DETECTION OF CRACKS IN HIGH-RISE BUILDINGS USING DEEP LEARNING MODEL

Prashant Kumar Gangwar
Woldia Institute of Technology, Woldia University,Woldia, Ethiopia
e-mail: er.prashantgangwar@gmail.com

Abstract
This research work focuses on the detection and analysis of fractures, more specifically, crack detection and classification using residual learning as well as crack analysis through the application of image processing technology, which serves as the primary method of investigation throughout this study. This study was carried out to better understand how to detect and analyze fractures. The authors describe the results of an in-depth investigation into cracks that they carried out themselves. This work presents an image-based automatic crack identification model for analyzing the development of buildings after natural or man-made catastrophes. In addition to this, we compared the results obtained by U-Net and those obtained by object-based image analysis, which are both based on deep learning.

Keywords: Crack Detection, Deep Learning, U-Net, Construction Engineering

1. Introduction

When it comes to the construction of bridges and other kinds of civil infrastructure, concrete is by far the material of choice in every region of the globe [1]. The process of visually identifying cracks can be conceptualized as a presence/absence classification problem for cracks, which is the work that needs to be completed [2]-[3]. It has been noticed that there are two distinct types of methodological approaches while carrying out the procedure of autonomous visual crack identification. The first school of thought proposes that researchers start by extracting features from data and then move on to classifying the data using techniques derived from machine learning [4, 5].

In this type of research, features are first extracted by using adaptive filters, transformations, and/or morphological processes, and then they are used to separate images that contain cracks from images that do not contain cracks. In other words, fractures are detected by analyzing the differences between the two types of images [6] [7].

The accuracy ratings for crack detection are consistently well above 90% across the board in each one of these tests. However, the performance of the studies was impacted by a variety of factors, such as the selection of data, the number and type of layers used beyond convolutional layers, the choice of filter widths, and other variables [8]. Specifically, the convolutional layer performance was negatively impacted. It is not immediately apparent from these studies how the efficiency of these frameworks is affected by the extent of the picture dataset and the depth of the network [9].

The quality of a structure is also directly impacted by the materials used and techniques employed in its construction [10]. When a structure has a decreased bearing capacity, it is easy to produce cracks in it, and the quality of a structure is also directly impacted by the materials used and techniques employed in its construction.

The useful lives of structures such as dams, bridges, and pipelines will be significantly shortened due to these fractures, and the probability of accidents will increase as a result. As soon as the system breaks, there will be irreparable harm done to both the lives of individuals and the success of the company. Cracks, along with several other potential hazards, are one of the earliest obvious

indications of structural damage. The prompt localization of cracks [11, 12] prepares the way for the elimination of latent dangers that might have resulted in injury.
This study focuses on the detection and analysis of fractures, more specifically crack detection, and classification using residual learning as well as crack analysis through image processing technology, which serves as the primary method of investigation throughout this study. In addition to this, we compared the results obtained by U-Net and those obtained by object-based image analysis.

