POTHOLE DETECTION SYSTEM USING CNN

Alan Shaji

20BRS1024

alan.shaji2020@vitstudent.ac.in

Kritish Palrecha

20BRS1074

kirish.palrecha@vitstudent.ac.in

ABSTRACT:

Potholes are a problem to anyone who uses roads and especially people using two-wheeler vehicles. So, we have made a project to implement a pothole detection system using computer vision and machine learning techniques. The system is designed to detect potholes on any kind of roads, classify them according to their severity, and store their location data and other information in a database that is accessible to the authorities who maintain the roads, to fix them. The system utilizes a camera mounted on the vehicle's front bumper to capture the road surface and processes them using a convolutional neural network (CNN) model to detect potholes. The severity of each detected pothole is determined based on the speed of the vehicle and the time spent by it to cross the pothole. This analysis will help authorities to list the potholes from the ones where the vehicle must slow down to a very low speed to get around it to the ones where the vehicles do not have to slow down much at all. This will help authorities to determine which ones cause a lot of traffic congestion and fix them first.

INTRODUCTION:

Roads are an important part of our daily commute. So, the smoother and better their conditions, the faster and without interruptions, we can reach our destinations. This is the main problem in India. Roads are always filled with potholes. Be it small street roads or highways or expressways, all these roads slowly develop potholes due to reasons like harsh weather in the region, poor quality of road tarring, etc. These potholes can cause a lot of problems like traffic jams, inconvenience to road users and can even lead to fatal accidents. In 2020, India recorded 1481 deaths and 3103 people injured due to potholes. Therefore, their timely detection and repair is very essential to ensure the safety and comfort of the people who drive on these roads.

To address this problem, we have developed a pothole detection system that uses computer vision and machine learning techniques to automate the process of pothole detection. The system is designed to detect potholes on roads and highways, classify them according to their severity, and store their location data in a database. The authorities can access the database anytime and the data will be sorted according to the severity, so that they can fix the ones with high severity.

The objective of this project report is to present the design, implementation, evaluation of the pothole detection system.

OBJECTIVE AND MOTIVATION FOR THE PROJECT:

The objective and the motivation to take up the project is that the authorities are blamed for not fixing roads on time. So, to make the work of the authorities easier and faster by having a system tell you the location of each pothole and a classification system to tell you which ones to fix first was our idea. This will in turn help commuters reach their destinations faster, safer, and more comfortably. The system will effectively detect potholes on the road surface, determine the severity of each pothole and send details about it, like the location, speed of the vehicle, time spent to get around the pothole and the severity of the pothole to a cloud database that is accessible to the authorities. It is also very cost effective as the only hardware it requires is a raspberry PI board and a camera.

LITERATURE SURVEY:

1. "A Deep Learning Approach for Automatic Pothole Detection and Classification using Smartphone Camera Images" (2018)

   Link: https://ieeexplore.ieee.org/document/8682248

   This paper uses a deep learning-based approach for pothole detection and classification using smartphone camera images. The authors used a dataset of images of potholes and normal road images using a smartphone camera and trained a Convolutional Neural Network (CNN) model to classify the images. They achieved an accuracy of over 90% in detecting potholes using this approach. The paper also discusses the potential of using this system for crowd-sourcing pothole data collection.

2. "Pothole Detection and Segmentation using Deep Learning Techniques" (2020)

   Link: https://ieeexplore.ieee.org/document/9319019

   This research paper proposes a deep learning-based approach for pothole detection and segmentation using road images. The authors use a dataset of road images with and without potholes, and trained a Mask R-CNN (Region-based Convolutional Neural Network) model to detect and classify the potholes in the images. They achieved an accuracy of over 92% in pothole detection and segmentation using this approach. The paper also discusses the potential of using this system for autonomous vehicle navigation and intelligent transportation systems.

3. "Real-time Pothole Detection and Alert System using Computer Vision" (2021)

Link: https://www.sciencedirect.com/science/article/pii/S1364815221002406

This research paper talks about a real-time pothole detection and alert system using computer vision. The authors used a camera mounted on a moving vehicle to capture road images and detect potholes in real-time using image processing techniques. The system generates alerts for drivers through a smartphone application when a pothole is detected on the road. The paper also presents experimental results and evaluates the system's performance in terms of accuracy and real-time response.

