“Proactive framework for object detection and navigation for visually impaired”

Project report submitted for

4th Semester Minor Project-1

in

Department of Computer Science and Engineering

By

Shreyan Manher (201000050)

Shreedhar Tiwari (201000049)

Miral Kumar Ratre (201000030)

Under The Guidance of

Dr. Santosh Kumar

Dr. Shyama Prasad Mukherjee

Department of Computer Science and Engineering

International Institute of Information Technology, Naya Raipur

(A Joint Initiative of Govt. of Chhattisgarh and NTPC)

Email: iiitnr@iiitnr.ac.in, Tel: (0771) 2474040, Web: www.iiitnr.ac.in

CERTIFICATE

This is to certify that the project titled “Proactive framework for object detection and danger alert for visually impaired by **Shreedhar Tiwari**, **Shreyan Manher**, and **Miral Kumar Ratre** has been carried out under my/our supervision and that this work has not been submitted elsewhere for a degree/diploma.

(Signature of Guide)

_____

Dr. Santosh Kumar

Assistant Professor

Department of CSE

Dr. SPM IIIT-NR

June 2022

Declaration

We declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Shreyan Manher
Shreedhar Tiwari
Miral Kumar Ratre

Date: 2nd June 2022

Plagiarism Report

Approval Sheet

This project report entitled “Proactive framework for object detection and navigation for visually impaired” by “Shreyan Manher, Shreedhar Tiwari, and Miral Kumar Ratre” is approved for 4th Semester Minor Project-1.

(Signature of Examiner - I)

_____

Name of Examiner -I

(Signature of Examiner - II)

_____

Name of Examiner -II

(Signature of Chair)

_____

Name of Chair

Date: 22/4/22 Place: _____

ABSTRACT

Vision is one of the most important human senses and plays an essential role in human perception of the environment, and one of the most important components of the environment is objects. This paper/project proposes a system based on machine learning that restores the main function of the human visual system which is the identification of surrounding objects. This paper/project is useful to differentiate personal objects, which cannot be recognized with pre-trained recognizers and may lack distinguishing tactile features. This paper proposes a system that will detect every possible day-to-day multiple objects on the other hand prompt voice feedback about the objects around them. The proposed system is developed using the Yolo_v3 algorithm, to get the audio Feedback GTTS (Google Text to Speech), and the python library is used to convert statements into audio speech. Testing of both the algorithms is done on the MS-COCO Dataset which consists of more than 200 K images. This paper proposes a system of distance calculation of objects from the camera and gives an alert if the object is close to the camera. This paper also proposes a navigation model for blind people by forming a grid that will help them in giving the proper direction of turn-based on the presence of the object in that particular grid.

Table of Contents

List of All Figures

CHAPTER 1

1.1 INTRODUCTION

It's a known fact that the estimated number of visually impaired person in the world is about 285 million, approximately equal to 20% of the Indian Population. They suffer regular and constant challenges in Recognizing objects and Navigation especially when they are on their own. They are mostly dependent on someone for even accessing their basic day-to-day needs. So, it's a quite challenging task and the technological solution for them is of utmost importance and much needed. One such try from our side is that we came up with an Integrated Machine Learning System which allows the Blind Victims to identify and classify Real-Time Based Common day-to-day Objects and generate voice feedback and calculate distance which produces warnings whether he/she is very close or far away from the object. The same system can be used for the Obstacle Detection Mechanism

The proposed solution divides the problem into 3 different parts

1. Object detection through photos/videos

2. Real-time object detection

3. Guided Navigation

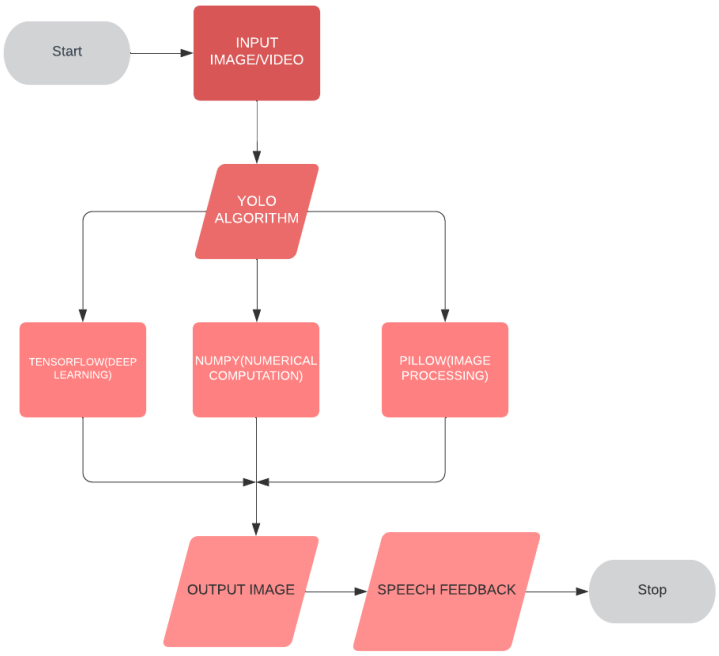


Fig 1.

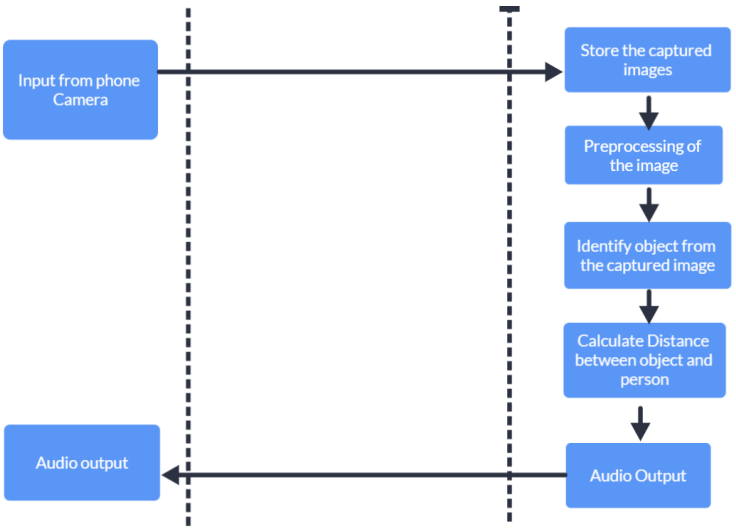

Input from phone Camera
Store the captured images
Preprocessing of the image
Identify object from the captured image
Calculate Distance between object and person
Audio Output
Audio output

Fig 2.

CHAPTER 2

2.1 LITERATURE REVIEW

Much work has been spent developing object detection algorithms that can be used with standard cameras without additional sensors. Deep neural networks are used in state-of-the-art object detection algorithms.

Convolutional neural networks (CNNs) are the main architecture of computer vision. CNNs don't have fully connected layers, but instead have a convolutional layer where filters are convoluted at different parts of the input to produce output. The convolution layer allows you to extract relational patterns from the input. In addition, the filter does not require the weight assigned to each output from each input, so the convolution layer learns less weight than a fully connected layer.

