 https://doi.org/10.24191/jeesr.v25i1.011

Performance Analysis of the Yolov5 Model for Traffic Sign Detection

Nur Izzati Anaz Anizan, Fazlina Ahmat Ruslan, Noorfadzli Abd Razak, Mohd Azri Abdul Aziz and Juliana Johari*

***Abstract*—This study evaluates the performance of the YOLOv5 model in the detection of traffic signs under a diverse range of environmental conditions, assessing its performance through a comprehensive set of experiments. This study assesses the model's precision in identifying signage categories across a variety of lighting conditions and perspectives by employing a robust dataset that includes 1,596 images of a wide range of traffic signs. The model's ability to maintain high detection accuracy in optimal conditions is the primary focus of the analysis, which also emphasizes the challenges encountered in adverse lighting conditions such as direct sunlight and low-light settings in parking lots. The results indicate that YOLOv5 is highly reliable in unobstructed and clear conditions, but its reliability decreases in complex environments. This paper examines potential enhancements and future research directions, such as exploring of alternative model architectures and the implementation of advanced data augmentation techniques, to improve the adaptability and robustness of traffic sign detection systems.**

***Index Terms*— Environmental Conditions, Machine Learning, Real-World Applications, Traffic Sign Detection, YOLOv5**

I. INTRODUCTION

The rapid development of autonomous driving technology has generated a critical demand for resilient and effective traffic sign detecting systems. Road signs play a crucial role in providing essential information that ensures the secure and effective movement of cars [1]. Autonomous vehicles need to have the capability to identify and react to various road signs, including speed limits, stop signs, yield signs, and warning signals. This is crucial for complying with traffic regulations and preventing potential risks [2], [3].

Then, road sign detection that is effective improves the vehicle's situational awareness, allowinSg it to make properly informed decisions in real-time. Furthermore, in the context of traffic management systems, the accurate identification and understanding of road signs are important in improving traffic flow and minimising congestion. This is achieved by ensuring that all vehicles, either autonomous or controlled by humans, can operate in accordance with established traffic regulations. Therefore, advancements in traffic sign recognition technologies play a crucial role in shaping the future of intelligent transportation systems and ensuring safer road conditions. Accurate detection along with comprehension of these indicators is vital for the safety and reliability of autonomous vehicles [4], [5]. Nevertheless, the task of zzdeveloping a system that can accurately identify a wide range of road signs under various environmental circumstances remains a significant challenge.

This study introduces a methodology for identifying different types of traffic signs using YOLOv5. The objective of our study is to assess the efficacy of YOLOv5 in detecting road signs in various scenarios and to compare its performance with other established techniques. This research primarily focuses on doing a thorough assessment of YOLOv5's performance using a wide-ranging dataset of road signs, an analysis of the model's capabilities and implementations in autonomous driving systems in real-world scenarios.

The subsequent sections of this paper are structured in the following manner: Section 2 provides an overview of previous research conducted in detection object. Section 3 provides a comprehensive explanation of the approach, which includes specific information detailed description of the experimental configuration and the criteria used to assess the performance on the YOLOv5 model and the dataset that was used. Section 4 provides the results and investigation into the data, followed by a discussion. Section 5 provides a review of the results and offers ideas for future study, and last section ultimately concluding the piece of writing.

This study aims to evaluating the effectiveness of YOLOv5 in detecting various types of signage. The project involves analysing the performance of the YOLOv5 model on different signage categories and investigating the factors influencing its detection accuracy, such as signage type, lighting conditions, and image quality. This study examined the detection results from a dataset containing 1596 images of different signage types. The findings provide insights into the strengths and limitations of YOLOv5 for signage detection. Additionally, the results help researchers and developers identify key factors that impact the model's performance, guiding improvements and adaptations for better detection accuracy.

This manuscript is submitted on 15 August 2024, revised on 12 September 2024, accepted on 18 September 2024 and published on 31 October 2024. Nur Izzati Anaz Anizan, Juliana Johari, Fazlina Ahmat Ruslan, Noorfadzli Abd Razak and Mohd Azri Abdul Aziz are with the College of Engineering, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia.

*Corresponding author
Email address: julia893@uitm.edu.my

II. Literature Review

YOLO, an acronym for 'You Only Look Once,' is a revolutionary object detection method known for its speed and accuracy. Various studies have demonstrated its effectiveness across multiple domains, highlighting its versatility and potential for innovation in industrial applications [6].

YOLO's real-time processing rates make it very efficient for instant object identification in applications like autonomous driving, surveillance, and interactive systems. The broad use and ongoing development of YOLO may be attributed to its ability to effectively balance accuracy and efficiency. Multiple versions of YOLO have been developed to improve its performance and improve its capabilities in many sectors.

Therefore, YOLOv5 stands for “You Only Look Once version 5”. It is a state-of-the-art object detection algorithm that is designed to detect and recognize objects within an image or video frame in real-time [7], [8]. It is part of the YOLO series, which are renowned for their speed and accuracy in object detection tasks.

According to Sharma [9], the researcher employed YOLOv5 for face mask detection which is a critical task during the COVID-19 pandemic, highlighting the algorithm's effectiveness in distinguishing objects within video feeds. However, there are several constraints associated with this system. For instance, the system can reliably determine whether a person is wearing a mask or not, but only when the individual is directly facing the camera.

