Received Date: 2nd November, 2022
Accepted Date: 2nd April, 2023

Object and Text Detection

Pooja Singh[1*], Richa Pokhrel[2], Asmita Jha[3], Pragya Jha[4]& Saroj Shakya[5]

[1]Department of Electronics Engineering, Thapathali Campus, E-mail : pooja.732419@thc.tu.edu.np
[2] Department of Electronics Engineering, Thapathali Campus, E-mail: richa.732419@thc.tu.edu.np
[3]Department of Electronics Engineering, Thapathali Campus, E-mail: asmita.732419@thc.tu.edu.np
[4]Department of Electronics Engineering, Thapathali Campus, E-mail: pragya.732419@thc.tu.edu.np
[5]Department of Electronics Engineering, Thapathali Campus, E-mail: sarojsh@tcioe.edu.np

***Abstract*— The main aim of our project is to develop a portable raspberry pi implemented gadget for object detection with relative motion and distance. This technology is basically used for conversion of sequence of real time objects into series of text whichcan be further stored into database and can be utilized to assist visually impaired people and in various security purposes as well. For that purpose, the conversion system is proposed in this project.Our system basically operates in 2 different modes. One is detecting the class of objects nearby with the help of R-CNN network, and the second one is obstacle detection using ultrasonic sensor. It includes 3 buttons for mode selection and the system operates on the basis of mode selection. It includes camera to capture an image as input, and input image is then passed to the R-CNN that recognizes number of objects inside image, their classes and types, text written inside and which is then can be passed to the database for a storage.**

***Keywords* — Region-Based Convolutional Neural Network (R-CNN), Region of Interest (ROI), Region Proposal Network (RPN), Google Text-to-Speech(gTTS)**

I. INTRODUCTION

According to WHO Globally, at least 2.2 billion people have a vision impairment or blindness, of whom at least 1 billion have a vision impairment that could have been prevented or has yet to be addressed. Object detection can give a great help to visually impaired [1] people to count objects in a scene and determine and track their precise locations, all while accurately labeling them. It initially started with digital image processing methods such as Edge detection, Recognition by parts, and Gradient matching.

With the recent advancements and introduction of deep neural networks object recognition has been more accurate and can be applied in real-time with faster implementation. Generally, the detection and classification are done as two different steps. RPN is used for detection and R-CNN is used for classification.The RPN networks which provide us with the ROI (Regions of Interest) are to be done as an external process and these regions are shared with Faster R-CNN. Some of these networks include Selective search, greedy merges, etc. Nevertheless, the region proposal step still consumes almost the same amount of time as the detection network. The two processes can be expensive on their own, but a cost-effective way is to share the convolutions between them. In this project, we merge RPN and R-CNN parts into a unified process which makes the process much easier. This can be an effective solution for better accuracy and detection time tradeoffs. RPN networks are constructed by adding convolutional layers on top of the convolutional feature maps that are used by detectors like faster R-CNN. By doing so, we create a fully connected network, which helps us generate region proposals for detection. This can be trained similar to the detection algorithms. As we consider multiple class image, we need the region proposals to be distinctive.

II. ARCHITECTURE

The device consists of a Raspberry [2] Pi 3B+, speaker or earphone, Raspberry pi camera, power supply (230V AC) and power adapter. The power adapter converts the 230V AC power supply to 5V DC/2.5A to power the Raspberry Pi. The camera must manually be pointed towards the text and a picture is captured. This picture is then processed by the Raspberry Pi and the audio output is heard through the speaker.

