International Journal of Scientific Research in Science, Engineering and Technology

Print ISSN - 2395-1990
Online ISSN : 2394-4099

Available Online at : www.ijsrset.com
doi : https://doi.org/10.32628/IJSRSET241126

Pothole Detection Using Machine Learning Models

Rahul Dhingra, Mayank Dhingra, Meghna Sharma*
The NorthCap University, Gurgaon, Haryana, India
*Corresponding Author, meghnasharma@ncuindia.edu

ARTICLEINFO

Article History:

Accepted: 10 March 2024
Published: 28 March 2024

Publication Issue :
Volume 11, Issue 2
March-April-2024

Page Number :
94-105

ABSTRACT

Potholes are damage caused to the ground by the formation of water and wear and tear over time. According to statistical data, bad road conditions account for about one- third of the total road accidents which has been increasing exponentially. Potholes have become so common that it has become second nature for people to learn how to spot and avoid them, which causes further accidents. The need of the hour is to build a dependable pothole detection system to accurately detect potholes and warn the drivers and government officials in advance. The process to build such a system is divided into two steps i.e. collection of data and pothole identification. The first step is achieved by taking the data from already available data sets on the Internet. The other step includes labeling the potholes in the data set which is usually done manually. This paper focuses mainly on Visual-based techniques to identify the best detection method by comparing popular Machine Learning models and algorithms. The obtained data set is trained using various transfer learning techniques like You Only Look Once (YOLO)[1] and Single Shot Detector (SSD) [1]. Apart from transfer learning, this paper also focuses on some proposed techniques using Convolutional Neural Net- works (CNN) and classification algorithms like Support Vector Machine (SVM)[21] to identify and localize potholes. The actual size of potholes is calculated using morphological operations, which is a just a straightforward technique to analyze figures using set theory. To analyze every model and find the best model, each model is trained on different sizes of data sets and the obtained result is validated and examined by considering different aspects like speed and accuracy in mind.

Index Terms : YOLO, CNN, Potholes, Data, Machine Learning, Regression, Dilation, Erosion, Closing.

I. INTRODUCTION

The number of occurrences of potholes increases rapidly in extreme weather such as snowfall and heavy rain which eventually impacts traffic safety and road damage. Due to these bad road conditions, the number of accidents has been increased in the last few decades. Automatic pothole identification methods are being studied for efficient road repair and pavement maintenance and have been proved to be an efficient method to solve this problem. There are many solutions available with the advancement in both machine and deep learning methodologies. Some of the main solutions are divided into categories such as Vibration based strategy, 3D reconstruction based Strategy and Vision based Strategy. The vibration based strategies are mathematical oriented and tend to use accelerometers to find a potential pothole by estimating the force that could be applied on it. Moreover, the 3d based strategy, uses some advanced laser scanner devices to reconstruct and plot the all new 3D images for the potholes. Furthermore, the Vision based strategies are a bit expensive as they utilize the real time images being captured from cameras. Using high definition cameras adds up to the overall cost which acts as one of the disadvantages for this technique. Apart from this the main advantage of this method that distinguish it from other strategies is that one doesn't need to go over a pothole in order to collect the data associated with that[6].

We decided to go with Visual based techniques as it was the cheaper option among the three, and there were several data sets available online.The current Visual based techniques only deal with the detection and identification of the pothole on the street. As of now, there are not any research work being completed that considers both the depth as well as the size of the pothole. These statistics are very important features and hence we included them in our research plan which was done in addition to the detection of the pothole data.

II. LITERATURE REVIEW

This research topic holds an important place in the com- munity of computer vision and hence there has been a lot of contributions being made in this research area. Below, is the literature overview which has been used in our research to estimate the pros and cons of the techniques been applied in the past by fellow research scholars. As proposed in [2], the novel approach of Support or State Vector Machines have been used for the detection of the potholes. The histograms of the pothole images were first created and a boundary region of the image was then pulled out. Once the image was extracted, SVM kernel algorithm was then applied to locate the appropriate pothole image. Using this algorithm the target set of images were accurately located.[12]

Furthermore, as in another research work [3]. Deep learn- ing approaches and algorithms were used to complete the same task. CNNs or as known as Convolutional Neural Net- works were used for the classification task. This algorithm proved to be better than previous approaches as this model was not at all affected by the noise and other redundant data which arise due to improper shadows and problems of illuminations in the images when they are captured.[21]

The efforts to use deep learning frameworks were further done by Hiroya Maeda [4], as the research team created a software framework that was able to detect the issue of potholes using their respective mobile devices. Convolutional Neural Network algorithm was used in this approach. The team made huge efforts to collect this massive data and attempt was being made to estimate the speed as well as the accuracy with which the detection was made possible by the system developed.

