Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine
การพัฒนาแอปพลิเคชันบนเว็บเพื่อการตรวจจับและจดจำป้ายจราจรอัตโนมัติโดยใช้เครื่องมือ Teachable Machine

Thunthanapon Pirom[1], Pawarisa Hamniam[2], Warattapop Thapatsuwan[3], and Peeraya Thapatsuwan[4*]
ธัญธนภรณ์ ภิรมย์[1], ปวริศา เหมเนียม[2], วรัทภพ ธภัทรสุวรรณ[3], และ พีรญา ธภัทรสุวรรณ[4*]

Received: 15 March 2025;
Revised: 6 June 2025;
Accepted: 13 June 2025;
Published: 15 August 2025;

Abstract

This research aims to develop a model for automatic traffic sign classification using Teachable Machine, a machine learning platform that requires no programming knowledge. The model employs Convolutional Neural Networks, specifically MobileNet architecture, trained using the Adam Optimizer on TensorFlow.js, enabling client-side deployment via a web application. The dataset used consists of 17,440 traffic sign images across 27 classes, compiled from the standard GTSRB dataset and additional photographs taken in Thailand. Data Augmentation techniques, including image rotation, lighting adjustment, noise injection, and perspective transformation, were applied to enhance data diversity and reduce overfitting risk. The model was trained with the following parameters: batch size of 16, 50 training epochs, and a learning rate of 0.001. Model performance was evaluated using standard metrics: Accuracy, Precision, Recall, and F1-Measure, calculated from a 27 × 27 Confusion Matrix. The learning behavior was further analyzed through Accuracy per Epoch and Loss per Epoch graphs. Experimental results indicate that the model achieved high performance, with an average accuracy of 98.93% and an F1-Measure of 98.90%, demonstrating the potential of Teachable Machine in developing accurate models applicable to both educational contexts and intelligent traffic systems.

Keywords: Traffic Sign Recognition, Teachable Machine, Convolutional Neural Networks, Data Augmentation, Web-based Image Classification

[1] Student, Program in Computer Science, Faculty of Liberal Arts and Science, Kasetsart University, Kamphaeng Saen Campus, Nakhon Pathom 73140, Thailand; นักศึกษา สาขาวิชาวิทยาการคอมพิวเตอร์ คณะศิลปศาสตร์และวิทยาศาสตร์ มหาวิทยาลัยเกษตรศาสตร์ จังหวัดนครปฐม 73140 ประเทศไทย; Email: thunthanapon.p@ku.th

[2] Student, Program in Computer Science, Faculty of Liberal Arts and Science, Kasetsart University, Kamphaeng Saen Campus, Nakhon Pathom 73140, Thailand; นักศึกษา สาขาวิชาวิทยาการคอมพิวเตอร์ คณะศิลปศาสตร์และวิทยาศาสตร์ มหาวิทยาลัยเกษตรศาสตร์ จังหวัดนครปฐม 73140 ประเทศไทย; Email: pawarisa.ha@ku.th

[3] Lecturer, Dr., Program in Computer Science, Faculty of Liberal Arts and Science, Kasetsart University, Kamphaeng Saen Campus, Nakhon Pathom 73140, Thailand; อาจารย์ ดร., สาขาวิชาวิทยาการคอมพิวเตอร์ คณะศิลปศาสตร์และวิทยาศาสตร์ มหาวิทยาลัยเกษตรศาสตร์ จังหวัดนครปฐม 73140 ประเทศไทย; Email: warattapop.t@ku.th

[4,*] Lecturer, Dr., Program in Computer Science, Faculty of Liberal Arts and Science, Kasetsart University, Kamphaeng Saen Campus, Nakhon Pathom 73140, Thailand; อาจารย์ ดร., สาขาวิชาวิทยาการคอมพิวเตอร์ คณะศิลปศาสตร์และวิทยาศาสตร์ มหาวิทยาลัยเกษตรศาสตร์ จังหวัดนครปฐม 73140 ประเทศไทย; Email: peeraya.t@ku.th

*Corresponding authors: Peeraya Thapatsuwan (peeraya.t@ku.th)

Citation: Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาโมเดลสำหรับจำแนกประเภทของป้ายจราจรอัตโนมัติ โดยใช้เครื่องมือ Teachable Machine ซึ่งเป็นแพลตฟอร์มการเรียนรู้ของเครื่องที่ไม่ต้องอาศัยความรู้ด้านการเขียนโปรแกรม โดยโมเดลที่พัฒนาอาศัยโครงข่ายประสาทเทียมแบบคอนโวลูชัน โดยเฉพาะสถาปัตยกรรม MobileNet ซึ่งฝึกด้วยอัลกอริทึม Adam Optimizer บนพื้นฐานของ TensorFlow.js สำหรับการใช้งานฝั่งผู้ใช้ผ่านเว็บแอปพลิเคชัน ทั้งนี้ ชุดข้อมูลที่ใช้ประกอบด้วยภาพป้ายจราจรรวม 17,440 ภาพ จาก 27 คลาส ซึ่งรวบรวมจากชุดข้อมูลมาตรฐาน GTSRB และภาพถ่ายจากประเทศไทย โดยได้มีการประยุกต์ใช้เทคนิค Data Augmentation ได้แก่ การหมุนภาพ การปรับแสง การเพิ่มสัญญาณรบกวน และการแปลงมุมมอง เพื่อเพิ่มความหลากหลายของข้อมูลและลดความเสี่ยงจาก Overfitting โมเดลถูกฝึกด้วยพารามิเตอร์พื้นฐาน ได้แก่ ขนาดกลุ่มเท่ากับ 16 จำนวนรอบการฝึก เท่ากับ 50 และค่า Learning Rate เท่ากับ 0.001 การประเมินประสิทธิภาพของโมเดลดำเนินการผ่านตัวชี้วัดประสิทธิภาพมาตรฐาน ได้แก่ Accuracy, Precision, Recall และ F1-Measure ซึ่งคำนวณจาก Confusion Matrix ขนาด 27 × 27 และวิเคราะห์พฤติกรรมการเรียนรู้ด้วยกราฟ Accuracy per Epoch และ Loss per Epoch ผลการทดลองแสดงให้เห็นว่าโมเดลสามารถจำแนกป้ายจราจรได้อย่างมีประสิทธิภาพ โดยมีค่าเฉลี่ยของ Accuracy เท่ากับ 98.93% และ F1-Measure เท่ากับ 98.90% สะท้อนถึงศักยภาพของ Teachable Machine ในการสร้างโมเดลที่แม่นยำและสามารถนำไปประยุกต์ใช้จริงได้ ทั้งในบริบทของการศึกษาและระบบจราจรอัจฉริยะ

คำสำคัญ: การรู้จำป้ายจราจร, เครื่องมือ Teachable Machine, โครงข่ายประสาทเทียมแบบคอนโวลูชัน, การเพิ่มข้อมูลเทียม, การจำแนกรูปภาพบนเว็บแอปพลิเคชัน

1. บทนำ (Introduction)

ในยุคที่ยานยนต์ไร้คนขับ (Autonomous Vehicles) และระบบช่วยเหลือผู้ขับขี่ขั้นสูง (Advanced Driver Assistance Systems: ADAS) กำลังกลายเป็นเทคโนโลยีหลักในภาคการขนส่ง ความสามารถในการรับรู้และตีความป้ายจราจรอย่างถูกต้องจึงเป็นองค์ประกอบสำคัญในการรักษาความปลอดภัยบนท้องถนน ป้ายจราจรทำหน้าที่สื่อสารคำสั่ง คำเตือน และข้อมูลสำคัญแก่ผู้ขับขี่ โดยเฉพาะในสถานการณ์ที่ซับซ้อนหรือทัศนวิสัยต่ำ เช่น ป้ายที่ถูกบดบัง แสงจ้าหรือมุมมองที่ผิดเพี้ยน ล้วนเป็นปัจจัยที่อาจนำไปสู่การตีความผิดพลาดและเพิ่มความเสี่ยงต่อการเกิดอุบัติเหตุ (Kozhamkulova et al., 2024; Zhu & Yan, 2022) ส่งผลให้การพัฒนาระบบรู้จำป้ายจราจรโดยอัตโนมัติ (Traffic Sign Recognition: TSR) ได้รับความสนใจอย่างมากในช่วงไม่กี่ปีที่ผ่านมา

ในอดีตการพัฒนาเทคโนโลยีรู้จำป้ายจราจรอาศัยเทคนิคประมวลผลภาพแบบดั้งเดิม เช่น Haar-Like Feature Detection หรือ Template Matching ซึ่งยังคงมีข้อจำกัดด้านความแม่นยำและความยืดหยุ่นในสภาวะแวดล้อมจริง (Khunarsa et al., 2021; Thipsanthia, 2021) จนกระทั่งแนวคิดการเรียนรู้เชิงลึก (Deep Learning) โดยเฉพาะโครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNNs) ได้รับการประยุกต์ใช้อย่างแพร่หลายและพิสูจน์แล้วว่ามีประสิทธิภาพสูงในการจำแนกภาพและวัตถุหลายประเภท สามารถปรับให้เข้ากับสภาพแวดล้อมที่หลากหลาย (Dhawan et al., 2023; Jantaragate, 2024)

อย่างไรก็ตาม ข้อจำกัดด้านความรู้ทางเทคนิคและทรัพยากรฮาร์ดแวร์ที่จำเป็นสำหรับการฝึกโมเดล CNNs ยังคงเป็นอุปสรรคสำคัญต่อผู้เริ่มต้น นักพัฒนาและนักวิจัยในระดับทั่วไป ด้วยเหตุนี้ งานวิจัยฉบับนี้จึงเลือกใช้ Teachable Machine ซึ่งเป็นแพลตฟอร์มแบบไม่ต้องเขียนโค้ด (No-Code Platform) ที่พัฒนาโดย Google Creative Lab โดยสามารถฝึกโมเดลการเรียนรู้ของเครื่อง (Machine Learning) บนเว็บเบราว์เซอร์ได้อย่างสะดวก พร้อมสนับสนุน

Citation: Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

สถาปัตยกรรม MobileNet และการฝึกด้วย Adam Optimizer ผ่าน TensorFlow.js ซึ่งช่วยให้สามารถฝึกและนำโมเดลไปใช้งานจริงในฝั่งผู้ใช้ (Client-side) ได้อย่างมีประสิทธิภาพโดยไม่ต้องพึ่งเซิร์ฟเวอร์ (Google Creative Lab, 2020; Kingma & Ba, 2014) การศึกษาโดย Chen et al. (2020) ระบุว่า Teachable Machine ช่วยเปิดโอกาสให้ผู้เรียนสามารถเข้าถึงแนวคิดพื้นฐานของ AI และ การเรียนรู้ของเครื่องได้อย่างเข้าใจง่าย ส่วนงานวิจัยของ Kurz et al. (2024) และ Yim & Su (2025) ชี้ให้เห็นว่าเครื่องมือนี้มีศักยภาพในการนำไปประยุกต์ใช้เพื่อการเรียนการสอนและการพัฒนาทักษะในระดับ K-12 และระดับมัธยมศึกษาตอนต้นได้อย่างมีประสิทธิภาพ