2. Related works

Deep learning techniques that rely on region recognition may employ either window sliding or region proposal as two of their methodology options. These techniques are utilized in order to ascertain a bounding box for each potential object that is visible within a image. This can be accomplished by referring to the image.
The authors in [11] used a combination of CNN and the sliding window technique to achieve increased accuracy in the classification of crack and non-crack images. This was accomplished by using both techniques together. The authors in [12] started out by removing noise with the help of an R-CNN model before moving on to the identification of fractures and fissures that had been patched up. R-CNN, on the other hand, is not a viable option for processing large quantities of images in an efficient manner because its processing method is dependent on window-sliding.
The use of conventional methods that are used to propose regions in images makes it difficult to select appropriate candidate regions from noisy images. This is because these methods are used to propose regions in images. The authors in [13] attempted to improve the computational effectiveness of region-based methods by employing parallel processing. Despite their efforts, this approach proved to be prohibitively expensive in terms of both the amount of computational time and the resources that were required.
The authors in [14] trained a more efficient R-CNN to examine images of concrete roadways and identify cracks within those images. By combining a faster region proposal convolutional neural network (Faster R-CNN) algorithm, a modified TuFF method, and a modified DTM for the crack identification and localization, The authors in [15] developed an integrated method to automate fracture location, segmentation, and quantification. This was accomplished by creating an integrated method to automate fracture location, segmentation, and quantification. Because of this, they were able to devise a technique that could automatically identify and pinpoint the location of cracks.
The authors in [16] identified sidewalk fractures by utilizing a model called a Faster-R-CNN. This was done to improve the accuracy of their findings. This was done in their study. In their paper, one of the applications that they described was exactly like this one.
The authors in [17] propose a responsible for making one such endeavor. However, despite utilizing this approach, the issue of decreasing resolution because of continuous pooling remained difficult to address.
The authors in [18] are responsible for the development of the automatic pavement detection system known as CrackNet. Convolutional neural networks are the foundation on which it is constructed. There are a total of five distinct phases that make up the CrackNet system. Feature maps make up the two levels of data that are utilized for use as input. The creation of these feature images was the work of the feature extractor. The projected image class ratings constitute a image output layer. These ratings are based on the image itself. The two obfuscated layers are made up of layers that are convoluted with one another and layers that are fully connected to one another. You can view all over a million of CrackNet characteristics on this page. These characteristics, which are taught to users during the learning process to increase the accuracy of crack extraction, are viewable here.
The authors in [19] suggest a spatial-channel hierarchical network (SCHNet) that is based on the Visual Geometry Group 19 in order to improve the accuracy and efficiency of crack identification. The authors of this study cite the Visual Geometry Group 19 as their foundation. (VGG19).
When it comes to planning measurements for applications, the authors believe that it is essential to have an understanding of both the benefits and the drawbacks of a measurement system, in addition to the model of the scanner that is being used. This is in addition to the fact that the authors believe it is

important to understand the scanner that is being used. To obtain measurements that are as accurate and useful as they possibly can be, it is essential to adjust the method of measurement so that it is appropriate for the specific circumstances of each crack.
The authors concluded that they needed to broaden their understanding of the topic as a result of the prevalence of wall cracks, the seriousness of the repercussions that result from having them, and the paucity of research on the constraints of the methodology that is currently being used. In this article, the authors describe the results of an in-depth investigation into cracks that they carried out themselves. Cavities, which are much larger than cracks, are much simpler to identify using deep learning technology.

3. Proposed Method

The completion of the first stage of the process, which involves obtaining video data of the building envelope, the data that was recorded is then processed by the contour detection algorithms. After the images have been preprocessed to eliminate noise through the use of the techniques of Gaussian blur and binarization, this step takes place. After that, the contour scanner analyzed the wall and the glass to locate any fractures or breaks. If a concrete block had fissures in it, and those fissures were more than 0.3 millimeters wide, then those fissures were considered to be cracks. However, it is essential to keep in mind that curved surfaces do not automatically point to the presence of injury. In addition, it was postulated that the wall construction of the building had a negligible impact on the temperature of the surface of the building that was exposed to the elements.

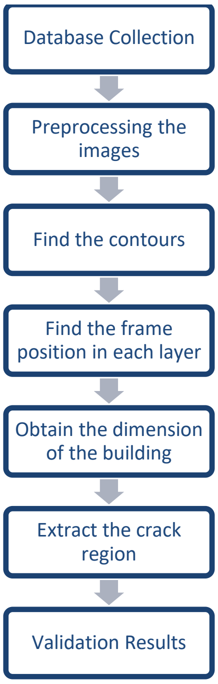

Figure 1: Proposed Modelling

The mode feed is another name for the information that is captured in its raw state by the digital camera. The preliminary image is then put through an image-thresholding technique, which employs the improved Otsu method that was suggested earlier in the process. Both the original Otsu algorithm as well as the M2GLD algorithm, both of which were covered in the section that came before this one, are components of the improved version of the Otsu technique that has been recommended. To get rid of the noncrack objects and noise pixels that were described in Figure 1,2,3, the process that is known as image concentrated effort, which is carried out after the process that is known as image penalization, is required.

It is generally agreed upon that the work presents an image-based automatic crack identification model for analyzing the development of buildings after natural or man-made catastrophes. An example of the author recommended approach to post-disaster building element analysis is demonstrated using a numerical experiment, which demonstrates the potential benefits of the approach. In addition, the search for fractures in the surface of the paved area is one of the aspects of concrete inspections that is considered to be of the utmost importance.

