

การเรียนรู้ถ่ายโอนสำหรับการจำแนกประเภทภาพเนื้อลายหินอ่อนเทียม
ด้วยโครงข่ายประสาทคอนโวลูชัน



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2563
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

Transfer Learning for Image Classification of Artificial Marbling Beef
with Convolutional Neural Networks



A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Computer Science
Department of Computer Engineering
FACULTY OF ENGINEERING
Chulalongkorn University
Academic Year 2020
Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การเรียนรู้ถ่ายโอนสำหรับการจำแนกประเภทภาพเนื้อลาย
	หินอ่อนเทียมด้วยโครงข่ายประสาทคอนโวลูชัน
โดย	น.ส.เกรช พานิชกรณ์
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร.ญาใจ ลีมียะกรณ์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้หัวข้อวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่ง
ของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.สุพจน์ เตชวรสินสกุล)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.สุกรี สินธุภิญโญ)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(รองศาสตราจารย์ ดร.ญาใจ ลีมียะกรณ์)

..... กรรมการภายนอกมหาวิทยาลัย
(อาจารย์ ดร.ภาสกร อภิรักษ์วรพินิต)

CHULALONGKORN UNIVERSITY

เกรช พานิชกรณ : การเรียนรู้ถ่ายโอนสำหรับการจำแนกประเภทภาพเนื้อลายหินอ่อน
 เทียมด้วยโครงข่ายประสาทคอนโวลูชัน. (Transfer Learning for Image
 Classification of Artificial Marbling Beef with Convolutional Neural
 Networks) อ.ที่ปรึกษาหลัก : รศ. ดร.ญาใจ ลีมปิยะกรณ

ในปัจจุบัน เทคนิคการประมวลผลภาพถูกนำมาใช้กันอย่างแพร่หลายในหลากหลาย
 อุตสาหกรรม หนึ่งในนั้นคือการควบคุมคุณภาพที่ใช้ในอุตสาหกรรมการผลิตอาหาร ขณะเดียวกัน
 หนึ่งในปัญหาที่การเรียนรู้เชิงลึกสามารถนำมาใช้ตอบโจทย์ได้ดีเยี่ยมคือปัญหาการจำแนกรูปภาพ
 ในมุมมองของการเรียนรู้เชิงลึก ปัญหาที่หลากหลายของการจำแนกประเภทภาพสามารถแก้ไขได้
 อย่างรวดเร็วผ่านการเรียนรู้ถ่ายโอน งานวิจัยนี้จึงนำเสนอวิธีการประยุกต์ใช้เทคนิคการเรียนรู้ถ่าย
 โอนในการฝึกสอนแบบจำลองโครงข่ายคอนโวลูชันเชิงลึกเพื่อจำแนกภาพเนื้อลายหินอ่อนเทียม
 หรือเป็นเนื้อลายหินอ่อนแท้ แบบจำลองที่เทรนมาเรียบร้อยแล้วสามแบบจำลอง ประกอบด้วย วิธี
 จี16 เรสเน็ต50 และ อินเซปชันวี3 ได้ถูกเลือกมาใช้ในการทดลองเพื่อสร้างแบบจำลองทั้งหมด 4
 ตัว ประกอบด้วย ซีเอ็นเอ็นปกติ ซีเอ็นเอ็น+วิธีจี16 ซีเอ็นเอ็น+เรสเน็ต50 และ ซีเอ็นเอ็น+อินเซป
 ชันวี3 พบว่า สมรรถนะแบบจำลองซีเอ็นเอ็น+อินเซปชันวี3 ให้ผลลัพธ์ที่ดีที่สุด จึงถูกเลือกนำไป
 ปรับละเอียด การประเมินผลบนชุดข้อมูลทดสอบของแบบจำลองซีเอ็นเอ็น+อินเซปชันวี3ภายหลัง
 การปรับแต่งให้ผลลัพธ์ค่าความแม่นยำที่ดีที่สุดคือ 96.7% เห็นได้ว่า แนวทางการจำแนกประเภท
 ภาพที่นำเสนอมีความหวังสามารถนำไปพัฒนาต่อยอดเพื่อเป็นประโยชน์ต่อผู้ซื้อในการตรวจสอบ
 เนื้อลายหินอ่อนเทียมที่อาจตั้งราคาสูงเกินจริง อันเป็นผลมาจากการฉ้อโกงให้มีลายมากมาย
 สวยงาม ซึ่งจะทำให้เนื้อไม้รสชาติดีขึ้นรวมทั้งสามารถตั้งราคาที่สูงขึ้นได้อีกด้วย

CHULALONGKORN UNIVERSITY

สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์

ปีการศึกษา 2563

ลายมือชื่อนิสิต

ลายมือชื่อ อ.ที่ปรึกษาหลัก

6270020421 : MAJOR COMPUTER SCIENCE

KEYWORD: Image classification, Convolutional neural networks, Transfer learning, Artificial marbling beef

Grace Panitchakorn : Transfer Learning for Image Classification of Artificial Marbling Beef with Convolutional Neural Networks. Advisor: Yachai Limpiyakorn

Nowadays, image processing is widely used as one of the quality control techniques applied in the food and neutral industrial sectors. Whereas one of the problem domains that deep learning excels is image classification. From a deep learning perspective, a variety of image classification problems could be quickly solved through transfer learning. This research thus presents a method of applying the technique of transfer learning to training deep convolutional neural networks models for binary classification if the image is artificial marbling beef. The three selected pre-trained models include VGG16, ResNet50, and InceptionV3. The experiments were carried out to construct the four models consisting of the CNN trained from scratch, the CNN+VGG16, the CNN+ResNet50, and the CNN+InceptionV3. Among all, the CNN+InceptionV3 achieved the best performance, the model was then chosen for fine tuning. The evaluation of the final CNN+InceptionV3 on test dataset reported the best accuracy of 96.7%. The proposed approach of image classification is promising. And since the intensely, beautifully marbled fat results in rich flavor and increased price of the cuts, more or less, the presented classifier would benefit the buyers for detecting imposter marbling beef with mark-up price.

Field of Study: Computer Science

Student's Signature

Academic Year: 2020

Advisor's Signature

กิตติกรรมประกาศ

วิทยานิพนธ์เล่มนี้สำเร็จสมบูรณ์ได้ด้วยดีเพราะได้รับความกรุณาอย่างยิ่งจาก รองศาสตราจารย์ ดร.ญาใจ ลิ้มปิยะกรรณ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ที่ความรู้ ให้คำแนะนำทั้งคำปรึกษา คอยผลักดัน และตรวจแก้ไขข้อบกพร่องมาโดยตลอด ผู้วิจัยขอกราบขอบพระคุณด้วยความเคารพอย่างสูงไว้ ณ ที่นี้

ขอขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร.สุกรี สินธุภิญโญ ประธานกรรมการสอบวิทยานิพนธ์ และ อาจารย์ ดร.ภาสกร อภิรักษ์วรพินิต กรรมการสอบวิทยานิพนธ์ที่กรุณาเสียสละเวลาอันมีค่า ในการตรวจสอบและให้คำแนะนำที่เป็นประโยชน์ในการทำวิทยานิพนธ์ในครั้งนี้

ขอขอบพระคุณบิดามารดา วงศาคุณาญาติทั้งหลายที่คอยเป็นห่วงเป็นใย ถามไถ่เสมอมา และขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ที่สนับสนุนและให้กำลังใจมาตั้งแต่เริ่มต้น คอยให้คำแนะนำ ความช่วยเหลือ และสนับสนุน รวมถึงขอบคุณตัวเองที่ไม่ยอมแพ้และท้อถอยและยืนหยัด ซึ่งทำให้วิทยานิพนธ์สำเร็จลุล่วงไปได้ด้วยดีผู้วิจัยหวังเป็นอย่างยิ่งว่าวิทยานิพนธ์ฉบับนี้จะเป็นประโยชน์ต่อผู้ที่สนใจต่อไปในภายภาคหน้า

เกรช พานิชกรณ

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ค
บทคัดย่อภาษาอังกฤษ.....	ง
กิตติกรรมประกาศ.....	จ
สารบัญ.....	ฉ
สารบัญตาราง.....	ฅ
สารบัญรูปภาพ.....	ญ
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์.....	2
1.3 ขอบเขตการดำเนินงาน.....	2
1.4 ขั้นตอนการดำเนินงาน.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.6 ลำดับการจัดเรียงวิทยานิพนธ์.....	3
1.7 ผลงานที่ได้รับการตีพิมพ์.....	3
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	4
2.1 ทฤษฎีที่เกี่ยวข้อง.....	4
2.1.1 เนื้อลายหินอ่อนเทียม (Artificial Marbling Beef).....	4
2.1.2 โครงข่ายประสาทคอนโวลูชัน (Convolutional Neural Network- CNN)	4
2.1.3 โครงข่ายประสาทแบบคอนโวลูชันแบบวีจีจี 16 (VGG16)	6
2.1.4 โครงข่ายประสาทเชิงลึกแบบ Residual Network (ResNet)	6
2.1.5 โครงข่ายประสาทแบบ InceptionV3	7

2.1.6 การเรียนรู้ถ่ายโอน (Transfer Learning).....	8
2.2 งานวิจัยที่เกี่ยวข้อง	9
2.2.1 งานวิจัยที่เกี่ยวข้องกับการคัดแยกคุณภาพเนื้อด้วยภาพ	10
2.2.2 งานวิจัยที่ประยุกต์ใช้ transfer learning กับชุดข้อมูลภาพ	11
บทที่ 3 แนวคิดและวิธีการวิจัย.....	15
3.1 การเตรียมชุดข้อมูล	15
3.2 การสร้างแบบจำลองเพื่อการเปรียบเทียบแบบจำลองการเรียนรู้ถ่ายโอน.....	18
3.3 การวัดสมรรถนะของแบบจำลอง.....	19
3.4 การปรับละเอียดสมรรถนะของแบบจำลอง.....	20
3.4.1 ปรับละเอียดจำนวนชั้นของแบบจำลอง	20
3.4.2 Stratified K-Fold Cross Validation (K=10)	20
บทที่ 4 การทดลองและผลการทดลอง	22
4.1 สภาพแวดล้อมที่ใช้ในการทดลอง	22
4.2 การเตรียมชุดข้อมูลที่ใช้ในการทดลอง	22
4.3 การดำเนินการสร้างแบบจำลอง.....	26
4.3.1 แบบจำลองโครงข่ายประสาทแบบคอนโวลูชัน (Convolutional Neural Network-CNN).....	26
4.3.2 แบบจำลองโครงข่ายประสาทคอนโวลูชันแบบวีจีจี 16 (VGG16).....	27
4.3.3 แบบจำลองโครงข่ายประสาทเชิงลึกแบบ Residual Network (ResNet50).....	29
4.3.4 แบบจำลองโครงข่ายประสาทแบบ InceptionV3	30
4.4 การฝึกสอนแบบจำลอง.....	31
4.5 ผลการทดลอง.....	33
4.6 การปรับละเอียดสมรรถนะแบบจำลอง	36
บทที่ 5 สรุปผลการวิจัย	45

บรรณานุกรม.....	47
ประวัติผู้เขียน.....	51



จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

สารบัญตาราง

หน้า

ตารางที่ 1 เปรียบเทียบสมรรถนะของแบบจำลอง VGG-16, ResNet-50 และ InceptionV3 จากการใช้ชุดข้อมูลรูปภาพ ImageNet ในการเรียนรู้ [13].....	8
ตารางที่ 2 ผลลัพธ์ของ InceptionV3 ที่ปรับแต่งอย่างละเอียดในชั้นต่างๆ	14
ตารางที่ 3 สรุปจำนวนข้อมูลสอน ข้อมูลตรวจสอบ ข้อมูลทดสอบของเนื้อทั้งสองประเภท.....	17
ตารางที่ 4 confusion matrix สำหรับการจำแนกเนื้อลายหินอ่อนเทียม	19
ตารางที่ 5 สมรรถนะของการฝึกสอนแบบจำลอง CNN, VGG16, RestNet50 และ InceptionV3.	32
ตารางที่ 6 สมรรถนะของการทดสอบแบบจำลอง CNN, VGG16, RestNet50 และ InceptionV3.	35
ตารางที่ 7 ค่าตัววัดสมรรถนะแบบจำลองการเรียนรู้แบบถ่ายโอน InceptionV3 หลังการปรับละเอียดการละทิ้งการตริงชั้นต่าง ๆ ในขั้นตอนการทดสอบแบบจำลอง	41
ตารางที่ 8 ค่าตัววัดสมรรถนะแบบจำลองการเรียนรู้แบบถ่ายโอน InceptionV3 หลังการปรับละเอียดการละทิ้งการตริงชั้น 228 และใช้ Stratified 10-Fold Cross Validation เพื่อปรับปรุงสมรรถนะแบบจำลอง.....	44

สารบัญรูปภาพ

	หน้า
รูปที่ 1 เนื้อลายหินอ่อนธรรมชาติ (ซ้าย) และเนื้อลายหินอ่อนเทียม (ขวา).....	4
รูปที่ 2 โครงสร้างโครงข่ายประสาทคอนโวลูชัน.....	5
รูปที่ 3 การสกัดฟังก์ชันลักษณะของรูปภาพในชั้น convolutional ที่มีการเพิ่ม padding และกำหนด stride เท่ากับ 1	5
รูปที่ 4 โครงสร้างโครงข่ายประสาทคอนโวลูชันแบบ VGG-16.....	6
รูปที่ 5 Residual block.....	7
รูปที่ 6 โครงสร้างโครงข่ายประสาทแบบ InceptionV3	8
รูปที่ 7 ภาพอัลตราซาวด์ของเนื้อวัวพันธุ์ฮันฮูของประเทศเกาหลีร่วมกับการตีกรอบจุดที่สนใจในภาพ	10
รูปที่ 8 ตัวอย่างรูปภาพพุทราตามลักษณะผลพุทรา.....	11
รูปที่ 9 ตัวอย่างชุดข้อมูลฝึกสอน และชุดข้อมูลทดสอบในงานตรวจจับรอยร้าวในอาคารจากรูปภาพ	12
รูปที่ 10 ตัวอย่างชุดข้อมูลภาพเนื้อเยื่อทั้ง 4 แบบที่ใช้ในการจำแนกเซลล์มะเร็งเต้านมจากรูปภาพจากการตรวจวินิจฉัยโดยตัดชิ้นเนื้อตรวจทางพยาธิวิทยา.....	13
รูปที่ 11 ตัวอย่างของการตัดสองวิธี: ตัดในแนวนอนและแนวตั้งจากภาพ M-NBI	14
รูปที่ 12 ขั้นตอนวิธีวิจัยการเรียนรู้ถ่ายโอนสำหรับการจำแนกประเภทภาพเนื้อลายหินอ่อนเทียม ...	15
รูปที่ 13 ตัวอย่างรูปเนื้อวัวชนิดไขมันจากร้านค้าออนไลน์ที่น่าเชื่อถือ	16
รูปที่ 14 การทำความสะอาดข้อมูลรูปภาพเนื้อ.....	16
รูปที่ 15 เพิ่มจำนวนชุดข้อมูลรูปภาพเนื้อแบบใช้มนุษย์.....	17
รูปที่ 16 โครงสร้างแบบจำลองร่วมกับการเรียนรู้การถ่ายโอน	18
รูปที่ 17 โครงสร้างแบบจำลองร่วมกับการเรียนรู้การถ่ายโอนร่วมกับการปรับละเอียด	20
รูปที่ 18 การแบ่งข้อมูลแบบ K-Fold Cross Validation	21

รูปที่ 19 การแบ่งข้อมูลแบบ Stratified K-Fold Cross Validation.....	21
รูปที่ 20 โครงสร้างโพลเดอร์ชุดข้อมูลที่ใช้ในการจำแนกประเภทภาพเนื้องายหินอ่อนเทียม.....	22
รูปที่ 21 ตัวอย่างภาพข้อมูลฝึกสอนโมเดลประเภทเนื้องายหินอ่อนเทียมใน google drive	23
รูปที่ 22 ตัวอย่างภาพข้อมูลฝึกสอนโมเดลประเภทเนื้อที่ไม่ใช่เนื้องายหินอ่อนเทียมใน google drive..	23
รูปที่ 23 ชุดคำสั่งในการเชื่อมต่อชุดข้อมูลใน google drive กับ google colab	24
รูปที่ 24 การกำหนดไคเรกทอรีเพื่อนำเข้าข้อมูลฝึกสอน ข้อมูลตรวจสอบและข้อมูลทดสอบ	24
รูปที่ 25 การเพิ่มจำนวนข้อมูล (data augmentation) สำหรับชุดข้อมูลที่ใช้ในการทดลอง	25
รูปที่ 26 โลบารี่ที่จำเป็นต่อการสร้างแบบจำลอง	26
รูปที่ 27 โครงสร้างของแบบจำลองโครงข่ายประสาทคอนโวลูชัน	27
รูปที่ 28 การเรียกใช้ pre-trained weight ร่วมกับการเรียนรู้แบบถ่ายโอนแบบวิธีจี 16	27
รูปที่ 29 การตรึงค่า (Freeze) แบบจำลอง และใช้ pre-trained weight จาก imagenet	28
รูปที่ 30 โครงสร้างแบบจำลองโครงข่ายประสาทแบบวิธีจี 16 เพื่อจำแนกเนื้องายหินอ่อนเทียม	28
รูปที่ 31 โครงสร้างแบบจำลองโครงข่ายประสาทคอนโวลูชันแบบวิธีจี 16.....	29
รูปที่ 32 โครงสร้างแบบจำลองโครงข่ายประสาทเชิงลึกแบบ Residual Network 50 ชั้น เพื่อจำแนกเนื้องายหินอ่อนเทียม.....	30
รูปที่ 33 โครงสร้างแบบจำลองโครงข่ายประสาทแบบ InceptionV3 เพื่อจำแนกเนื้องายหินอ่อนเทียม.....	31
รูปที่ 34 ฟังก์ชันในการวัดสมรรถนะของการฝึกสอนแบบจำลอง.....	32
รูปที่ 35 ฟังก์ชันในการวัดสมรรถนะของแบบจำลองด้วยชุดข้อมูลทดสอบ พร้อมทั้งการบันทึกผลรูปแบบไฟล์ excel	33
รูปที่ 36 ตัวอย่างไฟล์ผลลัพธ์การทดสอบแบบจำลองรูปแบบ excel.....	34
รูปที่ 37 ตัวอย่างผลลัพธ์การทดสอบแบบจำลองรูปแบบ confusion matrix และแสดงผลการทำนายแต่ละ class ด้วย classification_report.....	34
รูปที่ 38 ตัวอย่างความน่าจะเป็น และผลการจำแนกเนื้องายหินอ่อนเทียมกับชุดข้อมูลทดสอบร่วมกับการเรียนรู้ถ่ายโอนแบบ InceptionV3 เมื่อมีการกำหนดค่า learning rate เท่ากับ 0.001	37

รูปที่ 39 การปรับปรุงสมรรถนะของแบบจำลองการจำแนกเนื้อลายหินอ่อนเทียมร่วมกับการเรียนรู้ถ่ายโอนแบบ InceptionV3 เมื่อมีการกำหนดค่า learning rate เท่ากับ 0.001 38

รูปที่ 40 ปรับสมรรถนะแบบจำลองการเรียนรู้ถ่ายโอนแบบ InceptionV3 โดยทำการละทิ้งการตริงตั้งแต่ชั้นที่ 280, 249, 228, 165 และ 133 39

รูปที่ 41 ความแม่นยำของแบบจำลองการเรียนรู้แบบถ่ายโอน InceptionV3 จากการละทิ้งการตริงที่ชั้นต่างๆ ในขั้นตอนการฝึกสอนแบบจำลอง 39

