การจำแนกสายพันธุ์ถั่วพิสตาชิโอโดยใช้เทคนิคการเรียนรู้เชิงลึก

ณิชารีย์ ศรีบูจันดี[1*], เรืองศักดิ์ ตระกูลพุทธิรักษ์[2]

บทคัดย่อ

ปัจจุบันการนำเข้าถั่วพิสตาชิโอในไทย ซึ่งมูลค่าของถั่วพิสตาชิโอนั้นจะแตกต่างกันตามแต่ละสายพันธุ์ จึงจำเป็นต้องมีวิธีการและเทคโนโลยีใหม่ๆ ในการจำแนกภาพสายพันธุ์พิสตาชิโอต่างๆ มูลค่าของถั่วพิสตาชิโอแต่ละสายพันธุ์ไม่เท่ากัน ขึ้นอยู่กับคุณภาพ ลักษณะทางกายภาพ และความนิยมในตลาด [1] โดยสายพันธุ์ที่มีคุณภาพสูง จะมีความต้องการสูงในตลาด การพัฒนาวิธีการและเทคโนโลยีในการจำแนกสายพันธุ์จึงมีความสำคัญ เพื่อรักษามาตรฐานคุณภาพและมูลค่าของผลิตภัณฑ์ งานวิจัยนี้มีวัตถุประสงค์เพื่อการศึกษาการจำแนกสายพันธุ์ของถั่วพิสตาชิโอตามลักษณะทางกายภาพ โดยใช้ข้อมูลภาพสายพันธุ์ Kirmizi และ Siirt จากแหล่งข้อมูลสาธารณะ [2] และเก็บข้อมูลภาพสายพันธุ์ Kerman เพิ่มเติม ซึ่งแบ่งออกเป็นข้อมูลภาพ Kirmizi จำนวน 1,232 รูป และ Siirt จำนวน 916 รูป และ Kerman จำนวน 1,056 รูป ในงานวิจัยนี้ศึกษาแบบจำลองการเรียนรู้เชิงลึกทั้งหมด 5 ประเภท ได้แก่ แบบจำลองที่มีการเรียนรู้ตั้งแต่เริ่มต้น (CNN from Scratch), แบบจำลองที่ได้รับการเรียนรู้ข้อมูลมาก่อน (Pre-Trained Model) VGG16, VGG19, ResNet50 และ EfficientNetB0 โดยใช้ถ่ายทอดจากข้อมูลภาพ ImageNet รวมถึงการปรับแต่งแบบจำลอง (Fine Tuning) และปรับค่าพารามิเตอร์ต่างๆ (Hyperparameters) เพื่อให้ได้แบบจำลองที่มีประสิทธิภาพสูงที่สุด จากผลการศึกษาพบว่าแบบจำลอง ResNet50 เป็นแบบจำลองที่มีประสิทธิภาพสูงสุด โดยมีค่าความถูกต้องอยู่ที่ 0.9812 รองมาเป็นแบบจำลอง VGG16 VGG19 EfficientNetB0 และ CNN from Scratch ตามลำดับ ซึ่งมีค่าความถูกต้อง (Accuracy) อยู่ที่ 0.9781, 0.9750, 0.9625 และ 0.9219 ตามลำดับ สำหรับการวิจัยในอนาคต มีข้อเสนอในการเพิ่มจำนวนข้อมูลภาพและการปรับแต่งแบบจำลอง (Tuning Hyperparameters) โดยใช้เทคนิคต่างๆ เช่น Grid Search หรือ Bayesian Optimization เพื่อค้นหาค่าพารามิเตอร์ที่เหมาะสมที่สุดสำหรับแบบจำลอง

คำสำคัญ: ถั่วพิสตาชิโอ, การจำแนกภาพ, การเรียนรู้เชิงลึก, แบบจำลองที่ได้รับการเรียนรู้ข้อมูลมาก่อน

---

[1] หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิทยาการข้อมูล คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ กรุงเทพฯ 10110

[2] คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ กรุงเทพฯ 10110

[*] Corresponding author:Tel.:082-2939755 E-mail address: nicharee.sri@g.swu.ac.th

Classification of Pistachio Species using Deep Learning Techniques

Nicharee Sribuchandee[1*], Ruangsak Trakunphutthirak[2]

Abstract

Nowadays, the consistent and growing trend in the consumption of pistachios necessitates the development of new methods and technologies for image classification of different pistachio varieties. The value of each pistachio variety varies, depending on the quality, physical characteristics, and market popularity. [1]. Pistachios, particularly certain varieties, are considered high-quality and expensive due to their high demand in the market. The development of methods and technologies for species classification is crucial in maintaining product quality standards and value. The objective of this research is to develop a deep learning model for classifying pistachio cultivars based on physical characteristics such as shell color, size, and shape, using image data of the 1,232 images of the Kirmizi and 916 images of the Siirt from public sources [2], along with additional image data of the 1,056 images Kerman which were collected by the researcher. Five types of models were employed: a basic Convolutional Neural Network (CNN) model trained from scratch, and four pre-trained models - VGG16, VGG19, ResNet50, and EfficientNetB0, each transferring weights from ImageNet. Fine-tuning and hyperparameter adjustments were made to achieve optimal efficiency. The results indicated that the ResNet50 model had the highest efficiency, with an accuracy of 0.9812, followed by the VGG16, VGG19, EfficientNetB0, and CNN from Scratch models, in that order, with accuracies of 0.9781, 0.9750, 0.9625, and 0.9219, respectively. For future research, there are suggestions to increase the number of image data and to fine-tune the models (Hyperparameter Tuning) using various techniques such as Grid Search or Bayesian Optimization to find the most suitable parameters for the model.