RCNN

Regionalbased convolutional neural networks (RCNN) [11] take region proposals into account when detecting objects in images. A feature vector is extracted from each region proposal and fed into a convolutional neural network. For each subject, Support Vector Machines are used to evaluate the feature vectors (SVM). Despite the high accuracy of RCNN, the model cannot achieve real-time speed even with Fast RCNN [12] and Faster RCNN [13] due to the expensive training process and the inefficiency of the region proposition.

YOLO

You Only Look Once (YOLO) [10] was created to create a one-step detection and classification process. Following a single evaluation of the input image, bounding box and class predictions are made. On GPU-equipped computers, the fastest YOLO architecture can reach 45 FPS, and smaller versions of Tiny YOLO can reach up to 244 FPS (Tiny YOLOv2). YOLO differs from other traditional systems in that it performs bounding boxes and class predictions at the same time. The input image is first split into SS grids. Next, a B bounding box is defined in each grid cell and a confidence value is set for each.

2.2 Related Works

2.2.1 ReCog: Supporting Blind People in Recognizing Personal Objects

2.2.1.1. Problem Statement

To recognize objects by training a deep network with their own photos of such objects.

2.2.1.2. Solution

Machine learning and computer vision trained on large image datasets are used in new assistive technologies to help blind users recognize objects. Mobile applications like Microsoft Seeing AI are trained to recognize common objects (such as cars and mugs) or commercial products (e.g., Cola, Pepsi). General-purpose recognition, on the other hand,

cannot be used for personal items such as clothing, handmade items, local products, or photographs of loved ones.

To address this issue, we developed a mobile app that allows blind users to take photos of personal items and use them to train a deep neural network that can recognize those items. For blind people, aiming the camera to take photos is difficult, resulting in photos of varying quality. Consistent photos, on the other hand, are desirable because they improve recognition accuracy.

2.2.2 Object Detection Based on YOLO Network

2.2.2.1. Problem Statement

Reviewing of Object detection on YOLO Network

2.2.2.2. Solution

They built the models from degraded images, primarily using mathematical models to generate degraded images from standard datasets. These models are then used to train the network to adapt to the complex real-world environment. Finally, the model's ability to generalize complex images is improved. We use the neural network YOLO [14] to analyze traffic signs as our object of study. As a result, they contributed the following:

1) They developed a new image degradation model and tested it on various degraded images. The effect of various gradient images on the standard model is then compared. 1000

2) They modified the source network and ran 1001 different mining processes on the training set, as well as compared the accuracy of test sets on different models. They then ran more complex degradation processes on the training sets to obtain a more general detection network.

3) They optimized the object detection method based on the foregoing. In summary, the model's generalizability and object detection accuracy

2.2.3 An Object Detection System Based on YOLO in Traffic Scene

2.2.3.1. Problem Statement

Reviewing of Object detection on YOLO in Traffic Scene

2.2.3.2. Solution

YOLO, an excellent deep learning-based object detection approach, introduces a novel convolutional neural network for location and classification. The YOLO network's fully connected layers are replaced with an average group layer, resulting in a new network. After optimizing the loss function, the limit coordinate error ratio increases. YOLO (optimized YOLO) is a new object detection method that is 1.18 times faster than YOLO and outperforms other region-based approaches such as RCNN in accuracy. To improve accuracy even further, we combined OYOLO and RFCN in our system. For demanding images at night, the histogram equalization approach is used for pre-processing. In our test set, we achieved a mAP improvement of more than 6%.

We want a real-time object detection algorithm in the traffic scene, so we chose YOLO as a reference to develop a better method. The Convolutional Neural Network, inspired by the FCN (Fully Convolutional Network)[5,] can perform classification without a fully connected layer. An average cluster layer replaces the YOLO network's last two fully connected layers. This results in the formation of a new convolutional neural network. Our method is called OYOLO because it is YOLO-optimized. The test time speed per frame of OYOLO is 18 times that of YOLO.

with a frequency that could improve, as indicated by a review of cases, ranging from once per month to 3 times per week. Anyhow, since there is an increase of individuals with some type of mental problem, it is fundamental to accept the challenge of reducing or in any case slowing the growth of the number. One of the proposed solutions is the use of different patient assistance techniques provided through mobile phones and apps. Such strategies can be used to collect mental health data, and to motivate people to answer questions about what they do (or have done) and what they are experiencing (or have been). undergo) on a daily basis to perform a mental health-related intervention remotely, and to provide access to mental health assets, for example, to initiate communication channels Communicating with mental health professionals. this agreement is developed under the research area known as Mobile Health (mHealth).

But, nowadays with the advancement in technologies and facilities monitoring mental health is not at all a tough job. In this project, we have used natural language processing frameworks such as sentimental analysis and emotional analysis for predicting the mental health of an individual. We have made our own data dictionary for getting a more accurate result that could not be obtained with the available datasets on the internet.

CHAPTER 3

3.1 Object detection

Object detection is a computer vision approach for detecting things in photos and videos. To obtain relevant results, object detection algorithms often use machine learning or deep learning. We can recognize and locate objects of interest in photos or videos in a handful of seconds when we glance at them. The purpose of object detection is to use a computer to imitate this intelligence.

3.2 Object Localization

An image classification or image recognition model simply detects the probability of an object in an image. In contrast to this, object localization refers to identifying the location of an object in the image. An object localization algorithm will output the coordinates of the location of an object with respect to the image. In computer vision, the most popular way to localize an object in an image is to represent its location with the help of bounding boxes. Fig. 1 shows an example of a bounding box.

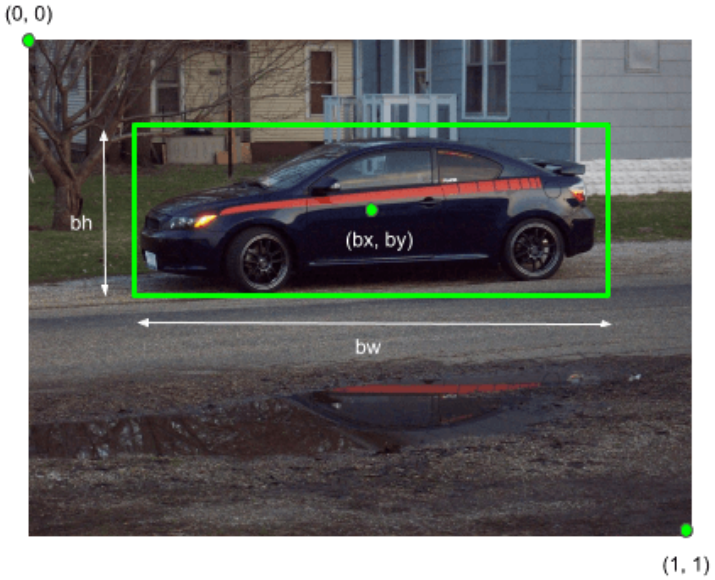


Fig 1. Bounding box representation used for object localization

A bounding box can be initialized using the following parameters:

bx, by: coordinates of the center of the bounding box

by : width of the bounding box w.r.t the image width

bh : height of the bounding box w.r.t the image height

CHAPTER 4

4.1 Algorithms for object detection

- Convolutional implementation of sliding windows
- The YOLO (You Only Look Once) Algorithm
- R-CNN – Region-based Convolutional Neural Networks

Among these, We will be using the Yolo algorithm because it is an algorithm that uses neural networks to give real-time object detection. This algorithm is popular because of its speed and delicacy. It has been used in various applications to detect traffic signals, people, parking meters, and animals.
YOLO is an abbreviation for the term 'you Only Look Once'. This is an algorithm that detects and recognizes various objects in a picture (in real-time). Object detection in YOLO is done as a regression problem and provides the class probabilities of the detected images.

