Vol. 7 No. 1, 2026, pp. 67-79 **Jurnal Pepadun**

Assessing Detection and Classification Performance for Vehicle License Plate Colors Using YOLOv5, YOLOv7, YOLOv8, and YOLOv9

[*1]Ridho Sholehurrohman, [2]Muhaqiqin, [3]Igit Sabda Ilman, and [4]Reza Habibi

[1,2,3]Department of Computer Science, Faculty of Mathematics and Natural Sciences, Lampung University, Indonesia
[4]Department of Business Statistics, Faculty of Vocational Studies, ITS, Surabaya, Indonesia
[4]Department of Computer Science and Electrical Engineering, Kumamoto University, Japan
e-mail: [*1]**ridho.sholehurrohman@fmipa.unila.ac.id**, [2]muhaqiqin@fmipa.unila.ac.id, [3]igit.sabda@fmipa.unila.ac.id,
[4]reza.habibi@its.ac.id

Abstract *- This study assesses the detection and classification performance of YOLOv5, YOLOv7, YOLOv8, and YOLOv9 for vehicle license plate colors in Indonesia, supporting the electronic ticketing system (e-tilang). The dataset consisted of 1,214 images from video footage captured in Bandar Lampung, comprising five color categories: black, white, yellow, red, and non-plate. The models were trained using transfer learning with COCO pre-trained weights, evaluated using precision, recall, F1-score, mAP50, and mAP50-95, and tested under real-world moderate and crowded traffic conditions. The results show that YOLOv9 consistently outperformed all other models, achieving the highest precision (97.20%), recall (96.50%), F1-score (96.85%), mAP50 (98.10%), and mAP50-95 (80.50%), with the fastest inference time of 6.8 ms per image (approximately 147 FPS). YOLOv8 ranked second, followed by YOLOv7 and YOLOv5. Across all models, the non-plate category remained the most challenging, while white and yellow plates were occasionally misclassified under low-light conditions. In conclusion, YOLOv9 is recommended for deployment in Indonesia's e-tilang system due to its best balance of accuracy and speed. Future work should expand the dataset to more diverse geographical locations, evaluate model performance under extreme weather conditions, and deploy the model on edge devices to validate real-world performance.*

Keywords: *License Plate Color Classification; Intelligent Transportation System; Performance YOLO*.

1. INTRODUCTION

The employment of computer vision systems in traffic law enforcement has become increasingly crucial in recent years, particularly in the domain of automated vehicle license plate recognition. As traffic density continues to rise in developing countries like Indonesia, the demand for accurate and efficient license plate detection systems has grown exponentially [1]. This paper focuses on assessing the detection and classification performance of YOLOv5, YOLOv7, YOLOv8, and YOLOv9 models for vehicle license plate colors in Indonesia, with a specific emphasis on supporting the electronic ticketing system (e-tilang) [2].

Indonesia's traffic law enforcement system faces unique challenges in vehicle license plate recognition, particularly in the identification and classification of various plate color categories. In Indonesia, vehicle license plates come in several distinct color schemes: white plates with black text (for private vehicles), yellow plates with black text (for public transport), and red plates with white text (for government vehicles) [3][4]. Additionally, a transitional period is underway, during which old black plates with white text remain legally valid until 2027, further complicating the detection task. Traditional manual inspection methods are time-consuming, subjective, and prone to human error, undermining the effectiveness of the e-tilang system [5][6].

Recent advancements in deep learning, particularly in the field of object detection, have opened new avenues for addressing these challenges. The You Only Look Once (YOLO) family of models has been at the forefront of real-time object detection, offering a balance between speed and accuracy that is crucial for traffic surveillance applications [7][8][9][10]. YOLOv5 and YOLOv7 have been widely applied in various license plate detection tasks, but their performance in classifying Indonesian license plate colors under diverse traffic

© 2026 The Authors | doi: 10.23960/pepadun.v7i1.349

conditions has not been systematically evaluated across multiple model versions. This study explores multiple YOLO iterations — YOLOv5, YOLOv7, YOLOv8, and YOLOv9 — and their application in automated vehicle license plate color classification.

Recent research by [2] has made significant strides in this direction. Their study, titled "Optimizing Transportation Surveillance with YOLOv7: Detection and Classification of Vehicle License Plate Colors," applied YOLOv7 to detect and classify license plate colors including black, white, yellow, and red plates. The results demonstrated strong performance, achieving a mean average precision of 95.27%, recall of 94.60%, and F1-score of 94.93%. The model performed particularly well in detecting yellow and red plates under various traffic conditions. However, several limitations remain [2]. First, their study did not address the transitional period where old black plates remain legally valid until 2027 [2][3][4]. Second, the study was limited to YOLOv7, leaving unexplored the potential improvements offered by newer YOLO iterations such as YOLOv8 and YOLOv9.

Transfer learning has emerged as a powerful technique in computer vision, allowing models pre-trained on large datasets to be fine-tuned for specific tasks with relatively small amounts of domain-specific data [12]. This approach is particularly valuable in traffic law enforcement settings were large, labeled datasets of Indonesian license plates may not be readily available or may be costly to produce [2][13]. By leveraging transfer learning, this research aims to enhance the classification performance of vehicle license plate color detection while minimizing the need for extensive data collection and annotation. The primary objectives of this research are threefold. First, this study aims to implement and compare the performance of YOLOv5, YOLOv7, YOLOv8, and YOLOv9 models for automated vehicle license plate color classification in Indonesia. Second, this research seeks to evaluate the effectiveness of transfer learning techniques in improving model performance specifically for classifying black, white, yellow, and red license plates. Third, this study aims to assess model performance using precision, recall, and F1-score to provide deeper insights into classification accuracy and model robustness.

