THIRD EYE: OBJECT RECOGNITION AND SPEECH GENERATION FOR VISUALLY IMPAIRED

A Project Report Submitted

in Partial Fulfilment of the Requirements

for the Degree of

Bachelor of Technology

in

Computer Science and Engineering

by

Yarlagadda Sai Bhavadeesh (Roll No. 2018BCS0082)

Peddi Shwejan (Roll No. 2018BCS0047)

Allu Harsha Vardhan (Roll No. 2017BCS0005)

Lavanya S (Roll No. 2017BCS0034)

Indian Institute of Information Technology Kottayam

to

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY

KOTTAYAM-686635, INDIA

November 2021

DECLARATION

We, **Yarlagadda Sai Bhavadeesh** (**Roll No: 2018BCS0082**), **Peddi Shwejan** (**Roll No: 2018BCS0047**), **Allu Harsha Vardhan** (**Roll No: 2017BCS0005**), **Lavanya S** (**Roll No: 2017BCS0034**), hereby declare that, the report entitled **"Third Eye: Object Recognition And Speech Generation For Visually Impaired"** submitted to Indian Institute of Information Technology Kottayam towards partial requirement of **Bachelor of Technology** in **Computer Science and Engineering** is an original work carried out by us under the supervision of **Dr. Koppala Guravaiah** and has not formed the basis for the award of any degree or diploma, in this or any other institution or university. We have sincerely tried to uphold the academic ethics and honesty. Whenever an external information or statement or result is used then, that have been duly acknowledged and cited.

Kottayam-686635

November 2021

Yarlagadda Sai Bhavadeesh - 2018BCS0082

Peddi Shwejan - 2018BCS0047

Allu Harsha Vardhan - 2017BCS0005

Lavanya S - 2017BCS0034

CERTIFICATE

This is to certify that the work contained in this project report entitled **"Third Eye: Object Recognition And Speech Generation For Visually Impaired"** submitted by **Yarlagadda Sai Bhavadeesh** (**Roll No: 2018BCS0082**), **Peddi Shwejan** (**Roll No: 2018BCS0047**), **Allu Harsha Vardhan** (**Roll No: 2017BCS0005**), **Lavanya S** (**Roll No: 2017BCS0034**) to Indian Institute of Information Technology Kottayam towards partial requirement of **Bachelor of Technology** in **Computer Science and Engineering** has been carried out by them under my supervision and that it has not been submitted elsewhere for the award of any degree.

Kottayam-686635	(Dr. Koppala Guravaiah)
November 2021	Project Supervisor

ABSTRACT

Visually impaired people face a lot of difficulties in doing their daily activities. There is a say that, Out of all the five sense organs, eyes are most important. The eyes are one of our most vital sense organs: 80% of what we perceive comes from our sense of sight [9]. Visually impaired need the help of either the third person or a stick. These methods are not always fruitful. Detecting and recognizing the objects and generating speech about the objects helps visually impaired in a great way in understanding their surroundings.

We aim to assist the visually impaired to travel independently with the ability to identify objects in their path, and the ability to generate speech describing the objects detected in the scene. The thesis employs training on YOLO (You Only Look Once) v5, Convolutional Neural Network (CNN) model for object detection. YOLO v5 is trained on custom dataset of 15 objects, along with MS COCO 2017 Dataset of 80 objects (95 objects overall). The output labels of the model are transformed to text and later converted to audio format and are presented to the visually impaired, through a speaker. We compared two python libraries for audio conversion, one is pyttsx3, and the other is gTTS. Pyttsx3 works offline, where as gTTS requires active internet connection. gTTS needs additional packages to play the audio, where as pyttsx3 has inbuit functions to play the audio.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Visually Impaired face a lot of difficulties in their daily lives. According to World Health Organization (WHO), nearly 2.2 billion individuals have a close or faraway vision impairment. Out of them, 49.1 million individuals are visually impaired. Yet the growth of the population is making a substantial improvement in the number of people affected. There are significant inter-regional and gender disparities, highlighting the need to scale up vision impairment prevention programs at all levels.

The visually impaired always need the help of either a stick or a person. Early-onset severe vision impairment can restrict a child's verbal, emotional, social, and cognitive development, which can have long-term effects. Vision impairment critically impacts the quality of life among the adult population. Social isolation, difficulty walking, a higher risk of falls and fractures, and premature admission to a nursing or care home can all result from vision impairment in older people. As a result, we decided to take on this project to assist visually impaired persons in recognizing their surroundings.

In recent years, deep learning has become a more popular technique for solving these problems of identifying objects. The deep learning systems achieve high accuracy rates at a lower cost. Many Convolutional Neural Network (CNN) methods like Single Shot Detector (SSD) and You Only Look Once (YOLO) are used to solve detection and recognition issues. There are other architectures such as Faster R-CNN and Mask R-CNN [37]. In this project, we used the YOLO v5 algorithm. Implemented a custom-made dataset including MS COCO 2017 dataset to attain good accuracy, and performance. After detecting and recognizing the objects, we generated speech for the recognized objects. This was achieved by using two popular Python libraries, called pyttsx3, and gTTS.

YOLO is an abbreviation for the term "You Only Look Once". YOLO algorithm detects and identifies diverse objects in a picture. YOLO uses convolution neural networks to deliver real-time object detection. YOLO v5 is faster, more accurate, and light-weight compared to other versions of YOLO. It has been used in various applications such as autonomous car driving, etc. YOLO v5 is one of the finest known models for Object Detection at the moment.

The detected object labels are converted to text. This text gives brief information about where the object is located in the view (let's say if the object is a person and is at the center of the screen, then the audio will be "mid-center person"). We have compared two libraries, i.e., pyttsx3, and gTTS.

Chapter 2

Literature Survey

Many works have been done on making life better for the visually impaired. There is various equipment for the visually impaired, such as sensor-powered walking sticks, speaking calculators, etc.

Rajwani, Roshan et al. [28] presented a system where the input is taken through an android camera, then the captured image is preprocessed using OpenCV, then the classification and identification is done in Cloud Vision API. Elmannai, Wafa M and Khaled M. Elleithy [11] proposed a system for object detection, where two camera sensors are used, which are then analyzed using computer vision methods. The ORB and KNN are used for object detection. Ye, Cang and Xiangfei Qian [36] in 2018, a 3-D Object Recognition for Visually Impaired people is proposed. The cane used by blind people is attached with a CV enhanced 3D Camera, it captures a 3D point cloud which is segmented into planar segments, which are then classified using Gaussian Model Mixture and clustered into the target objects. Bashiri, Fereshteh S et al. [6] proposed a system where the input is taken through a Google Glass Device, then classification and identification are done using Support Vector Machine Algorithm. Gianani, Sejal et al. [13] came up with a system where the image is captured through a camera device for the input and preprocessed using OpenCV. They used the SSD framework in conjunction with the MobileNet architecture. Nishajith, A et al. [21] suggested a framework that uses Raspberry Pi which has a Pre-trained CNN network. The image is captured through Noir Camera and preprocessing is done through OpenCV and they used Pre-trained object detection model 'ssd_mobilenet_v1_coco_11_06_2017' to classify the objects and text to speech conversion is done using eSpeak. Patel, Charmi T et al. [24] presented a technology where the image is captured through a USB webcam and preprocessing is done and it classifies and identifies the objects using the SVM Algorithm. Tosun, Selman and Enis Karaarslan [33] proposed a system where the image is captured using the android platform and preprocessing is done using OpenCV and Tiny YOLO is used for object detection which gives the audio output.

