B. TECH PROJECT REPORT

ON

PAVEMENT AND ROAD HEALTH MONITORING USING RANDOM FOREST TECHNIQUE

BY

SIMMA SAI RAM
190004038

DISCIPLINE OF CIVIL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY INDORE

Submitted on

November 28, 2022

PAVEMENT AND ROAD HEALTH MONITORING USING RANDOM FOREST TECHNIQUE

A PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of the degrees

of

BACHELOR OF TECHNOLOGY

In

CIVIL ENGINEERING

Submitted by:

Simma Sai Ram

Guided by:

Dr. Guru Prakash

(Assistant Professor, Civil Engineering)

CANDIDATE’S DECLARATION

I hereby declare that the project titled “PAVEMENT AND ROAD HEALTH MONITORING USING RANDOM FOREST TECHNIQUE”, submitted in partial fulfillment for the award of the degree of Bachelor of Technology in ‘Civil Engineering’ completed under the supervision of Dr. Guru Prakash, Assistant Professor, Civil Engineering, IIT Indore is an authentic work.

Further, I/we declare that I/we have not submitted this work for the award of any other degree elsewhere.

Sai Ram 28/11/2022

Signature and name of the student(s) with date

CERTIFICATE by BTP Guide(s)

It is certified that the above statement made by the students is correct to the best of my/our knowledge.

Guru Prakash 28/11/2022

Signature of BTP Guide(s) with dates and their designation

(Assistant Professor)

Preface

This report on "PAVEMENT AND ROAD HEALTH MONITORING USING RANDOM FOREST TECHNIQUE" is prepared under the guidance of Dr. Guru Prakash.

In this project, First I collected numerical simulated data from pothole lab and extracted 12 features from its z-axis accelerometer readings using sliding window technique. Then build different machine learning models like decision tree, random forest, support vector machines, K-nearest neighbours and artificial neural networks. I validated all the models with another dataset from same pothole lab. I compared different scores, predictions made by each model in order to select the best model to detect anomalies on field dataset.

I used Arduino UNO micro-controller, ADXL345 accelerometer and PuTTY software to record the road surface data from local roads of Indore. I pre-proceed the data and extracted same 12 features from it and fed into the prepared model to make predictions. I got precise position, length and number of road anomalies which was briefly explained in the results section of field study.

I have tried to the best of my abilities and knowledge to explain the content in a lucid manner using tables, source codes, flow charts etc.

Simma Sai Ram

190004038

B. Tech 4th Year

Discipline of Civil Engineering

IIT Indore

Abstract

Detection of road anomalies is crucial in order to prevent road accidents. Even with the advancement in technology, road accidents are still happening in our country. This is mainly because of the difficulty in detecting road anomalies, which involves high inspection and monitoring costs. Traditionally, electro-magnetic methods like RADAR, LASAR, GPR and visual inspections are used for road health monitoring which is costly, cumbersome and often not reliable. Moreover, the visual inspection of a road requires a lot of time and labour work, electro-magnetic inspections require high skilled labours and expensive equipment, hence it is not possible to implement it on a large scale for all road-network. To overcome the difficulty in detecting road anomalies, recently vibration monitoring devices like accelerometers, gyroscopes, and motion sensors are used in this field due to its low-cost, ease of use, can monitor irrespective of surrounding and seasonal conditions, time saving. machine learning (ML) techniques have been used in this study to detect anomalies from accelerometer readings.

In this paper, the training dataset is collected from pothole lab, an algorithm is developed to extract important features required to detect road anomalies using z-axis acceleration data. Then build some machine learning models from extracted features. The proposed paper uses the random forest (RF) technique for the detection and classification of anomalies. Random forest has been compared with other models, such as decision tree (DT), K- nearest neighbours (KNN) and support vector machine (SVM). The random forest (RF) classifier showed better performance with an accuracy of 90.6% and predicted the precise position of road defect along the length of road.

Keywords: Features, Pothole detection, Random Forest (RF), Road health monitoring, Road anomaly detection.

CONTENTS

list of Figures vii
List of tables viii
chapter 1 1
Introduction 1
1.1 Motivation 1
1.2 Literature review 2
1.3 Objective 3
Chapter 2 4
Methodology 4
2.1 Random Forest classifier 5
2.2 validation scores 6
Chapter 3 8
Numerical simulation study 8
3.1 Data collection 8
3.2 Feature Extraction 9
3.3 Model Building 12
3.3.1 Training phase 12
3.3.2 Testing phase 14
3.4 Evaluation 16
Chapter 4 18
Experimental Study 18
4.1 Vehicular model 18
4.2 Road Network 19
4.3 Data acquisition system 21
4.3.1 ARDUINO 21
4.3.2 ADXL345 22
4.3.3 PuTTY 24
4.3.4 connections 25
chapter 5 28
5.1 Results and discussion 28
5.2 conclusion 32
References 33

LIST OF FIGURES

Fig1. 1 From headlines on NDTV news official website. 1

Fig 2. 1 Methodology 4
Fig 2. 2 Random Forest technique Process 5
Fig 2. 3 Confusion matrix 6

Fig 3. 1 Pothole lab portal 8
Fig3. 2 Numerical simulated data of Training and Testing. 9
Fig 3. 3 Feature extraction process. 10
Fig 3. 4 Sliding window technique for feature extraction 10
Fig 3. 5 Phases of model building 12
Fig3. 6 Training Phase to build model 13
Fig3. 7 Testing Phase 14
Fig 3. 8 Validation scores of different models 15
Fig 3. 9 Number of potholes predicted by different models. 16
Fig3. 10 Accuracy score versus Number of decision trees 17
Fig3. 11 Accuracy score versus random state 17

Fig 4. 1 Data acquisition system on car dashboard 19
Fig4. 2 Routes for field study 19
Fig4. 3 Serial plot for different type of surface defects. 20
Fig 4. 4 Arduino micro-controller with labelling 22
Fig 4. 5 ADXL345 Tri-axial accelerometer 23
Fig 4. 6 Data acquisition system setup 25
Fig 4. 7 Accelerometer data from Tejaji Nagar to IIT gate no.1 26
Fig 4. 8 Accelerometer data from IIT gate no.2 to Gavhalu 27
Fig4.9 Feature weights/importance 27

Fig 5. 1 Predicted defects along the road from Tejaji Nagar to IIT. 28
Fig 5. 2 Predicted damages versus original damages 29
Fig 5. 3 Images numbers of some of predicted damages on tejaji to IIT road 30
Fig 5. 4 Referenced images to corresponding numbered stripes 31

LIST OF TABLES

Table 1 Features extracted from accelerometer data 11
Table 2 Classifiers and parameters used to build models. 13
Table 3 Car model design and suspension system of vehicle selected for study 18
Table 4 Description of pins in Arduino micro-controller 22
Table 5 Specifications of ADXL345 accelerometer... 23
Table 6 Description of ADXL345 accelerometer pins 24
Table 7 Pin to pin connections of Arduino board to ADXL345 25

CHAPTER 1

INTRODUCTION

1.1 MOTIVATION

Inspection of roads is necessary for better performance, and to reduce the growth of road anomalies such as potholes, rutting and cracks. These types of anomalies are possible due to usage of low quality of materials during construction, unusual vehicular loads, and due to climatic changes. Visual inspection for such a huge stretch of road is time consuming process and requires significant man power. Hence, we need an approach that is necessary to detect the road anomalies at low cost with relatively semi-skilled labour and time saving. Otherwise, these anomalies will grow and damage the entire road, which requires replacement. To do so the traffic need to be enrouted, it may lead to traffic congestion or blocking. Hence regular maintenance and monitoring of roads is crucial or else it may be problematic to the road users, heavy damage to vehicle suspension system and faster wear and tear of tyres, accidents may also possible. According to transport ministry of India, nearly 5,000 people were killed in road accidents caused by potholes in 2018 to 2020. The Fig 1.1 is the headline of official NDTV web page dated on august 22, 2022. So, recently different equipment based on vibration monitor come into picture in this field. This study focuses on monitoring of roads by detecting the road anomalies such as potholes, speed bumps, metal bumps using acceleration data. This helps the government and other agencies to hasten the process of road maintenance repair works. Most of the researches suggested road anomaly detection by collecting data either in the form of accelerometer or image data of road, later processing of data and feeding to machine learning (ML), deep learning (DL) algorithms.

Fig1. 1 From headlines on NDTV news official website.

1.2 LITERATURE REVIEW

From past 10 years many researches have been going on in the field of obtaining accurate data of road surface information using vibration monitoring sensors and machine learning. The researchers used different data acquisition systems like MEMS accelerometers, gyroscopes, smartphones, Vibration sensors to obtain road surface information and implemented different machine learning techniques like supervised, unsupervised, reinforcement and neural networks in order to detect road surface defects from obtained data. Accelerometer can give detailed information about road surface than image data. Some of the research techniques, methods and results by researches have been discussed below.

Carlos et al. [2] conducted studies using acceleration data and compared the traditional damage detection methods based on thresholds with the machine learning techniques. The same authors also proposed a novel methodology using SVM and detected the road anomalies. They developed an online platform for the virtual road accelerometer data. Many of the researchers [3–7], performed binary classification using SVM and conducted field studies for the detection of potholes. All of these publishes focused on classification of two classes i.e., pothole, no pothole using both sensor and imagery data. None of these considered multi class classification of road anomalies and repair cost estimation.

Egaji et al. [10] conducted a field study on pothole detection using different machine learning algorithms. They compared all the models and proposed K- Nearest Neighbours (KNN) and Random Forest (RF) has given better accuracy for detection of potholes.

Pandey et al. [11–13] developed an algorithm for the real time monitoring and detection of potholes. They proposed this using Convolutional Neural Networks (CNN) for the imagery data. When compared to the SVM, KNN and CNN, RF tree has been given high accuracy which has been mentioned in [1, 9]. Along with these techniques some of the researchers also proposed classification and detection of road anomalies using logistic regression [8, 14]. For road anomaly detection, sensory and imagery data has been used for the last couple of years. But still there is a perplexity in applying these algorithms for the real monitoring of roads especially using acceleration data

In [12], Thanuka Wickramarathne and Varun Garg report on new findings from experiments utilising accelerometers to detect road abnormalities. The predictive potency of features produced by acceleration sensors is examined, with a focus on the creation of a low-cost but reliable road irregularity detection system. Using a signal transfer model, a simplified method for "reconstructing" road surface conditions from vertical acceleration measurements is described. This method takes into account system dynamics and low-frequency filtering effects related to a vehicle's suspension system. The prediction accuracy of characteristics produced from acceleration sensors is also examined using a statistical method known as the denoising algorithm with this particular type of signal transmission in place. Through the use of actual data, the signal model and feature analysis method were presented.

The road surface conditions in Sri Lanka were monitored by Kasun et al. in [3] using an acceleration sensor board for their ongoing project (BusNet), which was primarily created for environmental pollution monitoring. By varying the vertical acceleration, acceleration sensor boards may detect the presence of potholes. They can also use the horizontal acceleration to detect changes in the vehicle's speed. Inconsistency in potholes is the biggest drawback of BusNet's road surface monitoring since a change in the horizontal components of acceleration does not always signify a rough section of road; it could be indicating a traffic gridlock. Most of the above-mentioned studies dealt with detection of anomalies with less accuracy and customized data. None of the researchers discussed the position of defect on the road. But in this paper my model detects the defects of road more accurate with precise location on the road.

1.3 OBJECTIVE

The main contributions of this study are the following:

i) Development of a novel algorithms with high accuracy for detection of road anomalies on simulated numerical data from pothole lab.
ii) Validation of the performance of each prepared model using testing dataset.
iii) Select best model out of them based on their prediction values and scores.
iv) Collect own data from field using Arduino UNO and ADXL345 accelerometer, recorded using PuTTY.
v) Observe the efficiency of the built model on field dataset

The rest of the paper is organized as follows: theoretical background of the proposed model has been explained in section 2, later on the process of collecting data from the road, followed by pre-processing of data includes feature extraction has been done in section 3. After that data has been fed to the proposed algorithms, accuracy comparison 2 of the proposed ones with other models and then testing of the proposed model is presented in section 4.