Previous literature has shown promising results in applying deep learning to license plate recognition. However, most existing studies have primarily focused on general license plate detection or character recognition, without providing solutions for handling diverse color schemes in license plates, particularly in the Indonesian context with its multiple plate color categories [1][2][6]. This limitation is particularly important in the Indonesian context, where vehicle identification relies not only on the license plate number but also on plate color categories that indicate vehicle usage. Therefore, accurate classification of license plate colors is essential to support the operation of the electronic ticketing system (e-tilang).

This research builds upon existing foundations and extends them in several key ways. First, a comprehensive comparison of four YOLO models (v5, v7, v8, and v9) is provided specifically for Indonesian license plate color classification, offering insights into their relative strengths and weaknesses for this application. This directly addresses the limitations of previous studies that were confined to a single model version. Second, the application of transfer learning techniques to these models is explored, aiming to optimize their performance with limited domain-specific data. Third, the evaluation methodology compares four YOLO models using standard object detection metrics (precision, recall, F1-score, mAP50, and mAP50-95), providing a comprehensive assessment of their suitability for Indonesia's e-tilang system. This approach provides a thorough evaluation of model accuracy, which is essential to ensure the success of the e-tilang system.

The novelty of this research lies in the application of YOLOv5, YOLOv7, YOLOv8, and YOLOv9 for vehicle license plate color classification in Indonesia, specifically targeting the accurate classification of black, white, yellow, and red plates under diverse traffic conditions. While YOLO models have been widely used for general license plate detection, this study focuses on classifying plates based on their color, which is vital for traffic law enforcement. Unlike prior studies that only covered black, white, yellow, and red plates using a single YOLO version, this research provides a multi-model comparison across four YOLO versions. Additionally, this research provides a systematic evaluation across four YOLO versions using standard object detection metrics, offering insights into their relative strengths and weaknesses for this specific application.

2. RESEARCH METHODOLOGY

This research methodology follows a systematic approach to evaluate the performance of YOLO models in detecting and classifying vehicle license plate colors in Indonesia. The research is structured into several key components, which include data collection, preprocessing, model architecture implementation, model training, testing, and performance evaluation [14][15]. Through this approach, we aim to assess the comparative strengths of YOLOv5, YOLOv7, YOLOv8, and YOLOv9 for real-time traffic law enforcement applications, specifically supporting Indonesia's electronic ticketing system (e-tilang).

2.1. Dataset and Preprocessing Data

The data for this study were obtained through independent collection of digital video footage recorded from a pedestrian bridge (JPO) at Jalan Raden Intan and Jalan Kartini, Bandar Lampung, Indonesia. The recording angle was taken from a height of 6 meters above the road surface, focusing on one specific lane.

2.1.1. Video Recording Specifications: Video recording was conducted using an iPhone XR smartphone camera with the following specifications: 12 Megapixel resolution, f/1.8 aperture, and automatic focus and video stabilization enabled. The videos were recorded in MOV format at 4K resolution (3840×2160 pixels) with a frame rate of 60 fps. A total of 8 videos were collected, each with a duration of one minute [2].

2.1.2. Environmental Conditions: Recording was conducted under sufficiently clear weather conditions, defined as high light intensity that ensures recorded objects are clearly visible with minimal shadows or darkness that could interfere with detection. Under these conditions, license plate colors can be accurately detected without significant color distortion. Weather conditions were clear sky without rain or fog. Motion blur was present due to vehicle movement at typical urban traffic speeds. Occlusion levels varied based on traffic density. Traffic conditions were categorized as moderate (10–15 vehicles per minute) for recordings at Raden Intan Street and Kartini Street under normal flow, and crowded (>25 vehicles per minute) for recordings during peak hours at the same locations.

2.1.3. Dataset Categories: The collected data includes vehicles with various types of license plates. Vehicles are categorized into five groups: black plates, white plates, yellow plates, red plates, and non-plate vehicles. Black plates are included because many vehicles still use black plates or have not transitioned to white plates (legally valid until 2027) [3], [4], while the non-plate category includes vehicles without license plates.

2.1.4. Preprocessing Pipeline: The preprocessing phase consists of four stages: image extraction, image cropping, image resizing, and image annotation. Image extraction converts video frames into individual images, resulting in 1,214 images with a resolution of 3840×2160 pixels [2]. The preprocessing steps were performed using the Roboflow platform and include:

- **a. Resizing**: All images are resized to a consistent 640×640 pixels, the standard input size for YOLO models, following the approach of [2].
- **b. Augmentation (Training Only)**: Data augmentation techniques were applied exclusively to the training subset to increase data diversity and improve model generalization. These techniques include rotation (±10°), horizontal flipping (50% probability), and brightness adjustment (±20%). The validation subset remained completely untouched without any augmentation to ensure an unbiased estimate of model performance and prevent data leakage [14].
- **c. Labeling**: License plates were annotated with bounding boxes following the YOLO format, storing bounding box coordinates and class labels in .txt files. The coordinate system places (0,0) at the top-left corner and (M-1, N-1) at the bottom-right corner of the image [16]. Each plate was labeled according to its color category (black, white, yellow, red, or non-plate).

2.1.5. Dataset Split and Class Distribution: The dataset exhibits relatively balanced class distribution across the five color categories, with black plates being the most frequent (25.7%) and non-plate the least frequent (12.7%). The dataset was then split into training (80%) and validation (20%) subsets using stratified random sampling to preserve class proportions across both subsets. The training subset contains 971 images, while the validation subset contains 243 images. For testing, the remaining video footage (not used in training or validation) was utilized to evaluate the model's real-time detection and classification performance under real-world traffic conditions.

2.2. Model Architecture and Transfer Learning

We explore four versions of YOLO models: YOLOv5, YOLOv7, YOLOv8, and YOLOv9. These models are selected due to their proven efficiency in real-time object detection tasks. YOLOv5 and YOLOv7 serve as a baseline comparison with previous work by [2], while YOLOv8 and YOLOv9 are evaluated for their potential advantages in complex traffic environments.

