International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-11, September 2019

Implementing IoT and Data Analytics To Overcome "Vehicles Danger"

Garima Sachdeva, Rishabh Verma, Deepak Chahal, Latika Kharb

Abstract*: In the fast pacing world, we commune from one place to another or one city to another with the means of vehicles. In recent years, data shows the worse conditions of the roads in Delhi with rapidly increasing of accidents. Roads in Delhi are more accident prone which creates more jeopardy to survive in Delhi for everyone. This problem persist from last few years due to various factors like open sewage, speed breakers, damaged roads which leads to loss of life of manhood and ecosystem. Since India is a developing nation there is a constant demand for good quality infrastructure, transportation and services. But since India is a vast country with quite a sizeable population this problem still has not yet addressed in totality. By using proposed methodology, for reducing the risk in the life of manhood as well as ecosystem by detecting all the factors for accidents and then improvement can be assured at the level of road safety management effectively and efficiently. Through this methodology, we can resolve these major issues by using Computer Vision, vision detector and Machine Learning so that drivers can easily aware of what ahead of them and act alertly before accident actually happens. Models are preinstalled on vehicles for safety purposes of the people, by using vision detector driver can acknowledge the obstruction ahead the car. Using the divergent and advanced technologies, live screening of the objects or obstacles ahead the vehicle will be visible to the driver for better ease and safe drive. Driver will get the notification from articulate assistant in any language by detecting the data ahead the car whilst specific range. Simple commands would comprise of "Speed Breaker ahead, slow down".

***Keywords* : Speed breakers, damaged roads, transportation and services, Computer Vision, Machine Learning.**

I. INTRODUCTION

In India, the modern age of computer automation defined in recent years led to certain solutions regarding technology, infrastructure, health and education. With higher and advanced technologies we are putting an effort to make people life at ease. In modern cities people are facing the impact of severe accidents due to varies causes that held people life at risk. Accidental causes can be open sewage, abrupt and disrupted speed breakers, unforeseen animals that can deform the life of ecosystem and manhood. This proposed methodology depicts the approach to manifest the sight of any vehicles' driver by using advanced technologies. Using this obscure approach, we render proficiency regarding Machine Learning, IoT and Sensors with the accomplishment of safety for manhood.

A. Machine Learning

Machine learning is the genius of the modern age computer to forge human life upgradable by rendering the Machine's Neural Network without any explicit commands. Artificial Intelligence involves Machine Learning and Deep Learning in which Machine learning is the subset of Artificial Intelligence, and Deep Learning is the subset of Machine Learning. This procedure train a machine or any system using data sets and by doing so machine will fetch the given data sets in a loop one task at a time that is stated by human. Data sets are the accumulated structured data which are further summoned up to form data loader for any task to perform whenever user gives input. The generated input will be pass through the different hidden layers of neural network and processed in each layer which detects a pattern in the input data. The final layer of the neural network will produce the predictions on the basis of given input

B. Internet of Things and Sensors

IoT are the physical devises used to communicate with each other without any interference of human. Once the Internet gets connected to any device or machine then it will be able to deploy its task automatically without any user command on its own. All these devices have an IP address installed which links and connect them together to fetch a significant and organized data and that data will be summoned up to main server for further use. IoT has few devices that are small and concealed which gets connected to the different sensors so that sensors will be able to fetch the data, and further processed it for user's convenience. Raspberry Pi and Arduino are the IoT devices which directly attach to sensors through Internet. As more and more applications are deployed using the Internet of Things (IoT) technologies, the fragmentation of general purpose IoT technologies to target particular sectors with different requirements is becoming necessary[1].

II. NEED OF THE PROPOSED SYSTEM

This journal uses double-blind review process, which means that both the reviewer (s) and author (s) identities concealed from the reviewers, and vice versa, throughout the review process. All submitted manuscripts are reviewed by three reviewer one from India and rest two from overseas.