4. "Smartphone-based Pothole Detection using Accelerometer Sensor" (2019)

   Link: https://ieeexplore.ieee.org/document/8748281

   This research paper proposes a smartphone-based pothole detection system using accelerometer sensor. The authors developed an Android application that uses the smartphone's built-in accelerometer sensor to detect potholes based on changes in the vehicle's vertical acceleration. The system generates alerts for drivers through the smartphone application when a pothole is detected on the road. The paper also presents experimental results and evaluates the system's performance in terms of accuracy and real-world applicability.

5. "Pothole Detection and Classification using Machine Learning Techniques" (2021)

   Link: https://ieeexplore.ieee.org/document/9470728

   This research paper proposes a machine learning-based approach for pothole detection and classification using road images. The authors used a dataset of road images with and without potholes, and trained a Support Vector Machine (SVM) classifier to classify the images as potholes or non-potholes. The system generates alerts for drivers through a smartphone application when a pothole is detected on the road. The paper also shows experimental results and evaluates the system's accuracy and computational efficiency.

6. "Pothole Detection using Image Processing Techniques" by Wang et al. (2018)

   Link: https://www.sciencedirect.com/science/article/pii/S2405452620301263

   This research paper proposes a pothole detection system using convolutional neural networks (CNNs) applied to road images. The authors used a dataset of road images with and without potholes and trained a CNN model to detect potholes in the images. They achieved an accuracy of over 96% in pothole detection using this approach. The paper also presents the system's accuracy and computational efficiency. Additionally, the authors propose a new metric for evaluating the performance of pothole detection systems based on the severity and location of detected potholes.

7. "Road Damage Detection and Classification using Deep Convolutional Neural Networks" (2019)

   Link: https://ieeexplore.ieee.org/document/8780249

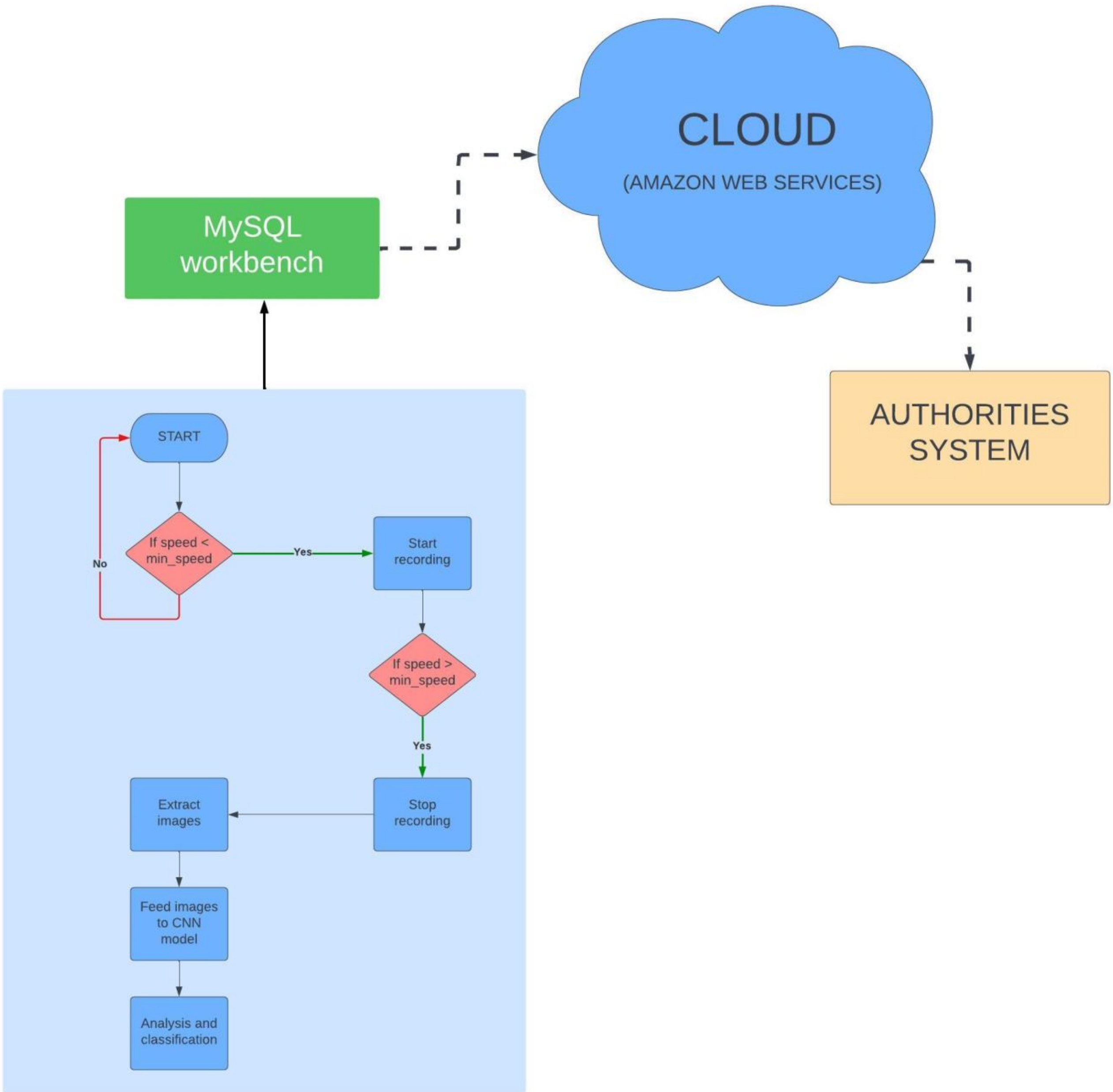
   This research paper proposes a road damage detection and classification system using deep convolutional neural networks (DCNNs). The authors collected a dataset of road images with various types of damages, including potholes, cracks, and patches. They trained a DCNN model to detect and classify these damages in real-time. The system generates alerts for drivers through a smartphone application when a road damage is detected on the road. The paper also presents experimental results and evaluates the system's performance in terms of accuracy and real-world applicability.

