# CLASSIFICATION OF ROAD CRACKS USING DEEP NEURAL NETWORKS

**Project report submitted in partial fulfillment of the Requirements for the Award of the Degree of**

**BACHELOR OF TECHNOLOGY**

**In**

**COMPUTER SCIENCE AND ENGINEERING**

**By**


**A.MEGHANA (178W1A0562)**

**K.T.V.S.SRIDHAR (178W1A0584)**

**M.MANASA (188W5A0518)**

**CH.SAI GREESHMANTH (178W1A0572)**


**Under the Guidance of**


**S.RAJESWARI,M.tech**

**Asst Professor**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**V.R SIDDHARTHA ENGINEERING COLLEGE**

**Autonomous and Approved by AICTE –Twice Accredited by NBA**

**Affiliated to Jawaharlal Nehru Technological University, Kakinada**

**Vijayawada 520 007**


**2021**

**V.R.SIDDHARTHA ENGINEERING COLLEGE**
**(AUTONOMOUS)**
**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

# CERTIFICATE

This is to certify that the MAJOR PROJECT Report entitled **“CLASSIFICATION OF ROAD CRACKS USING DEEP NEURAL NETWORKS”** being submitted by

| | |
|---|---|
| **A.Meghana** | **178W1A0562** |
| **K.T.V.S.Sridhar** | **178W1A0584** |
| **M.Manasa** | **188W5A0518** |
| **Ch. Sai Greeshmanth** | **178W1A0572** |

in partial fulfillment for the award of the Degree Of Bachelor of Technology in Computer science and Engineering to the Jawaharlal Nehru Technological University, Kakinada is a record of bonafide work carried out under my guidance and supervision.

S. Rajeswari, M.Tech
**Asst Professor & Guide**

Dr. D Rajeswara Rao M.Tech, Ph.D
**Professor & Head of Department**

# DECLARATION

We hereby declare that the major project entitled **"CLASSIFICATION OF ROAD CRACKS USING DEEP NEURAL NETWORKS"** submitted for the B.Tech Degree is our original work and the dissertation has not formed the basis for the award of any degree, associateship, fellowship or any other similar titles.

**Place: Vijayawada**
**Date:**

| | |
|---|---|
| **A.MEGHANA** | **(178W1A0562)** |
| **K.T.V.S.SRIDHAR** | **(178W1A0584)** |
| **M.MANASA** | **(188W5A0518)** |
| **CH.SAI GREESHMANTH** | **(178W1A0572)** |

# ACKNOWLEDGEMENT

Behind every achievement lies an unfathomable sea of gratitude to those who activated it, without whom it would ever have come into existence. To them we lay the words of gratitude imprinted with us.

We would like to thank our respected Principal, **Dr. A.V. Ratna Prasad** and **Dr. D. Rajeswara Rao,** Head of the Department, Computer Science and Engineering for their support throughout our Project.

It is our sincere obligation to thank our guide, **S.Rajeswari,** Assistant Professor Department of Computer Science and Engineering, for her timely valuable guidance and suggestions for this Project.

We owe our acknowledgements to an equally long list of people who helped us in this Project. Finally, We wish to thank all the supporting staff who gave us facility in lab for the completion of this Project.

Place: Vijayawada
Date:

**A.MEGHANA (178W1A0562)**
**K.T.V.S.SRIDHAR (178W1A0584)**
**M.MANASA (188W5A0518)**
**CH.SAI GREESHMANTH (178W1A0572)**

# ABSTRACT

Roads play an important role in the economic development of a country. Along with economic development, they even provide social benefits such as traveling with ease and reduce the accidents caused due to poor road maintenance. With the above-said importance of the road, a means of transport, poorly maintained roads may reduce mobility and even lead to a significant increase in road accidents and vehicle operating cost. These cons of poor road maintenance emphasize the need for frequent and quick road damage detection. Road damage detection is an important part of road maintenance, which requires a huge manual effort to get the task done. This work can be done within a minimum time by developing a model that will be able to detect the type of cracks present in the image. In this project, the user is provided with an option to upload the damaged road image which is analyzed against the yolov3 model which can detect whether the road crack is among pothole, alligator crack, longitudinal crack, and lateral crack. This result will be stored in an excel sheet along with the location details of the road crack. Next to this, we will be able to define the condition of the road using the same image as an input to a CNN model. In this way, a supervised deep neural network is trained to detect road damage and save the data attained.

**Keywords:** Road damage, Deep neural networks, Yolov3, CNN.

# TABLE OF CONTENTS

# LIST OF FIGURES

# 1.INTRODUCTION

Frequent road maintenance is important for safe driving and is one among the major decrees of transportation and corresponding maintenance authorities. One important component of maintaining a safer driving environment is to effectively monitor road surface degradation which is labor intensive and requires domain expertise. Also, the maintenance and rehabilitation of road surfaces necessitates having accurate information about the road crack type. It therefore, becomes extremely difficult to monitor road cracks manually and hence, the need to successfully implement computer vision and machine learning algorithms is of paramount importance. There has been growing interest in this field and recently a number of deep learning algorithms have been applied at automatically studying road surface quality. However, the detection and classification of damaged roads mainly rely on humans and high-performance sensors nowadays, which are time-consuming and expensive. It’s urgently needed to find a low-cost method which can detect and classify the damaged areas in images. In our project, we aim on detecting road damage using deep neural networks that would help in road monitoring activity without the need for human intervention for examining the road cracks. The overall procedure of manual inspection of the surface is expensive and extremely time-consuming. Therefore, to ensure constancy, a powerful system to automatically predict road cracks is proposed in this paper. Considering the advantages in recent times on using deep learning for computer vision problems, we implement a model trained on YOLO v3 object detection algorithm and a CNN model that can detect if the type of crack is among pothole, alligator crack, longitudinal crack, or lateral crack and define if the road condition is among good, poor, very poor, satisfactory respectively.

## 1.1 Basic Concepts

Object detection is used in many fields of human life. For example, health and education, etc. Darknet is an open-source library for object detection and segmentation created by the predijje platform and is used with the yolov3 to perform the detection of required classes.

**YOLOv3** – It is latest variant of a popular object detection algorithm yolo- You Only Look Once. Yolo is one of the faster object detection algorithm it means that this network divides the image into regions and predicts bounding boxes and probabilities for each region as in Fig 1.1

Prior detection systems use localizers or classifiers to carry out the detection process. Then the model is applied to an image at different scales and locations. The regions of the image with High scoring are considered for detections. YOLO algorithm uses a completely different approach. The algorithm applies a single neural network to the entire full image. Then this network divides that image into regions which provides the bounding boxes and also predicts probabilities for each region. These generated bounding boxes are weighted by the predicted probabilities.

