Journal of Applied Informatics and Computing (JAIC)
Vol.9, No.4, August 2025, pp. 1363~1373
e-ISSN: 2548-6861

YOLOv11-Based Detection of Indonesian Traffic Signs: Transfer Learning vs. From-Scratch Training

Ibnu Cipta Ramadhan [1]*, Akhmad Hendriawan [2]*, Hary Oktavianto [3]*
* Teknik Elektronika, Politeknik Elektronika Negeri Surabaya
ibnu.ramadhan@pens.ac.id [1], hendri@pens.ac.id [2], hary@pens.ac.id [3]

Article Info

Article history:

Received 2025-06-02
Revised 2025-06-25
Accepted 2025-07-10

Keyword:

Traffic Sign Detection,
Transfer Learning,
YOLOv11,
Deep Learning,
Indonesian Traffic Dataset.

ABSTRACT

Traffic sign detection is a fundamental component in intelligent transportation systems (ITS), autonomous driving, and advanced driver assistance systems (ADAS), enabling vehicles to interpret road conditions and enhance safety. Developing robust traffic sign detection models for specific regions requires high-quality, well-annotated local datasets, which are often challenging and costly to create. Even when such datasets are available, training deep learning models from scratch demands substantial computational resources and time. This study compares models trained from scratch and those using transfer learning based on the lightweight YOLOv11s architecture on an Indonesian traffic sign dataset. Evaluations using precision, recall, mean Average Precision at IoU 0.5 (mAP@0.5), and mean Average Precision across IoU thresholds 0.5 to 0.95 (mAP@0.5:0.95) demonstrate that the transfer learning model consistently outperforms the from-scratch model across all metrics. These findings highlight the effectiveness and efficiency of transfer learning for developing accurate and practical traffic sign detection systems adapted to local contexts.

I. INTRODUCTION

Traffic sign detection has become an essential component of modern intelligent transportation systems (ITS), autonomous vehicles, and advanced driver-assistance systems (ADAS). These systems rely on the real-time detection and accurate classification of traffic signs to ensure safe navigation, adherence to traffic regulations, and contextual awareness in dynamic driving environments. As the automotive industry advances toward full autonomy, the ability of machines to perceive and interpret road signs becomes increasingly critical [1], [2]. Any misinterpretation or failure to recognize important traffic signs could lead to serious consequences, including traffic violations or accidents.

The rise of deep learning revolutionized traffic sign detection. Convolutional Neural Networks (CNNs) demonstrated superior performance by automatically learning hierarchical feature representations directly from image data. Object detection frameworks like R-CNN [3], Faster R-CNN [4], SSD [5], and YOLO [6] significantly improved both accuracy and speed. Among them, the YOLO (You Only Look Once) family has become particularly prominent due to its real-time detection capabilities and end-to-end design. The recently released YOLOv11 [7] features an improved backbone and neck architecture for better feature extraction and task performance. It offers faster inference and higher accuracy with 22% fewer parameters than YOLOv8m, making it efficient for real-time applications.

Some state-of-the-art traffic sign detection models are trained using large-scale, publicly available datasets such as the German Traffic Sign Detection Benchmark (GTSDB) [8] or the LISA Traffic Sign Dataset [9]. However, these datasets predominantly represent Western or European traffic environments and may not accurately reflect the visual characteristics of traffic signs in other countries, including Indonesia. Differences in color schemes, sign shapes, iconography, language, and placement standards can all affect a model's ability to generalize. This discrepancy highlights a pressing need to develop and evaluate models tailored specifically for Indonesian road signage.

One of the core challenges in developing localized traffic sign models lies in the limitations of dataset size and variety. Collecting and annotating large-scale datasets is time-consuming and resource-intensive, particularly when accounting for the full range of possible traffic sign conditions (e.g., lighting, occlusion, angle, degradation) [10]. Consequently, training deep learning models from scratch under these constraints often leads to overfitting and suboptimal performance, especially for sign classes with fewer samples. Deep neural networks, particularly convolutional-based architectures used in object detection, typically require vast amounts of labeled data and computational power to achieve generalizable results. Without these, their learning capacity is underutilized, and their predictions become unreliable in real-world deployments [11].

Given these limitations, transfer learning emerges as a compelling solution. By leveraging knowledge from models pretrained on large and diverse datasets, such as COCO [12] or ImageNet [13], transfer learning allows researchers to fine-tune existing models on smaller, domain-specific datasets. This approach can significantly reduce training time, improve model convergence, and boost accuracy even in data-scarce conditions [14]. Applying transfer learning to the Indonesian traffic sign dataset could not only overcome the generalization issues faced by models trained from scratch but also contribute to the broader goal of building robust, locally adapted ITS solutions.

Several studies have addressed traffic sign detection in the Indonesian context. Traditional approaches utilizing handcrafted features, such as color and texture extraction combined with Support Vector Machine (SVM) classifiers, have demonstrated high precision and recall rates in detecting and recognizing traffic signs [15]. However, these methods often lack robustness and scalability when faced with diverse environmental conditions.

The emergence of deep learning has significantly advanced traffic sign detection. Convolutional Neural Networks (CNNs) have been employed to enhance classification accuracy, with reported performance reaching accuracy scores of approximately 93% and average F1-scores around 94% for Indonesian traffic signs [16]. These models benefit from automatic feature extraction and greater adaptability but require sufficient training data to generalize well.

To address the challenge of detecting smaller and distant traffic signs, recent work has proposed an enhanced Single-Shot Detector (SSD) tailored for Indonesian traffic signs. This approach improves mean average precision (mAP) by 2.52%, reaching a mAP of 97.87%, demonstrating the effectiveness of specialized object detectors for this domain [17]. Such improvements are critical for real-time traffic sign detection in dynamic environments.

Advancements in one-stage object detectors, especially the YOLO family, have also been explored. Studies using YOLOv4 on Indonesian datasets have shown promising results for Advanced Driver Assistance Systems (ADAS), balancing detection speed and accuracy [18]. More recent work applying YOLOv8 reported mAP values exceeding 99%, confirming its capability for highly accurate real-time detection in Indonesia's traffic contexts [19].

Despite these advances, there is a noticeable lack of studies comparing the effectiveness of training models from scratch versus utilizing transfer learning on localized datasets like Indonesian traffic signs. Transfer learning is known to reduce training time and enhance model performance on limited datasets, yet its specific impact in this application domain remains underexplored

This work aims to fill this gap by systematically comparing transfer learning and training from scratch approaches using the lightweight YOLOv11s architecture on Indonesian traffic sign data. The study evaluates model accuracy, confidence levels, and training efficiency under small-scale dataset constraints. The findings provide valuable insights for deploying optimized object detection models in intelligent transportation systems and ADAS tailored for Indonesia.

