Pothole Detection and Avoidance System

Shreekanya Kodate, Saurabh Jaiswal and Swati Meena
Dept. of Electrical Engineering, Indian Institute of Technology Bombay, India
Email: {183079006, 184074003 and 183070009}@iitb.ac.in

***Abstract*— Potholes can generate vehicle damage and major accidents. Quick and accurate detection of potholes can be very useful to avoid these damages and accidents. In this project we have tried to develop a system which can detect the potholes coming in the way of vehicle and alert the driver to avoid those potholes. For experiments, 2-D images of potholes in asphalt pavements were used. To detect the potholes in the images edge detection techniques, gray level thresholding and semantic segmentation were used. The results are promising, and the information collected using this method can be used not only for alerting the driver, but also in taking immediate action for repair and maintenance of potholes.**

Index Terms— Pothole, Edge detection techniques, Gray level thresholding, Semantic segmentation.

I. Introduction

Poor road surface conditions are major concerns for safe and comfortable transportation. The most common form of poor conditions are potholes, which can compromise safety and may lead to vehicle damage and casualties.

Also, Figure 1 shows the numbers of crashes and death related to bad road facilities for year 2017 in different states of India.

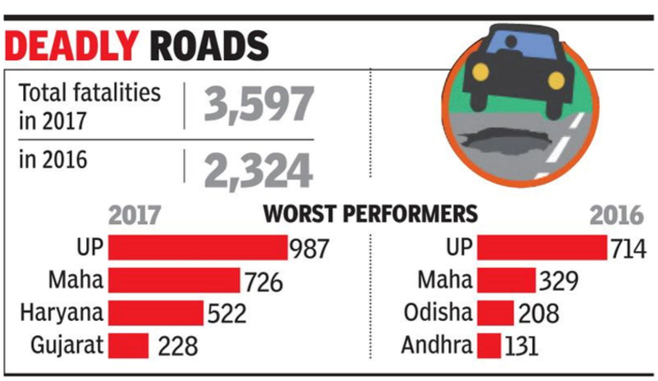


Fig. 1. Fatalities on road for year 2017 in India(Source : TOI news Potholes killed 3,597 across India in 2017, terror 803)

As shown in the figure , potholes are responsible for more than 30% of total number of fatalities in India. As one type of pavement distresses, potholes are important clues that indicate the structural defects of the asphalt road, and accurately detecting these potholes and using the information to avoid potholes can be effective to reduce the number of these fatalities.

A pothole is defined as a bowl-shaped depression in the pavement surface, and its minimum plan dimension is 150 mm [2]. Potholes are formed on the roads because of many reasons like heavy rain, oil spills, road accidents. which make the road difficult to drive upon and causes accidents. So it is very important to get the information, collect them and distribute it to other vehicles and warn the driver. So we thought of building a system in which we can use image processing and machine learning to detect the potholes based on the training provided to it, and alert the driver.

II. Background Read

Most of the content and methods accepted to complete the project were from the learnings of the book Digital image processing, by Gonzalez and Woods. The various techniques have been discussed below :

A. ***Edge detection techniques*** : Edges are the sudden changes in intensity levels, discontinuities or significant transitions in the image. They can be classified as vertical, horizontal and diagonal edges. Detecting edges is important as it gives most of the shape information. Potholes are sudden change in plain road with some intensity changes, so they can be differentiated using edge detection methods like :-

   a. *Sobel's operator* - The sobel operator is used to detect horizontal and vertical edges.. It is a derivative mask and is used for edge detection.

$$\begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix}$$

   b. *Canny edge detector* - It is one of the best edge detection algorithm that minimizes the data to be processed and minimizes the error rate. Sobel's edge detection detects all the edges in an image irrespective of whether its needed or not. We can set a threshold in canny edge detection to reduce the edges to our area of interest. The most important feature of this technique is non-maximum suppression that reduces the edges detected to edges that are in area of interest.

B. ***Gray level thresholding*** : Thresholding is an old, simple and popular technique for image segmentation. Thresholding can be done based on global information (e.g. gray level histogram of the entire image) or it can be done using local information (e.g. co-occurrence matrix) of the image. We used two techniques i.e. global thresholding and otsu's method.

   a. *Global thresholding* - Global means for all, so global thresholding means the threshold will be applicable to the whole image. A threshold is assumed at the beginning from the histogram information, then we search for a appropriate threshold to differentiate between to classes. Once the threshold is found the value that are below threshold will be set to 0 and above threshold to 255. It is a very crude method so it cannot be used in a real scenario.

   b. *Otsu's method* - Otsu's thresholding method involves iterating through all the possible threshold values and calculating a measure of spread for the pixel levels each side of the threshold, i.e. the pixels that either fall in foreground or background. The aim is to find the threshold value where the sum of foreground and background spreads is at its minimum[8]. In Otsu's method we exhaustively search for the threshold that minimizes the intra-class variance (the variance within the class). We used the equations given on wikipedia link [8] to use otsu's method. It gave better results than all other methods.