TABLE I: Comparison of work done

Researcher	Technique	Data Source	Method Type
Yu and Yu [1]	Acceleration Records	ICP accelerometer, Pc oscilloscope	Vibration Based
De Zoysa et al [2]	Acceleration and Busnet	MICAz mote with acceleration sensor	Vibration Based
Erikson et al [3]	Accelerometer Data	Three axis acceleration sensor and GPS Devices	Vibration Based
Mednis Et al [4]	Accelerometer sensor of phone	Android smartphones	Vibration Based
Koch and Brilakis [10]	2D images	Vehicle with robot and rear cameras	Vision Based
Jog et al [11]	Video based	Video Data	Vision Based
Lokeshwar et al [12]	Video based	1275 Video Frames	Vision Based
Koch et al [13]	Video based	Video Clips	Vision Based
Chang et al [6]	Using 3D Laser Scanning	3D Laser	3D Reconstruction Methods
Li et al [7]	Using 3D Laser Scanning	3D Laser and digital camera	3D Reconstruction Methods
Wang [5]	Using stereo vision	2 cameras	3D Reconstruction Methods
Jouber et al [8]	Kinetic Sensor	Kinetic and USB Camera	3D Reconstruction Methods
Moazzam et al [9]	Kinetic Sensor	Kinetic sensor	3D Reconstruction Methods

Apart from the traditional approach to use the CNNs in the deep learning framework, some research scholars tried their hands on the Binary Classification techniques in an attempt to predict whether the road image that was consideration belong to the pothole category or the flat normal road[17]. However, this framework required the feature extraction to be done for the images in advance before it was being entered into the classification system.

Many research scholars also, created new networks and framework, such as Cracknet [6]. The unique feature for this new neural network framework as that the widely used pooling layer was not included in this approach. The results were great from this approach and this new proposed system was very accurate and efficient enough to detect the potholes as well as the cracks along with the uneven surface patches on the images of roads[09].

As the size of the image dataset increased, the demand for the automation picked up and many research scholars started using cheap sensors [7] along with traditional deep neural networks with CNN models to process automation of the crack hole detection. This research work introduces a new approach which eliminated the need to perform feature extraction and the feature learning was done in an automated pipeline. Additionally, A. Tedeschi [8] and his team also attempted to develop solution based out for Android mobile devices. This model was real time based and could detect the cracks and holes on the fly.

III. DATA STATISTICS, PRE-PROCESSING AND ANALYSIS

We found a data set online that was created by the Elec- trical and Electronic Department, Stellenbosch University in 2015. The data set was collected by clicking pictures on smartphones by setting it up on the dashboard of a car. The entire data set consisted of two different parts, one was a simple data set and the other was more complex. Due to CPU constraints, We decided to work on the simpler data set. Apart from this data set, some pictures from Google Images and various other available data sets were combined with this data set to make it more accurate and robust.Therefore, the appropriate measures were taken to combine these data sets into one larger data set[17]. To find the accuracy of our Machine Learning models, we need a test data set. The final is divided into two parts i.e. training data set and test data set. These folders are then subdivided into following 2 subfolders 1) Positive- This data set contains images of roads that contain potholes 2) Negative - This data set contains images of roads that do not contain potholes

Since the class labels are important for the machine learning models, hence the image labels are to be created either manually or by automated processes. Rectangular boxes called the bounding boxes around the images were to be created manually for all the training data images. However, to accomplish this task manually, it posed to be a tedious procedure and required the need for the tool. An open source widely used tool, Labellmg was used to accomplish this class labelling task for the potholes detection. A line was dragged along the the required area of the pothole and it made the tedious task quite simpler.[18] Once, the drawing was completed for each image, an XML format based file was created per image. The file had the margin coordinates for the bounding area- that is both the bottom left as well as the top left. Once the XML based files were created they were fed to the machine learning based model to perform the detection and prediction task.

Fig. 1: Constructing of bounding boxes using LabelImg tool

IV. MACHINE LEARNING MODELS

There has been a lot of research done in this area, and it wasn't hard to find object detection algorithms online. After a little research, We decided to go with You Only Look Once (YOLO) and Single Shot Detector (SSD) since they were the fastest among all. Apart from transfer learning, We experimented with the algorithm that we made using Convolutional Neural Networks (CNN) and Support VectorMachine (SVM), which we will discuss later in this paper.