Zhao et al. [10] introduced a study on real-time detection of particleboard surface defects using an improved YOLOv5 model named PB-YOLOv5s. The precision-recall (PR) curve of the model demonstrated its strong performance across several error categories. The PB-YOLOv5s model demonstrated both accuracy and computational efficiency, exhibiting a significantly decreased inference time compared to other models. Efficiency is essential for the effective and immediate identification of defects in production situations. The improvements made to the YOLOv5 architecture, particularly designed for detecting defects in particleboard, had a significant role in the model's exceptional performance. The adjustments were made to improve the real-time performance and handle the extensive computational requirements of identifying errors.

In the agricultural sector, Jhatial et al. [11] applied YOLOv5 to detect rice leaf diseases, demonstrating its versatility in plant disease management. The model was trained for 100 epochs using a dataset which include images of four separate rice leaf diseases: Brown Spot, Bacterial Blight, Tungro and Blast. The confidence scores for disease detection were as follows: Brown Spot: 0.554, Bacterial Blight: 0.6, Tungro: 0.535, and Blast: 0.809. The research effectively confirmed the suitability of YOLOv5 for accurately diagnosing rice leaf diseases, achieving a high level of accuracy and recall. The model's performance indicators demonstrated its capacity for practical use in agriculture, particularly in the areas of disease control and crop health monitoring. Future work will concentrate on using this model in real-time situations to provide practical and useful information for farmers.

Therefore, Han et al. [12] extended its use to food quality inspection by employing YOLOv5 to assess the quality of cherries. By employing the flood filling algorithm for preprocessing, the YOLOv5 model achieved a high level of accuracy in a considerably shorter amount of training iterations (20 epochs) compared to other models that may require hundreds of iterations. The model attained a detection accuracy of 99.6% after only 20 training epochs, demonstrating its efficiency in training and improvement in detection accuracy. However, the accuracy of the model may still be affected by seasonal weather conditions and the complex variations in environmental factors, such as lighting and backdrop, if not well controlled.

The detection of unauthorized unmanned aerial vehicles (UAVs) using YOLOv5 and transfer learning, as explored by Al-Qubaydhi et al. [13] revealed its potential in enhancing UAV surveillance and security. The YOLOv5-based system successfully identified drones in photographs with various characteristics, including varies types, sizes, and backgrounds. This demonstrates its strong and efficient performance in real-time UAV identification applications. When compared to other models such as YOLOv3, YOLOv4, and Mask R-CNN, YOLOv5 demonstrated greater performance in terms of accuracy, recall, and mean Average accuracy (mAP) on both the training and testing datasets. The study highlighted the benefits of YOLOv5 in addressing difficulties associated with precisely detecting small moving objects and the need for real-time processing capabilities.

Moreover, Guo et al. [14] improved YOLOv5 by integrating transformer models to detect steel surface defects, highlighting the algorithm's adaptability and potential for innovation in industrial applications. To tackle the issues of cluttered background in defect images and the potential confusion of defect categories, researchers recommend integrating the TRANS module, which is based on the Transformer architecture, into the backbone and detecting head. MSFT-YOLO demonstrated enhanced detection accuracy, with a mean average precision (mAP) of 75.2% on the NEU-DET dataset. This is a 7% increase compared to the baseline YOLOv5 and an 18% increase compared to Faster R-CNN. Furthermore, the TRANS module is capable of effectively managing complex backgrounds, since it enhances the model's performance in situations including cluttered backgrounds and significant fluctuations in defect sizes. This makes it very reliable for industrial applications.

In this paper, a comparative assessment is carried out based on the performance of YOLOv5 in detecting various types of signage and the model's acceptance among users. A total of 1596 images, representing 12 different signage types, are used in this study. The images are sourced from various datasets to ensure diversity and comprehensiveness. The performance evaluation metrics and survey forms are used throughout the study to maintain consistency. The detection performance is analyzed across a staggered timeline, reflecting the model's progress and learning curve. Hence, this paper presents a comparative study of YOLOv5's detection capabilities based on a comprehensive dataset from various sources.

III. Methodology

The primary objective of this study is to evaluate the performance of object detection using the YOLOv5 model. The key objective is to evaluate the efficacy of YOLOv5 in accurately detecting and classifying various kinds of traffic signs. To do this, a comprehensive collection of traffic signs was gathered, including a wide range of forms, colors, and symbols that are regularly seen on roadways. The traffic signages serve as the objects of interest for detection, delivering an appealing test scenario to assess the model's capabilities.
The methodology of this study starts with the acquisition of data referring to traffic signages from several sources.

A. Data Collection

The data for the study was systematically gathered from several sources to provide a thorough and varied dataset. The main sources of data consist of footage obtained from cameras, photographs collected using mobile phones, and photos retrieved from Google Street View. This multifaceted methodology for data gathering enables the acquisition of a more comprehensive and diverse dataset, including traffic signs in various settings, lighting circumstances, and perspectives.

