BLIND - SIGHT: OBJECT DETECTION WITH VOICE FEEDBACK

A PROJECT REPORT

Submitted by

A. ANNAPOORANI	**160801011**
NEROSHA SENTHIL KUMAR	**160801059**

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

SRI VENKATESWARA COLLEGE OF ENGINEERING

(An Autonomous Institution)

SRIPERUMBUDUR

NOVEMBER 2019

SRI VENKATESWARA COLLEGE OF ENGINEERING
(An Autonomous Institution)

BONAFIDE CERTIFICATE

Certified that this project report "**BLIND - SIGHT: OBJECT DETECTION WITH VOICE FEEDBACK"** is the bonafide work of "**A. ANNAPOORANI (160801011)** and **NEROSHA SENTHIL KUMAR (160801059)"** who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr. V. VIDHYA, M.E., Ph.D.,	**Dr. V. VIDHYA, M.E., Ph.D.,**
HEAD OF THE DEPARTMENT	**SUPERVISOR**
	Head of the Department
Department of Information Technology	Department of Information Technology
Sri Venkateswara College of Engineering	Sri Venkateswara College of Engineering
Sriperumbudur Tk. - 602 117	Sriperumbudur Tk. - 602 117

Submitted for the Project Viva-Voce held on _______________ at

Sri Venkateswara College of Engineering, Sriperumbudur.

INTERNAL EXAMINER	**EXTERNAL EXAMINER**

ABSTRACT

Computer vision deals with how computers can be made to gain high-level understanding from digital images or videos. It seeks to automate tasks that the human visual system can do. Humans glance at an image and instantly know what objects are in the image, where they are, and how they interact. An estimate of 285 million people are visually impaired worldwide, stated by WHO. The proposed Blind Sight-Object Detection with Voice Feedback is a computer vision-based application that leverages state of the art object detection techniques. These are employed to detect objects in the vicinity. You Only Look Once (YOLO): Unified, Real-Time Object Detection a new approach to object detection is deployed in this proposed work. Yolo has 75 Convolutional Neural Network (CNN). Image classification techniques are used to identify the features of the image and categorize them into their appropriate class. The COCO dataset used in this project consists of around 123,287 hand labelled images classified into 80 categories. This wide set of data is used to describe spatial relationships between objects and their location in the environment. The text description of the recognised object will be sent to the Google Text-to-Speech API using the gTTS package. Voice feedback on the 1st frame of each second will be scheduled as an output to help the visually impaired hear what they cannot see.

ACKNOWLEDGEMENT

We thank our principal **Dr. Ganesh Vaidyanathan, Ph.D.**, Principal, Sri Venkateswara College of Engineering for being the source of inspiration throughout our study in this college.

We express our sincere thanks to **Dr. V. Vidhya, M.E., Ph.D.,** Head of Department, Information Technology and our guide for her continuous and valuable guidance throughout this project.

We are also thankful to **Ms. N. Devi, M.E., Mr. V. Rajaram, M.E., Mr. V. Ranjith, M.E.,** project coordinators for their continual support and assistance.

We are also thankful to all faculty members and supporting staff of the Department of Information Technology.

A. ANNAPOORANI

NEROSHA SENTHIL KUMAR

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

YOLO	YOU ONLY LOOK ONCE
CNN	CONVOLUTIONAL NEURAL NETWORK
RNN	RECURRENT NEURAL NETWORK
PCA	PRINCIPLE COMPONENT ANALYSIS
FPP	FALSE POSITIVE PRUNING
TTS	TEXT- TO -SPEECH
NMT	NEURAL MACHINE TRANSLATION
CTC	CONNECTIONIST TEMPORAL CLASSIFICATION
COCO	COMMON OBJECTS IN CONTEXT
CCD	CHARGED COUPLED DEVICE
FAST	FEATURES FROM ACCELERATED SEGMENT TEST

CHAPTER 1

INTRODUCTION

Computer vision is a field of computer science that works on enabling computers to see, identify and process images in the same way that human vision does, and then provide appropriate output. This state-of-the-art technology is deployed in the application Blind Sight-Object Detection with Voice Feedback. It is closely linked with artificial intelligence, as the computer interprets what it sees, and then performs appropriate analysis to help the visually challenged to hear what they cannot see. This chapter gives a detailed introduction about the domain and also discusses the various existing solutions available along with their disadvantages. It also emphasises and highlights the need for the proposed system.

1.1 OVERVIEW

Computer vision is an interdisciplinary scientific field that deals with how computers can be made to gain high-level understanding from digital images or videos. From the perspective of engineering, it seeks to automate tasks that the human visual system can do. Computer vision tasks include methods for acquiring, processing, analysing and understanding digital images, and extraction of high-dimensional data from the real world in order to produce numerical or symbolic information, e.g. in the form of decisions. Understanding in this context means the transformation of visual images (the input of the retina) into descriptions of the world that can interface with other thought processes and elicit appropriate action. This image understanding can be seen as the disentangling of symbolic information from image data using models constructed with the aid of geometry, physics, statistics, and learning theory.

1.2 OBJECTIVE OF THE PROJECT

The whole objective of the project is divided into two phases as phase I and phase II.

1.2.1 OBJECTIVE PHASE - I

In the first phase of the project, object detection and recognition were done. The model here is the You Only Look Once (YOLO) algorithm that runs through a variation of an extremely complex Convolutional Neural Network architecture called the Darknet. Previously, classification-based models were used to detect objects using localization, region-based classification or things such as the sliding window. Only high scoring regions of the images were considered as a detection and they could be very time-consuming. Instead, YOLO is regression-based. Predicting classes and bounding boxes for the whole image quickly in one run of the algorithm (just one look of the image's pixels) is so challenging that the predictions are informed by the global context in the image and the model is trained with the ImageNet dataset.

1.2.2 OBJECTIVE PHASE -II

The class prediction of the objects detected in every frame will be a string e.g. “cat”. The coordinates of the objects in the image is appended to the position “top” / “mid” / “bottom” & “left” / “centre” / “right” to the class prediction “cat”. The text description is sent to the Google Text-to-Speech API using the gTTS library. gTTS is a Python library and CLI tool to interface with Google Translate’s text-to-speech API.

1.3 APPLICATIONS OF OBJECT DETECTION

Object detection has various real time applications in various domains. Few of the applications are explained below.

1.3.1 OPTICAL CHARACTER RECOGNITION

Optical character recognition or optical character reader, often abbreviated as OCR, is the mechanical or electronic conversion of images of typed, handwritten or printed text into machine-encoded text, whether from a scanned document, a photo of a document, a scene-photo (for example the text on signs and billboards in a landscape photo) or from subtitle text superimposed on an image, characters are extracted from the image or video.

1.3.2 SELF DRIVING CARS

One of the best examples for the need of object detection is for autonomous driving is in order for a car to decide what to do in next step whether accelerate, apply brakes or turn, it needs to know where all the objects are around the car and what those objects are that requires object detection and essentially train the car to detect known set of objects such as cars, pedestrians, traffic lights, road signs, bicycles, motorcycles, etc.

1.3.3 TRACKING OBJECTS

Object detection system is also used in tracking the objects, for example tracking a ball during a football match, tracking movement of a cricket bat, tracking a person in a video. Object tracking has a variety of uses, some of which are surveillance and security, traffic monitoring, video communication, robot vision and animation.

1.3.4 OBJECT EXTRACTION FROM AN IMAGE OR VIDEO

Object Extraction is a closely related issue with the segmentation process. Image Segmentation is a process of dividing an image into sub partition based on some characteristics like colour, intensity etc. The main goal of object extraction is to change the representation of an image into something more meaningful. To extract an object from the image first the entire image is segmented. User select the region as background and foreground by using the markers and then the algorithm will segment the image and the foreground region will be extracted from the image. In future objects can also be extracted from video with further improvement of this technology.

1.4 EXISTING SYSTEM

Earlier work in object detection has had its application in areas like optical character recognition (OCR), self-driving cars, tracking objects, face detection, face recognition, image retrieval, security and surveillance. It has also been extended to help visually challenged people to recognise the objects. However, these detections have been carried out by developing extensive neural networks like CNN and numerous hidden layers. This decreases the speed of image retrieval and processing to a considerable amount.

1.5 ORGANIZATION OF THE REPORT

The report is organized as follows: Chapter 2 presents a Literature Survey. Chapter 3 discusses the architecture of the proposed work and the project schedule. Chapter 4 - 7 presents the modules of the proposed work. Chapter 8 presents the implementation and experimental results. Chapter 9 specifies Conclusion and Future work to be done.

