DYNA
http://dyna.medellin.unal.edu.co/

UNIVERSIDAD
NACIONAL
DE COLOMBIA

Classification of pothole distress severity in asphalt pavements using YOLOv8

Átila Marconcine de Souza, Vinicius Fier Cestari & Heliana Barbosa Fontenele

Departamento de Construção Civil, Universidade Estadual de Londrina, Londrina, Brasil. atila.marconcine@uel.br, vinicius.fier.cestari@uel.br, heliana@uel.br

Received: May 9th, 2025. Received in revised form: June 27th, 2025. Accepted: July 7th, 2025.

Abstract
Potholes are a type of distress that occurs in pavement surfaces. According to the method adopted for distress surveys, potholes are classified into three levels of severity: low, medium, and high. The severity assessment is traditionally performed through slow and labor-intensive manual procedures. To automate this process, this study employed the YOLOv8s and YOLOv8m models to detect pothole distress and classify its severity. During the training phase, YOLOv8m achieved the best evaluation metrics, while YOLOv8s outperformed in the testing phase, particularly in recognizing high-severity potholes. However, both models failed to effectively detect low and medium severity levels, indicating the need for improvements before field application. One possible explanation for this limitation is the lack of depth information in the input images, a factor that will be addressed in future research.

Keywords: deep learning; machine learning; computer vision; object detection.

Clasificación de la severidad del deterioro tipo bache en pavimentos asfálticos utilizando YOLOv8

Resumen
Los baches son un tipo de deterioro que ocurre en las superficies de los pavimentos. De acuerdo con el método adoptado para el levantamiento de deterioros, los baches se clasifican en tres niveles de severidad: baja, media y alta. La determinación del nivel de severidad se realiza mediante procedimientos manuales que son lentos y extenuantes. Con el objetivo de automatizar este proceso, este estudio utilizó los modelos YOLOv8s y YOLOv8m para detectar el deterioro tipo bache y clasificar su severidad. En la etapa de entrenamiento, el modelo YOLOv8m obtuvo las mejores métricas, mientras que en la etapa de prueba el YOLOv8s mostró el mejor desempeño, destacándose en el reconocimiento de baches con severidad alta. No obstante, ambos modelos fueron incapaces de reconocer con precisión los niveles de severidad baja y media, lo que indica la necesidad de mejoras para su aplicación en campo. Una posible explicación de esta limitación es la ausencia de información de profundidad en las imágenes utilizadas, cuestión que será abordada en estudios futuros.

Palabras clave: aprendizaje profundo; aprendizaje automático; visión por computadora; detección de objetos.

1. Introduction

The quality of roadways is essential to daily life, directly influencing user safety and comfort, as well as impacting vehicle and roadway maintenance costs. Asphalt pavements, due to their cost-effectiveness and ability to withstand heavy loads, are widely used but are susceptible to the emergence of distresses that can significantly compromise their functionality.

Thus, pavement performance must be monitored, and maintenance (both preventive and corrective) should occur at the appropriate time [1]. For this purpose, the implementation of a Pavement Management System (PMS) is required. The PMS is responsible for managing the existing road infrastructure, ensuring optimal traffic conditions. The core component of the PMS is pavement evaluation. Through this process, it is possible to identify the current condition of the pavement, the existing distresses, their severity level, and their location.

How to cite: de Souza, A.M., Cestari, V.F., and Fontenele, H.B., Classification of pothole distress severity in asphalt pavements using YOLOv8. DYNA, (92)238, pp. 47-56, July - September, 2025.

Universidad Nacional de Colombia.
Revista DYNA, (92)238, pp. 47-56, July - September, 2025, ISSN 0012-7353
DOI: https://doi.org/10.15446/dyna.v92n238.120252

Among the various types of distresses, potholes are particularly notable, as they represent the most common and impactful form of pavement deterioration, severely compromising serviceability, according to [2]. Potholes are cavities of varying sizes that can appear in any part of the pavement surface [3]. This type of distress can be caused by traffic loads and weather conditions (particularly in areas with interconnected cracking), construction failures, or disintegration due to mix design deficiencies [4]. Potholes are categorized into three severity levels (low, medium, and high), with each level defined based on area and depth [3].

Determining the severity level of distresses is fundamental for identifying which roads require Maintenance and Rehabilitation (M&R). Several methodologies exist for distress severity classification, such as ASTM D6433-24 [5], the Distress Identification Manual (DIM) [6], and the Brazilian Distress Identification Manual (Manual de Identificação de Defeitos - MID) [3]. Although each procedure presents distinct methods, they share a common characteristic: all require manual, in-field evaluation conducted by engineers and/or technicians. Due to the reliance on human labor, these evaluations are not only time-consuming and tiring but also prone to errors and subjectivity.

In the past decade, [7] observed a significant increase in studies employing smartphones in conjunction with Machine Learning (ML) techniques, thus providing a more automated methodology for processing field-acquired data. There are numerous ML and Deep Learning (DL) algorithms. Among the models capable of recognizing objects in images, the You Only Look Once (YOLO) architecture stands out. The first version of YOLO was developed by [8], and currently, the object detector has over ten versions, with YOLOv8 [9], YOLOv9 [10], YOLOv10 [11], YOLOv11 [12] and YOLOv12 [13] being the most recent. YOLO architectures are applied across various domains, such as medicine [14, 15] and agriculture [16,17]. In engineering, YOLO is used for the automatic detection of pavement distresses, being capable of identifying potholes [18-20], cracks [21-23], and multiple distresses simultaneously [24-26].

