XX International Conference on Building Pathology and Constructions Repair
29th to 31st May 2024 - Fortaleza, Brazil

A Deep Learning Algorithm to Detect Cracks on Civil Engineering Building Elements

Paulo H A BEZERRA[1], Heitor C DANTAS[2] , Letícia M G MORAIS[3], Rosana C B REGO[4]

[1] Federal University Rural of Semi Arid Region, LOADS®, Pau dos Ferros-RN, Brazil, paulo.bezerra@ufersa.edu.br
[2] Federal University Rural of Semi Arid Region, Pau dos Ferros-RN, Brazil, heitor.dantas@alunos.ufersa.edu.br
[3] Federal University Rural of Semi Arid Region, Pau dos Ferros-RN, Brazil, leticia.morais@alunos.ufersa.edu.br
[4] Federal University Rural of Semi Arid Region, LOADS®, Pau dos Ferros-RN, Brazil, rosana.rego@ufersa.edu.br

Abstract: The detection of cracks on building element surfaces is crucial for durability, structural integrity, and safety. Non-automated methods are time-consuming and can lead to misidentification of construction defects. Additionally, traditional computational methods often face limitations in accuracy and efficiency. This necessitates the exploration of advanced deep learning techniques. In this paper, we propose a novel approach for crack detection in construction elements, the U-Net-MobileNet combined deep learning model. This approach focuses on evaluating the algorithm's performance across various construction elements. Regarding performance, the model leverages the semantic segmentation capabilities of U-Net and the classification capacity of MobileNet, resulting in an accurate algorithm for surface crack detection. In terms of applicability, it provides a powerful tool for civil construction professionals to identify the presence of cracks in several types of building elements. To demonstrate the algorithm's functionality and accuracy, we performed simulations on a dataset containing 40,000 images of concrete crack surfaces. We employed metrics such as accuracy, precision, recall, and F1-score to evaluate performance. Results indicated the effectiveness of the proposed approach, achieving superior performance compared to individual models, and providing visual representations of crack segmentation and detection outcomes, thus suggesting the model's efficacy in accurately identifying and locating cracks on surfaces of civil construction elements. Furthermore, we analyzed a variety of building elements with cracks and demonstrated that the model can detect the presence of cracks with high accuracy. Furthermore, the model neither introduces false positives nor ignores straight cracks - which are similar to interfaces of building elements. This research contributes to advancing the field of civil construction maintenance and safety inspection by applying deep learning techniques.

Keywords: Crack Detection, Deep Learning, Building Elements, Civil Construction, Artificial Intelligence.

1. Introduction

Building cracks are defined as full or partial separations of a construction element into two or more portions due to breaking or fracturing (Khan et al., 2023; Bano et al., 2023). Therefore, detection of cracks on building elements is crucial in terms of durability, structural integrity and safety; besides, early identification prevents further damages and expensive costs of maintenance (Mignon et al., 2017). However, detecting cracks on surfaces presents several difficulties due to the nature of surface materials, environmental conditions, and the necessity for precise measurements (Yamaguchi et al., 2008). For that reason, researchers are actively tackling these challenges by proposing algorithms and technologies to better understand material behavior and surface properties.

The process of detecting cracks in the building elements can be performed in two ways, using destructive or non-destructive testing (Mohan et al., 2017). However, traditional methods for crack detection often depend on manual inspection, which can be time-consuming, labor-intensive, and prone to human error, as mentioned in Hüthwohl and Brilakis (2018). In that direction, several works have approached non-destructive computational detection and classification of cracks.

Mohan et al. (2017) discussed the limitations of manual crack inspection and proposed automatic image-based crack detection as a more objective alternative. The authors conducted a detailed survey of 50 research papers on crack detection, analyzing the image processing techniques, objectives, accuracy level, error level, and image datasets used. Moreover, Anitha et al. (2022) proposed a strategy for crack severity classification using a bilayer crack detection algorithm.