The dataset consists of a diverse collection of traffic signs frequently noticed on roadways. The signs include Stop sign, Bumper sign, Curve Left sign, Curve Right sign, No Entry sign, No Stopping sign, No Left Turn sign, One Way sign, Pedestrian sign, Turn Left Junction sign, Turn Right Junction sign, and Traffic Light sign. Each kind of sign was meticulously chosen to represent different traffic management methods and directional guidelines observed in practical driving situations. Among these signs, only the Stop sign contains a word, while the others consist solely of symbols.

Data collection spanned one month, involving collect a significant number of examples for every kind of signage to ensure that the dataset was both evenly distributed and indicative of the totality. This included carefully positioning cameras at various sites, manually taking photographs using mobile phones, and systematically relevant images from Google Street View. The outcome is a robust dataset that includes a broad range of traffic signs in various environments, serving as a strong basis for training and assessing the YOLOv5 model.

The comprehensive data collecting activity is vital for the study as it seeks to thoroughly evaluate the model's ability to reliably recognize and classify traffic signs [15]. The dataset's variety implies that the model can be assessed in different real-world circumstances, enhancing the study's conclusions' generalizability and practical applicability. The tabulation of the data is shown in Table I.

TABLE I. COLLECTION OF DATA FROM VARIOUS SOURCES

Sources	Total Images
Camera	429
Mobile phones	793
Google Street	374

Table I presents a table of the distribution of data obtained from different sources in this study. As illustrated in the table, the majority of the data was obtained using mobile phones, which contributed the highest number of images. This approach allowed for greater flexibility and ease in capturing traffic signages from multiple angles and in various locations.

In contrast, Google Street View provided the smallest amount of data. Although Google Street View offers a wide array of imagery from different geographic areas, the process of selecting relevant pictures was more time-consuming and less flexible when compared to using mobile phones. Consequently, this source made a lesser contribution to the entire dataset. The quantity of traffic signs are present for each sign in the dataset is shown in Table II.

TABLE II. THE QUANTITY OF TRAFFIC SIGNS

Traffic Sign	Total images
Stop	106
Bumper	157
Curve Left	124
Curve Right	157
No Entry	135
No Stopping	113
No Left Turn	108
One Way	146
Pedestrian Crossing	104
Turn Left Junction	146
Turn Right Junction	153
Traffic Light	147

According to Table II, the Bumper sign and Curve Right sign have the greatest number of traffic lights in the dataset. On the other hand, the Pedestrian Crossing sign has the lowest number of traffic lights, with just 104 images. The dataset is gathered with the purpose of undergoing a training procedure. There are a total sum of 1596 images including each kind of road sign.

B. Data Preprocessing

The data collected for this study is in JPEG format and has been universally resized to a resolution of 640x640 pixels to ensure equality in the size of the images. Standardization is essential for efficient processing and analysis using deep learning models. Next, the data is processed using Roboflow, a well renowned open-source application known for its effectiveness in image annotation [16]. Roboflow simplifies the annotation process by enabling users to manually outline the shape of the traffic signs in every image. This step is essential for training the YOLOv5 model to accurately detect and classify the various traffic signs.

Once the annotation process is finished, the dataset goes through data augmentation, which is an important step to improve the resilience and capacity of the model to apply to various situations. Roboflow provides a variety of data augmentation methods, such as flipping, rotation, scaling, and cropping. For this study, precise augmentation parameters were chosen to replicate real-world changes and enhance the model's performance. The photos were enhanced by applying rotations within the range of -3 to +3 degrees, adjusting brightness between -20% and +20%, and introducing noise that affected up to 0.28% of the pixels. The selection of these strategies was based on the objective of generating a training dataset that encompasses a wider range of variables often observed in real-life situations, including differences in angles, lighting conditions, and minor imperfections.

Following augmentation, the dataset is divided into three subsets: training, validation, and test sets. Dividing the data is essential for training the model, evaluating its performance throughout development, and assessing its ultimate accuracy. More specifically, 70% of the gathered data is assigned to the training set, which the model utilizes to acquire knowledge and detect patterns in traffic signs. Another 20% of the data is allocated for the validation set, which is used to refine the model and avoid overfitting by assessing its performance on unseen data during training. The remaining 10% comprises the test set, which offers an impartial assessment of the model's performance on entirely original data, validating it is ready for implementation in real-world applications. The dataset in Fig. 1 consists of the training dataset.

Fig. 1. The examples of training dataset.

The structured approach used for data processing, annotation, augmentation, and splitting guarantees that the YOLOv5 model is adequately trained to precisely detect and categorize traffic signs in various circumstances, hence enhancing the safety and effectiveness of traffic control systems.

C. Model Selection and Architecture

YOLOv5, a widely used model for detecting objects, comes in many variants such as YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. Each version is designed to achieve an ideal balance between accuracy and speed, suited to meet the specific needs of applications [17]. For this study, YOLOv5s version has been utilized. Despite its lower precision compared to the more comprehensive and computationally demanding YOLOv5x, YOLOv5s has been selected for its compact size and high-speed performance.

Although YOLOv5s may not attain the best level of accuracy compared to other YOLOv5 variations, it stands out in terms of detection speed [18]. This makes it especially well-suited for real-world situations where rapid processing is important. In applications such as real-time traffic signs detection, surveillance, and autonomous driving, the capacity to rapidly interpret and react to visual inputs is crucial . The YOLOv5s model has the benefit of fast inference times, allowing for prompt detection and decision-making.

