กลยุทธ์การตัดบนโยโลวีสามสำหรับการตรวจจับวัตถุแบบทันกาล

นายณัฐนนท์ กฤตยานวัช

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2562

Pruning Strategy on YOLOv3 for Real-Time Object Detection

Mr. Nattanon Krittayanawach

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 2019

หัวข้อวิทยานิพนธ์	กลยุทธ์การตัดบนโยโลวีสามสำหรับการตรวจจับวัตถุแบบทันกาล
โดย	นายณัฐนนท์ กฤตยานวัช
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผู้ช่วยศาสตราจารย์ ดร.พีรพล เวทีกูล

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

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

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ผู้ช่วยศาสตราจารย์ ดร.พีรพล เวทีกูล)

..... กรรมการ
(อาจารย์ ดร.ดวงดาว วิชาดากุล)

..... กรรมการภายนอกมหาวิทยาลัย
(ดร.ธนภัทร ฆังคะจิตร)

ณัฐนนท์ กฤตยานวัช : กลยุทธ์การตัดบนโยโลวีสามสำหรับการตรวจจับวัตถุแบบทันกาล. (Pruning Strategy on YOLOv3 for Real-Time Object Detection) อ.ที่ปรึกษาหลัก : ผศ. ดร.พีรพล เวทีกูล

ในงานตรวจจับวัตถุ แบบจำลอง YOLOv3 จัดว่าเป็นแบบจำลองที่มีประสิทธิภาพดีในด้านความแม่นยำ แต่ทว่าด้วยจำนวนตัวแปรในแบบจำลองที่มีมากกว่าสิบล้านตัวแปร ส่งผลให้ตัวแบบจำลองไม่เหมาะสมที่จะนำไปใช้งานบนกล้องหรืออุปกรณ์ขนาดเล็ก โดยงานวิจัยชิ้นนี้นำเสนอกลไกการบีบอัดแบบจำลองที่ออกแบบมาโดยเฉพาะสำหรับแบบจำลอง YOLOv3 เพื่อตัดตัวกรองที่ไม่จำเป็นออกจากตัวแบบจำลอง แต่เนื่องจากแบบจำลอง YOLOv3 นั้นประกอบไปด้วยองค์ประกอบ 2 ส่วน คือ โครงข่ายกระดูกสันหลัง และโครงข่ายพีระมิดฟีเจอร์ งานวิจัยชิ้นนี้จึงนำเสนอกลยุทธ์ 3 อย่างดังต่อไปนี้ 1) การตัดแบบแยกส่วน 2) การจำกัดการตัด และ 3) เกณฑ์การหยุด หลังจากนั้นจึงนำกลยุทธ์ทั้ง 3 อย่างมารวมกันเป็นกลไกการตัดแบบทนทานเพื่อตัดแบบจำลองแบบแยกส่วนกัน ด้วยวิธีการนี้ สามารถช่วยป้องกันการตัดส่วนใดส่วนหนึ่งของแบบจำลองมากเกินไป ส่งผลให้แบบจำลองมีเสถียรภาพมากขึ้น

สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์ ลายมือชื่อนิสิต .....

ปีการศึกษา 2562 ลายมือชื่อ อ.ที่ปรึกษาหลัก .....

6071010621 : MAJOR COMPUTER SCIENCE
KEYWORD:

Nattanon Krittayanawach : Pruning Strategy on YOLOv3 for Real-Time Object Detection. Advisor: Asst. Prof. Dr. PEERAPON VATEEKUL

For object detection, YOLOv3 has shown promising accuracy. Since the number of parameters in this network can be more than ten million parameters, it cannot be fit into a commodity camera or small devices. In this research, we propose a compression mechanism designed specifically for YOLOv3's network by removing unnecessary filters. Since YOLOv3 composes of two network components: backbone and pyramid networks, we propose the following techniques, (1) separated pruning, (2) minimum filter constraint, and (3) stopping criteria. Then, we combined these three mechanisms as a robust pruning mechanism to prune filters of each network separately. This can help to avoid over-pruning the network in some parts of the model making our model more robust.

Field of Study:	Computer Science	Student's Signature .....
Academic Year:	2019	Advisor's Signature .....

กิตติกรรมประกาศ

การที่วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดีนั้น นอกจากการทำงานของตัวผู้วิจัยแล้ว ยังมีบุคคลท่านอื่นที่เป็นส่วนสำคัญที่ได้ให้ความช่วยเหลือในการจัดทำวิทยานิพนธ์ฉบับนี้ขึ้นมา ผู้วิจัยรู้สึกซาบซึ้งในความกรุณาเหล่านี้เป็นอย่างมากจึงใคร่ขอใช้เนื้อหาในส่วนกิตติกรรมประกาศของวิทยานิพนธ์ฉบับนี้แสดงความขอบพระคุณเป็นอย่างสูงมา ณ ที่นี้

ขอขอบพระคุณอาจารย์ที่ปรึกษา ผศ. ดร. พีรพล เวทีกูล ผู้ที่คอยให้ความช่วยเหลือและให้คำปรึกษา รวมทั้งผลักดันให้งานวิจัยและวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้

ขอขอบพระคุณคณะกรรมการสอบวิทยานิพนธ์ซึ่งประกอบไปด้วย ศ. ดร. บุญเสริม กิจสิริกุล อ. ดร. ดวงดาว วิชาดากุล และ อ. ดร. ธนภัทร ฆังคะจิตร ที่ได้กรุณาให้เกียรติเป็นคณะกรรมการ รวมทั้งให้คำปรึกษาและข้อเสนอแนะอันเป็นประโยชน์อย่างมากต่อการทำวิจัยและวิทยานิพนธ์ฉบับนี้

ขอขอบคุณเพื่อน ๆ พี่ ๆ น้อง ๆ สมาชิก “Data Mining Group, MIND Lab” ทุกท่าน สำหรับกำลังใจ และคำแนะนำต่าง ๆ เพื่อนำมาประยุกต์ใช้ในวิทยานิพนธ์ฉบับนี้

สุดท้ายนี้ขอขอบพระคุณครอบครัวของผู้วิจัยที่ให้การสนับสนุนในทุก ๆ ด้าน และคอยให้กำลังใจตลอดระยะเวลาในการดำเนินการทำงานวิจัยนี้

ณัฐนนท์ กฤตยานวัช

สารบัญ

หน้า

สารบัญตาราง

หน้า

สารบัญรูปภาพ

หน้า

บทที่ 1
ที่มาและความสำคัญ

1.1 ที่มาและความสำคัญ

การตรวจจับวัตถุจากรูปภาพเป็นงานที่มีความสำคัญชิ้นหนึ่ง แบบจำลองการตรวจจับวัตถุจากรูปภาพสามารถนำไปใช้งานได้อย่างหลากหลายไม่ว่าจะเป็น รถยนต์ไร้คนขับ ระบบรักษาความปลอดภัย และระบบติดตามยานพาหนะ เป็นต้น ซึ่งจากการเข้ามาของการเรียนรู้เชิงลึก ทำให้ประสิทธิภาพของแบบจำลองการตรวจจับวัตถุมีการพัฒนาไปอย่างรวดเร็ว โดยปกตินั้น การประมวลผลตรวจจับวัตถุจะเหมาะสำหรับทำการประมวลผล ณ ตัวกล้องหรืออุปกรณ์ที่ถ่ายภาพเนื่องจากข้อมูลรูปภาพและวิดีโอมีขนาดใหญ่ แต่การใช้งานแบบจำลองการตรวจจับวัตถุประเภทการเรียนรู้เชิงลึก มีต้นทุนการคำนวณ (Computational Cost) สูง ทำให้ไม่สามารถนำแบบจำลองไปใช้งานบนอุปกรณ์ขนาดเล็กได้อย่างเต็มประสิทธิภาพ หรือไม่สามารถใช้งานได้เลย ในทางกลับกัน การส่งภาพหรือวิดีโอผ่านเครือข่ายเพื่อนำข้อมูลภาพและวิดีโอกลับมาประมวลผลแบบรวมศูนย์ก็ไม่สามารถใช้งานได้อย่างมีประสิทธิภาพ เนื่องจากขนาดของรูปภาพและวิดีโอที่มีขนาดใหญ่ ซึ่งส่งผลให้การส่งข้อมูลผ่านระบบเครือข่ายจำเป็นต้องใช้แบนด์วิดท์เป็นปริมาณมาก รวมถึงยังต้องคำนึงถึงเรื่องความล่าช้าในการส่งข้อมูลผ่านระบบเครือข่ายเนื่องจากขนาดของข้อมูลมีขนาดใหญ่ทำให้ใช้เวลามาก ซึ่งทำให้รูปแบบการประมวลผลแบบรวมศูนย์ไม่เหมาะกับการใช้งานกับโปรแกรมประยุกต์ในเชิงขนาดใหญ่ที่จำเป็นต้องมีการใช้งานแบนด์วิดท์เป็นปริมาณมาก

ในปัจจุบันการเรียนรู้เชิงลึกได้ช่วยเพิ่มประสิทธิภาพอย่างมีนัยสำคัญให้กับงานด้านการประมวลผลภาพ เช่น งานด้านการจำแนกภาพ (Image Classification) งานด้านการตรวจจับวัตถุ (Object Detection) และงานด้านการแบ่งส่วนวัตถุ (Object Segmentation) แบบจำลองการเรียนรู้เชิงลึกในด้านการประมวลผลภาพจะนิยมใช้คุณลักษณะ (Feature) ที่ถูกสร้างขึ้นมาจากแบบจำลองการเรียนรู้เชิงลึก ซึ่งแตกต่างจากแบบจำลองการเรียนรู้ของเครื่อง (Machine Learning) แบบดั้งเดิมเช่น Deformable Parts Model (DPM) [1] ที่ใช้งานฟีเจอร์ที่มนุษย์สร้างขึ้นมา ยกตัวอย่างเช่น SIFT [2] HOG [3] และ FV [4] โดยตามปกติแล้วแบบจำลองการตรวจจับวัตถุแบบการเรียนรู้เชิงลึกสามารถแบ่งออกได้เป็นสองประเภทหลักๆ คือ 1) สองขั้นตอน (Two-stage) เช่น R-CNN [5] Fast R-CNN [6] และ Faster R-CNN [7] และ 2) หนึ่งขั้นตอน (One-stage) เช่น YOLO [8] SSD [9] และ RetinaNet [10]

ในการทำงานของแบบจำลองสำหรับตรวจจับวัตถุแบบสองขั้นตอนจะประกอบไปด้วย 2 ส่วนดังนี้ 1) โครงข่ายเสนอขอบเขต (Region Proposal Network) และ 2) โครงข่ายจำแนก (Classification Network) แบบจำลองแบบสองขั้นตอนมีจุดเด่นด้านการตรวจจับวัตถุได้แม่นยำ แต่มีข้อเสียคือ แบบจำลองแบบสองขั้นตอนจะมีต้นทุนคำนวณสูงกว่าแบบหนึ่งขั้นตอนมาก ซึ่งไม่เหมาะสมสำหรับการนำไปใช้งานในอุปกรณ์ขนาดเล็กหรืออุปกรณ์ที่มีข้อจำกัดด้านความสามารถในการคำนวณและการใช้พลังงาน สำหรับการตรวจจับวัตถุแบบหนึ่งขั้นตอน (One-Stage) จะมีจุดที่แตกต่างที่สำคัญ คือ แบบจำลองไม่จำเป็นต้องใช้โครงข่ายเสนอขอบเขตเพื่อสร้างสี่เหลี่ยมในการปิดล้อมวัตถุ (Bounding Box) แต่ทำการสร้างสี่เหลี่ยมในการปิดล้อมวัตถุและจำแนกวัตถุออกมาพร้อม ๆ กันผ่านตัวโครงข่ายประสาทเทียมโดยตรง โดยการเปลี่ยนรูปแบบการมองโจทย์ปัญหาเรื่องการตรวจจับวัตถุเป็นแบบการวิเคราะห์ถดถอย (Regression Problem) ซึ่งจุดเด่นที่สำคัญของวิธีการนี้คือตัวแบบจำลองสามารถทำงานได้เร็ว

รวมถึงมีความเร็วต่อต้นทุนที่ใช้ในการคำนวณสูง ทำให้แบบจำลองการตรวจจับวัตถุมีความเหมาะสมที่จะใช้งานกับการประมวลผลภาพแบบทันที (Real-Time) มากกว่าแบบจำลองการตรวจจับวัตถุแบบสองขั้นตอน

โดยแบบจำลองการตรวจจับวัตถุแบบหนึ่งขั้นตอนทั้งจะประกอบไปด้วย SSD RetinaNet และ YOLOv3 เมื่อนำมาเปรียบเทียบกันแล้วจะพบว่า RetinaNet จัดเป็นแบบจำลองที่มีความแม่นยำที่สุดในกลุ่มของแบบจำลองการตรวจจับวัตถุแบบหนึ่งขั้นตอน ซึ่งมีความแม่นยำและความเร็วที่สูงกว่า SSD แต่ทว่าเมื่อเปรียบเทียบกับแบบจำลอง YOLOv3 แล้ว แบบจำลอง YOLOv3 จะมีความแม่นยำที่น้อยกว่า RetinaNet เพียงเล็กน้อย แต่มีความเร็วที่สูงกว่า จึงทำให้แบบจำลอง YOLOv3 มีความเหมาะสมที่จะนำไปใช้บนอุปกรณ์ขนาดเล็กที่มีทรัพยากรจำกัดมากกว่า ในงานวิจัยชิ้นนี้จึงเลือกใช้แบบจำลอง YOLOv3 เป็นแบบจำลองหลัก

อย่างไรก็ตามแบบจำลองแบบหนึ่งขั้นตอนก็ยังคงมีปัญหาในเรื่องจำนวนพารามิเตอร์ที่มีปริมาณมากเกินไป รวมถึงต้นทุนการคำนวณที่สูง ซึ่งไม่เหมาะสมสำหรับการนำไปใช้งานบนอุปกรณ์ขนาดเล็ก เพื่อแก้ปัญหานี้จึงมีการคิดค้นวิธีการบีบอัดแบบจำลองการเรียนรู้เชิงลึกเพื่อใช้ในการลดขนาดของแบบจำลองและต้นทุนในการคำนวณ วิธีการบีบอัดแบบจำลองการเรียนรู้เชิงลึกสามารถแบ่งออกได้เป็น 3 วิธีการ คือ 1) การแบ่งนับและการทวิภาคแบบจำลอง (Model Quantization and Binarization) 2) การตัดและการแบ่งปัน (Pruning and Sharing) และ 3) เมทริกซ์โครงสร้าง (Structural Matrix) การแบ่งนับและการทวิภาคแบบจำลองเป็นวิธีการที่ลดขนาดของบิตของข้อมูลที่ใช้เก็บค่าน้ำหนักเพื่อลดขนาดของแบบจำลองลงทำให้สามารถลดขนาดของแบบจำลองได้ แต่ทว่าวิธีการนี้จะส่งผลให้แบบจำลองทำงานช้าลงเนื่องจากความไม่เข้ากันของฮาร์ดแวร์กับรูปแบบของบิตที่เล็กลง สำหรับวิธีการเมทริกซ์โครงสร้างคือวิธีการที่พยายามลดมิติของข้อมูลลง เพื่อลดต้นทุนการคำนวณและขนาด แต่ทว่าเนื่องจากการหาตัวแทนของชั้น (Layer) ค่าน้ำหนักเดิมสามารถทำได้ยาก รวมถึงมีโอกาสทำให้ชั้นที่สร้างขึ้นมาใหม่มีความลำเอียงและไม่มีสูตรที่สามารถแก้สมการหาชั้นตัวแทนได้โดยตรง จึงไม่เหมาะสมสำหรับการนำมาใช้งานจริง สำหรับวิธีการตัดและการแบ่งปัน เป็นวิธีการบีบอัดตัวแบบจำลองด้วยการตัดตัวแปรน้ำหนักที่มีความสำคัญต่ำออกจากแบบจำลอง ทำให้สามารถลดขนาดของแบบจำลองลง รวมถึงยังทำให้แบบจำลองสามารถทำงานได้เร็วขึ้น จึงเหมาะสมที่จะใช้บีบอัดแบบจำลองสำหรับงานแบบทันที (Real-Time) บนอุปกรณ์ขนาดเล็ก

วิธีการตัด (Pruning) โครงข่ายประสาทเทียมถือเป็นวิธีการที่สำคัญ เพื่อใช้ในการลดขนาดของแบบจำลองการเรียนรู้เชิงลึก เนื่องจากว่าวิธีการนี้สามารถลดขนาดของแบบจำลองได้โดยตรงจากการตัดตัวแปรน้ำหนักที่ไม่สำคัญออก รวมถึงผลจากการตัดตัวแปรน้ำหนักยังทำให้ต้นทุนการคำนวณรวมของแบบจำลองลดลงอีกด้วย การตัดโครงข่ายประสาทเทียมสามารถแบ่งออกเป็นได้สองประเภทคือ 1) การตัดแบบไร้โครงสร้าง (Unstructured Pruning) [11] และ 2) การตัดแบบเชิงโครงสร้าง (Structured Pruning) [12] การตัดแบบไร้โครงสร้างคือการตัดที่เน้นที่การนำค่าน้ำหนักที่ไม่สำคัญออกจากแบบจำลองโครงข่ายประสาทเทียมโดยไม่คำนึงถึงโครงสร้างของชั้นต่าง ๆ ในแบบจำลอง ซึ่งวิธีนี้เมื่อนำน้ำหนักออกแล้ว จะทำให้แบบจำลองเกิดความเสียหายขึ้น และค่าน้ำหนักในแต่ละชั้นจะมีลักษณะมากเลขศูนย์ (Sparse) ด้วยเหตุนี้ทำให้แบบจำลองที่ตัดแล้วจึงไม่สามารถใช้ฮาร์ดแวร์และซอฟต์แวร์การเรียนเชิงลึกทั่วไปตามท้องตลาดได้ ในทางกลับการตัดแบบเชิงโครงสร้าง คือการตัดแบบจำลองโดยมีการคำนึงถึงโครงสร้างแบบมิติของแต่ละชั้นของโครงข่ายประสาทเทียม ซึ่งการตัดในลักษณะนี้จะมองค่าตัวแปรน้ำหนักในภาพที่ใหญ่กว่า ซึ่งโดยทั่วไปการตัดประเภทนี้จะมองที่ระดับตัวกรองของชั้นคอนโวลูชัน (Convolutional Layer) ทำให้แบบจำลองที่ผ่านการตัดไม่ได้รับความเสียหาย ส่งผลให้แบบจำลองที่ผ่านการตัดรูปแบบนี้สามารถใช้งานกับฮาร์ดแวร์และซอฟต์แวร์การเรียนเชิงลึกที่เป็นมาตรฐานได้

ในงานวิจัยชิ้นนี้ จะเสนอแบบจำลองการเรียนรู้เชิงลึก YOLOv3 แบบบีบอัดสำหรับตรวจจับวัตถุด้วยวิธีการตัด และเสนอวิธีการสำหรับเพิ่มประสิทธิภาพในการตัดค่าน้ำหนักของแบบจำลอง (1) การตัดแบบแยกส่วน (2) การจำกัดการตัดในแต่ละชั้น และ (3) เกณฑ์การหยุด เนื่องจากว่าแบบจำลองการตรวจจับวัตถุ โดยปกติแล้วจะประกอบด้วยโมดูลหลายชนิด ซึ่งตัวแปรน้ำหนักที่อยู่ในแต่ละโมดูลมีคุณลักษณะและช่วงของค่าที่แตกต่างกัน ทำให้การตัดแบบจำลองแบบทั้งหมดพร้อมกันนั้นไม่สามารถทำงานได้ดีเท่าที่ควร เนื่องจากว่ามีโอกาสที่บางส่วนของแบบจำลองจะถูกตัดมากเกินไป ทำให้ความแม่นยำของแบบจำลองลดลงอย่างรวดเร็ว

1.2 วัตถุประสงค์

เพื่อนำเสนอวิธีการบีบอัดแบบจำลองการเรียนรู้เชิงลึกสำหรับแบบจำลองการตรวจจับวัตถุ YOLOv3 โดยมุ่งเน้นไปที่การบีบอัดแบบจำลองการเรียนรู้เชิงลึกด้วยวิธีการตัด (Pruning)

1.3 ขอบเขตในการดำเนินงาน

1. ชุดข้อมูลตรวจจับวัตถุที่ใช้งานจะประกอบด้วย 2 ชุดข้อมูลหลักคือ 1) UA-DETRAC เป็นชุดข้อมูลสำหรับตรวจจับยานพาหนะ 2) PASCAL VOC (2007+2012) เป็นชุดข้อมูลสำหรับการตรวจจับวัตถุทั่วไป
2. แบบจำลองหลักในงานวิจัยชิ้นนี้คือ YOLOv3
3. ใช้วิธีการตัดเป็นวิธีหลักในการบีบอัดแบบจำลอง
4. เปรียบเทียบประสิทธิภาพของแบบจำลองที่ถูกตัดด้วยวิธีการแบบพื้นฐาน และวิธีที่เสนอ

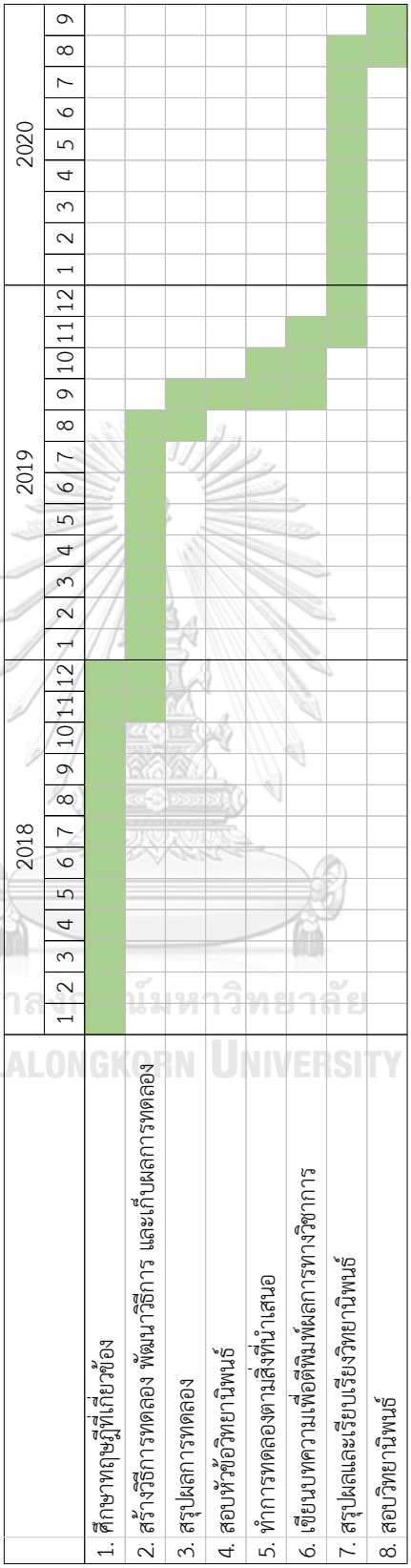
1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถลดขนาดของแบบจำลองและต้นทุนคำนวณของแบบจำลองการเรียนรู้เชิงลึกได้จากการบีบอัดแบบจำลอง
2. สามารถทำให้แบบจำลองการเรียนรู้เชิงลึกสำหรับการตรวจจับวัตถุมีความเหมาะสมที่จะนำไปใช้กับอุปกรณ์ขนาดเล็กมากยิ่งขึ้น

1.5 ขั้นตอนการดำเนินงานวิจัย

1. ศึกษาทฤษฎีที่เกี่ยวข้อง
2. สร้างวิธีการทดลอง พัฒนาแบบจำลอง และเก็บผลการทดลอง
3. สรุปผลการทดลอง
4. สอบหัวข้อวิทยานิพนธ์
5. ทำการทดลองตามสิ่งที่นำเสนอ
6. เขียนบทความเพื่อตีพิมพ์ทางวิชาการ
7. สรุปผลและเรียบเรียงวิทยานิพนธ์
8. สอบวิทยานิพนธ์

ตารางที่ 1 ขั้นตอนการดำเนินงาน

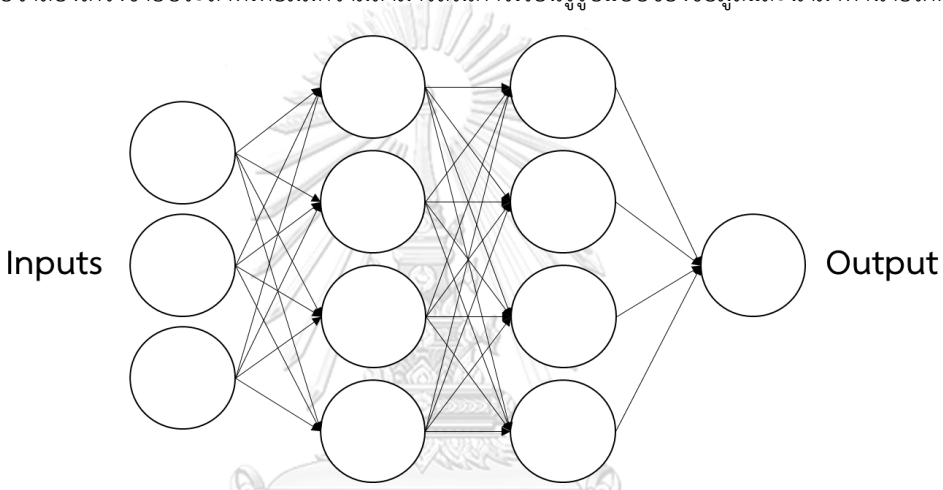


บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

ทฤษฎีที่เกี่ยวข้องของการออกแบบและพัฒนางานวิจัยนี้ ประกอบไปด้วยโครงข่ายประสาทเทียม การเรียนรู้เชิงลึก และการตัด

2.1 โครงข่ายประสาทเทียม

เป็นแบบจำลองการเรียนรู้ของเครื่องที่ได้รับแรงบันดาลใจมาจากโครงข่ายประสาทจากสมองของสิ่งที่มีชีวิต แบบจำลองโครงข่ายประสาทเทียมมีความสามารถในการเรียนรู้รูปแบบของข้อมูลและนำมาทำนายได้ดี



รูปที่ 1 แบบจำลองโครงข่ายประสาทเทียม

2.1.1 เพอร์เซปตรอน (Perceptron)

เพอร์เซปตรอนเป็นองค์ประกอบส่วนหนึ่งของโครงข่ายประสาทเทียม ซึ่งในหนึ่งโครงข่ายประสาทเทียมจะประกอบไปด้วยหลายเพอร์เซปตรอน โดยสามารถแสดงเป็นสมการได้ดังนี้

กำหนดให้ $f(x_1, x_2, x_3, \ldots, x_n)$ เป็นฟังก์ชันของเพอร์เซปตรอน โดย x_1 ถึง x_n เป็นข้อมูลรับเข้า โดยตัวฟังก์ชันสามารถคำนวณผลลัพธ์ออกมาได้ตามสมการดังนี้

$$\mathrm{f(x)} = \begin{cases} 1 & \text{if } \sum_{i=1}^{n} w_i x_i + \mathrm{b} > 0 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

กำหนดให้ w_i คือน้ำหนัก (weights) x_i คือข้อมูลนำเข้า และ b คือค่าไบแอส (bias) ซึ่งมีค่าเป็นค่าคงที่เป็นจำนวนจริง

2.1.2 ฟังก์ชันกระตุ้น (Activation Function)

ฟังก์ชันกระตุ้นคือฟังก์ชันสำหรับข้อมูลส่งออกของแต่ละเพอร์เซปตรอน โดยตัวฟังก์ชันกระตุ้นจะช่วยให้โครงข่ายประสาทเทียมสามารถแก้ไขปัญหาที่มีความซับซ้อนมากขึ้นได้ รวมถึงยังช่วยให้โครงข่ายประสาทเทียมสามารถเรียนรู้และทำการปรับค่าน้ำหนักได้ง่ายขึ้น โดยฟังก์ชันกระตุ้นที่ได้รับความนิยมจะมีดังต่อไปนี้

1. ฟังก์ชันซิกมอยด์ (Sigmoid Function)

เป็นฟังก์ชันที่มีค่าระหว่าง 0 ถึง 1

$$f(z) = \frac{1}{1 + e^{-z}} \tag{2}$$

2. ฟังก์ชันแทนเจต์ไฮเพอร์โบลิก (Hyperbolic Tangent Function)

เป็นฟังก์ที่เหมือนกับฟังก์ชันซิกมอยด์แต่มีค่าระหว่าง -1 ถึง 1

$$f(z) = \frac{2}{1 + e^{-z}} - 1 \tag{3}$$

3. ฟังก์ชันเรคติไฟต์เชิงเส้น (Rectified Linear Unit function หรือ ReLU)

เป็นฟังก์ชันที่มีค่าเป็นบวกเสมอ เหมาะสำหรับสมการที่มีค่าเป็นบวกเสมอ เช่น รูปภาพ

$$f(z) = \begin{cases} 0 \; if \; z < 0 \\ z \; if \; z \geq 0 \end{cases} \tag{4}$$