CHAPTER 2

METHODOLOGY

The project consists of four parts such as data collection and pre-processing, feature extraction, model building, and its evaluation. The Fig 2.1 below represents process involved in this project. Since the first part started with collection of data. We need some data in order to build machine learning models, we collected data from the pothole lab in numerical simulation study and from field in field study part. Applied sliding window technique to extracted features and labels. Then the extracted features were fed into different machine learning models. Finally evaluated the performance using testing dataset. The detailed process will be discussed in coming sections.

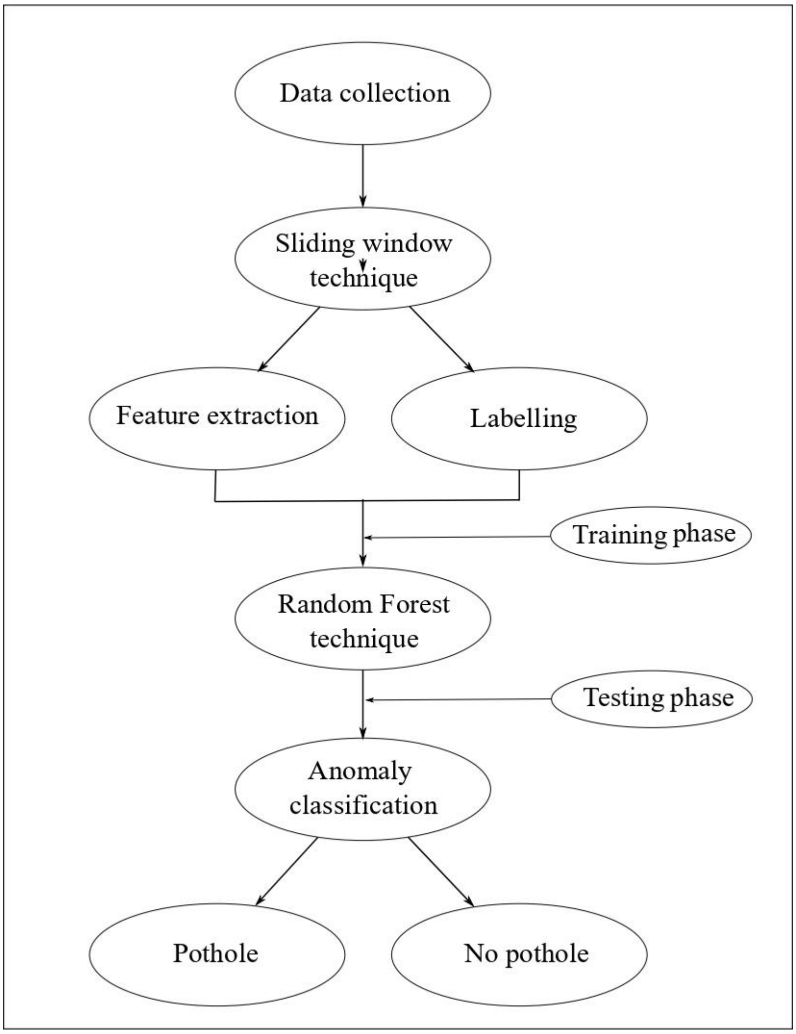


Fig 2. 1 Methodology

2.1 RANDOM FOREST CLASSIFIER

To train the model the raw acceleration data obtained from the road need to be pre-processed. Since random forest (RF) is a supervised learning, the data needs to be labelled.Hence sliding window and feature extraction has been done in the early stages. Now the data (R) is ready to train, each window of 30 observations and their corresponding labels are used. Labels are considered based on the anomaly time stamps given by pothole lab.It is not possible for a window to cover an event (pothole or no pothole) because the event length varies. Hence the left-over data of an event (pothole) not fulfilling window are also labelled as pothole. The data is divided into windows along with values of features for each window and their corresponding labels. Now, this data divides into n number of subsets with some random number of windows containing 12 features. It's time to choose number of decision trees, it helps to increase the prediction accuracy. Each subset R with random number of windows is assigned to each decision tree and the model learns whether this window is either pothole ('1') or no pothole ('0') based on features. Finally, the output will be based on majority of votes.

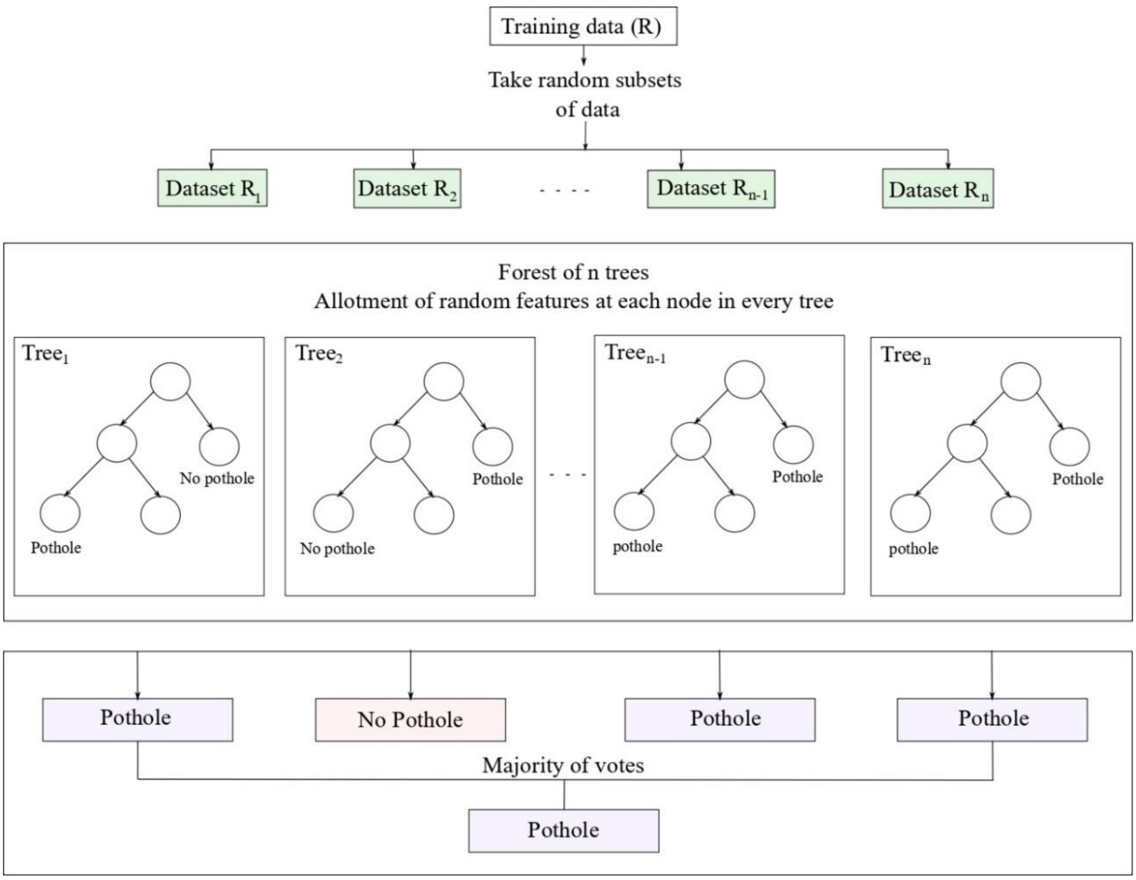


Fig 2. 2 Random Forest technique Process

2.2 VALIDATION SCORES

The performance of the machine learning model is validated using the validation scores of predicted values on testing dataset. Validation scores like Accuracy, precision, f1-score and recall can be used to validate the model.

There is positive (pothole or "1") and negative (smooth or "0") classes in this binary classification. Since there are two classes, we must first identify True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN), but this is simple. Finding TP, TN, FP, and FN for each unique class is what we need to do in this situation.

Actual classes / Predicted classes	Pothole (1)	No damage (0)
Pothole (1)	True positive	True negative
No damage (0)	False positive	True negative

Fig 2. 3 Confusion matrix

True Positive (TP): This term relates to how many predictions the classifier made that were accurate in predicting the positive class as positive**.**

True Negative (TN): The number of predictions in which the classifier properly identified the negative class as negative is known as True Negative (TN).

False Positive (FP): The number of predictions when the classifier mistakenly predicted the negative class as positive is referred to as this.

False Negative (FN): The number of predictions where the classifier mistakenly predicted the positive class as negative is what is meant by this.

Accuracy score: It provides you with the model's overall accuracy, or the percentage of all samples that the classifier properly identified.

Accuracy score = $\frac{(\mathrm{TP}+\mathrm{TN})}{(\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN})}$

Recall: It reveals what percentage of all positive samples the classifier properly identified as positive. True Positive Rate (TPR), Sensitivity, and Probability of Detection are additional names for it.

Recall = $\frac{\mathrm{TP}}{(\mathrm{TP}+\mathrm{FN}).}$

Precision: It reveals the percentage of forecasts in the positive class that were true positives.

Precision = $\frac{\mathrm{TP}}{(\mathrm{TP}+\mathrm{FP})}$

F1-score: It combines recall and precision into one measurement. It is the harmonic mean of recall and precision in mathematics.

F1-score = $2 \times \frac{\mathrm{Recall} \times \mathrm{Precision}}{\mathrm{Recall}+\mathrm{Precision}} = \frac{\mathbf{2}\mathrm{TP}}{2\mathrm{TP}+\mathrm{FP}+\mathrm{FN}}$

Accuracy of the random forest model was approximately 90 percent, but accuracy would fool us when it masks the issue of class imbalance. In our dataset the data contains only 1944 ‘1’ instances out of 15299 instances. So, a majority baseline classifier which always assigns the ‘0’ label would reach 90% accuracy since it would correctly predict 90% instances. Similarly, other validations scores have also same issue with imbalanced instances but can give a brief idea to validate. That is why we give clear indication of scores calculation.

CHAPTER 3

NUMERICAL SIMULATION STUDY

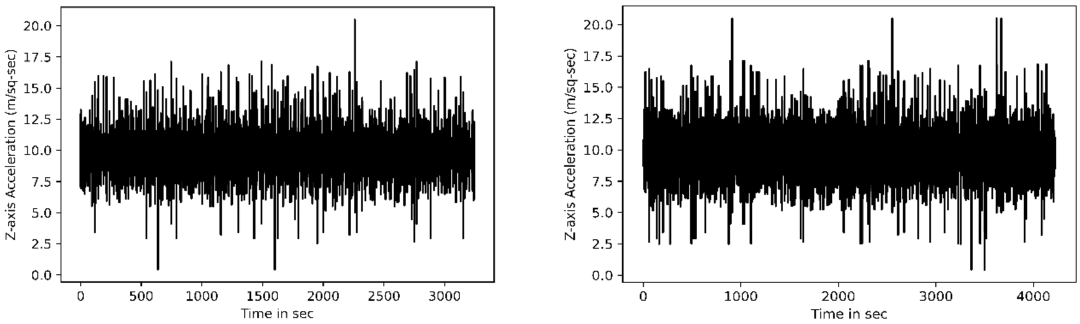
3.1 DATA COLLECTION

Collection of data plays an important role in building machine learning models. We collected data through virtual roads platform of pothole lab which was created by [2] Carlos et al and shown in Fig 3.1. It is a web platform to create virtual roads through accelerometer patterns of road anomalies. Modern smart phone equipped with different sensors like triaxial accelerometer, triaxial gyroscope, magnetometer, GPS, linear accelerometer etc. In this platform, the triaxial accelerometer data has been collected by the smartphone that kept on the dashboard of the car parallel to road surface. For training of model, we took a dataset of 150 potholes and for validation set we took a dataset of 200 potholes from pothole lab. The sampling frequency of 50 Hz was taken and average speed of 35 km/hr is maintained while collecting the data in pothole lab.