YOLO algorithm employs convolutional neural networks (CNN) to detect objects in real-time. As the name suggests, the algorithm requires only a single forward propagation through a neural network to detect objects.

This means that prediction in the entire image is done in a single algorithm run. The CNN is used to predict various class probabilities and bounding boxes simultaneously.

The YOLO algorithm consists of various variants. Some of the common ones include tiny YOLO and YOLOv3

The idea of YOLO differs from other traditional systems in that bounding box predictions and class predictions are done simultaneousl

How the YOLO algorithm works

YOLO algorithm works using the following three techniques:

- Residual blocks
- Bounding box regression
- Intersection Over Union (IOU)

1)_Residual blocks

First, the image is divided into various grids. Each grid has a dimension of S x S. The following image shows how an input image is divided into grids.

Fig 3.

In the image above, there are many grid cells of equal dimensions. Every grid cell will detect objects that appear within them. For example, if an object center appears within a certain grid cell, then this cell will be responsible for detecting it.

2)_Bounding box regression

A bounding box is an outline that highlights an object in an image.Every bounding box in the image consists of the following attributes:

- Width (bw)
- Height (bh)
- Class (for example, person, car, traffic light, etc.)- This is represented by the letter c.
- Bounding box center (bx,by)

The following image shows an example of a bounding box. The bounding box has been represented by a yellow outline.

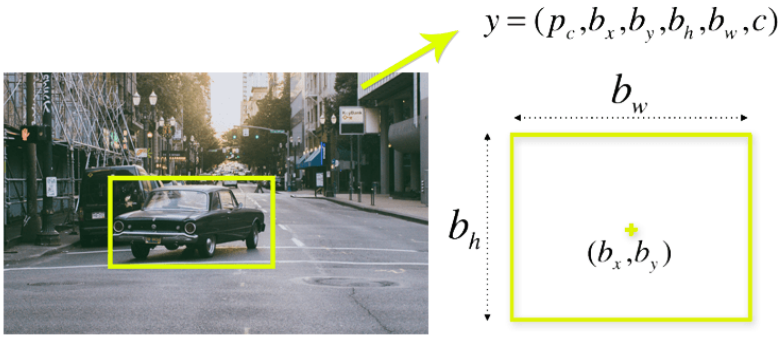
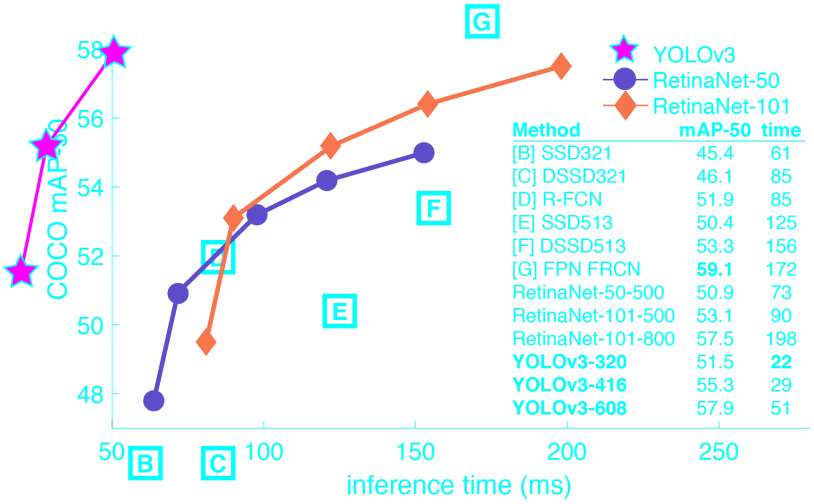


Fig 4.

YOLO uses a single bounding box regression to predict the height, width, center, and class of objects. The image above represents the probability of an object appearing in the bounding box.Intersection over Union (IOU)Intersection over Union (IOU) is a phenomenon in object detection that describes how boxes overlap. YOLO uses IOU to provide an output box that surrounds the objects perfectly.



Method	mAP-50	time
[B] SSD321	45.4	61
[C] DSSD321	46.1	85
[D] R-FCN	51.9	85
[E] SSD513	50.4	125
[F] DSSD513	53.3	156
[G] FPN FRCN	**59.1**	172
RetinaNet-50-500	50.9	73
RetinaNet-101-500	53.1	90
RetinaNet-101-800	57.5	198
YOLOv3-320	51.5	**22**
YOLOv3-416	55.3	29
YOLOv3-608	57.9	51

Each grid cell is responsible for predicting the bounding boxes and their confidence scores. The IOU is equal to 1 if the predicted bounding box is the same as the real box. This mechanism eliminates bounding boxes that are not equal to the real box.The following image provides a simple example of how IOU works.

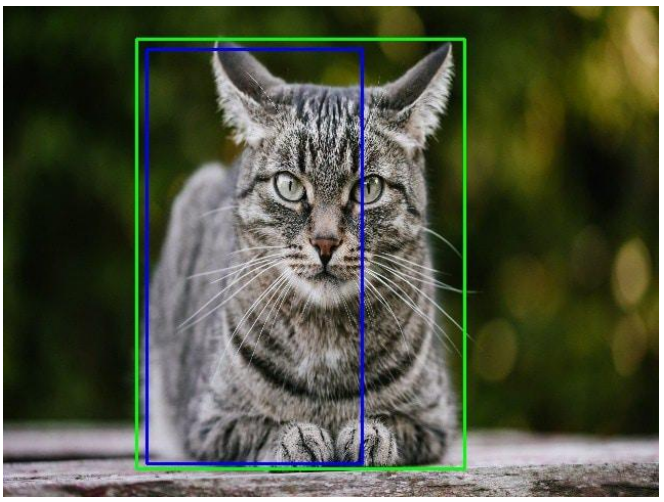


Fig 5.

In the image above, there are two bounding boxes, one in green and the other one in blue. The blue box is the predicted box while the green box is the real box. YOLO ensures that the two bounding boxes are equal.

4.2 Combination of the three techniques

The following image shows how the three techniques are applied to produce the final detection results.