Manuscript published on 30 September 2019.
*****Correspondence Author(s)
Garima Sachdeva, Student(MCA), Jagan Institute Of Management Studies, Delhi, India.
Rishabh Verma, Student(MCA), Jagan Institute Of Management Studies, Delhi, India.
Deepak Chahal, Professor, Jagan Institute Of Management Studies, Delhi, India.
Latika Kharb, Professor, Jagan Institute Of Management Studies, Delhi, India.

There should be proper comments of the reviewers for the purpose of acceptance/ rejection. There should be minimum 01 to 02 week time window for it.

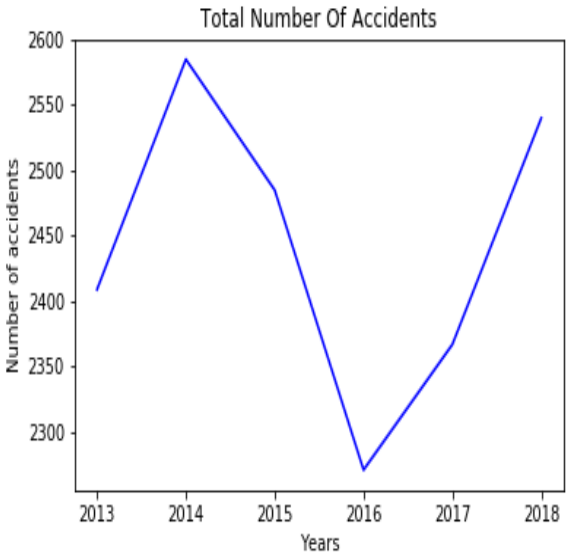

Figure 1:Total Number of Accidents Graph

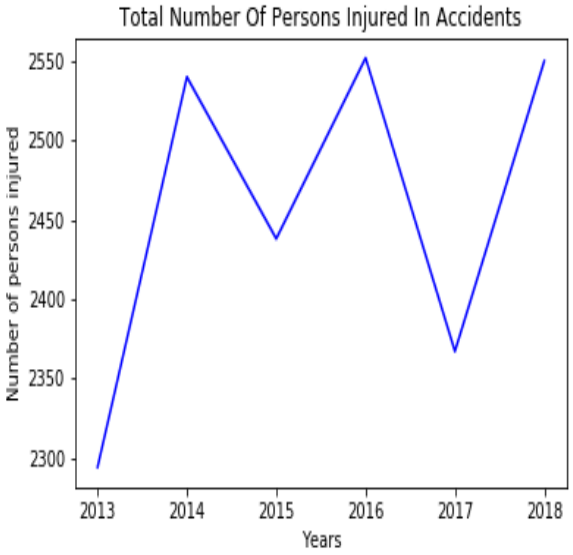

Figure 2: Total Number of Persons Injured in Accidents Graph

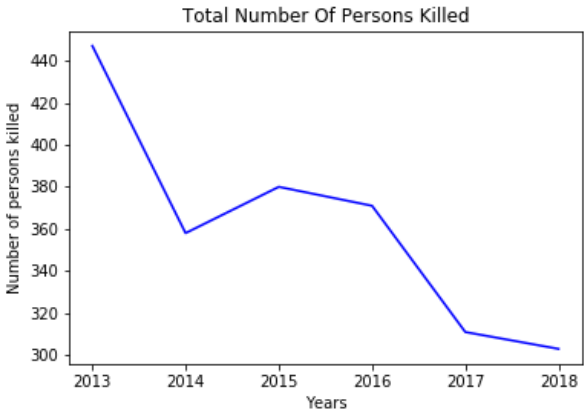

Figure 3: Total Number of Persons Killed in Accidents Graph

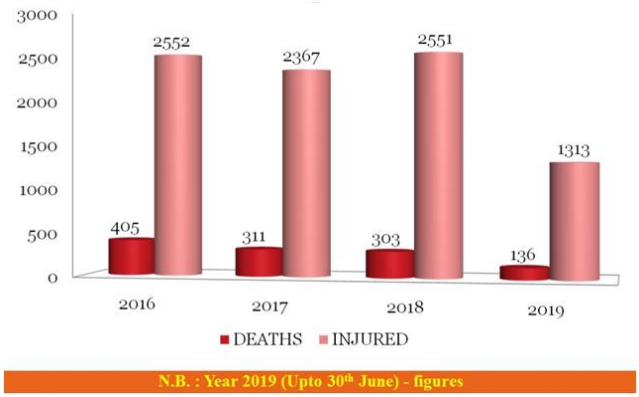

Figure 4: Bar chart of total deaths and persons injured till June 2019.

III. PROPOSED IDEA: VEHICLES DANGER

Through this Data Analysis on the “Delhi Traffic Police” author conclude that in the world of modernization we still lack the safety and prevention of the people in the advanced country with high-tech technologies. The major source of people’s devastation leads to the economic havoc which entails major issues of the country mostly with ignorance of traffic rules, damaged roads, speed breakers and more. People are more engaged in their busy life which leads to more distraction, stress and lack of focus which causes accidents.

To overcome such impact of traffic congestions, it is required to develop an IoT Based traffic control system. The proposed system would be based on the measurement of the actual traffic density on the road [2].Traffic rules are disregarded by the people which involves even basic rules such as not following traffic lights, ignoring seat belts, getting away with zebra crossing, using smartphones whilst driving and more reasons for people and ecosystem damage.

