



รายงาน

CarClassification

จัดทำโดย

กลุ่ม Textetion และสมาชิก

นายพลภัทร	สร้อยเสริมทรัพย์	6410450184
นายศรุต	คำยีน	6410450273
นายธนภัทร	บำรุงไทยวรกุล	6410451016

เสนอ

อ.ชาคริต วัชโรภาส

รายงานนี้เป็นส่วนหนึ่งของรายงานวิชา 01418364  
มหาวิทยาลัยเกษตรศาสตร์

ภาคปลาย ปีการศึกษา 2566

## คำนำ

รายงานการสร้างโมเดลในหัวข้อแยกแยะประเภทของรถยนต์ตามจำนวนล้อของรถยนต์ เพื่อนำไปใช้กับด่านทางด่วนเพื่อจัดเก็บค่าธรรมเนียมการใช้บริการตามประเภทของรถ การแยกแยะประเภทที่ผิดพลาดอาจนำไปสู่ปัญหาต่างๆ เช่น การสูญเสียรายได้ การจราจรติดขัด และความไม่พึงพอใจของผู้ใช้

โครงการนี้มุ่งเป้าไปที่การพัฒนาโมเดล Deep Learning สำหรับการแยกแยะประเภทรถยนต์ที่เข้ามาใช้งานทางด่วน โมเดลนี้จะถูกสร้างขึ้นโดยใช้สถาปัตยกรรม Convolutional Neural Network (CNN) และสามารถแยกแยะประเภทรถยนต์ตามจำนวนล้อ ดังนี้

- 4 ล้อ: รถยนต์ส่วนบุคคล รถกระบะ
- 6 ล้อ: รถบรรทุกขนาดเล็ก
- มากกว่า 6 ล้อ: รถบรรทุกขนาดใหญ่ รถพ่วง

### คณะผู้จัดทำ

นาย พลภัทร	สร้อยเสริมทรัพย์
นาย ศรุต	คำยีน
นาย ธนภัทร	บำรุงไทยวรกุล

## สารบัญ

เรื่อง	หน้า
คำนำ	2
สารบัญ	3
บทที่ 1 บทนำ	4
บทที่ 2 งานที่เกี่ยวข้อง	5
บทที่ 3 วิธีการ	7
บทที่ 4 ผลการทดลอง	9
บทที่ 5 สรุป	10
บรรณานุกรม	11

# บทที่ 1

## บทนำ

### 1. ความสำคัญของการแยกแยะประเภทรถยนต์บนทางด่วน

ระบบจัดการทางด่วนในปัจจุบันจำเป็นต้องมีการแยกแยะประเภทรถยนต์อย่างแม่นยำ เพื่อจัดเก็บค่าธรรมเนียมการใช้บริการตามประเภทของรถ การแยกแยะประเภทที่ผิดพลาดอาจนำไปสู่ปัญหาต่างๆ เช่น การสูญเสียรายได้จากการจัดเก็บค่าธรรมเนียมที่ไม่ถูกต้อง ส่งผลต่อรายได้ของระบบทางด่วน จึงเกิดความคิดที่จะต้องการนำ เทคโนโลยี Deep Learning ซึ่งมีศักยภาพในการแก้ปัญหาเหล่านี้ได้ โดยใช้โมเดล Deep Learning ที่สามารถแยกแยะประเภทรถยนต์ตามจำนวนล้อได้อย่างแม่นยำ เพื่อจะช่วยส่งผลดีต่อระบบจัดการทางด่วน

### 2. วัตถุประสงค์

โครงการนี้มุ่งเป้าไปที่การพัฒนาโมเดล Deep Learning สำหรับการแยกแยะประเภทรถยนต์บนทางด่วน โมเดลนี้จะถูกสร้างขึ้นโดยใช้สถาปัตยกรรม Convolutional Neural Network (CNN) และสามารถแยกแยะประเภทรถยนต์ตามจำนวนล้อ โดยคาดหวังผลที่จะพัฒนาโมเดล Deep Learning ที่สามารถแยกแยะประเภทรถยนต์ตามจำนวนล้อแต่ละประเภท ได้แก่ 4 ล้อ 6 ล้อ และมากกว่า 6 ล้อ และการทดสอบประสิทธิภาพ และวิเคราะห์ผลลัพธ์ ประเมินความเป็นไปได้ของโมเดล Deep Learning ที่พัฒนาขึ้นว่าสามารถนำโมเดลไปใช้จริงได้หรือไม่

