Live Pothole Detection Using Machine Learning

Submitted in partial fulfillment of the requirements
of the degree of

BACHELOR OF ENGINEERING

In

COMPUTER ENGINEERING

By

Mittul Daswani

Shubham Jaiswal

Tanay Kamath

Vedant Kangde

Guide:

PROF. NABANITA MANDAL
(Assistant Professor, Department of Computer Engineering,TSEC)

THADOMAL SHAHANI
TSEC
ENGINEERING COLLEGE

Computer Engineering Department
Thadomal Shahani Engineering
College University of Mumbai
2022-2023

Live Pothole Detection Using Machine Learning

Submitted in partial fulfillment of the requirements
of the degree of

BACHELOR OF ENGINEERING

In

COMPUTER ENGINEERING

By

Group No: 43

Roll No.	Name
1902031	Mittul Daswani
1902060	Shubham Jaiswal
1902068	Tanay Kamath
1902069	Vedant Kangde

Guide:

PROF. NABANITA MANDAL
(Assistant Professor, Department of Computer Engineering,TSEC)

Computer Engineering Department
Thadomal Shahani Engineering
College University of Mumbai

2022-2023

CERTIFICATE

This is to certify that the project entitled **"Live Pothole Detection Using Machine Learning"** is a bonafide work of

Roll No.	Name
1902031	Mittul Daswani
1902060	Shubham Jaiswal
1902068	Tanay Kamath
1902069	Vedant Kangde

Submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **"BACHELOR OF ENGINEERING"** in **"COMPUTER ENGINEERING"**.

Prof. Nabanita Mandal
Guide

Dr. Tanuja Sarode
Head of Department

Dr. G. T. Thampi
Principal

Project Report Approval for B.E

Project report entitled ***Live Pothole Detection Using Machine Learning*** by

Roll No.	Name
1902031	Mittul Daswani
1902060	Shubham Jaiswal
1902068	Tanay Kamath
1902069	Vedant Kangde

is approved for the degree of ***"BACHELOR OF ENGINEERING" in "COMPUTER ENGINEERING"***.

Examiners

1.________________________________

2.________________________________

Date:

Place:

Declaration

I declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

1) ________________________

Mittul Daswani 1902031

2) ________________________

Shubham Jaiswal 1902060

3) ________________________

Tanay Kamath 1902048

4) ________________________

Vedant Kangde 1902069

Date:

Abstract

In India, around 5000 people die due to accidents caused by potholes. Time and again we see distressing news about how a pothole caused an accident and claimed the lives of some commuters. Potholes not only cause accidents but also make transport slow and clumsy, damage vehicles causing their owners precious repairs and overall are an imminence to our country's logistics and hence is of utmost significance to bring it to our government's attention and fix them as soon as possible.

This composition proposes a system where images are taken from CCTV cameras as well as commuter reports. These images are given as input to the model which identifies roads grounded on their quality and creates a detailed report and as a result displays pothole areas on a chart to commuters which aids in avoiding accidents.

Table of Content

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Pothole detection is critical to accident prevention worldwide. Although many studies have been conducted, there are certain biases or tools for obtaining detector data. We provide a way to apply pothole detection using live feeds from government-installed security cameras and commuter travel reports via an app, and the brackets are done before literacy. After parentheses, individual agencies are notified of road conditions at separate locations.

Rapid advances in technology in recent years have had a major impact on the safety of transportation systems. Smart solutions for transportation systems that aim to improve transportation systems are becoming popular. For business safety, passengers often feel uncomfortable driving on rough roads, especially over potholes in the road. According to statistics from Taiwan's Ministry of Justice, from 2008 to 2011, government compensation is about $240 million. Potholes in the road are detrimental to driver safety. Therefore, a live pothole detection system can be built to improve safety of pedestrians, drivers, etc.[2].

Also, further and further widgets include sensors, compass, gyroscope, GPS, cameras, etc. Several functions use such detectors in mobile. Thus, using mobiles with their sensors and cameras to describe potholes is suitable and accessible.[1] This study proposes a pothole discovery system grounded on mobile sensors, cameras and shares the pothole information with commuters and government. For this purpose, the mobile device should be equipped with g- detectors and gps to collect data and position information as well as a camera to take photos and include it in the report.

1.2 Problem Statement and Objectives

Potholes pose a grave danger to safe travel on the roads. There were over 5000 deaths in India that were caused by potholes. Such bad road conditions pose a great threat to commuters worldwide. Safe traveling and reducing accidents and preventing loss of life is our main aim for this project. So to detect and report potholes we are proposing this project as a solution to detect potholes in real time via a video feed as well as manual reporting by users through our app. Input will be live video feed as well as photos uploaded by users for manual reporting. Output will be an automated system which reports all the information to the relevant authority of the area.

1.3 Scope

The scope of this project is to detect potholes in real time via live feed from CCTV cameras installed by the government at all traffic signals and take photo grabs at certain times of the day at a predetermined frequency and use those images as input for our trained model and categorize them into one of three classes in our model- Normal, Unpaved and Potholes. Apart from the video feed, we also take input from users via our app in which users upload images manually and the location is taken too, then images are fed as input to our model and it classifies them. After classification of images from the above mentioned two sources, we store them in our database and the concerned authorities with respect to the location and classification of the image are notified. This reporting of information to concerned authorities is an automated process in which contact details of the authorities are hard coded in the project and the information is sent out periodically.

Chapter 2

Review of Literature

2.1 Domain Explanation

Domain Explanation for Machine Learning and Web development are as follows:

1. Machine Learning

Machine learning is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so.Machine learning algorithms use historical data as input to predict new output values.
Recommendation engines are a common use case for machine learning. Other popular uses include fraud detection, spam filtering, malware threat detection, business process automation (BPA) and Predictive maintenance.
Supervised learning: In this type of machine learning, data scientists supply algorithms with labeled training data and define the variables they want the algorithm to assess for correlations. Both the input and the output of the algorithm is specified.
Neural Networks
Neural networks, also known as artificial neural networks (ANNs) or simulated neural networks (SNNs), are a subset of machine learning and are at the heart of deep learning algorithms. Their name and structure are inspired by the human brain, mimicking the way that biological neurons signal to one another.
Artificial neural networks (ANNs) consist of node layers, containing an input layer, one or more hidden layers, and an output layer. Each node, or artificial neuron, connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network.

Neural networks rely on training data to learn and improve their accuracy over time. However, once these learning algorithms are fine-tuned for accuracy, they are powerful tools in computer science and artificial intelligence, allowing us to classify and cluster data at ahigh velocity. Tasks in speech recognition or image recognition can take minutes versus hours when compared to the manual identification by human experts. One of the most well-known neural networks is Google's search algorithm.