Sl.No.	Month	U/s 304(a) (Fatal)				U/s 338 (Grevious Injury)				U/s 337 (Simple Injury)				Total accidents				No. of persons killed in accidents				No. of persons injured in accidents			
		2016	2017	2018	2019	2016	2017	2018	2019	2016	2017	2018	2019	2016	2017	2018	2019	2016	2017	2018	2019	2016	2017	2018	2019
1	January	19	32	22	24	9	15	17	15	145	174	181	192	173	221	220	231	20	33	22	26	197	224	229	222
2	February	31	19	20	26	10	11	10	12	144	165	164	180	185	195	194	218	31	20	21	28	167	199	195	217
3	March	42	31	23	19	16	14	18	10	166	159	156	179	224	204	197	208	45	31	24	19	253	197	204	217
4	April	37	23	20	20	14	17	16	19	160	148	155	147	211	188	191	186	38	26	21	21	210	193	195	202
5	May	39	23	25	25	14	16	10	18	157	139	143	174	210	178	178	217	40	25	26	27	196	168	185	238
6	June	32	21	36	15	14	9	9	14	155	170	182	169	201	200	227	198	33	21	36	15	192	200	214	217
7	July	38	20	26		11	10	13		174	173	166		223	203	205		38	20	27		213	203	203	
8	August	35	28	26		11	16	12		167	142	169		213	186	207		35	28	26		218	182	208	
9	September	35	28	27		14	9	14		154	148	167		203	185	208		35	28	29		195	185	201	
10	October	24	27	22		12	14	14		163	147	196		199	188	232		25	27	22		204	187	229	
11	November	31	25	24		14	12	21		184	159	223		229	196	268		32	26	26		228	196	274	
12	December	33	26	23		10	18	16		221	179	174		264	223	213		34	26	23		279	233	213	
	Total	396	303	294	129	149	161	170	88	1990	1903	2076	1041	2535	2367	2540	1258	406	311	303	136	2552	2367	2550	1313

Figure 5: Month wise Statement of various accidents for the years 2016-2019 (up to June 2019), Delhi City