Pothole lab provided data in a JSON file which has been used further in this study. The dataset obtained by pothole lab consists of X, Y, Z axes acceleration data, start and end time step of the anomaly indicates that it is already labelled dataset, vehicular model, vehicular average speed etc. Different types of anomalies such as metal bumps, speed bumps and potholes are found on the road. This study focuses only on unnecessary anomalies i.e., potholes, hence other type of anomalies are ignored. The anomaly generally present in z-axis direction, so we consider the acceleration readings in z-direction. The target labels were assigned by using starting and ending event defined in anomaly section of dataset.

Fig 3. 1 Pothole lab portal

The Fig 3.2 shown below represents the serial plot of numerically simulated dataset for training(a) and testing (b) respectively that are collected from pothole lab. In the plot x-axis represents time series in seconds and y-axis represents the acceleration data in z-direction.



a) Training dataset b) Testing dataset

Fig3. 2 Numerical simulated data of Training and Testing.

3.2 FEATURE EXTRACTION

The efficiency of building a machine learning model depends on the features selected to train the model. This project works on z-axis acceleration data which is a numerical time series data. So, we need to find how statistical features varies when it passes through an anomaly. The basic idea behind the project is when vehicle moves on a smooth surface the accelerometer data tends to stays around the original acceleration due to gravity(g) that is g=9.81m/s^2. When vehicle passes through an anomaly the sensor tends to move in z-axis direction that results in change in accelerometer readings. The major part of the project is to detect the accelerometer readings from the total dataset.

We considered 12 features such as to detect an anomaly, out of them 5 statistical features and non-statistical features 4 threshold coefficients for statistical features and 3 coefficients based on the 4 threshold coefficients. There are 5 statistical features which are calculated directly from the observations in each window, 7 non statistical features which are calculated with some threshold limits set by conducting different experiments by [15]. Table 1 specifies the features table. Along with these, some other features like median, peak value, skewness (for the random distribution), percentiles, quartiles and outliers are also available. But these features don't focus on all observations of the window and most of them focusses on how observations was varied from others, which is not effective for the anomaly detection. These features also effect the performance of the model by increasing dimensionality. The 12 calculated features and their threshold limits were shown in table1. There is no use of feature calculation for entire data set to detect an anomaly. So, we implemented sliding window technique over entire z-axis accelerometer readings. In each window we calculated 12 features and their corresponding labels. We validated different sized window and overlap conditions. Window size of 30 units and 50% over lap (step value of 15 units) gave better results among others. The extracted features were then used in build different models. The Fig 3.3 shown below is the process of extracting features.

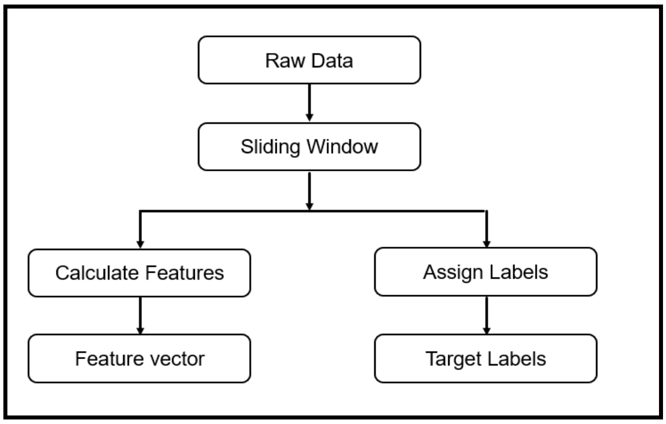


Fig 3. 3 Feature extraction process.

Sliding window

From the Fig 3.4 below we are trying to explain the background working principle of sliding window over z-axis accelerometer readings visually. where each same-coloured box represents that overlap with half of the readings of the succeeding and preceding window and contains 30 accelerometer readings.

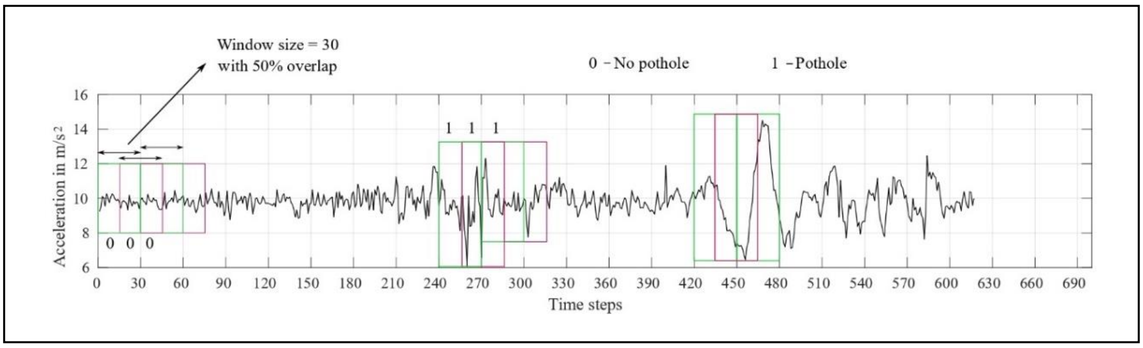


Fig 3. 4 Sliding window technique for feature extraction

List of Features

The 12 calculated features and their threshold limits were shown in table1. along with their feature type and their calculation.

Table 1 Features extracted from accelerometer data

Type	Feature	Calculation
Statistical features	Mean (μ)	$\frac{\sum_{i=0}^{i=30} ri}{30}$.
	Standard deviation(σ)	$\frac{\sum_{i=0}^{i=30} (ri-\mu)}{29}$.
	Variance (θ)	$(\sigma)^2$
	Coefficient of variance (α)	$\frac{\sigma}{\mu}$
	Difference (δ)	Difference of maximum and minimum values in each window.
Non-Statistical features	Capability potential(β)	Threshold value = $g \pm 0.3g$, if observed acceleration is > threshold value, β = 0.6 else 0.3.
	Threshold for std deviation(τ)	If σ > 0.15g then τ = 0.9 else 0.2.
	Threshold for Var(υ)	If θ > 0.15g then υ = 0.8 else 0.2.
	Cont(η)	υ, α, τ and β. If any three of these exceeds their limits, then η = 0.8 else 0.2
	Threshold for COV(χ)	if α > 0.015 then χ = 0.8 else 0.2
	SC(ε)	$\upsilon + \tau + \chi + \beta + \eta$
	Cs(γ)	If ε ≥ 2 then γ = 0.8 else 0.2

3.3 MODEL BUILDING

To build a model in supervised machine learning, it consists of four phases, first phase starts with feature extraction phase that we already discussed it in feature extraction phase before. Then training phase which took a lot of time in which we fed features and corresponding labels to machine learning classifiers. In training phase machine try to learn the hidden patterns to map the input feature vector to output labels. In testing phase, the built model make predictions for testing dataset. Finally, in validation phase the predicted outcomes were compared with original labels. The Fig 3.5 represents the model building procedure.

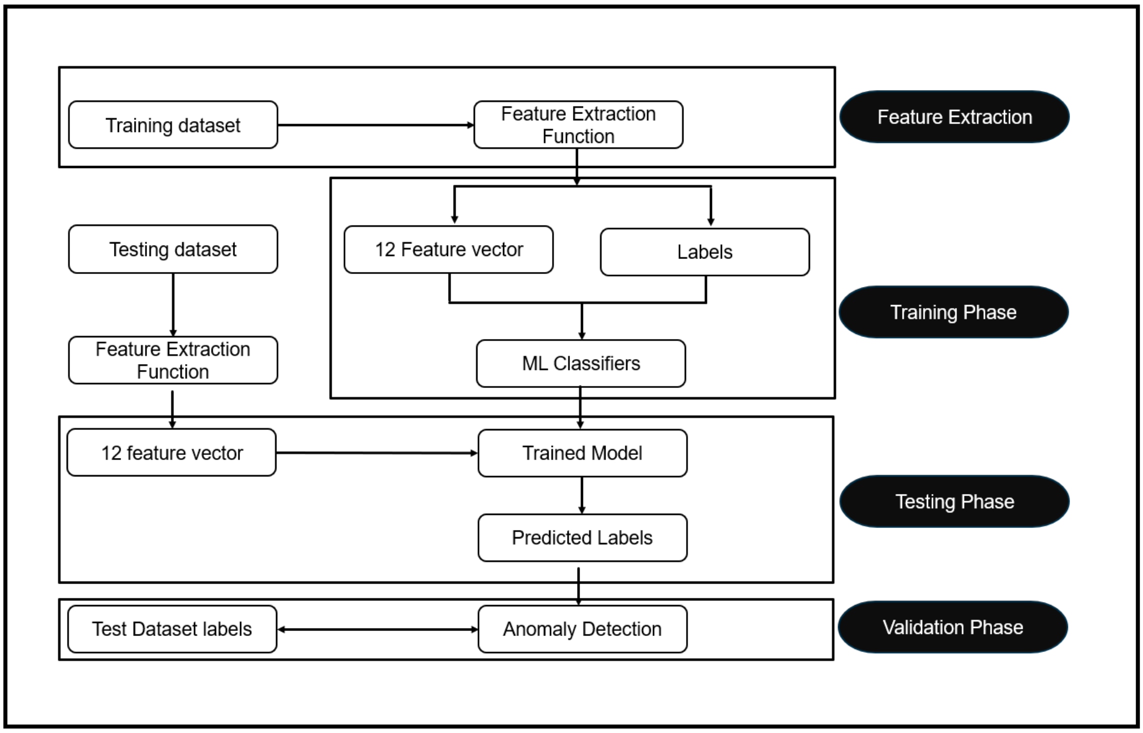


Fig 3. 5 Phases of model building

3.3.1 TRAINING PHASE

We have already discussed the feature extraction phase in above section. The output of this project consists of pothole or healthy segment of road, which depicts a classification type of problem, so supervised machine learning classification techniques were used in order to detect the anomalies. After feature extraction process, different machine learning models are build using feature vectors and corresponding labels. These models are trained and further validated using a separate dataset obtained from the pothole lab.

The data used for training contains 150 potholes. The below Fig3.6 shows the process of training phase. The data has been converted into windows as explained in the above section and corresponding labels are assigned. Total of five machine learning models such as decision tree (DT), random forest (RF), k-nearest neighbors (KNN), support vector machine (SVM) and artificial neural networks (ANN) have been trained.

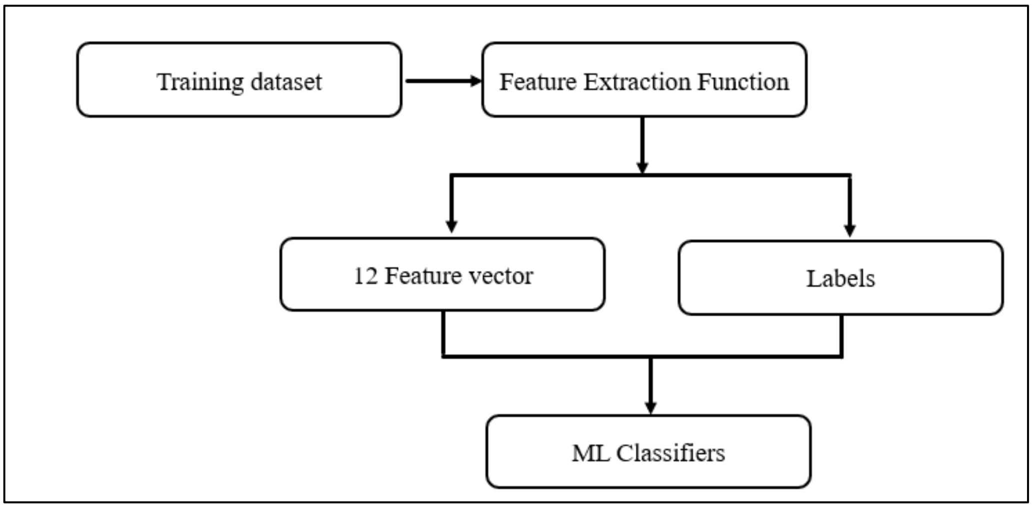


Fig3. 6 Training Phase to build model

The classifiers and parameters used in order to build models are shown in Table 2 below.

