Object Detection and Automated Voice Assistant

Authors Name: Deepraj Singh
Department: Computer Science Engineering
SRM Institute of Science and Technology,
KTR, Chennai, 603203, India.
Email Id: ds5925@srmist.edu.in

Authors Name: Prashant Pathak
Department: Computer Science Engineering
SRM Institute of Science and Technology,
KTR, Chennai, 603203, India.
Email Id: pp9069@srmist.edu.in

Abstract — In this project we propose a method to assist visually impaired person to navigate using mobile device and avoid collision with objects. It also provide a voice assistance about the object coming in front of the person.

It is trustworthy and appropriate for use as either a system that operates alone or as a system that can easily be included into a much larger system.

INTRODUCTION

With the advancement in technology, it is possible to have an application that could help a person with visual impairment to navigate with voice assistance.
In order to have this implemented we need identify the object in front of the person and then using voice assitance will be provided about the collision with objects.

Object Detection is a key technology in many applications such as autonomous vehicles, security systems, and robotics.

Voice alerts in object detection can be used to help differently-abled people. They are commonly used in security systems and autonomous vehicles.

This report will explore the state of the art in object detection and voice alerts. It will discuss the challenges and opportunities in this field.

Robotics, security systems, and autonomous cars are just a few of the numerous applications where object detection is a crucial technology.

Disabled people can benefit from voice notifications in object detection. They are frequently employed in driverless vehicles and security systems.

The most recent developments in object detection and voice alerts will be examined in this report. It will go over the difficulties and possibilities in this area.

The technique of extracting human, mobile, remote, and other real-world object instances from photos or movies is known as object detection. It provides understanding

Applications including image retrieval, national security, CCTV monitoring, and advanced driver assistance systems typically employ it (ADAS). It can be done in a variety of ways:

- Feature Based Object Detection
- Viola Jones Object Detection
- SVM Classifications with HOG Features
- Deep Learning Object Detection

Video surveillance from video is one of most important application for object detection.
There are many applications for object detection. Some examples include:

-Autonomous vehicles
-Facial recognition
-Security and surveillance
-Package delivery
-Robotics

A system that can detect objects in difficult scenarios, such as low-light or cluttered scenes.

A model for image classification or image recognition only determines the likelihood that an object is present in the image. In contrast, object localization refers to pinpointing a specific object's location within an image.
The coordinates of an object's location in relation to the image are output by an object localization algorithm. Bounding boxes are the most common tool used in computer vision to indicate the location of an object within an image. Figure depicts a bounding box in action.

In order to make machine learning more approachable for students, creators, and non-technical people, ML5JS was built on top of TensorFlow. There are no other external dependencies.

A model for image classification or image recognition only determines the likelihood that an object is present in the

image. In contrast, object localization refers to pinpointing a specific object's location within an image.
The coordinates of an object's location in relation to the image are output by an object localization algorithm. Bounding boxes are the most common tool used in computer vision to indicate the location of an object within an image. Figure depicts a bounding box in action.

A method of object detection that seeks to locate and recognise numerous things in a single image.

The COCO-SSD model has been ported to TensorFlow.js in this model. to learn more about the object detection API for Tensorflow. Single Shot MultiBox Detection is known as SSD.

The COCO dataset, which is a sizable object identification, segmentation, and captioning dataset, contains items that are specified using this model. The model can identify 80 different classes of things. Several of the items include:

- Person
- Bicycle
- Car
- Bottle
- Plane

You Only Look Once is a method widely used by scientists all over the globe since it is one of the most efficient methods for the detection of items (YOLO). Experts from Facebook's AI Research team said YOLO's unified architecture is very fast. The Fast YOLO network, although being smaller than the original YOLO model, achieves twice the maximum possible performance (mAP) of conventional real-time detectors by analysing an incredible 155 frames per second. The YOLO base model analyses images at a frame rate of 45 per second in real time. This technique outperforms previous detection methods like DPM and R-CNN when applied to new areas like art. The realm of art is an illustration of one of these fields.

STATE OF THE ART

Referencing a research by Geethapriya S. and N. Duraimurugan, they propose the YOLO technique for object detection using a single neural network. Since it is a general-purpose method, it outperforms alternative approaches after extrapolating from natural images to new areas. The technique is straightforward to implement and can be learned using only the entire input picture. Strategy implementations that propose a region constrain the classifier to that area alone. All-or-nothing approach to boundary prediction using the complete picture. As an added bonus, it makes more accurate predictions in low-priority settings. In terms of efficiency and speed, this classification method significantly outperforms its competitors.