C. ***Morphological operation*** : We used dilation after canny edge detection to provide a good visual of the pothole. but information provided by dilation depends on the size of structuring element.

D. ***Semantic segmentation*** : The process of taking input as an image and giving output as regions and structures is known as segmentation. Semantic segmentation is recognizing and understanding what's there in the image at pixel level and then differentiating between different objects, structures etc. [9].

III. Approach

This project was about training the machine to recognize a significant pothole. To find out the best technique for this purpose, various methods were used as discussed in the background read.

The simplest and first method thought of was edge detection using Sobel's operator and then according to the theory the best edge detection technique; canny's edge detection was used and it gave good results. Only detecting the edges was not enough so thresholding techniques were used to differentiate a pothole from the road. Simple gray level global thresholding was applied on the test image. Then the next method was otsu's thresholding, that gave best results.

But, to train a machine we needed a network which could remember and recognize the pothole from the road. So semantic segmentation network was used to build a network that could be trained on the dataset. We used image labeler to label different parts of the image containing pothole at pixel level classification and stored this data. Then we built a network with four layers (1)Input layer, (2) Downsampling layer, (3) Upsampling layer and (4) Pixel classification layer. We then trained the network by taking input as the original images and pixel level labeled data. This trained data was stored in a mat file that can be used in any machine to directly test the network. The next step was to test and validate the test results. Test results are shown in results section.

IV. Selection of Data Set

According to the approach followed by us, for techniques from edge detection to otsu's method we used two or three images for each technique to detect the potholes. But to build a system we had to train our model built using semantic segmentation. So we selected the dataset "Pothole dataset" available online from the link mentioned in references[6]. The dataset contains images of potholes(along with their rotated versions), asphalt images, images containing manholes and shadows. These images were needed to help the machine learn and differentiate between humps, potholes, manholes and shadows and therefore alert the driver when a dangerous pothole(with either more depth or width or both) was present in his/her path.

V. Results on Test Images

The results obtained from various methods have been shown below :

Fig. 2. Original image of pothole and asphalt pavement(test image used from the dataset)

