DIAGNOSIS OF OBJECTS FOR DRIVER-LESS VEHICLES (USING YOLO ALGORITHM)

A MINI PROJECT REPORT

Submitted by

KESARI ASHISH(B18ME024)
KANUGANTI SHANTHAN RAO(B18ME022)

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY
in
MECHANICAL ENGINEERING

Under the supervision of
Sri, J LAXMAN
Assistant Professor, MED

Department of Mechanical Engineering

KAKATIYA INSTITUTE OF TECHNOLOGY & SCIENCE: WARANGAL
(Autonomous Institute Under Kakatiya University, Warangal)
Apr/May–2021

DEPARTMENT OF MECHANICAL ENGINEERING

KAKATIYA INSTITUTE OF TECHNOLOGY AND SCIENCE, WARANGAL :506015

(An Autonomous Institute under Kakatiya University, Warangal)

CERTIFICATE

This is to certify that Mr. KESARI ASHISH (B18ME024) and Mr. KANUGANTI SHANTHAN RAO(B18ME022) has carried out the bonafide Minor project report entitled, **"DIAGNOSIS OF OBJECTS FOR DRIVER-LESS VEHICLES (USING YOLO ALGORITHM)"** in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Mechanical Engineering**.**

.

Project Supervisor,
Sri.J.Laxman
Assistant Professor, MED

Chairman,
Dr. U. Shrinivas Balaraj
Seminar Evaluation Committee

Dr. K. Sridhar

Professor & Head, MED

TABLE OF CONTENTS

TABLE OF FIGURES

ABSTRACT

Object detection is a key ability for most computer and robot vision systems. YOLO (YOU ONLY LOOK ONCE)-is an intelligent convolutional neural network (CNN) and is one of the most worth-while algorithms which is used for detection of objects in real time applications like Driverless vehicles. Object diagnosis or detection through YOLO, plays a major or crucial role in autonomous vehicle technology and even in computer vision. The algorithm applies a single neural network to the full picture, and afterwards separates the picture into regions and predicts bounding boxes and probabilities for every locale. These bounding boxes are weighted by the probabilities anticipated.

The process involved in diagnosis of object is- Image Classification, Localization and Detection. The operational principle of YOLO algorithm is to divide the image into grid cells and then furtherly for each grid cell, we can encore a vector. An image will be in multiple grid cells, so it finds the central point of the object and the object belongs to that particular grid cell. Applications of YOLO algorithm: Self-driving vehicles (braking distance), tracking objects, counting people, CCTV surveillance(security), Vehicle detection, Human computer interactions in Robotics, Electronics, Face Recognition.... etc.!!

Keywords: YOLO Algorithm, Object Detection, CNN, Driverless vehicles, Computer Vision, Braking Distance, Python, Robotics, Face Recognition, Human Computer Interface etc.…

ACKNOWLEDGEMENT

I wish to place on records my ardent and earnest gratitude to my Project Supervisor Sri. **J Laxman**, Assistant Professor, Department of Mechanical Engineering, KITSW.

I am extremely happy to mention a great word of gratitude to **Dr. K. Sridhar,** Head of Department of Mechanical Engineering, Kakatiya institute of technology and science, Warangal for providing me with all facilities for the presentation of this mini project.

I would also extend my gratefulness to all the staff members in my Department. I also thank all my friends and well-wishers who greatly helped me in my Endeavour.

KESARI ASHISH(B18ME024)
KANUGANTI SHANTHAN RAO(B18ME022)

1. INTRODUCTION

YOLO is so fast and has become standard way of detecting objects in the field of computer vision. Previously for object detection we used sliding window object detection and then more faster versions were invented such as R CNN, Fast R CNN, Faster R CNN, but in 2015 yolo was invented which outperformed all the previous object detection algorithms and, Full Form of YOLO is “YOU ONLY LOOK ONCE”.

YOLO is an object detection algorithm which is so efficient to detect multiple objects in a single image and provides a bounding box around the object and displays the object classification and probability of object recognized.

YOLO algorithm based on Regression instead of selecting the interesting parts of an image which predicts classes and bounding boxes for the whole image in one run of the algorithm. That is why it is called “YOU ONLY LOOK ONCE” because we are not repeating it. YOLO is so fast and efficient that it can even detect the objects in a video file.

Object detection plays a major role in Driver less cars (also called Self-Driving Cars) that helps the cars to drive by itself by finding the location of other objects like Cars, Traffic signs, Signs, Humans, etc.

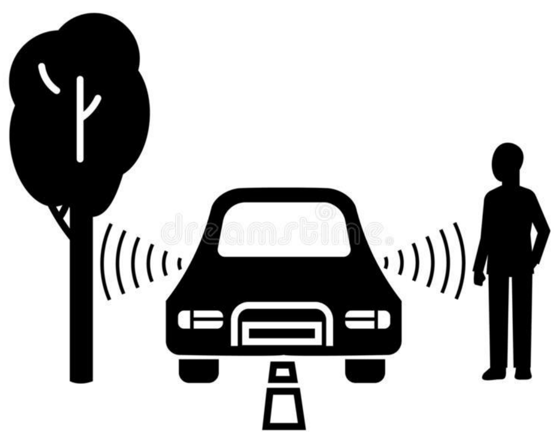


Figure-1.1

2. LITERATURE REVIEW

1.Research on Image Recognition of Insulators Based on YOLO Algorithm:

Authors: Shenghui Wang, North China Electric Power University, Baoding, China. & Leilei Niu, State Grid Baoding Electric Power Supply Company.
Published in: International Conference on Power System Technology (POWERCON)
Publisher: IEEE

This paper first and foremost presented the examination foundation of profound learning: the new YOLO (You Only Look Once) convolutional neural organization calculation, set up encasing picture data sets for train and test, and preprocessed the pictures of preparing separator pictures with the TensorFlow stage. With the YOLO calculation applied, the preparation of the picture information base for 5 days was finished, and a decent acknowledgment result was accomplished. At that point the outcomes were thought about between the Fast R-CNN calculation and the YOLO calculation in recognize speed and precision.

2. You Only Look Once: Unified, Real-Time Object Detection.

Authors: Joseph Redmon, University of Washington & Santosh Divvala, Allen Institute for AI.
Publisher: IEEE

YOLO, another way to deal with object discovery. Earlier work on object recognition repurposes classifiers to perform discovery. All things being equal, we outline object identification as a relapse issue to spatially isolated jumping boxes and related class probabilities. A solitary neural organization predicts jumping boxes and class probabilities straightforwardly from full pictures in a single assessment. Since the entire identification pipeline is a solitary organization, it tends to be enhanced start to finish straight forwardly on recognition execution.

3.YOLO-LITE: A Real-Time Object Detection Algorithm Optimized for Non-GPU Computers

Authors: Rachel Huang, School of Electrical and Computer Engineering, Georgia Institute of Technology, US. & Jonathan Pedoeem, Electrical Engineering, The Cooper Union, New York, United States.
Published in: 2018 IEEE International Conference on Big Data (Big Data).
Publisher: IEEE

This paper centers around YOLO-LITE, a continuous article recognition model created to run on versatile gadgets, for example, a PC or cellphone coming up short on a Graphics Processing Unit (GPU). The model was first prepared on the PASCAL VOC dataset then on the COCO dataset, accomplishing a mAP of 33.81% and 12.26% separately.

4. The Real-Time Detection of Traffic Participants Using YOLO Algorithm

Authors: Aleksa Ćorović RT-RK, Institute for Computer Based Systems, Novi Sad, Serbia & Velibor Ilić, RT-RK, Institute for Computer Based Systems, Novi Sad, Serbia & Siniša Đurić, RT-RK, Institute for Computer Based Systems, Novi Sad, Serbia.
Published in: 2018 26th Telecommunications Forum (TELFOR).
Publisher: IEEE

Item identification is one of the key programming parts in the up-and-coming age of independent vehicles. Traditional PC vision and AI approaches for object identification normally experience the ill effects of the sluggish reaction time. Current calculations and models dependent on counterfeit neural organizations, like YOLO (You Only Look Once) calculation, take care of this issue without accuracy misfortunes. In this paper we give the exhibit of the use of the most current YOLOv3 calculation for the discovery of traffic members. We have prepared the organization for 5 article classes (vehicle, truck, walker, traffic signs, and lights) and have shown the adequacy of the methodology in the assortment of the driving conditions (brilliant and cloudy sky, snow, haze, and night).

5. YOLO v3-Tiny: Object Detection and Recognition using one stage improved model

Authors: Pranav Adarsh, Delhi Technological University, Delhi, India & Manoj Kumar, Delhi Technological University, Delhi, India.
Published in: 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)
Publisher: IEEE

Object detection has seen numerous adjustments in calculations to improve execution both on speed and precision. By the consistent exertion of such countless specialists, profound learning calculations are developing quickly with an improved item identification execution. Different famous applications like walker location, clinical imaging, advanced mechanics, self-driving vehicles, face identification, and so forth diminishes the endeavors of people in numerous spaces.

3. YOLO ALGORITHM AND IT'S WORKING

Before getting into the yolo algorithm, firstly we will discuss about the difference between Classification, Localization and Detection.

Classification: Classification is the process of labelling an image. It refers to predicting the class of one object in an image.

For Example:

Figure 3.1.1

Here in the image, we have a car and we just label the image as car this is the process called classification.

Figure 3.1.2

Localization: This is the process which is just similar to the process classification but slightly differ by a bounding box around the object indicating that where is that object in an image. It refers to identifying the location of one or more objects in an image.
For ex:

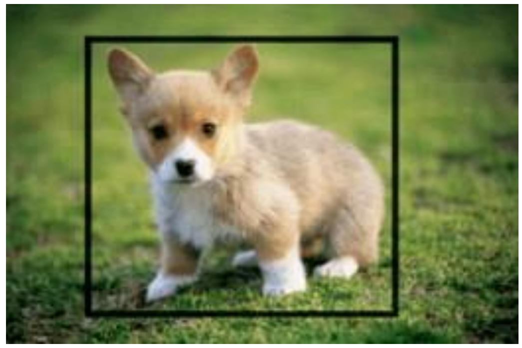


Figure-3.1.3

Figure 3.1.4

Here we have an image of a car and dog and also a bounding box around the car and dog describing where is the car and dog are located actually on frame.

Object Detection: It is the combined process of both object classification and localization takes place eventually and it is the process which is used to classify and localize multiple objects in an image.

Figure 3.1.5

It is also used to detect objects like cars bikes animals humans etc.., that is multiple objects in an image is detected eventually.