CHAPTER 2

LITERATURE SURVEY

2.1 YOU ONLY LOOK ONCE: UNIFIED, REAL-TIME OBJECT DETECTION

Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi (2015) proposed a “You Only Look Once: Unified, Real-Time Object Detection” to detect real - time objects. Frame object detection as a regression problem to spatially separated bounding boxes and associated class probabilities was proposed. A single neural network predicts bounding boxes and class probabilities directly from full images in one evaluation. Since the whole detection pipeline is a single network, it is optimized end-to-end directly on detection performance. The unified architecture is extremely fast. YOLO model processes images in real-time at 45 frames per second. A smaller version of the network, Fast YOLO, processes an astounding 155 frames per second while still achieving double the mAP of other real-time detectors. Compared to state-of-the-art detection systems, YOLO makes more localization errors but is far less likely to predict false detections where nothing exists.

2.2 CONVOLUTIONAL NEURAL NETWORK BASED LIVE OBJECT RECOGNITION SYSTEM AS BLIND AID

Kedar Potdar, Chinmay D. Pai, Sukrut Akolkar(2018) proposed "A Convolutional Neural Network based Live Object Recognition System as Blind Aid" to perform live object recognition. Live object recognition system is used that serves as a blind aid. The act of knowing what object is in front of the blind person without touching it (by hands or using some other tool) is very difficult. In some cases, the physical contact between the person and object is dangerous, and even lethal. A Convolutional Neural Network is employed for recognition of pre-trained objects on the ImageNet dataset. A camera, aligned with the systems pre-determined orientation, serves as input to a computer system, which has the object recognition Neural Network deployed to carry out real-time object detection. Output from the network can then be parsed to present to the visually impaired person either in form of audio or Braille text. Efficient weight sharing is one of the major merits of that proposed system. In addition, it uses good feature extractors to extract features. However, it cannot perform future learning which is a demerit.

2.3 ANDROID BASED OBJECT RECOGNITION INTO VOICE INPUT TO AID VISUALLY IMPAIRED

J.Prakash, P.Harish, Ms.K.Deepika (2015) proposed android based object recognition into voice input to aid visually impaired. The main features of software modules dedicated to the aid of visually impaired or blind users are mentioned. Reduce or elimination of the need of separate dedicated devices for object recognition and motion detection is the achieved. The software modules are designed for Android operating system, used in majority of the smart phones today. Principal component analysis (PCA) algorithm to recognize the object is deployed. To support real-time scanning of objects, a key frame extraction algorithm is framed that automatically retrieves high-quality frames from continuous camera video stream of mobile phones. The sequence is approximately capture 3 frames per second. The object is recognized then converted into text, BY text to speech application it is converted into the voice output. Easy Integration of voice command on android events like object detection system is done efficiently. However, it is hard to model long dependency using current recurrent neural networks (RNNs).

2.4 APPLICATION OF DEEP LEARNING IN OBJECT DETECTION

Xinyi Zhou , Wei Gong , Wen Long Fu , Fengtong Du (2017) elaborated on the application of deep learning in object detection. With the rapid development of deep learning, a number of research areas have achieved good results, and accompanied by the continuous improvement of convolution neural networks, computer vision has arrived at a new peak. The most popular choice of Smartphone’s among visually impaired users is Android based phones. Commonly, the non-operating system devices are not preferred by blind users as they do not offer special functions such as text to speech conversion. A number of dedicated devices for navigation and object recognition are in use. These wearable devices have the disadvantage that they are expensive in comparison to software. Also, the blind users are required to carry a number of gadgets and devices, each for a different purpose such as object identifiers. One of the major reasons to deploy deep learning in object detection is the accuracy of deep learning when trained with huge amount of data. However, it is hard to model long dependency using current recurrent neural networks.

2.5 OBJECT DETECTION COMBINING RECOGNITION AND SEGMENTATION

Liming Wang1, Jianbo Shi, Gang Song, and I-fan Shen1 (2017) proposed "Object Detection Combining Recognition and Segmentation" to detect, recognize and perform segmentation. An object detection method was developed combining top-down recognition with bottom-up image segmentation. The two main steps in that method were: a hypothesis generation step and a verification step. In the top-down hypothesis generation step, an improved Shape Context feature was designed, which is more robust to object deformation and background clutter. The improved Shape Context was used to generate a set of hypotheses of object locations and figure ground masks, which have high recall and low precision rate. In the verification step, a set of feasible segmentations that are consistent with top-down object hypotheses were constructed and then a False Positive Pruning (FPP) procedure was proposed to prune out false positives. The fact that false positive regions typically do not align with any feasible image segmentation was exploited. Object detection is an important, yet challenging vision task. It is a critical part in many applications such as image search, image auto-annotation and scene understanding; however, it is still an open problem due to the complexity of object classes and images.

2.6 TEXT-TO-SPEECH CONVERSION WITH NEURAL NETWORKS: A RECURRENT TDNN APPROACH

O. Karaali, G. Corrigan, I. Gerson, and N. Massey (1998) proposed "TEXT-TO-SPEECH CONVERSION WITH NEURAL NETWORKS: A RECURRENT TDNN APPROACH" which converts text to speech using recurrent TDNN approach. The design of a neural network that performs the phonetic-to-acoustic mapping in a speech synthesis system was described. The use of a time-domain neural network architecture limits discontinuities that occur at phone boundaries. Recurrent data input also helps smooth the output parameter tracks. Independent testing has demonstrated that the voice quality produced by this system compares favourably with speech from existing commercial text-to-speech systems. While neural networks have been employed to handle several different text-to-speech tasks, the system is the first system to use neural networks throughout, for both linguistic and acoustic processing. The text-to-speech task was divided into three subtasks, a linguistic module mapping from text to a linguistic representation, an acoustic module mapping from the linguistic representation to speech, and a video module mapping from the linguistic representation to animated images. The linguistic module employs a letter-to-sound neural network and a post lexical neural network. The acoustic module employs a duration neural network and a phonetic neural network. The visual neural network is employed in parallel to the acoustic module to drive a talking head.

2.7 DEEP VOICE: REAL-TIME NEURAL TEXT-TO-SPEECH

Sercan O. Arik, Mike Chrzanowski, Adam Coates, Gregory Diamos, Andrew Gibiansky, Yongguo Kang, Xian Li, John Miller, Andrew Ng, Jonathan Raiman, Shubho Sengupta, Mohammad Shoeybi have put forth a paper for real time neural text to speech. Deep Voice, a production-quality text-to-speech system constructed entirely from deep neural networks. Deep Voice lays the groundwork for truly end-to-end neural speech synthesis. The system comprises five major building blocks: a segmentation model for locating phoneme boundaries, a grapheme-to-phoneme conversion model, a phoneme duration prediction model, a fundamental frequency prediction model, and an audio synthesis model. For the segmentation model, a novel way of performing phoneme boundary detection with deep neural networks was developed using connectionist temporal classification (CTC) loss. For the audio synthesis model, a variant of WaveNet that requires fewer parameters and trains faster than the original was implemented. By using a neural network for each component, the system is simpler and more flexible than traditional text-to-speech systems, where each component requires laborious feature engineering and extensive domain expertise.

Finally, the inference with the system performed faster than real time and describe optimized WaveNet inference kernels on both CPU and GPU that achieve up to 400x speedups over existing implementations.