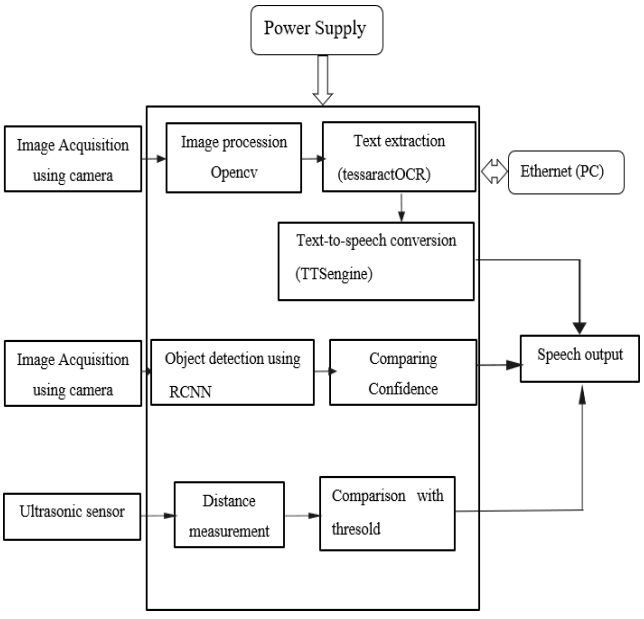

Figure 1: Architecture of the project

* Corresponding Author

A. Power Supply

Raspberry Pi is a very cheap computer that runs Linux, but it also provides a set of GPIO (general purpose input/ output) pins that allow you to control electronic components for physical computing and explore the Internet of Things (IoT). In our project, we deal with images processing and object recognition using DNN. This requires very high processing power as well as a lot of external storage space. In this regard, raspberry pi 3 b+ is the most suitable as its storage capacity can be enhanced easily using an SD card which is not possible in many other controllers like Arduino. Likewise, as compared to its previous versions, pi 3 b+ can be interfaced with a number of devices very conveniently as it contains multiple different ports and interfaces.

B. Image Acquisition using Camera

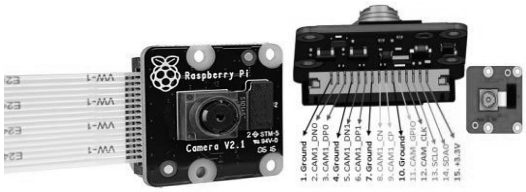

Figure 2: Camera module with the pin-out description

Pi Camera is used to take image of page that person wants to observe and read. The system operates in two different modes.In first mode to read texts from image, image is pre-processed using OpenCV [3]. The pre-processing stage consists of seven steps: Resizing, Gray scaling, Gaussian Blurring, Edge Detection, Perspective Transformation, Noise Removal andAdaptive Thresholding. The captured image is resized and gray scaled. For the removal of noise Gaussian Blurring is done. Now, four points of the paper to be processed is determined through canny edge detection and perspective transformation isapplied. There are possibilities of the image getting skewed with either left or right orientation. So, canny edge detection checks for an angle of orientation and applies perspective transformation till the lines match with the true horizontal axis,which produces a skew corrected image. The noise introduced during capturing or processing of image is cleared by applyingmorphological transformations. Finally, the image is binarizedusing adaptive thresholding. The next step is post-processing. Itinvolves segmentation of enhanced image and recognition of characters. The ASCII values of the recognized characters are processed by Raspberry Pi board using Tesseract. Here each ofthe characters is matched with its corresponding template and saved as normalized text. The recognized text is then converted into speech [4] through a headset using TTS Engine.

For the Object Detection, after capturing images the object is passed to the R-CNN layer. At first bounding boxes will be generated on all the objects detected in the image through anchor boxes, and each object is passed to the network for feature extraction. Each object will be labeled with the similar classesavailable in datasets. The confidence score is generated by using the Softmax function [5]. If the confidence threshold is less than 0.5 then the object will be classified as a background, if the confidence threshold is more than 0.5 then the output will be the object with the detected class label. The Softmax Activationfunction can be mathematically expressed as:

$$\sigma(z)_i = \frac{e^{zi}}{\sum_{j=1}^{K} e^{zj}} \quad (1)$$

C. Ultrasonic Sensor

Figure 3: Ultrasonic sensor

Ultrasonic sound waves are generated by Ultrasonic sensors [6] which are reflected through obstacles. These reflected signals are then received by the module. Calculating the time delay between the transmitted and received signal, a distance of the obstacle [7] is known from the formula.

