STM JOURNALS

Journal of
Artificial Intelligence Research & Advances

ISSN: 2395-6720
Volume 8, Issue 2, 2021
DOI Journal: 10.37591/JoAIRA

http://computers.stmjournals.com/index.php?journal=JoAIRA&page=index

An Insight for Visually Impaired using AI Techniques

Natheera Beevi M.[1], Amal K. Santhosh[2,*], Lekshmi S.S.[2]

Abstract

We know that the life of blind people is very risky. They always need an assistance or another person for helping them. In this project we introduce AI spectacles for blinds, which will help them to find what is happening in front of them and they will be able to find their own things without any help. In this proposed system, we are using a real time object detection using YOLOv3 model. 'You only look once', or YOLO, is one of the fastest object detection algorithms out there. Despite the fact that it is not, at this point, the most exact item identification calculation, it is a generally excellent decision when you need continuous location, without loss of a lot of exactness. A route collaborator for the outwardly debilitated individuals utilizing object identification and text to discourse. The present study "An Insight for visually impaired using AI techniques", suggests the spectacles consisting of an input camera and a part of headset. The working of the AI spectacles is such that the objects and obstacles in front of blind person are captured by the camera, detected, recognized, converted into voice and then passed through headsets.

Keywords: YOLO v3, Google text to speech, Raspberry Pi, Darknet NN-Pack

INTRODUCTION

We know that the persons who are suffering from blindness problems always think that they cannot do anything without the help of another person. They always need a helper to fulfil their needs. Our main aim of this project is to help the blind persons and give them a confidence that they always can find things, walk to the road, learn new things etc. without depending on others. So, we introduce an AI assisted spectacles with voice feedback. In the proposed system, AI spectacles for blinds, the spectacles consist of an input camera and a part of headset [1]. The working of the AI spectacles is such that the objects and obstacles in front of blind person are captured by the camera, detected, recognized converted into voice and then passed through headsets. In this system, the first step will be the identification of object; for that purpose, we are using some object detection algorithm; here we are using YOLO v3. After the object is recognized and identified, the next performed operation is the conversion of the detected object into the voice format; Google text to speech is used to play mp3 files [2]. So, we use it here to convert the detected object name to audio file. For that we first install TTS of google in raspberry pi then create a Python file that has the conversion program and that program converts the text file to audio. The google TTS works only if there is internet support. So, we can successfully get the audio of what in front of visually impaired person [3].

***Author for Correspondence**
Amal K. Santhosh

[1]Professor, Department of Master of Computer Applications, Thangal Kunju Musaliar College of Engineering Kollam, Kerala Technological University, Kerala, India
[2]Student, Department of Master of Computer Applications, Thangal Kunju Musaliar College of Engineering Kollam, Kerala Technological University, Kerala, India

Received Date: May 11, 2021
Accepted Date: May 20, 2021
Published Date: June 20, 2021

Citation: Natheera Beevi M., Santhosh Amal K., Lekshmi S.S. An Insight for Visually Impaired using AI Techniques. Journal of Artificial Intelligence Research & Advances. 2021; 8(2): 1–8p.

COMPARISON WITH EXISTING SYSTEM

Existing System

Based on a survey that we have conducted among the public, we are able to know that the existing systems have certain drawbacks. Some of them use guide dogs; they cannot help in certain situations. Blind stick and white stick have limited features that it helps only in finding the obstacles.

They cannot help in some complex situations and use lot of sensors [4]. Examples like blind sticks have improper working as well as the route confusions present in the system which make them unpopular. Some of the other issues are listed below:

- Low speed and accuracy.
- No distance measurement and less level of localization.
- It works only in short distance.
- No classification.
- Not suitable for emergency situation.
- High battery consumption.

Proposed System

In order to provide the blind people hearable environment on all their basic needs, this project focuses on the field of AI based assistive devices for visually impaired people. It converts the visual data from image by using YOLO v3 object detector into a sound modality that will be appropriate for a blind user [5]. Therefore, the use of artificial intelligence is really helpful with the following features:

- High Speed and accuracy.
- Distance measurement and high level of localization.
- Easy to carry and use.
- Maintenances will be easy.
- Light weighted.
- Reduced battery consumption.
- More features are added.