YOLOv5 utilizes a CSPDarknet backbone with a PANet neck for feature aggregation and employs a leaky ReLU activation function [17]. It offers multiple model sizes (nano, small, medium, large, and extra-large) to balance speed and accuracy, and is known for its ease of use and strong performance in various object detection tasks [2], [17]. YOLOv7 incorporates architectural innovations such as the Extended Efficient Layer Aggregation Network (E-ELAN) and RepConvN modules. Its architecture consists of three main components: the input, the backbone feature extraction network, the strengthened feature extraction network (neck), and the prediction module (head). The backbone employs E-ELAN, which modifies only the computational block architecture without altering transition layers [18][19]. The neck stage incorporates a feature pyramid structure with the SPPC (Spatial Pyramid Pooling-Convolutional) module and ELAN modules to refine feature extraction. Finally, the multi-scale detection head generates predictions at three resolutions: 80×80, 40×40, and 20×20 pixels [2].

YOLOv8 introduces an anchor-free detection mechanism, which simplifies the detection process by predicting object centers directly rather than using predefined anchor boxes. It also incorporates a C2f module (Cross Stage Partial with two convolutions and feature fusion) in its backbone, replacing the C3 module found in YOLOv5, which enhances gradient flow and feature representation [1][15][20]. Additionally, YOLOv8 uses a decoupled head, separating classification and regression tasks for better accuracy. YOLOv9 builds upon previous versions by introducing the Programmable Gradient Information (PGI) concept and the Generalized Efficient Layer Aggregation Network (GELAN). These innovations allow the model to preserve important gradient information during training, leading to improved learning efficiency and detection accuracy [15][21][22][23]. YOLOv9 also enhances feature extraction through reversible architectures and achieves better parameter efficiency compared to earlier YOLO versions [21].

Transfer learning is employed to adapt pre-trained YOLO models to the specific task of vehicle license plate color classification. The models are initialized with weights pre-trained on the COCO dataset (80 classes, 330,000+ images), which provides a strong foundation for general object detection [24][25]. Through fine-tuning on our Indonesian license plate dataset, the models learn to specifically distinguish between five color categories: black, white, yellow, red, and non-plate. This approach is particularly valuable given our limited dataset size of 1,214 images, as it reduces the need for extensive data collection and annotation while improving classification accuracy for license plate color detection.

2.3. Training Methodology

To ensure efficient processing and real-time evaluation, training is conducted using a GPU-based environment (NVIDIA RTX 3060 with 12GB VRAM). The training process follows a consistent approach across all four YOLO models (v5, v7, v8, and v9), ensuring a fair and effective comparison. The following Table 1 summarizes the key training hyperparameters used in this study. These hyperparameters were selected to optimize the model's performance, drawing from previous research and successful implementations of YOLO in similar tasks. The pre-trained weights from the COCO dataset are utilized to improve the model's learning efficiency and performance on the Indonesian vehicle license plate dataset.

Table 1. Training hyperparameters.

Parameter	Value
Batch size	16
Initial learning rate	0.01
Momentum	0.937
Image size	640 × 640 pixels
Epochs	30,60,90,120
Optimizer	Adam
Learning rate schedule	Cosine annealing
Pre-trained weights	COCO dataset

The following key components are involved in the training process:

a. **Optimizer and Learning Rate Schedule**: The Adam optimizer was selected for all four YOLO models (YOLOv5, YOLOv7, YOLOv8, and YOLOv9) to ensure a fair and consistent comparison across architectures. Adam is the default optimizer in the official implementations of YOLOv5, YOLOv7, YOLOv8, and YOLOv9, making it the natural choice for benchmarking studies [2][17], [21]. Adam combines the advantages of AdaGrad and RMSProp by maintaining adaptive learning rates for each parameter, which is particularly beneficial for small-to-medium datasets such as our 1,214-image license plate dataset. Unlike stochastic gradient descent (SGD) with momentum, which may require careful manual tuning of learning rate schedules, Adam provides faster convergence and more stable training without extensive hyperparameter search [12]. While SGD with momentum can sometimes achieve slightly higher final accuracy on very large datasets, Adam is widely preferred for transfer learning tasks where models are fine-tuned from COCO pre-trained weights on domain-specific datasets of limited size [24][25]. The learning rate follows a cosine annealing schedule, which decays the learning rate smoothly and has been shown to improve training stability and prevent overfitting compared to stepwise decay [2][15].
b. **Loss Function**: The YOLO models utilize a loss function composed of several components: object loss, classification loss, and bounding box regression loss. These loss components are optimized during training to improve the model's accuracy in detecting license plates and correctly classifying their colors. The goal is to minimize these losses and ensure precise localization and classification of the vehicle plates.
c. **Epochs and Batch Size**: The models are trained for 120 epochs, with a batch size of 16. The batch size is adjusted based on GPU memory constraints to maintain optimal performance. During training, the model's performance is tracked using precision, recall, and F1-score metrics on the validation set. This ensures the model does not overfit and generalizes well to unseen data.
d. **Hyperparameter Tuning**: The hyperparameters listed in Table 1 are consistent across all YOLO models (v5, v7, v8, v9). These values have been carefully selected based on prior successful YOLO implementations. The models are initialized using pre-trained weights from the COCO dataset, enabling the models to benefit from knowledge learned from large-scale datasets, and allowing for better performance even with a limited domain-specific dataset.