However, the aforementioned studies only explore the models' capability to detect distresses, without addressing severity classification. In the literature, the automatic classification of pavement distress severity levels remains underexplored, with few published works. Among the studies that address this topic using YOLO architectures, [27-29] can be mentioned. [27] used different versions of YOLOv5 to detect block cracks at two severity levels (low and high) using images obtained from an evaluation vehicle. The best results were achieved with YOLOv5m, which attained a precision of 72.3%, recall of 78.2%, and mAP of 76.7%. [28] employed three object detectors (YOLOv5, YOLOv8, and CenterNet) for the detection of five types of cracks (longitudinal, transverse, fatigue, diagonal, and block) and their respective three severity levels. YOLOv8 achieved the best results, with a precision of 52%, recall of 58.4%, and mAP of 60%. However, despite addressing severity, the criteria used to define it were not specified. In [29], the authors trained three public datasets using YOLOv4 for the recognition of multiple distresses and their severity levels. The distresses studied included cracks (fatigue, longitudinal, transverse, and edge), patching, and potholes. The reported precision, recall, and mAP were 90%, 90%, and 87.44%, respectively. Nonetheless, the authors noted that no depth-related information on the potholes was available, and severity classification was based solely on image texture (i.e., darker textures were assumed to represent higher severity and vice versa).

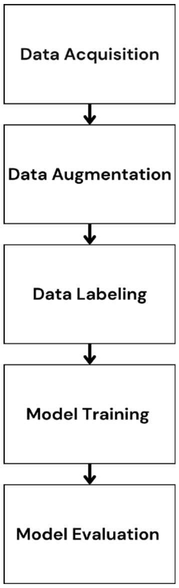


Figure 1. Methodology Steps
Source: The Authors.

Given the above, the present study aims to evaluate, in an automated manner, the severity classification of potholes in urban asphalt pavements. To this end, two versions of the YOLOv8 object detector (YOLOv8s and YOLOv8m) were used, trained on a proprietary dataset in which the severity levels were determined through field measurements. The YOLOv8 architecture was selected due to the optimized accuracy-speed tradeoff, making it suitable for real time object detection in diverse applications [9]. Furthermore, YOLO effectiveness in pavement distress and severity detection has been demonstrated in recent literature [27-29].

2. Methodology

This section presents and discusses all the procedures carried out throughout the study. It is divided into six stages (Fig. 1), namely: Data Acquisition, Data Augmentation, Data Labeling, Model Training, and Model Evaluation.

2.1 Data acquisition

The first stage consisted of image collection, during which an initial dataset of pothole images from urban roads was compiled. These images were captured using a Galaxy M52 smartphone equipped with a 64-megapixel camera.

Table 1.
Severity classification.

Severity	Depth (cm)	Area (m²)
Low	< 2.5	< 0.28
Medium	< 2.5	> 0.28
Medium	2.5 a 5.0	< 0.28
Medium	> 5.0	< 0.10
High	2.5 a 5.0	> 0.28
High	>5,0	> 0.10

Source: The Authors.

In addition to photographing the distresses, field measurements were also conducted to determine their severity levels. For this purpose, MID [3] was employed. According to [3] , potholes must be measured based on their depth (in centimeters) and surface area (in square meters). The area is determined using a circumscribed rectangle, with one side aligned parallel to the road axis.

Table 1 presents the severity classification criteria according to the MID. A total of 20 images were collected for each severity level (high, medium, and low), totaling 60 images. Of these, 36 were used for model training, 12 for validation, and 12 for testing. Additionally, the dataset developed by [30] was used, which includes 12 potholes images — 10 of high severity and 2 of medium severity.

2.2 Data augmentation

Due to the limited number of collected images, data augmentation techniques were applied. This technique is commonly used to increase the size and diversity of training sets by applying transformations to the original images [31]. Various transformations can be employed; in this study, zoom, shear, rotate, 90º rotation, and flip were used. The Python library Augmentor was used to implement data augmentation, via the PyCharm Integrated Development Environment (IDE). Figure 2 displays the code used, with each line representing an applied operation.

A total of 60 images underwent augmentation, including 48 original images and 12 from [30]. As a result, 2,618 images were generated — 2,225 for training (85%) and 393 for validation (15%). Table 2 shows the distribution of images per severity level in the final dataset.

2.3 Data labeling

After augmentation, the images were annotated. This process involved manually drawing bounding boxes around the objects of interest in the dataset. In this study, the target objects were potholes on pavements and their associated severity levels. All 2,618 images were manually labeled using the LabelStudio platform, as shown in Fig. 3.

2.4 Model training

The model selected for training was YOLOv8, a version released in 2023 by Ultralytics (also responsible for YOLOv5 and more recently YOLOv11). YOLOv8 offers state-of-the-art performance in terms of accuracy and speed compared to previous YOLO versions [9].

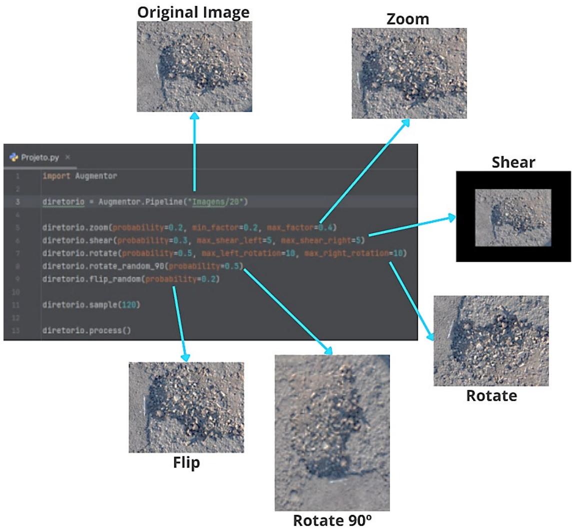
Its architecture is divided into Backbone, Neck, and Head, as illustrated in Fig. 4. Although structurally similar to YOLOv5, key differences, according to [32], include:

- Use of a modified CSPDarknet53 convolutional neural network in the Backbone;
- Replacement of the CSPLayer (used in YOLOv5) with the C2f module;
- Incorporation of a Spatial Pyramid Pooling Fast (SPPF) layer, which accelerates computation by pooling features into a fixed size map;
- Each convolution is followed by batch normalization and a SiLU activation;
- The Head is decoupled to process classification and regression tasks independently.

Table 2.
Dataset Split.

