TRANSPORTES
revistatransportes.org.br
anpet
associação nacional de pesquisa e ensino em transportes
anpet.org.br

Evaluation of automated detection of pavement defects using YOLOv3: impact of collection techniques

Avaliação da detecção automatizada de defeitos em pavimentos com YOLOv3: impacto das técnicas de coleta

Gabriel Tavares de Melo Freitas[1], **Ernesto Ferreira Nobre Júnior**[2], **Aline Calheiros Espindola**[3]

[1]Instituto Federal do Ceará, Fortaleza, CE, Brasil
[2]Universidade Federal do Ceará, Fortaleza, CE, Brasil
[3]Universidade Federal de Alagoas, Maceió, AL, Brasil

Contact: gabriel.tavares@ifce.edu.br, (GTMF); nobre@ufc.br, (EFNJ); aline.espindola@ctec.ufal.br, (ACE)

Submitted:
20 June, 2022
Revised:
23 January, 2024
Accepted for publication:
21 March, 2024
Published:
23 May, 2024
Associate Editor:
Kamilla Vasconcelos, Universidade de São Paulo, Brasil

Keywords:
Pavement.
Data collection.
Deep learning.

Palavras-chave:
Pavimento.
Coleta de dados.
Aprendizado profundo.

DOI: 10.58922/transportes.v32i2.2796

ABSTRACT
This study involved training six neural networks with tailored configurations to automatically detect problems in pavements, utilizing the YOLOv3 framework. The acquisition of images and videos depicting pavement defects was conducted using smartphones and action cameras, leading to the organization of six distinct datasets. Every neural network was subjected to training and validation with the goal of attaining optimal accuracy in automated object detection. Implementing YOLOv3 facilitated effective defect surveys, enhancing the assessment of pavement quality, and offering valuable information for decision-making in road transport management. Upon concluding the investigation, it was determined that the framing method with the highest efficacy attained a precision rate of 98%. The results demonstrate the efficacy of YOLOv3 in accurately detecting defects, underscoring the significance of data collecting and framing methods, and adding to the current body of knowledge on automated pavement defect detection.

RESUMO
Este estudo envolveu o treinamento de seis redes neurais com configurações personalizadas para detectar automaticamente defeitos nos pavimentos, utilizando o *framework* YOLOv3. A aquisição de imagens e vídeos retratando defeitos do pavimento foi realizada utilizando *smartphones* e câmeras de ação, levando à organização de seis *datasets* distintos. Cada rede neural foi submetida a treinamento e validação com o objetivo de atingir a precisão ideal na detecção automatizada de objetos. A aplicação do YOLOv3 possibilitou a realização eficiente de levantamentos de defeitos, contribuindo para o diagnóstico da qualidade do pavimento e fornecendo subsídios para a tomada de decisão na gestão dos transportes rodoviários. Ao final da análise, constatou-se que o método de enquadramento com maior eficácia atingiu uma taxa de precisão de 98%. Os resultados demonstram a eficácia do YOLOv3 na identificação dos defeitos, ressaltando a importância das técnicas de coleta e enquadramento e contribuindo para aumentando do conhecimento existente sobre detecção automatizada de defeitos em pavimentos.

1. INTRODUCTION

Ensuring the longevity of road pavements is of utmost importance. Therefore, it is essential to actively participate in the practices of preservation and rehabilitation at the most opportune moment to uphold their value as significant resources. Precise and reliable data regarding the condition of highways is crucial. Postponed maintenance of the road infrastructure can result in early aging and potentially trigger an irreversible process of deterioration (Paterson, 1987).

The objective of pavement assessment methods is to restore the comfort and safety of users. The Pavement Management System (PMS) has three key stages: data collection, data analysis, and

maintenance planning. The initial and important stage is collecting data, which entails assessing the present state of the pavement by either manual or automated techniques (Sholevar, Golroo and Esfahani, 2022). During the process of manual evaluation, examiners identify problems by considering their inherent characteristics, extent, and severity. Automated evaluation entails the deployment of cameras or sensors on vehicles to acquire visual data, such as images or films, of the state of the pavement. These recordings are subsequently analyzed in the office for assessment purposes (Balbo, 2007).

In order to ensure the proper maintenance of road infrastructure, it is crucial to accurately identify pavement defects. This study investigates the influence of data collection on the efficacy of the YOLOv3 framework in detecting pavement defects, acknowledging the significance of this stage in ensuring the reliability of the process. This differs from conventional approaches, which primarily focus on optimizing the hyperparameters of the neural network.

The research aims to clarify the impact of variables in data collection approaches, such as the diversity of devices employed and strategic positioning, on the accuracy of the system when utilizing the YOLOv3 detection framework.

This research not only adds to the current literature on automatic defect detection, but also presents a novel viewpoint that emphasizes the importance of the data collecting phase in improving the accuracy of identifying defects in YOLOv3 pavement.

2. LITERATURE REVIEW

This discussion will concisely cover Artificial Neural Networks (ANNs), YOLOv3, and studies related to the application of computer vision in identifying defects in road pavements.

2.1. Artificial Intelligence (AI)

Artificial Neural Networks (ANNs) are created to interpret information, much like the intricate nature of the human brain (Haykin, 1998). In the field of computer vision, artificial neural networks are utilized by computers to deduce labels from digital inputs, as elucidated by Khan and Al-Habsi (2020) and Prince (2012). These machines draw inspiration from the human ability to recognize patterns in images.

Object detection models have the ability to accurately locate an object inside an image and also assess its existence and classification. The essential components of these models comprise a feature extractor, proposed region, and a classification module. Object detection architectures like YOLO, R-CNNs, and SSDs are popular due to their speed, user-friendliness, and ability to locate certain objects in images. Object detection models are highly advantageous in scenarios with extensive and intricate images as they provide multiple benefits for accurately identifying the position and quantity of objects (Sholevar, Golroo and Esfahani, 2022).