4. ฟังก์ชันค่าสูงสุดอย่างอ่อน (Softmax Function)

เป็นฟังก์ชันที่แปลงค่าของผลลัพธ์ให้ออกมาอยู่ในช่วง 0 ถึง 1 เปรียบเสมือนค่าความน่าจะเป็นของแต่ละผลลัพธ์

$$f(z)_j = \frac{e^{z_j}}{\sum_{i=1}^{k} e^{z_j}} - 1 \tag{5}$$

2.1.3 ฟังก์ชันต้นทุน (Cost Function หรือ Loss Function หรือ Objective Function)

คือฟังก์ชันที่ระบุค่าความผิดพลาดของโครงข่ายประสาทเทียม ซึ่งโครงข่ายประสาทเทียมจะนำค่าจากฟังก์ชันต้นทุนไปใช้ปรับค่าน้ำหนักเพื่อลดค่าความผิดพลาดของฟังก์ชันต้นทุนให้มีค่าต่ำที่สุด โดยฟังก์ชันต้นทุนที่เป็นที่นิยมใช้จะมีดังนี้

1. ค่าเฉลี่ยความผิดพลาดกำลังสอง (Mean Square Error หรือ MSE)

$$J = \frac{1}{n}\sum_{i=1}^{n} (\hat{y}_i - y_i)^2 \tag{6}$$

2. ค่าเฉลี่ยครอสเอนโทรปีแบบทวิภาค (Binary Cross-Entropy)

$$J = -\frac{1}{n}\sum_{i=1}^{n} y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - y_i) \tag{7}$$

2.1.4 การหาค่าที่เหมาะสมที่สุด (Optimization)

การหาค่าน้ำหนักที่เหมาะที่สุดของโครงข่ายประสาทเทียมสามารถทำได้ด้วยวิธีการเคลื่อนลงตามความชัน (Gradient Descent Algorithm) โดยมีสมการดังนี้

$$\widehat{w}_i = w_i + \Delta w_i \tag{8}$$

$$\Delta w_i = \eta(\hat{y}_i - y_i)x_i \tag{9}$$

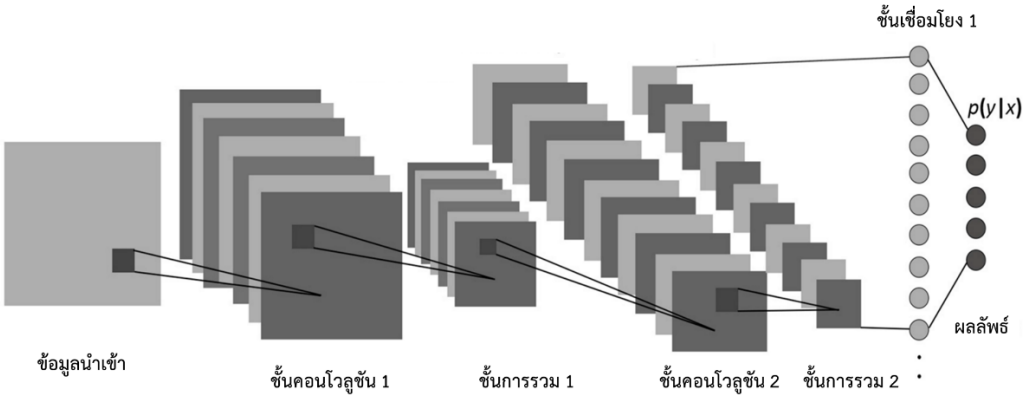
กำหนดให้ η คืออัตราการเรียนรู้

2.2 การเรียนรู้เชิงลึก

คือโครงข่ายประสาทเทียมที่ประกอบด้วยชั้นซ่อนตัว (Hidden Layer) เป็นจำนวนมาก ทำให้โครงข่ายประสาทเทียมสามารถเรียนรู้และสร้างฟีเจอร์ของข้อมูลที่เหมาะสมได้ ซึ่งแตกต่างกับโครงข่ายประสาทเทียมแบบเดิม ที่จำเป็นต้องมีฟีเจอร์ที่ใช้สำหรับให้โครงข่ายประสาทเทียมรับเข้าเป็นข้อมูลนำเข้า จุดนี้ส่งผลให้แบบจำลองการเรียนรู้เชิงลึกมีความสามารถในการเรียนรู้และจับรูปแบบของข้อมูลได้ดียิ่งขึ้น

2.2.1 โครงข่ายประสาทเทียมคอนโวลูชัน

เป็นโครงข่ายประสาทเทียมที่จัดเป็นเพอร์เซปตรอนหลายชั้นชนิดหนึ่ง มีจุดเริ่มต้นมาจากงานวิจัยด้านการจำแนกรูปภาพตัวอักษร โดยโครงข่ายประสาทเทียมคอนโวลูชันออกแบบมาเพื่อให้สามารถลดจำนวนของพารามิเตอร์ลง โครงข่ายประสาทเทียมแบบคอนโวลูชันจะใช้ตัวกรอง (Filter) เพื่อแปลงเป็นข้อมูลเพื่อใช้ในชั้นถัดไป



รูปที่ 2 ตัวอย่างชั้นคอนโวลูชัน

[ที่มา: https://towardsdatascience.com/how-to-teach-a-computer-to-see-with-convolutional-neural-networks-96c120827cd1 Accessed: Aug 13 2019]

2.2.1.1 ชั้นคอนโวลูชัน (Convolution Layers)

เป็นชั้นของโครงข่ายประสาทเทียมที่ทำหน้าที่สกัดฟีเจอร์จากข้อมูลรับเข้าตามขอบเขตที่กำหนดไว้ โดยจะใช้หลักที่สำคัญสองอย่างคือ 1) การเชื่อมต่อเฉพาะบางส่วน คือการที่ชั้นคอนโวลูชันรับผลลัพธ์จากชั้นก่อนหน้านี้เพียงบางส่วน 2) การเชื่อมใช้พารามิเตอร์ร่วมกัน คือการที่ชั้นของคอนโวลูชันที่การใช้ผลลัพธ์จากนิวรอนตัวเดียวกันซ้ำหลายครั้งตามส่วนที่ซ้อนทับกัน ซึ่งช่วยให้เมื่อมีผลลัพธ์จากนิวรอนที่มีความสำคัญ ก็จะทำให้มีการใช้ผลลัพธ์จากนิวรอนตัวนี้หลายครั้งได้ ซึ่งทำสามารถลดปริมาณพารามิเตอร์ที่จำเป็นต้องใช้ในแบบจำลองได้

- ขนาดของตัวกรอง (Filter)

  คือขนาดของตัวกรองซึ่งประกอบด้วยความกว้างและความยาวสำหรับใช้ทำคอนโวลูชันกับข้อมูลนำเข้า เช่นในรูปที่ 3 ที่ขนาดของตัวกรองมีขนาดเป็น 2x2

- ชนิดของการทำคอนโวลูชัน (Convolution Type)

  1. คอนโวลูชันแบบแคบ (Narrow Convolution)

     คือการทำคอนโวลูชันโดยที่ไม่เกินขอบเขตมิติของขนาดเมตริกซ์ข้อมูลนำเข้า โดยหากกำหนดให้ NxN คือขนาดของข้อมูลนำเข้า และ MxM คือขนาดของตัวกรอง ขนาดของผลลัพธ์จะเท่ากับ (N-M+1)x(N-M+1)

  2. คอนโวลูชันแบบกว้าง (Wide Convolution)

     คือการทำคอนโวลูชันโดยมีการทำเกินขอบเขตของข้อมูลนำเข้า โดยที่ส่วนที่เกินกว่าขอบเขตของข้อมูลนำเข้า จะถูกแทนค่าด้วย 0 ซึ่งเรียกว่าการเสริมเติม (Padding) โดยหากกำหนดให้ NxN คือขนาดของข้อมูลนำเข้า และ MxM คือขนาดของตัวกรอง ขนาดของผลลัพธ์ที่ได้จะเท่ากับ (N+M-1)x(N+M-1) โดยจุดประสงค์ของการทำคอนโวลูชันแบบกว้างจะมีจุดประสงค์เพื่อป้องกันไม่ให้ขนาดของมิติข้อมูลเปลี่ยนแปลงไปจากเดิม

- ขนาดของการก้าวข้าม (Stride Size)

  คือจำนวนมิติของข้อมูลรับเข้า ที่จะถูกเลื่อนเพื่อคำนวณค่ากับตัวกรอง ซึ่งตามปกติแล้วจะใช้ค่าเป็น 1 การที่มีขนาดก้าวข้ามมากกว่า 1 ก็จะทำให้ขนาดของผลลัพธ์เล็กลงไปด้วย

- จำนวนตัวกรอง (Number of Filters)

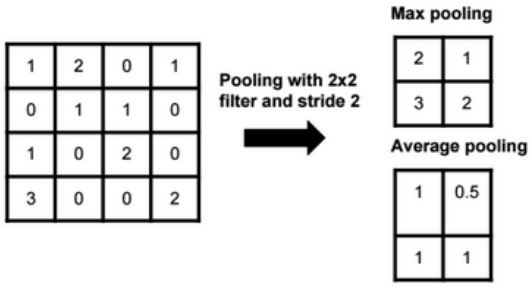
  คือจำนวนของตัวกรองที่ต้องการสำหรับชั้นของคอนโวลูชัน โดยความลึกของผลลัพธ์จะเท่ากับจำนวนของตัวกรอง

- จำนวนช่องสัญญาณ (Channel)

  จำนวนช่องสัญญาณ คือความลึกของข้อมูลนำเข้า เช่น สำหรับรูปภาพชนิดสามสี ประกอบด้วยสีแดง เขียว และน้ำเงิน จะถือว่ามีจำนวนช่องสัญญาณเท่ากับ 3

2.2.1.2 ชั้นการรวม (Pooling Layers)

เป็นชั้นที่ทำหน้าที่ลดมิติของข้อมูลด้วยการคัดใจความสำคัญของข้อมูลออกมา ซึ่งโดยทั่วไปจะใช้ควบคู่กับชั้นคอนโวลูชันเพื่อลดต้นทุนในการคำนวณ ประเภทของชั้นการรวมที่ได้รับความนิยมจะประกอบไปด้วยการเลือกข้อมูลที่มีค่ามากที่สุด (Max Pooling) และ ค่าเฉลี่ย (Average Pooling) โดยการเลือกข้อมูลจะเลือกข้อมูลจากพื้นที่เมตริกตามขนาดที่กำหนดไว้

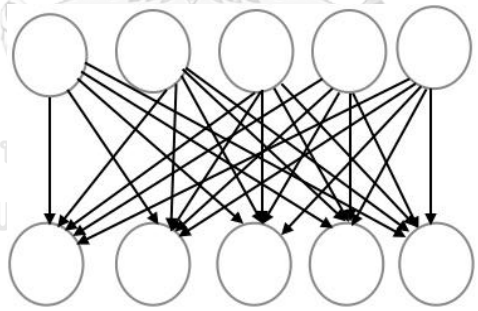


รูปที่ 3 ตัวอย่างชั้นการรวมแบบค่ามากที่สุดและค่าเฉลี่ย

[ที่มา: http://sqlml.azurewebsites.net/2017/09/12/convolutional-neural-network Accessed: Aug 13, 2019]

2.2.1.3 ชั้นเชื่อมโยง (Fully Connected Layers)

ชั้นเชื่อมโยงคือชั้นที่มีลักษณะการเชื่อมต่อกับชั้นก่อนหน้า โดยเพอร์เซปตรอนแต่ละตัวของชั้นเชื่อมโยงจะต่อกับกับทุกผลลัพธ์ในชั้นก่อนหน้า



รูปที่ 4 ชั้นเชื่อมโยง

[ที่มา: https://cmsc426spring2019.github.io/2019/proj/p4/ Accessed: Aug 13, 2019]

2.3 mAP (Mean Average Precision)

เป็นตัววัด (Metric) สำหรับวัดความแม่นยำของแบบจำลองตรวจจับวัตถุ โดยในการคำนวณค่ามาตรฐานตัวนี้จะเริ่มจากการคำนวณความแม่นยำ (Precision) รีคอล (Recall) และ IoU (Intersection over Union)

2.3.1 ความแม่นยำ (Precision)

เป็นหน่วยวัดที่ใช้วัดความแม่นยำของแบบจำลอง เช่น อัตราความแม่นยำที่แบบจำลองตรวจจับวัตถุได้ถูกต้องตรงตามผลเฉลยเป็นร้อยละ ซึ่งสามารถคำนวณได้จากสมการที่ (10)

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \tag{10}$$

กำหนดให้ $True\ Positives$ คือ ผลบวกจริง และ $False\ Positives$ คือ ผลบวกปลอม

2.3.2 รีคอล (Recall)

เป็นหน่วยวัดที่ใช้วัดความแม่นยำเฉพาะส่วนที่มีค่าเป็นจริง สามารถคำนวณได้จากสมการที่ (11)

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{11}$$

กำหนดให้ $False\ Negatives$ คือ ผลลบปลอม

2.3.3 IoU (Intersection over Union)

เป็นวิธีการวัดผลการตรวจจับวัตถุจากแบบจำลองว่าตรวจจับวัตถุได้ถูกต้องหรือไม่ ซึ่งวัดจากสัดส่วนการซ้อนทับกันระหว่างขอบเขตของกล่องซึ่งเป็นผลลัพธ์จากแบบจำลองและกล่องผลเฉลย โดยจะถือว่าการตรวจจับวัตถุถูกต้องก็ต่อเมื่อค่าของ IoU มากกว่าค่าขีดแบ่ง (Threshold) โดยค่า IoU สามารถคำนวณได้จากสมการที่ (12)

$$\text{IoU} = \frac{Area\ of\ Overlap}{Area\ of\ Union} \tag{12}$$

กำหนดให้ $Area\ of\ Overlap$ คือ พื้นที่ที่ทับกันระหว่างกล่องซึ่งเป็นผลลัพธ์จากแบบจำลองและกล่องผลเฉลย และ $Area\ of\ Union$ คือ พื้นที่ส่วนรวมของกล่องซึ่งเป็นผลลัพธ์จากแบบจำลองและกล่องผลเฉลย

2.3.4 AP (Average Precision)

เป็นค่าเฉลี่ยของความแม่นยำของทุก ๆ วัตถุที่อยู่ในรูปภาพสำหรับแต่ละคลาส โดยในการคำนวณจะต้องแยกคำนวณทีละคลาสเท่านั้น ซึ่งสามารถคำนวณได้จากสมการ (13)

$$AP = \sum (r_{n+1} - r_n)\ p_{interp}(r_{n+1}) \tag{13}$$

$$p_{interp}(r_{n+1}) = max_{\tilde{r}:\tilde{r} \geq r_{n+1}} p(\tilde{r}) \tag{14}$$

กำหนดให้ AP คือ ค่าความแม่นยำเฉลี่ยของวัตถุคลาสหนึ่งในทุก ๆ รูป

n คือ อันดับของข้อมูล

r_n คือ ค่ารีคอล ณ อันดับข้อมูล n

p คือ ค่าความแม่นยำ

p_{interp} คือ ค่าความแม่นยำที่ผ่านการแทรก (interpolate)

$max_{\tilde{r} \geq r_{n+1}} p(\tilde{r})$ คือ ค่าความแม่นยำที่มีค่าสูงสุด ณ ค่าของรีคอลที่มากกว่าเท่ากับ r_{n+1}

2.3.5 mAP (Mean Average Precision)

เป็นค่าเฉลี่ยของ AP ของทุก ๆ คลาส ซึ่งสามารถคำนวณได้จากสมการ (15)

$$\text{mAP} = \frac{1}{c}\sum_{1}^{c} AP \tag{15}$$

กำหนดให้ mAP คือ ค่าเฉลี่ยของ AP ของทุก ๆ คลาส c คือ จำนวนคลาสทั้งหมด

บทที่ 3
งานวิจัยที่เกี่ยวข้อง

3.1 การตรวจจับวัตถุ (Object Detection)

ในอดีตแบบจำลองการตรวจจับวัตถุด้วยการเรียนรู้ด้วยเครื่องจำเป็นต้องพึ่งคุณลักษณะที่ถูกสร้างขึ้นโดยมนุษย์ เช่น ฟีเจอร์ HOG หลังจากที่แบบจำลองการเรียนรู้เชิงลึกเริ่มได้รับความนิยม จึงมีการเปลี่ยนจากการใช้งานฟีเจอร์ที่ถูกสร้างโดยมนุษย์มาเป็นฟีเจอร์ที่ถูกเรียนรู้และสร้างโดยแบบจำลองการเรียนรู้เชิงลึก เช่น ชั้นคอนโวลูชัน สำหรับแบบจำลองการเรียนรู้เชิงลึกที่ถือเป็นก้าวสำคัญสำหรับการตรวจจับวัตถุ คือ R-CNN โดยแบบจำลองตัวนี้ได้นำฟีเจอร์ที่สร้างโดยชั้นคอนโวลูชันมาใช้แทนที่ฟีเจอร์ที่ถูกสร้างโดยมนุษย์ ทำให้แบบจำลองมีความแม่นยำที่สูงขึ้นมาก ซึ่งในเวลาถัดมา Fast R-CNN [6] และ Faster R-CNN [7] ก็ถูกพัฒนาขึ้นในฐานะรุ่นปรับปรุงของ R-CNN ซึ่งมีการพัฒนาทั้งในด้านของความแม่นยำและความเร็ว ต่อมา Redmon, et al [8] ได้นำเสนอ YOLO ซึ่งเป็นแบบจำลองการเรียนรู้เชิงลึกสำหรับตรวจจับวัตถุแบบหนึ่งขั้นตอนเป็นตัวแรกบนการเรียนรู้เชิงลึก โดย YOLO เป็นแบบจำลองที่ถูกพัฒนาขึ้นมาเพื่อเน้นที่ความเร็วโดยที่ยังสามารถคงความแม่นยำในระดับที่ดี จุดแตกต่างที่สำหรับของแบบจำลอง YOLO และ R-CNN คือ แบบจำลอง YOLO ได้เปลี่ยนการมองโจทย์ปัญหาการตรวจจับวัตถุให้กลายเป็นโจทย์สมการถดถอย (Regression) ส่งผลให้แบบจำลอง YOLO สามารถทำงานได้เร็วและมีความแม่นยำที่ดีจากการออกแบบ ต่อมา YOLOv2 [13] และ YOLOv3 [14] ก็ถูกพัฒนาขึ้นมาในฐานะรุ่นปรับปรุงของ YOLO ถึงแม้ว่าแบบจำลองตระกูล YOLO จะสามารถทำงานได้เร็วเมื่อเทียบกับแบบจำลองการเรียนรู้เชิงลึกตัวอื่น ๆ แต่แบบจำลองก็ยังถือว่าช้าและใช้งานหน่วยความจำปริมาณมากสำหรับอุปกรณ์ขนาดเล็ก เนื่องจากการที่แบบจำลอง YOLO ยังคงมีปัญหาเรื่องการมีตัวแปรน้ำหนักที่มากเกินความจำเป็น ซึ่งทำให้การใช้งานกับอุปกรณ์ขนาดเล็กยังคงเป็นไปได้ยาก

3.1.1 แบบจำลอง YOLOv3

ในปี 2015 [8] ได้นำเสนอแบบจำลองการเรียนเชิงลึกสำหรับตรวจจับวัตถุ YOLO ซึ่งจัดเป็นแบบจำลองการเรียนรู้เชิงลึกสำหรับตรวจจับวัตถุแบบหนึ่งขั้นตอนตัวแรก โดยการเปลี่ยนรูปแบบการสร้างสี่เหลี่ยมในการปิดล้อมวัตถุ (Bounding Box) จากการใช้โครงข่ายเสนอขอบเขตเป็นจากสร้างสี่เหลี่ยมในการปิดล้อมวัตถุโดยตรงจากตัวโครงข่ายประสาทเทียม ด้วยการเปลี่ยนมุมมองปัญหาการตรวจจับวัตถุเป็นแบบสมการถดถอย ด้วยเหตุนี้แบบจำลอง YOLO จึงสามารถทำงานได้เร็ว แต่ทว่าความแม่นยำยังคงต่ำกว่าแบบจำลองสองขั้นตอนอย่าง Faster-RCNN ต่อมาในปี 2017 Redmon and Farhadi [13] ได้นำเสนอ YOLOv2 ซึ่งเป็นรุ่นปรับปรุงจากตัวแบบจำลองตัวเดิมโดยเพิ่มความสามารถด้านความเร็วและความแม่นยำขึ้นจากตัวเดิม โดยมีการเพิ่มความละเอียดของภาพนำเสนอการใช้กล่องสมอ (Anchor Box) คำนวณขนาดของกล่องสมอด้วยการแบ่งกลุ่มแบบเคมีน ปรับการคำนวณตำแหน่งของสี่เหลี่ยมในการปิดล้อมวัตถุด้วยฟังก์ชันกระตุ้นแบบโลจิสติค ต่อมาในปี 2018 ได้มีการนำเสนอ YOLOv3 ซึ่งเป็นรุ่นปรับปรุงต่อจาก YOLOv2 โดยมีการนำโครงข่ายพีระมิดฟีเจอร์เข้ามาใช้งานเพื่อเพิ่มความสามารถในการตรวจจับวัตถุขนาดเล็ก ในงานวิจัยชิ้นนี้จะใช้แบบจำลอง YOLOv3 เป็นแบบจำลองหลักสำหรับงานวิจัยชิ้นนี้ เนื่องจากมีต้นทุนในการคำนวณต่ำ เมื่อเปรียบเทียบกับความแม่นยำที่ได้ ซึ่งมีความเหมาะสมสำหรับการใช้สำหรับการตรวจจับวัตถุแบบทันทีบนอุปกรณ์ขนาดเล็ก

ในงานวิจัยชิ้นนี้ จะเลือกใช้งานแบบจำลอง YOLOv3 เป็นแบบจำลองหลัก โดยจะเลือกใช้ MobileNet เป็นโครงข่ายกระดูกสันหลังเพื่อใช้ในการสกัดฟีเจอร์จากรูปภาพ และต่อด้วยโครงข่ายพีระมิดฟีเจอร์ สำหรับโครงสร้างของแบบจำลองทั้งหมดจะถูกแสดงไว้ในรูปที่ 5

	Type	Stride	Kernel	Channels	
	conv + bn + ReLU	2	3 x 3	32	MobileNet
	dw conv block (1, 64)	-	-	64	
	dw conv block (2, 128)	-	-	128	
	dw conv block (1, 128)	-	-	128	
	dw conv block (2, 256)	-	-	256	
	dw conv block (1, 256)	-	-	256	
	dw conv block (2, 512)	-	-	512	
	dw conv block (1, 512)	-	-	512	
x5	dw conv block (2, 1024)	-	-	1024	
	dw conv block (2, 1024)	-	-	1024	
	dw conv block (2, 1024)	-	-	1024	
	fpn block (512)	-	-	512	Feature Pyramid Network
	conv + bn + LReLU	1	3 x 3	1024	
	conv (output 1)	1	1 x 1	-	
	conv + bn + LReLU	1	1 x 1	1024	
	upsampling2d	-	-	-	
	concatenate	-	-	-	
	fpn block (256)	-	-	256	
	conv + bn + LReLU	1	3 x 3	512	
	conv (output 2)	1	1 x 1	-	
	conv + bn + LReLU	1	1 x 1	128	
	upsampling2d	-	-	-	
	concatenate	-	-	-	
	fpn block (128)	-	-	128	
	conv + bn + LReLU	1	3 x 3	256	
	conv (output 3)	1	1 x 1	-	

dw conv block (m, n)

convdw + bn + ReLU	m	3 x 3	-
conv + bn + ReLU	1	1 x 1	n

fpn block (n)

conv + bn + LReLU	1	1 x 1	n
conv + bn + LReLU	1	3 x 3	n x 2
conv + bn + LReLU	1	1 x 1	n
conv + bn + LReLU	1	3 x 3	n x 2
conv + bn + LReLU	1	1 x 1	n

รูปที่ 5 โครงสร้างแบบจำลอง YOLOv3 ที่ใช้ในงานวิจัยชิ้นนี้

3.1.2 RetinaNet

ในปี 2017 Lin, Goyal, Girshick, He and Dollár [10] ได้นำเสนอแบบจำลอง RetinaNet ซึ่งเป็นแบบจำลองการตรวจจับวัตถุแบบหนึ่งขั้นตอน โดยแบบจำลอง RetinaNet ได้นำฟังก์ชันต้นทุนแบบใหม่เข้ามาใช้งาน เพื่อเพิ่มประสิทธิภาพให้กับแบบจำลองด้วยการใช้งานฟังก์ชันต้นทุนจุดรวม (Focal Loss) โดยฟังก์ชันต้นทุนแบบใหม่ เพื่อที่จะถูกนำมาใช้เพื่อแก้ไขปัญหาความไม่สมดุลกันระหว่างคลาสพื้นหน้า (Foreground Class) และคลาสพื้นหลัง (Background Class) โดยตัว RetinaNet ถือเป็นแบบจำลองการเรียนรู้เชิงลึกสำหรับการตรวจจับวัตถุแบบขั้นตอนเดียวที่มีความแม่นยำใกล้เคียงกับ YOLOv3 สำหรับการตรวจจับวัตถุที่ไม่ต้องการความแม่นยำของตำแหน่งของสี่เหลี่ยมในการปิดล้อมวัตถุสูงมากนัก และมีความแม่นยำสูงกว่าสำหรับกรณีที่ต้องการเน้นที่ความแม่นยำของตำแหน่งของสี่เหลี่ยมในการปิดล้อมวัตถุ แต่ทว่าจุดด้อยแบบจำลอง RetinaNet คือยังมีความเร็วที่ช้ากว่าแบบจำลอง YOLOv3 เมื่อเทียบในระดับความแม่นยำที่เท่ากัน

3.2 วิธีการบีบอัดแบบจำลองการเรียนรู้เชิงลึก (Deep Learning Compression Techniques)

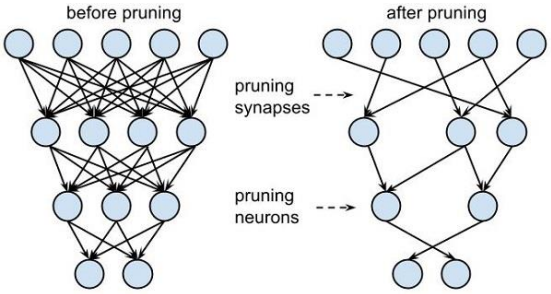
การบีบอัดแบบจำลองการเรียนรู้เชิงลึก คือวิธีการที่จะช่วยเพิ่มความเร็วให้กับแบบจำลองการเรียนรู้เชิงลึก โดยสามารถแบ่งได้ออกเป็น 3 กลุ่ม [15] คือ 1) การแบ่งนับและการทวิภาคแบบจำลอง (Model Quantization and Binarization) 2) การตัดและการแบ่งปัน (Pruning and Sharing) และ 3) เมทริกซ์โครงสร้าง (Structural Matrix) โดยวิธีการที่นิยมคือการตัดเนื่องจากสามารถช่วยเพิ่มความเร็วและลดขนาดของแบบจำลองการเรียนรู้เชิงลึกได้

3.2.1 การตัด (Pruning)

ในงานวิจัย [11] ได้เริ่มการทดลองเกี่ยวกับการบีบอัดแบบจำลองโครงข่ายประสาทเทียมด้วยวิธีการตัดตัวแปรน้ำหนักที่มีความสำคัญต่ำออกจากแบบจำลองโครงข่ายประสาทเทียม โดยมีจุดประสงค์เพื่อลดจำนวนตัวแปรและลดต้นทุนคำนวณ (Computational Cost) ของแบบจำลอง โดยที่ความแม่นยำไม่เปลี่ยนแปลงไปจากเดิมมากนัก ซึ่งทำให้ได้แบบจำลองโครงข่ายประสาทเทียมที่สามารถทำงานได้เร็วขึ้น และมีขนาดที่เล็กลง แต่มีความแม่นยำใกล้เคียงเดิม

3.2.1.1 การตัดแบบไร้โครงสร้าง (Unstructured Pruning)

การบีบอัดโครงข่ายประสาทเทียมด้วยวิธีการตัดในช่วงแรกอย่าง [11] และ [16] เป็นการตัดตัวแปรน้ำหนักแบบไม่เชิงโครงสร้าง กล่าวคือ เป็นการตัดตัวแปรน้ำหนักที่เน้นไปที่การเลือกและตัดค่าน้ำหนักที่ไม่มีความสำคัญออกจากแบบจำลองโครงข่ายประสาทเทียมโดยไม่คำนึงถึงโครงสร้างและมิติของข้อมูลในแต่ละชั้นของแบบจำลองโครงข่ายประสาทเทียม ส่งผลให้แบบจำลองที่ผ่านการตัดจะมีลักษณะมากเลขศูนย์ (Sparse) และทำให้มีโอกาสเกิดความเสียหายขึ้นกับแบบจำลองโครงข่ายประสาทเทียม ทำให้แบบจำลองไม่สามารถนำมาใช้งานบนซอฟต์แวร์และฮาร์ดแวร์การเรียนรู้เชิงลึกทั่วไปได้อย่างมีประสิทธิภาพซึ่งส่งผลให้แบบจำลองไม่ได้เร็วขึ้นมากนัก