นอกจากนี้ งานวิจัยยังเน้นการออกแบบชุดข้อมูลให้มีความครอบคลุมทั้งในแง่ของแหล่งที่มาและลักษณะของป้ายจราจร โดยผสานข้อมูลจากฐาน GTSRB และภาพป้ายจราจรในประเทศไทย พร้อมทั้งประยุกต์ใช้เทคนิค Data Augmentation เช่น การหมุนภาพ การปรับแสง และการแปลงมุมมอง เพื่อเพิ่มข้อมูลเทียมในชุดฝึก เพิ่มประสิทธิภาพของโมเดลให้สามารถรับมือกับความหลากหลายของภาพในแต่ละคลาส ลดปัญหา Overfitting และเพิ่มความสามารถในการจำแนกภาพในสถานการณ์ที่ซับซ้อน (Alsiyeu & Duisebekov, 2024; Dineley et al., 2024; Ge, 2023)

คุณูปการของงานวิจัยนี้อยู่ที่การนำเสนอแนวทางการพัฒนาโมเดลรู้จำป้ายจราจรที่มีความแม่นยำสูง โดยใช้เครื่องมือแบบไม่ต้องเขียนโปรแกรมอย่าง Teachable Machine ร่วมกับเทคนิคการเตรียมข้อมูลที่เหมาะสม ซึ่งสามารถนำไปประยุกต์ใช้งานจริงผ่านระบบเว็บแอปพลิเคชันแบบเรียลไทม์ โดยไม่จำเป็นต้องใช้ทรัพยากรคอมพิวเตอร์ในระดับสูง งานวิจัยจึงมีเป้าหมายเพื่อออกแบบและพัฒนาโมเดลที่เข้าถึงง่าย ประหยัดทรัพยากร และสามารถฝึกใช้งานได้แม้ในสภาพแวดล้อมที่มีข้อจำกัด ทั้งยังรองรับการใช้งานได้อย่างมีประสิทธิภาพบนอุปกรณ์ทั่วไป ไม่ว่าจะเป็นคอมพิวเตอร์หรืออุปกรณ์พกพา บทความฉบับนี้จะกล่าวถึงกระบวนการพัฒนาและเทรนโมเดลผ่าน Teachable Machine ตั้งแต่การเตรียมข้อมูล การฝึกและทดสอบโมเดล การออกแบบระบบเว็บแอปพลิเคชัน ไปจนถึงการวิเคราะห์ผลการทดลองเชิงปริมาณผ่านตัวชี้วัดมาตรฐาน ได้แก่ Accuracy, Precision, Recall และ F1-measure เพื่อยืนยันถึงความเป็นไปได้ในการนำแนวทางดังกล่าวไปใช้จริงในบริบทของระบบอัจฉริยะบนท้องถนน

2. วัตถุประสงค์งานวิจัย (Research Objectives)

เพื่อพัฒนาโมเดลสำหรับจำแนกประเภทของป้ายจราจรอัตโนมัติ โดยใช้เครื่องมือ Teachable Machine

3. การทบทวนวรรณกรรมและทฤษฎีที่เกี่ยวข้อง (Literature Review)

3.1 งานวิจัยที่เกี่ยวข้อง

การรู้จำป้ายจราจร (Traffic Sign Recognition: TSR) เป็นองค์ประกอบสำคัญของระบบช่วยเหลือผู้ขับขี่ขั้นสูง (Advanced Driver Assistance Systems: ADAS) และยานยนต์อัตโนมัติ ซึ่งมีบทบาทสำคัญในการเพิ่มความปลอดภัยและประสิทธิภาพในการจราจร โดยเฉพาะอย่างยิ่งเมื่อได้รับการสนับสนุนจาก พัฒนาการของเทคโนโลยีการเรียนรู้ของเครื่อง (Machine Learning) และการเรียนรู้เชิงลึก (Deep Learning) ที่ช่วยยกระดับประสิทธิภาพของระบบดังกล่าวอย่างมีนัยสำคัญ

งานวิจัยด้าน TSR ในช่วงทศวรรษที่ผ่านมาแสดงให้เห็นถึงการใช้โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks: CNNs) เป็นแกนหลักของการพัฒนา เนื่องจาก CNNs มีศักยภาพสูงในการสกัดและจำแนกลักษณะสำคัญของภาพได้แม้ในสภาพแวดล้อมที่ซับซ้อน ตัวอย่างเช่น Kozhamkulova et al. (2024) ใช้ CNNs เพื่อปรับปรุงความแม่นยำของการจำแนกป้ายในยานยนต์อัตโนมัติภายใต้สภาวะหลากหลาย Dhawan et al. (2023) มุ่งพัฒนาโมเดลเพื่อการตรวจจับและจำแนกป้ายในเมืองอัจฉริยะ ขณะที่ Zhu & Yan (2022) ศึกษาการเพิ่มความทนทานของโมเดลต่อสภาพแวดล้อมที่เปลี่ยนแปลง ก่อนการแพร่หลายของ CNNs วิธีการดั้งเดิม เช่น Haar-like Feature Detection และการจับคู่คุณลักษณะ (Feature Matching) ถูกนำมาใช้ แต่มีข้อจำกัดด้านความแม่นยำและความยืดหยุ่น (Khunarsa et al., 2021) งานต่อมาจึงพัฒนาโครงสร้างตรวจจับที่ซับซ้อนยิ่งขึ้น เช่น YOLOv3 ร่วมกับ Feature Pyramid Networks (FPN) เพื่อแก้ปัญหาการตรวจจับป้ายขนาดเล็ก (Thipsanthia, 2021) และการปรับโครงสร้างโมเดลให้รองรับการเปลี่ยนแปลงของรูปทรงและขนาดป้าย (Jantaragate, 2024) อย่างไรก็ตาม ความท้าทายยังคงมีอยู่ในกรณีของแสงจ้า การบดบัง หรือข้อจำกัดของชุดข้อมูล

Citation:
Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

จากภาพรวมดังกล่าวจะเห็นได้ว่า CNNs เป็นเทคโนโลยีหลักของงาน TSR ในปัจจุบัน โดยมีการพัฒนาอย่างต่อเนื่องทั้งในด้านสถาปัตยกรรม การจัดการข้อมูล และเทคนิคเพิ่มประสิทธิภาพ เพื่อยกระดับความแม่นยำและความทนทานของระบบให้พร้อมต่อการใช้งานจริง

3.2 โครงข่ายประสาทเทียมแบบคอนโวลูชัน (Convolutional Neural Networks: CNNs)

โครงข่ายประสาทเทียมแบบคอนโวลูชัน (CNNs) เป็นสถาปัตยกรรมของโครงข่ายประสาทเทียม (Artificial Neural Networks: ANNs) ที่ออกแบบมาสำหรับข้อมูลที่มีโครงสร้างเป็นตาราง เช่น ภาพหรือเสียง โดยเฉพาะในงานรู้จำวัตถุและจำแนกรูปภาพ องค์ประกอบสำคัญประกอบด้วย

1) ชั้นคอนโวลูชัน (Convolutional Layer) ทำหน้าที่สกัดคุณลักษณะจากภาพอินพุตด้วยตัวกรอง (Filter/Kernel) ขนาดเล็กที่เลื่อนครอบพื้นที่ย่อยของภาพ เพื่อสร้างแผนที่คุณลักษณะ (Feature Map) ที่สะท้อนลักษณะสำคัญของข้อมูล

2) ชั้นเปิดใช้งาน (Activation Layer) เพิ่มความไม่เชิงเส้นให้โมเดลเพื่อรองรับการเรียนรู้ข้อมูลที่ซับซ้อน โดยนิยมใช้ฟังก์ชัน ReLU ซึ่งมีความเรียบง่ายและช่วยลดปัญหา Gradient Vanishing

3) ชั้นลดขนาด (Pooling Layer) ลดมิติและจำนวนพารามิเตอร์ของข้อมูลเพื่อป้องกัน Overfitting ขณะยังคงรักษาคุณลักษณะสำคัญ นิยมใช้ Max Pooling หรือ Average Pooling ตามความเหมาะสมของงาน

4) ชั้นเชื่อมต่อเต็ม (Fully Connected Layer) รวมคุณลักษณะที่ได้จากชั้นก่อนหน้าเพื่อใช้ในการจำแนกประเภทหรือการทำนายอื่น ๆ โดยมักต่อด้วยฟังก์ชัน Softmax หรือ Sigmoid เพื่อแปลงค่าเป็นความน่าจะเป็น

5) ชั้นซอฟต์แมกซ์ (Softmax Output Layer) ใช้ในการจำแนกหลายคลาส โดยแปลงค่าล็อกิต (Logit) จากชั้นเชื่อมต่อเต็มให้เป็นค่าความน่าจะเป็นรวมเท่ากับ 1 และเลือกคลาสที่มีค่าสูงสุดเป็นคำตอบ

ดังนั้น CNNs จึงมีข้อได้เปรียบในความสามารถเรียนรู้คุณลักษณะหลายระดับ ตั้งแต่ขอบ รูปร่าง ไปจนถึงรูปแบบที่ซับซ้อน เหมาะสมกับลักษณะของป้ายจราจรที่ต้องอาศัยการจำแนกจากลักษณะเฉพาะของภาพ อีกทั้งสามารถปรับให้มีโครงสร้างเบา เช่น MobileNet หรือ EfficientNet เพื่อลดภาระประมวลผลบนอุปกรณ์ที่มีทรัพยากรจำกัด จึงเป็นเทคโนโลยีหลักที่สามารถประยุกต์ใช้ร่วมกับเครื่องมืออย่าง Teachable Machine ได้อย่างมีประสิทธิภาพ

3.3 Teachable Machine

Teachable Machine เป็นแพลตฟอร์มแบบไม่ต้องเขียนโค้ด (No-Code Platform) ที่พัฒนาโดย Google Creative Lab ออกแบบมาเพื่อให้ผู้ใช้ทุกระดับสามารถสร้างและฝึกโมเดลการเรียนรู้ของเครื่องได้โดยตรงบนเว็บเบราว์เซอร์ ผ่าน TensorFlow.js และใช้ Adam Optimizer ในการปรับค่าน้ำหนักของโมเดล โดยเฉพาะในงานจำแนกภาพ แพลตฟอร์มนี้รองรับการใช้สถาปัตยกรรม MobileNet ซึ่งมีความเหมาะสมกับสภาพแวดล้อมที่มีข้อจำกัดด้านทรัพยากร เช่น อุปกรณ์พกพาหรือการประมวลผลฝั่งผู้ใช้ (Google Creative Lab, 2020; Kingma & Ba, 2014)