Severity	Training	Validation
Low	396	132
Medium	569	129
High	1260	132
Total	2225	393

Source: The Authors.



```
import Augmentor

diretorio = Augmentor.Pipeline("Imagens/20")

diretorio.zoom(probability=0.2, min_factor=0.2, max_factor=0.4)
diretorio.shear(probability=0.3, max_shear_left=5, max_shear_right=5)
diretorio.rotate(probability=0.5, max_left_rotation=10, max_right_rotation=10)
diretorio.rotate_random_90(probability=0.5)
diretorio.flip_random(probability=0.2)

diretorio.sample(120)

diretorio.process()
```

Figure 2. Code and Operations.
Source: The Authors.

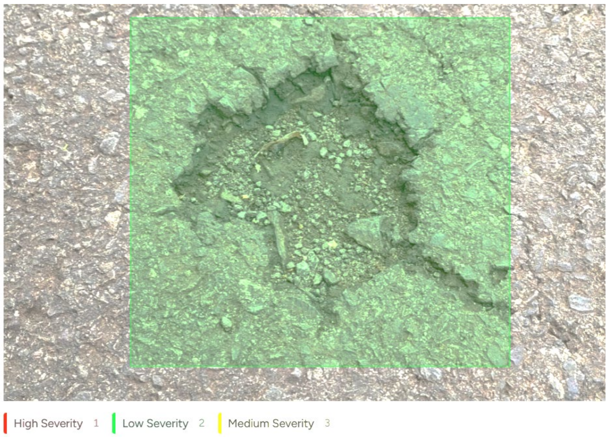


Figure 3. Labeling via LabelStudio
Source: The Authors.

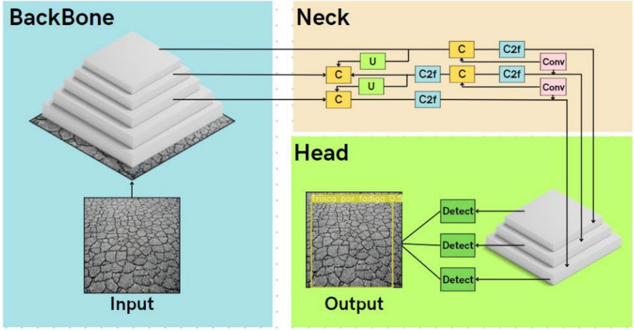


Figure 4. YOLOv8 architecture.
Source: [24].

YOLOv8 is available in five versions: YOLOv8n (nano), which is lightweight and fast but less accurate; YOLOv8s (small); YOLOv8m (medium); YOLOv8l (large); and YOLOv8x (extra-large), which is slower and heavier but delivers higher accuracy. In this study, the YOLOv8s and YOLOv8m versions were used.

Model training was conducted locally using the Anaconda development environment and the PyTorch library. YOLOv8s was trained on a notebook with an NVIDIA GeForce GTX 1660 Ti GPU, Intel Core i7 9th generation processor, and 16 GB of RAM, while YOLOv8m was trained on a notebook with an NVIDIA GeForce RTX 4050 GPU, Intel Core i7 13th generation processor, and 32 GB of RAM.

2.5 Model evaluation

To assess the model's performance, standard machine learning evaluation metrics were employed, including Confusion Matrix, Precision, Recall, Average Precision (AP), and mean Average Precision (mAP).

The Confusion Matrix (Fig. 5) is a 2x2 matrix in which the main diagonal represents correct predictions, and the secondary diagonal represents errors. The main diagonal includes True Positive (TP) and True Negative (TN) values. A TP occurs when the model correctly detects a pothole; a TN occurs when the model correctly identifies the absence of potholes. The secondary diagonal includes False Positive (FP) and False Negative (FN) values. An FP indicates the model predicted a pothole when there was none, while an FN indicates the model failed to detect an existing pothole.

Precision (eq. 1) uses TP and FP values to measure the model's ability to correctly identify and localize objects. Recall (eq. 2) uses TP and FN values to determine how many of the actual objects were correctly detected.

Using Precision and Recall, the precision-recall curve is generated. The area under this curve represents the AP (eq. 3), which quantifies the model's performance for a specific class. For performance across all classes, mAP (eq. 4) is used.

According to [24], YOLOv8 outputs two mAP values at the end of training: mAP@50 and mAP@50-95. The first measures performance at a 50% confidence threshold, while the latter evaluates performance across multiple thresholds ranging from 50% to 95%.

In addition to these metrics, the model was also evaluated through testing using 12 unseen images (4 per severity level) to assess practical performance in real-world conditions.

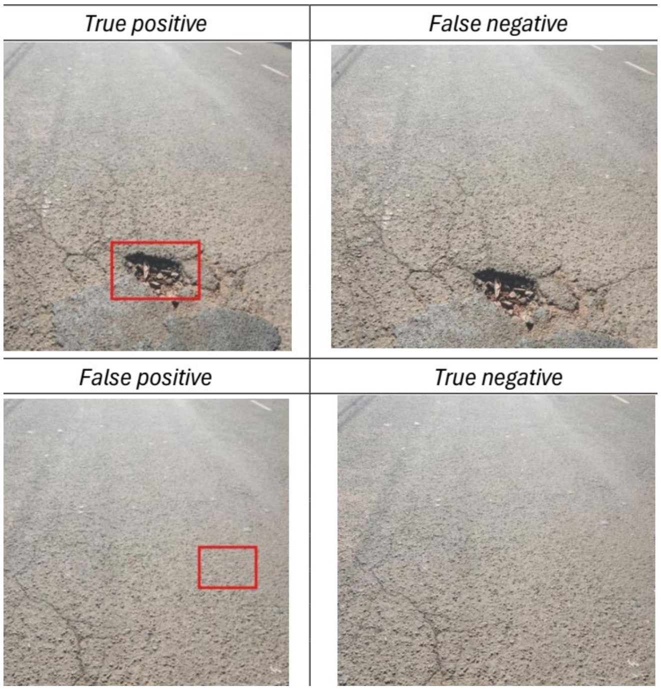


Figure 5. Confusion Matrix
Source: The Authors.

