International Journal on Perceptive and Cognitive Computing (IJPCC) Vol 9, Issue 1 (2023)
https://doi.org/10.31436/ijpcc.v9i1.372

Real-Time Automated Road Damages Inspection Using Deep Convolutional Neural Networks

Mohamad Faiq Mohd Shahrul Munir, Muhammad Amiruddin Bustamam, Amelia Ritahani Ismail*, Norlia Md-Yusof , Amir 'Aatieff Amir Hussin
Kulliyyah of Information and Communication Technology (KICT), International Islamic University Malaysia.

*Corresponding author: amelia@iium.edu.my
(Received: 1st January 2022; Accepted: 19th January 2023; Published on-line: 28th January 2023)

Abstract— This study focused on developing a real time automated road damage inspection using deep neural networks. The performance of the build image detection model is evaluated to get the best overall result. Thousands of images from selected dataset are trained using You Only Look Once (YOLO) v4 algorithm which based on Convolutional Neural Network (CNN). The model is deployed into smartphones to take advantage of its availability camera. The road damage inspection application (app) can help the road users and municipalities in inspecting the road surface. Thus, it can prevent heavy damages to a vehicle and help in providing a better road damage maintenance management.

Keywords— image detection, Convolutional Neural Network, deep learning, YOLO,

I. Introduction

Road surface inspection traditionally depends on humans' visual observations which tends to be inconsistent and unsustainable that eventually can increase the risk associated with aging road infrastructure. Aside from requiring experienced road managers, it also can be time consuming. On the other hand, quantitative analysis using widely utilised expensive machines such as mobile-measurement system (MMS) or laser-scanning method though highly accurate, it is considerably expensive for small municipalities that are short of financial resources.

The number of road traffic deaths continues to rise steadily, reaching 1.35 million in 2016. Between 20 and 50 million more people suffer non-fatal injuries, with many incurring a disability as a result of their injury [1]. By 2018, Malaysia has the third highest fatality rate from road traffic accidents in Asia and Asean. Road traffic accidents (5.4%) were the fourth most common cause of death in Malaysia in 2016 [2]. In normal conditions, experienced licensed drivers have the capability to make a right decision in split seconds to prevent traffic accidents. Unfortunately, that is not always the case as the road traffic environments are often susceptible to adverse causes such as road surface conditions. Furthermore, the current road surface visual inspection not only requires certain officials, but the process is very time consuming. Aside from being highly accurate, the inspection devices are still considerably expensive. Thus, considering the issues, several attempts have been made to develop a method for analyzing road properties using image processing technologies to inspect a road surface. However, such road damage detection

methods only focus on the determination of the existence of the damage. Hence, a practical damage detection model is necessary to distinguish and detect different types of road damage.

The objective of this study is to propose a real-time automated road damage inspection. It is an application to automatically recognize various types of road damages that cause harm and damage to road users and vehicles in real-time. Road damage image dataset is trained using end-to-end object detection algorithms i.e. YOLO based on CNNs.

II. Related Work

Three main artificial intelligence (AI) approaches to detect crack are image processing, machine learning and deep learning (DL) [3], [4]. We are interested to extend the research using the latter approach. Two general tasks for crack detection are crack classification and segmentation. In this study, the first task is our focus.

[5] divides four methods of crack classification using DL, i.e. deep convolutional neural networks (DCNN) [6], [7], [8], modified CNN [9], hybridization CNN and object detection models [10], [11]. The techniques for object detection models includes the Single Shot Multibox Detector (SSD), the region-based fully convolutional networks (R-FCN), the You Look Only Once (YOLO) and the Faster R-CNN. YOLO is the first-choice technique for a researcher to classify a detected crack. This is due to its accuracy and efficiency [4]. We adopted this technique in our study likewise.

Furthermore, [12] has prepared a large-scale road damage dataset composed of 9,053 images captured via a smartphone attached on a car, with 15,435 instances of road surface. The detection model is then trained using Single Shot Multibox Detector (SSD) Inception V2 and SSD MobileNet frameworks to be executed on a small computational device like a smartphone. The damage type finally can be classified with high accuracy.