Keywords: Pistachio, Image Classification, Deep Learning, Pre-Trained Model

---

[1] Data Science, Faculty of Science, Srinakharinwirot University, Bangkok, Thailand, 10110

[2] Faculty of Science, Srinakharinwirot University, Bangkok, Thailand, 10110

[*] Corresponding author:Tel.:082-2939755 E-mail address: nicharee.sri@g.swu.ac.th

บทนำ

ถั่วพิสตาชิโอเป็นพืชที่มีราคาสูง เป็นที่นิยมมากซึ่งในประเทศไทยนั้น แต่ในไทยนั้นยังไม่สามารถปลูกพืชชนิดนี้ได้ ทำให้ต้องมีการนำเข้าพืชชนิดนี้เพียงอย่างเดียว มูลค่าของถั่วพิสตาชิโอนั้นจะแตกต่างกันตามแต่ละสายพันธุ์ มีความสำคัญอย่างยิ่งต่อมูลค่าทางเศรษฐกิจของถั่วพิสตาชิโอ ซึ่งวิธีการจำแนกสายพันธุ์แบบดั้งเดิมยังมีความผิดพลาดเนื่องจากต้องอาศัยแรงงานมนุษย์ งานวิจัยนี้จึงมุ่งหวังที่จะใช้ระบบประมวลผลอัตโนมัติด้วยการเรียนรู้เชิงลึก (Deep Learning) โดยเป็นแบบจำลองที่ได้รับการเรียนรู้ข้อมูลมาก่อน (Pre-Trained Model) โดยใช้วิธีการถ่ายโอนการเรียนรู้ (Transfer Learning) เพื่อจำแนกและวิเคราะห์สายพันธุ์ของถั่วพิสตาชิโอ 3 สายพันธุ์ ได้แก่ Kirmizi, Siirt และ Kerman โดยจำแนกตามลักษณะทางภาพ เช่น สีของเปลือก ขนาด และรูปร่าง ได้อย่างรวดเร็วและมีประสิทธิภาพ การนำเทคโนโลยีการเรียนรู้ของเครื่อง (Machine Learning) มาใช้ในการจำแนกสายพันธุ์ถั่วพิสตาชิโอจึงมีแนวโน้มที่จะได้รับความนิยมเพิ่มขึ้น เนื่องจากสามารถเรียนรู้จากข้อมูลจำนวนมากและสามารถระบุรูปแบบที่ซับซ้อนได้ ซึ่งจะช่วยเพิ่มประสิทธิภาพและลดข้อผิดพลาดในการจำแนกสายพันธุ์ถั่วพิสตาชิโอได้

งานวิจัยนี้ศึกษาและนำเสนอเครื่องมือที่ช่วยในการจำแนกสายพันธุ์ถั่วพิสตาชิโอ ตามลักษณะทางกายภาพ ด้วยการจำแนกภาพ (Image Classification) โดยการใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN) เพื่อสร้างแบบจำลองด้วยโครงสร้างต่างๆ แล้วนำมาเปรียบเทียบประสิทธิภาพ รวมถึงการปรับค่าพารามิเตอร์ต่างๆ (Hyperparameters) เพื่อให้ได้แบบจำลองที่มีความแม่นยำและมีประสิทธิภาพมากขึ้น

งานวิจัยที่เกี่ยวข้อง

1. งานวิจัยเรื่อง Classification and Analysis of Pistachio Species with Pre-Trained Deep Learning Models [2]

งานวิจัยนี้ศึกษาการจำแนกประเภทภาพของเมล็ดพิสตาชิโอ 2 สายพันธุ์คือ Kirmizi และ Siirt โดยใช้แบบจำลองการเรียนรู้เชิงลึกที่ได้รับการเรียนรู้ข้อมูลมาก่อน (Pre-trained Deep Learning Models) ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN) ทั้งหมด 3 ประเภท ได้แก่ AlexNet, VGG16 และ VGG19 มีชุดข้อมูลทั้งหมด 2,148 รูป ซึ่งแบ่งออกเป็นสายพันธุ์ Kirmizi จำนวน 2,132 รูป และสายพันธุ์ Siirt จำนวน 916 รูป

ตารางที่ 1 แสดงค่า Performance Metrics ของแบบจำลองต่างๆ [2]

Performance Metrics	AlexNet	VGG16	VGG19
ค่าความถูกต้อง (Accuracy)	0.9442	0.9884	0.9814
ค่าความแม่นยำ (Precision)	0.9150	0.9828	0.9742
ค่า F-1 Score	0.9496	0.9884	0.9820
ค่าความไว (Sensitivity)	0.9869	0.9956	0.9913
ค่าความจำเพาะ(Specificity)	0.8955	0.9801	0.9701

จากตารางที่ 1 การประเมินประสิทธิภาพของแบบจำลองต่างๆ จะพบว่าแบบจำลองที่ใช้โครงสร้างแบบ AlexNet ใช้เวลาในการฝึกข้อมูลน้อยที่สุด เนื่องจากเป็นแบบจำลองที่มีความซับซ้อนน้อยที่สุด และแบบจำลองที่ใช้โครงสร้างแบบ VGG16 ได้ค่าความถูกต้อง (Accuracy) สูงที่สุดอยู่ที่ 0.9884 ตามด้วยแบบจำลองที่ใช้โครงสร้างแบบ VGG19 และแบบจำลองที่ใช้โครงสร้างแบบ AlexNet ตามลำดับ

