A Major Project On

Pothole Detection

Submitted in partial fulfilment of the requirements for the award of the

Bachelor of Technology

In

Department of Computer Science and Engineering

By

D Rohit Rajan	**19241A05U9**
Mohammad Khaja Faizan	**19241A05W5**
Vasu Sena Gunda	**19241A05V6**
Rajinikanth kundelu	**19241A05W2**

Under the Esteemed guidance of

Dr. Neha Nandal

Associate Professor

Department of Computer Science and Engineering

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

GOKARAJU RANGARAJU
INSTITUTE OF ENGINEERING AND TECHNOLOGY
(Autonomous)

CERTIFICATE

This is to certify that the major project titled "**Pothole Detection**" is a bonafide work done by D Rohit Rajan (19241A05U9) Mohammad Khaja Faizan (19241A05W5) Vasu Sena Gunda (19241A05V6) Rajinikanth kundelu (19241A05W2) in partial fulfilment for the award of Bachelor of Technology in Computer Science and Engineering of the **GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY**, Hyderabad and that this work has not been submitted for the award of any other Degree/Diploma of any Institution/University.

INTERNAL GUIDE	HEAD OF THE DEPARTMENT
Dr. Neha Nandal	**Dr. K. MADHAVI**
Associate Professor	Professor

EXTERNAL EXAMINER

DECLARATION

We hereby declare that the project titled "**Pothole Detection**" is original and bonafide work of our own in the partial fulfilment of the requirements for the award of Degree of Bachelor of Technology in Computer Science and Engineering, submitted to the Department of Computer Science and Engineering, **GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY (Autonomous),** Hyderabad under the guidance of **Dr. Neha Nandal**, Associate Professor and has not been copied from any earlier reports/works.

D Rohit Rajan	19241A05U9
Mohammad Khaja Faizan	19241A05W5
Vasu Sena Gunda	19241A05V6
Rajinikanth kundelu	19241A05W2

			
D Rohit Rajan (19241A05U9)	**Mohammad Khaja Faizan** (19241A05W5)	**Vasu Sena Gunda** (19241A05V6)	**Rajinikanth kundelu** (19241A05W2)

ACKNOWLEDGEMENT

We are very grateful to our Project guide **Dr.Neha Nandal**, Associate Professor, Department of Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology (Autonomous), for his extensive patience and guidance throughout our project work.

We are very grateful to our Project Coordinator **Ms V. Swapna**, Assistant. Professor, Department of Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology (Autonomous), for extending his support and assisting us throughout our project work.

We would like to thank **Dr.K.Madhavi**, Head of the Department of Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology (Autonomous), for providing the freedom to use all the facilities available in the department, especially the laboratories and library.

We also express deep gratitude to our Director Sir & Principal Sir, for providing us with adequate facilities to pursue our project and providing us with a conducive environment for carrying through our academic schedules and projects with ease.

We sincerely thank all the staff of the Department of Computer Science and Engineering, for their timely suggestions, healthy criticism and motivation during our project work.

Finally, we express our immense gratitude with pleasure to all who have either directly or indirectly contributed to our need at the right time for the development and success of our project work

D Rohit Rajan	19241A05U9
Mohammad Khaja Faizan	19241A05W5
Vasu Sena Gunda	19241A05V6
Rajinikanth kundelu	19241A05W2

ABSTRACT

Potholes are formed due to wear and tear and weathering of roads. They cause not only discomfort to citizens but also deaths due to vehicle accidents. In many developing countries and developed countries, the main problem is road deterioration. In the US nearly 2000 accidents were recorded due to potholes and road damage.

The main aim of this project is to decrease the road accidents happening daily around the world. Many people while driving a car cannot see some of the potholes on the road and if they do not slow down their vehicles, there is a chance of an accident or vehicle damage. Therefore to decrease this kind of accident we came up with a project which recognizes potholes on the roads and alerts the driver by giving a beep sound. To accomplish this objective we used the Yolo algorithm for pothole detection which uses neural networks.

If we can install the cameras on moving vehicles, then we can detect the potholes in real-time and avoid them by alerting the driver. At a very high level, the steps are as follows:

1)Data acquisition and preparation
2)Model training and evaluation
3)Model deployment for real-time detection

LIST OF FIGURES

ACRONYMS

YOLO	You only look once
IEEE	Institute of Electrical and Electronics Engineers
CV	Computer Vision
SVM	Support Vector Machine
FPS	Frames Per Second
RCNN	Regions with convolutional neural networks
CVPR	Conference on Computer Vision and Pattern
KNN	K-nearest neighbours
PIP	Package Installer for Python
RAM	Random Access Memory

TABLE OF CONTENTS

CHAPTER - 1

INTRODUCTION

1.1 Motivation

The strain on the infrastructure has increased as the world's population has grown. Traffic on the roads is at a record high. It has been harder and harder to control this traffic. This is the main driving force behind creating a car with enough intelligence to help the driver in several ways. Road conditions getting worse is one of the issues that are getting worse for the roadways. The road is challenging to drive for various reasons, including rain, traffic accidents, or normal wear and tear. Unexpected obstacles on the road could lead to more collisions. Additionally, the vehicle's fuel usage rises as a result of poor road conditions, wasting valuable gasoline. Due to these factors, learning about such dangerous roads is crucial.

Due to these factors, it is crucial to gather information about these hazardous road conditions and warn the driver about the road condition. We in this project try to design and build such a system that can help drivers find road conditions. This system collects information about potholes and road conditions and sends an alert to the driver.

1.2 Problem statement

The purpose of a pothole detection system is to alert drivers to the uneven potholes and roadways in its route. We examine the various means via which the system's objective can be accomplished. We provide justification for the methods we used in this project. We utilised the following ways to test this model:

- Building a model that will detect the potholes in its path.
- Highlighting the potholes so that the person is alerted.

1.3 Existing System

Experts have experimented with different crater detection methods as a fascinating research field. Here are some suggestions for drill detection techniques:

- ➔ Measure and visualize craters using Microsoft Kinect Sensor
- ➔ Pothole detection study based on 3D project transformation
- ➔ Pothole detection based on SVM of pavement emergency images
- ➔ Detection and sharing of road obstacles by smartphone multimedia sensor analysis
- ➔ Real-time using Android smartphone with accelerometer pothole detection
- ➔ Pothole detection algorithm using an efficient stereo vision system
- ➔ Excavation detection and warning system for autonomous vehicles
- ➔ Detection of potholes in autonomous vehicles

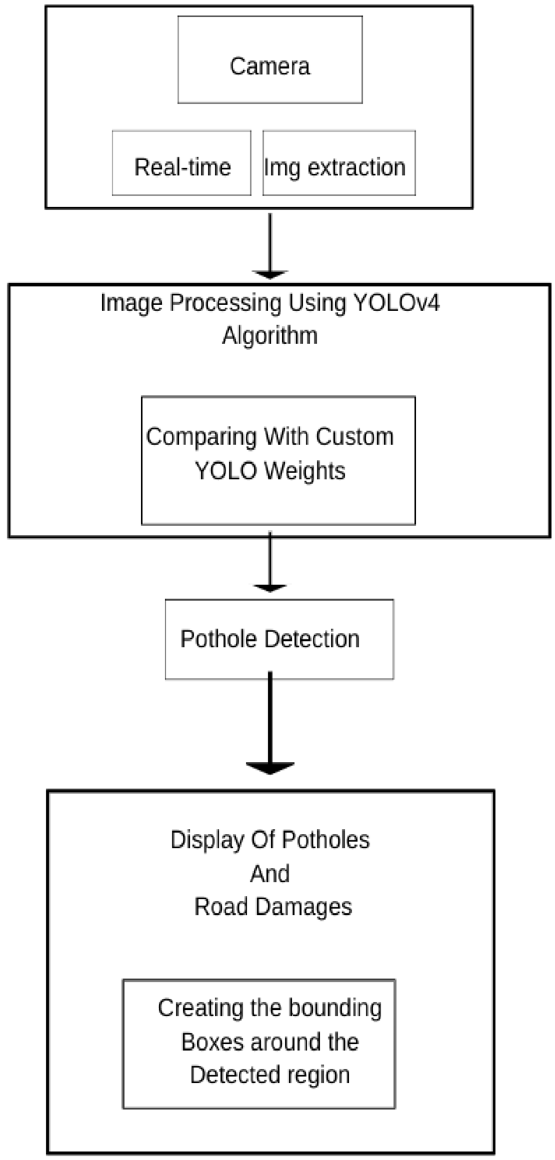
1.4 Proposed System

The main objective of this project is to develop a system that can detect gaps from video inputs. Hole detection can be done in real-time. The YOLO algorithm was used for this. The car's front camera records road video as input to the computer. The computer uses the YOLO algorithm to detect potholes in the video and alert the driver of road conditions.

1.4.1 Advantages over Existing System

- Fast and reliable
- Cost efficient

1.5 Methodology



The proposed system is divided into two phases. Determine and evaluate the size of the pit. The images captured by the camera are fed into the image detection system. The drilling detection unit of the proposed system is based on the YOLO family of object detectors. YOLO is short for "You Only Look Once". As its name suggests it is based on the principle that the objects in the input image can be recognized and recognized at a glance. This technique treats object detection as a regression task.

CHAPTER - 2

LITERATURE STUDY

[1] Deep Learning Based Pothole Detection and Reporting System (IEEE 2020):

- ➔ Using a GPS to pinpoint the location, an accelerometer and ultrasonic sensor were put in the bottom of an automobile and driving at a speed of 25 km/h.
- ➔ The control room is informed of the pothole's location by the microcontroller, which also senses it. The GPS is initialized by the microcontroller (ATmega328), which also gives us the coordinates. Comparative analysis of CNN, KNN (k-Nearest Neighbors), and Kirchhoff's Theory Method was the methodology adopted.

[2] The Modern Pothole Detection using Deep Learning (IEEE 2020):

- ➔ They Have installed a camera on the vehicle, identified the potholes, and marked their positions using an app they developed so that a vehicle without a camera could still get the information about the pothole and provide the driver with the necessary alerts.

[3] Pothole detection and reporting using image processing using a Raspberry Pi microcontroller(IEEE 2018):

- ➔ The Raspberry Pi microcomputer was used to implement the entire system, and 100% of the reports were successful.
- ➔ The web server, Dropbox, and the Internet were used to store and access the reported picture of the pothole and its location.

[4] In the article "Deep Learning," YOLO Neural Network-based learning algorithm for finding potholes in a pavement surface:

- ➔ For photographs of asphalt pavement, the Yolo neural network model was helpful in detecting potholes. T displays a suitable level of detection precision of the application-specific structure, Yolo v3, Yolo v3 Tiny, and Yolo v3 SPP. which have mAP values of 83.43%, 79.33%, and 88.93% correspondingly, and the precision of the area measurement correspondingly, 64.45%, 53.26%, and 72.10%
- ➔ As a result, there is a good chance that it will be developed and used. The Yolo V3 Algorithm was employed.

[5] The article written by E. N. Ukhwah et al is a YOLOv4-based pothole detection system.

