Journal of Theoretical and Applied Information Technology
31st August 2022. Vol.100. No 16
© 2022 Little Lion Scientific

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

AUTONOMOUS ROBOT SYSTEM FOR PAVEMENT CRACK INSPECTION BASED CNN MODEL

ALAA SHETA[1], SAHAR A. MOKHTAR[2]
[1]Professor, Southern Connecticut State University, Department of Computer Science, New Haven, CT, USA
[2]Researcher, Electronics Research Institute, Department of Computers and Systems, Cairo, Egypt

*Email: [1]shetaa1@southernct.edu, [2]sahar@eri.sci.eg

ABSTRACT

Maintaining the excellent state of the road is critical to secure driving and is an obligation of both transportation and regulatory maintenance authorities. For a safe driving environment, it is essential to inspect road surfaces for defects or degradation frequently. This process is found to be labor-intensive and necessitates primary expertise. Therefore, it is challenging to examine road cracks visually; thus, we must effectively employ computer visualization and robotics tools to support this mission. This research provides our initial idea of simulating an Autonomous Robot System (ARS) to perform pavement assessments. The ARS for crack inspection is a camera-equipped mobile robot (i.e., an Android phone) to collect images on the road. The proposed system is simulated using an mBot robot armed with an Android phone that gathers video streams to be processed on a server that has a pre-training Convolutional Neural Networks (CNN) that can recognize crack existence. The proposed CNN model attained 99.0% accuracy in the training case and 97.5% in the testing case. The results of this research are suitable for application with a commercial mobile robot as an autonomous platform for pavement inspections.

Keywords: *Autonomous Robot System, Deep Neural Network, Road Maintenance, Crack Detection, Pavement Crack, Automatic Detection.*

1. INTRODUCTION

Although Potholes might not seem a big deal, it is too expensive to fix. Recent statistics have suggested that potholes cost American drivers more than 6 billion dollars annually. Fixing potholes is nearly $35 to $50 per pothole. Add to this cost an initial mobilization of about $100 to $150 to bring trucks and crew to the repair site [5]. A report from TRIP, a transportation research group by Matt Spillane, on Nov. 14, 2018, stated that poor road conditions cost NYC-area drivers $2,800 a year.

Pavement inspections are a routine highway process, and various authorities implement local airport roads. Pavement authorities are liable for inspecting cracks, flatness, and distress. The most common method presented in the past was visual inspections. A regular pavement distress survey corresponds to a pre-defined strategy to monitor the pavement condition and standardize the proper maintenance and rehabilitation activities. Various types of distress can be examined, such as longitudinal cracking, crosswise cracking, chunk cracking, and patching [6]. The pavement inspection demands to be executed in numerous weather and traffic situations; thus, it is time-consuming and entails human errors.

Automating the inspection processes using robotics has been explored in [7]. The principal goal of numerous research articles was to build an autonomous robot capable of automating pavement inspection. The Autonomous Robot System (ARS) design can deliver great productivity in road inspection in a considerably better, faster, and safer manner. This process will reduce costs and human mistakes.

This work aims to provide a simulation-oriented pavement crack detection system using robotics. The proposed method can be simulated in a laboratory for education and research purposes. The system design concerned several components: recognizing pavement cracks, vehicle control, vision system, and data storage for designing self-driving vehicles. The main idea of this research is to simulate an Autonomous Robot System (ARS) to perform pavement assessments. The ARS for crack inspection is a camera-equipped mobile robot-like an Android phone, to collect images on the road. The proposed system is simulated using a mBot robot with an Android phone that gathers video streams to be processed on a server with pre-training Convolutional Neural Networks (CNN) that can recognize crack existence. The Convolution Neural Networks (CNNs) are utilized to classify the structure of the cracks from raw images collected by the Android mobile camera. The results of this research are suitable for application with a commercial mobile robot as an autonomous platform for pavement inspections.

The proposed autonomous system has a variety of integrated components, each with a specific task to produce an automated process for pavement crack detection. The process was simulated using the mBot robot and a MATLAB-generated code.

The paper is organized as follows: Section 2 provides background work on pavement crack inspection with a literature review on CNN in Section 3. Section 4 presents the proposed ARS framework for pavement crack detection. We also provide the functionality of each system component. The architecture of the CNN neural network and the setup for the experimental design is provided in the experimental setup Section 5. In Section 6, the system simulation and experimental results are provided.

2 BACKGROUND

Automated crack detection using computer imaging is a subject receiving massive investigation over the last few decades. Many practical applications of this technology have evolved in the commercial domain. For example, Fugro Roadware [8] recommends commercially available automated road condition detection systems that leverage image processing technology. The proposed system can detect cracks as small as 1*mm* [9]. The Pavement Management Service (PMS) commissioned an Automatic Road Analyzer (ARAN) vehicle equipped with a Laser Crack Measurement System (LCMS). The LCMS unit can identify a broader spectrum of defects simultaneously. This ability to gather a richer data set offers substantial financial gains in the pavement monitoring procedure. In Figure 1, we show the LCMS payload system vehicle.