3. Results and discussions

3.1 Training results

This section presents the results of the training and testing processes for YOLOv8s and YOLOv8m. The training for YOLOv8s was configured for 200 epochs, with the best results obtained at epoch 163.

$$Precision \ = \frac{TP}{TP+FP} \tag{1}$$

$$Recall \ = \frac{TP}{TP+FN} \tag{2}$$

$$AP \ = \int_{0}^{1} Precision(recall)\ drecall \tag{3}$$

$$mAP \ = \frac{1}{Number\ of\ classes} \times \sum_{i=1}^{Number\ of\ classes} APi \tag{4}$$

Fig. 6 shows the precision-recall curve, which relates the precision and recall metrics and is used to calculate the AP for each class and the overall mAP. The curve indicates that the model achieved a satisfactory mAP@50 of approximately 50%. High and medium severity levels yielded above-average AP values — 59.60% and 77.60%, respectively — while the low severity class underperformed, achieving only 9% AP.

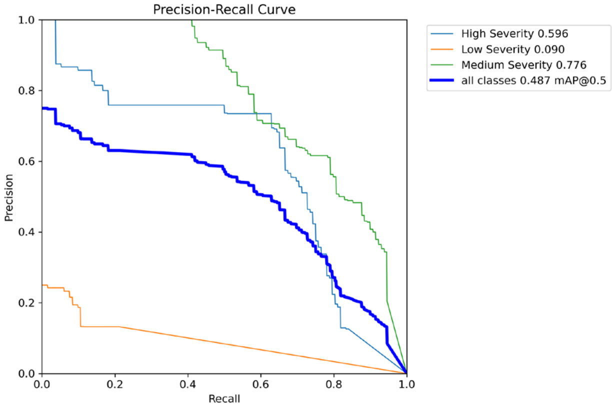


Figure 6. YOLOv8s Precision-Recall Curve
Source: The Authors

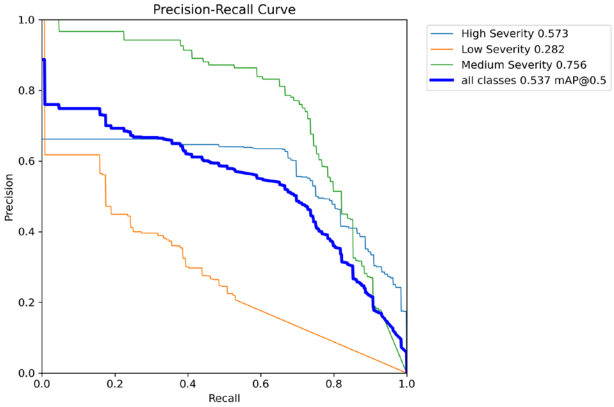


Figure 7. YOLOv8m Precision-Recall Curve
Source: The Authors.

For YOLOv8m, the precision-recall curve in Fig. 7 shows that the low severity class achieved better performance than with YOLOv8s, reaching 28.10% AP at epoch 125 — a 19.10% improvement. This also positively impacted the mAP@50 across all classes, which increased from 48.70% with YOLOv8s to 53.70% with YOLOv8m.

Table 3 presents the evaluation metrics for the best-performing epochs of the YOLOv8s and YOLOv8m models. It shows that the low severity class had the worst performance in YOLOv8s, with a precision of 23.80% and a recall of just 1.52%. However, significant improvement was observed in YOLOv8m, where the low severity class had the highest precision (61.30%). Additionally, recall improved for all three severity levels, reaching 76.6% for high severity.

Despite these improvements, YOLOv8m showed a decrease in precision and mAP@50 for the medium and high severity levels. Nevertheless, this decline did not affect the overall model performance. In conclusion, YOLOv8m outperformed YOLOv8s, achieving 56.90% precision, 56.20% recall, 53.70% mAP@50, and 32.40% mAP@50–95.

To potentially achieve better training evaluation metrics, more advanced versions such as YOLOv8x and YOLOv8xl could be considered. These models include more parameters and tend to produce better results. However, their training time is significantly longer and requires greater computational power.

Table 3.
Evaluation Metrics for the Training Phase

Model	C	Evaluation Metrics (%)			
		Precision	**Recall**	**mAP50**	**mAP50-95**
YOLOv8s	LS	23.80	1.52	9.03	3.83
	MS	70.40	60.50	77.60	50.8
	HS	68.80	64.40	59.60	35.10
	All	54.30	42.10	48.70	29.90
YOLOv8m	LS	61.30	15.90	28.0	12.10
	MS	60.20	76.00	75.60	51.3
	HS	49.30	76.60	57.30	33.90
	All	56.90	56.20	53.70	32.40

Legend: C: Class; LS: Low Severity; MS: Medium Severity; HS: High Severity.
Source: The Authors.

Table 4.
Confusion Matrix and Evaluation Metrics for the Test Phase

Model	C	NI	Predicted				EM (%)	
			LS	**MS**	**HS**	**NC**	**P**	**R**
YOLOv8s	LS	4	0	0	2	2	0.00	0.00
	MS	4	0	0	2	2	0.00	0.00
	HS	4	0	1	3	0	42.86	75.00
	All	12	0	1	7	4	14.29	25.00
YOLOv8m	LS	4	0	1	1	2	0.00	0.00
	MS	4	1	0	3	0	0.00	0.00
	HS	4	1	2	3	0	42.86	50.00
	All	12	2	3	7	2	14.29	16.67

Legend: C: Class; NI: Number of Images; LS: Low Severity; MS: Medium Severity; HS: High Severity; NC: Not Classified; P: Precision; R: Recall; EM: Evaluation Metrics.
Source: The Authors

3.2 Test results

As previously mentioned, 12 images from the dataset — 4 for each severity level — were reserved for testing and excluded from the training process. These 12 images were evaluated by both trained models using a 30% confidence threshold. Table 4 presents the confusion matrices with the testing results for both models.

Table 4 shows that during testing, both models yielded unsatisfactory results for the low and medium severity levels, failing to correctly detect any of the four images in each class, thus resulting in precision and recall values of 0%. In contrast, the high severity class showed better performance, 42.86% precision and 75.00% recall for YOLOv8s, and 42.86% precision and 50.00% recall for YOLOv8m. These stronger results for the high severity class can be attributed to its greater representation in the dataset.