Wong, Yan Chiew et al. [34] In 2019, a real-time CNN-based object identification system for visually impaired people was proposed. The object group was filmed in real-time with a webcam, and the picture function was turned off. Then, to detect the sight of visually handicapped people, a sound-based detector was devised. Nasreen, Jawaid et al. [20] presented a system for guiding visually impaired people through the process of item detection. The developed method imports a picture from the back camera into a website and sends it to the server, where the YOLO model is utilised to recognise the objects on the server side. Pardasani, Arjun et al. [22] presented a technology that is wearable like smart glasses and shoes. Both smart shoes and glasses detect the obstacle and pass an audio output to the user. Rahman, Ferdousi et al. [26] developed a visually impaired object detection model based on the YOLO algorithm. For the building model, MTCNN is used. The YOLO Algorithm and MTCNN Networking are used for object identification and facial recognition, respectively. Shah, Samkit et al. [29] compared different detection algorithms to detect multiple objects and they found that Haar Cascade is the fastest and CNN gives more accuracy. Jhinkwan, Piyush et al. [14] proposed a system that uses a convolutional network combined with fully connected layers. Chen, Xiaobai et al. [10] created an automatic DCNN quantization approach to decrease the data range to 4 or 5 bits. Sun, Minghui et al. [31] presented a data collection system based on Google Tango, which has an infrared (IR) sensor built in.

Afif, Mouna et al. [1] in 2020, introduced YOLO v3, on a custom dataset that has 16 indoor object classes. They attained 73.19% mAP, they focused on indoor navigation. Afif, Mouna et al. [2] later proposed a framework on deep CNN "RetinaNet" for detecting indoor objects, which showed better results than their earlier work. Fang, Wei et al. [12] introduced a method using the Tinier-YOLO model, which is 4 times smaller than Tiny-YOLO v3. trained on PASCAL VOC and COCO datasets. It's faster than other lightweight models. Li, Yongjun et al. [16] proposed another version of YOLO, that is YOLO-ACN, which showed better results. They mainly focused on small objects detection. Bhole, Swapnil and Aniket Dhok [7] proposed a transfer learning on Single-Shot Detection (SSD) mechanism for object detection, and implemented it for human as well as currency detection. They achieved

90.2% accuracy on currency detection. Yohannes, Ervin et al. [37] introduced a method to assist the visually impaired around an outdoor environment. They designed a model using DarkNet-53 as a backbone, input is taken from a ZED stereo camera, and the model is trained on PASCAL VOC and MS COCO datasets. Joshi, Rashika et al. [15] mentioned a method using Mobile Net SSD, and the images are taken using Jetson Nano, and PiV2 camera, and trained on PASCAL VOC dataset. Achieved pretty good results with the proposed model.

Atikur Rahman and Sheikh Sadi [27] proposed an IoT-enabled Automated Object Recognition using SSD Model, SIFT, and MS COCO dataset in 2021. Balachandar, Santhosh et al. [5] developed a technique in which a multi-view object tracking (MVOT) system is employed to address several cameras monitoring and capturing videos in this proposed system. And, by merging the information from the videos, a powerful and precise framework is created. Using the YOLO v3 algorithm, each segmented group of objects in one view is mapped to the equivalent group in another view. Blob gathers, which allow data to be transferred across cameras, corresponded to these agreeing sets. After being taken by the camera, these visuals are converted into vocal output. Mansi Mabendru and Sanjay Kumar Dubey [19] created a system employing two separate algorithms, YOLO and YOLO v3, and tested accuracy and performance. The SSD Mobile Net model is utilised in the YOLO Tensor flow, The Darknet model is used in YOLO v3. The python library gTTS is used to transform sentences into audio for the audio Feedback. Kanchan Patil et al. [25] proposed a wearable device with a virtual assistant system for visually challenged people, with a total of five components integrated into one system. These components can be navigated via hardware buttons and voice-over commands provided by the user. Mohana Priya et al. [4] presented a voice-based image caption generation, which is a task that requires the use of natural language processing.The best option in this project is a combination of CNN and LSTM; the major goal of this proposed study work is to produce the perfect caption for an image. The description will be transformed into text, and the text will be converted into a voice. You Only Look Once (YOLO) a Real-Time Object Detection is deployed by Annapoorani et al. [3] proposed a model where the image features are identified using image classification techniques, and the Indian money identification module is utilised to identify the denominations. Using the gTTS package, the written description of the identified object will be transmitted to the gTTS API. Sandeep Pandasupuleti et al. [23] proposed Voice Translation and Image Recognition using VCC, LSTM, and Flickr_8k dataset. The following table (table 2.1) describes about the methods and pros & cons discussed in literature.

Table 2.1: Trends & Technologies discussed in literature

Paper Title	**Authors**	**Methods**	**Pros & Cons**
Proposed System on Object Detection for Visually Impaired People. [28]	Rajwani, Roshan, Dinesh Purswani, Paresh Kalinani, Deesha Ramchandani, and Indu Dokare	Android Camera, OpenCV, Google Cloud Vision API, Compare it with Microsoft COCO Dataset and give output.	Since the output is through Android application, it should have enough battery.
A Highly Accurate and Reliable Data Fusion Framework for Guiding the Visually Impaired. [11]	Elmannai, Wafa M., and Khaled M. Elleithy	Two camera Sensors, Computer Vision Methods, Oriented FAST and Rotated BRIEF (ORB) and KNN Algorithm.	Accuracy of 96%, Used a motherboard connected with various sensors like gyro, compass, GPS, music, FEZ Spider board.
3-D Object Recognition of a Robotic Navigation Aid for the Visually Impaired [36]	Ye, Cang, and Xiangfei Qian	3D Camera(White Cane), Planar Segments, Gaussian Model Mixture	Trained on all indoor objects Accuracy over 90%

Continued on next page

Table 2.1: *Trends & Technologies discussed in literature ... Contd.*

Paper Title	**Authors**	**Methods**	**Pros & Cons**
Object Detection to Assist Visually Impaired People: A Deep Neural Network Adventure. [6]	Bashiri, Fereshteh S, Eric LaRose, Jonathan C. Badger, Roshan M. D'Souza, Zeyun Yu, and Peggy Peissig.	Marshfield Clinic Dataset, Google Glass Device, CNN Model, Support Vector Machine Algorithm	Limited number of objects (ex: doors, stairs, signs etc.,) Accuracy over 98%
JUVO - An Aid for the Visually Impaired [13]	Gianani, Sejal, Abhishek Mehta, Twinkle Motwani, and Rohan Shende	Camera,Image Capturing and Preprocessing, Object detection Using OpenCV, SSD Framework, MobileNet Architecture	Few objects in Dataset. Indoor Environment, Accuracy of 99.61%
Smart Cap-Wearable Visual Guidance System For Blind. [21]	Nishajith, A., J. Nivedha, Shilpa S. Nair, and J. Mohammed Shaffi.	Raspberry Pi Noir Camera, OpenCV Processing, COCO Model, eSpeak.	90 classes of objects in Dataset.