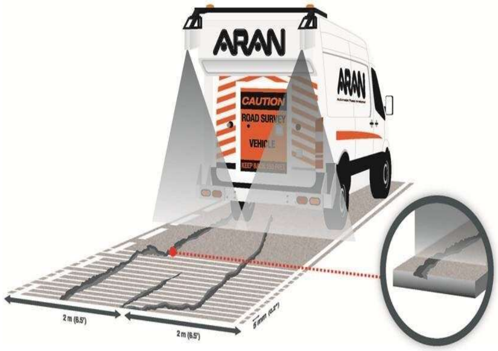


Figure.1. Laser Crack Measurement System Mounted On Mobile Car

Autonomous robot bridge scanning is reforming the procedure of bridge examination [10]. Nevertheless, one of the significant challenges is finding a method that can accurately interpret the enormous, collected images to understand bridge status. In [11], the authors presented a unique automated crack detection system based on a state-of-the-art robotic bridge scanning system, as given in Figure 2 [11].

An automated examination system-based mobile robot for tunnel crack detection was presented in [12]. The proposed mobile robot system was planned to maintain a constant distance from walls during the image collection using a Charge-Coupled Device (CCD) camera. An image processing approach was adopted to recognize cracks in tunnel walls. Detection of cracks in pipelines was also presented in [13]. Detecting cracks in buried pipes is essential in measuring the degree of pipe weakening for community and service operators. In [14], the authors proposed a mobile manipulator system for bridge crack inspection. This mission utilized a mobile vehicle's two CCD cameras and a four-axis manipulator system. Parallel cameras are used to detect cracks. The manipulator system is equipped with binocular CCD for assessing constructions that may not be straightforward to the eye.

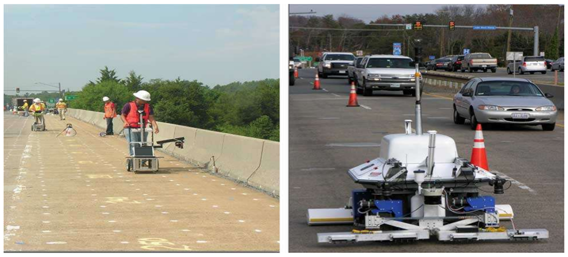


Figure 2. Testing site for bridge assessment at the Haymarket, VA, USA. (Right) RABIT - Robotic Evaluation Bridge Inspection Tool By The Federal

Highway Administration (FHWA) [15]. The Image Was Presented In [11].

In [16], the authors suggested an inspection system using a mobile robot with a camera that can gather images of the bridge deck cracks. Using camera calibration and robot localization, they utilized an image processing technique based on the Laplacian of Gaussian (LoG) algorithm to develop a crack map. A Genetic Algorithm (GA) was used to create an optimal path for the robot. GA helped find an optimal way that decreases the total turns and the moving distance. A possible layout of the crack inspection and mapping system named "Robotic Crack Inspection and Mapping (ROCIM) system" is given in Figure 3. The ROCIM system is positioned to examine the bridge lane by lane.

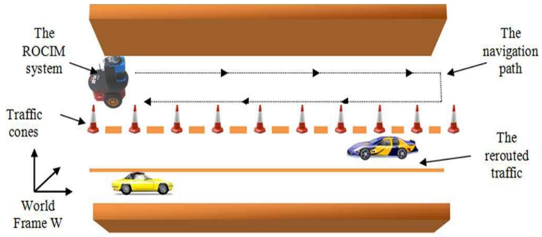


Figure 3. The Scenario Of Crack Inspection And Mapping Using The ROCIM System As Presented In [16]

Sophisticated technologies aided in monitoring and protecting the pavement. Recently flying drones [17] were also explored to monitor pavement. In [17], the authors provided a two-stage approach. In the first stage, images are collected using unmanned aerial vehicles (UAVs). Later a 3D model of the data is created using laser scanners. The second stage includes clustering the collected images using histogram analysis and peak detection. Possible cracks are identified using adaptive thresholds. In [18], the authors provided a multi-spectral pavement image analysis method was provided. A UAV was used to recognize pavement damages (e.g., cracks and potholes) using several machine learning algorithms, including a support vector machine, artificial neural network, and random forest.

In many research articles, CNN was effectively used to classify pavement cracks [19, 20]. In [19], the authors used Principal Component Analysis (PCA) method for features extraction with many PCA components. A CNN was trained with 400 images of road cracks with a 480×480 resolution. The achieved results show how PCA helped speed up CNN's learning process. In [21], the author's utilized image features automatically from hand-annotated image patches acquired by a low-cost sensor, i.e., a smartphone. CNN showed superior crack detection performance. A grid-based pavement crack detection using deep learning was presented in [22]. A pixel-wise classification of surface cracks was produced in [23]. A deep crack end-to-end trainable deep CNN for automatic crack detection was suggested in [24].

Several researchers explored the use of CNN for other applications, such as manufacturing process modeling [25]. For example, CNN's have been used for hand gesture recognition [26], handwritten digits recognition [27], smart agriculture [28], facial recognition [29], lung cancer classification [30], identification of abnormalities in ECG [31], classifying Thai Fast Food (TFF) [32], etc.

