Road Crack Detection and Classification Using Deep Learning

Harshal Bhoir[1], Amitkumar Pandey[2], Rushikesh Patil[3], Aparna Bhonde[4]

[1]BE student, Information Technology, Datta Meghe College of Engineering, Navi Mumbai, India
[2]BE student, Information Technology, Datta Meghe College of Engineering, Navi Mumbai, India
[3]BE student, Information Technology, Datta Meghe College of Engineering, Navi Mumbai, India
[4]Asst. Professor, Information Technology, Datta Meghe College of Engineering, Navi Mumbai, India

---***---

Abstract - *Road Crack detection and Crack Classification using a Deep Learning model is carried out in this project. At present, the road crack detection and analysis is a manual process. Using YOLO, a deep learning model and training it on a dataset for crack identification and classification, we arrive at the final results for the same, after which we generate an output file for the user that would show a report about which images contains cracks, their accuracies and which images do not contain any cracks. This system will help the authorities upload various images of a location in batches and get the results for the same via the text file output that will help ease the process of crack detection for the concerned government officials.*

***Key Words*: Road Cracks, Object Detection, Classification, Deep Learning, YOLO**

1.INTRODUCTION

Road cracks are a common sight in a majority of Indian cities as well as around the world. They also pose a health hazard. Poorly maintained roadways cause accidents in a variety of ways, mostly due to the fact that they create an enormous hazard to drivers. In many instances, a driver may attempt to avoid a certain situation, like a pothole or pooling water which could cause a serious accident. In such cases, it is of critical importance that such road cracks be tended to as soon as possible to avoid injuries and before they become fatal. Currently the road crack detection is a manual process which takes a lot of time from start to end. The cracks can be overlooked at times due to human error as well. The project tries to make this process a bit easier by automatically finding the various crack locations.

1.1 Proposed System

The proposed system predicts the distress type and localizes it within an input image and also gives a report to the user in an appropriate format which is displayed on the screen as well as available in the form of text document. The detection of crack is done using YOLO (You Only Look Once) object detection algorithm which is trained on over 6000 numbers of training images. While making the prediction model also consider some other factors apart from crack type which are commonly present on every road, for example white lines and construction joints. This system will help the authorities by giving a proper report of the roads about the number of cracks present, along with their type that will help in ascertaining which road segments need immediate attention.

1.2 YOLO (You Only Look Once) algorithm

YOLO (You Only Look Once) is a real-time object detection algorithm, which is one of the most effective object detection algorithms out there. YOLO uses convolutional neural networks (CNN) for object detection in real-time. The algorithm applies a single neural network to the full image, and splits the image into cells, typically a 19x19 grid. Each cell is then responsible for predicting K bounding boxes. An object is considered to lie in a specific cell only if the center coordinates of the anchor box lie in that cell. YOLO aims to predict a class of an object and the bounding box specifying object location. Each bounding box can be described using four descriptors 1)Center of the box (bx, by), 2)Width (bw), 3)Height (bh), 4)Value C corresponding to the class of an object. Along with that it predicts a real number pc, which is the probability that there is an object in the bounding box. The class with the maximum probability is chosen and assigned to that particular grid cell. Similar process happens for all the grid cells present in the image. After predicting the class probabilities, the next step is non-max suppression, it helps the algorithm to get rid of the unnecessary anchor boxes. It calculates the value of IoU for all the bounding boxes respective to the one having the highest-class probability, it then rejects the bounding boxes whose value of IoU is greater than a threshold. It signifies that those two bounding boxes are covering the same object but the other one has a low probability for the same, thus it is eliminated. Once done, algorithm finds the bounding box with next highest-class probabilities and does the same process, it is done until we are left with all the different bounding boxes. After this, almost all of our work is done, the algorithm finally outputs the required vector showing the details of the bounding box of the respective class.

2. Approach

The system will take a road image as input, then it detects crack and its type using deep learning algorithm YOLOv3. The first step of our project is data collection. We get the dataset from IEEE Big data 2020 challenge. It contains road images that contain cracks and some without cracks. The images were collected from the various cities of Japan. The crack information is provided as the coordinates of the bounding box and a label depicting the type of crack associated with the box. It has a total of seven types of labels. The crack types that this dataset has is as follows: Longitudinal, Lateral, Alligator, Pothole, Crosswalk, White Line Blur and Construction Joint.

The next part in our project is image pre-processing. In image pre-processing, the first step is Normalization. It ensures similar data distribution. We scale down the image pixels from range of [0,255] to [0,1]. The second step in pre-processing is Image Resizing. The training images were resized to a fixed size of 416 X 416 pixels. This step helps to reduce the computational cost. The third step is CLAHE filtering which is used to increase the image sharpness.

Now, the training dataset is ready for the model to train on. The training dataset used is roughly 70% of the total dataset images. The YOLOv3 algorithm is used for crack detection and classification. All this was coded in and ran on Google Colab. We have imported darknet, which is the underlying framework of YOLOv3 and then created all the files and folders which are required to train our model. The first file, obj.names contains the various crack types to be detected. The second file, obj.data contains information about total number of classes, path to training and testing dataset, and path to configuration file. After all this was done, we trained our deep learning model using YOLOv3 and the training images.