CHAPTER 3

PROPOSED ARCHITECTURE AND PLANNING OF BLIND - SIGHT

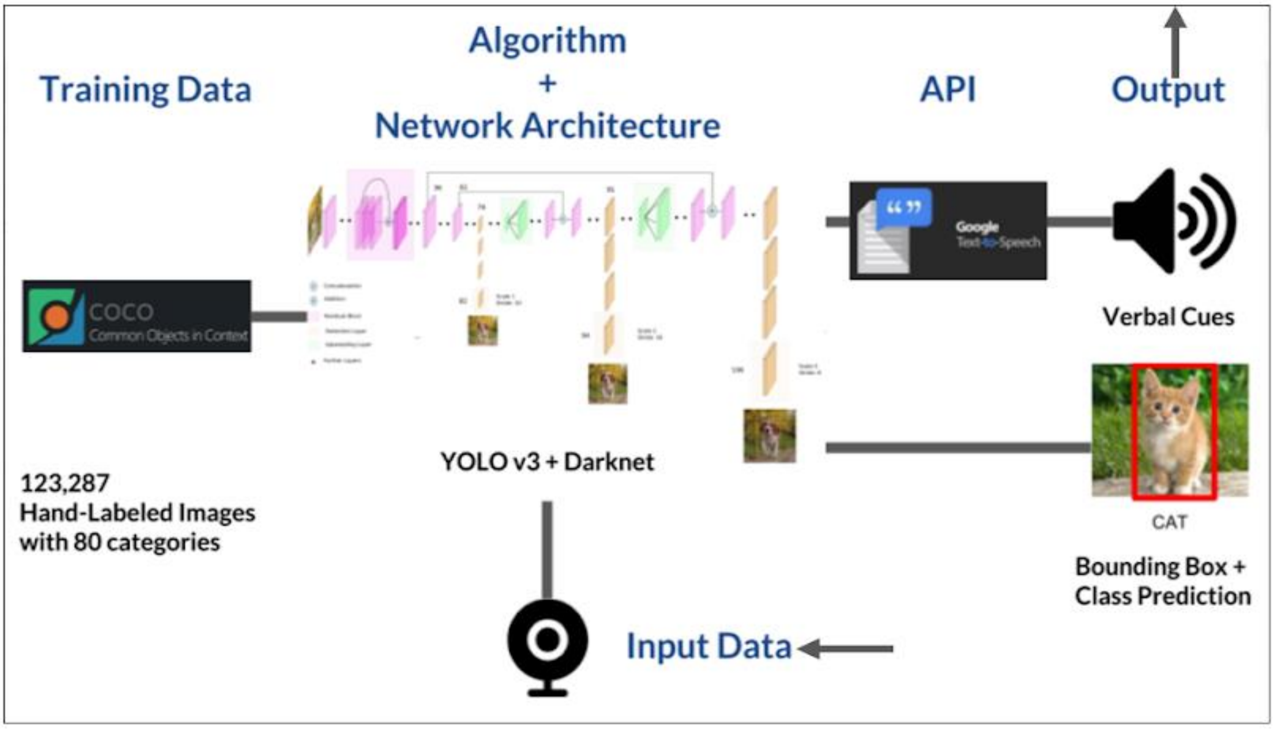


Figure 3.1 Proposed Architecture of Blind - Sight

Figure 3.1 shows the proposed architecture of Blind – Sight. Object Detection is a field of Computer Vision that detects instances of semantic objects in images/videos (by creating bounding boxes around them in this case). The annotated text is then converted into voice responses and gives the basic positions of the objects in the person/camera’s view.

3.1 DATA SET

Dataset is one of the foundations of deep learning, for many researchers to get enough data to carry out the experiment just by themselves is a big problem, so a lot of open source dataset is needed for everyone to use. Some commonly used datasets in computer vision are the following.

3.1.1 IMAGENET

The ImageNet dataset has more than 14 million images covering more than 20,000 categories. There are more than a million pictures with explicit class annotations and annotations of object locations in the image. The ImageNet dataset is one of the most widely used datasets in the field of deep learning. It is very widely used in the field of computer vision research, and has become the “standard” dataset of the current deep learning of image domain to test algorithm performance.

3.1.2 PASCAL VOC

The PASCAL VOC (pattern analysis, statistical modelling and computational learning visual object classes) provides standardized image data sets for object class recognition and provides a common set of tools for accessing the data sets and annotations. The PASCAL VOC dataset includes 20 classes and has a challenge based on this dataset. The PASCAL VOC Challenge is no longer available after 2012, but its dataset is of good quality and well-marked, and enables evaluation and comparison of different methods. And because the amount of data of the PASCAL VOC dataset is small, compared to the ImageNet dataset, very suitable for researchers to test network programs. The dataset is also created based on the PASCAL VOC dataset standard.

3.1.3 COCO

COCO (Common Objects in Context) is a new image recognition, segmentation, and captioning dataset, sponsored by Microsoft. COCO dataset has more than 300,000 images covering 80 object categories. The open source of this dataset makes great progress in semantic segmentation in recent years, and it has become a “standard” dataset for the performance of image semantic understanding, and also COCO has its own challenge.

3.2 TRAINING DATA

The model is trained with the Common Objects in Context (COCO) dataset. The training set is the material through which the computer learns how to process information. Machine learning uses algorithms – it mimics the abilities of the human brain to take in diverse inputs and weigh them, in order to produce activations in the brain, in the individual neurons. Artificial neurons replicate a lot of this process with software – machine learning and neural network programs that provide highly detailed models of how human thought processes work. For sequential decision trees and those types of algorithms, it would be a set of raw text or alphanumerical data that gets classified or otherwise manipulated. On the other hand, for convolutional neural networks that have to do with image processing and computer vision, the training set is often composed of large numbers of images. The idea is that because the machine learning program is so complex and so sophisticated, it uses iterative training on each of those images to eventually be able to recognize features, shapes and even subjects such as people or animals. The training data is absolutely essential to the process – it can be thought of as the "food" the system uses to operate. COCO is a large-scale object detection, segmentation, and captioning dataset. COCO has several features:

- Object segmentation
- Recognition in context
- Super pixel stuff segmentation
- 330K images (>200K labelled)
- 1.5 million object instances
- 80 object categories
- 91 stuff categories
- 5 captions per image
- 250,000 people with key points

3.3 MODEL

The model here is the **You Only Look Once** (YOLO) algorithm that runs through a variation of an extremely complex Convolutional Neural Network architecture called the Darknet. A more enhanced and complex YOLO v3 model is being used here. Also, the python **cv2** package has a method to setup Darknet from configurations in the yolov3.cfg file. A pre-trained model is being used here. This means that COCO has already been trained on YOLO v3 by others and the weights obtained are stored in a 200+mb file. If the weights are not sure, think of it as trying to find the Best Fit Line in Linear Regression. The right values of m and c in y=mx+c is found such that the line minimizes the error between all points. Now in a more complex prediction task, there are millions of Xs when the images are fed into the complex network. These Xs will each have an m and these are the predicted weights stored in yolo v3. weights file. The ms have been constantly readjusted to minimize some loss function.

3.4 INPUT DATA

Webcam to feed images at 30 frames-per-second to this trained model and is set it to only process every other frame to speed things up. A webcam is a compact digital camera that hooks up to the computer to broadcast video images in real. Just like a digital camera, it captures light through a small lens at the front using a tiny grid of microscopic light-detectors built into an image-sensing microchip (either a charge-coupled device (CCD) or, more likely these days, a CMOS image sensor). The image sensor and its circuitry converts the picture in front of the camera into digital format—a string of zeros and ones that a computer knows how to handle. Unlike a digital camera, a webcam has no built-in memory chip or flash memory card: it doesn't need to "remember" pictures because it's designed to capture and transmit them immediately to a computer. That's why webcams have USB cables coming out of the back. The USB cable supplies power to the webcam from the computer and takes the digital information captured by the webcam's image sensor back to the computer—from where it travels on to the Internet. Some cams work wirelessly and don't need to be connected to a computer: typically Wi-Fi is used to transmit their pictures to the Internet router, which can then make them available to other machines on the home network or, using the Internet, to anyone, anywhere in the world.

3.5 API

The class prediction of the objects detected in every frame will be a string e.g. “cat”. The coordinates of the objects in the image are obtained and appended to the position “top”/ “mid”/ “bottom” & “left”/ “centre”/ “right” to the class prediction “cat”. The text description is then sent to the Google Text-to-Speech API using the **gTTS** package. An **application programming interface** (**API**) is an interface or communication protocol between a client and a server intended to simplify the building of client-side software. It has been described as a “contract” between the client and the server, such that if the client makes a request in a specific format, it will always get a response in a specific format or initiate a defined action.

3.6 OUTPUT

The coordinates of the bounding box of every object detected in the frames are obtained, and the boxes are overlayed on the objects detected and the stream of frames are returned as a video playback. A voice feedback is the scheduled on the 1st frame of each second (instead of 30 fps) e.g. “bottom left cat” — meaning a cat was detected on the bottom-left of the camera view.

3.7 PLANNING

The schedule in Table 3.1 shows the time duration to complete the project.

Table 3.1 PROJECT SCHEDULE

ACTION	METHODOLOGY	EXPECTED OUTPUT	TIMELINE
Dataset collection and Literature Survey	ImageNet and COCO Datasets	To find the demerits of the existing methods	20 days
Configuring the existing layers of the neural network	Introducing more hidden layers to increase the depth of the network	To build an efficient neural network with fault tolerance and self-repair	25 days
Training the model	Training the YOLO network using the data collected	Model should detect the object and caption it	30 days
Deploying voice feedback module	Using the gTTS to enable voice output	Voice feedback of the detected object	20 days
Testing	Real time data is tested	Correct objection detection with voice feedback from the captured image	15 days
Report Generation	-	Report	10 days

CHAPTER 4

IMAGE CAPTURING

The first step in the working of the Blind-Sight application is Image Capturing. Live stream is captured with a camera. OpenCV provides a very simple interface to this. A video from the camera is captured, converted it into grayscale video and displayed. To capture a video, Video Capture object is created. Its argument is either the device index or the name of a video file. Device index is the number to specify which camera. Here one camera is connected and 0 (or -1) is passed. The second camera is selected by passing 1. After that, frame-by-frame capture is done. At the end release the capture is released. cap.read() returns a bool (True/False). When frame is read correctly, it is in True state. This is verified at the end of the video by checking this return value. The method cap.isOpened()is initialized. When it is in False state the method is called using cap.open(). Image from the camera is captured. A webcam, a compact digital camera is hooked up to the computer to broadcast video images in real time. Just like a digital camera, it captures light through a small lens at the front using a tiny grid of microscopic light-detectors built into an image-sensing microchip (either a charge-coupled device (CCD) or a CMOS image sensor). Some of the steps include:

a) import cv2

This statement includes the OpenCV library into the program/script, now all the methods and properties available in this library are accessed. In OpenCV, to capture/create an Image or Video, the VideoCapture() method is used which allows to capture the video stream from the webcam. The VideoCapture() method is accessed using the cv2 namespace.dv

b) videoStreamObject = cv2.VideoCapture(0)

In the above code statement that, the VideoCapture() method accepts an integer value as an argument. This integer value represents the camera connected to the device. In almost every laptop, the integrated web camera is the first camera and is accessed by passing the value as 0 (zero). It’s an array with indices pointing to the different available cameras. Thus, the above statement creates a VideoCapture object and returns it, hence the value is stored in a variable. Since a video is a stream of picture frames, using this VideoCapture object the camera is accessed the camera and each frame is retrieved and displayed on the screen.

