Open Journal of Civil Engineering, 2026, 16(1), 19-36
https://www.scirp.org/journal/ojce
ISSN Online: 2164-3172
ISSN Print: 2164-3164

Determination of the Pavement Surface Degradation Index Using the Instance Segmentation Method

Crespin Prudence Yabi[1]*, Gilchrist Legba[2], Gbènondé Sènan Gladys Milohin[2], Mohamed Gibigaye[2], Eric Alamou[1]

[1]Laboratory of Studies and Tests in Civil Engineering (L2EGC), National University of Sciences, Technologies, Engineering and Mathematics, Abomey, Benin
[2]Laboratory of Applied Energetic and Mechanic (LEMA), University of Abomey-Calavi, Abomey-Calavi, Benin
Email: *yabi.crespin@unstim.bj, milohingbenonde@gmail.com, ericalamou@gmail.com, gilchristlegba11@gmail.com, mohamed.gibigaye@uac.bj

How to cite this paper: Yabi, C.P., Legba, G., Milohin, G.S.G., Gibigaye, M. and Alamou, E. (2026) Determination of the Pavement Surface Degradation Index Using the Instance Segmentation Method. *Open Journal of Civil Engineering*, **16**, 19-36.
https://doi.org/10.4236/ojce.2026.161002

Received: December 9, 2025
Accepted: January 27, 2026
Published: January 30, 2026

Abstract

A nation's development depends in part on the quality of its road network. To this end, resources are being deployed to monitor the surface condition of our roads. Applying deep learning to road damage detection can significantly optimize roadway monitoring campaigns. According to the VIZIR method, roadway diagnosis cannot be performed without assessing the degradation index after monitoring. This work aims to develop an auscultation tool based on transfer learning, object tracking, and image processing to estimate the pavement deterioration index. To achieve this, the YOLOV11 instance segmentation and Roboflow instance segmentation 3.0 models were trained on five databases compiled from videos of degraded road surfaces taken on various roads in Benin. At the end of these various training sessions, the best model, named RIS3.5, obtained after training on the fifth database converted to grayscale and comprising 19 classes, had an accuracy of 95%, a mAP of 94.8%, and a recall of 90%. This model was then used to track objects in real time and enable the assessment of extent and SDI through a Python script every 5.5 m and then every 50 m, 200 m, and 500 m.

Keywords

Road Monitoring, Extent of Degradation, Artificial Intelligence, Image Processing, Surface Degradation Index, Transfer Learning, Models

1. Introduction

Populations are dependent on roads for access to employment, education, and healthcare. The importance is even greater for a developing country like Benin, where nearly 80% of freight traffic and 90% of passenger traffic are carried by road [1]. Unfortunately, roads deteriorate over time due to traffic, material fatigue, and water infiltration into the pavement. Construction errors or environmental conditions can also accelerate the deterioration process. If these types of damage (potholes, ruts, etc.) are not repaired, they worsen, reducing the quality of the road and affecting the comfort and safety of users. As stated by Madani *et al.* [2], "There are generally many causes of accidents, and it is not always the user and the vehicle that are responsible. In some cases, the road itself can also be the cause of very serious accidents if its surface is damaged". Kabir Boubacar Issa Baba of the PIARC association discusses the impact of damage on the economy in an article. He points out that a poorly maintained road network leads to increased transportation costs and food prices, which in turn lead to inflation [3]. Road maintenance is therefore of paramount importance for the safety and comfort of citizens, as well as for economic growth. It consists of a series of actions to be carried out on a road to maintain or restore its condition, ensuring comfortable and safe conditions for users. The maintenance work carried out is determined following inspection of the road surface and diagnosis.

Inspection involves locating damage and specifying its type, severity, and extent using a precise inspection method. Inspection leads to the estimation of the damage index used for diagnosis, which allows the calculation of reinforcements. The survey can be carried out in various ways. Manual surveys, which are traditional methods, are risky, time-consuming, and generate inconsistencies in spatial analyses [4]-[6]. Added to these disadvantages are the lack of reproducibility and the high costs of the campaigns carried out [7]. The software packages subsequently developed by SETRA, LCPC, and CEREMA (LCMS, DESY 2000, and VIZIROAD) have also failed to reduce costs and material, financial, and human resources. This does not allow us to envisage regular surveys of deterioration to ensure proper maintenance of the road network.

To address this, some authors have considered artificial intelligence and computer vision. Indeed, artificial intelligence and computer vision allow, among other things, to automate tasks, increase efficiency and productivity, and reduce human error. This technology could therefore lead to an improvement in the monitoring system.

Early work has primarily exploited image classification, an approach that predicted the presence or absence of a type of degradation in an image. Chen *et al.* designed a three-step model to detect the presence of cracks on pavements [8], while Pham *et al.* [9], focused on the distinction between degraded and non-degraded images. Yabi *et al.* developed models capable of classifying several types of damage [10].

To overcome these limitations, several works have opted for object detection, allowing a finer localization of pathologies. For example, Alqethami *et al.* [11], proposed a detection method based on convolutional neural networks (CNN) and a dataset collected on Saudi roads, comprising five steps: collection, preprocessing, and augmentation of data, implementation of different CNNs, experimentation, and evaluation.

Fan *et al.* [12], propose a method for automatically detecting problems related to road and bridge surfaces using fuzzy neural networks (FNN). Subsequently, Ayra *et al.* demonstrated the interest of transfer learning, to adapt pre-trained models to new geographical areas, which led to the adoption of models such as YOLO [13]. In this line, Zhang Y, *et al.*, proposed an improved version of YOLOv5 integrating NSPPF modules for crack detection [14]. Jeong *et al.* [15], applied the YOLO architecture to images captured via mobile phone, while Aghayan-Mashhady *et al.* [16], combined YOLO with data augmentation techniques, including adversarial methods, to improve road damage detection.

Sun *et al.* developed a detection model based on the YOLO model, into which they integrated a loss function for detecting road damage [17]. Ma *et al.* [18] present an improved YOLOv5 object detection algorithm that integrates dual-branch CA (Channel Attention) and SA (Spatial Attention) attention mechanisms as well as GIoU (Generalized Intersection over Union) loss. This approach has enhanced the model's detection accuracy and localization capabilities.