Now the deep learning model is ready for crack detection and classification. The next part is report generation. We take a set of images as input from the user and then we run our trained model on the given image. Our system will generate detailed report in text as well as JSON format. The report contains detected crack types along with their corresponding accuracies. If the image is not appropriate for our model or if it does not contain any cracks then an appropriate message is displayed.

All the output images which contain bounding boxes around the cracks and crack type along with accuracy are stored in the output folder. Users can get output images from that folder.

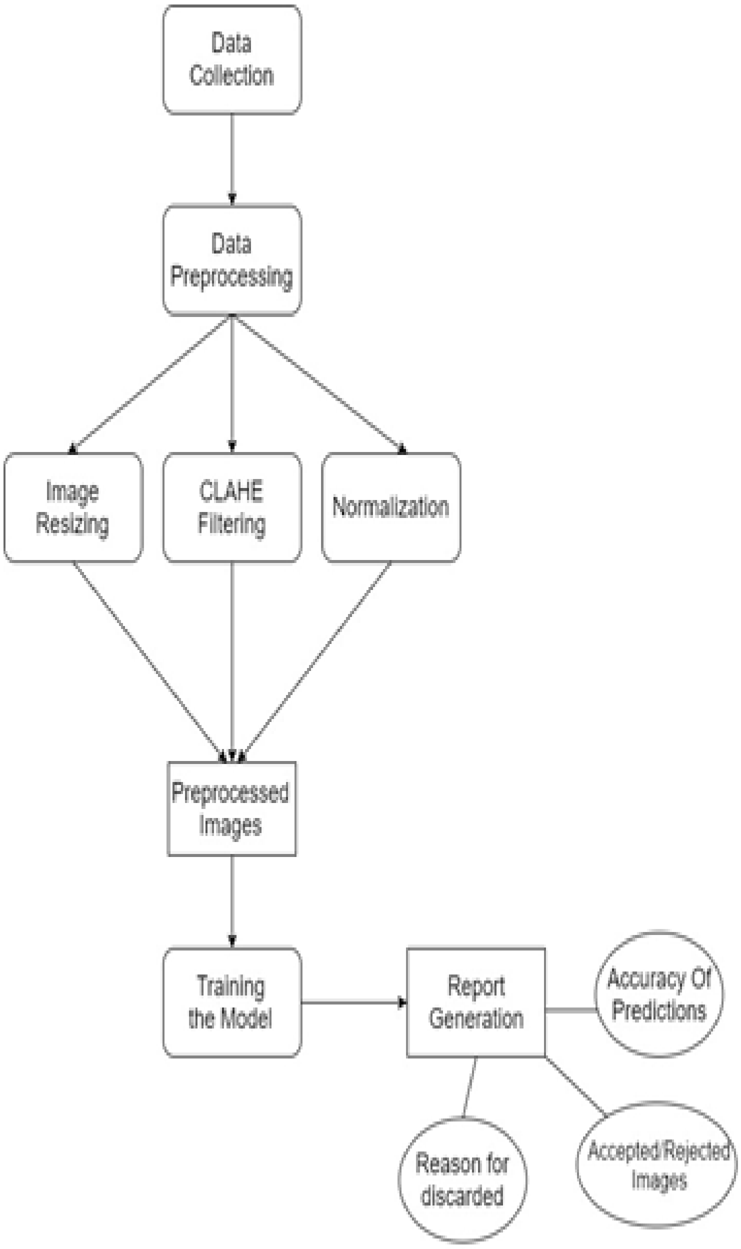

Fig - 1: Flow Chart

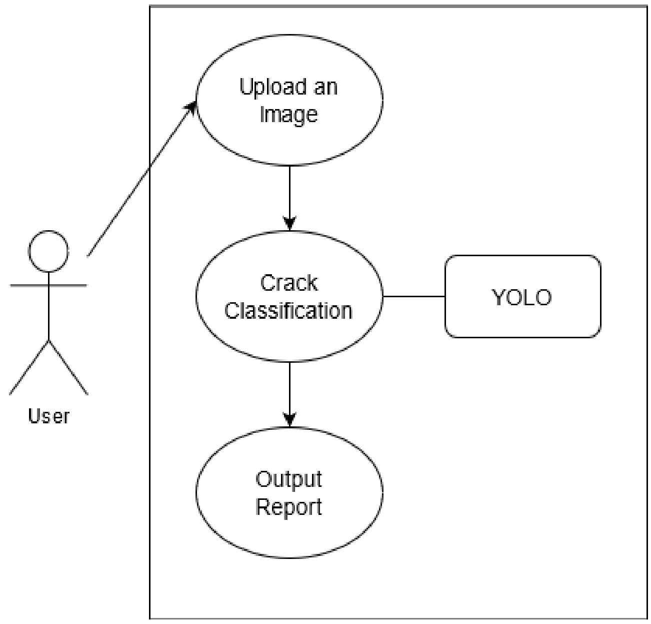

Fig - 2: Use Case Diagram

3. RESULTS

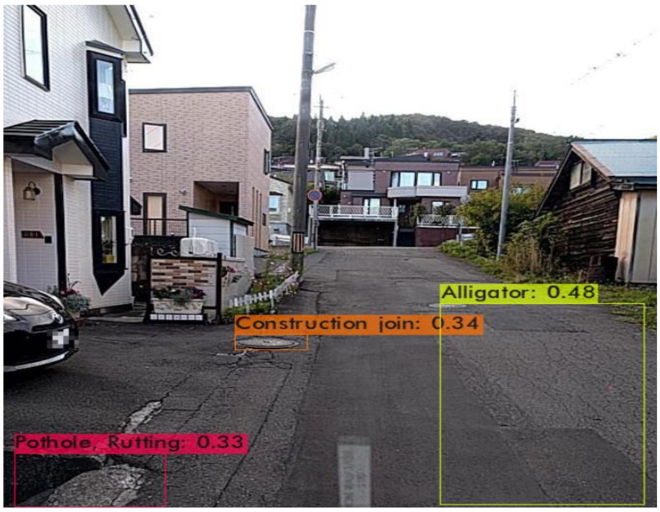

Fig – 3: Output Image I

The image shown in Fig 3 is the output image given by the model that clearly defines three different cracks on the road image considered and also provides us with their classification, that is, which type of crack they belong to. The image in Fig 4 is another example that shows us that the model has detected one crack in the image and has classified it as alligator crack.

Fig – 4: Output Image II

In Fig 5, we get the final output report in a text file that can be downloaded by the user for further use. It shows the various types of images that were uploaded for crack identification and outputs only those images that were clear and not of low quality, while the other images that may not contain road in them are discarded. The image name and its corresponding accuracy is mentioned in the report.

```
Image: Nagakute_20170912131506.jpg has 1 cracks/cracks
        Crack Type: Alligator            |   Accuracy: 66.98%

*****************************************************************

Image: Japan_000753.jpg has 3 cracks/cracks
        Crack Type: Construction join    |   Accuracy: 34.08%
        Crack Type: Pothole, Rutting     |   Accuracy: 32.83%
        Crack Type: Alligator            |   Accuracy: 48.06%

*****************************************************************

Image: Japan_001497.jpg has 1 cracks/cracks
        Crack Type: Alligator            |   Accuracy: 85.33%

*****************************************************************

*****************************************************************
There are total 1 image/images which are not suitable for our model
Names of the images:
        1. 1.jpg
Reasons could be:
        1. Crack is not visible properly
        2. Low quality of the image
```

Fig – 5: Output Report

4. CONCLUSION