- ➔ The YOLOv4 algorithm used in the system this paper proposes allows it to outperform the systems discussed before.
- ➔ A disadvantage of YOLOv3 based on object detection is that the results are less accurate than YOLOv4. YOLOv4 provides more accurate bounding boxes thus resulting in better IoU value.

CHAPTER - 3

SYSTEM SPECIFICATIONS

3.1 Scope of the Project

So many technologies will benefit from our effort. Some of them are that we will utilize our model to test different colours on the same sort of clothes before beginning the real task so that we can decide whether or not to begin. We can also design various sorts of automobiles based on the parameters provided. We can make genuine paintings and turn our imaginations into pictures. All This occurs in a very short period of time, thanks to our flawlessly trained model.

3.2 System Requirement Specification

3.2.1 Functional Requirements:

This is the basic functionality provided by the system requested directly by individual users.

1. To capture the video of the road a good webcam is needed.
2. A speaker to alert the driver.
3. Accurate model capable of detecting potholes on the road.

3.2.2 Non-Functional Requirements

These are the features that the developer adds which are required but not known by the end users like maintenance and security.

1. To build the best model which accurately detects the potholes.
2. Security.
3. Maintainability
4. Reliability
5. Performance
6. Portability
7. Trust

3.2.3 Software Specifications

Python 3.5 or above:

Python is a general-purpose high-level programming language. His design philosophy prioritizes code readability through broad cuts. It is as easy as reading English. It has a practical object-oriented approach designed to help programmers write logically clear code for both small and large projects. Python has a garbage collector and is dynamically typed.

Microsoft visual studio code:

Visual Studio Code is a lightweight IDE. For so many operating systems which run PCs. We can code many languages in VSCode. There are so many library extensions for so many languages and runtime environments.

Packages:

1. Pytorch
2. SciPy
3. NumPy
4. Sklearn
5. Pandas

3.2.4 Hardware Specifications:

1. Laptop/Desktop with webcam
2. CPU: quad-core or above
3. Memory: 8GB or above
4. Operating system: Any OS of 64-bit

3.3 Feasibility Study

Feasibility studies assess the feasibility of a project framework or project. The profitability of a project is tested in a feasibility study to determine the likelihood of success. The analysis aims to highlight issues and challenges that may arise during the project.

3.3.1 Economic Feasibility

We must spend money on webcams because we require them. This is the project's biggest expense. We may also employ cloud services to cut costs and boost scalability. There will be some operating and maintenance costs during the project's life. In addition, we will acquire more.

3.3.2 Operational Feasibility

Every project, as we all know, is maintained after completion. We need a device to fit in front of the vehicle. A fantastic UI/UX that is comfy for the user is necessary. So keeping these things will offer our work positive feedback from users.

3.3.3 Technical Feasibility

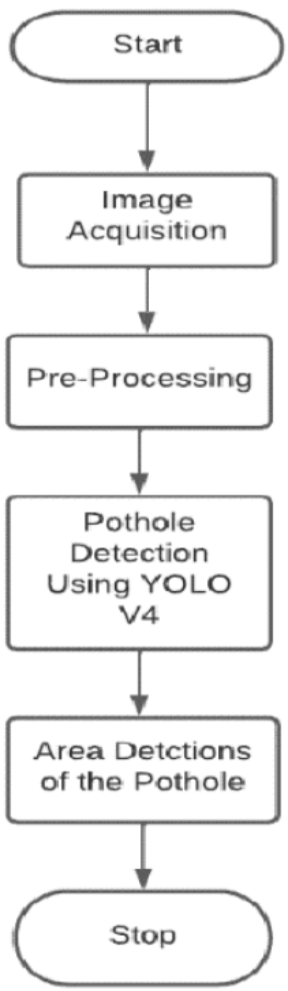
We are creating an Ml model with the YOLO algorithm. This is a publicly available module that we may utilise to simplify our job. Numpy and Scipy modules are also required for mathematical computations. Because these modules will be updated to newer versions in the future, we may need to modify our code if there are deprecated classes or methods. We require an excellent machine-learning team to monitor and work on changes. Because our trained models take a long time to train, updating them will be difficult. So, unless absolutely necessary, we should avoid updating and retraining the models. Working in Anaconda instead of Python will be really beneficial.

CHAPTER - 4

IMPLEMENTATION

4.1 Flow Chart

The system architecture design process identifies the subsystems that make up the system and frameworks for communication and coordination subsystems. The goal of architectural design is to define the overall structure of a software system.



Flowchart Of the Model

4.2 Algorithm Used (YOLO v4) You Only Look Once:

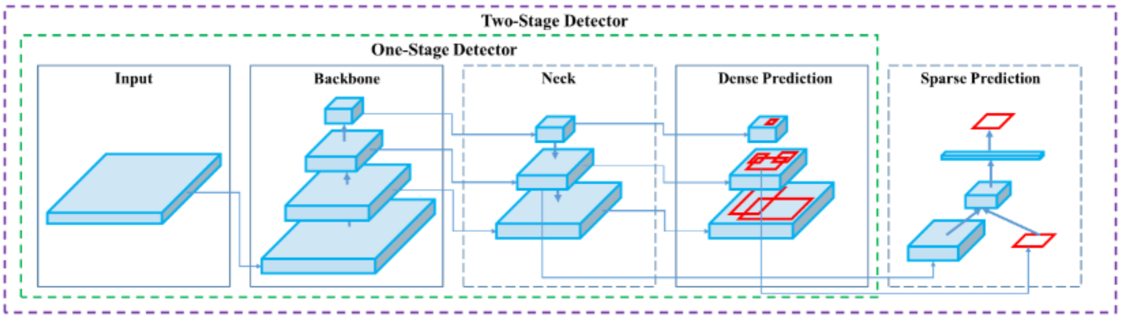
YOLOv4 is SOTA (state-of-the-art) model for real-time object detection. It was published in April 2020 by Alexey Bochkovsky and it is the fourth version of YOLO. It achieves SOTA performance on the COCO dataset consisting of 80 different object classes. YOLO is a single-stage detector.

1) The one-step method is one of the two main modern methods used for object detection tasks where speed of completion is important.
2) In the one-step detector model, the ROI (Region of Interest) is removed and classes and bounding boxes are predicted for the entire image.
3) So this makes them faster than two-stage detectors.
4) Another example is FCOS RetinaNet and SSD. The first version of YOLO was written in the Darknet Framework (which is a high-performance open-source framework for implementing neural networks written in C and CUDA).
5) DarkNet is usually the main network.
6) It divides the object detection task into a regression task followed by a classification task.
7) Regression predicts the class and bounding box for the entire image in one direction and helps identify the positions of objects.
8) Classification defines the class of objects.

Algorithm Overview:

The architecture consists of different parts, in general,

- The input comes first basically the set of training images we have which is fed to the network
- It is processed in a parallel cluster of GPUs. Next comes the spine and neck where feature extraction and assembly takes place.
- Neck detection and head detection together can be called object detection and finally the detection/prediction is done by the head.
- Basically, the head is responsible for identification (localization and classification).



Input: { Image, Patches, Image Pyramid, … }

Backbone: { VGG16 [68], ResNet-50 [26], ResNeXt-101 [86], Darknet53 [63], … }

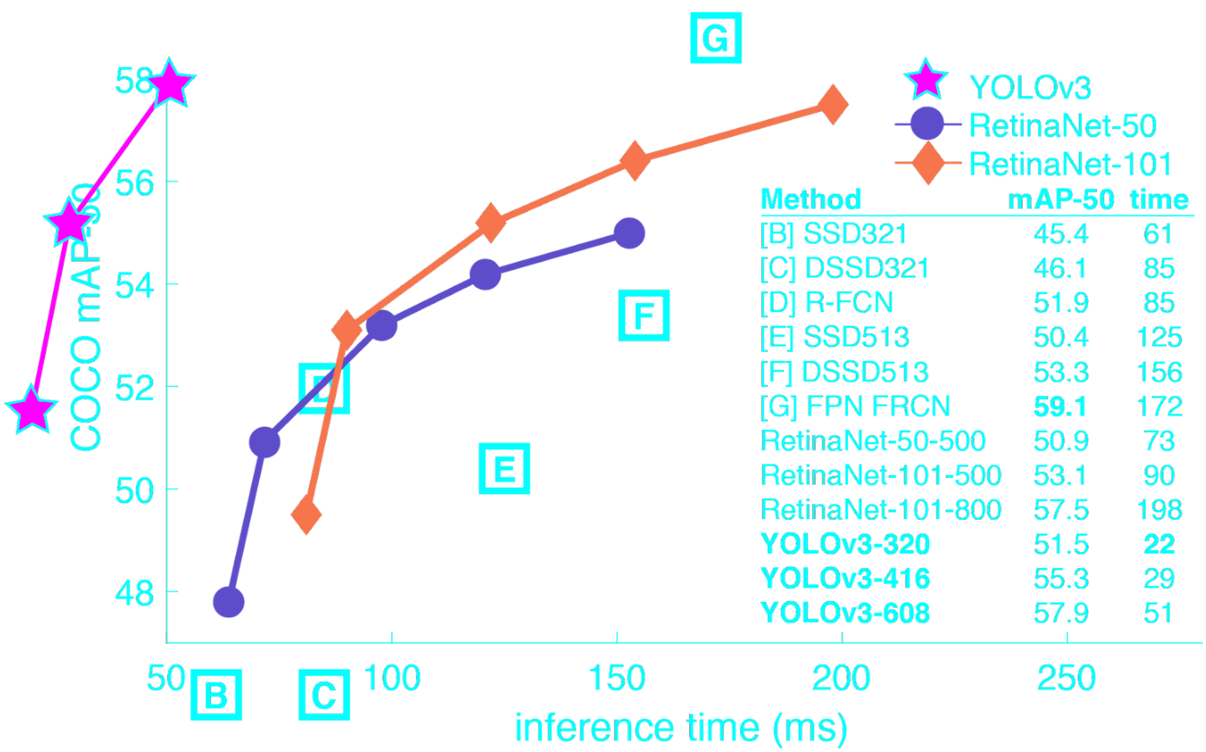
Neck: { FPN [44], PANet [49], Bi-FPN [77], … }