งานวิจัยหลายชิ้นได้ยืนยันศักยภาพของ Teachable Machine ทั้งในด้านการส่งเสริมการเรียนรู้และการประยุกต์ใช้งานจริง เช่น Chen et al. (2020) พบว่าเครื่องมือนี้ช่วยให้ผู้เรียนที่ไม่มีพื้นฐานทางเทคนิคสามารถเข้าใจแนวคิดของการเรียนรู้ของเครื่องผ่านการทดลองเชิงปฏิบัติ Kurz et al. (2024) แสดงให้เห็นว่า Teachable Machine เพิ่มความมั่นใจของครูผู้สอน AI ในระดับการศึกษาขั้นพื้นฐาน และ Yim & Su (2025) ชี้ให้เห็นว่ามีประโยชน์ต่อการเรียนการสอน AI ในระดับ K-12 นอกจากนี้ Teachable Machine ยังถูกนำไปใช้ในงานวิจัยด้านการรู้จำวัตถุต่าง ๆ เช่น การรู้จำชนิดของนก (Wong & Fadzly, 2022) และการวินิจฉัยทางการแพทย์ เช่น การตรวจสอบลักษณะของลิ้นที่มีรอยฟัน (Jeong, 2020) ซึ่งแสดงให้เห็นถึงความยืดหยุ่นและประสิทธิภาพของเครื่องมือนี้ในหลากหลายบริบท

ในการพัฒนาโมเดลรู้จำป้ายจราจร การใช้ Teachable Machine ร่วมกับเทคนิค Data Augmentation มีความสำคัญต่อการแก้ปัญหาความไม่สมดุลของข้อมูลและการเพิ่มความหลากหลายของชุดฝึก เทคนิคดังกล่าวครอบคลุมถึงการหมุนภาพ ปรับแสง เพิ่มสัญญาณรบกวน และการปรับมุมมอง ซึ่งช่วยลด Overfitting และเพิ่มความทนทานของโมเดล งานวิจัย เช่น Ge (2023), Alsiyeu & Duisebekov (2024) และ Dineley et al. (2024) แสดงให้เห็นว่า Data Augmentation มีส่วนช่วยเพิ่มความแม่นยำของโมเดล TSR ได้อย่างมีนัยสำคัญ

จากการทบทวนวรรณกรรมและงานวิจัยที่เกี่ยวข้อง พบว่า Teachable Machine เป็นเครื่องมือที่มีศักยภาพในการพัฒนาโมเดลสำหรับการรู้จำป้ายจราจร ด้วยอินเทอร์เฟซที่ใช้งานง่าย ความสามารถในการทำงานบนเว็บโดยไม่ต้องติดตั้งซอฟต์แวร์ และการรองรับสถาปัตยกรรมที่เหมาะกับข้อจำกัดด้านทรัพยากร Teachable Machine จึงเป็นเครื่องมือที่มี

Citation:
Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

ศักยภาพสูงสำหรับการพัฒนาโมเดลรู้จำป้ายจราจรในบริบทจริง โดยเฉพาะในกลุ่มผู้ใช้ที่ไม่มีพื้นฐานด้านการเขียนโปรแกรม แต่ต้องการสร้างระบบจำแนกประเภทที่มีประสิทธิภาพ

4. กรอบแนวคิดงานวิจัย (Conceptual Framework)

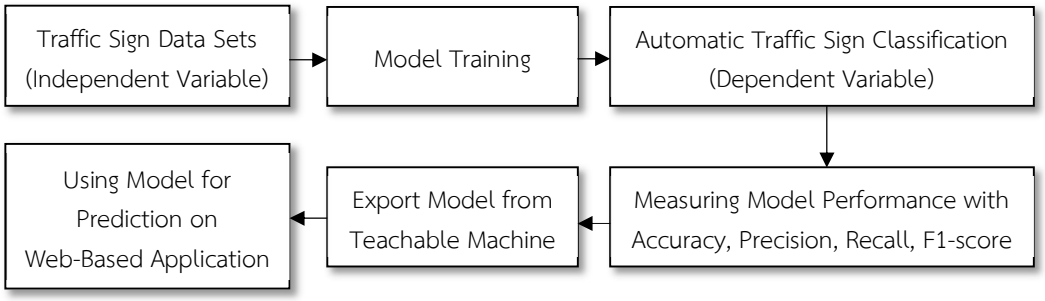


Figure 1. Conceptual Framework for Traffic Recognition System

จาก Figure 1. แสดงกรอบแนวคิดงานวิจัยโดยมีโครงสร้างแนวคิดดังนี้

1) ตัวแปรต้น (Independent Variable) เป็นข้อมูลภาพป้ายจราจรที่ถูกรวบรวมจากชุดข้อมูลมาตรฐาน German Traffic Sign Recognition Benchmark (GTSRB) ซึ่งสามารถดาวน์โหลดชุดข้อมูล GTSRB ได้จากเว็บไซต์ทางการ https://benchmark.ini.rub.de/gtsrb_dataset.html (Stallkamp, 2011) และภาพถ่ายในประเทศไทยซึ่งผู้วิจัยเป็นผู้เก็บภาพเพิ่มเติมเอง เพื่อเพิ่มความหลากหลายของข้อมูล

2) กระบวนการฝึกโมเดล (Model Training) ใช้ข้อมูลภาพที่เตรียมไว้มาฝึกสอนโมเดล ผ่านแพลตฟอร์ม Teachable Machine เพื่อให้สามารถจำแนกประเภทของป้ายจราจรได้อย่างแม่นยำ

3) ตัวแปรตาม (Dependent Variable) โมเดลที่ได้จะสามารถจำแนกป้ายจราจรประเภทต่าง ๆ ได้

4) การวัดประสิทธิภาพของโมเดล ระบบจำแนกป้ายจราจรอัตโนมัติ สามารถวัดผลด้วย Accuracy, Precision, Recall และ F1-Score

5) Export โมเดลออกจาก Teachable Machine เป็น TensorFlow.js พร้อมทั้งเตรียมโครงสร้างเว็บไซต์พื้นฐาน

6) นำโมเดลที่ได้ผลลัพธ์ดีที่สุดจากการฝึกด้วย Teachable Machine ไปพัฒนาเป็นเว็บแอปพลิเคชันสำหรับการตรวจจับและจดจำป้ายจราจรอัตโนมัติในสภาพแวดล้อมจริง

5. วิธีดำเนินงานวิจัย (Research Methodology)

วิธีดำเนินงานวิจัยแบ่งออกเป็น 2 หัวข้อใหญ่ คือ เครื่องมือและอุปกรณ์ที่ใช้ในงานวิจัย และกระบวนการพัฒนาและเทรนโมเดล โดยกระบวนการนี้ประกอบด้วย 4 ส่วนสำคัญ ได้แก่ (1) การเตรียมข้อมูลภาพป้ายจราจรและการจัดหมวดหมู่ข้อมูลอย่างมีโครงสร้าง (2) การพัฒนาและฝึกสอนโมเดลการเรียนรู้ของเครื่องโดยใช้แพลตฟอร์ม Teachable Machine (3) การทดสอบโมเดล และการจัดเก็บและบริหารข้อมูลผลการทำนายผ่านระบบฐานข้อมูล และ (4) การออกแบบส่วนติดต่อผู้ใช้งาน (User Interface) เพื่อให้สามารถใช้งานโมเดลในรูปแบบเว็บแอปพลิเคชันได้อย่างสะดวกและมีประสิทธิภาพ

5.1 เครื่องมือและอุปกรณ์ที่ใช้ในงานวิจัย

งานวิจัยนี้ใช้เครื่องมือและเทคโนโลยีหลากหลายประเภทเพื่อพัฒนาระบบจดจำป้ายจราจรอัตโนมัติบนเว็บแอปพลิเคชัน โดยแบ่งออกเป็น 3 กลุ่มหลัก ดังนี้

1) เครื่องมือสำหรับการพัฒนาเว็บไซต์ ใช้ภาษา HTML และ CSS ในการออกแบบและกำหนดโครงสร้างของเว็บเพจ ส่วนติดต่อผู้ใช้ (User Interface) พัฒนาโดยใช้ภาษา JavaScript เพื่อเพิ่มความสามารถในการโต้ตอบ เช่น การแสดงผลภาพและค่าความมั่นใจ (Confidence Score) แบบเรียลไทม์ สำหรับการประมวลผลฝั่งเซิร์ฟเวอร์ ใช้ Flask

Citation:
Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

Framework ในการจัดการ API และควบคุมการรับส่งข้อมูลระหว่างผู้ใช้กับโมเดล พร้อมทั้งใช้ Ngrok ในการสร้างลิงก์ที่ปลอดภัยเพื่ออำนวยความสะดวกในการทดสอบระบบผ่านเครือข่ายอินเทอร์เน็ตภายนอก

2) เครื่องมือสำหรับการพัฒนาและฝึกสอนโมเดลการเรียนรู้ของเครื่อง ใช้ Teachable Machine ซึ่งเป็นแพลตฟอร์มที่พัฒนาโดย Google เพื่อสร้างโมเดลโดยไม่ต้องเขียนโค้ด เลือกประเภท “Image Project” สำหรับการจำแนกรูปภาพ โดยกำหนดค่าพารามิเตอร์การฝึก ได้แก่ ขนาดกลุ่ม (Batch Size) 16, จำนวนรอบการฝึก (Epochs) 50 และอัตราการเรียนรู้ (Learning Rate) 0.001 ซึ่งเหมาะสมกับการฝึกบนเว็บเบราว์เซอร์ เมื่อการฝึกเสร็จสิ้น โมเดลจะถูกส่งออกในรูปแบบ TensorFlow.js เพื่อรองรับการประมวลผลฝั่งไคลเอนต์บนเว็บแอปพลิเคชัน

3) เครื่องมือสำหรับการประมวลผลภาพและการจัดการข้อมูล ใช้ OpenCV.js สำหรับการจัดการภาพ เช่น การจับภาพจากกล้อง การปรับแสง และการตัดขอบ เพื่อเตรียมข้อมูลให้อยู่ในรูปแบบที่เหมาะสมกับการประมวลผลโดยโมเดล จากนั้นใช้ TensorFlow, Keras และ NumPy สำหรับการประมวลผลเบื้องหลังและการทดสอบโมเดลในสภาพแวดล้อมที่ซับซ้อน พัฒนาและจัดการโค้ดผ่าน PyCharm ซึ่งเป็นสภาพแวดล้อมการพัฒนาแบบบูรณาการ (IDE) ที่รองรับไลบรารีด้านการเรียนรู้ของเครื่อง ในส่วนของระบบฐานข้อมูล ใช้ MySQL สำหรับจัดเก็บข้อมูล เช่น ชื่อป้าย ผลการจำแนก และเวลาที่ทำการพยากรณ์ โดยออกแบบให้มีความสัมพันธ์เชื่อมโยงกันอย่างเป็นระบบ เพื่อให้สามารถเรียกใช้และวิเคราะห์ข้อมูลย้อนหลังได้อย่างมีประสิทธิภาพ

5.2 กระบวนการพัฒนาและเทรนโมเดล

1) การเตรียมข้อมูลป้ายจราจร

งานวิจัยนี้ใช้ชุดข้อมูลภาพป้ายจราจรจากสองแหล่งหลัก ได้แก่ ฐานข้อมูลมาตรฐาน German Traffic Sign Recognition Benchmark (GTSRB) และภาพถ่ายป้ายจราจรในประเทศไทยที่งานวิจัยนี้ได้รวบรวมเพิ่มเติม เพื่อให้ครอบคลุมลักษณะป้ายที่ใช้จริงในบริบทท้องถิ่น รวมทั้งหมด 17,440 ภาพ ครอบคลุม 27 คลาส (รายละเอียดใน Table 1.)

