Mini Project Synopsis on

Real Time Object Detection

TE - I.T Engineering

Submitted By

Snehal Shanbhag 19104008

Akansha Rawat 19104007

Pranjali Shimpi 19104017

Under The Guidance of

Prof. Sonal Jain

Prof. Charul Singh

DEPARTMENT OF INFORMATION TECHNOLOGY

A. P. SHAH INSTITUTE OF TECHNOLOGY

G.B. Road, Kasarvadavali, Thane (W), Mumbai-400615

UNIVERSITY OF MUMBAI

Academic year : 2021-22

CERTIFICATE

This to certify that the Mini Project report on **Real Time Object Detection** has been submitted by Snehal Shanbhag 19104008, Akansha Rawat 19104007 and Pranjali Shimpi 19104017 who are Bonafide students of A. P. Shah Institute of Technology, Thane, Mumbai, as a partial fulfillment of the requirement for the degree in **Information Technology**, during the academic year **2021-2022** in the satisfactory manner as per the curriculum laid down by University of Mumbai.

Prof. Sonal Jain
Guide

Prof. Charul Singh
Co-guide

Prof. Kiran Deshpande
Head Department of Information Technology

Dr. Uttam D. Kolekar
Principal

External Examiner(s)
1.
2.

Place: A. P. Shah Institute of Technology, Thane
Date: 11/04/2022

ACKNOWLEDGEMENT

This project would not have come to fruition without the invaluable help of our guide **Prof. Charul Singh & Prof. Sonal Jain**. Expressing gratitude towards our HoD, **Prof. Kiran Deshpande**, and the Department of Information Technology for providing us with the opportunity as well as the support required to pursue this project. We would also like to thank our teacher **Prof. Nahid Shaikh** who gave us her valuable suggestions and ideas when we were in need of them. We would also like to thank our peers for their helpful suggestions.

TABLE OF CONTENTS

Chapter 1

Introduction

Object detection is a technology that detects the semantic objects of a class in digital images and videos. One of its real-time applications is self -driving cars. In this, our task is to detect multiple objects from an image. The most common object to detect in this application is the car, motorcycle, and pedestrian and other objects in a roadway. For locating the objects in the image we use Object Localization and have to locate more than one object in real-time systems. There are various techniques for object detection, they can be split up into two categories, first is the algorithms based on Classifications. CNN and RNN come under this category. In this, we have to select the interested regions from the image and have to classify them using the Convolutional Neural Network. This method is very slow because we have to run a prediction for every selected region. The second category is the algorithms based on Regressions. The YOLO method comes under this category. In this, we won't select the interested regions from the image. Instead, we predict the classes and bounding boxes of the whole image at a single run of the algorithm and detect multiple objects using a single neural network. The YOLO algorithm is fast as compared to other classification algorithms. In real time our algorithm processes 45 frames per second. YOLO algorithm makes localization errors but predicts less false positives in the background.

These algorithms are not tested with degraded images, i.e. they are trained with academic data sets, including ImageNet, COCO and VOC, etc. but they are not well tested with randomly captured data sets. The main issues of images captured in the real scene are :

1. Due to the instability of the camera, the captured images may be blurred.
2. The images can also not be clear enough because the object can be obstructed.
3. The images may have poor quality as a result of bad weather, overexposure or low resolution.

1.1. Purpose

- The main purpose of object detection is to identify and locate one or more effective targets from still image or video data.
- The goal of object detection is to recognize instances of a predefined set of object classes (e.g. {people, cars, bikes, animals}) and describe the locations of each detected object in the image using a bounding box.

1.2. Objectives

- The main purpose of object detection is to identify and locate one or more effective targets from still image or video data. It comprehensively includes a variety of important techniques, such as image processing, pattern recognition, artificial intelligence and machine learning.
- To automate the recognition and extraction process.
- To detect the object segment from the video frame.
- To classify the features in order to recognize the objects detected.
- To reduce human efforts.
- To provide efficiency and accuracy.