Artificial intelligence (AI) techniques, such as deep learning algorithms, can assist visual inspection on detecting cracks in their early stages, preempting potential serious issues and enabling timely corrective actions. Some researchers have turned to deep learning models to achieve surface crack detection (Yao et al., 2014). Deep learning models like convoluted neural networks (CNNs), originally developed for image recognition and video processing, can be leveraged to recognize cracks in construction. For example, in a study by Müller et al. (2021), a feedforward neural network was utilized to classify cracked and uncracked specimens in mechanical experiments. This classifier demonstrated a 99% accuracy rate in detecting the progression of cracks. In the same way, Kim et al. (2021) proposed a shallow convolutional neural network-based architecture called OLeNet for detecting surface concrete cracks. They compared its performance with established models such as Visual Geometry Group, Inception, and Residual Neural Network. Additionally, Joshi et al. (2022) introduced a deep learning model aimed at detecting and segmenting cracks in concrete surfaces.

In the past few years, the advent of deep learning has transformed the field of crack detection, offering automated solutions that can analyze large-scale datasets with remarkable accuracy and efficiency. However, some algorithms fail in segmentation capabilities or classification strengths. The algorithms frequently tackle distinct challenges, including model scale, complexity, computational efficacy, and training robustness. The selection of a specific deep neural network architecture is contingent upon the particular demands of the task at hand and the accessible computational capacities. For instance, MobileNet has demonstrated superior performance in numerous computer vision tasks (Liu et al. 2019). However, the classification model, like MobileNet, might misclassify due to its inability to detect cracks. To address this issue, segmentation techniques can be applied. The segmentation algorithms, such as CNNs, K-means, and Watershed Algorithm**,** separate objects or regions within an image to facilitate analysis or recognition tasks. In this way, we employed the U-Net model, a kind of CNN, for image segmentation. The model isolates the crack from the rest of the image (Dais et al., 2021). Subsequently, this segmented crack is utilized for classification, ensuring more accurate classification results.

Motivated by the aforementioned discussion, we presented a deep learning model for crack detection utilizing the combination of U-Net and MobileNet networks. The algorithm combines the segmentation capabilities of U-Net and the classification capability of MobileNet. The approach prioritizes identifying the presence of cracks over delineation, achieved through the utilization of semantic segmentation and classification algorithms. The method prioritizes the rapid identification of regions with potential vulnerabilities; in other words, it focuses on detecting the presence of cracks without requiring detailed crack contour delineation.

The work is divided as follows: In Section 2, the methodology is presented, with a detailed description of the dataset, where its characteristics, sources, and preprocessing steps are discussed. Following this, the U-Net architecture and the proposed model are presented. In Section 3, the performance metrics of the U-Net, MobileNet, and proposed models are compared, and test results are presented. Furthermore, the models' efficacy is tested in real-world scenarios. Finally, in Section 4 the conclusions are presented.

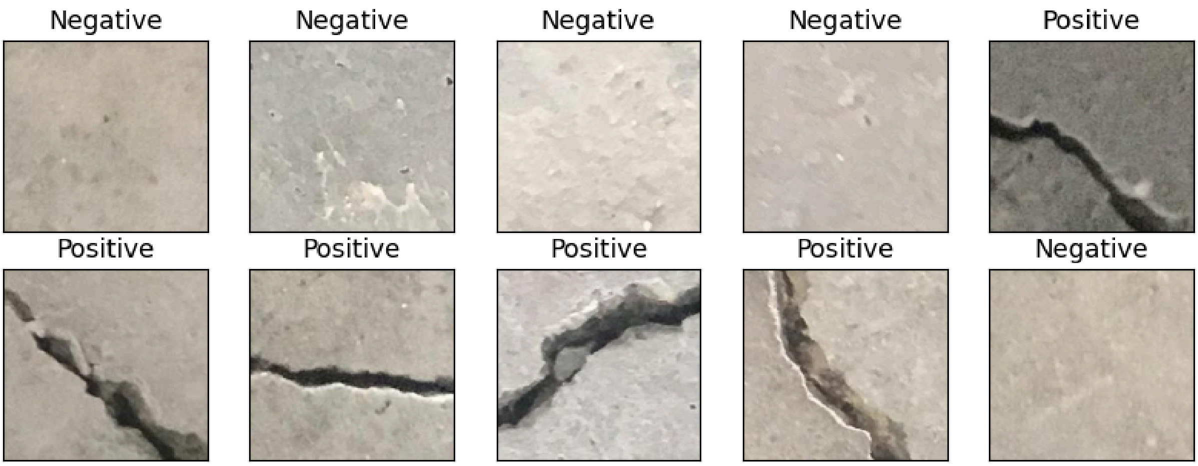
2. Methodology

This section provides an overview of the dataset utilized, the deep learning techniques, and the segmentation method applied to concrete crack detection.

