






Assessment of Deep Learning Models for Pavement Distress Detection in High-Resolution UAV Imagery

Mohamed E. Mohamed^{1,*}; Adel A. Esmat²; Ahmed M. Hamdy³; Amr M. El Sheshtawy⁴

1. Assistant lecturer, Department of Civil Engineering, Faculty of Engineering, Al-Azhar University, Cairo, Egypt

2. Professor, Department of Civil Engineering, Faculty of Engineering, Al-Azhar University, Cairo, Egypt

3. Associate professor, Department of Civil Engineering, Faculty of Engineering, Al-Azhar University, Cairo, Egypt

4. Assistant professor, Department of Civil Engineering, Faculty of Engineering, Al-Azhar University, Cairo, Egypt

* Corresponding author: Mohamedeid.2314@azhar.edu.eg

ARTICLE INFO

Article history:

Received: 08 August 2025

Revised: 14 September 2025

Accepted: 21 November 2025

Keywords:

YOLOv8;

YOLOv5;

Pavement distress detection;

Low-resource environments.

ABSTRACT

Pavement distress detection is a critical task for ensuring road safety and maintaining transportation infrastructure, particularly in environments with limited resources. This study conducts a comprehensive evaluation of ten YOLO object detection models comprising five YOLOv5 and five YOLOv8 variants (n, s, m, l, x) for identifying seven pavement defect classes using 500 high-resolution UAV images. The dataset, manually annotated and split into 70% training and 30% validation sets, was used to train all models under uniform hyperparameter settings. The performance was assessed using standard metrics: mAP@0.5:0.95, mAP@0.5, Recall, Precision and F1-score. Results showed that YOLOv8 consistently outperformed YOLOv5 across all dataset sizes, with YOLOv8l reaching the highest mAP@0.5:0.95 of 0.381, and YOLOv8m providing the best balance between accuracy (0.344), training time (67 minutes), and robustness. Statistical validation using Two-Way ANOVA confirmed significant performance differences ($F = 13.81$, $p = 0.0006$, Cohen's $d = 0.73$). Further analysis using Repeated Measures ANOVA and Bonferroni-corrected t-tests reinforced YOLOv8l's superiority over other variants. Despite the limited dataset size the findings demonstrate YOLOv8's effectiveness and reliability in low-resource conditions without the need for image augmentation or preprocessing. The main innovation lies in providing a statistically validated benchmark for UAV-based pavement monitoring under small dataset conditions, highlighting YOLOv8's superior efficiency and generalization without augmentation or preprocessing. This study provides a reproducible benchmark for real-time UAV-based pavement distress detection, offering insights for deploying lightweight deep learning systems in resource-constrained settings.

E-ISSN: 2345-4423

© 2025 The Authors. Journal of Rehabilitation in Civil Engineering published by Semnan University Press.

This is an open access article under the CC-BY 4.0 license. (<https://creativecommons.org/licenses/by/4.0/>)

How to cite this article:

Mohamed, M. Eid abdelshakour, Esmat, A. A., Hamdy, A. M. and El Sheshtawy, A. M. (2026). Benchmarking YOLOv8 Against YOLOv5 for Pavement Distress Detection in High-Resolution UAV Imagery: A Photogrammetric Analysis. (2415). Journal of Rehabilitation in Civil Engineering, 14(2), 2415 <https://doi.org/10.22075/jrce.2025.2415>

1. Introduction

Pavement distresses represent a critical challenge for transportation infrastructure, compromising both safety and operational efficiency. Traditional manual inspection techniques suffer from subjectivity, labor intensity, and time constraints [1]. In contrast, UAV-based systems offer distinct advantages including: (1) rapid area coverage without traffic disruption, (2) consistent data quality through standardized flight parameters, and (3) enhanced safety by eliminating human exposure to hazardous conditions. To complement efficient defect detection with reliable condition assessment, machine learning techniques such as Artificial Neural Networks (ANNs) have been vastly used to predict overall pavement performance metrics. Artificial Neural Networks (ANNs) are implemented to determine the Pavement Condition Index (PCI) for flexible pavement distress using the Long-Term Pavement Performance (LTPP) dataset [2]. Furthermore, results from related studies show that Neural Network models are effective in estimating the impact of various influential factors on pavement service life. The outputs derived from these models can support informed decisions related to rehabilitation planning, overlay design, budget allocation, and maintaining infrastructure sustainability [3]. In addition, recent research efforts have focused on developing improved predictive models for estimating the elastic modules of flexible pavements based on traffic volume, pavement age, temperature, and other parameters. Such models address typical pavement distresses, involving rutting, cracking, and surface deformation, by providing more reliable predictions using Artificial Neural Networks, ultimately supporting effective maintenance strategies and extending pavement service life [4]. While these contributions have advanced pavement-specific modeling, they remain part of the broader field of Structural Health Monitoring (SHM). SHM techniques cover a wide range of approaches, including sensor-based [5], signal-based [6], imaging [7], and deep learning frameworks [8]. While sensor- and vibration-based methods have proven effective in building and industrial applications [9], they are often costly and less scalable for transportation infrastructure. For pavements and road networks, imaging methods especially UAV-based deep learning are more practical and cost-efficient, offering scalable solutions for large-area monitoring and timely detection of surface distresses. Within this context, YOLO-based frameworks emerge as a powerful option, bridging the gap between efficient UAV imaging and automated defect recognition. YOLOv8's advanced architecture enhances detection accuracy and efficiency, making it suitable for resource-constrained UAV applications. The research contributes a first-of-its-kind statistical comparison between YOLOv5 and YOLOv8 architectures under controlled, low-resource conditions, utilizing rigorous ANOVA testing to ensure robust performance evaluation. It is hypothesized that YOLOv8 variants will outperform YOLOv5, especially in accurately identifying complex distress patterns when trained on limited datasets.

1.1. YOLO-Based deep learning approaches for pavement distress

The YOLO framework has been effectively applied to pavement distress detection, particularly crack identification. Zhu (2022) employed UAV imagery and trained three convolutional models YOLOv4, YOLOv3, and RCNN to compare their detection performance [10].

YOLOv8 was employed to detect and classify seven pavement defect types, involving transverse, longitudinal, alligator, and oblique cracks, as well as potholes and repairs, using a dataset of 5,796 aerial and terrestrial images. The model's lightweight, multi-scale architecture enabled effective extraction of high-level semantic features. It achieved strong performance with a mAP of 79%, F1-score of 74%, accuracy of 81%, precision of 77%, and recall of 75%. Challenges included small defect sizes, poor contrast, shadows, and overlapping distress types. A 70/30 train-test split was used to ensure balanced evaluation [11].

In earlier studies, only a few pavement distress types were addressed, overlooking common issues like repairs and delamination. Zhu et al. (2022) utilized UAV imagery to detect six pavement distress categories: transverse, longitudinal, oblique cracks, alligator, repairs, and potholes. Three CNN-based

object detectors (YOLOv3, YOLOv4, and Faster R-CNN) were tested on a pavement distress dataset. YOLOv3 performed best with 56.6% accuracy, though improvements remain necessary [12].

Yan et al. utilized three SSDs, an enhanced SSD, and YOLOv4 in 2021 for detecting pavement Cracks in UAV images. Five different types of distress were selected in this study: pavement markings, longitudinal, transverse, map cracks, and repairs. According to the results, the estimated model's mAP value for all test dataset categories was 85.11%, which was 0.55% and 10.4% higher than that of the original SSD and YOLO, respectively [13].

In 2019, Song et al. used Faster R-CNN to identify four types of pavement distress: cracks, oil stains, potholes, and spot surfaces, utilizing a dataset of 6,498 images. Their model achieved 90% accuracy and 89% detection performance. However, collecting the data was expensive and time-intensive, and other types of damage, like patches, were not included in the study [14].

In 2020, Silva et al. identified pavement potholes using a deep learning network with UAV imagery, achieving over 95% accuracy. However, other distress types, such as cracks, repairs, and delamination, were not considered [15].

SSD-TensorFlow achieved 73% accuracy and 32.5% mAP, while YOLOv3 reached 84% accuracy, 78% recall, and 83.62% mAP. YOLOv4 performed best with 85% accuracy, 81% recall, 85.39% mAP, and superior speed [16].

In 2021, Shu et al. applied YOLOv5 to crack detection on the Street View dataset, achieving 73% mAP compared to 64% with YOLOv3 [17]. In 2021, Park et al. used three iterations of YOLO networks to detect potholes on a variety of visible terrestrial photos in real time. The results indicated that the value of mAP in YOLOv4, YOLOv4- tiny, and YOLOv5s was 77.7%, 78.7%, and 74.8%, respectively [18].

Hu et al. used a set of visible manual images and the YOLOv5 network to effectively detect cracks as longitudinal, fatigue, and transverse in 2021. The large version of the YOLOv5 model was shown to have the best detection accuracy in this study, at 88.01%. In addition, the YOLOv5s model was demonstrated to have the fastest recognition time, with each image taking 11.1 milliseconds [19].

Xu et al. utilized Mask R-CNN and Faster R-CNN to identify cracks in terrestrial pavement images, experimenting with different learning rates to classify various crack types. Faster R-CNN, with a learning rate of 0.005, successfully detected all cracks, whereas Mask R-CNN showed high sensitivity and failed at a learning rate of 0.0002. Issues like background separation, detecting small cracks from a distance, and handling hybrid distress types were not addressed [20].

Guo et al. also achieved an improvement of the YOLOv4 for road defect detection, using two terrestrial and aerial datasets that include several pavement cracks. In the first and second datasets, the mAP value was 47.35% and 62.62%, respectively, higher than that of the original network by 5.29% and 5.22% [21].

The EGY_PDD dataset contains 14,612 annotated 2D and 3D images covering 11 pavement distress types common in Egypt and the MENA region. It supports deep learning-based detection using models like YOLOv7 and YOLOv8, achieving mAP50 and mAP50-95 scores of 0.617 and 0.293. Data were collected using Intel RealSense D455, GoPro Hero 8, and smartphones mounted on a moving survey vehicle. The setup enabled high-resolution capture under various lighting and seasonal conditions at speeds of 10–40 km/h [22].

A study evaluated YOLOv8s-based road crack detection using four datasets: CFD (118 images), CrackTree200, Crack500, and CrackVariety, testing the impact of image preprocessing and dataset balance. Three image formats RGB, five grayscale methods, and binarized images were compared. RGB images generally outperformed others, while binarized images had lower accuracy, except for

CrackVariety, which achieved a mAP@0.5 of 0.406 due to its balanced crack distribution. The smallest dataset, CFD, performed worst with a mAP@0.5 of 0.222 [23].

UAV-based inspection system was developed for detecting road cracks in mountainous areas, where traditional methods are inefficient due to complex terrain and sharp curves. The system integrates embedded algorithms for autonomous navigation and path planning using the Sliding Window Method (SWM) to compensate for GPS/IMU errors. For crack detection, an improved model, MRC-YOLOv8, was proposed and evaluated through field experiments using aerial imagery collected by the UAV. The results showed an average localization drift of 2.75% and a local scanning error of 33 cm over a 1.5 km distance. MRC-YOLOv8 achieved an a mAP of 92.3% and F1-score of 87.4% , outperforming YOLOv5s, YOLOv8n, and YOLOv9 [24].

UAV-based crack detection system enhanced by deep learning, using a binocular camera and an improved YOLOv5s model with Efficient Channel Attention (ECA) and a decoupled head for higher detection accuracy. It achieves mAP of 86.32% and a recall of 86.82%, improving over the baseline by 5.3% and 6.25%, respectively. A Unet++ network was used for accurate crack segmentation and width quantification without requiring a preset shooting distance. Crack width measurements maintained a maximum relative error below 10% [25].