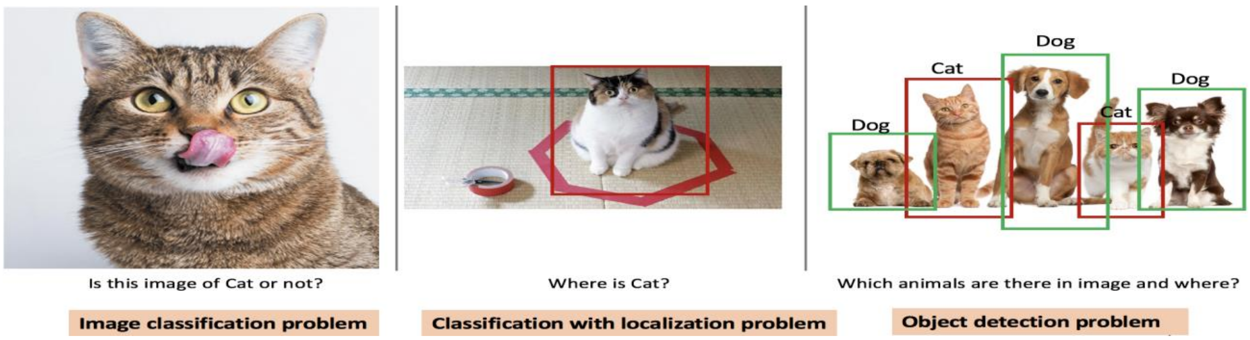


Figure-3.1.6

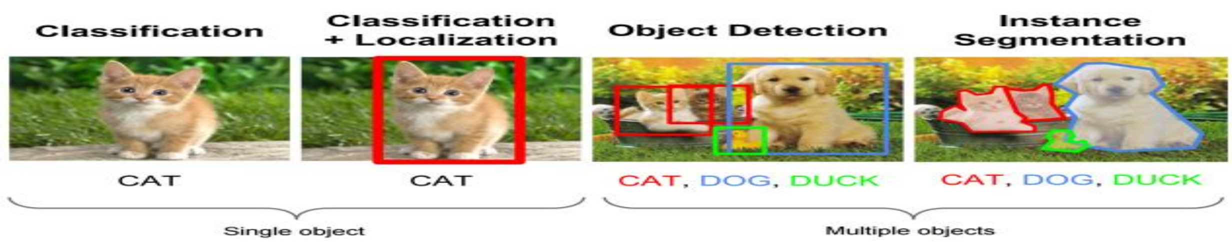


Figure-3.1.7

BOUNDING BOXES

Generally, when we pass an image, we also pass the co-ordinates of bounding boxes in order to get trained. Firstly, when we push an image, it straightly passes through Convolutional Neural network where image recognition takes place. After that the image is sent into Deep Neural Network and then we get Output Layer, which generally has SoftMax activation function in order to predict the classifications either it is car or bike or person etc...!

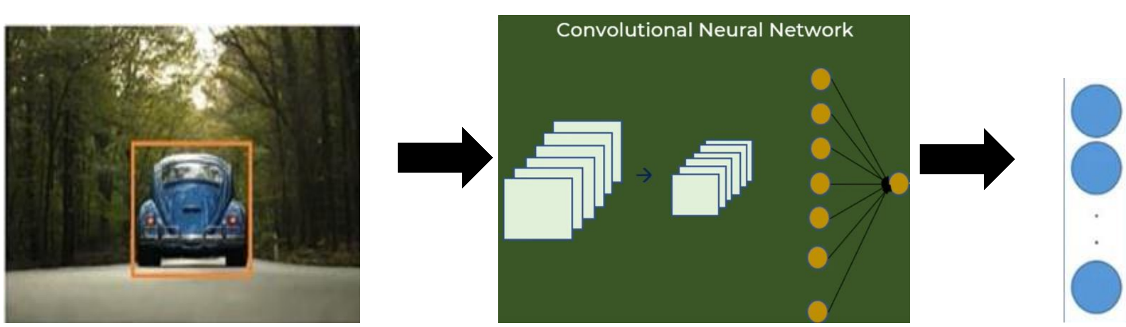


Figure 3.2.1

What is CNN...? And how to Train CNN...?

Convolutional Neural Network is a type of Neural Network which is applied to image processing problem and we can use CNN in natural language processing too. CNN is most important in AI and Deep Learning. They are more efficient for classifying and processing images. A regular neural network has an input layer hidden layer and an output layer. Input layers takes inputs of different formats while hidden layers perform calculations then the output layer delivers the outcome. Each of the layers contains neurons connected to the previous layer and each neuron has its own weight In CNN instead of connecting neurons to every neuron in previous layer it connects to only neuron which is close to it. The word convolutional refers to filtering process that happens in the process of network.

TRAINING A CNN:

If you are working with unlabeled data, you can use unsupervised learning methods or other option is to use a GAN'S (generative adversarial networks) with GAN to train two networks the first gives the artificial data samples that should resemble data in the training set and the second is a discriminative network it will distinguish between artificial and a true model.

How to train YOLO algorithm...?

- Suppose if we want to detect a monkey, firstly we will take a dataset of monkey images of around 300 to 400 images. There is a software called labelmg which is available for windows, Linux and mac.
- Open the software, and Create a new folder and store all the images of the monkeys and copy the folder path and paste it in the software.
- Read every pic and crop only the monkey object for every jpg image of monkey we will get a txt file (txt file contains of the vector) with the same name associated.
- Now select all the jpg and txt files and put them into a zip achieve and call this zip file name same as given in the yolo algorithm.
- Now upload this zip file in Google Drive and copy the Zip File Link.
- Open Google Colab Software Input the Yolo algorithm and paste the Zip file link in the algorithm.
- Run it to get final Output of the objects you trained in the YOLO algorithm.
- In object detection code using yolo algorithm you have to input the final output file and input the image you want to check by comparing the final output file to the input image it detects the objects.

Working with Bounding Boxes

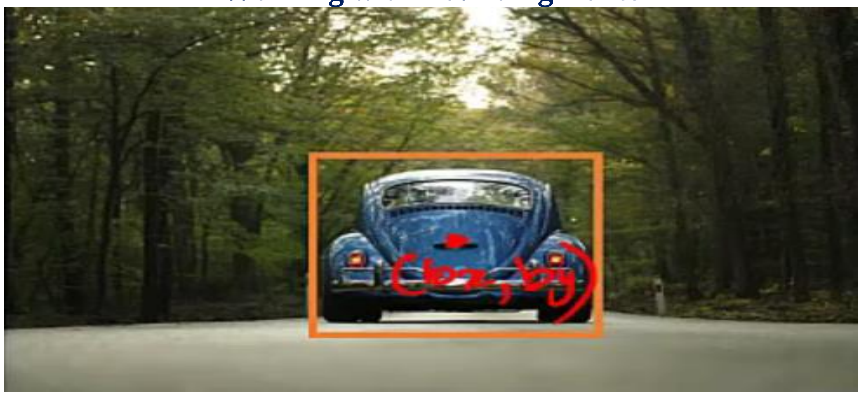


Figure 3.2.2

In those Let us take a car and we can draw bounding box co-ordinates from the center of the object let us take Bx and By at the center where bx represents input image and By represents Output image. We can also take Bh and Bw which represents Height and Width of the object. Generally, the bounding box co-ordinates lies between (0,0) and (1,1) and from our image let the x value be 0.5 and y value be 0.6 and height of the car is 30% so let it be 0.3 and width of the car is 40% and let it be 0.4.

What about the OUTPUT...?

Basically, we have different types of datasets which we push into our algorithm...! Ok then, What is a dataset..?

Dataset: A Dataset is a set of data under a variable whether the data may be in form of images, lists, Tables etc.

Let us have just basic datasets like Car Motorcycle and a person in our algorithm.

Figure 3.3.1

Figure 3.3.2

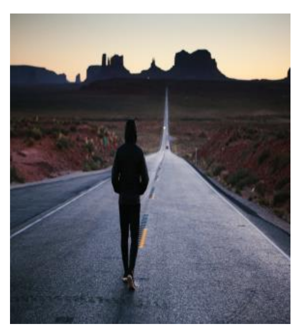


Figure 3.3.3

Let us now Divide our classifications: Car as C1 and motor cycle as C2 and person as C3.

Now Let us take a Vector “y”
where y ={Pc, Bx,By,Bh,Bw,C1,C2,C3} and where Pc is probability class which represents whether object is present or not.

We can Understand this by simple images:

Figure 3.3.4

Here in this image, we have an Object from our dataset “CAR”.
and from our Vector y, we have y= {1, Bx,By,Bh,Bw,0.96,0.4,0}.
Pc value=1 such that we have an object in a given image.
Bx and By represents the input and output of an image.
Bh represents the height of the object in an image.
Bw represents the width of an object in an image.

Here C1 represents the Probability that it is a car and that probability is 96% or 0.96.
Here C2 represents the Probability that it is a motorcycle and that probability is 4% or 0.4.
Similarly, C3 Represents the Probability that it is a Person and that probability is 0.

Let us take another image:

Figure 3.3.5

Here in this image we have an Empty Road with no objects from our dataset.
So y ={0,Bx,By,Bh,Bw,0,0,0}
Pc value=0 such that we don't have an object in a given image.
Bx and By represents the input and output of an image.
Bh represents the height of the object in an image.
Bw represents the width of an object in an image.

Here C1 represents the Probability that it is a car and that probability is 0.
Here C2 represents the Probability that it is a motorcycle and that probability is 0.
Similarly C3 Represents the Probability that it is a Person and that probability is 0.

Actual Working of YOLO Algorithm

Let us take an image and pass it into algorithm. Firstly the image passed is divided into tiny grid cells based upon the image the grid cells vary.

Figure 3.4.1

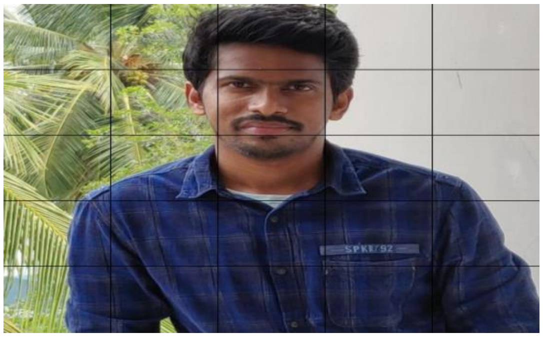


Figure 3.4.2

Here based upon the first image the image is divide into 3X3 i.e. 9 grid cells and Based upon the second image the image is divided into 5X5 i.e. 25 grid cells and then bounding boxes are drawn around the objects present in an image clearly with the help of Grid cells.