To further improve detection accuracy, some researchers have turned to instance segmentation, a technique for locating each occurrence of an object with high precision. This approach is particularly suitable for road monitoring, as it not only detects the presence of a pathology but also assesses its extent and morphology. Thus, some authors, Ha T, *et al.* [6], used deep learning segmentation models to quantify the severity of cracks [19]. Sandhya *et al.* developed a model capable of detecting potholes with high accuracy using the instance segmentation method based on the Yolov8m model architecture [20].

Furthermore, Kulumbayev *et al.* proposed a model based on the Mask R-CNN architecture for real-time detection, with high performance: precision of 0.9214, recall of 0.9876 and F1 score of 0.9571 [21].

Despite these advances, none of the studies mentioned are yet able to accurately segment road damage while estimating its severity and extent, which is necessary to determine the damage index according to the Vizir method. Furthermore, no approach to date incorporates real-time monitoring.

Thus, the present work aims to develop a model based on transfer learning, combining image segmentation, grayscale conversion, and object tracking algorithms, to improve the overall performance of the system. This model will automatically identify all types of degradation, assess their severity according to the steps defined by the Vizir method, and estimate the degradation index on any type of road axis. Ultimately, this approach should allow a more precise, faster and lower-cost assessment of the surface condition.

2. Research Methodology

The methodological strategy is based on four fundamental pillars: Vizir method, transfer learning, image processing by grayscale conversion, and object tracking.

Vizir method: Developed by the LCPC, the VIZIR method aims to qualify, quantify, and prioritize the condition of a roadway to establish appropriate maintenance strategies based on an SDI degradation index obtained by characterizing the type, severity (3 levels), and extent of each type of degradation observed on the section of roadway in question.

Transfer learning is an artificial intelligence technique that involves reusing a model that has already been trained on a given problem to solve another, similar or related problem.

Object tracking turns raw video into intelligent data about movement and behavior, powering many advanced applications in AI and automation.

Conversion to grayscale involves converting a color image into an image composed exclusively of shades of gray, ranging from absolute black to pure white. This is a fundamental preliminary step in image processing and computer vision that simplifies analysis by removing color information while retaining essential structural details such as contours, textures, and patterns.

These digital concepts were used to develop this tool in three phases: designing the segmentation model, implementing real-time monitoring, designing the program for calculating the degradation index.

2.1. Design of the Segmentation Model

The segmentation model was implemented in four stages: data collection, data pre-processing, database design, and model training.

2.1.1. Data Collection

- Equipment for data collection

The Canon Rebel T8i camera was used to obtain high-quality data. This camera can capture high-resolution videos. High-quality images with no noise are extracted from these videos. This camera offers the ability to adjust the image flow obtained from the videos and to have up to 60 frames per second with a resolution of 1920 × 1080.

The data come from videos taken on some asphalt-surfaced roads in Benin with low. The data were divided into data for model design and data for evaluating the degradation index calculation program. Table 1 presents the Structure of the collected data.

Table 1. Structure of the collected data.

Data Types	Data source
Data for model design	Abomey-Klouekanmè Sèmè-Porto-Novo Bohicon-Dassa
Data for the PCID program	Carrefour Zè-Zè Center Sèhouè-Massi

- Collection method

The collection carried out on the defined road axes initially consisted of a manual survey of the damage. In fact, for each degradation, parameters such as the following were measured in accordance with the Vizir method:

- length and width for calculating the surface area of the tears, such as peeling.
- the width of the lips for the cracks.
- the depth of the deformations.
- the diameter of the potholes.

The severity of each type of damage encountered was assessed using the Vizir method. For damage that did not require measurement, such as crazing, edge spalling, repairs, and other issues, a visual assessment of severity was made based on the criteria defined by the VIZIR method. The photographer then filmed the roadway, holding the camera at a height of 1.20 m and at an angle of 45˚.

2.1.2. Data Pre-Processing

The collected videos underwent a five-step pre-processing process: image extraction, selective sorting, image resizing, pixel normalization, and image annotation.

- Image extraction

Image extraction is the process of obtaining, according to a fixed step, these images that constitute the video. The extraction was possible using a script written in Python in the PyCharm environment.

- Selective sorting

To perform effectively, the model requires clear and varied images so that it can extract as many characteristics as possible. To achieve this, selective sorting was carried out to remove redundant (duplicate) or blurred images from the databases.

- Resizing images

Research has shown that an image that is too large requires more RAM than necessary, which slows down learning by neural networks.

The images in the databases were reduced by a quarter of their original size, from 1920 × 1080 pixels to 480 × 240 pixels. This resizing was done using a Python script called Resizing, written in PyCharm.

- Pixel Normalization

Pixel normalization, like resizing, simplifies data to reduce computation time for neural networks. In this work, min-max normalization was used thanks to a Python script in PyCharm called pixel normalization to perform pixel normalization.

- Image annotation

The model design through fine-tuning involves supervised learning. This requires prior identification of the images. Annotation was carried out based on the data collection summary table to indicate the damage present in each image and its severity. This step defines the model's performance because if there is no consistency between the annotations made, the model will have difficulty extracting the right characteristics to recognize the damage on its own. The annotation was carried out on the Roboflow platform using the Roboflow annotated feature. The polygon tool was used to accurately indicate the damage present in the images to the models, as bounding boxes do not allow us to accurately and distinctly capture our damage with irregular contours. Once the outline was drawn, the damage class and severity level were used to label damage considered type A by the VIZIR method, and only the damage class for damage considered type B. This is because type B damage is not used to calculate the SDI. Thus, when labeling, it was sufficient to indicate the class of degradation encountered after outlining. Figure 1 shows an example of an annotated image using the polygon tool.

Figure 1. Example of an annotated image using the polygon tool.