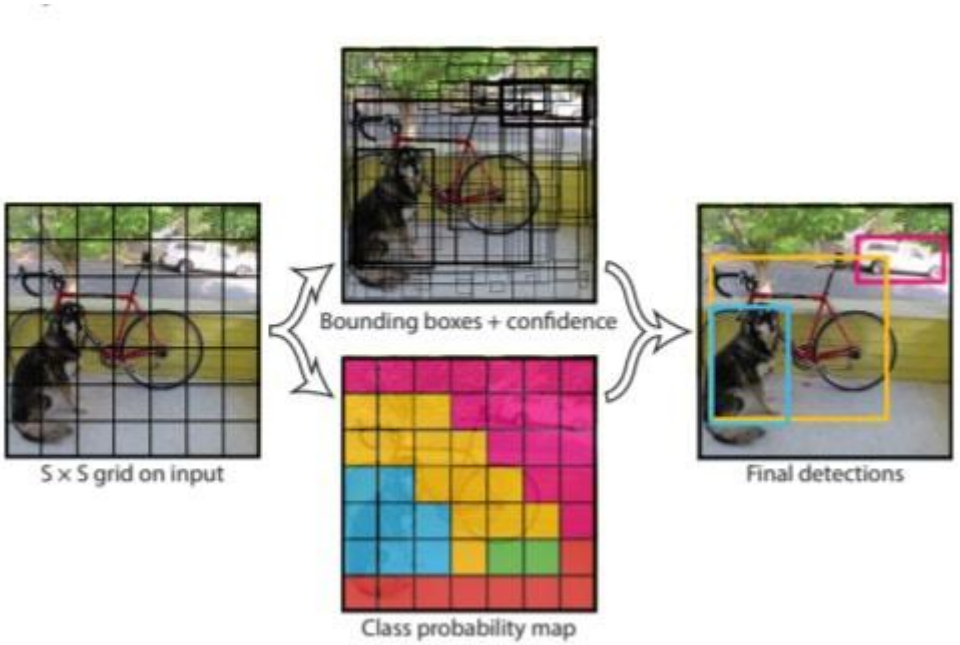
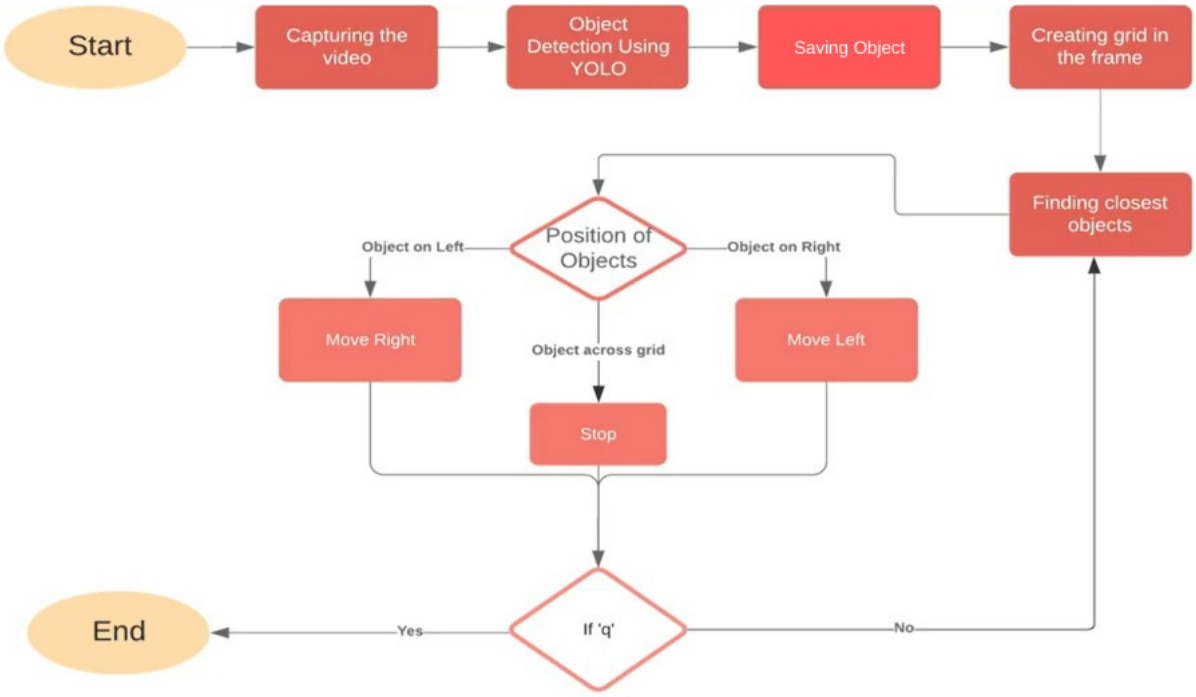


Fig 6

First, the image is divided into grid cells. Each grid cell forecasts B bounding boxes and provides their confidence scores. The cells predict the class probabilities to establish the class of each object.

For example, we can notice at least three classes of objects: a car, a dog, and a bicycle. All the predictions are made simultaneously using a single convolutional neural network.

Intersection over union ensures that the predicted bounding boxes are equal to the real boxes of the objects. This phenomenon eliminates unnecessary bounding boxes that do not meet the characteristics of the objects (like height and width). The final detection will consist of unique bounding boxes that fit the objects perfectly



Guided Navigation flowchart

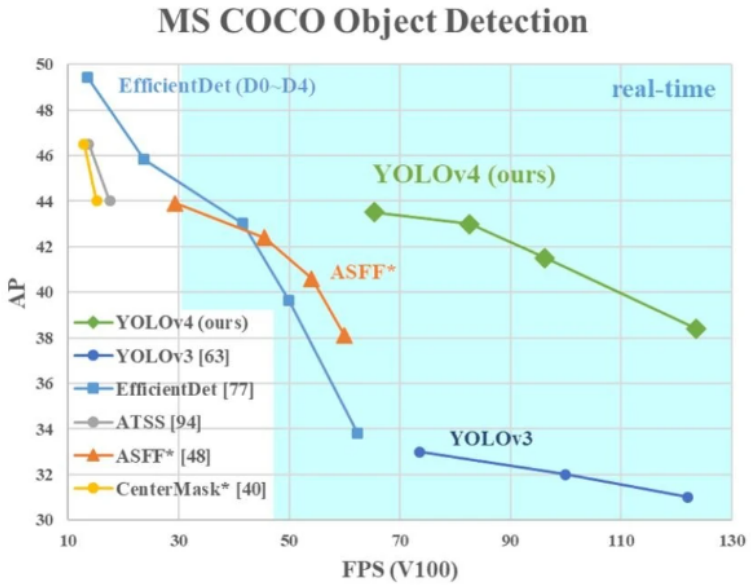
CHAPTER 5
MS COCO DATASET

5.1 Introduction

The MS COCO dataset (Microsoft Common Objects) is a large-scale dataset for object detection, segmentation, key-point detection, and captioning. There are 328K images in the dataset.