- Image dimensions: 36802760
- Image size: ¡ 2 MB
- Horizontal resolution: 72 dpi
- Vertical resolution: 72 dpi
- Total categories: 2
- Total images: 800
- Training data set size:600
- Validation data set size: 50
- Train-validation split: 90:10
- Test data set size: 200

Fig. 2: Statictics

A. You Only Look Once (YOLO) Algorithm

The single network, based neural frameworks are very fastand performance oriented algorithms. You look Only once, or YOLO is one such algorithm that exploits the vectors beingsetup using bounding boxes of the images. This algorithmis particularly better since they use single networks, not like the other algorithms that use a set of multi layered networks. Using this approach, firstly the image is divided into a huge grid with the size of SXS. in this new image, eachcell has an individual responsibility to find and predict Nboundary based boxes and hence find it's probability of itbeing a pothole. During analysis, it was found that most ofthese boxes computed a very low probability. A user defined threshold limit was being setup and the boundary boxes were removed if their probability was lower than this limit[9].

Once the elimination process is completed the rest of the boxes that passes the probability test were then moved towards the non max suppression mechanism. The aim was toremove the duplicate and similar boxes. YOLO is consideredas the fastest algorithm by many research scholars and they consider the many benefits of this framework, which made this an apt algorithm for this use case problem. With YOLO, the major highlight was that it didn't require any complex structure or pipeline unlike other frameworks and creates a simple solution since the object detection is considered a regression problem.

During the processing stage for this algorithm, new imageswere required and then these new images set to make new predictions. The YOLO algorithm scans the overall image and treats as a complete item package before it computes theresults for the predictions. Due to this approach, the YOLO algorithm is considered to be the fastest and most widely used algorithm in the computer vision and object detection community[4].

When YOLO, is compared with the other relative frame- works like Sliding window, it is considered better in termsof accuracy as it doesn't consider the scanning if the entire image during the training, processing and prediction analysis. YOLO is often compared with the traditional deep neural net-work algorithm that is CNN. Convolutional neural networks

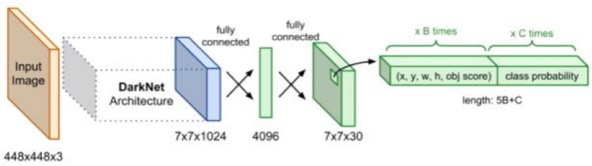

Fig. 3: YOLO ARCHITECTURE

have a relatively higher error rate, when compared since theyconsider the background of the image as well as one of the potential objects, since it is not able to treat the picture asa whole entity. Due to this fact, it is not able to derive the concrete and main context out of the image which is under consideration[11].

During recent studies it has been found that YOLO is ableto perform better and produce better results when it was compared with the high end superior algorithms. This is due to the fact that this algorithm learns the generic structure of the considered object and is stuck to analyze the exact shape and dimensions. This add on feature helps in making a very good prediction analysis for the natural photos which usuallyhave objects with varying shapes and sizes.

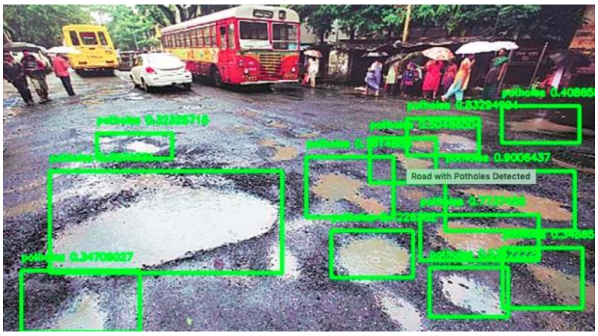

Fig. 4: YOLO Detection Example

Pros

- Being open source, anyone is able to access and utilizethe YOLO framework
- YOLO has another important advantage that it is ableto predict accurately with great speed.

Cons

- Whenever the close objects are analyzed, the spatial constraints limit their detection.
- The models accuracy takes a hit when the loss function being used in the YOLO model, considers the error aris- ing from the small bounding boxes to be equivalent to that from bigger boxes. This inaccurate error estimation further affects the prediction mechanism[27].

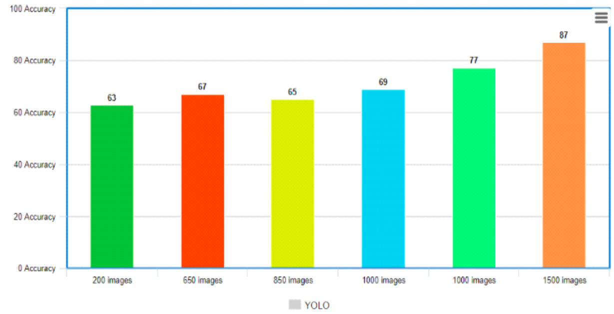

Fig. 5: Accuracy vs Dataset Size for YOLO model

Size	Accuracy	Initial Loss	Final Loss
200 Images	60%	110	4-5
650 Images	67%	109	5-6
850 Images	65%	108	3-4
1000 Images	69%	109	3-4
1100 Images	73%	110	3-4
1500 Images	82%	111	2-3