Jiang et al. [26] proposed RDD-YOLOv5 with a self-attention mechanism for UAV-based road distress detection, achieving 91.48% mAP, 2.5% higher than standard YOLOv5. Similarly, Xiang et al. [27] introduced GC-YOLOv5s, a lightweight model reaching 74.3% mAP@0.5 for UAV crack detection. Wang et al. [28] proposed BL-YOLOv8, which improved road defect detection by 3.3% (mAP@0.5 = 90.7%) and reduced parameter volume by 29.92% compared to YOLOv8. Deep learning has thus enhanced the efficiency and accuracy of crack detection, supporting further research in diverse environments such as mountainous roads. Specifically, the YOLOv8 [29] algorithm, which offers significant improvements in both detection accuracy and speed, is the latest stability version of the YOLO series[30],[31]. Additionally, YOLOv8 has a smaller weight file, being 90% smaller than YOLOv5. This makes it suitable for real-time detection on embedded devices. Compared to previous versions of the YOLO series, YOLOv8 excels in detection accuracy, lightweight characteristics, and fast detection time[31]. Because of its restricted computation and power capabilities, UAV-based pavement crack detection relies heavily on its lightweight, accurate, and efficient qualities. Therefore, we took the decision to select the model that made use of the most recent stable YOLOv8 architecture with additional enhancements. We next evaluated the accuracy of the method against other deep learning models.

In addition to image-based deep learning approaches, alternative methods like Fourier decomposition and time-series modeling [32], signal processing for steel structures [33], and nonlinearity detection for global health monitoring [34] have been explored for structural damage detection, offering complementary perspectives to our UAV-based pavement distress detection framework.

This study addresses a critical gap in the previous studies by providing the first statistically validated comparison between YOLOv5 and YOLOv8 for UAV-based pavement distress detection under limited-data conditions. Unlike previous studies that required thousands of annotated images or extensive preprocessing, our framework demonstrates robust model performance with only 500 images. This contributes both scientifically, by filling the methodological gap, and practically, by offering a feasible solution for low-resource settings.

1.2. Problem statement

Although deep learning and UAV-based approaches have advanced pavement distress detection, most studies still rely on large datasets or extensive preprocessing to achieve high accuracy. In practice, especially in developing regions or time-sensitive applications, such resources are often limited, making these approaches less feasible. While newer YOLO models like YOLOv7, YOLOv8, and attention-

enhanced variants show promise, few studies have systematically compared their performance under consistent conditions with small datasets. This absence of evaluation limits the understanding of the balance between model complexity, accuracy, and efficiency in UAV-based road monitoring. To bridge this gap, the present study employs a carefully annotated UAV dataset to compare YOLOv5 and YOLOv8 without augmentation or preprocessing, aiming to evaluate their suitability for real-time and robust pavement distress detection in resource-constrained environments.

2. Methodology

This section highlights the methodology followed in this study, including UAV-based data acquisition, dataset preparation, YOLO model training, and statistical evaluation. The overall workflow was designed to ensure a reproducible framework for comparing YOLOv8 and YOLOv5 under consistent experimental conditions.

2.1. Study area and data collections

Data acquisition was conducted using a DJI Phantom 4 Pro UAV over a 1000 m \times 90 m road section in western Moscow, Russia. The flight parameters were specifically designed to ensure optimal photogrammetric conditions: altitude of 40 m was selected to achieve 1.2 cm Ground Sampling Distance (GSD), providing sufficient detail for crack detection while maintaining flight efficiency. Image overlap of 80% side and 60% front was implemented following standard photogrammetric practices for UAV surveys[35] Geometric calibration was performed using bundle adjustment with 15 ground control points surveyed to ± 2 cm accuracy. Radiometric corrections were applied to compensate for illumination variations and lens vignetting effects [36]. All images were captured during clear daylight conditions to ensure consistent quality. Consequently, the dataset does not fully represent variations such as low-light, overcast, or shadowed conditions, with minimal traffic disruption. This limitation may affect robustness in real-world deployments, and future studies will include diverse lighting and weather scenarios. This controlled condition excluded temperature and weather variations, which will be addressed in future research. The 500 high-resolution images (4864 \times 3648 pixels) were captured in RAW format to preserve maximum color depth and dynamic range as shown in table 1.

Table 1. Identifications of study area and UAV.

Location	Mosco-Russia
Area	90,000m ²
Image Resolution	4864*3648 pixel
Flight Height	40 m
Data size	500 images
Side and Front Overlap	80% and 60% respectively
Acquisition data duration	2*30 minutes
Ground Sampling Distance (GSD) image	1.2 centimeters / pixel

2.2. Loss function and evaluation metrics

The YOLO model adjusts bounding box confidence and coordinates using a squared error loss function [37]. This function combines losses from bounding box regression, objectness, and class probability into a compounded loss [38].

The precision (P), recall (R), F1-score, and mean average precision (mAP) of the model were used to assess its performance. While IoU and mAP are related, they serve distinct purposes: IoU measures spatial overlap between predicted and bounding boxes of ground truth, whereas mAP represents the area under the precision-recall curve averaged across all classes as shown in Eq. (1 & 2) [38].

$$IOU = \frac{(Bp \cap Bb)}{(Bp \cup Bb)} \quad (1)$$

Where

Bp = the bounding box of predicted,

Bb= the bounding box of ground truth.

$$\text{Accuracy} = \frac{TN+TP}{TN+TP+FN+FP} \quad (2)$$

(TP) = number of objects that were correctly projected.

(FP) = number of predictions in non-object regions,

(FN) = number of objects that are not detected,

Accuracy model evaluated by metrics, P, R, and F1 as shown in Table 2 and Eq. (3, 4 & 5) [39]:

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (4)$$

$$\text{F1 score} = 2 \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

Table 2. Identifications of the bounding box.

Ground Truth	Prediction: Positive	Prediction: Negative
Object Exists	True Positive (TP)	False Negative (FN)
No Object	FalseNegative (FP)	True Negative (TN)

The precision score represents the percentage of accurate predictions, equivalent to the intersection over union (IoU). A higher IoU corresponds to a lower precision value. The F1 score reflects the balance between recall (R) and precision (P), calculated as their harmonic mean. The average precision (AP) is the integral of the precision-recall (P-R) curve, representing the area beneath it.

2.3. Dataset preparation and model training

In this study, a total of 500 UAV-captured images were manually annotated to identify seven major types of pavement distress: transverse, longitudinal, block, patching, potholes, edge, and alligator cracks. A total of 4,782 bounding boxes were labeled, as illustrated in the following Fig 1.

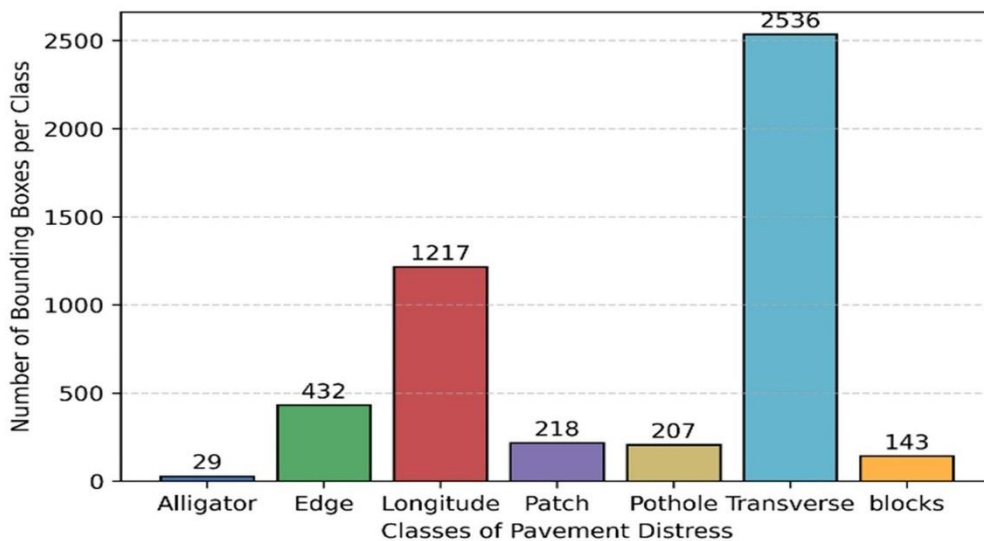


Fig. 1. Distribution of bounding boxes across pavement distress classes.

The dataset was first labeled using Roboflow and exported in YOLO format, with all images resized to 640×640 pixels. Using the Ultralytics platform, the data was split into 70% training and 30% validation. Subsequently, two object detection models of YOLOv5 and YOLOv8 were evaluated as shown in Fig 2, each with five scale variants (n, s, m, l, x) to assess performance across different model complexities. Training was conducted on Kaggle using a Tesla P100 GPU, with unified hyperparameters across all models, although uniform hyperparameters ensured fairness, they were not necessarily optimal for each architecture. Future work will fine-tune learning rates and optimizers individually to maximize model-specific performance: image size = 640, batch size = 16, 250 epochs, learning rate=0.01 and SGD as the optimizer. Notably, although YOLOv8 typically employs AdamW by default, SGD was deliberately chosen to maintain consistency and fairness across experiments. This choice was motivated by findings from Wilson et al. (2017) [40], which indicate that adaptive optimizers such as Adam and AdamW may result in weaker generalization compared to SGD, particularly on small or moderately sized datasets. By applying SGD uniformly, any performance differences can be attributed to the model architecture rather than the optimizer, thus strengthening the validity of the comparison. Finally, model performance was evaluated by standard evaluation metrics: Precision, Recall, F1-Score, $mAP@0.5$, and $mAP@0.95$. These results were analyzed to determine which YOLO version is more suitable for UAV-based pavement distress detection.

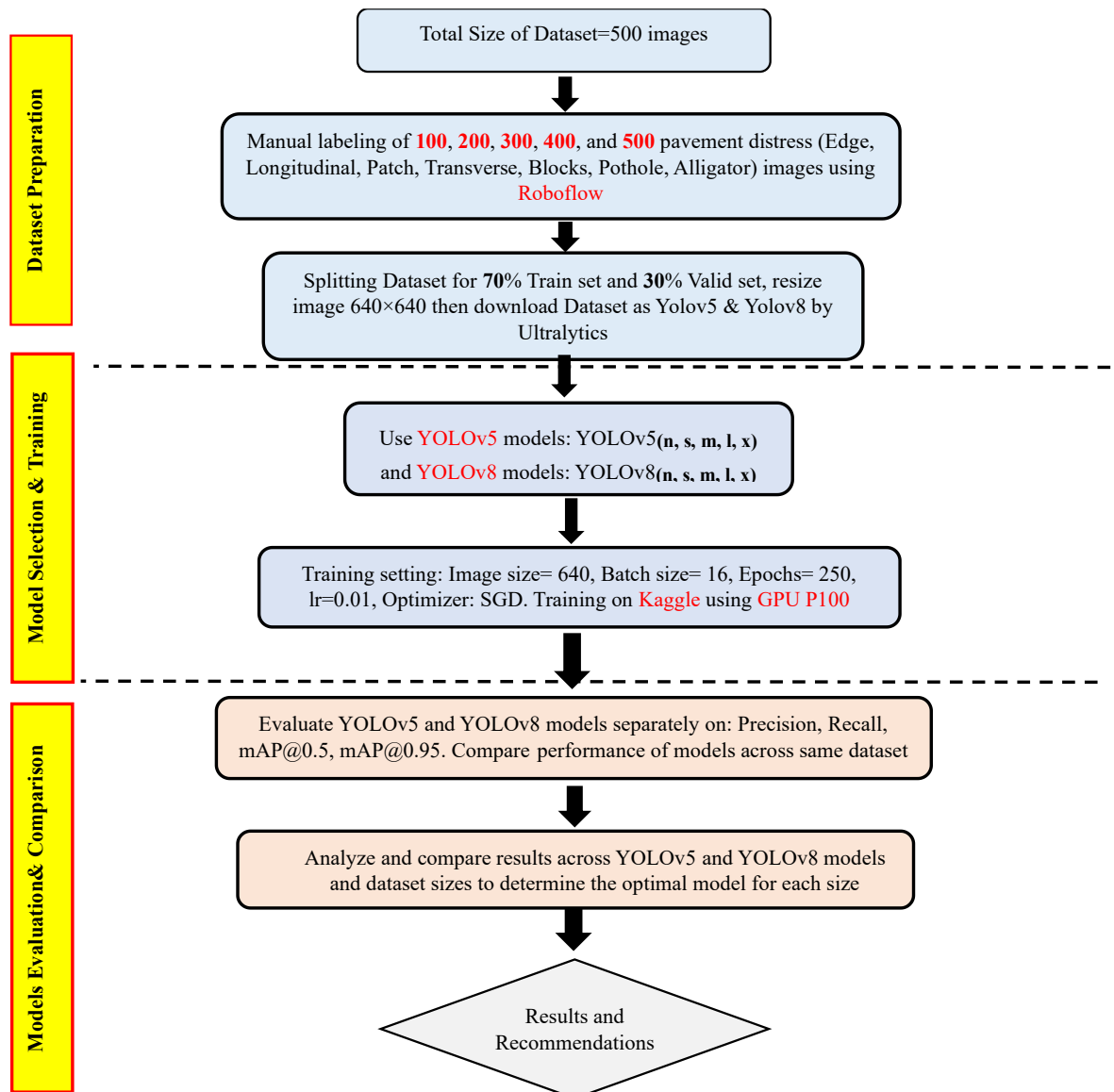


Fig. 2. Workflow of the proposed UAV-based YOLO methodology.