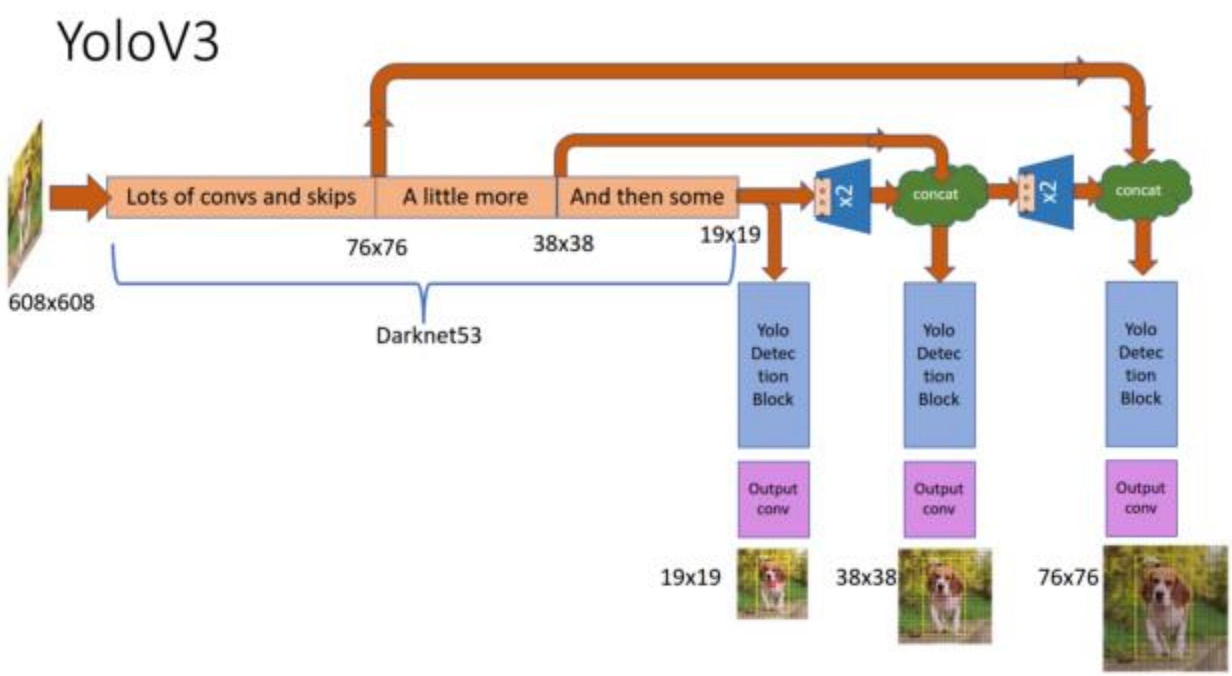


FIG 1.1 Yolov3 Architecture[11]

**FLASK**– Flask is a web development framework developed in Python. It is *easy to* learn and use. Flask is **"**beginner-friendly" because it does not have boilerplate code or dependencies, which can distract from the primary function of an application. This is originated in 2004 when a developer named Armin Ronacher created it as an April Fool's joke. However, it quickly gained popularity in the open-source community anyway. Consequently, it developed into a popular open-source project and gained a massive following, which it maintains today.

Flask provides a development server and a debugger, uses Jinja2 templates and provides integrated support for unit testing. Many extensions are available for Flask, which can be used to enhance its functionalities. Flask is known as a micro-framework (opposite of full-stack frameworks, which also offer additional modules for features such as authentication, database ORM, input validation and sanitization, etc.) because it is lightweight and only provides components that are essential. It only provides the necessary components for web development, such as routing, request handling, sessions, and so on.

For the other functionalities such as data handling, the developer can write a custom module or use an extension. This approach avoids unnecessary boilerplate code, which is not even being used. It consists of two folders static and template in which the images to be displayed within the page are usually placed in the static folder and the html files are placed in the template folder.

Flask-Uploads allows your application to flexibly and efficiently handle file uploading and serving the uploaded files. You can create different sets of uploads - one for document attachments, one for photos, etc. - and the application can be configured to save them all in different places and to generate different URLs for them.

If you're just deploying an application that uses Flask-Uploads, you can customize its behavior extensively from the application's configuration. Check the application's documentation or source code to see how it loads its configuration.

**DATA AUGMENTATION** –Data Augmentation helps us to increase the size of the dataset and introduce variability in the dataset, without actually collecting new data. This helps neural network to treat these images as distinct images anyway and reduce over-fitting.

Amongst the popular deep learning applications, computer vision tasks such as image classification, object detection, and segmentation have been highly successful. Data augmentation can be effectively used to train the DL models in such applications. Some of the simple transformations applied to the image are; geometric transformations such as Flipping, Rotation, Translation, Cropping, Scaling, and color space transformations such as color casting, Varying brightness, and noise injection as shown in fig 1.2

Geometric transformations work well when positional biases are present in the images such as the dataset used for facial recognition. The color space transformation can help address the challenges connected to illumination or lighting in the images.

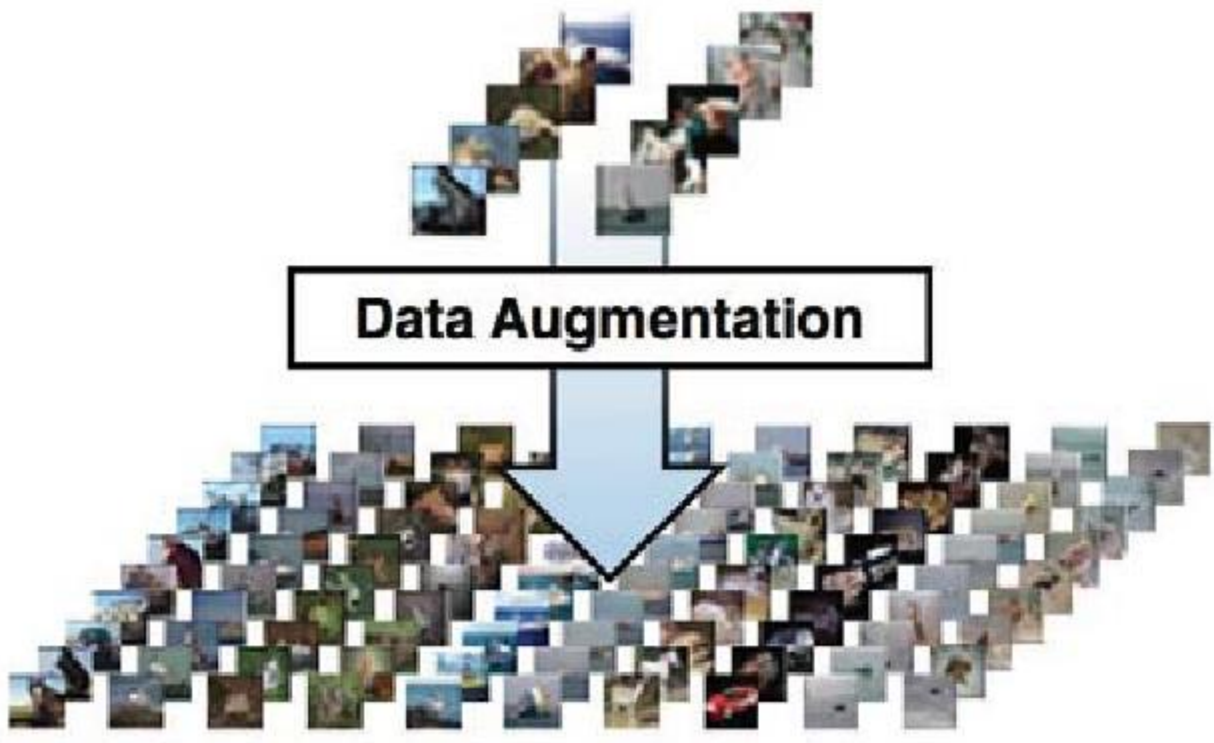


FIG 1.2 Data Augmentation[12]

**POOLING** –Pooling layers are used to reduce the dimensions of the feature maps. Thus, it reduces the number of parameters to learn and the amount of computation performed in the network. The pooling layer summarizes the features present in a region generated by a convolution layer.