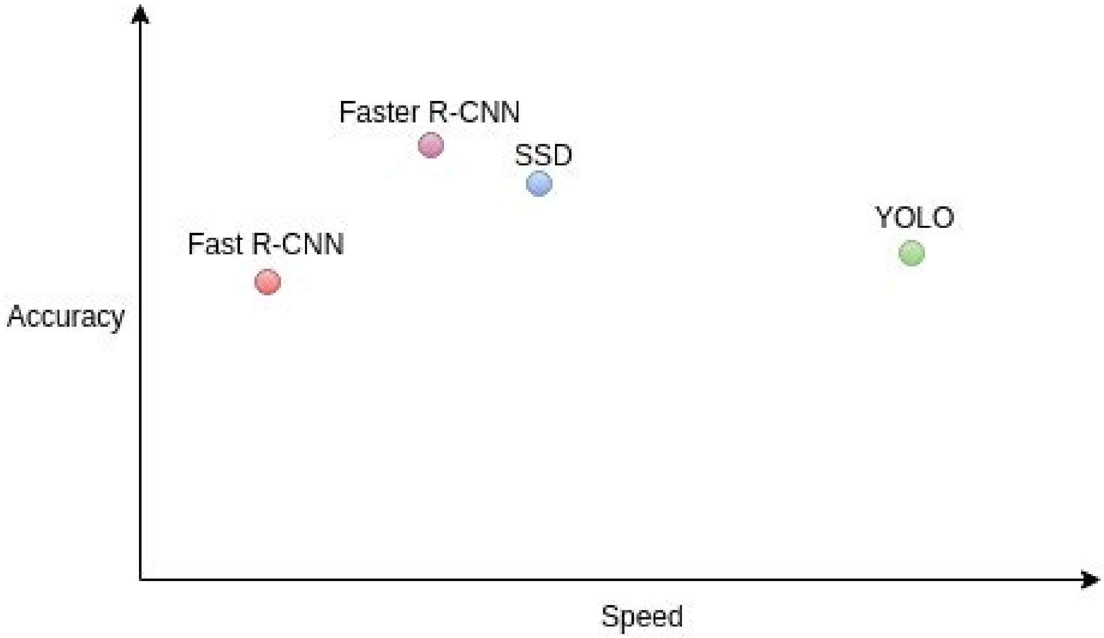
Head:

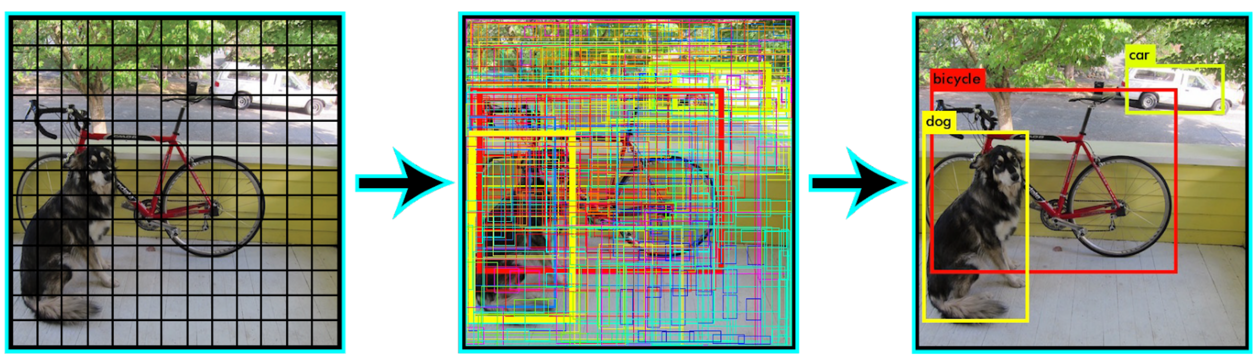
Dense Prediction: { RPN [64], YOLO [61, 62, 63], SSD [50], RetinaNet [45], FCOS [78], … }

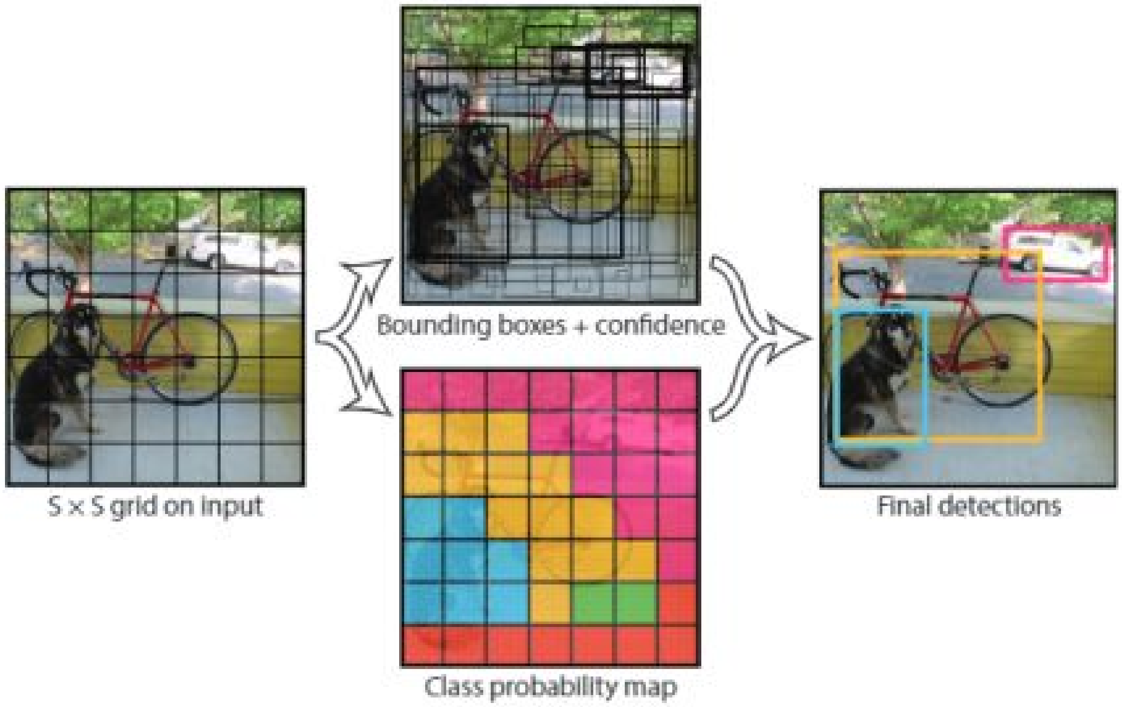
Sparse Prediction: { Faster R-CNN [64], R-FCN [9], … }

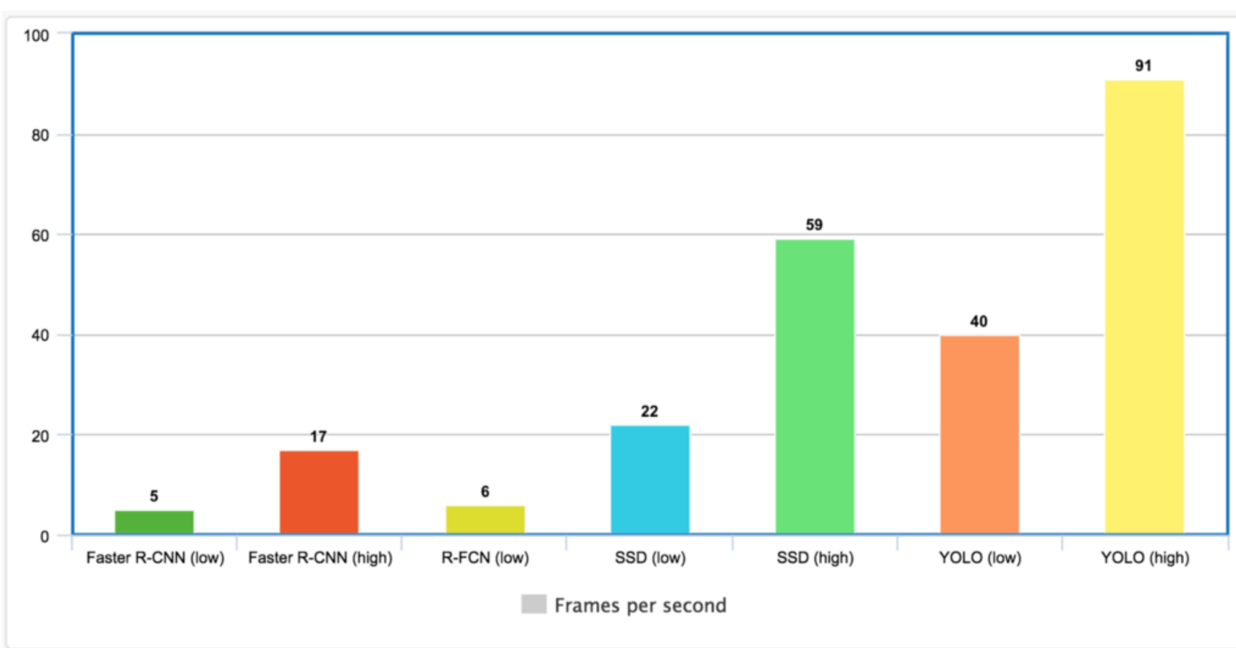
Figure 2: Object detector.

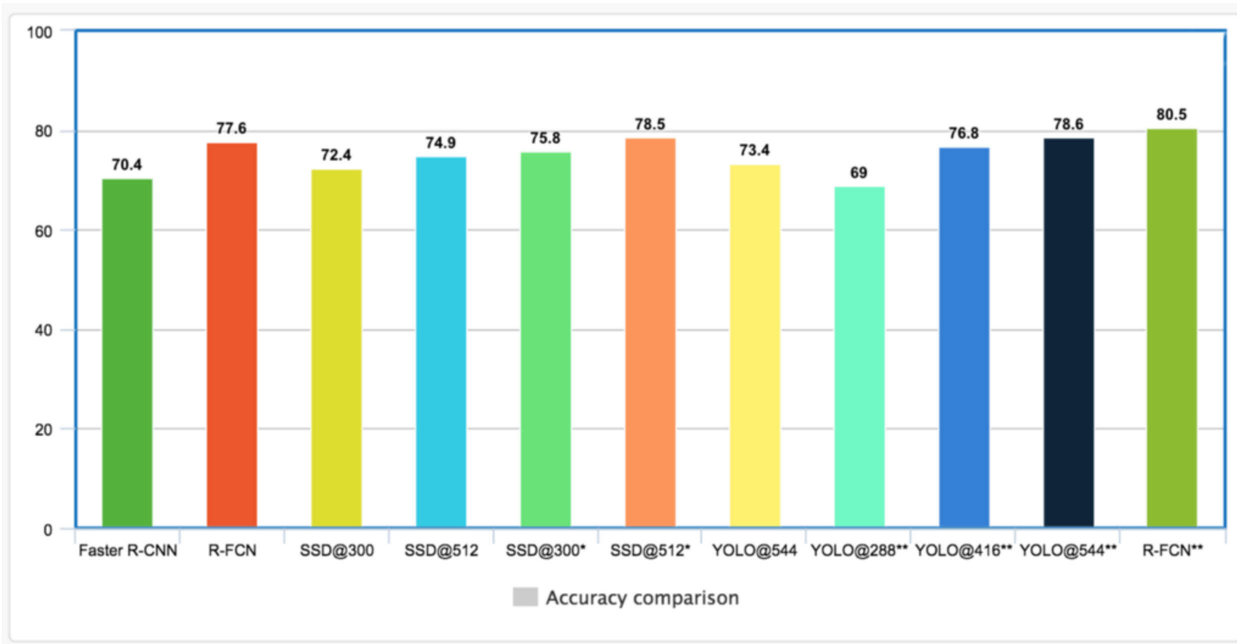



Faster R-CNN
SSD
YOLO
Fast R-CNN
Accuracy
Speed


car
bicycle
dog


S × S grid on input
Bounding boxes + confidence
Class probability map
Final detections


100
80
60
40
20
0
5
17
6
22
59
40
91
Faster R-CNN (low)
Faster R-CNN (high)
R-FCN (low)
SSD (low)
SSD (high)
YOLO (low)
YOLO (high)
Frames per second



4.3 Implementation

The project implementation is done using Python and the following Python concepts. They are as follows

4.3.1 NumPy

NumPy is used to deal with arrays. The name itself says that it is a library related to numbers. Most of the Linear Algebra is implemented in NumPy.