CrackNet, an efficient architecture based on CNN has been proposed by [13], which trained using 1,800 3D pavement images for automated pavement crack detection on 3D asphalt surfaces with the objective of pixel-perfect accuracy. CrackNet is then demonstrated to be successful in recognizing cracks on asphalt surfaces under various conditions using a different set of 200 3D images with high Precision (90.13%), recall (87.63%) and F-measure (88.86%). CrackNet is then compared with two other crack detection methods, 3D shadow modeling, an optimization of the projection angle (61 degree) used in the model and Pixel-SVM, a pixel level classification using SVM-based method. The results are shown Table 1.

TABLE 1

OVERALL PRECISION, RECALL AND F-MEASURE OF CRACKNET, 3D SHADOW MODELLING AND PIXEL-SVW

Method	Precision (%)	Recall (%)	F-measure (%)
Pixel – SVM	51.28	82.90	63.36
3D shadow modelling	66.84	77.19	71.64
CrackNet	90.13	87.63	88.86

Work done by [14] are proposed to classify the road damage based on the types, but it is only limited to road cracks type of damage namely linear, transverse and crocodile. Figure 1 shows the example of class label.

In this study, image classification methods are applied to input images rather than object detection. It is important to mention that classification refers to assigning a class label to an image, whereas detection means identifying the object's coordinates in an image and assigning them a class label. Deep learning approach specifically the CNN has been applied to classify and detect an object class. A graphical image annotation tools, labelling was used for data labelling and annotation during the pre-processing. The tool then automatically converted bounding-box positions drawn around object and attached label class into YOLO data format in XML file. For modelling, YOLO algorithm received images as inputs, split them into S x S grid and pass them through the neural network to create bounding boxes and class predictions to determine the final detection output.

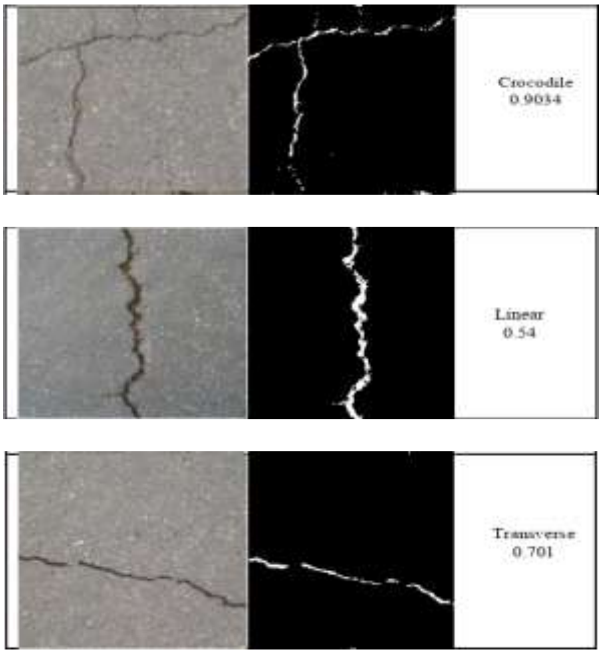

Fig. 1 Example of experimental results of road defect classes

YOLO v4 is used to train the road damage detection model based on the collected dataset. Results of the study will be compared with the previous studies in terms of accuracy and runtime speed. YOLO is a neural network that has the capability to classify and detect objects in given images [15]. It can detect multiple objects at a time and applies bounding boxes around the detected objects. Initially, DarkNet is used to implement YOLO [16]. It is an open-source neural networks training framework written in C/CUDA and serves as the basis for YOLO.

III. METHODOLOGY

Real-Time automated Road Damages Inspection Using Deep Convolutional Networks is a project that developed through deep learning. This road image detection model is built using YOLO algorithm which is based on DCNN. Later, the model is deployed into smartphone for a real-time road image detection using smartphone camera and processor.

The methodology of this study complied with the general machine learning pipeline as illustrated in Figure 2 in building an object detection model.