The major obstacle today in various cities are vandalize and wrecked roads which leads to more accident prone for the economy. These issues can sabotage the road blockage, traffic jams, accidents and can affect the tourism which further demolish the economy. Pit holes and open sewage are unpredictable and hazardous which harms the life of animals, deformed people, pregnant women and delays the transportation. During rainy season, pit holes and sewage filled up with polluted and contaminated water which cause the harmful mosquitos and parasites. The piled up filthy water in heavy rain makes the vague and uncertain prediction for the drivers and animals which sometimes cause them to fall into it. Various cities like Delhi, Hyderabad, Bangalore, Mumbai has unstructured and havoc roads which created a chaos and lead to ecosystem destruction. Through this proposal we tend to evolve secured and undamaged life for manhood as well as ecosystem. Since this approach guide any vehicle driver to have a clear vision with the help of Machine Learning and IoT, drivers would be able to easily and safely drive in dusk and dawn with the help of camera and voice assistant. User can easily install a sensor and camera or with a help of a technician, sensor will be directly connected to the camera and voice assistant so that driver will be able to use both of them easily. Camera which is connected to the sensor will detect and recognize sensor's data and gives the clear live screening. The voice assistant will verbalize an appropriate message for the driver safety.

Machine Learning and IoT are the major technologies used in this project for further collaboration of technologies used. These technologies in collaboration guide us to accomplish our project on "Vehicle Danger". This concludes the overall project provide service in saving the life of manhood and animals without any hindrance.

IV. IMPLEMENTATION OF PROPOSED IDEA

The execution of this project will begin from gathering the data sets till the voice assistant message. Data sets will be of pit holes and sewage, animals (dogs and cats) which guide Machine Learning to learn its task to perform further. Data will get clean for the afterward pre processing which creates the correct and meaningful data sets and get through the Neural Networks (model) for the training. In Machine Learning, we create and train Neural Networks for the predictions using different data sets, the predictions will depict the higher priority probability of the data set top classes objects in the images. The trained model will be connected to the IoT device (Raspberry pi) further it will receive the image as input through the sensors and it will make predictions.

The vision detector that will be installed in the car for the input of the data as Image that will be directly connected to the IoT device which passes the data further to the trained model. vision detector will be attached to the IoT device to get a required power supply for the further execution of its task.

The predicted output will be classified into four different classes which are pit holes, sewage and animals. The object seen in the Image will be classified in any of the above classes for the higher predictions probability. Using the probability generated from any of the object classes then it will use the voice assistant to give the message or command. Simultaneously, the live screening of the front view of the car will be shown on the monitor kit with the help of camera installed on bonnet of the car. So that it will ease the driver for using both articulate assistant and live screening.