4.3.2 SciPy

SciPy is one of the most versatile libraries in Python. SciPy Won't run without Numpy. So many things like optimization, and statistics are achieved by It.

4.3.3 Pip

It is a package manager in Python which helps to manage, install and update the required Python packages and dependencies.

4.3.4 Sklearn

Scikit-learn is a Python machine-learning library. It has powerful machine learning and statistical modelling tools such as cluster regression classification and dimensionality reduction. This library is primarily written in Python. Scikit-learn is based on NumPy SciPy and Matplotlib.

4.3.4 CV2

This package is very useful in working with videos in Python. It is only compatible with Python and is often used to create ML models. This package helps with OpenCV by providing functions for creating Color Masks and images, video feeds, and resizing and rotating them.

4.3.5 Set Up

To make the environment ready for implementation, it is essential to perform the following steps:-

Step-1: Let us start installing the python from the official website to the current OS (https://www.python.org/getit/)

Step-2: Install pip using the command (C:> py -m ensurepip --upgrade) in windows

Step-3: Install the required modules like cv2, NumPy, pandas using **pip install x,** where x is the required module

Step-4: Ensure the Yolo.cfg and Yolo.weights file in the right directory from here (https://pjreddie.com/darknet/yolo/)

Step-5: Open the python Script in the VScode and run the program

4.4 Test Cases Scenarios:

The following are the few test case scenarios that we have come across during our testing phase, to check the functionality of code logic that has been implemented. The test case is developed in such a way that it traverses each and every sub-module for performance and accuracy testing. The below table has the implementations of the main test case scenarios.

Test Case ID	Test case Scenario	Expected Results	Actual Results	Pass/Fail
T1	Given a plain road image without any pothole	Image without any bounding boxes	Got the image without any bounding boxes	Pass
T2	Given an image with pothole	Image with the bounding boxes around the pothole	Got the image with bounding boxes	Pass
T3	Given the road image with linear crack	Image with a bounding box around the crack	Got the image with bounding box around the crack	Pass
T4	Given the image with alligator crack	Image with the bounding box around the alligator crack	Image with a bounding box around the crack	Pass
T5	Given an image with semi water-filled pothole	Image with bounding boxes around the pothole with water	Image with the bounding boxes around the semi water-filled pothole	Pass

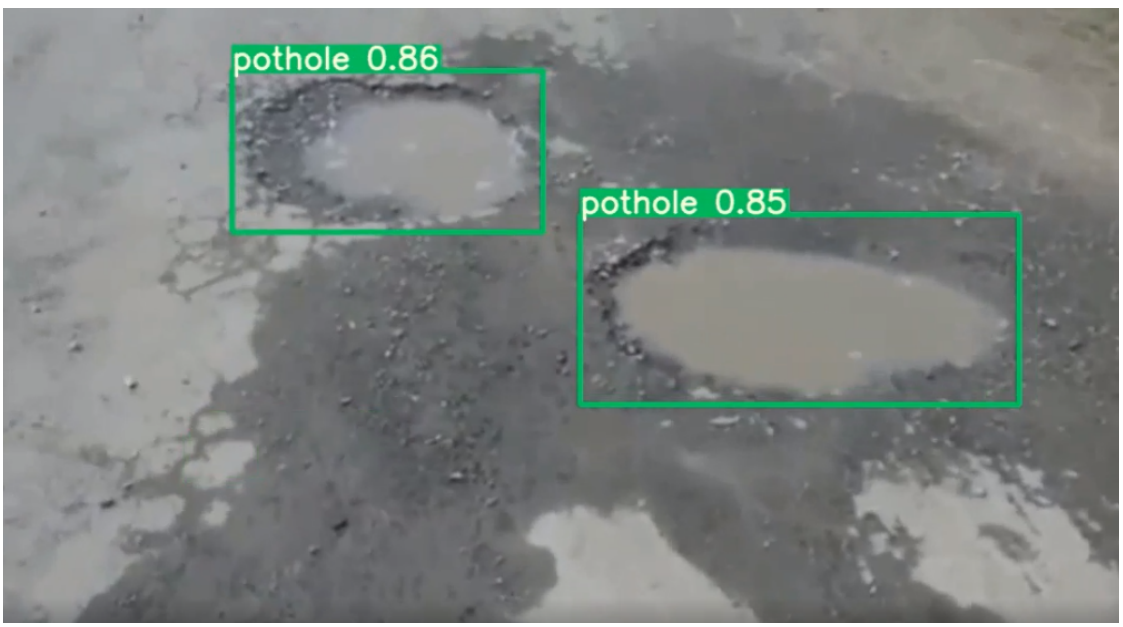
4.5 Code of our work

```
import cv2
import numpy as np

net = cv2.dnn.readNetFromDarknet('/home/vasusena/yolo.cfg' , '/home/vasusena/yolo.weights')
classes = []

with open('/home/vasusena/obj.names','r') as f:
    classes = f.read().splitlines()

cap = cv2.VideoCapture('/home/vasusena/pothole.mp4')

while True:
    _,img = cap.read()
    height, width, _ = img.shape
    blob = cv2.dnn.blobFromImage(img, 1 / 255, (416, 416), (0, 0, 0), swapRB=True, crop=False)
    net.setInput(blob)
    output_layers_names = net.getUnconnectedOutLayersNames()
    layerOutPuts = net.forward(output_layers_names)
```

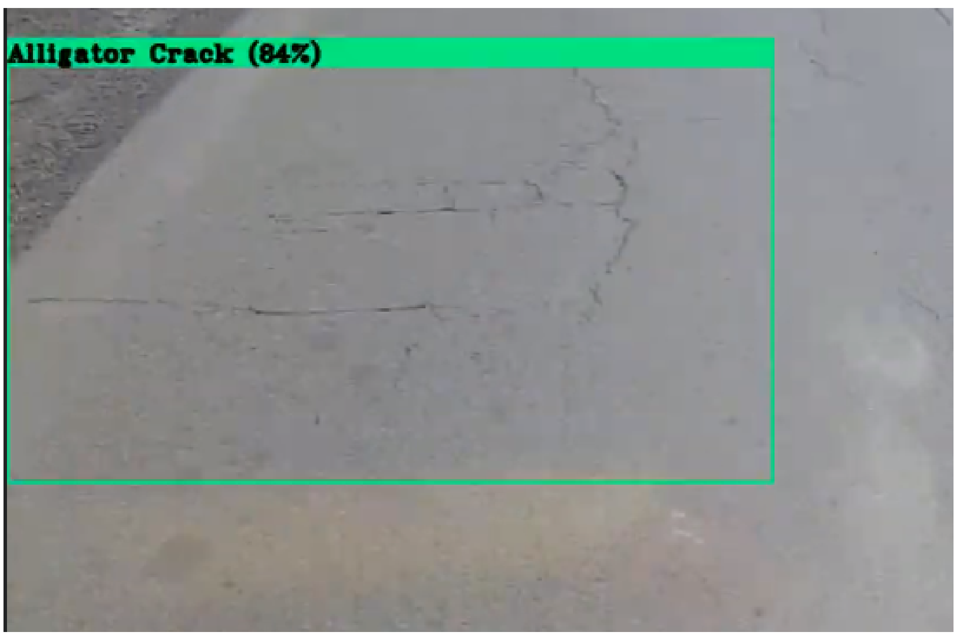
```
boxes = []
confidences = []
class_ids = []
for output in layerOutPuts:
    for detection in output:
        scores = detection[5:]
        class_id = np.argmax(scores)
        confidence = scores[class_id]
        if confidence > 0.5:
            center_x = int(detection[0] * width)
            center_y = int(detection[1] * height)
            w = int(detection[2] * width)
            h = int(detection[3] * height)

            x = int(center_x - w / 2)
            y = int(center_y - h / 2)

            boxes.append([x, y, w, h])
            confidences.append(float(confidence))
            class_ids.append(class_id)

    print(len(boxes))
```

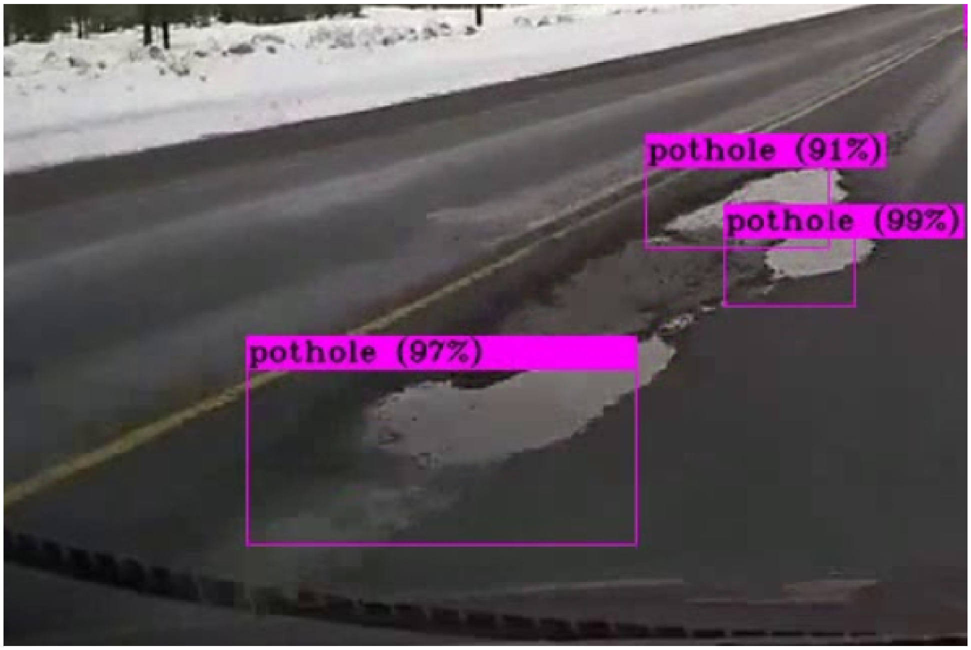
```
    indexes = cv2.dnn.NMSBoxes(boxes, confidences, 0.5, 0.4)
    font = cv2.FONT_HERSHEY_PLAIN
    colors = np.random.uniform(0, 255, size=(len(boxes), 3))

    if len(indexes) > 0:
        for i in indexes.flatten():
            x, y, w, h = boxes[i]
            label = str(classes[class_ids[i]])
            confidence = str(round(confidences[i], 2))
            color = colors[i]
            cv2.rectangle(img, (x, y), (x + w, y + h), color, 2)
            cv2.putText(img, label + " " + confidence, (x, y + 20), font, 2, (255, 255, 255), 2)

    cv2.imshow('image', img)
    key = cv2.waitKey(100)
    if key == 27:
        break

cap.release()
cv2.destroyAllWindows()
```