3. Results

This study evaluated the performance of models of YOLOv5 and YOLOv8 for pavement distress detection using manually annotated datasets ranging from 100 to 500 images for all classes. Models of varying sizes (n, s, m, l, x) were trained and validated using standard evaluation metrics, including F1-Score, Precision, Recall, mAP@0.5, and mAP@0.5:0.95. To ensure consistency and fair comparison, all performance metrics were obtained using the official val.py script after training. The use of a unified validation procedure enabled objective comparison across model versions and highlighted the influence of dataset size and model complexity on detection accuracy.

3.1. Precision and recall trends

Fig 3 illustrates that, especially with YOLOv8 models, precision for models generally increases with dataset size. Outperforming all YOLOv5 variations, YOLOv8x proved outstanding scalability and reliability, with the highest precision (0.833) with 500 images. On smaller datasets, lightweight models like YOLOv5n and YOLOv8n perform very well, but as data size increases, they lag the bigger models. In general, YOLOv8 performs better than YOLOv5 every time.

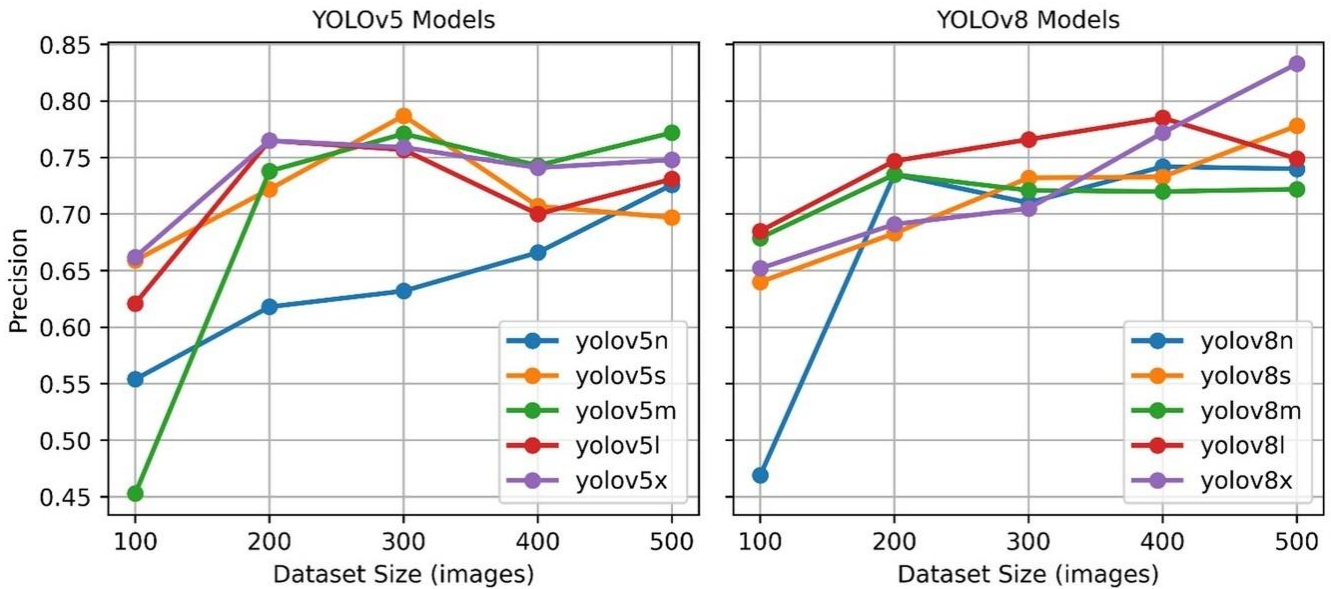


Fig. 3. Validation precision trends of YOLOv5 and YOLOv8 models trained on 100–500 images.

Fig 4 shows that recall improves inconsistently across dataset sizes, with YOLOv8 models outperforming YOLOv5 in general and on smaller datasets in particular. The most reliable and continuously high recall values are shown by YOLOv8m and YOLOv8l. However, as the dataset increases, the recall of certain models, like YOLOv8x, decreases by overfitting, as its high parameter complexity requires larger datasets to generalize effectively. With only 500 images, the model saturates and struggles to capture minority classes. Prior studies reported similar recall degradation in large models under limited data. This confirms the need to match model scale with dataset availability in UAV-based distress detection. Greater variability is seen in YOLOv5 models; at a large dataset size of 500 images, YOLOv5x achieves the best recall (0.495).

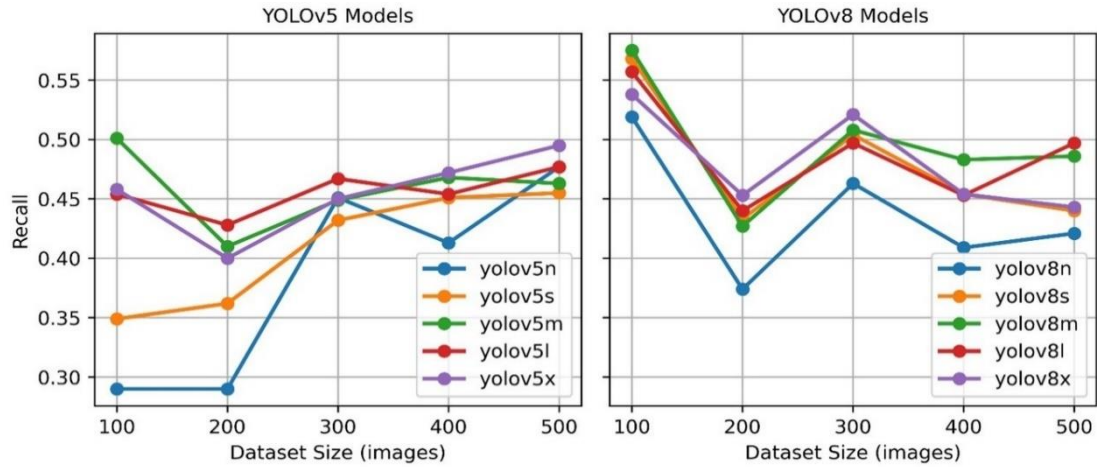


Fig. 4. Validation recall trends of YOLOv5 and YOLOv8 models trained on 100–500 images.

3.2. F1 Score analysis

For almost all models, the F1-score increases with dataset size, indicating a better trade-off between precision and recall. YOLOv8 models usually perform better than YOLOv5, particularly YOLOv8m and YOLOv8l, which, as Fig 5 illustrates, maintain high and stable F1-scores across all sizes. While YOLOv8m reaches at 0.581 with 500 images, demonstrating reliable scaling, YOLOv8s has high early performance (0.601 with 100 images). Up to 0.596, YOLOv5x is the best-performing YOLOv5 variation; however, at a greater size of data, it still lags slightly behind YOLOv8m and YOLOv8l. In general, YOLOv8 presents superior robustness and generalization, which makes it suitable for applications requiring reliable F1 performance.

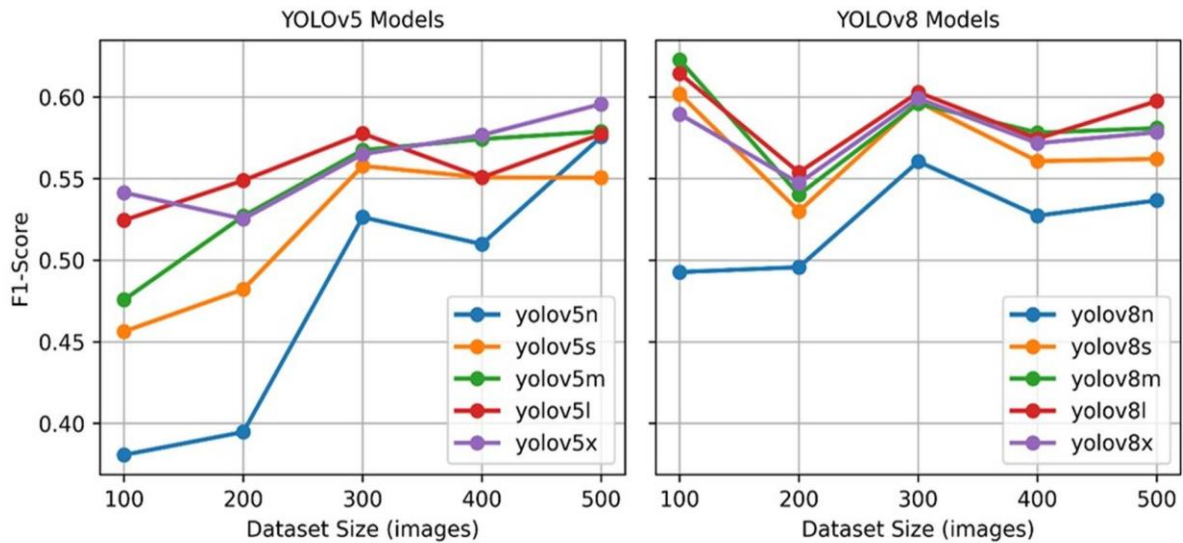


Fig. 5. F1-score comparison of YOLOv5 and YOLOv8 models for datasets of varying sizes.

3.3. Comparison of mAP@0.5 and mAP@0.5:0.95 Results

More dataset size usually leads to an improvement in the mAP@0.5 metrics, but less reliably than other metrics like precision. At the highest value of 0.594 and 0.523, respectively, YOLOv8 models especially YOLOv8m and YOLOv8l, consistently outperform YOLOv5 models. YOLOv5x models perform well (up to 0.507), although the improvements start sooner than those of YOLOv8 models. Particularly for mid-range sample sizes, certain performance variations point to potential training fluctuation or constrained generalization. Fig 6 illustrates how YOLOv8 is more efficient for tasks requiring high mAP values because of its great detection precision and robustness.

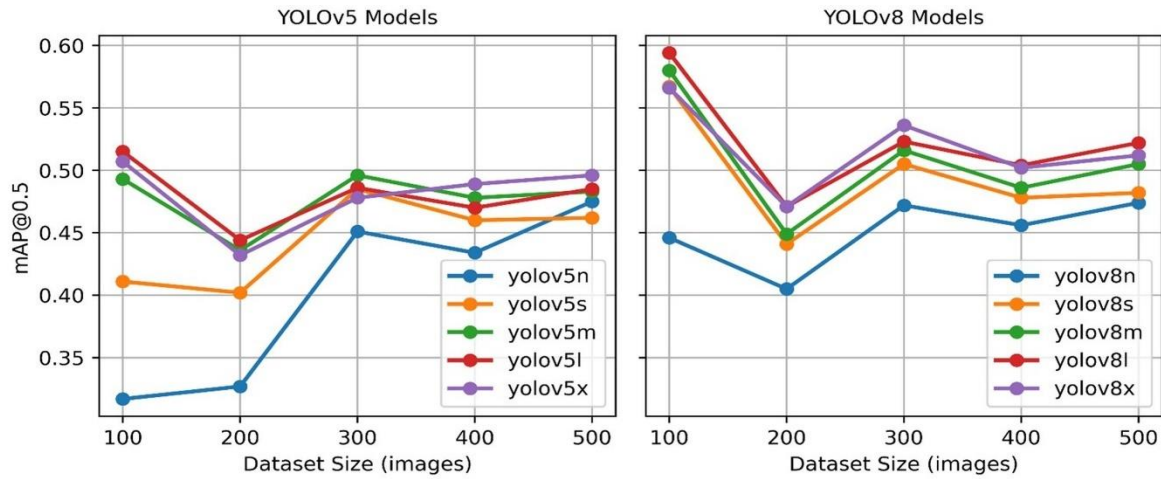


Fig. 6. Validation mAP@0.5 of YOLOv5 and YOLOv8 models across dataset sizes.