1.3. Scope

- Detection of pedestrians
- Detection of vehicles
- Detection of other roadway objects i.e. traffic signs and lights, etc.
- Can be used in :
    - Biometric recognition
    - Surveillance
    - Smart cars
    - Lane detection
    - Medical analysis

Chapter 2

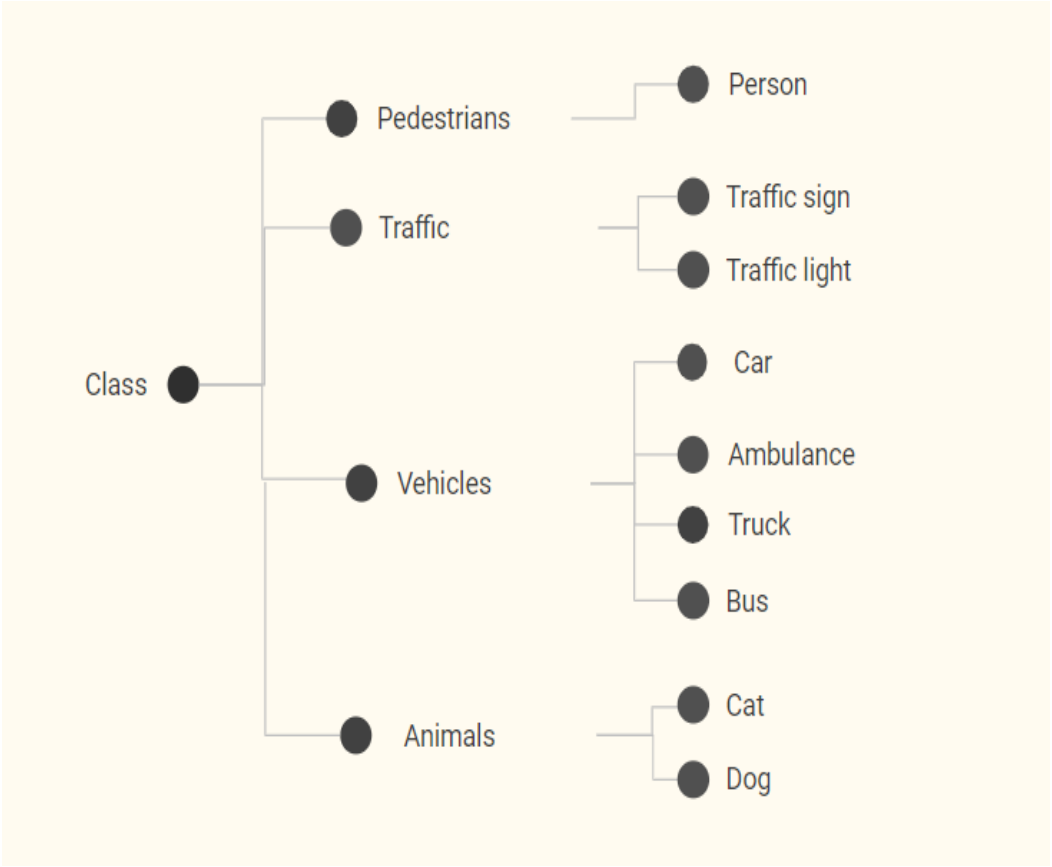
Problem Definition

"Real Time Object Detection" project detects objects efficiently based on the YOLO algorithm and applies the algorithm on image data and video data to detect objects. The objective is to detect and track the generic object in real time. In real life, therefore, we require rich information about the surrounding. We need to understand how the objects are moving with respect to the camera. It would also help to recognize the interaction between objects. For example, in the case of the self-driving car the knowledge about the interaction between the pedestrians will help to predict the pedestrian behavior accurately. This prediction will eventually help the self-driving car to make intelligent choices on a crowded road. To make a ML project on Real Time Object Detection with the best algorithm such that the project focuses on accuracy of the output, detection of 10 classes or as much as we can.