The most common pooling layer filter is of size 2x2, which discards three forth of the activations. Role of pooling layer is to reduce the resolution of the feature map but retaining features of the map required for classification through translational and rotational invariants. In addition to spatial invariance robustness, pooling will reduce the computation cost by a great deal.Backpropagation is used for training of pooling operation and it again helps the processor to process things faster.There are many pooling techniques. They are as shown in fig 1.3 and fig 1.4

**Max pooling** where we take largest of the pixel values of a segment.

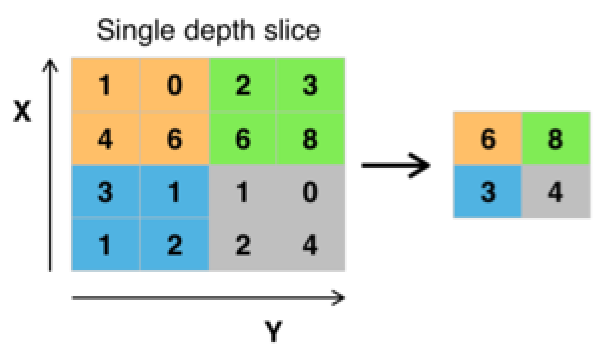


FIG 1.3 Max Pooling[13]

**Average pooling** where we take largest of the pixel values of a segment.

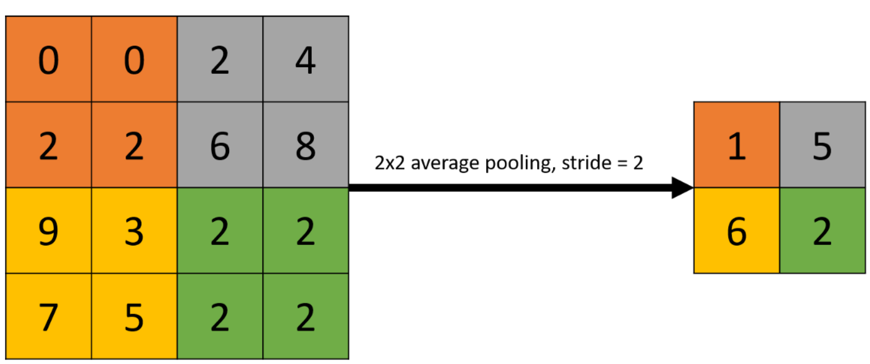


FIG 1.4 Average Pooling[13]

**GEOPY** –Python geopy library is used to obtain the location data in detail using the open street nominatim api and obtain the city, state, municipality and post code data.

## 1.2 Motivation

One of the critical tasks to allow timely repair of road damages quickly and efficiently detect and identify them as road damage detection is an important part of road maintenance. With the above said road damage detection importance we choose the project "Road crack classification using deep neural networks" in which we use CNN model to define the road condition and a yolov3 algorithm to design the model to detect type of road crack.

## 1.3 Problem Statement

Road damage caused due to poor conditions of roads and heavy truck loads this leads to accidents. So to develop a model able to detect the road damage using Deep Neural Networks that is able to specify the type of road crack and a CNN model to define the road condition for better evaluation of state of the road.

## 1.4 Scope

1. When the user uploads damaged road images, model trained against the related dataset will be able to classify the type of damage
2. This dataset consists of images related to different types of cracks and their annotated data saved in text format.
3. Reduces the manual time taken to identify the type of damage occurred with more precision when compared to visual inspection, which also require experienced road managers.
4. CNN model trained is used to define whether the road condition is among good, poor, satisfactory , very poor which helps us in analyzing the road condition even better.
5. Using the python geopy library and nanitam api we are able to get the location data in detail.

## 1.5 Objective

Road damage caused by the extreme weather conditions or any other reasons can be solved by the corresponding department mostly in well-known areas and it is even difficult for them to get to know every damage caused in every area. Instead it would be helpful if we were able to detect damage and classify the image uploaded by the user according to the damage for quick actions and be able to define the condition of the road for quick evaluation.

# 2. LITERATURE SURVEY

## 2.1 Rui Fan; Ming Liu, "Road Damage Detection Based on Unsupervised Disparity Map Segmentation", IEEE Transactions on Intelligent Transportation Systems, 2019

As mentioned in this paper use of unsupervised disparity mapping segmentation technique is seen to detect road damage.To train the model made use of datasets KITTI, Apollo Scape, EISATS with each having good number of images.

**Advantages:**

This method shows the damaged area effectively using segmentation.

**Disadvantages:**

The drawbacks of this method is user need to be using the machine of particular requirements which may not be feasible every time, even this method uses complex functions which are difficult to debug manually incase of any issues.

## 2.2 Anpreet Singh, Shashank Shekhar,"Road Damage Detection And Classification In Smartphone Captured Images Using Mask R-CNN", IEEE international conference on Big Data Cup, 2018

This paper discuss about the design and solution to the Road Damage Detection and Classification. Automatic detection and classification of damage in roads in different weather conditions can be used to improve maintenance and autonomous driving.

**Advantages:**

Uses mask-RCNN algorithm for object detection and instance segmentation of natural images, which perform task in fast manner and effective results.

**Disadvantages:**

NVIDIA GeForce 1080Ti graphic card is used, which is not cost efficient

## 2.3 Deeksha Arya, Hiroya Maeda,Sanjay Kumar Ghosh, Durga Toshniwal, Alexander Mraz, Takehiro Kashiyama ,Yoshihide Sekimoto, “Transfer Learning-based Road Damage Detection for Multiple Countries”, IEEE Big Data (GRDDC), 2020

The main aim is implementation of road damage detection systems in all the countries. Firstly, it assesses usability of the Japanese model for other countries. Secondly, it proposes a large-scale heterogeneous road damage dataset comprising 26620 images collected from multiple countries using smartphones. Thirdly, they propose generalized models capable of detecting and classifying road damages in more than one country.

**Advantages:**

Asserts that training the model with dataset with images from different countries (mixed dataset) will given better results.

**Disadvantages:**

Model discussed in this paper cannot reach its aim to give a model which can be used globally.

## 2.4 Rui Fan ,Mohammud Junaid Bocus ,Yilong Zhu ,Jianhao Jiao ,Li Wang ,Fulong Ma, Shanshan Cheng, Ming Liu,"Road Crack Detection Using Deep Convolutional Neural Network and Adaptive Thresholding " , IEEE Intelligent Vehicles Symposium ,2019

Road damage is detected over here using the adaptive thresholding technique which uses image classification and segmentation**.** The main novelties include a fully connected neural network for image classification and a k-mean clustering based image segmentation algorithm in which images are smoothened using bilateral filtering.