3 CONVOLUTIONAL NEURAL NETWORKS

A convolution neural network is based on Artificial Neural Networks (ANNs). It is a set of algorithms modeled after a human brain that are designed to recognize patterns [33,34]. The patterns they recognize are numerical, in which all real-world data, whether it be images, must be translated. ANN helps cluster and classify data. The word deep in deep learning denotes the increased number of layers in the network [35]. The proposed CNN model consists of three layers. Each layer contains convolution, max pooling, and RuLU components, and a fully connected layer is presented as a final layer for classification purposes. Convolutional neural networks have shown promising results recently.

3.1 Convolutional Layer

The convolution layer applies a filter matrix to the input. The filter matrix is a set of weights determined by the network multiplied by the elements under the filter. The sum of the resulting products is the value of the output element. In this example, the filter matrix is of size 3×3 (See Figure 4). For each component of the first 3×3 area of the input, the element is multiplied by the corresponding weight. In this example, the sum of the products is -3. This becomes the new element of the output matrix. The stride value then shifts the filter matrix. The filter will be shifted to the right by the column stride until it reaches the edge. Afterward, the filter will be moved down by the value of the row stride.

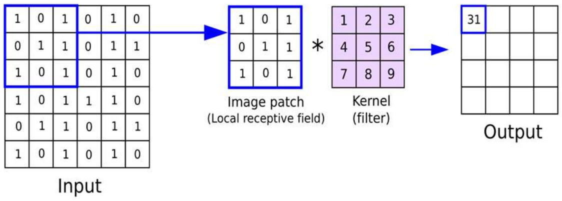


Figure 4. Convolution layer

3.2 Pooling Layer

The Pool layers operate to attain dimensional reduction on each feature map but keep the essential information. The depth of output and input feature maps are identical. Meanwhile, the size of each feature map (i.e., kernel) scales down, corresponding to the size of the sub-sampling window, as shown in Figure 5. Sub-sampling functions include average or maximum. Max-pooling is the most popular because the rate of change in the maximum value is very low compared to the rate of change of the average value of any receptive field (the window of the input feature map at which pooling is applied). Pooling operation helps extract a combination of features invariant to transnational shifts and minor distortions [36]. Also, pooling may reduce the over-fitting of the training data.

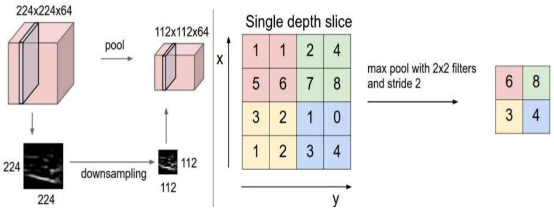


Figure 5. Pooling layer

3.3 ReLU Activation Layer

Activation functions are used in CNN to control the node output. The function is connected to each neuron to determine whether or not the outcome should be active. Activation functions must be simple computationally since they have computed each data sample thousands or millions of times. The ReLU is a piece-wise linear function. It outputs the same input if it's positive and zeroes for negative values. The ReLU function is computationally efficient and has no vanishing gradient problem for positive inputs. Still, the gradient becomes zero for negative inputs, and the network cannot perform back-propagation and cannot learn. This issue is called the dying ReLU. Equations 1 and 2 demonstrate the computations of ReLU.

$$R(z) = \max(0, z) \qquad (1)$$

$$R(z) = \begin{cases} 0 \; if \; z \leq 0 \\ z \; if \; z > 0 \end{cases} \qquad (2)$$

ReLU permeates the problem of diminishing gradients and speeds up the learning process. ReLU is considerably advised for designing CNN classification models [37].

4 PROPOSED ARS FRAMEWORK

The Architecture of the proposed ARS is shown in Figure 6. The system consists of three main components; 1) the Robot system following a path (i.e., the road), 2) the mobile device with an Internet Protocol camera or IP camera to locate cracks, and 3) the CNN for the recognition of pavement crack. The ARS setup is shown in Figure 7.

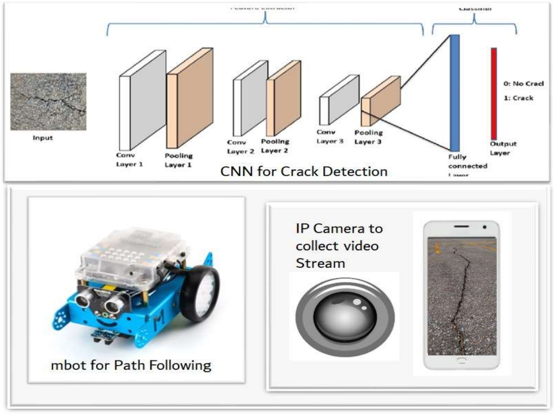


Figure 6. Proposed ARS Components

4.1 mBot Robot System

mBot is a low-cost, easy-to-run robot kit that can be controlled using a graphical programming software called Make blocks, founded in 2013. It was effectively used in teaching robotics to science, technology, engineering, and mathematics (STEM) students. mBot is an all-in-one solution to experience the hands-on experience of programming, electronics, and robotics. The mBot has several sensors that can be utilized for various applications such as obstacle avoidance, line-following, and many others. The mBot enjoys several features such as essay assembling, Arduino open-source platform, and supports IOS & Android Applications. It can also be programmed using Scratch. mBot can be controlled via Bluetooth or 2.4 GHz wireless modules