When deploying YOLOv5s, it is important to carefully weigh the balance between accuracy and speed. Although the model may not achieve the same level of accuracy as YOLOv5x in detecting objects, its faster performance makes it more suitable for real-time use in dynamic contexts. This equilibrium is crucial in situations when the delay might influence both safety and operational effectiveness. YOLOv5s prioritizes speed to maintain system responsiveness and effectiveness [7], even if it results in a little reduction in detection accuracy.

The illustration shows a usual object detection design with one- and two-stage detectors. The flow begins with an Input layer that feeds data into the Backbone, which extracts features using stacked convolutional layers [4]. The extracted features are then given to the Neck, which creates feature pyramids to facilitate multi-scale object recognition. For One-Stage Detectors make predictions (both dense and sparse) directly from feature maps using anchor boxes, providing bounding boxes and class predictions all at once. On the other hand, Two-Stage Detectors refine predictions by further processing the data before final classification, which improves accuracy but usually comes at the expense of speed. This architecture maintains a compromise between real-time detection performance and reliable object detection.

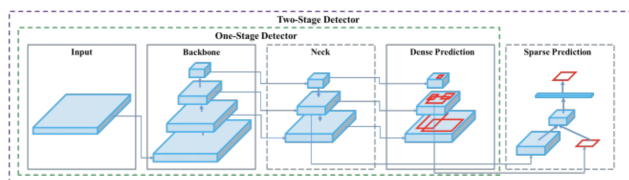

Fig. 3. The architecture of the Yolov5.

D. Training Process

To appropriately train the YOLOv5 model, it is essential to carefully choose the suitable hyperparameters. This is necessary to get the best possible performance and to avoid overfitting. Overfitting happens when the model excessively captures every detail of the training data, including its noise and outliers, hence affecting its performance on original, unseen data. Optimizing the hyperparameters is crucial for building a well-balanced model that can effectively generalize.

The selected hyperparameters for this training consist of a batch size of 16 and 100 epochs. The model's batch size of 16 signifies that it analyzes 16 pictures simultaneously while altering the weights. This size is selected with the aim of attaining a balance between the amount of memory being used and the efficiency of the computing process. Reducing the number of batches used in training might result in poor gradient approximations, whilst increasing batch sizes can enhance gradient approximations but need more memory. A batch size of 16 achieves a harmonious equilibrium, delivering consistent updates while being compatible with standard hardware limitations.

Training the model for 100 epochs requires looping the whole dataset through the model 100 times. This number of epochs is appropriate as it offers numerous chances for the model to acquire knowledge from the data without excessively intensive training durations. Inadequate number of epochs may lead to underfitting, characterized by insufficient learning from the data, while excessive number of epochs can result in overfitting. Evaluating the model's performance on the validation set ensures that the selected number of epochs is appropriate.

The training phase starts with loading the data, in which the annotated and enhanced dataset is retrieved and stored in the memory of the computer. The dataset comprises photos together with their related annotations, which provide precise information on the positions and classes of the items that need to be identified. The data loading stage ensures that the photos are appropriately structured and prepared for the training process.

After the data has been loaded, the training process begins. During the training process, the YOLOv5 model is supplied with enhanced and annotated data. The model progressively updates its weights and biases via backpropagation over ongoing epochs. Data augmentation methods, such as flipping, rotation, scaling, and brightness modifications, are used to increase the variety of the dataset. These enhancements improve the model's ability to generalize by emulating variances that occur in the actual world.

Validation is conducted periodically throughout training to assess the model's performance. The validation dataset, which is separate from the training data, is used to assess the accuracy of the model and other metrics. This approach assists in identifying overfitting, guaranteeing that the model does not excessively rely on the training data but instead displays good generalization to new data. Modifications to the model and hyperparameters may be implemented according on the validation performance in attempt to enhance the outcomes.

Once the training step is over, the final model is evaluated using a separate dataset. The dataset is only utilized for the final assessment and is not provided to the model during the training or validation stages. Testing entails executing the model on the test pictures and contrasting the anticipated bounding boxes and class labels with the accurate annotations of the ground truth. Key performance indicators, including precision, recall, F1-score, and mean Average Precision (mAP) [19] are computed to thoroughly evaluate the accuracy and resilience of the model.

IV. Results and Discussions

This study proposes to assess the effectiveness of object detection using the YOLOv5 model. The findings of this study rely on essential performance indicators, including Precision, Recall, F1-Score, and Accuracy. These indicators enable developers and researchers to optimize their models, assuring their performance in real-world scenarios and alignment with the requirements of their deployment environment.

A. *The Confusion Matrix*

A confusion matrix is used to assess the performance of machine learning classification algorithms. The display shows the number of true and false predictions by comparing the real values with the predicted values. Figure 3 illustrates how the confusion matrix offers valuable information about the model's accuracy in classifying every class. It helps in identifying any weaknesses in the model's performance across multiple classes [20].

In a confusion matrix, diagonal elements represent the classification results within the classes. These cells, flowing from the top left to the bottom right, are crucial for understanding a model's accuracy. They indicate the number of instances where the predicted label matches the true label, showing that the model correctly identified the class. In this matrix, the cell that corresponds to the label "Stop" in both the actual and predicted labels displays a value of 87, indicating that the model accurately identified 87 occurrences of "Stop" signs. Additionally, there were 135 occurrences when the "No Entry" signage was accurately identified as "No Entry."