$$s = \frac{vt}{2}; \text{ where } v = velocity\ of\ sound = 334m/s \quad (2)$$

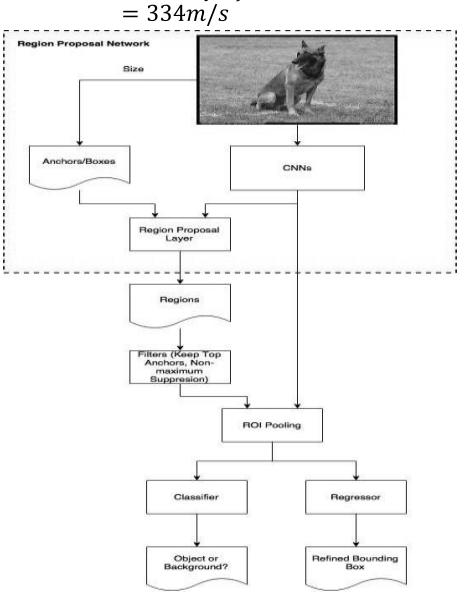

Figure 4: R-CNN working diagram

This distance is compared with threshold distance and warning message is sent to visually impaired people if distance is less than threshold. According to the distance obtained, he or she will be suggested to move in the direction with less or no obstacles. Headset is connected to 3.5 mm audio jack providedby Raspberry pi to give voice command to visually impaired people.

D. R-CNN

The architecture consists of the RPN as a region proposal algorithm and the ROI pooling layer. Before this let us know about CNN.

1) CNN

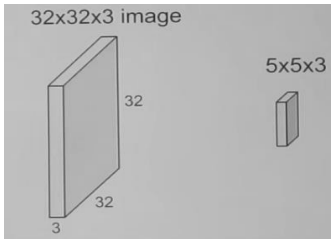

Figure 5: CNN example

Unlike neural networks, the input is a vector with a multi-channeled image (3 channeled- RGB in this case).

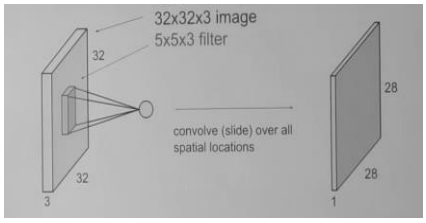

Figure 6: Convolution layer

Convolution [8] is the first layer to extract features from an input image. It preserves the relationship between pixels by learning image features using small squares of input data. It is a mathematical operation that takes two inputs such as image matrix and a filter or kernel.

From the above image we can observe that for our input of 32*32*3 we took a filter of 5*5*3 and slid it over the complete image and along the way take the dot product between the filterand chunks of the input image. The output results with an imageof size 28*28*1.

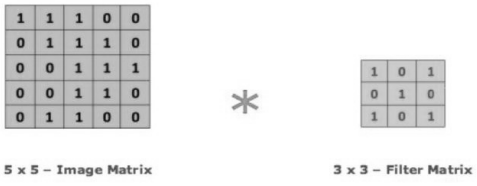

Figure 7: CNN example

E.g.: Consider a 5 x 5 whose image pixel values are 0, 1 and filter matrix 3 x 3 as shown in below.

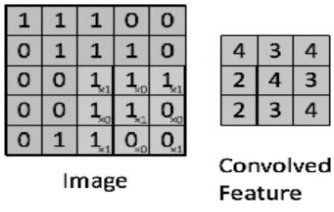

Figure 8: Convolution example

Then the convolution of 5 x 5 image matrix multiplies with 3 x3 filter matrix which is called “Feature Map” as output shown in below with a stride of 1.

i. Stride

Stride is the number of pixels shifts over the input matrix. When the stride is 1 then we move the filters to 1 pixel at a time. Whenthe stride is 2 then we move the filters to 2 pixels at a time andso on.

ii. Padding

If we increase the stride value the size of image keeps onreducing, padding with zeros across it solves this problem.