Continued on next page

Table 2.1: *Trends & Technologies discussed in literature ... Contd.*

Paper Title	**Authors**	**Methods**	**Pros & Cons**
Multisensor – based Object Detection in Indoor Environment for Visually Impaired People [24]	Patel, Charmi T, Vaidehi J. Mistry, Laxmi S. Desai, and Yogesh K. Meghrajani.	USB Webcam, Preprocessing, Statistical Analysis, SVM Classifier.	It can be used for outdoor environment but it is tested for indoor environment only.
Real-Time Object Detection Application for Visually Impaired People: Third Eye. [33]	Tosun, Selman, and Enis Karaarslan	Camera, OpenCV Processing, Tiny YOLO TensorFlow, Audio Output, COCO Dataset	Only 20 classes in the dataset, Manual selection.
Convolutional Neural Network for Object Detection System for Blind People. [34]	Y.C. Wong, J.A. Lai, S.S.S. Ranjit, A.R. Syafeeza, N. A. Hamid	Cnn, Used edge box algorithm, Caffnet model, softmax Cifar10 dataset has been used	The object detection models faced difficulty in classifying the object from a picture of ultimate scale

Continued on next page

Table 2.1: *Trends & Technologies discussed in literature ... Contd.*

Paper Title	**Authors**	**Methods**	**Pros & Cons**
Object Detection and Narrator for Visually Impaired People. [20]	Jawaid nasrren, warsi, Arif, Asad ali shaikh, Yahya Muhammad, Monaisha abdullah.	Used YOLO.It narrates to the user. It was trained on Imagenet dataset	Results showed that the accuracy is varying depending on phone camera quality and the light effects. iPhone and Samsung have better results than others.
Smart Assistive Navigation Devices for Visually Impaired People. [22]	Arjun Pardasani, Prithviraj N Indi, Sashwata Banerjee, Aditya Kamal, Vaibhav Garg	Open CV, Image processing, Used Smart glass and shoes	Both the devices have been developed by using simple, cheap sensors. Their motive is to make both the devices as a part of the user's regular and frequently used objects.

Continued on next page

Table 2.1: *Trends & Technologies discussed in literature ... Contd.*

Paper Title	**Authors**	**Methods**	**Pros & Cons**
An Assistive Model for Visually Impaired People using YOLO and MTCNN [26]	FerdousiRahman, IsratJahanRitun, NafisaFarhin, JiaUddin	Open CV, YOLO algorithm, Deep learning	The object detection process achieved 6-7 FPS processing with an accuracy rate of 63-80.%
CNN based Auto-Assistance System as a Boon for Directing Visually Impaired Person. [29]	Samkit Shah, Jayraj Bandariya, Garima Jain, Mayur Ghevariya, Sarosh Dastoor	Haar cascade, CNN, Deep learning COCO 2017 data Set was used	When processed on CPU, Haar cascade is the fastest algorithm, but CNN gives more accurate results when detecting multiple objects simultaneously for real time applications.
Object Detection Using Convolution Neural Networks [14]	Piyush Jhinkwan, Vaishali Ingale, Shubham Chaturvedi	Deep learning, CNN, Back propagation algorithm. For training CIFAR-100 dataset was used	It was trained with dropout and data augmentation to achieve better results.

Continued on next page

Table 2.1: *Trends & Technologies discussed in literature ... Contd.*

Paper Title	**Authors**	**Methods**	**Pros & Cons**
A 68 mw 2.2 Tops/w low bit-width and multiplierless DCNN object detection processor for visually impaired people. [10]	Xiaobai Chen, Jinglong Xu	Deep convolutional network, low-bit, multiplierless	reducing hardware cost by over 68% compared to the 16 bit fixpoint model with negligible accuracy loss.
"Watch Your Step": Precise Obstacle Detection and Navigation for Mobile Users Through Their Mobile Service [31]	MINGHUI SUN PENGCHENG DING, JIAGENG SON, MIAO SONG5, AND LIMIN WANG	Google Tango, built-in infrared (IR) sensor to collect data	The system cannot correctly distinguish complex situations such as obstacles leaning against a wall.
Research on Small Target Detection in Driving Scenarios Based on Improved YOLO Network. [35]	Qiwei Xu, Runzi Lin, Han Yue, Hong Huang, Yun Yang, Zhigang Yao	YOLO v3, 2080 Ti machine, Dataset used is Apollo Scape (Baidu's autopilot dataset).	Improvised YOLO v3 and it showed better results compared to YOLO v3. Accuracy is 84.76%.

Continued on next page

Table 2.1: *Trends & Technologies discussed in literature ... Contd.*

Paper Title	**Authors**	**Methods**	**Pros & Cons**
Tinier-YOLO: A Real-Time Object Detection Method for Constrained Environments. [12]	Wei Fang, Lin Wang, Peiming Ren	Tinier-YOLO-v3, PASCAL VOC (2007 + 2012), COCO.	Faster runtime speed compared to other lightweight models. But, is suitable for embedded systems (Low accuracy).
YOLO-ACN: Focusing on Small Target and Occluded Object Detection. [16]	Yongjun Li, Shasha Li, Haohao Du, Lijia Chen, Dongming Zhang, Yao Li	YOLO-ACN, MS COCO, Infrared pedestrian dataset KAIST, NVIDIA Tesla K40.	Doesn't improve performance much with the proposed method, compared to YOLO v3. focused on small objects detection.
Object Recognition and Classification System for Visually Impaired. [15]	Rashika Joshi, Meenakshi Tripati, Amit Kumar, Manoj Singh Gaur.	MobileNetSSD (SSD - Single Shot-Detector), PASCAL VOC 2007.	Got pretty good accuracy, but the dataset is small, not sufficient. Only for embedded systems.

Continued on next page

Table 2.1: *Trends & Technologies discussed in literature ... Contd.*

Paper Title	**Authors**	**Methods**	**Pros & Cons**
An Evaluation of RetinaNet on Indoor Object Detection for Blind and Visually Impaired Persons Assistance Navigation. [2]	Mouna Afif, Riadh Ayachi, Yahia Said, Edwige Pissaloux, Mohamed Atri	RetinaNet (ResNet, DenseNet, VGGNet based), Self prepared Dataset (Contains 8000 images).	Attained 84.61% mAP. Focused on only indoor navigation. the number of objects it can detect is very small. Got good results with proposed algorithm.
Indoor object detection and recognition for an ICT mobility assistance of visually impaired people. [1]	Mouna Afif, Riadh Ayachi, Edwige Pissaloux, Yahia Said, Mohamed Atri	YOLO v3, DarkNet-53. Dataset contains 8000 images and contains 16 indoor object classes.	Attained 73.19% mAP, and it's only focused on indoor navigation. Used pretrained model and trained on the new dataset.