PROPOSED SYSTEM:

The system we are proposing to build will comprise of a video camera mounted on the front bumper of the car connected to a raspberry PI board. The board will have the CNN (convolutional neural network) model which has already been trained with a huge dataset of images of roads with and without potholes already uploaded to it. The board will also be connected to the speedometer of the vehicle. When the speed of the vehicle decreases below a threshold, the camera starts recording the road in front of the vehicle. The recording stops when the vehicle has crossed the pothole and the speed increases again. The video is then sent to the raspberry PI board which converts it into a dataset of images. The images are then analysed by the CNN model and checks if a pothole has been detected. If a pothole is detected, it adds a record in the CSV file (stored in the R-Pi board itself) containing the location of the pothole, the speed of the vehicle while passing through it, the time spent to cross it and the severity of the pothole produced after analysis of the video is done. At the end of the drive or when the driver wants, the raspberry PI is connected to wi-fi and it uploads the CSV file to the AWS database using MySQL workbench.

The authorities can later access the database, download the CSV file and using filters, they can check the potholes have high severity and their locations too and fix them accordingly.

Architecture Diagram:



IMPLEMENTATION DETAILS:

Modules involved:

Data Collection:

The data for the project is collected from the camera connected to the raspberry PI board. The board is connected to the speedometer of the car and when the meter reads a value below a given speed, the board turns on the camera and starts recording. It continues recording till the speed of the vehicle is above the given speed again.

Data pre-processing and analysis:

The video is then converted into a dataset of images. The dataset of images is analysed using the CNN model that has already been trained. The model checks if each image has a pothole in it and classifies it. The severity of a pothole is determined by the time spent to get around the pothole and the speed at which the vehicle was moving while going around it.

Data Storage:

After the analysis is done, the data is stored in a CSV file that is stored locally on the board. When it is connected to a wi-fi, it uploads the CSV file to AWS cloud (RDS) using MySQL workbench.

CODE:

Import necessary libraries:

```
import tensorflow as tf
from tensorflow import keras
from keras import preprocessing as keras_preprocessing
from tensorflow.keras.preprocessing import image
from keras_preprocessing.image import ImageDataGenerator
from tensorflow.keras.optimizers import Adam
```

Import training dataset to train the CNN model:

```
pothole_training = "C:/Users/alans/OneDrive/Desktop/Sem6/ECE3502/J
component/kaggle_dataset"
training_datagen = ImageDataGenerator(rescale = 1./255, horizontal_flip=True,
rotation_range=30, height_shift_range=0.2, fill_mode='nearest')

pothole_validation = "C:/Users/alans/OneDrive/Desktop/Sem6/ECE3502/J
component/kaggle_dataset"
validation_datagen = ImageDataGenerator(rescale = 1./255)
```

```
pothole_train_generator =
training_datagen.flow_from_directory(pothole_training,target_size=(224,224),cl
ass_mode='categorical',batch_size = 64)

pothole_validation_generator =
validation_datagen.flow_from_directory(pothole_validation,
target_size=(224,224), class_mode='categorical', batch_size= 16)
```