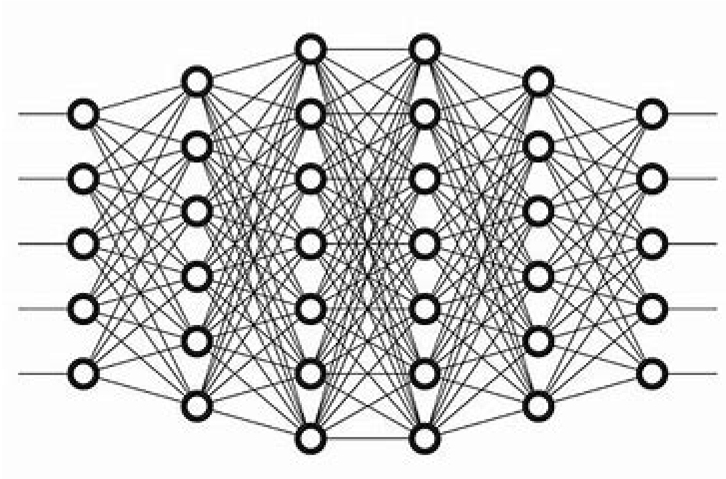


Fig 2.1 Neural Network Diagram

Convolutional Neural Network

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the ability to learn these filters/characteristics.The architecture of a ConvNet

is analogous to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex. Individual neurons respond to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of such fields overlap to cover the entire visual area.

Layers in CNN :

Convolution Layer

The convolution layer is the core building block of the CNN. It carries the main portion of the network's computational load.

This layer performs a dot product between two matrices, where one matrix is the set of learnable parameters otherwise known as a kernel, and the other matrix is the restricted portion of the receptive field. The kernel is spatially smaller than an image but is more in-depth. This means that, if the image is composed of three (RGB) channels, the kernel height and width will be spatially small, but the depth extends up to all three channels.

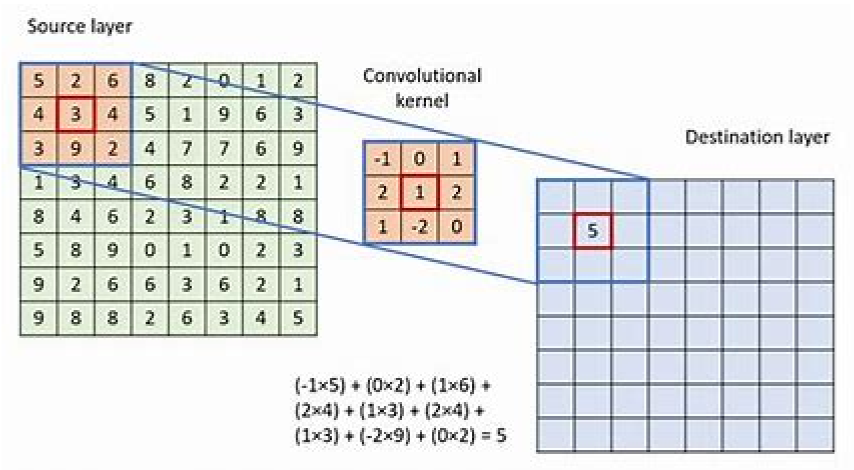


Fig 2.2 Convolution Layer

Pooling Layer

The pooling layer replaces the output of the network at certain locations by deriving a summary statistic of the nearby outputs. This helps in reducing the spatial size of the representation, which decreases the required amount of computation and weights. The pooling operation is processed on every slice of the representation individually.

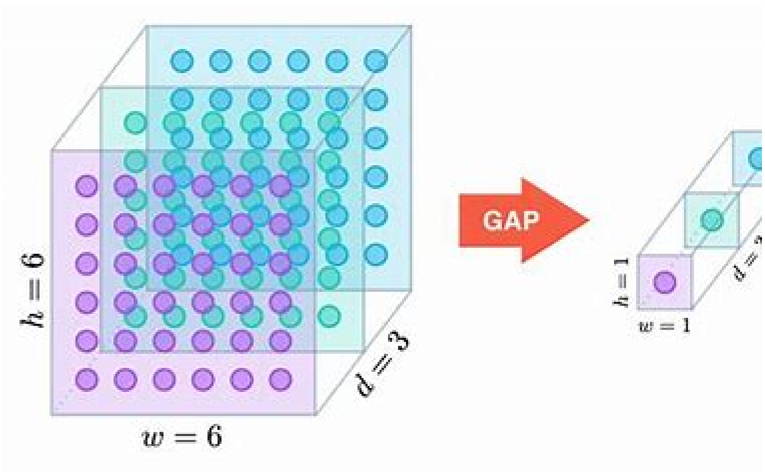


Fig 2.3 Pooling Layer

Fully Connected Layer

Neurons in this layer have full connectivity with all neurons in the preceding and succeeding layer as seen in regular FCNN. This is why it can be computed as usual by a matrix multiplication followed by a bias effect.

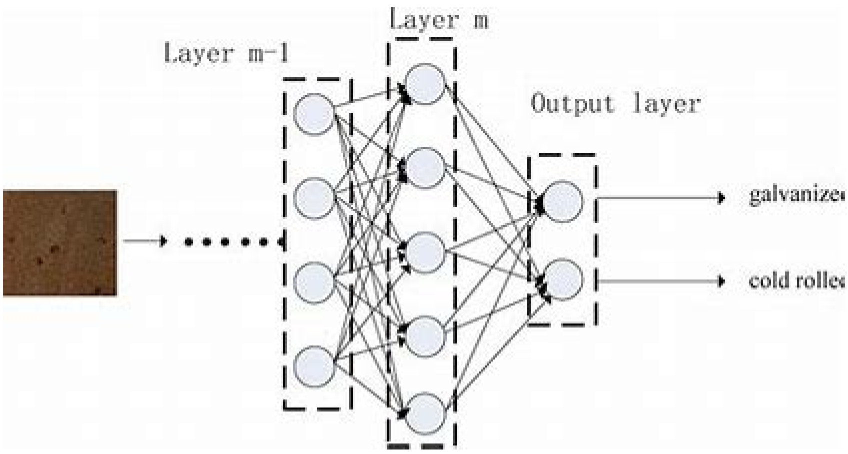


Fig 2.4 Fully Connected Layer

Computer Vision

Computer vision is a field of artificial intelligence (AI) that enables computers and systems to derive meaningful information from digital images, videos and other visual inputs — and take actions or make recommendations based on that information. If AI enables computers to think, computer vision enables them to see, observe and understand.