INTERSECTION OVER UNION

Intersection over union is one of the evaluating factors which determines how well the bounding box is predicted.

IOU(intersection over union)=intersection area/union area;

In this we find overlapping area and to find overlapping area we use IOU if they are completely overlapping then value will be 1 if they are not at all overlapping then value will be zero.

By using overlapping you will find the final bounding box this technique is called Non-max suppression so after neural network detects all the objects you apply no max suppression and you get the bounding boxes.

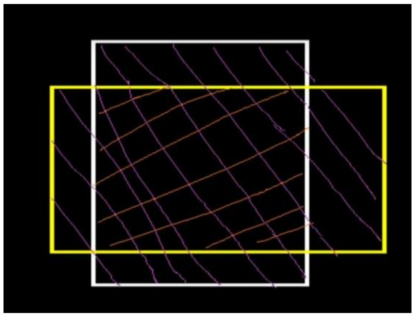


Figure 3.5.1

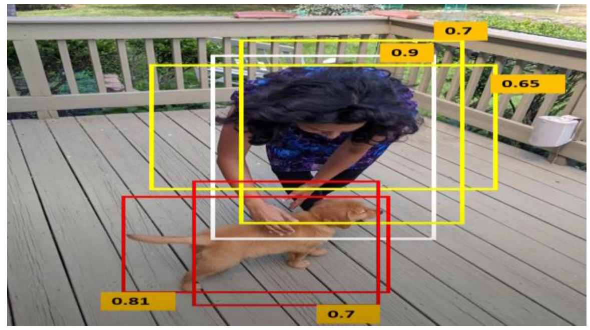
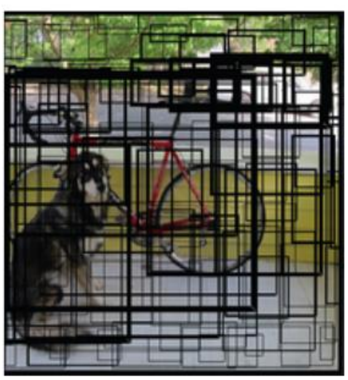


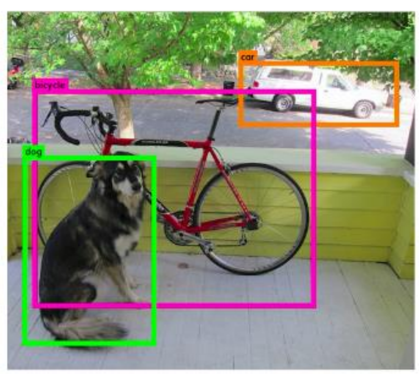
Figure 3.5.2

Non-Max Suppression

This is the final step in the object detection and this is the technique used to select most appropriate Bounding box for the object. Generally, the image consists of different objects of different shape and size. So, in that case, we have multiple Bounding boxes for multiple objects and then best bounding boxes are selected (consisting of max probability of object).



Multiple Bounding Boxes



Final Bounding Boxes

Figure 3.6.1

This technique of suppressing bounding boxes with less probability (of objects), is known as Non-Max Suppression.

The Non-Max Suppression considers mainly 2 things:

- Objectiveness Score given by Object(Probability)
- Overlapping of IOU
  For Ex:

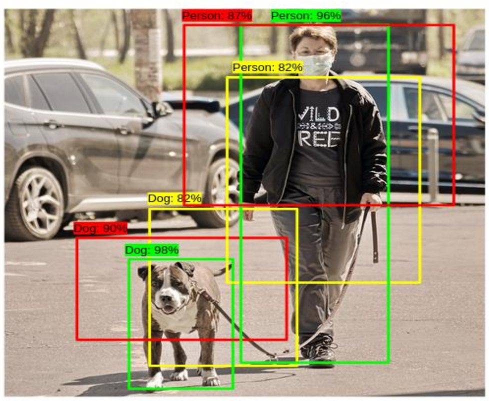
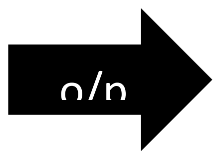


Figure 3.7.1



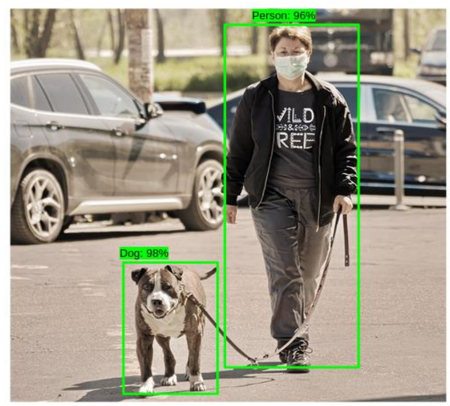


Figure 3.7.2

Here is an image of an Dog class and Person Class.
The Probabilities of above Dog Class are 98,90,82.Now By suppressing the probability classes of 82,90 we get the final Probability of Dog class which is 98.This method of suppressing Probabilities is called “Non-Max Suppression”.

Anchor boxes

Multiple Objects with same Grid cells:

If We Have Two Objects Having Centre In The Same Grid Cell Say Dog And A Person In This Case Both Dog And A Person Are In The Middle Of Grid Cell So Here We A Have A Dog Vector Of Size 7 And Person Vector Of Size 7 By Concatenating Both The Vectors We Get Final Vector Of Size 14 This Concept Is Called Anchor Boxes So It Has Two Anchor Boxes One Is Dog Anchor Box And Another Is Person Anchor Box You Can Have N Number Of Anchor Boxes Let’s Say There Are 15 Objects In A Single Image With Same Center Box Then You Have 15 Anchor Boxes.

