Pothole Detection using AI

Using FRCNN and YOLO algorithms

- Project Under the course of Artificial Intelligence

Kishore DT
Nirmal S
Varun Kishore S
Abhinandhan S
Senthamizh G
Sreekutty K P

What are Potholes?

Pothole

A pothole is a depression in a road surface, usually asphalt pavement, where traffic has removed broken pieces of the pavement. It is usually the result of water in the underlying soil structure and traffic passing over the affected area.

How to Detect Potholes?

How to Detect

- One method can be detecting it manually, but it is very painful, time consuming and not feasible.
- Here we can take advantage of our computers and their powerful processing and computing capabilities.
- People have researched various algorithms and techniques for object Detection and using these we can also detect Potholes.

Implementation of Research Paper

Research Paper we implemented

Used Faster-RCNN (FRCNN) for the detection of Potholes.

A Modern Pothole Detection technique using Deep Learning

Authors

Abhishek Kumar
Gaya College of Engineering, Gaya, Bihar, India
Chakrapani
Gaya College of Engineering, Gaya, Bihar, India
Dhruba Jyoti Kalita
Gaya College of Engineering, Gaya, Bihar, India
Vibhav Prakash Singh
National Institute of Technology, Allahabad, UP, India

Research Paper =>

https://ieeexplore.ieee.org/document/9170705

Creating Dataset

Dataset Link - https://www.kaggle.com/sachinpatel21/starter-code-to-view-dataset-images/data

Image labelling using - https://github.com/tzutalin/labelImg

We had got the Pothole images dataset and we had used labelImg which is an open tool for labelling your images.

Training of Faster-RCNN

TensorFlow provides several object detection models (pre-trained classifiers with specific neural network architectures) in its model zoo.

link to the models - https://github.com/tensorflow/models/tree/master/research/object_detection/models

And we choose the required F-RCNN model from it.

Now we trained our model using:

- prepared dataset
- .pbtxt file (which contains the labels)
- train.record(which contain config of every image contained in the train folder)
- test.record (similar to train.record)
- and the base model obtained from tensorflow.

Testing of Faster-RCNN

For testing we used:

- .pb file (Trained Model file)
- .pbtxt file (contained the labels)
- Testing image
- Class file

Now after executing the driver code with the aforesaid requirements we get our detection results.

Detection Results of Faster-RCNN

The model is effectively able to predict potholes and a group of potholes from images and videos provided to it.

The model works similar to the model suggested in the research paper, and the results are good.

Improvement of previously implemented research paper

Improvement of our previous detection model

After much research and testing we found about YOLO Object Detection Algorithm which we thought would perform better with respect to Faster-RCNN.

YOLO Object Detection Algorithm

YOLO is an algorithm based on regression and is a single-step process wherein, in one run, the image is screened, and objects are located with the bounding boxes as well as their class is predicted in the same run.

It is generally used for real-time detection where time, speed, and accuracy form prime concerns, it is primarily used in the application of Artificial intelligence and Deep Learning.

Creating Dataset

Dataset Link - https://www.kaggle.com/sachinpatel21/starter-code-to-view-dataset-images/data

Image labelling using - https://github.com/tzutalin/labelImg

We had got the Pothole images dataset and we had used labelImg which is an open tool for labelling your images.

Training of YOLO Object Detection Algorithm

YOLO is based on the darknet, built in C. Darknet is an open source neural network framework written in C and CUDA.

link to the darknet- https://github.com/pjreddie/darknet

And we chose YOLO v3 model from it.

Now we trained our model using:

- prepared dataset
- Config file (contains CNN Architectures (layers and activations) Anchor Boxes, Number of classes, Learning Rate, Optimization Technique, input size, probability score threshold, batch sizes)
- Default darknet weight (required for initial epochs)
- Obj.data (contains train.txt, test.txt which further contains the path to dataset images, backup and number of classes)
- Obj.names (to name classes)

Testing of YOLO Object Detection Algorithm

For testing we used:

- Trained Weights file (updated automatically after regular intervals while training)
- Obj.data
- Config file
- Testing image

Command to Test :

darknet detector test <obj.data> <config> <weights> <image>

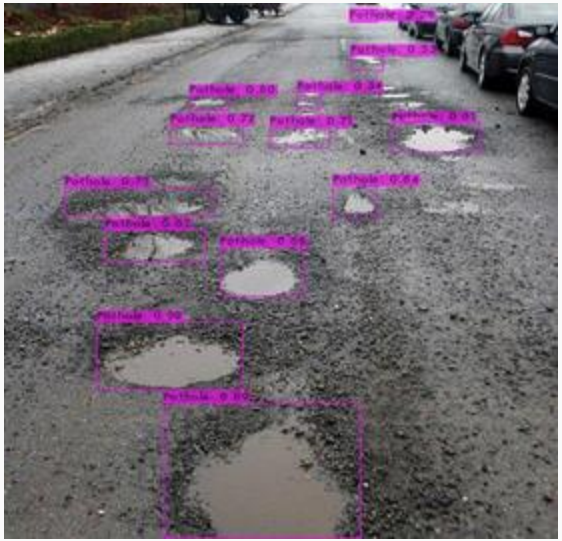
Example:

```
darknet detector test custom-cfg/obj.data custom-cfg/yolo-pothole-train.cfg .\custom-backup\yolo-pothole-train_last.weights .\test_17.jpg
```

Detection Results of YOLO Object Detection Algorithm

The model is effectively able to predict potholes and a group of potholes from images and videos provided to it.

The model works nicely and the results are good.



Comparison between Implementation of Research Paper and its Improvement

- FRCNN does not have a wide range in detection of Potholes whereas YOLO detects almost every Pothole within a wide range.
- FRCNN’s most of the detections are of large Potholes whereas YOLO tries to detect small Potholes also.
- FRCNN takes more time in detection than YOLO.

Conclusion

- Both the models were effectively able to detect potholes from images and videos provided to it.
- It was easily concluded that YOLO performs better than FRCNN with respect to speed and accuracy.

Thanks!

Drive Safe !!

The Roads are full of Potholes