High values on the diagonal suggest that the model is accurately predicting the outcomes for that specific class. As an example, the value of the diagonal element corresponding to the "Traffic Light" class is 147. This indicates that the model has a high level of accuracy in predicting this particular class. If a diagonal element has a low value compared to the total number of occurrences for a certain class, it suggests that the model is encountering difficulties in accurately predicting that particular class. For case, out of the total 157 instances of "Curve Right," only 122 were properly identified as diagonal. This indicates that there were 35 cases as "Curve Right" signs were misclassified. In addition, the presence of the "Curve Left" measure is shown by the diagonal element, which accounts for 118 out of a total of 124 cases, demonstrating a high level of accuracy for this particular class. There were just six instances that were categorised incorrectly.

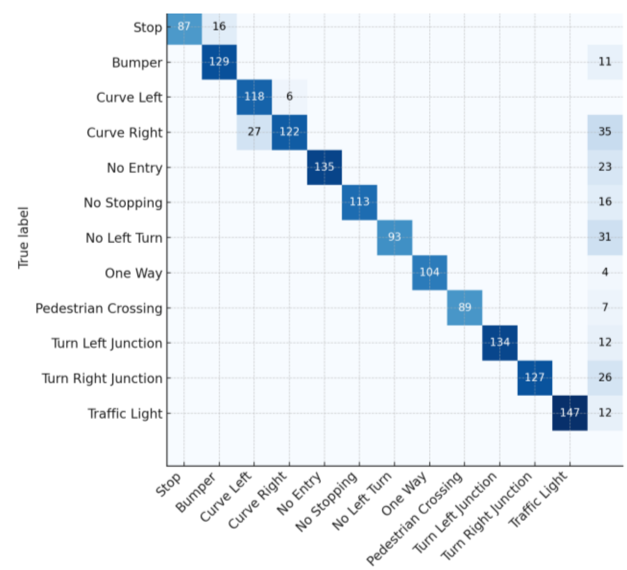

Fig. 3. The Confusion Matrix for all classes.

The off-diagonal columns in a confusion matrix indicate instances of misclassification by the model. These elements emphasise the characteristics and magnitude of the model's inaccuracies. Each non-diagonal element indicates a discrepancy between the predicted class by the model and the actual class. Higher values imply a higher frequency of misclassifications between certain classes, whereas lower values signal that the model is more proficient at separating between those classes.

In the matrix, there were 10 instances where the prediction "Curve Right" was inaccurately identified as "Curve Left," and 26 instances where the prediction "Turn Right Junction" was inaccurately identified as "Turn Left Junction." This shows the model's difficulty in differentiating between these two similar signs that seem extremely similar in their visual characteristics since the both signs have a bright yellow background, as is usual for warning signs, and a black symbol to indicate the direction of the curve. The persistent employment of these colours for both signs makes them visually identical, particularly when the model emphasises colour as a differentiating element. Both signs are diamond-shaped, which is a frequent design for warning signs. Considering the two signs have the same shape, the model cannot differentiate between them using the outline. Also, the "Curve Right" and "Curve Left" signs are essentially mirror images of each other. This symmetry can be particularly tough for a model, as it could be unable to differentiate between the two directions without a solid understanding of the arrow's unique orientation.

Furthermore, the model had a challenge in distinguishing between 31 instances of "No Left Turn" signs and misclassifying them as "No Entry." This implies that the model has difficulty in detecting visual or contextual similarities between these two types of signs since both signs are circular which is frequently used for restriction signs in traffic management and clearly display the colours red and white. The border colour is red, which represents restriction or warning, while white is used as the background in the "No Entry" sign and as a visual indication in the "No Left Turn" sign. This similar colour scheme might cause the model to focus on the dominant red colour, which is shared by both signs, rather than the unique elements that distinguish them.

B. The Performance Matrix on Test Set

The confusion matrix provides key metrics for evaluating a classification model's performance, including Accuracy, Precision, Recall, and F1-Score. Accuracy measures overall correctness, Precision assesses the accuracy of positive predictions, and Recall (or sensitivity) focuses on identifying all positive instances. The F1-Score combines Precision and Recall, offering a balanced measure.

As shown in Table III, the model's overall accuracy is generally high, with most classes achieving scores over 0.8. This suggests that the model is proficient at accurately categorizing road signs. However, it’s important to note that accuracy can be misleading, especially in imbalanced datasets. If one class dominates the dataset, the model might achieve high accuracy by simply predicting that class for most cases.