YOLO (You Only Look Once) is a Convolutional Neural Network (CNN) that operates in real-time and is designed to identify objects within images. The YOLO network may be utilized to effectively track, locate, and classify objects (Radovic et al., 2017). The third version of YOLO demonstrates impressive real-time processing velocity, achieving a rate of 45 frames per second on a Graphics Processing Unit (GPU) by employing the ResNet-50 architecture Convolutional Neural Network (CNN). This model employs a single-pass approach to conduct object detection and classification in an image, utilizing the object detection method (Redmon et al., 2016). The YOLOv3 detection

technique involves resizing the input image, submitting it to the CNN, and finally performing the detection.

2.2. YOLOv3 Metrics

YOLOv3 utilizes metrics to evaluate the model's quality, which are computed using the dataset. These metrics include Precision, Recall, F1-score, Intersection over Union (IoU), Average Precision (AP per class), and mean Average Precision (mAP). It is important to mention that in order to calculate Precision and Recall, one needs to have the values for true positives, false positives, and false negatives are required.

Precision is the ability of a model to identify only relevant objects (Padilla, Netto and Silva., 2020). The formula for calculating Precision is presented in Equation 1.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \tag{1}$$

Recall informs the model's ability to find all relevant cases (Padilla, Netto and Silva, 2020). The formula for calculating Recall is presented in Equation 2.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{2}$$

The F1-score is used to determine the ideal confidence that balances the values of Precision and Recall for the model (Sasaki, 2007). The formula for calculating the F1-score is presented in Equation 3.

$$F_1 = 2\,x\frac{\left(Precision\,x\,Recall\right)}{\left(Precision + Recall\right)} \tag{3}$$

IoU is used to measure how much the detected object's quadrant overlaps with the object demarcated in the labeling phase (Everingham et al., 2010). The formula for calculating IoU is presented in Equation 4.

$$IoU = \frac{Area\ Overlap}{Area\ of\ Union} \tag{4}$$

AP (Average Precision) is a popular metric in measuring YOLOv3's precision, where the average precision value is calculated for the Recall value above between 0 and 1 (Everingham et al., 2010). The formula for calculating AP is presented in Equation 5.

$$AP = \int_{r=0}^{1} p\left(r\right)dr \tag{5}$$

mAP (mean Average Precision) compares the objects demarcated in the labeling phase with the objects detected by the model and returns a score. The formula for calculating mAP is presented in Equation 6.

$$mAP = \frac{1}{k}\sum_{i}^{k} AP_i \tag{6}$$

2.3 Research related to the use of computer vision in detecting defects in road pavements.

Hoang (2018) proposed an artificial intelligence model that employs image processing methods such as gaussian filter, directional filter, and integral projection to extract features of potholes found on asphalt pavement. Subsequently, two machine learning algorithms, Least-Squares Support Vector Machine (LS-SVM) and Artificial Neural Network (ANN), were applied to assign the "pothole" class.

Maeda et al. (2018) proposed a defect detection method into eight categories and emphasizes the availability of a dataset that can be accessed by the public. A total of 9,053 images of pavement surfaces were collected using a low-cost smartphone in various towns in Japan. These images captured a total of 15,435 documented defects. The authors conducted a performance evaluation of the SSD MobileNet and Inception V2 algorithms, focusing on their primary contribution of examining and assessing road problems using a mobile application.

Espíndola, Freitas and Nobre Jr. (2021) introduced a method for identifying potholes, patches, and cracks in road pavements by utilizing YOLO versions 3 and 4. An action camera was affixed to the windshield of a vehicle to capture images of pavement surfaces. This resulted in a collection of 360 images, with each image comprising 500 defects for each class. The researchers examined the impact of image size metrics and the number of iterations on YOLO versions 3 and 4. The primary contribution of this work was the identification of imperfections in road pavements using a low-cost camera positioned on a vehicle.

The selection of the YOLOv3 framework in this investigation is justified by its efficacy and speed (Redmon et al., 2016), which are essential for immediate detection of road pavement issues. The primary objective is to determine the optimal technique for image framing, utilizing the findings to evaluate the precision of YOLOv3 in detecting defects. The objective of focusing on YOLOv3 for analysis is to not only improve defect identification but also highlight the importance of image collecting and framing tactics in achieving accurate defect detection results.

3. METHOD

The datasets employed in the study consisted of images captured in road segments of the BR-020 highway in the state of Ceará, where the presence of potholes and patches was evident.

To acquire images and videos of the highways, a combination of two smartphones (iPhone 12 Pro and Samsung Galaxy S20 FE) and two action cameras (GARMIN Virb Ultra 30 and GOPRO Hero 7) were used. Subsequent to the installation of these devices in the car, necessary modifications were made to accurately detect pavement defects, and the process of collecting data commenced at an average velocity of 80 km/h. Additionally, the datasets encompass a diversity of conditions, including various lighting conditions and shadows from objects at the roadside. Figure 1 illustrates the positioning of the equipment used in the data collection process of this investigation. .

3.1. Data acquisition using smartphones

Two smartphones were affixed to the inside windshield of a car to record panoramic images of the road. Smartphone Type 1, an iPhone 12 Pro (Figure 1a), and Smartphone Type 2, a Samsung Galaxy S20 FE (Figure 1b), were used for this purpose.

Using Smartphone Type 1 and Smartphone Type 2, four datasets were created by recording videos at 30 frames per second with 2x zoom through the devices' native camera applications. Both smartphones recorded videos in two distinct resolutions: FullHD (1920 x 1080 pixels) and 4K (3840 x 2160 pixels).

The datasets were designated as SmartPhone 1 – FHD, SmartPhone 1 – 4K, SmartPhone 2 – FHD, and SmartPhone 2 – 4K. The author traveled the same road section, totaling 200 km, to obtain these four sets of images. Recordings were made during the day and under favorable weather conditions.