รูปที่ 42 ความสูญเสียของแบบจำลองการเรียนรู้แบบถ่ายโอน InceptionV3 จากการละทิ้งการตริงที่ชั้นต่างๆ ในขั้นตอนการฝึกสอนแบบจำลอง 40

รูปที่ 43 การใช้ Stratified 10-Fold Cross Validation ปรับละเอียดแบบจำลอง Inception v3 เมื่อทำการละทิ้งการตริงตั้งแต่ชั้น 228 42

รูปที่ 44 ตัวอย่างการแสดงผลพรีชันของแบบจำลองรอบที่ 1 จากการปรับละเอียดโดยใช้ Stratified 10-Fold Cross Validation 43



บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของปัญหา

เนื้อสัตว์เป็นแหล่งโภชนาการที่สำคัญสำหรับคนจำนวนมากทั่วโลก ในช่วง 50 ปีที่ผ่านมา มีการผลิตมากกว่า 340 ล้านตันในแต่ละปี โดยการบริโภคเนื้อวัวต่อหัวสูงเป็นอันดับที่ 3 รองจากเนื้อสุกร และเนื้อสัตว์ปีก [1] ในส่วนของราคาเนื้อวัวนั้นจะตัดสินจากคุณภาพของเนื้อ ซึ่งมีระบบประเมินคุณภาพอยู่หลากหลายมาตรฐานแตกต่างกันตามสายพันธุ์ของวัวและแหล่งที่มา เช่น เนื้อวากิว (Wagyu) เนื้อแองกัส (Angus) รวมถึงเนื้อโคขุนของประเทศไทย ทั้งนี้ ปัจจัยสำคัญต่อการกำหนดคุณภาพของเนื้อวัวขึ้นอยู่กับลายหินอ่อน (Marbling) หรือระดับไขมันที่แทรกอยู่ในเนื้อวัว เมื่อความต้องการบริโภคเนื้อวัวที่มีลายหินอ่อนมีปริมาณสูง เทคโนโลยีการผลิตอาหารสำหรับเนื้อลายหินอ่อนเทียมก็ถูกพัฒนาเช่นกัน ซึ่งเทคโนโลยีการฉีดไขมันในเนื้อนั้น มีจุดประสงค์เพื่อให้เป็นเนื้อแดงที่มีความชุ่มฉ่ำจากการฉีดไขมันเพิ่ม อย่างไรก็ตาม หากผู้บริโภคไม่สามารถจำแนกได้ว่า เนื้อวัวที่มีไขมันนั้นเป็นเนื้อวัวที่ฉีดไขมันหรือไม่ได้ฉีดไขมัน ก็อาจต้องจ่ายเงินซื้อเนื้อวัวราคาที่สูงกว่าราคาจริงได้ [2] ดังนั้น วิธีการที่สามารถจำแนกเนื้อวัวที่ฉีดหรือไม่ได้ฉีดไขมัน จะเป็นประโยชน์ต่อผู้บริโภคในการเลือกซื้อได้ตรงตามความต้องการ

สำหรับการจำแนกคุณภาพเนื้อวัวนั้นมีการนำเทคโนโลยีการประมวลผลภาพ (image processing) ที่หลากหลายเพื่อคัดแยกชนิดของเนื้อสัตว์ การคัดแยกระดับของไขมันตามมาตรฐานต่างๆ หรือแม้กระทั่งการจำแนกเนื้อปนเปื้อนเชื้อโรค โครงข่ายประสาทคอนโวลูชันหรือซีเอ็นเอ็น (Convolution Neural networks – CNN) เป็นหนึ่งในเทคนิคการประมวลผลภาพที่ได้รับความนิยมและมีสมรรถนะสูงสำหรับจัดการกับปัญหาการจำแนกประเภทภาพที่คัดแยกคุณลักษณะของข้อมูลหลายมิติได้ดี จึงถูกประยุกต์ใช้ในหลากหลายอุตสาหกรรม รวมถึงอุตสาหกรรมอาหาร

วิทยานิพนธ์นี้ได้ศึกษาการจำแนกประเภทภาพเนื้อวัวลายหินอ่อนออกเป็น 2 ประเภท คือ 1) ภาพเนื้อวัวลายหินอ่อนเทียมที่ได้จากการฉีดไขมัน และ 2) ภาพเนื้อวัวลายหินอ่อนที่ไม่ได้ฉีดไขมัน จากชุดข้อมูลภาพที่ผู้วิจัยได้เก็บรวบรวมจากแหล่งที่น่าเชื่อถือบนอินเทอร์เน็ต ด้วยเทคนิคการเรียนรู้โครงข่ายประสาทคอนโวลูชันร่วมกับการใช้เทคนิคการเรียนรู้ถ่ายโอน จาก Pre-trained models ทั้งหมด 3 แบบ ประกอบด้วย VGG-16, ResNet50, และ InceptionV3 และนำ Pre-trained model ที่มีสมรรถนะที่ดีที่สุด มาปรับพารามิเตอร์แบบจำลองให้เหมาะสมกับชุดรูปภาพเนื้อวัวลายหินอ่อน และประเมินสมรรถนะของแบบจำลองที่ได้ด้วย accuracy, precision, recall และ F1

1.2 วัตถุประสงค์

1. ศึกษาการใช้เทคนิค Transfer Learning ร่วมกับโครงข่ายประสาทคอนโวลูชันในปัญหาการจำแนกภาพเนื้อวัวลายหินอ่อนเทียม
2. สร้างแบบจำลองโครงข่ายประสาทคอนโวลูชันร่วมกับ Pre-trained model ที่เลือกใช้ เพื่อจำแนกประเภทภาพเนื้อลายหินอ่อนเทียม

1.3 ขอบเขตการดำเนินงาน

1. แหล่งที่มาของภาพเก็บรวบรวมจากร้านค้าออนไลน์ ที่มีฉลากบอกประเภทว่าเป็นเนื้อวัวที่ฉีดยาหรือไม่ฉีดยาหรือไม่ฉีดยา
2. Pre-trained models ที่ใช้ในงานวิจัยนี้ ประกอบด้วย VGG-16, ResNet-50 และ InceptionV3
3. ประเมินสมรรถนะแบบจำลองผลลัพธ์ ด้วยค่า accuracy, confusion matrix และ F1-score

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาค้นคว้าความรู้จากทฤษฎีและงานวิจัยที่เกี่ยวข้อง
2. ทำการรวบรวมข้อมูลภาพเนื้อวัวแบบฉีดยาและไม่ฉีดยา
3. ทำการเตรียมข้อมูลสำหรับการสร้างแบบจำลอง
4. ออกแบบและสร้างแบบจำลองเพื่อจำแนกเนื้อวัวลายหินอ่อนเทียม
5. ปรับแต่งสมรรถนะของแบบจำลอง
6. สรุปผลการดำเนินงาน
7. เรียบเรียงและจัดทำบทความวิชาการ
8. เรียบเรียงและจัดทำเล่มวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้แบบจำลองที่สามารถใช้จำแนกภาพเนื้อลายหินอ่อน ออกเป็น 2 ประเภท คือ ภาพเนื้อฉีดยาและไม่ฉีดยา
2. ได้ต้นแบบแอปพลิเคชันที่สามารถพัฒนาต่อยอดเพื่อเป็นประโยชน์แก่ผู้บริโภคประกอบการตัดสินใจในการเลือกซื้อผลิตภัณฑ์เนื้อที่เหมะสมกับราคา

1.6 ลำดับการจัดเรียงวิทยานิพนธ์

เนื้อหาในวิทยานิพนธ์แบ่งออกเป็น 5 บท ได้แก่

บทที่ 1 บทนำ อธิบายถึงที่มาและความสำคัญของปัญหา วัตถุประสงค์ของงานวิจัย ขอบเขตงานวิจัย ประโยชน์ที่คาดว่าจะได้รับ และผลงานที่ได้รับการตีพิมพ์จากวิทยานิพนธ์

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

บทที่ 3 แนวคิดและวิธีการวิจัย

บทที่ 4 การทดลองและผลการทดลอง

บทที่ 5 สรุปผลการวิจัย และแนวทางการวิจัยในอนาคต

1.7 ผลงานที่ได้รับการตีพิมพ์

Panitchakorn, G. & Limpiyakorn, Y. (2021). Convolutional Neural Networks for Artificial Marbling Beef Classification. In 10th International Conference on Internet Computing for Science and Engineering. Guilin, China.



บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 เนื้อลายหินอ่อนเทียม (Artificial Marbling Beef)

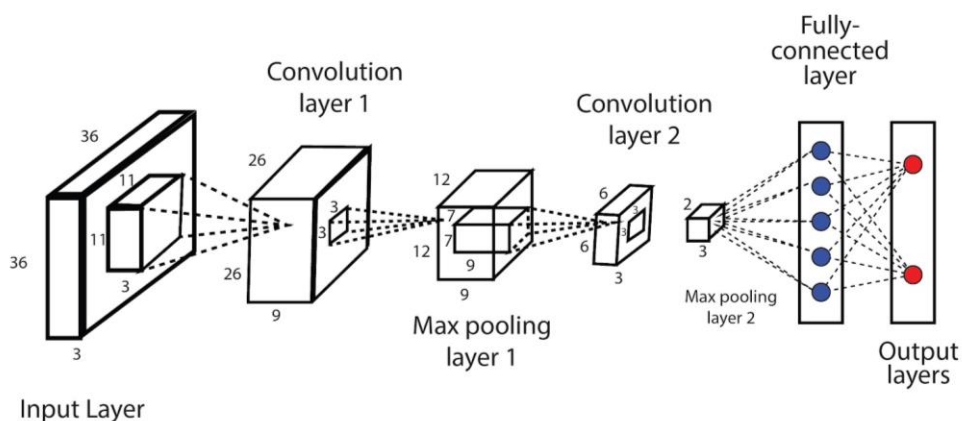
เนื้อลายหินอ่อนเทียม คือ เนื้อที่ผ่านเทคนิคกระบวนการผลิตอาหาร โดยเนื้อแดงจะถูกฉีดด้วยไขมัน ซึ่งอาจมาจากพืช สัตว์ หรือกรดไขมัน ผสมกับสารเติมแต่งต่าง ๆ ทั้งนี้ แต่ละบริษัทจะมี ส่วนประกอบและกระบวนการที่จำเพาะของตัวเอง เพื่อให้ไขมันสามารถฉีดผ่านเนื้อแดงได้ และ จำเป็นต้องรักษาคุณสมบัติของเหลวขณะทำการฉีดไว้ รวมทั้งทำให้อุณหภูมิลดลงเพื่อคงรูปไขมันที่ แทรกไปตามชั้นเนื้อ แล้วนำไปบรรจุใส่หีบห่อเพื่อจัดจำหน่ายต่อไป [3] ลักษณะทางกายภาพของเนื้อ ภายหลังการฉีดไขมันแล้วก็จะปรากฏลายหินอ่อนซึ่งมีความคล้ายกับลายหินอ่อนธรรมชาติ ดังรูปที่ 1 ฝั่งซ้าย [4] และเมื่อทำการฉีดไขมันจะปรากฏลายหินอ่อน ดังรูปที่ 1 ฝั่งขวา [5] ซึ่งเป็นการเพิ่ม ปัจจัยคุณภาพของเนื้อในกระบวนการตรวจสอบคุณภาพ เนื่องจากลายหินอ่อน หรือไขมันที่แทรกอยู่ ในเนื้อ เป็นหนึ่งในปัจจัยสำคัญของการกำหนดคุณภาพและราคาของผลิตภัณฑ์



รูปที่ 1 เนื้อลายหินอ่อนธรรมชาติ (ซ้าย) และเนื้อลายหินอ่อนเทียม (ขวา)

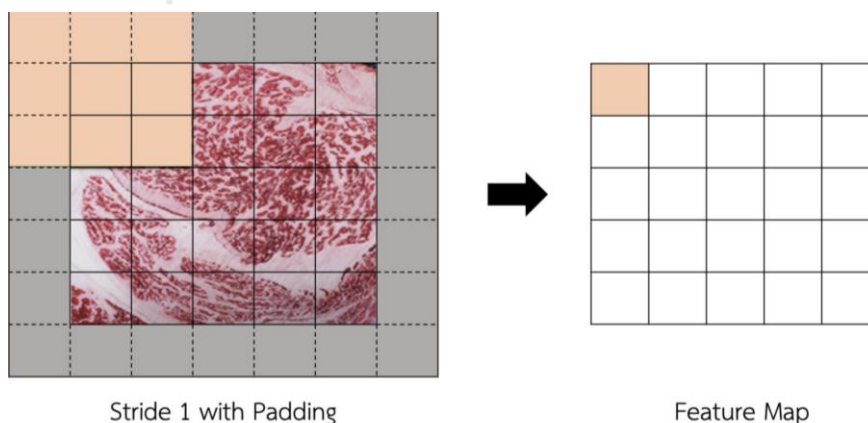
2.1.2 โครงข่ายประสาทคอนโวลูชัน (Convolutional Neural Network- CNN)

โครงข่ายประสาทคอนโวลูชันถูกพัฒนาขึ้นในปี 1998 [6] และมักใช้กับงานด้านการ ประมวลผลภาพ เนื่องจากโครงข่ายประสาทคอนโวลูชันสามารถแยกแยะคุณลักษณะของข้อมูลเข้าที่ มีหลายมิติได้เป็นอย่างดี โครงสร้างโครงข่ายประสาทเทียมแบบคอนโวลูชัน [7] ดังปรากฏในรูปที่ 2 ประกอบด้วย 3 ชั้น มีดังนี้



รูปที่ 2 โครงสร้างโครงข่ายประสาทคอนโวลูชัน

1. ชั้น Convolution (CovNets) เป็นชั้นนำข้อมูลภาพไปประมวลผลกับชุดของ Convolutional Filters ซึ่งจะดึงคุณลักษณะที่รับจากชั้นข้อมูลเข้าเกี่ยวข้องกับภาพ โดยแบ่งตามความกว้าง ความสูง และคุณลักษณะสีของภาพ เป็น $W \times W \times M$ โดย W คือ ความสูงหรือความกว้างของรูปภาพ และ M คือ มิติสีของรูปภาพ เช่น หากรูปภาพเป็นระบบ RGB ก็จะสามารถแบ่งออกเป็น 3 มิติ คือ สีแดง สีเขียว และสีน้ำเงิน โดยจะมีตัวกรอง (filter) หรือเคอเนล (kernel) จะทำการสแกนรูปภาพในแต่ละจุดของภาพเพื่อแยกองค์ประกอบของรูป เลื่อนต่อเนื่องจากจุดหนึ่งไปจนครบทุกพิกเซลในรูปภาพ โดยใช้ Stride เป็นตัวกำหนดการขั้นตอน (step) ในการเลื่อนตัวกรอง (filter) ได้เป็นผังคุณลักษณะ (feature map) ดังรูปที่ 3 นอกจากนี้หากรูปภาพจากชั้นข้อมูลนำเข้ามีองค์ประกอบที่สำคัญอยู่ในขอบภาพสามารถใช้ padding เพื่อเพิ่มพื้นที่กรอบให้รูปภาพนำไปประมวลผลในขั้นต่อไป

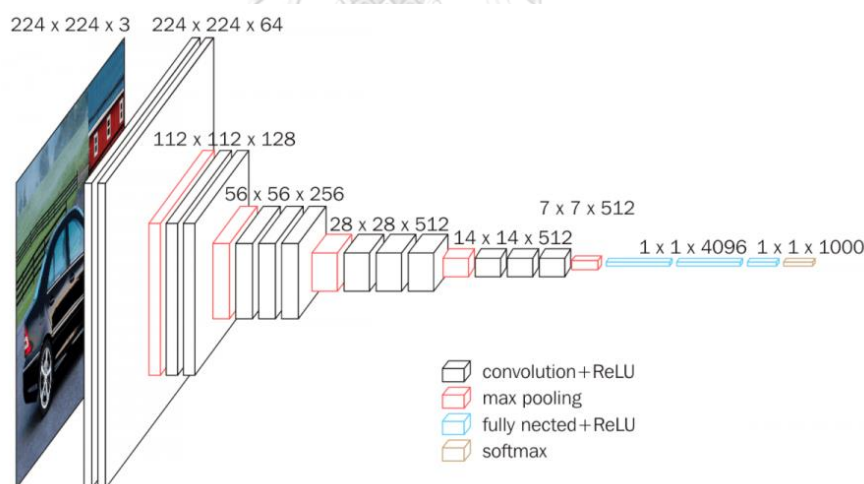


รูปที่ 3 การสกัดผังคุณลักษณะของรูปภาพในชั้น convolutional ที่มีการเพิ่ม padding และกำหนด stride เท่ากับ 1

2. ชั้น Pooling เป็นชั้นที่ลดขนาดของพารามิเตอร์จากชั้น ConvNets เพื่อลดความซับซ้อนของข้อมูล และดึงคุณลักษณะที่สำคัญของข้อมูล โดยผ่านตัวกรอง (filter) นำไปแปลงเป็นข้อมูลที่มีขนาดเล็กลง การพูลลิง (pooling) สามารถทำได้หลายรูปแบบ เช่น max pooling หรือ average pooling
3. ชั้น Fully-connected (FCN) เป็นชั้นที่นำเอาข้อมูลจากชั้นก่อนหน้าเชื่อมต่อกันทั้งหมด ให้ผลลัพธ์เป็น class ต่าง ๆ ตาม activation function ที่กำหนด

2.1.3 โครงข่ายประสาทแบบคอนโวลูชันแบบวีจีจี 16 (VGG16)

VGG-16 ชนะการแข่งขัน ImageNet ในปี 2014 [8] ซึ่งโจทย์ของการแข่งขันคือ object recognition ซึ่งต้องจำแนกวัตถุออกเป็นทั้งหมด 1,000 รูปแบบ หัวใจสำคัญที่ทำให้สถาปัตยกรรมนี้ชนะในการแข่งขันคือ ชั้น Softmax ในเลเยอร์สุดท้ายของโครงสร้าง VGG-16 ดังรูปที่ 4 [9] ซึ่งสามารถทำให้เวกเตอร์รูปภาพ ทำนายเป็นค่าความน่าจะเป็นของคลาสต่าง ๆ แต่ปัจจัยด้านขนาดของสถาปัตยกรรมนี้มีขนาดใหญ่ถึง 533MB ทำให้การนำแบบจำลองไปใช้ต่อค่อนข้างจำกัด อย่างไรก็ตามสมรรถนะของแบบจำลองก็ยังคงเป็นที่นิยมถูกนำมาใช้ในการจำแนกประเภทจนถึงปัจจุบัน

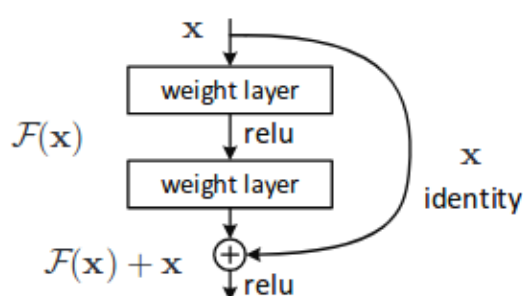


รูปที่ 4 โครงสร้างโครงข่ายประสาทคอนโวลูชันแบบ VGG-16

2.1.4 โครงข่ายประสาทเชิงลึกแบบ Residual Network (ResNet)

Deep Residual Network หรือ ResNet ถูกนำเสนอครั้งแรกในงานวิจัยที่ชื่อว่า Deep Residual Learning for Image Recognition [10] ซึ่งจำนวนชั้นของโครงข่ายก็จะถูกนำมาต่อท้ายชื่อ นั่นหมายถึงว่า ResNet-50 มีจำนวนโครงสร้างสร้างทั้งหมด 50 ชั้น และมี สถาปัตยกรรมนี้นำเสนอแนวคิดที่เรียกว่า Residual Network ซึ่งเป็นเทคนิคที่เรียกว่าข้ามการเชื่อมต่อ การเชื่อมต่อ

