช้เวลาในการคำนวณที่ค่อนข้างนาน ในขณะที่แบบจำลอง InceptionV3 มีการทำงานที่รวดเร็วกว่า ดังนั้นจึงเป็นข้อสมมติฐานข้อที่ 2 ของงานวิจัยคือการปรับปรุงสถาปัตยกรรมของ VGG16 ร่วมกับ InceptionV3 Block

## 4.2 การจำแนกโรคของใบพืชด้วยสถาปัตยกรรม VGG10 – InceptionV3 with Parameter Reduction Layer

ในส่วนนี้นำเสนอผลการดำเนินงานของการปรับปรุงสถาปัตยกรรมโครงข่ายประสาทเทียม VGG16 ร่วมกับ InceptionV3 with Parameter Reduction Layer ที่มีชื่อว่า VGG10 – InceptionV3 with Parameter Reduction Layer โดยมีจุดมุ่งหมายเพื่อปรับปรุงข้อจำกัดของ VGG16 ที่เป็นโครงข่ายขนาดใหญ่และต้องใช้ต้นทุนการคำนวณที่สูง ให้เป็นโครงข่ายที่มีขนาดเล็กลง โดยการปรับลดจำนวน

พารามิเตอร์และต้นทุนการคำนวณแต่แบบจำลองยังคงให้ค่าความถูกต้องในการจำแนกโรคของใบพืชที่สูง และให้สามารถใช้แบบจำลองที่มีขนาดเล็กบนอุปกรณ์พกพาหรืออุปกรณ์ที่มีทรัพยากรน้อยได้ และเพื่อการประเมินประสิทธิภาพของแบบจำลองที่นำเสนอ เบื้องต้นได้ทำการทดลองกับการจำแนกโรคของใบมะเขือเทศ โดยใช้ชุดข้อมูลจาก PlantVillage dataset ที่มีจำนวนรูปภาพใบมะเขือเทศทั้งหมด 18,159 รูป แบ่งเป็นใบมะเขือเทศที่เป็นโรคจำนวน 9 คลาส และใบที่มีสุขภาพดีจำนวน 1 คลาส ในขั้นตอนการประมวลผลรูปภาพมีการเปลี่ยนขนาดรูปภาพเป็น 224 x 224 พิกเซล และทำการลบพื้นหลังของรูปภาพ ซึ่งได้ทำการแบ่งชุดข้อมูลสำหรับการเรียนรู้ของแบบจำลอง (Training Set) 70% ชุดข้อมูลสำหรับการตรวจสอบ (Validation Set) 20% และชุดข้อมูลสำหรับทดสอบ (Testing Set) 10% การทดลองใช้ Adam เป็นเครื่องมือเพิ่มประสิทธิภาพ (Optimizer) กำหนดค่า Learning rate เป็น 0.001 ใช้ Batch size เป็น 64 และทำการทดลองจำนวน 50 epochs ผลการทดลองแสดงดังตารางที่ 4.3

ตารางที่ 4.3 เปรียบเทียบค่าความถูกต้องจากการปรับลดจำนวนชั้นของแบบจำลอง VGG16 ที่ใช้ร่วมกับ InceptionV3 with Parameter Reduction Layer

จำนวนชั้นของ VGG16	ค่าความถูกต้อง (%)	ฟังก์ชันการสูญเสีย	เวลา	พารามิเตอร์	ค่า F1 Score
16	94.70	0.2325	728s	14,715,108	92.44
15	97.70	0.0698	676s	12,355,300	97.07
14	99.12	0.0341	643s	9,995,492	98.85
13	98.86	0.0417	563s	7,648,996	98.41
12	91.44	0.2992	517s	5,289,188	89.32
11	97.10	0.0964	412s	2,929,380	96.10
<b>10</b>	<b>99.27</b>	<b>0.0267</b>	<b>386s</b>	<b>1,767,652</b>	<b>99.07</b>
9	98.69	0.0471	379s	1,177,572	98.32
8	97.75	0.0236	290s	1,029,988	97.20

จากตารางที่ 4.3 พบว่าเมื่อมีการประยุกต์ใช้ InceptionV3 with Parameter Reduction Layer ร่วมกับแบบจำลองพื้นฐานของ VGG16 ที่มีจำนวน 16 ชั้น สามารถปรับลดจำนวนพารามิเตอร์ของแบบจำลองจากประมาณ 138.4 ล้านพารามิเตอร์ เหลือเพียงประมาณ 14.7 ล้านพารามิเตอร์ และเมื่อทำ

การปรับลดจำนวนชั้นของแบบจำลอง VGG16 จนถึงจำนวน 10 ชั้น พบว่าสามารถให้ค่าความถูกต้องสูงสุดสำหรับการจำแนกโรคใบมะเขือเทศคือมีค่าความถูกต้องร้อยละ 99.27 และลดจำนวนพารามิเตอร์จากประมาณ 14.7 ล้านพารามิเตอร์เหลือเพียง 1,767,652 พารามิเตอร์ มีค่า F1 score เท่ากับร้อยละ 99.07 แต่เมื่อปรับลดจำนวนชั้นของแบบจำลอง VGG16 จนถึงจำนวน 9 ชั้นและ 8 ชั้น พบว่ามีจำนวนพารามิเตอร์ลดลงแต่ก็มีค่าความถูกต้องที่ลดลงเช่นกัน โดยวิธีที่นำเสนอให้ค่าความถูกต้องในการจำแนกโรคของใบมะเขือเทศแยกตามคลาสต่างๆ ตาม Confusion Matrix แสดงดังรูปที่ 4.8

class		0	1	2	3	4	5	6	7	8	9
Bacterial spot	0	460	0	0	0	0	0	0	0	0	0
Early blight	1	0	231	2	0	0	0	2	0	0	0
Late blight	2	0	5	407	1	2	0	0	0	1	0
Leaf Mold	3	0	0	0	223	1	1	0	0	0	0
Septoria leaf spot	4	1	2	0	0	385	0	0	1	0	0
Spider mites	5	0	0	0	0	0	364	0	2	4	0
Target spot	6	0	0	0	0	0	1	313	0	1	0
Yellow leaf curl virus	7	0	0	0	0	0	0	0	1106	0	0
Healthy	8	0	0	0	0	0	0	0	0	353	0
Mosaic virus	9	0	0	1	0	0	0	0	1	0	90

รูปที่ 4.8 Confusion Matrix การจำแนกโรคใบมะเขือเทศด้วย VGG10 – InceptionV3 with Parameter Reduction Layer

จากรูปที่ 4.8 คลาสที่สามารถจำแนกได้ค่าความถูกต้องสูงสุดคือ Bacterial Spot, Yellow leaf Curl Virus และ Healthy มีค่าความถูกต้องร้อยละ 100 รองลงมาคือ Target Spot และ Leaf Mold มีค่าความถูกต้องร้อยละ 99.37 และ 99.11 ตามลำดับ ส่วนคลาสนี้ที่มีค่าความถูกต้องน้อยที่สุดคือ Mosaic Virus และ Late blight มีค่าความถูกต้องร้อยละ 97.83 และ 97.84 ตามลำดับ เนื่องจากใบมะเขือเทศที่เป็นโรคในคลาส Late Blight ใบจะมีสีน้ำตาลไหม้เกือบทั้งใบจนมีลักษณะคล้ายกับใบที่เป็นโรคในคลาส Early Blight ดังนั้นแบบจำลองจึงจำแนกใบมะเขือเทศจากคลาส Late Blight เป็นคลาส Early Blight ส่วนใบมะเขือเทศที่เป็นโรค Mosaic Virus ใบจะมีสีเขียวอ่อนจนเกือบสีเหลืองคล้ายกับใบที่เป็นโรค Yellow Leaf Curl Virus และคลาส Late Blight จึงทำให้แบบจำลองจำแนกโรคใบมะเขือเทศผิดเล็กน้อย

นอกจากนี้เพื่อเป็นการประเมินประสิทธิภาพของแบบจำลองที่นำเสนอจึงได้ทำการเปรียบเทียบการจำแนกโรคใบมะเขือเทศโดยใช้วิธีที่นำเสนอเปรียบเทียบกับแบบจำลองอื่นๆ ของ Convolution Neural Network ได้แก่ VGG16, InceptionV3, DenseNet121, MobileNetV2, และ ResNet50 แสดงผลการเปรียบเทียบดังตารางที่ 4.4