TABLE III. PERFORMANCE MATRIX FOR ALL CLASSES

Traffic Signs	Accuracy	Precision	Recall	F1-Score
Bumper	0.89	0.89	0.89	0.89
Curve Left	0.95	0.95	0.95	0.95
Curve Right	0.78	0.78	0.78	0.78
No Entry	1.00	1.00	1.00	1.00
No Left Turn	1.00	1.00	1.00	1.00
No Stopping	1.00	1.00	1.00	1.00
One Way	0.86	0.86	0.86	0.86
Pedestrian Crossing	0.71	0.71	0.71	0.71
Stop	0.53	0.53	0.53	0.53
Traffic Light	0.86	0.86	0.86	0.86
Turn Left Junction	0.92	0.92	0.92	0.92
Turn Right Junction	0.83	0.83	0.83	0.83

Upon analysing Table III, it's evident that the Precision values closely correspond to the Accuracy values for most classes. This indicates that when the model predicts a given class, it is often correct. However, for classes with lower accuracy scores, such as "Stop," the model is more prone to producing false positives—mistakenly classifying items as "Stop" even when they are not. This might be due to the "Stop" sign's particular text-based design, compared to standard symbol-based traffic signs. In contrast, the "No Entry," "No Left Turn," and "No Stopping" signs achieve high accuracy because their distinct symbols and colours make them easily distinguishable from other objects in images, allowing the model to recognize them more effectively. Overall, the variations in performance indicate that while the model is effective, further enhancements could help improve the detection of more challenging traffic signs like "Pedestrian Crossing" and "Stop" signs.

Recall measures the model's ability to detect all positive instances by computing the ratio of correctly predicted positives to the total actual positives. Like Precision, Recall often correlates with Accuracy, showing the model's proficiency in identifying examples across categories. However, lower Recall scores, such as in "Pedestrian Crossing," indicate the model may miss some examples.

F1-Scores balance Precision and Recall, reflecting patterns in Accuracy and Recall. High Accuracy and Recall result in high F1-Scores, indicating strong performance, while lower scores highlight areas needing improvement.

The model demonstrates robust performance across most classes, as evidenced by the high Accuracy, Precision, Recall, and F1-Score metrics. However, certain classes, like "Stop," exhibit significantly poorer performance metrics, suggesting that the model may have difficulty precisely identifying and categorizing "Stop" signs. While the model shows high Recall for recognizing positive instances, it may sometimes mistakenly classify other objects as the target class, resulting in lower Precision, as seen in some classes.

The visualizations provide valuable insights into the model's efficiency in identifying signs across different real-life scenarios. By analysing these detections, we can evaluate the model's capabilities and limitations, particularly in its ability to adapt to varied situations, including changes in lighting, obstructions, perspective angles, and object sizes.

As depicted in Fig. 4, the final model was evaluated using the test dataset during the final assessment of the training process. The model exhibits high confidence under clear conditions, as indicated by its strong performance and high confidence ratings (e.g., "No Stopping 0.9", "Curve Right 0.9"). This suggests that the model performs exceptionally well in environments with clear and well-illuminated signage.

Fig. 4. The final model has been evaluated by test dataset at final assessment of training process.

Therefore, the model has the ability to identify various types of signs, from simpler ones like "Stop" signs to more complex designs like "Pedestrian Crossing." This indicates a strong capability to generalize learned knowledge across different classes. Additionally, many images demonstrate the model's ability to accurately identify signs from various angles, which is crucial for real-world applications, especially in vehicles where the viewpoint can change significantly.

Nevertheless, the model shows inconsistency when operating in complex environments. Some images have lower confidence scores in challenging situations, such as the "Pedestrian Crossing 0.6" prediction, which may suggest uncertainty due to factors like partial obstructions, intricate backgrounds, or insufficient lighting conditions.

C. Practical Implications

The system is tested in real-world scenarios to evaluate its accuracy using a vehicle equipped with specialized equipment. The vehicle is fitted with two high-definition cameras positioned at the front, one on the right and one on the left, designed to capture a wide range of visual data from different angles. These cameras are connected to a laptop configured to execute the model in real-time. This setup allows for the continuous evaluation and processing of visual data as the vehicle navigates through various environments.

Testing is conducted in two specific types of locations: an open space and a parking lot. These locations are chosen to provide a range of challenges and factors that might impact the model's performance, such as variations in lighting conditions, different types of obstructions, and unique vehicle movements. Tests are performed at noon in both locations to maintain consistent lighting conditions and to assess the model's effectiveness in handling direct sunlight, which can cause glare and shadows that the system must manage effectively. This thorough testing methodology helps refine the model, ensuring its reliability and accuracy in practical applications.

According to Fig. 5, the display shows perfect performance in open areas with unobstructed and uniformly illuminated signs. However, it faces challenges when exposed to direct sunlight, leading to high exposure conditions that can significantly affect detection accuracy. Direct sunlight often results in glare or overexposed images, making it difficult to distinguish critical features on the signs. This issue is particularly noticeable in situations with varying lighting conditions, as reflected by the differing levels of confidence in detecting the same types of signs under different lighting environments. These observations suggest that the model's performance is sensitive to changes in ambient lighting, potentially leading to inconsistent detections.

Fig. 5. The point of view of the dual camera and the screen when detection is on.

In contrast, the detection scenarios in parking lots present a different set of challenges, primarily due to inadequate lighting. The presence of shadows and low-light circumstances in such could reduce the visibility of signs, making it more difficult to accurately identify them. The algorithm also seems to have difficulty in accurately identifying smaller or more distant signs, which are more common in crowded parking spaces. The challenge of dealing with objects of different sizes and distances often leads to lower confidence scores and possible misclassifications.