**Advantages**:

Usage of bilateral filtering reduce the noisy pixels preserving the edges between cracks and road surface.

**Disadvantages**:

The drawbacks of this method is that it is computational expensive and, therefore, is not appropriate for real-time applications. The size of neighborhood has to be large enough to cover sufficient foreground and background pixels, otherwise poor threshold is chosen.

## 2.5 Stephen L.H.Lau, Edwin K.P Chong, Xu Yang, Wang, “ Automated Pavement Crack Segmentation Using U-Net Based Convolutional Neural Network”, IEEE Access, 2020

This method made use of one cycle learning schedule based on cyclical learning rates to speed up the convergence.

Introduced the concurrent spatial and channel squeeze and excitation modules in proposed network architecture to increase the performance and use of use of various techniques, including training with progressively increasing image sizes to obtain marginally better performances.

**Advantages**:

Use of SCSE modules lead to the better performance along with the use of various training techniques.

**Disadvantages**:

U-Net style architectures may slow down in the middle layers of deeper models, so there is some risk of the network learning to ignore the layers where abstract features are represented.

# 3. ANALYSIS AND DESIGN

## 3.1 Software Model

We build the model using an Incremental Model which is a process of software development where requirements divided into multiple standalone modules of the software development cycle. In this model, each module goes through the requirements, design, implementation and testing phases. Every subsequent release of the module adds function to the previous release. The process continues until the complete system achieved. The YOLOv3 algorithm used is an efficient and faster object detection method used for identifying the type of road crack in our project. This model is developed based on the requirements that are added in an incremental fashion.

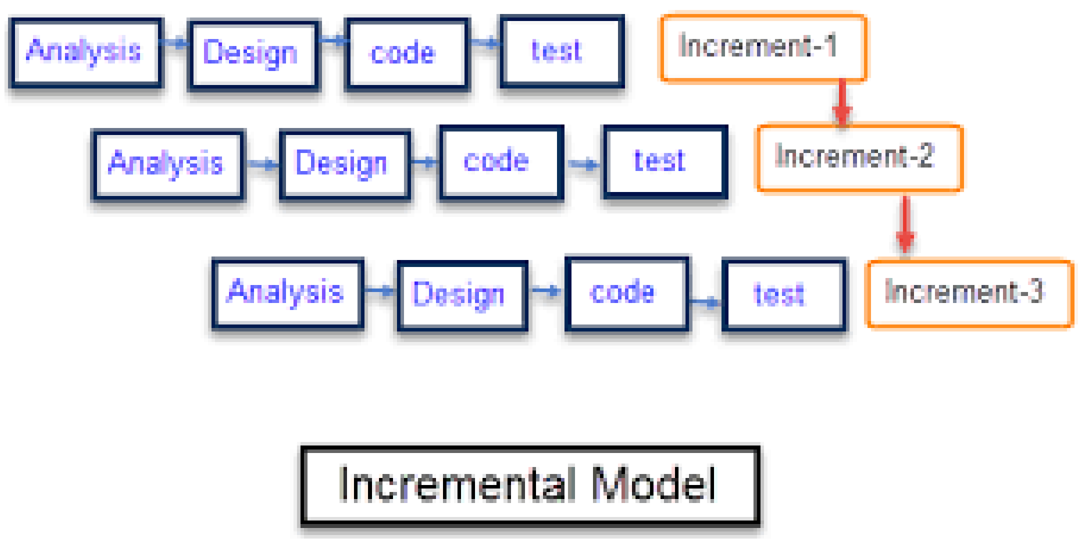


FIG 3.1 Incremental Model[14]

The above figure, fig 3.1 is an incremental model prototype as shown above in diagram in our project we will be able to add any new modules if required to the project as per the requirement. Whenever a new requirement is identified to be related to project that is being added in the corresponding next increment of the project. And once any increment is done the respective process is followed again through the code to get the insights for following process to be done.

## 3.2 Requirements

The following are the software requirements of the project

- Python3
- Google Colab
- Windows 10

Frameworks used,

- Dark net
- Flask (web framework)

Libraries to be used in Python3 are,

- Openpyxl
- Shutil
- Os
- Geopy
- Keras

### 3.3 Use case Diagram

Use case diagrams are usually referred to as behavior diagrams used to describe a set of actions. Each use case provide some observable and valuable result to the system. We are able to case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses.

As the fig 3.2 represents below the actor in this case is user who uploads the image and coordinates details. Here when the user uploads the image that is given to the CNN model and yolov3 model that includes darknet framework and their respective results are stored in a excel sheet that can be viewed by the respective authority.

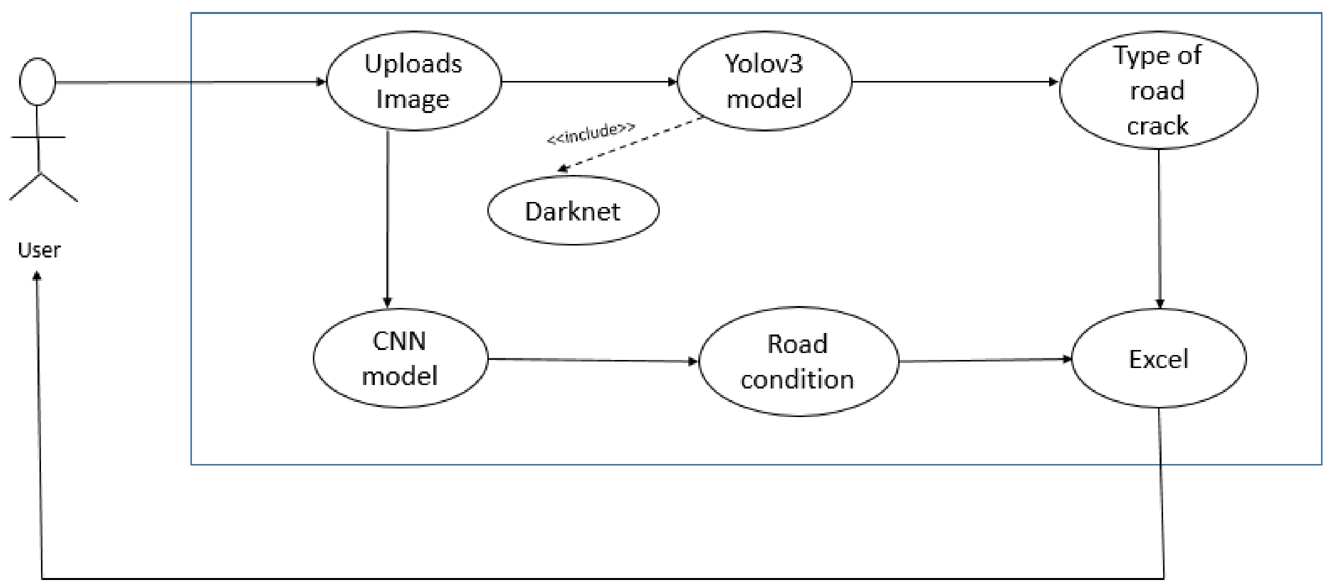


FIG 3.2: Use Case Diagram

## 3.4 Flow Chart

A flowchart is a diagram that depicts a process, system or computer algorithm. They are widely used in multiple fields to document, study, plan, improve and communicate often complex processes in clear, easy-to-understand diagrams. The fig 3.3 below represents the how the image is being evaluated to get the required data to be stored. This image when given as input to yolov3 model is checks if the image consists of any road crack images and if they do then their respective instances count is increased and in other case their count is not altered. In case of CNN model it will be able to detect the road condition. Next to this the data obtained so far is saved in excel for future use.