Table 2 Classifiers and parameters used to build models.

Model	Parameters
Random Forest	n-estimators = 100, random state = 3, criterion = Gini
Decision Tree	criterion = entropy, random state = 0
Support Vectors	kernel = poly, random state = 0
K-NN	Neighbours = 3
ANN	Layers=4, activation function (hidden=relu, output=sigmoid) Loss = binary cross entropy, optimizer=Adam Gradient = batch gradient

3.3.2 TESTING PHASE

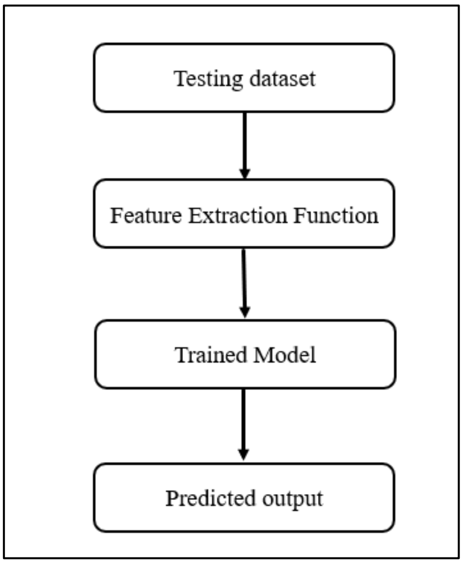


Fig3. 7 Testing Phase

To validate the trained model, it is necessary to test the model to test its performance. Hence for testing a data with 200 potholes has been taken from pothole lab. We have followed the process as shown in Fig 3.7 above to make predictions on testing data. Data has been fed to five ML models, now the data is converted into 12 features vectors containing windows. Since the models have trained based on given 12 features, labels also been assigned to each window of the testing data. During the testing phase, model performances has been found to be effective in detecting and classifying the potholes. Out of these four models RF classifier has shown better accuracy of 90.6%. Other performance details of the models have been presented in Fig 3.8.

The predictions made by the RF model are mostly '0' class, because time interval of occurrence of two consecutive potholes is more compared to time taken to cross a pothole. Since the predicted output consists of large number of '0' class outcomes which will affect the '1' class outcome scores such as recall, precision, and f1-score. These scores mainly depend comparison of true positivity rate with true negative and false positive values in the confusion matrix.

It is observed that approximate time taken by vehicle as to pass through an average size pothole is 2 seconds. The sampling frequency of the data used in this study is 50 Hz. So, an average size pothole has given, 100 acceleration observations.Window size of 30 observations with 50 percent overlap, it is taking five steps to crossan average sized pothole. Each step is giving an outcome as '1', continuous '1' outcomes are considered as a pothole. The length of each pothole is varying based on its size and speed of the vehicle, since the window size is fixed some of the pothole observations may not come into same window.

Accuracy we got in this project was approximately 90 percent, but accuracy would fool us when it masks the issue of class imbalance. In our dataset the data contains only 1944 '1' instances out of 15299 instances. So, a majority baseline classifier which always assigns the '0' label would reach 90% accuracy since it would correctly predict 90% instances. Similarly, other validations scores have also same issue with imbalanced instances but they would give a brief idea on true pothole rates.

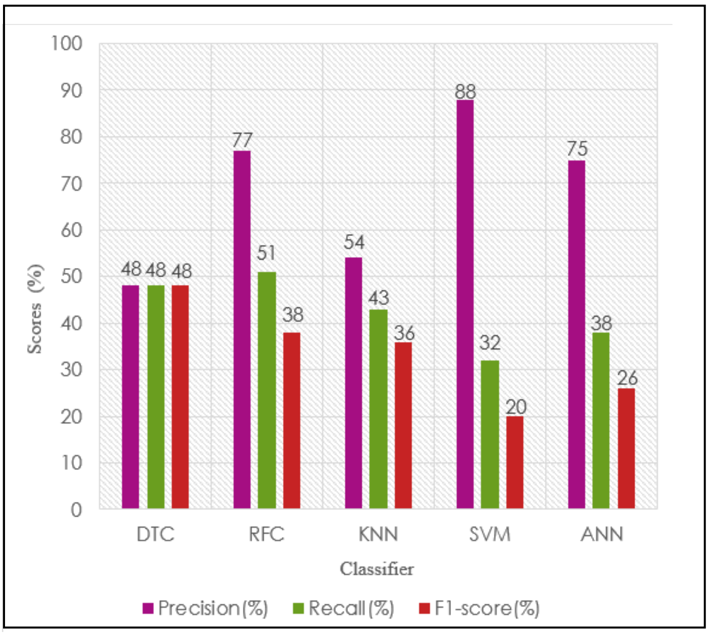


Fig 3. 8 Validation scores of different models

We have tested already labelled testing dataset, we took a test dataset that contains 200 potholes. The models made predictions on the testing dataset z-axis accelerometer data. In Fig 3.9 the pink bar of below bar graph represents the percentage of potholes that each the model predicts out of 200 given pothole data and green bar represents that the percentage of true potholes, that doesn't contain false predicted potholes. Let us discuss about this, in pink bar there may be some there may be some potholes which are not potholes, so there is a possibility of presence of some false potholes prediction. Where as in green bar the potholes that present in both predicted and actual set here, we include only true predicted potholes.

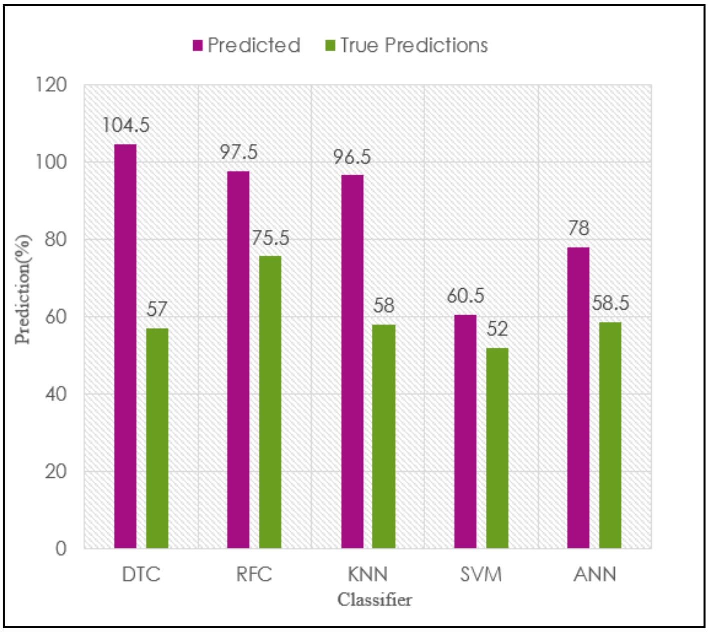


Fig 3. 9 Number of potholes predicted by different models.

3.4 EVALUATION

The RF classifier has predicted the defects with precision of 77%, recall of 51% and f1-score of 38% and it has better overall scores as compared to others. Out of curiosity, comparison of the predicted data with the original test data has been done. By comparing both the predicted and original testing data, it is observed that, these models are predicting some of the 'no pothole' instances as 'potholes' and some of the 'pothole' instances as 'no potholes'. It is clearly observed that random forest technique shown better accuracy in overall prediction that contains false potholes as well as without false potholes.

So, random forest classifier is considered as our best model for anomaly detection. The parameters for random forest model will also effects in detection of anomaly. The split at each node of decision tree in a random forest will be occur based on the randomness of the features. The randomness measuring indices in a decision tree are Gini, entropy and log loss and we can take any one of the three criterions. Gini impurity is taken over entropy and log loss in order to optimize time during training phase for making an optimum split. We can see in below equation 1,2,3 the computation time for log loss and entropy will took longer time than Gini due to complex logarithmic calculation. There is no significant effect of scores when we choose Gini over others.

$$\text{Gini Index} = 1-\sum P_i^2 \quad (1)$$

$$\text{Entropy} = -\sum P_i \log_2 P_i \quad (2)$$

$$\text{Log loss} = \frac{1}{N}\sum_{i=1}^{N} logloss_i$$

$$Logloss_i = -\frac{1}{N}\sum_{i=1}^{N}[\, y_i \ln P_i + (1-y_i)\ln(1-P_i)] \qquad (3)$$

The other parameters like random state and number of decision trees should be present in a random forest has been decided using the variation of accuracy score vs number of estimators and accuracy score vs random state graph. At nestimators of 35 or above the accuracy doesn't have significant increase so we choose estimators of 35 and at random state of 43 got maximum accuracy.