Continued on next page

Table 2.1: *Trends & Technologies discussed in literature ... Contd.*

Paper Title	**Authors**	**Methods**	**Pros & Cons**
Robot Eye: Automatic Object Detection and Recognition Using Deep Attention Network to Assist Blind People. [37]	Ervin Yohannes, Paul Lin, Chih-Yang Lin, Timothy K. Shih	Self-designed model (DarkNet-53 based), ZED Stereo camera, PASCAL VOC, MS COCO datasets.	Accuracy is 81%, better than YOLO v3. Used PASCAL VOC for classes, and mixed MS COCO. No-of classes are too small.
Deep Learning based Object Detection and Recognition Framework for the Visually-Impaired. [7]	Swapnil Bhole, Aniket Dhok	PASCAL VOC 2007 dataset, SSD, Inception v3 model.	Added currency detection to the dataset and achieved 90.2% acc. But the dataset contains only 20 classes.
IoT Enabled Automated Object Recognition for the Visually Impaired. [27]	Md. Atikur Rahman, Muhammad Sheikh Sadi	laser sensors, Single Shot Detector (SSD) model, SIFT, MS COCO dataset	YOLO accuracy is 95.99% and SSD 88.89%. YOLO seems to be better compare to SSD.

Continued on next page

Table 2.1: *Trends & Technologies discussed in literature ... Contd.*

Paper Title	**Authors**	**Methods**	**Pros & Cons**
Deep Learning Technique Based Visually Impaired People Using YOLO v3 Framework Mechanism. [5]	Balachandar, Santhosh, Suriyakrishna, Vigensh, Usharani, Manju Bala	YOLO v3, Cameras, M VOT, COCO dataset	They have used (videocon camera) its intra camera grahic. which does not highlights the features properly and exactly tally the model.
Real Time Object Detection with Audio Feedback using Yolo vs. Yolo_v3 [19]	Mansi Mahendru, Sanjay Kumar Dubey	Tensor flow, SSD, YOLO, YOLO v3, gTTS, Deep Learning	YOLO accuracy is 78.99 and YOLO v3 92.89% (seems to be better compare to YOLO).
Guidance System for Visually Impaired People. [25]	Kanchan Patil, Avinash Kharat, Pratik Chaudhary, Shrikant Bidgar, Rushikesh Gavhane	gTTS, YOLO v3, Pyttsx, AIML, Vice over chatbot	chat-bot cannot recognize the command in noisy environment, chat-bot may get confused between voice of an user and person nearby.

Continued on next page

Table 2.1: *Trends & Technologies discussed in literature ... Contd.*

Paper Title	**Authors**	**Methods**	**Pros & Cons**
Building A Voice Based Image Caption Generator with Deep Learning. [4]	Mohana priya R, Dr.Maria Anu, Divya	NLP ,CNN, LSTM (Long short term memory), RNN (recurrent neural network) flicker dataset, Accuracy 90%	The dataset is small. For better accuracy could be used big dataset, According to current trends, it's not sufficient.
Blind - Sight: Object Detection with Voice Feedback. [3]	A. Annapoorani, Nerosha Senthil Kumar, Dr. V. Vidhya	YOLO, COCO Dataset, gTTS	Live object recognition system cannot perform future learning which is a demerit.
Image Recognition and Voice Translation for Visually Impaired. [23]	Sandeep Pasupuleti, Lahari Dadi, Manikumar Gadi, R. Krishnaveni	Flickr_8k dataset, VGG, LSTM	Dataset is very small, the implementation can be enhanced by giving a greater number of images and text datasets with shorter captions for training.

Chapter 3

Proposed Work

Image is taken from camera, and passed to the trained YOLO v5 model, which returns the output tensor of detected objects. This output tensor is converted to text, and passed to a audio converter to provide audio feedback. Figure 3.1 describes the schematic diagram of proposed system.

YOLO is a real-time object identification technique that uses neural networks. Because of its speed and precision, this algorithm is very popular. It has been used to identify traffic signals, pedestrians, parking meters, and animals in a variety of applications.

YOLO is a regression-based technique that predicts classes and bounding boxes for the entire image in a single run of the algorithm, rather than selecting the interesting area of an image. Finally, we want to be able to forecast an object's class and the bounding box that defines its placement.

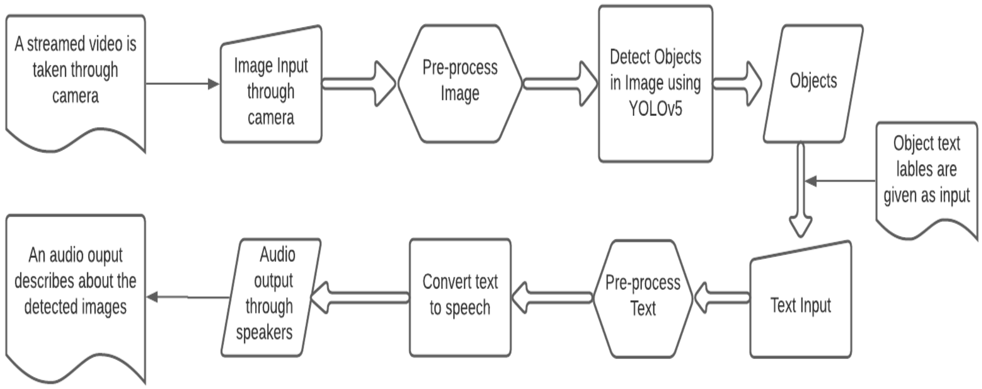

Figure 3.1: Schematic diagram of proposed system

3.1 Methods

The project uses YOLO algorithm that provides real-time object detection using neural networks and YOLO has different versions.

3.1.1 YOLO

YOLO starts with an input image, which is subsequently divided into grids by the framework (say a 3 X 3 grid). Each grid is subjected to image classification and localisation. The bounding boxes and their related class probabilities for items are then predicted using YOLO.

We will divide each image into different grids. For example, we divide an image into three 3 x 3 grids, and we want the items to be classified into three different classes. Pedestrian, Car, and Motorcycle are the three different classes. So, the label y for each grid cell will be an eight-dimensional vector. (as shown in figure 3.2) : pc determines whether or not an object exists in the grid (it is the probability), bx, by, bh, bw define the bounding box if an object is present, c1, c2, c3 represent the classes. For example, if the object is a car, c2 will be 1 and c1 & c3 will be 0, and so on. To train our model, we will use both forward and backward propagation.

<table>
<tr><td rowspan="8">y =</td><td>pc</td></tr>
<tr><td>bx</td></tr>
<tr><td>by</td></tr>
<tr><td>bh</td></tr>
<tr><td>bw</td></tr>
<tr><td>c1</td></tr>
<tr><td>c2</td></tr>
<tr><td>c3</td></tr>
</table>

Figure 3.2: Eight Dimensional Vector

3.1.2 YOLO v1

The YOLO v1 object detection model is a single-stage model.Object detection is described as a regression problem with spatially separated bounding boxes and class probabilities. A single neural network predicts bounding boxes and class probabilities from full images in a single assessment. Because the entire detection pipeline is a single network, detection performance can be adjusted directly from start to finish.

Limitations

- YOLO v1 has difficulties in detecting small objects that appear in groups.
- Detecting objects with different aspect ratios is tough for YOLO v1.