4.6 OutPut Of Our Work

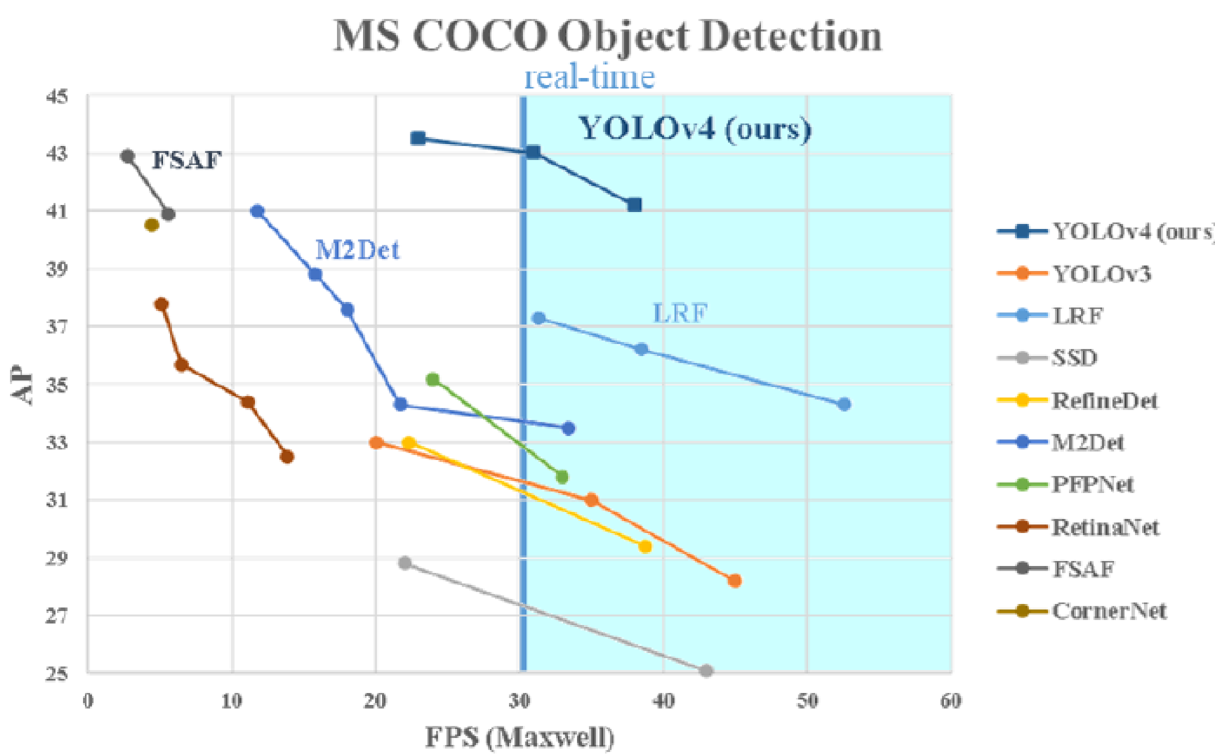


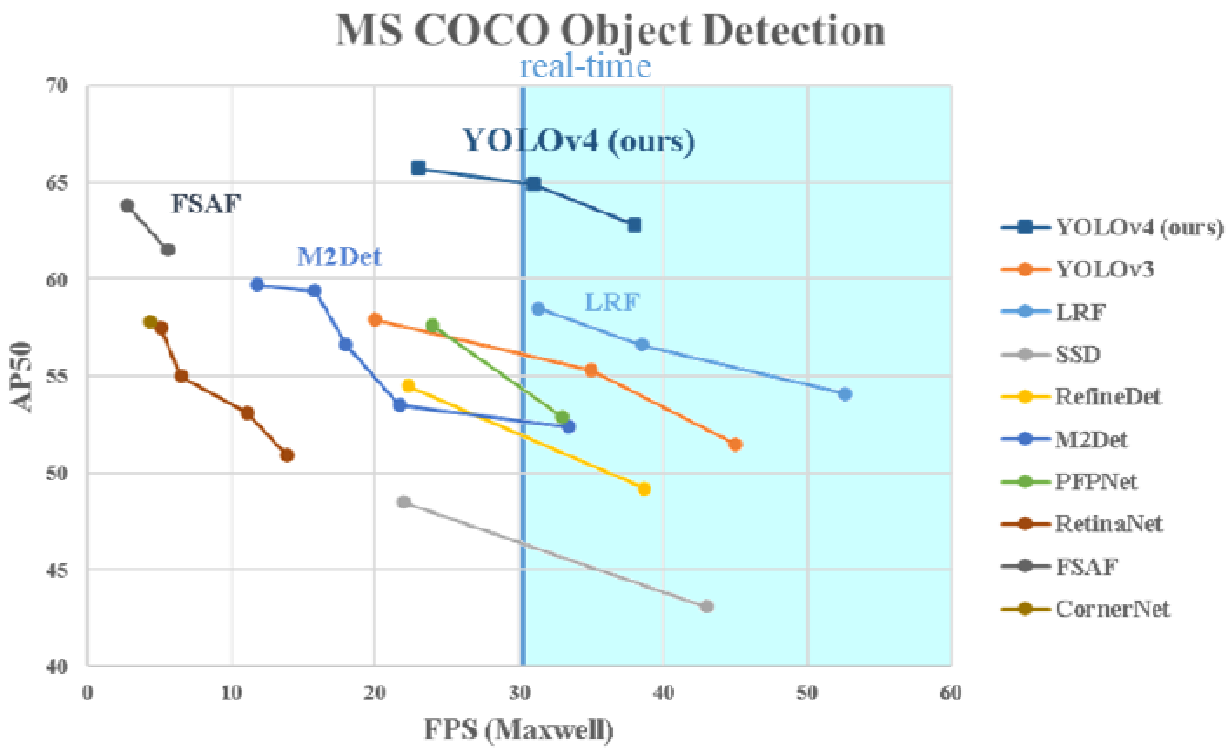

Linear Crack (42%)

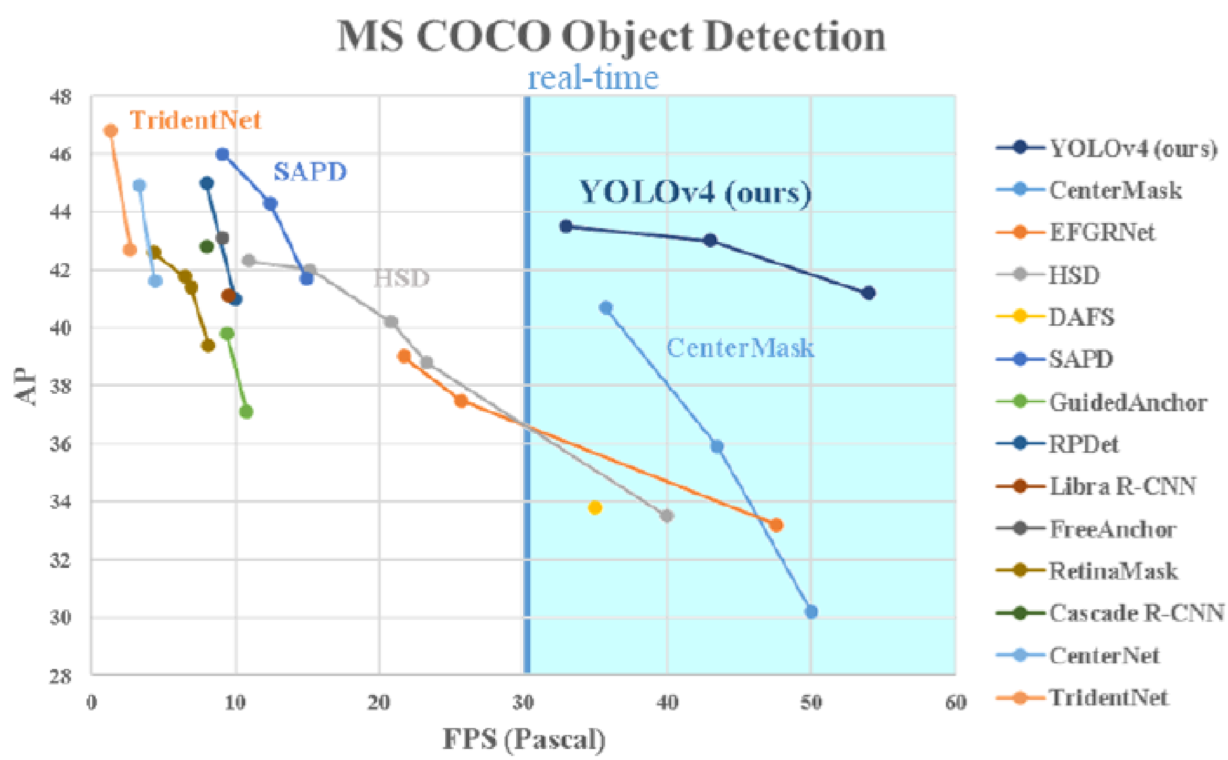

Alligator Crack (84%)

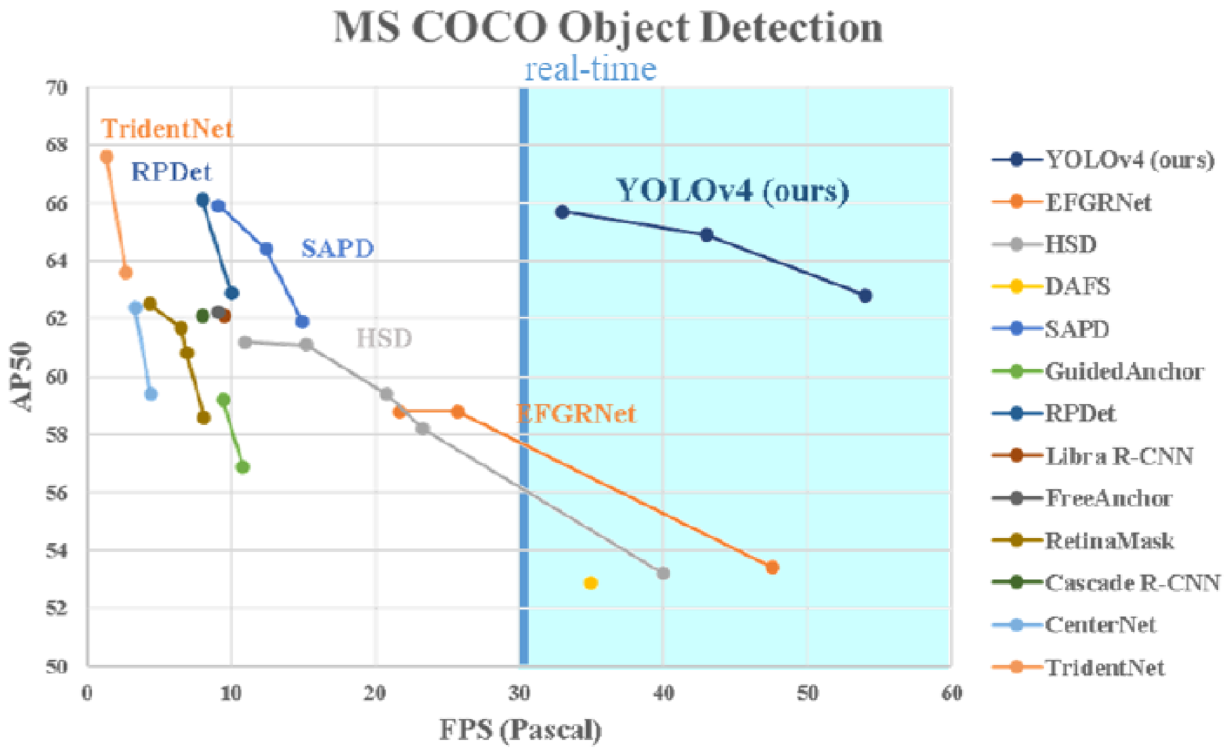

pothole (91%)
pothole (99%)
pothole (97%)