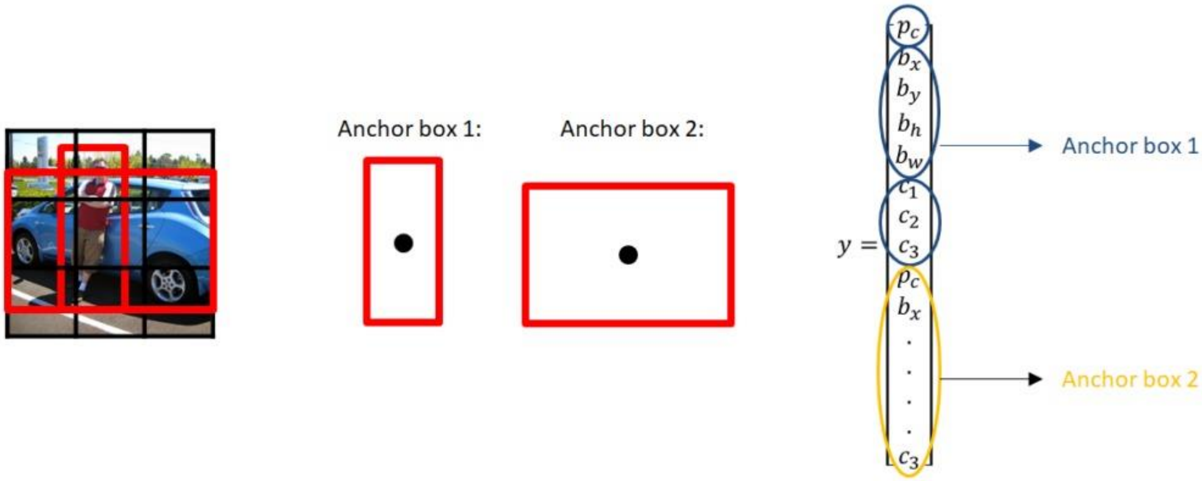


Figure 3.8.1

OUTPUTS OF YOLO ALGORITHM

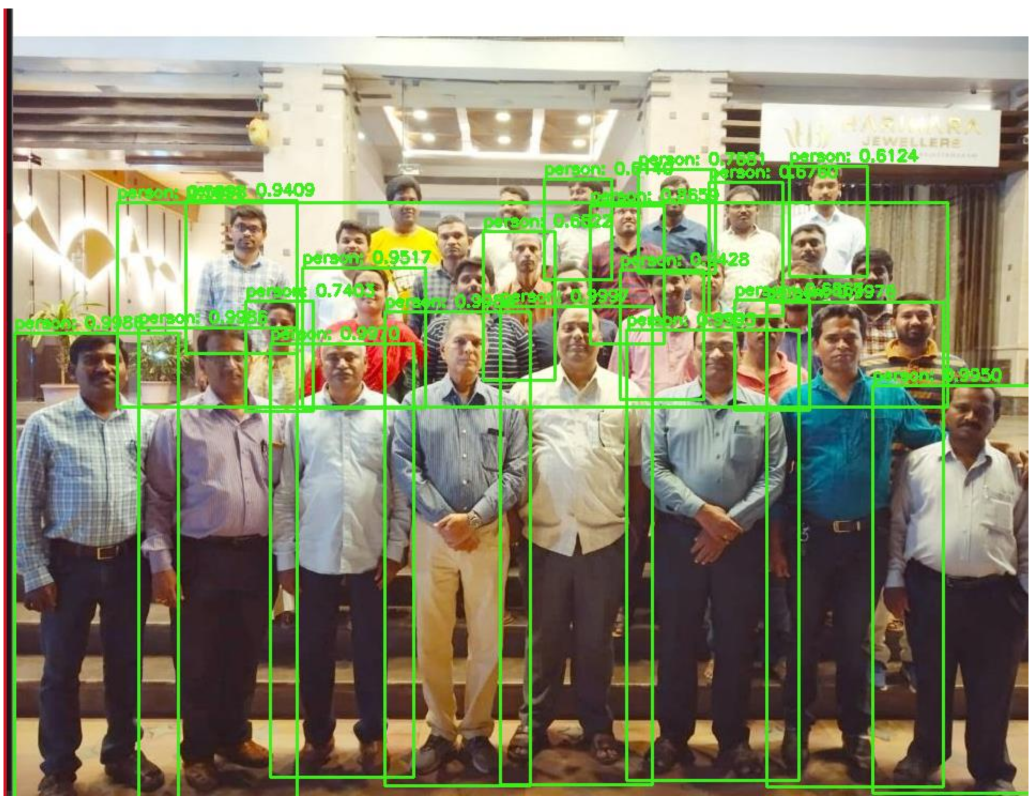


Figure 3.9.1

Figure 3.9.2

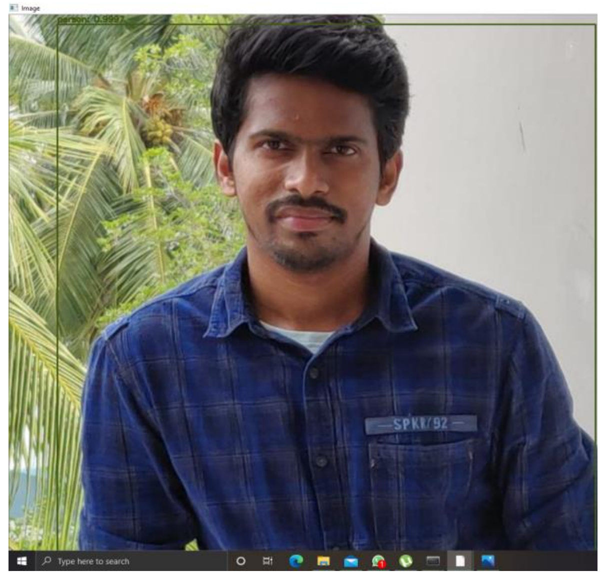


Figure 3.9.3

Figure 3.9.4

Figure 3.9.5

Figure 3.9.6

Figure 3.9.7

Figure 3.9.8

Figure 3.9.9

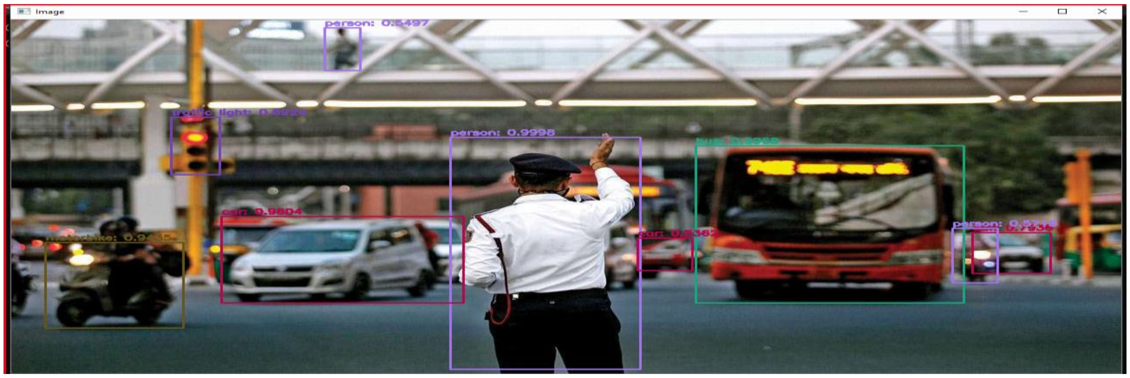


Figure 3.9.10

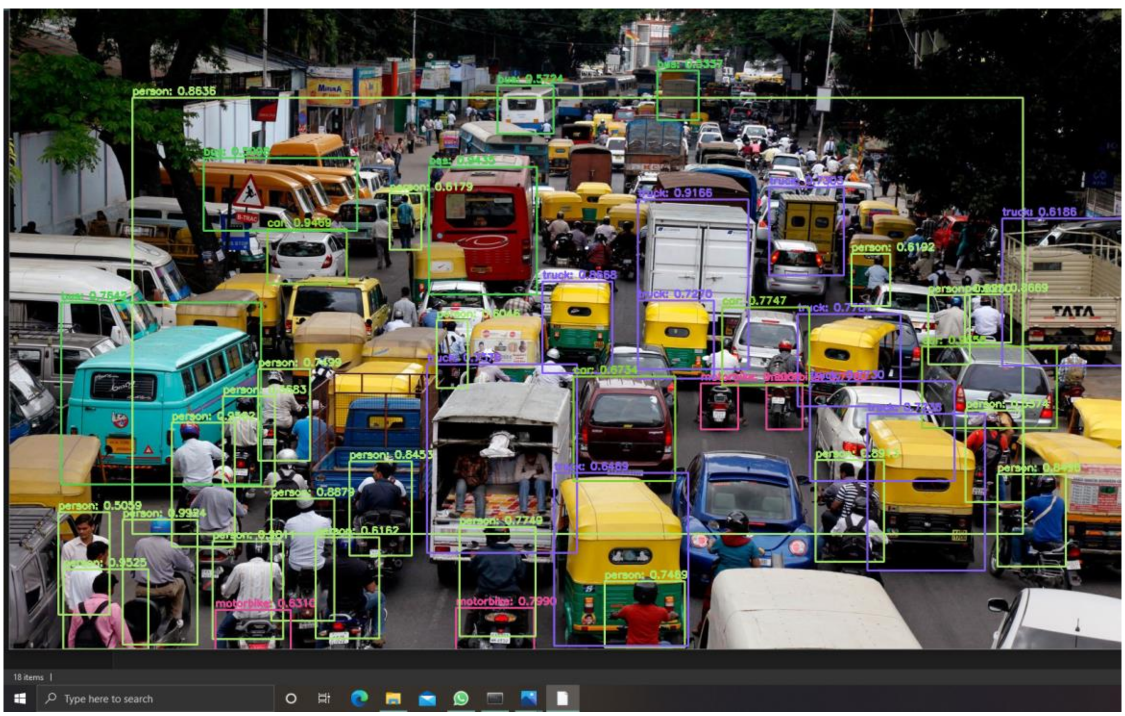


Figure 3.9.11

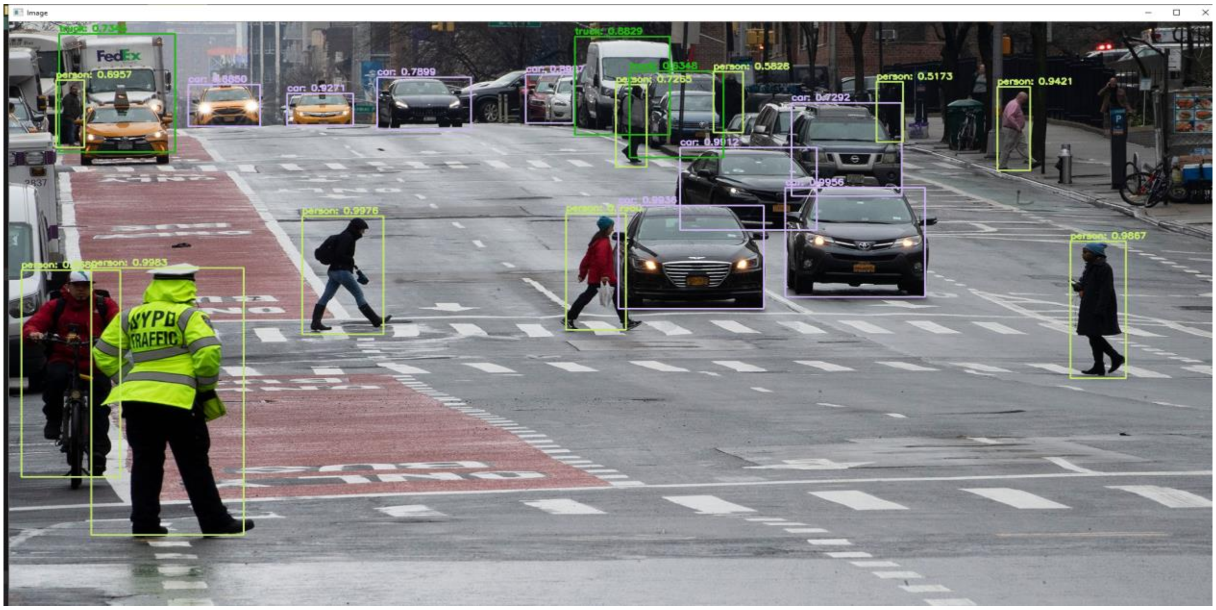


Figure 3.9.12

Figure 3.9.13

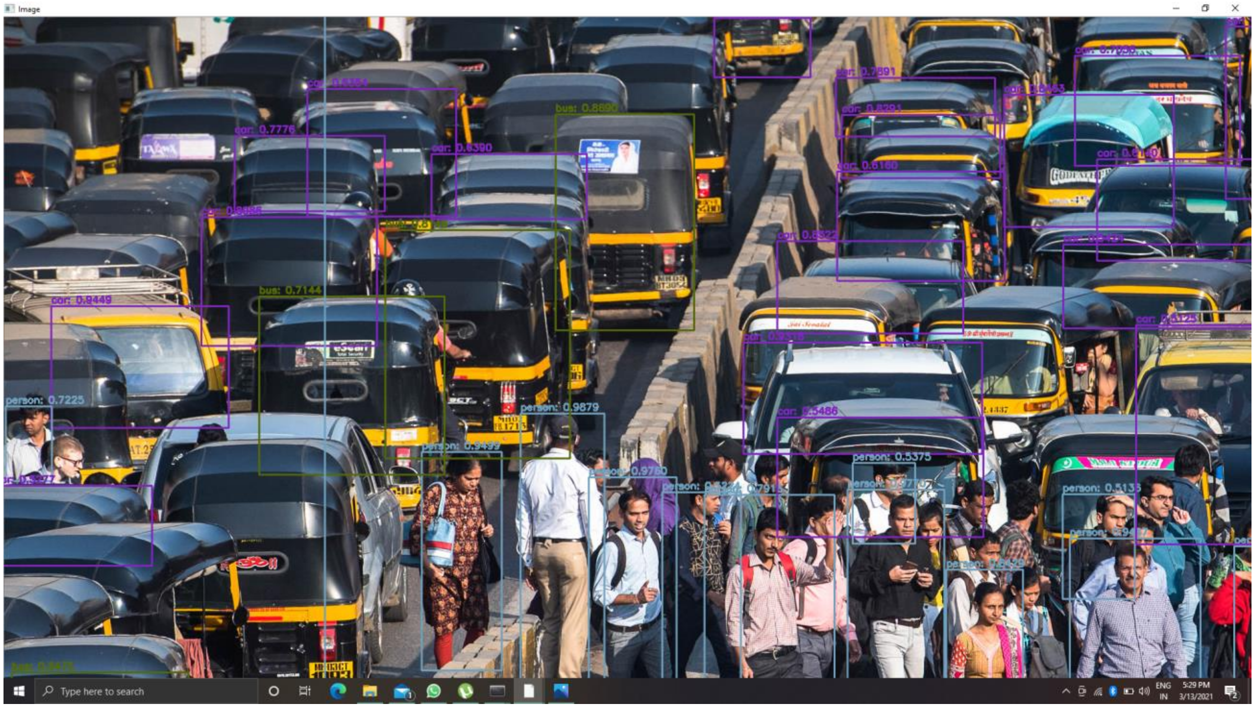


Figure 3.9.14

Figure 3.9.15

Figure 3.9.16

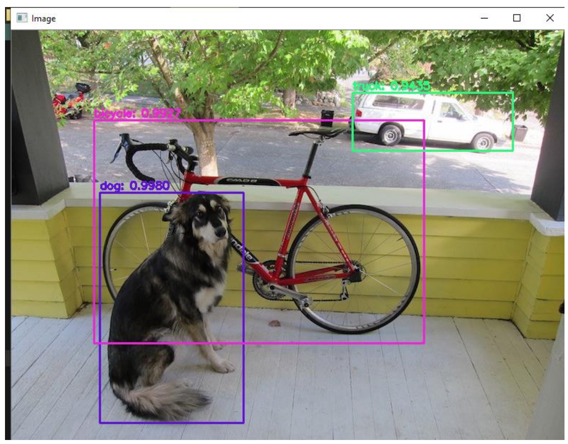


Figure 3.9.17

Figure 3.9.18

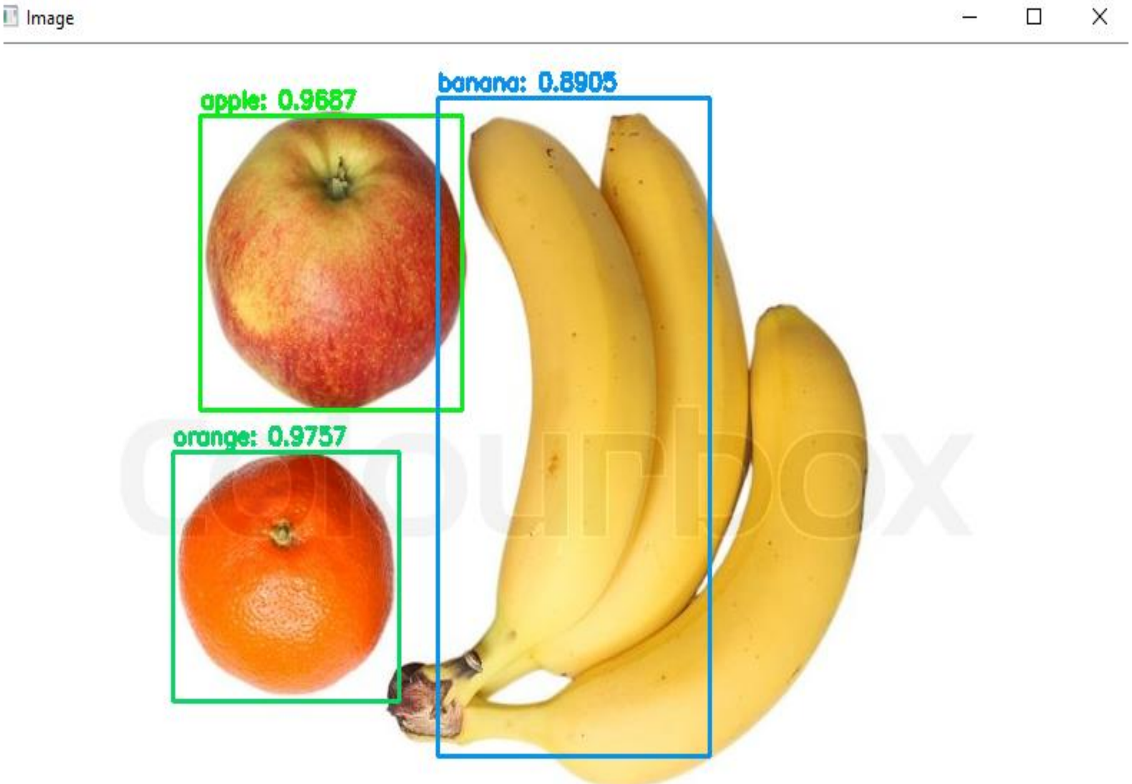


Figure 3.9.19

Figure 3.9.20

Applications of YOLO algorithm:

- Detection of Objects plays crucial task for the various real-world applications such as surveillance, security, and automated vehicle system and that even includes counting of the number of peoples at any junction.
  And other applications are:
- Self-driving vehicles (braking distance).
- Tracking objects
- Counting people
- Person detection
- Animal detection
- CCTV surveillance
- Vehicle detection
- Human computer interactions in robotics
  Face detection......etc.!!

- Vehicle recognition: Different sort of vehicle for example Vehicle, truck, bicycle, transport, train, boat, bike, flights all are distinguished by Yolo model in a picture and continuous both. At the point when any of the above vehicle is distinguished, a bouncing box is made around that vehicle and likelihood of identification is likewise appeared. Kind of vehicle is additionally appeared over the jumping box.
- Creature identification: We may utilize Yolo model for various sorts of creature discovery in the woodland. Yolo model is equipped for identifying horse, sheep, cow, elephant, bear and zebra, giraffe from pictures and continuous camera feed and recorded recordings. It is additionally fit for identifying feline, canine and bird.
- Individual discovery: Detecting people can be a critical application across various enterprises. Typical use cases fuse security applications that track who's voyaging all over, who's traveling every which way similarly as wellbeing frameworks expected to keep people out of harm's way. In PC vision, we use a strategy called object-recognition to distinguish the presence of people in an image. A large part of the time, people are just something single an article location model is good for identifying. Similarly, this procedure fluctuates facial acknowledgment in that it doesn't distinguish a specific individual, yet recognizes when a human is in the casing.