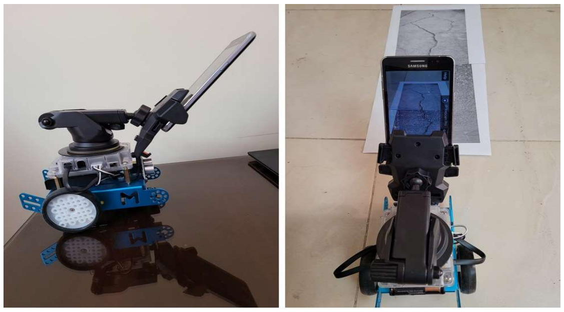


Figure 7 Proposed ARS system prototype for pavement crack detection

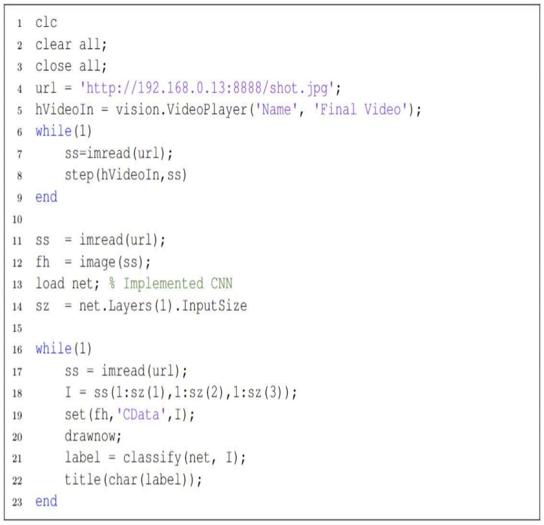
4.2. Mobile Phone IP Camera

An Internet Protocol camera, or IP camera, is a digital video camera that collects data and sends image data via an IP network. They are frequently used for surveillance; they need no local recording device, only a local area network. The IP camera software is application software that can be utilized for home surveillance, business, and family protection. This software can be installed on a mobile phone and allows the phone to collect video streams for motion or object detection. The software can be broadcast via an IP address fed to a website and processed for various aspects. Numerous Video Management Software (VMS) facilitates an easy view of recorded webcams and live streams. Several IP camera software is available in the Google App store [38]. To collect the stream of video from the IP camera and process it using the CNN model, we used the code presented in Figure 8.

5. EXPERIMENTAL SETUP

5.1 Crack Image Dataset

This research adopted a concrete crack dataset containing concrete images with/without cracks. Each class has 20000 images that have a 227×227 resolution [39,40]. No data augmentation was adopted



```
clc
clear all;
close all;
url = 'http://192.168.0.13:8888/shot.jpg';
hVideoIn = vision.VideoPlayer('Name', 'Final Video');
while(1)
    ss=imread(url);
    step(hVideoIn,ss)
end

ss  = imread(url);
fh  = image(ss);
load net; % Implemented CNN
sz  = net.Layers(1).InputSize

while(1)
    ss = imread(url);
    I = ss(1:sz(1),1:sz(2),1:sz(3));
    set(fh,'CData',I);
    drawnow;
    label = classify(net, I);
    title(char(label));
end
```

Figure 8. Matlab Code For Collecting Video Stream From The Ip Camera To The Server To Classify Pavement Cracks.

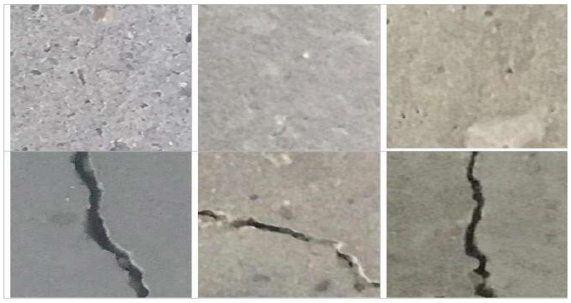


Figure 9. Concrete Crack Images for Classification. Upper row: Images with no cracks, Lower row: cracks (Source [39,40])