Tables 5, 6, and 7 provide a more detailed view of how the models classified the images. Table 5 shows the test results for the high severity class. It explains the drop in recall from 75% with YOLOv8s to 50% with YOLOv8m. This drop was due to the bounding box overlap in YOLOv8m (see IDs 1 and 2 in Table 5): while YOLOv8s correctly detected a single high-severity pothole, YOLOv8m assigned multiple severity levels to the same distress. Thus, YOLOv8s performed better for high severity in testing, correctly identifying 3 out of 4 images.

Tables 6 and 7 show the results for medium and low severity classes. As previously stated, no true positives were recorded in these classes, which justifies the precision and recall values of 0%. These test results suggest that the models are not yet viable for recognizing medium and low severity potholes.

One plausible explanation for these inadequate results is the absence of depth information in the images. As previously discussed, severity classification for potholes is based on both area and depth. However, all images used in training and testing were 2D, which means YOLO only extracted features in two dimensions, disregarding depth — a third-dimensional parameter. This hypothesis is supported by cases such as potholes in IDs 2 and 4 in Table 6 and ID 3 in Table 7. These potholes have large surface areas but shallow depths. In the field, they were classified as medium or low severity, but the model interpreted them as high severity due to their large visible area.

There are several alternatives to address this issue. As previously mentioned, [29] used texture-related information to estimate pothole depth. They converted RGB images to grayscale and applied image enhancement techniques, including Histogram Equalization, Average Filtering, Median Filtering, and Gaussian Filtering.

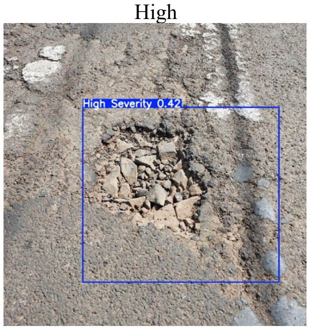
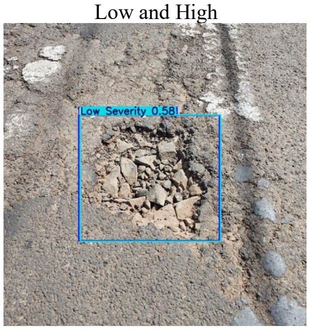
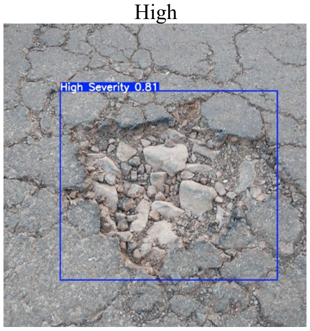
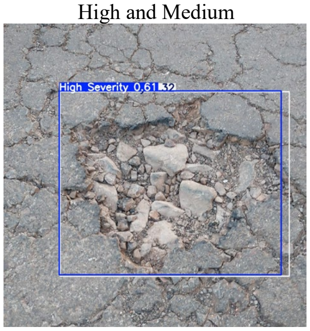
Another solution was proposed by [33], who first trained RetinaNet for pothole detection without considering severity levels, achieving 99.0% precision and recall. Then, they applied the photogrammetric process of Structure from Motion (SfM) to derive 3D geometric data from 2D images. This process produced a point cloud, which was used in a Python script to estimate pothole depth and, consequently, classify its severity. Their methodology first applies DL for defect detection, followed by severity estimation based on 3D analysis.

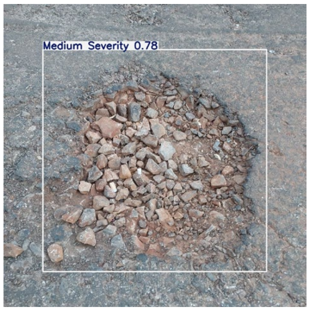
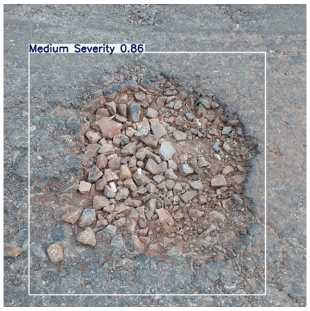
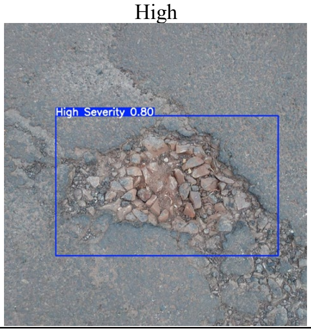
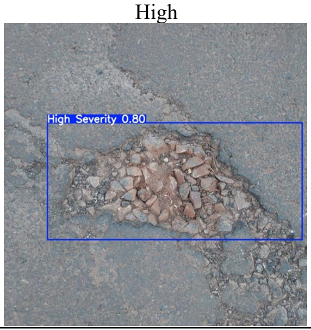
In addition to these approaches, sensors and lasers (e.g., LiDAR) can provide precise 3D data. However, such devices are typically expensive. [34] mention more affordable options such as Microsoft Kinect, Structure Sensor, and Intel RealSense.

Other computer vision techniques like 3D object detection, monocular 3D object detection, and monocular depth estimation may also prove useful for future research. However, the literature has not reported the use of these techniques for pothole detection

Finally, it is worth noting that the YOLO architecture has undergone further development beyond YOLOv8, with significant improvements in versions YOLOv9 [10], YOLOv10 [11], YOLOv11 [12], and YOLOv12 [13]. Future research is encouraged to explore these newer architectures.