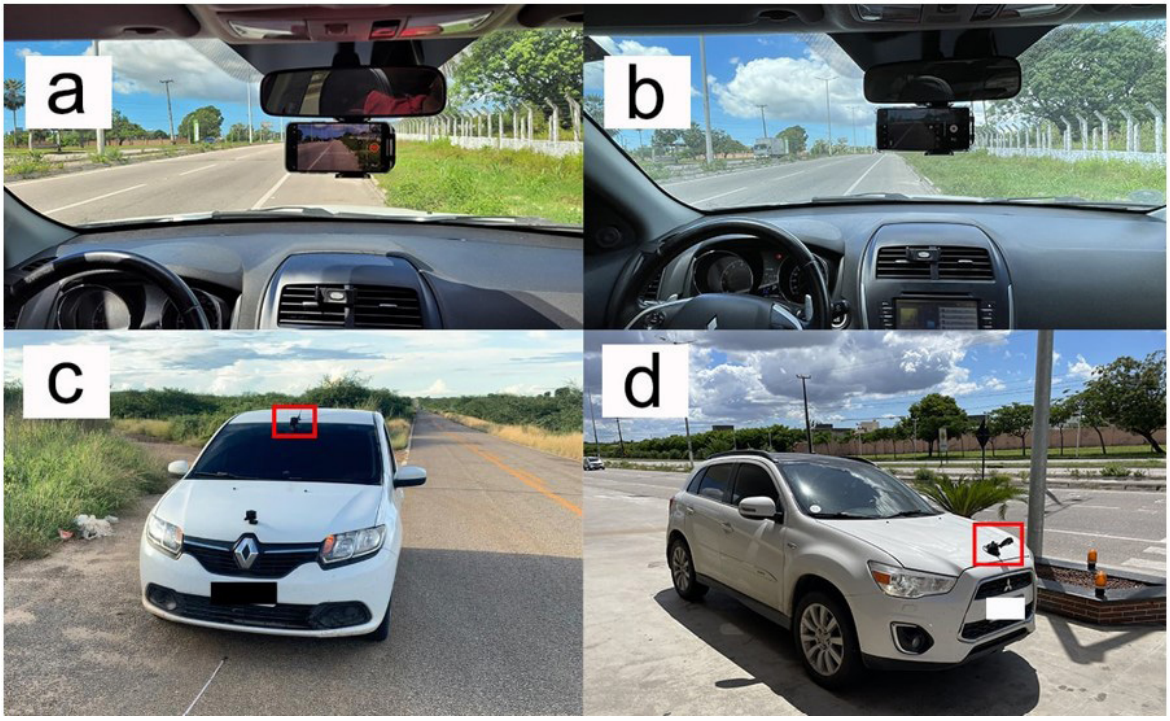

Figure 1. Image Acquisition by Equipment (a) Smartphone Type 1, (b) Smartphone Type 2, (c) Action Camera Type 1, and (d) Action Camera Type 2.

Upon completion of the survey, files were transferred from the smartphones to the computer, and frames from the videos were extracted to generate JPEG format images. Subsequently, after extraction and selection of valid images, i.e., those showing potholes and patches, 291 images were obtained for SmartPhone 1 - FHD, 519 for SmartPhone 1 - 4K, 330 for SmartPhone 2 - FHD, and 406 for SmartPhone 2 - 4K.

3.2. Data acquisition using Action Camera Type 1

Action Camera Type 1, a GARMIN Virb Ultra 30 (Figure 1c), was installed on the external windshield of a passenger vehicle, configured to capture panoramic images of the road.

The dataset generated by Company A, using action camera type 1, was obtained through the evaluation of approximately 800 km, collecting data in the form of JPEG images. Photo capture occurred every 20 meters using the “Interval between photos” option (Travelapse), with GPS enabled. All images were recorded during the day and under favorable weather conditions. It is important to note that the images were framed to provide a panoramic view of the road, resulting in limited visual information about the pavement but with extensive contextualization of the surrounding landscape. This framing choice was deliberate, aiming to analyze the CNN’s accuracy in panoramic image scenarios. However, the images in this dataset exhibited excessive brightness and limited sharpness, impairing the visibility of defects.

Approximately 50 images per kilometer were generated, totaling 40,000 images, stored in 4K resolution (3840 x 2160 pixels) on the camera's memory card. After extraction and selection of images, 10,000 images remained showing potholes and patches.

3.3. Data Acquisition Using Action Camera Type 2

Action Camera Type 2, a GoPro Hero 7 (Figure 1d), was mounted on the hood of a passenger vehicle (external part) at an average speed of 80 km/h and configured to focus on the pavement.

This dataset, developed by the author, covered a total distance of 200 km. Data collection occurred in the form of JPEG images, using the "Time Lapse Photo" option, programmed to capture an image every 0.5 seconds in 12-megapixel resolution, in linear mode, and with GPS enabled for location registration. The linear mode was chosen to avoid distortions in the images. All images were captured during the day and under favorable weather conditions. The framing of the images was specifically directed to the pavement, ensuring sharpness that allowed easy identification of defects with the naked eye.

At the end of the survey, approximately 14,000 images were recorded, stored in 4K resolution (3840 x 2160 pixels) on the camera's memory card. After extraction and selection, 1,389 images showing potholes and patches were obtained.

3.4. Structuring of Datasets

It is relevant to highlight those various sections of the road displayed pavement in satisfactory conservation conditions. In this context, it was imperative to perform a selection of the collected images, dividing them into two distinct categories: one exhibiting potholes and patches, and the other consisting of images that were defect free or had different defects. . Images classified as "defective" encompassed those containing potholes and/or patches, being directed to the labeling process. Conversely, images classified as "non-defective" were those that did not show any potholes or patches and were therefore not included in the labelling process.

The labeling procedure consisted of manual identification and location of defects present in the images. Once this step was completed, the resulting .txt label files were utilized for training and validating the CNN.. These collected data were integrated into a model designed specifically to determine the type of defect, classifying them as potholes or patches, and identifying the corresponding position in the visual field of the image where the defect manifests.

Table 1 provides the quantitative values of potholes and patches labeled per dataset at the end of the labeling phase.

Tabela 1: Quantitative of Images, Potholes, and Patches per Dataset.