Table 1. Data Sets

ID	Signs	Types	Thai Name	Engling Name	Amount
1		Red Mandatory Sign – Symbol Type	จำกัดความเร็ว 30 กม./ชม.	Speed limit 30 km./h.	250
2		Red Mandatory Sign – Symbol Type	จำกัดความเร็ว 40 กม./ชม.	Speed limit 40 km./h.	1,000
3		Red Mandatory Sign – Symbol Type	จำกัดความเร็ว 60 กม./ชม.	Speed limit 60 km./h.	816
4		Red Mandatory Sign – Symbol Type	ห้ามเลี้ยวขวา	No right turns	1,251
5		Red Mandatory Sign – Symbol Type	ห้ามเข้า	No entry	1,091
6		Red Mandatory Sign – Symbol Type	ห้ามหยุดรถ	No stopping	500
7		Red Mandatory Sign – Symbol Type	ให้รถสวนทางมาก่อน	Let incoming traffic go first	721
8		Red Mandatory Sign – Text Type	หยุด	Stop	250
9		Blue Mandatory Sign	ให้เลี้ยวซ้าย	Turn left	1,151
10		Blue Mandatory Sign	ให้เลี้ยวขวา	Turn right	500
11		Blue Mandatory Sign	ให้ชิดซ้าย	Keep left	311
12		Blue Mandatory Sign	จุดกลับรถ	U-turn	250
13		Blue Mandatory Sign	ทางเดินรถทางเดียว	One way	630
14		Blue Mandatory Sign	ให้ตรงไปหรือเลี้ยวซ้าย	Proceed straight or turn left	253

Citation: Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

Table 1. Data Sets (Cont.)

ID	Signs	Types	Thai Name	Engling Name	Amount
15		Blue Mandatory Sign	ตำแหน่งทางข้าม	Pedestrian crossing	250
16		Blue Mandatory Sign	ช่องเดิน รถจักรยาน	Bicycle lane	250
17		General Warning Sign	ทางโค้งขวา	Right curve	750
18		General Warning Sign	เตือนรถกระโดด	Beware of bouncing vehicles	250
19		General Warning Sign	ทางเอกติดกันรูปตัวที	Main roads intersect in a T shape	750
20		General Warning Sign	ทางเอกตัดกัน	Secondary roads intersect main roads	750
21		General Warning Sign	วงเวียน	Roundabout	877
22		General Warning Sign	ให้ไปทางซ้ายหรือขวา	Go left or right	897
23		General Warning Sign	ทางแคบด้านซ้าย	Narrow path on the left	750
24		General Warning Sign	ทางข้ามทางรถไฟไม่มีเครื่องกั้นทาง	Uncontrolled railway crossing	453
25		General Warning Sign	สัญญาณไฟจราจร	Traffic signal	627
26		General Warning Sign	เขตชุมชนลดความเร็ว	City limit produce speed	303
27		Orange Warning Sign – Text Type	ลดความเร็ว	Reduce speed	586

จากการวิเคราะห์พบว่าจำนวนภาพในแต่ละคลาสไม่สมดุลกัน (Class Imbalance) โดยบางคลาสมีภาพน้อยกว่า 100 ภาพ ขณะที่บางคลาสมีมากกว่า 500 ภาพ ความไม่สมดุลนี้อาจทำให้โมเดลเกิดการลำเอียง (Bias) ไปยังคลาสที่มีข้อมูลมาก เพื่อแก้ไขปัญหานี้ จึงนำเทคนิค Data Augmentation มาเพิ่มข้อมูลในคลาสที่มีจำนวนน้อย เช่น การหมุนภาพ (Rotation), การสะท้อนภาพ (Flipping), การปรับความสว่าง (Brightness Adjustment), การแปลงมุมมอง (Perspective Transformation) และการเพิ่มสัญญาณรบกวน (Noise Injection) เพื่อเพิ่มความหลากหลายของข้อมูลโดยไม่สูญเสียคุณลักษณะสำคัญ อีกทั้งช่วยลดปัญหา Overfitting โดยก่อนการฝึก ได้ดำเนินการ ทำความสะอาดข้อมูล (Data Cleaning) เพื่อลบภาพที่ไม่สมบูรณ์ เช่น เบลอเกินไป ถูกบดบัง หรือมีสัญญาณรบกวนสูง หลังการประมวลผลทั้งหมด ทำให้ได้ชุดข้อมูลที่มีภาพในแต่ละคลาสระหว่าง 250–1,251 ภาพ และปรับขนาดภาพเป็น 224 × 224 พิกเซล ซึ่งตรงกับข้อกำหนดอินพุตของสถาปัตยกรรม MobileNet

2) การเทรนโมเดลใน Teachable Machine

การฝึกดำเนินการบนแพลตฟอร์ม Teachable Machine โดยแยกข้อมูลเป็น ชุดฝึก (Training Set) 85% และชุดทดสอบ (Testing Set) 15% ของข้อมูลทั้งหมด การฝึกโมเดลเริ่มต้นด้วยการเลือกประเภทโปรเจกต์เป็น Image Project ถัดมาให้ดำเนินการอัปโหลดภาพป้ายจราจรพร้อมจัดกลุ่มตามคลาส เช่น ป้ายหยุด ป้ายห้ามเข้า หรือป้ายเตือนทางโค้ง จากนั้นตั้งค่าพารามิเตอร์โดยให้ Batch Size เท่ากับ 16 จำนวน Epochs เท่ากับ 50 และ Learning Rate เท่ากับ 0.001

การกำหนดค่า Epoch ที่ 50 เป็นค่ากลางที่สมดุลระหว่างการเรียนรู้เพียงพอกับการป้องกัน Overfitting หากกำหนดค่า Epoch น้อยเกินไป เช่น ต่ำกว่า 20 โมเดลอาจยังไม่สามารถจับรูปแบบของข้อมูลได้อย่างมีประสิทธิภาพ (Underfitting) ในขณะที่การตั้งค่าที่สูงเกินไป เช่น มากกว่า 100 อาจทำให้เกิดปัญหา Overfitting โดยเฉพาะในกรณีที่ชุด

Citation: Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

ข้อมูลมีขนาดเล็ก ทั้งยังส่งผลต่อระยะเวลาในการฝึกที่เพิ่มขึ้นโดยไม่จำเป็น ดังนั้นหากกำหนดจำนวน Epoch เท่ากับ 50 โมเดลจะผ่านชุดข้อมูลฝึกทั้งหมดจำนวน 50 ครั้ง ในแต่ละครั้งข้อมูลจะถูกแบ่งออกเป็นกลุ่มย่อยตาม Batch Size เพื่อทำการอัปเดตพารามิเตอร์ของโมเดลเป็นรายกลุ่ม เช่น ในกรณีของคลาส “Speed limit 40 km./h.” ที่มีภาพรวมทั้งหมด 1,000 รูป เมื่อแบ่งเป็นชุดฝึก 85% และชุดทดสอบ 15% จะได้ภาพสำหรับฝึกจำนวน 850 รูป ซึ่งเมื่อแบ่งออกเป็นกลุ่มย่อยขนาด 16 ภาพต่อกลุ่ม จะได้จำนวนกลุ่มฝึกทั้งหมดประมาณ 54 กลุ่มต่อ Epoch (ปัดเศษขึ้นจาก 53.125) โดยการฝึกที่ 50 Epoch จะส่งผลให้โมเดลได้เรียนรู้จากภาพ 850 รูปแบบซ้ำ ๆ ทั้งหมด 50 รอบ ส่วนการเลือก Batch Size ขนาดเล็กที่ 16 ภาพต่อกลุ่ม ช่วยให้โมเดลอัปเดตค่าน้ำหนักได้ถี่มากขึ้น เพิ่มความยืดหยุ่นในการเรียนรู้และทำให้โมเดลตอบสนองต่อการกระจายตัวของข้อมูลและสัญญาณรบกวนได้ดี และใช้หน่วยความจำน้อย เหมาะกับการฝึกบนเว็บเบราว์เซอร์ ลดเวลาในการประมวลผลและเพิ่มความสะดวกในการใช้งาน โดยเฉพาะอย่างยิ่งในบริบทที่มีข้อจำกัดด้านทรัพยากร ในขณะที่ Learning Rate กำหนดไว้ที่ 0.001 นั้นถือเป็นค่าที่เหมาะสมสำหรับการทำงานร่วมกับอัลกอริธึม Adam Optimizer (Kingma & Ba, 2014) เพื่อให้การลดค่าความสูญเสีย (Loss) เป็นไปอย่างเสถียร โดย Adam Optimizer เป็นอัลกอริธึมที่ Teachable Machine ใช้เป็นพื้นฐานในการฝึกโมเดล (Google Creative Lab, 2020) โดยค่า Learning Rate ที่ระดับนี้ให้ผลลัพธ์ที่เสถียร เร็วพอที่จะทำให้ Loss ลดลงภายในจำนวน Epoch ที่กำหนด และไม่เร็วเกินไปจนทำให้โมเดลข้ามค่าที่เหมาะสม (Overshooting) หากตั้งค่าต่ำเกินไป เช่น 0.0001 โมเดลจะใช้เวลาฝึกนาน และหากตั้งค่าสูงเกินไป เช่น 0.01 อาจทำให้ Loss แกว่งหรือไม่เสถียร ภายหลังจากการจัดหมวดหมู่และตั้งค่าพารามิเตอร์การฝึกเรียบร้อยแล้ว ระบบจะเริ่มกระบวนการฝึกโมเดลโดยอัตโนมัติ โดยระหว่างการฝึก ระบบจะแสดงกราฟ Accuracy และ Loss แบบเรียลไทม์ เพื่อให้ติดตามความคืบหน้าได้ หลังฝึกเสร็จ โมเดลสามารถส่งออกเป็น TensorFlow.js สำหรับใช้งานบนเว็บ หรือ TensorFlow Lite สำหรับอุปกรณ์พกพา