Chapter 3

Proposed System

- The proposed method uses Yolo v5 to develop a system model which consists of multilayers to classify the given objects into any of the defined classes.
- The schemes then use multiple images and detect the objects from these images, labeling them with their respective class label.



3.1. Features and Functionality

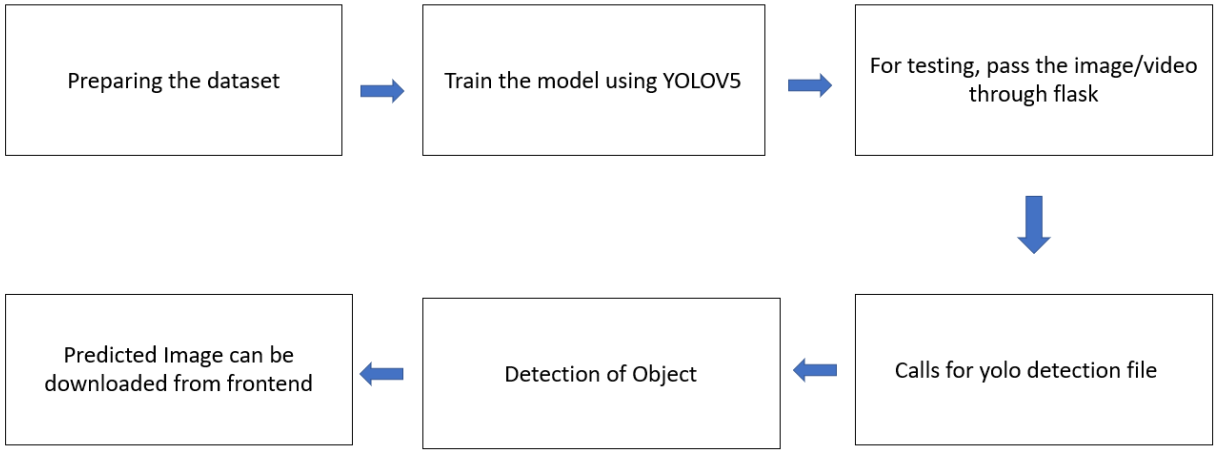
- Pedestrian detection with rectangular annotation.
- User-friendly interface.
- Helps understand the lane.
- Helps in identifying objects on the road.

Chapter 4

Project Outcomes

- The outcome of the Object Detection Project is to recognize and locate all trained objects in a frame related to road.
- Object detection is implemented in two ways:
    - Video streaming
    - Through images
- Users can upload an image or video that is to be detected through the web application and get the desired output with certain accuracy.

4.1 Block Diagram



Chapter 5

Software Requirements

- **Python:**

  Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small- and large-scale projects. Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library.

- **Open CV :**

  A huge open-source library for computer vision, machine learning, and image processing. OpenCV supports a wide variety of programming languages like Python, C++, Java, etc. It can process images and videos to identify objects, faces, or even the handwriting of a human. When it is integrated with various libraries, whatever operations one can do in Numpy can be combined with OpenCV. OpenCV is better than tensorflow as OpenCV's points of strength are in the deployment side, we are deploying our model as part of an API.

- **Yolo Algorithm :**

  YOLO an acronym for 'You only look once', is an object detection algorithm that divides images into a grid system. Each cell in the grid is responsible for detecting objects within itself.YOLO is one of the most famous object detection algorithms due to its speed and accuracy.YOLOv5 is a family of compound-scaled object detection models trained on the COCO dataset, and includes simple functionality for Test Time Augmentation (TTA), model ensembling, hyperparameter evolution, and export to ONNX, CoreML and TFLite.

- **Flask:**

  Flask was originally designed and developed by Armin Ronacher as an April Fool's Day joke in 2010. Despite the origin as a joke, the Flask framework became wildly popular as an alternative to Django projects with their monolithic structure and dependencies.Flask is a web application framework written in Python. Armin Ronacher, who leads an international group of Python enthusiasts named Pocco, develops it. Flask is based on the Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects.