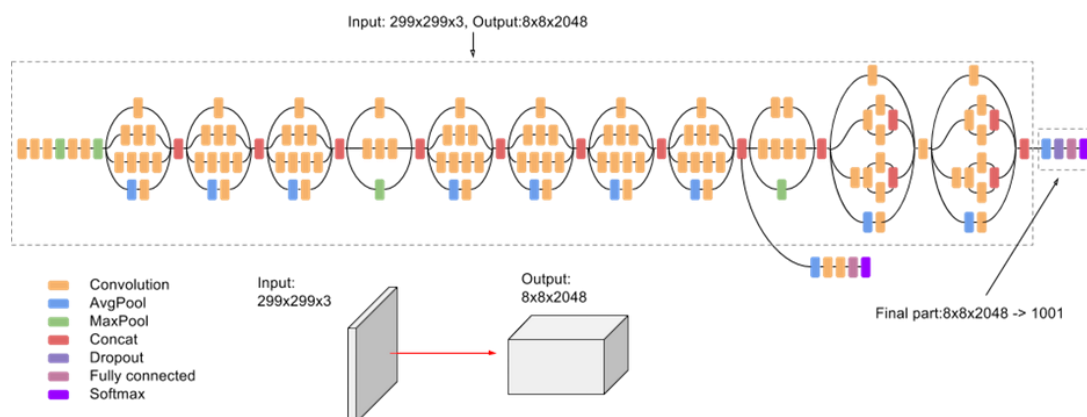
แบบข้ามจะข้ามการฝึกจากบางชั้นและเชื่อมต่อโดยตรงกับข้อมูลสัญญาณออก ดังรูปที่ 5 เพื่อแก้ปัญหา vanishing gradient ซึ่งเกิดขึ้นเมื่อเราฝึกสอนแบบโดยการปรับพารามิเตอร์เดิมซ้ำ ๆ ส่งผลให้การปรับพารามิเตอร์ในครั้งถัด ๆ ไป ค่า Loss จะเปลี่ยนแปลงน้อยลงมากจนไม่แบบจำลองไม่ Optimize หรืออาจจะต้องใช้จำนวนรอบในการฝึกสอนจำนวนหลายรอบ และมักจะทำให้เกิด exploding gradient ตามมาคือ Exploding gradients ซึ่งก็ส่งผลให้ Optimizer ไม่สามารถหาจุดที่ดีที่สุดของแบบจำลอง



รูปที่ 5 Residual block

2.1.5 โครงข่ายประสาทแบบ InceptionV3

สถาปัตยกรรม InceptionV3 ถูกพัฒนาโดยทีมจาก Google โดยเป็นรุ่นที่ 3 ในตระกูล Inception ชนะการแข่งขันสำหรับข้อมูลชุด ImageNet ในหัวข้อ ImageNet Large Scale Visual Recognition ในปี 2015 [11] ซึ่งข้อดีสำหรับแบบจำลองนี้คือจำนวนพารามิเตอร์ที่น้อย และโครงสร้างของสถาปัตยกรรมเพียง 42 ชั้น ดังรูปที่ 6 [12] ซึ่งเกิดขึ้นจากแนวความคิดหลักของสถาปัตยกรรมนี้ คือ factorization ซึ่งจะลดจำนวนของพารามิเตอร์ลง แต่ไม่ลดประสิทธิภาพของแบบจำลอง



รูปที่ 6 โครงสร้างโครงข่ายประสาทแบบ InceptionV3

สำหรับการเปรียบเทียบสมรรถนะของ pre-trained model ทั้ง VGG-16, ResNet-50 และ InceptionV3 ที่ใช้กับงานจำแนกประเภทของรูปภาพจากชุดข้อมูล ImageNet ได้ผลดังแสดงในตารางที่ 1

ตารางที่ 1 เปรียบเทียบสมรรถนะของแบบจำลอง VGG-16, ResNet-50 และ InceptionV3 จากการใช้ชุดข้อมูลรูปภาพ ImageNet ในการเรียนรู้ [13]

Model	Size	Parameters	Top-5 Accuracy
VGG16	528 MB	138,357,544	0.901
ResNet50	98 MB	25,636,712	0.921
InceptionV3	92 MB	23,851,784	0.937

2.1.6 การเรียนรู้ถ่ายโอน (Transfer Learning)

เนื่องจากการสอนแบบจำลองตั้งแต่เริ่มต้นจำเป็นต้องใช้ปริมาณข้อมูลจำนวนมาก เพื่อให้ได้แบบจำลองที่มีนัยทั่วไป (Generalization) นอกจากปริมาณของข้อมูล ยังมีปัจจัยเรื่องเวลาในการประมวลผลของข้อมูล และความซับซ้อนของแบบจำลอง เทคนิคการเรียนรู้ถ่ายโอนจึงถูกนำมาใช้เพื่อนำความรู้จากแบบจำลองที่ได้รับการฝึกสอน และทดสอบก่อนหน้า ซึ่งให้ผลลัพธ์การทำนายที่แม่นยำบนตัวอย่างการเรียนรู้ที่ใกล้เคียงกันสำหรับใช้เป็นส่วนหนึ่งของแบบจำลองที่สร้างขึ้นใหม่ เพื่อลดทั้งเวลาในการฝึกสอน และการดึงคุณลักษณะของแบบจำลองที่มีประสิทธิภาพมาประยุกต์ใช้กับแบบจำลองและชุดข้อมูลสอนใหม่ ซึ่งอาจจะมีขนาดใหญ่ หรือขนาดเล็กก็ได้ [14] อย่างไรก็ตาม การ

ประยุกต์ใช้ transfer learning มักจะต้องพิจารณาปัจจัยในการปรับค่าพารามิเตอร์ (fine-tuning) กับแบบจำลอง ให้เหมาะสมกับงานนั้น ๆ สำหรับความต้องการเบื้องต้นการใช้การเรียนรู้ถ่ายโอน คือแบบจำลองตั้งต้น (Pre-trained model) ดังนั้น ในการเลือก pre-trained model ที่เหมาะสมกับงานชุดใหม่จึงต้องมีการทดสอบเปรียบเทียบในเบื้องต้น ตัวอย่าง pre-trained model ที่มีประสิทธิภาพและเป็นที่ยอมรับใช้กับงานจำแนกประเภทรูปภาพ เช่น VGG-16, ResNet-50 และ InceptionV3

สำหรับขั้นตอนในการใช้การเรียนรู้ถ่ายโอนนั้น สามารถนำไปใช้ได้หลัก ๆ 3 ขั้นตอน คือ

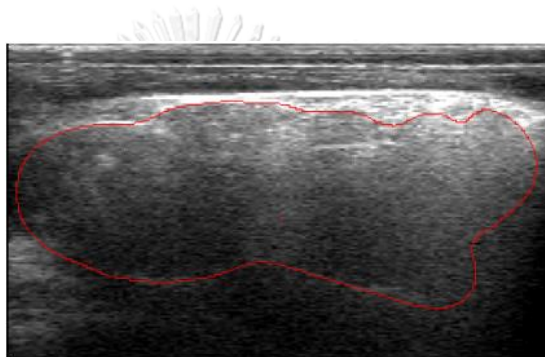
1. การเรียนรู้ถ่ายโอน (transfer learning)-ทำการนำเข้า pre-trained model และ weight จากสถาปัตยกรรมต่าง ๆ ที่ถูกฝึกสอนและทดสอบมาแล้ว โดยสามารถเลือกจากแบบจำลองที่มีสมรรถนะดี หรือสามารถพิจารณาเมื่อข้อมูลชุดใหม่มีลักษณะที่ใกล้เคียงกับลักษณะข้อมูลชุดเดิม หรือเป็นส่วนหนึ่งของข้อมูลเดิมที่ถูกฝึกสอนมา จากนั้นทำการตรึง (Freeze) ทุกชั้นของ convolution ของ pre-trained model
2. แยกคุณลักษณะเวกเตอร์ (extract feature)-ละทิ้งชั้น top model ซึ่งเป็นชั้นบนสุดของ pre-trained model และใช้ชั้นของ pre-trained model ในการคำนวณคุณลักษณะ pre-trained model และฝึกอบรมเฉพาะ fully-connection ใหม่ซึ่งเอามาแทนที่ top model ซึ่งปรับแต่งเองได้
3. การปรับละเอียด (Fine-tuning)-ทำการตรึง (Freeze) ในบางชั้นของ pre-trained model ที่ใกล้เคียงกับอินพุต และทำการฝึกชั้น convolutional ที่เหลือ ซึ่งมักฝึกสอนชั้นที่ใกล้เคียงกับชั้นผลลัพธ์ และ fully-connection

2.2 งานวิจัยที่เกี่ยวข้อง

จากการศึกษาค้นคว้างานวิจัยที่เกี่ยวข้องกับการคัดแยกคุณภาพด้วยรูปภาพของเนื้อในหัวข้อต่าง ๆ เช่น ระดับลายหินอ่อน หรือระดับไขมันในเนื้อ พบว่า มีการนำเสนอเทคนิคการเรียนรู้ที่หลากหลายแตกต่างกันออกไป ทั้งในแง่ของการเลือกแบบจำลองที่ใช้, การปรับพารามิเตอร์ที่เหมาะสม และวิธีวัดผลสมรรถนะของแบบจำลอง อีกทั้งยังมีวิธีการนำเสนอ transfer learning เมื่อใช้งานกับชุดข้อมูลรูปภาพใหม่ ไปจนถึงงานวิจัยที่ใช้ transfer learning กับชุดข้อมูลที่มีลักษณะใกล้เคียงกับชุดข้อมูลใหม่ที่รวบรวมขึ้นในงานวิจัยครั้งนี้ เพื่อทำการคัดเลือก pre-trained model และ pre-trained weight

2.2.1 งานวิจัยที่เกี่ยวข้องกับการคัดแยกคุณภาพเนื้อด้วยภาพ

Kim [15] นำเสนอการใช้ CNN เพื่อใช้ประมาณค่าไขมันแทรก (Intramuscular Fat Percentage: %IMF) ด้วยรูปภาพอัลตราซาวด์ของเนื้อวัวพันธุ์ฮันฮูของประเทศเกาหลี ดังรูปที่ 7 โดยมีการใช้ K-means ในการการคัดแยก %IMF ออกเป็น 10 รูปแบบ ใช้ภาพอัลตราซาวด์ร่วมกับการตีกรอบจุดที่สนใจในภาพ (region of interest-ROI) เพื่อทำการฝึกสอนแบบจำลอง CNNs ได้ค่าความแม่นยำของการประมาณค่าไขมันแทรกในเนื้อวัวฮันฮูเท่ากับ 98.2% และได้ค่า correlation coefficient เทียบระหว่างค่าปริมาณไขมันจริงและค่าประมาณปริมาณไขมันจากแบบจำลองเป็น 0.97



รูปที่ 7 ภาพอัลตราซาวด์ของเนื้อวัวพันธุ์ฮันฮูของประเทศเกาหลีร่วมกับการตีกรอบจุดที่สนใจในภาพ

Penning และคณะ [16] นำเสนอวิธีในการให้คะแนนคุณภาพซาก (carcass quality) โดยอัตโนมัติโดยใช้วิธีปัญญาประดิษฐ์ที่แตกต่างกัน โดยเปรียบเทียบการใช้การวิเคราะห์ภาพ และแบบที่สองใช้ Rapid Evaporative Ionization Mass Spectrometry ทั้งสองใช้เทคนิคการเรียนรู้ของเครื่องเพื่อเพิ่มความเร็วและความแม่นยำในการประเมินคุณภาพซาก ยกเว้นปัจจัยลายหินอ่อนในเนื้อ เปรียบเทียบวิธีเรียนรู้ทั้งหมด 8 อัลกอริทึมสำหรับให้คะแนนคุณภาพซาก ในปัจจัยต่าง ๆ ดังนี้ อัลกอริทึมที่ให้ความแม่นยำสูงสุดในด้านระดับคุณภาพคือ principal component analysis (PCA) ร่วมกับการใช้ feature selection และ linear discriminant analysis ได้รับความแม่นยำ 82% สำหรับการจำแนกแหล่งที่มา support vector machine-linear ได้รับความแม่นยำ 99% สำหรับการคัดแยกสายพันธุ์ support vector machine ได้ 85% ความนุ่มของเนื้อ XGBoost ได้รับความแม่นยำสูงสุดที่ 91%

Fitri และคณะ [17] นำเสนออัลกอริทึมการจำแนกที่ดีที่สุดสำหรับการระบุคุณภาพเนื้อตามลายหินอ่อน หรือชั้นไขมันแทรกในเนื้อ การศึกษานี้จำแนกคุณภาพของเนื้อลายหินอ่อนตามขนาดของลายหินอ่อน การจำแนกประเภทในการศึกษานี้ใช้อัลกอริทึม support vector machine, linear

discriminant analysis และ Decision Tree นอกจากนี้ ในขั้นตอนการทดสอบใช้ k-fold validation เพื่อให้ได้แบบจำลองที่ error estimator ดีที่สุด ชุดข้อมูลแบ่งเป็นข้อมูลการฝึกและทดสอบข้อมูลโดยใช้ 10-folds cross-validation ผลลัพธ์ของอัลกอริทึมที่ดีที่สุดในงานนี้ คือ Decision Tree ที่ความแม่นยำ 91.42%

Salsabila และคณะ [18] จำแนกประเภทเนื้อสัตว์จากเนื้อสัมผัส (Texture) โดยใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN) เพื่อแยกแยะเนื้อหมูและเนื้อวัวเนื่องจากสถานการณ์ราคาสูงและเกิดความสับสนเมื่อเลือกซื้อโดยไม่มีป้ายกำกับที่แน่ชัด มีการใช้เทคนิค regularization ร่วมด้วย ได้แก่ Dropout, L2 และ Max-Norm ถูกนำไปใช้กับโมเดลและเปรียบเทียบเพื่อให้ได้ผลลัพธ์การจำแนกประเภทเนื้อสัตว์ที่ดีที่สุด และทำนายข้อมูลใหม่ได้อย่างแม่นยำ ได้ว่าความแม่นยำสูงสุด 97.56% ที่ได้จากโมเดล CNN โดยใช้เทคนิค Dropout โดยใช้ 0.7 ที่ใช้ไฮเปอร์พารามิเตอร์ เป็น Adam's Optimizer, fully connected layer ขนาด 128, และใช้ activation function คือ ReLu

2.2.2 งานวิจัยที่ประยุกต์ใช้ transfer learning กับชุดข้อมูลภาพ

Ju และคณะ [19] ประยุกต์ใช้ CNN ร่วมกับ transfer learning เพื่อคัดแยกผลพุทราที่ได้รับ ความเสียหาย โดยแบ่งออกเป็น 5 รูปแบบตามลักษณะผลพุทรา คือ ปกติ, เน่า, แตก, เปลือกถูกปลอก และเป็นสีน้ำตาลอมแดง ดังรูปที่ 8



รูปที่ 8 ตัวอย่างรูปภาพพุทราตามลักษณะผลพุทรา

โดยชุดข้อมูลมีจำนวนน้อย เป็นลักษณะปัญหาแบบ fine-grained image classification (FGIC) เนื่องจากชุดข้อมูลแต่ละประเภทมีความแตกต่างเพียงเล็กน้อย จึงทำการนำองค์ความรู้จาก pre-trained model และ pre-trained weight จากการฝึกสอนบนข้อมูล ImageNet รวมถึงการสร้างแบบจำลองตระกูล ResNet เพื่อดึงคุณลักษณะผลพุทราที่ได้รับ ความเสียหาย โดยเปรียบเทียบค่าความแม่นยำ จากผลการทดลองพบว่าแบบจำลอง SE-ResNet50-CL ให้ความแม่นยำสูงถึง 94.15% ในการคัดแยกพุทรา

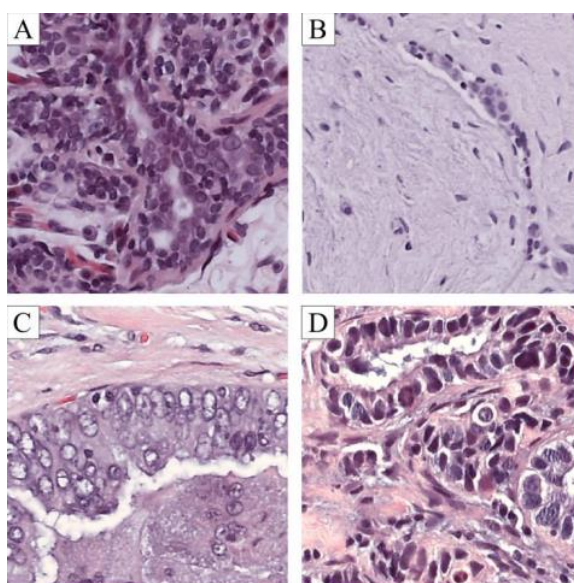
Özgenel และคณะ [20] ทำการศึกษาและเปรียบเทียบประสิทธิภาพของ Pre-trained model และ pre-trained weight ที่ถูกฝึกสอนบนชุดข้อมูล ImageNet เพื่อตรวจจับรอยร้าวในอาคารจากรูปภาพ ดังรูปที่ 9 ทำการวิเคราะห์ประสิทธิภาพของ AlexNet, VGG16, VGG19, GoogleNet, ResNet50, ResNet101 และ ResNet152 โดยเปรียบเทียบทั้งจำนวนชั้นของแบบจำลอง จำนวนของพารามิเตอร์ที่ถูกฝึกสอน รวมไปถึงเปรียบเทียบด้วยขนาดของชุดข้อมูลภาพที่ใช้ฝึกสอนและข้อมูลภาพทดสอบแบบจำลองที่ต่างกัน ตั้งแต่ 350, 700, 1750, 3500, 7000, 14000, 21000 ไปจนถึง 28000 ภาพ นำเสนอควบคู่กับจำนวน epoch ในการฝึกสอนแบบจำลอง จากผลการทดลองพบว่า AlexNet, VGG16, VGG19 และ GoogleNet ให้ความแม่นยำสูงกับชุดข้อมูลภาพรอยร้าวขนาด 350 ภาพ ในขณะที่ตระกูล ResNet ให้ประสิทธิภาพต่ำกับขนาดชุดข้อมูลที่จำกัดเมื่อฝึกสอนในรอบที่แรก แต่ทุกแบบจำลองให้ความแม่นยำมากกว่า 90% ในการฝึกสอนรอบที่สอง อย่างไรก็ตาม Pre-trained model ทุกแบบได้ประสิทธิภาพของแบบจำลองสูงกว่า 90% เมื่อชุดข้อมูลฝึกสอนมีขนาด 14000 ภาพเป็นต้นไป



รูปที่ 9 ตัวอย่างชุดข้อมูลฝึกสอน และชุดข้อมูลทดสอบในงานตรวจจับรอยร้าวในอาคารจากรูปภาพ

Hosny และคณะ [21] ได้ทำการจำแนกเซลล์มะเร็งเต้านมจากรูปภาพจากการตรวจวินิจฉัยโดยตัดชิ้นเนื้อตรวจทางพยาธิวิทยา (Histopathological) โดยใช้ transfer learning ในตระกูล AlexNet, GoogleNet, และ ResNet โดยใช้ชุดข้อมูลเนื้อเยื่อมะเร็งใหม่ที่มีขนาด 260 รูปภาพเท่านั้น แบ่งออกเป็น 240 รูปภาพสำหรับชุดข้อมูลฝึกสอน และ 20 รูปภาพสำหรับชุดข้อมูลทดสอบ ซึ่งมีตัวอย่างทั้งหมด 4 แบบ ดังรูปที่ 10 คือ A เนื้อเยื่อปกติ (Normal tissue), B ความผิดปกติที่ไม่