- When compared to Fast R-CNN, YOLO v1 commits more localization errors.

3.1.3 YOLO v2

The primary changes in this version are that it is better, faster, and more advanced in order to meet the faster R-CNN, which is an object identification technique that uses a Region Proposal Network to recognise items from image input and SSD (Single Shot Multibox Detector).

Improvements

- Batch Normalization: It scales and slightly alters the activations to equalise the input layer. mAP increased by 2%.
- Higher Resolution Classifier: Input changed from 224*224 to 448*448 mAP increased by 4%.
- Anchor Boxes: are designed to detect objects in the same grid.
- Fine Grained Features: Divides the image into 13*13 grid cells which helps identifying small objects, unlike V1.
- Multi Scale Training: Model is trained on different sizes of objects for the same images.
- Darknet - 19: For categorization objects, YOLO v2 uses the Darknet 19 architecture, which has 19 convolutional layers, 5 max-pooling layers, and a softmax layer. Darknet is a C-based CUDA-based neural network framework. It detects objects in a fraction of a second, which is essential

for real-time prediction.

- Results: At 67 frames per second, YOLO v2 can achieve a mAP of 76.8, while at 40 frames per second, the detector achieves a 78.6 mAP accuracy, exceeding state-of-the-art models such as the Quicker R-CNN and SSD while running at a far faster rate.

3.1.4 YOLO v3

The previous version, now called YOLO v3, has been enhanced incrementally. Because many object detection algorithms have been around for a long, the rivalry is all about how accurately and quickly items are recognised. YOLO v3 has all we need for accurate object recognition and categorization in real time.

Improvements

- Bounding Box Predictions: Logistic Regression is used to predict the objectiveness score.
- Class Predictions: Instead of using softmax, Logistic classifiers are used, allowing for multi-label classification.
- Feature Pyramid Networks.
- Darknet-53 Architecture: has 53 convolutional layers.

3.1.5 YOLO v4

The CSPDarknet53 backbone, spatial pyramid pooling extra module, PANet path-aggregation neck, and YOLO v3 head make up the architecture of YOLO v4. CNN can learn more effectively with the help of CSPDarknet53, a new backbone. The spatial pyramid pooling block is used across CSPDarknet53 to increase the receptive field and separate the most essential context features. Instead of the FPN utilized in YOLO v3, the PANet is used for parameter aggregation for different detector levels.

Improvements

- With equivalent performance, YOLO v4 is twice as quick as EfficientDet.
- With an AP value of 43.5 percent on the COCO dataset and a real-time speed of 65 frames per second on the Tesla V100, YOLO v4 is based on the Darknet as well, outperforming the fastest and most accurate detectors in terms of both speed and accuracy.
- In addition, AP (Average Precision) and FPS (Frames Per Second) improved by 10% and 12% compared to YOLO v3.

3.1.6 YOLO v5

As a result, YOLO v5 is touted to be much faster and lighter than YOLO v4, with accuracy comparable to the YOLO v4 test.

The Pytorch framework is used to create YOLO v5. Pytorch inferences are so rapid that many other AI practitioners often transfer the YOLO v3 and YOLO v4 weights into ultralytics Pytorch weights before releasing YOLO v5.

Improvements

- Unlike previous versions, YOLO v5 is a PyTorch implementation rather than a fork of the original Darknet.
- Like the YOLO v4, the YOLO v5 has a CSP backbone and a PANet neck.
- Two of the most notable improvements are mosaic data augmentation and auto-learning bounding box anchors.

3.2 Why YOLO v5

The YOLO v5 version is roughly 90% smaller than the YOLO v4 version. As a result, YOLO v5 is touted to be much faster and lighter than YOLO v4, with accuracy comparable to the YOLO v4 test. As a result, we chose YOLO v5.

3.3 Dataset

Training, Validation and Testing of proposed model YOLO v5 are done on a custom prepared dataset combined with MS COCO 2017 Dataset [17]. MS COCO 2017 dataset contains 80 different object classes likely, person, dog, chair, potted plant, etc. In addition, we added 15 more different object classes such as switchboard, pillow, locker, keys, open door, closed door, window, direction board, postbox, pole, shop, manhole, tree, upstairs, downstairs. Which are not mentioned in MS COCO 2017 Dataset (95 classes overall). These objects are relevant to Indian atmosphere. For each object class, we added 30 - 50 images, all together we added 500 images to dataset. By overall images we considered for doing image detection is 5000.

3.3.1 Annotation tool

We used makesense.ai [30] a data annotation tool to annotate our new dataset. Makesense provides a lot more flexibility than other tools in adding labels list, most of the other tools automatically order the labels alphabetically. But, makesense follows the order we provide, and it is also possible to download the annotated images in YOLO format. So, this is the reason why we choose makesense.ai as our annotation tool.

3.3.2 YOLO format

To train & validate on YOLO algorithm, we need a specific format of dataset, as shown in figure 3.3. In the images folder, we further need to divide it into 3 different folders namely, train, val, test, and save respective images in those folders. Similarly for labels folder, here all the labels will be text files. And finally, we need to specify the paths of all the images in respective folders, in their respective text files (train.txt, val.txt, test.txt).

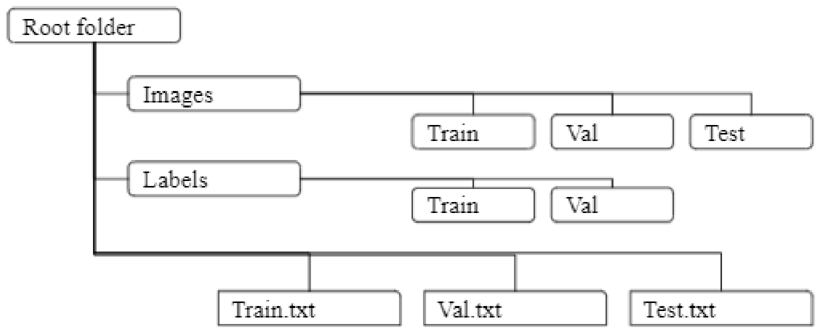

Figure 3.3: Dataset folder structure

3.4 Algorithm

3.4.1 YOLO v5

The object identification method YOLO, which stands for "You Only Look Once," focuses on detecting objects in photos and separates them into a grid structure. Each grid cell is in charge of detecting items within its boundaries. At the moment, YOLO v5 is one of the best object detection models available. The beautiful thing about this Deep Neural Network is that retraining it on our custom dataset is quite simple.

Architecture

YOLO v5's network design. It is organised into three sections: the backbone (CSPDarknet), the neck (PANet), and the head (YOLO Layer). The data is fed into two programmes: CSPDarknet, which extracts features, and PANet, which fuses them. Finally, the detection results are output by YOLO Layer (class, score, location, size).

The backbone of the Object Detector will be used to pre-train it, and the head will be used to predict classes and bounding boxes. Backbones can run on either GPU or CPU platforms. For the Sparse prediction object detector, the Head can be one-stage (e.g., YOLO, SSD, RetinaNet) or two-stage (e.g., Faster R-CNN) for Dense prediction. Object detectors have a layer called the Neck that collects feature maps and is located between the backbone and the head. The architecture of EfficientDet is shown in figure 3.4 [32].