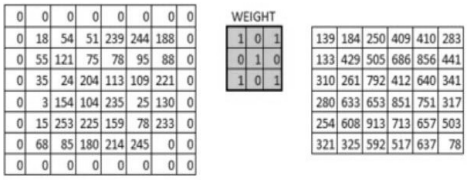

Figure 9: Image padding

From the below image we can observe the size of image isretained after padding zeros with stride 1

iii. Pooling

Sometimes when the images are too large, we would need to reduce the number of trainable parameters. It is then desired toperiodically introduce pooling layers between subsequent convolution layers. Pooling is done for the sole purpose of reducing the spatial size of the image. Pooling is done independently on each depth dimensions, therefore the depth ofthe image remains unchanged. The most common form of pooling layer generally applied is the max pooling.

2) Region Proposal Network (RPN)

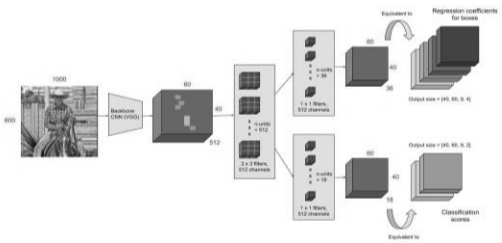

Figure 10: RPN architecture

The region proposal network (RPN) starts with the input image being fed into the backbone convolutional neural network. Theinput image is first resized such that its shortest side is 600px with the longer side not exceeding 1000px.

The output features of the backbone network (indicated by H xW) are usually much smaller than the input image depending on the stride of the backbone network. For both the possible backbone networks used in the paper (VGG, ZF-Net) the network stride is 16. This means that two consecutive pixels inthe backbone output features correspond to two points 16 pixelsapart in the input image.

For every point in the output feature map, the network has to learn whether an object is present in the input image at its corresponding location and estimate its size. This is done by placing a set of "Anchors" on the input image for each location on the output feature map from the backbone network. These anchors indicate possible objects in various sizes and aspect ratios at this location. The figure below shows 9 possible anchors in 3 different aspect ratios and 3 different sizes placed on the input image for a point A on the output feature map. For the PASCAL challenge, the anchors used have 3 scales of box area 128^2, 256^2, 512^2 and 3 aspect ratios of 1:1, 1:2 and 2:1.

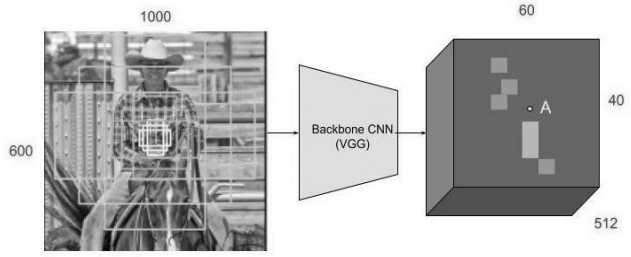

Figure 11: RPN feeding to VGG backbone

As the network moves through each pixel in the output feature map, it has to check whether these k corresponding anchors spanning the input image actually contain objects, and refine these anchors' coordinates to give bounding boxes as "Object proposals" or regions of interest.

First, a 3x3 convolution with 512 units is applied to the backbone feature map as shown in Figure 1, to give a 512-d feature map for every location. This is followed by two sibling layers: a 1x1 convolution layer with 18 units for object classification, and a 1x1 convolution with 36 units for boundingbox regression.

The 18 units in the classification branch give an output of size (H, W, 18). This output is used to give probabilities of whether or not each point in the backbone feature map (size: H x W) contains an object within all 9 of the anchors at that point.

The 36 units in the regression branch give an output of size (H, W, 36). This output is used to give the 4 regression coefficients of each of the 9 anchors for every point in the backbone feature map (size: H x W). These regression coefficients are used to improve the coordinates of the anchors that contain objects.