2.1 - Dataset

To develop the model presented in this work, the dataset used in the simulation - obtained from Özgenel (2019) - contained 40,000 images depicting both positive and negative samples of cracks on concrete surfaces. Each image serves as an essential input for training and testing the algorithm's capability to accurately extract and classify cracks. Figure 1 depicts some image samples used to train the neural network. Positive images offer insights into the characteristics of cracks in real-world scenarios and negative images provide counterexamples, used to effectively differentiate cracked from intact areas.

Figure 1 – Samples of images used to train the network.



We conducted preprocessing steps, such as eliminating duplicate images, resizing all images to dimensions to 96x96, and converting them to grayscale. These actions were taken to standardize the size and decrease the computational load.

2.1 - Deep learning network model

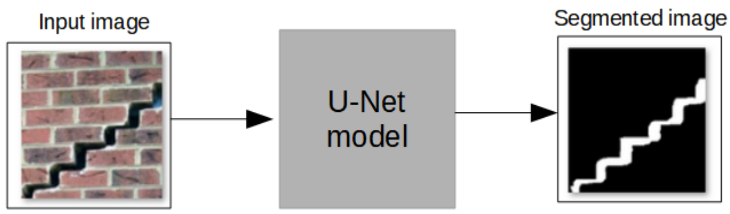
Concerning deep learning networks, as mentioned in Goodfellow et al. (2016), such models are a subset of machine learning algorithms that simulate the human brain's structure, consisting of interconnected layers of artificial neurons. These models are trained to learn complex patterns and relationships present within datasets.

In this paper, we employed two convolutional neural networks (CNN) as deep learning models: the U-Net architecture, proposed by Ronneberger et al. (2018), and MobileNet, introduced by Howard et al. (2017). Different types of CNNs are employed in tasks related to images, some popular ones include: LeNet-5, (LeCun et al., 1998); AlexNet (Krizhevsky et al., 2017; ResNet (He et al., 2016); MobileNet (Howard et al., 2017). For surface crack detection, we employed the MobileNet to predict the presence of cracks in the image and implemented the U-Net to enable the model to understand and analyze images more effectively by delineating boundaries between different regions of the image.

2.1.1 - U-Net image segmentation

The U-Net model segments the image to extract the cracks through a sequence of encoding and decoding layers. The encoder layers are used to extract features from the input image and the decoder layers are applied to reconstruct the segmented image, ultimately allowing it to accurately identify and extract cracks from the input image. In the encoding part, we used convolution blocks with two convolutional layers with a 3x3 kernel, followed by batch normalization and ReLU activation. Then we applied a pooling layer with a window size of 2x2 and a stride of 2, further reducing the resolution of the input image based on the highest-value pixels; finally, we added a dropout layer to prevent overfitting. Figure 2 illustrates how the U-Net model operates to extract cracks from the input image, generating the output, segmented image.

Figure 2 – Scheme of the U-Net operation



For training the model, we used the focal loss proposed by Lin et al. (2017), defined in Equation 1.

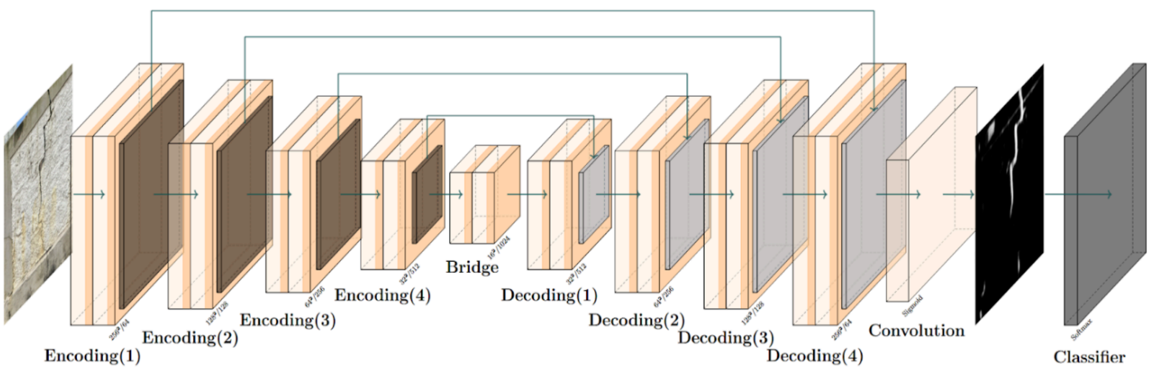
$$L(y, p) = \; - \alpha y(1 - p) \log log\,(p) \; - (1 - y)p^{\gamma} log(1 - p) \quad (1)$$

