



IJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** VIII **Month of publication:** August 2022

DOI: <https://doi.org/10.22214/ijraset.2022.46175>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

Road Damage Detection and Classification using YOLOv5 on an Android Device

Ameya Mahadev Gonal¹, B Chirag Baliga²

^{1,2}Department of Information Science and Engineering RV College of Engineering Mysore Road, Bengaluru - 560059

Abstract: Most of the rural or the urban municipalities and road authorities have hard times to map the surface damages caused due to various reasons such as heavy rains, natural calamities or other factors which lead to cracks and holes to appear on the surface of the roads. These organisations or private entities look out for solutions to implement automated methods of reporting damages on a surface of the road. In most cases, they lack the technology required for the purpose of mapping the damages on the roads.

One of the biggest problems for commuters is that they have to face a lot of damaged sections of road which leads to frequent reduction of the speed at which they travel, wasting a lot of time and effort from the riders perspective, which thereby increases the travel time to their destinations.

Damage to the road can be fatal many times when travelling at higher speed and all of a sudden meeting a damaged part of the road. Moreover, commuters are at more risk while driving at night time due to improper visibility of the damage to a part of the road.

Artificial Intelligence and Machine Learning has the potential to make traffic more efficient, ease traffic congestion in most parts of the road network in an urban or rural area, reduce driver's time and efforts. As AI helps to keep road traffic flowing, it can also enhance the efficiency of fuel consumption to an extent which are caused by vehicles idling when stationary and improve air quality of a city and urban planning for road networks. Moreover, it has the potential to detect frequent congestion and the reason behind it and also propose a solution for the same. Most of the time these congestion are caused by damages to the road making the commuters travel at a much lesser speed than what is recommended.

Keywords: Object detection, YOLOV5, Android device, TFLite model, Tensorflow, Road Damage.

I. INTRODUCTION

The road network forms a crucial part of today's economy, as it facilitates transportation, which in turn affects several important industries as well. It comes with no surprise that improper roads can affect the traffic, with significant impact in urban areas where the traffic is generally a bigger problem. The ever expanding road network calls for proper maintenance, which the local authorities generally take care of every year, or after receiving complaints from people living in the locality. This task of finding and fixing the damages on the roads soon gets cumbersome, and with time it becomes increasingly inefficient to maintain every road, which causes the conditions of roads to deteriorate eventually. The process of damaging the road and fixing them is never ending, as there are a plethora of reasons a road could get damaged.

The fact that most of this work is done manually is also a limiting factor when the speed and efficiency of the task at hand is considered. The local authorities have to make use of surveyors who go around the city looking for damages. This method is extremely inefficient and prone to human error.

Therefore, there is a need for a better performing and more efficient approach to automate the process of detecting and classifying road damages by getting a machine or a type of intelligent system to do this task effectively and diligently. The branch of Computer Vision can tackle this problem of the strenuous process of maintaining the quality of all the roads simultaneously. Artificial Intelligence and Machine Learning have worked wonders in the recent years and they can be effectively applied for this problem of managing these road damages and the affected areas and even completely automate the further required tasks such as reporting the identified damages to a centralised database, with proper categories of different types of damages, which the local authorities can make use of to perform an analysis of the severely affected areas and resolve the issues in an efficient manner.

II. LITERATURE SURVEY

A vast survey was conducted over the internet in relevant domains of this project.

The domains for this survey included Object detection methods, image enhancement techniques and speeds of various object detection models. The models based on ResNet and VGG used in [1] converged at a faster rate, with higher mAP than models based on DenseNet. The false positives found were where the model showed a shadow detected as road damage. Other shadows like tree branches also caused false positives.

[2] applied Faster R-CNN to detect and classify damaged roads. Through analyses of aspect ratios and sizes of the damaged areas of the road in the training dataset, they adjust relevant parameters of the model. In order to solve the problem of unbalanced data distribution of different classes, they introduced some data augmentation techniques (contrast transformation, brightness adjustment, and Gaussian blur) before training. Experimental results demonstrated that this method can achieve a Mean F1-Score of 0.6255.