https://doi.org/10.31436/ijpcc.v9i1.372

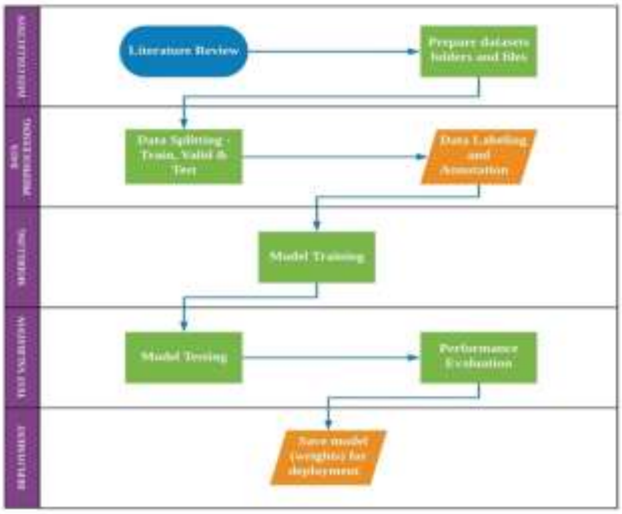

Fig. 2 Methodology Flowchart - Machine Learning Pipeline

A. Data Collection

The first phase of this study is by doing research and pilot study to build an object detection model to localize and classify the damage on asphalt road surface using the dataset provided by [12]. The dataset consists of 9,053 images that were captured on a smartphone (LG Nexus 5X) installed on a dashboard of a car. The images were taken in seven municipalities in Japan i.e. Adachi, Chiba, Ichihara, Muroran, Nagakute, Numazu and Sumida with diverse weather and illuminance with the resolution of 600 × 600 pixels.

B. Data Pre-processing

In the datasets, the road damage types are divided into eight categories. Each damage type is represented with a class name such as D40. Each type of damage is illustrated in the Table II.

TABLE II
ROAD DAMAGE TYPES AND THEIR CLASS NAME

Damage Type			Detail	Class Name
Crack	Linear Crack	Longitudinal	Wheel mark part	D00
			Construction joint part	D01
		Lateral	Equal interval	D10
			Construction joint part	D11
	Alligator Crack		Partial pavement, overall pavement	D20
Other Corruption			Rutting, bump, pothole, separation	D40
			White line blur	D43
			Cross walk blur	D44

Each image is then manually annotated using labelling a graphical image annotation tool by [17]. The annotated image file is a TXT file for YOLO data format and a XML file in PASCAL Visual Object Classes (PASCAL VOC) data format [18]. During the annotation, a bounding-box is drawn around an object in each image. Then, the class label is attached as shown in Figure 3.

Fig. 3 Bounding box are drawn to label the objects

Data exploration is done to statistically analyse the distribution of the type of road damages in the dataset. This is necessary to prevent class imbalance that will affect the model training. Based on Figure 4, the difference in number of images of road damages for each type is significant. Category D01 comes as the highest with the total of 3789 images followed by class D44 with 56 images short whereas class D40 recorded as the lowest with only 409 images.

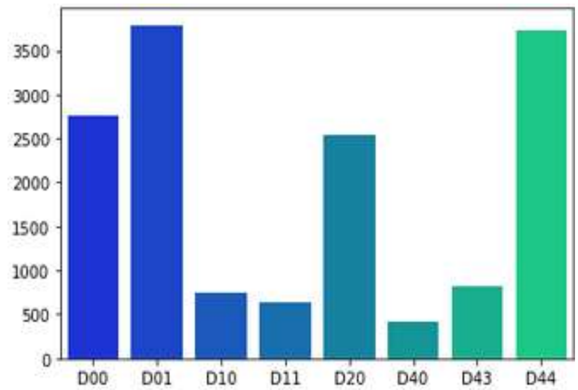

Fig. 4 Data distribution for the classes of road damages

C. Modelling

The initial model we use for this pilot study is trained using YOLO v4 object detection algorithm based on CNN. The training model is built with the help of DarkNet neural network training framework and Google Colab tools. These

https://doi.org/10.31436/ijpcc.v9i1.372

are the configurations and hyperparameters used for the training are shown in Table III.

TABLE III
YOLO TRAINING PARAMETER CONFIGURATIONS

Learning rate	Start at 0.001	Momentum	0.949
Batch size	64	Max batches	16000
Subdivisions	16	Filters	39
Resolution	416 x 416		

Figure 5 illustrates mean average precision (mAP) and loss versus iterations of training. After 1000 iterations, the training model is only capable of achieving the mAP value of 9.2% with average loss of 2.2326, which is the object model detection model still not good enough for detecting different types of road damages accurately.

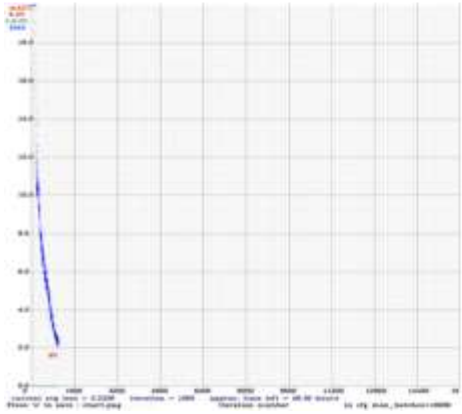

Fig. 5 mAP and loss vs iterations graph YOLO

The performance of the model built is tested. Details in section E.