Training the model:

```
model_pothole = tf.keras.models.Sequential([
tf.keras.layers.Conv2D(96, (11,11), strides=(4,4), activation='relu',
input_shape=(224, 224, 3)), tf.keras.layers.MaxPooling2D(pool_size = (3,3),
strides=(2,2)),
tf.keras.layers.Conv2D(256, (5,5), activation='relu'),
tf.keras.layers.MaxPooling2D(pool_size = (3,3), strides=(2,2)),
tf.keras.layers.Conv2D(384, (5,5), activation='relu'),
tf.keras.layers.MaxPooling2D(pool_size = (3,3), strides=(2,2)),
tf.keras.layers.Flatten(),
tf.keras.layers.Dropout(0.2),
tf.keras.layers.Dense(2048, activation='relu'),
tf.keras.layers.Dropout(0.25),
tf.keras.layers.Dense(1024, activation='relu'),
tf.keras.layers.Dropout(0.2),
tf.keras.layers.Dense(2, activation='softmax')])

model_pothole.compile(loss='categorical_crossentropy',optimizer=Adam(learning_
rate=0.0001),metrics=['acc'])

history = model_pothole.fit(pothole_train_generator,steps_per_epoch =
11,epochs = 40,validation_data = pothole_validation_generator,validation_steps
= 15)
```

Save the model:

```
model_pothole.save("C:/Users/alans/OneDrive/Desktop/Sem6/ECE3502/J
component/model_pothole.h5")  # Save the model to a file
```

Load the model:

```
from tensorflow.keras.models import load_model

model_pothole = load_model("C:/Users/alans/OneDrive/Desktop/Sem6/ECE3502/J
component/model_pothole1.h5")
```

Convert video to dataset of images:

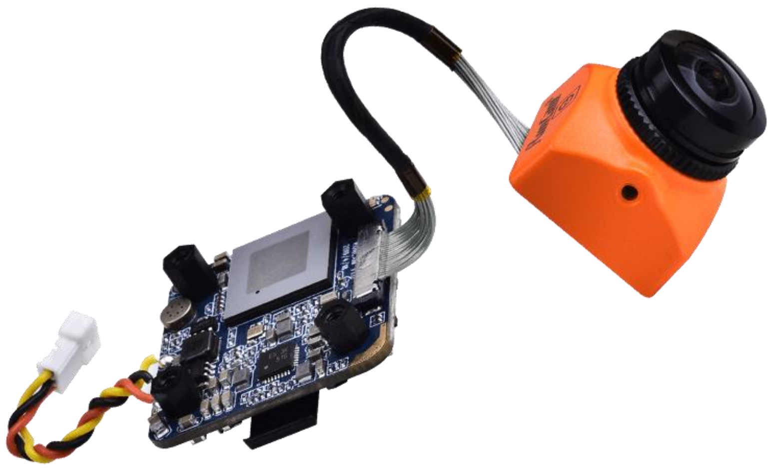
```python
import cv2

video_path = "C:/Users/alans/Downloads/DASH CAM 2016 01 29 (42 Miles of
Potholes).mp4"
output_folder = "C:/Users/alans/OneDrive/Desktop/Sem6/ECE3502/J
component/video_output"
frame_rate = 30 # the number of frames to extract per second

cap = cv2.VideoCapture(video_path)
count = 0

while cap.isOpened():
    ret, frame = cap.read()

    if not ret:
        break

    if count % frame_rate == 0:
        file_name = output_folder + "/frame{:d}.jpg".format(count)
        cv2.imwrite(file_name, frame)

    count += 1

cap.release()
```

Analysing the dataset of images:

```python
import numpy as np
import os
from matplotlib import pyplot as plt
path = "C:/Users/alans/OneDrive/Desktop/Sem6/ECE3502/J component/dataset"
for i in os.listdir(path):
    img = image.load_img(path + '/' + i, target_size = (224,224))
    plt.imshow(img)
    plt.show()
    x = image.img_to_array(img)
    x = np.expand_dims(x, axis=0) / 255
    classes = model_pothole.predict(x)
    pothole_flag = detect("Pothole/s",np.argmax(classes[0]) == 0 )
    print(pothole_flag)
```