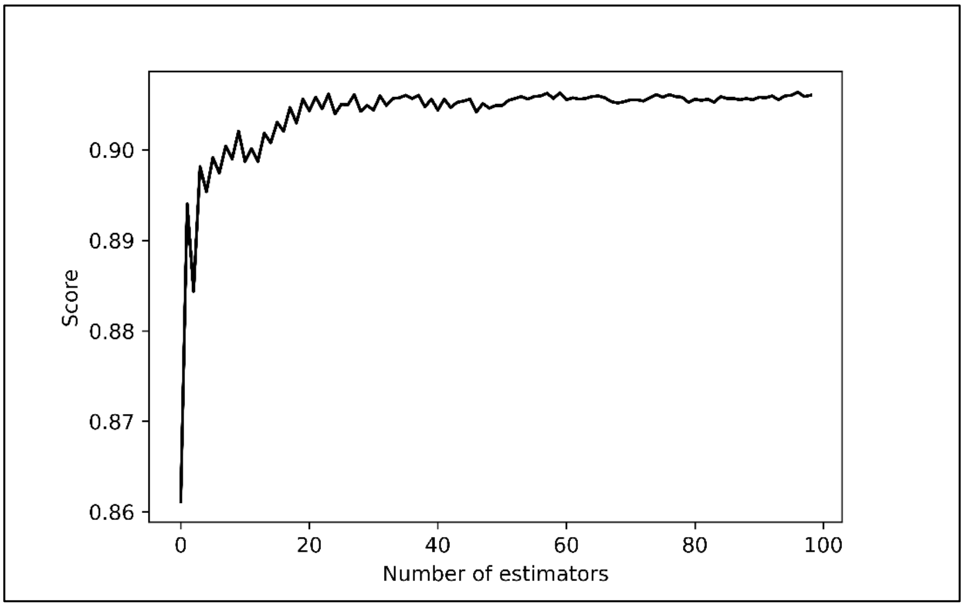


Fig3. 10 Accuracy score versus Number of decision trees

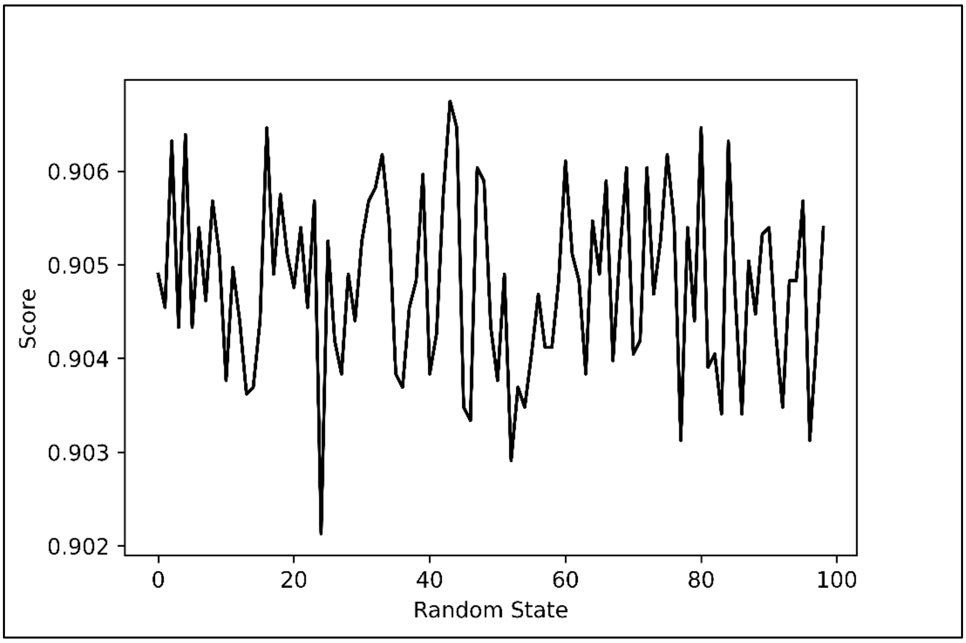


Fig3. 11 Accuracy score versus random state

CHAPTER 4

EXPERIMENTAL STUDY

4.1 VEHICULAR MODEL

For the field study we choose the car model of Maruti Suzuki Ertiga, has MacPherson Struts suspension system in front that work in tandem with a Torsion Beam suspension at the rear. The suspension setup feels the defects of pavement when we ride on speed below 50 km/hr. The long wheel base contributes to the good stability and wide tyres also cushion the ride well and offer decent grip. The sensor would be mounted on vehicle dashboard the suspension system at lower speeds would help in controlling the noises and can detect damage that we can clearly notice by observing accelerometer readings. The length of vehicle is high enough that the data sampling won't be disturbed when back wheel when passes an anomaly. The width of vehicle can easily cover majority of road anomalies on pavement surface. The dashboard is strong enough that won't produce unwanted noises, since the sensor mounted on dashboard. The detailed information about car and its suspension system is mentioned in Table3. The placement of the data acquisition system is shown in Fig 4.1 below.

Table 3 Car model design and suspension system of vehicle selected for study

Engine Type	K15C Smart Hybrid
Length (mm)	4395
Width (mm)	1735
Wheel Base (mm)	2740
Gross Weight (Kg)	1785
Power Socket	12V
Dashboard	Sculpted with metallic Teak-Wooden finish
Front suspension	Mac pherson strut & coil spring
Rear suspension	Torsion beam & coil spring

Fig 4. 1 Data acquisition system on car dashboard

4.2 ROAD NETWORK

Among all road networks in Indore, the road connecting Indore and Khandwa is one of the routes that vehicle owners could avoid to travel due to presence of potholes, defected pavement and unexpected curves. So, we selected small length of roads from IIT Indore gate no.1 to gavhalu, which is of length 18.7 km and IIT gate no.1 to tejaji nagar of length 13.2 km. The accelerometer data was collected over these two road networks by mounting sensor on the car dashboard. The routes maps that we collected the data were shown below in Fig 4.2.

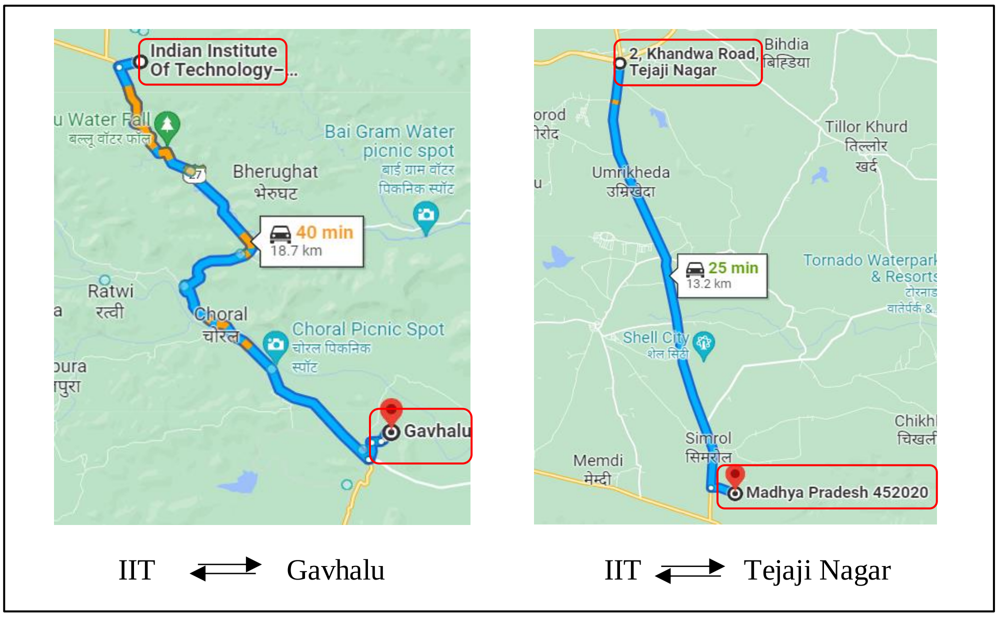


Fig4. 2 Routes for field study

The type of damages that we observed on the selected road are shown in below Fig 4.3 along with their corresponding accelerometer serial readings plot. In serial plot we can detect damage when the accelerometer readings exceed the threshold limit i.e., g+0.3g (12.8 m/s^2) or g-0.3g (6.87m/s^2), where g=9.81m/s^2. When vehicle descends into damage the acceleration due to gravity decreases and results in fall of z-axis accelerometer readings. When vehicle ascends out of damage the acceleration due to gravity increases and results in rise of z-axis accelerometer readings.

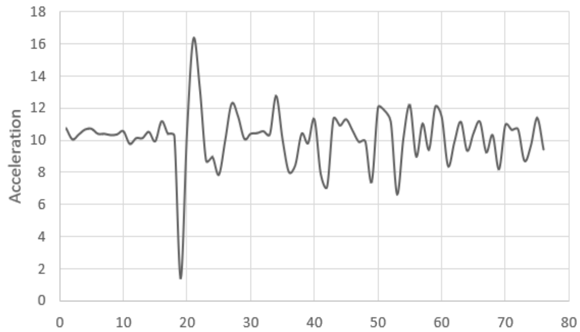
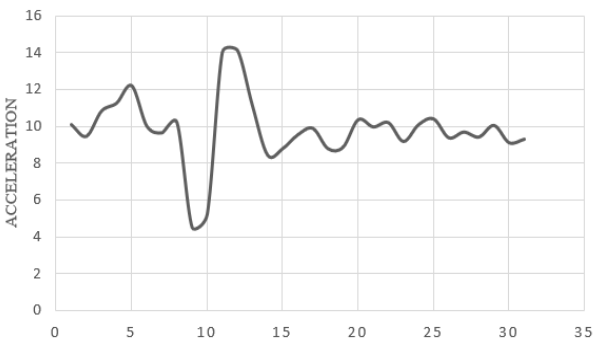
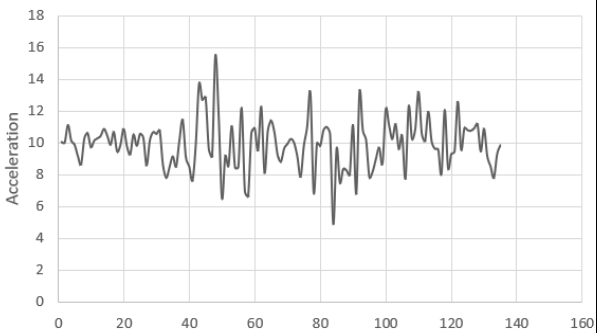
	Accelerometer serial plot (a)	Defect (b)
a		
b		
c		

Fig4. 3 Serial plot for different type of surface defects.

In Fig 4.3 the signal Fig 4.3.a along x-axis there is sudden fall and rise of acceleration and exceeds the threshold limits that is caused due to the car passed through the pothole and after that of there is still acceleration exceeds the threshold values due to vehicle movement on rut. As we can observe in corresponding image car first passes through pothole and continue on rut. In Fig 4.3.b there is fall and rise acceleration is due to pothole and after that there is no damage indication and can observe in the side image car first passes through pothole and then smooth surface. In Fig 4.3.c there is long fall and rise in acceleration is due to the vehicle crossed through rut and. we can observe in the side image car first passes through long rut.

4.3 DATA ACQUISITION SYSTEM

4.3.1 ARDUINO

The open-source Arduino platform is used to create electrical projects. Both a physical programmable circuit board (microcontroller) and computer software, known as the IDE (Integrated Development Environment), make up the Arduino platform. The Arduino board can be programmed with the IDE and then uploaded. The Arduino does not require a separate piece of hardware (referred to as a programmer) in order to load fresh code onto the board; instead, we can do it by using a USB cable attached to a computer. Additionally, the Arduino IDE employs a condensed form of C++ that makes learning to programme simpler.

Arduino IDE provides us similar environment like C++ IDE. We have to install required libraries in the IDE itself. After installation of libraries, we can easily compile the examples in that library. Micro-controller consists of reset button such that we can reset the code and can re-upload. The power can be supplied to the hardware Arduino board either through data logger from computer or by connecting external battery of 3.3 volts to 5 volts to power supply socket. The Fig 4.4 below depicts the Arduino board & power supplied through computer through USB cable.

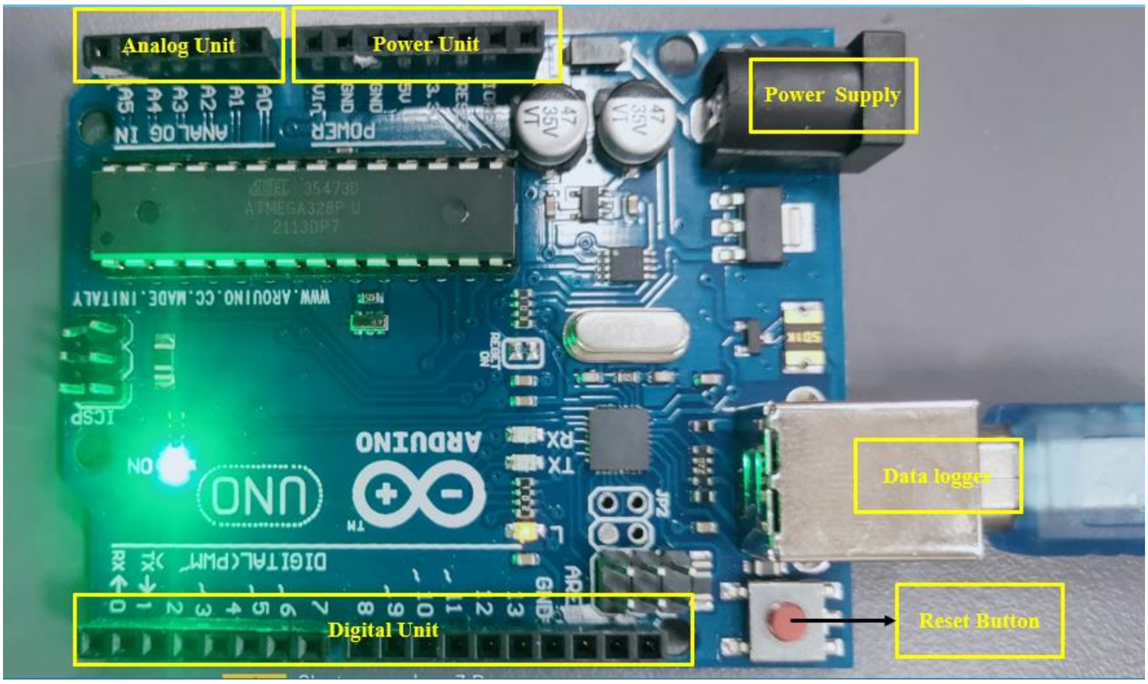


Fig 4. 4 Arduino micro-controller with labelling

Arduino micro-controller consists of power input unit, analog input unit and Digital input unit as labelled in above Fig 4.6. The input pins of the power unit, analog units and digital unit are shown in table4.