เป็นพิษเป็นภัย (Benign abnormality), C มะเร็งในแหล่งกำเนิด (In-situ carcinoma), D มะเร็งระยะลุกลาม (Invasive carcinoma) โดยแต่ละ pre-trained model จะถูกปรับปรุงขึ้น fully connected จาก 1000 ประเภทเนื่องจากใช้ pre-trained model จากการแข่งขันการจำแนกประเภทรูปภาพ ImageNet เหลือเพียง 4 ประเภท จากผลทดลองได้ว่า ResNet ได้รับความแม่นยำสูงถึง 85%

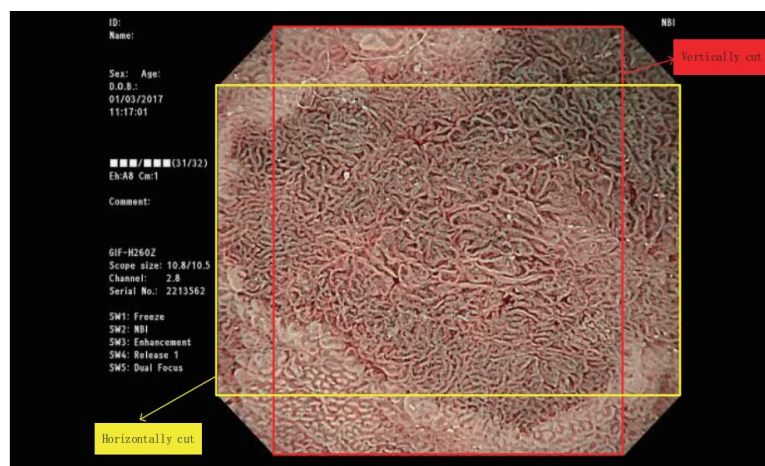


รูปที่ 10 ตัวอย่างชุดข้อมูลภาพเนื้อเยื่อทั้ง 4 แบบที่ใช้ในการจำแนกเซลล์มะเร็งเต้านมจากรูปภาพจากการตรวจวินิจฉัยโดยตัดชิ้นเนื้อตรวจทางพยาธิวิทยา

Demir และคณะ [22] นำเสนอการจำแนกเซลล์มะเร็งผิวหนัง 2 ประเภท คือ ปานที่ไม่อันตราย (benign) และ มะเร็งผิวหนัง (malignant) โดยใช้ pretrained model แบบ Resnet-101 และ Inception-v3 ที่ถูกฝึกสอนกับชุดข้อมูล ImageNet มาใช้ มีวัตถุประสงค์เพื่อตรวจจับเซลล์มะเร็งได้ตั้งแต่ระยะเริ่มต้น โดยใช้ชุดข้อมูลรูปภาพทั้งหมด 3297 รูปภาพ โดยแบ่งเป็นชุดข้อมูลฝึกสอน 2437 ภาพ ชุดข้อมูลทดสอบ 660 ภาพ และชุดข้อมูลตรวจสอบ 200 ภาพ พบว่า Resnet-101 ได้รับความแม่นยำ 84.09% ในขณะที่ Inception-v3 ได้รับความแม่นยำสูงถึง 87.42%

Liu และคณะ [23] นำเสนอการใช้การเรียนรู้ถ่ายโอนร่วมกับโครงข่ายประสาทเทียมแบบคอนโวลูชัน เพื่อการจำแนกมะเร็งกระเพาะอาหารในระยะแรกด้วยชุดข้อมูลรูปภาพจากเทคโนโลยีส่องกล้องแบบ M-NBI (Magnifying Narrow Band Imaging) ดังรูปที่ 11 จากแบบจำลองการเรียนรู้ถ่ายโอน 3 แบบ คือ VGG16, InceptionV3 และ InceptionResNetV2 อีกทั้งยังนำเสนอผลกระทบของ

การปรับละเอียดโครงข่ายในชั้นต่าง ๆ 3 รูปแบบด้วยกัน ได้แก่ ละทิ้งการตรึงทั้งหมด ละทิ้งการตรึงครึ่งหนึ่ง และ ไม่ทำการละทิ้งการตรึง โดยวัดสมรรถนะจากแบบจำลองการเรียนรู้ถ่ายโอนที่ให้สมรรถนะที่ดีที่สุดจาก 3 ตัววัดด้วยกัน คือ accuracy, sensitivity และ specificity พบว่าแบบจำลองการเรียนรู้ถ่ายโอนแบบ InceptionV3 ให้สมรรถนะที่ดีที่สุดเมื่อละทิ้งการตรึงทั้งหมดทุกชั้น ดังตารางที่ 2 หลังจากการปรับละเอียดในขั้นตอนละทิ้งการตรึง ได้ค่า accuracy, sensitivity และ specificity เป็น 0.985, 0.981 และ 0.989 ตามลำดับ ดังแสดงในตารางที่ 2



รูปที่ 11 ตัวอย่างของการตัดลองวิธี: ตัดในแนวนอนและแนวตั้งจากภาพ M-NBI

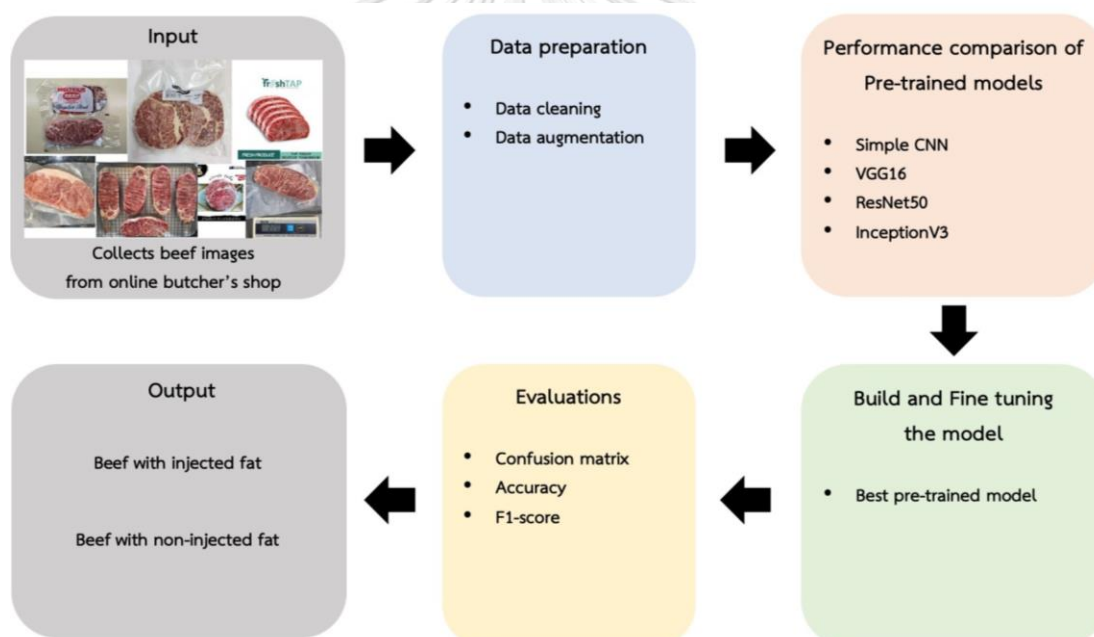
ตารางที่ 2 ผลลัพธ์ของ InceptionV3 ที่ปรับแต่งอย่างละเอียดในชั้นต่างๆ

Fine-tuning	Accuracy	Sensitivity	Specificity
Unfrozen	0.985	0.981	0.989
Half frozen	0.962	0.953	0.969
Total frozen	0.930	0.910	0.947

บทที่ 3

แนวคิดและวิธีการวิจัย

งานวิจัยนี้นำเสนอการสร้างแบบจำลองด้วยโครงข่ายประสาทคอนโวลูชันร่วมกับเทคนิคการเรียนรู้ถ่ายโอน เพื่อการจำแนกประเภทภาพเนื้อวัวลายหินอ่อนออกเป็น 2 ประเภท คือ 1) ภาพเนื้อวัวลายหินอ่อนเทียมที่ได้จากการฉีดไขมัน และ 2) ภาพเนื้อวัวลายหินอ่อนที่ไม่ได้ฉีดไขมัน ลำดับของการวิจัยแบ่งเป็น 4 ขั้นตอนหลัก ดังแสดงในรูปที่ 12 ประกอบด้วย 1) การเตรียมข้อมูลรูปภาพที่นำมาใช้ในงานวิจัย 2) การเปรียบเทียบแบบจำลองการเรียนรู้ถ่ายโอนเพื่อใช้สำหรับชุดข้อมูลที่สร้างขึ้นใหม่ 3) การปรับแต่งสมรรถนะแบบจำลอง และ 4) การประเมินสมรรถนะของแบบจำลองการจำแนกประเภทภาพเนื้อวัวลายหินอ่อน



รูปที่ 12 ขั้นตอนวิธีวิจัยการเรียนรู้ถ่ายโอนสำหรับการจำแนกประเภทภาพเนื้อวัวลายหินอ่อนเทียม

3.1 การเตรียมชุดข้อมูล

เก็บรวบรวมภาพเนื้อวัวลายหินอ่อน แบ่งออกเป็นเนื้อวัวที่มีการฉีดไขมัน และไม่ได้ฉีดไขมัน จากร้านค้าออนไลน์ที่น่าเชื่อถือ โดยคัดเลือกภาพที่มีการกำหนดแหล่งที่มาของเนื้อ และมีการกำหนดระดับไขมันอย่างชัดเจนตามมาตรฐานคุณภาพเนื้อวัว ดังแสดงในรูปที่ 13



รูปที่ 13 ตัวอย่างรูปเนื้อวัวชนิดไขมันจากร้านค้าออนไลน์ที่น่าเชื่อถือ

การเตรียมชุดข้อมูลประมวลผลก่อน (data preprocessing) ประกอบไปด้วย 2 ส่วนด้วยกัน คือ

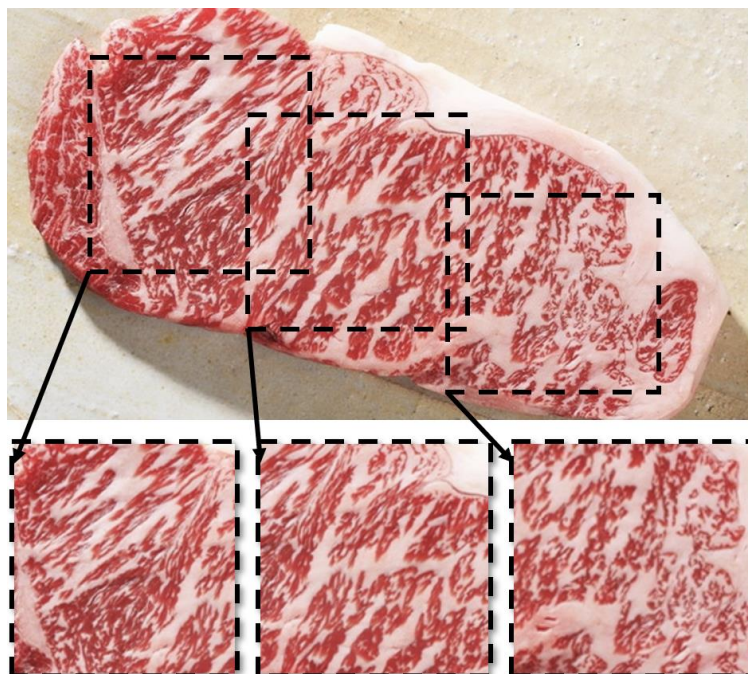
1) ทำความสะอาดข้อมูล (data cleaning) ตัดชิ้นส่วนของภาพเนื้อที่เลือก คัดเลือกส่วนของเนื้อที่แสดงลายไขมันที่ไม่ติดองค์ประกอบอื่น เช่น ป้ายยี่ห้อ ตัวอักษรในภาพ หรือไขมันส่วนนอกของชิ้นเนื้อ ดังตัวอย่างรูปที่ 14



รูปที่ 14 การทำความสะอาดข้อมูลรูปภาพเนื้อ

2) การเพิ่มจำนวนชุดข้อมูล (data augmentation) เพิ่มจำนวนชุดข้อมูลสำหรับชุดข้อมูลฝึกสอนแบบจำลอง ในส่วนการของเพิ่มจะนวนข้อมูลทดสอบจะถูกแบ่งออกเป็น 2 ส่วน คือ แบบใช้มนุษย์ และแบบใช้ไลบรารีอัตโนมัติ

เพิ่มจำนวนชุดข้อมูลแบบใช้มนุษย์ จะทำการเพิ่มจำนวนภาพจากภาพเนื้อชิ้นเดียวกัน หากชิ้นเนื้อที่มีขนาดใหญ่จะทำการแบ่งรูปเนื้อวัวได้มากกว่า 1 รูปภาพ ดังรูปที่ 15



รูปที่ 15 เพิ่มจำนวนชุดข้อมูลรูปภาพเนื้อแบบใช้มนุษย์

เพิ่มจำนวนชุดข้อมูลแบบใช้ไลบรารีอัตโนมัติ จะทำการ flip, zoom รวมถึง re-scale รูปภาพให้ได้ขนาดข้อมูลนำเข้าสำหรับชั้น input เตรียมไว้สำหรับโครงสร้าง Pre-trained model ทั้ง 3 แบบ คือ VGG-16, ResNet-50 และ Inceptionv3

ในการฝึกสอนและทดสอบแบบจำลอง จะทำการแบ่งข้อมูลออกเป็น ข้อมูลสอน ข้อมูลตรวจสอบ และข้อมูลทดสอบ ดังตารางที่ 3

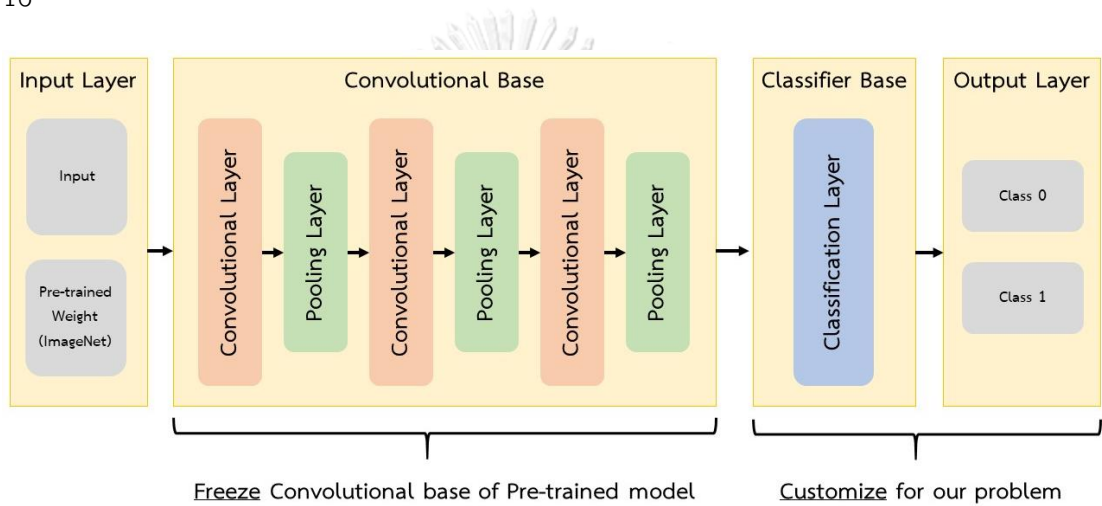
ตารางที่ 3 สรุปจำนวนข้อมูลสอน ข้อมูลตรวจสอบ ข้อมูลทดสอบของเนื้อทั้งสองประเภท

ประเภทรูปภาพ (image class)	ข้อมูลสอน (training data)	ข้อมูลตรวจสอบ (validation data)	ข้อมูลทดสอบ (test data)	จำนวน ทั้งหมด
เนื้อฉีดไขมัน	175	40	50	265
เนื้อไม่ได้ฉีดไขมัน	175	40	50	265
รวม	350	80	100	530

3.2 การสร้างแบบจำลองเพื่อการเปรียบเทียบแบบจำลองการเรียนรู้ถ่ายโอน

ทำการสร้าง baseline model ด้วยโครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN) ด้วยการฝึกสอนแบบจำลองตั้งแต่เริ่มต้นโดยใช้แบบชุดข้อมูลฝึกสอน และชุดข้อมูลตรวจสอบ และทำการทดสอบแบบจำลองการเรียนรู้ถ่ายโอนทั้ง 3 แบบ คือ VGG-16, ResNet-50 และ InceptionV3 โดยใช้ pre-trained weight จาก ImageNet และทำการละเว้นการตรึงชั้น classifier หรือ top model จากนั้นตรึงชั้น convolution ทั้งหมดของ pre-trained model ร่วมกับการเปลี่ยนชั้น classifier ให้เหมาะสมกับการจำแนกแบบไบนารี โครงสร้างแบบจำลองร่วมกับการเรียนรู้ถ่ายโอน แสดงดังรูปที่

16



รูปที่ 16 โครงสร้างแบบจำลองร่วมกับการเรียนรู้การถ่ายโอน

จากนั้นทำการบันทึกสมรรถนะของแบบจำลองทั้ง 4 แบบ คือ CNN, VGG-16, ResNet-50 และ InceptionV3

3.3 การวัดสมรรถนะของแบบจำลอง

การวัดสมรรถนะของแบบจำลองในงานวิจัยนี้ จะใช้ค่าความแม่นยำของชุดข้อมูลฝึก ชุดข้อมูลตรวจสอบ โดยใช้ค่าความแม่นยำที่ดีที่สุดขณะที่ฝึกสอนแบบจำลอง ประกอบกับ epoch ที่ได้ค่าความแม่นยำที่ดีที่สุด สำหรับการวัดสมรรถนะแบบจำลองบนข้อมูลทดสอบนั้น จะใช้ตัววัดปกติมาตรฐานทั่วไป คือ confusion matrix จากนั้นนำไปคำนวณค่าความแม่นยำ ค่าความเที่ยงตรง ค่าความครบถ้วน และค่าเอฟวัน

confusion matrix ประกอบด้วยค่าตัววัดฐาน 4 ตัว ดังตารางที่ 4 ประกอบด้วย

- 1) True Positive (TP) คือ ผลลัพธ์ที่แบบจำลองทำนายภาพเนื้อฉืดไขมันได้อย่างถูกต้อง
- 2) True Negative (TN) คือ ผลลัพธ์ที่แบบจำลองทำนายเนื้อไม่ฉืดไขมันได้อย่างถูกต้อง
- 3) False Positive (FP) คือ ผลลัพธ์ที่แบบจำลองทำนายเนื้อฉืดไขมันเป็นเท็จ
- 4) False Negative (FN) คือ ผลลัพธ์ที่แบบจำลองทำนายเนื้อไม่ฉืดไขมันเป็นเท็จ

ตารางที่ 4 confusion matrix สำหรับการจำแนกเนื้อลายหินอ่อนเทียม

		ผลทำนาย	
		เนื้อฉืดไขมัน	เนื้อไม่ฉืดไขมัน
ป้ายภาพจริง	เนื้อฉืดไขมัน	True Positive (TP)	False Negative (FN)
	เนื้อไม่ฉืดไขมัน	False Positive (FP)	True Negative (TN)

ค่าตัววัดสมรรถนะของแบบจำลองสามารถคำนวณได้จากค่าต่างๆใน confusion matrix ประกอบด้วย