HARDWARE DETAILS:

Runcam Split 3 Nano:



The Runcam Split 3 Nano is a small, lightweight camera that is used for the system. It is designed to be mounted on the front bumper of a car and captures high-resolution images and videos of the road surface. The camera features a 1/3" CMOS sensor with a resolution of 1080p at 60fps, which allows it to capture detailed images of the road surface. It also has a wide dynamic range, which means that it can capture images with a high level of contrast, making it suitable for use in different lighting conditions. It can be controlled and configured using software provided by the manufacturer, and supports real-time video output, which is useful for monitoring the road surface in real time.

Raspberry PI 4:

For the proposed system, the Raspberry Pi can be used as the main processing unit to run the computer vision algorithms and analyse the images captured by the camera. It has a variety of input/output interfaces, including USB, Ethernet, HDMI, and GPIO pins, which makes it compatible with a wide range of peripherals and sensors. The Raspberry Pi board is available in different models, each with its own specifications. For example, the Raspberry Pi 4 Model B has a quad-core ARM Cortex-A72 CPU, up to 8GB of RAM, and supports dual-band Wi-Fi and Bluetooth connectivity. This makes it a powerful and reliable choice for pothole detection applications.

SOFTWARE COMPONENTS:

Amazon RDS (AWS):

Amazon Web Services (AWS) is a cloud computing platform that provides a wide range of services for building, deploying, and managing applications and infrastructure in the cloud. One of these services is Amazon Relational Database Service (RDS), which is a managed database service that makes it easy to set up, operate, and scale a relational database in the cloud. With Amazon RDS, users can choose from several popular database engines, including MySQL, PostgreSQL, Oracle, and Microsoft SQL Server. The service provides automatic backups, automated software patching, and scalable storage, which means that users can easily scale their databases as their needs change.

Amazon RDS also provides high availability and fault tolerance, which ensures that databases are available and running smoothly even in the event of hardware failure or other issues. Users can easily create read replicas of their databases to offload read traffic and improve performance, and they can also use Amazon RDS to replicate their databases across multiple availability zones for added redundancy.

MySQL Workbench:

MySQL Workbench is a software tool designed to manage and administer MySQL databases. It provides a graphical user interface (GUI) that simplifies tasks such as creating and modifying database schemas, running queries, and analysing database performance. The software is used by developers and database administrators to streamline database management tasks, as well as optimize database performance.

One of the key features of MySQL Workbench is its database design tool. This tool allows users to create and modify database schemas visually, rather than writing SQL code. Users can create tables, define relationships between tables, and add constraints and indexes with ease. The tool also supports reverse engineering, allowing users to import an existing database schema and generate a visual representation of the database.

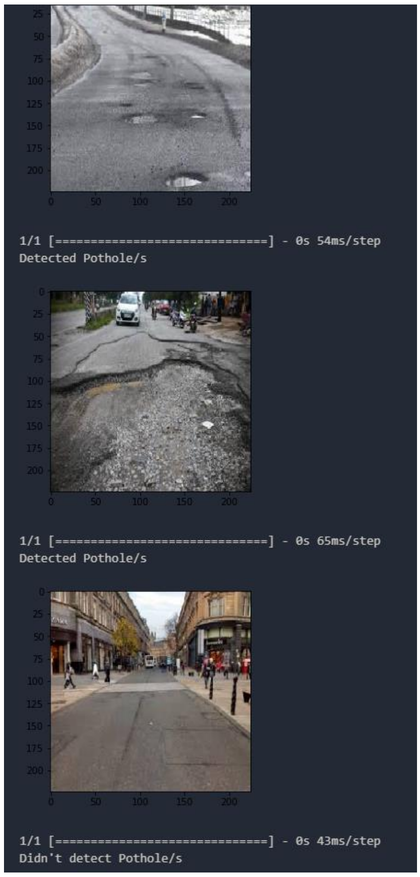
MySQL Workbench also includes a range of tools for database administration and management. These tools allow users to perform tasks such as user management, backups and restores, and server configuration. For example, the user management tool allows administrators to add and remove users, assign permissions, and manage user roles. The backup and restore tool allows users to create and restore backups of their databases, ensuring that data is protected in the event of a system failure or data loss.