2.1.3. Creation of Databases

The third step involved creating various databases. This involved creating a folder containing the annotated images, which were structured into training, validation, and test datasets. Before generating the databases, data augmentation was performed. Data augmentation involves artificially increasing the size of the dataset by creating modified versions of our annotated images. This process reduces overfitting and enhances the model's generalization. Ultimately, the goal is to make the model more robust to potential variations and distortions that may be present in the input data. This includes changes in lighting, orientation, and scale.

After that, the databases were subdivided into three sets. One set, called the "training set", contains the images used to train the model. A second set, called the "validation test", contains the photos used to validate the model. The last set, called the "test set", serves as test data.

Once the subdivision was completed, the designed databases were exported in COCO segmentation (Common Objects in Context) and YOLO V11 formats. Five databases are created, and their compositions are presented in Table 2. Table 3 presents the classes from dataset 5.

- Converting databases to grayscale

The latest database was converted to grayscale using a Python script called Grayscale Conversion in the PyCharm IDE, which uses the CvtColor method with the BGR2 GRAY filter from the OpenCV library. This choice was made because conversion to grayscale allows for faster processing, which is crucial for real-time applications. It also simplifies the model's input data, allowing it to focus on shapes and textures rather than color variations, which are not criteria for distinguishing between degradation classes. The conversion does not affect the outlines drawn around the damage or the annotations made.

Table 2. Composition of the databases created.

	Number of images annotated	Number of images after Increase data	Number of classes	Size of the images	Data training	Data from validation	Data of test
Dataset 1	1831	6936	21	480 × 270	6085	552	299
Dataset 2	3465	3465	22	640 × 640	3037	290	138
Dataset 3	2695	2695	19	640 × 640	1899	514	282
Dataset 4	2695	4501	19	640 × 640	3151	900	450
Dataset 5	2695	4501	19	640 × 640	3151	900	450

Table 3. Presentation of classes from the latest database.

N˚	Damages	Number of annotations
1	Rut 3	1040
2	Ravelling	952
3	Upheaval 3	870
4	Traces of a track accident	672
5	Potholes	666
6	Stripping	407
7	Aligator Crack 2	401
8	Cross crack 1	390
9	Repair	384
10	Peeling 2	327
11	Longitudinal crack 2	323
12	Longitudinal crack 1	190
13	Cross crack 2	115
14	Aligator Crack 3	109
15	Rut 2	99
16	Aligator Crack 1	99
17	Rut 1	48
18	Edge break	42
19	Longitudinal crack 3	26
	Total	7160

2.1.4. Training of YOLOV11 Instance Segmentation and ROBOFLOW3.0 Instance Segmentation Models

Once the various databases had been designed, they were used to train the different transfer learning models: YOLOV11 instance segmentation and the Roboflow 3.0 Instance Segmentation model. YOLOV11 and Roboflow 3.0 models were selected because of their optimal balance between speed and accuracy, as well as their strong performance in transfer learning. These characteristics make them particularly well-suited for road environments, where scenes must be analyzed quickly and with a high level of reliability. We chose these models because they have a very large community, comprehensive tutorials, are easier to implement, and are available on the Roboflow platform, which we used for annotation and database creation.

The training took place on the Roboflow platform. At the end of each training session, the performance of the models was evaluated using the following metrics:

- Precision.
- mAP (mean Average Precision).
- Recall.
- Confusion matrix.

A confidence threshold of 70% was set on the platform.

2.2. Object Tracking

The weight of the model recovered after training the latest database was inserted into the code for object tracking using one of the videos from the data collection on the Zè-Zè-centre intersection. Thanks to the weight of the model, the object tracking algorithm will be able to determine the damage present in the videos.

To track damage, the ByteTrack algorithm, renowned for its accuracy and reliability, and the Python Supervision library, which provides connectors to models such as YOLO, were used. Thanks to this Python library, which offers a range of annotators and numerous highly customizable utilities, it is possible to view, load, split, merge, and save our datasets in different formats.

The Python script identifies all damage present on the filmed road surface and then draws a boundary polygon around the detected problems. It then assigns each defect an identifier called an "Id-tracker" and flags their presence when the camera passes over them. The code ultimately produces a video showing how the damage was tracked within the video.

2.3. Design of the Degradation Index Calculation Program

2.3.1. Estimation of the Extent and Severity of Damage

The method used to estimate the extent consists, first, of retrieving the length, width, and coordinates of the center of the bounding boxes predicted by the model, as well as the identifier (id) of the predicted damage. This information is retrieved following the detections made by the model.

Then, based on the parameters set during field collection, namely:

- a constant collection speed of 10 km/h;

- a camera viewing angle fixed at 45˚;
- a shooting height of 1.20 m, so that the filmed video covers the entire width of the road.

The number *nif* of image frames from which one leaves the visual field of an image could be estimated.

Field measurements have shown that the width of an image taken while respecting the previously mentioned parameters occupies = 5.5 m of the roadway length. This information, combined with the speed set at 20 km/h and the camera frame rate set at 60 frames per second, was used to find this number *nif* = 119 image frames.

This information was used to calculate the extent of images spaced 119 frames apart. This was done using a Python script called calcul_étendue, written in PyCharm, which extracts images with a step size of 119 frames and then uses the detection model to make predictions on these images. Following the predictions, for the damage detected, the code retrieves the coordinates of the bounding box to calculate the areas occupied by each of the areas of damage present in the image. It then calculates the ratio between this area and that of the image, which is (1920 × 1080). This ratio, thus calculated, is the extent of the degradation calculated by the formula:

$$E = \frac{S_1}{S_T} \times 100$$

with S_T the total area of an image in $pixel^2$ (1080 × 1920 in our case).

2.3.2. Estimation of the Severity

During annotation, type A damage was annotated in the format “Damage name” + “Severity level”. This resulted in classes such as Ruts 3, indicating that the damage detected was a rut with a severity level of 3.

2.3.3. Evaluation of the Degradation Index

The degradation index, which characterizes the surface condition of our roads, is obtained by combining the extent and severity of the degradation. The method used until now has allowed the detection of the degradations with their level of severity and to estimate their extent. The SDI is obtained by combining the cracking index obtained from the extent and severity of cracks and alligator cracks, and the deformation index Id obtained by agglomerating the extent and severity of the deformations (Rut, Bead, Subsidence, Wane, etc.). The Vizir method presents tables and calculation principles for various indices (cracking index, deformation index, and surface degradation index).