DESIGN

Our protocol system, AI (Artificial Intelligence) spectacles project contains three main parts, a raspberry pi 4 model B, pi-camera and deep learning algorithm and artificial intelligence. When the person presses the button on device, the camera module starts to take a picture and analyze the image using YOLOv3 (Fastest object detection and reorganization algorithm) and detects anything in front of camera and gives voice feedback to the blind person [6].

Hardware Requirements

- Raspberry Pi 4 model B module,
- Bluetooth headset,
- Blind spectacles,
- Push buttons,
- USB-mini cam, and
- Power bank.

Raspberry Pi 4 Model B Module

The Raspberry Pi is microprocessor that can be used in many robotic projects. We are using Raspberry Pi for image and video processing as shown in Figure 1.

Figure 1. Image of raspberry pi 4 module B.

The Raspberry Pi 4 uses a Broadcom BCM2837 SoC with a 1.2 GHz 64 bit quad-core ARM Cortex-A53 processor, with 512 kB shared L2 cache. Raspberry Pi 4 Model B has 1 GB of RAM. The Raspberry Pi 4 (wireless) is equipped with 2.4 GHz Wi-Fi 802.11n (150 Mbit/s) and Bluetooth 4.1 (24 Mbit/s) based on Broadcom BCM43438 Full MAC chip with no official support for Monitor mode but implemented through unofficial firmware patching and the Pi 4 also has a 10/100 Ethernet port. The Raspberry Pi may be operated with any generic USB computer keyboard and mouse. Other peripherals can be attached to the various pins and connectors on the surface of the Raspberry Pi 0. The Raspberry Pi is connected to the system via a Rj45 cable. Raspberry pi consists of different slots for performing different functions.

Pi-Cam

The camera module consists of Sony IMX219 8-megapixel sensor (compared to the 5-megapixel Omni-Vision OV5647 sensor of the original camera). Camera module can take video and still photographs [7]. Pi cam has low price and is comfortable for suitable connection with raspberry pi as shown in Figure 2.

Here we have used the pi cam as it is good for easy connection and its size is very less as compared to USB-cameras.

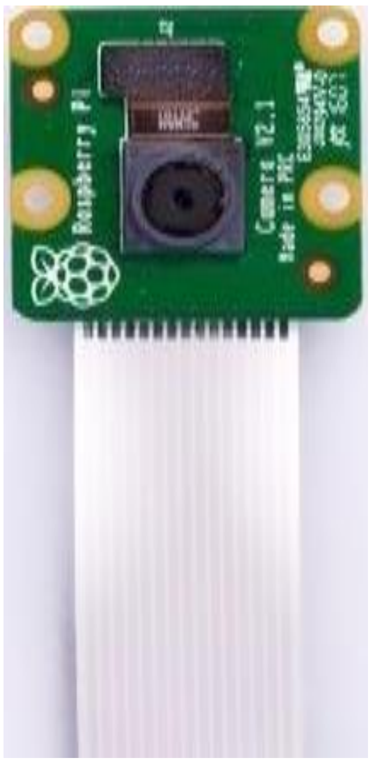


Figure 2. Figure of Raspberry pi cam.

Software Requirements

The following dependencies are required for detection, recognition and conversion of visual to audio:

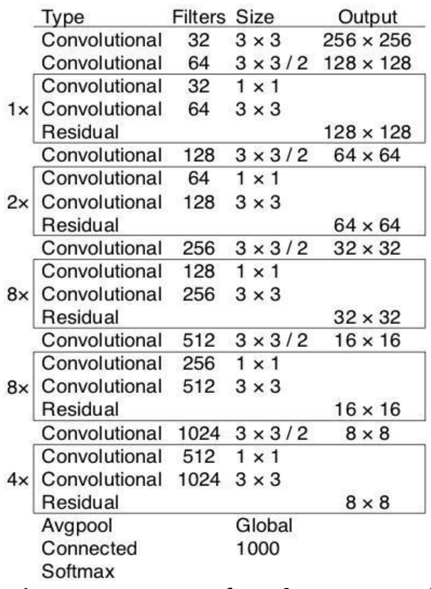
- Raspberry pi OS,
- Darknet NNPACK,
- YOLO v3 Algorithm,
- TensorFlow,
- GTTS (Google-Text to-Speech), and
- Python 3.6.