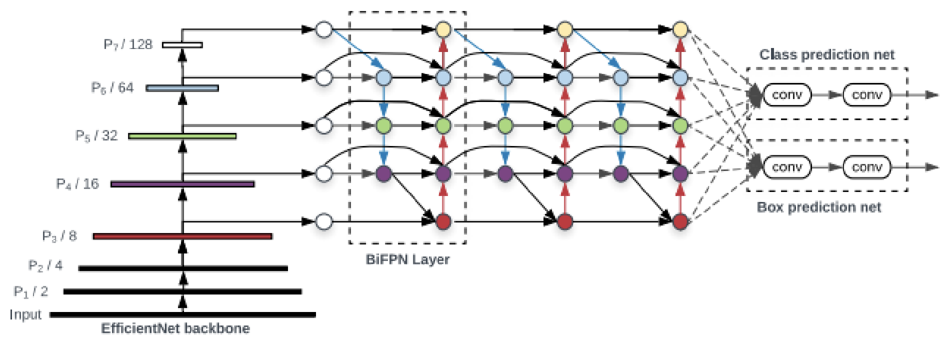

Figure 3.4: EfficientDet architecture

3.5 Text to speech conversion

Text to Speech conversion libraries uses text as input and gives a speech as output.

Some of the Text to speech conversion libraries:

- Google Text to Speech (gTTS).
- Text to speech conversion library in python (pyttsx3).

3.5.1 Text to speech synthesizer

Text to speech system (TTS) transforms text into a voice using a speech synthesizer. It artificially produces a human voice. A speech synthesiser is a computer system that is used for this purpose. Text processing and speech generation are two main elements of a text-to-speech system. The process of Text to speech synthesis is shown in figure 3.5 [18].

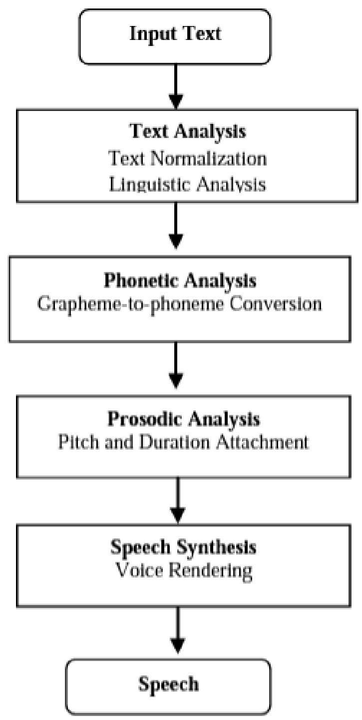

Figure 3.5: Text to speech synthesis - block diagram

Text Processing

The text processing component aims to process the given input text and generate a phonemic unit sequence that is appropriate. The incoming text is first processed, normalized, and transcribed into a phonetic or other linguistic representation in a text-to-speech system. Low-level processing difficulties like sentence segmentation and word segmentation are dealt with by text processing components.

Three Phases:

- Document Structure detection: The document structure can be detected by diagnosing punctuation marks and paragraph formatting.
- Text Normalization: The text normalization controls abbreviations and acronyms. The goal of normalization is to make the text correspond, for example, Dr could be represented as the doctor. Valid normalization constructs a fair result.
- Linguistic Analysis: Linguistic analysis contains a morphological analysis for syntactic analysis and accurate word pronunciation to promote accenting and phrasing to manage obscurities in written text.

Speech Generation

The speech generation segment procedure develops the speech by utilizing parameters such as:

- Phonetic Analysis: It concentrates on the phonetic level of each word. Each phone is labeled with details about what sound to construct and how to construct it, as well as style and emphasis.
  - Grapheme to phoneme conversion: Each word in the input sentences has its accurate diction established.

  - Homograph disambiguation: Figuring out whether the input sentence uses the past or present tense interpretation of the word. The dictionary is used to determine a word tense system.

- Prosodic Analysis: Prosodic analysis is crucial because it lays the groundwork for phonological prosodic processing, which involves marking prosodic effects surrounding our utterance plans, and phonetic prosodic processing, which involves determining appropriate rendering approaches for the marked prosody.

  - Create a symbolic phonological role for an abstract explanatory system that depicts observations of the behavior of the parameters of prosody within the auditory signal (fundamental frequency movement, intensity modifications, and period movement).

  - Create a phonological system that may be used as input to a processing system, resulting in an acoustic signal that has valid prosody when jugged by listeners.

3.5.2 pyttsx3

Pyttsx3 is a Python text-to-speech library. Unlike other libraries, it works both offline and online, and is compatible with both Python 2 and 3 versions. It works without any delay. There are some customization's available. we can change the voice of the engine. We can also change the speed of the voice engine. It supports several languages that is unicode. By default, the best driver for your platform is used.

Platform Specific Drivers

- sapi5 - SAPI5 on Windows
- nsss - NSSpeechsynthesizer on Mac OS x
- espeak- eSpeak on every other platform

3.5.3 gTTS

gTTS is a programme that turns text into audio files that may be saved as mp3 files. The gTTS API supports English, Hindi, Tamil, French, German, and a variety of additional languages. It includes a speech-specific sentence tokenizer that enables for endless amounts of text to be read while keeping accurate intonation, abbreviations, decimals, and more, as well as customisable text pre-processors that can improve pronunciation, among other things.

The application, tool, or software takes a user's input text and deduces the linguistics of the language and performs logical inference on it using natural language processing methods. The next block receives the processed text and performs digital signal processing on it. After that, a variety of algorithms and transformations are used to convert the processed text into a voice format. Throughout the procedure, speech is synthesised.

Chapter 4

Experimental Results

We carried out our training, validation, and testing on the google colab platform. Weights & Biases [8] is used to track the training and validation process for visualization.

4.1 Training

Tesla K80 with 12 GB RAM, powered by google colab is used for training the YOLO v5 model, with the help of PyTorch and PyTorch-Cuda libraries, coded in python. The model is trained on the dataset mentioned for 50 epochs, With a batch size of 8. Here are The class loss (shown in figure 4.1), Box loss (shown in figure 4.2), Object loss (shown in figure 4.3) results for the training set.

Figure 4.1: Training: Class loss vs number of epochs

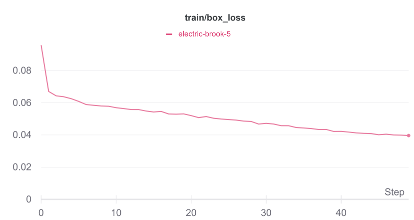

Figure 4.2: Training: Box loss vs number of epochs

Figure 4.3: Training: Object loss vs number of epochs

4.2 Validaton

Validation is done on each training epoch with a batch size of 16, for 50 epochs after each training epoch. Here are the class loss (shown in figure 4.4), Box loss (shown in figure 4.5), Object loss (shown in figure 4.6) results for the validation set.