Dataset	Image Files	Labeled Potholes	Labeled Patches
1 – SmartPhone 1 - FHD	291	157	613
2 - SmartPhone 1 - 4K	519	726	743
3 - SmartPhone 2 - FHD	330	349	320
4 - SmartPhone 2 - 4K	406	364	741
5 - Action Camera Type 1 - 4K	10,000	6,001	17,090
6 - Action Camera Type 2 - 4K	1,389	1,505	1,192

After creating the labels, the training phase of the model using YOLOv3 was initiated.

3.5. Computational infrastructure

The YOLOv3 framework was utilized for the training process, operating on the Linux Operating System (Distribution: Ubuntu 20.04 LTS). The studies were performed using a laptop that had an Intel Core i5 processor, 32GB of RAM, a 1TB NVMe M.2 SSD, and a graphics card with 4GB of dedicated memory. The training process for each neural network required around 24 hours. Both the detection and training operations were performed using the GPU, taking advantage of the computational power of the graphics card in the equipment.

3.6. YOLOv3 parameters

The batch size was configured to 64, and the subdivisions were set to 16, based on the system's graphical capabilities. The filter parameter was set to 21, max_batches to 4000, and steps to 3200 and 3600. The author of the YOLOv3 framework provided guidelines for defining these settings (Redmon et al., 2016).

It is relevant tomention that YOLOv3 resizes the images using the width and height parameters in the settings. It is important to highlight that any modification in resizing will affect the precision of the final result.. However, it is important to consider that the larger the size of the image sent to the model, the more memory and processing resources will be demanded. For this investigation, the dimensions of 512 were used for both the width and height.

3.7. YOLOv3 training

Once the model parameters are defined, training begins, generating a weight file every 1000 iterations. It is important to highlight that the training set corresponds to 90% of the dataset, while the remaining 10% constitute the validation set. The training and validation images were carefully selected to ensure distinctiveness and were chosen randomly to evaluate the effectiveness of the detections.

During training, the model receives as input the defined parameters and the images with their corresponding labels. As output, metrics and a weight file representing the network's learning are generated.

Following the conclusion of each training, the process of defect detection is performed for each of the datasets. A Python code was developed to read all images from the dataset directories in the file system, identifying potholes and patches. During the execution of the detections, new JPEG format image files are generated, representing the output of each model.

4. ANALYSIS AND DISCUSSION OF RESULTS

The analysis of the results obtained in the research considered the datasets structured in the context of this study. A comparison of the results achieved for each dataset that was subjected to training was performed.

4.1. Training analysis

Table 2 displays the results of the model training conducted in this study. By observing the evaluation metrics and considering the specific characteristics of each trained image set, it is possible to infer the most effective camera positioning for the automatic detection of the investigated defects.

Table 2: Results of the metrics from the six trainings conducted.

Dataset	AP (%)		Precision (%)	Recall (%)	F1-Score (%)	IoU (%)	mAP (%)
	Pothole	Patch					
1 – SmartPhone 1 - FHD	71.86	83.86	86	77	81	65.49	78.86
2 - SmartPhone 1 - 4K	63.31	76.38	77	65	71	55.54	69.84
3 - SmartPhone 2 - FHD	77.08	65.28	81	69	75	59.97	79.18
4 - SmartPhone 2 - 4K	66.8	73.3	81	71	76	61.86	78.05
5 - Action Camera Type 1 - 4K	57.82	76.42	73	68	70	53.41	57.12
6 - Action Camera Type 2 - 4K	99.89	95.91	98	97	98	80.14	84.90

In the first two trainings, it's possible to conclude that dataset 1 (SmartPhone 1 - FHD) exhibited superior combined metrics compared to dataset 2 (SmartPhone 1 - 4K). There was an increase in all metrics except for true positives (TP) – an input metric in Precision and Recall. Despite the camera framing being identical for both datasets, the 4K images yielded significantly inferior results compared to the FullHD images from Smartphone 1.

In comparing datasets 3 (SmartPhone 2 - FHD) and 4 (SmartPhone 2 - 4K), it's observed that the learning from the 4K dataset showed equal or superior results in Precision, Recall, F1-score, and true positives (TP) compared to the FullHD dataset. However, there was a significant increase in the number of false positives (FP) and false negatives (FN) in dataset 4, compromising the overall result. The limitation in the quantity of images may have induced a greater variation in the percentage of metrics, given the fluctuations in FP and FN numbers. This directly influenced Precision and Recall and consequently affected the F1-Score, AP, and mAP. Including more images in the training process would likely generate more consistent results.

The trainings conducted on smartphones had exactly the same camera framing. However, datasets 2 and 4 exhibited more FP than their respective FullHD counterparts (datasets 1 and 3). Despite this occurrence, it is observed that the smartphone datasets maintain an overall average (mAP) between 70% and 80% and a precision between 81% and 86%.

On the other hand, dataset 5 revealed the worst performance among the trainings. Despite having the largest number of labeled images compared to the others, superior results were expected given the volume of data for learning. However, the data collection phase was conducted outside a standard, resulting in images with little visual information of the highway and inadequate visibility of defects in the pavement.

Dataset 6 presented the best results in all metrics. In this set, the camera framing was fully focused and zoomed in on the pavement, unlike the other datasets, which provided a panoramic view of the road. Additionally, the images were quite clear, and the pavement defects could be visualized excellently.

4.2. Discussions

A subjective analysis highlights the main recurring scenarios during validation.

Figure 2 illustrates three scenarios from the dataset captured by SmartPhone Type 1 - FHD. The scenario in Figure 2a represents a successful case where both labeled patches were detected by the model. In the scenario of Figure 2b, a pothole was not labeled, but the model was able to detect

the defect, making it a favorable case, although it is considered a false negative in YOLOv3 metrics. In the third scenario (Figure 2c), a situation is observed where the furthest pothole was not detected due to its size relative to the image. However, due to the frequency of the images, more distant defects will be identified in subsequent records.