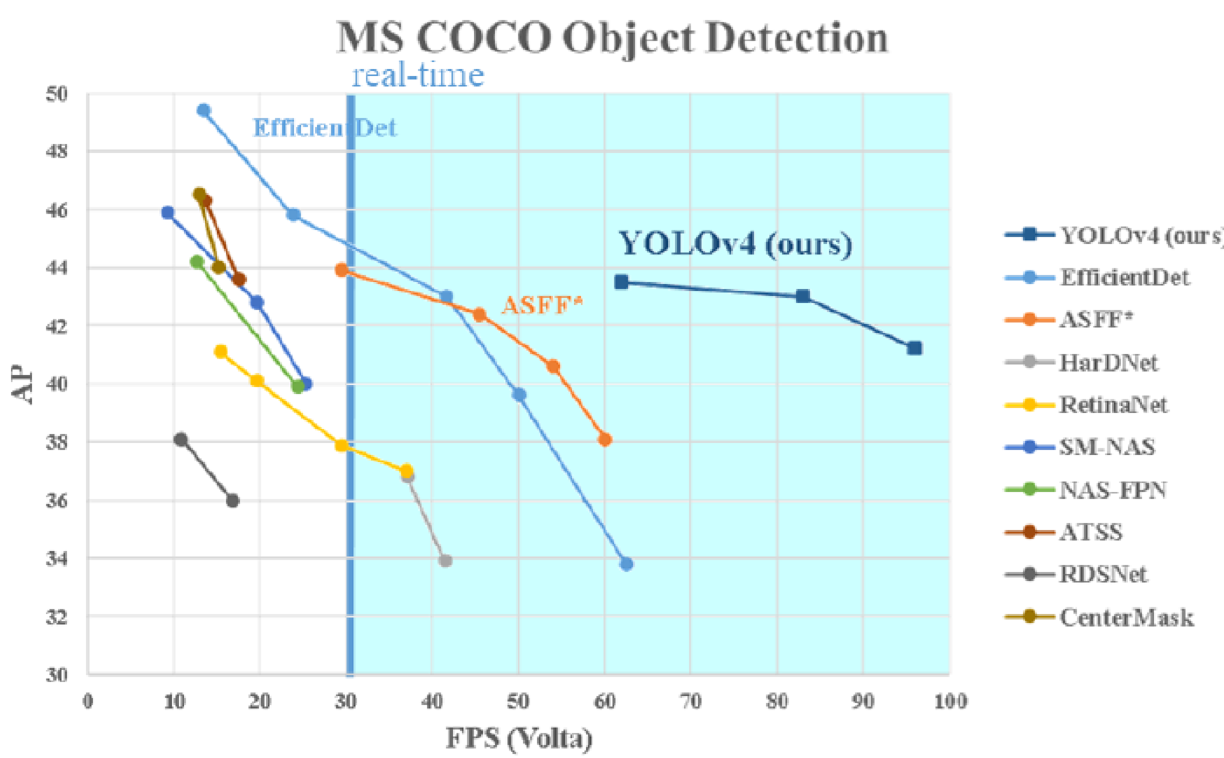
4.7 Result

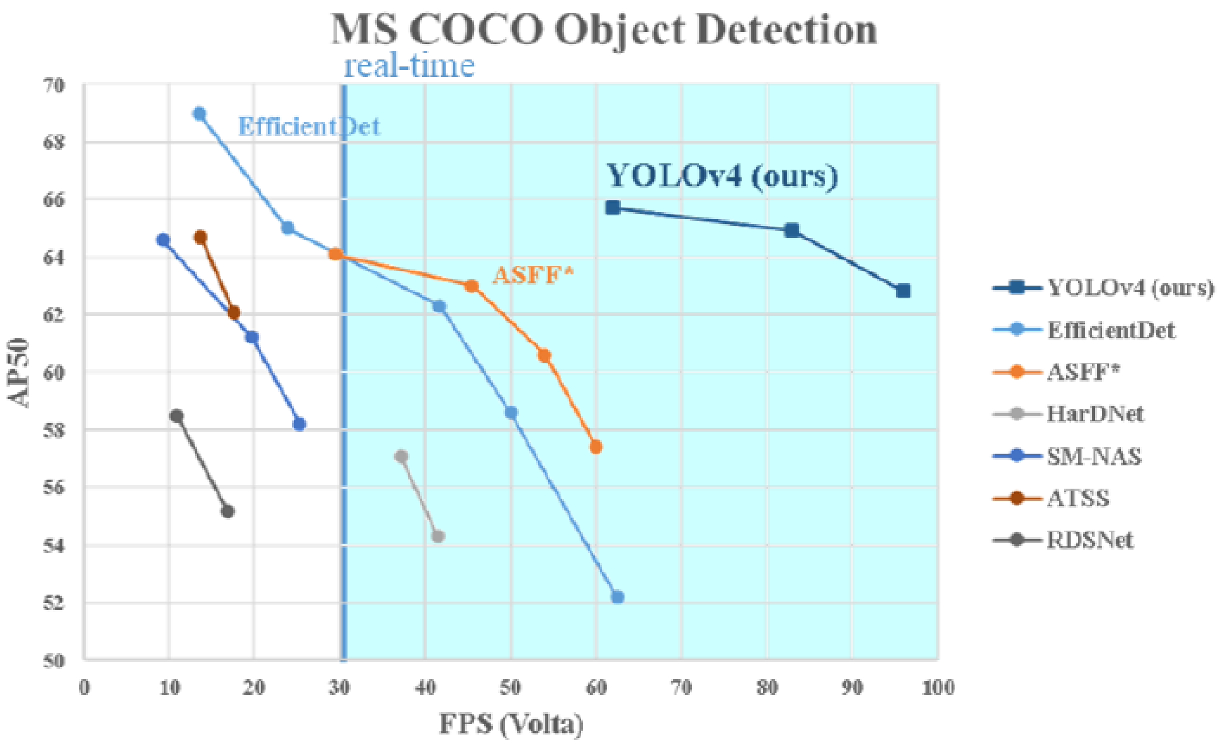

MS COCO Object Detection
real-time
YOLOv4 (ours)
FSAF
M2Det
LRF
AP
FPS (Maxwell)
45
43
41
39
37
35
33
31
29
27
25
0
10
20
30
40
50
60
YOLOv4 (ours)
YOLOv3
LRF
SSD
RefineDet
M2Det
PFPNet
RetinaNet
FSAF
CornerNet


MS COCO Object Detection
real-time
YOLOv4 (ours)
FSAF
M2Det
LRF
AP50
FPS (Maxwell)
70
65
60
55
50
45
40
0
10
20
30
40
50
60
YOLOv4 (ours)
YOLOv3
LRF
SSD
RefineDet
M2Det
PFPNet
RetinaNet
FSAF
CornerNet


MS COCO Object Detection
real-time
TridentNet
SAPD
HSD
YOLOv4 (ours)
CenterMask
AP
FPS (Pascal)
48
46
44
42
40
38
36
34
32
30
28
0
10
20
30
40
50
60
YOLOv4 (ours)
CenterMask
EFGRNet
HSD
DAFS
SAPD
GuidedAnchor
RPDet
Libra R-CNN
FreeAnchor
RetinaMask
Cascade R-CNN
CenterNet
TridentNet


MS COCO Object Detection
real-time
TridentNet
RPDet
SAPD
HSD
YOLOv4 (ours)
EFGRNet
AP50
FPS (Pascal)
YOLOv4 (ours)
EFGRNet
HSD
DAFS
SAPD
GuidedAnchor
RPDet
Libra R-CNN
FreeAnchor
RetinaMask
Cascade R-CNN
CenterNet
TridentNet


MS COCO Object Detection
real-time
EfficientDet
YOLOv4 (ours)
ASFF*
AP
FPS (Volta)
YOLOv4 (ours)
EfficientDet
ASFF*
HarDNet
RetinaNet
SM-NAS
NAS-FPN
ATSS
RDSNet
CenterMask


MS COCO Object Detection
real-time
EfficientDet
YOLOv4 (ours)
ASFF*
AP50
FPS (Volta)
YOLOv4 (ours)
EfficientDet
ASFF*
HarDNet
SM-NAS
ATSS
RDSNet

REFERENCES

[1] Deep Learning Based Pothole Detection and Reporting System (IEEE 2020) by Chellaswamy C

[2] The Modern Pothole Detection using Deep Learning (IEEE 2020) by Abhishek Kumar

[3] Pothole Detection and Reporting using image processing using a Raspberry Pi microcontroller(IEEE 2018) by Mae M. Garcillanosa

[4] In the article "Deep Learning," YOLO Neural Network-based learning algorithm for finding potholes in a pavement surface by Ernin Niswatul Ukhwah

[5] The article written by E. N. Ukhwah et al is a YOLOv4-based pothole detection system

[6] A_Real-Time_Pothole_Detection_Approach_for_Intelli by Hsiu-Wen Wang

[7] Automated Detection of Dry and Water-Filled Potholes using Multimodal Sensing System : (IIT HYD) by Shenu PM

[8] G. Anoosh, S. Raghunathan, A. Sundar, " Pothole Detection", M-Tech project report, IIT Madras. 2009-2010

[9] Accelerometer Sensor Network For Reliable Pothole Detection by Teodor Kalushkov

[10] Pothole Detection with Image Segmentation for Advanced Driver-Assisted Systems by H. K. I. S. Lakmal

[11] Automated Detection and Notification of Pothole to Aid Driver by Bhagyashree Ghodake

[12] Smart Pothole Detection and Mapping System by Rohan Borgalli

[13] Smart Pothole Detection Using Deep Learning Based on Dilated Convolution by Khaled R. Ahmed

[14] Intelligent Deep Learning-based Pothole Detection and Reporting System by Rohitaa r

[15] A Deep Learning Approach for Street Pothole Detection by K.R.Ping

Deep Learning Based Pothole Detection

D Rohit Rajan
Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology
Hyderabad, India
rohitrajan2307@gmail.com

Mohammad Khaja Faizan
Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology
Hyderabad, India
fabulousfaizan1234@gmail.com

Rajinikanth Kundelu
Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology
Hyderabad, India
kundelurajini7@gmail.com

Neha Nandal
Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology
Hyderabad, India
neha1607@grietcollege.com

Vasu Sena Gunda
Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology
Hyderabad, India
gundavasusena1234@gmail.com

***Abstract* - Potholes are formed due to wear and tear and weathering of roads. They cause not only discomfort to citizens but also deaths due to vehicle accidents. In many developing countries and developed countries, the main problem is road deterioration. In the US nearly 2000 accidents were recorded due to potholes and road damage. The main aim of this project is to decrease the road accidents happening daily around the world. Many people while driving a car cannot see some of the potholes on the road and if they do not slow down their vehicles, there is a chance of an accident or vehicle damage. Therefore to decrease this kind of accident we came up with a project which recognizes potholes on the roads and alerts the driver by making a beep sound. To accomplish this objective we used the Yolo algorithm for pothole detection which uses neural networks. If the cameras can be installed on moving vehicles, then the potholes can be detected in real-time and avoided by alerting the driver.**

Keywords - Pothole Detection, YOLO, You Only Look Once, Deep Learning, Machine Learning.