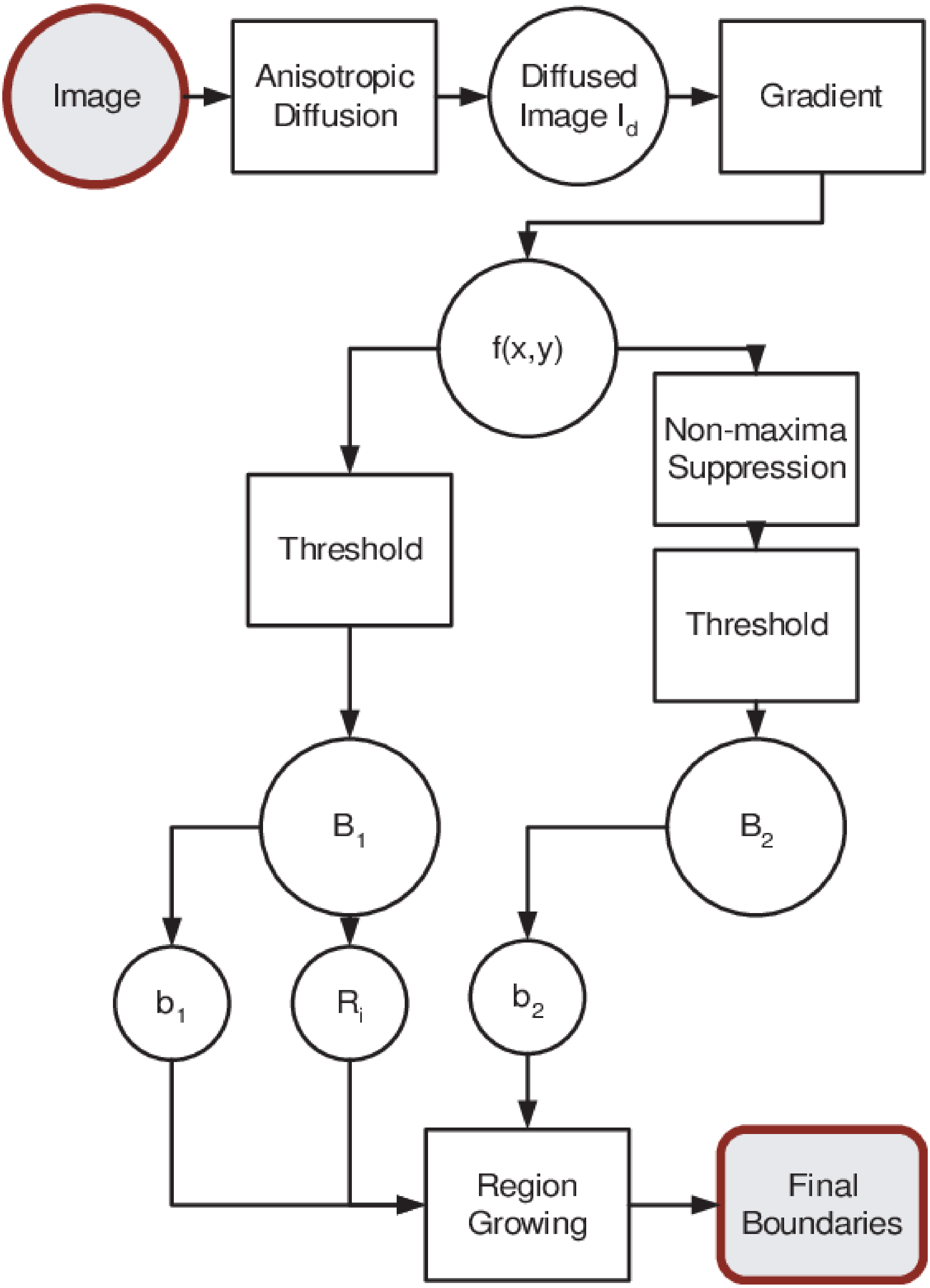


Figure 4.1 Process Flow of Image Capturing

Figure 4.1 shows the process flow of image capturing done in the application. Image processing methods are of two types. The methods used for image processing are:

i. Analog image processing or visual techniques of image processing: used for printouts and photographs.

ii. Digital image processing: processing digital images by using a computer. This technique includes three phases for processing images: pre-processing, enhancement and display, information extraction.

Image pre-processing or image restoration consists of correcting the image from different errors, noise and geometric distortions. Image enhancement improves the visual aspect of the image, after the correction of errors, to facilitate the perception or interpretability of information in the image. Information extraction utilizes the computer's decision-making capability to identify and extract specific pieces of information or pixels. The different image processing techniques used in the Phone Reader Project help in extracting the text contained in the image taken by the user.

CHAPTER 5

OBJECT DETECTION

The second step after image capturing is object detection. Object detection is a computer technology related to computer vision and image processing that deals with detecting instances of semantic objects of a certain class (such as humans, buildings, or cars) in digital images and videos. Previously, classification-based models were used to detect objects using localization, region-based classification or things such as the sliding window. Only high scoring regions of the image are considered as a detection and it could be very time-consuming. Instead, YOLO is deployed in this application which is regression-based. Predicting classes and bounding boxes for the whole image is performed quickly in one run of the algorithm just one look of the image’s pixels, so that the predictions are informed by the global context in the image. The YOLO framework (You Only Look Once) deals with object detection in a different way. It takes the entire image in a single instance and predicts the bounding box coordinates and class probabilities for these boxes. The biggest advantage of using YOLO is its superb speed – it’s incredibly fast and can process 45 frames per second. YOLO also understands generalized object representation. This is one of the best algorithms for object detection and has shown a comparatively similar performance to the R-CNN algorithms. The different techniques used in YOLO algorithm is described in the following steps:

- YOLO first takes an input image.
- The framework then divides the input image into grids (say a 3 X 3 grid)
- Image classification and localization are applied on each grid. YOLO then predicts the bounding boxes and their corresponding class probabilities for objects.

After the object is detected, the following pre-processing steps are carried out.

5.1 FRAME EXTRACTION

Extraction of high-resolution frames from the video sequences is the primary step. It involves extracting frame or set of frames that have a good representation of a shot. The application allows recognizing objects from images recorded by the camera of a mobile device. The object recognition algorithm is insensitive to image registration parameters, i.e. scale, rotation and lighting conditions. Moreover, the recognized object is robustly detected and localized in the image context (e.g. among other similar objects). An RGB image is captured using the mobile phone camera. First, this image is blurred to reduce effect of noise. A 5X5 kernel is used for the same and average of the R, G and B values for the 25 pixels is found and applied to the central pixel. Blurring is done in order to reduce defects in the image such as small dark spots. This image is then converted to Grayscale format. Thus, now only the intensity information of the image is contained in the pixel. This image which is made up of shades of grey is obtained by iteratively taking the average of the R, G and B values for each pixel forming the image.

5.2 EDGE DETECTION

Edge detection is an image processing technique for finding the boundaries of objects within images. It works by detecting discontinuities in brightness. Edge detection is used for image segmentation and data extraction in areas such as image processing, computer vision, and machine vision. The Sobel operator is used for edge detection in this application. A 3X3 Sobel matrix is used, which is convoluted with the source image. The x-coordinate is defined here as increasing in the right direction changes, and the y-coordinate is defined as increasing in the down direction changes. According to this the gradient is approximated. These gradient approximation in the x and y direction is used to calculate Gradient Magnitude. After calculating Gradient for the image, the required Edges are detected. The required features of the image (i.e. object to be detected) are extracted using thresholding method. In this method, a binary image is generated with only black and white colour. The required features are converted to white (or black) and the background is set to black (or white), respectively as shown in Figure 5.1.