In 2014, the first version of the MS COCO dataset was released. It has 164K photos divided into three sets: training (83K), validation (41K), and test (41K). A fresh test set of 81K photographs was released in 2015, which included all of the previous test images as well as 40K new images.

The training/validation split was modified from 83K/41K to 118K/5K in 2017 as a result of community feedback. The same images and annotations are used in the new split. The 2017 test set is a subset of the 2015 test set's 41K photos. A fresh unannotated dataset of 123K photos is included in the 2017 release.



5.2 Features of COCO DATASET

- Object segmentation with detailed instance annotations
- Recognition in context
- Superpixel stuff segmentation
- Over 200’000 images of the total 330’000 images are labeled
- 1.5 Mio object instances
- 80 object categories, the “COCO classes”, which include “things” for which individual instances may be easily labeled (person, car, chair, etc.)
- 91 stuff categories, where “COCO stuff” includes materials and objects with no clear boundaries (sky, street, grass, etc.) that provide significant contextual information.
- 250’000 people with 17 different key points, popularly used for Pose Estimation

5.3 List of the COCO Object Classes

The COCO dataset classes for object detection and tracking include the following pre-trained 80 objects:
'person', 'bicycle', 'car', 'motorcycle', 'airplane', 'bus', 'train', 'truck', 'boat', 'traffic light', 'fire hydrant', 'stop sign', 'parking meter', 'bench', 'bird', 'cat', 'dog', 'horse', 'sheep', 'cow', 'elephant', 'bear', 'zebra', 'giraffe', 'backpack', 'umbrella', 'handbag', 'tie', 'suitcase', 'frisbee', 'skis','snowboard', 'sports ball', 'kite', 'baseball bat', 'baseball glove', 'skateboard', 'surfboard', 'tennis racket', 'bottle', 'wine glass', 'cup', 'fork', 'knife', 'spoon', 'bowl', 'banana', 'apple', 'sandwich', 'orange', 'broccoli', 'carrot', 'hot dog', 'pizza', 'donut', 'cake', 'chair', 'couch', 'potted plant', 'bed', 'dining table', 'toilet', 'tv', 'laptop', 'mouse', 'remote', 'keyboard', 'cell phone', 'microwave', 'oven', 'toaster', 'sink', 'refrigerator', 'book', 'clock', 'vase', 'scissors', 'teddy bear', 'hair drier', 'toothbrush'

5.4 Performance of different models on COCO DATASET

Model	Train	Test	mAP	FLOPS	FPS	Cfg	Weights
SSD300	COCO trainval	test-dev	41.2	-	46	-	link
SSD500	COCO trainval	test-dev	46.5	-	19	-	link
YOLOv2 608x608	COCO trainval	test-dev	48.1	62.94 Bn	40	cfg	weights
Tiny YOLO	COCO trainval	test-dev	23.7	5.41 Bn	244	cfg	weights

Chapter 6

5.1 Distance Calculation and Danger Alert-

Traditionally We used to measure distance with ultrasonic sensors like the HC-sr04 or other high-frequency devices that create sound waves to calculate the distance traveled. When working with an embedded device create a compact design with features like object identification (with camera) and distance measuring.

We don't always want to add unnecessarily heavy hardware components to our device. We might choose a more practical and convenient approach to avoid such situations. We may utilize the depth information that the camera uses to construct the bounding boxes for localizing objects to compute the distance of that object from the camera because we have previously incorporated a camera for object recognition.

5.2 Working-

This formula is used for determining the distance

distance = (2 x 3.14 x 180) ÷ (w + h x 360) x 1000 + 3.

For measuring distance, at first, we have to understand how a camera sees an object.

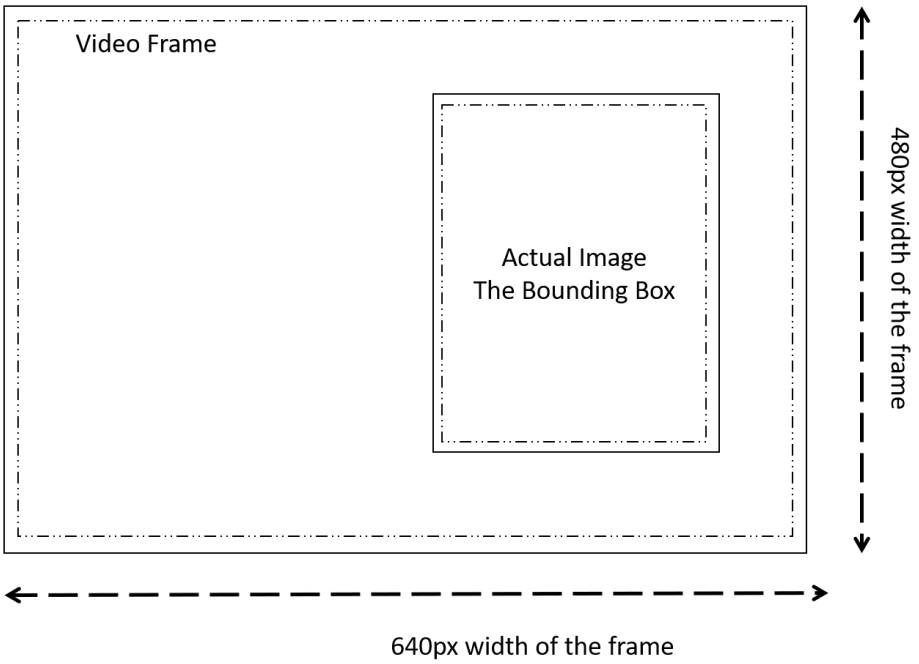


Fig 7.

We'll obtain four digits in the bounding box: (x0,y0, width, height). The bounding box is tiled or adjusted using x0,y0. These two variables are utilized in the calculation for measuring the

object and characterizing the detail of the detected object or objects. The width and height of an object will vary depending on its distance from the camera.

As we know, an image is refracted when it passes through a lens because light may enter the lens, whereas light can reflect in the case of a mirror, which is why we obtain an exact reflection of the picture. But in the case of the lens image gets a little stretched.

The following image illustrates how the image and the corresponding angles look when it enters through a lens.