[3] compared various existing methods that exist in the road damage detection domain. The best performing method was found to be the one that was based on ultralytics-YOLO (u-YOLO) [YOLOv5, 2020]. The proposed approach applied the test time augmentation (TTA) procedure on test data to improve their model's robustness. TTA augments the data by using several transformations (e.g., horizontal flipping, increasing image resolution) on each test image and generating new images

The study done in [4] compared the results provided by the SSD Inception V2 and SSD MobileNet. Although D01 and D44 were detected with relatively high recall and precision, the value of recall is low in the case of D11 and D40. When comparing values of Recall, MobileNet exceeded Inception in six categories, except D40 and D43. Overall, the SSDMobileNet yielded better results.

The study in [5] investigated data augmentation using PG-GAN and Poisson blending and demonstrated that the proposed method can improve the F-measure for the road pothole detection task. The results show that adding a synthesized road damage image to the training data improves the F-measure by 5% when the number of original images is small, and by 2% when this number is relatively large. A study was done in [6], comparing the two-stage Faster R-CNN with one-stage Yolov5 detection model for evaluation of framework and observed significant improvement in average F1 score. The primary issues in images are found to be due to low light conditions, camera mount positions, artifacts or shadows around objects of interest and inability to avoid looking far down the road.

Another approach termed the Ensemble model (EM) was used in [7]. The approach EM ensembles different variants of u-YOLO models. Given that training a u-YOLO model involves tuning different hyperparameters, using different combinations of these parameters generates different trained models. A subset of these models is selected such that they maximize the overall accuracy. Each image is passed through all the selected models, and predictions from each model are averaged before applying non-maximum suppression. This ensemble technique helps in achieving better accuracy by reducing the prediction variance.

After that, the team combines the two approaches and proposes the final solution termed as Ensemble Model with Ensemble Prediction (EM+EP). In this approach, EM is extended with the TTA procedure used in EP. That is, after transforming a test image using TTA, the augmented images are fed into each model of EM. Then, the predicted bounding boxes from the augmented images for each model are averaged before applying NMS.

The authors then compare the performance in terms of speed as well as accuracy for all three approaches (EM, EP, and EM+EP). The statistics show that while the accuracy is improved in the case of (EM+EP) providing the highest F1-score (0.67 for test1), the approach is worse in terms of speed of detection, as measured using Detection time per image.

Authors in [8] selected Faster R-CNN with X101-FPN as the architecture to tackle the detection tasks. This approach resulted in F1 scores of 51.0% and 51.4% for the test1 and test2 sets respectively. The low F1 score was arguably acceptable due to several issues with the ground-truth in the training sets. However, the main limitation of using Faster R-CNN with X101-FPN was found to be that it is slower to train and has a longer prediction time than other model types such as YOLO and SSD.

III. METHODOLOGY

The methodology contains three major steps which are described as follows:

A. Data Gathering And Pre-Processing

The data was obtained from an IEEE website which hosted a competition in 2020 named Big Data cup for Road Damage detection. The training dataset was 1.8 GB in size, and contained images with their corresponding annotations in a ZIP file. The collected data was from Japan, India and Czech Republic.

The original annotations were in the format of XML type files which was later converted to text files because of the compatibility with yolo training format. The .txt files which had redundant data were scrapped at the time of training along with the labels which had no numbering such as there are no label numbers corresponding to a bounding box and vice-versa were also taken care of at the time of training.



Figure 1: A pothole in the middle of the road.

In figure 1, there is a pothole damage type on the surface of the road.

Potholes are the most dangerous types of all the road damages on the surface due to its nature of trapping the wheel of a vehicle which increases the chances of a two-wheeler to lose its balance on the road which in turn increases the chance of a fatal accident.



Figure 2: Alligator cracks.

Alligator cracks are one of the most types of damage to a road section. They are caused due to repeated overloading of the vehicle traffic and need to be repaired quickly. If not repaired in time, an alligator crack can turn into a pothole which is much more dangerous for the incoming traffic flow.

In figure 3 it shows the blurred zebra crossing which is not directly dangerous to any vehicle but is dangerous for the pedestrians when the driver does not spot a zebra crossing and thereby does not slow down which may turn into an accident prone zone for the pedestrians.



Figure 3: Blurred Zebra crossing.

In table 1, we can see the various different kinds of damages that have been marked in the annotated dataset mentioned above.