- 1) ค่าความแม่นยำ (Accuracy) สามารถคำนวณได้จากสมการที่ 1

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

- 2) ค่าความเที่ยงตรง (Precision) สามารถคำนวณได้จากสมการที่ 2

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

- 3) ค่าความครบถ้วน (Recall) สามารถคำนวณได้จากสมการที่ 3

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

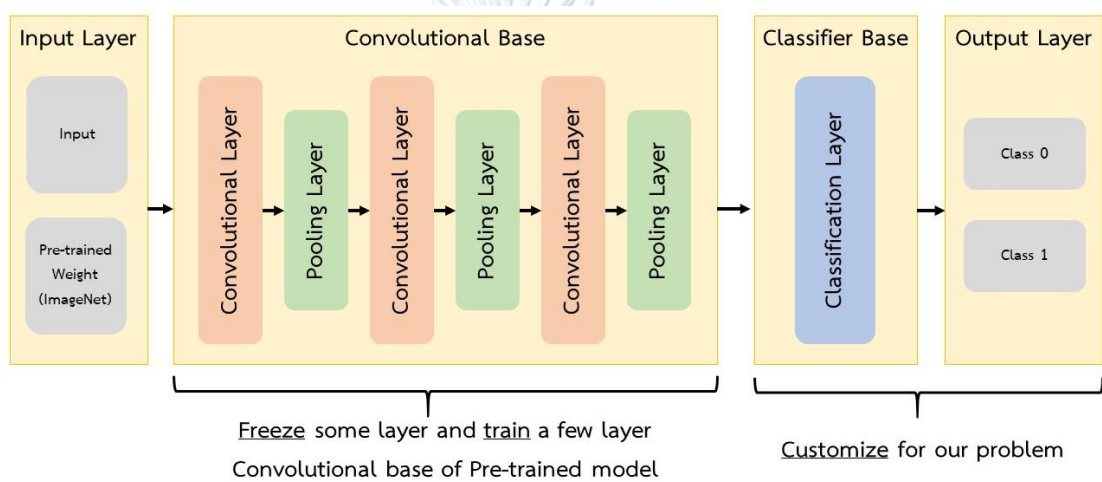
- 4) ค่าเอฟวัน (F1-score) สามารถคำนวณได้จากสมการที่ 4

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

3.4 การปรับละเอียดสมรรถนะของแบบจำลอง

3.4.1 ปรับละเอียดจำนวนชั้นของแบบจำลอง

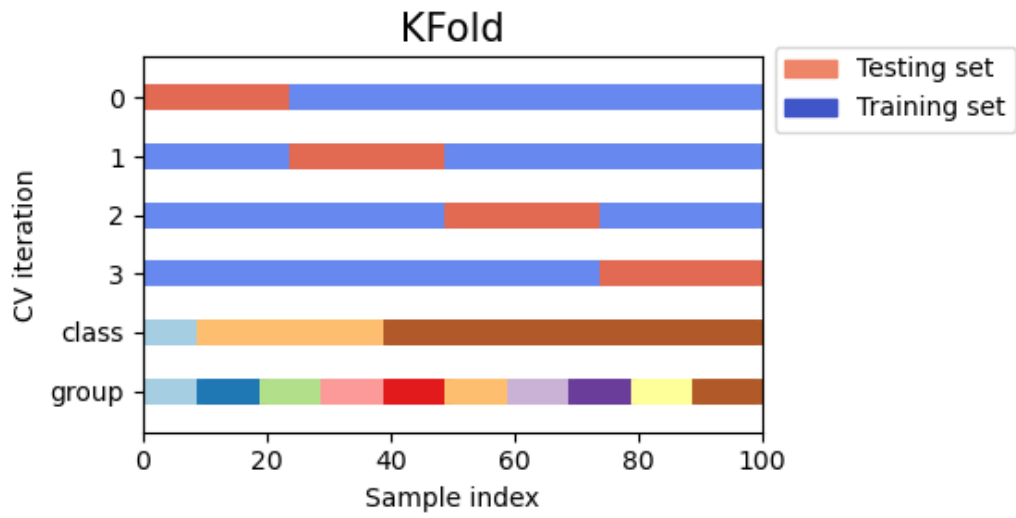
ทำการสร้างแบบจำลองโครงข่ายประสาทคอนโวลูชัน ร่วมกับ Pre-trained model ที่ได้สมรรถนะที่ดีที่สุดกับชุดข้อมูลเนื่อววลายหินอ่อน ร่วมกับการปรับละเอียดแบบจำลอง คือ ทำการปรับชั้น convolution base และทำการละทิ้งการตรึงในบางชั้น โดยจะทำการตรึง (Freeze) ในบางชั้นของ pre-trained model ที่ใกล้กับชั้นข้อมูลเข้า และทำการฝึกชั้น convolutional ที่เหลือ ซึ่งมักฝึกสอนชั้นที่ใกล้กับชั้นผลลัพธ์ และ fully-connection แล้วทำการบันทึกผลว่าการละทิ้งชั้นใดที่ยังคงดึงคุณลักษณะของชุดข้อมูลรูปภาพเก่า นำมาใช้กับชุดข้อมูลฝึกสอนเข้าใหม่ได้ ดังรูปที่ 17



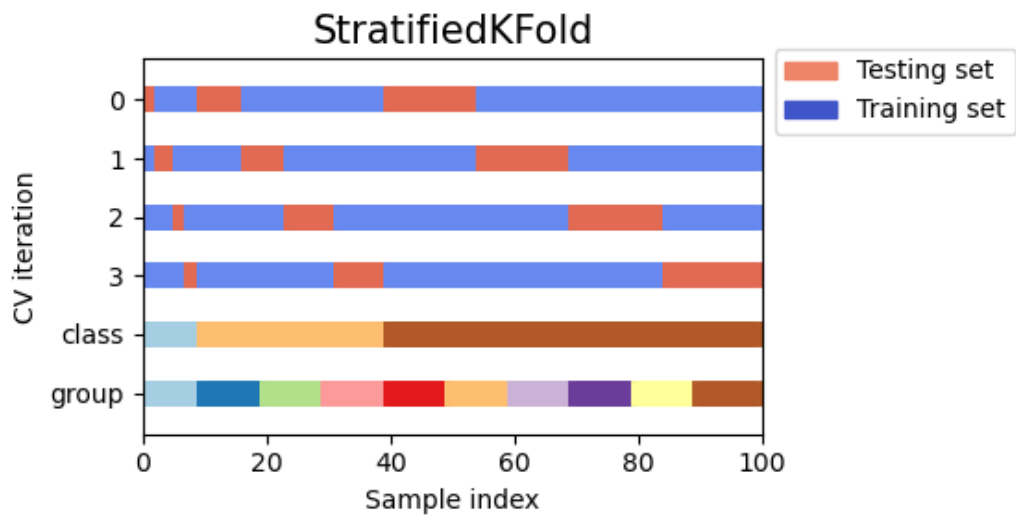
รูปที่ 17 โครงสร้างแบบจำลองร่วมกับการเรียนรู้การถ่ายโอนร่วมกับการปรับละเอียด

3.4.2 Stratified K-Fold Cross Validation (K=10)

อีกหนึ่งเทคนิคที่สามารถปรับปรุงสมรรถนะของแบบจำลอง เมื่อชุดข้อมูลมีจำนวนจำกัด คือ การใช้ 10-Fold Cross Validation เพื่อปรับปรุงสมรรถนะแบบจำลอง และลด bias ของแบบจำลอง โดยที่จะทำการแบ่งข้อมูลออกเป็น 10 ส่วน โดยในแต่ละส่วนของข้อมูลจะถูกแบ่งออกเป็น 2 ส่วนย่อย คือ ชุดข้อมูลฝึกสอน และ ชุดข้อมูลทดสอบ โดยทำการวนทดสอบจนครบ 10 รอบ แล้วนำมาสรุปผลสมรรถนะแบบจำลองโดยเฉลี่ย แต่เนื่องจากการแบ่งข้อมูลแบบ K-Fold Cross Validation ทัวไปอาจเกิดปัญหาที่ชุดข้อมูลที่ถูกแบ่งเพื่อนำมาฝึกสอนนั้นขณะรอบนั้น ๆ มีความ bias ในคลาสใดคลาสหนึ่ง ดังรูปที่ 18 ซึ่งการใช้ Stratified K-Fold Cross Validation จึงถูกนำมาใช้เพื่อแก้ปัญหาดังกล่าว เนื่องจากข้อมูลในแต่ละคลาสจะถูกแบ่งตามสัดส่วนแต่ละคลาสใน 1 รอบของการแบ่งข้อมูล ดังรูปที่ 19



รูปที่ 18 การแบ่งข้อมูลแบบ K-Fold Cross Validation



รูปที่ 19 การแบ่งข้อมูลแบบ Stratified K-Fold Cross Validation

บทที่ 4

การทดลองและผลการทดลอง

งานวิจัยนี้นำเสนอแนวทางการจำแนกข้อมูลภาพด้วยโครงข่ายประสาทเทียมคอนโวลูชัน ร่วมกับการเรียนรู้ถ่ายโอนทั้งหมด 3 แบบ คือ VGG16, ResNet50 และ InceptionV3 เปรียบเทียบกับการใช้โครงข่ายประสาทเทียมคอนโวลูชันปกติที่ต้องฝึกสอนตั้งแต่เริ่มต้น ขั้นตอนการทดลอง อธิบายในหัวข้อต่อไปนี้

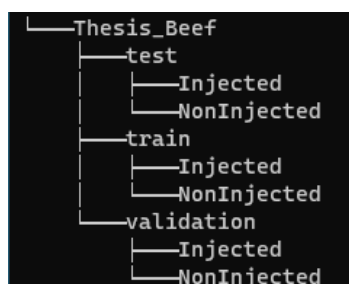
4.1 สภาพแวดล้อมที่ใช้ในการทดลอง

ใช้ Google Colab ซึ่งเป็นบริการแบบ Software as a Service (SaaS) โดยมีโฮสต์ โปรแกรมเป็น Jupyter Notebook ทำงานบน Cloud เป็นบริการที่อำนวยความสะดวกการพัฒนา ซอฟต์แวร์เป็นอย่างมาก เนื่องจากผู้พัฒนาสามารถดึงทรัพยากรบน Google Cloud มาใช้ในการ ประมวลผลได้ โดยใช้บริการแบบมีค่าใช้จ่าย (Google Colab Pro) ซึ่งสามารถใช้ (Graphic Processing Unit) ซึ่งเป็นหน่วยประมวลผลเกี่ยวกับกราฟิกโดยเฉพาะ และสามารถใช้นหน่วยความจำ แรมขนาด 25 กิกะไบต์ในโหมด High-RAM

ใช้ภาษา Python ในการพัฒนาแบบจำลอง ที่สามารถใช้งาน Google Colab ร่วมกับการ ใช้ไลบรารีจาก keras, ski-learn และ matlab

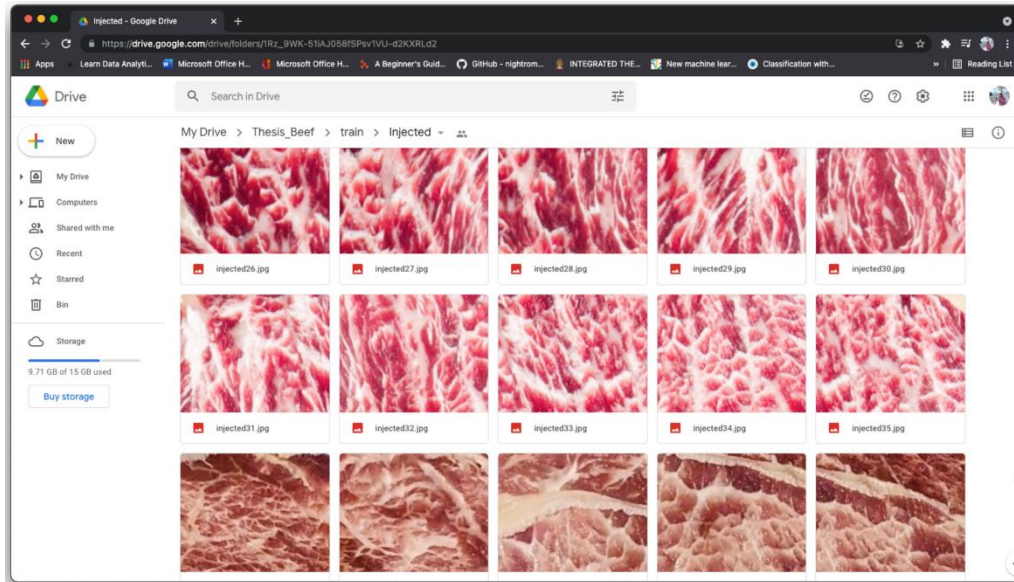
4.2 การเตรียมชุดข้อมูลที่ใช้ในการทดลอง

ทำการอัปโหลดชุดข้อมูลฝึกสอน ชุดข้อมูลตรวจสอบ และชุดข้อมูลทดสอบ ที่อยู่ภายใต้ โฟลเดอร์ Thesis_Beef สู่ Google Drive ซึ่งเป็นบริการเก็บข้อมูลแบบ cloud สามารถเข้าถึงชุด ข้อมูลได้จากทุกที่ โดยมีโครงสร้างโฟลเดอร์ ดังรูปที่ 20



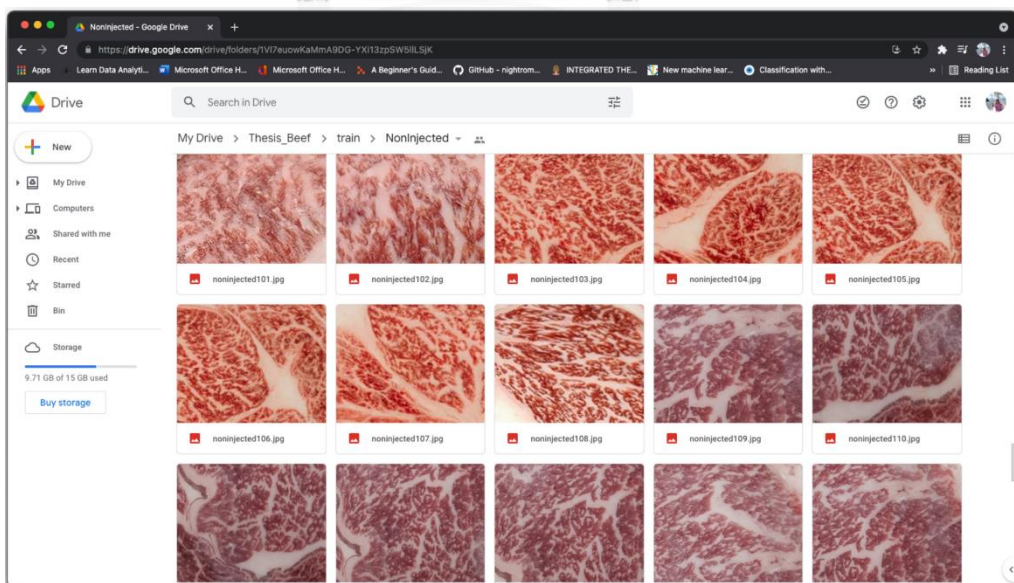
รูปที่ 20 โครงสร้างโฟลเดอร์ชุดข้อมูลที่ใช้ในการจำแนกประเภทภาพเนื้อลายหินอ่อนเทียม

สำหรับชุดข้อมูลเนื้อลายหินอ่อนเทียมอยู่ในโฟลเดอร์ที่มีชื่อว่า Injected เมื่อทำการดึงข้อมูลเพื่อใช้ในการฝึกสอนจะอยู่ในไดเรกทอรี ~/My Drive/Thesis_Beef/train/Injected ดังรูปที่ 21



รูปที่ 21 ตัวอย่างภาพข้อมูลฝึกสอนโมเดลประเภทเนื้อลายหินอ่อนเทียมใน google drive

สำหรับชุดข้อมูลเนื้อที่ไม่ใช่ลายหินอ่อนเทียมอยู่ในโฟลเดอร์ที่มีชื่อว่า NonInjected เมื่อทำการดึงข้อมูลเพื่อใช้ในการฝึกสอนอยู่ในไดเรกทอรี ~/My Drive/Thesis_Beef/train/NonInjected ดังรูปที่ 22



รูปที่ 22 ตัวอย่างภาพข้อมูลฝึกสอนโมเดลประเภทเนื้อที่ไม่ใช่ลายหินอ่อนเทียมใน google drive

ในขั้นตอนการเรียกใช้รูปภาพจาก Google drive จำเป็นต้องมีการ Mount drive เพื่อดึงข้อมูลมาใช้ใน google colab ใช้คำสั่ง drive.mount('/content/drive') ภายใต้ไลบรารี google.colab และทำการเปลี่ยนไดเรกทอรีไปยังพื้นที่ที่มีชุดข้อมูล ด้วยคำสั่ง %cd ~dir ดังรูปที่ 23

```

▶ from google.colab import drive
  drive.mount('/content/drive')

#Thesis Path : /content/drive/My Drive/Thesis Beef
%cd /content/drive/My Drive

↳ Mounted at /content/drive
  /content/drive/My Drive

```

รูปที่ 23 ชุดคำสั่งในการเชื่อมต่อชุดข้อมูลใน google drive กับ google colab

สำหรับการนำเข้าสู่ข้อมูลที่อยู่ในไดเรกทอรีของ google drive ที่ถูกแบ่งเป็นชุดข้อมูลฝึกสอน ชุดข้อมูลตรวจสอบ และชุดข้อมูลทดสอบนั้น จะอยู่ใน train, validation และ test ตามลำดับ ดังแสดงในรูปที่ 24 เนื่องจากต้องการกำหนดให้ชุดข้อมูลการจำแนกเนื้อลายหินอ่อนเทียมเพื่อการเปรียบเทียบสมรรถนะของแบบจำลองพื้นฐาน (baseline model) กับแบบจำลองถ่ายโอนการเรียนรู้ ทั้ง 3 ชนิด เป็นข้อมูลและรูปภาพเนื้อชุดเดียวกัน

```

[ ] base_dir = os.getcwd() + '/Thesis_Beef'
    train_dir = os.path.join(base_dir, 'train')
    validation_dir = os.path.join(base_dir, 'validation')
    test_dir = os.path.join(base_dir, 'test')

# Directory with our training injected fat pictures
train_inj_dir = os.path.join(train_dir, 'Injected')

# Directory with our training non-injected fat pictures
train_ninj_dir = os.path.join(train_dir, 'NonInjected')

# Directory with our validation injected fat pictures
validation_inj_dir = os.path.join(validation_dir, 'Injected')

# Directory with our validation non-injected fat pictures
validation_ninj_dir = os.path.join(validation_dir, 'NonInjected')

# Directory with our validation injected fat pictures
test_inj_dir = os.path.join(test_dir, 'Injected')

# Directory with our validation non-injected fat pictures
test_ninj_dir = os.path.join(test_dir, 'NonInjected')

```

รูปที่ 24 การกำหนดไดเรกทอรีเพื่อนำเข้าสู่ข้อมูลฝึกสอน ข้อมูลตรวจสอบและข้อมูลทดสอบ

เนื่องจากชุดข้อมูลมีขนาดเล็ก จึงใช้เทคนิคการเพิ่มจำนวนข้อมูล (data augmentation) ดังรูปที่ 25 สำหรับชุดข้อมูลที่ใช้ในการฝึกสอนแบบจำลอง (training dataset) ด้วยการ rescale, zoom และ flip โดยมีการกำหนดค่าให้พารามิเตอร์ ดังนี้

- ทำการ Normalized ข้อมูลแต่ละพิกเซลของรูปภาพให้อยู่ช่วง 0 ถึง 1 ด้วยพารามิเตอร์ (Rescale = 1./255)
- ซูมอินพุต (zoom_range = 0.2)
- พลิกอินพุตในแนวนอน (horizontal_flip = True)
- พลิกอินพุตในแนวตั้ง (vertical_flip = True)