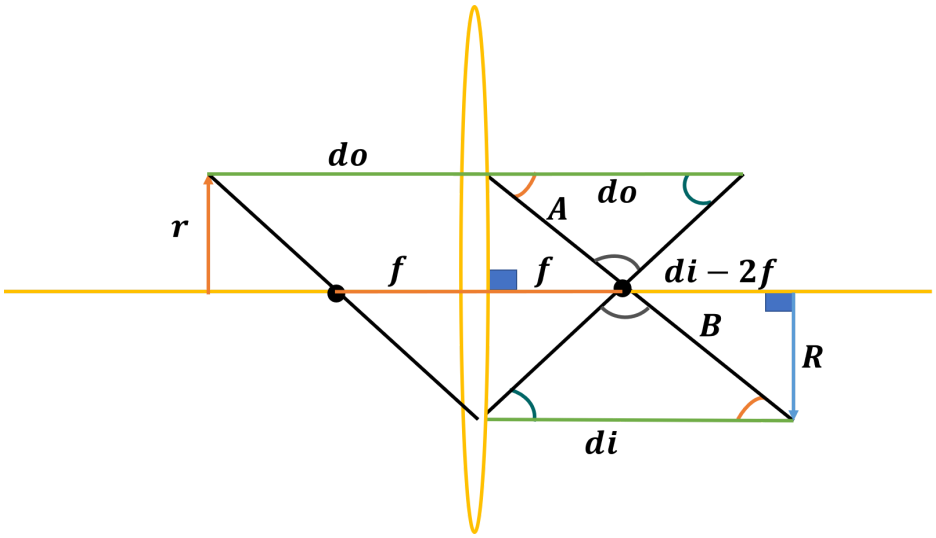


Fig 8.

If we see there are three variables named:

- do (Distance of object from the lens)
- di (Distance of the refracted image from the convex lens)
- f (focal length or focal distance)

5.3 Result-

As a result, the green line "do" indicates the object's actual distance from the convex length. And "di" provides you with an idea of how the image looks in real life. Consider a triangle on the left side of the picture (new refracted image) with the base "do" and draw a triangle on the right side that is comparable to the one on the left. As a result, the new base of the opposing triangle will have the same perpendicular distance as the old base. When we look at the two triangles from the right side, we can see that "do" and "di" are parallel, and the angles created on each side of both triangles are opposing. As a result, we can deduce that both triangles on the right side are comparable.

Because they are similar now, the ratios of the corresponding sides will be similar as well. As a result, do/di equals A/B. If we compare two triangles on the right side of the image where opposite angles are equal and one angle of both triangles is a right angle (90°), we can see that they are similar (dark blue area). As a result, A And B are both hypotenuses of a comparable triangle with a right angle.

As a result, the new equation is:

$$\frac{do}{di} = \frac{A}{B} = \frac{f}{di - f}$$

Now, if we derive from that equation we will find:-

$$\frac{1}{f} = \frac{1}{d_o} + \frac{1}{d_i}$$

And eventually will come to at

$$d = f + \frac{R}{r}$$

Where f is the focal length or also called the arc length by using the following formula

$$f = \frac{2 \times 3.14 \times 180}{360}$$

we will get our final result in "inches" from this formula of distance.

distance = (2 x 3.14 x 180) ÷ (w + h x 360) x 1000 + 3

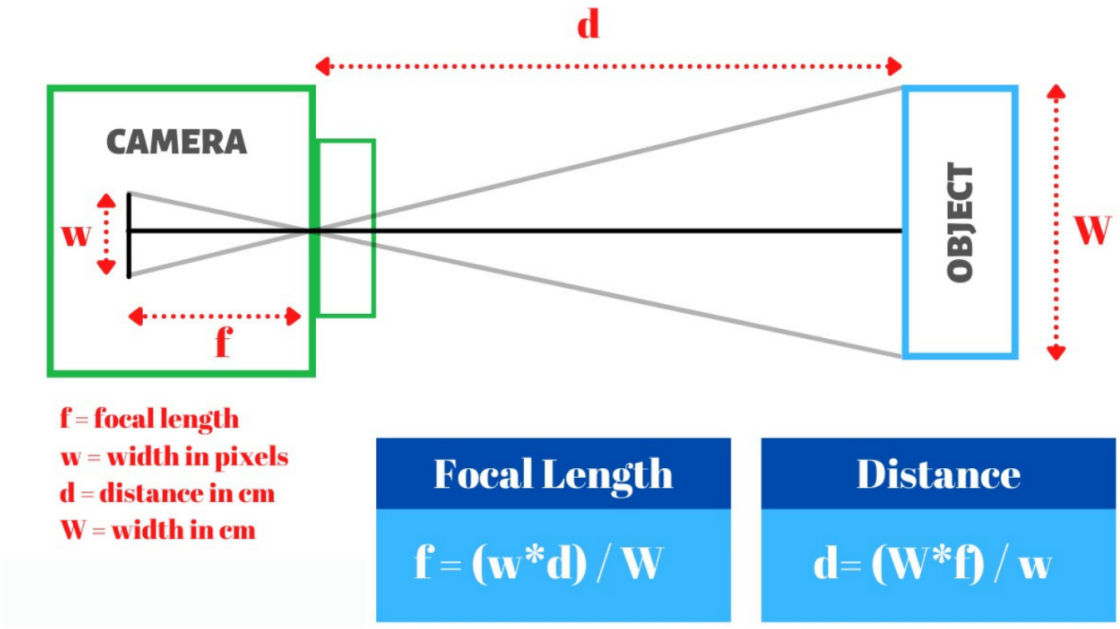


Fig 9.

CHAPTER 7

7.1 Conclusion

We designed a model that allows blind persons to recognize their personal objects, this is accomplished by placing objects in front of a web/mobile camera and using them to train a recognition model. Our model provides speech output in addition to object detection, which will assist blind persons in identifying objects. Our model also displays the object's distance from the camera and warns blind people if the object is too close. We've also worked on a navigation system for blind individuals that will direct them in a specific direction based on the obstacles in their path. So, for visually challenged persons, we've basically created a whole proactive framework for object identification and path navigation. In the future, we intend to incorporate more objects that are used in the day-to-day lives of blind people. To do so, we will visit a blind school and gain a better knowledge of the problem on the ground level.

7.2 Results-

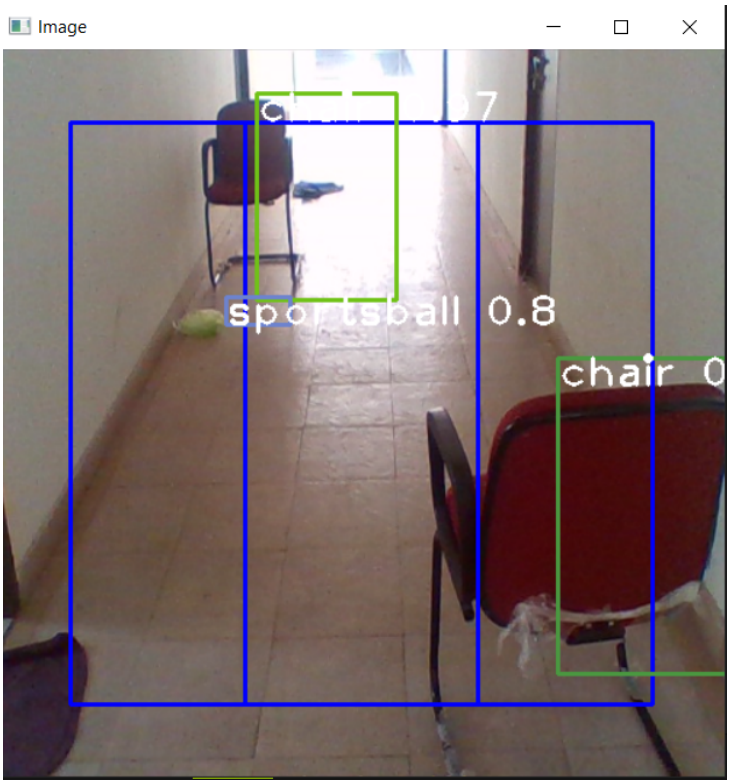


Fig 10.