Table 4 Description of pins in Arduino micro-controller

Input pins	Input pins
Power input	IOREF, RESET, 3.3V ,5V, GND, GND, VIn
Digital input	Integrated output pins of D0 to D13, GND, AREF
Analog input	A0, A1, A2, A3, A4, A5

4.3.2 ADXL345

ADXL345 (in Fig 4.7) is a MEMS (micro electro-mechanical sensor) triaxial accelerometer works with low power, high resolution, adjustable frequency, and can be interfaced with both Arduino board and raspberry pi. It operates on the premise that vibration-induced force causes the mass to contract, producing an electrical charge proportionate to the force applied to the piezoelectric material. In other words, when the sensor moves in a specific direction, it gives back a number for how quickly it travelled in terms of the acceleration brought on by gravity along all of its axes. ($g = 9.81 m/s^2$)

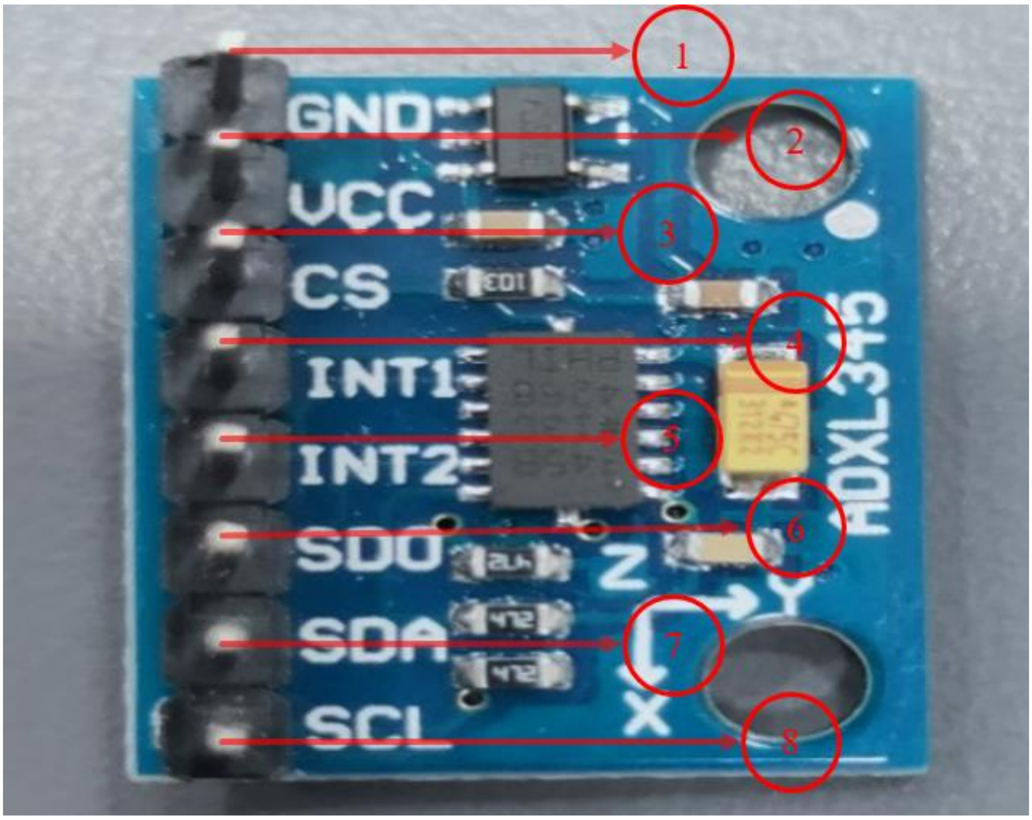


Fig 4. 5 ADXL345 Tri-axial accelerometer

Table 5 shown below represents the specifications of ADXL345 accelerometer taken for the study.

Table 5 Specifications of ADXL345 accelerometer

Accelerometer	ADXL345
Max Range	+16g m/s^2
Min Range	-16g m/s^2
Resolution	0.04 m/s^2
Data Rate	100 Hz
Selected range	±16g m/s^2
Sensitivity	32 LSB/g

In this study ADXL345 accelerometer is used for record the change of accelerometer data calibrated with Arduino micro-controller and interfaced to computer. It consists of 8 connectable pins such as GND, VCC, CS, INT1, INT2, SDO, SDA, SCL. The detailed information about these pins is shown in table 6. The pins are clearly visible in Fig 4.5 along with their numbering and the description of corresponding numbered pin is shown in table 6.

Table 6 Description of ADXL345 accelerometer pins

1	GND	Pin connected to ground
2	VCC	DC voltage supply
3	CS	Chip selected
4	INT1	Interrupt 1 output
5	INT2	Interrupt 2 output
6	SDO	Serial data output
7	SDA	Serial data (I^2C)
8	SCL	Serial communication clock

4.3.3 PuTTY

PuTTY is a network file transfer programme, serial console, and terminal emulator that is free and open-source. It was developed by Simon Tatham and is compatible with Linux, macOS, and Microsoft Windows. SCP, SSH, Telnet, rlogin, and raw socket connections are just a few of the network protocols that it supports. It can join with a serial port as well. It serves as a substitute for telnet clients. SSH offers a secure, encrypted connection to the remote system, which is its main benefit. It can also be transported on a floppy disc and is compact and self-contained. It is therefore perfect for securely connecting to Sussex systems from other places on the open Internet. Without the need for any hardware or other data logging devices, it is used to record any sensor data from an Arduino board to a computer. The advantages of PuTTY includes that it provides certain features when we working remotely, easier to configure, the remote connection and file transfer can be resumed, ease to use GUI, supports remote secure terminal configuration changes, connection can be restored after disconnection and the data can be saved directly in specified location after disconnection of sensor. The major disadvantages are it saves only username of session, emulator can’t compatible with some cisco networking equipment, copy and paste option will be disabled when connection is established.

4.3.4 CONNECTIONS

The data acquisition system consists of an ARDUINO genuine uno board, triaxial ADXL345 accelerometer, data logging cable and a laptop. The Fig 4.6 shown below is the data acquisition system. The Arduino and ADXL345 were connected by male-female wires. Connect the ground pin of Arduino to ground of ADXL345, voltage input of sensor to 5V of Arduino instead of 3.3V pin, because the power has been supplied through laptop. We acquire analog data series (I^2C) of the accelerometer, so the sensor has been connected with analog inputs A4, A5 connected to SDA, SCL respectively. The Table 7 shown below is the description of connections from ARDUINO board to ADXL345 accelerometer.

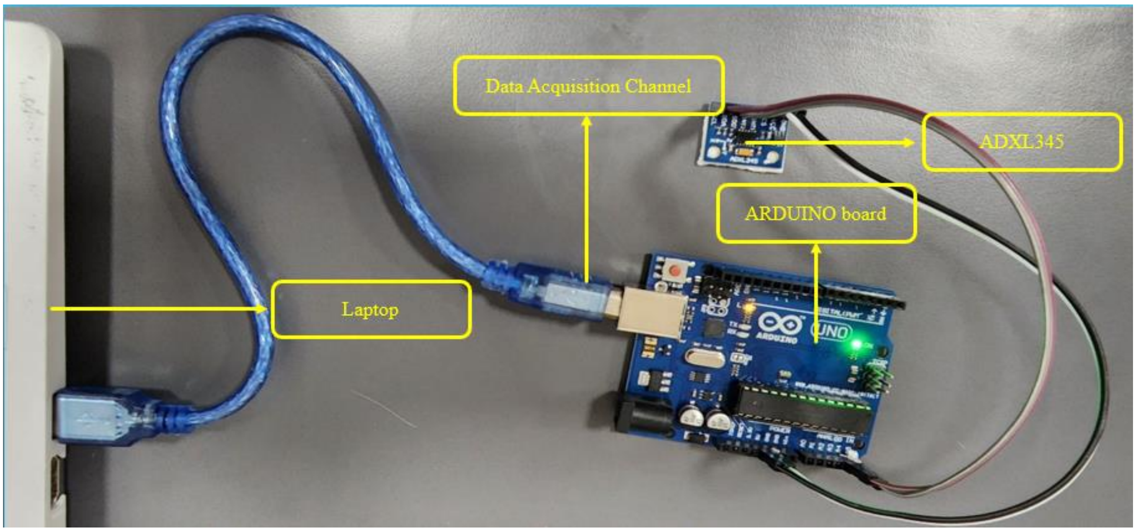


Fig 4. 6 Data acquisition system setup

Table 7 Pin to pin connections of Arduino board to ADXL345

ARDUINO Board	**ADXL345**
A5	SCL
A4	SDA
GND	GND
5V	VCC

After the completion of connections connect the data acquisition setup to laptop using A-B USB cable. Arduino is an open-source platform, download and install the Arduino IDE. We used

ADXL345 sensor, the sensor needs to be recognized by IDE so, install adafruit_sensor library from libraries section of IDE. We need the code to upload into Arduino board in order to acquire the acceleration data from ADXL345. The code available in examples section of adafruit_ADXL345 library with a name of sensor test. Import that code into Arduino IDE. Make delay value of less than 100 to update values faster set the sampling frequency of 100 Hz, subtract error in measuring acceleration in output function in three axes, set the acceleration range of ±16g m/s^2, remove the unwanted code like temperature readings, pressure readings etc. Finally, Compile and upload the code to Arduino and observed the readings in serial monitor. We obtained readings and specifications of sensor in serial monitor sensor were accurate.

The data we got on serial monitor wouldn't be recorded itself. To record those data, we would require a hardware device called SD/microSD breakout board with SD/microSD card or some software platforms like PuTTY, WinSCP, Bitwise, MobaXterm etc. To load and record the accelerometer data into computer, we installed PuTTY platform due to its advantages. To record data first compile and upload the code into Arduino board and observed the readings in serial monitor. PuTTY won't record when the data monitored in serial monitor. So, close the serial monitor and port the Arduino board in PuTTY and set path to save the data. Once the recording started in PuTTY, we can monitor serial readings on PuTTY terminal. After disconnecting the Arduino board, the data directly saved into the location that specified in PuTTY in the CSV format.

The Fig 4.7 shown below is the plot of road surface z-axis accelerometer data from tejaji-nagar to IIT Indore gate no.1.The X-axis of graph represents the length of road and Y-axis represents the acceleration value where acceleration due to gravity ($g = 9.81\ m/s^2$) was taken in this scenario.

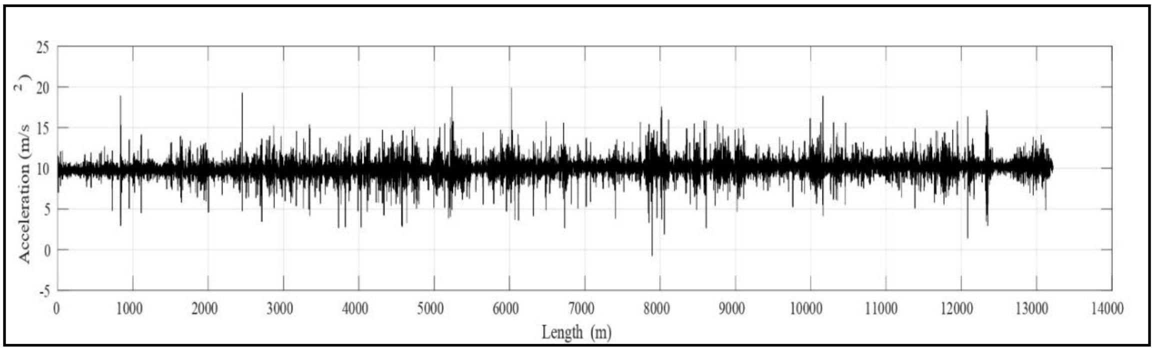


Fig 4. 7 Accelerometer data from Tejaji Nagar to IIT gate no.1

The Fig 4.8 below represents serial plot of the accelerometer data from IITI gateno.2 to gavhalu which is 17 km in length. Y-axis represents the accelerometer data where g=0 m/s^2 was taken in this scenario and X-axis represents length of road.