### 3. ผลลัพธ์ที่คาดหวัง

คาดว่าโมเดล Deep Learning ที่พัฒนาขึ้นนี้ จะสามารถแยกแยะประเภทรถยนต์ได้ และจะมีประสิทธิภาพมากพอที่จะนำไปใช้บนทางด่วนได้ เพื่อแยกแยะประเภทของรถได้อย่างแม่นยำ และสามารถนำไปประยุกต์ใช้กับระบบต่างๆ ในการจัดการทางด่วนในอนาคต

## บทที่ 2 งานที่เกี่ยวข้อง

โครงการนี้เกี่ยวข้องมีส่วนเกี่ยวข้องกับงานวิจัยดังนี้

- Convolutional Neural Networks (CNNs)
- CNN Transfer Learning for Image Classification
- ResNet50

### Convolutional Neural Networks (CNNs)

CNNs เป็นประเภทของ Deep Neural Networks ที่ถูกออกแบบมาสำหรับงานวิทัศน์คอมพิวเตอร์ โดยเฉพาะงาน image classification CNNs ทำงานโดยใช้ convolutional layers เพื่อเรียนรู้ features จาก images และใช้ fully connected layers เพื่อทำการ classification

คำสำคัญ:

- Convolutional Neural Networks (CNNs) - เครือข่ายประสาทเทียมเชิงลึกแบบ convolutional
- convolutional layers - ชั้น convolutional
- fully connected layers - ชั้นเชื่อมต่อแบบเต็ม
- image classification - การจำแนกประเภทภาพ

### **CNN Transfer Learning for Image Classification**

CNN Transfer Learning เป็นเทคนิคที่ใช้โมเดล CNN ที่ถูก trained บน dataset หนึ่ง มาปรับใช้กับงาน image classification อื่น โดยไม่ต้อง train โมเดลใหม่ทั้งหมด

คำสำคัญ:

- CNN Transfer Learning - การถ่ายโอนการเรียนรู้แบบ CNN
- Image Classification - การจำแนกประเภทภาพ
- fine-tuning - การปรับแต่ง

### **ResNet50**

ResNet50 เป็น CNN architecture ที่ถูกพัฒนาโดย He et al. (2016) ResNet50 ประกอบไปด้วย convolutional layers 50 layers และใช้ residual connections เพื่อช่วยให้โมเดลเรียนรู้ได้ลึกขึ้น

คำสำคัญ:

- ResNet50 - ResNet ห้าสิบ
- residual connections - การเชื่อมต่อแบบเหลือ
- image classification - การจำแนกประเภทภาพ

ข้อมูล

### บทที่ 3 วิธีการ

ชุดข้อมูลที่ใช้ในการสร้างโมเดล Deep Learning ของโครงงานนี้ได้นำมาจากเว็บไซต์ Kaggle จำนวน 5 ชุดข้อมูลดังนี้

- [vehicle-wheel-detection](#)
- [trucks-detection](#)
- [truck-licenseplate-dataset](#)
- [vehicles-image-dataset](#)
- [car-and-truck](#)
- [uk-truck-brands-dataset](#)

รวมไปถึงการเก็บรูปภาพบนอินเทอร์เน็ต

และรูปภาพที่ได้รวบรวมมาเองโดยรูปภาพเพิ่มเติมทำให้ได้รูปที่นำมาใช้เทรนโมเดลมีจำนวน 1724 รูปภาพ โดยรูปภาพรถยนต์นั้นจะเป็นตัวแปรอิสระ(features)