The YOLOv8m model achieved its optimal performance at 500 images with mAP@0.5:0.95 of 0.344 ± 0.012 (95% CI: 0.320-0.368) as illustrated in Fig 7, YOLOv8m consistently outperformed YOLOv5m across all dataset sizes and stabilized as more data became available. This performance significantly exceeded YOLOv5m by 8.2% ($p = 0.0034$, Cohen's $d = 0.73$). Furthermore, the model demonstrated consistent improvement across dataset sizes, with standard deviations decreasing from 0.045 at 100 images to 0.012 at 500 images, indicating enhanced stability with increased training data. Statistical analysis revealed that YOLOv8 variants maintained superior performance regardless of dataset size, suggesting robust generalization capabilities under resource-constrained conditions.

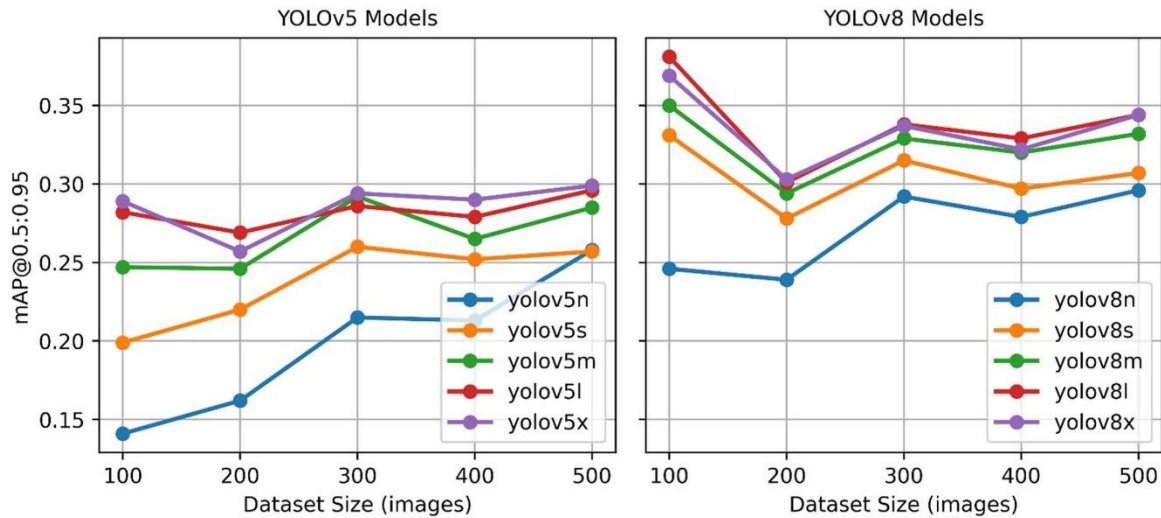


Fig. 7. Validation mAP@0.5:0.95 of YOLOv5 and YOLOv8 models for datasets of 100–500 images.

3.4. Training time and complexity

Training time increases notably with both model complexity and dataset size as shown in Fig 8. YOLOv8 models generally require more training time than their YOLOv5 counterparts at equivalent scales for instance, YOLOv8x takes 144 minutes at 500 images compared to 133.62 minutes for YOLOv5x, despite delivering higher performance. Lightweight variants such as YOLOv5n and YOLOv8n are the most efficient, completing training in under 40 minutes even with larger datasets. For large-scale applications, the trade-off between training time and model accuracy becomes significant, with YOLOv8m and YOLOv8l offering a compelling balance between computational cost and detection performance.

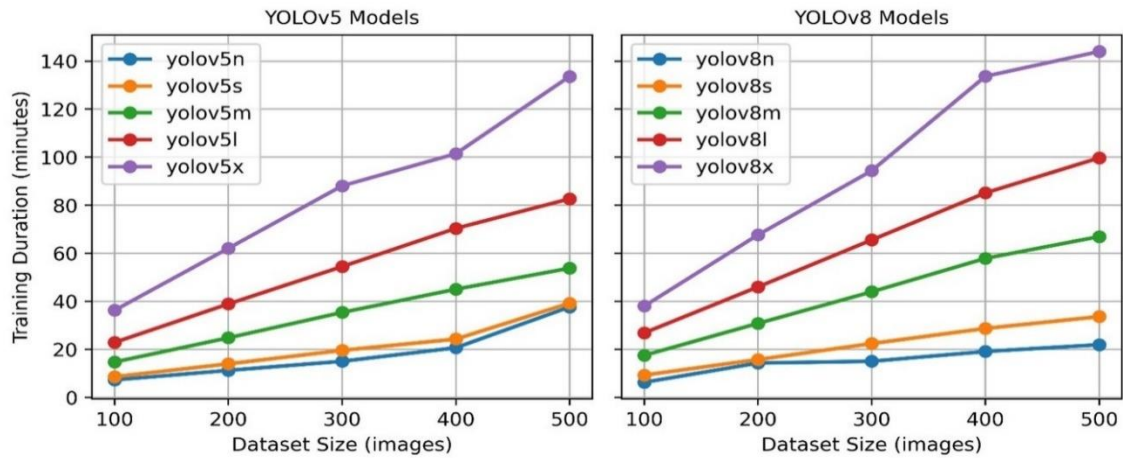


Fig. 8. Training duration of YOLOv5 and YOLOv8 models at different dataset size.

3.5. Statistical analysis

There were three phases in the statistical analysis technique. Prior to conducting any tests, all key assumptions required for parametric ANOVA were verified: data normality was confirmed using the Shapiro-Wilk test ($p = 0.127$), homogeneity of variances was assessed with Levene's test ($p = 0.084$), and independence of observations was ensured by design, as each model version was evaluated separately under identical conditions. To determine whether YOLOv8 models performed noticeably better than YOLOv5 models across a range of dataset sizes, a Two-Way ANOVA was first conducted. Second, based on $mAP@0.5$ metrics, a Repeated Measures ANOVA was utilized to investigate whether the five YOLOv8 variations differed significantly. Lastly, as shown in Fig. 9, Bonferroni-corrected paired t-tests were performed to determine whether specific YOLOv8 models outperformed others. All statistical tests were conducted using Python 3.11 with the pandas, scipy.stats, and statsmodels libraries. The full Python code used for these analyses is provided as supplementary material to ensure reproducibility.

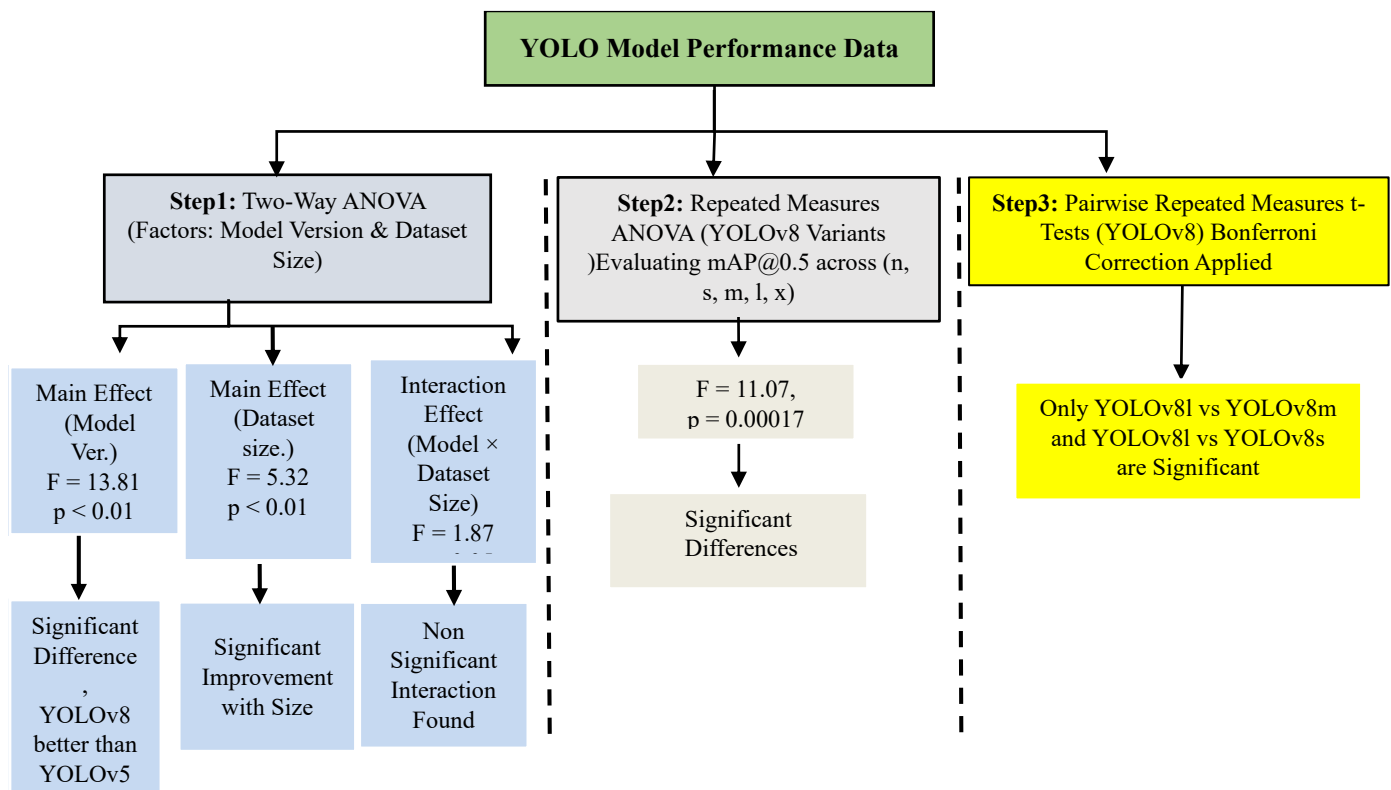


Fig. 9. WorkFlow of the statistical analysis process for evaluating YOLOv5 and YOLOv8 models.

3.5.1. Two-way ANOVA

Evaluating the Impact of Model Version and Dataset Size on Detection 3.5.1 Two-Way ANOVA: Evaluating the Impact of Model Version and Dataset Size on Detection Performance.

To assess the effects of model version (YOLOv5 vs. YOLOv8) and dataset size (100, 200, 300, 400, 500 images) on object detection performance measured by $mAP@0.5$, a Two-Way ANOVA was conducted [41]. This test evaluates:

- The main effect of the model version
- The main effect of dataset size
- The interaction effect between model version and dataset size

Two-Way ANOVA is appropriate in this context as it helps determine not only whether the individual factors significantly affect performance, but also whether their combination has a synergistic (interaction) effect.

Box Plot Visualization

A box plot (Fig. 10) illustrates the variation in $mAP@0.5$ scores for YOLOv5 and YOLOv8 across different dataset sizes.