3) การทดสอบโมเดล

การทดสอบใช้ชุดข้อมูล Testing Set ที่แยกไว้ตั้งแต่ต้น เพื่อประเมินประสิทธิภาพการจำแนกป้ายจราจรที่ไม่เคยเห็นมาก่อน โดยใช้ตัวชี้วัดหลัก ได้แก่ Accuracy, Precision, Recall และ F1-Measure นอกจากนี้ยังสร้าง Confusion Matrix เพื่อวิเคราะห์จุดแข็ง จุดอ่อน และความสับสนของโมเดลในแต่ละคลาส พร้อมตรวจสอบความเสถียรของการเรียนรู้ผ่านกราฟ Accuracy และ Loss เพื่อระบุแนวโน้ม Overfitting หรือ Underfitting โดยพิจารณาความสอดคล้องระหว่างค่าความแม่นยำของชุดฝึกและชุดทดสอบในแต่ละรอบการเรียนรู้ ซึ่งผลจากการทดสอบทั้งหมดจะถูกนำมาใช้ในการอภิปรายผล และพิจารณาความเหมาะสมของโมเดลสำหรับการนำไปใช้งานจริงในระบบเว็บแอปพลิเคชันที่พัฒนาขึ้น

4) การนำโมเดลไปใช้งานจริง

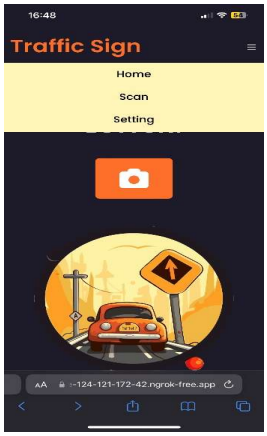
ภายหลังจากที่โมเดลได้รับการฝึกและประเมินผลแล้ว งานวิจัยนี้ได้นำโมเดลไปประยุกต์ใช้ในระบบเว็บแอปพลิเคชัน เพื่อให้ผู้ใช้งานสามารถจำแนกป้ายจราจรจากภาพถ่ายหรือจากกล้องเว็บแคมได้แบบเรียลไทม์ โดยไม่จำเป็นต้องติดตั้งซอฟต์แวร์เพิ่มเติม และสามารถใช้งานได้ผ่านเว็บเบราว์เซอร์ทั่วไปบนอุปกรณ์มาตรฐาน เช่น คอมพิวเตอร์โน้ตบุ๊ก แท็บเล็ต หรือสมาร์ตโฟน โดยโมเดลที่ได้จากแพลตฟอร์ม Teachable Machine จะถูกส่งออกในรูปแบบ TensorFlow.js ซึ่งประกอบด้วยไฟล์ .json และ .bin สำหรับใช้ในฝั่งไคลเอนต์ (Client-Side) โดยไม่ต้องพึ่งพาเซิร์ฟเวอร์ในการประมวลผลหลัก ทั้งนี้โมเดลจะถูกฝังลงในโค้ด HTML และเรียกใช้งานผ่าน JavaScript API โดยมีการออกแบบส่วนติดต่อผู้ใช้ (User Interface) ให้สามารถแสดงภาพจากกล้อง กล่องข้อความแสดงชื่อป้ายที่ตรวจพบ พร้อมค่าความมั่นใจ (Confidence Score) ที่เข้าใจง่ายและสื่อความหมายได้อย่างชัดเจน

การประมวลผลภาพดำเนินการผ่านกล้องแบบเรียลไทม์ (Live Webcam Feed) โดยใช้ไลบรารี TensorFlow.js ควบคู่กับ WebRTC สำหรับการเข้าถึงกล้องในฝั่งผู้ใช้ ทำให้สามารถประมวลผลและจำแนกภาพได้ในเครื่องผู้ใช้โดยตรง ส่งผลให้ลดความล่าช้า (Latency) และเพิ่มความปลอดภัยของข้อมูลภาพ ทั้งนี้ยังมีการใช้ OpenCV.js เพื่อเตรียมข้อมูลภาพ เช่น การปรับขนาด การจัดตำแหน่ง และการปรับค่าแสงให้อยู่ในรูปแบบที่เหมาะสมก่อนส่งเข้าสู่โมเดล

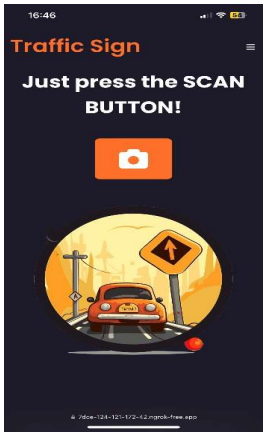
นอกจากการประมวลผลจากกล้องแล้ว ระบบยังรองรับการอัปโหลดภาพจากไฟล์ (Image Upload) เพื่อประมวลผลแบบ Single-Image Inference โดยแสดงผลการจำแนกทันทีที่การวิเคราะห์เสร็จสิ้น พร้อมทั้งมีการจัดเก็บผลลัพธ์ ได้แก่ ชื่อคลาส ค่าความมั่นใจ และเวลาที่จำแนก ลงในฐานข้อมูล MySQL เพื่อใช้ในการวิเคราะห์ผลย้อนหลังและประเมินประสิทธิภาพของระบบในสถานการณ์การใช้งานจริง ซึ่งการออกแบบระบบในลักษณะนี้เน้นความเรียบง่ายในการใช้งาน ประหยัดทรัพยากร และสามารถเข้าถึงได้ง่าย เหมาะสมกับวัตถุประสงค์ของงานวิจัยที่ต้องการพัฒนาเครื่องมือที่ใช้

Citation:
Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

งานได้จริงในบริบทการศึกษาและการประยุกต์ใช้งานทั่วไปในระดับสาธารณะ นอกจากนี้ ผู้ใช้งานสามารถให้คะแนนความพึงพอใจและรายงานปัญหาผ่านแบบฟอร์มความคิดเห็น เพื่อให้ผู้พัฒนานำข้อมูลที่ได้ไปปรับปรุงระบบให้ตอบสนองต่อการใช้งานได้ดียิ่งขึ้น ซึ่งการพัฒนาระบบ ได้มีการออกแบบหน้า User Interface ในการใช้งานดัง Figure 2.



(a) Main menu



(b) Signal prediction camera

(c) Introduction of traffic signs

(d) Traffic sign prediction results

(e) Evaluate satisfaction with the system

Figure 2. Web application user interface

จาก Figure 2. แสดงหน้าส่วนติดต่อผู้ใช้แอปพลิเคชันบนเว็บ ซึ่งประกอบด้วย (a) หน้าหลักของระบบ โดยเมื่อผู้ใช้กดปุ่มเมนู จะสามารถเลือก เมนู Home เพื่อเข้าสู่หน้าการเลือกประเภทของป้ายจราจร และสามารถเลือกใช้งานกล้องเพื่อทำการทำนายป้ายจราจรได้ดังที่ปรากฏในภาพ (b) ประเภทสีป้ายและกล้องทำนายป้าย เมื่อผู้ใช้กดปุ่มเปิดกล้อง ระบบจะนำเข้าสู่หน้าจอสำหรับการทำนายป้ายจราจร โดยเมื่อกล้องจับภาพป้ายจราจร ระบบจะดำเนินการตรวจจับและทำนายประเภทของป้าย พร้อมแสดงผลการทำนายเครื่องหมายจราจร ดังภาพ (d) หรือหากผู้ใช้เลือก เมนู Scan ในหน้า Main menu ระบบจะนำผู้ใช้ไปยังหน้าสำหรับการทำนายป้ายจราจรดังที่แสดงในภาพ (d) เช่นกันโดยระบบจะแสดงผลการทำนายป้ายจราจร ซึ่งประกอบด้วยชื่อป้ายจราจร และค่าความแม่นยำของการทำนาย พร้อมทั้งปุ่มสลับไปยังกล้อง เพื่อจับภาพและทำนายใหม่ ส่วนภาพ (c) แสดงการเลือกประเภทของป้ายจราจร โดยระบบจะแสดงคำแนะนำและรายละเอียดเกี่ยวกับป้ายจราจรในประเภทที่ผู้ใช้เลือกอย่างครบถ้วน และสุดท้ายในกรณีที่ผู้ใช้เลือก เมนู Setting ในหน้า Main menu

Citation:
Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

ระบบจะนำไปยังหน้าสำหรับการประเมินความพึงพอใจที่มีต่อระบบ ซึ่งแสดงในภาพ (e) สำหรับกระบวนการประเมินความพึงพอใจของผู้ใช้งาน เมื่อผู้ใช้กดปุ่ม Rating ระบบจะแสดงแบบฟอร์มให้ผู้ใช้กรอกความคิดเห็นและให้คะแนนความพึงพอใจที่มีต่อระบบ

6. ผลการวิจัย (Results)

ผลการพัฒนาโมเดลสำหรับจำแนกประเภทของป้ายจราจรอัตโนมัติ โดยใช้เครื่องมือ Teachable Machine

ชุดข้อมูลที่ใช้ในระบบตรวจจับและจดจำป้ายจราจรประกอบด้วยภาพป้ายจราจรที่หลากหลายรูปแบบ โดยแบ่งออกเป็น 27 คลาส แต่ละคลาสมีจำนวนตัวอย่างอยู่ที่ 250 – 1251 ภาพ ข้อมูลเหล่านี้ถูกรวบรวมผ่านกระบวนการจัดการและเก็บข้อมูลอย่างเป็นระบบ พร้อมทั้งมีการปรับคุณภาพของภาพ เช่น การตัดขอบและปรับสี เพื่อให้เหมาะสมสำหรับการประมวลผล ผลลัพธ์การทำนายป้ายจราจรถูกวัดด้วย เมทริกซ์ความสับสน (Confusion Matrix) ซึ่งแสดงได้ดัง Figure 3. และตัวชี้วัดประสิทธิภาพต่าง ๆ เช่น Accuracy, Precision, Recall และ F1-Measure แสดงข้อมูลที่วัดได้ดัง Table 2.