Darknet NNPACK

Darknet-NNPACK Darknet uses YOLO, a super-fast classifier which works better than other detections networks. For Raspberry Pi, we have used an optimized version of the original Darknet which is Darknet-NNPACK.

YOLO v3 Algorithm

Object detection is used in many fields of human life, for example, health and education, etc. As all these fields are growing rapidly, so, to match their requirements, one-stage models also need improvement [8]. The next advanced variant of YOLO is version 3 that uses logistic regression to compute the targetless score. It gives the score for all targets in each boundary box. YOLO v3 can give the multilabel classification because it uses a logistic classifier for each class in place of the SoftMax layer used in YOLO v2. YOLO v3 uses darknet 53. It has 53 layers of convolution [9]. These layers are more in-depth compared to darknet 19 used in YOLO v2. Darknet-53 contains mainly 3×3 and 1×1 filters as shown in Figure 3.



	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
	Convolutional	32	1 × 1	
1×	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
	Convolutional	64	1 × 1	
2×	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
	Convolutional	128	1 × 1	
8×	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
	Convolutional	256	1 × 1	
8×	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
	Convolutional	512	1 × 1	
4×	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

Figure 3. Figure of Darknet 53-Model.

YOLO v3 is the fastest object detection algorithm out there. In the YOLO v3 paper, the authors present a new, deeper architecture of a feature extractor called Darknet-53. As its name suggests, it contains 53 convolutional layers, each followed by a batch normalization layer and Leaky ReLU activation. This YOLO v3 helps in preventing the loss of low-level features often attributed to pooling [10].

DETAILED METHODOLOGY INCLUDING EXPERIMENTAL DESIGN

This system has different phases, the combined contribution of these phases makes excellent working. The following part explains each phase of them in detail as shown in Figure 4.

Phase-1

Push button-Raspberry pi: Initializing and stopping the execution on raspberry pi through its pins. configuration of these pins is in pin chart. Using this pin chart, we set the pins for two purposes. Here we are using 15th and 16th pin for start and stop. These pins are GPIO (Global Position Input Output) pins.

Phase-2

USB-Mini Cam-Raspberry pi: When the first push (GPI0-16) is pressed, the camera starts to capture the image in front of it, and this image is stored in a folder. Then this image is taken as the input image in YOLOv3 architecture Darknet + network; after object detection and recognition the Yolo’s output is send to a file; from that file we extract the number of object and its label name and save in another file.

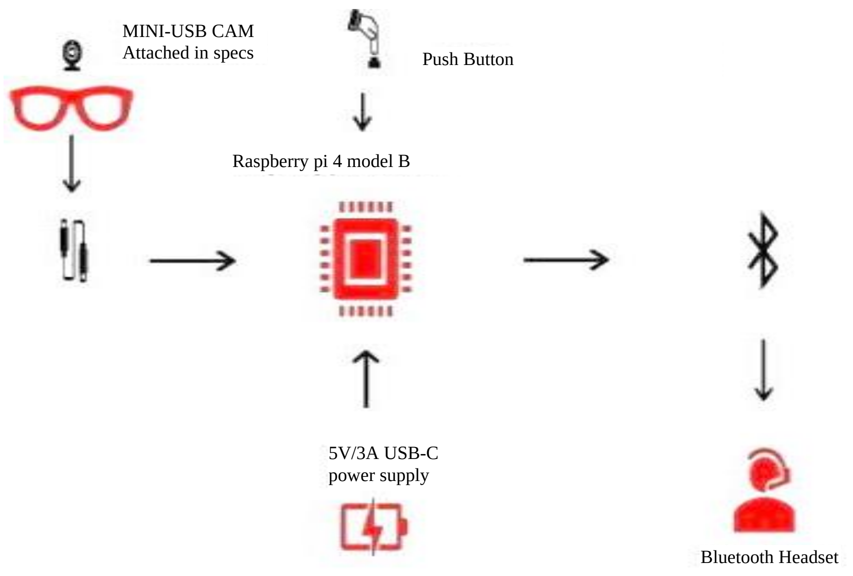


Figure 4. Design of the prototype device.