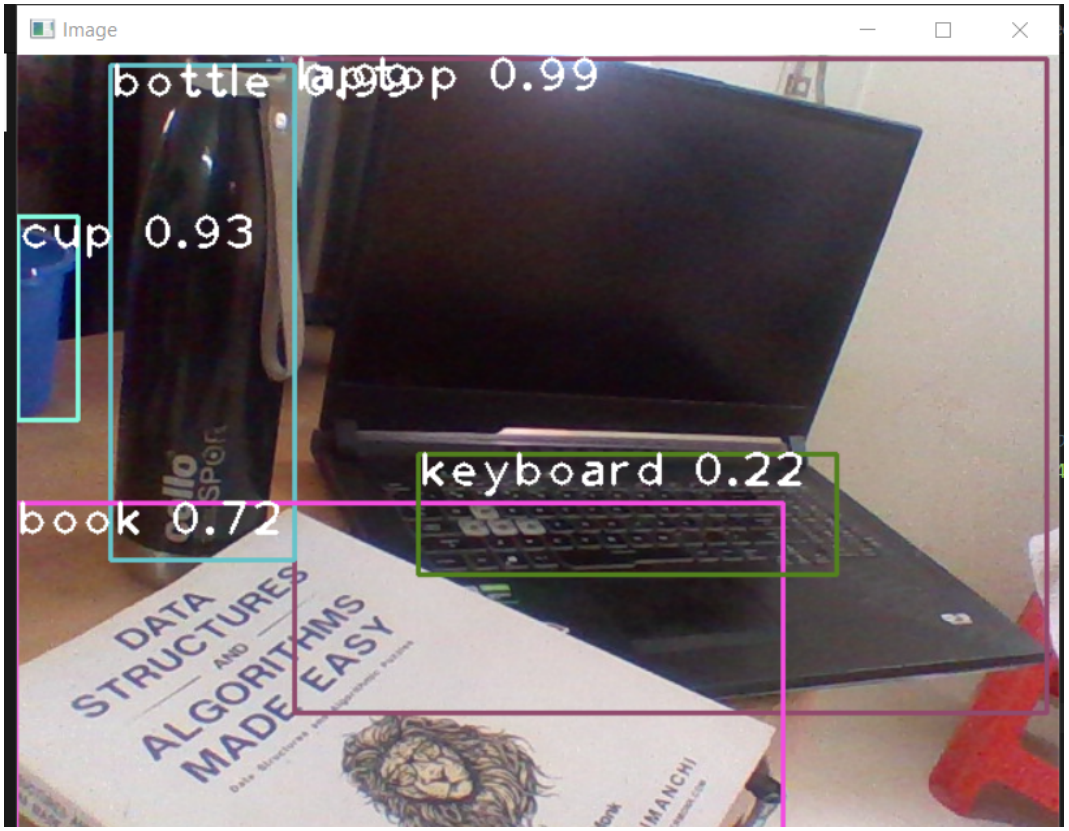


Image
bottle 0.99
laptop 0.99
cup 0.93
keyboard 0.22
book 0.72
DATA
STRUCTURES
AND
ALGORITHMS
MADE EASY

Fig 11.

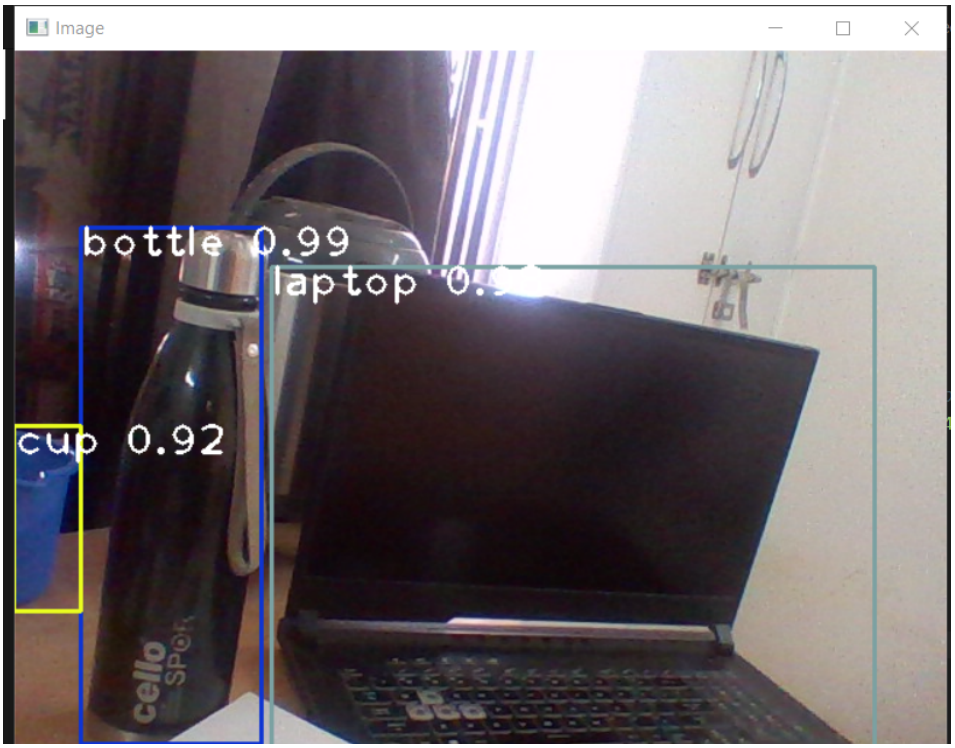


Image
bottle 0.99
laptop
cup 0.92

Fig 12.