และประเภทจำนวนล้อจะเป็นตัวแปรตาม(labels) โดยจะแทนเลข 0, 1, 2

เป็นเลขของรถยนต์ประเภท 4, 6 และ มากกว่า 6 ล้อ ตามลำดับ

รูปภาพที่นำมาใช้นั้นเป็นรูปภาพรถยนต์ที่อยู่แนวเฉียงข้าง ให้เห็นล้อหน้าและหลังทั้งหมด เนื่องจากจำเป็นต้องการระบุจำนวนล้อให้กับโมเดล และคัดกรองรูปภาพที่ใช้งานไม่ได้ออกไป โดยจำนวนรูปภาพทั้งหมดสามารถแบ่งตามจำนวนล้อ ดังนี้

- 4 ล้อ: 617 รูป
- 6 ล้อ: 564 รูป
- มากกว่า 6 ล้อ: 543 รูป

จากข้อมูลทั้งหมด ได้ถูกแบ่งออกเป็น 2 เซตในการเทรนโมเดลดังนี้:

- Training set: 70% ของข้อมูลทั้งหมด (1206 รูป)
- Test set: 30% ของข้อมูลทั้งหมด (518 รูป)



## อุปกรณ์และเครื่องมือ

### anaconda

เป็นโปรแกรมที่ใช้สร้างโมเดลแยกแยะรถยนต์ในโครงการครั้งนี้ด้วยภาษาโปรแกรม python พร้อมใช้งานกับ module และ library ดังนี้

- **jupyter notebook:** เครื่องมือสำหรับเขียนโค้ดและแสดงผลลัพธ์  
เหมาะสำหรับการวิเคราะห์ข้อมูลและสร้างโมเดล Machine Learning
- **tensorflow:** ไลบรารีโอเพนซอร์สสำหรับ Deep Learning มีประสิทธิภาพสูง  
รองรับงานหลากหลาย
- **cv2:** ไลบรารีโอเพนซอร์สสำหรับ Computer Vision รองรับงาน Computer Vision ทั่วไป เช่น การอ่านภาพ การแปลงภาพ การตรวจจับวัตถุ
- **h5py:** ไลบรารีโอเพนซอร์สสำหรับอ่านและเขียนไฟล์ HDF5  
รองรับการจัดเก็บข้อมูลขนาดใหญ่
- **numpy:** ไลบรารีโอเพนซอร์สสำหรับการคำนวณทางวิทยาศาสตร์  
รองรับการจัดการและวิเคราะห์ข้อมูล
- **keras:** ไลบรารีโอเพนซอร์สสำหรับ Deep Learning รองรับงาน Deep Learning ทั่วไป พัฒนากับ TensorFlow
- **sklearn:** ไลบรารีโอเพนซอร์สสำหรับ Machine Learning รองรับงาน Machine Learning ทั่วไป
- **matplotlib:** ไลบรารีโอเพนซอร์สสำหรับสร้างกราฟ ใช้งานง่าย  
รองรับการสร้างกราฟประเภทต่างๆ