II. RESEARCH METHOD

This study explores the performance of transfer learning using the YOLOv11 [7] model in detecting Indonesian traffic signs. We compare two training strategies: (1) fine-tuning a YOLOv11 model pretrained on a general object detection dataset, and (2) training a YOLOv11 model from scratch using only the Indonesian traffic sign dataset. The goal is to assess the effectiveness of transfer learning in a localized traffic environment. This section outlines the dataset utilized in the experiments, presents exploratory data analysis (EDA) to better understand the dataset characteristics, describes the model configuration and training setup, and explains the evaluation metrics used to assess model performance.

A. Dataset

This study utilizes two datasets: one for training and evaluating the traffic sign detection models, and another serving as the source of pretraining for transfer learning.

The primary dataset is the Indonesian Traffic Sign dataset [20], obtained from the Roboflow platform and specifically curated to include four common traffic sign classes in Indonesia: *belok kanan* (turn right), *belok kiri* (turn left), *dilarang putar balik* (no U-turn), and *putar balik* (U-turn). While Indonesian roads feature a wider variety of traffic signs, this dataset focuses only on these four to simplify the classification task and provide a controlled comparison.

The dataset comprises a total of 4,044 annotated images, which are divided into three subsets: 3,536 images for training, 339 for validation, and 169 for testing. This corresponds approximately to a split ratio of 87.4% training, 8.4% validation, and 4.2% testing. All images are uniformly sized at 640×640 pixels. Each image includes bounding box annotations identifying instances of the four traffic sign classes.

The secondary dataset is the Microsoft COCO (Common Objects in Context) dataset [12]. It serves as the source for pretraining the YOLOv11 model used in the transfer learning approach. COCO is a large-scale object detection, segmentation, and captioning dataset containing over 200,000 labelled images across more than 80 object categories. The model pretrained on this dataset has already learned general-purpose visual features, which can be fine-tuned to adapt to more specific tasks like Indonesian traffic sign detection. The COCO dataset is not used directly in this study for evaluation but is critical for initializing the weights of the transfer learning model.

B. Exploratory Data Analysis (EDA)

In this study, exploratory data analysis (EDA) was conducted exclusively on the Indonesian traffic sign dataset. This decision is based on the fact that the COCO dataset is only used for pretraining purposes and not involved in model evaluation or direct training on the target task.

The Indonesian Traffic Sign dataset exhibits a relatively balanced class distribution, which is crucial for training a machine learning model that can generalize well across different types of objects. As illustrated in Figure 1, the "Turn left" class appears most frequently with 939 instances, followed closely by "No U-turn" with 908 instances, "U-turn" with 863 instances, and "Turn right" with 856 instances. This distribution suggests that there is no overwhelming dominance of any particular class, which helps prevent the model from being biased toward one class. A balanced dataset ensures that the model is trained to correctly classify each type of action, without overly favouring one class at the expense of others.

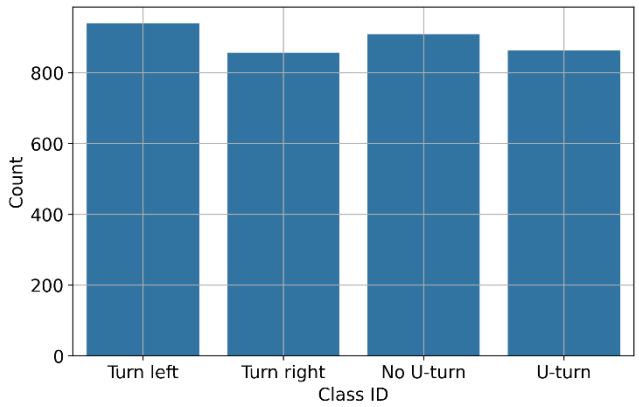

Figure 1. Class Distribution

In terms of bounding box sizes, there is considerable variation in both the width and height of the boxes. The bounding box width is distributed across 20 bins, with the range extending from 58.0 pixels to 640.0 pixels. The data reveals that most bounding boxes have a width in the range of 494.5 to 640.0 pixels, with the highest concentration of boxes falling in the range of 610.9 to 640.0 pixels. This range alone accounts for 1,329 instances, the largest of any bin. This indicates that most of the objects in the dataset are relatively large in width, which could correspond to larger or more prominent objects within the images. Figure 2 illustrates this distribution, highlighting the predominance of wider bounding boxes.

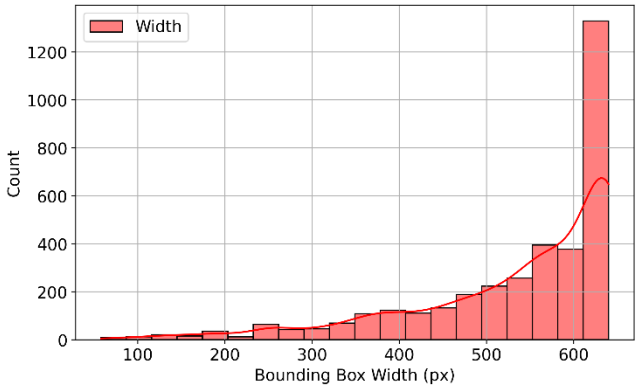

Figure 2. Bounding Box Width Distribution

The bounding box height distribution mirrors this trend, with a range spanning from 62.0 to 640.0 pixels. The largest concentration of bounding boxes falls between the 524.4 and 640.0 pixel range, with the peak being in the 582.2 to 611.1 pixel range, where 422 bounding boxes are located. This suggests that most of the objects in the images are similarly large, but there is also a substantial spread in terms of height, indicating a variety of object sizes within the dataset. The distribution is visualized in Figure 3, showing that taller bounding boxes are also prevalent.