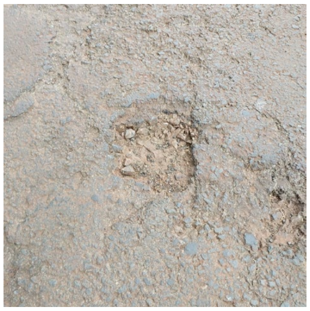
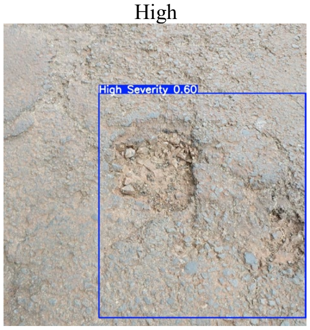
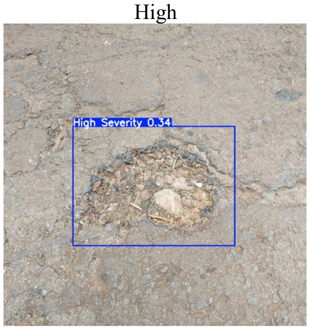
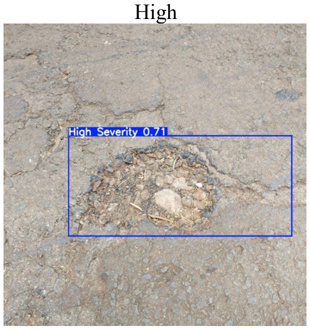
Table 5.
Testing Stage for High Severity

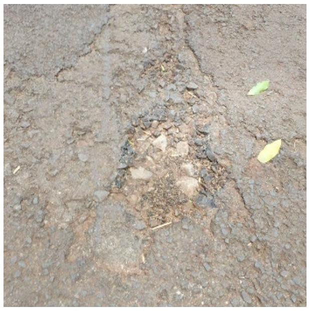
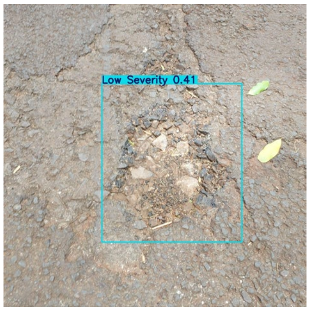
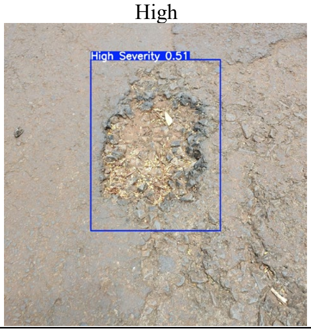
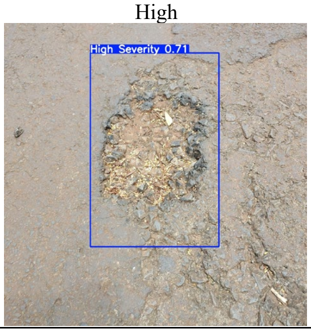
ID	Manual Field Measurements (cm)			Severity (in-situ)	Severity (YOLOv8s)	Severity (YOLOv8m)
	Length	Width	Depth			
1	36.60	33.50	6.40	High	High 	Low and High 
2	47.80	38.70	5.20	High	High 	High and Medium 
3	56.30	51.10	5.30	High	Medium	Medium

						
4	43.70	23.90	5.90	High	High 	High 

Source: The Authors.

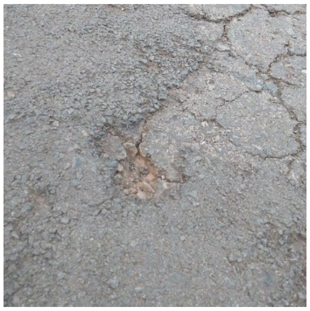
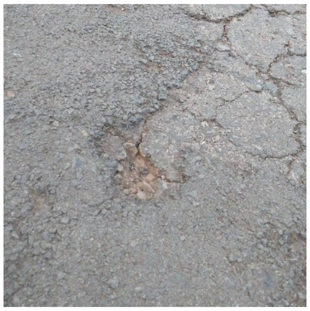
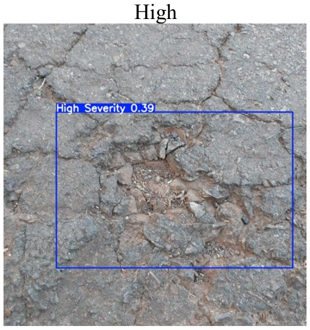
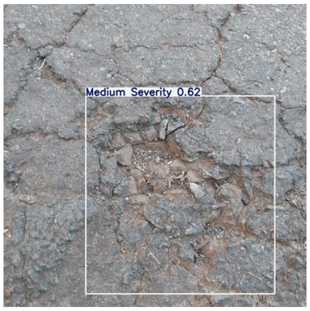
Table 6.
Testing Stage for Medium Severity

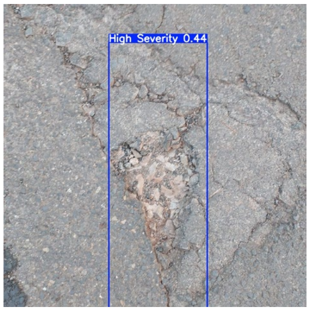
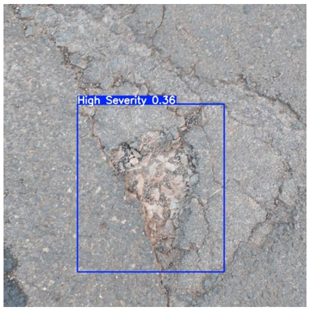
ID	Manual Field Measurements (cm)			Severity (in-situ)	Severity (YOLOv8s)	Severity (YOLOv8m)
	Length	Width	Depth			
1	21.50	19.40	2.80	Medium	Not Classified 	High 
2	31.30	18.90	3.10	Medium	High 	High 
3	31.80	24.30	2.90	Medium	Not Classified	Low

						
4	29.70	19.90	3.10	Medium	High 	High 

Source: The Authors.

Table 7.
Testing Stage for Low Severity

ID	Manual Field Measurements (cm)			Severity (in-situ)	Severity (YOLOv8s)	Severity (YOLOv8m)
	Length	Width	Depth			
1	7.90	7.30	2.20	Low	Not Classified 	Not Classified 
2	11.50	6.40	2.30	Low	High 	Medium 
3	19.90	14.40	2.50	Low	High	High

						
4	19,80	8,40	2,30	Low	Not Classified 	Not Classified

Source: The Authors.

Other object detectors outside the YOLO family may also be viable, such as R-CNN [35] and RetinaNet [36]. Furthermore, the use of deep learning architectures with segmentation blocks and monocular depth estimation blocks, as used in [37], can also provide better results in the accurate recognition of potholes and their severity levels.