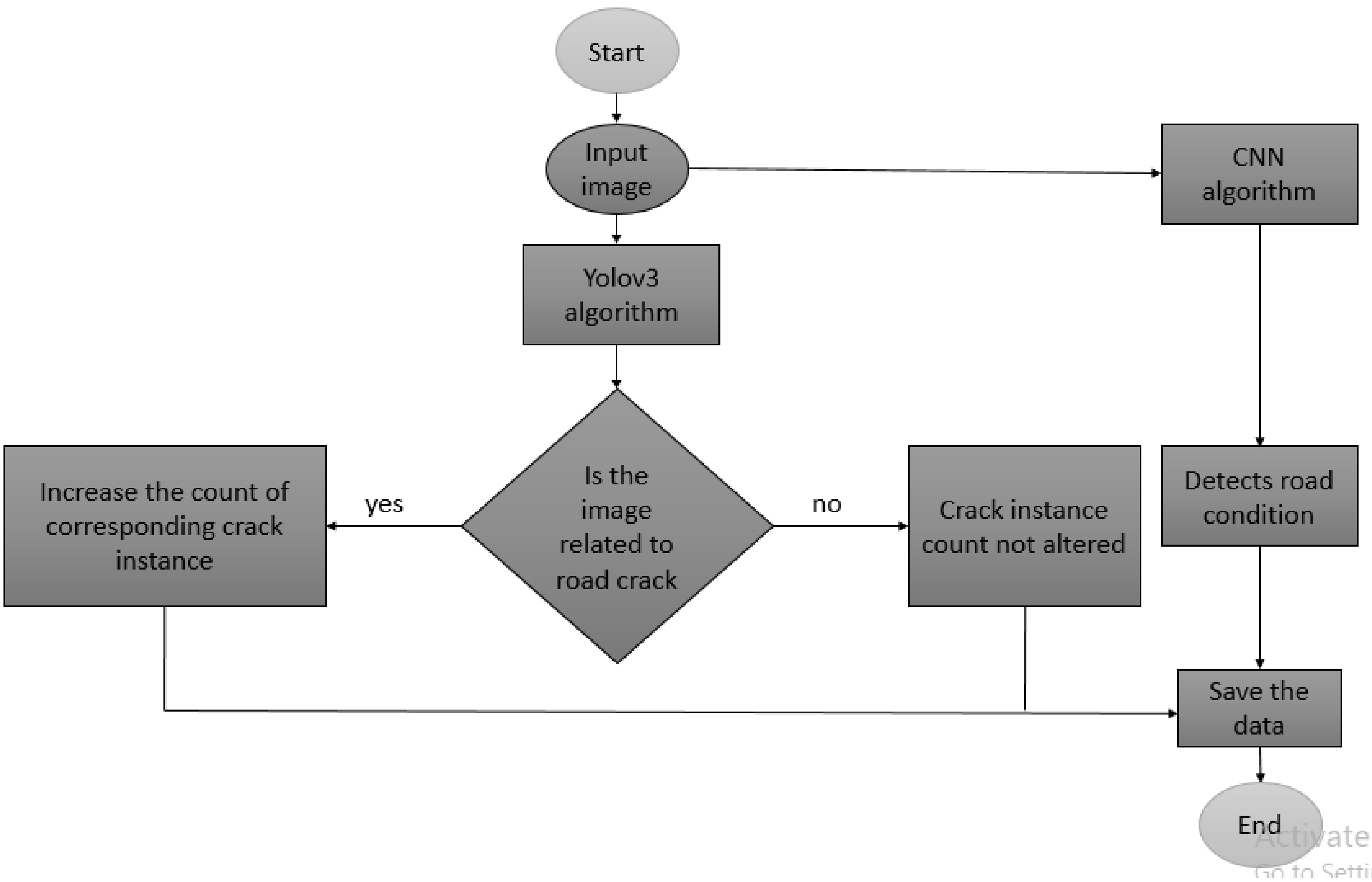


FIG-3.3: Flowchart

# 4. PROPOSED SYSTEM

## 4.1 Process Flow Diagram

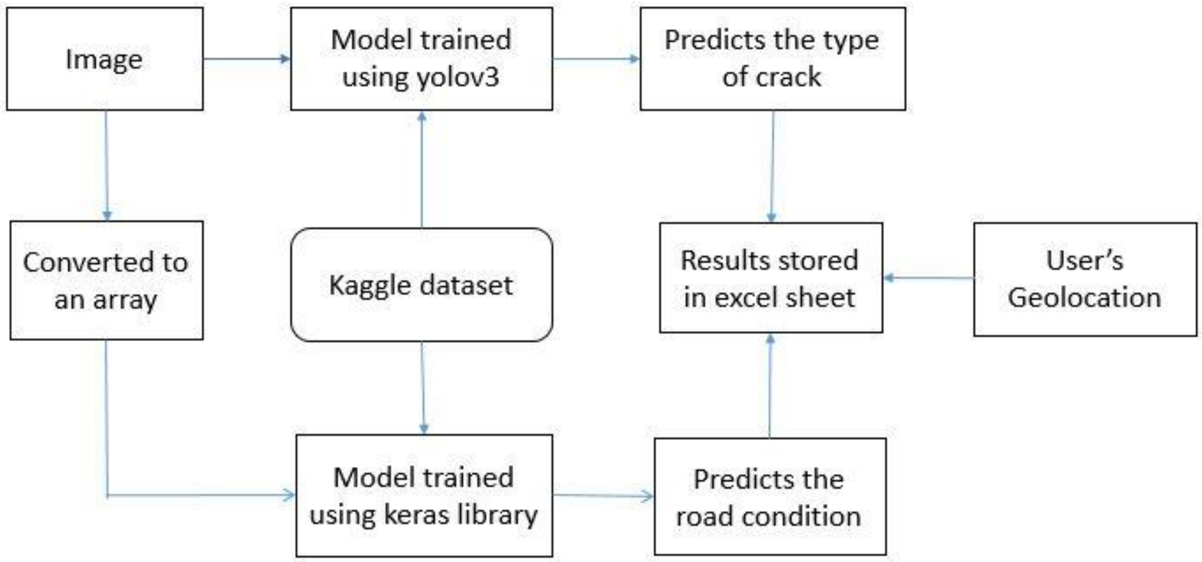


FIG 4.1 Process Flow diagram

The proposed system is a web application made using Html, CSS, javascript. As seen in fig 4.1, the process starts when an image is uploaded that is when the image is given as an input to the model which is trained against a dataset of corresponding images. Yolov3 model trained so can identify whether a type of cracks within the image is among the pothole, alligator crack, longitudinal crack, lateral crack. This data obtained is saved in a text file internally within the ide and using python code we can get the count of the individual crack instances. Next to this, we will try to predict the road condition from the image. This is done through converting the given image to an array as the road condition detecting CNN model is built using the Keras library, which doesn't take an image as input instead consider images converted to an array. Now, this converted array is given as input to the CNN and it will be able to predict whether the road condition is one among the four possibilities, good, poor, satisfactory, very poor. The road condition feature is added to the fact that it helps in prioritizing the work better along with the individual crack count. Using the road condition data along with the crack count helps in delivering more promising and accurate results in taking any action.