3) Region Of Interest (ROI) pooling:

Region of interest pooling (also known as ROI pooling) [9] is an operation widely used in object detection tasks using convolutional neural networks. For example, to detect multiple cars and pedestrians in a single image. Its purpose is to perform max pooling on inputs of nonuniform sizes to obtain fixed-size feature maps (e.g., 7×7).

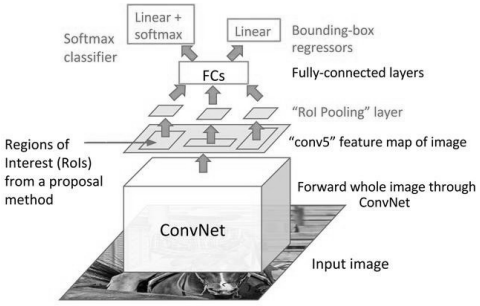

Figure 12: ROI pooling

After RPN, we get proposed regions with different sizes. Different sized regions mean different sized CNN feature maps. It's not easy to make an efficient structure to work on features with different sizes. ROI Pooling can simplify the problem by reducing the feature maps into the same size. The result is that from a list of rectangles with different sizes we can quickly get a list of corresponding feature maps with a fixed size. One of the benefits of ROI pooling is processing speed. If there are multiple object proposals on the frame (and usually there'll be a lot of them), we can still use the same input feature map for all of them. Since computing the convolutions at early stages of processing is very expensive, this approach can save us a lot of time.

IV. RESULT AND ANALYSIS

The system helps in capturing images obtained from the external environment. These objects are detected along with the image and then converted into text sequences. To detect the objects, the objects to be detected for each image are taken as ten with the flexibility of increasing to a hundred as the highest value. Each one is provided with its own confidence score by the utilization of the Softmax function. Along with this, the maximum value of threshold confidence is set as 0.5 so that the objects with a confidence score <50 per cent won't be detected. Conversion of small and capital letters is observed in the gTTS file. This work doesn't need an internet connection due to the use of text-to-speech conversion techniques. Further, it's found easier to use by the blind ones.

A. Object Detection Output

While detecting the person and bear, the bear is seen to have higher accuracy (97%) than the person (75%). Similarly, in the detection of person, cell phone and book, the person has greater accuracy (86.0%) than cell phone (71.0%) and book (59.0%).

B. Graphical Results

When the results obtained from the test of datasets are plotted with respect to the number of epochs, graphs of different natures are found. The loss graphs are seen to be decreasing exponentially with respect to the number of epochs. Likewise, the graph of elapsed time shows the highest value of 10 for 22 epochs and the lowest value of 2

for 20 epochs. Also, nearly maximum values are seen even for 20 epochs, 60 epochs and 120 epochs. A constant value of 8 is detected in the range of 60-

Figure 13: Output sample for object detection

120. Moreover, the mean overlapping boxes increase immediately from 0.5 to 2.87 at 0 epoch and then show a non- linear increment up to 60 epochs and a bit increment after that.Further, the class accuracy decreases from 0.88 to 0.74 up to 20epochs and increases non-linearly after that.