The code then estimates the cracking index by combining the severity of longitudinal and transverse cracks and crazing with their extent, as indicated in the VIZIR method and presented in **Table 4**. The same was done for deformations. The maximum cracking index obtained on the 5.5 m sections is the cracking index for the 200 m section. The same applies to the deformation index. The values obtained for the cracking and deformation indices of the 200 m section are combined, as shown in **Table 5**, to calculate the deterioration index. If repairs were detected in the video, the SDI index was corrected in accordance with the VIZIR method specifications specified in Table 6.

Table 4. Methodology for evaluating crack and deformation indices using the Vizir method.

	Extent Severity	0 - 10%	10 - 50%	>50%
Cracking index If and deformation index Id	1	1	2	3
	2	2	3	4
	3	3	4	5

Table 5. Methodology for assessing the degradation index according to the VIZIR method.

	If Id	0	1 - 2	3	4 - 5
First value of the SDI index in the absence of repair	0	1	2	3	4
	1-2	3	3	4	5
	3	4	5	5	6
	4-5	5	6	7	7

Table 6. Evaluation of the degradation index considering repairs according to the Vizir method.

	Extent Severity	0 - 10%	10 - 50%	>50%
Correction due to repairs	1	0	0	0
	2	0	0	+1
	3	0	+1	+1

Once the code was complete, 200-meter video sequences from the data collection carried out on the Crossroad Zè-Zè Centre road and the Sèhouè-Massi section were used to evaluate its performance in PyCharm. Subsequently, thanks to the manual survey carried out during the collection, the index was calculated manually and compared with the results from the various videos to improve the program's performance. After testing, the code was deployed within a mobile application.

3. Results and Discussion

3.1. Performance of the Models Following the Different Training Sessions

Figure 2 and Figure 3 show the mAp50 score obtained from validation data for models designed before and after switching to 19 classes, respectively. The performance of the models designed has gradually improved. Initially, marks appearing on the road surface as cracks were annotated and assigned to the traces of track accident class. These marks were the main sources of background errors. In addition, using various confusion matrices, other sources of background errors related to blurred images and annotation errors were corrected. The preprocessing process, in particular pixel normalization, reduced the model's sensitivity to variations in light intensity (sun, shade).

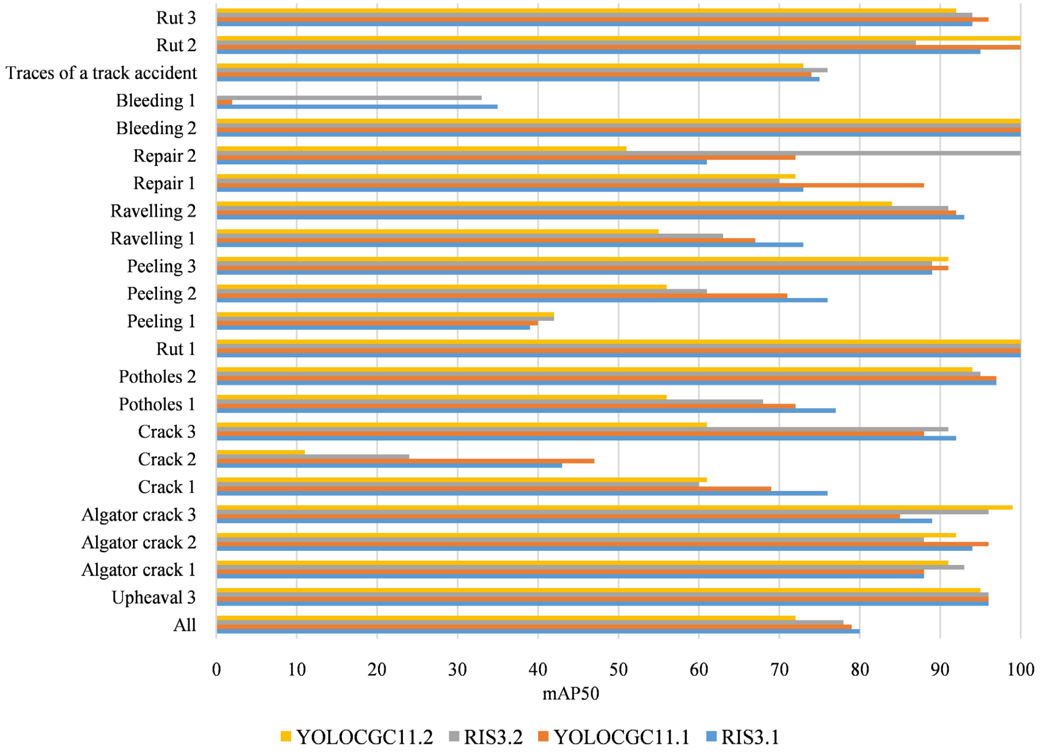


Figure 2. mAP50 scores obtained from validation data for models designed before switching to 19 classes.