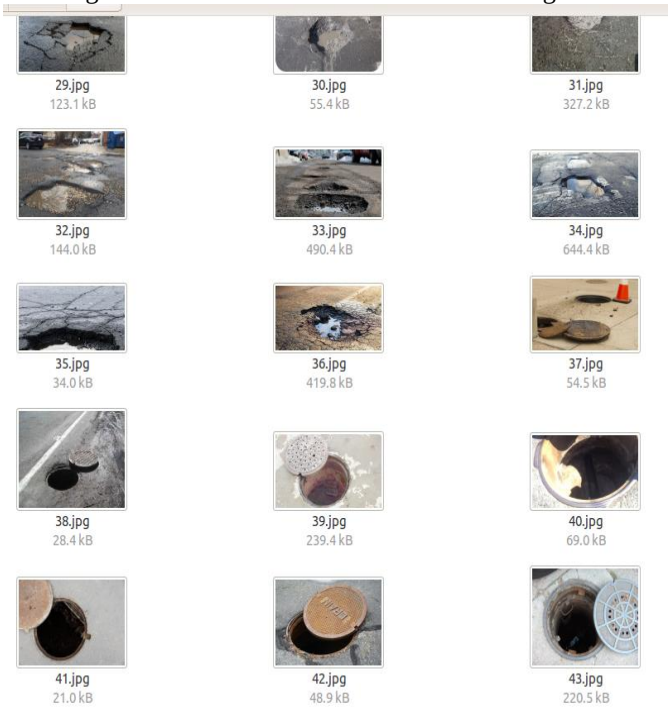

Figure 6: Pit Holes Data Sets

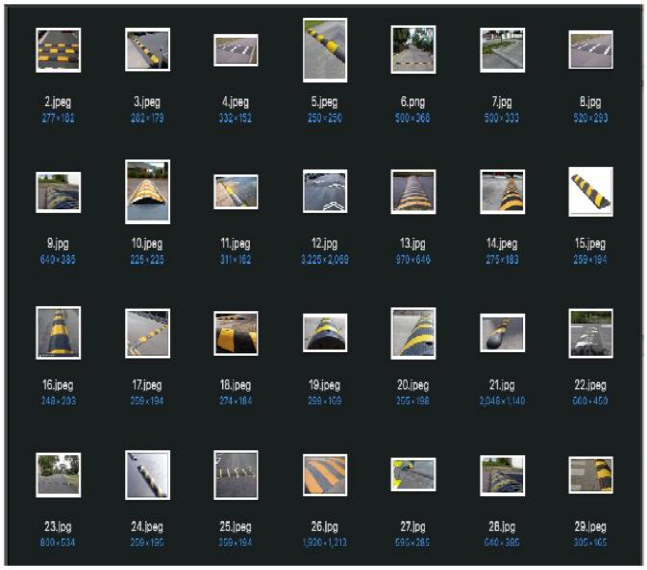

Figure 7: Speed Breaker Datase

A. PyTorch Deep Learning Framework

It is used for transfer learning and to build deep learning neural networks. Transfer Learning is used to achieve higher accuracy, we download the pre trained neural network and train it on our required data set. PyTorch is a Python library enabling GPU accelerated tensor computation, similar to NumPy. A few advantages of using PyTorch are it's multi-GPU support, dynamic computational graphs, custom data loaders, optimization of tasks, and memory managements [3].

B. Numpy Library

Python's popularity probably stems from its relative ease of use (even for non-computer scientists), huge ecosystem consisting of a number of libraries for every aspect of data science and its reliance via *NumPy* and *SciPy* wrappers on the fast implementations of a large number of scientific algorithms written in C and Fortran[4]. NumPy library is used to solve complex and scientific calculations. It also apply for execution of Image Pre-processing and remould into PyTorch tensors to the NumPy array.

C. Seaborn And Matplotlib Library

They are data visualization library based on matplotlib and used for creating high level statistical graphs. Tools in this category display analytics results in an interactive way, so that they can ease the understanding of difficult concepts and support decision makers. There are many data visualization packages at various levels of abstraction in R or Python; e.g., matplotlib, ggplot, seaborn, plotly, bokeh[5].MatPlot Library is used for multi-dimensional data visualization built on NumPy arrays and design statisticalgraphs.

```
import torch
import numpy as np
import torch.nn as nn
import matplotlib.pyplot as plt
import seaborn as sns
from torch import optim
from copy import deepcopy
from torchvision import datasets, transforms, models
from PIL import Image
```

Figure 8: Importing library

D. Data Transformations

We trained algorithms on data from the evaluation sample before they were used to predict the diagnostic outcome in the validation dataset. We compared the predictions made on the validation datasets with the real-world diagnostic decisions to calculate the accuracy, sensitivity, and specificity of the models[6].