Fig. 6: YOLO results

B. Single Shot Detector Algorithm:

The SSD algorithm has been a great add on to the computer vision research field. It involves only a simple layer based deep learning neural network and does not attempt to form the cycles like other frameworks. Some fixed boundary boxes are created. A score is provided for detection of the object in that box. Going further with the next steps, the ones that have the maximum overlaps are assigned the maximum score. This is known as the non-maximum compression. The localization as well as the classifications are done likewise YOLO, as they are also done in a single step. In this stage the images are further divided into a cluster of small grids with equalsizes being assigned[17].

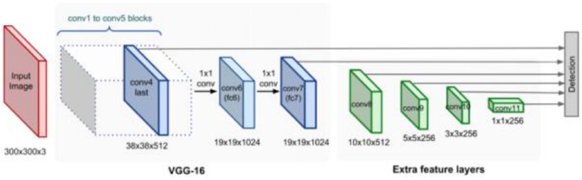

Fig. 7: SSD Architecture

SSD has the major job of matching the class labels with the default boxes as rectangles that are dashed. A match is reached when IOU value is more than 0.5 and each number being considered is mapped to a feature map. A VGG-

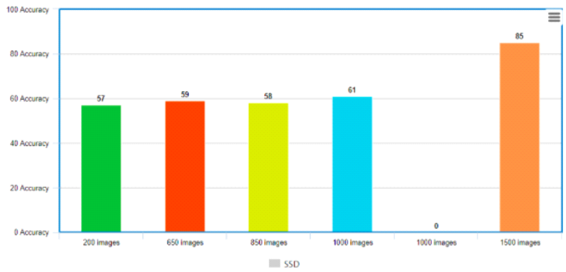

Fig. 8: Accuracy vs Dataset Size for SSD model

16 architecture has the benefit of being performance drive architecture for classification tasks and is mostly used in the problems associated with the transfer learning, hence SSD is used with this architecture for our analysis. Also, As per the figure shared above, SSD architecture does not use the fully connected layers when the VGG model is used[16]. In order to decrease the size of the inputs in some of the consecutive neural network layers, some auxiliary CNN layers were added. These layers were able to pull out various featuresin many different scaling dimensions.

- To define the measure of the objectness of the bounding box being considered, a parameter called Confidence Loss or CL is used.
- To measure the difference between the actual as well as the predicted bounding box objectness being predicted, a parameter measure called Location Loss or LL was used. LF = Confidence Loss + (Alpha) * (Location Loss) where the (Alpha) is used as a balance parameter and is used to balance out the contribution of the LL that is Location Loss Function parameter[4]

Size	Accuracy	Initial Loss	Final Loss
200 Images	57%	22	5-6
650 Images	59%	17	4-5
850 Images	58%	15	4-5
1000 Images	61%	19	5-7
1100 Images	70%	20	5-6
1500 Images	85%	22	5-6

Fig. 9: SSD results

Pros

- This algorithm proves to be fastest running one when
- it was compared with the other three popular models in the same setting and the same image data set[9].
- The prediction results that are computed are very accu- rate and the performance is great too since the features that are extracted are being taken out of the pool of features with different scales.

Cons

- Small objects that being considered for the analysis are very tedious for the detection and it also bears out on very high cost for the computation.
- When using SSD algorithm, performing the data aug- mentation is one of the key requirements and is animportant process[8].

C. Histogram of Oriented Gradients (HOG) with Support Vector Machine:

The most important feature to differentiate objects in an image is their size. Histogram of OrientedGradients (HOG) is an algorithm that is used to extract the main features from an image to make the final algorithm faster and easy to process. Every input image contains different objects and colors and a histogram is created for each orientation of the picture. It is the directional changein the color and the intensity of the image. It is explained in the steps below:

- In order to preserve some of the important aspects of the image or the features, the images were resized. resize() function from the OpenCV Library was used
- Converting the image to a different color space like LUV, YUV, RGB etc were required, in order to preserve the information. For example, using HLS color scheme the identification of images under shadow can be done.
- Using Numpy library a colored histogram is created. These figures are important since they have most of the information
- Using the hog () function, the HOG model is imple- mented. For instance, HOG visualization is as follow- ing:

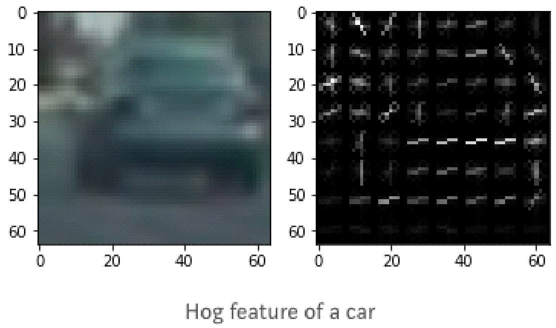

Fig. 10: HOG Feature of a car

After the feature extraction step is complete, we use SVM to train the classifier. The accuracy of SVM when we have to classify two classes is very high. Also, our dataset is not very large, so we need an algorithm that performs very well with smaller datasets.