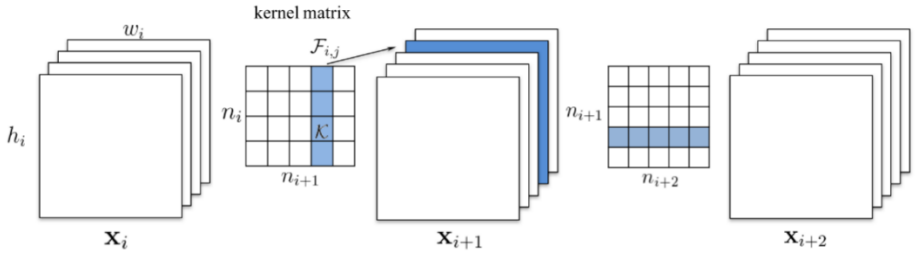


รูปที่ 6 การตัดโครงข่ายประสาทเทียม

[ที่มา: https://www.aitrends.com/ai-insider/deep-compression-pruning-machine-learning-ai-self-driving-cars-using-convolutional-neural-networks-cnn/ Accessed: Aug 13, 2019]

3.2.1.2 การตัดแบบเชิงโครงสร้าง (Structured Pruning)

ในปี 2017 งานวิจัย [12] ได้นำเสนอการตัดแบบเชิงโครงสร้างเพื่อแก้ปัญหาเรื่องลักษณะมากเลขศูนย์ (Sparse) ของตัวแปรน้ำหนักจากวิธีการตัดแบบไร้โครงสร้าง โดยจุดประสงค์หลักของวิธีการนี้คือการตัดตัวแปรน้ำหนักของแบบจำลองการเรียนรู้เชิงลึกที่ไม่มีความสำคัญออกเหมือนกับวิธีแรก แต่ว่าจะมีแตกต่างจากการตัดแบบไร้โครงสร้างในจุดที่ การตัดชนิดนี้จะคำนึงถึงมิติของข้อมูลในแต่ละชั้น ทำให้การตัดตัวแปรน้ำหนักไม่สร้างความเสียหายกับแต่ละชั้นของแบบจำลองที่ต้องการบีบอัด ซึ่งส่งผลให้แบบจำลองที่ทำการบีบอัดด้วยวิธีนี้สามารถใช้งานซอฟต์แวร์และฮาร์ดแวร์ตามท้องตลาดได้อย่างมีประสิทธิภาพ ทำให้วิธีการตัดในรูปแบบนี้จึงมีความเหมาะสมที่จะนำไปใช้งานจริงมากกว่าการตัดแบบไร้โครงสร้าง



รูปที่ 7 แสดงความสัมพันธ์ของการเปลี่ยนแปลงของชั้นคอนโวลูชันเพื่อทำการตัดตัวกรองออก

[ที่มา Fig.1 ใน [17]]

3.2.2 เกณฑ์การจัดอันดับ (Ranking Criteria)

คือเกณฑ์ที่ใช้ในการวัดระดับความสำคัญของตัวแปรน้ำหนักแต่ละตัวแปร โดยแหล่งที่มาที่เกณฑ์จัดอันดับใช้ในการคำนวณหรือประเมินจะประกอบไปด้วย 1) Magnitude และ 2) APoZ

3.2.2.1 Magnitude

งานวิจัย [17] ได้นำเสนอวิธีการใช้ค่าน้ำหนักเพื่อใช้ในการประเมินค่าความสำคัญของตัวแปรน้ำหนักแต่ละตัวในแบบจำลองโครงข่ายประสาทเทียม โดยผลลัพธ์ของวิธีการนี้ คือ ค่าระดับความสำคัญของตัวกรอง (Filter) ของชั้นคอนโวลูชัน โดยที่ถ้าผลลัพธ์มีค่ามากจะหมายความว่า ตัวกรองมีความสำคัญมาก และในทางกลับกันถ้าหากผลลัพธ์มีค่าน้อยก็จะหมายความว่าตัวกรองมีความสำคัญต่ำ จุดเด่นของวิธีการนี้คือ 1) มีการทำงานที่เรียบง่าย 2) ไม่จำเป็นต้องชุดข้อมูลที่ใช้ฝึกสอนโครงข่ายประสาทเทียมในการทำงาน และ 3) มีประสิทธิภาพดีในระดับนึง โดยมีสมการดังนี้

$$\sum |W(i, :, :, :)| \tag{16}$$

กำหนดให้ i คือมิติของตัวกรองของชั้นคอนโวลูชัน W คือค่าน้ำหนัก

3.2.2.2 APoZ (Average Percentage of Zeros)

งานวิจัย [18] ได้นำเสนอวิธีการใช้ค่าจากฟังก์ชันกระตุ้นเพื่อใช้ในการประเมินความสำคัญของค่าน้ำหนักแต่ละตัวในแบบจำลอง ซึ่งในการได้มาของค่าจากฟังก์ชันกระตุ้นจะได้มาจากการใช้ชุดข้อมูลตรวจสอบ (Validation Data) โดยผลลัพธ์ของวิธีการนี้ คือ ค่าระดับความสำคัญของตัวกรอง (Filter) ของชั้นคอนโวลูชัน ซึ่งถ้าหากว่าผลลัพธ์มีค่ามากจะหมายความว่า ตัวกรองมีความสำคัญมาก และในทางกลับกันถ้าหากผลลัพธ์มีค่าน้อยก็จะหมายความว่าตัวกรองมีความสำคัญต่ำ โดยมีสมการดังนี้

$$\frac{1}{N}\sum |Sparsity(I(i, :, :))| \tag{17}$$

กำหนดให้ i คืออันดับของตัวกรองในชั้นคอนโวลูชัน I คือค่าผลลัพธ์จากฟังก์ชันกระตุ้น Sparsity คือสัดส่วนค่าที่มีค่าเท่ากับศูนย์ และ N คือจำนวนชุดข้อมูลตรวจสอบ

3.2.3 การตัดเชิงวนซ้ำ (Iterative Pruning)

งานวิจัย [19] ได้นำเสนอการตัดเชิงวนซ้ำ ซึ่งเป็นรูปแบบการตัดที่เน้นเพื่อให้การตัดยังคงมีความแม่นยำสูงสุด โดยการทำการตัดน้ำหนักของตัวแบบจำลองสลับกับการฝึกสอนกับชุดข้อมูลใหม่ โดยขั้นตอนการทำงานสามารถแบ่งออกได้เป็น 4 ขั้นตอนดังนี้

1) ทำการจัดอันดับตัวกรองและเลือกตัวกรองที่มีความสำคัญต่ำสุด
2) ตัดตัวกรองที่ถูกเลือกไว้ในขั้นตอนขั้นตอนที่ 1) ออกจากแบบจำลองโครงข่ายประสาทเทียม
3) นำแบบจำลองโครงข่ายประสาทเทียมที่ผ่านการตัดค่าน้ำหนักมาฝึกสอนต่อ
4) ทำขั้นตอน 1 – 3 ซ้ำ

รูปแบบการตัดรูปแบบนี้ถึงแม้จะใช้เวลามากกว่าการตัดแบบครั้งเดียว แต่ถือเป็นวิธีที่สามารถคงความแม่นยำของตัวแบบจำลองได้ดีที่สุด

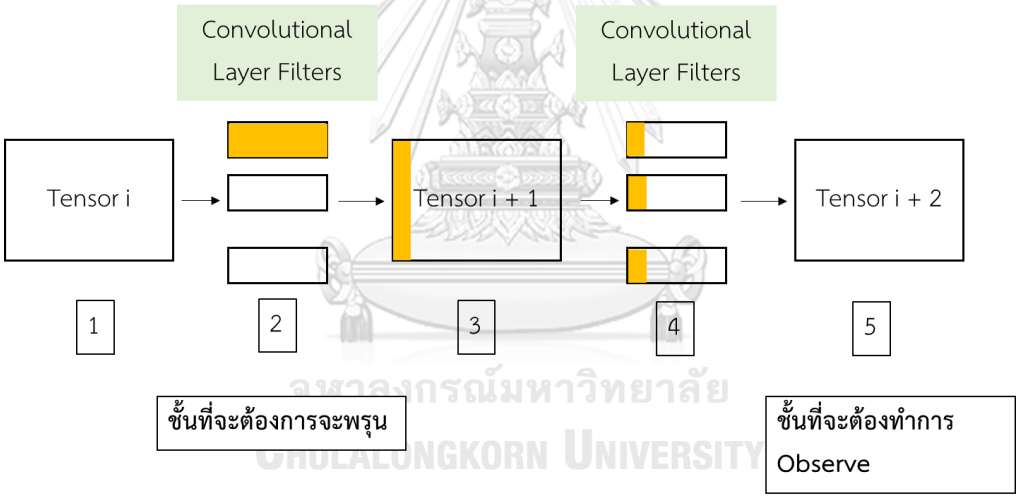
3.3 งานวิจัยที่เกี่ยวข้องกับการตัด

3.3.1 การบีบอัดแบบจำลองการตรวจจับวัตถุ

ในปี 2019 Singh, Manikandan, Matiyali and Namboodiri [20] ได้นำเสนอการบีบอัดแบบจำลองการตรวจจับวัตถุ SSD ด้วยวิธีการตัดด้วยค่าน้ำหนัก ผสมกับการฝึกสอนตัวแบบจำลองด้วยฟังก์ชันต้นทุนแบบเฉพาะ เพื่อให้ค่าน้ำหนักของแบบจำลองมีลักษณะมากเลขศูนย์ (Sparse) และตัดส่วนที่มีค่าน้ำหนักที่มีค่าเป็นศูนย์ออก เพื่อลดขนาดแบบจำลอง วิธีการของงานวิจัยชิ้นนี้จะใช้การตัดแบบครั้งเดียว ซึ่งแตกต่างกับงานวิจัยชิ้นอื่นที่จะใช้การตัดแบบวนซ้ำ งานวิจัยชิ้นนี้จะไม่นำวิธีการนี้มาเปรียบเทียบเนื่องจากในงานวิจัยชิ้นนี้จะเน้นที่มีรูปแบบการตัดแบบวนซ้ำ

3.3.2 ThiNet

ในปี 2017 Luo, Wu and Lin [21] ได้นำเสนอเกณฑ์การตัดด้วยการใช้ค่าจากฟังก์ชันกระตุ้น โดยจะเน้นไปที่การตัดตัวกรองของน้ำหนักที่ทำให้ค่าของชั้นถัดไปมีการเปลี่ยนแปลงน้อยที่สุด ในตัวงานนี้ได้ทำการทดลองเกณฑ์การตัด ThiNet บนแบบจำลองสำหรับการจำแนกวัตถุที่ประกอบด้วยชั้นคอนโวลูชันเป็นหลัก ในวิจัยชิ้นนี้จะนำ ThiNet เข้ามาใช้เพื่อเป็นตัวเกณฑ์การจัดอันดับตัวหนึ่ง



รูปที่ 8 ภาพรวมการทำงานของเกณฑ์การจัดอันดับ ThiNet

จากรูปที่ 8 ประกอบด้วยหมายเลข 1 ถึง 5 คือ 1) ข้อมูลนำเข้า ณ เทนเซอร์ (Tensor) i 2) ชั้นคอนโวลูชัน i+1 ที่ต้องการตัด 3) เทนเซอร์ i+1 ซึ่งเป็นผลลัพธ์จากหมายเลข 2 4) ชั้นคอนโวลูชัน i+2 และ 5) เทนเซอร์ i+2 ซึ่งเป็นผลลัพธ์จากหมายเลข 4 ซึ่งส่วนนี้จะถูกใช้งานเพื่อตรวจสอบผลกระทบของการตัดกับแบบจำลอง โดยที่ส่วนที่มีสี ณ หมายเลข 2 คือ ตัวกรองที่ต้องการตัด โดยที่ส่วนที่มีสีในส่วนอื่น ๆ คือ ตัวแปรที่มีความเกี่ยวข้องกับตัวกรองที่ต้องการตัดออก

ในกระบวนการทำงานของ ThiNet จะมีหลักการคือ ต้องการหาเซตย่อยของตัวกรองตามปริมาณที่สนใจที่ทำให้ค่าที่หมายเลข 5 มีขนาดการเปลี่ยนแปลงไปน้อยที่สุด โดยสมการที่ใช้วัดขนาดการเปลี่ยนแปลงของแบบจำลองสามารถเขียนได้ในรูปสมการที่ (18)

กำหนดให้ $\hat{y}$ คือ ขนาดของการเปลี่ยนแปลงซึ่งเลือกมาจากจุดของข้อมูลจากหมายเลข 5 ซึ่งได้มาจากการสุ่มเพื่อใช้เป็นตัวแทนเพื่อตรวจสอบการเปลี่ยนแปลงของแบบจำลองเมื่อมีการตัดตัวกรองออก $\hat{x}_c$ คือ ผลรวมของข้อมูลนำเข้าคูณกับตัวแปรน้ำหนัก เฉพาะส่วนที่เกี่ยวข้องกับ $\hat{y}$ โดย c คือ ช่องสัญญาณ และ S คือ เซตย่อยของตัวกรองที่ต้องการตัดออก

$$\hat{y} = \sum_{c \in S} \hat{x}_c \tag{18}$$

หลังจากได้เซตย่อยของตัวกรองที่ต้องการตัดออกแล้วก็จะมีการคำนวณตัวปรับค่าสำหรับตัวแปรน้ำหนักเพื่อลดขนาดของการเปลี่ยนแปลงให้เล็กลงไปอีก สามารถเขียนเป็นฟังก์ชันจุดประสงค์ได้ตามสมการที่ (19)
กำหนดให้ $\widehat{w}$ คือ ผลลัพธ์ค่าสัมประสิทธิ์เพื่อใช้ปรับขนาดของตัวแปรน้ำหนักโดยมีจุดประสงค์เพื่อลดขนาดค่าของสมการที่ (19) ให้ได้เล็กที่สุด y_i คือ ขนาดของการเปลี่ยนแปลงซึ่งเลือกมาจากจุดของข้อมูลจากหมายเลข 5 ตามรูปที่ 8 ซึ่งได้มาจากการสุ่มเพื่อใช้เป็นตัวแทนเพื่อตรวจสอบการเปลี่ยนแปลงของแบบจำลองเมื่อมีการตัดตัวกรองออก m คือ จำนวนมิติของตัวกรองของชั้นคอนโวลูชัน i+1 $\hat{X}_i$ คือ ค่าผลลัพธ์จากชั้น i ซึ่งเป็นจุดข้อมูลในตำแหน่งเดียวกับ y_i ซึ่งมาจากแบบจำลองที่ผ่านการตัดแล้ว และ w คือ ค่าสัมประสิทธิ์เพื่อใช้ปรับขนาดของตัวแปรน้ำหนัก

สมการ (19) สามารถใช้พีชคณิตเชิงเส้นแก้สมการให้อยู่ในรูปของสมการ (20) ได้

$$\widehat{w} = \underset{w}{\arg min} \sum_{i=1}^{m} (y_i - \hat{X}_i w)^2 \tag{19}$$

$$\widehat{w} = (\hat{X}^T \hat{X})^{-1} \hat{X}^T y \tag{20}$$

บทที่ 4
วิธีการที่นำเสนอ

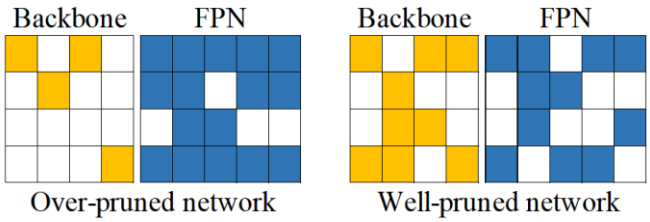
ในงานวิจัยชิ้นนี้จะทำการนำเสนอแนวทางการประยุกต์วิธีการตัดเพื่อใช้งานกับแบบจำลองการเรียนรู้เชิงลึกสำหรับการตรวจจับวัตถุ โดยมีจุดประสงค์ที่จะลดจำนวนตัวแปรของแบบจำลองการเรียนรู้เชิงลึกลง เพื่อลดขนาด และลดต้นทุนคำนวณของแบบจำลอง ซึ่งทำให้แบบจำลองสามารถนำไปใช้งานบนอุปกรณ์ขนาดเล็กอย่างเช่น กล้องวงจรปิด หรือคอมพิวเตอร์ขนาดเล็ก ได้อย่างมีประสิทธิภาพมากขึ้น

วิธีการที่นำเสนอในงานวิจัยชิ้นนี้ จะเน้นไปที่การบีบอัดแบบจำลองการเรียนรู้เชิงลึกด้วยกลยุทธ์การตัด ชุดข้อมูลที่ใช้ในงานวิจัยชิ้นนี้จะสามารถแบ่งได้เป็น 2 ประเภท คือ 1) ชุดข้อมูลตรวจจับยานพาหนะ และ 2) ชุดข้อมูลตรวจจับวัตถุทั่วไป โดยจะเน้นไปที่ชุดข้อมูลตรวจจับยานพาหนะเป็นหลัก แบบจำลองที่มุ่งเน้นในงานวิจัยชิ้นนี้ คือ แบบจำลอง YOLOv3 (สามารถอ่านรายละเอียดของแบบจำลองเพิ่มเติมได้ในบทที่ 3) ซึ่งจัดเป็นแบบจำลองการตรวจจับวัตถุแบบหนึ่งขั้นตอน (one-stage) ชนิดหนึ่ง ส่งผลให้ตัวแบบจำลองสามารถทำงานได้เร็ว และมีความเร็วต่อต้นทุนคำนวณที่สูง ซึ่งมีความเหมาะสมที่จะนำมาใช้บนอุปกรณ์ขนาดเล็กที่มีทรัพยากรและความสามารถในการประมวลผลจำกัด

จากการตรวจสอบค่าน้ำหนักของแบบจำลอง YOLOv3 ในเบื้องต้นพบว่า การแจกแจงของค่าของตัวแปรน้ำหนักมีความแตกต่างกันในแต่ละส่วนของแบบจำลอง ทำให้การตัดด้วยเกณฑ์การจัดอันดับ (Ranking Criteria) แบบธรรมดา มีโอกาสทำให้มีการตัดส่วนใดส่วนหนึ่งของแบบจำลองมากเกินไป (Over-pruning) ส่งผลให้ความแม่นยำของตัวแบบจำลองลดลงอย่างมีนัยสำคัญ เพื่อแก้ปัญหานี้ จึงเสนอแนวคิด 3 อย่างเพื่อแก้ปัญหาดังกล่าว 1) การตัดแบบแยกส่วน 2) การจำกัดการตัด และ 3) เกณฑ์การหยุด และในท้ายสุดจะกล่าวถึงการนำทั้ง 3 แนวคิดนี้มาประยุกต์ใช้งานร่วมกันเป็นอัลกอริทึมกลไกการตัดแบบทนทาน (Robust Pruning Mechanism หรือ RPM)

4.1 การตัดแบบแยกส่วน (Separated Pruning)

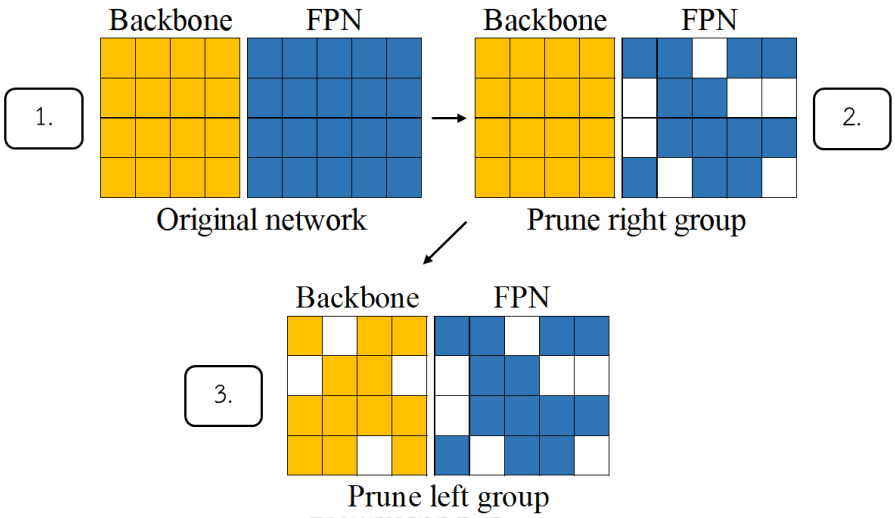
แบบจำลองการเรียนรู้เชิงลึกในปัจจุบันมีความซับซ้อนมากขึ้น และมักจะประกอบไปด้วยโมดูลหลาย ๆ ส่วนที่มีหน้าที่และจุดประสงค์ที่แตกต่างกัน ส่งผลให้ในแต่ละโมดูลมีคุณลักษณะของตัวแปรน้ำหนักที่ไม่เหมือนกัน ส่งผลให้เมื่อทำการตัดแบบจำลองทุกชั้นพร้อม ๆ กันด้วยวิธีการตัดแบบธรรมดา จะส่งผลให้ส่วนใดส่วนหนึ่งของแบบจำลองถูกตัดมากเกินไป ซึ่งแสดงให้เห็นในรูปด้านซ้ายของรูปที่ 9 ซึ่งส่งผลเสียต่อแบบจำลองเป็นอย่างมาก โดยจะส่งผลให้ประสิทธิภาพของแบบจำลองด้านความแม่นยำลดลงเป็นอย่างมาก จากการที่แบบจำลองมีตัวแปรน้ำหนักไม่เพียงพอสำหรับทำงาน ในทางกลับกัน หากว่าการตัดแบบจำลองสามารถกระจายไปแต่ละส่วนได้อย่างเหมาะสมดังที่แสดงในรูปด้านขวาของรูปที่ 9 จะส่งผลให้สามารถตัดแบบจำลองให้เล็กลงได้มากขึ้น



รูปที่ 9 รูปด้านซ้ายแสดงปัญหาการตัดส่วนใดส่วนหนึ่งของแบบจำลองมากเกินไปในส่วนของโครงข่ายกระดูกสันหลัง และรูปด้านขวาแสดงรูปแบบการตัดแบบจำลองที่ดี (Backbone คือ ส่วนของโครงข่ายกระดูกสันหลัง และ FPN คือ ส่วนของโครงข่ายพีระมิดฟีเจอร์) ช่องที่มีสี คือ ส่วนที่ยังไม่ได้ตัดตัวแปรน้ำหนักออก ช่องที่เป็นสีขาว คือ ส่วนที่ถูกตัดตัวแปรน้ำหนักออกแล้ว

เพื่อแก้ปัญหาที่ส่วนใดส่วนหนึ่งของแบบจำลองถูกตัดมากเกินไป งานวิจัยชิ้นนี้จึงได้นำเสนอแนวคิดการตัดแบบแยกส่วน (Separated Pruning) เพื่อใช้ในการตัดแบบจำลองแต่ละส่วนแยกกัน ในงานวิจัยชิ้นนี้ได้เลือกใช้งานแบบจำลอง YOLOv3 เป็นแบบจำลองหลัก ซึ่งประกอบด้วยโมดูลหลักสองส่วนคือ 1) โครงข่ายกระดูกสันหลัง (Backbone Network) และ 2) โครงข่ายพีระมิดฟีเจอร์ (Feature Pyramid Network หรือ FPN) ซึ่งจากการตรวจสอบในเบื้องต้นแล้วพบว่า ค่าของตัวแปรน้ำหนักของแบบจำลองทั้งสองส่วนมีความแตกต่างกัน แนวคิดและขั้นตอนการทำงานของการตัดแบบแยกส่วน ซึ่งถูกแสดงในรูปที่ 10 จะประกอบไปด้วย 3 ขั้นตอนหลัก ๆ ดังนี้

1) เตรียมที่จะเริ่มตัดแบบจำลองในแต่ละกลุ่มของชั้น (Layer) ของแบบจำลอง ซึ่งในกรณีนี้คือแบบจำลอง YOLOv3 ซึ่งประกอบไปด้วยสองส่วนคือ โครงข่ายกระดูกสันหลัง (Backbone Network) และโครงข่ายพีระมิดฟีเจอร์ (Feature Pyramid Network หรือ FPN) ซึ่งในการทดลองนี้จะเริ่มจากการตัดที่โครงข่ายพีระมิดฟีเจอร์ก่อนเป็นอันดับแรก และตามด้วยโครงข่ายกระดูกสันหลัง เนื่องจากมองว่า หน้าที่ของส่วนของโครงข่ายพีระมิดมีความสำคัญน้อยกว่าโครงข่ายกระดูกสันหลัง ซึ่งมีหน้าที่สกัดฟีเจอร์ของรูปภาพ
2) ทำการตัดเชิงวนซ้ำ (Iterative Pruning) ที่โครงข่ายพีระมิด ซึ่งจะทำการตัดไปเรื่อย ๆ จนกระทั่งความแม่นยำกับชุดข้อมูลตรวจสอบ (Validation Dataset) ของแบบจำลองเริ่มลดลง หลังจากนั้นจึงทำการหยุดการตัดแบบจำลองในส่วนของโครงข่ายพีระมิด และเลือกสถานะของแบบจำลองที่ความแม่นยำกับชุดข้อมูลตรวจสอบยังไม่ได้ลดลงเป็นแบบจำลองตั้งต้นสำหรับขั้นตอนถัดไป (วิธีการตัดสินว่าความแม่นยำลดลง และการเลือกแบบจำลองจะใช้งานโมดูลเกณฑ์การหยุด ซึ่งจะกล่าวในหัวข้อที่ 4.3)
3) ทำการตัดเชิงวนซ้ำที่โครงข่ายกระดูกสันหลัง ซึ่งจะทำการตัดไปเรื่อย ๆ จนกระทั่งความแม่นยำกับชุดข้อมูลตรวจสอบของแบบจำลองเริ่มลดลง หลังจากนั้นจึงทำการหยุดการตัดแบบจำลองในส่วนของโครงข่ายกระดูกสันหลัง และเลือกสถานะของแบบจำลองที่ความแม่นยำกับชุดข้อมูลตรวจสอบยังไม่ได้ลดลงเป็นแบบจำลองผลลัพธ์ของการตัดแบบจำลอง (วิธีการตัดสินว่าความแม่นยำลดลง และการเลือกแบบจำลองจะใช้งานโมดูลเกณฑ์การหยุด ซึ่งจะกล่าวในหัวข้อที่ 4.3)



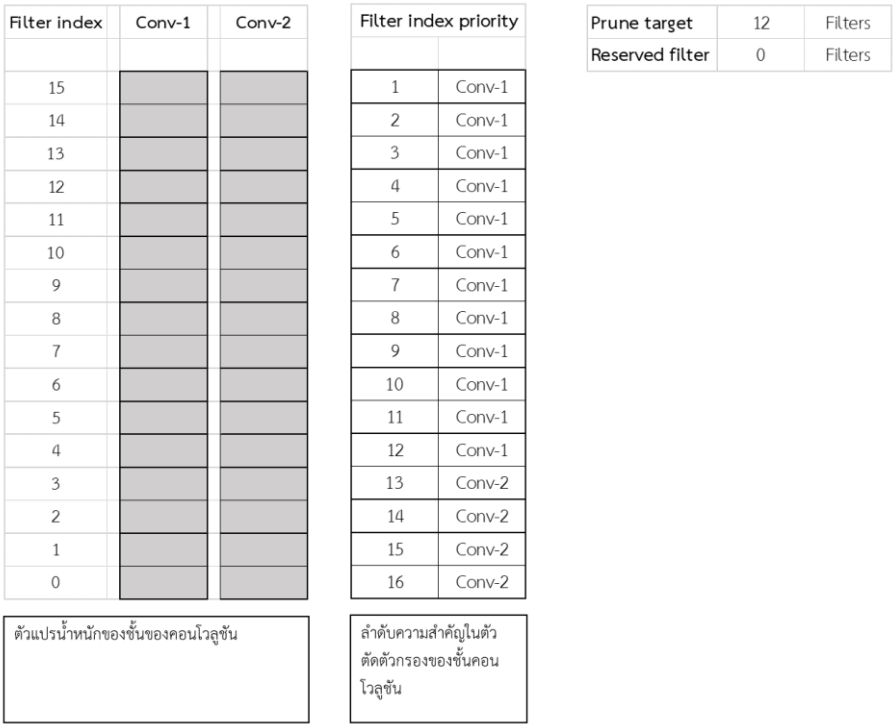
รูปที่ 10 ภาพรวมแนวคิดการตัดแบบแยกส่วน ช่องที่มีสี คือ ส่วนที่ยังไม่ได้ตัดตัวแปรน้ำหนักออก ช่องที่เป็นสีขาว คือ ส่วนที่ถูกตัดตัวแปรน้ำหนักออกแล้ว

4.2 การจำกัดการตัด (Minimum Filter Constraint)