E. Deployment

Road Damage Detection Application is the product of this study once the model is deployed into a smartphone.

1) *Machine Specification*: In this study, Google Colab cloud machine is used with the following specification: (i) Ubuntu 18.04.3 LTS (64 bit) as operating system, (ii) 13 GB Memory, (iii) Intel Xeon CPU v2 @ 2.3Ghz processor, (iv) Nvidia Tesla K80 GPU,(v) 60 GB Disk. Python programming language has been used in Road Damage Inspection app development. Python libraries used for the system implementation are ElementTree, minidom, matplotlib, seaborn, NumPy, TensorFlow, zipfile, pyplot and cv2.

2) *User Interface*: The interface of the system consists of basic image detection feature, report submission page and map tracking interface. The details of the interface are explained as follows.

The Homepage is the main page of the apps. From the homepage, the user can redirect to another pages. The user can start capturing road damage images. The screenshot of Home Page is shown in Figure 6.

Fig. 6 Homepage Interface

Live Capture page where the image detection process will be working. User can capture the image of the potholes and the apps will do the road damage detection. The screenshot of Live Capture is shown in Figure 7.

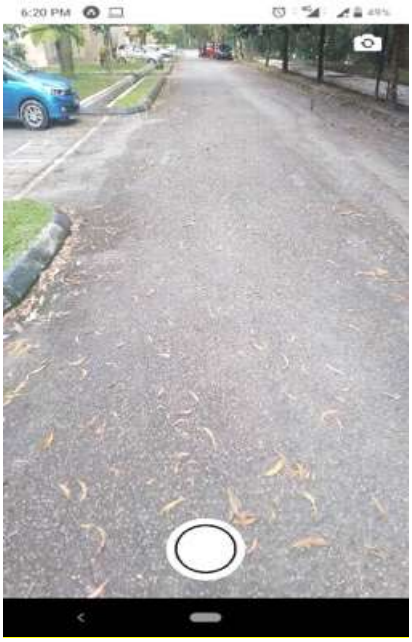

Fig. 7 Live Capture Interface

https://doi.org/10.31436/ijpcc.v9i1.372

Report submission page is the page for reporting the road damage to the municipalities. The user needs to fill in details and select the image of the detected potholes. The screenshot of Report Submission Page is shown in Figure 8.

Fig. 8 Report Submission Interface

Map Tracking shows nearby the location and picture of nearby road damage. The screenshot of Map Tracking is shown in Figure 9.

Fig. 9 Map Tracking Interface

IV. Result And Discussion

Based on Table IV, we can arrive at the reasoning that the model only can detect class D20 with high precision and barely detect class D43. One possible explanation for the low scores in every evaluation metric is the model needs to be trained more than initial 1000 iterations. The uneven object classes distribution also might be one of the factors to this poor performance.

TABLE IV
Average Precision For Every Classes And Evaluation Metrics Of Training Model

Classes	Average precision
D00	0.00%
D01	0.00%
D10	0.00%
D11	0.00%
D20	66.7%
D40	0.00%
D43	7.14%
D44	0.00%

Metrics	Score
Precision	0.20
Recall	0.14
F1-Score	0.17
Average IoU	13.38%

The train weight is imported and used to detect road damages in images from the test dataset. The inference time is recorded to ensure the damages detected in real-time. The results are shown in the Table V.

TABLE V
Detected Samples Using YOLO

Detection	Inference time	Confidence
	98.763 ms	• D01: 68% • D11: 66% • D01: 98% • D01: 99%
	98.772 ms	• D01: 82% • D01: 57%
	99.086 ms	• D44: 26%

V. Conclusions And Future Work

Real-time Road damage inspection is an app that utilized the image detection technology for road surface damage detection like potholes. Deep convolutional neural network-based algorithm i.e. YOLO is applied to build the model and later deployed into smartphone. Human visuals tend to be inconsistent and unsustainable that eventually can increase the risk associated with aging road infrastructure, thus the Road Damage Detection apps can help with that. It also can help municipalities in managing road damage management as the community can report the road damages with proof images. This paper has demonstrated how the Road Damage Detection App functions and how it can benefit the society.

For the future work, the integration with navigation app such as google maps or waze would be useful. Additionally, integration with APIs that related to the app also make the app more interactive. For example, voice alert can help the driver if the driver's eyes need focus or AI assistant that we can communicate with. Some automated features maybe can enhance the apps further. Whenever the camera detected road damages, warning will be automatically prompt, and the data may automatically send to the municipality. Advanced technology in modern car also opens a lot of potential in these automated features.

ACKNOWLEDGMENT

The authors would like to thank the Kulliyyah of ICT IIUM

CONFLICT OF INTEREST

The authors declare that there is no conflict of Interest

REFERENCES

[1] (2022). Road traffic injuries. [Online]. Available https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries
[2] (2018). Global status report on road safety. Available from https://www.who.int/violence_injury_prevention/road_safety_status/2018
[3] S.D. Nguyen, T.S. Tran, V.P. Tran, H. J. Lee, M. J. Piran, and V. P. Le, "Deep Learning-Based Crack Detection: A Survey," *International Journal of Pavement Research and Technology*, pp. 1-25, 2022.
[4] N. Ma, J. Fan, W. Wang, J. Wu, Y. Jiang, L. Xie, and R. Fan, "Computer vision for road imaging and pothole detection: A state-of-the-art review of systems and algorithms," *Transportation Safety and Environment,* pp. 1-16, 2022.
[5] L. Ali, F. Alnajjar, W. Khan, M.A. Serhani, H. Al Jassmi, "Bibliometric Analysis and Review of Deep Learning-Based Crack Detection Literature Published between 2010 and 2022," *Buildings,* vol. 12(4) p. 432, 2022.
[6] J. Fan, M. J. Bocus, L. Wang, and R. Fan, Deep convolutional neural networks for road crack detection: Qualitative and quantitative comparisons," *IEEE International Conference on Imaging Systems and Techniques (IST),* pp. 1-6, August 2021.
[7] C. V. Dung, "Autonomous concrete crack detection using deep fully convolutional neural network," *Automation in Construction,* vol. 99, pp. 52-58, Mar. 2019.
[8] V. Mandal, L. Uong, and Y. Adu-Gyamfi, "Automated road crack detection using deep convolutional neural networks," *IEEE International Conference on Big Data,* pp. 5212-5215, Dec. 2018.
[9] M. Naddaf-Sh, S. Hosseini, J. Zhang, N. A. Brake, and H. Zargarzadeh, "Real-time Road crack mapping using an optimized convolutional neural network," *Complexity,* pp.1-17, Sep 2019.
[10] R. Ali, D. Kang, G. Suh, and Y. J. Cha, "Real-time multiple damage mapping using autonomous UAV and deep faster region-based neural networks for GPS-denied structures," *Automation in Construction,* vol. 130, p. 103831, Oct. 2021.
[11] L. Ale, N. Zhang, and L. Li, "Road damage detection using RetinaNet," *IEEE International Conference on Big Data,* pp. 5197-5200, Dec. 2018.
[12] H. Maeda, Y. Sekimoto, T. Seto, T. Kashiyama, and H. Omata, "Road damage detection using deep neural networks with images captured through a smartphone," *arXivpreprint arXiv:1801.09454*, Jan. 2018.
[13] A. Zhang, K. C. Wang, B. Li, E. Yang, X. Dai, Y. Peng and C. Chen, "Automated pixel-level pavement crack detection on 3D asphalt surfaces using a deep-learning network," *Computer-Aided Civil and Infrastructure Engineering,* vol. 32(10), pp. 805-819, Oct. 2017.
[14] B. Akarsu, M. Karaköse, K. Parlak, E. Akin, and A. Sarimaden, "A Fast and Adaptive Road Defect Detection Approach Using Computer Vision with Real Time Implementation," *International Journal of Applied Mathematics, Electronics and Computers,* Special Issue-1, pp. 290-290, 2016.
[15] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv preprint arXiv:2004.10934,* Apr.2020.
[16] J. Redmon. (2016) Darknet: Open-source neural networks in C. [Online]. Available: https://pjreddie.com/darknet/
[17] L. Tzuta. (2015). LabelImg: Git code. [Online]. Available: https://github.com/tzutalin/labelImg.
[18] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *International Journal of Computer Vision,* vol. 111, pp. 98-136, Jan. 2015.