Computer vision works much the same as human vision, except humans have a head start. Human sight has the advantage of lifetimes of context to train how to tell objects apart, how far away they are, whether they are moving and whether there is something wrong in an image.

Computer vision trains machines to perform these functions, but it has to do it in much less time with cameras, data and algorithms rather than retinas, optic nerves and a visual cortex. Because a system trained to inspect products or watch a production asset can analyze thousands of products or processes a minute, noticing imperceptible defects or issues, it can quickly surpass human capabilities.

Android Application

An Android app is a software application running on the Android platform. Because the Android platform is built for mobile devices, a typical Android app is designed for a smartphone or a tablet PC running on the Android OS.Android apps are written in the Java programming language and use Java core libraries. They are first compiled to Dalvik executables to run on the Dalvik virtual machine, which is a virtual machine specially designed for mobile devices.

2. Web Development

Streamlit

Streamlit is a free and open-source framework to rapidly build and share beautiful machine learning and data science web apps. It is a Python-based library specifically designed for machine learning engineers. Data scientists or machine learning engineers are not web developers and they're not interested in spending weeks learning to use these frameworks to build web apps. Instead, they want a tool that is easier to learn and to use, as long as it can display data and collect needed parameters for modeling. Streamlit allows you to create a stunning-looking application with only a few lines of code.

The best thing about Streamlit is that you don't even need to know the basics of web development to get started or to create your first web application. So if you're somebody who's into data science and you want to deploy your models easily, quickly, and with only a few lines of code, Streamlit is a good fit.

PostgreSQL and PostGIS

PostgreSQL is a powerful, open source object-relational database system that uses and extends the SQL language combined with many features that safely store and scale the most complicated data workloads. The origins of PostgreSQL date back to 1986 as part of the

POSTGRES project at the University of California at Berkeley and has more than 35 years of active development on the core platform.

PostgreSQL has earned a strong reputation for its proven architecture, reliability, data integrity, robust feature set, extensibility, and the dedication of the open source community behind the software to consistently deliver performant and innovative solutions. PostgreSQL runs on all major operating systems, has been ACID-compliant since 2001, and has powerful add-ons such as the popular PostGIS geospatial database extender. It is no surprise that PostgreSQL has become the open source relational database of choice for many people and organisations.

2.2 Review of Existing Systems

Some pothole detection methods have been proposed and can be classified into two groups: image recognition method and mobile sensing method.

2.2.1 Image Recognition Method

Yu and Salari proposed a pothole detection approach based on laser imaging techniques to collect road information. Then the artificial neural network algorithm (ANN) was used to analyze the road information and detect potholes [6]. However, this approach which requires a big computation power to recognize the laser images is unsuitable for mobile devices.

Lin and Liu used the support vector machine algorithm (SVM) to analyze images about road information for pothole detection [4]. Although this approach can provide high accuracy, big computation power is required for image recognition. Therefore, this approach is also unsuitable for mobile devices.A low cost model for analyzing 3D pavement images was proposed, which utilizes a low cost Kinect sensor which gives the direct depth measurements, thereby reducing computing costs[3].

Nericell project used a smartphone based on Windows Mobile operation system which is equipped with G-sensor and GPS to collect and analyze accelerometer data for pothole detection [2]. However, the smartphone in this project should be equipped with the specific angle. Furthermore, this project only considered analyzing z-axis accelerometer data with high misjudgment.

2.2.2 Mobile Sensing Method

For BusNet project, the Gsensor and GPS are equipped in the on-board unit (OBU) in bus to collect accelerometer data and location information. These data can be sent to data processing center via wireless networks, and data processing center can analyze these data to check whether the vectors of accelerometer data exceed the thresholds for pothole detection [5]. However, this approach requires that the batch accelerometer data is sent when bus enters the bus station. Therefore, this approach cannot provide real-time pothole detection information.

The pothole patrol system which was proposed by a project team from Massachusetts Institute of Technology combined G-sensor and GPS. This system analyzed the x- axis accelerometer data and z-axis accelerometer data and equipped five data filters which include (1) speed, (2) highpass, (3) z-peak, (4) xz-ratio, and (5) speed versus z ratio [7]. Although these data filters can detect potholes, only z- peak of data filter can obtain the precise pothole information. However, high misjudgment of z-peak of data filters with the surge of road

Chapter 3

Proposed System

3.1 Analysis/Framework

A use case diagram is a type of UML (Unified Modeling Language) diagram that is used to describe the various interactions and relationships between actors (users, systems, or other entities) and the system under consideration.

In a use case diagram, each use case represents a specific task or goal that the system needs to accomplish, and actors represent the users or other systems that interact with the system to achieve these tasks or goals. The diagram also shows the relationships between the actors and the use cases, indicating which actors are involved in each use case and how they interact with the system.

Use case diagrams are useful in software development because they provide a visual representation of the system's functionality and how it is used by different actors. They can be used to clarify requirements, communicate with stakeholders, and identify potential issues or gaps in the system design.

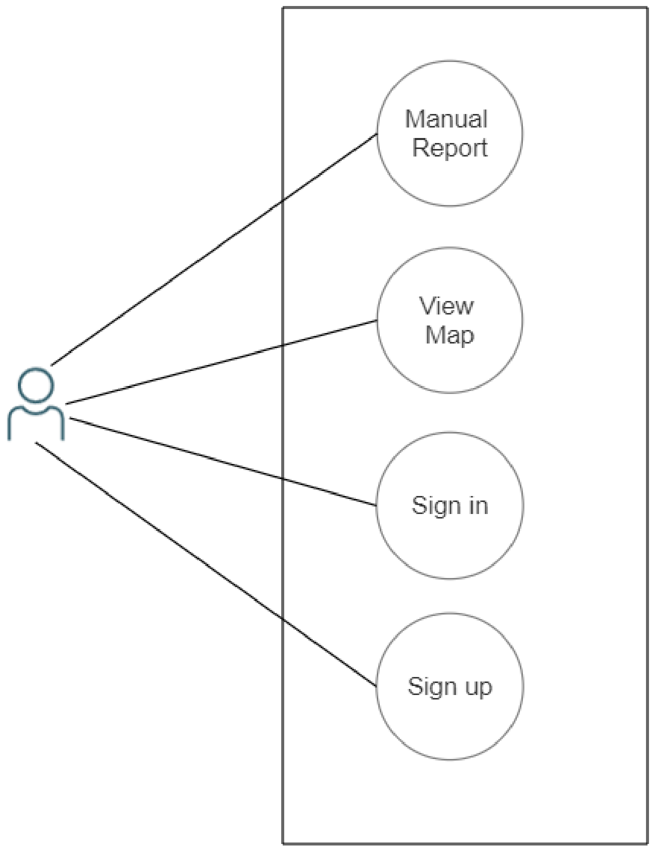


Fig 3.1 Use Case Diagram

3.2 Design Details

Design details typically encompass graphical user interface (GUI) design as well as data flow diagrams (DFDs) and entity-relationship (ER) diagrams, all of which are used to visually represent the architecture and flow of a system or application.