This is the process followed on the image uploaded and the user is also asked to enter the location coordinates of the area where damage is noticed, and if the user is not aware of the coordinates they can use the website link provided to get the corresponding location coordinates and we can provide location coordinates using javascript geolocation function and they can store these details for later use. As of above when we obtain the location coordinates we will use the python geopy library to obtain the address in detail as postcode, district, state, and municipality. On completion of all the above-said process, we now save the data obtained into an excel sheet using the python library openpyxl. Here we'll save the data along with the date of upload, name of the image uploaded followed by the count of instances of the corresponding crack in the given image, latitude, longitude, district, municipality, state, postcode, and condition of road obtained through convolutional neural network model. The process above said can be witnessed within the below architectural diagram. As shown in fig 4.2 user will be uploading an image from their gallery from a smartphone or the laptop to the application. This image uploaded will be selected as an input to the yolov3 model and the CNN model. They perform their respective process on the image selected and the output obtained is saved within an excel sheet. Along with the output from the model, location-related data obtained is also saved in the excel sheet and this sheet is accessible to the corresponding official.

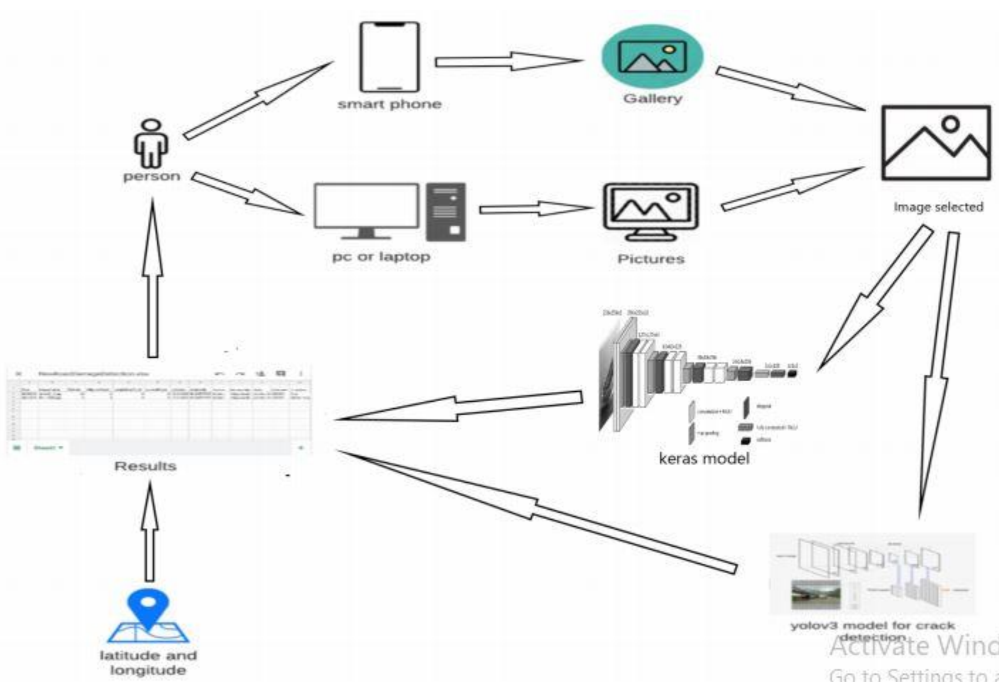


FIG 4.2 Architecture Diagram

## 4.2 Methodology

**Dataset**: We use the standard dataset consists of the labelling in yolov3 annotations format of images of four classes as defined in data collection each of good resolution to develop a yolov3 model

To develop a CNN model we use the standard dataset consists images saved in four folders with their respective images compiling the folder name.

**Model**: A model is built using one of the convolutional algorithm yolov3 which is a faster method of detecting the class type. The dataset considered above is used for training the model with minimum of 8000 iterations and will be able to detect the type of crack in the uploaded image. A CNN model is built to detect the road condition using the data augmentation technique to train the model in best way possible with the dataset we are holding.

**Software used:** We make use of python notebook such as google colab notebook and open source neural network frame work darknet53 that acts as the backbone of the yolov3 model and a keras library in python to develop the convolutional neural network of desired layers.

## 4.3 Data Collection

**Model 1:** Dataset from kaggle is being used as the input to the model and this dataset consists of the images and the annotations of the images in text format that consists of the image id and the coordinates of the respective cracks present in the image as,

- xmin

- ymin

- xmax

- ymax

values that are used for classifying. This dataset consists of images of four types of crack as pothole, alligator crack, longitudinal crack and lateral crack which are of 1080p resolution and of 1920x1080 dimensions. (https://www.kaggle.com/alvarobasily/road-damage) Few of the images from the dataset are


FIG 4.3 Dataset_1 images[15]

The images as shown in fig 4.3 are the road crack detected and these may contain more than one type of the crack in them.

**Model 2:** Dataset from kaggle is being used as the input to the model and this dataset consists of the four folders named as below,

- good

- poor

- satisfactory

- very_poor

These folders consists of the road images justifying the respective folder name. (https://www.kaggle.com/prudhvignv/road-damage-classification-and-assessment) Few of the images from the dataset are,

FIG 4.4 Dataset_2 images[16]

The images in fid 4.4 are representing the good, poor, satisfactory and very poor road conditions respectively in row wise.

## 4.4 Algorithm

Yolov3 algorithm is used for model 1.

**Input** : An image

Step 1: When the image is uploaded the first step performed is Feature Extraction, in which the image will be made into nxn grid and each scale with in that grid results in obtaining the features embedded in different scales.

Step 2: Now these obtained features are sent to a detector

Step 3: The detector uses a open source framework darknet53 which consists of predefined weights.

Step 4: After the classes are detected the class names are added to a file called resultTXT.txt.

Step 5: Based on the class names added we count the number of instance of particular crack

Step 6: Then we add the obtained geolocation and the image data to an excel sheet and save it in drive.

**Output**: Required classes are predicted if any and data is stored in an excel.

CNN algorithm is used for model 2.

**Input** : An image

Step 1: When the image is uploaded the first step performed is resizing the image into 224x224 and converting into an array

Step 2: The step followed by this Feature Extraction, in which the image will be made into nxn grid and each scale with in that grid results in obtaining the features embedded in different scales.

Step 3: This feature extraction is done using the convolutional neural network followed by pooling technique.

Step 4: Models gives the list of scores based on their similarity to the image given as input.