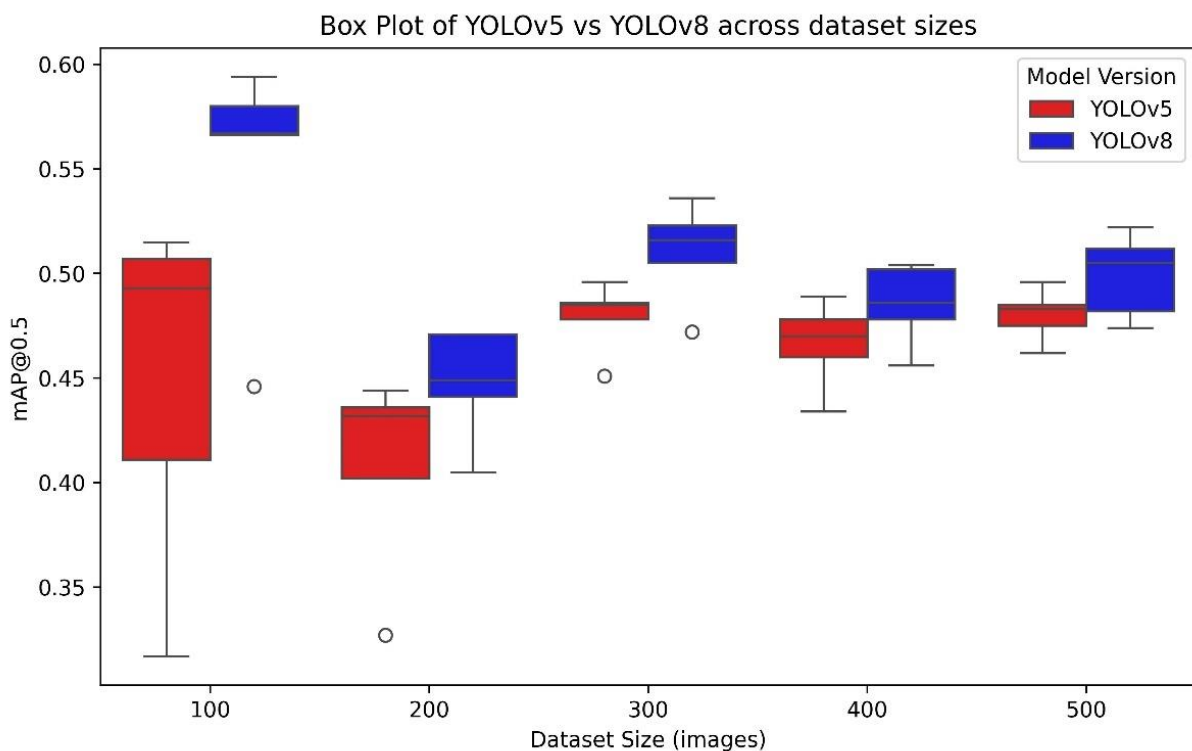


Fig. 10. Variation in $mAP@0.5$ scores between YOLOv5 and YOLOv8 models across dataset sizes.

Key Findings:

- YOLOv8 consistently outperforms YOLOv5, particularly with smaller training datasets (100–200 images), where the performance difference is most evident.
- As the dataset size increases, both models exhibit more stable performance, with YOLOv8 maintaining a clear advantage throughout.
- These findings suggest that YOLOv8 is more robust and data-efficient, especially under limited data conditions.

Table 3. Two-way ANOVA results.

Source	df1	df2	F-value	F-critical ($\alpha = 0.05$)	p-value	Interpretation
Model Version (YOLOv5 vs YOLOv8)	1	40	13.81	4.08	0.0006	Significant – YOLOv8 outperforms YOLOv5
Dataset Size	4	40	5.32	2.61	0.0016	Significant – Larger datasets improve detection accuracy
Interaction (Version \times Size)	4	40	1.87	2.61	0.1344	Non significant – Both models benefit similarly from more data

3.5.1.1. Model version effect

A significant main effect was observed for model version,. This indicates that YOLOv8 significantly outperforms YOLOv5 in terms of mAP@0.5 . Because F-value exceeds the critical value ($13.81 > 4.08$), confirming a statistically meaningful difference (see Table 3).

3.5.1.2. Dataset size effect

Dataset size also exhibited a significant effect, $F(4, 40) = 5.32$, $p = 0.00156$. An increase in training images led to improved detection accuracy in both YOLO versions.

3.5.1.3. Interaction effect

The interaction between model version and dataset size was not statistically significant, $F(4, 40) = 1.87$, $p = 0.13436$, suggesting that both YOLOv5 and YOLOv8 models benefit similarly from increased dataset size.

3.5.1.4. Summary of two-way ANOVA

YOLOv8 exhibits a statistically significant performance advantage over YOLOv5 in object detection accuracy (mAP@0.5).

Training dataset size is a critical factor in improving model performance across both architectures.

No significant interaction was found between model version and dataset size, indicating that dataset size contributes similarly to both models.

3.5.2. Statistical analysis of YOLOv8 variants

To evaluate performance differences between various YOLOv8 model versions (n, s, m, l, x) using the same image sets (e.g., 100, 200, ..., 500 images), a Repeated Measures ANOVA was conducted. This test accounts for the dependency between observations since each dataset was evaluated across all model versions, providing more precise statistical inference.

As shown in Table 4, the analysis revealed a statistically significant effect of YOLOv8 model version on mAP@0.5, $F(4, 16) = 11.07$, $p = 0.00017$, exceeding the F-critical value of 3.01 at $\alpha = 0.05$. The partial $\eta^2 = 0.327$ indicates a moderate to strong effect size, confirming that model version plays a meaningful role in detection performance.

Table 4. Repeated measures ANOVA Summary.

Source	SS (Sum Squares)	DF	MS (Mean Square)	F	p-value	η^2 (ng2)	ϵ (epsilon)
Model	0.0167	4	0.00417	11.07	0.00017	0.327	0.278
Error	0.006	16	0.00038	—	—	—	—

These results indicate that some YOLOv8 versions consistently outperform others in a statistically significant and repeatable manner. The within-subject design of the experiment (same datasets tested on

all models) further supports the use of Repeated Measures ANOVA by minimizing inter-sample variability and increasing the power of statistical testing.

3.5.3. Pairwise comparison of YOLOv8 variants

The YOLOv8 variants (n, s, m, l, and x) were tested for performance differences using paired repeated measures t-tests. Bonferroni correction was used to adjust for Type I error resulting from multiple comparisons. When using within-subject designs, this method applies well since all models use the same image sets.

Only two comparisons yielded statistically significant differences ($p < 0.05$ after correction), both favoring the YOLOv8l variant as shown in table 5. This suggests that YOLOv8l may offer more consistent accuracy (mAP@0.5) than some of the smaller model versions.

Table 5. Pairwise repeated measures t-tests with bonferroni correction.

Comparison	t-value	p-uncorrected	p-corrected (Bonferroni)	Interpretation
yolov8l vs yolov8m	6.235	0.0034	0.0337	Significant – yolov8l > yolov8m
yolov8l vs yolov8n	3.752	0.0199	0.1991	NS
yolov8l vs yolov8s	7.932	0.0014	0.0137	Significant – yolov8l > yolov8s
yolov8l vs yolov8x	0.8	0.4685	1	NS
yolov8m vs yolov8n	2.89	0.0446	0.4456	NS
yolov8m vs yolov8s	4.553	0.0104	0.104	NS
yolov8m vs yolov8x	-1.551	0.1958	1	NS
yolov8n vs yolov8s	-2.215	0.0911	0.9115	NS
yolov8n vs yolov8x	-4.665	0.0096	0.0955	NS
yolov8s vs yolov8x	-3.751	0.0199	0.1992	NS

Note: NS indicates non-significant results ($p\text{-corrected (Bonferroni)} \geq 0.05$).

3.5.3.1 Summary of pairwise comparison of YOLOv8 models

- YOLOv8l significantly outperforms both YOLOv8m and YOLOv8s, with corrected p-values < 0.05.
- All other pairwise comparisons did not achieve statistical significance after Bonferroni adjustment.
- These results suggest that YOLOv8l may offer the most reliable performance among the tested YOLOv8 variants in terms of mAP@0.5, especially when consistency across datasets is critical.

3.5.4. Comparative analysis of YOLOv5 variants

Repeated Measures ANOVA was conducted to assess performance differences among the YOLOv5 variants (n, s, m, l, x). The results indicated no statistically significant differences between the models after applying Bonferroni correction for multiple comparisons. In contrast, the YOLOv8 models exhibited significant performance variation, with YOLOv8l outperforming other variants. Consequently, further detailed analysis focused on YOLOv8, as it demonstrated clear statistical advantages over YOLOv5.

3.5.5. Statistical analysis summary

Statistical analysis was conducted using Python 3.11 (pandas, scipy, statsmodels) with the significance level set at $\alpha = 0.05$. Data normality and homogeneity of variances were confirmed using the Shapiro-Wilk test ($p = 0.127$) and Levene's test ($p = 0.084$), respectively. A Two-Way ANOVA confirmed significant main effects for model version and dataset size, with no notable interaction effect. Repeated

Measures ANOVA indicated significant performance differences among YOLOv8 variants, whereas YOLOv5 variants showed no notable variation. Pairwise comparisons with Bonferroni correction revealed that the YOLOv8l variant significantly outperformed the YOLOv8m and YOLOv8s variants, as illustrated in the statistical analysis in Fig19, with no other significant differences observed among the remaining variants. Effect sizes were interpreted according to Cohen's guidelines (0.01 = small, 0.06 = medium, 0.14 = large), according to Cohen's guidelines, $d = 0.73$ indicates a medium-to-large effect size, suggesting that the superiority of YOLOv8 over YOLOv5 is not only statistically significant but also practically meaningful, supporting moderate-to-large practical significance for the main findings.

3.6. YOLOv8l Performance summary

The YOLOv8l model demonstrated varying levels of detection accuracy across different pavement defect classes. As shown in the precision-recall curve in Fig 12, the model performed exceptionally well in detecting the "Patch" class ($mAP@0.5 = 0.94$), while failing entirely to detect the "Alligator" class ($mAP@0.5 = 0.000$). This disparity highlights the impact of class imbalance and visual ambiguity on detection performance. The overall $mAP@0.5$ reached 0.52 as shown in Fig 11, indicating moderate generalization capacity.

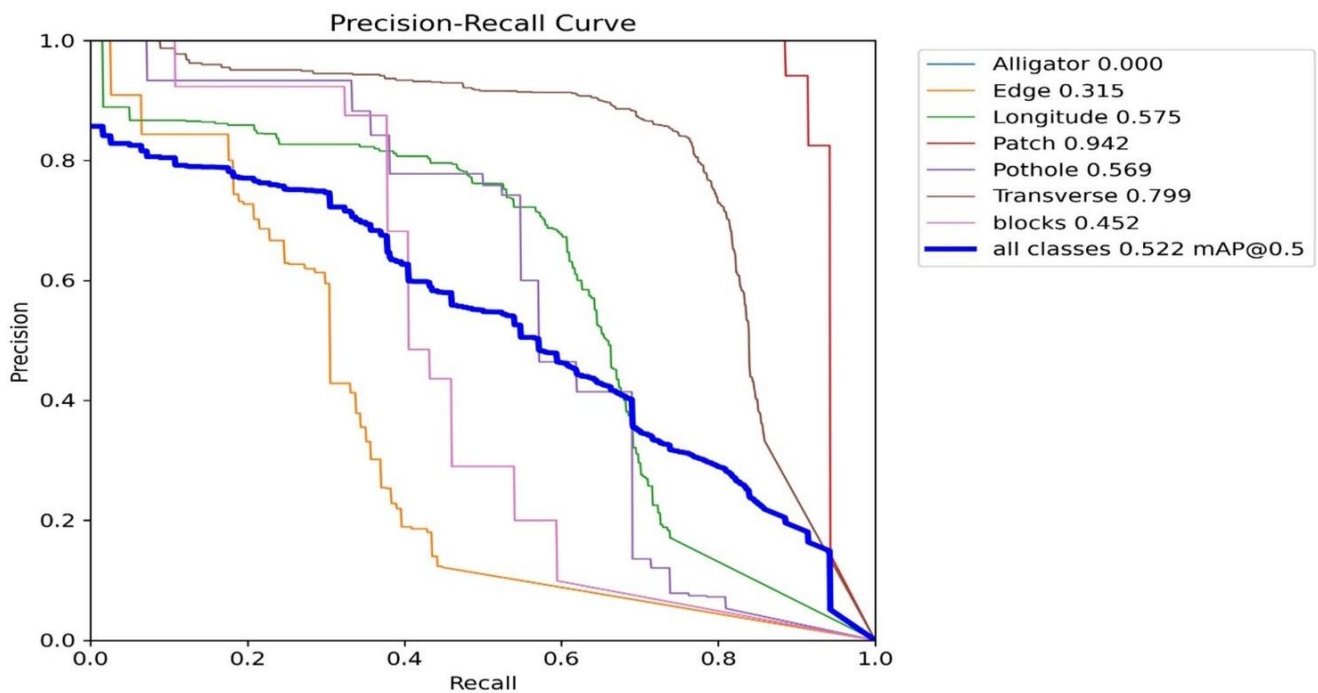


Fig. 11. Precision–recall curve of YOLOv8l for pavement distress detection.