The cost of a complete road rehabilitation project can be reduced by as much as 80 percent if road cracks are located and repaired as soon as they appear. As a direct result of this, a wide variety of specific image processing strategies have been developed, most notably with the intention of locating cracks in asphalt flooring. The process of image processing is broken down into two stages: first, the region of the image that contains fewer than a predetermined number of pixels (N) is eliminated, and then an axis is added to the image length of an object.

$aRI = LM/LN$, LM- major axis length and LN – minor axis length.

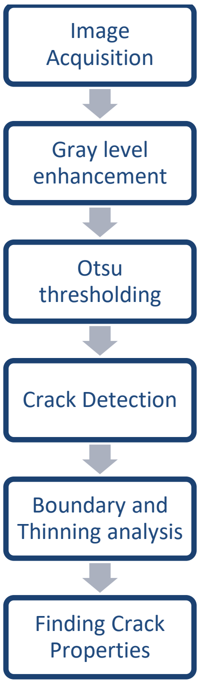

Figure 2: Pre-processing and Feature Extraction

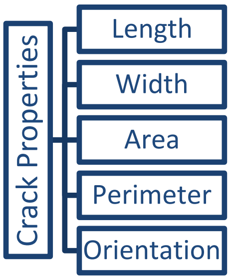

Figure 3: Crack Properties

Problem Definition

Medical image segmentation is the process of using computer vision and image analysis techniques to evaluate and process two-dimensional (2D) or three-dimensional (3D) images to segment, extract, reconstitute, and display human organs, soft tissues, and diseased bodies in three dimensions. This is done to facilitate medical diagnosis and treatment. These methods are utilized in the analysis and processing of two-dimensional or three-dimensional images to segment, extract, and display diseased bodies in three dimensions.

The image is divided into many different sections according to the degree to which different parts of the image have similarities to one another or differ from one another. It is possible for medical professionals to perform qualitative or quantitative analysis on lesions and other regions of interest by utilizing this technique, which significantly improves the accuracy of medical diagnosis and the dependability of the outcomes of treatment. The image cells themselves, along with the structures and organs that are associated with them, have been the most common kinds of object types up until this point in time.

The process of segmenting an image involves obtaining a division of the image such that:

$$\bigcup_{\substack{x=1 \\ R_x = I}}^{N} R_X \cap R_y = \phi \ \forall x \neq y; x, y \in [1, N]$$

where Rx satisfies both sets of all pixels in communication similarity constraint Ci (i = 1, 2,...), i.e. the image areas.

In addition to that, the x and y coordinates are used so that the various parts of Ry can be distinguished from one another. The number of sub-regions is denoted by the positive number N, which must be greater than or equal to 2, and it must be greater than or equal to 2. There are several distinct phases involved in the process of dividing medical images into their respective categories.

Step 1: The very first action that needs to be done is the collection of an imaging data collection. A training set, a validation set, and a test set should typically be included in a data compilation like this one.

Step 2: During the second stage, deep learning is utilized for the image processing, and the data collection is partitioned into three distinct sections.

Step 3: Third, to ensure the best possible performance, the network model is trained using the training set, verified using the verification set, and evaluated using the test set. This is done to guarantee the best possible results.

Step 4: The image goes through preprocessing and expansion during the fourth stage of the process. In most cases, this is accomplished by conducting standardization on the input image, random rotation on the input image, and random scaling on the input image to make the data set significantly larger.

Step 5: to segment the medical image, step 5 requires the application of an appropriate technique, which is then followed by the production of the segmented images. Ultimately, the goal of this step is to produce a segmented image.

Step 6: An analysis of how accurately the predictions were made constitutes the sixth stage in the process.

Step 7: The seventh step in the process entails establishing confirmed effective performance indicators for the efficacy of medical image segmentation.

Network Model Training

During the process of training the deep CNN model, both the labeled photograph as well as the 8-band Worldview image were employed. The purpose of this research required the extraction of image squares, each of which had a resolution of 256 pixels along each side. The tiles that had their boundaries lopped off in an arbitrary fashion were the ones that had the greatest potential to be utilized as training datasets. Better segmentation outcomes may be achieved through the provision of additional training data, which in turn assists in the strengthening of the network training.