The parameters in Equation 1 are: $y \in \{0, 1\}$, the binary class label - positive indicates the presence and negative the absence of cracks -; $p \in \{0, 1\}$, an estimate of the probability of the positive crack; y, the focusing parameter that determines the extent to which higher confidence correct predictions influence the overall loss; and α, a hyper parameter that dictates the balance between precision and recall - adjusting error weighting for the positive class accordingly. To minimize Equation 1, we used Adam, a stochastic gradient descent method proposed by Kingma et al. (2014), which is based on adaptive estimation of first-order and second-order moments. After segmentation, we classified the output images, determining whether cracks are present on them.

2.1.2 - U-Net-MobileNet classifier

Regarding the convolution neural network classifier, we implemented MobileNet version 1 (MobileNetv1), illustrated in Figure 3, a lightweight convolutional neural network architecture that serves as an efficient classifier - especially well-suited for mobile and embedded devices.

Figure 3 – U-Net-Mobilev1 deep learning model



We implemented the U-Net model with 23 convolutional layers, and the MobileNet with 28 convolutional layers. Therefore, the UNet-MobileNet model has 51 convolutional layers. We separated 60% of the images for training, 20% for validation, and 20% for testing the model. For the training process, we set the network with 100 epochs, a batch size of 4, a learning rate of 0.0004, and a dropout rate of 0.5. Moreover, we set the gradient algorithm as Adam. The U-Net-MobileNetv1 deep learning model combines the strengths of both architectures for efficient crack detection: U-Net outperforms in semantic segmentation, localizing cracks within images, while MobileNetv1 acts as a classifier, determining if cracks are present. This combination allows the proper detection of cracks in images while maintaining computational efficiency, making it suitable for real-time applications and resource-constrained environments.

3. Results and discussion

3.1 Deep learning architectures performance

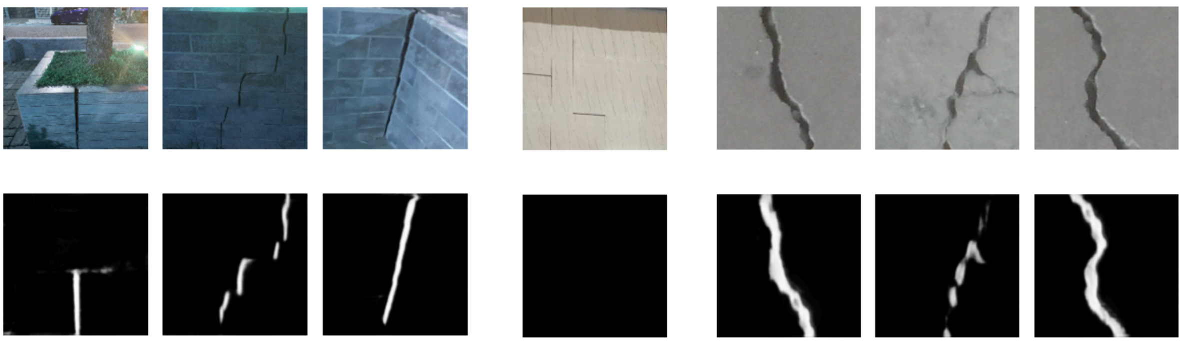
The simulations were performed on an NVIDIA GeForce® GTX 1660 Ti with Python 3.11, TensorFlow framework, and the Edge Impulse platform. According to Table 1, which displays the models' metrics evaluation, MobileNet and U-Net networks demonstrated overall performance with high accuracy, precision, recall, and F1-score. However, the proposed architecture (U-Net-MobileNet) achieved higher accuracy and correctly predicted 98% of cases in which there is a crack in the analyzed concrete surface.