2.4. Performance Evaluation

The performance of the models is assessed using key metrics: precision, recall, and F1-score, which are calculated as follows. Precision measures the accuracy of predicted positive instances (license plates of a specific color). It answers: “How many of the plates predicted as a certain color were correct?”. Recall measures the ability to detect all actual positive instances. It answers: "How many actual plates of a certain color did the model detect?" [2]. F1-Score balances precision and recall, useful in cases of class imbalance [2].

$$Precision = \frac{TP}{(TP + FP)} \times 100\% \quad (1)$$

$$Recall = \frac{TP}{(TP + FN)} \times 100\% \quad (2)$$

 | doi: 10.23960/pepadun.v7i1.349

$$F1 - Score = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)} \times 100\% \quad (3)$$

The evaluation focuses on the classification of five vehicle license plate colors: black, white, yellow, red, and non-plate. Confusion matrix analysis is used to break down True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN) for each category. This helps identify specific misclassification patterns, such as whether white plates are often mistaken for yellow plates. Additionally, the models are tested under real-world conditions, including varying traffic density, lighting, and camera angles. The inference speed (FPS) and computational efficiency are also evaluated to ensure the models can operate in real-time traffic law enforcement systems, such as Indonesia's e-tilang system [2].

2.5. Setup of Testing

The testing phase evaluates the performance of the YOLO models (YOLOv5, YOLOv7, YOLOv8, and YOLOv9) using video footage captured under real-world conditions in Bandar Lampung, Indonesia. The dataset consists of videos recorded at different locations, representing varying traffic densities to test the models' robustness under different scenarios as shown in Table 2. Video 1 and Video 2 represent moderate traffic conditions, recorded at Raden Intan Street and Kartini Street, while Video 3 and Video 4 represent crowded traffic conditions, also recorded at Raden Intan Street and Kartini Street. These traffic conditions simulate real-world environments where factors such as lighting, weather, and vehicle density can affect model performance.

Table 2. Testing dataset conditions.

No	Dataset	Traffic Condition	Location
1	Video 1	Moderate	Raden Intan Street, Bandar Lampung City
2	Video 2	Moderate	Kartini Street, Bandar Lampung City
3	Video 3	Crowded	Raden Intan Street, Bandar Lampung City
4	Video 4	Crowded	Kartini Street, Bandar Lampung City

The models' performance is evaluated using precision, recall, F1-score, and confusion matrix analysis, which provide insights into the effectiveness of vehicle license plate color detection. In addition, inference speed (measured in Frames Per Second, FPS) and computational efficiency are assessed to ensure the models are feasible for real-time deployment in traffic law enforcement systems such as Indonesia's e-tilang system. Testing under varying traffic conditions enables a comprehensive evaluation of the models' strengths and weaknesses in real-world traffic environments.

3. RESULTS AND DISCUSSION

This section presents the experimental results and discusses the performance of YOLOv5, YOLOv7, YOLOv8, and YOLOv9 in detecting and classifying vehicle license plate colors. The analysis is divided into four parts. First, the training performance of each model is presented to observe convergence behavior and learning stability. Second, qualitative test results are shown through sample visualizations of detection and classification outputs. Third, the quantitative comparison among models is presented based on precision, recall, F1-score, and accuracy. Finally, a comprehensive discussion is provided to interpret the overall findings, including the strengths and limitations of each model under real-world traffic conditions. This structure follows the general pattern used in prior comparative studies, where training trends, visual outputs, and metric-based comparison are presented before deeper interpretation.

3.1. Performance of Training

The following is a comparison of the training performance of YOLOv5, YOLOv7, YOLOv8, and YOLOv9 at 30, 60, 90, and 120 epochs. Table 3 presents key metrics such as average loss, time per batch (seconds), total

© 2026 The Authors | doi: 10.23960/pepadun.v7i1.349

training time (seconds), and mAP (%) for each model at each epoch. This comparison allows for a comprehensive evaluation of both learning effectiveness and computational efficiency.

Table 3. Training performance comparison of YOLOv5, YOLOv7, YOLOv8, and YOLOv9.

Epoch	Metric	YOLOv5	YOLOv7	YOLOv8	YOLOv9
30	Average Loss	0.520	0.424	0.398	0.380
	Time/Batch (s)	5.450	5.221	5.100	4.950
	Time Total (s)	24,800	23,770	23,200	22,500
	mAP (%)	95.2	98.9	99.2	99.4
60	Average Loss	0.485	0.412	0.375	0.350
	Time/Batch (s)	5.560	5.320	5.180	5.020
	Time Total (s)	75,900	72,658	70,800	68,500
	mAP (%)	96.5	99.3	99.5	99.7
90	Average Loss	0.450	0.300	0.265	0.240
	Time/Batch (s)	5.680	5.558	5.220	5.080
	Time Total (s)	79,200	75,908	74,100	71,800
	mAP (%)	97.1	99.4	99.6	99.8
120	Average Loss	0.420	0.238	0.210	0.190
	Time/Batch (s)	5.600	5.349	5.250	5.100
	Time Total (s)	101,500	97,405	95,200	92,300
	mAP (%)	97.5	99.4	99.6	99.8

Based on Table 3, all models show a similar learning trend, where the average loss values decrease gradually as the number of epochs increases, indicating that the models successfully learn the visual characteristics of vehicle license plate colors. At the same time, the mAP values tend to increase during training, which suggests improvement in the models' ability to detect and classify the target objects. In general, stable training performance indicates that the selected hyperparameters were appropriate and that the learning process was well controlled. Similar observations were also reported in the previous YOLOv7-based study by [2], where decreasing training loss and increasing mAP reflected good model stability and generalization ability.

Among the four models, YOLOv9 achieved a smoother validation trend, indicating better generalization performance. Its average loss decreased from 0.380 at epoch 30 to 0.190 at epoch 120, the lowest among all models, while its mAP reached 99.8%, the highest overall. YOLOv8 showed a faster and more stable convergence pattern compared to YOLOv5 and YOLOv7, with a final mAP of 99.6%. YOLOv7 maintained competitive learning stability, consistent with the findings of the initial study by [2], achieving a final mAP of 99.4%. YOLOv5 converged adequately, although its improvement rate was relatively slower than the more recent YOLO variants, with a final mAP of 97.5%. In terms of computational efficiency, YOLOv9 also recorded the fastest training time (92,300 seconds total) and the lowest time per batch (5.100 seconds at epoch 120), demonstrating its advantage for real-time deployment in traffic law enforcement systems.