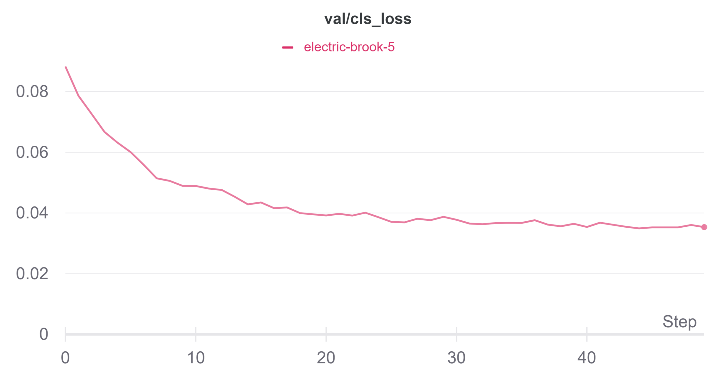

Figure 4.4: Validation: Class loss vs number of epochs

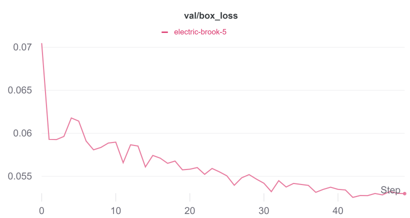

Figure 4.5: Validation: Box loss vs number of epochs

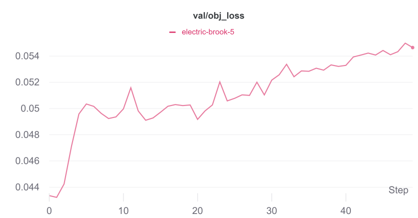

Figure 4.6: Validation: Object loss vs number of epochs

4.3 Evaluation metrics

Model is evaluated based on Precision, Recall, MAP (mean Average Precision).

4.3.1 Precision

Precision is one measure of a machine learning model's performance – the accuracy of a model's positive prediction. The number of true positives divided by the total number of positive predictions is known as precision. The precision achieved by our model is shown in figure 4.7.

4.3.2 Recall

A recall is a metric that measures how many right positive predictions were made out of all possible positive predictions. Positive predictions that were missed are indicated by the recall. The recall achieved by our model is shown in figure 4.8.

4.3.3 mean Average Precision (mAP)

Depending on the different detection problems that exist, the mean Average Precision or mAP score is calculated by taking the mean AP over all classes and/or overall IoU thresholds. The mAP of our model is shown in figure 4.9 & figure 4.10.

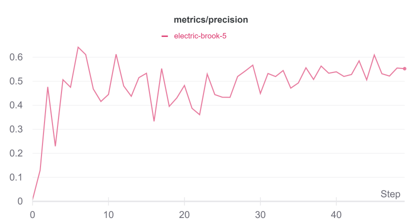

Figure 4.7: Evaluation metric: Precision vs number of epochs

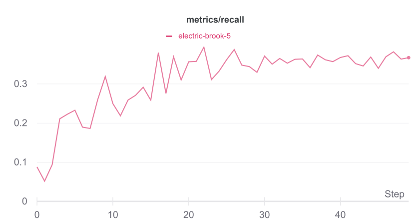

Figure 4.8: Evaluation metric: Recall vs number of epochs

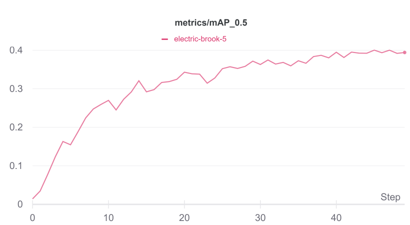

Figure 4.9: Evaluation metric: mAP_0.5 vs number of epochs

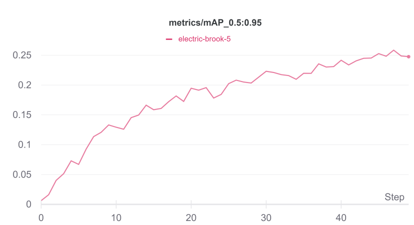

Figure 4.10: Evaluation metric: mAP_0.5:0.95 vs number of epochs

4.4 Output tensor to speech conversion

The output of YOLOv5 is a tensor of objects. Each object in the tensor contains six values. i.e., x, y, w, h, confidence, label. Here, (x, y) is the center of the detected box, and w, h are the width and height of the box, whereas confidence reflects how likely the box contains an object and how accurate is the bounding box, and finally label is the object that is detected.

To convert the output tensor to speech, we divided it into two parts, one is detected objects to text, and the other is text to speech.

4.4.1 Output tensor to text

A function is defined to generate text from output tensor, for further speech generation. The function takes the following parameters:

- results - output tensor of YOLOv5.
- H - Height of the window (Image).
- W - Width of the window (Image).
- names - The list of labels, with which the model is trained.

The function iterates over the results. In each iteration, it creates a text describing the position and object and adds to a list of text. The window (Image) is divided into nine parts (three parts - horizontally, three parts vertically, overall it makes nine). Each bounding box contains a center point, with which we find at which place the object lies in the view. Finally, all the text in the list is joined with a comma-separated delimiter, which is then returned.

4.4.2 Text to speech

To generate the speech using the text produced above, we can make use of either pyttsx3, or gTTS.

Using pyttsx3

To generate speech using pyttsx3, we first need to import pyttsx3 and initialize it. Then, pass the text generated above, to a method called ”say” to let the application speak the text given. A method, ”runAndWait” can be used to wait for the above process to complete before it moves on to the next frame.

Using gTTS

To generate speech using gTTS, we need to import the following packages: subprocess, gTTS, AudioSegment. Now, we can pass the text to the gTTS initializer and save the file returned by gTTS. The saved audio file is then transformed using AudioSegment, which is then played using a subprocess call.

Chapter 5

Conclusion

We are able to achieve precision as 0.55, recall as 0.37, and mAP as 0.4, with the proposed model. our model YOLO v5 is able to detect 95 different objects, with high confidence. with this model now we are able to detect objects those are most required for the visually impaired in their daily life.

From the two python libraries for speech generation, we observed that pyttsx3 doesn't require any internet connection, whereas, on the other side, gTTS need constant internet connectivity. gTTS sends text to Google's servers to generate a speech file, which is then returned. The speech generated by pyttsx3 is spoken comparatively faster than gTTS. The speech generated by both the libraries are 100% accurate.

Hence, we found pyttsx3 more helpful than gTTS, considering the time taken to produce audio, the delay in frames, the libraries required, and the network connectivity. Since, we want to develop a device that should not be affected by any external factors like bad network, etc. So, we used pyttsx3 as our main library.

Bibliography

[1] Mouna Afif, Riadh Ayachi, Edwige Pissaloux, Yahia Said, and Mohamed Atri. Indoor objects detection and recognition for an ict mobility assistance of visually impaired people. *Multimedia Tools and Applications*, 79(41):31645–31662, 2020.

[2] Mouna Afif, Riadh Ayachi, Yahia Said, Edwige Pissaloux, and Mohamed Atri. An evaluation of retinanet on indoor object detection for blind and visually impaired persons assistance navigation. *Neural Processing Letters*, pages 1–15, 2020.

[3] A Annapoorani, Nerosha Senthil Kumar, and V Vidhya. Blind-sight: Object detection with voice feedback. 2021.

[4] Maria Anu, S Divya, et al. Building a voice based image caption generator with deep learning. In *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pages 943–948. IEEE, 2021.