- Item identification: Object recognition is the method of finding and describing a variable number of articles on an image. The huge distinction is the "variable" part. Alternately with issues like arrangement, the yield of item recognition is variable long, since the number of articles recognized may change from one picture to another. Using Yolo model we can recognize different articles, for instance – traffic light, fire hydrant, stop sign, stopping meter, seat, baggage, umbrella, satchel, tie, pack, snowboard, sports ball , kite, debris, glove, skateboard, tennis racket, bottle, wine glass, cup, fork, sharp edge, spoon, bowl, TV screen, etc along these lines various items.
- Face acknowledgment: Face identification is perhaps the most difficult issues of example acknowledgment. Different face related applications like face confirmation, facial acknowledgment, grouping of face and so on are a piece of face location. Successful preparing should be completed for location and acknowledgment. The precision in face recognition utilizing the customary methodology didn't yield a decent outcome. The proposed model uses the convolutional neural organization as a methodology of profound learning for recognizing faces.
- Vehicle discovery: Intelligent vehicle recognition and including are getting progressively significant in the field of parkway the executives. In any case, because of the various sizes of vehicles, their discovery stays a test that straightforwardly influences the precision of vehicle tallies.
- Human PC communications in advanced mechanics: Human activity acknowledgment in video examination has been broadly concentrated as of late. However, the greater part of these strategies relegates a solitary activity name to video after either breaking down a total video or utilizing classifier for each casing. In any case, when contrasted with human vision procedure, it very well may be concluded that we (human) require only an example of visual information for acknowledgment of scene. Incidentally, little gathering of edges or even single edge from the video are sufficient for exact acknowledgment. Here, we present a way to deal with identify, limit and perceive activities of interest in practically constant from outlines got by a consistent transfer of video information that can be caught from a reconnaissance camera. The model takes input outlines after a predetermined period and can give activity name dependent on a solitary edge. Joining results throughout explicit time we anticipated the activity name for the flood of video. We show that YOLO is powerful strategy and similarly quick for acknowledgment and confinement.

BRAKING DISTANCE

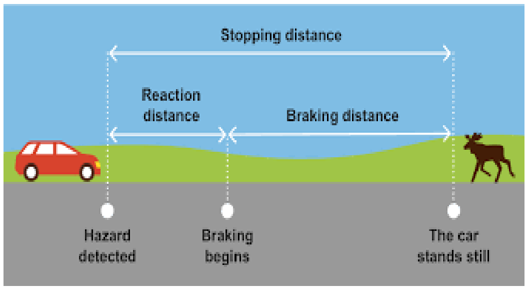


Figure 3.10.1 Figure 3.10.2

Braking Distance Refers to The Distance A Vehicle Will Travel From The Point When It's Brakes Are Fully Applied To When It Comes To A Complete Stop.

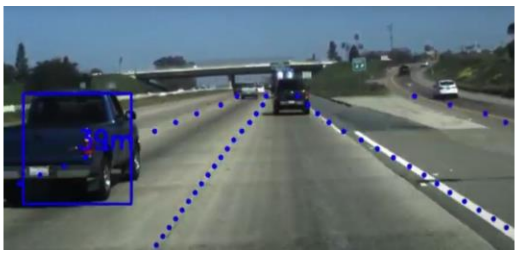


Figure 3.10.3 Figure 3.10.4

RADAR

RADAR stands for Radio Detection and Ranging System. It is basically an electromagnetic system used to detect the location and distance of an object from the point where the RADAR is placed. It works by radiating energy into space and monitoring the echo or reflected signal from the objects.