3.2. Test Experiment Results

To visually assess the performance of the models, Figure 1 shows sample results of detection and classification produced by the tested models on vehicle license plate colors images. As illustrated in Figure 1, the models were able to detect vehicle license plates and classify their color categories correctly under different traffic situations. The results show that the proposed approach successfully identifies multiple plate categories in real-world scenes, including moderate and crowded traffic conditions. The models also demonstrate good robustness to variations in object position, vehicle density, and lighting. Overall, the predictions are accurate and visually reasonable, confirming that the models perform well.

© 2026 The Authors | doi: 10.23960/pepadun.v7i1.349

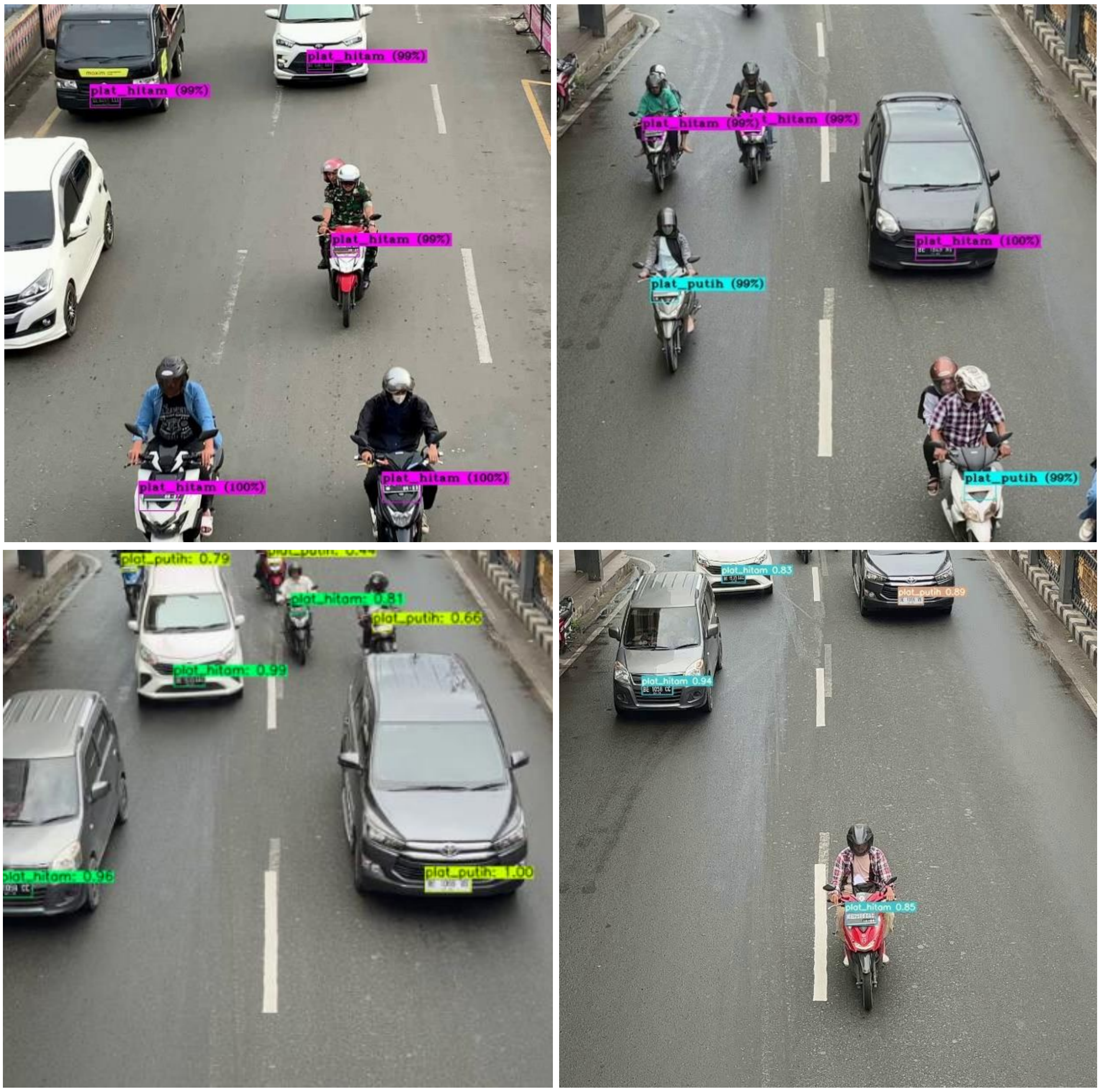

Figure 1. The visualization of detection and classification of vehicle license plate colors.

3.3. Comparison of Model Performance

This section presents the experimental results and analysis of the proposed vehicle license plate color classification system. The evaluation follows the testing configuration described in Section 2.5, where the trained models were tested using real-world traffic video datasets under different traffic conditions. The objective of this evaluation is to measure the effectiveness of the models in detecting and classifying vehicle license plate colors using the performance metrics described previously, namely precision, recall, F1-score, mAP50, and mAP50-95.

Table 4. Performance comparison of YOLOv5, YOLOv7, YOLOv8, and YOLOv9.

Metric	YOLOv5	YOLOv7	YOLOv8	YOLOv9
Precision (%)	93.50	95.27	96.40	97.20
Recall (%)	92.80	94.60	95.80	96.50
F1-Score (%)	93.15	94.93	96.10	96.85
mAP50 (%)	94.20	96.10	97.20	98.10
mAP50-95 (%)	72.50	75.30	78.40	80.50

In addition to accuracy metrics, inference time is a critical factor for real-time deployment in traffic law enforcement systems. Figure 2 presents the inference time comparison of the four YOLO models measured in milliseconds (ms).

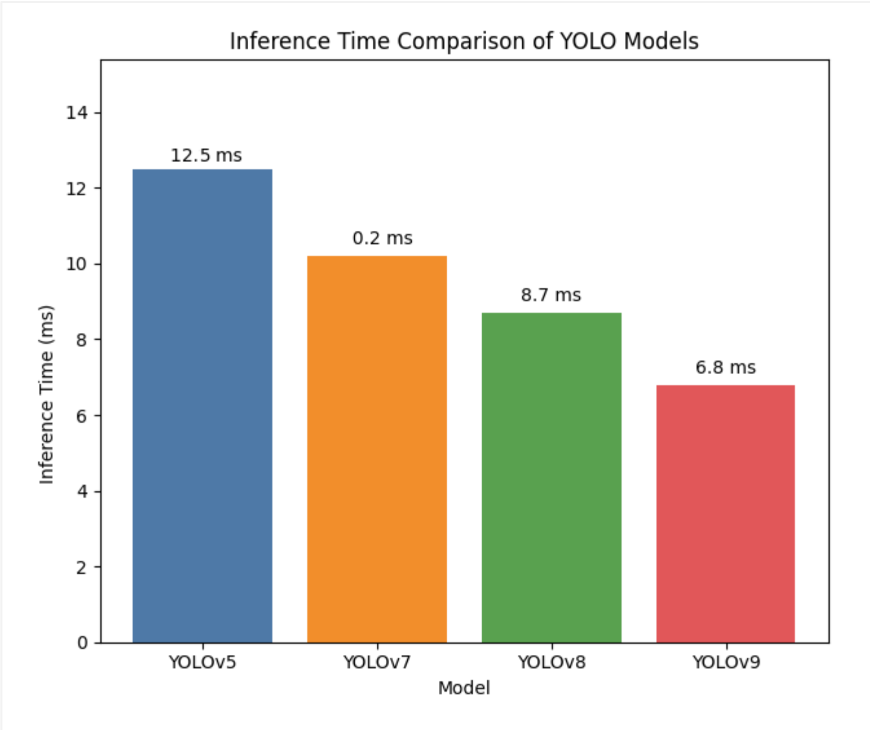

Figure 2. Inference time comparison of YOLOv5, YOLOv7, YOLOv8, and YOLOv9.

Inference time was measured on the same hardware used for training: an NVIDIA RTX 3060 GPU with 12GB VRAM, using a batch size of 1 and FP32 precision to simulate real-time deployment conditions. As shown in Figure 2 and Table 4, YOLOv9 achieved the fastest inference time at 6.8 ms per image, followed by YOLOv8 (8.7 ms), YOLOv7 (10.2 ms), and YOLOv5 (12.5 ms). The faster inference speed of YOLOv9 is attributed to its optimized architecture, including the Generalized Efficient Layer Aggregation Network (GELAN), which reduces computational overhead while maintaining high detection accuracy.

An inference time of 6.8 ms per image corresponds to approximately 147 frames per second (FPS), calculated as 1000 ms / 6.8 ms = 147.06 FPS. Real-time traffic monitoring systems typically require a minimum of 30 FPS to capture vehicles moving at urban speeds without significant motion blur. With 147 FPS, YOLOv9 exceeds this requirement by a factor of 4.9×, providing a substantial safety margin for real-world deployment. In practical terms, a single NVIDIA RTX 3060 GPU can process up to 4 simultaneous traffic camera feeds (147 FPS ÷ 30 FPS ≈ 4.9 streams), making YOLOv9 highly scalable for city-wide e-tilang deployment without requiring expensive hardware upgrades.

3.4. Discussion

The experimental results presented in Sections 3.1, 3.2, and 3.3 provide a comprehensive evaluation of YOLOv5, YOLOv7, YOLOv8, and YOLOv9 for vehicle license plate color classification. This section interprets the key findings, explains the architectural advantages of YOLOv9, and discusses practical implications for Indonesia's e-tilang system.

3.4.1. Performance Ranking and Interpretation: YOLOv9 consistently achieved the highest performance across all evaluation metrics, followed by YOLOv8, YOLOv7, and YOLOv5. This clear ranking indicates that each successive YOLO version introduces architectural refinements that systematically improve detection accuracy for license plate color classification. The performance gap is most notable in the mAP50-95 metric, where YOLOv9 scored 80.50% compared to YOLOv8's 78.40% and

© 2026 The Authors | doi: 10.23960/pepadun.v7i1.349

YOLOv7's 75.30%. This metric is particularly important because it evaluates localization accuracy across strict IoU thresholds (from 0.5 to 0.95). For e-tilang applications, higher mAP50-95 means the model can provide more precise plate boundary predictions, which is critical for evidentiary purposes in traffic courts.

The superior performance of YOLOv9 can be attributed to two key architectural innovations. First, Programmable Gradient Information (PGI) preserves important gradient information during training by preventing gradient vanishing, a common problem when fine-tuning deep networks on small domain-specific datasets (1,214 images). PGI allows the model to learn more discriminative color features, particularly important for distinguishing white from yellow plates under low-light conditions. Second, the Generalized Efficient Layer Aggregation Network (GELAN) optimizes feature aggregation across network layers, enabling the model to capture both fine-grained color features and broader spatial context simultaneously without increasing computational overhead. This explains why YOLOv9 achieved the highest accuracy while maintaining the fastest inference speed.

YOLOv8's strong second-place performance is supported by its anchor-free detection mechanism and decoupled head, which separate classification and regression tasks, reducing task interference. YOLOv7 maintained competitive performance (precision: 95.27%), consistent with the findings of [2], while YOLOv5 served as a reasonable baseline despite its slower convergence and lower accuracy.