Training Set Various transformation is applied on data sets like scalar, resize and rotation.The real world data can be of any form so our model must be generalise which can easily detect and recognize the objects in the image. Images recognized will be formed in PyTorch's tensor and Normalize the images to set its pixel value between 0-1. Images in the test set will be in 224 x 224 pixel so that model will receive the standard size image without any conflict. Also the images diagnosed will be reformed in PyTorch's tensor and Normalize it to set its pixel.

```
# Train data augmentation
train_transforms = transforms.Compose([
    transforms.Resize(255),
    transforms.CenterCrop(224),
    transforms.RandomHorizontalFlip(),
    transforms.RandomRotation(10),
    transforms.ToTensor(),
    transforms.Normalize((0.485, 0.456, 0.406), (0.229, 0.224, 0.225))
])

# Test data augmentation
test_transforms = transforms.Compose([
    transforms.Resize(255),
    transforms.CenterCrop(224),
    transforms.ToTensor(),
    transforms.Normalize((0.485, 0.456, 0.406), (0.229, 0.224, 0.225))
])
```

Figure 9:Training Data

V. Data Loaders

This helps to splits the data sets into training and test set and load the image from the datasets. The loaded images from the training directory will form the training set and similarly images from test directory will form the test set.
Using PyTorch data loader class, model will load the specified batch size images and store in trainloader and test loader respectively.

```
# Train Set
train_set = datasets.ImageFolder(train_dir, transform=train_transforms)

# Test Set
test_set = datasets.ImageFolder(test_dir, transform=test_transforms)

[ ] # Train Loader
train_loader = torch.utils.data.DataLoader(train_set, batch_size=batch_size,
                                           num_workers=num_workers, shuffle=True)

    # Test Loader
test_loader = torch.utils.data.DataLoader(test_set, batch_size=batch_size,
                                          num_workers=num_workers, shuffle=True)
```

Figure 10:Train set, Test set, Train loader and Test loader

VI. Final Output Layer

After Transfer Learning, final layer will be replaced by customized final layer so that model will classify the different prediction of classes according to our data set. This function will generate final layer as stated in our requirements, till now requirements are classified in three types of categories.

Retrieval Number: K21530981119/19©BEIESP
DOI: 10.35940/ijitee.K2153.0981119
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

```
[6]  def custom_classifier():
       return nn.Sequential(
           nn.Linear(512, 256),
           nn.ReLU(),
           nn.Dropout(p=0.2),
           nn.Linear(256, 2),
           nn.LogSoftmax(dim=1)
       )

     model.fc = custom_classifier()
```

Figure 11: Classify the different prediction of classes

VII. TRAINING CLASSIFIER AND VALIDATION PASS

For training the pit holes, sewage and animals classifier. It will help us to determine the goldilocks-spot (i.e. At which developer will stop training the model). This function will utilize the model and set on either CPU or GPU, training pass will iterates over number of epochs specified and it will train the model using BackPropogation methodology. Simultaneously, validation pass will iterate on each epoch's for evaluating our model accuracy. Finally, it returns the trained model with its best_model_weights(state_dict).

```
def train_model(model, criterion, optimizer, scheduler, device, n_epochs=20):

  model.to(device)

  test_loss_min = np.Inf

  best_acc = 0.0

  best_model_weights = deepcopy(model.state_dict())

  for epoch in range(n_epochs):

    train_loss, test_loss = 0.0, 0.0

    # learning rate scheduler step StepLR
    scheduler.step()

    # Training pass
    model.train()
    for images, labels in train_loader:

      images, labels = images.to(device), labels.to(device)

      optimizer.zero_grad()

      lops = model(images)

      loss = criterion(lops, labels)

      loss.backward()

      optimizer.step()

      train_loss += loss.item() * images.size(0)
```

Figure 12: Training Model

```
    # Validation Pass
    model.eval()
    with torch.no_grad():

      for images, labels in test_loader:

        images, labels = images.to(device), labels.to(device)

        lops = model(images)

        loss = criterion(lops, labels)

        test_loss += loss.item() * images.size(0)


    train_loss /= len(train_loader.dataset)
    test_loss /= len(test_loader.dataset)

    print('Epoch {}/{}\tTrain loss: {:.6f}\t Validation loss: {:.6f}'.format(
        epoch+1,
        n_epochs,
        train_loss,
        test_loss
    ))

    if test_loss <= test_loss_min:
      print('Validation loss decreaesd from {:.6f} ---> {:.6f}'.format(
          test_loss_min,
          test_loss
      ))
      test_loss_min = test_loss
      best_model_weights = deepcopy(model.state_dict())


  model.load_state_dict(best_model_weights)
  return model
```