2. งานวิจัยเรื่อง Transfer Learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images [3]

งานวิจัยนี้ใช้การเรียนรู้เชิงลึกที่ได้รับการเรียนรู้ข้อมูลมาก่อน (Pre-trained Deep Learning Models) โดยใช้แบบจำลอง VGG16 และ Transfer Learning จากชุดข้อมูล ImageNet เพื่อสร้างแบบจำลองที่จำแนกภาพได้เอง โดยการทำ Feature Extraction จากแบบจำลอง Pre-trained Deep Learning และถ่ายโอนการเรียนรู้มาที่แบบจำลองใหม่ ข้อดีของการใช้ Transfer Learning จะช่วยประหยัดเวลาในการฝึกข้อมูลแบบจำลองและประหยัดทรัพยากรอีกด้วย โดยศึกษาแบบจำลองที่มีการฝึกฝนข้อมูลและปรับพารามิเตอร์เท่านั้น (Training and Validation) และนำ Feature Extractor ที่ได้ไปประยุกต์ใช้กับงานอื่นๆที่ลักษณะคล้ายคลึงกัน เพื่อจำแนกข้อมูลประเภทใหม่ (New Classifier) ได้อย่างมีประสิทธิภาพ

ตารางที่ 2 การเปรียบเทียบค่าความถูกต้อง (Accuracy) ของแบบจำลองที่ใช้เทคนิคแตกต่างกัน [3]

Model	Training Accuracy	Validation Accuracy
Basic CNN	98.20%	72.40%
Fine Tuning CNN with Image Augmentation	81.30%	79.20%
Fine Tuning CNN with Pre-trained VGG16 Model and Image Augmentation	86.50%	95.40%

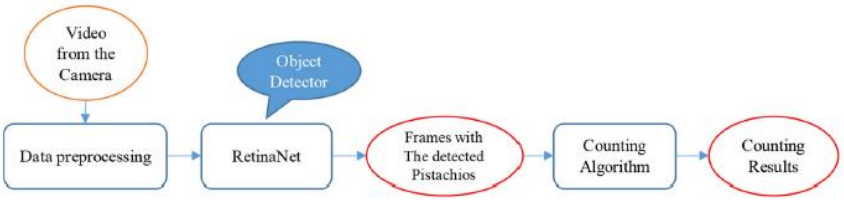
จากตารางที่ 2 พบว่าแบบจำลองที่เป็นแบบ Basic CNN และแบบจำลอง Fine Tuning CNN with Image Augmentation นั้นมีค่า Training Accuracy สูงกว่าค่า Validation Accuracy ซึ่งเป็นเพราะเกิด Overfitting ทำให้แบบจำลองทำนายได้ถูกต้องลดลง ส่วนแบบจำลอง Fine tuning CNN with Pre-trained VGG16 Model and Image Augmentation นั้นได้ค่า Validation Accuracy สูงสุดอยู่ที่ 95.40% เนื่องมีการทำ Pre-trained VGG16 และการเพิ่มข้อมูลภาพ (Image Augmentation) ทำให้แบบจำลองนั้นได้ฝึกฝนข้อมูลภาพที่หลายหลากมากกว่าจึงได้แบบจำลองที่มีประสิทธิภาพมากกว่าด้วย

3. งานวิจัยเรื่อง Deep Residual Learning for Image Recognition [4]

งานวิจัยนี้นำเสนอการจำแนกภาพโดยการใช้ Residual Neural Network (ResNet) ซึ่งเป็นโครงข่ายประสาทเทียมลึก (Deep Neural Network) โดยใช้แบบจำลอง ResNet ที่มีโครงสร้างแตกต่างกันหลายรูปแบบ ได้แก่ ResNet34, ResNet50, ResNet101 และResNet152 รวมทั้งใช้เทคนิค Residual Learning หรือ A Building Block ซึ่งเป็นการเชื่อมต่อข้ามชั้น (Skip Connection) ซึ่งจะช่วยให้แบบจำลองสามารถเรียนรู้ฟีเจอร์ที่ระดับต่ำ (Low-Level Features) และฟีเจอร์ที่ระดับสูง (High-Level Features) ได้อย่างมีประสิทธิภาพ และเป็นแบบ Element-Wise Addition คือการบวก Feature map จากชั้นก่อนหน้าเข้าด้วยกันเพื่อคำนวณฟีเจอร์ใหม่ โดยการเชื่อมต่อข้ามชั้น (Skip Connection) จะช่วยให้แบบจำลองสามารถรับข้อมูลจากชั้นก่อนหน้านี้ได้โดยตรงโดยไม่ต้องผ่านชั้นกลางทั้งหมด ช่วยให้แบบจำลองการเรียนรู้ได้ดีขึ้น และไม่ต้องเพิ่มจำนวนพารามิเตอร์ของแบบจำลองมากเกินไป อีกทั้งยังช่วยลดปัญหา Vanishing Gradient อีกด้วย

4. งานวิจัยเรื่อง Detecting and Counting Pistachios based on Deep Learning [5]