## โมเดล

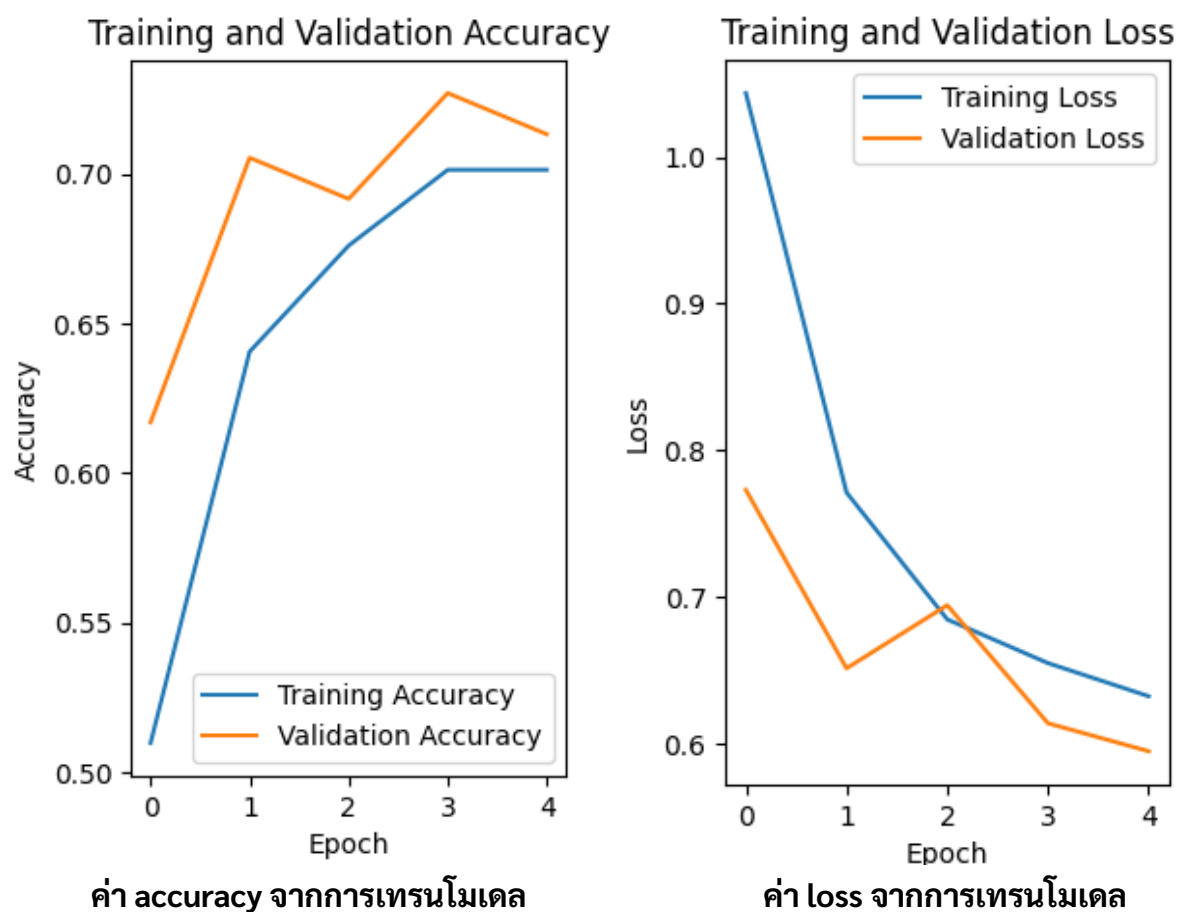
โมเดล Deep Learning ที่ใช้ในโครงการนี้คือ Convolutional Neural Network (CNN) เนื่องจากมีประสิทธิภาพสูงในการจำแนกภาพ เรียนรู้ features ทำงานกับข้อมูลขนาดใหญ่ ปรับแต่งได้ และ โมเดลที่ได้นำมาใช้งานในการทำโครงการครั้งนี้ ได้ใช้การ transfer learning จากโมเดล ResNet50 มาเป็น pre-trained model เพื่อเป็นโมเดลตั้งต้นในการสร้างโมเดลของโครงการ

## บทที่ 4

### ผลการทดลอง

#### การประเมินผล

ประสิทธิภาพของโมเดลถูกประเมินค่า แม่นยำ(accuracy) และ ค่าสูญเสีย(loss) ของผลลัพธ์ของโมเดลที่ได้จากการเทรนและ การนำไปใช้ในการทำนายข้อมูลที่นำเข้ามา โดยค่า accuracy ที่ได้คาดไว้ในช่วง 0.8 หรือร้อยละ 80



โดยผลลัพธ์ accuracy ที่ได้หลังการเทรนเสร็จสิ้นมีค่าอยู่ที่ 0.7132 หรือคิดอยู่ที่ร้อยละ 71 และเมื่อนำโมเดลไปใช้กับข้อมูลทดลองที่นำเข้าเพื่อการทำนาย ด้วยจำนวนรูป 30

รูปโดยแบ่งเป็นประเภทรถยนต์แต่ละประเภทละ 10 รูปภาพ โดยผลลัพธ์ที่ได้ออกมาอยู่ที่ 0.7667 หรือคิดอยู่ที่ร้อยละ 76 ซึ่งเป็นผลลัพธ์ที่ยังไม่ถึงตามที่คาดไว้ที่ร้อยละ 80

ผลลัพธ์ของการทำนายมีลักษณะดังนี้ [ค่าที่ถูกต้อง:ค่าที่ผิด]