4. Conclusion

In this study, two versions of the YOLOv8 architecture (YOLOv8s and YOLOv8m) were trained to recognize the severity level of pothole distresses in asphalt pavements. Based on the training results, the version with a larger number of parameters — YOLOv8m — achieved better evaluation metrics. However, during the testing phase, YOLOv8s showed superior performance, particularly in detecting high-severity potholes. For medium and low severity levels, both models proved ineffective, failing to correctly identify these classes — a shortcoming that may be attributed to the lack of depth-related data in the input images. It can be concluded that the trained models require significant improvements before being viable for practical applications.

Future studies will explore the possibility of developing a system capable of detecting potholes and determining their severity while incorporating depth information. In addition, new pothole images will be added to the database, thus improving the generalization capacity of YOLO models and their applicability in real-world.

Acknowledgements

The authors would like to thank the Brazilian Coordination for the Improvement of Higher Education Personnel (CAPES) for the scholarship granted, as well as the Transportation Engineering Laboratory of the Center for Technology and Urbanism (LET-CTU) and the State University of Londrina (UEL) for the infrastructure provided.

References

[1] Zanchetta, F., Sistema de Gerência de Pavimentos Urbanos: avaliação de campo, modelo de desempenho e análise econômica, PhD dissertation, Departamento de Transportes, Universidade de São Paulo, São Carlos, 2017.

[2] Singh, P., Wijethunga, R., Sadhu, A., and Samarabandu, J., Expert evaluation system for pothole defect detection. Expert Systems with Applications, 277, art. 127280, 2025. DOI: https://doi.org/10.1016/j.eswa.2025.127280

[3] Domingues, F.A.A., MID: manual de identificação de defeitos de revesttimentos asfálticos de pavimentos. São Paulo: Camargo Campos, 1993.

[4] Bernucci, L.B., Motta, L.M.G., Ceratti, J.A.P., and Soares, J.B., Pavimentação asfáltica: formação básica para engenheiros, 2a ed. Rio de Janeiro: petrobras: ABEDA, 2022.

[5] ASTM, D6433., Standard practice for roads and parking lots pavement condition index surveys, ASTM International, 2024.

[6] Miller, J.S., and Bellinger, W.Y., Distress identification manual for the long-term pavement performance program, 2014.

[7] Souza, V.M.A., Giusti, R., and Batista, A.J.L., Asfault: a low-cost system to evaluate pavement conditions in real-time using smartphones and machine learning. Pervasive and Mobile Computing, 51, pp. 121–137, 2018. DOI: https://doi.org/10.1016/j.pmcj.2018.10.008

[8] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A., You only look once: unified, real-time object detection, Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., pp. 779–788, 2016. DOI: https://doi.org/10.1109/CVPR.2016.91.

[9] Ultralytics, Ultralytics YOLOv8. [online]. 2025 [consultation, May 09th of 2025]. Available at: https://docs.ultralytics.com/models/yolov8/

[10] Wang, C.Y., Yeh, I.H. and M.-Liao, H.Y., YOLOv9: learning what you want to learn using programmable gradient information [online]. 2025 [consultation May 09th of 2025]. Available at: https://arxiv.org/abs/2402.13616

[11] Wang, A., et., al., YOLOv10: real-time end-to-end object detection. advances in neural information processing systems, 37 (NeurIPS), pp. 1–28, 2024. DOI: https://doi.org/10.48550/arXiv.2405.14458

[12] Ultralytics, Ultralytics YOLOv11. [online]. 2025 [consultation May 09th of 2025]. Available at: https://docs.ultralytics.com/models/yolo11/

[13] Tian, Y., Ye, Q., and Doermann D., YOLOv12: attention-centric real-time object detectors, [online]. 2025 [consultation May 09th of 2025]. Available at: https://arxiv.org/abs/2502.12524

[14] Sobek, J., et., al., MedYOLO: a medical image object detection framework. Journal of Imaging Informatics in medicine, pp. 3208–3216, 2024. DOI: https://doi.org/10.1007/s10278-024-01138-2

[15] Liu, Y., et., al., SOCR-YOLO: small objects detection algorithm in medical images. International Journal of Imaging Systems and Technology, 34(4), 2024. DOI: https://doi.org/10.1002/ima.23130

[16] Ajayi, O.G., Ashi, J., and Guda, B., Performance evaluation of YOLO v5 model for automatic crop and weed classification on UAV images. Smart Agricultural Technology, 5(3), art. 100231, 2023. DOI: https://doi.org/10.1016/j.atech.2023.100231

[17] Lippi, M., Bonucci, N., Carpio, R.F., Contarini, M., Speranza, S., and Gasparri, A.A., YOLO-based pest detection system for precision agriculture. In 2021 29th Mediterranean Conference on Control and Automation, MED 2021, Bari, Puglia, Italy, pp. 342–347, 2021. DOI: https://doi.org/10.1109/MED51440.2021.9480344

[18] Omar, M., and Kumar, P., PD-ITS: pothole detection using YOLO variants for intelligent transport system. SN Computer Science, 5(5), 2024. DOI: https://doi.org/10.1007/s42979-024-02887-1

[19] Bučko, B., Lieskovská, E., Zábovská, K., and Zábovský, M. computer vision-based pothole detection under challenging conditions. Sensors, 22(22), pp. 1–18, 2022. DOI: https://doi.org/10.3390/s22228878

[20] Park, S.S., Tran, V.T., and Lee, D.E., Application of various yolo models for computer vision-based real-time pothole detection. Applied Sciences, 11(23), art. 1229, 2021. DOI: https://doi.org/10.3390/app112311229

[21] Wang, S., et., al., Measurement of asphalt pavement crack length using YOLO V5-BiFPN. Journal of Infrastructure Systems, 30(2), pp. 1–10, 2024. DOI: https://doi.org/10.1061/jitse4.iseng-2389

[22] Chen, D.R., and Chiu, W.M., Deep-learning-based road crack detection frameworks for dashcam-captured images under different illumination conditions. Soft Computing, 27(19), pp. 14337–14360, 2023. DOI: https://doi.org/10.1007/s00500-023-08738-0