งานวิจัยนี้ได้ดำเนินการใช้ตัวตรวจจับวัตถุ RetinaNet ซึ่งใช้โครงข่ายประสาทเทียมลึก (Deep Neural Network) แบบ ResNet ดังรูปที่ 1 และใช้ชุดข้อมูลในการตรวจจับถั่วพิสตาชิโอในเฟรมวิดีโอที่เป็นวิดีโอทั้งหมด 6 รายการ มีระยะเวลารวม 167 วินาที และถั่วพิสตาชิโอที่ที่ป้ายกำกับ 3,927 เมล็ด สามารถตรวจจับและนับถั่วพิสตาชิโอที่มีเปลือกเปิดและเปลือกปิดได้ ในการฝึกสอนแบบจำลองสำหรับตัวตรวจจับวัตถุ RetinaNet นั้น จะแบ่งข้อมูลออกเป็น 5 ชุด และใช้แบบจำลองทั้งหมด 3 ประเภท ได้แก่ ResNet50, ResNet152 และVGG16 โดยที่แบบจำลอง ResNet50 ได้ผลลัพธ์ค่าเฉลี่ยความแม่นยำ (mAP) สูงสุดอยู่ที่ 91.87%



รูปที่ 1 แสดงกระบวนการทำงานในการตรวจจับและนับถั่วพิสตาชิโอ [5]

ข้อจำกัดของงานวิจัยนี้คือ ถั่วพิสตาชิโอเป็นวัตถุที่เคลื่อนไหวและมักจะหมุนตัวบนสายพานลำเลียง และจากมุมมองของกล้องอาจทำให้ตรวจจับผิดพลาดได้ เช่น ถั่วพิสตาชิโอที่เปลือกเปิดและถูกหมุนให้ด้านมีเปลือกเปิดกลับด้าน ทำให้ดูเหมือนกับถั่วพิสตาชิโอที่เปลือกปิดได้ และอาจจะปรากฏเป็นถั่วพิสตาชิโอที่เปลือกเปิดอีกครั้งในการตรวจจับรอบอื่นๆ อีกด้วย

5. งานวิจัยเรื่อง EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks [6]

งานวิจัยนี้ศึกษาการปรับโครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN) แบบ EfficientNet โดยใช้การถ่ายโอนการเรียนรู้ (Transfer Learning) จากชุดข้อมูล ImageNet และกปรับขนาดแบบจำลองแบบผสม (Compound Scaling) ด้วยการปรับขนาดทั้ง 3 มิติของแบบจำลอง คือความกว้าง (Width), ความลึก (Depth) และความละเอียด (Resolution) โดยการปรับค่า Compound Coefficient หรือค่าอัตราส่วนที่แน่นอน ตั้งแต่ 0 ถึง 7 โดย ซึ่งเรียกชื่อแบบจำลองตามค่า Compound Coefficient เป็น EfficientNetB0 จนถึง EfficientNetB7

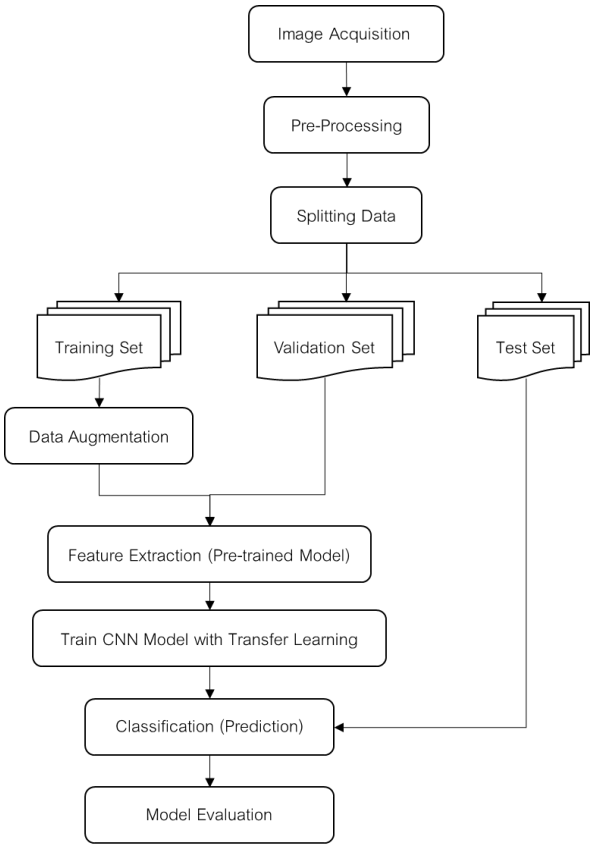
6. งานวิจัยเรื่อง Cultivar Identification of Pistachio Nuts in Bulk Mode through EfficientNet Deep Learning Model [7]

งานวิจัยนี้นำเสนอการประยุกต์ใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN) แบบ EfficientNetB3 (ค่า Compound Coefficient เท่ากับ 3) โดยมีโครงสร้างแบบจำลองดังภาพประกอบ 23 เพื่อจำแนกประเภทถั่วพิสตาชิโอทั้งหมด 4 สายพันธุ์ โดยมีการเพิ่มข้อมูลภาพ (Data Augmentation) และการปรับค่าพารามิเตอร์ต่างของแบบจำลอง (Fine Tuning) ผลลัพธ์งานวิจัยได้ค่าความถูกต้อง (Accuracy) อยู่ที่ 98.00%, ค่าเฉลี่ยความแม่นยำ (Average Precision) อยู่ที่ 96.73%, ค่า Recall อยู่ที่ 96.70% และค่า F1-Score อยู่ที่ 96.67% แต่ข้อจำกัดงานวิจัยนี้คือไม่ได้ดำเนินการเปรียบเทียบประสิทธิภาพแบบจำลองกับแบบจำลองโครงสร้างอื่นๆ

ทางผู้วิจัยได้ทบทวนวรรณกรรมที่เกี่ยวข้องและได้เลือกศึกษาแบบจำลองทั้งหมด 5 ประเภท ได้แก่ แบบจำลอง CNN ที่มีการเรียนรู้จากเริ่มต้น (CNN from Scratch), VGG16, VGG19, ResNet50 และ EfficientNetB0

วิธีดำเนินการ

1. กระบวนการทำงานของแบบจำลอง



รูปที่ 2 แสดงกระบวนการทำงานของงานวิจัย

จากรูปที่ 2 ขั้นตอนในการวิจัยเริ่มจากการเก็บรวบรวมข้อมูล (Image Acquisition) จากนั้นเตรียมข้อมูลภาพ (Pre-Processing) และแบ่งข้อมูลภาพ (Splitting) โดยแบ่งข้อมูลออกเป็น 3 ส่วนเป็น Training Set, Validation Set และ Test Set ในอัตราส่วน 80:10:10 จากนั้นทำการเรียนรู้แบบจำลอง โดยใช้เทคนิคการถ่ายโอนเรียนรู้ (Transfer Learning) จากนั้นทดสอบการจำแนกประเภทของข้อมูลที่ใช้ทดสอบ (Test Set) และประเมินประสิทธิภาพแบบจำลอง (Model Evaluation)

2. การเก็บรวบรวมข้อมูล (Image Acquisition)

ข้อมูลภาพทั้งหมดในงานวิจัยนี้อยู่ในรูปไฟล์ JPG จำนวน 3,204 รูป ซึ่งจำแนกออกเป็น 3 ประเภท ได้แก่ ถั่วพิสตาชิโอสายพันธุ์ Kirmizi, Siirt และ Kerman โดยที่ข้อมูลภาพสายพันธุ์ Kirmizi จำนวน 1,232 รูป และสายพันธุ์ Siirt จำนวน 916 รูป มาจากข้อมูลสาธารณะ [2] และได้เก็บข้อมูลภาพเพิ่มเติม เป็นสายพันธุ์ Kerman จำนวน 1,056 รูป ซึ่งเป็นชุดข้อมูลแบบ Balanced Dataset

รูปที่ 3 แสดงถั่วพิสตาชิโอสายพันธุ์ Kirmizi, Siirt และ Kerman ตามลำดับ

3. การเตรียมข้อมูล (Pre-Processing)

ดำเนินการย่อขนาดของรูปทั้งหมดให้มีขนาด 224×224 พิกเซล จากนั้นการแบ่งชุดข้อมูลดังกล่าวเป็น Training Set, Validation Set และ Test Set ในอัตราส่วน 80:10:10 และทำการสร้างโฟลเดอร์ของข้อมูลภาพทั้งหมด 3 โฟลเดอร์ คือ Train, Test, Validation และในแต่ละโฟลเดอร์หลักจะมีโฟลเดอร์ย่อยอีกจำนวน 3 โฟลเดอร์ ตามจำนวนประเภทของสายพันธุ์ถั่วพิสตาชิโอ คือ Kirmizi, Siirt และ Kerman รวมทั้งเพิ่มความหลากหลายของภาพและเพิ่มจำนวนให้กับชุดข้อมูลที่ใช้ในการเรียนรู้ (Training Set) โดยการสุ่มขยายหรือย่อภาพ (Zoom Range), การเลื่อนภาพแนวนอน (Width Shift Range), การเลื่อนภาพแนวตั้ง (Height Shift Range), การหมุนภาพ (Rotation Range), การเติมพิกเซล (Fill Mode) การพลิกภาพแนวนอน (Horizontal Flip), การพลิกภาพแนวตั้ง (Vertical Flip) และการปรับเปลี่ยนช่วงความสว่าง (Brightness Range)

4. การสร้างแบบจำลอง (Modeling)

ในงานวิจัยนี้ดำเนินการบน Google Colab โดยใช้ประเภท GPU คือ V100 และได้ทดลองกับแบบจำลองทั้งหมด 5 ประเภท ได้แก่ CNN from Scratch, VGG16, VGG19, ResNet50 และ EfficientNetB0 ซึ่งแต่ละแบบจำลองจะแตกต่างกันในส่วนของชั้นที่มีน้ำหนัก (Trainable Layers) และมีชั้นที่ไม่มีน้ำหนัก (Non-Trainable Layers) แต่ทุกแบบจำลองจะใช้ชั้นสำหรับการจำแนก (Classification Layers) ที่เหมือนกันทั้งหมด

สำหรับแบบจำลองที่ได้รับการเรียนรู้ข้อมูลมาก่อน (Pre-Trained Model) ทั้ง 4 แบบจำลอง ได้แก่ VGG16, VGG19, ResNet50 และ EfficientNetB0 จะใช้วิธีการถ่ายโอนการเรียนรู้ (Transfer Learning) จากข้อมูล ImageNet ซึ่งแต่ละแบบจำลองใช้พารามิเตอร์ในการสร้างแบบจำลอง มีรายละเอียดดังตารางที่ 3 และมีจำนวนพารามิเตอร์ทั้งหมดแสดงดังตารางที่ 4 โดยแต่แบบจำลองจะแบ่งย่อยอีก 2 รูปแบบ ดังนี้