Table 1: Class names for various types of road damage.

| Damage Type | | Detail | Class Name |
|-----------------|--------------|---|--------------------------------|
| Crack | Linear Crack | Longitudinal | Wheel-marked part D00 |
| | | | Construction joint part D01 |
| | Lateral | Equal interval D10 | |
| | | Construction joint part D11 | |
| Alligator Crack | | Partial pavement, overall pavement D20 | |
| Other Damage | | Pothole D40 | |
| | | Cross walk blur D43 | |
| | | White line blur D44 | |

B. Model Building And Training

The YOLOv5m model was chosen amongst all the variants of the Yolov5 since it was a good tradeoff between speed and the accuracy on an Android device. Ultralytics github repository provides the code for training a YOLOv5 model. There have to be three directories constructed for the training purposes each for training dataset, testing dataset and finally the validation dataset. Each one of the directories must contain two folders each one for images and another one for the annotations corresponding to each of the images present in the image folder. The model was trained for approx. 40 epochs on the Google Colab platform which boasts of high computing GPUs capable of handling complex tasks very easily and in a speedy manner.

The trained YOLOv5 model was converted to Tensorflow Lite model which is compatible with the libraries present in the Android environment which makes it possible to run inferences on any smartphone device running on Android OS without the need of the server computation.

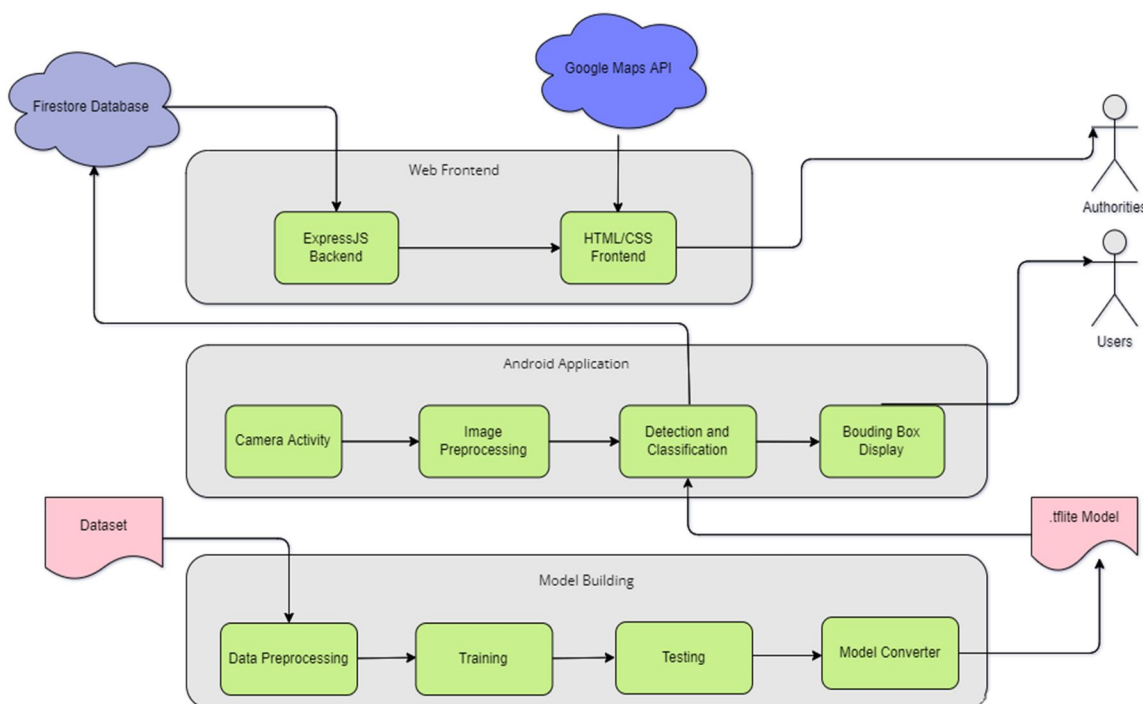


Figure 4: Architecture Diagram

C. Interface Development

The damages then detected in the Android application are then sent to a centralized database (MongoDB) for storing the details of the damage captured on the road. The details include the title of damage, GPS Location (longitude and latitude of the damage captured), locality/area, confidence score given for the detected damage, label from the dataset and lastly the timestamp of when it was captured.