Phase-3

Raspberry pi-gTTS (Google to Text): The name that is extracted from Yolo’s output is saved in a file and that file is further sent to google text to speech API. Using gTTS package we can convert the names into speech.

Phase-4

gTTS-Speaker: The output audio file is then played by using gTTS and send the verbal cues to speaker or headphone attached.

RESULTS AND DISCUSSION

Here we explain the prototype details testing results:

We are demonstrating our prototype device in Figure 5 having a Pi-Cam attached with cam module on the microprocessor (Raspberry Pi). Push buttons are connected with the pin module for monitoring and controlling and 5 V/3 A USB-C power supply provided through power bank and a Bluetooth headset is connected for audibility in our system.

Testing is the major quality measure employed during the software development. After the coding phase, computer programs available are executed for testing purpose. Testing not only has to uncover errors introduced during coding, but also locates errors committed during the previous phase. Thus, the aim of testing is to uncover requirements, design or coding errors in the program are shown in Figure 6.

Testing

We have four stages of testing:

- Stage 1: After successful installation of raspberry pi OS on the raspberry pi, we connect raspberry pi-cam module on the processor. After enabling the camera module on pi, we check start the pi-cam and start capturing and streaming videos.
- Stage 2: After successful implementation of YOLOv3, its weights and models checking of real-time detection and recognition of images for that we start the video-camera module and

YOLOv3 model takes automatically as input and we get multilabel classification on each frame runs continuously. We planned in this project to detect and recognition on camera captured images of bottles, phones and persons that we input manually; for that, first we test taking some raw images from internet and give as inputted to the algorithm, we get multilabel classification in the output image.

- Stage 3: Then another important part is text to speech conversion; for that we use gTTS (Google Text- to- Speech). For the checking of working of gTTS API we create a text file with contents, then we create a python file that contain gTTS module for text conversion into speech. Internet is mandatory for working of gTTS, so we start Wi-Fi and connect to internet. Then we write some characters values in that text file and save, after that we call this text file in gTTS program, after running it we get successful output of that text conversion. We get the verbal cues on the wired/Bluetooth headphone jak connected.
- Stage 4: Because of deploying this project in raspberry pi, it has switches or push buttons to control or initialize the working. So, here we are using push buttons to start and stop the device. Here we are using two push buttons; one is for starting the device and another one is stopping the execution. Initializing and stopping the execution on raspberry pi is through its pins. configuration of these pins is in pin chart. Using this pin chart, we set the pins for two purposes. Here we are using 15th and 16th pin for start and stop. These pins are GPIO (Global Position Input Output) pins.

Optimized Approaches for Object Detections on Raspberry pi

Raspberry Pi 4 B which we are using for this project comes with 1.2 GHz quad core processor and 2 GB of RAM. If you have worked with detection systems like Dark net before you'll be familiar that running detections on a GPU gives us a significant speed boost. However, Raspberry Pi does not contain a powerful enough GPU that can run detection algorithms in a short time. To overcome this constraint, we have looked at two methods which can take advantage of Pi's multicore processor and we are able to get pretty instant results.

The Biggest Advantage of Using YOLO is its Superb Speed

YOLO v3 is a super speed object detection algorithm. YOLO v3 uses darknet-53 algorithm. So, it has 53 layers of convolutions. There are different techniques and tricks used in YOLO algorithm.