3.2.1 GUI Design

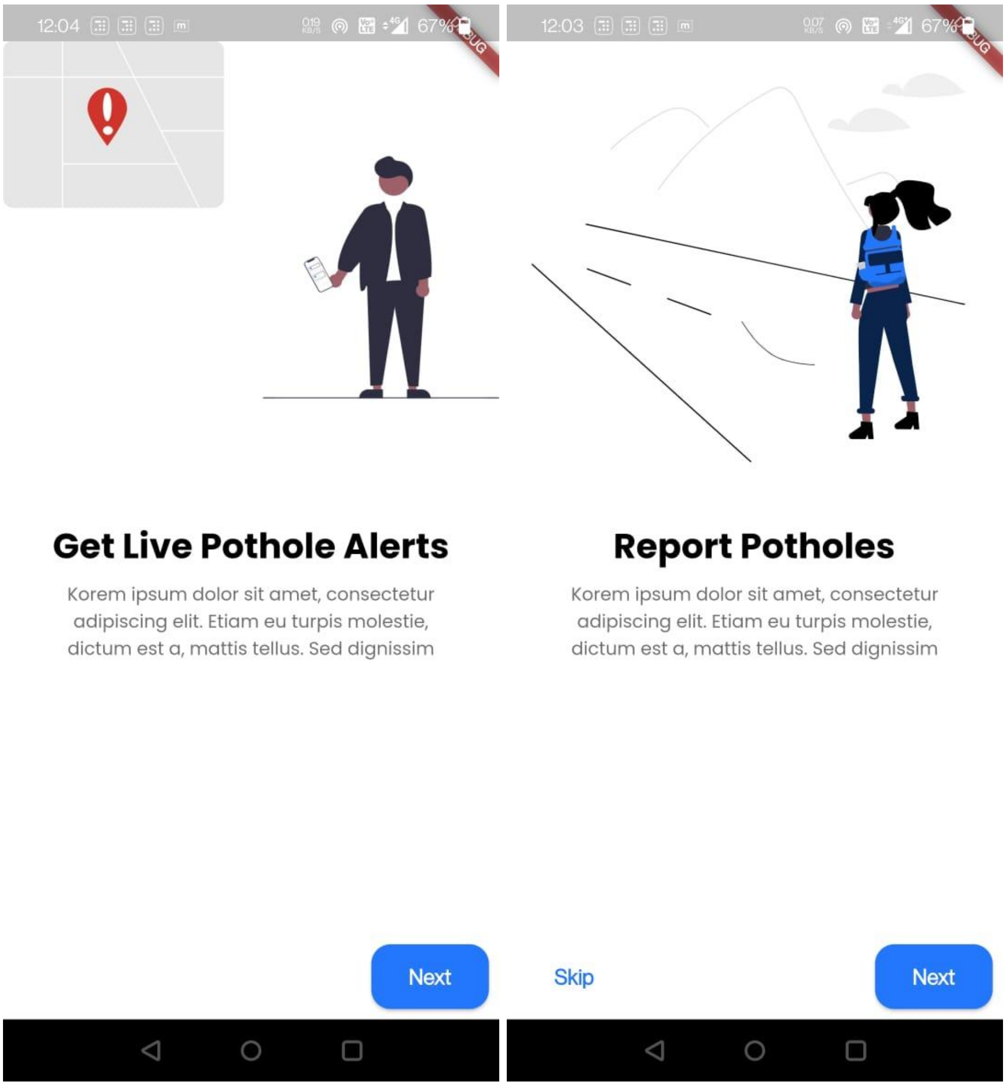


Fig 3.2 Welcome Pages Design

Fig 3.2 shows the Welcome Pages Design of the GUI. This page consists of pages that show the features of our app to help the user understand the features available in our app.

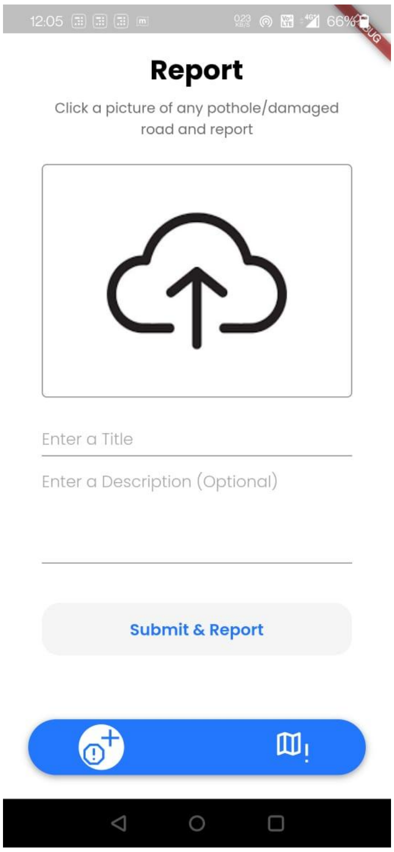


Fig 3.3 Reporting Page Design

Fig 3.3 shows the Home Page Design of the GUI. This page has an option pane which gives the user the options to choose between the reporting page or the map page. The reporting page consists of an option to upload images which can be taken from your photo library, a title for the report the user intends to submit and a description section to mention a brief description if the user wishes to as it’s optional.

3.2.2 Data Flow Diagram (DFD)

A data flow diagram (DFD) is a visual representation that illustrates the flow of data within a system or process. It is commonly used in systems analysis and design to depict how data is input, processed, and output within a system. A DFD typically consists of circles or rectangles, known as "processes," which represent the activities that manipulate data, and arrows, known as "flows," which represent the movement of data between processes, external entities (e.g., users, systems), and data stores (e.g., databases, files).

DFDs are used to model the logical flow of data and provide a clear overview of how data is transformed and exchanged within a system. They can be used to identify data sources, destinations, data transformations, and data stores, and can help in understanding the interactions between different components of a system.