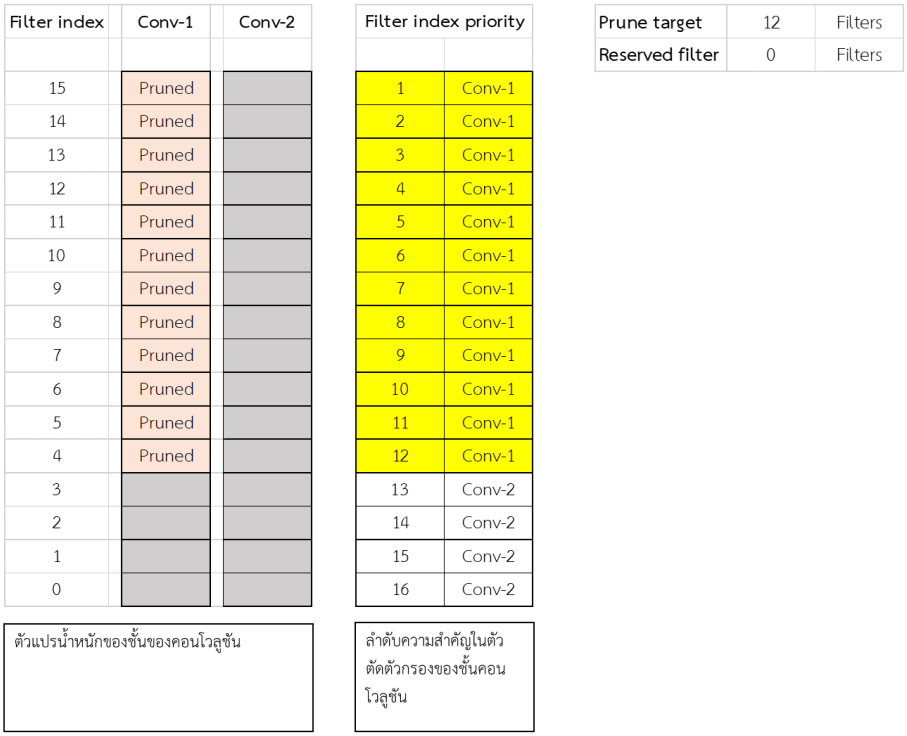
ในแต่ละชั้นของแบบจำลองการเรียนรู้เชิงลึก ย่อมมีขีดจำกัดที่จะตัดออกได้ เนื่องจากว่าในการที่แบบจำลองจะทำงานได้ ก็จำเป็นต้องมีตัวแปรค่าน้ำหนักไว้จำนวนหนึ่ง ซึ่งถ้าหากมีการทำการตัดแบบจำลองในบางชั้นมากเกินไป ก็จะส่งผลให้ความแม่นยำของแบบจำลองลดลงอย่างรวดเร็วมาก เนื่องจากว่าตัวแบบจำลองมีตัวแปรน้ำหนักที่ไม่เพียงพอสำหรับการทำงาน

ดังนั้นเพื่อป้องกันไม่ให้แต่ละชั้นของแบบจำลองถูกตัดมากเกินไป จนถึงจุดที่แบบจำลองไม่สามารถทำงานได้ตามปกติ งานวิจัยชิ้นนี้จึงเสนอการจำกัดการตัด โดยจุดมุ่งหมายของโมดูลตัวนี้คือ การป้องกันไม่ให้แต่ละชั้นของแบบจำลองถูกตัดจนเหลือตัวกรองน้อยกว่าจำนวนขั้นต่ำที่กำหนดไว้ แต่ทว่าโมดูลตัวนี้ควรใช้ในช่วงท้ายของการตัด เนื่องจากว่าไม่ใช่ทุกชั้นของแบบจำลองการเรียนรู้เชิงลึกที่จำเป็นที่จะต้องใช้งานการจำกัดการตัด ดังนั้นถ้าหากว่ามีการใช้การจำกัดการตัดนี้เร็วเกินไป ตั้งแต่ช่วงต้นของการตัด จะส่งผลให้ไม่ได้ตัดตัวกรองบางตัวที่สามารถถูกตัดออกได้ ส่งผลให้ผลลัพธ์ของการตัดแบบจำลองการเรียนรู้เชิงลึก สามารถตัดตัวแปรน้ำหนักออกได้น้อยลง

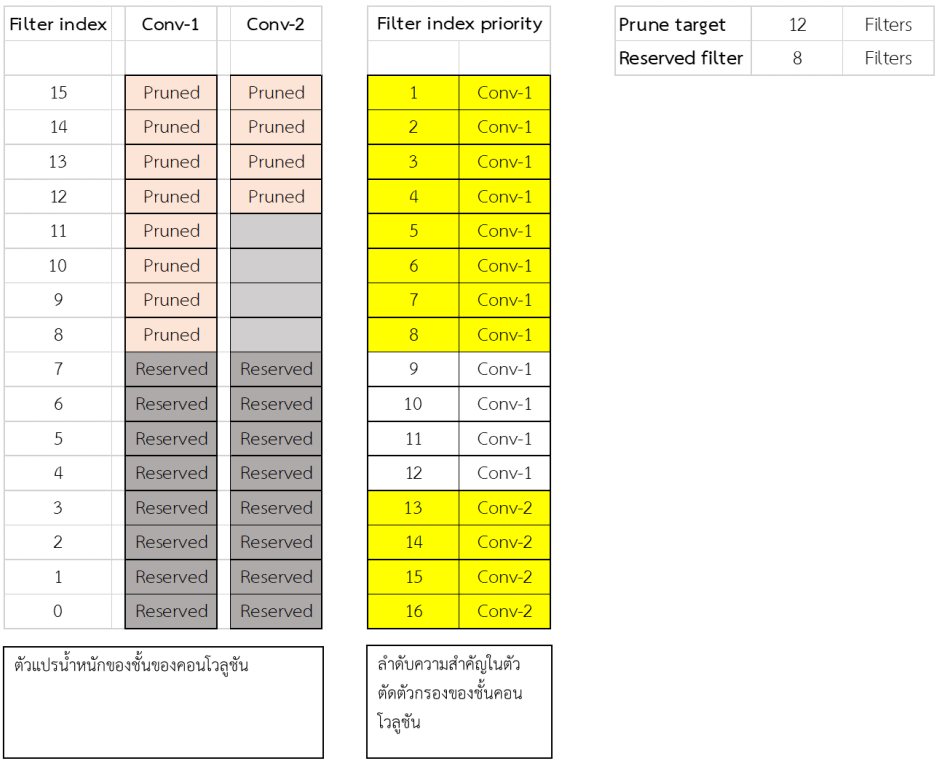
สำหรับตัวอย่างการทำงานของโมดูลการจำกัดการตัดจะถูกแสดงในรูปที่ 11 ถึง รูปที่ 13 โดยการทำงานจะเริ่มจากสถานะในรูปที่ 11 ซึ่งเป็นสถานะของแบบจำลองที่ประกอบไปด้วย 2 ชั้น คือ Conv-1 และ Conv-2 แต่ละชั้นมีจำนวนตัวกรอง (Filter) ทั้งหมด 16 ตัวกรอง มีลำดับความสำคัญตามตารางลำดับความสำคัญตัวกรอง (Filter Index Priority) และมีเป้าหมายที่จะตัดตัวกรองของชั้นคอนโวลูชันทั้งหมด 12 ตัวกรอง ถ้าหากว่าทำการตัดแบบจำลองโดยไม่ใช้โมดูลการจำกัดการตัด จะได้ผลลัพธ์ดังรูปที่ 12 โดยชั้น Conv-1 จะถูกตัดไปทั้งหมด 12 ตัวกรอง ส่งผลให้ชั้น Conv-1 เหลือจำนวนตัวกรองที่น้อยเกินไป ส่งผลให้ความแม่นยำของแบบจำลองลดลงอย่างมาก ในทางกลับกันถ้าหากมีการใช้โมดูลการจำกัดการตัด จะได้ผลลัพธ์ดังรูปที่ 13 โดยชั้น Conv-1 จะถูกตัดไปทั้งหมด 8 ตัวกรอง และชั้น Conv-2 จะถูกตัดไปทั้งหมด 4 ตัวกรอง ซึ่งจะเห็นได้ว่า เมื่อมีการใช้โมดูลการจำกัดการตัด จะส่งผลให้แต่ละชั้นของแบบจำลองมีตัวแปรน้ำหนักเหลืออยู่ที่สมดุลกว่า



รูปที่ 11 ภาพประกอบตัวอย่างการทำงานของโมดูลการจำกัดการตัด สถานะเริ่มต้นก่อนเริ่มทำการตัดแบบจำลอง ก่อนทำการตัดแบบจำลอง



รูปที่ 12 ภาพประกอบตัวอย่างการทำงานของโมดูลการจำกัดการตัด กรณีตัดแบบจำลองเมื่อไม่มีการใช้โมดูลการจำกัดการตัด



รูปที่ 13 ภาพประกอบตัวอย่างการทำงานของโมดูลการจำกัดการตัด กรณีตัดแบบจำลองเมื่อมีการใช้โมดูลการจำกัดการตัด

4.3 เกณฑ์การหยุด (Stopping Criteria)

ในขั้นตอนการทำงานของการตัดแบบแยกส่วน เพื่อให้การตัดแบบแยกส่วนสามารถทำงานได้อย่างต่อเนื่องและสามารถเลือกได้ว่าควรจะหยุดการตัดและเลือกแบบจำลอง ณ จุดที่เหมาะสมได้โดยอัตโนมัติโดยผู้ใช้งานไม่จำเป็นต้องตัดสินใจด้วยตัวเอง

งานวิจัยชิ้นนี้ออกแบบเกณฑ์การหยุดด้วยการนำหลักการความอดทน (Patience) เข้ามาใช้งาน โดยโมดูลตัวนี้จะทำการคอยตรวจดูค่าความแม่นยำของชุดข้อมูลตรวจสอบ (Validation Data) ในแต่ละช่วงของการตัดแบบจำลอง โดยโมดูลจะคอยตรวจดูว่าเมื่อไหร่ที่ค่าความแม่นยำของชุดข้อมูลตรวจสอบเริ่มลดลง และตัดสินใจว่าเมื่อไหร่ที่ควรจะหยุดการตัด และเลือกสถานะที่ดีที่สุดของแบบจำลองออกมาโดยการเลือกสถานะของแบบจำลองที่ถูกตัดได้มากที่สุดโดยความแม่นยำของชุดข้อมูลตรวจสอบยังไม่ลดลง

4.3.1 คำอธิบายฟังก์ชันที่เกี่ยวข้องกับเกณฑ์การหยุด

ในหัวข้อนี้จะอธิบายรายละเอียดต่าง ๆ ของฟังก์ชันทั้งหมด ที่มีการใช้งานกในเกณฑ์การหยุด ซึ่งอยู่ในรหัสเทียมในรูปที่ 14 และรูปที่ 15

1) ชื่อฟังก์ชัน : shouldStop
   พารามิเตอร์ (1) : modelHistory – อาเรย์ของแบบจำลอง ทำหน้าที่เปรียบเสมือนประวัติการตัดของแบบจำลอง (เลขของดัชนีสะท้อนถึงจำนวนครั้งที่ผ่านการตัดแบบจำลอง)
   พารามิเตอร์ (2) : threshold – ขีดแบ่งเพื่อระบุว่าความแม่นยำของแบบจำลองเริ่มลดลง
   ผลลัพธ์ : boolean - ตัวแปรเพื่อบ่งชี้ว่า ความแม่นยำของแบบจำลองอยู่ในสถานะลดลงแล้วหรือไม่
   คำอธิบาย : ตรวจสอบว่าความแม่นยำของแบบจำลองลดลงต่ำกว่าขีดแบ่งที่กำหนดไว้แล้วหรือไม่
2) ชื่อฟังก์ชัน : getLength
   พารามิเตอร์ : modelHistory – อาเรย์ของแบบจำลอง ทำหน้าที่เปรียบเสมือนประวัติการตัดของแบบจำลอง (เลขของดัชนีสะท้อนถึงจำนวนครั้งที่ผ่านการตัดแบบจำลอง)
   ผลลัพธ์ : length - จำนวนสมาชิกในอาเรย์
   คำอธิบาย : อ่านจำนวนสมาชิกในอาเรย์
3) ชื่อฟังก์ชัน : getModelByIndex
   พารามิเตอร์ (1) : modelHistory – อาเรย์ของแบบจำลอง ทำหน้าที่เปรียบเสมือนประวัติการตัดของแบบจำลอง (เลขของดัชนีสะท้อนถึงจำนวนครั้งที่ผ่านการตัดแบบจำลอง)
   พารามิเตอร์ (2) : index - ดัชนีของแบบจำลอง
   ผลลัพธ์ : model - แบบจำลองตามดัชนีที่เลือกไว้
   คำอธิบาย : เลือกแบบจำลองตามดัชนีที่เลือกไว้จากอาเรย์ของแบบจำลอง
4) ชื่อฟังก์ชัน : getValAcc
   พารามิเตอร์ : model - ตัวแปรแบบจำลอง
   ผลลัพธ์ : validationAccuracy - ความแม่นยำกับชุดข้อมูลตรวจสอบ
   คำอธิบาย : อ่านค่าความแม่นยำกับชุดข้อมูลตรวจสอบของแบบจำลอง
5) ชื่อฟังก์ชัน : getBestModel
   พารามิเตอร์ (1) : modelHistory – อาเรย์ของแบบจำลอง ทำหน้าที่เปรียบเสมือนประวัติการตัดของแบบจำลอง (เลขของดัชนีสะท้อนถึงจำนวนครั้งที่ผ่านการตัดแบบจำลอง)
   พารามิเตอร์ (2) : threshold – ขีดแบ่งเพื่อระบุว่าความแม่นยำของแบบจำลองเริ่มลดลง
   ผลลัพธ์ : model - แบบจำลองที่เหมาะสมที่สุด
   คำอธิบาย : เลือกแบบจำลองที่เหมาะสมที่สุดจากประวัติการตัดแบบจำลอง

4.3.2 รายละเอียดการทำงานของเกณฑ์การหยุด

การทำงานของเกณฑ์การหยุดนั้นจะถูกแบ่งออกเป็นสองส่วนหลักๆ คือ 1) ฟังก์ชันตรวจสอบการลดลงของความแม่นยำ และ 2) ฟังก์ชันเลือกแบบจำลองที่เหมาะสมที่สุด ซึ่งถูกแสดงอยู่ในรูปของรหัสเทียม (Pseudocode) ดังรูปที่ 14 และรูปที่ 15 โดยในกระบวนการทำงานจะเริ่มจากการใช้ฟังก์ชัน shouldStop เพื่อตรวจสอบสถานะของความแม่นยำกับชุดข้อมูลตรวจสอบว่าลดลงแล้วหรือยัง ด้วยการเปรียบเทียบแบบจำลองเริ่มต้นกับสองแบบจำลองล่าสุด ซึ่งโมดูลตัวนี้จะสั่งให้หยุดทำการตัดก็ต่อเมื่อความแม่นยำของชุดข้อมูลตรวจสอบหลังทำการตัดลดลงมากกว่าเงื่อนไขที่กำหนดไว้ หลังจากนั้นจะใช้งานฟังก์ชัน getBestModel เพื่อเลือกแบบจำลองที่เหมาะสมที่สุดโดยการเลือกสถานะของแบบจำลองก่อนที่ความแม่นยำจะลดลงเป็นแบบจำลองผลลัพธ์

Algorithm Accuracy Drop Observer

```
1:  procedure SHOULDSTOP(modelHistory, threshold)
2:      length ← getLength(modelHistory)
3:      model0 ← getModelByIndex(modelHistory, 0)
4:      model1 ← getModelByIndex(modelHistory, length − 2)
5:      model2 ← getModelByIndex(modelHistory, length − 1)
6:      valAcc0 ← getValAcc(model0)
7:      valAcc1 ← getValAcc(model1)
8:      valAcc2 ← getValAcc(model2)
9:      condition1 ← (valAcc0 - valAcc1) > threshold
10:     condition2 ← (valAcc0 - valAcc2) > threshold
11:     return condition1 and condition2
```

รูปที่ 14 รหัสเทียมของฟังก์ชันตรวจสอบการลดลงของความแม่นยำ

Algorithm Model Selector

```
1:  procedure GETBESTMODEL(modelHistory, threshold)
2:      length ← getLength(modelHistory)
3:      for i ← 0 to length − 3 do
4:          model0 ← getModelByIndex(modelHistory, 0)
5:          model1 ← getModelByIndex(modelHistory, i + 1)
6:          model2 ← getModelByIndex(modelHistory, i + 2)
7:          valAcc0 ← getValAcc(model0)
8:          valAcc1 ← getValAcc(model1)
9:          valAcc2 ← getValAcc(model2)
10:         condition1 ← (valAcc0 - valAcc1) > threshold
11:         condition2 ← (valAcc0 - valAcc2) > threshold
12:         if condition1 and condition2 then
13:             return getModelByIndex(modelHistory, i)
```

รูปที่ 15 รหัสเทียมฟังก์ชันเลือกแบบจำลองที่เหมาะสมที่สุด

4.4 กลไกการตัดแบบทนทาน (Robust Pruning Mechanism หรือ RPM)

ในหัวข้อนี้จะกล่าวถึง การนำสิ่งที่นำเสนอในขั้นต้น ซึ่งประกอบไปด้วย 1) การตัดแบบแยกส่วน 2) การจำกัดการตัด และ 3) เกณฑ์การหยุด มารวมเข้าด้วยกันเป็นอัลกอริทึมกลไกการตัดแบบทนทาน ซึ่งมีจุดประสงค์เพื่อใช้ในการตัดแบบจำลองให้มีขนาดเล็กที่สุด ในขณะที่คงความแม่นยำไว้ให้ใกล้เคียงกับระดับเดิม

ในการอธิบายรายละเอียดและวิธีการทำงานของกลไกการตัดแบบทนทาน จะแบ่งออกเป็นสองส่วนคือ 1) รายละเอียด และคำอธิบายฟังก์ชันที่เกี่ยวข้องในกลไกการตัดแบบทนทาน และ 2) รหัสเทียม (Pseudo Code) และรายละเอียดการทำงานของกลไกการตัดแบบทนทาน

4.4.1 คำอธิบายฟังก์ชันที่เกี่ยวข้องกับกลไกการตัดแบบทนทาน

ในหัวข้อนี้ จะอธิบายรายละเอียดต่าง ๆ ของฟังก์ชันทั้งหมด ที่มีการใช้งานกับกลไกการตัดแบบทนทาน ซึ่งอยู่ในรหัสเทียมในรูปที่ 16

1) ชื่อฟังก์ชัน : robustPruningMechanism
   พารามิเตอร์ (1) : model – ตัวแปรแบบจำลองตั้งต้น
   พารามิเตอร์ (2) : threshold – ขีดแบ่งเพื่อระบุว่าความแม่นยำของแบบจำลองเริ่มลดลง
   ผลลัพธ์ : model - ตัวแปรแบบจำลองผลลัพธ์ซึ่งผ่านการตัดด้วยกลไกการตัดแบบทนทาน
   คำอธิบาย : ตัดแบบจำลองด้วยกลไกการตัดแบบทนทาน
2) ชื่อฟังก์ชัน : getLayerGroup
   พารามิเตอร์ : model - ตัวแปรแบบจำลอง
   ผลลัพธ์ : groups - อาเรย์ของกลุ่มของชั้น (Layer)
   คำอธิบาย : นำแต่ละชั้นของแบบจำลองมาแบ่งเป็นกลุ่ม
3) ชื่อฟังก์ชัน : modelHistory.append
   พารามิเตอร์ : model - ตัวแปรแบบจำลอง
   ผลลัพธ์ : ไม่มี
   คำอธิบาย : บันทึกสถานะของแบบจำลอง (ตัวแปรน้ำหนัก และความแม่นยำ)
4) ชื่อฟังก์ชัน : shouldStop
   พารามิเตอร์ (1) : modelHistory – อาเรย์ของแบบจำลอง ทำหน้าที่เปรียบเสมือนประวัติการตัดของแบบจำลอง (เลขของดัชนีสะท้อนถึงจำนวนครั้งที่ผ่านการตัดแบบจำลอง)
   พารามิเตอร์ (2) : threshold – ขีดแบ่งเพื่อระบุว่าความแม่นยำของแบบจำลองเริ่มลดลง
   ผลลัพธ์ : boolean – ตัวแปรเพื่อบ่งชี้ว่า ความแม่นยำของแบบจำลองอยู่ในสถานะลดลงแล้วหรือไม่
   คำอธิบาย : ตรวจสอบว่าความแม่นยำของแบบจำลองลดลงต่ำกว่าขีดแบ่งที่กำหนดไว้แล้วหรือไม่

5) ชื่อฟังก์ชัน : getUnnecessaryFilters
   พารามิเตอร์ : group - กลุ่มของชั้น (Layer)
   ผลลัพธ์ : filters - อาเรย์ของตัวกรองที่จะถูกตัดออก
   คำอธิบาย : ใช้เกณฑ์การจัดอันดับ (Ranking Criteria) จัดอันดับความสำคัญ และเลือกตัวกรองที่มีความสำคัญต่ำ
6) ชื่อฟังก์ชัน : prune
   พารามิเตอร์ (1) : model - ตัวแปรแบบจำลอง
   พารามิเตอร์ (2) : filters - อาเรย์ของตัวกรองที่จะถูกตัดออก
   ผลลัพธ์ : model – ตัวแปรแบบจำลองที่ผ่านการตัดแล้ว
   คำอธิบาย : ตัดตัวกรองที่ไม่ต้องการออกจากแบบจำลอง
7) ชื่อฟังก์ชัน : finetune
   พารามิเตอร์ : model - ตัวแปรแบบจำลอง
   ผลลัพธ์ : model - ตัวแปรแบบจำลองที่ผ่านการฝึกแล้ว
   คำอธิบาย : ฝึกแบบจำลอง
8) ชื่อฟังก์ชัน : getBestModel
   พารามิเตอร์ (1) : modelHistory – อาเรย์ของแบบจำลอง ทำหน้าที่เปรียบเสมือนประวัติการตัดของแบบจำลอง (เลขของดัชนีสะท้อนถึงจำนวนครั้งที่ผ่านการตัดแบบจำลอง)
   พารามิเตอร์ (2) : threshold – ขีดแบ่งเพื่อระบุว่าความแม่นยำของแบบจำลองเริ่มลดลง
   ผลลัพธ์ : model - แบบจำลองที่เหมาะสมที่สุด
   คำอธิบาย : เลือกแบบจำลองที่เหมาะสมที่สุดจากประวัติการตัดแบบจำลอง
9) ชื่อฟังก์ชัน : modelHistory.clearHistory
   พารามิเตอร์ : ไม่มี
   ผลลัพธ์ : ไม่มี
   คำอธิบาย : ล้างประวัติการตัดแบบจำลอง
10) ชื่อฟังก์ชัน : getUnnecessaryFiltersWithLimit
   พารามิเตอร์ : model - ตัวแปรแบบจำลอง
   ผลลัพธ์ : filters - อาเรย์ของตัวกรองที่จะถูกตัดออก
   คำอธิบาย : ใช้เกณฑ์การจัดอันดับ (Ranking Criteria) จัดอันดับความสำคัญ และเลือกตัวกรองที่มีความสำคัญต่ำพร้อมกับใช้โมดูลการจำกัดการตัดเพื่อสงวนตัวกรองไว้ตามจำนวนที่กำหนดไว้

4.4.2 รายละเอียดการทำงานของกลไกการตัดแบบทนทาน

ในหัวข้อนี้จะกล่าวถึงขั้นตอน และรายละเอียดการทำงานของกลไกการตัดแบบทนทาน สำหรับแบบจำลอง YOLOv3 ซึ่งจะประกอบไปด้วย 4 ขั้นตอนหลักดังนี้

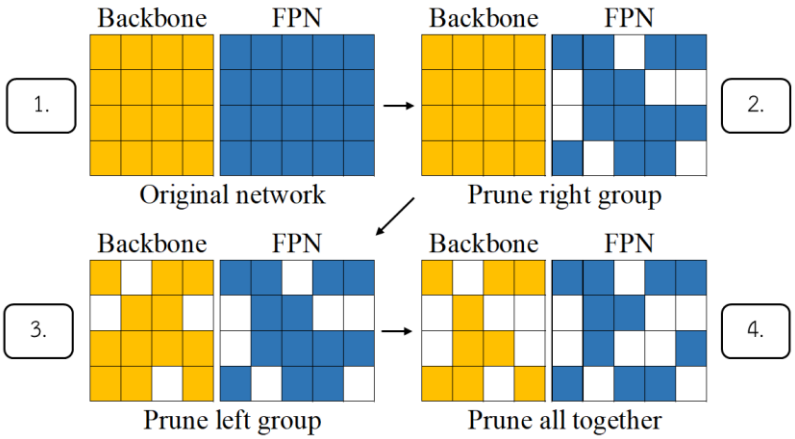
1) บรรทัดที่ 2–3 ของรหัสเทียมในรูปที่ 16 และขั้นตอนที่ 1 ในรูปที่ 17
   - จัดกลุ่มชั้น (Layer) ของแบบจำลอง ซึ่งในกรณีนี้คือแบบจำลอง YOLOv3 ซึ่งประกอบไปด้วยสองส่วน คือ โครงข่ายกระดูกสันหลัง (Backbone Network) และโครงข่ายพีระมิดฟีเจอร์ (Feature Pyramid Network หรือ FPN) ซึ่งในการทดลองนี้จะเริ่มจากการตัดที่โครงข่ายพีระมิดฟีเจอร์ก่อนเป็นอันดับแรก และตามด้วยโครงข่ายกระดูกสันหลัง เนื่องจากมองว่า หน้าที่ของส่วนของโครงข่ายพีระมิดมีความสำคัญน้อยกว่าโครงข่ายกระดูกสันหลังที่ทำหน้าที่สกัดฟีเจอร์ของรูปภาพ ซึ่งถือว่าเป็นส่วนที่มีความสำคัญสูงมากของตัวแบบจำลอง
   - บันทึกสถานะของตัวแปรในแบบจำลองไว้ในประวัติการตัด
2) บรรทัดที่ 4–12 ของรหัสเทียมในรูปที่ 16 และขั้นตอนที่ 2 ในรูปที่ 17
   - จากลำดับความสำคัญของตัวกรอง และคัดเลือกกลุ่มของตัวกรองที่มีความสำคัญต่ำที่สุดจากโครงข่ายพีระมิดฟีเจอร์
   - ตัดตัวกรองที่มีความสำคัญต่ำที่สุดซึ่งถูกคัดมาในขั้นตอนก่อนหน้านี้ออกจากแบบจำลอง
   - ทำการฝึกแบบจำลองที่ผ่านการตัดมาแล้วซ้ำเพื่อปรับตัวแปรน้ำหนักให้เหมาะสม
   - บันทึกสถานะของตัวแปรในแบบจำลองไว้ในประวัติการตัด
   - วนขั้นตอนก่อนหน้านี้ทั้ง 4 ขั้นตอนซ้ำไปเรื่อย ๆ จนกระทั่งความแม่นยำกับชุดข้อมูลตรวจสอบลดลงจากการตัดสินด้วยเกณฑ์การหยุด
   - เลือกแบบจำลอง ณ จุดก่อนที่ความแม่นยำกับชุดข้อมูลตรวจสอบลดลงด้วยเกณฑ์การหยุด
   - ล้างประวัติการตัด
   - ตั้งแบบจำลองผลลัพธ์เป็นแบบจำลองตั้งต้นเพื่อใช้ในขั้นตอนถัดไป
3) บรรทัดที่ 4–12 ของรหัสเทียมในรูปที่ 16 และขั้นตอนที่ 3 ในรูปที่ 17
   - จากลำดับความสำคัญของตัวกรอง และคัดเลือกกลุ่มของตัวกรองที่มีความสำคัญต่ำที่สุดจากโครงข่ายกระดูกสันหลัง
   - ตัดตัวกรองที่มีความสำคัญต่ำที่สุดซึ่งถูกคัดมาในขั้นตอนก่อนหน้านี้ออกจากแบบจำลอง
   - ทำการฝึกแบบจำลองที่ผ่านการตัดมาแล้วซ้ำเพื่อปรับตัวแปรน้ำหนักให้เหมาะสม
   - บันทึกสถานะของตัวแปรในแบบจำลองไว้ในประวัติการตัด
   - วนขั้นตอนก่อนหน้านี้ทั้ง 4 ขั้นตอนซ้ำไปเรื่อย ๆ จนกระทั่งความแม่นยำกับชุดข้อมูลตรวจสอบลดลงจากการตัดสินด้วยเกณฑ์การหยุด
   - เลือกแบบจำลอง ณ จุดก่อนที่ความแม่นยำกับชุดข้อมูลตรวจสอบลดลงด้วยเกณฑ์การหยุด
   - ล้างประวัติการตัด
   - ตั้งแบบจำลองผลลัพธ์เป็นแบบจำลองตั้งต้นเพื่อใช้ในขั้นตอนถัดไป