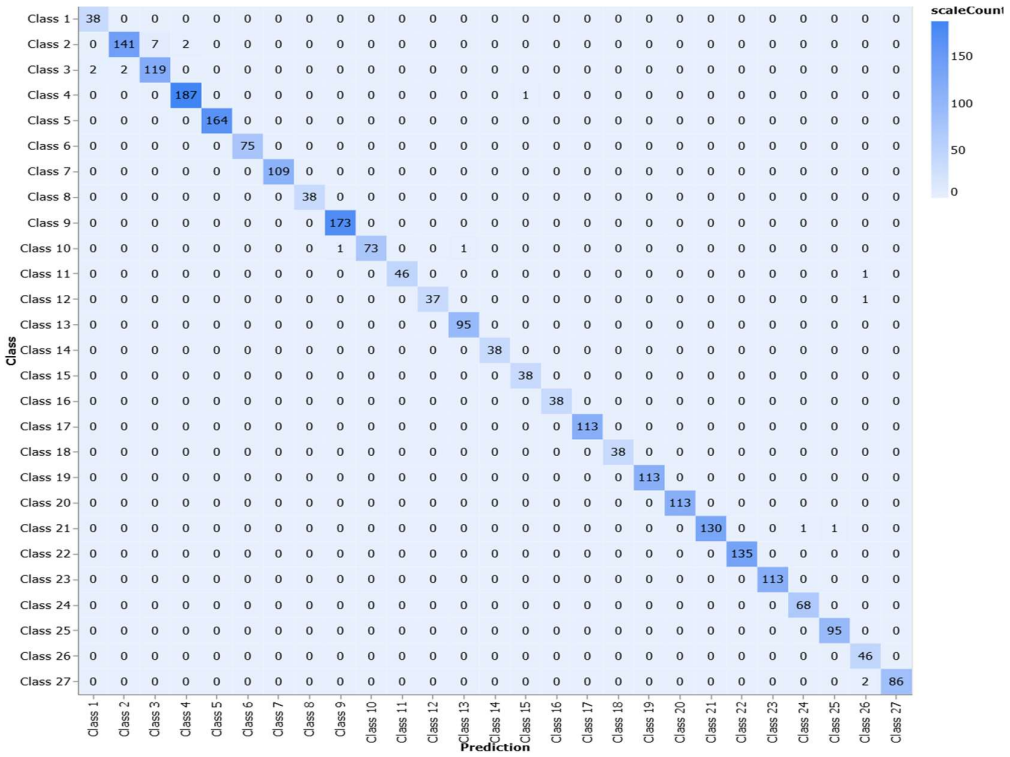


Figure 3. Confusion Matrix

Table 2. Accuracy, Precision, Recall and F1-measure

Class	Class Name	Images	Test Samples	Accuracy	Precision	Recall	F1-Measure
1	Speed limit 30 km./h.	250	38	1.00	0.95	1.00	0.97
2	Speed limit 40 km./h.	1,000	150	0.94	0.99	0.94	0.96
3	Speed limit 60 km./h.	816	123	0.97	0.94	0.97	0.95
4	No right turns	1,251	188	0.99	0.99	0.99	0.99
5	No entry	1,091	164	1.00	1.00	1.00	1.00
6	No stopping	500	75	1.00	1.00	1.00	1.00

Citation: Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

Table 2. Accuracy, Precision, Recall and F1-measure (Cont.)

Class	Class Name	Images	Test Samples	Accuracy	Precision	Recall	F1-Measure
7	Let incoming traffic go first	721	109	1.00	1.00	1.00	1.00
8	Stop	250	38	1.00	1.00	1.00	1.00
9	Turn left	1,151	173	1.00	0.99	1.00	0.99
10	Turn right	500	75	0.97	1.00	0.97	0.99
11	Keep left	311	47	0.98	1.00	0.98	0.99
12	U-turn	250	38	0.97	1.00	0.97	0.99
13	One way	630	95	1.00	0.99	1.00	0.99
14	Proceed straight or turn left	253	38	1.00	1.00	1.00	1.00
15	Pedestrian crossing	250	38	1.00	0.97	1.00	0.99
16	Bicycle lane	250	38	1.00	1.00	1.00	1.00
17	Right curve	750	113	1.00	1.00	1.00	1.00
18	Beware of bouncing vehicles	250	38	1.00	1.00	1.00	1.00
19	Main roads intersect in a T shape	750	113	1.00	1.00	1.00	1.00
20	Secondary roads intersect main roads	750	113	1.00	1.00	1.00	1.00
21	Roundabout	877	132	0.98	1.00	0.98	0.99
22	Go left or right	897	135	1.00	1.00	1.00	1.00
23	Narrow path on the left	750	113	1.00	1.00	1.00	1.00
24	Uncontrolled railway crossing	453	68	1.00	0.98	1.00	0.99
25	Traffic signal	627	95	1.00	0.99	1.00	0.99
26	City limit produce speed	303	46	1.00	0.92	1.00	0.96
27	Reduce speed	586	88	0.98	1.00	0.98	0.99
			Mean	**98.93%**	**98.89%**	**98.93%**	**98.90%**

จาก Table 2. แสดงผลการประเมินประสิทธิภาพของโมเดลสำหรับการจำแนกป้ายจราจร 27 คลาส โดยมีรายละเอียดข้อมูลของแต่ละคลาส ดังนี้

Class Name: แสดงชื่อของป้ายจราจรที่จำแนก รวมทั้งหมด 27 คลาส เช่น “Speed limit 30 km./h.”, “No entry”, และ “Stop” เป็นต้น ส่วน Images: แสดงจำนวนภาพทั้งหมดของแต่ละคลาสที่นำมาใช้ในกระบวนการฝึกและทดสอบโมเดล และ Test Samples: แสดงจำนวนตัวอย่างในชุดข้อมูลทดสอบ (Test Set) ของแต่ละคลาส โดยได้จากการแบ่งข้อมูลทั้งหมดเป็น 85% สำหรับฝึก (Training) และ 15% สำหรับทดสอบ (Testing) ตัวอย่างเช่น คลาส “Speed limit 30 km./h.” มีภาพทั้งหมด 250 ภาพ จะมีภาพสำหรับทดสอบ 250 × 0.15 = 37.5 ≈ 38 ตัวอย่าง ในขณะที่คลาส “Speed limit 40 km./h.” ที่มี 1,000 ภาพ จะมีภาพทดสอบจำนวน 1,000 × 0.15 = 150 ตัวอย่าง

ในส่วนของค่าชี้วัดประสิทธิภาพของโมเดลในแต่ละคลาสประกอบด้วย 4 ตัวชี้วัดหลัก ได้แก่

1) Accuracy วัดอัตราส่วนของตัวอย่างที่จำแนกถูกต้องต่อจำนวนตัวอย่างทั้งหมดของคลาสนั้น โดยคำนวณจากสมการ (1) หรือสมการ (2) ดังนี้

Citation:
Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

$$Accuracy_{class} = True\ Positive / Total\ Samples\ of\ That\ Class \quad (1)$$

$$Accuracy_{class} = \text{ค่าบนแนวทแยงของ Confusion Matrix / ผลรวมของแถวในคลาสนั้น} \quad (2)$$

2) Precision วัดความแม่นยำของการจำแนกว่าคำตอบที่โมเดลคาดการณ์ว่าเป็นคลาสนั้นถูกต้องมากน้อยเพียงใด โดยคำนวณจากสมการ (3) หรือสมการ (4) ดังนี้

$$Precision_{class} = True\ Positive / True\ Positive + False\ Positive \quad (3)$$

$$Precision_{class} = \text{ค่าบนแนวทแยงของ Confusion Matrix / ผลรวมของคอลัมน์ในคลาสนั้น} \quad (4)$$

3) Recall วัดความสามารถของโมเดลในการดึงข้อมูลของคลาสนั้นว่าออกมาได้ครบถ้วนหรือไม่ โดยคำนวณจากสมการ (5) หรือสมการ (6) ดังนี้

$$Recall_{class} = True\ Positive / True\ Positive + False\ Negative \quad (5)$$

$$Recall_{class} = \text{ค่าบนแนวทแยงของ Confusion Matrix / ผลรวมของแถวในคลาสนั้น} \quad (6)$$

4) F1-Measure: เป็นค่าที่ผสาน Precision และ Recall เข้าด้วยกัน เพื่อให้เห็นสมดุลของทั้งสองค่า โดยคำนวณจากสมการ (7) ดังนี้

$$F1_{class} = (2 \times Precision \times Recall) / (Precision + Recall) \quad (7)$$

ค่าทั้งหมดข้างต้นถูกคำนวณจากผลลัพธ์ของ Confusion Matrix ขนาด 27 × 27 ซึ่งแสดงการจำแนกผลระหว่างคลาสที่แท้จริงและคลาสที่โมเดลคาดการณ์ การแสดงผลในลักษณะนี้ทำให้สามารถวิเคราะห์จุดแข็ง จุดอ่อน และความสับสนระหว่างคลาสได้อย่างละเอียด นอกจากนี้ เพื่อให้เข้าใจพฤติกรรมของโมเดลตลอดช่วงการฝึก งานวิจัยนี้ได้นำเสนอ Figure 4. และ Figure 5. เพื่อแสดงการเปลี่ยนแปลงของค่า Accuracy per Epoch และ Loss per Epoch ตามลำดับ โดย Accuracy per Epoch หมายถึง เปอร์เซ็นต์ของตัวอย่างในชุดฝึกที่โมเดลสามารถจำแนกได้ถูกต้องในแต่ละรอบการฝึก (Epoch) หากจำแนกถูก 70 จาก 100 ตัวอย่าง ค่า Accuracy = 70/100 = 0.70 ส่วน Loss per Epoch แสดงค่าความผิดพลาดของการคาดการณ์ในแต่ละรอบการฝึก โดยค่า Loss ที่ต่ำแสดงว่าโมเดลเรียนรู้ได้ดีและสามารถจำแนกข้อมูลได้อย่างถูกต้องใกล้เคียงกับค่าจริง ทั้งนี้ ค่าทั้งหมดที่แสดงใน Table 2 สะท้อนถึงความแม่นยำของโมเดลในระดับคลาส ทำให้สามารถวิเคราะห์เชิงลึกได้ว่าโมเดลมีประสิทธิภาพในการจำแนกป้ายจราจรประเภทใดมากหรือน้อยเป็นพิเศษ ซึ่งเป็นข้อมูลสำคัญสำหรับการปรับปรุงในระยะต่อไป

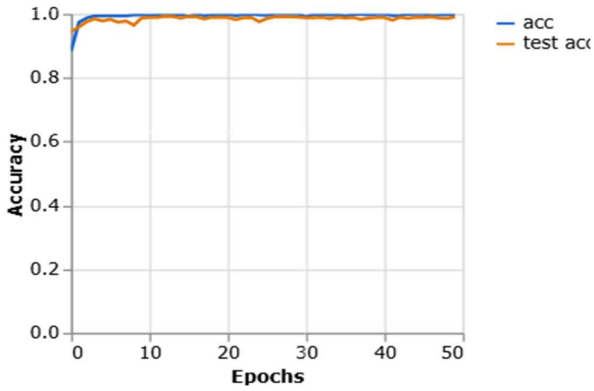


Figure 4. Accuracy per Epoch

จาก Figure 4. แสดงผลการฝึกโมเดลผ่านกราฟแสดงค่าความแม่นยำ (Accuracy) ต่อรอบการฝึก (Epoch) โดยมีแกนนอน (แกน X) แสดงจำนวนรอบการฝึกตั้งแต่ 1 ถึง 50 Epochs และแกนตั้ง (แกน Y) แสดงค่าความแม่นยำในช่วง 0 ถึง 1 ซึ่งเทียบเท่ากับ 0% ถึง 100% ในกราฟแสดงเส้นค่า Training Accuracy (acc) และ Testing Accuracy (test acc) เพื่อเปรียบเทียบประสิทธิภาพของโมเดลทั้งในชุดข้อมูลฝึกและข้อมูลทดสอบ จากรูปจะเห็นได้ว่าค่าความแม่นยำของชุดข้อมูลฝึก (acc) เพิ่มขึ้นอย่างต่อเนื่องในลักษณะเสถียร โดยเริ่มต้นจากค่าที่อยู่ระดับปานกลางและเพิ่มสูงขึ้นอย่างรวดเร็วในช่วง Epochs แรก ๆ ก่อนจะเริ่มมีแนวโน้มคงที่และเข้าใกล้ค่าสูงสุดที่ 1.0 ในช่วงท้ายของการฝึก ซึ่งสะท้อนว่าโมเดล