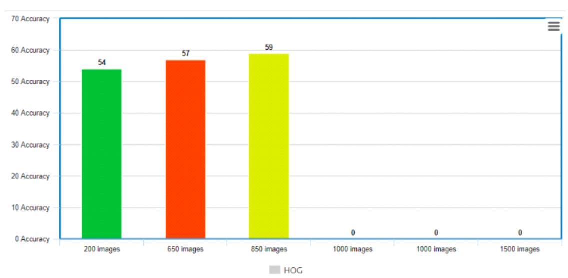

Fig. 11: Accuracy vs Dataset Size for HOG model

Size	Accuracy
200 Images	72%
650 Images	71%
850 Images	67%
1000 Images	69%
1100 Images	60%
1500 Images	74%

Fig. 12: HOG results

D. Convolution Neural Network:

These kind of models requires a lot of experimentation since finding the numberof layers to be used is very hard. We started with just asingle hidden layer and increased the number of layers until we achieved the maximum accuracy. we ended up using 10 layers which include 3 alternate pairs of convolutional layers and pooling layer which was followed by a total number of four fully connected layers. The purpose of all the above mentioned layers are explained below:

- Convolutionl layer: To reduce the input size by lowering the number of image dimensions.
- Pooling layer: To reduce the size of input date.
- Fully connected layer: In this layer every node in the previous layer is connected to every node of the next layer which eventually helps the classification process[4].

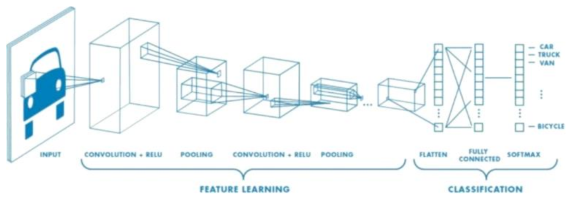

Fig. 13: CNN model Architecture

Size	Accuracy
200 Images	54%
650 Images	57%
850 Images	59%

Fig. 14 : CNN result

Advantages

- This method uses more complex mechanism thanYOLO which eventually makes it slower.
- This model, which is a prerequisite for SSD, does not require extensive data increase[1].
- Disadvantages
- Training time is more than it is appropriate to train two different networks.

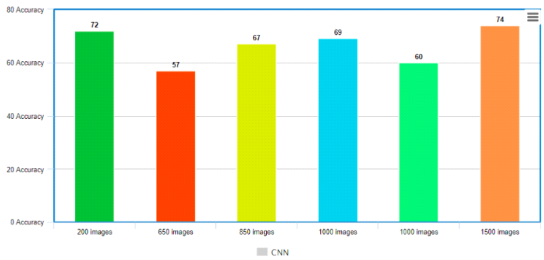

Fig. 15: Accuracy vs Dataset Size for CNN model

V. Justification of models selected

For the reasons listed below, we selected the YOLO modelas the best model for our pothole detection system:

- Out of the three models being analyzed, You look only once or YOLO, has proved to be the fastest one because our criteria for selection involved limitations on the computation.
- When objects for all different sizes are being considered YOLO's performance is decent enough to use for that use case. YOLO is particularly good for large objects as well where other algorithms fails usually.
- When a HPC or High Performance Computing system was used to train the model, the GPU time for YOLO was manageable and was not too high as well[5].
- YOLO's model was considered for a set of about 1000 images that were curated for the roads with holes. The hyperparameter, learning Rate of about 0.01 was used. To test the reliability and precision, other performance metrics like F-1 Score was used.
- The template being setup, in real time since the pro- cessing speed is about 45 frames per second.
- When using YOLOv3 over YOLO, the precision was improved and accuracy was also well.
- The position errors were also reduced since the change in MAP were done.
- Whenever any new function pyramid, is added which usually is done when the images for different scales were used, the predictive analysis was improved.