4) บรรทัดที่ 13–19 ของรหัสเทียมในรูปที่ 16 และขั้นตอนที่ 4 ในรูปที่ 17
   - จากลำดับความสำคัญของตัวกรอง และคัดเลือกกลุ่มของตัวกรองที่มีความสำคัญต่ำที่สุดจากทุกชั้น (Layer) ในแบบจำลอง พร้อมกับใช้งานโมดูลการจำกัดการตัด
   - ตัดตัวกรองที่มีความสำคัญต่ำที่สุดซึ่งถูกคัดมาในขั้นตอนก่อนหน้านี้ออกจากแบบจำลอง
   - ทำการฝึกแบบจำลองที่ผ่านการตัดมาแล้วซ้ำเพื่อปรับตัวแปรน้ำหนักให้เหมาะสม
   - บันทึกสถานะของตัวแปรในแบบจำลองไว้ในประวัติการตัด
   - วนขั้นตอนก่อนหน้านี้ทั้ง 4 ขั้นตอนซ้ำไปเรื่อย ๆ จนกระทั่งความแม่นยำกับชุดข้อมูลตรวจสอบลดลงจากการตัดสินด้วยเกณฑ์การหยุด
   - เลือกแบบจำลอง ณ จุดก่อนที่ความแม่นยำกับชุดข้อมูลตรวจสอบลดลงด้วยเกณฑ์การหยุด
   - คืนแบบจำลองสุดท้ายเป็นแบบจำลองผลลัพธ์ของอัลกอริทึมกลไกการตัดแบบทนทาน

Algorithm Robust Pruning Mechanism

```
1:  procedure RobustPruningMechanism(model, threshold)
2:      groups ← getLayerGroup(model)
3:      modelHistory.append(model)
4:      for each group ∈ groups do
5:          while not shouldStop(modelHistory, threshold) do
6:              filters ← getUnnecessaryFilters(group)
7:              model ← prune(model, filters)
8:              model ← finetune(model)
9:              modelHistory.append(model)
10:         model ← getBestModel(modelHistory, threshold)
11:         modelHistory.clearHistory()
12:         modelHistory.append(model)
13:     while not shouldStop(modelHistory, threshold) do
14:         filters ← getUnnecessaryFilterWithLimit(model)
15:         model ← prune(model, filters)
16:         model ← finetune(model)
17:         modelHistory.append(model)
18:     model ← getBestModel(modelHistory, threshold)
19:     return model
```

รูปที่ 16 รหัสเทียมกลไกการตัดแบบทนทาน



รูปที่ 17 ภาพรวมการทำงานของกลไลการตัดแบบทนทาน กล่องที่มีสี คือ ส่วนที่ยังไม่ได้ตัดตัวแปรน้ำหนักออก กล่องที่เป็นสีขาว คือ ส่วนที่ถูกตัดตัวแปรน้ำหนักออกแล้ว

บทที่ 5
การเตรียมการทดลอง

ในบทนี้จะกล่าวถึงเรื่องของการเตรียมการทดลอง เพื่อเตรียมที่จะทำการทดลองต่าง ๆ ซึ่งจะประกอบไปด้วย 4 หัวข้อย่อยดังต่อไปนี้

1) ฮาร์ดแวร์ และซอฟต์แวร์
2) ชุดข้อมูล
3) รายละเอียดการฝึกแบบจำลองพื้นฐาน
4) วิธีการพื้นฐาน

5.1 ฮาร์ดแวร์ และซอฟต์แวร์

ในหัวข้อนี้จะกล่าวถึงรายละเอียดของฮาร์ดแวร์ และซอฟต์แวร์ที่ใช้งานเพื่อทำการทดลองต่าง ๆ ในงานวิจัยชิ้นนี้

รายละเอียดของฮาร์ดแวร์ที่ใช้งาน เพื่อทำการทดลองต่าง ๆ ในงานวิจัยชิ้นนี้

ซีพียู	: Intel(R) Core(TM) i7-8700
จีพียู	: Nvidia GeForce RTX 2080 Ti
แรม	: 32GB DDR4 3000MHz
อุปกรณ์จัดเก็บข้อมูล	: Samsung SSD 970 EVO 500GB

2. ซอฟต์แวร์

รายละเอียดของซอฟต์แวร์ที่ใช้งาน เพื่อทำการทดลองต่าง ๆ ในงานวิจัยชิ้นนี้

ระบบปฏิบัติการ	: Windows 10 Build 1803
รุ่น Anaconda	: Anaconda 3 (64-bit)
รุ่น Python	: Python 3.5
ซอฟต์แวร์การเรียนรู้เชิงลึก	: Keras 2.2.4 (Tensorflow 1.11.0)

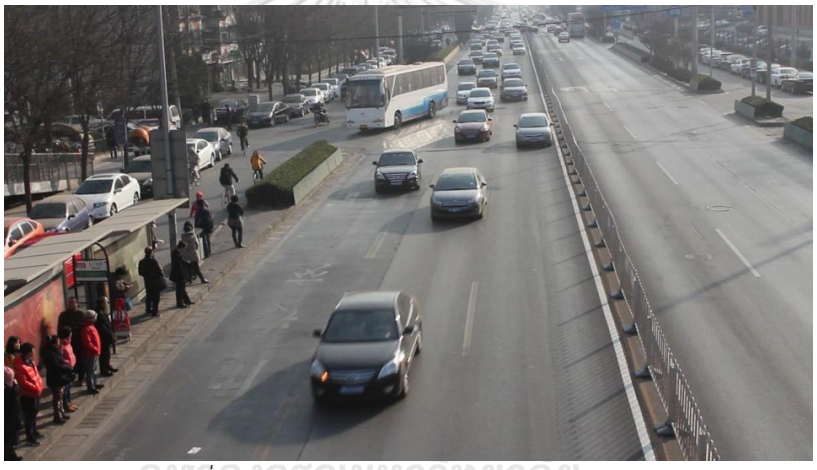
5.2 ชุดข้อมูล

ในงานวิจัยชิ้นนี้จะมีการใช้ชุดข้อมูล 2 ชุดข้อมูลเป็นหลัก ซึ่งประกอบไปด้วย 1) UA-DETRAC เป็นชุดข้อมูลตรวจจับยานพาหนะ และ 2) PASCAL VOC เป็นชุดข้อมูลตรวจจับวัตถุทั่วไป

5.2.1 ชุดข้อมูล UA-DETRAC

ข้อมูลชุดนี้ [22] เป็นชุดข้อมูลสำหรับการตรวจจับ และการติดตามยานพาหนะบนถนน ซึ่งในงานวิจัยชิ้นนี้จะยึดชุดข้อมูลชุดนี้เป็นชุดข้อมูลหลัก ตัวชุดข้อมูลประกอบไปด้วยวิดีโอความยาว 10 ชั่วโมง จากถนนสายต่าง ๆ มีจำนวนรูปทั้งหมด 82,085 รูป โดยแต่ละรูปภาพในชุดข้อมูลจะประกอบไปด้วย 1) ตำแหน่งของรถยนต์แต่ละคันบนรูปภาพ และ 2) ส่วนของพื้นที่ไม่ใช้งานของรูปภาพ

ในการทดลอง เนื่องจากว่ารูปภาพมีเป็นจำนวนมาก และกระบวนการตัดแบบจำลองด้วยการตัดเชิงวนซ้ำ (Iterative Pruning) มีความจำเป็นต้องทำการฝึกสอนโมเดลซ้ำเป็นจำนวนมาก จึงมีความจำเป็นต้องลดขนาดข้อมูลลงเพื่อให้ระยะเวลาที่ใช้ในการทดลองมีความเหมาะสม โดยชุดข้อมูลจะถูกแบ่งออกเป็น 3 ส่วนดังนี้ 1) ชุดข้อมูลฝึก 4,543 รูป 2) ชุดข้อมูลตรวจสอบ 3,184 รูป และ 3) ชุดข้อมูลทดสอบ 2,694 รูป



รูปที่ 18 ตัวอย่างข้อมูลจากชุดข้อมูล UA-DETRAC

5.2.2 ชุดข้อมูล PASCAL VOC

ข้อมูลชุดนี้ [23] เป็นชุดข้อมูลสำหรับการตรวจจับวัตถุทั่วไป ซึ่งในงานวิจัยชิ้นนี้จะใช้เป็นชุดข้อมูลรอง โดยข้อมูล PASCAL VOC จะสามารถแบ่งออกได้เป็นสองชุดคือ PASCAL VOC 2007 ประกอบด้วยรูปภาพจำนวน 5,011 รูปภาพ มีวัตถุจำนวน 12,608 วัตถุ และสำหรับ PASCAL VOC 2012 ประกอบด้วยรูปภาพจำนวน 11,540 รูปภาพ มีวัตถุจำนวน 27,450 วัตถุ มีจำนวนคลาสทั้งหมด 20 ชนิด ประกอบด้วย 1) เครื่องบิน 2) จักรยาน 3) นก 4) เรือ 5) ขวด 6) รถบัส 7) รถยนต์ 8) แมว 9) เก้าอี้ 10) วัว 11) โต๊ะกินข้าว 12) สุนัข 13) ม้า 14) จักรยานยนต์ 15) มนุษย์ 16) กระถางต้นไม้ 17) แกะ 18) โซฟา 19) รถไฟ และ 20) โทรทัศน์

ในการทดลอง เพื่อเพิ่มความหลากหลายของข้อมูล จึงมีการนำชุดข้อมูล PASCAL VOC 2007 (VOC07) และ PASCAL VOC 2012 (VOC12) มารวมกัน หลังจากนั้นชุดข้อมูลจะถูกแบ่งออกเป็น 3 ส่วนดังนี้ 1) ชุดข้อมูลฝึก 8,218 รูป ประกอบด้วยชุดข้อมูลฝึกจาก VOC07 2,501 รูป และ VOC12 5,717 รูป 2) ชุดข้อมูลตรวจสอบ 2,521 รูป จาก VOC07 2,521 รูป และ 3) ชุดข้อมูลทดสอบ 4,952 รูป จาก VOC07 4,952 รูป

รูปที่ 19 ตัวอย่างข้อมูลจากชุดข้อมูล PASCAL VOC

5.3 แบบจำลองพื้นฐานและรายละเอียดการฝึกแบบจำลองพื้นฐาน

ในหัวข้อนี้จะกล่าวถึงรายระเอียดในการฝึกแบบจำลองพื้นฐาน เนื่องจากในการทำการทดลองการตัดแบบจำลองจำเป็นต้องมีแบบจำลองพื้นฐานที่ผ่านการฝึกมาแล้วเป็นตัวตั้งต้นก่อน ถึงจำเป็นต้องมีการฝึกแบบจำลองไว้เป็นจำนวนสองชุดข้อมูลซึ่งประกอบไปด้วย 1) UA-DETRAC และ 2) PASCAL VOC

5.3.1 รายละเอียดการฝึกกับชุดข้อมูล UA-DETRAC

ในการฝึกแบบจำลอง YOLOv3 กับชุดข้อมูล UA-DETRAC เริ่มจากทำการแช่ตัวแปรค่าน้ำหนักในส่วนของโครงข่ายกระดูกสันหลัง (Backbone) และทำการฝึกเป็นจำนวน 200 Epochs หลังจากนั้นจึงยกเลิกการแช่ตัวแปรค่าน้ำหนักที่โครงข่ายกระดูกสันหลัง และทำการฝึกต่อเป็นอีกจำนวน 200 Epochs โดยแบบจำลองที่ผ่านการฝึกสอนแล้ว จะความแม่นยำกับชุดข้อมูลตรวจสอบที่ 51.14% และ ความแม่นยำกับชุดข้อมูลทดสอบที่ 63.68%

5.3.2 รายละเอียดการฝึกกับชุดข้อมูล PASCAL VOC

ในการฝึกแบบจำลอง YOLOv3 กับชุดข้อมูล PASCAL VOC เริ่มจากทำการแช่ตัวแปรค่าน้ำหนักในส่วนของโครงข่ายกระดูกสันหลัง (Backbone) และทำการฝึกเป็นจำนวน 200 Epochs หลังจากนั้นจึงยกเลิกการแช่ตัวแปรค่าน้ำหนักที่โครงข่ายกระดูกสันหลัง และทำการฝึกต่อเป็นอีกจำนวน 200 Epochs โดยแบบจำลองที่ผ่านการฝึกสอน มีความแม่นยำกับชุดข้อมูลตรวจสอบที่ 44.71% และ ความแม่นยำกับชุดข้อมูลทดสอบที่ 45.18%

5.4 วิธีการพื้นฐาน

ในหัวข้อนี้จะกล่าวถึงเกณฑ์การจัดอันดับพื้นฐานซึ่งมีหน้าที่ประเมินความสำคัญของตัวแปรน้ำหนักแต่ละตัวเพื่อที่จะสามารถเลือกตัดตัวแปรค่าน้ำหนักที่มีความสำคัญน้อยที่สุด หรือมีผลกระทบต่อแบบจำลองน้อยที่สุดออกไปก่อน โดยเกณฑ์จัดอันดับพื้นฐานเหล่านี้จะถูกนำมาใช้ในการทดลองเพื่อเปรียบเทียบกับวิธีการที่งานวิจัยชิ้นนี้นำเสนอ โดยเกณฑ์การจัดอันดับพื้นฐานประกอบไปด้วย 4 วิธีการดังนี้

1. Random เลือกตัวกรองด้วยการสุ่ม
2. Magnitude เลือกตัวกรองจากตัวแปรน้ำหนัก
3. APoZ เลือกตัวกรองจากค่าจากฟังก์ชันกระตุ้น
4. ThiNet เลือกตัวกรองจากค่าน้ำหนักที่ทำให้ค่าในชั้นถัดไปเกิดการเปลี่ยนแปลงน้อยที่สุด

บทที่ 6
การทดลองและผลการทดลอง

ในบทนี้จะกล่าวถึงเรื่องของการทดลอง และผลการทดลอง ของวิธีการที่นำเสนอในงานวิจัยชิ้นนี้ ซึ่งประกอบไปด้วย 6 หัวข้อย่อยดังต่อไปนี้

1) ปัญหาของค่าของตัวแปรน้ำหนักที่เกิดขึ้นในแบบจำลอง YOLOv3
2) ผลการทดลองกลไกการตัดแบบทนทาน
3) เวลาที่ใช้ประมวลผลของแบบจำลองหลังผ่านกระบวนการตัดด้วยกลไกการตัดแบบทนทาน
4) ประสิทธิภาพของการใช้งานโมดูลการจำกัดการตัดในแต่ละขั้นตอนการตัดแบบจำลอง
5) เปรียบเทียบผลลัพธ์ของการใช้งานเกณฑ์การหยุดเปรียบเทียบกับมนุษย์
6) การทดลองเพิ่มเติมอื่น ๆ

6.1 ปัญหาของค่าของตัวแปรน้ำหนักที่เกิดขึ้นในแบบจำลอง YOLOv3

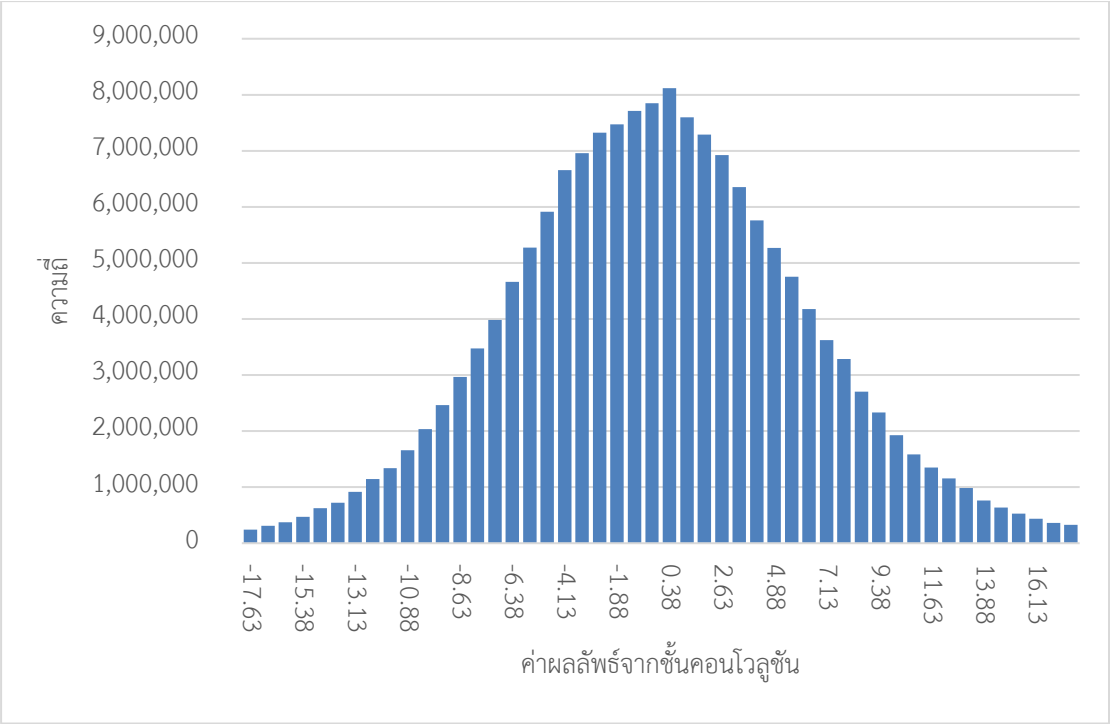
ในหัวข้อนี้จะกล่าวถึงปัญหาที่เกิดขึ้นในแบบจำลอง โดยจากการตรวจสอบค่าน้ำหนักเบื้องต้นของแบบจำลอง YOLOv3 ซึ่งประกอบไปด้วยโครงข่ายสองส่วนคือ 1) โครงข่ายกระดูกสันหลัง และ 2) โครงข่ายพีระมิดฟีเจอร์ โดยจะทำการวัดความแตกต่างของทั้งสองโครงข่ายนี้ด้วยเครื่องมือฮิสโทแกรม และค่าทางสถิติซึ่งประกอบไปด้วย 1) ค่าเฉลี่ย 2) ค่าเบี่ยงเบนมาตรฐาน 3) ความเบ้ (Skewness) 4) ความโด่ง (Kurtosis) 5) ค่าต่ำสุด และ 6) ค่าสูงสุด

6.1.1 ชุดข้อมูล UA-DETRAC

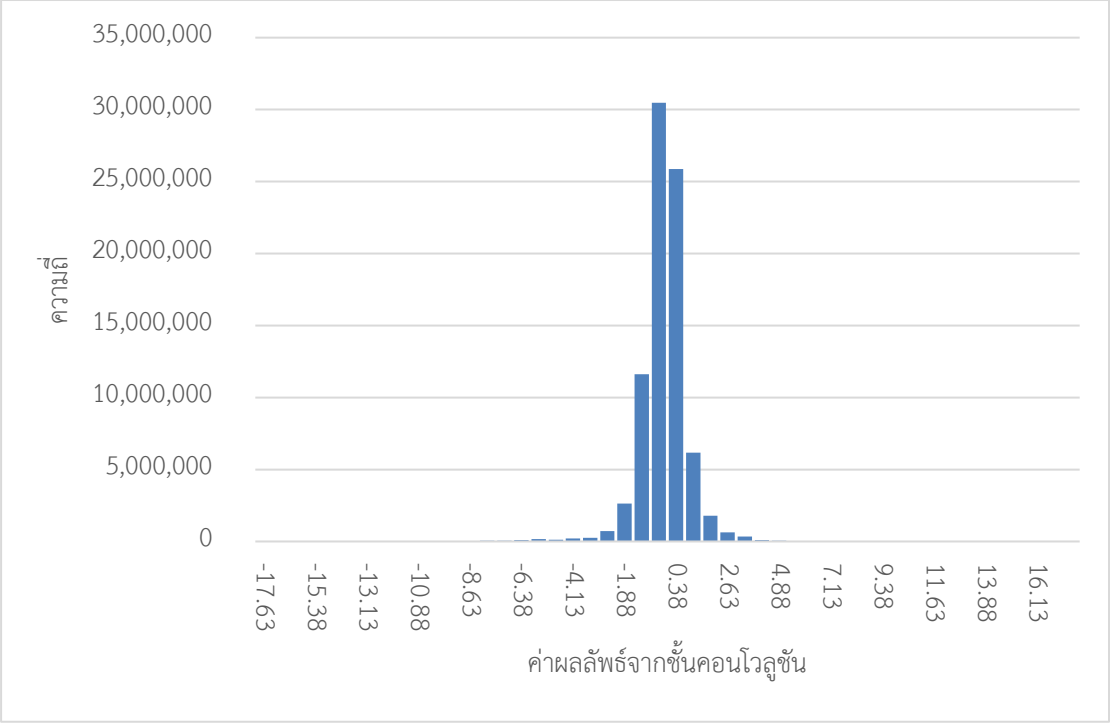
ในเบื้องต้น จากการตรวจสอบค่าของตัวแปรน้ำหนักของโครงข่ายประสาทเทียมด้วยเครื่องค่าทางสถิติและฮิสโทแกรม แล้วพบว่า ลักษณะการแจกแจงตัวของค่าจากฟังก์ชันกระตุ้นของโครงข่ายกระดูกสันหลังจะมีลักษณะใกล้เคียงกับการแจกแจงแบบปกติ (Normal Distribution) มากกว่าโครงข่ายพีระมิดฟีเจอร์ ซึ่งสามารถเห็นได้ชัดเจนจากรูปฮิสโทแกรม ในรูปที่ 20 และรูปที่ 21 นอกจากนี้หากตรวจสอบค่าอย่างละเอียดด้วยค่าทางสถิติจะพบว่า จากตารางที่ 2 ค่าทางสถิติบ่งชี้ว่าการแจกแจงมีลักษณะใกล้เคียงกับการแจกแจงแบบปกติด้วยเช่นกัน โดยมีค่าความเบ้ใกล้เคียงกับค่า 0 และค่าความโด่งใกล้เคียงกับค่า 3 ในทางกลับกันสำหรับค่าจากฟังก์ชันกระตุ้นจากโครงข่ายพีระมิดฟีเจอร์ จะลักษณะที่แตกต่างกับโครงข่ายกระดูกสันหลังอย่างชัดเจนจากภาพฮิสโทแกรม และค่าทางสถิติยังบ่งชี้ด้วยว่าการแจกแจงของส่วนของโครงข่ายพีระมิดฟีเจอร์มีความแตกต่างจากการแจกแจงแบบปกติอย่างชัดเจนเนื่องจากว่าค่าความเบ้มีความแตกต่างจากศูนย์และค่าความโด่งแตกต่างจากค่า 3 มาก

ตารางที่ 2 ค่าทางสถิติของแบบจำลอง YOLOv3 กับชุดข้อมูล UA-DETRAC

ชั้น	ค่าเฉลี่ย	ค่าเบี่ยงเบนมาตรฐาน	ความเบ้	ความโด่ง	ค่าต่ำสุด	ค่าสูงสุด
โครงข่ายกระดูกสันหลัง	-0.0626	6.7125	0.1374	3.7700	-51.0073	51.8578
โครงข่ายพีระมิดฟีเจอร์	-0.1439	1.0944	-1.0982	26.1652	-17.2669	16.6356



รูปที่ 20 การแจกแจงของค่าจากฟังก์ชันกระตุ้นในส่วนของชั้นในโครงข่ายกระดูกสันหลัง ของแบบจำลอง YOLOv3 กับชุดข้อมูล UA-DETRAC



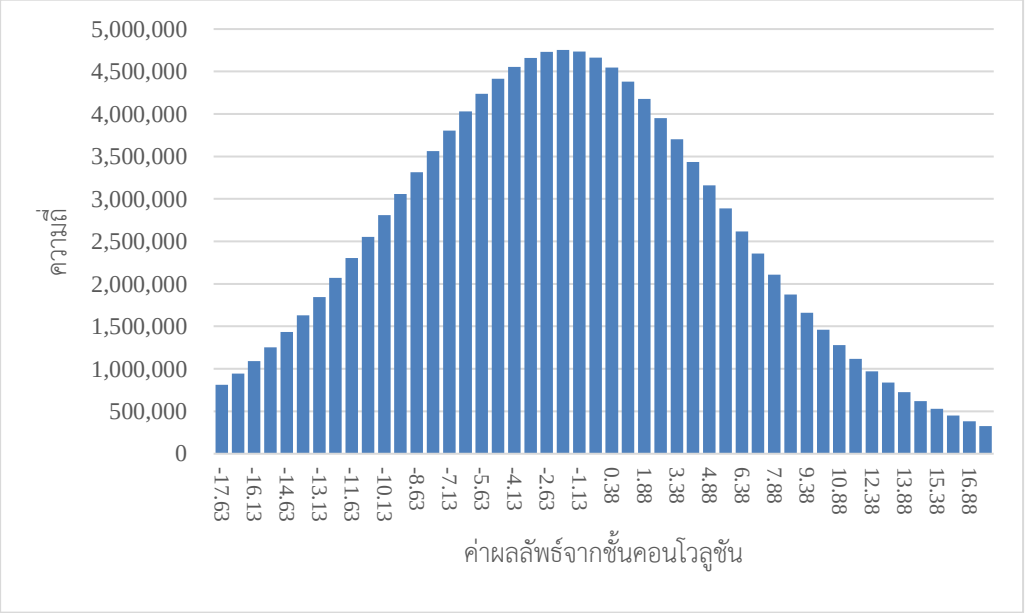
รูปที่ 21 การแจกแจงของค่าจากฟังก์ชันกระตุ้นในส่วนของชั้นในโครงข่ายพีระมิดฟีเจอร์ ของแบบจำลอง YOLOv3 กับชุดข้อมูล UA-DETRAC

6.1.2 ชุดข้อมูล PASCAL VOC

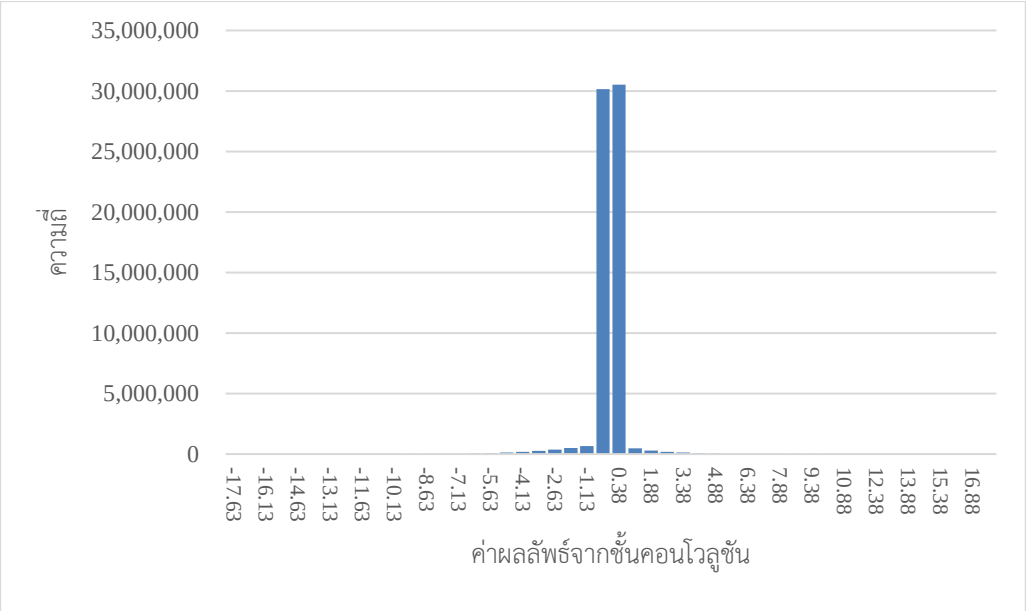
จากจากค่าทางสถิติพบว่าค่าของตัวแปรน้ำหนักจากแบบจำลอง YOLOv3 กับชุดข้อมูล PASCAL VOC ซึ่งแสดงไว้ในตารางที่ 3 และลักษณะการแจกแจงที่แสดงไว้ในรูปที่ 22 และรูปที่ 23 มีลักษณะไปในทิศทางเดียวกันกับชุดข้อมูล UA-DETRAC กล่าวคือ ลักษณะของตัวแปรค่าน้ำหนักของโครงข่ายกระดูกสันหลัง (Backbone Network) และโครงข่ายพีระมิดฟีเจอร์ (Feature Pyramid Network) มีความแตกต่างกัน โดยลักษณะของโครงข่ายกระดูกสันหลังมีความเหมือนกับการแจกแจงแบบปกติมากกว่า ในขณะที่โครงข่ายพีระมิดฟีเจอร์จะต่างกับการแจกแจงแบบปกติมากกว่า

ตารางที่ 3 ค่าทางสถิติของแบบจำลอง YOLOv3 กับชุดข้อมูล PASCAL VOC

ชั้น	ค่าเฉลี่ย	ค่าเบี่ยงเบนมาตรฐาน	ความเบ้	ความโด่ง	ค่าต่ำสุด	ค่าสูงสุด
โครงข่ายกระดูกสันหลัง	-0.0114	0.3301	-0.0601	3.3976	-2.6145	1.9013
โครงข่ายพีระมิดฟีเจอร์	-0.0000	0.0044	1.7576	1452.7191	-0.4377	0.5132



รูปที่ 22 การแจกแจงของค่าในโครงข่ายกระดูกสันหลังของแบบจำลอง YOLOv3 บน ชุดข้อมูล PASCAL VOC



รูปที่ 23 การแจกแจงของค่าในโครงข่ายพีระมิดฟีเจอร์ของแบบจำลอง YOLOv3 กับชุดข้อมูล PASCAL VOC