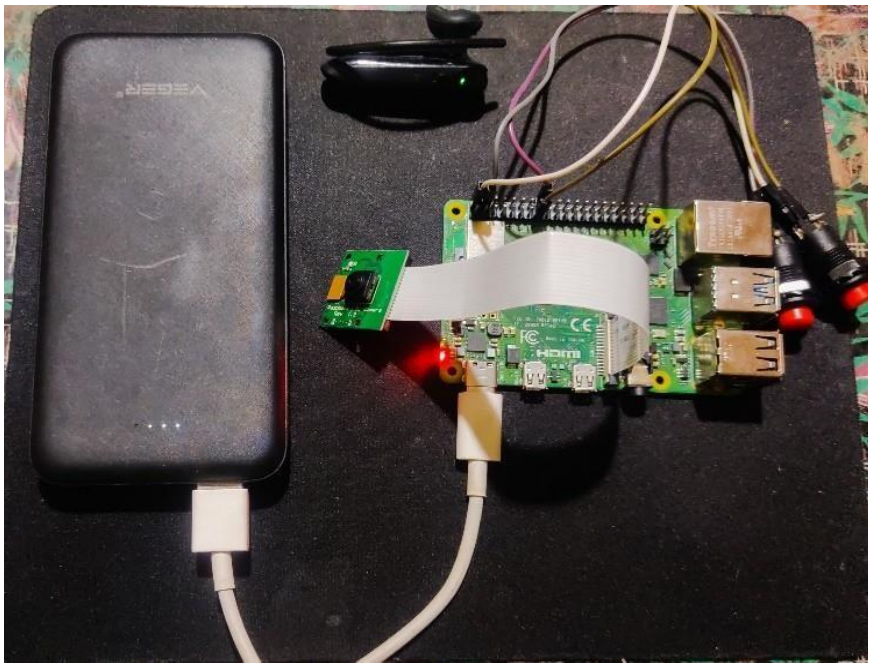
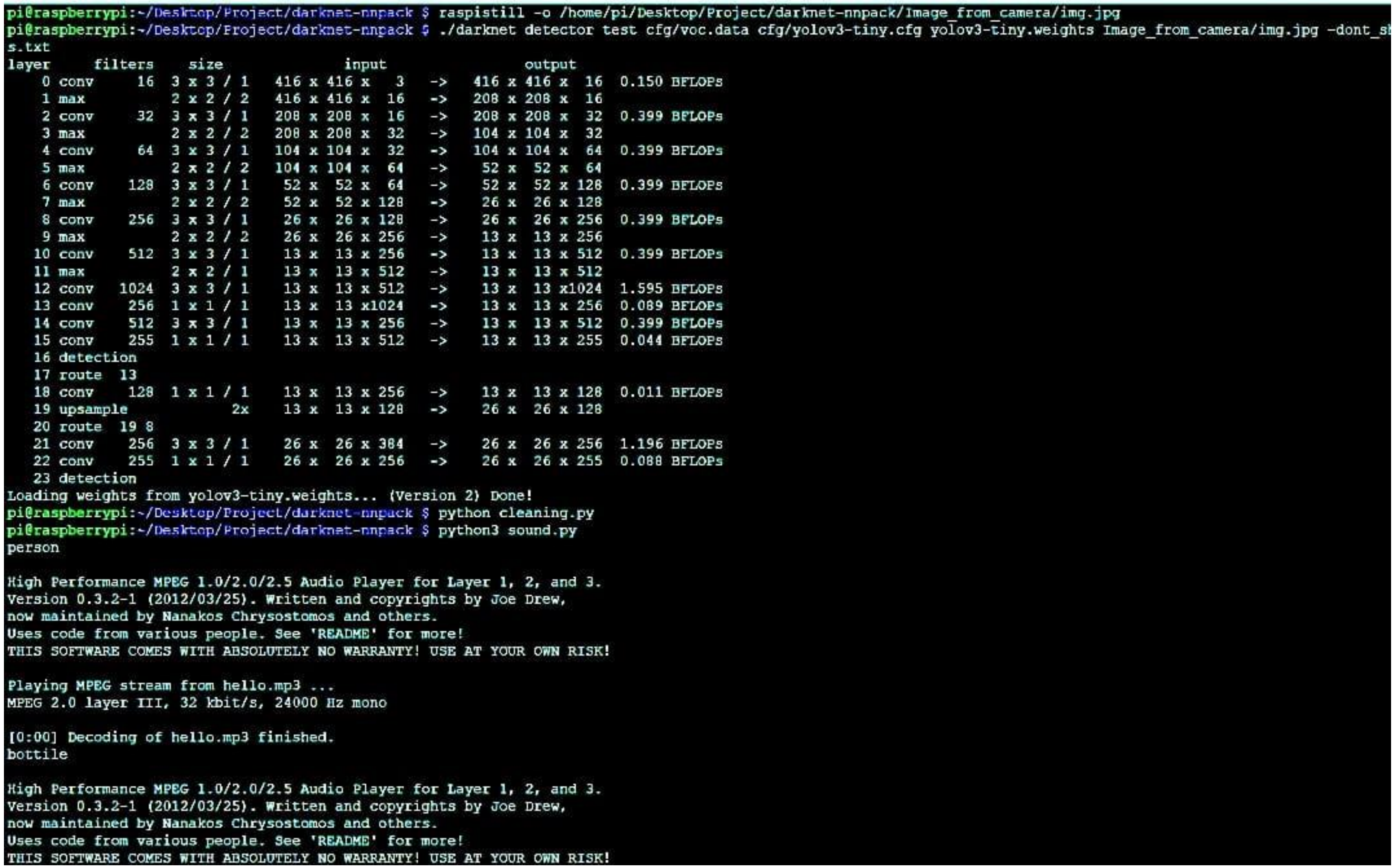