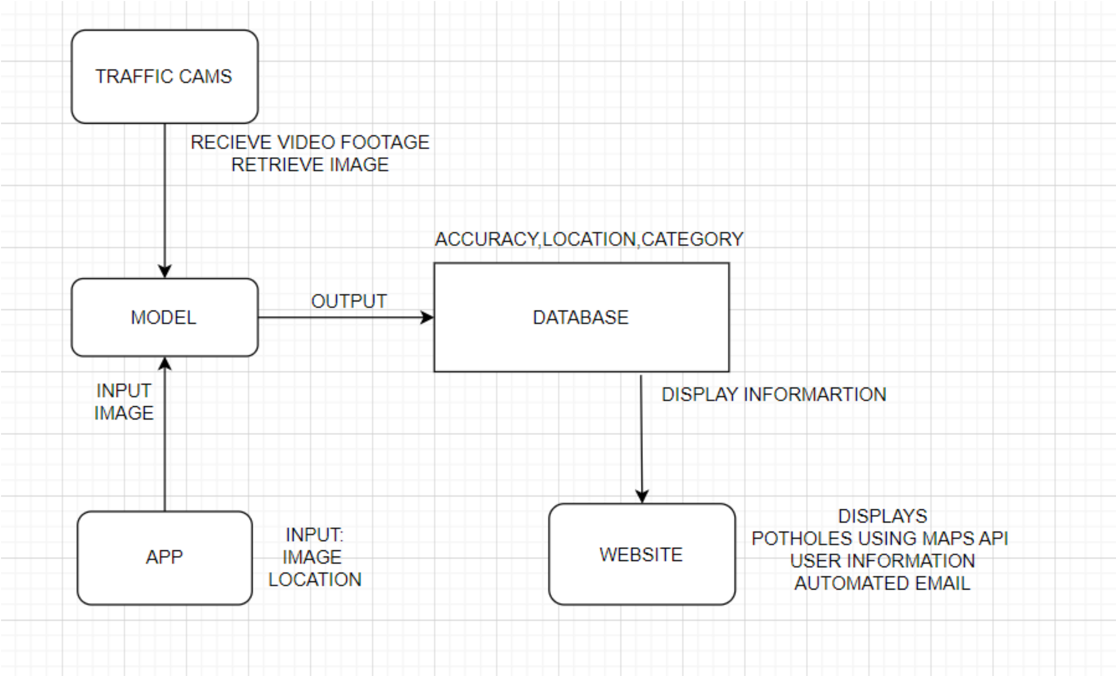


Fig 3.4 Data Flow Diagram

Fig 3.3 represents a Data Flow Diagram. There are 2 modes of input, one is live video feed from traffic cams, the other is manual reporting by users via the app along with location data. Location is taken by traffic camera installation location, or if it’s a manual report, then location is taken using GPS of the mobile device. Upon taking these inputs, they are passed to our CNN model which classifies the input image into 1 of 3 categories- Normal, Unpaved, Pothole. This output is then stored in our database and then reports containing potholes are sent to the respective government authorities automatically using the location data in the input.

3.3 Methodology

The methodology for Live Pothole Detection Using Machine Learning are discussed below:

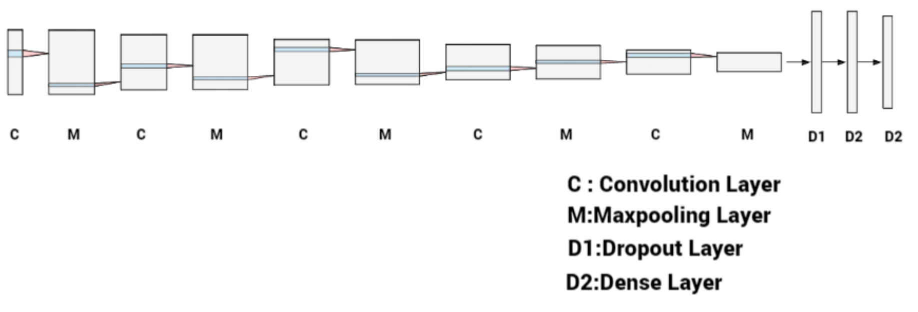


Fig 3.5 Model Architecture

Convolutional Neural Network(CNN)

CNNs are a particular kind of artificial neural network that are used for classifying images into different categories.They are similar to Recurrent neural networks used for sequence words.CNN is one the most popular algorithm for carrying out image recognition and classification tasks .A CNN model needs an image as an input after carrying out various preprocessing methods to remove unwanted information without comprising the key features of an image.A CNN model consists of different layers which are as follows:

Layers in CNN

Convolutional Layer

CNN's fundamental building blocks are convolutional layers. Contrary to the other layers, this layer is in charge of extracting features from the input images, and as a result, it requires more processing power. The process of feature extraction is done by sliding a window of a fixed size over the input image pixel by pixel.In case of RGB images or in other words for an image with more than one channel, the convolution process is done channel by channel.

Pooling Layer

The major purpose of using max pooling layers in a CNN model is to minimize the number of parameters, which significantly decreases the computing power.Hence, pooling layers are generally paired with convolution layers.

Fully Connected Layer

It is known as the output layer of the model.It takes input from the preceding layer and in return computes an uni directional array as an output which has the size as the number of classes.The above process is carried out with the help of matrix multiplication.

Chapter 4

Implementation Details

4.1 Experimental Setup

4.1.1 Dataset Description

The dataset includes the Road Dataset, which consists of 1200 images of Indian roads in various conditions, weather conditions, sharpness, and size. These images generally fall into three categories: normal, unpaved and potholes. As our proposed method mainly focuses on distinguishing potholes from other types of deformations and road conditions, this data set perfectly matches the parameters of our goal. The number of images is large enough to achieve very good accuracy and small enough not to consume excessive processing power and time.

4.1.2 Database Details

PostgreSQL database is used. PostgreSQL is an open source relational database management system (DBMS) developed by a worldwide team of volunteers. PostgreSQL is not controlled by any corporation or other private entity and the source code is available free of charge. relational database system. PostgreSQL supports both SQL (relational) and JSON (non-relational) querying. PostgreSQL is a highly stable database backed by more than 20 years of development by the open-source community. PostgreSQL is used as a primary database for many web applications as well as mobile and analytics applications.

PostGIS is a spatial database extension for the popular PostgreSQL object-relational database.

PostGIS “spatially enables” the PostgreSQL server, allowing it to be used as a back-end spatial database for geographic information systems (GIS) and web-mapping applications in the same manner as Microsoft's SQL Server Spatial and Oracle's Spatial database extension. PostGIS is helping you to store, analyze and manipulate your spatial data.

PostGIS Database is used to store pothole locations reported by user and from traffic cameras. While PostgreSQL Database is used to store user details, Authorities details to which Potholed queries will be send and Potholes analytics.

4.2 Software and Hardware Setup

Hardware Requirements:

Windows from 7 to all further versions, MAC, Linux operating systems. In windows system requirements are usually any core processor may work but specifically, Intel(R) Core (TM) with i5-4300U CPU @ 1.90GHz 2.50 GHz processor, and operating systems can be 32 bit or 64 bit but 64 bit would be more efficient and faster.