6.2 ผลการทดลองกลไกการตัดแบบทนทาน

ในหัวข้อนี้จะทำการทดลองใช้งานกลไกการตัดแบบทนทาน (RPM) ทดลองทำการตัดแบบจำลอง และเปรียบเทียบกับเกณฑ์การตัดพื้นฐาน หลังจากนั้นจะอธิบายถึงผลการทดลองการใช้กลไกการตัดแบบทนทานเมื่อเปรียบเทียบกับเกณฑ์การตัดพื้นฐาน ในขั้นตอนการทดลอง จะทำการตัดแบบจำลอง YOLOv3 ด้วยวิธีเกณฑ์การตัดแบบต่าง ๆ และนำมาความแม่นยำมาเปรียบเทียบกันในแต่ละสัดส่วนการตัด (Pruned Away) ของแบบจำลองที่เท่ากัน โดยความหมายของสัดส่วนการตัด คือ สัดส่วนของตัวแปรน้ำหนักที่ถูกตัดออกจากแบบจำลอง การทดลองจะประกอบไปด้วยชุดข้อมูล 2 ชุด คือ 1) UA-DETRAC และ 2) PASCAL VOC

6.2.1 ประสิทธิภาพของกลไกการตัดแบบทนทานกับชุดข้อมูล UA-DETRAC

เพื่อทดสอบกลไกการตัดแบบทนทาน จึงได้ทำการทดลองตัดแบบจำลองด้วยวิธีการต่าง ๆ ประกอบด้วยวิธีการพื้นฐาน 4 วิธีการ ประกอบไปด้วย 1) Random 2) APoZ 3) Magnitude และ 4) ThiNet โดยในการทดลองจะใช้วิธีการต่าง ๆ ทดลองทำการตัดแบบจำลอง เพื่อบีบอัดขนาดของแบบจำลองและเปรียบเทียบความแม่นยำในแต่ละสัดส่วนการตัดเพื่อเปรียบเทียบประสิทธิภาพของแต่ละวิธีการ

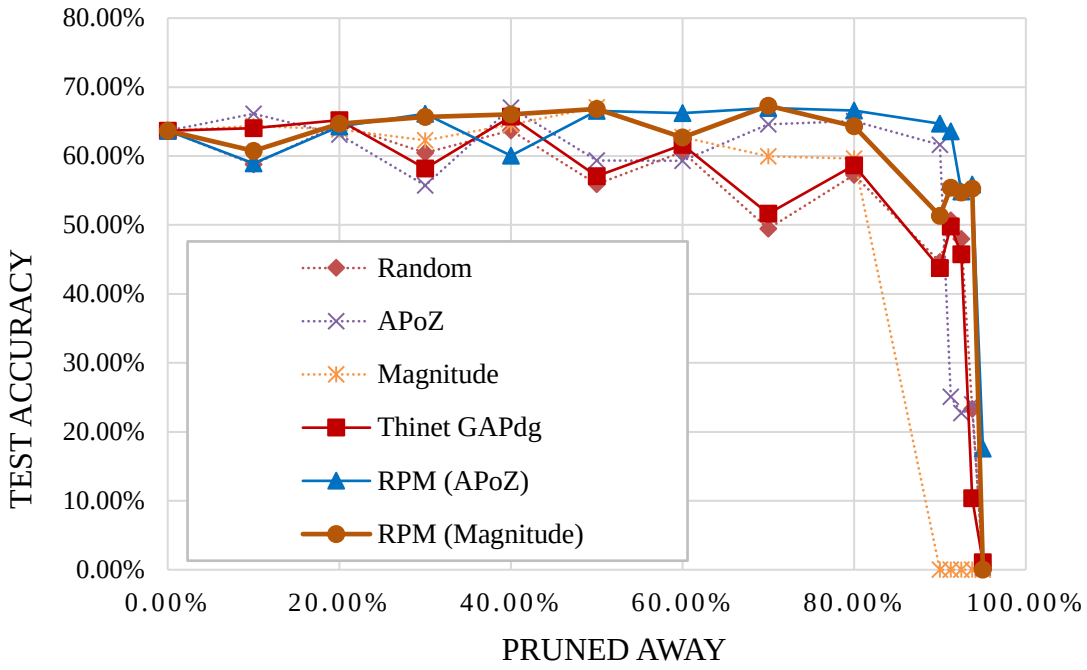
จากผลการทดลองในตารางที่ 4 พบว่า จากวิธีการพื้นฐานทั้ง 4 วิธี วิธีการ APoZ นั้นเป็นวิธีการที่ดีที่สุดและรองลงมาคือ วิธีการ Magnitude โดยสามารถสังเกตได้จากว่า วิธีการ APoZ นั้นสามารถคงความแม่นยำไว้ใกล้เคียงความแม่นยำเริ่มต้น ได้ถึงสัดส่วนการตัดที่ 90% ในขณะที่วิธีการ Magnitude สามารถคงความแม่นยำไว้ได้ถึงสัดส่วนการตัดที่ 80% นอกจากนี้ สำหรับวิธีการ Random ถึงแม้ว่าจะมีประสิทธิภาพพอใช้งานได้ แต่พบว่ามีความผันผวนของอัตราความแม่นยำของแบบจำลองสูงมาก จึงทำให้ไม่เหมาะสำหรับการนำมาใช้งานจริง ด้วยเหตุนี้จึงเลือกวิธีการ APoZ และ Magnitude ซึ่งเป็นวิธีการพื้นฐานที่ดีที่สุดมาเป็นตัวแทนเพื่อใช้กับกลไกการตัดแบบทนทาน

ในการทดลองถัดมา จะแบ่งเป็นสองการทดลองบนกลไกการตัดแบบทนทาน คือ 1) RPM (APoZ) และ 2) RPM (Magnitude) โดยจากตารางที่ 4 พบว่าวิธี RPM (APoZ) สามารถทำงานได้ดีที่สุด โดยสามารถคงความแม่นยำได้ที่ 63.57% ในขณะที่แบบจำลองมีสัดส่วนการตัดอยู่ที่ 91.25% ซึ่งมีประสิทธิภาพดีกว่าวิธีการอื่น ๆ ในสัดส่วนการตัดที่เท่ากัน

ตารางที่ 4 ความแม่นยำกับชุดข้อมูลทดสอบของผลการตัดแบบจำลอง YOLOv3 ด้วยกลไกการตัดแบบทนทาน และวิธีการพื้นฐาน

Pruned away	Random	APoZ	Magnitude	ThiNet	RPM (APoZ)	RPM (Magnitude)
0.00%	63.68%	63.68%	63.68%	63.68%	63.68%	63.68%
10.00%	58.82%	**66.13%**	64.36%	64.03%	58.92%	60.73%
20.00%	64.59%	63.13%	63.84%	**65.20%**	64.28%	64.70%
30.00%	60.47%	55.73%	62.29%	58.18%	**66.15%**	65.66%
40.00%	63.73%	**67.01%**	64.56%	65.71%	60.04%	66.04%
50.00%	55.92%	59.34%	**67.06%**	57.08%	66.57%	66.84%
60.00%	60.60%	59.30%	62.76%	61.64%	**66.22%**	62.71%
70.00%	49.45%	64.60%	59.94%	51.64%	66.97%	**67.30%**
80.00%	57.25%	65.06%	59.63%	58.68%	**66.62%**	64.31%
90.00%	44.62%	61.63%	0.00%	43.77%	**64.70%**	51.33%
91.25%	50.67%	25.07%	0.00%	49.80%	**63.57%**	55.41%
92.50%	47.98%	22.74%	0.00%	45.74%	**54.86%**	54.70%
93.75%	23.34%	23.98%	0.00%	10.36%	**55.89%**	55.27%
95.00%	0.63%	0.29%	0.00%	1.07%	**17.55%**	0.00%

ตัวอักษรหนา คือ วิธีการที่ดีที่สุด **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง **พื้นหลังไม่มีสี** คือ ความแม่นยำของแบบจำลอง



รูปที่ 24 กราฟผลการตัดแบบจำลอง YOLOv3 ด้วยกลไกการตัดแบบทนทาน และวิธีการพื้นฐาน

6.2.2 ประสิทธิภาพของกลไกการตัดแบบทนทานกับชุดข้อมูล PASCAL VOC

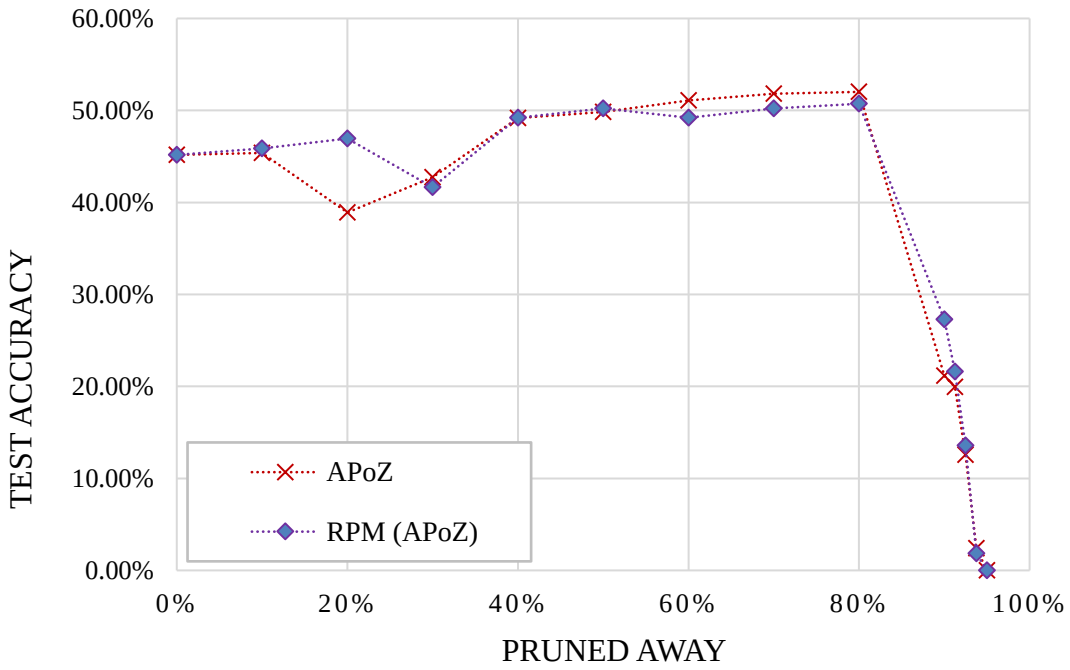
ในหัวข้อนี้ จะแสดงและสรุปผลการทดลองวิธีการตัดแบบจำลองที่นำเสนอเปรียบเทียบกับวิธีการพื้นฐานกับชุดข้อมูล PASCAL VOC ซึ่งเป็นชุดข้อมูลตรวจจับวัตถุทั่วไป เนื่องจากในการทดลองในหัวข้อที่ 6.2.1 พบว่าวิธีการ APoZ และ RPM (APoZ) เป็นวิธีการที่ดีที่สุด ดังนั้นการทดลองในหัวข้อนี้ จะเลือกทั้งสองวิธีการดังกล่าวเป็นตัวแทนในการทดลอง

จากผลการทดลองในตารางที่ 5 พบว่า ในสัดส่วนการตัดที่ 0% ถึง 80% วิธีการ APoZ และวิธีการ RPM (APoZ) มีประสิทธิภาพใกล้เคียงกัน แต่ว่าหลังจากนั้นในสัดส่วนการตัดตั้งแต่ 90% ขึ้นไป วิธีการ RPM (APoZ) จะมีความแม่นยำมากกว่าวิธีการ APoZ เล็กน้อย โดยสำหรับแบบจำลองที่เหมาะสมที่สุดจะเป็น แบบจำลองจากวิธีการ APoZ ที่สัดส่วนการตัด 80% เนื่องจากว่าเป็นแบบจำลองที่มีความแม่นยำสูงสุดที่ 52.02% และถูกตัดตัวแปรน้ำหนักไปในระดับที่เพียงพอต่อการใช้งานแล้ว โดยสำหรับสาเหตุที่วิธีการ RPM (APoZ) แย่กว่า APoZ นั้นจะทำการวิเคราะห์ และกล่าวในหัวข้อ 6.6.4 ต่อไป

ตารางที่ 5 ความแม่นยำกับชุดข้อมูลทดสอบของผลการตัดแบบจำลอง YOLOv3 กับชุดข้อมูล PASCAL VOC

Pruned away	APoZ	RPM (APoZ)
0.00%	45.18%	45.18%
10.00%	45.39%	**45.86%**
20.00%	38.92%	**46.96%**
30.00%	**42.72%**	41.67%
40.00%	49.17%	**49.20%**
50.00%	49.82%	**50.22%**
60.00%	**51.10%**	49.20%
70.00%	**51.81%**	50.23%
80.00%	**52.02%**	50.73%
90.00%	21.17%	**27.28%**
91.25%	19.95%	**21.61%**
92.50%	12.57%	**13.58%**
93.75%	**2.40%**	1.84%
95.00%	0.00%	0.00%

ตัวอักษรหนา คือ วิธีการที่ดีที่สุด **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง **พื้นหลังไม่มีสี** คือ ความแม่นยำของแบบจำลอง



รูปที่ 25 กราฟผลการตัดแบบจำลองกับชุดข้อมูล PASCAL VOC

6.3 เวลาที่ใช้ประมวลผลของแบบจำลองหลังผ่านกระบวนการตัดด้วยกลไกการตัดแบบทนทาน

ในหัวข้อนี้จะกล่าวถึงความเร็วของแบบจำลองหลังผ่านการตัดแบบจำลอง เพื่อตรวจสอบว่าแบบจำลองมีความเร็วมากแตกต่างจากแบบจำลองดั้งเดิมอย่างไร ในการวิเคราะห์จะเลือกใช้ FLOP (Floating Point Operation) เป็นแทนในการวัดประสิทธิภาพของแบบจำลอง ซึ่งผลการคำนวณต้นทุนของแบบจำลอง จะแสดงไว้ในตารางที่ 6 จากผลการคำนวณ FLOP ของแบบจำลองแล้วพบว่า หลังจากที่ตัดตัวแปรออกจากแบบจำลอง ส่งผลให้แบบจำลองมีความเร็วเพิ่มขึ้นจริง และพบว่าวิธีการ RPM (APoZ) เป็นวิธีการที่ดีที่สุด ซึ่งสามารถสังเกตได้จากที่สัดส่วนการพรุนที่ 93.75% ที่แบบจำลองมีค่าความแม่นยำอยู่ที่ 55.89% และค่า FLOP อยู่ที่ 1.80 ซึ่งพบว่า มีความแม่นยำสูงกว่าวิธีการอื่น ที่ FLOP ใกล้เคียงกัน

ตารางที่ 6 ค่า FLOP และค่าความแม่นยำ (ACC) กับชุดข้อมูลทดสอบ UA-DETRAC ของแต่ละสัดส่วนการตัดของแบบจำลองในวิธีการตัดในแบบต่าง ๆ

	ACC	FLOP	ACC	FLOP	ACC	FLOP	ACC	FLOP	ACC	FLOP
Pruned away	Random	Random	APOZ	APOZ	Mag	Mag	ThiNet	ThiNet	RPM (APoZ)	RPM (APoZ)
0.00%	63.68%	111.31	63.68%	111.31	63.68%	111.31	63.68%	111.31	63.68%	111.31
10.00%	58.82%	92.13	**66.13%**	97.25	64.36%	94.98	64.03%	91.72	58.92%	97.79
20.00%	64.59%	73.62	63.13%	80.03	63.84%	79.59	**65.20%**	74.16	64.28%	84.26
30.00%	60.47%	57.68	55.73%	68.41	62.29%	64.47	58.18%	58.28	**66.15%**	70.22
40.00%	63.73%	44.44	**67.01%**	58.82	64.56%	50.45	65.71%	44.43	60.04%	57.15
50.00%	55.92%	31.68	59.34%	46.18	**67.06%**	38.85	57.08%	32.28	66.57%	43.80
60.00%	60.60%	22.00	59.30%	35.47	62.76%	28.82	61.64%	22.10	**66.22%**	33.97
70.00%	49.45%	13.87	64.60%	23.76	59.94%	12.58	51.64%	13.66	**66.97%**	21.76
80.00%	57.25%	7.26	65.06%	11.80	59.63%	10.41	58.68%	7.19	**66.62%**	10.26
90.00%	44.62%	2.69	61.63%	3.26	0.00%	1.59	43.77%	2.47	**64.70%**	3.87
91.25%	50.67%	2.16	25.07%	2.48	0.00%	1.12	49.80%	2.06	**63.57%**	3.40
92.50%	47.98%	1.69	22.74%	1.80	0.00%	0.83	45.74%	1.61	**54.86%**	2.33
93.75%	23.34%	1.28	23.98%	1.33	0.00%	0.73	10.36%	1.20	**55.89%**	1.80
95.00%	0.63%	0.86	0.29%	0.88	0.00%	0.73	1.07%	0.87	**17.55%**	1.09

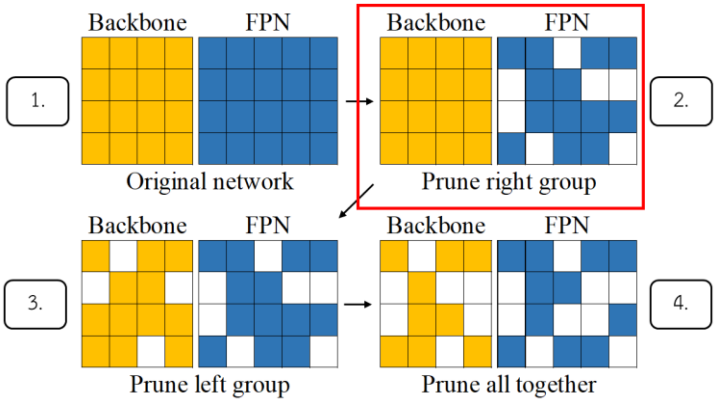
ตัวอักษรหนา คือ วิธีการที่ดีที่สุด **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง

6.4 ประสิทธิภาพของการใช้งานโมดูลการจำกัดการตัดในแต่ละขั้นตอนการตัดแบบจำลอง

ในหัวข้อนี้จะพูดถึงผลลัพธ์ของการใช้งานตัวโมดูลการจำกัดการตัด (Minimum Filter Constraint) เมื่อใช้งานในแต่ละขั้นของกลไกการตัดแบบทนทาน รวมอธิบายถึงผลลัพธ์ และสาเหตุของปัญหาจากการทดลอง

6.4.1 การใช้งานการจำกัดการตัดในขั้นตอนที่สองของกลไกการตัดแบบทนทาน

ในหัวข้อนี้จะพูดถึงผลกระทบของการใช้การจำกัดการตัดกับแบบจำลองเมื่อถูกใช้งานในขั้นที่ 2 ของกลไกการตัดแบบทนทาน ซึ่งเป็นขั้นตอนการตัดแบบจำลองที่ส่วนของฟีเจอร์พีระมิด ซึ่งแสดงให้เห็นดังรูปที่ 26



รูปที่ 26 ภาพสถานะที่ถูกใช้งานการจำกัดการตัด กล่องที่มีสี คือ ส่วนที่ยังไม่ได้ตัดตัวแปรน้ำหนักออก กล่องที่เป็นสีขาว คือ ส่วนที่ถูกตัดตัวแปรน้ำหนักออกแล้ว

จากผลการทดลองตามตารางที่ 7 พบว่า การใส่การจำกัดการตัดตั้งแต่ขั้นตอนที่ 2 ของกลไกการตัดแบบทนทานพบว่าให้ผลลัพธ์ที่แย่ลง ซึ่งจากผลการทดลองพบว่า การไม่ใช้โมดูลการจำกัดการตัดสามารถคงความแม่นยำไว้ที่ระดับใกล้เคียงเดิมได้จนถึงสัดส่วนการตัดที่ 80% โดยมีความแม่นยำอยู่ที่ 66.62% โดยเมื่อเทียบกับรูปแบบที่ใช้การจำกัดการตัดจะพบว่าสามารถคงความแม่นยำไว้ใกล้เคียงเดิมได้จนถึงสัดส่วนการตัดที่ 70% ซึ่งมีความแม่นยำอยู่ที่ 59.21% โดยเมื่อดูจากตารางที่ 8 และตารางที่ 9 จะพบว่าสาเหตุที่แบบจำลองทำงานได้ไม่ดีนั้นมีสาเหตุมาจากว่าโดยปกติแล้ว เมื่อไม่ได้มีการใช้งานโมดูลการจำกัดการตัด จะพบว่าที่อัตราส่วนการตัดที่ 80% ชั้นคอนโวลูชัน 4 ชั้นแรกนั้น สามารถถูกตัดจนเหลือเพียง 1 - 2 ตัวกรองได้โดยไม่กระทบต่อความแม่นยำของแบบจำลอง ในขณะเดียวกัน เมื่อมีการใช้งานโมดูลจำกัดการตัด ทำให้กระบวนการตัดจำเป็นต้องกันตัวกรองจำนวนหนึ่งไว้ในชั้นคอนโวลูชัน 4 ชั้นแรก ซึ่งส่งผลให้ชั้นอื่น ๆ จะถูกตัดมากกว่าแบบแรก ซึ่งการที่ความแม่นยำของแบบที่ใช้การจำกัดการตัดนั้นต่ำกว่า ก็สามารถสรุปได้ว่าชั้นคอนโวลูชันชั้นหลัง ๆ (conv2d_5 ถึง conv2d_22) มีความสำคัญมากกว่าชั้นที่อยู่อันดับต้น ๆ สำหรับส่วนของโครงข่ายพีระมิดฟีเจอร์

ตารางที่ 7 ผลการใช้การจำกัดการตัดในขั้นตอนการตัดส่วนของ FPN (ความแม่นยำกับชุดข้อมูลทดสอบ)

Pruned away	FPN (APoZ with minimum filter)	FPN (APoZ without minimum filter)
0.00%	63.68%	63.68%
10.00%	**63.21%**	58.92%
20.00%	64.18%	**64.28%**
30.00%	60.03%	**66.15%**
40.00%	**65.48%**	60.04%
50.00%	**66.81%**	66.57%
60.00%	65.40%	**66.22%**
70.00%	65.62%	**66.97%**
80.00%	59.21%	**66.62%**
82.50%	2.33%	**14.80%**

ตัวอักษรหนา คือ วิธีการที่ดีที่สุด **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง **พื้นหลังไม่มีสี** คือ ความแม่นยำของแบบจำลอง

ตารางที่ 8 จำนวนตัวกรอง (Filter) ที่เหลืออยู่ของส่วนของโครงข่ายพีระมิด กรณีไม่ใช้การจำกัดการตัด

Without constraint				
Test accuracy	63.68%	66.22%	66.97%	66.62%
Pruned away	0.00%	60.00%	70.00%	80.00%
conv2d_1	512	3	1	**1**
conv2d_2	1024	3	1	**1**
conv2d_3	512	5	3	**2**
conv2d_4	1024	3	3	**2**
conv2d_5	512	512	512	26
conv2d_6	1024	708	190	50
conv2d_10	512	18	16	15
conv2d_12	512	31	27	24
conv2d_14	512	512	512	50
conv2d_18	256	69	57	56
conv2d_20	256	60	55	52
conv2d_22	256	256	256	77

ตัวอักษรหนา คือ จุดที่ใช้เปรียบเทียบความแตกต่างของการตัดแบบมีการจำกัดการตัด และแบบที่ไม่มีการจำกัดการตัด **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง

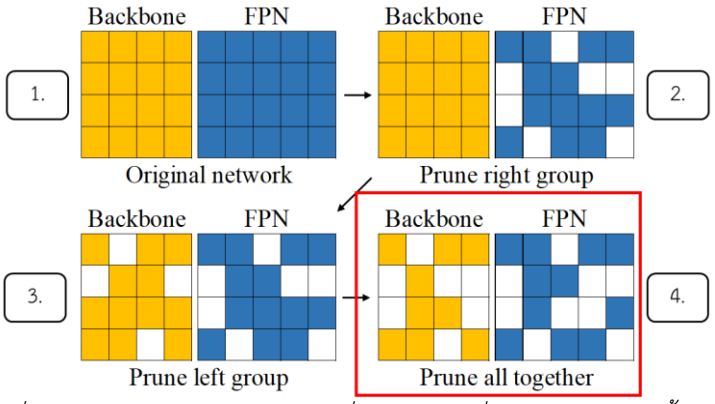
ตารางที่ 9 จำนวนตัวกรอง (Filter) ที่เหลืออยู่ของส่วนของโครงข่ายพีระมิด กรณีใช้การจำกัดการตัด ซึ่งสามารถเกตได้จาก conv2d_1 ถึง conv2d_4 ที่มีการสงวนตัวกรองไว้ 16 ตัวกรอง

With constraint				
Test accuracy	63.68%	65.40%	65.62%	59.21%
Pruned away	0.00%	60.00%	70.00%	80.00%
conv2d_1	512	16	16	**16**
conv2d_2	1024	16	16	**16**
conv2d_3	512	16	16	**16**
conv2d_4	1024	16	16	**16**
conv2d_5	512	512	512	16
conv2d_6	1024	685	159	42
conv2d_10	512	17	16	16
conv2d_12	512	24	23	17
conv2d_14	512	512	512	50
conv2d_18	256	58	54	20
conv2d_20	256	52	50	20
conv2d_22	256	256	256	54

ตัวอักษรหนา คือ จุดที่ใช้เปรียบเทียบความแตกต่างของการตัดแบบมีการจำกัดการตัด และแบบที่ไม่มีการจำกัดการตัด **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง

6.4.2 การใช้งานการจำกัดการตัดในขั้นตอนที่สี่ของกลไกการตัดแบบทนทาน

ในหัวข้อนี้จะพูดถึงผลกระทบของการใช้การจำกัดการตัดกับแบบจำลองเมื่อถูกใช้งานตั้งแต่ในขั้นที่ 4 ของกลไกการตัดแบบทนทานหรือขั้นตอนการตัดแบบจำลองที่รวมทั้งส่วนของโครงข่ายกระดูกสันหลังและพีระมิดฟีเจอร์ ซึ่งแสดงให้เห็นดังรูปที่ 27



รูปที่ 27 ภาพสถานะที่ถูกใช้งานการจำกัดการตัด กล่องที่มีสี คือ ส่วนที่ยังไม่ได้ตัดตัวแปรน้ำหนักออก กล่องที่เป็นสีขาว คือ ส่วนที่ถูกตัดตัวแปรน้ำหนักออกแล้ว

จากผลการทดลองตามตารางที่ 10 พบว่า การใส่การจำกัดการตัดตั้งแต่ขั้นที่ 4 ของกลไกการตัดแบบทนทาน นั้นให้ผลลัพธ์ที่ดีกว่าการที่ไม่ใช้งานโมดูลนี้ จากผลการทดลองพบว่า การใช้การจำกัดการตัดสามารถคงความแม่นยำไว้ที่ 56.00% ที่สัดส่วนการตัดที่ 93.75% โดยเมื่อเทียบกับรูปแบบที่ไม่ใช้โมดูลการจำกัดการตัดจะพบว่าสามารถคงความแม่นยำไว้ที่ 52.85% ที่สัดส่วนการตัดที่ 93.75% เนื่องจากว่าในสัดส่วนการตัดในช่วงท้ายนั้นมีโอกาสที่จะตัดตัวกรองบางชั้นมากเกินไปจนมีไม่เพียงพอที่จะทำงานได้ ทำให้แบบจำลองเกิดความเสียหายและไม่สามารถคงความแม่นยำไว้ได้ ซึ่งสามารถสังเกตได้จากตารางที่ 11 และ ตารางที่ 12 ว่ากรณีที่ใช้งานโมดูลการจำกัดการตัดจะมีการกันตัวกรองไว้จำนวนหนึ่งที่ conv2d_5 ถึง conv2d_22 แต่ในทางกลับกัน สำหรับกรณีที่ไม่ได้ใช้งานโมดูลการจำกัดการตัดจะเหลือจำนวนตัวกรองในส่วนของ conv2d_5 ถึง conv2d_22 จะเหลือน้อยมาก ส่งผลให้ตัวแบบจำลองไม่สามารถคงระดับความแม่นยำเอาไว้ได้

ตารางที่ 10 ผลการใช้การจำกัดการตัดในขั้นตอนการตัดส่วนของขั้นที่ 4 ของกลไกการตัดแบบทนทาน (ความแม่นยำกับชุดข้อมูลทดสอบ)

Pruned away	RPM (APoZ with minimum filter)	RPM (APoZ without minimum filter)
91.25%	63.57%	63.57%
92.50%	54.86%	**54.90%**
93.75%	**55.89%**	14.21%
95.00%	**17.55%**	0.08%

ตัวอักษรหนา คือ วิธีการที่ดีที่สุด **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง **พื้นหลังไม่มีสี** คือ ความแม่นยำของแบบจำลอง

ตารางที่ 11 จำนวนตัวกรอง (Filter) ที่เหลืออยู่ของแบบจำลอง กรณีไม่ใช้การจำกัดการตัด