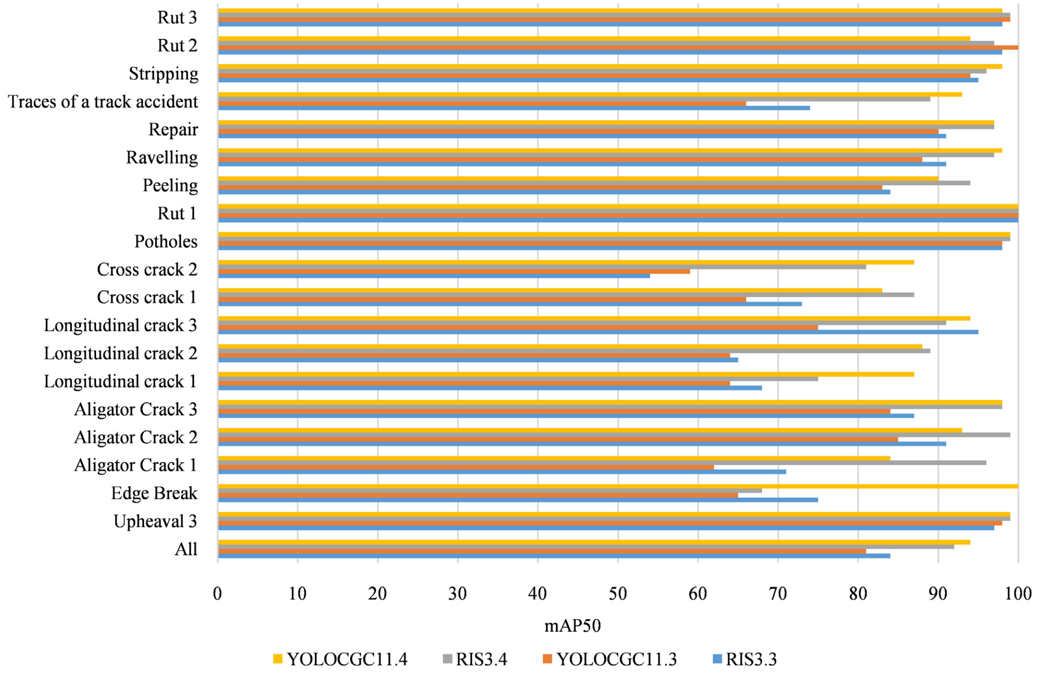


Figure 3. mAP50 values obtained from validation data for models designed after switching to 19 classes.

Furthermore, the number of classes has been reduced in the latest databases because pathologies in the class of material up lift or displacement are not useful for calculating the degradation index. As a result, the classes created to indicate their severity levels have been merged. For example, the Potholes 1 and Potholes 2 classes have been merged into the Potholes class. It is noticed in these Figures that the combination of some classes increases the mAP50.

Figure 4 shows that the mAP50 precision increased for RIS3.4 and RIS3.5. The comparison of the results in **Figure 4**, shows that the grayscale conversion improved the model's performance, resulting in a 2% improvement in accuracy, a 2.8% improvement in mAP, and a 0.2% improvement in Recall.

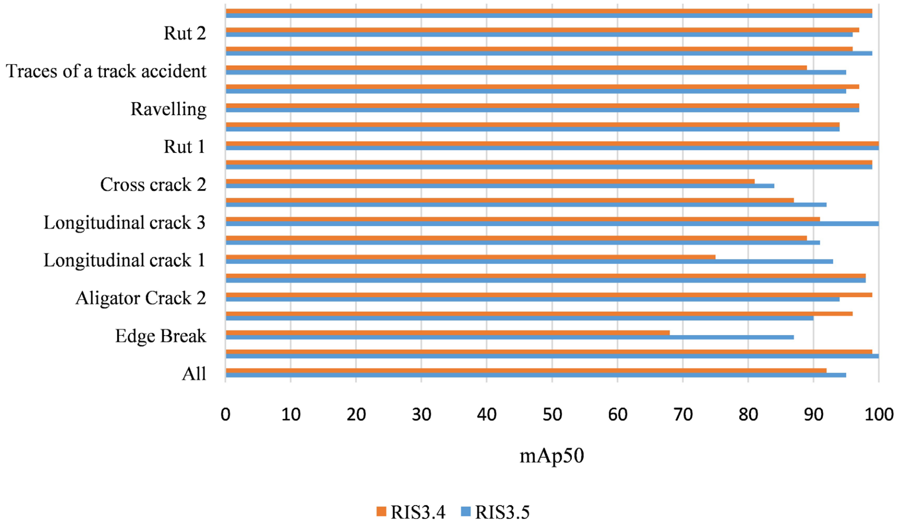


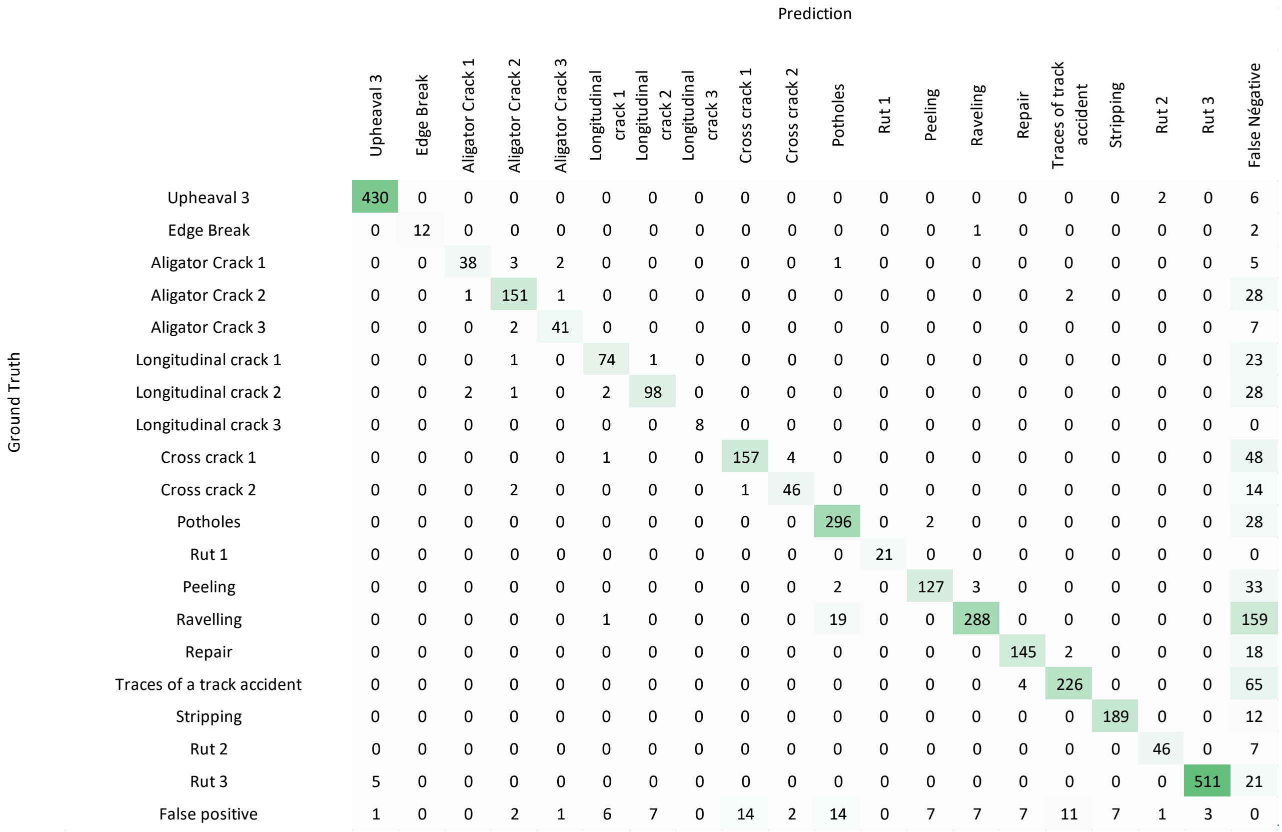
Figure 4. Evaluation of the impact of grayscale conversion on mAP50 for each degradation class.