Android Phone running Android 4.0 Jellybean or above is required, RAM of at least 4 GB is recommended.

Software Requirements:

Some of the software requirements for the project are -

1. **OS**-windows 7 or above or Linux: An operating system (OS) is system software that manages computer hardware, software resources, and provides common services for computer programs.

2. **Python 3** - Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

3. **NumPy** - NumPy is a Python package which stands for 'Numerical Python'. It is the core library for scientific computing, which contains a powerful n-dimensional array object, provide tools for integrating C, C++ etc.

4. **Scikit-learn** - Scikit-learn is an open source machine learning library that supports supervised and unsupervised learning. It also provides various tools for model fitting, data preprocessing, model selection, model evaluation, and many other utilities.

5. **Keras** - Keras is an API designed for human beings, not machines. Keras follows best practices for reducing cognitive load: it offers consistent & simple APIs, it minimizes the number of user actions required for common use cases, and it provides clear & actionable error messages. It also has extensive documentation and developer guides.

6. **OpenCV** - OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being an Apache 2 licensed product, OpenCV makes it easy for businesses to utilize and modify the code.

7. **TensorFlow**: TensorFlow is a popular open-source Python library that is widely used for machine learning and deep learning tasks. It was developed by Google's Brain Team and is often used by researchers, engineers, and data scientists to build and train artificial neural networks for a wide range of applications.

Chapter 5

Results and Discussion

5.1 Performance Evaluation Parameters

Accuracy

In machine learning, accuracy is a commonly used evaluation metric to measure the performance of a model. It represents the percentage of correctly predicted instances among all instances in a dataset. It is defined as:

Accuracy = (Number of Correct Predictions) / (Total Number of Predictions) — (1)

Precision

Precision is a performance metric that measures the proportion of true positive predictions out of all the positive predictions made by a model. In other words, precision measures how accurate the positive predictions made by the model are.

Precision is calculated as follows:

Precision = True Positives / (True Positives + False Positives) — (2)

True Positives (TP) are the number of correct positive predictions made by the model, and False Positives (FP) are the number of incorrect positive predictions made by the model.

A high precision score indicates that the model is making very few false positive predictions, meaning that the positive predictions are highly likely to be correct. A low precision score, on the other hand, indicates that the model is making many false positive predictions, meaning that the positive predictions are not very reliable.

Recall

Recall is a performance metric that measures the proportion of true positive predictions out of all the actual positive instances in the dataset. In other words, recall measures how well the model can identify all positive instances in the dataset.

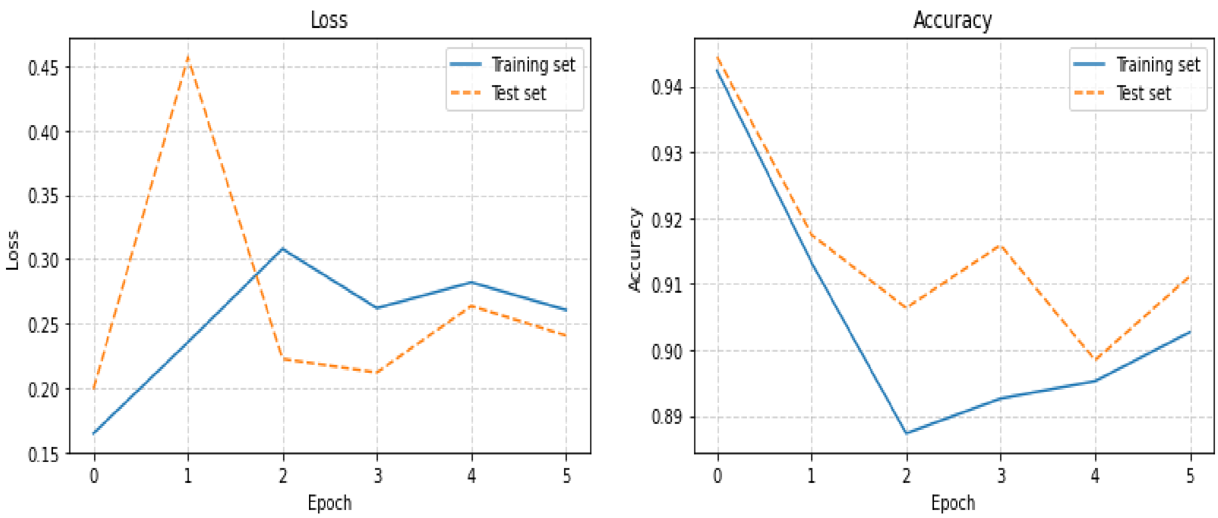
Recall is calculated as follows:

Recall = True Positives / (True Positives + False Negatives) — (3)

where True Positives (TP) are the number of correct positive predictions made by the model, and False Negatives (FN) are the number of positive instances in the dataset that were incorrectly predicted as negative by the model.

A high recall score indicates that the model is correctly identifying most of the positive instances in the dataset, meaning that the model is good at detecting positive instances. A low recall score, on the other hand, indicates that the model is missing many of the positive instances in the dataset, meaning that the model is not very effective at detecting positive instances.

The below graph indicates that there is a rise in model accuracy as the increase in epochs.In contrast with the accuracy the loss function decreases exponentially with the increase in epochs.



5.1 Accuracy and Loss function

The model was tested on using different validation split ratios to better understand the accuracy of the classification model.The table shows different accuracies for different ratios.The best accuracy was achieved at the 75:25 ratio.

Table 5.1 Validation split comparison ratios

Ratios	Accuracy
70:30	94.4%
75:25	94.45%
80:20	93.20%
90:10	90.20%

Table 5.2 Comparison study on factors affecting accuracy

NAME	ACCURACY(%)	AUC(%)	LOSS(%)	PRECISION(%)	RECALL(%)
1 CONVOLUTION 1 MAX POOLING	74.07	73.81	91.97	61.11	61.11
2 CONVOLUTION 2 MAX POOLING	85.24	90.11	1.8028	84.94	84.13
3 CONVOLUTION 3 MAX POOLING	89.74	87.44	18.70	94.91	94.76
4 CONVOLUTION 4 MAX POOLING	94.92	98.53	23.05	92.38	92.38
5 CONVOLUTION 5 MAX POOLING	96.56	98.87	18.70	94.91	94.76