[0 0 2 0 0 1 1 1 0 1 1 2 1 1 1 1 1 1 1 2 0 2 2 2 0 2 2 2 2]  
 4 ล้อ[5:5]      6 ล้อ[9:1]      6+ล้อ[8:2]

## บทที่ 5

### สรุป

#### ข้อจำกัดและอุปสรรค:

##### ข้อมูล(Dataset)

**ความหลากหลายของข้อมูล:** รูปภาพของรถต่างๆที่มีหลายรูปแบบ มุมมอง แสง สภาพอากาศ และความละเอียดที่หลากหลาย

##### ความซับซ้อนของรูปและฉาก:

ในรูปภาพอาจมีวัตถุอื่นๆรวมอยู่ด้วยทำให้รบกวนการแยกรถตามจำนวนล้อ  
 ไม่ว่าจะเป็นเงาที่ส่งผลต่อการมองเห็นจำนวนล้อในรูป หรือจะเป็นวัตถุต่างๆ

**ขนาดของชุดข้อมูล:** เนื่องจากเป็นโมเดล classification

ทำให้การมีชุดข้อมูลขนาดใหญ่เพื่อเทรนโมเดล Deep Learning ให้มีประสิทธิภาพได้มาก  
 แต่การหารูปเพื่อนำมาทำชุดข้อมูลนั้นหาได้ยากเพราะจำเป็นต้องหารูปให้ตรงกับความต้องการของโปรเจค

##### โมเดล

**ความซับซ้อนของโมเดล:** โมเดล Deep Learning

ที่มีความซับซ้อนสูงอาจต้องการทรัพยากรการคำนวณมาก

**การปรับแต่งโมเดล:** โมเดล Deep Learning อาจต้องปรับแต่งให้เหมาะกับชุดข้อมูลเฉพาะ

**การอธิบายผลลัพธ์:** โมเดล Deep Learning อาจไม่สามารถอธิบายผลลัพธ์ได้ชัดเจน

## การใช้งาน

ความเร็วในการประมวลผล: โมเดล Deep Learning

อาจทำงานช้าบนอุปกรณ์ที่มีทรัพยากรจำกัด

ความถูกต้อง: โมเดล Deep Learning อาจไม่แม่นยำ 100%

## บรรณานุกรม

### 1. Kaggle Datasets:

- Vehicle Wheel Detection: Dataclusterlabs. (2023, December 1). Vehicle Wheel Detection. Kaggle. <https://www.kaggle.com/datasets/dataclusterlabs/vehicle-wheel-detection>
- Trucks Detection: Beethoo. (2023, November 11). Trucks Detection. Kaggle. <https://www.kaggle.com/datasets/beethoo/trucks-detection>
- Truck License Plate Dataset: Amritesh Tiwari. (2023, October 14). Truck License Plate Dataset. Kaggle. <https://www.kaggle.com/datasets/amriteshtiwari20/truck-licenseplate-dataset>
- Vehicles Image Dataset: Md Moheminul Islam. (2023, September 24). Vehicles Image Dataset. Kaggle. <https://www.kaggle.com/datasets/mmohaiminulislam/vehicles-image-dataset>
- Car and Truck: Enes UMCU. (2023, August 23). Car and Truck. Kaggle. <https://www.kaggle.com/datasets/enesumcu/car-and-truck>

- UK Truck Brands Dataset: Big Nose the Third. (2023, July 12). UK Truck Brands Dataset. Kaggle. <https://www.kaggle.com/datasets/bignosethethird/uk-truck-brands-dataset>

## 2. arXiv:

- A Gentle Introduction to Convolutional Neural Networks: Zhang, Y., & Yeung, D. Y. (2016). A Gentle Introduction to Convolutional Neural Networks. arXiv preprint arXiv:1609.07100. <https://arxiv.org/abs/1609.07100>
- Transfer Learning for Image Classification: Pan, S. J., & Yang, Q. (2018). Transfer Learning for Image Classification. arXiv preprint arXiv:1807.06376. <https://arxiv.org/abs/1807.06376>
- ResNet50: He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. arXiv preprint arXiv:1512.03385. <https://arxiv.org/abs/1512.03385>