VI. Size Calculation of Potholes

It is impossible the calculate the depth of the pothole with just the detected pothole image. To find the actual depth, we used various morphological operations which are nothing but techniques to process geometrical structures using set theory.It is really hard to apply these operations on RGB images which is why the RGB images were converted to gray scale images. After this conversion, it is clearly seen that there are a lot of unwanted edges and lines due to various reasons like improper lighting or other objects in the image.

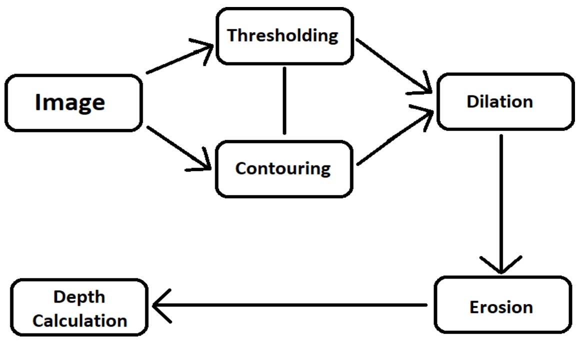

Fig. 16: Flow Chart of size calculation process

To remove the noise in the converted image, dilation was used which is explained below.

- Edge Detection are the set of processes to identify points in the image where the change in the brightness is sharp and is not continuous. These points are put together into a set of lines called edges.
- Dilation process removes the extra unwanted edges from the gray scale image.
- Erosion shrinks the objects in the gray scale image.
- Thresholds were used to track the edges and if the value is low, this shows that there are more edges.
- In order to not destroy the original shape of the image, Closing is used to increase the boundary of the bright regions.

After the coordinates prediction, the image is converted toa black and white image and it is clearly visible that the depth

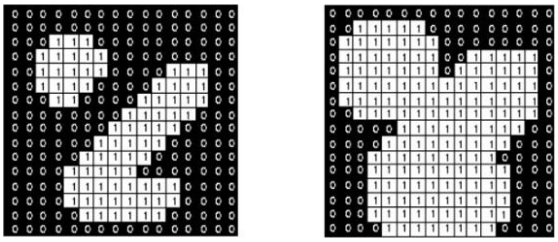

Fig. 17: Dilation process

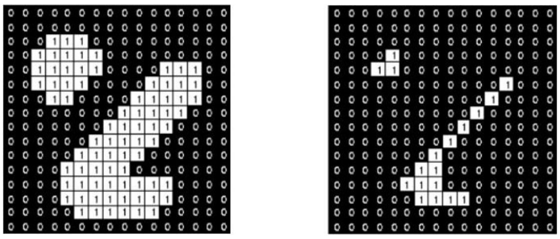

Fig. 18: Erosion process

of the potholes can be defined by the maximum number of continuous black pixels in the vertical direction. Since an image is nothing but an array, a nested for loop was used to find the depth of the pothole. There were a lot refinements that were needed to be made before finally calculating the depth. To find the best morphological operation, we experimented with a lot of kernel sizes and operations and got the best results after using a 9x9 kernel using a single iteration. After applying these operation, we still did not havethe exact size of the potholes and just had the size of potholesin pixels which clearly wasn't going to solve the problem. To solve this issue, multiple regression can be used by passing the size of the potholes in pixels and distance of the object from the bottom of the image as inputs and get the final actual size of the pothole.

Fig. 19: Original Image

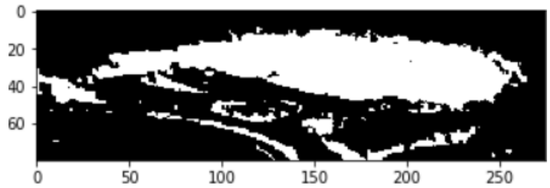

Fig. 20: After Black and white conversion

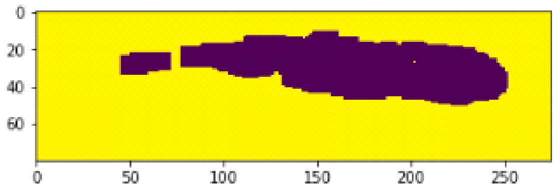

Fig. 21: Closing process

VII. COMPARISON

Although YOLO and SSD have a lot in common, we needed to find a single model that would best suit the project.SSD, being the model with most number of layers has the advantage of the best accuracy which also makes it the slowest among all. If we only talk about the accuracy, CNN is clealy out of the competition. The question arises, YOLO or SSD? Since all the work is needed to be done in realtime, we need a model with faster processing power whichis why YOLO would outperform in that area As shown in

Fig. 22: Speed vs Accuracy of different algorithms

the graph, YOLO is the fastest and has decent accuracy. On the other hand, SSD is comapratively slower which makes it unfit for this project. The accuracy of a model depends onthe size of the pothole in the image as shown in the figure. Overall, YOLO would be the best fit for a proect like this.