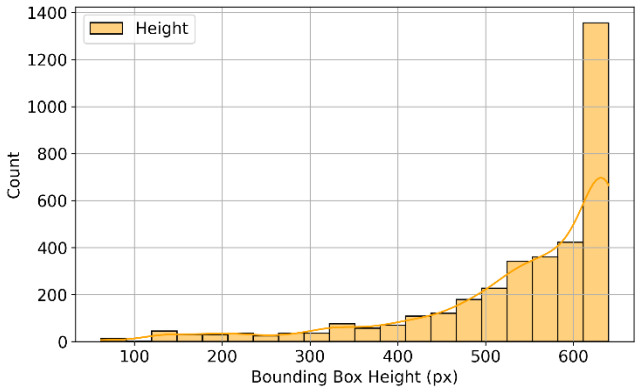

Figure 3. Bounding Box Height Distribution

The area of each bounding box follows a similar pattern to width and height. The distribution of bounding box areas is clustered into 20 bins, with the largest number of bounding boxes occurring in the area range of 288,784.6 to 409,600.0 px². Specifically, the area range of 349,192.3 to 369,328.2 px² accounts for the largest number of occurrences (376 bounding boxes), with the second highest concentration in the range of 329,056.4 to 349,192.3 px² (350 bounding boxes). These larger bounding boxes suggest that the dataset predominantly contains large objects, which may be indicative of the type of

scenes captured in the images, such as vehicles or other large items. The wider distribution of bounding box areas also indicates that the model will need to handle objects of varying sizes, which can be a challenge but also a sign of dataset diversity. This distribution is visualized in Figure 4, which clearly shows the dominance of larger bounding box areas in the dataset.

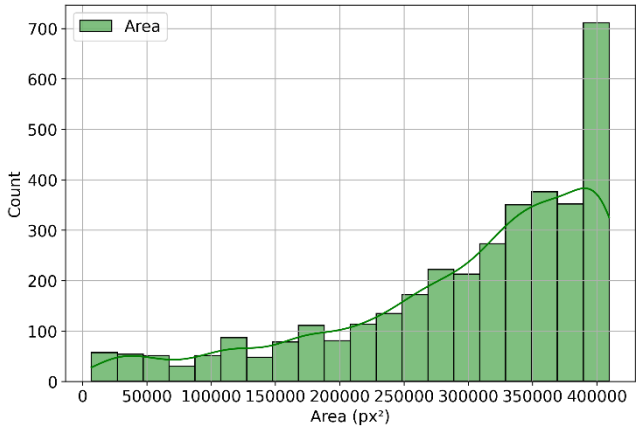

Figure 4. Bounding Box Area Distribution

Looking at the number of bounding boxes per image, 98.98% of the images contain only a single bounding box, while the remaining 1.02% contain two bounding boxes. This means that, in most cases, the images are labelled with a single object, which could be a specific vehicle or action. However, the small percentage of images with multiple bounding boxes suggests that some images may contain more than one object or action, albeit relatively infrequently. The fact that most images contain only one bounding box can help streamline the training process, as the model doesn't need to handle complex scenarios with many overlapping bounding boxes.

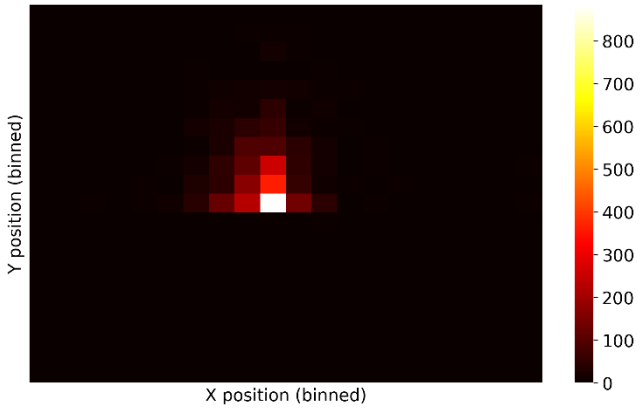

Figure 5. Heatmap of Bounding Box Center

To further understand the spatial distribution of objects within the dataset, a heatmap of the bounding box center coordinates was generated using a 20×20 grid as shown in the Figure 5. The heatmap reveals how frequently bounding boxes appear in different regions of the image frame, providing insight into spatial biases in object placement.

Out of the 400 total bins, 137 contained at least one bounding box center. The most populated bin recorded 885 bounding box centers, suggesting a strong concentration of objects in that specific region. This bin corresponds to the horizontal range of approximately 314.3 to 341.3 pixels and the vertical range of 300.8 to 327.7 pixels, which is near the center of the image. The least populated non-zero bin contains 3 bounding boxes, showing that while some peripheral regions are used, they are less common. This pattern suggests that most objects tend to appear toward the center of the image, which is typical in many visual datasets due to common framing practices.

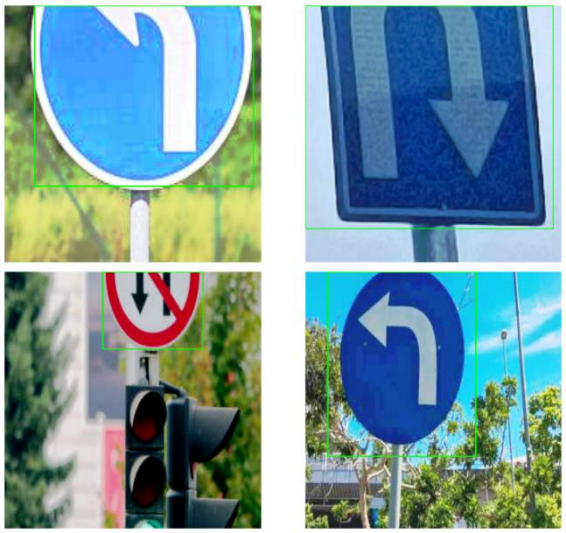

Figure 6. Sample Images and Their Bounding Box

Four sample images as illustrated by figure 6 are provided to demonstrate the variability in bounding boxes within the dataset. Each image showcases a different scenario with the corresponding bounding boxes drawn around the objects of interest. These visual examples highlight the range of object sizes, their different positions within the image, and the accuracy of the annotations. By examining these samples, we can observe the diverse nature of the dataset, which includes objects of varying scales, shapes, and placements.

This diversity not only enriches the dataset but also reflects the types of real-world scenarios the model will encounter. The bounding boxes in these images are crucial, as they help guide the model’s attention to the relevant regions, ensuring it focuses on the right areas during training. The variation in the bounding boxes, from smaller objects to larger ones, provides a comprehensive representation of how the model will handle diverse object sizes and placements. This will contribute to better generalization during the model's deployment, as it will be trained to detect objects in various contexts, enhancing its ability to work with unseen data.

TABLE I
SUMMARY OF EDA ON THE INDONESIAN TRAFFIC SIGN DATASET

Feature Analyzed	Key Findings
Class Distribution	Relatively balanced. Most frequent: "Turn left" (939), followed by "No U-turn" (908), "U-turn" (863), and "Turn right" (856).
Bounding Box Width	Ranges from 58 to 640 px. Most boxes fall in the 610.9–640.0 px range (1,329 instances).
Bounding Box Height	Ranges from 62 to 640 px. Highest concentration in the 582.2–611.1 px range (422 instances).
Bounding Box Area	Most common area range: 349,192–369,328 px² (376 instances). Dataset dominated by large objects.
Bounding Boxes per Image	98.98% of images contain a single bounding box. Only 1.02% contain two.
Object Position in Image	137/400 bins populated. Highest concentration (885 boxes) near the center of the image.