สำหรับชุดข้อมูลตรวจสอบ (validation dataset) และชุดข้อมูลทดสอบ (test dataset) จะทำการ Normalized ข้อมูลแต่ละพิกเซลของรูปภาพให้อยู่ช่วง 0 ถึง 1 ด้วยพารามิเตอร์ เท่านั้น หลังจากนั้นทำการกำหนด flow ของแต่ละชุดโดยมีการกำหนด path ของชุดข้อมูลต่าง ๆ กำหนดขนาดของรูปภาพเป็น 150x150 และกำหนด batch_size เท่ากับ 5 ซึ่งคือจำนวนข้อมูลที่น่าไปใช้ในการฝึกสอนแต่ละรอบ

```
# Adding rescale, zoom_range, vertical_flip, and horizontal flip
train_datagen = ImageDataGenerator(
    rescale=1./255,
    zoom_range=0.2,
    horizontal_flip=True,
    vertical_flip=True)

# validation data not augmented
val_datagen = ImageDataGenerator(rescale=1./255)

test_datagen = ImageDataGenerator(rescale=1./255)

# Flow training images in batches of 5 using train_datagen generator
train_generator = train_datagen.flow_from_directory(
    train_dir,
    target_size=(IMG_SIZE, IMG_SIZE),
    batch_size=BATCH,
    class_mode='binary')

# Flow validation images in batches of 5 using val_datagen generator
validation_generator = val_datagen.flow_from_directory(
    validation_dir,
    target_size=(IMG_SIZE, IMG_SIZE),
    batch_size=BATCH,
    class_mode='binary')

# Flow test images in batches of 5 using val_datagen generator
test_generator = test_datagen.flow_from_directory(
    test_dir,
    target_size=(IMG_SIZE, IMG_SIZE),
    batch_size=BATCH,
    class_mode='binary',
    shuffle = False)
```

รูปที่ 25 การเพิ่มจำนวนข้อมูล (data augmentation) สำหรับชุดข้อมูลที่ใช้ในการทดลอง

4.3 การดำเนินการสร้างแบบจำลอง

ทำการนำเข้าไลบรารีที่จำเป็นต่อการสร้างแบบจำลอง จาก keras ดังรูปที่ 26 ทั้งนี้จะนำมาใช้ในการสร้างแบบจำลองแบบจำลองโครงข่ายประสาทคอนโวลูชัน ที่ถูกฝึกสอนใหม่ตั้งแต่เริ่มต้น และนำไปใช้สร้างชั้น Fully-connected (FCN) สำหรับจำแนกประเภทเนื้อลายหินอ่อนเทียมให้กับแบบจำลองเรียนรู้ถ่ายโอน

```
from tensorflow.keras import Model, layers
from tensorflow.keras.optimizers import Adam
```

รูปที่ 26 ไลบรารีที่จำเป็นต่อการสร้างแบบจำลอง

4.3.1 แบบจำลองโครงข่ายประสาทแบบคอนโวลูชัน (Convolutional Neural Network-CNN)

แบบจำลองโครงข่ายประสาทแบบคอนโวลูชัน รับข้อมูลนำเข้าขนาด $150 \times 150 \times 3$ ตามความกว้าง ความสูง และจากชั้นสีแบบ RGB ของรูปภาพ จากชั้นนำเข้าชั้น convolution แบบ conv2d มีตัวกรอง (filter) ขนาด 3×3 เชื่อมต่อกับชั้น pooling แบบ max_pooling2D ขนาด 2×2 จำนวนทั้งหมด 3 ชั้น โดยแต่ละชั้นของ convolution มี 16, 32 และ 64 layer ตามลำดับ กำหนด strides ในการเลื่อนตัวกรองเป็น เลื่อนทีละ 1 และไม่ทำการเพิ่ม padding จากนั้น ใช้ activation function เป็น ReLU ที่มีขอบเขตตั้งแต่ 0 ถึง x โดยเป็น 0 เมื่อ x มีค่าน้อยกว่าหรือเท่ากับ 0 และเป็น x เมื่อมีค่ามากกว่า 0 ดังสมการที่ 5

$$R(x) = \max(0, x) \quad (5)$$

สำหรับชั้น Fully-connected (FCN) จะรับข้อมูลจากชั้นก่อนหน้า และผ่านการทำ flatten เปลี่ยนข้อมูลจากชั้นก่อนหน้าให้เป็นเวกเตอร์ขนาด 1 มิติ และส่งเข้าชั้น dense ขนาด 1024 ใช้ activation function เป็น ReLU และสำหรับชั้นสุดท้ายคือชั้น output เพื่อจำแนกประเภทรูปภาพเนื้อลายหินอ่อนเทียม จะใช้ชั้น dense ขนาด 1 ใช้ activation function เป็น Sigmoid ที่จะให้ค่าในช่วง 0 ถึง 1 เท่านั้น ดังสมการที่ 6

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (6)$$

จากการสร้างแบบจำลองแบบคอนโวลูชันในชั้นเชื่อมต่อทั้งหมดเข้าด้วยกัน จะได้โครงสร้างแบบจำลอง ดังแสดงในรูปที่ 27

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 150, 150, 3)]	0
conv2d (Conv2D)	(None, 148, 148, 16)	448
max_pooling2d (MaxPooling2D)	(None, 74, 74, 16)	0
conv2d_1 (Conv2D)	(None, 72, 72, 32)	4640
max_pooling2d_1 (MaxPooling2D)	(None, 36, 36, 32)	0
conv2d_2 (Conv2D)	(None, 34, 34, 64)	18496
max_pooling2d_2 (MaxPooling2D)	(None, 17, 17, 64)	0
flatten (Flatten)	(None, 18496)	0
dense (Dense)	(None, 1024)	18940928
dense_1 (Dense)	(None, 1)	1025

Total params: 18,965,537
 Trainable params: 18,965,537
 Non-trainable params: 0

รูปที่ 27 โครงสร้างของแบบจำลองโครงข่ายประสาทคอนโวลูชัน

4.3.2 แบบจำลองโครงข่ายประสาทคอนโวลูชันแบบวีจีจี 16 (VGG16)

การนำเข้าไลบรารีแบบจำลองการเรียนรู้แบบถ่ายโอนแบบวีจีจี 16 จากนั้นทำการรับข้อมูลนำเข้าขนาด 150x150x3 ตามความกว้าง ความสูง และจากชั้นสีแบบ RGB ของรูปภาพ และเนื่องจากชุดข้อมูลที่ต้องการจำแนกมีลักษณะการจำแนกต่างกับโครงสร้างแบบจำลองตั้งต้น ดังนั้น จึงไม่ต้องการชั้น Fully-connected (FCN) แบบจำลองการเรียนรู้ถ่ายโอนมาใช้ ต้องทำการตั้งค่าพารามิเตอร์ include_top ให้เป็น False และนำ weight จาก imagenet ที่ถูกฝึกสอนมาใช้กับแบบจำลอง ดังรูปที่ 28

```

from tensorflow.keras.applications.vgg16 import VGG16

# 1) instantiate a base model with pre-trained weights.
pretrained_model = VGG16(
    input_shape=(IMG_SIZE, IMG_SIZE, DIM_SIZE),
    include_top = False,
    weights='imagenet')
  
```

รูปที่ 28 การเรียกใช้ pre-trained weight ร่วมกับการเรียนรู้แบบถ่ายโอนแบบวีจีจี 16

จากแนวคิดและการวิจัยหัวข้อ 3.3 ในการสร้างแบบจำลองการเรียนรู้ถ่ายโอน จะต้องทำการตรึงค่า weight ของชั้น convolution ทั้งหมด โดยการตั้งค่าให้แต่ละ layer ตรึงค่า (Freeze) ให้ไม่มีการฝึกสอนซ้ำ ให้เป็น trainable เป็น False ดังรูปที่ 29

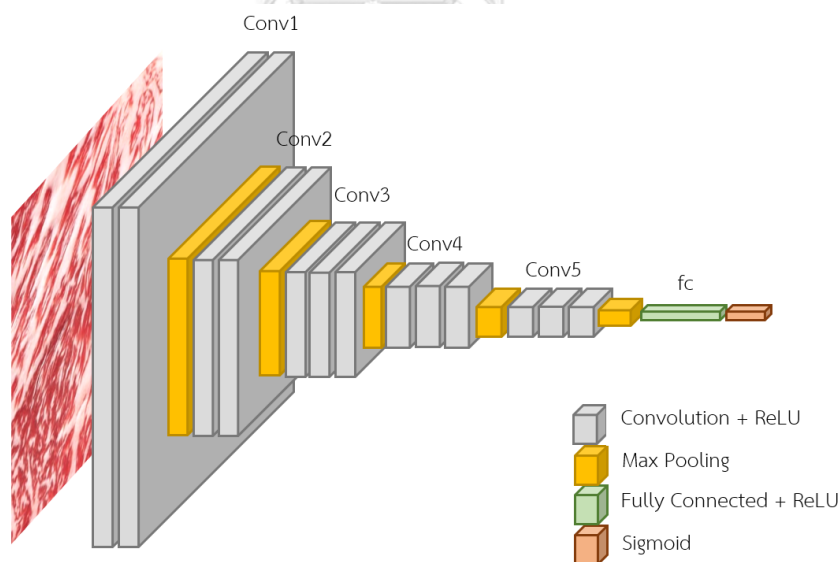

```
# 2) freeze the base model.
for layer in pretrained_model.layers:
    layer.trainable = False

# 3) Create a new model on top.
last_output = pretrained_model.output
```

รูปที่ 29 การตรึงค่า (Freeze) แบบจำลอง และใช้ pre-trained weight จาก imagenet

จากแนวคิดและการวิจัยหัวข้อ 3.5 ในการปรับปรุงสมรรถนะแบบจำลองการเรียนรู้ถ่ายโอน สามารถที่จะไม่ตรึงค่า weight บางชั้น convolution โดยการตั้งค่าให้แต่ละ layer ให้เป็น trainable เป็น True

จากนั้นใช้ชั้น Fully-connected (FCN) เหมือนแบบจำลองโครงข่ายประสาทเทียมแบบคอนโวลูชันมาเชื่อมต่อในชั้นถัดไป ซึ่งจะรับข้อมูลจากชั้นก่อนหน้า และผ่านการทำ flatten เปลี่ยนข้อมูลจากชั้นก่อนหน้าให้เป็นเวกเตอร์ขนาด 1 มิติ และส่งเข้าชั้น dense ขนาด 1024 ใช้ activation function เป็น ReLU และสำหรับชั้นสุดท้ายคือชั้น output ใช้ชั้น dense ขนาด 1 ใช้ activation function เป็น Sigmoid ดังรูปที่ 30



รูปที่ 30 โครงสร้างแบบจำลองโครงข่ายประสาทแบบวีจีจี 16 เพื่อจำแนกเนื้อลายหินอ่อนเทียม

สำหรับขนาดของโครงสร้างของแบบจำลองโครงข่ายประสาทแบบวีจีจี 16 หลังจากทำการปรับปรุงโครงสร้างให้เหมาะกับงานจำแนกเนื้อลายหินอ่อนเทียมจากหัวข้อ 3.3 ในงานวิจัยนี้ แสดงดังรูปที่ 31

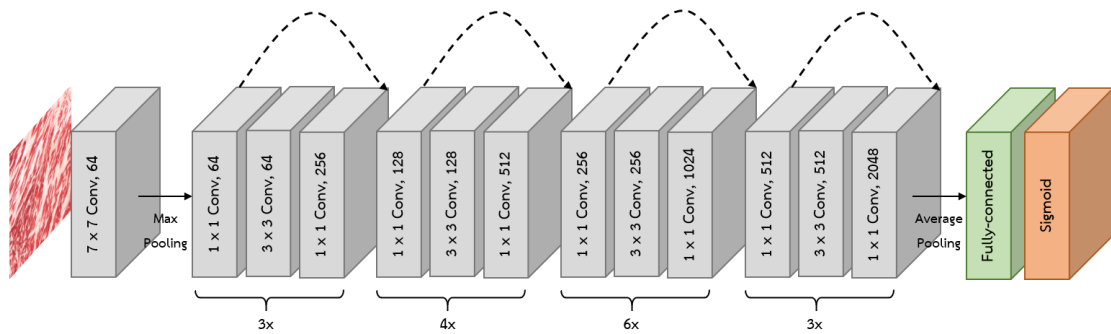
Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 150, 150, 3)]	0
block1_conv1 (Conv2D)	(None, 150, 150, 64)	1792
block1_conv2 (Conv2D)	(None, 150, 150, 64)	36928
block1_pool (MaxPooling2D)	(None, 75, 75, 64)	0
block2_conv1 (Conv2D)	(None, 75, 75, 128)	73856
block2_conv2 (Conv2D)	(None, 75, 75, 128)	147584
block2_pool (MaxPooling2D)	(None, 37, 37, 128)	0
block3_conv1 (Conv2D)	(None, 37, 37, 256)	295168
block3_conv2 (Conv2D)	(None, 37, 37, 256)	590080
block3_conv3 (Conv2D)	(None, 37, 37, 256)	590080
block3_pool (MaxPooling2D)	(None, 18, 18, 256)	0
block4_conv1 (Conv2D)	(None, 18, 18, 512)	1180160
block4_conv2 (Conv2D)	(None, 18, 18, 512)	2359808
block4_conv3 (Conv2D)	(None, 18, 18, 512)	2359808
block4_pool (MaxPooling2D)	(None, 9, 9, 512)	0
block5_conv1 (Conv2D)	(None, 9, 9, 512)	2359808
block5_conv2 (Conv2D)	(None, 9, 9, 512)	2359808
block5_conv3 (Conv2D)	(None, 9, 9, 512)	2359808
block5_pool (MaxPooling2D)	(None, 4, 4, 512)	0
flatten (Flatten)	(None, 8192)	0
dense (Dense)	(None, 1024)	8389632
dense_1 (Dense)	(None, 1)	1025
=====		
Total params: 23,105,345		
Trainable params: 8,390,657		
Non-trainable params: 14,714,688		

รูปที่ 31 โครงสร้างแบบจำลองโครงข่ายประสาทคอนโวลูชันแบบวีจีจี 16

4.3.3 แบบจำลองโครงข่ายประสาทเชิงลึกแบบ Residual Network (ResNet50)

สำหรับการนำเข้าไลบรารีแบบจำลองการเรียนรู้แบบถ่ายโอนแบบโครงข่ายประสาทเชิงลึกแบบ Residual Network 50 ชั้น ไปจนถึงการการสร้างแบบจำลองหลังจากทำการปรับปรุงโครงสร้างให้

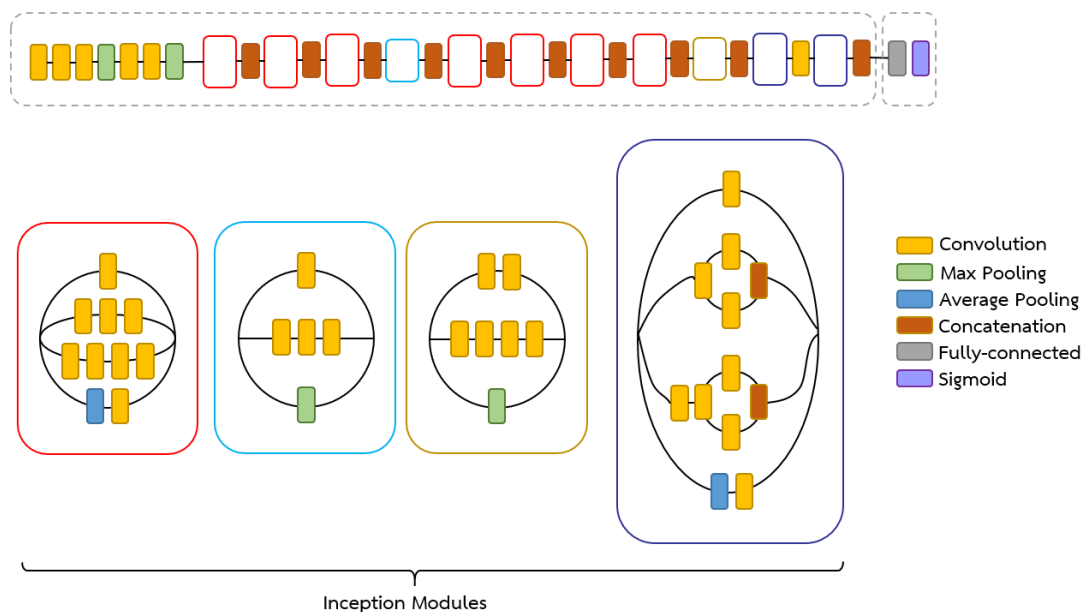
เหมาะกับการจำแนกเนื้อลายหินอ่อนเทียมจากหัวข้อ 3.3 มีขั้นตอนการสร้างเหมือนกับแบบจำลองโครงข่ายประสาทเทียมแบบวิจิจี 16 สำหรับขนาดของโครงสร้างของแบบจำลองเป็นไปดัง รูปที่ 32



รูปที่ 32 โครงสร้างแบบจำลองโครงข่ายประสาทเชิงลึกแบบ Residual Network 50 ชั้น เพื่อจำแนกเนื้อลายหินอ่อนเทียม

4.3.4 แบบจำลองโครงข่ายประสาทแบบ InceptionV3

สำหรับการนำเข้าไลบรารีแบบจำลองการเรียนรู้แบบถ่ายโอนแบบโครงข่ายประสาทแบบ InceptionV3 ไปจนถึงการสร้างแบบจำลองหลังจากทำการปรับปรุงโครงสร้างให้เหมาะกับการจำแนกเนื้อลายหินอ่อนเทียมจากหัวข้อ 3.3 มีขั้นตอนการสร้างเหมือนกับแบบจำลองโครงข่ายประสาทเทียมแบบวิจิจี 16 สำหรับขนาดของโครงสร้างของแบบจำลองเป็นไปดัง รูปที่ 33



รูปที่ 33 โครงสร้างแบบจำลองโครงข่ายประสาทแบบ InceptionV3
เพื่อจำแนกเนื้อลายหินอ่อนเทียม

4.4 การฝึกสอนแบบจำลอง

แบบจำลองทั้ง 4 โครงสร้าง ถูกกำหนดการคิด loss แบบ `binary_crossentropy` และมีการปรับเปลี่ยนค่า `learning rate` เป็น 0.01, 0.001 และ 0.0001 ตามลำดับ ใช้ค่าความแม่นยำ (accuracy) ในการวัดสมรรถนะในการฝึกสอนและตรวจสอบแบบจำลอง ใช้ optimizer แบบ Adam แล้วทำการ compile แบบจำลอง

เมื่อ compile แบบจำลองเสร็จสิ้น ทำการ fit แบบจำลอง กำหนด `batch_size = 5` ทำการฝึกสอนจำนวน 30 epoches ในการฝึกสอนแต่ละรอบใช้ค่า `step_per_epoch` จากการนำจำนวนข้อมูลฝึกสอนทั้งหมด หารด้วย `batch_size` ได้เท่ากับ 70 และได้ค่าเท่ากับ 20 สำหรับข้อมูลตรวจสอบ ทำการเก็บ weight ของการฝึกสอนแบบจำลองด้วยการใช้ `ModelCheckpoint` ที่ให้ค่าความแม่นยำ (accuracy) ที่ดีที่สุด `mode = max` และทำการบันทึกใน google drive ที่ได้ทำการ mount ไว้

สมรรถนะของการฝึกสอนแบบจำลองทั้ง 4 ใช้ `model.evaluation` และ `model.summary` ในการวัดสมรรถนะของแบบจำลองขณะฝึกสอน ดังรูปที่ 34 ทำการเปรียบเทียบค่าความแม่นยำ ค่า

ความสูญเสีย เวลาในการฝึกสอนแบบจำลองรวมไปจนถึงจำนวนพารามิเตอร์ของแต่ละแบบจำลอง
ดังตารางที่ 5

```
def beef_evaluate (model, validation_generator, steps_per_epoch):
    # Model evaluation
    final_loss, final_accuracy = model.evaluate( validation_generator,
                                                steps = steps_per_epoch)

    print("\n===== Training Performance =====")
    print("Final accuracy: {:.2f}%".format(final_accuracy*100))
    print("Final loss: {:.4f}".format(final_loss))
```