As seen in Fig.6, a Stop sign is incorrectly identified as a No Entry sign. This is likely because the Stop sign is not clearly visible, and the model struggles to differentiate between signals with similar red color in low lighting situations. This inaccuracy emphasizes the need for the model to more effectively adjust to various and ever-changing surroundings, where lighting conditions and sign visibility might change significantly. The "Stop" sign was partially obscured by shadows, and its red color was not distinctly visible, leading the model to confuse it with the similarly colored "No Entry" sign.

In order to improve the model's resilience and dependability, it would be advantageous to include sophisticated lighting correction algorithms that can dynamically modify the image's brightness and contrast. Enhancing the model's generalization skills might be achieved by training it on an additional wide dataset that covers a diverse range of lighting situations, sign sizes, and distances. Furthermore, the use of methods such as scale-invariant feature transformations might enhance the ability to accurately identify signs regardless of their dimensions and accessibility, hence assuring consistent performance in diverse real-world situations.

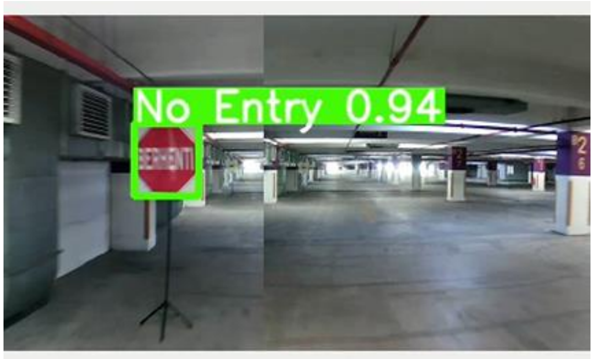

Fig. 6. The misclassification with other signs.

V. CONCLUSION

The objective of this study was to assess and enhance the performance of a detection model, namely YOLOv5, designed to identify traffic signs in different scenarios. The assessment centered on assessing the model's performance in several situations, including variable lighting conditions, occlusions, and environmental challenges. This was achieved using both qualitative and quantitative studies.

The model's performance was superior in situations that were well-illuminated as well as free from obstacles, allowing for clear and controlled detection of traffic signs. The system demonstrated resilience in identifying a variety of signs from varied perspectives, showcasing its training and architectural capabilities. Nevertheless, the model had difficulties when faced with intense lighting conditions and challenges, resulting in decreased precision and accuracy. Signs that were of smaller size, located at a greater distance, or partly hidden were not consistently identified, indicating a need for improved processing skills and training methods.

The study emphasized the critical need for flexible and resilient traffic sign recognition systems in the changing environment of autonomous driving and smart city management. Although the YOLOv5 model provides a solid foundation, ongoing enhancements along with adaptations are necessary to satisfy the growing requirements of real-world applications.

VI. FUTURE WORK

The YOLOv5 model exhibited high precision in optimal, transparent situations. However, it indicated inconsistency in problematic lighting circumstances, such as direct sunlight and low-light settings like parking lots. Both the confusion matrix and the precision-recall analysis clearly showed a drop in confidence and an increase in misclassifications under inadequate settings.

To enhance the model, advanced data augmentation approaches may be used. These techniques focus on simulating dynamic weather conditions like rain, fog, and snow during training. By doing so, the model's ability to handle environmental changes is improved. Hence, geometric transformations may also be implemented to the model. Incorporate advanced geometric transformations, such as rotations, scaling, and translations, to enhance the model's ability to generalize to signs seen from various angles and distances.

Furthermore, ensemble models may be implemented by employing the advantages of several architectures, such as YOLOv5 with Faster R-CNN or SSD, to improve detection precision and minimize the probability of misclassifications.

Future developments should leverage the insights gained from this study to enhance the model's practical application and ensure its reliability in real-world scenarios, future research should prioritize improving its robustness against operational challenges and expanding its capabilities to accommodate dynamic conditions. The insights gained from this study provide a solid foundation for achieving these objectives. Hence, expand the range of testing conditions to include various weather conditions, nighttime driving, and other challenging scenarios to a deeper analysis of the model.

ACKNOWLEDGMENT

The authors express their gratitude as well as appreciation to the School of Electrical Engineering, College of Engineering, Universiti Teknologi MARA Shah Alam for their assistance.

REFERENCES

[1] M. Biagiola, S. Klikovits, J. Peltomaki, and V. Riccio, “SBFT Tool Competition 2023 - Cyber-Physical Systems Track,” in *Proceedings - 2023 IEEE/ACM International Workshop on Search-Based and Fuzz Testing, SBFT 2023*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 45–48. doi: 10.1109/SBFT59156.2023.00010.

[2] P. Jing, H. Hu, F. Zhan, Y. Chen, and Y. Shi, “Agent-based simulation of autonomous vehicles: A systematic literature review,” *IEEE Access*, vol. 8, pp. 79089–79103, 2020, doi: 10.1109/ACCESS.2020.2990295.

[3] D. J. Fagnant and K. M. Kockelman, “The travel and environmental implications of shared autonomous vehicles, using agent-based model scenarios,” *Transp Res Part C Emerg Technol*, vol. 40, pp. 1–13, 2014, doi: 10.1016/J.TRC.2013.12.001.