The number of different types of cracks occurring on roads due to various human factors is increasing steadily and hence, there was the need to find a way to detect and identify the various cracks through images and the project aimed at finding and testing for road cracks and classifying them as well. For data pre-processing, Image Resizing, CLAHE Filtering and Image Standardization were the steps that were applied on the existing images dataset. The deep learning model used, YOLO (You Only Look Once) was able to accurately locate and detect more than a single type of crack for a given image along with good accuracies. The resulting image outputs were converted to a downloadable text file so that information regarding the images tested will be available to the user in a clean manner. This will help avoid road accidents as well, especially potholes that are a menace in many countries during the monsoon, such as India.

ACKNOWLEDGEMENT

Working on this project titled "Road Crack Detection and Classification using Deep Learning" was a source of immense knowledge to us. However, this would not have been possible without the equal support of all the members. We would also like to express our sincere gratitude to Asst. Prof. Aparna Bhonde for her guidance and valuable support throughout the project.

REFERENCES

[1] Henrique, Paulo Correia, "Automatic Road Crack Detection and Characterization", IEEE Transactions on Intelligent Transportation Systems, Volume 14, Issue 1, August 2012
[2] Yonh Shi, Limeng Cui, Z. Qi, Fan M., Z Chen, "Automatic Road Crack Detection Using Random Structured Forests", IEEE Transactions on Intelligent Transportation Systems, Volume 17, Issue 12, May 2016
[3] Lei Zhang, Fan Yang, Y. M. Zhang, and Ying J. Zh, "Road Crack Detection Using Deep Convolutional Neural Network", Department of Electrical and Computer Engineering, Temple University, Philadelphia, PA
[4] Y. Du, Z. Weng, C. Liu, and Difei Wu, "Dynamic Pavement Distress Image Stitching Based on Fine-Grained Feature Matching", Hindawi Journal of Advanced Transportation, Volume 2020, February 2020
[5] W. Song, G. Jia, Hong Zhu, Di Jia, Lin Gao, "Automated Pavement Crack Damage Detection Using Deep Multiscale Convolutional Features", Hindawi Journal of Advanced Transportation, Volume 2020, January 2020
[6] Joseph R., Santosh Divvala, Ross G., Ali Farhadi, "You Only Look Once: Unified, Real-Time Object Detection", 2016 Conference on Computer Vision and Pattern Recognition (CVPR), December 2016
[7] Shaoqing Ren; Kaiming He; Ross Girshick; Jian Sun, "R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", IEEE Transactions on Pattern Analysis and Machine Intelligence, June 2017