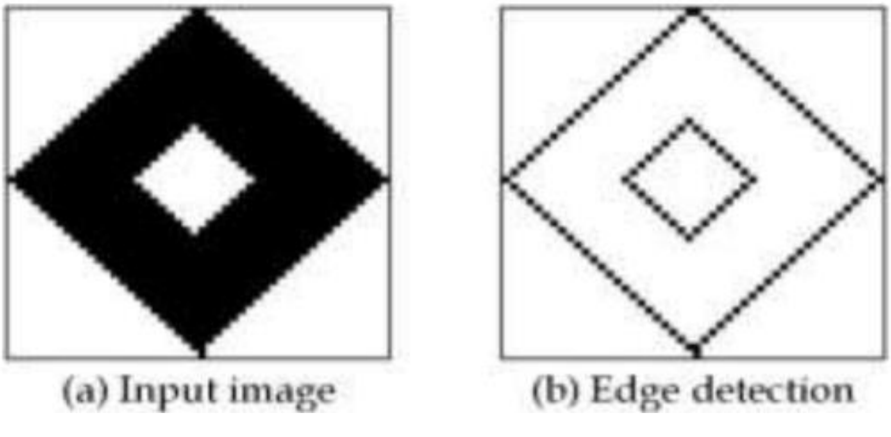


Figure 5.1 Detection of Edges from Input Image

5.3 THRESHOLDING

Thresholding is the simplest method of image segmentation. From a grayscale image, thresholding is used to create binary images with only black or white colours. It is used for feature extraction where required features of image are converted to white and everything else to black (and vice-versa). Figure 5.2 represents a thresholder image made up of only black and white colours.

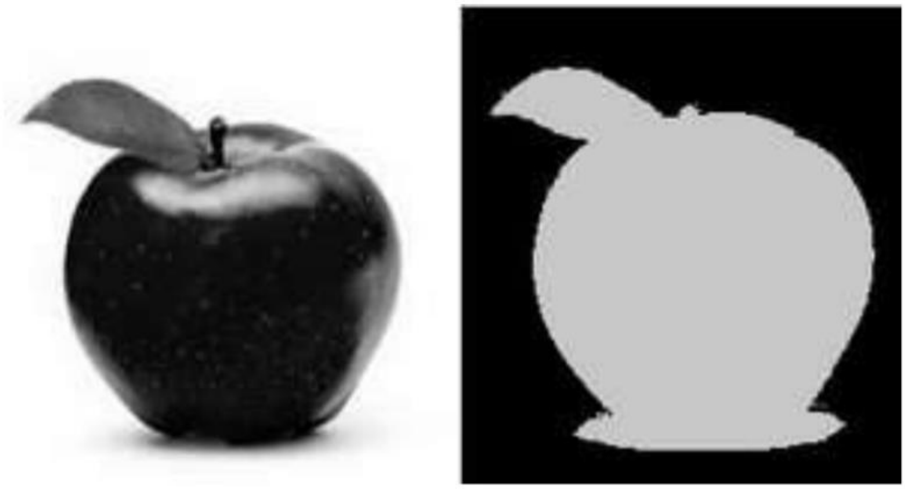


Figure 5.2 Image Segmentation by Using Threshold Method

5.4 BLOB DETECTION

The next operation performed is blob detection. Blob detection refers to mathematical methods that are aimed at detecting regions in a digital image that differ in properties, such as brightness or colour, compared to areas surrounding those regions. A blob is a region of a digital image in which some properties are constant or vary within a prescribed range of values.

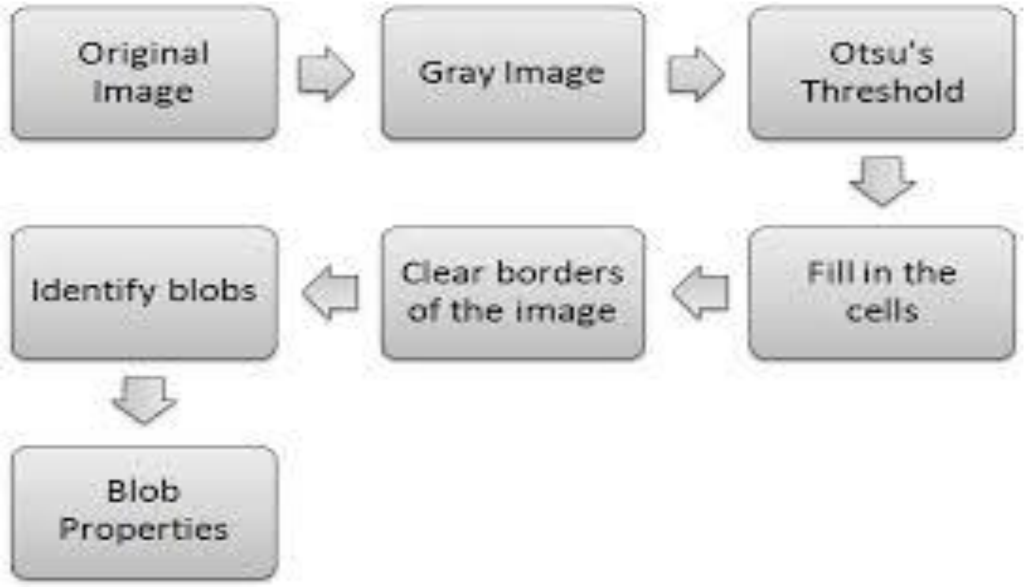


Figure 5.3 Blob Detection Process Flow

Figure 5.3 shows the process flow of blob detection. All the points in a blob are considered in some sense to be similar to each other. The foreground (say black) and back ground (white) of an image are separated in thresholding process. Hence by using those intensity values, x-coordinates of an image are compared to find x min and x max. Similarly, for Y-axis y min and y max are found and stored in a matrix and a blob is drawn over an object. Further, a key point detection procedure is performed. To improve performance of the application the Features from Accelerated Segment Test (FAST) algorithm is implemented. It is one of the fastest corner detection algorithms. The FAST corner detector is commonly used to track objects in different frames. That is, FAST corner detector algorithms extract feature information, and rotation and movement information in different frames is computed through feature matching, which is often based on a distance between the vectors, e.g. the Euclidean distance of feature vectors. In the application, corner information is extracted from the input image using the FAST corner detector and objects are recognized via BPNN (Back Propagation Neural Networks) machine learning.

5.5 HISTOGRAM

Histogram Key points descriptor obtained from SIFT method as shown in Figure 5.4 is based on gradient magnitudes computed for 16 or 4 pixels adjacent to a key point. These values are used to form the histogram. By looking at the histogram for a specific image a viewer is able to judge the entire tonal variation of an image. The left side of the horizontal axis represents the black and dark areas, the middle represents medium grey and the right-hand side represents white area. Thus, if an image is dark then majority points are located in the left side of the histogram similarly the grey points and the lighter points of an image is located respectively in the middle and the right side of a histogram.

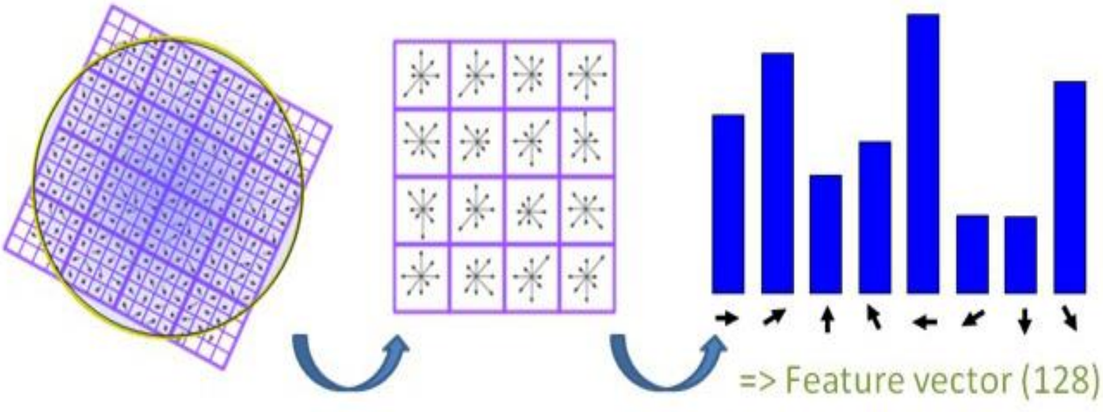


Figure 5.4 Histogram Key Points Descriptor Obtained from Sift Method

CHAPTER 6

OBJECT RECOGNITION

The third module deals with the recognition of the detected object. Object recognition refers to a collection of related tasks for identifying objects in digital photographs. Region-Based Convolutional Neural Networks, or R-CNNs, are a family of techniques for addressing object localization and recognition tasks, designed for model performance. You Only Look Once, or YOLO, is a second family of techniques for object recognition designed for speed and real-time use. This is deployed in the application to enhance performance.

6.1 RECOGNITION

The classical problem in computer vision, image processing, and machine vision is that of determining whether or not the image data contains some specific object, feature, or activity. Different varieties of the recognition problem are:

- **Object recognition** (also called **object classification**) – one or several pre-specified or learned objects or object classes is recognized, together with their 2D positions in the image or 3D poses in the scene.

- **Identification** – an individual instance of an object is recognized. Examples include identification of a specific person's face or fingerprint, identification of handwritten digits, or identification of a specific vehicle.
- **Detection** – the image data are scanned for a specific condition. Examples include detection of possible abnormal cells or tissues in medical images or detection of a vehicle in an automatic road toll system. Detection based on relatively simple and fast computations is sometimes used for finding smaller regions of interesting image data which is further analysed by more computationally demanding techniques to produce a correct interpretation.