Data augmentations such as mirroring, spinning, enhanced luminosity, and the addition of noise points were applied to the tiles to bring out more of the data unique characteristics. This was done to make the tiles more useful. This was done with the purpose of bringing out more of the prospective applications that could be made with the data. Throughout the course of the experiments, a learning rate of 0.01 was utilized to train the model over the course of 30 epochs, with 16 groups being allocated to each epoch. This was done to achieve the best possible results.

To train the models, 70% of the structures in the training sites were chosen at random from the training dataset, and 30% were chosen from the validation dataset. The remaining 10% of the structures were chosen from the test dataset. Because of this, we were able to utilize 70 percent of the structures at the training locations. The utilization of the period number guaranteed that both the accuracy and convergence of loss were preserved throughout the process.

As its optimization approach, the optimizer for this experiment made use of a method that is known as stochastic gradient descent. The algorithm has control over the learning rate, and the impact that it has on the convergence of the learned network is a function of that rate. A weight decay penalty in the amount of 0.00001 was added to the loss function so that the network model would not become overfit. This was done with the intention of preventing the model from becoming too accurate.

The value of momentum, which is defined in this context as the percentage of time that the model continues to update in the same fashion, has a predetermined range of 0.9. This range was predetermined before the value of momentum was calculated. To provide further clarity, the following cross-entropy was utilized in this training session as a loss function to evaluate the efficiency of the training method that was being utilized.

$$L_{loss} = -\frac{1}{N}\sum_{i=1}^{N} y_i \log \hat{y}_i - (1-y_i)\log(1-\hat{y}_i)$$

where

y_i - actual label and

$\hat{y}_i$ - predicted label.

Tensorflow and Keras are two Python-based deep learning frameworks. These frameworks were used to facilitate the implementation of the U-Net design as well as the entire semantic segmentation process. Both frameworks were written in Python. to carry out all of the other processing and analysis, open source frameworks and packages were employed. The operating system that was installed on the virtual computer was Ubuntu 16.04.4, and it had a random-access memory (RAM) capacity of 16 terabytes. The graphics processing unit (GPU) that was used was an NVIDIA Geforce GTX 1080.

Accuracy Assessment

With the assistance of the uncertainty matrix, the efficiency of the U-Net-based semantic segmentation that was investigated in this study was evaluated. To compute the OA, we began by taking the image that had been labeled as a reference and adding together the percentage of accurately identified pixels in each of the categories.

$$OA = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k}\sum_{j=0}^{k} p_{ij}}$$

where

p_{ii} - pixels for i class,

p_{ij} - pixels of i class recognized incorrectly as j, and

k - categories.

The F1-score, which is dependent on the precision and recall, can be quickly calculated and applied to the evaluation of the semantic segmentation by using the confusion matrix. This is possible because the F1-score is dependent on the precisions and recalls.

Because the F1 score is dependent on both metrics, it is feasible to accomplish this goal. The numerous confusion matrices that can be applied to the structure differentiation process are presented in Table 1. The precise numbers are listed in the table columns, while the rows contain various estimations of those numbers.

The F1-score is calculated by utilizing to determine how accurate a model is when taking into consideration both its precision and its recall. This is done to determine the overall quality of the model. It is feasible to obtain an average of the F1-score values for each of the different classes by performing calculations that are very similar to those used for segmentation and multi-class classification.

$$\text{Recall} = \frac{TP}{TP + FN}$$

The recall is the proportion of observations in a real class that are confirmed as affirmative, represented as a percentage of the total number of observations. The recall is the proportion of observations in a real class that are confirmed as negative. Accuracy is measured by looking at the proportion of accurately predicted positive observations in relation to the total number of such observations. The F1-score is a one-of-a-kind metric that takes into consideration not only the accuracy of the answers but also their recall.

$$\text{Precision} = \frac{TP}{TP + FP}$$

A measurement that can be used to evaluate and contrast two or more sets of data is called the Jaccard Index. This index is also known as the percentage that represents the intersection and union of two groups. (IoU). Both types of groups are referred to as the prediction group and the reference group, respectively, when discussing image semantic segmentation as a field of study. The procedure of image segmentation is carried out, and the results are then incorporated into an algorithm for the purpose of determining the IoU for each class.

$$F1\_score = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

where

A - prediction and

B - ground truth.