5.2 Proposed CNN Model

In this research, we proposed a CNN with 16 layers. The architecture of the CNN is provided in Table 1. Figure 10 shows the design of the CNN. This model was drawn via the use of the MATLAB Deep Learning App [41]. As shown in Table 1, the input image to the CNN is with size 227×227 with three channels representing red, green, and blue shades of any image (RGB). This image entered the first convolution layer that contains eight filters with a size of 5×5; many channels equal three (equal the number of channels input feature map (input image)). Stride equals one horizontally and one vertically. The output feature map from the convolution layer is entered Max-pooling layer with window size 2×2, stride two horizontally and two vertically, and Zero-padding on the four sides of the 2D channel. After the Max-pooling layer, Batch normalization to the 16 channels is performed to minimize the internal covariate shift and smoothen the objective function of the CNN, which results in improving the network performance. The ReLU activation layer is the fifth layer to determine which neurons should be activated in the next convolution layer to maximize the efficiency of the CNN. There are two other sets of these layers (convolution layer, Max-pooling with zero padding layer, batch normalization layer, and finally RELU activation layer) till layer number 13, but with a different number of filters in each convolution layer. The 14*th* layer contains 9 Fully connected layers in which all the neurons are connected with the weight of each neuron. After a fully connected layer, the soft Max-pooling is applied to the whole channels to get only a 1 × 1 window in each channel (nine channels equal to several classes from which we classify the input image). The last layer is to transfer the resulted element in each channel (nine channels) to a percentage that indicates how much the input image belongs to the corresponding class.

5.3. Evaluation Metrics

In this research, we adopted several metrics that include accuracy, precision, and recall classifying various pavements for whether they have a crack or not. They include:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+F} \quad (3)$$

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

$$Recall = \frac{TP}{TP+FN} \quad (5)$$

where TP, FP, TN and FN are the true positive, false positive, true negative, and false3 negative, respectively.

$$Sensitivity = Recall = \frac{TP}{TP+FN} \quad (6)$$

$$Specificity = \frac{TN}{FP+TN} \quad (7)$$

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (8)$$

6. EXPERIMENTAL RESULTS

In our experiments, we used 1000 images, with 80% for training and 20% for testing. Figure 12 illustrates the confusion matrix for the training case in which 800 images are used for training where 400 images include cracks (class 0) and 400 images without cracks (class 1). As shown in Figure 12, 408 images are trained to contain cracks with TP = 400 images while FN = 8 images and 392 images are trained with no cracks with FP = 392 images and TN = 0. In Table 2, we calculated the accuracy of training at 99%, sensitivity (and Recall function) at 98.04%, specificity at 100%, and precision at 100% in the training case. The confusion matrix in the testing case is presented in Figure 13 with 200 images [100 images include cracks (class 0) and 100 images without cracks (class 1)]. As shown in Figure 13, 99 images are predicted to contain cracks with TP = 96 images while FN = 2 images, and 99 images are predicted to be free of cracks with FP = 97 images and TN = 2 images. In Table 2, we calculated the accuracy, sensitivity, specificity, and precision at 97.5%, 97.03%, 97.97%, and 98%, respectively. Figure 11 shows both the accuracy and loss during training and validation of the CNN versus the number of iterations. As the number of iterations increases, loss during training and testing decreases dramatically. Also, the accuracy of training and validation increases.

7. CONCLUSION

This paper provides our initial idea of developing an Autonomous Robot System for pavement crack inspection. The proposed ARS system consists of three components, the robot, the CNN, and the IP camera. The training CNN was able to classify pavement crack images using a stream of video collected from an Android phone camera. The CNN produces a classification accuracy of over 97%. Additionally, the proposed ARS allows the robot to process video stream data in real time and generate an accurate classification. The proposed system can be effortlessly improved by equipping it with different sensors.

8. FUTURE WORK

In future work, the following research points shall be investigated 1) Explore the mBot extra capabilities to change its path and direction throughout the crack detection process, 2) To deploy the used CNN architecture on an FPGA (Field Programmable Gate Array) as an embedded system with the mBot robot to save the required time for communication between the IP-Camera and the server at which CNN architecture is uploaded, 3) Increase the accuracy by extending the current system to a multi-agent system through an increasing number of robots to get more images with different views and then get more accurate decisions through the whole seen of the roads and streets.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

9. REFERENCES

[1] NHTSA. National motor vehicle crash causation survey report to congress. [Online]. Available: http://www-nrd.nhtsa.dot.gov/Pubs/811059

[2] J. Blumgart. (2021) The u.s. needs to fx existing roads, not build new ones. [Online]. Available: https://www.governing.com/now/the-u-s-needs-to-fx-existing-roads-not-build-new-ones

[3] J. Lee, "Effects of pavement surface condition on traffc crash severity," *Journal of Transportation Engineering*, 05 2015.

[4] Wikipedia. (2007) Mississippi river bridge. [Online].Available:http://en.wikipedia.org/wiki/i-35w

[5] SealMaster. (2022) How much does it cost to fix a pothole? [Online]. Available: https://sealmaster.net/faq/much-cost-fx-pothole/

[6] S. Saat, A. R. M. Kamil, M. Z. M. Tumari, and A. S. R. A. Subki, "Development of an autonomous robot for inspection system," in *2018 IEEE 14th International Colloquium on Signal Processing Its Applications (CSPA)*, March 2018, pp. 272-276.

[7] T. Le, S. Gibb, N. Pham, H. M. La, L. Falk, and T. Berendsen, "Autonomous robotic system using non-destructive evaluation methods for bridge deck inspection," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 3672–3677.

[8] "Roadware: Equipment and software," 2020. [Online]. Available: https://www.fugro.com/our-services/asset-integrity/roadware/equipment-and-software

[9] J. L. Groeger, P. Stephanos, P. Dorsey, and M. Chapman, "Implementation of automated network-level crack detection processes maryland," *Transportation Research Record*, vol.1860, no. 1, pp. 109–116, 2003.