Figure 6.1 shows the flow diagram of the recognition process used in the proposed system.

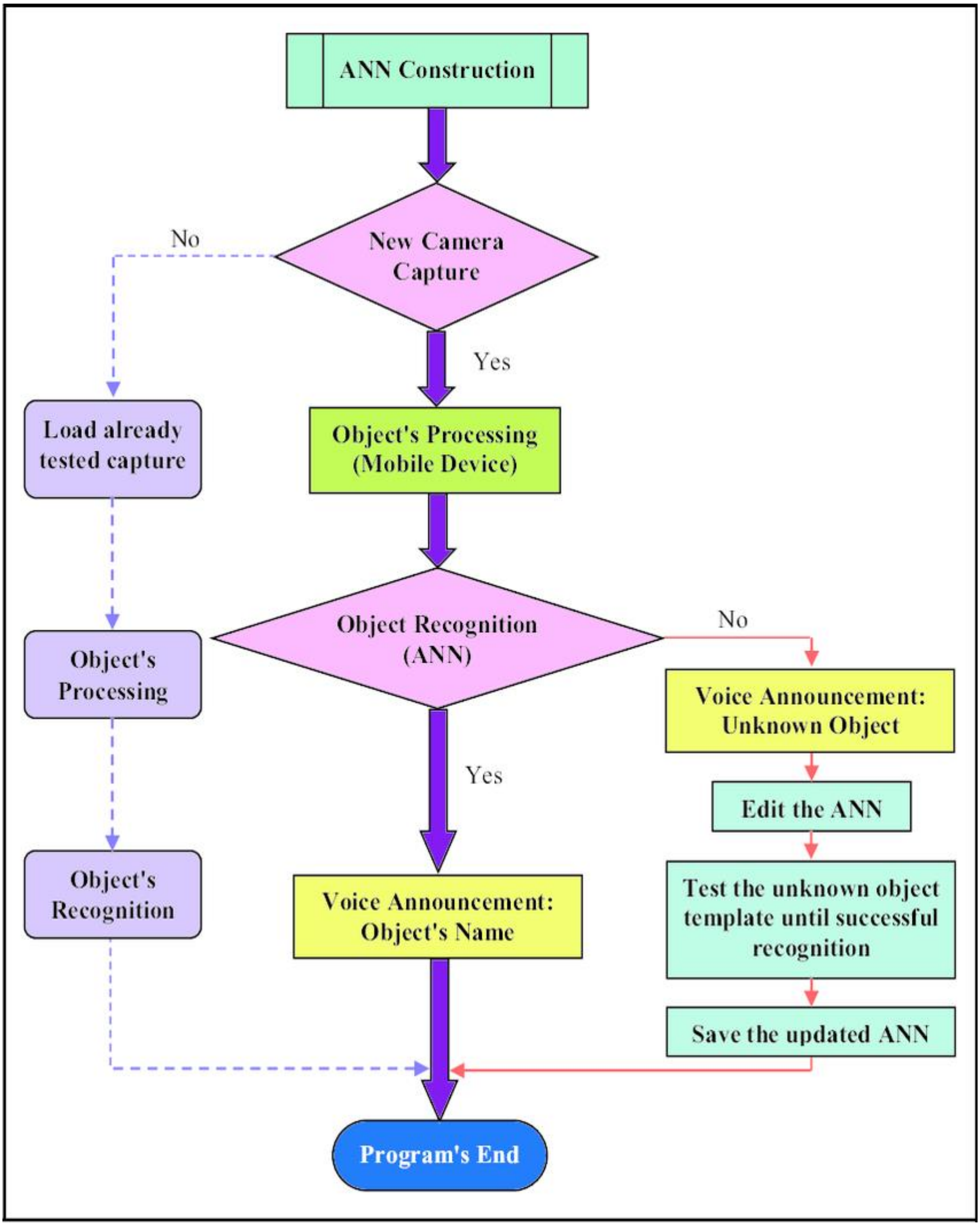

ANN Construction
New Camera Capture
No
Yes
Load already tested capture
Object's Processing (Mobile Device)
Object's Processing
Object Recognition (ANN)
No
Yes
Voice Announcement: Unknown Object
Edit the ANN
Object's Recognition
Voice Announcement: Object's Name
Test the unknown object template until successful recognition
Save the updated ANN
Program's End

Figure 6.1 Flow Diagram of Recognition Process

6.2 BACKGROUND REMOVAL

When the frame is captured from the video it is converted to pixels, the most repeated pixel is removed because it is a background image, it is done to make easy recognition of object soon. In this module the colour histogram of each frame is constructed and the colours that appear most frequently in the scene are removed. These removed pixels do not need to be considered in subsequent detection processes. Performing background colour removal cannot only reduce object info but also speed up the detection process. Figure 6.2 illustrates the block subtraction algorithm used for background removal.

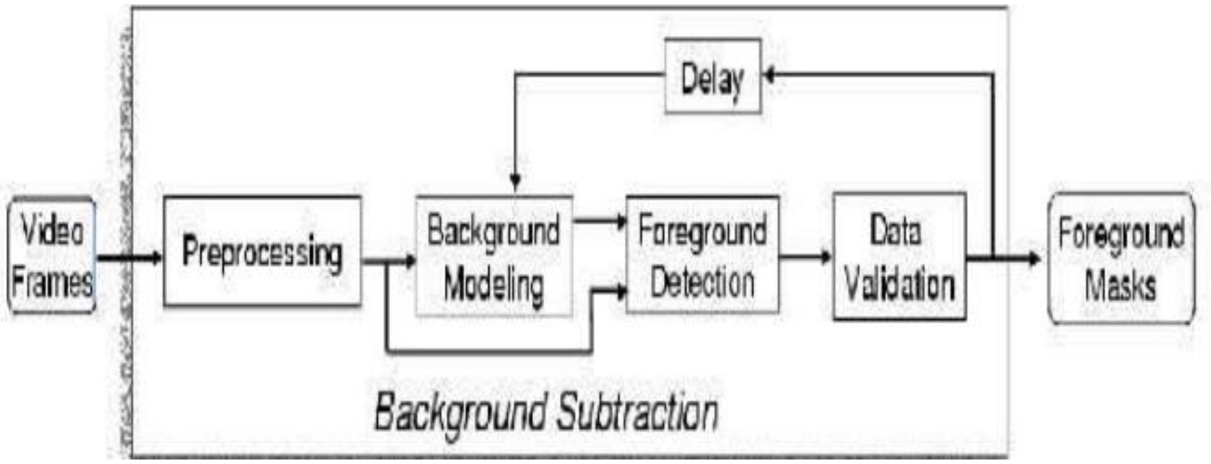


Figure 6.2 Flow Diagram of Generic Block Subtraction Algorithm

6.3 OBJECT EXTRACTION

After the background is removed from the captured image, object extraction is done. The detailed process of object extraction is described in figure 6.3 below.

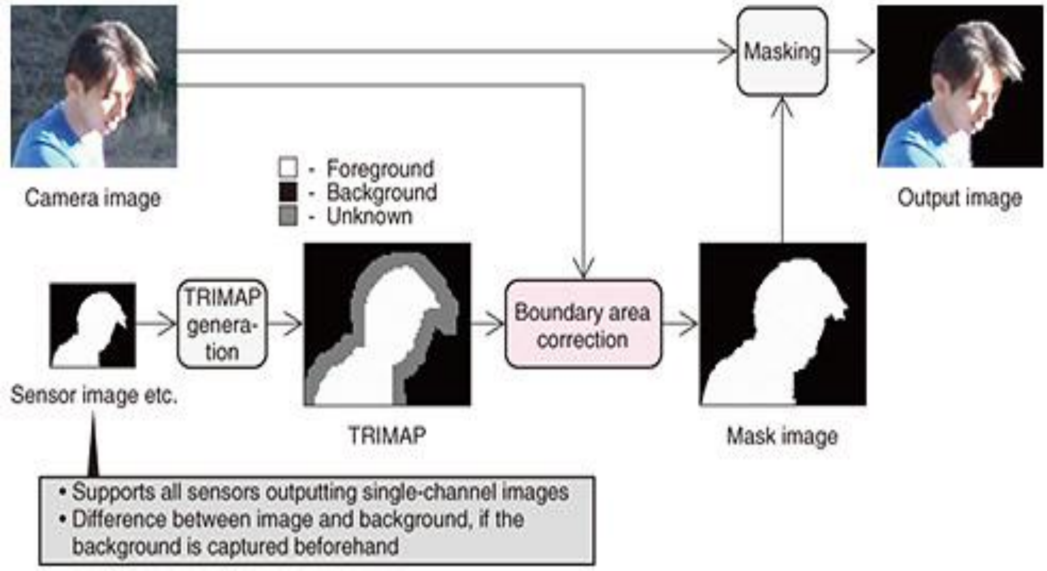


Figure 6.3 The Detailed Process Flow of Object Extraction

A video stream is taken using camera and is processed to estimate motion. The aim of motion detection is to identify moving objects from a sequence of image frames especially traffic which will alert the blind user. Several approaches have been proposed to the problem of motion segmentation. In conventional video surveillance systems, most of the motion detection is performed by using thresholding methods video stream is nothing but the stream of images taken continuously after 0.25 seconds. These images are analysed to detect motion and on successful detection the user is alerted about it. Along similar lines, users are alerted when the motion (specifically moving vehicle) has stopped and there is no motion in front of the camera. As seen in the object detection module, the image is blurred to reduce effect of noise. A 5X5 kernel is used for the same and average of the R, G and B values for the 25 pixels is found and applied to the central pixel. Blurring is done in order to reduce defects in the image such as small dark spots. This image is then converted to Grayscale format. Thus, now only the intensity information of the image is contained in the pixel. This image which is made up of shades of grey is obtained by iteratively taking the average of the R, G and B values for each pixel forming the image.