ตารางที่ 4.4 เปรียบเทียบค่าความถูกต้องของการจำแนกโรคใบมะเขือเทศระหว่างแบบจำลองที่นำเสนอกับแบบจำลองของ Convolution Neural Network

แบบจำลอง	ค่าความถูกต้อง (%)	ฟังก์ชันการสูญเสีย	เวลา	พารามิเตอร์
1. ResNet50	57.31	1.3035	699s	23,608,202
2. VGG16	86.19	0.4178	875s	14,719,818
3. InceptionV3	94.22	0.2311	196s	21,823,274
4. DenseNet121	94.42	0.1644	416s	7,047,754
5. MobileNetV2	98.59	0.0759	117s	2,270,794
6. Proposed Model	99.27	0.0267	386s	1,767,652

จากตารางที่ 4.4 ผลการเปรียบเทียบค่าความถูกต้อง พบว่าแบบจำลองที่นำเสนอให้ค่าความถูกต้องในการจำแนกโรคใบมะเขือเทศสูงที่สุดร้อยละ 99.27 รองลงมาคือ MobileNetV2, DenseNet121, InceptionV3, VGG16 และ ResNet50 มีค่าความถูกต้องร้อยละ 98.59, 94.42, 94.22, 86.19 และ 57.31 ตามลำดับ และแบบจำลองที่นำเสนอใช้จำนวนพารามิเตอร์ที่น้อยกว่า MobileNetV2 ซึ่งเป็นแบบจำลองขนาดเล็กเหมาะสำหรับการใช้งานบนอุปกรณ์พกพา ถึงแม้ว่าวิธีที่นำเสนอจะใช้เวลาในการประมวลผลนานกว่าแต่เมื่อเปรียบเทียบค่าความถูกต้องและค่าการสูญเสียแล้ววิธีการที่นำเสนอก็ยังคงให้ผลลัพธ์ที่ดีกว่า

จากการประเมินประสิทธิภาพเบื้องต้นของแบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer ด้วยการจำแนกโรคของใบมะเขือเทศและได้ผลการดำเนินงานที่อยู่ในระดับดี โดยมีค่าความถูกต้องในการจำแนกร้อยละ 99.27 วิธีการที่นำเสนอสามารถปรับลดจำนวนพารามิเตอร์ของแบบจำลองเหลือเพียง 1,767,652 พารามิเตอร์ ซึ่งถือได้ว่าเป็นแบบจำลองที่มีขนาดเล็กแต่ยังคงมีค่า

ความถูกต้องสูงสำหรับการจำแนกข้อมูล และเมื่อเปรียบเทียบกับแบบจำลอง CNN อื่นๆ เช่น VGG16, InceptionV3, DenseNet121, MobileNetV2 และ ResNet50 พบว่าวิธีการที่นำเสนอให้ค่าความถูกต้องในการจำแนกโรคใบมะเขือเทศสูงที่สุด ดังนั้นจึงประเมินประสิทธิภาพของแบบจำลองด้วยการจำแนกโรคของพืชที่หลากหลายชนิดมากขึ้น โดยใช้ข้อมูลรูปภาพใบพืชจาก PlantVillage Dataset ที่ประกอบด้วยใบพืช 14 ชนิด แบ่งออกเป็น 38 คลาส มีรูปภาพทั้งหมด 54,304 รูป โดยทดลองจำนวน 10 รอบเพื่อหาค่าความถูกต้องเฉลี่ยของแบบจำลอง ผลการทดลองแสดงดังตารางที่ 4.5

ตารางที่ 4.5 ผลค่าความถูกต้องเฉลี่ยของการจำแนกโรคใบพืช 38 คลาสด้วยแบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer

จำนวนรอบ การทดลอง	ค่าความถูกต้อง (%)	ฟังก์ชัน การสูญเสีย	เวลา	ค่า F1 Score
1	99.56	0.0148	394	99.47
2	99.37	0.0099	387	99.22
3	99.38	0.0097	405	99.26
4	99.47	0.0095	398	99.34
5	99.08	0.0112	336	98.78
6	99.11	0.0229	419	98.95
7	99.30	0.0105	402	99.19
8	99.56	0.0083	391	99.47
9	99.48	0.0115	389	99.39
10	99.00	0.0117	339	98.87
<b>Average</b>	<b>99.33</b>	<b>0.0120</b>	<b>386</b>	<b>99.19</b>

จากตารางที่ 4.5 แสดงผลค่าความถูกต้องเฉลี่ยของแบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer จากการทดลองจำแนกโรคของใบพืชจำนวน 38 คลาส ประเมินผลจากการทดลองทั้งหมด 10 รอบ แบบจำลองที่นำเสนอให้ค่าความถูกต้องของการจำแนกโรคใบพืชเฉลี่ยร้อยละ 99.33 ค่า F1 Score เฉลี่ยเท่ากับร้อยละ 99.19 ใช้เวลาเฉลี่ยเท่ากับ 386 วินาที/ 50 epochs เมื่อพิจารณาผลการจำแนกโรคของใบพืชแยกตามคลาสต่างๆ ตาม Confusion Matrix แสดงดังรูปที่ 4.9

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	
1	159	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
2	0	130	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	1	0	363	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
4	0	1	0	158	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	
5	0	0	0	0	334	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
6	0	0	0	0	0	245	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
7	0	0	0	1	0	0	203	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	0	0	0	0	0	0	0	130	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
9	0	0	0	0	0	0	0	0	273	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10	0	0	0	0	0	0	0	2	0	230	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
11	0	0	0	0	0	0	0	0	0	0	267	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
12	0	0	0	0	0	0	0	0	0	0	0	271	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
13	0	0	0	0	0	0	0	0	0	0	0	0	311	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
14	0	0	0	0	0	0	0	0	0	0	0	0	0	250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	129	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1136	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	494	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	130	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	234	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	1	327	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	235	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	232	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	130	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	130	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1052	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	402	0	0	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	254	0	0	0	2	0	0	0	0	0	0	0	
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	130	0	0	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	460	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	232	2	0	1	0	0	0	0	0	0	
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	411	0	0	0	0	0	0	1	0	
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	220	4	1	0	0	0	0	0	
33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	3	0	0	384	0	0	0	0	0	0	
34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	370	0	0	0	0		
35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	312	0	0	1	0	0		
36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1106	0	0	0	
37	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	6	0	0	345	0	0	
38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	130	0

รูปที่ 4.9 Confusion Matrix การจำแนกโรคใบพืช 38 คลาส ด้วย VGG10 – InceptionV3 with Parameter Reduction Layer

จากรูปที่ 4.9 ผลการจำแนกโรคของใบพืชทั้ง 38 คลาสด้วยแบบจำลองที่นำเสนอพบว่าแบบจำลองสามารถจำแนกโรคของใบพืชมีค่าความถูกต้องอยู่ในระดับดี มีความผิดพลาดของการจำแนกโรคใบพืชบางโรคซึ่งอาจเกิดจากรูปภาพมีลักษณะสีและรูปร่างของใบพืชคล้ายกัน แต่เมื่อพิจารณาจำนวนรูปภาพที่จำแนกผิดและคิดเป็นร้อยละของค่าความถูกต้องแยกตามคลาส แสดงดังตารางที่ 4.6

ตารางที่ 4.6 ค่าความถูกต้องแยกตามคลาสของการจำแนกโรคใบพืช 38 คลาสด้วยแบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer

คลาส	จำนวนรูปทั้งหมด	จำนวนรูปที่จำแนกถูก	ร้อยละ	คลาส	จำนวนรูปทั้งหมด	จำนวนรูปที่จำแนกถูก	ร้อยละ
1.	159	159	100	20.	330	327	99.09
2.	130	130	100	21.	235	235	100
3.	364	363	99.73	22.	235	232	98.72
4.	161	158	98.14	23.	130	130	100
5.	335	334	99.70	24.	130	130	100
6.	245	245	100	25.	1053	1052	99.91
7.	205	203	99.02	26.	402	402	100
8.	134	130	97.01	27.	256	254	99.22
9.	273	273	100	28.	130	130	100
10.	232	230	99.14	29.	460	460	100
11.	267	267	100	30.	235	232	98.72
12.	271	271	100	31.	416	411	98.80
13.	311	311	100	32.	225	220	97.78
14.	250	250	100	33.	389	384	98.71
15.	130	129	99.23	34.	370	370	100
16.	1136	1136	100	35.	315	312	99.05
17.	494	494	100	36.	1106	1106	100
18.	130	130	100	37.	353	345	97.73
19.	234	234	100	38.	130	130	100

จากตารางที่ 4.6 แสดงผลร้อยละค่าความถูกต้องของการจำแนกโรคใบพืชทั้ง 38 คลาสด้วยแบบจำลองที่นำเสนอพบว่าแบบจำลองที่นำเสนอสามารถจำแนกโรคของใบพืชได้แม่นยำร้อยละ 100

จำนวน 21 คลาส จาก 38 คลาส และค่าความถูกต้องมากกว่าร้อยละ 99 จำนวน 9 คลาส โดยมีคลาสที่มีค่าความถูกต้องน้อยกว่าร้อยละ 99 จำนวน 8 คลาส ได้แก่ Apple\_scab ร้อยละ 98.14 Corn\_leaf\_spot ร้อยละ 97.01 Potato\_Late\_blight ร้อยละ 98.72 Tomato\_Early\_blight ร้อยละ 98.72 Tomato\_Late\_blight ร้อยละ 98.80 Tomato\_Leaf\_Mold ร้อยละ 97.78 Tomato\_Septoria\_leaf\_spot ร้อยละ 98.71 และTomato\_healthy ร้อยละ 97.73 เมื่อพิจารณาค่าความถูกต้องแยกตามคลาสจะเห็นได้ว่า คลาสที่แบบจำลองให้ค่าความถูกต้องน้อยกว่าร้อยละ 99 ส่วนใหญ่จะเป็นโรคของใบมะเขือเทศ ที่อาจจะเกิดจากลักษณะสีของใบที่เป็นโรคต่างกันแต่มีสีที่คล้ายกันจึงทำให้แบบจำลองจำแนกโรคของใบพืชผิดพลาด จากนั้นจึงเปรียบเทียบค่าความถูกต้องของแบบจำลองที่นำเสนอกับแบบจำลองอื่นๆ ของ Convolution Neural Network ได้แก่ VGG16, InceptionV3, DenseNet121, MobileNetV2, และ ResNet50 แสดงผลการเปรียบเทียบดังตารางที่ 4.7

ตารางที่ 4.7 เปรียบเทียบค่าความถูกต้องของการจำแนกโรคใบพืช 38 คลาสด้วยแบบจำลองที่นำเสนอกับแบบจำลองของ Convolution Neural Network

แบบจำลอง	ค่าความถูกต้อง (%)	ฟังก์ชันการสูญเสีย	เวลา	พารามิเตอร์	ค่า F1 Score
1. ResNet50	84.67	1.5175	375	23,665,574	44.49
2. VGG16	89.61	0.3109	670	14,734,182	88.55
3. Inception-V3	94.43	0.1224	305	21,880,646	93.90
4. DenseNet121	94.69	0.1342	417	7,076,454	94.70
5. MobileNetV2	99.28	0.0054	244.50	2,306,662	99.15
6. Proposed Model	99.33	0.0120	386	1,886,708	99.19

จากตารางที่ 4.7 ผลการเปรียบเทียบค่าความถูกต้องของการจำแนกโรคใบพืช 38 คลาสด้วยแบบจำลองที่นำเสนอกับแบบจำลองของ CNN พบว่าวิธีการที่นำเสนอมีค่าความถูกต้องและค่า F1 score มากกว่าแบบจำลอง ResNet50, VGG16, InceptionV3, DenseNet121 และ MobileNetV2 และค่าการสูญเสียยังคงดีกว่าแบบจำลองอื่นๆ ยกเว้นแบบจำลอง MobileNetV2 และเมื่อพิจารณาด้านเวลาการประมวลผลพบว่าแบบจำลองที่นำเสนอใช้เวลามากกว่าแบบจำลอง MobileNetV2, InceptionV3 และ

ResNet50 แต่เมื่อพิจารณาค่าความถูกต้องและเวลาร่วมกัน พบว่าแบบจำลองที่นำเสนอดีกว่าแบบจำลอง ResNet50, VGG16, InceptionV3, DenseNet121 แต่แบบจำลองที่นำเสนอให้ผลค่าความถูกต้องใกล้เคียงกับแบบจำลอง MobileNetV2 และยังคงใช้เวลามากกว่า เนื่องจากแบบจำลองที่นำเสนอยังคงใช้สถาปัตยกรรมแบบลำดับชั้นของแบบจำลอง VGG16 ซึ่งส่งผลให้แบบจำลองที่นำเสนอมีเวลาการประมวลผลนานกว่า MobileNetV2, InceptionV3 และ ResNet50 นอกจากนี้ได้ทำการเปรียบเทียบผลการทดลองของแบบจำลองที่นำเสนอกับผลงานวิจัยที่เกี่ยวข้องในบทที่ 2 แสดงดังตารางที่ 4.8

ตารางที่ 4.8 เปรียบเทียบค่าความถูกต้องของแบบจำลองที่นำเสนอกับผลงานวิจัยที่เกี่ยวข้อง

	ผู้แต่ง	ชุดข้อมูล	ค่าความถูกต้อง	แบบจำลองที่นำเสนอ
1.	Monika Bansal, Munish Kumar, Manish Kumar	Caltech-101	83.27%	
2.	Payal Chhabra, Naresh Kumar Garg, Munish Kumar	Wang database Corel database	86.20% 99.53%	
3.	Surbhi Gupta, Munish Kumar, Anupam Garg	Caltech-101	85.6%	
4.	T. Gayathri Devi, P. Neelamegam	rice plants	98.63%	
5.	Lu Zhang, Yili Zheng, Gangliang Zhong, Qiang Wang	Flavia ICL	94.22% 87.82%	<b>94.71%</b>
6.	Satti R. G. Reddy, G. P. Saradhi Varma, Rajya Lakshmi Davuluri	Leaf Snap, UCI Leaf, PlantVillage, Flavia, Swedish	100%, 100%, 100%, 89.99%, 97.99%	99.33% <b>94.71%</b> <b>98.44%</b>
7.	Jagadeesh Basavaiah, Audre Arlene Anthony	Tomato	94% 90%	<b>99.27%</b>
8.	CHANGJIAN ZHOU, SIHAN ZHOU, JINGE XING, JIA SONG	AI CHALLENGER (Tomato)	95%	
9.	Alain B. Djimeli-Tsajio, Noulamo Thierry, Lienou T. Jean-Pierre, T. F. Kapche, P. Nagabhushan3	Tomato	98.90%	<b>99.27%</b>
10.	Rajasekaran Thangaraj, S. Anandamurugan, Vishnu Kumar Kaliappan	PlantVillage (Tomato)	99.55%	99.27%
11.	Poornima Singh Thakur, Tanuja Sheorey, Aparajita Ojha	PlantVillage, Embrapa, Apple, Maize, Rice	99.16% 93.66% 94.24% 91.36% 96.67%	<b>99.33%</b> <b>99.75%</b> <b>98.02%</b>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.8 เปรียบเทียบค่าความถูกต้องของแบบจำลองที่นำเสนอกับผลงานวิจัยที่เกี่ยวข้อง (ต่อ)

	ผู้แต่ง	ชุดข้อมูล	ค่าความถูกต้อง	แบบจำลองที่นำเสนอ
12.	Adem Tuncer	PlantVillage	99.27%	<b>99.33%</b>
13.	Trong-Yen Lee, I-An Lin, Jui-Yuan Yu, Jing-min Yang, Yu-Chun Chang	PlantVillage (Potato)	99.53%	99.50%
14.	PENG JIANG, YUEHAN CHEN, BIN LIU, DONGJIAN HE, CHUNQUAN LIANG	Apple	97.14%	<b>99.75%</b>
15.	Reva Nagi, Sanjaya Shankar Tripathy	PlantVillage (Grapevine)	98.4%	<b>99.73%</b>
16.	Prabhjot Kaur, Shilpi Harnal, Vinay Gautam, Mukund Pratap Singh, Santar Pal Singh	PlantVillage (Tomato)	98.92%	<b>99.27%</b>

จากตารางที่ 4.8 แสดงค่าความถูกต้องของการจำแนกโรคของใบพืชด้วยแบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer ที่นำเสนอเปรียบเทียบกับผลงานวิจัยที่เกี่ยวข้องจากบทที่ 2 โดยทำการทดลองจำแนกโรคของใบพืชจากชุดข้อมูลเดียวกัน ผลการทดลองพบว่า วิธีการที่นำเสนอให้ค่าความถูกต้องของการจำแนกโรคใบพืชที่ดีกว่าในหลายๆ ชุดข้อมูล ยกเว้นชุดข้อมูล PlantVillage ของงานวิจัยลำดับที่ 6 ชุดข้อมูลใบมะเขือเทศของงานวิจัยลำดับที่ 10 และชุดข้อมูลใบมันฝรั่งของงานวิจัยลำดับที่ 13 เนื่องจากงานวิจัยดังกล่าวได้มีการประยุกต์ใช้ขั้นตอนการถ่ายโอนการเรียนรู้และการปรับจูนแบบจำลอง แต่แบบจำลองที่นำเสนอไม่ต้องมีขั้นตอนดังกล่าวก็สามารถให้ค่าความถูกต้องที่ใกล้เคียงกันทำให้สามารถลดเวลาการทำงานขั้นตอนนี้ไปได้ นอกจากนี้บางงานวิจัยมีการใช้ชั้น Global Average Pooling แทนชั้นการเชื่อมต่อสมบูรณ์ (Fully Connected Layer) ส่งผลให้ค่าความถูกต้องเพิ่มขึ้น ดังนั้นจึงได้ปรับปรุงแบบจำลองที่นำเสนอโดยการปรับใช้ชั้น Global Average Pooling แทนชั้นการเชื่อมต่อสมบูรณ์ (Fully Connected Layer) เพื่อเปรียบเทียบค่าความถูกต้องของการจำแนกโรคใบพืช แสดงดังตารางที่ 4.9

ตารางที่ 4.9 เปรียบเทียบค่าความถูกต้องของแบบจำลองที่นำเสนอกับการใช้ชั้น Global Average Pooling

	ผู้แต่ง	ชุดข้อมูล	ค่าความถูกต้อง	แบบจำลองที่นำเสนอ	Global Average Pooling
1.	Monika Bansal, Munish Kumar, Manish Kumar	Caltech-101	83.27%		
2.	Payal Chhabra, Naresh Kumar Garg, Munish Kumar	Wang database Corel database	86.20% 99.53%		
3.	Surbhi Gupta, Munish Kumar, Anupam Garg	Caltech-101	85.6%		
4.	T. Gayathri Devi, P. Neelamegam	rice plants	98.63%		
5.	Lu Zhang, Yili Zheng, Gangliang Zhong, Qiang Wang	Flavia ICL	94.22% 87.82%	<b>94.71%</b>	<b>94.49%</b>
6.	Satti R. G. Reddy, G. P. Saradhi Varma, Rajya Lakshmi Davuluri	Leaf Snap, UCI Leaf, PlantVillage, Flavia, Swedish	100%, 100%, 100%, 89.99%, 97.99%	99.33% <b>94.71%</b> <b>98.44%</b>	99.31% <b>94.49%</b> <b>97.53%</b>
7.	Jagadeesh Basavaiah, Audre Arlene Anthony	Tomato	94% 90%	<b>99.27%</b>	<b>99.43%</b>
8.	CHANGJIAN ZHOU, SIHAN ZHOU, JINGE XING, JIA SONG	AI CHALLENGER (Tomato)	95%		
9.	Alain B. Djimeli-Tsajio, Noulamo Thierry, Lienou T. Jean-Pierre, T. F. Kapche, P. Nagabhushan3	Tomato	98.9%	<b>99.27%</b>	<b>99.43%</b>
10.	Rajasekaran Thangaraj, S. Anandamurugan, Vishnu Kumar Kaliappan	PlantVillage (Tomato)	99.55%	99.27%	99.43%
11.	Poornima Singh Thakur, Tanuja Sheorey, Aparajita Ojha	PlantVillage, Embrapa, Apple, Maize, Rice	99.16% 93.66% 94.24% 91.36% 96.67%	<b>99.33%</b>  <b>99.75%</b> <b>98.02%</b>	<b>99.31%</b>  <b>99.76%</b> <b>99.56%</b>
12.	Adem Tuncer	PlantVillage	99.27%	<b>99.33%</b>	<b>99.31%</b>
13.	Trong-Yen Lee, I-An Lin, Jui-Yuan Yu, Jing-min Yang, Yu-Chun Chang	PlantVillage (Potato)	99.53%	99.50%	<b>100%</b>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.9 เปรียบเทียบค่าความถูกต้องของแบบจำลองที่นำเสนอกับการใช้ชั้น Global Average Pooling (ต่อ)

	ผู้แต่ง	ชุดข้อมูล	ค่าความถูกต้อง	แบบจำลองที่นำเสนอ	Global Average Pooling
14.	PENG JIANG, YUEHAN CHEN, BIN LIU, DONGJIAN HE, CHUNQUAN LIANG	Apple	97.14%	99.75%	99.76%
15.	Reva Nagi, Sanjaya Shankar Tripathy	PlantVillage (Grapevine)	98.4%	99.73%	95.86%
16.	Prabhjot Kaur, Shilpi Harnal, Vinay Gautam, Mukund Pratap Singh, Santar Pal Singh	PlantVillage (Tomato)	98.92%	99.27%	99.43%

จากตารางที่ 4.9 เปรียบเทียบค่าความถูกต้องของการจำแนกโรคใบพืชโดยใช้แบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer ที่นำเสนอเปรียบเทียบกับการใช้ชั้น Global Average Pooling แทนชั้นการเชื่อมต่อสมบูรณ์ (Fully Connected Layer) ผลการทดลองพบว่า การเปลี่ยนมาใช้ชั้น Global Average Pooling แทนชั้นการเชื่อมต่อสมบูรณ์ (Fully Connected Layer) ในแบบจำลองที่นำเสนอสามารถเพิ่มค่าความถูกต้องของการจำแนกข้อมูลเพิ่มขึ้นสำหรับชุดข้อมูลที่มีจำนวนรูปภาพมากพอสมควร แต่สำหรับชุดข้อมูลที่มีจำนวนรูปภาพน้อย เช่น ชุดข้อมูล Flavia และ ชุดข้อมูล Swedish ทำให้ค่าความถูกต้องลดลง และเมื่อเปรียบเทียบเวลาการทำงานพบว่าการใช้ชั้น Global Average Pooling ใช้เวลาการทำงานในแต่ละรอบเพิ่มขึ้น แสดงดังตารางที่ 4.10

ตารางที่ 4.10 เปรียบเทียบเวลาการทำงานแต่ละรอบของแบบจำลองที่นำเสนอกับการใช้ชั้น Global Average Pooling

	ชุดข้อมูล	จำนวนรูปภาพ	จำนวนคลาส	เวลาแบบจำลองที่นำเสนอ	เวลา Global Average Pooling
1.	Swedish	1,125	15	21	23
2.	Flavia	1,906	32	32	36
3.	Potato	2,500	3	54	54
4.	Apple	3,396	4	71.60	73
5.	Corn	3,852	4	85.60	82

ตารางที่ 4.10 เปรียบเทียบเวลาการทำงานแต่ละรอบของแบบจำลองที่นำเสนอกับการใช้ชั้น Global Average Pooling (ต่อ)

ชุดข้อมูล		จำนวน รูปภาพ	จำนวน คลาส	เวลา แบบจำลองที่นำเสนอ	เวลา Global Average Pooling
6.	Grape	4,139	4	87.20	92
7.	Tomato	18,159	10	386	412
8.	PlantVillage	54,304	38	386	513.80

จากตารางที่ 4.9 และ 4.10 เปรียบเทียบค่าความถูกต้องของการจำแนกโรคของใบพืชและเวลาการทำงานแต่ละรอบของแบบจำลองที่นำเสนอกับการใช้ชั้น Global Average Pooling แทนชั้นการเชื่อมต่อสมบูรณ์ (Fully Connected Layer) ในแบบจำลองที่นำเสนอ ผลการทดลองพบว่า การใช้ชั้น Global Average Pooling แทนชั้นการเชื่อมต่อสมบูรณ์ (Fully Connected Layer) ในแบบจำลองที่นำเสนอมีผลค่าความถูกต้องของการจำแนกข้อมูลที่ไม่แตกต่างกัน แต่ส่งผลให้การทำงานของแบบจำลองแต่ละรอบใช้เวลาเพิ่มขึ้นโดยเฉพาะชุดข้อมูลขนาดใหญ่ที่มีรูปภาพจำนวนมากและมีคลาสหลายคลาส จึงมีการทดสอบข้อสมมติฐานการวิจัยด้วยค่าสถิติ T – test โดยเปรียบเทียบกับค่าความถูกต้องเฉลี่ยจากการทดลองกับชุดข้อมูลต่างๆ ตามตารางที่ 4.9 ของแบบจำลองที่นำเสนอกับการใช้ชั้น Global Average Pooling โดยทดสอบที่ระดับความเชื่อมั่น 95% และกำหนดข้อสมมติฐานการวิจัยดังนี้

$H_0$ : การใช้ชั้น GAP ร่วมกับสถาปัตยกรรมที่นำเสนอไม่มีผลต่อการเพิ่มประสิทธิภาพของการจำแนกโรคของใบพืช

$H_1$ : การใช้ชั้น GAP ร่วมกับสถาปัตยกรรมที่นำเสนอมีผลต่อการเพิ่มประสิทธิภาพของการจำแนกโรคของใบพืช ที่ระดับความเชื่อมั่น 95%

$$S_p^2 = \frac{(n_A - 1)S_A^2 + (n_B - 1)S_B^2}{n_A + n_B - 2} \quad (4.1)$$

$$S_p^2 = \frac{7(2.84) + 7(4.28)}{14} \quad (4.2)$$

$$S_p^2 = \frac{19.88 + 29.96}{14} \quad (4.3)$$

$$S_p^2 = 3.56 \quad (4.4)$$

โดยที่  $n_A$  แทนด้วย จำนวนประชากรของแบบจำลองที่นำเสนอ

$n_B$  แทนด้วย จำนวนประชากรของชั้น GAP ร่วมกับแบบจำลองที่นำเสนอ

$S^2_A$  แทนด้วย ค่าความแปรปรวนของแบบจำลองที่นำเสนอ

$S^2_B$  แทนด้วย ค่าความแปรปรวนของชั้น GAP ร่วมกับแบบจำลองที่นำเสนอ

$$t = \frac{(\bar{X}_A - \bar{X}_B) - (\mu_A - \mu_B)}{\sqrt{S_p^2 \left( \frac{1}{n_A} + \frac{1}{n_B} \right)}} \quad (4.5)$$

$$t = \frac{98.56 - 98.24}{\sqrt{3.56 \left( \frac{2}{8} \right)}} \quad (4.6)$$

$$t = 0.34 < t_{0.05}(2.1448) \gg \text{Accept } H_0$$

จากการทดสอบข้อสมมติฐานการวิจัยดังกล่าวพบว่า ค่า  $t = 0.34$  ซึ่งมีค่าอยู่ระหว่างค่า  $t$  ที่ระดับความเชื่อมั่น 0.05 จึงทำให้ยอมรับสมมติฐาน  $H_0$  ซึ่งสามารถสรุปได้ว่า การใช้ Global Average Pooling แทนชั้นการเชื่อมต่อสมบูรณ์ (Fully Connected Layer) ในแบบจำลองที่นำเสนอไม่มีผลต่อการเพิ่มประสิทธิภาพของการจำแนกโรคใบพืชที่ระดับความเชื่อมั่น 95%

#### 4.3 เปรียบเทียบจำนวนพารามิเตอร์ของสถาปัตยกรรม VGG10 – InceptionV3 with Parameter Reduction Layer

แบบจำลองที่นำเสนอเป็นการปรับปรุงบนพื้นฐานของแบบจำลอง VGG16 โดยการปรับลดจำนวนชั้นของ VGG16 จากจำนวน 16 ชั้นเหลือ 10 ชั้น และปรับปรุง InceptionV3 with Parameter Reduction Layer โดยการเพิ่มจำนวนชั้นคอนโวลูชันจาก 3 ชั้น เป็น 4 ชั้น เพื่อให้สามารถกระจายการสกัดคุณสมบัติของรูปภาพแบบขนานได้ครบถ้วนมากขึ้น และปรับขนาดตัวกรองของชั้นคอนโวลูชันใน InceptionV3 จาก  $3 \times 3$  เป็นขนาด  $1 \times 3$  และ  $3 \times 1$  แทนเพื่อช่วยลดจำนวนพารามิเตอร์ของแบบจำลอง ซึ่งทำให้จำนวนพารามิเตอร์ของ InceptionV3 with Parameter Reduction Layer ลดลงแสดงดังตารางที่ 4.11 ซึ่งสามารถลดจำนวนพารามิเตอร์ลงได้ร้อยละ 41.58 จาก InceptionV3 Block เดิม

ตารางที่ 4.11 จำนวนพารามิเตอร์ของ InceptionV3 Block และ InceptionV3 with Parameter Reduction Layer

Base InceptionV3 Architecture			InceptionV3 with Parameter Reduction Layer		
Layers	Filter size	Parameters	Layers	Filter size	Parameters
Conv2D1_1	1x1, 32	4,128	Conv2D1_1	1x1, 32	4128
Conv2D1_2	3x3, 32	9,248	Conv2D1_2	1x3, 16	1552
Conv2D1_3	3x3, 32	9,248	Conv2D1_3	3x1, 16	784
Conv2D2_1	1x1, 32	4,128	Conv2D2_1	1x1, 32	4128
Conv2D2_2	3x3, 32	9,248	Conv2D2_2	1x3, 16	1552
MaxPooling	3x3	-	Conv2D2_3	3x1, 16	784
Conv2D2	1x1, 32	4,128	Conv2D3_1	1x1, 16	2064
Conv2D3	1x1, 32	4,128	Conv2D3_2	1x3, 32	1568
			Conv2D3_3	3x1, 32	3104
			MaxPooling	3x3	-
			Conv2D4	1x1, 16	2064
			Conv2D5	1x1, 32	4128
Total Parameters		44,256	Total Parameters		25,856

จากตารางที่ 4.11 แสดงจำนวนพารามิเตอร์ของแบบจำลอง InceptionV3 Block มีจำนวน 44,256 พารามิเตอร์ และเมื่อทำการปรับปรุง InceptionV3 with Parameter Reduction Layer โดยเพิ่มจำนวนชั้นคอนโวลูชันและปรับขนาดของตัวกรองในชั้นคอนโวลูชันทำให้เหลือพารามิเตอร์จำนวน 25,856 พารามิเตอร์ ซึ่งแบบจำลองที่นำเสนอสามารถลดจำนวนพารามิเตอร์ได้ถึงร้อยละ 41.58 และเมื่อนำ InceptionV3 with Parameter Reduction Layer ใช้ร่วมกับแบบจำลอง VGG16 ที่ปรับเหลือ 10 ชั้น ทำให้สามารถลดจำนวนพารามิเตอร์ของแบบจำลอง VGG16 และเมื่อใช้งานร่วมกับ InceptionV3 with Parameter Reduction Layer ทำให้พารามิเตอร์ลดลงเหลือประมาณ 1,886,708 พารามิเตอร์เท่านั้น แบบจำลองที่นำเสนอสามารถลดจำนวนพารามิเตอร์ได้ร้อยละ 87.20 จาก VGG16 และร้อยละ 91.38 จาก InceptionV3 ดังแสดงไว้ในตารางที่ 4.4, 4.7 และ 4.11 โดยมีจำนวนพารามิเตอร์น้อยกว่าแบบจำลอง MobileNetV2 ที่เป็นแบบจำลองสำหรับการใช้งานบนอุปกรณ์พกพาขนาดเล็ก ดังนั้นแบบจำลองที่นำเสนอสามารถนำไปพัฒนาใช้งานบนอุปกรณ์พกพาขนาดเล็กได้

## บทที่ 5

### สรุปผลการวิจัยและข้อเสนอแนะ

งานวิจัยนี้มีวัตถุประสงค์เพื่อ 1) เปรียบเทียบประสิทธิภาพการจำแนกใบพืชโดยใช้การกำหนดจุดเด่นบนขอบใบด้วย Scale-Invariant Feature Transform (SIFT) 2) เพื่อนำเสนอแบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer สำหรับจำแนกโรคของใบพืช และ 3) เพื่อเปรียบเทียบประสิทธิภาพ จำนวนพารามิเตอร์และเวลาในการประมวลผลของสถาปัตยกรรม VGG10 – InceptionV3 with Parameter Reduction Layer ที่นำเสนอกับสถาปัตยกรรมอื่น ๆ ของ CNN เช่น VGG16, InceptionV3, ResNet50, DenseNet121 และ MobileNetV2 ในส่วนนี้จะกล่าวถึงบทสรุปข้อจำกัดและข้อเสนอแนะของงานวิจัย รวมไปถึงแนวทางการพัฒนาต่อยอดงานวิจัยและการประยุกต์ใช้ผลงานวิจัย เพื่อเป็นแนวทางสำหรับการนำไปใช้ประโยชน์ในอนาคต

#### 5.1 สรุปผลการวิจัย

ปัจจุบันงานทางด้านคอมพิวเตอร์วิชันกำลังได้รับความนิยมอย่างแพร่หลายเนื่องจากเป็นการทำงานที่เน้นการดำเนินงานด้วยรูปภาพ เช่น การจำแนกรูปภาพ การจัดกลุ่มรูปภาพ การตรวจจับวัตถุจากรูปภาพ เป็นต้น และมีการนำคอมพิวเตอร์วิชันไปประยุกต์ใช้กับงานหลากหลายอาชีพมากขึ้น โดยงานด้านคอมพิวเตอร์วิชันนิยมใช้โครงข่ายประสาทเทียมแบบคอนโวลูชันเป็นแบบจำลอง เนื่องจากโครงข่ายประสาทเทียมแบบคอนโวลูชันเป็นโครงข่ายประสาทเทียมที่พัฒนาเพื่อใช้สำหรับการดำเนินงานที่เกี่ยวข้องกับรูปภาพโดยเฉพาะ แบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชันแบบเดิมส่วนใหญ่จะมีขนาดของโครงข่ายที่ค่อนข้างใหญ่เพราะต้องใช้พารามิเตอร์จำนวนมากจึงทำให้โครงข่ายมีขนาดใหญ่และต้องใช้เวลาการประมวลผลที่ค่อนข้างนาน และไม่สามารถใช้งานบนอุปกรณ์ที่มีข้อจำกัดด้านทรัพยากรและอุปกรณ์พกพาขนาดเล็กได้ ดังนั้นจึงได้มีนักวิจัยจำนวนมากพยายามศึกษาวิจัยและทดสอบแบบจำลองแบบต่างๆ เพื่อลดขนาดของโครงข่ายประสาทเทียมแบบคอนโวลูชันเพื่อให้โครงข่ายมีขนาดเล็กลงและสามารถใช้งานได้บนอุปกรณ์พกพาหรืออุปกรณ์ที่มีทรัพยากรจำกัด ดังเช่นโครงข่ายประสาทเทียมแบบ MobileNet ที่พัฒนาขึ้นเพื่อให้สามารถใช้แบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชันบนอุปกรณ์พกพาที่มีขนาดเล็กได้ ดังนั้นงานวิจัยนี้จึงมีแนวคิดเกี่ยวกับการพัฒนาแบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชันโดยปรับปรุงมาจากแบบจำลองโครงข่ายประสาทเทียมแบบ VGG16 ที่มีขนาดของโครงข่ายขนาดใหญ่ ใช้พารามิเตอร์ประมาณ 138.4 ล้านพารามิเตอร์ และปรับปรุงแบบจำลองของ InceptionV3 with Parameter Reduction Layer ให้แบบจำลองมีขนาดกว้างขึ้นซึ่งสามารถช่วยลดเวลาการประมวลผลของแบบจำลองได้ จากนั้นประยุกต์ใช้งานร่วมกับของแบบจำลอง VGG16 ที่ปรับ

ลดจำนวนชั้นเหลือเพียง 10 ชั้น และ InceptionV3 with Parameter Reduction Layer เพื่อลดจำนวนพารามิเตอร์ของแบบจำลอง ลดเวลาการประมวลผล แต่ยังคงมีประสิทธิภาพด้านการจำแนกข้อมูลที่สูง โดยใช้ชื่อแบบจำลองคือ VGG10 – InceptionV3 with Parameter Reduction Layer จากผลการวิจัยสามารถสรุปได้ดังนี้

5.1.1 การจำแนกใบพืชโดยใช้การประมวลผลภาพด้วยการเปลี่ยนแปลงลักษณะทางสัญญาณวิทยาของรูปภาพก่อนใช้การตรวจจับเส้นขอบด้วยแคนนี่ และกำหนดจุดเด่นบนเส้นขอบของใบพืชด้วยวิธีด้วย Scale-Invariant Feature Transform (SIFT) สกัดคุณสมบัติด้วย Sequential Model ของโครงข่ายประสาทเทียมแบบคอนโวลูชันและจำแนกชนิดใบพืชด้วยแบบจำลองป่าสุ่ม (Random Forest) ผลการวิจัยพบว่าวิธีการที่นำเสนอสามารถช่วยให้แบบจำลองสามารถสกัดคุณสมบัติของใบพืชได้ครบถ้วนมากขึ้น และมีค่าความถูกต้องของการจำแนกชนิดของใบพืชที่สูงกว่าการใช้คุณสมบัติรูปร่าง คุณสมบัติพื้นผิว และการใช้คุณสมบัติรูปร่างร่วมกับคุณสมบัติพื้นผิว เมื่อเปรียบเทียบวิธีการที่นำเสนอกับการใช้แบบจำลองโครงข่ายประสาทเทียมแบบอื่นๆ พบว่าวิธีการที่นำเสนอยังคงให้ค่าความถูกต้องอยู่ในระดับดี

5.1.2 การปรับปรุงแบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer ที่มีพื้นฐานมาจากแบบจำลองโครงข่ายประสาทเทียมแบบ VGG16 และ InceptionV3 Block โดยการปรับปรุง InceptionV3 with Parameter Reduction Layer โดยเพิ่มจำนวนชั้นคอนโวลูชันของ InceptionV3 Block จาก 3 ชั้นเป็น 4 ชั้น เพื่อให้แบบจำลองมีความกว้างมากขึ้นและปรับขนาดของตัวกรองจาก  $3 \times 3$  เป็น  $1 \times 3$  และ  $3 \times 1$  แทน ซึ่งข้อดีของ InceptionV3 with Parameter Reduction Layer ที่นำเสนอทำให้แบบจำลองสามารถสกัดคุณสมบัติของรูปภาพแบบขนานพร้อมกันทำให้ได้คุณสมบัติของรูปภาพที่ครบถ้วนมากขึ้น และการปรับขนาดของตัวกรองทำให้แบบจำลองสามารถลดจำนวนพารามิเตอร์และเวลาการทำงานลงได้ จากนั้นนำ InceptionV3 with Parameter Reduction Layer ใช้ร่วมกับแบบจำลอง VGG16 และลดจำนวนชั้นของแบบจำลอง VGG16 จาก 16 ชั้นเหลือเพียง 10 ชั้น ทดลองแบบจำลองเบื้องต้นโดยการจำแนกโรคของใบมะเขือเทศ ผลการวิจัยพบว่าแบบจำลองที่นำเสนอให้ค่าความถูกต้องของการจำแนกข้อมูลใบพืชสูงถึงร้อยละ 99.27 สามารถลดจำนวนพารามิเตอร์และลดเวลาการประมวลผลลงได้ เมื่อเปรียบเทียบวิธีการที่นำเสนอกับแบบจำลองอื่นๆ ของโครงข่ายประสาทเทียมแบบคอนโวลูชัน พบว่า วิธีการที่นำเสนอให้ค่าความถูกต้องสูงที่สุดและใช้จำนวนพารามิเตอร์น้อยที่สุด ถึงแม้ว่าเวลาที่ใช้สำหรับการประมวลผลยังคงใช้เวลามากกว่าแบบจำลอง MobileNetV2 และ InceptionV3 แต่เมื่อพิจารณาจากค่าความถูกต้อง จำนวนพารามิเตอร์ และค่าฟังก์ชันการสูญเสีย แบบจำลองที่นำเสนอยังคงมีประสิทธิภาพโดยรวมที่ดีกว่า เมื่อประเมินประสิทธิภาพของแบบจำลองโดยการจำแนกชนิดของใบพืชที่มีความหลากหลายมากขึ้นและเป็นชุดข้อมูลรูปภาพขนาดใหญ่จาก PlantVillage Dataset ที่มีจำนวนรูปภาพทั้งหมด 54,304 รูป แบ่งออกเป็น 38 คลาส

ผลการวิจัยพบว่า แบบจำลองที่นำเสนอให้ค่าความถูกต้องของการจำแนกใบพืชเฉลี่ยเท่ากับร้อยละ 99.33 ซึ่งมีค่าความถูกต้องเพิ่มขึ้นจากการทดลองกับการจำแนกโรคของใบมะเขือเทศ และเมื่อเปรียบเทียบกับแบบจำลองอื่นๆ ของโครงข่ายประสาทเทียมแบบคอนโวลูชัน พบว่าวิธีการที่นำเสนอใช้จำนวนพารามิเตอร์น้อยที่สุด สามารถลดจำนวนพารามิเตอร์ได้ร้อยละ 87.20 จาก VGG16 และร้อยละ 91.38 จาก InceptionV3 ซึ่งเป็นแบบจำลองพื้นฐานของการนำเสนอแบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer และแม้ว่าแบบจำลองที่นำเสนอมีค่าความถูกต้องของการจำแนกข้อมูลสูงที่สุด ใช้จำนวนพารามิเตอร์น้อยที่สุด แต่ยังคงใช้เวลาการทำงานมากกว่าแบบจำลอง MobileNetV2 และค่าฟังก์ชันการสูญเสียยังคงมากกว่าแบบจำลอง MobileNetV2 เนื่องจากแบบจำลองที่นำเสนอใช้สถาปัตยกรรมพื้นฐานของแบบจำลอง VGG16 ที่มีการทำงานแบบลำดับขั้น จึงส่งผลให้แบบจำลองใช้เวลานานกว่า ในขณะที่แบบจำลอง MobileNetV2 มีการทำงานแบบขนาน นอกจากนี้ยังพบว่าแบบจำลองที่นำเสนอเมื่อนำไปใช้กับชุดข้อมูลรูปภาพขนาดใหญ่และมีจำนวนพืชที่หลากหลายส่งผลให้แบบจำลองใช้เวลาการประมวลผลที่นานขึ้น จึงอาจเป็นแนวทางการพัฒนางานวิจัยในอนาคตที่จะพัฒนาแบบจำลองให้สามารถลดเวลาการทำงานลงแต่ยังคงให้ค่าความถูกต้องของการจำแนกโรคของใบพืชที่สูง

## 5.2 ข้อจำกัดและข้อเสนอแนะของงานวิจัย

จากการดำเนินงานวิจัยเกี่ยวกับการจำแนกของใบพืชโดยใช้แบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer ที่พัฒนาขึ้น พบว่ามีข้อจำกัด 2 ประเด็นหลัก ดังนี้

5.2.1 ข้อจำกัดของขั้นตอนการกำหนดจุดเด่นบนเส้นขอบใบพืชด้วย SIFT ซึ่งเป็นการดำเนินงานตามข้อสมมติฐานการวิจัยที่ 1 พบว่าวิธีการที่นำเสนอสามารถใช้งานได้ดีกับชุดข้อมูลใบพืชที่มีลักษณะของใบที่แตกต่างกันชัดเจน เนื่องจากเป็นการกำหนดจุดเด่นบนเส้นขอบที่เป็นรูปร่างของใบพืช ซึ่งโรคบางโรคอาจจะไม่ได้ทำให้รูปร่างของใบพืชเปลี่ยนแปลงแต่มีสีที่เปลี่ยนแปลงไป ดังนั้นในอนาคตสามารถแก้ไขข้อจำกัดได้โดยการใช้คุณสมบัติสีร่วมด้วย

5.2.2 ข้อจำกัดของแบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer จากการดำเนินงาน พบว่าเมื่อใช้แบบจำลองที่นำเสนอจำแนกโรคของใบพืชที่มีจำนวนรูปภาพและคลาสที่ไม่เยอะเกินไปแบบจำลองสามารถทำงานได้ดี มีค่าความถูกต้องสูงและใช้เวลาการประมวลผลน้อย แต่เมื่อใช้แบบจำลองจำแนกข้อมูลใบพืชที่มีจำนวนหลายโรคและเป็นชุดข้อมูลขนาดใหญ่ แบบจำลองจะใช้เวลาการประมวลผลที่ค่อนข้างนานกว่าแบบจำลองอื่น ดังนั้นในอนาคตจึงเป็นแนวทางในการปรับปรุงแบบจำลองให้สามารถลดเวลาการประมวลผลของแบบจำลองลง

### 5.3 แนวทางการประยุกต์ใช้ผลงานวิจัย

แบบจำลอง VGG10 – InceptionV3 with Parameter Reduction Layer เป็นแบบจำลองที่พัฒนาขึ้นโดยมีวัตถุประสงค์เพื่อเพิ่มค่าความถูกต้องของการจำแนกข้อมูลรูปภาพและลดจำนวนพารามิเตอร์และเวลาการประมวลผลของแบบจำลอง ซึ่งจากการดำเนินงานพบว่าแบบจำลองที่นำเสนอสามารถให้ค่าความถูกต้องของการจำแนกข้อมูลสูง ใช้พารามิเตอร์จำนวนและพื้นที่หน่วยความจำน้อย และลดเวลาการประมวลผลได้ จึงสามารถนำแบบจำลองที่นำเสนอไปประยุกต์ใช้งานบนอุปกรณ์พกพาหรืออุปกรณ์ขนาดเล็กได้



## เอกสารอ้างอิง

- [1] Che, H. Brown, L.G. Foran, D.J. Noshier, J.L. and Hacihaliloglu, I. 2021. “Liver Disease Classification from Ultrasound using Multi-Scale CNN.”, *International Journal of Computer Assisted Radiology and Surgery* 16: 1537–1548.
- [2] Deepak, S. and Ameer, P.M. 2021. “Automated Categorization of Brain Tumor from MRI using CNN Features and SVM.”, *Journal of Ambient Intelligence and Humanized Computing* 12: 8357–8369.
- [3] Ibraheem, A.M. Rahouma, K.H. and Hamed, H.F.A. 2021. “3PCNNB-Net: Three Parallel CNN Branches for Breast Cancer Classification through Histopathological Images.”, *Journal of Medical and Biological Engineering* 41: 494–503.
- [4] Sivanandan, R. and Jayakumari, J. 2021. “A New CNN Architecture for Efficient Classification of Ultrasound Breast Tumor Images with Activation Map Clustering Based Prediction Validation.”, *Medical & Biological Engineering & Computing* 59: 957–968.
- [5] Barbhuiya, A.A. Karsh, R.K. and Jain, R. 2021. “CNN Based Feature Extraction and Classification for Sign Language.”, *Multimedia Tools and Applications* 80: 3051–3069.
- [6] Zatarain Cabada, R. Rodriguez Rangel, H. Barron Estrada, M.L. and Cardenas Lopez, H.M. 2020. “Hyperparameter Optimization in CNN for Learning-Centered Emotion Recognition for Intelligent Tutoring Systems.”, *Soft Computing* 24: 7593–7602.
- [7] He, J., Liu, T., Li, L. Hu, Y. and Zhou, G. 2023. “MFaster R-CNN for Maize Leaf Diseases Detection Based on Machine Vision.”, *Arabian Journal for Science and Engineering* 48: 1437–1449.
- [8] Mustafa, H., Umer, M., Hafeez, U. Hameed, A. Sohaib, A. Ullah, S. and Madni, H.A. 2022. “Pepper Bell Leaf Disease Detection and Classification using Optimized

- Convolutional Neural Network.”, *Multimedia Tools and Applications*.  
<https://doi.org/10.1007/s11042-022-13737-8>
- [9] Prabu, M. and Chelliah, B.J. 2022. “Mango Leaf Disease Identification and Classification using a CNN Architecture Optimized by Crossover-Based Levy Flight Distribution Algorithm”, *Neural Computing and Applications* 34: 7311–7324.
- [10] Yakkundimath, R., Saunshi, G., Anami, B. and Palaiah, S. 2022. “Classification of Rice Diseases using Convolutional Neural Network Models.”, *J. Inst. Eng. India Ser. B* 103: 1047–1059.
- [11] Jiang, P. Chen, Y. Liu, B. He, D. and Liang, C. 2019. “Real-Time Detection of Apple Leaf Diseases using Deep Learning Approach Based on Improved Convolutional Neural Networks.”, *IEEE Access*, vol. 7: 59069-59080.
- [12] Liu, J. and Wang, X. 2020. “Early Recognition of Tomato Gray Leaf Spot Disease Based on MobileNetv2-YOLOv3 Model.”, *Plant Methods* 16:83.
- [13] Wang, X. Liu, J. and Zhu, X. 2021. “Early Real-Time Detection Algorithm of Tomato Diseases and Pests in the Natural Environment”, *Plant Methods* 17:43.
- [14] Shelke, A. and Mehendale, N. 2022. “A CNN-Based Android Application for Plant Leaf Classification at Remote Locations.”, *Neural Comput & Applic* 35: 2601–2607.
- [15] Abdelmalek, B. Ahmed, K. and Amine, T.M. n.d. “A Survey on Lightweight CNN-Based Object Detection Algorithms for Platforms with Limited Computational Resources.”, *International Journal of Informatics and Applied Mathematics*, Vol. 2, No. 2: 28-44.
- [16] Lecun, Y. Bottou, L. Bengio, Y. and Hader, P. 1998. “Gradient-Based Learning Applied to Document Recognition.”, *PROC OF THE IEEE*: 1 – 46.
- [17] Simonyan, K. and Zisserman, A. 2015. “Very Deep Convolutional Neural Networks for Large-Scale Image Recognition.”, *arXiv:1409.1556v6*.
- [18] Szegedy, C. Vanhoucke, V. Ioffe, S. Shlens, J. and Wojna, Z. 2015. “Rethinking the Inception architecture for Computer Vision.”, *arXiv:1512.00567v3*.

- [19] Howard, A.G. Zhu, M. Chen, B. and Kalenichenko, D. 2017. “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.” *arXiv:1704.04861v1*
- [20] Sandler, M. Howard, A. Zhu, M. Zhmoginov, A. and Chen, L.C. 2019. “MobileNetV2: Inverted Residuals and Linear Bottlenecks.”, *arXiv:1801.04381v4*.
- [21] Huang, G. Liu, Z. Maaten, L. and Weinberger, K.Q. 2018. “Densely Connected Convolutional Networks.”, *arXiv:1608.06993v5*.
- [22] He, K. Zhang, X. Ren, S. and Sun, J. 2015. “Deep Residual Learning for Image Recognition.”, *arXiv:1512.03385v1*.
- [23] Goyal, N. Gupta, K. Kumar, N. 2019. “Multiclass Twin Support Vector Machine for Plant Species Identification.”, *Multimedia Tools and Applications* 78:27785–27808.
- [24] Lowe, D.G. 2004. “Distinctive Image Features from Scale-Invariant Keypoints.” *International Journal of Computer Vision*: 1 – 28.
- [25] Basavaiah, J. and Anthony, A.A. 2020. “Tomato Leaf Disease Classification using Multiple Feature Extraction Techniques.”, *Wireless Personal Communications* 115: 633–651.
- [26] Bansal, M. Kumar, M. and Kumar, M. 2021. “2D Object Recognition: A Comparative Analysis of SIFT, SURF and ORB Feature Descriptors.”, *Multimedia Tools and Applications*, vol. 80: 18839–18857.
- [27] Chhabra, P. Garg, N.K. and Kumar, M. 2020. “Content-Based Image Retrieval System using ORB and SIFT Features.”, *Neural Computing and Applications*, vol. 32: 2725–2733.
- [28] Gupta, S. Kumar, M. and Garg, A. 2019. “Improved Object Recognition Results using SIFT and ORB Feature Detector.”, *Multimedia Tools and Applications*, vol. 78: 34157–34171.
- [29] Devi, T.G. and Neelamegam, P. 2019. “Image Processing Based Rice Plant Leaves Diseases in Thanjavur Tamilnadu.”, *Cluster Computing*, vol. 22: 13415–13428.

- [30] Zhang, L. Zheng, Y. Zhong, G. and Wang, Q. 2019. “Research on Leaf Species Identification Based on Principal Component and Linear Discriminant Analysis.”, *Cluster Computing*, vol. 22: 7795–7804.
- [31] Reddy, S.R.G. Varma, G.P.S. and Davuluri, R.L. 2021. “Optimized Convolutional Neural Network Model for Plant Species Identification from Leaf Images using Computer Vision.”, *International Journal of Speech Technology*: 1-28.
- [32] Basavaiah, J. and Anthony, A.A. 2020. “Tomato Leaf Disease Classification using Multiple Feature Extraction Techniques.”, *Wireless Personal Communications 115*: 633–651.
- [33] Zhou, C. Zhou, S. Xing, J. and Song, J. 2019. “Tomato Leaf Disease Identification by Restructured Deep Residual Dense Network.”, *IEEE Access*, vol. 9: 28822-28831.
- [34] Djimeli-Tsajio, A.B. Thierry, N. Jean-Pierre, L.T. Kapche, T.F. and Nagabhushan, P. 2022. “Improved Detection and Identification Approach in Tomato Leaf Disease using Transformation and Combination of Transfer Learning Features.”, *Journal of Plant Diseases and Protection 129*: 665–674.
- [35] Thangaraj, R. Anandamurugan, S. and Kaliappan, V.K. 2021. “Automated Tomato Leaf Disease Classification using Transfer Learning-Based Deep Convolution Neural Network.”, *Journal of Plant Diseases and Protection 128*: 73–86.
- [36] Thakur, P.S. Sheorey, T. and Ojha, A. 2022. “VGG-ICNN: A Lightweight CNN Model for Crop Disease Identification.”, *Multimedia Tools and Applications*.  
<https://doi.org/10.1007/s11042-022-13144-z>
- [37] Tuncer, A. 2021. “Cost-Optimized Hybrid Convolutional Neural Networks for Detection of Plant Leaf Diseases.”, *Journal of Ambient Intelligence and Humanized Computing 12*: 8625–8636.
- [38] Lee, T.Y. Lin, I.A. Yu, J.Y. Yang, J. and Chang, Y.C. 2021. “High Efficiency Disease Detection for Potato Leaf with Convolutional Neural Network.”, *SN Computer Science 2*:297. <https://doi.org/10.1007/s42979-021-00691-9>

- [39] Jiang, P. Chen, Y. Liu, B. He, D. and Liang, C. 2019. “Real-Time Detection of Apple Leaf Diseases using Deep Learning Approach Based on Improved Convolutional Neural Networks.”, *IEEE Access*, vol. 7: 59069-59080.
- [40] Nagi, R. and Tripathy, S.S. 2022. “Deep Convolutional Neural Network Based Disease Identification in Grapevine Leaf Images.”, *Multimedia Tools and Applications 81*: 24995–25006.
- [41] Kaur, P. Harnal, S. Gautam, V. Singh, M.P. and Singh, S.P. 2022. “A Novel Transfer Deep Learning Method for Detection and Classification of Plant Leaf Disease.”, *Journal of Ambient Intelligence and Humanized Computing*. <https://doi.org/10.1007/s12652-022-04331-9>
- [42] Moussafr, M. Chaibi, H. Saadane, R. Chehri, A. Rharras, A.E. and Jeon, G. 2022. “Design of Efficient Techniques for Tomato Leaf Disease Detection using Genetic Algorithm-Based and Deep Neural Networks.”, *Plant Soil*. <https://doi.org/10.1007/s11104-022-05513-2>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ก

### รายชื่อผลงานวิจัยที่ได้ตีพิมพ์

1. Jiraporn Thomkaew and Sarun Intakosum, 2022. "Improvement Classification Approach in Tomato Leaf Disease using Modified Visual Geometry Group (VGG)-InceptionV3 ", *International Journal of Advanced Computer Science and Applications(IJACSA)*, 13(12): 362 - 370. <http://dx.doi.org/10.14569/IJACSA.2022.0131244>
2. Jiraporn Thomkaew and Sarun Intakosum, 2023. "Plant Species Classification Using Leaf Edge Feature Combination with Morphological Transformations and SIFT Key Point", *Journal of Image and Graphics*, Vol. 11, No. 1: 91-97.

## ประวัติผู้เขียน

ชื่อ	นางสาวจิราภรณ์ ถมแก้ว
วัน เดือน ปีเกิด	1 มิถุนายน 2529
ที่อยู่ปัจจุบัน	41/2 หมู่ที่ 3 ตำบลถ้ำใหญ่ อำเภอทุ่งสง จังหวัดนครศรีธรรมราช 80110
ประวัติการศึกษา	(2551) บริหารธุรกิจบัณฑิต สาขาระบบสารสนเทศทางคอมพิวเตอร์ เกรดเฉลี่ย 3.84 มหาวิทยาลัยเทคโนโลยีราชมงคลศรีวิชัย (2554) วิทยาศาสตร์มหาบัณฑิต สาขาวิทยาการคอมพิวเตอร์ เกรดเฉลี่ย 3.37 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ประสบการณ์ทำงาน	2555 - ปัจจุบัน อาจารย์ สังกัดคณะเทคโนโลยีการจัดการ มหาวิทยาลัยเทคโนโลยีราชมงคลศรีวิชัย วิทยาเขตนครศรีธรรมราช (ไสใหญ่)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้