I. Introduction

The strain on the infrastructure has increased as the world's population has grown. Traffic on the roads is at a record high. It has been harder and harder to control this traffic. This is the main driving force behind creating a car with enough intelligence to help the driver in several ways. Road conditions getting worse is one of the issues that are getting worse for the roadways. The road is challenging to drive for various reasons, including rain, traffic accidents, or normal wear and tear. Unexpected obstacles on the road could lead to more collisions. Additionally, the vehicle's fuel usage rises as a result of poor road conditions, wasting valuable gasoline. Due to these factors, learning about such dangerous roads is crucial.

Due to these factors, it is crucial to gather information about these hazardous road conditions and warn the driver about the road condition. We in this project try to design and build such a system that can help drivers find the road condition. This system collects information about potholes and road conditions and sends an alert to the driver.

The purpose of a pothole detection system is to alert drivers to the uneven potholes and roadways in its route. We examine the various means via which the system's objective can be accomplished. We justify the methods we used in this project. We utilised the following ways to test this model:

- Building a model that will detect the potholes in its path.
- Highlighting the potholes so that the person is alerted.

Experts have experimented with different crater detection methods as a fascinating research field. Here are some suggestions for drill detection techniques:

- Measure and visualize craters using Microsoft Kinect Sensor
- Pothole detection study based on 3D project transformation
- Pothole detection based on SVM of pavement emergency images
- Detection and sharing of road obstacles by smartphone multimedia sensor analysis
- Real-time using Android smartphone with accelerometer pothole detection
- Pothole detection algorithm using an efficient stereo vision system
- Excavation detection and warning system for autonomous vehicles
- Detection of potholes in autonomous vehicles

The main objective of this project is to develop a system that can detect Potholes in video inputs. Hole detection can be done in real-time. The YOLO algorithm was used for this. The car's front camera records road video as input to the computer. The computer uses the YOLO algorithm to detect potholes in the video and alert the driver to road conditions.

Advantages over Existing System

- Fast and reliable
- Cost efficient

II. Literature Study

Using a GPS to pinpoint the location, an accelerometer and an ultrasonic sensor was put in the bottom of an automobile and driving at a speed of 25 km/h [1]. The control room is informed of the pothole's location by the microcontroller, which also senses it. The GPS is initialized by the microcontroller (ATmega328), which also gives us the coordinates. Comparative analysis of CNN, KNN (k-Nearest Neighbors), and Kirchhoff's Theory Method was the methodology adopted. [2]They installed a camera on the vehicle, identified the potholes, and marked their positions using an app they developed so that a vehicle without a camera could still get the information about the pothole and provide the driver with the necessary alerts. [4]Raspberry Pi microcomputers were used to implement the entire system, and 100% of the reports were successful.

The web server, Dropbox, and the Internet were used to store and access the reported picture of the pothole and its location. [5]For photos of asphalt pavement, the Yolo neural network model helped identify potholes. It shows a reasonable level of location accuracy of the application-specific structure, Yolo v3, Yolo v3 Tiny, and Yolo v3 SPP which have mAP values of 83.43%, 79.33%, and 88.93% correspondingly, and the accuracy of the range estimation correspondingly, 64.45%, 53.26%, and 72.10%. As a result, there is a good chance that it will be developed and used. The Yolo V3 Algorithm was employed.

The YOLOv4 algorithm used in the system this paper proposes allows it to outperform the systems discussed before. A drawback of YOLOv3 based on object detection is that the results are less precise than YOLOv4. YOLOv4 gives more precise bounding boxes hence resulting in superior IoU esteem. [6]The accelerometer data is normalized by Euler angle computation and is embraced within the pothole detection algorithm to get the pothole data. The spatial interpolation strategy is utilized to decrease the area mistakes from GPS data. [7]Laser diodes with a camera (laser stripe) are utilized to detect water-filled potholes by lighting up them and capturing pictures of the pothole employing a camera beneath laser frequency, By implementing a feature extraction algorithm in image processing, we can detect dry and water-filled potholes, and calculate its range and depth. [8]Since ultrasonic sensors cannot penetrate the air-water interface, we use laser sensors to obtain the depth of water-filled potholes. The ultrasonic transmitter emits periodic 42 kHz pulses and the receiver listens for the echo. The echo return time is used to obtain the distance. The Msp430 is used to control the sensor in signal transmission and calculate the distance using the received echo signal. When a distance difference is achieved, the driver receives a clear warning.

The summarized literature survey is shown in Table 1.

TABLE I. Summary of Literature Survey

Teodor Kalushkov et. al. [9]	The pothole detection system uses two accelerator sensing nodes whose primary purpose is to analyze potholes in the road surface and map large potholes to increase the safety and comfort of all passengers. It easily becomes part of the environment for smart cities.
H. K. I. S. Lakmal et. al. [10]	Since most cars are equipped with camera systems, this paper proposes a simple image-based machine learning framework for pothole detection. In this paper, they investigate the contribution of texture features of edge features and Gabor-net features to water surface detection. First, features are ranked by importance. Three random forest classifier models were developed and trained to detect the potholes.
Bhagyashree Ghodake et. al. [11]	The proposed device tracks road conditions and identifies potholes. Identified information about potholes is made available to drivers and governments. The proposed system is divided into three units
Rohan Borgalli et. al. [12]	We have studied the various hole detection and mapping methods developed so far and propose an accurate and efficient hole detection and mapping system. We tend to target such insights to detect and locate potholes on multi-lane roads.
Khaled R. Ahmed et. al. [13]	This study demonstrates an improved VGG16 (MVGG16) network that reduces computational costs and improves training results by removing multiple convolutional layers and using different stretch scales. Furthermore, the fastest R-CNN uses MVGG16 as the base network in this study.
Rohitaa R et. al. [14]	A deep learning algorithm has been proposed to automatically detect potholes on highways. Trained and tested on YOLOv3. Use a rating scale to compare the outputs of the three models.
Ping Ping et. al. [15]	In this paper, an efficient pothole detection system is proposed to automatically identify potholes on roads using deep learning algorithms. Using the preprocessed dataset, the following four models - YOLO V3, SSD, HOG with SVM and Faster R-CNN were trained and tested. Then checked the performance and accuracy of the four models. According to the test results, the YOLO V3 model performed well due to its faster and more accurate detection results.

A. Abbreviations and Acronyms

The following are the most commonly used abbreviations and acronyms across the project.

YOLO	You only look once

CV	Computer Vision
SVM	Support Vector Machine
FPS	Frames Per Second
RCNN	Regions with convolutional neural networks
KNN	K-nearest neighbours
PIP	Package Installer for Python

III. Methodology

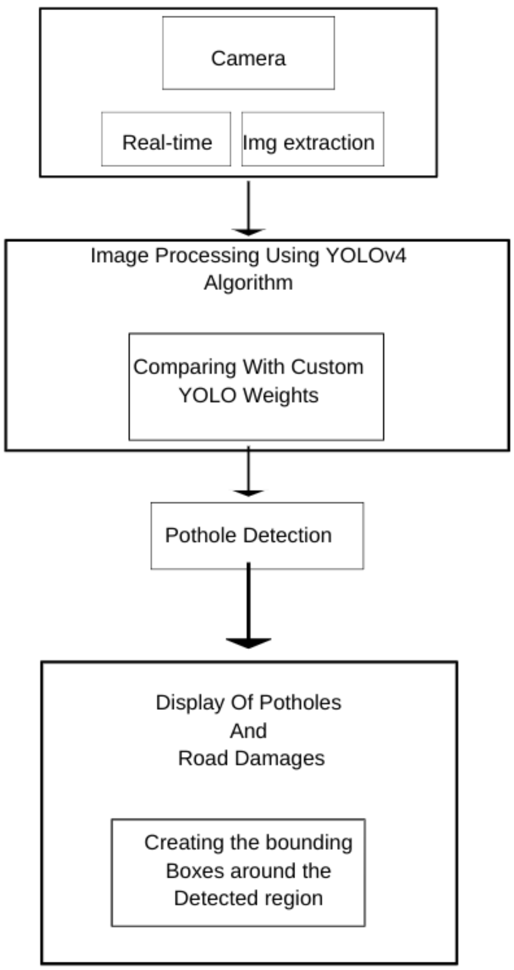


Fig. 1. System Block Diagram

The proposed system is divided into two steps, as shown in Figure 1. Determine and evaluate the size of the pit. The images captured by the camera are sent to the image detection system. The pothole detection unit of the proposed system is based on the YOLO series of object detectors. YOLO is the acronym for "You Only Look Once".

As the name suggests, it is based on the principle that objects in an input image can be identified and identified at a glance. This technique treats object detection as a regression task.

IV. Implementation

The system architecture design process as shown in Figure 2 identifies the subsystems that make up the system and frameworks for communication and coordination subsystems. The goal of architectural design is to define the overall structure of a software system.

V. YOLO Algorithm

YOLOv4 is a SOTA (State-of-the-art) model for real-time object detection, as shown in Figure 3. It was released by Alexey Bochkovsky in April 2020 and is the fourth release of YOLO. It achieves SOTA performance on the COCO dataset consisting of 80 different object classes. YOLO is a single-phase detector.