5.2 Implementation Results

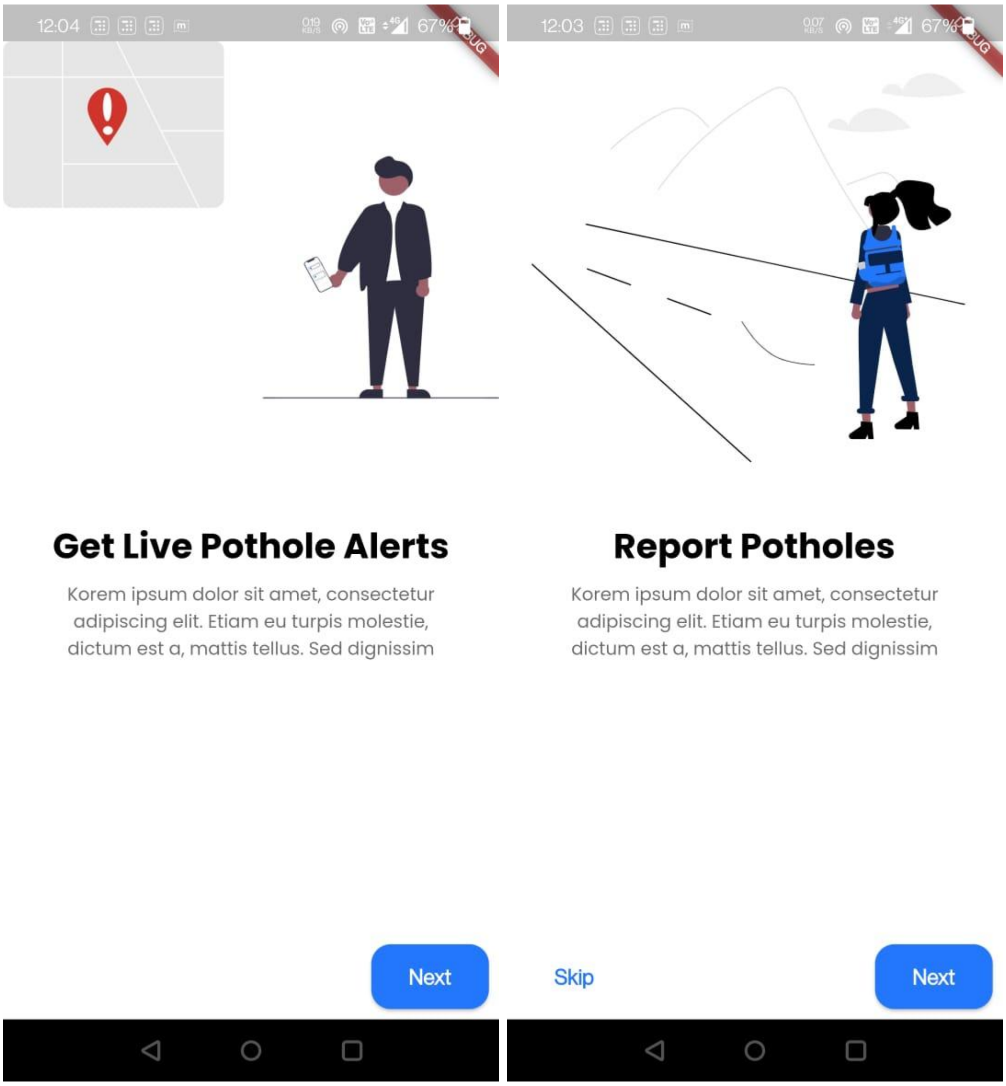


Fig 5.2 Onboarding Screens

Fig 5.2 shows the Welcome Pages Design of the GUI. This page consists of pages that show the features of our app to help the user understand the features available in our app.

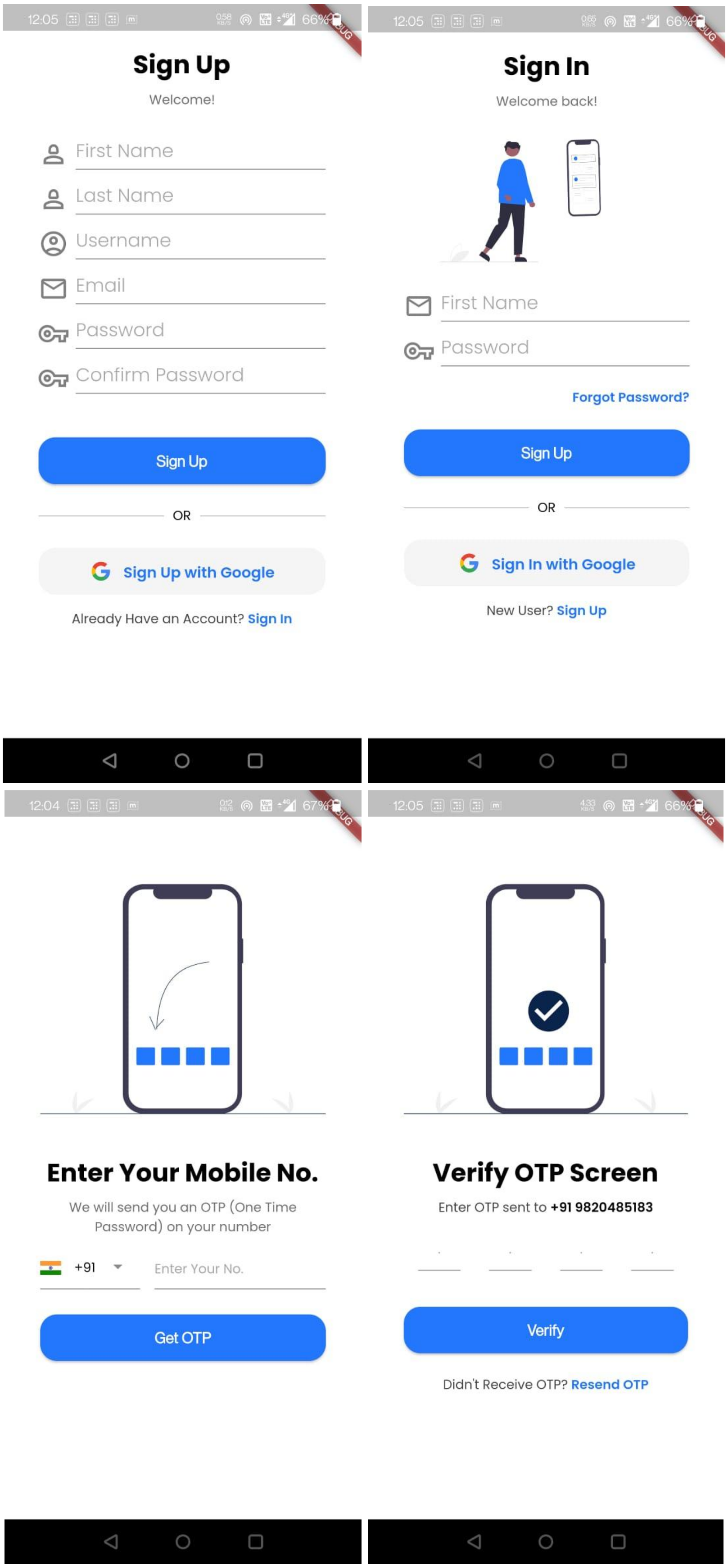


Fig 5.3 Login/Signup Pages Design

Fig 5.2 shows the Login/Signup Pages Design of the GUI. We give users the option to sign up with filling in all the details manually or a google sign in option which eliminates the need to do so. Furthermore, the user phone number is verified using OTP(One Time Password).