รูปที่ 34 ฟังก์ชันในการวัดสมรรถนะของการฝึกสอนแบบจำลอง

ตารางที่ 5 สมรรถนะของการฝึกสอนแบบจำลอง CNN, VGG16, ResNet50 และ InceptionV3

Models	Learning rate	Total params	Training time (s)	Final accuracy (%)	Final Loss
CNN	0.01	18,965,537	326	50.00	0.6932
VGG16		23,105,345	121	55.00	3.1192
ResNet50		76,018,561	125	50.00	0.6932
InceptionV3		40,679,201	126	83.75	0.7112
CNN	0.001	18,965,537	231	63.75	0.7996
VGG16		23,105,345	192	80.00	0.7363
ResNet50		76,018,561	94	56.25	1.0695
InceptionV3		40,679,201	95	82.50	0.4734
CNN	0.0001	18,965,537	263	52.50	0.6990
VGG16		23,105,345	120	85.00	0.5325
ResNet50		76,018,561	127	52.50	0.8496
InceptionV3		40,679,201	125	81.25	0.5759

4.5 ผลการทดลอง

จากการสร้างแบบจำลองทั้ง 4 โครงสร้าง และทำการทดสอบแบบจำลองด้วยชุดข้อมูลทดสอบ ใช้ prediction ค่าทำนายเท่ากับ 0 หมายถึง ทำนายว่าเป็นเนื้อฉีดไขมัน (Injected class) เมื่อค่า pred น้อยกว่าเท่ากับ 0.5 และทำนายได้ค่า 1 หมายถึง ทำนายว่าเป็นเนื้อไม่ฉีดไขมัน (NonInjected class) เมื่อค่า pred มากกว่า 0.5 โดยบันทึกผลการทดสอบเป็นรูปแบบ excel ประกอบไปด้วย ชื่อไฟล์รูปภาพ, ค่าความน่าจะเป็นของรูปภาพ และผลทำนายรูปภาพ จากนั้นทำการสรุปผลเป็น confusion matrix และแสดงผลการทำนายแต่ละ class ด้วย classification_report จะแสดงค่า average macro ของ precision, recall, f1 score และ accuracy ดังรูปที่ 35

```
def beef_predict (model, test_generator, filename):
    # Initialize filename of result
    save_file = base_dir+ "/" + filename + ".xlsx"
    # Get prediction value
    pred = model.predict(test_generator,
                        steps=len(test_generator),
                        verbose=1,
                        workers=0)

    # Get classes by probability > 0.5
    y_pred = np.where(pred > 0.5, 1, 0)
    # Get filenames of test set
    img_name = test_generator_filenames
    # Create data frame of prediction result
    results = pd.DataFrame({"Filename":img_name,
                          "Probability":pred[:,0],
                          "Predictions":y_pred[:,0]})

    # write result to excel file
    results.to_excel(save_file)
    print("\nResult was saved! =====")
    print (save_file)

    print("\n===== Confusion Matrix =====")
    cm = confusion_matrix(test_generator.classes, y_pred)
    print(cm)

    print("\n===== Classification Report =====")
    report = classification_report(test_generator.classes, y_pred,
                                  target_names=['Injected', 'NonInjected'])
    print(report)

    return results
```

รูปที่ 35 ฟังก์ชันในการวัดสมรรถนะของแบบจำลองด้วยชุดข้อมูลทดสอบ พร้อมทั้งการบันทึกผล
รูปแบบไฟล์ excel

เมื่อทำการรันการทดลองจะได้ผลลัพธ์เป็นไฟล์ excel ดังรูปที่ 36 จากนั้นจะสรุปผล
confusion matrix และ classification report ดังรูปที่ 37

Row	Image ID	File Path	Precision	Recall
40	38	Injected/Injected44.jpg	0.0000	0
41	39	Injected/Injected45.jpg	0.0000	0
42	40	Injected/Injected46.jpg	0.0000	0
43	41	Injected/Injected47.jpg	0.1299	0
44	42	Injected/Injected48.jpg	0.1866	0
45	43	Injected/Injected49.jpg	0.0047	0
46	44	Injected/Injected5.jpg	0.0603	0
47	45	Injected/Injected50.jpg	0.0624	0
48	46	Injected/Injected6.jpg	0.1917	0
49	47	Injected/Injected7.jpg	0.9553	1
50	48	Injected/Injected8.jpg	0.0078	0
51	49	Injected/Injected9.jpg	0.3375	0
52	50	NonInjected/NonInjected1.jpg	0.7606	1
53	51	NonInjected/NonInjected10.jpg	0.9996	1
54	52	NonInjected/NonInjected11.jpg	0.9654	1
55	53	NonInjected/NonInjected12.jpg	0.1509	0
56	54	NonInjected/NonInjected13.jpg	0.9893	1
57	55	NonInjected/NonInjected14.jpg	0.9143	1
58	56	NonInjected/NonInjected15.jpg	0.9987	1
59	57	NonInjected/NonInjected16.jpg	0.7513	1
60	58	NonInjected/NonInjected17.jpg	0.5816	1
61	59	NonInjected/NonInjected18.jpg	0.2413	0
62	60	NonInjected/NonInjected19.jpg	0.8096	1

รูปที่ 36 ตัวอย่างไฟล์ผลลัพธ์การทดสอบแบบจำลองรูปแบบ excel

```

===== Confusion Matrix =====
[[38 12]
 [ 9 41]]

===== Classification Report =====
              precision    recall  f1-score   support

   Injected           0.81         0.76         0.78         50
  NonInjected         0.77         0.82         0.80         50

 accuracy              0.79         0.79         0.79        100
 macro avg              0.79         0.79         0.79        100
 weighted avg           0.79         0.79         0.79        100

```

รูปที่ 37 ตัวอย่างผลลัพธ์การทดสอบแบบจำลองรูปแบบ confusion matrix และแสดงผลการ

ทำนายแต่ละ class ด้วย classification_report

สำหรับผลการทดลองทั้งแบบจำลองทั้ง 4 โครงสร้าง คือ CNN, VGG16, RestNet50 และ InceptionV3 สามารถดูสมรรถนะทั้งหมดได้จาก ตารางที่ 6

ตารางที่ 6 สมรรถนะของการทดสอบแบบจำลอง CNN, VGG16, ResNet50 และ InceptionV3

Model	Learning rate	Confusion matrix						Macro Average				Accuracy
		TP	FN	FP	TN	Precision	Recall	F1 score				
CNN	0.01	50	0	50	0	0.25	0.25	0.33	0.50			
VGG16		12	38	0	50	0.78	0.62	0.56	0.62			
ResNet50		0	50	0	50	0.25	0.50	0.33	0.50			
InceptionV3		39	11	14	36	0.75	0.75	0.75	0.75			
CNN	0.001	44	6	15	35	0.80	0.79	0.79	0.79			
VGG16		33	17	6	44	0.78	0.77	0.77	0.77			
ResNet50		8	42	2	48	0.67	0.56	0.48	0.56			
InceptionV3		45	5	14	36	0.86	0.86	0.86	0.86			
CNN	0.0001	3	47	0	50	0.76	0.53	0.40	0.53			
VGG16		34	16	7	43	0.78	0.77	0.77	0.77			
ResNet50		30	20	17	33	0.63	0.63	0.63	0.63			
InceptionV3		45	5	11	39	0.84	0.84	0.84	0.84			

จากตารางที่ 5 พบว่าเมื่อปรับค่า learning rate เท่ากับ 0.01 แบบจำลองไม่สามารถจำแนกรูปภาพทั้งสองประเภทได้ดีนัก จะมีตอบผลลัพธ์เป็นประเภทใดประเภทหนึ่งในแบบจำลอง CNN, VGG16 และ ResNet50 ยกเว้น InceptionV3 ที่ไม่ใช่ลักษณะตอบประเภทใดประเภทหนึ่ง อย่างไรก็ตาม ผลลัพธ์ที่ได้จากแบบจำลองที่มีการปรับละเอียด learning rate เท่ากับ 0.001 นั้นให้ค่าตัววัดทั้ง 4 ดีกว่าแบบจำลองที่มีการปรับละเอียด learning rate เท่ากับ 0.0001 แต่เมื่อพิจารณาถึงแบบจำลอง ResNet50 จะเห็นได้ว่าไม่สามารถดึงคุณลักษณะที่สำคัญจากชุดข้อมูลทดสอบได้มากเท่าที่ควร แบบจำลองยังคงให้ค่าความแม่นยำที่ต่ำกว่าแบบจำลองเปรียบเทียบพื้นฐานอย่าง CNN เนื่องจากลักษณะการทดสอบแบบดึงคุณลักษณะรูปภาพนั้นทำการดึงทุกชั้นของ Convolutional base และด้วยคุณลักษณะเด่นของแบบจำลองที่สามารถข้ามชั้นต่างๆ และไปประมวลผลในชั้นถัดไป อาจทำให้ไม่สามารถแยกคุณลักษณะของข้อมูลชุดใหม่ได้ดีนักเมื่อใช้สำหรับทดสอบสมรรถนะแบบจำลองในการดึงคุณลักษณะรูปภาพชุดใหม่

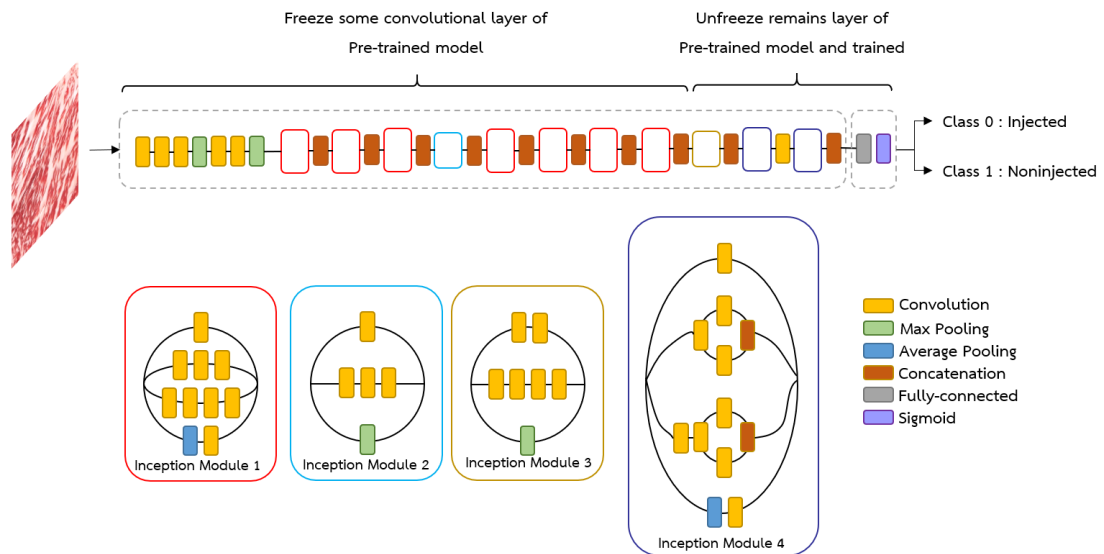
4.6 การปรับละเอียดสมรรถนะแบบจำลอง

จากสมรรถนะของการฝึกสอนแบบจำลองเห็นได้ว่าแบบจำลอง VGG16 และแบบจำลอง InceptionV3 มีสมรรถนะของแบบจำลองใกล้เคียงกันในระดับบวกไม่เกิน 0.05 แต่เมื่อพิจารณาสมรรถนะจากการทดสอบแบบจำลองจากไฟล์ผลลัพธ์ ดังรูปที่ 38 ซึ่งเป็นผลของการทำนายข้อมูลรูปภาพที่ไม่เคยเห็นมาก่อน โดยจะทำนายค่า class เท่ากับ 1 หรือเนื้อไม้ฉัดไขมันถ้าความน่าจะเป็นมีค่ามากกว่า 0.5 และจะทำนายค่า class เท่ากับ 0 หรือเนื้อฉัดไขมันถ้าความน่าจะเป็นมีค่าน้อยกว่าหรือเท่ากับ 0.5 ปรากฏว่า InceptionV3 ทำได้ดีกว่าทั้งในแง่ของค่า macro average ของ precision, recall, f1 score รวมไปถึงค่าความแม่นยำในการทำนาย จะได้ว่า การดึงคุณลักษณะของรูปภาพของแบบจำลอง InceptionV3 สามารถแยกความแตกต่างกันระหว่าง 2 ประเภทได้อย่างชัดเจน

	A	B	C	D	E	F	G	H	I	J	K
1		Filename	Probability	Predictions							
43	41	Injected/Injected47.jpg	0.1299	0							
44	42	Injected/Injected48.jpg	0.1866	0							
45	43	Injected/Injected49.jpg	0.0047	0							
46	44	Injected/Injected5.jpg	0.0603	0							
47	45	Injected/Injected50.jpg	0.0624	0							
48	46	Injected/Injected6.jpg	0.1917	0							
49	47	Injected/Injected7.jpg	0.9553	1							
50	48	Injected/Injected8.jpg	0.0078	0							
51	49	Injected/Injected9.jpg	0.3375	0							
52	50	NonInjected/NonInjected1.jpg	0.7606	1							
53	51	NonInjected/NonInjected10.jpg	0.9996	1							
54	52	NonInjected/NonInjected11.jpg	0.9654	1							
55	53	NonInjected/NonInjected12.jpg	0.1509	0							
56	54	NonInjected/NonInjected13.jpg	0.9893	1							
57	55	NonInjected/NonInjected14.jpg	0.9143	1							
58	56	NonInjected/NonInjected15.jpg	0.9987	1							
59	57	NonInjected/NonInjected16.jpg	0.7513	1							
60	58	NonInjected/NonInjected17.jpg	0.5816	1							
61	59	NonInjected/NonInjected18.jpg	0.2413	0							
62	60	NonInjected/NonInjected19.jpg	0.8096	1							

รูปที่ 38 ตัวอย่างความน่าจะเป็น และผลการจำแนกเนื้อลายหินอ่อนเทียมกับชุดข้อมูลทดสอบร่วมกับ การเรียนรู้ถ่ายโอนแบบ InceptionV3 เมื่อมีการกำหนดค่า learning rate เท่ากับ 0.001

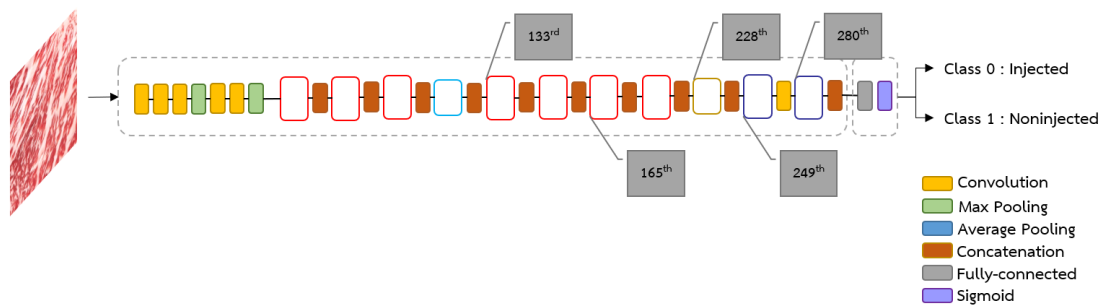
ดังนั้น สำหรับงานวิจัยการจำแนกเนื้อลายหินอ่อนเทียม จึงเลือกการเรียนรู้ถ่ายโอนแบบ InceptionV3 ร่วมกับการใช้ pre-trained weight จาก imagenet มาเพื่อปรับปรุงสมรรถนะ แบบจำลอง ทำการปรับปรุงการตั้งชั้นที่สามารถทำให้การดึงคุณลักษณะที่สำคัญของรูปภาพดีขึ้น ดังรูปที่ 39



รูปที่ 39 การปรับปรุงสมรรถนะของแบบจำลองการจำแนกเนื้อลายหินอ่อนเทียมร่วมกับการเรียนรู้ถ่ายโอนแบบ InceptionV3 เมื่อมีการกำหนดค่า learning rate เท่ากับ 0.001

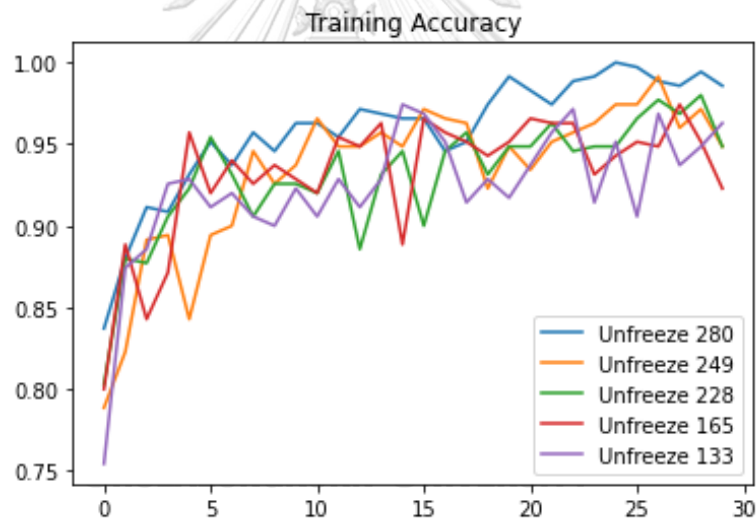
จากการทดสอบคอมพิวเตอร์แบบจำลอง inceptionV3 และแสดงจำนวนชั้นทั้งหมดในแบบจำลองการเรียนรู้แบบถ่ายโอน InceptionV3 ทำให้ทราบว่ามีชั้นจำนวนชั้นทั้งหมด 310 ชั้น ดังนั้น ในการปรับสมรรถนะแบบจำลองจะทำการละทิ้งการตรึง (Unfreeze) จาก top model ไปที่ละชั้นโดยเลือกพิจารณาจากการละทิ้ง จากกลุ่มของ Inception module ทั้ง 4 แบบ แล้วจึงบันทึกผล ดังรูปที่ 40

- จะทำการละทิ้งการตรึงตั้งแต่ Inception module 4 ลำดับที่ 1 คือชั้นที่ 280,
- จะทำการละทิ้งการตรึงตั้งแต่ Inception module 4 ลำดับที่ 2 คือชั้นที่ 249
- จะทำการละทิ้งการตรึงตั้งแต่ Inception module 3 คือชั้นที่ 228
- จะทำการละทิ้งการตรึงตั้งแต่ Inception module 1 ลำดับที่ 2 คือชั้นที่ 165
- จะทำการละทิ้งการตรึงตั้งแต่ Inception module 1 ลำดับที่ 4 คือชั้นที่ 133



รูปที่ 40 ปรับสมรรถนะแบบจำลองการเรียนรู้ถ่ายโอนแบบ InceptionV3 โดยทำการละทิ้งการตรึงตั้งแต่ชั้นที่ 280, 249, 228, 165 และ 133

สำหรับผลการทดลองปรับสมรรถนะของแบบจำลองการเรียนรู้แบบถ่ายโอน InceptionV3 ในขั้นตอนการฝึกสอนแบบจำลอง สามารถดูได้ดังรูปที่ 41 และ 42



รูปที่ 41 ความแม่นยำของแบบจำลองการเรียนรู้แบบถ่ายโอน InceptionV3 จากการละทิ้งการตรึงที่ชั้นต่างๆ ในขั้นตอนการฝึกสอนแบบจำลอง



รูปที่ 42 ความสูญเสียของแบบจำลองการเรียนรู้แบบถ่ายโอน InceptionV3 จากการละทิ้งการตรึงที่
ชั้นต่างๆ ในขั้นตอนการฝึกสอนแบบจำลอง