[5] A Balachandar, E Santhosh, A Suriyakrishnan, N Vigensh, S Usharani, and P Manju Bala. Deep learning technique based visually impaired

people using yolo v3 framework mechanism. In *2021 3rd International Conference on Signal Processing and Communication (ICPSC)*, pages 134–138. IEEE, 2021.

[6] Fereshteh S Bashiri, Eric LaRose, Jonathan C Badger, Roshan M D'Souza, Zeyun Yu, and Peggy Peissig. Object detection to assist visually impaired people: A deep neural network adventure. In *International Symposium on Visual Computing*, pages 500–510. Springer, 2018.

[7] Swapnil Bhole and Aniket Dhok. Deep learning based object detection and recognition framework for the visually-impaired. In *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, pages 725–728. IEEE, 2020.

[8] Lukas Biewald. Experiment tracking with weights and biases, 2020. Software available from wandb.com.

[9] Medical Eye Center. Importance of Eye Care, available at, https://www.medicaleyecenter.com/2016/06/20/importance-eye-care/, 2016. [Online; accessed 6-November-2021].

[10] Xiaobai Chen, Jinglong Xu, and Zhiyi Yu. A 68-mw 2.2 tops/w low bit width and multiplierless dcnn object detection processor for visually impaired people. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(11):3444–3453, 2018.

[11] Wafa M Elmannai and Khaled M Elleithy. A highly accurate and reliable data fusion framework for guiding the visually impaired. *IEEE Access*, 6:33029–33054, 2018.

[12] Wei Fang, Lin Wang, and Peiming Ren. Tinier-yolo: A real-time object detection method for constrained environments. *IEEE Access*, 8:1935–1944, 2019.

[13] Sejal Gianani, Abhishek Mehta, Twinkle Motwani, and Rohan Shende. Juvo-an aid for the visually impaired. In *2018 International Conference on Smart City and Emerging Technology (ICSCET)*, pages 1–4. IEEE, 2018.

[14] Piyush Jhinkwan, Vaishali Ingale, and Shubham Chaturvedi. Object detection using convolution neural networks. In *Proceedings of International Conference on Communication and Information Processing (ICCIP)*, 2019.

[15] Rashika Joshi, Meenakshi Tripathi, Amit Kumar, and Manoj Singh Gaur. Object recognition and classification system for visually impaired. In *2020 International Conference on Communication and Signal Processing (ICCSP)*, pages 1568–1572. IEEE, 2020.

[16] Yongjun Li, Shasha Li, Haohao Du, Lijia Chen, Dongming Zhang, and Yao Li. Yolo-acn: Focusing on small target and occluded object detection. *IEEE Access*, 8:227288–227303, 2020.

[17] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[18] Suhas R Mache, Manasi R Baheti, and C Namrata Mahender. Review on

text-to-speech synthesizer. *International Journal of Advanced Research in Computer and Communication Engineering*, 4(8):54–59, 2015.

[19] Mansi Mahendru and Sanjay Kumar Dubey. Real time object detection with audio feedback using yolo vs. yolo_v3. In *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pages 734–740. IEEE, 2021.

[20] Jawaid Nasreen, Warsi Arif, Asad Ali Shaikh, Yahya Muhammad, and Monaisha Abdullah. Object detection and narrator for visually impaired people. In *2019 IEEE 6th International Conference on Engineering Technologies and Applied Sciences (ICETAS)*, pages 1–4. IEEE, 2019.

[21] A Nishajith, J Nivedha, Shilpa S Nair, and J Mohammed Shaffi. Smart cap-wearable visual guidance system for blind. In *2018 International Conference on Inventive Research in Computing Applications (ICIRCA)*, pages 275–278. IEEE, 2018.

[22] Arjun Pardasani, Prithviraj N Indi, Sashwata Banerjee, Aditya Kamal, and Vaibhav Garg. Smart assistive navigation devices for visually impaired people. In *2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS)*, pages 725–729. IEEE, 2019.

[23] Sandeep Pasupuleti, Lahari Dadi, Manikumar Gadi, and R Krishnaveni. Image recognition and voice translation for visually impaired. *International Journal of Research in Engineering, Science and Management*, 4(5):18–23, 2021.

[24] Charmi T Patel, Vaidehi J Mistry, Laxmi S Desai, and Yogesh K Meghrajani. Multisensor-based object detection in indoor environment for visually impaired people. In *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, pages 1–4. IEEE, 2018.

[25] Kanchan Patil, Avinash Kharat, Pratik Chaudhary, Shrikant Bidgar, and Rushikesh Gavhane. Guidance system for visually impaired people. In *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, pages 988–993. IEEE, 2021.

[26] Ferdousi Rahman, Israt Jahan Ritun, Nafisa Farhin, and Jia Uddin. An assistive model for visually impaired people using yolo and mtcnn. In *Proceedings of the 3rd International Conference on Cryptography, Security and Privacy*, pages 225–230, 2019.

[27] Md Atikur Rahman and Muhammad Sheikh Sadi. Iot enabled automated object recognition for the visually impaired. *Computer Methods and Programs in Biomedicine Update*, page 100015, 2021.

[28] Roshan Rajwani, Dinesh Purswani, Paresh Kalinani, Deesha Ramchandani, and Indu Dokare. Proposed system on object detection for visually impaired people. *International Journal of Information Technology (IJIT)*, 4(1):1–6, 2018.

[29] Samkit Shah, Jayraj Bandariya, Garima Jain, Mayur Ghevariya, and Sarosh Dastoor. Cnn based auto-assistance system as a boon for directing visually impaired person. In *2019 3rd International Conference on*

Trends in Electronics and Informatics (ICOEI), pages 235–240. IEEE, 2019.

[30] Piotr Skalski. Make Sense available at, https://www.makesense.ai/, 2019. [Online; accessed 6-November-2021].

[31] Minghui Sun, Pengcheng Ding, Jiageng Song, Miao Song, and Limin Wang. "watch your step": Precise obstacle detection and navigation for mobile users through their mobile service. *IEEE Access*, 7:66731–66738, 2019.

[32] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10781–10790, 2020.

[33] Selman Tosun and Enis Karaarslan. Real-time object detection application for visually impaired people: Third eye. In *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*, pages 1–6. Ieee, 2018.

[34] Yan Chiew Wong, JA Lai, SSS Ranjit, AR Syafeeza, and NA Hamid. Convolutional neural network for object detection system for blind people. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, 11(2):1–6, 2019.

[35] Qiwei Xu, Runzi Lin, Han Yue, Hong Huang, Yun Yang, and Zhigang Yao. Research on small target detection in driving scenarios based on improved yolo network. *IEEE Access*, 8:27574–27583, 2020.

[36] Cang Ye and Xiangfei Qian. 3-d object recognition of a robotic navigation aid for the visually impaired. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(2):441–450, 2017.

[37] Ervin Yohannes, Paul Lin, Chih-Yang Lin, and Timothy K Shih. Robot eye: Automatic object detection and recognition using deep attention network to assist blind people. In *2020 International Conference on Pervasive Artificial Intelligence (ICPAI)*, pages 152–157. IEEE, 2020.