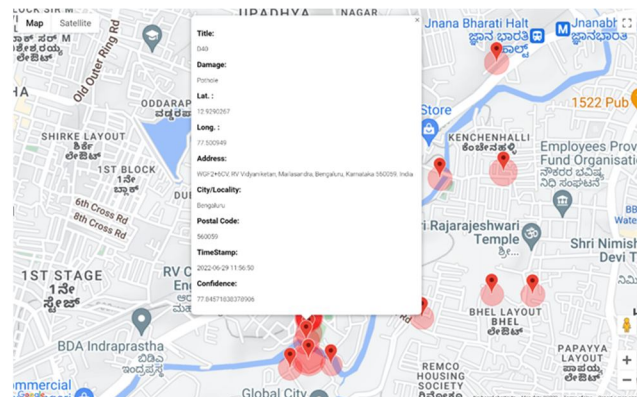


Figure 5: Interactive map for visualising road damages in the city of Bangalore

These details would then be fetched from the database and would be placed onto a map for better understanding. For visualising all the damages by a local authority in an easier manner, the development of an interactive map was done in order for the local authorities to identify which areas of the city are having damaged roads with just the click of an URL provided to them. Markers would be placed on the coordinates of a damaged road as shown in the figure 5 along with the proper description highlighting the details of the damage present at the location.

IV. RESULTS

The trained YOLOv5m model achieved an F1 score of 0.52. The training and validation loss showed a steady decrease after each epoch while training, and a suitable epoch count was chosen keeping overfitting in mind.

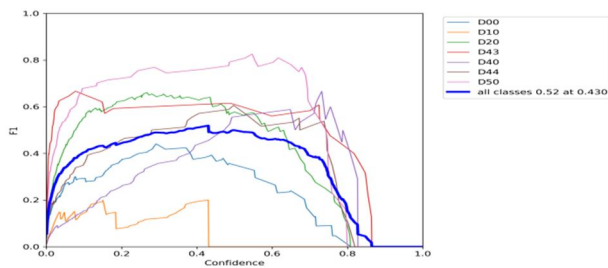


Figure 6: F1 score.

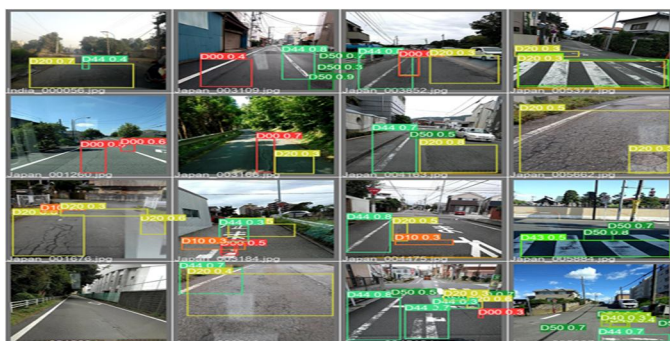


Figure 7: Detected labels by the model.

Figure 7 shows the labels of the predicted annotations/bounding boxes around the damaged parts of the road.

The true labels can be seen on the y-axis in the below figure for the confusion matrix against the predicted labels on the x-axis. Majority of the damages have been labelled correctly by the model which can be inferred from the confusion matrix in Figure 8.



Figure 8: Confusion matrix

In figure 9, the Yolo5m model which was later converted to Tensorflow Lite model for its compatibility with the Android Tensorflow libraries was deployed on an Android device for the purpose of running inference on real time footage captured from the camera mounted on the rear side of the device.

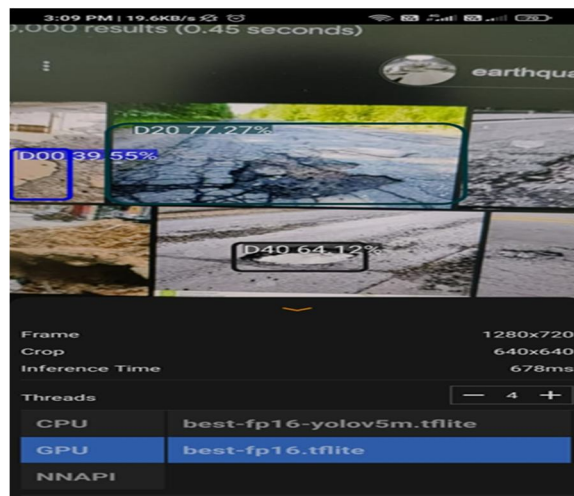


Figure 9: Android application running real time detections on the camera stream.