เมื่อพิจารณารูปที่ 41 จะเห็นได้ว่ายังมีการละทิ้งการตรึงมากเท่าไร ความแม่นยำในการฝึกสอนแบบจำลองยิ่งลดลงเมื่อเทียบกับขั้นตอนทดสอบก่อนหน้านี้ที่ได้สมรรถนะของแบบจำลองที่ไม่ละทิ้งการตรึงในชั้น convolutional ในขั้นตอนการฝึกสมรรถนะของแบบจำลองที่ได้ยังคงมีระดับที่ใกล้เคียงกันโดยที่มีเพียงการละทิ้งการตรึงตั้งแต่ Inception module 4 จำนวน 1 กลุ่ม และ การตรึงตั้งแต่ Inception module 3 เท่านั้นที่มีความแม่นยำที่มากกว่า 0.86 จากการทดลองก่อนหน้านี้ แต่หากพิจารณาตามรูปที่ 42 จะเห็นได้ว่าค่าความสูญเสียเพิ่มมากขึ้นเมื่อทำการละทิ้งการตรึง Inception module 1 แล้วฝึกสอนใหม่

สำหรับผลการทดลองหลังปรับสมรรถนะของแบบจำลองการเรียนรู้แบบถ่ายโอนแบบ InceptionV3 ในขั้นตอนการปรับละเอียดการละทิ้งการตรึงชั้นต่าง ๆ สามารถดูสมรรถนะทั้งหมดได้จาก ตารางที่ 7

ตารางที่ 7 ค่าตัววัดสมรรถนะแบบจำลองการเรียนรู้แบบถ่ายโอน InceptionV3 หลังการปรับละเอียดการละทิ้งการตรวจจับชั้นต่าง ๆ ในขั้นตอนการทดสอบแบบจำลอง

Model	Unfreeze starting	Confusion matrix					Macro Average				Accuracy
		TP	FN	FP	TN	Precision	Recall	F1 score			
InceptionV3	280	45	5	8	42	0.87	0.87	0.87	0.87	0.87	
	249	42	8	10	40	0.82	0.82	0.82	0.82	0.82	
	228	44	6	3	47	0.91	0.91	0.91	0.91	0.91	
	165	33	17	2	48	0.84	0.81	0.81	0.81	0.81	
	133	30	20	1	49	0.84	0.79	0.78	0.79	0.79	

ในขั้นตอนถัดมาจะทำการปรับละเอียดแบบจำลอง Inception v3 เมื่อทำการละทิ้งการตรึงตั้งแต่ชั้น 228 ที่ได้สมรรถนะที่ดีที่สุดในขั้นตอนก่อนหน้า ด้วยการใช้ Stratified 10-Fold Cross Validation เพื่อลดปัญหา overfitting เมื่อข้อมูลฝึกสอนมีขนาดเล็ก โดยที่จะมีขั้นตอนในการจัดการข้อมูล โดย training set กับ validation จากนั้นทำการฝึกสอนแบบจำลองแต่ละรอบเป็น 10 epochs จากนั้นเก็บสมรรถนะของแบบจำลองที่ได้รับการฝึกสอนแต่ละรอบและนำมาหาค่าเฉลี่ยทั้งหมดหลังการฝึกสอนแบบจำลอง ดังรูปที่ 43

```
# =====Stratified K-Fold=====
skf = StratifiedKFold(n_splits=10, shuffle=True)
skf.get_n_splits(X, Y)
foldNum=0

scores = []

for train_index, val_index in skf.split(X, Y):

    # Merged data from validation to train (if any exists)
    transferAllClassBetweenFolders('validation', 'train', 1.0)

    foldNum+=1
    print("=====")
    print("Results for fold ", foldNum)
    X_train, X_val = X[train_index], X[val_index]
    Y_train, Y_val = Y[train_index], Y[val_index]
    # Move validation data of this fold from train folder to the validation folder
    for eachIndex in range(len(X_val)):
        classLabel=''
        for i in range(len(classLabels)):
            if(Y_val[eachIndex]==i):
                classLabel=classLabels[i]
        #Copy the validation to the validation folder
        shutil.move(datasetFolderName+'/train/'+classLabel+'/'+X_val[eachIndex],
                    datasetFolderName+'/validation/'+classLabel+'/'+X_val[eachIndex])

    train_datagen = ImageDataGenerator(
        rescale=1./255,
        zoom_range=0.20,
        horizontal_flip=True,
        vertical_flip=True)
    validation_datagen = ImageDataGenerator(rescale=1./255)
    test_datagen = ImageDataGenerator(rescale=1./255)

    train_generator = train_datagen.flow_from_directory(
        train_path,
        target_size=(IMG_SIZE, IMG_SIZE),
        batch_size=BATCH,
        class_mode='binary',
        subset='training')

    validation_generator = validation_datagen.flow_from_directory(
        validation_path,
        target_size=(IMG_SIZE, IMG_SIZE),
        batch_size=BATCH,
        class_mode=None, # label only
        shuffle=False)

    # fit model
    history=model.fit(train_generator,
                      epochs=10,
                      verbose=2)

    # evaluate validation performance
    predictions = model.predict_generator(validation_generator, verbose=1)
    yPredictions = np.where(predictions>0.5, 1, 0)
    true_classes = validation_generator.classes

    print("***Performance on Validation data***")
    valAcc, valPrec, valRec, valFScore, = my_metrics(true_classes, yPredictions)
    scores.append([valAcc, valPrec, valRec, valFScore])
```

รูปที่ 43 การใช้ Stratified 10-Fold Cross Validation ปรับละเอียดแบบจำลอง Inception v3 เมื่อทำการละทิ้งการตรึงตั้งแต่ชั้น 228

เมื่อทำการฝึกสอนในแต่ละรอบจะรายงานผลด้วย confusion matrix, accuracy, precision, recall และ f-score สำหรับตัวอย่างการแสดงผลลัพธ์ของแบบจำลองรอบที่ 1 นั้นสามารถดังรูปที่ 44

```

=====
Results for fold 1
Found 481 images belonging to 2 classes.
Found 49 images belonging to 2 classes.
Epoch 1/10
97/97 - 21s - loss: 0.2533 - acc: 0.9376
Epoch 2/10
97/97 - 5s - loss: 0.1810 - acc: 0.9418
Epoch 3/10
97/97 - 5s - loss: 0.3201 - acc: 0.8940
Epoch 4/10
97/97 - 5s - loss: 0.2550 - acc: 0.8981
Epoch 5/10
97/97 - 5s - loss: 0.1820 - acc: 0.9459
Epoch 6/10
97/97 - 5s - loss: 0.2270 - acc: 0.9148
Epoch 7/10
97/97 - 5s - loss: 0.1506 - acc: 0.9501
Epoch 8/10
97/97 - 5s - loss: 0.1945 - acc: 0.9314
Epoch 9/10
97/97 - 5s - loss: 0.2003 - acc: 0.9252
Epoch 10/10
97/97 - 5s - loss: 0.2078 - acc: 0.9335
10/10 [=====] - 2s 83ms/step
***Performance on Validation data***
[[23  2]
 [ 1 23]]
=====
✓ 0s  completed at 7:47 AM
=====

```

รูปที่ 44 ตัวอย่างการแสดงผลลัพธ์ของแบบจำลองรอบที่ 1 จากการปรับละเอียดโดยใช้ Stratified 10-Fold Cross Validation

จากการทดสอบการปรับสมรรถนะแบบจำลองในขั้นตอนการใช้ Stratified 10-Fold Cross Validation สามารถดูผลลัพธ์จากการทำนายในแต่ละรอบ ดังแสดงในตารางที่ 8 จะเห็นว่าหลังจากการปรับปรุงสมรรถนะด้วย Stratified 10-Fold Cross Validation ทำให้สมรรถนะของแบบจำลองสูงขึ้นจากขั้นตอนก่อนหน้า ด้วยค่าเฉลี่ยของ precision, recall, f1 score และ accuracy มีค่าเป็น 97.51%, 95.87%, 96.63% และ 96.70% ตามลำดับ

ตารางที่ 8 ค่าตัววัดสมรรถนะแบบจำลองการเรียนรู้แบบถ่ายโอน InceptionV3 หลังการปรับละเอียดการละทิ้งการตรึงชั้น 228 และใช้ Stratified 10-Fold Cross Validation เพื่อปรับปรุงสมรรถนะแบบจำลอง

Model	Folds	Confusion matrix				Precision	Recall	F1 score	Accuracy
		TP	FN	FP	TN				
InceptionV3	1	23	2	1	23	0.9583	0.9200	0.9388	0.9388
	2	25	0	1	23	0.9615	1.0000	0.9804	0.9796
	3	25	0	1	23	0.9615	1.0000	0.9804	0.9796
	4	22	2	0	25	1.0000	0.9167	0.9565	0.9592
	5	24	0	0	25	1.0000	1.0000	1.0000	1.0000
	6	24	0	0	24	1.0000	1.0000	1.0000	1.0000
	7	24	0	1	23	0.9600	1.0000	0.9796	0.9792
	8	23	1	0	24	1.0000	0.9583	0.9787	0.9792
	9	20	4	2	22	0.9091	0.8333	0.8696	0.8750
	10	23	1	0	24	1.0000	0.9583	0.9787	0.9792
		Average				0.9751	0.9587	0.9663	0.9670

บทที่ 5

สรุปผลการวิจัย

วิทยานิพนธ์นี้ได้นำเสนอประยุกต์ใช้การเรียนรู้ถ่ายโอนร่วมกับโครงข่ายประสาทเทียมแบบคอนโวลูชัน เพื่อจำแนกประเภทภาพเนื้อลายหินอ่อนเทียม และไม่ใช้เนื้อลายหินอ่อนเทียม เนื่องจากชุดข้อมูลที่ใช้เป็นชุดข้อมูลที่ถูกรวบรวมจากแหล่งร้านค้าเนื้อออนไลน์ที่น่าเชื่อถือ และทำการเตรียมข้อมูลขึ้นเอง อีกทั้งเป็นชุดข้อมูลขนาดเล็กเพียง 530 รูปภาพ ดังนั้น การเปรียบเทียบการเรียนรู้ถ่ายโอนที่เหมาะสมกับรูปภาพเนื้อที่จะนำมาใช้ ให้แบบจำลองสามารถดึงคุณลักษณะที่สำคัญจากชุดข้อมูลชุดใหม่ที่มีขนาดเล็กจึงเป็นขั้นตอนที่จำเป็นในการคัดเลือกแบบจำลอง

สำหรับการเปรียบเทียบสมรรถนะของแบบจำลอง ผู้วิจัยได้ทำการสร้าง baseline model ซึ่งก็คือ โครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNN) ที่จะถูกฝึกสอนขึ้นเองตั้งแต่เริ่มต้น และสร้างแบบจำลองการเรียนรู้การถ่ายโอนประกอบไปด้วยแบบจำลอง 3 แบบด้วยกัน คือ โครงข่ายประสาทคอนโวลูชันแบบวีจีจี 16 (VGG16), โครงข่ายโครงข่ายประสาทเชิงลึกแบบ Residual Network 50 (ResNet50) และโครงข่ายประสาทแบบ InceptionV3 และทำการปรับสมรรถนะแบบจำลองด้วยการปรับ learning rate เป็น 0.01, 0.001 และ 0.0001 ตามลำดับ จากผลการทดสอบแบบจำลองทั้ง 4 แบบ พบว่าที่ learning rate เท่ากับ 0.001 แบบจำลอง VGG16, CNN และ InceptionV3 ให้ค่าความแม่นยำในการจำแนกประเภทเนื้อลายหินอ่อนเทียมเป็น 77% 79% และ 86% ตามลำดับ สำหรับแบบจำลอง ResNet50 ให้ค่าความแม่นยำสูงสุดสุดที่ 63% เมื่อ learning rate เท่ากับ 0.0001 ทำให้สรุปได้ว่าแบบจำลองการเรียนรู้ถ่ายโอน InceptionV3 สามารถใช้จำแนกประเภทรูปภาพเนื้อลายหินอ่อนเทียมได้

จากนั้นจึงนำแบบจำลองการเรียนรู้ถ่ายโอน InceptionV3 ไปปรับปรุงสมรรถนะด้วยทำการตั้งการตรึงใน pre-trained model ทำให้แบบจำลองสามารถเรียนรู้ข้อมูลรูปภาพชุดใหม่ได้ดียิ่งขึ้น จึงทำการละทิ้งการตรึงในแต่เริ่มต้นที่ชั้น 280, 249, 228, 165 และ 133 ตามลำดับ ผลปรากฏว่าแบบจำลองการเรียนรู้ถ่ายโอน InceptionV3 ที่ learning rate เท่ากับ 0.001 และละทิ้งการตรึงตั้งแต่ชั้น 228 ให้ผลการทดสอบการจำแนกเนื้อลายหินอ่อนเทียมได้ความแม่นยำที่ดีที่สุด เท่ากับ 91% และทำการใช้ Stratified 10-Fold Cross Validation เพื่อลดปัญหาการ overfitting เมื่อข้อมูลมีขนาดเล็กและมีคุณของรูปภาพที่ใกล้เคียงกัน ด้วยค่าเฉลี่ยของ precision, recall, f1 score และ accuracy มีค่าเป็น 97.51%, 95.87%, 96.63% และ 96.70%

จากผลการทดลองทั้งสิ้น 3 การทดลองในงานวิจัยการเรียนรู้ถ่ายโอนสำหรับการจำแนกประเภทเนื้อลายหินอ่อนเทียมนั้นจะเห็นได้ว่า 1) ชุดข้อมูลเนื้อลายหินอ่อนเทียม และไม่ใช้ลายหิน

อ่อนเทียม แบบการจำลองการเรียนรู้ถ่ายโอนแบบ Inceptionv3 สามารถจำแนกประเภทเนื้อทั้ง 2 แบบได้อย่างมีประสิทธิภาพ 2) จากขนาดของชุดข้อมูลที่มีจำกัด แบบจำลองการเรียนรู้ถ่ายโอนสามารถทำนายรูปภาพที่ไม่เคยเห็นมาก่อนได้เป็นอย่างดี 3) การปรับละเอียดสมรรถนะแบบจำลองทั้งแบบการละทิ้งการตรึงในชั้นต่าง ๆ ส่งผลต่อสมรรถนะของแบบจำลอง และ 4) ในขั้นตอนการปรับละเอียดแบบจำลองด้วย Stratified 10-Fold Cross Validation สามารถเพิ่มสมรรถนะของแบบจำลองได้ ทำให้แบบจำลองที่ได้นั้นสามารถพัฒนาต่อยอดเพื่อเป็นประโยชน์แก่ผู้บริโภค ประกอบการตัดสินใจในการเลือกซื้อผลิตภัณฑ์เนื้อที่เหมาะสมกับราคาต่อไป

สำหรับแนวทางการวิจัยในอนาคต เนื่องจากขั้นตอนในการเตรียมข้อมูลรูปภาพเนื้อที่ฉีกไขมัน และไม่ฉีกไขมันในส่วนบริเวณเนื้อเท่านั้นใช้เวลาในการเตรียมข้อมูลนาน ดังนั้น การสร้างแบบจำลองร่วมกับเทคนิคการตัดขอบเขตที่น่าสนใจ (Region of interest: ROI) เพื่อเตรียมกรอบภาพคัดเลือกเฉพาะบริเวณเนื้อที่สนใจอัตโนมัติจะช่วยลดเวลาในการเตรียมข้อมูลฝึกสอนแบบจำลอง และสะดวกต่อการนำไปใช้ต่อเมื่อพบกับรูปภาพที่ไม่ได้มีเพียงเนื้อและลายหินอ่อน

บรรณานุกรม

1. Ritchie, H. and M. Roser, *Meat and dairy production*. Our World in Data, 2017.
2. Spyrou, C. *How Artificial Marbling Has Been Tricking Steak Lovers For Years*. 2017 [cited 2021 May]; Available from: <https://www.foodbeast.com/news/beef-artificial-marbling/>.
3. *What is beef tallow injected meat? Similar technique for French cuisine*. 2013 [cited 2021 May]; Available from: https://style.nikkei.com/article/DGXNASFK0805T_Y3A101C1000000.
4. Academy, M. วิธีสังเกต “เนื้อวัวฉีดไขมัน” กินแล้วเป็นอันตรายต่อสุขภาพหรือไม่? [cited 2021 May]; Available from: https://www.matichonacademy.com/content/tipsandtricks/article_50036.
5. *Beef Meltique Striploin 200g*. [cited 2021 May]; Available from: <https://soonbutcher.com/products/meltique-striploin-200gm>.
6. LeCun, Y., et al., *Gradient-based learning applied to document recognition*. Proceedings of the IEEE, 1998. **86**(11): p. 2278-2324.
7. Balodi, T. *The architecture of Convolutional Neural Networks(CNN)*. 2021 [cited 2021 May]; Available from: <https://www.analyticssteps.com/blogs/convolutional-neural-network-cnn-graphical-visualization-code-explanation>.
8. Simonyan, K. and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*. arXiv preprint arXiv:1409.1556, 2014.
9. *The architecture of VGG16*. [cited 2021 May]; Available from: <https://neurohive.io/en/popular-networks/vgg16/>.
10. He, K., et al. *Deep residual learning for image recognition*. in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
11. Szegedy, C., et al. *Rethinking the inception architecture for computer vision*. in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
12. *Inception-v3*. [cited 2021 May]; Available from: <https://paperswithcode.com/method/inceptionv3>.

13. *Keras Applications*. [cited 2021 May]; Available from: <https://keras.io/api/applications/>.
14. Sarkar, D., *A comprehensive hands-on guide to transfer learning with real-world applications in deep learning*. Towards Data Science. Saatavissa: <https://towardsdatascience.com/a-comprehensive-handson-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a>. Hakupäivä, 2018. **20**: p. 2020.
15. Kim, S.H., *Method for Estimating Intramuscular Fat Percentage of Hanwoo (Korean Traditional Cattle) Using Convolutional Neural Networks in Ultrasound Images*. International journal of advanced smart convergence, 2021. **10**(1): p. 105-116.
16. Penning, B.W., W.M. Snelling, and M.J. Woodward-Greene, *Machine learning in the assessment of meat quality*. IT Professional, 2020. **22**(3): p. 39-41.
17. Handayani, H.H., et al. *The Best Classification Algorithm for Identification Beef Quality Based on Marbling*. in *2020 Fifth International Conference on Informatics and Computing (ICIC)*. 2020. IEEE.
18. Fitrianto, A. and B. Sartono, *Image Classification of Beef and Pork Using Convolutional Neural Network in Keras Framework*. International Journal of Science, Engineering and Information Technology, 2021. **5**(02): p. 245-248.
19. Ju, J., et al., *Classification of jujube defects in small data sets based on transfer learning*. Neural Computing and Applications, 2021: p. 1-14.
20. Özgenel, Ç.F. and A.G. Sorguç. *Performance comparison of pretrained convolutional neural networks on crack detection in buildings*. in *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*. 2018. IAARC Publications.
21. Hosny, K.M., M.A. Kassem, and M.M. Foad. *Skin cancer classification using deep learning and transfer learning*. in *2018 9th Cairo international biomedical engineering conference (CIBEC)*. 2018. IEEE.
22. Demir, A., F. Yilmaz, and O. Kose. *Early detection of skin cancer using deep learning architectures: resnet-101 and inception-v3*. in *2019 Medical Technologies Congress (TIPTEKNO)*. 2019. IEEE.

23. Liu, X., et al. *Transfer learning with convolutional neural network for early gastric cancer classification on magnifying narrow-band imaging images*. in *2018 25th IEEE International Conference on Image Processing (ICIP)*. 2018. IEEE.





จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ประวัติผู้เขียน

ชื่อ-สกุล	Grace Panitchakorn
วัน เดือน ปี เกิด	22 May 1994
สถานที่เกิด	Bangkok, Thailand
วุฒิการศึกษา	Bachelor of Engineering, Prince of Songkla University
ที่อยู่ปัจจุบัน	519/172 Sikan, Don Mueang, Bangkok, Thailand, 10210
ผลงานตีพิมพ์	G Panitchakorn, Y Limpitakorn. (2021). Convolutional Neural Networks for Artificial Marbling Beef Classification. ICICSE 2021 10th International Conference on Internet Computing for Science and Engineering, Virtual Conference.