Table 7 summarizes the Precision, the mAP50, and the recall of all tested models. Analysis of **Table 7** shows that the best model obtained, named RIS 3.5, was achieved after training the Roboflow instance segmentation model on the fifth database converted to grayscale.

Table 7. Summary of the performance of the models designed.

Database	Precision (%)	Map (%)	Recall (%)
RIS3.1	79.9	79.7	77.2
YOLOCGC11.1	79.4	78.6	74.7
RIS3.2	82.3	78.1	68.
YOLOCGC11.2	75.5	71.6	65
RIS3.3	83.9	84.3	78.6
YOLOCGC11.3	79.8	80.9	78.5
RIS3.4	93	92	89.8
YOLOCGC11.4	93.1	93.6	91.1
RIS3.5	95	94.8	90

The confusion matrix of the model RIS3.5 is presented in Figure 5. This confusion matrix indicates that there is low precision in predicting some types of degradation, such as raveling.



Prediction

Ground Truth

	Upheaval 3	Edge Break	Aligator Crack 1	Aligator Crack 2	Aligator Crack 3	Longitudinal crack 1	Longitudinal crack 2	Longitudinal crack 3	Cross crack 1	Cross crack 2	Potholes	Rut 1	Peeling	Raveling	Repair	Traces of track accident	Stripping	Rut 2	Rut 3	False Négative
Upheaval 3	430	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	6
Edge Break	0	12	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	2
Aligator Crack 1	0	0	38	3	2	0	0	0	0	0	1	0	0	0	0	0	0	0	0	5
Aligator Crack 2	0	0	1	151	1	0	0	0	0	0	0	0	0	0	0	2	0	0	0	28
Aligator Crack 3	0	0	0	2	41	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7
Longitudinal crack 1	0	0	0	1	0	74	1	0	0	0	0	0	0	0	0	0	0	0	0	23
Longitudinal crack 2	0	0	2	1	0	2	98	0	0	0	0	0	0	0	0	0	0	0	0	28
Longitudinal crack 3	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	0
Cross crack 1	0	0	0	0	0	1	0	0	157	4	0	0	0	0	0	0	0	0	0	48
Cross crack 2	0	0	0	2	0	0	0	0	1	46	0	0	0	0	0	0	0	0	0	14
Potholes	0	0	0	0	0	0	0	0	0	0	296	0	2	0	0	0	0	0	0	28
Rut 1	0	0	0	0	0	0	0	0	0	0	0	21	0	0	0	0	0	0	0	0
Peeling	0	0	0	0	0	0	0	0	0	0	2	0	127	3	0	0	0	0	0	33
Ravelling	0	0	0	0	0	1	0	0	0	0	19	0	0	288	0	0	0	0	0	159
Repair	0	0	0	0	0	0	0	0	0	0	0	0	0	0	145	2	0	0	0	18
Traces of a track accident	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	226	0	0	0	65
Stripping	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	189	0	0	12
Rut 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	46	0	7
Rut 3	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	511	21
False positive	1	0	0	2	1	6	7	0	14	2	14	0	7	7	7	11	7	1	3	0

Figure 5. Confusion matrix of the best model RIS3.5.

3.2. Comparison of the Performance of Our Models with Those Found in the Literature

Many authors have also tried other approaches to detecting pavement deterioration. Table 8 compares the developed model with those found in the literature. The number of pathology classes, methods, and evaluation metrics varies from one author to another. A comparison was made between the results obtained by the designed model and those found in the literature to assess its relevance. This comparison is presented in the following table.

The results presented in Table 8 show that the model RIS3.5 has good metrics compared to other models. It stands out for its good balance between accuracy, mAP, and recall. In addition, between the developed model is the only one that detects 19 classes of damage.

3.3. Evaluation of the Degradation Index with the Collection Evaluation Data

The surface degradation index calculation program was tested on the videos collected. A manual calculation of the index was also performed to verify the reliability of the program. The results obtained are presented in Table 9 and Table 10 for two sections of the road.

Table 8. Comparison of the model with those found in the literature.

Models	Authors & Years	Precision	mAP50	Recall	Classes
RIS 3.6	**Ours** 2025	0.95	0.948	0.90	19 classes
ResNet 50	Deru LI *et al.* 2023 [22]	0.76509	0.90257	0.86013	Longitudinal cracks, Transverse cracks, Block cracks, Alligator cracking, Patch, Potholes
YOLOv8	Samadzadegan *et al.* 2024 [23]	0.77	0.79	0.75	Transverse cracks, Longitudinal cracks, Alligator cracking, Diagonal/oblique cracks, Potholes, Patches, and Delamination
LTPLN's	W. Huang *et al.* 2024 [24]	0.935		0.935	Longitudinal cracks, Alligator cracking, Potholes and others
SMG-YOLOv8	S. Zhang *et al.* 2024 [25]	0.805	0.794	0.862	Longitudinal cracks, Transverse cracks, Alligator cracking, Potholes
PDDnet	Zhong *et al.* 2024 [26]	0.84	0.795	0.80	Longitudinal cracks, Transverse cracks, Diagonal/oblique cracks, Alligator cracking and Potholes, Sealed cracks, Patches
ViT	Yao Zhang *et al.* 2024 [14]	0.937	0.939	0.942	Longitudinal crack, Alligator cracking, Pothole, Non-deteriorated pavement
YOLOv8-MHSA-TA	Z.F. Elsharkawy *et al.* 2025 [27]	0.889	0.775%	0.848	Cracks (Crack500 dataset)

Table 9. Degradation index found manually and with the model on the Carrefour Zè-Zè Centre axis.