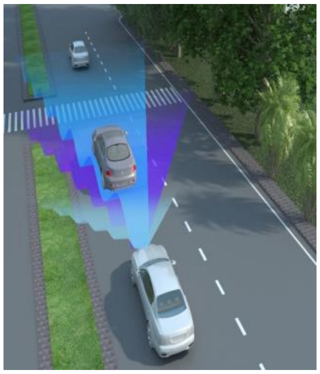


Figure 3.11.1

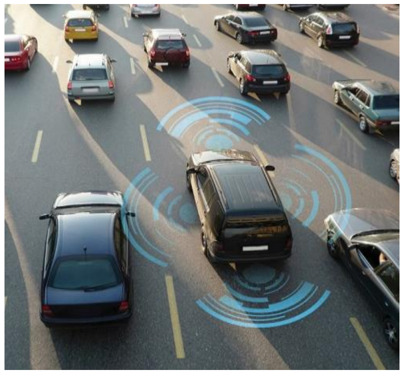


Figure 3.11.2

Figure 3.11.3

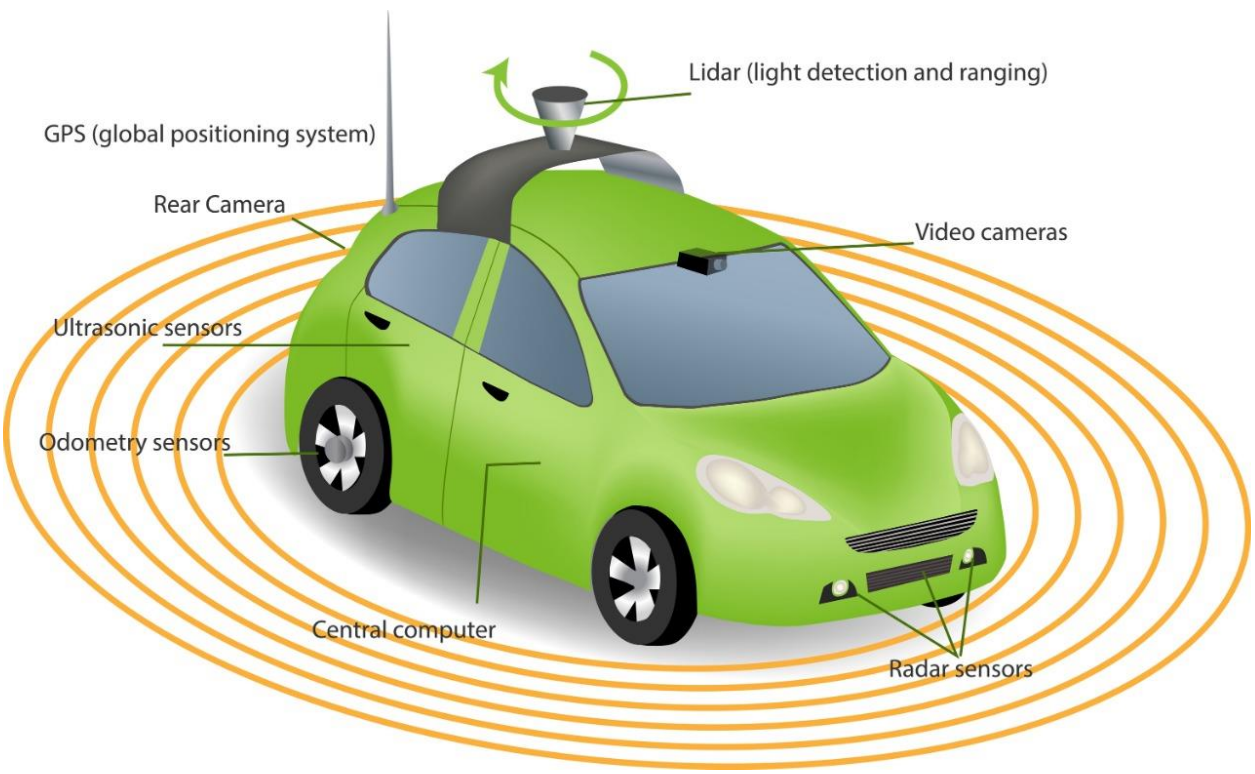


Figure 3.11.4

ADVANTAGES AND DISADVANTAGES

ADVANTAGES:

- The greatest benefit of YOLO is its eminent speed it is so quick and can deal with 45 edges.
- Yolo comprehends summed up object portrayal.
- This is probably the best calculation for object identification and has shown a relatively comparative execution to the R-CNN calculations.

DISADVANTAGES:

- To some degree moderate review and more localization blunder contrasted with quicker R-CNN.
- Battles to identify close items in light of the fact that every matrix can propose just 2 bouncing boxes.
- Battles to identify little articles.

IMPLEMENTATION OF YOLO ALGORITHM

D:\YOLO\yolo.py • - Sublime Text (UNREGISTERED)

File Edit Selection Find View Goto Tools Project Preferences Help

yolo.py

```python
import numpy as np
import argparse
import time
import cv2
import os
# construct the argument parse and parse the arguments
ap = argparse.ArgumentParser()
ap.add_argument("-i", "--image", required=True,
    help="path to input image")
ap.add_argument("-c", "--confidence", type=float, default=0.5,
    help="minimum probability to filter weak detections, IoU threshold")
ap.add_argument("-t", "--threshold", type=float, default=0.3,
    help="threshold when applying non-maxima suppression")
args = vars(ap.parse_args())

# load the COCO class labels our YOLO model was trained on
labelsPath = 'yolo-coco\\coco.names'
LABELS = open(labelsPath).read().strip().split("\n")

# initialize a list of colors to represent each possible class label
COLORS = np.random.randint(0, 255, size=(len(LABELS), 3),
    dtype="uint8")

# paths to the YOLO weights and model configuration
weightsPath = 'yolo-coco\\yolov3.weights'
configPath = 'yolo-coco\\yolov3.cfg'

# load our YOLO object detector trained on COCO dataset (80 classes)
net = cv2.dnn.readNetFromDarknet(configPath, weightsPath)

# load our input image and grab its spatial dimensions
image = cv2.imread(args["image"])
(H, W) = image.shape[:2]
```

Line 104, Column 27 Tab Size: 4 Python

Type here to search ENG 17:56 22-04-2021

D:\YOLO\yolo.py • - Sublime Text (UNREGISTERED)

File Edit Selection Find View Goto Tools Project Preferences Help

yolo.py

```python
(H, W) = image.shape[:2]

# determine only the *output* layer names that we need from YOLO
ln = net.getLayerNames()
ln = [ln[i[0] - 1] for i in net.getUnconnectedOutLayers()]

# construct a blob from the input image and then perform a forward
# pass of the YOLO object detector, giving us our bounding boxes and
# associated probabilities
blob = cv2.dnn.blobFromImage(image, 1 / 255.0, (416, 416),
    swapRB=True, crop=False)
net.setInput(blob)
layerOutputs = net.forward(ln)

# initialize our lists of detected bounding boxes, confidences, and
# class IDs, respectively
boxes = []
confidences = []
classIDs = []

# loop over each of the layer outputs
for output in layerOutputs:
    # loop over each of the detections
    for detection in output:
        # extract the class ID and confidence (i.e., probability) of
        # the current object detection
        scores = detection[5:]
        classID = np.argmax(scores)
        confidence = scores[classID]

        # filter out weak predictions by ensuring the detected
        # probability is greater than the minimum probability
        if confidence > args["confidence"]:
```

Line 104, Column 27 Tab Size: 4 Python

Type here to search ENG 17:57 22-04-2021

D:\YOLO\yolo.py • - Sublime Text (UNREGISTERED)

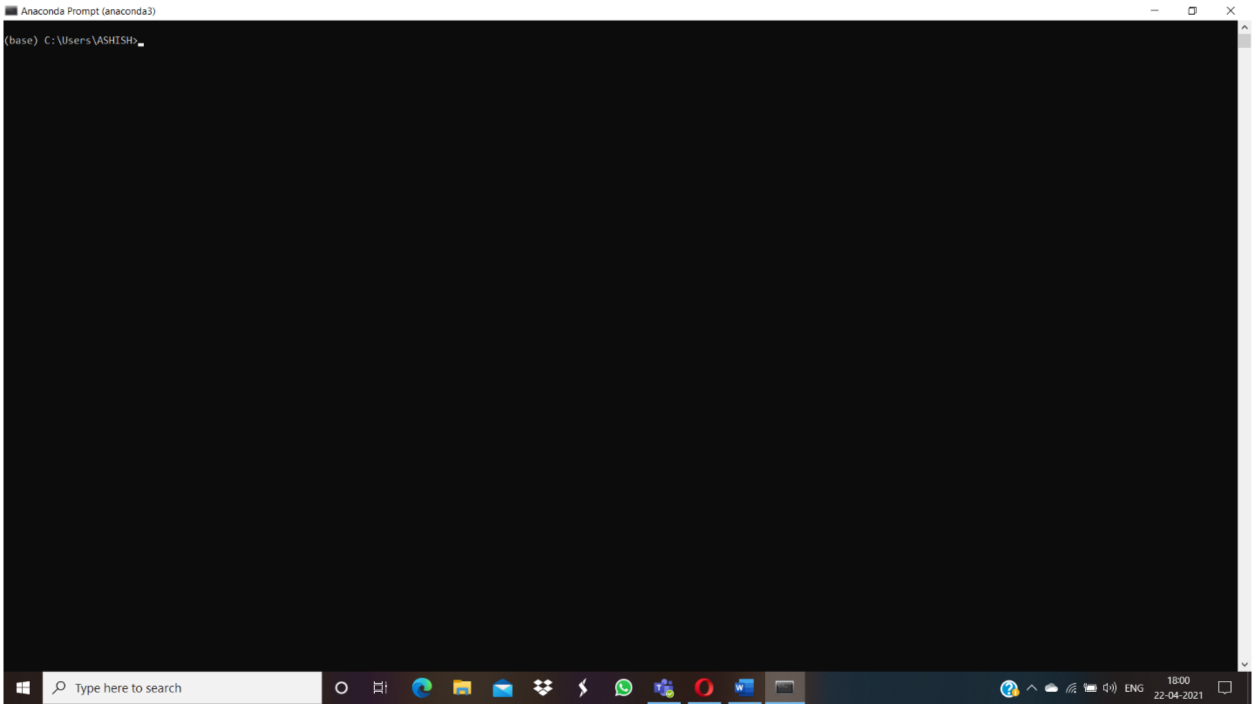
```python
        # filter out weak predictions by ensuring the detected
        # probability is greater than the minimum probability
        if confidence > args["confidence"]:
            # scale the bounding box coordinates back relative to the
            # size of the image, keeping in mind that YOLO actually
            # returns the center (x, y)-coordinates of the bounding
            # box followed by the boxes' width and height
            box = detection[0:4] * np.array([W, H, W, H])
            (centerX, centerY, width, height) = box.astype("int")

            # use the center (x, y)-coordinates to derive the top and
            # and left corner of the bounding box
            x = int(centerX - (width / 2))
            y = int(centerY - (height / 2))

            # update our list of bounding box coordinates, confidences,
            # and class IDs
            boxes.append([x, y, int(width), int(height)])
            confidences.append(float(confidence))
            classIDs.append(classID)

# apply non-maxima suppression to suppress weak, overlapping bounding boxes
idxs = cv2.dnn.NMSBoxes(boxes, confidences, args["confidence"],
    args["threshold"])

# ensure at least one detection exists
if len(idxs) > 0:
    # loop over the indexes we are keeping
    for i in idxs.flatten():
        # extract the bounding box coordinates
        (x, y) = (boxes[i][0], boxes[i][1])
        (w, h) = (boxes[i][2], boxes[i][3])

```

D:\YOLO\yolo.py • - Sublime Text (UNREGISTERED)