CHAPTER 7

TEXT DESCRIPTION

After the object is correctly recognised, a text description of the same is generated. Text description of the image is obtained by Image Captioning. Image Captioning is the process of generating textual description of an image. It uses both Natural Language Processing and Computer Vision to generate the captions.

7.1 SPEECH SYNTHESIS

Speech Synthesis is the artificial production of human speech by machine on the basis of written input. A computer system used for this purpose is called a speech computer or speech synthesizer, and is implemented in software or hardware products. A text-to-speech (TTS) system converts normal language text into speech. Figure 7.1 illustrates the process of the generation of audio feedback used in the application.

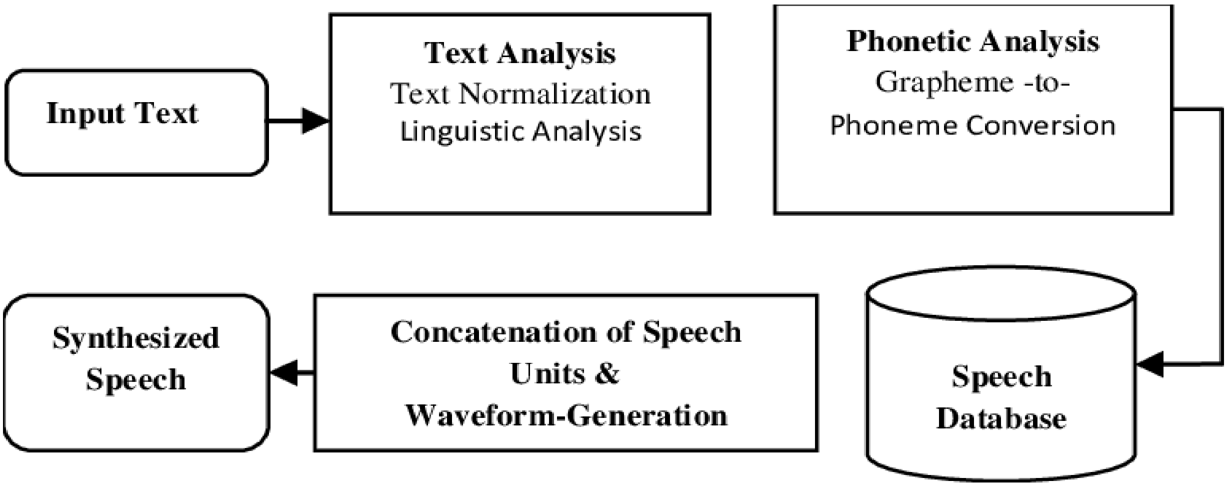


Figure 7.1 Generation of Audio Feedback

CHAPTER 8

IMPLEMENTATION AND EXPERIMENTAL RESULTS

8.1 SAMPLE DATASET

COCO is a large-scale object detection, segmentation, and captioning dataset. COCO has several features like Object segmentation Recognition in context, Super pixel stuff segmentation, 330K images (>200K labelled), 1.5 million object instances, 80 object categories, 91 stuff categories, 5 captions per image, 250,000 people with key points. Figure 8.1 shows the COCO Dataset labels.

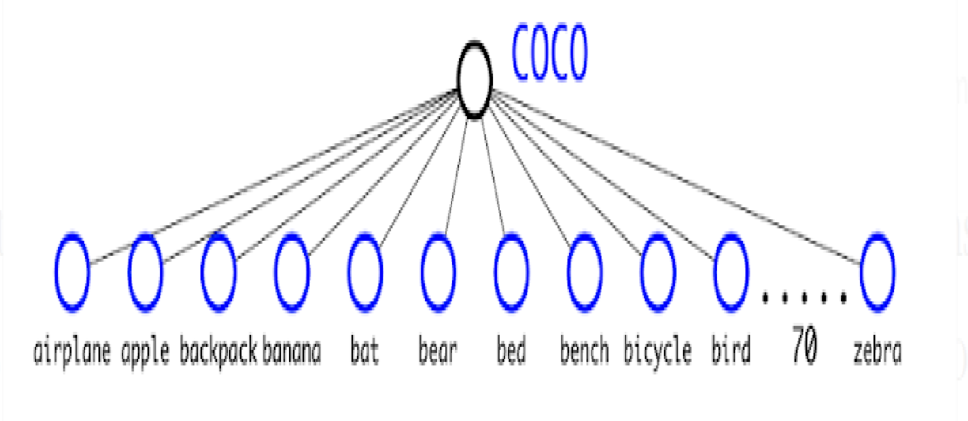


Figure 8.1 COCO Dataset Labels

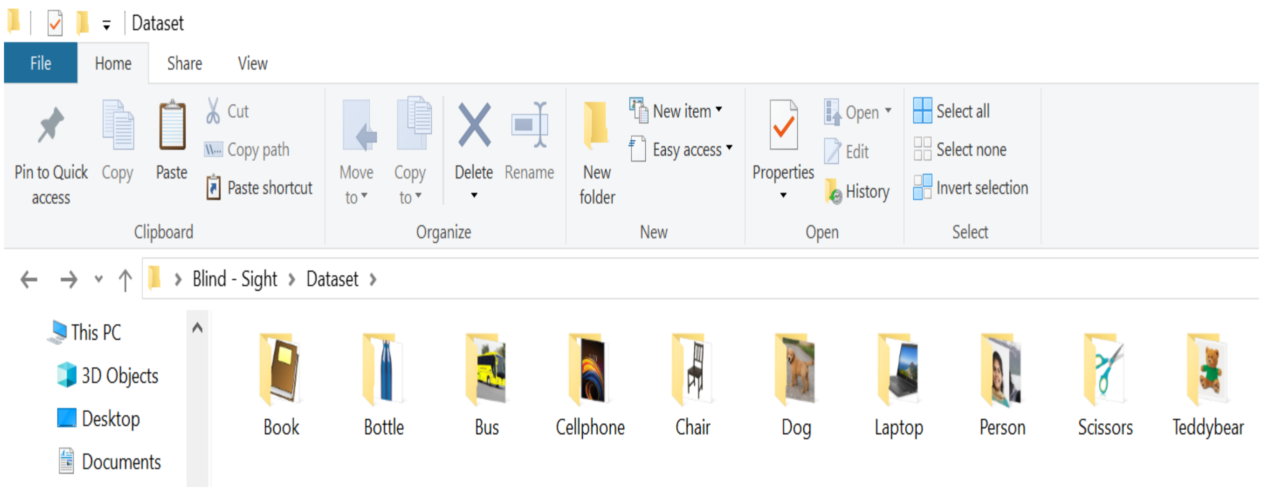


Figure 8.2 Sample Dataset

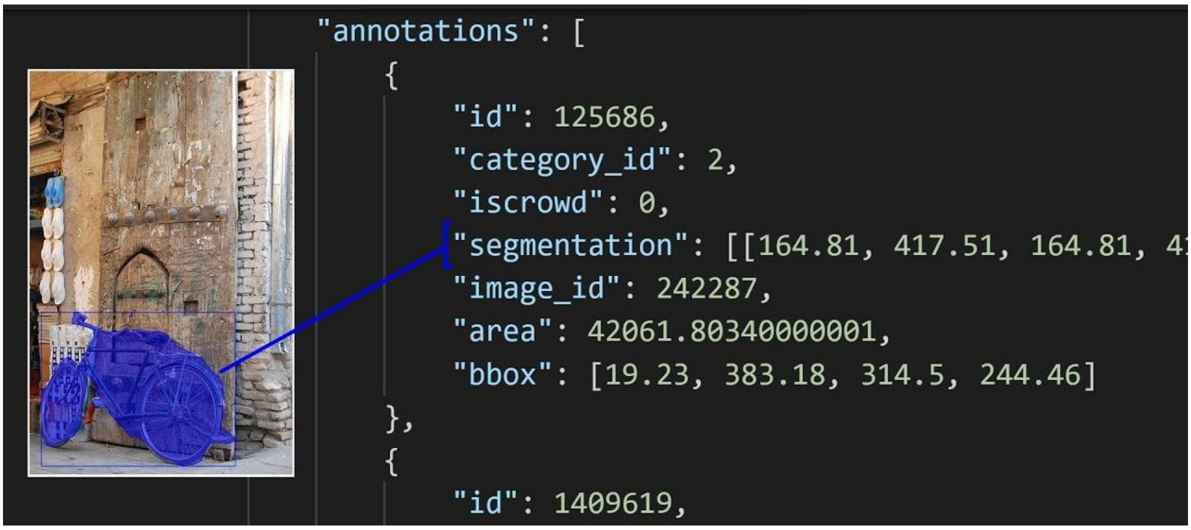
8.2 COCO DATASET FORMAT

The COCO dataset is formatted in JSON and is a collection of "info", "licenses", "images", "annotations", "categories" (in most cases), and "segment info" (in one case).

```
{
        "info": {...},
        "licenses": [...],
        "images": [...],
        "annotations": [...],
        "categories": [...], <-- Not in Captions annotations
```