We determined the overall accuracy, the F1-score, and the Intersection-over-Union value to evaluate the efficacy of deep learning for semantic segmentation employing U-Net. Additionally, we displayed an intuitive visualization of the predicted results in comparison with the ground truth. In addition to this, we compared the results obtained by U-Net, as well as those obtained by object-based image analysis.

$$IoU = \frac{|A \cap B|}{|A \cup B|}$$

4. Results and Discussions

When shooting with drones, it can be difficult to obtain many images of cracks in the surface of buildings made of concrete, such as industrial buildings, houses, and other types of structures, because the dataset for the research is the surface of these buildings. This is since cracks are not typical targets. To making this process as streamlined as it possibly can be, we take photographs with our phones and use the cracks in the pavement as a backdrop. There were over a thousand cracked images that were recorded, each of which had a distinctive and unique combination of the choices.

We identify the cracks by using an app called labelme, which is a free and open-source segmentation assignment annotation app that is hosted on GitHub. This app is used to annotate the cracks with the appropriate segmentation. Annotation, in contrast to the process of locating targets, entails nothing more than identifying the surroundings of the target and putting it into one of several categories. The annotation stage of the segmentation process is more challenging than the other parts of the work because it requires marking each individual point along the target shape.

The gap is not particularly wide, and there is not much in the way of patterning to be found in it either. In contrast to targets that take the form of individuals or vehicles, objectives of this type almost invariably receive a score that is lower than 10 points. To outline the shape of a complex crack, you will typically need at least 20 points, as doing so requires a lot of time and effort on your part. In most instances, however, you will only need 10 points.

We divided the crack data set into 750 sets for training and 250 sets for assessment as a component of the training process. The dataset only contains a single designated category that is referred to as crack, and the category that is used as the default is the category that is known as background. Because there is no other choice for the label, all the images that contain more than one crack are simply filed under the crack category. This is because there is no other option for the label.