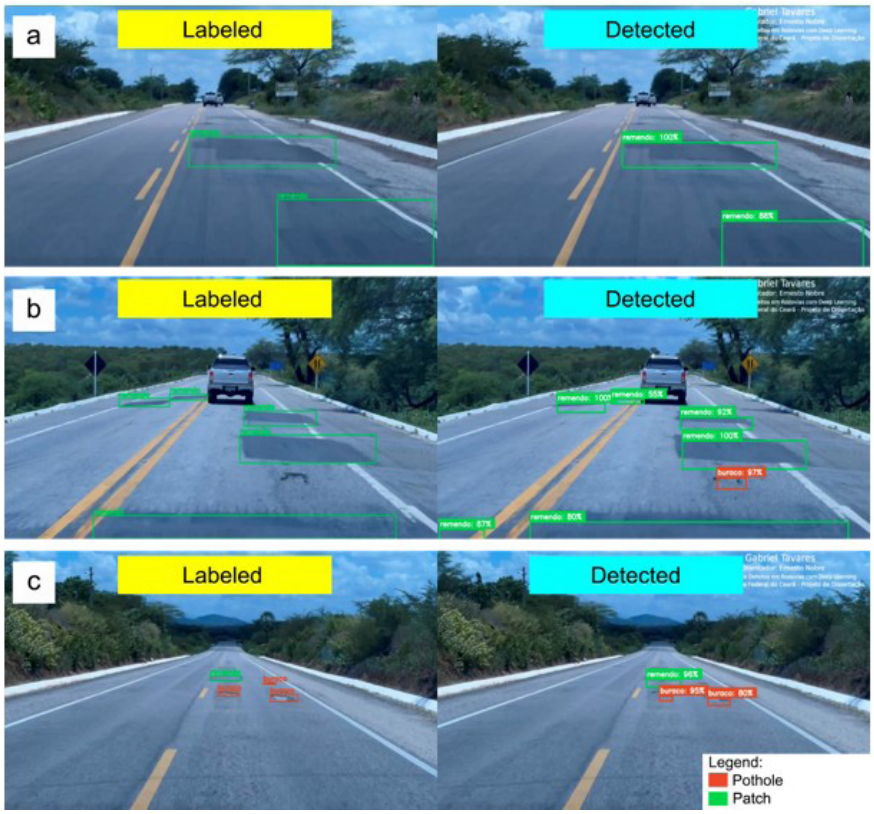

Figure 2. Validation images from the dataset of Smartphone Type 1 - FHD.

Figure 3 displays three scenarios from the dataset captured by Smartphone Type 1 - 4K. The scenario depicted in Figure 3a illustrates a successful case, where both labeled potholes were correctly identified during detection. In the scenario of Figure 3b, there is a situation where two labeled defects were not found, while one unlabeled defect was detected. Here, there are two false negatives, similar to the previous scenario (Figure 2b), but with the addition of two unidentified defects, indicating the need for a more robust training. The third scenario (Figure 3c) presents another case of false negative, with three defects not detected. This third case resembles the second, suggesting the need for new training with the inclusion of more images with different arrangements of defects and a review of the labels in this dataset.

Figure 4 also displays three scenarios from the dataset captured by Smartphone Type 2 - FHD. In the first case (Figure 4a), we have a successful situation where both defects were correctly identified. In the second scenario (Figure 4b), there are two false negatives, indicating an issue in metric generation, but also highlighting that the model recognized situations where labeling was not performed. In Figure 4c, we observe a problem of not identifying a pothole. This third scenario may result from the limited quantity of images in the dataset and could likely be addressed by including more defects and conducting a new training session.

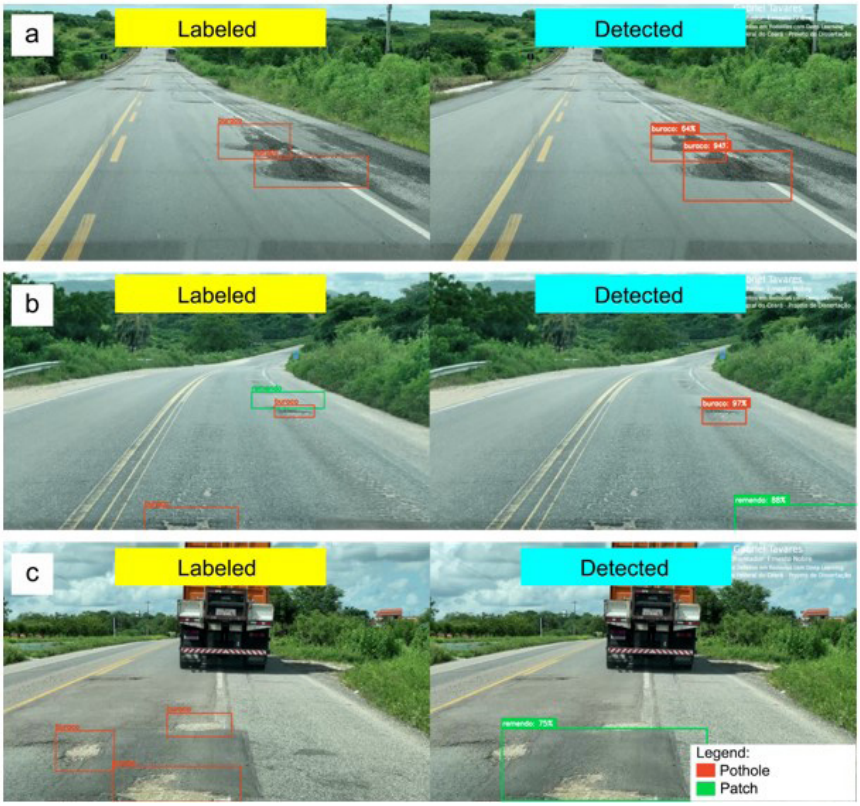

Figure 3. Validation images from the dataset of Smartphone Type 1 - 4K.