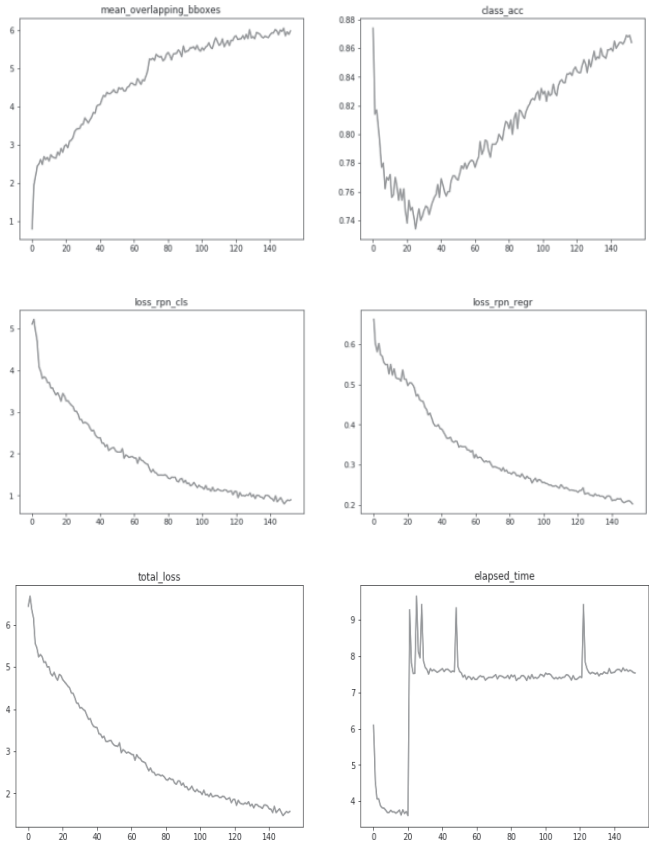

Figure 14:Plot of different parameters with respect to the epoch

i. Text recognition

Above table is the observation table for Times New Roman fonttaken 10 times. From above table, it can be concluded that for A4 sized paper accuracy lies in between 90 to 98 % for distanceranging from 43 to 45 cm. Also, accuracy was found above 95% in optimum distance for text size above 14 pt. Hence, optimum distance for A4 sized paper is [43, 45] cm and optimum text-size for A4 sized paper is above 14 pt*v*

Table 1:Observation table for text recognition

Distance (cm)	Correctly Detected Classes					Accuracy (1024)
	24pt (11)	16pt (547)	14pt (182)	12pt Bold (202)	12pt Normal (82)	
35	11 100%	541 98.90 %	181 99.45 %	198 98.01 %	77 93.90 %	1008 98.43%
37.5	11 100%	532 97.25 %	179 98.35 %	198 98.01 %	75 91.46 %	995 97.16%
40	11 100%	543 99.26 %	179 98.35 %	195 96.53 %	79 96.35 %	1007 93.45%
42.5	11 100%	532 97.25 %	177 97.25 %	184 91.08 %	72 87.90 %	976 95.31%
45	11 100%	535 97.80 %	174 95.6%	133 65.84 %	73 89.02 %	926 90.42%

ii. Different class labels

Table 2: Observation table for class detection

Distance (m)	Correctly Detected Classes						Accuracy (275)
	Person (80)	Remote (45)	Cellphone (50)	Bed (40)	Cup (45)	Car (15)	
0.5	80 100%	35 100%	50 100%	30 75%	44 97.77%	_	249 95.76 %
0.75	79	40	48	40	41	15	262 95.27 %
1	80	40	44	40	38	15	257 93.45 %
1.25	80	38	41	40	36	15	250 90.9 %
1.5	80	38	40	40	35	15	248 90.18 %

On iterating the result of 6 different classes 30-40 times, it is found that the system is able to recognize a hundred classes. 100% accuracy is found in the detection of a person, remote, and cellphone with an accuracy of 95.769% at a distance of 0.5 m, 1 m, 1.25 m, and 1.5 m. However, the accuracy is found to decrease from 90.181 % with the increment in the distance

C. System's Accuracy and Precision

While analyzing the graph for average precision of street objects with respect to the number of images for the purpose

offine-tuning, the precision is found to increase non-linearly up to450 epochs, remaining constant thereafter. In this case, the precision is tested for cars, people, bicycles, buses and motorbikes. The highest precision is seen for the bus later but at first, a person has the greatest precision. Similarly, on plotting the accuracy with distance for a distance of 20-45 centimeters on a paper of A5 size, with the aerial font. 96 % accuracy is seen for angles in the range [-30°, 30°]. On plottingthe accuracy with distance, a non-linear decrement is seen from99% to 93%