WORKING ENVIRONMENT:

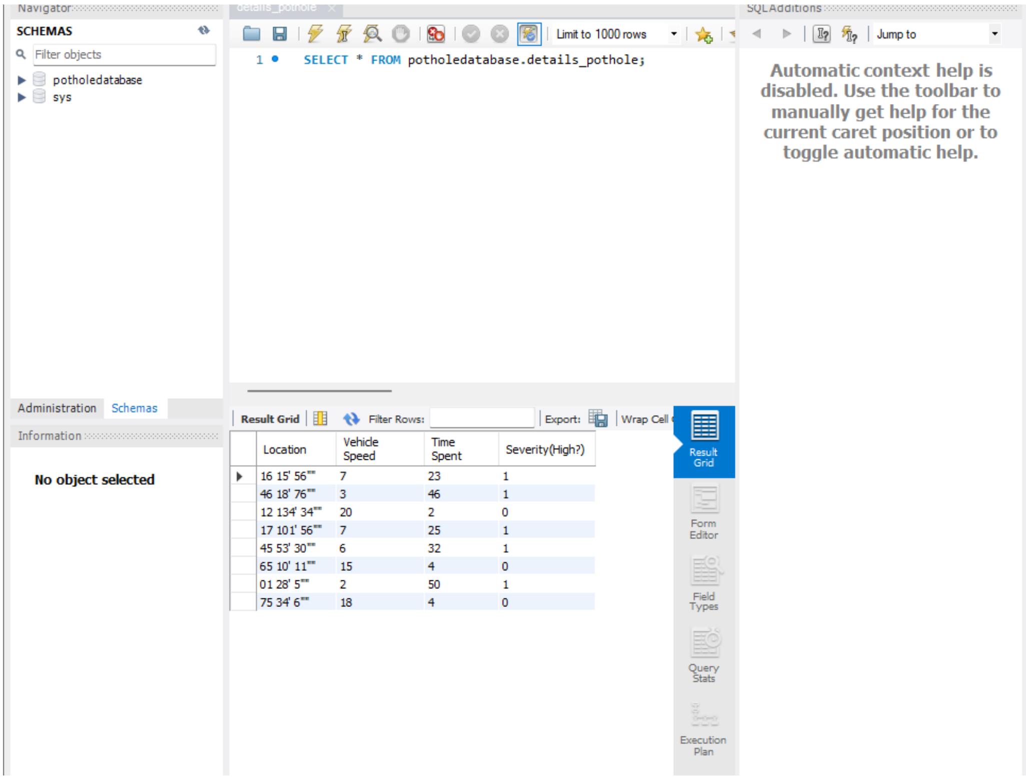
The working environment for the project is a road filled with potholes and also containing some smooth patches too, to ensure the proper working of the system.

RESULTS:

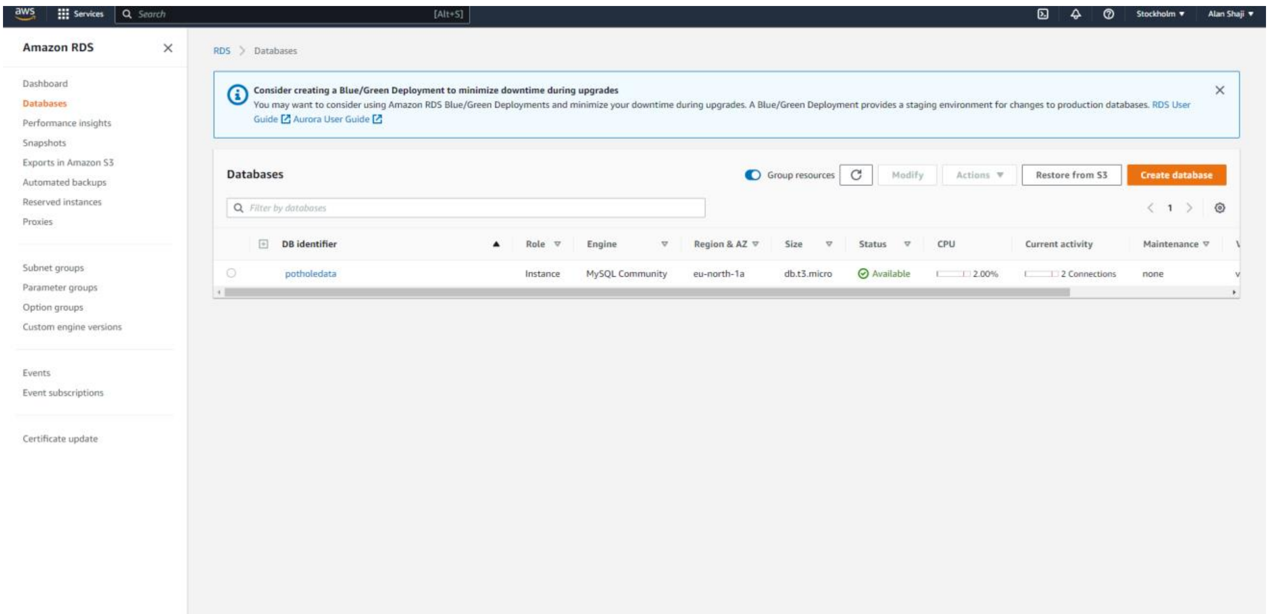
We have used a set of images randomly taken from the internet but not in the training dataset to check how accurate the model is:



The analysed data along with the location of the vehicle and speed and time spent to cross the pothole is saved as a CSV file. This file is uploaded to MySQL workbench:



When the data is uploaded to MySQL workbench, it is connected to Amazon RDS and it is simultaneously uploaded to the cloud too:



REFERENCES:

1. M. R. Ahmad, S. A. Malik, M. S. J. Hashmi, and M. T. Naseer, “An Automated Pothole Detection System using Machine Learning,” 2020 IEEE 4th International Conference on Computer and Communications (ICCC), Chengdu, China, 2020, pp. 210-213.

2. C. Gupta and R. K. Agrawal, “Real-Time Pothole Detection and Alert System for Smart Cities,” 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA), Dubai, United Arab Emirates, 2020, pp. 1-6.

3. V. K. Dey, R. P. Singh, and A. K. Singh, “Deep Learning-Based Pothole Detection and Classification using Vehicle-Mounted Camera,” 2020 International Conference on Emerging Trends in Smart Technologies (ICETST), Delhi, India, 2020, pp. 1-5.

4. M. Mishra, S. Sarangi, and S. Pattanaik, “Pothole Detection and Localization using Faster R-CNN,” 2020 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2020, pp. 1-6.

5. S. P. Abeyrathne, P. W. P. Jayasekara, and N. Kodikara, “Smart Pothole Detection and Localization System for Sri Lankan Roads,” 2019 Moratuwa Engineering Research Conference (MERCon), Colombo, Sri Lanka, 2019, pp. 67-72.

6. A. Mustafa and R. F. Ahmed, "Pothole Detection on Roadways using Deep Learning," 2019 IEEE 2nd International Conference on Information and Computer Technologies (ICICT), 2019, pp. 128-132.

7. M. Kaur, G. Kaur, and P. Gupta, "Real-Time Pothole Detection and Localization using Deep Learning," 2020 5th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2020, pp. 1-6.

8. H. A. Abdulsada, T. A. Muhammad, and N. A. Mohamad, "Pothole Detection and Classification using Convolutional Neural Network," 2020 IEEE 7th International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA), 2020, pp. 1-6.

9. Y. Fang, M. Zhang, and S. Liu, "A Deep Learning-based Pothole Detection System for Unmanned Ground Vehicles," 2019 International Conference on Robotics and Automation Sciences (ICRAS), 2019, pp. 536-540.

10. Y. B. Alqawqaz, S. S. Al-Maqtari, and A. A. Al-Ghobari, "Pothole Detection on Roadways using Convolutional Neural Network," 2018 International Conference on Computational Science and Computational Intelligence (CSCI), 2018, pp. 154-157.

11. M. R. Zaman, A. M. Mahmud, and S. S. Saha, "Pothole Detection and Analysis using Computer Vision Techniques," 2018 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE), 2018, pp. 61-64.

12. P. N. Patel, N. N. Patel, and N. M. Patel, "Real-Time Pothole Detection and Monitoring System Using Image Processing Techniques," 2018 IEEE International Conference on Innovative Research and Development (ICIRD), 2018, pp. 1-6.

13. S. S. Barde and S. G. Rahatekar, "Pothole Detection and Recognition using Image Processing," 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), 2017, pp. 1-5.