To complement the detailed descriptions above, a summary of the key findings from the exploratory data analysis is presented in Table 1. This table provides a concise overview of the dataset characteristics, including class distribution, bounding box dimensions, object locations, and image-level statistics. It is intended to give readers a quick reference to the most relevant insights that inform subsequent model training and evaluation.

C. Model and Training Setup

For this study, five object detection models were trained using the YOLOv11s architecture, which is a lightweight version of the recently released YOLOv11 series developed by Ultralytics [7]. YOLOv11 builds upon the success of previous YOLO (You Only Look Once) [6] versions. The YOLO family of models is known for its ability to perform real-time object detection with a single forward pass through a convolutional neural network, which balances detection accuracy and speed effectively. Transfer learning has been shown to be effective when labelled data is limited in the target domain [11], [21], which aligns with our use case involving a small, domain-specific dataset, making YOLOv11s a suitable choice for leveraging pretrained weights in a resource-constrained setting.

In our experiments, we evaluated two different training strategies (1) transfer learning models, trained for 10, 20, and 30 epochs using pretrained weights from the COCO dataset. (2) From-scratch models, trained for 30 and 60 epochs from randomly initialized weights (no pretraining).

The choice to include from-scratch models was motivated by the need to assess how well YOLOv11s can adapt to a small, domain-specific dataset—such as Indonesian traffic signs—without the benefit of large-scale prior knowledge. All experiments were conducted using Google Colab, which provided access to NVIDIA Tesla T4 GPUs with 16 GB of VRAM. The training was performed using the Ultralytics YOLOv11 training pipeline, which includes built-in data augmentation, automatic mixed precision (AMP), and a well-optimized training loop.

The goal of this setup was to compare the effectiveness and efficiency of transfer learning versus training from scratch for the task of Indonesian traffic sign detection. Transfer learning is expected to converge faster due to the knowledge transferred from large-scale datasets like COCO, while training from scratch is hypothesized to require more epochs and computational resources to reach competitive performance.

D. Evaluation Metrics

To evaluate the performance of the YOLOv11 model for Indonesian traffic sign detection, we adopt standard object detection metrics that jointly assess the model's localization and classification accuracy. These metrics are critical for understanding how well the model detects traffic signs of varying sizes and conditions in real-world scenarios.

A central concept in evaluating object detection is the Intersection over Union (IoU), which quantifies the overlap between the predicted bounding box and the ground truth bounding box. It is defined as:

$$\mathrm{IoU} = \frac{\left|\mathrm{B_p} \cap \mathrm{B_{gt}}\right|}{\left|\mathrm{B_p} \cup \mathrm{B_{gt}}\right|} \tag{1}$$

where B_p is the predicted bounding box and B_{gt} is the ground truth bounding box. A prediction is considered a True Positive (TP) if its IoU with a ground truth box exceeds a predefined threshold (commonly 0.5). Conversely, if the IoU is below the threshold or if the prediction does not match any ground truth, it is considered a False Positive (FP). Meanwhile, a False Negative (FN) occurs when a ground truth object is missed (i.e., not matched by any prediction above the IoU threshold). This makes IoU not only a localization metric but also a core criterion for classifying detection results.

Precision measures the ratio of correctly predicted positive samples (true positives) to the total predicted positives. It indicates how many of the detected objects are correct:

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{2}$$

Recall measures the ratio of correctly predicted positives to all actual positives in the ground truth. It reflects the model's ability to detect relevant objects:

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{3}$$

The F1 Score is the harmonic mean of precision and recall. It is a balanced metric that considers both false positives and false negatives. A high F1 score indicates a good balance between precision and recall.

$$\text{F1 Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

Finally, this study uses mean Average Precision (mAP) as the primary evaluation metric. It provides a summary of the model's precision across different recall levels. The mAP is computed by calculating the Average Precision (AP) for each class and then averaging those values. AP is calculated as the area under the precision-recall curve:

$$\text{AP} = \int_0^1 p(r)\ dr \quad (5)$$

where p denotes precision, and r denotes recall. mAP aggregates AP scores across all object classes. In our evaluation, we report both (1) $mAP@0.5$, where IoU is fixed at 0.5 and (2) $mAP@0.5{:}\,0.95$, which averages AP over IoU thresholds from 0.5 to 0.95 (in steps of 0.05):

$$\text{mAP@0.5:0.95} = \frac{1}{10} \sum_{\text{IoU}=0.5}^{0.95} \text{AP@IoU} \quad (6)$$

We follow the standard mean Average Precision (mAP) metric at a 0.5 IoU threshold as defined in the PASCAL VOC challenge [22] to evaluate model performance. To provide a more robust performance evaluation, we also report $mAP@0.5{:}\,0.95$ following the COCO evaluation protocol, [12] which penalizes localization inaccuracies more strictly.

By incorporating these metrics, we gain a comprehensive understanding of the model's ability to correctly localize and classify traffic signs, which is essential for safe and reliable deployment in real-world transportation systems.

III. Result and Discussion

A. Quantitative Results

This section presents a detailed comparison of the five trained models in terms of their detection performance on the validation set. The models include three variants using transfer learning (TL-10, TL-20, TL-30) and two models trained from scratch (FS-30, FS-60). Each model is evaluated using standard object detection metrics: Precision, Recall, mAP@0.5, and mAP@0.5:0.95, as previously defined in section 2.

TABLE 2
Model performance comparison

Model	Precision	Recall	mAP@0.5	mAP@0.5:0.95
TL-10	0.731	0.983	**0.782**	0.696
TL-20	**0.739**	0.988	0.780	0.710
TL-30	0.731	**0.994**	0.781	**0.711**
FS-30	0.728	0.978	0.763	0.691
FS-60	0.721	0.979	0.777	0.703

Table 2 summarizes the quantitative performance of all five models. The results indicate that all models demonstrate satisfactory performance, with Recall consistently exceeding 0.97 across all configurations. Among the transfer learning models, TL-30 yields the highest performance in terms of Recall, mAP@0.5, and mAP@0.5:0.95. TL-20 achieves the highest Precision value among the transfer learning models. In comparison, the models trained from scratch generally exhibit inferior performance across most metrics, particularly with respect to mAP@0.5:0.95. Nonetheless, FS-60 achieves a slightly higher mAP@0.5:0.95 than TL-10 (0.703 vs. 0.696), which may be attributed to the longer training duration.