The difference in accuracy increases as the object sizes becomes smaller. For the smaller objects, the performance ofthe YOLO is the best and next comes SSD and then CNN. The table below shows the mean average precision , frames per second and the GPU time needed by different models. It is clear that the training time needed for the SDD model is the highest.

VIII. CONCLUSION AND LEARNINGS

We decided to train the models using YOLO, SSD, HOG and CNNs as they were the most reliable and robust object detection algorithms out there. we worked with two already built models which were YOLO and SSD and created twoof our own models using HOG and CNNs. Out of all the models, the best suited model to solve this particular problemturned out to be YOLO with 82% accuracy. we noticed that as we increased the data to train the model, the accuracy increased which clearly indicates that the all the models were data starved. There is possibility the increase the accuracy even further. Due to the limited size of the dataset, we were able to achieve the accuracy of 82%.

Size	YOLO	SSD	HOG	CNN
200 Images	3 hours	4 hours	2 hours	2 hours
650 Images	4 hours	5 hours	3.5 hours	2.5 hours
850 Images	4 hours	5.5 hours	4 hours	3 hours
1000 Images	4 hours	6 hours	-	3.5 hours
1100 Images	4.5 hours	-	-	4 hours
1500 Images	5 hours	8.5 hours	-	6 hours

Fig. 23: Accuracy vs size of the objects

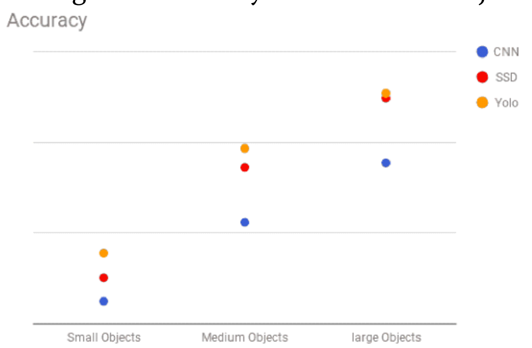

Fig. 24: Accuracy vs size of the objects

V. REFERENCES

[1]. Anon, (2019). [online] Available at: https://www.pothole.info/the- facts/ [Accessed 13 Mar. 2019].

[2]. J. Lin, Y. Liu, ”Potholes detection based on SVM in the pavement distress image”, Appl. Bus. Eng. Sci, pp. 544-547, Aug. 2010.

[3]. YoungJin Cha, Wooram Choi, Oral Bykztrk, ”Deep LearningBased Crack Damage Detection Using Convolutional Neural Networks”, 2017.

[4]. Hiroya Maeda, Yoshihide Sekimoto, Toshikazu Seto, Takehiro Kashiyama, Hiroshi Omata, Road Damage Detection Using Deep Neural Networks with Images Captured Through a Smartphone, 4-6-1 Komaba, Tokyo, Japan:University of Tokyo.

[5]. Justin Bray, Brijesh Verma, Xue Li, Wade He, ”A Neural Network based Technique for Automatic Classification of Road Cracks”, 2006 International Joint Conference on Neural Networks Sheraton Vancou- ver Wall Centre Hotel, July 16-21, 2006.

[6]. Allen Zhang, Kelvin C. P. Wang, Baoxian Li, Enhui Yang, Xianxing Dai, Yi Peng, Yue Fei, Yang Liu, Joshua Q. Li, Cheng Chen, ”Automated Pixel - Level Pavement Crack Detection on 3D

Asphalt Surfaces Using a Deep-Learning Network", Computer-Aided Civil and Infrastructure Engineering, vol. 00, pp. 1-15, 2017.

[7]. Lei Zhang, Fan Yang, Yimin Daniel Zhang, Ying Julie Zhu, "Road Crack Detection Using Deep Convolutional Neural Network".

[8]. A. Tedeschi, F. Benedetto, "A real-time automatic pavement crack and pothole recognition system for mobile Android-based devices", Advanced Engineering Informatics, vol. 32, pp. 11-25, 2017.

[9]. A. Sachdeva and A. Sachdeva, "YOLO - 'You only look once' for Object Detection explained," Medium, 26-Mar-2017. [Online]. Available: https://medium.com/diaryofawannapreneur/yolo-you-only- look-once-for-object-detection-explained-6f80ea7aaa1e. [Accessed: 13-Mar-2019].

[10]. M. Hollemans, Real-time object detection with YOLO. [On- line]. Available: http://machinethink.net/blog/object-detection-with- HYPERLINK "http://machinethink.net/blog/object-detection-with-%20yolo/" HYPERLINK "http://machinethink.net/blog/object-detection-with-%20yolo/"yolo/. [Accessed: 13-Mar-2019].