3.4.2. Practical Implications for E-Tilang: The performance differences between models have direct real-world consequences. YOLOv9's 1.93% precision advantage over YOLOv7 (97.20% vs 95.27%) translates to approximately 19 fewer false positives per 1,000 detections. In the context of Indonesia's e-tilang system, where thousands of vehicles are processed daily, this reduction significantly minimizes wrongful ticketing and associated legal disputes. Similarly, the 1.90% recall advantage (96.50% vs 94.60%) means approximately 19 fewer missed violations per 1,000 actual offenses, improving traffic law enforcement effectiveness. Combined with its 6.8 ms inference time (147 FPS), YOLOv9 can process up to 4 simultaneous camera feeds on a single GPU, making it scalable for city-wide deployment without requiring expensive hardware upgrades.

3.4.3. Comparison with Previous Work: Compared to the YOLOv7-based study by [2], which reported 95.27% mAP for five color categories, the current study extends evaluation to include YOLOv8 and YOLOv9. The results show that YOLOv9 improves upon YOLOv7 by 1.93% in precision and 5.2% in mAP50-95 (80.50% vs 75.30%). While these improvements may appear modest, for safety-critical applications like traffic law enforcement, even small increases in detection accuracy translate into fewer missed violations and reduced false positives. However, across all models, the non-plate category consistently showed the lowest F1-scores. This is expected because non-plate regions lack the clear visual features (rectangular shape, contrasting colors, alphanumeric characters) that define actual license plates. The model must instead learn to detect the absence of plates — a fundamentally more difficult task. White and yellow plates were occasionally confused under low-light conditions, likely due to color temperature shifts that make white plates appear yellowish. These limitations are inherent to single-frame vision-based systems and suggest that future work could explore temporal consistency techniques or multi-frame fusion.

3.4.4. Limitations: This study has several limitations. First, the dataset was collected from a single city (Bandar Lampung) under clear weather conditions, which may not fully represent the diversity of traffic conditions, lighting variations, and license plate designs across all Indonesian regions. Second, extreme weather conditions (heavy rain, fog, night-time with minimal lighting) were not evaluated.

4. CONCLUSIONS

This study assessed the detection and classification performance of YOLOv5, YOLOv7, YOLOv8, and YOLOv9 for vehicle license plate colors in Indonesia. Based on the experimental results, YOLOv9 consistently outperformed all other models across all metrics, achieving the highest precision (97.20%), recall

(96.50%), F1-score (96.85%), mAP50 (98.10%), and mAP50-95 (80.50%). YOLOv8 ranked second, followed by YOLOv7 and YOLOv5, indicating that newer YOLO architectures offer improved accuracy. In terms of computational efficiency, YOLOv9 achieved the fastest inference time at 6.8 ms per image (approximately 147 FPS), making it the most suitable model for real-time deployment. Across all models, the non-plate category remained the most challenging, while white and yellow plates were occasionally misclassified under low-light conditions.

The findings have direct practical implications for Indonesia's e-tilang system, where accurate and fast license plate color classification is essential. Based on the comprehensive evaluation, YOLOv9 is recommended for deployment due to its best balance of accuracy and speed. However, this study has several limitations, including the dataset being collected from a single city (Bandar Lampung) under clear weather conditions, and the lack of evaluation under extreme weather conditions such as heavy rain, fog, or night-time. Future work should expand the dataset to more diverse geographical locations, explore ensemble methods to improve non-plate detection, and deploy the model on edge devices to validate real-world performance.

ACKNOWLEDGMENTS

The authors wish to thank the Basic Computing Laboratory, Department of Computer Science, Faculty of Mathematics and Natural Sciences, Universitas Lampung, and especially the Private Laboratory (2nd Floor, Computer Science Building, FMIPA Universitas Lampung), for their invaluable support and facilities.

LITERATURE

[1] M. Ashkanani, A. AlAjmi, A. Alhayyan, Z. Esmael, M. AlBedaiwi, and M. Nadeem, "A Self-Adaptive Traffic Signal System Integrating Real-Time Vehicle Detection and License Plate Recognition for Enhanced Traffic Management," *Inventions*, vol. 10, no. 1, 14, 2025, doi: https://doi.org/10.3390/inventions10010014.

[2] R. Sholehurrohman, K. Muludi, and J. Triloka, "Optimizing transportation surveillance with YOLOv7: Detection and classification of vehicle license plate colors," *Jurnal Ilmu Teknologi dan Komputer (JITK)*, vol. 10, no. 4, pp. 812–820, May 2025, doi: https://doi.org/10.33480/jitk.v10i4.6260.

[3] Kepolisian Negara Republik Indonesia, "*Peraturan Kepolisian Negara Republik Indonesia Nomor 7 Tahun 2021 tentang Registrasi dan Identifikasi Kendaraan Bermotor*," 2021. [Online]. Available: https://peraturan.bpk.go.id/Details/225016/perpol-no-7-tahun-2021.

[4] Kepolisian Negara Republik Indonesia, "*Peraturan Kepala Kepolisian Negara Republik Indonesia Nomor 5 Tahun 2012 tentang Registrasi dan Identifikasi Kendaraan Bermotor*," 2012. [Online]. Available: https://www.peraturanpolri.com/2015/12/peraturan-kapolri-nomor-05-tahun2012.html.

[5] R. Sholehurrohman, "Pengembangan YOLOv3 dengan Fitur Ekstraktor MobileNetV2 untuk Deteksi dan Klasifikasi Kendaraan Bergerak", Master's thesis, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia, 2021.

[6] R. Sholehurrohman and B. Setiyono, "Detection and classification of moving vehicle based on YOLOv3 model," *AIP Conference Proceedings*, vol. 2540, p. 080017, 2023, doi: https://doi.org/10.1063/5.0105877.

[7] E. S. dos Santos Júnior, T. Paixão, and A. B. Alvarez, "Comparative performance of YOLOv8, YOLOv9, YOLOv10, and YOLOv11 for layout analysis of historical documents images," *Applied Sciences*, vol. 15, no. 6, p. 3164, 2025, doi: https://doi.org/10.3390/app15063164.