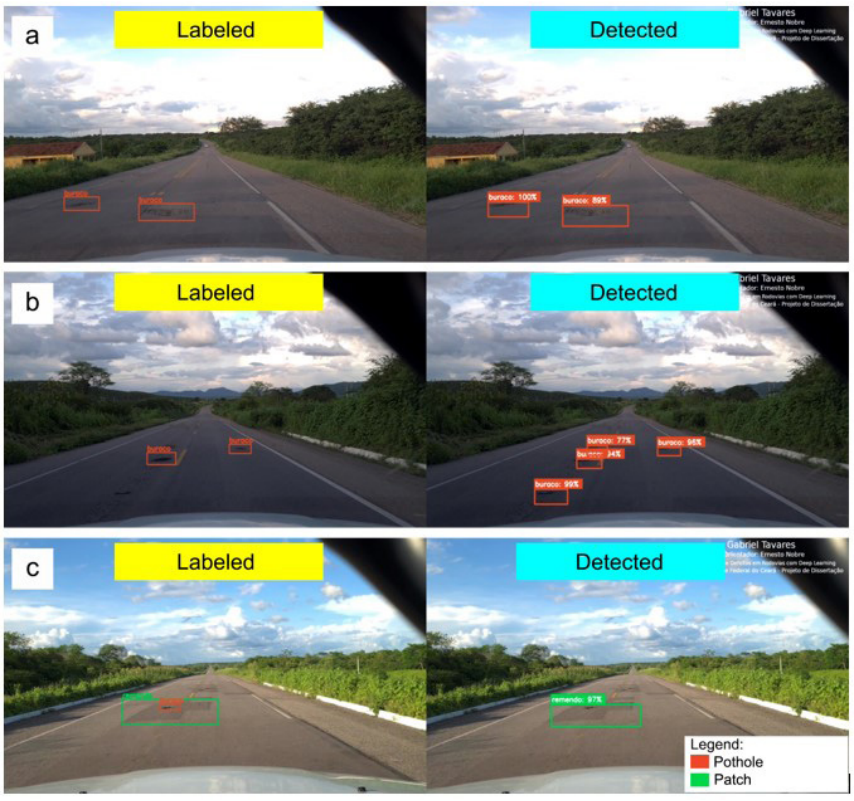

Figure 4. Validation images from the dataset captured by Smartphone Type 2 - FHD.

Figure 5 illustrates three scenarios from the dataset captured by Smartphone Type 2 - 4K. In the first scenario (Figure 5a), we observe a successful outcome where the defects were correctly identified according to the labels. In the second scenario (Figure 5b), an interesting observation arises: despite labeling two patches, the model detected them as a single defect. This suggests that the model learned to identify patches accurately, either as separate entities or as a combined defect. In Figure 5c, the model identified two potholes while failing to detect another two. Improvements in the second and third scenarios depicted in Figure 5 are anticipated with the addition of more images to the dataset, along with a thorough review and adjustment of incomplete or incorrect labeling.

The results from the dataset captured by Action Camera Type 1 are presented in Figure 6. In the first and second scenarios (Figure 6a and Figure 6b), the model successfully identified the defects. Due to the wide panoramic view of the images, the more distant defects appeared with small dimensions and low sharpness, making detection challenging. It is important to note that such cases are common in this dataset. In the third scenario (Figure 6c), the object in the scene is clearer and closer to the camera, enabling detection. The main challenge of this dataset is the camera's wide field of view, resulting in a cluttered scene, as only defects on the pavement are relevant. Even with a lower degree of accuracy, detection is still possible. To improve accuracy, the frequency of image capture in the field can be increased, taking a photo every 5 meters, ensuring that more distant defects are identified in subsequent images.

In the dataset from Action Camera Type 1, a predominance of sky and parts of the vehicle can be observed, accounting for approximately 70% of the image. Although the images were recorded in 4K resolution, during training, they were resized to 512 pixels in width, resulting in a reduction of about eight times. This resizing means that defects became extremely small, posing a greater challenge for the model in detecting these objects.

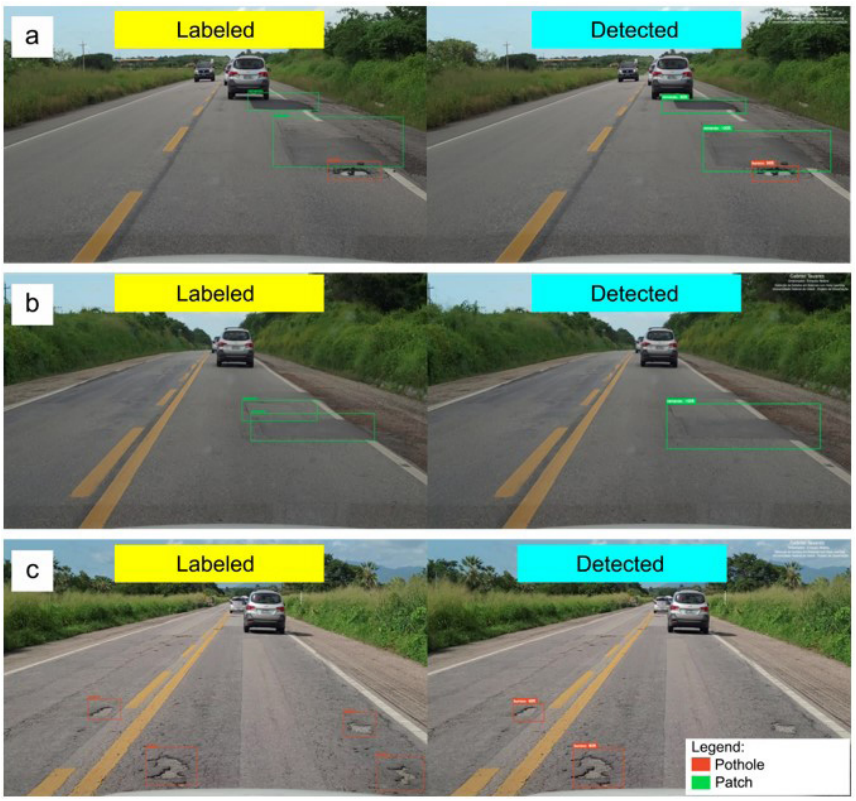

Figure 5. Validation images from the dataset captured by Smartphone Type 2 - 4K.