[10] H. M. La, R. S. Lim, B. Basily, N. Gucunski, J. Yi, A. Maher, F. A. Romero, and H. Parvardeh, "Autonomous robotic system for high-efficiency non-destructive bridge deck inspection and evaluation," in *2013 IEEE International Conference on Automation Science and Engineering(CASE)*, 2013, pp. 1053–1058.

[11] P. Prasanna, K. J. Dana, N. Gucunski, B. B. Basily, H. M. La, R. S. Lim, and H. Parvardeh, "Automated crack detection on concrete bridges," *IEEE Transactions on Automation Scienceand Engineering*, vol. 13, no. 2, pp. 591–599, 2016.

[12] S. Yu, J.-H. Jang, and C.-S. Han, "Auto inspection system using a mobile robot for detecting concrete cracks in a tunnel," *Automation in Construction*, vol. 16, pp. 255–261, 05 2007.

[13] S. K. Sinha and P. Fieguth, "Automated detection of crack defects in buried concrete pipe images," *Automation in Construction*, vol. 15, pp. 58 – 72, 2006.

[14] P.-C. Tung, Y.-R. Hwang, and M.-C. Wu, "The development of a mobile manipulator imaging system for bridge crack inspection," *Automation in Construction*, vol. 11, pp. 717–729, 2002.

[15] N. Gucunski, S. Kruschwitz, R. Feldmann, and C. Rascoe, "Meeting ltbp program objectives through periodical bridge condition monitoring by nondestructive evaluation," 04 2009, pp.1–10.

[16] R. S. Lim, H. M. La, and W. Sheng, "A robotic crack inspection and mapping system for bridge deck maintenance," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no.2, pp. 367–378, 2014.

[17] M. D. Phung, T. H. Dinh, V. T. Hoang, and Q. Ha, "Automatic crack detection in built infrastructure using unmanned aerial vehicles," *Proceedings of the 34th International Symposium on Automation and Robotics in Construction (ISARC)*, Jul 2017.

[18] Y. Pan, X. Zhang, G. Cervone, and L. Yang, "Detection of asphalt pavement potholes and cracks based on the unmanned aerial vehicle multispectral imagery," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 10, pp. 3701–3712, 2018.

[19] E. Endri, A. Sheta, and H. Turabieh, "Road damage detection utilizing convolution neural network and principal component analysis," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 6, 2020.

[20] A. Sheta, H. Turabieh, S. Aljahdali, and A. Alangari, "Pavement crack detection using convolutional neural network," in *Proceedings of 35th International Conference on Computers and Their Applications*, ser. EPiC Series in Computing, G. Lee and Y. Jin, Eds., vol. 69, 2020, pp.214–223.

[21] L. Zhang, F. Yang, Y. Daniel Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," in *2016 IEEE International Conference on Image Processing (ICIP)*, Sep. 2016, pp. 3708–3712.

[22] X. Wang and Z. Hu, "Grid-based pavement crack analysis using deep learning," in *2017 4th*

International Conference on Transportation Information and Safety (ICTIS), Aug 2017, pp.917–924.

[23] M. David Jenkins, T. A. Carr, M. I. Iglesias, T. Buggy, and G. Morison, “A deep convolutional neural network for semantic pixel-wise segmentation of road and pavement surface cracks,” in *2018 26th European Signal Processing Conference (EUSIPCO)*, Sep. 2018, pp. 2120–2124.

[24] Q. Zou, Z. Zhang, Q. Li, X. Qi, Q. Wang, and S. Wang, “Deepcrack: Learning hierarchical convolutional features for crack detection,” *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1498–1512, March 2019.

[25] W. Deabes, A. Sheta, and M. Braik, “Ect-lstm-rnn: An electrical capacitance tomography model-based long short-term memory recurrent neural networks for conductive materials,”*IEEE Access*, vol. 9, pp. 76 325–76 339, 2021.

[26] J. Nagi, F. Ducatelle, G. A. Di Caro, D. Cires¸an, U. Meier, A. Giusti, F. Nagi, J. Schmidhuber, and L. M. Gambardella, “Max-pooling convolutional neural networks for vision-based hand gesture recognition,” in *2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, Nov 2011, pp. 342–347.

[27] S. Pratt, A. Ochoa, M. Yadav, A. Sheta, and M. Eldefrawy, “Handwritten digits recognition using convolution neural networks,” *J. Comput. Sci. Coll.*, vol. 34, no. 5, p. 40–46, apr 2019.

[28] L. Ale, A. Sheta, L. Li, Y. Wang, and N. Zhang, “Deep learning based plant disease detection for smart agriculture,” *2019 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–6, 2019.

[29] J. Hiebert, F. Mazhar, M. Derosa, and A. Sheta, “Facial recognition using a lightweight deep neural networks,” *Journal of Advanced Computer Science Technology*, vol. 10, no. 1, pp. 1–8,2021.