[8] S. Dolhopolov, T. Honcharenko, D. Chernyshev, and O. Panina, "Advancing automated quality control in automotive manufacturing: A comparative analysis of YOLOv8, YOLOv9, and YOLOv10 for vehicle damage detection," in *Proc. 9th Int. Conf. Digital Technologies in Education, Science and Industry (DTESI)*, Oct. 2024. CEUR-WS.org/vol-3966/W3Paper10.pdf

[9] L. T. Ramos and A. D. Sappa, "A decade of you only look once (YOLO) for object detection," *arXiv preprint arXiv:2504.18586*, 2025, doi: https://doi.org/10.1109/ACCESS.2025.3630988.

[10] J. E. Gallagher and E. J. Oughton, "Surveying you only look once (YOLO) multispectral object detection advancements, applications, and challenges," *IEEE Access*, vol. 13, pp. 7366–7395, 2025, doi: https://doi.org/10.1109/ACCESS.2025.3526458.

[11] S. Farhadpour, T. A. Warner, and A. E. Maxwell, "Selecting and Interpreting Multiclass Loss and Accuracy Assessment Metrics for Classifications with Class Imbalance: Guidance and Best Practices," *Remote Sens*, vol. 16, no. 3, 533, 2024, doi: https://doi.org/10.3390/rs16030533.

[12] A. Panda, D. Panigrahi, S. Mitra, S. Mittal, and S. Rahimi, "Transfer Learning App to Computer Vision Problems: Survey on Current Progress, Limitations, and Opportunities," *arXiv:2409.07736*, 2024, doi: https://doi.org/10.48550/arXiv.2409.07736.

[13] A. A. Quraishi, F. Feyzi, A. Shahbahrami, "Detection and recognition of vehicle licence plates using deep learning in challenging conditions: A systematic review," *Int. J. Intell. Syst. Technol. Appl.,* vol. 22, no. 2, pp.105-150, 2024, doi: https://doi.org/10.1504/IJISTA.2024.139736.

[14] A. Alekseev, L. Kozhemyakin, V. Nikitin, and J. Bolshakova, "Data Preprocessing and Neural Network Architecture Selection Algorithms in Cases of Limited Training Sets—On an Example of Diagnosing Alzheimer's Disease," *Algorithms*, vol. 16, no. 5, 219, 2023, doi: https://doi.org/10.3390/a16050219.

[15] S. Dolhopolov, T. Honcharenko, V. Hots, and P. Kruk, "YOLOv8, YOLOv9, and YOLOv10: A study in automated vehicle damage detection," *in Proc. 4th Int. Workshop on Information Technologies: Theoretical and Applied Problems (ITTAP)*, Ternopil, Ukraine, Oct. 2024.

[16] R. Sholehurrohman, K. Sari, and A. Junaidi, "Color-based spot detection using automatic leaf segmentation in potato plants," *Integra: Journal of Integrated Mathematics and Computer Science*, vol. 2, no. 3, pp. 75–80, 2025, doi: https://doi.org/10.26554/integrajimcs.20252339.

[17] R. Khanam and M. Hussain, "What is yolov5: A deep look into the internal features of the popular object detector," *arXiv preprint arXiv:2407.20892*, 2024, doi: https://doi.org/10.48550/arXiv.2407.20892.

[18] Z. Yang, R. Xie, L. Liu, and N. Li, "Dense-YOLOv7: improved real-time insulator detection framework based on YOLOv7", *International Journal of Low-Carbon Technologies*, vol. 19, pp. 157–170, 2024, doi: https://doi.org/10.1093/ijlct/ctad122.

[19] A. Thakuria and C. Erkinbaev, "Improving the network architecture of YOLOv7 to achieve real-time grading of canola based on kernel health," *Smart Agricultural Technology*, vol. 5, p. 100300, Oct. 2023, doi: https://doi.org/10.1016/j.atech.2023.100300.

[20] V. Afifah and S. Erniwati, "Yolov8 for object detection: A comprehensive review of advances, techniques, and applications," *IJACI: International Journal of Advanced Computing and Informatics*, vol. 2, no. 1, pp. 53–61, 2026, doi: https://doi.org/10.71129/ijaci.v2i1.pp53-61.

[21] C. Y. Wang, I. H. Yeh, and H. Y. M. Liao, "Yolov9: Learning what you want to learn using programmable gradient information," *arXiv preprint arXiv:2402.13616*, 2024, doi: https://doi.org/10.48550/arXiv.2410.17725.

[22] M. Barr, "Coupling the Power of YOLOv9 with Transformer for Small Object Detection in Remote-Sensing Images," *Computer Modeling in Engineering & Sciences*, vol. 143, no. 1, pp. 593–616, 2025, doi: https://doi.org/10.32604/cmes.2025.062264.

[23] Z. Wang and J. Yang, "RE-YOLO: a lightweight small object detection method for UAV remote sensing imagery," *Annals of GIS*, vol. 32, no. 1, pp. 65-84, 2026, doi: https://doi.org/10.1080/19475683.2026.2617202.

[24] E. Pandilova, M. Petrov, V. Spasev, I. Dimitrovski, and I. Kitanovski, "Transfer Learning with Yolo for Object Detection in Remote Sensin," *in ICT Innovations 2024. TechConvergence: AI, Business, and Startup Synergy*; Risteska Stojkoska, B., Janeska Sarkanjac, S., Eds.; Communications in Computer and

© 2026 The Authors | doi: 10.23960/pepadun.v7i1.349

Information Science; Springer Nature: Cham, Switzerland, 2025; vol. 2436, pp. 121–135, doi: https://doi.org/10.1007/978-3-031-86162-8_9.

[25] S. Al-Saddik, M. E. Philip, and A. Haidar, "YOLO for Knowledge Extraction from Vehicle Images: A Baseline Study," *arXiv preprint arXiv:2507.18966*, 2025 Jul 25, doi: https://doi.org/10.48550/arXiv.2507.18966.