Table 1 – Model's metrics evaluation

Model	Accuracy	Precision	Recall	F1-score
U-Net	98%	70%	98%	80%
MobileNet	99%	99%	99%	99%
U-Net-MobileNet	97%	98%	97%	97%

Figure 4 depicts the surface crack segmentation achieved by employing the U-Net model. The segmentation technique accurately identifies and delineates the cracks on the surface, visually demonstrating the model's efficacy to locate cracks.

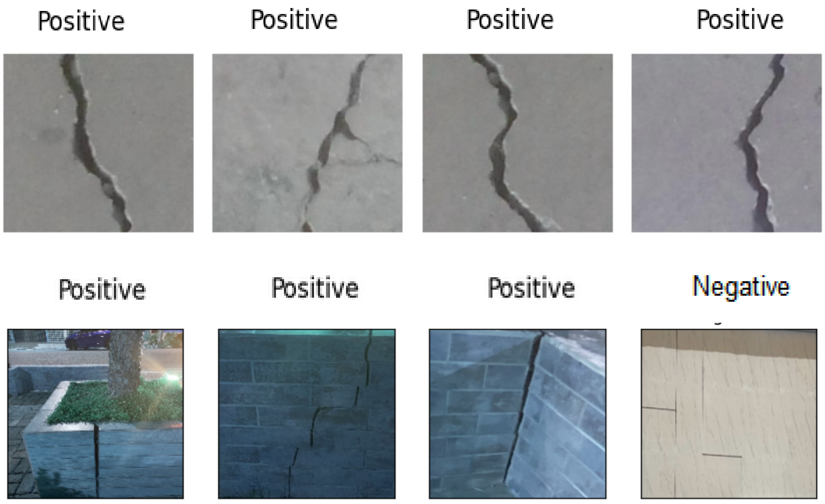
Figure 4 – Surface crack segmentation using U-Net: a – bricks, b- tiles, and c- concrete surface



a b c

In Figure 4 - b, we input a picture displaying tile grout, and the algorithm accurately did not segment the grout as if it was a straight crack. Figure 5 shows the surface crack detection results achieved by employing the U-Net-MobileNet model.

Figure 5 – Surface crack detection using U-Net-MobileNet



Therefore, the combined architecture of U-Net and MobileNet, that is, U-Net-MobileNet, effectively identifies and highlights the presence of surface cracks, demonstrating its performance of accurately detecting cracks within input images.

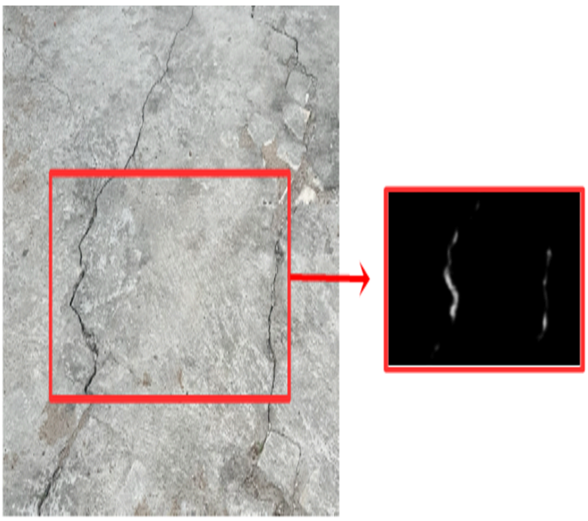
3.2 Tests on different construction elements

To demonstrate the efficiency of the deep learning model developed in this work to detect cracks, we selected pictures of different types of construction surfaces, namely structural members, masonry, plastering, pavement, and tiles.

3.2.1 - Stone block pavement

We input a picture of a stone block pavement partially covered with mortar, presented in Figure 6. In that image, we notice that the cracks were able to cause openings not only on the stone blocks interface grout but also on its covering mortar. In this test, the deep learning model successfully detected the presence of cracks, as shown in the highlighted in the rightmost part of Figure 6.