The results of the dataset from Action Camera Type 2 are presented in Figure 7. In the scenario of Figure 7a, a successful case is observed, where the model identified all defects. In the second case (Figure 7b), there were false negatives. In the third (Figure 7c), the model did not identify the water-covered pothole at the end of the image. In this last case, few labels similar to this were provided to the model, suggesting the inclusion of more situations of this nature to improve its accuracy. It is noteworthy that this dataset yielded the best results in terms of metrics, and a detailed analysis of each case individually can further contribute to the model's improvement, mainly through increasing the number of images. It is important to highlight that this dataset provided an enlarged view of the objects in the scene, allowing the defects to remain clearly visible, sharp, and with an appropriate size for identification by the YOLOv3 network.

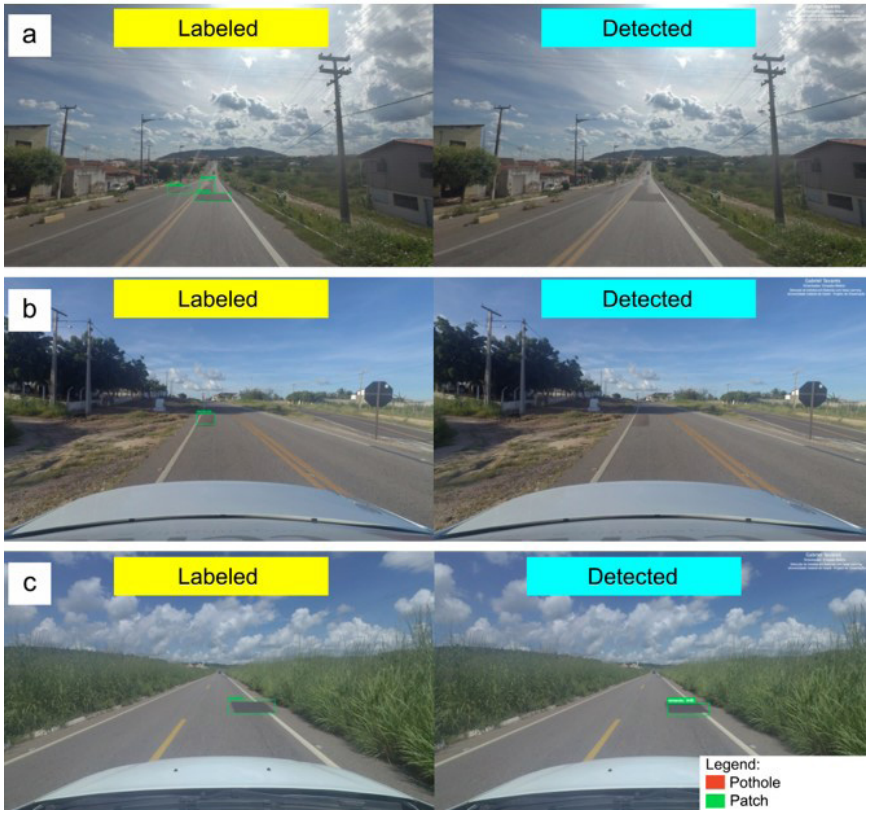

Figure 6. Validation images from the dataset of Action Camera Type 1.

Looking at the results in Table 2, it becomes evident that dataset 6 exhibits higher Average Precision (AP) values compared to the other datasets generated by smartphones (datasets 1, 2, 3, and 4) and Action Camera Type 1 (dataset 5). One possible explanation for this discrepancy is that the defects recorded in the images of dataset 5 occupy a very small space in the image, as the roadway represents only about 30% to 40% of the image, making them difficult to visualize. On the other hand, the defects recorded in dataset 6 occupy a considerable space in most cases, making them easier to detect and, consequently, contributing to a higher average precision.

It is worth noting that the classes in the datasets had imbalanced data, as the interest in this research was to evaluate whether even with lower quality data, the model would show better precision due to the quantity.

According to the literature, YOLOv3 presents challenges in detecting small objects, which may explain the results obtained with dataset 5. However, it is noteworthy that the value of average precision (AP) in the training of dataset 6 reached 98% precision. This demonstrates that the approach of using images with the camera directed at the pavement had a positive impact on the results.