TABLE 3
Class-wise Precision Comparison

Model	Turn Right	Turn Left	No U-turn	U-turn
TL-10	0.587	0.396	**0.973**	0.970
TL-20	0.592	**0.399**	0.966	**0.998**
TL-30	0.593	0.396	0.969	0.965
FS-30	**0.595**	0.396	0.932	0.990
FS-60	0.584	0.395	0.924	0.980

Table 3 presents the precision scores for each class across models. Overall, the transfer learning models consistently outperformed or matched the from-scratch models in most classes, especially during early training (e.g., TL-10 vs. FS-30). For example, the TL-10 model achieved a high precision of 0.973 and 0.970 for no U-turn and U-turn signs, respectively, which is already competitive with longer-trained from-scratch models. As the training epoch increased, the TL models demonstrated slight improvements, peaking at 0.998 precision for the U-turn class with TL-20.

The turn right and turn left classes showed smaller precision margins across models, with all models yielding similar values around 0.59 and 0.39, respectively. This suggests these two classes might be more challenging, potentially due visual similarity or limited dataset quality. Interestingly, while FS-60 was trained for twice as long as TL-30, it still failed to outperform TL-30 in most categories. This highlights the efficiency and effectiveness of transfer learning, especially when training resources or annotated data are limited.

TABLE 4
Class-wise Recall Comparison

Model	Turn Right	Turn Left	No U-turn	U-turn
TL-10	0.973	0.985	0.987	0.988
TL-20	0.980	0.985	0.987	**1.000**
TL-30	**1.000**	**1.000**	0.987	0.990
FS-30	0.976	0.970	0.987	0.979
FS-60	0.961	0.970	**1.000**	0.983

Table 4 shows the recall scores for each traffic sign class across all models. In general, models trained with transfer learning demonstrated superior recall performance, particularly as training epochs increased. The TL-30 model achieved perfect recall (1.00) on both the turn right and turn left classes, indicating that the model was able to correctly detect nearly all relevant instances of these signs. This trend was also observed in other TL models, which maintained high recall across all classes, even with fewer epochs (e.g., TL-10 and TL-20 scoring 0.985 or above in all categories).

From-scratch models also performed well but showed slightly lower recall, especially in the turn right class. For example, FS-60 only achieved 0.961 recall for turn right, compared to 1.00 in TL-30. Interestingly, FS-60 reached a perfect recall of 1.00 for the no U-turn class, but this did not translate to consistent improvement across the board. These results indicate that transfer learning provides better generalization and object recognition capability with fewer epochs, particularly for classes with more variability or fewer instances. In contrast, from-scratch training required more epochs but still occasionally underperformed, especially in recognizing turn right signs.

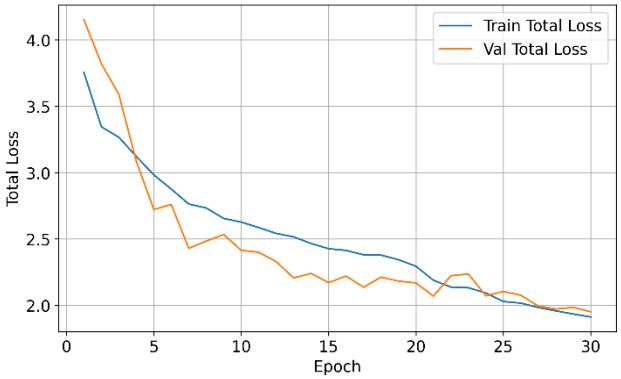

Figure 7. Training and Validation Losses for TL-30 Model

The TL-30 model, trained for 30 epochs using transfer learning, exhibits a consistent decrease in both training and validation losses, as shown in Figure 7. The training loss declines from an initial value of 0.7478 to 0.3155, while the validation loss decreases from 0.9888 to 0.4897. Although the validation loss shows some fluctuations during the early epochs, it stabilizes after Epoch 10. This pattern indicates good generalization and minimal overfitting by the end of training.

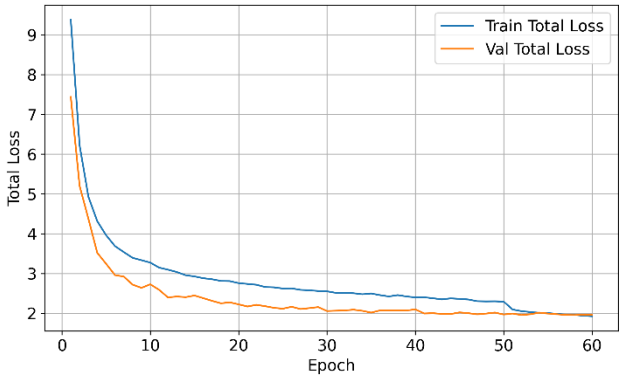

Figure 8. Training and Validation Losses for FS-60 Model

The FS-60 model, trained from scratch for 60 epochs, demonstrates a more gradual decline in losses, as illustrated in Figure 8. The training loss decreases from 2.4501 to 0.3253, and the validation loss from 1.6495 to 0.4906. While the trend reflects effective learning, the broader gap between training and validation losses suggests a tendency toward overfitting, particularly during the initial stages of training.

A direct comparison of the two models reveals that TL-30 converges more rapidly and achieves lower losses earlier in training, benefiting from the pretrained weights introduced through transfer learning. In contrast, FS-60 requires a substantially longer training period to reach comparable performance and exhibits greater susceptibility to overfitting. These results underscore the efficiency and stability offered by transfer learning in accelerating model convergence and improving generalization.

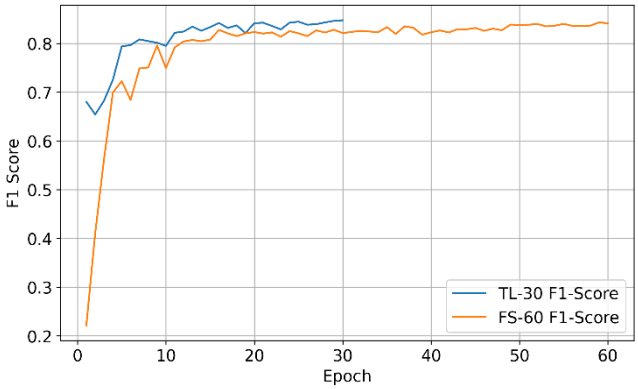

Figure 9. Comparison of Validation F1 Score Between TL-30 and FS-60

Figure 9 shows the comparison of validation F1 score between TL-30 and FS-60. The TL-30 model demonstrates a strong and consistent improvement in F1 score throughout the training process. Starting from an initial value of 0.68 at epoch 1, the F1 score steadily increases, reaching 0.84+ by epoch 20 and peaking at 0.847 by epoch 30. Progression suggests that the model quickly learns and stabilizes within the first 15 epochs, with only minor fluctuations in later stages, indicating good convergence and generalization on the validation set.