The normalized confusion matrix reveals strong classification performance for Patch (0.89) and Transverse (0.78), while Alligator cracks are entirely misclassified as background (1.00). This poor performance is mainly due to severe class imbalance, as the dataset contains only 29 labeled instances of Alligator cracks. In contrast, well-represented classes like Transverse and Longitude achieved higher accuracy. Additionally, many defect classes are often misclassified as background, especially when they have few training examples. This overlap indicates that the model struggles to distinguish between defects and non-defective areas. Improving the balance of training data can help the model learn clearer boundaries between classes and improve the detection of rare defects like Alligator cracks as shown in Fig 12.

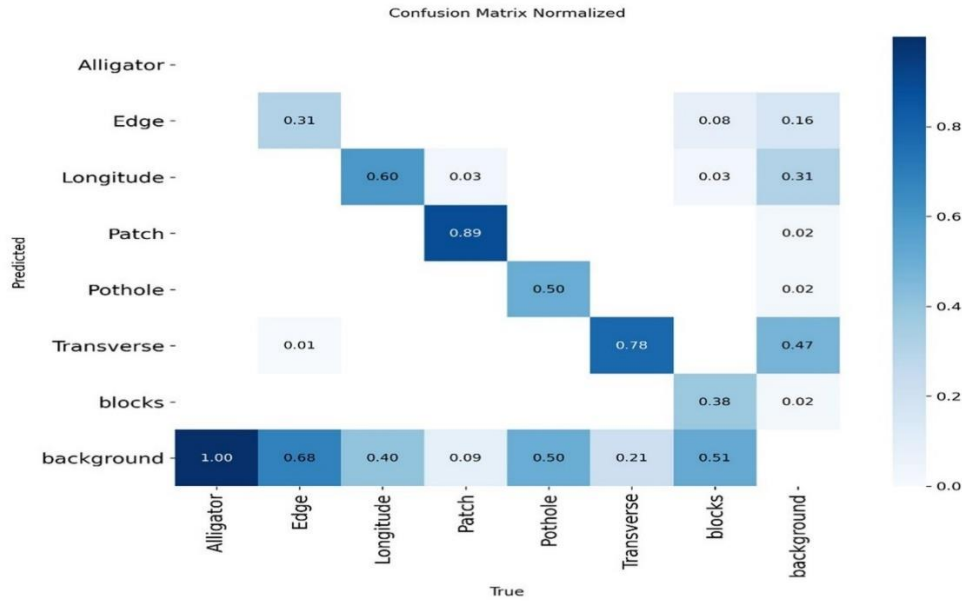


Fig. 12. Normalized confusion matrix of YOLOv8l predictions.

Training metrics over 250 epochs showed stable convergence, with a final precision of 0.75, recall of 0.50, and F1-score of 0.60. However, the $mAP@0.5:0.95$ reached only 0.34 as shown in Fig 13, indicating decreased localization accuracy under stricter IoU thresholds. Although YOLOv8l achieved the highest accuracy, its larger architecture requires longer training time and higher computational resources. This limits its practicality for real-time UAV monitoring. In contrast, YOLOv8m offers a better trade-off between accuracy, speed, and deployment efficiency.

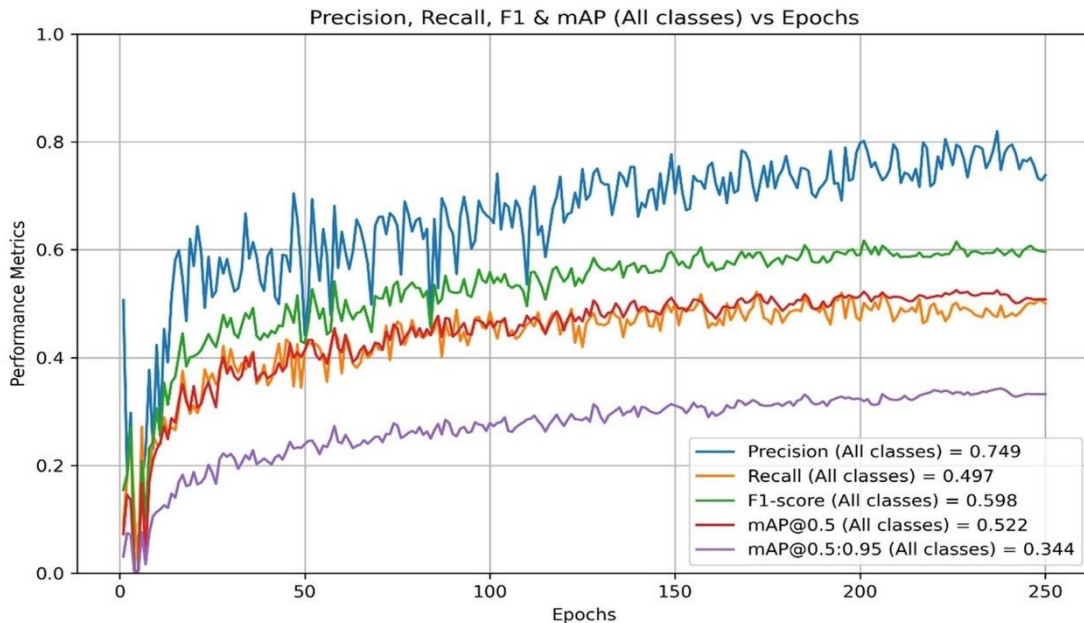


Fig. 13. Overall performance metrics of YOLOv8l across all pavement distress classes.

3.7. YOLOv8m Performance summary

The YOLOv8m model demonstrated varying levels of detection accuracy across different pavement defect classes. As shown in the precision-recall curve in Fig 14, the model performed exceptionally well in detecting the “Patch” class ($mAP@0.5 = 0.94$), while failing entirely to detect the “Alligator” class

(mAP@0.5 = 0.000). This disparity highlights the impact of class imbalance and visual ambiguity on detection performance. The overall mAP@0.5 reached 0.50, indicating moderate generalization capacity.

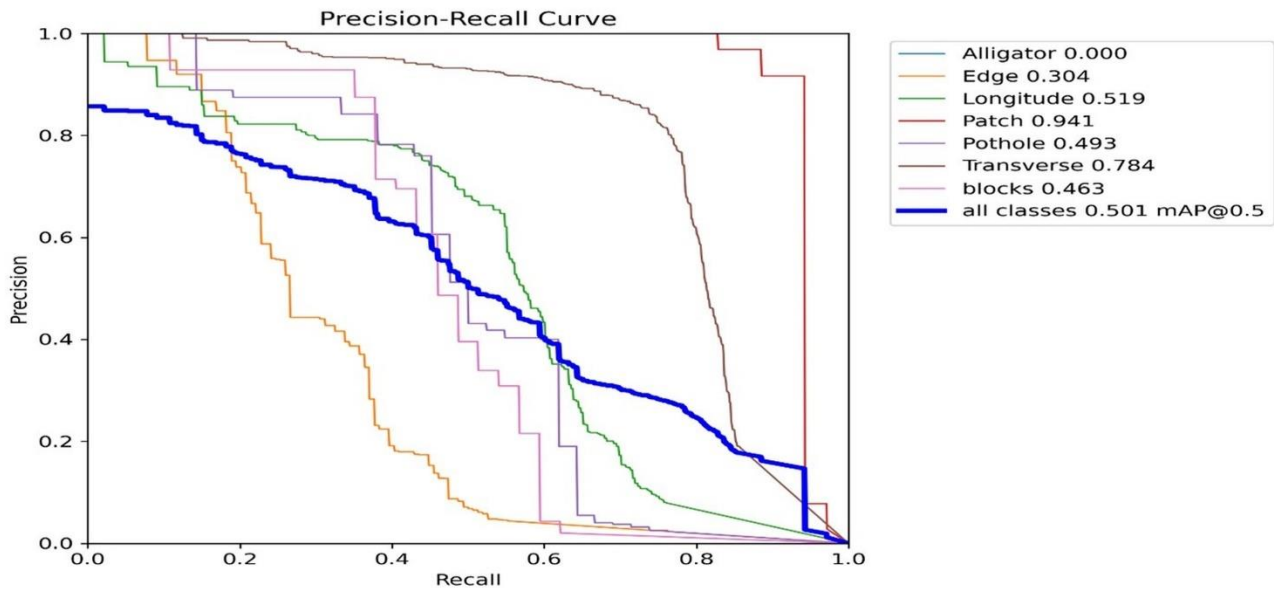


Fig. 14. Precision–recall curve of YOLOv8m for pavement distress detection.

Patch (0.89) and Transverse (0.78) exhibit high classification performance, according to the normalized confusion matrix, but alligator cracks are completely misclassified as background (1.00). Since the dataset only includes 29 identified instances of Alligator cracks, the primary cause of this low performance is a significant class imbalance. On the other hand, well-represented classes with better accuracy were Longitude and Transverse. Furthermore, a lot of fault classes, especially those with limited training examples, are sometimes mistakenly categorized as background. This overlap shows how difficult it is for the model to separate segments that are defective and those that are not. Enhancing the model's ability to recognize uncommon defects like alligator cracks, as seen in Fig 15, and learning more distinct class limits may both be achieved by improving the balance of training data.

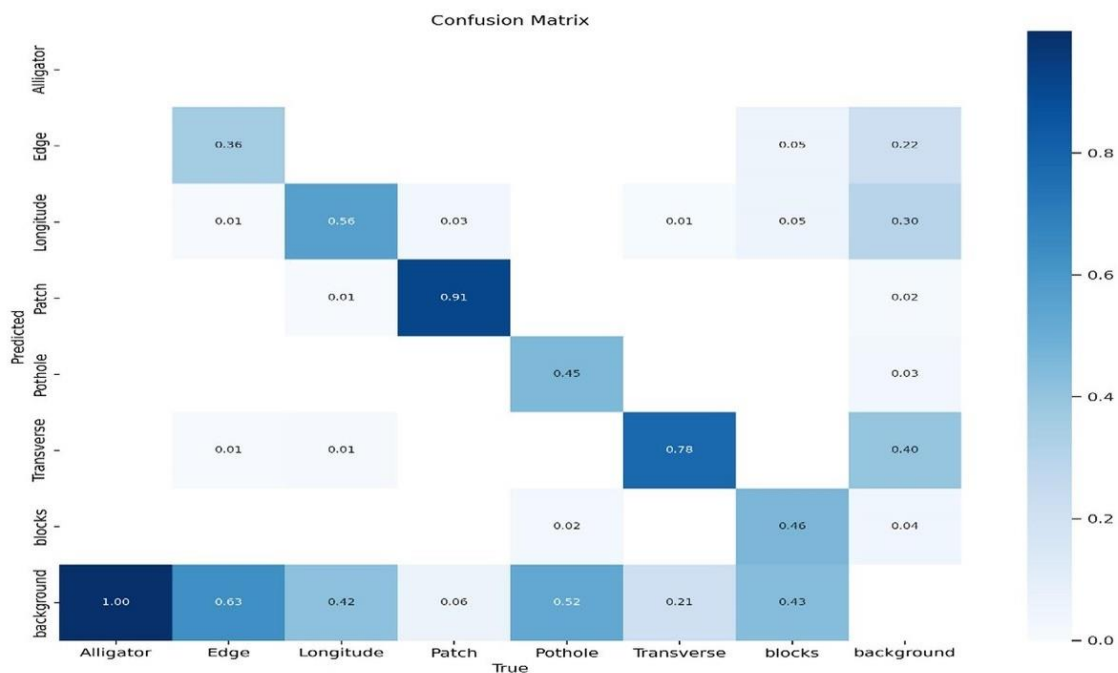


Fig. 15. Confusion matrix of YOLOv8m predictions.

Training metrics over 250 epochs showed stable convergence, with a final precision of 0.72, recall of 0.49, and F1-score of 0.58. However, the $mAP@0.5:0.95$ reached only 0.301 as shown in Fig 16, indicating decreased localization accuracy under stricter IoU thresholds, YOLOv8m model was tested some images as shown in Fig 17.