[30] T. Matsubara, T. Ochiai, M. Hayashida, T. Akutsu, and J. Nacher, “Convolutional neural network approach to lung cancer classifcation integrating protein interaction network and gene expression profles,” in *2018 IEEE 18th International Conference on Bioinformatics and Bioengineering (BIBE)*, Oct 2018, pp. 151–154.

[31] Z. Liu, X. Meng, J. Cui, Z. Huang, and J. Wu, “Automatic identifcation of abnormalities in 12-lead ecgs using expert features and convolutional neural networks,” in *2018 International Conference on Sensor Networks and Signal Processing (SNSP)*, Oct 2018, pp. 163–167.

[32] N. Hnoohom and S. Yuenyong, “Thai fast food image classifcation using deep learning,” in *2018 International ECTI Northern Section Conference on Electrical, Electronics, Computerand Telecommunications Engineering (ECTI-NCON)*, Feb 2018, pp. 116–119.

[33] L. Shao, D. Wu, and X. Li, “Learning deep and wide: A spectral method for learning deep networks,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 12, pp. 2303–2308, Dec 2014.

[34] G. Younes, C. Hadda, N. Attia, and Z. Djelloul, “Supervised learning and automatic recognition of asphalt pavement deteriorations,” in *2009 IEEE/ACS International Conference on Computer Systems and Applications*, May 2009, pp. 205–210.

[35] H. Yi, S. Shiyu, D. Xiusheng, and C. Zhigang., “A study on deep neural networks framework,” in *2016 IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, Oct 2016, pp. 1519–1522.

[36] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, “A survey of the recent architectures of deep convolutional neural networks,” *Artifcial intelligence review*, vol. 53, no. 8, pp. 5455-5516, 2020.

[37] V. Nair and G. E. Hinton, “Rectifed linear units improve restricted boltzmann machines,” in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp.807–814.

[38] A. Authority, “5 best home security apps and ip camera apps for android, ”2022, accessed = 2022-04-13. [Online]. Available: https://www.androidauthority.com/best-home-security-apps-android-842997/

[39] L. Zhang, F. Yang, Y. D. Zhang, and Y. J. Zhu, “Road crack detection using deep convolutional neural network,” *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 3708–3712, 2016.

[40] F. Ozgenel and A. G. Sorguc¸, “Performance comparison of pretrained convolutional neural ¨ 29 networks on crack detection in buildings,” in *Proceedings of the 35th International Symposium on Automation and Robotics in Construction (ISARC)*, J. Teizer, Ed. Taipei, Taiwan: International Association for Automation and Robotics in Construction (IAARC), July 2018, pp.693–700.

[41] MATLAB, *(R2019b)*. Natick, Massachusetts: The MathWorks Inc., 2019.

Table 1. A Detailed Description Of The Layers Of The Proposed CNN Architecture.

Layer	Name	Type	Details
1	’imageinput’	Image Input	227×227×3 images with ’zerocenter’ normalization
2	’conv 1’	Convolution	8 5×5×3 convolutions with stride [1 1] and padding ’same’
3	’maxpool 1’	Max Pooling	2×2 max pooling with stride [2 2] and padding [0 0 0 0]
4	’batchnorm 1’	Batch Normalization	Batch normalization with 16 channels
5	’relu 1’	ReLU	ReLU
6	’conv 2’	Convolution	16 5×5×16 convolutions with stride [1 1] and padding ’same’
7	’maxpool 2’	Max Pooling	2×2 max pooling with stride [2 2] and padding [0 0 0 0]
8	’batchnorm 2’	Batch Normalization	Batch normalization with 16 channels
9	’relu 2’	ReLU	ReLU
10	’conv 3’	Convolution	32 5×5×16 convolutions with stride [1 1] and padding ’same’
11	’maxpool 3’	Max Pooling	2×2 max pooling with stride [2 2] and padding [0 0 0 0]
12	’batchnorm 3’	Batch Normalization	Batch normalization with 32 channels
13	’relu 3’	ReLU	ReLU
14	’fc’	Fully Connected	9 fully connected layer
15	’softmax’	Softmax	softmax
16	’classoutput	Classification Output	crossentropyex with ’Black Sea Sprat’ and 8 other classes

Table 2. Evaluation Criteria

Case	Accuracy	Sensitivity	Specificity	Orecision	Recall	F_1
Training	0.9900	0.9804	1.0000	1.0000	0.9804	0.9901
Testing	0.9750	0.9703	0.9798	0.9800	0.9703	0.9751