Figure 13: Validation Pass

VIII. ACCURACY

This function makes the use of test set and passes each image in the test set to the trained model. The predicted class with highest probability will be compared with its corresponding labels through which we get the number of correct predicted class label.

```
def calc_accuracy(model):
  accuracy = 0.0
  model.eval()
  with torch.no_grad():
    for images, labels in test_loader:
      images, labels = images.to(device), labels.to(device)
      logps = model(images)
      ps = torch.exp(logps)
      _, top_class = ps.topk(1, dim=1)
      equals = top_class == labels.view(*top_class.shape)
      accuracy += torch.mean(equals.type(torch.FloatTensor)).item()
  print('Overall Accuracy is: {:.6f}'.format(accuracy/len(test_loader)))
calc_accuracy(model)
```

```
Overall Accuracy is: 0.980655
```

Figure 14: Calculating Accuracy

IX. NOMOGRAM VISUALIZATION

Figure 15 depicts the statistical graph of training and validation losses at each epoch by using matplotlib library. Losses shows how far our contemporary model in producing the right prediction.

Figure 15: Statistical graph of training and validation losses

Figure 16 shows the training and test accuracy model at each epoch and finally our model is manifest 90% accuracy.

Figure 16: Training and test accuracy model at each epoch

X. ASSUMPTION

This function takes a random image as input and pass it to the model which predicts the object in the image. Also it will display the given image and plot its probability distribution graph.

```
def plot_prediction(image_path, model=None):
    # Set up plot
    plt.figure(figsize=(6, 6))
    ax = plt.subplot(2, 1, 1)
    # Set up title
    title = coded_label[image_path.split('/')[2]]
    # Plot flower
    img = process_image(image_path)
    imshow(img, ax, title)
    # Make prediction
    probs, labs, outcomes = predict(image_path, model)
    # Plot bar chart
    plt.subplot(2, 1 ,2)
    sns.barplot(x=probs, y=outcomes, color=sns.color_palette()[0])
    plt.show()

plot_prediction('/content/speedbreak.jpg', model)
```

Figure 17:Plot prediction

XI. PREDICTION RESULTS

A focus is made on machines as machines cannot be understood by verbal communication it forms abstractions and concepts [7]. After training the model properly, the trained model are showing the correct predictions.