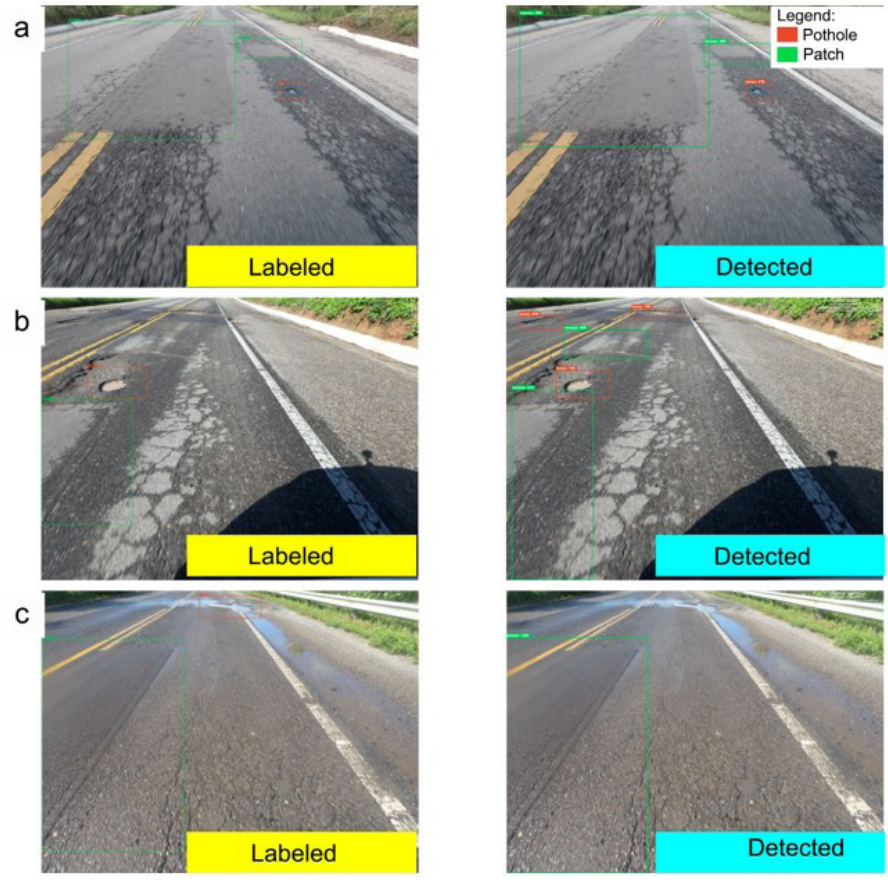

Figure 7. Validation images from the dataset of Action Camera Type 2.

5. CONCLUSIONS

In conclusion, the results of this research indicate that the choice of YOLOv3 for automated detection of road pavement defects was effective, demonstrating good accuracy, especially when the camera is directly focused on the pavement. Training and evaluation were conducted on six different datasets. The selection of data collection methods using smartphones and action cameras showed a significant impact on the results obtained. It was observed that the most effective framing, specifically focused on the pavement, achieved a remarkable accuracy of 98%, highlighting the critical importance of data collection and framing techniques in the model's efficiency.

Furthermore, the use of YOLOv3 proved to be an efficient tool for defect detection, showcasing its ability to accurately identify pavement defects even in challenging contexts. The comparative analysis among the datasets revealed important nuances, emphasizing that image quality, capture frequency, and camera focus played significant roles in the results. Datasets that provided a clearer and more detailed view of the defects achieved superior metrics, while those with limitations, such as objects occupying a large part of the scene, exhibited lower precision. Considering these

factors is crucial for the model's performance, indicating the need for careful planning in data collection and labeling to train more robust models.

Finally, the results of this study not only strengthen the existing literature on automated detection of pavement defects but also provide a solid foundation for future research and practical applications. The efficiency of YOLOv3, combined with careful consideration of data collection techniques, stands out as a crucial advancement in the field, offering contributions to effective management of road infrastructures and enhancing decision-making in the realm of road transportation.

As future suggestions, it is proposed to expand the quantity of images and labels related to potholes and patches in datasets 1, 2, 3, 4, and 6. This expansion aims to evaluate the performance and accuracy of the model in more comprehensive situations, enabling a more robust and comprehensive analysis of the results obtained.

REFERENCES

Balbo, J.T. (2007) *Pavimentação Asfáltica: Materiais, Projeto e Restauração*. São Paulo: Oficina de Textos.

Espíndola, A.C.; G.T.M. Freitas and E.F. Nobre Jr. (2021) Pothole and patch detection on asphalt pavement using deep convolutional neural network. In *Proceedings of the Joint XLII Ibero-Latin-American Congress on Computational Methods in Engineering; III Pan-American Congress on Computational Mechanics, ABMEC-IACM.* Rio de Janeiro: ABMEC, p. 1-7. Available at: <https://repositorio.ufc.br/bitstream/riufc/63680/1/2021_eve_acespindola1.pdf> (accessed 03/23/2022).

Everingham, M.; L.V. Gool; C.K.I. Williams et al. (2010) The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, v. 88, n. 2, p. 303-338. DOI: 10.1007/s11263-009-0275-4.

Haykin, S. (1998) *Neural Networks: A Comprehensive Foundation*. Hoboken: Prentice Hall PTR.

Hoang, N.D. (2018) An artificial intelligence method for asphalt pavement pothole detection using least squares support vector machine and neural network with steerable filter-based feature extraction. *Advances in Civil Engineering*, v. 2018, p. 7419058. DOI: http://doi.org/10.1155/2018/7419058.

Khan, A.I. and S. Al-Habsi (2020) Machine learning in computer vision. *Procedia Computer Science*, v. 167, n. 13, p. 1444-1151. DOI: 10.1016/j.procs.2020.03.355.

Maeda, H.; Y. Sekimoto; T. Seto et al. (2018) Road damage detection using deep neural networks with images captured through a smartphone. *Computer-Aided Civil and Infrastructure Engineering*, v. 33, p. 1127-41. DOI: 10.1111/mice.12387.

Padilla, R.; S.L. Netto and E.A.D. Silva (2020) A survey on performance metrics for object-detection algorithms. In *Proceedings of the 2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*. USA: IEEE, p. 237-242. DOI: 10.1109/IWSSIP48289.2020.9145130.

Paterson, W.D. (1987) *Road Deterioration and Maintenance Effects: Models for Planning and Management.* Baltimore: The Johns Hopkins University Press.

Prince, S.J. (2012) *Computer Vision: Models, Learning, and Inference*. Cambridge: Cambridge University Press. DOI: 10.1017/CBO9780511996504.

Radovic, M.; O. Adarkwa and Q. Wang (2017) Object recognition in aerial images using convolutional neural networks. *Journal of Imaging*, v. 3, n. 2, p. 21. DOI: 10.3390/jimaging3020021.

Redmon, J.; S. Divvala; R. Girshick et al. (2016) You only look once: unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. USA: IEEE, pp. 779-788. Available at: <https://www.cv-foundation.org/openaccess/content_cvpr_2016/html/Redmon_You_Only_Look_CVPR_2016_paper.html> (accessed 03/19/2021).

Sasaki, Y. (2007) The truth of the F-measure. *Teach Tutor Mater*, v. 1, n. 5, p. 1-5. Available at: <https://nicolasshu.com/assets/pdf/Sasaki_2007_The%20Truth%20of%20the%20F-measure.pdf> (accessed 02/12/2021).

Sholevar, N.; A. Golroo and S.R. Esfahani (2022) Machine learning techniques for pavement condition evaluation, *Automation in Construction*, v. 136, p. 104190. DOI: 10.1016/j.autcon.2022.104190.