[11]. Liu, Wei, Anguelov, Dragomir, Erhan, Dumitru, Scott, Cheng- Yang, Berg, A. C., and Reed, "SSD: Single Shot MultiBox Detector," arXiv.org, 29-Dec-2016. [Online]. Available: https://arxiv.org/abs/1512.02325. [Accessed: 13-Mar-2019].

[12]. Byeong-ho Kang , Su-il Choi, "Pothole detection system using 2D LiDAR and camera", 2017 Ninth International Conference on Ubiq- uitous and Future Networks (ICUFN)

[13]. Sudarshan S. Rode , Shonil Vijay , Prakhar Goyal, Purushottam Kulkarni, Kavi Arya, "Pothole Detection and Warning System: Infras- tructure Support and System Design", 2009 International Conference on Electronic Computer Technology

[14]. X. Yu, E. Salari, "Pavement pothole detection and severity measure- ment using laser imaging", 2011 IEEE INTERNATIONAL CONFER- ENCE ON ELECTRO/INFORMATION TECHNOLOGY

[15]. Yaqi Li, Christos Papachristou, Daniel Weyer, "Road Pothole Detec- tion System Based on Stereo Vision", NAECON 2018 - IEEE National Aerospace and Electronics Conference

[16]. Alfandino Rasyid, Mochammad Rifki Ulil Albaab, Muhammad Fajrul Falah, Yohanes Yohanie Fridelin Panduman, Alviansyah Arman Yusuf, Dwi Kurnia Basuki, Anang Tjahjono, Rizqi Putri Nourma Budiarti, Sritrusta Sukaridhoto, Firman Yudianto, Hendro Wicaksono, "Pothole Visual Detection using Machine Learning Method integrated with Internet of Thing Video Streaming Platform", IEEE, 2017

[17]. Vinay Rishiwal, Hamshan Khan, "Automatic pothole and speed breaker detection using android system",2016 39th International Con- vention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)

[18]. Ya-Wen Hsu, Jau-Woei Perng, Zong-Han Wu, "Design and imple- mentation of an intelligent road detection system with multisensor integration

[19]. Sandeep Venkatesh, E. Abhiram, S. Rajarajeswari, K. M. Sunil Kumar, Shreyas Balakuntala, Nitin Jagadish, "An intelligent system to detect, avoid and maintain potholes: A graph theoretic approach" 2014 Seventh International Conference on Mobile Computing and Ubiquitous Networking (ICMU)

[20]. Sumit Srivastava, Ayush Sharma, Harsh Balot, "Analysis and Improve- ments on Current Pothole Detection Techniques" 2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE)

[21]. Amita Dhiman, Reinhard Klette, "Pothole Detection Using Computer Vision and Learning",IEEE Transactions on Intelligent Transportation Systems, 2019
[22]. Andrew Fox ; B.V.K. Vijaya Kumar ; Jinzhu Chen ; Fan Bai, "Multi- Lane Pothole Detection from Crowdsourced Undersampled Vehicle Sensor Data" Year: 2017 — Volume: 16, Issue: 12 — Journal Article Publisher: IEEE
[23]. Ricardo Silveira Rodrigues, Marcia Pasin, Alice Kozakevicius, Vini- cius Monego, "Pothole Detection in Asphalt: An Automated Approach to Threshold Computation Based on the Haar Wavelet Transform", 2019 IEEE 43rd Annual Computer Software and Applications Con- ference (COMPSAC)
[24]. Mae M. Garcillanosa, Jian Mikee L. Pacheco, Rowie E. Reyes, Junelle Joy P. San Juan, "Smart Detection and Reporting of Potholes via Image-Processing using Raspberry-Pi Microcontroller", Year: 2018 — Conference Paper — Publisher: IEEE
[25]. Tran Duc Chung, M. K. A. Ahamed Khan, "Watershed-based Real- time Image Processing for Multi-Potholes Detection on Asphalt Road" 2019 IEEE 9th International Conference on System Engineering and Technology (ICSET)
[26]. Sung Won Lee ; SeokJin Kim, Jeong Han, Kwang Eun An, Seung-Ki Ryu , Dongmahn Seo, "Experiment of Image Processing Algorithm for Efficient Pothole Detection" 2019 IEEE International Conference on Consumer Electronics (ICCE)
[27]. Rui Fan ; Umar Ozgunalp ; Brett Hosking ; Ming Liu ; Ioannis Pitas "Pothole Detection Based on Disparity Transformation and Road Surface Modeling ",IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium