Deep Learning for Crack Detection using YOLOv4 and Darknet

Kristian Baturiano
Computer Engineering
Technological Institute of the Philippines
Quezon, City
qkjqbaturiano@tip.edu.ph

Dan Angelo A.Julongbayan
Computer Engineering
Technological Institute of the Philippines
Quezon, City
qdaajulongbayan@tip.edu.ph

J.C. Jeff E. Bonifacio
Computer Engineering
Technological Institute of the Philippines
Quezon, City
qjjebonifacio@tip.edu.ph

Rupert Conrad O. Mardrid
Computer Engineering
Technological Institute of the Philippines
Quezon, City
qrcomadrid@tip.edu.ph

***Abstract*— A concrete crack can be an indication of a structural severe damage which plays an important role to the structure health monitoring which requires classification as the first and critical stage for concrete SHM. Currently, deep learning is widely used in machines or devices. Thus, deep learning can be used automate this stage of health monitoring. Through deep learning we can train a system to identify the location of the cracks on any surface. Yolov4/Darknet is used to train the system into learning how to identify where on the surface is cracked.**

Keywords—structure health monitoring; concrete crack; deep learning; Yolov4/Darknet

I. INTRODUCTION

Defects on bridges can cause severe structural damage which plays an important role on monitoring the current health of the said structure. Classification is the first and critical stage for concrete SHM. [1] It is important to the safety of its user that the structure that they are using is on peak health to avoid disastrous accidents to occur. Chloride and Carbon Dioxide enhances corrosion which makes reinforced concrete weaker which became costly to maintain [2]. With this known it is better to prevent the cracks from happening than fixing the problem to avoid too many costs. Each cause of cracks has different preventive measures such as thermal movement, chemical reaction, shrinkage, etc. which can be prevented through proper planning and checking all the environment and materials of the concrete. [3]

There are some cases that even if we avoid the cracks from happening it may still occur at some point through natural causes such as aging or earthquakes. It is important that we can easily identify if there is a crack in an area. This checking for defects on concrete such as bridges can be proven costly and time-consuming which is where object detection can take place. A study where it compares two reinforcement learning based meta-learning approaches, MetaQNN and Efficient Neural Architecture search shows that learned architectures have fewer overall parameters which also yields better multi-target accuracy. [4] Yolov4 has a great design/architecture in object detection model because it has a higher robustness to images taken from different environments/area, and it also is a direct upgrade from its previous predecessors in terms of speed and performance. [5]

This project is composed of the authors fetching a dataset and then annotates the cracks on each photo inside the dataset which will be split into train and test dataset. The model created from the training will then be used to analyze images/videos for cracks and identify where it is located through a bounding box.

II. METHODOLOGY

A. Architecture

The YOLOv4 model using darknet framework was used to determine if any cracks were present on any surface of wall or concrete. YOLOv4 is a one-stage object detection model that is an improved versions of its predecessors (YOLOv3, YOLOv2, etc.) which works a lot faster than them with comparable performance.

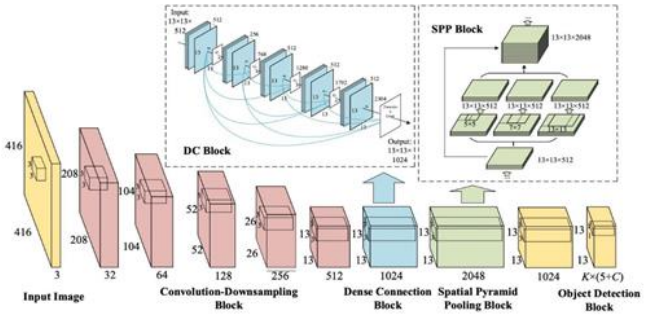

Figure 1 YOLOv4 Algorithm

Fig. 1 Shows the YOLOv4's Algorithm. The implementation of YOLO v4 in this research paper used its training method to create a model that learns based on the parameter, cracked, the authors of this paper identified. The object detection model was trained with 2000 epochs with 2 batch size.

B. Dataset

The dataset used to train the YOLOv4 model is composed of 499 images of concrete walls with cracks on them. The raw datasets (images) were gathered from a single source, "Concrete Crack Images for Classification" from Mendeley Data, which contains images from various METU Campus Buildings [6].

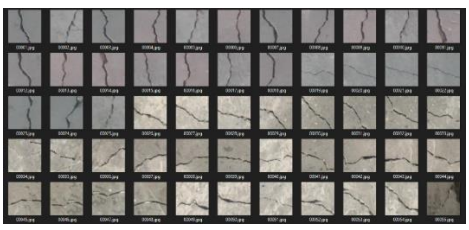

Figure 2 Raw Dataset

C. Data Pre-processing

The raw dataset was segregated randomly through a python script to split the dataset into two sets for the train dataset and test dataset. The dataset's cracks were annotated by the authors of this paper to serve as the guide for the deep learning process YOLOv4. The cracks were identified when it is a line of crack or a big hole that is part of the cracked line. Each individual crack is in its own bounding box, no matter the size. No data augmentation process was performed on the images as part of the training process.

D. Software Design

The system consists of a single software application to identify cracks from images. The software first check from its training model if a crack is present on an image which then adds a bounding box to the area where the crack is present inside the image. The program was run exclusively on a command line utilizing the previously stated python script and an .exe file. The exe file comes from a built version of Darknet, which is a C framework in which YOLO is built from. This framework supports CUDA from NVIDIA GPUs, enabling users to utilize their graphics cards for neural network computations [7].

E. Metric Evaluation

Mean average precision or mAP was utilized as it is the native metric evaluation method in YOLO. According to [8], the mean average precision is the comparison between the ground-truth bounding box and the detected box and returns it as a score. A higher score means that the detection box is more like the ground-truth box.

III. TESTING AND RESULTS

A. Training the Model

The training was conducted for cracks on a wall. The dataset was split into 80-20 train and test. The authors of this paper have undergone multiple reannotations to increase the accuracy of the model in detecting the cracks from any surface. The testing duration of the model with a total of 2000 epochs and 64 batch size was 3 hours.

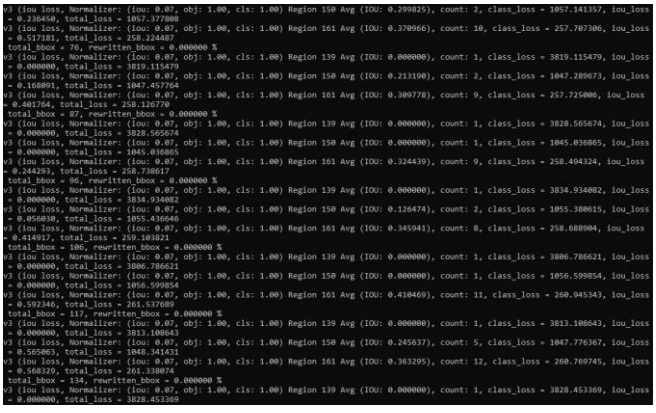

Figure 3 Training in Process in a Command Line

The accuracy of the model was tested and assessed through mAP which resulted to 96.4% accuracy which was observed on the 2000th epoch as seen on the Fig 3.

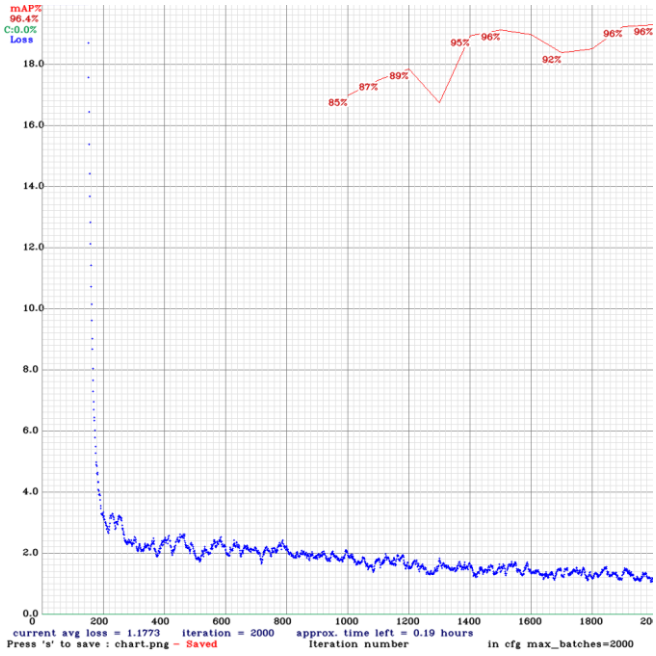

Figure 4 mAP Graph for the Accuracy and loss

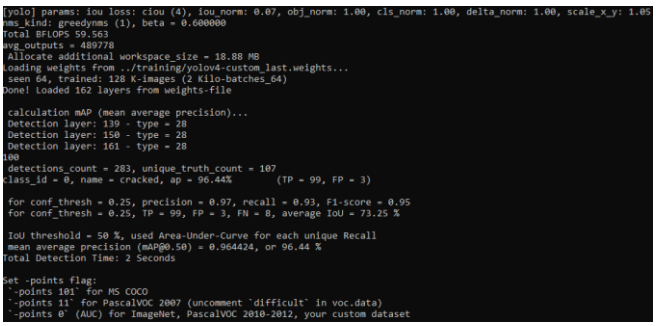

Figure 5 Command Line Output Showing mAP

B. Testing the Model

Figure 6 Testing Prediction and OpenCV Output

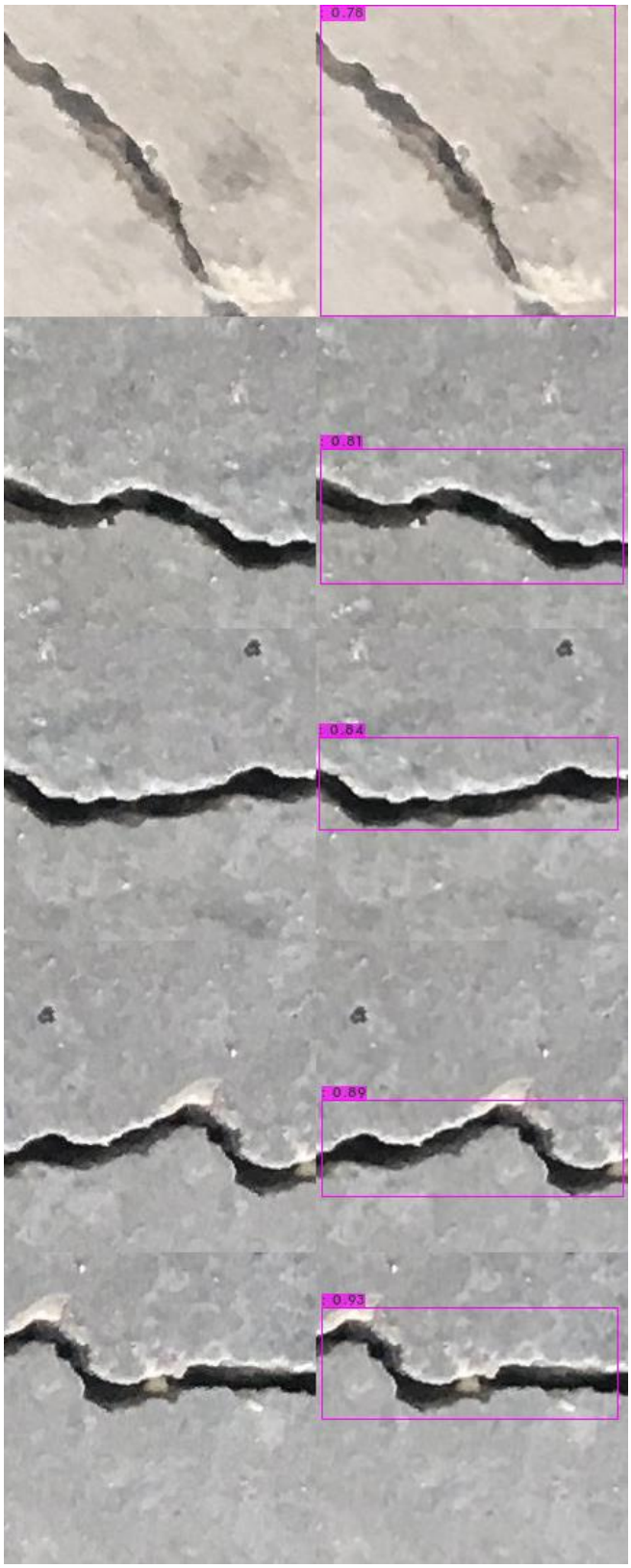

Figure 7 Sample Predictions of Model

The authors ran some other images in the original dataset for testing and prediction. The results from Fig. 7 proves a high percentage of confidence from the model which signifies that the model created from training is reliable in detecting the cracks and the bounding box successfully identify the location of the cracks on the image that the authors tested.

IV. CONCLUSIONS

After several training and reannotations that the authors of this paper have undergone, it can now be concluded that the design of this model that was created from training the model is able to detect cracks within any surface such as walls with a high amount of confidence which is important since this part of concrete health management is the most crucial part, and this model shows a high accuracy of crack detection. Furthermore, the model was able to automatically detect which part of the wall contains a crack that might be a hazard to its users.

ACKNOWLEDGMENT

The authors would like to thank Engr. Alonica Villanueva for giving valuable ideas and suggestions in implementing the YOLO model and in increasing the accuracy of our model, and by giving the authors of this research the necessary tools to complete this paper.

V. REFERENCES

[1] L. Guo, R. Li and B. Jiang, "A Cascade Broad Neural Network for Concrete Structural Crack Damage Automated Classification," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 4, pp. 2737-2742, 2020.

[2] S. Semwal, SushmitaDhonriyal, RakhiNegi, P. Gangwar and A. Bahuguna, "REVIEW PAPER ON CRACKS IN BUILDING AND THEIR REMEDIES," *International Research Journal of Engineering and Technology (IRJET)*, pp. 1-3, 2020.

[3] S. Abhyankar, "Research on Different Types of Cracks in Plain," *International Journal of Recent Technology and Engineering (IJRTE)*, pp. 1-3, 2019.

[4] M. Mundt, S. Majumder, S. Murali, P. Panetsos and V. Ramesh, "Meta-Learning Convolutional Neural Architectures for Multi-Target Concrete Defect Classification With the COncrete DEfect BRidge IMage Dataset," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11196-11205, 2019.

[5] R. Kanjee, "Augmented Startups," Augmented Startups, 28 April 2020. [Online]. Available: https://augmentedstartups.medium.com/yolov4-superior-faster-more-accurate-object-detection-7e8194bf1872#:~:text=There%27s%20just%20way%20too%20much,images%20obtained%20from%20different%20environments.. [Accessed 21 April 2022].

[6] M. Maguire, S. Dorafshan and R. J. Thomas, "Digital Commons @ USU," Utah State University, 17 May 2018. [Online]. Available: https://digitalcommons.usu.edu/all_datasets/48/. [Accessed 4 April 2022].

[7] J. Redmon, "Darknet: Open Source Neural Networks in C," 2016. [Online]. Available: http://pjreddie.com/darknet/. [Accessed 22 April 2022].

[8] A. F. Gad, "PaperspaceBlog," 2020. [Online]. Available: https://blog.paperspace.com/mean-average-precision/. [Accessed 22 April 2022].