The FS-60 model begins with a relatively low F1 score of 0.22 in epoch 1 but shows rapid gains in the early stages, surpassing 0.8 by epoch 12. The model continues to improve gradually, with F1 scores stabilizing between 0.82 and 0.84 from epoch 20 onward. By epoch 60, the F1 score reaches 0.841, suggesting that the model is capable of achieving high performance through extended training, despite a slower convergence rate.

While both models ultimately attain comparable F1 scores of approximately 0.84, the TL-30 model reaches this level of performance significantly earlier, highlighting the benefits of transfer learning in terms of training efficiency. FS-60, in contrast, necessitates the full training duration to match TL-30's performance, reflecting a more extended learning curve. This comparison further reinforces the advantage of leveraging pretrained models for faster convergence and enhanced performance with fewer resources.

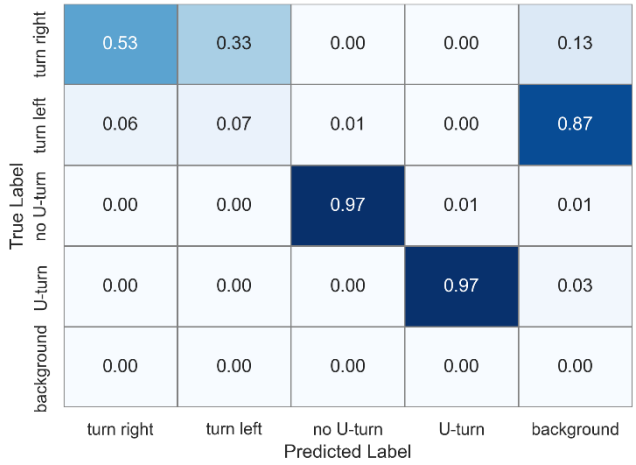

Figure 10. Normalized Confusion Matrix for TL-30 Model

Figure 10 presents the normalized confusion matrix for the TL-30 model. The results indicate excellent classification performance for the "no U-turn" and "U-turn" classes, with over 97% of instances correctly identified. The model encounters challenges in detecting the "turn left" class, which is frequently misclassified as background. Additionally, the "turn right" class shows some confusion with "turn left," indicating potential difficulty in distinguishing between similar directional signs.

Figure 11 shows the normalized confusion matrix for the FS-60 model. Similar to TL-30, it maintains strong performance in identifying the "no U-turn" and "U-turn" signs, albeit with slightly reduced precision. The FS-60 model demonstrates improved recognition of the "turn left" class, with fewer instances misclassified as background. Although some predictions fall into the background class, this behavior is typical in object detection and generally reflects cases with low confidence or partial visibility rather than systematic failure.

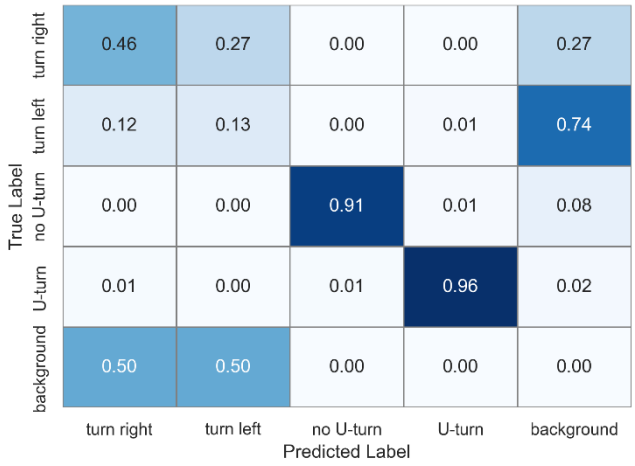

Figure 11. Normalized Confusion Matrix for FS-60 Model

These findings suggest that models incorporating transfer learning, particularly TL-30, are more suitable for practical deployment in real-world traffic sign detection systems. The TL-30 model achieves optimal performance more rapidly, thereby reducing training time and computational cost. This makes it more adaptable to real-time applications where retraining may be necessary. Future work may focus on addressing class-specific challenges, such as improving the classification of visually similar traffic signs, through targeted data augmentation or enhancements in model architecture. Moreover, investigating lightweight models could facilitate deployment on resource-constrained devices commonly used in edge computing scenarios.

B. Qualitative Results

To further illustrate the performance differences between the transfer learning (TL-30) and from-scratch (FS-60) models, we present qualitative analyses using representative images from each class. We examine the inference results on selected test images across four traffic sign classes (turn right, turn left, no U-turn, and U-turn) under diverse and challenging conditions, including low-light/nighttime environments, motion blur, and partial occlusion. For each class, two representative samples were chosen to illustrate how the models handle these edge cases. This analysis provides deeper insight into the robustness and generalization capabilities of each model beyond overall precision and recall scores.

As illustrated in Figure 12, two qualitative samples for the turn right class were evaluated under different challenging conditions. In the first sample (top row), the TL-30 model incorrectly classified the sign as turn left with a confidence score of 0.56, resulting in a false positive. Conversely, the FS-60 model correctly identified the sign as turn right, albeit with a lower confidence of 0.45. This suggests that in certain scenarios, the FS-60 model may demonstrate more cautious yet accurate predictions, whereas the TL-30 model, despite generally higher performance, can exhibit overconfident misclassification in ambiguous contexts.

Figure 12. Qualitative comparison of "turn right" predictions. Left column: TL-30. Right column: FS-60.

In the second sample (bottom row), the input image was affected by motion blur. Both models failed to correctly detect the turn right sign and instead misclassified it as turn left, with confidence scores of 0.50 (TL-30) and 0.49 (FS-60). This result highlights a shared vulnerability in handling blurred or low-quality visual input, indicating both models rely heavily on clear edge definitions and shape consistency for accurate detection. These observations from Figure 12 show that while TL-30 generally outperforms FS-60, the from-scratch model may occasionally offer better reliability in certain edge cases, and both models face challenges when operating under impaired visual conditions.

The turn left class is illustrated in Figure 13, featuring two samples that highlight model performance under both normal and occluded conditions. In the first sample, both the TL-30 and FS-60 models successfully detected the turn left sign in a clear, unobstructed image. TL-30 produced a correct prediction with a confidence score of 0.53, slightly outperforming FS-60, which also yielded a true positive but with a lower confidence of 0.47. This result aligns with the general trend observed in the quantitative evaluation, where TL-30 consistently offers higher detection confidence.