Citation: Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

สามารถเรียนรู้ลักษณะของข้อมูลฝึกได้อย่างมีประสิทธิภาพ สำหรับเส้นค่าความแม่นยำในชุดข้อมูลทดสอบ (test acc) พบว่ามีแนวโน้มเพิ่มขึ้นสอดคล้องกับชุดฝึก โดยไม่มีการแกว่งหรือลดลงอย่างมีนัยสำคัญ ซึ่งแสดงถึงความสม่ำเสมอของการเรียนรู้และบ่งชี้ว่าโมเดลไม่เกิดภาวะ Overfitting ตลอดช่วงการฝึก ทั้งนี้ลักษณะกราฟที่เข้าใกล้จุดอิ่มตัวในช่วงท้ายเป็นข้อบ่งชี้ว่าโมเดลสามารถเข้าถึงศักยภาพสูงสุดในการจำแนกข้อมูลได้ภายในจำนวน Epoch ที่กำหนด (50 Epochs)

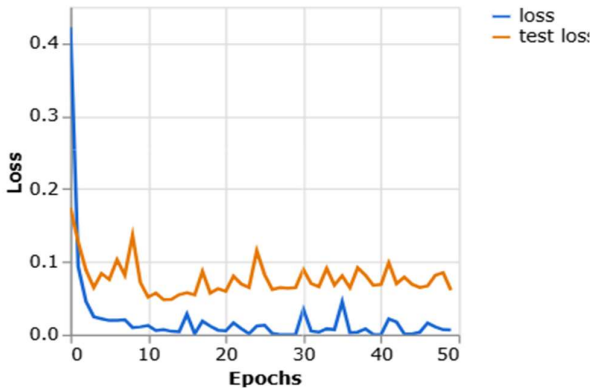


Figure 5. Loss per Epoch

จาก Figure 5. แสดงผลการฝึกโมเดลผ่านกราฟแสดงค่าความสูญเสีย (Loss) ต่อรอบการฝึก (Epoch) โดยมีแกนนอน (แกน X) แสดงจำนวน Epoch ตั้งแต่ 1 ถึง 50 และแกนตั้ง (แกน Y) แสดงค่าความสูญเสียที่เกิดขึ้นระหว่างการฝึก ซึ่งเป็นตัวชี้วัดระดับความคลาดเคลื่อนระหว่างค่าที่โมเดลคาดการณ์กับค่าจริงในชุดข้อมูล กราฟประกอบด้วยเส้นแสดงค่า Training Loss (loss) และ Testing Loss (test loss) เพื่อเปรียบเทียบความสามารถของโมเดลในการเรียนรู้ข้อมูลฝึกและการคาดการณ์ผลในข้อมูลทดสอบที่โมเดลไม่เคยเห็นมาก่อน จากรูปจะเห็นได้ว่าค่าความสูญเสียของชุดฝึก (loss) มีแนวโน้มลดลงอย่างต่อเนื่องในช่วงต้นของการฝึก ซึ่งเป็นสัญญาณว่าโมเดลสามารถปรับค่าน้ำหนักได้อย่างมีประสิทธิภาพและเรียนรู้รูปแบบของข้อมูลได้ดีขึ้นตามลำดับ สำหรับค่าความสูญเสียในชุดทดสอบ (test loss) พบว่ามีแนวโน้มลดลงในลักษณะคล้ายคลึงกับชุดฝึก โดยไม่มีการเพิ่มขึ้นในช่วงท้ายของการฝึก ซึ่งบ่งชี้ว่าโมเดลมีความสามารถในการเรียนรู้ที่สอดคล้องกับข้อมูลใหม่ และไม่เกิดภาวะ Overfitting ซึ่งอาจปรากฏหากค่าความสูญเสียในชุดทดสอบเพิ่มขึ้นในขณะที่ค่าชุดฝึกลดลง โดยรวมกราฟใน Figure 5. สะท้อนพฤติกรรมการเรียนรู้ของโมเดลที่มีประสิทธิภาพ มีเสถียรภาพในการฝึกและสามารถลดความผิดพลาดได้อย่างต่อเนื่องตลอดกระบวนการฝึก 50 Epochs ซึ่งสนับสนุนความเหมาะสมของพารามิเตอร์ที่กำหนด และแสดงให้เห็นว่าโมเดลมีความสามารถในการจำแนกป้ายจราจรได้อย่างแม่นยำในสภาวะการใช้งานจริง

7. สรุปและอภิปรายผลการวิจัย (Conclusion and Discussion)

7.1 สรุปผลการวิจัย

การวิจัยครั้งนี้มีวัตถุประสงค์เพื่อพัฒนาโมเดลการเรียนรู้เชิงลึกสำหรับจำแนกประเภทของป้ายจราจรโดยใช้แพลตฟอร์ม Teachable Machine ซึ่งเป็นเครื่องมือที่สามารถเข้าถึงได้ง่าย โดยได้ออกแบบกระบวนการเก็บข้อมูล แบ่งชุดข้อมูล เพื่อประเมินประสิทธิภาพของระบบ และฝึกโมเดลด้วยพารามิเตอร์พื้นฐาน ได้แก่ ขนาดกลุ่มเท่ากับ 16 จำนวนรอบการฝึก 50 รอบ และอัตราการเรียนรู้ 0.001 ซึ่งเป็นค่าที่เหมาะสมสำหรับการฝึกโมเดลในระดับผู้ใช้ทั่วไป โดยสามารถประมวลผลได้เร็ว ไม่ใช้ทรัพยากรมาก และให้ผลลัพธ์ที่มีความแม่นยำเพียงพอในบริบทของการใช้งานจริง

ผลการทดลองแสดงให้เห็นว่าโมเดลสามารถจำแนกป้ายจราจรได้อย่างแม่นยำ โดยมีค่าเฉลี่ยของ Accuracy, Precision, Recall และ F1-Measure อยู่ที่ 98.93%, 98.89%, 98.93% และ 98.90% ตามลำดับ นอกจากนี้ การแสดงผลผ่าน Confusion Matrix (Figure 3.) ยังสะท้อนถึงความสามารถของโมเดลในการแยกแยะลักษณะเฉพาะของแต่ละคลาสได้อย่างมีประสิทธิภาพ ซึ่งสอดคล้องกับผลการแสดงตามที่ปรากฏใน Table 2. ที่แสดงให้เห็นว่าคลาสส่วนใหญ่มีค่าสถิติดังกล่าวอยู่ในระดับสูงอย่างต่อเนื่อง โมเดลสามารถจำแนกป้ายจราจรที่มีลักษณะแตกต่างกันได้อย่างถูกต้อง โดยเฉพาะ

Citation:
Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

คลาสที่มีภาพตัวอย่างเพียงพอและมีลักษณะเด่นชัด เช่น “Stop” และ “No entry” ซึ่งให้ค่า Precision และ Recall สูงถึง 1.00 อย่างไรก็ตาม โมเดลยังมีแนวโน้มที่จะเกิดความสับสนในคลาสที่มีลักษณะใกล้เคียงกัน เช่น ป้ายจำกัดความเร็วในช่วง 30–60 กิโลเมตรต่อชั่วโมง ซึ่งส่งผลให้ค่า Precision หรือ Recall ของบางคลาสต่ำกว่าเกณฑ์เฉลี่ยเล็กน้อย ส่วนการเปลี่ยนแปลงของค่าความแม่นยำ (Accuracy per Epoch) และค่าความสูญเสีย (Loss per Epoch) ที่ปรากฏใน Figure 4. และ Figure 5. แสดงแนวโน้มที่ชัดเจนถึงกระบวนการเรียนรู้ที่เสถียร โดย Accuracy เพิ่มขึ้นอย่างต่อเนื่องจนเข้าสู่ภาวะคงที่ในช่วง Epoch ที่ 45–50 ขณะที่ค่า Loss แสดงการลดลงของค่าความสูญเสียอย่างต่อเนื่องในระหว่างการฝึก ซึ่งบ่งชี้ว่าโมเดลสามารถปรับค่าพารามิเตอร์ให้ลดความคลาดเคลื่อนระหว่างค่าคาดการณ์กับค่าจริงได้อย่างมีประสิทธิผลภายในช่วงการฝึกเพียง 50 รอบ โดยเฉพาะในช่วง Epoch แรก ๆ ก่อนที่จะเข้าสู่ภาวะเสถียรในช่วงท้าย

7.2 อภิปรายผลการวิจัย

จากผลการวิเคราะห์ เมื่อพิจารณาร่วมกันระหว่าง Figure 4. (Accuracy per Epoch) และ Figure 5. (Loss per Epoch) จะพบว่าโมเดลมีพฤติกรรมการเรียนรู้ที่สอดคล้องและแสดงแนวโน้มเชิงบวกอย่างมีนัยสำคัญตลอดกระบวนการฝึก 50 รอบ (Epochs) โดยค่าความแม่นยำในชุดข้อมูลฝึกและทดสอบ (Training Accuracy และ Testing Accuracy) มีแนวโน้มเพิ่มขึ้นอย่างต่อเนื่อง ในขณะที่ค่าความสูญเสีย (Loss) ในทั้งสองชุดข้อมูลมีแนวโน้มลดลงอย่างมั่นคง ลักษณะของกราฟแสดงว่าโมเดลสามารถเรียนรู้ข้อมูลได้อย่างมีประสิทธิภาพ โดยไม่มีสัญญาณของภาวะ Overfitting หรือ Underfitting ซึ่งมักปรากฏในรูปของความแตกต่างอย่างมีนัยสำคัญระหว่างค่าความแม่นยำของชุดฝึกและชุดทดสอบ หรือการแยกทิศทางของกราฟ Accuracy และ Loss ในช่วงท้ายของการฝึก