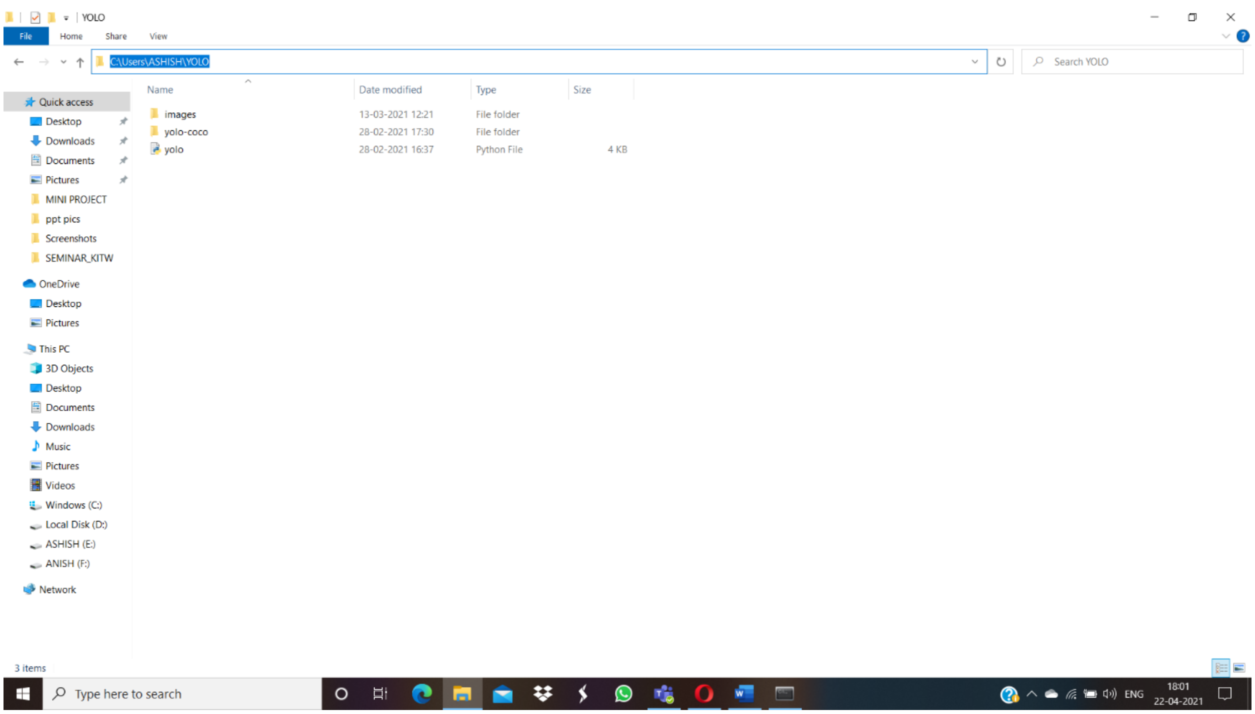
```python
# apply non-maxima suppression to suppress weak, overlapping bounding boxes
idxs = cv2.dnn.NMSBoxes(boxes, confidences, args["confidence"],
    args["threshold"])

# ensure at least one detection exists
if len(idxs) > 0:
    # loop over the indexes we are keeping
    for i in idxs.flatten():
        # extract the bounding box coordinates
        (x, y) = (boxes[i][0], boxes[i][1])
        (w, h) = (boxes[i][2], boxes[i][3])

        # draw a bounding box rectangle and label on the image
        color = [int(c) for c in COLORS[classIDs[i]]]
        cv2.rectangle(image, (x, y), (x + w, y + h), color, 2)
        text = "{}: {:.4f}".format(LABELS[classIDs[i]], confidences[i])
        cv2.putText(image, text, (x, y - 5), cv2.FONT_HERSHEY_SIMPLEX,
            0.5, color, 2)

# show the output image
cv2.imshow("Image", image)
cv2.waitKey(0)
```

HOW TO RUN YOLO ALGORITHM...?

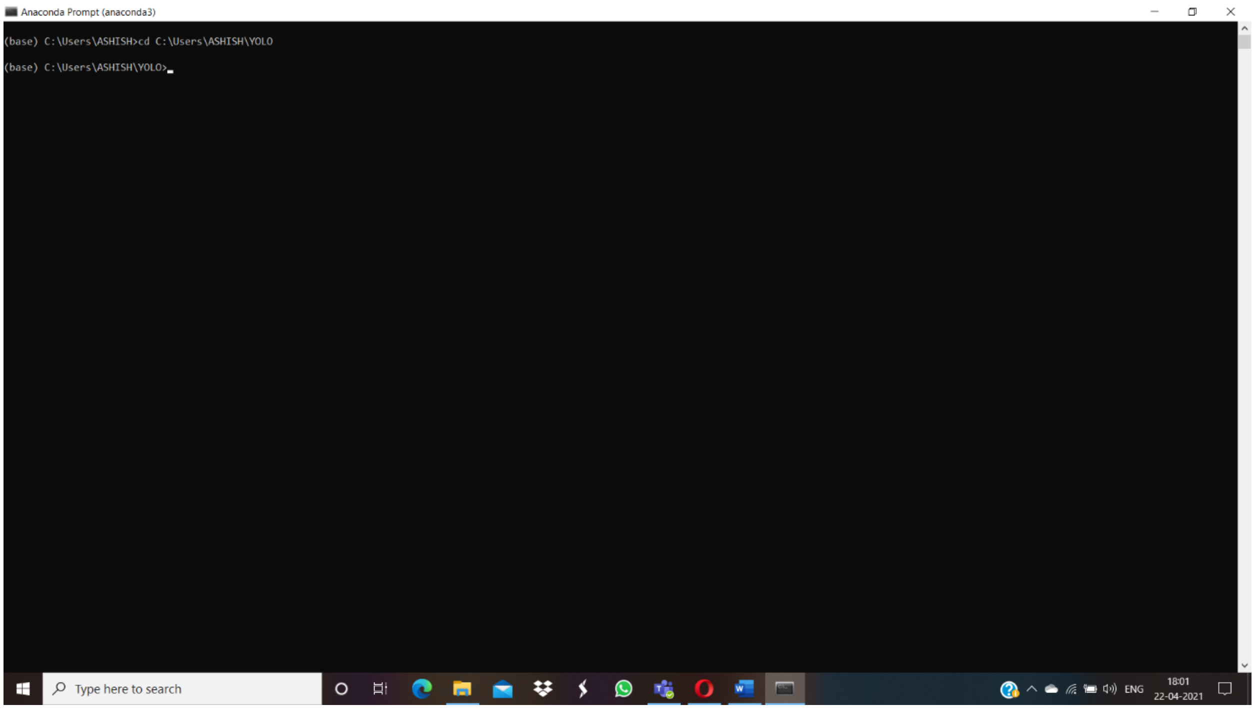
- Open anaconda prompt.



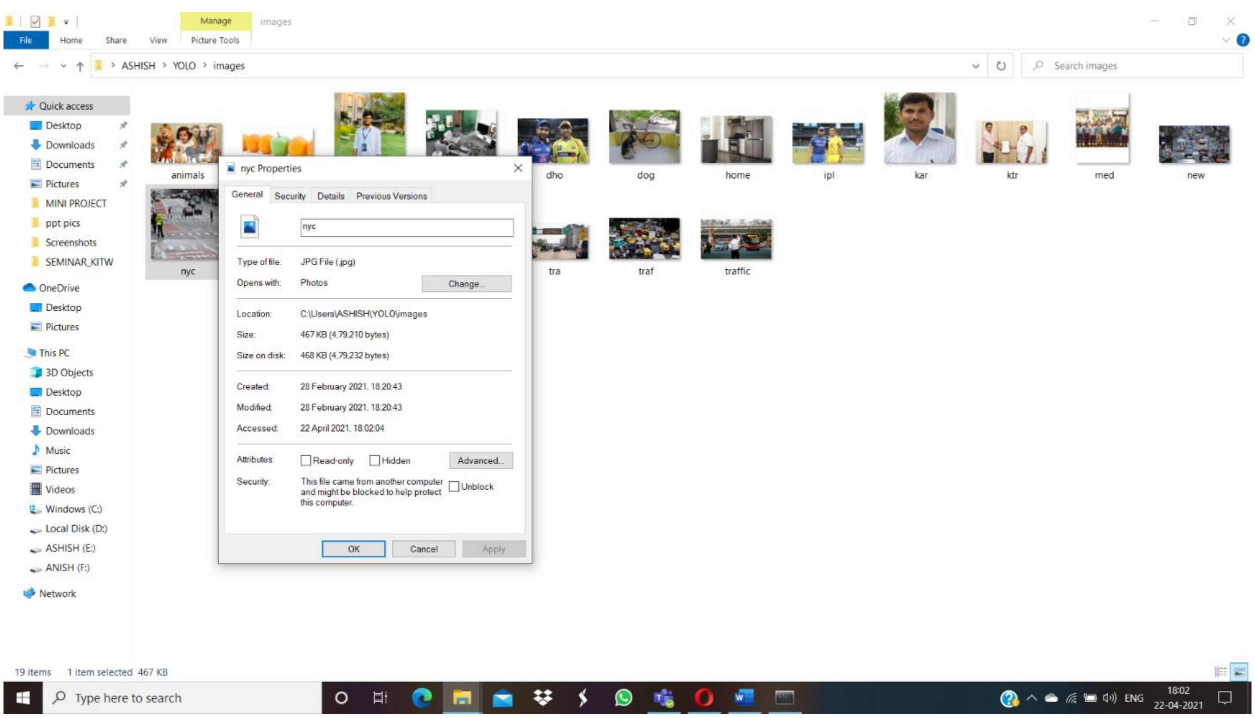
- Select the path of the folder.