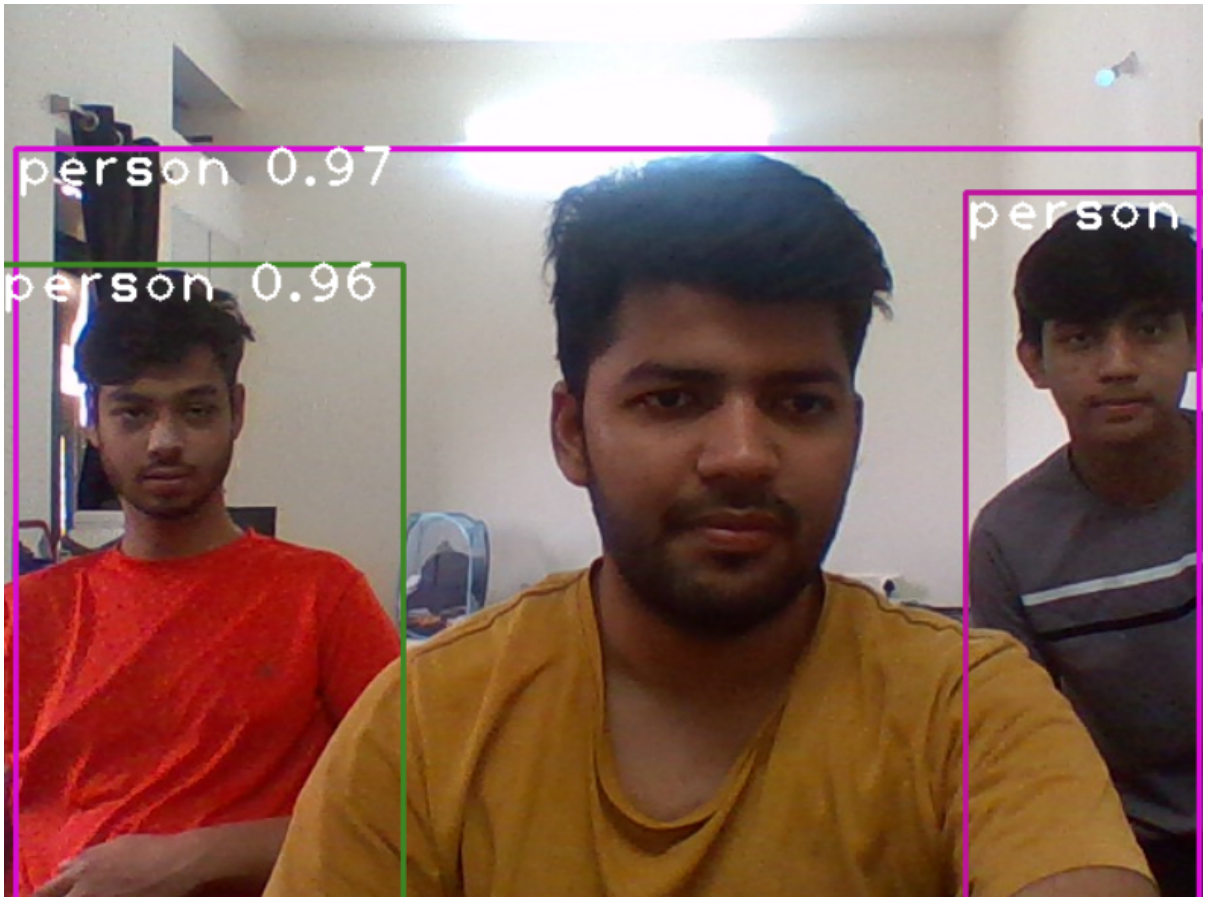


Fig 13.

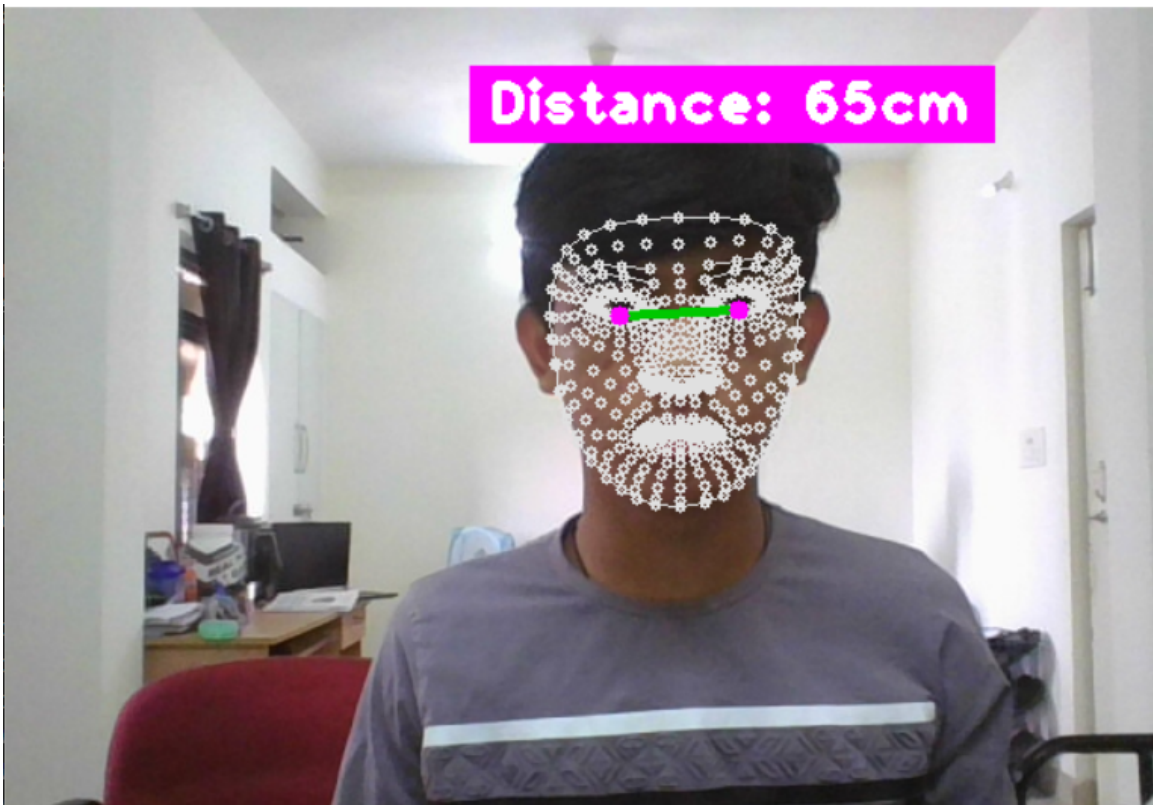


Fig 14.

Fig 15.

REFERENCES

1) Supporting Blind People in Recognizing Personal Objects by Dragan Ahmetovic, Tatsuya Ishihara, Daisuke Sato, Kris Kitani, Uran Oh, Chieko Asakawa

2) Object Detection Based on YOLO Network by Chengji Liu ,Yufan Tao ,Jiawei Liang ,Kai Li ,Yihang Chen

3)An Object Detection System Based on YOLO in Traffic Scene by Jing Tao, Hongbo Wang, Xinyu Zhang, Xiaoyu Li, Huawei Yang

4)YOLO-LITE: A Real-Time Object Detection Algorithm Optimized for Non-GPU Computers by Rachel Huang* Jonathan Pedoeem* Cuixian Chen

5) A Gentle Introduction to Object Recognition With Deep Learning by Jason Brownlee on May 22, 2019 in Deep Learning for Computer Vision

6)https://www.section.io/engineering-education/introduction-to-yolo-algorithm-for-object-detection/

7)https://www.youtube.com/watch?v=ag3DLKsl2vk