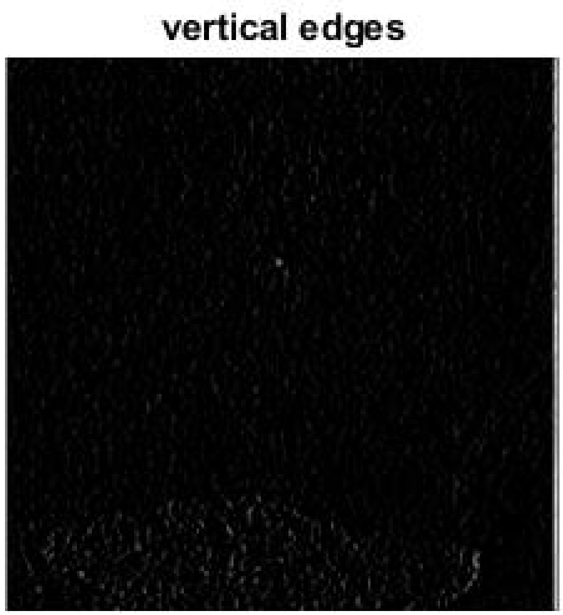


Fig. 3. Result of vertical edge detection using sobel’s operator

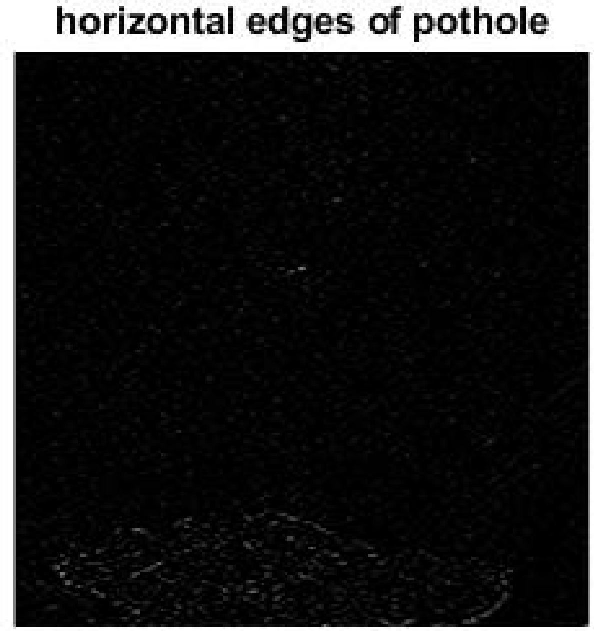


Fig. 4. Result of horizontal edge detection using sobel’s operator

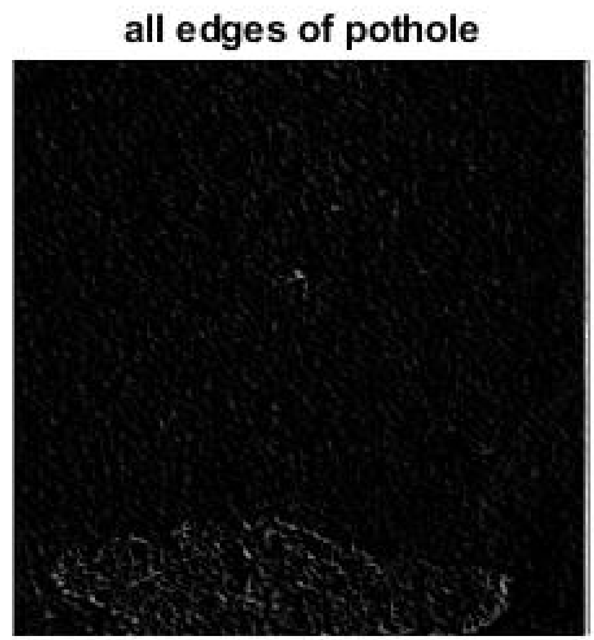


Fig. 5. Result of edge detection using sobel’s operator(all the edges in image are detected)

On using the Sobel’s operator for detection of vertical and horizontal edges and combining the results we got the image given in Fig.5.

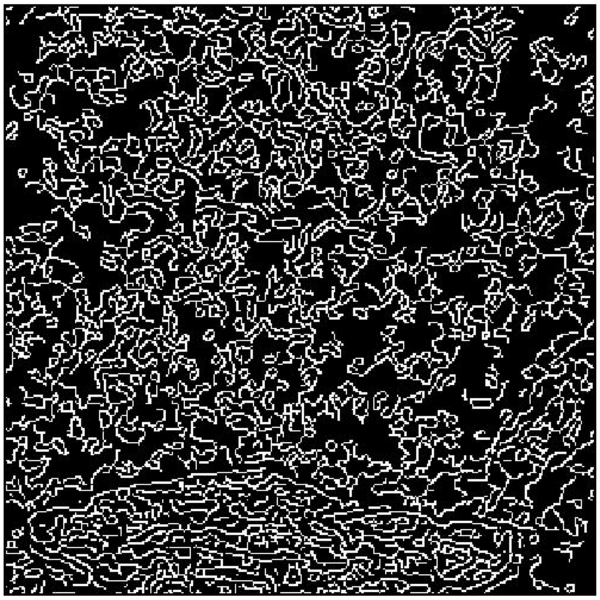


Fig. 6. Result of canny edge detection

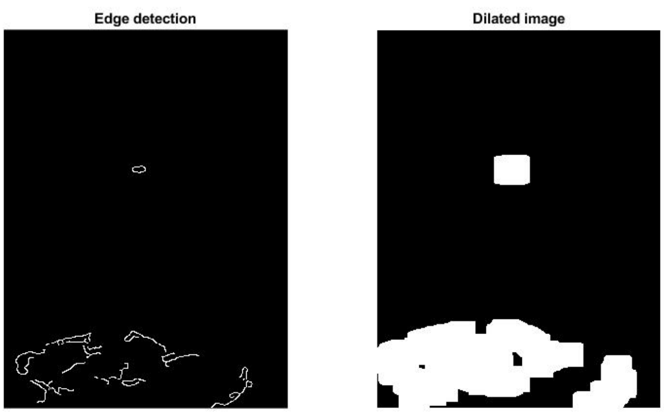


Fig. 7. Result of edge detection using canny edge detector with threshold(left) and dilated further to see the pothole (right)

On applying the canny edge detector and dilation we get better results compared to Sobel operator. Shape and size of potholes are more clear in the results of canny edge detector.

Fig. 8. Results of global thresholding

Global thresholding is done based on global information i.e. gray level histogram of the entire image. It is a very crude method so it cannot be used in a real scenario.