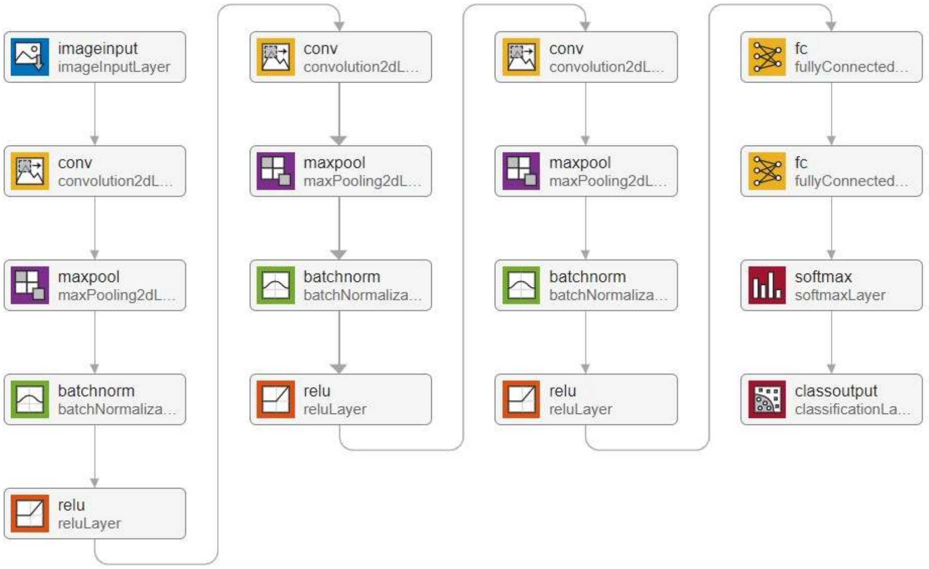


Figure 10: Description Of The CNN Layers Based Matlab

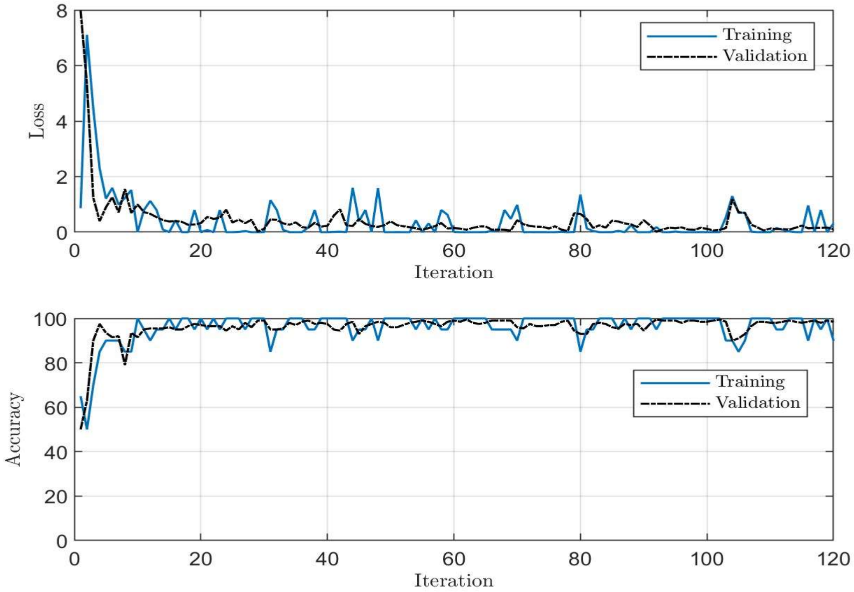
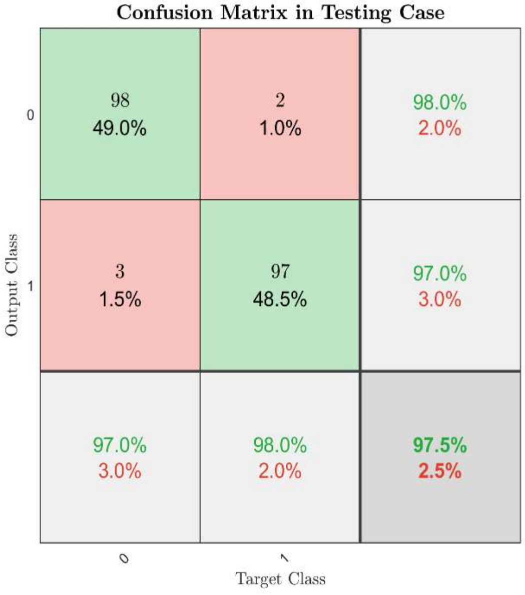


Figure 11: Loss And Accuracy Of The Trained CNN Model



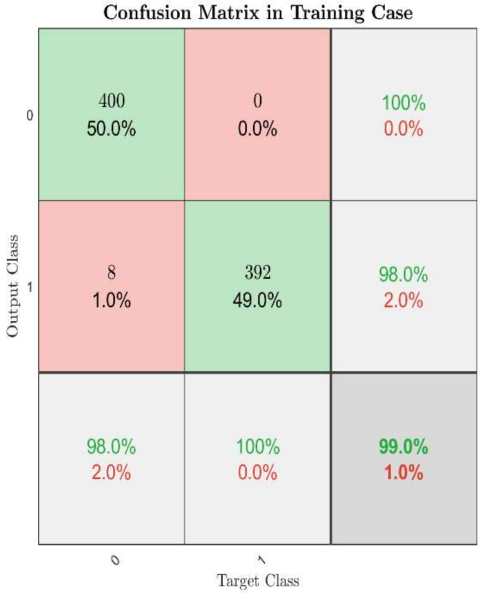


Figure 12: Training Case-Confusion Matrix.

Figure 13: Testing Case-Confusion Matrix