Step 5: Now we consider the index of the list with maximum value and the road condition corresponding to the index obtained so is shown as the final output.

**Output**: Road condition class obtained through above steps is given as output.

# 5. RESULTS AND ANALYSIS

## 5.1 Output Screen  Shots

As mentioned earlier a web application is made that is utilized by the user to upload an image and the procedure to be followed for uploading the image and the location details is as below images represent. The web application is made using the Flask web framework in which the Html file of the home page is placed within the templates folder. The templates folder and the static folder are created if they are not already present using the python os command. All the flask required packages are downloaded using the pip command in python. Flask-uploads package is used as it handles the documents, photos, or any other uploads to the flask server efficiently. In the project, this package is made use as road damage image is uploaded by the user.


FIG 5.1: Image upload

The fig 5.1 is the home page of the application which is also the image upload page. The user is asked to provide latitude and longitude details within the location section. If the user is not aware of the latitude and longitude of the area where road damage is found they can use the 'Obtain Latitude and Longitude' website link where road damage is detected. In the image section as seen a choose file button is provided which when clicked asks us to browse through the files within our smartphone or laptop and select the image in jpg format of the road damage and hit the open button.

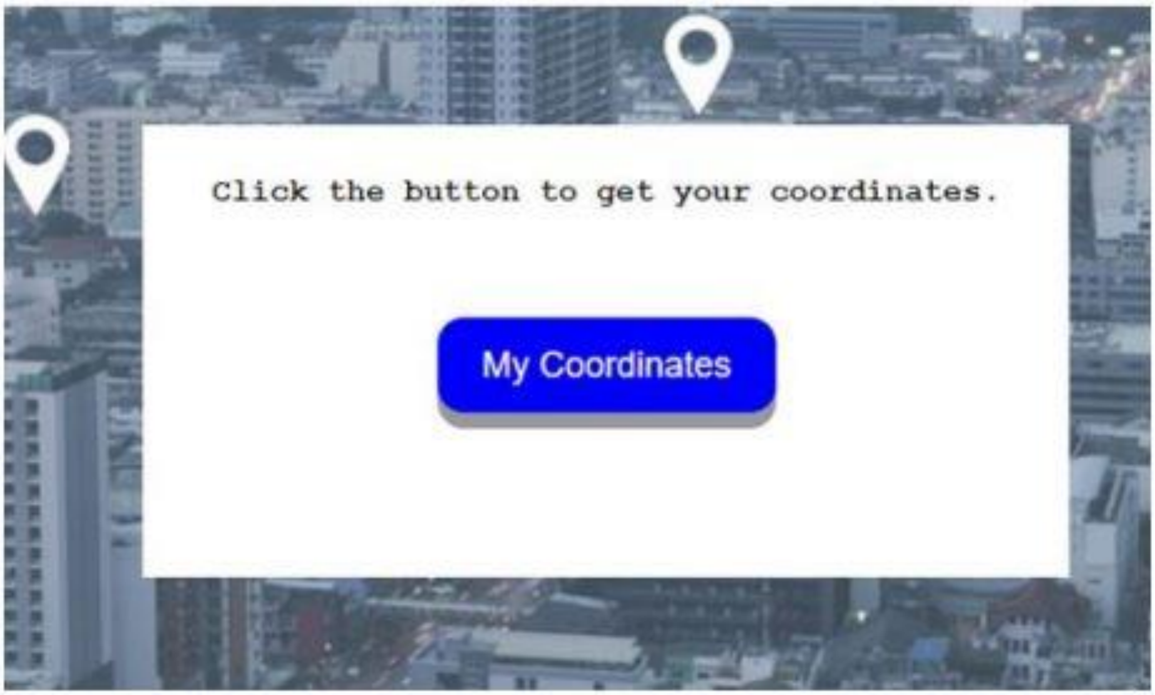


FIG 5.2: Location Identification

When the link in the upload page is accessed location page as shown in fig 5.2 is displayed with a single my coordinates button as above. When this button is hit a location access permission dialog box is shown. User need to select allow option in the dialog box. On selection the block option the location cannot be accessed and corresponding error message is displayed to the user. User can disable the location access granted whenever required in the chrome or the corresponding browser settings page the user is operating on.


FIG 5.3: Coordinates

On selecting the allow option in the dialog box the latitude and longitude options are shown as in fig 5.3. These values can be copied and saved for later use if incase user is not uploaded then and there itself.

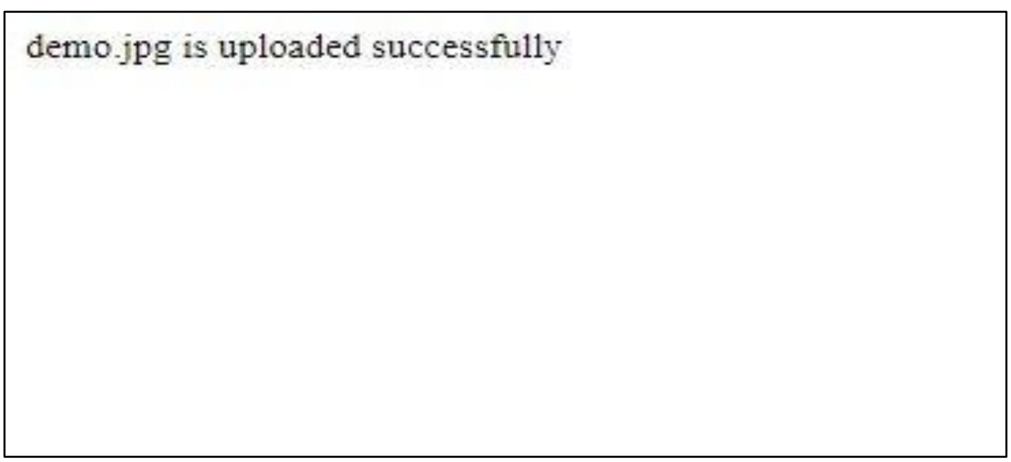


FIG 5.4: Success Page

When user clicks the upload button in the home page the above message is displayed when the image is successfully uploaded to the flask server as shown in fig 5.4.

| A | B | C | D | E | F | G | H | I | J | K | L | M |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Date | ImageName | Pothole | AlligatorCrack | LongitdinalCrack | LateralCrack | Latitude | Longitude | District | Muncipality | State | Postcode | Condition |
| 04/18/21 | demo2_1.jpg | 13 | 2 | 0 | 0 | 16.52485: | 80.6437797! | Krishna | Vijayawada (Urban) | Andhra Pradesh | 520001 | Poor |
| 04/27/21 | BH_9040.jpg | 0 | 3 | 0 | 0 | 16.52485: | 80.6437797! | Krishna | Vijayawada (Urban) | Andhra Pradesh | 520001 | Satisfactory |

FIG 5.5: Data stored in Excel

After all the process of image uploads and coordinates data is done the results are stored within an excel sheet as shown in fig 5.5, for future use. The image uploaded is also given as input to the CNN model to obtain the data regarding the road condition. Using the latitude, longitude data given by the user the location data is obtained in detail for better understanding and to frame a plan of action by the corresponding official.