The second sample introduces a challenging scenario where the turn left sign is partially occluded by overgrown vegetation. In this case, the TL-30 model successfully detected the correct class with a confidence of 0.52. In contrast, the FS-60 model misclassified the sign as turn right with a confidence of 0.43, resulting in a false positive. This suggests that the TL-30 model may possess better robustness to partial occlusion, likely due to the advantage of pre-learned feature representations from transfer learning.

Figure 13. Qualitative comparison of "turn left" predictions. Left column: TL-30. Right column: FS-60.

Figure 14 presents two qualitative samples for the no U-turn class, showcasing model performance under challenging visual conditions such as blur and low-light/nighttime environments. In the first sample, the image is affected by motion blur, yet both TL-30 and FS-60 models successfully produced true positive detections. Interestingly, the FS-60 model achieved a slightly higher confidence score of 0.86 compared to TL-30's 0.82.

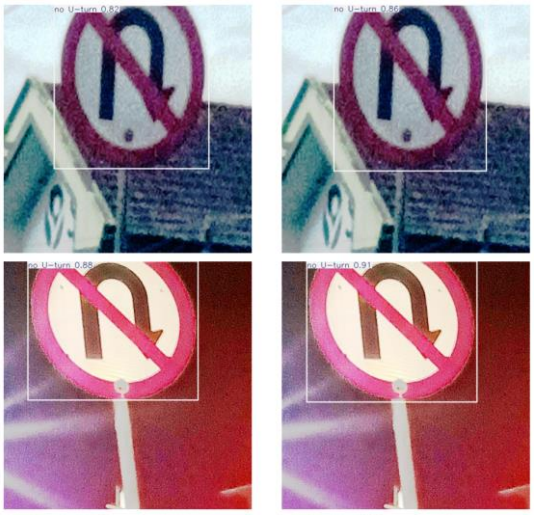

Figure 14. Qualitative comparison of "no U-turn" predictions. Left column: TL-30. Right column: FS-60.

The second sample was taken in a nighttime scenario with reduced lighting. Again, both models accurately detected the no U-turn sign, with FS-60 again showing a slightly higher confidence (0.91) than TL-30 (0.88). These results indicate that both models are highly reliable in detecting the no U-turn class, even under suboptimal conditions. Overall, Figure 14 demonstrates that for the no U-turn class, both TL-30 and FS-60 models perform consistently and confidently, with FS-60 showing a marginal advantage in confidence under blur and

low-light conditions. This could be attributed to the high visual contrast and distinctive shape of the no U-turn sign, making it easier to detect accurately across different training strategies

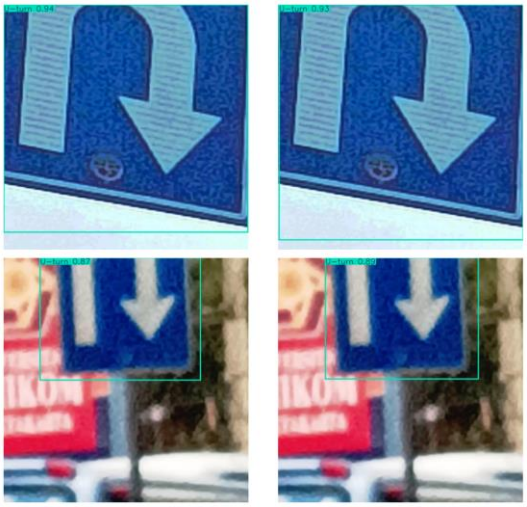

Figure 15. Qualitative comparison of "U-turn" predictions. Left column: TL-30. Right column: FS-60.

Figure 15 illustrates the qualitative results for the U-turn class, evaluated under both normal and blurry visual conditions. In the first sample, which features a clear and well-lit image, both the TL-30 and FS-60 models successfully produced true positive detections. TL-30 predicted the correct class with a confidence of 0.94, while FS-60 closely followed with a confidence of 0.93. This indicates that both models perform consistently well in optimal visual conditions, with only minimal differences in confidence levels.

In the second sample, the input image was affected by motion blur. Despite the visual degradation, both models again correctly detected the U-turn sign. The FS-60 model produced a slightly higher confidence of 0.89, compared to TL-30's 0.87. This reinforces the observation from previous classes that both models maintain strong performance even under blurry conditions, with FS-60 occasionally exhibiting marginally higher confidence in such scenarios.

The qualitative analysis provides further insight into the strengths and limitations of both the TL-30 and FS-60 models beyond what quantitative metrics alone can capture. Across various challenging conditions—including motion blur, low lighting, occlusion, and clear visibility—TL-30 consistently demonstrated higher robustness and generalization, particularly when signs were partially obscured or visually ambiguous. However, the FS-60 model occasionally outperformed TL-30 in specific scenarios, such as detecting certain signs in blurry or low-light conditions, albeit often with lower overall confidence. These findings highlight that while transfer learning significantly enhances detection reliability with fewer training epochs, combining it with data augmentation or robustness-focused training strategies could further improve model performance in real-world deployments.

C. Discussion

This study demonstrates the significant effectiveness of transfer learning in training traffic sign detection models, as evidenced by the higher mean Average Precision (mAP) achieved by the TL-30 model compared to the FS-60 model trained from scratch. Transfer learning leverages pretrained feature representations from large-scale datasets, enabling the model to converge faster and generalize better on the target task with fewer training epochs. This advantage is clearly reflected in the TL-30 model's rapid improvement in mAP, reaching strong performance within just 15 epochs, whereas the FS-60 model required substantially longer training to approach comparable accuracy.

A notable challenge observed in this study was the class-specific variability in detection performance, particularly for the "turn left" sign. Both quantitative confusion matrices and qualitative prediction analyses revealed frequent misclassification between "turn left" and visually similar classes, such as "turn right" and background. This suggests that the issue stems from low visual distinctiveness rather than class imbalance or labeling errors. Subtle directional cues, such as arrow orientation, may be difficult for the model to reliably detect under real-world conditions involving image noise, resolution limits, or occlusion. Addressing this limitation may require targeted data augmentation, higher-resolution inputs, or improved model sensitivity to fine-grained features.

From an application perspective, the superior performance and faster convergence of the transfer learning model have important implications for real-world deployment of traffic sign detection systems. Efficient training reduces development time and computational costs, facilitating timely updates and scalability in dynamic environments. Moreover, understanding the limitations in detecting certain classes guides future dataset refinement and model optimization efforts, ultimately improving the robustness and reliability of intelligent transportation systems.