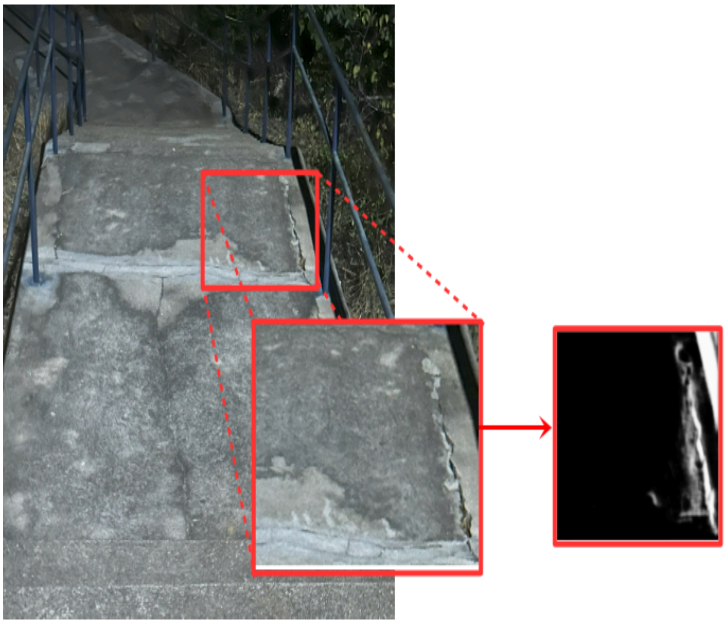
Figure 6 – Stone block pavement with mortar cover



3.2.2 - Cement concrete stair

In Figure 7, we show a picture of a landing, in a cement concrete stair, with multiple cracks. The highlighted region shows an actual crack – in its rightmost lateral part – and a repaired crack, for which a layer of mortar was applied to reconstruct the stair landing. In this test, the algorithm successfully detected the presence of the lateral crack and, as desired, did not detect cracks in the repair streak since it is not actually an opening anymore.

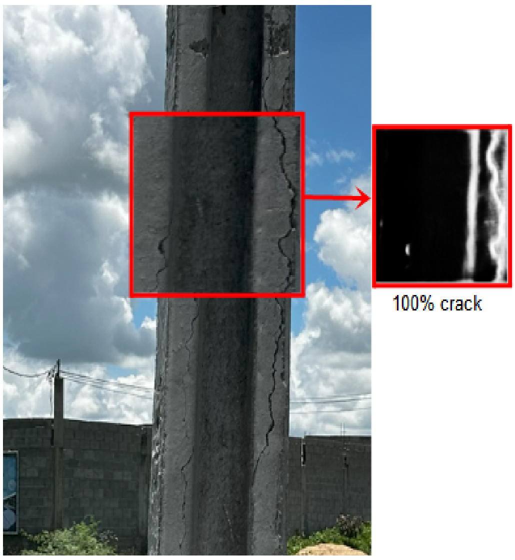
Figure 7 – Landing in cement concret stair



3.2.3 – Reinforced Concrete Street Light Pole

In Figure 8, we present a picture of a street light pole of reinforced concrete with straight cracks along reinforcement bars direction. In the picture, we show how the algorithm adequately interpreted the presence of a crack; the “100% crack” statement means that it has located the existence of a crack in the street light pole.