V. CONCLUSION

Road damage detection is a crucial problem, and many kinds of research have been done to break through this challenge. As a deep learning approach, we used a YOLO based solution to detect road damage by training the model with data from Czech Republic, India, and Japan. The dataset is collected by Smartphone applications from each of these countries. We evaluated various dataset scenarios and it showed some interesting points. Using Japan's road damage dataset with the Czech or India could affect the schedule of convergence of the model and generalization positively, but it does not always improve the performance of the model. For our YOLOv5 based solution, one pre-trained weight needs just 42MB memory and its inference speed is extremely fast. In the perspective of real road damage detection problems, Not only accuracy but also inference speed is important, even FPS can be a much more crucial point. Therefore, this solution could be an appropriate candidate for road damage detection on smartphone applications in real-time.

VI. FUTURE SCOPE

Following are some of the directions that can be considered for the future road damage detection tasks:

Currently, the road damage detection task is limited by the methods of training and on the available training data. In the future, more effective methods on any other newer datasets may also be made use of.

This paper mainly focuses on the task of detection and classification of surface road damages using an Android device.

Other challenges may include introducing new features such as Severity analysis, Pixel-level damage analysis for covering a wider range of surface road damages. Multiple evaluation metrics can be implemented for analyzing the performance better. For example, considering the inference rate and disk size of the trained model in addition to F1-Score would be of significance for developing faster real-time smartphone-based object detection.

Instead of considering only the average F1-score, the future versions may also consider the models' performance for individual damage classes. The road damage dataset could be augmented to include a more balanced representation of damage classes.

REFERENCES

- [1] L. Ale, N. Zhang and L. Li, "Road Damage Detection Using RetinaNet," 2018 IEEE International Conference on Big Data (Big Data), 2018, pp. 5197-5200, doi: 10.1109/BigData.2018.8622025.
- [2] W. Wang, B. Wu, S. Yang and Z. Wang, "Road Damage Detection and Classification with Faster R-CNN," 2018 IEEE International Conference on Big Data (Big Data), 2018, pp. 5220-5223, doi: 10.1109/BigData.2018.8622354. ISSN – 2321-9653
- [3] D. Arya et al., "Global Road Damage Detection: State-of-the-art Solutions," 2020 IEEE International Conference on Big Data (Big Data), 2020, pp. 5533-5539, doi: 10.1109/BigData50022.2020.9377790.
- [4] Maeda, Hiroya & Sekimoto, Yoshihide & Seto, Toshikazu & Kashiyama, Takehiro & Omata, Hiroshi. (2018). Road Damage Detection and Classification Using Deep Neural Networks with Smartphone Images: Road damage detection and classification. *Computer-Aided Civil and Infrastructure Engineering*. 33. 10.1111/mice.12387.
- [5] Maeda, Hiroya & Kashiyama, Takehiro & Sekimoto, Yoshihide & Seto, Toshikazu & Omata, Hiroshi. (2020). Generative adversarial network for road damage detection. *Computer-Aided Civil and Infrastructure Engineering*. 36. 10.1111/mice.12561.
- [6] Vishwakarma, Rahul & Vennelakanti, Ravigopal. (2021). CNN Model & Tuning for Global Road Damage Detection.
- [7] K. Doshi and Y. Yilmaz, "Road Damage Detection using Deep Ensemble Learning," 2020 IEEE International Conference on Big Data (Big Data), 2020, pp. 5540-5544, doi: 10.1109/BigData50022.2020.9377774.
- [8] V. Pham, C. Pham and T. Dang, "Road Damage Detection and Classification with Detectron2 and Faster R-CNN," 2020 IEEE International Conference on Big Data (Big Data), 2020, pp. 5592-5601, doi: 10.1109/BigData50022.2020.9378027.
- [9] He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 2015, 37, 1904–1916.
- [10] Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
- [11] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., "Tensorflow: a system for large-scale machine learning.,"
- [12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*, pp. 740–755, Springer, 2014.
- [13] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.
- [14] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.
- [15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.
- [16] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.
- [17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014. K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.
- [18] T.-Y. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, "Feature pyramid networks for object detection.," in *CVPR*, vol. 1, p. 4, 2017.
- [19] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, pp. 91–99, 2015.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)