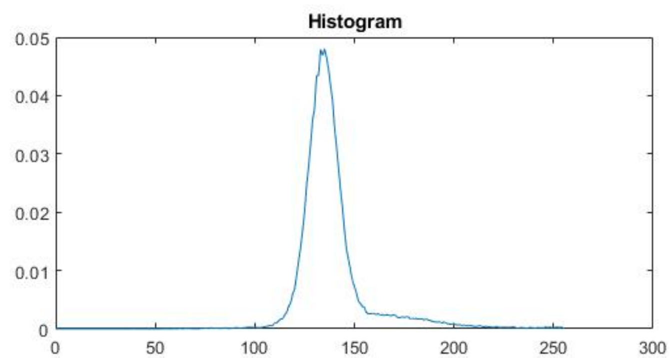


Fig. 9. Histogram of the test image

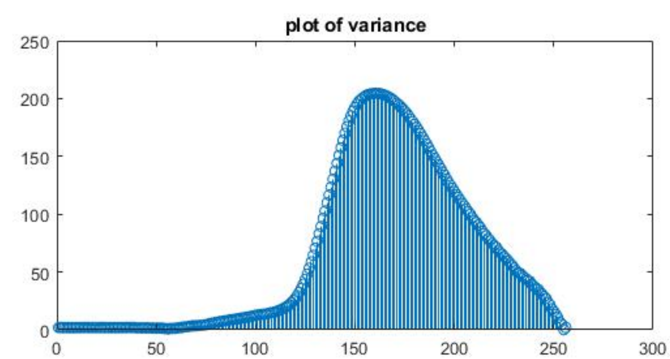


Fig. 10. Plot of the variance obtained in otsu's method

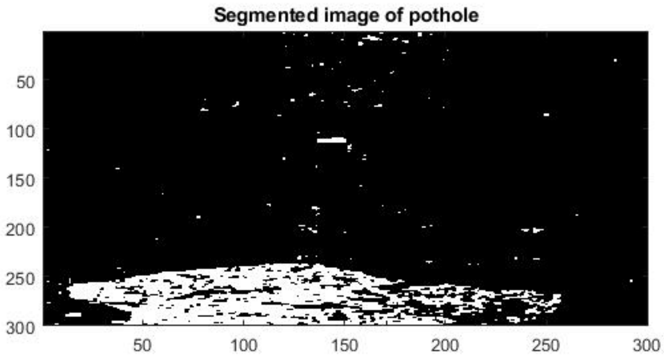
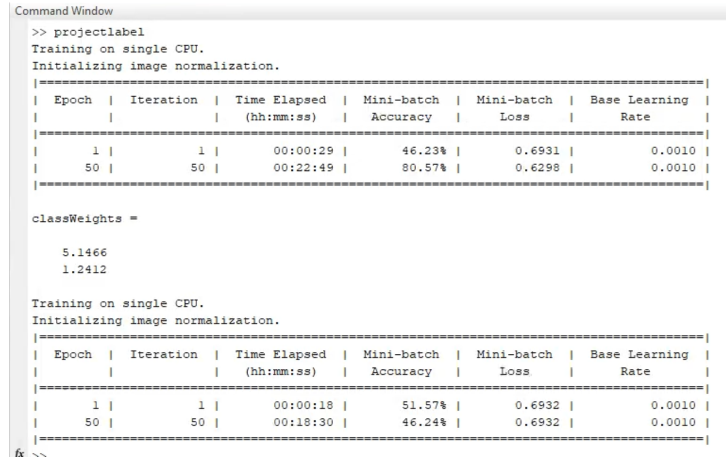


Fig. 11. Results of Otsu segmentation(pothole can be differentiated easily)

Otsu's thresholding method involves iterating through all the possible threshold values and calculating a measure of spread for the pixel levels each side of the threshold, i.e. the pixels that either fall in foreground or background. Otsu's thresholding method gave better results than all other methods.

Fig. 12. Test image used for semantic segmentation



Command Window

```
>> projectlabel
Training on single CPU.
Initializing image normalization.
|========================================================================================|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Mini-batch  |  Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |     Loss     |      Rate       |
|========================================================================================|
|       1 |           1 |       00:00:29 |       46.23% |       0.6931 |          0.0010 |
|      50 |          50 |       00:22:49 |       80.57% |       0.6298 |          0.0010 |
|========================================================================================|

classWeights =

    5.1466
    1.2412

Training on single CPU.
Initializing image normalization.
|========================================================================================|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Mini-batch  |  Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |     Loss     |      Rate       |
|========================================================================================|
|       1 |           1 |       00:00:18 |       51.57% |       0.6932 |          0.0010 |
|      50 |          50 |       00:18:30 |       46.24% |       0.6932 |          0.0010 |
|========================================================================================|
fx >>
```