Four representative datasets, including ECSSD [156], HKU-IS [146], PASCALS [157], and SOD [158], are used to evaluate several state-of-the-art methods. ECSSD consists of 1000 structurally complex but semantically meaningful natural images. HKU-IS is a large-scale dataset containing over 4000 challenging images. Most of these images have more than one salient object and own low contrast. PASCALS is a subset chosen from the validation set of PASCAL VOC 2010 segmentation dataset and is composed of 850 natural images. The SOD dataset possesses 300 images containing multiple salient objects. The training and validation sets for different datasets are kept the same as those in [152].

Two standard metrics, namely F-measure and the mean absolute error (MAE), are utilized to evaluate the quality of a saliency map. Given precision and recall values pre-computed on the union of generated binary mask B and ground truth Z, F-measure is defined as below

$$F_\beta = \frac{(1+\beta^2)Presion \times Recall}{\beta^2 Presion + Recall} \tag{7}$$

where β^2 is set to 0.3 in order to stress the importance of the precision value.

The MAE score is computed with the following equation

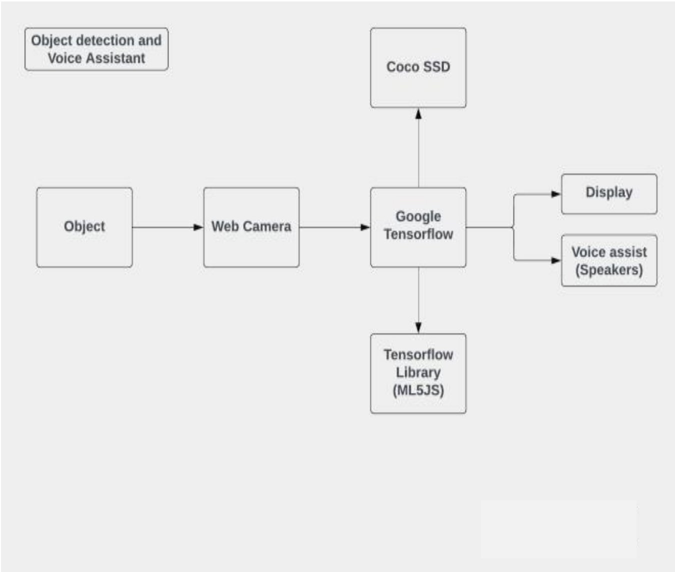
$$MAE = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \left| \hat{S}(i,j) = \hat{Z}(i,j) \right| \tag{8}$$

PROPOSED WORK

We are are planning to make a clean-fresh UI, which is also responsive. Also We are using system default voice and accessing it using a script written in JavaScript. Moreover are Using ML5JS which is a TensorFlow Library.We are using COCO SSD as the datasets.
A deep learning-based object detection system that can identify multiple objects in an image and provide accurate bounding boxes around them.
An object detection system that uses a combination of deep learning and traditional computer vision techniques to achieve high accuracy.
A real-time object detection system that can run on a mobile device or embedded system.
An object detection system that can be trained on large datasets and then deployed on a smaller device for real-time detection.

A system that can detect objects in difficult scenarios, such as low-light or cluttered scenes.

RESULTS DISCUSSSION



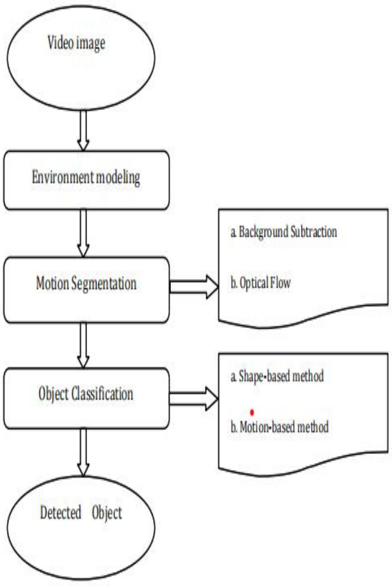


Figure 2: Schematic Representation of Object Detection

It provides specific information on the results of the system's testing as well as the many test cases that were utilized. The COCO pretrained dataset, which included a total of 80 classes, was the one that we utilised. Because adding additional classes would have led to an overfitting of the data, we decided to stick with eighty classes instead. Recent developments in deep learning have made it possible for generic object identification networks to become more accurate and efficient. For the training of highly trustworthy models, large datasets including pictures that are rich in detail and highly textured are required. In contrast, the general object detection system has a poorer performance in real-world scenarios when the items in question are:

(a) hidden by occlusions

(b) visible in photographs with low light

(c) combined with data from the backdrop.