- Paste the selected path in the anaconda prompt with Pre-fix “cd”



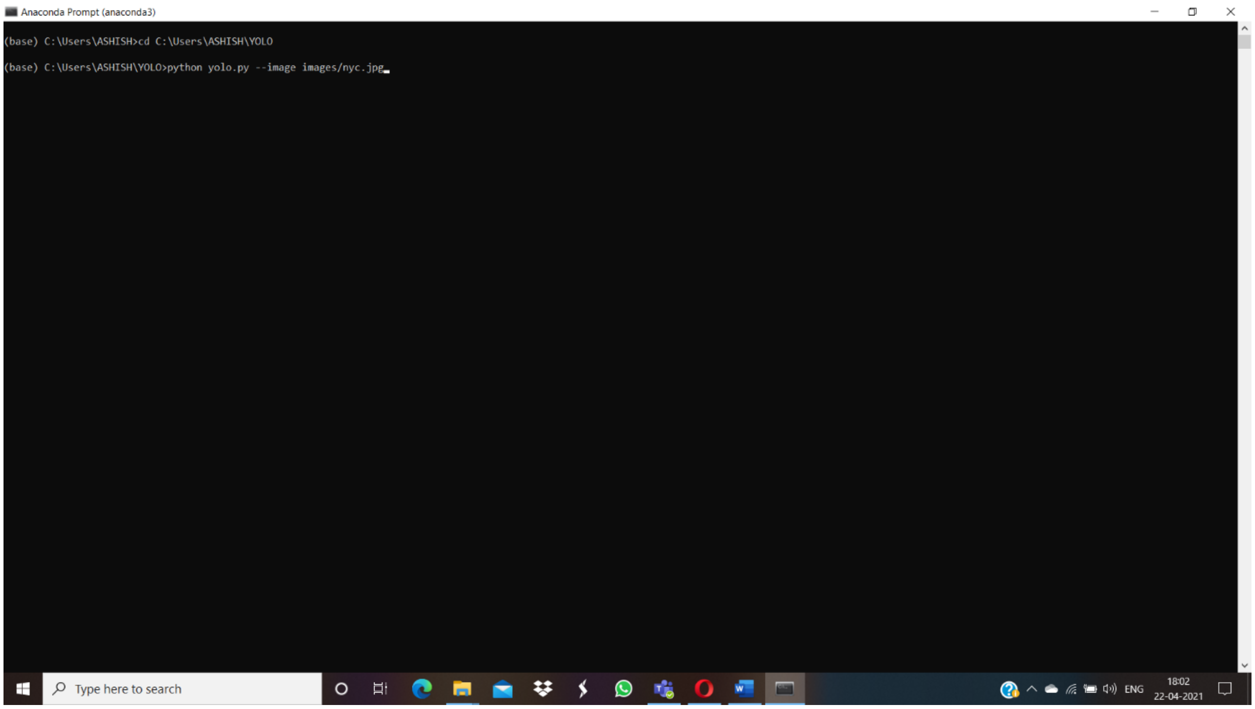
- **From the Properties of your image which you wanted to scan through YOLO, Get the image name and along with the file type with prefix “.” (i.e. img.jpeg)**



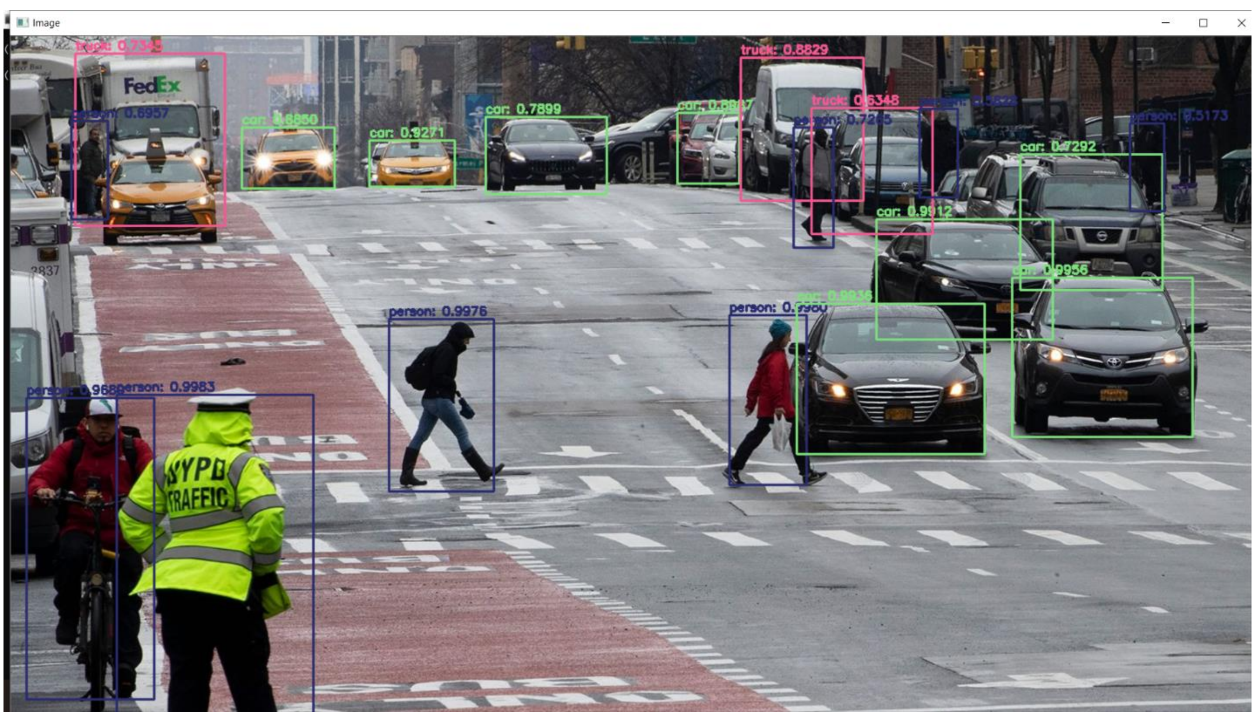
* Now paste the image name and file type in the prompt by prefix command:

  Python yolo.py - -image images/nyc.jpg

  Here nyc is my image name and “.jpg” is my image type.



* We will get output by just clicking “Enter” key.



4. METHODOLOGY/OVERVIEW

Object location is a critical capacity for most PC and robot vision frameworks. YOLO (YOU ONLY LOOK ONCE)- is a clever convolutional neural organization (CNN) and is perhaps the most worth-while calculations which is utilized for recognition of articles progressively applications like Driverless vehicles. YOLO is so quick and has become standard method of recognizing objects in the field of PC vision. Already for object discovery we utilized sliding window object recognition and afterward more quicker forms were created like R CNN, Fast R CNN, Faster R CNN, yet in 2015, yolo was imagined which beat all the past object identification calculations. Item discovery has seen various changes in estimations to improve execution both on speed and exactness. By the steady effort of such innumerable subject matter experts, significant learning computations are growing rapidly with an improved thing distinguishing proof execution. Distinctive renowned applications like walker area, clinical imaging, progressed mechanics, self-driving vehicles, face recognizable proof, etc lessens the undertakings of individuals in various spaces.

5. RESULTS AND DISCUSSION

YOLO is mainstream since it accomplishes high exactness while additionally having the option to run progressively. The calculation "just takes a gander" at the picture as in it requires just one forward proliferation go through the neural organization to make expectations. After non-max concealment (which ensures the article discovery calculation just identifies each item once), it at that point yields perceived articles along with the bouncing boxes. With YOLO, a solitary CNN all the while predicts different jumping boxes and class probabilities for those crates. YOLO trains on full pictures and straightforwardly advances identification execution. This model has various advantages over other item recognition techniques:

- YOLO is incredibly quick
- YOLO sees the whole picture during preparing and test time so it verifiably encodes relevant data about classes just as their appearance.
- YOLO learns generalizable portrayals of articles so when prepared on regular pictures and tried on fine art, the calculation beats other top recognition strategies. should survey the YOLO (You Only Look Once) ongoing article identification calculation, which is perhaps the best item recognition calculations that additionally includes a large number of the most creative thoughts emerging from the PC vision research local area. Article location is a basic capacity of self-governing vehicle innovation. It's a space of PC vision that is detonating and working such a great deal better compared to only a couple years prior. Toward the finish of this article, we'll several new updates to YOLO by the first scientists of this significant procedure. Contrasted with other locale proposition order organizations (quick RCNN) which perform recognition on different district recommendations and consequently wind up performing forecast on numerous occasions for different areas in a picture, Yolo engineering is more similar to FCNN (completely convolutional neural organization) and passes the picture (nxn) once through the FCNN and yield is (mxm) expectation. This the design is parting the info picture in mxm lattice and for every matrix age 2 bouncing boxes and class probabilities for those jumping boxes.

6. <u>CONCLUSION</u>

Automation is the creation and utilization of advances to create and convey labor and products with insignificant human intercession. YOLO is so quick and has become Standard method of distinguishing Objects in the field of Computer Vision. YOLO algorithm is an algorithm based on relapse, rather than choosing the fascinating piece of an Image, it predicts classes and bouncing boxes for the entire picture in one run of the Algorithm.

YOLO is so quick and has become Standard method of recognizing Objects in the field of Computer Vision. YOLO calculation is so productive and is so helpful in the cutting-edge Technological World. One of its principal application is Self-Driving Cars, where it distinguishes the impediments and here by deals with the versatile voyage control.

Contrasting with other classifier Algorithms, This Algorithm is considerably more proficient and quickest calculation to use in Real Time.

7. REFERENCES

1.Shenghui Wang, Leilei Niu “Research on Image Recognition of Insulators Based on YOLO Algorithm "North China Electric Power University, Baoding, China.

2. V. G. Maltarollo, K. M. Honório, and A. B. F. da Silva, “Applications of artificial neural networks in chemical problems,” Artificial neural networks-architectures and applications.

3. C. C. Nguyen, G. S. Tran, T. P. Nghiem, N. Q. Doan, D. Gratadour, J. C. Burie, and C. M. Luong, “Towards real-time smile detection based on faster region convolutional neural network,” in 2018 1st International Conference on Multimedia Analysis and Pattern Recognition (MAPR). IEEE, 2018.

4. Z. Deng, H. Sun, S. Zhou, J. Zhao, L. Lei, and H. Zou, “Multi-scale object detection in remote sensing imagery with convolutional neural networks,” ISPRS journal of photogrammetry and remote sensing.

5. G. Lewis, “Object detection for autonomous vehicles,” 2014.

6. C. M. Bishop, Pattern recognition and machine learning. springer, 2006.

7. H. Kuseto gullari and T. Celik, “Real time vehicle detection and driver warning system using computer vision,” 2010.

8. F. G. YaŞar and H. KusetoĞullari, “Underwater human body detection using computer vision algorithms,” in 2018 26th Signal Processing and Communications Applications Conference (SIU). IEEE, 2018.

9. H. Kusetogullari, H. Demirel, T. Celik, and S. Bayindir, “Real time detection and tracking of vehicles for speed measurement and license plate detection,” in The Seventh IASTED International Conference on Visualization, Imaging and Image Processing.