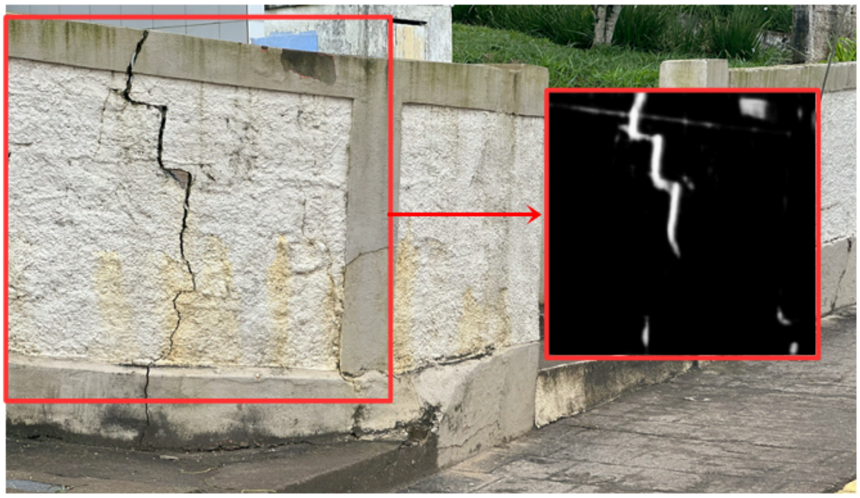
Figure 8 – Street light pole



3.2.4 – Reinforced Concrete Beam and Perforated Brick masonry

We input an image of a wall composed by perforated brick masonry, with rough cast plaster, under a reinforced concrete beam (Figure 9) to analyze how the presence of a crack could be detected if a the opening occurs continuously along adjacent different construction elements. The AI satisfactorily detected the presence of the crack crossing the construction elements.

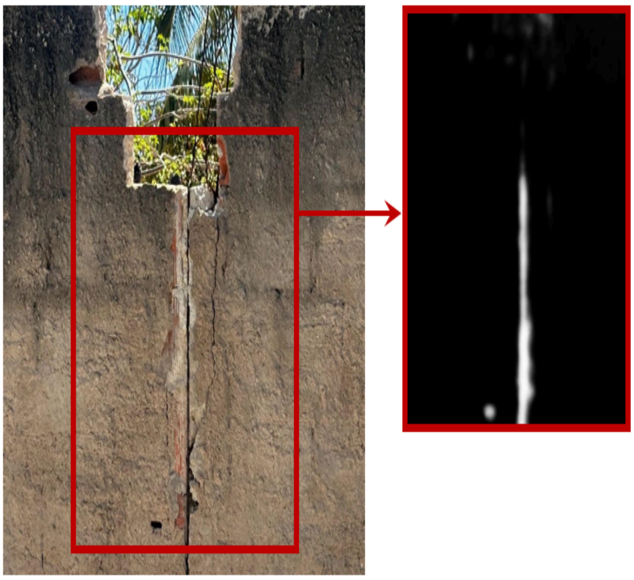
Figure 9 – Wall built using brick masonry with rough cast plaster and reinforced concrete structure



3.2.5 – Brick masonry wall with straight crack

We input an image of a brick masonry wall (Figure 10) to verify if the AI would ignore the straight crack – that is, interpret it as a contour or interface between bricks. As presented in Figure 10, the algorithm adequately identified the crack; therefore, straight cracks can be detected even if they have an appearance of a contour or interface between construction elements.

Figure 10 – Brick masonry wall with straight crack



3.2.6 - Construction elements with crack-like appearance

AI models may provide erroneous results, caused by factors such as biases in or insufficient training data, or even the model's wrong conclusions. To test if the AI in this work would provide false positives - that is, wrongly identify a crack-like pattern as if it was an actual crack -, we input a picture of a building façade with a porcelain marble effect tile (Figure 11). The AI successfully did not detect the presence of cracks in the image, since it stated "10% occurrence of crack", as presented in the pointed up black rectangle in Figure 11. We can then demonstrate how accurate the algorithm is to locate the existence of cracks in an image. The façade in Figure 11 also contains several lines and contours that could be confused for straight cracks, such as the frames of the glass windows and the interfaces between white and dark tiles. Even some plants on the façade and elements such as cables and cloud patterns reflected in the mirrored-glass elements could lead to misinterpretation of cracks if the model had insufficient accuracy.

Figure 11 – Building façade with porcelain marble effect tile and mirrored-glass elements.

4. Conclusions

The deep learning model presented in this paper is capable of accurately detecting the presence of cracks in images of civil construction elements. Concerning the AI architecture, through tests and model evaluation, we demonstrated the capability of U-Net's semantic segmentation and the classification efficiency of MobileNet to accurately identify cracks in various types of surfaces. The experiments demonstrate the efficiency of the algorithm for crack detection, potentially saving time and resources. The approach prioritizes efficiency and scalability, allowing for the timely identification of potential structural vulnerabilities without requiring meticulous boundary delineation. Therefore, the combined U-Net-MobileNet model achieved a balanced performance, showcasing high precision and highlighting the trade-offs between different metrics and the strengths of each model architecture for crack detection tasks.