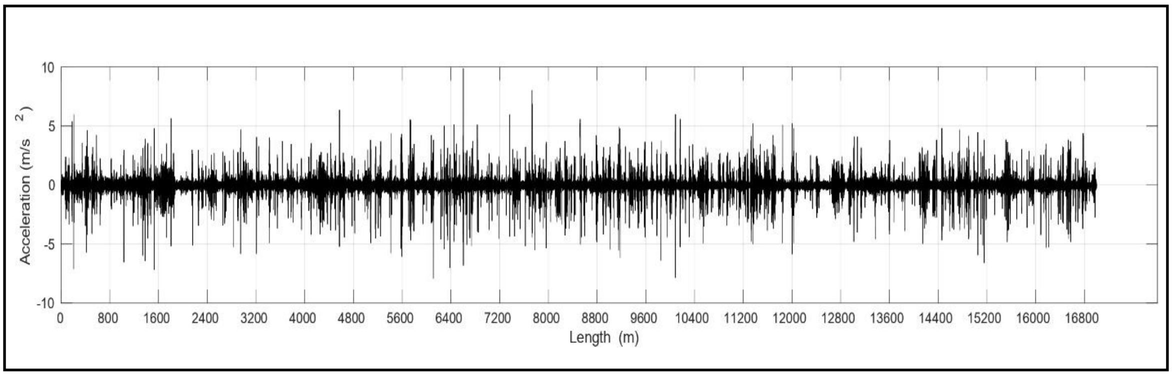


Fig 4. 8 Accelerometer data from IIT gate no.2 to Gavhalu

4.4 Extract features from field data

The data obtained from the accelerometer and Arduino (data acquisition system) is recorded into computer by using PuTTY application in csv format. The obtained csv data sheet then loaded into excel for pre-processing. The pre-processed data is then loaded using pandas library. I Apply sliding window technique on z-axis accelerometer data extract all 12 features from each window then fed them into prepared model and record the predictions for each window. Fig4.9 below shows the feature weights/importance of different features of trained model.

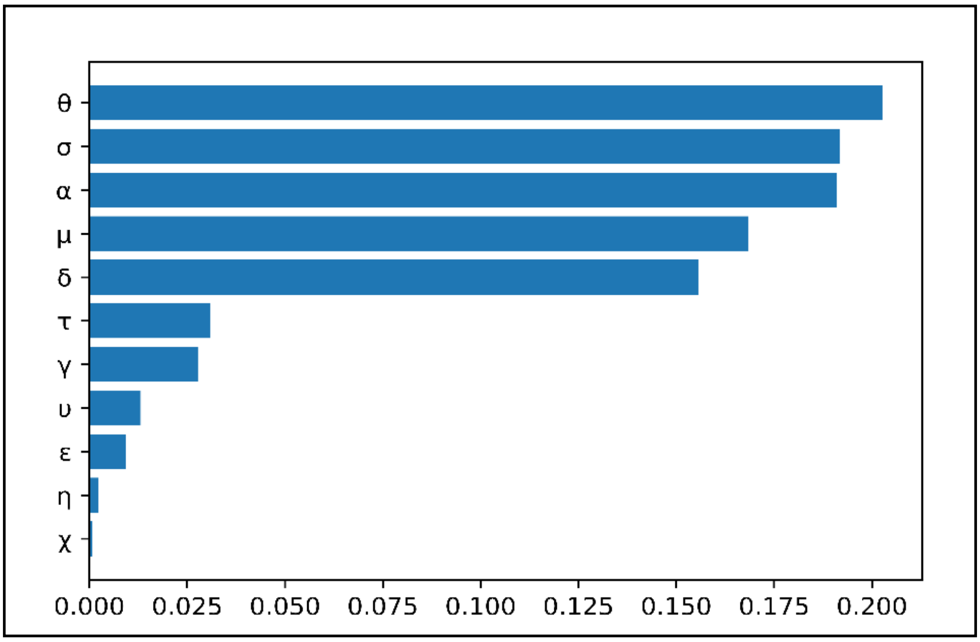


Fig4.9 Feature weights/importance

CHAPTER 5

5.1 RESULTS AND DISCUSSION

In numerical simulated study we built different models on simulated data and presented different scores, number of damages predicted for test dataset. After selecting the best model as discussed in the validation phase, we implemented the best model on field (experimental) data and make predictions. So, the best model random forest technique has been applied on field dataset.

In Fig 5.1 below, the serial plot of z-axis accelerometer readings from tejaji nagar to IIT Indore gate no.1, which is of length 13.2 kilometres. X-axis represents the length of road in meters and y-axis represents the z-axis accelerometer readings in m/s^2. The red dots represent the predicted position of defects along the road length, that is the distance of defect from starting point of studied road (tejajinagar) in meters. The position of defect on road is nothing but the perpendicular distance of corresponding red dot from y-axis in meters. As we can observe that the random forest model almost detects the exact position of defect on the road whenever the accelerometer readings exceed the threshold value.

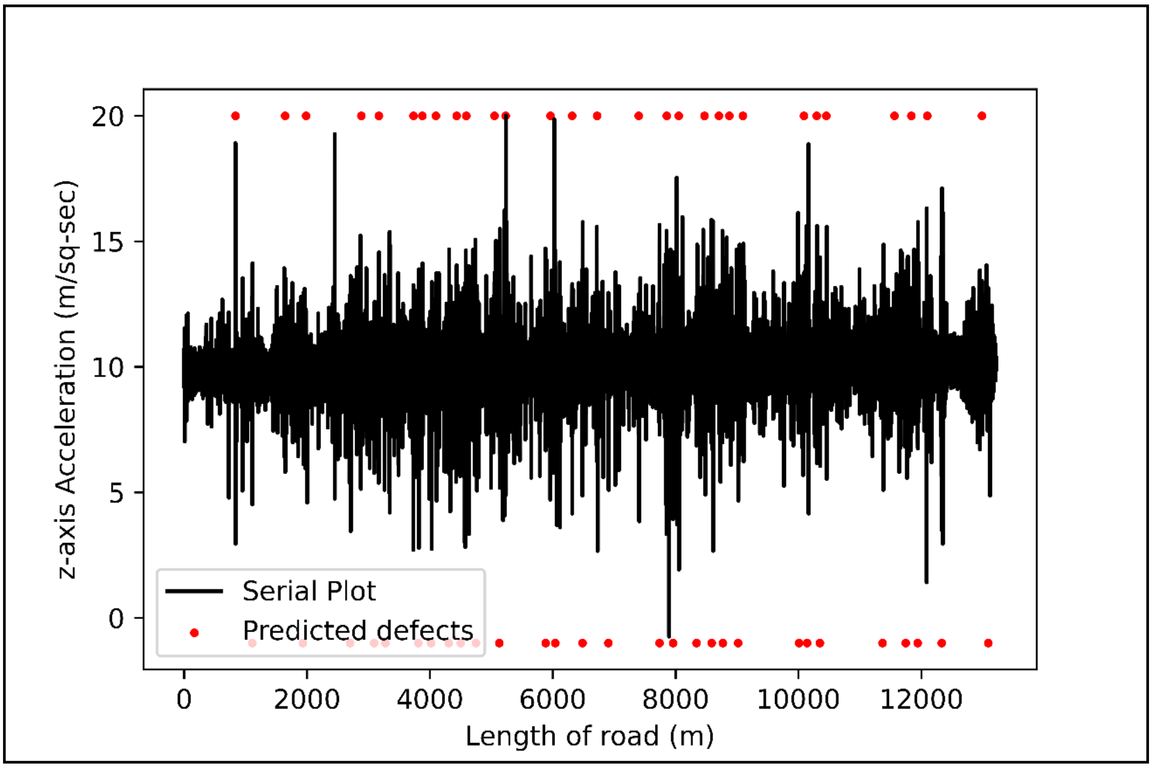


Fig 5. 1 Predicted defects along the road from Tejaji Nagar to IIT.

In Fig 5.2 below the red dashed lines represents the threshold barrier for accelerometer readings that contains defect. When the vehicle crosses the damaged surface, z-axis accelerometer readings exceed the threshold limit. We have made an observation from the field study for the majority of surface defects the threshold limit of z-axis acceleration was lies between g+0.3g (2.8 m/s^2) or g-0.3g (6.87m/s^2), where g=9.81m/s^2. In Fig 5.2 we can clearly observe that the predicted positions of damaged surfaces almost match with the original positions. So, from this we can say that our model is so accurate in prediction of precise position of road surface damage from its accelerometer readings.