Without constraint					
Test accuracy	63.68%	63.57%	54.90%	14.21%	0.08%
Pruned away	0.00%	91.25%	92.50%	0.9375	95.00%
conv2d_1	512	1	1	1	1
conv2d_2	1024	1	1	1	1
conv2d_3	512	2	2	1	1
conv2d_4	1024	2	1	1	1
conv2d_5	512	26	**4**	**3**	**2**
conv2d_6	1024	50	**36**	**24**	**12**
conv2d_10	512	15	**10**	**1**	**1**
conv2d_12	512	24	**13**	**1**	**1**
conv2d_14	512	50	**36**	**14**	**1**
conv2d_18	256	56	**6**	**1**	**1**
conv2d_20	256	52	**10**	**1**	**1**
conv2d_22	256	77	**45**	**12**	**1**
conv_pw_4	256	161	161	143	21
conv_pw_5	256	61	61	57	8
conv_pw_6	512	215	215	142	9
conv_pw_7	512	125	125	91	9
conv_pw_8	512	95	95	76	11
conv_pw_9	512	84	84	59	14
conv_pw_10	512	48	48	36	12
conv_pw_11	512	85	85	81	62
conv_pw_12	1024	123	123	78	33
conv_pw_13	1024	1	1	1	1

ตัวอักษรหนา คือ จุดที่ใช้เปรียบเทียบความแตกต่างของการตัดแบบมีการจำกัดการตัด และแบบที่ไม่มีการจำกัดการตัด **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง

ตารางที่ 12 จำนวนตัวกรอง (Filter) ที่เหลืออยู่ของส่วนของโครงข่ายพีระมิด กรณีใช้การจำกัดการตัด ซึ่งสามารถเกตได้จาก conv2d_5 ถึง conv2d_22 ที่มีการสงวนตัวกรองไว้ 16 ตัวกรอง

With constraint					
Test accuracy	63.68%	63.57%	54.90%	14.21%	0.08%
Pruned away	0.00%	91.25%	92.50%	0.9375	95.00%
conv2d_1	512	1	1	1	1
conv2d_2	1024	1	1	1	1
conv2d_3	512	2	2	2	2
conv2d_4	1024	2	2	2	2
conv2d_5	512	26	**16**	**16**	**8**
conv2d_6	1024	50	**28**	**16**	**7**
conv2d_10	512	15	**15**	**15**	**7**
conv2d_12	512	24	**16**	**16**	**7**
conv2d_14	512	50	**31**	**16**	**7**
conv2d_18	256	56	**16**	**16**	**7**
conv2d_20	256	52	**16**	**16**	**7**
conv2d_22	256	77	**42**	**16**	**7**
conv_pw_4	256	161	160	104	7
conv_pw_5	256	61	61	41	8
conv_pw_6	512	215	214	82	8
conv_pw_7	512	125	124	51	7
conv_pw_8	512	95	95	43	8
conv_pw_9	512	84	84	32	8
conv_pw_10	512	48	48	22	8
conv_pw_11	512	85	85	76	8
conv_pw_12	1024	123	120	63	7
conv_pw_13	1024	1	1	1	1

ตัวอักษรหนา คือ จุดที่ใช้เปรียบเทียบความแตกต่างของการตัดแบบมีการจำกัดการตัด และแบบที่ไม่มีการจำกัดการตัด **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง

6.5 ผลลัพธ์และประสิทธิภาพของการใช้งานโมดูลเกณฑ์การหยุด

ในหัวข้อนี้จะกล่าวถึงผลลัพธ์ของการใช้โมดูลเกณฑ์การหยุดในกลไกการตัดแบบทนทานอย่างละเอียด และหลังจากนั้นจะทำการเปรียบเทียบการตัดสินใจเลือกหยุดการตัดแบบจำลองของโมดูลเกณฑ์การหยุดกับการเลือกหยุดของมนุษย์

6.5.1 ผลลัพธ์ของการใช้งานโมดูลเกณฑ์การหยุดในกลไกการตัดแบบทนทาน

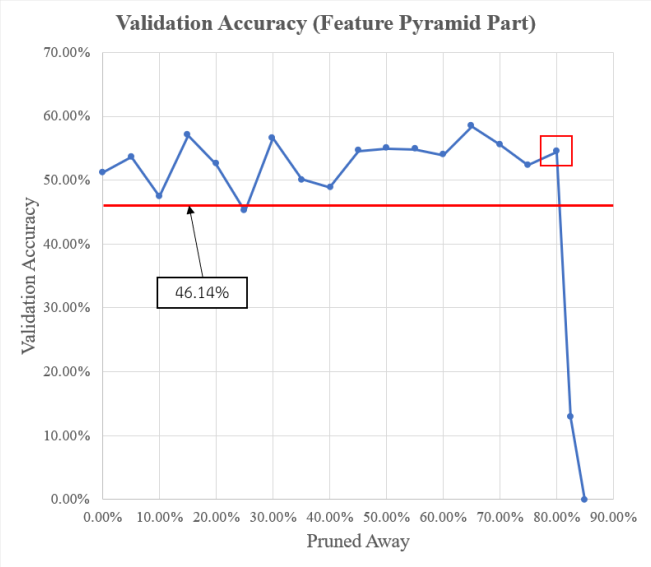
ในหัวข้อนี้จะกล่าวถึงผลลัพธ์ของการใช้งานเกณฑ์การหยุดซึ่งเป็นส่วนหนึ่งของกลไกการตัดแบบทนทาน โดยจะนำผลลัพธ์ของวิธีการ RPM (APoZ) มาแสดงอย่างละเอียดทีละขั้นตอน ซึ่งค่าขีดแบ่ง (Threshold) ของวิธีการ RPM (APoZ) จะใช้ที่ 5.00% โดยมีสาเหตุมาจากว่า 5.00% นั้นจัดเป็นขนาดที่นิยมใช้แยกความแตกต่างระหว่างแบบจำลองว่าใกล้เคียงตัวต้นฉบับหรือไม่

สำหรับส่วนแรก ซึ่งเป็นขั้นตอนที่สองของกลไกการตัดแบบทนทาน โดยส่วนนี้จะทำหน้าที่ตัดส่วนของโครงข่ายพีระมิดฟีเจอร์ จากผลลัพธ์ในตารางที่ 13 พบว่าที่สัดส่วนการตัดที่ 82.50% และ 85.00% มีการลดลงของความแม่นยำกับชุดข้อมูลตรวจสอบต่ำกว่าค่าขีดแบ่งต่อเนื่องกัน 2 ครั้ง โมดูลเกณฑ์การหยุดจึงเลือกแบบจำลองที่มีสัดส่วนการตัด 80% เป็นแบบจำลองผลลัพธ์สำหรับขั้นตอนที่สองของกลไกการตัดแบบทนทาน

ตารางที่ 13 ผลลัพธ์การใช้เกณฑ์การหยุดในขั้นที่ 2 ของกลไกการตัดแบบทนทาน (โครงข่ายพีระมิดฟีเจอร์) โดย Stop คือ จุดที่เลือกหยุด และเป็นแบบจำลองที่เลือกเป็นผลลัพธ์ Difference คือ ส่วนต่างระหว่างความแม่นยำกับชุดข้อมูลตรวจสอบ (Validation) เริ่มต้นกับความแม่นยำกับชุดข้อมูลตรวจสอบแถวปัจจุบัน

Pruned away	Validation accuracy	Test accuracy	Stop	Difference	> Threshold
0.00%	51.14%	63.68%	FALSE	0.00%	FALSE
5.00%	53.64%	63.25%	FALSE	-2.50%	FALSE
10.00%	47.40%	58.92%	FALSE	3.74%	FALSE
15.00%	57.04%	60.16%	FALSE	-5.90%	FALSE
20.00%	52.50%	64.28%	FALSE	-1.36%	FALSE
25.00%	45.19%	59.48%	FALSE	5.95%	TRUE
30.00%	56.56%	66.15%	FALSE	-5.42%	FALSE
35.00%	50.10%	59.37%	FALSE	1.04%	FALSE
40.00%	48.86%	60.04%	FALSE	2.28%	FALSE
45.00%	54.55%	64.81%	FALSE	-3.41%	FALSE
50.00%	54.93%	66.57%	FALSE	-3.79%	FALSE
55.00%	54.74%	66.31%	FALSE	-3.60%	FALSE
60.00%	53.88%	66.22%	FALSE	-2.74%	FALSE
65.00%	58.45%	66.34%	FALSE	-7.31%	FALSE
70.00%	55.57%	66.97%	FALSE	-4.43%	FALSE
75.00%	52.36%	65.59%	FALSE	-1.22%	FALSE
80.00%	**54.38%**	**66.62%**	**TRUE**	**-3.24%**	**FALSE**
82.50%	12.85%	14.80%	FALSE	38.29%	TRUE
85.00%	0.00%	0.00%	FALSE	51.14%	TRUE

ช่องที่มีตัวอักษรหนา คือ แบบจำลองผลลัพธ์ที่ใช้โมดูลเกณฑ์การหยุดเลือก **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง



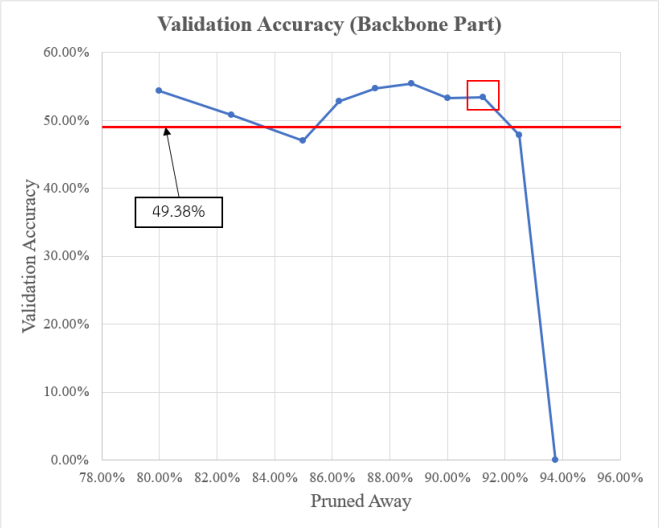
รูปที่ 28 กราฟผลลัพธ์การใช้เกณฑ์การหยุดในขั้นที่ 2 ของวิธีการ RPM (APoZ) (ตัดส่วนของโครงข่ายพีระมิดฟีเจอร์) (เส้นสีแดงที่ 46.14% มีไว้เพื่อแสดงให้เห็นว่าจุดใดที่ความแม่นยำของแบบจำลองจะเริ่มลดลงต่ำกว่าค่าขีดแบ่ง โดยคำนวณได้จาก ความแม่นยำตรวจสอบเริ่มต้น 51.14% ลบด้วยค่าขีดแบ่ง 5.00%)

สำหรับส่วนที่สอง ซึ่งเป็นขั้นตอนที่สามของกลไกการตัดแบบทนทาน โดยส่วนนี้จะทำหน้าที่ตัดส่วนของโครงข่ายกระดูกสันหลัง จากผลลัพธ์ในตารางที่ 14 พบว่าที่สัดส่วนการตัดที่ 92.50% และ 93.75% มีการลดลงของความแม่นยำกับชุดข้อมูลตรวจสอบต่ำกว่าค่าขีดแบ่งต่อเนื่องกัน 2 ครั้ง โมดูลเกณฑ์การหยุดจึงเลือกแบบจำลองที่มีสัดส่วนการตัด 91.25% เป็นแบบจำลองผลลัพธ์สำหรับขั้นตอนที่สามของกลไกการตัดแบบทนทาน

ตารางที่ 14 ผลลัพธ์การใช้เกณฑ์การหยุดในขั้นที่ 3 ของกลไกการตัดแบบทนทาน (โครงข่ายกระดูกสันหลัง) โดย Stop คือ จุดที่เลือกหยุด และเป็นแบบจำลองที่เลือกเป็นผลลัพธ์ Difference คือ ส่วนต่างระหว่างความแม่นยำกับชุดข้อมูลตรวจสอบ (Validation) เริ่มต้นกับความแม่นยำกับชุดข้อมูลตรวจสอบแถวปัจจุบัน

Pruned away	Validation accuracy	Test accuracy	Stop	Difference	> Threshold
80.00%	54.38%	66.62%	FALSE	0.00%	FALSE
82.50%	50.85%	64.37%	FALSE	3.53%	FALSE
85.00%	46.97%	59.48%	FALSE	7.41%	TRUE
86.25%	52.79%	66.10%	FALSE	1.59%	FALSE
87.50%	54.66%	65.00%	FALSE	-0.28%	FALSE
88.75%	55.46%	64.05%	FALSE	-1.08%	FALSE
90.00%	53.23%	64.70%	FALSE	1.15%	FALSE
91.25%	**53.42%**	**63.57%**	**TRUE**	**0.96%**	**FALSE**
92.50%	47.86%	55.55%	FALSE	6.52%	TRUE
93.75%	0.00%	0.00%	FALSE	54.38%	TRUE

ช่องที่มีตัวอักษรหนา คือ แบบจำลองผลลัพธ์ที่ใช้โมดูลเกณฑ์การหยุดเลือก **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง



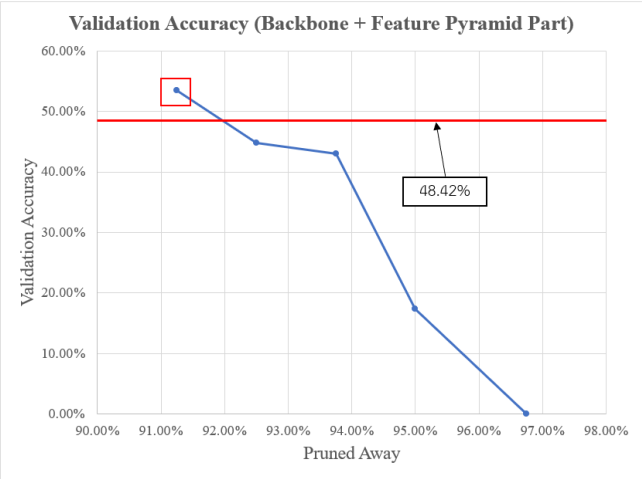
รูปที่ 29 กราฟผลลัพธ์การใช้เกณฑ์การหยุดในขั้นที่ 3 ของวิธีการ RPM (APoZ) (ตัดส่วนของโครงข่ายกระดูกสันหลัง) (เส้นสีแดงที่ 49.38% มีไว้เพื่อแสดงให้เห็นว่าจุดใดที่ความแม่นยำของแบบจำลองจะเริ่มลดลงต่ำกว่าค่าขีดแบ่ง โดยคำนวณได้จาก ความแม่นยำตรวจสอบเริ่มต้น 54.38% ลบด้วยค่าขีดแบ่ง 5.00%)

สำหรับส่วนสุดท้าย ซึ่งเป็นขั้นตอนที่สี่ของกลไกการตัดแบบทนทาน โดยส่วนนี้จะทำหน้าที่ตัดส่วนของโครงข่ายกระดูกสันหลัง กับโครงข่ายพีระมิดฟีเจอร์ร่วมกัน จากผลลัพธ์ในตารางที่ 15 พบว่าที่สัดส่วนการตัดที่ 92.50% และ 93.75% มีการลดลงของความแม่นยำกับชุดข้อมูลตรวจสอบต่ำกว่าค่าขีดแบ่งต่อเนื่องกัน 2 ครั้ง โมดูลเกณฑ์การหยุดจึงเลือกแบบจำลองที่มีสัดส่วนการตัด 91.25% เป็นแบบจำลองผลลัพธ์สำหรับขั้นตอนที่สี่ของกลไกการตัดแบบทนทาน

ตารางที่ 15 ผลลัพธ์การใช้เกณฑ์การหยุดในขั้นตอนการตัดแบบจำลองในขั้นที่ 4 ของกลไกการตัดแบบทนทาน โดย Stop คือ จุดที่เลือกหยุด และเป็นแบบจำลองที่เลือกเป็นผลลัพธ์ Difference คือ ส่วนต่างระหว่างความแม่นยำกับชุดข้อมูลตรวจสอบ (Validation) เริ่มต้นกับความแม่นยำกับชุดข้อมูลตรวจสอบแถวปัจจุบัน

Pruned away	Validation accuracy	Test accuracy	Stop	Difference	> Threshold
91.25%	**53.42%**	**63.57%**	**TRUE**	**0.00%**	**FALSE**
92.50%	44.86%	54.86%	FALSE	8.56%	TRUE
93.75%	43.03%	55.89%	FALSE	10.39%	TRUE
95.00%	17.40%	17.55%	FALSE	36.02%	TRUE
96.75%	0.00%	0.00%	FALSE	53.42%	TRUE

ช่องที่มีตัวอักษรหนา คือ แบบจำลองผลลัพธ์ที่ใช้โมดูลเกณฑ์การหยุดเลือก **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง



รูปที่ 30 กราฟผลลัพธ์การใช้เกณฑ์การหยุดในขั้นที่ 4 ของวิธีการ RPM (APoZ) (ตัดส่วนของโครงข่ายพีระมิดฟีเจอร์ และโครงข่ายกระดูกสันหลังร่วมกัน) (เส้นสีแดงที่ 48.42% มีไว้เพื่อแสดงให้เห็นว่าจุดใดที่ความแม่นยำของแบบจำลองจะเริ่มลดลงต่ำกว่าค่าขีดแบ่ง โดยคำนวณได้จาก ความแม่นยำตรวจสอบเริ่มต้น 53.42% ลบด้วยค่าขีดแบ่ง 5.00%)

6.6 การทดลองเพิ่มเติมอื่น ๆ

ในหัวข้อนี้จะรวบรวมการทดลองเพิ่มเติมซึ่งจัดเป็นการทดลองเสริม โดยจะประกอบด้วยสองการทดลองหลัก คือ 1) การฝึกสอนแบบสั้น และ 2) การตัดโครงข่ายพีระมิดฟีเจอร์

6.6.1 การฝึกสอนแบบสั้น

ในการทดลองนี้ จะทำการทดลองการตัดแบบจำลองด้วยวิธีการต่างๆ โดยจะมีความแตกต่างที่สำคัญคือในการตัดแต่ละครั้ง จะทำการเพิ่มอัตราการเรียนรู้ (Learning Rate) ให้มากที่สุดเท่าที่ทำได้ และลดจำนวนรอบการฝึกหลังการตัดแบบจำลองจาก 20 เหลือ 1 เนื่องจากว่าการฝึก 20 รอบนั้น ส่งผลให้แต่ละการทดลองมีการใช้เวลาสูงมาก จึงทำการทดลองลดรอบของการฝึกลง โดยมีสมมุติฐานว่า เมื่อตัดตัวแปรน้ำหนักจากแบบจำลองที่ผ่านการฝึกมาแล้ว ไม่น่าจะส่งผลกระทบต่อแบบจำลองมากนัก จึงไม่จำเป็นที่จะต้องฝึกสอนเป็นจำนวนมาก

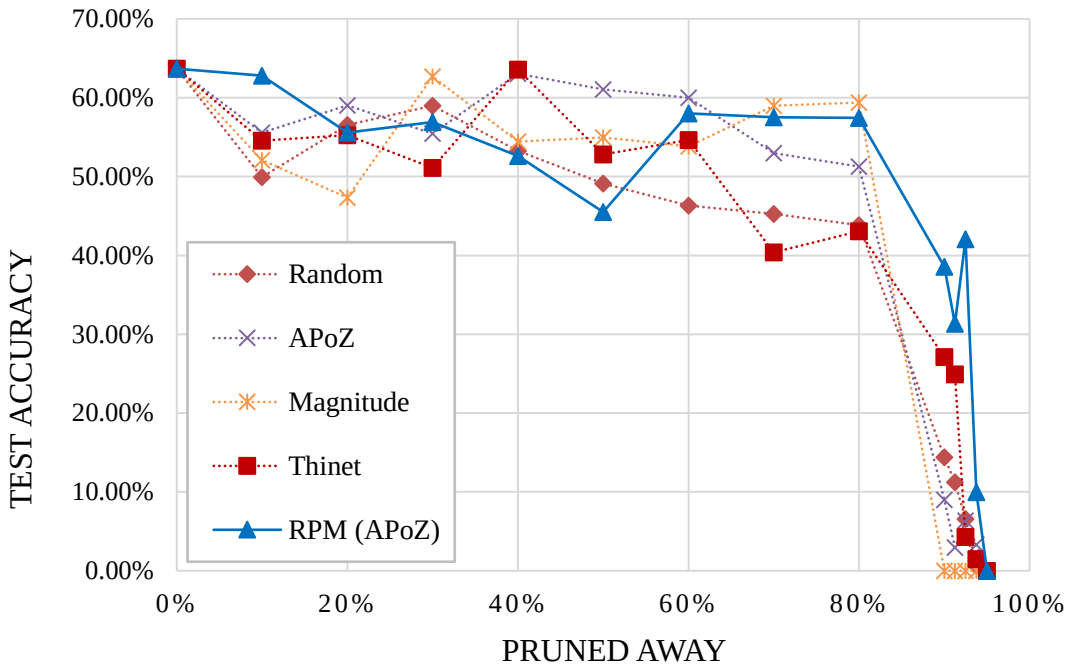
จากผลการทดลองในตารางที่ 16 พบว่าสำหรับสัดส่วนการตัดในช่วง 0% ถึง 80% วิธีการ RPM (APoZ) นั้นสามารถทำงานได้ใกล้เคียงกับวิธีการ APoZ และวิธีการ Magnitude แต่ว่าสำหรับสัดส่วนการตัดในช่วง 90% ขึ้นไปจะพบว่า วิธีการ RPM (APoZ) นั้นยังสามารถคงความแม่นยำได้ดีกว่าวิธีอื่นมาก

จากผลการทดลองพบว่าวิธีการ RPM (APoZ) นั้นสามารถทำงานได้ดีที่สุดในช่วงสัดส่วนการตัดที่ 90% ขึ้นไป และสาเหตุที่ผลวิธีการ RPM (APoZ) และวิธีการอื่น ๆ ที่มีความผันผวนมากกว่าในการทดลองปกติ นั้นมีสาเหตุมาจากการลดจำนวนรอบการฝึกลง ซึ่งส่งผลให้แบบจำลองไม่สามารถปรับตัวแปรน้ำหนักให้เข้าที่ได้ดีเพียงพอก่อนที่การฝึกแบบจำลองจะจบลง

ตารางที่ 16 ความแม่นยำกับชุดข้อมูลทดสอบของผลลัพธ์การตัดแบบการฝึกสอนแบบสั้นกับชุดข้อมูล UA-DETRAC

Pruned away	Random	APoZ	Magnitude	ThiNet	RPM (APoZ)
0.00%	63.68%	63.68%	63.68%	63.68%	63.68%
10.00%	49.91%	55.56%	52.10%	54.57%	**62.78%**
20.00%	56.55%	**59.01%**	47.32%	55.23%	55.56%
30.00%	58.97%	55.46%	**62.69%**	51.09%	56.90%
40.00%	53.26%	63.05%	54.43%	**63.57%**	52.61%
50.00%	49.11%	**61.07%**	54.95%	52.81%	45.50%
60.00%	46.33%	**60.00%**	53.82%	54.64%	57.99%
70.00%	45.26%	52.99%	**58.97%**	40.40%	57.52%
80.00%	43.82%	51.28%	**59.39%**	43.05%	57.43%
90.00%	14.41%	9.02%	0.00%	27.12%	**38.58%**
91.25%	11.25%	2.94%	0.00%	24.93%	**31.32%**
92.50%	6.55%	6.38%	0.00%	4.30%	**42.06%**
93.75%	0.64%	3.33%	0.00%	1.49%	**9.99%**
95.00%	0.00%	0.00%	0.00%	0.00%	0.00%

ตัวอักษรหนา คือ วิธีการที่ดีที่สุด **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง **พื้นหลังไม่มีสี** คือ ความแม่นยำของแบบจำลอง



รูปที่ 31 กราฟผลลัพธ์การตัดแบบเม็ดละเอียดสั้นกับชุดข้อมูล UA-DETRAC

6.6.2 การตัดโครงข่ายพีระมิดฟีเจอร์

ในการทดลองนี้จะทำการทดลองตัดแบบจำลองเฉพาะส่วนของโครงข่ายพีระมิดฟีเจอร์ด้วยวิธีการต่าง ๆ โดยมีจุดประสงค์เพื่อตรวจสอบว่าการตัดเฉพาะส่วนของโครงข่ายพีระมิดฟีเจอร์นั้นให้ผลดีกว่าหรือใกล้เคียงกับการตัดแบบโครงข่ายกระดูกสันหลังและโครงข่ายพีระมิดฟีเจอร์รวมกันหรือไม่ ในการทดลองจะประกอบไปด้วยวิธีการตัด 5 วิธีการ 1) Random 2) APoZ 3) Magnitude 4) ThiNet และ 5) RPM (APoZ) โดยวิธีการ 1 - 4 จะตัดเฉพาะส่วนของโครงข่ายพีระมิดฟีเจอร์ และวิธีการที่ 5 จะตัดแบบจำลองรวมทั้งสองส่วนตามวิธีการ RPM (APoZ)

จากผลการทดลองในตารางที่ 17 พบว่าวิธีการ APoZ เป็นวิธีการที่ดีที่สุดสำหรับกรณีการตัดเฉพาะส่วนของโครงข่ายพีระมิดฟีเจอร์เพียงอย่างเดียว โดยสามารถสังเกตได้จากที่สัดส่วนการตัดที่ 80% โดยวิธีการ APoZ สามารถคงความแม่นยำได้ที่ 66.62% ซึ่งมากกว่าวิธีการอื่น ๆ แต่ทว่าการตัดเฉพาะส่วนของโครงข่ายพีระมิดฟีเจอร์เพียงอย่างเดียวยังคงให้ผลลัพธ์ที่ด้อยกว่าการตัดแบบจำลองทั้งส่วนของโครงข่ายกระดูกสันหลังและโครงข่ายพีระมิดฟีเจอร์ร่วมกันค่อนข้างมาก ซึ่งสามารถคงความแม่นยำไว้ใกล้เคียงเดิมได้ถึงสัดส่วนการตัดที่ 91.25% โดยมีความแม่นยำอยู่ที่ 63.57%

จากการทดลองครั้งนี้พบว่า การตัดแบบจำลองเฉพาะส่วนของโครงข่ายพีระมิดฟีเจอร์นั้น ถึงแม้ว่าจะให้ประสิทธิภาพที่ดีในระดับหนึ่ง โดยสามารถคงความแม่นยำไว้ได้ถึงสัดส่วนการตัดที่ 80% แต่ทว่าก็ยังคงด้อยกว่าการตัดแบบจำลองแบบโครงข่ายกระดูกสันหลังและโครงข่ายพีระมิดฟีเจอร์ร่วมกัน ซึ่งสามารถคงความแม่นยำไว้ใกล้เคียงเดิมได้ถึงสัดส่วนการตัดแบบจำลองที่ 91.25%

ตารางที่ 17 ความแม่นยำกับชุดข้อมูลทดสอบของผลการตัดแบบจำลองในส่วนของโครงข่ายพีระมิดฟีเจอร์อย่างเดียวเทียบกับ RPM (APoZ) โดยสาเหตุความแม่นยำของ APoZ (FPN) กับ RPM (APoZ) (Backbone + FPN) จะมีค่าเท่ากันเนื่องจากทั้งสองวิธีการในช่วงสัดส่วนการตัดที่ 0% ถึง 80% มีกระบวนการทำงานที่เหมือนกันทุกประการ จึงสามารถใช้ผลลัพธ์ตัวเดียวกันได้

Pruned away	Random (FPN)	APoZ (FPN)	Magnitude (FPN)	ThiNet (FPN)	RPM (APoZ) (Backbone + FPN)
0.00%	63.68%	63.68%	63.68%	63.68%	63.68%
10.00%	61.14%	58.92%	60.73%	**61.57%**	58.92%
20.00%	57.15%	64.28%	**64.70%**	65.64%	64.28%
30.00%	66.06%	**66.15%**	65.66%	64.33%	**66.15%**
40.00%	61.47%	60.04%	**66.04%**	65.71%	60.04%
50.00%	58.81%	66.57%	**66.84%**	60.22%	66.57%
60.00%	61.26%	**66.22%**	62.71%	65.00%	**66.22%**
70.00%	58.27%	66.97%	67.30%	**67.44%**	66.97%
80.00%	53.67%	**66.62%**	52.83%	59.13%	**66.62%**
90.00%	-	-	-	-	**64.70%**
91.25%	-	-	-	-	**63.57%**
92.50%	-	-	-	-	**54.86%**
93.75%	-	-	-	-	**55.89%**
95.00%	-	-	-	-	**17.55%**

ตัวอักษรหนา คือ วิธีการที่ดีที่สุด **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง **พื้นหลังไม่มีสี** คือ ความแม่นยำของแบบจำลอง **ขีด** คือ ไม่สามารถตัดแบบจำลองไปมากกว่านี้ได้อีกแล้ว