"segment_info": [...] <-- Only in Panoptic annotations

}



```
"annotations": [
    {
        "id": 125686,
        "category_id": 2,
        "iscrowd": 0,
        "segmentation": [[164.81, 417.51, 164.81, 4
        "image_id": 242287,
        "area": 42061.80340000001,
        "bbox": [19.23, 383.18, 314.5, 244.46]
    },
    {
        "id": 1409619,
```

Figure 8.3 COCO Dataset Format

8.3 SAMPLE OUTPUT

The following is the result of object recognition. Here the system correctly identifies the objects with the following accuracies:

- Person: 0.9962
- Cell Phone: 0.9595
- Bottle: 0.9937
- Chair: 0.73

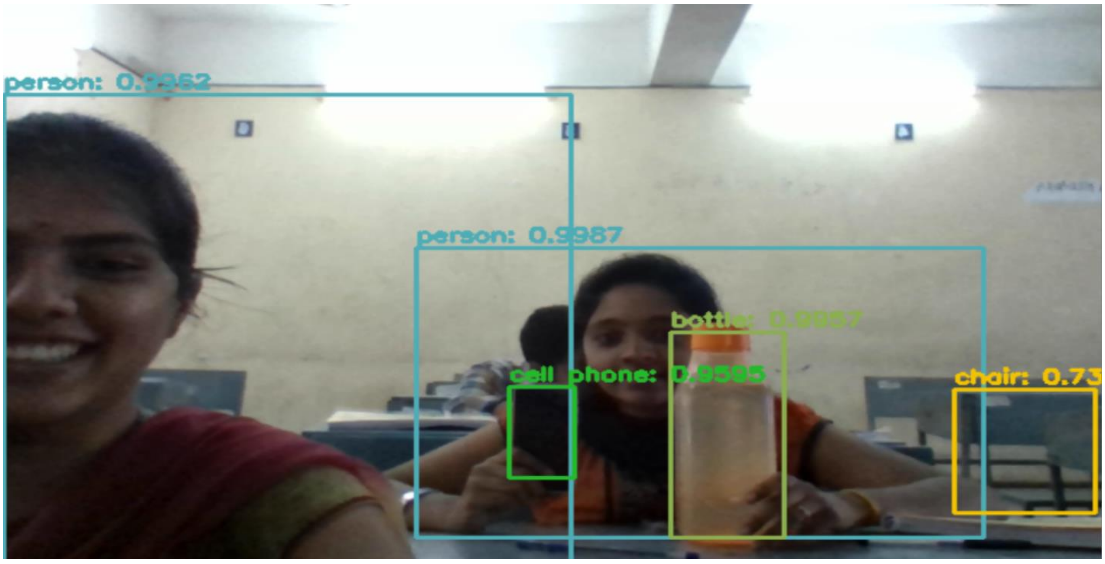
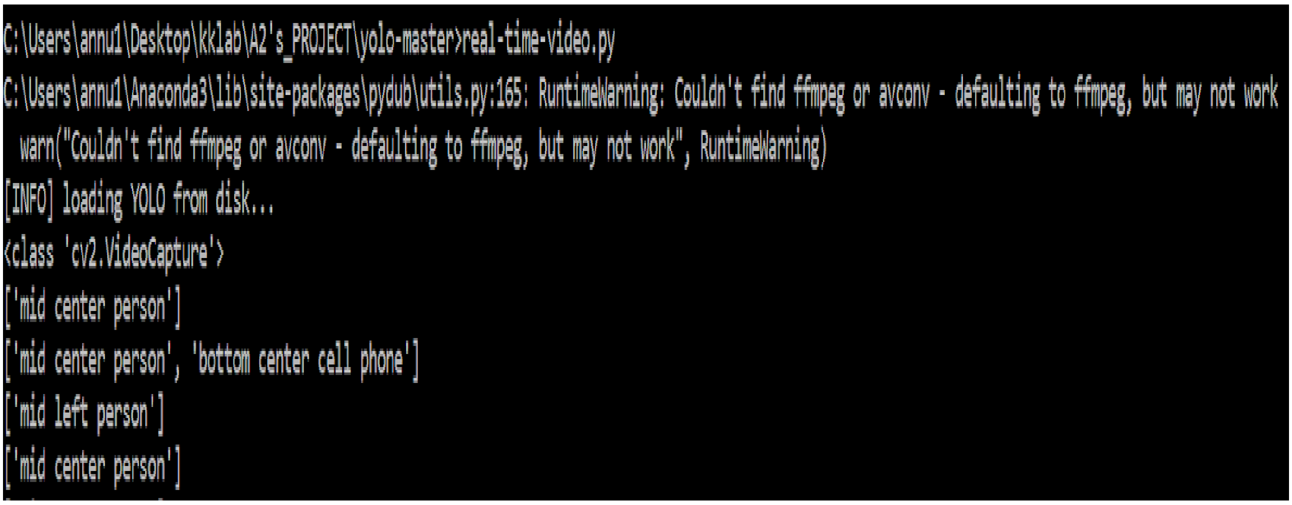


Figure 8.4 Object Recognition Output

8.4 TEXT OUTPUT WITH CO-ORDINATES

The textual annotation is generated along with the position of the objects from the detected image. This is then used to generate the audio feedback.



```
C:\Users\annu1\Desktop\kklab\A2's_PROJECT\yolo-master>real-time-video.py
C:\Users\annu1\Anaconda3\lib\site-packages\pydub\utils.py:165: RuntimeWarning: Couldn't find ffmpeg or avconv - defaulting to ffmpeg, but may not work
  warn("Couldn't find ffmpeg or avconv - defaulting to ffmpeg, but may not work", RuntimeWarning)
[INFO] loading YOLO from disk...
<class 'cv2.VideoCapture'>
['mid center person']
['mid center person', 'bottom center cell phone']
['mid left person']
['mid center person']
```

Figure 8.5 Output With Coordinates

CHAPTER 9

CONCLUSION AND FUTURE WORK

9.1 CONCLUSION

Thus, an application that helps visually challenged people to hear the objects that are in front of them is developed. Advancement in technology is effectively put to use employing the latest object detection techniques and combining with Google Text to Speech API. It works better than the other existing aids for visually challenged people since it uses YOLO Object Detection which is incredibly fast and can process 45 frames per second. YOLO also understands generalized object representation. This is one of the best algorithms for object detection. As we are working on Social cause, application will fulfil the requirement of blind person. In other words, technically we are donating the eye to blind person.

9.2 FUTURE WORK

With over 39 million visually impaired people worldwide, the need for an assistive device that allows the blind user navigate freely is crucial. An off-line navigation device that uses 3-D sounds to provide navigation instructions to the user will be developed in alignment with this application as future work.

REFERENCES

[1] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi (2015): “You Only Look Once: Unified, Real-Time Object Detection”

[2] KedarPotdar, Chinmay D. Pai, SukrutAkolkar (2018): “A Convolutional Neural Network based Live Object Recognition System as Blind Aid”

[3] J.Prakash, P.Harish, Ms.K.Deepika (2015): “Android Based Object Recognition Into Voice Input To Aid Visually Impaired.”

[4] Xinyi Zhou ; Wei Gong ; WenLong Fu ; Fengtong Du (2017) : “The Application Of Deep Learning In Object Detection.”

[5] Liming Wang1, Jianbo Shi, Gang Song, and I-fan Shen1 (2017): “Object Detection Combining Recognition and Segmentation”

[6] O. Karaali, G. Corrigan, I. Gerson, and N. Massey (1998): "Text-To-Speech Conversion with Neural Networks: A Recurrent TDNN Approach”

[7] Younggun Lee 1 Taesu Kim 1 Soo-Young Lee 2 (2018): “Voice Imitating Text-to-Speech Neural Networks”

[8] Naihan Li, Shujie Liu, Yanqing Liu, Sheng Zhao, Ming Liu, Ming Zhou (2018) : ‘ Neural Speech Synthesis with Transformer Network’

[9] Xinyi Zhou; Wei Gong ; WenLong Fu ; Fengtong Du (2017) : Application of Deep Learning in Object Detection.