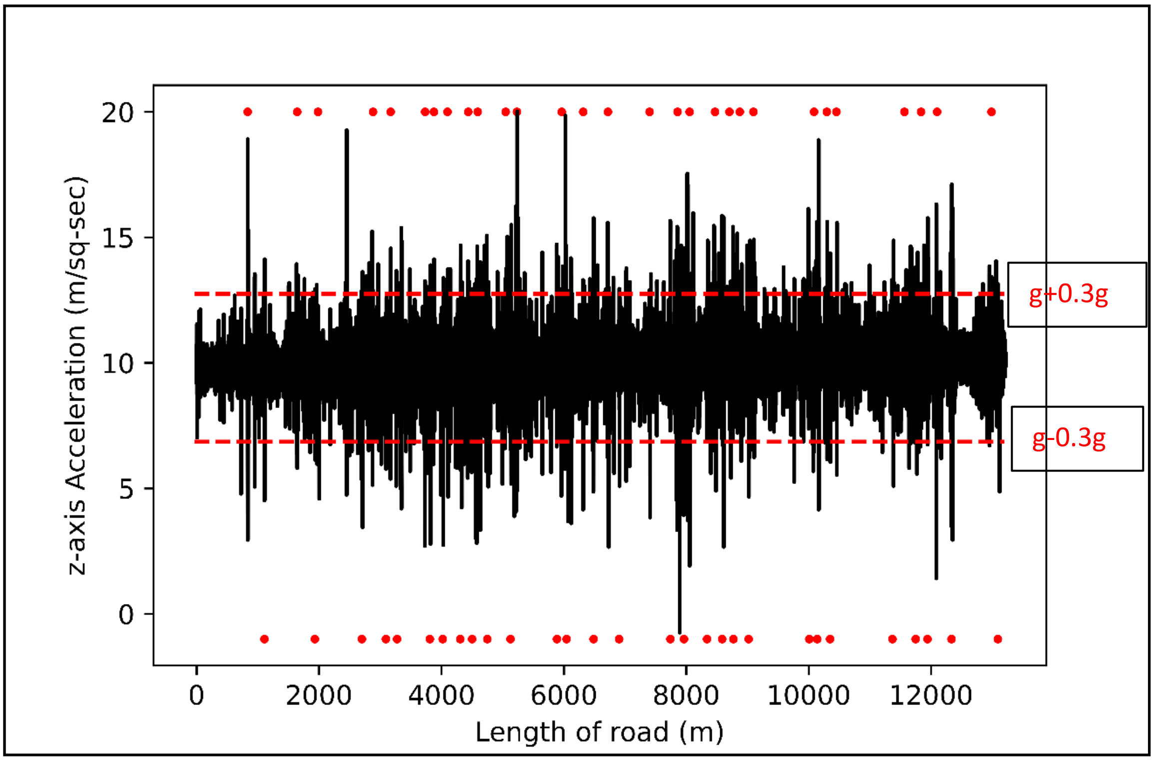


Fig 5. 2 Predicted damages versus original damages

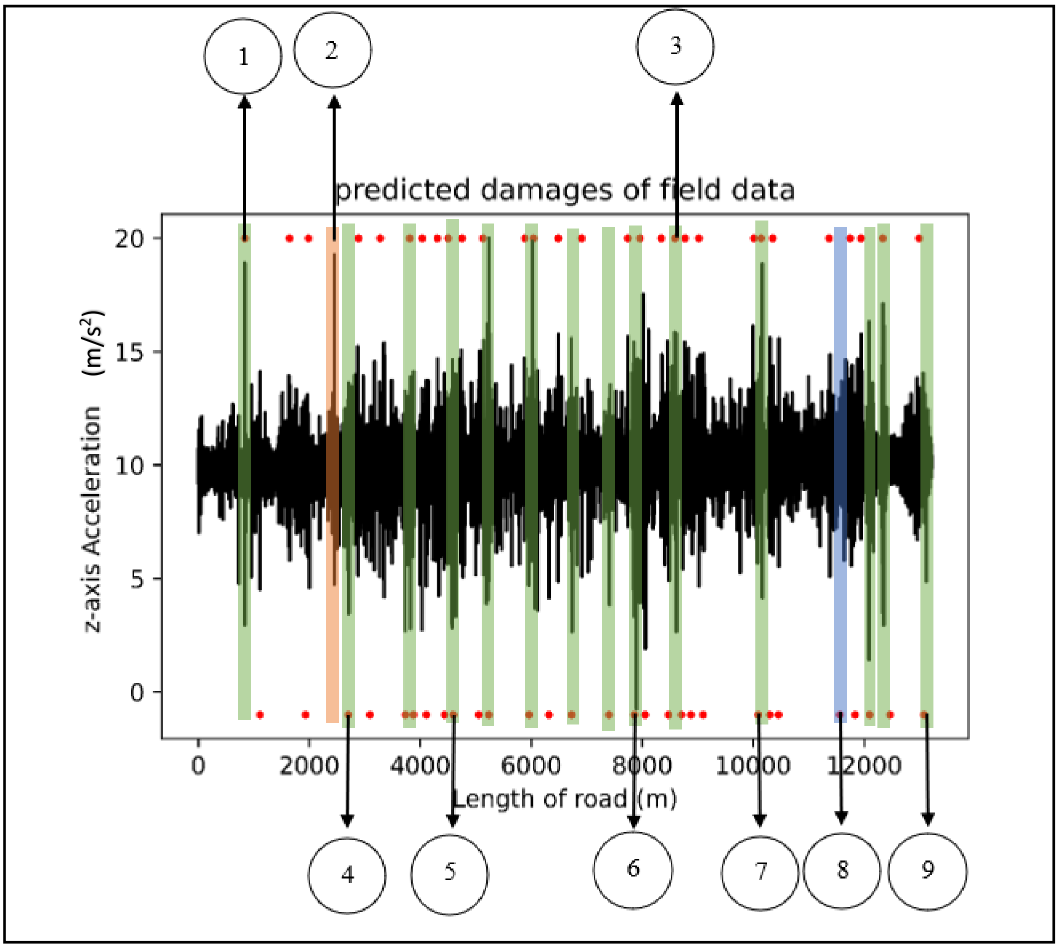


Fig 5. 3 Images numbers of some of predicted damages on tejaji to IIT road

From the above Fig 5.3 the green colour stripes represents the coincidence of both predicted and original damages along road of tejaji nagar to IIT gate no.1. The orange stripe (No.2) represents the wrong prediction of the model. Since there is some defect on road surface that we can observe in Fig 5.2 plot but our model didn't predict it as damage. The blue stripe (No.8) also represents the wrong prediction of the model. Since there is no defect on road surface that we can observe fin Fig 5.2 plot but our model predicts it as a damage. We have merged the results of both visual data (camera data) and accelerometer data the model accurately predicts the position of defect.

Some images of the road damages correspond to the strip numbers in Fig 5.3 are shown below. Each image in Fig 5.4 corresponds to the strip number in Fig 5.3.

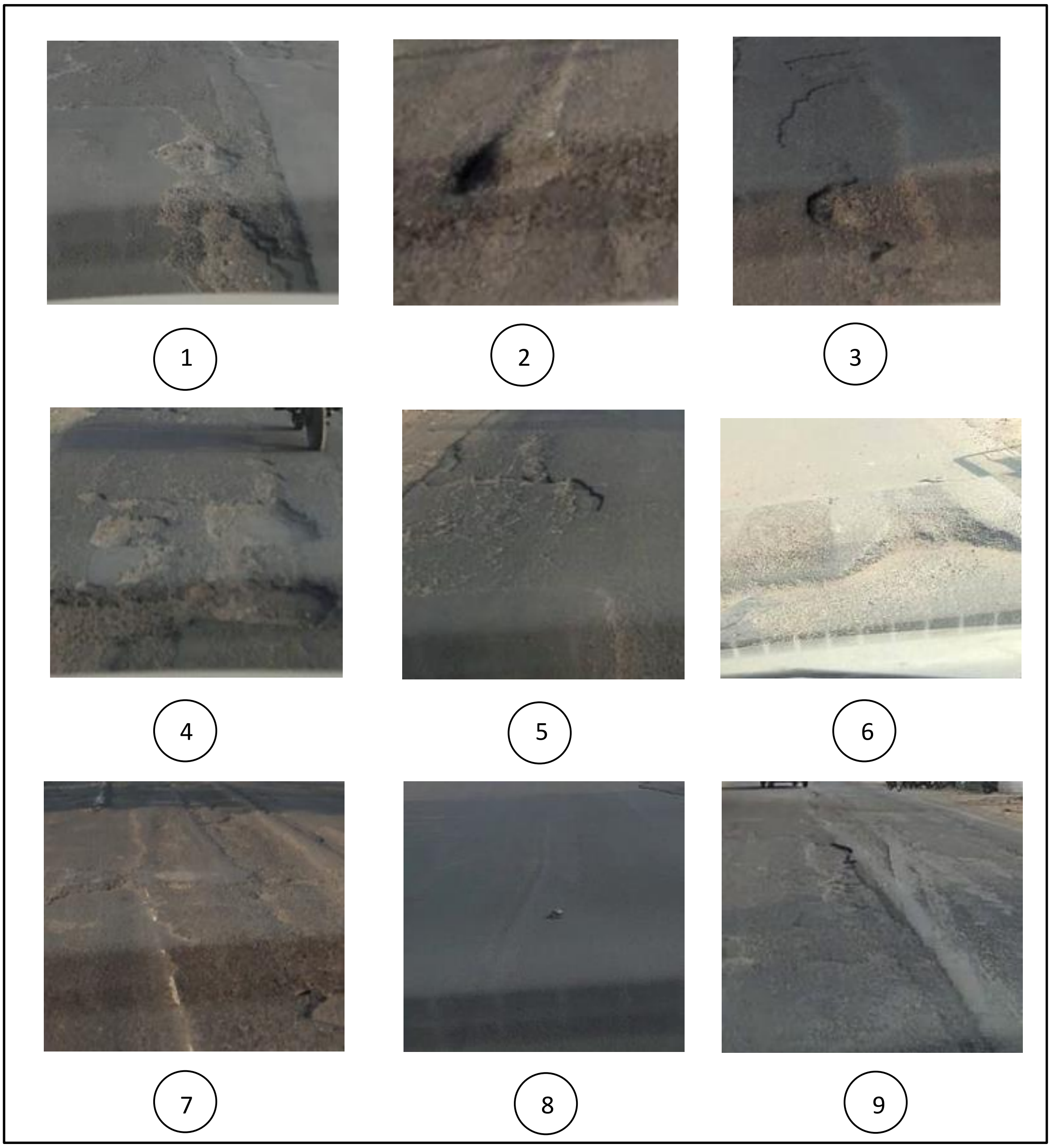


Fig 5. 4 Referenced images to corresponding numbered stripes

5.2 CONCLUSION

In this study, a novel algorithm using random forest (RF) technique has been developed for the detection of road anomalies. The performance of the proposedmodel has been compared with other models like decision tree (DT), k- nearest neigh-bours (KNN) and support vector machine (SVM). During testing, RF classifier has shown better accuracy for prediction of anomalies with duplicates and without duplicates. In both the cases, the performance of RF classifier is higher than the other models. The prepared model was applied on field data, it gave better and almost precise position and number of road surface defects. Hence it is clear that, random forest (RF) technique with the proposed features can be used for the health monitoring of longer road networks with semi-skilled labour and without procrastination.

Further we can make proceedings like cost estimation for repairing, alerting about upcoming pothole in rainy season etc., since this project detected position of damages on roads precisely. The rough length and depth of each anomaly can be estimated by accelerometer readings of damaged surface, this will be useful in the rough estimation of cost of repairing of road.

REFERENCES

1. Agrawal, H., Gupta, A., Sharma, A., and Singh, P. (2021). "Road pothole detection mechanism using mobile sensors." *2021 International Conference on Technological Advancements and Innovations (ICTAI)*, IEEE, 26–31
2. Carlos, M. R., Aragón, M. E., González, L. C., Escalante, H. J., and Martínez, F.(2018). "Evalu- ation of detection approaches for road anomalies based on accelerometer readings—addressing who's who." *IEEE Transactions on Intelligent Transportation Systems*, 19(10), 3334–3343.
3. Al Masud, A. J., Sharin, S. T., Shawon, K. F. T., and Zaman, Z. (2021). "Pothole detection using machine learning algorithms." *2021 15th International Conference on Signal Processing and Communication Systems (ICSPCS)*, IEEE, 1–5.
4. Fox, A., Kumar, B. V., Chen, J., and Bai, F. (2017). "Multi-lane pothole detection from crowd sourced undersampled vehicle sensor data." *IEEE Transactions on Mobile Computing,* 16(12), 3417–3430
5. Hoang, N.-D. (2018). "An artificial intelligence method for asphalt pavement pothole detection using least squares support vector machine and neural network with steerable filterbased feature extraction." *Advances in Civil Engineering,* 2018
6. Mohamed, A., Fouad, M. M. M., Elhariri, E., El-Bendary, N., Zawbaa, H. M., Tahoun, M., and Hassanien, A. E. (2015). "Roadmonitor: An intelligent road surface condition monitoringsystem." *Intelligent Systems' 2014,* Springer, 377–387
7. Anaissi, A., Khoa,N. L. D., Rakotoarivelo, T., Alamdari, M. M., andWang, Y. (2019). "Smart pothole detection system using vehicle-mounted sensors and machine learning." *Journal of Civil Structural Health Monitoring*, 9(1), 91–102
8. Anandhalli, M., Tanuja, A., Baligar, V. P., and Baligar, P. (2022). "Indian pothole detection based on cnn and anchor-based deep learning method." *International Journal of InformationTechnology,* 1–11.
9. Asad, M. H., Khaliq, S., Yousaf, M. H., Ullah, M. O., and Ahmad, A. (2022). "Pothole detection using deep learning: A real-time and ai-on-the-edge perspective." *Advances in CivilEngineering,* 2022

   Egaji, O. A., Evans, G., Griffiths, M. G., and Islas, G. (2021). "Real-time machine learningbased approach for pothole detection." *Expert Systems with Applications*, 184, 115562.
10. Bhatia, Y., Rai, R., Gupta, V., Aggarwal, N., Akula, A., et al. (2019). "Convolutional neural networks based potholes detection using thermal imaging." Journal of King Saud University-*Computer and Information Sciences.*
11. Pandey, A. K., Iqbal, R., Maniak, T., Karyotis, C., Akuma, S., and Palade, V. (2022). "Con- volution neural networks for pothole detection of critical road infrastructure." *Computers and Electrical Engineering*, 99, 107725.
12. Patra, S., Middya, A. I., and Roy, S. (2021). "Potspot: Participatory sensing based monitoring system for pothole detection using deep learning." *Multimedia Tools and Applications,* 80(16), 25171–25195
13. Pei, L., Shi, L., Sun, Z., Li,W., Gao, Y., and Chen, Y. (2022). "Detecting potholes in asphalt pavement under small-sample conditions based on improved faster region-based convolution neural networks." *Canadian Journal of Civil Engineering,* 49(2), 265–273.
14. Mednis, A., Strazdins, G., Zviedris, R., Kanonirs, G., and Selavo, L. (2011). "Real time pothole detection using android smartphones with accelerometers." *2011 International conference on distributed computing in sensor systems and workshops (DCOSS)*, IEEE, 1–6