Fig. 13. Training of semantic network

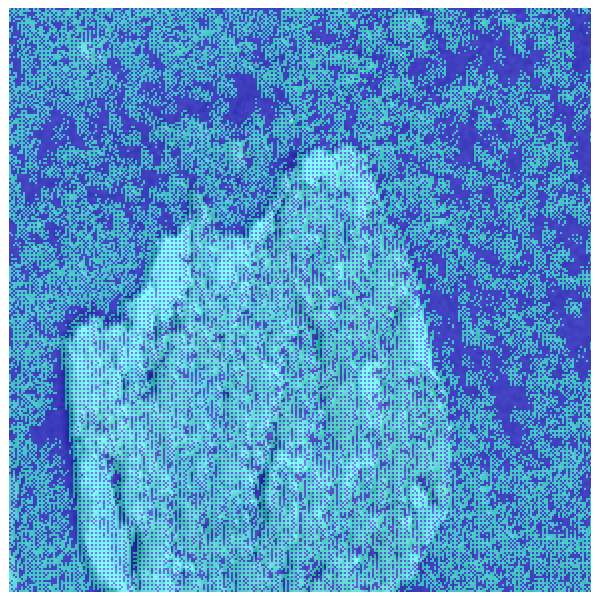


Fig. 14. Semantic segmented image results(light blue region shows the pothole)

Labeling different regions of interest with different label IDs and training the network with a dataset of 240 images gave quite satisfactory results in which pothole region is shown in light blue colour. To further improve the results we can take a larger dataset and train the network for more number of epochs.

VI. Conclusion

Edge-Detection techniques were useful to identify the locations of the potholes but they couldn't provide the accurate information about the shape and size of the potholes. Gray level thresholding improved the results in terms of visual perception. Morphological operations like dilation helped to determine the size and shape of the pothole.

In semantic segmentation the network was trained with a dataset of 240 images. The results on test images were quite satisfactory. In the result potholes are labeled in a different color than asphalt. To further improve the results we can take a larger dataset and train the network for more number of epochs

VII. Future work

The results of the semantic segmentation can be improved by training the network with more number of images with different regions like manholes, shadow, etc. Also by measuring depth of pothole and measuring its distance from the vehicle, we can alert the driver if the pothole is categorised as dangerous and needs to be avoided.

Acknowledgment

We would like to give special thanks to Prof. Amit Sethi Sir for helping us throughout the course and project.

We thank our TA Nikhil for helping us learn new techniques and for his consistent guidance, encouragement and valuable suggestion and support throughout the project.

We extend our sincere thanks to all our friends and our parents for all their love, inspiration and support over the years.

References

[1] Digital Image Processing (4th Edition) by Rafael C. Gonzalez , Richard E. Woods

[2] J. S. Miller and W. Y. Bellinger, “Distress identification manual for the long-term pavement performance program,” FHWA RD-03-031, Federal Highway Administration, Washington, DC, USA, 2003.

[3] NIKHIL R. PAL and SANKAR K. PAL, “A REVIEW ON IMAGE SEGMENTATION TECHNIQUES”, Pattern Recoyinition, Vol. 26, No. 9, pp. 1277 1294. 1993.

[4] *Pothole Detection: An Efficient Vision Based Method Using RGB Color Space Image Segmentation*. Available from: https://www.researchgate.net/publication/317358184_Pothole_Detection_An_Efficient_Vision_Based_Method_Using_RGB_Color_Space_Image_Segmentation [accessed Nov 27 2018].

[5] Dipak K. Dash, “Potholes killed 3,597 across India in 2017, terror 803”, India News - TOI, accessed on 27th november 2018. <http://timesofindia.indiatimes.com/articleshow/64992956.cms?utm_source=contentofinterest&utm_medium=text&utm_campaign=cppst>

[6] Link to pothole dataset used in this project <https://figshare.com/articles/PotDataset/599969>

[7] Leart about many edge detection techniques from <https://www.tutorialspoint.com/dip/concept_of_edge_detection.htm>

[8] Otsu's method of thresholding <http://www.labbookpages.co.uk/software/imgProc/otsuThreshold.html>
<https://en.wikipedia.org/wiki/Otsu's_method>

[9] Semantic segmentation theory - <http://www.cs.toronto.edu/~tingwuwang/semantic_segmentation.pdf>

[10] Semantic segmentation algorithm - <https://in.mathworks.com/help/vision/ug/semantic-segmentation-basics.html>