6.6.3 การทดลองเพิ่มจำนวนรอบในการฝึก

เนื่องจากว่าในการทดลองหลักในหัวข้อที่ 6.2.1 และ 6.2.2 ยังคงมีประเด็นปัญหาที่สำคัญคือ ค่าความแม่นยำมีการแกว่งขึ้นลงในแต่ละสัดส่วนการตัดของแบบจำลอง ซึ่งมีความเป็นไปได้ว่า มีสาเหตุมาจากที่ในแต่ละรอบของการตัดแบบจำลองนั้นจะมีการฝึกแบบจำลองเพียง 20 รอบ เท่านั้น ส่งผลให้ตัวแบบจำลองที่ผ่านการตัดมาแล้ว ได้รับการฝึกที่ไม่เพียงพอ ด้วยเหตุนี้ ในหัวข้อนี้จึงทำการทดลองเพิ่มเติม เพื่อหาคำตอบของปัญหานี้ โดยการนำจุดที่มีความแม่นยำที่แกว่งผิดปกติจาก RPM (APoZ) ที่สัดส่วนการตัดที่ 10% กับชุดข้อมูล UA-DETRAC มาทำการเพิ่มจำนวนรอบในการฝึกเป็น 80 รอบ และวิเคราะห์ผลลัพธ์

จากผลการทดลองในตารางที่ 18 และตารางที่ 19 พบว่าเมื่อมีการเพิ่มรอบของการฝึกแบบจำลองจาก 20 รอบ เป็น 80 รอบ ส่งผลให้ค่าความแม่นยำของตัวแบบจำลองขึ้นมาสูงใกล้เคียงกับค่าดั้งเดิมของตัวแบบจำลองโดยเพิ่มจากเดิมที่ 58.92% เป็น 63.93% จึงสามารถสรุปได้ว่า ในการทดลองที่ 6.2.1 และ 6.2.2 นั้น มีรอบการฝึกแบบจำลองที่ไม่เพียงพอ ดังนั้นเพื่อให้ได้ค่าความแม่นยำในแต่ละสัดส่วนการตัดที่ดีขึ้น จึงควรมีการเพิ่มรอบการฝึกแบบจำลองให้มากขึ้น

ตารางที่ 18 ผลลัพธ์การตัดแบบจำลองกับชุดข้อมูล UA-DETRAC ด้วยวิธีการ RPM (APoZ) (ฝึก 20 รอบ)

	RPM (APoZ) – 20 Epoch	
Pruned away	Validation accuracy	Test accuracy
0.00%	51.14%	63.68%
10.00%	47.40%	58.92%

พื้นหลังสีเทา คือ สัดส่วนการตัดแบบจำลอง **พื้นหลังไม่มีสี** คือ ความแม่นยำของแบบจำลอง

ตารางที่ 19 ผลลัพธ์การตัดแบบจำลองกับชุดข้อมูล UA-DETRAC ด้วยวิธีการ RPM (APoZ) (ฝึก 80 รอบ)

	RPM (APoZ) - 80 Epoch	
Pruned away	Validation accuracy	Test accuracy
0.00%	51.14%	63.68%
10.00%	53.21%	63.93%

พื้นหลังสีเทา คือ สัดส่วนการตัดแบบจำลอง **พื้นหลังไม่มีสี** คือ ความแม่นยำของแบบจำลอง

6.6.4 การทดลองตัดบนชุดข้อมูล PASCAL VOC แบบเพิ่มรอบการฝึก

จากข้อสรุปในหัวข้อที่ 6.6.3 นั้นพบว่า การฝึกแบบจำลอง 20 รอบ ต่อการตัดแบบจำลองนั้นไม่เพียงพอ ประกอบกับผลการทดลองในหัวข้อที่ 6.2.2 นั้นมีปัญหาจากการฝึกแบบจำลองที่น้อยเกินไปอย่างชัดเจน ส่งผลให้ค่าความแม่นยำในแต่ละสัดส่วนการตัดมีการแกว่งขึ้นลง ในหัวข้อนี้จึงทำการทดลองเพิ่มเติม โดยทำการเพิ่มรอบการฝึกให้กับการทดลองที่ 6.2.2 ให้มากขึ้น โดยจะทำการทดลองเฉพาะช่วงสัดส่วนการตัดที่ 10% ถึง 40% เท่านั้น และวิเคราะห์ผลลัพธ์

จากผลการทดลองในตารางที่ 20 และตารางที่ 21 พบว่า เมื่อเพิ่มรอบการฝึกตัวแบบจำลอง ทำให้สามารถแก้ปัญหาการแกว่งของค่าความแม่นยำในแต่ละสัดส่วนการตัดได้ แต่ว่าเนื่องจากผลการทดลองในตารางที่ 21 ยังมีการตัดแบบจำลองที่น้อยเกินไป จึงทำให้ยังไม่สามารถสรุปผลการทดลองได้ว่าวิธีการไหนจะเป็นวิธีการที่ดีที่สุดเมื่อเพิ่มรอบการฝึกเป็น 40 รอบ ดังนั้นจึงจำเป็นต้องอ้างอิงจากผลการทดลองเดิมในตารางที่ 20 ซึ่งพบว่าวิธีการ APoZ นั้น เป็นวิธีการที่ดีที่สุด ซึ่งสามารถสังเกตได้จากสัดส่วนการตัดที่ 80% ที่สามารถคงความแม่นยำกับชุดข้อมูลทดสอบไว้ได้ถึง 52.02% ซึ่งมากกว่าวิธีการ RPM (APoZ) ที่สามารถคงความแม่นยำกับชุดข้อมูลทดสอบไว้ได้เพียง 50.73%

โดยสำหรับสาเหตุที่วิธีการ RPM (APoZ) แย่กว่าวิธีการ APoZ นั้นมีสาเหตุมาจากความแตกต่างกันของชุดข้อมูล UA-DETRAC กับ PASCAL VOC โดย 1) ชุดข้อมูล PASCAL VOC นั้นมีความยากกว่า UA-DETRAC เนื่องจากประกอบด้วยวัตถุหลากหลายชนิดมากกว่า และ 2) พื้นหลังของ PASCAL VOC นั้นมีความหลากหลาย ซึ่งแตกต่างกับ UA-DETRAC ที่เป็นพื้นถนนเท่านั้น ด้วยเหตุผลสองข้อนี้ จึงทำให้การตัดแบบจำลอง PASCAL VOC นั้นทำได้น้อย และยากกว่าแบบจำลอง UA-DETRAC และในขณะเดียวกัน แบบจำลอง PASCAL VOC ก็จำเป็นที่ต้องใช้การฝึกที่มากกว่า เพื่อให้ได้ผลลัพธ์ที่ดี

ตารางที่ 20 ผลการทดลองการตัดกับชุดข้อมูล PASCAL VOC (ฝึก 20 รอบ โดยเป็นผลลัพธ์จากหัวข้อ 6.2.2)

20 Epoch	APoZ		RPM (APoZ)	
Pruned Away	Val Acc	Test Acc	Val Acc	Test Acc
0.00%	44.71%	45.18%	44.71%	45.18%
10.00%	44.93%	45.39%	44.39%	**45.86%**
20.00%	38.82%	38.92%	46.71%	**46.96%**
30.00%	41.63%	**42.72%**	40.81%	41.67%
40.00%	48.15%	49.17%	49.98%	**49.20%**
50.00%	50.05%	49.82%	49.82%	**50.22%**
60.00%	50.47%	**51.10%**	49.15%	49.20%
70.00%	50.98%	**51.81%**	49.45%	50.23%
80.00%	50.77%	**52.02%**	50.78%	50.73%
90.00%	21.06%	21.17%	27.35%	**27.28%**
91.25%	18.83%	19.95%	21.68%	**21.61%**
92.50%	12.57%	12.57%	14.36%	**13.58%**
93.75%	2.37%	**2.40%**	1.80%	1.84%

ตัวอักษรหนา คือ วิธีการที่ดีที่สุด **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง **พื้นหลังไม่มีสี** คือ ความแม่นยำของแบบจำลอง

ตารางที่ 21 ผลการทดลองการตัดกับชุดข้อมูล PASCAL VOC (ฝึก 40 รอบ โดยเป็นผลลัพธ์จากการทำการทดลองใหม่)

40 Epoch	APoZ		RPM (APoZ)	
Pruned Away	Val Acc	Test Acc	Val Acc	Test Acc
0.00%	44.71%	45.18%	44.71%	45.18%
10.00%	47.86%	47.60%	47.44%	**48.30%**
20.00%	50.46%	**51.11%**	49.33%	49.86%
30.00%	49.81%	**50.67%**	49.16%	50.58%
40.00%	50.32%	51.24%	50.75%	**51.74%**

ตัวอักษรหนา คือ วิธีการที่ดีที่สุด **พื้นหลังสีเทา** คือ สัดส่วนการตัดแบบจำลอง **พื้นหลังไม่มีสี** คือ ความแม่นยำของแบบจำลอง

6.6.5 การทดลองตัดแบบครั้งเดียว

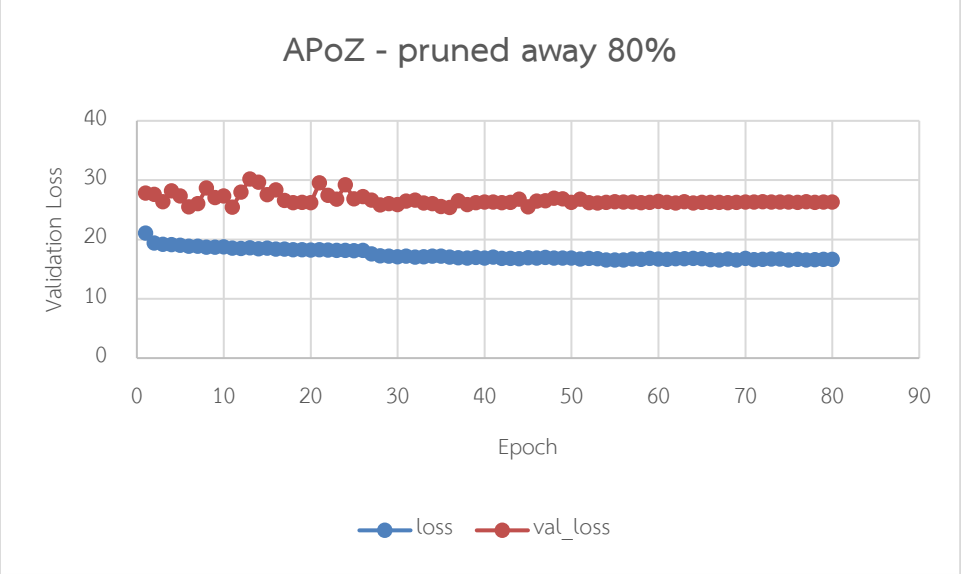
เนื่องจากในการทดลองในการวิจัยชิ้นนี้ มีการใช้งานการตัดเชิงวนซ้ำ (Iterative Pruning) เป็นหลัก โดยจะมีการตัดแบบจำลองทีละจำนวนน้อย สลับกับการฝึกแบบจำลอง ส่งผลให้ในแต่ละการทดลอง ใช้เวลาค่อนข้างมาก ในหัวข้อนี้จึงทำการทดลองเพิ่มขนาดที่ใช้ในการตัดแบบจำลองในแต่ละรอบ โดยจะทำการทดลองตัดแบบจำลองทีเดียวทันที 80% และ 90% ด้วยวิธีการ APoZ และ RPM (APoZ)

จากผลการทดลองในตารางที่ 22 พบว่าแบบจำลองยังสามารถทำงานได้เป็นปกติ โดยมีความแม่นยำกับชุดข้อมูลทดสอบอยู่ที่ 65.10% ซึ่งให้ผลลัพธ์ที่ใกล้เคียงกับการตัดเชิงวนซ้ำที่ได้ความแม่นยำกับชุดข้อมูลทดสอบอยู่ที่ 65.06% อ้างอิงจากตารางที่ 4 ในหัวข้อที่ 6.2.1

ตารางที่ 22 ผลการทดลองการตัดแบบจำลองทีเดียว 80% ด้วยวิธีการ APoZ

Pruned away	Epoch	Validation accuracy	Test accuracy
80.00%	80	55.54%	65.10%

พื้นหลังสีเทา คือ สัดส่วนการตัดแบบจำลอง



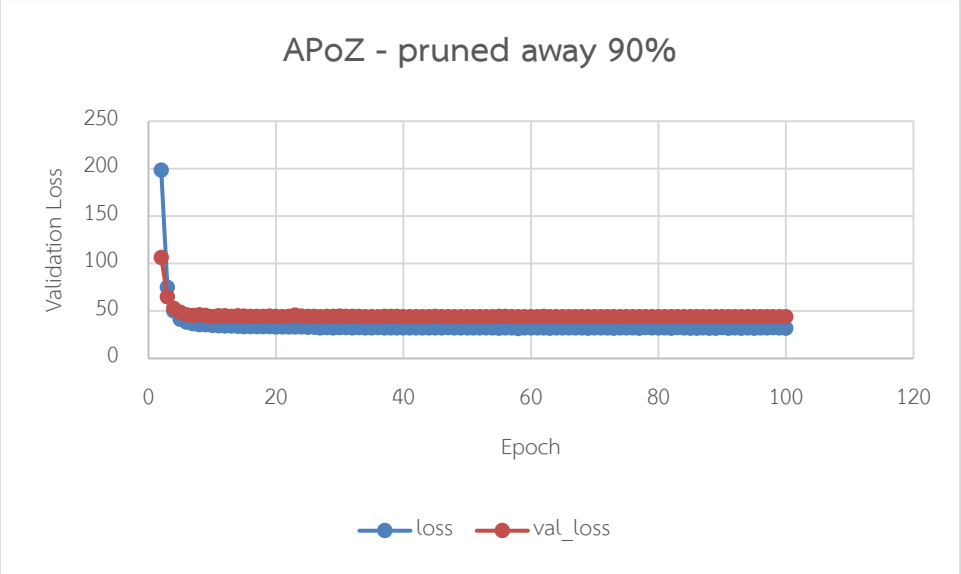
รูปที่ 32 กราฟค่าของฟังก์ชันต้นทุนของการตัดแบบจำลองทีเดียว 80% ด้วยวิธีการ APoZ

จากผลการทดลองในตารางที่ 23 พบว่าแบบจำลองไม่สามารถคงความแม่นยำไว้ได้ โดยมีความแม่นยำกับชุดข้อมูลทดสอบอยู่ที่ 6.70% ซึ่งให้ผลลัพธ์แย่กว่าการตัดเชิงวนซ้ำที่ได้ความแม่นยำกับชุดข้อมูลทดสอบอยู่ที่ 61.63% อ้างอิงจากตารางที่ 4 ในหัวข้อที่ 6.2.1

ตารางที่ 23 ผลการทดลองการตัดแบบจำลองทีเดียว 90% ด้วยวิธีการ APoZ

Pruned away	Epoch	Validation accuracy	Test accuracy
90.00%	100	5.98%	6.70%

พื้นหลังสีเทา คือ สัดส่วนการตัดแบบจำลอง



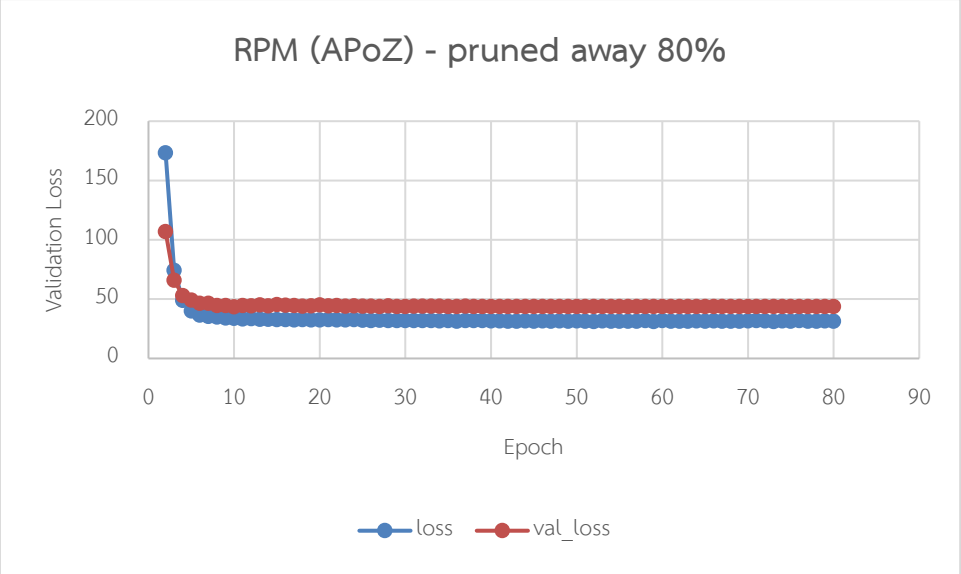
รูปที่ 33 กราฟค่าของฟังก์ชันต้นทุนของการตัดแบบจำลองทีเดียว 90% ด้วยวิธีการ APoZ

จากผลการทดลองในตารางที่ 24 พบว่าแบบจำลองไม่สามารถคงความแม่นยำไว้ได้ โดยมีความแม่นยำกับชุดข้อมูลทดสอบอยู่ที่ 6.96% ซึ่งให้ผลลัพธ์แย่กว่าการตัดเชิงวนซ้ำที่ได้ความแม่นยำกับชุดข้อมูลทดสอบอยู่ที่ 66.62% อ้างอิงจากตารางที่ 4 ในหัวข้อที่ 6.2.1

ตารางที่ 24 ผลการทดลองการตัดแบบจำลองทีเดียว 80% ด้วยวิธีการ RPM (APoZ)

Pruned away	Epoch	Validation accuracy	Test accuracy
80.00%	80	6.22%	6.96%

พื้นหลังสีเทา คือ สัดส่วนการตัดแบบจำลอง



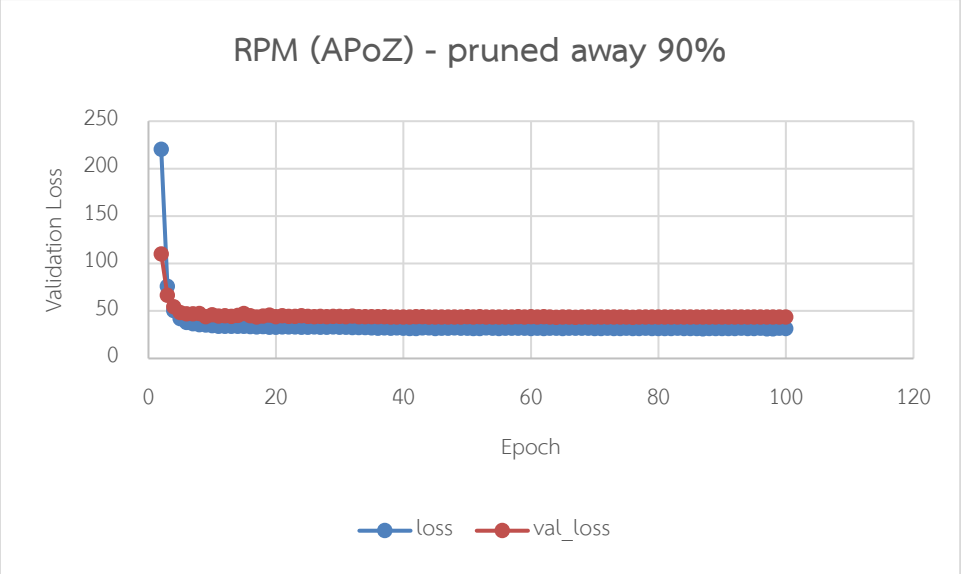
รูปที่ 34 กราฟค่าของฟังก์ชันต้นทุนของการตัดแบบจำลองทีเดียว 80% ด้วยวิธีการ RPM (APoZ)

จากผลการทดลองในตารางที่ 25 พบว่าแบบจำลองไม่สามารถคงความแม่นยำไว้ได้ โดยมีความแม่นยำกับชุดข้อมูลทดสอบอยู่ที่ 7.13% ซึ่งให้ผลลัพธ์แย่กว่าการตัดเชิงวนซ้ำที่ได้ความแม่นยำกับชุดข้อมูลทดสอบอยู่ที่ 64.70% อ้างอิงจากตารางที่ 4 ในหัวข้อที่ 6.2.1

ตารางที่ 25 ผลการทดลองการตัดแบบจำลองทีเดียว 90% ด้วยวิธีการ RPM (APoZ)

Pruned away	Epoch	Validation accuracy	Test accuracy
90.00%	100	6.35%	7.13%

พื้นหลังสีเทา คือ สัดส่วนการตัดแบบจำลอง



รูปที่ 35 กราฟค่าของฟังก์ชันต้นทุนของการตัดแบบจำลองทีเดียว 80% ด้วยวิธีการ RPM (APoZ)

จากผลการทดลองในหัวข้อนี้ทั้งหมดพบว่า การตัดแบบจำลองทีเดียวเป็นจำนวนมากนั้น ส่งผลให้ตัดแบบจำลองได้น้อยลงเมื่อเทียบกับการตัดแบบจำลองแบบเชิงวนซ้ำ ซึ่งสามารถสังเกตได้จาก วิธีการ APoZ ที่สามารถตัดได้เหลือเพียง 80% ในขณะที่สำหรับการตัดเชิงวนซ้ำสามารถตัดได้ถึง 90% และสำหรับวิธีการ RPM (APoZ) นั้นพบว่า ส่งผลให้ตัดได้น้อยกว่า 80% ในขณะที่สำหรับการตัดเชิงวนซ้ำสามารถตัดได้ถึง 91.25%

บทที่ 7
สรุปผลการวิจัยและแนวทางการวิจัยในขั้นถัดไป

7.1 สรุปผลการวิจัย

วิทยานิพนธ์ชิ้นนี้ ได้นำเสนอแนวคิดและแบบจำลองเพื่อใช้สำหรับการตรวจจับวัตถุบนอุปกรณ์ขนาดเล็กที่มีทรัพยากรจำกัด เช่น กล้องหรือคอมพิวเตอร์ขนาดเล็ก โดยงานวิจัยชิ้นนี้จะมุ่งเน้นไปที่การใช้กลยุทธ์การตัดเข้ากับแบบจำลองตรวจจับวัตถุ YOLOv3 โดยมีจุดประสงค์เพื่อลดการใช้ทรัพยากรของแบบจำลองการตรวจจับวัตถุเพื่อให้มีความเหมาะสมที่จะใช้งานในสภาพแวดล้อมที่ทรัพยากรจำกัด

วิธีการที่นำเสนอนั้นได้มีการนำแนวคิดมาจากความแตกต่างของลักษณะการแจกแจงของตัวแปรน้ำหนักที่อยู่ในแต่ละส่วนของแบบจำลอง จึงได้นำเสนอกลไกการตัดแบบทนทานซึ่งประกอบไปด้วยองค์กอบ 3 ส่วน คือ 1) การตัดแบบแยกส่วน 2) การจำกัดการตัด และ 3) เกณฑ์การหยุด โดยวิธีการที่นำเสนอมีจุดประสงค์เพื่อป้องกันการตัดส่วนใดส่วนหนึ่งของแบบจำลองมากเกินไปจากความแตกต่างของตัวแปรตัวที่อยู่ต่างชั้นกัน เพื่อให้แบบจำลองสามารถคงระดับความแม่นยำไว้ได้มากที่สุดและลดขนาดแบบจำลองลงให้ได้มากที่สุด

จากผลการทดลองโดยใช้ชุดข้อมูล UA-DETRAC ซึ่งแบบชุดข้อมูลตรวจจับรถยนต์บนถนน โดยจากการทดลองพบว่า RPM (APoZ) สามารถลดขนาดของแบบจำลองในขณะที่คงระดับความแม่นยำไว้ใกล้เคียงเดิมได้ดีที่สุด โดยสามารถลดจำนวนตัวแปรค่าน้ำหนักลงได้ถึง 91.25% และมีความแม่นยำอยู่ที่ 63.57% ซึ่งดีกว่าวิธีการอื่น ๆ แต่สำหรับชุดข้อมูล PASCAL VOC ซึ่งเป็นชุดข้อมูลตรวจจับวัตถุทั่วไป จากการทดลองพบว่า RPM (APoZ) นั้นด้อยกว่าวิธีการ APoZ ในสัดส่วนการตัดตั้งแต่ 80% ลงไป แต่ในทางกลับกัน RPM (APoZ) นั้นสามารถคงระดับความแม่นยำได้มากกว่าวิธีการ APoZ ในสัดส่วนการตัดที่ตั้งแต่ 90% ขึ้นไป แต่ก็ไม่สามารถคงระดับความแม่นยำเริ่มต้นเอาไว้ได้จึงไม่เหมาะสมสำหรับการนำไปใช้งาน สำหรับสาเหตุที่ RPM (APoZ) นั้นด้อยกว่า APoZ นั้นมีสาเหตุมาจากความแตกต่างกันของชุดข้อมูล UA-DETRAC และ PASCAL VOC

7.2 แนวทางการวิจัยถัดไป

1) โมดูลการจำกัดการตัดที่เสนอในงานวิจัยชิ้นนี้มีการใช้จำนวนตัวกรองสงวน (Reserved Filter) ที่เท่ากันในทุกชั้น (Layer) ของแบบจำลอง (Model) แต่ทว่าในความเป็นจริง จำนวนตัวกรองสงวนในแต่ละชั้นไม่จำเป็นต้องเท่ากันเนื่องจากว่าในแต่ละชั้นมีความต้องการใช้งานตัวกรองขั้นต่ำไม่เท่ากัน
2) ในการทดลอง มีความเป็นไปได้ว่าสามารถเพิ่มขนาดที่ใช้ในการตัดแบบจำลองในแต่ละรอบเพื่อลดระยะเวลาที่ใช้ในการทดลองในแต่ละรอบลง

บรรณานุกรม

[1] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part-based models,” *IEEE transactions on pattern analysis and machine intelligence,* vol. 32, no. 9, pp. 1627-1645, 2009.

[2] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International journal of computer vision,* vol. 60, no. 2, pp. 91-110, 2004.

[3] N. Dalal, and B. Triggs, "Histograms of oriented gradients for human detection." pp. 886-893.

[4] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, “Image classification with the fisher vector: Theory and practice,” *International journal of computer vision,* vol. 105, no. 3, pp. 222-245, 2013.

[5] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation." pp. 580-587.

[6] R. Girshick, "Fast r-cnn." pp. 1440-1448.

[7] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks." pp. 91-99.

[8] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection." pp. 779-788.

[9] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector." pp. 21-37.

[10] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection." pp. 2980-2988.

[11] Y. LeCun, J. S. Denker, and S. A. Solla, "Optimal brain damage." pp. 598-605.

[12] S. Anwar, K. Hwang, and W. Sung, “Structured pruning of deep convolutional neural networks,” *ACM Journal on Emerging Technologies in Computing Systems (JETC),* vol. 13, no. 3, pp. 32, 2017.

[13] J. Redmon, and A. Farhadi, “YOLO9000: better, faster, stronger,” *arXiv preprint*, 2017.

[14] J. Redmon, and A. Farhadi, “Yolov3: An incremental improvement,” *arXiv*

preprint arXiv:1804.02767, 2018.

[15] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, “A survey of model compression and acceleration for deep neural networks,” *arXiv preprint arXiv:1710.09282*, 2017.

[16] B. Hassibi, D. G. Stork, and G. J. Wolff, "Optimal brain surgeon and general network pruning." pp. 293-299.

[17] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. P. Graf, “Pruning filters for efficient convnets,” *arXiv preprint arXiv:1608.08710*, 2016.

[18] H. Hu, R. Peng, Y.-W. Tai, and C.-K. Tang, “Network trimming: A data-driven neuron pruning approach towards efficient deep architectures,” *arXiv preprint arXiv:1607.03250*, 2016.

[19] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural network." pp. 1135-1143.

[20] P. Singh, R. Manikandan, N. Matiyali, and V. Namboodiri, "Multi-layer pruning framework for compressing single shot multibox detector." pp. 1318-1327.

[21] J.-H. Luo, J. Wu, and W. Lin, "Thinet: A filter level pruning method for deep neural network compression." pp. 5058-5066.

[22] L. Wen, D. Du, Z. Cai, Z. Lei, M.-C. Chang, H. Qi, J. Lim, M.-H. Yang, and S. Lyu, “UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking,” *arXiv preprint arXiv:1511.04136*, 2015.

[23] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes (voc) challenge,” *International journal of computer vision,* vol. 88, no. 2, pp. 303-338, 2010.

ประวัติผู้เขียน

ชื่อ-สกุล	ณัฐนนท์ กฤตยานวัช
วัน เดือน ปี เกิด	14 ธันวาคม 2535
สถานที่เกิด	กรุงเทพมหานคร
วุฒิการศึกษา	บธ.บ. (เกียรตินิยมอันดับสอง) การเงิน และระบบสารสนเทศเพื่อการจัดการ มหาวิทยาลัยธรรมศาสตร์ (พ.ศ. 2554 - 2557)
ที่อยู่ปัจจุบัน	20/15 หมู่ 17 พหลโยธิน 60 พหลโยธิน คูคต ลำลูกกา ปทุมธานี 12130
ผลงานตีพิมพ์	N. Krittayanawach and P. Vateekul, "Robust Compression Technique for YOLOv3 on Real-Time Vehicle Detection," 2019 11th International Conference on Information Technology and Electrical Engineering (ICITEE)