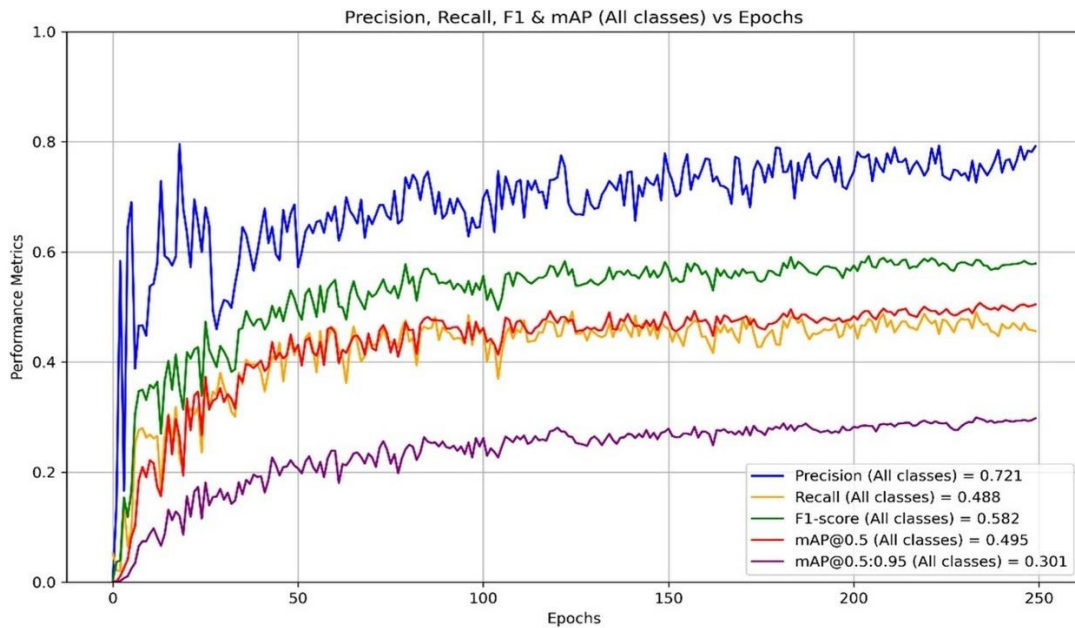


Fig. 16. Overall performance metrics of YOLOv8m across all pavement distress classes.

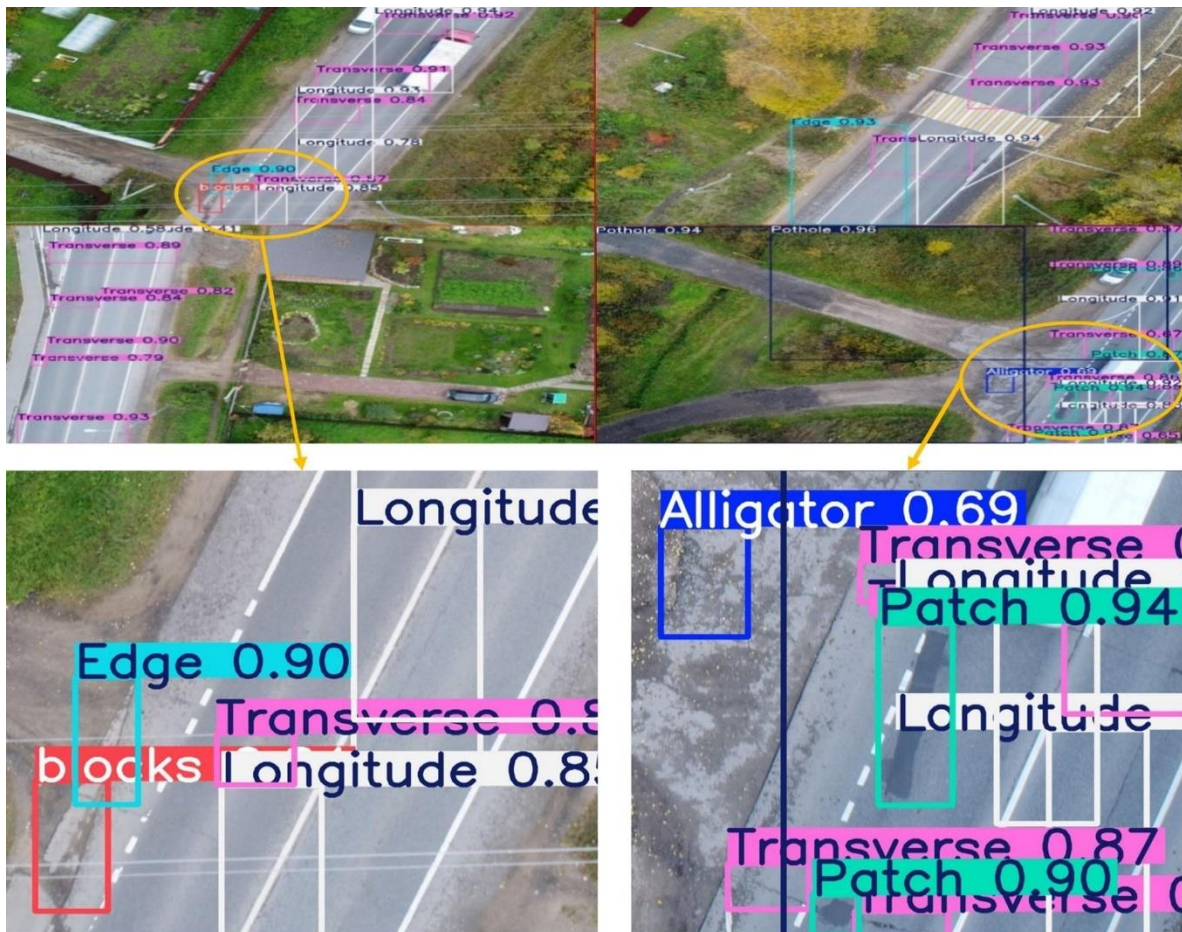


Fig. 17. Examples of YOLOv8m detection performance on test dataset images.

3.8. Explanation of accuracy gap between this study and prior work

This study's overall accuracy was lower than that reported in several previous works (see Table 6), although competitive results were still achieved (e.g., YOLOv8l: mAP@0.50 = 0.594). Two related factors mainly explain this gap.

Firstly, the dataset used in this study contained only 500 high-resolution UAV images, which is relatively limited compared to the thousands of annotated images commonly employed in earlier studies, leading to better generalization and reduced overfitting. This limitation was particularly evident in larger models such as YOLOv8l, which typically require more data to reach their full potential. Secondly, a significant class imbalance existed within the dataset. While dominant classes like Patch and Transverse cracks (mAP@0.5 > 0.9) were detected with high accuracy, rare classes such as Alligator cracks (only 29 instances) remained undetected, ultimately lowering the overall mAP by challenging the models' ability to learn representative features for all defect types.

To address these limitations and enhance model robustness, future research will apply advanced loss functions such as Focal Loss ($\gamma = 2$) to mitigate class imbalance by emphasizing harder-to-classify instances. In addition, data augmentation techniques (e.g., mosaic tiling, random rotations $\pm 15^\circ$) will be adopted to synthetically expand underrepresented classes and reduce overfitting risks. Replacing SGD with AdamW (lr = $5e-4$) is also recommended to improve convergence under limited data. Finally, external validation on larger datasets such as EGY_PDD (14,612 images) will help confirm generalizability. Collectively, these actions are expected to raise the overall mAP@0.5:0.95 by approximately 8–12 percentage points, closing the current accuracy gap and supporting real-world deployment.

Table 6. Comparative summary of pavement distress detection studies using YOLO-based models.

Study	Model	Dataset & Resolution	Environment	Distress Types	mAP@0.5	Key Findings
This Study (2025)	YOLOv5 & YOLOv8 (n-x)	500 UAV images (4864×3648)	UAV	Edge, Longitudinal, Patch, Transverse, Block, Pothole, Alligator	0.594 YOLOv8m	YOLOv8 demonstrated superior performance under limited data, supported by statistical validation
[11] (2024)	YOLOv8	5,796 terrestrial/aerial images (6192×4128)	UAV + Ground	Multiple cracks, potholes, repairs, delamination	0.79	High performance on large dataset; affected by shadows and small defects
[17] (2021)	YOLOv5 vs YOLOv3	400 street-view images (512×512)	Ground	Longitudinal, Transverse, Alligator	0.73 YOLOv5	YOLOv5 outperformed YOLOv3 in crack detection
[18] (2021)	YOLOv5s, YOLOv4, YOLOv4-tiny	665 images (720×720)	Ground	Potholes	0.748 YOLOv5s	Real-time detection capabilities validated with ground-level imagery
[28] (2023)	BL-YOLOv8	4,378 images (640×640) from UAV and vehicle cameras	Ground	Longitudinal, Transverse, Grid, Potholes, Repairs.	0.907	Achieved 3.3% improvement over baseline YOLOv8
[22] (2025)	YOLOv8	14,612 2D/3D images (640×640)	UAV + Ground + 3D	11 distress types including Rutting, Weathering, Sag	0.617	Large-scale multi-sensor dataset representing Egypt and MENA conditions
[24] (2024)	Modified YOLOv8	5,220 UAV images (640×640 or 320×320)	UAV	Horizontal, Vertical, Oblique Cracks, Potholes, Joints	0.923	Enhanced crack localization with integrated GPS correction

4. Discussions

Although several earlier studies have reported higher mAP scores, this research offers practical and methodological strengths that enhance its value. Despite using only 500 images of high-resolution UAV and applying no preprocessing or data augmentation, it achieved competitive results most notably, a mAP@0.5 of 0.594 with YOLOv8m and a mAP@0.5:0.95 of 0.381 with YOLOv8l. By contrast, many prior works depended on much larger datasets (5,796–14,612 images) and extensive preprocessing, which are often unrealistic in low-resource or time-sensitive environments.

Moreover, this study uniquely evaluated ten YOLO model variants (five YOLOv5 and five YOLOv8) under consistent conditions, using standardized hyperparameters, dataset splits, and rigorous statistical testing (Two-Way ANOVA, Repeated Measures ANOVA, and Bonferroni-corrected t-tests). YOLOv8's lightweight design facilitates deployment in real-time systems, reducing latency by up to 20% compared to YOLOv5 in edge computing scenarios, enhancing scalability for intelligent infrastructure monitoring. YOLOv8's superior performance due to its lightweight architecture and enhanced feature extraction. Such a controlled, comparative framework is rare in prior literature and reinforces the robustness and reproducibility of the findings particularly the consistent superiority of YOLOv8 over YOLOv5, and the strength of YOLOv8m and YOLOv8l.

Importantly, this work also addresses limitations common in previous research, such as class imbalance, over-reliance on ground-based imagery, and poor performance evaluation under limited data conditions. In contrast, the results here confirm that YOLOv8 models especially YOLOv8m maintain reliable accuracy and scalability, even without artificial enhancements.

The absence of augmentation and preprocessing, while reflecting low-resource conditions, limits generalizability under varying lighting and pavement textures. Thus, real-world deployment will require additional adaptation such as data augmentation, transfer learning, and larger datasets to maintain robustness.

Ultimately, this study presents a resource-efficient framework for UAV-based pavement monitoring, tailored to the operational constraints typical of developing regions. By emphasizing model efficiency, minimal data requirements, and statistical validation, it offers a practical benchmark for future intelligent infrastructure systems.

From a deployment perspective, YOLOv8's reduced file size translates into lower memory consumption and easier deployment on embedded devices. However, inference speed is influenced by both architecture and hardware acceleration, meaning that smaller models (YOLOv8n, YOLOv5n) offer higher FPS but at the expense of accuracy. Therefore, balancing accuracy, model size, and inference latency is critical when selecting models for UAV-based real-time monitoring.

5. Financial benefits

The outcomes of this research carry direct financial implications. Demonstrating that YOLOv8 models can deliver high accuracy with just 100–500 images and no preprocessing significantly reduces the need for large-scale data collection and costly annotation. Additionally, integrating UAVs with lightweight detection models minimizes reliance on manual inspections and survey vehicles, lowering labor expenses. Due to their low computational load, models like YOLOv8m can also run efficiently on embedded hardware, extending device lifespan and enabling deployment on affordable systems. Collectively, this approach reduces operational costs that save ~\$5,000–\$10,000 per km for UAV inspections versus traditional methods and ~\$2,000 per UAV unit versus survey vehicles, providing specific evidence for financial benefits. while enhancing detection speed and reliability. But Experimental deployment is constrained by UAV flight range, battery endurance, sensitivity to illumination and variable weather

conditions. Moreover, equipment quality and regulatory restrictions may limit large-scale surveys. Future research should address these challenges through optimized flight planning and sensor calibration to enhance scalability.