Figure 5. Figure of Pro type device.



```
pi@raspberrypi:~/Desktop/Project/darknet-nnpack $ raspistill -o /home/pi/Desktop/Project/darknet-nnpack/Image_from_camera/img.jpg
pi@raspberrypi:~/Desktop/Project/darknet-nnpack $ ./darknet detector test cfg/voc.data cfg/yolov3-tiny.cfg yolov3-tiny.weights Image_from_camera/img.jpg -dont_sh
s.txt
layer     filters    size              input                output
    0 conv     16  3 x 3 / 1   416 x 416 x   3   ->   416 x 416 x  16  0.150 BFLOPs
    1 max          2 x 2 / 2   416 x 416 x  16   ->   208 x 208 x  16
    2 conv     32  3 x 3 / 1   208 x 208 x  16   ->   208 x 208 x  32  0.399 BFLOPs
    3 max          2 x 2 / 2   208 x 208 x  32   ->   104 x 104 x  32
    4 conv     64  3 x 3 / 1   104 x 104 x  32   ->   104 x 104 x  64  0.399 BFLOPs
    5 max          2 x 2 / 2   104 x 104 x  64   ->    52 x  52 x  64
    6 conv    128  3 x 3 / 1    52 x  52 x  64   ->    52 x  52 x 128  0.399 BFLOPs
    7 max          2 x 2 / 2    52 x  52 x 128   ->    26 x  26 x 128
    8 conv    256  3 x 3 / 1    26 x  26 x 128   ->    26 x  26 x 256  0.399 BFLOPs
    9 max          2 x 2 / 2    26 x  26 x 256   ->    13 x  13 x 256
   10 conv    512  3 x 3 / 1    13 x  13 x 256   ->    13 x  13 x 512  0.399 BFLOPs
   11 max          2 x 2 / 1    13 x  13 x 512   ->    13 x  13 x 512
   12 conv   1024  3 x 3 / 1    13 x  13 x 512   ->    13 x  13 x1024  1.595 BFLOPs
   13 conv    256  1 x 1 / 1    13 x  13 x1024   ->    13 x  13 x 256  0.089 BFLOPs
   14 conv    512  3 x 3 / 1    13 x  13 x 256   ->    13 x  13 x 512  0.399 BFLOPs
   15 conv    255  1 x 1 / 1    13 x  13 x 512   ->    13 x  13 x 255  0.044 BFLOPs
   16 detection
   17 route  13
   18 conv    128  1 x 1 / 1    13 x  13 x 256   ->    13 x  13 x 128  0.011 BFLOPs
   19 upsample            2x    13 x  13 x 128   ->    26 x  26 x 128
   20 route  19 8
   21 conv    256  3 x 3 / 1    26 x  26 x 384   ->    26 x  26 x 256  1.196 BFLOPs
   22 conv    255  1 x 1 / 1    26 x  26 x 256   ->    26 x  26 x 255  0.088 BFLOPs
   23 detection
Loading weights from yolov3-tiny.weights... (Version 2) Done!
pi@raspberrypi:~/Desktop/Project/darknet-nnpack $ python cleaning.py
pi@raspberrypi:~/Desktop/Project/darknet-nnpack $ python3 sound.py
person

High Performance MPEG 1.0/2.0/2.5 Audio Player for Layer 1, 2, and 3.
Version 0.3.2-1 (2012/03/25). Written and copyrights by Joe Drew,
now maintained by Nanakos Chrysostomos and others.
Uses code from various people. See 'README' for more!
THIS SOFTWARE COMES WITH ABSOLUTELY NO WARRANTY! USE AT YOUR OWN RISK!

Playing MPEG stream from hello.mp3 ...
MPEG 2.0 layer III, 32 kbit/s, 24000 Hz mono

[0:00] Decoding of hello.mp3 finished.
bottile

High Performance MPEG 1.0/2.0/2.5 Audio Player for Layer 1, 2, and 3.
Version 0.3.2-1 (2012/03/25). Written and copyrights by Joe Drew,
now maintained by Nanakos Chrysostomos and others.
Uses code from various people. See 'README' for more!
THIS SOFTWARE COMES WITH ABSOLUTELY NO WARRANTY! USE AT YOUR OWN RISK!
```