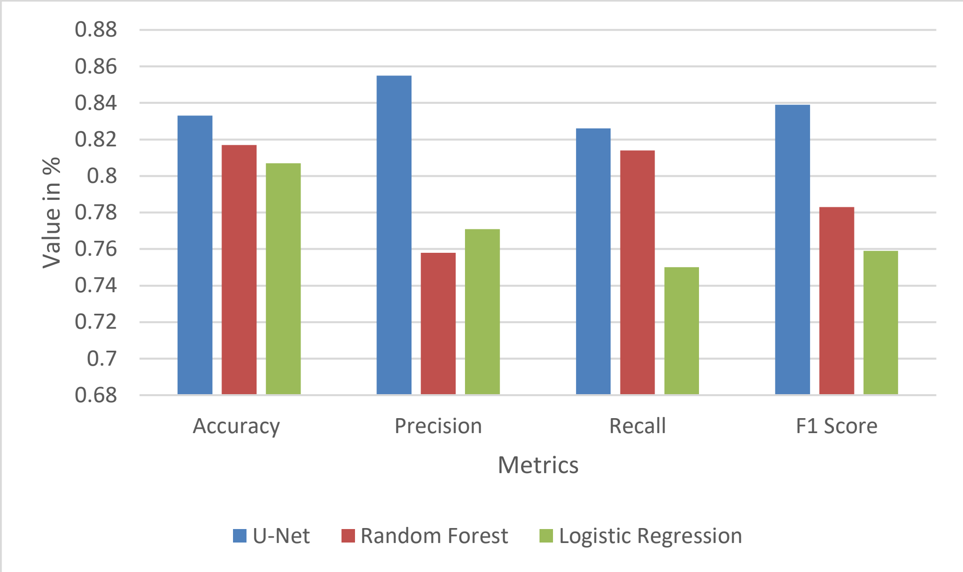

Figure 4: Training Metrics

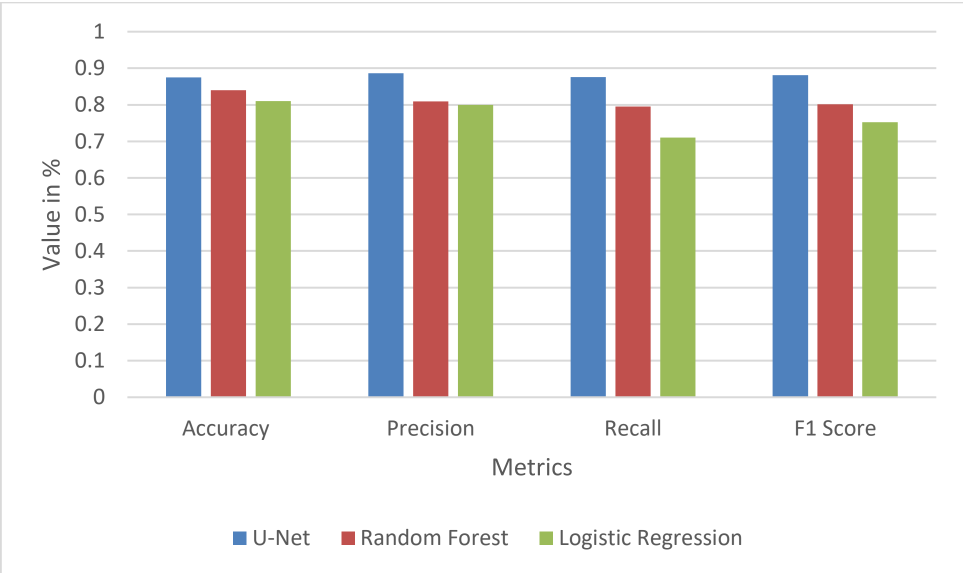

Figure 5: Testing Metrics

Figure 6: Validation Metrics

We will primarily introduce four sets of comparative experiments, which include: an experiment comparing different parameter values; an experiment comparing results obtained before and after optimizing the loss function; an experiment comparing results obtained with a traditional segmentation model; and an experiment comparing results obtained with some preexisting crack detection networks. These four sets of experiments are as follows: an experiment comparing different parameter values; an experiment comparing results obtained before and after optimizing the loss function; an experiment comparing results obtained with some existing technique.
After about 50 iterations, the error values in the training set and the verification set have a propensity to converge and become stable. This happens regardless of whether the iterations are repeated or not. Training results in network models that have optimal network weights. These weights are discovered by locating the weights that produce the smallest error value and the weights that produce the closest error value between the training set and the verification set, respectively. Training results in network models that have optimal network weights. These two different sets of weights are contrasted with one another to determine which of the two yields the superior result.
To accomplish control and management of the engineering output as well as the quality of the work that is done by all participants, it is necessary to take part in a full-scale mobilization and to naturally integrate all of the organization components. Only then will it be possible to achieve these goals. The quality management system within the construction company needs to be improved on a regular basis, and all parties involved in the conception, design, execution, and completion of the project need to place quality at the top of their list of priorities to ensure that it is met. This includes the planning, machinery, production, quality inspection, materials, logistics, and documentation teams.
As a consequence of this, to perform exhaustive quality control, one needs to have a comprehensive understanding of each and every aspect of project quality as well as the work that makes use of these aspects.
This could result in cost savings by shortening the route length that a drone needs to travel to conduct a wall inspection. This is because a high-resolution camera can capture images from a greater distance from the wall while still maintaining the information it needs to identify cracks.

5. Conclusions

This study focuses on the detection and analysis of fractures, more specifically crack detection, and classification using residual learning as well as crack analysis through the application of image processing technology, which serves as the primary method of investigation throughout this study. This study was carried out to better understand how to detect and analyze fractures. Traditional crack detection techniques and those based on deep learning are compared in this paper. The technique is selected for application because it is better adapted to the requirements of the scenario.

The results of this study were compared with those of traditional crack detection techniques and those based on deep learning in this paper. It was found that there is a need for research into the optimum regularity of data capture to guarantee the timely discovery of cracks while simultaneously minimizing the expenses of both operational and data storage activities.

Even though the findings of the study are encouraging, additional research needs to be done to investigate the expenses of setting up, implementing, and running deep learning software if it is used in the future for breach detection. This is necessary if it is the case that the software will be used. In addition, there is a need for research into the optimum regularity of data capture to guarantee the timely discovery of cracks while simultaneously minimizing the expenses of both operational and data storage activities. Finally, additional research is necessary to select the most effective deep learning algorithm. This research should consider a variety of variables, including the type of structure, the data that is readily available, and so on.