6. Conclusions

This study provides a comprehensive statistical validation of YOLOv8's superiority over YOLOv5 for UAV-based pavement distress detection, particularly under limited-data conditions and without relying on image preprocessing or augmentation. Using five dataset sizes (100–500 images), Two-Way ANOVA confirmed that YOLOv8 significantly outperformed YOLOv5 in terms of $mAP@0.5:0.95$ ($F = 13.81$, $p = 0.00062$), while dataset size also had a significant effect ($F = 5.32$, $p = 0.00156$) but no significant interaction was found, indicating stable performance trends. Repeated Measures ANOVA and Bonferroni-corrected pairwise t-tests further showed that YOLOv8l achieved the highest $mAP@0.5:0.95$ (0.381), while YOLOv8m delivered the best overall trade-off between accuracy (0.344), training time (67 minutes), and robustness across different dataset sizes. These findings strongly support the deployment of YOLOv8-based models in real-world UAV inspection workflows. Future enhancements include transfer learning with the EGY_PDD dataset and adaptive thresholding to improve robustness across diverse operational conditions. Future studies need to concentrate on integrating real-time geospatial capabilities to generate distress maps compatible with platforms like ArcGIS and QGIS, applying transfer learning to adapt models for diverse regions and conditions, and developing standardized UAV monitoring protocols that agencies worldwide can adopt. Such advancements will bridge the gap between experimental validation and operational practice, ultimately enhancing infrastructure monitoring efficiency and road safety. Future research ought to investigate the effect of varying UAV flight altitudes and the absence of multi-weather conditions on detection accuracy and structural health monitoring reliability. Although training efficiency was analyzed, inference speed on low-power embedded hardware (e.g., Jetson Nano, Raspberry Pi) remains to be quantified. Future work will benchmark YOLOv8m and YOLOv8l on such devices to assess real-time feasibility. An important consideration for future work is the risk of misdetections. In pavement infrastructure, even minor errors may result in costly maintenance or overlooked safety hazards. Future studies should integrate risk analysis to evaluate the practical consequences of model misclassifications.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflicts of interest

The authors declare no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Authors' Contribution Statement

Mohamed E. Mohamed: Conceptualization, Methodology, Data curation, Writing – original draft.

Adel A. Esmat: Supervision, Validation, Writing – review & editing.

Ahmed M. Hamdy: Formal analysis, Visualization, Software.

Amr M. El Sheshtawy: Investigation, Project administration, Resources.

Refreneces

- [1] Sekimoto DA and HM and SKG and DT and AM and TK and Y. Deep learning-based road damage detection and classification for multiple countries. *Autom Constr* 2021;132:103935. <https://doi.org/10.1016/j.autcon.2021.103935>.
- [2] Zahran Badr, Mohamed El Gendy MER. Using Artificial Neural Networks to Predict the Distresses of Flexible Pavement. *J Al-Azhar Univ Eng Sect* 2022;17:17–25. <https://doi.org/10.21608/aej.2022.216572>.
- [3] Marwan Elsayed; H. Mahdi; K. Kandil. Investigating the Factors Affecting Pavement Overlay Service Life. *Al-Azhar Univ Civ Eng Res Mag* 2019;41:152–64.
- [4] A. A. Rabah ASEHMMR and MEG. A New Revised Version Predictive Model of Elastic Modulus Using Artificial Neural Network. *Al-Azhar Univ Civ Eng Res Mag* 2021;43:285–96.
- [5] B MK a, B MG, B OR, C BD. Signal processing methodology for detection and localization of damages in columns under the effect of axial load. *Measurement* 2023;211. <https://doi.org/10.1016/j.measurement.2023.112595>.
- [6] Fakharian P, Naderpour H. Damage Severity Quantification Using Wavelet Packet Transform and Peak Picking Method. *Pract Period Struct Des Constr* 2022;27:4021063. [https://doi.org/10.1061/\(asce\)sc.1943-5576.0000639](https://doi.org/10.1061/(asce)sc.1943-5576.0000639).
- [7] A BM, B MK, B OR, B MG, Gholamreza Ghodrati Amiri a. Efficient modal flexibility signal processing methodology for innovative identification of debonding zones in concrete-filled steel tubes. *Results Eng* 2025;26. <https://doi.org/10.1016/j.rineng.2025.104683>.
- [8] Khanahmadi M. A cutting-edge framework for damage-sensitive feature extraction leveraging modal dynamic flexibility in signal processing-driven structural health monitoring. *Thin-Walled Struct* 2025;216. <https://doi.org/10.1016/j.tws.2025.113617>.
- [9] Naderpour H, Fakharian P. A synthesis of peak picking method and wavelet packet transform for structural modal identification. *KSCE J Civ Eng* 2016;20:2859–67. <https://doi.org/10.1007/s12205-016-0523-4>.
- [10] Zhu J. Pavement distress detection using convolutional neural networks with images captured via UAV. *Autom Constr* 2022. <https://doi.org/10.1016/j.autcon.2021.103991>.
- [11] Samadzadegan F, Dadrass Javan F, Ashtari Mahini F, Gholamshahi M, Nex F. Automatic road pavement distress recognition using deep learning networks from unmanned aerial imagery. *Drones* 2024;8:244. <https://doi.org/10.3390/drones8060244>.
- [12] Zhu J et al. Pavement distress detection using convolutional neural networks with images captured via UAV. *Autom Constr* 2022;133:103991. <https://doi.org/10.1016/j.autcon.2021.103991>.
- [13] Yan K, Zhang Z. Automated asphalt highway pavement crack detection based on deformable single shot multi-box detector under a complex environment. *IEEE Access* 2021;9:150925–150938. <https://doi.org/10.1109/ACCESS.2021.3125703>.
- [14] Song L, Wang X. Faster region convolutional neural network for automated pavement distress detection. *Road Mater Pavement Des* 2021;22:23–41. <https://doi.org/10.1080/14680629.2019.1614969>.
- [15] Silva LA, Sanchez San Blas H, Peral García D, Sales Mendes A, Villarubia González G. An architectural multi-agent system for a pavement monitoring system with pothole recognition in UAV images. *Sensors* 2020;20:6205. <https://doi.org/10.3390/s20216205>.
- [16] Shaghouri AA, Alkhatib R, Berjaoui S. Real-time pothole detection using deep learning. *ArXiv* 2021. <https://doi.org/10.48550/arXiv.2107.06356>.
- [17] Shu Z, Yan Z, Xu X. Pavement crack detection method of street view images based on deep learning. *J Phys Conf Ser* 2021;1952:022043. <https://doi.org/10.1088/1742-6596/1952/2/022043>.
- [18] Park S-S, Tran V-T, Lee D-E. Application of various yolo models for computer vision-based real-time pothole detection. *Appl Sci* 2021;11:11229. <https://doi.org/10.3390/app112311229>.
- [19] Hu GX, Hu BL, Yang Z, Huang Y, Li P. Pavement crack detection method based on deep learning models. *Wirel Commun Mob Comput* 2021;2021:5573590. <https://doi.org/10.1155/2021/5573590>.
- [20] Xu X et al. Crack detection and comparison study based on Faster R-CNN and Mask R-CNN. *Sensors* 2022;22. <https://doi.org/10.3390/s22031215>.
- [21] Guo L, Li R, Jiang B. A road surface damage detection method using YOLOv4 with PID optimizer. *Int J*

- Innov Comput Inf Control 2021;17:1763–1774. <https://doi.org/10.24507/ijicic.17.05.1763>.
- [22] El-Sayed MFA and MAH and ES and AAE-S and RFA-K and AM and E. EGY_PDD: a comprehensive multi-sensor benchmark dataset for accurate pavement distress detection and classification. *Multimed Tools Appl* 2025. <https://doi.org/10.1007/s11042-025-20700-w>.
- [23] Wang LF and ST and MKA b. MA and MISI and X. Impact of Image Preprocessing and Crack Type Distribution on YOLOv8-Based Road Crack Detection. *Sensors* 2025;25:2180. <https://doi.org/10.3390/s25072180>.
- [24] Zhang XC and CW and CL and XZ and YZ and TL and J. Autonomous Crack Detection for Mountainous Roads Using UAV Inspection System. *Sensors* 2024;24:4751. <https://doi.org/10.3390/s24094751>.
- [25] Guo JZ and HX and PL and KZ and WH and R. Pavement Crack Detection Method via Deep Learning and a Binocular-Vision-Based Unmanned Aerial Vehicle. *Appl Sci* 2024;14:1778. <https://doi.org/10.3390/app14051778>.
- [26] Lin YTJ and HTY and YRZ and KQW and RYL and CY. RDD-YOLOv5: Road Defect Detection Algorithm with Self-Attention Based on Unmanned Aerial Vehicle Inspection. *Sensors* 2023;23:8241. <https://doi.org/10.3390/s23198241>.
- [27] Wu XX and HH and YD and YZ and S. GC-YOLOv5s: A Lightweight Detector for UAV Road Crack Detection. *Appl Sci* 2023;13:11030. <https://doi.org/10.3390/app131911030>.
- [28] Li XW and HG and ZJ and Z. BL-YOLOv8: An Improved Road Defect Detection Model Based on YOLOv8. *Sensors* 2023;23:8361. <https://doi.org/10.3390/s23208361>.
- [29] Ultralytics. , YOLOv8. Available:<https://GithubCom/Ultralytics/Ultralytics> 2025.
- [30] Huang GW and YC and PA and HH and JH and T. UAV-YOLOv8: A Small-Object-Detection Model Based on Improved YOLOv8 for UAV Aerial Photography Scenarios. *Sensors* 2023;23:7190. <https://doi.org/10.3390/s23167190>.
- [31] Wang XC and CL and LC and XZ and YZ and C. A Pavement Crack Detection and Evaluation Framework for a UAV Inspection System Based on Deep Learning. *Appl Sci* 2024;14:1157. <https://doi.org/10.3390/app14031157>.
- [32] Nouri Y, Shahabian F, Shariatmadar H, Entezami A. Structural Damage Detection in the Wooden Bridge Using the Fourier Decomposition, Time Series Modeling and Machine Learning Methods. *J Soft Comput Civ Eng* 2024;8:83–101. <https://doi.org/10.22115/scce.2023.401971.1669>.
- [33] Soleimani Nezhad, S., Khademian, F., Naderpour, H., Kalantari, S. M., & Fakharian P. Signal Processing-Based Damage Detection of Steel Braced Frame Subjected to Consequent Excitations. *Innov Infrastruct Solut* 2024;9. <https://doi.org/10.1007/s41062-024-01762-5>.
- [34] A HRA, B NM, Mahmoud Bayat c. A novel damage identification method based on short time Fourier transform and a new efficient. *Structures* 2021;33:3605–14. <https://doi.org/10.1016/j.istruc.2021.06.081>.
- [35] Agüera-Vega F. C-RF. M-CP. Accuracy of digital surface models and orthophotos derived from unmanned aerial vehicle photogrammetry. *J Surv Eng* 2017;143:04016025. [https://doi.org/10.1061/\(ASCE\)SU.1943-5428.0000206](https://doi.org/10.1061/(ASCE)SU.1943-5428.0000206).
- [36] Pádua L. MP. HJ. AT. PE. MR. SJJ. UAV-Based Forest Fire Detection and Monitoring Using Multi-Temporal and Multi-Sensor Data. *Remote Sens* 2022;14:983. <https://doi.org/10.3390/rs14040983>.
- [37] Redmon J. You Only Look Once. *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016.
- [38] Astrup SP and R. Automatic detection of snow breakage at single tree level using YOLOv5 applied to UAV imagery. *Int J Appl Earth Obs Geoinf* 2022;112:102946. <https://doi.org/10.1016/j.jag.2022.102946>.
- [39] Lee JD and YL and VCS. Concrete crack detection with handwriting script interferences using faster region-based convolutional neural network. *Comput Aided Civ Infrastruct Eng* 2020;35:373–388. <https://doi.org/10.1111/mice.12497>.
- [40] Wilson AC. RR. SM. SN. RB. The marginal value of adaptive gradient methods in machine learning. *Adv Neural Inf Process Syst* 2017;30:4148–58. <https://doi.org/10.48550/arXiv.1705.08292>.
- [41] E. Fawaz. *Statistics lectures : ANOVA-test* 2022.