	Degradation index found manually	Degradation index found with the model
Portion cracking index	0	0
Portion deformation index	0	0
Degradation index SDI	1	1
SDI corrected	1	1

Table 10. Degradation index found manually and with the model on the Sèhouè-Massi axis.

	Degradation index found manually	Degradation index found with the model
Portion cracking index	3	3
Portion deformation index	2	3
Degradation index SDI	4	5
SDI corrected	5	5

3.3.1. Section on the Carrefour Zè-Zè Centre Axis

Table 9 and Table 10 demonstrate that the model accurately identified the damage present on the filmed road surface. There is a similarity between the predictions made by the model and those obtained through the manual survey conducted in the field. The SDI obtained by manual calculation is equal to one, given that the damage recorded on this section of road is mostly rutting. The SDI obtained by aggregating the SDI determined per image (5.5 m) with this program matches that determined manually. The program, therefore, also performed well in this respect.

3.3.2. Second Portion of 200 m (Sèhouè-Massi Axis)

In this case, the model detected cracks, tears, and some deformations. The detections made and the severity levels found by the model correspond to those observed in the field. This demonstrates that the model is working properly. Next, using various Python codes, the extent of the multiple types of damage detected and the damage index per image were evaluated. After agglomeration, the SDI was evaluated at 200 m intervals. The program found a damage index of five before and after corrections, while the manual calculation gave an index of four before correction and five after correction. This discrepancy is because the model detects repairs and already incorporates the related corrections when calculating the SDI for each image. As a result, at the model level, the SDI before correction is identical to that after correction.

For the two sections of road studied, the model closely matching the detections made in the field, accurately detected the various pathologies. The program also accurately predicted the SDI. These positive results demonstrate the reliability of this approach.

3.3.3. Model Limitations and Application Constraints

The designed model has certain limitations that should be highlighted. The model does not allow all types of damage to be identified. Certain types of damage, such as road surface deflection or road subsidence, cannot be detected by the model because they were not recorded during the data collection process. The accuracy of detection remains sensitive to adverse weather conditions such as rain, humidity, or the presence of standing water on the road surface. Furthermore, changing the camera or shooting angle would affect the model's performance.

4. Conclusion

This work aimed to make it easier to inspect road surfaces. A digital tool was developed that allows users to detect all damage present on a section of paved road, along with its severity, and then assess its extent and the surface degradation index SDI. To do this, data was first collected on roads in Benin. Next, using Python scripts developed in the PyCharm IDE, the data was pre-processed using a five-step process: extraction of images from videos (extract), resizing of the images obtained (resizing), pixel normalization (pixel normalization), and annotation in the Roboflow environment. Once annotated, the images were used to build various databases. The images in the fifth database were then converted to grayscale using a Python script (Grayscale conversion). Once the databases were created, they were used for transfer learning to train instance segmentation models (YOLOV11 instance segmentation and Roboflow instance segmentation). Different performances were obtained following training on the Roboflow and Google Colab platforms. The best model, named RIS 3.5, obtained after training on the fifth database comprising 19 classes, had an accuracy of 95%, a mAP of 94.8%, and a recall of 90%. This model was then integrated into a Python code named calcul_SDI to determine the extent and then the surface degradation index SDI every 5.5 m, aggregated at 50 m, 200 m, and 500 m intervals set by the VIZIR method. This method was tested on videos collected at the Zè crossroad-Zè center and Sèhouè-Massi road section. The results obtained for the SDI calculation are identical to those found during the manual survey. This ensured the reliability of the various programs.

Data Availability

The five datasets used here are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Yabi, C.P. (2018) Modélisation d'une plaque en sous revêtement d'une route sur sol élastique et inerte soumise à des charges dynamiques. Ph.D. Thesis, Université d'Abomey Calavi.

[2] Madani, B. and Mohamed, M.K. (2010) Les routes et leurs défauts d'etude, de conception et d'entretien pouvant causer des accidents de circulation-recommandations. *Séminaire national de la route et la sûreté de la circulation*, 29-30 Novembre 2010, Ouargla, 1-6.

[3] Baba, K.B.I. (2018) L'importance des routes dans le développement d'un pays comme le Niger. Fondation Hirondelle.

[4] Laboratoire Centrale des Ponts et Chaussée (1997) Relevé des dégradations des surfaces de chaussées (Méthode d'essai 38-2). Cedex.

[5] Fassmeyer, F., Chang, C. and Wang, J. (2013) Towards a Camera-Based Road Damage Assessment and Detection for Autonomous Vehicles. *Journal of Autonomous Systems and Applications*, **4**, 78-88.

[6] Ha, T., Park, K. and Jeon, S. (2022) Deep Learning-Based Road Damage Detection Using Thermal Images. *International Journal of Civil Engineering*, **13**, 111-121.