[4] I. Siniosoglou, P. Sarigiannidis, Y. Spyridis, A. Khadka, G. Efstathopoulos, and T. Lagkas, “Synthetic Traffic Signs Dataset for Traffic Sign Detection & Recognition in Distributed Smart Systems,” *Proceedings - 17th Annual International Conference on Distributed Computing in Sensor Systems, DCOS 2021*, pp. 302–308, 2021, doi: 10.1109/DCOSS52077.2021.00056.

[5] N. Bhatt, P. Laldas, and V. B. Lobo, “A Real-Time Traffic Sign Detection and Recognition System on Hybrid Dataset using CNN,” *7th International Conference on Communication and Electronics Systems, ICCES 2022 - Proceedings*, pp. 1354–1358, 2022, doi: 10.1109/ICCES54183.2022.9835954.

[6] R. Mulajkar and S. Yede, “YOLO Version v1 to v8 Comprehensive Review,” *7th International Conference on Inventive Computation Technologies, ICICT 2024*, pp. 472–478, 2024, doi: 10.1109/ICICT60155.2024.10544452.

[7] R. Nachet and T. B. Stambouli, “A Real Time Object Detection System for Infant Safe Sleep Based on YOLOv5 Algorithm,” *2022 3rd International Conference on Embedded and Distributed Systems, EDiS 2022*, pp. 16–21, 2022, doi: 10.1109/EDIS57230.2022.9996513.

[8] H. Wu, X. Zhao, J. Qiao, and Y. Che, “An Improved YOLOv5 Algorithm for Elderly Fall Detection,” *Proceedings - 2023 International Conference on Innovation, Knowledge, and Management, ICIKM 2023*, pp. 83–88, 2023, doi: 10.1109/ICIKM59709.2023.00024.

[9] Vinay Sharma, “Face Mask Detection using YOLOv5 for COVID-19,” 2020.

[10] Z. Zhao, X. Yang, Y. Zhou, Q. Sun, Z. Ge, and D. Liu, "Real-time detection of particleboard surface defects based on improved YOLOV5 target detection," *Sci Rep*, vol. 11, no. 1, Dec. 2021, doi: 10.1038/s41598-021-01084-x.

[11] Juman Jhatial Muhammad *et al.*, "Deep Learning-Based Rice Leaf Diseases Detection Using Yolov5," *Sukkur IBA Journal of Computing and Mathematical Science*, vol. 6, no. 1, 2022.

[12] W. Han, F. Jiang, and Z. Zhu, "Detection of Cherry Quality Using YOLOV5 Model Based on Flood Filling Algorithm," *Multidisciplinary Digital Publishing Institute, Foods*, vol. 11, no. 8, Apr. 2022, doi: 10.3390/foods11081127.

[13] N. Al-Qubaydhi *et al.*, "Detection of Unauthorized Unmanned Aerial Vehicles Using YOLOv5 and Transfer Learning," *Electronics (Switzerland)*, vol. 11, no. 17, Sep. 2022, doi: 10.3390/electronics11172669.

[14] Z. Guo, C. Wang, G. Yang, Huang, and G. Li, "MSFT-YOLO: Improved YOLOv5 Based on Transformer for Detecting Defects of Steel Surface," *Multidisciplinary Digital Publishing Institute,, Sensors*, vol. 22, no. 9, May 2022, doi: 10.3390/s22093467.

[15] Q. Lin, S. Zhang, and S. Xu, "Construction of Traffic Moving Object Detection System Based on Improved YOLOv5 Algorithm," *Proceedings - 2023 2nd International Conference on 3D Immersion, Interaction and Multi-Sensory Experiences, ICDIIME 2023*, pp. 268–272, 2023, doi: 10.1109/ICDIIME59043.2023.00057.

[16] S. Alexandrova, Z. Tatlock, and M. Cakmak, "RoboFlow: A flow-based visual programming language for mobile manipulation tasks," *Proc IEEE Int Conf Robot Autom*, vol. 2015-June, no. June, pp. 5537–5544, Jun. 2015, doi: 10.1109/ICRA.2015.7139973.

[17] A. Santos, P. A. Abu, C. Oppus, and R. Reyes, "Real-time traffic sign detection and recognition system for assistive driving," *Advances in Science, Technology and Engineering Systems*, vol. 5, no. 4, pp. 600–611, 2020, doi: 10.25046/AJ050471.

[18] J. Ning and J. Wang, "Automatic Driving Scene Target Detection Algorithm Based on Improved YOLOv5 Network," *Proceedings - 2022 International Conference on Computer Network, Electronic and Automation, ICCNEA 2022*, pp. 218–222, 2022, doi: 10.1109/ICCNEA57056.2022.00056.

[19] K. Li, Z. Huang, Y. C. Cheng, and C. H. Lee, "A maximal figure-of-merit learning approach to maximizing mean average precision with deep neural network based classifiers," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pp. 4503–4507, 2014, doi: 10.1109/ICASSP.2014.6854454.

[20] L. E. Pomme, R. Bourqui, R. Giot, and D. Auber, "Relative Confusion Matrix: Efficient Comparison of Decision Models," *Proceedings of the International Conference on Information Visualisation*, vol. 2022-July, pp. 98–103, 2022, doi: 10.1109/IV56949.2022.00025.