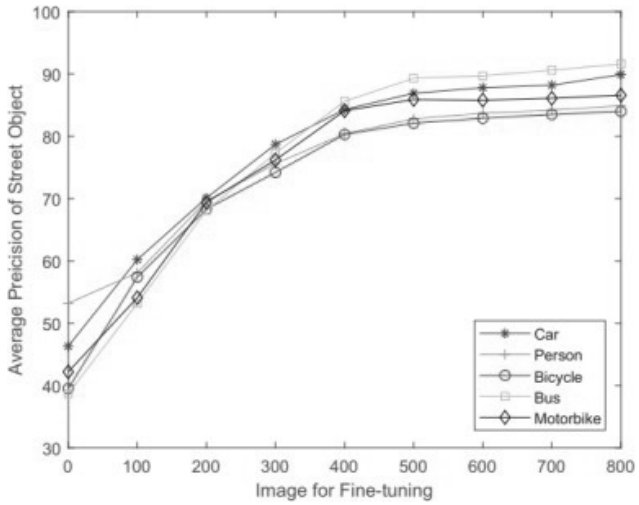

Figure 15: Average Precision of Street Objects

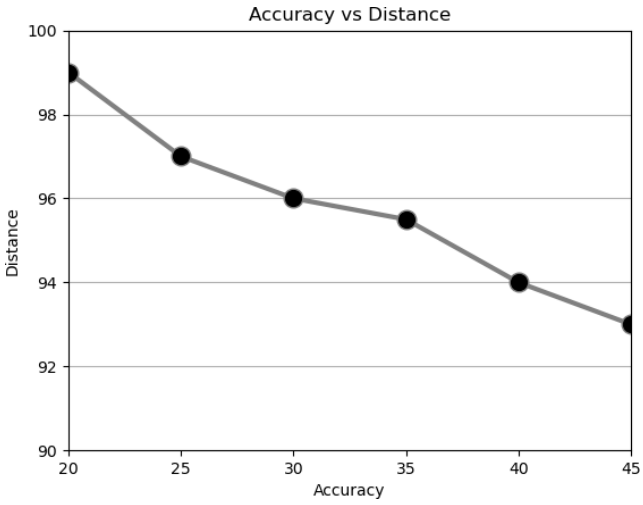

Figure 16: Accuracy vs Distance

IV. REFERENCES

[1] W. H. O. (WHO), "Blindness and vision impairment," WHO, 2021.

[2] S. Sarkar, G. Pansare, B. Patel, A. Gupta, A. Chauhan, R. Yadav and N. Battula, "Smart Reader for Visually Impaired Using Raspberry Pi," *International Conference on Innovations in Mechanical Sciences (ICIMS'21),* vol. 1132, pp. 2-7, 2021.

[3] N. Mahamkali, "OpenCV for Computer Vision Applications," *Proceedings of National Conference onBig Data and Cloud Computing (NCBDC'15),* pp. 2-6,2015.

[4] S. Venkateswarlu, K. D. B. K., J. K. R. Sastry and R. Rani, "Text to Speech Conversion," *Indian Journal ofScience and Technology,* pp. 1-3, 2016.

[5] T. Wood, "DeepAI," 1 August 2019. [Online]. Available: https://deepai.org/machine-learning-glossary-and- terms/ softmax-layer. [Accessed 20 October 2020].

[6] A. Harsur, "Voice Based Navigation System for Blind People Using Ultrasonic Sensor," *International Journal on Recent and Innovation Trends in Computing and Communication ,* vol. 3, no. 6, pp. 1-6, 2015.

[7] N. K, K. P D, P. Nivedha, P. B and L. G C, "Virtual Eye for Blind using IOT," *INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT),* vol. 8, no. 11, pp. 1-6, 2020.

[8] M. Mishra, "Towards Data Science," *ConvolutionalNeural Networks,* 27 August 2020.

[9] K. Sambasivarao, "Towards Data Science," 22 April2019. [Online]. Available: https://towardsdatascience.com/region-of-interest-pooling-f7c637f409af. [Accessed 13 January 2021].