การที่กราฟ Accuracy เข้าใกล้ค่าสูงสุด (1.0) และกราฟ Loss ลดลงสู่ค่าที่ต่ำอย่างต่อเนื่อง สะท้อนว่าโมเดลสามารถสร้างฟังก์ชันการเรียนรู้ที่เหมาะสมกับข้อมูล และมีความสามารถทั่วไป (Generalization) ที่ดีต่อข้อมูลใหม่ที่ไม่เคยเห็นมาก่อน นอกจากนี้ การเปลี่ยนแปลงของกราฟทั้งสองยังแสดงถึง ประสิทธิภาพของพารามิเตอร์การฝึก เช่น จำนวน Epochs, ค่า Learning Rate และขนาด Batch Size ซึ่งถูกกำหนดให้เหมาะสมกับลักษณะของข้อมูลและเครื่องมือที่ใช้ (Teachable Machine) ซึ่งผลการวิเคราะห์ดังกล่าวสนับสนุนความสำเร็จของการใช้ Teachable Machine ในการพัฒนาโมเดลจำแนกป้ายจราจร โดยแสดงให้เห็นว่าการฝึกโมเดลสามารถดำเนินไปได้อย่างมีเสถียรภาพ และบรรลุผลการเรียนรู้ที่มีคุณภาพภายในจำนวน Epoch ที่เหมาะสม โดยไม่ต้องใช้กระบวนการปรับจูนที่ซับซ้อน

อย่างไรก็ตาม แม้ภาพรวมของประสิทธิภาพโมเดลจะอยู่ในระดับดีมาก แต่ยังพบข้อจำกัดบางประการที่ส่งผลต่อความแม่นยำในบางคลาส โดยเฉพาะคลาสที่มีจำนวนภาพน้อยหรือมีลักษณะคล้ายกับคลาสอื่น เช่น กลุ่มป้ายจำกัดความเร็วที่มีความแตกต่างกันเพียงตัวเลข ส่งผลให้เกิดการทำนายผิดพลาดระหว่างคลาสที่มีลักษณะใกล้เคียงกัน ซึ่งสามารถสังเกตได้จากตำแหน่งนอกแนวทแยง (Off-diagonal) ใน Confusion Matrix สาเหตุหลักของปัญหาดังกล่าวมาจากความไม่สมดุลของข้อมูลในแต่ละคลาส ซึ่งบางคลาสมีภาพจำนวนมาก ขณะที่บางคลาสมีจำนวนน้อยกว่า 300 ภาพ ส่งผลให้โมเดลให้ความสำคัญกับคลาสที่มีข้อมูลมากเกินไป และมีโอกาสละเลยคลาสที่มีข้อมูลจำกัด

8. ข้อเสนอแนะงานวิจัย (Recommendation)

งานวิจัยฉบับนี้แม้จะประสบความสำเร็จในการพัฒนาโมเดลจำแนกป้ายจราจรที่ให้ผลลัพธ์แม่นยำในระดับสูง แต่ยังคงมีข้อจำกัดที่ควรได้รับการพิจารณาและแก้ไขในงานวิจัยถัดไป โดยเฉพาะอย่างยิ่งในประเด็นเรื่อง ความไม่สมดุลของจำนวนข้อมูลในแต่ละคลาส (Class Imbalance) ปัญหาดังกล่าวส่งผลต่อค่า Precision และ Recall ของบางคลาสที่มีลักษณะใกล้เคียงกันหรือมีจำนวนภาพจำกัด ซึ่งอาจทำให้เกิดการทำนายผิดและลดความแม่นยำในสถานการณ์ใช้งานจริง เพื่อเพิ่มประสิทธิภาพของโมเดลและลดผลกระทบจากปัญหา Class Imbalance จึงขอเสนอแนวทางในการพัฒนาต่อไปดังนี้

1) การจัดสมดุลจำนวนข้อมูลในแต่ละคลาส ควรเตรียมชุดข้อมูลให้มีจำนวนภาพในแต่ละคลาสใกล้เคียงกัน เช่น 300–500 ภาพต่อคลาส เพื่อลดความลำเอียงของโมเดล และเพิ่มโอกาสในการเรียนรู้ของคลาสที่มีข้อมูลน้อย

2) การใช้เทคนิค Data Augmentation สำหรับคลาสที่มีจำนวนภาพไม่เพียงพอ ควรประยุกต์ใช้เทคนิคเพิ่มข้อมูล เช่น การหมุนภาพ พลิกภาพ ปรับแสง และการขยายภาพ เพื่อเพิ่มความหลากหลายและปริมาณของข้อมูลโดยไม่ต้องเก็บข้อมูลเพิ่มเติม

Citation:
Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

3) การพิจารณาเลือกใช้เครื่องมือที่สามารถกำหนด Class Weight ได้ ซึ่ง Teachable Machine จะไม่เปิดให้กำหนดน้ำหนักของแต่ละคลาสได้ แต่หากย้ายไปใช้แพลตฟอร์มเช่น TensorFlow หรือ PyTorch นักวิจัยสามารถปรับค่าถ่วงน้ำหนักให้กับคลาสที่มีข้อมูลน้อยได้ เพื่อให้โมเดลให้ความสำคัญกับทุกคลาสอย่างสมดุลมากขึ้น

4) การประเมินผลเพิ่มเติมด้วยชุดข้อมูลภายนอกที่สมดุล เพื่อความน่าเชื่อถือของโมเดล ควรประเมินผลด้วยชุดข้อมูลทดสอบที่ออกแบบให้แต่ละคลาสมีจำนวนภาพเท่า ๆ กัน เพื่อหลีกเลี่ยงผลลัพธ์ที่ถูก Bias จากข้อมูลเดิม

โดยสรุป แม้งานวิจัยนี้จะประสบผลสำเร็จในการพัฒนาโมเดลที่มีประสิทธิภาพสูงสำหรับการจำแนกป้ายจราจร แต่การจัดการกับความไม่สมดุลของข้อมูลและการเพิ่มความหลากหลายของชุดข้อมูลจะเป็นปัจจัยสำคัญที่ช่วยยกระดับความสามารถของโมเดลในบริบทของการใช้งานจริงในอนาคต

9. เอกสารอ้างอิง (References)

Alsiyeu, U., & Duisebekov, Z. (2024). *Enhancing Traffic Sign Recognition with Tailored Data Augmentation: Addressing Class Imbalance and Instance Scarcity*. arXiv. https://doi.org/10.48550/ARXIV.2406.03576.

Chen, A., Pitaru, A., Webster, B., Alvarado, I., Griffith, J., Phillips, K., Carney, M., & Howell, N. (2020). *Teachable Machine: Approachable Web-Based Tool for Exploring Machine Learning Classification*. https://research.google/pubs/teachable-machine-approachable-web-based-tool-for-exploring-machine-learning-classification.

Dhawan, K., Srinivasa Perumal, R., & Nadesh, R. K. (2023). Identification of Traffic Signs for Advanced Driving Assistance Systems in Smart Cities Using Deep Learning. *Multimedia Tools and Applications*, *82*, 26465-26480. https://doi.org/10.1007/s11042-023-14823-1.

Dineley, A., Natalia, F., & Sudirman, S. (2024). Data Augmentation for Occlusion-Robust Traffic Sign Recognition Using Deep Learning. *ICIC Express Letters, Part B: Applications, 15*(4), 381-388. https://doi.org/10.24507/icicelb.15.04.381.

Ge, J. (2023). *Traffic Sign Recognition Dataset and Data Augmentation*. arXiv. https://doi.org/10.48550/ARXIV.2303.18037.

Google Creative Lab. (2020). *Teachable Machine Community Library (Image)*. GitHub. https://github.com/googlecreativelab/teachablemachine-community/blob/master/libraries/image/src/teachable-mobilenet.ts.

Jantaragate, T. (2024). *Deep Convolutional Neural Network for Speed Limit Signs Classification*. [Master's dissertation, Srinakharinwirot University]. DSpace at Srinakharinwirot University. http://ir-ithesis.swu.ac.th/dspace/handle/123456789/3127. (In Thai)

Jeong, H. (2020). Feasibility Study of Google's Teachable Machine in Diagnosis of Tooth-Marked Tongue. *Journal of Dental Hygiene Science*, *20*(4), 206–212. https://doi.org/10.17135/jdhs.2020.20.4.206.

Khunarsa, T., Sahoh, A., & Pubphasomtrakool, P. (2021, June 22). Traffic Signs Detection and Recognition. *The 1st Rajamangala University of Technology Suvarnabhumi National Conference*, 417-426. Rajamangala University of Technology Suvarnabhumi. (In Thai)

Kingma, D. P., & Ba, J. (2014). *Adam: A Method for Stochastic Optimization*. arXiv. https://doi.org/10.48550/ARXIV.1412.6980.

Kozhamkulova, Z., Bidakhmet, Z., Vorogushina, M., Tashenova, Z., Tussupova, B., Nurlybaeva, E., & Kambarov, D. (2024). Development of Deep Learning Models for Traffic Sign Recognition in Autonomous Vehicles. *International Journal of Advanced Computer Science and Applications*, *15*(5), 913-920. https://doi.org/10.14569/IJACSA.2024.0150593.

Citation:
Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.

Kurz, T. L., Jayasuriya, S., Swisher, K., Mativo, J., Pidaparti, R., & Robinson, D. T. (2024). The Impact of Teachable Machine on Middle School Teachers' Perceptions of Science Lessons after Professional Development. *Education Sciences*, *14*(4), 417. https://doi.org/10.3390/educsci14040417.

Stallkamp, J., Schlipsing, M., Salmen, J., & Igel, C. (2011). *Dataset*. Institut für Neuroinformatik. https://benchmark.ini.rub.de/gtsrb_dataset.html.

Thipsanthia, P. (2021). *Improvement of the Performance of Traffic Signs Detection*. [Doctoral's dissertation, Mahasarakham University]. DSpace at Mahasarakham University. http://202.28.34.124/dspace/handle123456789/2162. (In Thai)

Wong, J. J. N., & Fadzly, N. (2022). Development of Species Recognition Models using Google Teachable Machine on Shorebirds and Waterbirds. *Journal of Taibah University for Science*, *16*(1), 1096-1111. https://doi.org/10.1080/16583655.2022.2143627.

Yim, I. H. Y., & Su, J. (2025). Artificial Intelligence (AI) Learning Tools in K-12 Education: A Scoping Review. *Journal of Computers in Education, 12*(1), 93–131. https://doi.org/10.1007/s40692-023-00304-9.

Zhu, Y., & Yan, W. Q. (2022). Traffic Sign Recognition Based on Deep Learning. *Multimedia Tools and Applications*, *81*(13), 17779–17791. https://doi.org/10.1007/s11042-022-12163-0.

Citation:
Pirom, T., Hamniam, P., Thapatsuwan, W., & Thapatsuwan, P. (2025). Development of a Web-Based Application for Automatic Traffic Sign Detection and Recognition Using Teachable Machine. *Journal of Computer and Creative Technology, 3*(2), 169-184. https://doi.org/10.14456/jcct.2025.14.