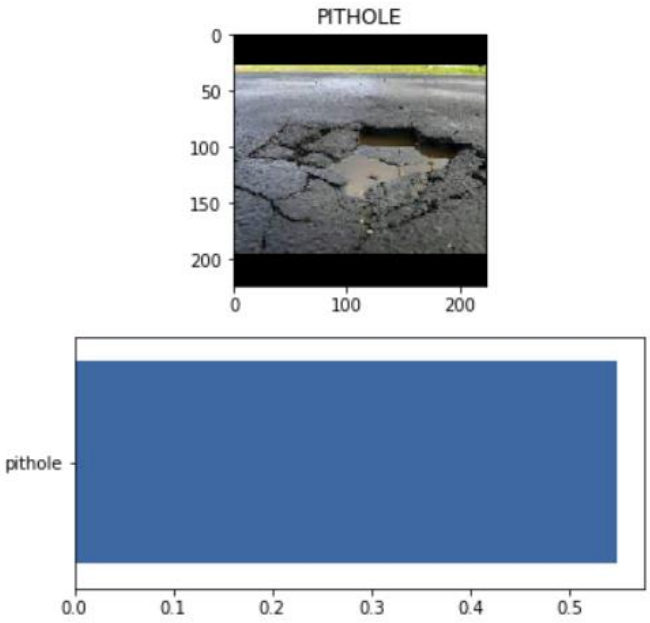

Figure 18: Pithole model

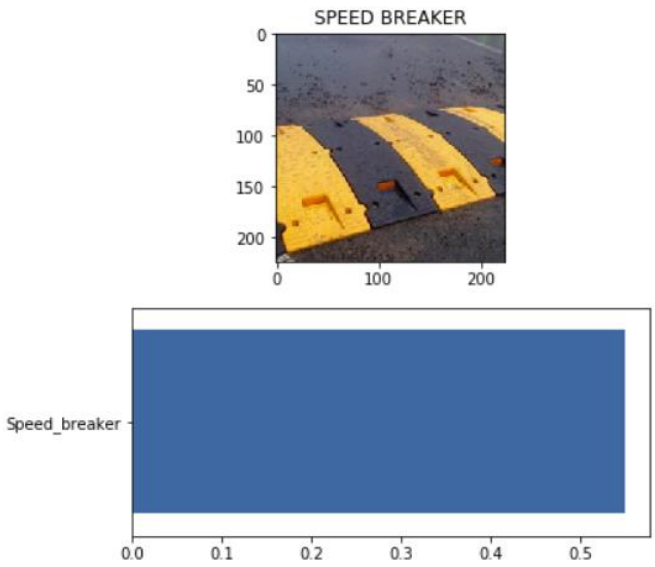

Figure 19:Speed breaker Model

XII. CONCLUSION

In this paper, our main emphasis is on to protect the life of manhood as well as nature through installation of this technology in the cars. Nowadays, people are more tensed and preoccupied by their own life and various activities like smart phones which makes it vulnerable for them to sustain. Drivers are not much attentive on the roads as there can be any potholes, speed breakers or any animals that can cause the accidents. And it endangers the life of animals as people unintentionally kill them without knowing what's in front of their cars. So in this paper we are introducing a new technology through which drivers can easily detect the danger ahead them and save their and animals lives on the road. We have trained our model and achieved the accuracy above 90%.

By concluding this project, we determined to extricate the life of manhood and ecosystem for better future and world in next few generation. So that it will decrease the accidents in the near coming future for better development of the society.

REFERENCES

1. K. L. Ang and J. K. P. Seng, "Application Specific Internet of Things (ASIoTs): Taxonomy, Applications, Use Case and Future Directions," in *IEEE Access*, vol. 7, pp. 56577-56590, 2019. doi: 10.1109/ACCESS.2019.2907793
2. A. Frank, Y. S. Khamis Al Aamri and A. Zayegh, "IoT based Smart Traffic density Control using Image Processing," *2019 4th MEC International Conference on Big Data and Smart City (ICBDSC)*, Muscat, Oman, 2019, pp. 1-4. doi: 10.1109/ICBDSC.2019.8645568
3. Abdelhafiz, D., Yang, C., Ammar, R. et al. BMC Bioinformatics (2019) 20(Suppl 11): 281. https://doi.org/10.1186/s12859-019-2823-4.
4. I. Stančin and A. Jović, "An overview and comparison of free Python libraries for data mining and big data analysis," *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija, Croatia, 2019, pp. 977-982. doi: 10.23919/MIPRO.2019.8757088
5. Nguyen, G., Dlugolinsky, S., Bobák, M. et al. Artif Intell Rev (2019) 52: 77. https://doi.org/10.1007/s10462-018-09679-z
6. Sidey-Gibbons, J. & Sidey-Gibbons, C. BMC Med Res Methodol (2019) 19: 64. https://doi.org/10.1186/s12874-019-0681-4
7. L. Kharb et al (2019) "Brain Emulation Machine Model for Communication" in International Journal of Scientific & Technology Research (IJSTR). pp 1410-1418. http://www.ijstr.org/final-print/aug2019/Brain-Emulation-Machine-Model-For-Communication.pdf