[23] Yao, H., Liu, Y., Li, X., You, Z., Feng, Y., and Lu, W., A detection method for pavement cracks combining object detection and attention mechanism. IEEE Transactions on Intelligent Transportation Systems, 23(11), pp. 22179–22189, 2022. DOI: https://doi.org/10.1109/TITS.2022.3177210

[24] Souza, A.M., Oliveira, C.E., Decker, P.H.B., Amorim, G.E.R., Correa, A.L.S.C., and Fontenele, H.B., Defect detection using YOLOv8 for determining the condition of asphalt pavements. ALCONPAT, 15(1), pp. 79–91, 2025. DOI: https://doi.org/10.21041/ra.v15i1.781

[25] Yao, H., Fan, Y., Wei, X., Liu, Y., Cao, D., and You, Z. Research and optimization of YOLO-based method for automatic pavement defect detection. Electronic Research Archive, 32(3), pp. 1708–1730, 2024. DOI: https://doi.org/10.3934/ERA.2024078

[26] Du, Y., Pan, N., Xu, Z., Deng, F., Shen, Y., and Kang, H., Pavement distress detection and classification based on YOLO network. International Journal of Pavement Engineering, 22(13), pp. 1659–1672, 2021. DOI: https://doi.org/10.1080/10298436.2020.1714047

[27] Valipour, P.S., Golroo, A., Kheirati, A., Fahmani, M., and Amani, M.J., Automatic pavement distress severity detection using deep learning. Road Materials and Pavement Design, 25(8), pp. 1830–1846, 2023. DOI: https://doi.org/10.1080/14680629.2023.2276422

[28] Ganeshan, D., Sharif, M.S., and Apeagyei, A., road deterioration detection: a machine learning-based system for automated pavement crack identification and analysis. In 2023 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies, 3ICT 2023, 2023, pp. 188–194. DOI: https://doi.org/10.1109/3ICT60104.2023.10391802

[29] Peraka, N.S.P., Biligiri, K.P., and Kalidindi, S.N., Development of a multi-distress detection system for asphalt pavements: transfer learning-based approach. Transportation Research Record, 2675(10), pp. 538–553, 2021. DOI: https://doi.org/10.1177/03611981211012001

[30] Ferrari, E.C., Garcia, C., Júnior, C.A.P.D.S., and Fontenele, H.B., Classificação do defeito buraco a partir de modelos 3D, in 37o ANPET - Congresso de Pesquisa e Ensino em Transportes, Santos, São Paulo, Brasil, [online], pp. 1–12, 2023. Available at: https://proceedings.science/anpet-2023/trabalhos/classificacao-do-defeito-buraco-a-partir-de-modelos-3d?lang=pt-br

[31] Buslaev, A., Iglovikov, V.I., Khvedchenya, E., Parinov, A., Druzhinin, M., and Kalinin, A.A., Albumentations: fast and flexible image augmentations. Information, 11(2), pp. 1–20, 2020. DOI: https://doi.org/10.3390/info11020125

[32] Terven, J., Córdova-Esparza, D.M., and Romero-González, J.A.A, Comprehensive review of YOLO architectures in computer vision: from YOLOv1 to YOLOv8 and YOLO-NAS. Machine Learning and Knowledge Extraction, 5(4), pp. 1680–1716, 2023. DOI: https://doi.org/10.3390/make5040083

[33] Ranyal, E., Sadhu, A., and Jain, K., Automated pothole condition assessment in pavement using photogrammetry-assisted convolutional neural network. International Journal of Pavement Engineering, 24(1), art. 83401 , 2023. DOI: https://doi.org/10.1080/10298436.2023.2183401

[34] Ranyal, E., Sadhu, A., and Jain, K., Automated pothole condition assessment in pavement using photogrammetry-assisted convolutional neural network. International Journal of Pavement Engineering, 24(1), 83401, 2023. DOI: https://doi.org/10.1080/10298436.2023.2183401

[35] Ren, S., He, K., Girshick, R., and Sun, J., Faster R-CNN: towards real-time object detection with region proposal Networks. In Advances in Neural Information Processing Systems 28 (NIPS 2015), pp. 91–99, 2015. DOI: https://doi.org/10.1109/TPAMI.2016.2577031

[36] Lin, T.Y., Goyal, P., Girshick, R., He, K., and Dollar, P., Focal loss for dense object detection. In IEEE International Conference on Computer Vision (ICCV 2017), pp. 2980–2988, 2017. DOI: https://doi.org/10.1109/ICCV.2017.324

[37] Dong, J., et., al., CBAM-Optimized automatic segmentation and reconstruction system for monocular images with asphalt pavement potholes. IEEE Transactions on Intelligent Transportation Systems, 25(8), pp. 10313–10330, 2024. DOI: https://doi.org/10.1109/TITS.2024.3353257

Á.M. de Souza, received his BSc. Eng. in Civil Engineering from the State University of the Tocantins Region of Maranhão (UEMASUL) in 2023, where he participated in research, innovation, and outreach projects in the areas of pavements and geotechnics. He is currently pursuing a MSc. in Civil Engineering at the State University of Londrina (UEL) and is a scholarship holder from the Brazilian Coordination for the Improvement of Higher Education Personnel (CAPES). His research interests include artificial intelligence, with a focus on the application of deep learning and computer vision for the automatic detection of pavement distresses.
ORCID: 0000-0002-4328-5558

V.F. Cestari, received his BSc. Eng. in Civil Engineering from the State University of Londrina (UEL) in 2025. He currently works in the field of hydraulics at ENGITEQ, focusing on the development of water supply, sewage, and fire prevention projects.
ORCID: 0009-0006-8127-3802

H.B. Fontenele, received her BSc. Eng. in Civil Engineering from the University of the Amazon (UNAMA) in Belém in 1997, and obtained her Ph.D. in Sciences in the field of Transportation Infrastructure from the University of São Paulo (USP) at the São Carlos School of Engineering (EESC) in 2012. Since 2010, she has been a professor at the State University of Londrina (UEL). She is the author of two books, more than 50 scientific papers, and holds one registered software. Her research interests include Pavement Management Systems, the use of artificial intelligence focused on pavement management, pavement performance, and road environmental performance.
ORCID: 0000-0003-2046-0568