[7] Cheng, H.D., Wang, J., Hu, Y.G., Glazier, C., Shi, X.J. and Chen, X.W. (2001) Novel Approach to Pavement Cracking Detection Based on Neural Network. *Transportation Research Record: Journal of the Transportation Research Board*, **1764**, 119-127. https://doi.org/10.3141/1764-13

[8] Chen, G., Teng, S., Lin, M., Yang, X. and Sun, X. (2022) Crack Detection Based on Generative Adversarial Networks and Deep Learning. *KSCE Journal of Civil Engineering*, **26**, 1803-1816. https://doi.org/10.1007/s12205-022-0518-2

[9] Pham, V., Nguyen, D. and Donan, C. (2022) Road Damage Detection and Classification with YOLOv7. 2022 *IEEE International Conference on Big Data* (*Big Data*), Osaka, 17-20 December 2022, 6416-6423. https://doi.org/10.1109/bigdata55660.2022.10020856

[10] Yabi, C.P., Gbehoun, G.F., Tamou, B.C.K., Alamou, E., Gibigaye, M. and Farsangi, E.N. (2025) Analysis and Classification of Distress on Flexible Pavements Using Convolutional Neural Networks: A Case Study in Benin Republic. *Infrastructures*, **10**, Article 111. https://doi.org/10.3390/infrastructures10050111

[11] Alqethami, S., Alghamdi, S., Alsubait, T. and Alhakami, H. (2022) RoadNet: Efficient Model to Detect and Classify Road Damages. *Applied Sciences*, **12**, Article 11529. https://doi.org/10.3390/app122211529

[12] Fan, X. and Liu, R. (2024) Automatic Detection and Classification of Surface Diseases on Roads and Bridges Using Fuzzy Neural Networks. 2024 *International Conference on Electrical Drives*, *Power Electronics & Engineering* (*EDPEE*), Athens, 27-29 February 2024, 770-775. https://doi.org/10.1109/edpee61724.2024.00148

[13] Arya, D., Maeda, H., Ghosh, S.K., Toshniwal, D., Mraz, A., Kashiyama, T., *et al.* (2021) Deep Learning-Based Road Damage Detection and Classification for Multiple Countries. *Automation in Construction*, **132**, Article ID: 103935. https://doi.org/10.1016/j.autcon.2021.103935

[14] Zhang, Y., Chen, J., Wu, Z., Guo, X. and Jia, S. (2025) Optimizing Pavement Distress Detection with UAV: A Comparative Study of Vision Transformer and Convolutional Neural Networks. *KSCE Journal of Civil Engineering*, **29**, Article ID: 100095. https://doi.org/10.1016/j.kscej.2024.100095

[15] Jeong, D.J. (2020) Road Damage Detection Using YOLO with Smartphone Images. https://www.researchgate.net/publication/346787751_Road_Damage_Detection_Using_YOLO_with_Smartphone_Images

[16] Aghayan-Mashhady, N. and Amirkhani, A. (2023) Road Damage Detection with Bounding Box and Generative Adversarial Networks Based Augmentation Methods. *IET Image Processing*, **18**, 154-174. https://doi.org/10.1049/ipr2.12940

[17] Sun, S. and Wang, B. (2025) Detection of Highway Defects Based on Image Recognition Using Unmanned Aerial Vehicles. 2025 *IEEE* 12*th Joint International Information Technology and Artificial Intelligence Conference* (*ITAIC*), Chongqing, 23-25 May 2025, 957-962. https://doi.org/10.1109/itaic64559.2025.11163122

[18] Ma, L. and Chen, M. (2025) Road Damage Detection Based on Improved YOLO Algorithm. *Scientific Reports*, **15**, Article No. 28506. https://doi.org/10.1038/s41598-025-14461-7

[19] Zhao, M., Xu, X., Bao, X., Chen, X. and Yang, H. (2024) An Automated Instance Segmentation Method for Crack Detection Integrated with Crackmover Data Augmentation. *Sensors*, **24**, Article 446. https://doi.org/10.3390/s24020446

[20] Sandhya Devi, R.S., Jeni Santina, A., Swathi, S. and Tamilsel, V.S.K. (2024) Pothole Detection and Instance Segmentation Using YOLO V8. 2024 *International Conference on IoT Based Control Networks and Intelligent Systems* (*ICICNIS*), Bengaluru, 17-18 December 2024, 1185-1190. https://doi.org/10.1109/icicnis64247.2024.10823139

[21] Kulambayev, B. (2022) A Deep Learning-Based Approach for Road Surface Damage Detection. *Computers*, *Materials & Continua*, **73**, 3403-3418. https://doi.org/10.32604/cmc.2022.029544

[22] Li, D., Duan, Z., Hu, X., Zhang, D. and Zhang, Y. (2023) Automated Classification and Detection of Multiple Pavement Distress Images Based on Deep Learning. *Journal of Traffic and Transportation Engineering* (*English Edition*), **10**, 276-290. https://doi.org/10.1016/j.jtte.2021.04.008

[23] Samadzadegan, F., Dadrass Javan, F., Ashtari Mahini, F., Gholamshahi, M. and Nex, F. (2024) Automatic Road Pavement Distress Recognition Using Deep Learning Networks from Unmanned Aerial Imagery. *Drones*, **8**, Article 244. https://doi.org/10.3390/drones8060244

[24] Huang, W., Feng, L. and He, Y. (2024) LTPLN: Automatic pavement distress detection. *PLOS ONE*, **19**, e0309172. https://doi.org/10.1371/journal.pone.0309172

[25] Zhang, S., Bei, Z., Ling, T., Chen, Q. and Zhang, L. (2024) Research on High-Precision Recognition Model for Multi-Scene Asphalt Pavement Distresses Based on Deep Learning. *Scientific Reports*, **14**, Article No. 25416. https://doi.org/10.1038/s41598-024-77173-4

[26] Zhong, J., Zhang, M., Ma, Y., Xiao, R., Cheng, G. and Huang, B. (2024) A Multitask Fusion Network for Region-Level and Pixel-Level Pavement Distress Detection. *Journal of Transportation Engineering*, *Part B*: *Pavements*, **150**, Article ID: 04024002. https://doi.org/10.1061/jpeodx.pveng-1433

[27] Elsharkawy, Z.F., Kasban, H. and Abbass, M.Y. (2025) Efficient Surface Crack Segmentation for Industrial and Civil Applications Based on an Enhanced Yolov8 Model. *Journal of Big Data*, **12**, Article No. 16. https://doi.org/10.1186/s40537-025-01065-1