Regarding applicability, civil construction professionals are capable of accurately identifying the presence of cracks in building elements, such as structural members, masonry, plastering, pavement, and tiles. In other words, the algorithm can be generally used to detect the presence of cracks in construction elements without presenting misleading results. Furthermore, the AI model is currently capable of detecting the presence of cracks, not entirely delineating them; that is, at the present, the model is meant to be applied as a detection tool, identifying the presence of cracks within surfaces rather than providing the precise contour of the cracks.

Future research directions may encompass refining the model's performance, exploring additional datasets, and broadening the application of deep learning methodologies to diverse aspects of infrastructure monitoring and maintenance. Namely, further implementations will allow the model to provide full contour, classify and identify the causes of cracks.

Acknowledgments

We specially thank the program to research incentive activity (PICI-UFERSA) for the financial support in granting a Scientific Initiation scholarship. We also thank both LOADS® and Cilab laboratory teams at UFERSA for all the support during this research.

References

Anitha, M. J., & Hemalatha, R. J. (2022). Efficient Crack Severity Level Classification Using Bilayer Detection for Building Structures. Computer Systems: Science & Engineering, 46(1), 1183-1200.

BANO, Samreen; JAISWAL, Ganesh; KUMAR, Rikshit; TIWARI, Avneesh; KARTHIKEYAN, M. (2023). Experimental Study on the Crack Repair Techniques of Concrete Structures: a case study. Iop Conference Series: Materials Science and Engineering, 1273(1), 12006.

Dais, D., Bal, I. E., Smyrou, E., & Sarhosis, V. (2021). Automatic crack classification and segmentation on masonry surfaces using convolutional neural networks and transfer learning. *Automation in Construction*, *125*, 103606.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

Hüthwohl, P., & Brilakis, I. (2018). Detecting healthy concrete surfaces. *Advanced Engineering Informatics*, *37*, 150-162.

Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., & Keutzer, K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. *arXiv preprint arXiv:1602.07360*.

Joshi, D., Singh, T. P., & Sharma, G. (2022). Automatic surface crack detection using segmentation-based deep-learning approach. *Engineering Fracture Mechanics*, *268*, 108467.

Khan, M. A.-M., Kee, S.-H., Pathan, A.-S.K., Nahid, A.-A. (2023). Image Processing Techniques for Concrete Crack Detection: A Scientometrics Literature Review. Remote Sensing, 15, 2400.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kim, B., Yuvaraj, N., Sri Preethaa, K. R., & Arun Pandian, R. (2021). Surface crack detection using deep learning with shallow CNN architecture for enhanced computation. *Neural Computing and Applications*, *33*(15), 9289-9305.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, *25*.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324.

Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision* (pp. 2980-2988).

Mignon, A., Snoeck, D., Dubruel, P., Van Vlierberghe, S., & De Belie, N. (2017). Crack mitigation in concrete: superabsorbent polymers as key to success?. *Materials*, *10*(3), 237.

Mohan, A., & Poobal, S. (2017). Crack detection using image processing: A critical review and analysis. Alexandria Engineering Journal, 57(2), 787-798.

Müller, A., Karathanasopoulos, N., Roth, C. C., & Mohr, D. (2021). Machine learning classifiers for surface crack detection in fracture experiments. International Journal of Mechanical Sciences, 209, 106698.

Özgenel, Çağlar Fırat (2019), "Concrete Crack Images for Classification", Mendeley Data, V2, doi: 10.17632/5y9wdsg2zt.2

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18* (pp. 234-241). Springer International Publishing.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9).

Tan, M., & Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning* (pp. 6105-6114).

Yamaguchi, T., Nakamura, S., Saegusa, R., & Hashimoto, S. (2008). Image‐based crack detection for real concrete surfaces. *IEEJ Transactions on Electrical and Electronic Engineering*, *3*(1), 128-135.

Yao, Y., Tung, S. T. E., & Glisic, B. (2014). Crack detection and characterization techniques—An overview. *Structural Control and Health Monitoring*, *21*(12), 1387-1413.