References

[1] Li, S., Gu, X., Xu, X., Xu, D., Zhang, T., Liu, Z., & Dong, Q. (2021). Detection of concealed cracks from ground penetrating radar images based on deep learning algorithm. *Construction and Building Materials*, *273*, 121949.

[2] 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*, 9289-9305.

[3] Chun, P. J., Yamane, T., & Tsuzuki, Y. (2021). Automatic detection of cracks in asphalt pavement using deep learning to overcome weaknesses in images and GIS visualization. *Applied Sciences*, *11*(3), 892.

[4] Zhang, Q., Barri, K., Babanajad, S. K., & Alavi, A. H. (2021). Real-time detection of cracks on concrete bridge decks using deep learning in the frequency domain. *Engineering*, *7*(12), 1786-1796.

[5] Hu, G. X., Hu, B. L., Yang, Z., Huang, L., & Li, P. (2021). Pavement crack detection method based on deep learning models. *Wireless Communications and Mobile Computing*, *2021*, 1-13.

[6] Jiang, S., & Zhang, J. (2020). Real-time crack assessment using deep neural networks with wall-climbing unmanned aerial system. *Computer-Aided Civil and Infrastructure Engineering*, *35*(6), 549-564.

[7] Le, T. T., Nguyen, V. H., & Le, M. V. (2021). Development of deep learning model for the recognition of cracks on concrete surfaces. *Applied computational intelligence and soft computing*, *2021*, 1-10.

[8] Alipour, M., & Harris, D. K. (2020). Increasing the robustness of material-specific deep learning models for crack detection across different materials. *Engineering Structures*, *206*, 110157.

[9] Le, T. T., Nguyen, V. H., & Le, M. V. (2021). Development of deep learning model for the recognition of cracks on concrete surfaces. *Applied computational intelligence and soft computing*, *2021*, 1-10.

[10] Hsieh, Y. A., & Tsai, Y. J. (2020). Machine learning for crack detection: Review and model performance comparison. *Journal of Computing in Civil Engineering*, *34*(5), 04020038.

[11] Bibi, R., Saeed, Y., Zeb, A., Ghazal, T. M., Rahman, T., Said, R. A., ... & Khan, M. A. (2021). Edge AI-based automated detection and classification of road anomalies in VANET using deep learning. *Computational intelligence and neuroscience*, *2021*, 1-16.

[12] Elghaish, F., Talebi, S., Abdellatef, E., Matarneh, S. T., Hosseini, M. R., Wu, S., ... & Nguyen, T. Q. (2022). Developing a new deep learning CNN model to detect and classify highway cracks. *Journal of Engineering, Design and Technology*, *20*(4), 993-1014.

[13] Bhowmick, S., Nagarajaiah, S., & Veeraraghavan, A. (2020). Vision and deep learning-based algorithms to detect and quantify cracks on concrete surfaces from UAV videos. *Sensors*, *20*(21), 6299.

[14] Aslam, Y., Santhi, N., Ramasamy, N., & Ramar, K. (2021). Localization and segmentation of metal cracks using deep learning. *Journal of Ambient Intelligence and Humanized Computing*, *12*, 4205-4213.

[15] Guan, J., Yang, X., Ding, L., Cheng, X., Lee, V. C., & Jin, C. (2021). Automated pixel-level pavement distress detection based on stereo vision and deep learning. *Automation in Construction*, *129*, 103788.

[16] Rezaie, A., Achanta, R., Godio, M., & Beyer, K. (2020). Comparison of crack segmentation using digital image correlation measurements and deep learning. *Construction and Building Materials*, *261*, 120474.

[17] Xu, X., & Yang, H. (2020). Vision measurement of tunnel structures with robust modelling and deep learning algorithms. *Sensors*, *20*(17), 4945.

[18] Mohtasham Khani, M., Vahidnia, S., Ghasemzadeh, L., Ozturk, Y. E., Yuvalaklioglu, M., Akin, S., & Ure, N. K. (2020). Deep-learning-based crack detection with applications for the structural health monitoring of gas turbines. *Structural Health Monitoring*, *19*(5), 1440-1452.

[19] Arya, D., Maeda, H., Ghosh, S. K., Toshniwal, D., & Sekimoto, Y. (2021). RDD2020: An annotated image dataset for automatic road damage detection using deep learning. *Data in brief*, *36*, 107133.