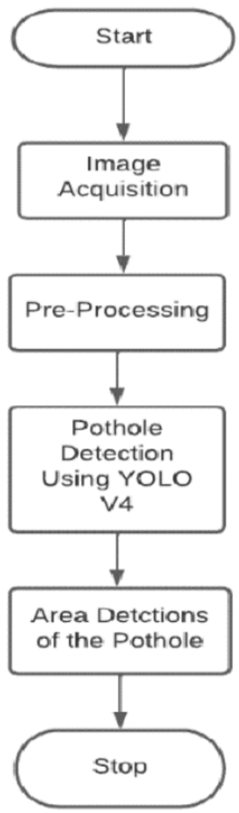


Fig. 2. Flowchart Of The Model

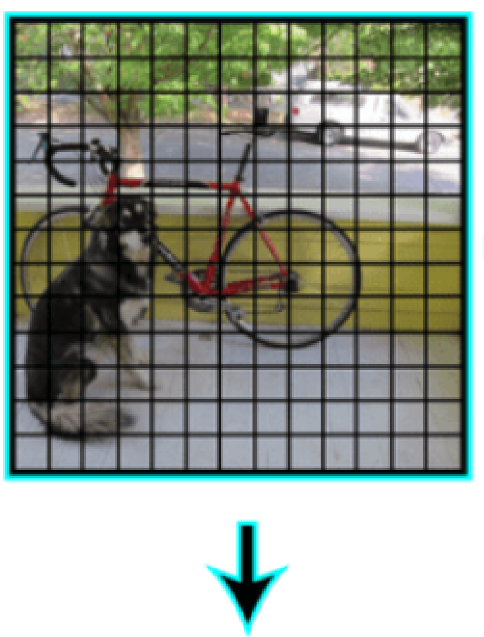
- One-step methods are one of the two main modern approaches for object detection tasks where speed of execution is important.
- In the one-step detector model, the ROI (region of interest) is removed and the class and bounding box are predicted for the entire image.
- So this makes them faster than two-stage detectors.

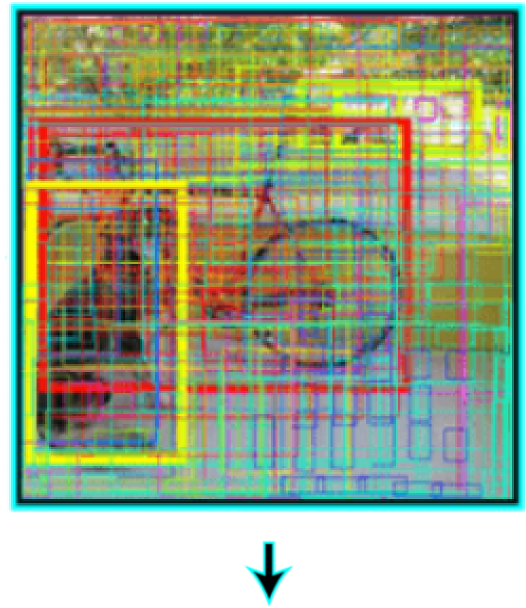
- Another example is FCOS RetinaNet and SSD. The first version of YOLO was written in the Dark Net Framework, a high-performance open-source framework for implementing neural networks written in C and CUDA.
- DarkNet is usually the main network.
- It divides object detection tasks into regression tasks and classification tasks.
- The regression predicts the category and bounding box of the entire image in one direction, which helps to identify the location of the object.
- The classification defines the category of the object.

A. Algorithm Overview:

The architecture consists of different parts, in general, the parts are as follows

- Input comes first, basically the set of training images we have been fed to the network
- It is processed in a parallel GPU cluster. Then comes the spine and neck for feature extraction and assembly.
- Neck detection and head detection together can be called object detection, and finally detection/prediction is performed by the head.
- Basically, the head is responsible for identification (localisation and classification).





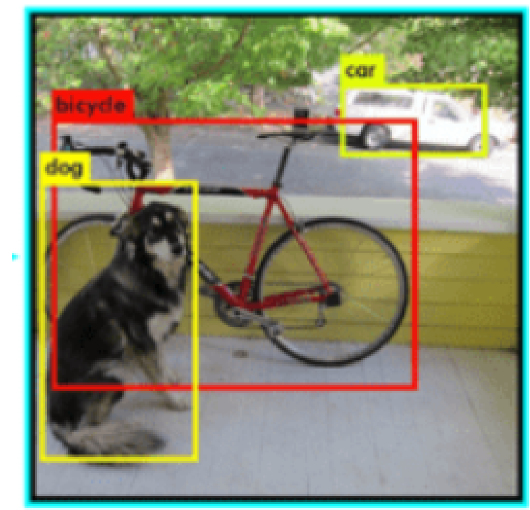


Fig. 3. YOLO Algorithm Over-view using bounding boxes

VI. Results And Discussion

After the execution of the entire program the following results are shown:

- Every pothole or road damage detected will be framed under a different coloured box with the accuracy of the detected damage as shown in Figure 4, figure 5 and Figure 6.

Fig. 4. Linear crack with accuracy

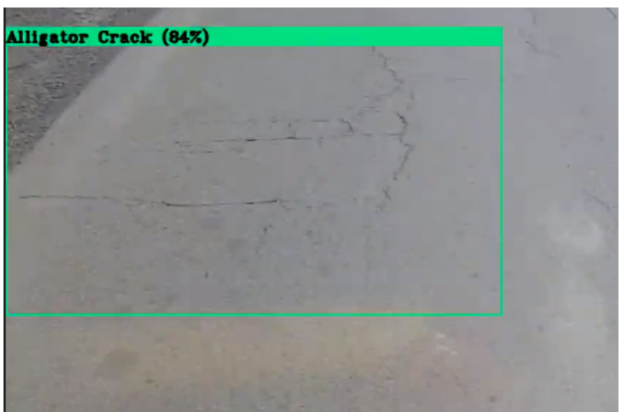


Fig. 5. Alligator cracks with accuracy

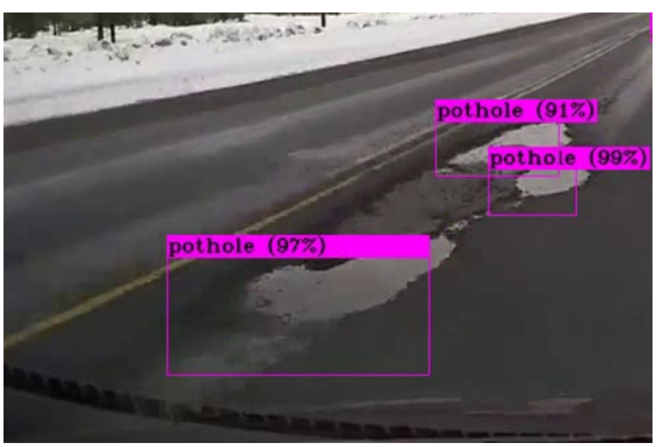


Fig. 6. Potholes with accuracy

- YOLO algorithm may not be the most accurate algorithm for pothole detection but it can certainly work at higher speeds when vehicles move faster
- When vehicles are at higher speeds YOLO Algorithm works best

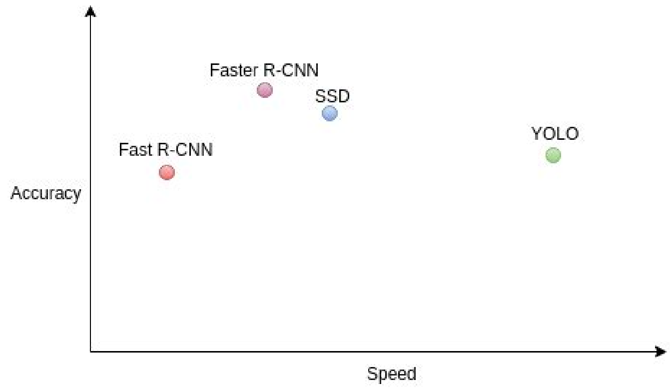


Fig. 7. Algorithm comparison in terms of speed and accuracy

- YOLO Algorithm is the fast algorithm when it comes to processing more number of frames per second in a given video.
- If an algorithm can process more FPS, then the time taken to get the result will be reduced drastically.

Figure 7 and Figure 8 show the comparative analysis of different algorithms implemented.

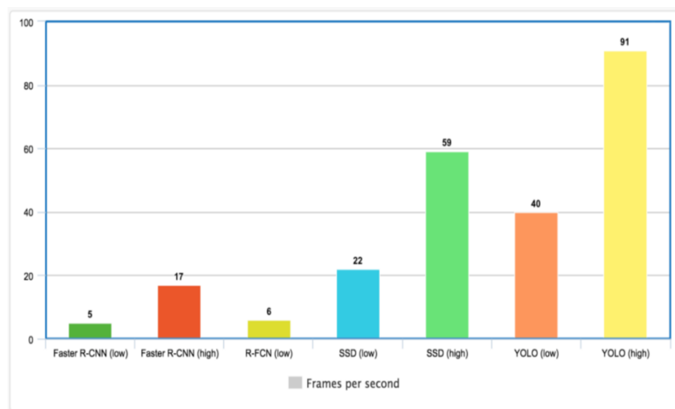


Fig. 8. Algorithm comparison in terms of FPS

VII. Conclusion

In this work, a model is created which is able to detect potholes that are present on the road. The tests conducted in this study used a dataset that included photos of potholes in various lighting situations, on various road types, and with various shapes and sizes. Our model is able to accurately predict potholes at a much higher rate. In our future research, we will develop a durable model to cope with these extreme conditions, such as very bad road conditions, and be able to predict potholes using unusual camera angles. In addition, as an extension of this study, the depths and distances (in meters) of potholes can also be estimated using calibrated stereo cameras.

References

[1] Deep Learning Based Pothole Detection and Reporting System (IEEE 2020) by Chellaswamy C

[2] The Modern Pothole Detection using Deep Learning (IEEE 2020) by Abhishek Kumar

[3] Pothole Detection and Reporting using image processing using a Raspberry Pi microcontroller(IEEE 2018) by Mae M. Garcillanosa

[4] In the article "Deep Learning," YOLO Neural Network-based learning algorithm for finding potholes in a pavement surface by Ernin Niswatul Ukhwah

[5] The article written by E. N. Ukhwah et al is a YOLOv4-based pothole detection system

[6] A_Real-Time_Pothole_Detection_Approach_for_Intelli by Hsiu-Wen Wang

[7] Automated Detection of Dry and Water-Filled Potholes using Multimodal Sensing System : (IIT HYD) by Shenu PM

[8] G. Anoosh, S. Raghunathan, A. Sundar, “ Pothole Detection”, M-Tech project report, IIT Madras. 2009-2010

[9] Accelerometer Sensor Network For Reliable Pothole Detection by Teodor Kalushkov

[10] Pothole Detection with Image Segmentation for Advanced Driver-Assisted Systems by H. K. I. S. Lakmal

[11] Automated Detection and Notification of Pothole to Aid Driver by Bhagyashree Ghodake

[12] Smart Pothole Detection and Mapping System by Rohan Borgalli

[13] Smart Pothole Detection Using Deep Learning Based on Dilated Convolution by Khaled R. Ahmed

[14] Intelligent Deep Learning-based Pothole Detection and Reporting System by Rohitaa r

[15] A Deep Learning Approach for Street Pothole Detection by Ping Ping