Hardware Requirements:

- i3 and above processor
- 8GB and above Ram

Chapter 6

Project Design

Frontend interface:

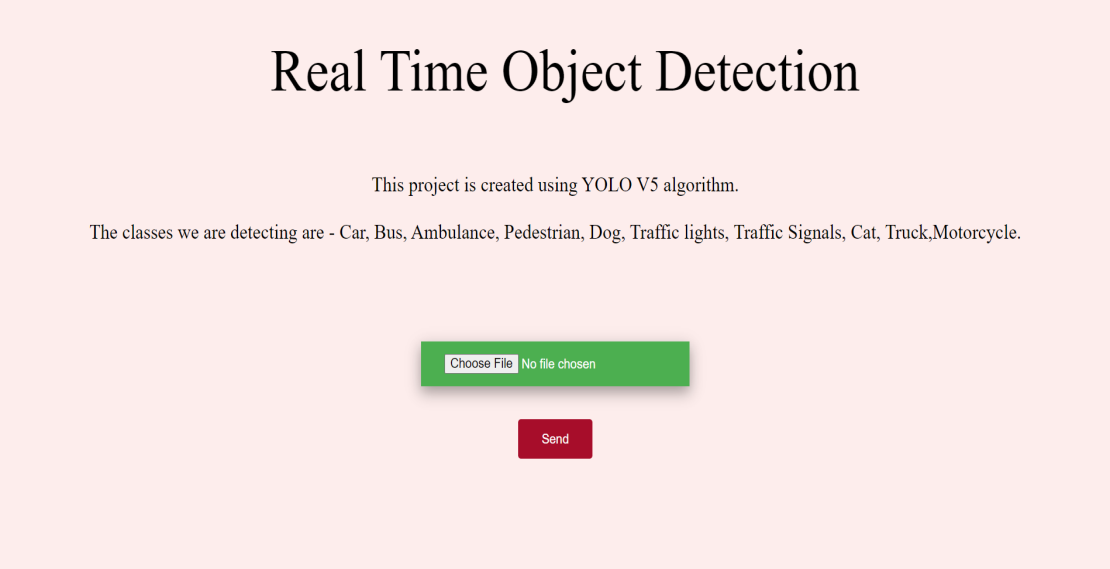


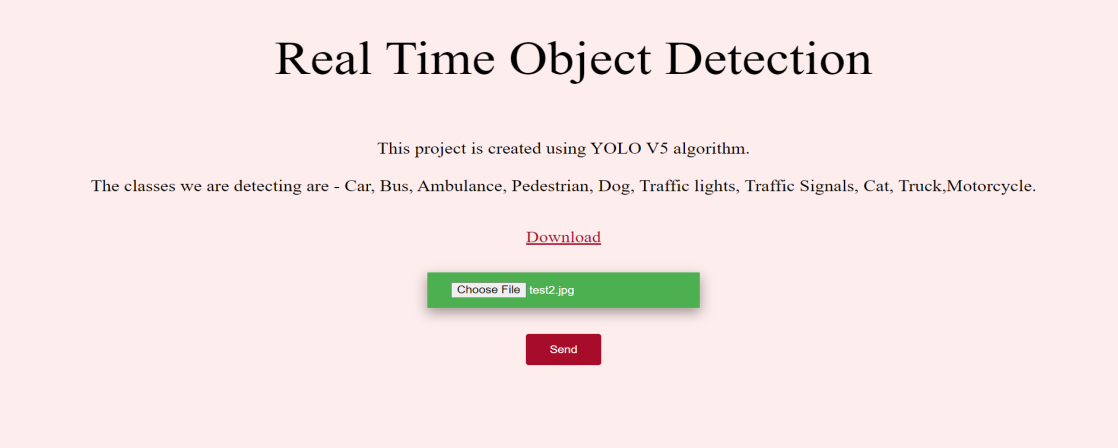
Image before detection:

Object Detection:

```
127.0.0.1 - - [24/Apr/2022 10:58:05] "POST /detect HTTP/1.1" 200 -
127.0.0.1 - - [24/Apr/2022 10:58:29] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [24/Apr/2022 10:58:29] "GET /static/index.css HTTP/1.1" 304 -
127.0.0.1 - - [24/Apr/2022 10:58:29] "GET /static/index.js HTTP/1.1" 304 -
<FileStorage: 'test2.jpg' ('image/jpeg')>
detect: weights=runs\train\exp\weights\best.pt, source=C:\Users\snehal\Downloads\yolov5custom\yolov5\instance\uploads\test2.jpg, data=data\dataset.yaml, imgsz=[
640, 640], conf_thres=0.25, iou_thres=0.45, max_det=1000, device=, view_img=False, save_txt=False, save_conf=False, save_crop=False, nosave=False, classes=None,
 agnostic_nms=False, augment=False, visualize=False, update=False, project=static, name=, exist_ok=True, line_thickness=3, hide_labels=False, hide_conf=False, h
alf=False, dnn=False
'git' is not recognized as an internal or external command,
operable program or batch file.
YOLOv5  2022-4-10 torch 1.11.0+cpu CPU

Fusing layers...
Model summary: 213 layers, 7037095 parameters, 0 gradients
image 1/1 C:\Users\snehal\Downloads\yolov5custom\yolov5\instance\uploads\test2.jpg: 352x640 2 persons, Done. (0.177s)
Speed: 1.4ms pre-process, 176.6ms inference, 0.0ms NMS per image at shape (1, 3, 640, 640)
Results saved to static
127.0.0.1 - - [24/Apr/2022 11:05:16] "POST /detect HTTP/1.1" 200 -
```

Output on frontend:.



Detected image/video:

Chapter 7

Project Scheduling

Sr no.	Group member	Time Duration	Work to be done
1.	Snehal Shanbhag	2nd week of March	Implementation of yolov5 Algorithm
	Snehal Shanbhag	2nd week of April	Implementation of training the Model
2.	Pranjali Shimpi	2nd week of March	Implementation of downloading the dataset from open source google api
	Pranjali Shimpi	2nd week of April	Implementation of UI using flask
3.	Akansha Rawat	2nd week of March	Implementation of training the model
	Akansha Rawat	2nd week of April	Testing of the trained model

Chapter 8

Conclusion

Object detection is a key ability for most computer and robot vision system. Being a unified object detection model that is simple to construct and train in correspondence with its simple loss-function, YOLO can train the entire model in parallel. YOLO is also better at generalizing Object representation compared with other object detection models and can be recommended for real-time object detection as the state-of-art algorithm in object detection. It should be noted that object detection has not been used much in many areas where it could be of great help. As mobile robots, and in general autonomous machines, are starting to be more widely deployed (e.g., quad-copters, drones and soon service robots), the need of object detection systems is gaining more importance. Finally, we need to consider that we will need object detection systems for nano-robots or for robots that will explore areas that have not been seen by humans, such as depth parts of the sea or other planets, and the detection systems will have to learn new object classes as they are encountered. In such cases, a real-time open-world learning ability will be critical. Comparing Yolo v3, v5, and v5 version - Yolov5 turned out to be the best algorithm taking less time and giving max accuracy.

References

1. https://leadingindia.ai/internshipproject
2. https://opencv.org/multiple-object-tracking-in-realtime/
3. https://towardsdatascience.com/implementing-real-time-object-detection-system-using-pytorch-and-opencv-70bac41148f7
4. https://www.mdpi.com/1424-8220/22/2/464/pdf
5. https://www.researchgate.net/publication/351411017_Real-Time_Object_Detection_Using_YOLO_A_Review