IV. CONCLUSION

This study investigated the performance of transfer learning versus training from scratch for traffic sign detection using a custom dataset. The transfer learning model demonstrated superior accuracy, faster convergence, and more efficient training compared to the model trained from scratch. While both models eventually achieved comparable detection performance, transfer learning significantly reduced the required training time and mitigated overfitting risks.

Class-specific analysis revealed challenges in accurately detecting visually similar signs such as "turn left," highlighting the need for improved dataset diversity and targeted augmentation strategies. These insights are valuable

for advancing traffic sign detection systems in real-world applications, where timely and reliable detection is critical for intelligent transportation and autonomous driving.

Future work may explore integrating additional data sources and refining model architectures to further enhance detection robustness, particularly for ambiguous classes. Overall, the findings affirm the effectiveness of transfer learning as a practical approach for developing high-performance object detection models with limited training data.

REFERENCES

[1] M.-Y. Fu and Y.-S. Huang, "A survey of traffic sign recognition," in *2010 International Conference on Wavelet Analysis and Pattern Recognition*, Qingdao, China: IEEE, Jul. 2010, pp. 119–124. doi: 10.1109/ICWAPR.2010.5576425.

[2] T. Chaudhari, A. Wale, A. Joshi, and S. Sawant, "Traffic Sign Recognition Using Small-Scale Convolutional Neural Network," *SSRN Journal*, 2020, doi: 10.2139/ssrn.3645805.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA: IEEE, Jun. 2014, pp. 580–587. doi: 10.1109/CVPR.2014.81.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, Eds., Curran Associates, Inc., 2015. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2015/file/14bfa6bb14875e45bba028a21ed38046-Paper.pdf

[5] W. Liu *et al.*, "SSD: Single Shot MultiBox Detector," in *Computer Vision – ECCV 2016*, vol. 9905, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., in Lecture Notes in Computer Science, vol. 9905. , Cham: Springer International Publishing, 2016, pp. 21–37. doi: 10.1007/978-3-319-46448-0_2.

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.

[7] G. Jocher, J. Qiu, and A. Chaurasia, *Ultralytics YOLO*. (Jan. 10, 2023). Ultralytics. [Online]. Available: https://github.com/ultralytics/ultralytics

[8] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, "Detection of traffic signs in real-world images: The German traffic sign detection benchmark," in *The 2013 International Joint Conference on Neural Networks (IJCNN)*, Dallas, TX, USA: IEEE, Aug. 2013, pp. 1–8. doi: 10.1109/IJCNN.2013.6706807.

[9] A. Mogelmose, M. M. Trivedi, and T. B. Moeslund, "Vision-Based Traffic Sign Detection and Analysis for Intelligent Driver Assistance Systems: Perspectives and Survey," *IEEE Trans. Intell. Transport. Syst.*, vol. 13, no. 4, pp. 1484–1497, Dec. 2012, doi: 10.1109/TITS.2012.2209421.

[10] D. Temel, M.-H. Chen, and G. AlRegib, "Traffic Sign Detection Under Challenging Conditions: A Deeper Look into Performance Variations and Spectral Characteristics," *IEEE Trans. Intell. Transport. Syst.*, vol. 21, no. 9, pp. 3663–3673, Sep. 2020, doi: 10.1109/TITS.2019.2931429.

[11] F. Zhuang *et al.*, "A Comprehensive Survey on Transfer Learning," *Proc. IEEE*, vol. 109, no. 1, pp. 43–76, Jan. 2021, doi: 10.1109/JPROC.2020.3004555.

[12] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," 2014, *arXiv*. doi: 10.48550/ARXIV.1405.0312.

[13] J. Deng, W. Dong, R. Socher, L.-J. Li, Kai Li, and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL: IEEE, Jun. 2009, pp. 248–255. doi: 10.1109/CVPR.2009.5206848.

[14] H.-C. Shin *et al.*, "Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning," *IEEE Trans. Med. Imaging*, vol. 35, no. 5, pp. 1285–1298, May 2016, doi: 10.1109/TMI.2016.2528162.

[15] C. Rahmad, I. F. Rahmah, R. A. Asmara, and S. Adhisuwignjo, "Indonesian traffic sign detection and recognition using color and texture feature extraction and SVM classifier," in *2018 International Conference on Information and Communications Technology (ICOIACT)*, Yogyakarta: IEEE, Mar. 2018, pp. 50–55. doi: 10.1109/ICOIACT.2018.8350804.

[16] A. I. Pradana, S. Rustad, G. F. Shidik, and H. Agus Santoso, "Indonesian Traffic Signs Recognition Using Convolutional Neural Network," in *2022 International Seminar on Application for Technology of Information and Communication (iSemantic)*, Semarang, Indonesia: IEEE, Sep. 2022, pp. 426–430. doi: 10.1109/iSemantic55962.2022.9920448.

[17] P. Chyan, N. T. S. Saptadi, and J. M. Leda, "Small Object Detection Approach Based On Enhanced Single-Shot Detector For Detection And Recognition Of Indonesian Traffic Signs," *BAREKENG: J. Math. & App.*, vol. 18, no. 4, pp. 2653–2662, Oct. 2024, doi: 10.30598/barekengvol18iss4pp2653-2662.

[18] A. Mulyanto, R. I. Borman, P. Prasetyawan, W. Jatmiko, P. Mursanto, and A. Sinaga, "Indonesian Traffic Sign Recognition For Advanced Driver Assistent (ADAS) Using YOLOv4," in *2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, Yogyakarta, Indonesia: IEEE, Dec. 2020, pp. 520–524. doi: 10.1109/ISRITI51436.2020.9315368.

[19] E. Yohannes *et al.*, "Indonesia Traffic Sign Recognition Using a One-Stage Detector YOLOv8," in *2024 Seventh International Conference on Vocational Education and Electrical Engineering (ICVEE)*, Malang, Indonesia: IEEE, Oct. 2024, pp. 169–174. doi: 10.1109/ICVEE63912.2024.10824005.

[20] Indonesia Traffic Sign, "Traffic Sign Dataset," *Roboflow Universe*. Roboflow, Apr. 2024. [Online]. Available: https://universe.roboflow.com/indonesia-traffic-sign/traffic-sign-kdnr1

[21] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010, doi: 10.1109/TKDE.2009.191.

[22] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *Int J Comput Vis*, vol. 88, no. 2, pp. 303–338, Jun. 2010, doi: 10.1007/s11263-009-0275-4.