# 6. CODING

Platform used : Google Colab

Sample Code:

**FLASK**

```
def upload():
    if request.method == 'POST' and 'photo' in request.files:
        global filename
        global latt
        global longg
        filename = photos.save(request.files['photo'])
        message=filename+" is uploaded successfully"
        latt=request.form.get("lat")
        longg=request.form.get("long")
        return message
    return render_template('index.html')
```

**TEST**

```
!./darknet detector test data/obj.data cfg/yolov3-custom.cfg /mydrive/yolov3/backup/yolov3-custom_last.weights  upload.jpg  -thresh 0.006 > resultTXT.txt
```

**DATA AUGMENTATION**

```
datagenerator = ImageDataGenerator(
fill_mode= 'nearest',
horizontal_flip=False,
vertical_flip=False,
shear_range=0.1,
zoom_range = 0.1, # Randomly zoom image
width_shift_range=0.2,  # randomly shift images horizontally (fraction of total width)
height_shift_range=0.2
)
datagenerator.fit(x)
```

**PICKLING**

```
pik=open('data.pickle','wb')
pickle.dump(data,pik)
pik.close()
```

**SAVING THE DATA**

```
test_filename='NewRoadDamageDetection.xlsx' ;
book = load_workbook('/content/gdrive/My Drive/yolov3/'+test_filename)
sheet = book.active ; now = time.strftime("%x")
rows = [[now,filename,cp,ca,clo,cla,latt,longg,district,municipality,state,postcode,condition]]
for row in rows: sheet.append(row)
book.save('/content/gdrive/My Drive/yolov3/NewRoadDamageDetection.xlsx')
```

# 7. Testing

| **Project Name**: Classification of Road Cracks using Deep Neural Networks |
|---|

**Test case id**:2482

**Test Priority**: High

**Module Name**: Upload

**Test Title**: Black box

**Test Executed Date**: 05-07-2021

**Description**: Testing upload page

**Test Designed by**: Team

**Test Designed Date:** 05-07-2021

**Test Executed by**: Team

**Pre Condition**: User should have an image to upload.

| Stage | Test steps | Test Data | Expected Result | Actual Result | status (Pass/Fail) | Remarks |
|---|---|---|---|---|---|---|
| 1 | Choose a file | cracks.jpg | On selecting upload user will be able to see a success page. | On selecting upload user will be able to see a success page. | Pass | Nil |
| 2 | Upload | | | | Pass | Nil |
| 3 | Choose a file | test.pdf | On selecting upload user will be able to see a success page. | An error message is shown on the screen | Fail | Nil |
| 4 | Upload | | | | Fail | Nil |

**Post Condition**: Success page to be displayed.

# 8. CONCLUSION AND FUTURE WORK

Road damage caused by the extreme weather conditions or any other reasons can be solved by the corresponding department mostly in well-known areas and it is even difficult for them to get to know every damage caused in every area. Our project aims to make a model that can detect the type of road crack based on the image uploaded by the user or anyone who encounters the road damage to reduce the time taken due to manual inspection to identify the damage and define the type of crack. With this feature, to be able to define the type of crack and to define the road condition whether it is poor, satisfactory, very poor, and good it will be helpful for the corresponding officials to work on the bad conditioned roads first based on the crack detected and prioritize their work accordingly. Location data obtained in detail is also helpful in prioritizing the work as the officials can know the municipality details and can give the information to them for working on the damage. As a part of future work, we would like to add a language translator so that the details in the websites can be presented based on the language chosen by the user. Even train the model to be able to identify any new cracks possible along with the four types of cracks we can identify now

# 9. REFERENCES

[1] Rui Fan; Ming Liu, “Road Damage Detection Based on Unsupervised Disparity Map Segmentation”, IEEE Transactions on Intelligent Transportation Systems,2019

[2] Anpreet Singh, Shashank Shekhar,”Road Damage Detection And Classification In Smartphone Captured Images Using Mask R-CNN”, IEEE international conference on Big Data Cup,2018.

[3] Deeksha Arya ,Hiroya Maeda,Sanjay Kumar Ghosh,Durga Toshniwal , Alexander Mraz, “Transfer Learning-based Road Damage Detection for Multiple Countries”, IEEE Big Data (GRDDC), 2020

[4] Stephen L. H. Lau,Edwin K. P. Chong,Xu Yang, Xin Wang,“Automated Pavement Crack Segmentation Using U-Net-Based Convolutional Neural Network”, IEEE Access, 2020

[5] Rui Fan ,Mohammud Junaid Bocus ,Yilong Zhu ,Jianhao Jiao ,Li Wang,Fulong Ma, ShanshanCheng, Ming Liu, "Road Crack Detection Using Deep Convolutional Neural Network and Adaptive Thresholding " , IEEE Intelligent Vehicles Symposium ,2019

[6] Teron Nguyen,Bernhard Lechner and Yiik Diew Wong," Response-based methods to measure road surface irregularity: a state-of-the-art review”,Springer,2019.

[7] N.Aboutabit,IPIM Laboratory, Sultan Moulay Slimane University,” Reduced Featured Based Projective Integral for Road Cracks Detection and Classification”,Springer,2020.

[8] Wenjun Wang, Chao Su,” Convolutional Neural Network-Based Pavement Crack Segmentation Using Pyramid Attention Network”, IEEE, 2020.

[9] Redmon J, Divvala S, Girshick R, et al. You Only Look Once: Unified, Real-Time Object Detection[C]// Computer Vision and Pattern Recognition. IEEE, 2016:779-788.

[10] F. Yang, L. Zhang, S. Yu, D. Prokhorov, X. Mei, and H. Ling, ‘‘Feature pyramid and hierarchical boosting network for pavement crack detection,’’ IEEE, 2020.

[11] K. Uri Almog, “Yolov3 Architecture,” Medium.com, Oct, 2020. [Online]. Available: https://towardsdatascience.com/yolo-v3-explained-ff5b850390f, [Accessed Mar. 18, 2020]

[12] Luke Taylor, “Data Augmentation,” Semanticscholor.org, Apr, 2018. [Online]. Available: https://www.semanticscholar.org/paper/Improving-Deep-Learning-with-Generic-Data-Taylor-Nitschke/27c495019bae2d7ab5b607e11a47a39cd9f1c519, [Accessed Feb. 28, 2020]

[13] Pavan Sanagapati, “Pooling,” Kaggle.com, Jun, 2018. [Online]. Available: https://www.kaggle.com/questions-and-answers/59502, [Accessed Feb. 28, 2020]

[14] Li Guo Guru99,“Incremental model,” Guru99.com, Jun, 2019. [Online]. Available: https://www.guru99.com/what-is-incremental-model-in-sdl,[Accessed Oct. 11, 2020],

[15] Alvaro Basily,“Dataset_1 Images,” Kaggle.com, Aug, 2020. [Online]. Available: https://www.kaggle.com/alvarobasily/road-damage,[Accessed Dec.5, 2020]

[16] Prudhvi ,“Dataset_2 Images,” Kaggle.com, Mar, 2021. [Online]. Available: https://www.kaggle.com/prudhvignv/road-damage-classification-and-assessment ,[Accessed Mar.20, 2021]