Figure 6. YOLO v3 model output.

CONCLUSION AND FUTURE SCOPE

Here we introduced a prototype system of AI spectacles for blinds for making the smart environment for visually impaired people from all its current drawbacks; also, it is a helping system for the public. We have demonstrated how the system works, its features and design procedures. Our project will make a revolutionary change in the public transport system.

By using CNN image classifier, we predict the correct answer with more than 90% accuracy rates. By doing so, we have achieved the state-of-art result on the COCO dataset. We have also used the trained model with real time image and obtained the correct label. Yolo v3 is the fastest and most accurate object detection algorithm. Neural compute stick enables rapid deployment of deep neural network (DNN) at the edge. Advanced features like multilinguistic book reading can be implemented in future as the main project. In this project, the AI spectacles will definitely help blinds with smart features and make their world colorful.

Here are some ideas that should be used to plan our upcoming feature announcement:

- Fastest and more accurate detection.
- Book reading.
- AI assistance.
- Date, Time, Location.
- Distance measurement from objects.

REFERENCES

1. Galvez Reagan L, Bandala Argel A, *et al.* Object Detection Using Convolutional Neural Networks. 2018 IEEE Region 10 Conference (TENCON 2018); 2018 Oct 28–31; Jeju, Korea (South). New York: IEEE; 2019.
2. Ross Girshick, Jeff Donahue, *et al*. Rich feature hierarchies for accurate object detection and semantic segmentation. IEEE Conference on Computer Vision and Pattern Recognition; 2014 Jun 23–28; Columbus, OH, USA. New York: IEEE; 2014.
3. Ross Girshick. Fast R-CNN. IEEE International Conference on Computer Vision (ICCV); 2015 Dec 7–13; Santiago, Chile. New York: IEEE; 2016.
4. Shaoqing Ren, Kaiming He, *et al*. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Trans Pattern Anal Mach Intell. 2017; 39(6): 1137–1149.
5. Peilei Dong, Wenmin Wang. Better region proposals for pedestrian detection with R-CNN. Visual Communications and Image Processing (VCIP); 2016 Nov 27–30; Chengdu, China. New York: IEEE; 2017.
6. Joseph Redmon, Ali Farhadi. YOLO9000: Better, Faster, Stronger. IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017 Jul 21–26; Honolulu, HI, USA. New York: IEEE; 2017.
7. Johannes Günther, Pilarski Patrick M, *et al*. First Steps towards an Intelligent Laser Welding Architecture Using Deep Neural Networks and Reinforcement Learning. Procedia Technol. 2014; 15: 474–483.
8. Rasim Caner Çalik, Fatih Demirci M. Cifar-10 Image Classification with Convolutional Neural Networks for Embedded Systems. IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA); 2018 Oct 28–Nov 1; Aqaba, Jordan. New York: IEEE; 2019.
9. Alex Krizhevsky, Ilya Sutskever, et al. ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems (NIPS 2012). 2012. doi: 10.1201/9781420010749.
10. Mark Everingham, Luc Van Gool, *et al*. The Pascal Visual Object Classes (VOC) Challenge. Int J Comput Vis. 2010; 88: 303–338.