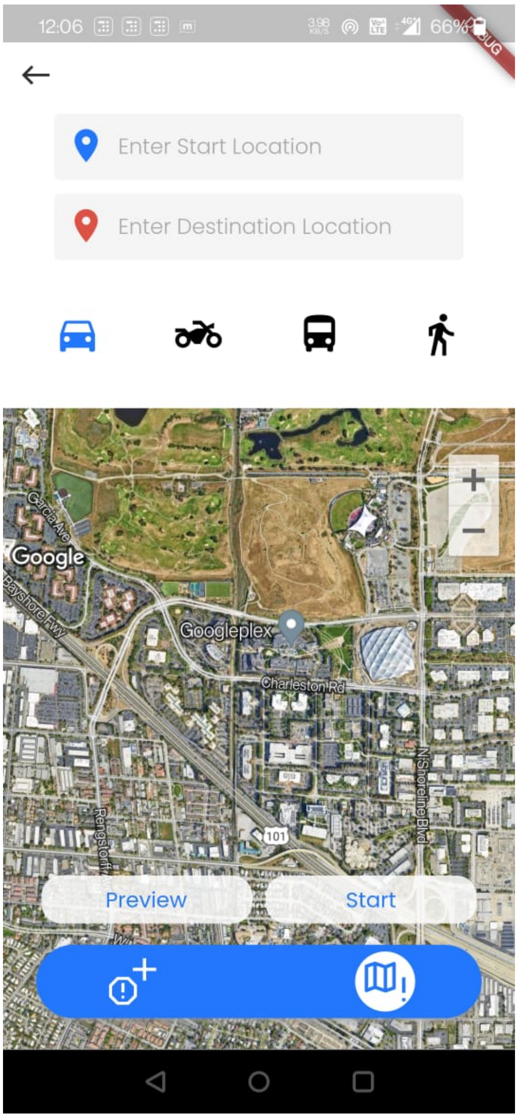


Fig 5.4 Report Pothole Map Tab

Fig 5.5 shows the map page wherein the users can see the potholes detected in real time. Here users can also put in their source and destination places and check for all the potholes they will encounter on the route and it also suggests alternate routes that will have less potholes and hence are safer to use to travel.

5.3 Results Discussion

The performance of different models used in the proposed system for pothole detection was evaluated. The results indicate that accuracy of the model increases with increase in convolution and max pooling layers. And so finally the best accuracy is achieved by a model with 5 convolution and 5 max pooling layers, which is approximately 97%. The proposed system with the aforementioned models and parameters can provide reliable and accurate results for crop management and disease detection. However, the performance of the system can be further improved by adding more diverse data, increasing the dataset size, and fine-tuning the model parameters.

Chapter 6

Conclusion and Future Work

This study proposes a live-time pothole detection method. This method uses Convolutional Neural Network to classify the image into potholes, unpaved, normal road and location of the user is fetched. All this input is maintained in a database and potholes data is sent to respective authorities of a particular state using automation. The proposed approach can help road maintenance authorities to formulate rapid and optimized actions for road infrastructure repairs. A more sophisticated solution with the help of the global position system (GPS) can detect and point out the location of pavement failures.

Future Work of Pothole Detection includes Making Android Application and syncing data using API. And another important feature is Google Map integration to incorporate potholes in the route. Getting video feeds from road cams. This work can further be extended to detect other pavement distresses, road depressions, classify roads as per quality, and depth estimation of potholes.

References

[1]Sukhad Anand ,Gunjan Chugh, Divya Bansal and Sanjeev Sofat, “Road Condition Detection Using Smartphone Sensor: A Survey”, International Journal of Electronic and Electrical Engineering.

[2]in Lin and Yayu Liu Ajit Danti, Jyoti Y. Kulkarni, and P. S. Hiremath, “An Image Processing Approach to Detect Lanes, Potholes and recognize road Signs in Indian Roads”, International Journal of Modeling and Optimization, Vol. 2, No. 6, December2012

[3]Kiran Kumar Vupparaboina and Roopak R. Tamboli "Overtaking & receding vehicle detection for driver assistance and naturalistic driving studies." In Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on, pp. 697-702. IEEE, 2014.

[4]Zhajian Li I. Kolmanovsky, E. Atkins, J. Lu, D. Filev, and J. Michelini, “Cloud aided safety- based route planning,” in 2014 IEEE International Conference on Systems,Man, and Cybernetics, pp. 2525–2530, October2014.

[5]I. Moazzam, K. Kamal, S. Mathavan, S. Usman, and M. Rahman, “Metrology and visualization of potholes using the microsoft Kinect sensor,” in Proc. 16th Int. IEEE Conf. Intell.Transp. Syst., Oct. 2013,pp. 1284–1291

[6]Sachin Bharadwaj Sundra Murthy and Golla Varaprasad, “Automatic Detection and Notification of Potholes and Humps on Roads to Aid Drivers”, IEEE Sensors J., Vol. 15, No. 8, August 2015.

[7]Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep residual learning for image recognition.” In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016

List Of Publication

Vedant Kangde, Mittul Daswani, Shubham Jaiswal, Tanay Kamath, Nabanita Mandal, "Live Pothole Detection: A Machine Learning Based Approach" in Multicon

Acknowledgement

We like to share our sincere gratitude to all those who helped us in completion of this project. During the work, we faced many challenges due to our lack of knowledge and experience. Still, these people helped us to get over all the difficulties and in final compilation of our idea to a shaped sculpture.

We would like to thank **Prof.Nabanita Mandal** and our **HOD Dr.Tanuja Sarode** for their governance and guidance, because of which our whole team was able to learn the minute aspects of the project work. Finally, we would like to thank the Department of Computer Science of Thadomal Shahani Engineering College for providing us such an opportunity to learn from these experiences.

All of our team is thankful to the evaluation committee and the respective faculties.

We are also thankful to our peers and our parents who have inspired us to face all the challenges and overcome all the hurdles faced throughout the project.

Thank you all.

Daswani Mittul

Jaiswal Shubham

Kamath Tanay

Kangde Vedant