1. แบบจำลองที่ไม่ได้ทำการปรับแต่งเพิ่มเติม (Non-Fine Tune) ต่อชั้น convolutional หรือชั้น dense ที่มีอยู่ ซึ่งน้ำหนักที่ได้จากการฝึกฝนบนชุดข้อมูล ImageNet ถูกใช้โดยตรง โดยไม่มีการเปลี่ยนแปลงใดๆ
2. แบบจำลองที่ทำการปรับแต่ง (Fine Tune) โดยปรับแต่งชั้นบางส่วนให้เป็นชั้นที่มีน้ำหนัก (Trainable Layers) ก่อนจะเข้าชั้นสำหรับการจำแนก (Classification Layers)

ตารางที่ 3 แสดงพารามิเตอร์ที่ใช้ในการสร้างแบบจำลอง

Optimizer	Epoch	Learning Rate	Batch Size
ADAM	10	10^{-4}	8
			16
			32
			64
		10^{-5}	8
			16
			32
			64

ตารางที่ 4 แสดงจำนวนพารามิเตอร์ทั้งหมดของแบบจำลองต่างๆ

Model		Trainable Parameters	Non-Trainable Parameters	Total Parameters
CNN from Scratch		399,008,739 (1.49 GB)	0 (0.00 Byte)	399,008,739 (1.49 GB)
VGG16	Non-Fine Tune	119,558,147 (456.08 MB)	14,714,688 (56.13 MB)	134,272,835 (512.21 MB)
	Fine Tune	126,637,571 (483.08 MB)	7,635,264 (29.13 MB)	
VGG19	Non-Fine Tune	119,558,147 (456.08 MB)	20,024,384 (76.39 MB)	139,582,531 (532.47 MB)
	Fine Tune	128,997,379 (492.09 MB)	10,585,152 (40.38 MB)	
ResNet50	Non-Fine Tune	427,839,491 (1.59 GB)	23,587,712 (89.98 MB)	451,427,203 (1.68 GB)
	Fine Tune	436,770,819 (1.63 GB)	14,656,384 (55.91 MB)	
EfficientNetB0	Non-Fine Tune	273,698,819 (1.02 GB)	4,049,571 (15.45 MB)	277,748,390 (1.03 GB)
	Fine Tune	275,041,587 (1.02 GB)	2,706,803 (10.33 MB)	

5. การประเมินผลแบบจำลอง (Model Evaluation)

ในการประเมินผลแบบจำลองโครงสร้างต่างๆ เพื่อดูประสิทธิภาพในการจำแนกข้อมูลภาพ รวมถึงเวลาที่ใช้ในการเรียนรู้ (Training Time) ผู้วิจัยได้กำหนดตัววัดประสิทธิภาพของแบบจำลองโดยใช้ Performance Metrics ได้แก่ ค่าความถูกต้อง (Accuracy), ค่าความแม่นยำ (Precision), ค่าความไว (Recall) และค่า F-1 Score

ผลการวิจัยและอภิปรายผลการวิจัย

1. แบบจำลอง CNN from Scratch ได้ผลการดำเนินการดังตารางที่ 5

ตารางที่ 5 การเปรียบเทียบประสิทธิภาพของแบบจำลอง CNN from Scratch

Learning Rate	Batch Size	Precision	Recall	F-1 Score	Accuracy
10^{-4}	8	0.9034	0.9000	0.9003	0.9000
	16	0.8966	0.8969	0.8966	0.8969
	32	0.8903	0.8906	0.8895	0.8906
	64	0.8965	0.8969	0.8965	0.8969
10^{-5}	8	0.8796	0.8750	0.8754	0.8750
	16	0.9092	0.9094	0.9092	0.9094
	32	0.8719	0.8719	0.8721	0.8719
	64	0.9218	0.9219	0.9218	0.9219

จากตารางที่ 5 พบว่าแบบจำลอง CNN from Scratch มีประสิทธิภาพที่ดีที่สุดเมื่ออัตราการเรียนรู้ (Learning Rate) เท่ากับ 10^{-5} และ ขนาดของกลุ่มข้อมูล (Batch Size) เท่ากับ 64 โดยได้ค่าความถูกต้อง (Accuracy) สูงสุดที่ 0.9219

2. แบบจำลอง VGG16 ได้ผลการดำเนินการดังตารางที่ 6

ตารางที่ 6 การเปรียบเทียบประสิทธิภาพของแบบจำลอง VGG16

Model	Learning Rate	Batch Size	Precision	Recall	F-1 Score	Accuracy
Non-Fine Tune	10^{-4}	8	0.9205	0.9156	0.9161	0.9156
		16	0.9471	0.9469	0.9467	0.9469
		32	0.9658	0.9625	0.9623	0.9625
		64	0.9596	0.9594	0.9594	0.9594
	10^{-5}	8	0.9627	0.9625	0.9625	0.9625
		16	0.9562	0.9563	0.9562	0.9563
		32	0.9564	0.9563	0.9561	0.9563
		64	0.9711	0.9688	0.9685	0.9688

ตารางที่ 6 (ต่อ)

Model	Learning Rate	Batch Size	Precision	Recall	F-1 Score	Accuracy
Fine Tune	10^{-4}	8	0.9724	0.9719	0.9718	0.9719
		16	0.9689	0.9688	0.9688	0.9688
		32	0.9656	0.9656	0.9656	0.9656
		64	0.9377	0.9313	0.9299	0.9313
	10^{-5}	8	0.9781	0.9781	0.9781	0.9781
		16	0.9688	0.9688	0.9688	0.9688
		32	0.9719	0.9719	0.9719	0.9719
		64	0.9470	0.9469	0.9469	0.9469

จากตารางที่ 6 พบว่าแบบจำลองที่ทำการปรับแต่ง (Fine Tune) มีประสิทธิภาพที่ดีที่สุดเมื่ออัตราการเรียนรู้ (Learning Rate) เท่ากับ 10^{-5} และ ขนาดของกลุ่มข้อมูล (Batch Size) เท่ากับ 8 โดยได้ค่าความถูกต้อง (Accuracy) สูงสุดที่ 0.9781

3. แบบจำลอง VGG19 ได้ผลการดำเนินการดังตารางที่ 7

ตารางที่ 7 การเปรียบเทียบประสิทธิภาพของแบบจำลอง VGG19

Model	Learning Rate	Batch Size	Precision	Recall	F-1 Score	Accuracy
Non-Fine Tune	10^{-4}	8	0.9596	0.9594	0.9595	0.9594
		16	0.9659	0.9656	0.9657	0.9656
		32	0.9657	0.9656	0.9655	0.9656
		64	0.9533	0.9531	0.9532	0.9531
	10^{-5}	8	0.9689	0.9688	0.9688	0.9688
		16	0.9689	0.9688	0.9688	0.9688
		32	0.9571	0.9563	0.9561	0.9563
		64	0.9692	0.9688	0.9687	0.9688
Fine Tune	10^{-4}	8	0.9264	0.9250	0.9252	0.9250
		16	0.9765	0.9750	0.9748	0.9750
		32	0.9753	0.9750	0.9749	0.9750
		64	0.9563	0.9563	0.9563	0.9563
	10^{-5}	8	0.9695	0.9688	0.9686	0.9688
		16	0.9594	0.9563	0.9557	0.9563
		32	0.9783	0.9781	0.9781	0.9781
		64	0.9657	0.9656	0.9656	0.9656

จากตารางที่ 7 พบว่าแบบจำลองที่ทำการปรับแต่ง (Fine Tune) มีประสิทธิภาพที่ดีที่สุดเมื่ออัตราการเรียนรู้ (Learning Rate) เท่ากับ 10^{-4} และ ขนาดของกลุ่มข้อมูล (Batch Size) เท่ากับ 16 โดยได้ค่าความถูกต้อง (Accuracy) สูงสุดที่ 0.9750

4. แบบจำลอง ResNet50 ได้ผลการดำเนินการดังตารางที่ 8

ตารางที่ 8 การเปรียบเทียบประสิทธิภาพของแบบจำลอง ResNet50

Model	Learning Rate	Batch Size	Precision	Recall	F-1 Score	Accuracy
Non-Fine Tune	10^{-4}	8	0.9758	0.9750	0.9749	0.9750
		16	0.9347	0.9281	0.9285	0.9281
		32	0.9660	0.9656	0.9657	0.9656
		64	0.9595	0.9594	0.9594	0.9594
	10^{-5}	8	0.9631	0.9625	0.9626	0.9625
		16	0.9641	0.9625	0.9629	0.9625
		32	0.9657	0.9656	0.9655	0.9656
		64	0.9659	0.9656	0.9657	0.9656
Fine Tune	10^{-4}	8	0.9750	0.9750	0.9750	0.9750
		16	0.9814	0.9812	0.9813	0.9812
		32	0.9674	0.9656	0.9653	0.9656
		64	0.9753	0.9750	0.9749	0.9750
	10^{-5}	8	0.9599	0.9594	0.9595	0.9594
		16	0.9505	0.95	0.9498	0.9500
		32	0.9688	0.9688	0.9688	0.9688
		64	0.9324	0.9313	0.9317	0.9312

จากตารางที่ 8 พบว่าแบบจำลองที่ทำการปรับแต่ง (Fine Tune) มีประสิทธิภาพที่ดีที่สุดเมื่ออัตราการเรียนรู้ (Learning Rate) เท่ากับ 10^{-4} และ ขนาดของกลุ่มข้อมูล (Batch Size) เท่ากับ 16 โดยได้ค่าความถูกต้อง (Accuracy) สูงสุดที่ 0.9812

5. แบบจำลอง EfficientNetB0 ได้ผลการดำเนินการดังตารางที่ 9

ตารางที่ 9 การเปรียบเทียบประสิทธิภาพของแบบจำลอง EfficientNetB0

Model	Learning Rate	Batch Size	Precision	Recall	F-1 Score	Accuracy
Non-Fine Tune	10^{-4}	8	0.8715	0.7875	0.7779	0.7875
		16	0.9573	0.9531	0.9525	0.9531
		32	0.9266	0.9125	0.9125	0.9125
		64	0.9389	0.9344	0.9351	0.9344
	10^{-5}	8	0.8884	0.8469	0.8327	0.8469
		16	0.9348	0.9281	0.9288	0.9281
		32	0.9371	0.9313	0.9316	0.9313
		64	0.9594	0.9563	0.9565	0.9563
Fine Tune	10^{-4}	8	0.9327	0.9313	0.9317	0.9313
		16	0.9372	0.9250	0.9232	0.9250
		32	0.9246	0.9094	0.9093	0.9094
		64	0.9260	0.9125	0.9095	0.9125
	10^{-5}	8	0.9192	0.9062	0.9067	0.9062
		16	0.9308	0.9156	0.9130	0.9156
		32	0.9592	0.9594	0.9592	0.9594
		64	0.9628	0.9625	0.9625	0.9625

จากตารางที่ 9 พบว่าแบบจำลองที่ทำการปรับแต่ง (Fine Tune) มีประสิทธิภาพที่ดีที่สุดเมื่ออัตราการเรียนรู้ (Learning Rate) เท่ากับ 10^{-5} และ ขนาดของกลุ่มข้อมูล (Batch Size) เท่ากับ 64 โดยได้ค่าความถูกต้อง (Accuracy) สูงสุดที่ 0.9625

สรุปผลการวิจัย

จากการศึกษาและเปรียบเทียบประสิทธิภาพของแบบจำลองที่ใช้ในงานวิจัยนี้ ทั้งหมด 5 ประเภท ได้แก่ CNN from Scratch, VGG16, VGG19, ResNet50 และ EfficientNetB0 เพื่อศึกษาวิเคราะห์และหาค่าไฮเปอร์พารามิเตอร์ที่เหมาะสมของแต่ละแบบจำลอง เพื่อสามารถนำผลลัพธ์ที่ได้ไปปรับใช้ให้เหมาะสมกับการจำแนกชุดข้อมูลภาพพถ่ายพิสตาชิโอ 3 สายพันธุ์ (Kirmizi, Siirt และ Kerman) ได้อย่างมีประสิทธิภาพและมีค่าความถูกต้องสูงสุด โดยการเปรียบเทียบประสิทธิภาพของแต่ละแบบจำลอง ดังตารางที่ 10 และสามารถอธิบายสรุปผลการดำเนินการวิจัยได้ดังนี้

1. แบบจำลองที่ได้รับการเรียนรู้ข้อมูลมาก่อน (Pre-Trained Model) โดยใช้วิธีการถ่ายโอนการเรียนรู้ (Transfer Learning) มีประสิทธิภาพดีกว่า แบบจำลองที่มีการเรียนรู้ตั้งแต่เริ่มต้น (CNN from Scratch)

2. สำหรับแบบจำลองที่ได้รับการเรียนรู้ข้อมูลมาก่อน (Pre-Trained Model) พบว่าแบบจำลองที่ทำการปรับแต่ง (Fine Tune) มีประสิทธิภาพดีกว่าแบบจำลองที่ไม่ได้ทำการปรับแต่งเพิ่มเติม (Non-Fine Tune)
3. แบบจำลอง ResNet50 เป็นแบบจำลองที่มีประสิทธิภาพสูงสุด โดยมีค่าความถูกต้องอยู่ที่ 0.9812 ตามด้วยแบบจำลอง VGG16 มีค่าความถูกต้อง (Accuracy) อยู่ที่ 0.9781, แบบจำลอง VGG19 มีค่าความถูกต้อง (Accuracy) อยู่ที่ 0.9750, แบบจำลอง EfficientNetB0 มีค่าความถูกต้อง (Accuracy) อยู่ที่ 0.9625 และแบบจำลองที่มีการเรียนรู้ตั้งแต่เริ่มต้น มีค่าความถูกต้อง (Accuracy) อยู่ที่ 0.9219 ตามลำดับ

ตารางที่ 10 การเปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 5 ประเภท

Model	Learning Rate	Batch Size	Accuracy
ResNet50 with Fine Tune	10^{-4}	16	0.9812
VGG16 with Fine Tune	10^{-5}	8	0.9781
VGG19 with Fine Tune	10^{-4}	16	0.9750
EfficientNetB0 with Fine Tune	10^{-5}	64	0.9625
CNN from Scratch	10^{-5}	64	0.9219

กิตติกรรมประกาศ

การจัดทำวิจัยได้รับการสนับสนุนจากบัณฑิตวิทยาลัย มหาวิทยาลัยศรีนครินทรวิโรฒ ในการนำเสนอผลงานวิจัย ผู้วิจัยจึง ขอขอบคุณมา ณ ที่นี้

เอกสารอ้างอิง

[1] Wikipedia. "Pistachio." Retrieved from https://en.wikipedia.org/wiki/Pistachio (accessed.

[2] D. Singh et al., "Classification and Analysis of Pistachio Species with Pre-Trained Deep Learning Models," Electronics, vol. 11, no. 7, p. 981, 2022, doi: 10.3390/electronics11070981.

[3] S. Tammina, "Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images," International Journal of Scientific and Research Publications, vol. 9, no. 10, pp. 143-150, 2019, doi: 10.29322/IJSRP.9.10.2019.p9420.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770-778, 2016, doi: 10.1109/CVPR.2016.90.

[5] M. Rahimzadeh and A. Attar, "Detecting and counting pistachios based on deep learning," Iran Journal of Computer Science, vol. 5, no. 1, pp. 69-81, 2022/03/01 2022, doi: 10.1007/s42044-021-00090-6.

[6] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," 36th International Conference on Machine Learning (ICML 2019), 2019, doi: https://doi.org/10.48550/arXiv.1905.11946.

[7] A. Soleimanipour, M. Azadbakht, and A. Rezaei Asl, "Cultivar identification of pistachio nuts in bulk mode through EfficientNet deep learning model," Journal of Food Measurement and Characterization, vol. 16, no. 4, pp. 2545-2555, 2022/08/01 2022, doi: 10.1007/s11694-022-01367-5.