Within the context of this article, each of these settings is referred to as a challenging environment. As a direct result of the recent explosion in the production of generic object identification algorithms, there has been a significant amount of progress made in the industry of detecting objects in challenging environments. These developments have been rather noticeable. However, there is not a single reference that covers everything there is to know about the state of the art in this sector. This paper presents, to the best of our knowledge, the first comprehensive review of modern strategies that have been employed to solve the issue of object detection in challenging scenarios. In addition, we tackle the challenging datasets that are now accessible and provide both quantitative and qualitative performance evaluations of a variety of methods. This article also analyses the effectiveness of the most cutting-edge generic object identification algorithms by evaluating their performance on three well-known and notoriously challenging datasets. In conclusion, we provide a variety of existing problems as well as prospective future directions..

.

I. CONCLUSION & FUTURE WORK

This project is will help in wide population. Also it will help the diffrently abled people for path detection. Also It can be used in CCTV surveillance.

The hybrid tracker performs better in terms of system resource usage, as extrapolated from the results of these benchmark tests. This is a highly desirable characteristic that could enable the system to function in a confined space (such as an em-bedded sensor environment). The tracker is completely modular, allowing any component to operate independently and provide its own processed output from its own input streams. The tracker's precision is its primary flaw, which, depending on the application, can be devastatingly catastrophic. As has been stated several times before, this tracker uses a segmentation technique that allows it to respond to changes in the foreground of the image.

This makes suggestions for potential upgrades and extensions to the tracking system that would enable it to function as a component of a larger system. It serves as a concluding comment to the report by briefly outlining any changes that might be made and expressing the author's own opinion of the project.

Autonomous Control of a Robot

Building a robot helicopter with a system control that could be used as an autonomous control system or by a user was one of the objectives put forth at the project's outset. A tracker, able to transmit the robot's current location to the control centre at any time, would be one of the modules needed for such a mission. Recognizing the objections made in section 5.3, it is possible to further enhance this tracker by supplying a more advanced algorithm and putting the system in place on a faster processor. Then this tracker might be employed in a situation like that.

Optimizations and Distribution

The code might benefit from a number of optimizations. For instance, the render time of the point cloud may be delayed due to the visualizer's threaded design. This is most likely a result of the application code's effectiveness. This might have been improved with extra time to allow for a more fluid rendering process. A command panel that allowed the user to access and close specific components of the system at whim would have been another addition that would have improved the user interface. However, as the author didn't think it was vital to show what the project performs, it wasn't used for this project.

For this author, building a distributed environment for the code to run in would be the most crucial optimization. It is evident that the next step would be to use this to speed up the system as ROS naturally supports the use of multiple machine control of the operating system. The tracker is currently modular, so parts may be easily run on different systems, increasing the amount of processing that can be done each frame.

This processing could include creating more pertinent data than the object's location, rendering a faster point cloud, or adding an additional filtering mechanism to the optical tracker (velocity etc.). Naturally, the system might be changed to include a model of the object being identified based on the lessons learnt from the roboearth detector. An additional machine could process the system itself after a single machine loads the model, offering even better performance. As a result, the tracker would be able to filter out undesired objects and offer a quick tracking paradigm with pose estimation.

Closing Comments

This project used contemporary sensor technologies to address the tracking issue. A system that actually performs as mentioned in this report was constructed with the aid of rich APIs. This report provided a full account of how the project was developed from a formal specification to design and implementation. The system's advantages and disadvantages were examined, and a comparison with other systems of a similar nature was done. The user should now have the best grasp of the tracking system that has been established thanks to this report, we hope. The project implementation and a created documentation directory are included with the report. This directory includes information on the code implementation as well as instructions for installing the system. The author hopes that the reader will try to implement the system where it is feasible so they may directly observe how it operates.

REFERENCES

1. Geethapriya S, N. Duraimurugan, S.P. Chokkalingam, “Real-Time Object Detection with Yolo”, International Journal of Engineering and Advanced Technology (IJEAT).

2. Abdul Vahab, Maruti S Naik, Prasanna G Raikar an Prasad S R4, “Applications of Object Detection System”, International Research Journal of Engineering and Technology (IRJET).

3. Hammad Naeem, Jawad Ahmad and Muhammad Tayyab, “Real-Time Object Detection and Tracking”,IEEE.

4. Meera M K, & Shajee Mohan B S. 2016, "Object recognition in images", International Conference on Information60
Science (ICIS).

5. Astha Gautam, Anjana Kumari, Pankaj Singh: "The Concept of Object Recognition", International
Journal of Advanced Research in Computer Science and Software Engineering, Volume 5, Issue 3, March 2015.

6. Joseph Redmon, Santosh Divvala, Ross Girshick, “You Only Look Once: Unified, Real-Time Object
Detection”, The IEEE Conference on Computer Vision and Pattern Recognition (CVPR),2016,pp. 779-
788